Big Data Capstone Design Assignment 1

Joel Shin

2023-04-22

Set library & Load data set

Set up the environment for EDA and proceed to set up packages and load data

1.Data explanation

The Dallas Crime Dataset is a comprehensive collection of crime records provided by the Dallas Police Department, spanning from 2014 to 2020. It encompasses a wealth of detailed variables such as crime zones, types, and response records, offering valuable insights into the city's public safety landscape. This dataset has wide application and versatility in social science as well as crime analysis and prevention.

1-1. Data Overview

```
## # A tibble: 6 × 6
##   incidentnum UCR_ctype   cnt ctype    servicenumberid watch
##   <chr>           <dbl> <dbl> <chr>    <chr>           <chr>
## 1 000001-2019        NA     1 ""       000001-2019-01  3
## 2 000001-2020        NA     1 ""       000001-2020-01  1
## 3 000002-2015         1     1 "MURDER" 000002-2015-01  3
## 4 000002-2018        NA     1 "FOUND"  000002-2018-01  1
## 5 000002-2019        NA     1 ""       000002-2019-01  1
## 6 000004-2015         6     1 "UUMV"   000004-2015-01  3
```

Table 1. Sample of a Dataset

The head(), ncol(), and nrow() functions were utilized to give an overview of the dataset. By using the ncol() and nrow() variables, we see that the data has 663249 rows and 107 columns. In addition, by using the head() function, The various missing values found besides NA, such as N/A and "" (blank).

1-2. Missing Value Processing

Various missing values in the data, such as “”, N/A, UNKNOWN, and unknown, were treated as NA values. We found a total of 14716551 missing values in the data, with an average of 175197 missing values per variable. In addition, Out of a total of 107 data variables, we found 22 variables with no missing data.

Typically, In Data preprocessing, NA values are removed or filled. However, This data is characterized by a large number of columns and a large number of identified NA values. So we decided to remove the NA values and dplyr::select the variables for the project through the univariate and multivariate analysis.

1-3. Checking Variable Type

Using the str() and typeof() functions to determine the class of each variable, we see that there are 103 character variables, 3 double variables, and 1 logical variable.

However, among the variables corresponding to character class, there are variables that can be factorized, such as variables with time information, variables with latitude information, and crime types, so we determined to change the type while checking the outlier, conducting univariate and multivariate analyses.

