



// NO.22-071 | 12/2022

DISCUSSION PAPER

// JAN BIERMANN, JOHN HORTON,
AND JOHANNES WALTER

Algorithmic Advice as a Credence Good

Algorithmic Advice as a Credence Good

Jan Biermann*
John Horton†
Johannes Walter‡

December 2022

Abstract

Actors in various settings have been increasingly relying on algorithmic tools to support their decision-making. Much of the public debate concerning algorithms—especially the associated regulation of new technologies—rests on the assumption that humans can assess the quality of algorithms. We test this assumption by conducting an online experiment with 1263 participants. Subjects perform an estimation task and are supported by algorithmic advice. Our first finding is that, in our setting, humans cannot verify the algorithm’s quality. We, therefore, argue that algorithms exhibit traits of a *credence good* – decision-makers cannot verify the quality of such goods, even after “consuming” them. Based on this finding, we test two interventions to improve the individual’s ability to make good decisions in algorithmically supported situations. In the first intervention, we explain the way the algorithm functions. We find that while explanation helps participants recognize bias in the algorithm, it remarkably decreases human decision-making performance. In the second treatment, we reveal the task’s correct answer after every round and find that this intervention improves human decision-making performance. Our findings have implications for policy initiatives and managerial practice.

Keywords: Human-algorithm decision making, algorithmic advice, credence goods
JEL Codes: C91, D79, D80, M21, O30

*University of Hamburg, jan.biermann@uni-hamburg.de

†MIT Sloan, jjhorton@mit.edu

‡ZEW & KIT, johannes.walter@zew.de

1 Introduction

Human decision makers increasingly receive advice from algorithmic recommendation systems—for example, when physicians decide which patients to give which treatment, when judges make punishment decisions, or when price managers set product discounts at online retailers. A common view among policymakers and in the academic literature implicitly assumes that human decision-makers can accurately assess the advice quality after observing the algorithm’s recommendation, for instance, by comparing it to their own judgment (e.g. in the Artificial Intelligence Act as proposed by the European Commission, 2021). From an economic point of view, this suggests that algorithm advice is perceived as an experience good, i.e., consumers can accurately assess the quality after consumption of the good. We challenge this assumption and argue that algorithms are often perceived to be *credence goods*. Even after “consumption” of the good – repeated interaction with the algorithm – humans cannot correctly assess its advice quality.

Our first contribution is to provide experimental evidence from a reasonable setting that many people cannot correctly assess the quality of algorithmic advice even after “consuming” it. It follows that they perceive algorithmic advice as a credence good.¹ Based on this finding, our second contribution is to test two commonly discussed interventions designed to improve a human’s ability to make good decisions in such situations. First, we provide participants with an explanation of how the algorithm arrives at its recommendation, which could allow them to assess the algorithm more accurately. Second, we reveal the solution to the prediction task after each round. This allows participants to better assess the algorithm’s quality as well as their own ability.

To do so, we conduct an online experiment and asked 1263 participants to estimate how many dots are in an image. Our subjects receive algorithmic advice and are free to choose to what extent—if at all—they want to incorporate this advice in their answer. The task is repeated for 16 rounds, and the image has so many dots that counting them directly is infeasible. As the ongoing debate focuses on a human’s ability to recognize and correct an algorithm when it malfunctions, we manipulate the algorithm and introduce a bias such that the algorithm considerably underestimates the number of dots in each image. Our algorithm exhibits a downward bias, i.e. it systematically underestimates the number of dots.

Over 16 rounds of playing this simple game, participants follow the algorithm closely, such that the average of the revised guesses never moves away from the downward biased algorithm recommendation and closer to the true number of dots. This behavior emerges despite the fact that, just as in many real-life decision situations, our participants can compare their initial guess with the algorithm’s guess in every round and realize that the algorithm performs poorly. This is particularly true for participants who do not benefit

¹In this sense, our work focuses on whether algorithms are perceived as credence goods from a consumer (i.e., decision-maker) perspective. The classic definition of a credence good market includes a second element: an (expert) provider of a good who *can* observe its quality (Darby and Karni, 1973). In this article, we remain agnostic to whether or not the developer of the algorithm can judge its quality.

from following the algorithm because their initial guess was already closer to the truth. Even this group of participants fails to correctly evaluate the algorithm’s input.

Regarding our first intervention, we find that providing an explanation of the algorithm decreases participants’ algorithm adherence, but, remarkably, it hurts their guessing performance. This illustrates important nuances that need to be considered before one can hope to have humans successfully oversee algorithms: The human decision-maker needs not only to recognize the existence of a bias but also to accurately assess the *size* and *direction* of bias. Further, humans need to be able to assess their performance in relation to the algorithm’s performance. In our experiment, participants seem to recognize a bias exists but fail to assess its size and direction as well as their performance in relation to the algorithm, all of which ultimately decrease their performance when seeing an explanation.

Informing participants of the true number of dots at the end of the round also makes participants follow the algorithm less, but in this case, it improves participants’ guessing performance. This is most likely because the correct answer from the last round(s) gives the participants a superior orientation point for future guesses. While participants do not perform very well at guessing the absolute number of dots, they are very much capable of assessing the *relative change* in the number of dots from round to round.²

Although there is already a large and rapidly growing body of literature on empirical human-subject studies on human-AI decision-making in multiple disciplines (for an overview see Lai et al., 2021), we are, to the best of our knowledge, the first to point out and empirically document the credence good nature of algorithmic advice and its implications. For a recent review of the literature on credence goods, see Balafoutas and Kerschbamer (2020). Two papers that are closely related to ours are Green and Chen (2019) and Park et al. (2019). In line with our results, the former paper finds that the participants in their experiment are unable to correctly assess their own ability as well as the algorithm’s quality. The latter paper finds that their participants can assess an algorithm better if the algorithm’s response time is increased. Glaeser et al. (2021) is one of the few field experiments on this topic; it finds that human decision-makers prefer to follow their own judgment despite the fact that the algorithm performs better than humans. Closely related to our topic is also the question about the determinants of and the extent to which humans trust in algorithms. Zhang, Liao, and Bellamy (2020) attempt to calibrate human trust in an algorithm by providing local explanations. Yin et al. (2019) explore how trust in a model is affected by its accuracy. Alufaisan et al. (2021) survey the relevant literature and find that the evidence on whether explainability improves decisions remains inconclusive. Human collaboration with algorithms is also strongly influenced by a variety of psychological effects. Dietvorst, Simmons, and Massey (2018) document algorithm aversion, Logg, Minson, and Moore (2019) find conditions under which humans

²For example, many subjects are capable of realizing that the number of dots in a given round has roughly doubled compared to the last round. Knowing the correct number from the previous round, they can provide good (initial) estimates.

display algorithm appreciation. Moreover, the assessment of an algorithm might also depend on the nature of the task. Castelo, Bos, and Lehmann (2019) show that people are more willing to trust algorithms that are perceived as objective in nature.

Our results have implications for both policymakers and managers. We show that there are situations in which humans are not able to accurately assess an advising algorithm’s quality. In the context of regulation, this casts doubt on the effectiveness of individual human decision-makers to recognize biased algorithms and to correct this bias to prevent harm. In the context of management, it means that organizations generally can neither rely on individual decision-makers to optimize decisions nor can they rely on feedback from their decision-makers about the quality of an algorithm as a product. Just as with any other market for credence goods this entails the possibility of harming consumers of algorithmic advice.

The remainder of this paper proceeds as follows: Section 2 describes the experimental set-up. Section 3 introduces the data and explains some pre-processing important for the analysis. Section 4 presents the results, and section 5 discusses the findings. Section 6 concludes.

2 Experimental Design

2.1 The experimental task

The central component of our experiment is the dot-guessing task. Subjects see images showing dots that follow a triangular distribution (such that they are clustered more densely in the center) and are asked to guess how many dots they think each of the images contains. Examples of dot images are shown in figure 2. The number of dots in the images is chosen randomly and varies between 942 and 3084 dots. Participants have 60 seconds to make their guesses, making it infeasible to count the dots in the image.³

Every round of the experiment consists of three stages: In the first stage, participants see the image for the first time and submit their guesses. In the second stage, subjects see the same image again and additionally receive an algorithmic prediction of the number of dots, before submitting a new guess. We call the two entries the subjects make initial guess $guess_i$ and revised guess $guess_r$, both of which are incentivized. In the third stage, participants can see their guesses $guess_i$ and $guess_r$ and some additional information depending on the treatment. They do not take any action at this stage. Every subject plays 16 rounds, each round including a new image and a new recommendation. In every round, all participants see the same image (i.e. the same number of dots) and receive the same recommendation. We employ a between-subject design and randomize our participants on an individual level.

³Our task is rooted in the tradition of Galton (1907). His research has produced the “wisdom of the crowd” finding and involved a contest in which people guessed the weight of a butchered ox.

2.2 Description of treatments

The treatments vary along two dimensions: explaining how the algorithm arrives at its prediction and revealing the true answer after the revised guess has been recorded.

When `ALGORITHMEXPLAINED` = 1, participants receive both a written explanation regarding how the algorithm derives its prediction and a visual enhancement of the image supporting the verbal explanation. When `ALGORITHMEXPLAINED` = 0 participants never learn about the functional principle of the algorithm. Explaining the algorithm provides an opportunity for participants to assess its quality of it in an abstract way based on the functioning of the algorithm.⁴

Furthermore, when `TRUTHREVEALEDExPOST` = 1 participants receive the information about the correct number of dots in the image at the end of every round. When `TRUTHREVEALEDExPOST` = 0, participants never find out the correct answer to the task. Seeing the solution provides an opportunity to assess the performance of the algorithm in a specific round. It also opens the chance to learn about one’s own performance.

To sum up, the two dimensions result in the four treatments `TRUTHREVEALEDExPOST` = 0 & `ALGORITHMEXPLAINED` = 0, `TRUTHREVEALEDExPOST` = 0 & `ALGORITHMEXPLAINED` = 1, `TRUTHREVEALEDExPOST` = 1 & `ALGORITHMEXPLAINED` = 0, `TRUTHREVEALEDExPOST` = 1 & `ALGORITHMEXPLAINED` = 1. These treatments enable us to analyze which type of information empowers humans to effectively assess the quality of algorithmic advice and how they influence task performance. The experimental design is visualized in figure 1.

