Explaining Decision-Making Algorithms through UI: Strategies to Help Non-Expert Stakeholders

Hao-Fei Cheng [1], **Ruotong Wang** [2], **Zheng Zhang** [1], **Fiona O'Connell** [3],
Terrance Gray [1], **F. Maxwell Harper** [1], **Haiyi Zhu** [1]
[1]University of Minnesota, [2]Macalester College, [3]University of Chicago
{cheng635, zhan5963, grayx501, max, zhux0449}@umn.edu
rwang2@macalester.edu, fionaoconnell@uchicago.edu

ABSTRACT

Increasingly, algorithms are used to make important decisions across society. However, these algorithms are usually poorly understood, which can reduce transparency and evoke negative emotions. In this research, we seek to learn design principles for explanation interfaces that communicate how decision-making algorithms work, in order to help organizations explain their decisions to stakeholders, or to support users' "right to explanation". We conducted an online experiment where 199 participants used different explanation interfaces to understand an algorithm for making university admissions decisions. We measured users' objective and self-reported understanding of the algorithm. Our results show that both interactive explanations and "white-box" explanations (i.e. that show the inner workings of an algorithm) can improve users' comprehension. Although the interactive approach is more effective at improving comprehension, it comes with a trade-off of taking more time. Surprisingly, we also find that users' trust in algorithmic decisions is not affected by the explanation interface or their level of comprehension of the algorithm.

CCS CONCEPTS

• **Human-centered computing** → **Interactive systems and tools**; **Empirical studies in HCI**;

KEYWORDS

Algorithmic Decision-making, Explanation Interfaces

CHI 2019, May 4–9, 2019, Glasgow, Scotland UK

ACM ISBN 978-1-4503-5970-2/19/05…$15.00
https://doi.org/10.1145/3290605.3300789

1 INTRODUCTION

Automated and artificially intelligent algorithmic systems are helping humans make important decisions in a wide variety of domains. To name a few examples, recidivism risk assessment algorithms such as COMPAS have been used to help judges decide whether defendants should be detained or released while awaiting trial [11, 19]. Allegheny County in Pennsylvania has been using an algorithm based on Predictive Risk Modeling (PRM) to help screen referral calls on child maltreatment [10]. And according to an article in The Wall Street Journal, the proportion of large companies using Applicant Tracking Systems to automatically filter and rank applicants is in the "high 90%" range [52].

Researchers, government bodies, and the media have argued that data users should have the "right to explanation" of all decisions made or supported by automated or artificially intelligent algorithms. The approval in 2016 of the European Union General Data Protection Regulation (GDPR) mandates that data subjects receive meaningful information about the logic involved in automated decision-making systems [51]. However, it is challenging for people who are not algorithm experts to understand algorithmic decision-making systems. Due to this literacy gap, recipients of the algorithm's output have difficulty understanding how or why the inputs lead to a particular outcome [7].

The recent surge of interest in explainable artificial intelligence (XAI) (see [6] for a review) has lead to great progress on transforming complex models (such as neural networks) into simple ones (such as linear models or decision trees) through approximation of the entire model [13, 48] or local approximation [45]. Despite its mathematical rigor, there are recent critiques that this line of research is based on the intuition of researchers, rather than on a deep understanding of actual users [38]. There is limited empirical evidence on whether these "intelligible models" and explanation interfaces are actually understandable, usable, or practical in real-world situations [1, 18]. On the other hand, HCI researchers have conducted surveys, done interviews, and analyzed public tweets to understand how real-world users perceive and adapt to algorithmic systems (e.g. [14, 15, 20, 21]). However, how these empirical findings can guide the design of more explainable algorithms or more effective explanation interfaces remains underexplored. Two recent review papers [1, 26] both suggest bridging these two isolated research areas, by drawing principles and methods from HCI to improve the usability of explanation interfaces and performing empirical studies to evaluate the efficacy of these interfaces.

The goal of this paper is to bridge these different research areas through conducting ***human-centered design*** and ***empirical comparisons*** of parallel interface prototypes to explore the ***effectiveness and trade-offs of different strategies to help non-expert stakeholders understand algorithmic decision making.*** We understand that there might not be a universally effective strategy for algorithmic decision-making. Therefore, we focus on whether there are more effective strategies in the context of ***profiling***, defined as the processing of personal data to evaluate certain aspects relating to a natural person [1][23]. In profiling tasks, the actual evaluation outcomes (e.g. the risk of offenders, or the suitability of applications to an organization or a university) are difficult to observe.

We examined two sets of strategies for designing interfaces to explain algorithmic decision-making: white-box vs. black-box (i.e. showing the internal workings of an algorithm or not), and static vs. interactive (i.e. allowing users to explore an algorithm's behavior through static visualizations or interactive interfaces). We conducted an online experiment where participants used four different explanation interfaces to understand an algorithm for making university admissions decisions. We developed measures to assess participants' objective and self-reported understanding of the algorithm. Our results show that interactive explanations improved both objective and self-reported understanding of the algorithm, while "white-box" explanations only improved users' objective understanding. Although the interactive approach is more effective for comprehension, it requires more of the user's time. Surprisingly, we also found that users' trust in algorithmic decisions was not affected by the explanation interface they are assigned to.

The contributions of our work are three-fold. First, our work provides concrete recommendations for designing effective algorithm explanations. Second, our findings suggest nuanced trade-offs between different explanation strategies. Third, we provide a framework to evaluate algorithmic understanding with end users in a real world application. Future researchers can use and adapt our framework to evaluate algorithmic understanding in other domains. The fundamental goal of our work is to contribute to the ongoing conversation regarding the accountability and transparency of algorithms and artificial intelligence.

2 RELATED WORK AND RESEARCH QUESTIONS

Algorithmic Decision-making

We define "algorithmic decision-making", or simply "algorithm", as the processing of input data to produce a score or a choice that is used to support decisions such as prioritization, classification, association, and filtering [16]. In some settings, algorithmic decision-making systems have been used to completely replace human decisions. But in most real-world scenarios, there is a human operator involved in the final decision, who is influenced by the algorithm's suggestions and nudging [16].

In this paper, we focus on algorithms generated through supervised machine learning-based approaches. The first step is to define a prediction target, often a proxy for the actual evaluation outcome. With reference to the examples cited above, this might consist of whether a defendant will be charged with a crime if released, whether a child will be removed from their home and placed in care, or whether a job applicant will receive a job offer. The second step is to use labeled training data, often in large volumes, to train and validate machine learning models. Finally, validated models are applied to new data from incoming cases in order to generate predictive scores.

Note that in this paper, the goal is to help users and other stakeholders understand the "algorithmic decision model", rather than the process of model training.

Explaining and Visualizing Machine Learning

Applied Machine Learning (ML) and visualization communities have long been working on developing techniques and tools to explain and visualize ML algorithms and models (e.g. [26]). However, there are two challenges in directly applying these techniques to help non-experts understand algorithmic decision-making, particularly profiling.