1-4. Checking Outlier

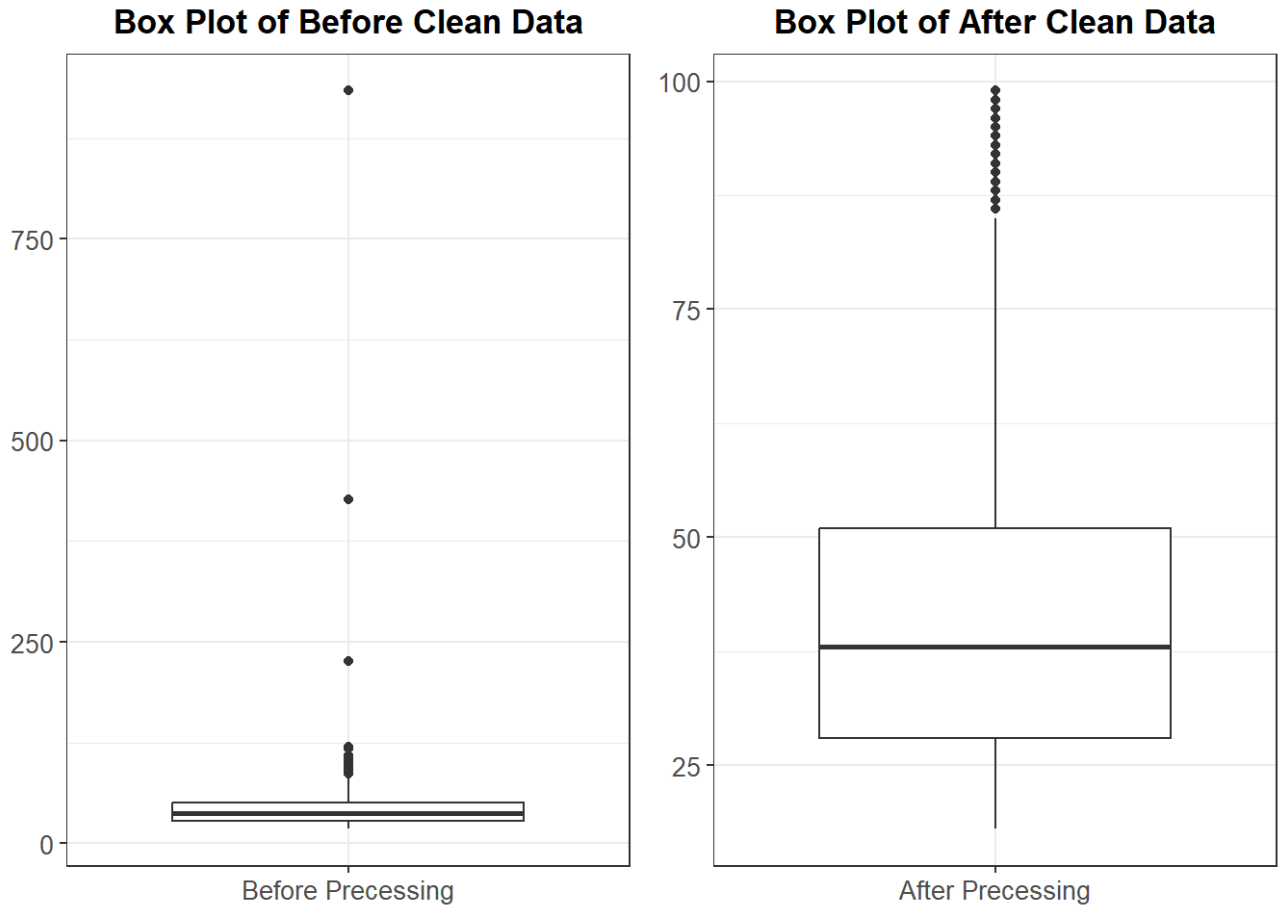

Figure 1. Boxplot Comparison of the Victimage Variable Before and After Processing (Y-axis is Age, Left is Before and Right is After)

```
##                Type Min. 1st Qu. Median             Mean 3rd Qu. Max.
## 1 Before Precessing   18      28     38 40.6727755929704      51  934
## 2  After Precessing   18      28     38 40.6555661689198      51   99
```

Table 2. Statistics Analysis comparison of the Victimage Variable Before and After Processing (Y-axis is Age, Left is Before and Right is After)

While looking at the descriptive statistics of numeric variables and variables that can be converted to numeric types in your data, you notice an outlier in the value of the victimage variable. We used quantile and boxplot analysis to determine the range and magnitude of the outliers.

However, given the number of data that needed to be removed was more than 300,000, it was determined that the logic of the data could be relatively preserved by handling and analyzing the outliers when performing EDA using the age variable, rather than immediately removing the outliers in that variable.

2. Univariate analysis

Univariate analysis is a fundamental statistical approach that examines a single variable to summarize its properties, including central tendencies, dispersion, and distribution shapes. This method allows for a better understanding of variable characteristics, serving as a foundation for more advanced techniques, such as multivariate analysis.

2-1. Univariate Descriptive Statistics Analysis

```
## # A tibble: 7 × 9
##   variable       mean  median    mode     max     min   iqr     sd       var
##   <chr>         <dbl>   <dbl>   <dbl>   <dbl>   <dbl> <dbl> <dbl>     <dbl>
## 1 watch             2       2       1       3       1     2     1         1
## 2 victimage        41      38      26     934      18    23    15       232
## 3 geo_lat          33      33      33      48      26     0     0         0
## 4 geo_long        -97     -97     -97     -73    -122     0     0         0
## 5 zipcode       75224   75224   75217   98004       0    22   176     30801
## 6 xcoordinate 2494419 2493161 2492641 2593936 2320290 28333 21916 480303722
## 7 ycordinate  6978451 6976100 6966517 7088137 6892807 46563 31331 981614774
```

Table 3. Statistics Analysis with Numeric Variables

For the univariate analysis, we converted all the variables that could be converted to numerical types and then used a for loop to extract descriptive statistics such as mean, minimum, range, and variance for each variable. Other than the victimage variable, which was found earlier, there were no other anomalies found.

