Why providing humans with interpretable algorithms may, counterintuitively, lead to lower decision-making performance

Timothy DeStefano[a], Katherine C. Kellogg[b], Michael Menietti[c], and Luca Vendraminelli[d]

[a]Georgetown
[b]MIT Sloan
[C]Harvard Business School - Laboratory for Innovation Science at Harvard (LISH)
[d]Politecnico di Milano

Abstract

How is algorithmic model interpretability related to human acceptance of algorithmic recommendations and performance on decision-making tasks? We explored these questions in a multi-method field study of a large multinational fashion organization. We first conducted a quantitative field experiment to compare the use of two models—an interpretable versus an uninterpretable algorithmic model— designed to assist employees with decision making around how many products to send to each of its stores. Contrary to what the literature on interpretable algorithms would lead us to expect, under conditions of high perceived uncertainty, decision makers' use of an uninterpretable algorithmic model was associated with higher acceptance of algorithmic recommendations and higher task performance than was their use of an interpretable algorithmic model with a similar level of performance. We next investigated this puzzling result using 31 interviews with 14 employees—2 algorithm developers, 2 managers, and 10 decision makers. We advance two concepts that suggest a refinement of theory on interpretable algorithms. First, *overconfident troubleshooting*—a decision maker rejecting a recommendation coming from an interpretable algorithm, because of their belief that they understand the inner workings of complex processes better than they actually do. Second, *social proofing the algorithm*—including respected peers in the algorithm development and testing process—may make it more likely that decision makers accept recommendations coming from an uninterpretable algorithm in situations characterized by high perceived uncertainty, because the decision makers may seek to reduce their uncertainty by incorporating the opinions of people with their own knowledge base and experience.

Keywords: Interpretable AI; Artificial intelligence; Machine learning, Algorithm aversion; AI Adoption, Firm productivity; AI and strategy; Human-in-the-loop decision making

Acknowledgements: This research received important input from Marco Iansiti and Karim Lakhani at the Harvard Business School and from Fabio Luzzi, Josh Ainsley, Chicheng Zhang and Jeremy King at Tapestry, Inc.

1. Introduction

With the continuing application of artificial intelligence (AI) technologies, algorithmic decision-making is becoming more efficient, often even outperforming humans. Despite this superior performance, human decision makers often consciously or unconsciously display reluctance to accept algorithmic recommendations, a phenomenon known as algorithm aversion. Algorithm aversion is particularly prevalent in decision making situations characterized by uncertainty, such as medical (Dietvorst & Bharti, 2020; Kawaguchi, 2021) and financial investment decision making contexts (Zhang et al., 2021). Yet, this tendency for decision makers to reject algorithmic recommendations in uncertain situations is particularly problematic, because such areas may be the ones that are most amenable to improved outcomes through the use of human-in-the loop decision making (Verganti et al., 2020).

One potential mechanism for increasing human decision maker acceptance of algorithmic decisions is that of interpretable AI, or designing algorithmic models that are inherently interpretable to humans (Rudin, 2019). The literature on interpretable AI puts forth two arguments that are relevant to our study. First, providing human decision makers with an "interpretable model that obeys a domain-specific set of constraints to allow it (or its predictions, or the data) to be more easily understood by humans" (Rudin et al., 2022, p. 3) should result in greater acceptance of recommendations from the model than providing human decision makers with an equally high performing uninterpretable model (Ashoori & Weisz, 2019; Brundage et al., 2020). Second, providing human decision makers with an interpretable algorithmic model with a similar level of model performance should allow for better human decision-making performance than providing human decision makers with an uninterpretable algorithmic model (Arrieta et al., 2019; Rudin & Radin, 2019).

While this research has been important in illuminating important issues related to algorithmic interpretability, human acceptance of algorithmic recommendations, and human task performance on algorithmically informed decision-making tasks, it cannot explain the results we found in our multi-method study in a large multinational fashion organization. We first performed a quantitative field experiment to test the effect of human decision makers' use of an interpretable (weighted moving average with clear inputs) versus an uninterpretable algorithmic model (recurrent neural network – Machine Learning), under conditions of high perceived uncertainty, on the dual outcomes of 1) human decision maker acceptance of algorithmic recommendations and 2) human decision maker task performance. We randomized algorithmic assistance throughout the company's allocation decision-making processes where half the employee decisions around how many products to send to each of its stores were assisted with recommendations from the interpretable algorithm and half were assisted with the uninterpretable algorithm.

Contrary to what the literature on interpretable algorithms would predict, we found that, under conditions of high perceived uncertainty, human decision makers' use of the *uninterpretable* algorithmic model was associated with greater acceptance of algorithmic recommendations and greater performance—fewer stockouts and higher sales measured by quantity and value— than was human decision makers' use of the interpretable model in situations with a similar level of model performance.

We next conducted 31 interviews with 14 employees to understand this puzzling result. The key theme that emerges from our qualitative analysis is that, under conditions of high perceived uncertainty, algorithmic models that are interpretable to humans may, counterintuitively, lead to lower acceptance of algorithmic recommendations. This may occur because allowing human decision makers to interrogate an algorithmic recommendation may lead

them to do what we call *overconfident troubleshooting*—a decision maker rejecting a recommendation coming from an interpretable algorithm, because of their belief that they understand the inner workings of complex processes better than they actually do. Because of the "illusion of explanatory depth" (Rozenblit & Keil, 2002)—humans' belief that we understand the causes, effects, and inner workings of complex mechanisms, events, and processes much better than we actually do—providing humans with an interpretable algorithm may make it more likely that they reject the recommendation coming from it.

Further, *social proofing the algorithm*, including respected peers in the algorithm development and testing process, may make it more likely that decision makers accept recommendations coming from an uninterpretable algorithm in situations characterized by high perceived uncertainty, because the decision makers may seek to reduce their uncertainty by incorporating the opinions of these peers. The mismatch between a human decision maker's initial judgment and algorithmic recommendation may lead the decision maker to become more uncertain in their judgment and want to reduce their uncertainty. When a human decision maker is not able to interrogate the reasoning behind the algorithmic recommendation, in a decision-making situation characterized by high perceived uncertainty, they may seek to reduce their uncertainty by incorporating the opinions of people like them—people with their knowledge base and experience— who have been involved in the algorithm development and testing process.

The rest of the paper continues as follows. In the next section, we describe the relevant literature and theoretical background. Section 3 describes the experimental setting and the organization's processes involved in product allocation. The quantitative methods and quantitative analysis are discussed in Section 4 and 5. Section 6 describes the qualitative methodology adopted. Section 7 reports the findings from the qualitative interviews, and explains our concepts of

overconfident troubleshooting and *social proofing the algorithm*. The theoretical lessons learned from this study are discussed in Section 8 followed by a brief conclusion in Section 9.

2. Background Literature

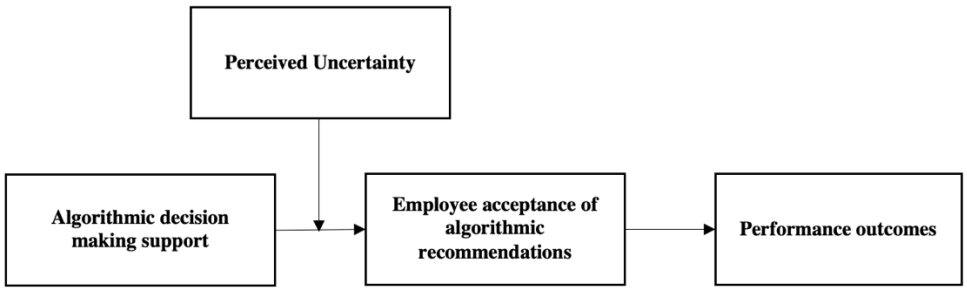

Figure 1: Theoretical Model

Our theoretical model (Figure 1) frames the effect of algorithmic model's decision making support on firm performance through the mediation of employees' acceptance of algorithmic recommendations, which represents the human-in-the-loop model of AI adoption (Kleinberg et al., 2018). The higher the algorithm aversion, the less the employee's acceptance of the algorithmic support. Finally, the framework involves the moderation of perceived uncertainty (Dietvorst & Bharti, 2020), which influences the employee acceptance of the algorithmic recommendation.

