Outcome Explorer: An Interactive Visual Interface for Interpretable Algorithmic Decision Making

Md Naimul Hoque, Klaus Mueller

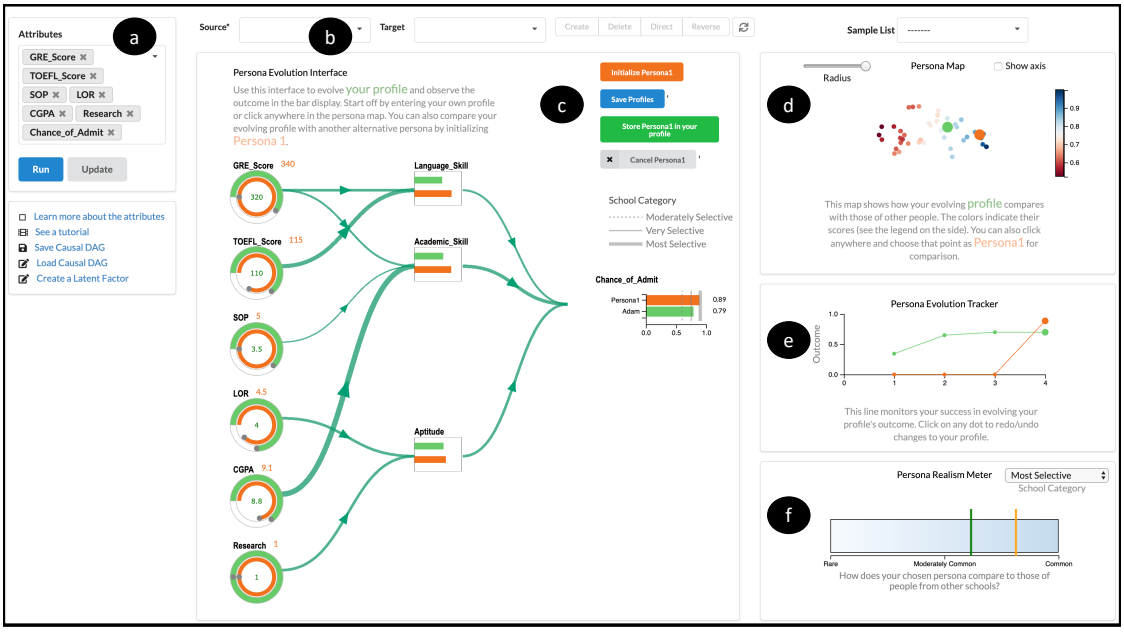

Fig. 1: Visualizing the difference between two profiles (green and orange) in Outcome Explorer. **(a)** The Initialization Menu allows users to create a causal network. **(b)** The Edge Manipulation Panel allows users to add, delete, reverse, and direct causal edges. **(c)** The Persona Evolution Interface includes the causal network with the feature values of two profiles integrated as circular knobs and edges signifying the strength and direction of the causal effects. The latent factors and the outcome variable are represented as horizontal bars. **(d)** The Persona Map visualizes the nearest neighbors of a data point (green disk). **(e)** The Persona Evolution Tracker tracks the evolution of profiles through a line chart and supports redo/undo operations. and **(f)** The Persona Realism Meter determines how common a profile is compared to other people.

Abstract—The widespread adoption of automated decision-making systems has called for the necessity of methods able to explain the decisions of these AI systems. This has given rise to the emerging field of explainable AI (XAI). However, even though end-users are the recipients of the automated decisions, current XAI interfaces are not overly accessible to mainstream users who typically do not possess the mathematical, machine learning, and visualization literacy required by these systems. This problem is difficult to address as the architectures and algorithmic models of the AI decision systems are complex, difficult to comprehend, and for this reason, often referred to as black-box models. In this paper, we explore how graphical models can offer a solution to this problem. We have implemented our methodology as a system we call Outcome Explorer. It first learns a causal model from the data and then represents it as a path diagram or a directed acyclic graph (DAG). Path Diagram offers great interpretability as users can simply follow the causal links to see how the variables are connected to each other, how they affect the model's outcome, and how a decision evolves across the DAG. Outcome Explorer supports several critical capabilities of an XAI system such as answer *What-If* questions, explore nearest neighbors, and facilitate instance comparison, among others. A heuristic analysis with 3 expert-users and a user study with 10 non-expert participants demonstrate that Outcome Explorer allows both expert and non-expert users to understand the decision-making process.

Index Terms—Explainable AI (XAI), Interpretability, Causal Network

1 Introduction

- *Md Naimul Hoque is with Stony Brook University. E-mail: mdhoque@cs.stonybrook.edu.*
- *Klaus Mueller is with Stony Brook University, Inc.. E-mail: mueller@cs.stonybrook.edu.*

Manuscript received xx xxx. 201x; accepted xx xxx. 201x. Date of Publication xx xxx. 201x; date of current version xx xxx. 201x. For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org. Digital Object Identifier: xx.xxxx/TVCG.201x.xxxxxxx

In recent years, algorithmic and automated decision-making systems have been deployed in several critical application areas across society [11, 13, 36]. Researchers and organizations have brought forth the ethical concerns and discriminatory effects associated with these systems. For example, COMPAS [11], a recidivism risk assessment system used in several courtrooms in the United States, has been found to have discriminatory effects on the African-American population [7].

A fundamental problem with algorithmic decision making is its opaqueness or lack of transparency. It makes it difficult for humans to assess whether the decision process is inconsistent or even unfair,

severely lowering the trust in the decisions being cast upon them. A most recent example in this matter is the introduction of apps around the globe to track the spread of the COVID-19 virus. There are reports of potential privacy risk, and misuse associated with such apps which stems from the fact that the algorithmic process in these apps is not transparent [2, 3].

The opaque nature of algorithmic decision-making systems have initiated the General Data Protection Regulation (GDPR), which mandates that the data controller should provide information to data subjects in a *concise, transparent, intelligible, and easily accessible form* [46]. Thus, there exists a genuine need to explain machine-generated decisions and that has led to the recent surge of XAI systems [4, 6, 22, 24, 25]. While these systems have shown to be effective in explaining the decision-mechanism of a wide variety of models, they require a degree of mathematical, machine learning, and visualization literacy that is hard to expect from people who are not algorithmic experts.

"How to develop an XAI interface that is understandable by both expert and non-expert users? "— that is the central research question (**RQ**) we concentrated on in this paper. In other words, our goal is to support the explanation needs of both expert and non-expert users. To support our research agenda, we explore the opportunity of graphical models in this paper. In particular, we concentrate on the causal model, which has gained attention in recent years due to its capability of inferring the causal relationships between sensitive and outcome variables [28, 54]. In addition to its effectiveness in fairness research, a causal model is represented as a path diagram or a DAG which is easy-to-understand and inherently interpretable. Based on a path diagram, a user can intuitively understand how variables are related to each other and how they affect the outcome variable, which meets the property of a white-box, fully transparent model.

In this paper, we present Outcome Explorer, a causality-based interactive decision-making platform. We designed Outcome Explorer keeping in mind two sets of users: *Policy-maker* and *Policy-receiver*. We define *Policy-maker* as an individual or a group of people who wants to devise a policy or a product through automated-systems. A policy-maker is an expert-user with domain knowledge such as an ML practitioner or a project manager, responsible for creating and moderating the policy through Outcome Explorer. Conversely, a *policy-receiver* is the recipient of the policy or product offered by the *policy-maker*.

Outcome Explorer combines the predictive and explanatory components of an XAI system into one interface. Thus, there is no explanation paradigm in our system, rather the predictive platform simultaneously works as a predictive and explanatory platform. Informed by prior research, Outcome Explorer lets policy-makers build the causal model from scratch, facilitates several XAI functionalities such as answer What-If questions, explore nearest neighbors, compare instances, etc., and finally allows policy-makers to publish the policy with different degrees of XAI functionalities enabled.

On the other hand, Outcome Explorer works as a plug and play platform from a policy-receiver's perspective. As shown by Dietvorst et al. [15] humans tend to understand and trust a system much more when they are given some control – even just a slight amount – to play (tinker) with it and observe its reactions. In our scenario, it would empower policy-receivers to better understand the encoded policy, build trust, and motivate self-improvement. Our system allows users to plug in their data to see the policy outcome in the path diagram and it facilitates interactive probing of the model so that users can understand the reason behind the outcome and how to achieve alternative outcomes.

