Report - San Francisco Crime Classification

Rui Zhang, Xinyu Zhang, Yongfeng Chen

April 22, 2016

1 Introduction

Effective prediction on what types of crime will happen in certain circumstances can help the police department to allocate their officers and resources in more of a reasonable manner. This could in turn prevent or greatly reduce the quantities and severities of the crimes. In this report we used the dataset on Kaggle, which provides nearly 12 years of crime reports from across all of San Francisco's neighborhoods, to construct a classifier of crime types given the time and location of the crimes.

The dataset includes incidents from every other week between 01/01/2003 to 05/13/2015 from SFPD Crime Incident Reporting System. The crimes are classified into 39 classes, including assault, drug/narcotic, missing person, robbery, warrants, etc. There are around 870000 observations in total. For each observation, we are given the following features:

- Dates: MM/DD/YYYY, and time of crime (precision to minute).
- DayOfWeek: One of Monday, Tuesday, ... , Sunday, crime time.
- PdDistrict: the police department district. There are 10 of them.
- Address: The (approximate) street address of the crime incident.
- X: Longitude of the location.
- Y: Latitude of the location.

Only in training set:

- Category: class of crime incidents.
- Description: Description of the crime incidents.
- Resolution: how the crime incidents were resolved.

We found there are around 20000 observations that are outside the region of San Francisco, whose region is $X \times Y \notin [-122.5, -122.38] \times [37.71, 37.82]$. For the remaining data, we partition it into 60% training and 40% as a validation set.

2 EDA and Visualization

The density plots for each category of crime:

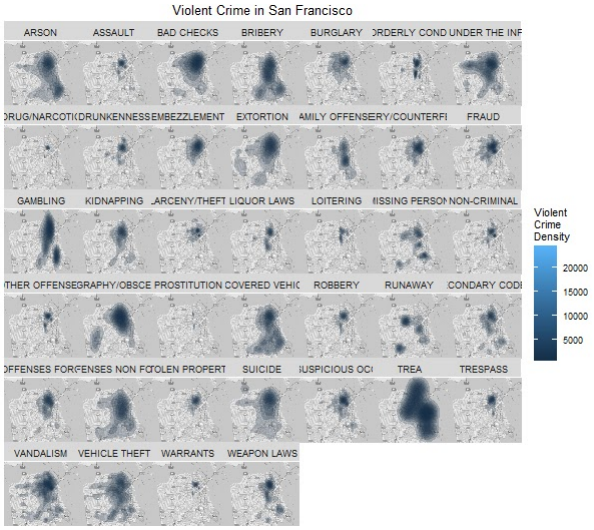

Figure 1: Density Plots for each category

The number of incidents happened in each Police District, Day of Week and Month:

Bayview	Central	Ingleside	Mission	Northern
89431	85460	78845	119908	105296
Park	Richmond	Southern	Taraval	Tenderloin
49313	45209	157182	65596	81809

Friday	Monday	Saturday	Sunday	Thursday	Tuesday	Wednesday
133734	121584	126810	116707	125038	124965	129211

April	August	December	February	January
78096	68540	65006	70813	73536
July	June	March	May	November
69971	70892	76320	79644	72975
October	September			
80274	71982			

We observe that the number of crime incidents spread evenly over time (days of week and months). However, the distribution is quite unevenly over the police districts. Observe that the district with highest number of cases is Southern district, which has four times as many cases as those in Park and Richmond. Generally speaking, the crimes in the downtown area explains most of the incidents. The resources allocation in these areas could be advised by this proportion.