Differences in algorithmic support: Accuracy and interpretability

Algorithmic support refers to the information provided by an artifact capable of cognitive tasks. Herm et al (2020) classify algorithms with respect to accuracy and interpretability. First, algorithms on average differ in their capability to predict the likelihood of a future event to happen and last (Bonde Thylstrup et al., 2019; Henriksen & Bechmann, 2020). For example, ML models perform better than rule-based models in high data-frequency contexts with high variability and turbulent conditions, or in changing contextual conditions that require learning and adaptation (Herm et al., 2022). Second, algorithms differ in their interpretability. Compared to ruled-based

algorithms, which are based on hierarchies of rules and control flows (Lebovitz et al., 2022), ML algorithms are less intelligible to users, who have difficulties understanding the quality of the information received. The reduction of algorithmic interpretability correlates to the increasing mathematical knowledge required to understand the model and the information processing required for humans to replicate algorithmic paths and rules (Burrell, 2016).

The mechanics of employee acceptance: Algorithm aversion under conditions of perceived uncertainty

With the progress in AI technologies, the accuracy of algorithms is increasing, often even outperforming humans. The advancement of algorithmic support in decision-making loops aims to increase human decision-making performance by providing better data. Although the study of ML-based decision-making support is in an early stage, ML-based decision-making support has already shown the importance of considering the human-in-the-loop behavior when looking at the ML effect on firm performances. For example, in court decisions, Kleinberg et al (2018) show an average performance increase in bail decisions when decision making is informed by ML-based decision-making support. In discussing the findings, they however supported the hypothesis that "*good predictors do not necessarily improve decisions*.""

Indeed, despite this superior performance, human decision makers are sometimes reluctant to accept algorithmic recommendations, displaying algorithm aversion. Algorithm aversion is particularly prevalent in decision making domains characterized by perceived uncertainty (Dietvorst & Bharti, 2020; Feng & Gao, 2020; Kawaguchi, 2021; Sutherland et al., 2016; Zhang et al., 2021). For instance, in highly uncertain environments such as medical decision-making (Dietvorst & Bharti, 2020; Kawaguchi, 2021) and financial investment decision-making (Zhang et al., 2021), human decision makers frequently reject algorithmic recommendations. This may be

due to people feeling insecure regarding the outcome, and concerned about the consequences of this outcome (Grgić-Hlača et al., 2019; Lennartz et al., 2021).

Building interpretable algorithms

Algorithm aversion can be problematic, especially considering the increase in the accuracy of algorithms. Thus, scholars have explored potential mechanisms for increasing human acceptance of algorithmic decisions in settings of high uncertainty, in hopes of increasing human performance on decision-making tasks. Vaccaro & Waldo (2019) focused on understanding the role of human mediation between the recommendation and the human decision maker's bail decision, showing an anchoring bias to the recommendations provided; decision-makers tended to deviate by a number of units depending on the absolute value received from the algorithm. Their analysis suggests that, if the algorithm accuracy increases, algorithm aversion can decrease the positive effect on performance.

The literature on interpretable AI has suggested that making algorithms transparent can help to reduce algorithm aversion. The field of interpretable AI design is growing rapidly, and identifying new ways to design algorithms that are naturally interpretable by humans (e.g. Rudin et al., 2022). The rationale is that people are often averse to recommendations if they cannot interpret the actual prediction results of the algorithm (Ashoori & Weisz, 2019; Brundage et al., 2020). It follows that providing an interpretable model should lead to greater acceptance of the model's recommendations by human decision-makers than providing an equally powerful model that is not interpretable. However, to our knowledge, the interpretability of AI has never been connected to firm performance in field experiments.

3. Empirical Context

The experiment was conducted at Tapestry, Inc. (NYSE: TPR), a leading New York-based house of iconic accessories and lifestyle brands consisting of Coach (founded in 1941), Kate Spade, and Stuart Weitzman, acquired in 2017 and 2015, respectively. In 2022, the Tapestry group counted more than 18,000 employees globally, and sales of $6.7B.

At the supply chain level, the firm (like other retail companies) faces the problem of optimizing product allocations to stores, which means placing the right number of products, at the right time, in the right store to maximize sales and reduce misallocation. The supply chain is organized in a Make to Stock model. Raw materials are purchased, and products are manufactured based on aggregate demand per geographical region (e.g., North America, Asia Pacific, EMEA). The manufactured products are then stored in fulfillment centers (FC) located strategically within the geographical regions. As new products are available in the FC, teams of employees oversee their allocation to stores within the region, deciding how much of each SKU should be sent to each store across the geographic area. This decision is based on short-term forecasts, which require high precision decision making. Therefore, product allocations are optimized if the firm can predict the demand per product in each store at the right time and organize the supply chain accordingly.

Poor allocation accuracy can adversely affect firm performance such as increased stockouts, declines in sales, costs associated with shipping excess product to different stores and so on. For this reason, Tapestry made investments in developing an integrated AI forecasting model with the objective of improving allocation accuracy. Prior to this investment, the demand forecast was traditionally calculated with a ruled-based algorithm, a weighted moving average (WMA) of the known historical sales from the previous weeks.

The ML is a recurrent neural network model which differs from the WMA both in the diverse data inputs it uses and for the sophistication and accuracy of the model (Taddy, 2018). First, concerning the data employed, the ML expands the array of temporal data, using 16 weeks of data to calculate the prediction instead of the 3 weeks considered by the WMA, and it adds to the calculation contextual data like promotions and holidays, which expands the data variety available to make predictions. Second, the RNN contains a more complex function to turn inputs into output predictions. Unlike, the WMA which is interpretable by human decision makers, the ML model is considerably more complex, rendering it unfeasible for human decision makers to interpret its decision-making process or actual prediction results.

In this experiment, we study the allocation decisions for the North America region, which entail allocating around 5,000 SKUs to roughly 200 stores[1]. Each allocator in the allocation team working in the greater New York area is assigned to a subset of SKUs. We randomized algorithmic assistance so that half the allocator decisions on how many SKUs to send to each of its stores were assisted with recommendations from the WMA (interpretable) algorithm and half were assisted with the ML (uninterpretable) algorithm. Every week, for each SKU, allocators iterate the following fixed procedure. They select one product from the list in the software interface and they select the model randomly assigned to that product to make the forecast (either WMA or ML).

They receive an algorithmic recommendation from the software on how many units of that SKU will be sold in that store in the following weeks. The decision that allocators are required to make is either to confirm the recommendation received and ship the quantity or to deviate from the recommendation, shipping a different number of products. The allocation decision is hence

[1] The number of SKUs can vary overtime as new products are introduced and older products as discontinued. Within our study we focus on existing products which includes 216 SKU varieties.

informed by the algorithmic recommendation provided by the software, and mediated by the allocators' acceptance of or deviation from the recommendation.

4. Quantitative Methods

Experimental design

A subset of products and locations were used in the experiment. For those products and locations used, the experimenters intervened by providing the predicted demand from the new ML algorithm in place of the prediction from the current WMA system in half of the decision situations. The allocators were free to deviate from the recommendations received from either the WMA algorithm or the ML algorithm, allowing observation of any differences in the allocators' use of the recommendations from the two sources.

The product-locations used were determined primarily to satisfy feasibility constraints. The products included were selected to have at least 16 weeks of historical sales to satisfy the needs of the ML algorithm, and priority was given to products that were frequently allocated to maximize the number of observations during the experiment.

The experiment took place over 3 weeks in the summer of 2021. Sales and inventory data was collected for an additional two weeks to construct outcome measures for the final weeks of the experiment. Randomization occurred at the "style-color" level for compatibility with the existing workflow. Style-color is a slightly coarser identifier than product. Whereas "product" identifies a particular style, in a particular color, and particular size, style-color groups different sizes together. For example, a black, flat-style shoe may be available in sizes from 5 to 11. In the experiment, all sizes were treated in the same way; algorithmic recommendations were provided for all sizes or none of the sizes. Randomization was stratified by "department". Departments group related products together, e.g., "Women's Bags". There are 17 departments in total. The

experiment used a balanced, 1x2 design with 50 percent of the style-colors in the treated group and 50 percent in the control group. In total 241 style-colors were used in the experiment, allocated across 186 locations.

Data

The dataset is an unbalanced panel of 17,245 allocation decisions. Each decision is identified by a product, retail location, and week. The products are allocated 79.84 times on average with a maximum of 402 and a minimum of 1.

For each decision we observe the product, retail location, timestamp, the allocator anonymized, number of units allocated, the demand forecast of the WMA, the demand forecast of the ML system, and the system experimentally assigned for the allocation. For each product we observe its department. In addition, we used a supplementary dataset of historical sales data to calculate a priori characteristics of the products and retail locations. Using data from January 2019 to January 2021 (two years prior to the experiment), we calculate average sales volume at each retail location.

