Project Report

CRIME ASSOCIATION RULES MINING IN ATLANTA

Team Number: 8

Team Members:

Name: Urvi Patel	CUID: C43960061
Name: Changlong Li	CUID: C49064171
Name: Mohammed Abdul Najeeb Farooqui	CUID: C28040948

ABSTRACT

Crime is an inevitable problem in all major cities of the world. In the United States, in particular, cities like Chicago and New York have some of the highest rates of crime. Atlanta, one of the largest cities in the Southeastern U.S. and the largest metropolitan area closest to us, is no exception in the occurrence of crime. Like most other major cities, it experiences a wide range of crimes on a daily basis, from minor crimes like shoplifting and pickpocketing, to major crimes like rape, murder and narcotics trafficking.

In the past, due to inadequate data collection and surveillance techniques, it was difficult to recognize a pattern to these daily crimes, much less apprehend the criminals behind these violations. Fortunately, we now have access to data that is produced every second of every day as well as systematic recording and evidence collection methods. The devices used to collect this rich source of data take the form of public cameras, personal cellphones, and various electronic devices found in police vehicles.

Aggregating all of the data collected from these many devices would facilitate a more accurate pattern analysis of all types of crime, enabling the police to predict and prevent crimes that, all too often, adversely affect the lives of Atlanta's citizenry. Additional analysis of the data has the potential to almost immediately impact the fight against crime that plagues the city. In addition, this data can help the city government of Atlanta better understand the daily living, working, and recreational environments of the local population, and consequently better serve the city's interests.

Table Of Content

CHAPTER 1

INTRODUCTION

1.1 PROJECT SUMMARY

The project is “CRIME ASSOCIATION RULES LEARNING IN CITY OF ATLANTA”. The Projects main aim was to be able to analyze the crimes happening all around in Atlanta and be able to form an association as to where and when certain types of crimes normally happen.

1.2 PURPOSE

The project can form association rules between the time of the crime reported, the location of the crime and the type of crime that happened. This enables us to discover a pattern of the crime and may help users take precaution measures to avoid it.

Some of the crimes that seem common are in fact something the users could have avoided, if the patterns of the crimes are studied. The criminals usually tend to follow a certain pattern when doing a crime, like the locations where the crime happened or the time during which the crimes happens. Studying and analyzing the patterns can help the citizens of that city be a step ahead and try preventing it from happening.

1.3 OBJECTIVE

The objective of this project is to be able to generate visualizations and facts by studying the patterns of all the crimes happening in the city and be able to see any pattern that a certain crime usually follows. The visualizations would be a way of projecting the results from the algorithms used to obtain them from.

1.4 TOOLS AND TECHNOLOGY

RStudio

RStudio is a free and open source integrated development environment (IDE) for R.

R is a programming language and free software environment for statistical computing and graphics. The R language is widely used among statisticians and data miners for developing statistical software and data analysis.

There are several reasons of choosing R over the other languages. First being that, R is free, open source code. It is very stable and reliable. It runs anywhere, be it Windows, Unix systems or Mac. R performs a wide variety of functions, such as data manipulation, statistical modeling, and graphics.

Tableau

For the user to easily understand the results obtained from the data, the team saw it better to provide an interactive visualization where the user can be able to see the visualization as they wish, see the results for the specific location that they wish. we chose tableau for our rendering because:

- It converts unstructured Statistical information into comprehensive logical results, which are fully functional, interactive and provides appealing dashboards.
- It is easy to use and understand.
- The software supports establishing connections with many data sources, which improves the data analytics quality and facilitates the creation of a more unified and informative dashboard.
- It can handle large amounts of data.

CHAPTER 2

PROJECT MANAGEMENT

2.1PROJECT PLANNING

2.1.1 Project Development Approach and Justification

The project development approach used for this project is the Iterative Waterfall Model.

Iterative process starts with a simple implementation of a subset of the software requirement and iteratively enhances the evoking versions until the full system is implemented. At each iteration, design modifications are made and new functional capabilities are added.

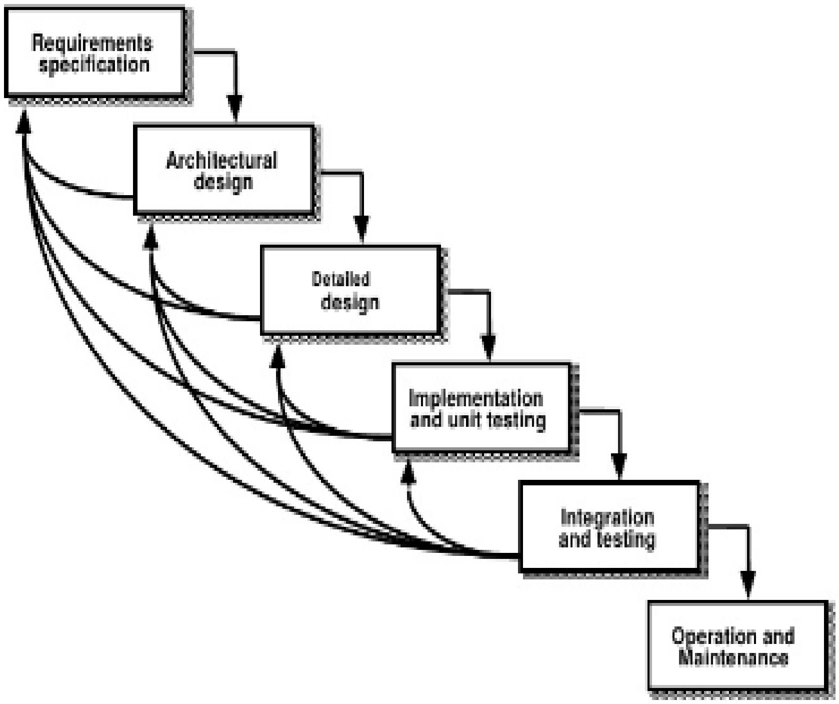

Requirements specification
Architectural design
Detailed design
Implementation and unit testing
Integration and testing
Operation and Maintenance

Fig 2.1 Iterative Waterfall Model design

Advantages:

- In iterative model we can only create a high-level design of the application before we actually begin to build the product and define the design solution for the entire product. Later on we can design and build a skeleton version of that, and then evolve the design based on what had been built.
- In iterative model we are building and improving the product step by step. Hence we can track the defects at early stages. This avoids the downward flow of the defects.
- In iterative model we can get the reliable user feedback. When presenting sketches and blueprints of the product to users for their feedback, we are effectively asking them to imagine how the product will work.
- In iterative model less time is spent on document and more time is given for designing.

