Analyzing and Classifying Crime in Chicago

Dhruv Gupta, Shinjini Srivastava, Hsiu-Yuan Fan
Department of Computer Science
Georgia State University
Atlanta, Georgia

Abstract— This study uses the Chicago Crimes dataset to conduct an exploratory data analysis on the factors affecting crime rate and find correlations in the elements that can be used for crime prediction. To do this, we have used several machine learning algorithms to answer research questions like rate of change in crime rate over a period of time, types of crimes which have the highest occurrence, etc. Specifically, we built a multi class – classifier and did performance comparison. The Naïve Bayes classifier gave us 75 per cent accuracy, while on the other hand the classifier built using Random Forest gave our model an accuracy which was greater than 95 per cent.

Keywords – Data Mining; Classification;

I. INTRODUCTION

Crime is an inherent aspect of a rules-based society. Like many previous generations before us, modern society is still plagued by the same ills as before. The success of the social structure, economy and other factors related to the ability of a person to live and progress in life are linked to the ability of the policing organizations to respond to changes in how crime is committed and redress is provided to the victims by identifying the culprits. Keeping this in mind, this project aims to study some of the factors that affect crime with the use of machine learning techniques. Using data mining techniques authorities can better understand and predict types of crime making them better at both preventing crime and catching the culprits.

Crime rates in a city vary drastically over a period of time and are dependent on multiple different factors. The factors that we want to study are the correlation between the type of crime and location, and the seriousness of the crime.

II. PROBLEM STATEMENT

- Conduct data cleaning
- Visualize initial trends in crime data
- Conduct exploratory data analysis (EDA) to find the most important variables for predicting type crime
- Confirm that such a prediction is possible to make using machine learning techniques
- Compare different approaches for classification and identify the most efficient method

III. BACKGROUND

Data Set

The dataset being used in this exercise is available online [1].

The data set has the following features:

- ID: Unique identifier for the record.
- Case Number: The Chicago Police Department RD Number (Records Division Number), which is unique to the incident.
- Date: when the incident occurred. this is sometimes a best estimate.
- Block: The partially redacted address where the incident occurred, placing it on the same block as the actual address.
- IUCR: The Illinois Uniform Crime Reporting code.
- Primary Type: The primary description of the IUCR code.
- Description: The secondary description of the IUCR code, a subcategory of the primary description.
- Location Description: Description of the location where the incident occurred.
- Arrest: Indicates whether an arrest was made.
- Domestic: Indicates whether the incident was domestic-related as defined by the Illinois Domestic Violence Act.
- Beat: Indicates the beat where the incident occurred. A beat is the smallest police geographic area – each beat has a dedicated police beat car. Three to five beats make up a police sector, and three sectors make up a police district. The Chicago Police Department has 22 police districts.
- District: Indicates the police district where the incident occurred.
- Ward: The ward (City Council district) where the incident occurred.
- Community Area: Indicates the community area where the incident occurred. Chicago has 77 community areas.
- FBI Code: Indicates the crime classification as outlined in the FBI's National Incident-Based Reporting System (NIBRS)
- X Coordinate: The x coordinate of the location where the incident occurred in State Plane Illinois East NAD 1983 projection. This location is shifted from the actual location for partial redaction but falls on the same block.
- Y Coordinate
- Year: Year the incident occurred.
- Updated On: Date and time the record was last updated.
- Latitude
- Longitude
- Location: Combination of X-coordinate and Y coordinate

Based on the location information predict the type of crime

Data Mining Tools

Principal Component Analysis

Principal components analysis (PCA) is one of a family of techniques for taking high-dimensional data and using the dependencies between the variables to represent it in a more tractable, lower-dimensional form, without losing too much information.

Each principle component is a curve fitted to the data of all the features being reduced

Each PC represents the % of total variance of the model and reduces with each new PC

K – Means

K means clustering is an unsupervised machine learning technique which is useful for partitional clustering. It gives insight into the internal structure of the data and partitions between the different groups.