First, the majority of these techniques and tools are designed to support expert users like data scientists and ML practitioners (e.g. [2, 29, 31, 41, 44]) or serve educational purposes for people who are machine learning novices but often have good technical literacy [46]. For instance, these tools often depend on performance measures (e.g. accuracy, precision, recall, confusion matrices, and area under the ROC curve measures) to help people understand and compare different models; these techniques might not help non-expert users, especially those with low technical literacy.

Second, although there are some studies seeking to help non-expert end-users interpret and interact with ML models, they focus on applications such as image classification (e.g. [8]), translation (e.g. [24]), text mining and text classification

[1] This definition is consistent with the definition of profiling in GDPR. Examples mentioned above, including recidivism risk assessment algorithms, risk modeling algorithms for the maltreatment of children, and job application assessment algorithms, are considered profiling algorithms

(e.g. [33, 36, 47]), and context-aware systems (e.g. [5, 34]). As far as we know, there has been limited work seeking to help end-users understand algorithmic decision-making systems that address social problems, such as profiling algorithms.

Although the specific techniques and tools developed by the visualization community cannot be directly applied to explaining profiling algorithms, some high-level strategies might still be relevant. Specifically, we examine two sets of strategies: a black-box approach versus a white-box approach, and a static approach versus an interactive approach.

White-box vs. Black-box Explanation

There are two distinct approaches for explaining algorithms: the "white-box" approach (i.e. explaining the internal workings of the model) and the "black-box" approach (i.e. explaining how the inputs relate to the outputs without showing the internal workings of the model). Examples of the white-box technique include showing probabilities of the nodes for Bayesian networks [4], projection techniques [9] and Nomograms [28] to see the "cut" in the data points for Support Vector Machines, and the visualization of the graph of a neural network [50]. In contrast to the white-box approach, the black-box approach focuses on explaining the relationships between input and output, regardless of how complicated the model itself is. For example, Krause et al. design an analytics system [31], Prospector, to help data scientists understand how any given feature affects algorithm prediction overall. Plate et al. [43] and Olden [39] propose methods to show how input features influence the outcome of neural network classifications. Martens and Provost [36] show removal-based explanations such as "the classification would change to [name of an alternative class] if the words [list of words] were removed from the document." However, we posit that the relative strengths and weaknesses of white-box and black-box approaches in helping non-expert users understand profiling algorithms remain unestablished. For example, one possible trade-off is that the white-box approach can give users a comprehensive understanding of the model, but might cause information overload and create barriers for users who are not technologically savvy [22].

Research Question 1: *How effective are the white-box and black-box strategies in helping non-expert users understand profiling algorithms?*

Interactive vs. Static Explanation

In practice, most algorithm explanations are static and assume that there is a single message to convey. However, as Abdul et al. [1] suggest in their review paper, an alternative approach would be to "allow users to explore the system's behavior freely through interactive explanations." Interaction can be a powerful means to enable people to iteratively explore, gather insight from large amounts of complex information, and build a deep understanding of an algorithm. Weld and Bansal sketched a vision for an interactive explanation system [53], which should support follow-up questions and drill-down actions from the users, such as redirecting the answer or asking for more details. We posit that the interactive interface for algorithm explanation for non-experts is promising but still relatively underexplored.

Research Question 2: *How effective are the interactive and static interfaces in helping non-expert users understand profiling algorithms?*

Interpersonal Difference

Users who are not algorithm experts may still vary substantially in terms of their education levels and general technical literacy, potentially influencing how effective the explanation interfaces are. Our third research question is:

Research Question 3: *How will users' personal characteristics (i.e. education level and technical literacy) influence the effectiveness of the explanation interface in helping them understand profiling algorithms?*

Relationship between Explanation and Trust

Research has shown that many users and stakeholders distrust algorithmic systems and thus are not willing to use such systems. For instance, studies have shown that even when algorithmic predictions are proved to be more accurate than human predictions, both domain experts and laypeople remain resistant to using the algorithms (e.g. [17]). Therefore, our final research question is as follows:

Research Question 4: *Will explanation interfaces increase users' trust in the profiling algorithms?*

3 METHODS

In our study, we used a mixed-method approach and conducted two main activities: (1) design workshops to create parallel algorithm explanation prototypes using different strategies; and (2) online experiments to evaluate their effectiveness. We began by selecting the task domain, creating a dataset, and developing a machine learning model to be explained in the design and empirical evaluation stages.

Task Domain, Dataset and Model

Task domain. Student admission is a classic profiling task, as university admissions offices increasingly use algorithms to profile and predict students' behaviors [40]. Furthermore, student admission is a domain for which we can recruit a large group of people who have some amount of personal experience. According to a survey conducted by Pew Research Center [25], 51% of Mechanical Turkers have a bachelor's degree or higher. Therefore, student admission is suitable for both small-scale design workshops in the lab and large-scale crowd studies. In the study, we use the task domain of graduate school admission in the US.

Dataset. We created a dataset based on publicly available aggregate statistics of applicants and admitted students for a public university. In other words, we generated the dataset without using any real information from individual students, but is nonetheless consistent with the distribution of the actual applicant population. The dataset contains 100 student profiles, with attributes that admission committees actually consider in the real admission process [2]. We also introduced three "additional attributes" to assess users' comprehension of the algorithms (see "Evaluation Metrics" section for more details). To generate labels for the purpose of training, two authors evaluated the strengths and weaknesses of each individual student profile, and manually classified each profile into one of the four admission decisions (strong reject, weak reject, weak accept, strong accept) based on their experience of student admissions at the university.

Model. We selected a multi-category linear model to explain in our study for several reasons: (i) linear models are widely used in real-world problems (e.g. [42]); (ii) we can approximate complex models like neural networks to simple linear models or generalized additive models (at least locally) [45]; (iii) although linear models are "intelligible models" [53], there is a lack of understanding of whether these models and explanation interfaces are actually usable in real-world scenarios [1, 18], especially when they have a wide range of features. We trained the multi-category linear model with the labeled dataset described above. We used the linear regression model as a decision classifier by discretizing the predicted response into one of the four decision labels. The area under the macro-average ROC curve of the four classes is 0.88.

Design Workshops

We conducted a series of design workshops:

Design Workshop 1: We invited one algorithm expert, two UI designers (current graduate students at a public university), three prospective students interested in applying to the university, two current graduate students, and one faculty member (an HCI researcher who has served on the committee for graduate school admissions at the public university over the last three years) to join the workshop. We introduced the goal of positioning participants as experts in their own right and equalizing the power between researchers and participants. We shared the identified challenges, opportunities, and design choices in algorithm explanation and asked participants to respond, reflect, and critique these thoughts. Then the participants and researchers worked together to make sketches and paper prototypes, informed by different design directions (white-box vs. black-box, and static vs. interactive). After Design Workshop 1, the research team developed medium-fidelity prototypes inspired by the paper sketches generated in Design Workshop 1, which were then used in Design Workshop 2.

