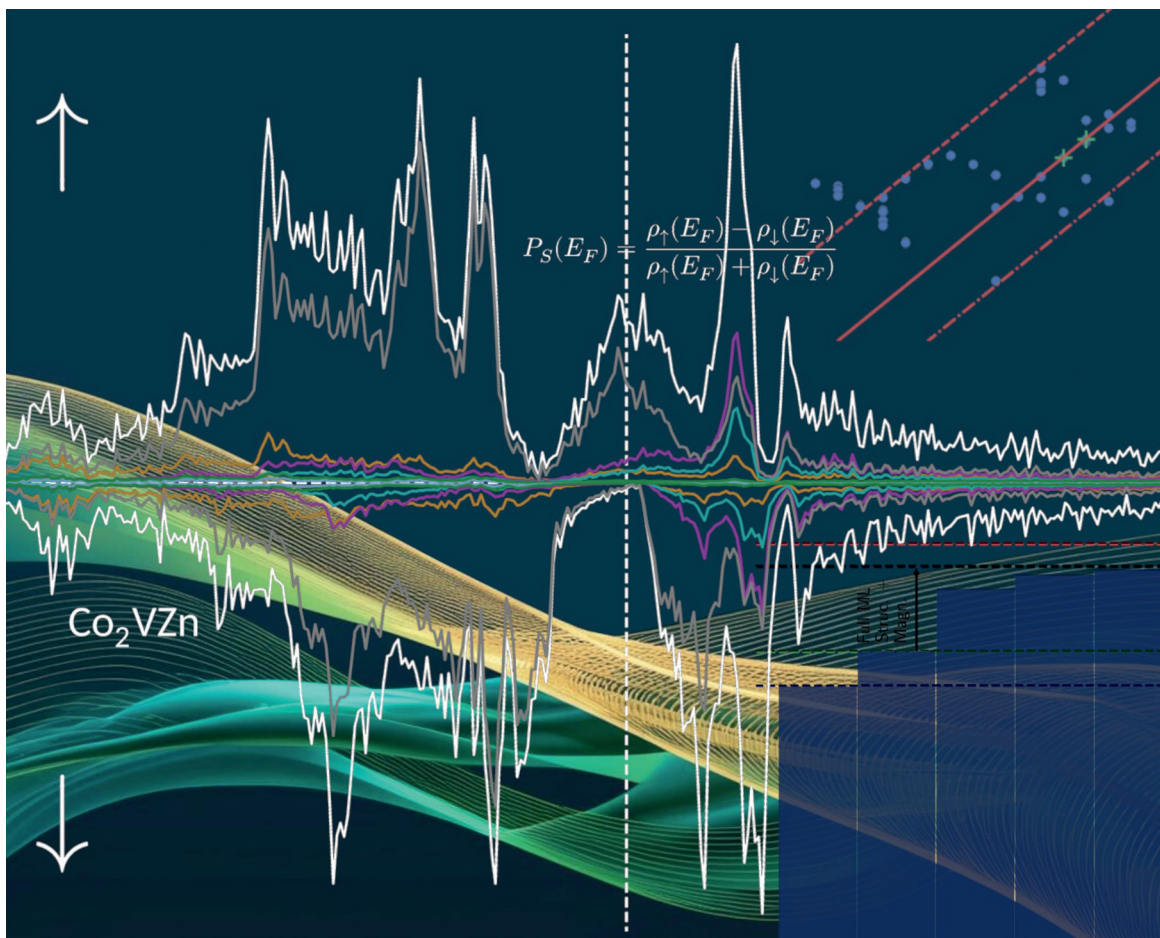




Prediction of Magnetic Materials for Energy and Information Combining Data-Analytics and First-Principles Theory

Robin Hilgers

Schlüsseltechnologien / Key Technologies

Band / Volume 288

ISBN 978-3-95806-795-0

Forschungszentrum Jülich GmbH
Peter Grünberg Institut (PGI)
Quanten-Theorie der Materialien (PGI-1)

Prediction of Magnetic Materials for Energy and Information Combining Data-Analytics and First-Principles Theory

Robin Hilgers

Schriften des Forschungszentrums Jülich
Reihe Schlüsseltechnologien / Key Technologies

Band / Volume 288

ISSN 1866-1807

ISBN 978-3-95806-795-0

Bibliografische Information der Deutschen Nationalbibliothek.
Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der
Deutschen Nationalbibliografie; detaillierte Bibliografische Daten
sind im Internet über <http://dnb.d-nb.de> abrufbar.

Herausgeber
und Vertrieb: Forschungszentrum Jülich GmbH
 Zentralbibliothek, Verlag
 52425 Jülich
 Tel.: +49 2461 61-5368
 Fax: +49 2461 61-6103
 zb-publikation@fz-juelich.de
 www.fz-juelich.de/zb

Umschlaggestaltung: Grafische Medien, Forschungszentrum Jülich GmbH

Druck: Grafische Medien, Forschungszentrum Jülich GmbH

Copyright: Forschungszentrum Jülich 2024

Schriften des Forschungszentrums Jülich
Reihe Schlüsseltechnologien / Key Technologies, Band / Volume 288

D 82 (Diss. RWTH Aachen University, 2024)

ISSN 1866-1807
ISBN 978-3-95806-795-0

Vollständig frei verfügbar über das Publikationsportal des Forschungszentrums Jülich (JuSER)
unter www.fz-juelich.de/zb/openaccess.



This is an Open Access publication distributed under the terms of the [Creative Commons Attribution License 4.0](https://creativecommons.org/licenses/by/4.0/),
which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

On ne fait jamais attention à ce qui a été
fait; on ne voit que ce qui reste à faire.

Marie Skłodowska Curie

Abstract

The essential role of magnetic materials in information technology and the corresponding energy consumption of data storage centers is crucially underestimated in modern society. Saving energy resources is the societal challenge of the 21st century. One of the leading scientific objectives is finding ways to reduce energy consumption and make resource usage more efficient. This thesis aims to shed light on possible contributions of materials science simulations towards a green IT transformation by providing workflows and best-practice guidelines for high-throughput materials screening tasks. An instance of such a screening task is the search for magnetic materials for the next generation of storage and data processing devices. However, as the simulation process itself is time-consuming, this thesis explores not only the material phase space but also the application opportunities for data science and machine learning (ML) in the material's property prediction process. As a prime example of a complex magnetic material property, which is a limiting quantity when it comes to methodological applicability, the critical temperature T_c of existing magnetic simulation data of Heusler alloys will be predicted using ML models. The capability and limitations of these models will be analyzed and discussed. It is shown that it is possible to extract physical relations and knowledge from trained ML models without any prior knowledge of the underlying physics and system mechanics. Whether a Heusler compound has a T_c high enough to be relevant for an application in magnetic data storage and processing devices could be predicted with over 90 % accuracy using lightweight ML model algorithms on typical materials science data set sizes. Beyond that, the phenomenon of near half-metallicity in Heusler compounds was examined, including the successful ML-based prediction of compounds displaying this property which were not known to be nearly half-metallic before ($\text{L2}_1\text{Co}_2\text{HfIn}$, $\text{XA Mn}_2\text{TaGe}$, and $\text{L2}_1\text{Co}_2\text{ScSn}$). This particular study used existing first-principles data of full and inverse Heusler compound's spin-polarized density of states, in order to screen publicly available structural and magnetic *ab initio* data for compounds exhibiting near half-metallic properties. The relations learned by the underlying ML models are discussed and compared to a known physical model. It was determined that ML models have the capability to extend and complement known physical models and relations when applied to existing (and potentially imperfect) data. Finally, large-scale high-throughput ultrathin film simulations of 3d transition metal layers on face-centered cubic noble metal substrates were performed to understand the magnetic properties of these magnetic multilayer films, which are predicted to represent well-suited host platforms for room temperature stable Skyrmions and hence are considered candidate materials for spintronics-based storage and data processing device applications. Tailored to high-throughput *ab initio* workflows, a scalable method—that increased the overall convergence rate from 64.8 % to 94.3 % and exhibited the potential to save up to 17 % of the computational time required, as well as to reduce the number of needed *ab initio* relaxation steps to relax a multilayer film system by up to 29 % in this systematic study, while being flexible enough also to be applicable to future use cases—using the integration of batch learning into high-throughput workflows, was developed. The use, restrictions, implementation, starting conditions, and benefits of ML-based techniques and explainable artificial intelligence are discussed in depth in this thesis.

Kurzzusammenfassung

Die wesentliche Rolle magnetischer Materialien in der Informationstechnologie und der damit verbundene Energieverbrauch von Datenspeichertzentren wird in der modernen Gesellschaft entscheidend unterschätzt. Die Einsparung von Energieressourcen ist die Herausforderung des 21. Jahrhunderts und eines der wichtigsten wissenschaftlichen Ziele ist es, Wege zu finden, den Energieverbrauch zu reduzieren und die Ressourcennutzung effizienter zu gestalten. Diese Arbeit zeigt mögliche Beiträge materialwissenschaftlicher Simulationen zu einer grünen IT-Transformation auf und stellt Arbeitsabläufe für Material-Screening-Aufgaben mit hohem Durchsatz vor wie z.B. die Suche nach magnetischen Materialien für die nächste Generation von Speicher- und Datenverarbeitungsgeräten. Da solche Simulationen jedoch rechenzeitintensiv sind, untersucht diese Arbeit auch die Anwendungsmöglichkeiten für Data Science und maschinelles Lernen (ML) bei der Vorhersage von Materialeigenschaften. Als Paradebeispiel für eine komplexe magnetische Materialeigenschaft, die eine limitierende Größe für die methodische Anwendbarkeit darstellt, wird die kritische Temperatur T_c aus vorhandenen magnetischen Simulationsdaten von Heusler-Legierungen mit Hilfe von ML-Modellen vorhergesagt, um im Anschluss die Möglichkeiten und Grenzen dieser Modelle zu analysieren und diskutieren. Es wird gezeigt, dass es möglich ist, physikalische Zusammenhänge und Erkenntnisse aus trainierten ML-Modellen zu extrahieren, ohne dass Vorkenntnisse der zugrunde liegenden Physik des Systems erforderlich sind. So war es möglich vorherzusagen, ob eine Heusler-Verbindung ein T_c hat, das hoch genug ist, um für eine Anwendung in magnetischen Speichergeräten relevant zu sein. Im präsentierten Beispiel war dies mit einer Genauigkeit von über 90 % möglich. Darüber hinaus wurde auch das Phänomen der Halbmetallizität in Heusler-Verbindungen untersucht, einschließlich der erfolgreichen ML-basierten Vorhersage von Materialien, die diese Eigenschaft annähernd aufweisen und von denen bisher nicht bekannt war, dass sie nahezu halbmetallisch sind ($L2_1$ Co₂HfIn, XA Mn₂TaGe und $L2_1$ Co₂ScSn). In dieser Arbeit wurden vorhandene *ab initio*-Daten von spinpolarisierten $L2_1$ und XA Heusler-Zustandsdichten verwendet, um Strukturen i.V.m. deren magnetischen Dichtefunktionaltheorie (DFT) Daten nach Verbindungen mit halbmetallischen Eigenschaften zu durchsuchen. Die von den ML-Modellen gelernten Relationen werden diskutiert und mit einem bekannten physikalischen Modell verglichen. Es wurde festgestellt, dass ML-Modelle in der Lage sind, bekannte physikalische Modelle zu erweitern und zu ergänzen, wenn sie auf vorhandene (und ggf. unvollkommene) Daten angewendet werden. Schließlich wurden Simulationen von ultradünnen Schichten aus 3d-Übergangsmetallen auf fcc-Edelmetallsubstraten durchgeführt, um die magnetischen Eigenschaften dieser Mehrschichtfilme zu verstehen, die als gut geeignete Plattformen für raumtemperaturstabile Skyrmionen vorhergesagt wurden und daher als Kandidatenmaterialien für Spintronik-basierte Speicheranwendungen gelten. In dieser Arbeit wurde eine auf solche DFT Studien mit hohem Durchsatz zugeschnittene, skalierbare Methode entwickelt, die nicht nur die Gesamtkonvergenzrate von 64.8 % auf 94.3 % verbesserte, sondern auch das Potenzial zeigte, bis zu 17 % der benötigten Rechenzeit einzusparen sowie die Anzahl der benötigten *ab initio*-Relaxationsschritte zur Relaxation solcher Filmsysteme um bis zu 29 % zu reduzieren, während sie flexibel genug ist, um auch für zukünftige Anwendungsfälle nutzbar zu sein—unter Verwendung der Integration von Batch-Learning in *ab initio*-Workflows. Der Einsatz, die Einschränkungen, die Implementierung, die Ausgangsbedingungen und die Vorteile von ML-basierten Techniken und erklärbarer künstlicher Intelligenz werden in dieser Arbeit eingehend diskutiert.

Contents

List of Figures	vii
List of Tables	xi
List of Algorithms	xiii
Abbreviations and Acronyms	xiv
1 Introduction	1
2 Methods	5
2.1 Data Driven Materials Design Process	6
2.1.1 Design Goal	6
2.1.2 Data Requirements	7
2.2 Density-Functional Theory	9
2.2.1 Hohenberg-Kohn Theorem	9
2.2.2 Kohn-Sham System	12
2.2.3 Full-Potential Linearized Augmented Plane-Wave Method	14
2.2.4 Magnetism within the FLAPW Method	17
2.2.5 Self-Consistent Field Loop	19
2.2.6 Korringa-Kohn-Rostoker Method	20
2.3 Selected Magnetic Material Properties & Phenomenons	21
2.3.1 Critical Temperature	22
2.3.2 Half-Metallicity	24
2.3.3 Requirements for Skyrmions	27
2.4 Automated Interactive Infrastructure and Database for Computational Science	28
2.4.1 Framework	28
2.4.2 Workflows	29
2.4.3 Data Management	32
2.4.4 Materials Science Data for Predictive Analysis Using Machine Learning	33
2.5 Machine Learning and Data Science	34
2.5.1 Machine Learning	34
2.5.2 Model Types	45
2.5.3 Gaining Physical Knowledge from Trained Models	57

2.5.4	Availability of Data	60
3	Results	63
3.1	Predicting Critical Temperatures for Materials Screening	64
3.1.1	Heusler Alloys	65
3.1.2	The Database	66
3.1.3	Data Wrangling	68
3.1.4	Machine Learning Modeling	75
3.1.5	Explainable Artificial Intelligence	91
3.2	Half-Metallicity in $L2_1$ and XA Heusler Alloys	98
3.2.1	Data	98
3.2.2	Modeling Goal	102
3.2.3	Screening Workflow	104
3.2.4	Model Predictions	106
3.3	Ultrathin Films of (001) oriented 3d Transition Metal Layers on fcc Noble Metal Substrates	121
3.3.1	Calculation & Film Setup	122
3.3.2	Workflow Outputs	126
3.3.3	DFT-Integrated Machine Learning	128
3.3.4	Explainable Artificial Intelligence	146
3.3.5	Data Analysis	149
4	Summary & Outlook	171
A	JuHemd LDA Data Overview	175
B	Bond Length Estimates	177
C	Guess Error Development	179
D	Substrate Elements Density of States	181
E	Modified Pair Interaction Algorithm	183
	Bibliography	185

List of Figures

2.1	Schematic depiction of the HK theorem	10
2.2	Schematic depiction of the KS approach.	13
2.3	Spatial subdivision of a crystal lattice using an augmented plane-wave basis.	15
2.4	Illustration of different classes of magnetic ordering in the case of collinear magnetism.	18
2.5	Typical SCF loop in an <i>ab initio</i> calculation.	19
2.6	Schematic depiction of the relaxation procedure in FLAPW.	20
2.7	Exemplary visualization of a 2D Ising model below and above the critical temperature.	22
2.8	Schematic depiction of the steps in the <i>ab initio</i> -based determination of the critical temperature.	23
2.9	Exemplary DOS for one conductor, one semiconductor and one insulator.	26
2.10	Depiction of different input option combinations required for the film relaxation workflow.	30
2.11	Flowchart of the individual steps which are performed in the film relaxation work chain.	31
2.12	Depiction of the individual steps in the structural setup procedure.	32
2.13	Depiction of the FAIR principles	33
2.14	Possible workflow from a technical requirement for a material exhibiting distinguished properties to predictive modeling.	33
2.15	Depiction of different ML tasks with short descriptions of each task's objective.	35
2.16	Schematic depiction of supervised, unsupervised, and reinforcement learning with the corresponding training inputs and model outputs.	36
2.17	Depiction of the field of AI related to ML and neural network models.	36
2.18	Depiction of the different stages and data usage in ML model optimization procedures.	38
2.19	Depiction of data splitting using the holdout method and a single validation set.	38
2.20	Depiction of the effect emerging of the bias and variance contribution.	40
2.21	Schematic depiction of a CV.	45
2.22	Overview over various ML models, including a rough classification of the models.	46
2.23	Exemplary depiction of a simple decision tree for a regression task.	48
2.24	Schematic depiction of the training process of a stacking model.	49
2.25	Depiction of the training process for a bootstrap aggregation modeling method.	50
2.26	Depiction of a boosting estimator training process.	51
2.27	Schematic depiction of a batch learning workflow.	62

3.1	Depiction of different Heusler alloy structures.	66
3.2	Structural phases contained in the JuHemd.	67
3.3	Distribution of critical temperatures extracted from JuHemd.	73
3.4	Distribution of atomic numbers in the Heusler compounds extracted from JuHemd per lattice site.	74
3.5	Distribution of critical temperatures of the complete and test data set.	76
3.6	Schematic depiction of the layered T_c determination, including ML.	77
3.7	Depiction of combinations to evaluate which arise in this study.	77
3.8	Depiction of the extra trees regression model predictions, compared to the test labels.	82
3.9	Depiction of the LASSO model predictions, compared to the test labels.	83
3.10	Kernel density estimations of the relative residuals of the extra trees regression and the LASSO model.	84
3.11	Depiction of the gradient boosting regression model, which was trained under exclusion of the DFT-originated features, predictions, compared to the test labels and residuals.	86
3.12	Confusion matrix of the predictions for models trained on the complete set of features.	89
3.13	Confusion matrix of the predictions for models trained on the reduced descriptor set.	90
3.14	SHAP values for the extra trees regression model trained on the full set of features.	92
3.15	Depiction of the compound's total moments and the sum of absolute magnetic moments, compared to the corresponding critical temperature.	93
3.16	Depiction of the magnetic moments at the first structural site and the corresponding absolute magnetic moments, compared to the corresponding critical temperature.	94
3.17	SHAP values for the gradient boosted trees regression model trained on the reduced set of features.	95
3.19	Relation between the total number of valence electrons of the compound to the critical temperature.	96
3.18	Relation between selected compound's fractional densities to the critical temperature.	97
3.20	Distribution of the spin-polarization at the Fermi level within the data set.	99
3.21	Atomic number distribution for the $L2_1$ Heusler phase structures within the training database.	100
3.22	Atomic number distribution for the XA Heusler phase structures within the training database.	101
3.23	Depiction of Slater-Pauling behavior on the screened data.	103
3.24	Atomic number distribution for the $L2_1$ Heusler phase structures in the screened structure's database.	105
3.25	Atomic number distribution for the XA inverse Heusler phase structures in the screened structure's database.	106
3.26	Spin resolved DOS of $L2_1$ Co_2VZn	113
3.27	SHAP summary plot for the 4 th predicting half-metallicity on the training database.	114
3.28	Overview of selected features and their relation to the corresponding SHAP values.	116
3.29	Depiction of the relation between the total magnetic moment and the corresponding SHAP value.	117
3.30	Relation between the features e_X^{val} and m_Y and the corresponding SHAP values.	118
3.31	Local explanations for selected compounds from the training set.	120
3.32	Schematic depiction of film structures resulting from the structural setup workflow.	125
3.33	Depiction of structural outputs of the Create-Magnetic-Film workflow.	127
3.34	Schematic depiction of the iterative DFT-integrated ML workflow.	128

3.35	Test scores of the models, which were used for ILD and magnetic moment input parameter prediction, based on the previously converged calculations.	130
3.36	Cumulative development of the number of converged film systems under an increasing number of DFT-integrated ML iterations.	134
3.37	Depiction of the counts of the successfully relaxed film structures by their corresponding elemental composition.	135
3.38	Depiction of the convergence rates per DFT-integrated ML iteration for films with two and single magnetic layers.	136
3.39	Depiction of total SCF iterations and the number of required relaxation steps to reach a force threshold of $10^{-3} \frac{\text{Ha}}{a_0}$ for each DFT-integrated ML iteration.	137
3.40	Depiction of the average number of total SCF iterations, the average number of relaxation steps to reach complete force convergence, and the average number of relaxation steps to reach a maximum absolute force of $10^{-3} \frac{\text{Ha}}{a_0}$	138
3.41	Comparison of the guessing MAE and the prediction MAE depending on the available training data.	139
3.42	Formal schematic depiction of the DFT-integrated ML scheme as an iterative procedure.	141
3.43	Posterior depiction of both the model's test MAE values and the MAE arising from the difference of the predicted ILDs and the predicted magnetic moments.	142
3.44	Depiction of the distribution of the difference of input and relaxed ILDs for each DFT-integrated ML iteration.	144
3.45	Depiction of the distribution of the difference of input and input magnetic layer moments for each DFT-integrated ML iteration.	145
3.46	Depiction of the average magnitudes of the SHAP values obtained using the same models as were used for the DFT-integrated ML procedure for each input feature and each predicted quantity.	147
3.47	Depiction of the predicted quantity-specific average magnitude of SHAP values in relation to the input features encoded using the corresponding neighboring degree.	148
3.48	Distribution of total magnetic film moments for each substrate element.	149
3.49	Distribution of the magnetic moments situated at the different magnetic layer sites of the relaxed film structure.	150
3.50	Depiction of the film's unit cell magnetic moments across the different DFT-integrated ML iterations.	152
3.51	Relative count of FM and AFM layerwise pair interaction types for induced and inducing layer elements.	154
3.52	Depiction of the total substrate moment distribution for each substrate element.	155
3.53	Depiction of the induced substrate magnetic moments for the substrate layers A, B, and C for each substrate element.	157
3.54	Development of the average induced magnetic moment magnitude, average induced magnetic moment for FM substrate orientations, and average induced magnetic moment for FM substrate orientations with positive moment direction for each substrate element and substrate layer.	158
3.55	Scatter plot of the relation of the different neighboring magnetic induced substrate moments.	160
3.56	Depiction of the symmetrized data, depicted in Figure 3.54, including the corresponding fit function, which describes the induced magnetic moment development inside the substrate.	162

3.57	Overview over the magnetic configurations which successfully relaxed throughout different iterations of DFT-integrated ML.	165
3.58	Transition matrix depiction the classification of the ML predicted magnetic states and the <i>ab initio</i> resulted magnetic configurations.	166
3.59	Heatmap of the pair interaction counts for the FM interaction type for the A and B layer pairs and the B and C layer pairs resolved for the different elemental pairs. . . .	167
3.60	Heatmap of the pair interaction counts for the AFM interaction type for the A and B layer pairs and the B and C layer pairs resolved for the different elemental pairs. . . .	168
3.61	Heatmap of the pair interaction counts for the non-magnetic interaction type for the A and B layer pairs and the B and C layer pairs resolved for the different elemental pairs.	169
A.1	Post-processing distribution of critical temperatures in the LDA data extracted from the JuHemd.	175
A.2	Distribution of atomic numbers in the compounds extracted from the JuHemd contained LDA results.	176
C.1	Development of the MAE on a test set resulting from the comparison of the initial ILD guesses and the relaxed <i>ab initio</i> ILDs for increasing amounts of available data. . .	179
C.2	Development of the MAE resulting from the comparison of the unanimously fixed initial magnetic moment guess and the resulting <i>ab initio</i> moments for increasing amounts of available data.	180
D.1	PDOS of the fcc noble metals from group 9 of the periodic table.	181
D.2	PDOS of the fcc noble metals from group 10 of the periodic table.	182
D.3	PDOS of the fcc noble metals from group 11 of the periodic table.	182

List of Tables

2.1	Advantages and disadvantages of the MAE.	42
2.2	Advantages and disadvantages of the MSE.	42
2.3	Advantages and disadvantages of the coefficient of determination.	42
2.4	Confusion matrix incorporating different ratios which could be used as metrics. . . .	43
3.1	Descriptors in the processed data with short description.	70
3.2	Characterization of extracted and constructed features from JuHemd and added atomic descriptors.	71
3.3	Overview of the models evaluated for the regression task.	79
3.4	Regression results on the entire feature set.	80
3.5	Regression results on the reduced feature set.	85
3.6	Overview of the models evaluated for the classification task.	88
3.7	Classification results on the entire descriptor set.	88
3.8	Classification results on the reduced descriptor set, excluding <i>ab initio</i> -based features. .	90
3.9	Overview of data set sizes per phase.	98
3.10	Overview of features in the database.	99
3.11	Overview of descriptors derived from those included in the data base.	100
3.12	Overview of screening data set sizes obtained from the Materials Project.	104
3.13	Overview of different model iterations used in this study.	107
3.14	Predictions of the 1 st model iteration.	108
3.15	Predictions of the 2 nd model iteration.	109
3.16	Predictions of the 3 rd model iteration.	110
3.17	Combined predictions of the different model iterations together with FLAPW validations.	112
3.18	Overview of the substrate elements EOS resulted lattice constants compared to a GGA FLAPW reference and experimental values.	123
3.19	Overview of quantities extracted from the AiiDA database for data analysis and DFT-integrated ML model training.	129
3.20	MAE of the initial guess methods used to relax the initial batch of structures. . . .	132
3.21	Development of successfully relaxed film structures per DFT-integrated ML iteration. .	132
3.22	Overview of the neighboring degree of the ML model's input features to the respective predicted quantities.	148

B.1	Bond length estimations of all magnetic element combinations within the given setup conditions.	177
B.2	Bond length estimations of all magnetic and substrate element combinations within the given setup conditions.	178
B.3	Bond length estimations of all substrate element combinations within the given setup conditions.	178

List of Algorithms

- 3.1 Classification of induced magnetic moments and coupling type determination. . . . 153
- 3.2 Magnetic state determination algorithm for the moment's orientation of the mag-
netic film layers. 164
- E.1 Modified algorithm computing the magnetic layer pair interaction types. 183

Abbreviations and Acronyms

AdaBoost	Adaptive Boosting
ADES	Automation, Data, Environment, and Sharing
AFM	Anti-Ferromagnetic
AI	Artificial Intelligence
AiiDA	Automated Interactive Infrastructure and Database for Computational Science
ANN	Artificial Neural Network
API	Application Programming Interface
BO	Bayesian Optimization
BZ	Brillouin Zone
CLAIX	Cluster Aix-la-Chapelle
CNN	Convolutional Neural Network
CV	Cross-Validation
DFT	Density-Functional Theory
DNF	Did Not Finish
DOS	Density of States
EOS	Equation of States
extra trees	Extremely Randomized Trees
FAIR	Findable, Accessible, Interoperable, and Reusable
fcc	Face-Centered Cubic
FLAPW	Full-Potential Linearized Augmented Plane-Wave
FM	Ferromagnetic
FPLO	Full-Potential Local-Orbital
GAN	Generative Adversarial Network
Gradient Boosting	Gradient Boosted Decision Trees
GF	Green's Function
GGA	Generalized Gradient Approximation
hcp	Hexagonal Close-Packed
HK	Hohenberg-Kohn
HM	Half-Metal
HPC	High Performance Computing
ICSD	Inorganic Crystal Structure Database

ILD	Interlayer Distance
JuHemd	Jülich-Heusler-magnetic-database
JURECA-DC	Jülich Research on Exascale Cluster Architectures Data Cen- tric
KKR	Korringa-Kohn-Rostoker
KS	Kohn-Sham
$L2_1$	Full Heusler Structure
LAPW	Linearized Augmented Plane-Wave
LARS	Least Angle Regression
LASSO	Least Absolute Shrinkage and Selection Operator
LCAO	Linear Combination of Atomic Orbitals
LDA	Local-Density Approximation
LLMs	Large Language Models
LMTO	Linear Muffin-tin Orbital
LOESS	Locally Estimated Scatterplot Smoothing
MAE	Mean Absolute Error
MC	Monte Carlo
ML	Machine Learning
MRAM	Magnetoresistive Random Access Memory
MSE	Mean Squared Error
OOS	Out-of-Sample
PBE	Perdew-Burke-Ernzerhof
PDOS	Projected Density of States
PID	Process Identifier
PW	Plane-Wave
PW-PAW	Plane-Wave Projector Augmented Wave
R^2	Coefficient of Determination
SCF	Self-Consistent Field
SHAP	SHapley Additive exPlanations
SOC	Spin-Orbit Coupling
spintronics	Spin Transport Electronics
SP	Slater-Pauling
SQL	Structured Query Language
UUID	Universally Unique Identifier
XA	Inverse Heusler Structure
XAI	Explainable Artificial Intelligence
XC	Exchange-Correlation
XGBoost	eXtreme Gradient Boosting

Introduction

The communication technology sector, which includes numerous data centers across the globe, is projected to account for over half of the world's total energy consumption by 2030. This development could also lead to a significant increase in greenhouse gas emissions, potentially contributing up to 23 % of the global total. It is projected that this trend will continue. Hence, the necessity for more efficient data-related technologies is obvious. [1]

Green IT

Possible solutions for this technological transition process include the development of novel data storage, transmission, and processing devices. Spin transport electronics (spintronics) based devices that incorporate concepts like neuromorphic computing, reservoir computing, and racetrack memory represent candidates for the in-demand application in future data utilization architectures. Unconventional computing approaches have the potential to develop into low-power alternatives to today's computing, data processing, and data storage technologies. [2–8]

Unconventional Computing

The aforementioned concepts are closely related to a magnetic phenomenon known as Skyrmions. A Skyrmion is a quasi-particle characterized by a 2-dimensional topological magnetic texture. The fact that Skyrmions can be manipulated (created, deleted, etc.) enables us to use them as information transmitters. In an experiment, Skyrmion-based neuromorphic computing was able to achieve nearly the same accuracies on a recognition task as software-based trained machine learning (ML) algorithms. [4] One of the present challenges is the search for materials that exhibit special magnetic properties, making them suitable host materials and enabling the emergence of Skyrmions at device operating temperatures.

Skyrmions and Spintronics

This thesis aims to provide structured workflows using high-throughput first-principles simulations and applying ML and data science methods that can be used in materials screening processes which are dedicated to determining and predicting promising material candidates with potential application in the field of e.g. future magnetic storage and data processing devices. This is possible as, given a particular application, the sought-after material properties are known by the technical requirements beforehand. [3] In the context of high-throughput first-principles studies, it is clear that the systematic electronic structure study of entire material classes using density-functional theory (DFT) represents a computationally intensive endeavor. Such investigations can easily consume millions of core hours. Beyond the computational time requirements, data management has represented an issue that prevented the large-scale success of systematic *ab initio* high-throughput studies. The development of the Automated Interactive Infrastructure and Database for Computational Science (AiiDA) framework [9, 10] and DFT code-specific plugins [11] simplifies research data handling and management.

High-Throughput First-Principles Simulations

High-Throughput Challenges

Beyond the complications inherent to high-throughput projects, there are also DFT-related peculiarities to address when conducting large-scale electronic structure simulations. This includes finding appropriate convergence parameters for the class of material being examined. Hence, before an actual systematic high-throughput study is conducted, various test calculations need to be performed in order to determine those convergence parameters representing a reasonable trade-off between methodological accuracy and computational efficiency.

Choice of
Convergence
Parameters

Another issue of high-throughput *ab initio* investigations is the fact that a DFT calculation can fail to reach self-consistency. This is especially true if the inputs (structure, initial magnetic moments, etc.) of the calculation setup have been chosen insufficiently. In DFT-based studies dedicated to individual structures it is possible to hand-tune the inputs. Of course, this is not feasible in a high-throughput setting. It is necessary, for large-scale studies, to automatize the generation of appropriate input parameters. The established approach to tackle the problem of input parameter optimization is trial-and-error-based. Unsurprisingly, this approach is neither systematic nor efficient regarding the computing time associated with such a guess for the input parameters. In this thesis, it was investigated whether this approach not only increased the overall success rate of the high-throughput study conducted during this thesis, but the optimized inputs also caused the subsequent DFT calculations to require fewer computational resources. The data used to train the ML models used for the prediction of the input parameters stems from the already converged results of the study itself. This usage of data beyond an analysis at the end of the large-scale electronic structure study is still relatively uncommon when it comes to high-throughput DFT investigations. Combining ML and data analytics methods with *ab initio* data has the capability to accelerate advances and knowledge discovery in the field of materials science and solid-state physics. [12] Consolidation of these methodologies is the approach that can be found in each results section of this thesis.

Input Parameters
in
High-Throughput
DFT

Gaining Value
from Data

ML and data science is a broad and continuously growing field. Hence, it is impossible to cover the related topics in their entirety in this thesis. Section 2.5 will discuss the embedding and application of ML and data science techniques and methods used in or related to this project and the scientific benefit added by combining *ab initio* simulation data with methods from these emerging fields.

Machine
Learning and
Data Science

Beyond high-throughput applications, this thesis demonstrates that ML can complement first-principles methods by enabling fast and efficient materials screening for Heusler compounds with an application-relevant magnetic critical temperature even before an *ab initio* calculation was performed. Usually, determining the critical temperature requires a two-step process combining a DFT and a subsequent Monte Carlo (MC) computation. Additionally, in a second project, it was possible to utilize full and inverse Heusler DOS data collected by collaborators to screen publicly available databases containing DFT results for half-metallic properties in the aforementioned compounds. This screening revealed three compounds for which the property of near half-metallicity was previously unknown. Both Heusler-related projects highlight how ML techniques can assist in screening for compounds with complex magnetic properties required to fulfill technological demands, even on relatively small (a few hundred data points) but typical materials science data sets.

Half-Metallicity
and Critical
Temperature

Using dedicated game-theory-based methods, it is possible to explain the prediction of an ML model and hence gain insights about hidden mechanisms and relations contained in the data and even extract physical insight from a trained model. This method is part of the continuously growing Explainable Artificial Intelligence (XAI) field. The capability to retrieve relations and correlations of physical quantities from trained ML models in the context of magnetic materials science data is investigated in this thesis. [13]

Explainable
Artificial
Intelligence

The underlying magnetic phenomena and the relevant material properties related to the phenomena will be discussed in-depth in section 2.3. The methodical requirements that were necessary to achieve the results described in this thesis range from the field of theoretical solid-state physics to applied data science techniques. Used frameworks, theoretical foundations of DFT, as well as the detailed orientation of what materials design and materials screening is and which challenges this process imposes will be introduced in the sections 2.1, 2.2, and 2.4.

During this thesis, three practical application cases have been examined. These cases are:

1. Predictive analysis of the Curie-Temperature (T_c) of magnetic Heusler alloys using data from the existing Jülich-Heusler-magnetic-database [14] (JuHemd). (Section 3.1)
2. Prediction of half-metallicity of Heusler alloys based on density of states (DOS) data as an application of ML-assisted materials screening. (Section 3.2)
3. Simulation setup, computation, analysis of results, and predictive analysis of 2-dimensional transition metal film systems on face-centered cubic (fcc) noble metal substrates and implementation of the DFT-integrated ML method. (Section 3.3)

All mentioned application cases follow essentially different approaches. However, each is a prime example of the applicability and versatility of ML techniques in materials design challenges, ranging from ML-assisted *ab initio* calculations and workflows to predictive materials and property discovery. At the same time, some materials design tasks can be mainly carried out by ML models (see e.g. [15]); other instances allow ML to assist *ab initio* methods in e.g. finding better initial starting points or filtering compounds, prior to a high-throughput study being conducted, for relevance. Hence, scientists can use ML to complement existing methods, which allows us to profit from the synergistic effects of both methods.

Methods

2.1	Data Driven Materials Design Process	6
2.1.1	Design Goal	6
2.1.2	Data Requirements	7
2.2	Density-Functional Theory	9
2.2.1	Hohenberg-Kohn Theorem	9
2.2.2	Kohn-Sham System	12
2.2.3	Full-Potential Linearized Augmented Plane-Wave Method	14
2.2.4	Magnetism within the FLAPW Method	17
2.2.5	Self-Consistent Field Loop	19
2.2.6	Korringa-Kohn-Rostoker Method	20
2.3	Selected Magnetic Material Properties & Phenomenons	21
2.3.1	Critical Temperature	22
2.3.2	Half-Metallicity	24
2.3.3	Requirements for Skyrmions	27
2.4	Automated Interactive Infrastructure and Database for Computational Science	28
2.4.1	Framework	28
2.4.2	Workflows	29
2.4.3	Data Management	32
2.4.4	Materials Science Data for Predictive Analysis Using Machine Learning	33
2.5	Machine Learning and Data Science	34
2.5.1	Machine Learning	34
2.5.2	Model Types	45
2.5.3	Gaining Physical Knowledge from Trained Models	57

Interdisciplinary

This section is designed to give the reader an overview of the applied methods and the terminology used during this thesis. It is clear that most readers do not have a background in data science/data analytics and at the same time have extensive knowledge about theoretical condensed matter physics, *ab initio* simulations, and magnetic phenomena. As it is impossible to cover the individual scientific disciplines in their entirety in a single thesis, a brief introductory section is dedicated to each field. Formulas in this thesis are given in Hartree atomic units.

2.1 Data Driven Materials Design Process

Specific Material Properties

Many modern technological devices (such as e.g. solid-state drives) rely heavily on components specifically engineered to fulfill a very special purpose. Examples of such devices include resistors, transistors, rechargeable batteries, sensors (including optical sensors), light-emitting diodes, and many more. For some of these components to be constructed operational, there are sought-after materials that have very specific properties such as e.g. superconductivity, semi-conductivity, magnetic stability, low electrical resistance or half-metallicity. Famous examples of in-demand materials are e.g. rare-earth metals, silicon, or conducting metals such as gold and copper.

It is a long way from a technical need for a material exhibiting a particular property to e.g. the final manufactured novel electrical component or device. Data acquired using first-principles simulations can assist in some cases in finding suitable materials for a given application.

2.1.1 Design Goal

Application Driven

At the beginning of the materials design process, there is a need for a material that meets special requirements in terms of the material's properties or the material's behavior in certain environmental conditions like e.g. high temperatures or external magnetic fields. These requirements are typically application-driven and highly specific to the use case from which they arise.

Workflows and Tools

Critical Temperature

This thesis aims to provide workflows and methods that can be used to search for materials that meet the requirements of novel magnetic storage devices as for e.g. the racetrack memory. A physically intuitive requirement for a novel magnetic storage device is that the intrinsic magnetism in the storage material needs to be stable, at least at room temperature conditions. If this was not the case, the storage would require permanent cooling or the information stored on the memory material would get erased, which contradicts the purpose of a storage device. This requirement can be interpreted as a need for a material-specific magnetic quantity, which is the critical temperature. A detailed discussion of the interpretation of this quantity can be found in section 2.3.1. However, taking the room temperature as a requirement for the critical temperature would be too short-sighted. Actually, temperatures in computers, data centers, server rooms, etc. are typically elevated compared to the typical room temperature. Also, given that a loss of stored information would significantly impact the material's applicability, the requirement should include a buffer zone. The size of this additional buffer zone should be determined by the following considerations:

Sufficiently High T_c

1. The stored information needs to be stored in a magnetic state which has a robust stability against altering or decay even over time

2. The accuracy of theoretical predictions of the critical temperature is limited, hence these inaccuracies should be included in the buffer zone *i.e.* *ab initio* methods are known to underestimate [16, 17] the critical temperature
3. While the critical temperature needs to be high enough, the buffer zone should not be chosen too large, as this can cause materials with potential application to be overlooked
4. Modeling on top of data with *e.g.* ML modeling can introduce additional errors on top of systematic errors already contained in the data

Taking into account these reasons and starting from room temperature of about 290 K a requirement for a critical temperature of about 400 — 500 K can be expressed for the use case of a novel reliable magnetic storage material which does not require external permanent cooling, including the discussed buffer zone. [18]

Generally speaking, more specific requirements emerge for more dedicated devices as *e.g.* the mentioned racetrack memory. More general requirements could include:

- A certain electric conductivity type
- Presence of Dzyaloshinskii–Moriya interaction [19, 20]
- Certain magnetic moment
- Certain magnetic ordering
- Magnetocrystalline anisotropy below/above a certain threshold [21, 22]

2.1.2 Data Requirements

As the name already suggests, in the data-driven materials design process, data plays an essential role. However, it is clear that the mere fact that data is accumulated during a research project is not enough to justify the label “Data-Driven”, but rather the continual use of data that has been accumulated to gain additional insights and subsequently accelerate materials discovery which includes the utilization of data-based modeling opportunities stemming from the field of ML. The data-driven scientific approach has been called the “fourth scientific paradigm” [23, 24] besides experiments, classical laws (mathematical expressions), and simulations. This additional paradigm relies heavily on research data analysis and pattern discovery. This also mandates that researchers validate the different paradigms against each other on a regular basis. [25]

Fourth Scientific
Paradigm

However, upholding established scientific standards and double-checking scientific results with different approaches also implies that we think about the data itself and our requirements in terms of conventions on how to ensure data quality and the critical use of data. This thought process gives rise to the following requirements:

Data Quality

Data Consistency

Data consistency represents the issue of comparability of different data points inside a single data set. Data inconsistencies can arise from simulation data if *e.g.* different data points were computed using different methods, approximations, code versions, or boundary conditions. Data inconsistencies also might arise after the data collection from defective data transmission and incorrect data processing. [26]

Inconsistencies

Data Uniqueness

Near Duplicates

Data uniqueness addresses the fact that data can sometimes contain duplicates. Duplicates in a data set can be wanted (e.g. when observations are counted and a certain event occurred multiple times) or unwanted. Suppose there are unwanted duplicates in the data set. In that case, removing them before any modeling is done is desirable, as they can reduce the performance of the modeling itself. However, while true duplicates are easy to remove, this can be more difficult for near duplicates depending on the nature of the data set. [26, 27]

Data Coverage

Representative

Data coverage addresses the issue of how much of the whole phase space the given data covers. Typically, data coverage is an issue for small data sets in a much larger context of possible other data points. Coverage can hinder us from drawing generalizable conclusions from data sets when our coverage is not representative enough for the whole phase space. [26]

Data Accuracy

External Sources

Data accuracy raises the question, “Are the individual data points correct?”. The answer to this question requires a comparison to some type of other external source or reference. It is obvious that data accuracy represents an issue for experimental data due to the error that arises inherently from the measurements. Despite that, data accuracy is also important when examining simulation data, even though the errors encountered in a simulation environment are more systematic as they naturally arise from approximations that are applied within the corresponding simulation method. Such systematic errors can emerge from something as simple as the discretization of a problem. However, considerations should be made before collecting the data if the chosen computational method is suitable to generate accurate data and information in the given research context. [26]

Suitable Methods

Source Trustworthiness

Integrity

In any case, in which the data used for a research project is not collected and used in one place, the integrity has to be questioned, and the trustworthiness of the source has to be evaluated and discussed. It is imperative not to use data for research projects that might have been altered or manipulated. [26]

Modeling Prerequisites

Predictive Value

Besides assuring data quality and integrity, we need to consider the data's predictive value. This is especially crucial when choosing descriptors from our data set to model a target quantity. There are different ways to explore the predictive capabilities of descriptors in relation to the target quantities beforehand, such as e.g. the statistical correlation and the ML-based predictive power score. [28, 29]

Additionally, features should not only be selected based on their capability to predict the target quantity, but also according to their availability. Assessing the added value that originates from predicting the target quantity using a given set of features is essential for an impact and meaningful research project.

Feature Selection

2.2 Density-Functional Theory¹

The investigation of materials in the materials design process requires tackling many-electron systems. Commonly this is done by setting up the Schrödinger equation as in equation (2.1). [31]

$$\hat{H}|\Psi\rangle = E|\Psi\rangle \quad (2.1)$$

Solving the Schrödinger equation leads to the state $|\Psi\rangle$ of the investigated system. Solving this equation analytically is impossible, even for a single helium atom with only two electrons *i.e.* three particles, without applying any approximations or restrictions. Hence, solving the Schrödinger equation for elemental He imposes challenges similar to the classical three-body problem. However, the dimension of the problem to solve with the quantum mechanical Schrödinger equation for N particles, assuming each particle can be described by k independent numbers—which might include spin, position, *etc.*—is given by k^N at a single point. Grid discretization of the problem on a $n \times n \times n$ 3-dimensional grid leads to a computational task with the dimension of $(n^3)^{kN}$ for the many-body system's state. This example illustrates that solving the Schrödinger equation analytically is computationally extremely expensive for a real-world problem. Hence, different approximations and approaches are used to reduce the computational effort in investigating quantum mechanical many-body systems. [32] The established approach to electronic structure computations is DFT. DFT describes many-body systems by their electron density. This reduces the degrees of freedom from at least three spatial degrees of freedom per electron—which is required to set up the corresponding wave function—to three spatial degrees of freedom in total. Hence, the electronic charge density is described as a function of the spatial coordinates $n(\mathbf{r})$. [33] Including the spin as an additional degree of freedom to describe magnetic quantum mechanical systems adds another dependent variable to each individual electron. The achieved shrinkage in degrees of freedom reduces the required memory to compute the many-body problem to a controllable size in comparison to computing the wave function directly. The possibility to describe a many-body system using the corresponding electronic charge density originates from the Hohenberg-Kohn (HK) theorem. [32, 34]

Schrödinger Equation

Classical Three-Body Problem

Density Functional Theory

Charge Density $n(\mathbf{r})$

2.2.1 Hohenberg-Kohn Theorem

The quantum mechanical many-body Hamiltonian $\hat{H}$ contains contributions of the kinetic energy of the many-body systems atom cores and electrons and the acting Coulomb interaction potential. The different contributions to the Hamiltonian are summarized in equation (2.2). By convention, in the following equations, capital letters as *e.g.* R and M denote the corresponding atom core's positions and masses, while the spatial positions of the electrons are denoted by the vectors $\mathbf{r}$. The corresponding indices distinguish the different atom cores and electrons.

¹ The content of the sections 2.2 to 2.2.3 is based on a corresponding part from my master's thesis and has been edited from this original version to match the scope of this thesis. [30]

$$\hat{H} = - \sum_i \frac{1}{2} \nabla_{\mathbf{r}_i}^2 - \sum_k \frac{1}{2M_k} \nabla_{\mathbf{R}_k}^2 + \frac{1}{2} \left(\sum_{i \neq j} \frac{1}{|\mathbf{r}_i - \mathbf{r}_j|} + \sum_{k \neq l} \frac{1}{|\mathbf{R}_k - \mathbf{R}_l|} \right) - \sum_{i,k} \frac{1}{|\mathbf{r}_i - \mathbf{R}_k|} \quad (2.2)$$

Assuming fixed nuclei reduces the core-core interaction as well as the kinetic energy contribution to a constant energy E_N . The electron-core Coulomb interaction part can be written as a potential depending on the electron positions using the same assumptions. This is summarized as a sum of $V_{\text{ext}}(\mathbf{r}_i)$ contributions in equation (2.3). Generally, the term $V_{\text{ext}}(\mathbf{r}_i)$ summarizes all contributions which act on the electron system besides the electrons themselves.

$$\hat{H}_e = - \sum_i \frac{1}{2} \nabla_{\mathbf{r}_i}^2 + \frac{1}{2} \sum_{i \neq j} \frac{1}{|\mathbf{r}_i - \mathbf{r}_j|} + \sum_i V_{\text{ext}}(\mathbf{r}_i) + E_N \quad (2.3)$$

Born-
Oppenheimer
Approximation

Rewriting the electronic Hamiltonian in the way presented in equation (2.3) by assuming fixed atom core positions, is also known as the Born-Oppenheimer approximation. [35] However, the external potential might also be spin-dependent when including other external potentials, such as an external magnetic field.

The Schrödinger equation (see equation (2.1)) allows us to compute the ground-state wave function Ψ_0 . Knowing the ground-state wave function—which represents the probabilistic distribution of the electrons in the many-body system in the ground state—the electronic ground-state charge density $n_0(\mathbf{r})$ can be derived. The HK theorem states that a unique external potential $V_{\text{ext}}(\mathbf{r})$ can be found given exclusively the ground-state density. This potential again allows the computation of the ground-state using the Schrödinger equation. The previously outlined relations and dependencies are visualized in Figure 2.1. [34]

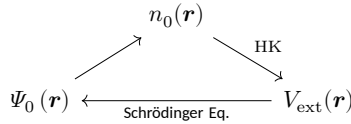


Figure 2.1: Schematic depiction of the HK theorem and the corresponding dependencies. The electronic ground-state density is calculated from the ground-state wave function. The HK theorem's consequence is that a potential exists that determines the ground-state wave function is labeled accordingly. This potential is unique up to a constant shift. This depiction was adapted from [30].

The first part of the HK theorem is given by theorem 1.

Theorem 1. “For any system of interacting particles in an external potential $V_{\text{ext}}(\mathbf{r})$, the potential, $V_{\text{ext}}(\mathbf{r})$ is determined uniquely, except for a constant, by the ground-state particle density $n_0(\mathbf{r})$.” Theorem taken from [36].

Hence, according to the HK theorem, the ground-state density determines all properties of the many-body system as the ground-state density fully determines the Hamiltonian besides a constant this also holds for excited states. [36] This is, of course, also true for the energy of the many-body system, as the second part of the HK theorem states. This second part is included as theorem 2.

$n_0(\mathbf{r})$
Determines the
Many-Body
System

Theorem 2. “A universal functional for the energy $E_{\text{HK}}[n(\mathbf{r})]$ in terms of the density $n(\mathbf{r})$ can be defined, valid for any external potential $V_{\text{ext}}(\mathbf{r})$. For any particular $V_{\text{ext}}(\mathbf{r})$, the exact ground-state energy of the system is the global minimum value of this functional [while keeping the number of interacting particles constant [34]], and the density $n(\mathbf{r})$ that minimizes the functional is the exact ground-state density $n_0(\mathbf{r})$.” Theorem taken from [36].

Therefore, knowing that the energy functional $E_{\text{HK}}[n(\mathbf{r})]$ determines the exact ground-state energy as well as the exact ground-state density, this energy functional can be defined as a sum of multiple functionals as shown in equation (2.4).

$E_{\text{HK}}[n(\mathbf{r})]$

$$E_{\text{HK}}[n(\mathbf{r})] = F_{\text{HK}}[n(\mathbf{r})] + \int V_{\text{ext}}(\mathbf{r})n(\mathbf{r}) d^3r + E_N \quad (2.4)$$

The functional $F_{\text{HK}}[n(\mathbf{r})]$ in equation (2.4) contains the electron-electron interactions as well as the electronic kinetic energy contributions as shown in equation (2.5).

$$F_{\text{HK}}[n(\mathbf{r})] = \langle \psi[n(\mathbf{r})] | \hat{T} + \hat{V}_{ee} | \psi[n(\mathbf{r})] \rangle \quad (2.5)$$

The functional $E_{\text{HK}}[n(\mathbf{r})]$ is commonly called the HK energy functional. [36]

As the proof of the HK theorem is well documented in the literature (see e.g. [36–38]) and goes beyond the scope of this thesis, it is not provided here. The HK theorem is not limited to non-magnetic and non-spin-polarized systems. However, in practice, the charge density is represented by individual spin contributions. In the case of collinear magnetic setups, the individual electronic densities for the spin-up ($\uparrow$) and spin-down ($\downarrow$) combine to a total density as shown in equation (2.6).

Collinear
Spin-Directions

$$n(\mathbf{r}) = n(\mathbf{r}, \uparrow) + n(\mathbf{r}, \downarrow) \quad (2.6)$$

By defining the electronic charge density in this way, only collinear magnetic configurations along a single magnetic axis are allowed. This restriction can be lifted as discussed in [30, 39]. Furthermore, this additional spin degree of freedom and e.g. potential external magnetic fields can be included by defining the magnetization density as shown in equation (2.7).

$$m(\mathbf{r}) = n(\mathbf{r}, \uparrow) - n(\mathbf{r}, \downarrow) \quad (2.7)$$

Including the magnetization density into the energy functional, one obtains the functional from equation (2.8).

$$E_{\text{HK}}[n(\mathbf{r}), m(\mathbf{r})] = E_{\text{HK}}[n(\mathbf{r})] + \int B_{\text{ext}}(\mathbf{r})m(\mathbf{r}) d^3r \quad (2.8)$$

Assuming that an external magnetic field does act on the spins but not on the orbital motion of the electrons, it is possible to prove the HK theorem also for this functional. The additional expression of the contribution originating from the external magnetic field is very similar to the external potential and particle density contribution. Hence, the proof of the HK theorem in the presence of an external magnetic field is analogous to the regular case and can, therefore, be extended to magnetic systems. [37] However, the HK theorem states that an external potential $V_{\text{ext}}(\mathbf{r})$ which determines the ground-state of the quantum mechanical many-body system exists but not how to obtain it for practical calculations.

Hohenberg-Kohn
for Magnetic
Systems

2.2.2 Kohn-Sham System

Non-Interacting Auxiliary System

Approximations in many-body problems in physics often depend on the properties of the individual problem. The Kohn-Sham (KS) approach does not rely on any system-specific properties. Kohn and Sham first assumed that the exact ground-state density of an arbitrary system can be represented as the ground-state density of an auxiliary system of the non-interacting particles. The auxiliary system is represented by a Hamiltonian, which contains only kinetic operator contributions and an effective local potential $V_{\text{eff}}[n(\mathbf{r})](\mathbf{r})$ but no interaction terms. Hence, the Hamiltonian of the auxiliary system is given by equation (2.9).

$$\hat{H}_{\text{aux}} = -\frac{1}{2}\nabla^2 + V_{\text{eff}}(\mathbf{r}) \quad (2.9)$$

The effective potential $V_{\text{eff}}[n(\mathbf{r})](\mathbf{r})$ contains the external potential $V_{\text{ext}}(\mathbf{r})$ from equation (2.3), the exchange-correlation (XC) potential, and the Hartree potential contribution. In the auxiliary system of N independent electrons, the states are occupied according to the Pauli exclusion principle. The particle density of the corresponding system is given by equation (2.10) while $\psi_i(\mathbf{r})$ denotes the single electron states wave functions. [40]

$$n(\mathbf{r}) = \sum_i^N |\psi_i(\mathbf{r})|^2 \quad (2.10)$$

KS Energy Functional

Summarizing the KS auxiliary system contributions, one obtains the KS energy functional from equation (2.11) for the many-body system. This functional expresses the density-dependent energy functional of the many-body system based on the KS auxiliary system. [40]

$$\begin{aligned} E_{\text{KS}}[n(\mathbf{r})] = & T_s[n(\mathbf{r})] + \int V_{\text{ext}}(\mathbf{r})n(\mathbf{r}) d^3r \\ & + E_{\text{Hart}}[n(\mathbf{r})] + E_{\text{xc}}[n(\mathbf{r})] + E_{\text{N}} \end{aligned} \quad (2.11)$$

The kinetic energy operator of the non-interacting electron system as a result of this simplifies to the expression in equation (2.12) which also includes the density-based representation of the Hartree energy contribution from equation (2.11).

$$T_s = \frac{1}{2} \sum_{i=1}^N \int |\nabla \psi_i(\mathbf{r})|^2 d^3r \quad E_{\text{Hart}}[n(\mathbf{r})] = \frac{1}{2} \int \int \frac{n(\mathbf{r})n(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} d^3r d^3r' \quad (2.12)$$

The E_{xc} term from equation (2.11) contains the complex many-body contributions of exchange and correlation and is—in practice—subject to approximations as it is not analytically known. Kohn and Sham defined an effective Hamiltonian as included in equation (2.13).

$$\hat{H}_{\text{KS}} = -\frac{1}{2}\nabla^2 + V_{\text{eff}}[n(\mathbf{r})](\mathbf{r}) \quad (2.13)$$

For this Hamiltonian, the effective KS potential $V_{\text{eff}}[n(\mathbf{r})](\mathbf{r})$ consists of multiple contributions, depending on both the charge density $n(\mathbf{r})$ and explicitly on the vector $\mathbf{r}$, as shown in equation (2.14), while equation (2.15) represents the explicit contribution of the Hartree potential.

$$V_{\text{eff}}[n(\mathbf{r})](\mathbf{r}) = V_{\text{ext}}(\mathbf{r}) + \frac{\delta E_{\text{Hart}}[n(\mathbf{r})]}{\delta n(\mathbf{r})} + \frac{\delta E_{\text{xc}}[n(\mathbf{r})]}{\delta n(\mathbf{r})} \quad (2.14)$$

$$V_{\text{Hart}}[n(\mathbf{r})] = \frac{\delta E_{\text{Hart}}[n(\mathbf{r})]}{\delta n(\mathbf{r})} = \int \frac{n(\mathbf{r}')}{|\mathbf{r} - \mathbf{r}'|} d^3r' \quad (2.15)$$

The resulting effective Hamiltonian can be used to solve the Schrödinger-like equations (2.16) which are called KS equations and represent single particle equations. [40]

$$(H_{\text{KS}} - \epsilon_i)\psi_i(\mathbf{r}) = 0 \quad (2.16)$$

The solution to this eigenvalue problem in principle determines all properties of the many-body system. In addition to the connection between external potential and ground state established by the HK theorem, the KS approach links the real-world many-body system to the auxiliary system of independent electrons. This additional connection is depicted in Figure 2.2.

The depicted approach is an exact way to tackle the many-body Hamiltonian, besides the necessary approximation of the XC potential. Different approximations for the XC potential include local-density approximation (LDA), generalized gradient approximation (GGA), and hybrid functional-based methods. With the KS approach, it is possible to compute the many-body particle density self-consistently using the non-interacting particle system with the effective KS potential V_{eff} . [36]

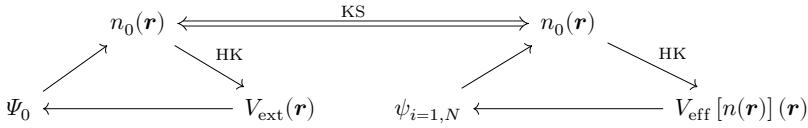


Figure 2.2: Schematic depiction of the KS approach to many-body systems. Compared to Figure 2.1, there is an additional auxiliary system (right) that is connected to the many-body system (left) via the KS approach, which links the many-body ground state density to a non-interacting electron gas system with the same electronic density. The external potential from Figure 2.1 has been substituted by the density-dependent effective KS potential in the KS approach. This figure was adapted from [30].

KS and Spin DFT

Using two-component Pauli wave functions, the spin as an additional degree of freedom—enabling the computation of magnetic systems—can be included as shown in equation (2.17).

$$\psi_i(\mathbf{r}) = \begin{pmatrix} \psi_{i,\uparrow}(\mathbf{r}) \\ \psi_{i,\downarrow}(\mathbf{r}) \end{pmatrix} \quad (2.17)$$

Given this representation the particle density and the magnetization density are redefined as in the equations with the number (2.18) which are generalizations of the expressions in the equations (2.6) and (2.7).

$$n(\mathbf{r}) = \sum_{i=1}^N |\psi_i(\mathbf{r})|^2 \quad \mathbf{m}(\mathbf{r}) = \sum_{i=1}^N \psi_i^*(\mathbf{r}) \hat{\sigma} \psi_i(\mathbf{r}) \quad (2.18)$$

In this case $\hat{\sigma}$ denotes the vector of Pauli spin matrices given by the expressions in equation (2.19).

$$\hat{\sigma} = \frac{1}{2} \left(\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \right)^T \quad (2.19)$$

Single Particle
Equations

Exchange-
Correlation
Potential
Approximation

Magnetic
Systems

Including external magnetic fields, the more general spin-dependent KS Hamiltonian shown in equation (2.20) as well as the corresponding single particle equations result analogously for magnetic many-body systems. The following depiction of magnetism within DFT restricts to the description of non-collinear magnetism.

$$\hat{H}_{\text{KS}}^{\sigma} = -\frac{1}{2}\nabla^2 + V_{\text{eff}}[n(\mathbf{r})](\mathbf{r}) + \mu_B \hat{\sigma}_z \cdot B_{\text{eff}}[n(\mathbf{r})](\mathbf{r}) \quad (H_{\text{KS}}^{\sigma} - \epsilon_{i,\sigma})\psi_{i,\sigma}(\mathbf{r}) = 0 \quad (2.20)$$

Analogous to the KS effective potential, the effective magnetic field $B_{\text{eff}}(\mathbf{r})$ and the corresponding XC field contribution are given by the expression in the equation number 2.21.

$$B_{\text{eff}}[n(\mathbf{r})](\mathbf{r}) = B_{\text{xc}}[n(\mathbf{r})](\mathbf{r}) + B_{\text{ext}}(\mathbf{r}) \quad B_{\text{xc}}[n(\mathbf{r})](\mathbf{r}) = \frac{\delta E[n(\mathbf{r}), \mathbf{m}(\mathbf{r})]}{\delta \mathbf{m}(\mathbf{r})} \quad (2.21)$$

In the absence of external magnetic fields, the XC field is the only spin-dependent contribution to the effective KS Hamiltonian. The previously introduced formalism allows a calculation of electronic particle densities given certain spin directions (e.g. spin-up or spin-down density) similar to as they have been used in equation (2.6) as shown in equation (2.22). [39]

$$n_{\sigma} = \sum_{i=1}^N |\psi_{i,\sigma}(\mathbf{r})|^2 \quad (2.22)$$

2.2.3 Full-Potential Linearized Augmented Plane-Wave Method

Different methodologies have been developed from the theoretical foundations introduced in the previous sections. While some of them include e.g. precomputed pseudopotentials and plane-wave (PW) basis sets to model the system's potential, there is a more dedicated method that uses the full physical potential and a different augmented basis set close the atoms center positions. This method is called the Full-Potential Linearized Augmented Plane-Wave Method (FLAPW) method, which itself is based on the linearized augmented plane-wave (LAPW) method [41, 42] but combines the augmented basis set with the full physical potential to model the Hamiltonian. The mentioned augmented basis set describes an atomic-like modeled orbital basis set for each atom in the system. A spatial decomposition into muffin-tin spheres, representing the region around the individual atom positions, and the interstitial region, representing the space between the muffin-tin spheres, is performed. Within the muffin-tin spheres, a basis set that consists of atomic-like orbitals is used to construct the wave function, while in the interstitial region, a plane-wave basis is used. Outside these atomic orbitals—which are called muffin-tin spheres—there is an interstitial region, which itself is modeled using a plane-wave basis set. The core application of the FLAPW method is the electronic structure computation of solid crystal structures. Periodic boundary conditions of the lattice in the real space in solid crystals allow a mapping to the Brillouin zone (BZ) in the reciprocal space of the Bloch vectors $\mathbf{k}$ in which the wave function is expanded. [43] Besides the Bloch vectors, wave functions in crystals typically depend on the band index and spin component. The band index denotes that a wave function can be assigned to a certain energy band of the crystal. [44, 45]

LAPW Basis

Applying an augmented plane-wave basis set divides the space into atomic muffin-tin spheres and interstitial regions. This spatial subdivision used in the FLAPW method is depicted in Figure 2.3. An additional spatial region needs to be considered in the case of thin-film systems. For film systems, a vacuum region is added outside the overall film thickness. [46]

Full Potential
Method

Atomic-Like
Orbitals in the
Muffin-Tin
Spheres

Electronic
Structure

The spatial decomposition allows the computing of local and non-local potential expressions in the different regions. This avoids the necessity to approximate the potential close to the atomic nuclei using pseudo-potentials close to the nuclei. [44]

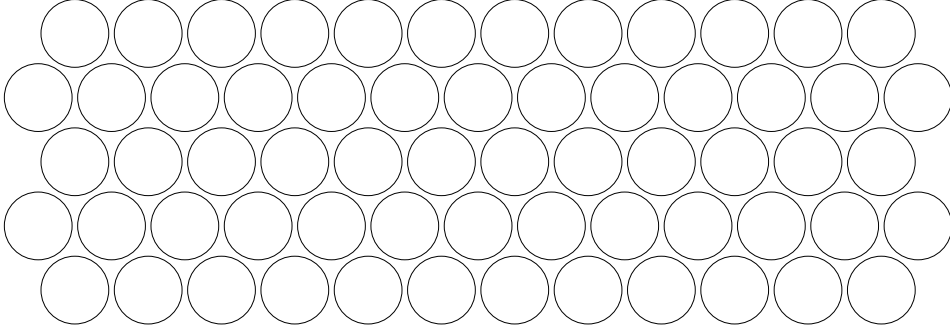


Figure 2.3: Spatial subdivision of a crystal lattice using an augmented plane-wave basis. Each circle depicts the muffin-tin sphere of an individual atom in the crystal structure. The area not contained in a circle is called the interstitial area.

The wave function $\Psi_{\mathbf{k},\nu}^\sigma(\mathbf{r})$ expansion is given by the expression in equation (2.23).

$$\Psi_{\mathbf{k},\nu}^\sigma(\mathbf{r}) = \sum_{|\mathbf{G}+\mathbf{k}| \leq K_{\max}} c_{\mathbf{k},\nu}^{\mathbf{G},\sigma} \varphi_{\mathbf{G},\mathbf{k}}^\sigma(\mathbf{r}) \quad (2.23)$$

Dependencies of the wave function include the Bloch vector $\mathbf{k}$, the spatial position where the wave function is evaluated $\mathbf{r}$, the spin component σ as well as the band index ν . equation (2.23) also includes the expansion coefficients $c_{\mathbf{k},\nu}^{\mathbf{G},\sigma}$ as well as the cutoff parameter $K_{\max}$ which determines the size of the basis set based on which the wave function is constructed. For a given Bloch vector $\mathbf{k}$ the wave function is constructed using the basis vectors, which correspond to all reciprocal lattice vectors $\mathbf{G}$ which fulfill the condition $|\mathbf{G}+\mathbf{k}| \leq K_{\max}$. The expression $\varphi_{\mathbf{G},\mathbf{k}}^\sigma(\mathbf{r})$ denotes the LAPW basis functions consisting of both the plane-wave part and the augmented contributions. The individual contributions to the LAPW basis functions are shown in equation (2.24).

Plane-Wave and
Augmented Basis
Parts

$$\varphi_{\mathbf{G},\mathbf{k}}^\sigma(\mathbf{r}) = \begin{cases} \frac{1}{\sqrt{\Omega_{\text{Cell}}}} e^{i(\mathbf{k}+\mathbf{G}) \cdot \mathbf{r}} & \text{interstitial region} \\ \sum_{lm} \left(a_{lm,\sigma}^{\mu,\mathbf{G},\mathbf{k}} u_{l,\sigma}^\mu(r_\mu) + b_{lm,\sigma}^{\mu,\mathbf{G},\mathbf{k}} \dot{u}_{l,\sigma}^\mu(r_\mu) \right) Y_{lm}(\mathbf{r}_\mu) & \text{muffin-tin } \mu \end{cases} \quad (2.24)$$

The coefficients $a_{lm,\sigma}^{\mu,\mathbf{G},\mathbf{k}}$ and $b_{lm,\sigma}^{\mu,\mathbf{G},\mathbf{k}}$ are used in the LAPW basis set to enforce the continuity of the basis functions and their corresponding derivative at the system's muffin-tin boundaries for both spin components. The volume of the unit cell Ω_{Cell} is included as a normalization factor for the interstitial plane-wave basis used within the FLAPW method. The spherical harmonics Y_{lm} in the basis set inside the muffin-tin sphere match the spherical contribution of the physical potential. Also, the radial contribution emerging from the radial solutions of the Schrödinger equation $u_l^\mu(r_\mu)$ is part of the basis within the muffin-tin spheres. The radial contribution and the spherical harmonics dependency on the position $\mathbf{r}_\mu$ within the muffin-tin sphere μ emerges from the spatial localization of the spheres. The appearance of both the normalized radial function $u_l^\mu(r_\mu)$ and the matching energy derivative $\dot{u}_l^\mu(r_\mu)$ is the reason for the linearity of the eigenvalue problem within the FLAPW method.

The sum $\sum_{lm}$ abbreviates the summation $\sum_{l=0}^{l_{\max}^{\mu}} \sum_{m=-l}^l$ where $l_{\max}^{\mu}$ represents a cutoff parameter which can be set for atoms of the same type prior to a calculation within the FLAPW method. [44] To examine films surrounded by vacuum regions, it is possible to extend the LAPW basis by additional basis functions for this distinct region. However, while during this thesis, different concepts of the FLAPW method and implementations of the FLEUR code are used, the detailed discussion of these concepts is not the goal of this thesis. Details on the theory of thin films and the vacuum region within the FLAPW method can be found in e.g. [39, 46]. Also, details on the theory of spin-orbit coupling (SOC) within the FLAPW method can be found in e.g. [47].

Eigenvalue Problem in the FLAPW Method

As the LAPW basis set does not form an orthogonal basis set, the eigenvalue problem introduced in equation (2.16) transforms to an eigenvalue problem which contains an overlap matrix. The wave function corresponding to the—now spin-dependent—eigenvalue $\epsilon_{\nu,k}^{\sigma}$ used in the FLAPW method is given by a sum of basis functions, as shown in equation (2.25).

$$|\psi_{\nu,k}^{\sigma}\rangle = \sum_{|\mathbf{G}+\mathbf{k}|\leq K_{\max}} c_{\mathbf{k},\nu}^{G,\sigma} |\varphi_{\mathbf{G},\mathbf{k}}^{\sigma}\rangle \quad (2.25)$$

From the basis representation in equation (2.25) the Schrödinger equation can be written as equation (2.26).

$$\hat{H} |\psi_{\nu,k}^{\sigma}\rangle = \sum_{|\mathbf{G}+\mathbf{k}|\leq K_{\max}} c_{\mathbf{k},\nu}^{G,\sigma} \hat{H} |\varphi_{\mathbf{G},\mathbf{k}}^{\sigma}\rangle = \epsilon_{\nu,k}^{\sigma} \sum_{|\mathbf{G}+\mathbf{k}|\leq K_{\max}} c_{\mathbf{k},\nu}^{G,\sigma} |\varphi_{\mathbf{G},\mathbf{k}}^{\sigma}\rangle \quad (2.26)$$

Multiplying equation (2.26) on the left side with one of the basis functions $\langle \varphi_{\mathbf{G}',\mathbf{k}}^{\sigma} |$ one ends up with equation (2.27).

$$\sum_{|\mathbf{G}+\mathbf{k}|\leq K_{\max}} c_{\mathbf{k},\nu}^{G,\sigma} \langle \varphi_{\mathbf{G}',\mathbf{k}}^{\sigma} | \hat{H} |\varphi_{\mathbf{G},\mathbf{k}}^{\sigma}\rangle = \epsilon_{\nu,k}^{\sigma} \sum_{|\mathbf{G}+\mathbf{k}|\leq K_{\max}} c_{\mathbf{k},\nu}^{G,\sigma} \langle \varphi_{\mathbf{G}',\mathbf{k}}^{\sigma} | \varphi_{\mathbf{G},\mathbf{k}}^{\sigma}\rangle \quad (2.27)$$

The expression $\langle \varphi_{\mathbf{G}',\mathbf{k}}^{\sigma} | \varphi_{\mathbf{G},\mathbf{k}}^{\sigma}\rangle$ on the right side of the equation represents the overlap integral and can be written as the overlap matrix S . This notation results in the eigenvalue problem, as in equation (2.28).

$$(H - \epsilon_{\nu,k}^{\sigma} S) c_{\mathbf{k},\nu}^{\sigma} = 0 \quad (2.28)$$

In this case $c_{\mathbf{k},\nu}^{\sigma}$ represents the coefficient vector which contains the expansion coefficients of the wave function $|\psi_{\mathbf{k},\nu}^{\sigma}\rangle$ for all $\mathbf{G}$ which are included using the corresponding cutoff condition. As the basis set has individual spatial contributions for the interstitial region and the muffin-tin spheres, this is also the case for the Hamiltonian. The separation of the Hamiltonian is possible due to the locality of the potential contribution. Details on this spatial separation can be found in [39, 44].

During this project, the FLEUR [48, 49] *ab initio* code developed in Jülich—based on the FLAPW method—has been used in combination with the AiiDA framework [9, 10] which is used to apply the previously described method on the scale of high-throughput studies and hence enables large systematic materials screening studies. To use FLEUR within the AiiDA framework, both the corresponding plugin [11] and the Materials Science Tools package [50] are required.

2.2.4 Magnetism within the FLAPW Method

As magnetism is an essential material property in the context of materials discovery for spintronics, a brief overview is given on how magnetic moments and states are evaluated in magnetic collinear FLAPW calculations. It should be mentioned that the computational effort required when moving from a non-magnetic to a magnetic DFT calculation—assuming nothing else in the computational setup changes—is doubled. [44]

Doubled
Computational
Effort

Magnetic Moment

Computing the magnetic moment of an individual atom in FLAPW is closely linked to the representation of atoms by muffin-tin spheres. Using the magnetization density given by equation (2.7), the magnetic moment of an individual atom m_μ is calculated by evaluating the integral over the space occupied by the corresponding muffin-tin sphere. Hence, the N_{At} atoms system total magnetic moment is given by equation (2.29).

Total Magnetic
Moment

$$M_{\text{Tot}} = \int_{\text{Interstitial}} m(\mathbf{r}) d^3r + \sum_{\mu=1}^{N_{\text{At}}} m_\mu = \int_{\text{Interstitial}} m(\mathbf{r}) d^3r + \sum_{\mu=1}^{N_{\text{At}}} \int_{\text{MT } \mu} m(\mathbf{r}) d^3r \quad (2.29)$$

However, the term “total magnetic moment” within this thesis refers to the summation of local atomic moments m_μ , as shown on the right of equation (2.29), if not stated otherwise explicitly. The expression M_{Tot} on the left of equation (2.29) is referred to as total cell moment with regard to the unit cell volume Ω_{Cell} and includes the interstitial contribution to the magnetic moment.

Magnetic Ordering

With the ability to calculate the magnetic moment of individual atoms using DFT, the possibility to classify different magnetic states by their ordering arises. Regarding collinear magnetism, four different categories can be clearly distinguished for atoms in crystal lattices:

1. Non-magnetic: Vanishing total magnetization, but all individual magnetic moments are vanishing.
2. Ferromagnetic (FM): All atomic magnetic moments in a lattice have the sign.
3. Ferrimagnetic: The total magnetization is not vanishing, and there are different signs for atomic magnetic moments present.
4. Anti-ferromagnetic (AFM): The total magnetization is vanishing, while individual atoms have non-zero magnetic moments.

The different classes of magnetic ordering are illustrated in Figure 2.4. There exist other classes of magnetic ordering for non-collinear magnetic calculations such as e.g. spin-spirals. Such configurations are examined in research dedicated particularly to these phenomenons with—most often—hand-tuned calculation setups. Moreover, examining spin-spirals and related phenomenons requires a non-collinear treatment of magnetism, which increases the computational cost associated with such calculations. However, the high-throughput study presented within this thesis is restricted to collinear magnetic configurations.

Spin-Spirals

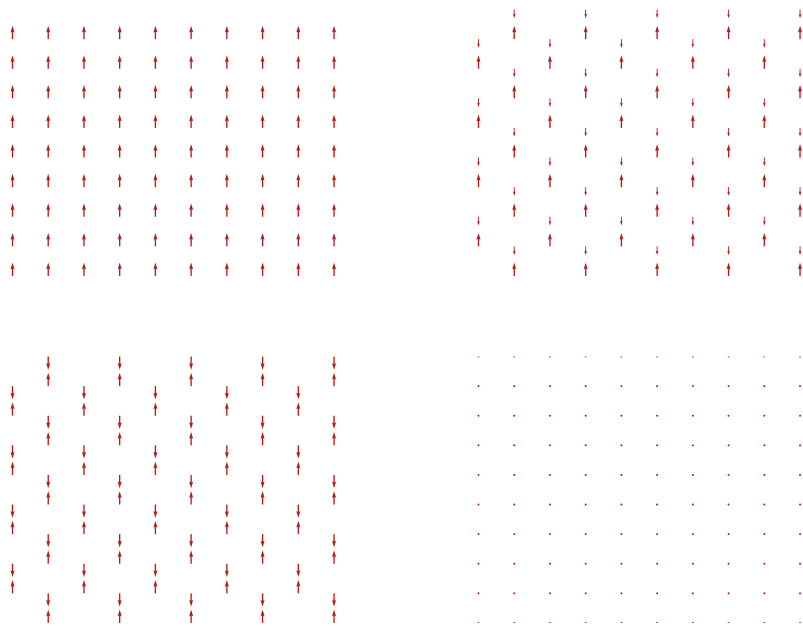


Figure 2.4: Illustration of different classes of magnetic ordering in the case of collinear magnetism. The order in the figure is. FM (Top left), Ferrimagnetic (Top right), AFM (Lower left), and Non-magnetic (Lower right). The length of an arrow indicates the magnitude of a magnetic moment, while the arrow's orientation denotes the sign.

2.2.5 Self-Consistent Field Loop

The effective KS single particle equations are solved iteratively—as the exact charge density of a system is typically unknown, and hence the charge density describing the system’s properties appropriately needs to be determined iteratively using an initially constructed starting density—until a given convergence criteria concerning the electronic ground state density $n_0(\mathbf{r})$ is reached. Typically, it is enforced that the distance between the input and output charge densities within an iteration should be below a given threshold, as shown in equation (2.30). [51] The $\mathcal{L}^p$ norm included in equation (2.30) is defined as in [52].

Iterative Solving

$$\frac{1}{\Omega_{\text{Cell}}} \left\| n_{\text{Out}}^{\uparrow/\downarrow}(\mathbf{r}) - n_{\text{In}}^{\uparrow/\downarrow}(\mathbf{r}) \right\|_{\mathcal{L}^2(\Omega_{\text{Cell}})} < \beta \quad (2.30)$$

Typical values for the charge distance cutoff parameter β range from $10^{-5} \frac{\text{m e}^-}{a_0^3}$ to $10^{-3} \frac{\text{m e}^-}{a_0^3}$ depending on the required accuracy and the scale of the target quantity which is desired, with a_0 denoting the Bohr radius. For instance, the magnetocrystalline anisotropy, which often is in the order of $10 \mu\text{Ha}$ per unit cell, will require a much stronger convergence criterion than the molecular binding energy, which can be around the scale of 1 eV per binding. In principle, this discrepancy of energy scales translates to the required convergence criterion. A schematic self-consistent field (SCF) loop is depicted in Figure 2.5.

Convergence
Criterion Scale

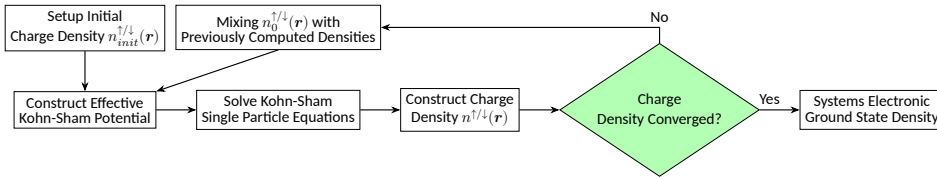


Figure 2.5: Depiction of a typical SCF loop in an *ab initio* calculation from the initial starting density to the converged electronic ground state density. Figure inspired by [36].

There are different mixing schemes (e.g. straight linear and Broyden [53] to mention a few) used with the FLAPW method which have all their own advantages and drawbacks which makes them subject to change depending on the nature and difficulties of the computed system. Within the FLEUR code, an implementation of a Kerker [51, 54] preconditioner is available. Deciding on the mixing scheme is not the only choice at the beginning of an SCF loop. Also, cutoff parameters like the mentioned K_{max} and l_{max} from section 2.2.3 are set at this stage. Of course, an *ab initio* calculation also requires an input structure to be provided. Based on the given structure, the reciprocal lattice of k points can be reduced based on existing lattice symmetries.

Typically, at the end of a successful *ab initio* calculation, not only the computed electronic ground state density is stored, but the acquired density is also used to compute system properties and provide them as additional calculation output.

Structural Relaxation Loop in DFT

Performing a structural relaxation using DFT embeds the previously discussed SCF loop in a relaxation procedure, which again represents an iterative process as with each successful SCF calculation a set of forces acting in the system on each atom is generated. These forces $F_{i,\alpha}$ —while i represents the atom occupying site i in the lattice and α denotes the axis of the force—are then used to:

1. Determine if the structure is relaxed using a force threshold and the condition given by the expression in equation (2.31).
2. Shift the atom position occupying site i according to the force vector F_i .

$$\max_{\{i,\alpha\}} \left(\|F_{i,\alpha}^{\text{New}} - F_{i,\alpha}^{\text{Prev}}\|_1 \right) < F_{\text{Thres}} \quad (2.31)$$

The definition from equation (2.31) uses the p -norm definition from equation (2.58). A scheme that shows how the SCF loop is embedded into the relaxation procedure is shown in Figure 2.6.