Based on input from the partner firm, outcomes two weeks from the point of the decision were used to assess decision-making. We observe inventory levels in units and sales volume in units and dollars. We consider a stockout has occurred if for a given product at a given location in a given week inventory is less than or equal to zero.

Summary statistics on the main variables used for the analysis are included in **Error! Reference source not found.**. The treated group (ML algorithm) on average allocated less product than the control group, while at the same time the recommendations received by the treatment group were lower on average (14.08 units) than the control group (21.56). Deviation on average for treatment group was lower than those in the control where the absolute value deviation (measured by allocation-recommendation) is 10.62 and 13.33, respectively. In terms of average

performance, the treatment group has higher inventory, sales units and net revenue and lower stockouts than the control group. The definitions for all variables used in the study can be found in Table A1 in the Appendix.

Table 1 Summary statistics of main variables across all, treatment, and control groups

	All		ML		WMA	
	Mean	SD	Mean	SD	Mean	SD
Allocation	12.35	23.76	12.05	22.07	12.69	25.59
Recommendation	17.54	49.3	14.08	38.76	21.56	59.02
Deviation	11.87	32.08	10.62	23.15	13.33	40
Stockouts	0.29	0.45	0.23	0.42	0.35	0.48
Beginning Inventory	12.9	28.46	13.56	29.65	12.14	26.98
Sales Units	2.48	6.42	2.55	6.81	2.4	5.93
Net Revenue	164.47	485.82	178.42	568.67	148.22	365.68

5. Quantitative Analysis

The following section presents the empirical results from our quantitative field experiment to test the effect of human decision makers' use of an interpretable (weighted moving average with clear inputs – WMA) versus an uninterpretable algorithmic model (recurrent neural network – Machine Learning), under conditions of high perceived uncertainty, on the dual outcomes of 1) human decision maker acceptance of algorithmic recommendations and 2) human decision maker task performance. We start by statistically confirming that ML generated distinct product quantity recommendations in comparison to the WMA. We then test the robustness of our treatment on product allocation by incrementally including fixed effects and by deriving robust standard errors. To identify whether our treatment affects the decision making of allocators, we control for the recommendations provided by either the WMA or the ML algorithm. We next turn to assessing the effect of ML on decision-making task performance as measured by stockouts, beginning inventory, sales volume, and sales value. Finally, we examine if there are heterogeneous effects in our treatments both in terms of performance and allocation by a priori store sales volume.

ML algorithm and system recommendations

The first section of our empirical strategy is to confirm whether the ML recommendations affect the average size of the recommendation seen by allocators. We examine the experimental outcomes within a regression framework. The simplest specification is illustrated in Equation 1:

$$y_{ijt} = \beta_{ML} 1_{ML} + \mu_{dep} + \epsilon_{ijt}$$

Equation 1

y_{ijt} is the outcome of interest which captures recommendations of product-store-time presented to allocations. $\beta_{ML}$is the coefficient on an indicator for the recommendation being included in the treatment group. Note the indicator is for inclusion in the treatment group, not that

the ML recommendation was used. As such, we are presenting intent-to-treat estimates of the treatment effect. As the randomization was stratified by department, we include department fixed-effects, μ_{dep}. The coefficient of interest is β_{ML} measures the average effect of the ML intervention on recommendations.

The result in Table 2 demonstrates the average effect of the ML intervention on the system recommendations. As expected, there is a detectable difference in the average recommendation produced by the ML system. This naïve estimate suggests that ML induced a statistically significant reduction in the amount of product recommendations, by an average of 6 products.

Table 2: Effect of ML on system recommendations

Dependent Variable: Recommendation	(1)
Treatment	-6.6100***
	(0.7669)
R Squared	0.08
Observations	17,245

*Note: The dependent variable represents system recommendation quantities. The treatment variable takes the value of one when recommendations are made with ML and zero otherwise. The model includes department fixed effects. Statistical significance is defined as follows *** p<0.01, ** p<0.05, * p<0.10.*

ML algorithm and product allocation

After confirming the statistical difference in recommendation quantities between the two technologies, next we assess the extent to which the treatment impacts allocation decisions within the firm (see Equation 2).

$$y_{ijt} = \beta_{ML} 1_{ML} + \beta_r r_{ijt} + \mu_{dep} + \nu_t + \delta_{all} + \epsilon_{ijt}$$

Equation 2

y_{ijt} signifies the allocation quantity of a product i location j combination at week t. β_{ML} captures the average effect of the ML intervention on allocation quantities. β_r is the coefficient on the recommended allocation from the system, either from the WMA in the case of a control

observation or from the ML system in the case of a treated observation. In addition to department μ_{dep}, we also include week and allocator fixed effects specified by ν_t and δ_{all}, respectively.

The results in Table 3 illustrate the effects of ML algorithm on allocation decisions. Model One to Model Three incrementally add department, week, and allocator fixed effects. The coefficient in Model Three suggests that the average effect of the ML algorithm intervention is to reduce allocation by about 1.7 units. This estimate combines the effect of the difference in recommendations coming from the ML algorithm as well as the effect of any difference in the allocators' response to the recommendation. To isolate the behavioral change in allocator decisions we control for the system recommendation in Model Four. We find that the ML algorithm leads to an average increase in the allocation decision by roughly one product (0.918).

Table 3 Effect of ML on allocation decisions. Specifications incrementally include fixed-effects

Dependent Variable: Allocation	(1)	(2)	(3)	(4)
Treatment	-1.3250***	-1.7820***	-1.6963***	0.9178***
	(0.3813)	(0.4034)	(0.4033)	(0.2370)
Recommendation				0.3704***
				(0.0139)
Controls				
Department	✓	✓	✓	✓
Week		✓	✓	✓
Allocator			✓	✓
R Squared	0.10	0.11	0.11	0.65
Observations	17,245	17,245	17,245	17,245

*Note: The dependent variable reflects allocation units. The treatment variable takes the value of one when decisions are made with ML and zero otherwise. Model One to Model Four, incrementally add fixed effects including department, week, and allocator and Model Four controls for recommendations provided to allocators. Robust standard errors are in parenthesis. Statistical significance is defined as follows *** p<0.01, ** p<0.05, * p<0.10.*

ML algorithm and human decision maker task performance

To assess the effects of the ML algorithm on decision maker task performance we augment Equation 2 by replacing y_{ijt} with measures of performance which are expected to improve when

allocation decisions are made with better recommendation. We examine the effect of the ML algorithm on the probability of stockouts, inventory levels, units sold, and revenue at the product-location and week level. To estimate these regressions, we rely on the most restrictive specification consistent with Model Four in Table 3, which controls for department, week, allocator, and recommendations.

The results in Table 4 indicate changes to product allocation decisions induced by the ML algorithm had significant effects on performance outcomes. We find that ML algorithm resulted in a reduction in the probability of stock outs, increases in inventory levels, raised sales quantity, and raised revenue. The average effect of each outcome is precisely estimated and highly significant at the 1% level. ML reduced the probability of stock outs by 9%, which led to higher sales of about half a product (0.62) and increased sales by $38.74 on average per allocation.

Table 4 Effect of ML on the probability of stockouts, inventory levels, units sold and net revenue

	(1)	(2)	(3)	(4)
Dependent Variable	Stock outs	Beginning inventory	Sales quantity	Sales value
Treatment	-0.0860***	2.5326***	0.6180***	38.7361***
	(0.0069)	(0.4465)	(0.1060)	(8.3883)
Recommendation	0.0000	0.1671***	0.0530***	3.2377***
	(0.0001)	(0.0234)	(0.0062)	(0.5506)
R Squared	0.18	0.17	0.23	0.19
Observations	17,245	17,245	17,245	17,245

*Note: The dependent variables include probability of stockouts, inventory levels, product units sold, and net revenue. The treatment variable takes the value of one when decisions are made with ML and zero otherwise. All models include department, time, and allocation fixed effects and controls for system recommendations. Robust standard errors are in parenthesis. Statistical significance is defined as follows *** p<0.01, ** p<0.05, * p<0.10.*

Heterogeneous treatment effects on allocation decisions and performance

While we find significant average effects of ML algorithm on performance, it is important to understand where and how these performance gains were achieved. The next section of the

analysis examines the existence of heterogeneous treatment effects. To examine this heterogeneity by the average historic sales volume, quintiles of the sales distribution across location were calculated. A set of dummy variables was created indicating the quintile. The dummies are then included in the regression specification as well as interacted with the treatment indicator to calculate the treatment effect at each quintile.

$$y_{ijt} = \beta_{ML} 1_{ML} + \Gamma_j \cdot 1_{ML} \cdot \mathrm{B}_{MLq} + \Gamma_j \cdot \mathrm{B}_q + \beta_r r_{ijt} + \mu_{dep} + \nu_t + \delta_{all} + \epsilon_{ijt}$$

Equation 3

Γ_j is a vector of dummy variables indicating the quintile with the distribution of sales for the j^{th} location (see Equation 3). When assessing the heterogeneous treatment effects on allocation y_{ijt} refers to allocation units by product-location and time. When assessing the heterogeneous treatment effects on performance, y_{ijt} signifies, probability of stockouts, inventory levels, sales units, and revenue by product-location and time.

