# Assignment_1

Henry Lewis 20202851

30/09/2021

# Task 1: Manipulation

## Task 1.1: Load the dataset EurostatCrime2019.csv. Notice that the first column of the csv file contains the names of the countries that must be read as row names

```
data <- read.csv("EurostatCrime2019.csv") # reads the csv to a variable data
head(data,3)
```

```
##       ï.. Intentional.homicide Attempted.intentional.homicide Assault
## 1 Albania                 2.03                           3.25    5.52
## 2 Austria                 0.84                           1.93   43.29
## 3 Belgium                 1.27                           8.87  556.36
##   Kidnapping Sexual.violence  Rape Sexual.assault Robbery Burglary
## 1       0.14            5.38  2.69           2.69    3.42       NA
## 2       0.07           50.90 18.92          26.64   29.67   613.22
## 3         NA           77.45 33.33          44.12  140.14   565.92
##   Burglary.of.private.residential.premises   Theft
## 1                                   40.42  168.84
## 2                                   99.31 1302.92
## 3                                  410.12 1951.96
##   Theft.of.a.motorized.land.vehicle
## 1                             11.11
## 2                             44.22
## 3                            109.76
##   Unlawful.acts.involving.controlled.drugs.or.precursors
## 1                                                  70.26
## 2                                                 494.05
## 3                                                 547.74
```

## Task 1.2: What is the size (number of rows and columns) and the structure of this dataset?

```
cols <- ncol(data) # number of columns
rows <- nrow(data) # number of rows
cat("The number of columns are:", cols, "whilst the number of rows are", rows)
```

```
## The number of columns are: 14 whilst the number of rows are 41
```

```
str(data)
```

```
## 'data.frame':    41 obs. of  14 variables:
##  $ ï..                                                : chr  "Albania" "Austria" "Belgi
um" "Bosnia and Herzegovina" ...
##  $ Intentional.homicide                               : num  2.03 0.84 1.27 NA 1.14 0.8
1 1.48 0.76 0.91 NA ...
##  $ Attempted.intentional.homicide                     : num  3.25 1.93 8.87 NA 0.54 2.4
1.71 0.58 2.57 NA ...
##  $ Assault                                            : num  5.52 43.29 556.36 NA 39.54
...
##  $ Kidnapping                                         : num  0.14 0.07 NA NA 1.03 0.02
0.91 0.11 NA NA ...
##  $ Sexual.violence                                    : num  5.38 50.9 77.45 NA 8.64
...
##  $ Rape                                               : num  2.69 18.92 33.33 NA 1.87
...
##  $ Sexual.assault                                     : num  2.69 26.64 44.12 NA NA ...
##  $ Robbery                                            : num  3.42 29.67 140.14 NA 16.9
...
##  $ Burglary                                           : num  NA 613.2 565.9 NA 79.8 ...
##  $ Burglary.of.private.residential.premises           : num  40.4 99.3 410.1 NA NA ...
##  $ Theft                                              : num  169 1303 1952 NA 474 ...
##  $ Theft.of.a.motorized.land.vehicle                  : num  11.1 44.2 109.8 NA 18.9
...
##  $ Unlawful.acts.involving.controlled.drugs.or.precursors: num  70.3 494.1 547.7 NA 78.1
...
```

Task 1.3: Produce appropriate commands to do the following actions.

