To Trust or Not to Trust a Regressor: Estimating and Explaining Trustworthiness of Regression Predictions

Kim de Bie, [1] **Ana Lucic,** [1] **Hinda Haned** [1]
[1]University of Amsterdam
kimdebie@outlook.com, a.lucic@uva.nl, h.haned@uva.nl

Abstract

In hybrid human-AI systems, users need to decide whether or not to trust an algorithmic prediction while the true error in the prediction is unknown. To accommodate such settings, we introduce RETRO-VIZ, a method for (i) estimating and (ii) explaining trustworthiness of regression predictions. It consists of RETRO, a quantitative estimate of the trustworthiness of a prediction, and VIZ, a visual explanation that helps users identify the reasons for the (lack of) trustworthiness of a prediction. We find that RETRO-scores negatively correlate with prediction error across 117 experimental settings, indicating that RETRO provides a useful measure to distinguish trustworthy predictions from untrustworthy ones. In a user study with 41 participants, we find that VIZ-explanations help users identify *whether* a prediction is trustworthy or not: on average, 95.1% of participants correctly select the more trustworthy prediction, given a pair of predictions. In addition, an average of 75.6% of participants can accurately describe *why* a prediction seems to be (not) trustworthy. Finally, we find that the vast majority of users subjectively experience RETRO-VIZ as a useful tool to assess the trustworthiness of algorithmic predictions.

1 Introduction

Machine learning algorithms are increasingly used in high-stakes domains. For example, algorithms are used by judges to predict recidivism (Tan et al. 2018), by doctors to aid cancer screening (McKinney et al. 2020) and by banks to predict credit card fraud (Awoyemi, Adetunmbi, and Oluwadare 2017). This is not without risk: a trained model can produce erroneous predictions, especially when it is asked to generalize beyond situations it is familiar with (Amodei et al. 2016). Often, algorithmic failure does not affect everyone equally: research has shown that in many AI systems, algorithmic failure disproportionately affects marginalized groups (Buolamwini and Gebru 2018; Larson et al. 2016).

To prevent failures that would result from an AI system working autonomously, algorithms are often used in hybrid systems where humans aid in the decision-making process (Kamar 2016). In such systems, humans can decide to disregard a particular prediction if they believe it is erroneous or untrustworthy, thereby acting as safety mechanisms to prevent large errors from being made (Elish 2019).

It remains unclear *how* humans are expected to recognize erroneous algorithmic predictions in production settings, where the true error in individual predictions is usually unknown. Beyond global performance metrics, such as the error on a test set, many machine learning systems do not provide estimates of the reliability of individual predictions (Nushi, Kamar, and Horvitz 2018). Additionally, the complex nature of many machine learning algorithms means that the user has little insight into how an algorithm has reached a prediction, which makes it even more difficult to assess whether or not the prediction is trustworthy (Bansal et al. 2019).

Explainable AI (XAI) methods have been proposed as a tool for assessing the reliability of algorithmic predictions (Doshi-Velez and Kim 2017). However, a limiting factor of existing XAI methods is that many are not specifically equipped to explain errors: predictions are explained without awareness of their correctness. In practice, researchers have struggled to demonstrate that XAI methods allow users to recognize (un)trustworthy predictions (Lai, Carton, and Tan 2020). Overall, current methods are insufficient to help users understand when, and for what reason, algorithmic predictions are (not) trustworthy in production settings where the true error is unknown. On the one hand, confidence scores for individual predictions – when available – do not provide reasons for a measured low confidence. On the other hand, most XAI methods do not differentiate between correct and incorrect predictions.

To address this gap, we propose **RE**gression **TR**ust sc**O**res with **VI**suali**Z**ations (RETRO-VIZ). It consists of two parts: (i) **RETRO**, a method for quantitative estimation of trustworthiness in regression predictions when the true error is unknown, and (ii) **VIZ**, a visualization that helps users identify the reasons for the estimated trustworthiness. The goal of RETRO-VIZ is to provide insight into algorithmic trustworthiness in a way that aids human-algorithm cooperation. This goal is based on the needs of OurCompany, a large international retailer headquartered in OurCountry, where understanding the trustworthiness of individual algorithmic predictions is important for a wide range of tasks, such as forecasting sales or the effect of an upcoming promotion. Currently, OurCompany often uses relatively simple, transparent regression models for its prediction tasks. Although stakeholders have expressed interest in adopting more complex methods because of potential increases in overall performance, they have also expressed hesitation – they are concerned that the lack of transparency in such methods makes it more difficult to assess how and when the model is making mistakes. By proposing improved tools for assessing and understanding algorithmic errors, we aim to make the introduction of more complex methods more acceptable.

We aim to answer the following research questions:

RQ1 Do the estimates of trustworthiness that **RETRO** produces correlate with the errors in algorithmic predictions, and if so, how?

RQ2 Under which conditions does **RETRO** perform best, given different (a) model architectures, (b) data dimensionalities and (c) causes of error?

RQ3 Do **VIZ**-explanations *objectively* help users recognize whether and why algorithmic predictions are (un)trustworthy?

RQ4 Do users *subjectively* evaluate **VIZ**-explanations as being valuable in practice?

Following Rajendran and LeVine (2019), we evaluate the performance of RETRO-VIZ under specific causes of error. Generally, two main categories of uncertainty in algorithmic predictions are distinguished (Kendall and Gal 2017). Firstly, *aleatoric uncertainty* is uncertainty due to inherent noise or randomness in the data, and cannot be avoided by creating better models: for example, it is impossible to predict the outcome of rolling a dice without uncertainty. Secondly, *epistemic uncertainty* reflects the lack of knowledge in a model. Theoretically, it can be fully solved by creating better models, for instance by acquiring more training data or improving the model architecture. In this research, we assess the performance of RETRO-VIZ when predictions suffer from epistemic uncertainty: because epistemic uncertainty is avoidable, we are especially interested in addressing it. Algorithmic errors due to epistemic uncertainty can arise from several causes, and we address those which are most common during model development and when the model is in production. The causes of error due to epistemic uncertainty which we study are the following:

- **Distributional shift**. The model provided a good fit for the data available during training, but during inference the model is applied to data from a different distribution, resulting in erroneous predictions.
- **Model overfit**. The model has picked up on randomness in the training data that is not part of the underlying distribution and extrapolates this noise onto new predictions, resulting in errors on inputs not part of the train set.
- **Model underfit**. The model is not complex enough to account for the variation in the data, and predictions during both train and inference time are of a poor quality.

To gain an understanding of the settings in which RETRO-VIZ performs well and in which it does not, we design our experiments in such a way that we expect errors to arise from one of these three causes. Details of our experimental setup can be found in Section 4.

