GAMUT: A Design Probe to Understand How Data Scientists Understand Machine Learning Models

Fred Hohman
Georgia Institute of Technology
Atlanta, GA, USA
fredhohman@gatech.edu

Andrew Head
UC Berkeley
Berkeley, CA, USA
andrewhead@berkeley.edu

Rich Caruana
Microsoft Research
Redmond, WA, USA
rcaruana@microsoft.com

Rob DeLine
Microsoft Research
Redmond, WA, USA
rob.deline@microsoft.com

Steven M. Drucker
Microsoft Research
Redmond, WA, USA
sdrucker@microsoft.com

ABSTRACT

Without good models and the right tools to interpret them, data scientists risk making decisions based on hidden biases, spurious correlations, and false generalizations. This has led to a rallying cry for model interpretability. Yet the concept of interpretability remains nebulous, such that researchers and tool designers lack actionable guidelines for how to incorporate interpretability into models and accompanying tools. Through an iterative design process with expert machine learning researchers and practitioners, we designed a visual analytics system, GAMUT, to explore how interactive interfaces could better support model interpretation. Using GAMUT as a probe, we investigated why and how professional data scientists interpret models, and how interface affordances can support data scientists in answering questions about model interpretability. Our investigation showed that interpretability is not a monolithic concept: data scientists have different reasons to interpret models and tailor explanations for specific audiences, often balancing competing concerns of simplicity and completeness. Participants also asked to use GAMUT in their work, highlighting its potential to help data scientists understand their own data.

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

ACM ISBN 978-1-4503-5970-2/19/05…$15.00
https://doi.org/10.1145/3290605.3300809

CCS CONCEPTS

• **Human-centered computing** → **Empirical studies in visualization**; *Visualization systems and tools*; • **Computing methodologies** → **Machine learning**.

KEYWORDS

Machine learning interpretability, design probe, visual analytics, data visualization, interactive interfaces

ACM Reference Format:
Fred Hohman, Andrew Head, Rich Caruana, Rob DeLine, and Steven M. Drucker. 2019. GAMUT: A Design Probe to Understand How Data Scientists Understand Machine Learning Models. In *CHI Conference on Human Factors in Computing Systems Proceedings (CHI 2019), May 4–9, 2019, Glasgow, Scotland UK*. ACM, New York, NY, USA, 13 pages. https://doi.org/10.1145/3290605.3300809

1 INTRODUCTION

With recent advances in machine learning (ML) [29, 37, 58, 65], people are beginning to use ML to address important societal problems like identifying and predicting cancerous cells [14, 32], predicting poverty from satellite imagery to inform policy decisions [27], and locating buildings that are susceptible to catching on fire [43, 59]. Unfortunately, the metrics by which models are trained and evaluated often hide biases, spurious correlations, and false generalizations inside complex, internal structure. These pitfalls are nuanced, particularly to novices, and cannot be diagnosed with simple quality metrics, like a single accuracy number [66]. This is troublesome when ML is misused, with intent or ignorance, in situations where ethics and fairness are paramount. Lacking an explanation for how models perform can lead to biased and ill-informed decisions, like representing gender bias in facial analysis systems [7], propagating historical cultural stereotypes in text corpora into widely used AI components [8], and biasing recidivism predictions by race [3]. This is the problem of *model interpretability*.

Although there is no formal, agreed upon definition of model interpretability [38], existing research focuses on human understanding of the model representation [4, 20, 45, 48, 54]. Government policy makers are also joining the discussion through the recent General Data Protection Regulation (GDPR) requirements [51]. Articles 13 and 22 state a "right to explanation" for any algorithm whose decision impacts a person's legal status [51].

To address model interpretability, a burgeoning research field of explainable artificial intelligence (AI) has emerged, whose general goal is to create and evaluate effective explanations for model decisions to better understand what a model has learned [23]. Recently, information visualization [9] has been used as an medium for explanation [1, 25, 41, 42]. This is a natural fit, since visualization and interactive visual analytics [13] excel at graphical communication for complex ideas and meaningful summarization of information. Model explanations come in many forms (e.g., textual, graphical), and two competing paradigms have emerged: global and local explanations. *Global explanations* roughly capture the entire space learned by a model in aggregate, favoring simplicity over completeness. Conversely, *local explanations* accurately describe a single data instance's prediction.

In this work, we take a human-centered approach to studying model interpretability. Through an iterative design process with expert machine learning researchers and practitioners at a large technology company, we designed Gamut, an interactive visual analytics system for model exploration that combines both global and local explanation paradigms. Using Gamut as a probe into interpretability, we conducted a user study to investigate why and how data professional data scientists interpret models and how interface affordances support data scientists in answering question about model interpretability. In designing our probe, we sought a balance between low graphicacy skills needed to learn about the model and a high level of accuracy so that users of the probe would trust its predictions were accurate and realistic. Therefore, we ground our research on a class of models, called generalized additive models (GAMs) [24], that perform competitively to state-of-the-art models yet contain a relatively simple structure [10, 39, 40, 63]. The study included 12 professional data scientists with ranging levels of expertise in machine learning. Our investigation shows that interpretability is not a monolithic concept: data scientists have different reasons to interpret models and tailor explanations for specific audiences, often balancing the competing concerns of simplicity and completeness. We also observed that having a tangible, functional interface for data scientists helped ground discussions of machine learning interpretability. Participants also asked to use Gamut in their work, highlighting its potential to help data scientists understand their own data. In this work, our contributions include:

- **A human-centered operationalization of model interpretability.** We contribute a list of capabilities that explainable machine learning interfaces should support to answer interpretability questions.
- **An interactive visualization system for generalized additive models (GAMs).** Gamut, an interactive visualization system built for exploring and explaining GAMs, iteratively designed with machine learning professionals.
- **A design probe evaluation with human subjects.** Results from a user study with professional data scientists using Gamut as a probe for understanding interpretability.

We hope the lessons learned from this work help inform the design of future interactive interfaces for explaining more kinds of models, including those with natural global and local explanations (e.g., linear regression, decision trees), as well as more complex models (e.g., neural networks).

2 RELATED WORK

Definitions of Interpretability

While existing definitions of interpretability center on human understanding, they vary in the aspect of the model to be understood: its internals [20], operations [4], mapping of data [48], or representation [54]. Hence, a formal, agreed upon definition remains open [15, 38]. These discussions make a distinction between *interpretability* (synonymous with explainability) and an *explanation*. An explanation is a collection of features from an interpretable domain that relate a data instance to a model's outcome [48, 54]. An explanation can be truthful or deceptive, accurate or inaccurate, all with varying degrees of success. Therefore, multiple explanations are often used to gain an ultimate interpretation of a model. Miller argues that interpretability research should leverage the literature from philosophy, psychology, and cognitive science for the history of how people define, generate, select, evaluate, and present explanations [45]. In this work, we build upon existing interpretability literature by using a human-centered approach to understand why data scientists need interpretability, how they use it, and how human-computer interaction (HCI) methods can help design interfaces to explain models.

Audience for interpretability. Recent work argues that the sophistication and completeness of both interpretability and explanations depends on the audience [20, 54]. Model builders may prefer global, aggregate model explanations; whereas, model users may prefer local, specific decision examples. Both will impact the interpretability of a system. Indeed, rather than considering interpretability as a monolithic concept, it may be more useful to identify properties that AI systems should obey to ensure interpretability, such as simulatabilitiy, decomposability, and algorithmic transparency [38].

