International Institute of Information Technology, Bangalore

GEN-511 Machine Learning Project

San Francisco Crime Classification

Jagmeet Singh *MT2018042*
Karanbir Singh *MT2018047*
Sonveer Tomar *MT2018118*

December 7, 2018

Contents

List of Figures

List of Tables

1 Introduction

San Francisco is famous for various types of crimes. San Francisco Police Department recorded all these crimes based on date, time, place, Description of type of crime, category of crime, Resolution of crime etc. The date is of range starting 1/1/2003 to 5/13/2015.Data is made public by San Francisco Police Department on Kaggle. This project is aimed to analyze crimes in San Francisco that took place from 2003 to 2015. Using data modeling approaches such as Logistic Regression, Random Forest, Naive Bayes we created models that can be used to classify category of crime given the location and time.

2 Dataset Description

The San Francisco Crime Classification dataset contains the following set of features:

2.1 Features

Every entry in our training dataset is about a particular crime, and contains the following information:

- **Dates** The date and timestamp of the crime in the *YYYY-mm-dd hh:MM:ss* format. So Year, Month, Day, Hour, Minute and Second can be extracted from this column.
- **Category** The type of the crime. This is the target/label value which is predicted by the model.
- **Descript** Description of crime. So this feature should not used for prediction.
- **DayOfWeek** Day of week on which crime occured.
- **PdDistrict** District of police station where crime was reported.
- **Resolution** Action taken against a crime to resolve it.
- **Address** Approximate street address where crime occured.
- **Longitude X** longitude of the location in San Francisco city map where crime occurred.
- **Latitude Y** latitude of the location in San Francisco city where crime occurred.
- **Id** unique id to identify each crime record in the dataset.

2.2 Data Distribution

i **Hour** This feature is taken from Dates label. It provides correlation between number of particular crimes occured in hours[0-23]. Figure 1 explain the number of LARCENY/THEFT during 18th hour is nearly 14000.

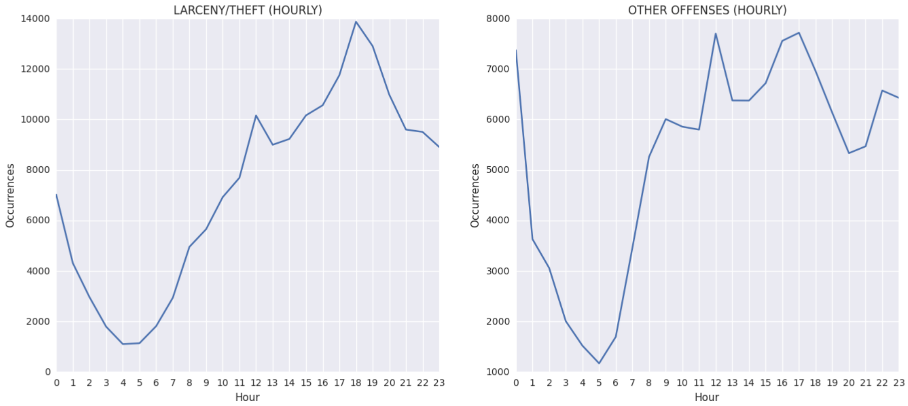

Figure 1: Correlation between each hour and number of particular crimes occurrences.

ii **District** District label gives the details of top 5 crimes occurred based on numbers in a particular district. Figure 2 represent number of theft crime in BAYVIEW district is 10000. and Figure 3 represents total number crimes occurred in a particular district.

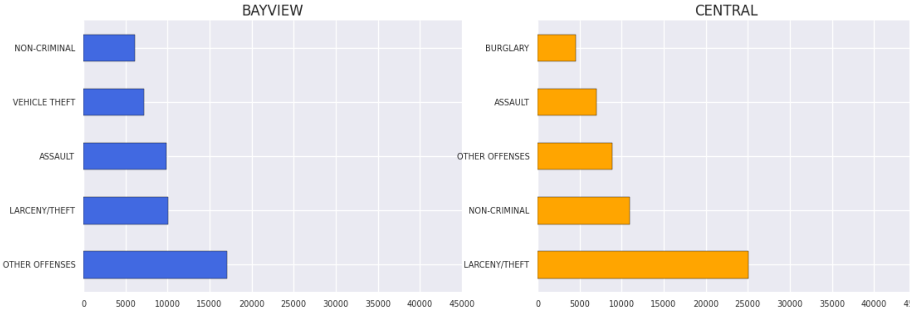

Figure 2: Correlation between districts and top 5 crimes based on number.

