Using Explainable Boosting Machines (EBMs) to Detect Common Flaws in Data

Zhi Chen[1,2(✉)], Sarah Tan[3], Harsha Nori[1], Kori Inkpen[1], Yin Lou[4], and Rich Caruana[1(✉)]

[1] Microsoft, Redmond, WA, USA
rcaruana@microsoft.com
[2] Duke University, Durham, NC, USA
zhi.chen1@duke.edu
[3] Cornell University, Ithaca, NY, USA
[4] Ant Group, Sunnyvale, CA, USA

Abstract. Every dataset is flawed, often in surprising ways that data scientists might not anticipate. However, popular machine learning methods are mostly black-boxes. Due to their lack of interpretability, they might learn defective knowledge from these datasets, which can be difficult to detect. In this work, we show how interpretable machine learning methods such as EBMs can help users detect problems that are lurking in their data. Specifically, we provide a number of case studies, where EBM discovers various types of common dataset flaws, including missing values, confounding and treatment effects, data drift, bias and fairness, and outliers. In each case study, we analyze the flaws using visualization of EBM shape functions combined with domain knowledge. We also demonstrate that in some cases interpretable learning methods such as EBMs provide simple tools for correcting problems when correcting the data is difficult.

Keywords: Interpretability · Generalized additive model · Debugging datasets · Model editing · Missing values · Treatment effects · Fairness

1 Introduction

Data is the center of the machine learning pipeline. As machine learning models are usually trained and evaluated on static datasets, they are encouraged to learn every detail in the dataset. However, the data collection process can never be perfect, resulting in pervasive flaws in almost all datasets. These flaws range from simple problems such as missing values [1], and data drift [8], to serious societal bias that could cause damages to the public [18]. In this case, machine learning models trained and tested on contaminated datasets may mistakenly learn the defective information.

Most widely deployed machine learning models, such as deep neural networks, gradient boosted trees, nonlinear SVMs, are all black-boxes whose prediction

M. Kamp et al. (Eds.): ECML PKDD 2021 Workshops, CCIS 1524, pp. 534–551, 2021.
https://doi.org/10.1007/978-3-030-93736-2_40

Table 1. Summary of datasets used in the paper

Dataset name	Target	Associated dataset flaws
Pneumonia [7]	Pneumonia mortality	Missing values, treatment effects
MIMIC-II [22]	ICU mortality	Missing values, treatment effects, data drift
COMPAS [11]	Defendant recidivism	Bias and fairness
Housing price[a]	Housing price	Outliers

[a]The dataset is proprietary

processes are highly complex and not interpretable by humans. Once these black-box models are trained on a flawed dataset, they might learn biased knowledge embedded in the data, which data scientists can struggle to detect. In fact, researchers have already found many unexpected problems in these black box models that originated from flaws lurking in the datasets [4,5,17]. Making the situation even riskier, we do not know how many more flaws are hidden in the dataset and in what way.

Recently, some high-accuracy and interpretable machine learning models have been proposed. Because of their interpretability, data scientists can examine what the models have learned from a dataset, and potentially also discover flaws in those datasets. Explainable Boosting Machines (EBMs) [6,15,16] in particular can achieve accuracy on par with the best black-box models. More importantly, the model itself is the sum of visualizable shape functions created for individual features (or their pairwise interactions), and these shape functions are often expressive enough to capture subtleties embedded in the datasets, especially for continuous features. This makes EBM an ideal base machine learning model to detect and analyze flaws in datasets.

In this work, we provide a series of case studies that show how EBMs can help users detect flaws that are lurking in their data, and in some cases potentially correct problems caused by these flaws. The dataset flaws we study includes missing values, confounding and treatment effects, data drift, bias and fairness, and outliers. For each type of common dataset flaw, we provide one or more examples of EBM shape function graphs that help us identify the problem. Table 1 is a summary of datasets used in this study and their associated flaw types. Through our case studies, we found that

1. EBM shape function graphs can be helpful in identifying various types of dataset flaws.
2. In many cases, users with domain expertise are needed to examine what the model has learned.
3. In some cases, EBMs provide simple tools for correcting problems in the models, when correcting the data is not feasible or too difficult.

The later sections are organized as follows. Section 2 briefly introduces the EBM model. Sections 3–7 use EBMs to identify one type of dataset flaw per section, and discuss the best approaches to handle these problems. In Sect. 8, we conclude experimental findings and discuss three possible directions for future study.

2 Explainable Boosting Machines

Suppose an input sample is denoted as $(\mathbf{x}, y)$, where $\mathbf{x}$ is the p dimensional feature vector and y is the target. Denote the j^{th} dimension of the feature vector as x_j. Then a generalized additive model (GAM), first introduced by [9], is defined as

$$g(E[y]) = \beta_0 + f_1(x_1) + f_2(x_2) + \cdots + f_p(x_p) \tag{1}$$

where β_0 is the intercept, $f_j's$ are the shape functions and g is the link function, e.g. identity function for regression and logistic function for classification. Since one can add any offset to f_j while subtracting it from β_0 or other shape functions, we usually set the population mean of f_j, i.e. $E_{x\sim\mathcal{X}}[f(x_j)]$ to 0. Note that, each shape function f_j only operates one single feature x_j, and thus the shape function can directly be plotted. This makes GAMs interpretable since the entire model can be visualized through 2D graphs. In early work of GAM, the shape functions are usually modeled as splines with smoothness constraints. Explainable Boosting Machine (EBM) [15] formulates $f_j's$ as ensemble of trees using ensemble techniques such as bagging and gradient boosting. Incorporating tree based ensemble learning algorithms significantly improves the performance of GAM. EBM also outperforms traditional GAMs in terms of interpretability, as its shape function has more complexity to capture nuances hidden in the dataset. The GA^{2}M model further improves accuracy by adding a small number of pairwise interactions, i.e.

$$g(E[y]) = \beta_0 + \sum_{j=1}^{p} f_j(x_j) + \sum_{k=1}^{K} f_k(x_{k_1}, x_{k_2}) \tag{2}$$

in which K pairs of features (k_1, k_2) are chosen greedily (see the FAST algorithm in [16]). Including pairwise interactions will not affect the interpretability because the interaction terms can be visualized as heatmaps.