Interpretability guidelines. The GDPR's recent declaration of the "right to explanation" [51] has sparked discussion for what this means in practice and what impact it will have on industry and research agendas [21]. While the updated version of the GDPR only requires explanation in limited contexts, AI and policy scholars expect explanations to be important in future regulations of AI systems [16]. Researchers have introduced a framework to turn the vague language of the GDPR into actionable guidelines, which include (1) identifying the factors that went into a decision, (2) knowing how varying a factor impacts a decision, and (3) comparing similar instances with different outcomes [16]. However, within this framework an AI-system need only satisfy one of the three above guidelines to be considered interpretable. Other useful post-hoc techniques for explaining decisions have also been proposed, such as using counterfactuals (that is, "What if" questions [62]), textual explanations, visualizations, local explanations, and representative examples of data [38]. We add to this existing work by contributing a list of capabilities that explainable interfaces should support to help people interpret models.

Visual Analytics for Explainable Machine Learning

Previous work demonstrates that interaction between users and machine learning systems is a promising direction for collaboratively sharing intelligence [60]. Since then, interactive visual analytics has succeeded in supporting machine learning tasks [25, 41, 42, 55, 63]. Example tasks include interactive model debugging and performance analysis [2, 44, 53], feature ideation and selection [6, 34], instance subset inspection and comparison [30, 31], model comparison [67], and constructing interpretable and trustworthy visual analysis systems [11].

Two visual analytics systems in particular are related to our work. Prospector [36] and Google's What-If Tool [50] use interactive partial dependence diagnostics and localized inspection techniques to allow data scientists to understand the outcomes for specific instances. These partial dependency charts are similar to the shape functions used in GAMs, explained later [47]. Both systems support using counterfactuals and modifying feature values on data instances to observe how changes could impact prediction outcome. In preliminary follow-up work, researchers investigated the effectiveness of providing instance explanations in aggregate, similarly identifying the distinction between global and local explanation paradigms [35]. We contribute to visual analytics literature by developing Gamut, an interactive visualization system used as a design probe to investigate how data scientists use global and local explanation paradigms.

Human Evaluation for Explainable AI

Human-centered machine learning recognizes that ML work is inherently human work and explores the co-adaptation of humans and systems [19]. Therefore, AI and ML systems should not only be developed with humans, but evaluated by humans. Unfortunately, the intrinsic probabilistic nature of ML models makes evaluation challenging. A taxonomy of evaluation approaches for interpretability includes application-grounded, human-grounded, and functionally grounded evaluations [15]. Our work falls into a human-grounded evaluation. Other studies have investigated the effectiveness of different explanations, taking initial steps toward identifying what factors are most important for providing human explanations [49]. Another study uses simulatability as the main task that human subjects perform to compare the trust humans have in white-box and black-box linear regression models [52]. Using human trust as a metric of evaluation for the effectiveness of explanations has also been studied [54]. However, simulatability and trust may not be ideal metrics to base evaluation on. An application-grounded evaluation for a pair of explainable ML interfaces deployed in the wild on a fraud detection team found that different explanation techniques yield widely varying results, yet are still considered reasonably valid and useful [12]. This is troublesome when in the case of incongruency domain experts were unaware of explanation disagreements and were eager to trust any explanation provided to them [12].

3 DESIGN RATIONALE

A Technology Probe for Model Interpretability

A technology probe is an "instrument that is deployed to find out about the unknown—returning with useful or interesting data," and should balance three broad goals: *design:* inspire reflection on emerging technologies; *social science:* appreciate needs and desires of users; and *engineering:* field-testing prototypes [26]. Technology probes are a common approach for contextual research in human-computer interaction that invite user participation [18, 22].

While building and deploying ML models is now a standard software practice, interpreting models is not. We therefore use a technology probe to understand this emerging practice, balancing these three goals:

- *Engineering:* we iteratively developed an explainable interface that works on real data and models.
- *Social science:* we used qualitative methods for data collection to learn about data scientists' behavior during an in-lab user study and quantitative measures for a preliminary usability assessment.
- *Design:* the visualization prototype inspired participants to reflect on interpretability and how they use it in their own work.

Assessing the Probe's Features

We took two approaches to design a visualization system to probe machine learning interpretability. First, we performed a literature survey to compare the many definitions of what makes a machine learning model interpretable. We focused on recent work that postulates interactive explanations will be key for understanding models better, as summarized in section 2. Second, we conducted a formative study through a series of interviews with both machine learning researchers and practitioners to gather questions a user should be able to ask a machine learning model or AI-powered system. The participants included 4 senior ML researchers and 5 ML practitioners (3 female and 6 male), who were recruited based on their expertise in ML and their interest in ML interpretability. Together, we synthesized our findings into the following list of capabilities that an explainable machine learning interface should support. While there is no guarantee of completeness, we, the authors and participants, find this list to be effective for operationalizing interpretability in explainable ML interfaces. Each capability provides an example interpretability question, which all reference a real-estate model that predicts the price of homes given the features of a house.

C1. Local instance explanations. PREDICTION
Given a single data instance, quantify each feature's contribution to the prediction.
Example: Given a house and its predicted price of $250,000, what features contributed to its price?

C2. Instance explanation comparisons. PREDICTION
Given a collection of data instances, compare what factors lead to their predictions.
Example: Given five houses in a neighborhood, what distinguishes them and their prices?

C3. Counterfactuals. PREDICTION
Given a single data instance, ask "what-if" questions to observe the effect that modified features have on its prediction.
Example: Given a house and its predicted price of $250,000, how would the price change if it had an extra bedroom?
Example: Given a house and its predicted price of $250,000, what would I have to change to increase its predicted price to $300,000?

C4. Nearest neighbors. DATA
Given a single data instance, find data instances with similar features, predictions, or both.
Example: Given a house and its predicted price of $250,000, what other houses have similar features, price, or both?
Example: Given a house and a binary model prediction that says to "buy", what is the most similar real home that the model predicts "not to buy"?

C5. Regions of error. MODEL
Given a model, locate regions of the model where prediction uncertainty is high.
Example: Given a house price prediction model trained mostly on older homes ranging from $100,000 - $300,000, can I trust a model's prediction that a newly built house costs $400,000?

C6. Feature importance. MODEL
Given a model, rank the features of the data that are most influential to the overall predictions.
Example: Given a house price prediction model, does it make sense that the top three most influential features should be the square footage, year built, and location?

Selecting the Probe's Model Class

Given the set of capabilities we uncovered during our formative study, our probe should work with a class of ML models having many ideal characteristics:

- The model should have a simple enough structure to allow the user to see the model globally.
- Understanding the model's computation should require average math skills, to support non-expert users.
- Similarly, visualizing the model's structure should require average graphicacy, i.e., data visualization literacy.
- The model should be compositional, so that the effect of features can be understood in isolation.
- The model should have high accuracy, so that deploying it is realistic.

Of course, no single class of model can be optimal for all these attributes [23]. For example, simpler models, like linear regression and decision trees, have simple global structure, but suffer from poor accuracy; more complex models, like deep neural networks, achieve superior performance at the cost of complex structure and lack of clear compositionality [10, 21, 54]. Our choice of model for the probe therefore represents a compromise among these criteria.