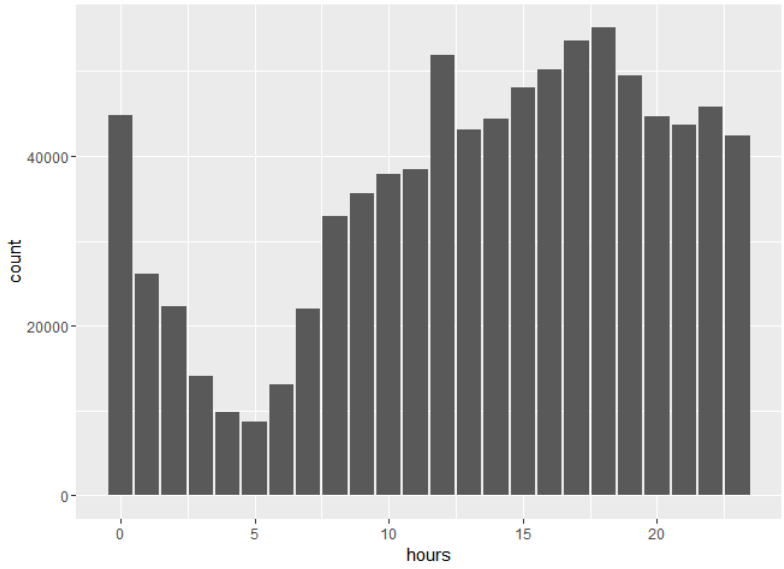

Figure 2: Number of Crimes by Hours

From the above plot we see that the number of crimes greatly differ at different times during a day. The peak of crimes happens at around 6pm, and is drastically lower in the early morning. Though this is only the marginal distribution, we infer that this might be a strong predictor. This is verified by the variable importance plot later in the discussion. Also, we suggest that based on this plot it is beneficial to have the more officers on duty during the afternoon.

Plots incorporating the type of incidents:

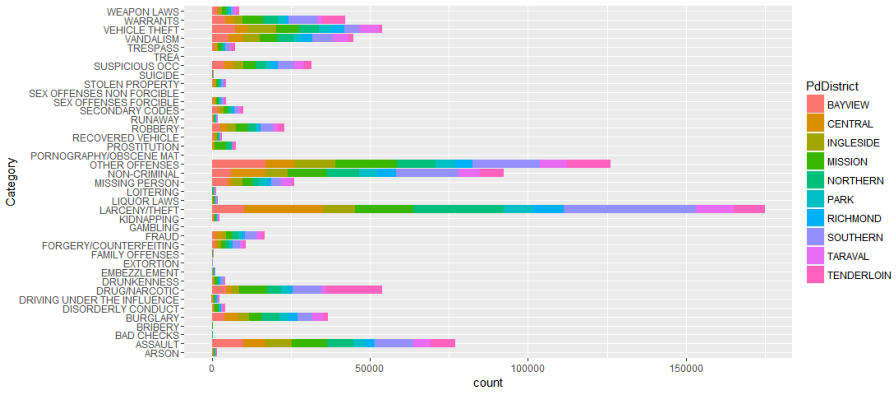

Figure 3: Category vs. District

From the above plot we can see that each class is highly unbalanced. So later in the parameter tuning for random forest we want to incorporate this unbalance into the class weights parameter. Also, we can see that the ratio of each police district is very different over various (main) type of crimes. For instance, We see that for the biggest type of crime theft, there are many incidents from the police district "Central", "Northern" and "Southern". However, in the type "other offenses", "warrants" and "Missing Person", the ratio of incidents from "Central" are among the lowest three in all district. The "Southern" district has the largest number of crime incidents but the ratio in the type "vehicle theft" is very low. So we also infer that police district might be an important predictor.

On the other hand, the ratio for each "day of week" and "month" remains quite consistent over the major types of crimes, as suggested by the following two plots:

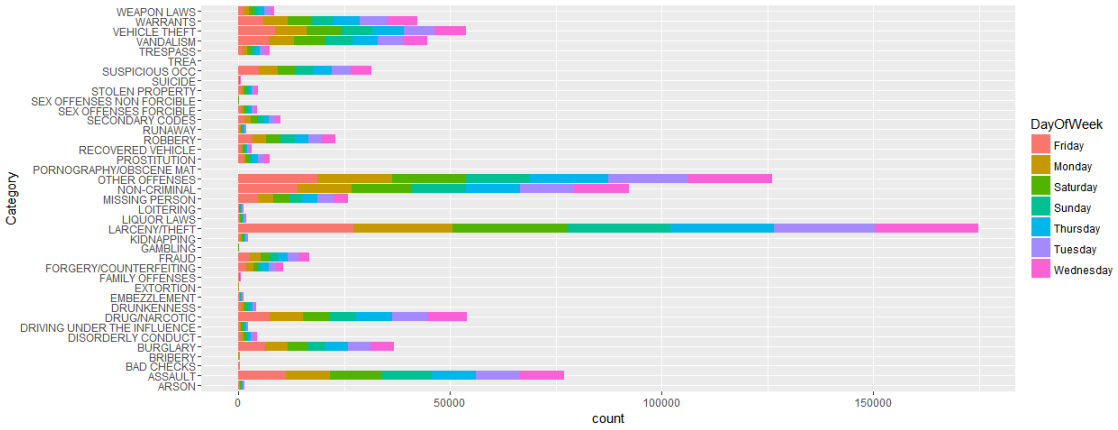

Figure 4: Category vs. Days of Week

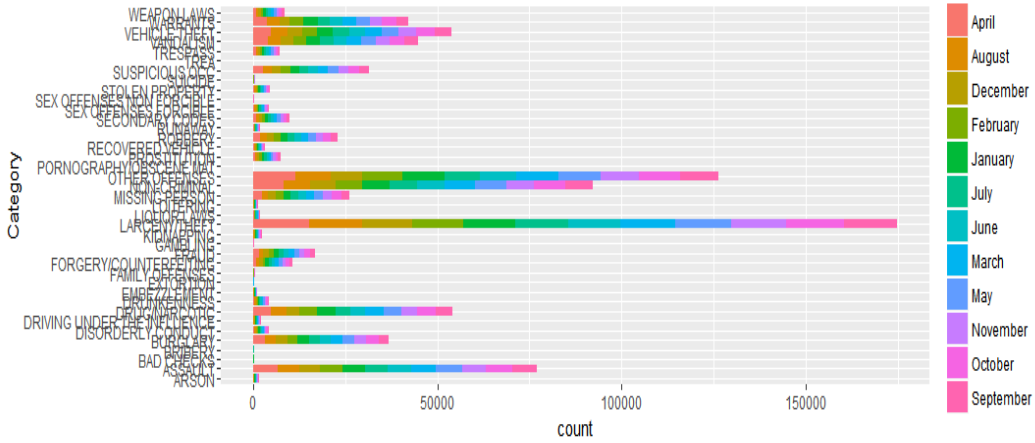

Figure 5: Category vs. Days of Week

3 Variables transformation

As we can see, our data have so many observations, but only 6 variables as the predictors. Therefore, we need to think how to map the features to a more informative feature space.

3.1 Hours and coordinates

These variables have non-linear effects. Following, I will illustrate that by using "Assault" and "Larceny/Theft" as examples.

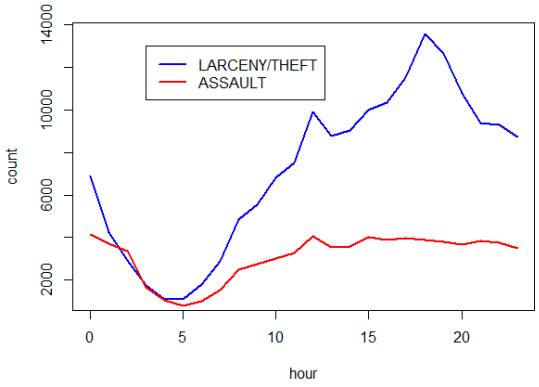

Figure 6: Association between the number of the crimes and the hours

As we can see from figure 6, the number of crimes should be a periodical function of hours, which means the number of the crimes at hour 0 should be close to the number at hour 23. Moreover, we can see the number of crimes attains the bottom at around 5 a.m (which is true for most of the crimes), and attains the peak at afternoon.

As for the longitudes and latitudes, we can see from the figure 7 and figure 8, most of the crimes have two or three maxima in their density map.

