An analysis and preparation of Terrorism data set

BSc in computing and IT 4th Year: Data Exploration and Preparation

Luciano Gimenez
2023370

Contents

CCT College Dublin

Module Title:	Data exploration & preparation
Assessment Title:	CA 1 project
Lecturer Name:	Dr Muhammad Iqbal
Student Full Name:	Luciano Gimenez
Student Number:	2023370
Assessment Due Date:	3rd December 2023
Date of Submission:	3rd December 2023

Declaration

By submitting this assessment, I confirm that I have read the CCT policy on Academic Misconduct and understand the implications of submitting work that is not my own or does not appropriately reference material taken from a third party or other source. I declare it to be my own work and that all material from third parties has been appropriately referenced. I further confirm that this work has not previously been submitted for assessment by myself or someone else in CCT College Dublin or any other higher education institution.

Introduction

When talking about global crime we could not exclude terrorism as one of the biggest problems societies face nowadays. We chose an open-source database that compiles information on terrorist attacks from 1970 through 2020 (START, 2022).

The main scope of our analysis would be around three main areas, geographical spread of attacks, the temporal evolution, and the consequences of these attacks. When talking of the spread of the attacks we would analyse what are the regions that bear most of the effect. While referring to temporal evolution we would explore frequency and how it evolved over time by region. When discussing the consequences, we would centre on the casualties and material losses, as well we would include main targets and types of attack.

Due to the number of variables on the set, 135 features, we would choose a subset taking out the columns that are meant to keep explanatory text, as summaries or notes. After that we started with the cleaning, handling of missing data and outliers, we will not perform feature ranking because we do not have a target variable. We found that the data set contains some columns that are not part of our analysis like nkillus, Number of US Fatalities, so we decided to take them out.

We performed exploratory data analysis showing some graphs that represent the information we have the focus of our analysis on and calculating correlation coefficients, and Pearson's using chi squared.

Data Preparation

Preparing the dataset for understanding it and the decisions we made to clean or substitute de values of certain variables was difficult due to the size of the data set. At the start of the process, it contained 135 columns, and we reduced it to 24 after iterating in the process.

When preparing the data set, we started by reducing the number of columns. We started by dropping some variables that are present in the data base that are meant to keep explanatory data, due to the size of the data set these variables will not be part of our analysis. As well we would create date column and drop the individual columns for iyear, imonth, iday. For the scope of this project, where our preparation is not meant to use it to train an AI model in this stage, we would keep the columns that keep text instead of a mapping code. For the scope of our analysis, we would sum the nwound with nwoundte columns and repeat the same process with nkill and nkillter, to count total cost in human lives disregarding of their role in the attack.

The data set contains several columns where the source did not report the material damage nor casualties, we decided to replace this missing data for 0 to be able to analyse the casualties, knowing that there might be cases where the source did not inform them.

After counting the number of occurrences of zero value in certain columns, using boxplots and checking the maximum value of each numeric column we decided to keep the outliers.

Exploratory Data Analysis

We understand that analysing a data set is an iterative process, part of the CRISP-DM. We iterated thought Data understanding and Data preparation phases. On these cycles we employed a set of graphical representations, central tendency measures. We would describe each column and give the measures of centre for all of them, as well we calculate the correlation between the three numerical variables and used the chi-squared test to show the correlation between some of the categorical variables. Next, we would show our findings.

After we prepared the data set, we kept several character variables, that are categorical except for date. The int variables represent quantitative data, except decade and region. Region was recoded by us when experimenting with the data. On the variables we can observe some logical variables, that are binary, success, suicide, and crit1, political, economic, religious, or social, crit2, intention to coerce or publicize to larger audiences, and crit3 outside international humanitarian law, being these criteria met by the attacks.

```
data.frame':   209706 ob
$ eventid         : num
$ date            : chr
$ country_txt     : chr
$ region_txt      : chr
$ provstate       : chr
$ city            : chr
$ crit1           : logi
$ crit2           : logi
$ crit3           : logi
$ success         : logi
$ suicide         : logi
$ attacktype1_txt : chr
$ targtype1_txt   : chr
$ targsubtype1_txt: chr
..
$ target1         : chr
$ natlty1_txt     : chr
$ gname           : chr
$ weaptype1_txt   : chr
$ weapsubtype1_txt: chr
$ property        : logi
$ propextent_txt  : chr
$ propvalue       : num
$ total_wound     : int
$ total_kill      : int
$ decade          : num
$ casualties      : int
$ region          : int
```

