Eurostat CSO (Central Statistics Office, Ireland)

```r
crime <- import("crim_off_cat_1_Data.csv",setclass = "tibble")

# There are two ways to pull the imigration data (CSO):
# 1. Use utility on the web link to downlad csv file of selected columns
# 2. Download .px file from CSO link and using PC-Axis software convert it to csv and then import it into R

#1.
imig <- import("20191224185114590075PEA1504147693656.csv",setclass = "tibble", skip = 2)
imig <- t(imig)                      #transpose
colnames(imig) <- imig[1,]           #Giving colnames from the 1st row
imig<-as_tibble(imig)                #Converting matrix imig2 to a tibble
imig<-imig[-1,]                      #Removing 1st row as info is now in colnames
colnames(imig)[1] <- "TIME"          #Setting colname of 1st col to TIME
imig <- lapply(imig,as.numeric)      #Converting each column to numeric
imig <- as_tibble(imig)              #converting list from lapply to tibble
imig$TIME <- as.integer(imig$TIME)   #Converting Years to integer

#2.
imig <- import("PEA15.csv",setclass = "tibble",skip = 2)
colnames(imig)[1] <- "TIME"                              #Setting colname of 1st col to TIME
imig[,c(2,3,5,6)] <- lapply(imig[,c(2,3,5,6)],as.numeric)  #Converting required columns to numeric
```

```r
#Structure ->
str(crime)
```

```
## tibble [5,330 x 6] (S3: tbl_df/tbl/data.frame)
##  $ TIME            : int [1:5330] 2008 2008 2008 2008 2008 2008 2008 2008 2008 2008 ...
##  $ GEO             : chr [1:5330] "Belgium" "Belgium" "Belgium" "Belgium" ...
##  $ ICCS            : chr [1:5330] "Intentional homicide" "Attempted intentional homicide" "Assault" "Ki
dnapping" ...
##  $ UNIT            : chr [1:5330] "Per hundred thousand inhabitants" "Per hundred thousand inhabitants"
"Per hundred thousand inhabitants" "Per hundred thousand inhabitants" ...
##  $ Value           : chr [1:5330] "1.91" "6.39" "715.43" "9.50" ...
##  $ Flag and Footnotes: logi [1:5330] NA NA NA NA NA NA ...
```

```r
str(imig)
```

```
## tibble [69 x 9] (S3: tbl_df/tbl/data.frame)
##  $ TIME            : int [1:69] 1951 1952 1953 1954 1955 1956 1957 1958 1959 1960 ...
##  $ Annual births   : num [1:69] NA NA NA NA NA NA NA NA NA NA ...
##  $ Annual deaths   : num [1:69] NA NA NA NA NA NA NA NA NA NA ...
##  $ Natural increase : num [1:69] 26.6 27.3 29.1 28.2 24.7 25.6 27.8 25.4 24.9 27.5 ...
##  $ Immigrants      : num [1:69] NA NA NA NA NA NA NA NA NA NA ...
##  $ Emigrants       : num [1:69] NA NA NA NA NA NA NA NA NA NA ...
##  $ Net migration   : num [1:69] -35 -35 -33 -36 -45 -48 -41 -58 -32 -41 ...
##  $ Population change: num [1:69] -8.4 -7.7 -3.9 -7.8 -20.3 -22.4 -13.2 -32.6 -7.1 -13.5 ...
##  $ Population       : num [1:69] 2961 2953 2949 2941 2921 ...
```

```r
# Value in Crime Dataset read as char because as per the data description, those values
# which were not available were filled with ":"
# using gsub to first convert numbers as char (seperated with commas) to numbers as numeric

crime$Value<- as.numeric(gsub(",","",crime$Value))
```

```
## Warning: NAs introduced by coercion
```

```
str(crime)
```

```
## tibble [5,330 x 6] (S3: tbl_df/tbl/data.frame)
##  $ TIME              : int [1:5330] 2008 2008 2008 2008 2008 2008 2008 2008 2008 2008 ...
##  $ GEO               : chr [1:5330] "Belgium" "Belgium" "Belgium" "Belgium" ...
##  $ ICCS              : chr [1:5330] "Intentional homicide" "Attempted intentional homicide" "Assault" "Ki
dnapping" ...