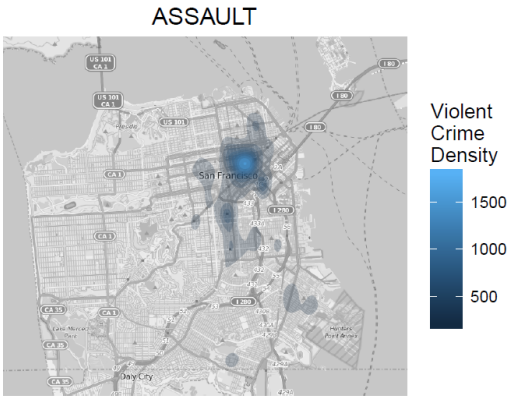

Figure 7: Density map of Assault

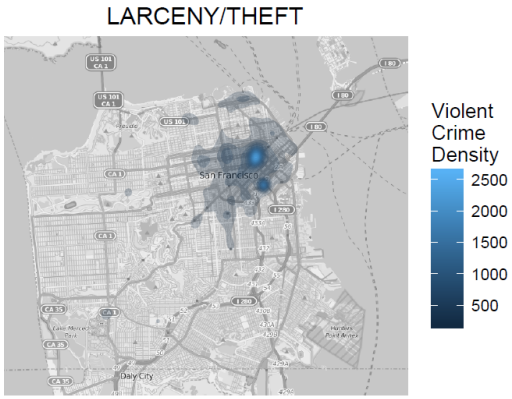

Figure 8: Density map of Larceny/Theft

Therefore, our treatments for these variables are as follow.

- Add quadratic and cubic terms of these variables as well as the interaction.
- Scale them to be mean 0 and variance 1

3.2 Address

In the data set, we have over 20,000 address name. They are expressed in two forms. One is number plus certain street, the other is the corner of a street and an avenue. In the literatures related to this data set, most authors discard this variable and claim that the longitude, latitude and PdDistrict can fully capture the information of the Address. However as shown in figure 9, no matter what the quantities are, crimes always scatter everywhere in the San Francisco.

Figure 9: Crime locations of Fraud and Runaway

Therefore, it is hard to believe that Address are useless. So our group's intuition is that maybe, other than the location, there are some other latent features of the address relating to the type of

the crime. For example, the house price, the function of that area, a business district or residential land, a school or a bar, a hospital or restaurant.

3.2.1 Address clustering

We come up an idea of clustering the address according to the distribution of the crimes happened at certain address. As figure 10,We build a matrix with 39 columns corresponding to the crimes and over 20,000 rows corresponding to the addresses. The elements of each row are the probability of certain crime given in the address.

	Crime1	Crime2	.....	Crime38	Crime39
Address1	Probability1	Probability2	.....	Probability38	Probability39

Figure 10: matrix of the crime rates

Our hypothesis is that, if the latent features of two address are similar, then they should have a similar distribution of the crimes there.

3.2.2 Kmeans

We apply kmeans algorithms to do the clustering. The reason of using kmeans is that applying kmeans does not need to much assumption of the distribution. It is not like EM, which needs guassian assumption. Since our goal is to cluster the address with similar distribution of crimes, we think the Euclidean distance of the vectors can represent this idea. I choose the cluster number K by looking into the ratio of between group variance and total variance. We can see, in figure 11, when K is larger than 12, the ratio increase slowly. So we choose K =12.

