ECONtribute
MARKETS & PUBLIC POLICY

ECONtribute
Discussion Paper No. 078

The Risk of Algorithm Transparency: How Algorithm Complexity Drives the Effects on Use of Advice

Christiane B. Haubitz
Cedric A. Lehmann
Andreas Fügener
Ulrich W. Thonemann

April 2021

www.econtribute.de

UNIVERSITÄT BONN
Universität zu Köln

Funding by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany´s Excellence Strategy – EXC 2126/1– 390838866 is gratefully acknowledged.

The Risk of Algorithm Transparency: How Algorithm Complexity Drives the Effects on Use of Advice

Christiane B. Haubitz, Cedric A. Lehmann, Andreas Fügener, Ulrich W. Thonemann
Department of Supply Chain Management and Management Science, University of Cologne, 50923 Cologne, Germany,
christiane.haubitz@uni-koeln.de, cedric.lehmann@uni-koeln.de, andreas.fuegener@uni-koeln.de, ulrich.thonemann@uni-koeln.de

Algorithmic decision support is omnipresent in many managerial tasks, but human judgment often makes the final call. A lack of algorithm transparency is often stated as a barrier to successful human-machine collaboration. In this paper, we analyze the effects of algorithm transparency on the use of advice from algorithms with different degrees of complexity. We conduct a preregistered laboratory experiment where participants receive identical advice from algorithms with different levels of transparency and complexity. The results of the experiment show that increasing the transparency of a simple algorithm reduces the use of advice, while increasing the transparency of a complex algorithm increases it. Our results also indicate that the individually perceived appropriateness of algorithmic complexity moderates the effects of transparency on the use of advice. While perceiving an algorithm as too simple severely harms the use of its advice, the perception of an algorithm being too complex has no significant effect on it. Our results suggest that managers do not have to be concerned about revealing complex algorithms to decision makers, even if the decision makers do not fully comprehend them. However, making simple algorithms transparent bears the risk of disappointing people's expectations, which can reduce the use of algorithms' advice.

Key words: Algorithm Transparency; Decision Making; Decision Support; Use of Advice

1. Introduction

In recent decades, advances in data availability and computational power have resulted in an increasing use of algorithms for day-to-day decision-making. People increasingly rely on complex 'black-box' algorithms that reveal little about their underlying principles and often only provide a final recommendation. In various decision-making domains, complex algorithms, such as machine learning models, have been successfully employed to improve decision quality (Brynjolfsson and McAfee 2014, LeCun et al. 2015).

However, there is also a belief that the black-box nature of such algorithms results in a low acceptance of advice (Burton et al. 2020), an issue that can be addressed by explaining the underlying algorithmic principles (Glikson and Woolley 2020). This observation motivated a project involving the service division of a large equipment manufacturer, for which we developed a spare parts inventory optimization tool. The company decided to make the algorithmic principles transparent to users because they expected greater adherence to the tool's recommendations with transparency than without. The algorithm is now used by planners in over 80 country organizations. The feedback indicates that the planners appreciate the transparency, but it remains an open question whether the positive perception of transparency actually increases the use of advice compared to a less transparent tool. To explore this issue, we conducted behavioral experiments, which are an emerging and promising methodology in IS research (Gupta et al. 2018).

We asked 450 undergraduate students in an operations management class about their attitudes towards the use of algorithms in managerial decisions. The vast majority (94.2 %) indicated that they would like to receive algorithmic advice when facing a managerial task. Approximately as many (94.9 %) wanted to be informed about the algorithmic procedures. Although the questions were of a hypothetical nature, they indicate a preference for transparent algorithmic advice.

In the same class, we conducted an experiment related to the inventory management problem faced by the company that motivated our research. For 15 different spare parts sets, the participants had to decide which parts to stock, considering part demand rates, part weights, and a total weight limit — a classical combinatorial optimization task known as the 'knapsack-problem' (Dantzig 1957). The problem can be solved by a greedy algorithm, which sorts the spare parts by their value-to-weight ratio and selects the part with the highest ratio as long as the capacity is not exceeded. Although this procedure does not necessarily optimally solve the problem, it usually produces good results (Diubin and Korbut 2008). In our experiment, the greedy algorithm's solutions were optimal. We divided the students into two groups. Both groups received the same algorithmic advice, but one group received an explanation of the algorithmic principles, while the other did not. The group that received the explanation followed the algorithmic advice less often than the

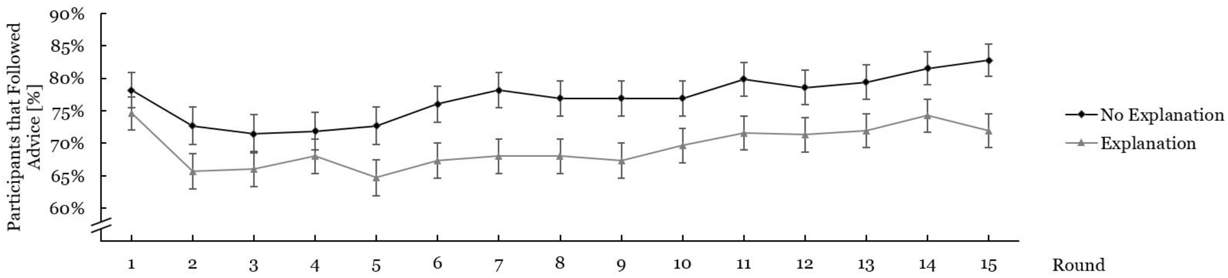


Figure 1 **Share of participants who followed the algorithmic advice by round (randomized sequence of spare parts sets). Error bars indicate standard errors.**

group that did not receive an explanation (Figure 1). We provide further details on the experiment in Online Appendix EC.1.

The results are somewhat surprising. The participants reported that they would like to receive algorithmic advice and that they would appreciate transparency about the algorithm. However, when they were provided with both for the relatively simple algorithm that we used, they followed its advice less with transparency than without. These observations raise questions regarding the value of transparency: Does greater transparency generally reduce the use of advice, or does the relationship depend on factors such as an algorithm's complexity? In other words, is complexity a moderator for the effect of algorithm transparency on the use of advice? This issue has received little attention in the literature, and we address it in this paper.

We analyze the effects of the transparency of algorithms with different complexities on the use of advice in managerial decision tasks. We conduct a preregistered laboratory experiment where we manipulate the level of transparency by informing only a subgroup of participants about the underlying principles of different advice-giving algorithms. To test whether transparency about simple algorithms has a different effect than transparency about more complex algorithms, we also vary the degree of algorithm complexity. We provide explanations of two different algorithms that vary in their underlying principles but generate identical advice. The results indicate that the effect of transparency on the use of algorithmic advice depends on algorithm complexity. While making complex algorithms transparent does not reduce the use of advice, making simple algorithms transparent does. Our results provide a better understanding of the benefits and risks of algorithm transparency and indicate when managers should and should not make algorithms transparent and when not.

2. Literature Review

Decision makers appreciate receiving advice to improve decision quality and share the responsibility for possible negative consequences (Harvey and Fischer 1997, Yaniv 2004b,a). Research on advice-taking and decision-making is broad and we refer to Bonaccio and Dalal (2006) for a comprehensive review. We focus on drivers of the use of advice and specifically on the role of algorithm transparency in this context. We then review advice-taking in demand forecasting, a critical managerial task that we consider in our experiments.

2.1. The Use of Algorithmic Advice

The advice-taking literature suggests that receiving advice often improves decisions (Sniezek et al. 2004, Yaniv 2004a). However, decision makers tend to not follow advice as much as would be in their interest (Bonaccio and Dalal 2006) and place insufficient weight on the advice (Dietvorst et al. 2015, Harvey and Fischer 1997, Logg et al. 2019, Yaniv 2004b,a). In a cue-learning task, Harvey and Fischer (1997) observe a weight of only 20% to 30% on human advice, even if the advisor has more expertise than the decision maker. In different estimation tasks, Logg et al. (2019) observe a weight of 34% to 52% on algorithmic advice.

Yu et al. (2019) suggest that people use algorithmic advice based on its perceived performance. Even without revealed performance measures, humans are capable of detecting the quality of algorithmic advice through outcomes and adapting their use accordingly (De Baets and Harvey 2020). Dietvorst et al. (2015) observe that people lose confidence in algorithms over time when they see algorithms err. If similar advice is provided by a human, confidence in their advice does not fade as quickly. They refer to this phenomenon as “algorithm aversion”.

There are different beliefs about whether the black-box nature of algorithms is harmful for advice utilization. Some studies indicate that people are not opposed to taking advice from algorithms without having much information on their underlying principles. For example, Logg et al. (2019) show that people are generally willing to work with black-box algorithms. Dietvorst et al. (2015)

find no contrary evidence, as participants in one of their studies prefer to receive advice from a black-box algorithm over taking advice from a human when there is no direct feedback. However, Burton et al. (2020) argue that opaque algorithms result in a low acceptance of advice. Glikson and Woolley (2020) suggest that this could be addressed by transparency of algorithmic principles, which can increase the trust in algorithms.

The level of transparency moderates the extent to which the underlying principles of an algorithm are visible to the decision maker. A high level of transparency typically reveals the data used and explains the procedures that transform data into advice (Bertino et al. 2019, Dhaliwal and Benbasat 1996). It therefore also uncovers the complexity of applied principles. Previous research indicates that a greater algorithmic complexity might be associated with higher trust in an algorithm (Glikson and Woolley 2020).