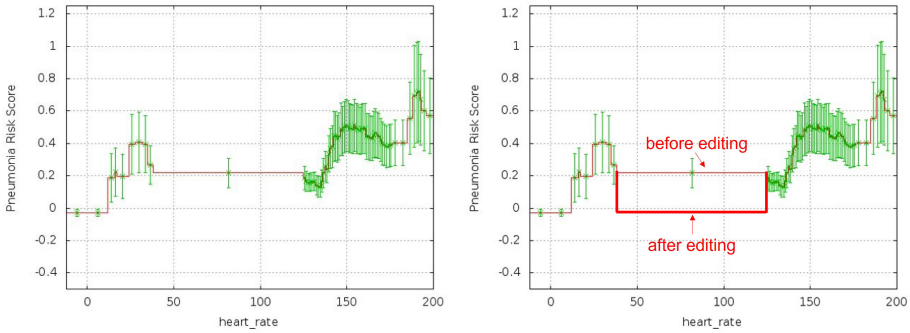
3 Missing Values

A variety of problems can arise when there are missing values in data. In this section, we explore a few of these issues, and show how interpretable models such as EBMs can be used to detect, and in some cases fix these problems.

3.1 Missing Values Assumed Normal

In some domains such as healthcare, it is common for feature values such as lab tests to be missing in the dataset because clinicians believe the patient is likely to be "normal" for this measurement, and thus the lab test is not performed. In other cases, the measurement may be made, but the value may not be recorded if it is within normal range—clinicians tend to focus on abnormal findings.

Figure 1 (a) shows what an interpretable EBM model has learned for predicting pneumonia mortality risk as a function of heart rate. As expected, risk is



(a) Original shape function (b) Edited shape function

Fig. 1. EBM shape function of "heart rate" for predicting pneumonia mortality risk.

elevated for patients with abnormally low (10–30) or high heart rate (125–200). The graph, however, shows a surprising region of flat risk between HR 40 and 125, which is considered to be normal heart rate for patients in a doctor's office. Moreover, the model surprisingly predicts patients who have *normal* heart rate are at *elevated* risk: it adds 0.22 to the risk for patients in this region.

On further inspection, it turns out that there are no patients in the data set with heart rates between 40 and 125, and 91% of patients are missing their heart rate which has then been coded as zero. In other words, there are no data to support the model in the normal range of HR 40–125, and instead the patients who would be in this range, are all coded as zero in the data and on the graph. This explains why the model predicts the lowest risk = -0.04 for patients with HR = 0, because these are the patients with the most normal heart rate.

Any model trained on this data (e.g., boosted trees, random forest, neural networks) is likely to learn to make similar predictions in this region because there is no data to support learning the correct risk in this range, and because most models will then learn to interpolate between the regions where they do have data. One exception might be Bayesian models with strong priors where the prior might dominate in regions of little or no data and cause predictions in this region to be closer to a baseline lower-risk value. However, even Bayesian approaches would not learn to predict the correct value in this region, but they might learn a less incorrect value. The key advantage of using interpretable models such as EBMs is that we can easily see these problems in the model, that were caused by problems in the data.

If the model will only be used to make predictions for patients where all heart rates in the range 40–125 will be coded as zero, then the model will make accurate predictions and the elevated risk predicted by the model in the range 40–125 will not be a problem because no patient will ever fall in that range. However, if the model might be used to make predictions for patients whose true heart rate would be coded in this region, the model will then make incorrect, possibly dangerous predictions for patients who have normal heart rate. Because

this is risky, it usually is important to correct this kind of problem. One might expect that the data scientist would detect this kind of problem in the data prior to training a model, however in our experience these kinds of problems can be difficult to detect in the raw data and are easier to detect once an interpretable EBM model is trained.

There are several ways to correct this kind of problem. Of course, the best approach would be to collect and record the true heart rates for all patients. Unfortunately, it is often not possible to go back and correct data in this way. An alternate approach would be to edit the data so that patients coded as zero are randomly assigned heart rates in the interval 40–125, i.e., impute the missing heart rates with a random value selected uniformly from the region where we believe most of the missing values arise from. This, however, does make the assumption that all missing values arise from this one region, and that no patients with low or high heart rate had a missing heart rate. An alternate approach is to use a more sophisticated method of imputing missing values such as random forest imputation [24]. As we will see in Sect. 3.2, imputing with the mean or median missing value is probably not recommended.

An alternate approach when interpretable EBM models is used is to directly edit the graph so that the region 45–120 predicts risk similar to the prediction the model has learned to make for patients with HR = 0. The resulting graph is show on Fig. 1 (b). This approach has the following advantages:

1. Editing shape functions provides an opportunity for experts to use their professional training to correct and improve models in ways that may not be adequately represented in the training data.
2. Editing the model can not only improve the accuracy of the model in the real world, but make the shape plots more "correct" and trusted by experts.
3. Editing an EBM shape function can be done without retraining the model.
4. Correcting the model by editing the data is often much more difficult.

3.2 Missing Values Imputed with the Mean

Because many machine learning methods can not deal with missing values, it is common for data scientists to impute missing values before training the model. There are many different ways to impute missing values: with the mean, the median, with a unique value such as 0 or -99 or +99, or by using a machine learning method such as random forest imputation. See [14] for an overview of imputation methods.