2-2. Victim Race

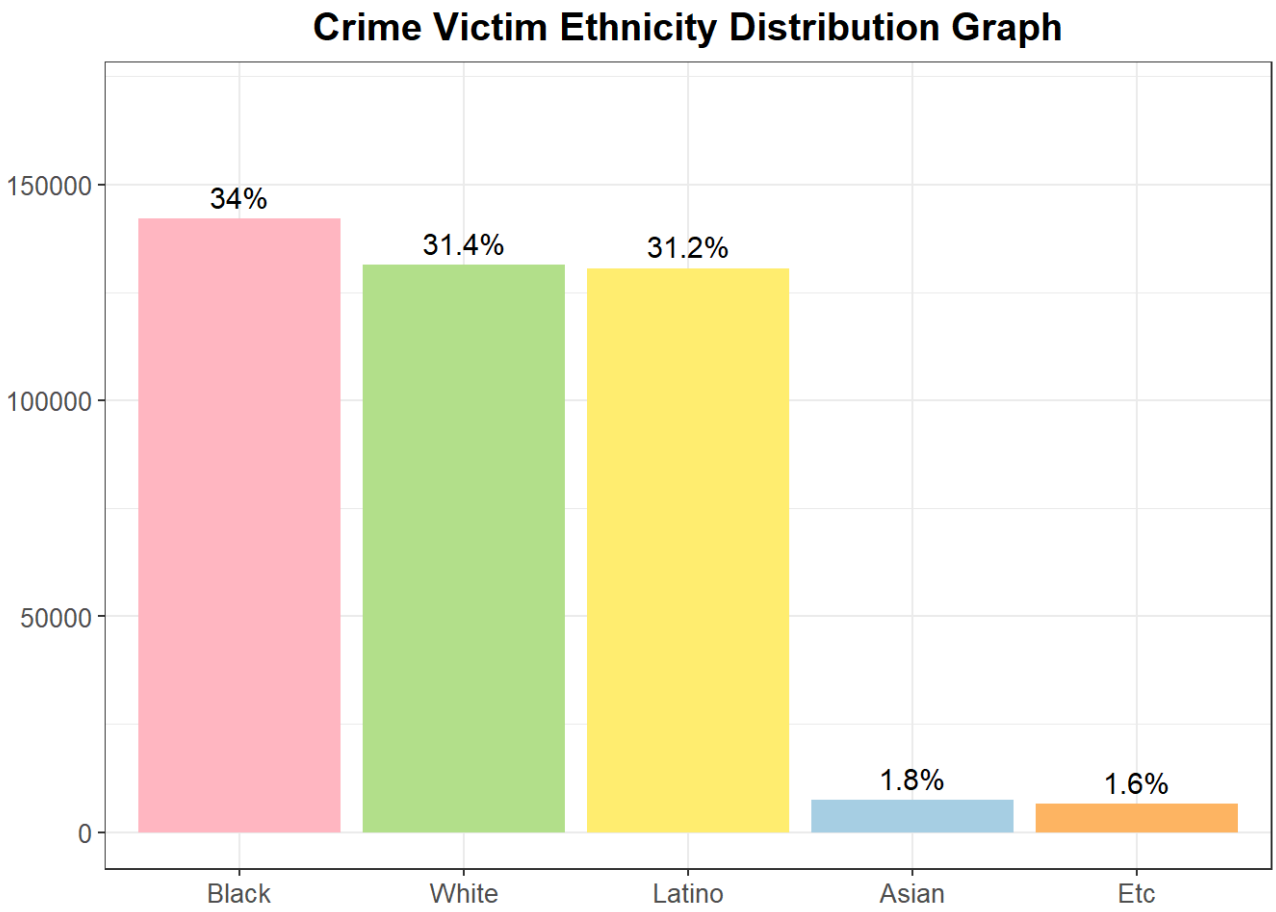

Figure 2. Crime Victim Ethnicity Distribution

We performed a bar graph analysis of the races of victims present in the data. The visualization graph shows that over the past four years, the highest number of crimes were committed against blacks, followed by whites and Latinos.

However, considering that the U.S. Census Bureau reports that whites make up about 62% of the U.S. population, Latinos about 20%, and blacks about 12% as of 2021, the percentage of black victims is relatively high compared to other races.

2-3. Offense Type

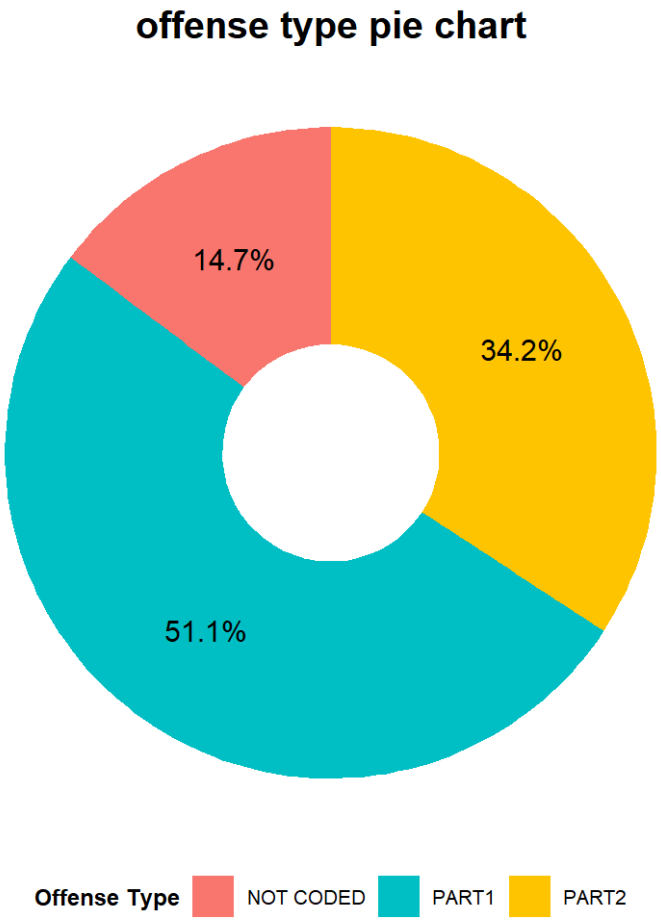

Figure 3. Offense Type Pie Chart

In the United States, crimes are classified into two primary categories: Part I Crimes and Part II Crimes, as used in the Uniform Crime Reporting (UCR) program managed by the Federal Bureau of Investigation (FBI). Part I Crimes, or “Index Crimes,” encompass eight serious offenses, including homicide, rape, robbery, aggravated assault, burglary, larceny-theft, motor vehicle theft, and arson. Part II Crimes, on the other hand, include lesser offenses such as fraud, vandalism, drug-related offenses, and simple assaults.

When we chart the graph with this information, we can see that the number of felonies in the United States is relatively higher than the number of misdemeanors. Additionally, given the existence of unrefined data labeled as “Not coded”, we need a process to predict “Not coded”.

Multivariate analysis

Multivariate analysis is an advanced statistical approach that investigates relationships, patterns, and interactions among multiple variables. This method enables the analysis of complex data sets and informs decision-making in various fields. Techniques include regression analysis, factor analysis, cluster analysis, and discriminant analysis, which help predict outcomes, simplify data, identify patterns, and classify groups based on variable interactions.

