

Assignment1

Junhao Ou & 21200066

2022-09-26

Task 1: Manipulation

1. Load the dataset EurostatCrime2019.csv

In this section, we will load the dataset

```
df = read.csv('EurostatCrime2019.csv', row.names = 1)
```

2. Structure Check

In this section, we will check the size (number of rows and columns) and the structure of this data.

```
str(df)
```

```
## 'data.frame':   41 obs. of  13 variables:
## $ Intentional.homicide           : num  2.03 0.84 1.27 NA 1.14 0.81 1.48 0.7
## $ Attempted.intentional.homicide : num  3.25 1.93 8.87 NA 0.54 2.4 1.71 0.58
## $ 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
## $ 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 ...
```

As we can see, the size of the table is 41 rows by 13 columns.

3. Produce appropriate commands to do the following actions.

(i) Remove the columns of Rape and Sexual.assault.

```
df = df[, -which(names(df) %in% c('Rape', 'Sexual.assault'))]
str(df)
```

```
## 'data.frame': 41 obs. of 11 variables:
## $ Intentional.homicide : num 2.03 0.84 1.27 NA 1.14 0.81 1.48 0.7
## $ Attempted.intentional.homicide : num 3.25 1.93 8.87 NA 0.54 2.4 1.71 0.58
## $ 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
## $ Sexual.violence : num 5.38 50.9 77.45 NA 8.64 ...
## $ 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 ...
```

(ii) Remove the columns involving theft and burglary

```
library(dplyr)
```

```
##
## 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
```

```
df = select(df, -contains("Theft"))
df = select(df, -contains("Burglary"))
str(df)
```

```
## 'data.frame': 41 obs. of 7 variables:
## $ Intentional.homicide : num 2.03 0.84 1.27 NA 1.14 0.81 1.48 0.7
## $ Attempted.intentional.homicide : num 3.25 1.93 8.87 NA 0.54 2.4 1.71 0.58
## $ 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
## $ Sexual.violence : num 5.38 50.9 77.45 NA 8.64 ...
## $ Robbery : num 3.42 29.67 140.14 NA 16.9 ...
## $ Unlawful.acts.involving.controlled.drugs.or.precursors: num 70.3 494.1 547.7 NA 78.1 ...
```

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

```
df['Overall Record'] = rowSums(df[,c(1:7)])
```

4. List the countries that contain any missing data.

```
row.names(df[rowSums(is.na(df)) > 0,])
```

```
## [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"
```

5. Remove the countries with missing data from the dataframe.

```
df = subset(df, rowSums(is.na(df)) <= 0)
```

6. How many observations and variables are in this new dataframe?

```
str(df)
```

```
## 'data.frame': 22 obs. of 8 variables:
## $ Intentional.homicide : num 2.03 0.84 1.14 0.81 1.48 0.76 1.59 0
## $ Attempted.intentional.homicide : num 3.25 1.93 0.54 2.4 1.71 0.58 5.96 2
## $ Assault : num 5.52 43.29 39.54 18.06 20.09 ...
## $ Kidnapping : num 0.14 0.07 1.03 0.02 0.91 0.11 0.02 5
## $ Sexual.violence : num 5.38 50.9 8.64 21.05 1.94 ...
## $ Robbery : num 3.42 29.67 16.9 20.56 6.28 ...
## $ Unlawful.acts.involving.controlled.drugs.or.precursors: num 70.3 494.1 78.1 272.2 117.8 ...
## $ Overall Record : num 90 621 146 335 150 ...
```

Task 2: Analysis

1. According to these data what were the 3 most common crimes in Ireland in 2019?

Then, we will implement a function that can return the 3 most common crimes with input country.

```
maxthree = function(data, row) {
  x <- data[row,1:7]
  return(x[order(x, decreasing = T)][1:3])
}
```

Check the 3 most common crimes in Ireland in 2019

```
maxthree(df, "Ireland")
```

```
## Warning in xtfm.data.frame(x): cannot xtfm data frames
```

```
##      Unlawful.acts.involving.controlled.drugs.or.precursors Assault
## Ireland                                     421.84  102.18
##      Sexual.violence
## Ireland                                     67.86
```