Naïve Bayes Classifier

Naïve Bayes Classifier is a supervised machine learning technique which is useful for predicting an outcome by using a probabilistic model using mutually exclusive / independent variables.

Being a probabilistic model, it is highly scalable however the accuracy for non gaussian distributions is low. Care also needs to be taken to ensure that the variables are independent of each other.

Decision Tree

Learning the simplest decision tree is an NP complete problem

Greedy Heuristics must therefore be used to build a decision tree

- Start with empty tree
- Choose best feature to split tree
- Recurse
- Best feature for split
- Good split increases certainty about classification after split = Reduces Entropy
- Information Gain = Entropy before split – Entropy after split
- Tree Building Summary
- If output values are all same in the dataset, return leaf node
- If all inputs are same, return leaf node that predicts most of the output
- Otherwise find attribute with max information gain

Random Forest

- Random Forrest is an ensemble learner which can be used for Classification and Regression.
- Ensemble learners are composite ML tools which use multiple different learners in combination
- Random Forest is composed of multiple decision trees
- Each tree in the Forest is built from a different combination of columns
- Finally for classifying data, each new data element runs through each tree and the mode of results from all trees is returned => Bagging [Bootstrap Aggregation]

Receiver Operating Curve

The receiver operating characteristics is a plot of the True positive versus the False positive rate. Another paradigm to present the ROC is the grouping of best decision boundaries for relative costs of TP and FP [2]. ROC curve can be swept out by manipulating the balance of training samples for each class in the training set.

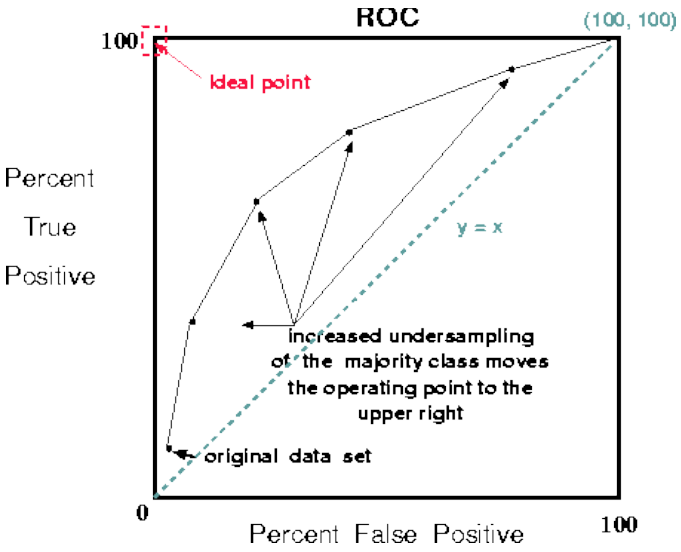


Figure 1: Explanation for working of ROC Curve

IV. METHODOLOGY

Data Cleaning

Initial Data Set

ID	Case Number	Date	Block	IUCR	Primary Type	Description	Location Description	Arrest	...	Ward	Community Area	FBI Code	X Coordinate
10508693	HZ250496	05/03/2016 11:40:00 PM	013XX S SAWYER AVE	0486	BATTERY	DOMESTIC BATTERY SIMPLE	APARTMENT	True	...	24.0	29.0	08B	1154907.0
10508695	HZ250409	05/03/2016 09:40:00 PM	061XX S DREXEL AVE	0486	BATTERY	DOMESTIC BATTERY SIMPLE	RESIDENCE	False	...	20.0	42.0	08B	1183066.0
10508697	HZ250503	05/03/2016 11:31:00 PM	053XX W CHICAGO AVE	0470	PUBLIC PEACE VIOLATION	RECKLESS CONDUCT	STREET	False	...	37.0	25.0	24	1140789.0