In summary, we make the following research contributions-

- A mechanism to facilitate prediction in a causal model based on the established statistical causality framework.
- The design and implementation of Outcome Explorer, an interactive visual tool to support the explanation needs of both expert and non-expert users in a decision platform.
- A heuristic analysis with 3 expert users to measure the usability of the system and to understand how Outcome Explorer supports policy-makers' explanation needs.
- A user study with 10 non-expert participants to evaluate the effectiveness of Outcome Explorer in model understanding from policy-receivers' perspective.

In the following, Section 2 discusses related research, while Section 3 describes the theoretical background and the underlying methodology of our approach. Section 4 presents the design challenges and goals of the tool, while Section 5, 6, 7, and 8 together present the interactive interface. Section 9 presents evaluation results through a heuristic analysis and a user study. Finally, Section 10 discusses the implication and limitation of the tool with conclusions drawn in Section 11.

2 Related Works

2.1 Explainable AI (XAI)

2.1.1 Interpretability vs Explanation

Black-box models such as Deep Learning are being used for diverse tasks [29]. Several systems have been developed to provide explanations to these complex and opaque models [23–25, 52]. Recent research has shown that explanations to black-box models can be misleading, and can complicate model understanding [40]. Thus, researchers are urging for using models that are simple and interpretable instead of black-box models coupled with posthoc explanation models [40]. In the existing literature, model interpretability usually pertains to human understanding of a model [31, 40], although there are no agreed-upon definitions [30]. According to this notion, causal models are interpretable, as humans can understand the models visually through the path diagrams. In this paper, we seek to evaluate the causal model as a transparent and interpretable decision-making platform.

2.1.2 Users

Irrespective of whether an XAI system is explaining a white-box or a black-box model, there are recent critiques that XAI systems tend to support the explanation needs of the model-builders, not the actual users who are the recipient of the decisions [4, 12]. It is not clear from the empirical evidence whether these XAI systems are understandable and usable by the end-users [4, 16]. This paper embarks on the task of creating an easy-to-understand predictive platform for diverse users through the explanatory properties of a causal model.

2.1.3 Visual Analytics and XAI

Interactive visual systems are shown to be an effective way to augment a human's understanding of the automated decision-making paradigm [6, 22, 24, 27]. A recent example of visual analytics tool to promote human-AI collaboration is GAMUT [22], which employs multiple coordinated views to explain the prediction mechanism of the Generalized Additive Model (GAM). GAMUT was specifically designed to support the explanation needs of ML practitioners, while we aim towards a more generic goal of supporting the needs of both expert and non-expert users.

We also note that there are certain similarities of our approach at least in spirit with the recent *What-If Tool* [50]. The tool uses scatter plots, line plots, and text interfaces to allow users to query and compare the outcomes of different decision models, but without showing the model itself. While this leads to an understanding of the model's behavior in a counterfactual sense (the 'if') it does not explain, and allow a user to play with the reasoning flow within the system (the 'why') which our approach facilitates.

2.2 Interactive Causal Analysis

We employ the visual causality analysis approach introduced by Wang et al. [48] in this paper. The authors used Total Conditioning and the PC algorithm (details can be found in [18]) to infer a causal model from the dataset and then used SCM [39] (Linear and Logistic regression) to parameterize it. The paper by Wang et al. presented an improvement over their previous work [47], offering a flow-based visual encoding of the causal network with the additional analytical functionality of inferring separate causal networks for data sub-clusters and then allowing users to fuse sets of them within a visual interface. While [47]

provided the first fully interactive and visual platform for causal analysis, similar graph-based visual analyses have been developed for other purposes, such as visualizing belief networks [55], correlation networks [56], uncertainty networks [41], annotation graphs [57], and for visualizing ambiguity in graphs [49]. Prior efforts that employed such flows/networks for causality visualization include *DecisionFlow* [20], *reactFlow* [14], *outFlow* [51], and others [17, 45].

2.3 Fairness and Causality

As mentioned in the introduction, causal models are becoming increasingly popular among researchers in computational fairness as they convert the problem of fairness to a simple question – "Are there cause and effect relations among the protected variables and the target variables?" which has high explanatory value [28, 54]. Kusner et al. [28] defined a decision to be *counterfactually fair* if it is the same in (a) the actual world and (b) a counterfactual world where the individual belonged to a different demographic group. Our visual representation of the causal network can give the user the necessary clarification and transparency on whether the model is affected by the protected variables or not. Hence, it provides a way to convey the counterfactual fairness message to the end-user. Although we did not use any of the standard fair learning algorithms for causal inference in this paper, all of these can be used to infer a causal network from the given dataset and then integrate this network into our system. This will make these systems *Fair, Accountable,* and *Transparent* simultaneously.

3 Background for Statistical Causal Model

In this section, we formally define causal models and provide necessary background knowledge to understand our methodology in Section 5.

3.1 Causal Model

We follow Pearl's Structural Causal Model (SCM) [38, 39] to define causal relationship between variables. According to SCM, causal relations between variables are expressed in a Directed Acyclic Graph (DAG), also known as a Path Diagram. In a path diagram, variables are categorized as either Exogenous (U) or Endogenous (V). Exogenous variables have no parents in a path diagram and are considered to be independent and unexplained by the model. On the other hand, endogenous variables are fully explained by the model and presented as the causal effects of the exogenous variables. In figure 2, A, B are exogenous variables, whereas C, D, and E are endogenous variables.

Pearl defined Causal Model as a set of triples (U, V, F) such that

- U is the set of exogenous variables , and V is the set of endogenous variables.
- Structural equations [8] (F) is a set of functions $\{f_1, ..., f_n\}$, one for each $V_i \subseteq V$, such that $V_i = f_i(pa_i, U_{pa_i})$, $pa_i \subseteq V \setminus \{V_i\}$ and $U_{pa_i} \subseteq U$.

The notation "pa_i" refers to the "parents" of V_i. Pearl suggested Structural Equation Modeling (SEM) [42] to learn the weights of the path diagram [38]. Once the data is standardized, the weights are popularly known as path coefficients, beta weights, or standardized estimates. We will use the notation beta weights in this paper. The terms "causal model" and "path diagram" are used interchangeably in this paper.

3.2 Direct, Indirect, and Total Effects

We define the outcome variable as an endogenous variable, the variable we want to predict. For example, variable E in figure 2 can be the outcome variable. In linear regression, we explicitly concentrate on the direct effects of the predictor variables on an outcome variable. In contrast, in a causal model, a variable may have a direct effect on an outcome variable as well as indirect effects running through other variables. For example, in figure 2, variable A does not have any direct effect on variable E but has indirect effects through variable C and D. The total effect of a predictor (P) on the outcome variable (O) is the sum of all the direct and indirect effects of P on O. The idea of direct, indirect, and total effects were introduced by Wright in path analysis [53].

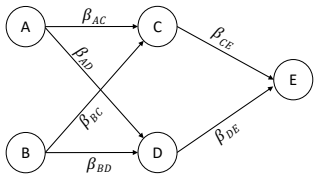

Fig. 2: Example of Path Diagram

In path analysis [53], the total effect (T) of a variable on another variable is the sum of the effects incurred by all the different pathways between the two variables. More formally:

$$T_{ij} = \sum_{m=1}^{Pa} \prod_{(x,y) \in Pa_m} \beta_{xy} \tag{1}$$

Here, Pa denotes the set of distinct paths or causal chains from variable i to j. The notation "$(x,y) \in Pa_m$" denotes all the edges in a single path. For example, the total effect of variable A on variable E is:

$$T_{AE} = \beta_{AC}\beta_{CE} + \beta_{AD}\beta_{DE} \tag{2}$$

Once all the total effects are learned, we can use them to obtain a weighted sum of the outcome variable. According to figure 2, we can mediate/estimate the variable E by the following equation:

$$\hat{E} = T_{AE}A + T_{BE}B + T_{CE}C + T_{DE}D \tag{3}$$

Since the data is standardized and therefore centered, the intercept for estimating E is zero.