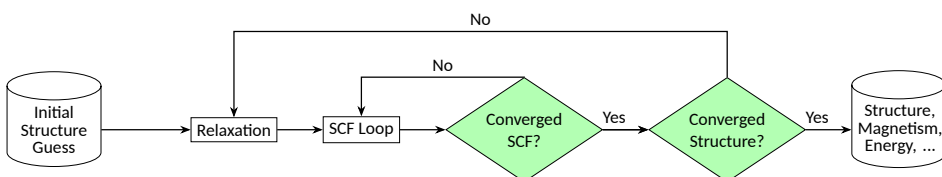


Figure 2.6: Schematic depiction of the relaxation procedure in FLAPW including the embedded SCF procedures starting from an initial structure guess and yielding an *ab initio* relaxed structure as result of the workflow, besides other DFT typical outputs like e.g. magnetic moments and properties, total energy and the possibility to determine electronic band structure and the DOS. Figure adapted from [55].

Commonly, for force relaxations—being a non-linear optimization problem—there are existing implementations of a BFGS scheme to improve the relaxations' convergence. [56–59] However, for convergence stability reasons, BFGS is not activated by default but rather used as soon as the force is reasonably low after an SCF loop. Typically, until this point is reached, a linear scheme is applied.

2.2.6 Korringa-Kohn-Rostoker Method

The Korringa-Kohn-Rostoker (KKR) method [60, 61] is an approach to DFT based on Green's functions (GF). Instead of diagonalizing a Hamiltonian, the Green's function is found directly from the Kohn-Sham Hamiltonian, employing multiple scattering theory. Within the KKR method, the full-potential expansion around the atomic centers is typically used, which takes the full geometry of the crystal structure into account, rather than using the concept of muffin-tin spheres as spacial separation. A versatile open-source implementation of the KKR method is the JuKKR code. [62] The JuKKR code is also linked to the AiiDA framework [9, 10] using the AiiDA-KKR [63] plugin and hence enables the method's application on a high-throughput scale.

Korringa-Kohn-Rostoker Method

High-Throughput

The Green's function formulation of the KKR method has the advantage of an extensive range of possible applications. [64] Non-standard features that are often more cumbersome to obtain in wave function-based methods include:

- Disordered systems with, e.g. impurities and defects
- Transport properties as e.g. spin and anomalous Hall conductivities
- Magnetic response functions as e.g. spin-susceptibility in linear response
- Pair interaction parameters
- Magnetocrystalline anisotropy

The pair interaction parameters J_{ij} , computed using the KKR method, are closely related to magnetic properties of the system as they map the electronic structure onto the classical Heisenberg Hamiltonian as shown in equation (2.33). The J_{ij} are also called exchange coupling parameters, as they couple two spins to each other in the Heisenberg model Hamiltonian. Exchange coupling parameters are computed using the local force theorem [65, 66]. In equation (2.32) the calculation of the exchange coupling tensor $\hat{J}_{ij}^{\alpha\alpha'}$ is shown. This expression results from scattering theory and was derived using linear response theory and not even assuming a collinear magnetic moment. [66–69]

Exchange
Coupling
Parameters J_{ij}

$$\hat{J}_{ij}^{\alpha\alpha'} = -\frac{1}{\pi} \text{Im} \left(\int_{-\infty}^{E_F} \text{Tr} \left(\Delta V_i^\alpha G_{i \rightarrow j} \Delta V_j^{\alpha'} G_{j \rightarrow i} \right) dE \right) \quad (2.32)$$

In equation (2.32) E_F denotes the Fermi energy, i.e. the energy at which the energetically highest occupied state of a material is existent—at the temperature $T = 0$ K. Furthermore, the GF $G_{i \rightarrow j}$ represents the propagation between the two sites i and j in the lattice, and ΔV_i^α denotes the potential change caused by an infinitesimal rotation of magnetic moment at site i to the axis α while α is given by the e_x , e_y , and e_z unit vectors. [68, 69] The exchange coupling tensor allows us to directly compute the pairwise exchange coupling parameter J_{ij} from the trace of the exchange coupling tensor. Besides the exchange coupling constant, it is possible to acquire the elements of the Dzyaloshinskii-Moriya vector based on this approach. [69]

Dzyaloshinskii-
Moriya
Interaction

2.3 Selected Magnetic Material Properties & Phenomenons

There are many magnetic phenomenons known in modern-day solid-state physics (e.g. spin-spirals, Skyrmions, and superconductivity, to name a few). In fact, in magnetism, phase changes of materials can be observed, where a material's magnetic configuration changes from one magnetic state to another. [70] While some observed magnetic effects are only observable in very special conditions, others shape materials such that their properties emerge through magnetic effects (as in the case of e.g. half-metallicity). [71–77] This section serves as an introduction to effects that are either observed or predicted during the course of this thesis or otherwise related to applications.

Phase Changes

2.3.1 Critical Temperature

A given material's stable magnetic state can be subject to change as external parameters of the material's environment change. This can include magnetic phase changes from e.g. FM to non-magnetic states. If the phase change can be associated with changing the material's temperature, then the temperature at which this phase change spontaneously occurs is called critical temperature. In the special case of FM and AFM materials, which exhibit vanishing net magnetization given a temperature rise, this critical temperature is also called Curie or Néel temperature, respectively. For an FM system, this essentially means that after the critical temperature has been exceeded, the net magnetic moment—which was present before—has vanished. Above the critical temperature, the magnetic ordering in a solid crystal structure is lost. The difference between the two states is exemplary shown in Figure 2.7.

Critical
Temperature

Magnetic
Ordering

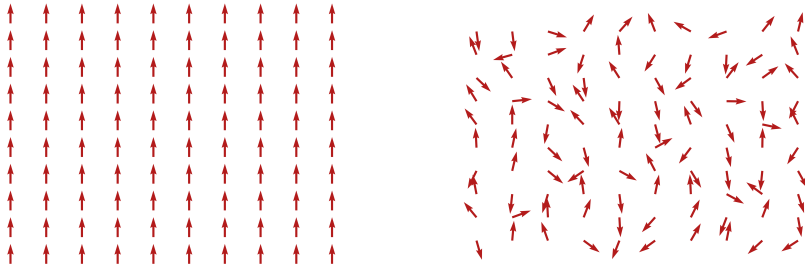


Figure 2.7: Exemplary visualization of an FM 2D Ising model [78] below (left) and above (right) the critical temperature. The pair exchange parameters have been chosen $J_{ij} > 0$ to enforce ferromagnetism as initial ordering.

As shown in the classical Heisenberg model Hamiltonian, which can be used to describe magnetic configurations, from equation (2.33), the pair exchange coupling constant is directly related to the Hamiltonian of the system. Also, the uniaxial anisotropy (K_{Crys}) of the crystal lattice can be found in equation (2.33) which is responsible for e.g. the magnetocrystalline anisotropy energy as K_{Crys} leads to an alignment of the spins within the lattice to be preferred to align the easy axis α . $e_{\alpha,i}$ denotes the unit vector in the direction of the easy axis α at lattice site i . [79]

Magnetocrys-
talline Anisotropy
Energy

$$\hat{H} = -\frac{1}{2} \sum_{ij} J_{ij} (\mathbf{e}_i \cdot \mathbf{e}_j) - K_{\text{Crys}} \sum_i (\mathbf{e}_{\alpha,i})^2 \quad (2.33)$$

The Hamiltonian from equation (2.33) is also called the classical Heisenberg Hamiltonian in the absence of an external magnetic field. [80] Mapping from the quantum mechanical DFT electronic structure to the classical Heisenberg Hamiltonian is a multiscale approach that allows the description of the magnetic order on a length scale, which would be infeasible with first-principles methods alone. This multi-step process and the required or acquired data in each step are depicted in Figure 2.8. It is important to mention that the *ab initio* simulation is approximately just as computationally expensive as the MC algorithm.

The accuracy of the determined T_c using this method is rather accurate, as sources report errors compared to experimental values of about 10 % to 15 %. [81, 82] The two simulation steps required to determine T_c are briefly described in the following.

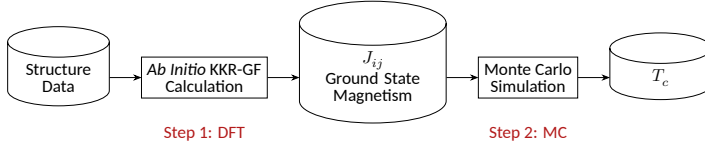


Figure 2.8: Schematic depiction of the steps in the *ab initio*-based determination of the critical temperature. Cylinders depiction of input and output data of the different steps.

Simulation Step 1: *Ab Initio* KKR Calculation for the Exchange Coupling Constants

Given a crystal structure—which also might include impurities and other structural disorders—a KKR calculation can be set up and performed until self-consistency is reached, as it was previously described in section 2.2.5. This gives access to major system properties such as the ground state magnetic moments and configuration, as well as the exchange coupling tensor $\hat{J}_{ij}^{\alpha\alpha'}$ for all lattice sites i and j and the spin directions α and α' . Only the diagonal elements are required from the exchange coupling tensor to compute the pair exchange coupling constants J_{ij} as equation (2.34) relates the tensor to the exchange coupling parameters using its trace along the spin axes. [68]

Pair Exchange
Coupling
Constants

$$J_{ij} = \frac{\text{Tr}(\hat{J}_{ij}^{\alpha\alpha'})}{3} \quad (2.34)$$

Simulation Step 2: Monte Carlo

Using the DFT-originated J_{ij} parameters and structure, it is possible to compute the classical energy difference ΔE of a given lattice between two different states using a model similar to the 3-dimensional Ising lattice. For the MC-based approach, an initial spin setup of the lattice is randomly chosen. From there, a randomly selected spin gets flipped based on a probabilistic method. The acceptance probability $P(i \rightarrow j)$ of a flipped state j compared to the original state i is given by the expression from equation (2.35). [83–85]

Random Spin
Flips

$$P(i \rightarrow j) = \begin{cases} 1 & \Delta E < 0 \\ e^{-\frac{\Delta E}{k_B T}} & \Delta E \geq 0 \end{cases} \quad (2.35)$$

While ΔE is the energy of the new state subtracted by the Energy of the previous spin state as given in equation (2.36). Hence, a new state is always accepted if it reduces the system's energy given by the model Hamiltonian. But if the overall energy increases, the probability is given by a function dependent on the temperature and ΔE . [84]

Temperature
Dependent

$$\Delta E = E_j - E_i \quad (2.36)$$

The amount of how many iterations of spin-flips are necessary vary depending on the inspected system. However, the literature suggests about 10^5 to 10^6 flips per spin to find an equilibrium and then use about 10^6 more steps for the averaging process. [86] For the computed system, it is necessary to average over the performed steps to acquire the expressions shown in equation (2.37). [84]

Number of Spin
Flips

$$a = \left\langle \left(\frac{1}{V} \sum_i^N \sigma_i \right)^2 \right\rangle_{\text{MC}} \quad b = \left\langle \left| \frac{1}{V} \sum_i^N \sigma_i \right|^2 \right\rangle_{\text{MC}} \quad (2.37)$$

Magnetic
Susceptibility

In equation (2.37) the variable V denotes the volume of the lattice and σ_i denotes the net spin of the atom occupying the i^{th} site in our model system. Using this notation, the magnetic susceptibility is given by equation (2.38). [84]

$$\chi(T) = V \frac{a - b}{k_B T} \quad (2.38)$$

Magnetic
Susceptibility
Peak

From equation (2.38) it is obvious that it is not enough to do the MC simulation once. It needs to be done for every temperature step to determine the magnetic susceptibility with reasonable accuracy for the given system. This temperature sampling gives sufficient resolution to search for the peak of the magnetic susceptibility, as this peak marks the critical temperature T_c . [86]

Sources of Errors

Sources of errors in the determination of T_c can include but are not limited to:

- DFT-based error as XC functional is an approximation. [82]
- Use of classical Heisenberg model. [82]
- The J_{ij} are assumed to be constant under temperature change. [82]
- Assumption of collinear spins in the Ising lattice.

Beyond these systematic error sources, additional numerical convergence issues of both the KKR-GF *ab initio* calculation and the MC simulation can introduce additional errors.

2.3.2 Half-Metallicity

Density of States

Given the presence of a magnetic material, it is possible that the $\uparrow$ -spin part of a DOS is exceeding the Fermi energy E_F while the $\downarrow$ -spin lies below the Fermi level. Hence, one of both spins is conducting while the other is not. In the following, this phenomenon and the resulting applications are discussed.

Fermi Level

Density of States

Number of
States per Unit
Energy

The DOS is a concept closely related to a material's conductivity properties and is deeply rooted in electronic structure theory. The DOS $\rho(E)$ represents a number of states at a certain energy range inside a single unit cell's volume Ω_{Cell} and is defined using the delta distribution $\delta(x)$ [87] in an integral over the BZ as in equation (2.39) while $\epsilon_n^\sigma(\mathbf{k})$ denotes the state's energy of the crystal for every state with spin σ . [36]

$$\rho(E) = \frac{\Omega_{\text{Cell}}}{(2\pi)^3} \sum_{n,\sigma} \int_{\text{BZ}} \delta(E - \epsilon_n^\sigma(\mathbf{k})) dk^3 \quad (2.39)$$

The DOS can be computed from *ab initio* calculations as each band's energy for every spin and k are determined within DFT. By construction, the number of states below the Fermi level matches the number of electrons N_{e^-} in the unit cell, as shown in equation (2.40). [88]

$$\int_{-\infty}^{E_F} \rho(E) dE = N_{e^-} \quad (2.40)$$

The relation from equation (2.40) is intuitively clear, as in the ground state all states below the Fermi energy E_F are occupied by electrons. Besides the total DOS, the method of projected density of states (PDOS) exists, which projects the DOS on e.g. atomic orbitals or the interstitial to learn which atom, spatial region, or orbital contributes to the DOS at the different energy levels. This way, one can not only learn about the system's physical properties, but using PDOS, it is possible to understand where they originate from. Projecting the DOS on the interstitial can give an insight into which degree of localization the electrons in the compound exhibit, as electrons in the interstitial tend to be delocalized. Thus, classifying materials based on their conductivity properties is possible using the DOS, which is discussed in the following.

Projected DOS

Localization

Conductivity Classification

Given the DOS of a compound, it is possible to classify the conductivity in the material. This can be done as the DOS contains information about the so-called valence and conduction bands. Valence bands of a material are states entirely located below the Fermi level, while conduction bands are located above. However, both band types can overlap; this concludes the fact that there exists no band gap (ΔE_{Band}) which is the case for e.g. metals. A band gap is defined as a range of energy values around the Fermi level where the DOS is zero; hence, no states exist within that gap.

Band Gap

Knowing the band gap between the valence and conduction band—in the non-magnetic case—conductivity can be classified into conductors, semiconductors, and insulators. Conductors do not have any band gap as both of the bands overlap. Semiconductors generally behave like insulators at very low temperatures, but thermal excitations can cause electrons to occupy conduction band states and hence give rise to conducting properties. Hence, semiconductors have a band gap around the Fermi level, but it must not be as large as the band gap of insulators, which typically do not exhibit any conductivity. Examples of all three conductivity types are shown in Figure 2.9. However, including magnetism and examining the DOS for both spin states individually, it is possible to find a band gap in one spin channel and states (i.e. a non-zero DOS) in the other spin state of the DOS. Such materials are called half-metals. Half metals can either be semiconducting or insulating in the non-conducting spin channel, depending on the band gap. Ideal half metals that are completely spin-polarized are considered important to spintronics as they are able to utilize the capabilities of spintronic devices entirely. [89]

Thermal
Excitations

Half-Metals

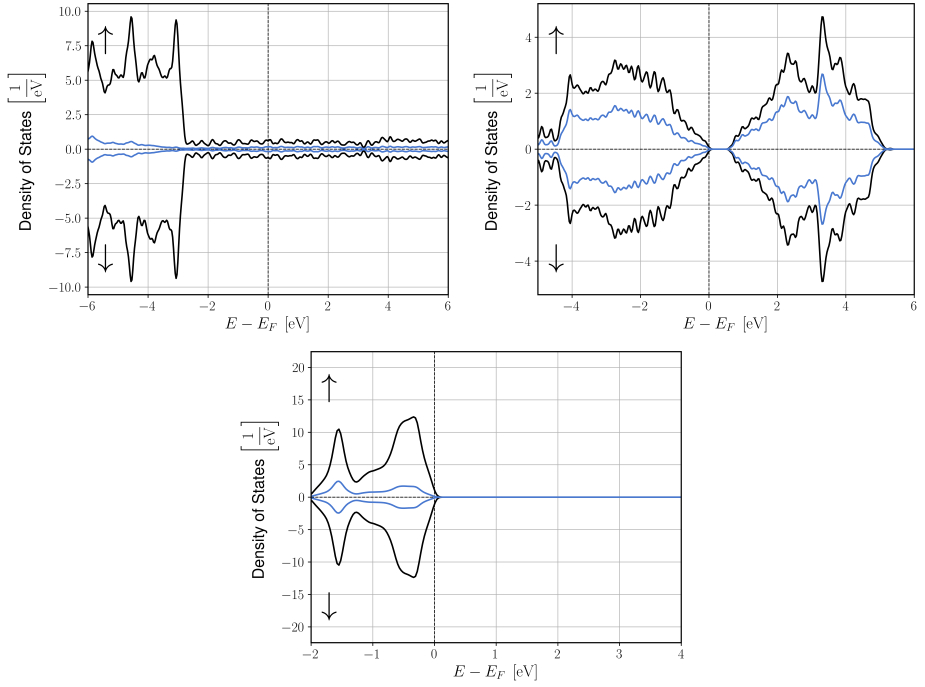


Figure 2.9: Exemplary DOS for one conductor (Ag top left), one semiconductor (Si top right) and one insulator (NaCl lower) obtained from FLAPW calculations. The black curves show the corresponding DOS, while the blue curve shows the PDOS, which originates from the states in the interstitial of the muffin-tin spheres. The dotted line represents the Fermi level. The upper half of each DOS plot shows the $\uparrow$ -spin contribution. The lower half shows the $\downarrow$ -spin contribution as indicated by the arrows on top of each plot and the negative sign on the number of states per eV on the x-axis on the right.

2.3.3 Requirements for Skyrmions

One technological concept within spintronics is the racetrack memory. This particular application, amongst others, builds on the magnetic phenomenon of magnetic domains. An example of such a magnetic domain is the particle-like Skyrmion. To understand the emergence of Skyrmions, it is necessary to include an antisymmetric exchange term into the simple model Hamiltonian from equation (2.33) and define the Dzyaloshinskii–Moriya vector $\mathbf{D}_{ij}$ as shown in equation (2.41) by using the definition of the exchange coupling tensor from equation (2.32). The introduction of the Dzyaloshinskii–Moriya vector is necessary as the Hamiltonian from equation (2.33) does not explain the stabilization of Skyrmions and the Dzyaloshinskii–Moriya interaction, which emerges from the spin-orbit interaction of the electrons, is needed to describe the phenomenon of stable Skyrmions. [68]

Antisymmetric
Exchange

Stabilization of
Skyrmions

$$\mathbf{D}_{ij} = \frac{1}{2} \begin{pmatrix} \hat{J}_{ij}^{yz} - \hat{J}_{ij}^{zy} \\ \hat{J}_{ij}^{xz} - \hat{J}_{ij}^{zx} \\ \hat{J}_{ij}^{xy} - \hat{J}_{ij}^{yx} \end{pmatrix} \quad (2.41)$$

Using the definition of the Dzyaloshinskii–Moriya vector from equation (2.41) the more general model Hamiltonian from equation (2.42) can be defined. [69]

$$\hat{H} = -\frac{1}{2} \sum_{ij} J_{ij} (\mathbf{e}_i \cdot \mathbf{e}_j) - K_{\text{Crys}} \sum_i (\mathbf{e}_{\alpha,i})^2 - \sum_{ij} \mathbf{D}_{ij} \cdot (\mathbf{e}_i \times \mathbf{e}_j) \quad (2.42)$$

Model Hamiltonian's like the expression from equation (2.42) are used to model phenomenons like spin-spirals and Skyrmions. It is also obvious that $\mathbf{D}_{ij}$ acts as a driving force that favors a non-collinear spin configuration rather than a collinear alignment, which is suggested by the exchange coupling constants and the uniaxial anisotropy. Both effects together and their interplay as competing interactions are the underlying reason for the emergence of magnetic phenomenons like Skyrmions and spin-spirals. However, in any case, a stable magnetic configuration in the carrier material to host such effects is required. Hence, the critical temperature also is directly relevant to the existence of spin transport properties.

Spin-Spirals

Competing
Effects

In the special case of 2-dimensional systems, one consequence of the Mermin-Wagner theorem [90] is that there are no long-range magnetic orders and subsequently no magnetic phase transitions present for the isotropic Heisenberg model at finite temperatures. The stabilization, in this case, is prevented by thermal fluctuations. [90] This concludes that Skyrmions cannot be stabilized in a system based on such a model Hamiltonian. Consequently the interplay of the Dzyaloshinskii–Moriya interaction and the magnetocrystalline anisotropy energy play a crucial role in stabilizing Skyrmions in thin film systems.

Mermin-Wagner
Theorem

2.4 Automated Interactive Infrastructure and Database for Computational Science

AiiDA The AiiDA framework [9, 10] is an open-source package that serves as a code base and infrastructure enforcing common standards for different materials science-related code plugins, including parsers, workflows, and data handling tools. The framework ensures data provenance and provides high-performance computing (HPC) plugins and interfaces that enable to run *e.g.* complex multi-layered *ab initio*-based workflows on a high-throughput scale. AiiDA stores operations and data node—including inputs and results—belonging to a workflow as a directed acyclic graph, which allows the reconstruction of the entire workflow even if the calculation was performed by someone else and were published in a database. Initializing and running an AiiDA workflow and a corresponding AiiDA code plugin is done using a high-level Python-based interface which can be executed *e.g.* inside a Jupyter Notebook. [9, 10, 91] The framework was developed following the pillars:

- Automation
- Data
- Environment
- Sharing

ADES Model These pillars are also referred to as the ADES model. [91] More details about the framework—including information specific to the AiiDA-FLEUR plugin [11]—are discussed in the following section.

2.4.1 Framework

The AiiDA framework allows developers to build up on the existing code base and classes while providing the infrastructure for workflows and data management. In the AiiDA context, workflows are Python programs that perform one or more computation steps, using their input to acquire the desired output for which the workflow is designed. This includes quantities that require multiple layers or steps of *ab initio* calculations to determine them. This is the case for *e.g.* the relaxation of a crystal structure.

The data provided as input to the workflows and retrieved as output along with metadata regarding the calculations (*e.g.* on which machine and using which code version the workflow was executed) is stored in nodes, which all have a unique identifier. The framework provides classes to handle data (As *e.g.* structure data, user-specified inputs, calculations parameters, and retrieved outputs) efficiently. This assists the setup of new—potentially more advanced—workflows, as the infrastructure is versatile and reusable. Hence, existing workflows can also be incorporated into more advanced workflows in a layered computation procedure. This concept finds full utilization as the most basic workflow implemented in AiiDA-FLEUR is the standard SCF workflow, which is performed as part of basically every other workflow at some point.

To interact with the AiiDA framework, the FLEUR code [48] requires the AiiDA-FLEUR [11, 92, 93] plugin and the corresponding parser [50] which parses the FLEUR input and output files into Python readable dictionaries. [92]

Also included in the AiiDA-FLEUR plugin are analysis and visualization tools for e.g. band structures, density of states, or the graph representing a workflow in AiiDA-FLEUR with all the individual steps performed during the execution of the workflow. [92]

2.4.2 Workflows

Workflows available in AiiDA-FLEUR [11] which are relevant to the results presented in the results section include the following functionalities:

- SCF run
- Equation of States (EOS)
- Inpgen execution
- Set up a magnetic film structure
- Relaxation

Other workflows computing spin-spirals, the Dzyaloshinskii-Moriya interaction and the magnetocrystalline anisotropy energy exist. Furthermore, the inputs required by each workflow vary, and even within a single workflow, multiple options exist for providing the necessary input to the workflow defined in the code. This is discussed in detail in the documentation of the AiiDA-FLEUR plugin. [11] it is possible to adapt existing workflows depending on the requirements. This can be necessary if:

1. There is additional output required which is not retrieved by the used workflow. (Could also require parser changes)
2. The input type options do not match the available input data despite the existing input data being sufficient to set up the calculations input.
3. It is desired to change an input parameter—which AiiDA-FLEUR currently determines in a workflow—manually.
4. A step in the workflow can be skipped (e.g. as it is already available to the user starting the workflow), but this optional functionality is missing.

In general, the adaptation of a workflow can be required as soon as additional functionality is needed. If required, entirely new workflows can also be set up to provide functionality beyond the outputs of existing workflows. Of course, creating an entirely new workflow can be a time-consuming endeavor. It might be an overkill to set up a corresponding workflow for only a few calculations. Of course, a potential high-throughput application mandates the setup of automation using a dedicated workflow. Hence, it should be considered if the workflow is used frequently enough to justify the time and effort of creating a new workflow. During this thesis, high-throughput FLAPW calculations have been performed. The workflow used mainly in this study is discussed in the following.

Adaptable
Workflows

Create-Magnetic-Film-Workflow

The AiiDA-FLEUR [11] plugin contains a dedicated workflow to set up symmetric and non-symmetric layered films. The possible input combinations for this workflow and the outputs are shown in Figure 2.10. [11, 92]

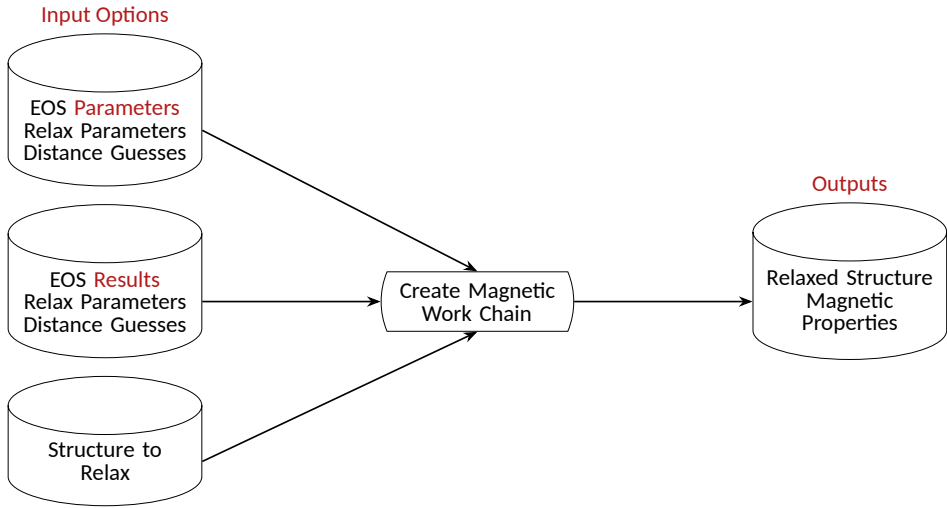


Figure 2.10: Flowchart of different inputs which can be used to acquire the relaxation results, which consist of the relaxed film structure and the magnetic configuration. Cylinders represent the input and output data of the workflow.

The film relaxation workflow can be used in combination with the EOS workflow—and hence requires the EOS workflow’s input parameters in this case—and also takes data-based distance guesses.

Data-based
Initial Lattice
Constant Guess

Within AiIDA-FLEUR [11], there exists a routine that computes initial lattice constant guesses, based on data stored in the database of the Materials Project [94], and stores them in a Python dictionary for future use. Passing the routine for the initial guesses, a list of elements and subsequently all possible element pair combinations are determined and—using the Materials Project Application Programming Interface (API)—a request for all bond lengths stored in the database for each elemental pair combination is sent, one after the other. At the end of a single API request, the bond lengths obtained are averaged, and the average is then stored for this element combination before the next request is sent. Hence, in the dictionary output of the routine, each combination of elements X and Y have a mean bond length stored with them a_{Guess}^{XY} which can be used as a data-based initial guess for the starting point of the EOS and relaxation workflows. [11, 55, 92]

The Create-Magnetic-Film workflow in AiIDA-FLEUR [11] can work with either the outputs of an EOS workflow or perform the EOS computation at the beginning of the film relaxation workflow. An EOS workflow computes the bulk lattice constant of the film substrate layers. An initial guess of the lattice constant $a_{\text{Guess}}^{\text{SubSub}}$ taken from the Materials Project [94] distance guesses dictionary is then subsequently scaled as shown in equation (2.43). [55]

$$a_n = (0.9025 + 0.005n) a_{\text{Guess}}^{\text{SubSub}} \quad (2.43)$$

EOS Sampling

While restricting n to fulfill $0 \leq n \leq 39$. This results in a discrete sampling of scaled lattice constants ranging from 90 % to about 110 % in relation to the initial lattice constant guess. However, performing the EOS evaluation for every workflow started is unnecessary. The requirement to perform the EOS evaluation vanishes once each substrate’s EOS has been evaluated.

As—using the discretization discussed for equation (2.43)—with each EOS evaluation, 40 individual SCF calculations are performed, and the corresponding total energy E_n of the bulk substrate system is evaluated every time the workflow is executed. From this point on, it is possible to recycle previously performed EOS computations (See Figure 2.10 second input option from above) when submitting a film relaxation workflow. The film substrate will be set up using the substrate lattice constant a_n , which minimizes E_n . However, suppose the supposed minimum determined by the workflow is located at the smallest or largest scaled lattice constant. In that case, it is possible that this scaling indeed does not represent a minimum of the total energy but rather the lowest energy within the given interval. If so, the sampling scale should be reevaluated and adjusted to determine a suitable lattice constant for the subsequent calculations. The input parameters for the EOS workflow specify e.g. the sampling and scaling of the initial lattice constant guess. [55, 92]

All steps of the AiiDA-FLEUR [11] film relaxation workflow are shown in Figure 2.11.

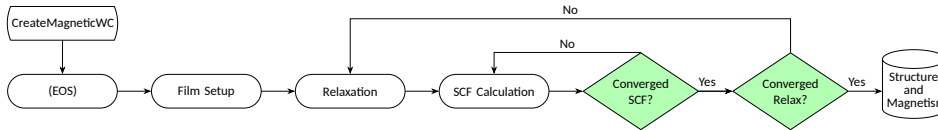


Figure 2.11: Flowchart of the individual steps which are performed after starting the film relaxation work chain. As the EOS step's necessity depends on the fact that the previous EOS output of the chosen substrate is available, the step is put in brackets as it might not be required. The relaxation workflow from Figure 2.6 is fully incorporated into the AiiDA-FLEUR film relaxation workflow. The cylinder shape represents the output resulting from the AiiDA-FLEUR workflow. Depiction adapted from [55].

Unfortunately, not every film relaxation succeeds. Hence, in practice, after cutoff parameters for e.g. number of SCF iterations or relaxation steps are reached, the workflow will be stopped, and the relaxation will be marked as unsuccessful. However, AiiDA-FLEUR [11] autonomously tries to fix some common problems that arise during relaxations by e.g. restarting individual failed SCF calculations or adjusting the film structure or the muffin-tin setup. [11, 92] The film setup follows a structured procedure. First, after the EOS results are available, the layered film with a total number of n_{Layer} layers is completely set up using the substrate lattice constant from the EOS results and the substrate element. This first step is shown on the left half of Figure 2.12. [11, 92] In the second step, the user-specified layers are replaced with the magnetic layers, either on a single or on both sides of the substrate. The interlayer distances (ILDs) between two neighboring layers are computed based on the initial lattice constant guesses acquired from the Materials Project [94] bond length data. It is established practice to scale the ILDs that lie in between the outermost magnetic layers and their neighbors—with a number slightly smaller than one—as experience shows that they tend to compress towards the film center. For the high-throughput calculations performed during this thesis, a factor of 0.95 has been used, however, also lower values are possible. [92] No further adjustments to the film are performed before the relaxation procedure. [11, 92] Which layers undergo a relaxation (and in which direction) is subject to user input. A common choice is to relax the magnetic layers only (and maybe the substrate interface layer) while exclusively relaxing the z -direction to maintain the substrate-governed lattice. [92]

Interlayer
Distances

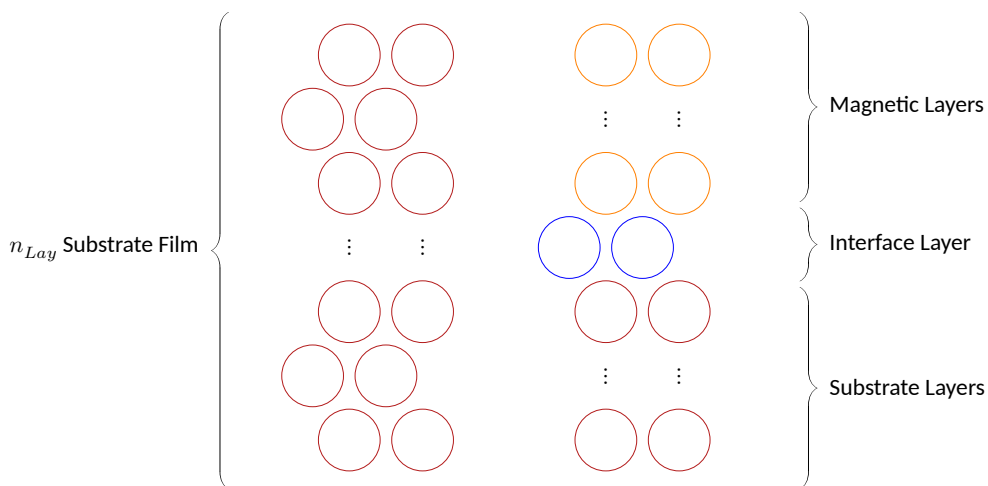


Figure 2.12: Depiction of the individual steps in the structural setup procedure. The initial film setup on the left uses only the substrate lattice constant and the substrate element. The second step is depicted on the right, where a user-specified number of layers is replaced with the elements specified for the magnetic layer. The interface layer can either belong to the substrate or the magnetic layers defined by the user-provided structural setup specifications. For visualization simplicity, the magnetic layers are colored in the same color, and a non-symmetric film has been depicted. However, different elements can occupy each magnetic layer, and symmetric films can be constructed this way. When writing this thesis, the workflow can set up films based on fcc and body-centered cubic substrate lattices.

2.4.3 Data Management

Scientific Data
Silos

Research data management is becoming an increasingly urgent and crucial topic. Often, research data is stored on local machines (data silos) only accessible to a few researchers or even a single individual—while others might not even be aware of the data's existence. Publicly funded research has to ensure that research results and corresponding data are examined, stored, and accessible to as many people as possible so the accumulated data can be reused as often as possible in different contexts to generate as much scientific value as possible from this data. [95, 96] To achieve this, the FAIR principles [95] have been developed and established in the scientific community, which focuses on designing software and data architectures in a way that sustainable data management is ensured. FAIR [95] denotes the four underlying principles, which are displayed in Figure 2.13. Obviously, there are similarities between the ADES model mentioned in the introductory part of section 2.4 and the FAIR principles. Hence, it is no surprise the AiiDA framework enables compliance of data management with both ADES and FAIR. [9, 10, 93] For example, the AiiDA framework labels each data node with a unique identifier (UUID) as well as each process with a process identifier (PID) and references related nodes using these identifiers. Furthermore, the locally stored AiiDA database is Structured Query Language (SQL) based, and data nodes can e.g. be user-described with labels and descriptions. AiiDA also allows exporting and importing entire databases or segments as compressed files. However, AiiDA also saves input and output calculations files in a structured file repository. [93, 97]

Sustainable Data
Management

FAIR and ADES

This ensures that data can be shared, reused and does not remain in a data silo with the potential risk that the data's existence will be forgotten and hence, the data can neither contribute towards further scientific advances nor can be used as training data for future materials screening or materials design studies. [95]

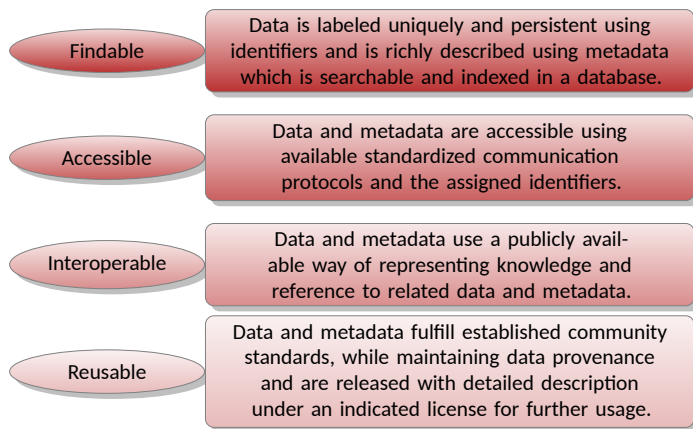


Figure 2.13: Depiction of the FAIR principles. The content of this figure is based on [95].

2.4.4 General Workflow—Materials Science Data for Predictive Analysis Using Machine Learning

Tackling a materials discovery or materials screening task is a multistep process, ranging from a possible technical application to specific predictions of either material properties or candidate materials related to the application case. A typical workflow for predictive materials discovery and screening challenges is shown in Figure 2.14.

Materials
Screening

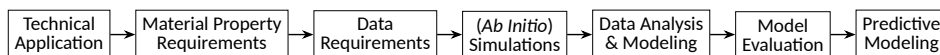


Figure 2.14: Possible workflow from a technical requirement for a material exhibiting distinguished properties to predictive modeling.

This workflow also incorporates the previously discussed design goals and data requirements steps, as well as the *ab initio* simulations. Starting from a technical application, an arising need for a material having specific properties sets the goal of the materials design tasks even before any data has been collected using computational simulations. In the particular case of this thesis, the computational step is carried out using first-principles (*ab initio*) calculations. Generally, *ab initio* methods are the method of choice for specific technical requirements like e.g. magnetic and conductivity properties. However, while in general and interdisciplinary contexts a very similar methodology is established and quite commonly found (e.g. in drug design [98]) using other simulation methods it is an emerging methodology in the context of the growing field of high-throughput DFT studies. [92, 93, 99] In the context of these studies, ML models and techniques play a key role due to their capability to use the generated data to get additional insights and applications that can range beyond the knowledge that can be acquired by analyzing the data.

Interdisciplinary
Contexts

Also, beyond high-throughput applications, ML models are used in the materials science community to address numerous challenges which include—but are not limited to—superconductivity [100, 101], entropy changes [102], materials synthesis [103], band gap predictions [101] and topological states [101]. Besides the use of ML, the methodology of data mining has led to knowledge discovery based on material science data. The discovery of material properties from data can also lead to new application fields beyond the initial research objective. A famous example of such a discovery represents the initial examination of polytetrafluoroethylene as a refrigerant, which then became renowned for the material's inherent anti-adhesive properties. [104, 105] Compared to publications that include the use of ML models in the field of materials science and materials design, approaches and publications that interpret and explain the models and use XAI techniques are not commonly found in the domain yet. [106] Summing up, it is worth mentioning that, despite the rising popularity of ML methodologies within the scientific community, ML methods are typically not part of the physics curriculum, which is why—in the following—an overview is given over techniques and methodologies which are relevant or related to the application cases discussed in section 3.

2.5 Machine Learning and Data Science

ML-based methods are a rapidly growing field that influences many research areas and disciplines. Hence, there exists a plurality of definitions for terms that have been used to this point in this thesis. In the following, common definitions are included to establish a common understanding of what is meant by which term.

Definition 1. Data science refers to the methodology and interdisciplinary academic field that is dedicated to systematically extracting and hence gaining knowledge from data. Data science is connected to the field of data analytics, statistics, and the field of ML. [107]

Definition 2. Machine Learning (ML) is a research area that employs algorithms to derive models from data in order to reproduce the laws underlying the relation between data and the information contained in the data. [108, 109] Applications of ML models are not exclusive to a single scientific area but rather diverse. [98, 99, 109] ML is considered a subbranch of the more general area of artificial intelligence (AI). [109]

Defining AI itself is a much more peculiar task, as the definition raises philosophical questions on the nature of intelligence. [110] The question of what intelligence actually is when dealing with computing machines solving tasks human-like goes back to the Turing test. [111] Generally accepted is that AI applications aim to achieve the same—or even outperform—problem-solving and decision-making capabilities of humans. This typically includes that these capabilities improve with experience. [112, 113]

2.5.1 Machine Learning

Within ML, a plurality of models and methods exist to tackle different tasks, such as regression, classification, clustering, and dimensionality reduction. Short descriptions of each of the mentioned model tasks are included in Figure 2.15. The applications outlined in this thesis focus on regression and classification tasks, which align with the goal of predictive materials modeling and screening materials. [114]

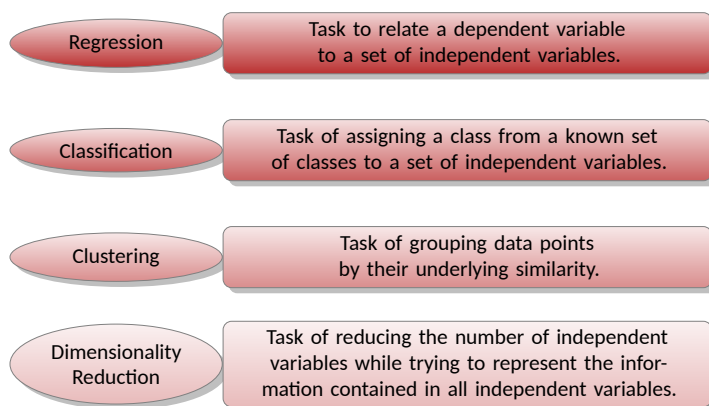


Figure 2.15: Depiction of different ML tasks with short descriptions of each task's objective. The dependent variable is also called the target quantity or predicted quantity. Independent variables are often called features or descriptors as they act as individual features of a data point describing the data point and hence determining the dependent variable for the model. Typically, for regression tasks, the dependent variable is continuous. [115–117]

Naturally, as the focus lies on regression and classification, the ML methods used during the course of this thesis represent applications of supervised learning. However, clustering and dimensionality reduction are common tasks for unsupervised learning. For overview, the input and output constellations for supervised learning and for semi-supervised and unsupervised learning approaches are shown in Fig 2.16. [117, 118]

Focus on
Regression and
Classification

Reinforcement learning is based on a training process that involves a repeated trial-and-error procedure performed based on actions (*e.g.* decisions) performed by the model. When a model's trial is successful, the model receives a reward, based on some type of reward function, as denoted in Figure 2.16 by the dashed arrow. Using a reward, the model will—in the long run, after multiple cycles of trial-and-error—make decisions that maximize the reward function. [119] In the case of supervised learning, a metric is used in order to calculate an error that numerically expresses the difference between the original—ground truth—labeling and the model's label prediction. The goal of supervised learning is to minimize a chosen loss function. A loss function $\mathcal{L}$ incorporates a metric $\mathcal{M}$ which measures the error of the model's prediction, in comparison to the known ground-truth labels, and potentially a so-called regularization term, depending on the used model. By convention, loss functions are defined in a way that a larger value of $\mathcal{L}$ corresponds to a worse model performance. Hence, the ML training process—which is an optimization problem of the model's parameters—always is given as a minimization task. Hence, the model parameters are optimized so that they minimize the loss function on the training set. Different metrics that can be used in loss functions and their properties are discussed in the following section about metrics. In general, loss functions depend on the model's prediction as well as the true labels, but can also depend on model parameters and parameters that are related to a potential regularization term. The reward function, which is used in reinforcement learning, is conceptually the opposite to the loss function, as higher values of the reward function correspond to a better model performance. One class of models within ML are artificial neural networks (ANNs). ANNs are known to perform well on unstructured non-tabular data like *e.g.* images, audio sequences, e-mails, and videos. [120–123]

Supervised
Learning
Loss Function

ANNs

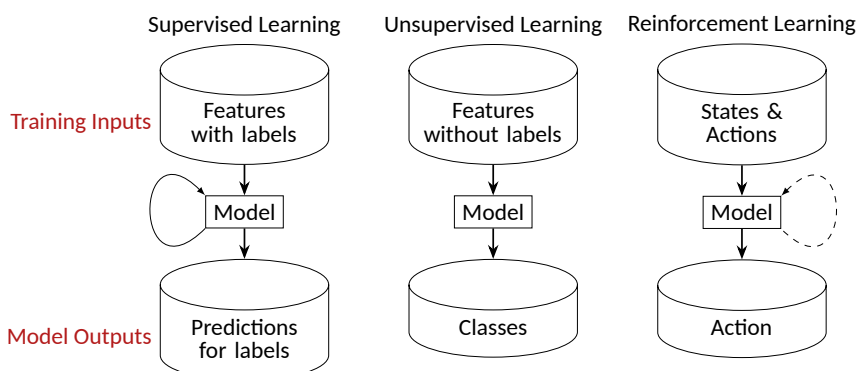


Figure 2.16: Schematic depiction of supervised, unsupervised, and reinforcement learning with the corresponding training inputs and model outputs. The term labels refers to entries for the dependent variable. The bent arrow denotes the error that results from the model's prediction in combination with the fact that the ground truth values for the labels are known from the training data set. The dashed bent arrow denotes the reward a reinforcement learning model receives, based on a particular action.

Applicability of ANNs

However, when it comes to structured tabular data, ANNs are typically outperformed in terms of performance and required computational time during model training by tree-based ensemble models. [124–128] Within the family of ANN models, there exists a number of categories like e.g. graph neural-networks, generative adversarial networks (GANs), convolutional neural networks (CNNs), deep learning models, autoencoders, and foundation models like for example large language models (LLMs). [129–131] Structuring the discussed relations within the area of AI, a possible visualization as shown in Figure 2.17 can be compiled.

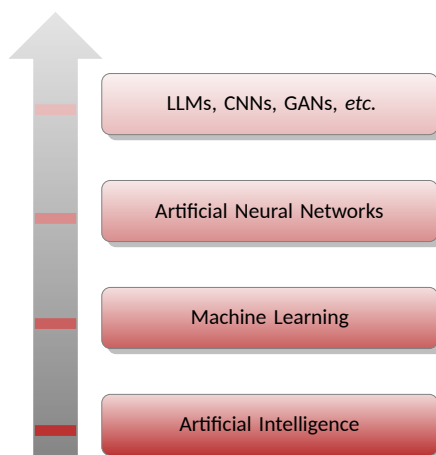


Figure 2.17: Depiction of the field of AI related to ML and neural network models. Each upper category is a subclass of the field depicted in the category below. The field of AI forms the foundation for all other ML techniques but also includes other topics and fields like e.g. robotics which are beyond the scope of this thesis.

Mathematical Interpretation of a Model

Consider a task where a single feature vector x_i is k -dimensional. Hence, the task has a number of k features. For a data set of N data points, there exist corresponding labels $y_i \forall i \in [1, N]$. Defining a function that performs a mapping as shown in equation (2.44) can be defined. This function is called predictor or estimator and corresponds to the model that maps the feature space onto the label space. [132]

$$f : \mathbb{R}^k \rightarrow \mathbb{R} \quad (2.44)$$

Predictor
Function

In fact, the label space is likely much smaller than $\mathbb{R}$ in application cases, especially for classification tasks.

Model Choice and Hyperparameter Optimization

Besides the dependent variables, many models depend on a set of hyperparameters Θ . Hence, the predictor $f_{\Theta}(x)$ will not only depend on the provided feature vector but also on a model-specific hyperparameter vector. The hyperparameters are not a result of the learning process but rather are required as input to the model before the beginning of the training process. [133] An example of a typical hyperparameter would be the number of estimators combined within an ensemble model, which are introduced in section 2.5.2.

Hyperparameters
as Model Input

The parameters that are trained during the learning process are model-specific. For example, in a simple linear regression predictor $f(x)$ —for a single feature x —defined by equation (2.45), the learned parameters are a and b .

Linear Regression

$$f(x) = ax + b \quad (2.45)$$

Due to the underlying simplicity, the classical linear regression does not have any hyperparameters. Hence, the hyperparameter optimization step shown in Figure 2.18 can be skipped for this particular model. From Figure 2.18 it also becomes apparent that the initial choice of features already influences the training process as well as the hyperparameter selection in general. Additionally, when many features are available for a given task, a dimensionality-reducing feature selection is often performed. It is important not to perform feature selection and learning on the same data set, as this might lead to an additional feature selection bias. [134]

Feature Selection
Bias

To determine hyperparameters, different strategies are common. The simplest method—often referred to as the holdout method [135]—splits the available data into a training set and a test set prior to model training. The test set is used after training the model using the training data and different combinations of hyperparameters to compare the performance of different models. However, this precludes any evaluation of the choice of hyperparameters with out-of-sample (OOS) data, as the test data was involved in the process of determining the hyperparameters. Hence, it remains unclear if the chosen hyperparameters are performing well on the test data set or generalize well to other data sets. It is a better practice to use separate sets of data for the training process, the model, the hyperparameter choice process, and the final evaluation to estimate the generalization error. The process of transitioning from the holdout method to further splitting of the data is depicted schematically in Figure 2.19. However, using a single dedicated validation set is only the simplest—but also computationally fast—way to introduce a validation step. A more sophisticated approach is to use the technique of cross-validation (CV).

Holdout Method

Generalization
Error
Validation Step

Before splitting the data, it can be useful to randomize the entire data set to avoid certain clusters being overly dominant in either set. Extreme clustering in the data set can be detrimental to the model's performance and generalization abilities, as this can—in extreme cases—lead to cases where the model has not learned a specific relation that might not be found in the test data but in the training data due to clustering. Such a case can easily be imagined, looking at the fact that the structural formula of the examined compound sorts some databases in materials science. This way, the model would likely never encounter e.g. the Element Zn at a certain site in the compound if one chose the last 20 % of the data set to be the test data set. Hence, some kind of randomization has to be performed before splitting the data, which can either mean drawing the validation and test set randomly from the entire data beforehand or randomizing the entire data set before the data-splitting procedure.

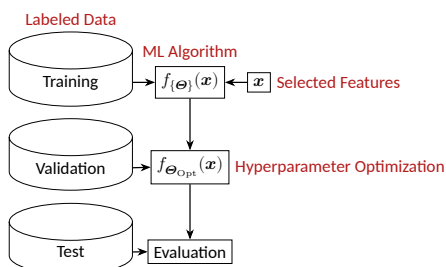


Figure 2.18: Depiction of the different stages and data usage in ML model optimization procedures. Θ_{Opt} is chosen using a validation set. The holdout method would merge the hyperparameter optimization and evaluation steps. Figure inspired by [133].

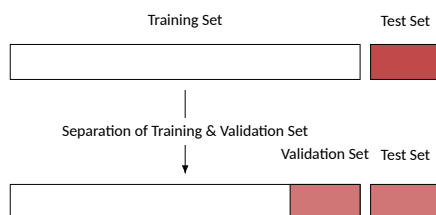


Figure 2.19: Depiction of data splitting using the holdout method (top row) as well as an additional validation set (further splitting in the lower row), where the available data is split into training, validation and test set.

In Figure 2.18 the notation $f_{\Theta}(x)$ denotes the ML model, which is trained using the features and labels from the training data set for a set of hyperparameter vectors $\{\Theta\}$ while $f_{\Theta_{Opt}}(x)$ denotes the model with the best-found hyperparameter vector Θ_{Opt} from the set of hyperparameter vectors $\{\Theta\}$. There are different methods to search for appropriate hyperparameters in the space of all possible combinations, which include grid search, random search [136], and Bayesian optimization (BO) [137] based methods. However, while grid search samples the whole provided hyperparameter space of combinations for choices of individual hyperparameters and hence will always find the best-performing set of hyperparameters out of all combinations, simply sampling all possible combinations is not an efficient approach. Random search algorithms, which sample all possible combinations of hyperparameters a given number of times randomly, are known to be more compute-time efficient than grid search approaches. [136] BO-based methods can improve the computational efficiency in hyperparameter search even further than random search algorithms. Using Gaussian processes, BO methods for hyperparameter optimization probabilistically estimate the error when sampling the hyperparameter space. The next point chosen to be sampled by the optimization algorithm will be the point in the hyperparameter space, with the largest probabilistic uncertainty. Hence, the phase space of possible hyperparameters—with each additional sample—is sampled systematically in regions where the algorithm has the least information about the performance of the sample point's associated hyperparameter combination in relation to other points. This procedure is known to be scalable to large dimensions of hyperparameter vectors. [138, 139]

Hence, depending on the dimensionality of the hyperparameter optimization problem as well as the data size, random search as well as BO can be used to increase the computational efficiency in finding adequate model parameters.

Computational
Efficiency

Bias-Variance Decomposition

Consider the data labels $\mathbf{Y}_{\text{True}}$ on which the model $f_{\boldsymbol{\theta}_{\text{Opt}}}(\mathbf{x})$ was trained. Let us assume that they may contain some random noise, but that they follow the true mapping $f_{\text{True}}(\mathbf{x})$. Any individual label y_i in $\mathbf{Y}_{\text{True}}$ is linked to $f_{\text{True}}(\mathbf{x})$ by equation (2.46). Then, the error that arises by approximating $f_{\text{True}}(\mathbf{x})$ with the model $f_{\boldsymbol{\theta}_{\text{Opt}}}(\mathbf{x})$ using an arbitrary loss function $\mathcal{L}(\mathbf{Y}_A, \mathbf{Y}_B)$, can be expressed as in equation (2.47). The loss function must be symmetric, such that the relation $\mathcal{L}(\mathbf{Y}_A, \mathbf{Y}_B) = \mathcal{L}(\mathbf{Y}_B, \mathbf{Y}_A)$ is fulfilled. [140, 141] In the loss function, $\mathbf{Y}_A$ and $\mathbf{Y}_B$ each represent a label vector of all labels from the according data set acquired by two different mappings A and B . In the following, it is assumed that one of both mappings represents the true mapping (i.e. the ground truth) of the data, and the other mapping is a model attempting to model the true mapping based on the data.

Noisy Data

$$Y_{\text{True}}^i = f_{\text{True}}(\mathbf{x}_i) + \text{Noise} \quad (2.46)$$

$$\begin{aligned} \mathbb{E} \left[\mathcal{L} \left(f_{\text{True}}(\mathbf{x}), f_{\boldsymbol{\theta}_{\text{Opt}}}(\mathbf{x}) \right) \right] = & \quad (2.47) \\ \mathbb{E} \left[\mathcal{L} \left(f_{\text{True}}(\mathbf{x}), \arg \min_{\mu} \left(\mathbb{E} \left[f_{\boldsymbol{\theta}_{\text{Opt}}}(\mathbf{x}) - \mu \right] \right) \right) \right] \\ + \mathbb{E} \left[\mathcal{L} \left(f_{\text{True}}(\mathbf{x}), \arg \min_{\mu} \left(\mathbb{E} \left[f_{\boldsymbol{\theta}_{\text{Opt}}}(\mathbf{x}) - \mu \right] \right) \right) - \mathcal{L} \left(f_{\text{True}}(\mathbf{x}), \arg \min_{\mu} \left(\mathbb{E} \left[f_{\text{True}}(\mathbf{x}) - \mu \right] \right) \right) \right] \\ + \mathbb{E} \left[\mathcal{L} \left(f_{\text{True}}(\mathbf{x}), f_{\boldsymbol{\theta}_{\text{Opt}}}(\mathbf{x}) \right) - \mathcal{L} \left(f_{\text{True}}(\mathbf{x}), \arg \min_{\mu} \left(\mathbb{E} \left[f_{\boldsymbol{\theta}_{\text{Opt}}}(\mathbf{x}) - \mu \right] \right) \right) \right] \end{aligned}$$

$\mathbb{E}[f(x)]$ denotes the expected value of an arbitrary function $f(x)$. The very first term on the right side of equation (2.47) represents the noise coming with the actual mapping $f_{\text{True}}(\mathbf{x})$. This error of the true data labels themselves is considered the irreducible error, as it is inherent to the examined data, and no model prediction can systematically exceed the accuracy of the data on which it has been trained on. The middle term on the right side of equation (2.47) represents the squared bias between the true mapping $f_{\text{True}}(\mathbf{x})$ and the model function $f_{\boldsymbol{\theta}_{\text{Opt}}}(\mathbf{x})$. The bias between these two mappings can be intuitively understood as the expectation value's deviation between both mappings. Of course, since the goal is that the model should be close to the actual underlying mapping, it is desirable for the bias to be small. The last term can be interpreted as the variance arising from the model's prediction error. This error is not only model-dependent but also is caused by the fact that only a finite amount of training data is available. [140, 141] The interplay between bias and variance terms visible in this particular decomposition is often called bias-variance trade-off, as typically, to decrease the bias, it requires a more complex model. At the same time, a more complex model also requires more training data to decrease the model's variance to the level before the complexity was increased. [142] A low-bias and low-variance model would be considered ideal. Low-bias and high-variance situations are commonly referred to as overfitting, while high-bias and low-variance constellations are typically described as underfitting. Empirically, the bias and variance of a model can be determined using in-sample $\mathbf{Y}_{\text{In}}$ and OOS $\mathbf{Y}_{\text{Out}}$ predictions. The OOS predictions can be acquired using e.g. the test set that has not been used in the model training and hyperparameter selection process.

Irreducible Error

Finite Amount of
Training Data

Bias-variance
Trade-off
Overfitting &
Underfitting

Given a loss function, $\mathcal{L}(\mathbf{Y}_A, \mathbf{Y}_B)$ bias and variance are approximately given by the expressions in the equations (2.48) and (2.49). [141]

$$\text{Bias} \approx \frac{\mathcal{L}(\mathbf{Y}_{\text{In}}, \mathbf{Y}_{\text{True}}) + \mathcal{L}(\mathbf{Y}_{\text{Out}}, \mathbf{Y}_{\text{True}})}{2} \quad (2.48)$$

$$\text{Variance} \approx \frac{\mathcal{L}(\mathbf{Y}_{\text{Out}}, \mathbf{Y}_{\text{True}}) - \mathcal{L}(\mathbf{Y}_{\text{In}}, \mathbf{Y}_{\text{True}})}{2} \quad (2.49)$$

The loss function $\mathcal{L}(\mathbf{Y}_{\text{Out}}, \mathbf{Y}_{\text{True}})$ does not only appear in the bias and variance terms in the equations (2.48) and (2.49), but also represents an estimate for the generalization error, as it evaluates the model performance on unseen data. Hence, large values for the variance correlate with a large generalization error, which indicates that the model is too complex, given the amount of used training data. In order to increase the generalization capabilities—while simultaneously reducing potential overfitting—of a model, the model complexity reduction is an option. This can be achieved by either changing the model type entirely to a less complex one or by applying regularization. [142] A depiction of the influence of bias and variance when training a model to hit a target in the center and with a small spread in the hits is shown in Figure 2.20.

Avoiding
Overfitting

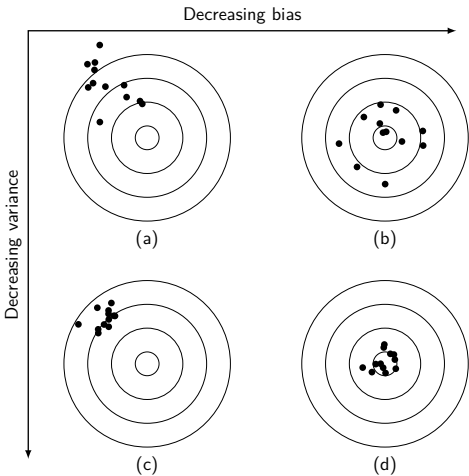


Figure 2.20: Depiction of the effect of the bias and variance contribution to the result of the task of hitting a target multiple times in the center. In subfigure (d), it can be seen that despite low bias and low variance, a certain level of noise remains, which is why not all hits are perfectly centered even in this case.

Regularization

Regularization can be done explicitly by modifying the loss function of an ML model optimization process by including a regularization term $R(f)$, which depends on the model function itself, as in equation (2.50), or implicitly by reducing the model complexity differently. For instance, in a decision tree model, the depth of a tree would be reduced; for an ANN model, the depth and width of the network architecture would be reduced. [141, 143]

Model
Complexity

$$\tilde{\mathcal{L}}_{\lambda}(\mathbf{Y}_A, \mathbf{Y}_B, R(f)) = \mathcal{L}(\mathbf{Y}_A, \mathbf{Y}_B) + \lambda R(f) \quad (2.50)$$

Here, λ is a hyperparameter, which is used to increase or decrease the regularization effect. The loss function could include a generalized regularization term for linear models, as in equation (2.51). This term is defined using the p -norm definition from equation (2.58).

$$\tilde{\mathcal{L}}_{\lambda}(\mathbf{Y}_A, \mathbf{Y}_B, \mathbf{a}) = \mathcal{L}(\mathbf{Y}_A, \mathbf{Y}_B) + \lambda \|\mathbf{a}\|_p \quad (2.51)$$

The vector $\mathbf{a}$ —in the case of a linear model—is the k -dimensional vector containing the slopes for the modeled k -dimensional feature space. This leads to lower slopes for regularized linear models like *e.g.* the Least Absolute Shrinkage and Selection Operator (LASSO) [144] and Ridge [145] regression, as larger slopes result in an increased loss function value. This behavior can even be used to select features based on the linear model's slopes, as less impactful features can be strongly suppressed compared to others using this technique. [141, 143, 146] However, when it comes to validation and evaluation, all available methods rely on metrics to measure the model's performance, given the model's predicted labels and the corresponding true labels. Metrics which play a key role in this thesis are introduced in the following.

Feature Selection

Metrics for Regression Tasks

There are various metrics $\mathcal{M}$ available for use in regression task loss functions. Commonly used metrics include the mean absolute error (MAE), as shown in equation (2.52), the mean squared error (MSE), as shown in equation (2.53), and the coefficient of determination R^2 (equation (2.54)). In fact, due to its differentiability and simplicity, the MSE is used within most model's loss functions. However, as the MSE is sensitive to outliers, there might be instances where it is not the best choice. The word “mean” in the names of both the MAE and the MSE refers to the fact that the error sums are divided by the number of computed errors for the N labels in each of the label vectors $\mathbf{Y}_A$ and $\mathbf{Y}_B$. [147–149]

Mean Squared Error

$$\mathcal{M}^{\text{MAE}}(\mathbf{Y}_A, \mathbf{Y}_B) = \frac{\|\mathbf{Y}_A - \mathbf{Y}_B\|_1}{N} \quad (2.52) \quad \mathcal{M}^{\text{MSE}}(\mathbf{Y}_A, \mathbf{Y}_B) = \frac{\|\mathbf{Y}_A - \mathbf{Y}_B\|_2^2}{N} \quad (2.53)$$

$$\mathcal{M}^{R^2}(\mathbf{Y}_{\text{True}}, \mathbf{Y}_{\text{Pred}}) = 1 - \frac{\|\mathbf{Y}_{\text{True}} - \mathbf{Y}_{\text{Pred}}\|_2^2}{\|\mathbf{Y}_{\text{True}} - \mathbb{E}[\mathbf{Y}_{\text{True}}]\|_2^2} \quad (2.54)$$

Each metric has different advantages and drawbacks, which is the reason different metrics are used for disparate challenges. Especially for the use of loss functions, the differentiability of the chosen metric is advantageous for the implementation of the optimization problem, as the minimums of convex differentiable functions are straightforward to compute. The advantages and disadvantages that come with each metric are compared in detail in the Tables 2.1, 2.2, and 2.3 and are part of the process to decide which metric should be chosen for a given problem. As the coefficient of determination from equation (2.54) is not symmetric, the dependencies on the true data labels $\mathbf{Y}_{\text{True}}$ and the model predictions $\mathbf{Y}_{\text{Pred}}$ of the R^2 metric are displayed explicitly.

Advantages of the MAE	Drawbacks of the MAE
• Due to the fact that the MAE is calculated using a p-norm with $p = 1$, the MAE has the same unit as the predicted quantity. • All errors are weighted linearly based on their size. Hence, average performance will be better. • Despite being linear in the scaling, deviations in either direction are treated equally due to the absolute value being taken.	• As all errors are weighted linearly, there is no inherent suppression for outliers. Hence, outliers with large deviations have a larger impact. • Non-differentiable expression, which makes optimization more difficult to implement.

Table 2.1: Advantages and disadvantages of the MAE.

Advantages of the MSE	Drawbacks of the MSE
• Outliers are suppressed, as the squared deviation is part of the loss function. • Differentiable expression, which makes optimization easier.	• Large individual outliers can strongly alter the scale of the error, which can lead to a model that is overruled by another despite a good performance apart from the individual outliers. • Given in squared units of the target quantity.

Table 2.2: Advantages and disadvantages of the MSE.

Advantages of R^2	Drawbacks of R^2
• R^2 is scale-independent, which makes it easy to interpret. The closer to 1 the value of R^2, the better. • The coefficient of determination quantifies how well the target quantity is determined by the features using the examined model.	• The scale and unit of the target variable are completely omitted, which gives no insight into how large an actual error could be. • The coefficient of determination is not defined for single-sample applications.

Table 2.3: Advantages and disadvantages of the coefficient of determination.

Metrics for Classification Tasks

For binary classification tasks, many metrics could be used to determine the goal of an optimization, depending on the target. The model's predicted labels are compared to the actual labels to determine the metrics. Based on this comparison, the number of correct predictions for both classes—one is assumed to be a positive class p and the other one assumed to be the negative class n —is given by the number of true positives T_p and true negatives T_n . Subsequently, assuming the model is imperfect and misclassifies labels, the number of false negatives, meaning the model assigned a data point to the negative class n while the positive class p would have been correct, is denoted by F_n . The number of false positives, meaning the model assigned a data point to the positive class p , while the negative class n would have been correct, is denoted by F_p . While in some cases only the accuracy $\left(\frac{T_p+T_n}{T_p+T_n+F_p+F_n}\right)$ is of relevance, in other applications it can be important to avoid false negatives or false positives. Common metrics are shown in Table 2.4 using a schematic confusion matrix.

Accuracy-score

		Ground Truth		
		True Label Positive	True Label Negative	
Predictions	Prediction Label Positive	True positive T_p (Correct)	False positive F_p (Incorrect)	Precision $\frac{T_p}{T_p+F_p}$
	Prediction Label Negative	False negative F_n (Incorrect)	True negative T_n (Correct)	Negative Predictive Value $\frac{T_n}{T_n+F_n}$
		Recall $\frac{T_p}{T_p+F_n}$	Specificity Rate $\frac{T_n}{T_n+F_p}$	

Table 2.4: Confusion matrix incorporating different ratios which could be used as metrics for a binary classification problem.

Not only the optimization goal should be considered but also the nature of the data set. Class imbalances can greatly impact a metric's meaningfulness and need to be considered before an optimization goal is set. Beyond the metrics shown in Table 2.4 and the accuracy, the F1 score should be mentioned. The F1 score can be interpreted as the harmonic mean between precision and recall and is given by equation (2.55). The F1 score is especially useful for data that exhibits a strong class imbalance, *i.e.* where one class is much more populated than the other. [150]

F1 score

$$\mathcal{M}^{F_1}(T_p, F_p, F_n) = \frac{2T_p}{2T_p + F_p + F_n} \tag{2.55}$$

For the accuracy, recall, and F1 scores it should be mentioned that these scores do not follow the loss function convention that a larger score corresponds to worse model performance. Hence, to fulfill this convention, these metrics require to be multiplied with a factor -1 .

Categorical Cross-Entropy

Beyond the confusion matrix scores, also the categorical cross-entropy is commonly used as a loss function for classification tasks. In the binary case ($y_i \in \{0, 1\}$), the cross-entropy loss function would be given by equation (2.56).

$$L^{\text{CE}}(\mathbf{y}_{\text{True}}, \mathbf{p}_{\text{Pred}}) = - \sum_{i=1}^N (y_i^{\text{True}} \log(p_i^{\text{Pred}}) + (1 - y_i^{\text{True}}) \log(1 - p_i^{\text{Pred}})) \quad (2.56)$$

In equation (2.56), p_i^{Pred} denotes the probability with which the model predicted the class 1 for the i^{th} label vector entry. This means if the used model would estimate a high chance that the label 1 is correct for the i^{th} label, the probability will be close to 1. However, if the underlying model estimates the label is likely 0, the probability would be close to 0. This way, cross-entropy penalizes wrong labels and increments the loss if the model is very certain about a wrong prediction. This probabilistic approach to loss estimation is well suited for such models, which estimate the prediction's probability—and hence provide a measure of confidence—as they are naturally probabilistic by construction. Cross-entropy could even be used as a loss function for models that are not inherently probabilistic, as it is possible to estimate the corresponding probabilities in several cases. [151–155] Furthermore, cross-entropy can be generalized to non-binary classification problems.

Probabilistic Models

CV

Hyperparameter Optimization

Using a single validation set, as illustrated in Figure 2.19 already detaches the training process from the hyperparameter optimization procedure. However, a consequence of a single validation set is that a large proportion of the data is involved in the parameter training, but a much smaller portion is involved in the hyperparameter selection. To use a larger fraction of data for hyperparameter optimization without reducing the amount of training data used for model parameter training, k -fold CV is used. Like in the holdout method, a dedicated test set is first separated from the whole data and not used in the training and hyperparameter selection steps. The training and validation set is split into k similarly sized data sets in the following step. Now, using the k data sets, the model will be trained on all data except for one set, as this particular set is used for validation afterward i.e. to calculate a validation score based on the predictions for the validation set features and a chosen metric. This pattern is repeated until each data set has been used for validation. This results in k score results, calculated using the metric $\mathcal{M}$. To determine the overall performance for a choice of hyperparameters, all the calculated scores, which resulted based on the chosen hyperparameters, have to be averaged to an overall CV score. This procedure is depicted simplified in Figure 2.21. [156]

k -fold CV

Leave-One-Out

Hence, the validation procedure using a single fixed validation set is fully incorporated into the CV scheme. This scheme can be performed to the extent of the leave-one-out cross validation, where the validation set size is exactly one data point. However, this is also a computationally very expensive procedure, especially for large data sets. In fact, depending on how many combinations of hyperparameters are subject to the validation process, this can be a computationally expensive procedure even without the use of a leave-one-out CV. There is no general rule on how many CV folds should be performed. Some sources suggest fold sizes ranging from 3 to 10 folds depending on the given problem, the computational time a single model training process takes, and the amount of available data. [157, 158]

Fold Sizes

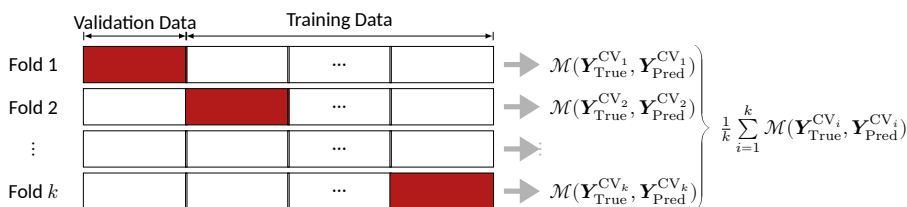


Figure 2.21: Schematic depiction of a CV that is used to determine hyperparameters, using multiple validation data sets and averaging over all evaluations afterward.

After the hyperparameter selection procedure has finished, the data that has been used for training and hyperparameter selection processes can be used to retrain the entire model for a final evaluation on the test set using the determined hyperparameters. Beyond retraining, after the hyperparameter optimization and evaluating the CV performance of a model, it is possible to use the test set for a final evaluation of unseen data and afterward retrain the model using the chosen hyperparameters with the complete data set. However, while this is favorable for small data sets as additional training data has the capabilities to reduce the generalization error [159] of the model, and while it is likely—assuming proper data randomization—that retraining the chosen model will increase the model’s performance in the predictive modeling application, it has to be kept in mind that the retrained model is in principle not validated and evaluated in this case. Some authors also suggest using only training and validation sets in combination with CV for ML applications with only comparatively small amounts of data available. [159] However, this way, the CV score is the only indicator of how the model would perform on unseen data, as the CV procedure has been involved in the hyperparameter optimization process, the CV score obtained from the validation step cannot be viewed as unbiased as a test score would be. On the other hand, it could also be argued that after performing the CV, an additional test set, in principle, only adds another evaluation fold to the CV procedure, but without training on the test in the remaining CV folds. From two of the applications, in the results section of this thesis, it can be seen that models that are retrained on the entire available data after CV-based hyperparameter selection and testing can indeed be successfully applied to materials science applications. However, the predictions of such retrained models should always be treated with care, even though the full data retraining procedure for finalizing an ML model before it is deployed in production for predictive modeling is standard procedure in industry applications. [160]

Retrain After CV

Retrain on Full Data

2.5.2 Model Types

ML methods feature a plurality of models that can be used to tackle diverse academic and industry challenges. The zoo of ML models contained in the open-source Python scikit-learn library alone features more than 100 estimators. [114] Combining this knowledge with the famous “no free lunch” theorem, which essentially includes the consequence that it is impossible to know beforehand which model will fit a given data set best, this concludes that to find the best possible estimator for a given problem it would be required to test and evaluate all estimators. It is clear this is an exhaustive task to complete. Hence, the goal of ML is not to find the best imaginable model but to find an estimator that represents the relations of the underlying data up to a standard that ensures the intended application purpose can be fulfilled with the desired confidence. [161]

Best Estimator

Often, at the beginning of a study, various models are trained on a given data set to evaluate their individual performance. This practice is sometimes referred to as the “shotgun approach”, which translates to the procedure of trying various models on the data to determine which models are performing well and then proceeding with the knowledge obtained from this approach.

ML models can be classified as parametric or non-parametric models. The difference between these classes is that parametric models assume that the underlying true mapping follows a probability distribution, which is defined by a finite and fixed number of distribution parameters. In contrast to parametric models, non-parametric models only assume the smoothness of the underlying true mapping. [162, 163] A visual representation of a selection of models is given in Figure 2.22. In the following, a selection of model types is briefly introduced, which are of relevance to the results acquired during this thesis.

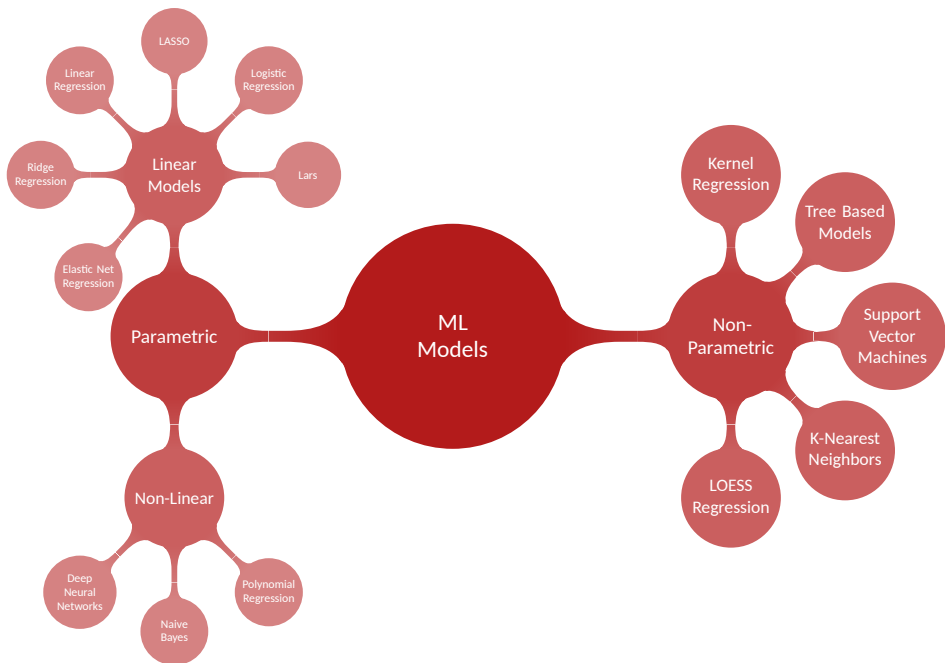


Figure 2.22: Overview over various ML models, including a rough classification of the models in parametric and non-parametric, as well as linear and non-linear in the parametric case. As non-parametric models do not assume an underlying parametric true mapping of the fitted data, they cannot be classified as linear or non-linear mapping models.

Linear Models

Linear models are characterized by the fact that they assume a linear relationship between the data features and the corresponding labels. In a regression task with a single regressand, the general formula that unifies linear models and links the feature vector x to the target quantity y is given by equation (2.57).

$$y = x \cdot a + b \quad (2.57)$$

In equation (2.57) b denotes the constant intercept and the vector $\mathbf{a}$ contains the slopes corresponding to the individual features. Both the intercept and the slope vector are fitted in the loss function minimization process. Let us recall the regularized loss function from equation (2.51), where the regularization term contains the slope vector $\mathbf{a}$. If we choose to use the p -norm as defined in equation (2.58) for an n -dimensional feature vector $\mathbf{x}$, the loss function for the ridge regression for the choice ($p = 2$) and, likewise, the loss function for the LASSO regression for the choice ($p = 1$) can be retrieved. [144, 145]

Loss Function
Minimization

p -Norm for
Regularization

$$\|\mathbf{x}\|_p := \left(\sum_{i=1}^n |x_i|^p \right)^{\frac{1}{p}} \quad (2.58)$$

The elastic net regression's loss function can be acquired summing the regularization terms $p = 1$ and $p = 2$ from ridge regression and LASSO regression together into the loss function, each with an individual scaling term for the regularization strength, λ_1 and λ_2 , respectively. [164]

Decision Tree Based Models

Decision trees use nodes in a graph to split the data based on a criterion assigned to each node that is not a terminal node to determine a non-parametric decision-based model to tackle classification and regression tasks. Each decision made on a node involves a feature and aims to split the data. This splitting subsequently leads to a spatial separation of the feature space into terminal regions, which correspond to the individual terminal nodes. Starting from a root node which contains all data provided to the model, decision trees use a greedy algorithm [165] to determine the locally optimal feature and threshold combination to split and distribute the data to child nodes. Hence, all future decisions depend on previous node decisions, as no backtracking is performed, which includes that there is no guarantee that a globally optimal split is determined during the modeling procedure. An example of a simple tree, including different decision thresholds and features, is given in Figure 2.23. A simple implementation of this greedy algorithm approach would be to iterate over every feature and all possible decision thresholds, which are located in between all neighboring values of an individual feature. However, to determine the locally optimal split criterion in this algorithm, some kind of metric is required to compare the split performance. For the purpose of evaluating splits, the information gain ΔH is defined in equation (2.59) for each non-terminal node m . [114, 166–168]

Data Splitting on
Nodes

Greedy
Algorithm

Information Gain
 ΔH

$$\Delta H_m = H_m - \frac{1}{\Omega_m} \sum_{i \in C(m)} H_i \quad (2.59)$$

H_m in the classification case denotes the information theory based entropy [155] for the data $D(m)$ assigned to the node m . In the regression case, the variance for the data on node m , Ω_m denotes the number of data points assigned to node m , while $C(m)$ denotes the set of all child nodes of node m . In a classification task, H_i is closely related to the cross-entropy loss function from equation (2.56). The entropy and variance expressions for an individual node, respectively, are given in equation (2.60), with p_{ij} denoting the probability of a data point of class j occurring in node i . For unseen data—outside the training process—this quantity is given based on the class occurrences of the training set. [114, 167]

Entropy

Probability in
Decision Trees

$$H_i^{\text{Class}} = - \sum_j p_{ij} \log(p_{ij}) \quad H_i^{\text{Reg}} = \sum_{y \in D(i)} (y - \mathbb{E}[y])^2 \quad (2.60)$$

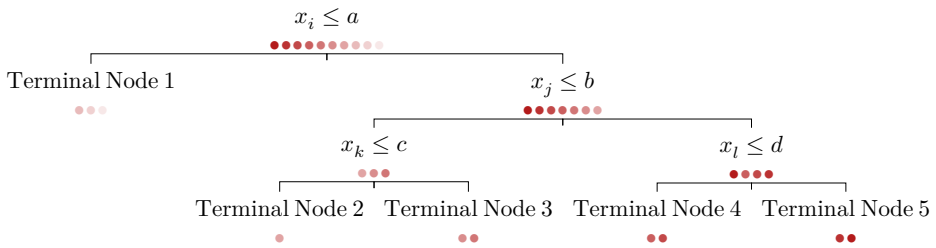


Figure 2.23: Exemplary depiction of a simple decision tree for a regression task to predict the redness of given colors based on some arbitrary features. This illustrates the splitting at each non-terminal node and grouping within terminal nodes, as well as the feature space separation using the splitting conditions. The groups forming at terminal nodes show similarities in their properties. This is a result of the attempt to split the data in a way that the information gain from each splitting is as large as possible.

