Received 26 April 2016; | Revised 6 June 2016; | Accepted 6 June 2016

DOI: xxx/xxxx

ARTICLE TYPE

Visual Diagnostics of a Model Explainer – Tools for the Assessment of LIME Explanations

Katherine Goode*[1] | Heike Hofmann[1,2]

[1]Department of Statistics, Iowa State University, Iowa, United States
[2]Center for Statistics and Applications in Forensic Evidence (CSAFE), Iowa State University, Iowa, United States

Correspondence
*Corresponding author. Email: kgoode@iastate.edu

Present Address
This is sample for present address text this is sample for present address text

Summary

This is sample abstract text.

KEYWORDS:
LIME, black box models, interpretability, diagnostics

Generally: avoid 'can be'. Replace by 'is'.

1 | INTRODUCTION

In the field of statistics, there are two main uses for models: interpretation and prediction. Machine learning models are often used for the latter purpose. While these models have been proven to provide accurate predictions in a wide range of problems, this predictive ability comes at the cost of interpretability. These models are typically complex and lack a functional form, so the ability to directly interpret the models is lost. This has earned machine learning models the reputation as being black boxes. The ability to interpret a model helps to diagnose the model and understand the predictions. In some areas of application, the need to be able to explain the results from a model is critical. (These last few sentences could use some work so that the ideas are better connected from one sentence to the next.) As a result, an area of research on how to explain the output from machine learning algorithms has emerged.

The forensics sciences and the heath care industry are two areas that have benefitted from machine learning but have an obvious need for explainable predictions. For example, Hare et al. [1] discuss the use of a random forest model in the forensics sciences to determine whether two bullets were fired from the same gun. Yu et al. [2] describe how healthcare has advanced through the use of machine learning in ways such as using neural networks to automate medical image diagnoses and implementing Bayesian networks to predict clinical outcomes. In both of these cases, the decisions made based on the results of the machine learning models would greatly impact the lives of human beings. If it is not possible to explain the output from a model, it becomes questionable whether to trust the results from the model to make decisions.

As an example, suppose that forensics examiners are trying to determine if a bullet used in a crime was fired from the gun that belongs to a suspect in the investigaion. A random forest model could be used to produce a probability that the bullet was fired from the gun under question. Even if the model returns a high probability of a match, it will be difficult for a jury to trust the results from the model when deciding whether to convict the suspect without the forensics examiners having the ability to explain which factors in the model led to the high probability.

The European Union has taken the desire to provide explanations from machine learning models a step further. In May 2018, the General Data Protection Regulation (GDPR) went into effect (https://eugdpr.org/the-regulation/gdpr-faqs/). The regulation includes a policy that gives the right for subjects affected by decisions made via automated algorithms to understand the logic behind the decisions being made. As Goodman and Flaxman (2016) point out, this regulation will have a great effect on the way that machine learning algorithms are used to make decisions. At minimum, it will require that an explanation of how the input variables relate to the predictions.

One approach to remedy the problem of uninterpretable models is to develop a model explainer that provides insight into the performance of the complex model. A model explainer is a method that is separate from the model but uses the model and output from the model to provide understanding into mechanism of how the model produces predictions. Recently, a handful of model explainers have been developed (reference). LIME (local interpretable model explanations) is one such model explainer (reference).

While some model explainers are focused on understanding a model at the global level, LIME was designed to explain the model at the local level for a single prediction of interest. Conceptually, LIME fits a simple interpretable model, referred to as the explainer, that is meant to capture the behavior of the complex ”black box model” in a local region around a prediction of interest. (a good place to include an image to help with this explanation)

The explanation that is produced from the explainer is also a prediction or an element of a model. Since it is not directly extracted from the complex model, it is important to ask the question, “How reliable is the explanation?”. Currently, there are no recommendations for how to assess the explanations from LIME (need to check to make sure this is a true statement). I'd like to expand on this paragraph since this is really the focus of the paper. Then the last paragraph in the intro can outline the paper.