3-1. Carmér V analysis

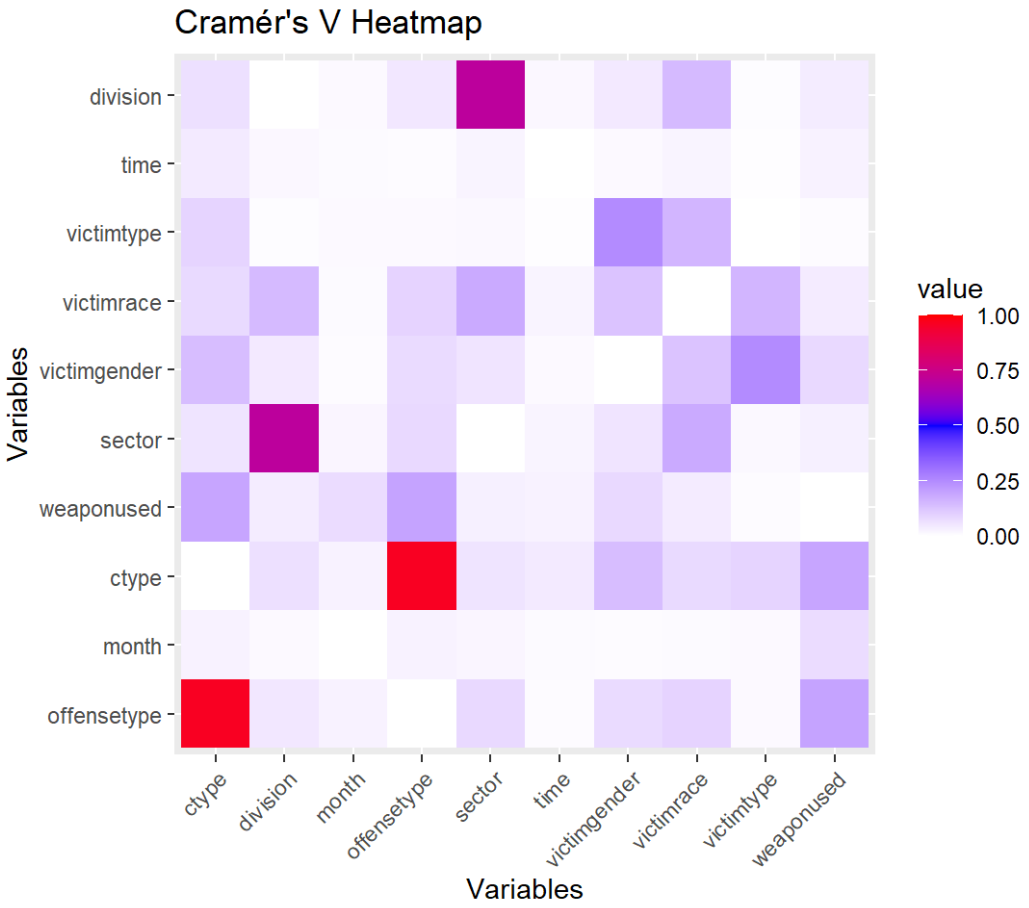

Figure 4. Cramér's V Heatmap

Cramérs V is a correlation coefficient used to measure the association between two nominal variables and to interpret the results of a Chi-square test. A value close to 0 indicates no association, while a value close to 1 indicates a strong association. The criteria for determining the strength of the association are $V < 0.1$ (weak), $0.1 \leq V < 0.3$ (moderate), and $0.3 \leq V$ (strong).

We extracted variables related to victims and police districts that were present in the data and analyzed them. And we found a strong association between the offensetype and ctype variables. This is consistent with our previous insights on offensetype. We also found that the weaponused variable was associated with both of these variables, as most violent crimes involve the use of a weapon.

Since division and sector are about the jurisdiction of the case, we found a relatively stronger association than other variables. One relationship that stood out was between victimtype and victimrace. We found that there was an association between race and victim type, which we assume to be the case for sexual offenses. We also found a reasonable association between victimrace and sector, indicating that the police are trying to customize their response based on crime cases and trends.