The sum over the respective node's loss functions over the whole tree T gives the loss of the entire tree. [114] However, during training, the entire tree loss at once is not optimized, but the information gain of each node splitting is maximized. Intuitively, a negative information gain implies that additional data splitting does not provide additional predictive power on the training set. [114, 167] Hence, a node becomes a terminal node when either:

- The information gain of all possible splittings is negative.
- A node only contains a single class (classification) or a single numerical value (regression). (So-called pure nodes)
- An additional splitting of the data at the current node would lead to the tree exceeding the maximum tree depth hyperparameter.

Setting the maximum depth of a tree is intentionally used to avoid overfitting of the training data set. So-called decision stumps—decision tree models with only a single decision performed—are commonly used as weak learners [169, 170] in some boosting and bootstrap aggregation methods. Boosting and bootstrap aggregation methods belong to the ensemble model methods, which are discussed in the following section. [171, 172] It is worthwhile to note that the decision tree model does not always manage to end up with all terminal nodes as pure nodes when regularization is applied, for instance, by limiting the tree depth. In these cases, the prediction for unseen data follows the majority vote (classification) or the mean (regression) of all the training labels in the terminal node that the unseen data point ended up in.

Ensemble Models

Ensemble models do not rely on a single underlying model but combine multiple models to increase the overall predictive power and performance. This section briefly introduces the ensemble methods of stacking, bootstrap aggregation, and boosting. In stacking models, multiple models combined—even such based on completely different methods—feed their predictions into a meta learner, which combines the predictions of the first learning layer to determine the actual prediction based on the provided features. Typically, stacking models perform as well as the best estimator in the first learning layer but are capable of exceeding the predictive power of the best estimator. [114, 173, 174] The scheme that is underlying a stacking model is shown in Figure 2.24.

Emergence of
Terminal Nodes

Avoiding
Overfitting in
Tree Models

Ensemble
Methods

Meta Learner

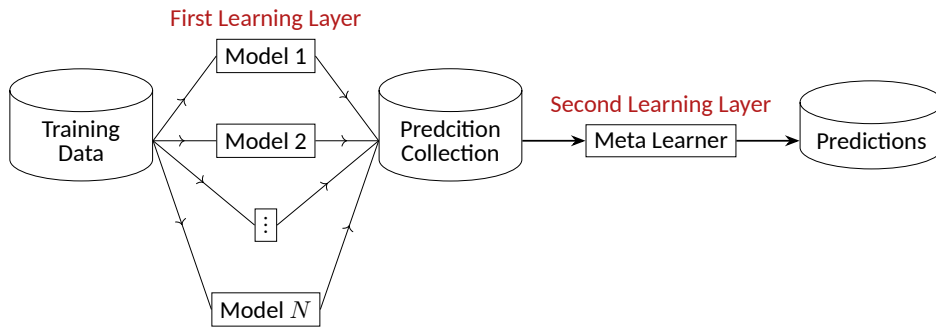


Figure 2.24: Schematic depiction of the training process of a stacking model. The first learning layer does not limit the N models to a single model type. The predictions of the individual estimators from the first learning layer are collected and used as a feature vector to make the final prediction of the stacking model using the meta-learner of the second learning layer. The depicted model architecture can be used for both regression and classification tasks. Cylinders depict data being used at different points of the stacking model's architecture.

Bootstrap aggregation (bagging) methods involve the distribution of random subsets of the training data onto N individual estimators of the same kind, collecting their predictions and aggregating them together. Aggregation, in this case, means that the N estimators' predictions are either averaged (regression) or determined by a majority vote (classification). This is also the procedure of an unseen feature vector that would be predicted by the bootstrap aggregation estimator. [175] Random subsets with regard to the training set means that a random subset of the training data is used to train each estimator, which is part of the bagging model. If the estimators are decision trees, then this fact ensures that the base estimator decision trees do not all share the same architecture and decision thresholds. [176] Often, decision tree models are used as base estimators in bootstrap aggregation models. A depiction of the training process using multiple tree estimators T_i and randomized training data subsets X_i is shown in Figure 2.25.

Random
Sub-Sampling

As already mentioned, decision tree models use a greedy algorithm to determine the locally optimal split of a node. This can lead to very similar trees with only slight deviations in the used training subset. To avoid this, the random forest model uses a specified number of decision trees which are not only trained on a random subset of the training data, but it also randomly determines which features are taken into account for each split of tree nodes. This random component aims to decrease the variance. The implementation of the random forest model in the popular scikit-learn package, however, does not perform a simple majority vote in the classification case but performs an averaging of the probabilistic classification output for each base estimator. [114, 177] Beyond the random forest model, the extremely randomized trees (extra trees) model is included within the scikit-learn package. In the extra trees model, not only the feature sampling for the splitting is performed randomly, but also the determination of the decision thresholds is randomized. This leads to additional randomization and eventually to an expected additional decrease in variance, but in consequence, a slight increase in bias. As the computation of the decision boundaries is omitted, the model's computational efficiency is also slightly increased in comparison to random forest models. [114, 178]

Random Forests

Extremely
Randomized
Trees

Decreased
Variance

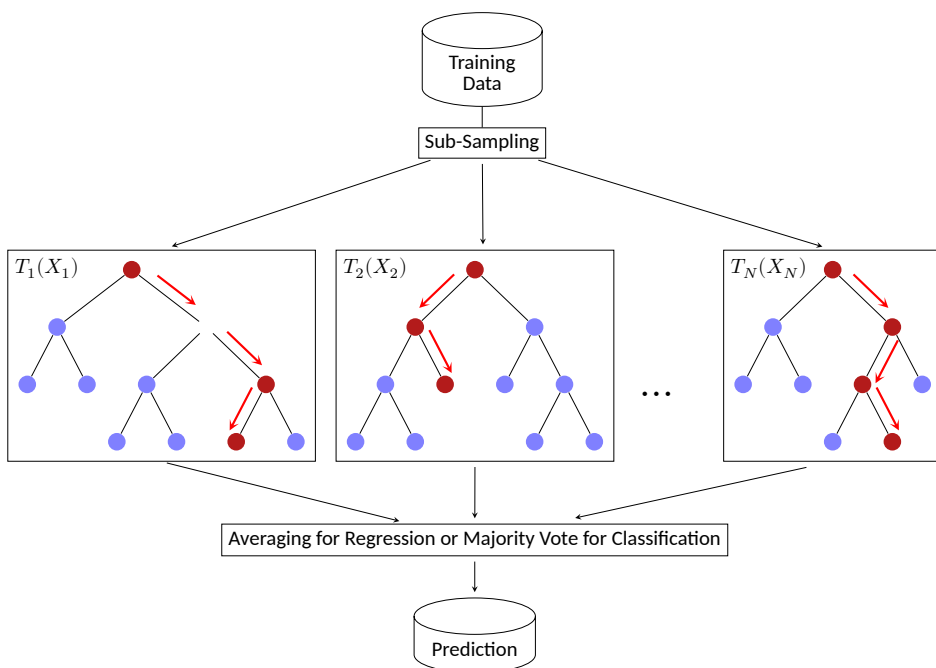


Figure 2.25: Depiction of the training process for a bootstrap aggregation modeling, using decision tree base estimators. It can be seen that from training on different subsets of training data points X_i , different base estimators T_i emerge. Combining the predictions of all base estimators by majority vote or averaging allows the bootstrap aggregation model to make predictions using all N base estimators on unseen data. Cylinders depict both model input and prediction data.

Beyond stacking and bagging, a commonly used ensemble method is boosting. Unlike stacking and bagging, where the base estimators are predicting a quantity independent of the predictions of other base estimators, boosting involves base estimators correcting the errors of other estimators. This requires that the model predictions are computed not in parallel, but rather in serial. The idea behind boosting is that each model makes a prediction and also computes some kind of weight for each training data point. This weight is increased if the underlying base estimator fails to predict the individual data point correctly or decreased if the prediction of the estimator is accurate. This causes weak learners to focus on learning the properties of more difficult data points, as they get falsely predicted more often. Essentially, the weights serve as an incentive for the model's base estimators to correct the predictions of data points that previous estimators failed to accurately predict. This implements the concept of self-correction for each subsequent tree. [114, 170, 179] A depiction of the model training process and corresponding architecture is given in Figure 2.26. In order to compile a prediction, in boosting the predictions of all models and the weights which resulted from the individual base estimators weaknesses are combined to a weighted overall prediction.

Boosting

Incentive for
Self-Correction

Model Weakness

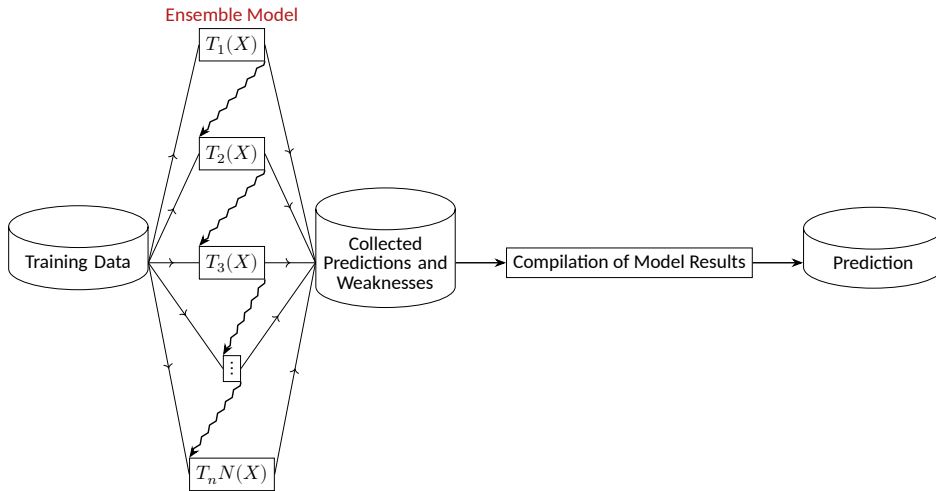


Figure 2.26: Depiction of a boosting estimator training process using N weak tree-based estimators T_i . The wavy line passing from the previously evaluated tree T_{i-1} to each subsequent tree T_i denotes the information transfer from the neighboring weak estimators about the predictive weaknesses of the predecinging tree T_{i-1} . Using the passed information about the weaknesses of the previous models, the learning objective of the latter trees can be adjusted accordingly to predict more difficult data points correctly. Cylinders depict data being used at different points of the boosting model's architecture.

The weight that a base estimator receives in the final prediction depends on the estimator's performance in comparison to the other base estimators. Hence, the use of weights has two advantages:

- The model's base estimators are required to determine a way to predict difficult data points as the corresponding weight increases. [180]
- The weighted process of prediction assigns more importance for the overall result to accurate base estimators. [180]

Both bagging and boosting aim to decrease the variance of their predictions. But it has been established that boosting outperforms bagging when it comes to decreasing the variance. [181] There are different algorithms and implementations of boosting ensemble models available. An example of a boosting algorithm is Adaptive Boosting (AdaBoost) [182]. In AdaBoost, the weights for data points are initialized uniformly and automatically adjusted for each set-up base estimator. A later iteration in the history of boosting methods is represented by the gradient-boosted decision trees (gradient boosting) method. This method, introduced in the following section, extends the boosting method beyond the computation of the weights to the optimization of a continuous and convex loss function. Whereas AdaBoost relies on a specific, exponential loss function, gradient boosting does not rely on a majority vote or average to predict unseen data but on an additive procedure. [180]

AdaBoost

Gradient-
Boosted Decision
Trees

Additive
Prediction

Gradient Boosting

While gradient boosting represents a boosting algorithm, there are subtle differences when compared to earlier implementations of boosting like AdaBoost. Some of them include:

- Gradient boosting is not limited to tree stumps as base estimators, but typically uses deeper trees, i.e. decision trees with more than a single decision node.
- Individual base estimators are also multiplied by a weight and a learning rate α before they are compiled into the overall model result. However, the weight is chosen by an optimization step for all base tree estimators except the first estimator (Typically referred to as estimator $m = 0$). In gradient boosting this m^{th} base-estimator's weight is denoted as β_m and does not directly indicate each base estimators predictive capabilities, as in e.g. AdaBoost.
- All base estimators—except the first one—do not actually learn to predict the target quantity from the training set, but rather learn to predict a pseudo-residual that results from all previous predictions compared to the actual label of the individual data point in the training data.
- In order to optimize the model, gradient descent [183] is used, which requires the loss function to be convex and differentiable. [184]

The overall model prediction at base estimator m is given by the expression in equation (2.61) and depends on the prediction of the previous base estimator $m - 1$, as well as the residual prediction of model m for the data point i , which is denoted as r_{mi} . The residual approach can be used for both regression and classification problems. In the classification task, the pseudo-residuals are actually the residuals of the probabilities, determined by the model, for data points x_i to lie within a given class, compared to its true class. [185]

$$f_m(x_i) = f_{m-1}(x_i) + \alpha\beta_m r_{mi} \quad (2.61)$$

Hence, the prediction of base estimator tree m' of the boosting model is given by equation (2.62). Therefore, the ensemble prediction is based on the initial guess from the single-node estimation at $m = 0$ and then adds the predicted residuals from each base estimator scaled by the learning rate α and the corresponding estimator's weight. This way, the initial guess is iteratively corrected by each base estimator one after another. [185]

$$f_{m'}(x_i) = f_0(x_i) + \sum_{m=1}^{m'} \alpha\beta_m r_{mi} \quad (2.62)$$

Using equation (2.62) the prediction of the overall model can be determined by setting $m' = N$. However, at this point, it is unclear how the r_{mi} are acquired. In order to understand where the residuals come from, a loss function has to be defined. The loss function typically used in regression during gradient boosting is the MSE, similar as shown in equation (2.53). Computing the loss function—using the convention that a single data point is denoted via (x_i, y_i) with $i \in [1, N_{\text{Dat}}]$ within the training set—at base tree estimator m' the equation (2.63) emerges. For classification tasks, the cross-entropy loss from equation (2.56) is commonly used. Again, the loss function of the entire boosting model can be acquired setting $m' = N$. [185]

$$\mathcal{L}_{m'}(\mathbf{y}, \mathbf{y}_{\text{Pred}}^{m'}) = \frac{1}{N_{\text{Dat}}} \sum_{i=1}^{N_{\text{Dat}}} \mathcal{L}(y_i, f_{m'}(x_i)) = \frac{1}{N_{\text{Dat}}} \sum_{i=1}^{N_{\text{Dat}}} (y_i - f_{m'}(x_i))^2 \quad (2.63)$$

In equation (2.63) the relation shown in equation (2.64) has been used.

$$\mathbf{y}_{\text{Pred}}^{m'} = \sum_{i=1}^{N_{\text{Dat}}} f_{m'}(\mathbf{x}_i) \mathbf{e}_i \quad (2.64)$$

In this representation $\mathbf{e}_i$ denotes the unit vector with which the label vector $\mathbf{y}$ can be decomposed. Generally, the pseudo-residuals used for training the m^{th} base estimator are given by the derivative in equation (2.65).

$$r_{mi} = - \left. \frac{\partial \mathcal{L}(y_i, f(\mathbf{x}_i))}{\partial f(\mathbf{x}_i)} \right|_{f(\mathbf{x}_i)=f_{m-1}(\mathbf{x}_i)} \quad (2.65)$$

The idea behind the expression in equation (2.65) becomes clear as the MSE-based loss function is inserted to evaluate this expression. This is done in equation (2.66).

$$\begin{aligned} r_{mi} &= - \frac{1}{N_{\text{Dat}}} \left. \frac{\partial (y_i - f(\mathbf{x}_i))^2}{\partial f(\mathbf{x}_i)} \right|_{f(\mathbf{x}_i)=f_{m-1}(\mathbf{x}_i)} \\ &= \frac{2}{N_{\text{Dat}}} (y_i - f_{m-1}(\mathbf{x}_i)) \end{aligned} \quad (2.66)$$

Hence, the residuals which are used to train the m^{th} base estimator are given by the true training labels and the predictions of the previous models $f_{m-1}(\mathbf{x}_i)$ only, which makes the pseudo-residuals simple to compute. However, the applicability of equation (2.65) is not restricted to the MSE-based loss function. The training data for the m^{th} estimator is represented by the tuples $(\mathbf{x}_i, r_{mi}) \forall i \in [1, 2, \dots, N_{\text{Dat}}]$.

In an additional step, the tree weight is determined using an optimization step, as given in equation (2.67) for each base estimator tree. [185]

$$\beta_m = \arg \min_{\beta} \sum_{i=1}^{N_{\text{Dat}}} \mathcal{L}(y_i, f_{m-1}(\mathbf{x}_i) + \beta r_{mi}) \quad (2.67)$$

Hence, the weight β_m is not originating from the performance evaluation of a given tree, as were the weights in AdaBoost, but is chosen such that the loss function containing the previously computed pseudo-residuals is minimized with regard to the weight. Summing this up, gradient descent provides the direction in which the value of the previous prediction needs to be corrected. However, as the pseudo-residuals are scaled by the general learning rate and by the base estimator weights in the overall estimation, a plurality of trees has to be used in order to approach the true value of the target quantity asymptotically. This leads to the mentioned scheme, where each tree corrects the errors of the previous base estimator tree by iteratively optimizing the loss function using gradient descent. In fact, an overly large learning rate can lead to an overshooting prediction instead of an asymptotic approach of the predicted value towards the true label's value. This is intuitively clear as gradient descent provides the correct direction, the corresponding pseudo-residual needs to be corrected towards, but not the exact scale. The scale of r_{mi} obtained in the process depends on the scaling of the used loss function $\mathcal{L}$, which is not absolute but relative. Also, the learning rate can act as a form of regularization to prevent overfitting since it has been found that models using learning rates $\alpha < 0.1$ are more capable of generalization than such with higher learning rates. [179]

Learning Rate

Regularization

As the estimator $f_0(\mathbf{x})$ is given by a single tree node without any decision learned, this first estimator is basically a constant value. The nature of this value is loss function dependent, as the constant initial estimation is given by the expression in equation (2.68). [185]

$$f_0(\mathbf{x}) = \arg \min_{\kappa} \sum_{i=1}^{N_{\text{Dat}}} \mathcal{L}(y_i, \kappa) \quad (2.68)$$

This concludes that, for the MSE-based loss function, the initial prediction is given in equation (2.69).

$$f_0(\mathbf{x}) = \arg \min_{\kappa} \sum_{i=1}^{N_{\text{Dat}}} (y_i, \kappa) = \frac{1}{N_{\text{Dat}}} \arg \min_{\kappa} \sum_{i=1}^{N_{\text{Dat}}} (y_i - \kappa)^2 \quad (2.69)$$

Minimizing the latter expression with regard to κ in equation (2.69) leads to the expression in equation (2.70) that is essentially the mean of all training labels y_i .

$$\kappa = \frac{1}{N_{\text{Dat}}} \sum_{i=1}^{N_{\text{Dat}}} y_i \quad (2.70)$$

Hence, in the case of $\mathcal{L}$ being given by the MSE, κ is given by the average of the label vector. The same reasoning holds true for the output of a node within a base estimator with multiple data points assigned during the training. In this case, the predicted value of the individual node is the average of the assigned data points, as this minimizes the MSE loss function.

Extreme
Gradient
Boosting

Beyond gradient-boosted trees, the implementation of extreme gradient boosting (XGBoost), among others, introduces additional hyperparameters that allow further tuning of the model, as well as tweaks that improve computational efficiency. This more advanced implementation is briefly introduced in the following section.

Extreme Gradient Boosting

Additional
Regularization

The XGBoost [125] method adds two regularization terms to the loss function of each base tree estimator similarly to equation (2.50) by including a term that encourages tree pruning (T_m denoting the number of terminal tree nodes in the m^{th} tree) and the squared norm of the vector ω_m that contains all tree node outputs associated with the m^{th} tree. The regularization used by XGBoost in that sense is very similar to the regularization introduced by [186]. The modified XGBoost loss function $\mathcal{L}^{\text{XGB}}$ can be expressed using the gradient-boosted trees loss function $\mathcal{L}^{\text{GB}}$ from equation (2.63) and combining the loss function with the regularization terms as shown in equation (2.71). [125]

$$\mathcal{L}^{\text{XGB}}(\mathbf{y}_{\text{True}}, \mathbf{y}_{\text{Pred}}, f) = \mathcal{L}^{\text{GB}}(\mathbf{y}_{\text{True}}, \mathbf{y}_{\text{Pred}}) + \sum_{m=1}^{N_{\text{Trees}}} R(f_m) \quad (2.71)$$

$$\text{with } R(f_m) = \gamma T_m + \frac{\lambda}{2} \|\omega_m\|_2^2$$

However, in order to optimize the tree base estimators sequentially, the loss function for individual trees has to be used, which is given in equation (2.72). Again, the relation from equation (2.64) has been used here. [125]

$$\mathcal{L}_m^{\text{XGB}}(\mathbf{y}_{\text{True}}, \mathbf{y}_{\text{Pred}}, f_m) = \frac{1}{N_{\text{Dat}}} \sum_{i=1}^{N_{\text{Dat}}} \mathcal{L}(y_i, f_m(\mathbf{x}_i)) + \gamma T_m + \frac{\lambda}{2} \|\omega_m\|_2^2$$

Hence, the computation of the loss function of the base estimator m essentially is the task of computing the average of all data label losses after the prediction of the m^{th} tree and determining the tree-specific regularization term. Here, T_m denotes the number of terminal nodes in tree m . Both λ and γ are regularization constants. However, rewriting $f_m(\mathbf{x}_i)$ using equation (2.62) and expanding the resulting expression in $f_m(\mathbf{x}_i)$ using a second order Taylor expansion [187] results in the expression as given in equation (2.72). [125, 188]

Second Order
Loss Function

$$\begin{aligned} \mathcal{L}_m^{\text{XGB}}(\mathbf{y}_{\text{True}}, \mathbf{y}_{\text{Pred}}, f_m) \approx & \quad (2.72) \\ & \frac{1}{N_{\text{Dat}}} \sum_{i=1}^{N_{\text{Dat}}} \left(\mathcal{L}(y_i, f_m(\mathbf{x}_i)) + f_m(\mathbf{x}_i) \frac{\partial \mathcal{L}(\mathbf{y}_{\text{True}}, f_{m-1}(\mathbf{x}_i))}{\partial f_{m-1}(\mathbf{x}_i)} \right. \\ & \left. + \frac{1}{2} (f_m(\mathbf{x}_i))^2 \frac{\partial^2 \mathcal{L}(\mathbf{y}_{\text{True}}, f_{m-1}(\mathbf{x}_i))}{\partial (f_{m-1}(\mathbf{x}_i))^2} \right) + R(f_m) \end{aligned}$$

In the representation given in equation (2.72) for the m^{th} tree, both derivatives are evaluated with regard to the tree $m-1$ which makes them a constant for the optimization of the m^{th} tree. Hence, the derivative-originated constant factors are rewritten as ∂_{m-1}^i and $\partial \partial_{m-1}^i$. Also, the zeroth order loss function is a constant in this representation and can, therefore, be omitted for the optimization objective. Therefore, the considered loss function of the m^{th} base estimator tree reads as in equation (2.73). [125]

$$\begin{aligned} \mathcal{L}_m^{\text{XGB}}(\mathbf{y}_{\text{True}}, \mathbf{y}_{\text{Pred}}, f_m) \approx & \quad (2.73) \\ & \frac{1}{N_{\text{Dat}}} \sum_{i=1}^{N_{\text{Dat}}} \left(f_m(\mathbf{x}_i) \partial_{m-1}^i + \frac{1}{2} (f_m(\mathbf{x}_i))^2 (\partial \partial_{m-1}^i) \right) + R(f_m) \end{aligned}$$

Using the earlier notation of $D(l)$ denoting the data subset assigned to the tree node l in the given m^{th} base estimator and the node score ω_{ml} is technically equivalent to the value of $f_m(\mathbf{x}_i)$ when $\mathbf{x}_i \in D(l)$ as the base estimator's prediction for an individual data point solely depends on the corresponding terminal node score the data point ends up in, the previous expression can be simplified as done in equation (2.74). [125]

$$\begin{aligned} \mathcal{L}_m^{\text{XGB}}(\mathbf{y}_{\text{True}}, \mathbf{y}_{\text{Pred}}, f_m) \approx & \quad (2.74) \\ & \frac{1}{N_{\text{Dat}}} \sum_{l=1}^{T_m} \left(\omega_{ml} \sum_{\mathbf{x}_i \in D(l)} (\partial_{m-1}^i) + \frac{1}{2} \omega_{ml}^2 \left(\sum_{\mathbf{x}_i \in D(l)} (\partial \partial_{m-1}^i) + N_{\text{Dat}} \lambda \right) \right) + \gamma T_m \end{aligned}$$

Now optimizing the loss function from equation (2.74) with regard to the individual node score, this concludes that the optimized scores are given by equation (2.75). However, this assumes that the tree structure, including the number of terminal nodes T , remains unchanged. [125]

Optimized Tree
Weights

$$\omega_{ml} = - \frac{\sum_{\mathbf{x}_i \in D(l)} (\partial_{m-1}^i)}{\sum_{\mathbf{x}_i \in D(l)} (\partial \partial_{m-1}^i) + \tilde{\lambda}} \quad (2.75)$$

Within the previous expression, it has been used that λ is a scaling regularization term, which is now combined as $\tilde{\lambda} = N_{\text{Dat}}\lambda$. In order to acquire equation (2.75), the derivative $\frac{\partial \mathcal{L}}{\partial \omega_{ml}}$ was set to zero and solved for ω_{ml} . Hence, the optimization requires the second derivative in ω_{ml} to be positive in order for the solution to indeed present as a minimum. This is true, as the loss function is required to be convex and differentiable. Using the squared error loss function, the optimized scores as contained in equation (2.76) emerge.

$$\omega_{ml} = 2 \frac{\sum_{i \in \{i | \mathbf{x}_i \in D(l)\}} (y_i - f_{m-1}(\mathbf{x}_i))}{\tilde{\lambda} + \sum_{\mathbf{x}_i \in D(l)} 2} \quad (2.76)$$

In order to interpret the expression from equation (2.76), one can look at the numerator and denominator separately. The numerator is proportional to the sum of the residuals acquired using the loss function in the terminal node l . The denominator is proportional to the number of data points—and hence also the number of residuals—in the given terminal node l . It is only accompanied by some constant regularization parameter, which effectively lowers the terminal tree node scores as a measure of regularization to prevent overfitting.

Now, using equation (2.75) and inserting the optimized node scores in equation (2.74) leads—after simplification—to the expression shown in equation (2.77).

$$\mathcal{L}_m^{\text{XGB}}(\mathbf{y}_{\text{True}}, \mathbf{y}_{\text{Pred}}, f_m) \approx -\frac{1}{2} \sum_{l=1}^{T_m} \frac{\left(\sum_{\mathbf{x}_i \in D(l)} \partial_{m-1}^i \right)^2}{\sum_{\mathbf{x}_i \in D(l)} (\partial \partial_{m-1}^i) + \tilde{\lambda}} + \gamma T \quad (2.77)$$

Information Gain in Node Split

This approximation used as loss function allows us to compute the information gain of a split in the case of a binary split. Assuming a parent node P is split by a decision into two child nodes C_1 and C_2 , the information gain of the chosen split is given by equation (2.78). [125]

$$\Delta H = \frac{1}{2} \left(\frac{\left(\sum_{\mathbf{x}_i \in D(C_1)} \partial_{m-1}^i \right)^2}{\sum_{\mathbf{x}_i \in D(C_1)} (\partial \partial_{m-1}^i) + \tilde{\lambda}} + \frac{\left(\sum_{\mathbf{x}_i \in D(C_2)} \partial_{m-1}^i \right)^2}{\sum_{\mathbf{x}_i \in D(C_2)} (\partial \partial_{m-1}^i) + \tilde{\lambda}} - \frac{\left(\sum_{\mathbf{x}_i \in D(P)} \partial_{m-1}^i \right)^2}{\sum_{\mathbf{x}_i \in D(P)} (\partial \partial_{m-1}^i) + \tilde{\lambda}} \right) - \gamma \quad (2.78)$$

The expression from equation (2.78) can easily be acquired by projecting the loss from equation (2.74) on a single node and taking the difference. As both child node losses are subtracted from the loss of the parent node, and each node considered for itself has $T = 1$, only γ remains beside the first and second loss function derivative-dependent terms. Hence, the regularization parameter γ is crucial to the decision if a split is performed or avoided for the underlying base estimator, as increasing γ leads to a depressed information gain, and a negative value for the information gain would result in a split option being rejected.

Besides the discussed loss function properties and the larger degree of regularizability compared to gradient boosting, the implementation of XGBoost [125] has some more benefits to offer, which are not discussed here in detail. Those include:

Features of
XGBoost

- XGBoost has built-in handling for sparse input, which is already taking effect during the split-finding.
- XGBoost is not limited to using an exact greedy algorithm but is also capable of using an approximation.
- XGBoost is highly parallelized regarding split finding in the training process.

These properties make XGBoost well suited for the use on extensive data sets. [125] Also, the initial estimation—as occurring in gradient boosting—is not determined by a minimization but is accessible as input to the model while training. However, if no input is provided, the value of 0.5 is a hard-coded default as an initial estimation for both classification and regression.

2.5.3 Gaining Physical Knowledge from Trained Models

ML models—especially ensemble models—typically learn from provided data and afterward predict unseen data as a black-box function and, therefore, are hard to interpret and explain. However, many applications mandate that decisions can not be made based on a black-box function; instead, they have to be explainable, especially for use cases where people's lives are affected by a model's prediction such as credit and insurance decisions based on ML modeled risk assessment. [189–191] As institutions may be held liable for decisions based on ML models, it is essential that the reasoning behind a model's decision is understood within the institution and also is explainable to those affected by the model output. Hence, the field of XAI emerged alongside ML models. This goes to the extent that some models already come with model-specific explainer routines. For instance, the decision tree models in scikit-learn come with the ability to plot the decision tree, including decisions, as well as the ability to visualize the feature space separation of a given decision tree's decisions. [114] For other models, like LASSO, the modeling coefficients (slopes) can be interpreted as a measure of the importance of individual features. However, while these are feasible options for these specific models, a model-agnostic approach backed by a rigorous mathematical theory underlying the approach is desirable. Luckily, the SHAP package fulfills both conditions and is publicly available as open-source code [128]. In the following sections, a brief introduction to the underlying theory of coalition game theory, Shapley values, and the package's capabilities is given. [13, 128] However, whenever ML models are interpreted, it is crucial to be aware of confirmation bias [192] and not only look for results that support the own hypothesis but also for evidence that has the potential to validate another hypothesis or hints on the fact that the model has not indeed learned relations in the data but rather overfitted artifacts (e.g. noise) that are contained within the data.

Interpretability &
Explainability

Requirement for
XAI

SHapley Additive
exPlanations
Game Theory

Confirmation
Bias

Explainable AI in Natural Sciences

Beyond applications in society and liabilities emerging from the use of ML models, explainability is crucial in understanding underlying mechanisms. Beyond the fact that ML models provide black-box predictions, to add value to the disciplines of natural sciences, predictions have to have a reasoning behind them and need to be understood to contribute towards the accumulation of knowledge.

In natural sciences, discovering a particular underlying pattern is generally worth more than an individual prediction obtained by a black-box model. While there exist applications where it is acceptable to have a black-box model and use the obtained predictions, this is not the case for the natural sciences, as a rule.

XAI methods can be used after a model has been trained on a data set to potentially discover complex and previously unknown relations—that could otherwise be overlooked. The application of the scientific methods does, of course, not stop at the point where an XAI-based approach has found a relation between quantities based on a trained model. Still, XAI can provide the incentive for a further investigation into that specific relation. In that sense, XAI should be seen as an additional tool for scientists to discover knowledge, relations, and reasoning in data. [193, 194]

Coalition Game Theory

Introducing coalition game theory begins by defining the elements of a coalitional game $\mathbb{V}$ with a transferable payoff, as done in definition 3.

Definition 3. “A **coalitional game** with transferable payoff consists of the following elements:

1. A finite set of N players, the so-called coalition.
2. A function v that associates with every nonempty subset of players S of the set N a real number $v(S)$ (the payoff value achieved by the player set S).”

This definition is taken from [195].

In the sense of definition 3, the function $v(S)$ represents a payoff that is achieved by the set of S players in a collective effort. However, the effort put into achieving this payoff—due to their common actions—may differ for each individual in the group. Depending on the subset S chosen from the coalition, there might be individuals who—when working together—add disproportional more value to the group than they would have cumulatively added as individuals in the subset S . There might also be individuals who do not contribute any value to the task. This is where Shapley values come into play, as Shapley values attempt to determine a fair split of the total payoff given by the function $v(S)$, based on the average marginal contribution of each player in all possible subsets S of player coalitions. [195] However, there is the requirement to define what a fair split is exactly. The definition of a fair split suggested by Shapley values is based on axioms and given in definition 4. Definition 4 is taken analogously from [196].

Definition 4. Let be $\varphi_i(v)$ the split that player i gets and v the overall payoff in a game $\mathbb{V}$, then a **fair split** fulfills:

1. $\forall v \in \mathbb{V}$ it holds $\sum_{i \in N} \varphi_i = v(N)$
2. $\forall v \in \mathbb{V} \wedge \forall i \in N$ which fulfill $v(S \cup \{i\}) - v(S) = 0 \forall S \subseteq N$ it holds $\varphi_i(v) = 0$
3. $\forall v, w \in \mathbb{V} \wedge \forall i \in N$ it holds $\varphi_i(v + w) = \varphi_i(v) + \varphi_i(w)$
4. $\forall v \in \mathbb{V} \wedge \forall i \in N \wedge \forall \nu \in \mathbb{R}$ it holds $\varphi_i(\nu v) = \nu \varphi_i(v)$
5. $\forall v \in \mathbb{V} \wedge \forall i, j \in N$ that are symmetric in v , it holds that $\varphi_i(v) = \varphi_j(v)$

Payoff for
Common Actions

Shapley Values

Axioms for
Coalitional Game

These axioms include that a player who did not add to the value acquired in the game $\mathbb{V}$ does not get rewarded afterward. Also, the axioms fulfill the intuition that equally contributing players get equal rewards, as well as that consistency is ensured by the fact that all players' splits add up to the total payoff $v(N)$. Shapley values determine the split based on the previously mentioned axioms and are introduced in the next section.

Shapley Values

The Shapley values can be determined using definition 5, which was taken analogously from [197].

Definition 5. The marginal contribution of an individual player $i \in N$ for a game $\mathbb{V}$ with $v \in \mathbb{V}$ being the payoff value is given by equation (2.79).

$$\varphi_i(N, v) = \sum_{S \subseteq N \setminus \{i\}} \frac{(|N| - 1 - |S|)!|S|!}{|N|!} (v(S \cup \{i\}) - v(S)) \quad (2.79)$$

The interpretation of Shapley values is that the values represent the marginal added value (i.e. the marginal contribution) to the game of the player i in the coalition of players for all possible subsets $S \subseteq N$ including player i , averaged over the number of players and the number of all coalitions without the player i . [195]

Marginal Added Value

Beyond the contribution of the individual player i , it is possible to describe the effect on the interaction of player i with an additional player j using the Shapley interaction index as shown in equation (2.80). [128, 198]

Shapley Interaction Index

$$\varphi_{ij}(N, v) = \sum_{S \subseteq N \setminus \{i, j\}} \frac{(|N| - 2 - |S|)!|S|!}{2|N| - 1!} (v(S \cup \{i, j\}) - v(S \cup \{i\}) - v(S \cup \{j\}) - v(S)) \quad (2.80)$$

The interaction indices are related to the Shapley values as shown in equation (2.81). [128]

$$\varphi_{ij}(N, v) = \varphi_i(N, v) - \sum_{j \neq i} \varphi_{ij}(N, v) \quad (2.81)$$

This also includes that summing a Shapley interaction index of two players i and j over one of both players results in the Shapley value for the other player. This is very similar to the construction of the Shapley values φ_i of individual players, which add up to the total payoff value.

After this small dive into coalitional game theory, it remains to discuss how Shapley values should contribute to ML models' explainability. If the individual players are interpreted as single features, and the set N contains all features of a data set an ML model has been trained on, then the ML model can be related to the payoff value function $v(S)$. While this relation is true in principle, there is more to it than simply replacing the payoff value function with the model, as discussed in the following section. In fact, the SHAP software package is capable of computing Shapley values with low-grade polynomial time for tree-based models using knowledge about the structure of the underlying decision tree models. This is useful, as going through all possible combinations of features can be computationally extremely expensive for large feature sets and actually presents as an NP-hard problem. [199]

Optimization for Tree-Based Models in SHAP

However, in principle, SHAP can be used model-agnostic but with dedicated implementations for certain models, including tree-based models, linear models, and DNN models. [128]

SHAP Software Package

Additivity

As included in the name of the SHAP package (Shapley additive explanations), Shapley values are used to explain ML models additively. The meaning of this additive property can be understood by looking at equation (2.82). For this equation, may $f(x)$ be the model to be explained, x a feature vector, and X the set of all features. [128, 200]

$$f(x) = \mathbb{E}[f(x)] + \sum_{i \in X} \phi_i(f(x), x) \quad (2.82)$$

In equation (2.82) the function ϕ_i denotes the SHAP value attributed to the feature i . The SHAP values are based on the concept of Shapley values but denote the conditional expectation function corresponding to the model $f(x)$. [200] Hence, SHAP values give a measure of the contribution of each feature to the shift of each prediction of the model relative to the expectation value. All the shifts combined then result in the overall prediction. In fact, SHAP has built-in visualization routines that allow the user to use SHAP values for both local (single data-point) [201] and global understanding of the model's prediction. Additionally, interactions of features can also be visualized in order to understand model predictions. When using SHAP, it is important to understand that SHAP does not detect causation between features and target quantity but rather explains which features gave rise to the model's prediction to which degree. This is intuitively clear, as the trained model is not identical to the underlying real-world distribution of the target variable.

Local and Global Explanations

SHAP Values Approximation for Independent Features

SHAP is also capable of approximating SHAP values when the exact computation procedure is not feasible. However, this approximation assumes that features are independent of each other. This translates to the interpretation that for all features, the Shapley interaction indices would be zero everywhere except on the diagonal. This is, however, unlikely for a model that is trained on real-world data with more than a single feature. [200]

Bias-Variance Trade-off

Another consideration before applying XAI techniques in general regards the model's accuracy. In fact, while linear regression models appear to be easily interpretable, a simple linear model might not capture the underlying relations contained in the training data enough to lead to a meaningful interpretation. This actually relates to the bias-variance trade-off, as low bias models tend to be more explainable. [128, 202]

2.5.4 Availability of Data

Sequentially Available Data

Online & Incremental Learning

In many cases, when dealing with materials screening applications, the data used for a research project is often not (or not entirely) available at the beginning of the study. Usually, the data is either acquired during the project or becomes available sequentially over time. There are different ways to deal with these particular circumstances. The simplest and most obvious procedure is to retrain the used model occasionally as the data set is updated. However, this approach is far from systematic. Another approach would be online learning. [203] In online learning, ML models continue to learn as data becomes sequentially available. This is also known as incremental learning, as the models are not retrained on the entire data set but on the additional data that became available. [204]

However, as it was feasible to retrain the acquired models entirely, including the additionally acquired data, for the course of this doctoral project, it was not necessary to implement incremental learning. Hence, the method of choice in the cases where data became available sequentially during this project was the batch learning method, which will be introduced in the following section. However, if the data characteristics are not expected to change rapidly compared to the training set, the flexibility and adaptive capabilities an online learning algorithm provides are not required.

Batch Learning

In batch learning—opposite to online learning—additional data that becomes available posterior to the training process of an ML model, the model is not trained incrementally on the additionally available data only but retrained on the entire available training data set. This implicates an increased computational cost for the retraining process than in an online learning scheme. However, depending on the use case, this is outweighed by the fact that after retraining the model on the updated data, an evaluation of the retrained model is performed, intended to serve as an accuracy-ensuring measure. It further allows for tracking the model's predictive power as the amount of training data increases. For online learning approaches with ANNs, it is known that a phenomenon called “catastrophic interference” [205] can occur, where a model forgets a previously learned relation when learning incrementally. This can lead to a significant increase in prediction errors. A scientific application requires continuous monitoring of a model's performance, which is easier to maintain in the batch learning approach. The schematic workflow of batch learning is shown in Figure 2.27. [206] Batch learning can also be used as an iterative procedure that is performed as the necessity arises from an additionally acquired amount of data. The paradigms, which indicate when a model should be retrained, vary. Possible strategies can include a certain time since the last retraining step, the number of newly accumulated data points, and milestones in the research projects. The most extreme case would be if each additional data point acquired immediately triggers the model to be retrained on the entire data, including the new data point. Automatized model training and evaluation cycles can be implemented using Machine Learning Operations (MLOps) [207] techniques, which aim to deploy ML models and predictions in an automated, continuous fashion. Using automated training and evaluation pipelines allows real-time tracking of resulting predictions and the predictive performance with each change to the model or the acquired data. Continuous tracking allows for more informed decisions regarding using the trained models. For example, a continuously monitored test performance can indicate whether the retrained model's performance is indeed increased compared to the previous model. It can thus indicate whether switching to a retrained model is advised. [206, 207]

Retraining on
Updated Data

Catastrophic
Interference

MLOps

Missing Data

Besides the constraint of more data becoming available at a later stage during a research project, it can also happen that features of individual data points or labels are missing in a given data set. If it is possible to obtain the correct (or at least meaningful by e.g. interpolation [208, 209]) entries for the missing values, this is, of course, preferred. However, as this is not always possible, it can be necessary to employ sparsity aware ML algorithms (like e.g. XGBoost [125]) if it is likely that possible applications of the model will require the model algorithm to deal with missing values. [210]

Sparsity Aware
Algorithms

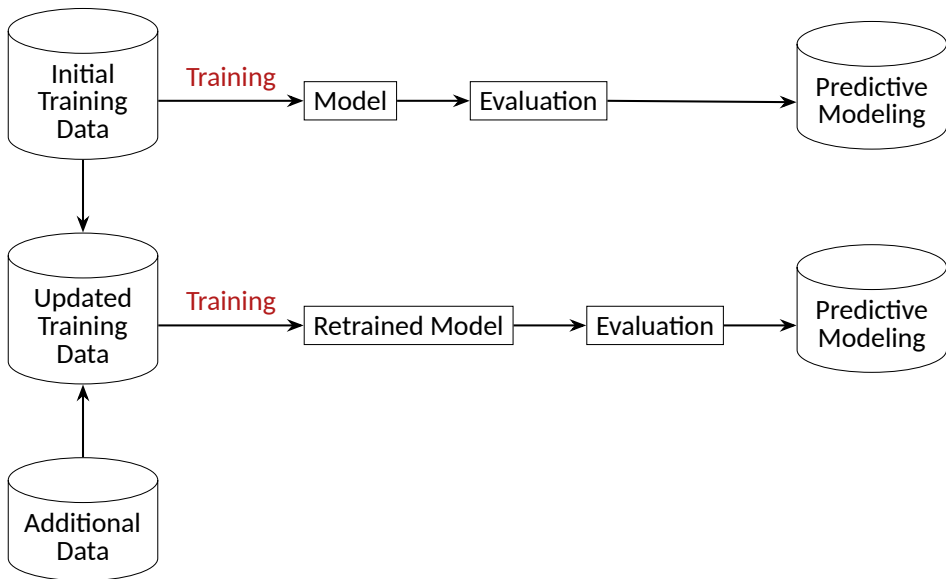


Figure 2.27: Schematic depiction of a batch learning workflow, which includes a trained model and additional data which is intended to be used as an addition to the existing training data. Typically, until the retrained model has been evaluated and validated, the predictions of the previous model are used. Cylinders depict the different reservoirs of data.

Results

3.1	Predicting Critical Temperatures for Materials Screening	64
3.1.1	Heusler Alloys	65
3.1.2	The Database	66
3.1.3	Data Wrangling	68
3.1.4	Machine Learning Modeling	75
3.1.5	Explainable Artificial Intelligence	91
3.2	Half-Metallicity in $L2_1$ and XA Heusler Alloys	98
3.2.1	Data	98
3.2.2	Modeling Goal	102
3.2.3	Screening Workflow	104
3.2.4	Model Predictions	106
3.3	Ultrathin Films of (001) oriented 3d Transition Metal Layers on fcc Noble Metal Substrates	121
3.3.1	Calculation & Film Setup	122
3.3.2	Workflow Outputs	126
3.3.3	DFT-Integrated Machine Learning	128
3.3.4	Explainable Artificial Intelligence	146
3.3.5	Data Analysis	149

In the following sections, three different applications of the previously discussed methods within the field of materials science are presented. These studies represent the results gathered during this thesis and are related to different physical phenomena and systems. Each section can be considered as a project representing different use cases of data analytics and ML in the field of computational materials science.

3.1 Predicting Critical Temperatures for Materials Screening

The critical temperature, introduced in section 2.3.1, is a complex quantity to predict as the quantity depends on material-specific properties, including e.g. structure, chemical composition, magnetic state, magnetic moments, and interatomic interactions. However, the critical temperature is relevant to industrial applications due to the fact that in the transition from a temperature above the critical temperature to below, a phase change from a magnetic (e.g. FM, AFM, spin-spiral, etc.) to a non-magnetic state occurs in the material. The critical temperature is closely related to the concept of magnetic stability, as the magnetic ordering is lost below this temperature. [211] Heusler alloys [212, 213] are known for multiple interesting properties including, but not limited to e.g. superconductivity [214], half-metallicity [215, 216], permanent magnets free of rare-earth metals [217], thermoelectricity [218], high-temperature magnets [14] and piezoelectricity [219]. Heusler alloys are known to have applications as magnetic shape memory [220] and tuneable topological insulators [221]. The combination of magnetic stability (*i.e.* reasonably high critical temperature) and half-metallicity in a single compound gives rise to applications in the field of spintronics for this compound, as this would allow spin-polarized charge currents, so-called spin-currents [222, 223], to occur and potentially be stable at operating conditions. This is inherently relevant for applications such as spintronics-based storage devices. [224–226] Hence, this combination is particularly interesting. This thesis's first results section discusses the key properties of magnetic Heusler alloys with high critical temperatures. The occurrence of half-metallicity in Heusler alloys is discussed in section 3.2.

In the material science community, efforts were made to predict the Curie temperature for more general material classes than Heusler alloys. [227, 228] While these studies use ML to predict a special case of the critical temperature, the Curie temperature, they are also very different from the results presented in the following. The aforementioned studies used a few thousand data points of experimental results while simultaneously restricting the field of interest to FM materials. However, typical challenges in materials science simulations include the sparsity of data, as acquiring data can be expensive. This also holds for the critical temperature, as described in section 2.3.1, due to the necessity to perform both the *ab initio* computation of a compound and the MC simulation. There has been an earlier study [12] which used a regression based on experimental results in conjunction with DFT to predict compounds with large critical temperatures. However, this work focuses on Heusler alloys consisting only of transition metals. Also, existing work on feature importance for critical temperature estimation is restricted to subgroups of magnetic Heusler alloys like e.g. to alloys containing only transition metals and rare-earth elements. [229, 230]

The results presented in the following should be seen as a small-scale example of how to use existing *ab initio* results in combination with ML methodologies using typical materials simulation data set sizes (A few hundred data points), which include many features, in order to:

- Approach predictive modeling tasks with high-throughput materials screening applications in mind.
- Reuse existing data purposefully beyond a simple analysis.
- Discussing the applicability of a materials screening approach on small data sets, with and without *ab initio*-originated results present.
- Explain model predictions and potential patterns discovered in the data using state-of-the-art XAI methods.

The results presented in this chapter are partly published in [231]. The data [232] processed from the original data set [14] and most of the code [233], which was written to obtain and visualize the results discussed in the following, are publicly available.

3.1.1 Heusler Alloys

Heusler alloys represent a subgroup of the extensive phase space in materials sciences. This subgroup exhibits a large amount of structural homogeneity. This homogeneity allows a complete structural description of an individual ordered alloy using very few quantities as the lattice constant, the component elements, and the symmetry group of the alloy. However, disordered Heusler compounds exist beyond that, which can occur due to e.g. present impurities, vacancies, and atomic displacements in the corresponding alloy structure. [234, 235] Compared to their ordered counterparts, disordered alloys can exhibit different material properties. [236] This concludes that disordered Heusler structures and properties that emerge due to the structural disorder are of interest to the scientific community. In fact, the occurrence of structural disorders in real-world compounds depends on multiple factors. Such include the growth conditions of the material and the difference between the free energy of the ordered and the disordered structures. The latter represents a competing effect of the free energies and is also influenced by the compound's elemental composition, which explains why there are configurations for which either the ordered or the disordered structures are energetically favored. Beyond that, temperature-dependent order-disorder phase transformations are known. [237] Temperatures for which a structural phase transition is observed indicate that the energetically lowest structures are different below and above this particular temperature.

Quantities
Required to
Describe a
Heusler Structure

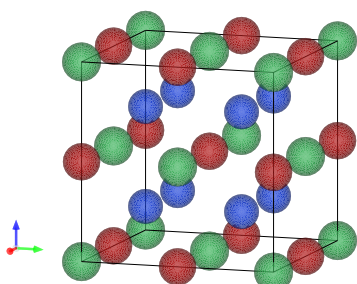
Disordered
Heuslers

Given some transition metals X and Y and an element Z from the main groups 13 to 15 from the periodic table, ordered Heusler alloys (sometimes referred to as full Heusler alloys or $L2_1$ phase) follow the molecular formula X_2YZ . [212, 213] For an inverse Heusler alloy (XA phase), the molecular formula is the same, but X is taken from the main groups 13 to 15 and Z and Y each denote a transition metal element. In the case of a structural half Heusler ($C1_b$ phase), the elements are chosen as in the $L2_1$ phase, but the molecular formula is given by XYZ. [238] Beyond that, modern solid state physics also recognized quaternary Heusler alloys given by the molecular formula $X'XYZ$, while the elements are chosen from the periodic table similarly as in the $L2_1$ phase case, but X' is also chosen from the transition metals. An overview of the discussed structure types is given in Figure 3.1.

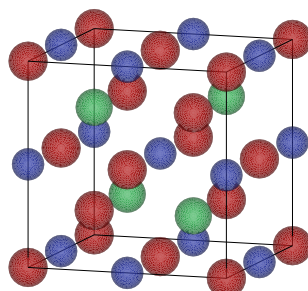
Structural
Families

There are different types of quaternary Heusler structures known—typically denoted as Y phase. These types can be characterized by the order in which the constituting elements are aligned on the conventional unit cell diagonal. [240, 241] Figure 3.1 also illustrates how the knowledge of the constituents, the lattice constant, and the structure type completely define the crystal structure. This structural homogeneity restricts the phase space of possible Heusler-like materials, which simplifies the feature space, compared to general 3-dimensional solid crystals, when it comes to modeling and predicting quantities. This allows for accurate predictions based on only a fraction of the required data, which would be necessary to achieve similar accuracies on the feature space of the entire materials science chemical and structural landscape. However, including disordered Heusler alloys already starts to lift the constraint imposed by the structural homogeneity due to the number of possible disorders and disordered structures. Often, disordered Heusler structures are also characterizable by phases.

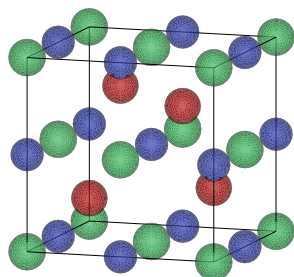
Structural
Homogeneity



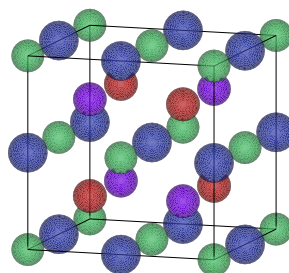
(a) Structure of a full ($L2_1$) Heusler alloy.
This structural depiction is based on Cu_2MnAl .



(b) Structure of an inverse (XA) Heusler alloy.
This structural depiction is based on Hg_2CuTi .



(c) Structure of a half ($C1_b$) Heusler alloy.
This structural depiction is based on MnNiSb .



(d) Structure of a quaternary Heusler alloy.
This structural depiction is based on CoFeTiGe .

Figure 3.1: Depiction of different Heusler alloy structures. The color code used for the atoms in the individual depictions is X'XYZ . The depictions were created using the VESTA software. [239] The structure data of the mentioned compounds was taken from the Materials Project. [94]

For example, the A2 phase is known, in which the X and Y or the X and Z sites are intermixed in comparison to the $L2_1$ phase. Another disordered phase is given by the B2 phase, where the sites Y and Z are occupied randomly by atoms that would be located in these sites given the ordered $L2_1$ phase. [242] Also, binary Heusler alloys are known, which consist of only two elements. For binary Heusler compounds, the molecular formula concerning the $L2_1$ sites is given by X_2XZ and referred to as D0_3 . [243]

3.1.2 The Database

JuHemd
KKR-GF-Based

The JuHemd [14] is a database of Heusler alloy's critical temperatures. It contains published experimental results and such originating from *ab initio* KKR-GF calculations combined with an MC-based simulation approach—as discussed in section 2.3.1—based on the crystal structures from the experimental publications. [14] Alongside the critical temperatures, structural information like the symmetry group, elemental composition, and lattice constant are stored in the database. Also, the theoretical magnetic moments of the individual atoms, which result from the first-principles calculations, are included. Counting ordered and disordered Heusler structures, half Heusler alloys, inverse Heusler compounds, and quaternary Heusler formations, the database contains 776 unique crystal structures.

776 Different
Structures

The counts of the structural phases' characterization for the compounds contained within the JuHemd is shown in Figure 3.2. The total contains 61 different symmetry group configurations of ordered, disordered, and fractionally disordered configurations included in the JuHemd. The fractionally disordered configurations represent 17 individual structures and symmetry groups.

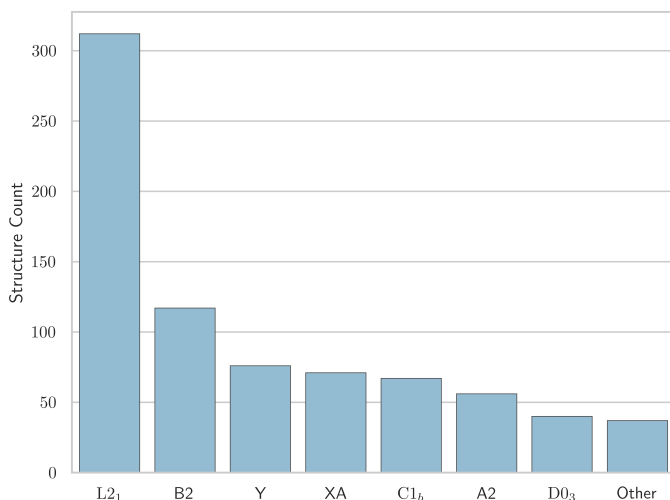


Figure 3.2: Overview of proportions of the structural phases contained in the JuHemd. The phases contained in the JuHemd include ordered structures ($L2_1$, XA, Y and $D0_3$) as well as disordered structures (mainly A2 and B2). It can be seen that the regular and ordered Heusler phase $L2_1$ is the most prevalent in the database. In cases of fractional disorders, which only represent a minority in the JuHemd, these phases are counted as “other” together with unlabeled symmetry groups and special minority cases like e.g. the noncentrosymmetric group $R\bar{3}m$ and the structure groups B1 (NaCl structure) and B32 (NaTi structure).

The data stored in the database is not exclusively given in numeric values, but the database also contains string and character data type entries for each compound, which describe and classify the structure of each alloy. Additional metadata like e.g. a label assigned to an alloy based on the chemical composition, where the published experimental results can be found, and when the results were published are stored within the JuHemd. To ensure that all potentially relevant features of a given structure are extracted from the data, some of the metadata needed to be processed as well to include additional features in the subsequent modeling and analysis steps. It is worth mentioning that the reported experimental structure results may vary for identical structures. This may be the case when the published lattice constant of a given alloy slightly deviates from an identical previously reported structure due to measurement inaccuracies. Hence also the computed critical temperature differs based on the deviations of the structural setup. Only the most recently reported structure and the corresponding computation were considered in such cases. As the theoretical critical temperature is computed in a layered process in which first the Heusler structure—based on the empirically reported structure—is calculated using a KKR-GF DFT calculation, which also computes the pairwise exchange coupling parameters for each alloy. The exchange parameter can then be used to calculate the critical temperature further using the mentioned MC approach. This two-step process is shown in Figure 2.8.

Heterogeneous
Data Types

Metadata

Two-Step
Process

Exchange
Parameters

Exchange-
Correlation
Functionals

Measurement
Compound
Purity

The *ab initio* results in JuHemd have been computed using both an LDA-based [244] XC functional as well as GGA-based [245] calculations. As a result there are two theoretical predictions for the critical temperature contained in the JuHemd for many compounds. As some systems are easier to compute with either of both functionals, computations can be successful for one functional but fail for the other. This explains why there are not the same number of theoretical results in the JuHemd for both functionals. As the quality of experimental critical temperature results is hard to determine and varies drastically from publication to publication—primarily for reasons of purity of the measured compound—the following analysis and discussion are based on the theoretical predictions for the critical-temperatures only.

3.1.3 Data Wrangling

The JuHemd contains many entries for each compound data point. Some entries are given as simple numeric values (e.g. lattice constant, *ab initio* computed total energy, and the critical temperature itself). Other entries are given as more complex data types which are encoding information, some of them such that different properties or quantities have been consolidated (e.g. symmetry group, system magnetic structure factors, and the system sites).

Constructed
Descriptors

The heterogeneity and complexity of the stored data made extensive preprocessing necessary prior to any further usage of the data. During this preprocessing step, additional material descriptors have been constructed using the available information. Since disordered Heusler alloys are included in the JuHemd, constructing certain features is more complex than if only ordered structures had been included. For these disordered alloy systems, the site-specific fraction of occupying elements in the compound is a quantity that influences many descriptors that could be constructed. For example, the total magnetic moment is computed by taking the moments of the individual atoms from the KKR-GF calculation and multiplying the moment with the fraction of the atom's occurrence on each site. This is simple for regularly ordered Heusler alloys and inverse Heusler structures. However, extra handling is required for half Heusler $C1_b$ phases, quaternary Heusler alloys, and also for disordered Heusler-like structures. This affects e.g. the total magnetic moments, the absolute magnetic moments, and the constructed atom densities. The aforementioned densities include the constructed density of FM atoms (Fe, Ni, and Cobalt) within the compounds. Using the fraction of an individual element in the compound as a descriptor allows us to figure out later which elements on the Heusler sites have an increasing or a decreasing impact on the model's prediction for each of the compound's critical temperatures. The concept of SHAP values, as introduced in section 2.5.3, is used to conduct this discussion. As an ML model bases the prediction on the relations learned by the data it was trained on, this allows us to understand both the prediction and discover the relations represented by the training data.

Prediction
Interpretation

Missing Data
Points

Furthermore, incomplete data points were removed during the data processing step. Unfortunately, the JuHemd contains multiple instances of missing data, e.g. if the initial first-principles calculation does not converge and hence does not yield magnetic properties, then only the structure and the collected experimental data might be contained in the JuHemd. [14] An approach to handle missing data would be to interpolate missing features during preprocessing. However, given the severity of features that were abundant in these cases (e.g. the fractions of atoms in the compound, magnetic moments), an interpolation was not possible with reasonable accuracy without additional information.

Furthermore, supplement atomistic descriptors from the literature and the corresponding compound totals—weighted by the atomic fraction of occupation—have been added during the preprocessing step. By including the totals, no assumptions are made beforehand if it is physically reasonable to sum e.g. the atomic numbers or not, but leave it to the feature importance, impact analysis and the used model to determine if a given feature has predictive value or not. An overview of features, including a short description for each feature, directly obtained from the JuHemd without extensive processing, constructed descriptors, and additionally included atomistic quantities with their corresponding totals is given in Table 3.1. The index i denotes a site in the set of inequivalent [246] sites contained in the alloy structure. The summation over the lattice sites i included in Table 3.1 denotes the summation over all inequivalent alloy sites. For the magnetic moments, the index l refers to the position of occurrence of the element the magnetic moment originates from in the molecular formula of the alloy. Hence, for e.g. the $L2_1$ and XA phases $l \in \{1, 2, 3\}$ but for phases, like e.g. the quaternary Y configuration, with four unique elements contributing to the structure $l \in \{1, 2, 3, 4\}$.

Feature
Importance &
Impact

By looking at Table 3.1, it is clear that considering all descriptors will introduce many features to the ML models. While some descriptors will likely be highly correlated with each other, the number of features used here already suggests an approach using an ML model that can either shrink the coefficients of less relevant features (e.g. such as LASSO) or determine the most impactful features with regard to the prediction (like e.g. tree-based models). It is commonly known as “one-in-ten-rule” [247] that—in order to avoid overfitting and the learning of spurious relations, i.e. such that are not causally related, within the data—there should not be more than one feature included for every 10 data points in the training set. While this rule is considered a best practice, it is not inherently true for all models, as e.g. the LASSO [144] is used to perform a selection of essential features [248] and hence, can handle more features than indicated by the “one-in-ten-rule”. However, this has to be kept in mind, as other measures capable of reducing the potential overfitting might be needed in this case. In Table 3.2, the features in the processed Heusler data set are grouped by their origin, either directly extracted from the JuHemd, constructed out of JuHemd information, or added by external libraries [249] based on the constituting atoms contained in the compound. Table 3.2 shows that the total number of descriptors adds up to 119, which are available to a model to predict the target quantity, the critical temperature.

One-in-Ten-Rule

At the beginning of this PhD project, only 162 compounds (state on 8th of December 2020) representing exclusively ordered structures were already computed at that time. The amount of data that was collected and ultimately published increased over time. With each iteration of the data set, the data processing, the ML modeling, and the interpretation of the model predictions using XAI were refined. Using the published version of the JuHemd [14], the original number of structures of 776 reduces to 387 post-processing for which the LDA XC [244] functional has been used in the *ab initio* computation process. 408 structures are extracted post-processing, computed using a GGA XC [245] functional. These 408 structures only include magnetic compounds from JuHemd with a critical temperature greater than zero.

Post-Processing
Data Amount

Label	Description
lattice_constant	Lattice constant of the Heusler
resval	T_c value in Kelvin
etotal (Ry)	Total energy of the compound E_{Tot}
formula	Chemical formula of the compound
Ferromagnetic Density	Fractional density of FM elements (Fe, Ni, Co) in the Compound
Rare earth Density	Fraction of rare earth components in the Compound
Symmetry Code	An integer encoding the compound's symmetry group by occurrence in the sorted array of all unique symmetry groups in the JuHemd
Individual Magnetic Moments	Individual magnetic moments m_l of all constituent atoms, ordered by their occurrence l in the compound's molecular formula
Absolute Magnetic Moments	Individual absolute magnetic moments $ m_l $ of all constituent atoms, ordered by their occurrence l in the compound's molecular formula
Total magnetic moment	$M = \sum_l m_l$
Sum of absolute magnetic moments	$M_{\text{Abs}} = \sum_l m_l $
Magnetic State	Four digit integer encoding the magnetic state (FM, ferrimagnetic, AFM, and spin-spiral)
Stoichiometry	Five digit integer encoding the stoichiometry of the compound
Density by Atomic Number	Fractional density of each atomic number is encoded by an individual descriptor
Atomic Number	Atomic number of the constituents Z_i
Number of Neutrons	Number of neutrons of the constituents
Nominal Mass	Nominal mass of the constituents atoms
Number of Electrons	Number of electrons of the constituents
Exact Mass	Exact mass of the constituents atoms
Atomic Radius	Atomic radii of the constituents atoms
Number of Valence Electrons	Number of valence electrons of the constituents atoms e^{val}
Covalence Radius	Covalence radius of the constituents atoms
Period	Period number in the PSE of the constituents atoms
Electronegativity	Electronegativity of the constituents atoms $\chi^{(i)}$
Van der Waals Radius	Van der Waals radius of the constituents atoms r_i^{vdw}
Electron Affinity	Electron affinity of the constituents atoms E_i^{ea}

Table 3.1: List of entries in the processed data set by their label, including a short description. This table is adapted from [231].

Directly extracted Features	Constructed Features	Atomic Features
lattice_constant [†]	Magnetic State [†]	Atomic Number [*]
resval [‖]	Symmetry Code [†]	Number of Neutrons [‡]
etotal (Ry) [†]	Density by Atomic Number ^{††}	Nominal Mass [‡]
formula [†]	Ferromagnetic Density [†]	Number of Electrons [‡]
	Rare earth Density [†]	Exact Mass [‡]
	Total magnetic moment [†]	Electron Affinity [‡]
	Stoichiometry [†]	Atomic Radius [‡]
	Individual Magnetic Moments [*]	Electronegativity [‡]
	Absolute Magnetic Moments [*]	Covalence Radius [‡]
	Sum of absolute magnetic moments [†]	Period [‡]
		Number of Valence Electrons [‡]
		Van der Waals Radius [‡]

[†] One descriptor per compound

[‡] Five descriptors per compound (Including the compound total)

^{††} 31 elemental densities per compound and a single density for unoccupied sites

^{*} Four descriptors per compound

[‖] target quantity T_c

Table 3.2: Grouping of the descriptor labels by the way they were obtained. For those acquired directly from the JuHemd (left column), the label from JuHemd has been used. Each descriptor is accompanied by the corresponding number of individual entries, per compound, which are added to the processed data set. Cumulated, the processed data set contains 119 individual features.

It was necessary to define a cutoff to distinguish which compounds are considered magnetic. This definition had to include a plurality of possible magnetic configurations, like e.g. FM and AFM. Hence, it was necessary to define this cutoff independent of the net magnetization within the unit cell, as the net magnetization of ideal AFM structures is vanishing. In this application, a compound is considered magnetic if it fulfills equation (3.1). Using the sum of the absolute moments over all atomic occupants, i ensures that neither AFM states are missed and compounds containing only a single magnetic site occupant are included. However, both criteria, the magnetic cutoff and the exclusion of compounds with a critical temperature of zero, can easily be changed in the data processing script [233] to ensure that these criteria can be adapted to different research needs in the future.

$$\sum_i |m_i| > 0.1 \mu_B \quad (3.1)$$

The ML-based analysis and model evaluation results vary slightly for the data sets based on the two different XC functionals regarding model performance. However, more training data for ML models is generally preferred, as more training data potentially increases the model performance and generalizability as a larger data amount is reducing the chances of overfitting to occur. [247, 250, 251] Therefore, the results that were computed using the GGA XC functional during the *ab initio* KKR-GF computation are discussed in the following.

The processed compound's data collection has been made publicly available [232] in the curated material's science database Materials Cloud. [252] The key differences of this publication compared to the JuHemd include the following:

- The data has been cleaned, in the sense that incomplete data points have been removed.
- Constructed descriptors have been included, which were deduced from JuHemd but not directly available from the original database.
- Every feature contains a numeric value for every compound, except for the molecular formula.
- The stored data is displayed as more human-readable in rows and columns.
- As the publication only contains theoretical predictions of the critical temperature, the meta-data from the empirical publications was not included.

However, the described data cleaning process also creates some artifacts. The construction of descriptors is performed before incomplete data points are removed. This is due to the fact that in order to construct the descriptors for each alloy, the processing script iterates sequentially over all compounds in the JuHemd, as the individual structures are also stored sequentially in the database. Therefore, all features are collected for a particular compound before going to the following compound. If a descriptor is unavailable for a structure, the data point will temporarily be stored as incomplete. All incompletely stored structures are removed at the end of the processing after collecting all available descriptors for each compound. Hence, some elements contained in the original JuHemd have no occurrence in the cleaned data. Therefore, elemental fractional densities are present for these atomic numbers included in the processed data, which have a sample variance of zero through the entire data set—which at the current version of the JuHemd affects 11 atomic densities. If the JuHemd is extended in the future, these densities could turn into a meaningful descriptor and should be included in the processed data set.

As the data processing script is published [233], it would be easy to execute the corresponding script for an updated JuHemd—assuming JuHemd maintains the current structure and retains the existing keywords. Also, a structure ruled out due to incomplete data but completed in the JuHemd in a future version would be included automatically.

Before the ML training process began, as the last step of processing the JuHemd data before compiling the processed results, the zero variance features were removed as they are meaningless to the model—as well as the underlying physics—and only would cost additional computing power in the training process without adding any value to the model performance.

After the previously discussed processing steps, the distribution of critical temperatures matching the compounds which have a full descriptor set is shown in Figure 3.3.

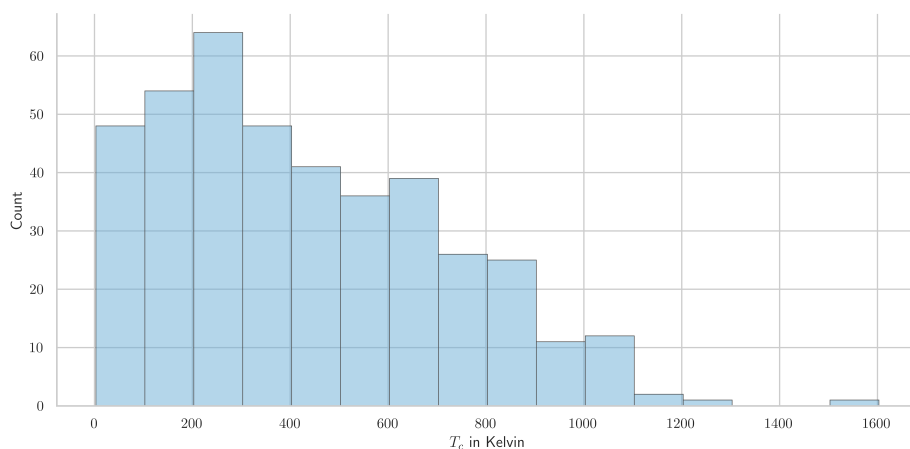


Figure 3.3: Post-processing distribution of critical temperatures in the GGA data extracted from the JuHemd. This depiction of the critical temperature distribution has been generated using the GGA data set. However, the LDA distribution looks very similar, as can be seen in appendix A. This figure is adapted from [231].

From Figure 3.3, it can be seen that a large proportion of the critical temperatures contained in the processed data is located below 400 K, which means that a significant amount of compounds is located below the temperature range in which a real-world application would be considered as briefly discussed in section 2.1.1. For increasing temperatures, the trend shown in the histogram in Figure 3.3 is that higher critical temperatures are less likely when a random compound from the data set is chosen. Ultimately, when crossing the 1100 K mark, only very few critical temperatures lie above that. In fact, there is a single compound with a critical temperature located above 1500 K in the processed data set. A gap larger than 200 K exists between the compounds with the highest critical temperature. It is easy to see that it is unlikely that a model can learn to predict this particular data point due to the fact that the data point's critical temperature is located far from the critical temperatures of the remaining data. If randomly assigned to a test set, the highest critical temperature would represent a prediction that is entirely OOS.

Outlier Removal

Due to these arguments, the data point corresponding to the critical temperature located above 1500 K is removed as an outlier prior to the training process. However, this data point is located in the data publication [232] as it might be relevant for analysis methodologies and applications that differ from the prediction of critical temperatures using ML.

The distribution of atoms that constitute the compounds contained in the processed data per site is shown in Figure 3.4.

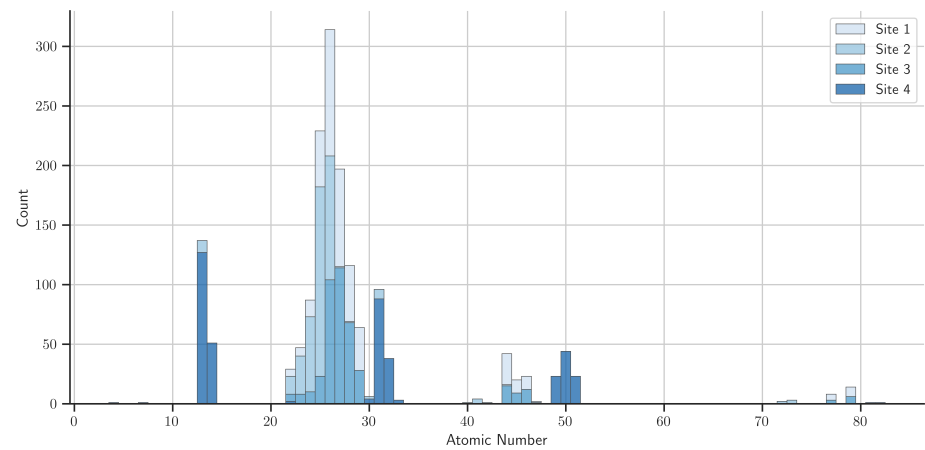


Figure 3.4: Distribution of atomic numbers in the compounds extracted from the JuHemd database, with color-coded lattice site positions, post-processing and after outlier removal. This depiction of the atomic number distribution has been generated using the GGA data set. However, the LDA distribution looks very similar, as can be seen in appendix A.

From Fig 3.4 it is obvious that a large proportion of the compounds contained in the database contains 3d transition metal elements. The fact that 3d transition metals have a high prevalence is not random. As the critical temperature is an inherently magnetic property, the elemental selection of the JuHemd is biased towards magnetic elements. Hence, it is expected that the 3d transition metal group, which contains the classical FM elements iron, cobalt, and nickel as constituents, contribute to a significant proportion of the observed magnetic properties in Heusler-like compounds. It is also known that manganese plays an important role in the critical temperature for ordered and disordered Heusler alloy, which is the reason why in Figure 3.4 manganese is a constituent of many compounds in the data set. [253–255]

After the data has been processed, the next step is to use the acquired processed data to model the critical temperatures for high-throughput applications using lightweight ML algorithms, as discussed in the following.

Strong Transition
Metal
Representation

3.1.4 Machine Learning Modeling

An ML modeling process requires multiple steps in the context of a research study. These steps typically include:

- Defining a modeling goal and choosing an appropriate metric according to the selected goal
- Preparing the data to be compatible with the modeling goal and potential models
- Select models and subsequently select hyperparameters
- Evaluation of the model predictions on data which has neither been used for training nor hyperparameter selection

In this section the previously discussed steps of the ML modeling process are discussed with respect to the data acquired from the JuHemd. Potential additional steps include e.g. retraining the selected model with the chosen hyperparameters on the entirety of the available data post evaluation, performing predictive modeling, discussion of remarkable model predictions (Correct or incorrect), and analysis of the properties learned by the model by using XAI.

ML Prerequisites

There are a few ML-specific prerequisites necessary before the actual modeling is started. As a first step, the order of compounds is randomized to avoid any clustering of compounds in the training, validation, or test set. A random seed has been set at the beginning of each program file for every step that includes data randomization in this thesis. [233] This allows for the reproducibility of results without waiving the advantages of randomization in data randomization. In this section, the models that involve randomization, inherently in their architecture, are given a fixed seed. This section of the results gathered within this thesis is the sole section that strongly involves models that rely heavily on randomization, such as the random forest and the extra trees models. Hence, specifying a randomization seed for the models was not necessary within the other result sections.

Data
Randomization

As mentioned, the data will be split into training, validation, and test data sets. The individual data sets' ratios generally depend on the overall data amount of the given modeling task. Generally, larger data sets are fine with smaller ratios for validation and test data, as even the small ratio still includes several data points, allowing for a reasonable average score to be computed based on its size. [256] However, the opposite is true for small data sets. [158] It has established that a good choice for small data sets like the one examined in this study is to use 60 % of the data for training and 20 % each for validation and testing. This is easily achievable with the CV methods by holding back 20 % of the data for testing and choosing to perform a 4-fold CV on the remaining data to find suitable hyperparameters. After the splitting has been completed, the features of all data sets are scaled according to equation (3.2) using the mean μ_j^{Train} and standard deviation σ_j^{Train} of the training set for all individual features j and data points i . While it is not strictly necessary to scale the data for all ML models, it is known for some models to improve their predictive capabilities. [257] Hence, the data which the models are trained on are given by the matrix constructed by z_{ij} using the unscaled features x_{ij} .

Data Splitting

Feature Scaling

$$z_{ij} = \frac{x_{ij} - \mu_j^{\text{Train}}}{\sigma_j^{\text{Train}}} \quad (3.2)$$

After outlier removal and splitting of the data, the distribution of the critical temperature for both the complete and the test data set is shown in Figure 3.5. One can see that, given the sizes of the test data set, the range of the critical temperature in the complete data set is reasonably sampled by the chosen test set. At this point, the data itself is prepared for the ML training process.

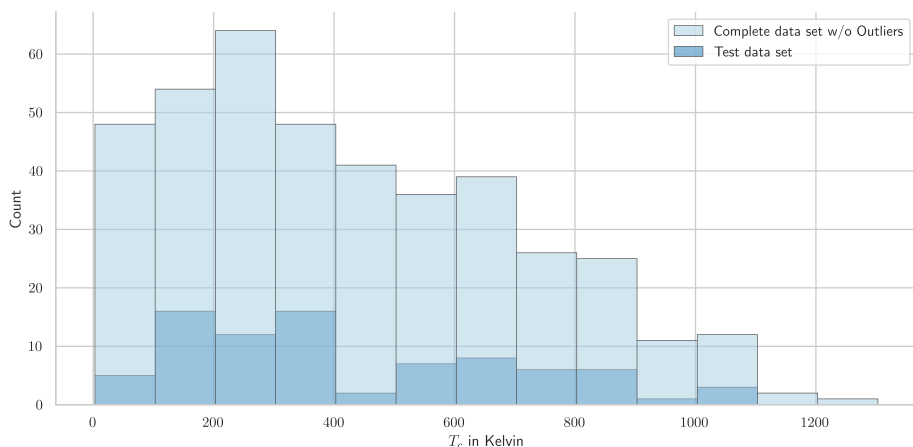


Figure 3.5: Post-processing distribution of critical temperature in the GGA data after outlier removal, including the distribution of the test set. This figure is adapted from [231].

Defining Modeling Goals

Capabilities of
ML

Regression and
Classification

The task of choosing a modeling goal boils down to the intended application of the trained model. In this particular case, this study aimed to gauge the capabilities of ML algorithms to predict a complex magnetic quantity like the critical temperature on typical materials science data set sizes. However, with a materials screening application in mind, in principle, a classification would be enough if the critical temperature lies in the range where a technical application is reasonably possible. However, a regression study is necessary for materials design tasks, for which a specific value range of a certain quantity is required. Knowing this, it is reasonable to explore both approaches on this data set and compare the regression and classification performances by assigning a class to the regression models predictions on the test set. These classifications, derived from regression models, are referred to as indirect classification models within this thesis. While this is not commonly done, it can be seen as an additional sanity check for both modeling applications to see if both approaches yield similar predictive capabilities. This is especially interesting as determining the theoretical critical temperature is given by a layered process, as seen in Figure 2.8. This leads to the intuitive question of whether both layers (DFT and MC) can be replaced by an ML approach, which also includes the question of whether the data available only from atomic and structural descriptors (As the DFT-originated descriptors such as *e.g.* magnetic moments, magnetic states, and total energy are not available if the DFT step is not performed this reduces the set of descriptors to a total of 107 features compared to all the features listed in Table 3.2) is sufficient to predict a magnetic property such as the critical temperature using ML models.

However, besides replacing both steps, it would be imaginable only to replace the MC step with ML-based methods and include the DFT-originated features. Also, the latter approach would be favorable as the MC step consumes the same order of magnitude of computing time [253] as the *ab initio* calculation. The different levels of invoking ML models into the modeling process of the critical temperature are shown in Figure 3.6.

With and
Without DFT
Results

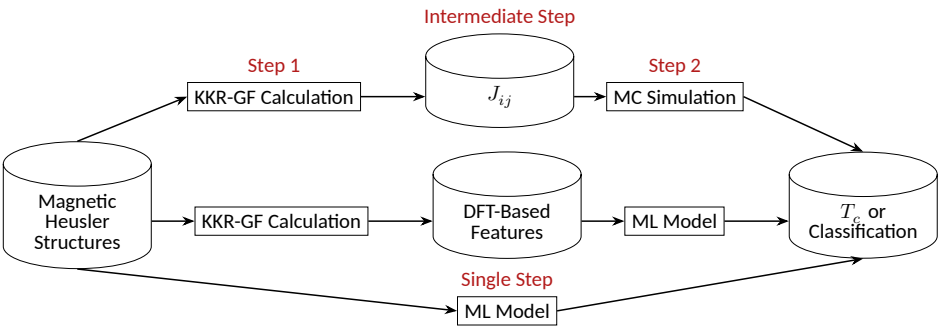


Figure 3.6: Schematic depiction of the layered T_c determination with different ML integration levels with increasing incorporation of ML models and modeling complexity from top to bottom. Cylinders depict where different data is collected during the multi-stage process. This figure is adapted from [231].

