Unveiling Urban Insights: Enhancing Law Enforcement Efficiency through Data Mining of Complex Criminal Activities

Blake Wang

August 2023

Abstract

The rise of complex criminal activities in modern metropolises has given rise to concerns about labour shortages in law enforcement. Efficient allocation of police resources has become imperative. This data analysis focuses on historical crime data to address dynamic criminal activities, aiming to identify critical crimes, expedite case resolutions, and enhance the efficiency of law enforcement operations. As the size of data grows, extracting insights from big data becomes challenging. This project employs data mining techniques to uncover patterns in a large crime dataset from Chicago City. The analysis covers four data mining techniques: principal components analysis (PCA), K-means clustering, Naïve Bayes classification, and association rule mining. K-means clustering identifies critical areas and timeframes of crime activity. Naïve Bayes achieves high accuracy under specific conditions. Association rule mining uncovers hidden associations within the data. The insights from this study aid in optimizing the allocation of police resources and contribute to city security management. However, limitations include the need for continuous updates due to evolving criminal activities, the influence of external factors, and the possibility that the applied method may not uncover other hidden patterns. Future research could consider additional factors and explore other data mining techniques for deeper insights.

Introduction

As the development of modern metropolis, the landscape of criminal activities become sophisticated, the law enforcement labour shortage has drawn public's concerns (Ray, 2022), thus, the efficient allocation of police resources has emerged as a critical necessity. This data analysis aims to address this challenge by focusing on the strategic distribution of police resource in response to dynamic criminal activities. The main objective is to quickly discover critical criminal activities, expedite case resolution and enhance operational efficiency. As the dataset become larger and larger, we are facing many challenges to discovering insight from big data (Sagiroglu & Sinanc, 2013), this project seeks to uncover patterns in a large crime dataset by using several data mining methods. The outcomes of this project can provide meaningful insight for the city law enforcement department, which can support the decision-making of law enforcement resources allocation and maintain the health and safety of modern metropolis.

Data

Data Description

The main dataset initially consists of a total 1,220,182 observations and 22 variables of crime data from 2017 to 2021 in Chicago City. It obtained from the Chicago Data Portal (Chicago Police Department), which collected by the Police Department according to the reporting records. We also proposed a supporting weather dataset, as a research indicated that weather data such as temperature is correlated to the crime activities (Ranson, 2014). The weather dataset was extracted from National Centres for Environmental Information Data access (NCEI), which contains historical daily maximum temperature, minimum temperature, and precipitation from Chicago Midway Airport weather station.

Data Pre-processing

To ensure better data quality and achieve high precision in our analysis, conducting a data preprocessing step is essential. Firstly, we excluded a total of 15 variables that are not needed for this analysis, such as ID Case Number and FBI Code. Secondly, we identified 40 observations containing missing data. Additionally, we noted that District 31 is not part of Chicago Police Department list, which comprises only 40 observations. This discrepancy might be due to a typing error. Considering the dataset's size, we removed a total of 80 observations. Thirdly, we created three new variables: weekday, hour, and month. We extracted this information from the Date variable. Furthermore, we categorized the daily temperature into very cold, cold, cool, warm, and hot, assigning these categories to a new variable named TEMP. Lastly, we assumed that a precipitation measurement greater than zero indicated a rainy day, leading us to create a binary variable called RAIN.

After the above pre-processing, we merged two datasets by using the *merge()* function. The cleaned dataset consists of 1,220,102 observations and 13 variables as *Table 1* shows.

Table 1: Dataset Overview

Variable Name	Description	Data Type	Variable Type
DATE	Date and time of the incident happened	Date	Ordinal
Primary Type	Crime types	Character	Nominal
Description	Crime description	Character	Nominal
Location Description	Place of crime happen	Character	Nominal
Arrest	True or False	Binary	Discrete
District	23 police districts	Integer	Nominal
Block	Small blocks incident happened	Character	Nominal
Community Area	77 Community Area	Integer	Nominal
Weekday	Monday to Sunday	Integer	Discrete
Hour	0 – 24 hours	Integer	Discrete
Month	12 months of a year	Integer	Discrete
TEMP	Temperature reference	Double	Continuous
RAIN	Rain or not?	Binary	Discrete

Method

Software

R Studio was employed to this study. Essential packages include *dplyr* (Wickham, François, et al., 2023), *tidyverse* (Wickham, Vaughan, et al., 2023) and ggplot2 (Wickham, 2016). *dplyr* and *tidyverse* specifically apply in data processing part, and ggplot2 is our main visualisation tool.

Data Processing

In this data analysis, the *Arrest* variable is our indicate of a critical crime. An exploratory data analysis was conducted before the data mining process. We discovered *Theft* and *Battery* are the most frequent crimes. However, these two types of crimes only have a low arrest rate. As our main goal is to optimize law enforcement resources across Chicago City, therefore, we are going to divide these police districts according to the frequency of critical crimes. A new dataset was created by summarizing *hour*, *District and* counts of arrest using *filter(), group_by(), summarise()*, and *pivot_wider().* The dataset contains 23 observations which represent the police district, and 24 continuous numeric variables represent each hour range of the day. As some districts might suffered great number of crime and some might not, we scaled our dataset using *scale()* before the next step.