part (i): Remove the columns Rape and Sexual.assault.

```
# Use subset on dataframe and remove the specified columns of rape and sexual assault with se
lect parameter and -c for removal
data <- subset (data, select = -c(Rape,Sexual.assault))
colnames(data)
```

```
##  [1] "ï.."
##  [2] "Intentional.homicide"
##  [3] "Attempted.intentional.homicide"
##  [4] "Assault"
##  [5] "Kidnapping"
##  [6] "Sexual.violence"
##  [7] "Robbery"
##  [8] "Burglary"
##  [9] "Burglary.of.private.residential.premises"
## [10] "Theft"
## [11] "Theft.of.a.motorized.land.vehicle"
## [12] "Unlawful.acts.involving.controlled.drugs.or.precursors"
```

part (ii): Remove the columns involving theft and burglary.

```
# Use subset on dataframe and remove the specified columns elect parameter and -c for removal
data <- subset (data, select = -c(Theft,Theft.of.a.motorized.land.vehicle,Burglary,Burglary.o
f.private.residential.premises))

# Take the columns names to ensure removal has taken place
colnames(data)
```

```
## [1] "ï.."
## [2] "Intentional.homicide"
## [3] "Attempted.intentional.homicide"
## [4] "Assault"
## [5] "Kidnapping"
## [6] "Sexual.violence"
## [7] "Robbery"
## [8] "Unlawful.acts.involving.controlled.drugs.or.precursors"
```

## part (iii): Add a column containing the overall record of offences for each country (per hundred thousand inhabitants)?

```
# Create a column Overall_offences and populate it with the sum across each row, ignoring the
NA values
data$Overall_offences <- apply(data[,-1], 1, sum, na.rm=TRUE)

# Change the column names of the data frame to those specified
names(data) <- c('Country','Murder', 'Attempted_Murder', 'Assault', 'Kidnapping' ,'Sexual_Vio
lance', 'Robbery','Drugs', 'Overall_offences')

# Take the top 3 rows
head(data,3)
```

```
##   Country Murder Attempted_Murder Assault Kidnapping Sexual_Violance Robbery
## 1 Albania   2.03             3.25    5.52       0.14            5.38    3.42
## 2 Austria   0.84             1.93   43.29       0.07           50.90   29.67
## 3 Belgium   1.27             8.87  556.36         NA           77.45  140.14
##    Drugs Overall_offences
## 1  70.26            90.00
## 2 494.05           620.75
## 3 547.74          1331.83
```

## Task 1.4: Work with the dataset you just created, and list the countries that contain any missing data.

```
# Take the rows where countries have NA in rows
list_countries_na <- data[!complete.cases(data),]

# Take the country column (1st column)
list_countries_na[,1]
```

```
##  [1] "Belgium"                "Bosnia and Herzegovina" "Denmark"
##  [4] "England and Wales"      "Estonia"                "France"
##  [7] "Hungary"                "Iceland"                "Liechtenstein"
## [10] "Netherlands"            "North Macedonia"        "Northern Ireland (UK)"
## [13] "Norway"                 "Poland"                 "Portugal"
## [16] "Scotland"               "Slovakia"               "Sweden"
## [19] "Turkey"
```

Task 1.5: Remove the countries with missing data from the dataframe.

```
# Removes the countries with missing data
df <- data[complete.cases(data),]

# Gets top 3 rows
head(df,3)
```

```
##    Country Murder Attempted_Murder Assault Kidnapping Sexual_Violance Robbery
## 1  Albania   2.03             3.25    5.52       0.14            5.38    3.42
## 2  Austria   0.84             1.93   43.29       0.07           50.90   29.67
## 5 Bulgaria   1.14             0.54   39.54       1.03            8.64   16.90
##    Drugs Overall_offences
## 1  70.26            90.00
## 2 494.05           620.75
## 5  78.14           145.93
```

Task 1.6: How many observations and variables are in this new dataframe?.

```
# Gets the rows and columns
observations <- ncol(df)
observations
```

```
## [1] 9
```

```
variables<- nrow(df)
variables
```

```
## [1] 22
```

# Task 2: Manipulation

Task 2.1: According to these data what were the 3 most common crimes in Ireland in 2019?

```
# Gets the row with Ireland and the selected columns
new_row <- subset(df, Country == "Ireland", select=Murder:Drugs)
# sort the rows
sort_row <- apply(new_row , 1, sort, decreasing = TRUE)
# Take the top 3 rows
most_common_crimes <- head(sort_row,3)
most_common_crimes
```

```
##                    18
## Drugs           421.84
## Assault         102.18
## Sexual_Violance  67.86
```

