

Applying LIME to Bullet Data Journal

Katherine Goode

December 10, 2018

December 5, 2018 to December 11, 2018

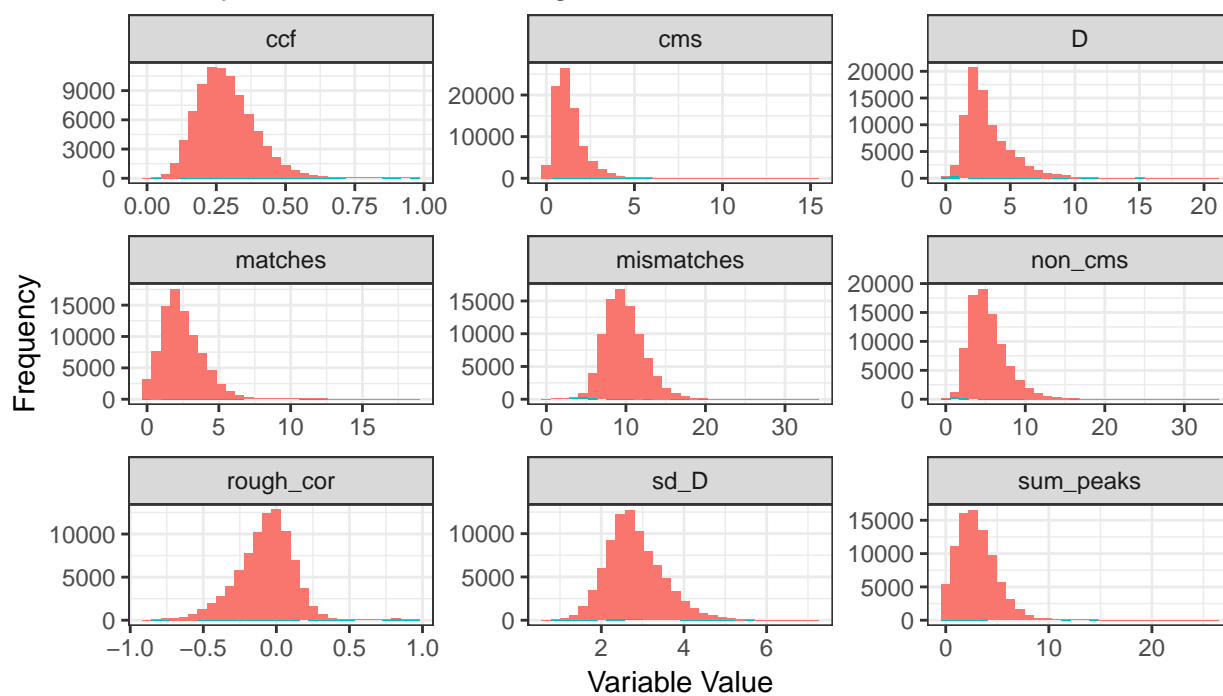
I decided to start a journal this week to keep track of the work I am doing in regards to applying LIME to the Hamby bullet data. I have been trying to decide which input values to use for LIME in the forensic examiners' paper, and I have decided that I would like to simplify matters even more by only focusing on a few input options. I have considered a handful of ways of answering this question (accuracy within one implementation of lime, accuracy across 10 implementations of lime, etc.), but I would like to reserve some of these ways for a future more technical paper. However, I would like a place to store my work to use for future purposes, and it would be nice to have a place to work where I do not feel the need to optimize code. I am hoping that this will be a place where I can write drafts of code and create first versions of graphics. Then I can clean up and organize the code when I write the paper.

Prior to this journal, I had been working on all of my ideas in the R markdown document for the paper, but now I am going to transfer a lot of that code to this journal. Many of the sections below include work that I did earlier this semester. I am just transferring them here now. Later I will move the necessary parts to the paper.

Feature Plots of Training and Testing Data Filled by `samesource`

I created the following two plots of the training data as suggested by Heike. The histograms below show the distributions of the features used in the random forest `rtrees`. The histograms are filled by the `samesource` variable, which is the truth of whether or not the comparison is from the same barrel and land. The default histograms make it hard to compare the distributions of the matches and non-matches since there are many more comparisons that have `samesource == FALSE`.