Principle Components Analysis

As our target variables are continuous numeric, we suggest the K-means Clustering as a dataset separation technique. Before conducting K-mean Clustering, we performed a principal components analysis (PCA). PCA can reduce the dataset's dimensionality while minimizing information loss. This, in turn, enhances the centroid point for clustering (Zhu et al., 2019).We performed PCA using prcomp() from the R stat package. In this step, we also employed scree plot and cumulative curve to look for the optimal representative dimensions.

K-mean Clustering

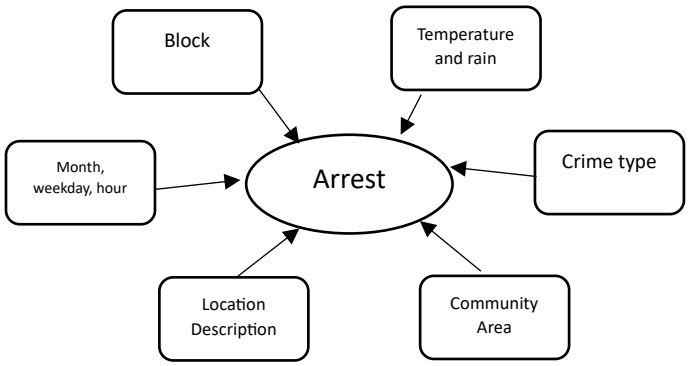
The process of K-mean Clustering is based on the initial assignment of k clusters, each cluster will iterative compute centroid and assign new clusters according the centroid point until the centroid stop changing (Gareth et al., 2022). Thus, we applied both the Elbow method and the Silhouette method to look for the optimal *k* for our clustering. We used *forloop()* to iteratively compute the total within-cluster sum of squares in result of k-mean for Elbow Method, and employed *silhouette()* in *cluster* package for Silhouette method. Finally, we utilized *merge()* to label the clusters number into our main dataset as a new variable. As each cluster represents a collection of police districts according to critical criminal activities, we will analyse each cluster individually to gain insight.

Naïve Bayes classification

As the main dataset contains numbers of categorical variables, we proposed a model as *Figure 1* shows to predict the probability of arrest if the police department received

a crime report. Naïve Bayes Classifier is a simple probability base classifier that required assumptions on features' independence. Despite potential violated of assumption, Naïve Bayes Classifier is giving good performance in big and complex dataset (Vural & Gök, 2017). We understand that the month and temperature might be correlated with each other; however, we don't want to lose some specific information related to the month, such as holiday seasons. Therefore, Naïve Bayes Classifier was chosen to implement on our model. As we assumed the different characteristics of clusters, we performed Naïve Bayes individually to each cluster. In R, we utilized *naiveBayes* in *e1071* package.

Figure 1: Model for predict the arrest of the crime.



Association Rule Mining

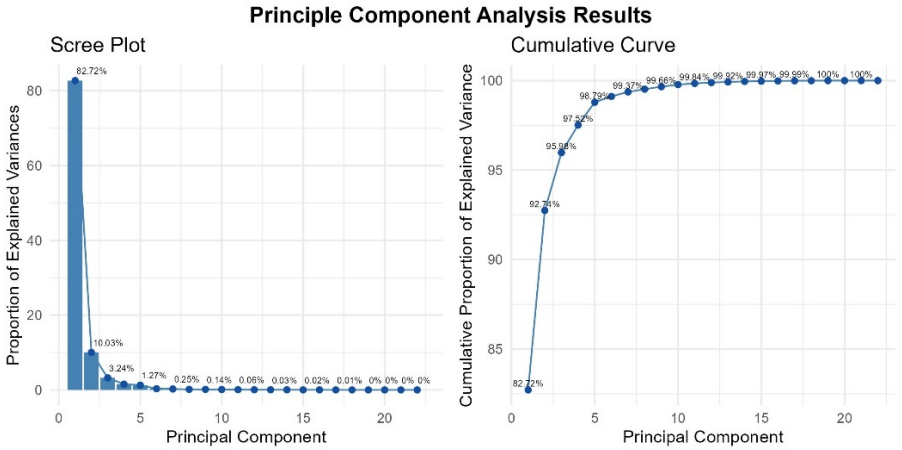
Lastly, we performed association rule mining using the Apriori method, a data mining technique employed to discover frequent itemsets within a dataset and derive association rules from them (Hegland, 2007).

The two crucial indicators in this technique are support and confidence. In a simple term, if the association rule is represented as R: $A=>B$, *A* is any possible combinations of variable values within a row, *B* is the other variable values in this row. Support refers to the probability of occurrence of A in the entire dataset, while confidence represents the probability of both A and B appearing in the same row where A is present. This approach allows us to uncover latent associations within our dataset, thereby enhancing the applicability of our previously proposed model. We applied the *apriori()* function in *arules* package to perform association rule mining.