##  $ UNIT              : chr [1:5330] "Per hundred thousand inhabitants" "Per hundred thousand inhabitants"
"Per hundred thousand inhabitants" "Per hundred thousand inhabitants" ...
##  $ Value             : num [1:5330] 1.91 6.39 715.43 9.5 101.05 ...
##  $ Flag and Footnotes: logi [1:5330] NA NA NA NA NA NA ...
```

```
crime.wider <- pivot_wider(crime,names_from = "ICCS", values_from = "Value")

crime.wider$All_Theft <-  rowSums(crime.wider[,c('Burglary',
                                                 'Burglary of private residential premises',
                                                 'Theft',
                                                 'Theft of a motorized land vehicle')],
                          na.rm = TRUE)
# Keeping only required columns
crime.sub <- subset.data.frame(crime.wider, select = -c(3:4,13:16))

str(crime.sub)
```

```
## tibble [410 x 12] (S3: tbl_df/tbl/data.frame)
##  $ TIME                                            : int [1:410] 2008 2008 2008 2008 2008 2008 200
8 2008 2008 2008 ...
##  $ GEO                                             : chr [1:410] "Belgium" "Bulgaria" "Czechia" "D
enmark" ...
##  $ Intentional homicide                            : num [1:410] 1.91 2.29 1.09 0.96 0.8 6.28 1.14
1.29 0.89 1.6 ...
##  $ Attempted intentional homicide                  : num [1:410] 6.39 0.8 0.86 3.59 1.96 1.49 0.24
1.48 1.94 1.66 ...
##  $ Assault                                         : num [1:410] 715.4 42.8 52.2 30.4 630.6 ...
##  $ Kidnapping                                      : num [1:410] 9.5 1.69 0.15 NA 2.26 0.15 1.3 0.
27 0.52 3.24 ...