Design Workshop 2. We then invited three new participants, who were prospective or current students at the public university, to represent the stakeholders who are affected, directly and indirectly, by the algorithmic decisions. We asked these participants to reflect on their own experiences, use and critique the prototypes, and provide feedback. For instance, one participant pointed out that the interactive interfaces lacked appropriate feedback: it was unclear whether the tool had incorporated changes to the values of attributes or not. To address this issue, we added a loading animation and refreshed the algorithmic decision explicitly every time users made an adjustment, regardless of whether the decision changed. The research team then integrated all the feedback from participants and developed four high-fidelity prototypes.

Explanation Interface Prototypes

We created four interface prototypes (white-box interactive, white-box static, black-box interactive, and black-box static) to explain student admission algorithms (see Figure 1(a)). All versions of the interfaces followed the principles below:

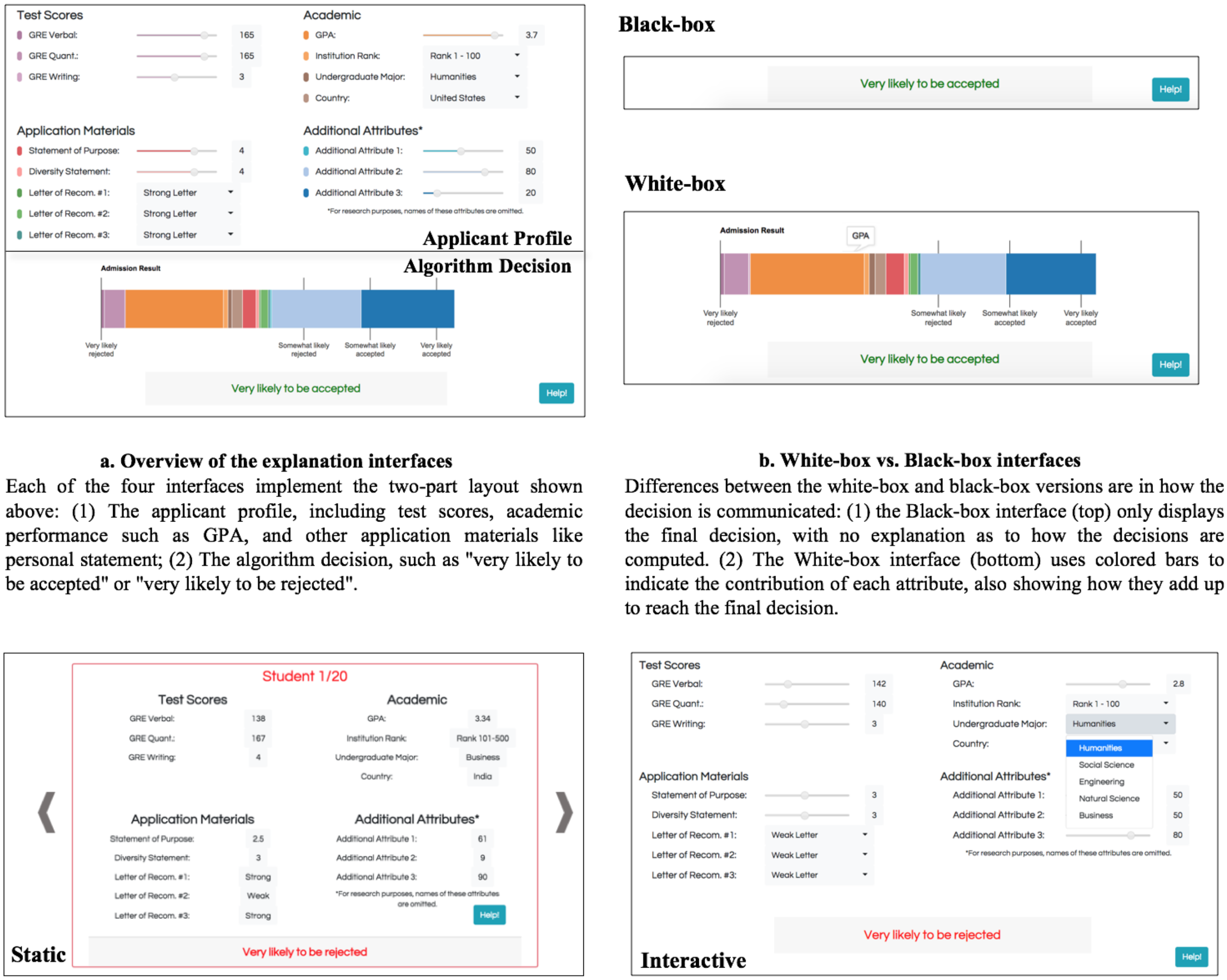
(1) We used a "card"-based design. The interface presents a student's fifteen attributes and corresponding values, as well as the algorithm's decision (i.e. strong accept, weak accept, weak reject, and strong reject). The users could obtain a quick overview of all the information relevant to one student.

(2) We presented the student's attributes in groups. Specifically, we categorized the fifteen attributes into four groups (test scores, academic performance, application materials and additional attributes). Detailed description of the attributes was provided when users hovered over their labels.

Next, we describe how the interface of the tool varies between different explanation strategies.

White-box vs. Black-box. The key difference between the white-box and black-box explanation is *whether the inner workings of the model are visualized or not.* The white-box explanation shows how the algorithmic decisions are computed. As shown in Figure 1 (b), there is a bar chart in the white-box interface illustrating the breakdown of the decision: how the weighted attributes add up to the final output (i.e. the algorithm's recommendation). The influence of each attribute is represented by a distinct color on the bar. The decision boundaries of the adjacent output categories are also labeled on the bar.

[2] The dataset and details of the attributes can be viewed at: https://github.com/flyerfei/algorithm-explanation-chi19



Test Scores
GRE Verbal: 165
GRE Quant.: 165
GRE Writing: 3

Academic
GPA: 3.7
Institution Rank: Rank 1 - 100
Undergraduate Major: Humanities
Country: United States

Application Materials
Statement of Purpose: 4
Diversity Statement: 4
Letter of Recom. #1: Strong Letter
Letter of Recom. #2: Strong Letter
Letter of Recom. #3: Strong Letter

Additional Attributes*
Additional Attribute 1: 50
Additional Attribute 2: 80
Additional Attribute 3: 20
*For research purposes, names of these attributes are omitted.

