Interpreting Random Forest Predictions for Firearm Identification Using LIME

Katherine Goode and Heike Hofmann

November 06, 2018

1 Introduction

(need to finish adding sources)

The discipline of firearm identification examines bullets to determine the likelihood that a bullet found in a criminal case was fired from a particular gun. To do this, the bullet from the crime will be compared with a bullet that was known to be fired from the gun under evaluation. Traditionally, this is a procedure that has been performed by hand. Specially trained examiners visually compare the microscopic bullet striations that were created when the bullets passed through the gun barrel. Often a comparison microscope is used that allows the examiners to view both bullets at the same time (National Research Council 2009). The examiners use this ability to determine whether the two bullets were fired from the same gun barrel.

Recently, the scientific community has been encouraging the inclusion of more data driven techniques to be used in forensic investigations. These methods would allow for the reporting of a measure of uncertainty in addition to the conclusion drawn from the analysis. This led Hare, Hofmann, and Carriquiry (2017) to propose a new computer automated method of bullet matching that could supplement the visual inspection by the firearm examiners. Their method involves obtaining bullet signatures of the striations from the scans of the two bullets, computing variables that measure the similarity of the two signatures, and using a trained random forest model to determine the probability of a match based on the similarity variables. They trained their model on a set of known bullet matches and non-matches from the Hamby study. They demonstrated how the random forest model can be used to make prediction on a new set of bullet signature comparisons. The results from the paper suggest that the random forest model leads to highly accurate bullet matching predictions. The authors found that when their model was applied to a testing dataset (?), the resulting error rate was 0%.

While random forest models often result in good predictions as seen in Hare, Hofmann, and Carriquiry, it is well known that a disadvantage of random forest models and other machine learning techniques is that it is difficult to interpret the models (reference?). For example, it is not possible to tell which variables played an important role in the creation of individual predictions. This issue led to the development of LIME (reference), which is an algorithm that examines the behavior of the complicated model on a local scale around a new prediction using a linear regression model. This allows for the ability to understand which were the driving variables that led to a prediction of interest.

Since firearm identification identification is commonly used as evidence for convictions in court cases, it is important to be able to understand and assess the model that is being used to quantify the probability that a bullet was fired from a gun. LIME provides the ability to understand which were the key variables used by the random forest model to make a prediction, which would allow firearm examiners to check whether or not the predictions created by the random forest are based on reasonable variables.

This paper provides an example of the application of LIME to a bullet matching problem. (provide more details about what is contained in the papers - i.e. section 2 describes the Hamby data; section 3 describes the random forest model and LIME; . . .)

2 Data

2.1 Training Data: The Hamby 173 and 252 Datasets

2.2 Testing Data: The Hamby 224 Clone

The Hamby 224 Clone is organized as a test set of a cloned (sub-)set of the Hamby 224 bullets. As with all Hamby sets (Hamby, Brundage, and Thorpe 2009), Hamby set 224, is a collection of 35 bullets, organized as 20 known bullets and 15 questioned bullets. The known bullets are fired in pairs of two through one of ten consecutively manufactures P-85 barrels. Clone set 224 is arranged as a test set of fifteen tests, one for each questioned bullet. Each test set is arranged as a combination of three bullets: two known bullets and a questioned bullet. The test asks for a decision on whether the questioned bullet comes from the same source as the two known bullets or from a different source. This situation is similar to what a firearms and toolmarks examiner might encounter in case work.

3 Methods

3.1 Random Forest Model

3.2 Overview of LIME

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin developed LIME in response to the interpretabiliby issue of many machine learning techniques. Many models used in machine learning are referred to as black box prediction models since they are complex models that may provide accurate predictions but are not able to be interpreted. That is, it is not possible to understand the relationship between the features and the response variable. Without being able to interpret the models, it becomes difficult to diagnose why a model is producing the predictions that it is. The developers claim that LIME allows users to check which variables are driving the predictions made by the model. This further allows users to determine whether their model is trustworthy.

LIME was designed to be a generalized algorithm that can be applied to any predictive model to return an "explanation" for an individual prediction of interest. This explanation is meant to convey which features from the model are driving the prediction created by the complex model. In the next section, I will provide a detailed description of how the LIME algorithm obtains the explanation when applied to a random forest model with numeric features. For now, general overview of the LIME algorithm will be explained.