Considering both the layered process required to determine the critical temperature and the fact that both classification and regression approaches are examined, this leads to 4 cases to be examined and evaluated for their practical applicability. The examined combinations are shown in Figure 3.7.

ML Modeling Integration		
	MC → ML	DFT + MC → ML
Regression	Regression on Full Feature Set	Regression on Reduced Feature Set
Classification	Classification on Full Feature Set	Classification on Reduced Feature Set

Figure 3.7: Depiction of combinations of modeling tasks to evaluate, which arise in this study from the combination of either modeling a classification or a regression task and replacing either the MC step with an ML model or both the *ab initio* and the MC step with a single ML model.

However, still the choice has to be made, by which metric a model and the corresponding hyperparameters should be chosen as well as how predictions on the test set should be evaluated. It is not the goal of this study to optimize both hyperparameters and models to a great extent but rather to explore the capabilities of ML models on a rather typical materials science data set and a common materials screening task. There is only a small selection of hyperparameters included per model, which allows us to restrict the selection process of the hyperparameters to a basic grid search.

As already discussed in section 2.5.1, different metrics for classification and regression are available. Given the discussed modeling goal for the regression task, the metric used in this study to select the best estimator and hyperparameters using the 4-fold CV procedure is the coefficient of determination (R^2) from equation (2.54). The coefficient of determination has been chosen as employed metric as R^2 reflects how well the features reflect the change of the target quantity with the given model. Hence, technically, the R^2 metric should obtain the most meaningful model, in the physical sense, even though the particular model obtained does not necessarily have a lower prediction error than another model. However, the dimensionless R^2 value is accompanied by the MAE to determine how large a typical error for model predictions on unseen data would be. For the model training and optimization, the MSE was used due to its convexity and differentiability, as discussed in section 2.5.1 and 2.5.2.

The classes should be defined before choosing the appropriate metric for the classification evaluation and model selection. As already discussed in section 2.1.1 and also mentioned in the literature [258], an experimentally measured critical temperature of 400 K includes a decent buffer zone compared to room temperature for technical applications. Hence, also considering potential deviations by the *ab initio* + MC approach and additional errors introduced by the ML modeling on top of the theoretical T_c data, which might be present compared to the experimental critical temperatures, the threshold for critical temperatures to be labeled as “High T_c ” was chosen to be above 473.15 K which corresponds to 200 K above 0° C and hence is located around 180 K above room temperature which constitutes a buffer zone. A compound with a critical temperature below the previously introduced threshold is classified as “Low T_c ”. Of course, this choice of threshold represents a trade-off between including as many compounds that might be suitable for application as possible and excluding those, for which the model would overestimate the modeled critical temperature, while the actual experimental critical temperature does not allow an application within operating temperature conditions.

Having the classes for the classification task defined allows us to make a well-founded choice of metric. A few things have to be considered for this choice as discussed in section 2.5.1 and Table 2.4, these include:

- From Figure 3.5 it can be seen that the classes, given the discussed threshold, are slightly unbalanced, which means that the data considered is unequally distributed.
- Precision is essential, as a low precision would include false positives, leading to materials being flagged as “High T_c ” while they are “Low T_c ” compounds. As the “High T_c ” classified compounds in a materials screening application would be subject to further examination, this would constitute a waste of resources on the false positive classifications.
- Recall is also essential, as a low recall would include the occurrence of false negatives in the predictions. This translates to a significant amount of compounds with a potential application due to their “High T_c ” being missed in a materials screening application.

Hence, the F1 score was chosen, as the F1 score represents the harmonic mean of precision and recall; this metric is well suited to meet both discussed requirements. Beyond that, the F1 score is known to work well with unbalanced data. Alongside the test F1 score, the accuracy, precision, and recall include additional dimensions to interpret the model's performances.

F1 score

Regression

To select a model that could be used to predict the value of the critical temperature in a high-throughput materials screening setting, a plurality of models should be evaluated—following the mentioned shotgun approach—in order to determine an appropriate model type which is matching the data complexity, for the regression task, on this particular data set. An overview of the models evaluated on both the complete and the reduced data set, which does not contain ab initio-originated descriptors and hence skips both modeling steps for the critical temperature, as depicted in Figure 3.6, is given by the models listed in Table 3.3.

Linear	Non-linear	Ensemble
LASSO	K-Nearest Neighbors	Random Forest Regression
LASSOLars	Decision Tree Regression	Extra Trees Regression
Linear Regression		Gradient Boosting Regression
		XGBoost Regression

Table 3.3: Overview of the models evaluated for the regression task using the training data on the processed JuHemd data set. The models are grouped by their categorization as either linear, non-linear, or ensemble predictors. As the K-Nearest Neighbors model and the LASSOLars model, which combines the discussed LASSO model with the least angle regression (Lars), were both not discussed in section 2.5.2, details on the model architectures and the underlying theory can be found in [259–262] and [42, 144] respectively.

For all models listed in Table 3.3, except for XGBoost [125], the open source scikit-learn [114] implementation has been used. This also holds for the models used in the later discussed classification task as seen e.g. Table 3.6.

Using the 4-fold CV procedure and evaluating on 20% of the randomly selected test set—as previously discussed—the CV and metrics results obtained from the different models including the DFT-based features are shown in Table 3.4.

From Table 3.4 it can be deduced that the ensemble regression models outperform both the linear and the non-linear models. Naturally, ensemble models also represent a group of non-linear estimators; however, as ensemble models compile their overall predictions out of a plurality of model predictions, they are often considered a distinct subclass of non-linear predictors. [263] The insight that ensemble models are well suited for this task is not surprising, as ensemble models are known to handle tabular data well. [124, 127] However, while it is not surprising, it could not have been assumed beforehand that ensemble models are capable of predicting a complex magnetic property like the critical temperature on such a small data set and with the large structural phase space of ordered and disordered Heusler type alloys with a reasonable error, using the discussed features.

Tree Model's
Performance on
Tabular Data

	CV-Score	Train R^2	Test R^2	Train MAE [K]	Test MAE [K]
LASSO	0.66	0.78	0.66	101.78	121.61
LASSOLars	0.67	0.78	0.66	101.65	121.53
Linear Regression	<< 0	0.83	<< 0	88.4	>> 1000

K-Nearest Neighbors	0.49	0.65	0.57	125.69	131.6
Decision Tree	0.55	1	0.64	0	123.35

Random Forest	0.73	0.97	0.81	34.86	93.35
Extra Trees	0.77	1	0.85	0.0	81.87
Gradient Boosted Trees	0.77	1	0.84	0.0	82.14
XGBoost	0.69	1	0.79	0.2	93.43

Table 3.4: Regression CV and test evaluation results on the complete descriptor set, including descriptors acquired by the *ab initio* computation such as magnetic moments, magnetic state, and the total energy of the system determined using the KKR code JuKKR [62]. The models are grouped similarly to the order in Table 3.3 into linear, non-linear, and ensemble predictors. All values are rounded to the second digit after the decimal to avoid the impression that they are meaningful, as the scores over the validation and test sets represent an average over less than a hundred compounds. This digit convention is also used in the Table 3.5, 3.7, and 3.8.

As discussed in section 2.3.1, the DFT + MC-based calculation procedure error compared to experimental values ranges typically around 10 % to 15 %. Considering the lowest MAE on the test set (≈ 84 K) of the models in Table 3.4, it is clear that the ML-originated error will have a larger deviation to experimental values for most of the compounds as this additional modeling layer adds another source of errors. However, this error can be further decreased with more training data available. Using the first iteration of the JuHemd, available for training, only including 162 ordered compounds, the best achieved R^2 in CV at that time was about 0.25. This demonstrates the power of ML models to scale their predictive power, modeling capabilities, and accuracy with additionally acquired data and also explains the improved (For all models except the linear models) test score compared to the CV-score, as in the CV procedure—for each CV fold—only 60 % of the available data was used for training but, to determine the test score 80 % of the data was used for training the model which was chosen after a coarse hyperparameter optimization. The fact that the linear models did not profit from the additional training data could be attributed to their limited learning capabilities and the non-linear relations determining the critical temperature. A discussion of the non-linearity of this particular modeling task is included in section 3.1.5. While the improved R^2 could also be a result of a particularly easy-to-predict test set—compared to the individual validation sets in the CV process—this is unlikely as the data has been randomized before splitting it additionally, other random splits have been observed, during this study, with comparable performances. However, interpreting the R^2 -score on the validation and test set, it is remarkable that 77 % or respectively 85 % of the change of the critical temperature's variance can be linked to the included descriptors. This concludes that at least 15 % of the critical temperatures' variance is not sufficiently described using the available features. In this context, it is important to mention that this result is dependent on the overall data quality. Therefore, parts of the critical temperatures' unexplained variance could be caused by insufficient convergence parameters in either the DFT or the MC calculation step.

Table 3.4 displays a significant amount of overfitting—low bias and high variance—for all models except the K-Nearest Neighbors model. This can be intuitively understood, as the K-Nearest Neighbors model determines a set of closest points within the training data to a point the model should predict. Hence, the typical prediction of the K-Nearest Neighbors model represents an average of “close” data points from the training set. A consequence is that the model’s prediction accuracy is highly dependent on the density of data points within the training set regarding the prediction region. This, on the one hand, limits the model’s learning capabilities, as the model links data points by their similarity rather than relating individual features to the target quantity. Still, on the other hand, this particular model architecture reduces the likelihood of overfitting as the averaging over multiple “close” data points prevents the learning of *e.g.* spurious correlations within the training data. The training R^2 is increased—by model construction—compared to the test R^2 as the data points, which shall be predicted to determine the training R^2 -score, is, of course, included in the training set and hence contributes the correct T_c value to the average.

The regularized linear models exhibit a significantly decreased amount of overfitting compared to the tree-based models. Given the number of features in this data set, it is important to apply feature selection or regularization to extract only the meaningful descriptors rather than letting every feature contribute to the prediction. This lack of feature selection and regularization explains the simple linear regression’s poor CV and test performance. Every descriptor is fitted within this model, regardless of the individual feature’s predictive power. Hence, the model’s performance on unseen data is poor. In direct contrast, the LASSO model, which shrinks the modeling coefficient, depending on the regularization parameter, of inferior (Compared to other features within the data set) features to 0, shows far better generalization capabilities. Both regularized linear models exhibit some degree of overfitting. Still, due to the limited model complexity, the overfitting—but also the CV and test performance—is decreased compared to the tree-based models. The overfitting of the tree-based models could be decreased by an extended hyperparameter search, which would include reducing the tree sizes (pruning) or increasing regularization parameters in the case of the ensemble models. Generally, increasing the amount of training data would also assist in reducing the degree of overfitting.

Summing up, ensemble models perform best in predicting the critical temperature value. The corresponding prediction errors are reasonable on unseen data. It can be concluded that—given the reduced computational time consumption compared to the computationally intensive DFT + MC approach—ensemble models such as the extra trees regression model and the gradient boosted trees regression model could indeed be used—especially with more training data—to predict the value of a complex quantity such as *e.g.* the magnetic critical temperature. It is worth mentioning that both steps, the DFT and the MC calculation, can take several hours, running highly parallelized on one or more supercomputer nodes, each, while the ML training, hyperparameter optimization, evaluation, and prediction on this particular data set can be done on a laptop within a few hours.

In Figure 3.8, the test predictions and relative residuals of the extra trees model are shown, compared to the DFT + MC-based theoretical value of the critical temperatures. While metrics such as the coefficient of determination and the MAE quantify a model’s performance, it is crucial to examine the predictions of a model carefully and hence gain an understanding of potential systematic deviations. An example of such deviations in a model’s prediction can be seen when comparing the prediction of the extra trees model to the LASSO model’s predictions, shown in Figure 3.9.

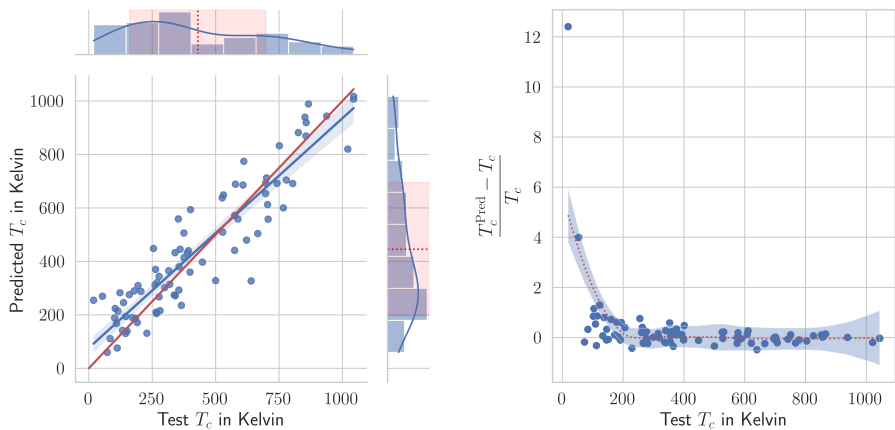


Figure 3.8: On the left: Depiction of the extra trees regression model predictions, compared to the DFT + MC determined critical temperatures including the distributions for both axes quantities (blue marginal histograms and lines, while the lines represent the smoothed distributions), the mean value (red dashed line in marginal distributions), and the standard deviation around the mean (red colored area in the marginal distributions). In this comparison, the red line marks the “ideal” prediction, where the predicted value matches the test label. The blue line represents a linear regression fitted to the blue data points, representing the comparison between model prediction and test label. The blue envelope around the linear regression denotes a 95 % confidence interval, which was determined using an approach based on bootstrapping. [264] On the right: Depiction of the relative residuals of the extra trees regression model’s predictions over the range of all test labels. The dotted red line corresponds to a LOESS [265, 266], and the corresponding blue envelope represents a point-wise computed 95 % confidence interval. These depictions are similar to [231].

From Figure 3.8, it can be seen that there are two considerable outliers existing on the lower end of the T_c range. While the deviation would not be substantial in absolute values, the outlier is significant in this relative depiction. Furthermore, the slope of the blue line fitted to compare predicted labels to the true critical temperature labels is slightly lower than the ideal prediction line in red. Overall, the extra trees regression model appears to model the critical temperature appropriately. However, the model has a slight tendency to overestimate small values of the critical temperature—which is supported by the fact that the slope of the line fitted to the predictions is smaller than the slope of the ideal red line and the majority of relative residuals as well the locally estimated scatter plot smoothing (LOESS) fit on the right of Figure 3.8 are positive for smaller test T_c values. This overestimation is also what caused the mentioned outliers. However, starting from approximately 200 K, the LOESS fit appears to remain close to the value of zero. This indicates that large asymmetric errors—which would appear in the smoothed relative residuals—in either direction seem to be absent for this model. While this is true for the extra trees regression model, this does not hold for less complex models, such as the LASSO model, as seen in Figure 3.9.

Symmetric Errors
Above 200 K

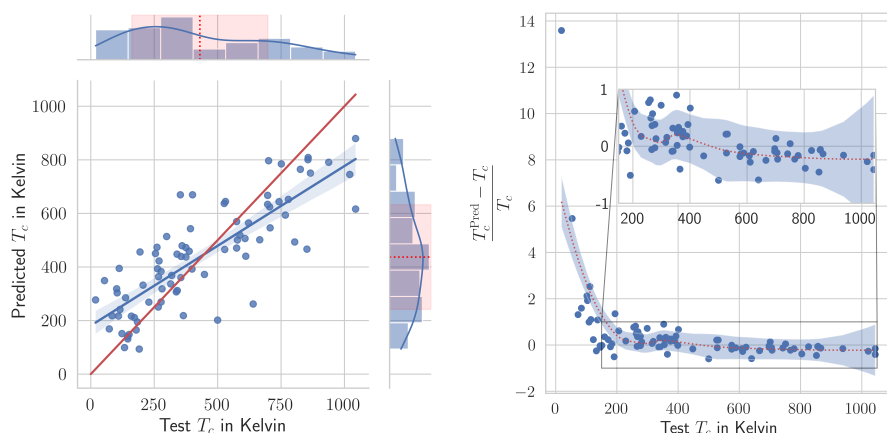


Figure 3.9: On the left: Depiction of the LASSO model predictions, compared to the DFT + MC approach, analogous to the left of Figure 3.8. On the right: Depiction of the relative residuals of the LASSO model's predictions over the range of the entirety of test labels, including the depiction of a LOESS fit, analogous to the right of Figure 3.8. The left depiction is similar to [231].

For the LASSO model, it can be seen from the slope of the blue fitted line in Figure 3.9 that the model is not able to model the nuances of the critical temperature comparably to the extra trees regression model. Also, the distribution of the predicted values is much more narrow and, hence, more centered around the mean value. Therefore, extremely high and low critical temperature values are predicted less accurately. From the relative residuals plot on the right of Figure 3.9, it can be seen—in comparison to Figure 3.8—that the outliers at the low end of the T_c range exhibit an even more significant deviation from the actual labels. Also, the LOESS fit of the LASSO residuals approaches the value of zero slower, exceeds it afterward again, and then continues to decrease below zero. The sign change of the LOESS fit is located around the mean of the predictions, which is clear evidence that a majority of values below the average prediction are overestimated, and a majority of the values above the mean prediction are underestimated. This is another hint, besides the metrics, that the LASSO model fails to model the critical temperature appropriately.

LASSO
Predictions for
High & Low T_c
Values

While both depictions of Figure 3.8 and Figure 3.9 are excellent for determining systematic issues within the individual model's predictions for different segments of the critical temperature range, a more macroscopic impression of the predictive performance is gained by examining the residuals (or relative residuals) distribution. The kernel density estimation [267, 268] of the relative residual distribution for both models is shown in Figure 3.10. The kernel density estimation is used to smoothen the distribution, as the low number of data points within the test set is insufficient to shape a histogram similar to the probability density function of the residuals. Using a smoothing Gaussian kernel, an estimation of the probability density function can be acquired using the kernel density estimation.

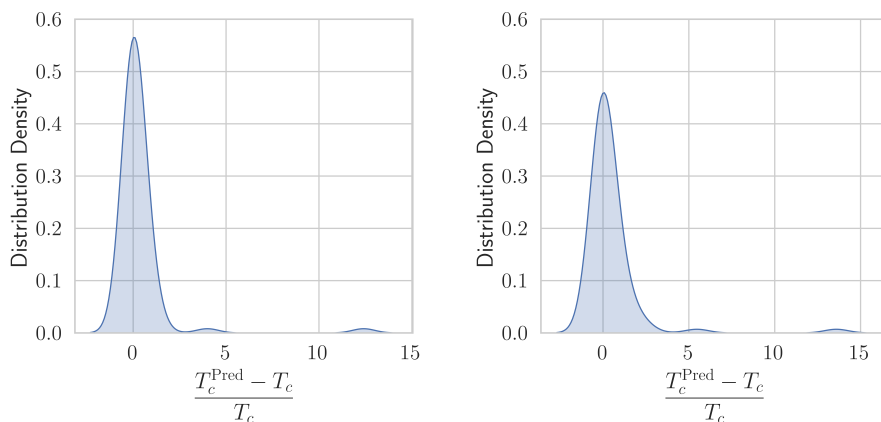


Figure 3.10: Kernel density estimations of the relative residuals of the extra trees regression (left) and the LASSO (right) model.

From Figure 3.10 the previously discussed shortcomings of the LASSO model at the task of predicting the critical temperature on this data set, compared to a more complex ensemble model, become even more apparent as the distribution's main peak of the LASSO residuals is broader and more asymmetric towards an overall overestimation of the predictions. Also, the outliers, visible as bumps beside the main peak, are located further from the main peak, as indicated by the previous residual plot. For the extra trees regression model, the distribution of the relative residuals is nearly symmetric at the main peak. A completely symmetric residual distribution would indicate the absence of systematic deviations of the predicting model.

Beyond the models trained on the entire set of descriptors, the same models were trained on the reduced descriptor set, which excluded all *ab initio* obtained features. The results of the training and evaluation are listed in Table 3.5.

	CV-Score	Train R^2	Test R^2	Train MAE [K]	Test MAE [K]
LASSO	0.31	0.58	0.63	142.29	129.92
LASSOLars	0.31	0.54	0.59	150.17	129.92
Linear Regression	<< 0	0.62	<< 0	132.23	>> 1000
K-Nearest Neighbors	0.35	0.5	0.53	157.09	144.51
Decision Tree	0.3	1	0.5	0	127.62
Random Forest	0.56	0.95	0.75	45.3	95.33
Extra Trees	0.53	1	0.74	0.0	100.02
Gradient Boosted Trees	0.57	0.94	0.72	52.07	94.52
XGBoost	0.55	0.93	0.69	57.78	100.11

Table 3.5: Regression CV and test evaluation results on the reduced descriptor set, excluding descriptors acquired by the *ab initio* computation as magnetic moments, magnetic state, and the total energy of the system. The models are grouped similarly to the order in Table 3.3 into linear, non-linear, and ensemble predictors.

From Table 3.5, it is apparent that, again, ensemble models are performing best even on the reduced set of features. However, from the CV-scores, it can be seen that the excluded features, compared to the evaluation shown in Table 3.4, were essential to the modeling task, as the CV-scores dropped significantly. This also translates to the scores for the test set and the corresponding errors. This unanimous drop in the model's performance results from the fact that the magnetic quantities, which are closest related to the magnetic critical temperature, were taken out of the feature set. Hence, the models training on this reduced data set had the same modeling task but less information about the physical system. The models have to rely on structural and atomic properties with magnetic information excluded. This increased modeling complexity transfers to the evaluated model performances. The lowest MAE from Table 3.5 is about 15.5 % higher than compared to the lowest MAE from Table 3.4. An additional observation that can be derived from Table 3.5 is that the training scores of the majority of the ensemble models have decreased. Combined with the overall decrease in CV and test scores, it is safe to say that the modeling task indeed became unanimously harder for all models, even on the training set. This underlines the importance of the magnetic properties to the critical temperature modeling. Again, the increase in scores comparing CV and test scores can be explained by the fact that the test scores are evaluated on models trained using a more extensive training database.

Increased
Modeling
Complexity

As in the previous, a detailed look is taken at the model performing particularly well given the metrics from Table 3.5. In this case, this is the gradient boosting regression model. In Figure 3.11 the predictions, relative residuals, and the kernel density estimation of the probability distribution density of the relative residuals of the gradient boosting regression model, trained on the reduced feature set, for the test set are shown.

Gradient
Boosting
Regression
Model

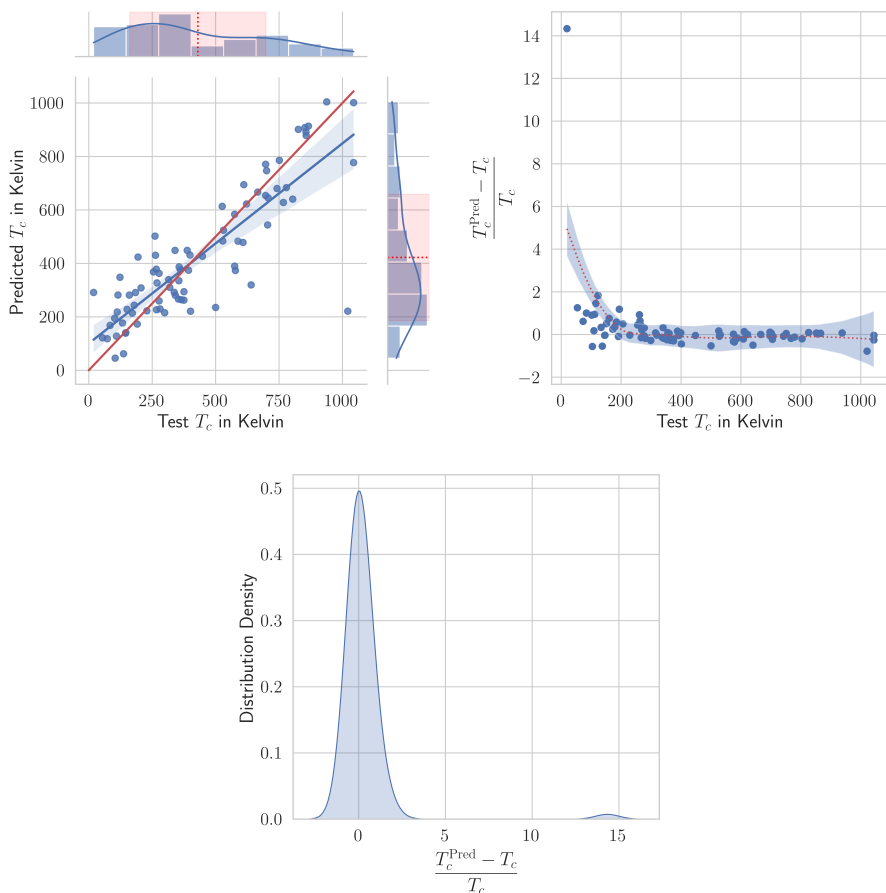


Figure 3.11: On the upper left: Depiction of the gradient boosting regression model's predictions, compared to the DFT + MC approach, analogous to the left of Figure 3.8. The model was trained under the exclusion of the DFT-originated features. On the upper right: Depiction of the relative residuals of the gradient boosting regression model's predictions over the range of the entirety of test labels, including the depiction of a LOESS fit, analogous to the right of Figure 3.8. On the lower: Distribution of relative residuals of the gradient boosting regression model's predictions, which was smoothened using a Gaussian kernel density estimation.

Various conclusions can be drawn from the depictions in Fig 3.11. First, comparing the prediction to the *ab initio* + MC-originated critical temperatures on the top left of Figure 3.11 shows that large outliers are present at both ends of the critical temperature range. This translates to the relative residual depiction, where a significant outlier in the small temperature region and one outlier at the high end of the critical temperature in the residual plot can be seen. The outlier on the higher end has an absolute residual value of about 800 K. Also, comparing the dashed red average line on the marginal of the comparison depiction, it can be seen that the prediction's average is lower than the test range's average. Hence, a systematic underestimation is present, which translates to the relative residual depiction. From the kernel density estimation of the probability distribution function of the relative residuals, it can be seen that the main peak is nearly symmetric with a single significant outlier, representing the outlier from the lower end of the temperature range. The slope of the blue line fitted to the prediction comparison to the test set is lower than e.g. the corresponding fitted line from the extra trees regression model, which still had DFT-originated features present. However, as also indicated by the asymmetric envelope, the certainty of the linear regression is inferior to the corresponding line fit to the extra trees regression predictions in Figure 3.8. The significant outlier on the high end of the critical temperature range also causes this uncertainty. Furthermore, the LOESS fit remains below zero, starting at about 300 K. This again indicates a systematic underestimation. It could also be concluded that in the upper half of the critical temperature range, the prediction errors seem to be more significant in the direction of a lower temperature. Compared to the kernel density estimation of the relative residuals of the extra trees regression model from Figure 3.10, the kernel density estimation of the relative residuals of the gradient boosting regression model trained on the reduced feature set, the main peak of the distribution in Figure 3.11 is broader, which is no surprise as the gradient boosting regression models performance metrics, quantifying the predictive power and accuracy, already indicate the decreased predictive capabilities compared to the extra trees regression trained on all available features including the magnetic properties.

Outliers

Slope of Fitted Line

Systematic Underestimation

Wider Main Peak Compared to Complete Feature Set

Considering that the performance of the models, which were trained without the magnetic descriptors, is considerably decreased and even significant outliers like those seen with the 800 K residual can occur, this approach is probably not well suited for an accurate materials screening approach. Hence, if the materials screening procedure should not depend on the DFT-originated features, transforming the problem into a classification task and, therefore, simplifying the modeling task might represent a suitable approach to enable a materials screening application that does not rely on the output of DFT calculations.

Moving to Classification Task

Classification

Moving to a classification task reduces the modeling complexity by construction, as the goal is not anymore to predict a certain value but rather classify the critical temperature into high and low, which corresponds to “potentially relevant” for applications requiring stable magnetism at operating temperatures and “likely not relevant” for such applications. The models used for the classification task are listed in Table 3.6.

Class Interpretation

Linear	Non-linear	Ensemble	Indirect
Logistic Regression	K-Nearest Neighbors Classification Decision Tree Classification	Random Forest Classification Extra Trees Classification Gradient Boosting Classification XGBoost Classification	Extra Trees Classification LASSO

Table 3.6: Overview of the models evaluated for the classification task using the training data on the processed JuHemd data set. The models are grouped by their categorization as either linear, non-linear, ensemble or indirect estimator. As the linear Logistic Regression model for classification was not discussed in section 2.5.2, details on the model architecture and the underlying theory can be found in [269, 270].

The group of “Indirect” classification models shown in Table 3.6 represent the previously trained regression models, for which their predictions were classified according to the previously mentioned classification thresholds. While it is unusual to use regression models and turn them into classification models by simply labeling the predictions with a class based on a threshold, this provides the opportunity to compare regression and classification models. This is useful as scores for classification models are naturally elevated compared to regression models due to the more restricted label space. Hence, the scores of the indirect models serve in assisting the reader to justify the quality of the classification models.

The evaluation results of the classification models trained on the entire set of descriptors, including the DFT-originated features, are shown in Table 3.7.

	CV-Score	Train F1	Test F1	Test Accuracy	Test Precision	Test Recall
Logistic Regression	0.82	0.91	0.86	0.89	0.85	0.88
K-Nearest Neighbors	0.66	0.85	0.88	0.91	1	0.78
Decision Tree	0.71	1	0.84	0.88	0.87	0.81
Random Forest	0.84	1	0.89	0.91	0.88	0.91
Extra Trees	0.82	1	0.91	0.93	0.91	0.91
Gradient Boosting	0.83	1	0.84	0.87	0.8	0.88
XGBoost	0.84	1	0.86	0.89	0.87	0.84
Indirect Extra Trees	n/a.	1	0.88	0.9	0.88	0.88
Indirect LASSO	n/a.	0.86	0.81	0.85	0.81	0.81

Table 3.7: Classification CV and test evaluation results of models trained on the complete descriptor set, including descriptors acquired by the *ab initio* computation. The models are grouped into linear, non-linear, ensemble, and indirect classification models.

From Table 3.7, a few observations are apparent. It can be seen that the linear logistic regression model exhibits a similar performance as the much more complex ensemble models. Additionally, comparing the train and test performances, it is apparent that the logistic regression model, while reaching similar performance scores as the ensemble models, exhibits less overfitting than the ensemble models. The K-Nearest Neighbors classification model, while not achieving a great CV-score, displays the least degree of overfitting and increased performance compared to the regression task. For the K-Nearest Neighbors model, it is apparent that the additional 20 % training data, which were used before computing the test scores, had a significant impact on the model's predictive capabilities. Overall, the performance scores have improved compared to the regression task, which reflects that the classification task poses a simpler task than the regression task. As the indirect models were trained on the regression task, the model selection and validation used the R^2 -based 4-fold CV-score; the CV-score has not been included in Table 3.7 as well as Table 3.8, as the score is based on a different metric and is hence not comparable to the other score contained in the corresponding tables. It can also be seen from Table 3.7 that the indirect classification models perform similarly to the direct classification models of similar model type, i.e. the linear logistic regression model and the ensemble models. In Figure 3.12, both the logistic regression model's and the extra trees classification model's confusion matrices of the test set predictions are shown.

Linear Model's
Performance

Indirect
Classification
Performance
Similar to Direct
Approach

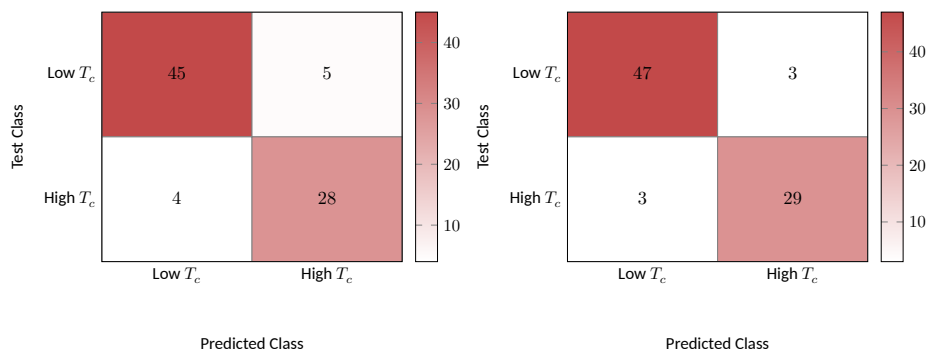


Figure 3.12: Confusion matrix of the predictions on the test set acquired from the logistic regression (left) and the extra trees (right) classification models trained on the entire set of features, including such resulting from the *ab initio* computation step of the critical temperature T_c .

From the confusion matrices in Figure 3.12 it becomes apparent, as already suggested from the metrics displayed in Table 3.7, that the ensemble model manages to lower the number of false classifications by a factor of one-third compared to the logistic regression classification model. While the metrics provide an abstract understanding of the model's performances, the confusion matrices clearly depict the strong predictive capabilities of the classification models on this particular task. Only three out of 32 high T_c compounds were falsely classified as structures with low critical temperature on unseen data. In a materials screening setting, this would translate to a high T_c compounds rate, which would be overlooked by the classification model of below 10 %. However, to screen a compound, the DFT calculation would still be necessary in this approach. Training the models on the DFT independent feature set and evaluating their performance using the discussed metrics and methods leads to the results shown in Table 3.8.

F_n Rate < 10 %

	CV-Score	Train F1	Test F1	Test Accuracy	Test Precision	Test Recall
Logistic Regression	0.68	0.75	0.75	0.79	0.7	0.81
K-Nearest Neighbors	0.6	0.8	0.88	0.91	1	0.78
Decision Tree	0.63	1	0.78	0.82	0.73	0.84
Random Forest	0.74	1	0.85	0.88	0.82	0.88
Extra Trees	0.74	1	0.84	0.87	0.8	0.88
Gradient Boosting	0.74	0.97	0.9	0.93	0.93	0.88
XGBoost	0.76	0.96	0.87	0.89	0.83	0.91
Indirect Extra Trees	n/a.	1	0.92	0.94	0.94	0.91
Indirect LASSO	n/a.	0.75	0.84	0.88	0.84	0.84

Table 3.8: Classification CV and test evaluation results of models trained on the reduced descriptor set, excluding descriptors acquired by the *ab initio* computation. The models are grouped into linear, non-linear, ensemble, and indirect classification models.

As can be seen from Table 3.8, the CV-scores of all trained classification models dropped unanimously. While this is expected, as the modeling complexity is increased with the magnetic descriptors withheld from the models, it is interesting to observe that the performance drop on the unseen test set is less significant. Furthermore, observing that the logistic regression model is neither under nor overfit is an additional indicator that the modeling complexity has increased compared to the modeling task where magnetic information was included in the set of descriptors. Still, the ensemble models outperform other examined models. This also holds for the indirect extra trees classification model, for which the test score outperforms every other test score in Table 3.8. Hence, the confusion matrices of the XGBoost classification model and the indirect extra trees classification model obtained by predicting the test set data using each model are shown in Figure 3.13.

Increased
Modeling
Complexity

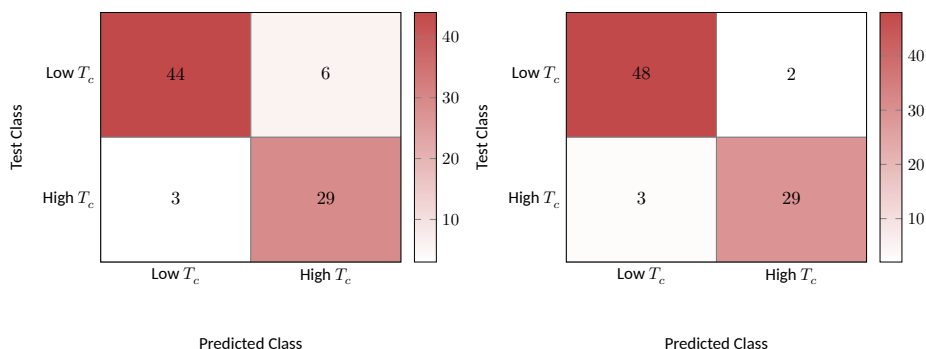


Figure 3.13: Confusion matrix of the predictions on the test set acquired from the XGBoost (left) and the indirect extra trees (right) classification models trained on the reduced set of features, excluding such resulting from the *ab initio* computation step of the critical temperature T_c .

Examining Figure 3.13 it is obvious that the XGBoost classification model displayed on the left, while having the same amount of false negative classifications as the extra trees classification trained on the entire feature set, the XGBoost classification model suggests a larger number of false positives on the test set. Surprisingly, the indirect classification model built on the extra trees regression model trained on the reduced feature set, displayed on the right side of Figure 3.13 is outperforming every other examined classification model trained on the reduced feature set and even outperforms the direct extra trees classification model's test performance included on the right of Figure 3.12, which was trained on the entire feature set available. The fact that the indirect classification model outperforms the other classification models is likely a coincidence rather than something that occurs regularly. However, this concludes that also without performing the *ab initio* calculations, a materials screening process would be possible and beneficial, as e.g. the indirect extra trees classification model maintains the false negative classification rate of below 10 % even in the absence of the DFT-originated features. This allows for large-scale and computationally very inexpensive materials screening applications. This demonstrates the capabilities of ML models within materials science to add value to the scientific community based on existing scientific data.

Indirect
Classification
Outperforming
Other Models

F_n Rate < 10 %
Even Without
DFT-originated
Features

3.1.5 Explainable Artificial Intelligence

While obtaining accurate predictions and quantifying model performances is important for application cases, the discussed metrics do not allow for an insight on why an individual data point has been predicted to a certain value or class. However, this insight can be acquired using XAI methods such as SHAP values discussed in section 2.5.3. In the following, plots of SHAP values, which provide an overview of the impact of individual features on the model's predictions, are discussed. These depictions are referred to as SHAP summary plots. While this discussion could be conducted for both regression as well as classification tasks, the discussion in this section is restricted to selected regression models. However, the discussion of the SHAP values corresponding to the classification models would be analogous. In this section, the discussion is also restricted to the nine features with the largest sum of impact magnitudes for each model due to the large feature space in both the entire and the restricted feature set. To generate the SHAP values depictions, the model is passed to the SHAP package's [128] explainer routine, and the marginal feature contribution to the overall model prediction, away from the base value (the average model prediction), is determined for data points which were part of the training data set. In this case, using the entire training data set to generate the feature-importance depictions was feasible. The SHAP values for each of the nine features with the largest sum of impact magnitudes are shown for the extra trees regression model trained on the entire feature set (as included in Table 3.4) are shown in Figure 3.14. For each SHAP summary plot, a selection of features, which is indicated to be remarkably impactful to the model's prediction by the depiction of the features SHAP values, is depicted separately. This approach represents an example of data-driven science and analysis.

SHAP Summary
Plots

Nine Most
Impactful
Features

From Figure 3.14, it can be seen that all of the nine most impactful features except for one (the electronegativity of the atom at the inequivalent lattice site two which is denoted by $\chi^{(2)}$) represent magnetic properties of the compound. However, also the interplay of atomic electronegativities in Heusler alloys is known to affect the magnetic properties of the compound [271, 272], which explains the occurrence of an atomic electronegativity in the SHAP summary plot from Figure 3.14. Four of the features displayed in the SHAP summary plot represent DFT-originated features, namely M_{Abs} , $|m_1|$, M , and m_1 . Apparently, the magnetism of the element in the first position in the molecular formula is more important to the model's prediction than the moments at the other positions. This is expected for all $L2_1$ compounds, representing the majority in the data set, as this element occupies half of the alloy's structural lattice sites.

Eight of Nine
Most Impactful
Features Related
to Magnetism

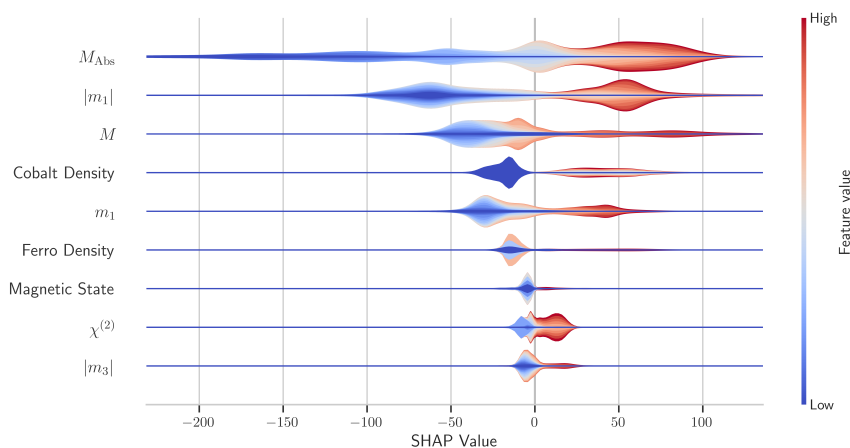


Figure 3.14: SHAP values for the nine features with the largest accumulated magnitude of impact on the prediction from the prediction base values of the extra trees regression model trained on the full set of features. SHAP summary plots like this one relate feature values (color) to the feature impact on the prediction (SHAP value). The layered coloring serves the purpose of visualizing the number of data points in the region of SHAP values (width of color). The color itself indicates the feature value relative to the mean feature value. This figure is adapted from [231].

This impact on the model's prediction emerging from this element appears in both the absolute and the actual value of the magnetic moment, including the sign. Hence, for many of the structures in the database, the magnetic moment arising from the element in the first position in the molecular formula determines a large proportion of the overall magnetic moment. For all quantities, except the density of ferromagnetic constituents, an increasing SHAP value of the feature seems to correlate with an increasing feature value. However, this proportionality is less clearly displayed for the magnetic state, which is expected as the magnetic state descriptor is represented by an integer encoding the magnetic state type and manifestation degree. The magnetic state consists of digits ordered like *e.g.* FAS. While each letter represents the manifestation degree of either an FM (F), an AFM (A), or spin-spiral (S) state extracted from the JuHemd, the values of each digit reach from 0 to 9. A state code of 000 hence represents a non-classifiable magnetic state according to the JuHemd. Hence, it is clear that large numbers for the state encoding correlate to stable FM states, which are beneficial to a larger critical temperature, which is indicated by the present red area at the magnetic state feature on the right. From Figure 3.14, it is apparent that the sum of absolute magnetic moments and the total moment within a compound are considered impactful to the prediction outcome by the extra trees regression model. Hence, both quantities are shown in Figure 3.15. Additionally, both the magnetic moment and the absolute magnetic moment of the element on the first position in the molecular formula of the compound are shown in Figure 3.16.

From Figure 3.15, it can be seen that a decent amount of compounds exhibit relatively large critical temperatures but, at the same time, a vanishing compound's magnetic moment, however, not a vanishing sum of absolute moments. This is an indication of strong AFM configurations.

Magnetic State
Interpretation

Strong AFM
Configurations

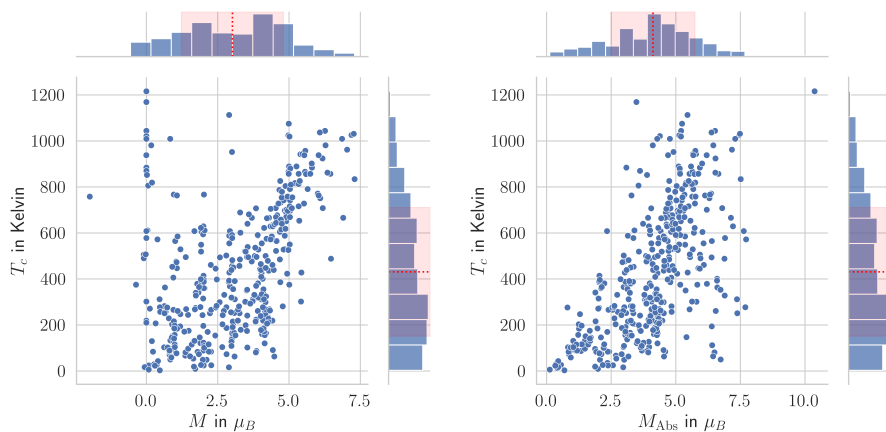


Figure 3.15: Depiction of the unsplit, but processed GGA compound's total moments in comparison to the theoretical, critical temperature of the compound (left) and the sum of absolute magnetic moments of the compound compared to the corresponding critical temperature (right). The marginal distributions, averages (dashed red line), and standard deviation intervals (red colored area) are added for the reader's convenience. Right figure is adapted from [231].

However, when the sum of absolute moments is examined, it becomes clear that increasing values M_{Abs} correlate with larger critical temperatures. More precisely, large critical temperatures do not occur for compounds with small sums of absolute magnetic moments. At the same time, a large M_{Abs} does not conclude that the corresponding compound has a high value of T_c . This caused the lower right triangle shape constituted by the data points on the right of Figure 3.15. This relation can be expressed as in equation (3.3), using a constant C , and is an example of the mentioned non-linearity of the task to model the critical temperature. [231] In this sense, the absolute magnetic moment is acting as an upper boundary for the critical temperature.

Triangle Shape

$$T_c \leq CM_{\text{Abs}} \quad (3.3)$$

From Figure 3.16, it can be seen that a significant number of atomic magnetic moments at the first molecular formula position are located close to zero or even exactly zero. However, examining the depictions, it can be seen that there is no clear relation between this particular magnetic moment and the critical temperature. While a larger moment generally correlates with a larger critical temperature, the relation between both quantities follows no simple, *e.g.* linear relation. Beyond that, the depictions in Figure 3.16 reveal properties of the magnetic moment, specific to the moment of the first element X of the compound's molecular formula, emerging from the processing of the JuHemd. Considering the theoretical upper moment limit for d magnetism is situated at $5 \mu_B$ and the upper limit for f magnetism is located at $7 \mu_B$, the moments appear too large. However, it is essential to keep in mind that for *e.g.* both the $L2_1$ and XA phase of the corresponding magnetic moment originating from the atom constituent X actually contain two lattice site's magnetic moments. A more extreme case represents the binary alloy Cr_3Al with a theoretical GGA critical temperature of 1216 K. In this case, the magnetic moment at the first molecular formula position is combined out of six individual moments of this AFM alloy, as the AFM structure was computed using a larger unit cell.

Discussion of m_1 and $|m_1|$

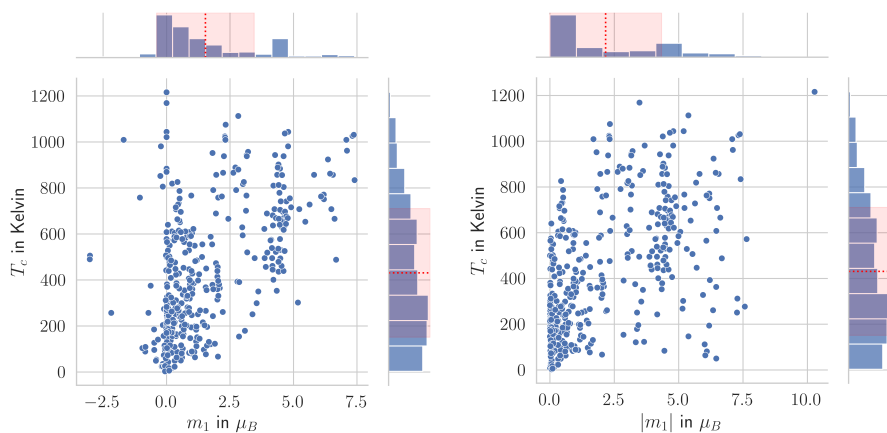


Figure 3.16: Depiction of the unsplitted, but processed GGA magnetic moments emerging from the element on the first position in the molecular formula in comparison to the theoretical, critical temperature of the compound (left) and the corresponding absolute magnetic moment compared to the critical temperature (right) of the alloy. The marginal distributions, averages (dashed red line), and standard deviation intervals (red colored area) are added for the reader's convenience.

Despite that the magnetic moments are only determined per unique constituting element and not per site and unique constituting element occupying the site, which would accumulate to a significant number of descriptors when disordered compounds are considered, represents an oversimplification of the site's occupations, the ML models can relate this simplified quantity to the critical temperature.

The SHAP values corresponding to a selection of the most impactful features on the gradient boosting model's regression, which was trained without the DFT-originated descriptors, are displayed in Figure 3.17. From Figure 3.17 it can be seen that only two features are directly associated with magnetic compounds, namely the fractional cobalt density and the fractional density of FM atoms (*i.e.* Fe, Co, and Ni) in the compound. From the SHAP summary plot, it can be told that very large fractional densities of FM atoms within a compound strongly contribute to the gradient-boosted trees regression model, predicting a large critical temperature for that particular compound. However, low and average values of the FM density seem to have a lowering impact on the critical temperature prediction. In contrast to the FM density, the impact attributed to the cobalt density is more clear, as very low amounts of cobalt (*i.e.* absence of cobalt) within a compound cause the model to predict a reduced critical temperature, while larger values of the fractional cobalt density appear to have an increasing effect on the T_c prediction. The individual densities of the FM constituents and the combined FM density, in relation to the critical temperature, are depicted in Figure 3.18. Both the cobalt and the FM density were also among the nine features that were shown in the SHAP summary plot obtained by the model, which has been trained on all features, including such based on *ab initio* results.

Fractional
Densities

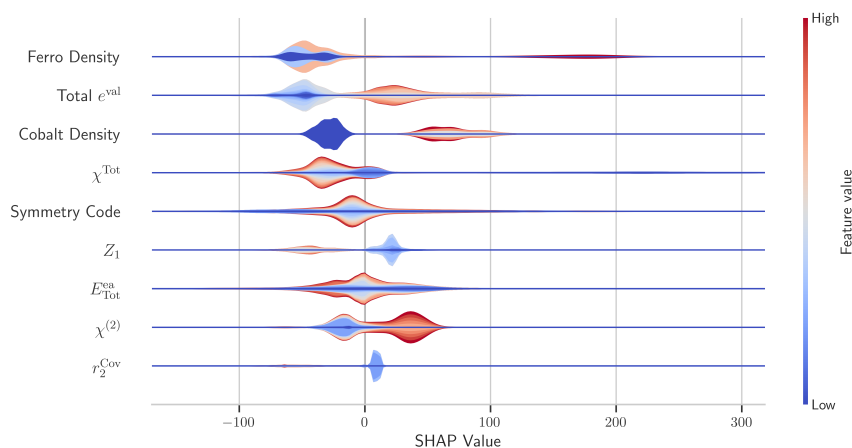


Figure 3.17: SHAP values for the nine features with the largest accumulated magnitude of impact on the prediction from the prediction base values of the gradient boosted trees regression model trained on the reduced set of descriptors, excluding the DFT-originated magnetic features.

From Figure 3.17, it can be seen that the total (summation of atomic) electronegativity has a significant impact on the model's prediction. While this is not surprising, as already previously discussed, the electronegativity is indeed related to the magnetic properties of a Heusler alloy; interestingly, a large sum of total electronegativities appears to be related to lower predictions of the critical temperature. However, it is important to mention that a sum of electronegativities indeed does not represent a measurable physical quantity but implies properties of the individual alloy's constituents. Furthermore, it is observable that the total number of valence electrons within the Heusler compound exhibits a significant impact on the model's T_c prediction. However, it is well-known for Heusler alloys that the number of valence electrons is relevant to a plurality of material properties, including magnetic phenomena. [273, 274] The relation between the total number of valence electrons and the critical temperature is shown in Figure 3.19. From the previous observations, it is safe to say that in the case of the model, which was trained under the exclusion of DFT-originated features, other quantities, which are closely related to the DFT-originated magnetic properties appeared in the SHAP summary plot as those features were excluded. Hence, these quantities seem to provide the model information, which the model did not need to rely on when the DFT-originated features were available. It is noteworthy that the model successfully identifies a correlation between some features and the critical temperature. For these features, this is consistent with the existing physical understanding. While it is intuitively clear that the symmetry code and, hence, the structure of the compound affects the magnetic properties, the interpretation of the symmetry code's impact in this plot is arbitrary, as the symmetry's encoding here represents an integer which is assigned to the compound, based on the order of occurrence of the compounds corresponding symmetry group in a list of symmetry groups uniquely occurring within JuHemd. Hence, the SHAP summary plot shows no apparent relation to this particular feature. Unsurprisingly, the SHAP values indicate that the atomic number of the first site Z_1 has an impact on the critical temperature's prediction. It seems that lower values for the first site's atomic number might be beneficial to a higher magnetic critical temperature.

Total
Electronegativity

Total Number of
Valence
Electrons

Interpretability of
the Symmetry
Group

Inspecting Figure 3.4, it is clear that lower relative feature values for the atomic number on site one correspond to atoms in the range of $3d$ transition metals, which are known to exhibit magnetic properties. Examining Figure 3.18, one can see that there are trends of proportionality between the critical temperature and the corresponding density visible for the iron density as well as the FM density. However, neither for cobalt nor nickel this trend could be observed. The proportionality is very clear for the FM density. This observation validates the order provided in the SHAP summary plot in Figure 3.17. From this SHAP summary plot, it can also be seen that especially low densities of cobalt are—on average—contributing to a prediction of a lower critical temperature, which is in line with the relation of the cobalt density with the critical temperature due to the very populated lower T_c region in the depiction for low cobalt fractions.

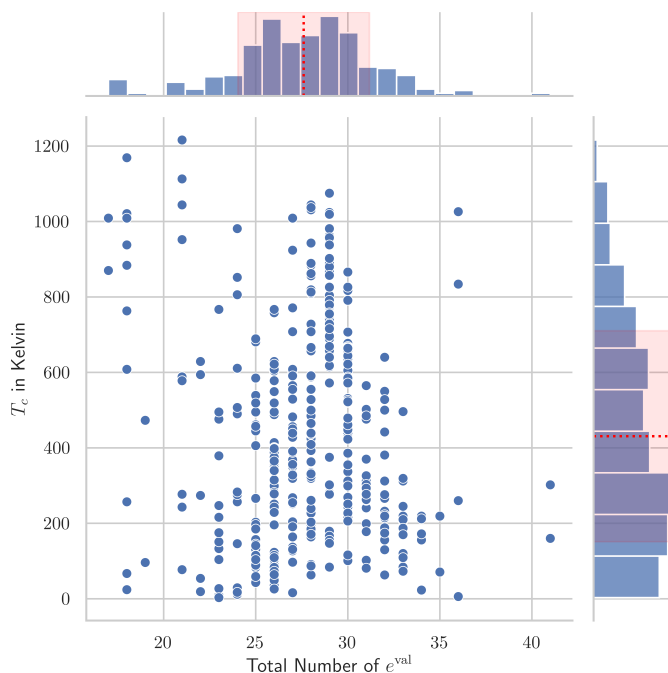
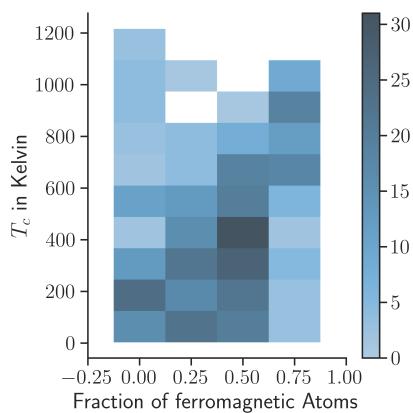
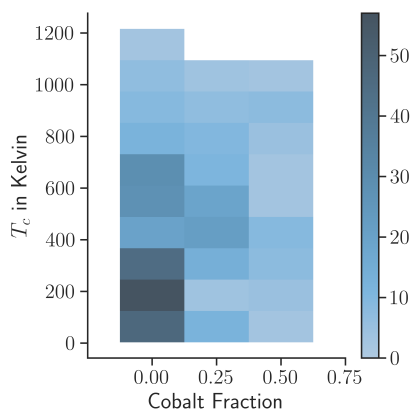


Figure 3.19: Relation between the total number of valence electrons of the compound to the critical temperature on the whole GGA data set. The marginal distributions, averages (dashed red line), and standard deviation intervals (red colored area) are added for the reader's convenience.

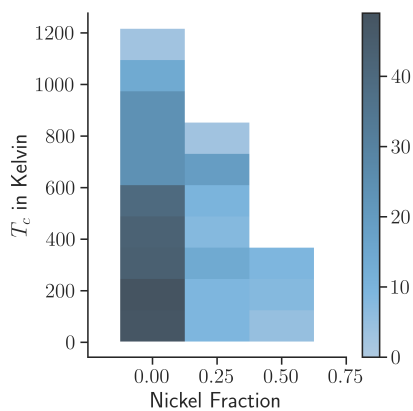
While there is no simple trend visible in the depiction of the total number of valence electrons of a compound in comparison to the corresponding critical temperature, the discretely clustered data points seem first to increase and then lower in the average critical temperature for increasing valence electron numbers. This average T_c peaks at about 29 valence electrons within the compound. It is understandable that a model considers this relation when concluding a prediction. The fact that this relation is indeed non-linear is already visible from the corresponding SHAP value distribution shown in Figure 3.17.



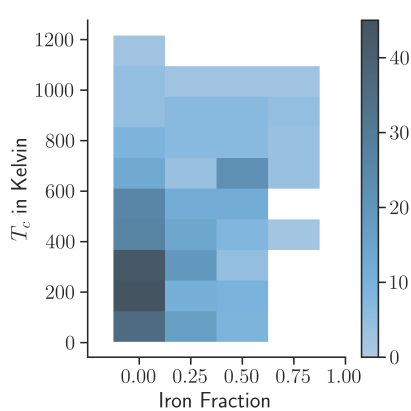
(a) Fractional density of FM atoms



(b) Fractional density of cobalt



(c) Fractional density of nickel



(d) Fractional density of iron

Figure 3.18: Relation between the compound's fractional densities of FM (i.e. Fe, Ni, and Co) atoms, cobalt, nickel, and iron to the critical temperature of the whole GGA data set. The subfigures (a), (b), and (c) are adapted from [231].

3.2 Half-Metallicity in $L2_1$ and XA Heusler Alloys

Half-Metallicity
as Magnetic
Phenomenon

Similar to the critical temperature, half-metallicity as a property of magnetic materials is the result of complex physical interactions and an interplay (like e.g. hybridization of orbitals) within the material related to e.g. elemental composition, electron configuration, and structural disorder. [75, 275] Materials that are conducting in a spin state (by convention the spin-up state is denoted as $\uparrow$) but at the same time do not conduct in the other spin state (spin-down is denoted $\downarrow$)—i.e. the materials have a gap in the DOS of the minority spin—have applications in concepts of spintronics, in particular for spin-injection devices. [276] These concepts include e.g. the concept of magnetic random access memory (MRAM), which—in contrast to semiconductor-based dynamic RAM—represents a non-volatile memory technology, which is already applied in special use cases. Due to the reduced consumption of power, the further development of the MRAM technology—which allows an application beyond the established areas—would be desirable to reduce the power consumption of IT systems. [277]

Magnetic
Random Access
Memory

Slater-Pauling
Behavior & *Ab
Initio*

In the particular case of the ordered Heusler alloys phases $L2_1$ and XA, the known Slater-Pauling behavior is able to identify half-metallic compounds based on their elemental composition and magnetic configuration. [75, 278] While the Slater-Pauling behavior is extremely helpful when screening for half-metallic compounds, the relations formulated by [75] rely on the fact that the total magnetic moment of a half-metallic alloy has an integer value. However, as *ab initio* calculations include approximations, there is an error margin for the magnetic moment present even when using the Slater-Pauling behavior for materials screening. This section demonstrates the application of ML methods for predictive modeling in this particular materials screening application, including an analysis of the predictions compared to the Slater-Pauling behavior and the use of the SHAP package to explain the relations learned by the used model.

3.2.1 Data

Ab Initio Data

The data used for training, validation, and testing, which contains spin-polarization fractions at the Fermi energy for different $L2_1$ and XA Heusler compounds, was collected by collaborators from the University of Alabama and published by the time of writing this thesis. [216] The fractions have been determined using DOS computations obtained using the plane-wave pseudopotential code QUANTUM ESPRESSO. [279–281]

Over time, two different iterations of the mentioned data were available for the presented discussion. The number of individual data points per phase and both phases combined of the different data set versions are shown in Table 3.9. From Table 3.9, it can be seen that in both data set iterations, the amount of data points is balanced for both $L2_1$ and XA phases. The polarization fractions at the Fermi energy contained in the data set are depicted in Figure 3.20.

Data Version	$L2_1$ Compounds	XA Compounds	Combined
1	98	98	196
2	179	165	344

Table 3.9: Overview of data set sizes per phase at both iterations of the data set.

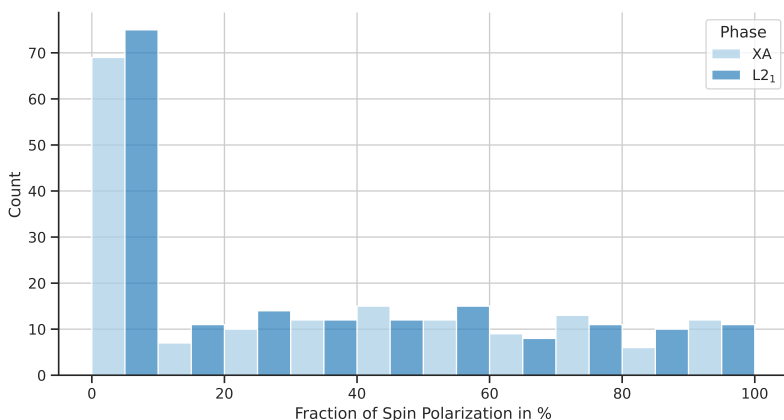


Figure 3.20: Distribution of the spin-polarization at the Fermi level within the data set. Each vertical bar indicates a 10 % range starting from zero polarization on the left. The different colored bars show the distribution for L2₁ and XA separately.

From Figure 3.20, it is evident that the lower spin-polarization fractions at the Fermi energy are dominant within this data set. Using the presented data as a training database for a materials screening application, which can be translated to an ML classification task, which uses structural and magnetic data to classify compounds for half-metallicity as an inherent property.

Structural	Magnetic
Label [†]	M^*
Phase [‡]	m_{X_1}
Lattice Constant	m_{X_2}
	m_Y
[†] X_2YZ , * $M = \sum_i m_i$ [‡] L2 ₁ or XA	

Table 3.10: Overview of features stored alongside the spin-polarization at the Fermi level within the database.

Using the descriptors from Table 3.10, during this study, a set of features has been constructed, which is shown in Table 3.11. The descriptors constructed, using the features from Table 3.10, accumulate to a total of 15 features. With this number, the training data is well within the one-in-ten rule. Also, it is apparent from Table 3.10, that the magnetic moment of the main group element within the Heusler alloy was missing in the initial database. While the magnetic moments consider the respective elements, the magnetic moment might not have a significant size or impact on the model prediction. However, this should not be assumed beforehand and, therefore, was examined in the XAI analysis and model evaluation process instead of simply omitting the feature entirely. Hence, as m_Z was not directly contained in the database, the value was determined using the formula (3.4).

$$m_Z = M - m_{X_1} - m_{X_2} - m_Y \quad (3.4)$$

Structural	Magnetic	Electronic
Z_X	m_{X_1}	e_X^{val}
Z_Y	m_{X_2}	e_Y^{val}
Z_Z	m_Y	e_Z^{val}
Phase [†]	m_Z	$e_{\text{Tot}}^{\text{val}}$
Lattice Constant	M	
	$M_{\text{Abs}}^{\ddagger}$	

[†] One Hot Encoded ($L2_1 = 1$ & $XA = 0$)

[‡] $M_{\text{Abs}} = \sum_i |m_i|$

Table 3.11: Overview of descriptors derived from those included in the data base.

Using the spin-polarized DOS $\rho_\sigma(E)$ acquired using *ab initio* calculations, the spin-polarization fraction at the Fermi energy $P_S(E_F)$ is determined using the expression from equation (3.5).

$$P_S(E_F) = \frac{\rho_\uparrow(E_F) - \rho_\downarrow(E_F)}{\rho_\uparrow(E_F) + \rho_\downarrow(E_F)} \quad (3.5)$$

From Figure 3.21 it can be seen that only six different elements occupy the Z site in the $L2_1$ Heusler structure within the training data. On the other hand, the X and Y structure sites are occupied by various 3d and a few 4d and 5d (only on the X site) transition metals. Of course, since half-metallicity represents a magnetic phenomenon, the strong involvement of 3d transition metals is expected. This description also applies to the atomic number distribution of the lattice sites for the inverse Heusler XA phase, as shown in Figure 3.22.

Site-Specific
Elemental
Composition in
Training Data

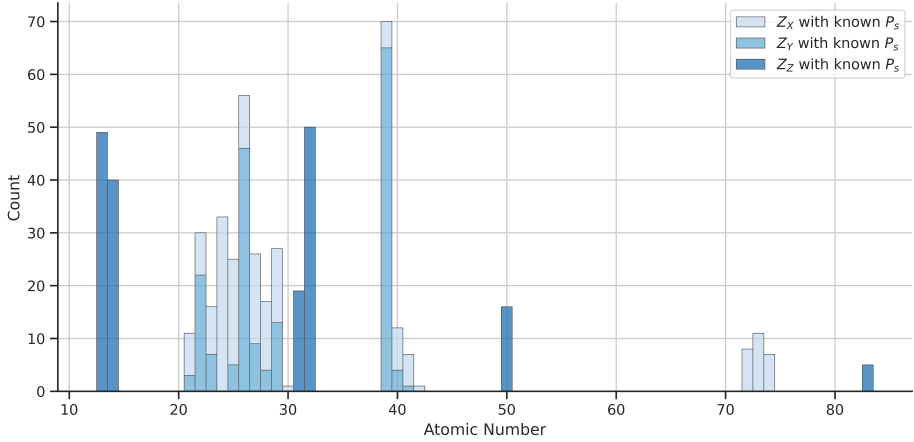


Figure 3.21: Atomic number distribution for the $L2_1$ phase structures within the training database, colored to depict the occupation of different structural sites.

However, when moving to the materials screening using compounds different from the training database, one should be aware that, the less similar a predicted compound is compared to compounds from the training database, the more OOS this particular prediction will be. It is clear that in an ML-assisted materials screening application, some degree of OOS prediction is unavoidable to discover new materials. Still, this should be kept in mind when judging a model's predictions. A way to quantify the individual OOS prediction degree would be to count the number of elements in the screened compound, which are known from the training data at the corresponding sites. If all elements on the individual compound sites were in the training set at some point, then the prediction would not be OOS. However, if the training data does not contain any elements of the compound at these specific sites—however, possibly on other sites—the prediction would be called entirely OOS. Generally, one would expect a higher predictive performance for screened compounds that are less OOS.

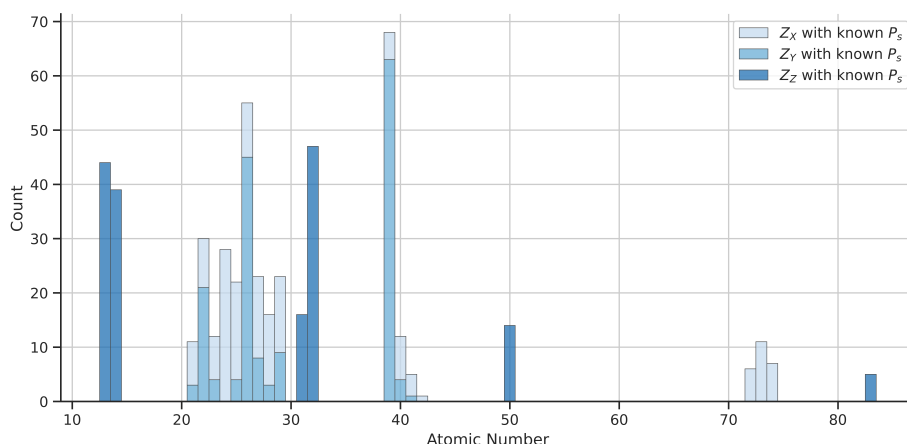


Figure 3.22: Atomic number distribution for the XA Heusler phase structures within the training database, colored to depict the occupation of different structural sites.

3.2.2 Modeling Goal

It has already been clarified that this particular study aims to screen materials *ab initio* data of ordered full and inverse Heusler alloys for half-metallic properties. However, to approach this task using ML modeling, the problem must be translated into a classification task. Given the value of $P_S(E_F)$ is continuous, it is necessary to define a threshold that separates the compounds from the training database into “high” grade spin-polarized and “low” grade spin-polarized. When defining this particular threshold, a few factors have to be considered:

Classification
Classes

- As seen in Figure 3.21 and Figure 3.22, the higher spin-polarization range is less represented in the data set, which might lead to larger error margins when predicting highly spin-polarized compounds
- At the same time, setting a threshold too high would also cause the training database to shrink in this particular polarization region
- A false negative error has to be weighted worse than a false positive, as missing a compound prevents further scientific examination, while a posterior validation using *ab initio* calculations of a false positive is able to rule out false positives easily

Considering the previously mentioned arguments, the interval for low-grade spin-polarization at the Fermi level was set to be $[0, 0.6)$ while the complement high-grade interval was set to be $[0.6, 1]$. While a spin-polarization fraction of 60 % at the Fermi level is quite far from actual half-metallic behavior, this ensures that fewer compounds get overlooked in the screening application and the training base for the higher region—within the larger data set size—still represents about 34 % of the training data. For the same reasons the threshold has been chosen, it was decided that the recall would be used to select and tune an appropriate model alongside a 4-fold CV procedure and 20 % of the original data set size as test data. After selection, tuning, and evaluation, the model is retrained on the entire available data to improve the data variety and size the model has been trained on before applying it to screen for (near) half-metallic alloys. Before the data splitting, the data has been randomized to avoid clustering. Also, a scaling, as shown in equation (3.2), has been performed for the data, which has been used for model training and evaluation but has also been applied to the screening data.

As mentioned, half-metallicity in Heusler $L2_1$ and XA phases can, in principle, be determined using the known Slater-Pauling behavior. This behavior implied that a half-metallic alloy would lie on either of the lines represented by equation (3.6). [75, 278] There exist similar rules for other materials classes such as the Heusler phases $C1_b$ and Y. [282]

$$M_{L2_1} = e_{\text{Tot}}^{\text{val}} - 24 \quad M_{\text{XA}} = \begin{cases} e_{\text{Tot}}^{\text{val}} - 28 \\ e_{\text{Tot}}^{\text{val}} - 24 \\ e_{\text{Tot}}^{\text{val}} - 18 \end{cases} \quad (3.6)$$

Physical Intuition
Behing
Slater-Pauling
Behavior

The physical reasoning behind these lines is that the total number of valence electrons within the compound is indeed an integer value. However, if also the total magnetic moment M is an integer value, both majority and minority band numbers must therefore be integers. This fact can be related to d orbital hybridization between the transition metal constituents hence—in the $L2_1$ phase—forming a total of 12 $\downarrow$ states below the Fermi level, which results in the rule for the $L2_1$ phase in equation (3.6). A similar reasoning holds for the XA phase but is well documented in the literature. [282]

It could be argued that if the Slater-Pauling behavior is known, using different materials screening approaches is unnecessary as one could apply the rules deduced from equation (3.6). However, there is no guarantee that the reasoning behind the Slater-Pauling behavior is the exclusive origin of half-metallicity in Heusler alloys. In fact, there are examples known, such as $L2_1$ Mn_2CuGe , which violate this rule with an integer total magnetic moment as it has 29 valence electrons while exhibiting half-metallicity at the same time. [283]

The computation result of magnetic moments by *ab initio* methods can be impacted by the applied DFT method, chosen XC functionals, and used convergence parameters. For some compounds, these computational differences can lead to a deviation of the total magnetic moment from a value that would otherwise be an integer value. Hence, this particular compound would be missed by simply applying the relation from equation (3.6). Also, the presence of spin-orbit interaction affects the spin-polarization of the DOS and, subsequently, the half-metallicity of a compound. [284] Combining these reasons, it is clear that other methods besides the Slater-Pauling behavior are needed to effectively screen Heusler compounds data, computed with *ab initio* methods, to discover half-metallic materials and complement existing methods. In Figure 3.23, the lines from equation (3.6) as well as the screened $L2_1$ and XA Heusler *ab initio* data obtained from the Materials Project [94] database are shown. For this depiction, a deviation of the total magnetic moment of $0.05 \mu_B$ from an integer value is in line with margins chosen by other works as e.g. by [75].

Necessity for
Screening
Methods Beyond
Slater-Pauling
Curve

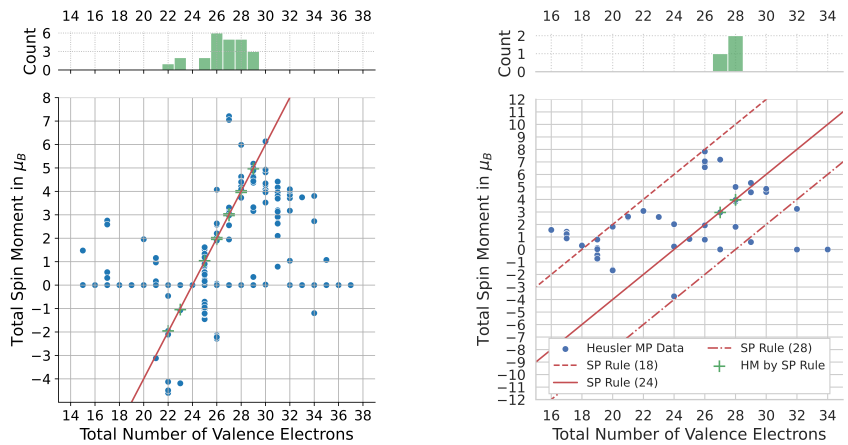


Figure 3.23: Depiction of Slater-Pauling behavior on the screened data from the Materials Project. [94] The markers meeting the criterion (red lines) from equation (3.6) are highlighted in green. The marginal distributions on top of the depictions show the distribution of the compounds meeting the Slater-Pauling behavior.

What cannot be seen directly from the scatter plots in Figure 3.23, but from the marginal distribution on top of the visualizations, is that 24 $L2_1$ and three XA compounds are situated exactly on the green cross markers, indicating that those compounds meet the relation from equation (3.6) and hence will display half-metallic properties.

27 Compounds
in Screening
Data Fulfilling
Slater-Pauling
Behavior

At the beginning of this study, a selection of models very similar to those examined in section 3.1 have been tested on the training database, and from this selection of models, the XGBoost [125] model has been chosen as the best fit for this particular classification problem. The hyperparameter optimization was performed using a grid search algorithm and was later replaced by a BO hyperparameter search algorithm. [285] Furthermore, in the last model iteration, which has been used for the XAI-based analysis of the model's predictions, the 4-fold CV process has been replaced by a stratified 4-fold CV approach in which the balance of classes is maintained within each validation set. It is known that stratified CV procedures can lead to improved model performances when imbalanced data is used for ML model training. [286, 287]

The following section outlines the workflow for the screening process, including details on how the screening data was obtained from the Materials Project [94] database, processed, filtered, and screened.

3.2.3 Screening Workflow

After the model selection and training—of the selected model—on the data, which was subsequently published in [216], the materials screening workflow first required the obtainment of suitable data, which in this case is *ab initio* magnetic and structure data from the Materials Project [94], for the screening itself. The Heusler phase $L2_1$ belongs to the space group $Fm\bar{3}m$, while the XA phase belongs to the space group $F\bar{4}3m$. It is possible to filter the database for space groups using the Materials Project API. However, not all compounds listed in the database under these particular space groups fulfill the definition of Heusler alloys discussed in section 3.1.1. Besides compounds that do not fall into the typical definition of a Heusler alloy, there are also compounds contained in the database, which simultaneously are part of the training database. While the main goal of this study is the examination of half-metallicity in full and inverse Heusler alloys, on the side also transition metal Heusler type alloys, which were predicted by multiple model versions used in this study to have a highly spin-polarized DOS at the Fermi energy, have been examined, for which a brief discussion is included in a later section. An overview of the data set sizes, regarding the space groups initially as well as after the removal of both, compounds that appear in the training database and compounds which do not fall into the classical definition of Heusler alloys as by section 3.1.1 is given by Table 3.12.

Processing Stage	Set Size for Space Group $Fm\bar{3}m$	Set Size for Space Group $F\bar{4}3m$
Full Space Group Size	4394	200
Removed Training Compounds	4375	196
Applied Criteria for Heusler Composition	428	54

Table 3.12: Overview of screening data set sizes obtained from the Materials Project [94] before any removal of structures, after removal of structures which appeared in the training database, and after removing structures which do not fulfill the typical definition of Heusler alloys.

From the obtained structural data, the compilation of a screening database is straightforward. The data obtained from the Materials Project [94] includes the individual atomic magnetic moments, the atom types on the different sites, the volume of the cubic unit cell, and the corresponding space group. From these quantities, it is possible to construct the descriptors shown in Table 3.11, which were used to train the model. In principle, the Materials Project [94] database also contains DOS information for many compounds. However, together with collaborators from the University of Alabama [281], it was determined that the DOS data from this database is insufficient to screen for half-metallic compounds as e.g. Fe_2CrSi has a DOS [288, 289] which resolves both spin states, displaying metallic characteristics, in the database but is known to be half-metallic by e.g. [290]. This example also serves as a reminder to consider the quality of the examined data, as also discussed in section 2.1.2. While the data included in the Materials Project database [94] is computed mainly using the PW-based code VASP [291], it should be kept in mind that the computations available there are performed in a high-throughput fashion, which concludes the necessity to question if the data quality provided in such databases is sufficient for the intended use case. In this case, it was concluded that the DOS data is not sufficient for the intended screening application. However, computing a high-quality DOS is a much more challenging endeavor than performing an SCF *ab initio* calculation and thereby computing the magnetic moments based on ICSD [292] structures. Therefore, the chosen modeling approach is appropriate in this case because the magnetic data from the training database was also computed using a PW-based method.

Descriptor
Determination
from Screened
Data

Quality of DOS,
Magnetic, and
Structure Data

An overview of the screened compounds regarding the atomic numbers occupying each structural site is shown in Figure 3.24 for the L2_1 Heusler phase and in Figure 3.25 for the XA inverse Heusler phase.

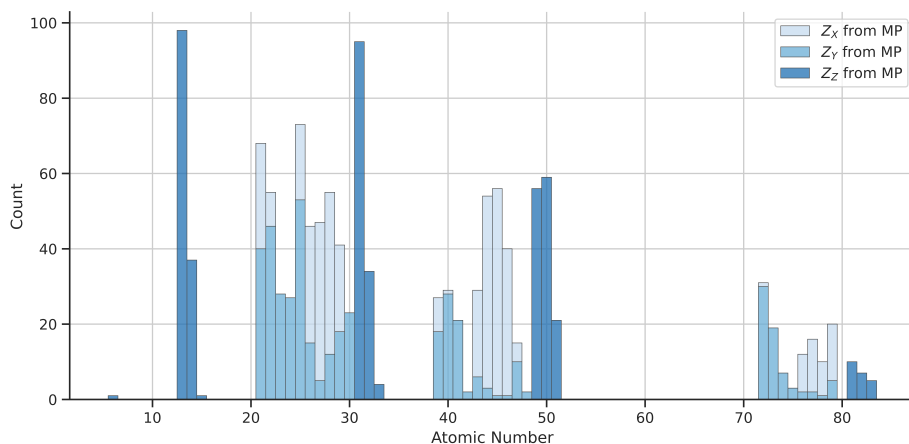


Figure 3.24: Atomic number distribution for the L2_1 phase structures in the screened structure's database obtained from the Materials Project [94], colored to depict the elemental occupation of different structural sites.

From Figure 3.24, it can be seen that the $4d$ and $5d$ region for the X and Y sites are much more populated than in the training set distribution from Figure 3.21. Also, compared to the training database, there are not five different elements occupying the Z site but rather 13 different elements.

Site-Specific
Elements in
Screened Data

The immediate consequence of both of these observations is that there will be a decent amount of predictions that this screening data will be OOS to some degree. However, as can be seen in Figure 3.25, this is not the case to this extent for the XA inverse Heusler phase, which can be attributed to the comparably lower number of structures available from the screened database for this phase.

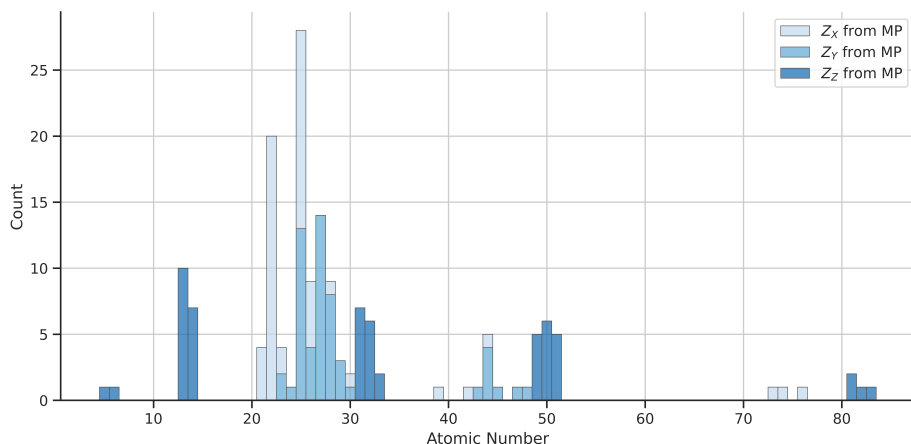


Figure 3.25: Atomic number distribution for the XA inverse Heusler phase structures in the screened structure's database obtained from the Materials Project [94], colored to depict the elemental occupation of different structural sites.

