Basic Statistical Analysis and data cleaning

Since the datasets were different in terms of how information is stored in each dataset, we used different binning strategies. For the Montgomery County crime dataset, we used binning on the feature called “class”. In the dataset, class is represented in number and each number refers to a type of crime (class description).

For example, “0316” in class means robbery by firearm at financial institution and “0315” means robbery by firearm at residential place. The difference between 0316 and 0315 is a location where robbery by firearm occurred. If a class number has a different hundreds digit, then a type of crime differs. For instance, “0411” means aggravated assault using firearm at citizen.

Noticing this pattern in class, we used bins that separate class numbers by 100. The benefit of this bin is that we can immediately know that any class number that contains 03 as the first digits means robbery by firearm. Using this binning strategy, we can differentiate class types effectively and efficiently.

For the Chicago crime dataset, we used binning on the feature called “time”. In this case, time refers to a time when a crime happened in Chicago. First, due to the fact that time was in a string format, we turned time into a numerical value by using our definition of hour. Afterwards, we binned the times by every 2 hour. For example, if a crime happens in 5:00 pm, then the crime gets placed in the bin of 4 and 6, (4,6]. The benefit of this binning strategy is that we can make times clean and easier to understand when the crimes happened.

For missing values in the two datasets, we handled differently. For the Montgomery County crime dataset, we have filled missing values with 0s. For example, there were lots of missing values in these two particular attributes in the dataset we wanted to use, latitude and longitude. We were interested in the attributes because we wanted to use them for the clustering part of the project. As we looked through the attributes, we noticed an interesting pattern; if a latitude is missing in one row, a longitude is also missing in the row but if a latitude is not missing in a row, a longitude is also not missing in the row.

Knowing the pattern, we figured out that filling the missing values in the attributes with two zeros would not have any consequences regarding the clustering. Also, filling the missing values with two zeros will maintain the same number of rows for each attribute, which makes it easier for us to handle the data. It was the same for the Chicago crime dataset; we left the attributes with missing values as they were if we did not use in our analysis. For the attributes we used in our analysis, we have dropped the missing values not to affect our results in an unintended way.

Histograms

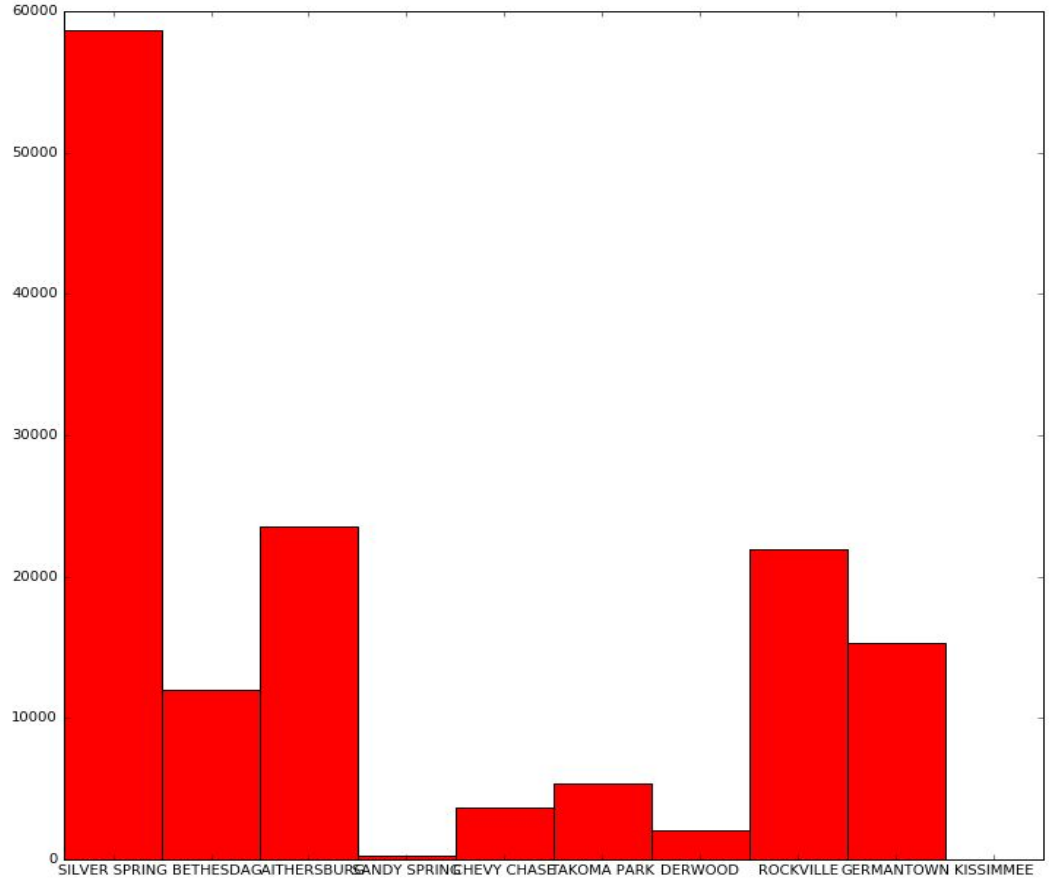
We have generated a total of four histograms, one from the Montgomery County crime dataset and three from the Chicago crime dataset. The histograms provided so much information about our datasets. The histogram from the Montgomery County crime dataset is built on the feature called “city”. The city represents a place where a crime happened in the Montgomery County. However, the traditional dataframe.hist() did not work when we first tried. To compensate this, we have found the unique rows of the column and frequency corresponding to each unique row. Then, we created the histogram using the information. According to the histogram, the city where the most of crimes happened in the Montgomery County is Silver Spring. The city where crimes happened the second most is Gaithersburg.

