This is one of many R code and Python projects that i have done on my Freelance data analysis role

I will Print as Pdf for easy reading and easy access, but should you need the actual r code work, i will be glad to share.

PART 1: LOADING DATASET INTO R AND DATA INTEGRATION

In [1]:
```
# Load required packages
library(dplyr) # For data manipulation
library(ggplot2) # For data visualization
```

```
Warning message:
"package 'dplyr' was built under R version 4.2.3"

Attaching package: 'dplyr'


The following objects are masked from 'package:stats':

    filter, lag


The following objects are masked from 'package:base':

    intersect, setdiff, setequal, union


Warning message:
"package 'ggplot2' was built under R version 4.2.3"
```

In [2]:
```
library(tidyverse) # metapackage of all tidyverse packages
```

```
Warning message:
"package 'tidyverse' was built under R version 4.2.3"
Warning message:
"package 'tibble' was built under R version 4.2.3"
Warning message:
"package 'tidyr' was built under R version 4.2.3"
Warning message:
"package 'readr' was built under R version 4.2.3"
Warning message:
"package 'purrr' was built under R version 4.2.3"
Warning message:
"package 'stringr' was built under R version 4.2.3"
Warning message:
"package 'forcats' was built under R version 4.2.3"
Warning message:
"package 'lubridate' was built under R version 4.2.3"
── Attaching core tidyverse packages ──────────────────────── tidyverse
2.0.0 ──
✓ forcats   1.0.0     ✓ stringr   1.5.0
✓ lubridate 1.9.2     ✓ tibble    3.2.1
✓ purrr     1.0.1     ✓ tidyr     1.3.0
✓ readr     2.1.4
── Conflicts ────────────────────────────────────────── tidyverse_conflic
ts() ──
✖ dplyr::filter() masks stats::filter()
✖ dplyr::lag()    masks stats::lag()
ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all
conflicts to become errors
```

In [5]:

```
# Reading datasets into R for 2014
# Set up directory, year and months

data_path <- 'C:/Users/Data/Dataset - Assignment/'
setwd(data_path)

year <- '2014'
monthsAll <- c('January', 'February', 'March', 'April', 'May', 'June', 'Ju

# Read and combine data for 2014

offense_data <- list()

for (m in monthsAll) {
  data_file <- paste(data_path, year, '/principal_offence_category_', m, '_
  offense_data[[m]] <- read.csv(data_file, stringsAsFactors = FALSE,
         colClasses = c("Number.of.Theft.And.Handling.Unsuccessful" = "char
}

data <- bind_rows(offense_data, .id = "Month")

data$Month <- monthsAll[match(data$Month, monthsAll)]

# Add a year column
data$Year <- year


# Remove % sign from values
# Remove dashes and replace with null
data <- lapply(data, function(x) gsub("-", "", x, fixed = TRUE))
data <- lapply(data, function(x) gsub("%", "", x, fixed = TRUE))

# Convert back to a dataframe
data <- as.data.frame(data)

# Rename the first column
names(data)[2] <- "offense_region"
# Rearranging the datasets

data <- data[,c(2, 1, 53, 3:length(data))]

# Print the updated data
head(data)
```

A data.frame: 6 × 54

	offense_region	Month	Year	Number.of.Homicide.Convictions	Percentage.of.Homicide.C
	<chr>	<chr>	<chr>	<chr>	
1	National	January	2014	51	
2	Avon and Somerset	January	2014	0	
3	Bedfordshire	January	2014	0	
4	Cambridgeshire	January	2014	0	
5	Cheshire	January	2014	0	
6	Cleveland	January	2014	2	

In [6]:
```
# removing the last column
data <- data[,-54]
```

In [ ]:

```
# Reading datasets into R for 2016

library(readr)
data_apr <- cbind(Month="April",Year="2016",read_csv("C:Dataset - Assignmer
data_nov <- cbind(Month="November",Year="2016",read_csv("C:/Data/Dataset -
data_jan <- cbind(Month="January",Year="2016",read_csv("C:Dataset - Assignr
data_may <- cbind(Month="May",Year="2016",read_csv("C:/Dataset - Assignment
data_sep <- cbind(Month="September",Year="2016",read_csv("CDataset - Assigr
data_oct <- cbind(Month="October",Year="2016",read_csv("C:/Dataset - Assigr
data_Dec <- cbind(Month="December",Year="2016",read_csv("C://Dataset - Assi

# Integrating datasets
# combine data frames into one
data_16 <- rbind(data_jan, data_apr, data_may,data_june, data_july, data_au

# Renaming first column

data_16 <-data_16%>% rename(offense_region = ...1)

# Remove % sign from values
# Remove dashes and replace with null
data_16 <- lapply(data_16, function(x) gsub("-", "", x, fixed = TRUE))
data_16 <- lapply(data_16, function(x) gsub("%", "", x, fixed = TRUE))

# Convert back to a dataframe
data_16 <- as.data.frame(data_16)

# Rearranging the datasets

data_16 <- data_16[,c(3, 1, 2, 4:length(data_16))]

head(data_16)
```

In [9]:

```
# combing 2014 and 2016 data

data<- rbind(data,data_16)
head(data)
```

A data.frame: 6 × 53

	offense_region	Month	Year	Number.of.Homicide.Convictions	Percentage.of.Homicide.C
	<chr>	<chr>	<chr>	<chr>	
1	National	January	2014	51	
2	Avon and Somerset	January	2014	0	
3	Bedfordshire	January	2014	0	
4	Cambridgeshire	January	2014	0	
5	Cheshire	January	2014	0	
6	Cleveland	January	2014	2	

In [10]:

```
# change the data type from chr to numeric

data[-c(1,2,3)] <- lapply(data[-c(1,2,3)], as.numeric)

head(data)
```

```
Warning message in lapply(data[-c(1, 2, 3)], as.numeric):
"NAs introduced by coercion"
Warning message in lapply(data[-c(1, 2, 3)], as.numeric):
"NAs introduced by coercion"
Warning message in lapply(data[-c(1, 2, 3)], as.numeric):
"NAs introduced by coercion"
Warning message in lapply(data[-c(1, 2, 3)], as.numeric):
"NAs introduced by coercion"
Warning message in lapply(data[-c(1, 2, 3)], as.numeric):
"NAs introduced by coercion"
Warning message in lapply(data[-c(1, 2, 3)], as.numeric):
"NAs introduced by coercion"
Warning message in lapply(data[-c(1, 2, 3)], as.numeric):
"NAs introduced by coercion"
Warning message in lapply(data[-c(1, 2, 3)], as.numeric):
"NAs introduced by coercion"
Warning message in lapply(data[-c(1, 2, 3)], as.numeric):
"NAs introduced by coercion"
Warning message in lapply(data[-c(1, 2, 3)], as.numeric):
"NAs introduced by coercion"
Warning message in lapply(data[-c(1, 2, 3)], as.numeric):
"NAs introduced by coercion"
```

A data.frame: 6 × 53

	offense_region	Month	Year	Number.of.Homicide.Convictions	Percentage.of.Homicide.C
	<chr>	<chr>	<chr>	<dbl>	
1	National	January	2014	51	
2	Avon and Somerset	January	2014	0	
3	Bedfordshire	January	2014	0	
4	Cambridgeshire	January	2014	0	
5	Cheshire	January	2014	0	
6	Cleveland	January	2014	2	

PART 3: DATA CLEANING

Now that we have our integrated dataset, we can start cleaning and preparing it for analysis. We will need to check for missing values, outliers, and other data quality issues and fix them before proceeding with further analysis. We can use various techniques such as imputation, filtering, and aggregation to clean and refine the data. For instance, we can use the following R code to remove missing values and outliers:

In [11]:
```
# check for missing values
null_data<- sum(is.na(data))
null_data
```

829

In [12]:
```
# Removing missing values
data <- na.omit(data)

head(data)
```

A data.frame: 6 × 53

	offense_region	Month	Year	Number.of.Homicide.Convictions	Percentage.of.Homicic
	<chr>	<chr>	<chr>	<dbl>	
3	Bedfordshire	January	2014	0	
6	Cleveland	January	2014	2	
8	Derbyshire	January	2014	0	
11	Durham	January	2014	1	
13	Essex	January	2014	3	
15	GreaterManchester	January	2014	2	

In [59]:
```
# check for outliers
boxplot(numeric_data)
```

```
Error in boxplot(numeric_data): object 'numeric_data' not found
Traceback:

1. boxplot(numeric_data)
```

In [ ]:
```
# Removing outliers
data_no_outliers <-  numeric_data [which(numeric_data$`Numer of Homicide Co
numeric_data$`Numer of Criminal Damage Convictions` <
quantile(numeric_data$`Numer of Homicide Convictions` +
numeric_data$`Numer of Criminal Damage Convictions`, 0.99)),]

data_no_outliers
```

In [ ]:
```
# Removing outliers
data <-  numeric_data [which(numeric_data$`Numer of Homicide Convictions` -
numeric_data$`Numer of Criminal Damage Convictions` !=
max(numeric_data$`Numer of Homicide Convictions` +
numeric_data$`Numer of Criminal Damage Convictions`)),]
```

In [13]:
```
# check for duplicates
data_dublicate<- sum(duplicated(data))

data_dublicate
```