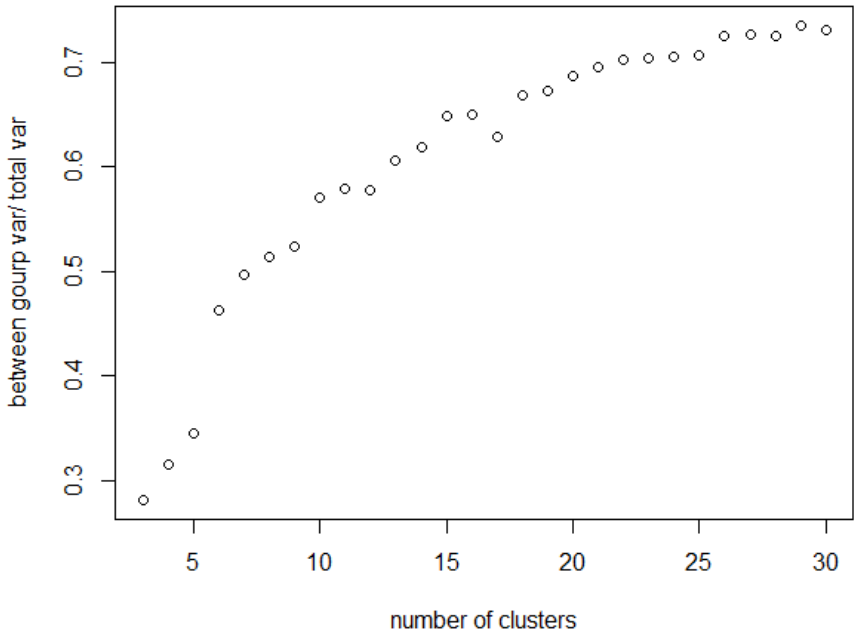

Figure 11: Criterion of choosing cluster number K

3.2.3 Result

In figure 12, the distributions within clusters are indeed distinguishing. For example, in cluster 1, the mainly crime is the Other offenses and, in cluster 6, the mainly crime is the Larceny/Theft. After we look up the addresses in the Google Map, we found some shared features within the cluster. In cluster 11, there are basically the addresses that there is no parking lots and all the cars park along the street. While in cluster 5, most of them have not on-street parking. We can see that this clustering indeed help us find the features we can't catch before, providing information in classifying the crimes. Therefore, we can expect that if we divide the data set into 12 groups, the accuracy in each group can be increased. Except group 8, which distribution is close to uniform.

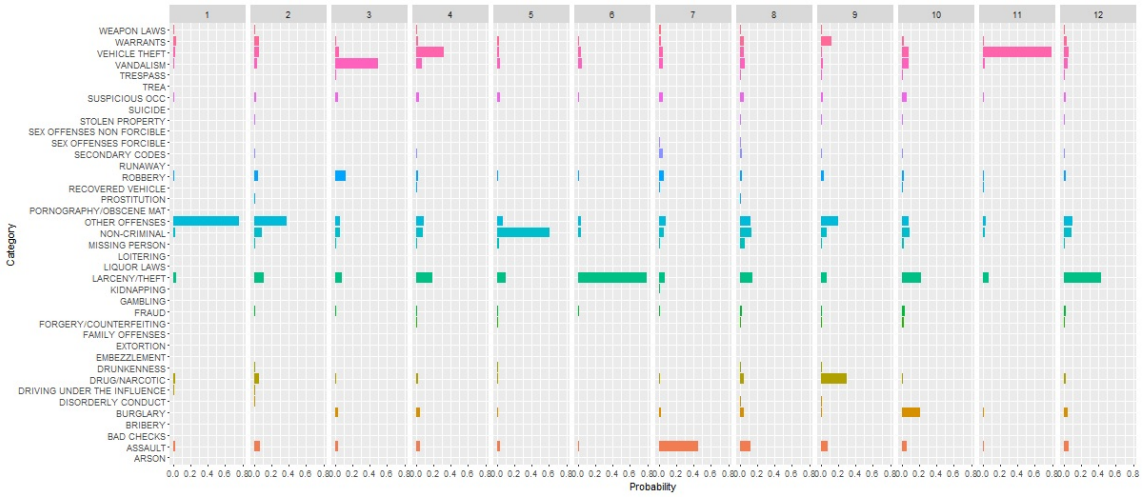

Figure 12: Distribution of the crimes within each cluser

At last, I want to point out one problem of this division. We can see, none of the low-number crimes are well represented by the cluster. If those low-number crimes are indeed related to some address features, this problem could be fixed by increasing the cluster number K. However, if the relationship between those crimes and address are weak, then this division basically divide them averagely into each cluster, which makes the classification of them more difficult, because we are making the unbalanced data set even more unbalanced.

4 Methods and Implementation

4.1 Training input

We have tried three different training input, but the data are all from the same dataset.

- Input Dataset 1:
  DayOfWeek(a seven dimension array),PdDistrict(a ten dimension arrray),HourOfDay,Longtitude, Latitude.