JUSTIFICATION:

We used the iterative waterfall model since the requirements of the complete system are clearly defined and understood. When the project is big, the iterative waterfall model is best used. We had the major requirements defined however some details had evolved with time.

2.1.2 Roles and Responsibilities

The project comprises of three members:

1. Urvi Patel
2. Changlong Li
3. Mohammed Abdul Najeeb Farooqui

Pair programming is an agile software development technique in which programmers work together at one workstation. One, the **driver**, writes code while the other, the **observer**, **pointer** or **navigator**, reviews each line of code as it is typed in. The programmers switch roles frequently.

Roles	**Responsibilities**
Urvi Patel	Req. gathering, Coding, Maintenance
Changlong Li	Req. Analysis, Coding, Testing
Mohammed Abdul Najeeb Farooqui	Req. Analysis, Coding, Testing

Table 2.1.1 Group Dependencies

CHAPTER 3

SYSTEM REQUIREMENT STUDY

3.1 User Characteristics

The target audience for the project is the people who may use this study to be able to gain analyzed information to be able to take precautions.

3.2 Software Requirements

The ideal or the recommended configurations are listed as under:

Software Specifications:-

- Front End: R
- Environment: windows, unix systems or Mac
- Tools: RStudio and Tableau

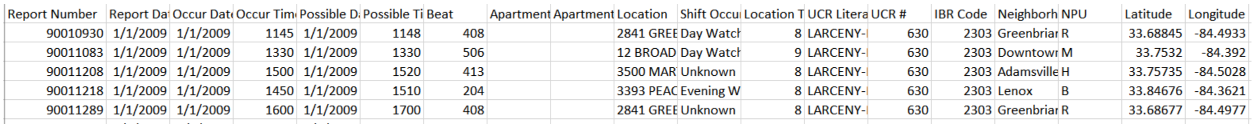
3.3 Assumption and Dependencies

- The user understands how to use the interactive visualization to gain information they specifically want.

CHAPTER 4

SYSTEM DESIGN

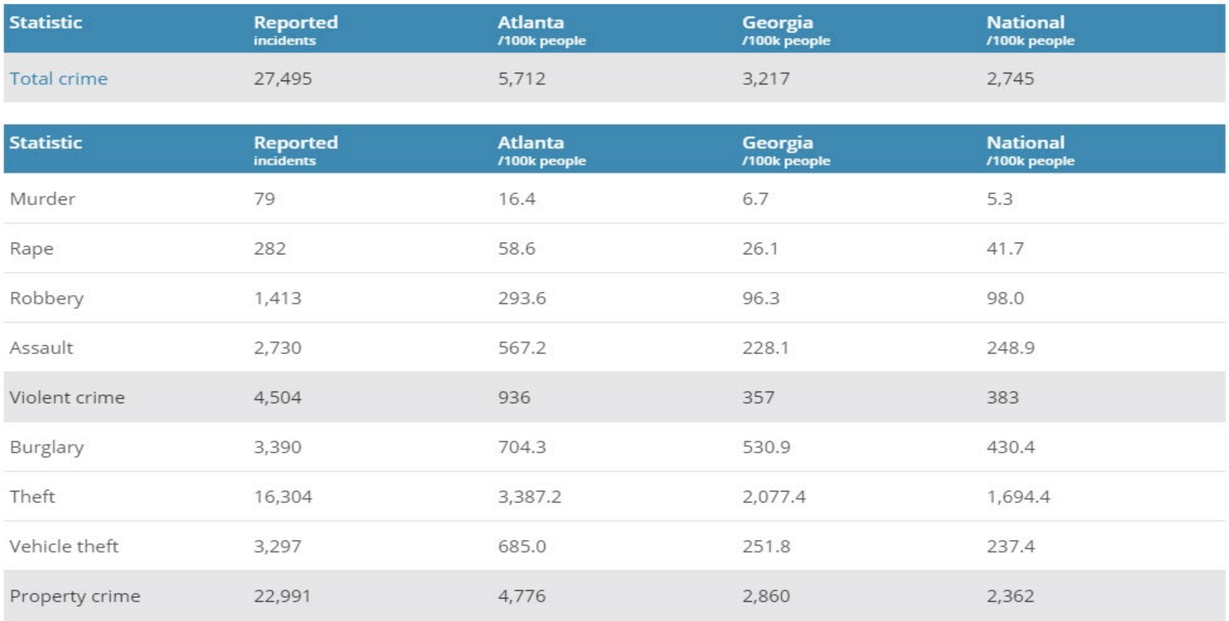
Dataset



Report Number	Report Dat	Occur Date	Occur Tim	Possible Da	Possible Ti	Beat	Apartment	Apartment	Location	Shift Occu	Location T	UCR Litera	UCR #	IBR Code	Neighborh	NPU	Latitude	Longitude
90010930	1/1/2009	1/1/2009	1145	1/1/2009	1148	408			2841 GREE	Day Watch	8	LARCENY-	630	2303	Greenbriar	R	33.68845	-84.4933
90011083	1/1/2009	1/1/2009	1330	1/1/2009	1330	506			12 BROAD	Day Watch	9	LARCENY-	630	2303	Downtown	M	33.7532	-84.392
90011208	1/1/2009	1/1/2009	1500	1/1/2009	1520	413			3500 MAR	Unknown	8	LARCENY-	630	2303	Adamsville	H	33.75735	-84.5028
90011218	1/1/2009	1/1/2009	1450	1/1/2009	1510	204			3393 PEAC	Evening W	8	LARCENY-	630	2303	Lenox	B	33.84676	-84.3621
90011289	1/1/2009	1/1/2009	1600	1/1/2009	1700	408			2841 GREE	Unknown	8	LARCENY-	630	2303	Greenbriar	R	33.68677	-84.4977