The first histogram from the Chicago crime dataset was built on the feature “primary type”. In this dataset, primary type refers to a type of a crime occurred in Chicago. We have used the same method to generate a histogram from primary type. We have obtained a very interesting fact about a common type of crimes occurred in Chicago from the histogram. The common type of crimes in Chicago is theft. The next common type followed by theft is battery. The second histogram from the dataset was based on “dayofWeek”. The “dayofWeek” represents a day when crimes occurred on. Using the same method, we have produced a histogram.

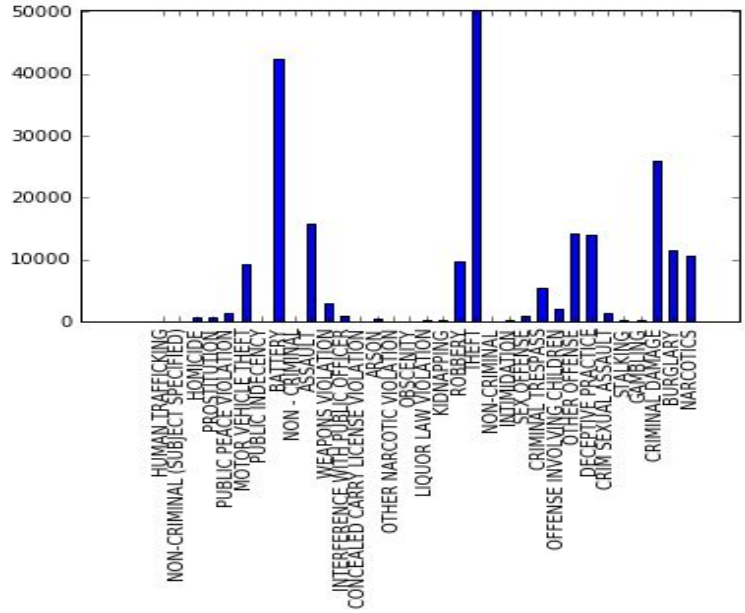
From this histogram, we realized that there is no a particular day when the majority of crimes happened. However, overall, there were more crimes happened on Friday than the other days in week. The third histogram from the dataset is generated based on the time period of crimes occurred. We have a total of twelve time periods of two hour starting from 0:00 AM. So, if a crime happened in 1:00AM, the crime belongs to 0:00AM to 2:00 AM time period. Using the same method, we created a histogram. According to the histogram, overall, lots of crimes occurred between 18:00 and 20:00 followed by 16:00 and 18:00.