Result and Discussion

The goal of the PCA process is to reduce the dimensions of the dataset. *Figure 2* shows that the first two principal components represent nearly 94% of the explained variance. This reduced our dataset for K-mean clustering to two dimensions, which is helpful, because K-mean clustering can produce more robust results in datasets of lower dimensionality (Zhu et al., 2019).

Figure 2: Scree Plot and Cumulative Curve of Principle Components



As mentioned earlier, *two* methods *were used* to determine the optimal value of k for clustering. As shown in *Figure 3*, while the Silhouette method suggests that k = 2 might be a preferable choice, the Elbow method demonstrates a more pronounced *changing* between 2 and 3. *Therefore*, we opted to select k = 3 for implementation in our clustering.

Figure 3: Elbow Method and Silhouette Method Results

Figure 4 shows the outcome of K-mean clustering. It is evident that district 11 is far from the other points, forming a *single* cluster on its own. *Next*, we will delve into investigating the reasons behind this occurrence.

Figure 4: K-Mean Clustering Results

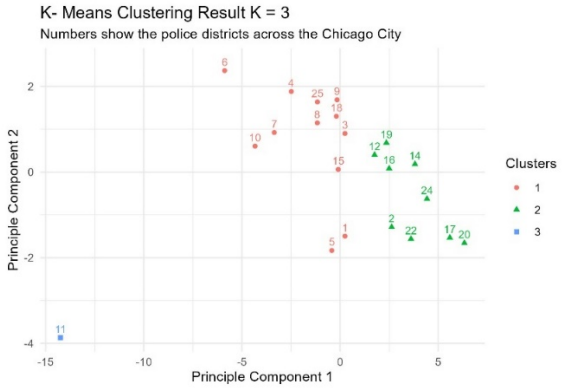


Figure 5 illustrates the arrest frequency within the three clusters across 24 hours. In comparison to Cluster 1 and Cluster 2, Cluster 3 exhibits a notably higher arrest rate. Furthermore, both Cluster 2 and Cluster 3 show a high number of arrest frequency around 6 PM – 8 PM.

Figure 5: Counts of Crimes by Each Hour of a Day

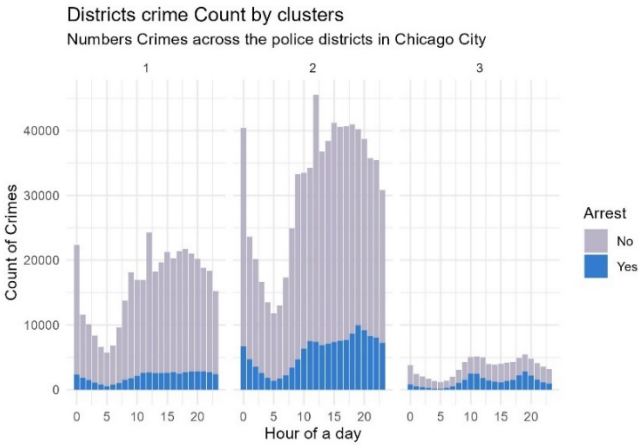


Figure 6 reveals that Cluster 3 experiences a considerable number of Narcotics-related crimes. The predominant arrest crime type in Cluster 1 is Battery. In Cluster 2, although both Narcotics and Battery incidents are prominent, weapon violations show a considerably higher occurrence.

Figure 6: Numbers of arrest by Primary Crime Types

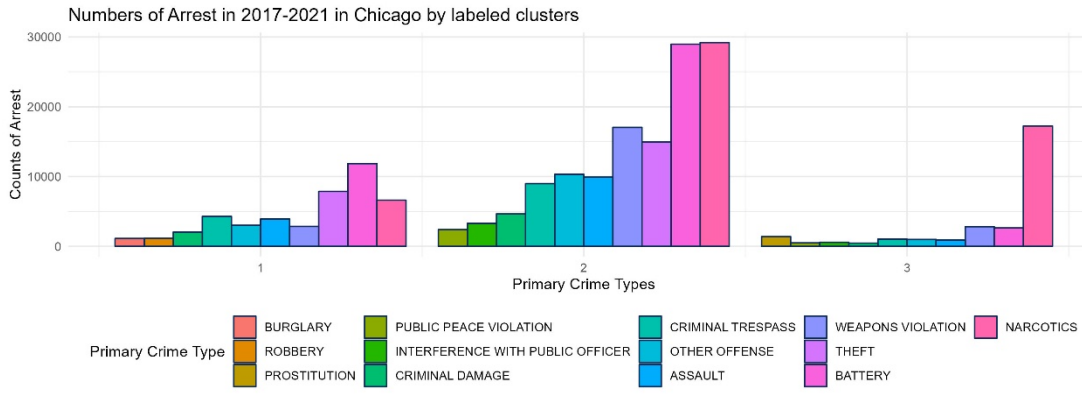


Table 2 presents the results of the Naïve Bayes classification, with the F1 score serving as our indicator for evaluating the classifier. The results demonstrate high accuracy. However, the specificity or true positive in Cluster 1 is relatively low. Cluster 2 gives slightly better performance, while Cluster 3 demonstrates the highest sensitivity. The prior probabilities are also provided in *Table 2*. In comparison, Naïve Bayes continues to provide a significant improvement.