3-2. Crime Victim Ethnicity Tendency

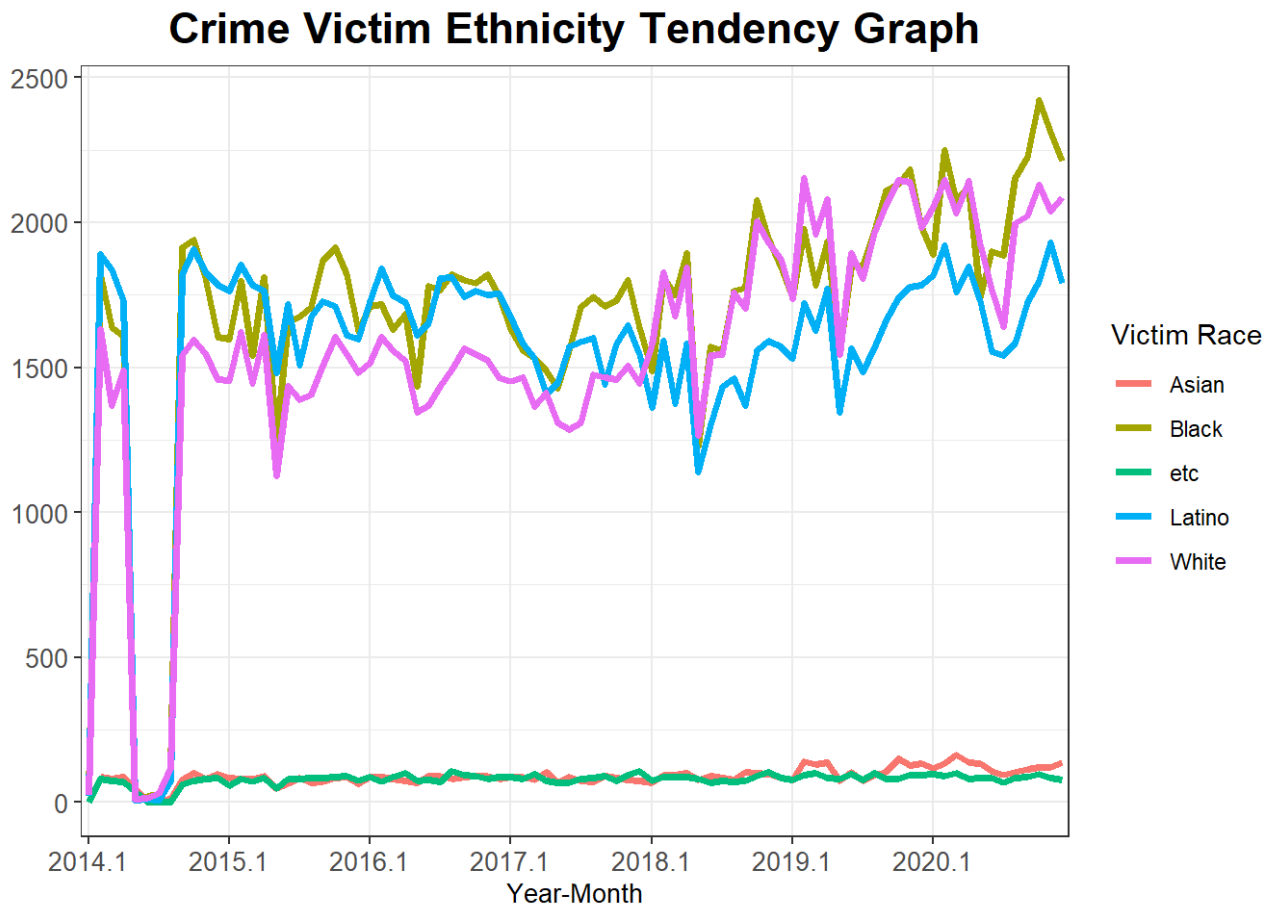

Figure 5. Crime Victim Ethnicity Tendency Graph

When I checked the line graphs for victim race by year and month, I saw that the overall number of crimes was increasing. In addition, in relation to hate crimes, which have recently become a social issue, we can see that Black, White, and Latino victims are the most prevalent ethnicities in the six months of 2020. We also see that the number of Asian victims has remained flat since 2019. One anomaly in the data is the absence of crime data for 2014, which is believed to be a data management issue.

4. Suggestion

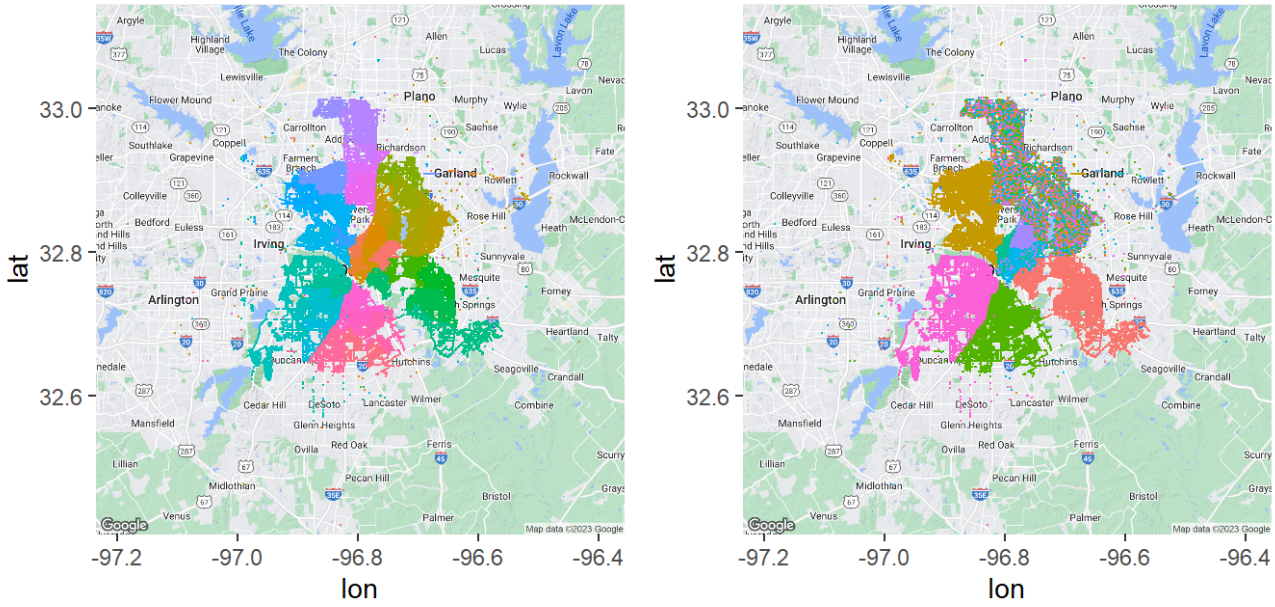

Figure 6. Compare the Original Sectors to the Resulting Sectors from Clustering

Along with South Korea, the U.S. is also concerned about population decline as a society. Population decline affects all areas of society. Ultimately, it leads to a decrease in the labor force, which requires measures to ensure a smooth social cycle.