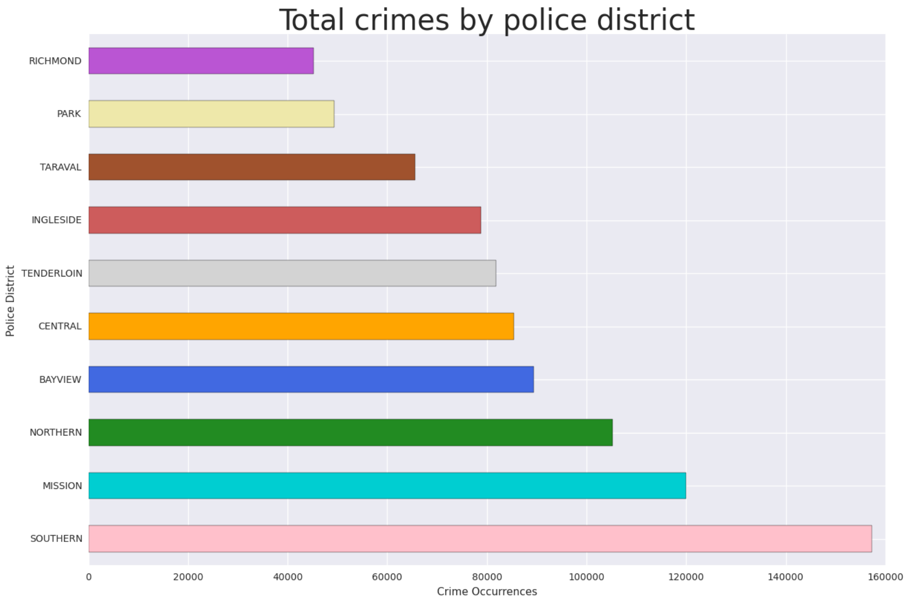

Figure 3: Total number crimes occurred in each district

iii **Yearly** It shows the relation between number of particular crime occured in a particular year. It contains number of crime starting from 2003 to 2015. Figure 4 represent number of LARCENY/THEFT in 2014 is 18400 and 7750 in 2015. Figure 5 represents correlation between total number of crimes occured in each year, month and hour of day.

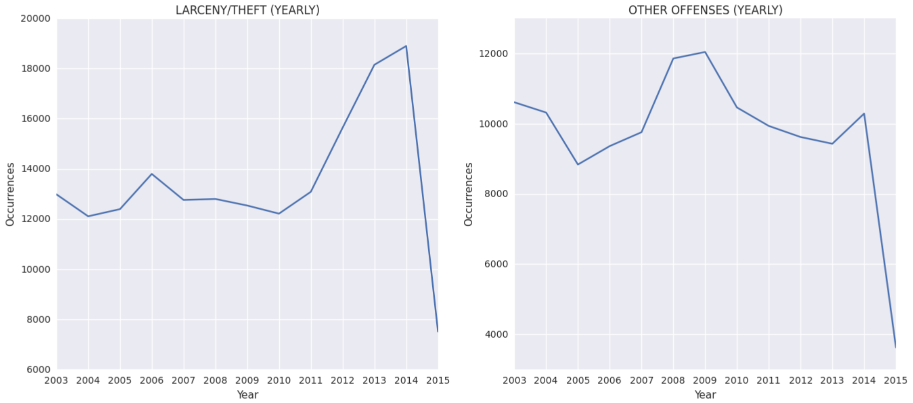

Figure 4: Correlation between year and number of crime of particular type.