Perhaps the most common form of missing value imputation is with the mean. Figure 2 shows an EBM plot of the mortality risk of ICU patients as a function of their PFratio. PFratio is a measure of how well a patient converts O2 in the air they breathe into O2 in their blood: low PFratio indicates patients with low blood O2 whose lung function is impaired, while PFratio around 1000 and higher indicates normal lung function. As expected, the learned shape function captures this, and also shows interesting small jumps at clinically meaningful values such

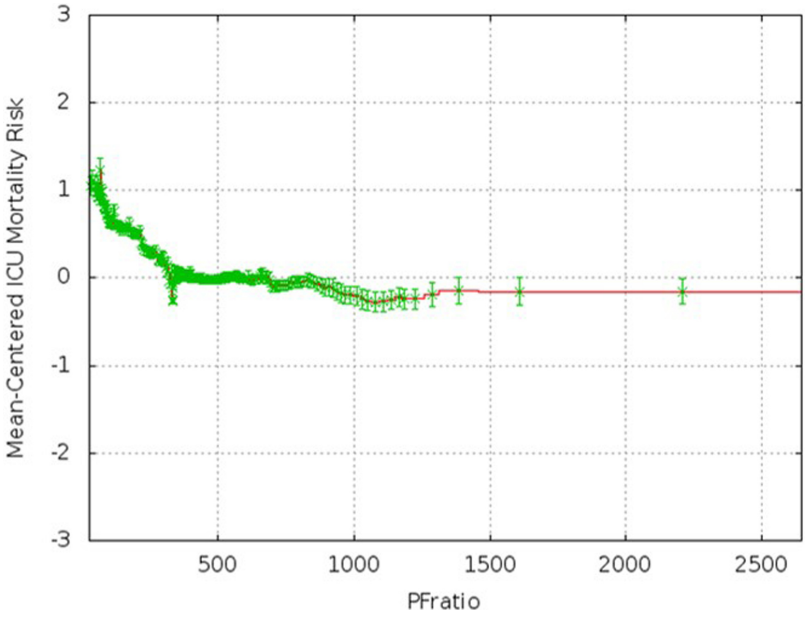


Fig. 2. EBM shape function of "PFratio" for predicting ICU mortality.

as 800, 700, 200, and 100. What is surprising, however, is the large drop in risk at about PFratio = 323. What could cause that?

A simple test for blood O2 levels is to pinch a fingertip and see how quickly color returns to the skin. If color returns quickly, clinicians know that the blood O2 is good and do not bother to measure PFratio—as discussed in the previous section, the PFratio is assumed normal. In this dataset, however, the missing PFratio values were imputed with the mean instead of being coded as 0 as they were in Fig. 1. In this dataset 60% of patients are missing PFratio. The mean PFratio when not missing (40% of the data) is 323.6, so 60% of patients have had their PFratio imputed with this value. Because this is a large sample of healthy patients with strong respiration, the model learns that their risk is comparable to the risk of other healthy patients with PFratio above 1000. This explains why the graph dips at 323, yet predicts higher risk just before and after this value. Although this anomaly does not significantly hurt the accuracy of the model because it has learned to make appropriate low-risk predictions for the 60% of patients at this value, it is risky to leave this anomaly in the model because there are real patients with PFratio≈323 who will be predicted to have low risk but who are genuinely at elevated risk. For this reason, it would be better to either use a more sophisticated method of imputing missing values such as random forest imputation, or to leave the missing value coded as a unique value such as 0. Model editing is not a good solution for this problem because imputation with the mean has caused patients who are low risk (missing values) and elevated risk

(PFratio near 323) to fall at the same place on the shape function, thus there is no edit to the graph that can predict the correct risk for both groups.

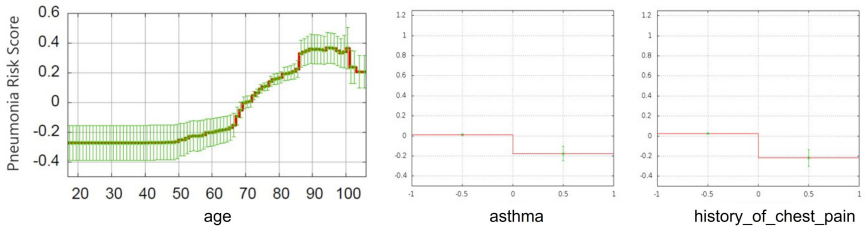
4 Confounding and Treatment Effects

Due to the complexity of the world, there will always be variety of confounding variables with chosen features in the dataset. Since many of these confounders are not included in the dataset, their treatment effects would be accounted by the chosen features, while the actual effects of the chosen features are contaminated. In this section, we first show how some of these confounders and treatment effects would affect the correctness of the model, and potentially cause serious problems. Based on this, we will also discuss how interpretable models like EBM might provide tools to correct these types of problems. In addition, we will also show through examples that in some cases if these treatment effects can be identified by the interpretable models, they can even help create new science.

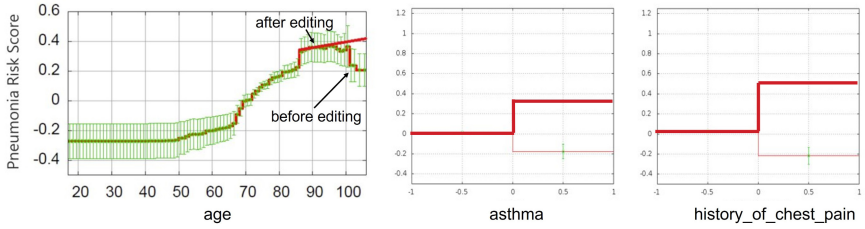
4.1 Treatment Effects and Model Correctness

In the medical domain, one of the most famous examples of treatment effects is that patients who have a history of asthma have lower pneumonia mortality risk than general population. This counterintuitive pattern was first found by rule-based model [2]. The pattern can be interpreted by the fact that patients with asthma history would be admitted directly into ICU and get more aggressive care, thereby lowering their risk of death.