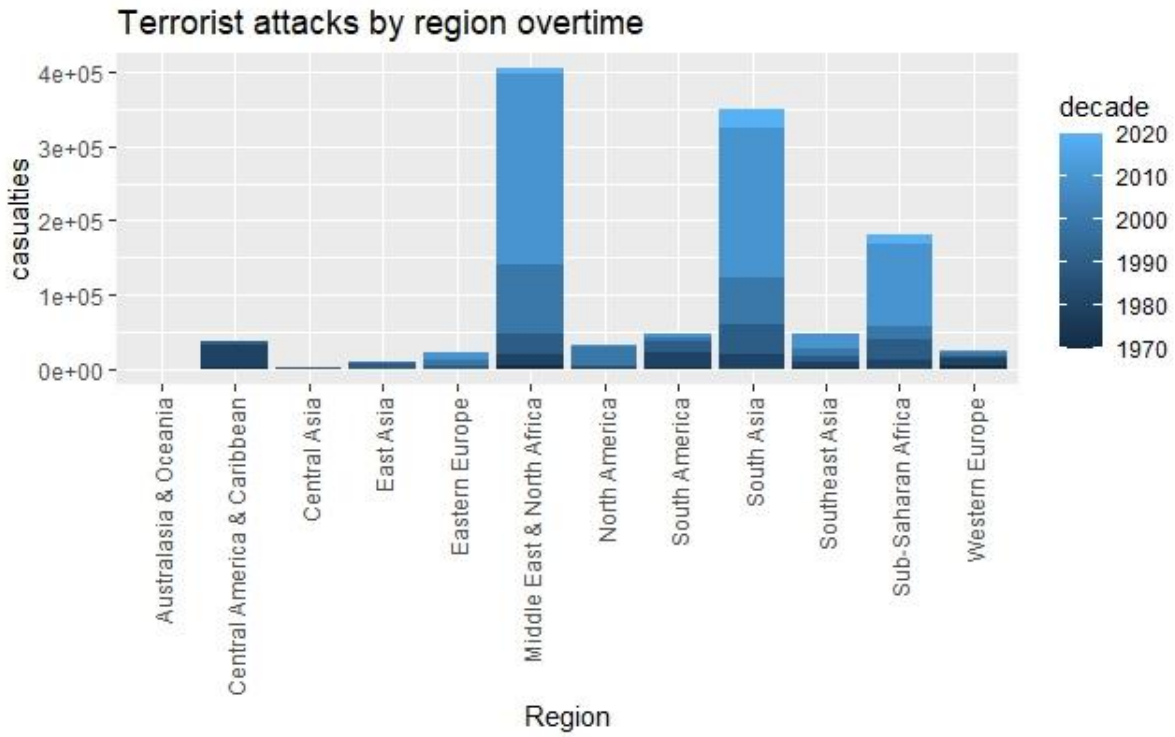
Here we can see a short summary of all the binary variables with a high predominance of True values in all columns except in the suicide column. As well we can see in our quantitative variables the quartiles, mean and mode, with a predominance of the value 0 and a wide range.

```
> summary(data$property)
   Mode   FALSE    TRUE    NA's
logical   78731  102927   28048
> summary(data$crit1)
   Mode   FALSE    TRUE
logical    2461  207245
> summary(data$crit2)
   Mode   FALSE    TRUE
logical    1407  208299
> summary(data$crit3)
   Mode   FALSE    TRUE
logical   28049  181657
> summary(data$success)
   Mode   FALSE    TRUE
logical   24404  185302
> summary(data$suicide)
   Mode   FALSE    TRUE
logical  202268    7438
> summary(data$total_wound)
    Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
   0.000    0.000    0.000    2.884    2.000 10878.000
> summary(data$propvalue)
     Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
0.000e+00 0.000e+00 0.000e+00 4.039e+04 0.000e+00 2.700e+09
> summary(data$total_kill)
    Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
   0.000    0.000    0.000    2.663    2.000 1700.000
>
```