2 Related Work

RETRO-VIZ bridges the gap between uncertainty estimation and explainability in machine learning: we provide an estimate of the trustworthiness of individual regression predictions, as well as an explanation for why a prediction is (not) trustworthy. Here, we provide an overview of the existing work in the fields of uncertainty estimation and explainability, and explain why existing methods are insufficient to systematically understand when and why algorithms fail in production settings. In addition, we examine how trustworthiness is defined in different fields and relate this to the definition we use in this work.

2.1 Uncertainty Estimation in Machine Learning

Some algorithms, such as neural classifiers or Bayesian architectures, provide a measure of confidence in their predictions by default (Kendall and Gal 2017). When available, these uncertainty estimates can suffer from various limitations, such as poor calibration (Guo et al. 2017) or unreliability (Goodfellow, Shlens, and Szegedy 2014). To address this, some approaches attempt to model uncertainty directly through changing the architecture of the model (Gal and Ghahramani 2016; Lakshminarayanan, Pritzel, and Blundell 2017; Papernot and McDaniel 2018). Still, in practice, it may not always be feasible to alter the model architecture, for example when a model is already in production. Therefore, we propose a method that estimates trustworthiness without depending on particular characteristics of the model architecture, as many common architectures do not inherently provide a measure of confidence in individual predictions.

Several model-agnostic methods exist for estimating predictive uncertainty. Jiang et al. (2018) proposed a 'Trust Score', which is a simple method to estimate the trustworthiness of individual predictions made by a classifier. The method estimates the trustworthiness of a prediction by measuring the agreement between a classifier and a modified nearest-neighbor classifier on a particular test instance. If an instance is classified into a very different class than similar instances, there is reason to believe that the prediction might be faulty. The intuition behind this is similar to the main idea behind case-based reasoning (CBR) methods, which assume that similar problems should have similar solutions (Aamodt and Plaza 1994; Kenny and Keane 2019; Li et al. 2018).

As Rajendran and LeVine (2019) showed, the Trust Score correlates with errors that are caused by distributional shift: instances that are far removed from the training data will receive a low score. However, the Trust Score correlates less strongly with classification error arising from other causes. Although Rajendran and LeVine (2019) proposed an alternative to the Trust Score that captures a wider range of uncertainty types, their method suffers from the same limitations as Jiang et al. (2018): (i) both methods are only applicable to classification problems, and (ii) both methods do not provide any explanation as to *why* a prediction is (not) trustworthy. In contrast, RETRO-VIZ estimates the trustworthiness in regression predictions, since regression models are applicable across a wide range of real-world problems, including those at OurCompany. Moreover, to provide context for the estimated trustworthiness, RETRO-VIZ also provides a visual explanation.

2.2 Explainable Machine Learning

Increasing the interpretability of complex models can help users make better use of algorithmic predictions (Doshi-Velez and Kim 2017). Specifically, model explanations may help with debugging and detecting errors: if an algorithm is making decisions on the basis of irrelevant factors, this could be an indication that there is something wrong (Ribeiro, Singh, and Guestrin 2016). In a study of how organizations use explanation methods in practice, Bhatt et al. (2020) found that detecting errors is the most common use case for explanations. However, researchers have struggled to demonstrate that explanations, which commonly focus on explaining model internals, lead to an improvement in decision quality (Lai, Carton, and Tan 2020).

Lucic, Haned, and de Rijke (2020) developed a method for explaining errors produced by regression models, which is similar to what we propose. However, this method can only be used to explain *known* errors, i.e. errors for which the ground-truth target value is available, which is usually not the case when the model is in production. In contrast, RETRO-VIZ provides estimates as well as explanations of the trustworthiness of predictions when the true value is not (yet) known and can therefore be employed in production environments.

Antorán et al. (2020) propose CLUE, a method for creating counterfactual explanations of uncertainty estimates from Bayesian Neural Networks. The counterfactuals provided indicate how an input would have to be changed such that the model becomes more certain about its prediction. While similar to what we propose, this method differs from ours in several ways. Firstly, RETRO-VIZ does not rely on existing uncertainty estimates, but creates its own. Therefore, RETRO-VIZ can provide explanations for any regressor, unlike CLUE which is only applicable to Bayesian models. Secondly, the explanations we provide are more similar to feature-based explanations such as LIME (Ribeiro, Singh, and Guestrin 2016) or SHAP (Lundberg and Lee 2017) than counterfactual examples, since VIZ-explanations are framed around showing deviations in feature values as opposed to suggesting changes to feature values that result in alternative predictions.

2.3 Trustworthiness

This work proposes a method that measures the *trustworthiness* of regression predictions. The concepts of trust and trustworthiness are used differently in various branches of machine learning scholarship, and we provide a brief overview in this section. We relate this to the definition of trustworthiness that we use our own work and how this is distinct from the correctness of a prediction.

Operational Definition Jiang et al. (2018) propose a 'Trust Score', which estimates the trustworthiness of a prediction by measuring the agreement between a classifier and a nearest-neighbor classifier on a test instance. In this case, a prediction is *trustworthy* if it is reasonable in light of the training data, in the sense that the train data behaves in a similar way to the new instance. As the method we propose for the numeric estimation of trustworthiness is heavily inspired by Jiang et al. (2018), we adopt this definition of trustworthiness: a prediction is trustworthy if it is aligned with the data the model was trained on. We note that this intuition is also very similar to the idea behind case-based reasoning methods (Aamodt and Plaza 1994; Kenny and Keane 2019; Li et al. 2018).

The notion of trustworthiness as 'grounded-ness' in train data relates to epistemic uncertainty, which reflects a lack of knowledge in the learned model (Kendall and Gal 2017). Namely, for a model that is a poor fit to the true data distribution that a new instance is part of, predictions for new instances will not be consistent with the train data distribution. However, for aleatoric uncertainty, the uncertainty is not necessarily related to trustworthiness in the sense of alignment with the train data. Specifically, a model may produce a highly erroneous prediction for an instance if it suffers from large aleatoric uncertainty or uncertainty due to inherent randomness, even if the instance and the prediction align closely with the train data.

Trustworthiness vs. Correctness Based on the above, we explicitly distinguish between the *trustworthiness* and the *correctness* of a prediction: trustworthiness expresses whether a prediction is aligned with the train data without having access to the ground truth, while correctness expresses whether a prediction is erroneous or not, which requires access to the ground truth. Although they can be correlated (as shown by Jiang et al. (2018)), trustworthiness and correctness are therefore not synonymous: trustworthiness does not capture the inherent or aleatoric uncertainty present in algorithmic predictions. However, we expect that trustworthy predictions tend to be more accurate because they are grounded in the training data, and we assess this in **RQ1** and **RQ2**.

Trust in XAI As Lipton (2018) points out, 'improving trust' is an important motivation for the development of XAI methods and is featured prominently in existing work (Lundberg and Lee 2017; Ribeiro, Singh, and Guestrin 2016), but a clear definition of trust or trustworthiness is lacking in the XAI literature. The implicit understanding is that an algorithm is trustworthy if its predictions are based on factors that are reasonable from the perspective of a domain expert instead of on 'spurious correlations' (Ribeiro, Singh, and Guestrin 2016). Similarly, Doshi-Velez, Budish, and Kortz (2017) explicitly relate explanations and trustworthiness, as explanations can "help validate whether a process was performed appropriately or erroneously", which can help increase trust in algorithmic predictions.