##  $ Sexual violence                                 : num [1:410] 101.05 9.96 16.24 33.09 69.07 ...
##  $ Rape                                            : num [1:410] 30.29 3.48 5.11 17.81 8.87 ...
##  $ Sexual assault                                  : num [1:410] 70.76 6.48 11.13 15.29 60.2 ...
##  $ Robbery                                         : num [1:410] 214.5 38.1 44.9 62.1 60.7 ...
##  $ Unlawful acts involving controlled drugs or precursors: num [1:410] 139.2 38 27.2 370 291.9 ...
##  $ All_Theft                                       : num [1:410] 3998 983 2300 8224 3212 ...
```

```
#creating another dataset which will also contain a column having Total offences for each country in each ye
ar
crime.total <- crime.sub
crime.total$Totals <- rowSums(crime.sub[,3:length(crime.sub)],na.rm = TRUE)

#Renaming some names which were quite large
crime.total[grep("Germany",crime.total$GEO),2] <- "Germany"
crime.total[grep("Kosovo",crime.total$GEO),2] <- "Kosovo"
```

```
knitr::kable(crime.sub[order(rowSums(crime.sub[,c(3:ncol(crime.sub))]),na.rm = TRUE),decreasing = TRUE)[1],])
```

```
a <- aggregate(crime.total$Totals,list(crime.total$TIME),max,na.rm= TRUE)
yearwisemax <- merge(crime.total,a,by.x = 'Totals', by.y = 'x')
knitr::kable(yearwisemax[order(yearwisemax$TIME),c(1:3)])
```

```
#Converting original dataframe into dataframe with 3 columns
#viz, Country, Type of Offence and Value

crime.longsub <- pivot_longer(crime.sub, cols = 3:12,names_to = "ICCS", values_to = "Values")