Y Coordinate	Year	Updated On	Latitude	Longitude	Location
1893681.0	2016	05/10/2016 03:56:50 PM	41.864073	-87.706819	(41.864073157, -87.706818608)
1864330.0	2016	05/10/2016 03:56:50 PM	41.782922	-87.604363	(41.782921527, -87.60436317)
1904819.0	2016	05/10/2016 03:56:50 PM	41.894908	-87.758372	(41.894908283, -87.758371958)

1. Remove Repeated Values

Some of the values are not useful for analysis because they are either repeated, highly correlated to another variable or are completely unique to each entry acting like a primary key

- Unnamed
- ID
- Case Number
- Location
- X coordinate
- Y Coordinate

Unnamed, ID and Case Number are unique to each record and are not useful for machine learning techniques.

Location is composed of X-coordinate and Y-coordinate and so it can be dropped since they already exist in separate columns.

X – coordinate and Y- coordinate are coordinates from a Chicago only map, since the scale is different compared to a global map the coordinates are different however highly correlated to the Longitude and Latitude and therefore can be dropped with the preference for the long lat.

2. Imputation

```
Primary Type 0
Description 0
Location Description 1658
Arrest 0
Domestic 0
Beat 0
District 1
Ward 14
Community Area 40
FBI Code 0
X Coordinate 37083
Y Coordinate 37083
Year 0
Updated On 0
Latitude 37083
Longitude 37083
Location 37083

dtype: int64
```

Figure 2: Figure showing count of missing values in the dataset

The dataset is missing significant number of values under the columns that are needed by the machine learning classifiers. Therefore imputation of the missing values is required.

Imputation of Longitude and Latitude

For the available location information, the least about of data that is missing in in the Ward column. A ward is the smallest administrative – geographic division and there the average longitude and latitude for the ward will have relatively high accuracy for the estimation of missing values.

The imputation is carried out using the method in the following steps:

- Calculate the mean Long / Lat for each ward
- Build lookup table
- For each missing Long Lat, look up the ward and replace the missing value with the value in look up table

```
features = ["Ward", "Longitude", "Latitude"]
temp = (df[features]).groupby("Ward").mean()
temp.head()
```

Ward	Longitude	Latitude
1.0	-87.681166	41.910664
2.0	-87.653171	41.870792
3.0	-87.626850	41.814090
4.0	-87.603932	41.814684
5.0	-87.588531	41.765896

Figure 3: Figures showing process of calculating mean long lat and building look up table

```
df.loc[df['Longitude'].isnull(),'Longitude'] = df['Ward'].map(temp.Longitude)
df.loc[df['Latitude'].isnull(),'Latitude'] = df['Ward'].map(temp.Latitude)
```

Figure 4: Performing Imputation

```
Updated On          0
Latitude            0
Longitude           0
Location        37083
dtype: int64
```

Figure 5: Results of Imputation

Imputation of Location Description

- For each ward, the most occurring community area is calculated and kept in a table
- For each entry with a missing community area:
  - Lookup the table with the ward as key and replace missing value

Ward	Location Description	
	ABANDONED BUILDING	17
	AIRPORT VENDING ESTABLISHMENT	1
1.0	ALLEY	489
	ANIMAL HOSPITAL	4
	APARTMENT	2438
...	...	...
	VEHICLE - DELIVERY TRUCK	1
	VEHICLE - OTHER RIDE SERVICE	2
50.0	VEHICLE NON-COMMERCIAL	190
	VEHICLE-COMMERCIAL	8
	WAREHOUSE	4

Ward	Location	cou
1.0	STREET	75
2.0	STREET	109
3.0	STREET	88
4.0	STREET	59
5.0	APARTMENT	90
6.0	STREET	115
7.0	APARTMENT	88

Figure 6: Look up table being created for missing location description

The replacement of the location description based on mode of location descriptors of the ward is important because a certain block is developed based on land use patterns which are set forth by the city council. This implies that there can't be an even mix of warehouses and apartment complexes in the same ward. Since the crime profile is different for different land use patterns the missing location descriptions.