Such treatment effects are pervasive in medical datasets. Figure 3 (a) shows three graphs with noticeable treatment effects, learned by an EBM model to predict pneumonia mortality risk. First, in the middle of Fig. 3 (a), the asthma effect has also been found by EBM. Similar to the asthma effect, shown in the right subfigure of 3 (a), EBM also discovers that a history of chest pain can reduce the mortality risk. The history of chest pain is highly related to heart attacks. Patients with a history of heart disease might confuse the earlier symptom of pneumonia with signs of a heart attack. Therefore, they might call ambulances or get admitted to emergency room as soon as possible, and consequently diagnose pneumonia early and get high-quality care. Another interesting example of treatments effects is the influence of age on the pneumonia mortality risk, shown in the left of Fig. 3 (a). The risk score remains low between age 18 to 50, and slightly goes up from age 50 to 67. Right around age 67, the risk increases rapidly. Interestingly, this tipping point coincides with the average retirement age in the US, suggesting that the increase of risk might be from retirement: things associated with retirement, such as changes in insurance provider and urgency of care, might cause risk to arise rapidly. This is yet another example of how confounders such as "retirement" might affect the behavior of features included in the dataset such as "age". In the age graph, there is another surprising treatment effect that the mortality risk suddenly drops (0.15 in log odds) near age 100. Such an effect is very unlikely to be true biologically. We suspect



(a) Original shape function graphs



(b) Edited shape function graphs

Fig. 3. EBM shape function graphs for predicting pneumonia mortality risk. From left to right are shape functions of features "age", "asthma" and "history of chest pain."

this is associated with social effects that doctors might try even harder to cure the patients if their age passes 100—given that pneumonia is relatively a treatable disease and centenarian are very rare even worldwide, doctors would not give up on these patients.

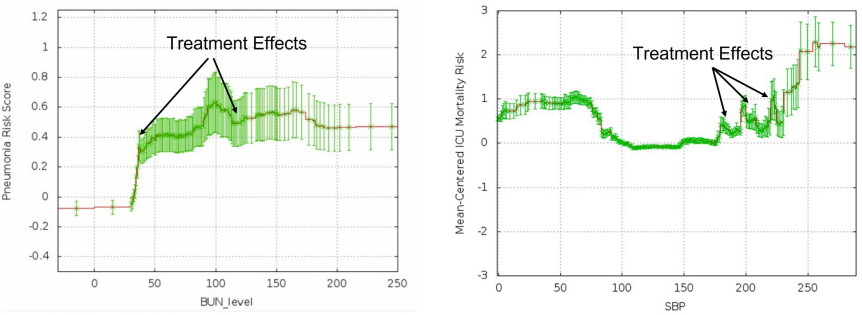
Many of these issues have already been mentioned by earlier works [6]. However, their impacts on model correctness and ways to fix them have not been carefully discussed yet. These treatment effects might be real in the dataset or even in practice due to the existence of confounders. Does it mean we should keep them in the model? Our suggestion is that *the model correctness depends on the purpose of the model*. For example, if the model is used by an insurance company to create insurance policy, then these effects might be fine. Because the model successfully captures the treatment effects that happen in reality, the insurance company can then use this model to adjust the insurance charges to patients and potentially make more profits. However, if the models can affect what care patients might receive, using these datasets to train models can be problematic. Certainly, we should not expect doctors to give mild or no care to a patient who is 105 with asthma and a history of chest pain only because the model predicts they have low mortality risk. There is a dangerous feedback loop behind this problem: while the patients' risks are lower because we give them extra treatment, if we then use the predicted risks to determine whether they would get extra treatment, the treatment effects will be removed and their mortality risks increase.

How should we fix these problems caused by confounders and treatment effects in the dataset? One approach would be to include confounding variables that reflect the treatments given to patients as additional features in the model. Sometimes this allows the model to learn that the reduction in risk is caused by the treatment and not the condition that caused the treatment to be given. In some cases, however, treatments and conditions are so tightly linked that models can not distinguish treatments and conditions reliably. Moreover, detailed information about treatments is not always available. Collecting new or additional data seems reasonable, but can be very hard in practice. First, collecting medical data can be very expensive itself, since the samples have to be real cases. More importantly, it could be unethical or impossible to collect the data that could help fixing the problem. For example, in order to fix the asthma problem, doctors need to randomly withhold the extra treatment asthma patients receive, which would be risky to those patients. Also, fixing the drop at age 100 would require doctors treating centenarians equivalently to other patients, which is almost impossible since that requires controlling doctors' care. One might think the problems can be solved if we remove the asthma feature from the model. Unfortunately, this usually will not solve the problem. The bias comes from the target rather than the input features. Since correlations are common between medical features, e.g. history of heart disease and body weight, even if the asthma feature is removed, the treatment effects might be learned implicitly through other correlated features. Even worse, such signals can be distributed in multiple features and become impossible to monitor. Thanks to the interpretability of the EBM model, we can directly fix the problem by editing the graphs. Figure 3 (b) shows some examples of edited graphs that could be more reasonable biologically. For example, having asthma and history of chest pain would be edited to have higher risk than normal. Also, for patients with age>85, we might want to edit the graph such that their risk goes up slowly and monotonically. Note that model edits are high-stakes decisions, and therefore should be based on interaction with domain experts such as doctors. Doctors would also need to monitor the effects of the models and most likely adjust the edited graph based on real-world feedback.