#Removing NA rows, as that will cause discrepancy in plots
#Note : Not deleting full country rows but only those combinations of Country-Type of Offence for which no v
alues are present
y <- subset.data.frame(crime.longsub,complete.cases(crime.longsub))
y[grep("Germany",y$GEO),2] <- "Germany"
y[grep("Kosovo",y$GEO),2] <- "Kosovo"

#Plotting the graph
p1 <- ggplot(y,aes(fill = y$ICCS, y = y$Values, x = reorder(y$GEO, -y$Values))) +
  geom_bar(position = "stack", stat = "identity") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1,vjust = 1),
        legend.position = "bottom" ) +
  labs(fill = "Types of Offences", y = "Offences value per thousand inhabitants",
       x = "Countries", title = "Crime Data for European Countries {closest_state}") +
  transition_states(y$TIME, transition_length = 10, state_length = 1)

animate(p1,  width = 900, height = 750, end_pause = 50, renderer = gifski_renderer("gganimq.gif"))
```
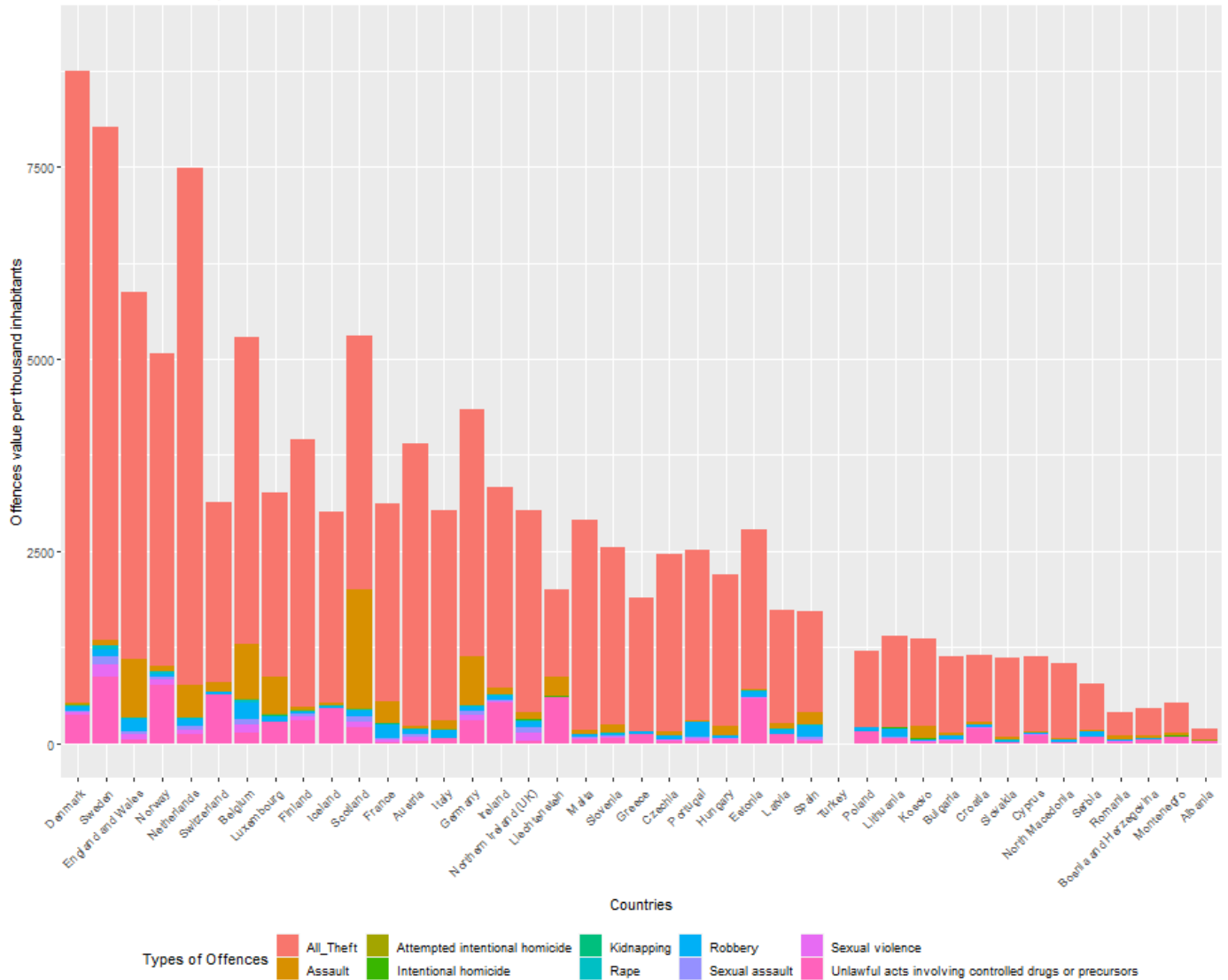
Crime Data for European Countries 2008
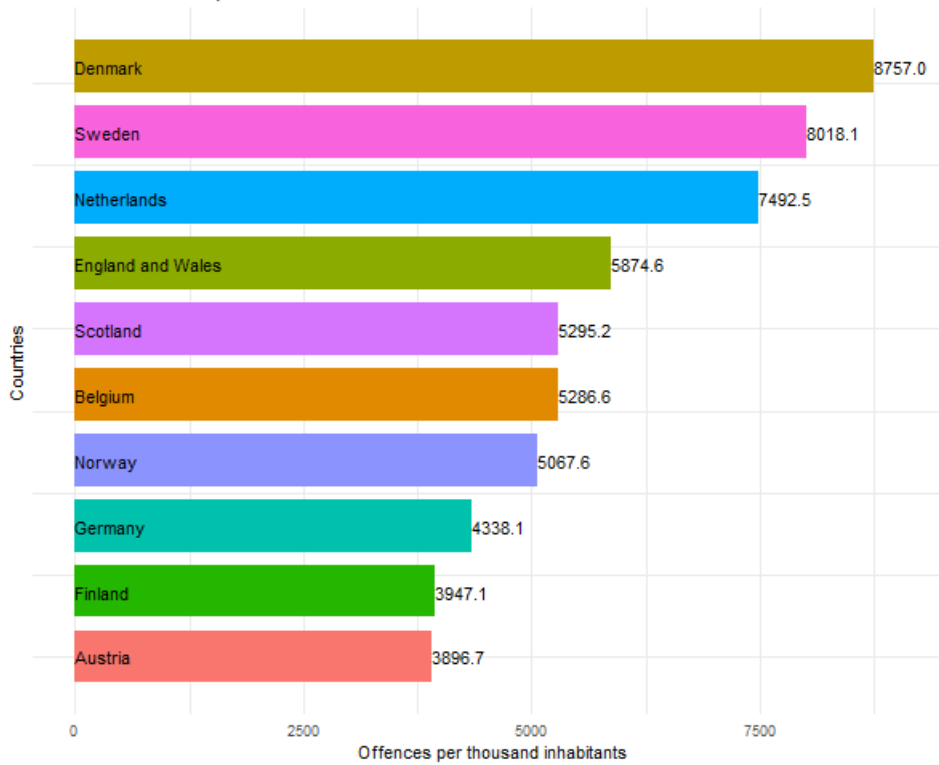
#Still need some work to remove those blank chunks in plot (would happily accept any suggestions)

```
p2 <- crime.total %>%
    group_by(TIME) %>%
    mutate(ordering = rank(-Totals)) %>%
  group_by(GEO) %>%
  filter(ordering <= 10) %>%
  ggplot(aes(-ordering, Totals, fill = GEO)) +
  geom_col(width = 0.8, position = "identity") +
  coord_flip() +
  geom_text(aes(-ordering, y=0, label = GEO, hjust=0)) +        #country label
  geom_text(aes(-ordering, y=Totals, label = scales::number(Totals,accuracy = 0.1,big.mark = "")), hjust=0))