Table 2: Naïve Bayes Classifier Results with all features

	F1 Score	Specificity	Sensitivity	Prior False	Prior True
Cluster 1	0.8943005	0.4103594	0.8791472	0.8699670	0.1300330
Cluster 2	0.9021979	0.4743581	0.9236014	0.8094298	0.1905702
Cluster 3	0.9204466	0.7871774	0.9516496	0.6412130	0.3587870

Furthermore, to ensure that these data are not dominantly contributed by a single feature, we conducted multiple iterations of Naïve Bayes Classification by individually removing each feature. We observed that the crime type contributes the most to the model, as evidenced by its significant impact when removed as shown in *Table 3*.

Table 3: Naïve Bayes Classifier Results Drop Primary Crime Type

	F1 Score	Specificity	Sensitivity	Prior False	Prior True
Cluster 1	0.8604773	0.2282674	0.8410689	0.8699670	0.1300330
Cluster 2	0.8684417	0.2167462	0.9091052	0.8094298	0.1905702
Cluster 3	0.8169801	0.6116570	0.8369536	0.6412130	0.3587870

Based on the findings from the Naïve Bayes Classifier, we conducted association rule mining without considering the primary crime type for each cluster. For Cluster 1 and 2, for the reason of the large dataset, we set the minimum support to 0.005, and for Cluster 3, the smaller dataset, we set it to 0.01. The confidence values were set based on the results of specificity in *Table 3* for each cluster.

Tables 4, 5, and 6 present certain example rules that can outperform the previous section's results when dealing with primary types that are unknown.

Table 4: Association Rules in Cluster 1

Pre.Rule	Post.Rule	Support	Confidence
{Location.Description=GROCERY FOOD STORE}	{Arrest=true}	0.0080088	0.4321916
{Community.Area=76}	{Arrest=true}	0.0051990	0.2605042
{Location.Description=RESIDENCE-GARAGE}	{Arrest=false}	0.0127279	0.9621611
{Location.Description=RESIDENCE,hour=12}	{Arrest=false}	0.0128311	0.9604094
{Location.Description=RESIDENCE,Community.Area=6}	{Arrest=false}	0.0064968	0.9537879
{Location.Description=RESIDENCE,hour=9}	{Arrest=false}	0.0109914	0.9523810
{Location.Description=OTHER (SPECIFY)}	{Arrest=false}	0.0081430	0.9480324
{Location.Description=RESIDENCE,Community.Area=16}	{Arrest=false}	0.0050107	0.9473171
{Location.Description=RESIDENCE,hour=10}	{Arrest=false}	0.0079030	0.9453704
{Location.Description=RESIDENCE,Community.Area=2}	{Arrest=false}	0.0054080	0.9407540
{Location.Description=RESIDENCE,Community.Area=28}	{Arrest=false}	0.0056067	0.9406926
{Location.Description=RESIDENCE,hour=0}	{Arrest=false}	0.0109914	0.9401898
{Location.Description=RESIDENCE,hour=14}	{Arrest=false}	0.0070025	0.9400762
{Location.Description=APARTMENT,hour=12}	{Arrest=false}	0.0095672	0.9399240
{Location.Description=RESIDENCE,hour=15}	{Arrest=false}	0.0077559	0.9396686

Table 5: Association Rules in Cluster 2

Pre.Rule	Post.Rule	Support	Confidence
{Location.Description=DEPARTMENT STORE}	{Arrest=true}	0.0078499	0.3960583
{Location.Description=SIDEWALK,TEMP=Cold}	{Arrest=true}	0.0066941	0.3515032
{Location.Description=SIDEWALK}	{Arrest=true}	0.0233089	0.3295070
{Community.Area=29}	{Arrest=true}	0.0142015	0.3208899
{Location.Description=SIDEWALK,TEMP=Warm}	{Arrest=true}	0.0086338	0.3198375
{Location.Description=ALLEY}	{Arrest=true}	0.0057577	0.2843928
{Location.Description=STREET,TEMP=Very Cold}	{Arrest=true}	0.0082164	0.2707993
{Location.Description=RESTAURANT,Community.Area=32}	{Arrest=false}	0.0070392	0.9458925
{Location.Description=RESIDENCE-GARAGE}	{Arrest=false}	0.0082526	0.9450061
{Location.Description=OTHER (SPECIFY)}	{Arrest=false}	0.0057550	0.9372549
{Location.Description=RESIDENCE,hour=12}	{Arrest=false}	0.0128156	0.9366445
{Community.Area=7}	{Arrest=false}	0.0146818	0.9302424
{Location.Description=RESIDENCE,hour=9}	{Arrest=false}	0.0106217	0.9263797
{Location.Description=APARTMENT,Community.Area=8}	{Arrest=false}	0.0050272	0.9226614