2.3 Description of the dot guessing algorithm

We employ a simple dot-counting algorithm. It randomly samples subareas, but only from the edge of an image, calculates the average of dots within these subareas, and extrapolates this average to the entire surface of the image. The dots follow a triangular distribution with a denser center and fewer dots at the edges. By choosing a triangular distribution and limiting the sampled subareas to the edges of the image, we introduce a bias in the algorithm: The algorithm will always predict and recommend a dot number that is too low. Note that this provides participants with a lower bound of how many dots are visible. We do not explicitly tell participants about this bias in the algorithm, although participants in the two treatments where `ALGORITHMEXPLAINED` = 1 have all the necessary information available to arrive at this conclusion themselves. When `ALGORITHMEXPLAINED` = 1, participants in the second stage of each round, i.e. when they can state their revised guess, get to see the same image of dots they saw in the first stage, but this time it is overlaid with squares indicating the subareas the algorithm samples from. This visual overlay is illustrated in figure 2, for more details on how the

⁴To see the exact wording and visual presentation of the explanations, see the experimental interfaces in the appendix.

Figure 1: Visualization experimental design

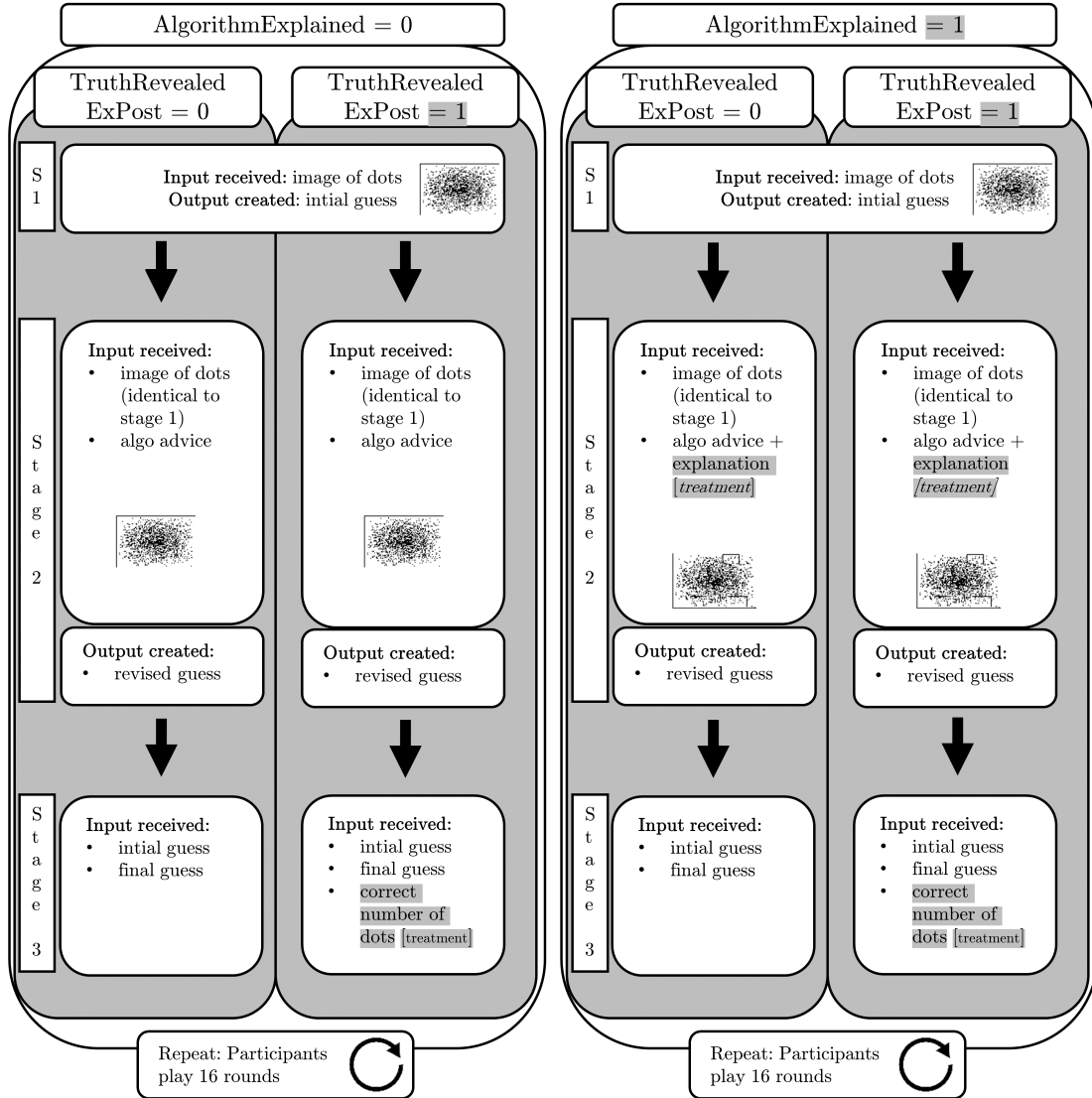
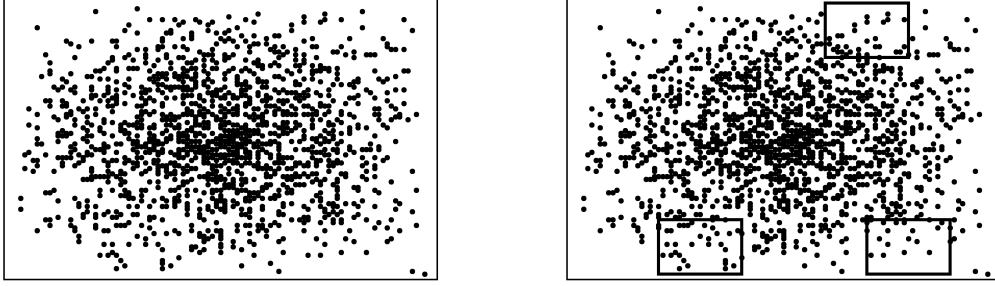


Figure 2: Functioning dot guessing algorithm



(a) Example dot image in treatments with $\text{ALGORITHMEXPLAINED} = 0$

(b) Example dot image in treatments with $\text{ALGORITHMEXPLAINED} = 1$

Notes: The algorithm arrives at its prediction by first randomly sampling three subareas from the edges of each image and counting the number of dots within each subarea. It then calculates the average number of dots over areas and projects this average to the entire image. Importantly, the algorithm always samples the three subareas at the edges of a given image, never from the center. Through the combination of triangular dot distribution and biased sampling from the edges, we introduce a bias in the algorithm prediction. The image in panel (a) is an example of a dot image that all participants see in the first stage of the experiment. Panel (b) shows the same image but this time overlaid with the rectangular subareas from which the algorithm samples dots. Only participants in the treatments where $\text{ALGORITHMEXPLAINED} = 1$ see this image from panel (b) in the second stage, complementing the verbal algorithm explanation.

algorithm works see the explanation in the appendix.⁵

2.4 Payment scheme and experimental procedure

Our subjects receive a flat fee of \$0.9 for completing the study. Every guess a participant makes is incentivized. More specifically, participants receive \$0.15 for a perfect guess and this bonus is diminished by \$0.0002 for every point deviation. That implies that participants receive some bonus when their guess is within the range of ± 749 dots from the true answer. The experiment contains 16 rounds and each of the rounds involves two incentivized guesses (initial and revised). Therefore, subjects could earn a maximum of \$4.80 bonus payment in addition to the flat fee.

We conducted our online experiment in December 2021. The experiment was developed using the software oTree (Chen, Schonger, and Wickens, 2016). We recruited our subjects via Amazon’s crowd-working platform Mechanical Turk (MTurk). All of them are based in the US, have completed at least 500 tasks on MTurk, and have an approval rate of at least 95%. We conducted five sessions (two sessions with 200 and 3 sessions and 400 participants). On average, participants have taken 14 minutes and 18 seconds to

⁵Details on the algorithm’s performance and its performance relative to our participants will be addressed in the results section.

complete the study and have earned \$2.33. This translates to a hypothetical hourly wage of \$9.82.

3 Data

The main data we elicit from our subjects is their guesses with respect to the number of dots in the image they see. When eliciting these guesses, we do not set an upper bound (e.g. by employing a slider) as such an upper bound would serve as an orientation point for some of our subjects. As a result, participants can enter very high numbers, and in fact, some choose to do so. We, therefore, see large outliers in our distributions. Three approaches are common to address the issue of outliers in the data: top-coding, winsorizing, and taking the natural logarithm. We employ the latter method. It has the advantage of not requiring us to exclude any observations from our analysis. Our data contains 1263 observations.

Further, since the number of dots and the algorithmic advice changes every round, examining (the logarithm of) the guesses at their face value would have little meaning. We are rather interested in the relation between guesses and i.) the algorithmic advice or ii.) the correct number of dots. Therefore, we focus on two main outcome variables throughout the paper: algorithm adherence, calculated as $|\log(algo) - \log(guess)|$, and guessing performance, calculated as $|\log(truth) - \log(guess)|$.

Various parts of the analysis are based on a comparison among treatments in which case we pool all rounds together. We recognize that the guesses of each individual are not independent of each other. We, therefore, preprocess the data by conducting the mean of the values of interest (e.g. distance to algorithmic recommendation) of all 16 rounds for each individual.

4 Results

We ask the question of whether people can assess the quality of algorithmic advice (and if they act accordingly) after “consuming” this advice. To answer this question, one can inspect figure 3, which shows the densities of the initial and revised guesses of the first four rounds in treatment `TRUTHREVEALEDExPOST = 0 & ALGORITHMEXPLAINED = 0`.

In the first round, the initial guess density is flat: Participants vary vastly in their initial guess. After observing the algorithm recommendation, participants state their revised guess, resulting in the revised guess density. Participants strongly react to the algorithmic advice and many subjects follow the advice closely. This can be directly inferred from the revised density: It is centered above the algorithm recommendation and its variance is greatly reduced. The second round illustrates that the initial guesses in round two are still influenced by the algorithmic advice from round one: The density of the initial distribution is centered above the previous rounds algorithm recommendation.

The algorithm recommendation in one round serves as an orientation point for the next initial guess. The revised guess density in round two peaks again above the algorithm recommendation. In rounds three and four one again observes the two effects (1) the revised guesses move closer to the algorithm and (2) their variance is reduced (both compared to the initial guess density in the same round). In fact, this pattern holds for all subsequent 12 rounds (see figures 5 to 8).

