Actionable Model-Centric Explanations (Student Abstract)

Cecilia G. Morales, Nicholas Gisolfi, Robert Edman, James K. Miller, Artur Dubrawski

Auton Lab, Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh, PA 15213
{cgmorale,ngisolfi,redman,mille856,awd}@andrew.cmu.edu

Abstract

We recommend using a model-centric, Boolean Satisfiability (SAT) formalism to obtain useful explanations of trained model behavior, different and complementary to what can be gleaned from LIME and SHAP, popular data-centric explanation tools in Artificial Intelligence (AI). We compare and contrast these methods, and show that data-centric methods may yield brittle explanations of limited practical utility. The model-centric framework, however, can offer actionable insights into risks of using AI models in practice. For critical applications of AI, split-second decision making is best informed by robust explanations that are invariant to properties of data, the capability offered by model-centric frameworks.

Introduction

Artificial Intelligence (AI)-driven decision making is increasingly used to support human decisions. In practice, the adoption of intelligent systems hinges on the ability of users to understand why a prediction was generated and how to ensure a desired output. Trust and transparency in AI is essential in high-stakes application domains, including healthcare, where wrong decisions may bear grave consequences. Common explanatory tools, including Local Interpretable Model-Agnostic Explanations (LIME) (Ribeiro, Singh, and Guestrin 2016) and Shapley Additive Explanations (SHAP) (Lundberg and Lee 2017), are data-centric. They assess contributions of individual attribute values to predictive performance of the models. Insights gleaned from such analyses are primarily of confirmatory value; a clinician can confirm that the model pays attention to similar features that she would consider in analyzing her current patient. However, these tools can be brittle, hide biases, and do not provide useful diagnostic information about safety of the models. Conversely, formal methods can be used to mathematically prove desired reliability properties of the models, eliminating human biases and statistical errors from the process.We extend these methods to provide minimal-distance counterfactuals that find minimal changes to attribute values needed to cause the model to change its prediction. This analysis can expose limitations of models and data used to train them, enabling development of provably robust AI-driven decision support systems.

Methodology

Data and Models. We use the publicly available Breast Cancer Wisconsin (Diagnostic) dataset (Dua and Graff 2017).It contains 30 numeric features which we standardized by removing the mean and scaling to unit variance. We explain Scikit-learn (Pedregosa et al. 2011) random forests, trained with a 50% train/test split. Our model consists of ten decision trees of the maximum depth of ten.

Experiments. We formally test trained tree ensembles with a SAT formalism. To find minimal distance counterfactual explanations, we start with a query and a local neighborhood in which to search for another data point to which the model assigns a different predicted label. If such a point exists, a satisfying assignment detailing the model state for the two points is returned, otherwise, we increase the size of the local neighborhood and search again. This process continues until a counterfactual is found. The relevant differences between the two points which form the counterfactual can be revealed by finding all differences within the satisfying assignments. We only report encoded threshold values which must be crossed in order for the model to change its prediction, which may be interpreted as, 'if we change select attribute values in a query by a small amount, the model will change its prediction'.

We studied the stochasticity of feature importance in the LIME (Ribeiro, Singh, and Guestrin 2016) and SHAP (Lundberg and Lee 2017) frameworks. Using those tools, we returned the n most important features where $n = 1...30$. We iterated over all test data and recorded the n most important features. We then evaluated the probability that each feature was one of the most influential variables in the test dataset when n was set. Since LIME and SHAP displayed similar behaviors, we show feature importance attribution plots for just LIME where three features that were consistently picked as important are displayed in the bottom row of Fig. 2. The horizontal axis in each of these plots shows the index of the feature in the importance ranking and the vertical axis shows the estimated probability of the feature attaining such rank. Generally, truly important features would have this probability raise quickly as a function of the rank index, and stay high. E.g., in the example shown in Fig. 2, feature *mean concave points* appears slightly more important than *worst concave points* and *worst concavity*.

We explored explanations generated by SAT. We looked

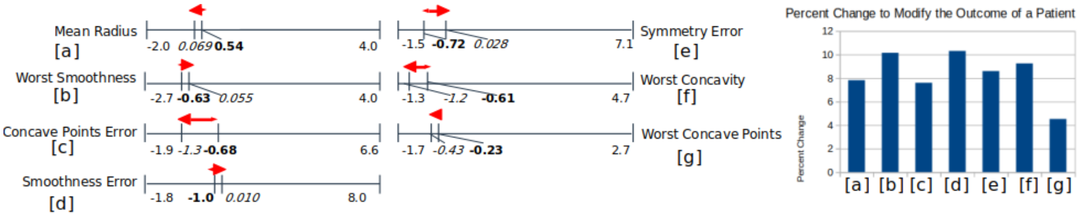

Figure 1: SAT Explanation - Counterfactual. Our explanation presents a set of the smallest changes necessary to change the model output. The bar graph shows the percent change that is required along each attribute to change the output of the model.