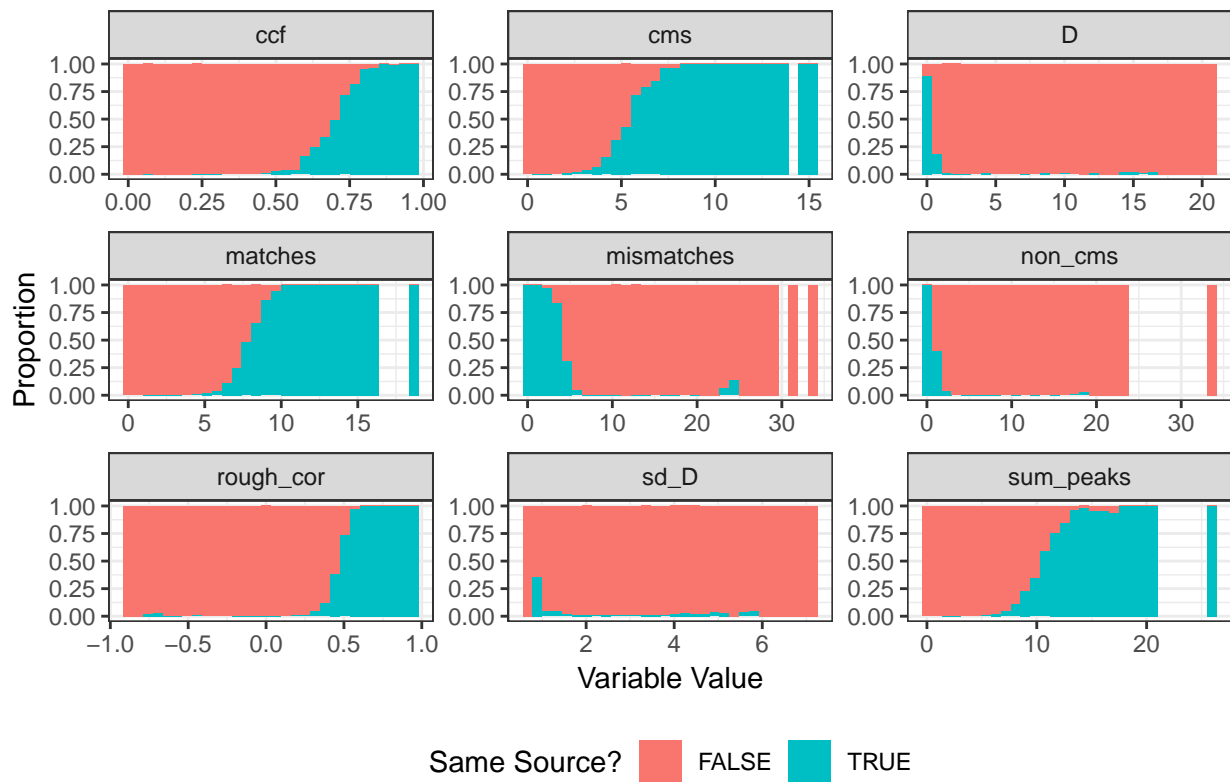
Hamby 173 and 252 Training Data



Same Source? ■ FALSE ■ TRUE

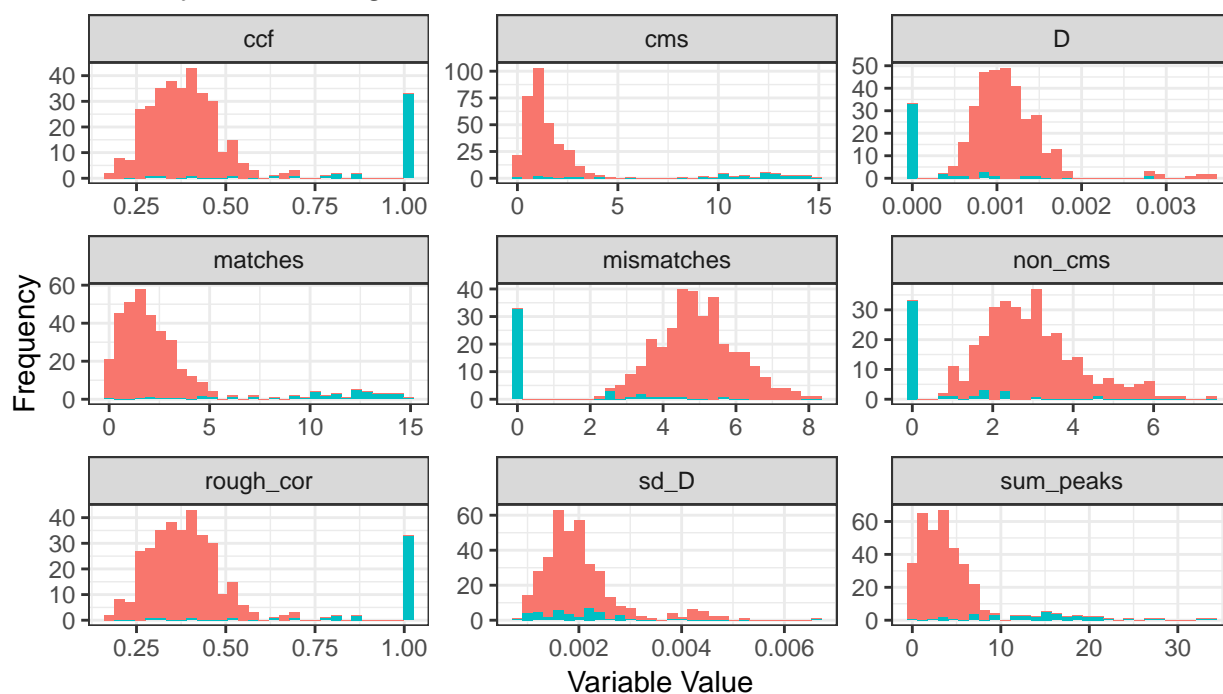
By setting `position = "fill"` in the `geom_histogram` function, it is easier to compare the matches and non-matches. These plots could be used in the future to hand select the bins for lime. Additionally, fitting a logistic regression to this