Broader Scope Other authors use a much broader definition of trust in the context of machine learning systems. For example, Lee and See (2004) define trust as "the attitude that an agent will help achieve an individual's goals in a situation characterized by uncertainty and vulnerability". Similarly, Toreini et al. (2020) relate trust in AI to the belief that an AI system will behave in a way that is generally beneficial to the trustor, which they relate to larger issues of justice and fairness (e.g. does this system discriminate against me?). They point out that the 'ability' of a system, which relates to an objective quality of predictions, is only a part of trustworthiness in this sense, but also that trustworthiness cannot exist without it. In our research, we take a much narrower perspective on trustworthiness and focus on 'ability' (i.e., the objective quality of predictions alone). Moreover, we focus on trustworthiness of individual predictions rather than of the AI system as a whole.

3 Method

In this section, we introduce RETRO-VIZ, a method for identifying and explaining the trustworthiness of regression predictions. It consists of two components: (i) RETRO, which provides a numerical estimation of trustworthiness for individual predictions, and (ii) VIZ, which provides a visual explanation of the RETRO-score.

We build upon the method proposed by Jiang et al. (2018), which estimates trustworthiness in predictions by a classifier. The central notion behind this method is that similar instances should receive a similar prediction, which is also the core idea underlying case-based reasoning methods (Aamodt and Plaza 1994; Kenny and Keane 2019; Li et al. 2018).

First, we calculate the RETRO score. RETRO requires a (i) trained regression model, and (ii) the data that the model was trained on. All input and output variables must be numeric. In order to accommodate all possible regression models, we treat the model as a black box and therefore do not require access to the internals or parameters. Given a new instance x_p, its predicted target value $\hat{y}_p$ and training data (X_{train}, Y_{train}), the method consists of three phases, which are discussed below.

3.1 RETRO Phase 1: Preparing the Reference Set

RETRO relies on a reference set which is based on the training data. As will be outlined in Phase 2, RETRO leverages the distance between the new instance and similar instances from this set to estimate the trustworthiness of a prediction.

Step 1A. Filter errors from the train data RETRO leverages a reference set for which the regression model can produce high-quality predictions. If the regression model has been trained on data that is similar to the new instance, we assume that the model will be able to make a high-quality prediction for this instance. However, this does not hold when the model has not sufficiently captured a region of the input space in order to produce high-quality predictions, despite it being in the training data distribution. Therefore, we remove those instances from the train set for which the model produces an (absolute) error in the top-α fraction of all training data, where $\alpha = 0.1$ in our experiments.

Step 1B. Reduce dimensionality of the data (optional) Since RETRO involves a k-Nearest Neighbors (kNN) procedure, which is known to perform poorly in high-dimensional spaces (Beyer et al. 1999), we reduce the dimensionality of the data when the number of features is > 15. We do this by training a Multi-Layer Perceptron (MLP) whose penultimate layer represents the desired dimensionality (10 in our experiments). The MLP is trained to predict Y_{train} given X_{train}. In our experiments, we found that this method performs better than classic dimensionality-reduction methods such as PCAs or autoencoders. We postulate that this is because the MLP maintains those aspects of the original input most relevant to making the prediction.

3.2 RETRO Phase 2: Determining Relationship to Neighbors

In Phase 2, we assess whether the predicted target value for the new instance seems trustworthy in light of the reference set created in Phase 1.

Step 2A. Find the closest neighbors of the new instance in the train data Using the kNN method, we find the K instances in the reference set that are most similar to our new instance x_p as measured through Euclidean distance.

Step 2B. Find the average distance to the closest neighbors (d_1) For a trustworthy prediction, we expect that the new instance has neighboring instances at a relatively close distance in the reference set. If this is not the case, the new instance is likely to be an outlier. Given the Euclidean distance δ_k of the new instance x_p to each of its K neighbors, we take the average of all δ_k to find the mean distance to the neighbors, denoted as d_1. The larger d_1, the less the new instance resembles the reference data, and therefore the lower the expected quality of the prediction.

Step 2C. Find the distance between the predicted and ground-truth targets (d_2) For a trustworthy prediction, we also expect that the target prediction for the new instance is close to the target value of the neighbors in the reference set. Then, given the ground-truth target variable y for each of the K neighbors, which we denote as y_k, we take the average across all y_k to find the mean ground-truth target value of the neighbors from the train set. Next, we determine the absolute distance to the predicted value for the new instance $\hat{y}_p$, which we denote as d_2.

$$d_2 = \left| \left(\frac{1}{K} \sum_{k=1}^{K} y_k \right) - \hat{y}_p \right| \tag{1}$$

As we expect similar instances to receive similar predictions, d_2 should be small for high-quality predictions.

3.3 RETRO Phase 3: Scoring and Normalization

In Phase 3, we determine the RETRO-score for a prediction and optionally normalize it.

Step 3A. Calculate the RETRO-score R from d_1 and d_2 The RETRO-score R is defined as follows:

$$R = -\left(\beta \cdot d_1 + [1 - \beta] \cdot d_2\right) \tag{2}$$

Here, β may be used to weight the contribution from d_1 relative to d_2. Based on our experiments, we found $\beta = 0.5$ to be a good default value.

Step 3B. Normalize the RETRO-score (optional) R as it is calculated in Step 3A has a potential range of $[-\infty, 0]$, given that both d_1 and d_2 are unbounded. This makes R difficult to interpret in isolation since its value is largely dependent on the particular dataset and model. To normalize R to a range of $[0, 1]$, we calculate the RETRO-score $\tilde{R}$ for each of the instances in the training set and obtain the minimum and the maximum score, $\tilde{R}_{min}$ and $\tilde{R}_{max}$. We normalize the RETRO-scores for new instances to this range, where values that fall outside the range are clipped to 0 or 1. Note that this normalization is optional, and the need to do so is dependent on the domain.

3.4 VIZ: Visually Explaining Trustworthiness

Besides identifying predictions that are potentially erroneous, we want to provide users with an actionable tool that helps them understand *why* a particular prediction is (un)trustworthy. We use Parallel Coordinate Plots, as proposed by Inselberg (1985) for displaying multi-dimensional data, to visualize the new instance and its K neighbors as retrieved through the RETRO-method. Example VIZ-plots are provided in Figures 1, 2 and 3. Note that in practice, these plots are interactive: users can observe the exact feature values of each instance (represented by a line) by hovering the cursor over the line. While dimensionalities of up to 10 can be reasonably displayed on a single VIZ-plot, features have to be selected for higher-dimensional feature spaces. In such cases, the number of features displayed at one time may be reduced in several ways, such as selecting features by their importance, where importance could be determined by a method such as LIME (Ribeiro, Singh, and Guestrin 2016) or SHAP (Lundberg and Lee 2017), or selection of features could be based on user expertise.