Crime statistics

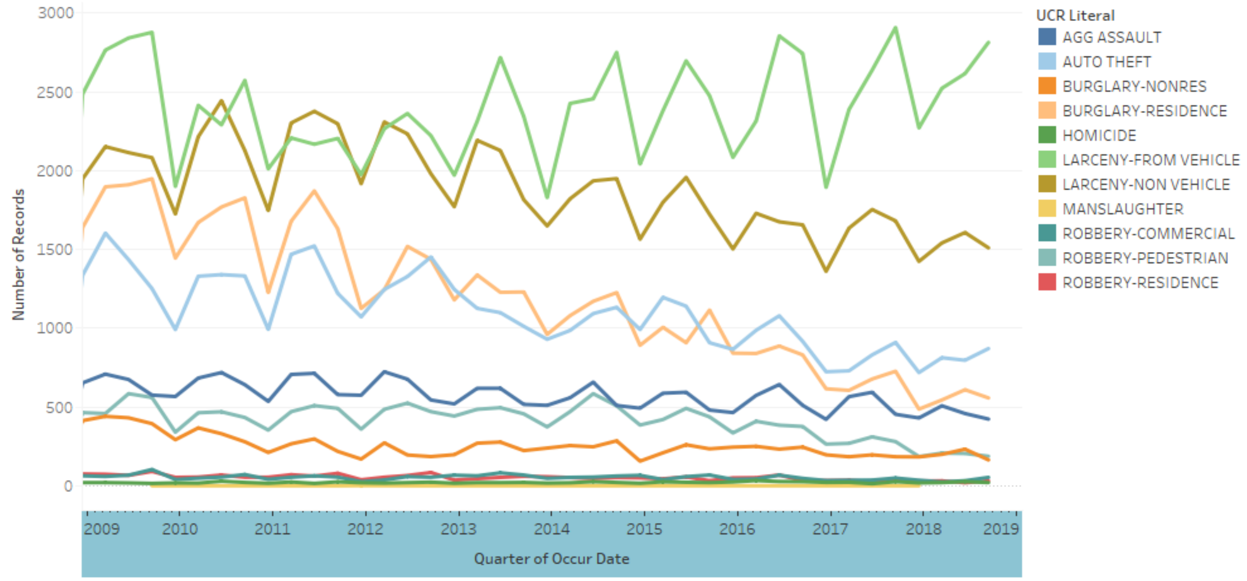
To be able to come up to a decision of what attributes to use for generating association rules, we first did statistical analysis of the data based on what crimes were most prevalent, which time they were prevalent, what location was most affected by them and what day and month was it most frequent during.



Statistic	Reported incidents	Atlanta /100k people	Georgia /100k people	National /100k people
Total crime	27,495	5,712	3,217	2,745

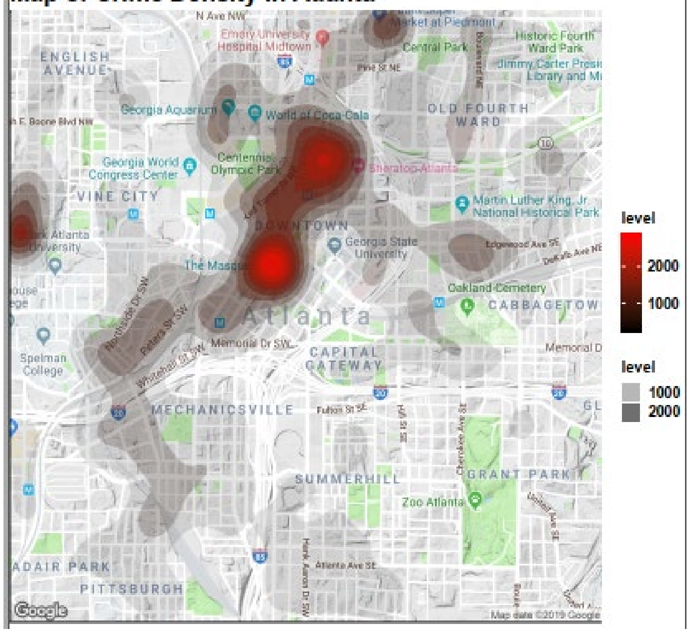
Statistic	Reported incidents	Atlanta /100k people	Georgia /100k people	National /100k people
Murder	79	16.4	6.7	5.3
Rape	282	58.6	26.1	41.7
Robbery	1,413	293.6	96.3	98.0
Assault	2,730	567.2	228.1	248.9
Violent crime	4,504	936	357	383
Burglary	3,390	704.3	530.9	430.4
Theft	16,304	3,387.2	2,077.4	1,694.4
Vehicle theft	3,297	685.0	251.8	237.4
Property crime	22,991	4,776	2,860	2,362

<Time Series plot of Altlanta crimes (2009-2018)>

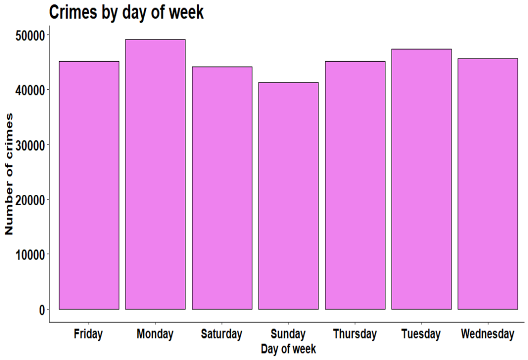


The trend of sum of Number of Records for Occur Date Quarter. Color shows details about UCR Literal.