data could also be used to determine the LC50, LC10, and LC90, which could be used as the bins for lime.

Hamby 173 and 252 Training Data



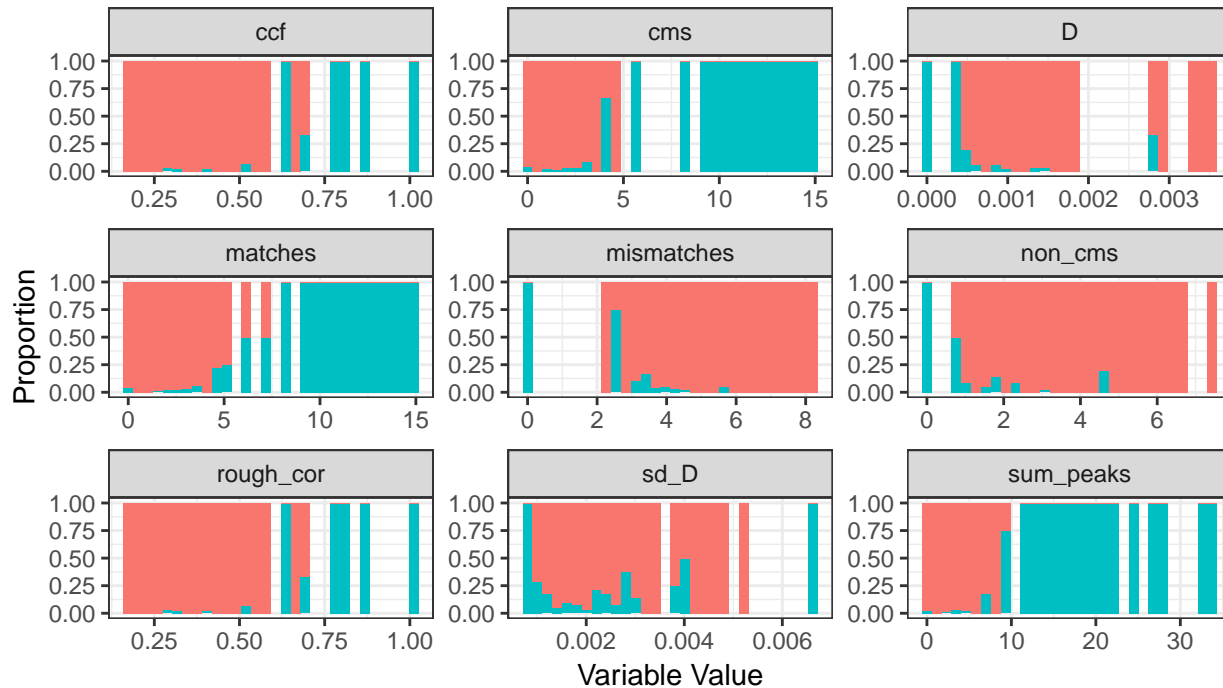
The plots below have the same structure, but they are created with the Hamby 224 testing data. I chose to not separate the testing data by sets, but this is something that could be done later if necessary.