VIZ-plots make it straightforward to identify why a prediction has received a high or low RETRO-score. A low RETRO-score can be obtained for a new instance for two main reasons. Firstly, it could be that the new instance has no instances in the reference set that lie relatively close to it, because the new instance deviates in one or multiple features. An example of this is provided in Figure 1. Secondly, it could be that the predicted target variable lies far away from the ground-truth target variables of the K neighbors (see Figure 2 for an example). In contrast, when a new instance and its prediction are aligned with the reference data, they will receive a high RETRO-score, and the lines on the VIZ-plot will lie close to each other, like in Figure 3.

VIZ-plots allow users to leverage their domain expertise: based on the way in which the new instance deviates from its neighbors, the domain expert can design an appropriate response. For example, in Figure 1, we observe that the new instance is a much larger store (measured by floor_surface), and spends more on advertising than the most similar stores in the reference set, so it is likely that true sales are higher than predicted by the model.

4 Evaluating the RETRO-score

4.1 Experimental Setup

To answer **RQ1**, we assess the Pearson correlation coefficient ρ between the RETRO-score and the absolute error in predictions (which is unknown in production settings). We expect a negative correlation: a lower RETRO-score, which implies a lower trustworthiness of predictions, should correspond to larger errors. To answer **RQ2**, we assess the correlation across a wide range of models (**RQ2a**), data dimensionalities (**RQ2b**), and error causes (**RQ2c**). Following Rajendran and LeVine (2019), we select models and datasets to evaluate specific causes of error: (i) *Distributional Shift*, (ii) *Model Overfit* and (iii) *Model Underfit*, resulting in 117 experimental settings. We use these error causes because they represent common sources of algorithmic failure, both during model development (overfit/underfit) and when the model is in production (distributional shift). We use 13 datasets in our experiments, which originate from *scikit-learn*[1], the UCI ML repository (Dua and Graff 2020) and OurCompany. We use 80% of the data to train the models, which leaves 20% as test data on which we evaluate the RETRO-score.

Error Cause (i): Distributional Shift Distributional shift occurs when a model produces erroneous predictions on a test set because it was trained on data from a different distribution. Three of our datasets, *Store sales*, *Wine quality* and *Autompg*, can be split in such a way that distributional shift naturally occurs. For example, the *Wine quality* dataset consists of white and red wines, so we train on white and test on red wines.

For the remaining datasets, we induce distributional shift manually. To do this, we select the 30% of most important features in the test set according to their SHAP value (Lundberg and Lee 2017). Then, for each feature k, we find its maximum value, x_k^{max}, and generate noise for each instance n_k^i from a Gaussian distribution, where $n_k^i \sim \mathcal{N}(x_k^{max},\, 0.1 \cdot x_k^{max})$. We then add this noise to the original feature, so that $\tilde{x}_k^i = x_k^i + n_k^i$. In this way, we ensure that the values of the perturbed features lie outside the original distribution.

We train five types of models: a Linear Regression (LR), a Support-Vector Regressor (SVR), a Multi-Layer Perceptron (MLP) with layer sizes $[50, 20, 10]$, a Decision Tree (DT), and a Random Forest (RF) with 100 trees. Both the DT and RF have maximum depth 15.

Error Cause (ii): Model Overfit Model overfit occurs when a model fits the train data 'too well' and picks up on randomness that is not part of the underlying distribution. To evaluate the RETRO-score for predictions suffering from this type of error, we train a very deep DT (depth 10,000; DT-10k) and a Gaussian Process (GP) Regressor with an RBF kernel. We train these models such that they produce a near-zero error on the train data, but a much larger error on test data.

[1] https://scikit-learn.org/stable/

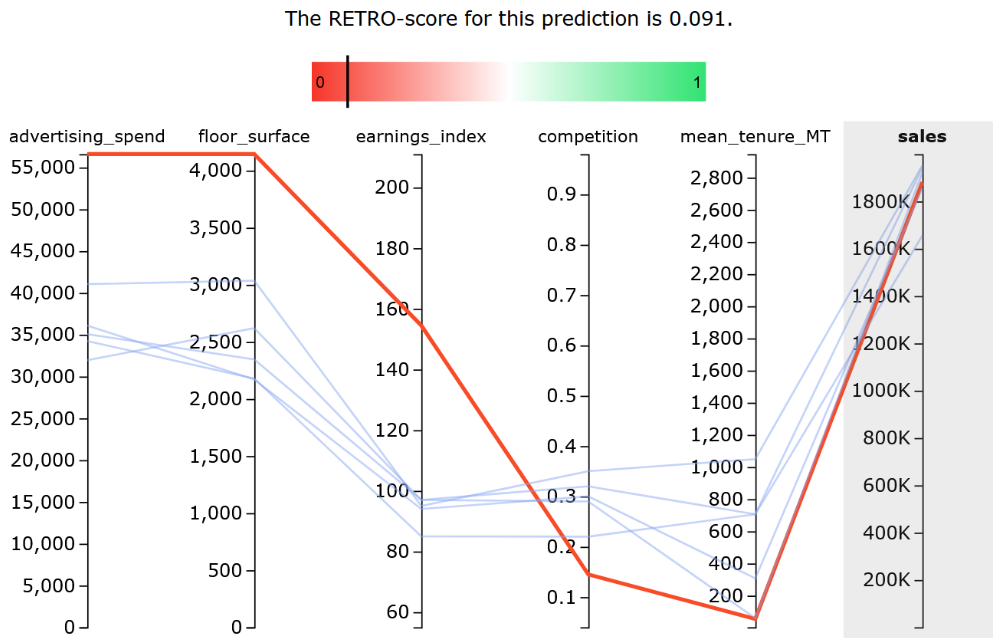

Figure 1: Example RETRO-VIZ output for an untrustworthy prediction. This model predicts sales based on five features. The RETRO-score is 0.091 for the test instance (red). The VIZ-explanation shows that its nearest neighbors from the reference set (blue) have feature values that are quite different from its own, even though the predicted sales value for the new store is similar to the ground-truth sales values of the neighboring instances. This indicates that the new instance is not well-represented in the training set and the prediction may be unreliable.