Table 1 quantifies these differences and shows the average of the individual log distances to the algorithm and the standard deviation for the initial and revised guesses. It also exhibits p-values from a t-test comparing the initial and revised distance to the algorithm recommendation and Levene’s test for homogeneity of variances of the initial and revised densities.

The average distance to the algorithm recommendation is smaller for the revised guesses than for the initial guesses in all 16 rounds (this difference is always significant except for one round). In other words, revised guesses move closer to the algorithm. The standard deviations of the revised guess densities are smaller for the revised guesses in a majority of cases.⁶

If the algorithmic advice would exhibit traits of an experience good, we would expect our subjects to learn to optimally incorporate this advice into their decision-making and adjust how strongly they adhere to the algorithm. Note that in each round, participants have the possibility to compare the algorithm recommendation with their own guess, which was elicited in the first stage. Given their own guess as a reference point, participants could realize that the algorithm recommendation is consistently too low. Over time (i.e. over rounds), if participants would have this realization repeatedly, and assuming they would also optimally react based on this insight, we would see a shift in the revised guess density away from the algorithm and closer to the true number of dots. As can be seen from figure 3 and table 1 we see no evidence for this. In figure 3 there is no increase in probability mass in the region above the algorithm recommendation for the revised guesses over rounds. In table 1 average distance of the revised guesses to the algorithm is in every round smaller than the distance for the initial guesses. In other words, the revised guesses always move closer to the algorithm. So the participants never learn to move further away from the algorithm recommendation (or ignore it).

Table 1: Distances to algorithmic recommendation per round: TRUTHREVEALED-EXPOST = 0 & ALGORITHMEXPLAINED = 0

Round	$ \log(algo) - \log(guess_i) $		$ \log(algo) - \log(guess_r) $		p-values	
	mean	sd	mean	sd	t-test	levene
1	1.29	1.78	0.59	1.39	0.00	0.00
2	1.20	0.73	0.88	1.13	0.00	0.00

⁶Due to outliers that remain even after the logarithmic transformation of our data the p-values of these differences paints a less clear picture.

3	0.78	0.93	0.42	0.74	0.00	0.13
4	0.51	0.67	0.45	1.02	0.28	0.33
5	0.66	0.89	0.35	0.74	0.00	0.77
6	0.67	1.16	0.42	1.00	0.00	0.22
7	0.60	1.06	0.33	0.90	0.00	0.06
8	0.61	0.91	0.37	0.81	0.00	0.21
9	0.58	0.96	0.42	1.03	0.00	0.66
10	0.62	0.99	0.40	0.98	0.00	0.60
11	0.46	0.74	0.31	0.90	0.00	0.69
12	0.47	0.70	0.29	0.68	0.00	0.68
13	0.53	1.08	0.33	0.89	0.00	0.19
14	0.64	0.74	0.42	0.95	0.00	0.82
15	0.52	1.07	0.34	1.01	0.00	0.31
16	0.68	0.89	0.49	0.98	0.00	0.24

Notes: Table contains the mean and standard deviation of the distance between the guesses and the algorithmic recommendation for every round. This distance is included with respect to the initial and the revised guesses. All values are logs. Table also contains the p-values for the t-test and the Levene-test for homogeneity of variance to test the difference between the values for initial and revised guesses for every round. Values refer to $\text{TRUTHREVEALED} = 0$ & $\text{ALGORITHMEXPLAINED} = 0$.

Especially participants whose initial guesses fall within this range between the algorithm and the true value should not move closer to the algorithm to maximize their payoff. Yet, our data shows that participants move closer to the algorithmic advice. Our results therefore rather indicate that the algorithmic advice exhibits traits of a credence good: Even after “consuming” the advice repeatedly, our participants appear not to be able to assess the quality of the advising algorithm.

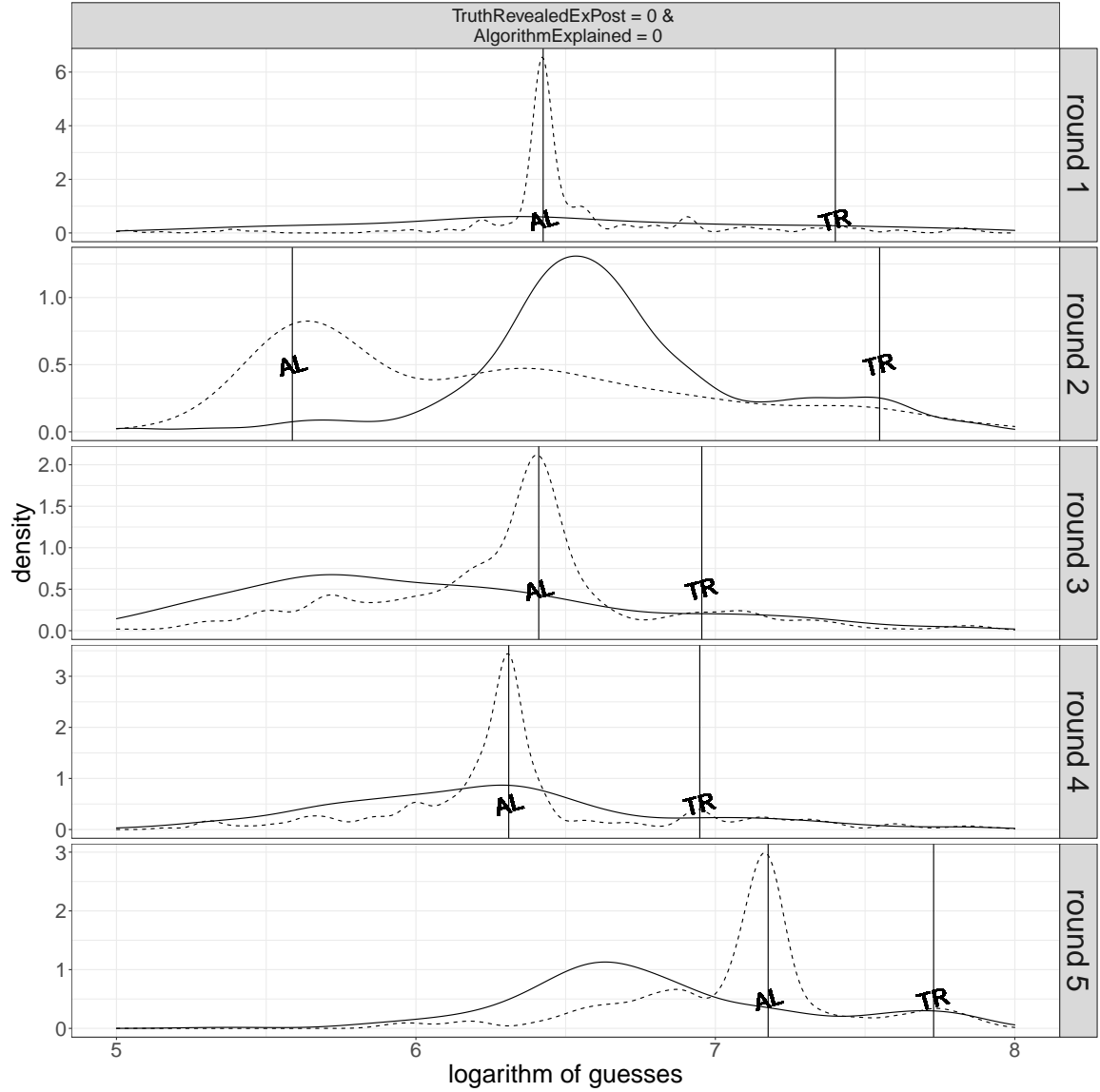
Consequently, we document as our first result:

Result 1: *The human decision makers perceive the algorithmic advice as a credence good.*

The fact that decision-makers cannot reliably ascertain the quality of the advising algorithm entails two major problems: First, it challenges the idea that humans can judiciously oversee algorithmic decisions to prevent harm in high-stake decisions. Second, markets for credence goods are prone to economic inefficiencies. We will discuss these implications further in section 5. Knowing these negative traits of credence goods, we ask how one can improve decision-making in this situation. More specifically, we are interested in whether immediate feedback (revealing the correct answer at the end of the round) and an explanation of the mechanics of the algorithm can help subjects learn from the interaction with the algorithm.

In our setting, ideally, one of our two treatment dimensions (explaining the algorithm or revealing the truth *ex post*) would enable participants to simultaneously lower

Figure 3: Densities of initial and revised guesses by treatment for the first four rounds



Notes: Initial (black line) and revised (dotted line) guess densities for $\text{TRUTHREVEALEDEXPOST} = 0$ & $\text{ALGORITHMEXPLAINED} = 0$ for the first five rounds. Algorithm recommendation (AR) is the leftmost vertical line and the true number of dots (TR) is indicated by the rightmost vertical line. The figure only shows the range of $\log(\text{guess})$ from 5 to 8 and therefore does not display the tails of the distributions. The range of the axis showing the density differs between rounds.

algorithm adherence (i.e., moving further away from our biased algorithm) because they understand the bias and improve their guessing performance (i.e. moving closer to the true number of dots) due to this better understanding. First, we examine the effects of providing an explanation and revealing the truth *ex post* on algorithm adherence. We do so by looking at the distance between the average revised guess and the algorithm for each treatment. The bar graph in figure 4a allows us to compare algorithm adherence by treatment. Table 2 in the appendix contains more specific numeric information regarding algorithm adherence. Figure 4a illustrates that feedback reduces algorithm adherence compared to the baseline treatment. The same is true for explanations with an even stronger effect. In treatment $\text{TRUTHREVEALEDExPOST} = 1 \ \& \ \text{ALGORITHMEXPLAINED} = 1$ algorithm adherence is also the most reduced.

Result 2a: *Explanation reduces algorithm adherence (weakest effect).*

Result 2b: *Revealing the truth reduces algorithm adherence (medium effect).*

Result 2c: *Combining explanation and revealing truth reduces algorithm adherence (strongest effect).*

Knowing that both interventions reduce algorithm adherence, we now turn to the question of how the treatments influence guessing performance. We hence examine the distance between the revised guess and the correct answer. We present the results in figure 4 and table 3 in the appendix. Figure 4 reveals that compared to the baseline treatment $\text{TRUTHREVEALEDExPOST} = 0 \ \& \ \text{ALGORITHMEXPLAINED} = 0$ providing the explanation increases the average distance to the true number of dots, i.e. the explanation makes participants perform worse.