0

In [14]:
```
# Calculating conviction rate for each offence category

Conviction_Rate_Homicide <- (data$Number.of.Homicide.Convictions / (data$Nu
Conviction_Rate_offence_against_Person<- (data$Number.of.Offences.Against.T
Conviction_Rate_Sexual_offence<- (data$Number.of.Sexual.Offences.Conviction
Conviction_Rate_Burglary<- (data$Number.of.Burglary.Convictions/ (data$Numb
Conviction_Rate_Robbery<- (data$Number.of.Robbery.Convictions/ (data$Number
Conviction_Rate_Fraud.And.Forgery<- (data$Number.of.Fraud.And.Forgery.Convi
Conviction_Rate_Drugs.Offences<- (data$Number.of.Drugs.Offences.Convictions
Conviction_Rate_Public.Order.Offences<- (data$Number.of.Public.Order.Offenc
Conviction_Rate_All.Other.Offences..excluding.Motoring<- (data$Number.of.Al
Conviction_Rate_Motoring.Offences<- (data$Number.of.Motoring.Offences.Convi
```

In [15]:
```
# Integrating conviction rate to the datasets

data_with_conviction_rate <- cbind(data,Conviction_Rate_Homicide,Conviction
Conviction_Rate_Fraud.And.Forgery,Conviction_Rate_Drugs.Offences,Conviction

# Convert back to a dataframe
data_with_conviction_rate <- as.data.frame(data_with_conviction_rate)

head(data_with_conviction_rate, 5)
```

A data.frame: 5 × 63

	offense_region	Month	Year	Number.of.Homicide.Convictions	Percentage.of.Homicide.(
	<chr>	<chr>	<chr>	<dbl>	
3	Bedfordshire	January	2014	0	
6	Cleveland	January	2014	2	
8	Derbyshire	January	2014	0	
11	Durham	January	2014	1	
13	Essex	January	2014	3	

In [16]:
```
# Calculating conviction rate for each offence category

Homicide <- sum(data$Number.of.Homicide.Convictions / sum(data$Number.of.Ho
Offence_against_Person<- sum(data$Number.of.Offences.Against.The.Person.Con
Sexual_offence<- sum(data$Number.of.Sexual.Offences.Convictions / sum(data$
Burglary<- sum(data$Number.of.Burglary.Convictions/ sum(data$Number.of.Burg
Robbery<- sum(data$Number.of.Robbery.Convictions/ sum(data$Number.of.Robber
Fraud.And.Forgery<- sum(data$Number.of.Fraud.And.Forgery.Convictions/ sum(d
Drugs.Offences<- sum(data$Number.of.Drugs.Offences.Convictions/ sum(data$Nu
Public.Order.Offences<- sum(data$Number.of.Public.Order.Offences.Conviction
All.Other.Offences..excluding.Motoring<- sum(data$Number.of.All.Other.Offen
Motoring.Offences<- sum(data$Number.of.Motoring.Offences.Convictions/ sum(d
```

In [17]:
```
# conviction rate
Conviction_Rate<- rbind(Homicide, Offence_against_Person, Sexual_offence, E


Conviction_Rate
```

A matrix: 10 × 1 of type dbl

Homicide	81.19987
Offence_against_Person	77.11473
Sexual_offence	72.89363
Burglary	85.27652
Robbery	79.06317
Fraud.And.Forgery	85.84748
Drugs.Offences	93.62348
Public.Order.Offences	84.70702
All.Other.Offences..excluding.Motoring	84.77270
Motoring.Offences	85.66015

In [18]:
```


# Convert back to a dataframe
Conviction_Rate<- as.data.frame(Conviction_Rate)


Conviction_Rate
```

A data.frame: 10 × 1

	V1
	<dbl>
Homicide	81.19987
Offence_against_Person	77.11473
Sexual_offence	72.89363
Burglary	85.27652
Robbery	79.06317
Fraud.And.Forgery	85.84748
Drugs.Offences	93.62348
Public.Order.Offences	84.70702
All.Other.Offences..excluding.Motoring	84.77270
Motoring.Offences	85.66015

In [19]:
```
# Convert matrix to data frame
Conviction_Rate_df <- as.data.frame(Conviction_Rate)

# Rename columns
colnames(Conviction_Rate_df) <-  "Conviction_rate"

# Arrange in descending order by Conviction_rate
Conviction_Rate_df <- Conviction_Rate_df %>%
  arrange(desc(Conviction_rate))

  Conviction_Rate_df
```

A data.frame: 10 × 1

	Conviction_rate
	<dbl>
Drugs.Offences	93.62348
Fraud.And.Forgery	85.84748
Motoring.Offences	85.66015
Burglary	85.27652
All.Other.Offences..excluding.Motoring	84.77270
Public.Order.Offences	84.70702
Homicide	81.19987
Robbery	79.06317
Offence_against_Person	77.11473
Sexual_offence	72.89363

Copy the above data set into excel then add Header to the first column as Offence_Category and Save as csv

Then load back to R, as shown below

In [20]:
```
data_conviction <- read_csv("CData/Conviction Rate.csv")

data_conviction
```

```
Rows: 10 Columns: 2
── Column specification ────────────────────────────────────────────────
────────
Delimiter: ","
chr (1): Offence_Category
dbl (1): Conviction_Rate

ℹ Use `spec()` to retrieve the full column specification for this data.
ℹ Specify the column types or set `show_col_types = FALSE` to quiet this m
essage.
```

A spec_tbl_df: 10 × 2

Offence_Category	Conviction_Rate
<chr>	<dbl>
Drugs.Offences	93.64712
Fraud.And.Forgery	85.89617
All.Other.Offences..excluding.Motoring	85.30524
Motoring.Offences	84.96977
Burglary	84.90678
Public.Order.Offences	84.61201
Homicide	81.13839
Robbery	78.84134
Offence_against_Person	77.49556
Sexual_offence	72.99429

In [21]:

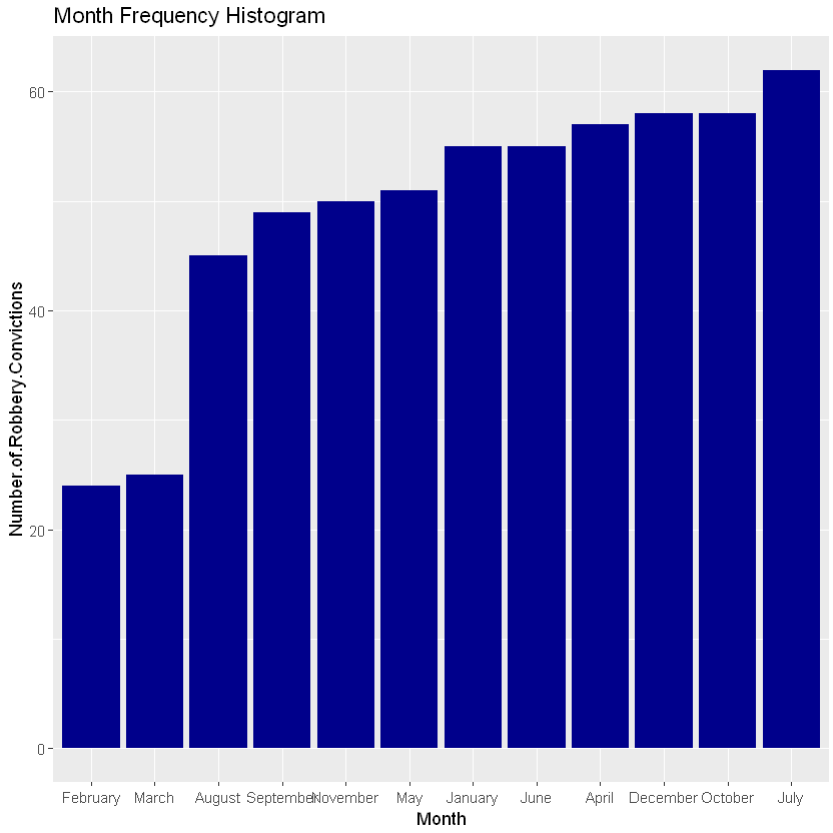
```
conviction_freq <- as.data.frame(table(data$Month))
colnames(conviction_freq) <- c("Month","Number.of.Robbery.Convictions")
conviction_freq
```

A data.frame: 12 × 2

Month	Number.of.Robbery.Convictions
<fct>	<int>
April	57
August	45
December	58
February	24
January	55
July	62
June	55
March	25
May	51
November	50
October	58
September	49