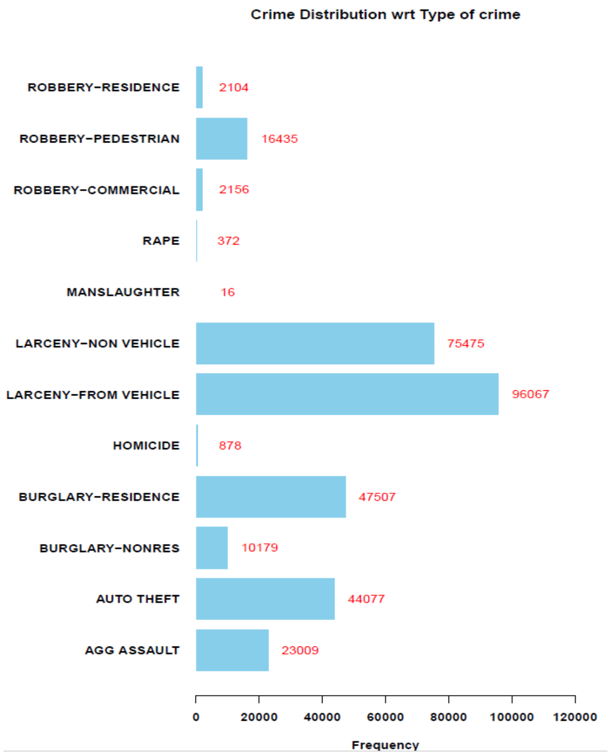
Map of Crime Density in Atlanta

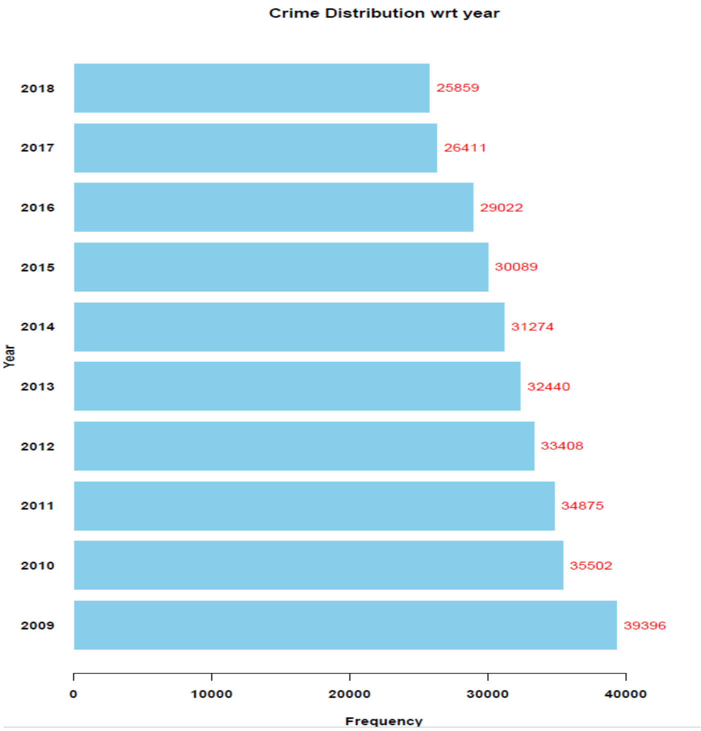


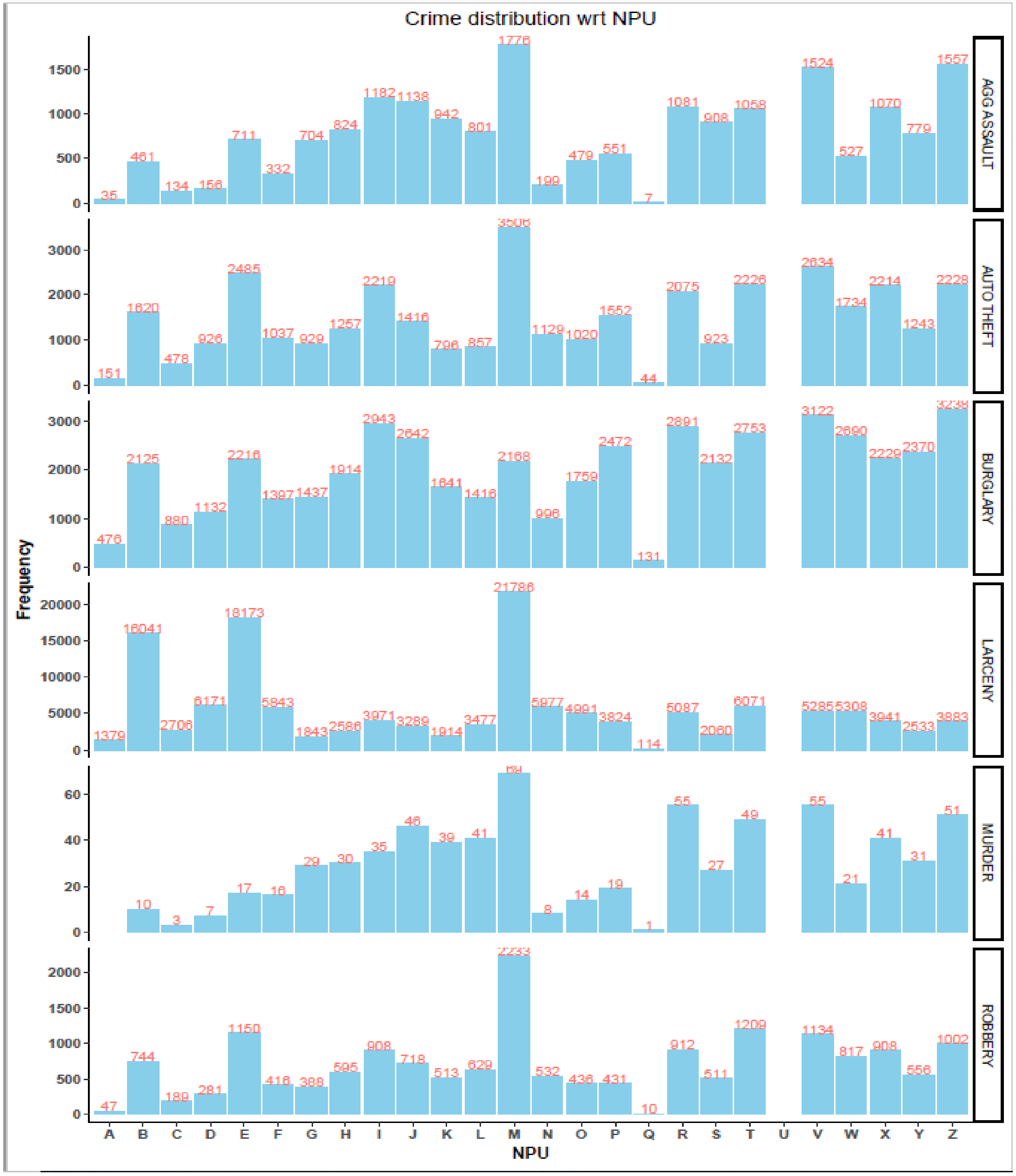
Crimes by day of week

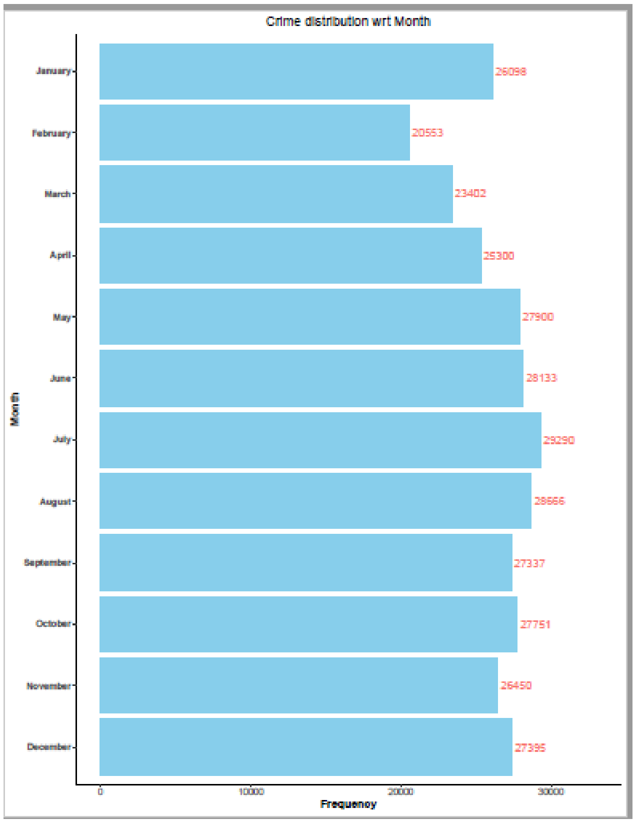


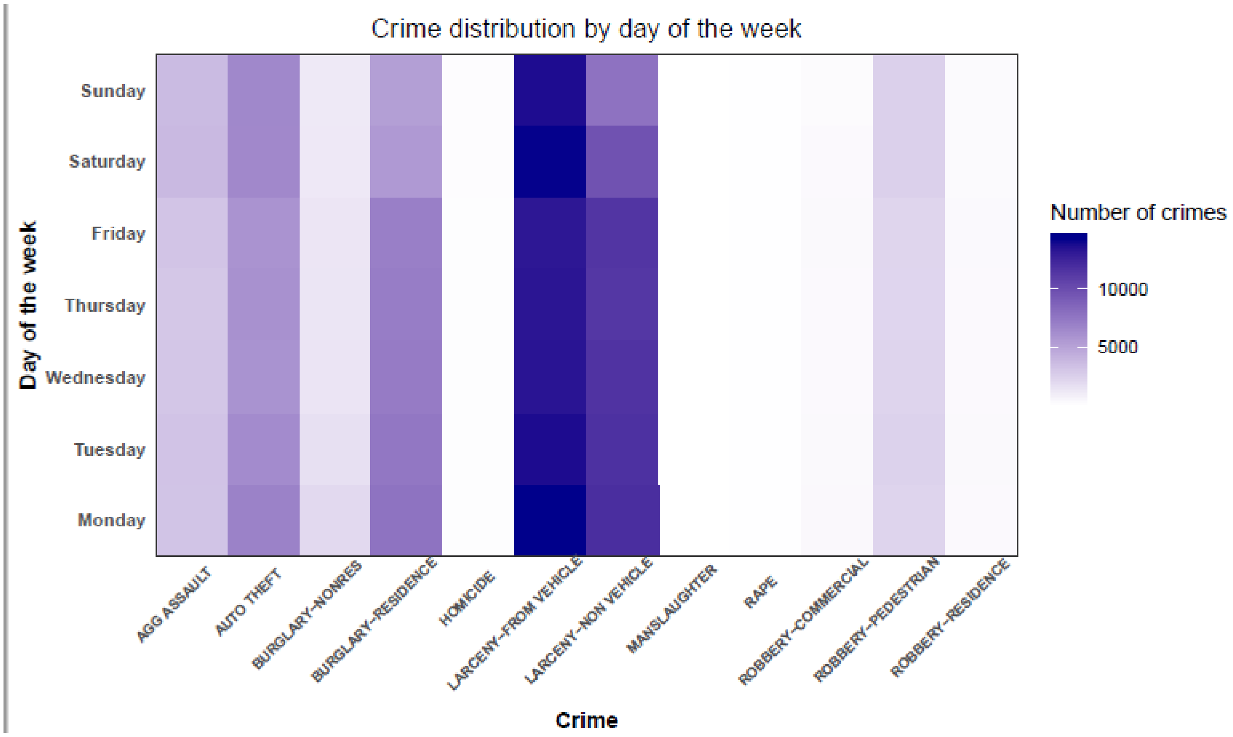
The crime density map above was able to tell us that the Downtown of Atlanta has more crimes happening than other parts of Atlanta. That helps us narrow down that some NPUs(divisions) have higher crime rates than others.

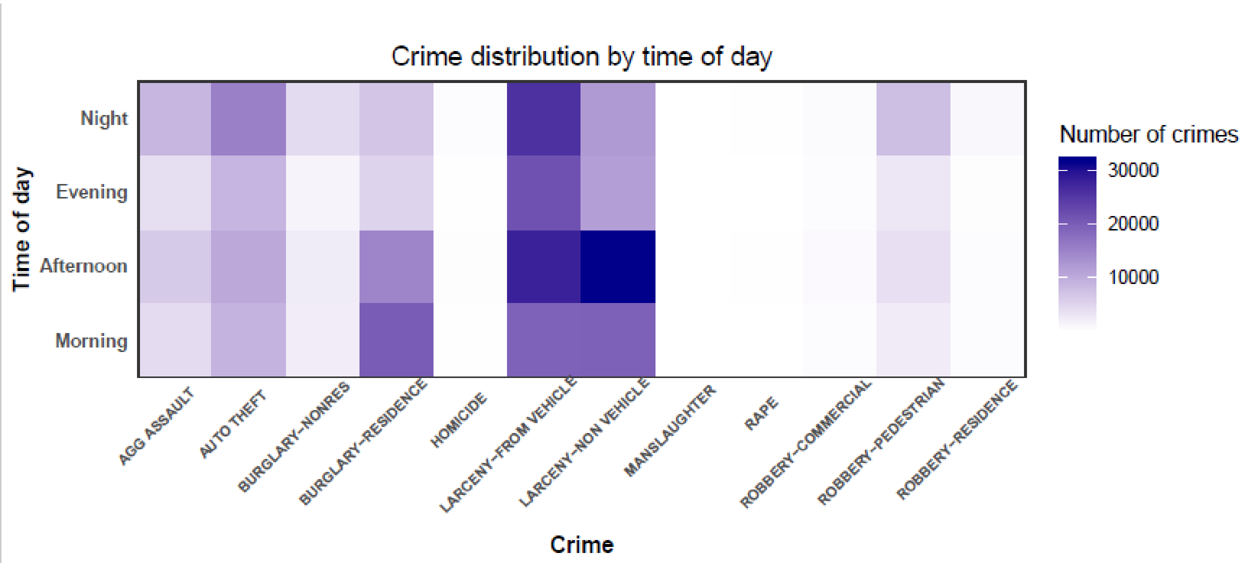

Crime Distribution wrt Type of crime
ROBBERY−RESIDENCE
2104
ROBBERY−PEDESTRIAN
16435
ROBBERY−COMMERCIAL
2156
RAPE
372
MANSLAUGHTER
16
LARCENY−NON VEHICLE
75475
LARCENY−FROM VEHICLE
96067
HOMICIDE
878
BURGLARY−RESIDENCE
47507
BURGLARY−NONRES
10179
AUTO THEFT
44077
AGG ASSAULT
23009
0
20000
40000
60000
80000
100000
120000
Frequency


Crime Distribution wrt year
Year
2018
25859
2017
26411
2016
29022
2015
30089
2014
31274
2013
32440
2012
33408
2011
34875
2010
35502
2009
39396
0
10000
20000
30000
40000
Frequency