While general jobs are finding a way out by utilizing overseas job seekers, specialized jobs such as police and military personnel, which fall under sovereign territory, need to devise new solutions. In particular, since the United States is a country where people can own firearms, it is necessary to introduce police security systems and technologies that are tailored to social phenomena.

While conducting the EDA, I realized that police operations could be more efficient if the current police coverage areas were modified according to recent crime trends and data. Therefore, I conducted clustering using information on crime areas and latitude and longitude information. As with the Cramér analysis, where it was difficult to find a significant association, there are some limitations in identifying the global optimum with current data in the case of sector classification using clustering.

However, as the results of clustering show, it is expected that new solutions can be proposed for some areas through consolidated area operations, and for areas that are unfortunately not well clustered, it is believed that if additional data and related variables are accumulated to propose better jurisdictional areas, more efficient police operations will be possible along with improved police technology and systems.

5. Appendix

5-1. Sources

Data sources (%22https://dallaspolice.net/%22)

Map Data sources (%22https://cloud.google.com/apis?hl=ko%22)

U.S. Population Decline Article (%22https://url.kr/sefv2w%22)

5-2. Experiment Setting

OS : Windows 11 Pro 21H2 Version

CPU : 12th Gen Intel(R) Core(TM) i9-12900

GPU : NVIDIA GeForce RTX 3090 X 2EA

RAM : Samsumg 32GB DDR4 25600 X 4EA

5-3. Code

```
# set seed and set directory
setwd("C:/Users/user/Downloads/")

# set the coding environment
set.seed(2023)
options(scipen = 100)

# set library
library(tidyverse)
library(haven)
library(moments)
library(vcd)
library(reshape2)
library(ggmap)
library(gridExtra)
library(lubridate)
library(gmodels)
library(cluster)
library(factoextra)
library(gower)
library(clustMixType)
library(klaR)
register_google(key = "AIzaSyBBRLxoRLmaE1spvQ5a0myycvPhifa8blY")

# load dallas data set
df <- read_dta("raw_Dallas.dta")
```

```
head(df)
ncol(df)
nrow(df)
```

```
print(paste0("The number of data set columns : ", ncol(df)))
print(paste0("The number of data set rows    : ", nrow(df)))
```

```
head(df[,c(1,2,3,4,6,7)])
```

```
df <- data.frame(lapply(df, function(x) ifelse(x == "", NA, x)))
df <- data.frame(lapply(df, function(x) ifelse(x == "N/A", NA, x)))
df <- data.frame(lapply(df, function(x) ifelse(x == "UNKNOWN", NA, x)))
df <- data.frame(lapply(df, function(x) ifelse(x == "Unknown", NA, x)))
df <- data.frame(lapply(df, function(x) ifelse(x == "unknown", NA, x)))

df_na_info <- c()
col_names <- names(df)
for (i in 1:ncol(df)) {
  mini_df <- data.frame(columns_name = col_names[i], NA_num = sum(is.na(df[,
i])))
  df_na_info <- rbind(df_na_info, mini_df)
}

df_na_info %>% dplyr::filter(NA_num != 0) %>% summarise(sum_of_na = sum(NA_num),
mean_of_na = mean(NA_num)) -> df_na_info

print(paste0("The sum of data set NA           : ", df_na_info$sum_of_na))
print(paste0("The mean of data each columns NA : ", round(df_na_info$mean_of_n
a)))
```

```
df_str_info <- c()
col_names <- names(df)
for (i in 1:ncol(df)) {
  mini_df <- data.frame(columns_name = col_names[i], type = typeof(unlist(df[,
i])))
  df_str_info <- rbind(df_str_info, mini_df)
}

df_str_info %>% group_by(type) %>% summarise(n = n())
```

```
df_str_info <- c()
col_names <- names(df)
for (i in 1:ncol(df)) {
  mini_df <- data.frame(columns_name = col_names[i], type = typeof(unlist(df[,
i])))
  df_str_info <- rbind(df_str_info, mini_df)
}

df_str_info %>% group_by(type) %>% summarise(n = n()) -> df_str_info
print(paste0("The number of character variables : ", df_str_info[1,2]))
print(paste0("The number of double variables    : ", df_str_info[2,2]))
print(paste0("The number of logical variables   : ", df_str_info[3,2]))
```

```
df$victimage <- as.numeric(df$victimage)
df %>% dplyr::select(victimage) %>% dplyr::filter(is.na(victimage) != TRUE) -> d
f_age
as.numeric(df_age$victimage) -> df_age

df_age_before <- data.frame(Group = rep("Before Precessing", 390576), value = df
_age)
Before <- ggplot(df_age_before, aes(x = Group, y = value)) +
  geom_boxplot() + theme_bw() +   xlab(NULL) + ylab(NULL) +
  labs(title = "Box Plot of Before Clean Data") +
  theme(plot.title = element_text(size = 13, face = "bold", hjust = 0.5),
        axis.title = element_text(size = 10), axis.text = element_text(size = 1
0))

box_stats <- boxplot.stats(df_age)
df_age[df_age < 100] -> df_age2
