FICO Explainable Machine Learning Challenge

Creating visual explanations to black-box machine learning models

Steffen Holter
NYU Tandon School of Engineering
Brooklyn, NY 11201, USA
sh3628@nyu.edu

Oscar Gomez
NYU Tandon School of Engineering
Brooklyn, NY 11201, USA
oag229@nyu.edu

Enrico Bertini
NYU Tandon School of Engineering
Brooklyn, NY 11201, USA
eb2488@nyu.edu

This project focuses on increasing interpretability of machine learning models by creating a multifaceted black-box model explanation. By combining instance level explanations and a general global model interpretation we are creating an interactive application to visualize the logic behind each of the model's decision. The solution identifies the most important features contributing to a single decision and suggests the minimal set of changes needed to flip the model's output. The application will be tested as part of the Explainable Machine Learning Challenge launched by FICO. By using their authentic home equity data set the results can be evaluated by relating it to intuitive features.

I. Introduction

Recent advances in machine learning have allowed for the creation of models with great predictive accuracy in a variety of applications. However, the complexity of such models make understanding and interpreting them difficult, and it is often the case that neither the trained model nor its individual predictions are readily explainable. This poses a considerable problem for work concerning high-risk data sets and sensitive decisions where reliance on only the model's output is not feasible. Fields such as medicine require understanding the underlying logic behind each prediction as every decision can have serious and longstanding implications.

Similarly, even with their great predictive potential, complex machine learning solutions are struggling in finding widespread acceptance in the financial sector where the lack of explicability makes it difficult to fulfill the industry's regulatory requirements. To incentivize research in this area, FICO has launched a challenge with a home equity credit data set where the objective is to create models that are both accurate and interpretable.

While white-box analysis techniques that allow for straightforward human interpretation are available, they are usually limited to simple models that cannot achieve the accuracy of more complex ones such as Support Vector Machines (SVMs) or Deep Neural Networks (DNNs). Thus, our project takes a more widely applicable approach which considers the model as a black-box. This allows for the solution to be applied to a model of any type and is not constrained to a particular solution.

II. Data Processing

The raw data set provided as part of the FICO challenge included a variety of complicating factors such as a sporadic distribution of special values and the inclusion of a few categorical features in a predominantly continuous data set. The negative effect of these characteristics on training our model was corroborated by the poor initial accuracy of 68% which was achieved when training on the unaltered data. To ensure more comprehensive results the data was thoroughly processed before training the model.

A. Special value -9

According to the explanations provided with the home equity data set the special value of -9 was assigned to those fields for which no credit history or score information was available. Within the data set most of these values occurred together in samples where every single feature had such a value. In other words, there was a total of 588 features for which all the fields where filled with the -9 special value. Since no information can be deduced from such instances because they simply act as noise, we decided to omit these samples when training the model.

There were also ten cases when a -9 value occurred in the External Risk Estimate feature column indicating that no record was found from an external source. For these instances we opted for the use of simple linear regression to extrapolate the values from the remaining data points. To ensure no label leaking occurs the target column was excluded for this process.

B. Special value -8

The special value -8 indicated that no usable or valid cases were found. This meant that the accounts, trades and inquiries in question were either inactive or very old.

To replace these values with a feasible prediction we applied the k-nearest neighbors (k-NN) algorithm to perform regression. Since the all the feature columns with -8 values are continuous, k-NN regression allows for a simple prediction of what the value would have been had it not expired. In addition, this method is effective in handling the noisy data. For our purposes we achieved the most accurate results with 5 neighbors and a mean weighted approach.

C. Special value -7

According to the explanation dictionary the -7 value was attributed to cases when the sample did not meet a specific requirement, or a condition was not met. For example, this was evident in cases where the time that had elapsed since a previous delinquency was considered. In an instance where the person had never been delinquent a -7 value was assigned. The special value -7 was the most troublesome when training the model due to its negative value directly contradicting with the monotonicity of the features it occurred in. Continuing with the same example where the months since last delinquency was examined, it is clear that the feature is monotonically decreasing yet the best case where no delinquency has occurred is located at a negative value.