In essence, we sought a balance between low graphicacy skills needed to learn about the model and a high level of accuracy so that users of the probe would trust its predictions were accurate and realistic. One particular model class, the *generalized additive model* (GAM) [24], has recently attracted attention in the ML community. Thanks to modern ML techniques such as boosting [56], GAM performance on predictive tasks on tabular data competes favorably with more complex, state-of-the-art models, yet GAMs remain intelligible and more expressive than simple linear models [10, 39, 40]. Understanding a GAM requires only the ability to read a line chart. A GAM has a local explanation similar to linear regression, but also lends itself to a global explanation (shape function charts, described later), which other models lack; this allows us to test the relative value

users place on having global understanding versus a purely local understanding of a model.

GAMs are a generalization of linear models. To illustrate the difference, consider a dataset $D = \{(\mathbf{x_i}, y_i)\}^N$ of N data points, where $\mathbf{x_i} = (x_{i1}, x_{i2}, \ldots, x_{iM})$ is a feature vector with M features, and y_i is the target, i.e., the response, variable. Let x_j denote the jth variable in feature space. A typical linear regression model can then be expressed mathematically as:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_N x_N$$

This model assumes that the relationships between the target variable y_i and features x_j are *linear* and can be captured in slope terms $\beta_1, \beta_2, \ldots, \beta_N$. If we instead assume that the relationship between the target variable and features is *smooth*, we can write the equation for a GAM [24]:

$$y = \beta_0 + f_1(x_1) + f_2(x_2) + \cdots + f_N(x_N)$$

Notice here that the previous slope terms $\beta_1, \beta_2, \ldots, \beta_N$ have been replaced by smooth, shape functions f_j. In both models β_0 is the model intercept, and the relationship between the target variable and the features is still additive; however, each feature now is described by one shape function f_j that can be nonlinear and complex (e.g., concave, convex, or "bendy") [28].

Since each feature's contribution to the final prediction can be understood by inspecting the shape functions f_j, GAMs are considered intelligible [10]. In this paper, we omit the details of how to train GAMs, mean center shape functions, and distinguish their regression and classification versions, which are covered in the literature [39, 40, 57, 64]. We also note that GAM shape function charts differ from partial dependency (PD) [17] used in [36, 50]. PD assumes that features are uncorrelated, and PD averages over the other features not included in the chart. Therefore, PD only captures the effect of modifying one feature independent of the others, whereas GAM shape function charts, which are trained in parallel, are effectively the entire model—predictions are made by summing values from all charts together and take into account correlation among features to prevent multiple counting of evidence. All together, this makes GAMs uniquely suited as a model that maximizes our previous criteria and ties global and local explanations closely together.

4 GAMUT

Given the capabilities described in section 3, we present GAMUT, an interactive visualization system that tightly integrates three coordinated views to support exploration of GAMs (Figure 1): the Shape Curve View (A); the Instance Explanation View (B); and the Interactive Table (C). To explain these views, we use an example real-estate model that uses a house's features to predict its sale price in US dollars. The three views show different aspects of a user-selected instance, in this case a chosen house. Throughout the description we link features to the capabilities **(C1)**–**(C6)** that the features support.

Shape Curve View

The Shape Curve View displays each feature's shape function as a line chart (Figure 1A). The user can choose which features are displayed through the Feature Sidebar (Figure 2A): an ordered list the features of the data, sorted by importance to the model **(C6)**. We will first describe the encoding for one shape function chart. Consider the *OverallQual* feature and its shape function chart (Figure 2B). This chart shows the impact that the *OverallQual* feature has on the overall model predictions **(C6)**. The x-axis is the dimension of the feature, in this case, a rating of the house's overall material and finish quality, between 2 and 10; the y-axis is the contribution of the feature to the output of a prediction, in this case, US dollars. The chart shows that having a rating of 9 adds $50,000 to the predicted price, for example. Below the x-axis is a histogram of the data density for the dimension. This is useful for determining how many data points exists in a particular part of feature space **(C5)**, e.g., in Figure 2B, we see that most houses have a *OverallQual* of 5 to 8.

The selected instance's specific feature values are shown as amber points on the shape function charts **(C1)**. A data instance has one value for every feature, i.e., one amber point on each shape function chart, which shows where the selected instance is located in the global model **(C5)**. The color of the line for each shape function encodes the final predicted value if we were to vary the selected amber point's value to all other possible values. This is reinforced when a user brushes over a line chart: a new point, colored by its final prediction, is shown on the shape function curve, while projected crosshairs track with the mouse cursor, enabling users to ask interactive counterfactuals for any feature **(C3)**.

Since the Shape Curve View shows multiple shape function charts at once, we provide a Normalize toggle for accurate comparison. Turning Normalize on plots all the shape functions on a common scale, allowing visual comparison of the features' different degrees of impact on the predictions. Charts with high slopes indicate more impact on predictions, whereas charts with relatively flat lines contribute only a little **(C6)**. Turning off Normalize plots each chart on its own scale, emphasizing the shape of low-impact (flat) features.

Instance Explanation View

The Instance Explanation View shows a visualization of individual instance predictions (Figure 1B) **(C1)**. A GAM converts each feature value of a data instance into its direct contribution on the final prediction. Since GAMs are additive models, to obtain a prediction for a single data instance with M features, we compute the amount each feature contributes to