The specification in Equation 3 is used to estimate the heterogeneous effects of ML on product allocation decisions by historic location sales volume. The results represented in Table 5 highlight significant heterogeneity with the effect only becoming significant in the highest quintile where allocators tended to increase allocations by almost 3 units relative to similar incumbent recommendations. This result implies that much of the increase in product allocation induced by ML is driven by allocation to the highest selling stores.

To see whether the performance delta due to ML adoption is achieved amongst certain store locations, we estimate Equation 3, where we replace allocation as a dependent variable with probability of stockouts, inventory, sales units, and net revenue. The results in

Table 6 also find marked heterogeneity in the treatment across location quintiles for certain outcomes. The estimates show larger magnitude effects in the highest quintile for stockouts and revenue. Focusing on Model Four, we find that ML algorithm leads to an increase in sales revenue by $36.95 in the 4th quintile and by $104.96 in the 5th quintile on average per allocation. Consistent with where more product is being allocated to, we find statistically significant gains in stockouts and revenue at the highest selling stores.

Table 5 Heterogeneous effects of ML on Allocation decision by the quintile of prior location sales volume

Dependent Variable: Allocation	(1)
Treatment	0.0800
	(0.2308)
Treatment*Location Sales Q2	-0.3886
	(0.3291)
Treatment*Location Sales Q3	-0.0652
	(0.3869)
Treatment*Location Sales Q4	0.2923
	(0.4877)
Treatment*Location Sales Q5	2.9539***
	(0.8472)
Location Sales Q2	0.6779**
	(0.2691)
Location Sales Q3	1.0979***
	(0.3523)
Location Sales Q4	2.4717***
	(0.4466)
Location Sales Q5	4.9284***
	(0.6327)
Recommendation	0.3569***
	(0.0144)
R Squared	0.66
Observations	17,245

*Note: The dependent variables allocation units. The treatment variable takes the value of one when decisions are made with ML and zero otherwise. Quintile interactions are derived from ex-ante average location sales volume (2 years before the start of the experiment). All models include department, time, allocation and recommendation controls. Robust standard errors are in parenthesis. Statistical significance is defined as follows *** p<0.01, ** p<0.05, * p<0.10.*

Table 6 Heterogeneous effects of ML on stockout probability, inventory levels, sales units and net sales by the quintile of prior location sales volume

	(1)	(2)	(3)	(4)
Dependent Variable	Stock outs	Beginning inventory	Sales quantity	Sales value
Treatment	-0.0493***	1.2865***	0.3011***	-3.2149
	(0.0137)	(0.3703)	(0.0744)	(5.9650)
Treatment*Location Sales Q2	-0.0367*	0.1762	0.1502	13.5315
	(0.0200)	(0.5082)	(0.1122)	(8.6533)
Treatment*Location Sales Q3	-0.0369**	0.3945	0.0951	12.3060
	(0.0184)	(0.6151)	(0.1305)	(9.9633)
Treatment*Location Sales Q4	-0.0501**	1.1172	0.2332	36.9570***
	(0.0195)	(0.7998)	(0.1578)	(11.3892)
Treatment*Location Sales Q5	-0.0589***	1.3801	0.5358	104.6963***
	(0.0200)	(1.6233)	(0.3708)	(31.1632)
Location Sales Q2	0.1027***	0.6212*	0.2786***	14.7734**
	(0.0148)	(0.3579)	(0.0863)	(6.7989)
Location Sales Q3	0.0003	4.7150***	0.7223***	41.9336***
	(0.0139)	(0.5045)	(0.1203)	(10.1797)
Location Sales Q4	0.0692***	5.6253***	1.1336***	62.3735***
	(0.0146)	(0.7240)	(0.1642)	(13.6249)
Location Sales Q5	0.0904***	13.9734***	2.4316***	129.4310***
	(0.0151)	(1.2422)	(0.2910)	(26.3256)
Recommendation	-0.0001	0.1365***	0.0475***	2.8674***
	(0.0001)	(0.0246)	(0.0065)	(0.5765)
R Squared	0.19	0.20	0.25	0.21
Observations	17,245	17,245	17,245	17,245

*Note: The dependent variables include probability of stockouts, inventory levels, product units sold, and net revenue. The treatment variable takes the value of one when decisions are made with ML and zero otherwise. Quintile interactions are derived from ex-ante average location sales volume (2 years before the start of the experiment). All models include department, time, allocation and recommendation controls. Robust standard errors are in parenthesis. Statistical significance is defined as follows *** p<0.01, ** p<0.05, * p<0.10.*

Finally, we examine the mediating factor of human deviation from the recommendations received. The difference between the allocation and the recommendation can be interpreted as a measure of acceptance or rejection of the algorithm's recommendation.

Table 7 shows the estimates of the heterogeneous effects of the ML algorithm on absolute deviations using the specification in Equation 3. Allocators' behavior varies across the sales distribution with absolute deviations when using the ML algorithm falling more than 2 units in the highest quintile relative to the bottom quintile. Moreover, the effect is increasing from the lowest to highest selling stores. These results demonstrate that allocators accept ML recommendations more when the product will go to the highest selling stores. Conversely, allocation decisions are more likely to deviate when the product will go to the lowest selling store. These results imply that the performance gains achieved by ML occur when allocation decisions deviate less from the recommendations made by ML.

Table 7 Heterogeneous effects of ML absolute deviation decisions by the quintile of prior location sales volume

Dependent Variable: Deviation	(1)
Treatment	1.6371***
	(0.3182)
Treatment*Location Sales Q2	-1.0807***
	(0.3158)
Treatment*Location Sales Q3	-1.4347***
	(0.4164)
Treatment*Location Sales Q4	-1.9643***
	(0.5232)
Treatment*Location Sales Q5	-2.0249***
	(0.7705)
Location Sales Q2	0.5912
	(0.4593)
Location Sales Q3	-0.8132
	(0.5000)
Location Sales Q4	-1.4197**
	(0.6213)
Location Sales Q5	-3.0792***
	(1.0235)
Recommendation	0.5576***
	(0.0180)
R Squared	0.70
Observations	17,245

*Note: The dependent variable is the absolute value of deviation=abs(allocation-recommendation). The treatment variable takes the value of one when decisions are made with ML and zero otherwise. Quintile interactions are derived from ex-ante average location sales volume (2 years before the start of the experiment). All models include department, time, allocation and recommendation controls. Robust standard errors are in parenthesis. Statistical significance is defined as follows *** p<0.01, ** p<0.05, * p<0.10.*

6. Qualitative Methods

We carried out a post-experiment interview study to understand this unexpected pattern of allocator acceptance of recommendations from and superior task performance on decision making informed by the uninterpretable model versus the interpretable model in situations with a similar level of model performance under conditions of high perceived uncertainty.

Sample characteristics

We conducted 31 interviews with 14 employees—2 developers, 2 managers, and 10 allocators— during the Spring and Summer of 2022 (Table 9). In the field experiment, 1 manager and 9 allocators were randomized to receive from the interpretable or uninterpretable (ML) algorithm algorithmic assistance with decision making around how many products to send to each of its stores. We interviewed this 1 manager and all 7 of the 9 allocator participants who were still employed at the organization at the time the interviews were conducted. The 2 allocators not included in the sample left the organization for reasons of relocation rather than performance.

In addition to interviewing these 8 employees who had been participants in the experiment, we also interviewed 3 allocators who were not employed at the organization at the time of the experiment, but who had post-experiment experience with receiving algorithmic assistance from both the interpretable and the uninterpretable algorithm. Finally, to gain additional information on the broader context, we interviewed 3 employees from other departments—2 developers from the data science department and 1 manager from the IT department— who had helped to develop the uninterpretable machine learning algorithm.