In an attempt to ensure the special value conforms to the feature's general monotonic constraints, the -7 special value was moved to a positive large value. After attempting a regression based algorithmic approach to identify what this new value should be, the results remained inconclusive due to the variability of each sample. Regression would have introduced one additional level of dimensionality which is not feasible. As a make shift solution, we used a trial and error based approach to test the model's response to a wide range of large positive values. The training accuracy of the model peaked when the -7 value was replaced by 150 which makes sense in the context of the problem. However, even though it worked in this case a future version of the solution should aim to find a more algorithmically rigorous way of identifying the new value.

D. Categorical features

In addition to the special values a problematic aspect of the original data set was the occurrence of two categorical feature columns amongst overwhelmingly continuous data. The two features labeled *MaxDelq2PublicRecLast12M* and *MaxDelqEver* both had values ranging from 1-9 with each one symbolizing a mix of cases. The problem was that while some of the values adhered to a given monotonic constraint there were also some other numbers that simply corresponded to specific situations. In addition, while these two features had the same nine explanations the values they corresponded to in each of the columns were different.

To help rectify this problem we mapped the values such that both columns follow the same structure. In addition, we attempted to order the values in a manner that allowed all of the categories to fulfill the increasing monotonic constraint. The new values for both columns after mapping can be seen in Table 1 below.

TABLE I. RESULT OF MAPPING DATA

Meaning	***Value***
Derogatory comment	0
120+ days delinquent	1
90 days delinquent	2
60 days delinquent	3
30 days delinquent	4
Current and never delinquent	5
Unknown delinquency	6
All other & No such Value	7

III. MODEL & ALGORITHMS

A. SVM model with Linear kernel

Our solution aims to be model independent, however, for the purpose of testing our prediction results as well data pre-processing we evaluated a variety of different models. The most success was achieved by using a Support Vector Machine (SVM) model with a Linear kernel. While options such as Random Forests produced promising results, they lacked the overall accuracy achieved by the SVM.

The initial test accuracy of the model when evaluating an entirely unaltered data set was approximately 68%. After performing the data processing decisions highlighted in the previous section a more adequate accuracy of 74.8% was achieved. Other improvements such as implementing feature selection and regularization failed to improve the results significantly. Since our interactive solution does not rely on the accuracy of the model this result of nearly 75% was sufficient for the project. Future work could examine in depth a wider selection of effective models and identify the optimal results for this specific data set.

B. Discretization of Data

In attempts to simplify the problem of interpretability we opted for the use of discretization. This meant that instead of having a varying continuous range for each unique feature all columns would be distributed into ten bins. To account for the variability in each individual property the bins were created in a range of two standard deviations below the mean to two above it. This meant that the bins contained the optimal majority of the data. Otherwise using a simple minimum to maximum range could have created a very uneven distribution due to a single large outlier. Using a fixed number of bins helps ensure a more manageable time complexity when analyzing each feature. In addition, such an approach also increases scalability if the data set were to be increased in size.

C. Key Features Algorithm

As one of the main components of our visual explanation we endeavored to identify the key features that the model uses to make any single decision. In other words, the solution would allow the credit specialist to see whether the model made the decision for the right reasons. Even though there are options that attempt to identify the weights of each of the features when making a decision such as Local Interpretable Model-Agnostic Explanations (LIME) [1], we instead opted for simply highlighting the features that are of paramount importance from the model's point of view.

This was done using a modified version of the Anchor's Algorithm created by M.Ribeiro [2]. The fundamental premise relies on the notion that if a so-called key feature is kept at its current level then it does not matter how the other values are changed as it would not alter the decision. Similarly, it would indicate that the fixed feature is the main reason for the model's decision.