Very likely to be accepted

Very likely to be accepted

Very likely to be accepted

a. Overview of the explanation interfaces

Each of the four interfaces implement the two-part layout shown above: (1) The applicant profile, including test scores, academic performance such as GPA, and other application materials like personal statement; (2) The algorithm decision, such as "very likely to be accepted" or "very likely to be rejected".

b. White-box vs. Black-box interfaces

Differences between the white-box and black-box versions are in how the decision is communicated: (1) the Black-box interface (top) only displays the final decision, with no explanation as to how the decisions are computed. (2) The White-box interface (bottom) uses colored bars to indicate the contribution of each attribute, also showing how they add up to reach the final decision.

Student 1/20

Test Scores
GRE Verbal: 138
GRE Quant.: 167
GRE Writing: 4

Academic
GPA: 3.34
Institution Rank: Rank 101-500
Undergraduate Major: Business
Country: India

Application Materials
Statement of Purpose: 2.5
Diversity Statement: 3
Letter of Recom. #1: Strong
Letter of Recom. #2: Weak
Letter of Recom. #3: Strong

Additional Attributes*
Additional Attribute 1: 61
Additional Attribute 2: 9
Additional Attribute 3: 90
*For research purposes, names of these attributes are omitted.

Very likely to be rejected

Test Scores
GRE Verbal: 142
GRE Quant.: 140
GRE Writing: 3

Academic
GPA: 2.8
Institution Rank: Rank 1 - 100
Undergraduate Major: Humanities
Country:

Application Materials
Statement of Purpose: 3
Diversity Statement: 3
Letter of Recom. #1: Weak Letter
Letter of Recom. #2: Weak Letter
Letter of Recom. #3: Weak Letter

Additional Attributes*
Additional Attribute 1: 50
Additional Attribute 2: 50
Additional Attribute 3: 80
*For research purposes, names of these attributes are omitted.

Very likely to be rejected

c. Interactive vs. Static interfaces

The Static interface (left) displays a selection of 20 unique application profiles (one at a time) for users to scroll between and analyze. The Interactive interface (right) provides sliders to modify the values of attributes. The tool will re-compute and update the algorithm's decision.

Figure 1: The explanation prototypes designed to communicate how the admission decision-making algorithm works.

Interactive vs. Static. The key difference between the interactive and static explanations is that the interactive interface *allows users to explore the algorithm freely through adjustable inputs*. Figure 1(c) shows a comparison between interactive and static interfaces. In the interactive prototypes, users can change the attribute values and observe how the algorithmic decision changes accordingly. Users gain an understanding of the algorithm by exploring different input combinations. In contrast, the static explanation does not allow users to change students' profiles. Instead, users can browse through a fixed set of profiles by scrolling left and right. We randomly picked 20 student profiles to display in the black-box prototypes, selecting an equal number of profiles from each output category (e.g. strong accept or weak reject).

Evaluation Metrics

Objective Understanding. It has been a challenge in the XAI research community to quantitatively measure "understandability" or "interpretability" [35]. We adapt Weld and Bansal (2018)'s definition that a human user "understands" the algorithm if *the human can see what attributes cause the algorithm's action and can predict how changes in the situation can lead to alternative algorithm predictions.*

Following this principle, we designed three types of quiz questions to assess participants' objective understanding of the algorithm: Unnamed Attributes Questions, Alternative Prediction Questions, and Decision Prediction Questions. Each type of question measures one aspect of users' understanding of the algorithm.

Question Type 1: Unnamed Attributes. We want to measure to what extent users understand the influence of individual attributes on the algorithm's output. However, in any real-world scenarios, users' *pre-existing beliefs* can complicate the assessment of algorithm understandability. For example, if asked how increasing GPA would influence the algorithm's decision, most people would probably say the chance of acceptance would increase. However, we cannot tell whether this answer is based on their understanding of how the algorithm works, or simply on a preexisting assumption that a higher GPA is a good thing. To address this issue, we introduced the idea of "unnamed attributes". We intentionally hid the names of some attributes, and then asked users whether increasing each of these "unnamed attributes" would increase, decrease, or have no impact on an applicant's chance of acceptance.

- An example question in this category: "Considering the following profile: If other attributes remain unchanged, what effect does increasing 'Unnamed Attribute 1' from 40 to 70 have on the algorithm's decision for this applicant?"

Question Type 2: Alternative Prediction. We want to measure whether people can predict how changes in the input profile lead to alternative algorithm predictions. Specifically, we presented a profile and then asked which of the listed changes would give that applicant the best chance of acceptance.

- An example question in this category: "Which change would give an applicant a higher chance of acceptance — Increasing the GRE Verbal score from 150 to 165, or increasing the GRE Quant score from 150 to 165?"

Question Type 3: Decision Prediction. We want to measure people's holistic understanding of the algorithm by asking them to predict the algorithm's actions. We created two types of prediction questions:

- We presented one profile and asked "How would the algorithm categorize this applicant? "
- We presented three different profiles and asked "Which of the following three applicant profiles has the highest chance of being accepted?"

We created 12 objective understanding questions in total (3 unnamed attribute questions, 4 alternative prediction questions, and 5 decision prediction questions).

Self-reported Understanding. We measured the self-reported understanding of participants using a 7-point Likert scale. Participants answered the question "I understand the admission algorithm", on a scale from "Strongly disagree" (1) to "Strongly agree" (7).

Time Cost. We measured how much time participants spent using the tool and answering the objective understanding questions.

Trust. We adapted Corritore et al.'s definition [12] and define trust as the confident expectation that one's vulnerability will not be exploited. We adapted the questions from prior research measuring trust in human-machine systems into a 7-point Likert scale [32]. Details of the questions can be found in the auxiliary material.

Technical Literacy. To evaluate technical literacy, we asked participants to self-report their (1) familiarity with popular applications of computer algorithms (e.g. email spam filter, Amazon recommendation), and (2) programming experience. We adapted these questions from [49, 54].

Algorithm Literacy. To measure users' literacy in algorithm, we asked participants to report their knowledge of computer algorithms. We adapted these questions from [33].

Demographic Information. We asked participants to report their education level, gender, and age. In the analysis, we operationalized education level as whether or not the participants had completed a bachelor's degree.

Open-response questions. As the Likert scale questions alone do not tell us *why* people understand/ trust the algorithm or not, we added a number of open-ended questions to uncover the mechanisms underlying algorithm understanding and trust. We asked participants to discuss how the interface helped their understanding of the algorithm. We also asked participants to explain the reasons why they trust or do not trust the algorithm.

Experimental Design

To evaluate the effectiveness of the four explanation interfaces, we conducted a randomized between-subject experiment on Mechanical Turk (MTurk). We used a 2x2+1 design, resulting in five conditions: white-box interactive, white-box static, black-box interactive, black-box static, and control. In the first four conditions, participants were given access to the explanation interface of the respective condition. They were allowed to spend as much time as they needed to understand the algorithmic decision with help of the interface. In the control condition, participants were only provided a static webpage which displayed the list of attributes considered by the algorithm.