+ # value label
  theme_minimal() +
  theme(legend.position = "none",axis.text.y=element_blank(), axis.ticks.y = element_blank()) +
  labs(y = "Offences per thousand inhabitants",
       x = "Countries", title = "Crime Data for European Countries {closest_state}") +
  transition_states(TIME, transition_length = 10, state_length = 14, wrap = FALSE) +
  ease_aes("sine-in-out")

animate(p2, fps = 20, width = 600, height = 500,nframes = 300, end_pause = 30, start_pause = 30, renderer =
gifski_renderer("gganim_top10.gif"))
```

Crime Data for European Countries 2008

| Country	Offences per thousand inhabitants
Denmark	8757.0
Sweden	8018.1
Netherlands	7492.5
England and Wales	5874.6
Scotland	5295.2
Belgium	5286.6
Norway	5067.6
Germany	4338.1
Finland	3947.1
Austria	3896.7

```r
plotdata.total <- crime.total %>%
    group_by(TIME) %>%
    mutate(ordering = rank(-Totals)) %>%
    ungroup()

plotorddatalong <- pivot_longer(plotdata.total,cols = 3:12, names_to = "ICCS", values_to = "Values")

p3 <- plotorddatalong %>%
  group_by(TIME) %>%
  group_by(GEO) %>%
  filter(ordering <= 15) %>%
  ggplot(aes(x = -ordering, y = Values, fill = ICCS)) +
  geom_bar(position = "stack", stat = "identity") +
  coord_flip() +
  geom_text(aes(-ordering,y=Totals,label = GEO, hjust=0)) +
  theme(legend.position = "bottom" , axis.text.y=element_blank(), axis.ticks.y = element_blank())+
  labs(fill = "Types of Offences", y = "Offences per thousand inhabitants",
       x = "Countries", title = "Crime Data for European Countries {closest_state}") +