The algorithm itself works by fixing one feature at a time and perturbing all the other columns by their respective Gaussians. For one fixed feature we decided to perturb 100 samples. Our

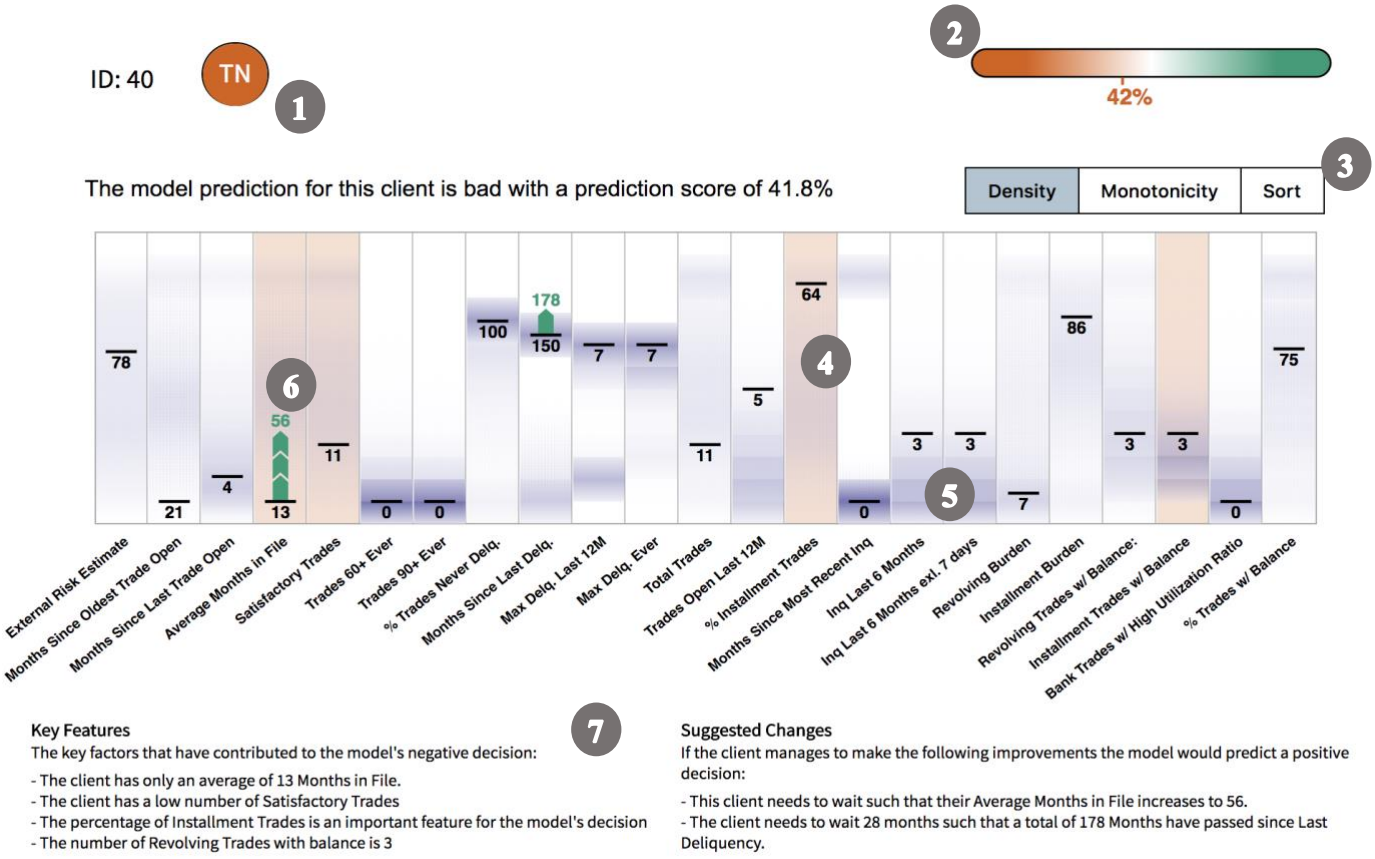

Figure 1: Demonstrating a single local explanation. (1) shows classification correctness (2) indicated model's percentage prediction (3) are buttons that allow modifying the arrangement (4) highlights a key feature for this decision (5) shows the density distributions (6) gives the minimum changes needed to reverse the decision

algorithm accepts a column as a key feature only if over 95% of the perturbations result in the same decision. For most instances a single feature does not eclipse this threshold in which case the highest percentage column is recorded, and the same process is attempted with combinations of the first feature and a new one. For our purposes we decided that if more than four features would need to be fixed then it can no longer be classified as a key feature, thus none of the local explanations have more than four key features.