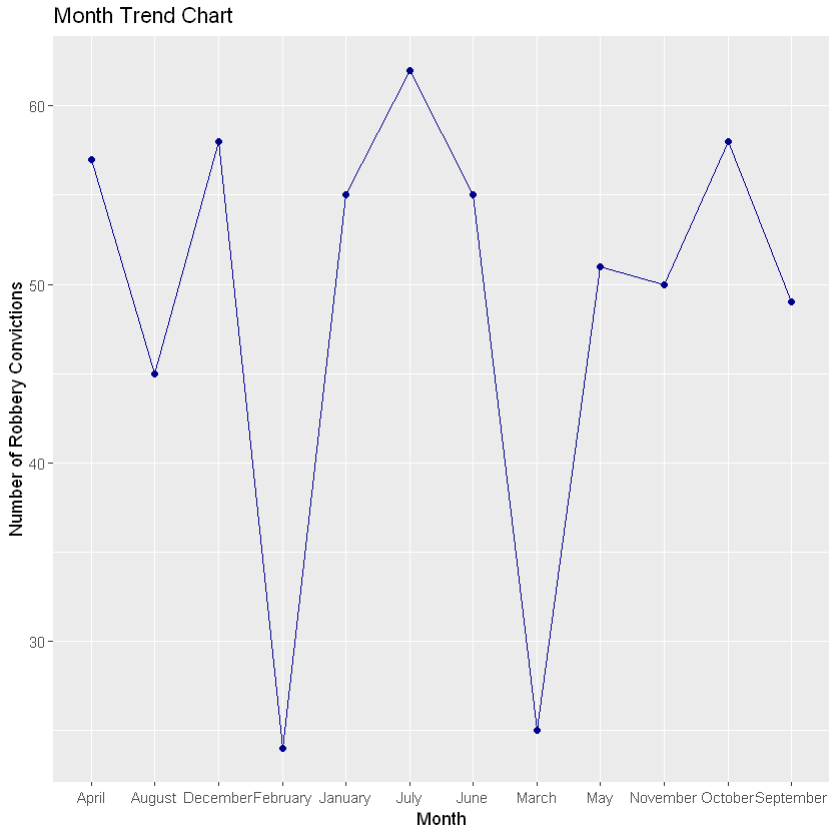
In [22]:

```
# Plotting a frequency histogram of offences

library(forcats)

ggplot(conviction_freq , aes(x = fct_reorder(Month, Number.of.Robbery.Convi
  geom_bar(stat="identity", fill="dark blue") +

  labs(x="Month", y="Number.of.Robbery.Convictions") +
  ggtitle("Month Frequency Histogram")
```



In [23]:

```
# Load the necessary libraries
library(ggplot2)
library(forcats)

# Create the plot with trend line
ggplot(conviction_freq, aes(x = Month, y = Number.of.Robbery.Convictions, g
  geom_line(color = "dark blue") +
  geom_point(color = "dark blue") +
  labs(x= "Month", y= "Number of Robbery Convictions") +
  ggtitle("Month Trend Chart")
```



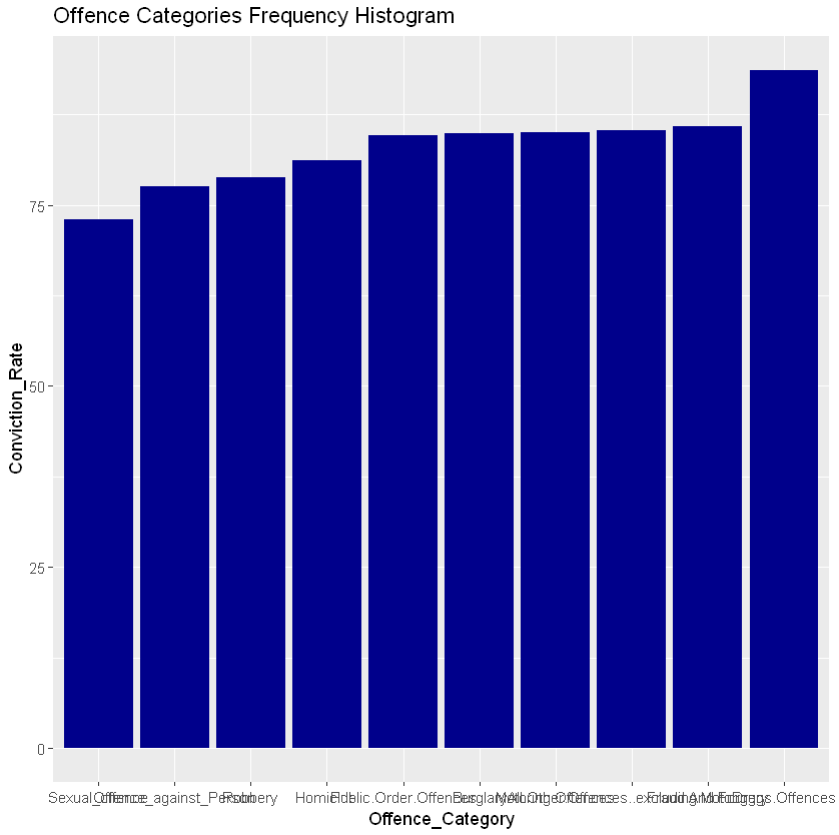
In [25]:

```
library(plotly)

plot_ly(data_conviction, labels = ~data_conviction$Offence_Category, value
```

```
Warning message:
"package 'plotly' was built under R version 4.2.3"

Attaching package: 'plotly'


The following object is masked from 'package:ggplot2':

    last_plot


The following object is masked from 'package:stats':

    filter


The following object is masked from 'package:graphics':

    layout
```

In [26]:
```
plot_ly(data, labels = ~data$Year, values = ~data$Number.of.Robbery.Convict
```

In [27]:
```
# Plotting a frequency histogram of offences

library(forcats)

ggplot(data_conviction , aes(x = fct_reorder(Offence_Category, Conviction_R
  geom_bar(stat="identity", fill="dark blue") +

  labs(x="Offence_Category", y="Conviction_Rate") +
  ggtitle("Offence Categories Frequency Histogram")
```



In [ ]:
```
colnames(data)
```

In [28]:

```
Total_Conviction<- (data$Number.of.Homicide.Convictions + data$Number.of.O
 data$Number.of.Robbery.Conviction + data$Number.of.Theft.And.Handling.Conv
 data$Number.of.Public.Order.Offences.Convictions + data$Number.of.All.Othe
Total_Conviction
```

```
430  966  831  596 1296 2642  614 1203 1608  766  532 1556 1963  847 1442
1286  562 1458 2178 1988  387  767  892  332  471  490  337 1224  422  985
1146 1510  816 1762  430  619  515  906 2307 1157  478 1553  779 2100 1674
1767  491  844  577 1079  440 1090 1163  844  839 1041 1497  874  448 1614
 486 1606  558  866 1446 1002 1854 2179 1710  374  984  827  738 1026 2290
1215  789 1064 1289  655 1315  636  443 1381  579  422  839 1045  492 1149
 742 1995 1734  326 1010  307  763  662  804 1102  728 1029 1375 1333  529
 892 1783  723  868 1734  784 1895 1548 1027  333  871  754  464  729 2214
 373  734 1268 1558 1471  632  808 1480 1000 1428  255  690 1921 1485 1340
 326  803  747  754  370 2440  482 1380  670  985 1216 1622  618 1466  686
1583  590  606 1048 2404 1265  755  534 1304 2278 1757  934  329  367  566
 915  241 2234  444  807  960 1057  462  553  859  934  544  820 1363 1819
1625 1139  370 1177  271 2343  875 1242 1468  551  630  476  635  659  985
1187  774 1187  765 2046 ...  597  429 1386 1010 1149  920  792 1174  336
 793 2781 1535  392 36418  979  292  890  645  690  397  383  257  916  284
1693  352  880  658  612  873  912  516 1103 6178 1198  529  486  894 1296  923
 761  380  397  771 1121  279 2380 1604 36294  405  904  393  548  707  459  923
1613  351  871  672  635  847  995  465  580 6006  353 1087 1254  796 1228
 824 2366 37005 1054  315  375  535  702  361  325  828 1723  909  639  879
 939  532 1194 6230  562 1083  535  832 1226  744  812 1262  859 2515 1596
 375 35925 1312  433  874  646  408  348  260  861 1636  359  904  710  658  859
 911  563  424 1031 5881  659  383 1070  550  507  871 1257  822  674 1247  773
2440 1466 37358  314  421  948  540  722  409  903 1578  399  880  705  673
 925 1033  497  528 1258 6263 1157  500  942 1222  952  342 1149  846 2550
1632 32590  930  297  415  735  619  378  434  717  360  230  817  222 1484  375
 795  544  662  749  878  435  455  977 5449  284 1020  440  804 1083  777  347
1075  275  717 2084 1349
```

In [30]:
```
data<- rbind(data,Total_Conviction)
data<- as.data.frame(data)
head(data)
```

```
Warning message in rbind(deparse.level, ...):
"number of columns of result, 53, is not a multiple of vector length 589
of arg 2"
```

A data.frame: 6 × 53

	offense_region	Month	Year	Number.of.Homicide.Convictions	Percentage.of.Homicic
	<chr>	<chr>	<chr>	<dbl>	
3	Bedfordshire	January	2014	0	
6	Cleveland	January	2014	2	
8	Derbyshire	January	2014	0	
11	Durham	January	2014	1	
13	Essex	January	2014	3	
15	GreaterManchester	January	2014	2	

In [31]:
```
# Calculating Total Number of offences for each offence category

Conviction_Homicide <- sum(data$Number.of.Homicide.Convictions )
Conviction_offence_against_Person<- sum(data$Number.of.Offences.Against.The
Conviction_Sexual_offence<- sum(data$Number.of.Sexual.Offences.Convictions
Conviction_Burglary<- sum(data$Number.of.Burglary.Convictions)
Conviction_Robbery<- sum(data$Number.of.Robbery.Convictions)
Conviction_Fraud.And.Forgery<- sum(data$Number.of.Fraud.And.Forgery.Convict
Conviction_Drugs.Offences<- sum(data$Number.of.Drugs.Offences.Convictions)
Conviction_Public.Order.Offences<- sum(data$Number.of.Public.Order.Offences
Conviction_All.Other.Offences..excluding.Motoring<- sum(data$Number.of.All
Conviction_Motoring.Offences<- sum(data$Number.of.Motoring.Offences.Convict
```