Table 9: Interview Study Sample

Organizational Position	Participant in experiment?	Number of Interviews
Developer	0	4
Developer	0	4
IT Manager	0	3
Allocation Manager	1	3
Allocator	1	3
Allocator	1	3
Allocator	1	3
Allocator	1	1
Allocator	1	1
Allocator	1	1
Allocator	1	1
Allocator	0	1
Allocator	0	1
Allocator	0	2
Allocator*	1	0
Allocator*	1	0
Total	10	31

Note: The two allocators not included in the interview study sample left the organization before the interviews were conducted for reasons of relocation rather than performance

Interview Questions

In our interviews with developers, managers, and the allocators involved in the development of the algorithms, we asked about the development process for both the interpretable algorithm and the uninterpretable algorithm. In particular, we asked about how workflows were mapped and datasets were constructed during model design, how the model was built, validated and tested for accuracy during model development, and how the model was incorporated into allocator workflows during model integration.

In our interviews with the allocators, we asked how they experienced making allocation decisions for low volume stores versus for high volume stores. And, we asked how they decided whether or not to accept recommendations from each type of algorithm. Crucially, interviews confirmed the pattern of allocator acceptance of recommendations that we had identified from our

earlier quantitative field experiment: that is, allocators noted that they perceived a high degree of uncertainty when making allocation decisions for high volume stores, and that they were less likely to accept recommendations from the weighted moving average algorithm (interpretable) than from the ML-based algorithm (uninterpretable) when making these decisions.

7. Qualitative Analysis

Factors previously shown to be related to algorithm aversion cannot explain puzzling findings from the field experiment

The interpretable and uninterpretable algorithms were well matched on the contextual and organizational factors that have been shown to be important to algorithm aversion—the conscious or unconscious display of reluctance to accept algorithmic recommendations. Thus, these factors cannot explain our puzzling findings of why an interpretable model was associated with lower rather than higher acceptance of algorithmically based recommendations, and performance on algorithmically informed decision-making tasks than was an uninterpretable model.

Both algorithms provided assistance to the same set of individuals. As noted earlier, we randomized algorithmic assistance throughout the company's allocation decision-making processes where half the employee decisions were assisted with recommendations from interpretable algorithm (weighted moving average) and the other half of the employee decisions were assisted with recommendations from the uninterpretable algorithm (ML-based). Both algorithms also delivered guidance around the same decision-making task. As noted earlier, the two algorithms were designed to assist employees with decision making around how many products to send to each of its stores.

Both algorithms supported decision making situations that allocators perceived to be characterized by a high degree of uncertainty. For example, allocators noted that, regardless of

which algorithm was offering recommendations, allocation to high volume stores was characterized by a high degree of uncertainty. On allocator noted:

> "[For higher selling stores], you're feeding into sales, and it's always volatile. Things are more subjective at the high end, so it's less clear how much to send."

In contrast, allocation to low volume stores was less uncertain for allocators because they used particular "rules of thumb" around "minimum quantities." One allocator explained:

> "There's no hard and fast rule. But you generally want stores to have enough to have 1 unit to sell and 1 to display. So, you're usually not going to send only one unit to a store."

Both algorithms provided recommendations that conflicted with the allocators' own initial judgment regarding how many products to send to each of its stores. Allocators noted that, upon reviewing the recommendations informed by both algorithms, they often felt a mismatch between what their own expertise suggested sending and what the algorithm was recommending they send. The mismatch led allocators to become more uncertain in their judgement and want to reduce their uncertainty.

New barrier to acceptance of algorithmic recommendations: *Overconfident troubleshooting*

Our analysis identified a new barrier to acceptance of algorithmic recommendations from an interpretable algorithm—*overconfident troubleshooting*—that helps to explain why an interpretable model was associated with lower rather than higher acceptance of algorithmically based recommendations, and performance on algorithmically informed decision-making tasks, than was an uninterpretable model. We found that the allocators engaged in interrogation of the interpretable model and in troubleshooting of counter-intuitive recommendations. However, their very ability to engage in interrogation and troubleshooting resulted in their lower acceptance of the recommendations from the interpretable algorithm.

Allocators attempted to reduce their uncertainty by interrogating and troubleshooting the reasoning behind the algorithmic recommendation by viewing key inputs to the recommendation. As noted earlier, the mismatch between allocator initial judgement and algorithmic recommendation led allocators to become more uncertain in their judgement and want to reduce their uncertainty. In the case of the interpretable algorithm, allocators reported that they attempted to troubleshoot the reasoning behind the algorithmic recommendation in order to reduce their uncertainty. One allocator noted:

> "With [the interpretable model], I can check to see why the system is making the recommendation it is making. I can see all of the numbers that [the interpretable model] is basing its recommendations on. So, for example, I can see that it's basing its recommendation on 3 weeks of trend. If there were inconsistencies across the 3 weeks—it was 200, then 5, then 5—I can see this information."

Similarly, another allocator reported:

> "The [interpretable model] is making recommendations based on the last 3-4 weeks of sales and projected weeks of inventory on hand. It doesn't take into account that there may have been greater sales in the last few weeks because of a holiday or a bump in traffic due to a specific event. If I look in [the interpretable model], I can see that one week ago we sold 4 units, two weeks ago we sold 12 units, and 3 weeks ago we sold 4 units."

In our interviews, allocators reported how they often created narratives for themselves to explain the inner workings of the interpretable algorithm. And, because they believed they understood the causes, effects, and inner workings of the algorithm, this often led them to overrule the algorithm's recommendations. One allocator said:

> "If I see store SKU inconsistencies across the 3 weeks, I would say, '[interpretable model] is telling me to send 1500 units, but I'm not sure I'm comfortable doing that, because I see the inconsistencies. There was a really high week at 200. I think, there was probably a convention in Las Vegas. So, I'm going to take it down a little."

Another allocator noted:

> "I can see that one week ago we sold 4 units, two weeks ago we sold 12 units, and 3 weeks ago we sold 4 units. I assume that the store had an event

> two weeks ago that led to the spike in sales to 12 units. [I make a guess that] maybe this was due to the NFL draft. I don't know that for sure, but it must be something like that to explain that pattern. Whenever there is an unaccounted for event, then sales spike. I assume that [the interpretable algorithm] isn't taking into account things like a bump in traffic for a specific event. It's probably recommending that I send more units than I really should send. So I would adjust it down from what [the interpretable algorithm] recommended."

A third allocator explained:

> "We had a view where we could see the past 3 weeks for each SKU, what was sold and what was in inventory. So, if saw 10 units sold, then 20 units sold, then 200 units sold, then I would guess that there was probably a price change. Like, if the [name of the product] price was cut from 60% off to 70% off, then sales would probably go from 50 units to 100 units. So, if I see a jump in sales, then I know that it was probably due to a price cut. I wouldn't want [the algorithm to apply] weeks on hand to that sale for the week, because there was a probably a price cut. So, I'd overrule that recommendation."

Our results from the interviews suggest that allocators' interrogating and troubleshooting of the interpretable algorithm often led them to reject its recommendations. Allocators reported that they often created narratives for themselves to explain the relationship between model inputs and outputs, and that this often led them to overrule the interpretable algorithm's recommendations. Our results from the quantitative experiment show that allocators' lower acceptance of the recommendations from the interpretable algorithm than from the uninterpretable algorithm in situations with a similar level of model performance resulted in lower rather than higher task performance. Taken together, these results suggest that, unexpectedly, under conditions of high perceived uncertainty, allocators' interrogating and troubleshooting of the interpretable algorithm was associated with their lower acceptance of recommendations from this algorithm; in turn, allocators lower acceptance of recommendations from the interpretable algorithm was associated with lower performance on the algorithmically informed decision-making task of deciding how much of each SKU should be sent to each store.

It is important to point out that overconfidence bias alone cannot explain the difference in acceptance of algorithmic recommendations from the interpretable versus the uninterpretable algorithm. When deciding whether to accept algorithmic assistance from both algorithms, the allocators were equally confident in their general knowledge about how many products to send to stores. Indeed, as noted earlier, both algorithms provided assistance to the same set of individuals. What was different between the two kinds of situations was that, in the case of the interpretable algorithm, being able to see model inputs allowed allocators to create a narrative that supported their overconfidence in particular decision-making situations.

