5. Method and Results:

we have integrated the methodology and the results we obtained while working on classifiers in this section.

a.

dataset information include information of training and test data.

The majority class of the dataset is LARCENY/THEFT which consists of around 19% of the total training data so if we assign all values to this class we must get an accuracy of 19%, so the accuracy of the classifier should at least be more than 19%.

b.

The project consists of two parts firstly the visualization of data and secondly the classification of 39 class variables.

Considering the data we decided to use the following classifiers:

1. Gaussian Naïve Bayes
2. Multinomial Naïve Bayes
3. Decision Tree
4. Random Forest
5. Support Vector Machine

Approach 1: converting continuous variables to categorical

For testing these classifiers on the data we transformed the continuous variables from the dataset like Address, time, X (Longitude) and Y (Latitude) and further map these categorical variables with a particular number as the scikit learn library works only on numbers.

The dates column which was like *5/13/2015 11:53:00 AM* was further split into year, month, day and hour.

X and Y were discretized into 5 bins.

Address which was of the form

1. 1500 Block of LOMBARD ST
2. OAK ST / LAGUNA ST

Was processed and only the street was extracted. The first one is an actual location whereas the second is the intersection of two streets. Thus the number of unique addresses was drastically reduced and later each unique address was mapped with a unique number.

The first 5 rows of the transformed dataset is as follows

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **DayOfWeek** | **PdDistrict** | **Address** | **X** | **Y** | **time** | **day** | **month** | **year** |
| 2 | 4 | 7677 | 3 | 0 | 23 | 13 | 5 | 2015 |
| 2 | 4 | 7677 | 3 | 0 | 23 | 13 | 5 | 2015 |
| 2 | 4 | 10556 | 3 | 0 | 23 | 13 | 5 | 2015 |
| 2 | 4 | 1641 | 3 | 0 | 23 | 13 | 5 | 2015 |
| 2 | 5 | 762 | 4 | 0 | 23 | 13 | 5 | 2015 |

Table 1: Initial Dataset

Initially, we selected all the variables and fed the data to the classifier, all of the classifiers gave the accuracy below par that is it was below 19% for all the classifiers as shown in table 2 left.

Later on we decided we find Pearson’s correlation between all variables and category as follows and then select the top correlated terms as shown in table 2 right.

|  |  |
| --- | --- |
|  | **Category** |
| **DayOfWeek** | 0.001078 |
| **PdDistrict** | -0.040674 |
| **Address** | 0.050874 |
| **X** | 0.02951 |
| **Y** | 0.00287 |
| **time** | 0.023524 |
| **day** | 0.000805 |
| **month** | 0.000008 |
| **year** | -0.021803 |

|  |  |
| --- | --- |
| **Classifier** | **Accuracy(%)** |
| Gaussian Naïve Bayes | 18.92 |
| Decision Tree | 3.35 |
| Random Forest(estimators=2) | 6.62 |
| Support Vector Machines | 4.21 |

Table 2: Initial Accuracy and Pearson Correlation

Then we dropped the columns having lower correlation coefficient that is nearer to 0 so we eliminated month, day and DayOfWeek. After running all the classifiers we got the accuracy as follows in table 3.

|  |  |
| --- | --- |
| **Classifier** | **Accuracy (%)** |
| Gaussian Naïve Bayes | 18.68 |
| Decision Tree | 10.04 |
| Random Forest(estimators=2) | 10.38 |
| Support Vector Machines | 5.32 |

Table 3: Accuracy after removal of least correlated columns

Here we can see that the accuracy of of all classifiers have increased except that of Naïve Bayes. This clearly indicates that there is a nonlinear relationship between the class and other columns hence the accuracy is below the benchmark that is 19%. So we need to combine this attributes such that they give us a good nonlinear correlation and hence increase the accuracy.

After this we also tried normalizing the values of each rows and columns using Z-Score Normalization and Min Max normalization but still the result did not change.

After many trial and errors and considering the combination of various attributes we got the best accuracy for the following combinations

|  |  |
| --- | --- |
| **Classifier** | **Accuracy (%)** |
| Gaussian Naïve Bayes | 20.20 |
| Decision Tree | 13.46 |
| Random Forest(estimators=2) | 12.67 |
| Support Vector Machines | 7.37 |

Table 4: best accuracy after trial and error

The highest accuracy we got was that for Naïve Bayes that was just somewhat above 20% which was better than the benchmark but still needed some improvement, whereas the other classifiers were still performing badly. After all these results we thought of some other approaches to move on then we moved to the next approach

Approach 2: Binarization of categorical variables

In this approach the data points consist of only binary variables 1 and 0 depending if the attribute is present in the data point. As shown in figure 1.