## Task 2.2: What proportion of the overall crimes was due to Assault in Ireland in 2019?

```
# Gets the assault data for Ireland
assault <- subset(df, Country == "Ireland", select=Assault)

# Calculate the proportion of assault as a percentage of overall crimes
assault_proportion <- (assault/subset(df, Country == "Ireland", select=Overall_offences))*100
assault_proportion
```

```
##     Assault
## 18 16.05316
```

## Task 2.3: Which country had the highest record of kidnapping in 2019 (per hundred thousand inhabitants)?

```
# Generate a dataframe of country and kidnapping columns
k <- subset(df, select=c("Country", "Kidnapping"))

# Order based on kidnapping values
country <- k[order(-k$Kidnapping),]

# select the top country
country_kidnapping <- country$Country[1]
country_kidnapping
```

```
## [1] "Luxembourg"
```

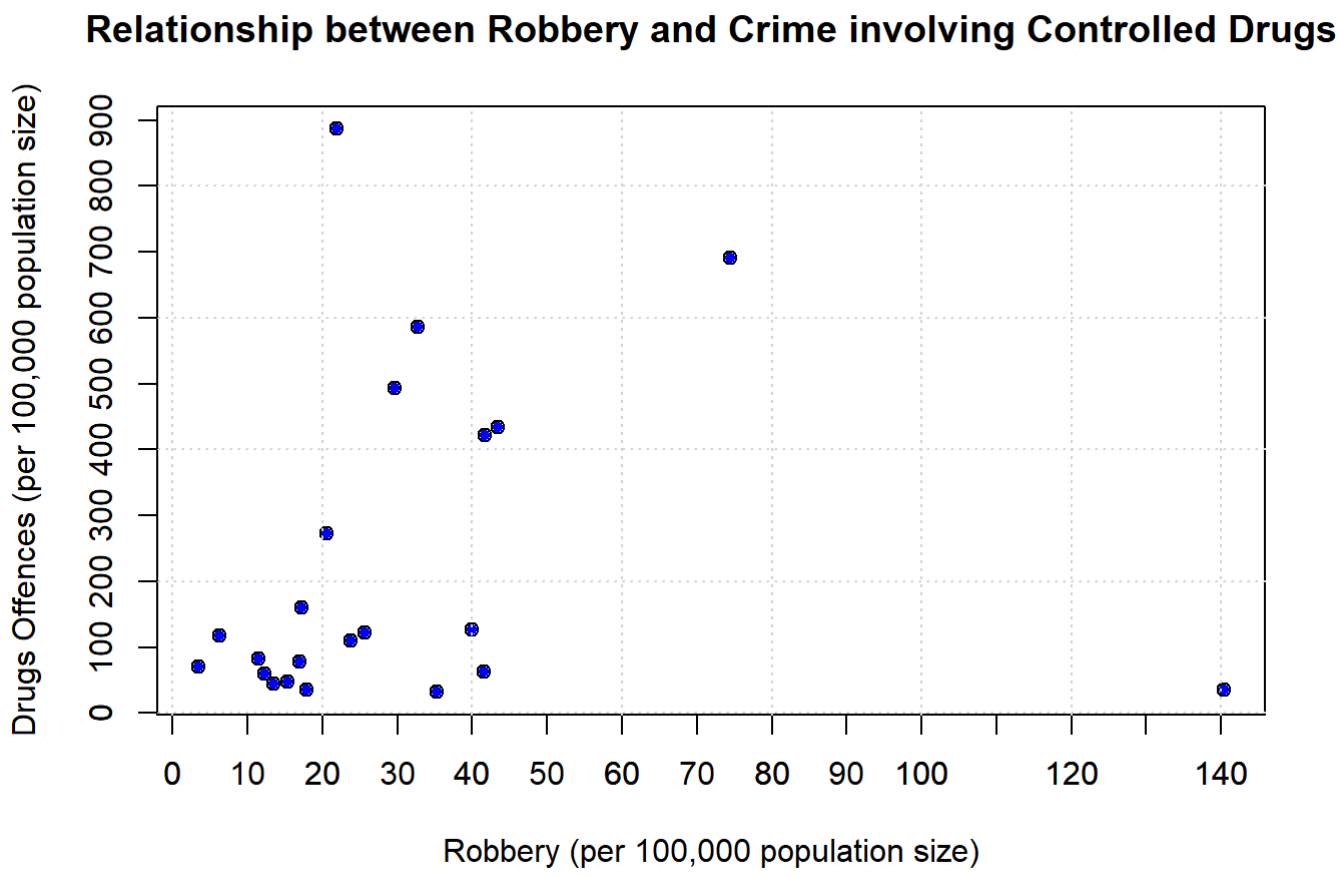
## Task 2.4: Which country had the lowest overall record of offences in 2019 (per hundred thousand inhabitants)?

```
# Generate a dataframe of country and overall offences columns
overall_offence_country <- subset(df, select=c("Country", "Overall_offences"))

# Order the dataframe on overall offences
overall_offence_country_sort <- overall_offence_country[order(overall_offence_country$Overall
_offences),]

# Take top country
overall_offences_low <- overall_offence_country_sort$Country[1]
overall_offences_low
```

```
## [1] "Romania"
```

