**Predicting Category of Crimes in San Francisco**



**Team Members: Jinxin Ma, Juan Pablo Oberhauser, Jennifer Zhu**

**December 8, 2016**

**Executive Summary**

To understand and predict the category of crimes in San Francisco, we built models using 14-year incident data derived from [San Francisco Police Department](http://sanfranciscopolice.org/) (SFPD) crime incident reporting system. Despite the complexity of the problem, we achieved an overall accuracy of 54% in predicting the top 5 major types of crimes, with a best accuracy of 86% for the “larceny/theft” category. We also generated multiple data visualization results such as displaying our results with map APIs.

We explored multiple approaches to engineer features and select models. Based on the log loss metrics, we chose a tuned extreme gradient boosting classifier as our “best” model. And the important features we discovered included the date, day of the week and time of the crime incident, a detailed description of the crime incident, the district in which it happened, the resolution of the crime, the address and the X and Y coordinates.

**Introduction**

Crime Category prediction is very beneficial to understand which factors drive which kinds of crime in a big city like San Francisco. With relatively simple exploratory data analysis, some very important pieces of information can be recovered from 14 years of data of crime reports. For example, knowing which crimes happen the most in a certain neighborhood and at which hours of the day could be of great use for knowing where patrols should be placed. Like any efficiency problem, knowing where to place limited resources is very important, so knowing where policing is needed or where cameras suffice for example could be very beneficial in crime prevention. Going beyond the exploratory data analysis, building a model that accurately predicts the category of a crime based on several factors linked to that crime report can help explain which specific factors are important in separating violent from nonviolent crimes for example.

**Data and Features**

***Data Acquisition***

For this project, our group utilized SFPD crime incident reporting system data to predict the category of the crime. The data ranged from 1/1/2003 to 11/18/2016, giving us a very hefty amount of information for crime classification. It consisted of 14 years of crime data that were reported to the San Francisco Police Department. It contained several pieces of information for each of the crimes reported, including the date and time of the crime incident, the category (which is the response variable), a detailed description of the crime incident, day of the week, the district in which it happened, the resolution of the crime, the address and the X and Y coordinates. We split the dataset into a training set and a test set. The training set and test set rotate every week, meaning week 1, 3, 5, 7...belong to the test set, week 2, 4 , 6, 8 belong to the training set.

***Exploratory Data Analysis***

Before jumping to model fitting, our team created a fairly comprehensive data exploration script where we tried to see how well time and location of the crime would be in separating different categories of crimes. At first glance, this problem seemed very complicated in the way that there were more than 30 categories of crimes. Another big challenge was that it was an unbalanced data set when it came to the features. For example, in the training set, the “Larceny/Theft” category occurred a total of 207,113 times (19.9% of the total incidents). The second most common category was “Other Offenses”, which was recorded a total of 126,182 times (14.4% of the total incidents), but they were heterogeneous and probably not very informative. Overall, there were 29 categories which had under 25,000 counts (less than 2.3% of the total incidents), and the percentage of crimes in the top 20 categories was 97%.

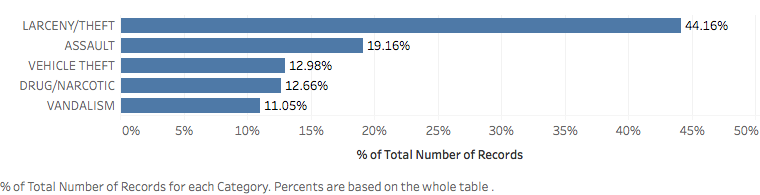
Next, we analyzed every crime category and their distributions over time of the day when the incident happened, since time of day was probably an important feature for certain types of crimes. For example, driving under the influence and reckless behavior were two crime categories where the incidents happened mostly late at night or very early in the morning, making the hour of the day an important identificator for these categories. On the other hand, bad checks for example was a category for which the incidents reported did not necessarily happen at a specific time of the day.

There was also a lot of variation in crimes over the year, suggesting that the month of the year could also be a good predictor for crime category. Most importantly though, some of the best predictors were probably the geo-temporal variables. These included minute, hour, day, month, year, latitude, longitude, and several features derived from the latitude and longitude variables (such as districts and locations in the street, i.e., in the corner *etc*).

***Data Processing***

The data cleaning part of this project proved challenging given that our team had to build a whole exploration pipeline to validate coordinates. After exploring these, we realized that many coordinates were in the North Pole. It was unclear whether these coordinates were imputed after the fact as place holders, or they are simple data-entry errors. We decided to get rid of these coordinates by building a grid around San francisco, and any coordinates that fell outside of the grid were removed.

After analyzing the original problem and testing out a few algorithms for benchmarks, we decided to simplify the problem by reducing the number of categories to top 5 categories (excluding the “other” categories). The remaining categories were Larceny/theft, Assault, Vehicle Theft, Drug/Narcotic, and Vandalism. In the original dataset, these five categories accounted for approximately 45.03% of the crimes (Figure 2). The underlying logics was that narrowing down the problem to the most prevalent crime categories would greatly improve our prediction accuracy, without having to throw away too much of the data. Furthermore, if we built a model that had good predictive power on the most frequent crime categories in San Francisco, this model could be useful right away in a real-life setting. Therefore, We decided to build several classifiers to predict these five major crime categories.