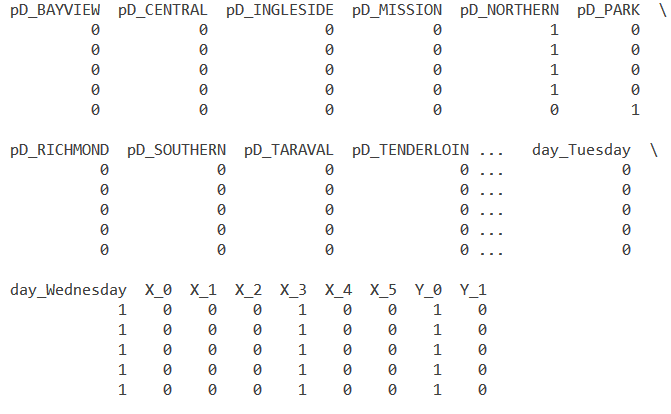


Figure 1: first 5 rows of Initial Dataset for Binarization

Here we see that the first row of the dataset is in district Northern hence its value 1 and 0 for other districts and similarly day is Wednesday. We have made 5 bins for continuous variable X and 2 bins for 2 bins for Y after trial and error for optimacy.

Here instead of Gaussian we used multinomial Naïve Bayes which is best suited for such kind of data.

After running the classifiers over this transformation we got the following accuracies

|  |  |
| --- | --- |
| **Classifier** | **Accuracy (%)** |
| Multinomial Naïve Bayes | 21.96 |
| Decision Tree | 22.08 |
| Random Forest(estimators=2) | 22.06 |
| Support Vector Machines | 20.33 |

Table 5: initial accuracy for Binarization

Here we see that the accuracy has significantly increased and all classifiers give accuracy above the benchmark that is 19%.

After this we created many extra features from the existing dataset which worked for increasing the accuracy of the classifiers.

**Dimensionality reduction**

1. Generating feature based on time:

If we see from above visualizations (Insert figure numbers) we see that the state of crime changes with time firstly we created the feature *awake* which is 1 when the people are awake and 0 otherwise in night hours, the accuracy further increased after grouping them into 4 categories as early morning, morning, afternoon, evening, and night as we see in figure (). We further generalized and divided them into a group of 6 categories as per hours as follows

[4,5,6,7], [8,9,10,11], [12,13,14,15], [16,17,18,19], [20,21,22,23], [0,1,2,3] as just arbitrary alphabets a,b,c,d,e,f respectively.

1. Generating feature based on address:

There are around 9000 unique addresses in the previous approach after just taking the streets replicating them in this approach won’t work as it would increase the dimensionality and in turn may suffer from the curse of dimensionality. Hence we decided to make the feature as *intersection* if the address contains “/” as it indicates intersection of two streets. It indicates 1 if it is an intersection and 0 otherwise

1. Generating features based on month:

We also tried to group months into *seasons* viz. summer, winter, spring and fall but it had an adverse effect on accuracy hence later dropped it.

1. Generating features based on days of month:

We also tried to generate feature *first half* indicating 1 if day is less than 15 and 0 otherwise and other combinations by grouping but this too had an adverse effect on accuracy hence dropped it.

After trial and error method with all the methods we got best accuracy till now considering the following feature vector.

*['pD\_BAYVIEW', 'pD\_CENTRAL', 'pD\_INGLESIDE', 'pD\_MISSION', 'pD\_NORTHERN', 'pD\_PARK', 'pD\_RICHMOND', 'pD\_SOUTHERN', 'pD\_TARAVAL', 'pD\_TENDERLOIN', 'day\_Friday', 'day\_Monday', 'day\_Saturday', 'day\_Sunday', 'day\_Thursday', 'day\_Tuesday', 'day\_Wednesday', 'a', 'b', 'c', 'd', 'e', 'f', 'intersection', 'X\_0', 'X\_1', 'X\_2', 'X\_3', 'X\_4', 'X\_5', 'Y\_0', 'Y\_1']*

The accuracy of various classifiers we got is as follows:

|  |  |
| --- | --- |
| **Classifier** | **Accuracy (%)** |
| Multinomial Naïve Bayes | 22.5 |
| Decision Tree | 23.16 |
| Random Forest(estimators=2) | 23.06 |
| Support Vector Machines | 21.27 |

Table 6: Best accuracy till date

After this we submitted our results to Kaggle with the decision tree classifier which had the best accuracy on training dataset. We got a rank of **1149 with a score of 2.64 out of around 1886** competitors, and later when we sent our submission for Naïve bayes we got a rank of **672 with a score of 2.55**



From the link <https://www.kaggle.com/c/sf-crime/leaderboard>

As the test dataset seemed to more favorable to Naïve Bayes classifier than the decision tree.