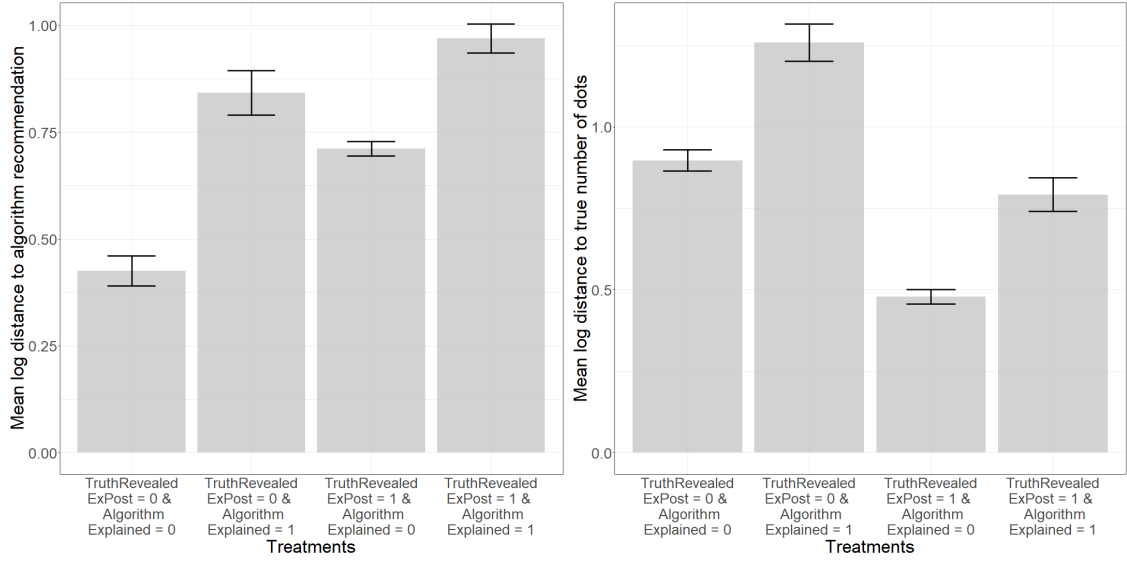
In contrast, revealing the truth improves guessing performance. Remarkably, the effects of these two treatments have approximately the same size (and opposite directions). As a result, in treatment $\text{TRUTHREVEALEDExPOST} = 1 \ \& \ \text{ALGORITHMEXPLAINED} = 1$, the two effects appear to cancel each other out. Hence, the average performance under $\text{TRUTHREVEALEDExPOST} = 1 \ \& \ \text{ALGORITHMEXPLAINED} = 1$ is statistically indistinguishable from $\text{TRUTHREVEALEDExPOST} = 0 \ \& \ \text{ALGORITHMEXPLAINED} = 0$ (t-test is not significant on a 5%-level). The effects on guessing performance can be summarized as:

Result 3a: *Explanation hurts performance.*

Result 3b: *Feedback improves performance.*

Result 3c: *When providing participants with both explanation and feedback, the two aspects neutralize each other and performance remains unchanged.*

Figure 4: Mean distance to the algorithm and the true number of dots by treatment



(a) Mean distance to the algorithm recommendation by treatment. (b) Mean distance to the true number of dots by treatment.

Notes: The bar graph in panel (a) illustrates the treatment effects on algorithm adherence (mean distance of the revised guesses to the algorithm recommendation per treatment). The numerical treatment effects on algorithm adherence can be found in table 2 in the appendix. The bar graph in panel (b) illustrates the treatment effects on guessing performance (mean distance of the revised guesses to the true number of dots per treatment). The numerical treatment effects on guessing performance can be found in table 3 in the appendix. The barplots also include the standard errors around the mean. We pre-process the data by taking log values and calculating the mean over all 16 rounds for each individual. For more details see section 3.

5 Discussion

Participants in the baseline treatment $\text{TRUTHREVEALEDExPOST} = 0$ & $\text{ALGORITHMEXPLAINED} = 0$ repeatedly see biased advice in a task that does not require expert knowledge. Akin to many real-life decision situations, in this treatment, they therefore can compare their own prediction (note that we prime participants by specifically eliciting their initial guesses before seeing the algorithm) with the algorithm prediction. In principle, this could allow any participant to realize that the algorithm is biased and produces dot predictions that are too low. The question is how many participants realize this and if they can correctly adjust for this bias.

From the written feedback that participants could state at the end of the experiment, we know that some participants in the baseline treatment indeed recognize that the algorithm is downward biased (e.g. “I thought the algorithm consistently underestimated the number of dots”, “I did not trust the algorithm. It seemed to be generating numbers that were too low.”), yet others fail to assess the existence, magnitude, or direction of the bias (e.g. “I quickly began to depend on the algorithm and as the study progressed, got close to guessing what the algorithm predicted”, “I figured that it was overestimating”). Ultimately, the question about the nature of the algorithm as a good is an empirical one: How do the majority of participants assess the algorithm? As described in the previous section, most participants follow the algorithm closely and never realize that they could improve their payoff if they move away from a biased recommendation.

The fact that algorithms can be perceived as credence goods has, broadly speaking, implications for two areas: policy initiatives regulating algorithmic systems and managerial practice. Attempts at regulating algorithms rely—among other measures—on the idea of “human oversight”: algorithmic systems should be designed and developed in such a way that natural persons can recognize biased decisions to adjust or overrule the algorithm (e.g. Article 14 of the proposed Artificial Intelligence Act, European Commission, 2021). Entertaining the assumption that humans are in principle able to provide oversight, entails implicitly several assumptions: Humans need to be able to recognize that a bias exists, they need to be able to correctly assess the magnitude and direction of the bias, and finally they need to be able to judge their performance and biases in relation to the algorithm. For example, a physician using a medical image recognition algorithm might notice irregularities around race and gender, but be unable to determine which group is negatively affected or be unable to quantify the extent of negative effects.

Managerial practice is equally affected. Even if decisions in non-high-stake situations do not require regulation, it is still of interest whether humans can provide oversight, as it is in an organization’s best interest to optimize algorithm-advised decisions. Moreover, in markets for credence goods inefficiencies often arise due to an information asymmetry between consumers and producers. Typically, producers can exploit their expert knowledge to the consumers’ detriment. In this paper we do not make any claims about whether producers of algorithmic advice software are aware of its quality, as this is a different

empirical question and it does not affect the central implication: Organizations cannot rely on their individual decision-makers to provide high-quality feedback about the performance of an advising algorithm. For example, when a manager makes pricing decisions and receives advice from an algorithm, they might recognize that the algorithm recommendation is biased. Yet it still remains an open question if the price manager can suggest a superior (in the sense of profit-maximizing) price.

In our experiment, the best way to compare human and algorithm performance is to compare initial guesses in the first round with the algorithm prediction, as only the first-round initial guesses have been stated entirely without the influence of the algorithm. In later rounds, the algorithm recommendation serves as a strong orientation point, thereby influencing human performance. In the first round, the logarithm of the true number is 7.4, while the algorithmic advice is 6.4 and the average guess is 6.0. This suggests that the algorithm is somewhat better than the average participant in the initial round. However, there is a sizable proportion of people who have performed better as well as worse than the algorithm. This evidence shows that there is heterogeneity in whether more or less algorithm adherence is better for performance. Despite the fact that we introduced a strong bias in the algorithm prediction, some participants would still benefit from following the algorithm more closely. This illustrates that increased algorithm adherence can both hurt or improve performance. At any rate, we know that the participants whose initial guess was above the algorithm and below the true value in treatment `TRUTHREVEALEDExPOST = 0 & ALGORITHMEXPLAINED = 0` fail to recognize its bias because in all rounds they revise their initial guesses closer to the algorithm (and thereby further away from the true value).

We now turn to a discussion of the results from treatment `TRUTHREVEALEDExPOST = 0 & ALGORITHMEXPLAINED = 1`. Compared to the baseline treatment, providing an explanation increases the participants' average distance to the algorithm. This suggests that participants correctly infer from the explanation that the algorithm is biased. At the same time explanation *hurts* performance. This is in line with the superior algorithm performance mentioned in the previous paragraph. It also nicely illustrates that the ability to recognize bias and the ability to correctly address this bias are separate skills. This result can also be explained in light of the findings provided by Dietvorst, Simmons, and Massey (2015). The authors show that humans lose trust in algorithms when they see them making mistakes. Similar behavior is likely in our setting: Our explanation illustrates that the algorithm is biased, participants recognize this and follow the advice less. However, many are overconfident with regard to their own performance. They fail to realize that while the algorithm is far from perfect, it would still be optimal for them to follow its advice to some extent.

In treatment `TRUTHREVEALEDExPOST = 1 & ALGORITHMEXPLAINED = 0` participants move further away from the algorithm, but in this treatment, they also move closer to the true number of dots. One potential mechanism of how subjects arrive at their guess is the following: They start with an *orientation point* and adjust this amount of

dots based on their judgment. Examining figures 5 to 8 (in the appendix) provides some evidence that this is indeed an important mechanism. The distribution of initial guesses is very flat in the first round. In addition, participants strongly react to the algorithmic advice. The specific number of dots recommended by the algorithm gives the participants a potential orientation point to use.

The algorithmic advice is an orientation point that is always available when revising the guess. Importantly, in the treatments where $\text{REVEALTRUTHExPOST} = 1$ an alternative orientation point is provided: The true answer from the previous round(s). Therefore revealing the true answer does not only offer the opportunity to better assess the quality of the algorithmic advice. It also provides an alternative orientation point. Further, it can also influence participants through a third channel: Providing feedback about their own abilities.

The concept of the orientation point might be considered particularly important after realizing that figures 5 to 8 suggest that participants are remarkably often very close to the true answer in treatments where $\text{TRUTHREVEALEDEXPOST} = 1$. It appears that participants are strong in taking the true number of dots as an orientation point, estimating the relative change of dots in the next round, and delivering a good estimate for the current round.