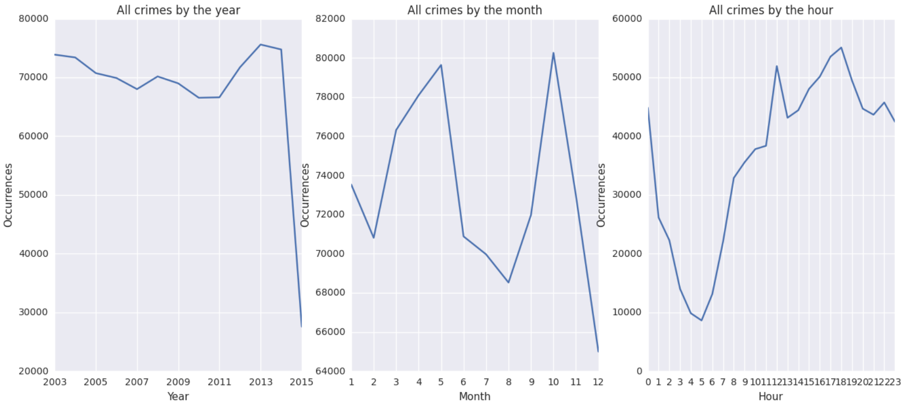

Figure 5: Correlation between total number of crimes occurred yearly, monthly and hourly.

iv **Category** This is the label we want to predict. It gives number of occurrences of particular crime. Figure 6 gives the absolute count of particular crime in the dataset. Total number of other offenses are 12414. Figure 7 shows the absolute count and cumulative percentage of all crimes.

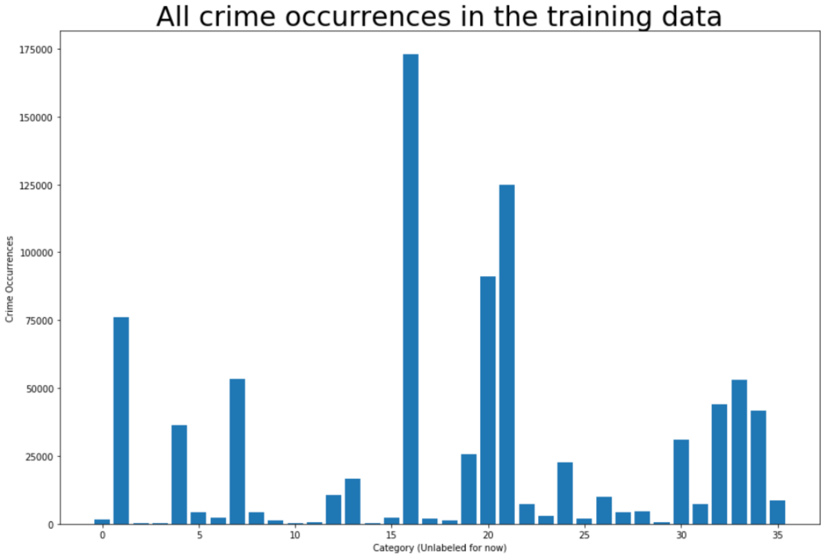

Figure 6: Number of occurrences of particular crime.

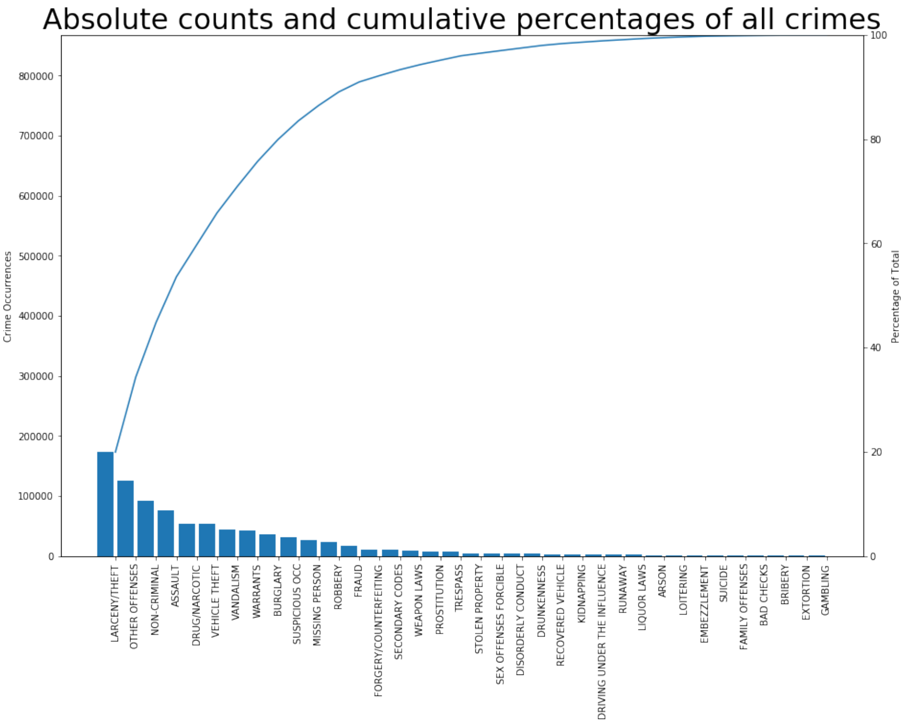

Figure 7: Absolute count and cumulative percentage of all crimes.