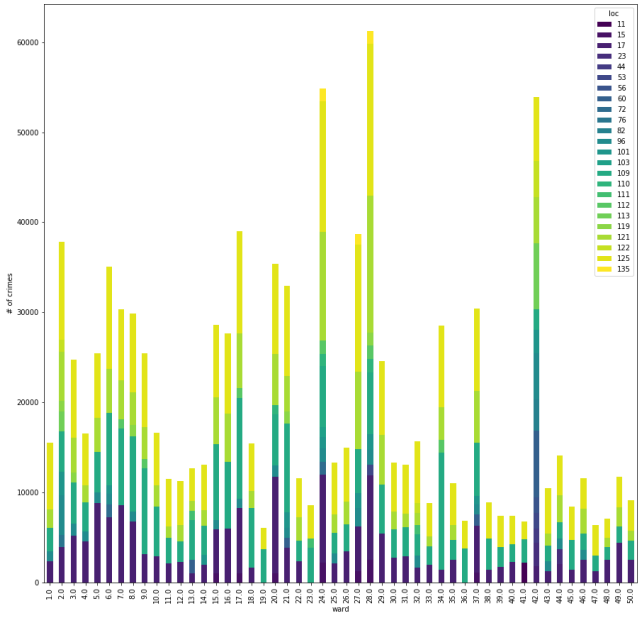


Figure 7: Graph showing the breakdown in the number of crimes per ward. The color represents types of crime

Imputation of Community Area

- For each ward, the most occurring community area is calculated and kept in a table
- For each entry with a missing community area:
  - Lookup the table with the ward as key and replace missing value

```
features = ["Ward", "Community Area"]
temp = (df[features]).groupby(features).aggregate({"Community Area":['count']})
temp = temp.reset_index()
temp.columns = ['Ward', 'Community', 'count']

df_agg = temp.groupby(['Ward','Community']).agg({'count':sum})
g = df_agg['count'].groupby(level=0, group_keys=False)
res = g.apply(lambda x: x.sort_values(ascending=False).head(1))

res = res.reset_index()
df.loc[df["Community Area"].isnull(),"Community Area"]  = df['Ward'].map(res.Community)
```

Imputation of District Information

Since ward is smallest measure of geography in the dataset under consideration a given district will contain multiple wards and each ward will be associated to only one district. As a result when the district information is missing, the ward information can be used as a look up index and the result of the look up can be used for imputation.

```
features = ["Ward", "District"]
temp = (df[features]).groupby(features).aggregate({"District":['count']})
temp = temp.reset_index()
temp.columns = ['Ward', 'District', 'count']

df_agg = temp.groupby(['Ward','District']).agg({'count':sum})
g = df_agg['count'].groupby(level=0, group_keys=False)
res = g.apply(lambda x: x.sort_values(ascending=False).head(1))

res = res.reset_index()
df.loc[df["District"].isnull(),"District"]  = df['Ward'].map(res.District)
```

```
# df["Community Area"].map()
df.loc[df["Community Area"].isnull(),"Community Area"]
```

Imputation of Ward

The number of wards with missing data is very small (14) out of a dataset size of 1,456,714. Due to the insignificant number of missing values the rows with missing Ward information is simply dropped.