D. Minimal Set of Changes

To supplement the solution, we found that for the client facing credit specialist it would be useful to understand what changes the client could make to get a different decision. This would mean that if the person got a positive prediction from the model the algorithm would give the specialist certain minimum values which cannot be passed for the decision to remain the same. Inversely, if a person did not pass, the algorithm would suggest the necessary changes need to achieve a passing mark. These suggestions are limited therefore some predictions that either have a very high or very low percentage cannot be reversed without making unrealistically large or too many changes.

The algorithm is constructed by moving between the discretized bins discussed above. Thus, the suggestion actually refers to a particular range the new value should fall in as opposed to a precise number. To identify these necessary changes the process uses a greedy approach. A one increment change is tested for all the features and the one which has the biggest effect on the percentage is chosen. This is repeated until the 50% mark is passed or the maximum number of feasible changes, which we set at four increments on a maximum of five features, is exceeded. To decrease time complexity, we relied on the monotonicity values provided with the data set to choose the direction of the incrementation. Only, if the monotonicity was unknown did the model check both directions.

To customize the solution for our data set and desired purpose we made some hard-coded limitations for the algorithm. Firstly, we limited incrementing to and from the top most bin as it is open ended and can include the large outliers. Secondly, the algorithm does not allow for changing the external risk estimate column as it is not realistically alterable in normal circumstance. In future versions we aim to add an option to interactively limit certain features to allow for the user to optimize the algorithm's operation based on their specific needs and situations.

IV. Local Explanation

The local explanations were created to understand more about the model's decision for each individual person as well as compare their specific results against the rest of the data set. As such, the resulting interactive solution, which can be seen in Figure 1, aims to provide a multilayered overview about any individual sample.

Figure 2: Demonstrating the global explanation when examining key features. (1) shows the given monotonicity of the feature (2) indicates the number of samples where this feature is key (3) is the selected feature (4) shows the combination of features used for explanation (5) highlights the best positive/negative examples with these key features (6) represents a single sample with its prediction percentage shown (7) is an individual segment (8) is a miniaturized version of the individual explanation which can be clicked to return to full local explanation

When constructing the visual solution, we considered the information the client facing credit specialist could quickly use to understand the decision made by the model as well as provide enough details that can be relayed to the client as a reason for accepting or declining the loan.

Firstly, the model generates a prediction percentage indicating the confidence that the person in question would successfully pay back their loan. Any value above 50% is classified as good. This is shown as a percentage bar with a color gradient to quickly place where the sample lies in the spectrum. To supplement this prediction the explanation also indicates whether this classification was correct or not.

Most of the instances indicate the key features that were paramount to the model's decision. A column that is deemed a key feature is highlighted in its respective color ((4) in Figure 1). In this example a negative decision is explained by a red background.

The solution also demonstrates the necessary changes where available. In Figure 1 the transformation needed to make a negative decision positive is shown. Each green polygon indicates one increment as described by the algorithm in the aforementioned section.

To allow for a more detailed analysis there are three buttons that alter the view. The density option adds a purple gradient to signify how the values are distributed across the data set. Darker areas signify ranges where many values are located amongst the samples. Orienting to account for monotonicity reorganizes the columns such that the so-called better value is placed at the top. This makes it easy to distinguish which features have good values. Finally, the sort function simply organizes the columns according to their relative value.

The last component of the individual explanation is a summarizing text explanation which combines all the aforementioned features into an easy to understand form. In addition, it provides, when available, a list of similar instances where the suggested change has already occurred and therefore the prediction is different.

V. Global Explanation

The aim of the global explanation is to aggregate all the individual results to a comprehensive visualization. This global view is constructed to benefit those who are interested in understanding the general operation of the model and its preferences when generating results. It is divided into three main panels with each one revealing additional information and eventually culminating in an option to view the local explanation itself [3]. To efficiently run this solution the program uses a pre-generated data file that has recorded the results for when the model was applied to each of the, over ten thousand samples, present in the data set. This raw data is then combined and visualized.

Since the individual solution can be effectively categorized into two main components: (1) the key features the model used to make the decision (2) the minimal set of changes that would result in the reversal of the decision. As such the global explanation also works in two views: one which analyses how the key features are distributed across the data set seen in Figure 2 and another that addresses the various combinations of changes that can occur seen in Figure 3.