Table 6: Association Rules in Cluster 3

Pre.Rule	Post.Rule	Support	Confidence
{Block=033XX W FILLMORE ST}	{Arrest=true}	0.0166980	0.9244792
{Community.Area=29,Block=033XX W FILLMORE ST}	{Arrest=true}	0.0166980	0.9244792
{Location.Description=SIDEWALK,hour=10}	{Arrest=true}	0.0103598	0.8053016
{Location.Description=SIDEWALK,hour=11}	{Arrest=true}	0.0115240	0.7973963
{Location.Description=SIDEWALK,hour=19}	{Arrest=true}	0.0122413	0.7946565
{Location.Description=SIDEWALK,Community.Area=23,TEMP=Cold}	{Arrest=true}	0.0134407	0.7702156
{Location.Description=SIDEWALK,month=1}	{Arrest=true}	0.0105362	0.7612574
{Location.Description=SIDEWALK,TEMP=Very Cold}	{Arrest=true}	0.0179915	0.7522124
{Location.Description=RESIDENCE,TEMP=Hot}	{Arrest=false}	0.0132056	0.8998397
{Location.Description=APARTMENT,Community.Area=26,TEMP=Warm}	{Arrest=false}	0.0137112	0.8996914
{Location.Description=RESIDENCE,Community.Area=27}	{Arrest=false}	0.0213076	0.8934911
{Location.Description=APARTMENT,month=7}	{Arrest=false}	0.0118062	0.8924444
{Location.Description=RESIDENCE,weekday=Mon}	{Arrest=false}	0.0138170	0.8921792
{Location.Description=APARTMENT,weekday=Mon}	{Arrest=false}	0.0197084	0.8919638
{Location.Description=APARTMENT,month=6}	{Arrest=false}	0.0123236	0.8903993

These rules can be described as below examples:

1. In Cluster 1, if the report location is resident area at 12PM, the arrest is false in the confidence of 96%.
2. In Cluster 2, if the report location is department store, the arrest is true in the confidence of 39%.
3. In Cluster 3, if the report location is the sidewalk in community area of 23 and in a cold day, the arrest is true in the confidence of 77%

Conclusion

We conducted several data mining techniques to delve into the crime data within the city of Chicago. The outcomes show critical crimes can predict by various features such as location, hour of a day, crime types. For example, District 11 exhibits a noteworthy prevalence of narcotics-related incidents, which dominates substantial law enforcement resources. However, not all our findings are applicable, in the district from Custer 1 and Cluster 2, it is hard to predict an arrest, which indicates there are more sophisticated patterns in these regions. Nonetheless, our findings provide some insight of crime patterns which can contribute to the city's security management department's decision-making on assignment of law enforcement resources.

However, our analysis is circumscribed by certain limitations. Firstly, our timeframe was confined to the years from 2017 to 2021, as the rapid change of criminal activities, re-analysis should be conduct when new data is available. Secondly, external factors, such as special events, school holidays, and the gender or race of the individual reporting the incident, can also significantly influence decision-making processes. Thirdly, while our methodology offers valuable insights, it may not perfectly fit the dataset. Hence, future research can conduct different data mining techniques to unearth more insights.

References

Chicago Police Department. *Crimes - 2001 to present*. https://data.cityofchicago.org/Public-Safety/Crimes-2001-to-present-Dashboard/5cd6-ry5g

Gareth, J., Daniela, W., Trevor, H., & Robert, T. (2022). *An introduction to statistical learning with applications in R*. Springer Science and Business Media.

Hegland, M. (2007). The apriori algorithm–a tutorial. *Mathematics and computation in imaging science and information processing*, 209-262.

NCEI. *Global Historical Climatology Network - Daily (GHCN-Daily), Version 3*. https://www.ncei.noaa.gov/

Ranson, M. (2014). Crime, weather, and climate change. *Journal of environmental economics and management*, *67*(3), 274-302.

Ray, Y. R. M. S. D. S. (2022). *'We need them desperately': US police departments struggle with critical staffing shortages*. CNN. Retrieved 21 Aug from https://edition.cnn.com/2022/07/19/us/police-staffing-shortages-recruitment/index.html

Sagiroglu, S., & Sinanc, D. (2013). Big data: A review. 2013 international conference on collaboration technologies and systems (CTS),

Vural, M. S., & Gök, M. (2017). Criminal prediction using Naive Bayes theory. *Neural Computing and Applications*, *28*, 2581-2592.

Wickham, H. (2016). ggplot2: Elegant Graphics for Data Analysis. In: Springer-Verlag New York.

Wickham, H., François, R., Henry, L., Müller , K., & Vaughan, D. (2023). dplyr: A Grammar of Data Manipulation. In.

Wickham, H., Vaughan, D., & Girlich, M. (2023). tidyr: Tidy Messy Data. In.

Zhu, C., Idemudia, C. U., & Feng, W. (2019). Improved logistic regression model for diabetes prediction by integrating PCA and K-means techniques. *Informatics in Medicine Unlocked*, *17*, 100179.