The literature discusses the effects of algorithm transparency on use of algorithmic advice. Previous research has focused on subjective tasks that involve personal taste. In an early study, Sinha and Swearingen (2002) conclude that people appreciate music recommender systems that they perceive as transparent and exhibit higher confidence in their advice. Similarly, Wang and Benbasat (2007) show that explaining how an e-commerce recommender system derived a product recommendation increased the users' trust in the technical competence of the recommender agent. Cramer et al. (2008) analyze the effects of transparency of an art recommender system. While the explanation of why certain artwork is recommended increased the acceptance of the recommendations, it did not affect the general trust in the recommender system. Kizilcec (2016) examines the effects of transparency of a grading algorithm and finds that revealing excessive information can have a negative effect on trust in an algorithm. Springer and Whittaker (2018) analyze the effects of the transparency of an algorithm that predicts the users' mood based on a short self-written text about a past emotional experience. Increased transparency led to reduced perceived accuracy, even when expectations were met.

The results of these studies point in different directions. While earlier work indicates that transparency has a positive effect on trust (Wang and Benbasat 2007), appreciation, and confidence

(Sinha and Swearingen 2002), later studies reveal mixed (Cramer et al. 2008, Kizilcec 2016) or even negative effects (Springer and Whittaker 2018) on trust or perceived accuracy. The studies were conducted with subjective tasks, such as product recommendations or grading. For objective tasks, the effects of algorithm transparency on the use of advice have received little attention. Moreover, most studies measure the perception of algorithmic advice with indicators such as trust, perceived accuracy, or algorithm appreciation without linking it to the actual use of advice. They also do not consider the role of algorithm complexity. This paper addresses these gaps and analyzes the effect of algorithm transparency on use of algorithmic advice for an objective managerial task, that is, demand forecasting.

2.2. Advice-taking in Demand Forecasting

Demand forecasting is a critical task for organizations. It is also a commonly analyzed task in behavioral operations management (Lawrence et al. 2006, Kremer et al. 2011, Moritz et al. 2014). Forecasting tasks have also been used to analyze the general human attitude towards algorithms (Dietvorst et al. 2015, 2018, Logg et al. 2019).

Forecasting algorithms have been continuously improved (Fildes 2006), and they have been increasingly used in practice (Fildes and Petropoulos 2015). Nevertheless, most companies do not entirely rely on algorithmic forecasts but include human judgment in their forecasting routines (e.g., Fildes and Goodwin (2007), Fildes and Petropoulos (2015), Sanders and Manrodt (2003), Klassen and Flores (2001)). This approach is referred to as judgmental forecasting.

Research on judgmental forecasting analyzes how to efficiently combine human judgment and algorithms (Webby and O'Connor 1996, Perera et al. 2019, Arvan et al. 2019). While human forecasters can be inconsistent, prone to biases, and affected by wrong incentives (Fildes and Goodwin 2007, Fildes et al. 2009), forecasting algorithms are based on a fixed set of rules that might not adapt to specific situations. Most frequently applied are judgmental adjustments of algorithmic advice (Perera et al. 2019, Arvan et al. 2019), where statistically generated forecasts are manually adjusted by the forecaster, for example to include intuition, additional information, and expertise (e.g., Webby and O'Connor (1996), Fildes and Goodwin (2007), Franses and Legerstee (2011)).

Depending on the characteristics of the decision maker, the situation, the quality of the algorithm, and the information available, these adjustments improve or impair forecast accuracy (e.g., Fildes et al. (2009), Alvarado-Valencia et al. (2017), Fildes et al. (2019)).

De Baets and Harvey (2020) find that people rely more on good forecasting models than on poor forecasting models but not as much as they should. This is in line with Goodwin and Fildes (1999), who found that statistical forecasts, despite their quality and appropriateness, are often underutilized. This has been attributed to general distrust towards forecasting advice (Goodwin et al. 2013) and algorithm aversion (Dietvorst et al. 2015). While there is some indication that "lack of transparency in [...] underlying processes can contribute to a forecaster's distrust in systems" (Arvan et al. 2019), little is known about the effect of algorithm transparency on use of algorithmic advice or about the role of algorithm complexity in this relationship. We address these issues in this paper.

3. Development of Hypotheses

The effects of algorithm transparency on human decisions depend on the decision makers' expectations (Burton et al. 2020, Kizilcec 2016, Springer and Whittaker 2019). Logg et al. (2019) survey the expectations of human decision makers towards the approaches used by algorithms (survey of 226 respondents using thematic coding). A plurality, 42% of the respondents, believes that an algorithm is a set of mathematical equations, 26% regard it as a step-by-step procedure, and 14% expect an algorithm to be a single formula or logic. These results indicate that many humans expect some algorithm complexity, even if such complexity is not always necessary to provide good advice and to outperform human judgment (Fischer and Harvey 1999). We define *simple* and *complex* algorithms as approaches that tend to be below and above the expectations surveyed by Logg et al. (2019). A *simple algorithm* is a mathematical approach that can be easily understood by a typical user, such as a simple formula, and a *complex algorithm* is a sophisticated approach that is difficult for a typical user to understand, such as an artificial neural network.

Building on the insights of Logg et al. (2019), we expect the majority of people to be disappointed and underwhelmed by the nature of simple algorithms when they become aware of their complexity, resulting in poor perception of advice. Because a poor perception of advice leads to a low use of advice (Yaniv 2004b), we hypothesize that the transparency of simple algorithms decreases the use of advice compared to non-transparent algorithms:

Hypothesis 1. *Increasing the transparency of simple algorithms reduces the use of algorithmic advice.*

On the other hand, we expect that complex algorithms tend to exceed the complexity expectations of the majority of human decision makers. Parasuraman and Manzey (2010) and Dzindolet et al. (2003) report a positive attitude towards advanced technologies, and we hypothesize that humans appreciate high algorithmic complexity such that making a complex algorithm transparent leads to an increased use of advice:

Hypothesis 2. *Increasing the transparency of complex algorithms increases the use of algorithmic advice.*

Hypotheses 1 and 2 state opposing effects of transparency on the use of advice for algorithms that either go beyond or fall short of humans' general expectations. The survey of Logg et al. (2019) shows that expectations towards algorithms vary widely. Decision makers with different expectations may have different perceptions of the capability of an algorithm to provide meaningful advice for a decision task. One decision maker can perceive an algorithm as appropriate to solve a task while another decision maker perceives the same algorithm as inappropriate to solve the same task.

Therefore, we analyze the effects of algorithmic transparency on the use of advice with respect to decision makers' individual perceptions of the appropriateness of algorithmic complexity. We analyze whether a deviation of the perceived algorithm complexity from an appropriate level of complexity affects the value of advice and the use of advice. Given the lack of literature on this relationship, we cannot formulate a clear hypothesis but follow the advice of Gupta et al. (2018) and inductively explore this issue to pave the way for new theories.

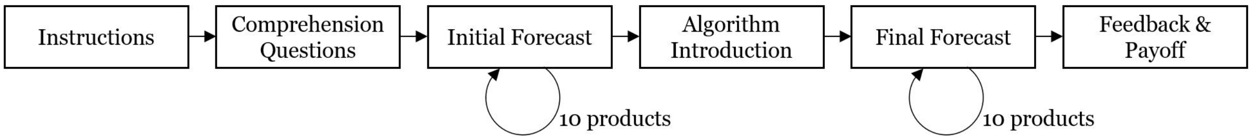


Figure 2 **Sequence of events within the experiment.**

4. Experimental Study

We consider a demand forecasting task and design an experiment with a judge-advisor system structure. First, the decision maker (the "judge") makes an initial demand forecast. Second, he or she receives advice from an algorithm. Third, the judge can update the initial forecast and provide a final forecast. Judge-advisor systems are often used in the literature to analyze human behavior in advice-taking (Yaniv 2004b, Gino 2008, Logg et al. 2019). The judgment is measured before and after receiving the advice, and the relative shift in judgment indicates the use of advice (Bonaccio and Dalal 2006). The laboratory experiment was preregistered at the Open Science Framework: [*link blinded for review*].[1]

4.1. Experimental Design

We designed an experiment in which participants had to forecast demand before and after they received a demand forecast from an algorithm. We used three treatments that differ in the transparency of the algorithmic procedures and in the complexity of the algorithms. In all treatments, the participants received identical numerical advice, but the information provided and the algorithmic procedures differed.

Figure 2 depicts the sequence of events in the experiment. After reading the instructions and answering comprehension questions, participants made their initial forecasts for 10 different products. For each product, they observed the demand history of 10 periods and forecasted the demand in period 11 by entering an integer number in an input field. We simulated stationary demand data for each product with a mean value between 300 and 1200 and a coefficient of variation of

[1] Note that in the course of writing this paper, we slightly adapted the formulation of our research question and the hypotheses after preregistration. Nevertheless, all preregistered hypotheses hold, and we summarize them in Online Appendix EC.2.

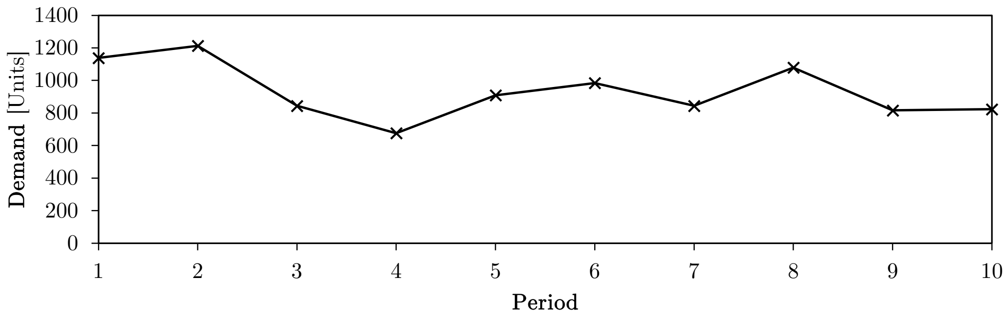


Figure 3 **Demand history of a product with a mean of 1000 and a coefficient of variation of 0.3.**

0.3. The demand history was displayed in a line chart (Figure 3), which is common in practice and perceived to be the best presentation style when the characteristics of the demand data are unknown (Harvey and Bolger 1996). All subjects received the same demand data.