- Input Dataset 2:
  DayOfWeek(a seven dimension array),PdDistrict(a ten dimension arrray),HourOfDay,Second order of Hour,Longtitude(X), Latitude(Y),Second order term of X and Y, Year,Month,Day and correlation terms between Year, Month and Day

- Input Dataset 3: Variables of Input Dataset 2 + Address

Here, DayOfWeek, PdDistrict and Address are considered as categorical data. However, we didnot assign just one single numerical value to each category, i.e. we donot assign Monday to 1, Tuesday to 2 etc. The reason is that the nearby numerical values will indicates some kinds of similarity or dissimilarity. However, we do not know the relation between each class (Monday and Tuesday for example) in each category in contributing the number of crime.

Therefore, we assign each category a vector. For example, DayOfWeek contains seven class: Monday, Tuesday, Wednesday, Thursday, Friday, Saturday and Sunday. We assign each class vectors as follow:

Monday: (1,0,0,0,0,0,0)
Tuesday: (0,1,0,0,0,0,0)
Wednesday:(0,0,1,0,0,0,0)
Thursday: (0,0,0,1,0,0,0)
Friday: (0,0,0,0,1,0,0)
Saturday: (0,0,0,0,0,1,0)
Sunday: (0,0,0,0,0,0,1)

4.2 Methods

We try many different classifications methods such as Naive Bayes, Logistic Regression, SVM, Neural network and random forest. However, due to the large computional cost and long waiting time, we find neural network and SVM are not very practical in our dataset so that we only pick three prospective methods.

- Naive Bayes

  Naive Bayes methods is simple and fast in implementation. The model is simple so that it can be served as a baseline for other complicated model. Since Naive Bayes is often used in discrete dataset, we do not include continuous variables such as longtitude and latitude in doing the fitting job. In addition, the Naive bayes has assumptions that each features are independent of other other features given the class labels. It may fail to learn the interaction between features. Therefore,Navie Bayes can be done very poorly in some cases.

- Logisitc Regression

  Logistic Regression is a good model when we want the output variable to be discret rather than continuous. Error term does not have to be normal and it can handle nonlinear effects. In addition, it does not assume a linear relationship between input variables and output.

- Random Forest [1]

  Random Forest is consist of a set of decision trees(which will be characterized as n_estimator in python sklearn package). It is easy to handle feature interactions and they are non-parametric. Therefore, people do not have to worry about linearly separable and outliers. It builds classification using tree strucutres: diving a dataset into smaller subsets. At the same time, an associated decision tree is incrementally developed. The final result is a tree with decision nodes and leaf nodes. Leaf node represents a classification or decision.

4.3 Input Dataset Performance

- logloss:

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

N is the number of incidents in the validation set,M is the number of class labels, log is the natural logarithm,y_{ij} is 1 if observation i is in class j and 0 otherwise, and p_{ij} is the predicted probability that observation i belongs to class j. Log loss punishes this lack of certainty. Therefore, log loss characterizes the extent for being more certain that the true answer is correct, while accuracy characterizes the extent for having the highest probability for the true answer (being "correct").

- Accuracy:

$$\text{Accuracy} = \frac{\#\{i : y_{\text{predict},i} = y_{\text{true},i}\}}{\#\text{test data size}}$$

Here, the classification program will first go through training process given a dataset with around $500,000$ sample size containing information about the locations, districts and times of the crime. Then we will give some test data to test the performance of the classification program. The output will give two set of value. The first set of value is an integer from ranging from 0 to 38 for each input. This is used for evaluating the accuracy. The second set of output is an array with dimension 39. Each entry (i entry for instance) on the array indicates the probability that crime i will be identified given the input data. This is used for calculating the log loss.