2. What proportion of the overall crimes was due to Assault in Ireland in 2019?

We will implement a function that can return the proportion of any type of crime of any country.

```
propCal = function(data, country, crime) {
  prop = data[country, crime] / data[country, 'Overall Record']
  return(prop)
}
```

Check the proportion of the overall crimes was due to Assault in Ireland.

```
propCal(df, "Ireland", 'Assault')
```

```
## [1] 0.1605316
```

3. Which country had the highest record of kidnapping in 2019.

```
rownames(df)[order(df[, 'Kidnapping'], decreasing = T)][1]
```

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

4. Which country had the lowest overall record of offences in 2019 (per hundred thousand inhabitants)?

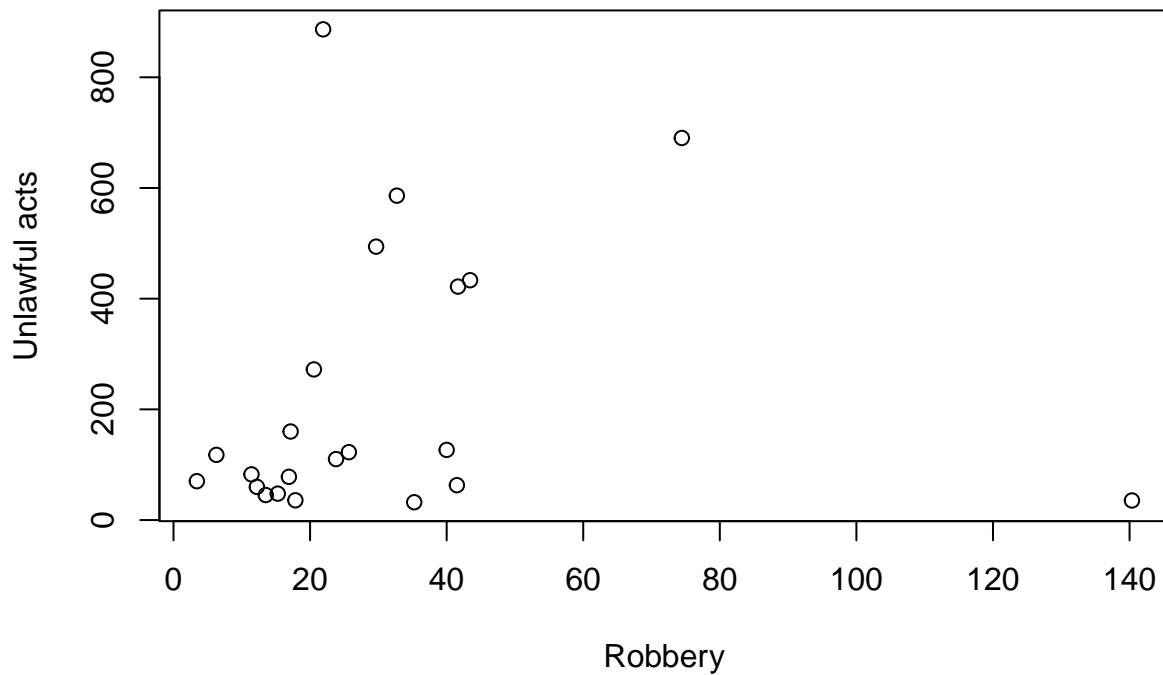
```
rownames(df)[order(df[, 'Overall Record'], decreasing = F)][1]
```

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

5. Create a plot displaying the relationship between robbery and unlawful acts involving controlled drugs or precursors. Make the plot look “nice” i.e. change axis labels etc.

```
plot(df$Robbery, df$Unlawful.acts.involving.controlled.drugs.or.precursors, xlab='Robbery', ylab='Unlawful
```