After the participants had forecasted the demand for the 10 products, they were informed that an algorithm had also computed a forecast. Between the treatments, we varied the level of algorithm transparency and complexity. Participants could use the algorithmic advice for each product to update their initial forecast. After making their final forecasts, the participants were asked to indicate their agreement with the statement "I understood how the algorithm derived its recommendations" on a 7-point Likert-type scale (1 – strongly disagree; 4 – neither agree nor disagree; 7 – strongly agree). Furthermore, we asked the participants who received information about the algorithmic principles about the perceived appropriateness of algorithmic complexity to solve the forecasting task on a 7-point Likert-type scale (1 – much less than appropriate; 4 – appropriate; 7 – much more than appropriate).

At the end of the study, the participants observed the actual demand and their resulting forecast error. We monetarily incentivized high forecast accuracy using a payoff composition similar to that in Kremer et al. (2011). It consists of a fixed reward of \$0.50 and a bonus of up to \$0.10 per product, depending on the final forecast accuracy. The accuracy was measured with the absolute percentage error that we bounded by 0 and 1:

$$\text{absolute percentage error} = \frac{|\text{final forecast} - \text{actual demand}|}{\text{actual demand}}$$

The bonus for each product was $\$0.10 \cdot (1 - \text{absolute percentage error})$.

We measured the use of the algorithmic advice by the weight on advice (e.g., Bonaccio and Dalal 2006):

$$\text{weight on advice} = \frac{\text{initial forecast} - \text{final forecast}}{\text{initial forecast} - \text{algorithmic advice}}$$

The weight on advice computes the relative shift between the initial forecast and the final forecast with respect to the algorithmic advice. A weight on advice of 0 implies that the participant did not change his or her initial forecast; a value of 1 implies that the initial forecast was replaced by the algorithmic advice. If the initial forecast equals the algorithmic advice, we set the weight on advice to 0. Values less than 0 or greater than 1 are winsorized to increase interpretability (e.g., Logg et al. 2019). In our analyses, we use the participant's average weight on advice over all 10 products.

We used three treatments. Participants in all treatments received the same numerical advice, but the information provided and the algorithmic principles differed.

In the *non-transparent* treatment, participants were merely informed that an algorithm had computed a forecast and received no additional information on its underlying principles.

In the *transparent-simple* treatment, the algorithm determines the arithmetic mean demand of the 10 periods as forecast for period 11. This algorithmic forecast is an unbiased and ex ante error minimizing estimation for stationary demand. In the experiment, participants were informed that the algorithm "calculates the forecast for a product by computing the average of the demand history of the last 10 periods". We illustrated the computation of the average with the corresponding formula for an example product that was not used in the actual experiment.

In the *transparent-complex* treatment, the algorithm is a neural network that uses the demand history as input and provides the demand forecast for period 11 as output. It was trained with 100.000 structurally similar stationary demand histories. In the experiment, we explained the basic principles of an artificial neural network and provided the calculations necessary to comprehend and reproduce the recommended forecast.

While the underlying principles of the algorithms differ substantially, they generate the same algorithmic advice for the 10 different products. Detailed explanations for all treatments are provided in Online Appendix EC.3.3.

Table 1 Result overview – means of the relevant measures in the different treatments (standard error)

Measure	Non-Transparent	Transparent-Simple	Transparent-Complex
Appropriateness of Complexity	-	3.02 (0.10)	4.55 (0.10)
Understanding	3.65 (0.13)	6.36 (0.07)	4.64 (0.12)
Weight on Advice [%]	49.30 (1.72)	36.47 (2.13)	55.64 (2.07)
Performance (MAE)	111.24 (3.23)	135.00 (4.24)	106.79 (3.32)

4.2. Experimental Protocol

The experiment was programmed in oTree (Chen et al. 2016) and conducted on Amazon's Mechanical Turk (MTurk) on May 12 and May 13, 2020. We sought to collect data from 500 participants to detect a medium-sized effect ($f = 0.15$) at a power of 0.85 and $\alpha = 0.05$ (ANOVA with 3 groups). A total of 1322 MTurk workers agreed to work on the experiment. A total of 176 workers did not proceed to the comprehension questions, and 626 failed to answer them correctly on the first attempt, which was a requirement to work on the experiment. Eleven workers did not finish the experiment after having passed the comprehension questions. This left us with 509 participants. At the beginning of the experiment, the participants were randomly assigned to one of the three treatments. A total of 171 participants were assigned to the non-transparent treatment, 171 participants to the transparent-simple treatment, and 167 participants to the transparent-complex treatment. On average, the participants took 10.4 minutes to conduct the experiment and earned $1.36.

5. Results

Table 1 summarizes the main results of our experiment. We first check whether the manipulation of the different treatments was successful. Next, we analyze the effects of transparency on weight on advice of the simple and the complex algorithm compared to the non-transparent algorithm (Hypotheses 1 and 2). Then, we explore the perceived appropriateness of complexity as a moderator of the effect of algorithm transparency on the weight on advice. Unless stated otherwise, we conduct two-sided Mann-Whitney U tests with Holm-Bonferroni correction for multiple tests (Holm 1979).

5.1. Validation of Treatment Manipulation

To check whether the manipulation of the different treatments was successful, we analyze three questions: First, are the two algorithms intended to represent simple and complex algorithms also perceived to be "simple" or "complex"? Second, does providing transparency lead to a higher level of understanding of an algorithm? Third, is the simple algorithm better understood than the complex algorithm?

Table 1 shows that participants in the transparent-simple treatment rated the appropriateness of complexity on average as 3.02 and participants in the transparent-complex treatment rated it on average as 4.55. The ratings are different from one another ($p < 0.001$) and different from 4 ($p < 0.001$ for both tests), the value that corresponds to an appropriate level of complexity. This indicates that our simple and complex algorithms were also perceived as such by the participants.

The level of understanding of the algorithms' mechanisms was self-reported at the end of the experiment. Participants in the non-transparent treatment indicated a level of understanding of 3.65 on the 7-point Likert-type scale. The understanding in the transparent-simple treatment was 6.36 and significantly higher than that in the non-transparent treatment ($p < 0.001$). The understanding in the transparent-complex treatment was 4.64 and also significantly higher than that in the non-transparent treatment ($p < 0.001$) but significantly lower than that in the transparent-simple treatment ($p < 0.001$). We can conclude that transparency leads to a higher level of understanding and that the simple algorithm is better understood than the complex algorithm.

The results suggest that our manipulation was successful.

5.2. The Effects of Transparency of Simple & Complex Algorithms on Weight on Advice

To test our hypotheses, we analyze the difference in weight on advice across treatments (Figure 4-a). In the non-transparent treatment, participants have a weight on advice of 49.3%. In the transparent-simple treatment, the weight on advice is 36.5% and significantly lower than that of the non-transparent treatment ($p < 0.001$), which provides support for Hypothesis 1. In the transparent-complex treatment, the weight on advice is 55.6% and significantly higher than that of the non-transparent treatment ($p = 0.019$), which provides support for Hypothesis 2.

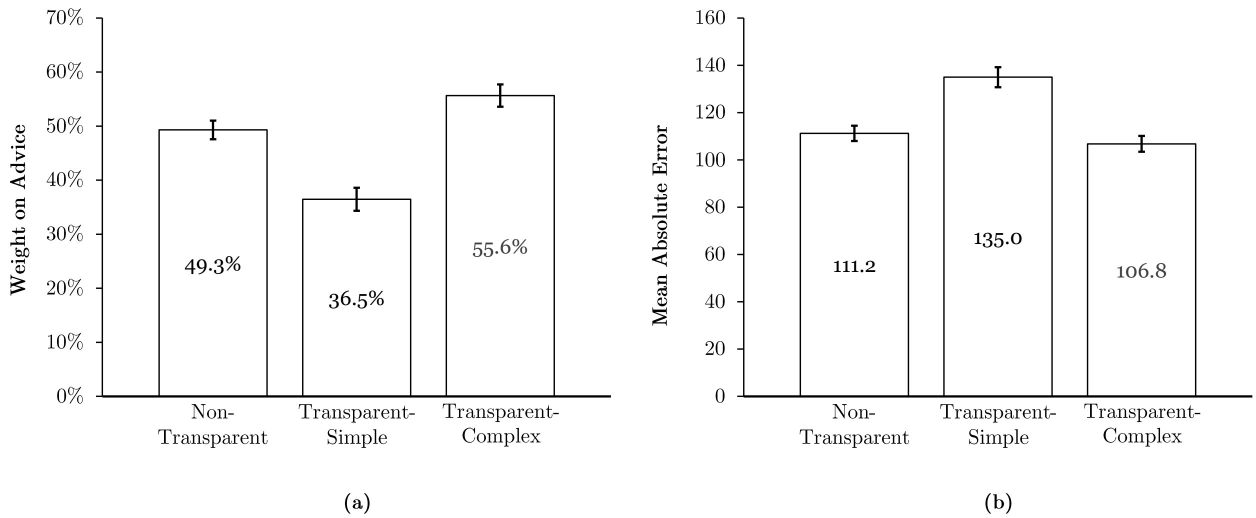


Figure 4 **Comparison of average (a) weight on advice and (b) mean absolute error for the three treatments. Error bars indicate standard errors.**

As a robustness check, we conduct an ordinary least squares (OLS) regression (Table 2) with the two transparent treatments as independent variables and the weight on advice as the dependent variable (Model 1). The results indicate that the observed effects remain significant when controlling for participants' age, gender, and level of education (Model 2; detailed effects of controls are reported in Table EC.3 in the online appendix). This provides further support for Hypotheses 1 and 2.