The first panel is identical for both views showing a sorted list of feature names with a count bar indicating how many samples do these key features or changes occur in. The two colors in each bar demonstrate the relative occurrence of samples that are predicted to be good and bad. To add additional information a so-called monotonicity indicator is used to easily understand how each feature should be changing. For example, a green arrow indicates that the predictive percentage should

Figure 3: Demonstrating the global explanation when examining suggested changes. (1) shows the given monotonicity of the feature (2) indicates the number of samples where this feature is key (3) is the selected feature (4) shows the combination of features used for explanation (5) is the total number of samples with these features eliciting changes (6) represents all the samples where such a combination of changes is present and where opacity indicates occurrence (7) is an individual segment (8) is a miniaturized version of the individual explanation which can be clicked to return to full local explanation

increase as the value for that feature increases. Clicking on the feature name of interest reveals the next segment of the explanation. The solution also allows for multiple feature names to be selected if a specific combination of categories requires attention.

The middle panel is where the two views (key features and changes) differ due to the nature of the each of the explanations. In Figure 2 the view that opens for key features can be seen. The panel is split into a number of segments each characterized by a list of one to four feature names. These correspond to the selection made in the first panel and represent all the different ways the selected feature(s) combine with other features in the data set. It is automatically sorted by occurrence meaning that the most popular combinations are revealed at the top. Each segment also has small squares that populate the rows. Every single square represents a single sample, with the number indicating the model's prediction for that sample. These squares allow for the comparison of what types of cases occur for a certain combination of features. For examples, a lot of dark red squares suggest that these features predominantly act as key features for negative decisions. To get additional information about these samples one can click to reveal either all positive or negative cases ((5) in Figure 2).

The final panel reveals a compilation of small individual explanations in simplified form. In the key features view these correspond to the small squares in the middle panel. This allows for a quick overview into how these samples look and provides a means to easily compare a large number of cases. There is also an option to select a single sample and open the full local explanation.

The middle panel when using the minimal changes view can be seen in Figure 3. Similarly, to the previous view there are a number of segments corresponding to a particular sequence of features. Each of the squares represent a set of samples that all include a specific combination of changes. In other words, all the samples represented by the square have that particular change in their local explanation. Furthermore, the opacity of each of these squares is used to reflect the occurrence rate while the same color scheme is used to indicate positive and negative samples. To compare across the individual segments there is an overall count bar to reflect how many samples in total have changes in those particular feature columns. In addition, each of the squares is clickable and reveals the miniaturized versions of the local explanations in a way that is similar to those in the key features view.

In both views this hierarchy should allow the user to inspect in depth how the model makes combinations between features and understand the main relationships in the data set. It also acts as a very useful tool to find particular cases and traits in the local explanations.

VI. Conclusions and Future Work

In this paper we have explained the operation of a visual interactive solution that can be used understand the operation of any black-box model. The local explanation identifies the most important features contributing to a decision and suggests the minimal set of changes needed to alter the model's output. This is done by systematically perturbing a sample instance and measuring the resistance to change against a predetermined threshold, and by altering the feature values through a greedy procedure. The global view aggregates all the individual instances into an interactive multi-layered tool.

Further work on the project could be to expand it to other datasets such as multiclass classification problems, as well as incorporating other explanation algorithms that can complement the ones already implemented.

ACKNOWLEDGMENT

The authors thank NYU Tandon School of Engineering's Office of Undergraduate Academics and the NYUAD Undergraduate Research Program for generous funding of the project.

REFERENCES

[1] Ribeiro, Marco Tulio, Sameer Singh, and Carlos Guestrin. "Why should i trust you?: Explaining the predictions of any classifier." Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining. ACM, 2016

[2] Ribeiro, Marco Tulio, Sameer Singh, and Carlos Guestrin. "Anchors: High-precision model-agnostic explanations." AAAI Conference on Artificial Intelligence. 2018.

[3] Tamagnini, Paolo, et al. "Interpreting black-box classifiers using instance-level visual explanations." Proceedings of the 2nd Workshop on Human-In-the-Loop Data Analytics. ACM, 2017.