	Objective Understanding			Self-report Understanding			Time Cost			Trust		
	Model 1	Model 2	Model 3	Model 4	Model 5	Model 6	Model 7	Model 8	Model 9	Model 10	Model 11	Model 12
	Coef.	Coef.	Coef.	Coef.	Coef.	Coef.	Coef.	Coef.	Coef.	Coef.	Coef.	Coef.
	(S.E.)	(S.E.)	(S.E.)	(S.E.)	(S.E.)	(S.E.)	(S.E.)	(S.E.)	(S.E.)	(S.E.)	(S.E.)	(S.E.)
Intercept	4.63***	4.88***	5.70***	4.70***	4.82***	2.01**	4.81***	5.29***	6.90	4.46***	4.47***	1.69*
	(.30)	(.22)	(1.30)	(.17)	(.13)	(.69)	(1.15)	(.85)	(4.98)	(.18)	(.13)	(.75)
Whitebox Interactive	2.87***			0.69**			7.07***			0.05		
vs control	(.44)			(.25)			(1.66)			(.26)		
Blackbox Interactive	2.31***			0.64*			5.23**			0.24		
vs control	(.45)			(.26)			(1.73)			(.27)		
Whitebox Static	1.34**			0.30			2.71			0.24		
vs control	(.45)			(.26)			(1.72)			(.27)		
Blackbox Static	0.55			0.28			1.06			0.02		
vs control	(.45)			(.26)			(1.71)			(.27)		
Interactive		2.06***	1.69		0.51*	2.41*		4.75**	12.26		0.24	2.06
vs Static (IvS)		(.41)	(1.86)		(.23)	(1.00)		(1.55)	(7.15)		(.24)	(1.07)
Whitebox		1.09**	2.33		0.18	-1.08		2.24	-3.44		0.23	0.12
vs Blackbox (WvB)		(.40)	(1.87)		(.23)	(1.00)		(1.53)	(7.18)		(.24)	(1.08)
IvS * WvB		-0.54	-0.46		-0.13	-0.28		-0.40	-0.18		-0.43	-0.50
		(.62)	(.62)		(.35)	(.33)		(2.34)	(2.37)		(.37)	(.36)
techLiteracy			-0.31			0.59***			-0.32			0.56***
			(.25)			(.14)			(.97)			(.15)
hasBachelor			1.11*			-0.27			0.04			-0.08
			(.45)			(.24)			(1.72)			(.26)
techLiteracy * IvS			0.19			-0.38			-1.31			-0.37
			(.38)			(.20)			(1.45)			(.22)
hasBachelor * IvS			-0.90			0.09			-1.36			0.08
			(.68)			(.37)			(2.63)			(.39)
techLiteracy * WvB			-0.17			0.18			0.72			-0.03
			(.38)			(.20)			(1.47)			(.22)
hasBachelor * WvB			-0.55			0.56			3.18			0.46
			(.69)			(.37)			(2.63)			(.39)
Adjusted R-Sq	0.21	0.21	0.22	0.03	0.03	0.17	0.09	0.09	0.09	-0.01	-0.01	0.08

p-value significance: * $p < 0.05$; ** $p < 0.01$; *** $p < 0.001$

Table 1: Results of the Explanation Strategies on Algorithm Understanding, Time Cost and Trust.

Participant Recruitment. We recruited 202 participants from MTurk in August 2018 for the study. To ensure the quality of the survey responses, we only recruited participants with a HIT approval rate of 90% or above, who reside in the US and are aged 18 or above. We randomly assigned each participant to one of the five conditions. The average time for completing the survey was 20 minutes. Each participant received a base payment of $2 and an additional bonus (up to $3) based on the number of correct answers they gave for the objective understanding questions. On average, each participant received a payment of $3, which is above the US minimum wage ($7.25/ hour at the time of writing).

Study Procedure. After consenting, participants completed a background survey. In the survey, participants reported their familiarity with the US graduate school admission process, their algorithm literacy, and their general technical literacy. We told the participants that a computer algorithm had been developed to make automated decisions for university admission for a master's program at a pubic university. Each participant was then randomly assigned to one of the five conditions. Participants explored the interface and then completed a survey which evaluated their understanding of the algorithm and trust in the algorithm [3].

We used an instructed-response question for an attention check, which directed respondents to choose a specific answer in order to detect careless responses [37]. The attention check we included is "Please choose 'disagree' for this question.". We excluded participants who failed the attention check from the study, and did not use their data in our analysis.

4 RESULTS

202 participants completed the study on MTurk. 199 responses were recorded after filtering out the 3 participants that failed the attention check. Regarding algorithm literacy, 70.35% of participants indicated they had "No Knowledge" or "A little knowledge" of algorithms, while 22.61% indicated

[3] The full survey is available in the auxiliary material.

they had “Some knowledge”, and 7.04% indicated they had “A lot of knowledge” of algorithms. The population is aligned with our target user group – people who are not algorithm experts.

Overview of Statistical Models

We used linear regression models to examine whether the different interface conditions led to different levels of objective and self-reported understanding of the algorithm, trust in the algorithm, and time costs (see Table 1). For example, with objective understanding we first used the control group as a baseline, comparing the four experimental groups against it (Model 1); then we examined the comparison between white-box and black-box, and interactive and static, as well as how the two sets of strategies interact with each other (Model 2); finally we included the education level and technical literacy as moderating variables (Model 3).

RQ1 and RQ2: What are the Trade-offs of Different Strategies?

Overview. We found that, compared to the *static* conditions, participants in the *interactive* conditions not only scored higher in the "objective understanding" quizzes but also self-reported a higher level of understanding of the algorithm. Participants in the *white-box* conditions scored higher in the objective quizzes but did not have a higher self-reported understanding than those in the *black-box conditions*. To answer the same number of questions, participants in the interactive conditions spent more time than those in the static conditions; there was no significant difference in the amount of time spent between white-box conditions and black-box conditions.

Objective Understanding. Model 1 and Model 2 in Table 1 show the differences in objective understanding across different conditions. Model 1 illustrates that all four versions of the explanation interfaces led to significant increases in participants' "objective understanding" of the algorithm compared to the text-based explanation (see Figure 2 for a visual presentation). Model 2 explicitly compares the two sets of interfaces: interactive versus static and white-box versus black-box. On average, participants in the interactive conditions answered two more questions (out of twelve) correctly compared to the participants in the static conditions (Coef.= 2.06, p < 0.001). Participants in the white-box conditions answered one more question correctly compared to those in the black-box conditions (Coef.= 1.09, p < 0.01).