Since the use of algorithmic advice should improve the final forecast, we analyze the accuracy of the final forecast. As a measure, we use the mean absolute error (MAE), that is, the absolute deviation of the forecast from the mean demand, averaged over all products. We find that participants in the transparent-simple treatment performed worse than those in the other two treatments (Figure 4-b). Compared to the non-transparent treatment, the MAE of the final forecast is 21.4% higher in the transparent-simple treatment ($p < 0.001$). The participants' MAE in the transparent-complex treatment is not significantly different from the MAE in the non-transparent treatment ($p = 0.173$). In line with weight on advice, participants did not exploit the full potential of the algorithmic forecast, which has an MAE of 66.2 units.

In summary, we find support for Hypotheses 1 and 2. Increasing transparency reduced weight on advice for the simple algorithm and increased weight on advice for the complex algorithm. In our

Table 2 Effect of algorithm transparency in the two treatments on weight on advice

	Dependent Variable:	
	Weight on Advice	
	(1)	(2)
Transparent-Simple	-0.128^{***}	-0.138^{***}
	(0.028)	(0.027)
Transparent-Complex	0.063^{*}	0.055^{*}
	(0.028)	(0.027)
Constant	0.493^{***}	0.490^{***}
	(0.020)	(0.063)
Observations	509	509
Controls	N	Y
R^2	0.087	0.183
Adjusted R^2	0.084	0.156
Residual Std. Error	0.258 (df = 506)	0.248 (df = 492)
F Statistic	24.245^{***} (df = 2; 506)	6.876^{***} (df = 16; 492)
Note:		*p<0.05; **p<0.01; ***p<0.001

experiment, where both advice-giving algorithms were optimal and provided identical forecasts, increasing transparency reduced forecast accuracy for the simple algorithm without significantly affecting the forecast accuracy of the complex algorithm. This suggests that managers should be cautious in making simple algorithms transparent. We do not find an indication that they should be concerned about making complex algorithms transparent.

5.3. Perceived Appropriateness of Complexity as a Moderator of the Effects of Algorithm Transparency on Weight on Advice

The previous analyses focused on the effect of algorithm transparency on the mean weight on advice for two algorithms with different complexity levels. We will next analyze the extent to which the effect of algorithm transparency on weight on advice is moderated by the perceived appropriateness of algorithm complexity.

Figure 5 shows how individuals perceived the appropriateness of algorithm complexity in relation to their weight on advice in the transparent treatments. To analyze whether a deviation from the appropriate level of complexity in both directions has an influence on weight on advice, we run a piecewise linear regression model with unknown breakpoints (Muggeo 2003) (Figure 5).

The model estimates a breakpoint at the value of 4, the *appropriate* level of complexity (Davies' test $p < 0.001$; Davies (1987)). For values smaller than the breakpoint (*too simple*), we observe a significantly positive slope of the regression model ($\beta_1 = 0.1508$, $p < 0.001$). For levels greater than the breakpoint (*too complex*), the slope of the regression model is not significant ($\beta_2 = 0.0144$, $p = 0.395$). In other words, weight on advice significantly decreases, the more the transparent algorithm is perceived simpler than appropriate. Perceiving the algorithm as more complex than appropriate has no significant effect on weight on advice.

To further explore the relationship between perceived appropriateness of complexity and the weight on advice, we run an OLS regression with the non-transparent treatment as constant. In addition to the two transparent treatments as control variables, we introduce three independent variables that capture the degree to which the algorithm was perceived as *appropriate* (assigning a value of 1 if the reported value for the appropriateness question was 4 and 0 otherwise), *too simple* (translating a value of 1, 2, or 3 for the appropriateness question to 3, 2, or 1) or *too complex* (translating a value of 5, 6, or 7 for the appropriateness question to 1, 2, or 3). The results are presented

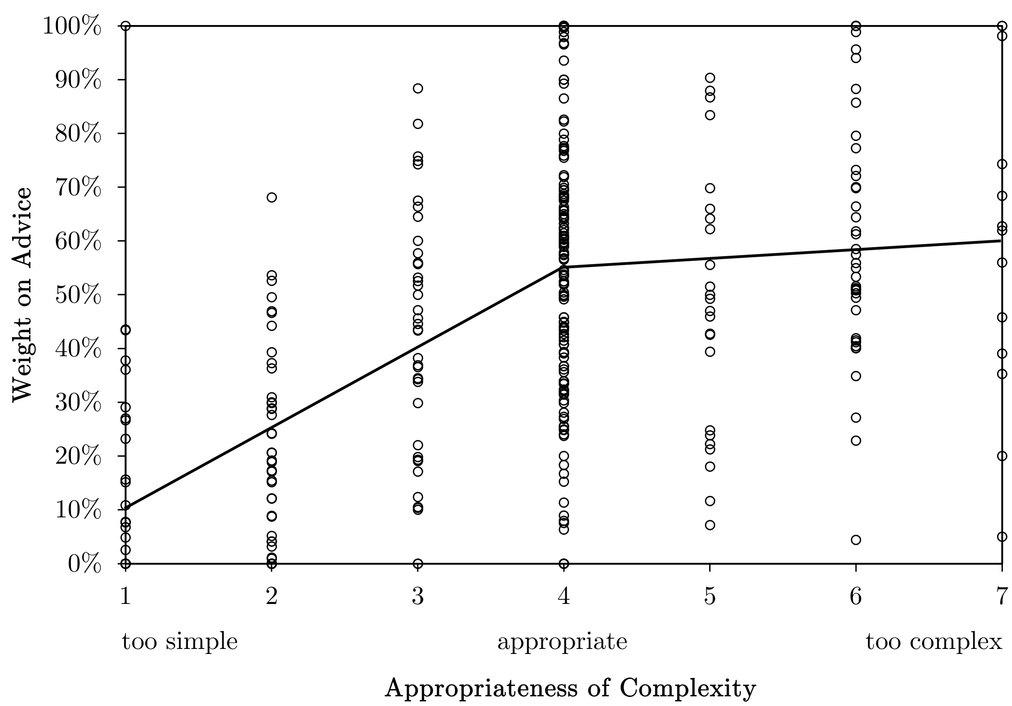


Figure 5 **Piecewise linear regression for the effect of the level of perceived appropriateness of algorithmic complexity on weight on advice.**

Table 3 Effect of transparency, appropriateness of algorithm complexity and treatments on weight on advice with the non-transparent treatment as the constant

	Dependent Variable:					
	Weight on Advice					
	(1)	(2)	(3)	(4)	(5)	(6)
Transparent & Perceived as too simple	−0.115***	−0.100***	−0.110***	−0.104***	−0.166**	−0.117*
	(0.025)	(0.025)	(0.029)	(0.029)	(0.056)	(0.056)
Transparent & Perceived as too complex	0.033	0.041	0.048	0.061	0.018	0.036
	(0.027)	(0.026)	(0.045)	(0.045)	(0.039)	(0.038)
Transparent & Perceived as appropriate	0.064	0.065	0.052	0.040	0.047	0.075
	(0.052)	(0.051)	(0.069)	(0.068)	(0.080)	(0.081)
Transparent-Simple	−0.027	−0.054	−0.031	−0.044		
	(0.054)	(0.053)	(0.064)	(0.064)		
Transparent-Complex	0.027	0.013			0.055	0.015
	(0.053)	(0.052)			(0.078)	(0.078)
Constant	0.493***	0.484***	0.493***	0.445***	0.493***	0.543***
	(0.018)	(0.059)	(0.018)	(0.069)	(0.018)	(0.076)
Observations	509	509	342	342	338	338
Controls	N	Y	N	Y	N	Y
R^2	0.233	0.298	0.228	0.294	0.092	0.186
Adjusted R^2	0.225	0.271	0.219	0.255	0.081	0.140

Note: Models (1)&(2) / (3)&(4) / (5)&(6) contain the responses of participants in all / the non-transparent & transparent-simple / the non-transparent & transparent-complex treatments, respectively. *p<0.05; **p<0.01; ***p<0.001.

in Table 3. Differences in weight on advice are no longer explained by the different algorithms but by the individuals' perception of the appropriateness of algorithm complexity (Model 1). We find that appropriateness of complexity is a moderator for the effect of transparency on the weight on advice when the algorithm is perceived as 'too simple' (significant negative effect). On the other hand, perceiving an algorithm as 'too complex' or 'appropriate' does not have a significant effect on the weight on advice. The results remain robust when we control for participants' gender, age, and level of education (Model 2) and for models containing only one of the two transparent treatments (Models 3 and 4 for the transparent-simple treatment; Models 5 and 6 for the transparent-complex treatment).

In summary, we find that participants who perceive a transparent algorithm as too simple place a lower weight on advice on the algorithm's recommendation than participants who perceive the algorithm as appropriate or too complex. Perceiving an algorithm as too complex has no significant effect on the weight on advice. This further strengthens the managerial implications of our results:

while the transparency of algorithms that might be perceived as too simple bears a risk, the transparency of algorithms that might be perceived as too complex does not seem to harm the use of algorithmic advice.

6. Conclusion

In this paper, we analyze how algorithm complexity influences the effects of algorithm transparency on the use of advice in managerial decision tasks. We conducted a preregistered laboratory experiment where decision makers received advice from algorithms with different levels of transparency and complexity. We find that, depending on the complexity of the underlying principles, the transparency of an algorithm can benefit or harm the use of advice. In our experiment, increasing the transparency of a simple algorithm reduces the use of advice and the accuracy of the final decision. The transparency of a complex algorithm increases the use of advice. We find that the use of advice is moderated by the individually perceived appropriateness of algorithm complexity. A decision maker who perceives a transparent algorithm as too simple discounts its advice, whereas perceiving a transparent algorithm as too complex does not significantly affect the use of advice.