Suppose that a complex model has been fit on a training dataset with k features and a response variable that may either be numeric or categorical. This trained model will be applied to a testing dataset containing the k features to obtain predictions. It is of interest to understand which variables played an important role in the predictions. LIME will be applied to one of the cases in the testing dataset to obtain an explanation. To create the explanation, LIME starts by approximating the distribution of each of the k features in the training dataset. It then simulates a large dataframe from these distributions. The simulated data are used to perform feature selection to determine the m most important features. Then a weighted ridge regression model is fit to standardized versions of the selected m features from the simulated data. The model assigns higher weights to observations closer to the observed case of interest. A linear model is used since it has a well known form and coefficients that can be interpreted. The coefficients from the model are extracted and used to "explain" the importance of the variables. Features with a larger absolute value of their coefficient are implied to play a larger role in the creation of the prediction.

3.3 Applying LIME

The developers of LIME wrote a Python package to implement LIME, and Thomas Lin Pedersen adapted their work to create an R package called `lime`. This paper will use version 0.4.0 of the R package `lime` in the analysis of the bullet matching data. 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 $X_1, ..., X_9$ correspond to the nine features in the training dataset. Let $X_1', ..., X_9'$ 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‘ draws a sample from a Uniform(0,1) distribution and uses the bin information to compute a sample drawn from the feature distribution.

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.

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

   For the prediction of interest,

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

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

```
##   model_intercept   feature feature_weight model_prediction rfscore
## 1       0.6882838         D   -0.000206695        0.6873396       0
## 2       0.6882838 rough_cor   -0.043971396        0.6873396       0
## 3       0.6882838       ccf    0.043233935        0.6873396       0
```

```
## [1] 0.6873396
```

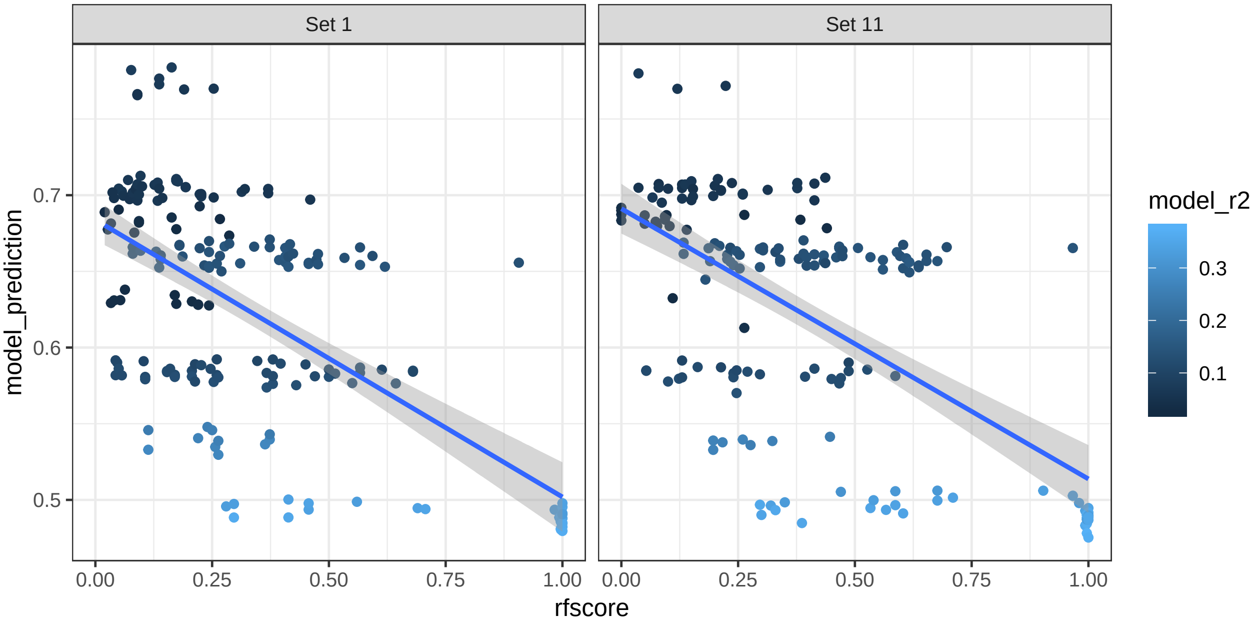
```
## [1] 0.6873396
```

```
## [1] 36.7975
```