Appendix: R Code

Required Packages

```
library(ggplot2)     # Visualization
library(dplyr)       # Processing Data
library(tidyverse)   # Processing Data
library(factoextra)  # for analysis pcr
library(gridExtra)   # Regrid figures
library(ggpubr)      # output figures
library(knitr)       # output table
```

Data Pre-processing

```
## Weather Data Processing ----------------------------------------------------------------
weather <- read.csv("weather.csv",   # Reading data
                    header=T,   # The data has header
                    sep = ",",
                    dec = ".",
                    stringsAsFactors = F # decimal points in data
                   )
#summary(weather) # overview of data, check missing values
#glimpse(weather) # variables check
weather$DATE <- as.POSIXct(weather$DATE,
                           format = "%Y-%m-%d",
                           )

weather <- weather %>%
  mutate(TEMP = factor(case_when(((TMAX+TMIN)/2)/10 <= 0 ~ "Very Cold",
                          ((TMAX+TMIN)/2)/10 <= 10 ~ "Cold",
                          ((TMAX+TMIN)/2)/10 <= 15 ~ "Cool",
                          ((TMAX+TMIN)/2)/10 <= 25 ~ "Warm",
                          TRUE ~ "Hot"),
               levels = c("Very Cold", "Cold", "Cool", "Warm", "Hot")),  # Calculate daily av
erage temperature
         RAIN = factor(ifelse(PRCP > 0, "TRUE", "FALSE"))) %>%    # Precipitation to 1: rain,
0 false) %>%  # Snow to numeric
  select(DATE, TEMP, RAIN) %>%
  filter(year(DATE)>2016 & year(DATE)<2022) # Select our target years

## Crime data Processing ------------------------------------------------------------------
chicago <- read.csv("chicago_crime.csv",   # Reading data
                    header=T,   # The data has header
                    sep = ",",
                    dec = ".",
                    stringsAsFactors = T
                   )
# Check repetition
# table(duplicated(chicago$ID)) # no duplication found
# table((chicago$District))
# glimpse(chicago)
# extract time and week days and create a new dataset crime
chicago <- chicago %>%
  mutate(Date = as.POSIXct(chicago$Date,
                           format = "%m/%d/%Y %I:%M:%S %p")) %>%
  mutate(weekday = wday(Date,  label = TRUE, week_start = 1),
         hour = hour(Date),
         month = month(Date),
         DATE = as.POSIXct(format(Date, "%Y-%m-%d"))) %>%
  mutate(Community.Area = factor(Community.Area),
         District = factor(District),
         ) %>%
  select(-c(1, 2, 3, 5, 10, 11, 13, 15, 16, 17, 18, 19, 20, 21, 22))  %>% # Remove unnecessary
 columns
  filter(District!="31")

chicago <- na.omit(chicago) # Remove missing values 42

## Merge crime data and weather -----------------------------------------------------------
cw <- merge(chicago, weather, by = "DATE") # cleaned dataset
```

Principle Component Analysis

```
## New Dataset according where police station is ------------------------------------------
cw.dist.hour <- cw %>%
```

```
  filter(Arrest == "true") %>%  # we consider arrest could be a critical event
  group_by(District, hour) %>%
  summarise(count = n()) %>%
  pivot_wider(names_from = hour,
              names_prefix = "H",
              values_from = count,
              values_fill = 0)

cw.dist.hour[,-1] <- scale(cw.dist.hour[,-1])

## Apply PCA ==============================================================================
## 1) model---------------------------------------------------------------------------------
cw.pca <- prcomp (cw.dist.hour[,-1],  # selected all gene columns
                  scale = FALSE   # Normalization has done previous step
                  )
## 2) Get result ---------------------------------------------------------------------------
cw.eig <- get_eig(cw.pca)

## 3) Create screeplot----------------------------------------------------------------------
sc.cw.pca <- ggplot(cw.eig,
                    aes(x = c(1:22),
                        y = variance.percent)
                    ) +
  # Display barplot, points, and line
  geom_col(fill = "steelblue") +
  geom_line(col = "steelblue") +
  geom_point(col = "#1450A3")+
  # Add text to each points
  geom_text(aes(label = paste(round(variance.percent, 2),
                              "%", sep = "")),
            vjust = -1, hjust = -0.05,
            check_overlap = T,
            size = 2) +
  # Specify Labels and theme on the plot
  labs(title = "Scree Plot",
         x = "Principal Component",
         y = "Proportion of Explained Variances") +
  theme(axis.text.x = element_text(angle = 45, vjust = 0.5,
                                   hjust=0.5, size = 6)) +
  theme_minimal()
## 4) Cumulative Explained Variance by Principal Component----------------------------------
cm.cw.pca <- ggplot(cw.eig,
                    aes(x = c(1:22),
                        y = cumulative.variance.percent)) +
  # Display points, and line
  geom_line(col = "steelblue") +
  geom_point(col = "#1450A3")+
  # Add text to each points
  geom_text(aes(label = paste(round(cumulative.variance.percent, 2),
                              "%", sep = "")),
            vjust = -0.5,
            check_overlap = T,
            size = 2) +
  # Specifiy Labels and theme on the plot
  labs(title = "Cumulative Curve",
       x = "Principal Component",
       y = "Cumulative Proportion of Explained Variance") +
  theme(axis.text.x = element_text(angle = 45, vjust = 0.5,
                                   hjust=0.5, size = 6)) +
  theme_minimal()
## 5) Combine two plots---------------------------------------------------------------------
fig.1 <- annotate_figure(ggarrange(sc.cw.pca, cm.cw.pca, ncol = 2, nrow = 1),
                top = text_grob("Principle Component Analysis Results",
                color = "black", face = "bold", size = 14))
ggsave("fig 1.jpg", plot = fig.1, width = 8, height = 4)
```