New facilitator of acceptance of algorithmic recommendations: *Social proofing the algorithm*

Our analysis also identified a new facilitator of acceptance of algorithmic recommendations from an uninterpretable algorithm—involvement of respected peers in the development and testing process—that helps to explain why an uninterpretable model was associated with higher rather than lower acceptance of algorithmically based recommendations, and performance on algorithmically informed decision-making tasks than was an interpretable model.

Allocators attempted to reduce their uncertainty by incorporating the opinions of others whom they perceived as having a similar knowledge base and experience—peers who had been involved in the development of the uninterpretable algorithm.

As noted earlier, the mismatch between allocator initial judgement and algorithmic recommendation led allocators to become more uncertain in their judgement and want to reduce their uncertainty. Yet, in the case of the uninterpretable algorithm, allocators were not able to interrogate and troubleshoot the reasoning behind the algorithmic recommendation in order to reduce their uncertainty. One allocator explained:

> "I often get recommendations that seem wonky from both models. But, with [the uninterpretable model], I can't see what goes into the recommendation."

In this ambiguous situation where allocators were unable to determine the appropriate mode of action, they sought to reduce their uncertainty by incorporate the opinions of others who they perceived themselves as similar to through the use of social proof. In particular, they incorporated the opinions of peers who had been involved in the development process. One allocator said:

> "Since I couldn't see what was behind the recommendation, I was only willing to accept it because I knew that [peers] had spent a lot of time with the developers beforehand making sure that the model was accurate. So, you can't know why it's recommending what it is in each case, but [peers] told us how accurate the [uninterpretable] model was."

Another allocator explained:

> "With [the uninterpretable algorithm, we often didn't agree with particular recommendations. It's not like we trusted the model at the level of the recommendation. It's that we trusted it at the more macro level, because [peers] had been involved in development."

A third allocator noted:

> "We heard that initially [during development], the model wasn't accurate. But [peers] worked with [developers] to rectify the problems. After working through those touchpoints, [peers] felt like those issues were corrected, and they became willing to trust the model."

It is important to point out that involvement of peers in the development process alone cannot explain the difference in acceptance of algorithmic recommendations from the interpretable versus the uninterpretable algorithm. Allocators' peers were involved in the development of both the interpretable and the uninterpretable algorithm. It was the combination of peer involvement in development with not being able to interrogate the uninterpretable model that made allocators more likely to accept recommendations from the uninterpretable model than from the interpretable model.

8. Discussion

Uninterpretable algorithm was associated with higher recommendation acceptance, and higher decision-making task performance

Prior research on algorithmically advised human decision making—where humans receive an algorithmically-based recommendation before making a final decision— has focused on the need for inherently interpretable algorithms, and has assumed that making algorithms more amenable to human interrogation and troubleshooting will result in higher human acceptance of algorithmic recommendations and higher performance on decision making tasks. This has been suggested to be particularly important in situations of high uncertainty. Indeed, it has led scholars of interpretable algorithmic models to go so far as to argue that: "Let us consider a possible mandate that, for certain high-stakes decisions, no black box [algorithmic model] should be deployed when there exists an interpretable model with the same level of performance" (Rudin, 2019, p. 10).

Our analysis suggests otherwise. In the present paper, we have demonstrated how the concepts of *overconfident troubleshooting* and *social proofing the algorithm* add important nuance to the current literature's understanding of human-in-the-loop decision making. Here, we expand our discussion of these concepts to explore how they can inform our understanding of algorithmically advised human decision making.

Ironically, under conditions of high uncertainty, human decision makers' interrogating and troubleshooting of interpretable algorithms can lead to lower acceptance of algorithmically based recommendations, and lower performance on algorithmically informed decision-making tasks (Figure 2). Understanding how humans navigate human-in-the-loop decision making when using

an interpretable versus uninterpretable algorithm is vital to understanding algorithmically informed outcomes. Two key theoretical implications follow from our analysis.

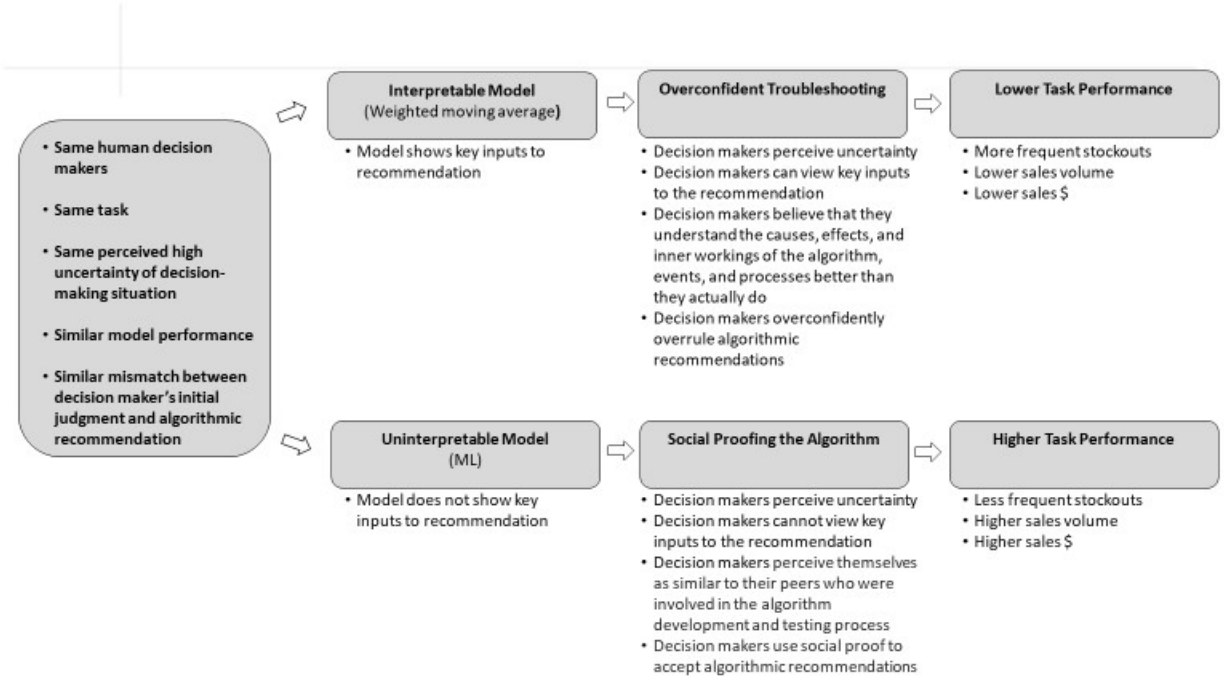

Figure 2: Overconfident troubleshooting, social proofing the algorithm, and algorithmic acceptance and task performance.

New barrier to algorithmic recommendation acceptance and task performance: *Overconfident troubleshooting*

Research on human-in-the loop decision making has depicted that well-intentioned human decision makers are often thwarted in using recommendations provided by algorithmic systems because of their inability to interpret the algorithm's actual prediction results. Nevertheless, despite extensive and granular depictions of interpretation-related barriers to algorithmic recommendation acceptance, the extant literature assumes that human decision makers have little need to navigate barriers related to their own human decision -making biases. Sometimes, the literature implicitly assumes that if humans can understand the reasoning behind algorithmic recommendations, they will be more likely to accept them and, thus, make more accurate decisions; other times, this

literature explicitly suggests that providing humans with interpretable models will be associated with better performance on decision making tasks.

In contrast, we find that, in situations where human decision makers perceive a high degree of uncertainty, they may, indeed, engage in interrogation of the model and in troubleshooting of counter-intuitive recommendations. However, their very ability to engage in interrogation and troubleshooting may result in lower acceptance of the recommendations and lower task performance. These outcomes arise from what we call *overconfident troubleshooting*.

Humans and algorithms can only outperform algorithms alone when humans appreciate what they know and do not know (Fugener et al., n.d.). Overconfidence bias can lead humans to misperceive that their personal abilities, including their own knowledge, are better than they really are. Further, because of the "illusion of explanatory depth" (Rozenblit & Keil, 2002)—humans' belief that we understand the causes, effects, and inner workings of complex mechanisms, events, and processes much better than we actually do—allowing human decision makers to interpret an algorithm may make it more likely that they reject the recommendation coming from the algorithm.

New facilitator of algorithmic recommendation acceptance and task performance: S*ocial proofing the algorithm*

We also contribute the insight of *social proofing the algorithm* rather than interpretable models as a facilitator of algorithmic recommendation acceptance and decision-making task performance. We found that human decision makers were persuaded to accept algorithmic recommendations even when they could not understand the reasoning behind them because they knew that people like them—people with their knowledge base and experience—had had input into how and why these recommendations were being made and had tested the performance of the algorithm. We call this *social proofing the algorithm*.