	(i) Distributional shift					*(ii) Model Overfit*		*(iii) Model Underfit*		
Dataset (features)	**LR**	**SVR**	**MLP**	**DT-15**	**RF**	**GP**	**DT-10k**	**MLP-S**	**DT-1**	**Mean (SD)**
Cyclepower (4)	-0.947	-0.370	-0.690	-0.327	-0.372	$-0.282^{\dagger}$	-0.389	-0.906	-0.741	**-0.593 (0.258)**
Airfoil (5)	-0.880	-0.739	-0.697	-0.495	-0.567	-0.999	-0.199	-0.909	-0.393	**-0.653 (0.263)**
*Store sales*** (5)	-0.062	-0.782	-0.663	-0.614	-0.650	-1.000	-0.534	-0.780	-0.834	**-0.658 (0.262)**
Fish toxicity (6)	-0.695	-0.589	-0.563	-0.460	-0.419	-1.000	-0.414	-0.791	-0.456	**-0.599 (0.198)**
Abalone (7)	-0.634	-0.427	-0.399	-0.063	-0.109	-0.999	-0.609	-0.552	-0.212	**-0.445 (0.295)**
Autompg (7)	-0.566	-0.892	-0.954	-0.017	-0.210	-0.992	-0.542	-0.914	-0.586	**-0.630 (0.345)**
*Cal. housing** (8)	-0.995	-0.607	-0.964	-0.263	-0.261	-1.000	-0.475	-0.271	-0.433	**-0.585 (0.322)**
Energy eff. (8)	-0.980	-0.415	-0.948	-0.472	-0.409	-0.734	-0.225	-0.940	-0.458	**-0.620 (0.284)**
*Diabetes** (10)	-0.056	-0.237	-0.246	-0.341	-0.305	-0.751	-0.255	-0.460	-0.147	**-0.311 (0.200)**
Wine quality (11)	-0.163	-0.289	-0.512	-0.483	-0.234	-0.999	-0.448	-0.311	-0.212	**-0.406 (0.255)**
*Boston** (13)	-0.348	-0.863	-0.793	-0.864	-0.657	-0.699	-0.433	-0.687	-0.412	**-0.640 (0.197)**
Supercond. (81)	-0.946	-0.275	-0.361	-0.546	-0.511	-1.000	-0.463	-0.719	-0.745	**-0.618 (0.251)**
Communities (100)	-0.880	-0.323	-0.186	-0.718	-0.408	-0.617	-0.604	-0.620	-0.535	**-0.543 (0.210)**
Mean (SD)	**-0.627 (0.358)**	**-0.524 (0.234)**	**-0.614 (0.264)**	**-0.436 (0.238)**	**-0.393 (0.170)**	**-0.899 (0.150)**	**-0.430 (0.134)**	**-0.682 (0.228)**	**-0.474 (0.214)**	***-0.561 (0.268)***

Table 1: Pearson correlation coefficient ρ between the absolute error and the RETRO-score for all experimental settings. Correlations closer to -1 are more desirable. The number of features is provided in the first column. See Section 4 for the model acronyms. *: Datasets from *scikit-learn*. **: Internal OurCompany dataset. Other datasets from UCI-ML (Dua and Graff 2020). † GP did not lead to overfitting, so the value was not included in the mean and standard deviation.

Error Cause (iii): Model Underfit When a model is not complex enough to capture the patterns present in the data, it *underfits* the data and produces poor-quality predictions on both the train and test set. We induce this error by fitting a very shallow MLP with one hidden layer of width 10 trained for 3 iterations (MLP-S) and a DT of depth 1 (DT-1) to the train data.

4.2 Results

An overview of all 117 experimental settings is given in Table 1, which lists the Pearson correlation coefficient ρ for each dataset and model. In all cases, ρ is negative, which confirms our expectations: RETRO-scores are lower when the prediction error is larger. Across all settings, the average value of ρ is $-\mathbf{0.561}$. Thus, similar to Jiang et al. (2018), we confirm that trustworthiness as measured by RETRO correlates with correctness as measured by the absolute error in predictions. While this correlation is not perfect, this is not unexpected: as outlined in Section 2.3, trustworthiness as measured by RETRO does not capture aleatoric or inherent randomness in predictions. This answers **RQ1** and shows that RETRO is useful to help users distinguish erroneous predictions.

To answer **RQ2**, we study the conditions under which RETRO performs best by examining the magnitude of ρ across settings.

RQ2a: Performance across model architectures We first examine for which model architectures RETRO performs best. We obtain the best performance for predictions generated by an overfitting GP (mean of $-\mathbf{0.899}$), the second-best performance for the underfit MLP-S ($-\mathbf{0.682}$), followed by the non-tree distributional shift models (mean of $-\mathbf{0.627}$ for LR, $-\mathbf{0.524}$ for SVR and $-\mathbf{0.614}$ for MLP).

We find that RETRO performs worse for tree-based models than for other model architectures. This can likely be explained by the properties of tree-based models: they exclusively predict from a set of values that is based on the training data. As a result, component d_2 of the RETRO-score, which measures the distance between the predicted target for the new instance and the ground-truth target values of the neighboring train instances, is always reasonably small. Therefore, for tree-based models, the RETRO-score leans heavily on component d_1, which measures the distance between the nearest neighbors and the new instance. We contrast this to other types of model architectures, whose predictions are not constrained in this way and can take more extreme values, leading to potentially larger values of d_2. Then, for other model architectures, the RETRO-score can optimally leverage both of its components d_1 and d_2, resulting in the observed better performance.

RQ2b: Performance across data dimensionalities Next, we examine the effect of dimensionality on the RETRO-score. In our experiments, the number of independent features of the datasets used ranges from 4 to 100. Recall from Section 3 that the two largest datasets, with 81 and 100 features, are reduced in dimensionality. In our settings, we find that the number of features does not have a clear impact on the performance of the method. This can be observed in the right-most column of Table 1: we do not see a clear trend across the feature sizes.

RQ2c: Performance across causes of error Finally, we examine how well RETRO performs for different causes of predictive error. As noted earlier in this section, the three models for which RETRO performs best (GP, MLP-S and LR) span the three different error causes used in our experiments. Likewise, for the tree-based models, we find an approximately equal performance across all error causes. Therefore, we conclude that there does not seem to be a clear relationship between the cause of error and the performance of RETRO.

5 Evaluating VIZ-explanations

In practice, it is users who decide (not) to use an algorithm or an explanation method. Indeed, "if the users do not trust a model [...], they will not use it" (Ribeiro, Singh, and Guestrin 2016). Therefore, potential improvement in objective performance can only be realized when humans accept the method as valuable. For this reason, we are interested in objectively as well as subjectively evaluating RETRO-VIZ with users. Specifically, we want to understand (i) whether VIZ helps users to recognize the trustworthiness of predictions (**RQ3**), and (ii) whether VIZ is subjectively perceived as satisfying by potential users (**RQ4**).

5.1 User Study Setup

We evaluate VIZ-plots in a user study to answer **RQ3** and **RQ4** with 41 participants from OurCompany, all of whom are data scientists or analysts. To emulate a realistic setting that our users are familiar with, we generate VIZ explanations for a model trained on a store sales dataset from OurCompany. We use a *human-grounded* evaluation (Doshi-Velez and Kim 2017): a simplified task to assess whether users can make sense of the VIZ-explanations, and whether they allow users to recognize untrustworthy predictions. Here, the task is to predict store sales based on five input variables (see Figure 1). The user study consists of two parts, namely an objective and subjective evaluation of RETRO-VIZ.