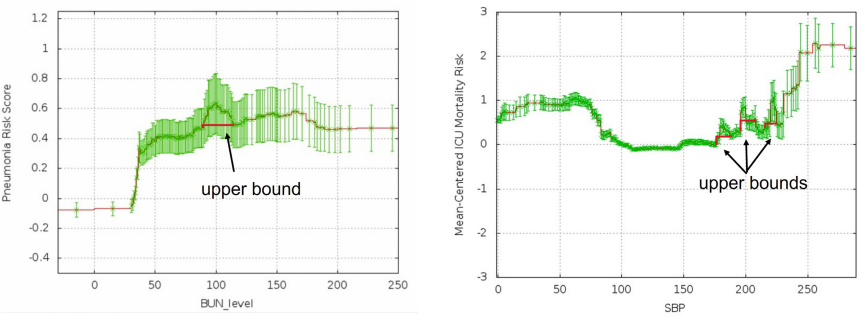
4.2 Discovering New Science

In Sect. 4.1, we show many negative examples where confounding and treatment effects might produce wrong and dangerous models. However, not all treatment effects are harmful. Sometimes useful knowledge associated with the treatment effects exist that domain experts were not aware of.

The left part of Fig. 4 (a) shows the EBM shape function of predicting pneumonia death risk with blood urea nitrogen (BUN) level. Lower BUN level is believed to be healthy which is clearly reflected in the figure as low risk. Interestingly, the figure also discovers that although the risk of death rises at BUN level around 30, the risk curve gets flat when BUN level $\approx$ 50. In fact, most doctors start to give patients medication if their BUN hit 50. This means the patients are getting effective medications and that flattens their risk of death.



(a) Original shape function graphs



(b) Upper bound of shape function graph if more aggressive treatments happen earlier

Fig. 4. EBM shape function graphs with treatment effects. Left: predicting pneumonia mortality risk using blood urea nitrogen (BUN) level; right: predicting ICU mortality risk using systolic blood pressure (SBP).

Another point of interest on the graph is the peak at BUN level 100: the risk continues to get higher after 50 and starts to decrease at 100. Notice that 100 is a "round number" at which doctors would start to give patients more aggressive treatment. In practice, when the patients' BUN level exceed 100, doctors would start to give them dialysis, which is a process of purifying the blood of the patient. Dialysis treatment quickly reduces BUN and accordingly reduces mortality risk. Some doctors may not give dialysis starting at 100 which accounts for why the risk declines until 120.

Similar effects can also be found in the shape function of predicting ICU mortality with blood pressure (right of Fig. 4 (b)). The risk curve peaks at several "round numbers" such as 175, 200 and 225. Treatments might be provided by the doctors to lower the patients' mortality risk and thus confounding the nature risk of high blood pressure.

So far, although we are able to explain when doctors give different treatments, these findings are not significantly different from the treatment effects described in Sect. 4.1, since they essentially just match with what doctors already know.

What if we take further steps to reason counterfactually on what could happen if we move the treatment threshold for more aggressive treatment earlier, e.g. move the dialysis threshold from 100–120 down to 80? How would such a move influence the shape function graph? To reason counterfactually without collecting new data, we should make some additional assumptions. Take BUN level as an example, we propose two reasonable assumptions

1. Getting dialysis would not increase the risk of death.
2. Given the same treatment, the patients' mortality increase monotonically with BUN level.

Given these two simple assumptions, the upper bound of the graph when moving dialysis treatment threshold to 80, is shown in the left of Fig. 4 (b). The modified part on the graph is *at most* a step function with max value equals the risk at 120—the two ends of the modified part have at most the same risk as the original graph (assumption 1); no middle points has risk higher than the right end (assumption 2). This is just the most conservative guess. In practice, the graph might be a U-shape curve that is under the red line. Given this new graph, we can estimate how many lives could be saved based on total number of pneumonia cases, proportion of sample fall in this region of graph, and the change of probability of death between the red line and the original curve. Surprisingly, even the most conservative estimation suggests we could save 2500 lives per year in just United States.

Similarly, we can infer about what could happen to ICU mortality risk if we change the treatment criterion based on blood pressure. Shown in the right of Fig. 4 (b), the three bumps caused by "inappropriate" treatment thresholds can be least flattened to the red lines. This could save another thousands of lives. Note that, these are just estimations based on correctness of the model and assumptions of the treatment. Rigorous clinical trails are needed to examine if these estimations are correct.

5 Data Drift

Data drift means the statistical property of the data might change over time, and thus making the model trained on the dataset outdated. In this section, we show when training on newer data, how EBM can be used to correct data drift problems in outdated models.

Figure 5 compares two models, on how much risk does "having AIDS" add to the ICU mortality rate. One model, shown as the red bar, is the risk score given by a EBM trained on the MIMIC-II dataset. The other model, shown in the blue bar, is SAPS-II model [12], which is a widely used scoring system for predicting ICU mortality risk. The risk scores predicted by two models, however, completely disagree with each other. SAPS model suggests that if a patient has AIDS, it adds 1.3 to the log odds of the mortality probability, or on average equivalent to adding around 12% to the risk. In fact, this is one of the most important factor in the SAPS-II model. The EBM model, while training on dataset with

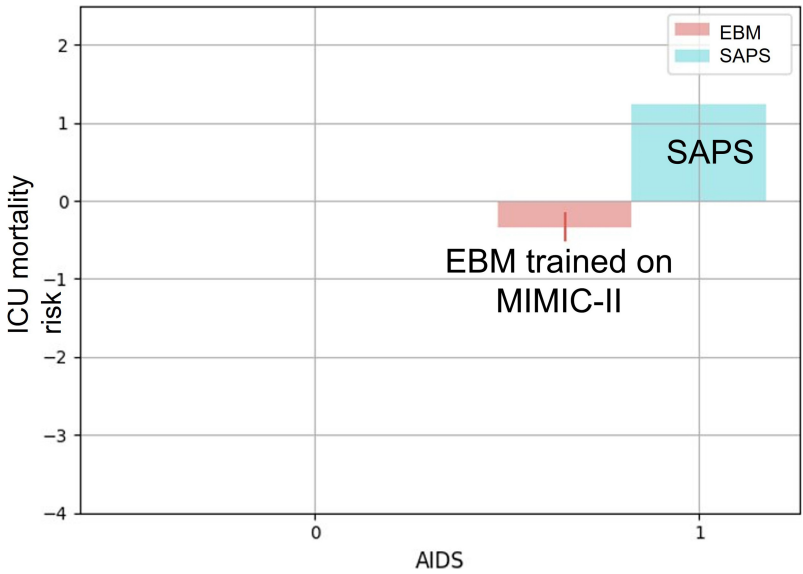


Fig. 5. Comparing contributions of AIDS to ICU mortality risk.