The code written and used during this study for interfacing with the Materials Project API, data processing, model training, model evaluation, prediction, and structured processing/filtering of the predictions has been published. [293]

3.2.4 Model Predictions

In the following, the predictions of the different model iterations and their differences are presented. Subsequently, using the last trained model, an analysis of the model's predictions on the test set data is outlined. This assists in understanding which features were relevant to the model for the classification of each individual compound.

Prediction Analysis

The data set, provided and collected by collaborators [281], was extended during the course of this study. Even before the data set was published [216], exploratory data analysis, experimentation with different models, and predictions were performed. This led to multiple model iterations and multiple batches of predictions based on the screening data set. At the beginning of this study, it was experimented with a three-class approach, separating the compounds in classes of vanishing P_S , $0 < P_S < 0.6$, and $P_S \geq 0.6$.

While this approach gives more distinct insights into the degree of DOS polarization at the Fermi level, this approach was both harder to interpret using XAI methods and had decreased test performance compared to a two-class model when it comes to finding compounds with high DOS spin-polarization at the Fermi level in a screening application. Hence, the three-class approach was swapped for a binary classification task. An overview of the different model iterations and what is distinct to them compared to the previous XGBoost model is given in Table 3.13.

Model Iteration	Difference to Previous Version	Test Recall	Test Accuracy	Test Size	Class Size $P_S \geq 60\%$
1	Initial Model Trained on 196 Data Points	50 %	75 %	40	12
2	Extended Training Data Base to 344 Data Points Included M_{Abs} as a Feature	60 %	86 %	69	15
3	Added Numbers of Valence Electrons as Features Used BO for Hyperparameter Optimization	66 %	86 %	69	15
4	Included m_z as a Feature Moved to stratified 4-fold CV	57 %	83 %	69	14

Table 3.13: Overview of different model iterations which were used during this study and the characteristics which differ compared to the previous iteration. All score values have been rounded to the second decimal point.

The 4th model iteration introduced in Table 3.13 was exclusively used for the XAI analysis of the relation the model has learned. Of course, in principle, the predictive process could have been repeated with arbitrary many versions of (potentially improved) models. However, the information gain per predictive iteration would be minimal at some point due to the cumulative number of predictions already made in the previous model iterations. This was also the case here, which is why this last model iteration was only included in this thesis for the XAI discussion. However, while it is intuitive that a weakly magnetic element at the Z site is contributing little to the overall half-metallicity of the compound, it would introduce an omitted variable bias not to include this magnetic compound property based on this intuition, which is why it was in the feature importance analysis. As m_Z was not included initially in the training data, this particular feature was not added earlier. While the test performance is comparable across all models, the 4th model represents the least biased model regarding the human-made decision on which features to include. The last model iteration and the previous data splitting were initialized using a fixed random seed—for increased reproducibility—hence, the test set's size and performance changed slightly compared to the other models.

The following presents the predictions of the different model iterations on the screening data set. For each prediction of a compound with high spin-polarization of the DOS at the Fermi energy, a literature search has been conducted. The subsequently collected results of this search are included in the Tables 3.14, 3.15, and 3.16. They are accompanied by an indication of which model iterations predicted this particular compound to have $P_S \geq 60\%$ and the degree to which this particular prediction is OOS. If the table entry related to the literature search is labeled with “No Source” this means that at the time of writing this thesis, no literature stating *ab initio*-based conclusions about the half-metallicity of this particular compound has been found during the search. The OOS column in each table quantifies how many elements within the individual compound are occurring on the specific sites in the training data.

“No” in this sense means that all elements were occurring in the training data at the corresponding sites. “Slightly” means all elements but a single element were occurring in the training data at the corresponding sites. “Considerably” means that only a single element of the compound was known at that particular site. “Entirely” means that not a single element in the given compound was included in the training set at the specific sites.

Whenever published *ab initio* results were found that characterize the compound as half-metallic, nearly half-metallic, or the particular compound was reported with an explicit value for P_S , which is situated above the classification threshold, the compound was not examined further. If sources were found, the type of *ab initio* methods used in the found source is also included in the tables. However, if sources suggested other conductivity properties or no sources were found in a literature search, then a FLAPW electronic structure calculation was performed to decide if the model’s predictions were correct. This FLAPW validation will be discussed after the predictions of the individual model iterations are presented.

Also, suppose a compound was found to be half-metallic by the Slater-Pauling behavior shown in Figure 3.23. In that case, an “SP” in the literature columns indicates this in the tables. Of course, an unknown number of false negatives are involved with the prediction of highly spin-polarized DOS at the Fermi level. However, validating all the predictions—including the predicted low spin-polarized compounds is not feasible with the number of structures given. This is partly due to the computing time requirements associated with the electronic structure calculations. The predictions for the 1st model iteration are shown in Table 3.14. Compounds are ordered alphabetically in the following tables. Each table includes the compounds first predicted to have a highly spin-polarized DOS at the Fermi energy by the corresponding model iteration. Besides that, the information whether a later model also predicted the model to be at the higher range of P_S is included in an additional column; this way, multiple appearances of compounds are avoided.

Compound	Symmetry	Literature Conductivity	Model Version	OOS
Co ₂ FeIn	L2 ₁	HM by FLAPW [294]	1 & 2	Slightly
Co ₂ HfIn	L2 ₁	No Source	1	Considerably
Co ₂ MnSi	L2 ₁	HM by FLAPW [295]	1 & 2 & 3	Slightly
Mn ₂ CoGe	XA	HM by FLAPW [296] and SP	1 & 2 & 3	No
Mn ₂ CuGe	XA	No Source	1& 3	Slightly
Mn ₂ RuSi	XA	HM by PW [297]	1 & 3	Slightly
Rh ₂ FeSn	L2 ₁	No Source	1 & 3	Slightly
Ru ₂ FeGe	L2 ₁	Metallic but Polarized [298]	1 & 2 & 3	Slightly
Ru ₂ FeSi	L2 ₁	No Source	1 & 2	Slightly
Ti ₂ MnSn	XA	HM by PW-PAW [299]	1 & 2 & 3	Slightly

Table 3.14: Predictions of the 1st model iteration. Green cells mark for which compounds, either the literature search or the Slater-Pauling behavior, validated the model’s prediction of a DOS spin-polarization at the Fermi level of at least 60 %. Grey cells are validated later using a FLAPW electronic structure calculation. PAW is short for projector augmented wave method. HM in this table and the following tables is short for half-metallic.

From Table 3.14, it can be seen that at least half of the compounds, for which the 1st model predicted a highly spin-polarized DOS around the Fermi level, are actually half-metallic. The predictions of the 2nd model iteration, which did not already appear in Table 3.14 are shown in Table 3.15.

Compound	Symmetry	Literature Conductivity	Model Version	OOS
Co ₂ CrSb	L2 ₁	HM by PW [300]	2 & 3	Considerably
Co ₂ HfAl	L2 ₁	HM by FLAPW [301]	2 & 3	Slightly
Co ₂ HfGa	L2 ₁	HM by FLAPW [301]	2	Slightly
Co ₂ MnSb	L2 ₁	HM by FLAPW [302]	2 & 3	Slightly
Co ₂ ScGe	L2 ₁	$P_S = 60\%$ by FLAPW + U [303]	2	No
Co ₂ ZrGa	L2 ₁	Nearly HM by FLAPW [304]	2	No
Fe ₂ CrSb	L2 ₁	HM by FLAPW [305] and SP	2 & 3	Considerably
Fe ₂ MnP	L2 ₁	HM by FLAPW [306] and SP	2 & 3	Slightly
Fe ₂ TaGe	L2 ₁	HM by FLAPW [307]	2	Slightly
Fe ₂ TiAs	L2 ₁	HM by PW [308]	2	Slightly
Fe ₂ TiGa	L2 ₁	Nearly HM by PW [308]	2 & 3	No
Fe ₂ TiIn	L2 ₁	Nearly HM by PW [308]	2 & 3	Slightly
Fe ₂ TiSb	L2 ₁	HM by PW [308]	2	Slightly
Ir ₂ FeGa	L2 ₁	No Source	2	Slightly
Ir ₂ TcTi	L2 ₁	No Source	2	Entirely
Mn ₂ CoSb	XA	HM by FLAPW [299] and SP	2 & 3	Slightly
Mn ₂ TaGe	L2 ₁	No Source	2 & 3	Slightly
Mn ₂ VGe	L2 ₁	HM by ASW [309]	2	No
Mn ₂ WAl	L2 ₁	No Source	2 & 3	Slightly
Mn ₂ WGa	L2 ₁	HM by DFT [278]	2	Slightly
Ni ₂ MnSn	L2 ₁	Metallic by FLAPW [304]	3	No
Rh ₂ FeGa	L2 ₁	$P_S = 79\%$ by PW-PAW [310]	2	Slightly
Rh ₂ FeIn	L2 ₁	Nearly HM by PW-PAW [310]	2	Considerably
Rh ₂ MnSi	L2 ₁	No Source, known by SP	2	Slightly
Rh ₂ MnSn	L2 ₁	No Source, known by SP	2 & 3	Considerably
Ti ₂ CoAl	XA	HM by PW [311]	2 & 3	No
Ti ₂ CoGa	XA	HM by FLAPW [312]	2 & 3	No
Ti ₂ CoGe	XA	HM by PW-PAW [299]	2 & 3	No
Ti ₂ CoIn	XA	HM by FPLO [313]	2 & 3	Slightly
Ti ₂ CoSi	XA	HM by FLAPW [314]	2 & 3	No
Ti ₂ CuAl	XA	Nearly HM by FPLO [315]	2	No
Ti ₂ FeGa	XA	HM by FLAPW [299]	2 & 3	No
Ti ₂ NiAl	XA	HM by FLAPW [299]	2	No
Ti ₂ NiGa	XA	HM by FLAPW [299]	2	No
Ti ₂ NiIn	XA	HM by FPLO [313]	2	Slightly
Ti ₂ ZnAl	XA	No Source	2	Slightly

Table 3.15: Predictions of the 2nd model iteration, which did not occur in the previous model's predictions. Green cells mark for which compounds, either the literature search or the Slater-Pauling behavior, validated the model's prediction of a DOS spin-polarization at the Fermi level of at least 6.0 %. Grey cells are validated later using a FLAPW electronic structure calculation. ASW is short for the Augmented Spherical Wave method. FPLO is short for the full-potential local-orbital method.

From Table 3.15, it can be seen that 30 compounds are already known to exhibit half-metallic behavior, while for five compounds, no sources were found at the time of writing. For only one compound, it was found that it exhibits metallic properties, which will be investigated using the FLAPW method later. The predictions of compounds with highly spin-polarized DOS at the Fermi level, which first appeared in the 3rd model iteration, are shown in Table 3.16.

Compound	Symmetry	Literature Conductivity	OOS
Co ₂ CrGa	L2 ₁	HM by LMTO [316]	Slightly
Co ₂ CrSi	L2 ₁	HM by PW [317] and SP	Slightly
Co ₂ HfSn	L2 ₁	HM by FLAPW [318]	Slightly
Co ₂ NbSn	L2 ₁	Metallic by PW-PAW [304]	No
Co ₂ ScSn	L2 ₁	No Source	Slightly
Fe ₂ CoGa	L2 ₁	HM by LCAO [319]	No
Fe ₂ CoGe	L2 ₁	Metallic by PW [320]	No
Fe ₂ CoSi	L2 ₁	HM by FLAPW [296]	No
Fe ₂ CrGa	L2 ₁	HM by PW [321]	Slightly
Fe ₂ CrSi	L2 ₁	HM by FLAPW [290]	Slightly
Fe ₂ MnSi	L2 ₁	HM by PW-PAW [322]	No
Mn ₂ CuSb	XA	No Source	Slightly
Ti ₂ CoIn	L2 ₁	HM by FPLO [323]	Slightly
Ti ₂ CoIr	L2 ₁	No Source	Slightly

Table 3.16: Predictions of the 3rd model iteration, which did not occur in previous model predictions. Green cells mark for which compounds, either the literature search or the Slater-Pauling behavior, validated the model's prediction of a DOS spin-polarization at the Fermi level of at least 60 %. Grey cells are validated later using a FLAPW electronic structure calculation. LMTO is short for the linear muffin-tin orbital method. LCAO is short for the linear combination of atomic orbitals method.

From Table 3.16, it can be derived that nine compounds were correctly predicted to be highly spin-polarized by comparison with published work. For three compounds, sources are lacking, and those will be examined deeper together with the remaining two compounds, which are reported by sources to have metallic properties.

Combining the compounds associated with the cells for which the literature columns have been marked gray leads to the selection of compounds collected in Table 3.17. For these compounds, either no sources have been found at the time of writing, or they were reported as not having at least a 60 % spin-polarized DOS at the Fermi energy. Subsequently, the corresponding DOS polarizations at the Fermi level were computed using the FLAPW method-based code FLEUR.

The values of P_S have been computed using the DOS values corresponding to the energy value on or the first value above the Fermi energy itself. This choice is necessary as the Fermi energy is not always an element of the computed discrete energy scale. From the DOS, computed using FLEUR¹ the spin-polarization at the Fermi level has been calculated using equation (3.5).

¹ The computations presented in this section of the results were performed using the FLEUR git commit hash e9d2b5ad19c1e89fe0f6b820b76700cd18663cf9 compiled and executed on the supercomputer JURECA-DC.

All of the presented validation calculations have been published within a separate AiiDA database in [324]. All the calculation and convergence parameters used to compute the DOS data necessary to compile the following results have been left as the FLEUR suggested defaults. However, it was chosen that the SCF calculation is performed on a grid of $30 \times 30 \times 30$ k-points and the DOS would be determined on a grid of $50 \times 50 \times 50$ k-points using the GGA-based Perdew–Burke–Ernzerhof (PBE) [325] XC functional. The Gaussian smearing used to calculate the DOS, which was subsequently used to compute P_S , was set to $\sigma = \frac{0.005}{3}$ Ha. For all the presented calculations, spin-orbit interaction has been considered within the FLAPW calculation. Due to the consideration of spin-orbit interaction, no perfect half-metal will appear during the computation—with perfect 100 % spin-polarization of the DOS at the Fermi energy. This is because spin-orbit interaction causes the spin not to be a good quantum number [326] anymore. Therefore, spin-orbit interaction will cause states to be induced within the half-metallic gap as a result of states above and below the gap being coupled. This results in a lowered spin polarization—at the Fermi level—compared to a calculation that neglects this particular interaction. [89] Beyond that, the smearing discretization can introduce an additional error regarding the computation of the spin-polarization of the DOS at the Fermi level. Hence, in the following discussion, which included the FLAPW validation results, if $P_S \geq 75$ % the corresponding compound is referred to as nearly half-metallic, while for $P_S \geq 90$ % the compound is exhibiting factual half-metallic properties and is hence referred to as half-metal.

k-points Gird and
FLEUR
Convergence
Parameters

From Table 3.17 it can be seen that three regular $L2_1$ Heusler compounds (Co_2HfIn , Co_2ScSn , and Mn_2TaGe) have been found to be nearly half-metallic in *ab initio* (FLAPW) simulations, which was—after no literature regarding the near half-metallicity of these compounds has been found at the time of writing—prior unknown. Furthermore, the transition metal alloys Co_2VZn and Co_2NbZn display practically half-metal behavior. The correct prediction of the factual half-metallic transition metal elements is remarkable, as the data set did not contain Heusler alloys entirely composed of 3d transition metal elements. Therefore, this successful prediction represents a clear OOS prediction. The corresponding DOS is shown in the following as Co_2VZn represents the compound with the most significant FLAPW-validated DOS spin-polarization at the Fermi level. This depicted DOS serves as an exemplary successfully predicted factual half—metal, using the presented ML-based materials screening approach. All other compounds from Table 3.17 do not fulfill either the set criterion to classify as true positive classification, near half-metallic or even half-metallic, even though, with 56 % DOS spin-polarization at the Fermi level, four compounds are close to the set classification threshold of 60 %.

Nearly HM
Compounds
Without
Literature
Sources Found

The not converging FLAPW calculation in the case of Mn_2WAl has been investigated. The reason the DFT calculation did not finish is likely due to a faulty input structure. Using the FLEUR input generator, the individual atoms got muffin-tin radii assigned by geometrical consideration using the unit cell. Here, the Mn muffin-tin sphere turns out to be far too large compared to the W sphere, which leads to the conclusion that the entire structural setup is probably corrupted due to an incorrect input structure.

Failed Mn_2WAl
Computation

Summarizing the results, it is easy to see that from the predictions highlighted in the previous tables, 49 materials were correctly predicted to have spin-polarization of the DOS at the Fermi level of above 60 %, many of which are half-metallic, by which half-metallicity was known for 7 materials by applying the Slater-Pauling behavior alone. For 12 compounds, it could be determined that they were predicted to have spin-polarized DOS to the same degree but were, in fact, false positive predictions. This corresponds to a precision of approximately 80 %, not considering the case of Mn_2WAl .

Summary of
Prediction
Precision

Compound	Symmetry	Prediction Class	FLAPW Validation Spin Pol. P_S	Hit/Miss (Color) Novelty of nearly HM	Model Version
		Classical	Heusler Alloys		
Co ₂ HfIn	L2 ₁	$P_S \geq 60\%$	83 %	Yes	1
Co ₂ NbSn	L2 ₁	$P_S \geq 60\%$	19 %	No	3
Co ₂ ScSn	L2 ₁	$P_S \geq 60\%$	76 %	Yes	3
Fe ₂ CoGe	L2 ₁	$P_S \geq 60\%$	43 %	No	3
Ir ₂ FeGa	L2 ₁	$P_S \geq 60\%$	56 %	No	2
Ir ₂ TcTi	L2 ₁	$P_S \geq 60\%$	0 %	No	2
Mn ₂ CuGe	XA	$P_S \geq 60\%$	38 %	No	3
Mn ₂ CuSb	XA	$P_S \geq 60\%$	29 %	No	3
Mn ₂ TaGe	L2 ₁	$P_S \geq 60\%$	83 %	Yes	2 & 3
Mn ₂ WAl	L2 ₁	$P_S \geq 60\%$	DNF		2 & 3
Ni ₂ MnSn	L2 ₁	$P_S \geq 60\%$	10 %	No	1
Rh ₂ FeSn	L2 ₁	$P_S \geq 60\%$	36 %	No	1
Ru ₂ FeGe	L2 ₁	$P_S \geq 60\%$	58 %	No	1 & 2 & 3
Ru ₂ FeSi	L2 ₁	$P_S \geq 60\%$	56 %	No	2
Ti ₂ CoIr	L2 ₁	$P_S \geq 60\%$	56 %	No	3
Ti ₂ ZnAl	XA	$P_S \geq 60\%$	0 %	No	3
		Transition Metal	Heusler Alloys		
Co ₂ NbZn	L2 ₁	$P_S \geq 60\%$	89 %	Known HM by PW [12]	2 & 3
Co ₂ VZn	L2 ₁	$P_S \geq 60\%$	93 %	Known HM by PW [12]	2 & 3

Table 3.17: Combined predictions of the different model iterations for $P_S \geq 60\%$ compounds from the Tables 3.14, 3.15, and 3.16. Also including two transition metal Heusler-like alloys predicted to have a highly spin-polarized DOS at the Fermi energy by multiple models. However, those are, in principle, OOS predictions. DNF is short for the circumstance that the FLAPW calculation did not finish for some reason. Hence, as sources for Mn₂WAl were not available at the time of writing, no decision can be made whether the prediction was right or wrong.

In the following, the DOS of the factual half-metallic 3d transition metal L2₁ alloy Co₂VZn is shown in Figure 3.26. To generate this particular DOS depiction, the smearing was decreased to $\sigma = 10^{-9}$ Ha.

Even though L2₁ Co₂VZn does not achieve a perfect half-metallic P_S value of 1, it is evident that this compound practically represents a half-metal in the presented FLAPW examination, despite being predicted to be metallic by other sources. Additionally, in the experiment, additional effects can cause a material not to represent a perfect half-metal, even if simulations suggest this particular property. Such effects include e.g. phonon scattering effects, potential fluctuations, and temperature effects, which cause states to shift into the minority spin gap of the DOS. As seen in the caption of Figure 3.26, Co₂VZn displays a total magnetic moment of $1.07 \mu_B$ which is situated above the integer value of $1 \mu_B$.

Hence, the magnetic moment deviates from an integer value slightly above the additional error margin of $0.05 \mu_B$. As mentioned earlier, the margin was introduced when the data set was screened for compounds exhibiting Slater-Pauling behavior. This observation highlights the importance of complementary methods in materials screening processes and applications. The very small DOS in the minority spin channel at the Fermi level visible in Figure 3.26 just below E_F might appear a bit larger in the experiment due to effects such as e.g. phonon scattering or thermal excitation.

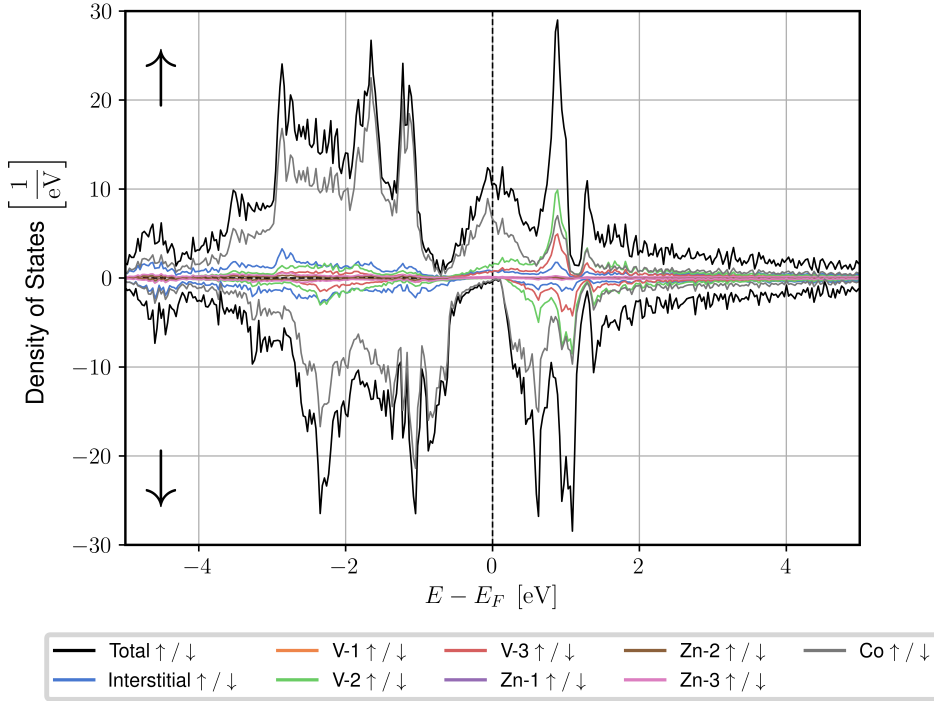


Figure 3.26: Spin resolved DOS of $L2_1$ Co_2VZn , generated using the FLAPW code FLEUR. The total DOS, the interstitial DOS, and the projected atomic DOS contributions are shown. The structure on which this DOS computation is based was obtained from [327], where the compound is described as a metallic ferrimagnet with a total magnetic moment of $1.07 \mu_B$. The very small DOS in the minority spin channel is clearly visible at the Fermi level.

Explainable Artificial Intelligence Analysis

After discussing the predictions of the different model iterations, understanding the impacting features behind the model's predictions is the next logical step. An overview of all features used in the training of the 4th model iteration and their corresponding impacts on the model's prediction on the training database is given by the SHAP summary plot in Figure 3.27.

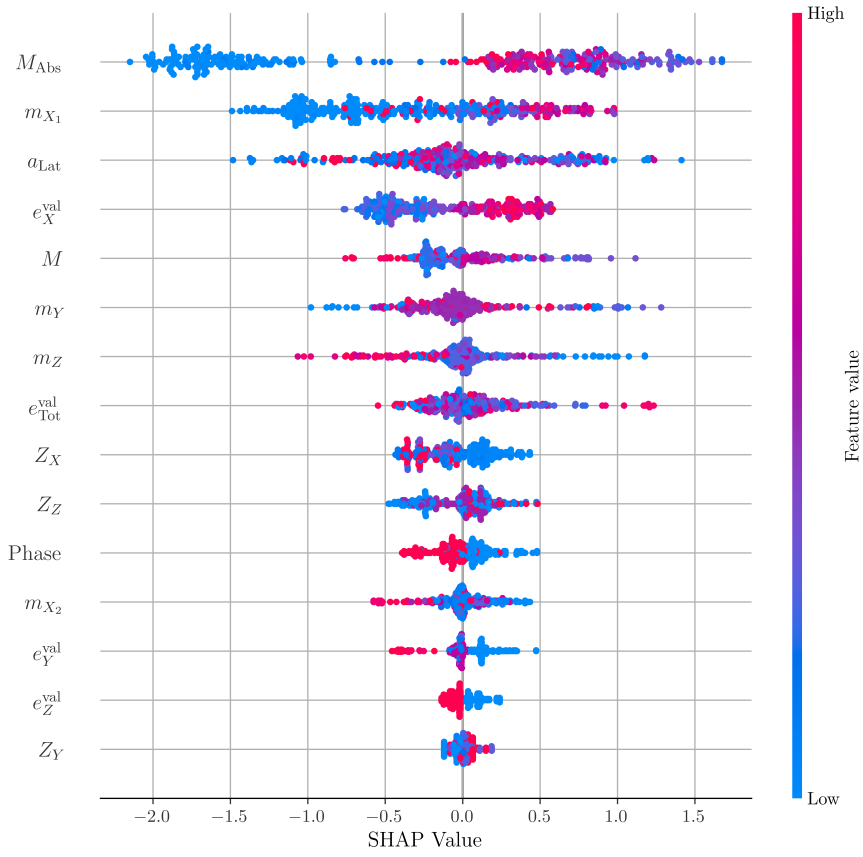


Figure 3.27: SHAP summary plot including all features used to train the 4th model iteration in descending order concerning the largest accumulated magnitude of impact on the prediction per feature. Each impact for every feature on a single prediction of the training data is shown. Negative SHAP values indicate that the feature's impact contributes to a predicted "low" DOS spin-polarization at the Fermi level. In contrast, positive SHAP values indicate a feature impact towards predicting a "high" P_S . Compared to the summary plot from Figure 3.14, the individual points have not been smoothened to a distribution-like depiction.

From Figure 3.27, it can be seen that a couple of features have a comparable impact magnitude. More precise, it appears that, besides the most impactful feature, which is the sum of absolute magnetic moments M_{Abs} , the features can be grouped by impact magnitude into the following groups, in decreasing order:

1. m_{X_1} and the lattice constant a_{Lat}
2. e_X^{val} , M , m_Y , m_Z and $e_{\text{Tot}}^{\text{val}}$
3. Z_X , Z_Z , Phase, m_{X_2} and e_Y^{val}
4. e_Z^{val} and Z_Y

Grouping by
Impact
Magnitude

The features within these subgroups represent a mix of magnetic, structural, and elemental composition-specific properties. While the magnetic descriptors appear as the most impactful, it cannot be concluded that either one of the other feature types is less significant. However, an observation from the development, as this project progressed, is that the atomic numbers went lower in the order of the SHAP summary plots, as the valence electron numbers were introduced to the model, compared to earlier model iterations. Of course, a specific atomic number corresponds to a single number of valence electrons. However, this relation is not fulfilled the other way around. Hence, the information gain, which the atomic numbers provided to the model to constitute the overall prediction, seems to be decreased compared to the previous versions. This could conclude that the half-metallicity classification depends more on the number of valence electrons associated with an atom at a given site than the exact element situated at this given site. This observation would be in line with the known Slater-Pauling behavior. It is no surprise to see from Figure 3.27 that low values of the sum of absolute magnetic moments have an impact that drives the prediction towards the “low” spin-polarization class. Of course, the lowest number this feature can take is 0.

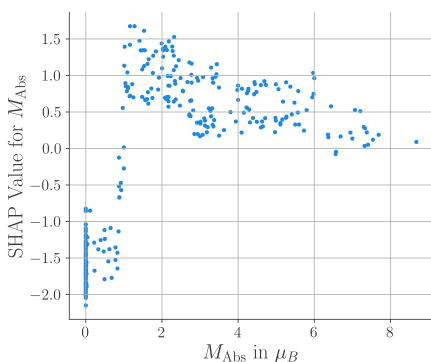
Comparing
Descriptor Types

Non-Magnetic
Materials in
Summary Plot

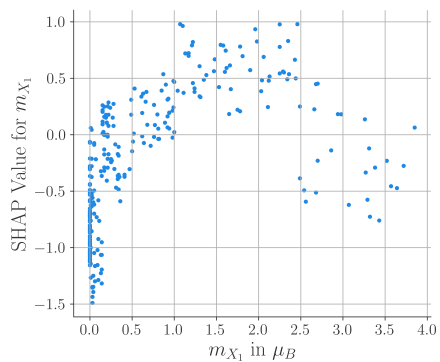
As half-metallicity is an inherently magnetic phenomenon, it is intuitively clear that the vanishing sum of absolute magnetic moments constitutes the absence of half-metallicity. A similar reasoning can be applied to the SHAP values distribution for the magnetic moment m_{X_1} . Of course, X being a representative of the transition metal group, a low m_{X_1} can be associated with a weak magnetism in the whole compound. However, the observation that m_{X_2} has a significantly decreased impact on the model's prediction, compared to the impact of the magnetic moment m_{X_1} , can be explained by a convention present in the training database—and hence also enforced in the screening data set—which defines the axis of the magnetic moment m_{X_1} as the reference axis. However, it is worth mentioning that the relation between feature value and SHAP value is inverted for m_{X_2} as can be seen in Figure 3.27. The impact of the structural lattice constant on the classification cannot be clearly understood by the depiction in the SHAP summary plot. Furthermore, an increasing number of valence electrons associated with site X impacts the model's prediction towards the “high” spin-polarization class. The large magnitude of impact can easily be understood, as the X site constitutes the majority of valence electrons towards the system's total number of valence electrons. However, this relation is less clear for the total number of valence electrons. The magnetic moment associated with the Z site typically ranges between $-0.25 \mu_B$ and $0.75 \mu_B$, while most of the moments are close to $0 \mu_B$ and exhibits an impact on the model's prediction, which shifts the prediction towards a “low” spin-polarized state as m_Z increases. For Z_X , m_{X_2} , e_Y^{val} and e_Z^{val} the relation, that increasing feature values, causes the model to predict the “low” spin-polarization class, also holds. The phase as binary variable slightly favors a “high” class prediction for XA alloys, while the impact on the model prediction favors a “low” polarization prediction for L2_1 compounds.

Importance of X
Site

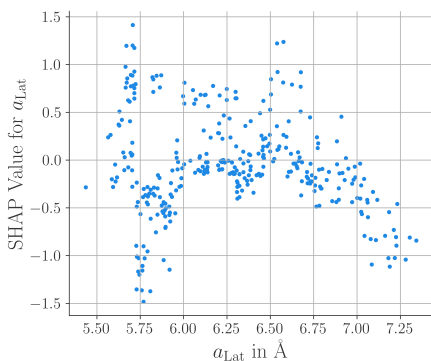
In Figure 3.28 depictions showing the relations between the features M_{Abs} , m_{X_1} , lattice constant, and $e_{\text{Tot}}^{\text{val}}$ and the corresponding obtained SHAP values are shown in detail. These depictions allow a more thorough explanation than the rather macroscopic summary plot.



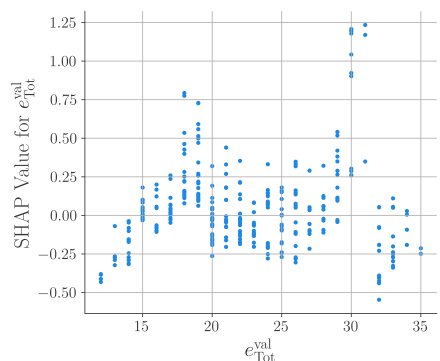
(a) SHAP value and feature relation for M_{Abs} .



(b) SHAP value and feature relation for m_{X_1} .



(c) SHAP value and feature relation for lattice constant.



(d) SHAP value and feature relation for the total number of valence electrons.

Figure 3.28: Overview of selected features (explicitly: M_{Abs} , m_{X_1} , lattice constant, and total number of valence electrons) and their relation to the corresponding SHAP values.

From Figure 3.28a the physical intuition expressed before, that vanishing sums of absolute magnetic moments benefit a classification as a compound with a “low” spin-polarization of the DOS at the Fermi level. Increasing values of M_{Abs} cause an increasing SHAP value until a value of about $M_{\text{Abs}} = 1 \mu_B$ from which the SHAP value continues to decrease. Observing this pattern can be compared to the proportionality observed in Figure 3.27 for m_Z . An increasing overall magnetic moment, contributed to by m_Z , possibly causes a shift in the DOS, which prevents half-metallicity from arising. Examining Figure 3.28b for the magnetic moment m_{X_1} the SHAP value increases with increasing moment until about $m_{X_1} = 2 \mu_B$ and then begins to decline again. It can be seen that m_{X_1} has a similar relation to the corresponding SHAP values as M_{Abs} for small values of features. This is a consequence of the convention applied, that the axis of m_{X_1} is chosen as the reference axis for the compound and hence, by definition, non-negative.

For the lattice constant, as seen in Figure 3.28c, there is a complex relation between the lattice constant and the impact on the model's prediction. Very low lattice constants favor a highly spin-polarized DOS at the Fermi level. Beyond that, in the middle range, a few predictions are impacted towards the "high" spin-polarization by the lattice constant, while for many compound predictions, the impact is nearly neutral. There is the range at both the lower and the higher end, where the lattice constant contributes towards predicting a low spin-polarized DOS at the Fermi energy. Observations from Figure 3.28d could include that for a total number of 18 and 19 valence electron in the compound's unit cell, the prediction is shifted for all the data points with these total valence electron numbers towards a highly spin-polarized DOS. Furthermore, an increase in the SHAP values can be observed starting from a total number of valence electrons of 27 up to 31. These values include the values for the total number of valence electrons of 18 and 28, which are known from the Slater-Pauling behavior of inverse Heusler alloys. However, the value of 24, which is established from the Slater-Pauling behavior for both $L2_1$ and XA phases, is not clearly visible in this depiction. Examining the total magnetic moment of the compound's unit cells and the resulting SHAP values from Figure 3.29, it can be concluded that for the integer magnetic moments $-1 \mu_B$, $1 \mu_B$, $2 \mu_B$, $3 \mu_B$ and $4 \mu_B$ the SHAP value spikes to some extent, which indicates the prediction to be impacted towards a highly spin-polarized DOS at the Fermi energy for these particular compounds, based on the total magnetic moment's integer value. This is another indication of the known Slater-Pauling behavior. However, starting from a magnetic moment of $5 \mu_B$, the prediction impact of the total magnetic moment changes to shift towards a low spin-polarization, which is in line with the observations from the features M_{Abs} and m_Z impacts.

Lattice Constant Impact

XAI Depicts Partly Slater-Pauling-Like Behavior

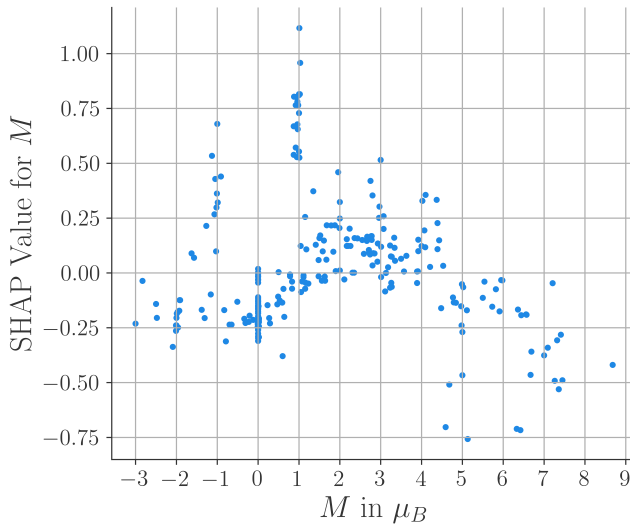


Figure 3.29: Depiction of the relation between the total magnetic moment M and the corresponding SHAP value.

Furthermore, while m_Y has a complex relation to the corresponding SHAP value, as can be seen by Figure 3.27, the value and its relation to the SHAP value is illustrated in Figure 3.30 alongside with the e_X^{val} feature.

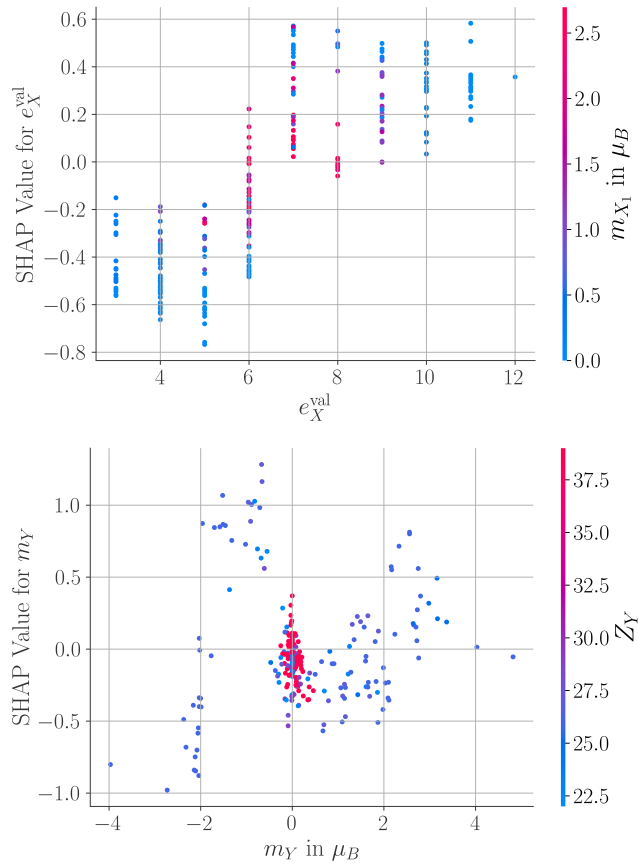


Figure 3.30: Relation between the features e_X^{val} (upper) and m_Y (lower) and the corresponding SHAP values computed on the training data set together with the relation to another feature value, each encoded using a color bar.

From Figure 3.28d multiple conclusions can be drawn. First, it is clear, that valence electron numbers at the X site contribute to a predicted lower spin-polarization of the DOS at the Fermi level for values below 6. Coincidentally, transition metals with such valence electron numbers typically do not exhibit strong magnetic properties, as can be seen by the coloring introduced in the figure, which depicts the magnetic moment at one of the X sites of the unit cell. For valence electron numbers at the X site above or equal 7, the feature impact suggests a highly spin-polarized DOS at the Fermi level.

Also, the relation between m_Y and the corresponding SHAP value can be observed from Figure 3.30. The very low feature values ($m_Y < -2 \mu_B$) favor—in the model's prediction—a low spin-polarization for the DOS at the Fermi level is suggested. For feature values in the ranges $-2 < m_Y < -0.5$ and $2 < m_Y < 4$, it appears that a highly spin-polarized DOS at the Fermi level would be favored in the model's prediction. Especially in the range around a vanishing $m_Y \mu_B$ is densely populated and contributes barely any impact on the final prediction whatsoever. From the introduced color bar, it is apparent that many of those compounds located in the range of a vanishing m_Y represent the Y site as being occupied by $4d$ elements, which typically exhibit less strong magnetic moments than $3d$ elements.

Besides all the depictions showing more macroscopic explanations for the model's predictions, local explanations for selected compounds from the training set are included within Figure 3.31. Local explanations aim to provide insights into why an individual prediction was made rather than explaining the impact trend a feature has across multiple predictions. These local explanations are included in this discussion to demonstrate, that XAI tools can also be used to understand individual predictions. For the local explanation, a red arrow denotes a feature that impacts a prediction of a highly spin-polarized DOS at the Fermi level for this particular compound. Each blue arrow indicates a predictive feature impact towards a low spin-polarization for the given compound. As previously explained, the prediction is shifted from the base value (the expectation value of the model) towards the prediction by contributions of each feature impact.

Local
Explanations

For the $L2_1$ compound Mn_2YSi , for which the local explanation is shown in Figure 3.31a the impact of the individual features is visualized. There, it can be seen that the total magnetic moment is given by an integer value. However, the total number of valence electrons is 21, which does not satisfy the typical Slater-Pauling behavior. Mn_2YSi is part of the training data set with a DOS spin-polarization at the Fermi energy of 91 %, which does not represent a perfect half-metallic polarization but already is close to a factual half-metal. The prediction having the closest to average probability—according to the ML model—of all training compounds has been included to demonstrate the very different impacts of features like *e.g.* m_Z which are very similar for $L2_1$ Mn_2YSi and XA Ni_2CoAl but still have a very different impact on the overall prediction. The compound Ta_2TiSi shown in Figure 3.31c is classified as a compound having a low spin-polarized DOS at the Fermi level, which is obvious by the fact that XA Ta_2TiSi is entirely non-magnetic.

Mn_2YSi

Ni_2CoAl

Ta_2TiSi

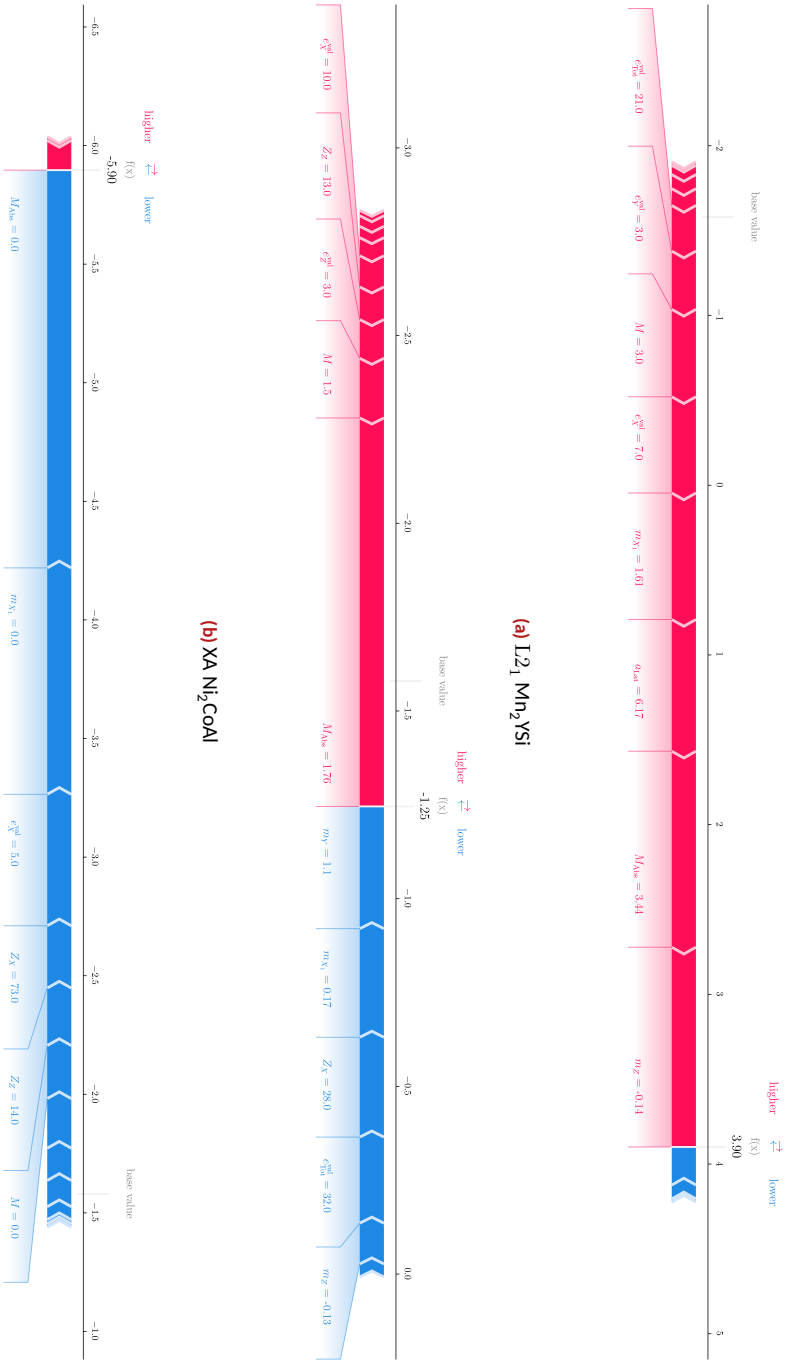


Figure 3.31: Local explanations for the selected compounds ($L2_1 \text{ Mn}_2\text{YSi}$, $\text{XA Ni}_2\text{CoAl}$, and $\text{XA Ta}_2\text{TiSi}$) from the training set. The compounds were selected by the probability of the individual prediction given by the model. The compound with the highest probability to have a highly spin-polarized DOS at the Fermi level (Mn_2YSi), the probability closest to the average probability (Ni_2CoAl), and the lowest probability (Ta_2TiSi). The compounds, with the highest and lowest probability to have a highly spin-polarized DOS around the Fermi level, have been classified correctly.

3.3 Ultrathin Films of (001) oriented 3d Transition Metal Layers on fcc Noble Metal Substrates

Ultrathin film systems, which consist of multiple layers including such that exhibit magnetic properties, are commonly referred to as magnetic multilayers. Within magnetic multilayers, atomic magnetism is typically more prevalent than in bulk materials. The reason is that the surface atoms have fewer neighbors; hence, their electronic and magnetic configuration is closer to the corresponding atomic properties. Additionally, the magnetism in magnetic multilayer systems is tunable by various structural properties of the film system and environmental conditions, which allows complex magnetic configurations to form. Hence, magnetic multilayers are capable of hosting stable nanoscale Skyrmions (the scale refers to the radius of the Skyrmion being around 1 nm and hence exceeding the scale of atomic radii which typically have an order of magnitude of 1 Å) at room temperature. [328] While Skyrmions also appear in bulk materials [329], layered magnetic film systems—from a materials design perspective—can be tuned to create a very sophisticated environment for the long-term stabilization of Skyrmions at operating temperatures of spintronics devices. Furthermore, the importance of surface effects in film systems relative to theoretical bulk systems sets layered magnetic film systems apart from bulk systems. Such are typically modeled without any surface. It is apparent that due to the ratio of surface atoms being significantly larger in film systems than in real 3-dimensional crystal lattices, surface effects contribute to stronger magnetic multilayer systems properties. [330] A special consequence of these relevant surface effects is that film systems tend to be sensitive to small external magnetic fields, which can cause significant property changes in the magnetic multilayer systems. [330, 331] Some aspects in which magnetic multilayer systems can be tuned—which are investigated in this study—are listed in the following:

- Layer and film thickness [332, 333]
- Layer composition [332]
- Layer order [332]
- Choice of FM or AFM order [330, 332, 334]

Beyond the tunable film properties, which are examined within this project, there are also environmental conditions—which are more relevant experimental settings—impacting the system's properties, as listed in the following:

- Film growth conditions [335]
- Operating temperature [331]

External fields also represent a tunable property of a film system's environment, which could be investigated using DFT. [331] However, this project is restricted to evaluating structure-related tunable film parameters. The number of tunable system properties yields a large space of possible structures and operating environments, which is challenging to explore systematically. Examining the layer order and composition dimensions alone constitutes a structural phase space that grows exponentially with the number of layers. In this growth, the number of unique layer compositions represents the basis. To engage in systematic studies of magnetic multilayer systems of this scale, dedicated high-throughput frameworks, strongly parallelized *ab initio* codes, and (HPC) infrastructures and resources are required, which (in part) were not available until recently.

Magnetism in Thin-Films

Magnetic Multilayers as Nanoscale Skyrmion Host Systems

Tunability of Magnetic Multilayers

Surface Effects

Recent Technological Advancements

Room temperature stable Skyrmions are known to emerge in film layer combinations of e.g. Fe, Co, Ir, and Pt. [332, 336, 337] Combining this knowledge with the FM and AFM properties being inherent to a significant proportion of the $3d$ transition metal group constituted the motivation to conduct a systematic high-throughput *ab initio* study. The magnetic properties of films consisting of layers of $3d$ transition metals on fcc noble metal substrates are discussed in this section. The analysis of the magnetic properties of the magnetic multilayer systems and the use of ML within this high-throughput study are discussed in detail in the following. The use of predictive ML to predict improved *ab initio* inputs showed the capability to improve the convergence rates and reduce computing costs in high-throughput applications.

3.3.1 Calculation & Film Setup

Before a high-throughput *ab initio* study can be performed on a large set of structures, it is necessary to set up the examined structures correctly. For some high-throughput projects, setting up the structures might be as easy as downloading and loading crystal structure information files using the AiiDA framework. [9, 10] However, in this project, a dedicated workflow within the AiiDA-FLEUR [11, 92] plugin has been used to set up the layered film structures. The scripts used to set up the structures, submit the FLAPW first-principles calculations, and perform the data analysis are published in [338]. Also, the AiiDA database containing the computed film systems is published in [97]. The structure and calculation setup are discussed in detail in the following.

Choice of Film Constituents

It is known that layered film systems containing the fcc noble metals Ir and Pt alongside the FM $3d$ transition metals Fe and Co are capable of hosting Skyrmions. [332, 336, 337] This suggests that the combination of layers consisting of elements from the fcc noble metal family (containing Rh, Pd, Ag, Ir, Pt, and Au) and layers containing $3d$ transition metals (Sc, Ti, V, Cr, Mn, Fe, Co, Ni, Cu, and Zn) represents a promising subclass, of the structural phase space of magnetic multilayers, to exhibit spintronics applications relevant properties. It is well known for $3d$ transition metal elements to exhibit magnetic properties. [339] However, as film systems, surface effects play an important role. These effects can increase the observed atomic magnetic moment in film systems at layers close to a film surface. It has been observed that also $4d$ and $5d$ transition metal elements can exhibit significant magnetic moments in ultrathin films. [339] Still, considering the different elements in each row of the d block of the periodic table, the $3d$ transition metals are most likely to exhibit stable magnetism. [339] Hence, the decision was made to restrict the selection of magnetic layer elements to the $3d$ transition metals during this project. The fcc noble metals are used as substrate material.

It could be argued that the noble metals Os and Ru also represent suitable substrate materials for layered magnetic films. However, both elements occur naturally in a hexagonal close-packed (hcp) configuration. This would require a different structural setup and configuration. Calculation results based on Os and Ru substrates would be more difficult to compare to the films with fcc-based substrates. For this reason, this study is restricted to fcc-based noble metal substrate film systems.

Film Construction

The film setup follows the Create-Magnetic-Film workflow described previously in section 2.4.2, publicly available within the AiiDA-FLEUR plugin. [11] Within this workflow, at first, EOS calculations are performed to compute the substrate bulk lattice constants. This is especially important, as the substrate lattice constant and substrate structure also determine the in-plane position coordinates for the magnetic layers added to the substrate. Beyond the EOS computation, the substrate layers are kept fixed for the relaxation steps performed within the Create-Magnetic-Film workflow. Therefore, only the magnetic layers are allowed to relax along the direction of the film's out-of-plane axis (in this case, this is chosen to be the z -axis). The results for the substrate fcc lattice constants are shown in Table 3.18. The accuracy of the lattice constants is primarily determined by the EOS scaling resolution (in this case, 5 % of the initial bond length guess computed using the Materials Project database [94]). Hence, the EOS-originated lattice constants shown in Table 3.18 are rounded to the second decimal digit.

Substrate
Defines In-plane
Film Structure

Magnetic Layer
Relaxation Along
 z -axis

Substrate Element	EOS Minimum [Å]	GGA FLAPW Reference [Å]	Experimental Values [Å]
Rhodium	3.83	3.83 [340]	3.80 [341]
Palladium	3.94	3.94 [340]	3.89 [342]
Silver	4.14	4.15 [340]	4.09 [342]
Iridium	3.87	3.87 [340]	3.83 [343]
Platinum	3.97	3.97 [340]	3.92 [344]
Gold	4.15	4.16 [340]	4.08 [342]

Table 3.18: Overview of the substrate elements EOS resulted lattice constants compared to a GGA (PBE [325]) FLAPW reference [340] and experimental values.

From Table 3.18, a slight systematic overestimation of the EOS-originated, as well as the GGA FLAPW reference lattice constants can be seen in comparison to the experimental values, which is commonly known for GGA XC-based *ab initio* results. Furthermore, it can be seen that there is an excellent agreement between the EOS resulting lattice constants, acquired using the Create-Magnetic-Film workflow, and the GGA FLAPW reference values, which are accessible in the supplemental material of [340]. After the EOS lattice constant computation has been performed once for every substrate, within the Create-Magnetic-Film workflow, the acquired lattice constants have been reused for the submission of the Create-Magnetic-Film workflow for other film systems.

Lattice
Constants Agree
With GGA
Reference

To ensure reproducibility, the bond length estimates, computed using the bond length data from the Materials Project [94], for the combinations of all magnetic and substrate elements are provided in appendix B, together with an equation directly relating the bond length estimates to the ILDs computed as an initial guess. This is necessary, as the computed bond length estimates could change over time as the Materials Project database will be extended.

To efficiently use computational resources, considerations were made to optimize the structure setup, with the goal in mind to increase the number of structural symmetry operations within the film systems. This led to the decision to compute symmetrical films instead of film systems with magnetic layers added to only one substrate side. The magnetic layers are added along the fcc $[0\ 0\ 1]$ direction. Using this setup, the number of equivalent atoms within the structure is increased compared to non-symmetric film structures within the assumption that the magnetic layers on either side share the same magnetic behavior and orientation, independent of each other.

Choice for
Symmetric Film
Setup

The middle substrate layer is placed at $z = 0 \text{ \AA}$ in this setup. The goal of setting up symmetrical film systems for computational efficiency also concludes that the substrate layer count has to be given by an odd number of layers, as this choice conserves the symmetry operations of z -reflection and inversion.

Based on previous calculations, it was known that the ILDs of the magnetic layers only deviated by lower than 5 % computed using three substrate layers compared to the relaxed ILDs calculated using a film with a substrate thickness of 15 layers. However, to ensure that the results would be comparable to films with a larger substrate layer count, it was decided to continue with five substrate layers. This eliminates the necessity to recompute the film's structural relaxation for systems with thicker substrates. Hence, the assumption that the magnetic layers on both sides of the substrate do not interfere with the relaxation of each other notably—given the chosen substrate thickness—also holds according to these preliminary calculations.

Knowing that the film structure is entirely defined using the ILD guesses based on the bond length estimation and the EOS fcc substrate lattice constant and considering that this project's scope is to examine magnetic films, it is still necessary to determine an appropriate magnetic initialization for the magnetic layers. At first, the initial magnetic moment for each magnetic layer was set to be given by $1 \mu_B$. This choice was motivated by the fact that the intention was to magnetize each magnetic layer to the same degree initially and with Sc having only a single unpaired electron per atom, the only integer atomic moment that could hence be chosen is $1 \mu_B$. With the outlined setup, the number of film systems resulting from n_{Subst} different substrate elements and n_{Mag} magnetic elements occupying a given number of n_{Lay} magnetic layers on the substrate is provided by the expression in equation (3.7).

$$n_{\text{Syst}} = n_{\text{Subst}} (n_{\text{Mag}})^{n_{\text{Lay}}} \quad (3.7)$$

Due to the exponential scaling of equation (3.7) with the number of layers, the structural phase space is practically unlimited when it comes to possible magnetic layer orderings and elemental combinations. Considering the computational resources available during this project, it was determined that three magnetic layers on each substrate side are reasonable to compute within the given resources and time constraints. This results in 6000 possible film combinations by the combinatorics of the problem. This potential data set size exceeds the aforementioned data projects in this thesis. Furthermore, the opportunity was taken to compute the corresponding films with 2 and 1 magnetic layers on each side. This essentially translates to the outer magnetic layer site being unoccupied or the exterior and the middle magnetic layer site being unoccupied compared to the film systems having three magnetic layers on each substrate side. This increases the number of combinations according to equation (3.8) to 6660 systems.

$$n_{\text{Syst}} = n_{\text{Subst}} \sum_{n_{\text{Lay}}=1}^3 (n_{\text{Mag}})^{n_{\text{Lay}}} \quad (3.8)$$

A schematic depiction of the resulting film structures and quantity naming conventions is given in figure 3.32. As already discussed in section 2.4.2 the initial guess for the ILD of the magnetic layer on the outside of the film is scaled by a factor of 0.95 as it is well known, that the surface layer of a film tends to adhere to the sole neighboring layer slightly more than a layer which has two adjacent layers.

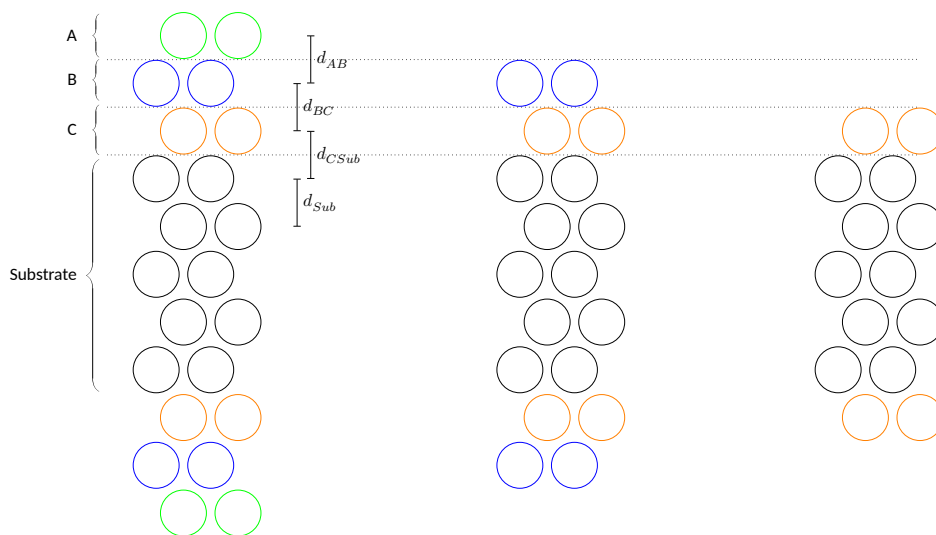


Figure 3.32: Schematic depiction of (001) oriented film structures resulting from the structural setup workflow, including naming conventions. The magnetic A, B, and C layers are colored. The individual ILDs between the layers d_{XY} are labeled within the visualization. Figure similar to [55].

Choice of Simulation & Workflow Parameters

All film calculations have been performed using the GGA type PBE [325] XC functional using the FLAPW code FLEUR² on a grid of $20 \times 20 \times 1$ k-points using a $K_{\max}$ cutoff of 3.8. Both the k-point grid as well as the $K_{\max}$ cutoff have been validated regarding their impact on the relaxed ILD results compared to computationally more intensive settings such as $50 \times 50 \times 1$ and $K_{\max} = 5$ to alter the resulting ILDs by less than 5 %. The charge density SCF mixing has been defensively set to 0.5 %. This relatively small mixing parameter assists with preventing the so-called charge sloshing. Charge sloshing describes relatively long-ranged oscillations of the charge density distributions between SCF iterations. The Kerker mixing scheme can also avoid charge sloshing in film systems. [51, 54] However, the Kerker mixing scheme was tested on a small subgroup of the examined films before the actual high-throughput study and led to lower convergence rates than simply lowering the charge density mixing parameter. Hence, the decision was made to change the mixing parameter rather than decrease the convergence rate by employing the Kerker mixing scheme.

Computational
Setup Details

Overcome
Charge Sloshing

The force mixing parameter relevant to the relaxation was set to 1.7. A relaxation is said to be converged as the largest force acting parallel to the out-of-plane z -axis, on the atoms constituting the magnetic layers, is lower than $5 \cdot 10^{-5} \frac{\text{Ha}}{a_0}$. This force convergence criterion could be considered relatively strict. However, this particular convergence criterion has been chosen to ensure that the structures can be regarded as entirely relaxed.

² The computations performed underlying this section of the results were performed using the FLEUR git commit hash e9d2b5ad19c1e89fe0f6b820b76700cd18663cf9 compiled and executed on the supercomputers JURECA-DC and CLAIK.

While the ILD changes itself would not be remarkably large compared to a less restrictive criterion, already minor deviations in the structure can significantly affect the magnetic properties of the film, which are the main focus within this project. Initially, the force mixing scheme was set to be straight linear mixing. The first relaxation iteration to exhibit a force lower than $0.025 \frac{\text{Ha}}{a_0}$ for each relaxed atomic layer changes the mixing scheme to a BFGS-based scheme for the following relaxation iterations. The individual SCF calculation, which is part of the relaxation workflow, is said to be converged starting reaching a charge density distance below $10^{-3} \frac{\text{m}}{a_0^3}$. A single *ab initio* SCF calculation was given 100 iterations to converge, and a SCF calculation was allowed to be restarted nine times during a Create-Magnetic-Film workflow if the individual SCF calculation failed for a reason which could be fixed by continuing the SCF calculation. A single Create-Magnetic-Film workflow performs at most 70 relaxation steps to relax a film system. The muffin-tin radii of the atoms in the film are first set by the FLEUR input generator based on the crystal structure and, correspondingly, the space available. The radii are scaled using a factor of 0.96 before the first relaxation step to allow the muffin-tins to relax.

3.3.2 Workflow Outputs

From the Create-Magnetic-Film workflow, a plurality of physical output data like *e.g.* structure, magnetic moments, total energy, *etc.* is available. Beyond the physical outputs, additional information about the workflow execution itself is available. This information includes the number of required relaxation steps and the total number of performed SCF iterations. The main outputs of the workflow are briefly discussed in the following.

Film Structures

The relaxed film structure is stored within a node of the corresponding AiiDA database at the end of each successful Create-Magnetic-Film workflow. Using the previously discussed symmetric film setup, which is also documented in a dedicated code publication [338], the workflow outputs both the symmetric film and the equivalent non-symmetric film systems. In the latter case, the lower three substrate layers are not replaced with magnetic layers in the setup process as described in section 2.4.2. Hence, the substrate thickness in these non-symmetric film systems is increased by three layers. The two stored structures at the end of a successful workflow are shown in Figure 3.33. Within AiiDA-FLEUR [11], there also exist dedicated routines that are capable of either reducing or increasing the number of substrate layers. Therefore, the relaxed magnetic layers are portable to substrates of different thicknesses by reusing the relaxed ILDs. This portability is part of the reason it was ensured that the ILDs deviation, for increasing substrate layer counts, is lower than 5 % compared to the selected number of substrate layers.

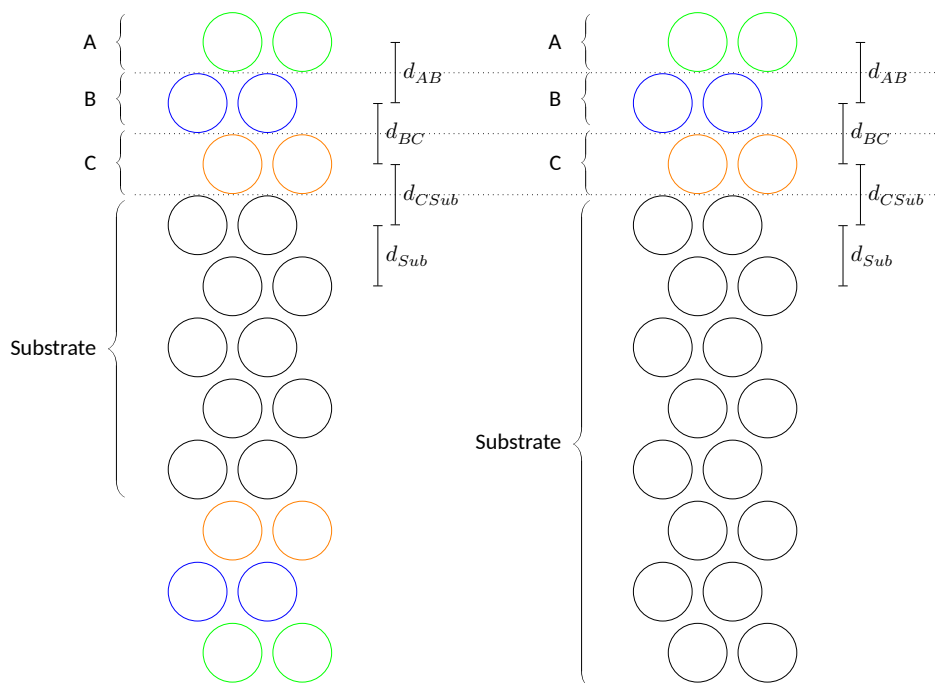


Figure 3.33: Depiction of structural outputs (symmetric and non-symmetric film systems) of the AiiDA-FLEUR [11] Create-Magnetic-Film workflow. The magnetic layers within the films are colored, and the substrate layer atoms are depicted as black circles.

Displacements & Magnetic Properties

Beyond the structure, the Create-Magnetic-Film workflow also collects the positional displacement of the relaxed atoms. This displacement corresponds to the coordinate difference of the initialized positions of the atoms compared to the relaxed positions. Furthermore, the magnetic moment of each nonequivalent atom in the set-up structure is stored alongside the total unit cell moment, which also contains an interstitial moment contribution. This includes the converged magnetic atom's moments of the individual magnetic layers and the induced moments from each substrate layer. Additional information is stored in the workflow outputs, originating from the FLEUR output file, such as the film system's total energy and the Fermi energy.

Displacements

3.3.3 DFT-Integrated Machine Learning

Computing all 6660 structures using the described Create-Magnetic-Film workflow with the discussed calculation parameters and cutoffs, a successful *ab initio*-based relaxation has been performed for 64.8 % of the examined structures. In high-throughput DFT applications, error rates of around 10 % are not uncommon. [93] This concludes that the error rate in the presented high-throughput study is far too high. A naive approach to tackle convergence issues in high-throughput settings like this would be a trial-and-error-based approach in which the calculation parameters, magnetic initialization, and structures are altered. However, this represents a somewhat unstructured and unsystematic approach to improve the convergence rate and hence might include the necessity to use significant amounts of computing time before the tuned calculation parameters lead to additional structures relaxing successfully. To systematically tackle this challenge, an ML-based method, which combines the methodology of batch learning—using converged DFT results—and the described first principles-based workflow. Within this method, which is further denoted as DFT-integrated ML, it was possible to predict both the ILDs and the 3*d* transition metal magnetic layers moments based on the already relaxed *ab initio* results. This process can be performed iteratively as the training data for the ML models increased as soon as more DFT relaxations finished successfully, as shown in Figure 3.34. While the DFT-integrated ML scheme has been applied to predict the ILDs and magnetic moments during this project, this scheme can, in principle, be applied to any quantity which is both input and output of an *ab initio* calculation *i.e.* is optimized through a DFT calculation. Hence, the methodology could also be used to optimize *e.g.* non-collinear magnetic angles or even the charge density itself.

Initial Relaxation
Success Rate
64.8 %

Requirement for
Systematic Input
Optimization
Strategies

Portability to
Other Input
Properties

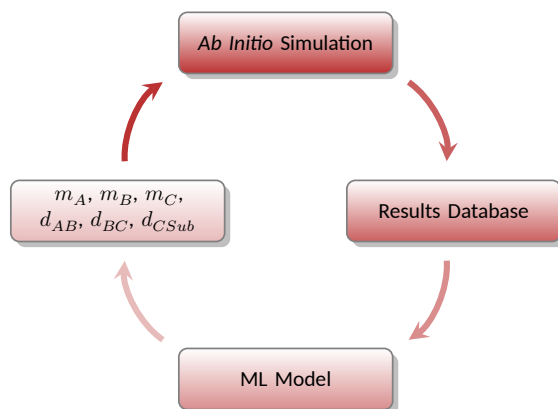


Figure 3.34: Schematic depiction of the iterative DFT-integrated ML workflow in this particular application, representing a form of batch learning in conjunction with *ab initio* methods integration. From the results database, the predicted target quantities and the film's constituents' atomic numbers are used in the model training process.

However, the ability to set ILDs manually without using the Materials Project-originated bond length estimates as initial guesses was not available in the AiiDA-FLEUR plugin and needed to be included. [11] I developed this additional functionality for the AiiDA-FLEUR plugin. While this development occurred, the first batch of inputs containing ML-predicted magnetic moments for the magnetic layers could already be used in FLAPW computations.

Extension of
AiiDA-FLEUR

Within the DFT-integrated ML approach, the sole input features used during this project were the atomic numbers of each magnetic layer and the substrate. In the case of an unoccupied layer corresponding to films with only two or one magnetic layer, the respective atomic number has been chosen to the value zero. This section discusses the methodology and the advantages of DFT-integrated ML in high-throughput applications in detail in the following.

Atomic Numbers
as Sole Input
Features

ML Prerequisites & Modeling Goal

To train an ML model using the DFT-based data acquired by the successful run of AiiDA-FLEUR [11] workflows and also to further analyze the data, the data had to be extracted from the SQL type AiiDA database. An overview of the quantities extracted from the database is shown in Table 3.19.

Data Extraction

Relaxation Output	Last SCF Output	Workflow Meta Information
$d_{AB}, d_{BC}, d_{CSub}, d_{Sub}^{\dagger}$	m_A, m_B, m_C	$n_{Iter}^{Tot SCF}, n_{Force Iter}^{Tot Conv}$
$\Delta d_A, \Delta d_B, \Delta d_C, d_{Film}^{Total}$	$m_{SubA}, m_{SubB}, m_{SubC}$	$n_{Force Iter}^{5 \cdot 10^{-3}}, n_{Force Iter}^{10^{-3}}$
Z_A, Z_B, Z_C, Z_{Sub}	$M_{Cell}^{Tot}, M^{Tot}, M_{Abs}$	$n_{Force Iter}^{5 \cdot 10^{-4}}, n_{Force Iter}^{10^{-4}}$
Number of structural unique atoms	$E_F, E^{Tot}, E_{Val}, \Delta E_{Band}$	$n_{Iter}^{DFT IntML*}$

- [†] EOS result, not relaxed during the relaxation step.
- ^{††} Beyond the global relaxation, convergence criterion of $5 \cdot 10^{-5} \frac{Ha}{a_0}$, also the number of force relaxation steps, required to achieve other force thresholds are included in the extracted data set. The alternative force criteria are denoted in the corresponding superscripts, in units of $\frac{Ha}{a_0}$.
- * Each successful calculation is assigned a number $n_{Iter}^{DFT IntML}$ which indicates how many DFT-integrated ML iterations have been performed before the individual film system converged.

Table 3.19: Overview of quantities extracted from the AiiDA database for data analysis and DFT-integrated ML model training, grouped by the data origin, indicating at which point of the workflow execution and computation the individual quantity has been extracted. The naming convention order for the magnetic moment has been continued on the substrate layer moments. Hence, the magnetic moment of the substrate interface layer is referred to as m_{SubA} , and the magnetic moment of the central substrate layer is referred to as m_{SubC} . In this table, δd_X denotes the deviation of the position of layer X in the relaxed configuration compared to the initial structural setup. M_{Cell}^{Tot} denotes the total moment of a film's unit cell, including interstitial contributions, while M^{Tot} denotes the summation of film's atomic moments. The z directional spatial displacement corresponding to the layer X is denoted as Δd_X .

The data set extracted from the AiiDA database has been published. [338] This is also the case for the generated AiiDA database itself. [97] However, the DFT-integrated ML-based structural and magnetic moment predictions are entirely based on the four atomic numbers Z_A, Z_B, Z_C , and Z_{Sub} as training features. Prior to any modeling presented in this section, a test set was sampled randomly from the extracted data. Subsequently, the model input features have been scaled according to equation (3.2). The test set was set to be the size of 20 % of the entire data set. First, a suitable model had to be selected to train an ML model for the predictive DFT-integrated ML application. Using a similar set of different models and model types, as in the previously discussed projects, the XGBoost model [125] exhibited the best 4-fold CV performance on the randomized remaining 80 % of the data.

Randomly
Sampled Test Set

The CV score during the model selection process has been determined for the mentioned set of models without any hyperparameter optimization. Given that the prediction of both the ILDs (d_{AB} , d_{BC} , and d_{CSub}) and also the magnetic layer moments (m_A , m_B , and m_C) to be used as ML-optimized starting parameters, for an *ab initio*-based calculation, imposes a regression problem, it is necessary to think of a metric suitable to evaluate the models. The MAE has been used as both CV and test metric as this metric returns errors in the unit of the predicted quantity. This allows for the interpretation of the resulting error and the decision of whether the MAE magnitude is acceptable for the predictive modeling process. This is crucial as the MAE prediction error can be compared to the error associated with the presented ILD guessing method based on the Materials Project data. In principle, this results in six individual regression problems to be modeled. However, this task is relatively easy to generalize, given the similarities of the modeling tasks. [338] During the predictive modeling application, a hyperparameter optimization was performed based on the 4-fold CV score calculated within a grid search algorithm. The set of hyperparameters that has been searched is documented in [338]. After the hyperparameter selection and evaluation of the model using a test set, the models were retrained before being used to predict optimized structural and magnetic input properties of the previously not successfully relaxed film systems.

Model Performance & Modeling Results

As outlined previously, a model evaluation is performed before the models are retrained on the entire data set before the predictive procedure. This evaluation also includes calculating a test MAE, determined using the holdout method, based on 20 % of the available data set at that particular DFT-integrated ML iteration. The test MAE values for both the magnetic layer ILD predictions and the predictions for the magnetic moments of the magnetic layer are depicted in Figure 3.35.

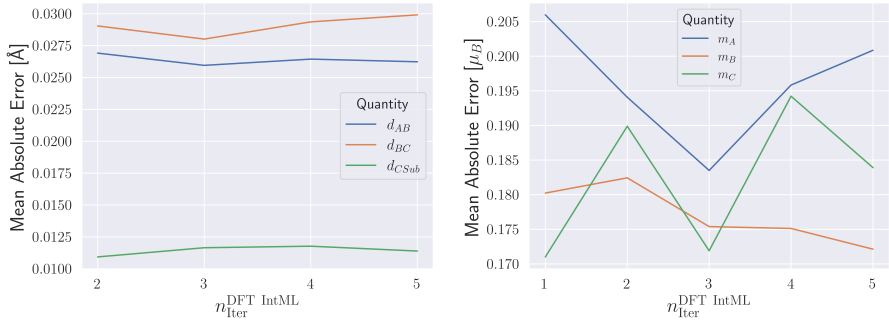


Figure 3.35: Test scores of the models, which were used for ILD (left) and magnetic moment (right) input parameter prediction, based on the previously converged calculations.

From Figure 3.35 it is apparent, that there is no clear trend for both the ILDs MAE and the magnetic moments MAE with increasing data set size. Observing the scale of the magnetic moment test MAE, it is apparent that the errors have comparable sizes for the different layers. This is not as clear the case for the MAE of the ILDs. The quantity d_{CSub} displays the lowest MAE in Figure 3.35, which can be explained by the fact that the substrate always neighbors the C layer and hence, as the impact of the element situated at the B layer on the MAE for the quantity d_{CSub} can be assumed as relatively lower than the impact of the element situated at the C layer and the substrate element, which makes d_{CSub} comparably easier to learn for the models. As the subspace of the substrate elements is lower than that of the magnetic layer elements, this results in a minor variation in the d_{CSub} compared to the other ILDs. A similar argument can be brought up when comparing the MAE of the ILD d_{AB} , as the A layer has only one neighbor and hence less elemental variation in the spatial proximity of the elements separated by the ILD d_{AB} . However, the subspace of occupant elements on layers A and B is still more extensive than the occupant's subspace for the ILD d_{CSub} . Combining both arguments concludes that the MAE for the quantity d_{AB} is found below the MAE of d_{BC} , which has inherently the most considerable variation of elemental occupants in the spatial proximity of the ILD d_{BC} .

No Trend in Prediction Models MAE for Increasing Amounts of Training Data

This section will compare the test errors to errors determined in a posterior analysis using the predictions and quantities extracted from the successful workflows. The posterior discussion is crucial, as the model training in further DFT-integrated ML iterations assumes that the resulting ILDs and magnetic layer moments are independent of the iteration during which the corresponding structures were successfully relaxed. Precisely, simply mixing the different DFT-integrated ML iterations and using the FLAPW output data, as ground truth, for model training assumes implicitly:

Assumptions Associated with DFT-Integrated ML

- All film structures are equally challenging to relax/compute and hence have a comparable modeling complexity across all DFT-integrated ML iterations. [55]
- Each DFT-integrated ML iteration samples the phase space of all possible film structures equally well. [55]
- No input parameters, besides the predicted input quantities, change using the DFT-integrated ML predictions, *i.e.* the other calculation parameters (*e.g.* muffin-tin sphere sizes, cutoffs, *etc.*) are independent of the predicted input quantities. [55]
- The relaxed structure and the corresponding magnetic state represent the sole possible *ab initio* solution.

Knowing that multiple magnetic configurations can result from a DFT calculation, depending on the provided input configuration, for the same compound, the last assumption is untrue. Despite that, it will be shown that this approach can improve magnetic and structural starting points for *ab initio* simulations in the high-throughput scale. Upon examining the test MAE values from Figure 3.35 and comparing them to the MAE values of the guessing approach, shown in Table 3.20, it becomes apparent that the bond length-based structure guesses from Materials Project [94] and the initial magnetic moment guess of $1 \mu_B$ would have been outperformed in terms of accuracy of the suggested inputs by the DFT-integrated ML approach. This evaluation of the MAE associated with the guessing approach has been performed on the data collected in the initial DFT-integrated ML batch.

Guessing Method Errors

Guessed Quantity	MAE	MAE Unit	Relative MAE
d_{AB}	0.092	Å	6.4 %
d_{BC}	0.093	Å	5.6 %
d_{CSub}	0.073	Å	4.0 %
m_A	0.997	μ_B	116.8 %
m_B	0.979	μ_B	216.1 %
m_C	1.018	μ_B	148.8 %

Table 3.20: MAE of the initial guess methods used to relax the initial batch of structures, before the DFT-integrated ML process, to converge the database which the ML models have been trained on initially. For the relative MAE, the MAE has been divided by the average quantity itself. The average of each quantity has been computed using the relaxed film's data. The relative MAE is calculated using the standard deviation of the corresponding quantity and dividing it by its mean value. Table extended compared to [55].

From Table 3.20, it can be seen in comparison with Figure 3.35 that the test MAE of the ML-based predictions is significantly decreased compared to the guessing method. Also, as the structural guess is based on the Materials Project [94] database and the magnetic moment is estimated equal for all elements initially, the guessing error is independent of the number of data points computed previously during this study. The ML models test prediction MAE is around at least 33 % better than the guessing MAE. It is self-explanatory that lower deviations from the inputs to the converged/relaxed quantities correlate with an increased likelihood of an *ab initio* computation and relaxation to be successful. Intuitively, drastically improved initial structures can also lead to fewer relaxation steps being required. Furthermore, from Table 3.20, it can be seen that the relative errors of the constant magnetic moment guess are very large; hence, given these large relative deviations between the initial guess and the converged output moments, one would assume that improving the magnetic starting point would already cause a boost in the achieved convergence rate.

The incremental development of the number of successfully relaxed film systems is shown in Table 3.21, which resulted from the ML model's predictions being used as input for re-computed Create-Magnetic-Film workflows to provide optimized starting setups for the SCF calculations and relaxations.

$n_{\text{Iter}}^{\text{DFT IntML}}$	Converged Structures (Abs./Rel.)	Additionally Converged (Abs./Rel.)	ML Predicted Inputs
0	4316 / 64.8 %	n/a.	None
1	4886 / 73.3 %	570 / 8.6 %	Magnetic Layer Moments
2	5917 / 88.8 %	1031 / 15.5 %	Magnetic Layer Moments & ILDs
3	6149 / 92.3 %	232 / 3.5 %	Magnetic Layer Moments & ILDs
4	6228 / 93.5 %	79 / 1.2 %	Magnetic Layer Moments & ILDs
5	6282 / 94.3 %	54 / 0.8 %	Magnetic Layer Moments & ILDs

Table 3.21: Development of successfully relaxed film structures per DFT-integrated ML iteration, including labeling which iteration predicted which input quantities and relative development of the data set size. The 3rd, 4th, and 5th DFT-integrated ML iterations are only distinguished from the 2nd by the additionally acquired data, which is used for model training prior to new input predictions and not by methodological differences.