K mean Clustering

```
## 1) Cluster Validation ==================================================================
set.seed(789)
library(cluster)
## Create a for loop for Elbow and Silhoutte------------------------------------------------
k.vl <- data.frame() # A empty dataframe for assign test values
 # for silhuotte()
```

```r
crime.pca.k <- cw.pca$x[,1:2]
for(i in 2:10){
  crime.pca.cl <- kmeans(crime.pca.k,
                   centers = i,
                   nstart = 50)
  # Elbow method calculation
  km.wss <- sum(crime.pca.cl$tot.withinss)

  # Silhouette  calculation
  s.crm <- silhouette(dist(crime.pca.k), # dissimilarity
                      x=crime.pca.cl$cluster # cluster label
                     ) # individual silhouette
  avg.sil <- mean(s.crm[,3]) # average silhuott values
  k.vl <- rbind(k.vl,
                c(i, km.wss, avg.sil)) # store in the list
}
colnames(k.vl) <- c("k", "k.wss", "avg.sil")

## 2) Visulization Cluster validation ==========================================================
## Elbow method plot-------------------------------------------------------------------------
eb.p <- ggplot(data = k.vl, aes(x=k, y=k.wss)) +
  geom_point(col = "#1450A3")+
  geom_line(col = "steelblue") +
  theme_minimal()+
  labs(title="Elbow Method", y = "Total Within-Cluster Sum of Squares" )

## Silhuotte method plot ---------------------------------------------------------------------
s.p <- ggplot(data = k.vl, aes(x=k, y=avg.sil)) +
  geom_point(col = "#1450A3")+
  geom_line(col = "steelblue") +
  theme_minimal()+
  labs(title="Silhouette Method", y = "Average Silhouette Width" )
fig.2 <- annotate_figure(ggarrange(eb.p, s.p, ncol = 2, nrow = 1),
                top = text_grob("Cluster Validation Results",
                color = "black", face = "bold", size = 14))

ggsave("fig 2.jpg", plot = fig.2, width = 8, height = 4)

## K-Means Clustering Result Visulization ------------------------------ --------------------
crime.k <- kmeans(cw.pca$x[,1:2], 3,  nstart = 50)
crime.k.df <- data.frame("Clusters" = as.factor(crime.k$cluster),
                  "Type" = cw.dist.hour$District, cw.pca$x[,1:2])
fig.3 <- ggplot(crime.k.df,
                aes(x = PC1, y =PC2,
                    col = Clusters, shape = Clusters)) +
  geom_point() +
  geom_text(aes(label = cw.dist.hour$District),
            nudge_y = 0.2, size = 3, show.legend = FALSE) +
  theme_minimal() +
  labs(title= "K-Means Clustering Result K = 3",
       subtitle = "Numbers show the police districts across the Chicago City",
         x="Principle Component 1", y="Principle Component 2") +
  guides(label = FALSE)
ggsave("fig 3.jpg", plot = fig.3)
```

Cluster Analysis

```r
## Select Clusters ----------------------------------------------------------------------------
cw.cluster <- cw %>%
  mutate(cluster = ifelse(District %in% c(2, 12, 14, 16,
                                          17, 19, 20, 22, 24),
                          1,
                          ifelse(District != 11, 2, 3)))
## Visualization by arrest -------------------------------------------------------------------
fig.4 <- cw.cluster %>%
          #filter(Arrest == "true") %>%
          group_by(cluster, hour, Arrest) %>%
          summarise(count = n()) %>%
          ggplot() +
          geom_col(aes(y = count, x= hour, fill = Arrest)) +
          scale_fill_manual(values = c(true = "#337CCF",
                                       false = "#B9B4C7"),
                            labels = c("No",
                                       "Yes")) +
```

```
          facet_wrap(~cluster) +
          theme_minimal() +
          labs(title= "Districts crime Count by clusters",
               subtitle = "Numbers Crimes across the police districts in Chicago City",
                 x="Hour of a day", y="Count of Crimes")
ggsave("fig 4.jpg", plot = fig.4)

## Visualization by crime types ----------------------------------------------------------------
fig.5 <- cw.cluster  %>%
          filter(Arrest == "true") %>%
          group_by(cluster, Primary.Type) %>%
          summarise(count = n()) %>%
          arrange(-count) %>%
          slice(1:10) %>%
          ggplot(aes(fill = reorder(Primary.Type, count), y = count, x = cluster)) +
          geom_col(position = "dodge", col= "#102C57") +
         # facet_wrap(~cluster) +
          theme_minimal() +
          labs(x = "Primary Crime Types",
               y = "Counts of Arrest",
               title = "Numbers of Arrest in 2017-2021 in Chicago by labeled clusters",
               fill = "Primary Crime Type") +
          theme(legend.position = "bottom")
fig.5
ggsave("fig 5.jpg", plot = fig.5, width = 10, height = 5)
```