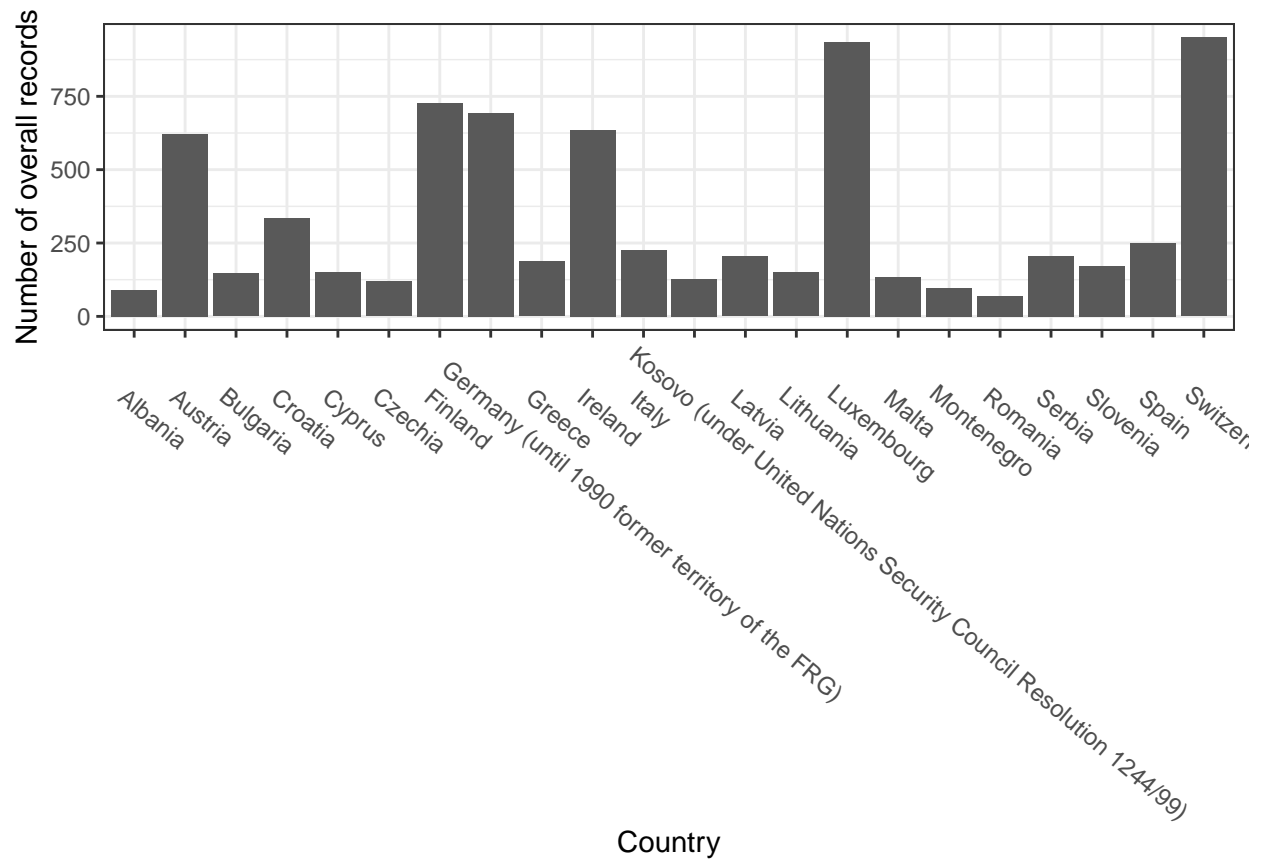
The relationship between robbery and unlawful acts



Task 3: Creativity

1. We will plot the histogram of Overall Record for each country

```
library(reshape2)
library(ggplot2)
df1 <- melt(df$'Overall Record', id.vars="Row.names")
ggplot(df1, aes(x = rownames(df), y=value)) + geom_bar(stat="identity", position = "dodge") + theme_bw() + theme
```



From the plot above, we can conclude that the top 5 countries with the greatest number of crime records are Luxembourg, Switzerland, Finland, Germany and Ireland.