3.3 Confirmatory Factor Analysis (CFA)

A latent factor [8] is a variable that is not observed, rather it can be inferred from the observed variables. Latent factors are important features of a causal model as they are often the center of interest in causal analysis and can add meaning/explanation to an existing causal model. For example, in medical science CFA is exercised as a dimensionality reduction technique to measure latent variables such as *anxiety* from observed variables such as *blood pressure and heart rate*. CFA is a part of SEM and is often called the measurement model of SEM [42]. Unlike Exploratory Factor Analysis (EFA), CFA assumes that the construct of the factor is already known by the researcher, and confirms the hypothesis through statistical measures.

4 Design

We decompose the RQ by first surveying the capabilities of existing XAI systems from a policy-maker's perspective. We then augment those capabilities into our system, keeping in mind the literacy gap between policy-makers and policy-receivers.

Explanations in the existing XAI literature can be broadly categorized as either *Global* or *Local*. Examples of global explanations include explaining the model structure, feature importance, and evaluation metrics. On the other hand, local explanation concentrates on the prediction paradigm around a single or a cluster of data. We build our design guidelines from the analysis of these two broader categories.

4.1 Global Explanation

Ideally, global explanations should be readily available to users. Users need not do something to understand the broader model mechanism. In our case, a user should be able to understand the global overview of the causal model without interacting with the interface. Our interface should allow the user then to interact with the system to obtain the local explanations, resembling the visualization mantra of "overview first, zoom and filter, then details on demands" [43].

4.2 Local Explanation

Local explanations try to explain the outcome of a model given a set of input features. XAI systems provide these outcome-oriented explanations through several functionalities. These functionalities seem to vary from system to system, and often largely depend on the chosen model. Hohman et al. [22] listed six capabilities needed to support ML practitioners explanation needs. We briefly list four of them here which are relevant to the local explanation needs of a policy-maker:

- **C1** *Local instance explanations*: Explain the prediction for a single instance through feature contribution.
- **C2** *Instance explanation comparisons*: Allow a comparison between instances in terms of the prediction mechanism.
- **C3** *Counterfactuals*: Allow the user to explore "What-If" questions as a probing method.
- **C4** *Nearest neighbors*: Allow the exploration of nearest neighbors of a data instance in terms of a prediction or features.

We consider these capabilities while designing the tool for policy-makers. To understand the policy-receivers' needs, we turn to GDPR's agenda which identifies that consumers should receive meaningful explanations of the decisions they receive. This eventually means that policy-receivers should have access to C1 and C3. There are potential privacy-risk (e.g. leaking other's personal information) associated with C2 and C4, and that is why a policy-receiver's access to these capabilities should be optional.

4.3 Interaction

Prior research has shown that interactivity can be a powerful way to improve human understanding of the automated process [4]. We turn to Norman's "Action Cycle" to design an interaction paradigm in our system [35]. According to Norman, humans start using a system ("the world") with a goal in mind, then do something to achieve that goal (execution), and evaluate the impact of that execution in the world (evaluate). In the context of our system, the world would be the causal model, the goal would be the outcome (e.g. "getting credit card approval"), execution would be changing the variable's value, and evaluation would be checking how close the outcome variable is to the goal.

4.4 Design Guidelines

Based on this prior research and design requirements, we list the following design guidelines:

- **G1** The global structure of the model should be readily available to the users.
- **G2** The interface should support the local explanation needs of a policy-maker. This includes creating the model from scratch, evaluating the model, and supporting C1-C4.
- **G3** The interface should work as a playful interface for a policy-receiver. The policy-receiver should be able to plugin input features in the interface, receive a decision, then be able to interact with the system to understand the process (according to the action cycle). The policy-receiver should have access to C1 and C3 with optional access to C2 and C4.
- **G4** Our tool should employ an easy-to-understand visual interface. Each component should be created keeping in mind the literacy gap between a policy-maker and a policy-receiver.
- **G5** Our tool should provide information on how realistic an outcome is in comparison to the actual world.

The fact that people might try to "game" a transparent system such as ours is the reason why we introduced G5. In the literature, this is known as *transparency-gameability trade-off*. As discussed by Rudin [40], gameability is acceptable as long as it incentivizes people to modify their behavior in positive ways. Nevertheless, our tool should have a notion of validity in terms of the real world. We link the capabilities (**C1-C4**) and guidelines (**G1-G5**) to the development of Outcome Explorer in the following sections.

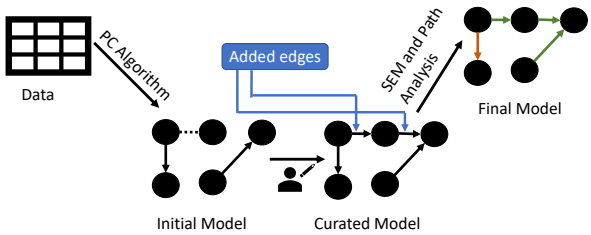

Fig. 3: Workflow of Outcome Explorer. The pipeline starts with creating an initial model using the PC algorithm, then manually curating the model to verify the causal relations, and finally parameterizing the model through SEM and Path Analysis.

5 Methodology

Causal networks do not allow prediction by default. In this section, we describe how this can be accomplished using a three-step method gleaned from structural equation modeling and path analysis.

1. Learning the Causal DAG: The first step in the causal analysis is to obtain the causal DAG. The causal DAG or the path diagram can be learned through controlled or randomized experiments. Scientists also employ prior knowledge to direct the edges between variables. Conversely, a series of automated algorithms [18] have been proposed to learn the causal structure between variables. While these automated algorithms may unveil counter-intuitive causal relationships, they can not be trusted on their own as the causal relations are derived from observed data and may not hold true in the actual world. Thus, we adopt a mixed-initiative approach to learn the causal DAG, as suggested by Wang et al [48]. We employ the PC algorithm [18] as a starting point of the analysis and then allow the users to direct or manipulate edges between variables interactively, based on their prior knowledge obtained from experiments or studies or well-known phenomena. This mixed-initiative approach is also consistent with the IBM SPSS AMOS [9].

2. Obtaining Beta Weights: Once the causal DAG is learned, we employ SEM to quantify/parameterize the relationship between variables. The data is standardized so the weights in SEM essentially become beta weights.

3. Prediction: Finally, we calculate the total effects of each variable on the outcome variable using path analysis (as described in Section 2.2). Once the total effects are learned we estimate the outcome variable as the weighted sum of the variables, where the weights are the total effects. Intuitively, this method is similar to Linear Regression, except we consider a causal structure between variables where the total effects work as regression coefficients.

Figure 3 presents the workflow for allowing prediction in a path diagram. We used the causal model to define the relations between variables, SEM to parameterize the model, and finally path analysis for estimating the outcome variable. We considered *do*-operator for our purpose, but *do*-operator simulates physical intervention in the causal model by setting a variable to a constant while removing some causal relations from the model and keeping the rest of the model unchanged [39].

6 Dataset

Outcome Explorer is particularly suited for *profiling* [19], defined as the process of analyzing aspects of an individual's personality, behavior, interests, and habits to make predictions or decisions about them. Although Outcome Explorer can be adapted for widespread usage (see Section 8), our primary focus is on high-stake decision tasks such as admission decisions, loan approvals, etc.

We demonstrate Outcome Explorer based on an example model built from a graduate admissions dataset [5]. Automated university admission has seen considerable interest in recent years [1, 12, 37]. Further, for this dataset, we can recruit expert and non-expert users (academicians) for evaluating our system with a mutual interest in the task domain.