Self-reported Understanding. Model 4 and Model 5 in Table 1 show the differences in self-reported understanding of the algorithm across different conditions. The results suggest that *only* the interactive interfaces increased participants' self-reported understanding of the algorithm. For instance,

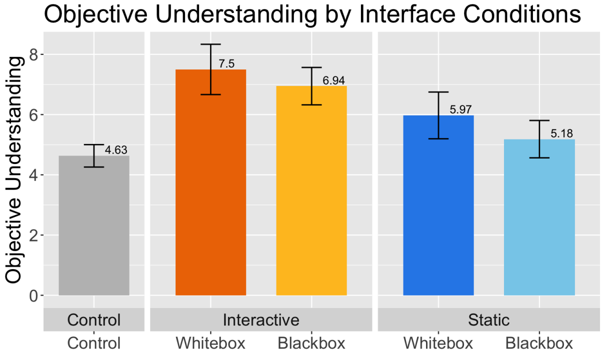


Figure 2: Participants' objective understanding of the algorithms by interface conditions. Error bars represent 95% confidence intervals.

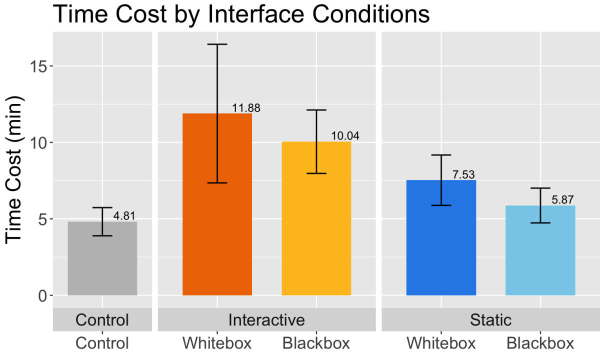


Figure 3: Time cost of understanding the algorithm by interface conditions. Error bars represent 95% confidence intervals.

the white-box interactive interface increased participants' self-reported understanding by 0.69 points on a 7-point scale (p<0.01) compared to the control condition with a text-based explanation; the black-box interactive interface increased self-reported understanding by 0.64 points (p<0.05) compared to the control condition. On average, participants in the interactive conditions self-reported their understanding of the algorithm 0.51 points higher than those in the static conditions (p<0.05). However, there is no evidence suggesting that the white-box interfaces increased self-reported understanding.

Time Cost. Model 7 and Model 8 in Table 1 shows the time participants spent using the explanation interfaces to answer quiz questions. Model 7 suggests that the participants in the two interactive conditions spent 7.07 more minutes (p<0.001) and 5.23 more minutes (p<0.01) respectively compared to the control condition (see Figure 3 for a visual presentation). Model 8 shows that it costs participants in the interactive conditions 4.75 more minutes compared to their static counterpart (p<0.01). We did not observe a statistically significant difference in time spent between the white-box interfaces and black-box interfaces. Therefore, while the interactive interfaces are more effective at increasing comprehension, the trade off is that users take more time gaining that comprehension.

RQ3: Moderating Effects of Individual Characteristics

We found that education level and technical literacy had main effects on the understanding of the algorithm. However, the effects of the explanation interfaces were not affected by how educated or technically literate a participant was. In other words, we did not observe moderating (interaction) effects.

Models 3, 6, and 9 in table 1 include technical literacy and education level, as well as their interactions with the explanation conditions. According to Model 3, having a bachelor degree had a positive effect on objective understanding (Coef. = 1.10, p<0.05). According to Model 6, higher technical literacy was associated with higher levels of self-reported understanding (Coef. = 0.59, p<0.001). However, we did not find interaction effects between education, technical literacy, and the explanation interfaces.

Overall, we did not find evidence that the effectiveness of the explanation interfaces depends on participants' education level or technical literacy.

RQ4: Will the Explanation Improve Users' Trust?

Interestingly, we found that the explanation interfaces had no effect on users' trust in the algorithm. Model 10 and 11 in Table 1 both show that there was no significant difference in the reported trust in the algorithm across all five conditions. Model 12 shows that people with higher technical literacy trust the algorithm more (Coef. = 0.56, p<0.001). However, the explanation interfaces did not increase or decrease the level of trust, no matter how educated or technically literate the participants were.

5 SUMMARY AND DISCUSSION OF RESULTS

Our paper investigates the relative effectiveness of two sets of explanation strategies in helping non-expert stakeholders understand algorithmic decision-making. Findings include:

- *Interactive* interfaces increased both objective understanding and self-reported understanding of the algorithm compared to the *Static* interfaces. At the same time, users using the *Interactive* interfaces spent more time answering the questions (**RQ1**).
- *White-box* interfaces increased objective understanding but not self-reported understanding compared to the *black-box* interfaces (**RQ2**).
- The effects of the explanation interfaces were not influenced by how educated or technically literate the participants were (**RQ3**).
- The explanation interfaces increased users' understanding of the algorithm, but not their trust in the algorithm (**RQ4**).

Why White-box Interfaces Did Not Increase Self-reported Understanding

We found that although white-box interfaces increased objective understanding, they had no effect on self-reported understanding. One possible reason for this is that white-box interfaces reveal more complexity than black-box interfaces (i.e. the chart shows some complexity of the model), which makes participants think they do not understand it as well as they actually do. Another possible explanation is that, although white-box interfaces increase people's understanding of the inner working of the algorithm, they also introduce additional cognitive workload which might reduce people's confidence. As a result, white-box interfaces do not increase people's self-reported understanding of the algorithm.

Why an Improved Understanding Did Not Increase Trust

Although the explanation interfaces were effective in increasing users' comprehension of the algorithm, none of them increased participants' trust in the algorithm over the control condition. We looked into the open-ended responses to better understand this finding.

Some participants reported that they simply feel uncomfortable about the idea of using an algorithm to make important decisions like graduate school admissions. Increasing the transparency of the algorithm does not reduce such concerns. For instance, P54 described the concern precisely: *"[The algorithm] takes a lot of factors into account so I think it is reliable, but I am still uneasy about the idea of a machine making the final decision."*

P119 directly compared humans and computers and believes that humans will consider exceptions that algorithms might ignore.*"There are situations that are an exception that an algorithm could not detect or consider. I think it is important for there to be considerations, and that cannot be done using a computer."*

P79 pointed out that one can "sell" their cases and appeal to human decision makers, which is not possible in algorithmic decision making. *"Because it would allow me to interview and sell my case to the admissions officer directly."*

Other participants think humans are "less cold" and more forgiving than algorithms. *"I want a person to get a feel for who I am by my application."* – P4, *"Programs can not judge who I am as a person only what I have done in the past."* – P32.

We also found that our "unnamed attributes" questions, which were designed to assess people's level of understanding of the algorithm, caused some distrust in the algorithm. Although we explained to the participants that we hid the names of the "additional attributes" with the purpose of testing their comprehension of the algorithm, some participants thought this made them unable to fully trust the algorithm. For instance, P156 pointed out *"The mystery variables have a lot of weight and without knowing what they are, I can't have full trust in the algorithm."* Note that the unnamed attributes were displayed in all conditions (including the control condition). Therefore, although the unnamed attributes may reduce people's level of trust, they do not affect our main results.