Naive Bayes

```
library(e1071)
library(caret)
set.seed(123)
acc.mes.list <- list()
## Perform Naive Bayes for all different sets----------------------------------------------
for (j in 0:10){
  acc_mes <- data.frame()
  # Select all features
  mod.fect <- c(Arrest, Location.Description,
                weekday, hour, TEMP, month, RAIN,
                Community.Area, Primary.Type, Block)
    ## Naive Bayes Classifier
    for (i in 1:3){
    #select interest features
      crime.s <- cw.cluster %>%
        filter(cluster==i) %>%
        select(ifelse(j == 0,
                      mod_fect,
                      mod_fect[-j])  # part of features
               )

      no_obs <- dim(crime.s)[1]
      # Randomly Extract a series of test index numbers
      test_index <- sample(no_obs,
                           size = as.integer(no_obs*0.2), # 10% of the total observation
                           replace = FALSE)
      # Training index number
      train_index <- -test_index   # removing test data from the dateset
      # Partition dataset
      test_naive <- crime.s[test_index,]    # Test dataset
      train_naive <- crime.s[train_index,]  # Training dataset

      nB.mod <- naiveBayes(Arrest ~., data=train_naive)
      pred.mod <- predict(nB.mod, test_naive)
      conf.mat <- confusionMatrix(pred.mod, test_naive$Arrest)
      precision <- conf.mat$byClass['Pos Pred Value']
      sens <- conf.mat$byClass['Sensitivity']
      spec <- conf.mat$byClass["Specificity"]
      f1_score <- conf.mat$byClass['F1']
      # True Positive
      acc_mes <- rbind(acc_mes, c(f1_score,
                                  spec, sens,
                                  (nB.mod$apriori/sum(nB.mod$apriori))[1],
                                  (nB.mod$apriori/sum(nB.mod$apriori))[2]))
      }
  colnames(acc_mes) <- c("F1 Score", "Specificity",
```

```r
                        "Sensitivity", "Prior False", "Prior True")
  rownames(acc_mes) <- c("Cluster 1", "Cluster 2", "Cluster 3")
  acc.mes.list[[j+1]] <- acc_mes
}
kable(acc.mes.list[[1]])   # all features
kable(acc.mes.list[[10]])  # feature without crime types
## Apply apriori to Cluster 1 to 3-----------------------------------------------------------
library(arules)
assorule.table <- list()
for (i in 1:3){
  cw.as <- cw.cluster %>%
    filter(cluster == i) %>%
    mutate(hour = factor(hour), month = factor(month))%>%
    select(c("Location.Description", "Arrest", "hour",
             "Community.Area", "TEMP", "month", "weekday", "Block"))
  cw.as <- as(cw.as, "transactions")
  # apriori algorithm
  rules.c <-
    apriori(data = cw.as,
            parameter = list(support = ifelse(i %in% c(1,2), 0.005, 0.01),
                             confidence = case_when(i == 1 ~ 0.2282674,
                                                    i == 2 ~ 0.2167462,
                                                    i == 3 ~ 0.6116570),
            minlen = "2", target = "rules"))

  rules.c.df <- data.frame(Pre.Rule = labels(lhs(rules.c)),
                           Post.Rule = labels(rhs(rules.c)),
                           Support = quality(rules.c)$support,
                           Confidence = quality(rules.c)$confidence)
  # Rules containr arrest=true
  rules.c.df.t <- rules.c.df %>%
    filter(Post.Rule == "{Arrest=true}") %>%
    arrange(desc(Confidence)) %>%
    slice_head(n = 8)
  # Rules containr arrest=true
  rules.c.df.f <- rules.c.df %>%
    filter(Post.Rule == "{Arrest=false}") %>%
    arrange(desc(Confidence)) %>%
    filter(case_when(i == 1 ~ Confidence > 0.8410689,
                     i == 2 ~ Confidence > 0.9091052,
                     i == 3 ~ Confidence > 0.8369536))
  assorule.table[[i]] <- rbind(rules.c.df.t, rules.c.df.f)
}
# Cluster 1 Rules Examples
kable(assorule.table[[1]][1:17,])
# Cluster 2 Rules Examples
kable(assorule.table[[2]][1:15,])
# Cluster 3 Rules Examples
kable(assorule.table[[3]][1:15,])
```