In [32]:

```
Total_Number_of_Offences<- rbind(Conviction_Homicide,Conviction_offence_aga
Conviction_Burglary,Conviction_Robbery,Conviction_Fraud.And.Forgery,Convict
Conviction_All.Other.Offences..excluding.Motoring,Conviction_Motoring.Offe

Total_Number_of_Offences<- as.data.frame(Total_Number_of_Offences)

# Rename columns
colnames(Total_Number_of_Offences) <-  "Total_Number_of_Offences"

# Arrange in descending order by Conviction_rate
Total_Number_of_Offences <- Total_Number_of_Offences %>%
  arrange(desc(Total_Number_of_Offences))


Total_Number_of_Offences
```

A data.frame: 10 × 1

	Total_Number_of_Offences
	<dbl>
Conviction_offence_against_Person	257564
Conviction_Motoring.Offences	194742
Conviction_Drugs.Offences	104324
Conviction_Public.Order.Offences	89353
Conviction_Burglary	35354
Conviction_All.Other.Offences..excluding.Motoring	35081
Conviction_Sexual_offence	28487
Conviction_Fraud.And.Forgery	23994
Conviction_Robbery	14002
Conviction_Homicide	3723

Copy the above data set into excel then add Header to the first column as Offence_Category and Save as csv

Then load back to R, as shown below

In [33]:
```
Total_conviction <- read_csv("C:/Users/Vincent/Desktop/BI Training/Project/

Total_conviction
```

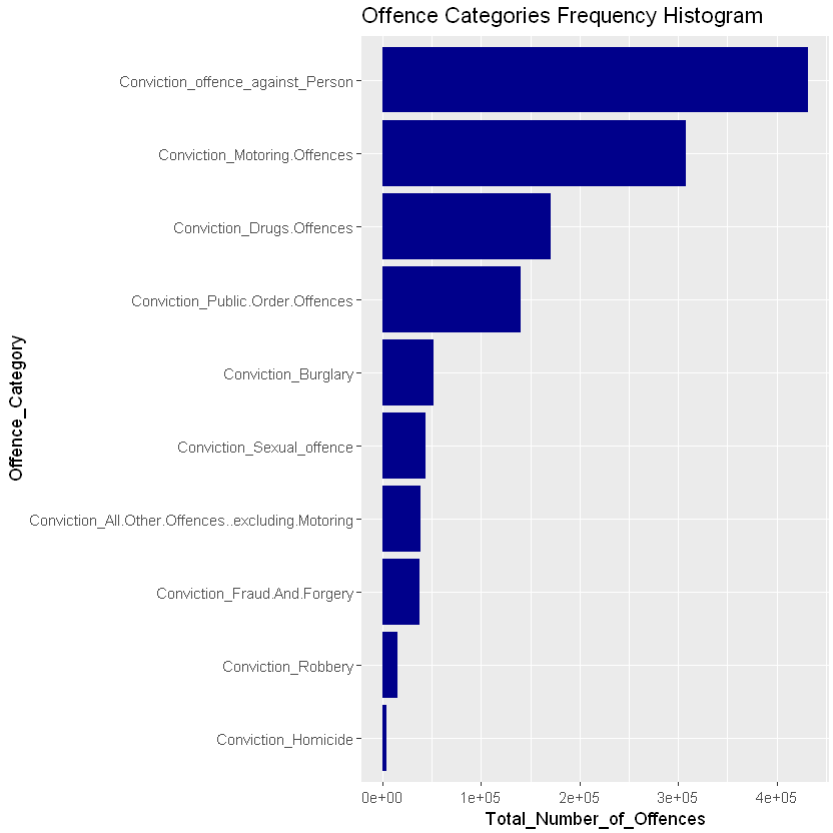
```
Rows: 10 Columns: 2
── Column specification ────────────────────────────────────────────────────
────────
Delimiter: ","
chr (1): Offence_Category
dbl (1): Total_Number_of_Offences

ℹ Use `spec()` to retrieve the full column specification for this data.
ℹ Specify the column types or set `show_col_types = FALSE` to quiet this m
essage.
```

A spec_tbl_df: 10 × 2

Offence_Category	Total_Number_of_Offences
<chr>	<dbl>
Conviction_offence_against_Person	432092
Conviction_Motoring.Offences	307786
Conviction_Drugs.Offences	170552
Conviction_Public.Order.Offences	140615
Conviction_Burglary	51462
Conviction_Sexual_offence	43690
Conviction_All.Other.Offences..excluding.Motoring	38790
Conviction_Fraud.And.Forgery	37510
Conviction_Robbery	15732
Conviction_Homicide	4362

In [34]:

```
# Plotting a frequency histogram of offences

library(forcats)

ggplot(Total_conviction , aes(x = fct_reorder(Offence_Category, Total_Numbe
  geom_bar(stat="identity", fill="dark blue") +
       coord_flip() +
  labs(x="Offence_Category", y="Total_Number_of_Offences") +
  ggtitle("Offence Categories Frequency Histogram")
```



In [35]:
```
# Load requiredlibraries
#webshot::install_phantomjs()
library(cowplot)
library(ggplot2)
library(gtsummary)
library(factoextra)
library(plotly)
```

```
Warning message:
"package 'cowplot' was built under R version 4.2.3"

Attaching package: 'cowplot'


The following object is masked from 'package:lubridate':

    stamp


Warning message:
"package 'gtsummary' was built under R version 4.2.3"
#BlackLivesMatter

Warning message:
"package 'factoextra' was built under R version 4.2.3"
Welcome! Want to learn more? See two factoextra-related books at https://
goo.gl/ve3WBa (https://goo.gl/ve3WBa)
```

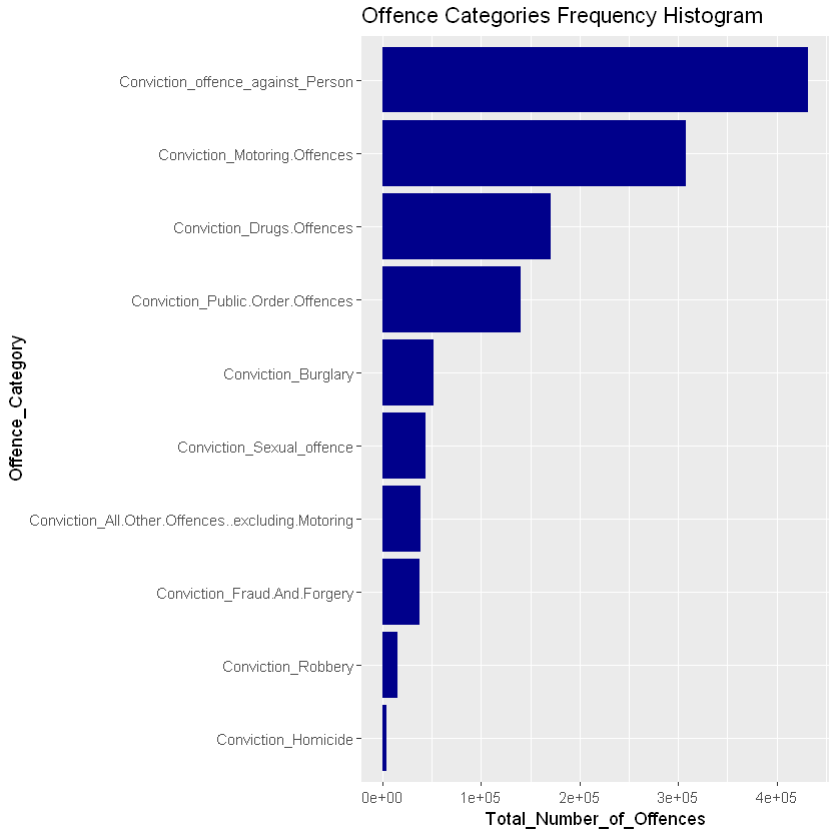
In [120]:

```
library(plotly)
plot_ly(Total_conviction, labels = ~Total_conviction$Offence_Category, valu
```

In [122]:

```
    library(treemap)

treemap(Total_conviction,
        index = "Offence_Category",
        vSize = "Total_Number_of_Offences",
        type = "value",
        title = "Total Convictions"
  )
```

In [123]:
```
# Plotting a frequency histogram of offences

library(forcats)

ggplot(Total_conviction , aes(x = fct_reorder(Offence_Category, Total_Numbe
  geom_bar(stat="identity", fill="dark blue") +
  coord_flip() +
  labs(x="Offence_Category", y="Total_Number_of_Offences") +
  ggtitle("Offence Categories Frequency Histogram")
```



In [74]:
```
data<- cbind(data,Total_Conviction)

data<- as.data.frame(data)
head(data, 5)
```