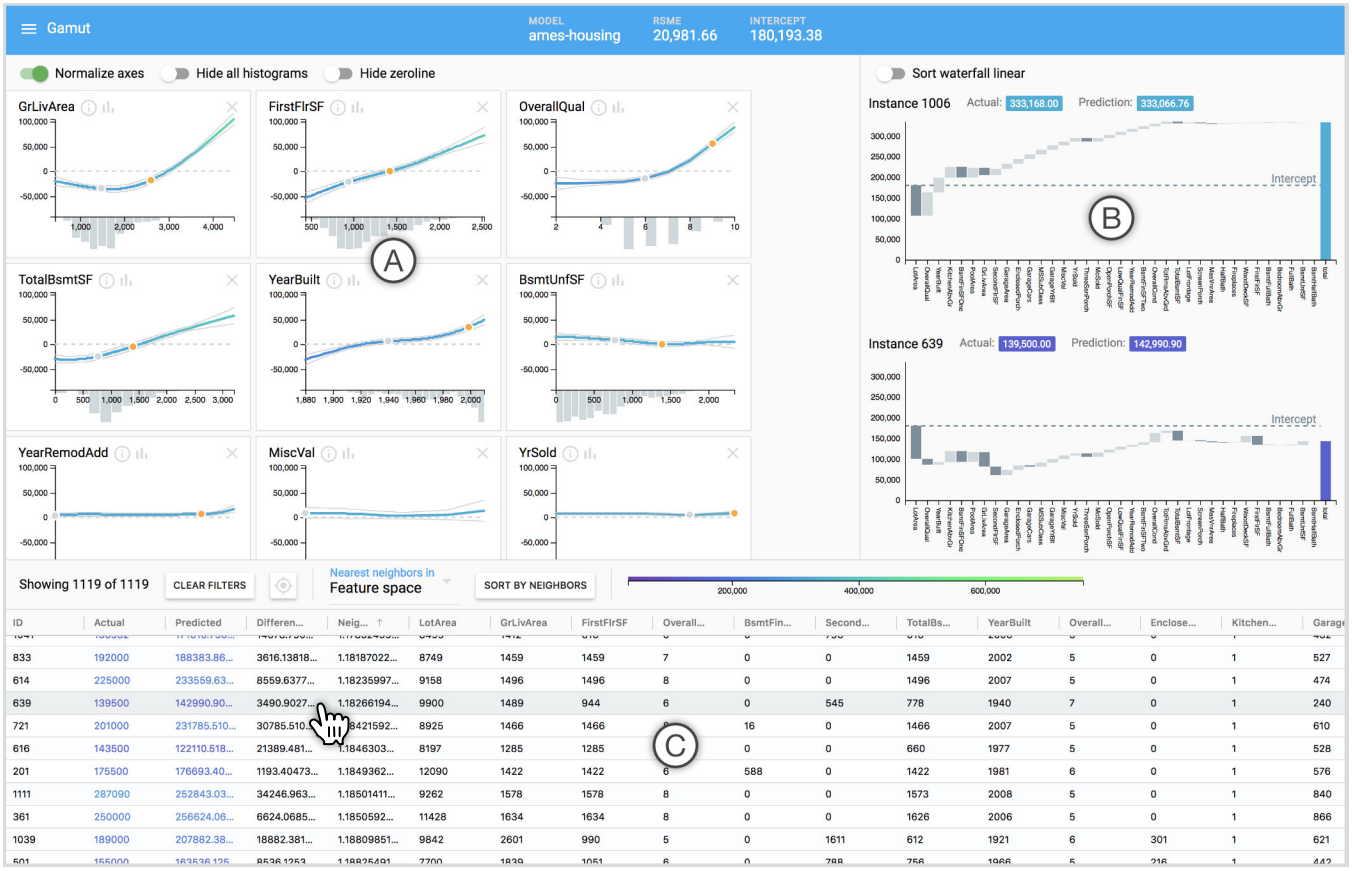

Figure 1: The Gamut user interface tightly integrates multiple coordinated views. (A) The Shape Curve View displays GAM shape functions as line charts, and includes histograms of the data density for each feature. The charts can be normalized to better compare the impact each shape function has on the model. **(B) The Instance Explanation View** displays a waterfall chart for two data instances. Each chart encodes the cumulative impact each feature has on the final prediction for one data instance. **(C) The Interactive Table** displays the raw data in an interactive data grid where users can sort, filter, and compute nearest neighbors for data instances.

the total prediction and add them all up. We also add the intercept (the average predicted value for the dataset), for a total of $M + 1$ values. The Instance Explanation View shows these $M + 1$ values as a waterfall chart **(C1)**. The x-axis is a categorical axis of all the features, and the y-axis is the final prediction. These values can be positive or negative, as indicated by the dark and light gray shades of each of piece of the waterfall chart. The x-axis is sorted by the absolute value of each feature's contribution; the leftmost values drive the majority of the overall prediction. For example, consider the waterfall chart in Figure 2C for Instance 550. From the colored tag, we see this house was predicted as costing $190,606. We also see the first three features greatly reduce the price of the house (three dark gray rectangles), but the next four increase the price. Another interesting characteristic is the long tail of features towards the end of the waterfall chart; a single feature value hardly contributes to the over prediction alone, but together the small contributions account for a non-trivial amount of the final prediction.

The Instance Explanation View also allows easy comparison of multiple instances (Figure 2C). The first chart is the selected instance, which is pinned to the interface. This selected instance's values are the same amber dots in the Shape Curve View. The second chart visualizes a different instance that updates as the user brushes over a different data instance from the Interactive Table, described in the next subsection. Since two instance predictions could have a different x-axis ordering, we impose the ordering of the selected instance on the second instance. Combined with automatically normalizing both y-axes for the two waterfall charts, this enables direct comparison of both waterfall charts **(C2)**.

Brushing over either waterfall chart provides several cues to aid comparison: a tooltip with the exact feature value and GAM contribution for both waterfall charts (Figure 2C)

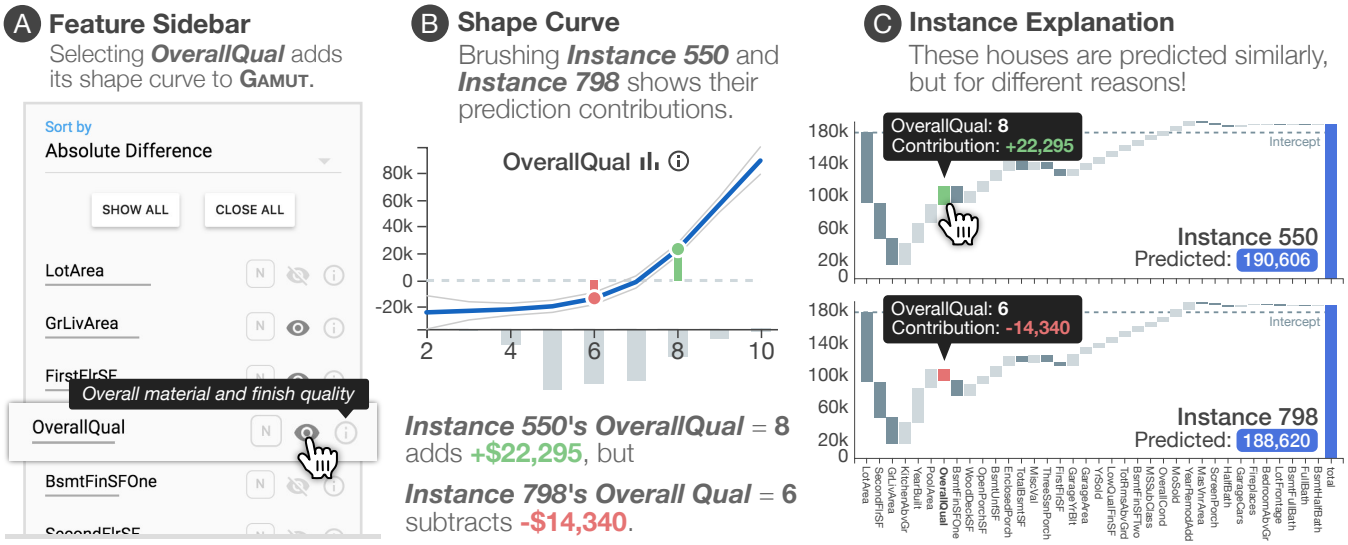

Figure 2: Interacting with GAMUT's multiple coordinated views together. (A) Selecting the *OverallQual* feature from the sorted Feature Sidebar displays its shape curve in the Shape Curve View. **(B)** Brushing over either explanation for *Instance 550* or *Instance 798* shows the contribution of the *OverallQual* feature value for both instances. **(C)** Notice these two houses are similarly predicted ($190,606 and $188,620), but for different reasons!

(C1); highlights in the corresponding shape function charts in the Shape Curve View; and plotting both points on the shape function charts (Figure 2B) **(C2)**. The two instances, i.e., houses, shown in Figure 2C are close in predicted price, ($190,606 and $188,620), and have similar shapes in their waterfall charts. However, Instance 550 has an *OverallQual* of 8 which adds +$22,295 to the prediction cost; whereas, Instance 798 has a *OverallQual* of 6 which reduces the cost -$14,340. While a few other values must differ to make up for this particular difference, we have found two houses that are predicted with similar prices, but achieve those prices by different means **(C4)**.

Interactive Table

The Interactive Table is a scrollable data grid of the raw data used to train the model (Figure 1C). The rows of the data grid are individual data instances, and the columns are the features, plus five additional columns on the left: Instance ID; Actual value (or label) of the data instance; Predicted value (or label) of the data instance; Difference between actual and predicted value; and Nearest Neighbor Distance from the selected instance. The column headers provide familiar data grid features, like resizing, sorting, and filtering columns.