In this paper, we will present some visualizations tools to assess the explanations from LIME in classification problems. Additionally, we will show an example using these tools on explanations from a forensics bullet matching dataset, and we will discuss how the results from the example provide some possible insights into the workings of LIME. still to come: outline of the remainder of the paper.

2 | BACKGROUND

Predictive models are used in both regression and classification settings. For this paper, we will focus on the classification setting, and in particular, we will only discuss the case with a dichotomous response variable. Accepted best practice for all predictive models is that during the model fitting process, the data is divided into training and testing portions. The training data is used to fit the complex model, and the testing data is used to assess the model. This is done to prevent overfitting and to get a more accurate assessment of the prediction error of the model. The complex model is then applied to the features in the testing data, and the resulting predictions are compared to ground truth. Of particular interest during this assessment are the cases when the model is wrong. In the dichotomous response classification case, there are two ways in which the model can be wrong. The model can make a type I error in which..., or the model can make a type II error in which...

2.1 | Overview of LIME

It does this by using the features from the training data to simulate a new dataset on which the simple model is fit. The complex model is applied to the simulated dataset to obtain predictions. The observations associated with predictions are used as the response variable in a ridge regression model with the the simulated features as the predictor variable with the highest weight given to observations closest to the prediction of interest. Feature selection is performed to identify the most important variables in the local region. A final ridge regression model is fit with the selected features, and the coefficients of the model are used to interpret the behavior of the complex model.

2.2 | LIME Algorithm

The LIME algorithm is presented in the original paper in a general context that includes cases for text classification and feature recognition. For this paper, we are only considering a situation with a binary categorical response variable and continuous feature variables. The procedure below is defined in this context. Furthermore, the implementation of LIME used in this paper is via the lime R package, and the procedure describes the methods used by the R package. This deviates a bit from the original procedure described by Riberio, Singh, and Guestrin (2016). We attempted to highlight these deviations.

Let $\mathbf{X}$ be an n by p data matrix with p features and n observations.

- $x = \left(x_1 \; x_2 \; \ldots \; x_p \right) \in \mathbb{R}^p$: original representation of an instance being explained (e.g. observations from a set of p continuous features for a specific case in the data)
- $x' \in \mathbb{R}^{p'}$: vector for the interpretable representation of the instance being explained (e.g. a vector of indicator variables associated with the features chosen through feature selection indicating whether or not the observed value is in the bin created by LIME for the feature)
- G: class of potentially interpretable models (e.g. linear models, decision trees, rule lists)
- g: explanation model where $g : \{0,1\}^{p'} \to \mathbb{R}$ and $g \in G$
- $\Omega(g)$: measure of complexity of g (e.g. depth of a tree, number of non-zero coefficients in a linear model fit using LASSO)
- f: model that is being explained where $f : \mathbb{R}^p \to \mathbb{R}$ (In classification $f(x)$ is the probability that x belongs to a certain class)
- $\pi_x(z)$: proximity measure between an instance z to x which defines a locality around x
- $\mathcal{L}(f, g, \Pi_x)$: the fidelity functions which is a measure of how unfaithful g is in approximating f in the locality defined by π_x
- $\xi(x)$: explanation produced by LIME where

$$\xi(x) = \underset{g \in G}{\arg\min}\; \mathcal{L}(f, g, \Pi_x) + \Omega(g)$$

(i.e. want to minimize $\mathcal{L}(f, g, \Pi_x)$ and keep $\Omega(g)$ low enough to be interpretable by humans)

2.3 | Implemention of LIME in R

(section describing the procedure that is currently used by the LIME R package)

The LIME R package allows for the following four methods to sample the perturbations based on the distributions of the features from the training data.

- Equally Spaced Bins
- Quantile Bins
- Normal Approximation
- Kernel Density Approximation

The methods of equally spaced bins and quantile bins also allow the user to specify the number of bins. As of now, there are no recommendations or procedures provided for how to determine which method to use. By default, LIME uses four quantile bins. It was of interest to see how the explanations from LIME varied across the four sampling methods when applied to the bullet matching data. The LIME algorithm was applied to each prediction from the test data obtained from the 'rtrees' random forest model for each of the four sampling methods. Within the bin based sampling methods, the algorithm was applied for 2 to 6 bins. It was decided to only go up to 6 bins since the more bins used the more complex the explanation becomes.