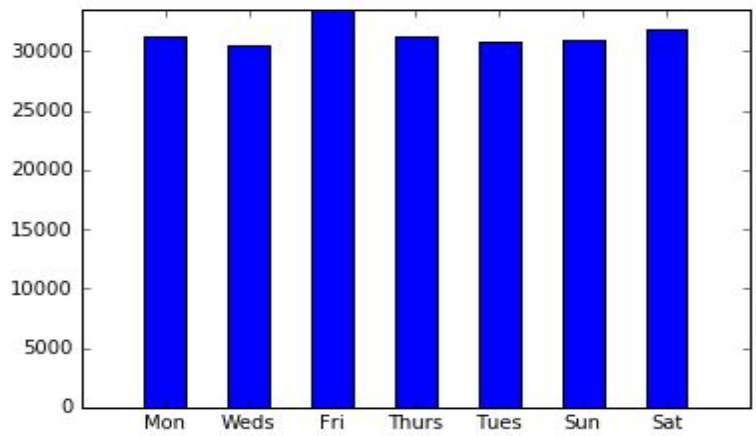
For the scatterplot, because most of the data in our data are all categorical, we decided to chose the numerical attributes “time_hour”, “weekdayNum” and “community_area” as our scatterplot’s attributes. From the graph of community_area and time_hour we can see that nearly all of the graph are cover by dark blue and there is a few white dots in the graph. The white dots mean in that time period there is no crime occur which points out to be community 11, 47 and 76 in the morning. The other graph represents crime occur on Mon. to Sun. in each time and the community areas. Because of the huge data, the graph shows nearly as a straight line. Basically, there is no correlation among the three attributes.

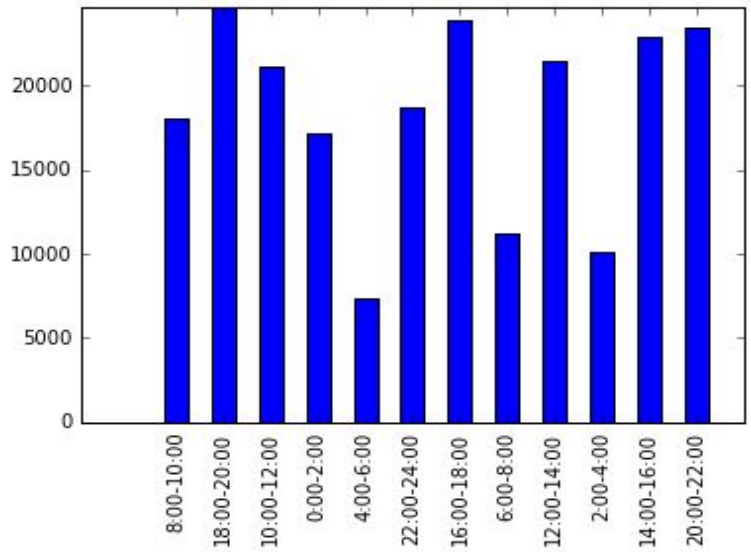
Montgomery histogram(There is a total of five histograms and I am including only one that contains the most amount of data):