Our research provides important implications for the interaction between humans and algorithms. Our results suggest that managers might not have to be concerned about revealing the principles of complex algorithms to decision makers, even if the decision makers do not fully comprehend them. However, revealing the underlying principles of a simple algorithm or an algorithm that might be perceived as being too simple can do more harm than good. Therefore, practitioners should carefully analyze and ponder the potential effects of making advice-giving algorithms more transparent. Particularly if the algorithms apply simple methods, efforts to increase algorithm transparency can backfire and harm the use of advice and the performance of the final decision.

The concept that users should understand the algorithms they are working with is prevalent in many literature streams. For example, evolving research on explainable artificial intelligence focuses on the question of how to make complex machine learning algorithms appear simple, easy to understand, and interpretable (Preece 2018). Our findings suggest that these efforts could also turn out to be a pitfall. Although the explanation improved the understanding of the algorithms in our experiment, this did not translate into an increased use of algorithmic advice.

Our research is a first step in understanding how algorithms should be designed for successful human-machine collaboration. We focused on the effects of algorithm transparency on the use of algorithmic advice for algorithms that provided perfect advice for the respective task. Future research could analyze settings in which this is not the case. For example, the decision maker and the algorithm could have complementary information. In such situations, additional challenges to the design of proper algorithms occur since mere compliance with an algorithm does not necessarily lead to successful human-machine collaboration. How could such collaboration be designed such that the decision maker uses the algorithmic advice and adapts it only when necessary and appropriate? How does transparency affect such cooperation? It would also be insightful to analyze how other dimensions, such as ethical standards and fairness, influence the use of algorithmic advice if the underlying principles of the algorithms are revealed.

Acknowledgments

Funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany´s Excellence Strategy – EXC 2126/1– 390838866.

References

Alvarado-Valencia J, Barrero LH, Önkal D, Dennerlein JT (2017) Expertise, credibility of system forecasts and integration methods in judgmental demand forecasting. *International Journal of Forecasting* 33(1):298–313.

Arvan M, Fahimnia B, Reisi M, Siemsen E (2019) Integrating human judgement into quantitative forecasting methods: A review. *Omega* 86:237–252.

Baron RM, Kenny DA (1986) The moderator–mediator variable distinction in social psychological research: Conceptual, strategic, and statistical considerations. *Journal of Personality and Social Psychology* 51(6):1173–1182.

Bertino E, Merrill S, Nesen A, Utz C (2019) Redefining data transparency: A multidimensional approach. *Computer* 52(1):16–26.

Bollen KA, Stine R (1990) Direct and indirect effects: Classical and bootstrap estimates of variability. *Sociological Methodology* 20:115–140.

Bonaccio S, Dalal RS (2006) Advice taking and decision-making: An integrative literature review, and implications for the organizational sciences. *Organizational Behavior and Human Decision Processes* 101(2):127–151.

Brynjolfsson E, McAfee A (2014) *The second machine age: Work, progress, and prosperity in a time of brilliant technologies* (New York: WW Norton & Company).

Burton JW, Stein MK, Jensen TB (2020) A systematic review of algorithm aversion in augmented decision making. *Journal of Behavioral Decision Making* 33(2):220–239.

Chen DL, Schonger M, Wickens C (2016) otree—an open-source platform for laboratory, online, and field experiments. *Journal of Behavioral and Experimental Finance* 9:88–97.

Cramer H, Evers V, Ramlal S, Van Someren M, Rutledge L, Stash N, Aroyo L, Wielinga B (2008) The effects of transparency on trust in and acceptance of a content-based art recommender. *User Modeling and User-Adapted Interaction* 18(5):455–496.

Dantzig GB (1957) Discrete-variable extremum problems. *Operations Research* 5(2):266–288.

Davies RB (1987) Hypothesis testing when a nuisance parameter is present only under the alternative. *Biometrika* 74(1):33–43.

De Baets S, Harvey N (2020) Using judgment to select and adjust forecasts from statistical models. *European Journal of Operational Research* 284(3):882–895.

Dhaliwal JS, Benbasat I (1996) The use and effects of knowledge-based system explanations: Theoretical foundations and a framework for empirical evaluation. *Information Systems Research* 7(3):342–362.

Dietvorst BJ, Simmons JP, Massey C (2015) Algorithm aversion: People erroneously avoid algorithms after seeing them err. *Journal of Experimental Psychology: General* 144(1):114–126.

Dietvorst BJ, Simmons JP, Massey C (2018) Overcoming algorithm aversion: People will use imperfect algorithms if they can (even slightly) modify them. *Management Science* 64(3):1155–1170.

Diubin G, Korbut A (2008) Average behavior of greedy algorithms for the minimization knapsack problem: General coefficient distributions. *Computational Mathematics and Mathematical Physics* 48(9):1521–1535.

Dzindolet MT, Peterson SA, Pomranky RA, Pierce LG, Beck HP (2003) The role of trust in automation reliance. *International Journal of Human-Computer Studies* 58(6):697–718.

Fildes R (2006) The forecasting journals and their contribution to forecasting research: Citation analysis and expert opinion. *International Journal of Forecasting* 22(3):415–432.

Fildes R, Goodwin P (2007) Against your better judgment? how organizations can improve their use of management judgment in forecasting. *Interfaces* 37(6):570–576.

Fildes R, Goodwin P, Lawrence M, Nikolopoulos K (2009) Effective forecasting and judgmental adjustments: an empirical evaluation and strategies for improvement in supply-chain planning. *International Journal of Forecasting* 25(1):3–23.

Fildes R, Goodwin P, Önkal D (2019) Use and misuse of information in supply chain forecasting of promotion effects. *International Journal of Forecasting* 35(1):144–156.

Fildes R, Petropoulos F (2015) Improving forecast quality in practice. *Foresight: The International Journal of Applied Forecasting* 36:5–12.

Fischer I, Harvey N (1999) Combining forecasts: What information do judges need to outperform the simple average? *International journal of Forecasting* 15(3):227–246.

Franses PH, Legerstee R (2011) Experts' adjustment to model-based sku-level forecasts: Does the forecast horizon matter? *Journal of the Operational Research Society* 62(3):537–543.

Gino F (2008) Do we listen to advice just because we paid for it? the impact of advice cost on its use. *Organizational Behavior and Human Decision Processes* 107(2):234–245.

Glikson E, Woolley AW (2020) Human trust in artificial intelligence: Review of empirical research. *Academy of Management Annals* 14(2):627–660.

Goodwin P, Fildes R (1999) Judgmental forecasts of time series affected by special events: Does providing a statistical forecast improve accuracy? *Journal of Behavioral Decision Making* 12(1):37–53.

Goodwin P, Gönül MS, Önkal D (2013) Antecedents and effects of trust in forecasting advice. *International Journal of Forecasting* 29(2):354–366.

Gupta A, Kannan K, Sanyal P (2018) Economic experiments in information systems. *MIS Quarterly* 42(2):595–606.

Harvey N, Bolger F (1996) Graphs versus tables: Effects of data presentation format on judgemental forecasting. *International Journal of Forecasting* 12(1):119–137.

Harvey N, Fischer I (1997) Taking advice: Accepting help, improving judgment, and sharing responsibility. *Organizational Behavior and Human Decision Processes* 70(2):117–133.

Holm S (1979) A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics* 6(2):65–70.

Kizilcec RF (2016) How much information? effects of transparency on trust in an algorithmic interface. *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, 2390–2395.

Klassen RD, Flores BE (2001) Forecasting practices of canadian firms: Survey results and comparisons. *International Journal of Production Economics* 70(2):163–174.

Kremer M, Moritz B, Siemsen E (2011) Demand forecasting behavior: System neglect and change detection. *Management Science* 57(10):1827–1843.

Lawrence M, Goodwin P, O'Connor M, Önkal D (2006) Judgmental forecasting: A review of progress over the last 25 years. *International Journal of Forecasting* 22(3):493–518.

LeCun Y, Bengio Y, Hinton G (2015) Deep learning. *Nature* 521(7553):436–444.

Logg JM, Minson JA, Moore DA (2019) Algorithm appreciation: People prefer algorithmic to human judgment. *Organizational Behavior and Human Decision Processes* 151:90–103.

Madsen M, Gregor S (2000) Measuring human-computer trust. *11th australasian conference on information systems*, volume 53, 6–8 (Citeseer).

Malhotra MK, Singhal C, Shang G, Ployhart RE (2014) A critical evaluation of alternative methods and paradigms for conducting mediation analysis in operations management research. *Journal of Operations Management* 32(4):127–137.

Moritz B, Siemsen E, Kremer M (2014) Judgmental forecasting: Cognitive reflection and decision speed. *Production and Operations Management* 23(7):1146–1160.

Muggeo VM (2003) Estimating regression models with unknown break-points. *Statistics in Medicine* 22(19):3055–3071.

Parasuraman R, Manzey DH (2010) Complacency and bias in human use of automation: An attentional integration. *Human Factors* 52(3):381–410.

Perera HN, Hurley J, Fahimnia B, Reisi M (2019) The human factor in supply chain forecasting: A systematic review. *European Journal of Operational Research* 274(2):574–600.

Preacher KJ, Hayes AF (2008) Asymptotic and resampling strategies for assessing and comparing indirect effects in multiple mediator models. *Behavior Research Methods* 40(3):879–891.

Preece A (2018) Asking 'why' in ai: Explainability of intelligent systems–perspectives and challenges. *Intelligent Systems in Accounting, Finance and Management* 25(2):63–72.

Sanders NR, Manrodt KB (2003) The efficacy of using judgmental versus quantitative forecasting methods in practice. *Omega* 31(6):511–522.

Sinha R, Swearingen K (2002) The role of transparency in recommender systems. *CHI'02 extended abstracts on human factors in computing systems*, 830–831.

Sniezek JA, Schrah GE, Dalal RS (2004) Improving judgement with prepaid expert advice. *Journal of Behavioral Decision Making* 17(3):173–190.