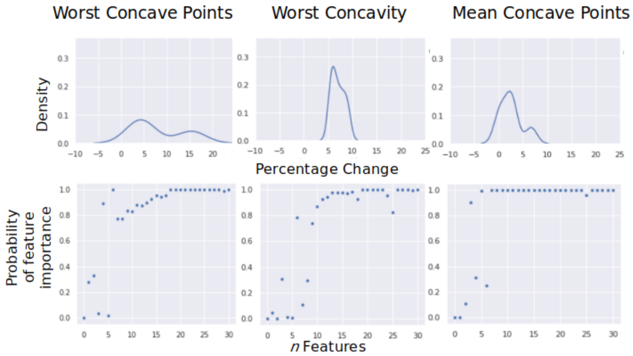

Figure 2: Feature attribution for LIME [bottom] and aggregate summary of SAT counterfactual explanations [top]

at the 17 points in the test data that were misclassified. Each explanation had a set of features that needed to be changed to flip the prediction. We iterated over all the explanations and recorded the percentage change over the range of values of each feature that would be required to modify the prediction, and visualized their distributions in kernel density estimation plots. The top row of Fig. 2 shows these characteristics for the same three features as above. Model-centric explanation suggests that fixing these 17 errors will require increasing the value of *worst concavity* by 5-10%. The other two features do not show such a consistent recipe for error correction, even though LIME suggests they are important.

Analysis

Fig. 1 shows an example model-centric explanation generated by our framework for one query. The current feature value for the suspected breast cancer mass is denoted in **bold** while the value it would need to be modified to is shown in *italics*. The left part of the graph shows the seven attribute changes required by the model to change its prediction, and the right part shows the percentage change of value of each feature needed to cause such change. This type of explanation brings our attention to features that can compromise robustness of the model: the lower the relative value change needed to affect the output, the narrower the margin for measurement error in the model.

This result has multiple potentially useful consequences. One example is confirmatory analysis of a prediction made by an AI-driven tool used by a clinician to help her diagnose a patient. If only a small change of one of the features, reflecting, e.g., some laboratory test result, can flip the prediction, the doctor may consider repeating the test to ascertain its outcome, or order a more precise test. Similarly, when designing AI-based decision systems that operate on measurements collected with noisy sensors with known sensor noise models, the design engineer may verify that the magnitude of sensor noise does not exceed the range of robustness revealed for the corresponding feature of the AI model.

To accomplish the second task type, we can leverage SAT score characteristics analogical to those shown in the top row of Fig. 2, but in this case we would obtain them using correctly classified test data. Features *mean concave points* and *worst concave points* are highly important according to LIME, but their SAT score distribution characteristics show large probability masses in the range of very small percentage changes needed to invert the model prediction. The physician and the engineer from our examples should be very careful and, if possible, measure values of these features with very high accuracy. On the other hand, feature *worst concavity* leaves our engineer with some margin of error, because non-trivial changes of its value are required to impact the model prediction.

Model-centric formal methods provide useful capabilities complementary to the existing explanatory analysis tools. They are based on mathematical logic and yield provable results that can be verified exactly, as opposite to the prevalent statistical methods that produce results with margins of confidence. We envision beneficial use of these methods at all stages of life of AI systems: from design to field application.

Acknowledgements

This work was supported by a STRI grant from NASA's Space Technology Research Grants Program; DARPA [award FA8750-17-2-0130]; and the U.S. Department of Homeland Security Countering Weapons of Mass Destruction Office [award 18-DN-ARI-00031].

References

Dua, D.; and Graff, C. 2017. UCI Machine Learning Repository [http://archive.ics.uci.edu/ml].

Lundberg, S. M.; and Lee, S.-I. 2017. A Unified Approach to Interpreting Model Predictions. 4765–4774. Curran Associates, Inc.

Pedregosa, F.; et al. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12: 2825–2830.

Ribeiro, M. T.; Singh, S.; and Guestrin, C. 2016. "Why Should I Trust You?": Explaining the Predictions of Any Classifier. 1135–1144. Association for Computing Machinery.