*Figure 1 Percentage of Top 5 Crime Categories*

**Methods**

***Feature Engineering***

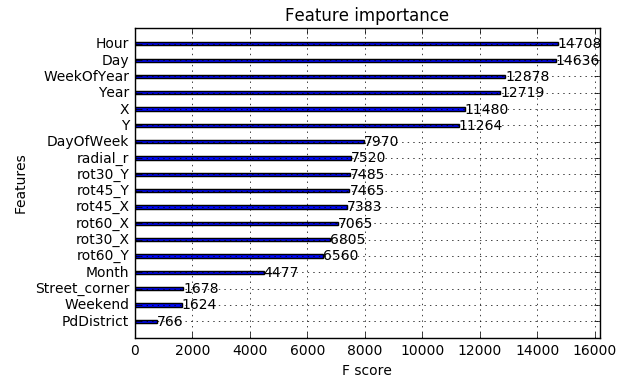
For this multi-label classification problem, we implemented several classification algorithms after making multiple features from the original data. Some of those features have been described in the sections above, but this section will include a comprehensive list of those features.

First, we created several temporal features from the “Date” column. These features included the day, month, year, and hour of when the crime happened. Additionally, we constructed the binary features of whether the crime happened during a weekend or weekday. We also included a new feature which specified the day of the week (i.e. Sunday, Monday, *etc.*) when the crime happened.

Next, we constructed several geospatial features. The first one was categorical variables of the district where the crime happened, such as “Mission” and “Richmond”. Based on the address information, we also differentiated between crimes that happened in a street corner (i.e an intersection) and crimes that happened at a specific address. Additionally, we included the latitude and longitude coordinates, as well as transformations of the coordinates (such as the transformations into polar and radial coordinates). These transformations provided more geospatial information than using latitude and longitude alone.

In summary, the feature importance was shown in Figure 1 (based on our final model described in the next section).

*Figure 2 Measurement of Features Importance by F-score*



***Model Fitting and Selection***

We built several classifiers including (but not limited to) random forest classifier, extremely randomized trees, linear discriminant analysis, logistic regression, gradient boosted trees, and extreme gradient boosting. Both the best overall accuracy and within class accuracy were achieved by using extreme gradient boosting. The hyper-parameter tuning for this model was achieved through a grid-search of several parameters.

Extreme gradient boosting is a tweak of gradient boosting algorithm. The way gradient boosting works is by building several classification decision trees, but using the results of the previous trees and its mistakes to correct the next decision tree. Gradient boosting allows us to combine many weaker classification decision trees and to get a majority vote from all of those. This improves performance significantly, and we decided to use this algorithm by comparing performance based on a 80-20 train/test split with 5-fold cross validation. The performance metrics we observed were overall accuracy, precision, F-1 scores, and those same metrics for each individual class we were trying to predict. The results were quite good and will presented in next section.

**Modeling Results and Prediction**

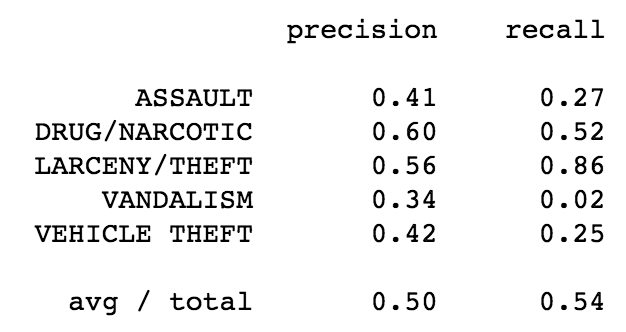
We used logarithmic loss (log loss) to assess the performance of our models. Log loss is a classification loss function which is often used in Kaggle. It is defined as:



where N is the number of samples, M is the number of possible classes, yij is a binary indicator of whether label j is the correct one for sample i, pij is the probability of assigning label j to sample i. A good classifier usually has a log loss close to zero.

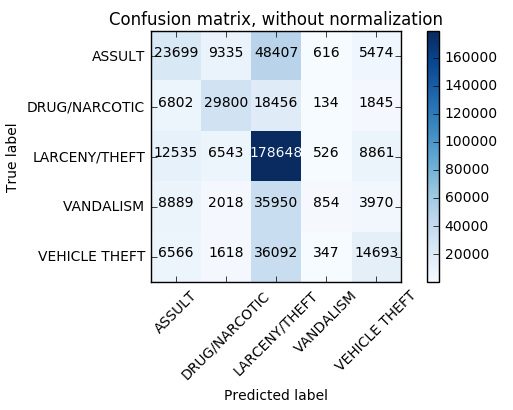
Using extreme gradient boosting, we were able to achieve a log loss of 1.17, indicating that our model had good performance. The overall results for our final model are presented in Table 1.

*Table 1 Overall Results of the Extreme Gradient Boosting Model*



In the table, the overall recall of our model indicated an overall or average accuracy score of 54%. The surprising results came when we realized how good of a job our model was doing in predicting the “larceny/theft” category. Our overall results were negatively impacted by categories where our model was really failing to predict with any accuracy. For example, the accuracy for vandalism is 2%. This was where some domain knowledge could have come in handy. The overall confusion matrix was presented in Figure 3.

*Figure 3 Confusion Matrix of the Extreme Gradient Boosting Model*



**Conclusion**

We used 14 years of data of crime that were reported in San Francisco to predict the category of crime. We started by applying some data visualizations with all of the geographic data contained. After understanding the data, our idea was to build a multi-class classification tree and ensemble of trees to benchmark the first results and improve the prediction accuracy by engineering new features and implementing new classification methods.

The problem was challenging in that we were only given limited features such as the time and geospatial information. We extracted from the available data several new features including hour, day of week, day of year, solar coordinates that could potentially benefit the modeling process. Using machine learning techniques, such as extreme gradient boosting, random forest, and extremely randomized trees, we achieved an overall accuracy of 54% for predicting the top 5 categories of crime. Particularly, we obtained an forecast accuracy rate of 86% for Larceny/Theft.

In the future, we would like to incorporate external data and continue engineering new features. We are confident that after further refinement, our model can be put into production and help predict the category of crime.