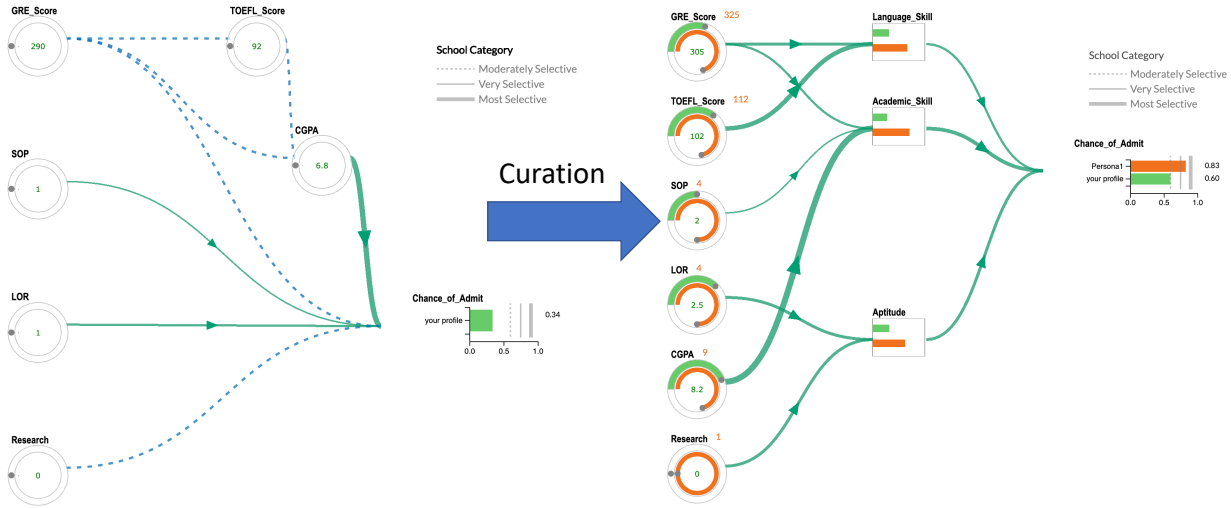

Fig. 4: Model Curation. (Left) Initial model obtained from the PC algorithm. (Right) Curated model with latent variables, showing the difference between two profiles (green and orange).

Model Curation. We developed the interface iteratively, holding several formal and informal meetings with two expert-users (evaluators) to discuss several aspects of the tool. Our first session was conducted to hypothesize the causal model from the admission dataset. The admission dataset has 500 students profiles with features such as CGPA, GRE, TOEFL, SOP (Statement of Purpose), LOR (Letter of Recommendation), Research Experience (dummy variable). Figure 4 (left) presents the causal model obtained from running the PC algorithm on the dataset. From the figure, it is evident that the automated algorithm could not determine the direction of some edges (dotted lines), and the overall model is not conclusive enough. Further, the evaluators agreed that attributes like *GRE* can be a measure of a student's academic and language skills. Together, they hypothesized that the model should include latent factors that will make the model more explanatory and interpretable. Figure 4 (right) presents the refined model with three new latent factors introduced through CFA: Academic_Skill, Language_Skill, and Aptitude.

Conventionally, latent factors are seen to be the causes of the observed variables (latent $\rightarrow$ observed). But, in this case, latent factors are represented as the constructs of the observed variables by reversing the edge directions (observed $\rightarrow$ latent). The evaluators agreed that representing the latent factors in this way added more explanatory power to the interface. Besides, the evaluators stated that the usual representation (latent $\rightarrow$ observed) might confuse the end-users and hinder the natural flow of the graph (left to right) (G4). The regression weights work as correlations, so we did not violate any mathematical assumptions by reversing the edges.

Model Fit. The final SEM model had a χ^2 value of 103.16 with $p = 0.007 (< 0.05)$. As discussed by Kline [26] and Tanaka [44], it is very hard to obtain a non-significant χ^2 value and this is acceptable as long as other metrics are satisfactory. Our model had a good fit in terms of the GFI (0.95), CFI (0.968), and RMSEA (0.085).

Profiles. In our interface, a student's profile (user) is represented with **green** color. We also introduced a second profile, Persona1, represented by **orange** color to facilitate the comparison between two profiles. The two profile mechanism lets a student compare the student's profile with an existing student from the database (C2, C4) and obtain answers to what-if questions (C3) by observing the impact of changing one profile while keeping the other profile fixed. For example, figure 4 (right) presents how a student can compare student's profile with an existing student (Persona1) from the database. This comparative mechanism is consistent among all the coordinated views and the usefulness of this method is explained in detail as we go along with the components of the tool. The name of these two profiles are task-dependent and may vary depending on the dataset and task at hand.

Labels. Finally, we introduced three labels to the dataset to generalize the model for both regression and classification tasks. The labels indicate the thresholds for getting admitted to "Moderately Selective", "Very Selective", and "Most Selective" schools (Figure 4).

7 Outcome Explorer

In this section, we present the visual interface of Outcome Explorer and discuss some of the design choices that were made, how they have evolved during the iterative process and alternatives that were considered to fulfill design guidelines and capabilities in the development of the tool. At first, the full-fledged interface is presented for a policy-maker. Later, we demonstrate how the full-fledged interface can be adapted for policy-receivers. We used Python as the back-end language and D3 [10] for interactive web-based visualization.

Outcome Explorer is divided into six coordinated components with the central focus on the causal model. The components are (A) Initialization Menu, (B) Edge Manipulation Menu, (C) Persona Evolution Interface (the Causal Model), (D) Persona Map, (E) Persona Evolution Tracker, and (F) Persona Realism Meter.

7.1 Initialization Menu

The leftmost panel (A) in figure 1 is the attribute selection panel. The user selects the attributes to be considered in the causal model. The attributes are sorted according to their PCA loadings. We chose this ordering so that the user can choose the attributes that can explain most of the variance of the dataset. As the causal model can become very complex and difficult to analyze visually when the number of nodes increases this panel is helpful to narrow down the important attributes for the network.

The initialization menu also facilitates several other functionalities: (1) description of the attributes, (2) a video tutorial of the system, (3) saving and loading a causal model for repeated use, and (4) creating a latent factor from the observed variables.

7.2 Edge Manipulation Menu

The second panel (b) is the causal edge manipulation panel. Causal network inference from observational data can sometimes misdirect edges due to the statistical noise in the data. Our interface allows users to employ expert knowledge or just common sense to correct these edges, or also add or remove edges.

7.3 The Causal Model

The causal model is presented as a flow diagram (left to right) to visualize the global structure of the model (G1). The nodes represent the variables and the edges signify the relations between the variables.

7.3.1 Observed Variables

The observed variables in the causal model are presented as knobs in the interface. Each knob consists of two circular bars, green bar for the user, and the orange bar for Persona1. Additionally, each knob has two numbers that correspond to two profiles. Finally, the variable names or labels are shown above the knobs.

Orange numbers are presented on the right side of the labels as they are only available when Persona1 is activated, whereas green numbers are presented in the center of the knobs as they correspond to the actual users. This also facilitates a less clumsy representation of the numbers.

A user can obtain a decision from the interface by setting the input variables (C1). The range for the input knobs is set from the min to the max of a particular variable. Each knob provides a grey handle which a user can use to move the knob through mouse drag action. The user can also set the numbers directly in the input boxes. This allows a user to input an exact number or even input a number that is out of range (outside $(max - min)$ range) for that variable. In case of an out of range value, the circular bar is simply set to min or max, whichever extrema are closer to the value.

We also evaluated vertical and horizontal bars for representing the nodes, but all three evaluators agreed that presenting the nodes as circular bars are the most aesthetically pleasing representation given the natural flow of the causal network (G4).

7.3.2 Latent Factors and Outcome Variable

Latent factors are constructed as weighted sums of the observed variables and are not editable. The weighted sums are converted to a 0-5 scale to depict the strength of the factors and presented with horizontal colored bars. We chose horizontal bars as the flow diagram already has a natural horizontal flow (left to right) (G4).

Similar to the latent factors, the outcome variable is presented as a bar chart in the causal model. There are two bars in the chart if Persona1 is activated, one otherwise. The chart provides vertical lines to indicate the thresholds for the classes if the underlying task is a classification task.

7.3.3 Edges

Edges between two numerical variables can have two different colors. A green color indicates a positive cause and effect relationship between the variables while a red edge indicates a negative relation. Furthermore, the thickness of an edge indicates the effectiveness of the relation (beta weight) between the variables.