3. Exploratory Data Analysis

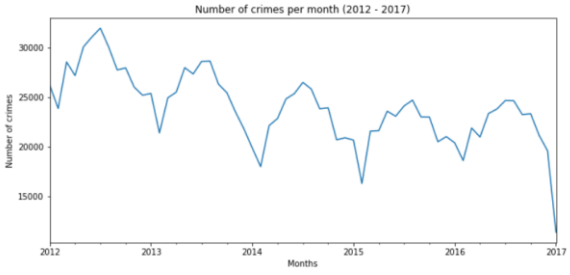


Figure 8: Change in Crime Rate in Chicago from 2012 to 2017

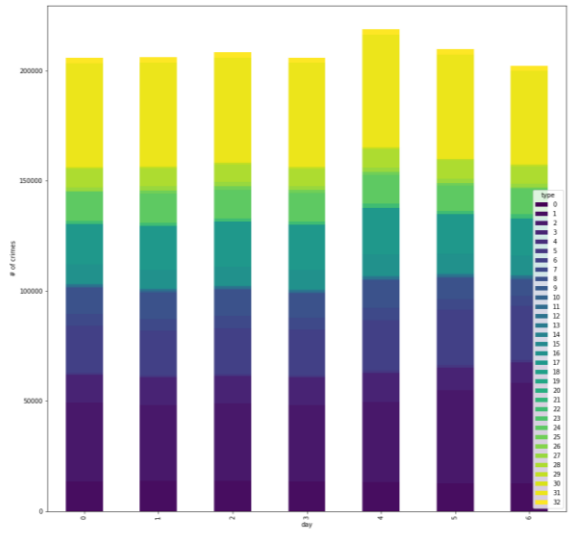


Figure 9: Graph showing number of crimes based on day with a breakdown of type of crime

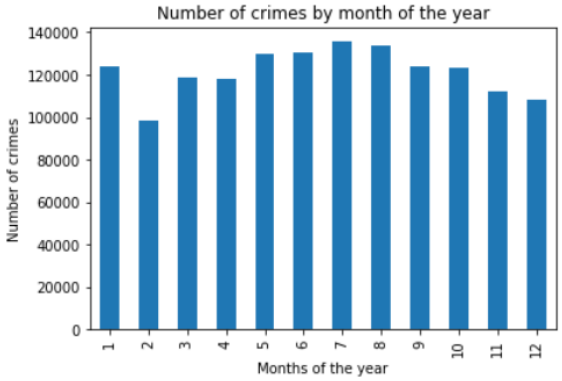


Figure 10: Change in the number of crimes by month

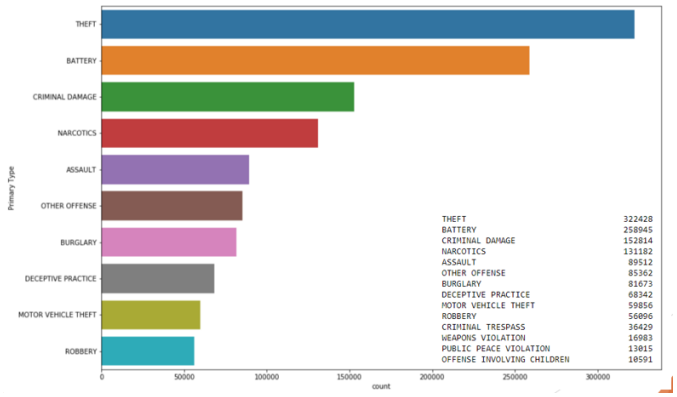
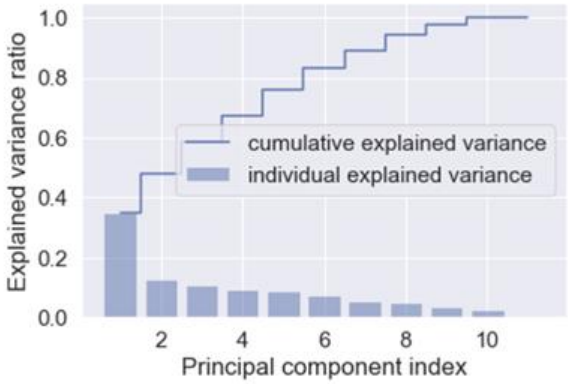