6 LIMITATIONS AND FUTURE WORK

As with any study, it is important to note the limitations of this work. One concern is the choice of using an experimental approach. While the experimental approach allows us to draw causal conclusions, it limits our ability to observe how the users actually interact with the explanation tools. In the future work, we will observe how students and admission committees actually use the admission algorithm and the explanation interfaces in real world settings, which can potentially complement our experimental findings.

Supporting users' "right to explanation" is an important issue in a wide variety of domains that involve algorithmic decision-making. We have developed and evaluated explanation strategies and interfaces in the specific context of student admission. Future work is needed to use different contexts to replicate and validate our findings. Through replication, we can either validate our findings, or better understand the circumstances in which these findings do or do not apply.

One possible domain to replicate and validate our findings is recidivism prediction. As mentioned in the introduction, algorithms have been developed to help judges assess the risk of recidivism and decide whether defendants should be detained or released while awaiting trial. We can adapt our interface prototypes to explain recidivism prediction algorithms, with a real-world public dataset [3] and state-of-the-art models (e.g. [30]) or their local linear approximations using the method proposed by Ribeiro et al [45]. For example, future work can be conducted with judges and people with prior criminal histories to: (1) assess whether they can understand the recidivism prediction algorithms with the help of explanation interfaces; (2) ask participants to compare the different explanation interfaces and see if the results are consistent with our findings; and (3) collect their concerns, critiques, and reflections on algorithmic decision-making.

Our findings suggest that people do not fully trust algorithms for various reasons, even when they have a better idea of how the algorithm works. However, we argue that it is not in society's best interest to revert to complete human decision-making and ignore the significant efficiencies gained from automated approaches. One promising direction for addressing this trust issue is to seek valuable synergies between human control and automated machine learning approaches in decision-making, which is similar to the mixed-initiative approach in user interface design [27]. For instance, we can design innovative algorithmic approaches and "human-in-the-loop" systems to leverage people's ability to deal with exception cases and machine's advantages of consistency and efficiency, or we can design mechanisms to support people who wish to appeal to algorithmic decisions.

7 CONCLUSION

Artificial intelligence is rapidly shaping modern society towards increased automation, in some cases making important decisions that affect human welfare. We believe that HCI researchers should strive to find ways to help people understand the automated decisions that affect their livelihood. In this paper, we took steps toward that goal by examining user interface strategies for explaining profiling algorithms. We found that our experimental interfaces increased algorithm comprehension, and that features supporting interacting with and visualizing the inner workings of an algorithm help improve users' objective comprehension.

ACKNOWLEDGMENTS

We would like to thank Daniel Takata, Bob Liu, Estelle Smith, Mark Yousef and our friends at GroupLens Research for their help in this project.

REFERENCES

[1] Ashraf Abdul, Jo Vermeulen, Danding Wang, Brian Y Lim, and Mohan Kankanhalli. 2018. Trends and trajectories for explainable, accountable and intelligible systems: An hci research agenda. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. ACM, 582.

[2] Saleema Amershi, Max Chickering, Steven M Drucker, Bongshin Lee, Patrice Simard, and Jina Suh. 2015. Modeltracker: Redesigning performance analysis tools for machine learning. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. ACM, 337–346.

[3] Julia Angwin. 2016. Make algorithms accountable. *The New York Times* 1 (2016), 168.

[4] Barry Becker, Ron Kohavi, and Dan Sommerfield. 2002. Visualizing the Simple Bayesian Classifier. In *Information visualization in data mining and knowledge discovery*. Morgan Kaufmann, 237–249.

[5] Victoria Bellotti and Keith Edwards. 2001. Intelligibility and accountability: human considerations in context-aware systems. *Human–Computer Interaction* 16, 2-4 (2001), 193–212.

[6] Or Biran and Courtenay Cotton. 2017. Explanation and justification in machine learning: A survey. In *IJCAI-17 Workshop on Explainable AI (XAI)*. 8.

[7] Jenna Burrell. 2016. How the machine 'thinks': Understanding opacity in machine learning algorithms. *Big Data & Society* 3, 1 (2016), 2053951715622512.

[8] Ángel Cabrera, Fred Hohman, Jason Lin, and Duen Horng Chau. 2018. Interactive Classification for Deep Learning Interpretation. *arXiv preprint arXiv:1806.05660* (2018).

[9] Doina Caragea, Dianne Cook, and Vasant G Honavar. 2001. Gaining insights into support vector machine pattern classifiers using projection-based tour methods. In *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 251–256.

[10] Alexandra Chouldechova, Diana Benavides-Prado, Oleksandr Fialko, and Rhema Vaithianathan. 2018. A case study of algorithm-assisted decision making in child maltreatment hotline screening decisions. In *Conference on Fairness, Accountability and Transparency*. 134–148.

[11] Sam Corbett-Davies, Emma Pierson, Avi Feller, Sharad Goel, and Aziz Huq. 2017. Algorithmic decision making and the cost of fairness. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 797–806.

[12] Cynthia L Corritore, Beverly Kracher, and Susan Wiedenbeck. 2003. On-line trust: concepts, evolving themes, a model. *International journal of human-computer studies* 58, 6 (2003), 737–758.

[13] Mark Craven and Jude Shavlik. 1999. Rule extraction: Where do we go from here. *University of Wisconsin Machine Learning Research Group working Paper* 99 (1999).

[14] Michael A DeVito, Jeremy Birnholtz, and Jeffery T Hancock. 2017. Platforms, people, and perception: Using affordances to understand self-presentation on social media. In *Proceedings of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing*. ACM, 740–754.

[15] Michael A DeVito, Jeffrey T Hancock, Megan French, Jeremy Birnholtz, Judd Antin, Karrie Karahalios, Stephanie Tong, and Irina Shklovski. 2018. The Algorithm and the User: How Can HCI Use Lay Understandings of Algorithmic Systems?. In *Extended Abstracts of the 2018 CHI Conference on Human Factors in Computing Systems*. ACM, panel04.

[16] Nicholas Diakopoulos. 2016. Accountability in algorithmic decision making. *Commun. ACM* 59, 2 (2016), 56–62.

[17] Berkeley J Dietvorst, Joseph P Simmons, and Cade Massey. 2015. Algorithm aversion: People erroneously avoid algorithms after seeing them err. *Journal of Experimental Psychology: General* 144, 1 (2015), 114.

[18] Finale Doshi-Velez and Been Kim. 2017. A roadmap for a rigorous science of interpretability. arXiv preprint. *arXiv preprint arXiv:1702.08608* (2017).

[19] Julia Dressel and Hany Farid. 2018. The accuracy, fairness, and limits of predicting recidivism. *Science advances* 4, 1 (2018), eaao5580.