7.4 Persona Map

According to C2 and C4, one of the primary functionalities of an XAI system is to allow a user to explore the nearest neighbors of a data point and allow comparisons between data points in terms of the outcome of the model. To facilitate this we introduce a biplot as the Persona map, where the points are the samples from the database and the vectors are the variables. To compute the biplot, we run PCA on the selected points. A user can control the radius of the neighborhood (range of outcome variable) through the "Radius" slider. Figure 5 shows an example of persona map in terms of a user (green disk). The neighbors are colored according to the outcome value (color map on the right of the plot).

A user can take the mouse over any point on the map to set that particular point as the Persona1 and see how user and Persona1 differ in terms of variables and outcomes in the causal model. Subsequently, the user can click on any point to set that point as Persona1 for a more detailed analysis. Both green and orange disks move around the map as the user changes the profiles in the causal model.

We introduced the checkbox "show axis" in the map in our second design iteration as the evaluators found the axes to be cluttered and confusing (violating G4). On the other hand, they acknowledged that the axes can be useful to users as they depict directions towards goodness or badness of the neighbors in terms of the outcome variable. Keeping that in mind, instead of completely removing the axes, we introduced the checkbox so that a user can click on the checkbox to see the axes.

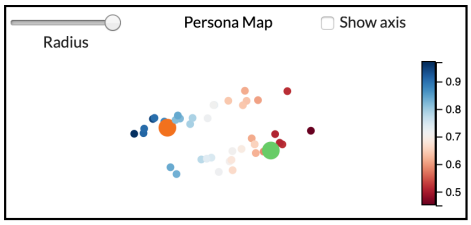

Fig. 5: Persona Map

Fig. 6: Persona Evolution Tracker

7.5 Persona Evolution Tracker

One of the fundamental features of any UI is the support for a redo and undo. A user should be able to move back and forth between different states easily. We introduced Persona Evolution Tracker (figure 6) to facilitate that. The tracker is a simple line chart with two lines for two profiles in the system. The x-axis represents the saved state while the y-axis shows the outcome values at that particular state. A user can click on the "Save Profiles" button to save a particular state in the tracker and can click on any point in the tracker to go back and forth between different states.

7.6 Persona Realism Meter

According to G5, there is a certain danger that to achieve a specific outcome in Outcome Explorer a user might opt for a profile that is unlikely to be a real person in the actual world. We introduced a realism meter to facilitate the notion of how "real" a profile is in terms of the existing profiles in the database.

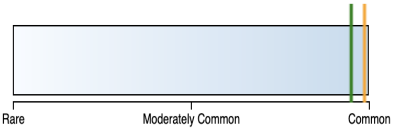

Fig. 7: Realism Meter

To determine how (dis)similar a profile is to the existing users, we opt for a multi-dimensional method similar to detecting an outlier in one dimension using the z-score. At first, we fit a Gaussian Mixture Model on the existing users. A mixture model with K Gaussians or components is defined as

$$P(X) = \prod_{n=1}^{N} \sum_{k=1}^{K} P(X_n|C_k)P(C_k) = \prod_{n=1}^{N} \sum_{k=1}^{K} \phi_k N(X_n|\mu_k, \Sigma_k) \quad (4)$$

where N is the number of datapoints, $\phi_k = P(C_k)$ is the mixture weight or prior for component k, and μ_k, Σ_k are the parameters for the k-th Gaussian.

Once the parameters are learned through the Expectation-Maximization algorithm, we can calculate the probability of a datapoint x belonging to a component C_i using the following equation

$$P(C_i|x) = \frac{P(C_i)P(x|C_i)}{\sum_{k=1}^{K} P(C_k)P(x|C_k)} = \frac{\phi_i N(x|\mu_i, \Sigma_i)}{\sum_{i=1}^{K} \phi_k N(x|\mu_k, \Sigma_k)} \quad (5)$$

A high value of $P(C_i|x)$ implies that x is highly likely to belong to C_i, whereas a low value $P(C_i|x)$ implies that the features of x is not common among the members of C_i. Thus, $P(C_i|x)$ can be interpreted as a scale of how "real" a datapoint is to the other members of a component. We translate $P(C_i|x)$ to a human understandable meter with $P(C_i|x) = 0$

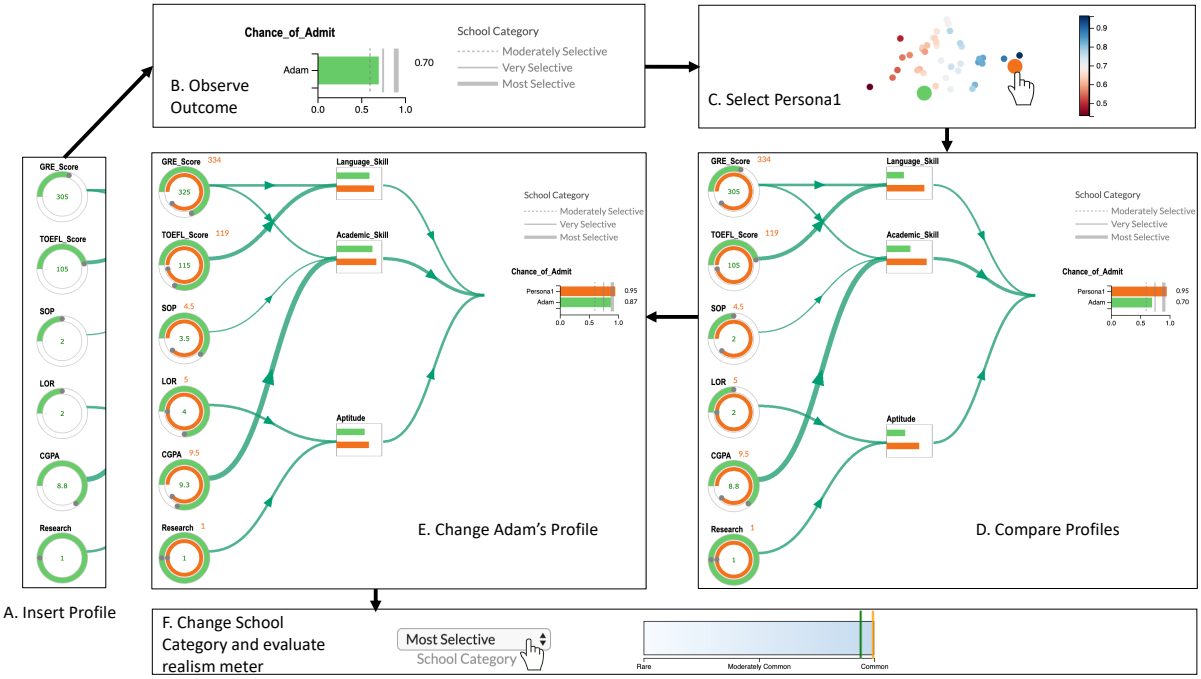

Fig. 8: Interaction with Outcome Explorer. Consider an admission counselor who is using Outcome Explorer to provide admission suggestions to Adam, a prospective student. (A) The counselor inserts Adam's profile in the interface. (B) The counselor observes Adam's Chance of Admission. (C) Suppose, Adam is targeting the "Most Selective" schools and wants to know how to improve the Chance of Admission. The counselor selects a previous student with a higher chance of admission for comparison from the Persona Map. (D) The counselor compares the two profiles and (E) changes Adam's profile to increase Adam's admission chance. (F) Finally, the Counselor changes School Category in the Realism Meter and observes how common Adam's new profile is to the existing students from the most selective schools.

interpreted as "Rare", $P(C_i|x) = 0.5$ as "Moderately Common", and $P(C_i|x) = 1$ as "Common". Figure 7 presents the visual abstraction of the realism meter. The vertical green and orange line represent the current location of the user and Persona1 in the meter.

A user can select the reference component C_i ("Most Selective" schools for example) from the dropdown menu attached to the Realism Meter. In case of a classification problem, K is set to the number of classes and the dropdown menu has $K+1$ options with one extra option being "All" to compare a datapoint to all the data points in the database. In the case of regression, the policy-maker can select the number of components for the gaussian fit in the interface.

We considered visualizing the distribution of the members (probabilities) of a class through a histogram in the realism meter. But the distributions were skewed, unpredictable, and the evaluators found them confusing and distracting (violating G4).