From Table 3.21, it is immediately apparent that the structural and initial magnetic ML-based optimization had a significantly higher impact on the convergence rate than the sole optimization of the magnetic starting configuration. This can be understood using the workflow failure reasons observed in the initially computed film structures using the described guessing methods. The errors that caused the Create-Magnetic-Film workflow to fail can roughly be classified into multiple different reasons:

Reasons
Relaxations Fail

1. SCF calculation failed to find a stationary point in the energy landscape and did not converge within the set number of iterations (100 per SCF calculation) and SCF calculation restart cutoff (seven restarts).
2. The relaxation failed to converge within the set cutoff of relaxation steps (70 steps).
3. Relaxation failure due to factors related to the structural setup, such as:
 - Muffin-tins of the outer magnetic layer spilled into the vacuum during the relaxation, resulting in a failed relaxation.
 - Muffin-tins of two neighboring layers overlapped, *i.e.* the layers collided with one another, causing the computation to fail.

It is important to point out that only the last of the previously mentioned reasons is LAPW-specific and is hence transferable to other *ab initio* methods. An optimized magnetic configuration can only solve the SCF convergence issue by improving the magnetic starting configuration of the SCF calculation. The sole optimization of the magnetic moments does not alter the initial structural configuration. This way, both other reasons, which are underlying the workflow failure, are not solved. However, an improved input film structure—while improved here means “a structure comparably closer to the actual relaxed film”—has the capability to:

Difference in
Optimizing
Magnetic &
Structural
Configuration

- Reduce the required number of relaxation steps
- Find input structures closer to the relaxed system configuration

Hence, the combination of an optimized initial magnetic configuration and an improved structural input setup has the chance of drastically boosting the convergence rate, by tackling the different workflow failure reasons simultaneously, while at the same time reducing the number of required relaxation steps to acquire a successfully relaxed film configuration. The convergence rate improvement is visualized in Figure 3.36.

From both Table 3.21 and Figure 3.36, it can be seen that after the second DFT-integrated ML iteration, which optimizes both ILDs and moments of the magnetic layers, each of the following iterations has a lower additional convergence rate, compared to the previous iteration. As the predicted inputs have not been extended in the subsequent iterations, this concludes that the incremental changes starting from the third iteration are the result of improved learning of the corresponding models, based on the workflows which successfully relaxed the film based on the predictions of the previous DFT-integrated ML iteration.

Incremental
Convergence
Rate as Result of
Better ML
Modeling

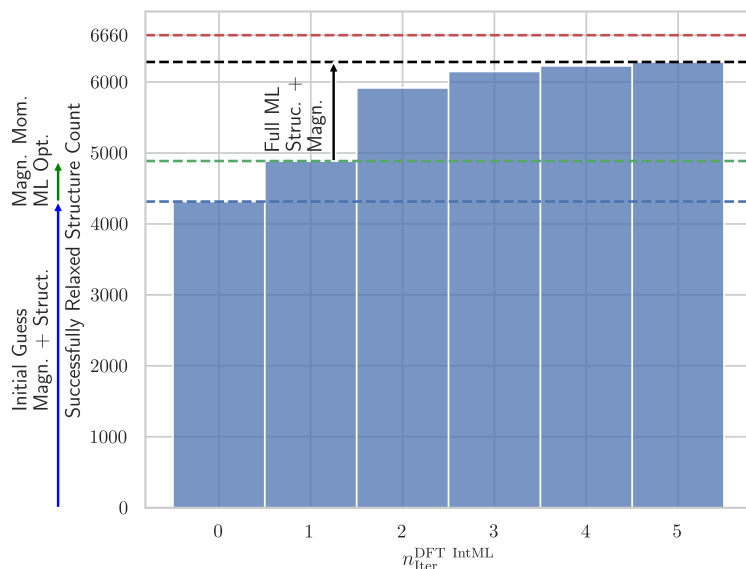


Figure 3.36: Cumulative development of the number of converged film systems under an increasing number of DFT-integrated ML iterations. The horizontal blue line marks the initial count of converged structures without any ML-optimized input parameters, and the horizontal green line marks the count of converged structures, which was achieved by solely improving the magnetic starting point of the SCF calculation using ML. Furthermore, the horizontal black line marks the count of converged structures, which resulted from optimizing both the initial structure and the magnetic starting point using ML, and the horizontal red line marks the number of 6660 structures. This number represents the entire phase space of structures using the previously described film setup and element selection. Posterior to the last ML input modeling step, the failure to reach self-consistency represented the dominant error. Figure adapted from [55].

However, to test the previously mentioned assumptions, it is necessary to examine the convergence rate and the composition of the successfully relaxed structures of the individual DFT-integrated ML iterations. This includes the composition of the different iterations, as well as the overall count of successfully relaxed structures per each element at each possible layer site. This can be used to determine if there are elements that are particularly difficult to relax within the given film structures. The elemental composition for the different iterations is shown in Figure 3.37.

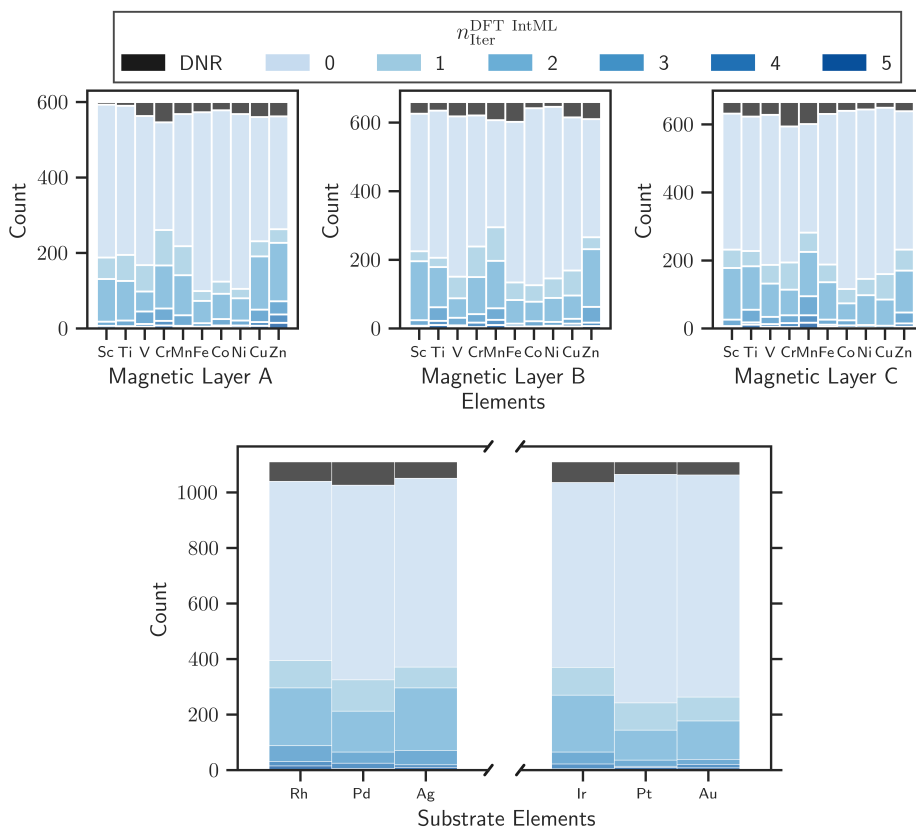


Figure 3.37: Depiction of the counts of the successfully relaxed film structures by their corresponding elemental composition, colored for each DFT-integrated ML integration to gauge the sampling homogeneity of the ML-based input optimization approach. “DNR” is short for “Did not relax” and indicates the number of film systems with the corresponding elemental occupation that did not successfully relax within the Create-Magnetic-Film workflow despite the ML-optimized input parameters. Figure extended compared to [55].

From Figure 3.37, it can be seen that the sampling in each DFT-integrated ML iteration is not equally distributed. However, this concludes that the actual prediction error of the ML-based input parameter optimization approach is likely larger than the test MAE values presented. This is not surprising as the film systems that relaxed in a previous DFT-integrated ML step improve the sampling quality for similar compounds with similar composition within the entire phase space. This leads to local sampling improvements, allowing the prediction of other film systems’ input parameters more accurately than before, eventually causing the corresponding film systems to relax in a subsequently executed workflow. It is also apparent that Cr occupied magnetic layers have difficulty successfully relaxing in the A and C layers. For the C layer, Mn stands out for not successfully relaxing within the Create-Magnetic-Film workflow. Within the B layer, elemental occupations of Mn, Fe, Cu, and Zn seem to relax successfully at a comparably lower rate.

Local Sampling
Improvements

However, overall, there are no extreme trends visible like that of a particular group of elements as e.g. magnetic or non-magnetic prevent a film relaxation systematically in specific layers. This is also true for the substrate layer occupants. As Figure 3.37 does not explicitly display the successful relaxation fraction for film systems where either the magnetic layer A or both the magnetic layer A and B are unoccupied, this information can be found in Figure 3.38.

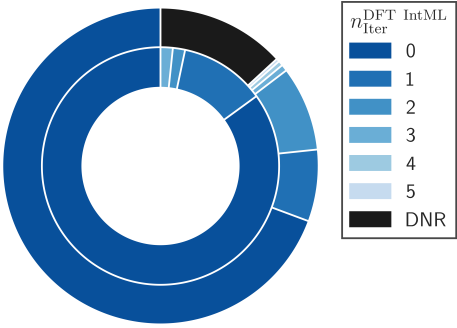


Figure 3.38: Depiction of the convergence rates per DFT-integrated ML iteration for films with two (outer circle) and single (inner circle) magnetic layers. There are 600 structures with only two magnetic layers and 60 with only one magnetic layer in the phase space of all possible structures using the presented setup. Again, “DNR” is short for “Did not relax” and holds the same meaning as in Figure 3.37.

From Figure 3.38, it is apparent about an eighth of the entire phase space of films with only two magnetic layers did not relax successfully. Given that the simulation complexity for only two magnetic layers is reduced compared to three magnetic layers, this is unintuitive. However, the reason might be that significantly less training data is available in the two magnetic layers case, which might lead to less accurate input predictions as in the three magnetic layer setup. Considering that all structures converged for the film systems having only a single magnetic layer—i.e. layers A and B hence are unoccupied—the observation could be related to the fact that there is an even number of magnetic layers present in the intermediate case. However, for a single magnetic layer, the modeling complexity is significantly reduced due to the lack of another relaxing magnetic layer interfering with the relaxation process of the inner layer.

Impact of DFT-integrated ML on Number of Relaxation Steps and SCF Iteration

After the observations that DFT-integrated ML for input optimization based on DFT-originated data has the potential to increase the fraction of successfully relaxed structures from 64.8 % to 94.3 %, in the following the impact of the optimized inputs on the required computing time is examined, followed by an analysis of the actual real-world prediction errors. The required number of total SCF iterations during the Create-Magnetic-Film workflows and the required number of relaxation steps to reach a maximum absolute force of $10^{-3} \frac{\text{Ha}}{a_0}$ are shown in Figure 3.39.

Availability of
Training Data for
two & one
Magnetic Layer
Films

94.3 % of
Structures
Successfully
Relaxed

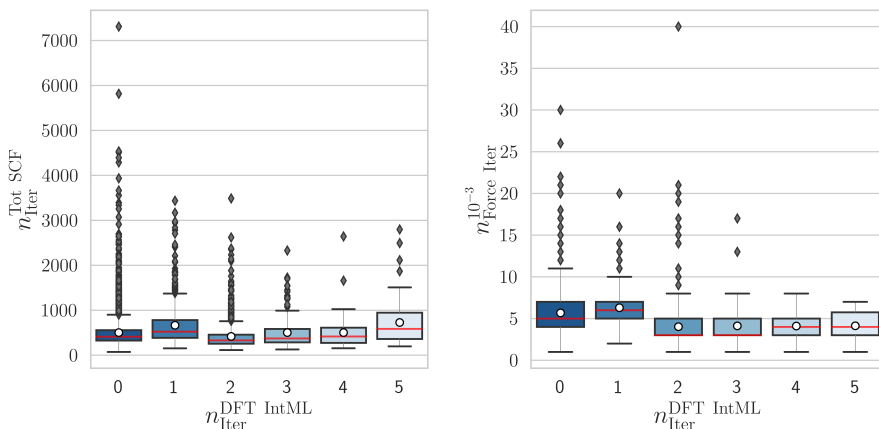


Figure 3.39: Depiction of total SCF iterations (left) and the number of required relaxation steps (right) to reach a force threshold of $10^{-3} \frac{\text{Ha}}{a_0}$ for each DFT-integrated ML iteration. The white dot marks the average value for the corresponding DFT-integrated ML step, while the red line marks the median value. For all following boxplots, the whiskers denote 1.5 times the interquartile range from the median, and the diamonds indicate outliers that exceed the whisker range. The interquartile range corresponds to the boxes' overall height, which means that a box contains half of the data points for which the distribution is depicted.

From Figure 3.39, it can be seen that the total number of SCF iterations performed during the Create-Magnetic-Film workflows in the different DFT-integrated ML iterations first increase on average for the iteration in which only the magnetic initial moments have been ML predicted, which indicates that the modeling complexity of the relaxation process is increased despite the improved SCF magnetic starting point. However, as both the structure alongside the magnetic moments are predicted using the ML model, the average number of total iterations is lowered below the initially—*i.e.* without any ML optimized input parameters solely using the described constant magnetic guess and the bond length estimate—required average number of total iterations. From this point on, the average number of total SCF iterations is increasing, which is expected, as for each additional DFT-integrated ML iteration, the films which remain to be computed have not been successfully relaxed multiple times. Hence, each further DFT-integrated ML iteration leaves film systems to be computed, which appear harder to relax than the structures relaxed in the previous iterations. A very similar trend as for the average total number of SCF iterations can be seen for the number of required relaxation steps to reach an absolute maximum force of $10^{-3} \frac{\text{Ha}}{a_0}$. From Figure 3.39, it can also be seen that most systems do not require nearly the set cutoff of at most 70 relaxation steps to relax successfully.

Based on the results shown in Figure 3.39, it is clear that there is a significant improvement when it comes to the average required computational time. The averages from Figure 3.39 are shown in Figure 3.40 alongside the average number of relaxation steps needed to match the strict set force criterion of an absolute maximum force of $10^{-5} \frac{\text{Ha}}{a_0}$ per DFT-integrated ML iteration.

Decreased
Number of Total
SCF Iterations
Required for
Complete Film
Relaxation for
Structural &
Magnetic ML
Optimization

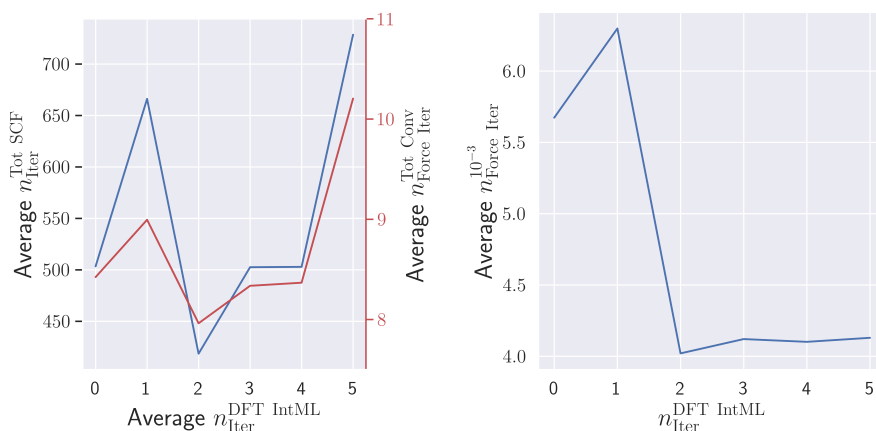


Figure 3.40: Depiction of the average number of total SCF iterations in each DFT-integrated ML iteration (left, blue graph left axis), average number of relaxation steps to reach complete force convergence for each DFT-integrated ML Step (left, red graph right axis), and average number of relaxation steps to reach a maximum absolute force of $10^{-3} \frac{\text{Ha}}{a_0}$ in each DFT-integrated ML iteration (right). It is visible that the number of needed relaxation steps and the number of required SCF iterations for each batch are correlated. This figure was adapted from [55].

From Figure 3.40, the trend that was already visible in Figure 3.39 becomes clearer, as the trend is the same for all three convergence indicators. The average required number of force iterations to reach an absolute force of less than $10^{-3} \frac{\text{Ha}}{a_0}$ for each DFT-integrated ML iteration shows a less steep increase of average required iterations for larger DFT-integrated ML iteration numbers as both other relaxation convergence related quantities. This can be interpreted in the sense that to lower the force convergence criterion in the relaxation from $10^{-3} \frac{\text{Ha}}{a_0}$ to $10^{-5} \frac{\text{Ha}}{a_0}$ significantly more iterations are required than were necessary to—in average—completely relax the film structures before any ML-based input parameter optimization. However, relaxations using a force criterion similar to the threshold of $10^{-3} \frac{\text{Ha}}{a_0}$ can already be sufficiently relaxed, depending on the intended use of the relaxed structures. Comparing how many SCF iterations are required on average without any ML-based input parameter optimization to how many are needed with both the magnetic moments and structure inputs predicted, it is possible to reduce the number of required SCF iterations in an average film relaxation by around 17 % and reduce the number of average required relaxation steps from—in this example—at least 6 % up to around 29 %, depending on the chosen convergence criterion. The 17 %, on average, less required SCF iterations directly translate to effectively 17 % less required computing time for the average film relaxation. This does not directly apply to the number of required relaxation steps. However, fewer relaxation steps ultimately translate to fewer DFT calculations that must be performed to relax the structure successfully. This, again, translates to a lower number of calculations needed to be submitted to a computing center or device. Therefore, this approach reduces the overall workload of the system and the subsequently occurring traffic and computing time consumption.

Potentially 17 %
Less SCF
Iterations

Potentially 29 %
Less Relaxation
Steps

Comparing the average total number of required SCF iterations for the magnetic ML optimized inputs and for both the magnetic as well as the structurally optimized inputs, the average reduction of required SCF iterations even increases up to around 37 %. Hence, it is safe to say that utilizing acquired data during high-throughput computations to optimize further computations inputs can drastically decrease the required computing time with relatively low implementation effort when implemented as a particular application case of batch learning while at the same time boosting high-throughput DFT convergence rates. As already discussed, it should be aimed to optimize as many input quantities as possible relevant to the workflow and simultaneously achieve the most significant impact of the presented DFT-integrated ML methodology. However, this also raises the question of when the ML-based optimization of the input parameters should be started.

Data Requirements of DFT-integrated ML

As seen from Figure 3.35, the test MAE values of the used models do not change drastically for additional DFT-integrated ML steps even though more data is acquired. Of course, the ideal amount of available training data to start with the presented methodology will vary for each problem depending on a few factors, including the modeling complexity of the particular problem. To determine the accumulated amount of data, for which starting with the first iteration of DFT-integrated ML for input parameter optimization would have been beneficial, it is possible to examine the model performance on unseen data as a function of the accumulated data amount. A posterior analysis has been performed for which the relative error comparing the guessing MAE and the prediction MAE is shown as a function of the used amount of training data. This relative error comparison is shown in Figure 3.41.

Ideal Starting
Point

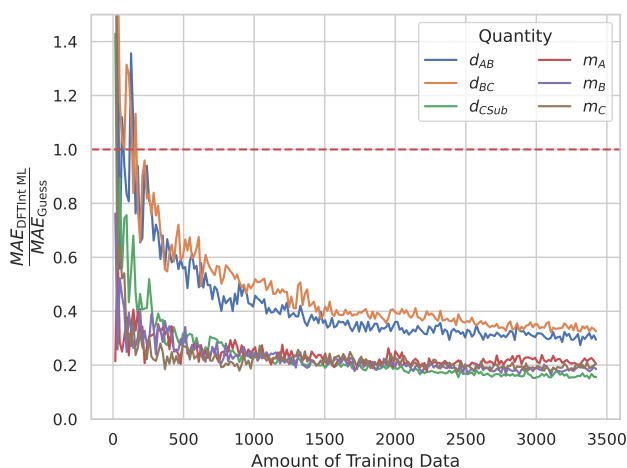


Figure 3.41: Comparison of the guessing MAE and the prediction MAE depending on the available training data using a relative error. The red dashed line marks the point below which the methodology of DFT-integrated ML outperforms the described guessing method. The development of the guess MAE, depending on the increase of additionally acquired data, is relatively constant, which is briefly shown in appendix C. Figure adapted from [55].

For the posterior analysis displayed in Figure 3.41, the ML models used to predict the optimized inputs have been trained on different sizes of training data sets shown on the x -axis in incremental steps of 20 data points. The training data sizes indicated on the x -axis result from an 80 % to 20 % training and test splitting. Hence, the MAE of the prediction could be evaluated by predicting the 20 % test set and comparing the result to the actual DFT result. Similarly, the guess error was assessed using subsets of the input guess from the initially 4300 relaxed structure data and comparing it to the DFT results. The hyperparameter optimization in this posterior analysis, compared to the actual predictive modeling case, was skipped, and a generic XGBoost [125] with a fixed number of estimators (as documented in [338]) has been used. From figure 3.41, it is apparent that the DFT-integrated ML approach would have had lower MAE values and would hence have provided better input parameters than the discussed structural and magnetic guesses, already after less than a few hundred training data points have been available. Furthermore, as the relative MAE improvement rate is significantly higher for small training data sets, an earlier start of the DFT-integrated ML scheme would ideally be combined with more frequent model retraining at the beginning, to benefit from the incrementally improving prediction accuracies. The slope of the visible relational error curve also explains why there is no clear trend for the test MAE visible in Figure 3.35 for increasing data amounts for the different model iterations, as all the predictive ML models were trained on data set sizes beyond 4300 data points. This indicates that an earlier start concerning the prediction instead of guessing input quantities would have been beneficial to the quality of inputs and subsequently also to the convergence rate and effectiveness of computing time usage. Concerning future high-throughput studies using DFT-integrated ML, automated and continuous monitoring to recognize when ML model predictions could be used as inputs instead of an initial guess is advised.

As previously described, DFT-integrated ML can be seen and applied as an iterative process, which is formally depicted in Figure 3.42. However, what defines an individual iteration varies depending on the high-throughput setting, project scale, and available resources and infrastructure for *e.g.* frequent model training. This relates to the decision field in Figure 3.42, which decides if a “significant” amount of additional data has been accumulated by successfully finished computations. The term “significant” could be defined as small as an individual additional data point. This would conclude that retraining the prediction model could be performed for each additionally acquired data point in the most extreme case, which would possibly justify switching to an online learning approach - with the foreseeable drawback of having a model more difficult to evaluate. However, more practical would be defining a “significant” additional data amount as a simple threshold and retraining when an increased model test performance can be achieved based on the additionally acquired data. This more frequent retraining would improve the input’s accuracy on the given data basis as soon as possible. A possible consequence would be that the observed computing time-reducing effects of DFT-integrated ML could arise earlier, compared to the presented example, for which most successful relaxations did not benefit from optimized input parameters.

As the sampling of the individual DFT-integrated ML iterations was governed by the successful relaxation of the structures rather than a specific sampling strategy, it is essential to mention that if DFT-integrated ML is applied in a high-throughput setting, sampling has to be considered for both the initially gathered training data—which is acquired before any ML integration—and the additionally acquired data as a result of each model iteration to avoid OOS prediction as the high-throughput study progresses, as OOS predicted input parameters could negatively impact the success rate of the executed calculations and workflows. A possible solution strategy is randomly sampling each structure, which has not yet been successfully computed, upon each workflow submission to the used computing architecture.

Start After a Few
Hundred Relaxed
Structures Would
Have Been Ideal

Discussion of
Batch Data Sizes

Careful Sampling
to Avoid OOS
Predictions

As already discussed by the comparison of Figure 3.35 and Table 3.20, it can be seen in Figure 3.41 that for large amounts of training data, the ML prediction MAE relative to the guess error approaches a value of about $\frac{1}{3}$ for the ILDs d_{AB} and d_{BC} , while the other predicted quantities approach MAE values of about a fifth of the MAE of the initial guesses, which also reflects the modeling complexity for the individual ILDs, based on their position in the film structure, as previously discussed. This translates to drastically improved (Improvement ranging from approximately 66 % up to 80 %) input parameters using the DFT-integrated ML approach than using the described guessing methods in this posterior analysis of the films, which relaxed using only the initial guesses. While, of course, the fixed initial moments guess is rather simple, it is surprising that the ML-based input parameter optimization can significantly improve the initial ILDs compared to the bond length-based estimates, as the initial ILD estimation based on the Materials Project [94] database bond lengths already used dedicated routines designed to determine fitting initial ILD guesses. Details on the computation of ILDs from the initial guesses can be found in appendix B. [11, 92]

Theoretical Input
Improvement
Range of 66 %
up to 80 %

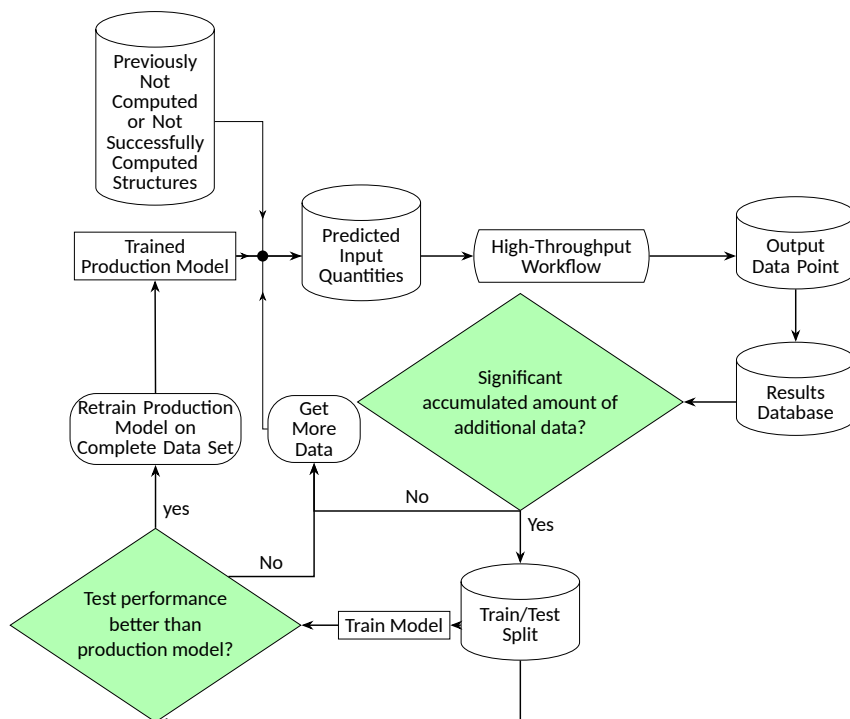


Figure 3.42: Formal schematic depiction of the DFT-integrated ML scheme as an iterative procedure. The green colored fields represent decision fields. Cylinder-shaped fields represent stored, predicted, or handled data. As soon as sufficient data has been accumulated to train a model that outperforms the guessing method, this trained initial model represents the starting point of the iterative DFT-integrated ML scheme. Figure adapted from [55].

While the posterior analysis, based on the ML models test MAE values, yields promising insights regarding the relatively improved predicted input quantities compared to the initial guessing method. This posterior approach again assumes that the predicted ILDs and magnetic moments do not affect other calculation setup parameters and that the results from different DFT-integrated ML iterations do not exhibit systematic deviations when compared. However, it is also possible to compute the MAE of each input quantity prediction, which resulted from a successfully executed Create-Magnetic-Film workflow, compared to the relaxation results obtained from the successful *ab initio* workflows. This MAE is in the following referred to as the actual prediction MAE. The actual prediction MAE is displayed together with the test MAE for the different DFT-integrated ML iterations for both the ILDs and magnetic layer moments in Figure 3.43.

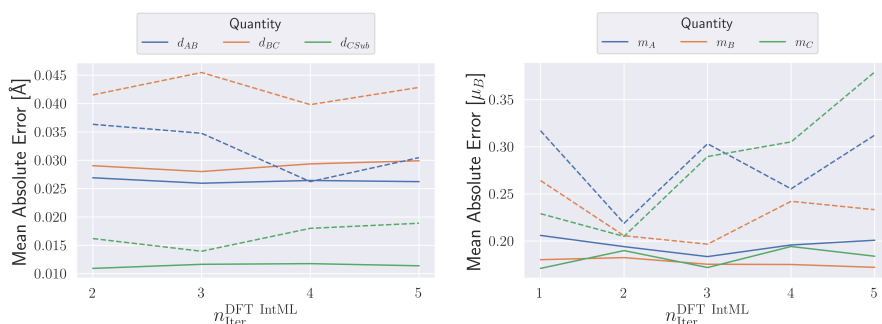


Figure 3.43: Posterior depiction of both the model's test MAE values, depicted as continuous lines, as already seen in Figure 3.35, and the MAE arising from the difference of the predicted ILDs (left) and the predicted magnetic moments (right) compared to the DFT results, depicted as dashed lines. This figure is adapted from [55].

From Figure 3.43, it can be seen that the test MAE values are systematically smaller than their prediction MAE value counterparts. Still, for the ILD predictions, the largest MAE is 50 % smaller than the corresponding initial ILD guess MAE. For the magnetic moment's predictions, the largest MAE remains over 60 % smaller than the constant initial guess MAE. Increasing values for the prediction MAE for the last DFT-integrated ML iteration indicate that the magnetic moments are more complex to model for the films, which were still present in this final iteration. This is a similar effect observed in Figure 3.40 for the increasing number of relaxation and SCF iteration steps. Furthermore, for the DFT-integrated ML step for which the magnetic layer moments and the ILDs have been optimized simultaneously for the first time, an MAE minimum is observed for the magnetic layer moments MAE depiction is visible. This could be interpreted in the sense that a large proportion of the systems relaxed in this particular iteration acquired magnetic moments and were close to the ML-modeled moments. Again, this indicates that optimizing both the structure and the magnetic layer moments simultaneously was crucial for both the improved convergence rate and the reduced number of required iterations. Modeling both structural and magnetic film properties simultaneously enabled the relaxed film properties to be approximately matched best using the ML predictions before computing the particular structure.

For the ILD predictions, the MAE values show only slight variations over increasing numbers of DFT-integrated ML iterations. A potential explanation for the fact that the test MAE values are located systematically below the actual prediction MAE values is that the different DFT-integrated ML iterations—as already observed—do not equally well sample the phase space of not yet successfully relaxed structures and are systematically different from each other. The following examines whether there is a systematic difference between the different DFT-integrated ML iterations present. Summarizing the findings about the prediction improvement, it can be said that the posterior analysis revealed that the MAE of the actual prediction is increased compared to the test MAE. However, a 50 % to 60 % improvement of the predicted input quantities in comparison to the initial guesses is achieved, and the actual prediction MAE serves as a more realistic estimation for the predictive performance of this model than the prior test MAE values. Additionally, the observation that a 17 % speedup of the *ab initio* calculations could be achieved by ML optimizing the magnetic moment inputs together with the ILD inputs is remarkable. It serves as a promising proof of concept for the DFT-integrated ML methodology.

Batch Independence

Examining the systematic differences between the DFT-integrated ML iterations, *i.e.* the batches of this particular batch learning application, can be done by comparing the deviations of the model's predictions to the DFT relaxation outcome. This would reveal if any of the DFT-integrated ML iterations stands out amongst the others or if there are trends visible like *e.g.* progressively increasing or decreasing deviations of predicted input quantities and the properties resulting from the relaxation or spread thereof. For the ILDs, the distribution of deviations of the ILDs Δd_{XY} which are defined for arbitrary neighboring layers X and Y as in equation (3.9), is displayed in Figure 3.44.

Examination of
Systematic
Deviations
Between Batches

$$\Delta d_{XY} = d_{XY}^{\text{DFT}} - d_{XY}^{\text{Pred}} \quad (3.9)$$

From Figure 3.44, it can be seen that in particular for the ILD d_{BC} , there is a comparably significant deviation of the predictions from the DFT results present. For the other ILDs, no apparent large deviations from the value of 0 Å are visible, representing the desirable value regarding residuals of the predicted values and the DFT-based ground truth. For each ILD it is apparent that for data batches for which the inputs ILDs are represented by the initial average bond length-based guess ($n_{\text{Iter}}^{\text{DFT IntML}} \leq 1$), a large spread of residuals can be observed compared to the following DFT-integrated ML iterations.

Discussion of
Predictions
Residuals as
Measure of
Systematic
Deviations

Beyond the spread, the initial guess seems to overestimate the ILDs across all three magnetic layers systematically. At the same time, the ML-based predictions are far more centered around the value of 0, indicating nearly vanishing residuals. Also, as already previously observed (like *e.g.* in Table 3.20, Figure 3.43, and Figure 3.35), the spread is decreased for the residuals of the ILD between the magnetic C layer and the substrate, compared to the other ILDs. This indicates the reduced modeling complexity of this particular ILD (d_{CSub}) compared to both others.

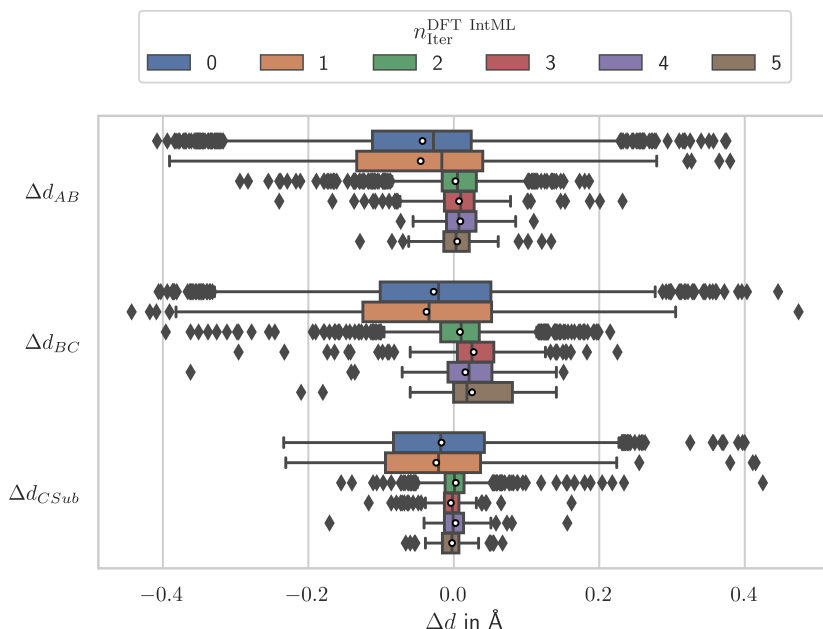


Figure 3.44: Depiction of the distribution of the difference of input and relaxed ILDs for each DFT-integrated ML iteration. Inside the colored boxes, the dark lines represent the median values. The white circle markers represent the mean values.

Figure 3.45 shows the differences between the input magnetic layer moments and the *ab initio* results. The sign convention chosen here is analogous to equation (3.9). From Figure 3.45, it can be seen that for the magnetic moments of the magnetic layers, A and B, the magnetic moment distribution spread, denoted by the whiskers, decreases for iterations applying the DFT-integrated ML procedure. For all magnetic moment layers, the median residual values are situated closely to the value of $0 \mu_B$. The spread decrease observed for the A and B magnetic layer moments seems reversed for the magnetic C layer. The outliers are significantly more prominent for the magnetic layer moments residuals than in the ILD residual depiction. Combining the observations gathered from both Figures 3.44 and 3.45, it can be concluded that fewer systematic deviations are present for the magnetic layer moment residuals. However, this can be explained by revisiting the previously mentioned assumption about the independence of calculation setup parameters.

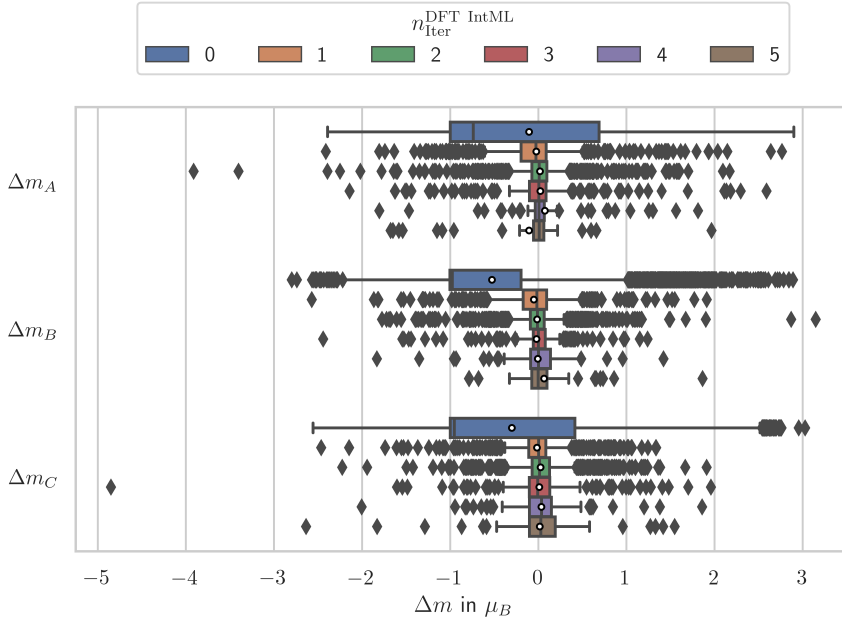


Figure 3.45: Depiction of the distribution of the difference of predicted and converged magnetic layer moments, obtained from the relaxed film structures *ab initio* results, for each DFT-integrated ML iteration. Inside the colored boxes, the dark lines represent the median values. The white circle markers represent the mean values.

The film setup is entirely performed from scratch as the structural setup is altered using the ML-predicted ILDs. Hence, during the setup, the muffin-tin radii were recomputed by the FLEUR input generator. The change of the muffin-tin radii subsequently changes the forces computed, which are acting on the atoms and changing their positions during the relaxation. This is not true for LAPW calculations with considerably more dedicated convergence parameter choices. However, choosing such convergence parameters in high-throughput settings is not feasible due to the attached increased computation cost. Therefore, the changed ILDs can slightly alter the muffin-tin radii, which impacts the relaxed structure. This co-dependence of the relaxation outcome on the altered structural and muffin-tin setup also potentially explains the observed (see Figure 3.40) substantially different development—for increasing DFT-integrated ML iteration numbers—of the required number of relaxation steps to:

1. reach the maximum absolute force convergence criterion of $10^{-3} \frac{\text{Ha}}{a_0}$
2. reach the maximum absolute force convergence criterion of $10^{-5} \frac{\text{Ha}}{a_0}$

It is possible that without the results shifting impact of the predicted input structures, the structural input optimization using DFT-integrated ML could reduce the actual prediction MAE even further. This could also lower the number of required relaxation steps to reach a completely relaxed structure—even applying a similarly strict convergence criterion—also for increasing iteration numbers of DFT-integrated ML. The reason the number of required relaxation steps to achieve completely relaxed film structures is increasing as more structures—also including such that converged using ML optimized input structures—have been learned is related to the data homogeneity of the structural setups (including e.g. ILDs and muffin-tin sphere radii), which at first is very high, as the subset of potential ILD guesses is very small and hence also the subset of elemental muffin-tin radii is more homogeneous. However, as the data fraction of structures relaxed using ML-optimized structural inputs increases, homogeneity gradually decreases. This is because the model is not aware of the systematic peculiarities of different DFT-integrated ML iterations, such as the structure-dependent muffin-tin radii used in the setup. In fact, the model suggested relaxed ILDs may deviate slightly from the real relaxed ILDs in further DFT-integrated ML iterations. This is due to the predictions being compiled by models that have been trained on data originating from multiple different DFT-integrated ML iterations. As a result, the slight systematic residual relaxed ILD deviations emerging from the altered muffin-tin radii remain unaccounted for. There are two possible solutions to overcome this challenge. These changes can be accounted for by manually setting constituent element-specific muffin-tin radii across all DFT-integrated ML steps of a project. Alternatively, a relaxation method that does not incorporate the concept of muffin-tin spheres in the structural setup can be used. Implementing either of these solutions can potentially achieve a further decrease in required relaxation steps, making the methodology of DFT-integrated ML more effective for future studies.

3.3.4 Explainable Artificial Intelligence

Beyond examining the ML model's predictions and residuals, it is possible to explore the marginal feature impacts on the predicted quantities of the used models using the SHAP [128] package for global model explanations. However, rather than examining the SHAP summary plot of each model individually in the following, a comparative approach is discussed. The average absolute SHAP values corresponding to the used features for each predicted quantity are displayed in Figure 3.46. This visualization was compiled in a posterior analysis by retraining the models used for the predictive modeling process on the entire available data set after a brief evaluation. It should be mentioned that the evaluation of the retrained models, for which the average SHAP value magnitudes are shown in Figure 3.46, yielded similar test MAE values as already presented in Figure 3.35 previously. Furthermore, the computed coefficient of determination on the test set of the models—before retraining on the entire available data—was calculated to be larger than 0.98 for the ILD prediction models and larger than 0.85 for the magnetic layer moments. Hence, it is clear that the ILDs can nearly perfectly be modeled using only the four atomic numbers.

This is not the case for the magnetic layer moments, as by the interpretation of R^2 , up to around 15 % of the magnetic layer moments variation can not be accounted for using the trained model. However, a few candidates for additional features could increase the coefficient determination for the magnetic layer moments, including the structure and the muffin-tin radii.

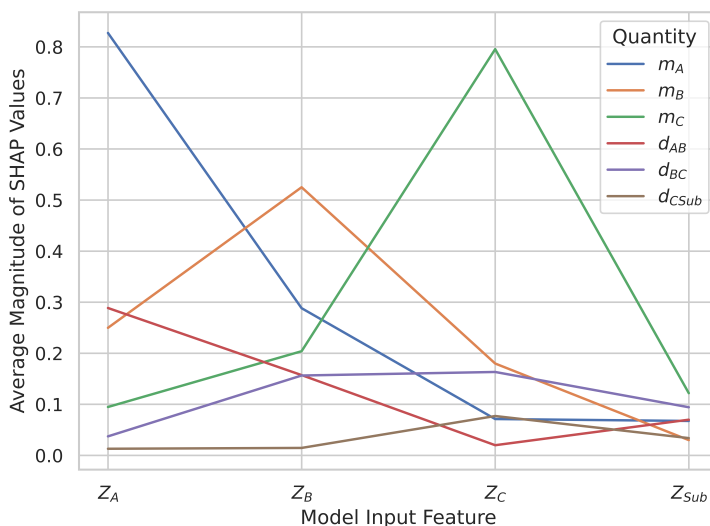


Figure 3.46: Depiction of the average magnitudes of the SHAP values obtained using the same models as were used for the DFT-integrated ML procedure of each input feature and each predicted quantity. The models used to compute the SHAP values, which this depiction is based on, have been trained using all successfully relaxed structures available after 5 DFT-integrated ML iterations.

From Figure 3.46 it can be seen that for all predicted quantities, the specific film layer site's atomic numbers, which can be considered the closest related to the particular predicted quantity, have the most significant impact on the predicted quantity. For the magnetic layer moments, this concludes that the element situated at the moment for which the magnetic moment is predicted has the most absolute average impact on the prediction. For the predicted ILDs, this translates to the atomic numbers of both layers separated by the corresponding predicted ILD having the largest impact on the ILD prediction. This observation of the most impactful features follows the physical intuition that the material composition in proximity should determine local material properties. A neighboring degree is defined in the following to examine this intuition further. It describes in which order the potential impact of the corresponding input feature on the predicted quantity is by effectively counting how many ILDs are situated between the layer the feature is associated with and the next layer, which has the closest proximity possible to the predicted quantity. Using this definition, the neighboring degree is given in Table 3.22 for each feature and predicted quantity.

Using the definition of the neighboring degree and the average magnitude of SHAP values presented in Figure 3.46, it is possible to examine the relation between the average impact magnitudes of the different features on the different predicted quantities (feature-specific SHAP values) and the spatial distance of the corresponding feature to the proximity of the predicted quantity in units of ILDs (neighboring degree). The corresponding depiction is given in Figure 3.47.

SHAP Value
Magnitudes
Directly
Impacted by
Atomic Number
Features in
Spatial Proximity
Neighboring
Degree

Predicted Quantity	Z_A	Z_B	Z_C	Z_{Sub}
m_A	0	1	2	3
m_B	1	0	1	2
m_C	2	1	0	1
d_{AB}	0	0	1	2
d_{BC}	1	0	0	1
d_{CSub}	2	1	0	0

Table 3.22: Overview of the neighboring degree of the ML model's input features to the respective predicted quantities. The numbers given for every predicted quantity and feature pair represent the neighboring degree.

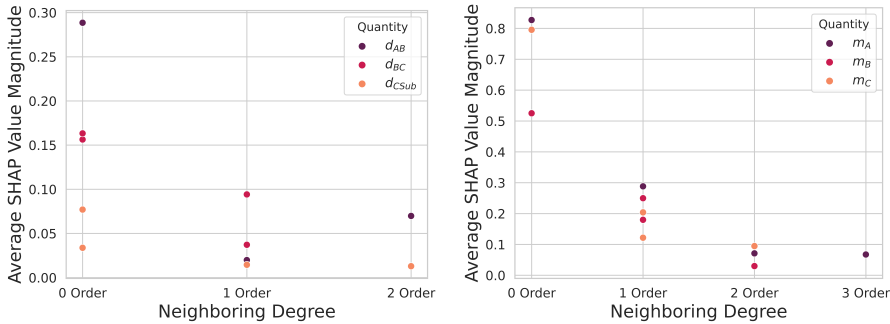


Figure 3.47: Depiction of the predicted quantity-specific average magnitude of SHAP values in relation to the input features encoded using the corresponding neighboring degree. The neighboring degree is used as established in Table 3.22. The plot containing the ILDs is shown on the left, while the plot containing the magnetic layer moments is depicted on the right.

From Figure 3.47 it can be seen that indeed there is a clear trend visible that the average feature impact magnitude is decreasing for increasing neighbor degrees. The substrate choice significantly impacts the ILD d_{AB} , already visible in 3.46. It is clear that the substrate choice significantly influences the film's (electronic) structure simply because the substrate makes up a significant proportion of the film's unit cell atoms and governs the in-plane lattice constant for the magnetic layers. The trend for the magnetic moment associated with average SHAP value magnitudes even suggests an exponentially decreasing behavior with increasing neighboring order. This further supports the previously mentioned idea of locality of materials properties, which will also be used in the following section. This application of the SHAP package [128] is a prime example of how XAI can be used to gain insights about the underlying physics contained in the data, which an ML model has learned. As demonstrated in the next section, a distance-based exponentially decreasing impact on the neighbor's magnetic properties can be explained using the previously suggested locality assumption.

Suggested
Exponential
Decrease of
Average Impact
in Relation to
Increasing
Spatial Proximity

3.3.5 Data Analysis

This section presents an in-depth analysis of the collected magnetic data. This discussion covers the magnetic layer moments, the substrate layer moments, the magnetic states, and the magnetic inter-layer coupling.

Magnetic Layer Moments

This section discusses the distributions of total magnetic moments of the relaxed films and the magnetic moments of the individual magnetic layers. The distribution of the total film moments per substrate element is depicted in Figure 3.48.

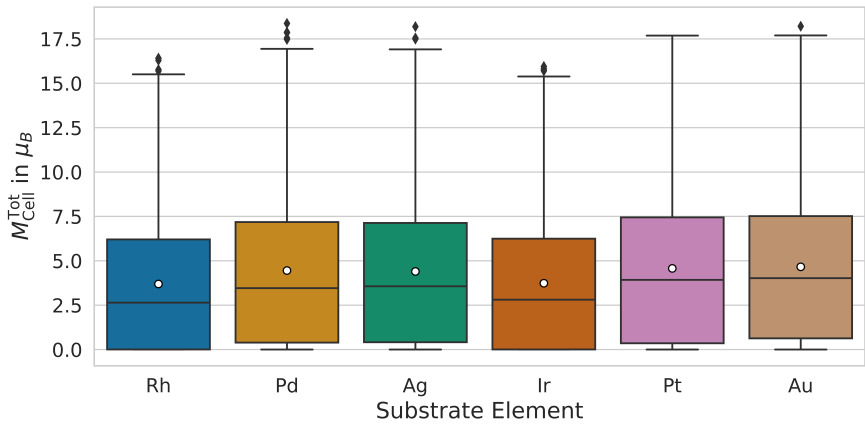


Figure 3.48: Distribution of total magnetic film moments for each substrate element. The horizontal lines in the boxes denote the median, while the white circle markers denote the average of the total film's magnetic moments.

From Figure 3.48 it can be seen that both elements from group 9 of the periodic table exhibit lower average total magnetic cell moment than the other substrates. Beyond the slight variation in averages, the distributions of the moments exhibit similar widths. It is worth mentioning that not a single compound switched the sign of the total moment despite individual negative layer moments appearing within the data set. This means that the initially given magnetic direction was maintained, even after the magnetic moments had been initialized using an ML model. The largest total film's magnetic unit cell moment within the data set was determined to be $18.373 \mu_B$ for the film structure $\text{MnCoFePd}_5\text{FeCoMn}$. Beyond the presented macroscopic overview of the film's unit cell's total magnetic moment, the investigation of the resulting magnetic moment for each magnetic layer is possible and depicted in Figure 3.49.

Group 9 Noble
Metal Substrate
Films Exhibit
Smaller Average
Total Magnetic
Moment

Maximum Total
Magnetic
Moment
 $18.373 \mu_B$

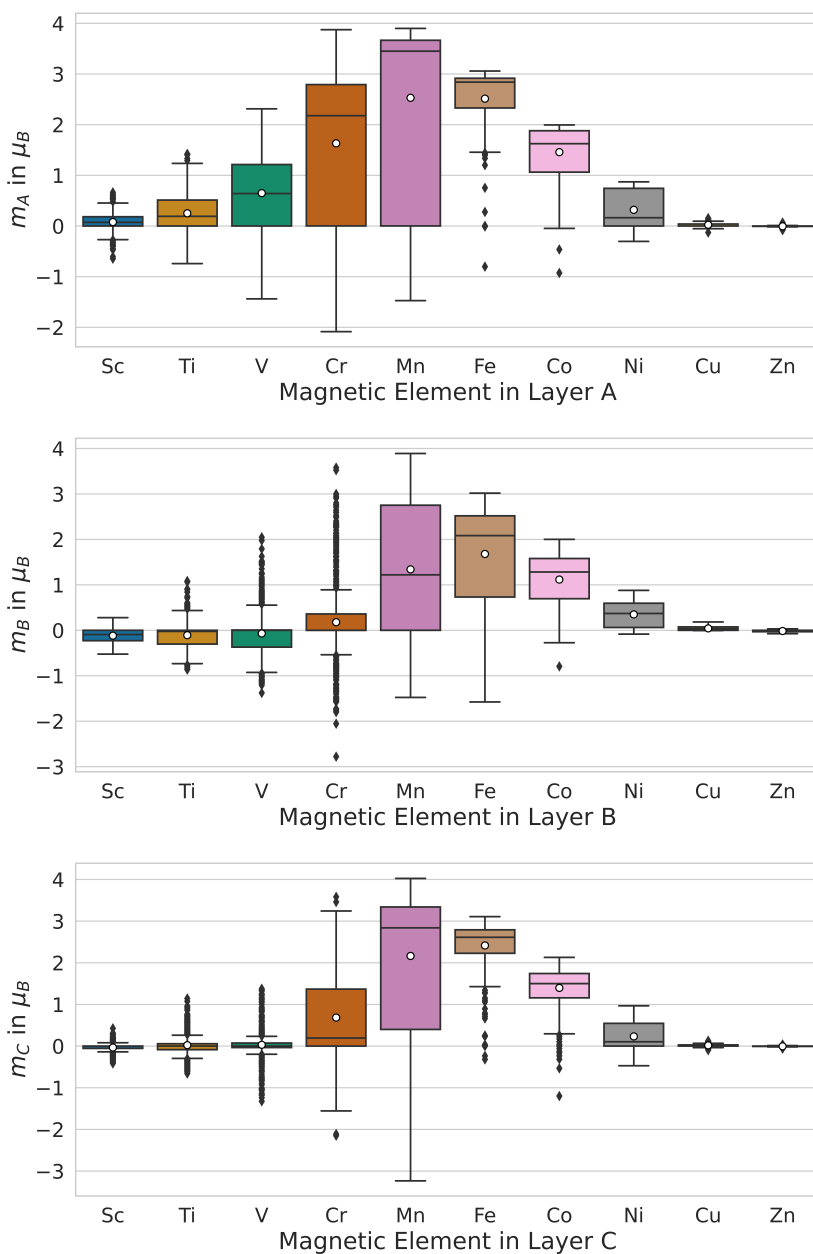


Figure 3.49: Distribution of the magnetic moments situated at the different magnetic layer sites of the relaxed film structure. The horizontal lines in the boxes denote the median, while the white circle markers denote the average of the individual layers' magnetic moments. The A layer's magnetic moments distribution is shown on top, the B layer distribution is given below, and the magnetic moment distribution for the elements of the C layer is displayed at the bottom.

In Figure 3.49, it can be seen that there are substantial differences in the distributions of the magnetic moments present for different elements as well as in different layers. At first glance, it can be seen that in all layers, the largest magnetic moment can unanimously be associated with the Mn atoms occupying the respective layer and is about $4 \mu_B$. Additionally, it can be observed that the average magnetic moment sizes can be arranged as $\overline{m}_A > \overline{m}_C > \overline{m}_B$. Remarkably, in both the A and the C layers, there are outliers—denoted as black diamonds within the shown box plots—for iron, occupying the respective layer, very close to a vanishing magnetic moment. This is also the case for cobalt in the magnetic C layer. This is surprising for these elements, as both elements are part of the group of inherently FM elements. From the elements that typically do not exhibit significant magnetic moments in their bulk systems, like Sc, Ti, and V, it can be observed that both the spread and the average of the magnetic moments do decrease towards the layers situated farther away from the film's surface. This reflects how surface effects in film systems can enhance magnetic properties, which are not observed in bulk systems. Additionally, in Figure 3.49, the impact of alternating moments can also be observed.

Uniformity of Maximum Magnetic Moment for Each Layer

Surface Effects Impact

A particular observation in this regard is related to the averages of the Cr, Mn, and Fe magnetic moments. The averages are increased in layers A and C while simultaneously suppressed in the B layer. Inspecting the spread indicated by the whiskers in the box plots in Figure 3.49 of both the Cr and Fe originating moments, it can be seen that the spread width is precisely opposed to the respective other element's moment spread. At the same time, the spread width also displays an altering behavior for both elements. In the C layer, the average magnetic moments originating from every element except Cr, Mn, Fe, Co, and Ni are vanishing. Unsurprisingly, the Zn associated magnetic moment is vanishing for all layers unanimously. In fact, both Cr and Mn tend to also appear with a negative magnetic moment in the A and C layers, indicating that these elements tend to exhibit AFM pair exchange interactions with other elements from the selection of magnetic layer elements.

AFM Pair Exchange Interactions

Another observation regarding the impact of DFT-integrated ML can be derived from the distribution of the film's unit cell's magnetic moment's distribution, shown in Figure 3.50. From Figure 3.50 it can be observed, that even though the initial magnetic moment of $1 \mu_B$ per magnetic layer atom, which has been used—for the magnetic initialization—prior to the use of ML optimized input parameters, was used, a broad spectrum of total film cell moments including moments above $15 \mu_B$ resulted from the initial guess input parameters. For the following DFT-integrated ML iterations, it can be observed that the fraction of non-magnetic or weakly-magnetic relaxed films increased compared to the absence of ML optimized magnetic input moments. Across all iterations of DFT-integrated ML, the average magnetic moment of a film's unit cell only exhibits slight variations, suggesting no systematic discrepancies in the sense that a specific model iteration favored the prediction of either exceedingly large moments or considerably underestimated magnetic moment configurations. However, there are also systems contained in the data set where magnetic moment induction amongst the magnetic layer elements occurs, which will be discussed in the following.

Strong Resulting Magnetic Moments Despite Relatively Low Initialization

Induced Magnetic Moments in Magnetic Layers

Before discussing induced moments within the magnetic layers, it is necessary to clarify what is considered an induced magnetic moment. Considering the magnitude of substrate moments—which by definition are induced by the neighboring magnetic layers—depicted in Figure 3.53, it is safe to say that most of the induced substrate moments are situated well below $0.4 \mu_B$.

Definition of Induced & Inducing Moments

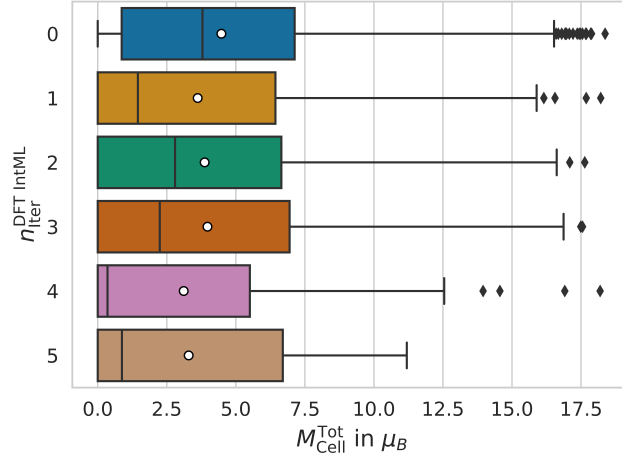


Figure 3.50: Depiction of the film's unit cell magnetic moments across the different DFT-integrated ML iterations. The horizontal lines in the boxes denote the median, while the white circle markers denote the average of the total film's magnetic moments per the corresponding DFT-integrated ML iteration.

Hence, in the following, moments of a magnitude smaller than or equal $0.4 \mu_B$ are referred to as induced moments regardless of their corresponding sign. In contrast, moments with a magnitude greater than $0.4 \mu_B$ are referred to as inducing magnetic moments. Beyond magnetic and induced moments, defining the magnetic pair interaction type between two neighboring layers is necessary. However, when classifying the magnetic pair interaction types, the sign of the neighboring moments is considered to characterize the magnetic interaction of the layers. Generally, suppose a single-inducing magnetic moment is neighbored by a single-induced magnetic moment. In that case, the pair interaction type can be determined by simply multiplying both magnetic moments and examining the sign and magnitude of the result. Effectively, if the result of the moment's product is zero, the pair interaction type is non-inducing as one of both neighbors is not magnetic at all. If the result represents a positive number, the pair interaction type is FM, as both neighbors share the same magnetic moment direction. However, if the moment's product is negative, the pair interaction type is AFM, as the neighboring magnetic moments are antiparallel. This definition is in line with Figure 2.4. However, given that more than just two film layers are contained within the examined structure's magnetic layers, the presented definition is sufficient for determining the interaction type of layers A and C only. This assumes that the actively inducing effect of the substrate on the magnetic moment of C is negligible. Special treatment is necessary to determine the pair interaction type of an induced B layer moment, as the layer could be induced by both neighboring magnetic layers simultaneously. For simplicity, only such pair interactions have been considered for the B layer, where the pair interaction type between the layers A and B matches the pair interaction type of the layers B and C. The pseudocode algorithm which was effectively used in order to determine the pair interaction types, based on the computed magnetic layer moments, is shown in Algorithm 3.1.

Pair Interaction
Types

Treatment of an
Induced B Layer
Moment

Algorithm 3.1: Classification of induced magnetic moments and coupling type determination.
This algorithm has been implemented in the code publication [338].

```

1 function getpairInteractionType ( $m_A, m_B, m_C$ );
2 Input : Three floats  $m_A, m_B$ , and  $m_C$ 
3 Output : Two ints pairIntTypeAB, pairIntTypeBC and & three bools activeA, activeB, activeC
4 cutoff  $\leftarrow 0.4$ ;
5 if  $|m_A| \leq \text{cutoff} \ \& \ |m_B| \leq \text{cutoff} \ \& \ |m_C| \leq \text{cutoff}$  then
6   pairIntTypeAB  $\leftarrow$  None ;
7   pairIntTypeBC  $\leftarrow$  None;
8 end if
9 if  $m_A == 0 \ || \ m_B == 0$  then
10   pairIntTypeAB  $\leftarrow$  None;
11 else
12   if  $(m_A > 0 \ \& \ m_B < 0) \ || \ (m_A < 0 \ \& \ m_B > 0)$  then
13     pairIntTypeAB  $\leftarrow -1$ ;
14   end if
15   if  $(m_A > 0 \ \& \ m_B > 0) \ || \ (m_A < 0 \ \& \ m_B < 0)$  then
16     pairIntTypeAB  $\leftarrow 1$ ;
17   end if
18 end if
19 if  $m_B == 0 \ || \ m_C == 0$  then
20   pairIntTypeBC  $\leftarrow$  None;
21 else
22   if  $(m_B > 0 \ \& \ m_C < 0) \ || \ (m_B < 0 \ \& \ m_C > 0)$  then
23     pairIntTypeBC  $\leftarrow -1$ ;
24   end if
25   if  $(m_B > 0 \ \& \ m_C > 0) \ || \ (m_B < 0 \ \& \ m_C < 0)$  then
26     pairIntTypeBC  $\leftarrow 1$ ;
27   end if
28 end if
29 if  $|m_A| > \text{cutoff}$  then
30   activeA  $\leftarrow$  True;
31 else
32   activeA  $\leftarrow$  False;
33 end if
34 if  $|m_B| > \text{cutoff}$  then
35   activeB  $\leftarrow$  True;
36 else
37   activeB  $\leftarrow$  False;
38 end if
39 if  $|m_C| > \text{cutoff}$  then
40   activeC  $\leftarrow$  True;
41 else
42   activeC  $\leftarrow$  False;
43 end if

```

A few different tasks are performed within the shown pseudocode Algorithm 3.1. These tasks include:

- Filtering film systems, for which no magnetic layer exceeds the set threshold of an induced magnetic moment (lines 5 to 8)
- Determine the pairwise layer interaction type for the layers A and B, by first checking if either layer is non-magnetic (which then are not further considered in this analysis) and then comparing the signs of the magnetic layer moments (lines 9 to 18)
- Determine the pairwise layer interaction type for layers B and C, by first checking if either layer is non-magnetic and then comparing the signs of the magnetic layer moments (lines 19 to 28)
- Determine which layers are inducing (active) or display induced magnetic moments (passive) for all three layers (lines 29 to 43)

Algorithm's
Capability to
Handle Thinner
Films

The presented pseudocode Algorithm 3.1 is compatible with films for which only two magnetic layers are present. The reason is that the magnetic moment of layer A, in this case, would have the value of $0 \mu_B$ in the extracted data. Hence, such a film is treated as a non-magnetic interaction between the A and B layers, causing it to be filtered out for the following considerations. However, the interaction between the B and C layers would be unaffected by this. Of course, there are no pair interactions with other magnetic layers for a film system with a single magnetic layer, which also causes these films to be filtered out for the same reason.

After determining the layer-wise pair interaction types, it is possible to relate the interaction types to the neighboring elements within the layers by counting the number of FM and AFM layer-wise pair interactions and visualizing the relative count of interactions together with the inducing and induced elements as in Figure 3.51.

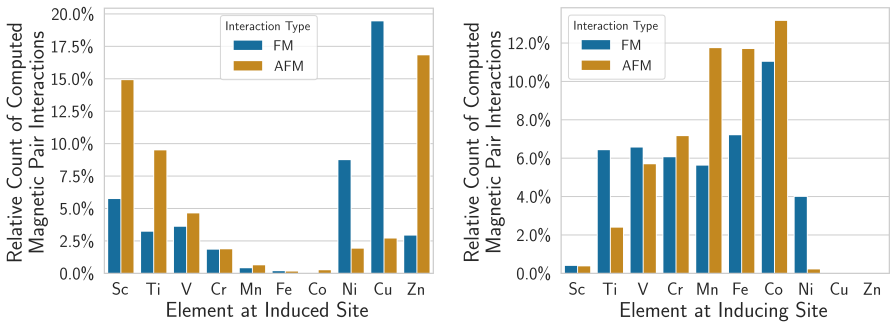


Figure 3.51: Relative count of FM and AFM layerwise pair interaction types for induced (left) and inducing (right) layer elements. This figure explicitly highlights the interaction type for inducing and induced sites. A magnetic pair interaction type overview without regard to whether an element is induced or inducing can be found in Figures 3.59, 3.60, and 3.61.

From Figure 3.51, it can be seen that the elements that typically would be considered non-magnetic in the bulk system are more likely to have an induced moment than to induce a magnetic moment. Examples of such elements are Sc, Ti, Cu, and Zn. These elements are comparably more often represented by induced magnetic moments. Furthermore, the elements that are known either for their inherent ferromagnetism, such as Fe and Co, or for the arrangement as AFM [345–347] ordering structures, such as Mn or Cr, are comparably more often inducing a moment. Surprisingly, even a few Fe and Co layers are contained within the data set for which the inherent magnetism is suppressed to the extent that they are considered induced moments according to the previously discussed definitions. Unsurprisingly, the typically non-magnetic elements Cu and Zn containing layers are not inducing a single neighboring magnetic moment as by the definitions previously introduced. For all induced moments, the AFM pair interaction count is increased compared to the FM count for all induced elements besides Ni and Cu, while for the inducing layer elements only Ti, V, and Ni display a magnetic interaction type more often. The depiction in Figure 3.51 can serve as a starting point for future materials design tasks in search of specific magnetic pair interaction types in films. As previously discussed, the interaction type is tuneable by the choice of film constituents. Therefore, knowing the magnetic interaction type can assist in creating an environment or platform which e.g. supports the emergence and stabilization of Skyrmions.

Observation of Elemental Tendencies to Gain Induced or Induce Magnetic Moments

Besides the induced moments in the magnetic layers, there are also induced magnetic moments in the different substrate layers, which are examined in the following.

Induced Magnetic Substrate Moments

As already seen, some elements are more likely to gain an induced moment than others. To determine if this observation from the induced magnetic layer elements translates to the substrate layer elements, the total induced magnetic substrate moments for each substrate element are shown in Figure 3.52.

Pd and Pt Average Total Induced Substrate Magnetic Moments Increased

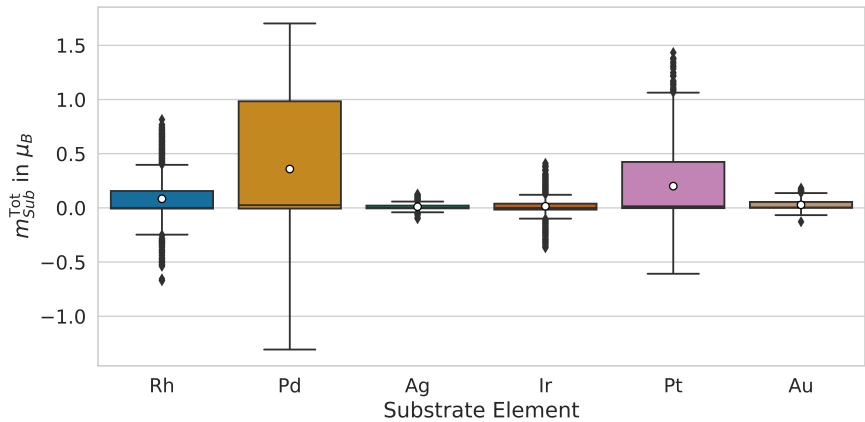


Figure 3.52: Depiction of the total substrate moment distribution for each substrate element. The horizontal lines in the boxes denote the median, while the white circle markers indicate the average of the individual substrate elements' magnetic moments.

From Figure 3.52, it can be seen that the average total magnetic substrate moment associated with the substrate elements Pd and Pt is significantly larger than the corresponding averages of the other substrate elements. Knowing that the average total film cell magnetic moments are roughly comparable for the different substrates from Figure 3.48, the observation from Figure 3.52 concludes that both palladium Pd and Pt comparably are either more likely to gain an induced magnetic moment than the other substrate elements or these particular elements tend to develop a magnetic ordering which causes the total moment to cancel out. For the substrate elements Ag, Ir, and Au, the distributions from Figure 3.52 show vanishing average total magnetic substrate moments. To determine if the magnetic ordering of Ag, Ir, and Au caused the total induced magnetic substrate moment to vanish, the induced magnetic moments of the individual substrate layers distribution for each substrate element needs to be examined and is hence depicted in Figure 3.53.

In the case of Rh, it can be seen from Figure 3.53 that the average substrate magnetic layer moment is positive in the substrate layer A but negative in substrate layers B and C, contributing to a smaller average total magnetic substrate moment. Also, for Ir, the average magnetic substrate moment in layers A and C is positive while at the same time negative in layer B. Hence, it is true that for Rh and Ir, the magnetic ordering contributes to a reduced element-specific total substrate moment. However, in the cases of the periodic table group 11 elements Ag and Au, the induced substrate moment averages indeed vanish for each layer individually for the B and C layer, while the corresponding average induced moments in the A layer are given by a small positive value. Examining the spread widths and averages of Pd and Pt-based substrate moments in combination with Figure 3.52, it can be concluded that both substrates have a tendency for an FM ordering within the substrate layers rather than an AFM configuration. Surprisingly, especially for Pd, it can be seen from Figure 3.53 that the average induced magnetic substrate moment in the layer A is the largest moment compared to the other layers for that specific element which seem to be comparably large. At the same time, by actually looking at the numbers, it turns out that $\overline{m}_{SubA}^{Pd} > \overline{m}_{SubC}^{Pd} > \overline{m}_{SubB}^{Pd}$ which is counter-intuitive as it would naively be assumed that the magnitude of the induced magnetic moment is decreasing for the substrate layers further away from the next magnetic layer. Further examining the average magnitudes of the substrate, layer-specific, induced magnetic moments for films with three magnetic layers—due to the previously discussed observation for Pd—it could be observed that the physical intuition that magnetic moments should decrease in magnitude within the substrate when progressing further into the substrate starting from the magnetic C layer indeed is not true—as can be seen in Figure 3.54—for both Pd and Pt, even more this assumption is also not true, when the focus is shifted from the magnetic moment's magnitude to the average induced magnetic substrate moments for only positively oriented (along the z -axis) FM substrate configurations for Pd and Pt. In the latter case, the induced moment's progression of Rh looks different from expected, observing the decreased ratios from substrate layer A to B, comparing it from layer B to C, and comparing the progression to e.g. the development of the curve for Ag and Au.

Average
Magnetic Pd
Substrate C
Layer Moment
Significantly
Increased
Compared to the
Average
Substrate B
Layer Moment

Physical Intuition
Suggests
Decreasing
Induced
Moments

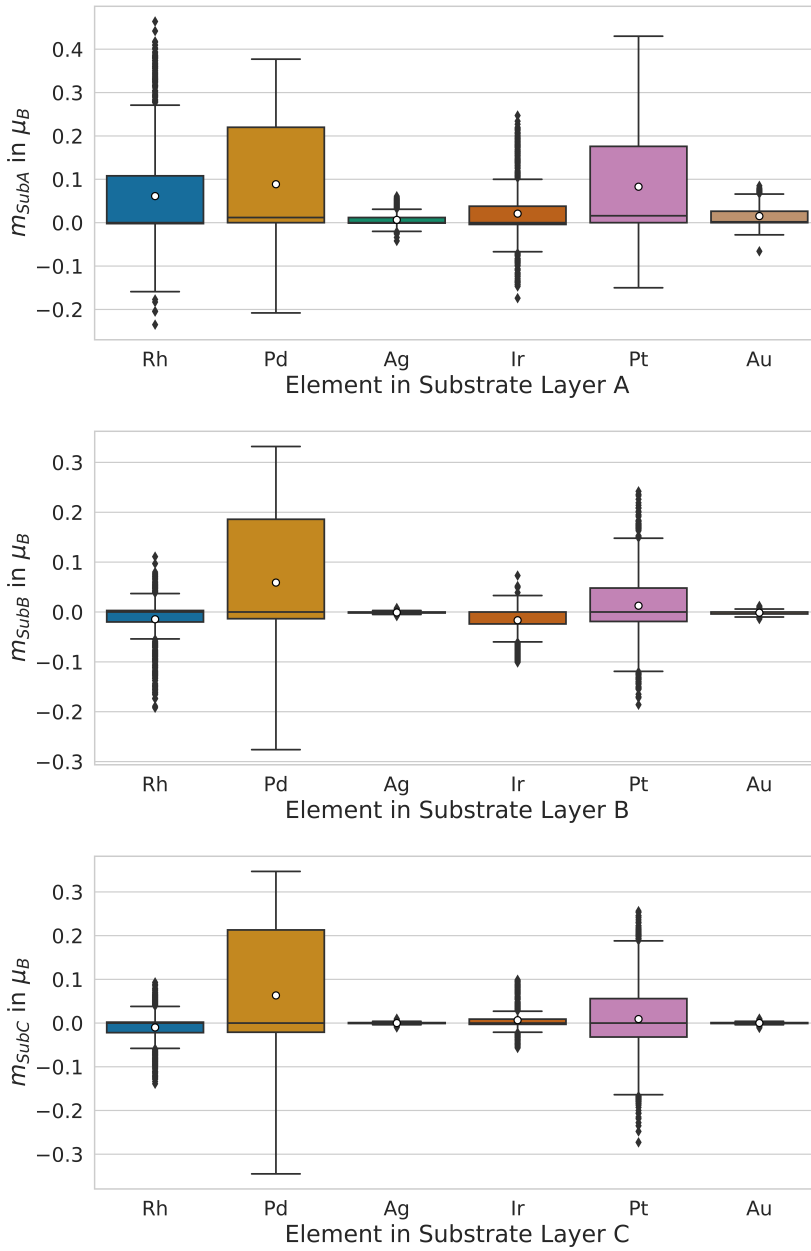


Figure 3.53: Depiction of the induced substrate magnetic moments for the substrate layers A (top), B (middle), and C (lower) for each substrate element. Depiction of the total substrate moment distribution for each substrate element. The horizontal lines in the boxes denote the median, while the white circle markers indicate the average of the individual substrate elements' magnetic moments.

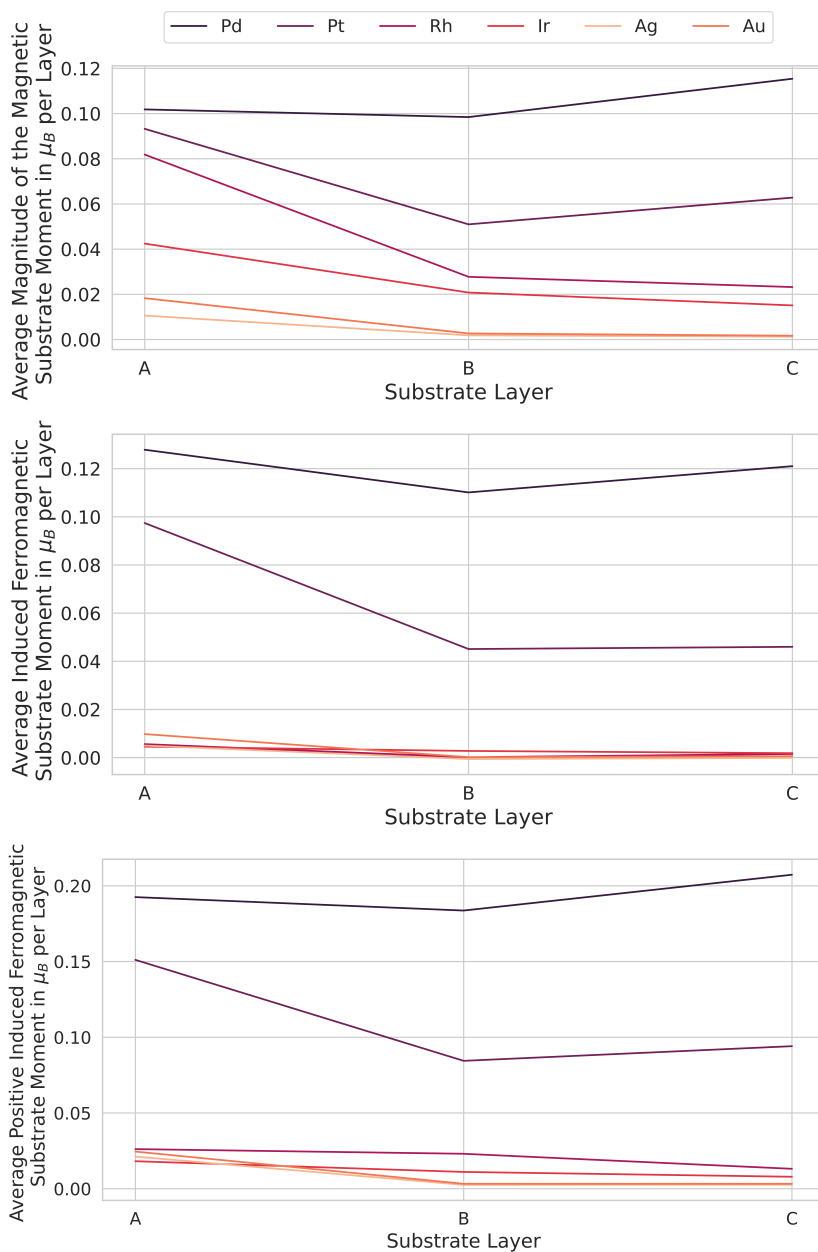


Figure 3.54: Development of the average induced magnetic moment magnitude (top), average induced magnetic moment for FM (as per algorithm 3.2) substrate orientations (middle), and average induced magnetic moment for FM (as per algorithm 3.2) substrate orientations with positive moment direction (lower) for each substrate element and substrate layer.

From Figure 3.54, it is apparent that both Pd and Pt represent significant deviations from the previously discussed intuition that the average magnitudes of the induced magnetic substrate moments should decrease towards the film's center for each of the three different cases. Examining those outliers and considering the observed effects emerge in a symmetric magnetic multilayer film setting, it is possible to explain this finding using the quantum confinement effect. [348] The finite film thickness forces the wave function to fulfill conditions with regard to the wave function wavelength. A special case of the quantum confinement effect is the quantum well, in which the electronic states are restricted to two dimensions instead of three. In a quantum well, states exist within a potential bordered at the edges of the quantum well by a higher potential, limiting the states' probability distribution to the space within the quantum well. In this case of symmetric multilayer films on a substrate, the boundaries of the quantum well would directly result from the limited total film thickness and potentially are also determined by the substrate's thickness. The individual quantum well states are affected by spin-polarization inequivalently. This can lead to different manifestations, including the occurrence of magnetic moment oscillations, in different element substrates. Knowing the substrate elements DOS, as presented in appendix D, it is clear that both Ag and Au have filled d states well below the Fermi level. Hence, the induced spin-polarization of these d states would not contribute to the magnetization and the induced moment. Furthermore, the DOS for these elements only has s states around the Fermi level, having a smaller induced magnetization and reduced overall DOS. Hence, according to the Stoner criterion [349, 350], which includes the consequence that a large DOS around the Fermi level benefits the emergence of ferromagnetism, it is apparent that Ag and Au, due to the fact that a larger DOS facilitates a larger response, will likely not exhibit significant magnetic spin-splitting. While for the Rh and Ir substrates, the d states are not found exclusively below the Fermi energy, the largest DOS of these elements is not found in close proximity to the Fermi level. However, for Pt, there is a DOS peak found relatively close to the Fermi level, indicating that a magnetic spin-splitting would be energetically more favored than fore previously discussed elements. In the case of Pd, the DOS peak close to the Fermi energy is the largest peak in the presented DOS. It has also been reported that bulk fcc Pd nearly meets the Stoner criterion [351–354] and hence is close to exhibiting ferromagnetism. While bulk Pd does not show FM behavior, quantum well states can cause the DOS to oscillate within the substrate layers, eventually leading to emerging magnetic ordering in the substrate. [352]

Also Pt Deviates
From the
Physical Intuition

Stoner Criterion

Quantum well states were previously observed for thin multilayer film systems [355] and also reported [352, 356, 357] to cause periodicity of the FM emerging in Pd thin film systems. However, at the time of writing, no study reports that there seems to exist an observable transition from extensive quantum well manifestation in Pd films over a reduced manifestation in Pt-based films to no apparent induced moment oscillation in Ag and Au thin films within the fcc noble metal group has been found in the literature. To describe the observed behavior first, in the following, the relation between the different induced magnetic substrate moments is shown in Figure 3.55.