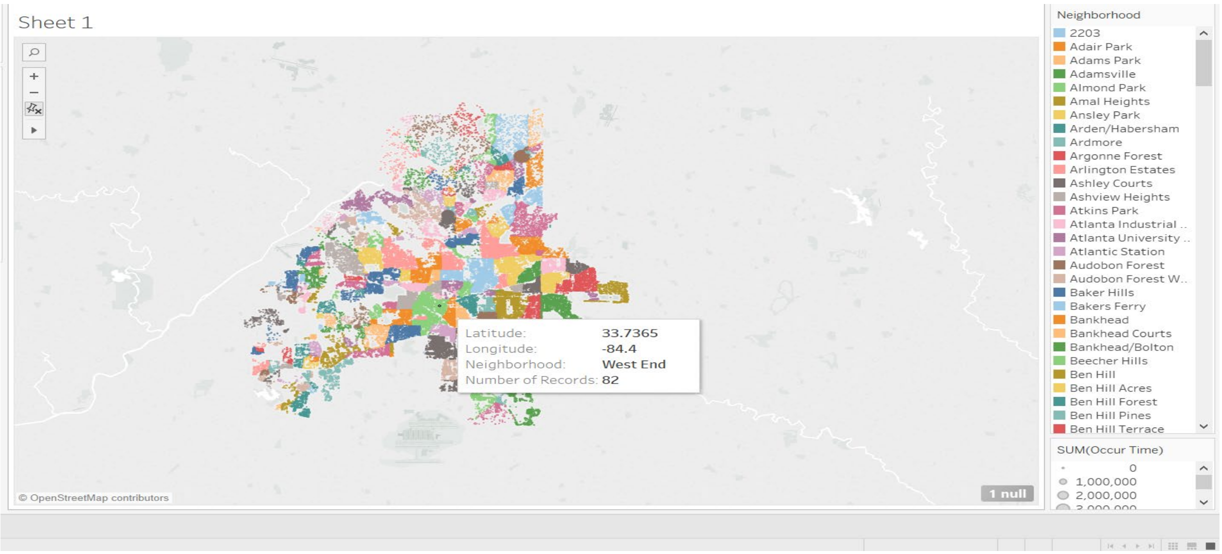

Crime distribution wrt NPU
Frequency
NPU
A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
AGG ASSAULT
0 500 1000 1500
35 461 134 156 711 332 704 824 1182 1138 942 801 1776 199 479 551 7 1081 908 1058 1524 527 1070 779 1557
AUTO THEFT
0 1000 2000 3000
151 1620 478 926 2485 1037 929 1257 2219 1416 796 857 3506 1129 1020 1552 44 2075 923 2226 2634 1734 2214 1243 2228
BURGLARY
0 1000 2000 3000
476 2125 880 1132 2216 1397 1437 1914 2943 2642 1641 1416 2168 996 1759 2472 131 2891 2132 2753 3122 2690 2229 2370 3238
LARCENY
0 5000 10000 15000 20000
1379 16041 2706 6171 18173 5843 1843 2586 3971 3289 1914 3477 21786 5977 4991 3824 114 5087 2060 6071 5285 5308 3941 2533 3883
MURDER
0 20 40 60
10 3 7 17 16 29 30 35 46 39 41 69 8 14 19 1 55 27 49 55 21 41 31 51
ROBBERY
0 500 1000 1500 2000
47 744 189 281 1150 416 388 595 908 718 513 629 2233 532 436 431 10 912 511 1209 1134 817 908 556 1002


Crime distribution wrt Month
Month
January
February
March
April
May
June
July
August
September
October
November
December
26098
20553
23402
25300
27900
28133
29290
28666
27337
27751
26450
27395
0
10000
20000
30000
Frequency


Crime distribution by day of the week
Day of the week
Sunday
Saturday
Friday
Thursday
Wednesday
Tuesday
Monday
AGG ASSAULT
AUTO THEFT
BURGLARY-NONRES
BURGLARY-RESIDENCE
HOMICIDE
LARCENY-FROM VEHICLE
LARCENY-NON VEHICLE
MANSLAUGHTER
RAPE
ROBBERY-COMMERCIAL
ROBBERY-PEDESTRIAN
ROBBERY-RESIDENCE
Crime
Number of crimes
10000
5000



The above heat maps were able to tell us that some crimes were at peak during certain days of the week and certain times of the day. The bar charts were able to tell us what type of crime was prominent in what NPU and in what month.



Latitude: 33.7365
Longitude: -84.4
Neighborhood: West End
Number of Records: 82

After the selection of the attributes which were, type of crime, time of crime(time, day of week and month) and location of crime(divided into 26 divisions based on NPU), we start implementing the algorithm.

CHAPTER 5

IMPLEMENTATION PLANNING

5.1 Implementation Environment

We have used R to develop the project.

R is a programming language, mainly dealing with the statistical computation of data and graphical representations. Many data science experts claim that R can be considered as a very different application, of its licensed contemporary tool, SAS. The various offerings of this tool include linear and non-linear modelling, classical statistical tests, time-series analysis, clustering and graphical representation. It can be referred to as a more integrated suite of software facilities, for data manipulation, calculation and data visualization. The R environment is more of a well-developed space for an R programming language, inclusive of user-defined recursive functions as well as input and output facilities.

When it comes to graphical representation, the related attributed to R are extremely exemplary. It has over 4800 packages available, in its environment which belong to various repositories with specialization in various topics like econometrics, data mining, spatial analysis and bioinformatics.

5.2 Steps involved in developing the project:

The overall process of finding and interpreting patterns from data involves the repeated application of the following steps:

1. Data Cleaning: Data cleaning is defined as removal of noisy and irrelevant data from collection.
   - Cleaning in case of Missing values.
   - Cleaning noisy data, where noise is a random or variance error.
   - Cleaning with Data discrepancy detection and Data transformation tools.
2. Data Integration: Data integration is defined as heterogeneous data from multiple sources combined in a common source (Datawarehouse).
   - Data integration using Data Migration tools.
   - Data integration using Data Synchronization tools.

    - Data integration using ETL(Extract-Load-Transformation) process.