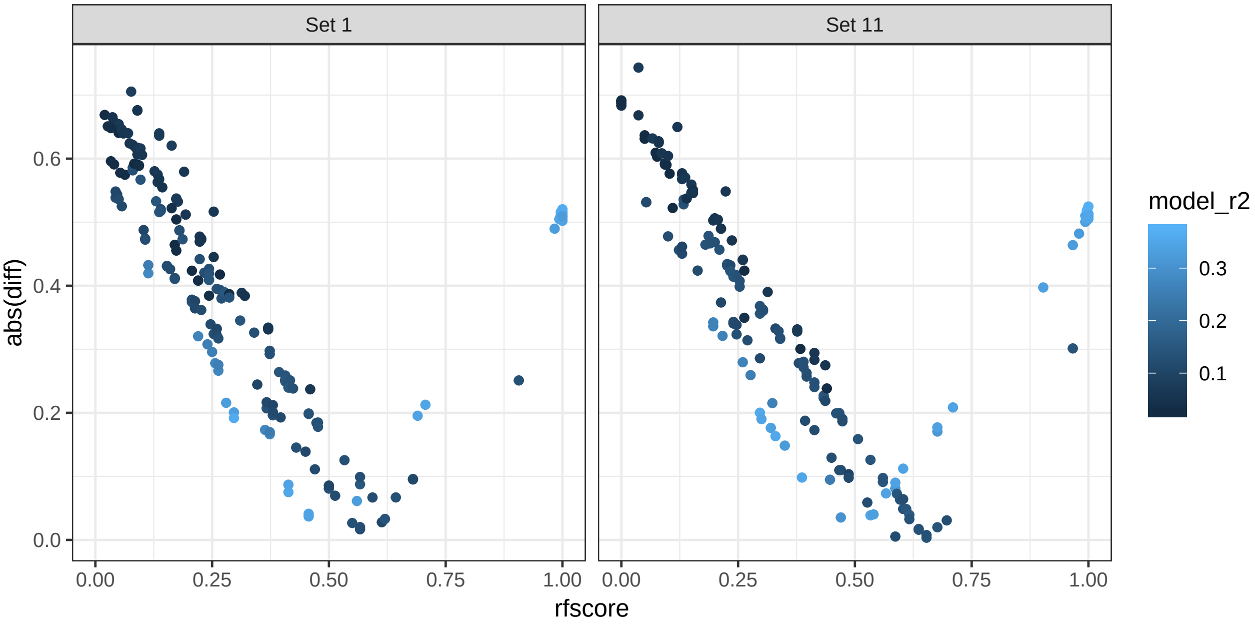
```
## [1] 27.81988
```

Below is are plots of the probability of a match computed from the "simple" (ridge regression) model in `lime` versus the probability of a match computed from the `rtrees` random forest model for both the Hamby224 Set 1 and Set 11 test cases. Linear regression lines have been fit through the points. Both plots show that there is a weak negative linear relationship between the two variables. This suggests that as the probability of a match increases in the random forest, the simple model is doing a worse job of computing a probability that agrees with the random forest model. However, the Hamby 224 test sets 1 and 11 do not have many observations with high random forest scores. Each test set should only have 18 matches. The trend may be different if there was more cases with high random forest scores. However, it appears that the cases that are a match have some of the lowest model_prediction scores overall.