Brushing over a row in the Interactive Table updates the second waterfall chart in the Instance Explanation View and normalizes both waterfall charts to ensure direct comparison between the two visualized instances is accurate **(C2)**. Brushing over a row also plots that instance's values on the Shape Curve View as gray points to compare against the selected instance's amber points described above **(C2)**.

Implementation

GAMUT is a client-side web app, using D3 [5] for visualization and ag-Grid[1] for the data grid. We pre-train our GAMs in Python using the pyGAM [57] package. pyGAM uses splines to fit the GAM shape curves; however, more advanced techniques exist for training GAMS as cited in section 3.

5 USER STUDY

We used GAMUT as a design probe during an in-lab study to understand how data scientists understand machine learning models and answer interpretability questions. We aimed to answer the following research questions:

RQ1. Why do data scientists need interpretability and how do they answer interpretability questions?

RQ2. How do data scientists use global explanations and local explanations?

RQ3. How does interactivity play a role in explainable machine learning interfaces?

Participants

We invited 200 randomly selected professional data scientists at a large technology company and received 33 replies (17% response rate). We selected 12 participants (7 female, 5 male),

[1]https://www.ag-grid.com/

all with bachelor's degrees, 6 with graduate degrees. Half of the participants had only 1 year of experience with ML, while the other half had at least 3 years, with two participants having more than 5 years. One participant uses ML on a daily basis, five on a weekly basis, while the other six use ML less often. Ten of the participants reported they use visualization in their work, mostly dashboard-style analytics. Nine participants reported using tabular data in their own work. Six of participants reported that they have used explanations for models before; five said their explanations were static, with only one reporting their explanation being interactive. We compensated participants with a $25 Amazon Gift card.

Study Design

The study duration was 1½-hours per participant. To start, each participant signed a consent form and filled out a background questionnaire. The session then consisted of a GAMUT tutorial, with a model that predicts the price of 1,000 diamonds, based on 9 features.

Participants thought aloud while using GAMUT to explore two models, one regression and one binary classification. Participants were free to choose one of three regression models that predict: the price of 506 houses in Boston, Massachusetts, based on 13 features (6 chose this); the price of 1,119 houses in Ames, Iowa, based on 36 features (5 chose this); or the quality of 1,599 wines, based on 11 features (1 chose this). Similarly, participants were free to choose one of three binary classification models that predict: the survival of 712 Titanic passengers, based on 7 features (4 chose this); heart disease in 261 patients, based on 10 features (5 chose this); or diabetes in 392 patients, based on 8 features (2 chose this).

Once a participant chose a dataset, we provided them with the feature names and their textual descriptions. We then gave them 5 minutes to brainstorm their own hypotheses about the model, using their own intuition. We then allowed them to use GAMUT to explore the model, guided by a list of questions we provided (≈10 per dataset) that exercise GAMUT's capabilities, ordered so that adjacent questions test different capabilities. All participants completed all the questions for one model in the allotted time, around 15 minutes. If they had not already addressed their initial questions, we returned to them to see if they were able to after. We then repeated this process for the second dataset. Each session ended with a usability questionnaire and an exit interview that asked participants to reflect on their process of explaining ML models in their own work, their process of using GAMUT, and if GAMUT could be useful for them.

6 RESULTS

Every participant was successful at answering both their own and our prepared questions about the different models, despite being new to GAMs and GAMUT. We also observed that having a tangible, functional interface for data scientists helped ground the discussion of interpretability. In the following sections we summarize the results from our study, both during the participant usage of GAMUT and the conversations during the exit interviews.

RQ1: Reasons for Model Interpretability

Hypothesis generation. As participants used GAMUT, they constantly generated hypotheses about the data and model while observing different explanations. This was insightful, since after only a brief tutorial, the participants were comfortable answering a variety of questions about the models and started to reason about them in ways they could not before. We also noticed that participants were using the model to confirm prior beliefs about the data, slowly building trust that the model was producing accurate and believable predictions. However, participants were eager to rationalize explanations without first questioning the correctness of the explanation itself. While forming new hypotheses about one's data and model can lead to deeper insight, this could be troublesome when participants trust explanations without healthy skepticism. While these results corroborate existing literature [12, 35], it suggests further studies to evaluate human trust in model explanations.

Data understanding. Participants also used interpretability as a lens into data, which prompted us to ask participants about this during the exit interviews. While a predictive model has its own uses, e.g., inference and task automation, many participants explained that they use models to gain insight into large datasets, as mentioned in [33]. One participant said, *"It's more like a data digging process. So it's finding the important features to help us understand the data better."* While there are many academic and commercial tools for data exploration without statistical models, a model-based approach gave participants a new perspective on the data. About GAMUT, one participant said, *"This would help me and expedite my workflow to get to valuable nuggets of information, which is what [my stakeholders] are ultimately interested in."* Related, another reason that emerged from the interviews was that data scientists use interpretability to understand the feature importance of a dataset. Most of our participants said that computing a metric (for which there are many) for feature importance across all features provides valuable information about what characteristics of a dataset are most important for making predictions. This allows data scientists to focus on accurately representing these features in a model. With regards to learning representations, a few participants said that interpretability also ensures customer privacy is upheld, by discovering what features are correlated with identifiable information so they can be removed.

Communication. Throughout the study, the prepared questions asked participants to communicate their process of discovering the answers. During the exit interviews, nearly every participant described a scenario in which they were using model explanations to communicate what features were predictive to stakeholders who wanted to deploy a model in the wild. One participant noted that *"different audiences require different explanations,"* describing a common trade-off between explanation simplicity and completeness. This was further supported by a participant who frequently presents reports to stakeholders: *"When you're going to craft your story, ...you're going to have to figure out what you want emphasize and what you want to minimize. But you have to always lay out everything. Know your audience and purpose."* She also emphasized that she encourages fellow data scientists on her team to share knowledge about what they have learned to other non-scientists. Lastly, a participant said she uses explainable data analysis to change organizational behavior on her team, by using models to inspect and understand data quality. She described how some analysts claim they can predict a value, but neglect to explain why, which diminishes the impact: *"What are the features? How are you getting those features? What are the quality of those features? They're just literally saying, 'I'm forecasting the number—here's the number you use.' I'm going, 'That just is not satisfying.'"* By using feature importance metrics, she ensures that the important features of data are accurately collected, recognizing that "clean" data creates better models.

Model building. Participants who have experience in developing models recognize that interpretability is also critical to model builders. Understanding characteristics about one's data and model helps guide model improvement. Regarding the intelligibility versus accuracy trade-off, one participant said that he starts his work using simpler models to become familiar with the data, before moving onto more complex models. Having a solid understanding of one's data is more important than incrementally improving model accuracy: *"I want to understand bit by bit how the dataset features work with each other, influence each other. That is my starting point."* Another participant said his team uses two natural language processing models in production: a simpler, rule-based model that performs multiple checks before inference; if the checks pass, the data is passed to the more complex model for a final prediction.

RQ2: Global versus Local Explanation Paradigms

While using Gamut, every participant used both the global and local explanations to answer interpretability questions, often moving between the two. This shows that global and local explanation paradigms are in fact complementary. Participants used the shape function charts of the model to explain a feature of the dataset, but grounded the explanation with local context using the data histogram. Conversely, participants described single-instance explanations using the global context of the shape function charts, i.e., overlaying the amber points of a waterfall chart on shape function charts. One participant said, *"If I want to see what the overall ecosystem is doing, [global explanations are] significantly better. If I wanted to find specific use cases that are interesting, then I'm going to use [local explanations] as case studies. So, I see it as having both."*

Broadly speaking, we noticed the expertise of a participant correlated with which explanation paradigm they preferred: (1) the ML novices gravitated towards the local explanations, (2) more expert participants used global explanations more frequently, and (3) the most expert participants fluidly used both to reason about a prediction and a model. For example, a common practice in ML is to consider only the top features, since likely those are driving the prediction. However, one participant noticed that the visualizations in the Instance Explanation View argued otherwise—the long tail of a waterfall chart sometimes contributed a non-trivial percentage of a prediction—and observed that the top features were insufficient. This is an interesting example of how a local explanation can inform a global characteristic of a model.