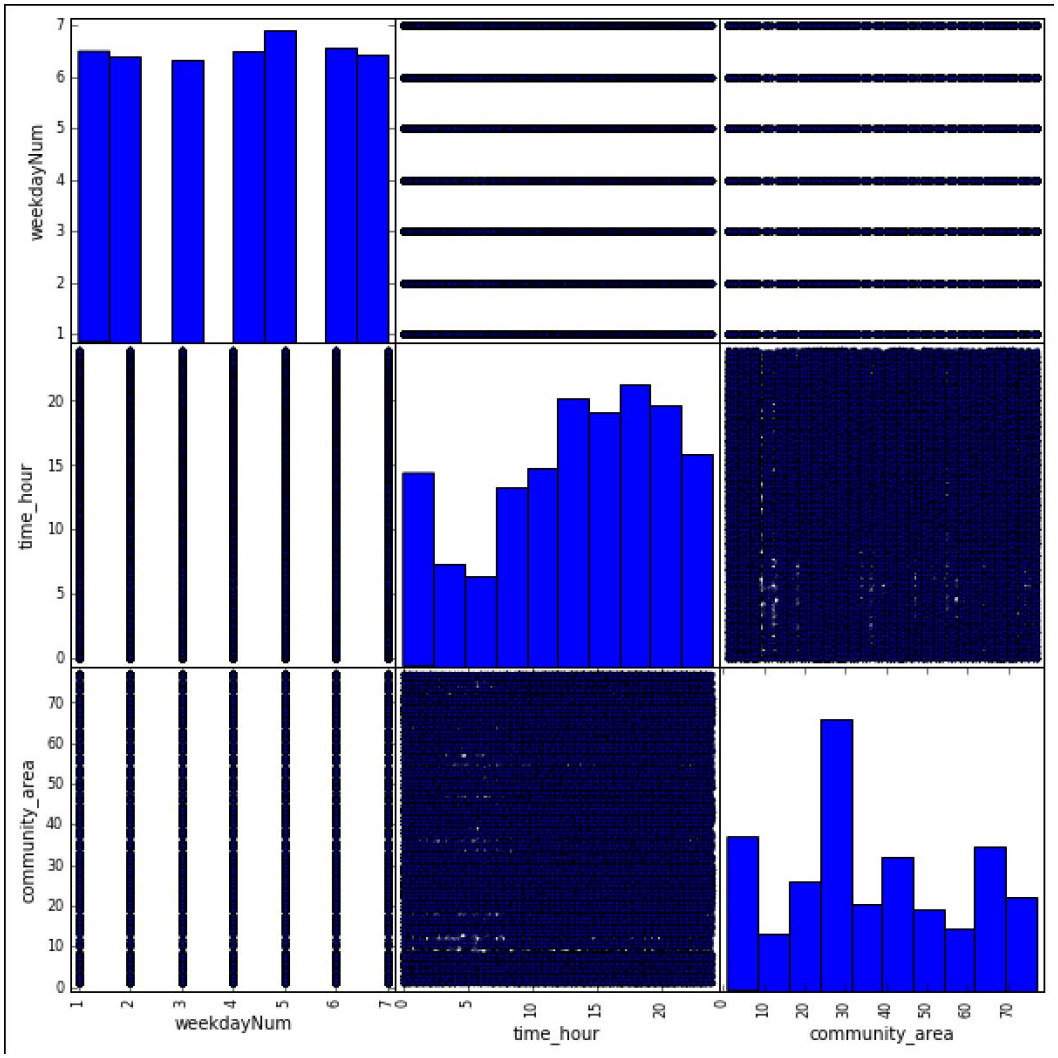
Chicago histograms:







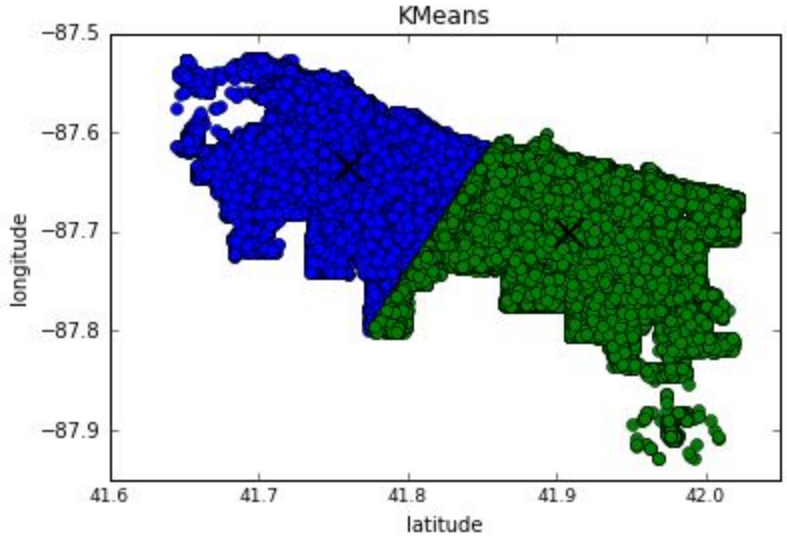
Scatterplot:



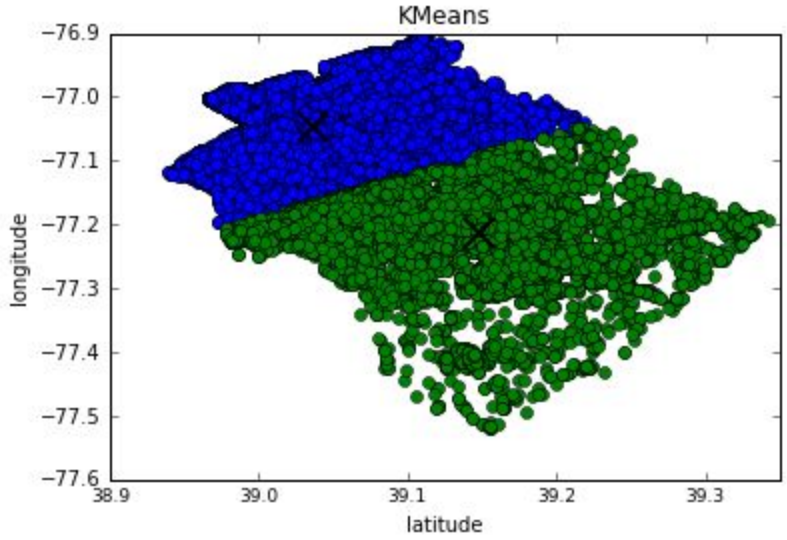
Cluster Analysis

For the cluster analysis part of the project, we had quite a trouble because we did not have numeric values to data points in the two datasets. At the end, we figured we can use latitude and longitude for the both datasets because when latitude and longitude for each data point in the entire datasets were plotted in K-Means, the plots looked like the city of Chicago and the map of the Montgomery County.

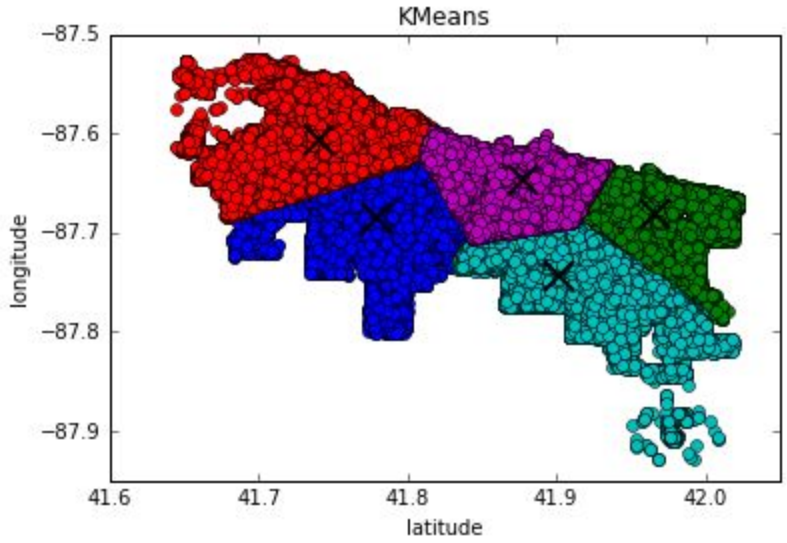
When k=2, Chicago:



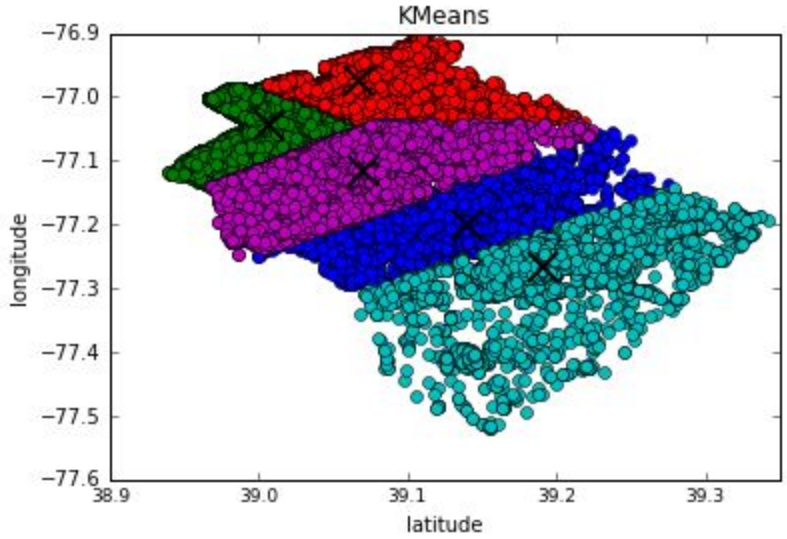
The Montgomery County:



When k = 5, we can see different clusters in each plot.
Chicago:



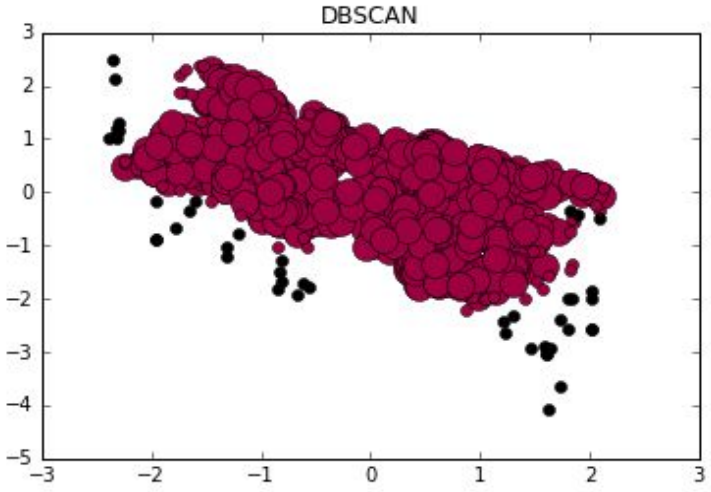
The Montgomery County:



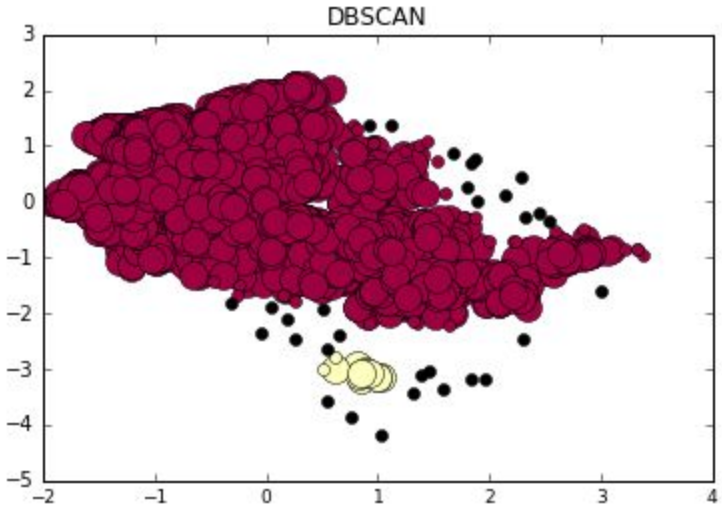
Particularly, we observed different densities of clusters. For the Montgomery crime dataset, the green cluster is more dense than the red cluster, which we can deduce that lots of crimes happened not far away from the centroid of the green cluster. We can also say the same thing for the teal cluster by observing the plot; crimes were dispersed and spread out in the teal cluster.

The DBSCAN plots for the both datasets show the same results; the results look like the city of Chicago and the Montgomery County.

Chicago:



Montgomery County:



Black dots represent noises in the datasets but in our datasets, we did not have noises. So, we have another way of interpreting these results. For the Chicago crime dataset, we can say that the majority of crimes happened in where the big red cluster is but some crimes happened outside the clusters. Those black dots can be considered as unusual crime locations.