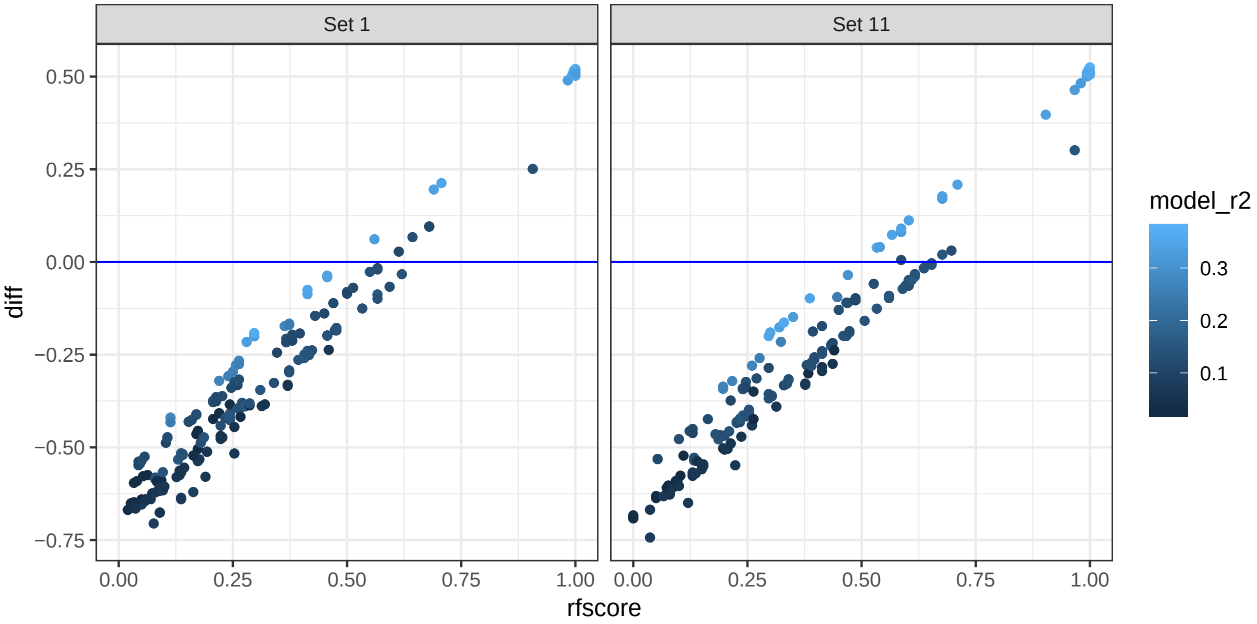
THe points are also colored by the R^2 value from the simple model. This plot suggests that the simple models with a prediction close to 0.5 have a better fit, but the R^2 values are still low. I am not sure what to make of this. However, it does not appear that the R^2 is related to the probability from the random forest.



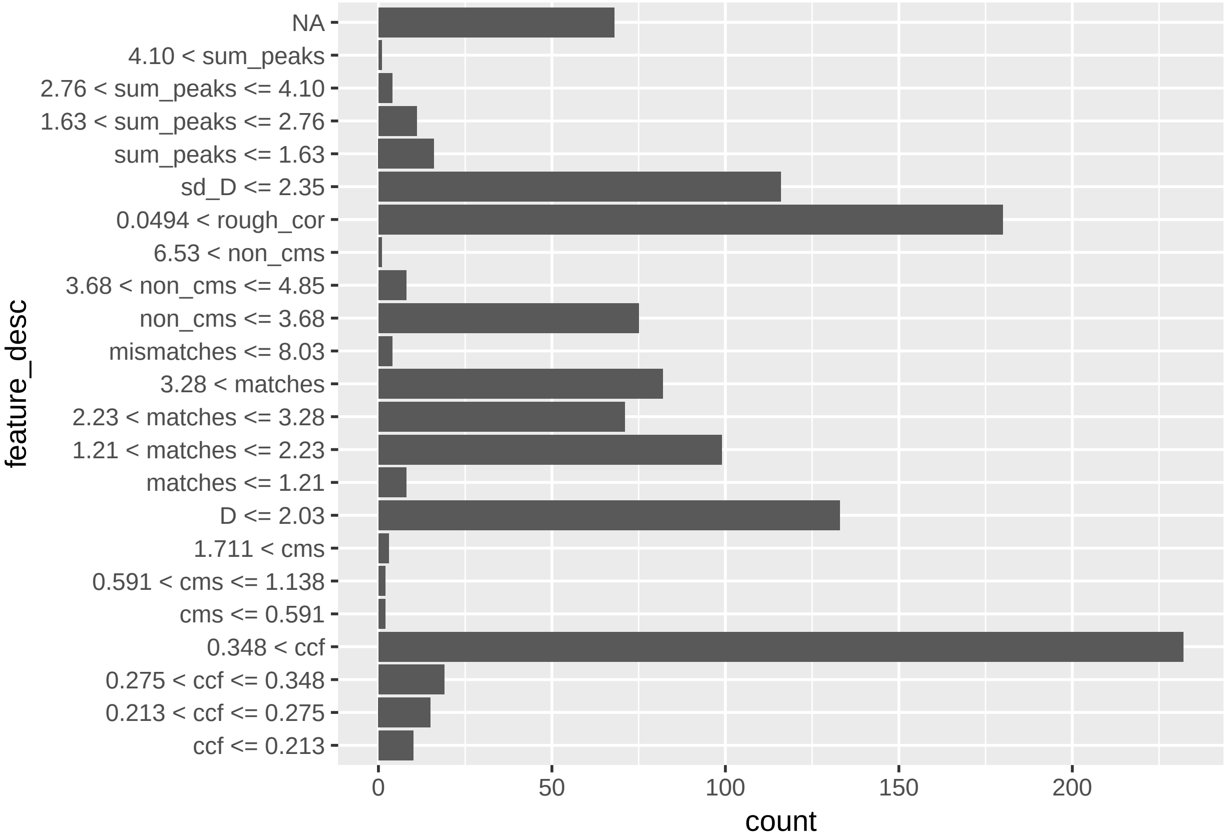
The plots below show the absolute value of the difference between the random forest probability and the simple model probabilty versus the random forest probability. These plots show a clear v-shaped trend that suggests that the simple model is doing the best job of replicating the random forest probabilty around the random forest probability of 0.5. That is, the simple model seems to be doing a worse job of computing accurate predictions in the extremes of the random forest scores.