Quantum Well
States in Thin
Film Systems

From Figure 3.55, it can be seen that the relation between m_{SubB} and m_{SubC} is indeed clearly of a linear type, while the relation between m_{SubB} and m_{SubA} is exhibiting a positive correlation it does not display linearity to that extent than the relation between m_{SubB} and m_{SubC} . However, in the case of m_{SubB} and m_{SubA} , the induced moment of the substrate layer A is mainly governed by the moment induced from the magnetic C layer rather than stemming from an interaction between the substrate layers such as the relation between m_{SubB} and m_{SubC} , as the substrate C layer is only neighbored by the substrate B layer elements.

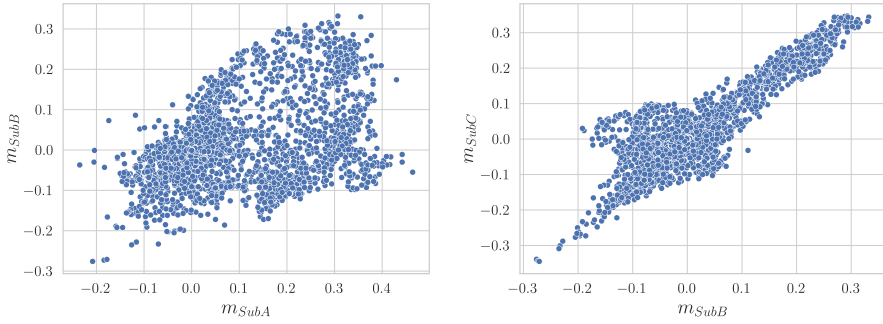


Figure 3.55: Scatter plot of the relation of the different neighboring magnetic induced substrate moments. The left depiction shows the relation between m_{SubB} to m_{SubA} , and the right depiction shows the relation between m_{SubC} to m_{SubB} .

Induced
Magnetic
Substrate
Moments as
Linear Response

Based on that observation, it is safe to assume that the magnetic moments of the substrate are a rather local phenomenon, as discussed previously in this section. This can be formalized, using a linear-response approach, in a sense that the induced moment of the substrate layer l of a—at first—non-symmetric film (as depicted in Figure 3.33), while l represents the number of layers—as a measure of distance—to the next magnetic layer, is solely determined by the external magnetic field at the corresponding layer position B_{ex}^l . The magnetic field B_{ex}^l at the layer l is an external field in the sense that it is itself stemming from the inducing neighbor layer $l-1$. At first, it is possible to relate the magnetic susceptibility χ to the local external magnetic field, as in equation (3.10). This expression can be concluded knowing that the present magnetic field is related to the magnetization by the magnetic susceptibility χ as it holds $m = \chi B$.

$$m_l \approx \chi B_{ex}^l \quad (3.10)$$

As the expression from equation (3.10) can be extended to the next nearest neighbor to the substrate layer l , equation (3.11) results. However, this assumes the locality of the magnetic moments. Locality in this context means that each magnetic moment induces a magnetic field only in the nearest neighboring layers but not in the next nearest neighbors.

$$m_l \approx \chi^2 B_{ex}^{l-1} m_{l-1} \quad (3.11)$$

The observation from the equations (3.10) and 3.11 is that this particular linear response approach can be performed inductively. Hence, the equations (3.10) and 3.11 generalize as shown in equation (3.12), where the induced magnetic moment of layer l is related to the magnetic moment for layer j , which is located more close to a magnetic layer but still within the substrate.

$$m_l \approx \chi^{l-j+1} m_j \quad (3.12)$$

Now, assuming that the layer j represents the first magnetic layer outside the substrate (the C layer in this case), equation (3.12) can be expressed as equation (3.13). This equation also relates the layers l and the magnetic layer C by their corresponding spatial distance $d_{l, MagC}$ using an exponential ansatz with a and b to be fitted to the individual previously presented data.

Exponential
Ansatz

$$m_l \approx \chi^l \chi m_C = a e^{-b d_{l, MagC}} \quad (3.13)$$

Now, considering the symmetry of the examined multilayer film systems and hence including the fact that the magnetic moment is induced into the substrate from both sides of the substrate this ansatz can be expanded. An expression for the magnetic moment in layer l , as given in equation (3.14), with the distances $d_{\text{MagC}\uparrow}$ and $d_{l,\text{MagC}\downarrow}$ denoting the distance from the substrate layer l to the magnetic layer located above (MagC $\uparrow$) and below (MagC $\downarrow$) the selected substrate layer l can be formulated.

$$m_l \approx a \left(e^{-bd_{l,\text{MagC}\uparrow}} + e^{-bd_{l,\text{MagC}\downarrow}} \right) \quad (3.14)$$

The expression presented in equation (3.14) is completely sufficient to model the average magnitude of the induced magnetic substrate moments of Ag, Au, Rh, and Ir almost perfectly. However, the expression from equation (3.14) fails to model the development of the average magnitude of the induced magnetic substrate moments for both Pd and Pt. Some oscillation occurs for the substrate elements originating from group 10 of the periodic table. From the publication [358] it was observable that the film's center layer—for quantum well oscillations—typically represents an extremum of the induced magnetic moment. Hence, it was decided to extend the ansatz presented previously by an oscillating cosine contribution, with the additional fitting parameters c and ω introduced. This approach is depicted in equation (3.15) and shifts the ansatz to be centered in line with the center of the substrate. Here, z denotes the spatial deviation from the substrate center—with D denoting the total thickness of the examined film—in either direction along the z -axis.

$$m(z) \approx a \left(e^{-b(z-\frac{D}{2})} + e^{-b(z+\frac{D}{2})} \right) (1 + c \cos(z\omega)) \quad (3.15)$$

It is found that using the expression from equation (3.15) does not only fit the average magnitudes of the magnetic induced substrate magnetic nearly perfectly for Pd and Pt, but also improved the already very good fits achieved using the expression from equation (3.14) for the substrate elements Ag, Au, Rh, and Ir, indicating that, while there is no apparent oscillation taking place, there might be small quantum well-originated contributions contained in the averaged data, despite that the contribution is far less significant than the quantum well effects visible in Pd and Pt films induced magnetic moment averages. Considering the different DOS of the substrate elements, which can be seen in appendix D, the observed properties of the electronic structure-related induced magnetic moment oscillations are expected to be distinct amongst the substrate elements as previously discussed. The results of the corresponding fits are shown in Figure 3.56 in a rearrangement of the data displayed in Figure 3.54.

Using the Film's
Symmetry

Average
Magnitudes of
Induced
Moments Well
Described for Ag,
Au, Rh, and Ir

Including
Oscillating
Contribution

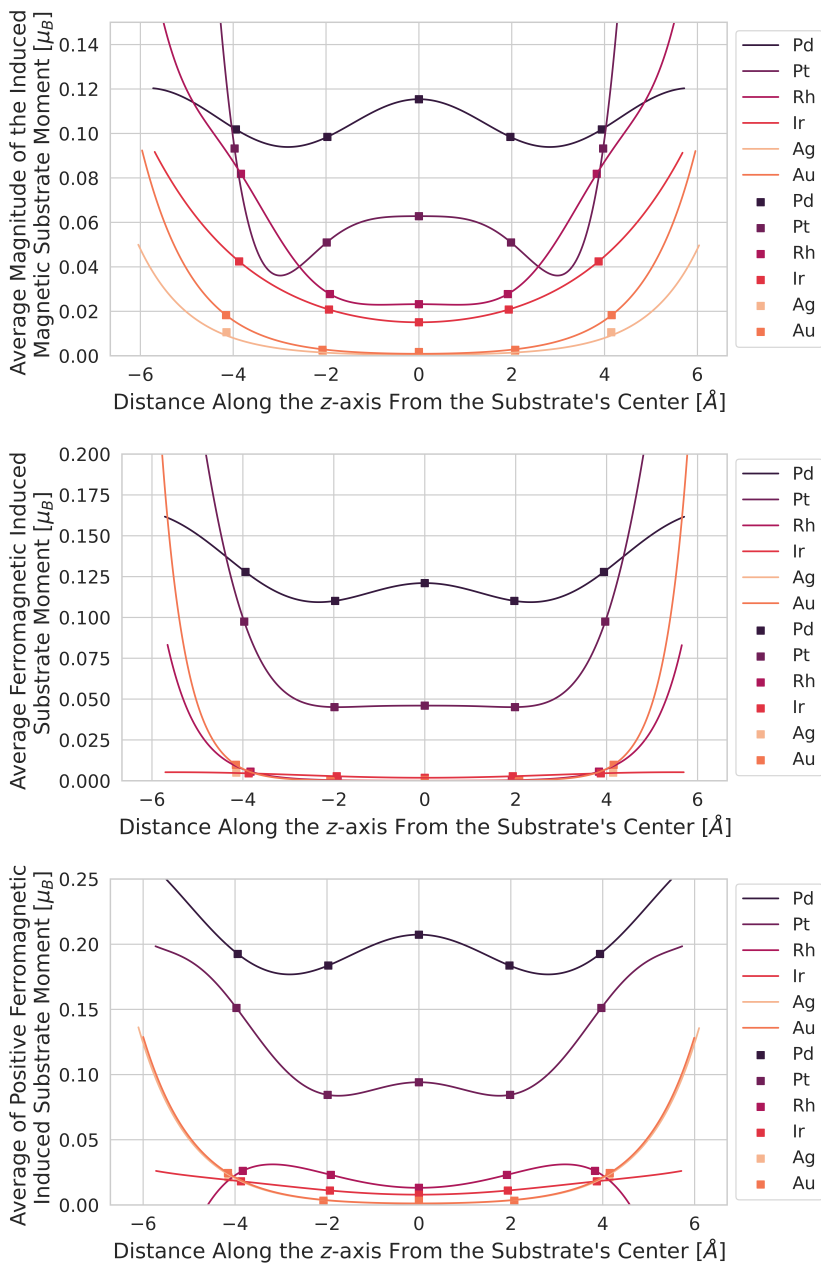


Figure 3.56: Depiction of the symmetrized data, depicted in Figure 3.54, including the corresponding fit function from equation (3.15), which describes the induced magnetic moment development inside the substrate. The fit functions are displayed from the position of the upper magnetic C layer to the position of the lower magnetic C layer.

In fact, from Figure 3.56, it can be seen that the chosen ansatz indeed models the induced magnetic substrate moments exceptionally well in all three examined data subsets. For the positively oriented FM configurations, the average moments of Rh depicts the previously mentioned remarkable behavior, which—in the depiction of Figure 3.56—turns out to be the exact opposite effect as displayed by Pd substrates which can be described by using a different sign, for the Rh-based substrates, of the fitted parameter c than used for the fitting of the Pd curve. Upon examination of the induced Rh substrate moments, it was observed that Rh does exhibit AFM coupling for some film configurations, which has been previously reported [359] to be related to Rh thickness and neighboring layer elements dependent interlayer exchange coupling. Notably, this Rh specific observation is averaged out as the examination is not restricted to a ferromagnetic ordering of the magnetic layer elements. This demonstrates that the fit function from equation (3.15) can describe the oscillations of the average induced substrate moment for the very different data subset representations.

Rh Displays
Oscillation
Adverse to Pd
and Pt

Resulting Magnetic Configurations

After discussing magnetic interaction types on a neighboring layer type basis, this section examines the magnetic ordering of the magnetic layers beyond the pair interactions. However, similarly to the pair interaction discussion, at first it is necessary to define what constitutes a FM, AFM, ferromagnetic, and non-magnetic configuration. For simplicity, the used definition's of the previously mentioned magnetic states configurations are briefly summarized in the following:

- In a FM configuration all three magnetic layer moments are parallel with the same sign of the magnetic moments
- In an AFM configuration the magnetic layer moments have altering signs—under vertical progression through the magnetic layers—attached to the magnetic moments
- Non-magnetic film systems are characterized by the fact that all magnetic moments magnitudes are situated below a threshold of $0.4 \mu_B$
- In a ferrimagnetic configuration the sum of absolute moments is exceeding the sum of the moments, which translates to the fact that any magnetic layer moment could be negative, without restricting to alternating signs of the moments, hence excluding such states which could otherwise be classified as AFM

These brief definitions can be found in the form of a pseudocode algorithm in Algorithm 3.2.

Algorithm 3.2: Magnetic state determination algorithm for the moment's orientation of the magnetic film layers. This algorithm's implementation [338] considers film systems with only two and a single magnetic moment layer to be present, respectively. The presented pseudocode algorithm can easily be expanded to those cases.

```

1 function getMagneticState ( $m_A, m_B, m_C$ );
2 Input : Three floats  $m_A, m_B$ , and  $m_C$ 
3 Output : String magneticState
4 cutoff  $\leftarrow 0.4$ ;
5 if  $|m_A| < \text{cutoff} \ \& \ |m_B| < \text{cutoff} \ \& \ |m_C| < \text{cutoff}$  then
6   magneticState  $\leftarrow$  "Non-Magnetic";
7 else
8   if  $(m_A < \text{cutoff} \ \& \ m_B < \text{cutoff} \ \& \ m_C < \text{cutoff}) \ || \ (m_A > \text{cutoff} \ \& \ m_B > \text{cutoff} \ \& \ m_C > \text{cutoff})$  then
9     magneticState  $\leftarrow$  "Ferromagnetic";
10  else
11    if  $(m_A < 0 \ \& \ m_B > 0 \ \& \ m_C < 0) \ || \ (m_A > 0 \ \& \ m_B < 0 \ \& \ m_C > 0)$  then
12      magneticState  $\leftarrow$  "Anti-Ferromagnetic";
13    else
14      if  $|m_A| + |m_B| + |m_C| < m_A + m_B + m_C$  then
15        magneticState  $\leftarrow$  "Ferrimagnetic";
16      end if
17    end if
18  end if
19 end if

```

Using Algorithm 3.2 to discriminate the different magnetic configurations emerging from the relaxed magnetic multilayer structures computed during this project and comparing the relative counts of the corresponding magnetic states for different iterations of DFT-integrated ML leads to the depiction in Figure 3.57.

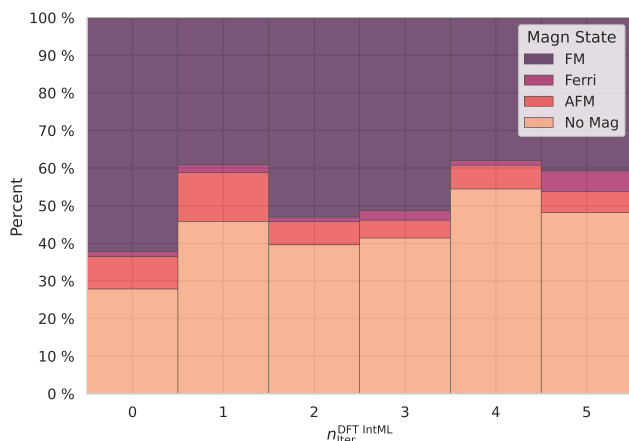


Figure 3.57: Overview over the magnetic configurations which successfully relaxed throughout different iterations of DFT-integrated ML. Each vertical column represents the entire relative distribution of the different DFT-integrated ML iterations.

Surprisingly, it is apparent that even though in the initial setup, no other magnetic configurations than ferromagnetism were set up as input of the Create-Magnetic-Film workflow, over 35 % of resulting magnetic configurations are not of the FM type. Furthermore, in each DFT-integrated ML iteration, all types of magnetic configurations are indeed present. However, examining the correlation between obtained relaxed magnetic configurations and the ML-suggested magnetic input states is not possible by the depiction in Figure 3.57. To examine the transitions from the ML-predicted magnetic moment configurations to the resulting magnetic states, the transition matrix shown in Figure 3.58 has been prepared.

Initially FM
Inputs Resulted
in Over 35 %
Non-FM
Configurations

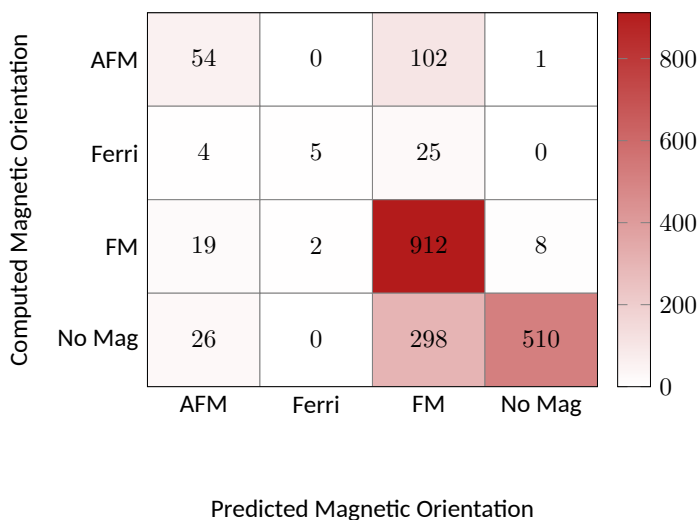


Figure 3.58: Transition matrix depiction the classification of the ML predicted magnetic states and the *ab initio* resulted magnetic configurations based on the classification presented in Algorithm 3.2.

Within Figure 3.58 all ML-based input moment predictions—across all DFT-integrated ML iterations, as the transition ratios for the initial setup can be derived from Figure 3.57—and the corresponding magnetic state classification emerging from Algorithm 3.2 has been compared to the *ab initio* resulting magnetic configurations, which also have been classified using the same algorithm. From Figure 3.58 it is found that over 75 % of the magnetic configurations, which result from the *ab initio* computations, have been correctly predicted using DFT-integrated ML, despite the ML models only learned to predict individual magnetic moments independently rather than taking the bigger picture into account to predict or consider the resulting magnetic state.

Setting the cutoff introduced in Algorithm 3.1 to $0 \mu_B$ and modifying the very same algorithm—for which the modified algorithm is displayed in appendix E—to also return a layer-wise pair interaction type (encoded as the integer 0) for the case in which either layer has a vanishing magnetic layer moment allows for a detailed insight into the relations between neighboring elemental layers and their respective interaction types. The heatmap depicting the FM interaction type counts for the A and B, as well as the B and C layer pairs, is shown in Figure 3.59. From Figure 3.59, it is apparent that the majority of FM interaction types are expectedly centered around the elements Mn, Fe, Co, and Ni and combinations amongst thereof. However, also Sc-Sc, Zn-Cu, and Sc-Zn pairs exhibit some degree of FM interaction according to Figure 3.59, while it should be noted that, given the elements involved in these combinations, the resulting magnetic moments magnitudes are likely not going to be large. In fact, in the B and C layer pairs, Cu, typically considered an element that does not exhibit strong magnetic properties, seems to strongly favor an FM interaction type with most of the other elements. The corresponding counts for the AFM interaction type are shown in Figure 3.60.

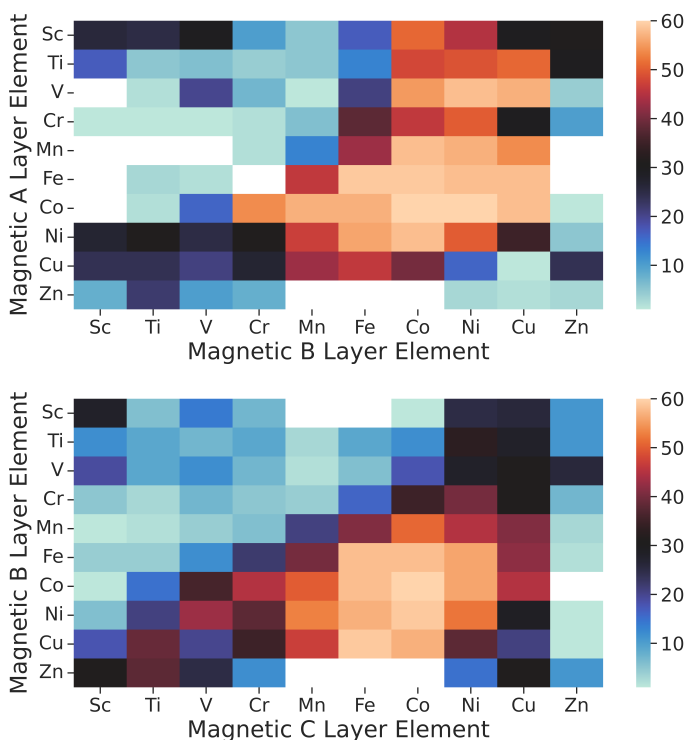


Figure 3.59: Heatmap of the pair interaction counts for the FM interaction type for the A and B layer pairs (top) and the B and C layer pairs (lower) resolved for the different elemental pairs. White square fields denote that such an interaction type is missing in the data set.

From Figure 3.60, it can be seen that a few elements tend to form AFM interaction types such as Zn, Ti, and Sc in both considered layer pairs. Particularly, Zn tends to especially form AFM types of interactions combined with layers consisting of Mn, Fe, and Co. It is also clear that Figure 3.60 misses all the elemental combinations that were contained in 3.59, displaying a significant tendency towards an FM pair interaction. A similar depiction containing the interactions for which either of the neighboring magnetic layer moments has the value of $0 \mu_B$ is shown in Figure 3.61.

Elements
Tendency to
AFM
Orientations

From Figure 3.61, it can be seen that for the A and B layer pairs, mainly the combinations of such pairs containing Zn, Cu, and Ni in the A layer exhibit vanishing magnetic moments. Of course, a typically non-magnetic layer on the outer side of the film paired with an element incapable of inducing a moment into the element of the outer layer results in an interaction type of vanishing magnetism. However, in that sense, the B and C layer non-magnetic pair interactions are more surprising in this regard. Between layers B and C, vanishing moments of some sort seem to occur very frequently as 3d elements with relatively small atomic numbers, such as Ti, V, and Cr neighbor each other.

Combinations of
Typically
Non-Magnetic
Elements Result
in Vanishing
Magnetism

This particular overview of pair interaction types—including the non-magnetic type—might be useful in the future (similarly as the discussion around the depiction of the FM and AFM induced moment types ratios from Figure 3.51) when a selection of materials is made to achieve a specific combination of magnetic interaction types.

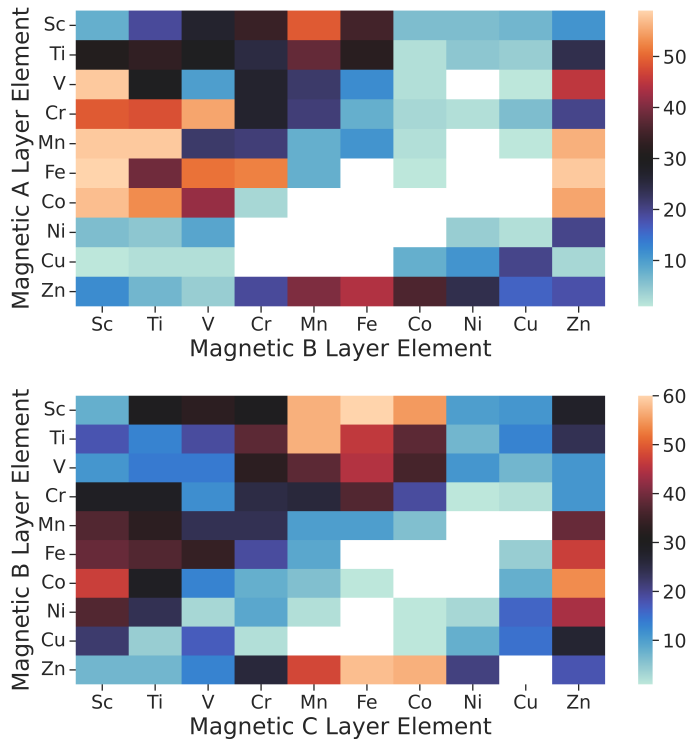


Figure 3.60: Heatmap of the pair interaction counts for the AFM interaction type for the A and B layer pairs (top) and the B and C layer pairs (lower) resolved for the different elemental pairs. White square fields denote that such an interaction type is missing in the data set.

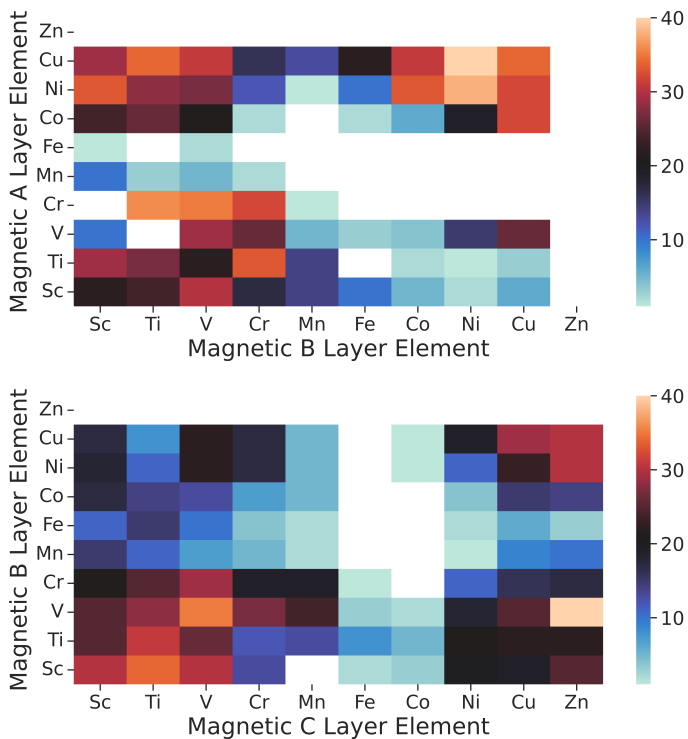


Figure 3.61: Heatmap of the pair interaction counts for the non-magnetic interaction type for the A and B layer pairs (top) and the B and C layer pairs (lower) resolved for the different elemental pairs. White square fields denote that such an interaction type is missing in the data set.

Summary & Outlook

In this thesis, a plurality of examples were examined in which data analytics methods and ML models add value to existing materials' science data and assist in the data collection process within a high-throughput setting. The different peculiarities associated with the corresponding data for each outlined modeling goal have been discussed in each of the presented examples. It has been illustrated how a materials screening problem associated with desirable material properties and property quantities could be approached. The power of ML methods lies in their ability to uncover meaningful relationships within data without the need for carefully tailored problem-specific models. Instead, machine learning methods rely on versatile, statistically well-founded frameworks, and learning algorithms that can adapt to new data sets. This approach has already begun to define the fourth scientific paradigm of data-driven science. However, contrary to popular belief, it was demonstrated across all projects engaged during this thesis that ML models complement the other scientific paradigms instead of replacing them. ML models allow for an alternative approach to a problem, as seen with modeling the critical temperature T_c for Heusler compounds. They can also assist in solving problems that traditionally are difficult to engage by using existing methods. An example of such a problem is the optimization of *ab initio* input parameters in a high-throughput context. By integrating ML models for the input parameter optimization into the workflow of a high-throughput DFT study, it was possible to successfully relax 45.5 % more film structures than without the use of ML models. This highlights the potential of combining established computational techniques with ML methods to improve performance and efficiency.

Determining the critical temperature of magnetic materials is a complex multistep process crucial for their design with practical applications in magnetic devices under operating conditions in mind. This material property essentially governs whether a material can maintain stable magnetism at room temperature conditions. During this project, an ensemble ML model was trained on DFT- and MC-based data of magnetic Heusler alloys achieving a classification test accuracy of up to 94 %, relying solely on the non-magnetic properties of the compounds as features. These features are available to researchers before an *ab initio* calculation is performed. Hence, this model would be well-suited for large-scale materials screening applications in the future, with a false negative classification rate on a test set of around 10 %, to explore further material properties of the class of Heusler alloys. Including magnetic—DFT-originated—information about the alloys, allowed to account for 85 % of the critical temperature's variance, using an ML regression model, for this typical materials science-sized data set of 408 structures.

Using the models, which achieved the outlined results, to explain the ML model learned relations using state-of-the-art XAI, it was possible to recover from the model's that the magnetic—or magnetism-associated—material properties had the most significant impact on the model's prediction. Hence, for either the prediction of T_c or—the classification—if a particular compound either has the material property of a critical temperature well above room temperature or not, this validates the physical intuition. Beyond applying the trained models in a materials screening application, it would be imaginable and desirable to extend the presented work to predict the critical temperatures of other material classes besides the Heusler alloys.

While the critical temperature itself is essential to the practical usability of a magnetic material, dedicated applications, such as spintronics-based data storage devices such as the MRAM, require the combination of multiple desirable material properties such as e.g. thermal magnetic stability and half-metallicity. Hence, as a next step within this project, the property of the spin-polarization at the Fermi level in full and inverse Heusler alloys was examined. Using existing data collected by collaborators to train, tune, and evaluate ensemble classification models, it was possible to successfully predict near half-metallicity for Heusler compounds ($L2_1$ Co_2HfIn , $\text{XA Mn}_2\text{TaGe}$, and $L2_1$ Co_2ScSn) for which this property was previously unpublished. For these predictions, publicly available *ab initio*-based structural and magnetic screening data—with an initial data amount of 4394 structures—has been systematically screened. In addition, the compounds that were classified to exhibit (near) half-metallicity by the ML classification model were further confirmed to be indeed (nearly) half-metallic through the application of FLAPW first-principles electronic structure calculations, thereby supporting the ML model's predictions. This resulted in a prediction precision of about 80 % using the Materials Project magnetic and structure data in the screening process. Furthermore, it was possible to partly observe that the ML model appears to have learned the physically established Slater-Pauling behavior. The known Slater-Pauling behavior itself would have correctly classified 7 structures amongst the 49 correctly ML predicted compounds, as such, with a significantly polarized DOS, as outlined in this thesis. This again emphasizes that ML models complement established physical models and known analytical relations to expand our knowledge of material properties further. This work could be continued by using the trained model for database screening—other than the Materials Project—to extend the number of screened Heusler structures. Furthermore, a generalization—requiring a significant extension of the training data set—of the presented approach to e.g. disordered Heusler, doped Heusler alloys, or half-Heusler alloys, such as in the previous T_c study would be possible. This would allow for combining both outlined Heusler-related projects, as it would be possible to screen for half-metallicity and room-temperature stable magnetism simultaneously and, therefore, extract the most promising material candidates for more detailed studies. Beyond Heusler alloys, an application of the presented methodology to other material classes would be imaginable.

The final project of this thesis aimed for a systematic high-throughput first-principles screening of symmetric thin magnetic multilayer film systems, which are theoretically well-known to represent tuneable host systems for room temperature stable Skyrmions, consisting of three layers of 3d transition metal elements on each side of a five atom layer thick fcc noble metal substrate. Within this study, due to the described setup, a phase space of 6660 possible film structures—also allowing film systems with only two and one magnetic layers on each side of the substrate layers—emerged. This large-scale, high-throughput study aimed first for the relaxation of the initialized structure—which was set up using average database bond length guesses for the ILDs and a fixed initial magnetic layer moment as initialization values—and subsequently examining the resulting magnetic properties. It was found that an ML-based optimization of the structural and magnetic input quantities was capable of improving the fraction of successfully relaxed film systems from 64.8 % to 94.3 %. This concludes that 31.3 % of the successfully relaxed film systems were successfully relaxed solely due to ML models using previously acquired *ab initio* data to optimize the setup. For the ML-predicted initial magnetic configurations, 75 % were confirmed by the subsequent DFT calculations to represent a valid magnetic ordering for the relaxed film structure. In fact, this approach of using ML to optimize input parameters solved a very prominent problem in high-throughput *ab initio* computational studies. It is current practice to approach convergence issues in a trial-and-error fashion, which lacks systematicity and computational efficiency. Furthermore, the use of ML-based input parameter optimization proved to have the potential of reducing the average number of needed relaxation steps by up to 29 % and simultaneously decreasing the average number of required total relaxation steps by up to 17 %. This is achieved, as on average, the ML prediction error of the magnetic layer moments is decreased by 60 % compared to the initial fixed guess, while the ILD prediction error is reduced by 50 % in comparison to the initial bond length ILD estimates. Beyond the substrate, the tendency of the 3d elements in the magnetic layers to either receive an induced magnetic moment or induce their neighboring layer with a magnetic moment has been examined. It was found that—consistent with the physical intuition—the upper and lower end of atomic numbers within the 3d transition metal group tend to gain an induced moment, while Mn, Fe, and Co tend to induce a magnetic moment into their neighboring layer significantly more often.

The developed methodology regarding the systematic and high-throughput compatible ML-based input parameter tuning could also be applied to quantities other than the ILDs and magnetic moments, representing both an input and an output of a DFT calculation. In principle, this methodology could be applied to the charge density itself to either boost convergence or provide a better starting density. This systematic study could be easily extended by increasing the number of magnetic layers placed on the substrate layers or by increasing the selection of magnetic layer elements to e.g. 4d or even 5d transition metal elements. Also, as fcc noble metal substrates have been examined, an extension to other substrates or even substrate structures such as e.g. hcp Ru and Os would be imaginable. Eventually, the published relaxed film structures could be used to extend the presented study by including antiparallel magnetic layer moments—on opposite substrate interface layers—in the initialization of the magnetic structures and the investigation of conductivity properties of different film setup configurations and compositions. This could also include the study of films with intermixing of layers, which can occur during the growth process of thin-film systems.

JuHemd LDA Data Overview

In order to complete the picture provided during the discussion of the JuHemd GGA data set, the T_c distribution for the LDA data set is given in Figure A.1.

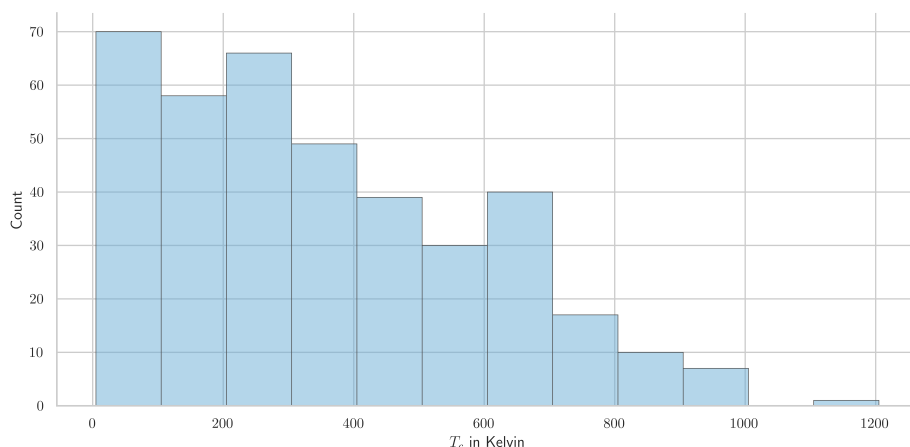


Figure A.1: Post-processing distribution of critical temperatures in the LDA data extracted from the JuHemd. The depiction complements the presented picture of JuHemd data from Figure 3.3.

From Figure A.1 it can be seen that the distribution of critical temperatures is slightly shifted towards the lower range when compared to the GGA critical temperature distribution. The GGA distribution of the site-specific atomic number occupations is shown in Figure A.2. It is apparent from Figure A.2 that the distribution of site-specific atomic numbers in the LDA data is very close to the GGA atomic numbers distribution.

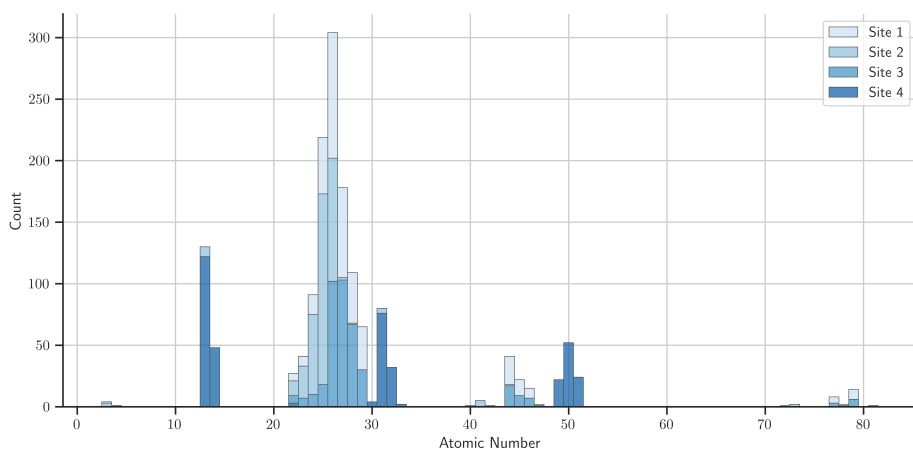


Figure A.2: Distribution of atomic numbers in the compounds extracted from the JuHemd contained, with color-coded lattice site positions, post-processing and after outlier removal. This depiction of the atomic To number distribution has been generated using the LDA data set. The depiction complements the presented picture of JuHemd data from Figure 3.4.

Bond Length Estimates

During the initial ILD estimation, average bond lengths computed using data from the materials science database operated by the Materials Project [94] have been used before implementing the DFT-integrated ML approach. To compute the ILDs for the film systems, the bond length estimation for all magnetic element combinations, magnetic and substrate element combinations, and the individual substrates must be known. The bond length estimates for all magnetic element combinations are displayed in Table B.1.

Elemental Pair	Sc	Ti	V	Cr	Mn	Fe	Co	Ni	Cu	Zn
Sc	3.27	2.99	3.04	2.85	2.87	2.89	2.86	2.74	2.82	2.91
Ti	2.99	2.91	2.71	2.82	2.78	2.57	2.52	2.60	2.81	2.82
V	3.04	2.71	2.7	2.49	2.49	2.58	2.60	2.61	2.59	2.69
Cr	2.85	2.82	2.49	2.56	2.44	2.47	2.52	2.46	2.55	2.57
Mn	2.87	2.78	2.49	2.44	2.48	2.50	2.47	2.52	2.62	2.51
Fe	2.89	2.57	2.58	2.47	2.50	2.58	2.46	2.53	2.53	2.66
Co	2.86	2.52	2.60	2.52	2.47	2.46	2.49	2.49	2.51	2.50
Ni	2.74	2.60	2.61	2.46	2.52	2.53	2.49	2.48	2.48	2.53
Cu	2.82	2.81	2.59	2.55	2.62	2.53	2.51	2.48	2.56	2.61
Zn	2.91	2.82	2.69	2.57	2.51	2.66	2.50	2.53	2.61	2.60

Table B.1: Bond length estimations of all magnetic element combinations within the given setup conditions. Whenever the bond length of an elemental bulk was available for the fcc lattice, this estimate was used on the diagonal of this table. All estimates from this table have the unit of Å.

The bond length estimates for combinations of magnetic and substrate elements are presented in Table B.2. The bond length estimates for the substrates (and substrate combinations) are provided in Table B.3.

Elemental Pair	Rh	Pd	Ag	Ir	Pt	Au
Sc	2.80	2.86	2.98	2.81	2.87	2.96
V	2.67	2.69	2.74	2.62	2.69	2.73
Ti	2.68	2.80	2.90	2.74	2.82	2.86
Cr	2.59	2.68	2.92	2.60	2.64	2.77
Mn	2.55	2.71	2.57	2.57	2.75	2.81
Fe	2.55	2.63	2.67	2.68	2.64	2.67
Co	2.66	2.68	2.71	2.58	2.69	2.80
Ni	2.59	2.64	2.60	2.69	2.72	2.60
Cu	2.64	2.72	2.67	2.64	2.79	2.68
Zn	2.71	2.63	2.88	2.65	2.78	2.72

Table B.2: Bond length estimations of all magnetic and substrate element combinations within the given setup conditions. All estimates from this table have the unit of Å.

Elemental Pair	Rh	Pd	Ag	Ir	Pt	Au
Rh	2.72	2.76	2.89	2.74	2.74	2.83
Pd	2.76	2.80	2.86	2.76	2.81	2.83
Ag	2.89	2.86	2.94	2.92	2.89	2.94
Ir	2.74	2.76	2.92	2.74	2.77	2.84
Pt	2.74	2.81	2.89	2.77	2.81	2.89
Au	2.83	2.83	2.94	2.84	2.89	2.95

Table B.3: Bond length estimations of all substrate element combinations within the given setup conditions. Whenever the bond length of an elemental bulk was available for the fcc substrate lattice, this estimate was used on this table's diagonal. All values from this table have the unit of Å.

However, the presented bond length estimates alone do not represent a reasonable guess for the ILDs. Rather, the estimates need to be scaled according to equation (B.1) to the geometry of the given problem. Within equation (B.1), the ILD d_{XY}^{ILD} between an upper layer X and a lower layer Y is determined. In this case Y could represent the substrate interface or a lower magnetic layer. This computation required the magnetic X element's bulk length estimate d_X^{Guess} , the substrate's bulk length estimate $d_{\text{Sub}}^{\text{Guess}}$, and the average computed bond length of the elements X and Y represented by d_{XY}^{Guess} . [11, 92]

$$d_{XY}^{\text{ILD}} = \frac{d_X^{\text{Guess}}}{d_{\text{Sub}}^{\text{Guess}}} \cdot \frac{d_{XY}^{\text{Guess}}}{\sqrt{2}} \quad (\text{B.1})$$

Also, it should be noted, that the initial bulk lattice constant used to set up the EOS computation of the substrates is given by the diagonal's in Table B.3.

Guess Error Development

Figure C.1 shows the development of the MAE calculated when comparing the initial ILD guesses, computed using average bond lengths, and the relaxed *ab initio* ILD.

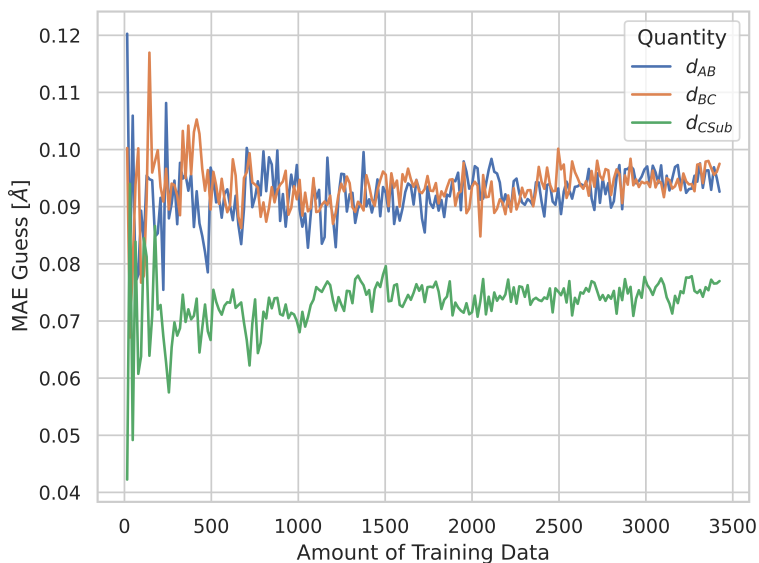


Figure C.1: Development of the MAE resulting from the comparison of the initial ILD guesses—used during the first two DFT-integrated ML iterations—and the relaxed *ab initio* ILDs for increasing amounts of available data. The ILD errors seen in this depiction have been used to visualize the relation displayed in Figure 3.41.

From Figure C.1 it is apparent that the error values do not exhibit an increasing or decreasing trend—which is expected as the guessing method is independent of the number of data points collected by the successful relaxations—in contrast to the ML-based input parameter optimization approach. It can also be seen that the MAE for the d_{CSub} ILD is reduced compared to the other. This can be explained by the fact that this particular ILD only has a non-relaxing substrate on one side, simplifying the relaxation for this specific layer. Furthermore, it can be seen in Figure C.2 that the error development is also relatively invariant regarding the amount of accumulated training data. Of course, there is some oscillation around the average taking place for both the ILDs and magnetic moment's error development. However, this oscillation is solely governed by the statistics of the chosen test set for which the error is evaluated for each amount of training data considered.

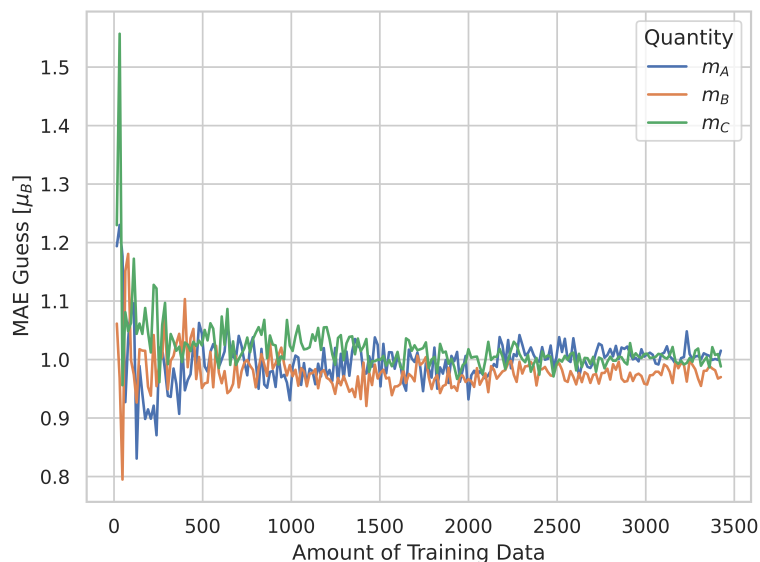


Figure C.2: Development of the MAE on a test set resulting from the comparison of the unanimously fixed initial magnetic moment guess—used during the first DFT-integrated ML iteration—and the resulting *ab initio* moments for increasing amounts of available data. The ILD errors seen in this depiction have been used to visualize the relation displayed in Figure 3.41.

Substrate Elements Density of States

Despite belonging to the group of fcc noble metals, the DOS of the elements used as film substrates in section 3.3 already indicate that the resulting film systems will have inequivalent properties. A key difference between the DOS of Ag and Au in comparison to the other elements' DOS, is that both fcc Ag and fcc Au have fully occupied d electron states, as can be seen in Figure D.3. For the remaining fcc noble metals, it can be seen in Figures D.1 and D.2 that the DOS exhibits a non-zero d PDOS at the Fermi level. Therefore, the d states cannot be fully occupied for these elements. The DOS computations, which were performed in order to obtain the displayed figures, follow the computation setup described for the FLAPW validation calculations in section 3.2.4 using a Fermi smearing of $\sigma = \frac{0.005}{3} \text{ Ha}$ and the twofold number of k-points. From the presented DOS depictions, it is apparent that for increasing periodic table group numbers, the corresponding element's d states are filled up.

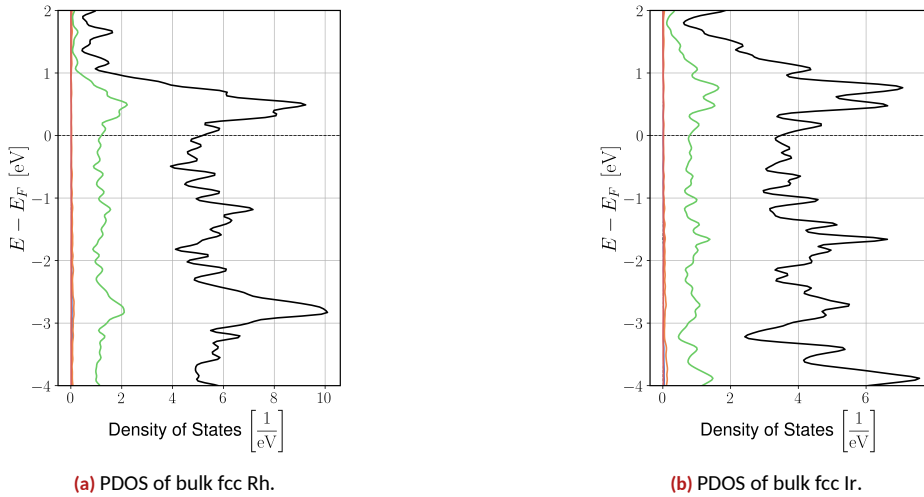
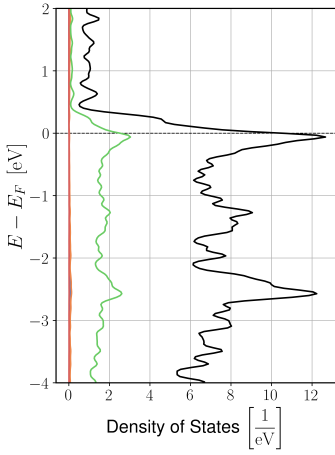
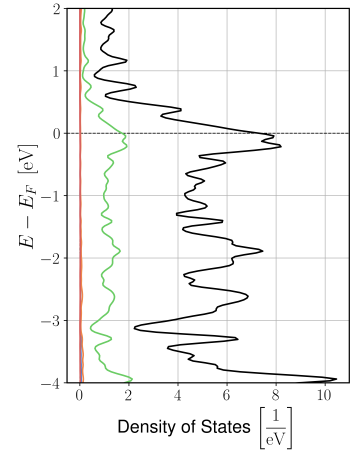


Figure D.1: PDOS of the fcc noble metals from group 9 of the periodic table. In this figure, the black line represents the total DOS of the corresponding fcc unit cell, and the green line represents the PDOS contribution to the total DOS of a single atom's d states.

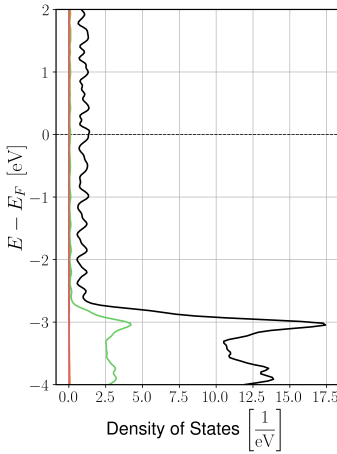


(a) PDOS of bulk fcc Pd.

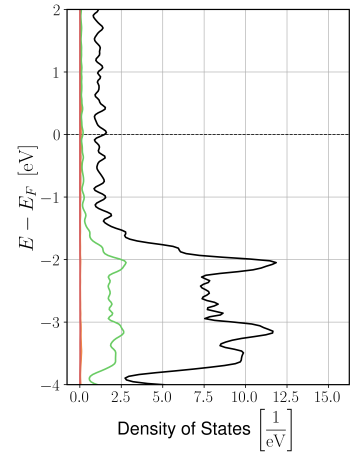


(b) PDOS of bulk fcc Pt.

Figure D.2: PDOS of the fcc noble metals from group 10 of the periodic table. In this figure, the black line represents the total DOS of the corresponding fcc unit cell, and the green line represents the PDOS contribution to the total DOS of a single atom's d states.



(a) PDOS of bulk fcc Ag.



(b) PDOS of bulk fcc Au.

Figure D.3: PDOS of the fcc noble metals from group 11 of the periodic table. In this figure, the black line represents the total DOS of the corresponding fcc unit cell, and the green line represents the PDOS contribution to the total DOS of a single atom's d states.

Modified Pair Interaction Algorithm

The modified algorithm, which was used, to compile the heatmap plots displaying the FM, AFM, and non-magnetic pair interaction counts amongst all relaxed film structures, collected during this thesis, is shown in Algorithm E.1.

Algorithm E.1: Modified algorithm (adapted from Algorithm 3.1) computing the magnetic layer pair interaction types, without regard of the active or passive magnetic moment induction and under consideration of non-magnetic layers as separate interaction type. [338]

```

1 function getpairInteractionTypeMod ( $m_A, m_B, m_C$ );
2 Input : Three floats  $m_A, m_B$ , and  $m_C$ 
3 Output : Two ints pairIntTypeAB and pairIntTypeAB
4 if ( $m_A > 0 \ \& \ m_B > 0$ ) || ( $m_A < 0 \ \& \ m_B < 0$ ) then
5 |   pairIntTypeAB  $\leftarrow$  1;
6 end if
7 if  $m_A == 0$  ||  $m_B == 0$  then
8 |   pairIntTypeAB  $\leftarrow$  0;
9 end if
10 if ( $m_A > 0 \ \& \ m_B < 0$ ) || ( $m_A < 0 \ \& \ m_B > 0$ ) then
11 |   pairIntTypeAB  $\leftarrow$  -1;
12 end if
13 if ( $m_B > 0 \ \& \ m_C > 0$ ) || ( $m_B < 0 \ \& \ m_C < 0$ ) then
14 |   pairIntTypeBC  $\leftarrow$  1;
15 end if
16 if  $m_B == 0$  ||  $m_C == 0$  then
17 |   pairIntTypeBC  $\leftarrow$  0;
18 end if
19 if ( $m_B > 0 \ \& \ m_C < 0$ ) || ( $m_B < 0 \ \& \ m_C > 0$ ) then
20 |   pairIntTypeBC  $\leftarrow$  -1;
21 end if

```

Bibliography

- [1] Anders S. G. Andrae and Tomas Edler. On Global Electricity Usage of Communication Technology: Trends to 2030. In: *Challenges* **6** (2015). DOI: [10.3390/challenges6010117](https://doi.org/10.3390/challenges6010117).
- [2] Weisheng S. Zhao et al. Magnetic Domain-Wall Racetrack Memory for high density and fast data storage. In: *2012 IEEE 11th International Conference on Solid-State and Integrated Circuit Technology*. 2012. DOI: [10.1109/ICSICT.2012.6466687](https://doi.org/10.1109/ICSICT.2012.6466687).
- [3] Markus Hoffmann et al. Skyrmion-Antiskyrmion Racetrack Memory in Rank-One DMI Materials. In: *Frontiers in Physics* **9** (2021). DOI: [10.3389/fphy.2021.769873](https://doi.org/10.3389/fphy.2021.769873).
- [4] Kyung Mee Song et al. Skyrmion-based artificial synapses for neuromorphic computing. In: *Nature Electronics* **3** (2020). DOI: [10.1038/s41928-020-0385-0](https://doi.org/10.1038/s41928-020-0385-0).
- [5] Xichao Zhang et al. Skyrmion-electronics: writing, deleting, reading and processing magnetic skyrmions toward spintronic applications. In: *Journal of Physics: Condensed Matter* **32** (2020). DOI: [10.1088/1361-648X/ab5488](https://doi.org/10.1088/1361-648X/ab5488).
- [6] Tomohiro Taniguchi et al. Reservoir computing based on spintronics technology. In: *Reservoir Computing: Theory, Physical Implementations, and Applications*. 2021. DOI: [10.1007/978-981-13-1687-6_14](https://doi.org/10.1007/978-981-13-1687-6_14).
- [7] Giovanni Finocchio et al. The promise of spintronics for unconventional computing. In: *Journal of Magnetism and Magnetic Materials* **521** (2021). DOI: [10.1016/j.jmmm.2020.167506](https://doi.org/10.1016/j.jmmm.2020.167506).
- [8] Wencong Jiang et al. Physical reservoir computing using magnetic skyrmion memristor and spin torque nano-oscillator. In: *Applied Physics Letters* **115** (2019). DOI: [10.1063/1.5115183](https://doi.org/10.1063/1.5115183).
- [9] Sebastiaan P. Huber et al. AiiDA 1.0, a scalable computational infrastructure for automated reproducible workflows and data provenance. In: *Scientific Data* **7** (2020). DOI: [10.1038/s41597-020-00638-4](https://doi.org/10.1038/s41597-020-00638-4).
- [10] Martin Uhrin et al. Workflows in AiiDA: Engineering a high-throughput, event-based engine for robust and modular computational workflows. In: *Computational Materials Science* **187** (2021). DOI: [10.1016/j.commatsci.2020.110086](https://doi.org/10.1016/j.commatsci.2020.110086).
- [11] Jens Bröder et al. JuDFTteam/aiida-fleur: AiiDA-FLEUR. Zenodo. 2023. DOI: [10.5281/ZENODO.7893716](https://doi.org/10.5281/ZENODO.7893716).
- [12] Stefano Sanvito et al. Accelerated discovery of new magnets in the Heusler alloy family. In: *Science Advances* **3** (2017). DOI: [10.1126/sciadv.1602241](https://doi.org/10.1126/sciadv.1602241).
- [13] Stan Lipovetsky and Michael Conklin. Analysis of regression in game theory approach. In: *Applied Stochastic Models in Business and Industry* **17** (2001). DOI: [10.1002/asmb.446](https://doi.org/10.1002/asmb.446).

- [14] Roman Kováčik, Phivos Mavropoulos, and Stefan Blügel. The JuHemd (Jülich-Heusler-magnetic-database) of the Monte Carlo simulated critical temperatures of the magnetic phase transition for experimentally reported Heusler and Heusler-like materials. Materials Cloud Archive. 2022. DOI: [10.24435/MATERIALSCLOUD:WW-PV](https://doi.org/10.24435/MATERIALSCLOUD:WW-PV).
- [15] Volker L. Deringer et al. Realistic Atomistic Structure of Amorphous Silicon from Machine-Learning-Driven Molecular Dynamics. In: *The Journal of Physical Chemistry Letters* **9** (2018). DOI: [10.1021/acs.jpclett.8b00902](https://doi.org/10.1021/acs.jpclett.8b00902).
- [16] Jonathan Chico et al. First-principles studies of the Gilbert damping and exchange interactions for half-metallic Heuslers alloys. In: *Phys. Rev. B* **93** (2016). DOI: [10.1103/PhysRevB.93.214439](https://doi.org/10.1103/PhysRevB.93.214439).
- [17] Denis Comtesse et al. First-principles study of spin-dependent thermoelectric properties of half-metallic Heusler thin films between platinum leads. In: *Phys. Rev. B* **89** (2014). DOI: [10.1103/PhysRevB.89.094410](https://doi.org/10.1103/PhysRevB.89.094410).
- [18] Jodi M. Iwata-Harms et al. High-temperature thermal stability driven by magnetization dilution in CoFeB free layers for spin-transfer-torque magnetic random access memory. In: *Scientific Reports* **8** (2018). DOI: [10.1038/s41598-018-32641-6](https://doi.org/10.1038/s41598-018-32641-6).
- [19] Igor E. Dzyaloshinsky. A thermodynamic theory of “weak” ferromagnetism of antiferromagnetics. In: *Journal of Physics and Chemistry of Solids* **4** (1958). DOI: [10.1016/0022-3697\(58\)90076-3](https://doi.org/10.1016/0022-3697(58)90076-3).
- [20] Tōru Moriya. Anisotropic Superexchange Interaction and Weak Ferromagnetism. In: *Phys. Rev.* **120** (1960). DOI: [10.1103/PhysRev.120.91](https://doi.org/10.1103/PhysRev.120.91).
- [21] Gerardo H. O. Daalderop, Paul J. Kelly, and Martin F. H. Schuurmans. First-principles calculation of the magnetocrystalline anisotropy energy of iron, cobalt, and nickel. In: *Phys. Rev. B* **41** (1990). DOI: [10.1103/PhysRevB.41.11919](https://doi.org/10.1103/PhysRevB.41.11919).
- [22] Dieter Krause. Über die magnetische Anisotropieenergie kubischer Kristalle. In: *physica status solidi (b)* **6** (1964). DOI: [10.1002/pssb.19640060110](https://doi.org/10.1002/pssb.19640060110).
- [23] Teng Zhou, Zhen Song, and Kai Sundmacher. Big Data Creates New Opportunities for Materials Research: A Review on Methods and Applications of Machine Learning for Materials Design. In: *Engineering* **5** (2019). DOI: [10.1016/j.eng.2019.02.011](https://doi.org/10.1016/j.eng.2019.02.011).
- [24] Heather J. Kulik et al. Roadmap on Machine learning in electronic structure. In: *Electronic Structure* **4** (2022). DOI: [10.1088/2516-1075/ac572f](https://doi.org/10.1088/2516-1075/ac572f).
- [25] Yasuhiko Igarashi et al. Three levels of data-driven science. In: *Journal of Physics: Conference Series* **699** (2016). DOI: [10.1088/1742-6596/699/1/012001](https://doi.org/10.1088/1742-6596/699/1/012001).
- [26] Danette McGilvray. Executing data quality projects: Ten steps to quality data and trusted information. San Diego, CA, USA: Academic Press, 2021. ISBN: 978-0-12-818015-0.
- [27] Francesco Sarracino and Malgorzata Mikucka. Bias and efficiency loss in regression estimates due to duplicated observations: a Monte Carlo simulation. en. In: *Survey Research Methods* **Vol 11** (2017). DOI: [10.18148/SRM/2017.V11I1.7149](https://doi.org/10.18148/SRM/2017.V11I1.7149).
- [28] Robert Ferber. Are Correlations any Guide to Predictive Value? In: *Applied Statistics* **5** (1956). DOI: [10.2307/2985494](https://doi.org/10.2307/2985494).
- [29] Florian Wetschoreck, Tobias Krabel, and Surya Krishnamurthy. 8080labs/ppscore: zenodo release. Zenodo. 2020. DOI: [10.5281/ZENODO.4091345](https://doi.org/10.5281/ZENODO.4091345).

- [30] Robin Hilgers. Extension of FLAPW Method to Fully-unconstrained Treatment of Non-collinear Magnetism in the Muffin-tin Sphere. MA thesis. RWTH Aachen University, 2020.
- [31] Ernst Schrödinger. An Undulatory Theory of the Mechanics of Atoms and Molecules. In: *Phys. Rev.* **28** (1926). DOI: [10.1103/PhysRev.28.1049](https://doi.org/10.1103/PhysRev.28.1049).
- [32] Klaus Capelle. A bird's-eye view of density-functional theory. In: *Brazilian Journal of Physics* **36** (2006). DOI: [10.1590/s0103-97332006000700035](https://doi.org/10.1590/s0103-97332006000700035).
- [33] Feliciano Giustino. Materials Modelling Using Density Functional Theory: Properties and Predictions. London, UK: Oxford University Press, 2014. ISBN: 978-0-19-966244-9.
- [34] Pierre Hohenberg and Walter Kohn. Inhomogeneous Electron Gas. In: *Phys. Rev.* **136** (1964). DOI: [10.1103/PhysRev.136.B864](https://doi.org/10.1103/PhysRev.136.B864).
- [35] Max Born and J. Robert Oppenheimer. Zur Quantentheorie der Molekeln. In: *Annalen der Physik* **389** (1927). DOI: [10.1002/andp.19273892002](https://doi.org/10.1002/andp.19273892002).
- [36] Richard M. Martin. Electronic Structure: Basic Theory and Practical Methods. Cambridge, UK: Cambridge University Press, 2004. DOI: [10.1017/CB09780511805769](https://doi.org/10.1017/CB09780511805769).
- [37] Reiner M. Dreizler and Eberhard Engel. Density Functional Theory: An Advanced Course. Berlin, Germany: Springer, 2011. DOI: [10.1007/978-3-642-14090-7](https://doi.org/10.1007/978-3-642-14090-7).
- [38] Robert G. Parr and Yang Weitao. Density-Functional Theory of Atoms and Molecules. International Series of Monographs on Chemistry. London, UK: Oxford University Press, 1994. ISBN: 978-0-19-535773-8.
- [39] Philipp Kurz. Non-collinear magnetism at surfaces and in ultrathin films. PhD thesis. RWTH Aachen University, 2000.
- [40] Walter Kohn and Lu Jeu Sham. Self-Consistent Equations Including Exchange and Correlation Effects. In: *Phys. Rev.* **140** (1965). DOI: [10.1103/PhysRev.140.A1133](https://doi.org/10.1103/PhysRev.140.A1133).
- [41] Ole Krogh Andersen. Linear methods in band theory. In: *Phys. Rev. B* **12** (1975). DOI: [10.1103/PhysRevB.12.3060](https://doi.org/10.1103/PhysRevB.12.3060).
- [42] Bradley Efron et al. Least angle regression. In: *The Annals of Statistics* **32** (2004). DOI: [10.1214/0090536040000000067](https://doi.org/10.1214/0090536040000000067).
- [43] Lev Kantorovich. Quantum Theory of the Solid State: An Introduction. Dordrecht, Netherlands: Springer, 2004. DOI: [10.1007/978-1-4020-2154-1](https://doi.org/10.1007/978-1-4020-2154-1).
- [44] Stefan Blügel and Gustav Bihlmayer. Full-Potential Linearized Augmented Planewave Method. In: *NIC Series*. **31**. Jülich, Germany: Forschungszentrum Jülich, 2006. ISBN: 3-00-017350-1.
- [45] Erich Wimmer et al. Full-potential self-consistent linearized-augmented-plane-wave method for calculating the electronic structure of molecules and surfaces: O₂ molecule. In: *Phys. Rev. B* **24** (1981). DOI: [10.1103/PhysRevB.24.864](https://doi.org/10.1103/PhysRevB.24.864).
- [46] Henry Krakauer, Michel Posternak, and Arthur J. Freeman. Linearized augmented plane-wave method for the electronic band structure of thin films. In: *Phys. Rev. B* **19** (1979). DOI: [10.1103/PhysRevB.19.1706](https://doi.org/10.1103/PhysRevB.19.1706).
- [47] Philipp Kurz et al. Ab initio treatment of noncollinear magnets with the full-potential linearized augmented plane wave method. In: *Phys. Rev. B* **69** (2004). DOI: [10.1103/PhysRevB.69.024415](https://doi.org/10.1103/PhysRevB.69.024415).
- [48] Daniel Wortmann et al. FLEUR. Zenodo. 2023. DOI: [10.5281/ZENODO.7576163](https://doi.org/10.5281/ZENODO.7576163).

- [49] The FLEUR project. <https://www.flapw.de/>.
- [50] Henning Janssen et al. JuDFTteam/masci-tools: Material science tools. Zenodo. 2023. DOI: [10.5281/ZENODO.7892182](https://doi.org/10.5281/ZENODO.7892182).
- [51] Miriam Winkelmann et al. Kerker mixing scheme for self-consistent muffin-tin based all-electron electronic structure calculations. In: *Phys. Rev. B* **102** (2020). DOI: [10.1103/PhysRevB.102.195138](https://doi.org/10.1103/PhysRevB.102.195138).
- [52] Herbert Amann and Joachim Escher. Analysis III. 2nd ed. Grundstudium Mathematik. Basel, Switzerland: Birkhauser Verlag AG, 2008. DOI: [10.1007/978-3-7643-8884-3](https://doi.org/10.1007/978-3-7643-8884-3).
- [53] Charles G. Broyden. A class of methods for solving nonlinear simultaneous equations. In: *Mathematics of Computation* **19** (1965). DOI: [10.1090/s0025-5718-1965-0198670-6](https://doi.org/10.1090/s0025-5718-1965-0198670-6).
- [54] G. P. Kerker. Efficient iteration scheme for self-consistent pseudopotential calculations. In: *Phys. Rev. B* **23** (1981). DOI: [10.1103/PhysRevB.23.3082](https://doi.org/10.1103/PhysRevB.23.3082).
- [55] Robin Hilgers, Daniel Wortmann, and Stefan Blügel. Application of batch learning for boosting high-throughput *ab initio* success rates and reducing computational effort required using data-driven processes. In: *arXiv: cond-mat.mtrl-sci* (2023). Preprint. DOI: [10.48550/arXiv.2311.15430](https://doi.org/10.48550/arXiv.2311.15430).
- [56] Charles G. Broyden. The Convergence of a Class of Double-rank Minimization Algorithms 1. General Considerations. In: *IMA Journal of Applied Mathematics* **6** (1970). DOI: [10.1093/iamamat/6.1.76](https://doi.org/10.1093/iamamat/6.1.76).
- [57] Roger Fletcher. A new approach to variable metric algorithms. In: *The Computer Journal* **13** (1970). DOI: [10.1093/comjnl/13.3.317](https://doi.org/10.1093/comjnl/13.3.317).
- [58] Donald Goldfarb. A family of variable-metric methods derived by variational means. In: *Mathematics of Computation* **24** (1970). DOI: [10.1090/s0025-5718-1970-0258249-6](https://doi.org/10.1090/s0025-5718-1970-0258249-6).
- [59] David F. Shanno. Conditioning of quasi-Newton methods for function minimization. In: *Mathematics of Computation* **24** (1970). DOI: [10.1090/s0025-5718-1970-0274029-x](https://doi.org/10.1090/s0025-5718-1970-0274029-x).
- [60] Jan Korringa. On the calculation of the energy of a Bloch wave in a metal. In: *Physica* **13** (1947). DOI: [10.1016/0031-8914\(47\)90013-X](https://doi.org/10.1016/0031-8914(47)90013-X).
- [61] Walter Kohn and Norman Rostoker. Solution of the Schrödinger Equation in Periodic Lattices with an Application to Metallic Lithium. In: *Phys. Rev.* **94** (1954). DOI: [10.1103/PhysRev.94.1111](https://doi.org/10.1103/PhysRev.94.1111).
- [62] Philipp Rüßmann et al. JuDFTteam/JuKKR: v3.6. Zenodo. 2022. DOI: [10.5281/ZENODO.7284738](https://doi.org/10.5281/ZENODO.7284738).
- [63] Philipp Rüßmann and Jens Broeder. JuDFTteam/aiida-kkr v1.1.10. Zenodo. 2020. DOI: [10.5281/ZENODO.3628251](https://doi.org/10.5281/ZENODO.3628251).
- [64] Hubert Ebert, Diemo Ködderitzsch, and Ján Minár. Calculating condensed matter properties using the KKR-Green's function method—recent developments and applications. In: *Reports on Progress in Physics* **74** (2011). DOI: [10.1088/0034-4885/74/9/096501](https://doi.org/10.1088/0034-4885/74/9/096501).
- [65] Alexander I. Liechtenstein et al. Local spin density functional approach to the theory of exchange interactions in ferromagnetic metals and alloys. In: *Journal of Magnetism and Magnetic Materials* **67** (1987). DOI: [10.1016/0304-8853\(87\)90721-9](https://doi.org/10.1016/0304-8853(87)90721-9).

- [66] Ondřej Šipr et al. Magnetic moments, exchange coupling, and crossover temperatures of Co clusters on Pt(111) and Au(111). In: *Journal of Physics: Condensed Matter* **19** (2007). DOI: [10.1088/0953-8984/19/9/096203](https://doi.org/10.1088/0953-8984/19/9/096203).
- [67] Hubert Ebert and Sergiy Mankovsky. Anisotropic exchange coupling in diluted magnetic semiconductors: Ab initio spin-density functional theory. In: *Phys. Rev. B* **79** (2009). DOI: [10.1103/PhysRevB.79.045209](https://doi.org/10.1103/PhysRevB.79.045209).
- [68] László Udvardi et al. In: *Phys. Rev. B* **68** (2003). DOI: [10.1103/PhysRevB.68.104436](https://doi.org/10.1103/PhysRevB.68.104436).
- [69] Philipp Rüßmann. Spin scattering of topologically protected electrons at defects. Dissertation. RWTH Aachen University, 2018. DOI: [10.18154/RWTH-2018-226738](https://doi.org/10.18154/RWTH-2018-226738).
- [70] Jia Yan Law et al. A quantitative criterion for determining the order of magnetic phase transitions using the magnetocaloric effect. In: *Nature Communications* **9** (2018). DOI: [10.1038/s41467-018-05111-w](https://doi.org/10.1038/s41467-018-05111-w).
- [71] Flaviano J. dos Santos, Manuel dos Santos Dias, and Samir Lounis. Modeling spin waves in noncollinear antiferromagnets: Spin-flop states, spin spirals, skyrmions, and antiskyrmions. In: *Phys. Rev. B* **102** (2020). DOI: [10.1103/PhysRevB.102.104436](https://doi.org/10.1103/PhysRevB.102.104436).
- [72] Marie Hervé et al. Stabilizing spin spirals and isolated skyrmions at low magnetic field exploiting vanishing magnetic anisotropy. In: *Nature Communications* **9** (2018). DOI: [10.1038/s41467-018-03240-w](https://doi.org/10.1038/s41467-018-03240-w).
- [73] John Bardeen, Leon N. Cooper, and John R. Schrieffer. Theory of Superconductivity. In: *Phys. Rev.* **108** (1957). DOI: [10.1103/PhysRev.108.1175](https://doi.org/10.1103/PhysRev.108.1175).
- [74] Yong Liu, Shyamal Kumar Bose, and Josef Kudrnovský. Half-metallicity and magnetism of GeTe doped with transition metals V, Cr, and Mn: A theoretical study from the viewpoint of application in spintronics. In: *Journal of Applied Physics* **112** (2012). DOI: <https://doi.org/10.1063/1.4750031>.
- [75] Iosif Galanakis, Peter H. Dederichs, and Nikos Papanikolaou. Slater-Pauling behavior and origin of the half-metallicity of the full-Heusler alloys. In: *Physical Review B* **66** (2002). DOI: [10.1103/PhysRevB.66.174429](https://doi.org/10.1103/PhysRevB.66.174429).
- [76] Iosif Galanakis. Appearance of half-metallicity in the quaternary Heusler alloys. In: *Journal of Physics: Condensed Matter* **16** (2004). DOI: [10.1088/0953-8984/16/18/010](https://doi.org/10.1088/0953-8984/16/18/010).
- [77] Martin Jourdan et al. Direct observation of half-metallicity in the Heusler compound Co₂MnSi. In: *Nature communications* **5** (2014). DOI: <https://doi.org/10.1038/ncomms4974>.
- [78] Ernst Ising. Beitrag zur Theorie des Ferromagnetismus. In: *Zeitschrift für Physik* **31** (1925). DOI: [10.1007/bf02980577](https://doi.org/10.1007/bf02980577).
- [79] Roman Brinzanik. Monte Carlo Study of Magnetic Nanostructures During Growth. PhD thesis. Freie Universität Berlin, 2003. DOI: [10.17169/REFUBIUM-9751](https://doi.org/10.17169/REFUBIUM-9751).
- [80] David S. G. Bauer et al. Thermally activated magnetization reversal in monatomic magnetic chains on surfaces studied by classical atomistic spin-dynamics simulations. In: *Journal of Physics: Condensed Matter* **23** (2011). DOI: [10.1088/0953-8984/23/39/394204](https://doi.org/10.1088/0953-8984/23/39/394204).
- [81] Sergii Grytsiuk et al. Ab initio analysis of magnetic properties of the prototype B20 chiral magnet FeGe. In: *Phys. Rev. B* **100** (2019). DOI: [10.1103/PhysRevB.100.214406](https://doi.org/10.1103/PhysRevB.100.214406).

- [82] Marjana Ležaić et al. Exchange interactions and local-moment fluctuation corrections in ferromagnets at finite temperatures based on noncollinear density-functional calculations. In: *Phys. Rev. B* **88** (2013). DOI: [10.1103/PhysRevB.88.134403](https://doi.org/10.1103/PhysRevB.88.134403).
- [83] Nicholas Metropolis et al. Equation of State Calculations by Fast Computing Machines. In: *The Journal of Chemical Physics* **21** (1953). DOI: [10.1063/1.1699114](https://doi.org/10.1063/1.1699114).
- [84] Wolfhard Janke. Monte Carlo Simulations in Statistical Physics – From Basic Principles to Advanced Applications. In: *Order, Disorder and Criticality*. 2013. ISBN: 978-981-4417-88-4. DOI: [10.1142/9789814417891_0003](https://doi.org/10.1142/9789814417891_0003).
- [85] David P. Landau and Kurt Binder. A Guide to Monte Carlo Simulations in Statistical Physics. Cambridge, UK: Cambridge University Press, 2014. DOI: [10.1017/cbo9781139696463](https://doi.org/10.1017/cbo9781139696463).
- [86] Eugen Birsan. Monte Carlo study on bilayer thin films magnetic properties dependence with discontinuous anisotropy parameters. In: *Open Physics* **6** (2008). DOI: [10.2478/s11534-008-0014-4](https://doi.org/10.2478/s11534-008-0014-4).
- [87] Dieter Landers and Lothar Rogge. Nichtstandard-Analysis reeller Funktionen. In: *Nichtstandard Analysis*. 1994. DOI: [10.1007/978-3-642-57915-8_11](https://doi.org/10.1007/978-3-642-57915-8_11).
- [88] Gerd Czycholl. Theoretische Festkörperphysik Band 1. Berlin, Germany: Springer, 2016. DOI: [10.1007/978-3-662-47141-8](https://doi.org/10.1007/978-3-662-47141-8).
- [89] Marjana Ležaić. Spin-gap materials from first principles: properties and applications of half-metallic ferromagnets. PhD thesis. RWTH Aachen University, 2005.
- [90] N. David Mermin and Herbert Wagner. Absence of Ferromagnetism or Antiferromagnetism in One- or Two-Dimensional Isotropic Heisenberg Models. In: *Phys. Rev. Lett.* **17** (1966). DOI: [10.1103/PhysRevLett.17.1133](https://doi.org/10.1103/PhysRevLett.17.1133).
- [91] Giovanni Pizzi et al. AiiDA: automated interactive infrastructure and database for computational science. In: *Computational Materials Science* **111** (2016). DOI: [10.1016/j.commatsci.2015.09.013](https://doi.org/10.1016/j.commatsci.2015.09.013).
- [92] Vasily Tseplyaev. Private Communication. Jülich, Germany, 2023.
- [93] Jens Bröder. High-throughput all-electron density functional theory simulations for a data-driven chemical interpretation of X-ray photoelectron spectra. Dissertation. RWTH Aachen University, 2021. DOI: [10.18154/RWTH-2021-03326](https://doi.org/10.18154/RWTH-2021-03326).
- [94] Anubhav Jain et al. Commentary: The Materials Project: A materials genome approach to accelerating materials innovation. In: *APL Materials* **1** (2013). DOI: [10.1063/1.4812323](https://doi.org/10.1063/1.4812323).
- [95] Mark D. Wilkinson et al. The FAIR Guiding Principles for scientific data management and stewardship. In: *Scientific Data* **3** (2016). DOI: [10.1038/sdata.2016.18](https://doi.org/10.1038/sdata.2016.18).
- [96] Jayesh Patel. Bridging data silos using big data integration. In: *International Journal of Database Management Systems* **11** (2019). DOI: <http://dx.doi.org/10.5121/ijdms.2019.11301>.
- [97] Robin Hilgers, Daniel Wortmann, and Stefan Blügel. Relaxed thin film structures of one, two, and three magnetic 3d transition metal layers on FCC noble-metal substrates based on FLAPW PBE calculations. Materials Cloud Archive. 2023. DOI: [10.24435/materialscloud:dk-wq](https://doi.org/10.24435/materialscloud:dk-wq).
- [98] Han Shi et al. Predicting drug-target interactions using Lasso with random forest based on evolutionary information and chemical structure. In: *Genomics* **111** (2019). DOI: [10.1016/j.ygeno.2018.12.007](https://doi.org/10.1016/j.ygeno.2018.12.007).

- [99] Riccardo Bonetto and Vincent Latzko. Chapter 8 - Machine learning. In: *Computing in Communication Networks*. 2020. DOI: [10.1016/B978-0-12-820488-7.00021-9](https://doi.org/10.1016/B978-0-12-820488-7.00021-9).
- [100] Paulino Jose Garcia Nieto, Esperanza García-Gonzalo, and José Paredes-Sánchez. Prediction of the critical temperature of a superconductor by using the WOA/MARS, Ridge, Lasso and Elastic-net machine learning techniques. In: *Neural Computing and Applications* **33** (2021). DOI: [10.1007/s00521-021-06304-z](https://doi.org/10.1007/s00521-021-06304-z).
- [101] Jonathan Schmidt et al. Recent advances and applications of machine learning in solid-state materials science. In: *npj Computational Materials* **5** (2019). DOI: [10.1038/s41524-019-0221-0](https://doi.org/10.1038/s41524-019-0221-0).
- [102] Huseyin Ucar, Durga Paudyal, and Kamal Choudhary. Machine learning predicted magnetic entropy change using chemical descriptors across a large compositional landscape. In: *Computational Materials Science* **209** (2022). DOI: [10.1016/j.commatsci.2022.111414](https://doi.org/10.1016/j.commatsci.2022.111414).
- [103] Keisuke Takahashi and Yuzuru Tanaka. Material synthesis and design from first principle calculations and machine learning. In: *Computational Materials Science* **112** (2016). DOI: [10.1016/j.commatsci.2015.11.013](https://doi.org/10.1016/j.commatsci.2015.11.013).
- [104] Anubhav Jain et al. New opportunities for materials informatics: Resources and data mining techniques for uncovering hidden relationships. In: *Journal of Materials Research* **31** (2016). DOI: [10.1557/jmr.2016.80](https://doi.org/10.1557/jmr.2016.80).
- [105] Roy J. Plunkett. The History of Polytetrafluoroethylene: Discovery and Development. In: *High Performance Polymers: Their Origin and Development*. Springer Netherlands, 1986. DOI: [10.1007/978-94-011-7073-4_25](https://doi.org/10.1007/978-94-011-7073-4_25).
- [106] Xiaoting Zhong et al. Explainable machine learning in materials science. In: *npj Computational Materials* **8** (2022). DOI: [10.1038/s41524-022-00884-7](https://doi.org/10.1038/s41524-022-00884-7).
- [107] Aya Rizk and Ahmed Elragal. Data science: developing theoretical contributions in information systems via text analytics. In: *Journal of Big Data* **7** (2020). DOI: [10.1186/s40537-019-0280-6](https://doi.org/10.1186/s40537-019-0280-6).
- [108] Mariette Awad and Rahul Khanna. Machine Learning. In: *Efficient Learning Machines: Theories, Concepts, and Applications for Engineers and System Designers*. 2015. DOI: [10.1007/978-1-4302-5990-9_1](https://doi.org/10.1007/978-1-4302-5990-9_1).
- [109] Thomas W. Edgar and David O. Manz. Chapter 6 - Machine Learning. In: *Research Methods for Cyber Security*. 2017. DOI: [10.1016/B978-0-12-805349-2.00006-6](https://doi.org/10.1016/B978-0-12-805349-2.00006-6).
- [110] Pei Wang. On Defining Artificial Intelligence. In: *Journal of Artificial General Intelligence* **10** (3919). DOI: [doi:10.2478/jagi-2019-0002](https://doi.org/10.2478/jagi-2019-0002).
- [111] Alan M. Turing. I.—COMPUTING MACHINERY AND INTELLIGENCE. In: *Mind* **LIX** (1950). DOI: [10.1093/mind/LIX.236.433](https://doi.org/10.1093/mind/LIX.236.433).
- [112] Avinash Kumar et al. A Survey on IBM Watson and Its Services. In: *Journal of Physics: Conference Series* **2273** (2022). DOI: [10.1088/1742-6596/2273/1/012022](https://doi.org/10.1088/1742-6596/2273/1/012022).
- [113] Yanqing Duan, John S. Edwards, and Yogesh K. Dwivedi. Artificial intelligence for decision making in the era of Big Data – evolution, challenges and research agenda. In: *International Journal of Information Management* **48** (2019). DOI: [10.1016/j.ijinfomgt.2019.01.021](https://doi.org/10.1016/j.ijinfomgt.2019.01.021).