[20] Motahhare Eslami, Karrie Karahalios, Christian Sandvig, Kristen Vaccaro, Aimee Rickman, Kevin Hamilton, and Alex Kirlik. 2016. First i like it, then i hide it: Folk theories of social feeds. In *Proceedings of the 2016 cHI conference on human factors in computing systems*. ACM, 2371–2382.

[21] Megan French and Jeff Hancock. 2017. What's the folk theory? Reasoning about cyber-social systems. (2017).

[22] Nahum Gershon. 1998. Visualization of an imperfect world. *IEEE Computer Graphics and Applications* 4 (1998), 43–45.

[23] Bryce Goodman and Seth Flaxman. 2016. European Union regulations on algorithmic decision-making and a "right to explanation". *arXiv preprint arXiv:1606.08813* (2016).

[24] Spence Green, Jason Chuang, Jeffrey Heer, and Christopher D Manning. 2014. Predictive Translation Memory: A mixed-initiative system for human language translation. In *Proceedings of the 27th annual ACM symposium on User interface software and technology*. ACM, 177–187.

[25] Paul Hitlin. 2016. Research in the crowdsourcing age, a case study. *Pew Research Center* 11 (2016).

[26] Fred Matthew Hohman, Minsuk Kahng, Robert Pienta, and Duen Horng Chau. 2018. Visual Analytics in Deep Learning: An Interrogative Survey for the Next Frontiers. *IEEE Transactions on Visualization and Computer Graphics* (2018).

[27] Eric Horvitz. 1999. Principles of mixed-initiative user interfaces. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*. ACM, 159–166.

[28] Aleks Jakulin, Martin Možina, Janez Demšar, Ivan Bratko, and Blaž Zupan. 2005. Nomograms for visualizing support vector machines. In *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*. ACM, 108–117.

[29] Mary Beth Kery, Amber Horvath, and Brad A Myers. 2017. Variolite: Supporting Exploratory Programming by Data Scientists.. In *CHI*. 1265–1276.

[30] Jon Kleinberg, Himabindu Lakkaraju, Jure Leskovec, Jens Ludwig, and Sendhil Mullainathan. 2017. Human decisions and machine predictions. *The quarterly journal of economics* 133, 1 (2017), 237–293.

[31] Josua Krause, Adam Perer, and Kenney Ng. 2016. Interacting with predictions: Visual inspection of black-box machine learning models. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. ACM, 5686–5697.

[32] John Lee and Neville Moray. 1992. Trust, control strategies and allocation of function in human-machine systems. *Ergonomics* 35, 10 (1992), 1243–1270.

[33] Min Kyung Lee and Su Baykal. 2017. Algorithmic Mediation in Group Decisions: Fairness Perceptions of Algorithmically Mediated vs. Discussion-Based Social Division.. In *CSCW*. 1035–1048.

[34] Brian Y Lim. 2012. Improving understanding and trust with intelligibility in context-aware applications. (2012).

[35] Zachary C Lipton. 2016. The mythos of model interpretability. *arXiv preprint arXiv:1606.03490* (2016).

[36] David Martens and Foster Provost. 2013. Explaining data-driven document classifications. (2013).

[37] Adam W Meade and S Bartholomew Craig. 2012. Identifying careless responses in survey data. *Psychological methods* 17, 3 (2012), 437.

[38] Tim Miller. 2017. Explanation in artificial intelligence: insights from the social sciences. *arXiv preprint arXiv:1706.07269* (2017).

[39] Julian D Olden and Donald A Jackson. 2002. Illuminating the âĂIJblack boxâĂİ: a randomization approach for understanding variable contributions in artificial neural networks. *Ecological modelling* 154, 1-2 (2002), 135–150.

[40] Cathy O'Neil. 2016. *Weapons of math destruction: How big data increases inequality and threatens democracy*. Broadway Books.

[41] Kayur Patel, Steven M Drucker, James Fogarty, Ashish Kapoor, and Desney S Tan. 2011. Using multiple models to understand data. In *IJCAI Proceedings-International Joint Conference on Artificial Intelligence*, Vol. 22. 1723.

[42] Yi Peng, Guoxun Wang, Gang Kou, and Yong Shi. 2011. An empirical study of classification algorithm evaluation for financial risk prediction. *Applied Soft Computing* 11, 2 (2011), 2906–2915.

[43] Tony A Plate, Joel Bert, John Grace, and Pierre Band. 2000. Visualizing the function computed by a feedforward neural network. *Neural computation* 12, 6 (2000), 1337–1353.

[44] Donghao Ren, Saleema Amershi, Bongshin Lee, Jina Suh, and Jason D Williams. 2017. Squares: Supporting interactive performance analysis for multiclass classifiers. *IEEE transactions on visualization and computer graphics* 23, 1 (2017), 61–70.

[45] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. Why should i trust you?: Explaining the predictions of any classifier. In

Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining. ACM, 1135–1144.

[46] Daniel Smilkov, Shan Carter, D Sculley, Fernanda B Viégas, and Martin Wattenberg. 2017. Direct-manipulation visualization of deep networks. *arXiv preprint arXiv:1708.03788* (2017).

[47] Simone Stumpf, Vidya Rajaram, Lida Li, Weng-Keen Wong, Margaret Burnett, Thomas Dietterich, Erin Sullivan, and Jonathan Herlocker. 2009. Interacting meaningfully with machine learning systems: Three experiments. *International Journal of Human-Computer Studies* 67, 8 (2009), 639–662.

[48] Sebastian Thrun. 1995. Extracting rules from artificial neural networks with distributed representations. In *Advances in neural information processing systems*. 505–512.

[49] Meng-Jung Tsai, Ching-Yeh Wang, and Po-Fen Hsu. [n. d.]. Developing the Computer Programming Self-Efficacy Scale for Computer Literacy Education. *Journal of Educational Computing Research* ([n. d.]), 0735633117746747.

[50] F-Y Tzeng and K-L Ma. 2002. *Opening the black box-data driven visualization of neural networks*. 383–390 pages.

[51] Sandra Wachter, Brent Mittelstadt, and Luciano Floridi. 2017. Why a right to explanation of automated decision-making does not exist in the general data protection regulation. *International Data Privacy Law* 7, 2 (2017), 76–99.

[52] Lauren Weber and RE Silverman. 2012. Your resume vs. oblivion. *The Wall Street Journal* 24 (2012).

[53] Daniel S Weld and Gagan Bansal. 2018. Intelligible Artificial Intelligence. *arXiv preprint arXiv:1803.04263* (2018).

[54] Ann Wilkinson, Alison E While, and Julia Roberts. 2009. Measurement of information and communication technology experience and attitudes to e-learning of students in the healthcare professions: integrative review. *Journal of advanced nursing* 65, 4 (2009), 755–772.