The results from treatment $\text{TRUTHREVEALEDEXPOST} = 1$ & $\text{ALGORITHMEXPLAINED} = 0$ suggest that participants do not have a better understanding of the algorithm quality. Nonetheless, their performance improves due to the availability of a better orientation point. The practical implication of this is that whenever the decision environment does not fundamentally change (including the task, the algorithm, and the human decision-makers) revealing the solution as soon as it is available can help improve decisions. This will not always be possible. But in cases where it is, it seems appropriate to consider.

Treatment $\text{TRUTHREVEALEDEXPOST} = 1$ & $\text{ALGORITHMEXPLAINED} = 1$ shows an interesting cumulative result: Since providing an explanation and revealing the truth both increase the distance to the algorithm, their combination does as well. And as these treatments individually have opposed effects on guessing performance, their combination seems to annihilate any effect. Guessing performance remains unchanged compared to the baseline. This illustrates that practitioners must be careful when considering tools to improve decision-making. Generally, one cannot simply assume the more help for the decision-maker, the better.

Finally, another important aspect is whether our interventions show their effect immediately or some repeated interaction is required for the effect to unfold. Overall, there appears to be no pattern that evolves. The interventions do not seem to require warm-up time. One exception is the first round of $\text{TRUTHREVEALEDEXPOST} = 0$ & $\text{ALGORITHMEXPLAINED} = 1$. Participants move closer to the algorithm in the first two rounds after receiving the advice. In round three, the average distance remains the same. Starting from round four, participants always move further away from the recommendation (note that the t-test indicates that the differences are often insignificant). This suggests that

participants must see the explanation multiple times before the effect starts unfolding.

Some important caveats are in order. We do not claim that all algorithms are always perceived as credence goods. We merely point out that algorithms can be perceived as credence goods in many applications. Clearly, one important determinant for this is how the human and algorithm abilities compare. Our results and implications could be considered in scenarios where neither human nor algorithm is obviously better, but where there is ambiguity about performance comparison. Moreover, the nature of our task is rather mathematical and objective. Humans might react very differently when the task nature is more subjective (Castelo, Bos, and Lehmann, 2019).

In this section, we have stated the assumption that individual humans can successfully provide oversight for algorithms is flawed and discuss the implications of this. None of the two interventions that we tested can fully remedy this problem. A natural adjustment is, therefore, to put less emphasis on *individual* oversight and instead shift the focus to *collective* oversight. For example, organizations could audit algorithmic advice systems. This could entail checking possible training data, systematically challenging the algorithm or controlled human-subject field experiments before deployment.⁷

6 Conclusion

We design an experiment in which subjects are asked to guess the number of dots they see in an image while receiving advice from a (biased) algorithm. We find strong evidence that our participants perceive the algorithmic advice to be a credence good: Even after “consumption” of the algorithm advice (i.e., observing the algorithm advice repeatedly), our participants have difficulties assessing its quality correctly. This can lead to a variety of problems with managerial implications for the use of algorithm advice in organizations: If individual decision-makers cannot generally be expected to correctly infer the quality of an advising algorithm, the idea that humans can improve and correct an algorithm through oversight is flawed.

And yet, currently, human oversight is of great importance in many recent policy initiatives to ensure unbiased algorithmic decisions and avoid disparate impact. Certainly, the feasibility and efficacy of human oversight as a means to ensure unbiased decisions depend on various aspects, including the task and the decision context. Still, our work provides evidence in a reasonable setting showing that the idea of people being able to learn and oversee algorithms is fundamentally problematic.

We also test a set of interventions - providing an explanation of the algorithm, revealing the solution *ex post*, or both - and ask whether they can reduce adherence to a biased algorithm and improve decision performance. We find that while the explanation decreases participants’ adherence to a biased algorithm, it hurts their performance. Importantly, subjects guess on average worse than the biased algorithm. This points to an interesting

⁷This idea is similar to Green (2022), which suggests “institutional oversight”.

effect: decision-makers correctly identifying a bias does not guarantee better decisions. What matters is the relation between the human’s and the algorithm’s decision ability, whether humans can correctly assess this relationship and the magnitude and direction of the algorithm bias. This finding is in line with the results from the literature on algorithm explainability: Its ability to improve decisions also depends on various factors, and explanations are no panacea.

Finally, we find that only revealing the truth *ex post* decreases algorithm adherence and improves decision performance. One should be careful to attribute this solely to participants learning about the algorithm’s quality. This treatment also provides feedback on their own performance as well as a benchmark for calibrating future guesses.

Several practical recommendations are suggested from these findings. First, in organizational settings, if concrete feedback about previous decisions can be provided and the decision environment does not fundamentally change, managers should seek to disclose the outcome of past decisions. Second, our study indicates that explanations concerning how an algorithm functions must be provided with caution, as they do not necessarily improve human assessment of algorithm quality. Third, given the difficulty that individuals have in assessing algorithm performance, mechanisms focusing on individual oversight are likely insufficient. Instead, one conceivable path forward for policymakers and managers could be to focus less on *individual* decision makers to prevent harm and optimize decisions but instead to lean more heavily on what could be called *collective* oversight. This could involve conducting algorithm audits in which training data could be tested for representativeness and timeliness or the algorithm could be systematically challenged to identify undesired predictions.

References

- Alufaisan, Yasmeen, Laura R Marusich, Jonathan Z Bakdash, Yan Zhou, and Murat Kantarcioglu. 2021. “Does explainable artificial intelligence improve human decision-making?” In *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35. 6618–6626.
- Balafoutas, Loukas and Rudolf Kerschbamer. 2020. “Credence goods in the literature: What the past fifteen years have taught us about fraud, incentives, and the role of institutions.” *Journal of Behavioral and Experimental Finance* 26:100285.
- Castelo, Noah, Maarten W Bos, and Donald R Lehmann. 2019. “Task-dependent algorithm aversion.” *Journal of Marketing Research* 56 (5):809–825.
- Chen, Daniel L., Martin Schonger, and Chris Wickens. 2016. “oTree—An open-source platform for laboratory, online, and field experiments.” *Journal of Behavioral and Experimental Finance* 9:88–97.
- Darby, Michael R and Edi Karni. 1973. “Free competition and the optimal amount of fraud.” *The Journal of law and economics* 16 (1):67–88.
- Dietvorst, Berkeley J., Joseph P. Simmons, and Cade Massey. 2015. “Algorithm aversion: people erroneously avoid algorithms after seeing them err.” *Journal of Experimental Psychology: General* 144 (1):114.
- . 2018. “Overcoming algorithm aversion: People will use imperfect algorithms if they can (even slightly) modify them.” *Management Science* 64 (3):1155–1170.
- European Commission. 2021. “Proposal for Artificial Intelligence Act.” URL <https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=celex%3A52021PC0206>.
- Galton, Francis. 1907. “Vox populi.” *Nature* 75 (7):450–451.
- Glaeser, Edward L, Andrew Hillis, Hyunjin Kim, Scott Duke Kominers, and Michael Luca. 2021. “Decision authority and the returns to algorithms.” Tech. rep., Harvard Business School Working Paper.
- Green, Ben. 2022. “The flaws of policies requiring human oversight of government algorithms.” *Computer Law & Security Review* 45:105681.
- Green, Ben and Yiling Chen. 2019. “The principles and limits of algorithm-in-the-loop decision making.” *Proceedings of the ACM on Human-Computer Interaction* 3:1–24.
- Lai, Vivian, Chacha Chen, Q Vera Liao, Alison Smith-Renner, and Chenhao Tan. 2021. “Towards a science of human-ai decision making: a survey of empirical studies.” *arXiv preprint arXiv:2112.11471* .

- Logg, Jennifer M., Julia A. Minson, and Don A. Moore. 2019. “Algorithm appreciation: People prefer algorithmic to human judgment.” *Organizational Behavior and Human Decision Processes* 151:90–103.
- Park, Joon Sung, Rick Barber, Alex Kirlik, and Karrie Karahalios. 2019. “A slow algorithm improves users’ assessments of the algorithm’s accuracy.” *Proceedings of the ACM on Human-Computer Interaction* 3 (CSCW):1–15.
- Yin, Ming, Vaughan Wortman, Jennifer Wortman, and Hanna Wallach. 2019. “Understanding the effect of accuracy on trust in machine learning models.” In *Proceedings of the 2019 chi conference on human factors in computing systems*. 1–12.
- Zhang, Yunfeng, Q Vera Liao, and Rachel KE Bellamy. 2020. “Effect of confidence and explanation on accuracy and trust calibration in AI-assisted decision making.” In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*. 295–305.

A Tables and figures

Table 2: Treatment effect on algorithm adherence: Log-distance to algorithm recommendation: Overview

Treatment	n	min	mean	max	std. err.
TruthRevealedExPost = 0 & AlgorithmExplained = 0	324	0.001	0.426	9.514	0.035
TruthRevealedExPost = 0 & AlgorithmExplained = 1	314	0.001	0.842	5.188	0.052
TruthRevealedExPost = 1 & AlgorithmExplained = 0	312	0.001	0.711	1.989	0.017
TruthRevealedExPost = 1 & AlgorithmExplained = 1	313	0.001	0.969	4.271	0.034

Notes: The values in this table refer to the barplot in panel (a) of figure 4a.
“Std. err.” refers to the standard error of the mean.

Table 3: Treatment effect on guessing performance: Log-distance to true number of dots: Overview

Treatment	n	min	mean	max	std. err.
TruthRevealedExPost = 0 & AlgorithmExplained = 0	324	0.001	0.897	9.122	0.033
TruthRevealedExPost = 0					

& AlgorithmExplained = 1	314	0.001	1.258	4.678	0.057
TruthRevealedExPost = 1					
& AlgorithmExplained = 0	312	0.001	0.479	2.025	0.022
TruthRevealedExPost = 1					
& AlgorithmExplained = 1	313	0.001	0.792	5.087	0.052

Notes: The values in this table refer to the barplot in panel (b) in figure 4.
“Std. err.” refers to the standard error of the mean.