- [114] Fabian Pedregosa et al. Scikit-learn: Machine Learning in Python. In: *Journal of Machine Learning Research* **12** (2011). DOI: <https://doi.org/10.5555/1953048.2078195>.
- [115] Anil K. Jain, M. Narasimha Murty, and Patrick J. Flynn. Data Clustering: A Review. In: *ACM Comput. Surv.* **31** (1999). DOI: [10.1145/331499.331504](https://doi.org/10.1145/331499.331504).
- [116] Laurens Van Der Maaten, Eric O. Postma, and Hendrik J. van den Herik. *Dimensionality reduction: A comparative review*. Technical Report TiCC TR 2009-005. Tilburg, Netherlands: Tilburg University, 2009.
- [117] Ramalingam Saravanan and Pothula Sujatha. A State of Art Techniques on Machine Learning Algorithms: A Perspective of Supervised Learning Approaches in Data Classification. In: *2018 Second International Conference on Intelligent Computing and Control Systems (ICICCS)*. 2018. DOI: [10.1109/ICCONS.2018.8663155](https://doi.org/10.1109/ICCONS.2018.8663155).
- [118] Sotiris B. Kotsiantis. Supervised Machine Learning: A Review of Classification Techniques. In: *Informatica* **31** (2007). DOI: [10.5555/1566770.1566773](https://doi.org/10.5555/1566770.1566773).
- [119] Leslie P. Kaelbling, Michael L. Littman, and Andrew W. Moore. Reinforcement learning: A survey. In: *Journal of artificial intelligence research* **4** (1996). DOI: <https://doi.org/10.1613/jair.301>.
- [120] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. In: *nature* **521** (2015). DOI: [10.1038/nature14539](https://doi.org/10.1038/nature14539).
- [121] Ji Wan et al. Deep Learning for Content-Based Image Retrieval: A Comprehensive Study. In: *Proceedings of the 22nd ACM International Conference on Multimedia*. MM '14. Orlando, FL, USA: Association for Computing Machinery, 2014. DOI: [10.1145/2647868.2654948](https://doi.org/10.1145/2647868.2654948).
- [122] Li Deng, Geoffrey Hinton, and Brian Kingsbury. New types of deep neural network learning for speech recognition and related applications: an overview. In: *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*. 2013. DOI: [10.1109/ICASSP.2013.6639344](https://doi.org/10.1109/ICASSP.2013.6639344).
- [123] Ronan Collobert and Jason Weston. A Unified Architecture for Natural Language Processing: Deep Neural Networks with Multitask Learning. In: *Proceedings of the 25th International Conference on Machine Learning*. New York, NY, USA, 2008. DOI: [10.1145/1390156.1390177](https://doi.org/10.1145/1390156.1390177).
- [124] Ravid Shwartz-Ziv and Amitai Armon. Tabular data: Deep learning is not all you need. In: *Information Fusion* **81** (2022). DOI: [10.1016/j.inffus.2021.11.011](https://doi.org/10.1016/j.inffus.2021.11.011).
- [125] Tianqi Chen and Carlos Guestrin. XGBoost: A Scalable Tree Boosting System. In: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. KDD '16. San Francisco, CA, USA: ACM, 2016. DOI: [10.1145/2939672.2939785](https://doi.org/10.1145/2939672.2939785).
- [126] Zhi-Hua Zhou. *Ensemble Methods: Foundations and Algorithms*. 1st ed. Cambridge, UK: Chapman & Hall/CRC, 2012. DOI: [10.5555/2381019](https://doi.org/10.5555/2381019).
- [127] Léo Grinsztajn, Edouard Oyallon, and Gaël Varoquaux. Why do tree-based models still outperform deep learning on typical tabular data? In: *Advances in Neural Information Processing Systems* **35** (*NeurIPS* 2022). 2022. ISBN: 9781713871088.
- [128] Scott M. Lundberg et al. From local explanations to global understanding with explainable AI for trees. In: *Nature Machine Intelligence* **2** (2020). DOI: [10.1038/s42256-019-0138-9](https://doi.org/10.1038/s42256-019-0138-9).

- [129] Laith Alzubaidi et al. Review of deep learning: concepts, CNN architectures, challenges, applications, future directions. In: *Journal of Big Data* **8** (2021). DOI: [10.1186/s40537-021-00444-8](https://doi.org/10.1186/s40537-021-00444-8).
- [130] Frank Emmert-Streib et al. An Introductory Review of Deep Learning for Prediction Models With Big Data. In: *Frontiers in Artificial Intelligence* **3** (2020). DOI: [10.3389/frai.2020.00004](https://doi.org/10.3389/frai.2020.00004).
- [131] Munazza Zaib, Quan Sheng, and Wei Emma Zhang. A Short Survey of Pre-trained Language Models for Conversational AI-A New Age in NLP. In: *Proceedings of the Australasian Computer Science Week Multiconference*. 2020. DOI: [10.1145/3373017.3373028](https://doi.org/10.1145/3373017.3373028).
- [132] Marc P. Deisenroth, A. Aldo Faisal, and Cheng S. Ong. *Mathematics for Machine Learning*. Cambridge, UK: Cambridge University Press, 2020. DOI: [10.1017/9781108679930](https://doi.org/10.1017/9781108679930).
- [133] Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. *Foundations of Machine Learning*. 2nd ed. Adaptive Computation and Machine Learning. Cambridge, MA, USA: MIT Press, 2018. ISBN: 978-0-262-03940-6.
- [134] Jerzy Krawczuk and Tomasz Łukaszuk. The feature selection bias problem in relation to high-dimensional gene data. In: *Artificial Intelligence in Medicine* **66** (2016). DOI: [10.1016/j.artmed.2015.11.001](https://doi.org/10.1016/j.artmed.2015.11.001).
- [135] *Encyclopedia of Machine Learning and Data Mining*. New York, NY, USA: Springer, 2017. DOI: [10.1007/978-1-4899-7687-1](https://doi.org/10.1007/978-1-4899-7687-1).
- [136] James Bergstra and Yoshua Bengio. Random search for hyper-parameter optimization. In: *Journal of machine learning research* **13** (2012). DOI: [10.5555/2188385.2188395](https://doi.org/10.5555/2188385.2188395).
- [137] Jia Wu et al. Hyperparameter Optimization for Machine Learning Models Based on Bayesian Optimization. In: *Journal of Electronic Science and Technology* **17** (2019). DOI: [10.11989/JEST.1674-862X.80904120](https://doi.org/10.11989/JEST.1674-862X.80904120).
- [138] Ryan Turner et al. Bayesian Optimization is Superior to Random Search for Machine Learning Hyperparameter Tuning: Analysis of the Black-Box Optimization Challenge 2020. In: *Proceedings of the NeurIPS 2020 Competition and Demonstration Track*. **133**. Proceedings of Machine Learning Research. 2021.
- [139] James Bergstra, Daniel Yamins, and David Cox. Making a Science of Model Search: Hyperparameter Optimization in Hundreds of Dimensions for Vision Architectures. In: *Proceedings of the 30th International Conference on Machine Learning*. **28**. Proceedings of Machine Learning Research. 2013. DOI: [10.5555/3042817](https://doi.org/10.5555/3042817).
- [140] Gareth M. James. Variance and bias for general loss functions. In: *Machine learning* **51** (2003). DOI: <https://doi.org/10.1023/A:1022899518027>.
- [141] Pankaj Mehta et al. A high-bias, low-variance introduction to Machine Learning for physicists. In: *Physics Reports* **810** (2019). DOI: [10.1016/j.physrep.2019.03.001](https://doi.org/10.1016/j.physrep.2019.03.001).
- [142] Stuart Geman, Elie Bienenstock, and René Doursat. Neural Networks and the Bias/Variance Dilemma. In: *Neural Computation* **4** (1992). DOI: [10.1162/neco.1992.4.1.1](https://doi.org/10.1162/neco.1992.4.1.1).
- [143] Emmanuel J. Candes and Terence Tao. Decoding by linear programming. In: *IEEE transactions on information theory* **51** (2005). DOI: [10.1109/TIT.2005.858979](https://doi.org/10.1109/TIT.2005.858979).
- [144] Robert Tibshirani. Regression Shrinkage and Selection via the Lasso. In: *Journal of the royal statistical society series b-methodological* **58** (1996). DOI: [10.1111/J.2517-6161.1996.TB02080.X](https://doi.org/10.1111/J.2517-6161.1996.TB02080.X).

- [145] Arthur E. Hoerl and Robert W. Kennard. Ridge Regression: Biased Estimation for Nonorthogonal Problems. In: *Technometrics* **42** (2000). DOI: [10.1080/00401706.2000.10485983](#).
- [146] Xue Ying. An Overview of Overfitting and its Solutions. In: *Journal of Physics: Conference Series* **1168** (2019). DOI: [10.1088/1742-6596/1168/2/022022](#).
- [147] Cort J. Willmott and Kenji Matsuura. Advantages of the Mean Absolute Error (MAE) over the Root Mean Square Error (RMSE) in Assessing Average Model Performance. In: *Climate Research* **30** (2005). DOI: [10.3354/cr030079](#).
- [148] Zhou Wang and Alan C. Bovik. Mean squared error: Love it or leave it? A new look at signal fidelity measures. In: *IEEE signal processing magazine* **26** (2009). DOI: [10.1109/MSP.2008.930649](#).
- [149] Davide Chicco, Matthijs J. Warrens, and Giuseppe Jurman. The coefficient of determination R-squared is more informative than SMAPE, MAE, MAPE, MSE and RMSE in regression analysis evaluation. In: *PeerJ Computer Science* **7** (2021). DOI: [10.7717/peerj-cs.623](#).
- [150] László Jeni, Jeffrey Cohn, and Fernando De la Torre. Facing Imbalanced Data - Recommendations for the Use of Performance Metrics. In: *2013 Humaine Association Conference on Affective Computing and Intelligent Interaction*. 2013. DOI: [10.1109/ACII.2013.47](#).
- [151] Shichao Zhang. Cost-sensitive KNN classification. In: *Neurocomputing* **391** (2020). DOI: [10.1016/j.neucom.2018.11.101](#).
- [152] Bianca Zadrozny and Charles Elkan. Obtaining Calibrated Probability Estimates from Decision Trees and Naive Bayesian Classifiers. In: *Proceedings of the Eighteenth International Conference on Machine Learning*. ICML '01. 2001. DOI: [10.5555/645530.655658](#).
- [153] Stefan Rüping. *A Simple Method For Estimating Conditional Probabilities For SVMs*. Technical Report. Dortmund, Germany, 2004. DOI: [10.17877/DE290R-15242](#).
- [154] Jiantao Jiao et al. Justification of Logarithmic Loss via the Benefit of Side Information. In: *IEEE Transactions on Information Theory* **61** (2015). DOI: [10.1109/TIT.2015.2462848](#).
- [155] Claude E. Shannon. A mathematical theory of communication. In: *The Bell System Technical Journal* **27** (1948). DOI: [10.1002/j.1538-7305.1948.tb01338.x](#).
- [156] Ron Kohavi. A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection. In: *Proceedings of the 14th International Joint Conference on Artificial Intelligence*. 2. IJCAI'95. 1995. DOI: [10.5555/1643031.1643047](#).
- [157] Yun Xu and Royston Goodacre. On Splitting Training and Validation Set: A Comparative Study of Cross-Validation, Bootstrap and Systematic Sampling for Estimating the Generalization Performance of Supervised Learning. In: *Journal of Analysis and Testing* **2** (2018). DOI: [10.1007/s41664-018-0068-2](#).
- [158] Roshan V. Joseph. Optimal Ratio for Data Splitting. In: *Statistical Analysis and Data Mining: The ASA Data Science Journal* **15** (2022). DOI: [10.1002/sam.11583](#).
- [159] Max Kuhn, Kjell Johnson, et al. *Applied predictive modeling*. New York, NY, USA: Springer, 2013. DOI: [10.1007/978-1-4614-6849-3](#).
- [160] Wolfgang Maass and Veda C. Storey. Pairing conceptual modeling with machine learning. In: *Data & Knowledge Engineering* **134** (2021). DOI: [10.1016/j.datak.2021.101909](#).
- [161] David H. Wolpert and William G. Macready. No free lunch theorems for optimization. In: *IEEE Transactions on Evolutionary Computation* **1** (1997). DOI: [10.1109/4235.585893](#).

- [162] Seymour Geisser and Wesley O. Johnson. Modes of Parametric Statistical Inference. Chichester, UK: John Wiley & Sons, 2006. ISBN: 978-0-471-74312-5.
- [163] Zhibiao Zhao. Parametric and nonparametric models and methods in financial econometrics. In: *Statistics Surveys* **2** (2008). DOI: [10.1214/08-SS034](#).
- [164] Hui Zou and Trevor Hastie. Regularization and Variable Selection via the Elastic Net. In: *Journal of the Royal Statistical Society. Series B (Statistical Methodology)* **67** (2005). DOI: [10.1111/j.1467-9868.2005.00503.x](#).
- [165] Edsger W. Dijkstra. A note on two problems in connexion with graphs. In: *Numerische Mathematik* **1** (1959). DOI: [10.1007/bf01386390](#).
- [166] James N. Morgan and John A. Sonquist. Problems in the Analysis of Survey Data, and a Proposal. In: *Journal of the American Statistical Association* **58** (1963). DOI: [10.1080/01621459.1963.10500855](#).
- [167] Wei-Yin Loh. Fifty years of classification and regression trees. In: *International Statistical Review* **82** (2014). DOI: [10.1111/insr.12016](#).
- [168] Leo Breiman. Classification and regression trees. London, UK: Routledge, 2017. DOI: [10.1201/9781315139470](#).
- [169] Leslie G. Valiant. A theory of the learnable. In: *Communications of the ACM* **27** (1984). DOI: [10.1145/1968.1972](#).
- [170] Robert E. Schapire. The strength of weak learnability. In: *Machine Learning* **5** (1990). DOI: [10.1007/bf00116037](#).
- [171] Lev Reyzin and Robert E. Schapire. How Boosting the Margin Can Also Boost Classifier Complexity. In: *Proceedings of the 23rd International Conference on Machine Learning*. ICML '06. 2006. DOI: [10.1145/1143844.1143939](#).
- [172] Wayne Iba and Pat Langley. Induction of One-Level Decision Trees. In: *Machine Learning Proceedings 1992*. San Francisco, CA, USA: Morgan Kaufmann, 1992. DOI: [10.1016/B978-1-55860-247-2.50035-8](#).
- [173] Thomas G. Dietterich. Ensemble Methods in Machine Learning. In: *Multiple Classifier Systems*. Berlin, Germany: Springer, 2000. DOI: [10.1007/3-540-45014-9_1](#).
- [174] David H. Wolpert. Stacked generalization. In: *Neural Networks* **5** (1992). DOI: [10.1016/S0893-6080\(05\)80023-1](#).
- [175] Leo Breiman. Bagging predictors. In: *Machine learning* **24** (1996). DOI: [10.1007/BF00058655](#).
- [176] Tin Kam Ho. Random decision forests. In: *Proceedings of 3rd International Conference on Document Analysis and Recognition*. **1**. 1995. DOI: [10.1109/ICDAR.1995.598994](#).
- [177] Leo Breiman. Random forests. In: *Machine learning* **45** (2001). DOI: [10.1023/A:1010933404324](#).
- [178] Pierre Geurts, Damien Ernst, and Louis Wehenkel. Extremely randomized trees. In: *Machine Learning* **63** (2006). DOI: [10.1007/s10994-006-6226-1](#).
- [179] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. The elements of statistical learning: data mining, inference and prediction. 2nd ed. New York, NY, USA: Springer, 2009. DOI: [10.1007/978-0-387-84858-7](#).

- [180] Andreas Mayr et al. The evolution of boosting algorithms. In: *Methods of information in medicine* **53** (2014). DOI: [10.3414/ME13-01-0122](#).
- [181] Leo Breiman. *Bias, Variance, And Arcing Classifiers*. Technical Report 460. Oakland, CA, USA, 2000.
- [182] Yoav Freund and Robert E. Schapire. A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting. In: *Journal of Computer and System Sciences* **55** (1997). DOI: [10.1006/jcss.1997.1504](#).
- [183] Edwin K. P. Chong and Stanislaw H. Żak. An introduction to optimization. Vol. 75. Nashville, TN, USA: John Wiley & Sons, 2013. ISBN: 978-1-118-27901-4.
- [184] Llew Mason et al. Boosting Algorithms as Gradient Descent. In: *Advances in Neural Information Processing Systems*. **12**. London, UK: MIT Press, 2000. ISBN: 0-262-19450-3.
- [185] Jerome H. Friedman. Greedy function approximation: A gradient boosting machine. In: *The Annals of Statistics* **29** (2001). DOI: [10.1214/aos/1013203451](#).
- [186] Rie Johnson and Tong Zhang. Learning Nonlinear Functions Using Regularized Greedy Forest. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* **36** (2014). DOI: [10.1109/TPAMI.2013.159](#).
- [187] Brook Taylor. *Methodus incrementorum directa & inversa*. Latin. London, UK: Impensis Gulielmi Innys, 1715.
- [188] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. Additive logistic regression: a statistical view of boosting (with discussion and a rejoinder by the authors). In: *The annals of statistics* **28** (2000). DOI: [10.1214/aos/1016218223](#).
- [189] Bryce Goodman and Seth Flaxman. European Union regulations on algorithmic decision-making and a “right to explanation”. In: *AI magazine* **38** (2017). DOI: [10.1609/aimag.v38i3.2741](#).
- [190] Sandra Wachter, Brent Mittelstadt, and Luciano Floridi. Why a right to explanation of automated decision-making does not exist in the general data protection regulation. In: *International Data Privacy Law* **7** (2017). DOI: [10.2139/ssrn.2903469](#).
- [191] Thomas Wischmeyer. Regulierung intelligenter Systeme. In: *Archiv des öffentlichen Rechts* **143** (2018). DOI: [10.1628/aoer-2018-0002](#).
- [192] Joshua Klayman. Varieties of Confirmation Bias. In: *Psychology of Learning and Motivation*. Elsevier, 1995. DOI: [10.1016/s0079-7421\(08\)60315-1](#).
- [193] Ribana Roscher et al. Explainable machine learning for scientific insights and discoveries. In: *Ieee Access* **8** (2020). DOI: [10.1109/ACCESS.2020.2976199](#).
- [194] Mario Krenn et al. On scientific understanding with artificial intelligence. In: *Nature Reviews Physics* **4** (2022). DOI: [10.1038/s42254-022-00518-3](#).
- [195] Martin J. Osborne and Ariel Rubinstein. A course in game theory. London, UK: MIT press, 1994. ISBN: 0-262-65040-1.
- [196] André Casajus and Frank Huettnner. Null players, solidarity, and the egalitarian Shapley values. In: *Journal of Mathematical Economics* **49** (2013). DOI: [10.1016/j.jmateco.2012.09.008](#).
- [197] Lloyd S. Shapley. A Value for n-Person Games. In: *Contributions to the Theory of Games II*. Princeton: Princeton University Press, 1953. DOI: [10.1515/9781400881970-018](#).

- [198] Katsushige Fujimoto, Ivan Kojadinovic, and Jean-Luc Marichal. Axiomatic characterizations of probabilistic and cardinal-probabilistic interaction indices. In: *Games and Economic Behavior* **55** (2006). DOI: [10.1016/j.geb.2005.03.002](https://doi.org/10.1016/j.geb.2005.03.002).
- [199] Yasuko Matsui and Tomomi Matsui. NP-completeness for calculating power indices of weighted majority games. In: *Theoretical Computer Science* **263** (2001). DOI: [10.1016/S0304-3975\(00\)00251-6](https://doi.org/10.1016/S0304-3975(00)00251-6).
- [200] Scott M. Lundberg and Su-In Lee. A Unified Approach to Interpreting Model Predictions. In: *Advances in Neural Information Processing Systems* **30**. 2017. DOI: [10.5555/3295222.3295230](https://doi.org/10.5555/3295222.3295230).
- [201] Scott M. Lundberg et al. Explainable machine-learning predictions for the prevention of hypoxaemia during surgery. In: *Nature Biomedical Engineering* **2** (2018). DOI: [10.1038/s41551-018-0304-0](https://doi.org/10.1038/s41551-018-0304-0).
- [202] Stefan Haufe et al. On the interpretation of weight vectors of linear models in multivariate neuroimaging. In: *NeuroImage* **87** (2014). DOI: [10.1016/j.neuroimage.2013.10.067](https://doi.org/10.1016/j.neuroimage.2013.10.067).
- [203] Shai Shalev-Shwartz et al. Online learning and online convex optimization. In: *Foundations and Trends® in Machine Learning* **4** (2012). DOI: [10.1561/2200000018](https://doi.org/10.1561/2200000018).
- [204] Thomas Seidl et al. Indexing Density Models for Incremental Learning and Anytime Classification on Data Streams. In: *Proceedings of the 12th International Conference on Extending Database Technology: Advances in Database Technology*. EDBT '09. Saint Petersburg, Russia: Association for Computing Machinery, 2009. DOI: [10.1145/1516360.1516397](https://doi.org/10.1145/1516360.1516397).
- [205] Michael McCloskey and Neal J. Cohen. Catastrophic Interference in Connectionist Networks: The Sequential Learning Problem. In: *Psychology of Learning and Motivation*. **24**. 1989. DOI: [10.1016/s0079-7421\(08\)60536-8](https://doi.org/10.1016/s0079-7421(08)60536-8).
- [206] Shai Ben-David, Eyal Kushilevitz, and Yishay Mansour. In: *Machine Learning* **29** (1997). DOI: [10.1023/a:1007465907571](https://doi.org/10.1023/a:1007465907571).
- [207] Dominik Kreuzberger, Niklas Kühl, and Sebastian Hirschl. Machine Learning Operations (MLOps): Overview, Definition, and Architecture. In: *IEEE Access* **11** (2023). DOI: [10.1109/access.2023.3262138](https://doi.org/10.1109/access.2023.3262138).
- [208] Norazian M. Noor et al. Comparison of linear interpolation method and mean method to replace the missing values in environmental data set. In: *Materials Science Forum*. **803**. Trans Tech Publ. 2015. DOI: [10.4028/www.scientific.net/msf.803.278](https://doi.org/10.4028/www.scientific.net/msf.803.278).
- [209] Norazian M. Noor et al. Filling missing data using interpolation methods: Study on the effect of fitting distribution. In: *Key Engineering Materials* **594** (2014). DOI: [10.4028/www.scientific.net/KEM.594-595.889](https://doi.org/10.4028/www.scientific.net/KEM.594-595.889).
- [210] Martin J. Shepperd and Michelle Cartwright. Predicting with sparse data. In: *IEEE Transactions on Software Engineering* **27** (2001). DOI: [10.1109/32.965339](https://doi.org/10.1109/32.965339).
- [211] Victor Laliena, Germán Albalade, and Javier Campo. Stability of the skyrmion lattice near the critical temperature in cubic helimagnets. In: *Phys. Rev. B* **98** (2018). DOI: [10.1103/PhysRevB.98.224407](https://doi.org/10.1103/PhysRevB.98.224407).
- [212] Friedrich Heusler, W. Starck, and Erich Haupt. Magnetisch-chemische studien. In: *Verh. Dtsch. Phys. Ges* **5** (1903), pp. 219–232.

- [213] Friedrich Heusler and Emil Take. The nature of the Heusler alloys. In: *Transactions of the Faraday Society* **8** (1912). DOI: [10.1039/TF9120800169](#).
- [214] Hüseyin Y. Uzunok et al. Physical properties and superconductivity of Heusler compound LiGa₂Rh: A first-principles calculation. In: *Solid State Communications* **311** (2020). DOI: [10.1016/j.ssc.2020.113859](#).
- [215] Samira Idrissi et al. Half-Metallicity and Magnetism in the Full Heusler Alloy Fe₂MnSn with L21 and XA Stability Ordering Phases. In: *Journal of Low Temperature Physics* **202** (2021). DOI: [10.1007/s10909-021-02562-2](#).
- [216] Adam Hauser et al. Dataset on Density Functional Theory investigation of Ternary Heusler Alloys. 2023. DOI: [10.17632/BY523YWZS9.2](#).
- [217] Qiang Gao et al. Designing rare-earth free permanent magnets in Heusler alloys via interstitial doping. In: *Acta Materialia* **186** (2020). DOI: [10.1016/j.actamat.2019.12.049](#).
- [218] Bernhard Hinterleitner et al. Thermoelectric performance of a metastable thin-film Heusler alloy. In: *Nature* **576** (2019). DOI: [10.1038/s41586-019-1751-9](#).
- [219] Anindya Roy et al. Half-Heusler Semiconductors as Piezoelectrics. In: *Physical Review Letters* **109** (2012). DOI: [10.1103/physrevlett.109.037602](#).
- [220] Tarek Bachagha et al. NiMn-based Heusler magnetic shape memory alloys: a review. In: *The International Journal of Advanced Manufacturing Technology* **103** (2019). DOI: [10.1007/s00170-019-03534-3](#).
- [221] Claudia Felser et al. Basics and prospective of magnetic Heusler compounds. In: *APL Materials* **3** (2015). DOI: [10.1063/1.4917387](#).
- [222] Zhe An et al. The universal definition of spin current. In: *Scientific Reports* **2** (2012). DOI: [10.1038/srep00388](#).
- [223] Claude Chappert, Albert Fert, and Frédéric Nguyen Van Dau. The emergence of spin electronics in data storage. In: *Nature materials* **6** (2007). DOI: [10.1038/nmat2024](#).
- [224] Angie Davidson et al. Perspectives of electrically generated spin currents in ferromagnetic materials. In: *Physics Letters A* **384** (2020). DOI: [10.1016/j.physleta.2019.126228](#).
- [225] Atsufumi Hirohata and David C. Lloyd. Heusler alloys for metal spintronics. In: *MRS Bulletin* **47** (2022). DOI: [10.1557/s43577-022-00350-1](#).
- [226] Hiroyuki Chudo et al. Spin pumping efficiency from half metallic Co₂MnSi. In: *Journal of Applied Physics* **109** (2011). DOI: [10.1063/1.3556433](#).
- [227] James Nelson and Stefano Sanvito. Predicting the Curie temperature of ferromagnets using machine learning. In: *Physical Review Materials* **3** (2019). DOI: [10.1103/PhysRevMaterials.3.104405](#).
- [228] Joshua F. Belot et al. Machine learning predictions of high-Curie-temperature materials. In: *Applied Physics Letters* **123** (2023). DOI: [10.1063/5.0156377](#).
- [229] Youhei Yamaji. Making Data-Driven Curie-Temperature Prediction Interpretable. In: *JPSJ News and Comments* **15** (2018). DOI: [10.7566/JPSJNC.15.10](#).
- [230] Hieu Chi Dam et al. Important Descriptors and Descriptor Groups of Curie Temperatures of Rare-earth Transition-metal Binary Alloys. In: *Journal of the Physical Society of Japan* **87** (2018). DOI: [10.7566/JPSJ.87.113801](#).

- [231] Robin Hilgers, Daniel Wortmann, and Stefan Blügel. Machine Learning-based estimation and explainable artificial intelligence-supported interpretation of the critical temperature from magnetic *ab initio* Heusler alloys data. In: *arXiv: cond-mat.mtrl-sci* (2023). Preprint. DOI: [10.48550/arXiv.2311.15423](https://doi.org/10.48550/arXiv.2311.15423).
- [232] Robin Hilgers, Daniel Wortmann, and Stefan Blügel. ML-ready Curie temperatures and descriptors extracted from the JuHemd database. Materials Cloud Archive. 2022. DOI: [10.24435/materialscloud:W1-YF](https://doi.org/10.24435/materialscloud:W1-YF).
- [233] Robin Hilgers, Daniel Wortmann, and Stefan Blügel. Data processing for the JuHemd database and ML-training and evaluation scripts. Zenodo. 2023. DOI: [10.5281/ZENODO.8436741](https://doi.org/10.5281/ZENODO.8436741).
- [234] Vitaliy V. Romaka et al. Determination of structural disorder in Heusler-type phases. In: *Computational Materials Science* **172** (2020). DOI: [10.1016/j.commatsci.2019.109307](https://doi.org/10.1016/j.commatsci.2019.109307).
- [235] Kamil Ciesielski et al. Structural, thermodynamic and magnetotransport properties of half-Heusler compound HoPtSb. In: *Journal of Alloys and Compounds* **829** (2020). DOI: [10.1016/j.jallcom.2020.154467](https://doi.org/10.1016/j.jallcom.2020.154467).
- [236] Zihang Liu et al. Design of High-Performance Disordered Half-Heusler Thermoelectric Materials Using 18-Electron Rule. In: *Advanced Functional Materials* **29** (2019). DOI: [10.1002/adfm.201905044](https://doi.org/10.1002/adfm.201905044).
- [237] Rie Y. Umetsu, Akinari Okubo, and Ryosuke Kainuma. Magnetic and chemical order-disorder transformations in $\text{Co}_2\text{Fe}(\text{Ga}_{1-x}\text{Si}_x)$ and $\text{Co}_2\text{Fe}(\text{Al}_{1-y}\text{Si}_y)$ Heusler alloys. In: *Journal of Applied Physics* **111** (2012). DOI: [10.1063/1.3700220](https://doi.org/10.1063/1.3700220).
- [238] Frederick Casper et al. Half-Heusler compounds: novel materials for energy and spintronic applications. In: *Semiconductor Science and Technology* **27** (2012). DOI: [10.1088/0268-1242/27/6/063001](https://doi.org/10.1088/0268-1242/27/6/063001).
- [239] Koichi Momma and Fujio Izumi. VESTA3 for three-dimensional visualization of crystal, volumetric and morphology data. In: *Journal of Applied Crystallography* **44** (2011). DOI: [10.1107/S0021889811038970](https://doi.org/10.1107/S0021889811038970).
- [240] Mohamed H. Elahmar et al. Structural, mechanical, electronic and magnetic properties of a new series of quaternary Heusler alloys CoFeMnZ ($\text{Z}=\text{Si}, \text{As}, \text{Sb}$): A first-principle study. In: *Journal of Magnetism and Magnetic Materials* **393** (2015). DOI: [10.1016/j.jmmm.2015.05.019](https://doi.org/10.1016/j.jmmm.2015.05.019).
- [241] Srikrishna Ghosh and Subhradip Ghosh. Half-Metallicity in Quaternary Heusler Alloys with 3d and 4d Elements: Observations and Insights from DFT Calculations. In: *physica status solidi (b)* **256** (2019). DOI: [10.1002/pssb.201900039](https://doi.org/10.1002/pssb.201900039).
- [242] Jiamin Chen. Development of Heusler-alloy-based Current- Perpendicular-to-Plane Giant Magnetoresistive Devices. PhD thesis. University of Tsukuba, 2017.
- [243] Chris J. Palmstrøm. Heusler compounds and spintronics. In: *Progress in Crystal Growth and Characterization of Materials* **62** (2016). DOI: [10.1016/j.pcrysgrow.2016.04.020](https://doi.org/10.1016/j.pcrysgrow.2016.04.020).
- [244] Burkhard Fricke et al. Use of the DV $X[\alpha]$ -Method in the Field of Superheavy Atoms. In: *Advances in Quantum Chemistry*. **29**. 1998. DOI: [10.1016/S0065-3276\(08\)60265-8](https://doi.org/10.1016/S0065-3276(08)60265-8).
- [245] John P. Perdew, Kieron Burke, and Yue Wang. Generalized gradient approximation for the exchange-correlation hole of a many-electron system. In: *Phys. Rev. B* **54** (1996). DOI: [10.1103/PhysRevB.54.16533](https://doi.org/10.1103/PhysRevB.54.16533).

- [246] Kemal Özdoğan and Iosif Galanakis. Effect of order on the half-metallic gap in Heusler compounds. In: *Journal of Applied Physics - J APPL PHYS* **110** (2011). DOI: [10.1063/1.3642990](#).
- [247] Peter Peduzzi et al. A simulation study of the Number of events per variable in logistic regression analysis. In: *Journal of clinical epidemiology* **49** (1996). DOI: [10.1016/s0895-4356\(96\)00236-3](#).
- [248] Makoto Yamada et al. High-Dimensional Feature Selection by Feature-Wise Kernelized Lasso. In: *Neural Computation* **26** (2014). DOI: [10.1162/NECO_a_00537](#).
- [249] Łukasz Mentel. mendeleev – A Python resource for properties of chemical elements, ions and isotopes. Zenodo. Version 0.12.0. 2014. DOI: [10.5281/ZENODO.7308900](#).
- [250] Chris Gibbons et al. Supervised machine learning algorithms can classify open-text feedback of doctor performance with human-level accuracy. In: *Journal of medical Internet research* **19** (2017). DOI: [10.2196/jmir.6533](#).
- [251] Sayan Mukherjee et al. Estimating dataset size requirements for classifying DNA microarray data. In: *Journal of computational biology* **10** (2003). DOI: [10.1089/106652703321825928](#).
- [252] Leopold Talirz et al. Materials Cloud, a platform for open computational science. In: *Scientific Data* **7** (2020). DOI: [10.1038/s41597-020-00637-5](#).
- [253] Roman Kováčik. Private Communication. Jülich, Germany, 2021.
- [254] Kemal Özdoğan et al. Search for half-metallic ferrimagnetism in V-based Heusler alloys Mn₂VZ (Z = Al, Ga, In, Si, Ge, Sn). In: *Journal of Physics: Condensed Matter* **18** (2006). DOI: [10.1088/0953-8984/18/10/013](#).
- [255] Ruben Weht and Warren E. Pickett. Half-metallic ferrimagnetism in Mn₂VAl. In: *Phys. Rev. B* **60** (1999). DOI: [10.1103/PhysRevB.60.13006](#).
- [256] Jan Larsen and Cyril Goutte. On optimal data split for generalization estimation and model selection. In: *Neural Networks for Signal Processing IX: Proceedings of the 1999 IEEE Signal Processing Society Workshop (Cat. No.98TH8468)*. 1999. DOI: [10.1109/NNSP.1999.788141](#).
- [257] Md Manjurul Ahsan et al. Effect of Data Scaling Methods on Machine Learning Algorithms and Model Performance. In: *Technologies* **9** (2021). DOI: [10.3390/technologies9030052](#).
- [258] Tian Shang et al. Design of magnetic spirals in layered perovskites: Extending the stability range far beyond room temperature. In: *Science Advances* **4** (2018). DOI: [10.1126/sciadv.aau6386](#).
- [259] Min-Ling Zhang and Zhi-Hua Zhou. A k-nearest neighbor based algorithm for multi-label classification. In: *2005 IEEE international conference on granular computing*. **2**. IEEE. 2005. DOI: [10.1109/GRC.2005.1547385](#).
- [260] Charles J. Stone. Consistent Nonparametric Regression. In: *The Annals of Statistics* **5** (1977). DOI: [10.1214/aos/1176343886](#).
- [261] Evelyn Fix and Joseph Lawson Hodges. Discriminatory analysis. Nonparametric discrimination: Consistency properties. In: *International Statistical Review/Revue Internationale de Statistique* **57** (1989). DOI: [10.2307/1403797](#).
- [262] Thomas Cover and Peter E. Hart. Nearest neighbor pattern classification. In: *IEEE Transactions on Information Theory* **13** (1967). DOI: [10.1109/TIT.1967.1053964](#).

- [263] Rajarshi Guha and Peter C. Jurs. Development of Linear, Ensemble, and Nonlinear Models for the Prediction and Interpretation of the Biological Activity of a Set of PDGFR Inhibitors. In: *Journal of Chemical Information and Computer Sciences* **44** (2004). DOI: [10.1021/ci049849f](#).
- [264] Bradley Efron. Bootstrap Methods: Another Look at the Jackknife. In: *The Annals of Statistics* **7** (1979). DOI: [10.1214/aos/1176344552](#).
- [265] William S. Cleveland. Robust Locally Weighted Regression and Smoothing Scatterplots. In: *Journal of the American Statistical Association* **74** (1979). DOI: [10.1080/01621459.1979.10481038](#).
- [266] Hassan Kibirige. scikit-misc. Version 0.3.0. <https://github.com/has2k1/scikit-misc>. Aug. 13, 2023.
- [267] Murray Rosenblatt. Remarks on Some Nonparametric Estimates of a Density Function. In: *The Annals of Mathematical Statistics* **27** (1956). DOI: [10.1214/aoms/1177728190](#).
- [268] Emanuel Parzen. On Estimation of a Probability Density Function and Mode. In: *The Annals of Mathematical Statistics* **33** (1962). DOI: [10.1214/aoms/1177704472](#).
- [269] David W. Hosmer, Stanley Lemeshow, and Rodney X. Sturdivant. Applied Logistic Regression. Hoboken, NJ, USA: Wiley, 2013. DOI: [10.1002/9781118548387](#).
- [270] Joseph Berkson. Application of the Logistic Function to Bio-Assay. In: *Journal of the American Statistical Association* **39** (1944). DOI: [10.1080/01621459.1944.10500699](#).
- [271] Shiv Om Kumar, Vineeta Shukla, and Sanjeev K. Srivastava. Role of electronegativity on the bulk modulus, magnetic moment and band gap of Co₂MnAl based Heusler alloys. In: *Journal of Science: Advanced Materials and Devices* **4** (2019). DOI: [10.1016/j.jsamd.2019.02.001](#).
- [272] Keyan Li, Congying Kang, and Dongfeng Xue. Electronegativity calculation of bulk modulus and band gap of ternary ZnO-based alloys. In: *Materials Research Bulletin* **47** (2012). DOI: [10.1016/j.materresbull.2012.04.115](#).
- [273] Ashish Kumar et al. Evolution of a weak magnetic moment in the FeNbSb based HH materials via Ni doping at Fe site. In: *Journal of Magnetism and Magnetic Materials* **554** (2022). DOI: [10.1016/j.jmmm.2022.169306](#).
- [274] Hongzhi Luo et al. Slater–Pauling behavior and half-metallicity in Heusler alloys Mn₂CuZ (Z=Ge and Sb). In: *Computational Materials Science* **50** (2011). DOI: [10.1016/j.commatsci.2011.05.037](#).
- [275] Mukhtiyar Singh et al. Disorder dependent half-metallicity in Mn₂CoSi inverse Heusler alloy. In: *Journal of Solid State Chemistry* **208** (2013). DOI: [10.1016/j.jssc.2013.09.041](#).
- [276] Kirill D. Belashchenko, James K. Glasbrenner, and Aleksander L. Wysocki. Spin injection from a half-metal at finite temperatures. In: *Phys. Rev. B* **86** (2012). DOI: [10.1103/PhysRevB.86.224402](#).
- [277] Sabpreet Bhatti et al. Spintronics based random access memory: a review. In: *Materials Today* **20** (2017). DOI: [10.1016/j.mattod.2017.07.007](#).
- [278] Sergey V. Faleev et al. Unified explanation of chemical ordering, the Slater-Pauling rule, and half-metallicity in full Heusler compounds. In: *Phys. Rev. B* **95** (2017). DOI: [10.1103/PhysRevB.95.045140](#).

- [279] Paolo Giannozzi et al. Advanced capabilities for materials modelling with Quantum ESPRESSO. In: *Journal of Physics: Condensed Matter* **29** (2017). DOI: [10.1088/1361-648X/aa8f79](https://doi.org/10.1088/1361-648X/aa8f79).
- [280] Paolo Giannozzi et al. QUANTUM ESPRESSO: a modular and open-source software project for quantum simulations of materials. In: *Journal of Physics: Condensed Matter* **21** (2009). DOI: [10.1088/0953-8984/21/39/395502](https://doi.org/10.1088/0953-8984/21/39/395502).
- [281] Ka Ming Law. Private Communication. Tuscaloosa, AL, USA, 2023.
- [282] Iosif Galanakis. Slater-Pauling Behavior in Half-Metallic Heusler Compounds. In: *Nanomaterials* **13** (2023). DOI: [10.3390/nano13132010](https://doi.org/10.3390/nano13132010).
- [283] Xiao-Ping Wei et al. Half-metallic ferrimagnetism in Mn_2CuGe . In: *Journal of Magnetism and Magnetic Materials* **322** (2010). DOI: [10.1016/j.jmmm.2010.05.063](https://doi.org/10.1016/j.jmmm.2010.05.063).
- [284] Iosif Galanakis and Peter H. Dederichs. Half-Metallicity and Slater-Pauling Behavior in the Ferromagnetic Heusler Alloys. In: *Half-metallic Alloys*. 2006. DOI: [10.1007/11506256_1](https://doi.org/10.1007/11506256_1).
- [285] Fernando Nogueira. Bayesian Optimization: Open source constrained global optimization tool for Python. <https://github.com/bayesian-optimization/BayesianOptimization>. 2014.
- [286] Szilvia Szeghalmy and Attila Fazekas. A Comparative Study of the Use of Stratified Cross-Validation and Distribution-Balanced Stratified Cross-Validation in Imbalanced Learning. In: *Sensors* **23** (2023). DOI: [10.3390/s23042333](https://doi.org/10.3390/s23042333).
- [287] Victoria López, Alberto Fernández, and Francisco Herrera. On the importance of the validation technique for classification with imbalanced datasets: Addressing covariate shift when data is skewed. In: *Information Sciences* **257** (2014). DOI: [10.1016/j.ins.2013.09.038](https://doi.org/10.1016/j.ins.2013.09.038).
- [288] The Materials Project. Materials Data on CrFe_2Si by Materials Project. 2020. DOI: [10.17188/1350145](https://doi.org/10.17188/1350145).
- [289] Jason M. Munro et al. An improved symmetry-based approach to reciprocal space path selection in band structure calculations. In: *npj Computational Materials* **6** (2020). DOI: [10.1038/s41524-020-00383-7](https://doi.org/10.1038/s41524-020-00383-7).
- [290] Bothina A. Hamad. The effect of defects on the electronic structure and magnetic map of the Fe_2CrSi Heusler alloy: ab-initio calculations. In: *The European Physical Journal B* **80** (2011). DOI: [10.1140/epjb/e2011-10605-5](https://doi.org/10.1140/epjb/e2011-10605-5).
- [291] Georg Kresse and Jürgen Furthmüller. Efficient iterative schemes for ab initio total-energy calculations using a plane-wave basis set. In: *Phys. Rev. B* **54** (1996). DOI: [10.1103/PhysRevB.54.11169](https://doi.org/10.1103/PhysRevB.54.11169).
- [292] Dejan Zagorac et al. Recent developments in the Inorganic Crystal Structure Database: Theoretical crystal structure data and related features. In: *Journal of Applied Crystallography* **52** (2019). DOI: [10.1107/S160057671900997X](https://doi.org/10.1107/S160057671900997X).
- [293] Robin Hilgers, Daniel Wortmann, and Stefan Blügel. Code collection for machine learning assisted materials screening in the search for novel half-metallic Heusler alloys. Zenodo. 2023. DOI: [10.5281/ZENODO.10075055](https://doi.org/10.5281/ZENODO.10075055).
- [294] M. El Amine Monir et al. Study of structural, electronic and magnetic properties of CoFeIn and Co_2FeIn Heusler alloys. In: *Journal of Magnetism and Magnetic Materials* **394** (2015). DOI: [10.1016/j.jmmm.2015.06.077](https://doi.org/10.1016/j.jmmm.2015.06.077).

- [295] S. Javad Hashemifar, Peter Kratzer, and Matthias Scheffler. Preserving the Half-Metallicity at the Heusler Alloy $\text{Co}_2\text{MnSi}(001)$ Surface: A Density Functional Theory Study. In: *Phys. Rev. Lett.* **94** (2005). DOI: [10.1103/PhysRevLett.94.096402](https://doi.org/10.1103/PhysRevLett.94.096402).
- [296] Markus Meinert. Modified Becke-Johnson potential investigation of half-metallic Heusler compounds. In: *Phys. Rev. B* **87** (2013). DOI: [10.1103/PhysRevB.87.045103](https://doi.org/10.1103/PhysRevB.87.045103).
- [297] Ting Song et al. Electronic Structure, Phase Stability, and Elastic Properties of Inverse Heusler Compound Mn_2RuSi at High Pressure. In: *Journal of Superconductivity and Novel Magnetism* **30** (2016). DOI: [10.1007/s10948-016-3888-1](https://doi.org/10.1007/s10948-016-3888-1).
- [298] Abdelkarim Hamri et al. $\text{Ru}_2\text{Ti}_{1-x}\text{Fe}_x\text{Ge}$: Novel candidate for the spintronic applications. In: *International Journal of Modern Physics B* **29** (2015). DOI: [10.1142/s0217979215500575](https://doi.org/10.1142/s0217979215500575).
- [299] Jianhua Ma et al. Computational investigation of inverse Heusler compounds for spintronics applications. In: *Phys. Rev. B* **98** (2018). DOI: [10.1103/PhysRevB.98.094410](https://doi.org/10.1103/PhysRevB.98.094410).
- [300] Ulvi Kanbur and Gökhan Gökoğlu. Half-metallic magnetism of Co_2CrX ($\text{X}=\text{As}, \text{Sb}$) Heusler compounds: An ab initio study. In: *Journal of Magnetism and Magnetic Materials* **323** (2011). DOI: [10.1016/j.jmmm.2010.12.034](https://doi.org/10.1016/j.jmmm.2010.12.034).
- [301] Halima S. Cherif et al. Computational determination of structural, electronic, magnetic and thermodynamic properties of Co_2HfZ ($\text{Z}=\text{Al}, \text{Ga}, \text{Si}$ and Sn) full Heusler compounds for spintronic applications. In: *Journal of Alloys and Compounds* **894** (2022). DOI: [10.1016/j.jallcom.2021.162503](https://doi.org/10.1016/j.jallcom.2021.162503).
- [302] Hai-Ming Huang, Shi-Jun Luo, and Kai-Lun Yao. First-principles investigation of the electronic structure and magnetism of Heusler alloys CoMnSb and Co_2MnSb . In: *Physica B: Condensed Matter* **406** (2011). DOI: [10.1016/j.physb.2011.01.031](https://doi.org/10.1016/j.physb.2011.01.031).
- [303] Dibya P. Rai and Ram K. Thapa. An abinitio study of the half-metallic properties of Co_2TGe ($\text{T}=\text{Sc}, \text{Ti}, \text{V}, \text{Cr}, \text{Mn}, \text{Fe}$): LSDA+ U method. In: *Journal of the Korean Physical Society* **62** (2013). DOI: [10.3938/jkps.62.1652](https://doi.org/10.3938/jkps.62.1652).
- [304] Tufan Roy, Dhanshree Pandey, and Aparna Chakrabarti. Probing the possibility of coexistence of martensite transition and half-metallicity in Ni and Co-based full-Heusler alloys: An ab initio calculation. In: *Phys. Rev. B* **93** (2016). DOI: [10.1103/PhysRevB.93.184102](https://doi.org/10.1103/PhysRevB.93.184102).
- [305] Mansour Benidris et al. Electronic structure, thermoelectric, mechanical and phonon properties of full-Heusler alloy (Fe_2CrSb): a first-principles study. In: *Bulletin of Materials Science* **44** (2021). DOI: [10.1007/s12034-021-02496-1](https://doi.org/10.1007/s12034-021-02496-1).
- [306] Nazmiye Kervan and Selçuk Kervan. Half-metallicity in the full-Heusler Fe_2MnP compound. In: *Intermetallics* **24** (2012). DOI: [10.1016/j.intermet.2012.01.030](https://doi.org/10.1016/j.intermet.2012.01.030).
- [307] Shakeel A. Khandy and Jeng-Da Chai. Novel half-metallic L21 structured full-Heusler compound for promising spintronic applications: A DFT-based computer simulation. In: *Journal of Magnetism and Magnetic Materials* **487** (2019). DOI: [10.1016/j.jmmm.2019.165289](https://doi.org/10.1016/j.jmmm.2019.165289).
- [308] Hongzhi Luo et al. Half-metallicity in Fe-based Heusler alloys Fe_2TiZ ($\text{Z}=\text{Ga}, \text{Ge}, \text{As}, \text{In}, \text{Sn}$ and Sb). In: *Journal of Magnetism and Magnetic Materials* **324** (2012). DOI: [10.1016/j.jmmm.2012.05.033](https://doi.org/10.1016/j.jmmm.2012.05.033).
- [309] Ersoy Şaşıoğlu, Leonid M. Sandratskii, and Patrick Bruno. First-principles study of exchange interactions and Curie temperatures of half-metallic ferrimagnetic full Heusler alloys Mn_2VZ ($\text{Z}=\text{Al}, \text{Ge}$). In: *Journal of Physics: Condensed Matter* **17** (2005). DOI: [10.1088/0953-8984/17/6/017](https://doi.org/10.1088/0953-8984/17/6/017).

- [310] Danil R. Baigutlin, Vladimir V. Sokolovskiy, and Vasiliy D. Buchelnikov. Interplay of electronic structure and magnetism in Fe₂- and Rh₂-based Heusler alloys. In: *Phys. Rev. B* **107** (2023). DOI: [10.1103/PhysRevB.107.245130](https://doi.org/10.1103/PhysRevB.107.245130).
- [311] Ying Chen et al. Half-metallicity and magnetism of the quaternary inverse full-Heusler alloy Ti_{2-x}M_xCoAl (M = Nb, V) from the first-principles calculations. In: *The European Physical Journal B* **87** (2014). DOI: [10.1140/epjb/e2013-40961-9](https://doi.org/10.1140/epjb/e2013-40961-9).
- [312] Nazmiye Kervan and Selçuk Kervan. A first-principle study of half-metallic ferrimagnetism in the Ti₂CoGa Heusler compound. In: *Journal of Magnetism and Magnetic Materials* **324** (2012). DOI: [10.1016/j.jmmm.2011.09.003](https://doi.org/10.1016/j.jmmm.2011.09.003).
- [313] Xiao-Ping Wei et al. First-principles study on stability, electronic and thermodynamic properties of Ti₂CoIn and Ti₂NiIn. In: *The European Physical Journal B* **86** (2013). DOI: [10.1140/epjb/e2013-40784-8](https://doi.org/10.1140/epjb/e2013-40784-8).
- [314] Djihad Mokhtari et al. Structural, electronic, magnetic and thermoelectric properties of inverse Heusler alloys Ti₂CoSi, Mn₂CoAl and Cr₂ZnSi by employing Ab initio calculations. In: *Philosophical Magazine* **100** (2020). DOI: [10.1080/14786435.2020.1731926](https://doi.org/10.1080/14786435.2020.1731926).
- [315] Sotiris Skaftouros et al. Generalized Slater-Pauling rule for the inverse Heusler compounds. In: *Phys. Rev. B* **87** (2013). DOI: [10.1103/PhysRevB.87.024420](https://doi.org/10.1103/PhysRevB.87.024420).
- [316] Rie Y. Umetsu et al. Magnetic properties and band structures of half-metal-type Co₂CrGa Heusler alloy. In: *Applied Physics Letters* **85** (2004). DOI: [10.1063/1.1790029](https://doi.org/10.1063/1.1790029).
- [317] Yutong Li et al. Ab initio predictions of magnetism and half-metallicity of (111)-surfaces of Co₂CrSi full-Heusler alloy. In: *Vacuum* **192** (2021). DOI: [10.1016/j.vacuum.2021.110455](https://doi.org/10.1016/j.vacuum.2021.110455).
- [318] Salma Babiker A., Guoying Gao, and Kailun Yao. Half-metallicity and magnetism of Heusler alloys Co₂HfZ (Z=Al, Ga, Ge, Sn). In: *Journal of Magnetism and Magnetic Materials* **441** (2017). DOI: [10.1016/j.jmmm.2017.04.099](https://doi.org/10.1016/j.jmmm.2017.04.099).
- [319] Kumari Seema and Ranjan Kumar. An ab-initio study of full Heusler alloy Fe₂CoGa. In: *AIP Conference Proceedings* **1536** (2013). DOI: [10.1063/1.4810472](https://doi.org/10.1063/1.4810472).
- [320] Zhi Ren, Songtao Li, and Hongzhi Luo. Structure and magnetic properties of Fe₂CoGe synthesized by ball-milling. In: *Physica B: Condensed Matter* **405** (2010). DOI: [10.1016/j.physb.2010.04.008](https://doi.org/10.1016/j.physb.2010.04.008).
- [321] A. Amudhavalli, Ratnavelu Rajeswarapalanichamy, and Kombiahthevar Iyakutti. Investigation of half-metallic and magnetic phase transition in Co₂TiZ (Z = Al, Ga, In) Heusler alloys. In: *Phase Transitions* **92** (2019). DOI: [10.1080/01411594.2019.1653463](https://doi.org/10.1080/01411594.2019.1653463).
- [322] Oksana N. Draganyuk, Vyacheslav S. Zhandun, and Natalia G. Zamkova. Half-metallicity in Fe₂MnSi and Mn₂FeSi heusler compounds: A comparative ab initio study. In: *Materials Chemistry and Physics* **271** (2021). DOI: [10.1016/j.matchemphys.2021.124897](https://doi.org/10.1016/j.matchemphys.2021.124897).
- [323] Xiao-Ping Wei, Yan-Dong Chu, and Jian-Bo Deng. Effect of Ni and Sn doping on the half-metallicity of full Heusler Ti₂CoIn alloy. In: *Journal of Magnetism and Magnetic Materials* **354** (2014). DOI: [10.1016/j.jmmm.2013.11.035](https://doi.org/10.1016/j.jmmm.2013.11.035).
- [324] Robin Hilgers, Daniel Wortmann, and Stefan Blügel. Density of states of full and inverse Heusler magnetic alloys. Materials Cloud Archive. 2023. DOI: [10.24435/materialscloud:v5-5z](https://doi.org/10.24435/materialscloud:v5-5z).

- [325] John P. Perdew et al. Atoms, molecules, solids, and surfaces: Applications of the generalized gradient approximation for exchange and correlation. In: *Phys. Rev. B* **46** (1992). DOI: [10.1103/PhysRevB.46.6671](https://doi.org/10.1103/PhysRevB.46.6671).
- [326] Lu Lu et al. Magnetism and local symmetry breaking in a Mott insulator with strong spin orbit interactions. In: *Nature Communications* **8** (2017). DOI: [10.1038/ncomms14407](https://doi.org/10.1038/ncomms14407).
- [327] The Materials Project. Materials Data on VZnCo_2 by Materials Project. 2020. DOI: [10.17188/1718820](https://doi.org/10.17188/1718820).
- [328] Seonghoon Woo et al. Observation of room-temperature magnetic skyrmions and their current-driven dynamics in ultrathin metallic ferromagnets. In: *Nature Materials* **15** (2016). DOI: [10.1038/nmat4593](https://doi.org/10.1038/nmat4593).
- [329] Lars J. Bannenberg et al. Multiple low-temperature skyrmionic states in a bulk chiral magnet. In: *npj Quantum Materials* **4** (2019). DOI: [10.1038/s41535-019-0150-7](https://doi.org/10.1038/s41535-019-0150-7).
- [330] Robert E. Camley. Chapter 6 - Thermal Properties of Magnetic Multilayers and Nanostructures: Applications to Static and Dynamic Behavior. In: *Magnetism of Surfaces, Interfaces, and Nanoscale Materials*. 5. Handbook of Surface Science. 2015. DOI: <https://doi.org/10.1016/B978-0-444-62634-9.00006-0>.
- [331] Abhishek Talapatra, J. Arout Chelvane, and Jyoti R. Mohanty. Tuning magnetic microstructure in Gd-Fe thin films: Experiment and Simulation. In: *Journal of Magnetism and Magnetic Materials* **448** (2018). DOI: [10.1016/j.jmmm.2017.07.092](https://doi.org/10.1016/j.jmmm.2017.07.092).
- [332] Anjan Soumyanarayanan et al. Tunable room-temperature magnetic skyrmions in Ir/Fe-Co/Pt multilayers. In: *Nature Materials* **16** (2017). DOI: [10.1038/nmat4934](https://doi.org/10.1038/nmat4934).
- [333] C. Liu and Samuel D. Bader. Perpendicular surface magnetic anisotropy in ultrathin epitaxial Fe films. In: *Journal of Vacuum Science & Technology A* **8** (1990). DOI: [10.1116/1.576657](https://doi.org/10.1116/1.576657).
- [334] Hongying Jia et al. Material systems for FM-/AFM-coupled skyrmions in Co/Pt-based multilayers. In: *Phys. Rev. Mater.* **4** (2020). DOI: [10.1103/PhysRevMaterials.4.094407](https://doi.org/10.1103/PhysRevMaterials.4.094407).
- [335] Michael S. Pierce et al. Influence of structural disorder on magnetic domain formation in perpendicular anisotropy thin films. In: *Physical Review B* **87** (2013). DOI: [10.1103/PhysRevB.87.184428](https://doi.org/10.1103/PhysRevB.87.184428).
- [336] Stefan Heinze et al. Spontaneous atomic-scale magnetic skyrmion lattice in two dimensions. In: *Nature Physics* **7** (2011). DOI: [10.1038/nphys2045](https://doi.org/10.1038/nphys2045).
- [337] Masapogu Raju et al. The evolution of skyrmions in Ir/Fe/Co/Pt multilayers and their topological Hall signature. In: *Nature Communications* **10** (2019). DOI: [10.1038/s41467-018-08041-9](https://doi.org/10.1038/s41467-018-08041-9).
- [338] Robin Hilgers, Daniel Wortmann, and Stefan Blügel. Workflow, Data Processing, Data Analysis and predictive ML scripts used for DFT integrated Machine Learning methodology in combination with the FLAPW code FLEUR. Zenodo. 2023. DOI: [10.5281/ZENODO.10056614](https://doi.org/10.5281/ZENODO.10056614).
- [339] Stefan Blügel. Two-dimensional ferromagnetism of 3d, 4d, and 5d transition metal monolayers on noble metal (001) substrates. In: *Phys. Rev. Lett.* **68** (1992). DOI: [10.1103/PhysRevLett.68.851](https://doi.org/10.1103/PhysRevLett.68.851).
- [340] Kurt Lejaeghere et al. Reproducibility in density functional theory calculations of solids. In: *Science* **351** (2016). DOI: [10.1126/science.aad3000](https://doi.org/10.1126/science.aad3000).

- [341] Howard E. Swanson and George M. Ugrinic. Standard X-Ray Diffraction Powder Patterns. In: *Circular of the Bureau of Standards* No. 539 3 (1954). DOI: [10.6028/NBS.CIRC.539v3](#).
- [342] Howard E. Swanson. Standard X-Ray Diffraction Powder Patterns. In: *Circular of the Bureau of Standards* No. 539 1 (1953). DOI: [10.6028/NBS.CIRC.539v1](#).
- [343] Howard E. Swanson and George M. Ugrinic. Standard X-Ray Diffraction Powder Patterns. In: *Circular of the Bureau of Standards* No. 539 5 (1955). DOI: [10.6028/NBS.CIRC.539v5](#).
- [344] Ka H. Hong et al. Synthesis, Crystal Structure, and Magnetic Properties of MnFe_3O_5 . In: *Zeitschrift für anorganische und allgemeine Chemie* **642** (2016). DOI: [10.1002/zaac.201600365](#).
- [345] Sangsuk Lee et al. Antiferromagnet IrMn thickness dependence in exchange-biased perpendicular magnetic anisotropy based on $\text{CoFe/Pt/CoFe}[\text{IrMn}(\text{tIrMn})]$ multilayers. In: *Journal of Magnetism and Magnetic Materials* **304** (2006). DOI: [10.1016/j.jmmm.2006.01.188](#).
- [346] Chun Feng et al. Giant Strain Control of Antiferromagnetic Moment in Metallic FeMn by Tuning Exchange Spring Structure. In: *Advanced Functional Materials* **30** (2020). DOI: [10.1002/adfm.201909708](#).
- [347] Paul M. Marcus, Shen L. Qiu, and Victor L. Moruzzi. The mechanism of antiferromagnetism in chromium. In: *Journal of Physics: Condensed Matter* **10** (1998). DOI: [10.1088/0953-8984/10/29/014](#).
- [348] Patrick Bruno. Theory of interlayer exchange interactions in magnetic multilayers. In: *Journal of Physics: Condensed Matter* **11** (1999). DOI: [10.1088/0953-8984/11/48/305](#).
- [349] Edmund C. Stoner. Collective Electron Ferromagnetism. In: *Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences* **165** (1938). DOI: [10.1098/rspa.1938.0066](#).
- [350] Edmund C. Stoner. Collective electron specific heat and spin paramagnetism in metals. In: *Proceedings of the Royal Society of London. Series A-Mathematical and Physical Sciences* **154** (1936). DOI: [10.1098/rspa.1936.0075](#).
- [351] Allan H. MacDonald et al. Influence of relativistic contributions to the effective potential on the electronic structure of Pd and Pt. In: *Phys. Rev. B* **23** (1981). DOI: [10.1103/PhysRevB.23.6377](#).
- [352] Thi H. Ho, Sonny H. Rhim, and Soon C. Hong. Role of quantum well in Pd(111) thin film magnetism. In: *Journal of Magnetism and Magnetic Materials* **563** (2022). DOI: [10.1016/j.jmmm.2022.169891](#).
- [353] Takenao Shinohara, Tetsuya Sato, and Tomoyasu Taniyama. Surface ferromagnetism of Pd fine particles. In: *Physical Review Letters* **91** (2003). DOI: [10.1103/PhysRevLett.91.197201](#).
- [354] Victor L. Moruzzi and Paul M. Marcus. Magnetism in fcc rhodium and palladium. In: *Phys. Rev. B* **39** (1989). DOI: [10.1103/PhysRevB.39.471](#).
- [355] Stuart S. P. Parkin, R. Bhadra, and Kevin P. Roche. Oscillatory magnetic exchange coupling through thin copper layers. In: *Phys. Rev. Lett.* **66** (1991). DOI: [10.1103/PhysRevLett.66.2152](#).
- [356] Susanne Mirbt, Börje Johansson, and Hans L. Skriver. Quantum-well-driven magnetism in thin films. In: *Phys. Rev. B* **53** (1996). DOI: [10.1103/PhysRevB.53.R13310](#).

- [357] Anders M. N. Niklasson et al. Quantum-well-induced ferromagnetism in thin films. In: *Phys. Rev. B* **56** (1997). DOI: [10.1103/PhysRevB.56.3276](#).
- [358] Jiri Mathon et al. Quantum oscillations of the spin density in magnetic multilayers. In: *Phys. Rev. B* **59** (1999). DOI: [10.1103/PhysRevB.59.6344](#).
- [359] Salah Blizak et al. Interlayer exchange coupling between FeCo and Co ultrathin films through Rh(001) spacers. In: *Phys. Rev. B* **91** (2015). DOI: [10.1103/PhysRevB.91.014408](#).
- [360] Charles R. Harris et al. Array programming with NumPy. In: *Nature* **585** (2020). DOI: [10.1038/s41586-020-2649-2](#).
- [361] Till Tantau. *The TikZ and PGF Packages. Manual for version 3.0.0*. 2013.
- [362] John D. Hunter. Matplotlib: A 2D graphics environment. In: *Computing in Science and Engineering* **9** (2007). DOI: [10.1109/MCSE.2007.55](#).
- [363] Thomas A. Caswell et al. matplotlib/matplotlib: REL: v3.4.3. Zenodo. 2021. DOI: [10.5281/ZENODO.5194481](#).
- [364] Michael L. Waskom. seaborn: statistical data visualization. In: *Journal of Open Source Software* **6** (2021). DOI: [10.21105/joss.03021](#).
- [365] Skipper Seabold and Josef Perktold. Statsmodels: Econometric and statistical modeling with Python. In: *9th Python in Science Conference*. 2010. DOI: [10.25080/MAJORA-92BF1922-011](#).
- [366] Pauli Virtanen et al. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. In: *Nature Methods* **17** (2020). DOI: [10.1038/s41592-019-0686-2](#).

Acknowledgements

First of all, I would like to thank my supervisor, Prof. Dr. Stefan Blügel, for giving me the opportunity to work on a cutting-edge topic that combines theoretical solid-state physics but simultaneously includes state-of-the-art approaches from data science and machine learning which gave me the opportunity to learn and grow during this project in a hands-on fashion. This enabled me to fail and learn from every failure. Building an oasis in the desert has become a metaphor during this project. Stefan also encouraged and enabled me to present the results I gathered during my project at various conferences and occasions, which I highly appreciated. Furthermore, I would like to thank Prof. Dr. Matthias Wuttig and Prof. Dr. Ira Assent for offering as examiners and referees for this project.

The outstanding support of Dr. Daniel Wortmann has to be mentioned with gratefulness. Despite being a very sought-after man, Daniel contributed to the success of this thesis by many in-depth discussions with his experience and patience.

I also want to mention the assistance of the administrative and technical staff of the Forschungszentrum Jülich. In particular, Ute Winkler, Yasmin Ergüven, and Daniel Kaiser must be mentioned as essential. They kept the processes within the institute running even during a global pandemic.

I am pleased to thank Alexander Neukirchen for being the best imaginable office mate who not only shared this fantastic time with me but also did so during our master thesis and the car rides from Aachen to Jülich and back most of the days. In this sense, I also want to mention Uriel Aceves, who proved himself to be an excellent complement to the “Qubit”.

Furthermore, I want to mention my appreciation for Hiroshi Katsumoto's motivating words on the challenges of my doctoral project during our joint visit to the American Physical Society March Meeting 2023.

For their extraordinary support, I acknowledge those who contributed to this thesis by proof-reading parts of the manuscript, namely Dr. Daniel Wortmann, Dr. Philipp Rüßmann, Dr. Gregor Michalicek, M. Sc. Johannes Wasmer, Dipl. Ing. Frank Reiter, M. Edu. Tim Mattick, M. Sc. Alexander Neukirchen, B. Sc. Katrin Reiter, and Dr. Hiroshi Katsumoto.

This work was performed as part of the Helmholtz School for Data Science in Life, Earth and Energy (HDS-LEE) and received funding from the Helmholtz Association of German Research Centres. I also want to acknowledge the computing time granted by the JSC (Project: fleur4thc) on the supercomputer JURECA-DC at Forschungszentrum Jülich and also the computing resources granted by RWTH Aachen University (Projects: jara0234 and jara0182) on CLAIX, which both were essential to acquire the data analyzed and discussed in this thesis.

The ML applications, data analysis, data management, and visualization tasks presented in this thesis were performed using dedicated open-source software package implementations of various algorithms and frameworks. These packages are acknowledged here [9–11, 48, 50, 114, 125, 249, 265, 285, 360–366].

I also want to thank my collaborators and (former) colleagues for various fruitful discussions about the respective methodologies covered in this thesis. Besides discussions with Daniel Wortmann, in particular, the exchange with Ka Ming Law, Vasily Tseplyaev, Fabian Lux, and Dirk Witthaut inspired me to keep experimenting with new methods and approaches during the past three years. Along these lines, I thank Christian-Roman Gerhorst for sharing his PhD thesis' $\text{\LaTeX}$ template with me.

At last, but with the utmost value, I thank my grandmother, Ilse Hilgers, for her continuous support during the past years, during which I could not have asked for better backing. In this context, it is indispensable to thank my love, Katrin, for always having my back and contributing to this project with your unwavering patience and consideration.

Eidesstattliche Erklärung

Robin Alexander Hilgers, M. Sc. RWTH

erklärt hiermit, dass diese Dissertation und die darin dargelegten Inhalte die eigenen sind und selbstständig, als Ergebnis der eigenen originären Forschung, generiert wurden.

Hiermit erkläre ich an Eides statt:

- Diese Arbeit wurde vollständig oder größtenteils in der Phase als Doktorand dieser Fakultät und Universität angefertigt;
- Sofern irgendein Bestandteil dieser Dissertation zuvor für einen akademischen Abschluss oder eine andere Qualifikation an dieser oder einer anderen Institution verwendet wurde, wurde dies klar angezeigt;
- Wenn immer andere eigene- oder Veröffentlichungen Dritter herangezogen wurden, wurden diese klar benannt;
- Wenn aus anderen eigenen- oder Veröffentlichungen Dritter zitiert wurde, wurde stets die Quelle hierfür angegeben. Diese Dissertation ist vollständig meine eigene Arbeit, mit der Ausnahme solcher Zitate;
- Alle wesentlichen Quellen von Unterstützung wurden benannt;
- Wenn immer ein Teil dieser Dissertation auf der Zusammenarbeit mit anderen basiert, wurde von mir klar gekennzeichnet, was von Anderen und was von mir selbst erarbeitet wurde;
- Teile dieser Arbeit wurden zuvor veröffentlicht, und zwar in:
 - Robin Hilgers, Daniel Wortmann, and Stefan Blügel. Application of batch learning for boosting high-throughput *ab initio* success rates and reducing computational effort required using data-driven processes. In: *arXiv: cond-mat.mtrl-sci* (2023). Preprint. DOI: [10.48550/arXiv.2311.15430](https://doi.org/10.48550/arXiv.2311.15430)
 - Robin Hilgers, Daniel Wortmann, and Stefan Blügel. Machine Learning-based estimation and explainable artificial intelligence-supported interpretation of the critical temperature from magnetic *ab initio* Heusler alloys data. In: *arXiv: cond-mat.mtrl-sci* (2023). Preprint. DOI: [10.48550/arXiv.2311.15423](https://doi.org/10.48550/arXiv.2311.15423)

Aachen, den 1. Oktober 2024

Unterschrift

Curriculum Vitae & Publication List

Robin Alexander Hilgers

Email: robin.hilgers@rwth-aachen.de

Born: August 11th, 1995 in Neuss, Germany

Nationality: German

Current position

PhD Candidate,
Peter Grünberg Institute and Institute for Advanced Simulation,
Forschungszentrum Jülich
Since September 2020

Education

- 2015 A-levels, Gesamtschule Barmen, Wuppertal
- 2018 B. Sc. RWTH in Physics, RWTH Aachen, Aachen
- 2020 M. Sc. RWTH in Physics with a focus on theoretical & computational Physics, RWTH Aachen, Aachen
- *Since 2020* Graduate student within the Helmholtz School for Data Science in Life, Earth and Energy, Jülich

Publications & Talks

Articles

- Robin Hilgers, Daniel Wortmann, and Stefan Blügel. Machine Learning-based estimation and explainable artificial intelligence-supported interpretation of the critical temperature from magnetic *ab initio* Heusler alloys data. In: *arXiv: cond-mat.mtrl-sci* (2023). Preprint. doi: [10.48550/arXiv.2311.15423](https://doi.org/10.48550/arXiv.2311.15423)
- Robin Hilgers, Daniel Wortmann, and Stefan Blügel. Application of batch learning for boosting high-throughput *ab initio* success rates and reducing computational effort required using data-driven processes. In: *arXiv: cond-mat.mtrl-sci* (2023). Preprint. doi: [10.48550/arXiv.2311.15430](https://doi.org/10.48550/arXiv.2311.15430)

For both articles a journal submission is planned.

Publicly Available Reports

- Roman Kováčik and Robin Hilgers. *Properties of magnetic materials calculated using a high throughput framework for ab-initio Korringa-Kohn-Rostoker Green function method*. CLAIX Projects 2020 High Performance Computing at RWTH Aachen University. Aachen, Germany: IT Center, RWTH Aachen University, 2020

Data & Code Publications

- Robin Hilgers, Daniel Wortmann, and Stefan Blügel. Data processing for the JuHemd database and ML-training and evaluation scripts. Zenodo. 2023. doi: [10.5281/ZENODO.8436741](https://doi.org/10.5281/ZENODO.8436741)
- Robin Hilgers, Daniel Wortmann, and Stefan Blügel. ML-ready Curie temperatures and descriptors extracted from the JuHemd database. Materials Cloud Archive. 2022. doi: [10.24435/materialscloud:W1-YF](https://doi.org/10.24435/materialscloud:W1-YF)
- Daniel Wortmann, Gregor Michalícek, Nadjib Baadji, Markus Betzinger, Gustav Bihlmayer, Jens Bröder, Tobias Burnus, Jussi Enkovaara, Frank Freimuth, Christoph Friedrich, Christian-Roman Gerhorst, Sabastian Granberg Cauchi, Uliana Grytsiuk, Andrea Hanke, Jan-Philipp Hanke, Marcus Heide, Stefan Heinze, **Robin Hilgers**, Henning Janssen, Daniel Aaron Klüppelberg, Roman Kovacic, Philipp Kurz, Marjana Ležaić, Georg K. H. Madsen, Yuriy Mokrousov, Alexander Neukirchen, Matthias Redies, Stefan Rost, Martin Schlipf, Arno Schindlmayr, Miriam Winkelmann, and Stefan Blügel. FLEUR. Zenodo. 2023. doi: [10.5281/ZENODO.7576163](https://doi.org/10.5281/ZENODO.7576163)
- Robin Hilgers, Daniel Wortmann, and Stefan Blügel. Code collection for machine learning assisted materials screening in the search for novel half-metallic Heusler alloys. Zenodo. 2023. doi: [10.5281/ZENODO.10075055](https://doi.org/10.5281/ZENODO.10075055)
- Robin Hilgers, Daniel Wortmann, and Stefan Blügel. Density of states of full and inverse Heusler magnetic alloys. Materials Cloud Archive. 2023. doi: [10.24435/materialscloud:v5-5z](https://doi.org/10.24435/materialscloud:v5-5z)

- Robin Hilgers, Daniel Wortmann, and Stefan Blügel. Workflow, Data Processing, Data Analysis and predictive ML scripts used for DFT integrated Machine Learning methodology in combination with the FLAPW code FLEUR. Zenodo. 2023. doi: [10.5281/ZENODO.10056614](https://doi.org/10.5281/ZENODO.10056614)
- Robin Hilgers, Daniel Wortmann, and Stefan Blügel. Relaxed thin film structures of one, two, and three magnetic 3d transition metal layers on FCC noble-metal substrates based on FLAPW PBE calculations. Materials Cloud Archive. 2023. doi: [10.24435/materialscloud:dk-wq](https://doi.org/10.24435/materialscloud:dk-wq)

Conferences, Workshops, and Contributions

- Robin Hilgers. Workshop Talk: Magnetism in FLEUR. Virtual FLEUR Workshops, Forschungszentrum Jülich, Jülich, Germany. 2021 and 2022
- Psi-k Conference, Lausanne, Switzerland. 2022
- Robin Hilgers, Daniel Wortmann, Stefan Blügel, Dirk Witthaut, and Ira Assent. Poster: Curie Temperature Prediction of Magnetic Heusler Alloys Using Ab-initio Data. HDS-LEE Midterm Evaluation, Jülich, Germany. URL: <http://hdl.handle.net/2128/34320>. 2023
- Robin Hilgers, Daniel Wortmann, Stefan Blügel, and Roman Kováčik. Talk: Curie Temperature Prediction Models of Magnetic Heusler Alloys Using Machine Learning Methods Based on First-Principles Data From Ab-initio KKR-GF Calculations. APS March Meeting, Las Vegas, NV, USA. URL: <http://hdl.handle.net/2128/34323>. 2023
- Robin Hilgers, Daniel Wortmann, and Stefan Blügel. Poster: Application of Machine-Learning Models and XAI in Materials Science Using Magnetic DFT Data. Helmholtz.AI, Hamburg, Germany. 2023. doi: [10.34734/FZJ-2023-02298](https://doi.org/10.34734/FZJ-2023-02298)
- Robin Hilgers, Daniel Wortmann, and Stefan Blügel. Talk: Magnetic Multilayers: From High-Throughput Ab-initio Calculations to Predictive Machine Learning. CMD30 FisMat2023, Milan, Italy. 2023. doi: [10.34734/fzj-2023-03694](https://doi.org/10.34734/fzj-2023-03694)

Band / Volume 274

Multicellular defense against phage infection in *Streptomyces* – impact of secondary metabolites and mycelial development

L. Kever (2023), iv, 246 pp

ISBN: 978-3-95806-724-0

Band / Volume 275

Investigation of the electronic band structure of 2D transition metal dichalcogenides via angle-resolved photoemission spectroscopy

B. Parashar (2023), xvii, 156 pp

ISBN: 978-3-95806-725-7

Band / Volume 276

Strain- and process engineering for polyketides production with *Pseudomonas taiwanensis* VLB120 in two-phase cultivations

T. P. Schwanemann (2023), 230 pp

ISBN: 978-3-95806-726-4

Band / Volume 277

Quantitative atomic-level investigation of solid materials through multidimensional electron diffraction measurements

H. L. Lalandec-Robert (2024), xxi, 152 pp

ISBN: 978-3-95806-735-6

Band / Volume 278

Studies on the cAMP-responsive regulatory network of *Corynebacterium glutamicum*

N. Wolf (2024), iii, 122 pp

ISBN: 978-3-95806-736-3

Band / Volume 279

Rare-earth atoms on two-dimensional materials: ab initio investigation of magnetic properties

J. P. Carbone (2024), 235 pp

ISBN: 978-3-95806-740-0

Band / Volume 280

Communities of Niche-optimized Strains (CoNoS) – a novel concept for improving biotechnological production

R. Zuchowski (2024), VIII, 168 pp

ISBN: 978-3-95806-743-1

Band / Volume 281

Enabling mixed microbial upcycling of plastic monomers

Y. S. Ackermann (2024), XVI, 203 pp

ISBN: 978-3-95806-749-3

Band / Volume 282

**Folding and structural studies of *saccharomyces cerevisiae*
Phosphoglycerate Kinase**

N. Bustorff (2024), xxvi, 126 pp

ISBN: 978-3-95806-754-7

Band / Volume 283

**The role of cellular development in multicellular antiphage
defense of *Streptomyces***

T. Luthe (2024), vi, 173 pp

ISBN: 978-3-95806-768-4

Band / Volume 284

**Probing the Transformation from Transition Metal Complexes to Extended
Two-Dimensional Nanostructures**

D. Baranowski (2024), XII, 103 pp

ISBN: 978-3-95806-772-1

Band / Volume 285

Neutron Scattering

Lectures of the JCMS Laboratory Course held at Forschungszentrum Jülich
and at the Heinz-Maier-Leibnitz Zentrum Garching

edited by S. Förster, K. Friese, M. Kruteva, S. Nandi, M. Zobel, R. Zorn (2024),
ca. 365 pp

ISBN: 978-3-95806-774-5

Band / Volume 286

***Ab initio* investigation of intrinsic antiferromagnetic solitons**

Amal Jawdat Nayef Aldarawsheh (2024), xv, 164 pp

ISBN: 978-3-95806-785-1

Band / Volume 287

**Understanding the dynamics of Plant-Bacteria-Bacteriophage
interactions as a means to improve plant performance**

S. H. Erdrich (2024), ix, 176 pp

ISBN: 978-3-95806-791-2

Band / Volume 288

**Prediction of Magnetic Materials for Energy and Information
Combining Data-Analytics and First-Principles Theory**

R. Hilgers (2024), xv, 215 pp

ISBN: 978-3-95806-795-0

Weitere **Schriften des Verlags im Forschungszentrum Jülich** unter
<http://wwwzb1.fz-juelich.de/verlagextern1/index.asp>

Schlüsseltechnologien / Key Technologies
Band / Volume 288
ISBN 978-3-95806-795-0