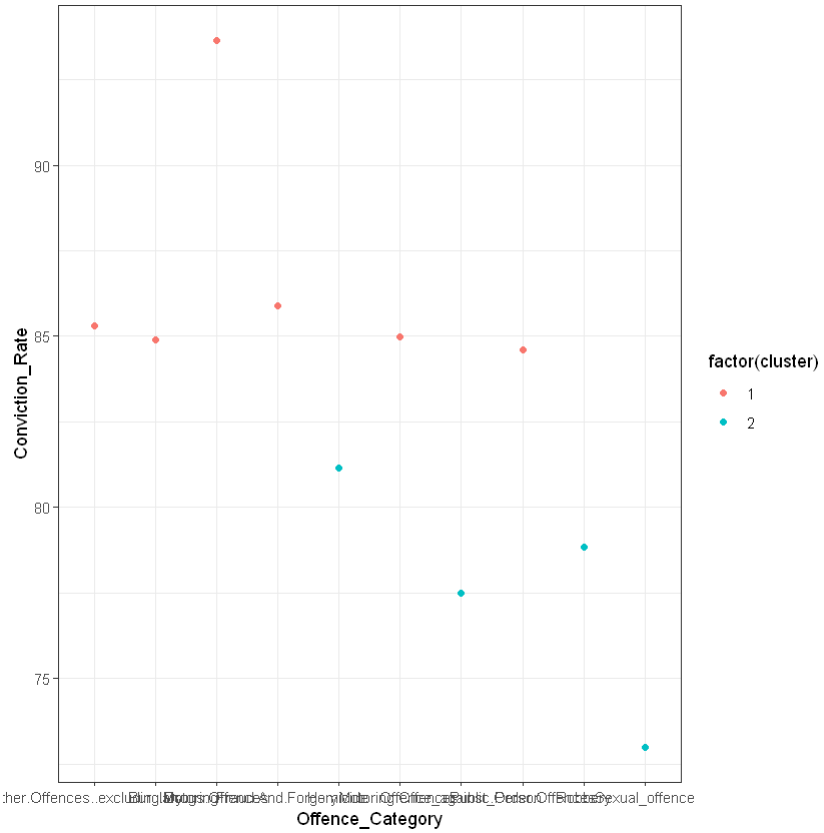
A data.frame: 5 × 54

	Month	offense_region	Number.of.Homicide.Convictions	Percentage.of.Homicide.Convictio
	<chr>	<chr>	<dbl>	<d
3	january	Bedfordshire	0	
6	january	Cleveland	2	
8	january	Derbyshire	0	
11	january	Durham	1	
13	january	Essex	3	

In [125]:

```
# Create a matrix of the features you want to use for clustering
features <- data_conviction[,2]

# Normalize the features
features_norm <- scale(features)

# Compute the distance matrix
dist_mat <- dist(features_norm)

# Apply hierarchical clustering with two clusters
hc <- hclust(dist_mat, method = "ward.D2")
clusters <- cutree(hc, k = 2)
clusters

# Add the cluster labels to the data
data_conviction$cluster <- clusters

# View the resulting clusters
ggplot(data_conviction, aes(x = Offence_Category, y = Conviction_Rate, colo
  geom_point() +
  theme_bw()
```

1 1 1 1 1 1 2 2 2 2



In [76]:
```
head(data, 5)
```

A data.frame: 5 × 54

	Month	offense_region	Number.of.Homicide.Convictions	Percentage.of.Homicide.Convictic
	<chr>	<chr>	<dbl>	<d
3	january	Bedfordshire	0	
6	january	Cleveland	2	
8	january	Derbyshire	0	
11	january	Durham	1	
13	january	Essex	3	

In [126]:
```
# select only the numeric columns in the data
numeric_data <- data[, sapply(data, is.numeric)]


head(numeric_data)
```

A data.frame: 6 × 50

	Number.of.Homicide.Convictions	Percentage.of.Homicide.Convictions	Number.of.Homicide.
	<dbl>	<dbl>	
3	0	0.0	
6	2	40.0	
8	0	0.0	
11	1	100.0	
13	3	100.0	
15	2	66.7	

PART 4: DATA ANALYSIS

Descriptive analytics

In [127]:
```
# Descriptive analytics

# summary statistics
summary(data)
```

```
 offense_region        Month               Year
 Length:818         Length:818         Length:818
 Class :character   Class :character   Class :character
 Mode  :character   Mode  :character   Mode  :character



 Number.of.Homicide.Convictions Percentage.of.Homicide.Convictions
 Min.   :  0.000                Min.   :   0.00
 1st Qu.:  1.000                1st Qu.:  75.00
 Median :  2.000                Median : 100.00
 Mean   :  6.061                Mean   :  86.96
 3rd Qu.:  4.000                3rd Qu.: 100.00
 Max.   :596.000                Max.   :1296.00
 Number.of.Homicide.Unsuccessful Percentage.of.Homicide.Unsuccessful
 Min.   :   0.000                 Min.   :  0.00
 1st Qu.:   0.000                 1st Qu.:  0.00
 Median :   0.000                 Median :  0.00
 Mean   :   4.469                 Mean   : 15.25
```

In [128]:
```
# correlation matrix
head(cor(numeric_data),5)
```

A matrix: 5 × 50 of type dbl

	Number.of.Homicide.Convictions	Percenta
Number.of.Homicide.Convictions	1.0000000	
Percentage.of.Homicide.Convictions	0.6844596	
Number.of.Homicide.Unsuccessful	0.8188163	
Percentage.of.Homicide.Unsuccessful	0.5327220	
Number.of.Offences.Against.The.Person.Convictions	0.5939913	

In [129]:
```
# hypothesis testing
# Example: test if the percentage of successful homicide cases is higher th

t.test(data$Percentage.of.Homicide.Convictions, data$Percentage.of.Homicide
```

```
        Welch Two Sample t-test

data:  data$Percentage.of.Homicide.Convictions and data$Percentage.of.Hom
icide.Unsuccessful
t = 35.223, df = 1409.8, p-value < 2.2e-16
alternative hypothesis: true difference in means is greater than 0
95 percent confidence interval:
 68.36028      Inf
sample estimates:
mean of x mean of y
 86.96198  15.25073
```

Next, we will develop prediction models using linear regression, clustering and classification algorithms. For instance, we can use multiple regression to predict the number of cases dismissed based on the number of cases prosecuted and convicted. We can use the following R code to develop our linear regression model

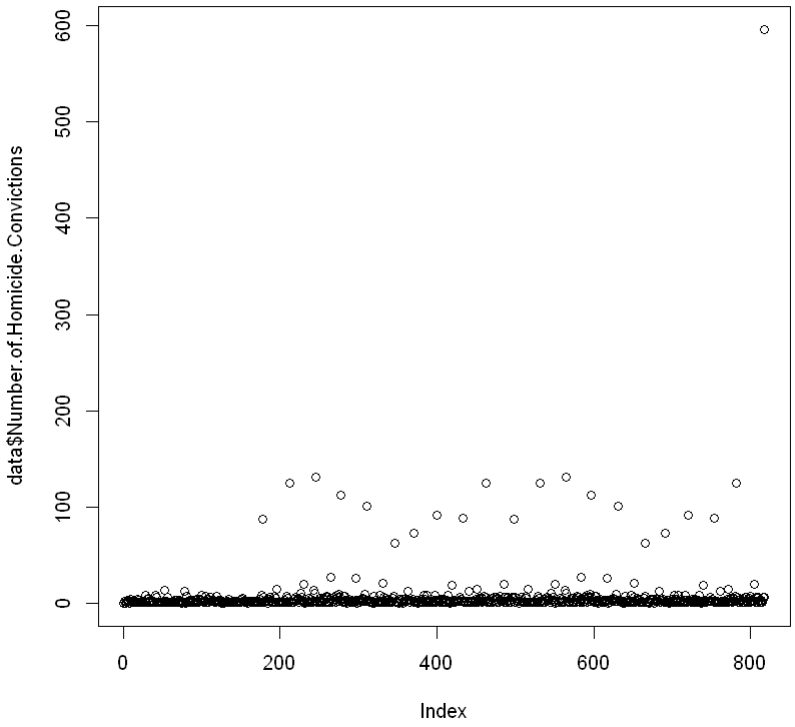
In [136]:
```
# linear regression model
model_lm <- lm(data$Number.of.Homicide.Convictions ~ data$Number.of.Offence
summary(model_lm)
```

```
Call:
lm(formula = data$Number.of.Homicide.Convictions ~ data$Number.of.Offence
s.Against.The.Person.Convictions +
    data$Number.of.Sexual.Offences.Convictions, data = data)

Residuals:
    Min      1Q  Median      3Q     Max
-50.294  -2.455   0.454   2.378  56.929

Coefficients:
                                                              Estimate Std. Er
ror
(Intercept)                                                  0.4313668  0.2699
157
data$Number.of.Offences.Against.The.Person.Convictions -0.0279199  0.0005
157
data$Number.of.Sexual.Offences.Convictions              0.3691534  0.0048
436
                                                            t value Pr(>|t|)
(Intercept)                                                   1.598     0.11
data$Number.of.Offences.Against.The.Person.Convictions -54.136   <2e-16 *
**
data$Number.of.Sexual.Offences.Convictions              76.214   <2e-16 *
**
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 7.336 on 815 degrees of freedom
Multiple R-squared:  0.9204,    Adjusted R-squared:  0.9202
F-statistic:  4710 on 2 and 815 DF,  p-value: < 2.2e-16
```