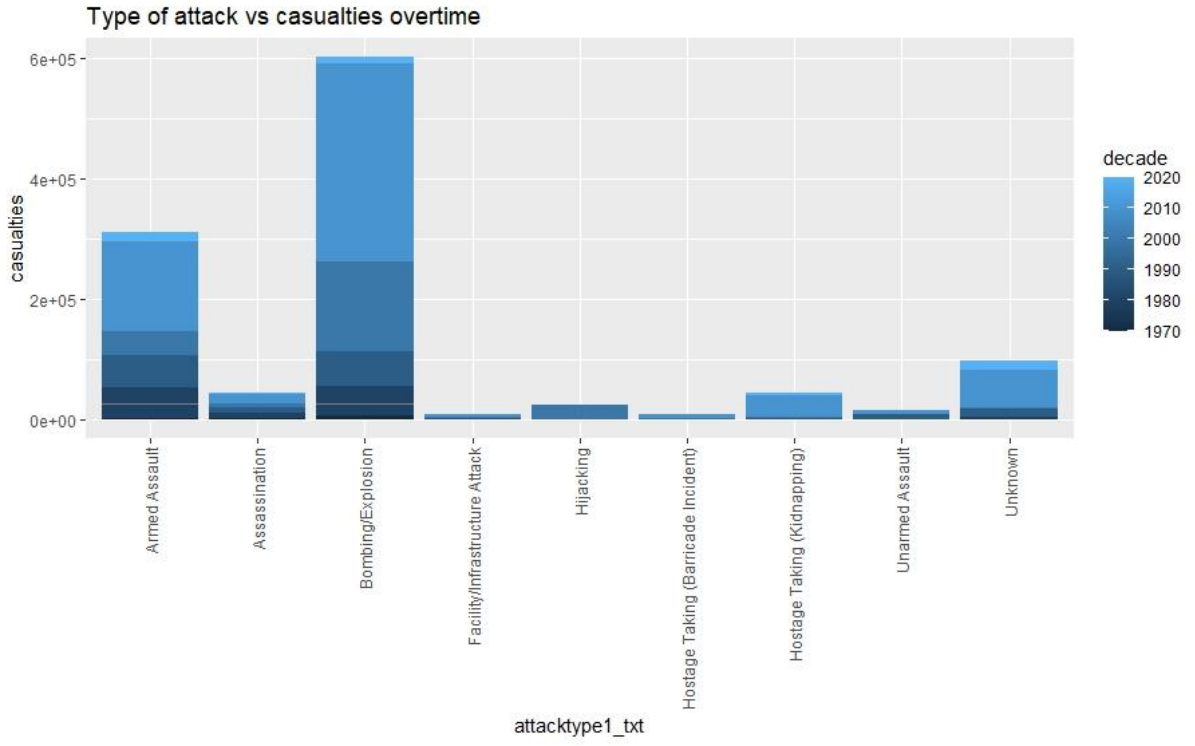
A different methodology was employed to recognize modes across all columns, specifically addressing missing values coded as "unknown." We took the approach of identifying the second most recurring value with a different method. We used this method on gname, Attacker group, and propextent_txt, a range of material damage of the attack.

```
> #Get measures of center
> getMode(data$country_txt)
[1] "Iraq"
> getMode(data$region_txt)
[1] "Middle East & North Africa"
> getMode(data$provstate)
[1] "Baghdad"
> getSecondMode(data$city)
[1] "Baghdad"
> getMode(data$attacktype1_txt)
[1] "Bombing/Explosion"
> getMode(data$targtype1_txt)
[1] "Private Citizens & Property"
> getMode(data$target1)
[1] "Civilians"
> getMode(data$natlty1_txt)
[1] "Iraq"
> getSecondMode(data$gname)
[1] "Taliban"
> getMode(data$weaptype1_txt)
[1] "Explosives"
> getMode(data$weapsubtype1_txt)
[1] "Unknown Explosive Type"
> data$property <- as.logical(data$property)
> getSecondMode(data$propextent_txt)
[1] "Minor (likely < $1 million)"
> getMode(data$total_kill)
[1] 0
> getMode(data$total_wound)
[1] 0
>
```