Table 4: Distances to algorithmic recommendation per round: TRUTHREVEALED-EXPOST = 0 & ALGORITHMEXPLAINED = 1

Round	$ log(algo) - log(guess_i) $		$ log(algo) - log(guess_r) $		p-values	
	mean	sd	mean	sd	t-test	levене
1	1.45	1.95	1.26	1.89	0.14	0.58
2	1.44	1.19	1.24	1.16	0.00	0.04
3	0.82	1.12	0.84	1.17	0.77	0.04
4	0.65	0.93	0.79	1.14	0.03	0.00
5	0.75	0.94	0.91	1.46	0.04	0.00
6	0.66	0.99	0.80	1.25	0.06	0.00
7	0.63	0.98	0.76	1.30	0.05	0.00
8	0.62	0.77	0.70	1.02	0.19	0.00
9	0.60	0.69	0.78	1.20	0.00	0.00
10	0.70	1.02	0.79	1.12	0.20	0.00
11	0.58	0.68	0.75	1.13	0.00	0.00
12	0.56	0.82	0.75	1.21	0.00	0.00
13	0.60	0.95	0.73	1.12	0.04	0.00
14	0.63	0.66	0.69	1.02	0.34	0.00
15	0.59	0.87	0.82	1.33	0.00	0.00
16	0.81	0.81	0.86	1.01	0.32	0.00

Notes: The structure of the table is the same as in table 1, but the values refer to TRUTHREVEALED-EXPOST = 0 & ALGORITHMEXPLAINED = 1.

Table 5: Distances to algorithmic recommendation per round: TRUTHREVEALED-EXPOST = 1 & ALGORITHMEXPLAINED = 0

Round	$ log(algo) - log(guess_i) $		$ log(algo) - log(guess_r) $		p-values	
	mean	sd	mean	sd	t-test	levене
1	1.25	1.76	0.44	1.05	0.00	0.00
2	1.71	0.72	1.52	0.94	0.00	0.00
3	0.71	0.86	0.68	0.87	0.58	0.50

4	0.80	0.87	0.72	0.75	0.14	0.73
5	0.52	0.35	0.59	0.99	0.23	0.01
6	0.82	0.89	0.77	0.82	0.20	0.94
7	0.56	0.66	0.51	0.64	0.36	0.99
8	0.76	0.74	0.63	0.52	0.00	0.15
9	0.85	0.75	0.71	0.42	0.00	0.13
10	0.86	0.77	0.75	0.71	0.06	0.93
11	0.75	0.60	0.69	0.51	0.11	0.74
12	0.64	0.70	0.61	0.69	0.47	0.86
13	0.76	0.58	0.70	0.60	0.06	0.58
14	0.42	0.37	0.45	0.69	0.39	0.11
15	0.65	0.51	0.59	0.50	0.00	0.70
16	1.07	0.54	1.01	0.56	0.12	0.61

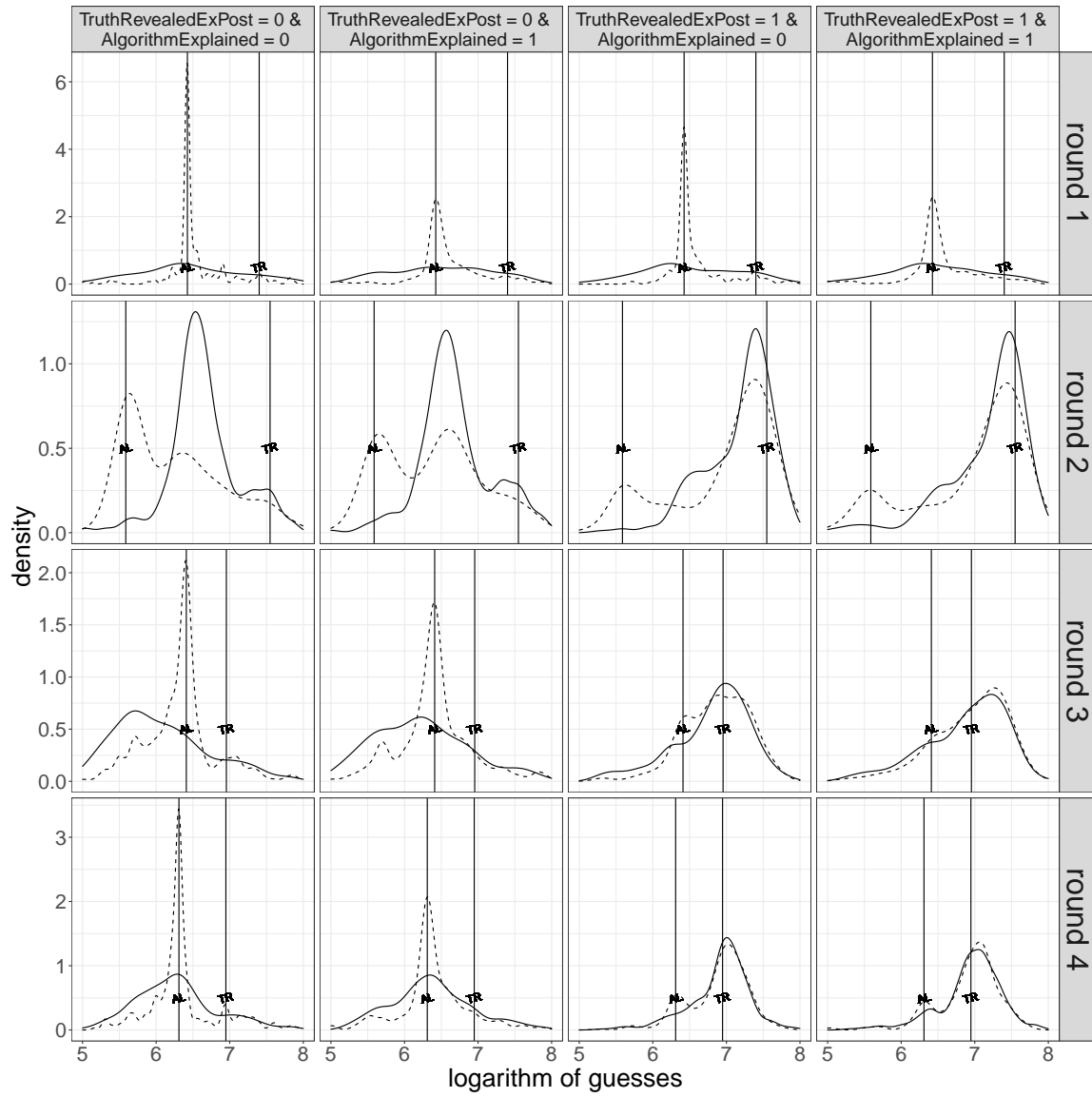
Notes: The structure of the table is the same as in table 1, but the values refer to TRUTHREVEALEDExPOST = 1 & ALGORITHMEXPLAINED = 0.

Table 6: Distances to algorithmic recommendation per round: TRUTHREVEALEDExPOST = 1 & ALGORITHMEXPLAINED = 1

Round	$ log(algo) - log(guess_i) $		$ log(algo) - log(guess_r) $		p-values	
	mean	sd	mean	sd	t-test	levене
1	1.23	1.86	1.25	1.89	0.90	0.14
2	1.74	0.71	1.64	0.96	0.09	0.00
3	0.81	0.95	0.92	0.99	0.06	0.19
4	0.84	0.82	0.91	0.90	0.19	0.19
5	0.67	1.02	0.89	1.35	0.00	0.01
6	0.85	0.84	0.95	0.91	0.07	0.12
7	0.64	0.85	0.74	0.91	0.15	0.12
8	0.79	0.87	0.91	1.05	0.06	0.08
9	0.89	0.79	0.96	0.85	0.15	0.16
10	0.89	0.71	1.07	1.26	0.01	0.00
11	0.86	0.79	0.93	0.92	0.16	0.10
12	0.73	0.82	0.85	0.89	0.04	0.16
13	0.88	0.90	0.91	0.88	0.61	0.48
14	0.51	0.67	0.59	0.70	0.08	0.11
15	0.75	0.80	0.84	0.90	0.12	0.05
16	1.10	0.63	1.14	0.68	0.41	0.04

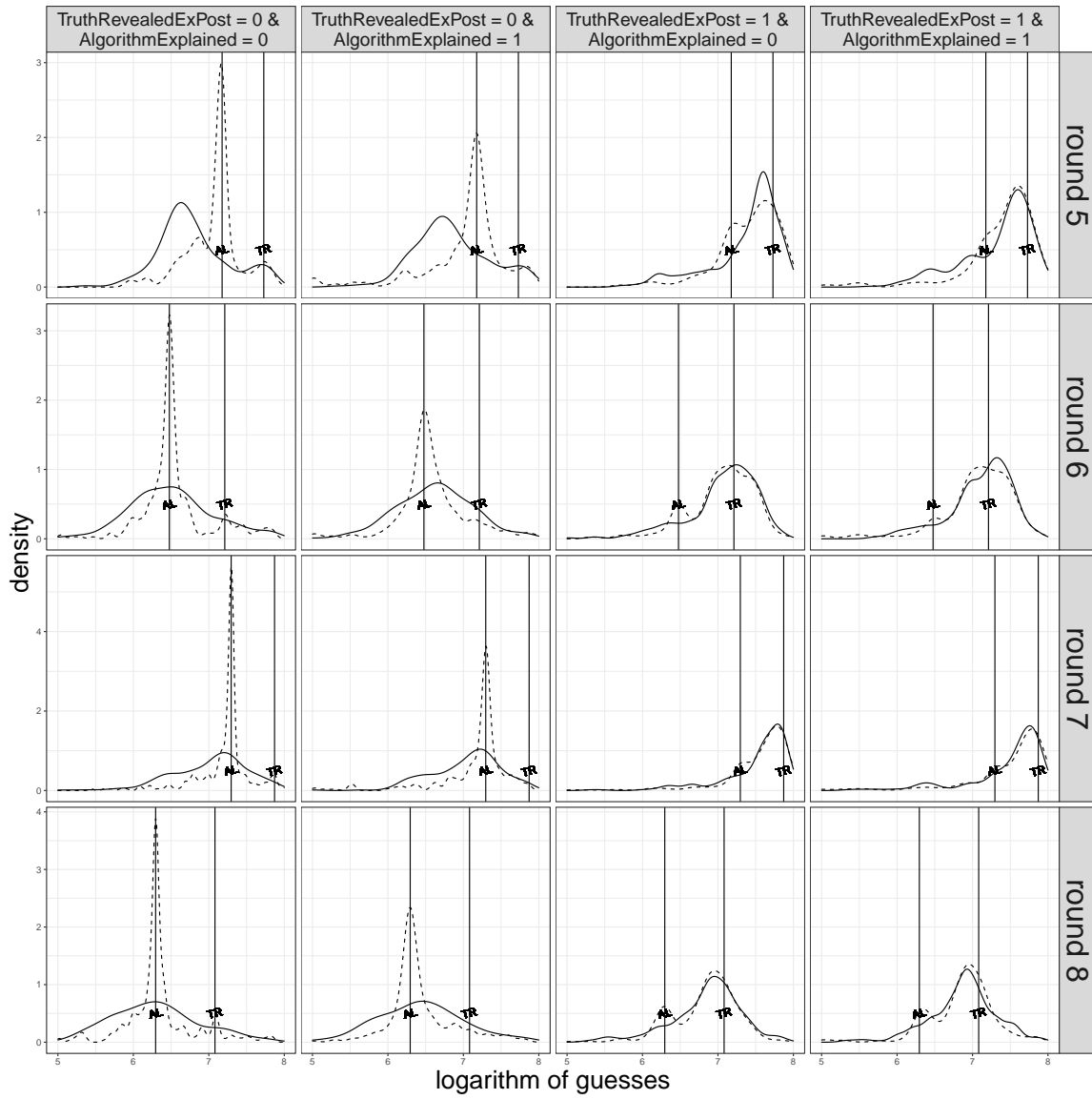
Notes: The structure of the table is the same as in table 1, but the values refer to TRUTHREVEALEDExPOST = 1 & ALGORITHMEXPLAINED = 1.