  transition_states(TIME, transition_length = 1, state_length = 5) +
  ease_aes("sine-in-out")

animate(p3,  width = 756, height = 630, end_pause = 50, renderer = gifski_renderer("gganimtest.gif"))
```

Crime Data for European Countries 2008
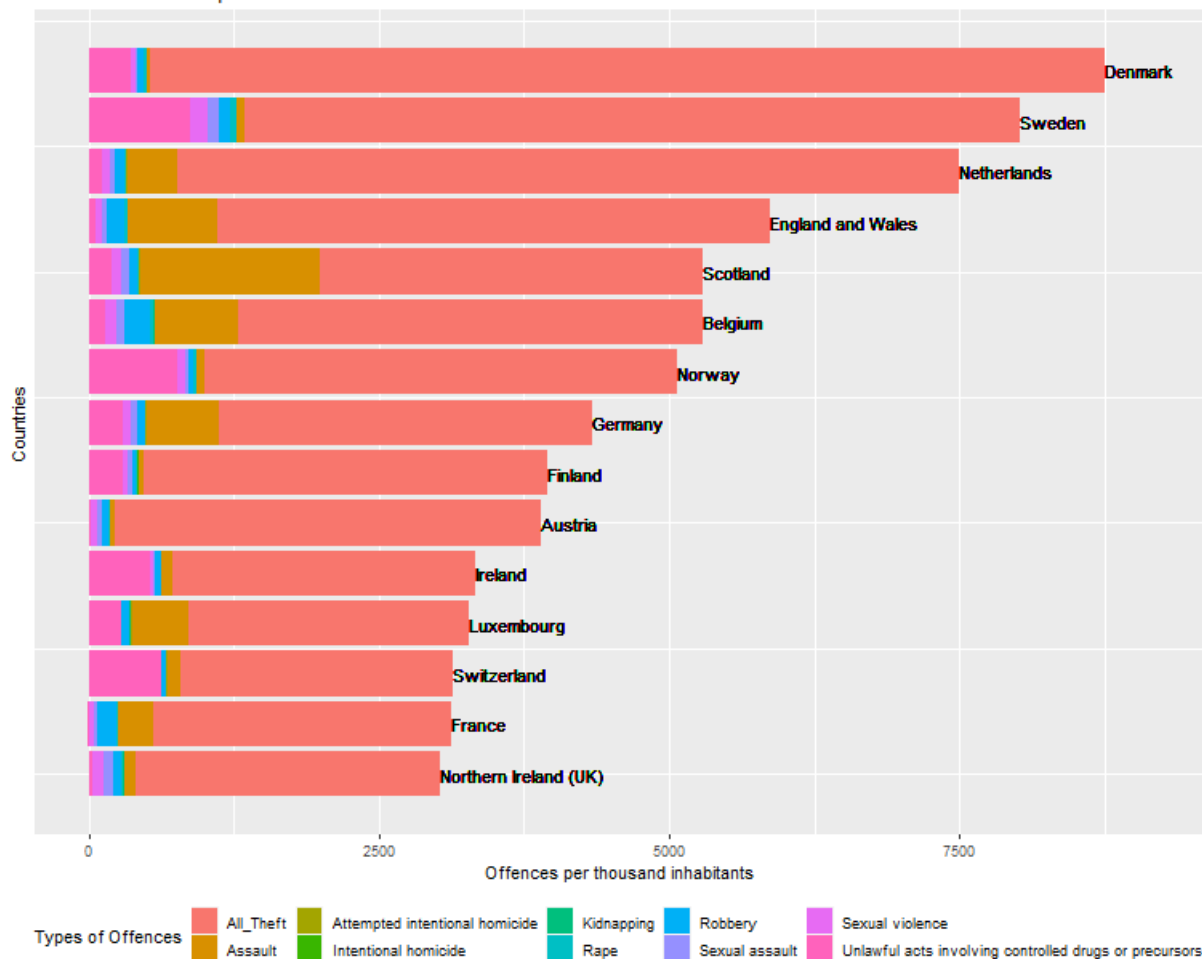
```
crime.ireland <- crime.sub[crime.sub$GEO == "Ireland",]
crime.ireland
```

```
## # A tibble: 10 x 12
##      TIME GEO   `Intentional ho~ `Attempted inte~ Assault Kidnapping
##     <int> <chr>            <dbl>            <dbl>   <dbl>      <dbl>
## 1   2008 Irel~             1.14             0.24    86.2       1.3
## 2   2009 Irel~             1.17             0.15    82.2       1.85
## 3   2010 Irel~             1.21             0.15    80.9       1.8
## 4   2011 Irel~             0.92             0.17    77.5       1.68
## 5   2012 Irel~             1.13             0.19    70.1       1.41
## 6   2013 Irel~             1.11             0.28    66.9       1.74
## 7   2014 Irel~             1.15             0.33    68.2       1.3
## 8   2015 Irel~             0.66             0.28    75.0       1.97
## 9   2016 Irel~             0.78             0.51    76.0       1.57
## 10  2017 Irel~             0.86             0.27    84.6       1.59
## # ... with 6 more variables: `Sexual violence` <dbl>, Rape <dbl>, `Sexual
## #   assault` <dbl>, Robbery <dbl>, `Unlawful acts involving controlled drugs or
## #   precursors` <dbl>, All_Theft <dbl>
```

```
crime.ireland2 <- pivot_longer(crime.ireland, cols = 3:12,names_to = "ICCS", values_to = "Values")
str(crime.ireland2)
```