Springer A, Whittaker S (2018) What are you hiding? algorithmic transparency and user perceptions. *The 2018 AAAI Spring Symposium Series*, 455–459.

Springer A, Whittaker S (2019) Making transparency clear. *Joint Proceedings of the ACM IUI 2019 Workshops*.

Wang W, Benbasat I (2007) Recommendation agents for electronic commerce: Effects of explanation facilities on trusting beliefs. *Journal of Management Information Systems* 23(4):217–246.

Webby R, O'Connor M (1996) Judgemental and statistical time series forecasting: A review of the literature. *International Journal of Forecasting* 12(1):91–118.

Yaniv I (2004a) The benefit of additional opinions. *Current Directions in Psychological Science* 13(2):75–78.

Yaniv I (2004b) Receiving other people's advice: Influence and benefit. *Organizational Behavior and Human Decision Processes* 93(1):1–13.

Yu K, Berkovsky S, Taib R, Zhou J, Chen F (2019) Do i trust my machine teammate? an investigation from perception to decision. *IUI'19: ProceedinSgs of the 24th International Conference on Intelligent User Interfaces*, 460–468.

Online Appendix

EC.1. Preliminary Study

To obtain an initial indication of the effects of algorithm transparency on the use of algorithmic advice, we conducted a preliminary study in the domain of inventory management with undergraduate students in an operations management class. A participant was either assigned to the *non-transparent* treatment or the *transparent* treatment. Both treatments received the same algorithmic advice that solved the problem optimally, but only participants in the transparent treatment were informed about the underlying principles of the algorithm.

EC.1.1. Experimental Design

The participants had to select spare parts for bicycle repairs. A repair required exactly 1 out of 15 spare parts. Each spare part had a certain weight and a probability of being needed for a repair job. The selected parts were not allowed to exceed a given weight limit. This combinatorial optimization task is called the 'knapsack-problem'. A popular and simple solution technique for this problem is a greedy algorithm, which sorts the spare parts by their value-to-weight ratio and iteratively selects the part with the next-highest ratio as long as the capacity is not exceeded. Although this greedy algorithm does not necessarily solve the problem optimally, it usually produces good results. For our problem sets, the solutions of the greedy algorithm were optimal.

After the participants had read the task description and answered comprehension questions, they were assigned to either the non-transparent or transparent treatment. All participants were informed that an algorithm had worked on the task as well and provides advice. The participants in the transparent treatment received a detailed briefing on the underlying principles of the greedy algorithm. Then, participants started the spare part selection. They saw a table containing 15 different spare parts in a random order with the corresponding probability of being needed for a repair job, the part weights, and the selection advice of the algorithm. After the participants had finalized their spare part selection, they were informed about the spare part that was required for the maintenance job. Participants made spare part decisions for 15 rounds with different spare part sets and weight limits.

For each round, we measured whether the advice of the algorithm was adopted completely (advice adopted = 1) or if the participant deviated from the advice (advice adopted = 0). We assessed the level of trust in the algorithm through a questionnaire at the end of the experiment. We designed this questionnaire based on Madsen and Gregor (2000), who identified the five major constructs of human computer trust as *perceived understandability*, *perceived technical competence*, *perceived reliability*, *personal attachment*, and *faith*.

EC.1.2. Experimental Protocol

The experiment was set up with oTree (Chen et al. 2016) and conducted as part of an undergraduate operations management class on October 30, 2019. All 562 students who were present in the class started the experiment. Fifteen students did not pass the comprehension questions, and 6 students did not complete the spare part selection or the questionnaire at the end. This left us with 541 participants, with 303 being assigned to the non-transparent treatment and 238 to the transparent treatment. A total of 265 participants were female, 271 were male, and 5 did not report their gender. The students were incentivized by performance-based bonus credit points for the class. On average, the completion of the study took 23 minutes and participants earned 17.44 out of a maximum of 20 bonus points.

EC.1.3. Results

Our results indicate that transparency has a negative effect on the adoption of advice. As presented in Table EC.1, the general willingness to use the advice of a black-box algorithm was high. A thorough explanation of the greedy algorithm significantly decreased advice adoption by 7.6%.

Figure EC.1 shows the reported levels of trust based on the five trust constructs derived by Madsen and Gregor (2000). While all participants reported having relatively high trust in the algorithm, there is an indication that the trust levels in the transparent treatment are lower. In particular, the perceived technical competence and the perceived reliability of the algorithm are significantly lower when the participants received an explanation of the algorithm (two-sided Mann-Whitney U tests, $p < 0.001$).

Table EC.1 Effect of transparency on the adoption of the algorithmic advice

	Dependent variable:
	Adoption of Advice
Transparent	-0.076^{**}
	(0.030)
Constant	0.769^{***}
	(0.022)
Observations	541
R^2	0.012
Adjusted R^2	0.010
Residual Std. Error	0.343 (df = 539)
F Statistic	6.476^{**} (df = 1; 539)
Notes: OLS regression.	Standard errors in parentheses.
	*p<0.1; **p<0.05; ***p<0.01

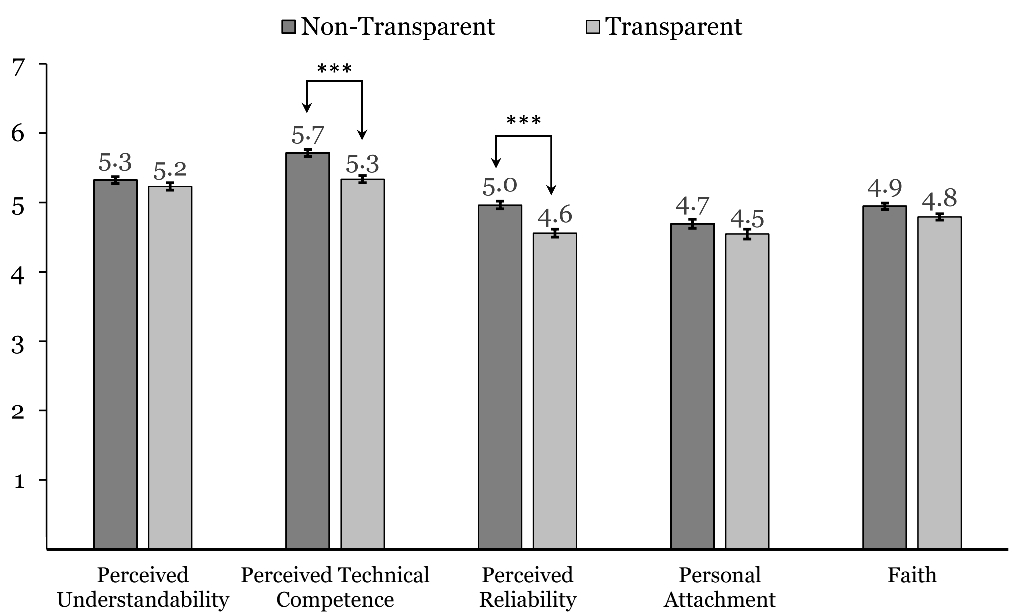


Figure EC.1 Reported levels of trust on a 7-point Likert-type scale for the five trust constructs of Madsen and Gregor (2000). Error bars indicate standard errors.

EC.2. Preregistered Hypotheses

We preregistered the laboratory experiment at the Open Science Framework ([*link blinded for review*]). In addition to the study design, we also preregistered hypotheses and the corresponding data analyses. In the course of writing this paper, we slightly adapted the formulation of our research question and the hypotheses after preregistration. Nevertheless, all preregistered hypotheses hold, and we present them and the corresponding results in the following.

HYPOTHESIS EC.1.

(a) Increasing the transparency of a simple algorithm through explanation leads to a lower perceived value of its advice compared to not explaining it.

(b) This effect declines when the algorithm complexity increases.

We measured the participants' perceived value of advice in the three treatments by the participants' agreement with the statement "I think the recommendations of the algorithm will be valuable" on a Likert-type scale from 1 (strongly disagree) to 7 (strongly agree). Figure EC.2 summarizes the results.

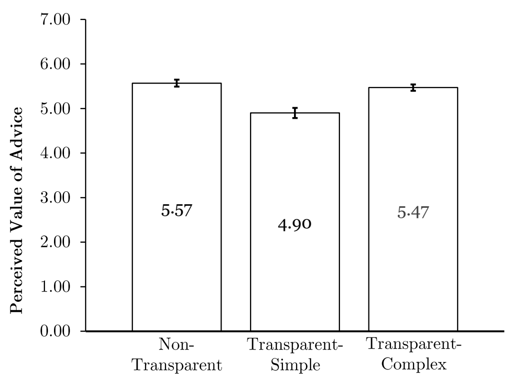


Figure EC.2 Perceived value of advice in the three treatments. Error bars indicate standard errors.

Participants in the transparent-simple treatment reported a significantly lower perceived value of advice after the explanation of the underlying principles than those in the non-transparent treatment (two-sided Mann-Whitney U test, $p < 0.001$), which provides support for Hypothesis EC.1-a. This negative effect is reduced if the complex algorithm is explained (two-sided Mann-Whitney U test, $p < 0.001$), which provides support for Hypothesis EC.1-b.

Hypothesis EC.2. *A higher perceived value of advice increases the utilization of advice.*

We run an ordinary least squares (OLS) regression to test the effect of the perceived value of advice on the utilization of advice. We use the indicated agreement with the statement "I think the recommendations of the algorithm will be valuable" as the independent variable and weight on advice as the dependent variable. The results demonstrate that the perceived value of advice has a positive effect on the utilization of advice (Table EC.2, Model 1). The results remain when controlling for gender, age, and education (Model 2). This provides support for our second hypothesis.

Table EC.2 Effect of perceived value of advice on weight on advice.