Hamby 224 Testing Data



Same Source? FALSE TRUE

Hamby 224 Testing Data

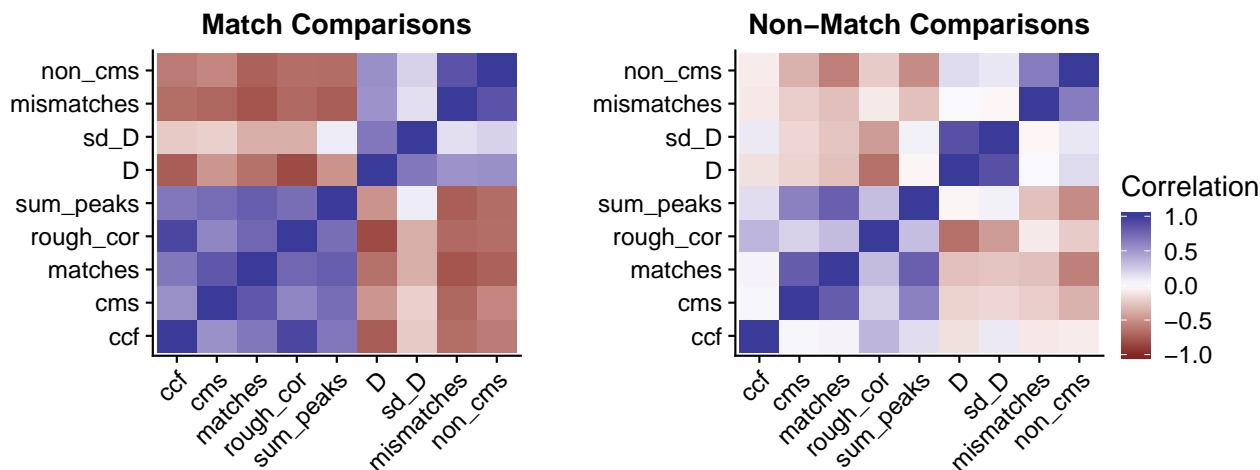


Same Source? FALSE TRUE

Correlation Plots of Training and Testing Data

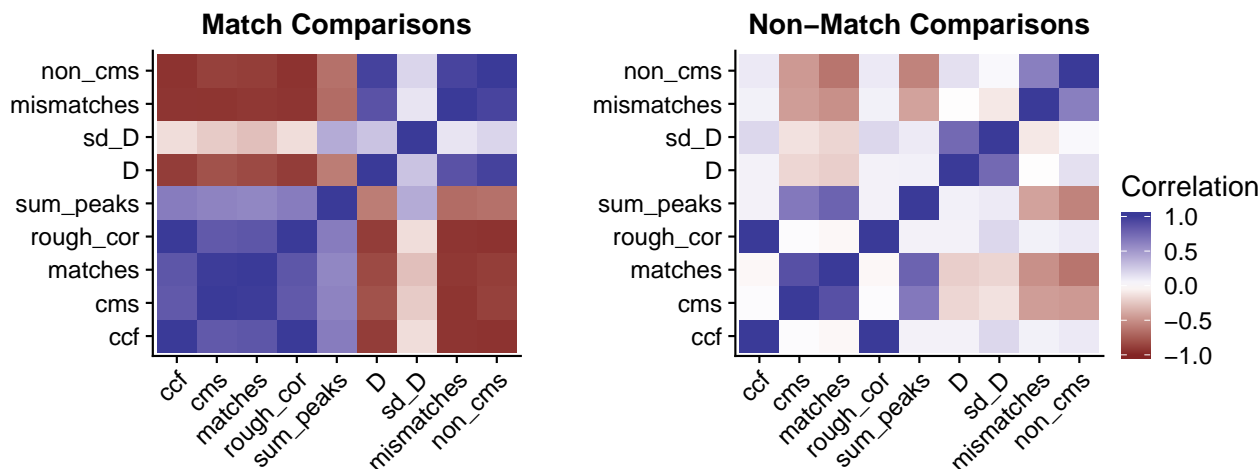
I made these plots to look at the correlation between features in the training data within the TRUE and FALSE cases of `samesource`. The features are highly correlated for the match comparisons. It is clear that the variables are more correlated with the match comparisons than the non-match comparisons. However, there are still some variables that are relatively highly correlation with the non-match comparisons.