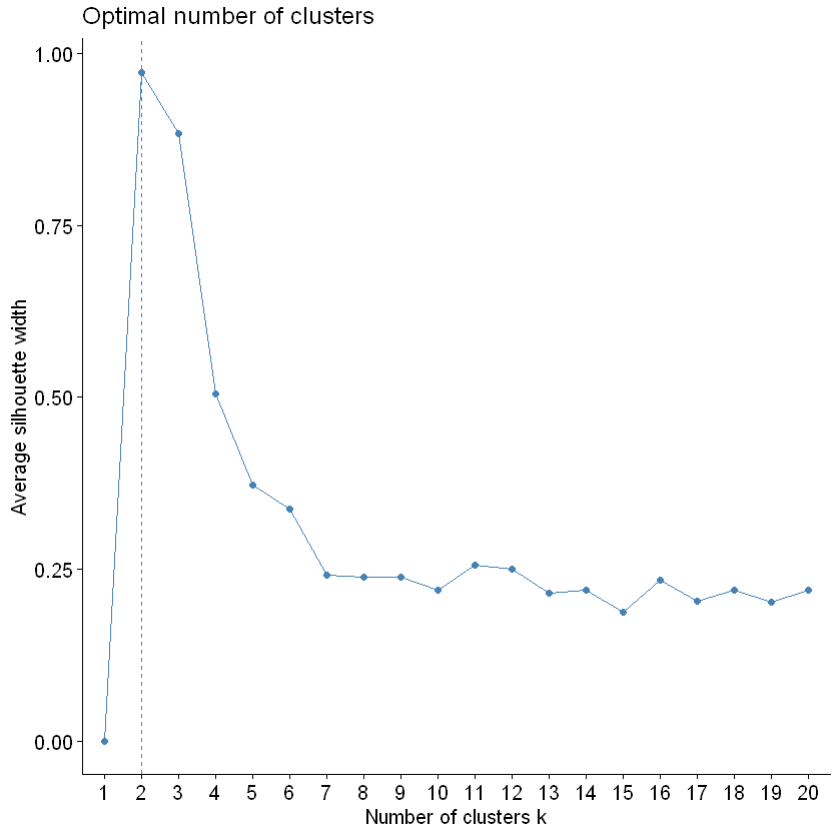
In [137]:
```
# linear regression model
model <- lm(data_with_conviction_rate$Conviction_Rate_Drugs.Offences ~ data
summary(model)
```

```
Call:
lm(formula = data_with_conviction_rate$Conviction_Rate_Drugs.Offences ~
    data_with_conviction_rate$Number.of.Drugs.Offences.Convictions +
        data_with_conviction_rate$Number.of.Drugs.Offences.Unsuccessful,
    data = data_with_conviction_rate)

Residuals:
    Min      1Q  Median      3Q     Max
-11.819  -1.507  -0.224   1.823   8.098

Coefficients:
                                                                  Estimate
(Intercept)                                                      94.619672
data_with_conviction_rate$Number.of.Drugs.Offences.Convictions    0.021955
data_with_conviction_rate$Number.of.Drugs.Offences.Unsuccessful -0.326724
                                                                 Std. Erro
r
(Intercept)                                                        0.10170
1
data_with_conviction_rate$Number.of.Drugs.Offences.Convictions     0.00117
5
data_with_conviction_rate$Number.of.Drugs.Offences.Unsuccessful    0.01697
6
                                                                 t value
(Intercept)                                                       930.37
data_with_conviction_rate$Number.of.Drugs.Offences.Convictions     18.69
data_with_conviction_rate$Number.of.Drugs.Offences.Unsuccessful   -19.25
                                                                 Pr(>|t|)
(Intercept)                                                        <2e-16
***
data_with_conviction_rate$Number.of.Drugs.Offences.Convictions     <2e-16
***
data_with_conviction_rate$Number.of.Drugs.Offences.Unsuccessful    <2e-16
***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.76 on 814 degrees of freedom
Multiple R-squared:  0.318,     Adjusted R-squared:  0.3163
F-statistic: 189.7 on 2 and 814 DF,  p-value: < 2.2e-16
```

In [138]:
```
plot(data$Number.of.Homicide.Convictions)
```



In [139]:
```
# Plot clustering indices to identify optimal number of clusters
fviz_nbclust(numeric_data, kmeans, method = "silhouette", k.max = 20)
```



The optimal value for k based on the silhoutte score above is 2. As such, the final k-means model will be fit using 2 clusters.

In [140]:
```
# Compute k-means clustering with k = 2
set.seed(123)
final <- kmeans(numeric_data, 2, nstart = 25)

final
```

```
K-means clustering with 2 clusters of sizes 798, 20

Cluster means:
  Number.of.Homicide.Convictions Percentage.of.Homicide.Convictions
1                       3.709273                           87.10789
2                      99.900000                           81.14000
  Number.of.Homicide.Unsuccessful Percentage.of.Homicide.Unsuccessful
1                        3.994987                            15.16028
2                       23.400000                            18.86000
  Number.of.Offences.Against.The.Person.Convictions
1                                          276.9586
2                                        10614.1000
  Percentage.of.Offences.Against.The.Person.Convictions
1                                               80.9609
2                                               78.0000
  Number.of.Offences.Against.The.Person.Unsuccessful
1                                           83.2406
2                                         2990.9000
  Percentage.of.Offences.Against.The.Person.Unsuccessful
```

In [93]:
```
data$Cluster <- final$cluster
#observations in each cluster
cluster_counts <- table(final$cluster)
#
p <- data.frame(cluster = names(cluster_counts),
count = as.numeric(cluster_counts))
plot_ly(p, labels = ~cluster, values = ~count, type = "pie")
```

Most of the cells were assigned to cluster 1 (96.7%)

In [141]:

```
data %>%
select(Cluster, Number.of.Homicide.Convictions ,
Number.of.Homicide.Unsuccessful ) %>%
tbl_summary(by = Cluster)
```

```
Error in `select()`:
! Can't subset columns that don't exist.
✕ Column `Cluster` doesn't exist.
Traceback:

1. data %>% select(Cluster, Number.of.Homicide.Convictions, Number.of.H
omicide.Unsuccessful) %>%
 .     tbl_summary(by = Cluster)
2. tbl_summary(., by = Cluster)
3. data %>% ungroup()
4. ungroup(.)
5. select(., Cluster, Number.of.Homicide.Convictions, Number.of.Homicid
e.Unsuccessful)
6. select.data.frame(., Cluster, Number.of.Homicide.Convictions,
 .     Number.of.Homicide.Unsuccessful)
7. tidyselect::eval_select(expr(c(...)), data = .data, error_call = err
or_call)
8. eval_select_impl(data, names(data), as_quosure(expr, env), include =
include,
```

You need to copy this output and save as HTML File so as to see the table

From the summary table above, we note that clusters 1 and 2 predominantly contain cells from Number.of.Homicide.Convictions and Number.of.Homicide.Unsuccessful

In [95]:
```
cols <- c(2:28)

data %>%
select(Cluster, cols ) %>%
tbl_summary(by = Cluster)
```

```
Warning message:
"Using an external vector in selections was deprecated in tidyselect 1.
1.0.
i Please use `all_of()` or `any_of()` instead.
  # Was:
  data %>% select(cols)

  # Now:
  data %>% select(all_of(cols))

See <https://tidyselect.r-lib.org/reference/faq-external-vector.html>."
```

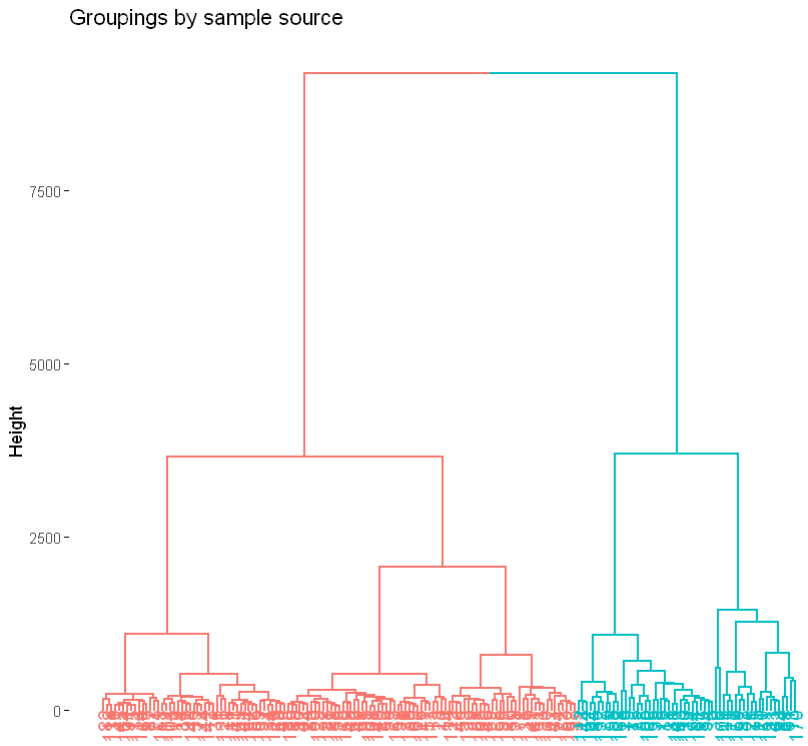
```
<div id="hvvgpqvrqb" style="padding-left:0px;padding-right:0px;padding-
top:10px;padding-bottom:10px;overflow-x:auto;overflow-y:auto;width:aut
o;height:auto;">
  <style>#hvvgpqvrqb table {
  font-family: system-ui, 'Segoe UI', Roboto, Helvetica, Arial, sans-se
rif, 'Apple Color Emoji', 'Segoe UI Emoji', 'Segoe UI Symbol', 'Noto Co
lor Emoji';
  -webkit-font-smoothing: antialiased;
```