	Dependent variable:	
	Weight on Advice	
	(1)	(2)
Perceived Value of Advice	0.084^{***}	0.079^{***}
	(0.009)	(0.009)
Constant	0.026	0.035
	(0.050)	(0.077)
Observations	509	509
Controls	N	Y
R^2	0.143	0.217
Adjusted R^2	0.141	0.193
Residual Std. Error	0.250 (df = 507)	0.242 (df = 493)
F Statistic	84.568^{***} (df = 1; 507)	9.082^{***} (df = 15; 493)
Note:		$^{*}p<0.05$; $^{**}p<0.01$; $^{***}p<0.001$

Hypothesis EC.3.

(a) Increasing transparency of a simple algorithm through explanation leads to lower utilization of its advice, mediated by the perceived value of advice.

(b) This effect declines when the algorithm complexity increases.

The hypothesized effect is a consistent mediation with a single mediator. We use the Baron and Kenny procedure (Baron and Kenny 1986), an approach that is often used to test for simple and consistent mediations (e.g., Malhotra et al. (2014)).

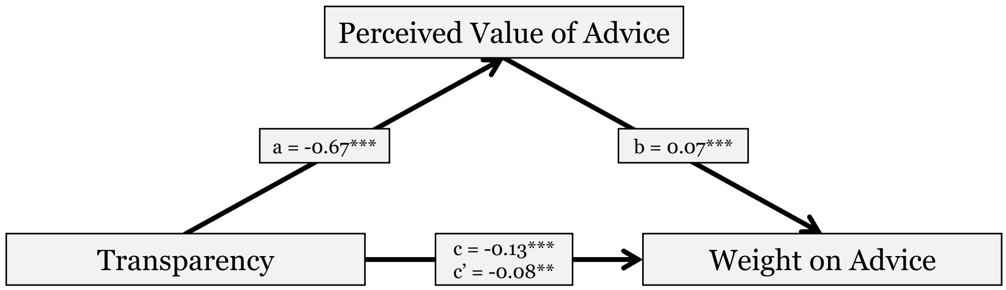


Figure EC.3 **Mediation model with effect of transparency on perceived value of advice (a), effect of perceived value of advice on weight on advice (b), and the total effect (c) and direct effect (c') of transparency on weight on advice. Note. *p<0.05; **p<0.01; ***p<0.001 (two-tailed)**

As depicted in Figure EC.3, the mediation analysis reveals a significant total effect c of the transparency of the simple algorithm on the use of advice. Transparency is negatively correlated with the perceived value of advice (effect a), which provides support for Hypothesis EC.1. To estimate weight on advice, we use both transparency and the perceived value of advice as independent variables. We find that the perceived value of advice has a significant positive effect b (Hypothesis EC.2), while transparency has a significant negative effect c'. Since the direct effect c' is smaller than the total effect c, the weight on advice is partially mediated by the perceived value of advice. Bootstrapping (Bollen and Stine 1990, Preacher and Hayes 2008) with 5,000 iterations provides a 95% confidence interval on the indirect effect $ab = -0.05$ (transparency mediated by perceived value of advice) of $CI = \pm 0.02$. This provides support for Hypothesis EC.3-a.

Analyzing the mediation analysis for the effect of transparency of the complex algorithm provides support for Hypothesis EC.3-b. Transparency has no significant effect on perceived value ($a = -0.05$, $p = 0.35$), while the effect of perceived value on weight on advice is still significant ($b = 0.05$, $p < 0.001$). Since the total effect $c = 0.03$ equals the direct effect c', we cannot conclude that the weight on advice is mediated by the perceived value of advice for the complex algorithm.

EC.3. Laboratory Experiment

EC.3.1. Task Description

Task Description

Your Task

Your task is to forecast the next period's demand for 10 different products. For each product, the demand history of the 10 previous periods is presented. All products are independent from each other, that is, the demand of one product does not influence the demand of any other product.

You will be informed about the actual demand after you have forecasted all 10 products.

Reward

In addition to the fixed reward of $0.50, you receive a bonus reward based on the quality of your forecast. The closer your forecast is to the actual demand, the higher is your bonus reward.

The bonus reward is calculated based on the forecast error, that is, the absolute difference between your forecast and the actual demand divided by the actual demand:

$$\text{forecast error} = \frac{|\text{your forecast} - \text{actual demand}|}{\text{actual demand}}$$

For each product, you earn $\mathbf{\$0.10 * (1{-}forecast\ error)}$. This bonus reward cannot be negative.

Elements of this HIT

This HIT consists of comprehension questions about your task, which you have to answer correctly in order to continue. Subsequently, you have to forecast 10 different products, and complete a short questionnaire. Afterwards, you will receive information about the actual demand and your total reward, which we will pay within 24 hours.

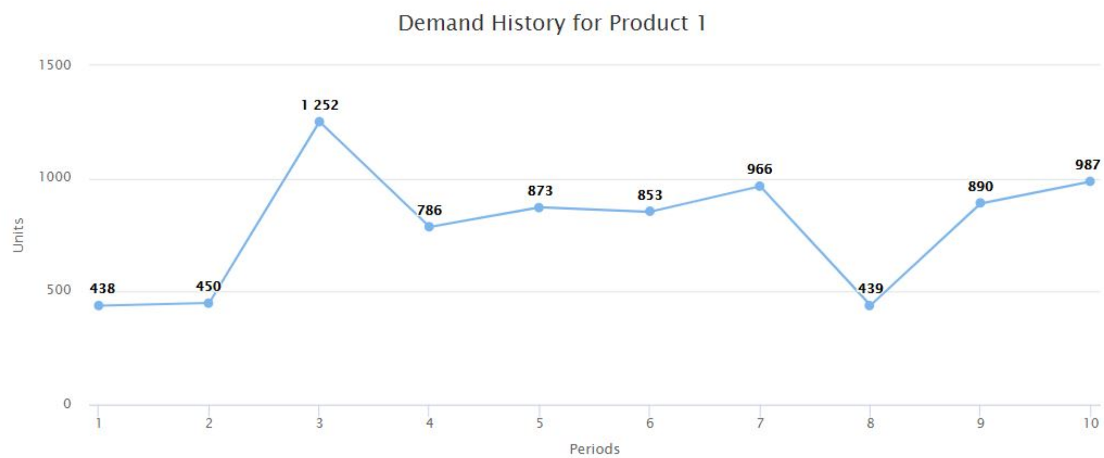
If you have understood your task and are ready to answer the comprehension questions, please click "Next".
Please note that you cannot return to this page, and need to answer the comprehension questions correctly in order to complete this HIT.

Next

EC.3.2. Initial Forecast

Forecasting | Product 1

This is the demand history of the 10 previous periods for Product 1:



Please forecast the demand for Period 11:

Please press "Next" to continue.

Next

EC.3.3. Algorithm Explanation

EC.3.3.1. Non-Transparent Treatment

Algorithm Support

An **algorithm has determined a demand forecast** for the 10 products.

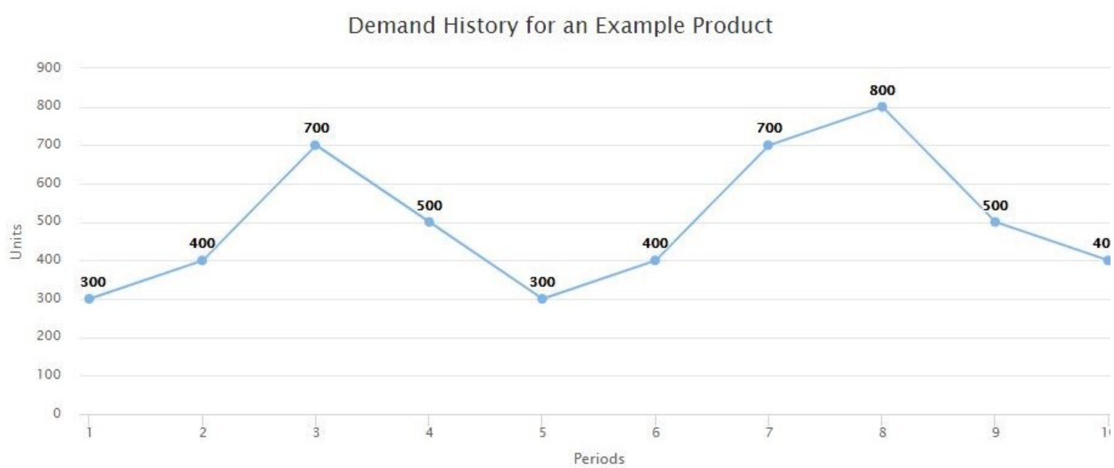
In the following, **you will see the 10 products again**. You will see your **initial forecast** and the **forecast of the algorithm**. If you wish, **you can update your final forecast** for the product.

Your bonus reward will only depend on the final forecast.

EC.3.3.2. Transparent-Simple Treatment

Algorithm Support

An **algorithm has determined a demand forecast** for the 10 products. It calculates the forecast for a product by computing the **average of the demand history of the last 10 periods**.



The algorithm recommends a demand forecast of 500 units for Period 11.

This is computed by:

$$\frac{300+400+700+500+300+400+700+800+500+400}{10} = 500$$

In the following, **you will see the 10 products again**. You will see your **initial forecast** and the **forecast of the algorithm**. If you wish, **you can update your final forecast** for the product.