Figure 5: Distribution of initial and revised guesses by treatment for rounds 1 to 4



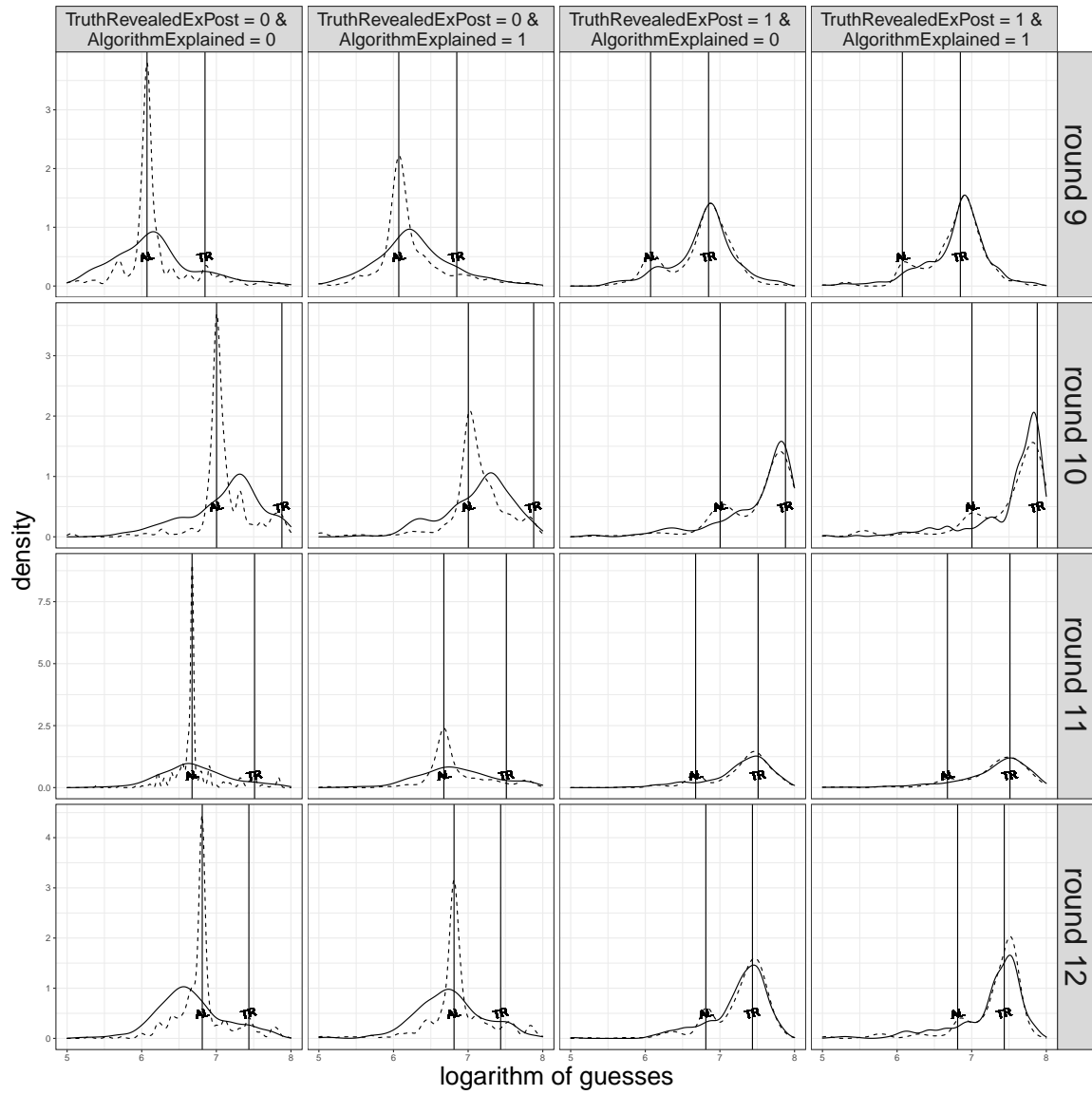
Notes: Initial and revised guess densities for round 1 to 4.

Figure 6: Distribution of initial and revised guesses by treatment for rounds 5 to 8



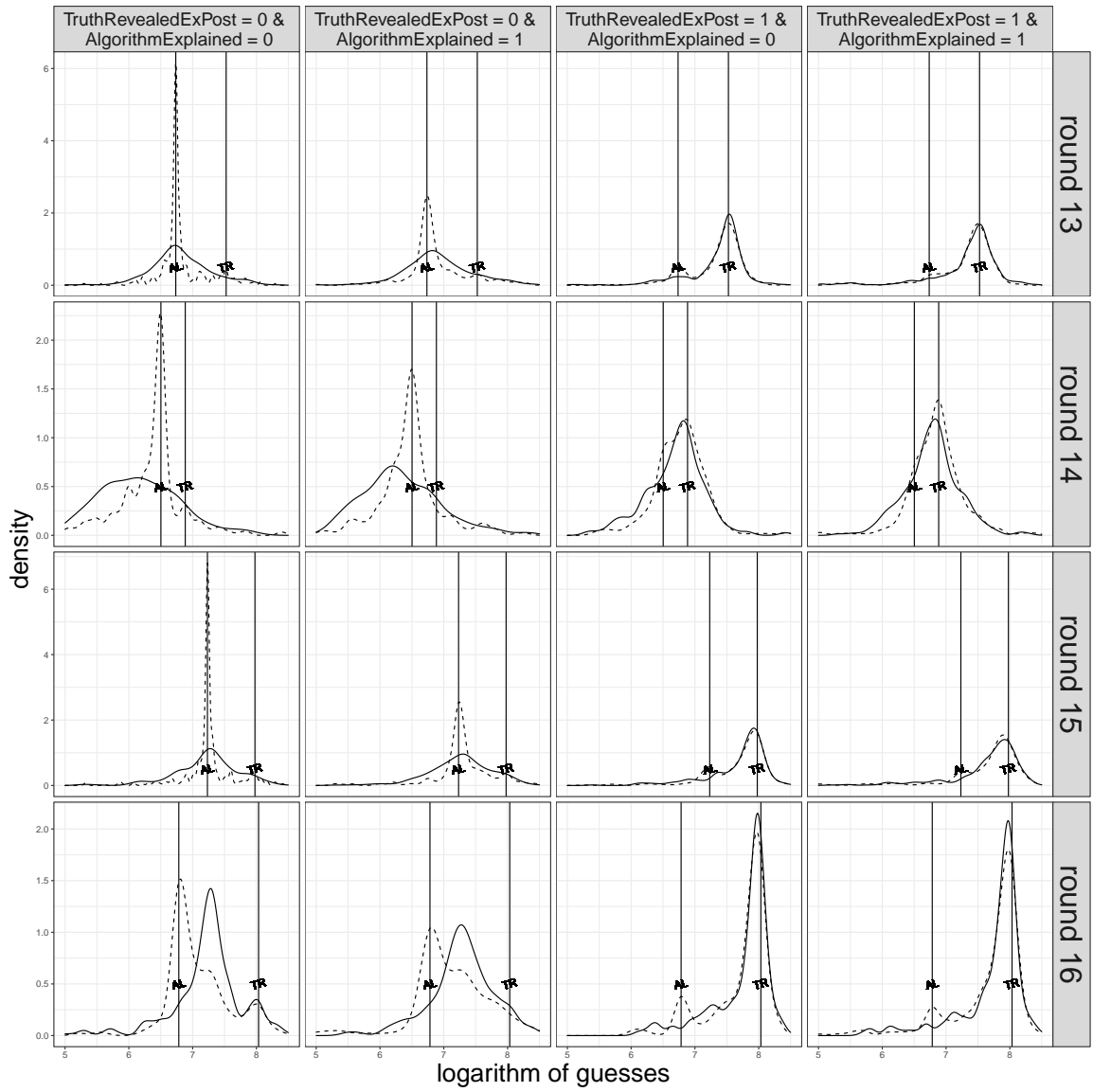
Notes: Initial and revised guess densities for round 5 to 8.

Figure 7: Distribution of initial and revised guesses by treatment for rounds 9 to 12



Notes: Initial and revised guess densities for round 9 to 12.

Figure 8: Distribution of initial and revised guesses by treatment for rounds 13 to 16



Notes: Initial and revised guess densities for round 13 to 16.

B Prediction Algorithm

The algorithm used to provide recommendations for participants is intended to be as simple as possible, while also allowing for easy experimental control. Each dot image is a measuring 100×100 units. The algorithm randomly samples three areas of size 20×20 , as shown in panel (b) in figure 2. Importantly, it only samples from the fringes of each image, never from the center. It then counts the number of dots in each of the three areas separately and calculates the average of these three areas. This average is then multiplied by 25, as the entire image is covered by 5×5 subareas. The procedure is described in algorithm 1.