Objective evaluation In the first part, we introduce the method and evaluate whether VIZ-plots are *objectively* understood by participants. We do this by showing users a pair of plots with differing RETRO-scores and asking them to indicate which plot corresponds to a more trustworthy prediction (*Select Best*). The correct answer is the plot with the higher RETRO-score. However, it should be noted that we *do not* show the RETRO-score to the users: we only use the RETRO-score to determine which prediction is more trustworthy. We repeat this question three times, with three different pairs of plots. We refer to these questions as *Select Best (SB)*.

Next, we show users a single VIZ-plot and ask them to explain in words *why* they believe the prediction to be (un)trustworthy. We refer to these questions as *Identify Reason (IR)*. We do this three times, with three different plots (IR1-3), and manually inspect answers to assess whether users correctly identified the reasons for (not) trusting a prediction. In IR1, we show a sales prediction which strongly deviates from the ground-truth sales for the most similar instances in the train set (similar to Figure 2), so we expect users to refer to this deviation in the target variable. In IR2 we show a prediction similar to that in Figure 1, and we expect users to refer to the fact that the new instance does not

		Correct	χ^2
Select Best	***SB1***	95.1%	33.390**
	SB2	97.6%	37.098**
	SB3	92.7%	29.878**
Identify Reason	***IR1***	68.3%	5.488*
	IR2	82.9%	26.561**
	IR3	75.6%	10.756**

Table 2: Percentage of Correct responses to the objective questions. Significant differences are denoted using * ($p \leq 0.05$) and ** ($p \leq 0.01$).

resemble the train data. In IR3, we show a prediction similar to Figure 3, and expect users to argue the prediction is trustworthy because it is aligned with the train data.

Subjective evalution In the second part of the user study, we use an adapted version of the Explanation Satisfaction Scale as proposed by Hoffman et al. (2018) to evaluate participants' *subjective* attitudes towards the VIZ-plots. This scale attempts to standardize existing approaches to subjective evaluation of XAI methods, and measures "the degree to which users feel that they understand [...] the process being explained to them" (Hoffman et al. 2018). The questionnaire is based on an extensive literature review and has been tested for validity and internal consistency. Therefore, we slightly adapt the questions from the Explanation Satisfaction Scale to fit our context better. The adapted questions are listed in Table 3. Answers are provided on a five-point Likert scale.

5.2 Results

In this section, we discuss the results of the objective and subjective evaluation as performed in the user study.

Results of objective evaluation Table 2 summarizes participants' answers to the objective questions. We find that for all objective questions, the vast majority of users are able to answer the question correctly, for both *Select Best* and *Identify Reason*. Using a χ^2 test, we find that this majority is significant for all questions ($p \leq 0.05$). This answers **RQ3**: VIZ-plots *objectively* help users understand whether *(Select Best)* and why *(Identify Reason)* algorithmic predictions are trustworthy. Then, perhaps surprisingly, it seems that VIZ-plots alone are sufficient for users to distinguish trustworthy predictions from untrustworthy ones, even when users do not have access to the associated RETRO-score.

Results of subjective evaluation In Table 3, we show the subjective questions from the Explanation Satisfaction Scale, adapted to fit our context from Hoffman et al. (2018), and summarize the reponses. We omit neutral responses. For 5 out of 6 questions, a significantly larger number of participants give a positive answer than a negative one ($p \leq 0.05$). We find that participants in the user study experience VIZ-plots as understandable (**Q4**) tools to discover whether (**Q2**) and why (**Q3**) algorithmic predictions are trustworthy and accurate (**Q1**), and that they believe VIZ-plots could be valuable in their work (**Q6**; $p \leq 0.05$ for all questions). There is no significant difference in the proportion of participants agreeing and disagreeing with **Q5**, indicating that further tools and analysis about whether to trust a prediction may be required. This is not unexpected: to determine whether deviations in either features or predictions are problematic and to decide what actions should be taken, users must rely on their domain expertise and may leverage other sources of information within their organization. This answers **RQ4**: participants clearly *subjectively* experience VIZ-plots as valuable to deal with the uncertainty in algorithmic predictions.

6 Discussion

We proposed RETRO-VIZ, a method for assessing the trustworthiness of regression predictions. In all 117 experimental settings, we found that there was a negative correlation between the trustworthiness of a prediction as measured by RETRO and its correctness, as desired. In addition, in our user study, we find that VIZ-plots help users distinguish whether and why algorithmic predictions are trustworthy. However, similarly to other XAI methods (Lakkaraju and Bastani 2020; Slack et al. 2020), RETRO-VIZ must be used with caution to avoid misguided trust. While RETRO-scores generally correlate with error, the correlation is not perfect, and therefore trustworthiness as measured by RETRO does not equal correctness. Therefore, user awareness of this limitation is crucial.

In principle, RETRO-VIZ is fully model-agnostic and we find that RETRO-VIZ performs reasonably well for a diverse set of models. However, the correlation between RETRO-scores and errors is weaker for tree-based models. This is due to the properties of these models: they exclusively predict from a set of values that is based on the training data, such that component d_2 of the RETRO-score is always small, in contrast to other models, whose predictions are not constrained in this way and can take more extreme values (see Section 4 for more details). Future research could investigate how to make RETRO-VIZ more applicable to tree-based models.

7 Conclusion

We introduced RETRO-VIZ, a method for estimating (RETRO) and explaining (VIZ) the trustworthiness of individual predictions from regression models when the true error is unknown, as is common in production settings. RETRO-VIZ contributes to better end-to-end decision-making in hybrid human-AI systems by helping users identify untrustworthy and potentially erroneous predictions. In this way, as RETRO-VIZ offers a way to mitigate the risk of potential failure of complex models, RETRO-VIZ could help to make such methods more acceptable to organizational stakeholders.