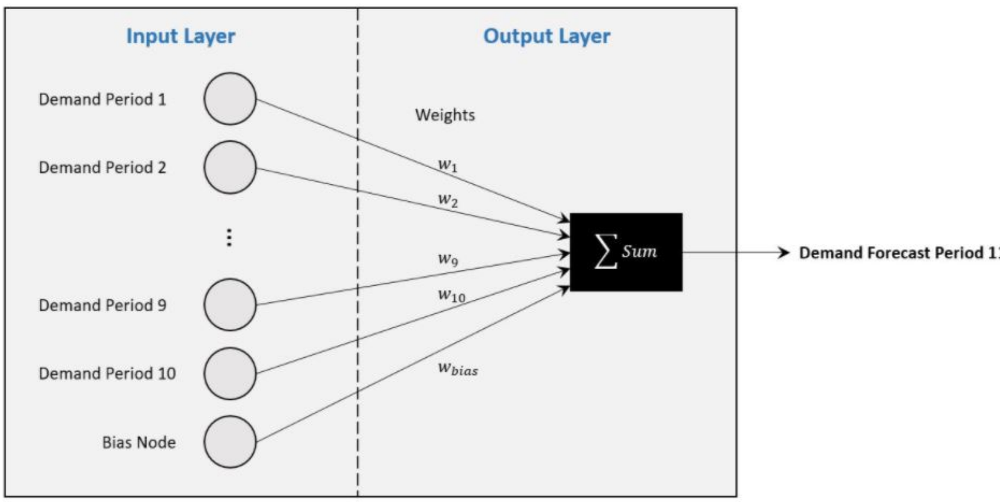
Your bonus reward will only depend on the final forecast.

EC.3.3.3. Transparent-Complex Treatment

Algorithm Support

An **algorithm has determined a demand forecast** for the 10 products. The algorithm applies **artificial intelligence**: it is a **neural network** that imitates the human brain. It uses elements called neurons to process information, analyze complex problems, and make decisions.

Our neural network consists of an input layer and an output layer:



The **input layer** is the **demand history**. Each demand in the demand history is one input node. Moreover, the algorithm contains a bias node. The result of the **output layer** is the **demand forecast for period 11**.

The algorithm calculates the demand forecast for period 11 for a product by executing the following steps:

1. It normalizes the demands of the demand history to values between 0 and 1:

$$\text{demand period } t_{norm} = \frac{\text{demand period } t - \text{demand}_{min}}{\text{demand}_{max} - \text{demand}_{min}}$$

   demand_{min} and demand_{max} are the minimum and maximum demand values observed so far.

2. It multiplies the normalized demands and the bias unit with the weights $w_1, \ldots, w_{10}, w_{bias}$:

$$\text{weighted demand period } t_{norm} = \text{demand period } t_{norm} \cdot w_t$$

3. It sums up all weighted demands and the weighted bias node:

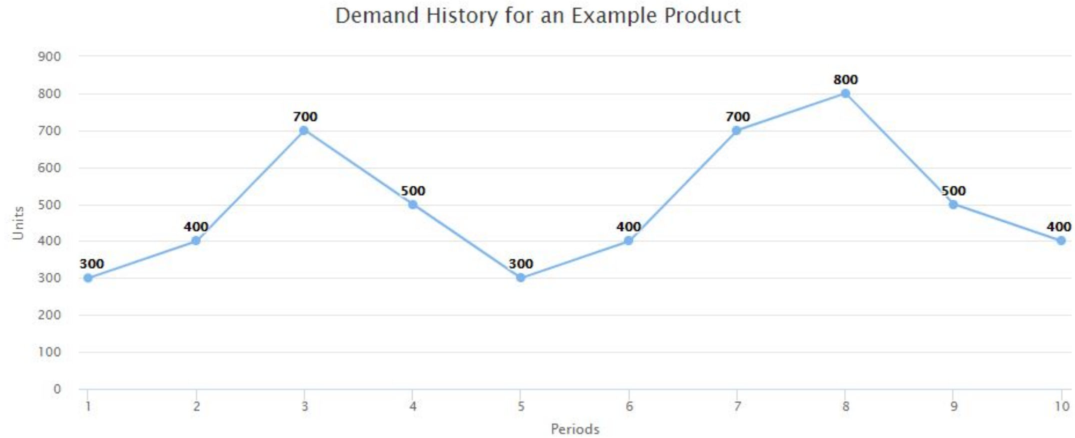
$$\text{demand forecast period } 11_{norm} = \text{weighted demand period } 1_{norm} + \ldots + \text{weighted demand period } 10_{norm} + \text{weighted bias node}_{norm}$$

4. It transforms the normalized values back to real values:

$$\text{demand forecast period } 11 = \text{demand forecast period } 11_{norm} \cdot (\text{demand}_{max} - \text{demand}_{min}) + \text{demand}_{min}$$

To compute a forecast for period 11, the weights $w_1, \ldots, w_{10}, w_{bias}$ are required. The algorithm has been trained with **100,000 structurally similar demand histories** to determine appropriate weights.

Example



The following table displays the results of steps 1 to 4 of the algorithm: the demands, normed demands, weights and weighted normed demands for the demand history (periods 1 to 10). Finally, the normed demand forecast for period 11 is transferred back to real values.
Having been trained with **100,000 structurally similar demand histories**, the neural network suggests a demand forecast of **500 units** for Period 11.

Period t	Demand	Normed Demand	Weight	Weighted Normed Demand
1	300	0.222	0.191	0.043
2	400	0.25	0.181	0.045
3	700	0.443	0.179	0.079
4	500	0.327	0.187	0.061
5	300	0.244	0.198	0.048
6	400	0.258	0.177	0.046
7	700	0.402	0.183	0.073
8	800	0.412	0.187	0.077
9	500	0.302	0.184	0.056
10	400	0.276	0.192	0.053
Bias		1.000	-0.236	-0.236
	Demand forecast for period 11 (normed)			0.344
	Demand forecast for period 11			**500**

In the following, **you will see the 10 products again**. You will see your **initial forecast** and the **forecast of the algorithm**. If you wish, **you can update your final forecast** for the product.

Your bonus reward will only depend on the final forecast.

EC.4. Regressions with Controls

Table EC.3 Treatment effect of weight on advice with all control variables

	Dependent variable:
	Weight on Advice
Transparent-Simple	-0.138^{***}
	(0.027)
Transparent-Complex	0.055^{*}
	(0.027)
Age: 31 - 40	-0.001
	(0.027)
Age: 41 - 50	-0.077^{*}
	(0.033)
Age: 51 - 60	-0.070
	(0.040)
Age: 61 - 70	-0.114^{*}
	(0.057)
Age: 71 - 80	0.125
	(0.113)
Gender: Male	-0.091^{***}
	(0.022)
Gender: Other	-0.251^{*}
	(0.115)
Education: Bachelor's degree	0.101
	(0.061)
Education: Doctoral degree (PhD)	0.033
	(0.119)
Education: High school graduate	0.208^{**}
	(0.072)
Education: I don't want to report.	0.129
	(0.155)
Education: Less than high school degree	0.012
	(0.256)
Education: Master's degree	0.018
	(0.065)
Education: Some college but no degree	0.041
	(0.063)
Constant	0.490^{***}
	(0.063)
Observations	509
R^2	0.183
Adjusted R^2	0.156
Residual Std. Error	0.248 (df = 492)
F Statistic	6.876^{***} (df = 16; 492)
Note:	$^{*}p<0.05$; $^{**}p<0.01$; $^{***}p<0.001$

Table EC.4 Effect of transparency, appropriateness of algorithm complexity and treatments on weight on advice with the non-transparent treatment as the constant and all control variables

	Dependent variable:		
	Weight on Advice		
	(1)	(2)	(3)
Transparent-Simple	−0.054	−0.044	
	(0.053)	(0.064)	
Transparent-Complex	0.013		0.015
	(0.052)		(0.078)
Transparent & Perceived as too simple	−0.100***	−0.104***	−0.117*
	(0.025)	(0.029)	(0.056)
Transparent & Perceived as too complex	0.041	0.061	0.036
	(0.026)	(0.045)	(0.038)
Transparent & Perceived as appropriate	0.065	0.040	0.075
	(0.051)	(0.068)	(0.081)
Gender: Male	−0.062**	−0.044	−0.091***
	(0.021)	(0.025)	(0.026)
Gender: Other	−0.239*	−0.251	−0.273*
	(0.107)	(0.133)	(0.109)
Age: 31 - 40	−0.003	0.016	−0.023
	(0.025)	(0.029)	(0.031)
Age: 41 - 50	−0.064*	−0.045	−0.067
	(0.031)	(0.036)	(0.039)
Age: 51 - 60	−0.068	−0.070	−0.092*
	(0.037)	(0.047)	(0.044)
Age: 61 - 70	−0.093	−0.105	−0.132
	(0.053)	(0.066)	(0.068)
Age: 71 - 80	0.074	0.209	0.008
	(0.105)	(0.163)	(0.139)
Education: Bachelor's degree	0.088	0.109	0.066
	(0.056)	(0.068)	(0.073)
Education: Doctoral degree (PhD)	−0.019	−0.046	−0.019
	(0.111)	(0.133)	(0.138)
Education: High school graduate	0.190**	0.211**	0.090
	(0.067)	(0.081)	(0.087)
Education: I don't want to report.	0.114	−0.052	0.227
	(0.144)	(0.174)	(0.179)
Education: Less than high school degree	−0.008	−0.006	−0.018
	(0.238)	(0.236)	(0.243)
Education: Master's degree	0.029	0.059	−0.024
	(0.061)	(0.072)	(0.078)
Education: Some college but no degree	0.023	0.052	−0.013
	(0.058)	(0.070)	(0.075)
Constant	0.484***	0.445***	0.543***
	(0.059)	(0.069)	(0.076)
Observations	509	342	338
R^2	0.298	0.294	0.186
Adjusted R^2	0.271	0.255	0.140
Residual Std. Error	0.230 (df = 489)	0.225 (df = 323)	0.231 (df = 319)
F Statistic	10.926*** (df = 19; 489)	7.471*** (df = 18; 323)	4.048*** (df = 18; 319)

Note: Models (1) / (2) / (3) contain the responses of participants in all / the non-transparent & transparent-simple / the non-transparent & transparent-complex treatments, respectively. *p<0.05; **p<0.01; ***p<0.001.