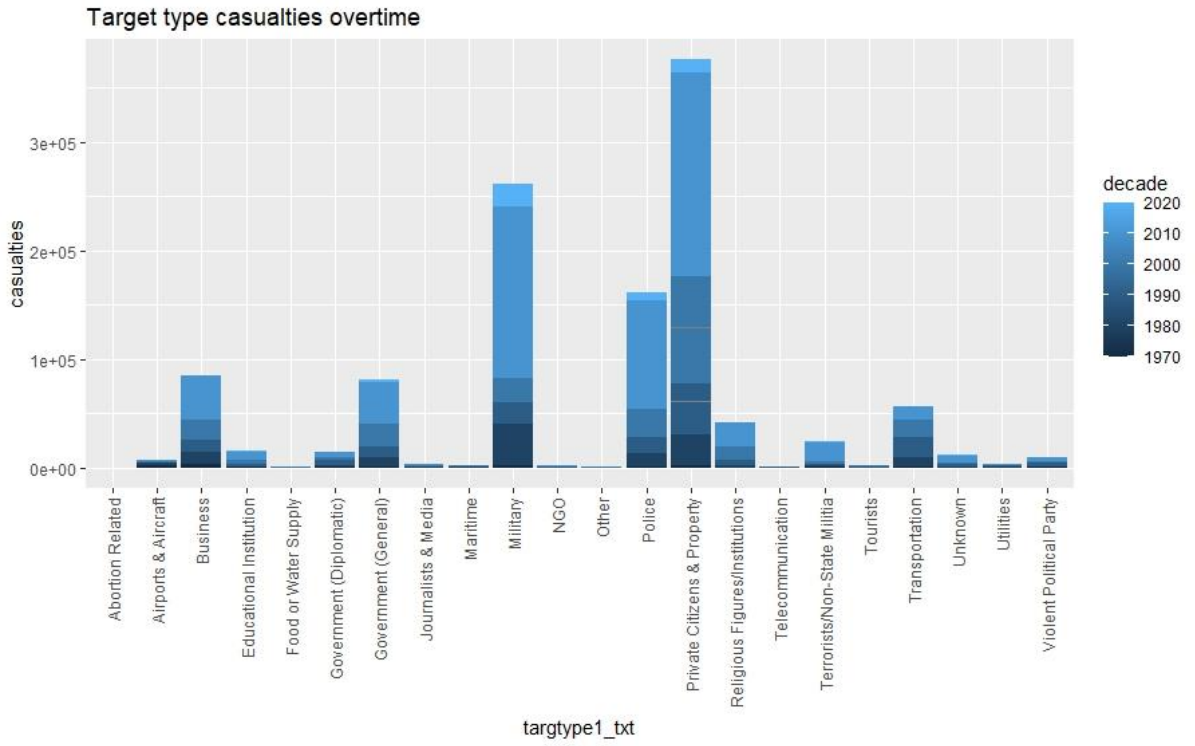
In the next graph we can see the terrorist attacks by region overtime, we observe that the number of casualties experienced a hight increase in certain regions in last three decades being the regions with more casualties, Middle East and north Africa, South Asia and Sub-Saharan Africa.