3. Data Selection: Data selection is defined as the process where data relevant to the analysis is decided and retrieved from the data collection.
4. Data Transformation: Data Transformation is defined as the process of transforming data into appropriate form required by mining procedure.
   Data Transformation is a two-step process:
    - Data Mapping: Assigning elements from source base to destination to capture transformations.
    - Code generation: Creation of the actual transformation program.
5. Data Mining: Data mining is defined as clever techniques that are applied to extract patterns potentially useful.
    - Transforms task relevant data into patterns.
    - Decides purpose of model using classification or characterization.
6. Pattern Evaluation: Pattern Evaluation is defined as as identifying strictly increasing patterns representing knowledge based on given measures.
    - Find interestingness score of each pattern.
    - Uses summarization and Visualization to make data understandable by user.
7. Knowledge representation: Knowledge representation is defined as technique which utilizes visualization tools to represent data mining results.
    - Generate reports.
    - Generate tables.
    - Generate discriminant rules, classification rules, characterization rules, etc.

5.3 Algorithms:

Apriori Algorithm

Name of the algorithm is Apriori because it uses prior knowledge of frequent itemset properties. We apply an iterative approach or level-wise search where k-frequent item sets are used to find k+1 item sets.

To improve the efficiency of level-wise generation of frequent item sets, an important property is used called *Apriori property* which helps by reducing the search space.

Apriori Property –

All non-empty subset of frequent itemset must be frequent. The key concept of Apriori algorithm is its anti-monotonicity of support measure. Apriori assumes that *"All subsets of a frequent itemset must be frequent (Apriori property). If an itemset is infrequent, all its supersets will be infrequent."*

While choosing the algorithm we had two choices:

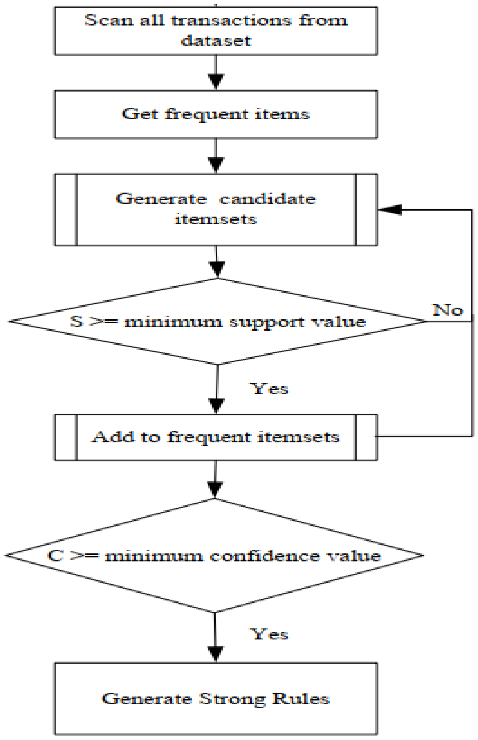
- The FP Growth algorithm
- Apriori algorithm

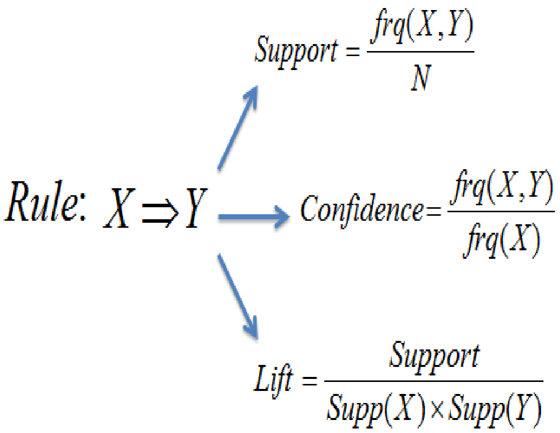
Comparing the two algorithms, theoretically, FP Growth has several advantages one of the main being that since it does a maximum of two scans it is faster than the Apriori algorithm which has to scan every time.

During execution of the project though, the FP growth took almost seven minutes to gives us the same results where by Apriori took less than a minute. So, for this project we chose Apriori over FP growth.

5.4 Execution

To generate rules Apriori has requires two parameters: Support and confidence. The figure below shows general steps for the algorithm.


Scan all transactions from dataset
Get frequent items
Generate candidate itemsets
S >= minimum support value
No
Yes
Add to frequent itemsets
C >= minimum confidence value
Yes
Generate Strong Rules


$Rule: X \Rightarrow Y$
$Support = \frac{frq(X,Y)}{N}$
$Confidence = \frac{frq(X,Y)}{frq(X)}$
$Lift = \frac{Support}{Supp(X) \times Supp(Y)}$

The general rule for the associations were:

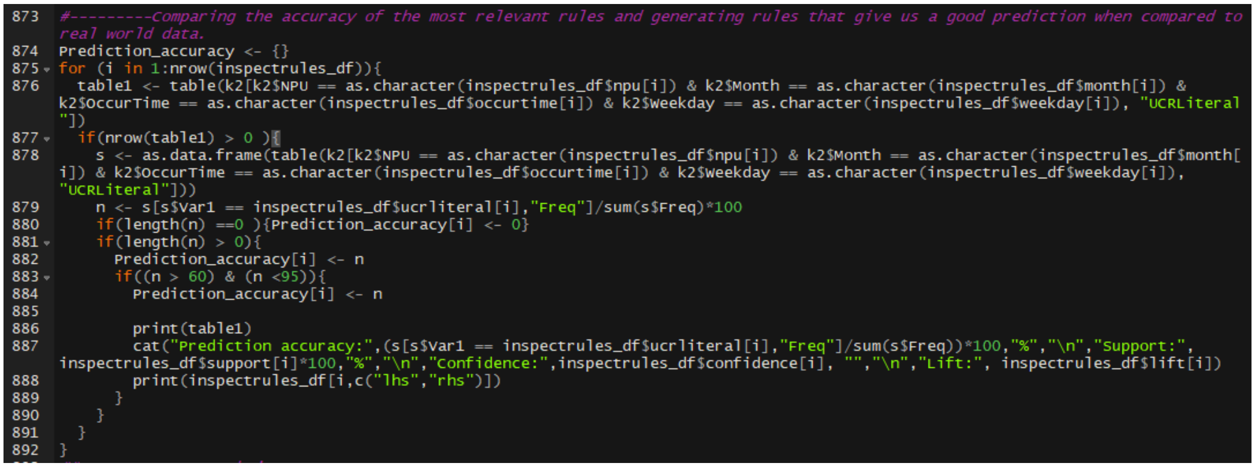
Whenever X ∈ crime reports

NPU(x,NPU_value)^Month(x,Month)^OccurTime(x,OT_value)^Weekday(x,day_of_the_week) → crime(x,UCR_literal)

After the association rules were generated, we had to refine our associations as most of the associations generated were not very accurate.

To refine the rules, we realized that we had to test our results on the data too. So we divided the crime data from 2008 to 2015 to generate the associations. The associations generated were tested on the test data which contained data from the year 2016 to 2018.