v **Day of week** Day of week gives us details about which crime occurred in a particular day of week.

vi **Address** Train.csv contains about 87000 number of record which gives details about which crime occurred on a given address.

3 Evalution Metrics

This metric is used to evaluate the quality of the classifier. The multiclass logarithmic loss is used to compare the performance of model.

$$logloss = -\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} y_{ij} log(p_{ij}) \tag{1}$$

4 Feature Engineering

4.1 Data Preprocessing

Following steps are taken for data preprocessing.

i Firstly load the training data set given in *Train.csv* file and see the various statistical properties like number of columns, mean etc.

ii Drop the null values in the dataset.

iii Check whether longitude and latitude are valid or not.

iv Date column is split into Minute, Hour, Day, Month, Year, Hour_Zone, Season, WeekOfYear, DayOfWeek_Num.

v Hour are splitted into 5 different zone 2-8, 8-12, 12-18, 18-22, 22-2.

vi Month are splitted into 4 different zone each having 3 months.

vii Address column contains 2 Street which is splited into Street 1 and Street 2.

viii Longitude and Latitude are converted from cartesian coordinate to polar coordinate.

5 Algorithms and Technique

In this work we used 4 different types of classifier.

i Logistic Regression

ii Random Forest

iii Naive Bayes classifier

iv Decision Tree

5.1 Logistic regression

Logistic Regression is a Machine Learning classification algorithm that is used to predict the probability of a categorical dependent variable. In logistic regression, the dependent variable is a binary variable that contains data coded as 1 (yes, success, etc.) or 0 (no, failure, etc.). In other words, the logistic regression model predicts P(Y=1) as a function of X. Logistic Regression is a Machine Learning classification algorithm that is used to from sklearn.linear model import Logistic Regression

5.2 Random Forest

Random forest classifier creates a set of decision trees from randomly selected subset of training set. It then aggregates the votes from different decision trees to decide the final class of the test object.

5.3 Naive bayes classifier

It is a classification technique based on Bayes' Theorem with an assumption of independence among predictors. In simple terms, a Naive Bayes classifier assumes that the presence of a feature in a class is unrelated to the presence of any other feature. Naive Bayes model is easy to build and particularly useful for very large data sets. Along with simplicity, Naive Bayes is known to outperform even highly sophisticated classification methods.

5.4 Decision Tree

Decision tree is a type of supervised learning algorithm (having a pre-defined target variable) that is mostly used in classification problems. It works for both categorical and continuous input and output variables. In this technique, we split the population or sample into two or more homogeneous sets (or sub- populations) based on most significant splitter / differentiator in input variables. Scikit learn contain class which implement decision tree classifier. We can reg- ulate the performance of decision tree by tuning various parameter.

6 Classification

6.1 Features

We have classified features into different types to train our models.

6.1.1 Time features vector

gives us the details about time and date when a particular crime occurred. It also contains Hour_zone field which give information about the zone in which crime occurred.

```
time_features = ['Minute', 'Hour', 'Day', 'Month', 'Year',
                'Hour_Zone', 'Season','WeekOfYear', 'DayOfWeek_Num']
```

6.1.2 idf

is used for mapping the result set with kaggle to get proper result

```
idf = ['Id']
```

6.1.3 Address feature vector

contains the location where crime occurred.

```
address_features = ['Street1', 'Street2', 'PdDistrict_Num',
                    'Is_Intersection', 'Is_Block']
```