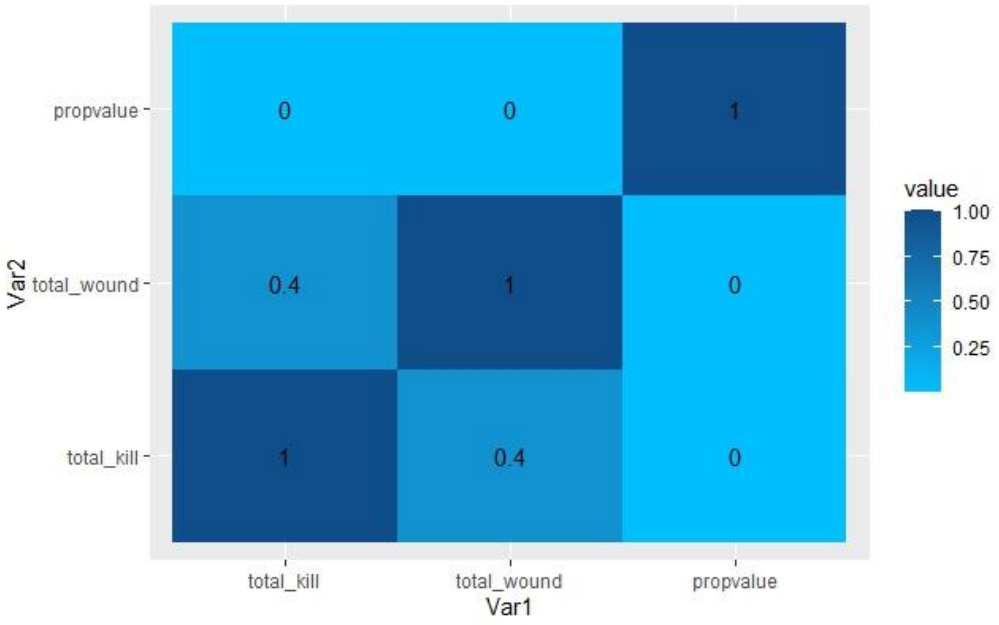
In the next graph we can see that the types of attack with more casualties and the evolution in time. The type of attack that generated the most casualties was Bombing/Explosion followed by Armed Assault.



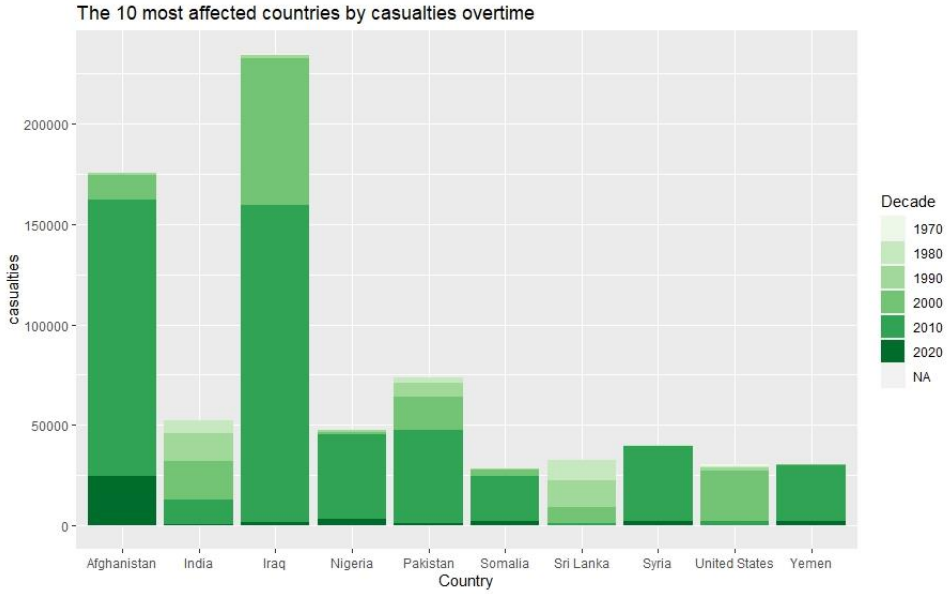
Whiting the next graph we can analyse that the most casualties were target to Private Citizens and property, followed by Military and police respectively, and followed Business and Government.



The next graph shows the correlation between the numeric variables, presenting a low correlation coefficient, being 0 between propvalue and total_kill and 0.4 between total_wound and total_kill. We were expecting a higher coefficient between total_kill and total_wound.



When dividing the data by country we separated the top 10 countries with the most casualties, Iraq and Afghanistan surpass by far the rest of the countries in number of casualties.



After this getting this information from the data we decided to use person correlation using chi-squared, showing a p value in the order of x10e-16, showing a high correlation between some categorical variables. The next variables were highly related, attackType with country, gname with country and attacktype1_txt and target1.