Figure 11: Most Common crimes in Chicago

Figure 12: Map of locations with the most theft occurring

V. RESULTS

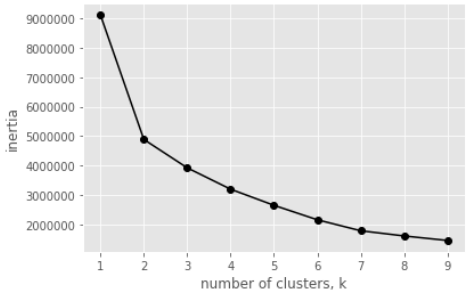
Principle Component Analysis




```
Variance: [0.38379316 0.12059172
0.10386811]
```

K – Means

K means clustering requires the number of clusters to be predefined. To find out the optimum number of clusters, the inertia vs number of clusters graph can be calculated. The goal is to have low inertia and low number of clusters. The "Elbow" is the point in the graph which represents a good balance between the inertia and number of clusters



From the study of the inertia curve, the number of clusters is chosen as 6.

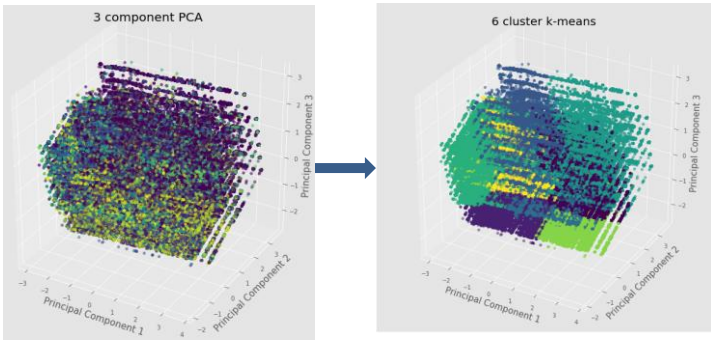


Figure 13: Results of clustering the dataset into 6 clusters.

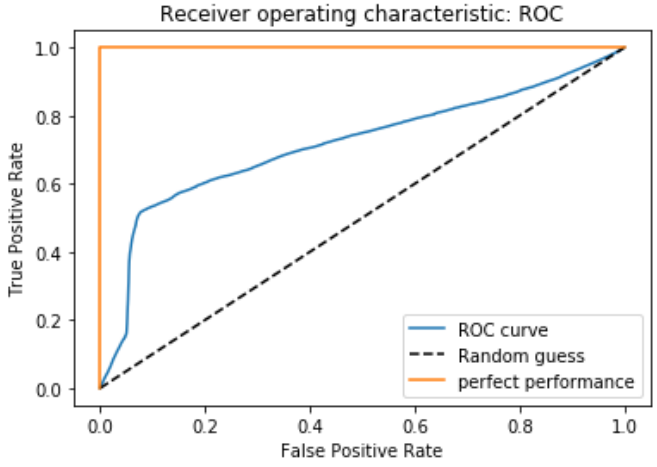
The clustering process shows that the 6 groups are well clustered and the overlap between classes is not too much. In the figure 1.5 million points are plotted together. This gives the authors confidence about the ability to get good results from a classification algorithm. It also shows that the data is

Predicting Arrests

Naïve Bayes

Naïve Bayes is a probabilistic classifier that calculates the independent probabilities of each event and the conditional probabilities with the target variable and then uses this probabilistic model to classify the test data. The classifier is Naïve since it assumes that the input variables are all independent of each other.

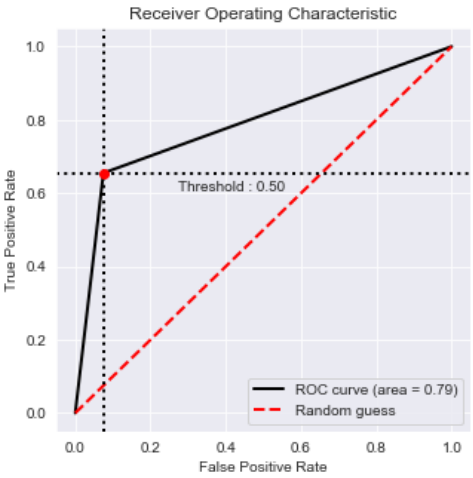
In this section, the 'Arrest' variable has been used as the target variable and location information including longitude - latitude, ward, community area, location description etc. are taken as input variables.