## Task 2.5: Create a plot displaying the relationship between robbery and unlawful acts involving controlled drugs or precursors.

```
# Plotting information
x <-df$Robbery
y <- df$Drugs

par(mfcol = c(1,1))
# Setting the plot and axes information
plot(x,y,
xlab = "Robbery (per 100,000 population size)", ylab = "Drugs Offences (per 100,000 populatio
n size)")
title("Relationship between Robbery and Crime involving Controlled Drugs")
axis(1, at = pretty(x, n = 10))
axis(2, at = pretty(y, n = 10))
points(x, y, col = "blue", pch = 20)
grid()
```

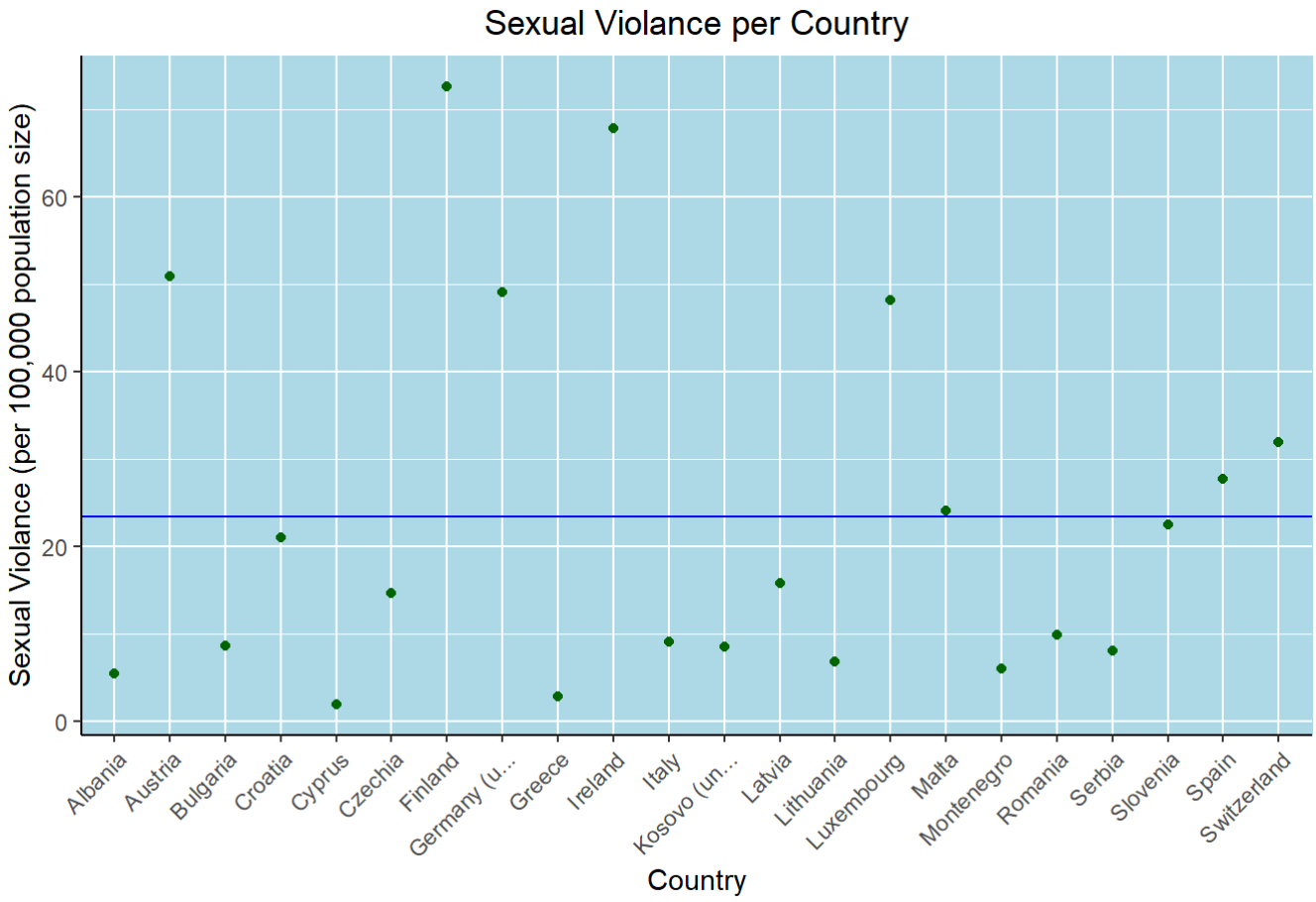



# Task 3: Creativity

## Do something interesting with these data

The plot below details the levels of sexual violence on a per country basis. From the graph is it clear that both Finland and Ireland have the highest levels of sexual violence at greater than 60 offences (per 100,000), whilst Austria, Germany and Luxembourg have similar rates of approximately 50 offences (per 100,000). Countries such as Spain and Switzerland are slightly higher than the average, with Malta just above average.

```
library(ggplot2)
library(stringr)
theme_set(theme_classic())

ggplot(df, aes(x = Country, y = Sexual_Violance))+
  geom_point(color = "darkgreen")+
  # angles the labels
  theme(axis.text.x = element_text(angle = 45, hjust=1))+
  # adds intercept for average per country
  geom_hline(aes(yintercept = mean(Sexual_Violance)), color="blue") +
  # truncates labels
  scale_x_discrete(label = function(x) stringr::str_trunc(x, 13)) + ggtitle("Sexual Violance
 per Country") +
  # Adds a theme and adds a centered title
  theme(plot.title = element_text(hjust = 0.5))+
  # Adds detailed caption information alongside a theme
  labs(caption = "Data source: Eurostat",
              x = "Country", y = "Sexual Violance (per 100,000 population size)") + theme(
                                            panel.background = element_rect(fill = "lightblu
e",
                                                                            colour = "lightblu
e",
                                                                            size = 0.5, linetyp
e = "solid"),
                                            panel.grid.major = element_line(size = 0.5, linet
ype = 'solid',
                                                                            colour = "white"),
                                            panel.grid.minor = element_line(size = 0.25, line
type = 'solid',
                                                                            colour = "white")
                                            )
```




Data source: Eurostat