8 Use Cases

Figure 8 presents a use case based on the admissions dataset. The use case presents how an admissions counselor can benefit from our interface and can provide insightful advice to a hypothetical prospective graduate student, Adam, based on the evidence gathered from Outcome Explorer. The use case demonstrates the implementation of G2 in Outcome Explorer.

We present a second use case based on the boston housing dataset [21]. We chose this dataset as it appeared in the previous XAI literature [22] and the task is significantly different than that of predicting the chance of admission (profiling task). The task, in this case, is to predict the median prices of houses in different neighborhoods based on 13 contextual variables. We chose the 8 most contributing variables (in terms of explained variance) for building the causal model. Figure 9 presents the causal model obtained from the PC algorithm.

Understand Causal Relations. Outcome Explorer can be useful in untangling complex and non-intuitive relations among variables. For example, from figure 9 we can see that "Distance from City" negatively

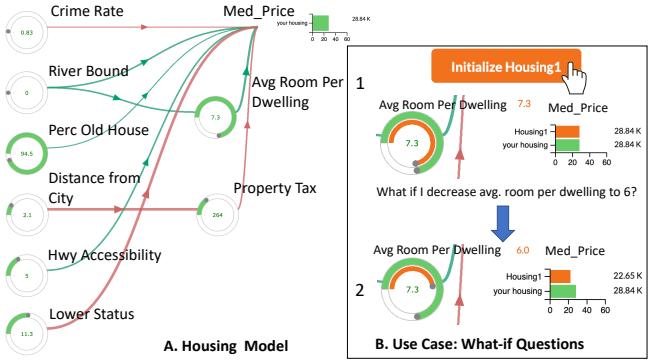

Fig. 9: **A.** Causal Model on the Boston Housing Dataset. **B.** Suppose a user wants to ask a series of what-if questions. **1.** The user clicks on the "Initialize Housing1" to initialize the orange profile to be exactly the same as the green profile (one variable of interest is shown for brevity). **2.** The user changes the orange bar while keeping everything else constant to observe the effect in the outcome variable.

affects "Property Tax" of a neighborhood, meaning neighborhoods away from the city usually have lower property tax. Subsequently, property tax has a negative effect on the housing prices, meaning houses in a neighborhood with higher property tax are priced less than houses in a neighborhood with lower property tax. The inference is that property tax decreases as one moves away from the city, but this subsequently increases the housing price. Note that this causal structure is only true for this dataset (houses in Boston in 1980); it may not hold true for houses in other areas.

What-if Questions (C3). A user can readily obtain what-if answers from Outcome Explorer by simply changing the knobs or the input

boxes. The two profile comparative mechanism offers a more elegant way to obtain what-if answers. Figure 9B presents a use case to demonstrate the implementation of G3 based on the housing dataset. A user can make changes to the green and orange bars interchangeably to ask similar sorts of what-if questions. The Persona Tracker allows a user to go back and forth between different comparison states. An example of this comparative mechanism also appears in figure 8.

9 EVALUATION

Our evaluation is two-folded. At first, we evaluated the usability of our tool by conducting a heuristic analysis with 3 expert-users i.e. policy-makers. Secondly, we conducted a user study to evaluate how policy-receivers or end-users use Outcome Explorer. We conducted post-study interviews with both policy-makers and policy-receivers and summarized the discussion points in Section 10.

9.1 Heuristic Analysis

Heuristic Analysis [33, 34] is a widely used method to evaluate the usability of a user interface (UI). It is intended to uncover errors or usability problems in a UI. Since policy-receivers, who do not have domain expertise, are one of our target users, our tool must not have usability problems. Otherwise, the usability problems can hinder a user from interpreting the decision mechanism. Also, the analysis lets us evaluate the tool quantitatively.

Analysis Setup. Prior research suggests that 3-5 participants are sufficient to conduct a heuristic analysis [34]. We recruited three domain experts, who were not involved in the design phase, to conduct the heuristic analysis. All three participants have post-graduation degrees and have conducted research in the field of XAI, Fairness, and Data Ethics for at least five years. Each participant completed a task list in separate sessions which typically lasted for 1-1.5 hours.

To conduct the study we deployed the interfaces on the web and conducted the sessions via Skype. Participants shared their screen as they performed the tasks. One author communicated with the participants during the sessions while another author took notes. Each session started with a tutorial period. Participants were allowed to use the system as long as they wanted and one of the authors answered their questions about the interface. After that, participants were given a scenario and a task list. The scenario resembled the interaction scenario in figure 8 where each participant took the role of a graduate admission counselor. Figure 8 represents a subset of the tasks carried out by the participants. All the tasks were designed to be outcome-oriented to reflect the action cycle (Section 4).

The participants reported the usability problems as they performed the tasks. While reporting, the participants provided a severity score to each problem on a scale of 0 to 4, as suggested by Nielson [33], where 0 corresponds to "Not a usability problem", and 4 corresponds to "Usability catastrophe". Further, the participants explained the problems in terms of Nielson's 10 heuristics [32] and provided insights on how to solve the problems.

Results. All three participants completed all the tasks. They found 3 minor usability problems (severity 2), 2 major usability problems (severity 3), and no usability catastrophe (severity 4) in the UI. The number of problems reported in the analysis indicated that the interface was already in good standing. Nevertheless, we conducted a final design and implementation step to refine the UI based on the participant's comments. Section 7 presented the refined UI. Since the sessions were task-oriented, the analysis can be interpreted as a user study with expert-users.

9.2 User Study

After evaluating Outcome Explorer from the policy-makers' perspective through Heuristic Analysis, we conducted a user study with non-expert users to understand the policy-receivers' perspective.

We used Outcome Explorer as a design probe to understand the role of interpretability when policy-receivers interact with a system to receive automated decisions. Formally, we aimed at evaluating the following three hypotheses:

H1. The interpretable and interactive interface of Outcome Explorer will increase a user's efficiency in receiving and evaluating automated decisions.
H2. Outcome Explorer will increase a user's model understanding.
H3. Outcome Explorer will be simple and easy to use.

9.2.1 Participants

We recruited 10 participants (7 males, 3 females) through local mailing lists, social networks, and word-of-mouth. The participants varied in age from 19 to 35 ($M = 25$, $SD = 4.21$). None of the participants had domain expertise, except one who had a bachelor's degree in Computer Science. The participants were comfortable in using web technology and had a high-level idea of automated decision-making through exposure to credit-card approval and loan approval systems. Additionally, two participants had experience with interactive visualization through interactive online news.

9.2.2 Tasks

The tasks were designed to evaluate how Outcome Explorer contributes to a user's efficiency in receiving and evaluating automated decisions. Each participant was provided with a scenario and an initial profile. Upon receiving the decision on the initial profile, the participants were asked to obtain a series of alternative decisions (goals) by interacting with the interface while minimizing the number of changes and magnitude of changes. We anticipated that interpretability will help users understand the prediction mechanism and will decrease the number of changes and magnitude of changes made to achieve alternative decisions.

9.2.3 Study Design

We conducted a repeated-measures within-subject experiment. Our study had two conditions.

C1. Black-box: This prototype represented a black-box automated decision platform, where users could change variables using textboxes as well as horizontal sliders, to observe change in the outcome variable. This condition did not include any visual components or explanations.
C2. Outcome Explorer-Lite: This prototype included only the interactive path diagram with other components of Outcome Explorer removed.

We chose to include only the path diagram in C2 as we aimed at evaluating the effectiveness of interpretability. None of the removed components were necessary for the study and they would hinder a fair comparison between C1 and C2 as C1 did not include those components. This is also in line with the guideline (G3) presented in Section 4. To minimize the learning effect, we picked two datasets and counterbalanced the ordering of study conditions and datasets. The two datasets are the admission and housing dataset discussed above.

Similar to the heuristic analysis, we conducted the sessions via web and Skype. An experiment session began with the participant signing a consent form. Following this, the participants were introduced with the assigned first condition and received a brief description of the interface. The participants then interacted with the system (with a training dataset), during which they were encouraged to ask questions until they were comfortable. Each participant was then given a scenario and a task list for the first condition. After completing the tasks, the participants filled out a questionnaire. The same process was carried out for all the conditions. Each session lasted around $\sim$1.5 hours and ended with an exit-interview.