- Input Dataset 1:

	Traning Accuracy	Validation Accuracy	Log_Loss
Naïve Bayes	22.2090	22.1980	2.6097
Logistic Regression	22.3074	22.2510	2.5923
Random Forest	33.0549	26.4296	2.5778

- Input Dataset 2:

	Traning Accuracy	Validation Accuracy	Log_Loss
Naïve Bayes	19.3093	19.2731	2.8582
Logistic Regression	23.2612	23.2289	2.5794
Random Forest	43.5927	29.1193	2.4000

- Input Dataset 3:

	Traning Accuracy	Validation Accuracy	Log_Loss
Naïve Bayes	24.4711	24.4686	2.8471
Logistic Regression	27.8296	27.7968	2.4661
Random Forest	37.6344	31.0043	2.3288

As we can see from the table, among the three methods, Random Forest perform the best, hitting the optimal validation accuracy of around 31%. Due to the simple and naive assumption of the Naive Bayes assumption, it performs the worst among all three cases. When we add more input variables just we did in dataset 2 and dataset 3, the independent variable assumption of Naive Bayes is heavily violated.

If we look at the accuracy as we refine our dataset, we can also see an increasing tendecy. For random forest, we saw an nearly 3% accuracy increase when we add the second order term of times and also second order and correlation terms of the locations. Also, when we do EDA and believe that address will also have an impact on the predicting performance, we add the address variable and get 2% accuracy increase for random forest.

Log loss is also a good indicator of how well the performance of classification is. In generally, a lower log loss will indicated higher accuracy. However, it is not always the case. For example, in three-class classification program. Model A predicts the probability of category [a,b,c] to be [0.3,0.4,0.3], model B predicts it to be [0.1,0.41,0.49], and the correct category is "b". Then model A will predict it correctly but model B will predict it wrong. The when we calcualte the log loss, we take the second order term to compute (since "b" is correct). In this case, model B is wrong but $0.41 > 0.4$.

5 Parameter Tuning and DivideandConquer

5.1 Parameter Tuning

- n estimator:The number of trees in the forest.
- max features: max features at each split.
- max depth: The maximum depth of the tree.

For n_estimator, we have

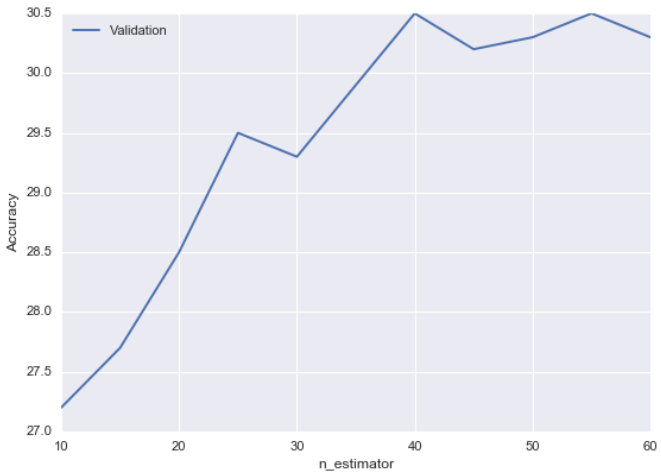

Figure 13: n_estimator v.s. accuracy

For n_estimator, in general, as we increase the number of decision tree, we will be more robust to the new input data. As the figure indicates, as n_estimator increases, the validation accuracy also increases. However, when the number of decesion trees grows, the computation time also grows.

In order to find a balance between accuracy and cost of time, We pick n_estimator to be 40 for the classification job.

For max_feature and max_depth, we have

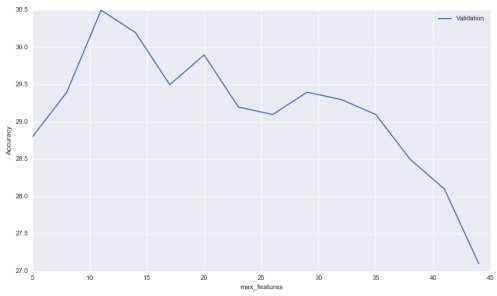

Figure 14: max_feature v.s. accuracy

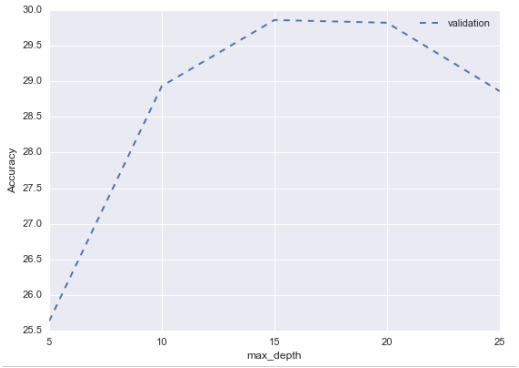

Figure 15: max_depth v.s. accuracy

max features and max depth are also very important parameter when we are tuning the random forest. If we pick these two variables too large, we will run into overfitting problem, which will draw the validation accuracy lower. Therefore, we pick (from the plots above) max_feature to be 12 and max_depth to be 20.