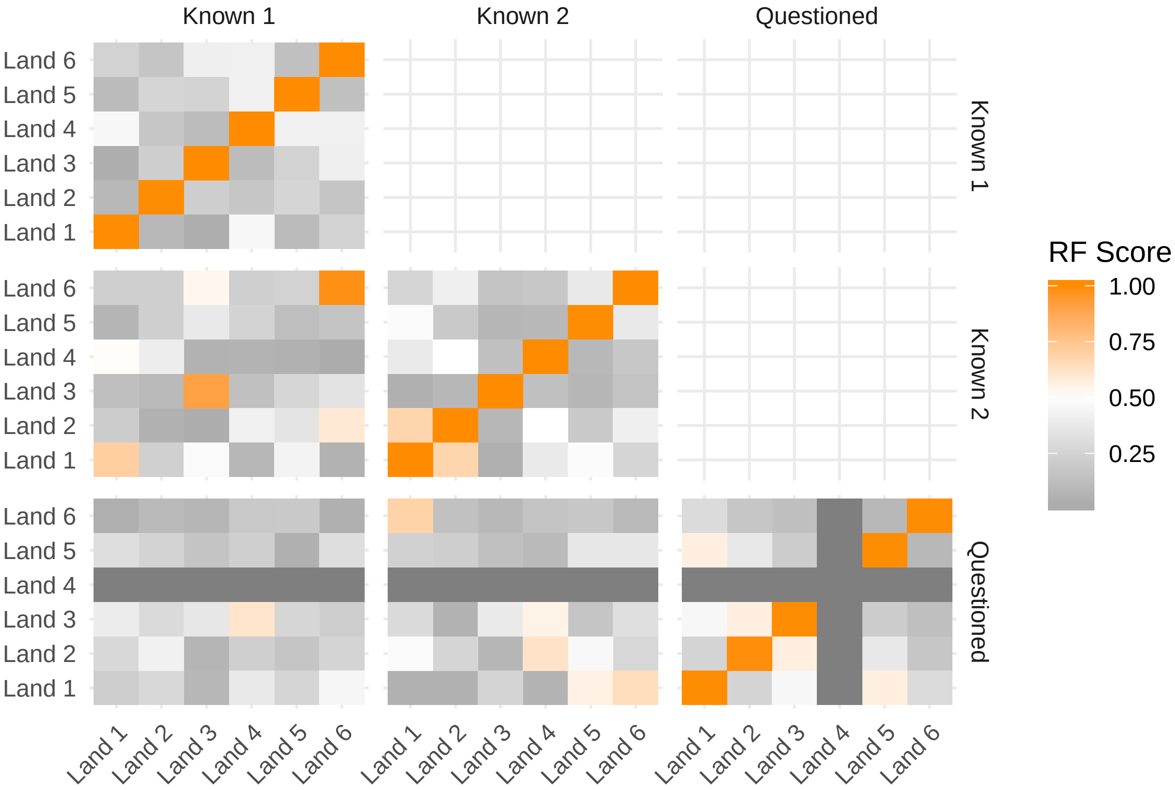


The plot below shows the difference of the two model probabilities versus the random forest score. This shows that when the random forest score is low, the simple model is over estimating the probability, and when the random forest score is high, the simple model is underestimating the probability.



3.4 Visualizing the LIME Explanations





4 Results

5 Discussion

References

Hamby, James E., David J. Brundage, and James W. Thorpe. 2009. "The Identification of Bullets Fired from 10 Consecutively Rifled 9mm Ruger Pistol Barrels: A Research Project Involving 507 Participants from 20 Countries." *AFTE Journal* 41 (2): 99–110.