- Precision = TP/(TP+FP) = 0.8438
- Recall = Sensitivity = TP/(TP+FN) = 0.9208
- Specificity = TN/(TN+TP) = 0.1662
- Accuracy: 0.8157

Random Forrest

Random Forest has been used to run a comparative classifier to Naïve Bayes. The accuracy obtained from Random Forest is 2% better than Naïve Bayes



	precision	recall	f1-score	support
0	0.88	0.93	0.90	314271
1	0.76	0.65	0.70	111610
accuracy			0.85	425881
macro avg	0.82	0.79	0.80	425881
weighted avg	0.85	0.85	0.85	425881

Figure 14: ROC curve and classification report for Random Forest

Predicting Type of Crime

Random Forest

Random forest classifier is an ensemble learner (learners made of multiple base learners), composed of multiple decision trees. Each decision tree is built using a separate set of variables and the process of calculating the maximum decrease in Entropy is used recursively form a tree.

The Random forest implementation is used to classify the type of crime based on the other features of the dataset including location information (Longitude, Latitude, Ward, Community, etc.), Arrest information, Domestic Abuse, and FBI code. The target consists of 33 separate values.

K cross validation is used with a varying size of k to confirm that the model is not overfitting and the resulting accuracies are listed in the table below.

# of folds	Accuracy
3	0.9657
4	0.9677
5	0.9695
6	0.9700
7	0.9703
8	0.9705
9	0.9704
10	0.9707
11	0.9706
12	0.9706
13	0.9707
14	0.9709

Figure 15: Accuracies of random forest

	precision	recall	f1-score	support
0	0.98	0.92	0.95	414
1	0.99	1.00	0.99	17922
2	1.00	1.00	1.00	51803
3	1.00	1.00	1.00	16326
4	0.38	0.13	0.19	23
5	0.89	0.91	0.90	1288
6	0.99	1.00	1.00	30439
7	0.70	0.71	0.70	7112
8	1.00	1.00	1.00	13647
9	0.98	0.98	0.98	449
10	0.99	0.88	0.93	538
11	0.00	0.00	0.00	6
12	0.43	0.46	0.44	1110
13	0.04	0.02	0.02	117
14	0.44	0.25	0.31	219
15	0.92	0.93	0.93	427
16	1.00	1.00	1.00	11960
17	0.98	0.98	0.98	26294
18	0.00	0.00	0.00	7
19	0.00	0.00	0.00	18
20	0.00	0.00	0.00	1
21	0.06	0.03	0.04	33
22	0.75	0.60	0.66	2111
23	0.00	0.00	0.00	8
24	0.82	0.87	0.84	17246
25	0.98	0.99	0.99	1517
26	0.00	0.00	0.00	8
27	0.69	0.58	0.63	2706
28	0.99	0.99	0.99	11243
29	0.82	0.77	0.80	891
30	0.21	0.03	0.05	160
31	1.00	1.00	1.00	64526
32	0.98	0.98	0.98	3352
accuracy			0.97	283921
macro avg	0.64	0.61	0.62	283921
weighted avg	0.97	0.97	0.97	283921

Figure 16: Classifier precision report for random forest classifier

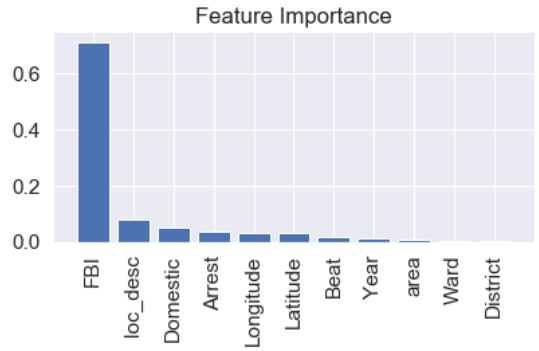


Figure 17: Feature importance in the random forest implementation

Due to the high dependence of the result on the FBI code being provided, a separate model is studied without the FBI code being present.