9.2.4 Results

Quantitative Measures. We measured three quantitative metrics: (1) time taken to complete the tasks, (2) number of changes, and (3) the magnitude of changes (%) made to reach the target outcomes. We performed a paired t-test between the study conditions for all three metrics.

We did not find any significant effect of study condition on the time taken to complete the tasks as the participants took an almost

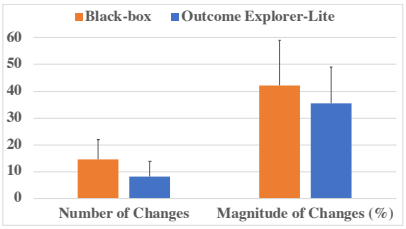

Fig. 10: The average number of changes and magnitude of changes (%) made to reach the target outcomes during the user study. Error-bars show +1 SD

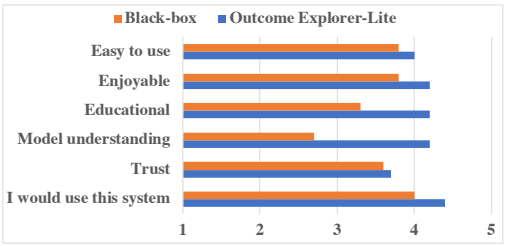

Fig. 11: Subjective Measures

similar amount of time on both conditions. The participants reduced the number of changes and the magnitude of those changes to reach the target outcomes while using Outcome Explorer-Lite compared to Black-box. On average, the number of changes were 14.6 ($SD = 7.245$) for Black-box, and 8.1 ($SD = 5.76$) for Outcome Explorer-Lite, as shown in figure 10. The difference was statistically significant with $p = 0.04$ (two-tailed). Figure 10 shows a similar result for the magnitude of changes (%), although the difference was not statistically significant. The results support H1 as the participants were able to reduce the number of changes and the magnitude of changes when using Outcome Explorer than using the Black-box condition, although the study conditions did not have a significant effect on the magnitude of changes.

Subjective Measures. Each participant rated the study conditions (interfaces) on a Likert scale ranging from 1 (Strongly Disagree) to 5 (Strongly Agree) based on 6 subjective metrics. We found a significant effect of study condition on "Model Understanding" ($p = 0.007$) and "Educational" ($p = 0.03$), meaning participants understood the model better and learned more about the overall process while using Outcome Explorer-Lite compared to Black-box (figure 11). No other comparisons were found statistically significant. The results support H2, but not H3, as the measure "Easy to use" had only a small difference between the study conditions.

10 Discussion and Limitation

10.1 RQ: XAI for Diverse Users

The user study presented in Section 9 provided promising results for an end-user oriented XAI system. The interpretable interface of Outcome Explorer helped them probe the model and achieve target outcomes efficiently. Expert-users also provided their feedback during and after the heuristic analysis session. When asked about the overall interface, one expert-user replied:

"The most impressive fact about the interface is that I can understand the model without solely relying on the counterfactual. I can actually see the model and I do not have to do any guesswork."

Another expert-user mentioned: *"I like the idea of integrating two profiles in the diagram (causal model). The comparison helped me understand how I am different than others and what I can or need to do to improve my chances."*

The heuristic analysis and the user study together show that Outcome Explorer can support the explanation needs of both expert and non-expert users (validating RQ). We believe an exhaustive evaluation with more analysis will validate our research even more. For example, our study did not include any posthoc explanation models. We concentrated primarily on evaluating the interpretability of our system as prior research already suggests explaining black-box models can complicate model understanding [40]. Nevertheless, such a condition would have revealed interesting insights and we intend to include that in our future work.

10.2 Interactivity, Interpretability, and Learning

On average, the participants spent a similar amount of time to complete the tasks in both user study conditions, but for different reasons. In the post-study interview, several participants mentioned that they felt curious, spent more time to learn the relations, and thought before taking an action while using Outcome Explorer. On the other hand, while using the Black-box interface, they felt directionless and tried to obtain the target outcomes mostly through random variable changes. We further observed that the participants spent more time on Outcome Explorer when the task domain was unfamiliar. A participant who is a senior college student mentioned:

"The interface (Outcome Explorer) is explanatory. I felt like I learned something. The interface is fun, attractive as well as educational. I did not know much about housing prices before this session. But, I think I now have a much better understanding of housing prices. If available in public when I buy a house in the future, it will help me make an informed decision."

Their comments were also reflected in the self-reported model understanding and learning scores (figure 11). Thus, interpretability coupled with interactivity helped them understand the decision process as well as gather knowledge from unfamiliar domains. Interestingly, interpretability did not increase users' trust in the system (figure 11), replicating the results found in [12].

10.3 Policy Refinement

During the user study, one participant asked whether the admission model belonged to a specific school. One expert-user during the heuristic analysis also expressed confusion about the structure of the housing model. We assured them that the structures are not absolute and that these are example models. Indeed, it makes sense to assume that different schools will have different admission policies. Our interface allows policy-makers to refine their policy as long as the model fit is acceptable. Interactive edge manipulation and latent factor creation allow policy-makers to alter certain problematic relations in the policy, even if the relations were historical.

10.4 Graphical Models

To the best of our knowledge, Outcome Explorer is the first-ever interactive interface for algorithmic decision-making based on a graphical model. Our findings suggest graphical representation is an effective way to convey inner-workings of the predictive models. Our design and visual encoding can be extended to other graphical models. For example, Bayesian Network is also represented as DAG, and Outcome Explorer can incorporate Bayesian networks without altering any of the front-end visual components.

10.5 Limitation

One limitation of the path diagram is that as the number of edges increases, the network can become increasingly difficult and complex to analyze visually, even for expert users. So the knowledge and insight gathering process might become difficult if the network becomes too complex. A long causal chain can also complicate model understanding. Our interface provides two methods (PCA and CFA) to handle this problem, but we have not evaluated our system on large datasets with hundreds of variables.

11 Conclusions

We presented Outcome Explorer— an interactive visual interface that exploits the explanatory power of the causal model and provides a visual design that can be extended to other graphical models. Outcome Explorer advances research towards interpretable interfaces and

provides critical findings through heuristic analysis, user study, and detailed discussion for future research.

REFERENCES

[1] Admission tracker, college data. https://www.collegedata.com/en/prepare-and-apply/admissions-tracker/get-started/. Accessed: 2020-04-20.

[2] Coronavirus tracking apps meet resistance in privacy-conscious europe. https://www.washingtonpost.com/world/europe/coronavirus-tracking-app-europe-data-privacy/2020/04/18/89def99e-7e53-11ea-84c2-0792d8591911_story.html note = Accessed:2020-04-29.

[3] Nhs rejects apple-google coronavirus app plan. https://www.bbc.com/news/technology-52441428 note = Accessed:2020-04-29.

[4] A. Abdul, J. Vermeulen, D. Wang, B. Y. Lim, and M. Kankanhalli. 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*, pp. 1–18, 2018.

[5] M. S. Acharya, A. Armaan, and A. S. Antony. A comparison of regression models for prediction of graduate admissions. In *2019 International Conference on Computational Intelligence in Data Science (ICCIDS)*, pp. 1–5. IEEE, 2019.

[6] S. Amershi, M. Chickering, S. M. Drucker, B. Lee, P. Simard, and J. Suh. Modeltracker: Redesigning performance analysis tools for machine learning. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, pp. 337–346, 2015.

[7] J. Angwin, J. Larson, S. Mattu, and L. Kirchner. Machine bias. *ProPublica, May*, 23:2016, 2016.

[8] P. M. Bentler and D. G. Weeks. Linear structural equations with latent variables. *Psychometrika*, 45(3):289–308, 1980.

[9] N. Blunch. *Introduction to structural equation modeling using IBM SPSS statistics and AMOS*. Sage, 2012.

[10] M. Bostock, V. Ogievetsky, and J. Heer. D^3 data-driven documents. *IEEE transactions on visualization and computer graphics*, 17(12):2301–2309, 2011.

[11] T. Brennan, W. Dieterich, and B. Ehret. Evaluating the predictive validity of the compas risk and needs assessment system. *Criminal Justice and Behavior*, 36(1):21–40, 2009.