df_age_after <- data.frame(Group = rep("After Precessing", 390493), value = df_a
ge2)
After <- ggplot(df_age_after, aes(x = Group, y = value)) +
  geom_boxplot() + theme_bw() + xlab(NULL) + ylab(NULL) +
  labs(title = "Box Plot of After Clean Data") +
  theme(plot.title = element_text(size = 13, face = "bold", hjust = 0.5),
        axis.title = element_text(size = 10), axis.text = element_text(size = 1
0))

grid.arrange(Before, After, ncol = 2)
```

```
as.data.frame(cbind(Type = c("Before Precessing", "After Precessing"),
                    rbind(summary(df_age, na.rm = T),
                          summary(df_age2, na.rm = T))))
```

```
analysis_results <- tibble(
  variable = character(),
  mean = numeric(),
  median = numeric(),
  mode = numeric(),
  max = numeric(),
  min = numeric(),
  iqr = numeric(),
  sd = numeric(),
  var = numeric(),
  skewness = numeric(),
  kurtosis = numeric()
)

df_stat <- df %>%
  dplyr::select(watch, victimage, geo_lat, geo_long, zipcode, xcoordinate, ycord
inate)

for (col in colnames(df_stat)) {
  current_var <- df_stat[[col]]

  mean_val <- mean(current_var, na.rm = T)
  median_val <- median(current_var, na.rm = T)
  mode_val <- as.numeric(names(table(current_var))[which.max(table(current_va
r))])
  max_val <- max(current_var, na.rm = T)
  min_val <- min(current_var, na.rm = T)
  iqr_val <- IQR(current_var,  na.rm = T)
  sd_val <- sd(current_var, na.rm = T)
  var_val <- var(current_var, na.rm = T)
  skewness_val <- skewness(current_var, na.rm = T)
  kurtosis_val <- kurtosis(current_var, na.rm = T)

  analysis_results <- analysis_results %>%
    add_row(
      variable = col,
      mean = mean_val,
      median = median_val,
      mode = mode_val,
      max = max_val,
      min = min_val,
      iqr = iqr_val,
      sd = sd_val,
      var = var_val,
      skewness = skewness_val,
      kurtosis = kurtosis_val
    )
}
```

```
analysis_results$mean <- round(analysis_results$mean)
analysis_results$median <- round(analysis_results$median)
analysis_results$mode <- round(analysis_results$mode)
analysis_results$max <- round(analysis_results$max)
analysis_results$min <- round(analysis_results$min)
analysis_results$iqr <- round(analysis_results$iqr)
analysis_results$sd <- round(analysis_results$sd)
analysis_results$var <- round(analysis_results$var)

analysis_results[,-c(10:11)]
```

```
df_vicimrace <- as.data.frame(table(df$victimrace))
df_vicimrace <- rbind(df_vicimrace, data.frame(Var1 = c("Etc","Latino"),
                                               Freq = c(df_vicimrace[4,2] +
                                                 df_vicimrace[8,2] +
                                                 df_vicimrace[10,2] +
                                                 df_vicimrace[1,2] +
                                                 df_vicimrace[6,2] +
                                                 df_vicimrace[7,2],
                                                 df_vicimrace[5,2] +
                                                 df_vicimrace[9,2])))

df_vicimrace <- df_vicimrace[-c(1,4,5,6,7,8,9,10),]

df_vicimrace %>%
  ggplot(aes(factor(Var1, levels = Var1[order(Freq, decreasing = TRUE)]), Freq))
+
  geom_col(fill = c("#A6CEE3", "#FFB6C1", "#B2DF8A", "#FDB462", "#FFED6F")) +
  theme_bw() +
  labs(title = "Crime Victim Ethnicity Distribution Graph", fill = "Offense Typ
e") +
  xlab(NULL) +
  ylab(NULL) +
  ylim(0, 170000) +
  theme(plot.title = element_text(size = 15, face = "bold", hjust = 0.5),
        axis.title = element_text(size = 10),
        axis.text = element_text(size = 10)) +
  geom_text(aes(label = paste0(round(Freq/sum(Freq)*100, 1), "%")),
            vjust = -0.5, size = 4)
```

```
df_offensetype <- as.data.frame(table(df$offensetype))

df_offensetype %>%
  ggplot(aes(x = "", y = Freq, fill = Var1)) +
  geom_col(width = 1) +
  coord_polar(theta = "y") +
  theme_void() +
  scale_fill_manual(values = c("#F8766D", "#00BFC4", "#FFC400")) +
  labs(title = "offense type pie chart", fill = "Offense Type") +
  xlab(NULL) +
  ylab(NULL) +
  geom_text(aes(label = paste0(round(Freq/sum(Freq)*100, 1), "%")),
            position = position_stack(vjust = 0.5), size = 4) +
  theme_void() +
  theme(plot.title = element_text(size = 15, face = "bold", hjust = 0.5),
        legend.position = "bottom",
        legend.direction = "horizontal",
        legend.title = element_text(size = 9, face = "bold"),
        legend.text = element_text(size = 8),
        axis.line = element_blank(),
        axis.text = element_blank(),
        axis.ticks = element_blank()) +
   geom_blank(aes(x = 0, y = 0, fill = NA), size = 1.5)
```

```
month.name <- c("January", "February", "March", "April", "May", "June",