```
## tibble [100 x 4] (S3: tbl_df/tbl/data.frame)
##  $ TIME  : int [1:100] 2008 2008 2008 2008 2008 2008 2008 2008 2008 2008 ...
##  $ GEO   : chr [1:100] "Ireland" "Ireland" "Ireland" "Ireland" ...
##  $ ICCS  : chr [1:100] "Intentional homicide" "Attempted intentional homicide" "Assault" "Kidnapping" ...
##  $ Values: num [1:100] 1.14 0.24 86.2 1.3 26.91 ...
```

```
ggplot(crime.ireland2,aes(TIME,Values)) +
  geom_line() +
  geom_point() +
  facet_wrap(~ICCS,scales = "free") +
  scale_x_continuous(breaks = seq(2006,2018,2)) +
  labs(x= "Time", y = "Number of Incidents Per Hundred Thousand Inhabitants")
```
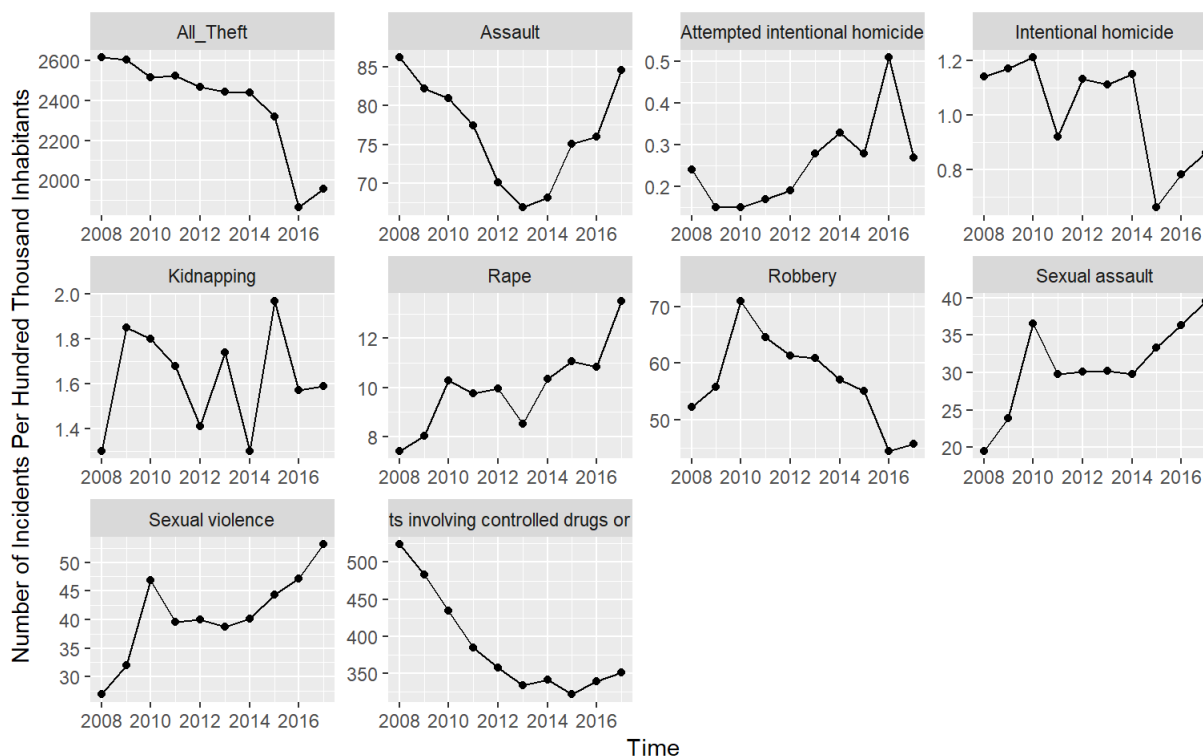


```
imig10yr <- imig[imig$TIME >= 2008 & imig$TIME <= 2017,]

crime.ireland$Immigrants <- imig10yr$Immigrants

cor(crime.ireland[,c(3:12)], crime.ireland$Immigrants)
```

```
##                                                              [,1]
## Intentional homicide                                   -0.2243421
## Attempted intentional homicide                          0.3692720
## Assault                                                 0.4773286
## Kidnapping                                             -0.3676157
## Sexual violence                                        -0.3530245
## Rape                                                   -0.1433708
## Sexual assault                                         -0.4040114
## Robbery                                                -0.7805264
## Unlawful acts involving controlled drugs or precursors  0.3526378
## All_Theft                                              -0.2079925
```

```
#corrplot::corrplot.mixed(cor(final[,3:12]))

#pairs(final[,2:12])
```