This helped us attain better accuracy on the associations generated.



```r
#---------Comparing the accuracy of the most relevant rules and generating rules that give us a good prediction when compared to real world data.
Prediction_accuracy <- {}
for (i in 1:nrow(inspectrules_df)){
  table1 <- table(k2[k2$NPU == as.character(inspectrules_df$npu[i]) & k2$Month == as.character(inspectrules_df$month[i]) & k2$OccurTime == as.character(inspectrules_df$occurtime[i]) & k2$Weekday == as.character(inspectrules_df$weekday[i]), "UCRLiteral"])
  if(nrow(table1) > 0 ){
    s <- as.data.frame(table(k2[k2$NPU == as.character(inspectrules_df$npu[i]) & k2$Month == as.character(inspectrules_df$month[i]) & k2$OccurTime == as.character(inspectrules_df$occurtime[i]) & k2$Weekday == as.character(inspectrules_df$weekday[i]), "UCRLiteral"]))
    n <- s[s$Var1 == inspectrules_df$ucrliteral[i],"Freq"]/sum(s$Freq)*100
    if(length(n) ==0 ){Prediction_accuracy[i] <- 0}
    if(length(n) > 0){
      Prediction_accuracy[i] <- n
      if((n > 60) & (n <95)){
        Prediction_accuracy[i] <- n

        print(table1)
        cat("Prediction accuracy:",(s[s$Var1 == inspectrules_df$ucrliteral[i],"Freq"]/sum(s$Freq))*100,"%","\n","Support:", inspectrules_df$support[i]*100,"%","\n","Confidence:",inspectrules_df$confidence[i], "","\n","Lift:", inspectrules_df$lift[i])
        print(inspectrules_df[i,c("lhs","rhs")])
      }
    }
  }
}
```

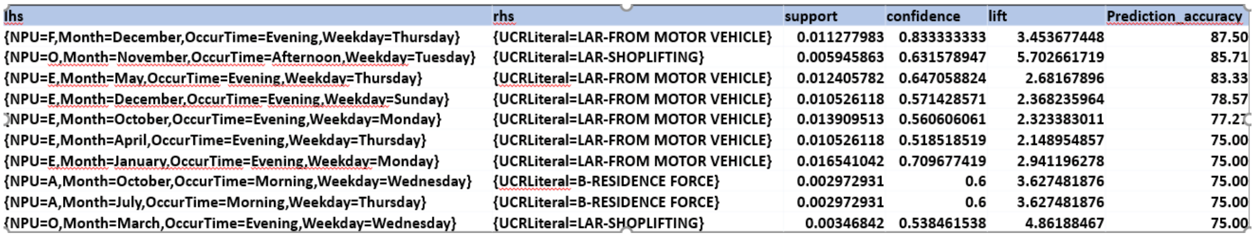
CHAPTER 6

CONCLUSION

At the end of the project, we were able to obtain 57 strong rules with the minimum accuracy prediction threshold being 65%.

The best rule generated was at a accuracy level of 87.5%.

The rule basically stated "In the NPU(division) F and months of December, during Thursday Evenings, Larceny from motor vehicle is very common".



lhs	rhs	support	confidence	lift	Prediction_accuracy
{NPU=F,Month=December,OccurTime=Evening,Weekday=Thursday}	{UCRLiteral=LAR-FROM MOTOR VEHICLE}	0.011277983	0.833333333	3.453677448	87.50
{NPU=O,Month=November,OccurTime=Afternoon,Weekday=Tuesday}	{UCRLiteral=LAR-SHOPLIFTING}	0.005945863	0.631578947	5.702661719	85.71
{NPU=E,Month=May,OccurTime=Evening,Weekday=Thursday}	{UCRLiteral=LAR-FROM MOTOR VEHICLE}	0.012405782	0.647058824	2.68167896	83.33
{NPU=E,Month=December,OccurTime=Evening,Weekday=Sunday}	{UCRLiteral=LAR-FROM MOTOR VEHICLE}	0.010526118	0.571428571	2.368235964	78.57
{NPU=E,Month=October,OccurTime=Evening,Weekday=Monday}	{UCRLiteral=LAR-FROM MOTOR VEHICLE}	0.013909513	0.560606061	2.323383011	77.27
{NPU=E,Month=April,OccurTime=Evening,Weekday=Thursday}	{UCRLiteral=LAR-FROM MOTOR VEHICLE}	0.010526118	0.518518519	2.148954857	75.00
{NPU=E,Month=January,OccurTime=Evening,Weekday=Monday}	{UCRLiteral=LAR-FROM MOTOR VEHICLE}	0.016541042	0.709677419	2.941196278	75.00
{NPU=A,Month=October,OccurTime=Morning,Weekday=Wednesday}	{UCRLiteral=B-RESIDENCE FORCE}	0.002972931	0.6	3.627481876	75.00
{NPU=A,Month=July,OccurTime=Morning,Weekday=Thursday}	{UCRLiteral=B-RESIDENCE FORCE}	0.002972931	0.6	3.627481876	75.00
{NPU=O,Month=March,OccurTime=Evening,Weekday=Wednesday}	{UCRLiteral=LAR-SHOPLIFTING}	0.00346842	0.538461538	4.86188467	75.00

FUTURE ENHANCEMENTS

There is a lot of scope of improvement in the project.

If given access to the right data, we can use the location of public amenities like hospitals, schools, restaurants and et cetera to show what type of crime normally happens around them.

We can also use the detailed data of the area's economy to enhance on the prediction of what type of crime happens in the less economically advanced areas and which in the "rich" areas.

The above stated improvements can be done if credible data is obtained.

Another enhancement could be made toward identifying the possible victims of certain crimes by having the detailed criminal records and mining on what type of people they would normally choose as their victims.

CHAPTER 7

REFERENCES

This project was completed using other sources from the internet and articles by other authors.

- https://www.tutorialspoint.com/big_data_analytics/association_rules.htm
- https://www.slideshare.net/StutiDeshpande/crime-dataset-analysis-for-city-of-chicago?from_action=save
- http://www2.cs.uregina.ca/~dbd/cs831/notes/kdd/1_kdd.html
- https://www.tutorialspoint.com/data_mining/dm_knowledge_discovery.htm
- https://www.guru99.com/r-tutorial.html
- Buczak, A. L., & Gifford, C. M. (2011). *Fuzzy association rule mining for community crime pattern discovery*. https://doi.org/10.1145/1938606.1938608
- Mehmet Sevri, Hacer Karacan, and M. Ali Akcayol (2017). Crime Analysis Based on Association Rules Using Apriori Algorithm http://doi: 10.18178/ijiee.2017.7.3.669