For the Montgomery County crime dataset, the majority of crimes happened in the big red cluster but there is another yellow cluster where crimes were heavily concentrated in terms of location. The black dots can be considered as unusual crime locations. For Ward clustering using latitude and longitude, we could not find any specific interpretation regarding the graphs.

Association Rules

For the association rules, we use the package orange for running Apriori algorithm for the dataset in order to generate the frequency item.

First we generated the csv into a table, and from the chosen attributes from both dataset(“places”, “city”, ”incident_type”) we first build a dictionary and convert into numeric data and started to run for the algorithm. After that we convert the numeric data back into the original categorical data for reader to read easily.

We generate the top 5 Association Rules in both dataset.

```
----------------------total rules num: 243-------------------------

['2812', 'Street - In vehicle', 'SILVER SPRING']:2723
['Street - In vehicle', '1834', 'SILVER SPRING']:2186
['2812', 'Street - In vehicle', 'GAITHERSBURG']:1521
['ROCKVILLE', '2812', 'Street - In vehicle']:1182
['2812', 'Street - In vehicle', 'GERMANTOWN']:844
```

The Montgomery Crime data the most frequently one is ‘2812’, ‘Street – In vehicle’, ‘Silver Spring’ which 2812 represents “DRIVING UNDER THE INFLUENCE”. The support value for the association rule is 2723.

The outcome does not surprised us, because Silver Spring, 2812 and Street – In vehicle they were already the highest frequent one from our previous statistic analytic. According to the rules, police officer can expect the crime is going to happen in those districts, and prepare the equipment that can fit to fight those particular crime.

```
----------------------total rules num: 357-------------------------

['25', 'THEFT', 'Sat']:1124
['25', 'BATTERY', 'Fri']:991
['25', 'THEFT', 'Thurs']:783
['28', 'BATTERY', 'Sat']:755
['Sun', '67', 'THEFT']:661
>>>
```

The Chicago Crime data the most frequently one is ‘25, ‘Theft’, ‘Sat’ which 25 represents “Community 25 “in Chicago which is Austin. The support value for the association rule 1124.

The outcome does not surprised us, because 25, Theft, Sat they were already the highest frequent one from our previous statistic analytic. From the top 5 association rules, we can increase the patrol frequency within those days in the communities.

Hypothesis Testing

1st hypothesis:

“A theft will most likely happen in community 23 from 18:00 to 20:00 on Friday in Chicago”

Logistical Regression:
First, we tried linear regression on the attributes we chose but there is no linear relationship among the attributes. Therefore, we decided to use logistical regression. First, we created a data frame with community area, weekday number, an intercept column of theft and dummy variable for time_section. Then, we instantiated X and y where X is equal to community area, weekday number and dummy variable and y is an intercept column. When we ran the regression, we got the probability of “a theft will most likely happen in community 23 from 18:00 to 20:00 on Friday in Chicago” is equal to .27365, which is very low. So, this method does not support our hypothesis.

Decision Tree:
Decision Tree takes as two input arrays: an array X where X is equal to community area, weekday number, and dummy variable and an array y holds the class label = theft. We used X and y to build decision tree model and tested the accuracy on the given test data and label. We found the accuracy of the model equal to .77227, which means the model is accurate. Then, we used this model to test the hypothesis. The probability of “A theft will most likely happen in community 23 from 18:00 to 20:00 on Friday in Chicago” is .17544, which is very low. So this method does not support our hypothesis.

KNN:
We used the same training sample to build a KNN model. Also, we used the same test sample to find whether the model is accurate. The accuracy is equal to .72615, which means the model is pretty accurate. Using the model, we tested the hypothesis and the probability of the hypothesis being true is equal to 0, which means our hypothesis is not true.

Naive Bayes:
We used the same training sample to build a Naive Bayes model. Also, we used the same test sample to find whether the model is accurate. The accuracy is equal to .7709, which means the model is pretty accurate. Using the model, we tested the hypothesis and the probability of the hypothesis being true is equal to .27375. Since the probability is low, it means our hypothesis is not true.

Support Vector Machine:
For this method, SVM model takes a long time to produce our result. The model computes the distance between each pair of data points. For this particular Chicago crime dataset, we have more than 220,000 data points, so it takes a long time. We did not include the result here but the code is still in the program.