You need to copy this output and save as HTML File so as to see the table

From the summary table above, we note that clusters 1 predominantly contain cells from Percentage.of.Homicide.Convictions,Percentage.of.Offences.Against.The.Person.Convictions, Percentage.of.Sexual.Offences.Convictions among others while cluster 2 contains predomenantly Number.of.Homicide.Convictions, Number.of.Homicide.Unsuccessful among others

In [96]:
```
# Perform hierarchical clustering
hc <- hclust(dist(numeric_data))
hc <- hcut(dist(numeric_data) , k =2)
hc$labels <- data$cols
# Plot dendrogram with sample_source labels
fviz_dend(hc, main = "Groupings by sample source")
```

```
Warning message:
"The `<scale>` argument of `guides()` cannot be `FALSE`. Use "none" inste
ad as
of ggplot2 3.3.4.
ℹ The deprecated feature was likely used in the factoextra package.
  Please report the issue at <https://github.com/kassambara/factoextra/is
sues>."
```



In [97]:
```
table(hc$cluster)
```

```
  1   2
121  56
```

In [98]:

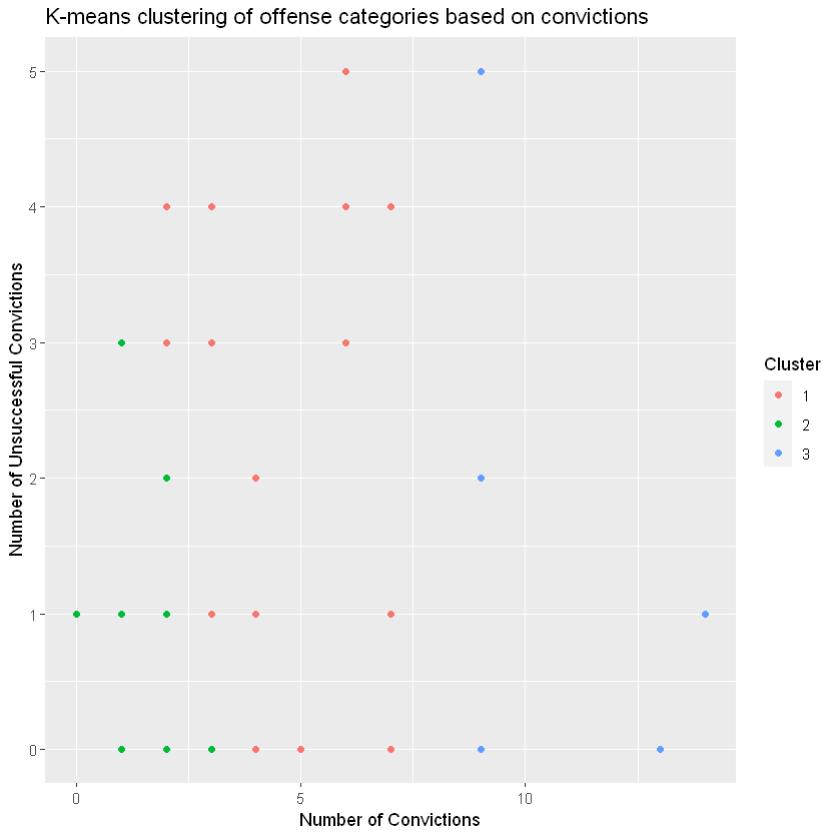
```
# clustering model

library(cluster)
model_km <- kmeans(data[c("Number.of.Homicide.Convictions", "Number.of.Moto
table(model_km$cluster)
```

```
 1  2  3
60  7 110
```

In [99]:

```
# clustering model

library(cluster)
model_km <- kmeans(data[c("Number.of.Homicide.Convictions", "Number.of.Hom
table(model_km$cluster)

# Add cluster assignment to the data
data$cluster <- model_km$cluster

# Plot the data with color-coded clusters
ggplot(data, aes(x = data$Number.of.Homicide.Convictions    , y = data$Numb
  geom_point() +
  ggtitle("K-means clustering of offense categories based on convictions")
  xlab("Number of Convictions") +
  ylab("Number of Unsuccessful Convictions") +
  scale_color_discrete(name = "Cluster")
```

```
  1   2   3
 23 148   6
```



In [102]:

```
# classification model
# Classification with decision tree model:
library(rpart)
model_tree <- rpart(data$offense_region  ~ data$Number.of.Offences.Against
printcp(model_tree)
plot(model_tree)

# Predictive model

new_data <- data.frame(Number.of.Offences.Against.The.Person.Convictions =
                       Number.of.Sexual.Offences.Convictions = data$Number

prediction <- predict(model_tree, newdata = new_data, type = "class")
prediction <- as.data.frame(prediction)
prediction
```

```
Classification tree:
rpart(formula = data$offense_region ~ data$Number.of.Offences.Against.T
he.Person.Convictions +
    data$Number.of.Sexual.Offences.Convictions, data = data)

Variables actually used in tree construction:
[1] data$Number.of.Offences.Against.The.Person.Convictions
[2] data$Number.of.Sexual.Offences.Convictions

Root node error: 169/177 = 0.9548

n= 177

        CP nsplit rel error  xerror      xstd
1 0.047337      0   1.00000 1.04734 0.0000000
2 0.041420      1   0.95266 1.04142 0.0059004
3 0.035503      3   0.86982 0.98817 0.0181754
4 0.029586      5   0.79882 0.94083 0.0237937
```

In [104]:

```
# classification model
# Classification with decision tree model:
library(rpart)
model_tree <- rpart(data$offense_region  ~ data$Number.of.Homicide.Convict
printcp(model_tree)
plot(model_tree)

# Predictive model

new_data <- data.frame(Number.of.Homicide.Convictions = data$Number.of.Hom
                       Number.of.Homicide.Unsuccessful = data$Number.of.Hor

prediction <- predict(model_tree, newdata = new_data, type = "class")
prediction <- as.data.frame(prediction)
prediction
```

```
Classification tree:
rpart(formula = data$offense_region ~ data$Number.of.Homicide.Convictio
ns +
    data$Number.of.Homicide.Unsuccessful, data = data)

Variables actually used in tree construction:
[1] data$Number.of.Homicide.Convictions

Root node error: 169/177 = 0.9548

n= 177

        CP nsplit rel error xerror xstd
1 0.011834      0    1.0000 1.0473    0
2 0.010000      3    0.9645 1.0473    0
```

A data.frame: 177 × 1

In [106]:
```
data<-cbind(data,prediction)
head(data)
```

A data.frame: 6 × 58

	Month	offense_region	Number.of.Homicide.Convictions	Percentage.of.Homicide.Convi
	<chr>	<chr>	<dbl>	
3	january	Bedfordshire	0	
6	january	Cleveland	2	
8	january	Derbyshire	0	
11	january	Durham	1	
13	january	Essex	3	
15	january	GreaterManchester	2	

In [108]:
```
predicted <- data$prediction
actual <- data$offense_region

# Create confusion matrix
conf_matrix <- table(actual, data$prediction)
conf_matrix
```

```
actual               Avon and Somerset Bedfordshire Cambridgeshire Ches
hire
  Avon and Somerset                  4            0              0
0
  Bedfordshire                       3            0              0
0
  Cambridgeshire                     1            0              0
0
  Cheshire                           0            0              0
0
  Cleveland                          2            0              0
0
  Cumbria                            0            0              0
0
  Derbyshire                         2            0              0
0
  Devon and Cornwall                 1            0              0
0
  Durham                             0            0              0
```

In [329]:
```
colnames(data)
```

'Offence.category' 'Number.of.Homicide.Convictions'
'Percentage.of.Homicide.Convictions' 'Number.of.Homicide.Unsuccessful'
'Percentage.of.Homicide.Unsuccessful'
'Number.of.Offences.Against.The.Person.Convictions'
'Percentage.of.Offences.Against.The.Person.Convictions'
'Number.of.Offences.Against.The.Person.Unsuccessful'
'Percentage.of.Offences.Against.The.Person.Unsuccessful'
'Number.of.Sexual.Offences.Convictions' 'Percentage.of.Sexual.Offences.Convictions'
'Number.of.Sexual.Offences.Unsuccessful' 'Percentage.of.Sexual.Offences.Unsuccessful'
'Number.of.Burglary.Convictions' 'Percentage.of.Burglary.Convictions'
'Number.of.Burglary.Unsuccessful' 'Percentage.of.Burglary.Unsuccessful'
'Number.of.Robbery.Convictions' 'Percentage.of.Robbery.Convictions'
'Number.of.Robbery.Unsuccessful' 'Percentage.of.Robbery.Unsuccessful'
'Number.of.Theft.And.Handling.Convictions'
'Percentage.of.Theft.And.Handling.Convictions'
'Number.of.Theft.And.Handling.Unsuccessful'
'Percentage.of.Theft.And.Handling.Unsuccessful'
'Number.of.Fraud.And.Forgery.Convictions'
'Percentage.of.Fraud.And.Forgery.Convictions'
'Number.of.Fraud.And.Forgery.Unsuccessful'
'Percentage.of.Criminal.Damage.Convictions' 'Number.of.Criminal.Damage.Unsuccessful'
'Percentage.of.Criminal.Damage.Unsuccessful' 'Number.of.Drugs.Offences.Convictions'
'Percentage.of.Drugs.Offences.Convictions' 'Number.of.Drugs.Offences.Unsuccessful'
'Percentage.of.Drugs.Offences.Unsuccessful'
'Number.of.Public.Order.Offences.Convictions'
'Percentage.of.Public.Order.Offences.Convictions'
'Number.of.Public.Order.Offences.Unsuccessful'
'Percentage.of.Public.Order.Offences.Unsuccessful'
'Number.of.All.Other.Offences..excluding.Motoring..Convictions'
'Percentage.of.All.Other.Offences..excluding.Motoring..Convictions'
'Number.of.All.Other.Offences..excluding.Motoring..Unsuccessful'
'Percentage.of.All.Other.Offences..excluding.Motoring..Unsuccessful'
'Number.of.Motoring.Offences.Convictions' 'Percentage.of.Motoring.Offences.Convictions'
'Number.of.Motoring.Offences.Unsuccessful'
'Percentage.of.Motoring.Offences.Unsuccessful'
'Number.of.Admin.Finalised.Unsuccessful'
'Percentage.of.L.Motoring.Offences.Unsuccessful'