Our finding about the conditions under which humans may be willing to accept recommendations coming from an uninterpretable algorithm is informed by Cialdini's concept of "social proof." Social proof has been shown to be prominent in ambiguous social situations where people are unable to determine the appropriate mode of behavior and is driven by their assumption that the surrounding people possess more knowledge about the current situation than they do themselves. This social proof was likely particularly powerful in the case we studied both because of the situation of perceived uncertainty (which leads people to be more likely to incorporate the opinions of others) and because of decision makers perceived similarity to those who had helped develop the uninterpretable algorithm (people are more likely to incorporate the opinions of others through the use of social proof when they perceive themselves as similar to the people who performed the same actions before them).

Boundary conditions and future research

We expect that *overconfident troubleshooting* will be most important in two contexts that should be kept in mind when drawing on the concepts from this study. First, we are likely to see *overconfident troubleshooting* in contexts where individuals have a lower level of appreciation for what they know and do not know. Research by Fugener and colleagues (2021, 2022) suggests that individuals vary in their ability to assess their own capabilities. Second, we may be more likely to see *overconfident troubleshooting* in decision making contexts where humans have a higher level of algorithm aversion. Recent reviews of the literature on algorithm aversion suggest that algorithm aversion varies not only by situational factors such as perceived uncertainty of decision making task, but also by other task factors such as complexity, subjectivity, and perceived moral nature, by individual factors such as psychology, personality, familiarity, and demography, and by organizational factors, societal factors, and cultural factors (Mahmud et al., 2022).

While it is likely that the kinds of algorithmic recommendation acceptance and decision-making task performance dynamics we observed are present in other settings, our setting allowed us to observe these dynamics in high relief. We faced two constraints, however, by studying human-in-the-loop decision making with a combination of field experiment and interviews in the context of fashion allocation. First, the design of the experiment did not allow us to understand, in real time and in specific situations, why allocators were accepting or rejecting recommendations from the interpretable versus non-interpretable algorithms. We chose not to observe allocators in real time and ask them to explain their reasoning for acceptance versus rejection in particular situations, because this may have changed their decision-making behaviors. Future research could explore the concepts of *overconfident troubleshooting* and *social proofing the algorithm* using other methods.

Second, while the decision-making task situation of fashion allocation to high volume stores was perceived by allocators to be a situation of high uncertainty, this situation is clearly different than situations such as medical diagnosis and treatment or situations such as bail decisions or credit lending decisions in which people's life chances depend on the decision. It is possible that, given the decision-making context, the human decision makers in our study took less time to troubleshoot than they have in the contexts studied by scholars of interpretable algorithms. Future research could explore the concepts of *overconfident troubleshooting* and *social proofing the algorithm* in other decision- making contexts.

Finally, our results raise the question of the conditions under which *social proofing the algorithm* will result in higher decision-making task performance. In our context, the peers who were involved in the development of the uninterpretable machine learning algorithm had input into how and why these recommendations were being made and had rigorously tested the performance

of the algorithm. One could imagine that in other contexts, peers might be involved in algorithm development, but not have their suggestions for improvement incorporated, or that the algorithm might not be rigorously tested. Future research could explore the conditions required to allow *social proofing the algorithms* to result in higher decision-making performance.

9. Conclusion

Advances in AI are increasing the accuracy of recommendations provided to decision-makers in an increasing variety of use cases throughout the economy and society. Contrary to popular belief, AI is rarely fully automating decision making but is instead providing guidance and recommendations to humans who then make the subsequent decision.

Yet, human decision makers often display reluctance to accept algorithmic recommendations, particularly in decision-making situations characterized by uncertainty, a phenomenon known as algorithm aversion. The literature on interpretable AI suggests that providing decision makers with an algorithmic model that is inherently interpretable to humans should result in greater acceptance of algorithmic recommendations and better decision-making performance than providing decision makers with an uninterpretable algorithmic model.

To test these ideas in a real-world setting, we constructed a quantitative field experiment within a large fashion company in North America. We measured the effect of human decision makers' use of an interpretable (weighted moving average with clear inputs) versus an uninterpretable algorithmic model (recurrent neural network – Machine Learning), under conditions of high perceived uncertainty, on the dual outcomes of 1) human decision maker acceptance of algorithmic recommendations and 2) human decision maker task performance. In the experiment, human decision makers were allowed to accept or reject the recommendations made by either the uninterpretable algorithm or the interpretable algorithm.

Contrary to what the literature on interpretable algorithms would predict, we found that, under conditions of high perceived uncertainty, human decision makers' use of the *uninterpretable* algorithmic model was associated with greater acceptance of algorithmic recommendations and greater performance—fewer stockouts and higher sales measured by quantity and value— than was their use of the interpretable model in situations with a similar level of model performance. Our subsequent interview study identified that the concepts of *overconfident troubleshooting* and *social proofing the algorithm* explain these counterintuitive results.

Paradoxically, under conditions of high uncertainty, providing human decision makers with more interpretable algorithms may lead to lower acceptance of algorithmic recommendations and lower performance on decision making tasks. By further testing the concepts of *overconfident troubleshooting* and *social proofing the algorithm*, we may be able to both improve organizational performance and advance the life chances of populations in need. It is certainly a possibility worth exploring.

10. References

Arrieta, A. B., Díaz-Rodríguez, N., Del Ser, J., Bennetot, A., Tabik, S., Barbado, A., García, S., Gil-López, S., Molina, D., Benjamins, R., Chatila, R., & Herrera, F. (2019). *Explainable Artificial Intelligence (XAI): Concepts, Taxonomies, Opportunities and Challenges toward Responsible AI* (arXiv:1910.10045). arXiv. https://doi.org/10.48550/arXiv.1910.10045

Ashoori, M., & Weisz, J. D. (2019). *In AI We Trust? Factors That Influence Trustworthiness of AI-infused Decision-Making Processes* (arXiv:1912.02675). arXiv. https://doi.org/10.48550/arXiv.1912.02675

Bonde Thylstrup, N., Flyverbom, M., & Helles, R. (2019). Datafied knowledge production: Introduction to the special theme. *Big Data & Society*, *6*(2), 2053951719875985. https://doi.org/10.1177/2053951719875985

Brundage, M., Avin, S., Wang, J., Belfield, H., Krueger, G., Hadfield, G., Khlaaf, H., Yang, J., Toner, H., Fong, R., Maharaj, T., Koh, P. W., Hooker, S., Leung, J., Trask, A., Bluemke, E., Lebensold, J., O'Keefe, C., Koren, M., … Anderljung, M. (2020). *Toward Trustworthy AI Development: Mechanisms for Supporting Verifiable Claims* (arXiv:2004.07213). arXiv. https://doi.org/10.48550/arXiv.2004.07213

Burrell, J. (2016). How the machine 'thinks': Understanding opacity in machine learning algorithms. *Big Data & Society*, *3*(1), 2053951715622512. https://doi.org/10.1177/2053951715622512

Dietvorst, B. J., & Bharti, S. (2020). People Reject Algorithms in Uncertain Decision Domains Because They Have Diminishing Sensitivity to Forecasting Error. *Psychological Science*, *31*(10), 1302–1314. https://doi.org/10.1177/0956797620948841

Feng, X., & Gao, J. (2020). Is optimal recommendation the best? A laboratory investigation under the newsvendor problem. *DECISION SUPPORT SYSTEMS*, *131*, 113251-. https://doi.org/10.1016/j.dss.2020.113251

Fugener, A., Grahl, J., Ketter, W., & Gupta, A. (n.d.). *Cognitive challenges in human-AI collaboration: Investigating the path towards productive delegation*. 39.