[12] H.-F. Cheng, R. Wang, Z. Zhang, F. O'Connell, T. Gray, F. M. Harper, and H. Zhu. Explaining decision-making algorithms through ui: Strategies to help non-expert stakeholders. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, pp. 1–12, 2019.

[13] A. Chouldechova, D. Benavides-Prado, O. Fialko, and R. Vaithianathan. A case study of algorithm-assisted decision making in child maltreatment hotline screening decisions. In *Conference on Fairness, Accountability and Transparency*, pp. 134–148, 2018.

[14] T. N. Dang, P. Murray, J. Aurisano, and A. G. Forbes. Reactionflow: an interactive visualization tool for causality analysis in biological pathways. In *BMC proceedings*, vol. 9, p. S6. BioMed Central, 2015.

[15] B. J. Dietvorst, J. P. Simmons, and C. Massey. Overcoming algorithm aversion: People will use imperfect algorithms if they can (even slightly) modify them. *Management Science*, 64(3):1155–1170, 2016.

[16] F. Doshi-Velez and B. Kim. A roadmap for a rigorous science of interpretability. *arXiv preprint arXiv:1702.08608*, 2, 2017.

[17] N. Elmqvist and P. Tsigas. Animated visualization of causal relations through growing 2d geometry. *Information Visualization*, 3(3):154–172, 2004.

[18] C. Glymour, K. Zhang, and P. Spirtes. Review of causal discovery methods based on graphical models. *Frontiers in Genetics*, 10, 2019.

[19] B. Goodman and S. Flaxman. European union regulations on algorithmic decision-making and a "right to explanation". *AI magazine*, 38(3):50–57, 2017.

[20] D. Gotz and H. Stavropoulos. Decisionflow: Visual analytics for high-dimensional temporal event sequence data. *IEEE transactions on visualization and computer graphics*, 20(12):1783–1792, 2014.

[21] D. Harrison Jr and D. L. Rubinfeld. Hedonic housing prices and the demand for clean air. 1978.

[22] F. Hohman, A. Head, R. Caruana, R. DeLine, and S. M. Drucker. Gamut: A design probe to understand how data scientists understand machine learning models. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, pp. 1–13, 2019.

[23] F. Hohman, M. Kahng, R. Pienta, and D. H. Chau. Visual analytics in deep learning: An interrogative survey for the next frontiers. *IEEE transactions on visualization and computer graphics*, 25(8):2674–2693, 2018.

[24] F. Hohman, H. Park, C. Robinson, and D. H. P. Chau. S ummit: Scaling deep learning interpretability by visualizing activation and attribution summarizations. *IEEE transactions on visualization and computer graphics*, 26(1):1096–1106, 2019.

[25] M. Kahng, P. Y. Andrews, A. Kalro, and D. H. P. Chau. A cti v is: Visual exploration of industry-scale deep neural network models. *IEEE transactions on visualization and computer graphics*, 24(1):88–97, 2017.

[26] R. B. Kline. *Principles and practice of structural equation modeling*. Guilford publications, 2015.

[27] J. Krause, A. Perer, and K. Ng. Interacting with predictions: Visual inspection of black-box machine learning models. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, pp. 5686–5697, 2016.

[28] M. J. Kusner, J. Loftus, C. Russell, and R. Silva. Counterfactual fairness. In *Advances in Neural Information Processing Systems*, pp. 4066–4076, 2017.

[29] Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *nature*, 521(7553):436–444, 2015.

[30] Z. C. Lipton. The mythos of model interpretability. *Queue*, 16(3):31–57, 2018.

[31] T. Miller. Explanation in artificial intelligence: Insights from the social sciences. *Artificial Intelligence*, 267:1–38, 2019.

[32] J. Nielsen. Ten usability heuristics. https://www.nngroup.com/articles/ten-usability-heuristics/. Accessed:2020-04-29.

[33] J. Nielsen. Finding usability problems through heuristic evaluation. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 373–380, 1992.

[34] J. Nielsen and R. Molich. Heuristic evaluation of user interfaces. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 249–256, 1990.

[35] D. Norman. *The design of everyday things: Revised and expanded edition*. Basic books, 2013.

[36] Z. Obermeyer and S. Mullainathan. Dissecting racial bias in an algorithm that guides health decisions for 70 million people. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pp. 89–89, 2019.

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

[38] J. Pearl. *Causality*. Cambridge university press, 2009.

[39] J. Pearl. An introduction to causal inference. *The international journal of biostatistics*, 6(2), 2010.

[40] C. Rudin. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, 1(5):206–215, 2019.

[41] C. Schulz, A. Nocaj, J. Goertler, O. Deussen, U. Brandes, and D. Weiskopf. Probabilistic graph layout for uncertain network visualization. *IEEE transactions on visualization and computer graphics*, 23(1):531–540, 2016.

[42] R. E. Schumacker and R. G. Lomax. *A beginner's guide to structural equation modeling*. psychology press, 2004.

[43] B. Shneiderman. The eyes have it: A task by data type taxonomy for information visualizations. In *Proceedings 1996 IEEE symposium on visual languages*, pp. 336–343. IEEE, 1996.

[44] J. S. Tanaka. " how big is big enough?": Sample size and goodness of fit in structural equation models with latent variables. *Child development*, pp. 134–146, 1987.

[45] G. Vigueras and J. A. Botia. Tracking causality by visualization of multi-agent interactions using causality graphs. In *International Workshop on Programming Multi-Agent Systems*, pp. 190–204. Springer, 2007.

[46] P. Voigt and A. Von dem Bussche. The eu general data protection regulation (gdpr). *A Practical Guide, 1st Ed., Cham: Springer International Publishing*, 2017.

[47] J. Wang and K. Mueller. The visual causality analyst: An interactive interface for causal reasoning. *IEEE transactions on visualization and computer graphics*, 22(1):230–239, 2015.

[48] J. Wang and K. Mueller. Visual causality analysis made practical. In *2017 IEEE Conference on Visual Analytics Science and Technology (VAST)*, pp. 151–161. IEEE, 2017.

[49] Y. Wang, Q. Shen, D. Archambault, Z. Zhou, M. Zhu, S. Yang, and H. Qu. Ambiguityvis: Visualization of ambiguity in graph layouts. *IEEE transactions on visualization and computer graphics*, 22(1):359–368, 2015.

[50] J. Wexler, M. Pushkarna, T. Bolukbasi, M. Wattenberg, F. Viégas, and J. Wilson. The what-if tool: Interactive probing of machine learning models. *IEEE transactions on visualization and computer graphics*, 26(1):56–65, 2019.

[51] K. Wongsuphasawat and D. Gotz. Exploring flow, factors, and outcomes of temporal event sequences with the outflow visualization. *IEEE Transactions on Visualization and Computer Graphics*, 18(12):2659–2668, 2012.

[52] K. Wongsuphasawat, D. Smilkov, J. Wexler, J. Wilson, D. Mane, D. Fritz, D. Krishnan, F. B. Viégas, and M. Wattenberg. Visualizing dataflow graphs of deep learning models in tensorflow. *IEEE transactions on visualization and computer graphics*, 24(1):1–12, 2017.

[53] S. Wright. The method of path coefficients. *The annals of mathematical statistics*, 5(3):161–215, 1934.

[54] Y. Wu, L. Zhang, X. Wu, and H. Tong. Pc-fairness: A unified framework for measuring causality-based fairness. In *Advances in Neural Information Processing Systems*, pp. 3399–3409, 2019.

[55] J.-D. Zapata-Rivera, E. Neufeld, and J. E. Greer. Visualization of bayesian belief networks. In *Proceedings of IEEE Visualization'99, Late Breaking Hot Topics*, pp. 85–88, 1999.

[56] Z. Zhang, K. T. McDonnell, E. Zadok, and K. Mueller. Visual correlation analysis of numerical and categorical data on the correlation map. *IEEE transactions on visualization and computer graphics*, 21(2):289–303, 2014.

[57] J. Zhao, M. Glueck, S. Breslav, F. Chevalier, and A. Khan. Annotation graphs: A graph-based visualization for meta-analysis of data based on user-authored annotations. *IEEE transactions on visualization and computer graphics*, 23(1):261–270, 2016.