the same purpose and similar feature set, believes the log odds of the mortality probability of a patient diagnosed with AIDS should subtract 0.3 points. Why are these two models have such a large discrepancy on the influence of AIDS: one predicting AIDS to be one of the highest single risk factors while the other model predicting that AIDS is not too bad for the patient?

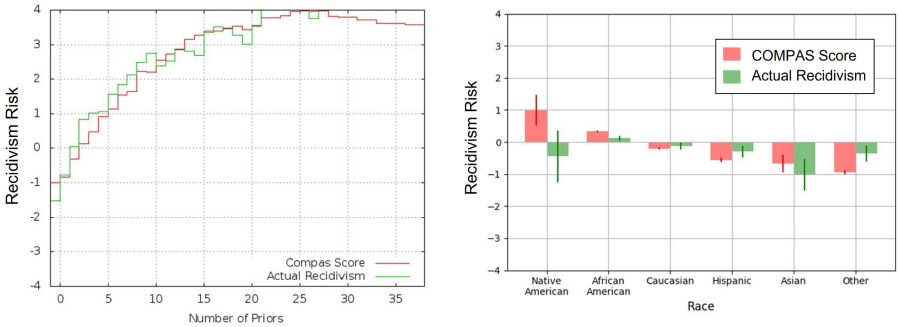
In fact, the discrepancy is because of progress in medicine. The SAPS model is created in 1993, during which AIDS was often considered as a terminal illness without effective treatments. Especially if the patients were in the ICU, they would have a large chance to be in near terminal stage of HIV. However, the data in MIMIC-II used to train EBM was collected between 2001 and 2008. By then, effective treatments such as HAART and Combivir became the standard, making HIV gradually become a chronic illness rather than a terminal illness. Since the death rate of HIV decreased significantly, there could be much worse reason for why the patient stays in ICU, and therefore making "having AIDS" reduce mortality risk.

This shows why interpretability is important in dealing with data drift. Because the prediction process of the model is transparent, one can compare what the model learns from earlier data and what it learn on new data, and identify how the model changes when the world changes. Particularly, this case study reveals that, although SAPS is still widely used, it is a stale model that does not represent modern health care. Doctors might want to abandon this model and replace it with interpretable models trained on up-to-date dataset. However, if the model is not interpretable, we can never monitor the changes of the model, and thus cannot tell whether the model goes stale and should be replaced.

6 Bias and Fairness

Biases, especially racial and gender biases, are well known dataset flaws that have drawn significant public attention. By learning or even amplifying the biases in the data, machine learning models might cause serious damage to society. Biases have been found in machine learning models used for different application domains, such as criminal justice [17], image super-resolution [19], word embedding [4] etc. For more systematic discussions on bias and fairness issues in machine learning, please see [18]. In this section, we show how EBM can help identify racial biases in datasets and models, and possibly help fix the biases.

One famous example of racial biases in machine learning model is the COMPAS (Correctional Offender Management Profiling for Alternative Sanctions) model which tries to predict the a defendant's risk of recidivism, i.e. recommitting a crime. Judges use COMPAS to help make parole decisions. Many other previous studies have been also done to analyze this model [10,21]. However, investigation on this black-box model suggests that it is likely to predict higher recidivism risk for African American than Caucasian [11,17]. In this work, we train EBM models on the dataset associated with the COMPAS model, which contains recidivism outcomes of criminal defendants in Broward County, Florida. The dataset has both the COMPAS predicted scores and the actual labels for recidivism in two years. We train two EBMs, one mimicking the COMPAS model, the other fitting the true labels.



(a) Shape function of "prior crimes"

(b) Shape function of "race"

Fig. 6. EBM shape functions for recidivism prediction. Red bars: EBM approximating the COMPAS model prediction; Green bars: EBM trained on the actual recidivism label. (Color figure online)

Figure 6 compares the shape functions of two EBMs. In Fig. 6 (a), we compare the shape functions of "number of prior convictions". For the effect of "number of prior convictions", the two models seem to agree with each other. Since the "number of prior convictions" is a very important feature, the result might suggest that the EBM mimics the COMPAS accurately. Figure 6 (b) shows the shape

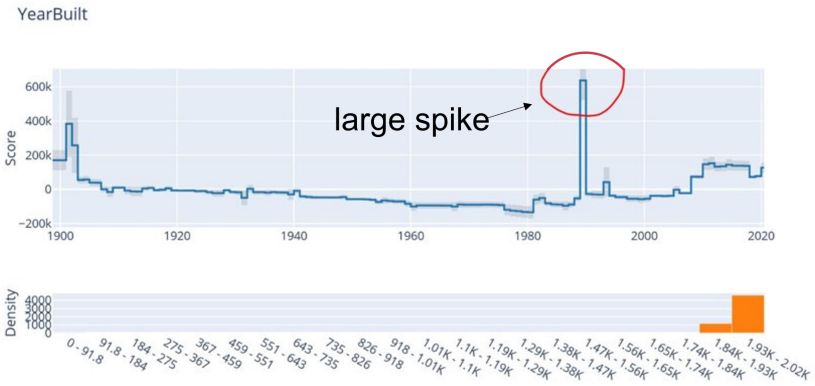
functions of the sensitive feature "race". The EBM graphs suggest that racial biases exist in both the dataset and the COMPAS model, e.g. African Americans are predicted to have higher risk than Caucasians. However, the two models disagree on the scale of the bias. The mimicked COMPAS model predicts African American having even higher risk and Caucasian having lower risk than the predictions of EBM trained on true labels. This suggests that COMPAS might even amplify the bias in the dataset. Surprisingly, the largest discrepancy of the two models is on Native Americans. While EBM trained on true label predicting Native Americans have low recidivism risk, the mimicked COMPAS model believes they have very high risk. We suspect that these discrepancies might be because the COMPAS is trained on a larger dataset containing nationwide data, and that there is a population difference between defendants nationwide and defendants in Broward County.