Correlation of Feature Variables in the Training Data



The plots below show the correlations for the testing data. The patterns in the plots look really similar to ones of the training data.

Correlation of Feature Variables in the Testing Data



Draft of LIME Procedure from the `lime` R Package

I wrote this up to start thinking about how to describe the procedure that the `lime` R package uses to implement the LIME algorithm. It needs a lot of work, but it is a start. The final version of this will end up in the technical stats paper critiquing LIME. The version in the firearm examiner's paper will be much simpler.

The steps below explain the procedure that the R package is using to apply the LIME algorithm to the bullet matching predictions on the Hamby 224 clone dataset made by the random forest model from Hare. For simplicity, the steps are described as what happens to one case in the test data. Thus, the steps (2) through (7) are repeated for each observation in the testing dataset.

Let

$$Y_{jk} = \begin{cases} 1 & \text{if bullets } j \text{ and } k \text{ were fired from the same gun barrel} \\ 0 & \text{otherwise} \end{cases}$$

be the response variable in the training dataset, and X_1, \dots, X_9 correspond to the nine features in the training dataset. Let X'_1, \dots, X'_9 be the

1. Distributions for each of the features in the training data are obtained.

The method that 'lime' uses to obtain the distribution differs based on the feature type. All of the features in the Hamby datasets are numeric. For numeric features, the default option in 'lime' ('quantile_bins = TRUE') computes the quantiles of each feature based on the number of bins selected. The default number of bins is 4 ('n_bins = 4').

2. n samples from each of the feature distributions are drawn.

To do this, 'lime' has several options (mostly quoted from 'lime' package for now):

- bin_continuous = TRUE should continuous variables be binned?
- quantile_bins = TRUE should the ins for n_bins be based on quantiles or spread evenly
- n_bins = 4 number of bins if bin_continuous is TRUE
- use_density = TRUE if bin_continuous is FALSE, should continuous data be sampled using kernel density estimation (if not, then will assume normal for continuous variable)

3. Predictions for the testing data using the random forest model are computed.

The random forest model 'rtrees' is used to make a prediction for the observation from the test dataset and each of the $n = 5000$ samples as to whether or not the comparison of the two bullets in the test case are a match. Since the random forest is a classification model, 'lime' is set to return the prediction probabilities.

4. Similarity score between the observation in the testing data and each of the $n = 5000$ sampled values are obtained.

The way that the similarity score is computed depends on the type of feature. Since all of the features in the Hamby 224 test dataset are continuous, the simulated values are first converted into 0-1 features where a 1 indicates that the feature from the simulated value falls in the same bin as the observed data point and a 0 indicates that the feature is not in the same bin as the observed data point. Then, by default, the Gower distance is used to compute the similarity score. (using the 'gower' package in R)

5. Feature selection is performed by fitting some type of regression model weighted by the similarity scores is to the simulated data and the observed value. The 0-1 versions of the features are used.

The user can specify the number of features, m , they would like to select to explain the prediction. 'lime' supports the following options for feature selection

- (a) forward selection with ridge regression
- (b) highest weight with ridge regression
- (c) LASSO model
- (d) tree model
- (e) default: forward selection if $m \leq 6$ with a ridge regression model, highest weight with ridge regression otherwise

6. A ridge regression model is fit as the simple model by regressing the prediction probabilities on the m selected predictor variables and weighted by the similarity scores. If the response is categorical, the user can select how many categories and which categories they want to explain.

$$P(\textit{Match} = \textit{TRUE}) = \beta_0 + \beta_1 \cdot I[X_1 \in \text{obs bin}] + \beta_2 \cdot I[X_2 \in \text{obs bin}] + \beta_3 \cdot I[X_3 \in \text{obs bin}]$$

For the prediction of interest,

$$P(\textit{Match} = \textit{TRUE}) = \beta_0 + \beta_1 + \beta_2 + \beta_3.$$

7. The feature weights are extracted and used as the explanations.

Note: I realized that if `bin_continuous = FALSE`, then bins are not used at all. Instead, a kernel density estimator is used to sample from the distribution (or a normal distribution if specified), and then the ridge regression models are fit without “numerified” values.