RETRO measures trustworthiness by assessing whether the prediction for a new instance is reasonable in light of the train data. The RETRO-score is based on the similarity between the new instance and its nearest neighbors in the train data: if the new instance receives a very different prediction from its neighbors, or these neighbors lie very far away, the prediction is not considered trustworthy. In contrast, if the new instance closely resembles its neighbors from the

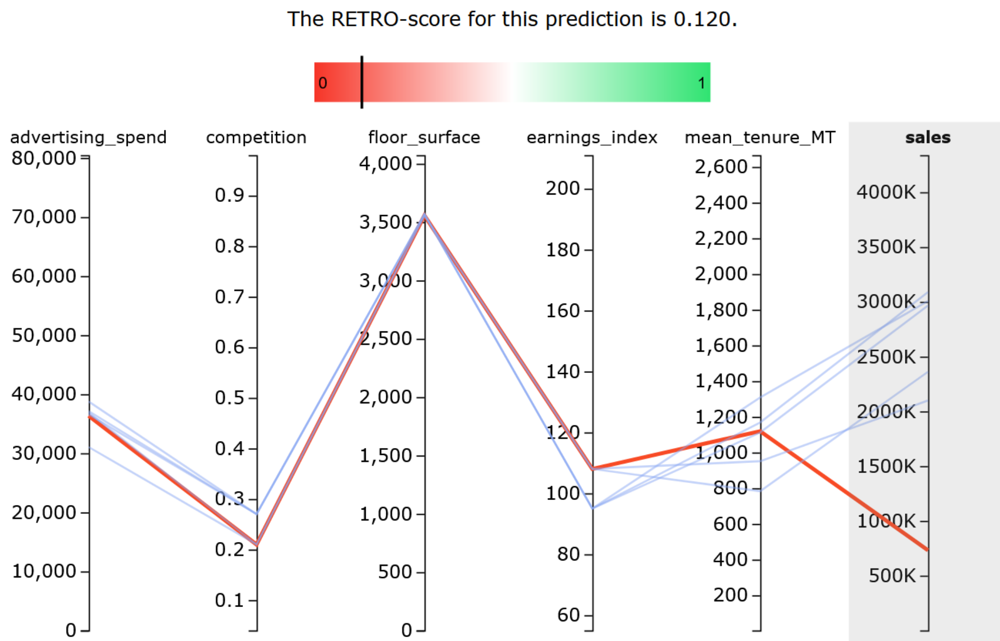

Figure 2: Example RETRO-VIZ output for an untrustworthy prediction. The model used is identical to that described in Figure 1. The RETRO-score is 0.120 for the test instance (red). The VIZ-explanation shows that its nearest neighbors from the reference set (blue) have similar feature values, but the predicted sales for the new instance are much larger. This indicates that the prediction is untrustworthy.

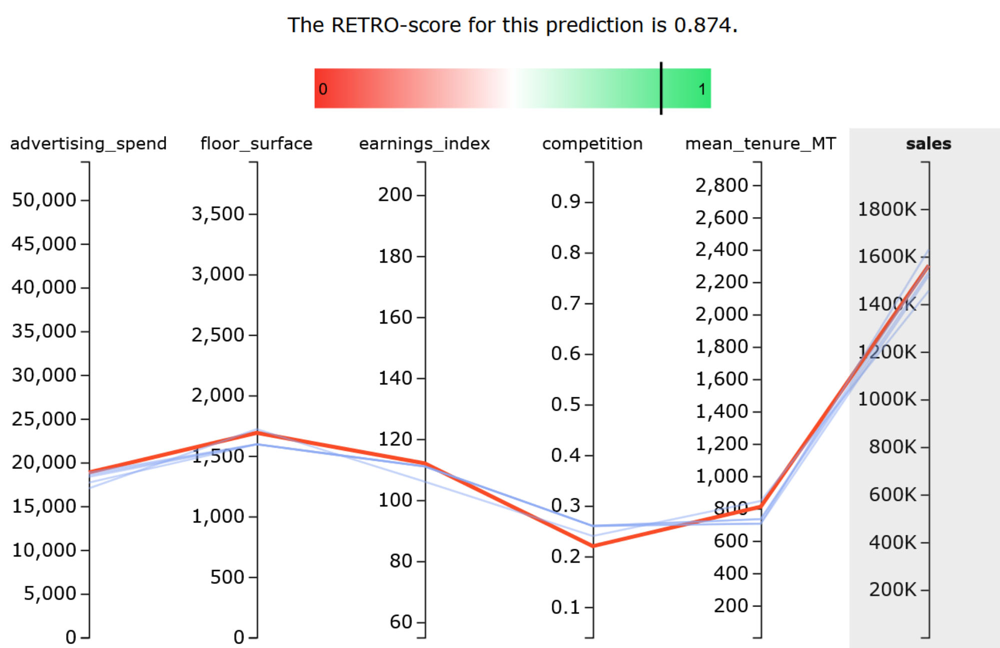

Figure 3: Example RETRO-VIZ output for a trustworthy prediction. The model used is identical to that described in Figure 1. The RETRO-score is 0.874 for the test instance (red). The new instance is well-aligned with its neighbors from the test set and therefore the prediction seems trustworthy.

	Agree	Disagree	χ^2
Q1. The visualization shows me how accurate the prediction is.	63.4%	29.3%	5.16*
Q2. The visualization lets me judge when I should trust or not trust the prediction.	87.8%	7.3%	27.92**
Q3. From the visualization, I understand why the algorithmic prediction is trustworthy.	78.0%	19.5%	14.40**
Q4. The visualization is satisfying: I understand what it is showing.	87.8%	4.9%	30.42**
Q5. The visualization gives me all information I need to assess the trustworthiness of a prediction.	34.1%	43.9%	0.50
Q6. I think the visualization of whether the prediction should be trusted is useful in operational settings.	87.8%	4.9%	30.42**

Table 3: Percentage of users who (strongly) Agree or (strongly) Disagree with each statement. As neutral answers are omitted, the total is not 100%. Significant differences are denoted using * ($p \leq 0.05$) and ** ($p \leq 0.01$).

train data and receives a similar prediction, the prediction is trustworthy. VIZ provides a visualization of the predicted instance and its nearest neighbors in the train set in a Parallel Coordinate Plot, allowing the user to explore the reasons for a low or high trustworthiness of the prediction.

We evaluate RETRO quantitatively, and VIZ through a user study. We show that RETRO negatively correlates with error across all 117 experimental settings, confirming that trustworthiness as measured by RETRO is a useful concept to help users identify erroneous predictions. In a user study with 41 participants, we find that VIZ-plots allow users to identify (i) whether and (ii) why predictions are trustworthy in a task derived from their day-to-day activities, and that users experience the plots as valuable to their daily work. Therefore, despite associated risks and limitations, we conclude that RETRO-VIZ can contribute to improved decision-making in hybrid human-AI systems.

In future work, we aim to explore ways to make RETRO-VIZ applicable to settings beyond regression with quantitative features. For example, we aim to explore including categorical features and extending RETRO-VIZ to time series regression problems. In addition, we aim to expand on our user study by investigating to what extent RETRO-VIZ leads to a better end-to-end system performance as compared to other explainability methods. Lastly, building on our finding that users seem to be able to identify untrustworthy predictions from a VIZ-plot alone, without having access to the RETRO-score (see Section 5), we aim to explore how RETRO and VIZ can be used in a complementary way in deployed hybrid human-AI systems. For example, RETRO-scores could be used as an automated monitoring tool, where only for predictions whose RETRO-score is below a threshold a VIZ-plot is shown to the user.

Open-Source Software and Data

RETRO-VIZ can be installed via pip as the Python package ‘retroviz’. Code for the experiments is also available at https://github.com/kimdebie/retroviz-tutorial.