Such biased models should not be deployed, and these biases need to be fixed. This raises the question of how to remove biases in the dataset. One possible solution is to remove sensitive features like gender and race, or even make them inaccessible. However, as mentioned in Sect. 3, such bias comes from the target rather than the input features. Even if sensitive features like "race" are removed, the model can learn their effect via their correlated features. For example, in the US, race is highly correlated with zip code, education and income. If we remove the sensitive feature, the bias would spread in complex ways among other features, and thus becomes impossible to fix or even detect. Therefore, we suggest to first learn the biases with interpretable model like EBM. After letting the biases get concentrated in the model, we can edit the graphs to "zero out" the learned effects for sensitive features, thus mitigating the learned bias.

For certain types of biases, re-collecting data might be helpful. For example, for facial recognition dataset, one can collect more samples for the minority groups. However, in many cases, such as the recidivism example, biases can be impossible to fix by re-collecting the data, as the bias is rooted in complicated societal problems.

7 Outliers

Figure 7 (a) shows the "year of built" shape function of EBM trained on a proprietary housing price prediction dataset. We notice that an anomaly (huge spike) appears at year 1989 which adds +$600k to the housing price. Surprisingly, the error bar of the 1989 bin is relatively small, which means the bin has relatively large support. Why is there such a huge spike in the shape function? We search though the training set and select all the records with year 1989, see Fig. 7 (b). Among the 8 selected samples (year = 1989), 7 samples have sold price of exactly $8,094,000. This completely explains the existence of the anomaly in the EBM shape function. Interestingly, they are all condos and have the same zipcode. Although they do differ in number of bedrooms and bathrooms, their house sizes are all very small and their lot sizes are almost the same. We do not know if these records are real. We suspect that these records might be included



(a) Shape function of "year built"

	SoldPrice	NEW House Type	NEW Zipcode	Bedrooms	Bathrooms	HouseSizeSqm	LotSizeSqm	YearBuilt	New City
58799	8094000	Condo/Coop/Timeshare	98136	1	1.00	50.91	1375.93	1989	Seattle
58798	8094000	Condo/Coop/Timeshare	98136	1	1.00	50.91	1375.93	1989	Seattle
58797	8094000	Condo/Coop/Timeshare	98136	1	1.00	48.31	1375.93	1989	Seattle
58789	8094000	Condo/Coop/Timeshare	98136	2	2.00	70.98	1393.17	1989	Seattle
58788	8094000	Condo/Coop/Timeshare	98136	2	2.00	70.61	1393.17	1989	Seattle
58787	8094000	Condo/Coop/Timeshare	98136	2	2.00	66.89	1393.17	1989	Seattle
58786	8094000	Condo/Coop/Timeshare	98136	1	1.00	47.94	1375.93	1989	Seattle
53120	1940000	Single Family	98102	4	3.00	318.66	340.68	1989	Seattle

(b) Outliers accociated with the spike at 1989

Fig. 7. Outliers and their effects on EBM shape function for predicting housing price.

into the dataset accidentally, e.g. the values of certain columns of one record get overwritten on that of the other records. But regardless of why these outliers are included in the dataset, models trained on these outliers should be corrected, as anomalies caused by these outliers can make the model less robust in the real world. Such outliers could be detected with simpler statistical methods if one knew in advance exactly what to look for. EBMs help outlier detection by making outliers that are not expected easy to find.

The capability of identifying anomalies in the dataset also makes the model easier to deal with security threats such as data poisoning attack [3,13]. In data poisoning attack, the attacker is capable of modifying part of the training data, for example by providing a poisoned public dataset or by compromising the server storing the dataset. After a small fraction of training data being modified, the machine learning model trained on the dataset would make predictions that benefit the attackers. Taking the housing price dataset as an example, although we have no evidence that this is a data-poisoning attack, by including 7 samples with high sold prices ($8,094,000) and the same year of built (1989), the model would overestimate the price of all houses built in 1989. However, adding 7 training samples would not significantly affect the test accuracy of the model,

and thus data scientists training black-box models on the dataset might not detect it. Suppose the model was deployed to estimate the housing price but data is not available to costumers, it would benefit all the house owners whose houses are built in 1989, which might include the attacker. Nevertheless, as shown in previous paragraphs, EBM can help data scientists detect such anomalies in the dataset, since the anomalies might be identifiable by the shape function plots. Note that, many detection methods of data poisoning attack have been proposed, including methods that are based on ourlier and anomaly detection [20,23]. Systematic studies are need to prove the effectiveness of detecting data poisoning attack using EBM.

8 Discussion

We present a series of case studies on detecting common data flaws using EBM shape function graphs. From the results, we show that EBM can help data scientists identify common flaws in the dataset. In some cases, EBM even provide simple tools to fix the problems in the models introduced by data flaws. This is extremely helpful when re-collecting data is expensive or impossible, as discussed in Sect. 4 and 6.