                "July", "August", "September", "October", "November", "Decembe
r")
df$month <- df$month1ofoccurence
df$time <- ifelse(as.numeric(substr(df$time1ofoccurrence, 1, 2)) > 18:00 &
                    as.numeric(substr(df$time1ofoccurrence, 1, 2)) < 7, 1,2)

df %>% dplyr::select(offensetype, month, ctype, weaponused, sector, victimgende
r,
             victimrace, victimtype, time, division) -> carmer_df

convert_to_factor <- function(df) {
  df[] <- lapply(df, as.factor)
  return(df)
}
drop_na(carmer_df) -> carmer_df
convert_to_factor(carmer_df) -> carmer_df

cramers_v <- function(x, y) {
  tbl <- table(x, y)
  chi2 <- chisq.test(tbl)$statistic
  n <- sum(tbl)
  phi2 <- chi2 / n
  k <- min(ncol(tbl), nrow(tbl))
  return(sqrt(phi2 / (k - 1)))
}

variables <- colnames(carmer_df)
result_matrix <- matrix(0, nrow = length(variables), ncol = length(variables))
colnames(result_matrix) <- variables
rownames(result_matrix) <- variables

for (i in 1:length(variables)) {
  for (j in 1:length(variables)) {
    if (i != j) {
      result_matrix[i, j] <- cramers_v(carmer_df[[variables[i]]], carmer_df[[var
iables[j]]])
    }
  }
}

result_matrix <- as.data.frame(result_matrix)
heatmap_data <- as.data.frame(result_matrix)
heatmap_data$variables <- rownames(heatmap_data)
heatmap_data <- reshape2::melt(heatmap_data, id.vars = "variables")

ggplot(heatmap_data, aes(x = variables, y = variable, fill = value)) +
  geom_tile() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
```

```
  labs(x = "Variables", y = "Variables", title = "Cramér's V Heatmap") +
  scale_fill_gradient2(low = "white", mid = "blue", high = "red", midpoint = 0.
5, limits = c(0, 1)) +
  coord_equal()
```

```
month.name <- c("January", "February", "March", "April", "May", "June",
                "July", "August", "September", "October", "November", "Decembe
r")
df %>%
  dplyr::select(year, month1ofoccurence, victimrace) %>%
  dplyr::filter(!is.na(year) & !is.na(month1ofoccurence) & !is.na(victimrace)) %
>%
  mutate(month = month1ofoccurence) %>%
  dplyr::select(-month1ofoccurence) %>%
  group_by(year, month, victimrace) %>%
  summarise(n = n()) -> series_df

series_df$victimrace <- ifelse(series_df$victimrace == "American Indian or Alask
a Native", "etc",
                    ifelse(series_df$victimrace == "H", "etc",
                           ifelse(series_df$victimrace == "Hispanic or Latino",
"Latino",
                                  ifelse(series_df$victimrace == "Middle Easter
n", "etc",
                                         ifelse(series_df$victimrace == "NH", "e
tc",
                                                ifelse(series_df$victimrace ==
"Native Hawaiian/Pacific Islander", "etc",
                                                       ifelse(series_df$victimra
ce == "TEST", "etc",
                                                              ifelse(series_df$v
ictimrace == "Non-Hispanic or Latino", "Latino", series_df$victimrace))))))))
series_df$month <- match(series_df$month, month.name)
series_df$ym <- paste0(series_df$year,".",series_df$month)

series_df %>%
  group_by(ym, victimrace) %>%
  summarise(n = sum(n)) %>%
  ggplot(aes(ym, n, group = victimrace, color = victimrace)) +
  geom_line(size = 1.3) +
  scale_color_discrete(name = "Victim Race") +
  scale_x_discrete(breaks = seq(min(series_df$ym), max(series_df$ym), by = 1)) +
  labs(title = "Crime Victim Ethnicity Tendency Graph", fill = "Offense Type") +
  theme_bw() +
  xlab("Year-Month") +
  ylab(NULL) +
  theme(plot.title = element_text(size = 17, face = "bold", hjust = 0.5),
        axis.title = element_text(size = 10),
        axis.text = element_text(size = 10))
```

```
dallas_map <- get_map(location = "dallas", zoom = 10)

df %>% dplyr::select(division,
                     geo_lat,
                     geo_long,
                     sector) -> carmer_df

convert_to_factor <- function(df) {
  df[] <- lapply(df, as.factor)
  return(df)
}
convert_to_factor(carmer_df) -> carmer_df
drop_na(carmer_df) -> carmer_df
result <- kmodes(carmer_df, 7, iter.max = 10, fast = T, weighted = F)
carmer_df$cluster <- result$cluster

carmer_df$cluster <- as.factor(carmer_df$cluster)
carmer_df$sector <- as.factor(carmer_df$sector)
carmer_df$geo_long <-as.numeric(as.character(carmer_df$geo_long))
carmer_df$geo_lat <- as.numeric(as.character(carmer_df$geo_lat))
ggmap(dallas_map) +
  geom_point(data = carmer_df, aes(x = geo_long, y = geo_lat, color = cluster),
size = 0.1) +
  theme(legend.position = "none") -> new
ggmap(dallas_map) +
  geom_point(data = carmer_df, aes(x = geo_long, y = geo_lat, color = sector), s
ize = 0.1) +
  theme(legend.position = "none") -> old
grid.arrange(old, new, ncol = 2)
```