The Interactive Table was a critical mechanism for linking global and local explanations. Participants frequently sorted columns (i.e., features) to see how data aggregates along a single feature, but also inspected many single data instances for exact feature values; to our surprise, sorting by nearest neighbors was only used a couple times per participant. Some participants were initially confused about whether a particular visualization was describing global or local model behavior (e.g., mistaking a waterfall chart to describe the global behavior of a model instead of a single data instance), suggesting that either the initial tutorial could be improved, or that the level of graphicacy required for Gamut was higher than anticipated; regardless, by the end of every 1½-hour session, it was clear all participants understood how Gamut's representations connected together.

RQ3: Interactive Explanations

When choosing a model explanation, regardless of the type (e.g., textual, graphical), most explanations are static. Only recently has the notion of *interactive explanations* attracted attention. In Gamut, interactivity refers to instance-based selection, brushing and linking between local and global views, quick comparison of instances and their explanations, sorting and filtering the Interactive Table, hovering over a shape function chart for asking counterfactuals, and computing nearest neighbors for a single instance.

Throughout the studies it became clear that interactivity was the primary mechanism for exploring, comparing, and

explaining instance predictions and the chosen models by the participants. Interactivity was so fundamental for our participants' understanding of the models, that when we prompted them to comment on interactivity, people could not conceive non-interactive means to answer both their hypotheses and prepared questions, even though the current best practice for understanding GAMs entails flipping through static print outs of all the shape function charts.

Participants liked the interactivity of Gamut, but we think there is potential to alleviate redundant interactions by incorporating automated insight discovery techniques in explanation systems. Examples include algorithmically surfacing the most accurate explanations and finding the most relevant data (e.g., similar neighbors, counterfactuals) given interpretability-focused constraints.

Participants also suggested several additional features. First, while Gamut supports comparing two instance explanations at once, participants wanted to compare multiple groups of instances (e.g., user-defined groups, or a group of nearest neighbors); they also wanted deeper comparison, such as changing the visual representation to a stacked bar chart to more easily compare the contributions of multiple instance by feature. Second, the more expert participants wanted more support for feature selection and importance, such as leaving one feature out of the model and seeing its effect on performance. Lastly, we noticed most participants used counterfactuals often throughout their exploration, both as a direct task and as a sanity check for feature sensitivity; therefore, there could be opportunities to support automatic counterfactual identification in combination with computing nearest neighbors to enable data scientists to understand models faster and more confidently.

Usability

The exit questionnaire included a series of Likert-scale (7 point) questions about the utility and usefulness of the various views in Gamut (Figure 3). From the high ratings, we are confident that Gamut's role as a design probe was not hampered by usability problems. Similarly, the uniformity of the feature ratings suggests that participants did not disfavor any particular feature because of a usability problem.

Even though Gamut was designed as a probe, all 12 participants desired to use it to understand their own data. Some participants suggested using the system in its entirety, while others wanted to use specific parts of the interface, such as the Instance Explanation View, to include in reports to their stakeholders. One participant who frequently uses visual analytics tools said, *"I really like that it's splitting out each of the individual features into its own chart. ...I can't tell you how useful that is for me. Parameterizing dimensions is just not available with Tableau, Power BI, or anything else."* Another participant wanted to use Gamut to not only predict when

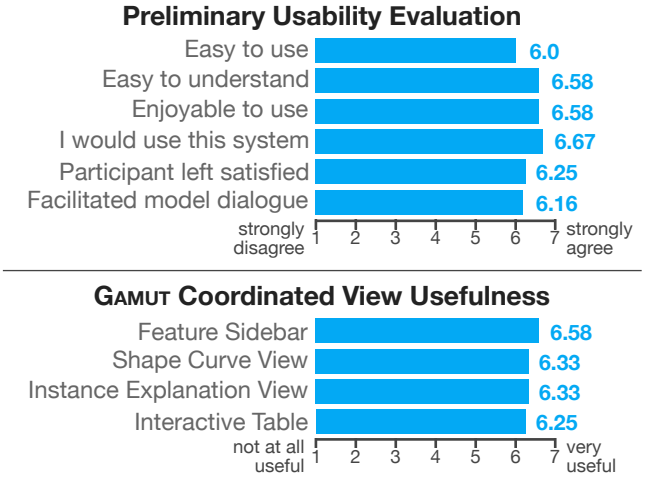

Figure 3: Gamut subjective ratings. In a preliminary usability evaluation, participants thought Gamut was easy to use and enjoyable. Of Gamut's multiple coordinated views, all were rated favorably. This also supports our finding that both global and local explanations are valuable for understanding a model's behavior.

customers would renew a product subscription, but to understand why and how they renew. A participant who frequently engages with legal discourse suggested a potential user for Gamut that we had not considered: *"I definitely would use something like this, especially when it comes to privacy issues. I even would show this to lawyers."* Several participants have followed up after the conclusion of the study and actively pursued using Gamut in their teams with their own data.

7 LIMITATIONS

Gamut only visualizes one class of ML model. While Gamut's design rationale, visualizations, and interactions were informed by multiple interviews and collaboration with ML researchers and practitioners, there could be an another complementary view that could have elicited better qualitative results during our user study. Regardless, to the best of our knowledge there is no existing interactive interface for GAMs. We think Gamut is a useful interface for exploring GAMs, as supported by our usability ratings in section 6 and participants desire to use Gamut for their own work, perhaps by using GAMs to explain more complex models, as discussed in the following section.

Understanding a model's domain likely provides an advantage to understanding how a model works. Different participants entered the study with different domain knowledge. To mitigate this risk, we both provided a variety of models about approachable topics and allowed participants to choose the models that fit their own knowledge and expertise.

More technically, one participant with a PhD in statistics was concerned about correlated features and suggested that interaction terms should be considered. We discuss this implication in the following section.

8 FUTURE RESEARCH DIRECTIONS

Through Gamut and our user study, we suggest the following set of future directions for improving interactive interfaces for understanding machine learning models:

Integrating better GAMs. ML researchers are developing a new GAM extension called GA^{2}Ms that includes interaction variables that are showing even better performance [10]. However, visualizing interaction shape function charts, which are 2D surfaces instead of 1D lines, is an open design challenge, especially when visualizing their error surfaces (analogous to confidence intervals for the 1D case). These interaction shape function surface explanations will also require a higher level of statistics and graph literacy in users.

Using GAMs to explain and compare other models. We have shown the power of intelligibility of GAMs, including the valuable combination of global and local explanations, over other more complex models, such as random forests or deep neural networks. However, more complex models are still used in practice. Using a GAM to model one of these more complex models could be a promising approach for bringing the intelligibility of GAMs to more performance-focused models. Existing work supports this idea by using surrogate models for improved interpretability, for example, employing model distillation [61] or visualizing extracted rule-based knowledge representations [46]. GAMs could also help data scientists explore multiple models at once, since multiple shape function charts for the same feature can be overlaid, enabling direction comparison between differing models.