Most flaws discussed in this paper are based on identifying bumps or sudden changes in the EBM shape function. Although most of these findings could be supported by domain knowledge, these changes in the shape functions might also come from noise, overfitting or correlation with other variables. To get more reliable results in the future, rigorous hypothesis testing methods for changes in the EBM shape functions are needed. Currently, the uncertainty quantification of EBM solely relies on calculating the standard deviations over all bootstrapped models, which tends to underestimate the uncertainty. Also, joint distributions are needed to test whether changes are significant in the plot, since scores before and after the change are not independent.

In terms of model editing, many important problems need to be investigated. For example, how to work with domain experts to edit models? Once we have an edited model, should we only show the edited model, or both the original and edited model? Also, if the model has been edited multiple times, how to present the editing history to the users, or even redo edits when the model is retrained?

In our case studies, we show that EBM shape functions can help detect common flaws in data such as treatment effects, improper handling of missing values, racial bias and outliers. However, other methods have been developed to address some these problems and future work is needed to systematically compare EBMs with these existing methods.

References

1. Acock, A.C.: Working with missing values. J. Marriage Family **67**(4), 1012–1028 (2005)

2. Ambrosino, R., Buchanan, B.G., Cooper, G.F., Fine, M.J.: The use of misclassification costs to learn rule-based decision support models for cost-effective hospital admission strategies. In: Proceedings of the Annual Symposium on Computer Application in Medical Care, p. 304. American Medical Informatics Association (1995)
3. Barreno, M., Nelson, B., Joseph, A.D., Tygar, J.D.: The security of machine learning. Mach. Learn. **81**(2), 121–148 (2010). https://doi.org/10.1007/s10994-010-5188-5
4. Bolukbasi, T., Chang, K.W., Zou, J.Y., Saligrama, V., Kalai, A.T.: Man is to computer programmer as woman is to homemaker? debiasing word embeddings. Adv. Neural Inf. Process. Syst. **29**, 4349–4357 (2016)
5. Buolamwini, J., Gebru, T.: Gender shades: Intersectional accuracy disparities in commercial gender classification. In: Conference on Fairness, Accountability and Transparency, pp. 77–91. PMLR (2018)
6. Caruana, R., Lou, Y., Gehrke, J., Koch, P., Sturm, M., Elhadad, N.: Intelligible models for healthcare: Predicting pneumonia risk and hospital 30-day readmission. In: Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 1721–1730 (2015)
7. Cooper, G.F., et al.: Predicting dire outcomes of patients with community acquired pneumonia. J. Biomed. Inf. **38**(5), 347–366 (2005)
8. Gama, J., Žliobaitė, I., Bifet, A., Pechenizkiy, M., Bouchachia, A.: A survey on concept drift adaptation. ACM Comput. Surv. (CSUR) **46**(4), 1–37 (2014)
9. Hastie, T., Tibshirani, R.: Generalized additive models: some applications. J. Am. Stat. Assoc. **82**(398), 371–386 (1987)
10. Kleinberg, J., Mullainathan, S., Raghavan, M.: Inherent trade-offs in the fair determination of risk scores. arXiv preprint arXiv:1609.05807 (2016)
11. Larson, J., Mattu, S., Kirchner, L., Angwin, J.: How we analyzed the compas recidivism algorithm. ProPublica **9**(1) (2016)
12. Le Gall, J.R., Lemeshow, S., Saulnier, F.: A new simplified acute physiology score (saps ii) based on a European/north American multicenter study. Jama **270**(24), 2957–2963 (1993)
13. Li, B., Wang, Y., Singh, A., Vorobeychik, Y.: Data poisoning attacks on factorization-based collaborative filtering. Adv. Neural Inf. Process. Syst. **29**, 1885–1893 (2016)
14. Lin, W.-C., Tsai, C.-F.: Missing value imputation: a review and analysis of the literature (2006–2017). Artif. Intell. Rev. **53**(2), 1487–1509 (2019). https://doi.org/10.1007/s10462-019-09709-4
15. Lou, Y., Caruana, R., Gehrke, J.: Intelligible models for classification and regression. In: Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 150–158 (2012)
16. Lou, Y., Caruana, R., Gehrke, J., Hooker, G.: Accurate intelligible models with pairwise interactions. In: Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 623–631 (2013)
17. Mayson, S.G.: Bias in, bias out. YAle lJ **128**, 2218 (2018)
18. Mehrabi, N., Morstatter, F., Saxena, N., Lerman, K., Galstyan, A.: A survey on bias and fairness in machine learning. arXiv preprint arXiv:1908.09635 (2019)
19. Menon, S., Damian, A., Hu, S., Ravi, N., Rudin, C.: Pulse: Self-supervised photo upsampling via latent space exploration of generative models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2437–2445 (2020)

20. Paudice, A., Muñoz-González, L., Gyorgy, A., Lupu, E.C.: Detection of adversarial training examples in poisoning attacks through anomaly detection. arXiv preprint arXiv:1802.03041 (2018)
21. Rudin, C., Wang, C., Coker, B.: The age of secrecy and unfairness in recidivism prediction. Harvard Data Sci. Rev. **2**(1), 1811 (2018)
22. Saeed, M., Lieu, C., Raber, G., Mark, R.G.: Mimic ii: a massive temporal ICU patient database to support research in intelligent patient monitoring. In: Computers in Cardiology, pp. 641–644. IEEE (2002)
23. Steinhardt, J., Koh, P.W., Liang, P.: Certified defenses for data poisoning attacks. In: Proceedings of the 31st International Conference on Neural Information Processing Systems, pp. 3520–3532 (2017)
24. Stekhoven, D.J., Bühlmann, P.: MissForest-non-parametric missing value imputation for mixed-type data. Bioinformatics **28**(1), 112–118 (2011). https://doi.org/10.1093/bioinformatics/btr597