3 | METHODS

3.1 | Diagnostic Tool 1

3.2 | Diagnostic Tool 2

3.3 | etc...

4 | APPLICATION

1. explainer model generally has very low R^2 (probably due to binning) 2. "local" explanations are not local but are driven by the ("global") marginal distributions of covariates

In order to assess the LIME explanations created using different sampling methods, it was of interest to compare the top three features chosen as the important predictors by lime within a case from the test data across the different sampling methods. Figure ... is a heatamp showing the top feature chosen by lime for each of the cases in the test data and different bin based sampling methods. The rows represent the cases in the test data, and the columns represent the sampling methods. There are twenty methods included in the plot. These include the equally spaced bins and quantile bins from the lime package and the random forest score tree based bins and the same source tree based bins proposed in this paper. The rows are faceted by the test set and whether or not the observation is a match or not. The columns are faceted by these methods, and the columns within a facet represent the number of bins. Each method has 2 to 6 bins. The colors represent the top feature chosen by lime.

The variables of ccf and cms immediately show up as common variables chosen across all of the sampling methods. However, the patterns across the number of bins withing the sampling methods are different. When equally spaced bins are used, the top feature chosen is consistent across all cases within a number of bins category. For example, ccf is almost always chosen (change to actual number) with 2 equally spaced bins, matches is almost always chosen with 3 equally spaced bins, and non_cms is always chosen for the nonmatches with 5 and 6 equally spaced bins. This shows that with the bullet matching data, the top feature chosen with equally spaced bins is an artifact of the number of bins used. With equally spaced bins, this figure suggest that LIME is providing global explanations as opposed to local explanations. It would be preferable that the top feature chosen was more consistent across the number of bins and more variable across the cases. This would suggest that the top feature chosen is dependent on the feature values associated with a particular case and not just on which feature is the best explainer when b number of bins are used.

ACKNOWLEDGMENTS

This is acknowledgment text. Provide text here. This is acknowledgment text. Provide text here. This is acknowledgment text. Provide text here. This is acknowledgment text. Provide text here. This is acknowledgment text. Provide text here. This is acknowledgment text. Provide text here. This is acknowledgment text. Provide text here. This is acknowledgment text. Provide text here. This is acknowledgment text. Provide text here.

Author contributions

This is an author contribution text. This is an author contribution text. This is an author contribution text. This is an author contribution text. This is an author contribution text.

Financial disclosure

None reported.

Conflict of interest

The authors declare no potential conflict of interests.

SUPPORTING INFORMATION

The following supporting information is available as part of the online article:

Figure S1. 500 hPa geopotential anomalies for GC2C calculated against the ERA Interim reanalysis. The period is 1989–2008.

Figure S2. The SST anomalies for GC2C calculated against the observations (OIsst).

How to cite this article: Goode K., H. Hofmann, (2019), title, *journal*, *volume, number and page.*

APPENDIX

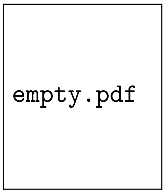
A SECTION TITLE OF FIRST APPENDIX

References

[1] Hare, E., H. Hofmann, and A. Carriquiry, 2016: Automatic Matching of Bullet Lands. *Annals of Applied Statistics*, doi:http://adsabs.harvard.edu/abs/2016arXiv160105788H.

[2] Yu, K.-H., A. L. Beam, and I. S. Kohane, 2018: Artificial intelligence in healthcare. *Nature Biomedical Engineering*, **2**, 719, doi:10.1038/s41551-018-0305-z.

AUTHOR BIOGRAPHY



Author Name. This is sample author biography text this is sample author biography text this is sample author biography text this is sample author biography text this is sample author biography text this is sample author biography text this is sample author biography text this is sample author biography text this is sample author biography text this is sample author biography text this is sample author biography text this is sample author biography text this is sample author biography text this is sample author biography text this is sample author biography text this is sample author biography text this is sample author biography text this is sample author biography text this is sample author biography text this is sample author biography text.