Grgić-Hlača, N., Engel, C., & Gummadi, K. (2019). Human Decision Making with Machine Assistance: An Experiment on Bailing and Jailing. *Proceedings of the ACM on Human-Computer Interaction*, *3*(CSCW), 1–25. https://doi.org/10.1145/3359280

Henriksen, A., & Bechmann, A. (2020). Building truths in AI: Making predictive algorithms doable in healthcare. *Information, Communication & Society*, *23*(6), 802–816. https://doi.org/10.1080/1369118X.2020.1751866

Herm, L.-V., Heinrich, K., Wanner, J., & Janiesch, C. (2022). Stop ordering machine learning algorithms by their explainability! A user-centered investigation of performance and explainability. *International Journal of Information Management*, 102538. https://doi.org/10.1016/j.ijinfomgt.2022.102538

Kawaguchi, K. (2021). When Will Workers Follow an Algorithm? A Field Experiment with a Retail Business. *Management Science*, *67*(3), 1670–1695. https://doi.org/10.1287/mnsc.2020.3599

Kleinberg, J., Lakkaraju, H., Leskovec, J., Ludwig, J., & Mullainathan, S. (2018). Human Decisions and Machine Predictions. *The Quarterly Journal of Economics*, *133*(1), 237–293. https://doi.org/10.1093/qje/qjx032

Lebovitz, S., Lifshitz-Assaf, H., & Levina, N. (2022). To Engage or Not to Engage with AI for Critical Judgments: How Professionals Deal with Opacity When Using AI for Medical Diagnosis. *Organization Science*, *33*(1), 126–148. https://doi.org/10.1287/orsc.2021.1549

Lennartz, S., Dratsch, T., Zopfs, D., Persigehl, T., Maintz, D., Hokamp, N. G., & Santos, D. P. dos. (2021). Use and Control of Artificial Intelligence in Patients Across the Medical Workflow: Single-Center Questionnaire Study of Patient Perspectives. *Journal of Medical Internet Research*, *23*(2), e24221. https://doi.org/10.2196/24221

Mahmud, H., Islam, A. K. M. N., Ahmed, S. I., & Smolander, K. (2022). What influences algorithmic decision-making? A systematic literature review on algorithm aversion. *Technological Forecasting and Social Change*, *175*, 121390. https://doi.org/10.1016/j.techfore.2021.121390

Rozenblit, L., & Keil, F. (2002). The misunderstood limits of folk science: An illusion of explanatory depth. *Cognitive Science*, *26*(5), 521–562. https://doi.org/10.1207/s15516709cog2605_1

Rudin, C. (2019). Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, *1*(5), Article 5. https://doi.org/10.1038/s42256-019-0048-x

Rudin, C., Chen, C., Chen, Z., Huang, H., Semenova, L., & Zhong, C. (2022). Interpretable machine learning: Fundamental principles and 10 grand challenges. *Statistics Surveys*, *16*(none), 1–85. https://doi.org/10.1214/21-SS133

Rudin, C., & Radin, J. (2019). Why Are We Using Black Box Models in AI When We Don't Need To? A Lesson From An Explainable AI Competition. *Harvard Data Science Review*, *1*(2). https://doi.org/10.1162/99608f92.5a8a3a3d

Sutherland, S. C., Harteveld, C., & Young, M. E. (2016). Effects of the Advisor and Environment on Requesting and Complying With Automated Advice. *ACM Transactions on Interactive Intelligent Systems*, *6*(4), 27:1-27:36. https://doi.org/10.1145/2905370

Taddy, M. (2018). *The Technological Elements of Artificial Intelligence* (Working Paper No. 24301). National Bureau of Economic Research. https://doi.org/10.3386/w24301

Vaccaro, M., & Waldo, J. (2019). The effects of mixing machine learning and human judgment. *Communications of the ACM*, *62*(11), 104–110. https://doi.org/10.1145/3359338

Verganti, R., Vendraminelli, L., & Iansiti, M. (2020). Innovation and Design in the Age of Artificial Intelligence. *Journal of Product Innovation Management*, *37*(3), 212–227. https://doi.org/10.1111/jpim.12523

Zhang, L., Pentina, I., & Fan, Y. (2021). Who do you choose? Comparing perceptions of human vs robo-advisor in the context of financial services. *The Journal of Services Marketing*, *35*(5), 634–646. https://doi.org/10.1108/JSM-05-2020-0162

11. Appendix

Figure A1: Relationship between performance and historical location sales

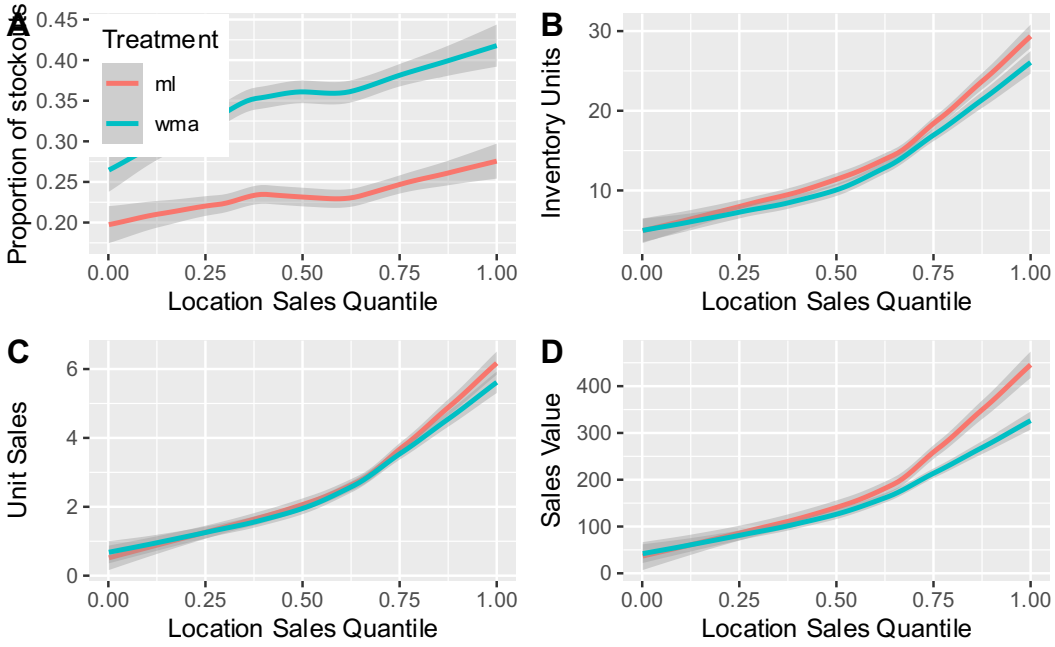

Figure A2: Relationship between (A) units recommended and(B) units allocated with historical location sales

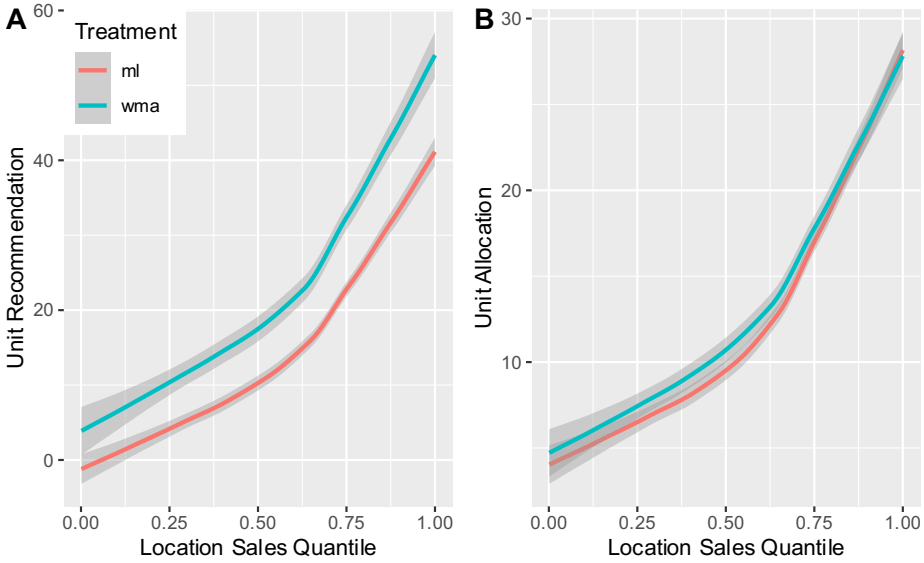

Table A1 Variable definitions

Variable	Description
Location sales volume	Historic average sales volume in units for each retail location, calculated over the period from January 2019 to January 2021.
Allocation	Number of units shipped for a particular product at a particular retail location.
Stockouts	Indicator for the condition that the beginning inventory is less than or equal to zero.
Beginning Inventory	Reported stock of a particular product at a particular retail location at the beginning of a week.
Sales units	Reported sales in units of a particular product at a particular retail location during a week.
Revenue	Reported sales in dollars of a particular product at a particular retail location during a week.
Deviation	The absolute difference between the allocation and recommendation for a particular product at a particular retail location.