In [109]:

```
# Predictive model

new_data <- data.frame(Number.of.Drugs.Offences.Convictions = data_with_cor
                       Number.of.Drugs.Offences.Unsuccessful = data_with_c

prediction <- predict(model_tree, newdata = new_data, type = "class")
prediction <- as.data.frame(prediction)
prediction
```

A data.frame: 177 × 1

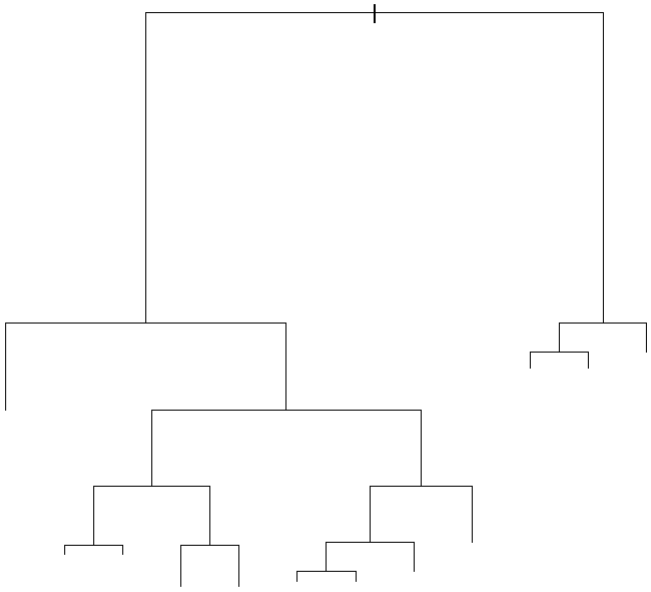
	prediction <fct>
3	North Yorkshire
6	Avon and Somerset
8	North Yorkshire
11	North Yorkshire
13	Avon and Somerset
15	Avon and Somerset
18	Avon and Somerset
19	North Yorkshire
21	Thames Valley
22	Avon and Somerset
23	North Yorkshire
24	Avon and Somerset
28	Avon and Somerset
29	North Yorkshire
33	Avon and Somerset
34	Avon and Somerset
35	North Yorkshire
38	Thames Valley
41	North Yorkshire
42	Avon and Somerset
43	North Yorkshire
45	North Yorkshire
48	North Yorkshire
52	North Yorkshire
56	North Yorkshire
58	North Yorkshire
60	Avon and Somerset
61	North Yorkshire
63	Nottinghamshire
64	North Yorkshire
⋮	⋮
292	Avon and Somerset
296	North Yorkshire
298	North Yorkshire

	prediction
	<fct>
299	Thames Valley
300	Nottinghamshire
303	Avon and Somerset
304	Avon and Somerset
307	North Yorkshire
309	North Yorkshire
312	North Yorkshire
315	Avon and Somerset
316	North Yorkshire
317	Avon and Somerset
318	Avon and Somerset
321	Avon and Somerset
322	Avon and Somerset
323	Avon and Somerset
324	North Yorkshire
325	North Yorkshire
327	Avon and Somerset
329	North Yorkshire
332	Avon and Somerset
333	Avon and Somerset
334	North Yorkshire
335	North Yorkshire
338	North Yorkshire
339	Thames Valley
341	North Yorkshire
342	North Yorkshire
343	Nottinghamshire

In [110]:

```
# classification model
# Classification with decision tree model:
library(rpart)
model_tree <- rpart(data_with_conviction_rate$Conviction_Rate_Drugs.Offence
printcp(model_tree)
plot(model_tree)
```

```
Regression tree:
rpart(formula = data_with_conviction_rate$Conviction_Rate_Drugs.Offences
~
    data_with_conviction_rate$Number.of.Drugs.Offences.Convictions +
        data_with_conviction_rate$Number.of.Drugs.Offences.Unsuccessful,
    data = data_with_conviction_rate)

Variables actually used in tree construction:
[1] data_with_conviction_rate$Number.of.Drugs.Offences.Convictions
[2] data_with_conviction_rate$Number.of.Drugs.Offences.Unsuccessful

Root node error: 2009.3/177 = 11.352

n= 177

         CP nsplit rel error  xerror     xstd
1  0.361993      0   1.00000 1.01770 0.108159
2  0.101672      1   0.63801 0.72002 0.101109
3  0.088134      2   0.53633 0.69601 0.093960
4  0.069310      3   0.44820 0.66231 0.079553
5  0.065718      4   0.37889 0.60053 0.075553
6  0.047697      5   0.31317 0.58137 0.076519
7  0.026670      6   0.26547 0.43683 0.060290
8  0.020285      7   0.23881 0.40734 0.051702
9  0.018763      8   0.21852 0.40782 0.051216
10 0.011852      9   0.19976 0.39633 0.050542
11 0.010635     10   0.18790 0.37466 0.050116
12 0.010000     11   0.17727 0.37240 0.049769
```



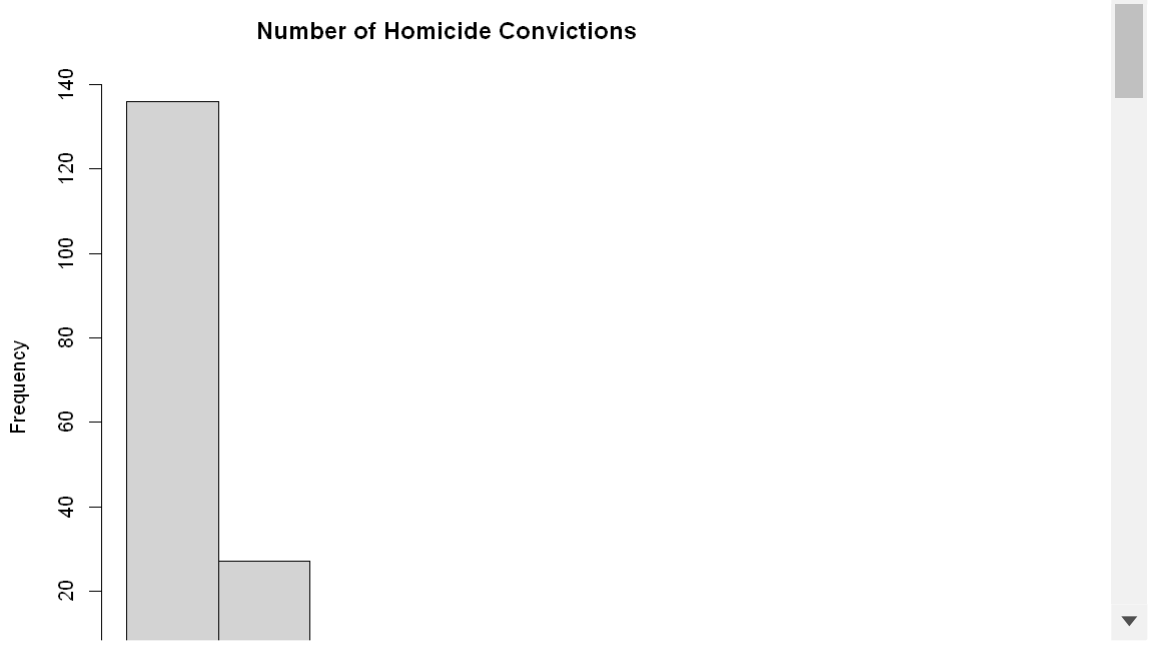
In [111]:

```
# Predictive model

new_data <- data.frame(Number.of.Offences.Against.The.Person.Convictions =
                       Number.of.Sexual.Offences.Convictions = data$Number

prediction <- predict(model_tree, newdata = new_data, type = "class")
prediction <- as.data.frame(prediction)
prediction
```

```
Error in predict.rpart(model_tree, newdata = new_data, type = "class"): I
nvalid prediction for "rpart" object
Traceback:

1. predict(model_tree, newdata = new_data, type = "class")
2. predict.rpart(model_tree, newdata = new_data, type = "class")
3. stop("Invalid prediction for \"rpart\" object")
```

PART 5: DATA VISUALIZATION

In [112]:
```
# Visualizing the data using histograms
hist(data$Number.of.Homicide.Convictions, main = "Number of Homicide Convi
hist(data$Number.of.Offences.Against.The.Person.Convictions, main = "Numbe
hist(data$Number.of.Sexual.Offences.Convictions , main = "Number of Sexual
hist(data$Number.of.Offences.Against.The.Person.Convictions, main = "Numbe
hist(data$Number.of.Motoring.Offences.Convictions, main = "Number.of.Motor
```



THANKS