Scalability. The six datasets from our study are considered small by current ML standards. Preliminary work has shown that as scale increases, interpretability and satisfaction decreases [49]. Therefore, it would be useful to see similar studies to ours use larger datasets to see how interpretability is affected by both the number of data points and the number of features. In Gamut's current design, the shape function charts scale well with the number of data points, but not with the number of features; the waterfall charts become harder to read as the number of features grows.

Supporting both explanation paradigms. Although different participants favored using different strategies, from our study we found that participants used both global and local explanations fluidly together, showing that these two paradigms are complementary. Therefore, future explainable systems and interactive interfaces should provide both model-level and instance-level explanations to flexibly support people's differing processes.

9 CONCLUSION

In this work, through an iterative design process with expert machine learning researchers and practitioners at a large technology company, we identified a list of explainable machine learning interface capabilities, designed and developed an interactive visualization system, Gamut, that embodied our capabilities, and used it as a design probe for machine learning interpretability through a human-subjects user study. Our results show that data scientists have many reasons for interpretability, answer interpretability questions using both global and local explanations, and like interactive explanations. Gamut's tightly interactive coordinated views enabled deeper understanding of both models and predictions. All participants wanted to use Gamut on their own data in the course of their every day work. From our study, it is clear there is a pressing need for better explanatory interfaces for machine learning, suggesting that HCI, design, and data visualization all have critical roles to play in a society where machine learning will increasingly impact humans.

ACKNOWLEDGMENTS

We thank Sarah Tan, Jina Suh, Chris Meek, Duen Horng (Polo) Chau, and the anonymous reviewers for their constructive feedback. We also thank the data scientists at Microsoft who participated in our interviews and studies. This work was supported by a NASA Space Technology Research Fellowship.

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 *ACM 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 *ACM Conference on Human Factors in Computing Systems*. ACM, 337–346.

[3] Julia Angwin, Jeff Larson, Surya Mattu, and Lauren Kirchner. 2016. Machine bias. *ProPublica, May* 23 (2016).

[4] Or Biran and Courtenay Cotton. 2017. Explanation and justification in machine learning: a survey. In *IJCAI Workshop on Explainable AI*.

[5] Michael Bostock, Vadim Ogievetsky, and Jeffrey Heer. 2011. D^3 data-driven documents. *IEEE Transactions on Visualization and Computer Graphics* 12 (2011), 2301–2309.

[6] Michael Brooks, Saleema Amershi, Bongshin Lee, Steven M Drucker, Ashish Kapoor, and Patrice Simard. 2015. FeatureInsight: visual support for error-driven feature ideation in text classification. In *IEEE Conference on Visual Analytics Science and Technology*. IEEE, 105–112.

[7] Joy Buolamwini and Timnit Gebru. 2018. Gender shades: intersectional accuracy disparities in commercial gender classification. In *ACM Conference on Fairness, Accountability and Transparency*. 77–91.

[8] Aylin Caliskan, Joanna J Bryson, and Arvind Narayanan. 2017. Semantics derived automatically from language corpora contain human-like biases. *Science* 356, 6334 (2017), 183–186.

[9] Mackinlay Card. 1999. *Readings in information visualization: using vision to think*. Morgan Kaufmann.

[10] Rich Caruana, Yin Lou, Johannes Gehrke, Paul Koch, Marc Sturm, and Noemie Elhadad. 2015. Intelligible models for healthcare: predicting pneumonia risk and hospital 30-day readmission. In *ACM International Conference on Knowledge Discovery and Data Mining*. ACM, 1721–1730.

[11] Jason Chuang, Daniel Ramage, Christopher Manning, and Jeffrey Heer. 2012. Interpretation and trust: designing model-driven visualizations for text analysis. In *ACM Conference on Human Factors in Computing Systems*. ACM, 443–452.

[12] Dennis Collaris, Leo M Vink, and Jarke J van Wijk. 2018. Instance-level explanations for fraud detection: a case study. *ICML Workshop on Human Interpretability in Machine Learning* (2018).

[13] Kristin A Cook and James J Thomas. 2005. *Illuminating the path: the research and development agenda for visual analytics*. Technical Report. Pacific Northwest National Lab. Richland, WA, USA.

[14] Joseph A Cruz and David S Wishart. 2006. Applications of machine learning in cancer prediction and prognosis. *Cancer Informatics* 2 (2006), 117693510600200030.

[15] Finale Doshi-Velez and Been Kim. 2017. Towards a rigorous science of interpretable machine learning. *arXiv preprint arXiv:1702.08608* (2017).

[16] Finale Doshi-Velez, Mason Kortz, Ryan Budish, Chris Bavitz, Sam Gershman, David O'Brien, Stuart Schieber, James Waldo, David Weinberger, and Alexandra Wood. 2017. Accountability of AI under the law: the role of explanation. *arXiv preprint arXiv:1711.01134* (2017).

[17] Jerome H Friedman. 2001. Greedy function approximation: a gradient boosting machine. *Annals of Statistics* (2001), 1189–1232.

[18] Bill Gaver, Tony Dunne, and Elena Pacenti. 1999. Design: cultural probes. *Interactions* 6, 1 (1999), 21–29.

[19] Marco Gillies, Rebecca Fiebrink, Atau Tanaka, Jérémie Garcia, Frédéric Bevilacqua, Alexis Heloir, Fabrizio Nunnari, Wendy Mackay, Saleema Amershi, Bongshin Lee, et al. 2016. Human-centred machine learning. In *ACM Conference Extended Abstracts on Human Factors in Computing Systems*. ACM, 3558–3565.

[20] Leilani H Gilpin, David Bau, Ben Z Yuan, Ayesha Bajwa, Michael Specter, and Lalana Kagal. 2018. Explaining explanations: an approach to evaluating interpretability of machine learning. *arXiv preprint arXiv:1806.00069* (2018).

[21] Bryce Goodman and Seth Flaxman. 2016. European Union regulations on algorithmic decision-making and a "right to explanation". *ICML Workshop on Human Interpretability in Machine Learning* (2016).

[22] Connor Graham and Mark Rouncefield. 2008. Probes and participation. In *Conference on Participatory Design*. Indiana University, 194–197.

[23] David Gunning. 2017. Explainable artificial intelligence (xai). *Defense Advanced Research Projects Agency (DARPA)* (2017).

[24] Trevor J Hastie and Robert Tibshirani. 1990. Generalized additive models. In *Chapman & Hall/CRC*.

[25] Fred 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).

[26] Hilary Hutchinson, Wendy Mackay, Bo Westerlund, Benjamin B Bederson, Allison Druin, Catherine Plaisant, Michel Beaudouin-Lafon, Stéphane Conversy, Helen Evans, Heiko Hansen, et al. 2003. Technology probes: inspiring design for and with families. In *ACM Conference on Human Factors in Computing Systems*. ACM, 17–24.

[27] Neal Jean, Marshall Burke, Michael Xie, W Matthew Davis, David B Lobell, and Stefano Ermon. 2016. Combining satellite imagery and machine learning to predict poverty. *Science* 353, 6301 (2016), 790–794.

[28] Kelvyn Jones and Simon Almond. 1992. Moving out of the linear rut: the possibilities of generalized additive models. *Transactions of the Institute of British Geographers* (1992), 434–447.

[29] Michael I Jordan and Tom M Mitchell. 2015. Machine learning: trends, perspectives, and prospects. *Science* 349, 6245 (2015), 255–260.