Random Forest:
The way we applied this method is the same to the previous methods. The accuracy of this model is .7683. The probability of the hypothesis being true is .18855. Since the probability is low, it means our hypothesis is not true.

2nd hypothesis:
(T-test can not be used in the 1st hypothesis so we come up with a new hypothesis which can be used in t-test)

“The time that crime most likely happens in Chicago is same as Montgomery.”

T-test:
We use the function “stats.ttest_ind()” from the package scipy of stats in python to test our hypothesis. Inputs of the function are “time_hour” of Chicago and “time_hour” of Montgomery. The function returns the T statistic and P-value. The p-value is 1.599e-08 which is very low. So, this method does not support our hypothesis.

3rd hypothesis:
“Driving under influence will most likely happen in Silver Spring from 16:00 to 18:00 on Tuesday in Montgomery”

Logistical Regression:
First, we created a data frame with cityNum, day_of_week, an intercept column of Driving under influence and dummy variable for time_section. Then, we instantiated X and y where X is equal to cityNum, day_of_week and dummy variable and y is an intercept column. When we ran the regression, we got the probability of “a Driving under influence will most likely happen in Silver Spring from 16:00 to 18:00 on Tuesday in Montgomery” is equal to .01037, which is very low. So, this method does not support our hypothesis.

4th hypothesis:
“The day of week that crime most likely happens in Chicago is same as Montgomery”

T-test:
We use the function “stats.ttest_ind()” from the package scipy of stats in python to test our hypothesis. Inputs of the function are “day_of_week” of Chicago and “day_of_week” of Montgomery. The function returns the T statistic and P-value. The p-value is 8.0979e-192 which is very low. So, this method does not support our hypothesis.

Overall Story

Crime is still a big issue whether it is in big cities or counties. Police officers work to decrease illegal activities and it is important for them to manage crime by some sorts of prediction. Crime has patterns just like everything else. Through this project, we plan to investigate crime patterns of the city of Chicago and the Montgomery County of Maryland. Using historical crime data, we can predict where and when a next crime will likely take a place. This prediction will help police officers to implement prevention work. It will significantly reduce the number of incidents of crime and reduce the harm caused by crime.

From our datasets, we have found numerous suggestions to the city government of Chicago and the government of the Montgomery County of Maryland. A very common type of crimes in Chicago is theft followed by battery throughout the city of Chicago. From this fact, Chicago Police Department can train police officers to handle theft in an efficient way. Based on the Frequent Itemset Mining algorithm, we found out that the day thefts happened on the most is Saturday. The CPD can increase the number of police force on Saturday to ensure the safety of innocent civilians. According to one of the histograms, the most frequent time period in which crimes occurred is 6:00 PM to 8:00 PM. To sum it up, the CPD can increase the number of police force who can patrol around the areas where the most thefts occurred between 6:00 PM to 8:00 PM on Saturday. This will significantly reduce the number of crimes throughout the city of Chicago.

The Montgomery County historical crime dataset shows a quite different pattern from the Chicago crime dataset. A common type of crimes in the Montgomery county is the class number “2812”, which means driving under the influence. To tackle this crime behavior, the Montgomery County Police Department increase the number of police force specifically targeting drunk drivers around the areas where bars are heavily populated. The histogram generated based on the cities where crimes occurred the most in the Montgomery County states that Silver Spring is the most dangerous area. An interesting fact is that the Frequent Itemset Mining algorithm tells us that DUI incidents happened the most in the location “in street - in vehicle” in Silver Spring. To ensure the safety of non-drunk drivers in the Silver Spring area, the Montgomery County Police Department can heavily focus on arresting drivers under the influence, which can significantly reduce the number of car accidents and save many people’s lives on road.

From running analyses on the two historical crime datasets, we have found the hotspots of crime in the two areas. We have identified the common types of crimes occurred in the both areas. The city of Chicago and the Montgomery County should increase the number of police forces specifically targeting the common types in places where the crimes mostly occurred. From these insights, we can find the crime patterns, reduce the crime rates, and prevent crimes from happening at the first place.