6.1.4 Geometry feature vector

contains the longitude and latitude description of the crime location. We converted cartesian coordinates into polar coordinates because Polar coordinates are a non Euclidean coordinate system. This means that lines created in this coordinate system are not straight, so it isn't usually used as the primary coordinate system in a game, but rather are used in conjunction with standard Cartesian coordinates when there is a use for Polar coordinates.

```
    geometry_features = ['X', 'Y', 'Rot45_X', 'Rot45_Y',
                         'Rot30_X','Rot30_Y','Rot60_X',
                         'Rot60_Y', 'Radius', 'Angle']
```

6.2 Selection of features

Random Forest provides a feature_importances_ attribute when fitted. FeatureImportances visualizer utilizes this attribute to rank and plot relative importances. Feature Importance of the fit is as shown in Table 1

	Feature	Importance
0	Minute	0.100902
1	Rot30_Y	0.059297
2	Rot45_Y	0.055899
3	Y	0.055593
4	Rot60_X	0.053397
5	Rot60_Y	0.052418
6	Hour	0.052409
7	Angle	0.051868
8	Rot45_X	0.050898
9	Year	0.050163
10	X	0.049994
11	Radius	0.049869
12	Rot30_X	0.049784
13	Day	0.037278
14	Street1	0.035652
15	WeekOfYear	0.031690
16	Hour_Zone	0.024949
17	Month	0.022477
18	DayOfWeek_Num	0.021005
19	Is_Intersection	0.018728
20	Street2	0.017164
21	Is_Block	0.017106
22	PdDistrict_Num	0.016017
23	Season	0.011151
24	Street_Type	0.007238
25	Is_Weekend	0.007056

Table 1: Features and their importance

It is very surprising that Minute is the most important feature, while the Month just doesn't matter for Random Forest. This tells us that never ignoring a feature or put extra weights on a feature based on intuition. Only data itself can judge whether a feature is important or not. However, it is also possible that the model gets confused here, and over-fit for the noise in the Minute feature.

If that is true, this model will do a poor job in prediction. Here we believe that these features are weighted correctly, because we have a large number of trees in the forest.

6.3 Comparison of classification models

On performing the baseline model comparison Table 2, we found out that the Random Forest and Logistic Regression outperform other models greatly. why random forest are performing better than other models ? Between Random Forest and Logistic Regression, we think Random Forest may be the better option theoretically and practically. Our data set contains a mix of continuous numerical features and categorical features. The scope, mean, and variance of those feature variables are very different. The Random Forest model will split at the best point of a feature variable, so whether it is continuous or categorical doesn't matter that much for Random Forest. In contrast, the best practice for Logistic Regression is to one-hot encode all categorical features before training the model. Since we have a huge number of categorical features, one-hot encoding will increase the scale of our train data set to the next level and we will end up having a pretty sparse train data matrix, which is not ideal for Regression models. From now on, we will stick to the Random Forest in the later sections.

	Log_Loss	Model
0	2.283047	Random Forest
1	2.392173	Logistic Regression
2	7.444781	Naive Bayes
3	8.075287	Decision Tree

Table 2: Log loss of baseline model comparison

7 Result

After applying various models,our private leaderboard rank on https://www.kaggle.com is 2nd. The score we get is 2.17090(1) by applying the Random Forest with max_depth=70, n_estimators=250.

8 Conclusion and Future Work

We have explored a wide spectrum of possible classifiers that might be a good fit for solving the San Francisco Crime Classification problem. We achieved a log-loss metric 2.17 that was higher than most of the published solutions, with subtle feature engineering, and classifier parameter tuning. We still think that we can achieve much higher accuracy when we employ more feature engineering. we can use a Neural Network to train on the data to achieve a much higher accuracy with the use of a Neural Net with lot of hidden layers(2). We need to experiment with the concept of classifier fusions(3) to bring out more better results.

References

[1] Keggle Leaderboard
https://www.kaggle.com/c/IIITB-ML-Project-sfo-crime-classification/leaderboard

[2] C. Papadopoulos, *Predicting Crime Categories with Address Featurization and Neural Nets*, 2015. [Online]. https://www.kaggle.com/c/sf-crime/discussion/15836

[3] D. Ruta and B. Gabrys, *An Overview of Classifier Fusion Methods*, Computing and Information Systems, vol. 7, pp. 1–10, 200
https://www.researchgate.net/publication/229078442$_{A}n_{O}verview_{o}f_{C}lassifier_{F}usion_{M}ethods$