[30] Minsuk Kahng, Pierre Y Andrews, Aditya Kalro, and Duen Horng Polo Chau. 2018. Activis: Visual exploration of industry-scale deep neural network models. *IEEE Transactions on Visualization and Computer Graphics* 24, 1 (2018), 88–97.

[31] Minsuk Kahng, Dezhi Fang, and Duen Horng Polo Chau. 2016. Visual exploration of machine learning results using data cube analysis. In *Workshop on Human-In-the-Loop Data Analytics*. ACM.

[32] Konstantina Kourou, Themis P Exarchos, Konstantinos P Exarchos, Michalis V Karamouzis, and Dimitrios I Fotiadis. 2015. Machine learning applications in cancer prognosis and prediction. *Computational and Structural Biotechnology Journal* 13 (2015), 8–17.

[33] Josua Krause, Aritra Dasgupta, Jordan Swartz, Yindalon Aphinyanaphongs, and Enrico Bertini. 2017. A workflow for isual diagnostics of binary classifiers using instance-level explanations. *IEEE Conference on Visual Analytics Science and Technology* (2017).

[34] Josua Krause, Adam Perer, and Enrico Bertini. 2014. INFUSE: interactive feature selection for predictive modeling of high dimensional data. *IEEE Transactions on Visualization and Computer Graphics* 20, 12 (2014), 1614–1623.

[35] Josua Krause, Adam Perer, and Enrico Bertini. 2018. A user study on the effect of aggregating explanations for interpreting machine learning models. *ACM KDD Workshop on Interactive Data Exploration and Analytics* (2018).

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

[37] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*. 1097–1105.

[38] Zachary C Lipton. 2016. The mythos of model interpretability. *ICML Workshop on Human Interpretability in Machine Learning* (2016).

[39] Yin Lou, Rich Caruana, and Johannes Gehrke. 2012. Intelligible models for classification and regression. In *ACM International Conference on Knowledge Discovery and Data Mining*. ACM, 150–158.

[40] Yin Lou, Rich Caruana, Johannes Gehrke, and Giles Hooker. 2013. Accurate intelligible models with pairwise interactions. In *ACM International Conference on Knowledge Discovery and Data Mining*. ACM, 623–631.

[41] Junhua Lu, Wei Chen, Yuxin Ma, Junming Ke, Zongzhuang Li, Fan Zhang, and Ross Maciejewski. 2017. Recent progress and trends in predictive visual analytics. *Frontiers of Computer Science* 11, 2 (2017), 192–207.

[42] Yafeng Lu, Rolando Garcia, Brett Hansen, Michael Gleicher, and Ross Maciejewski. 2017. The state-of-the-art in predictive visual analytics. In *Computer Graphics Forum*, Vol. 36. Wiley Online Library, 539–562.

[43] Michael Madaio, Shang-Tse Chen, Oliver L Haimson, Wenwen Zhang, Xiang Cheng, Matthew Hinds-Aldrich, Duen Horng Chau, and Bistra Dilkina. 2016. Firebird: predicting fire risk and prioritizing fire inspections in atlanta. In *ACM International Conference on Knowledge Discovery and Data Mining*. ACM, 185–194.

[44] Sean McGregor, Hailey Buckingham, Thomas G Dietterich, Rachel Houtman, Claire Montgomery, and Ronald Metoyer. 2017. Interactive visualization for testing markov decision processes: MDPVIS. *Journal of Visual Languages & Computing* 39 (2017), 93–106.

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

[46] Yao Ming, Huamin Qu, and Enrico Bertini. 2019. RuleMatrix: visualizing and understanding classifiers with rules. *IEEE Transactions on Visualization and Computer Graphics* 25, 1 (2019), 342–352.

[47] Christoph Molnar. 2018. *Interpretable machine learning*. https://christophm.github.io/interpretable-ml-book/. https://christophm.github.io/interpretable-ml-book/.

[48] Grégoire Montavon, Wojciech Samek, and Klaus-Robert Müller. 2017. Methods for interpreting and understanding deep neural networks. *Digital Signal Processing* (2017).

[49] Menaka Narayanan, Emily Chen, Jeffrey He, Been Kim, Sam Gershman, and Finale Doshi-Velez. 2018. How do humans understand explanations from machine learning systems? an evaluation of the human-interpretability of explanation. *arXiv preprint arXiv:1802.00682* (2018).

[50] Google PAIR. 2018. What-If Tool. (2018). https://pair-code.github.io/what-if-tool/

[51] Parliament and Council of the European Union. 2016. General Data Protection Regulation. (2016).

[52] Forough Poursabzi-Sangdeh, Daniel G Goldstein, Jake M Hofman, Jennifer Wortman Vaughan, and Hanna Wallach. 2017. Manipulating and measuring model interpretability. *NIPS Women in Machine Learning Workshop* (2017).

[53] 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.

[54] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. Why should i trust you?: explaining the predictions of any classifier. In *ACM International Conference on Knowledge Discovery and Data Mining*. ACM, 1135–1144.

[55] Dominik Sacha, Michael Sedlmair, Leishi Zhang, John A Lee, Jaakko Peltonen, Daniel Weiskopf, Stephen C North, and Daniel A Keim. 2017. What you see is what you can change: human-centered machine learning by interactive visualization. *Neurocomputing* 268 (2017), 164–175.

[56] Matthias Schmid and Torsten Hothorn. 2008. Boosting additive models using component-wise P-splines. *Computational Statistics & Data Analysis* 53, 2 (2008), 298–311.

[57] Daniel Servén and Charlie Brummitt. 2018. pyGAM: generalized additive models in python. https://doi.org/10.5281/zenodo.1208723

[58] David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, Adrian Bolton, et al. 2017. Mastering the game of Go without human knowledge. *Nature* 550, 7676 (2017), 354.

[59] Bhavkaran Singh Walia, Qianyi Hu, Jeffrey Chen, Fangyan Chen, Jessica Lee, Nathan Kuo, Palak Narang, Jason Batts, Geoffrey Arnold, and Michael Madaio. 2018. A dynamic pipeline for spatio-temporal fire risk prediction. In *ACM International Conference on Knowledge Discovery & Data Mining*. ACM, 764–773.

[60] 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.

[61] Sarah Tan, Rich Caruana, Giles Hooker, and Yin Lou. 2018. Distill-and-compare: auditing black-box models using transparent model distillation. *AAAI/ACM Conference on Artificial Intelligence, Ethics, and Society* (2018).

[62] Sandra Wachter, Brent Mittelstadt, and Chris Russell. 2017. Counterfactual explanations without ppening the black box: automated decisions and the GDPR. *arXiv preprint arXiv:1711.00399* (2017).

[63] Daniel S. Weld and Gagan Bansal. 2018. Intelligible artificial intelligence. *arXiv preprint arXiv:1803.04263* (2018).

[64] Simon N Wood. 2006. *Generalized additive models: an introduction with R*. Chapman and Hall/CRC.

[65] Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. 2016. Google's neural machine translation system: bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144* (2016).

[66] Qian Yang, Jina Suh, Nan-Chen Chen, and Gonzalo Ramos. 2018. Grounding interactive machine learning tool design in how non-experts actually build models. In *Designing Interactive Systems Conference*. ACM, 573–584.

[67] Jiawei Zhang, Yang Wang, Piero Molino, Lezhi Li, and David S Ebert. 2019. Manifold: a model-agnostic framework for interpretation and diagnosis of machine learning Models. *IEEE Transactions on Visualization and Computer Graphics* 25, 1 (2019), 364–373.