Algorithm 1 Dot guessing algorithm

$$\begin{aligned} sum_1 &\leftarrow \text{numberofdotsinarea}_1 \\ sum_2 &\leftarrow \text{numberofdotsinarea}_2 \\ sum_3 &\leftarrow \text{numberofdotsinarea}_3 \end{aligned}$$
$$\begin{aligned} Average &\leftarrow \frac{s_1+s_2+s_3}{3} \\ Prediction &\leftarrow Average \times 25 \end{aligned}$$

Since the dots follow a triangular distribution and the algorithm is restricted to sample from the sparsely populated areas, the algorithm will produce a downward-biased prediction. Participants in treatments where `ALGORITHMEXPLAINED` = 1 see the explanation provided in figure 18. In principle any participant is therefore enabled to draw the conclusion that the algorithm at best suggests a lower bound: the true number of dots will necessarily be larger than the algorithm prediction.

C Experimental Interface

Figure 9

Welcome & Informed Consent

Thank you for participating in this study! The purpose of this study is to explore human decision-making.
This study is anonymous. We will not ask for your name or any information that will make you identifiable.
There is **no deception** in this study. Everything you see or read is true.

The study takes most participants less than 10 minutes to complete.
You will receive a fixed payment of \$0.90 (base reward) for your participation. You will also have the chance to earn up to \$4.80 additional dollars depending on your behavior during the study (bonus rewards).

The risks to participating are no greater than those encountered in everyday life. Your participation in this study is completely voluntary, and you may refuse to participate or withdraw from the study without penalty. Compensation will be awarded upon completion of the entire study. If you have any questions, please contact us via MTurk. Please feel free to print or save a copy of this consent form.

Please tick the following box to be able to continue:
☐ I have read and understood this consent form and wish to participate in this study.

Next

Notes: All participants saw this text as their first page.

Figure 10

Feedback

You're almost done!

We value your feedback! Did you find anything unclear or misleading? Any technical issues? Any other feedback regarding any aspects of the study? Would you like to explain your behavior in the study? (For example, did you trust the algorithm? How did this develop over time?) Let us know!

Feedback

Click below to receive the completion code and finish the study.

Next

Notes: All participants saw this as their final page before exiting the experiment.

Figure 11

Final Result

Your bonus payoff is \$0.00

You have completed the study. Your completion code is MERRY_CHRISTMAS. Please copy this code and return to MTurk.

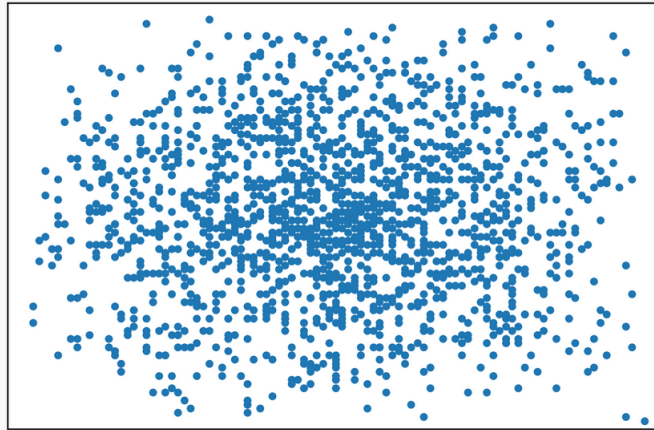
Notes: All participants learned about their final payoff and received a completion code.

Figure 12

Guess: How many dots are in this graph? Round 1/2

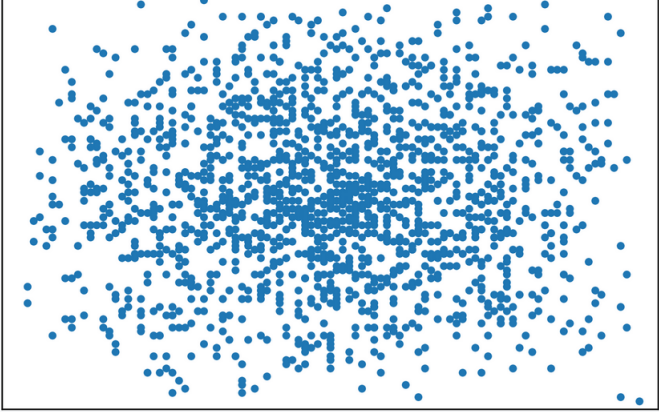
Time left to complete this page: 0:49

Below you see a new image.



Notes: Upper part of initial guess page: All participants saw a this page (with the number of dots changing from round to round) before stating their initial guess. Participants had 60 seconds to state their guess.

Figure 13



Payment info

You will earn additional money for a good guess. You will receive \$0.15 if you guess the correct number. This amount will be reduced by \$0.0002 for every dot by which you deviate from the correct number.

Please enter your guess:

Next

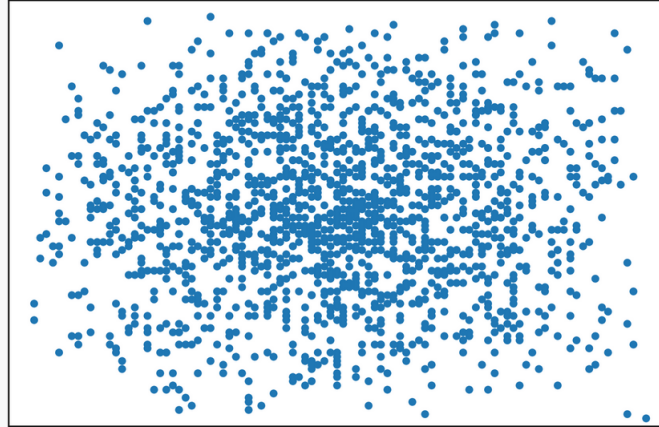
Notes: Lower part of initial guess page: All participants saw this when stating their initial guess.

Figure 14

Machine guess: Round 1/2

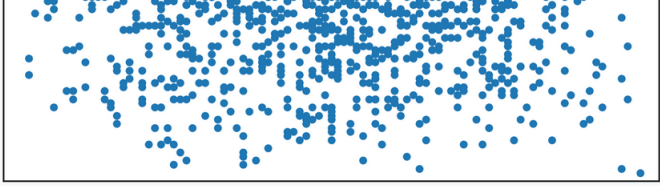
Time left to complete this page: 0:23

Below you see the image from the previous page.



Notes: In treatments where $\text{ALGORITHMEXPLAINED} = 0$, participants saw the same image again before stating their revised guess.

Figure 15



We will now provide you with an algorithmic prediction regarding the number of dots in this picture. This prediction comes from an algorithm that is trying to solve the same problem as you have.

The algorithm predicts that there are 617 dots in this picture.

Your initial guess was 2333.

In light of the new information, you can now modify your guess. Please enter your revised guess below.

[Payment info](#)

You will earn additional money for a good guess. You will receive \$0.15 if you guess the correct number. This amount will be reduced by \$0.0002 for every dot by which you deviate from the correct number.

Please enter your revised guess:

[Next](#)

Notes: All participants saw the algorithm prediction before stating their revised guess.

Figure 16

Result Round 1/2

Your initial guess was 2333.

Your final guess was 333.

This round has ended. Click below to get to the next round.

[Next](#)

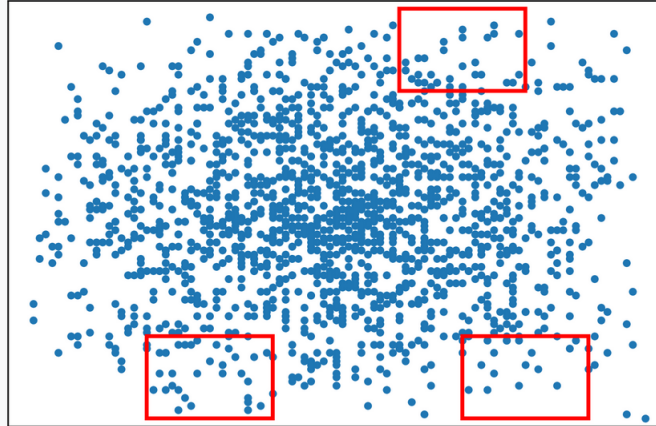
Notes: Participants in treatments where $\text{TRUTHREVEALEDExPOST} = 0$ saw their initial and revised guess again, before moving on to the next round.

Figure 17

Machine guess: Round 1/2


Time left to complete this page: 0:51

Below you see the image from the previous page.



Notes: Participants in treatments where $\text{ALGORITHMEXPLAINED} = 1$ saw the same image again but overlaid with red rectangles indicating the areas from which the algorithm sampled.

Figure 18



We will now provide you with an algorithmic prediction regarding the number of dots in this picture. This prediction comes from an algorithm that is trying to solve the same problem as you have.

The algorithm predicts that there are 617 dots in this picture.

Explanation of algorithm:

1. The algorithm counts the number of dots within each red square.
2. It then calculates the average of dots over the three squares.
3. Finally, it multiplies this average by 25, because 25 squares cover the entire area.

Your initial guess was 33.

In light of the new information, you can now modify your guess. Please enter your revised guess below.

Payment info

You will earn additional money for a good guess. You will receive \$0.15 if you guess the correct number. This amount will be reduced by \$0.0002 for every dot by which you deviate from the correct number.

Please enter your revised guess:

Next

Notes: Participants in treatments where $\text{ALGORITHMEXPLAINED} = 1$ additionally were provided a verbal explanation of the algorithm.

Figure 19

Result Round 1/2

Your initial guess was 22.

Your final guess was 33.

The true number was 1637.

This round has ended. Click below to get to the next round.

Next

Notes: Participants in treatments where $\text{TRUTHREVEALEDPOST} = 1$ saw their initial and revised guess as well as the solution before moving on to the next round.



Download ZEW Discussion Papers:

<https://www.zew.de/en/publications/zew-discussion-papers>

or see:

<https://www.ssrn.com/link/ZEW-Ctr-Euro-Econ-Research.html>

<https://ideas.repec.org/s/zbw/zewdip.html>



IMPRINT

**ZEW – Leibniz-Zentrum für Europäische
Wirtschaftsforschung GmbH Mannheim**

ZEW – Leibniz Centre for European
Economic Research

L 7,1 · 68161 Mannheim · Germany

Phone +49 621 1235-01

info@zew.de · zew.de

Discussion Papers are intended to make results of ZEW research promptly available to other economists in order to encourage discussion and suggestions for revisions. The authors are solely responsible for the contents which do not necessarily represent the opinion of the ZEW.