The results showed that the new classifier accuracy fell to 14% when trying to classify the crime as one of 33 distinct types of crime. To improve the accuracy of the classifier, the groups created by k means are used as buckets of crime type instead. The new labels are evaluated by taking the mode of the cluster label within each crime type. For eg. If theft has 100 datapoints labeled as cluster 1 and 10 datapoints labelled as cluster 2, it is assumed that theft belongs to cluster 1. This re-evaluated model is used to train a Random Forest classifier and the results showed an increase in performance to 57% accuracy. The model was also executed with 31 trees but the model accuracy went up to only 58%. The experiments also repeated the process with Max-Min Scaler in use without much change in accuracy.

primary_type	cluster	
0	1	647
1	1	15725
2	2	71905
3	8	18549
4	5	36
5	1	1120
6	6	40889
7	5	14155
8	6	18361
9	3	1102
10	1	700
11	1	7
12	5	3000
13	6	133
14	6	248
15	3	940
16	6	21123
17	3	66725
18	1	22
19	6	25
20	2	3
21	5	62
22	2	3037
23	3	14
24	2	14408
25	3	4311
26	5	33
27	5	5520
28	1	16115
29	6	839
30	2	196
31	6	94949
32	5	8588

Figure 18: new target buckets being evaluated using k means

Classifier	Accuracy
Naïve Bayes (Arrests)	84 %
Radom Forrest (Arrests)	86 %
Random Forest (1 tree)	93 %
Random Forest (2 tree)	91 %
Random Forest (15 trees)	97%
Random Forest (15 trees) without FBI	38 %
Random Forest w/o FBI w/ (with cluster targets)	73 % (15 trees)

VI. CONCLUSION

Naïve Bayes is a good classifier for predicting if someone is like to get arrested based on location and the type of crime being committed. Random Forest is performed for the same target variable and is found to be marginally better.

Random Forest has good accuracy to predict the type of crime based on the input vector consisting of all features in the dataset except Primary Type. However, the prediction was found to be very dependent on the FBI code and a study of the model minus the FBI code produced a model of only 38% accuracy. In case where Primary Types were clustered together, the Random Forest algorithm was able to predict the correct cluster at 73% accuracy. From the results the authors provide an acceptable baseline model for predicting the type of crime.

VII. FUTURE WORK

There is scope for expansion on 2 fronts. Firstly, a further study should be carried out considering the timestamp in the dataset. Predictions might improve with the month or day of the week taken into consideration.

Secondly, an improvement in the prediction might be realized if a Neural Network like an LSTM were applied to the model. A comparative study could be useful.

VIII. REFERENCES

[1] Chicago Crimes Dataset. (n.d.). Retrieved December 8, 2019, from https://www.kaggle.com/currie32/crimes-in-chicago/downloads/Chicago_Crimes_2012_to_2017.csv/1.h

[2] (n.d.). Retrieved from http://www.cs.cmu.edu/afs/cs/project/jair/pub/volume16/chawla02a-html/node2.html.

[3] Sklearn User Guide. (n.d.). Retrieved from https://scikit-learn.org/stable/user_guide.html.

[4] Raschka, Sebastian, and Vahid Mirjalili. *Python machine learning*. Packt Publishing Ltd, 2017.

[5] Friedman, Jerome, Trevor Hastie, and Robert Tibshirani. *The elements of statistical learning*. Vol. 1. No. 10. New York: Springer series in statistics, 2001.

[6] color example code: colormaps_reference.py. (n.d.). Retrieved from https://matplotlib.org/examples/color/colormaps_reference.html.