5.2 Divide and Conquer

- Dataset: Variables of Input Dataset 2 + Address
- In order to further improve our prediction performance. We use a "divide and conquer method" for our prediction job. Since we cluster the address into 12 different class. We separate the training data into 12 sub-train data according to the different class of address. We then make a classification separately on these 12 training dataset. The result is as follow.

	Traning Accuracy	Validation Accuracy	Log_Loss
Logistic Regression	27.9954356	27.923565	2.501734944
Random Forest	41.4543526	31.96940749	2.405594683

As we can see from the table, we can see a 2% accuracy increase from dataset 3 to dataset_divideandconquer in random forest case. The logistic regression technique also demonstate an improved performance. But we should note that the log loss increase compared to dataset 3. As I mention in section 4.3, it is possible that we have higher accuracy but log loss also increase.

To make a summary of all the dataset we use, I plot the accuracy with respect to dataset under random forest classification.

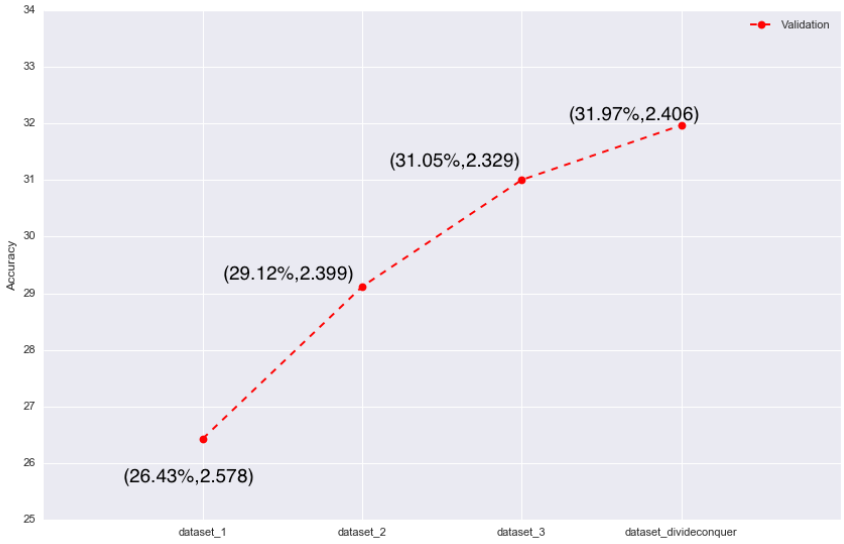

Figure 16: summary of accuracy

To get a idea of how well our performance, here is some benchmark for refernece.

- Compare with Kaggle Leaderboard: The total number of competitors is 2216 and the total number of entry is 13135. The median Log Loss is 2.60027.Our best performance is 2.32 (it happens in the dataset 3 case), rank top 10% (rank 197 out of 2216) of all competitors.

6 Potential Approaches

- Include any other information pertaining to the geography or demographics of the city. It is possible to use information from other datasets that would be useful for predicting crime, e.g. using power consumption, public transit travel data during various times of the day.[2],[3]
- Neural networks. In many applications, neural networks have displaced the previous state-of-the-art methods as they deliver better classification accuracy.

References

[1] S. Ihm A. Nasridinov and Y. Park. A decision tree- based classification model for crime prediction. *Infor- mation Technology Convergence*, pages 531–538, 2013.

[2] Brian Lam. Correlation between weather condition and the type of crime. *https://nycdatascience.com/correlation-between-weather-condition-and-the-type-of-crime/*.

[3] A. Bogomolov. Once upon a crime: Towards crime prediction from demographics and mobile data. 2014.