References

Aamodt, A.; and Plaza, E. 1994. Case-based reasoning: Foundational issues, methodological variations, and system approaches. *AI Communications* 7(1): 39–59.

Amodei, D.; Olah, C.; Steinhardt, J.; Christiano, P.; Schulman, J.; and Mané, D. 2016. Concrete problems in AI safety. *arXiv preprint arXiv:1606.06565* .

Antorán, J.; Bhatt, U.; Adel, T.; Weller, A.; and Hernández-Lobato, J. M. 2020. Getting a CLUE: A Method for Explaining Uncertainty Estimates. *arXiv preprint arXiv:2006.06848* .

Awoyemi, J. O.; Adetunmbi, A. O.; and Oluwadare, S. A. 2017. Credit card fraud detection using machine learning techniques: A comparative analysis. In *2017 International Conference on Computing Networking and Informatics (ICCNI)*, 1–9. IEEE.

Bansal, G.; Nushi, B.; Kamar, E.; Weld, D. S.; Lasecki, W. S.; and Horvitz, E. 2019. Updates in human-ai teams: Understanding and addressing the performance/compatibility tradeoff. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, 2429–2437.

Beyer, K.; Goldstein, J.; Ramakrishnan, R.; and Shaft, U. 1999. When is “nearest neighbor” meaningful? In *Proceedings of the International Conference on Database Theory*, 217–235. Springer.

Bhatt, U.; Xiang, A.; Sharma, S.; Weller, A.; Taly, A.; Jia, Y.; Ghosh, J.; Puri, R.; Moura, J. M.; and Eckersley, P. 2020. Explainable machine learning in deployment. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, 648–657.

Buolamwini, J.; and Gebru, T. 2018. Gender shades: Intersectional accuracy disparities in commercial gender classification. In *Conference on fairness, accountability and transparency*, 77–91.

Doshi-Velez, F.; Budish, R.; and Kortz, M. 2017. The Role of Explanation in Algorithmic Trust. Technical report, Berkman Klein Center for Internet and Society.

Doshi-Velez, F.; and Kim, B. 2017. Towards a rigorous science of interpretable machine learning. *arXiv preprint arXiv:1702.08608* .

Dua, D.; and Graff, C. 2020. UCI Machine Learning Repository. URL http://archive.ics.uci.edu/ml.

Elish, M. C. 2019. Moral crumple zones: Cautionary tales in human-robot interaction. *Engaging Science, Technology, and Society* 5: 40–60.

Gal, Y.; and Ghahramani, Z. 2016. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *Proceedings of the 33th International Conference on Machine Learning*, 1050–1059.

Goodfellow, I. J.; Shlens, J.; and Szegedy, C. 2014. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572* .

Guo, C.; Pleiss, G.; Sun, Y.; and Weinberger, K. Q. 2017. On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning*, 1321–1330.

Hoffman, R. R.; Mueller, S. T.; Klein, G.; and Litman, J. 2018. Metrics for explainable AI: Challenges and prospects. *arXiv preprint arXiv:1812.04608* .

Inselberg, A. 1985. The plane with parallel coordinates. *The Visual Computer* 1(2): 69–91.

Jiang, H.; Kim, B.; Guan, M.; and Gupta, M. 2018. To trust or not to trust a classifier. In *Advances in Neural Information Processing Systems*, 5541–5552.

Kamar, E. 2016. Directions in Hybrid Intelligence: Complementing AI Systems with Human Intelligence. In *Proceedings of the International Joint Conference on Artificial Intelligence*, 4070–4073.

Kendall, A.; and Gal, Y. 2017. What uncertainties do we need in bayesian deep learning for computer vision? In *Advances in Neural Information Processing Systems*, 5574–5584.

Kenny, E. M.; and Keane, M. T. 2019. Twin-systems to explain artificial neural networks using case-based reasoning: comparative tests of feature-weighting methods in ANN-CBR twins for XAI. In *Proceedings of the International Joint Conference on Artifical Intelligence*, 2708–2715.

Lai, V.; Carton, S.; and Tan, C. 2020. Harnessing Explanations to Bridge AI and Humans. *arXiv preprint arXiv:2003.07370* .

Lakkaraju, H.; and Bastani, O. 2020. " How do I fool you?" Manipulating User Trust via Misleading Black Box Explanations. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, 79–85.

Lakshminarayanan, B.; Pritzel, A.; and Blundell, C. 2017. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Advances in Neural Information Processing Systems*, 6402–6413.

Larson, J.; Mattu, S.; Kirchner, L.; and Angwin, J. 2016. How we analyzed the COMPAS recidivism algorithm. *ProPublica (5 2016)* 9(1).

Lee, J. D.; and See, K. A. 2004. Trust in automation: Designing for appropriate reliance. *Human factors* 46(1): 50–80.

Li, O.; Liu, H.; Chen, C.; and Rudin, C. 2018. Deep learning for case-based reasoning through prototypes: A neural network that explains its predictions. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

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

Lucic, A.; Haned, H.; and de Rijke, M. 2020. Why does my model fail? contrastive local explanations for retail forecasting. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, 90–98.

Lundberg, S. M.; and Lee, S.-I. 2017. A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems*, 4765–4774.

McKinney, S. M.; Sieniek, M.; Godbole, V.; Godwin, J.; Antropova, N.; Ashrafian, H.; Back, T.; Chesus, M.; Corrado, G. C.; Darzi, A.; et al. 2020. International evaluation of an AI system for breast cancer screening. *Nature* 577(7788): 89–94.

Nushi, B.; Kamar, E.; and Horvitz, E. 2018. Towards accountable ai: Hybrid human-machine analyses for characterizing system failure. In *Proceedings of the AAAI Conference on Human Computation and Crowdsourcing*.

Papernot, N.; and McDaniel, P. 2018. Deep k-nearest neighbors: Towards confident, interpretable and robust deep learning. *arXiv preprint arXiv:1803.04765* .

Rajendran, V.; and LeVine, W. 2019. Accurate Layerwise Interpretable Competence Estimation. In *Advances in Neural Information Processing Systems*, 13981–13991.

Ribeiro, M. T.; Singh, S.; and Guestrin, C. 2016. "Why should i trust you?" Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, 1135–1144.

Slack, D.; Hilgard, S.; Jia, E.; Singh, S.; and Lakkaraju, H. 2020. Fooling lime and shap: Adversarial attacks on post hoc explanation methods. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, 180–186.

Tan, S.; Adebayo, J.; Inkpen, K.; and Kamar, E. 2018. Investigating Human + Machine Complementarity for Recidivism Predictions. *arXiv preprint arXiv:1808.09123* .

Toreini, E.; Aitken, M.; Coopamootoo, K.; Elliott, K.; Zelaya, C. G.; and van Moorsel, A. 2020. The Relationship between Trust in AI and Trustworthy Machine Learning Technologies. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, FAT* '20, 272–283. New York, NY, USA: Association for Computing Machinery. ISBN 9781450369367.