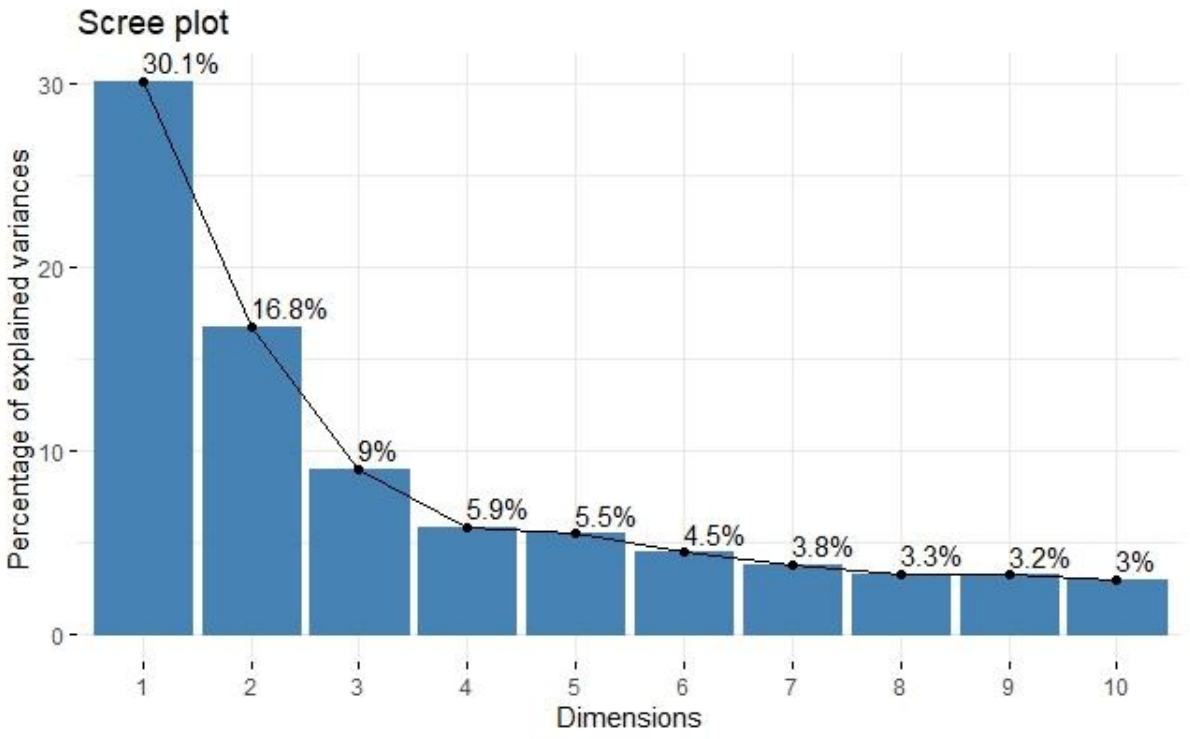
Principal Component Analysis

Principal Component Analysis, PCA, shows vital for handling "wide" datasets where plotting raw data is impractical. It transforms numerous correlated variables into a set of abstract uncorrelated principal components.

PCA achieves this by calculating indices based on the variance of the values within the dataset. The indices, known as principal components, are a linear combination of the original variables and each of them captures the maximum variability possible in the data. PCA improves visualization and analysis by reducing the dimensionality. (Datacamp, 2023)

From using summarize after performing PCA, we can notice that around 67% of the variance is represented by the first five principal components. This would be a reduction from 23 to 5 variables this represents a 79% reduction in the number of variables. After, the bar chart provides us with a visual representation of the relationship between the first ten components and the value of the proportion of variance.

```
> summary(data.pca)
Importance of components:
                          Comp.1    Comp.2    Comp.3    Comp.4    Comp.5    Comp.6    Comp.7    Comp.8    Comp.9   Comp.10   Comp.11
Standard deviation     0.6285810 0.4690745 0.34321953 0.27768967 0.26820726 0.2418837 0.22358312 0.20743650 0.20604929 0.19781926 0.18344060
Proportion of Variance 0.3009588 0.1675978 0.08972812 0.05873595 0.05479307 0.0445654 0.03807699 0.03277592 0.03233902 0.02980724 0.02563159
Cumulative Proportion  0.3009588 0.4685566 0.55828467 0.61702062 0.67181369 0.7163791 0.75445608 0.78723201 0.81957102 0.84937826 0.87500985
                          Comp.12    Comp.13    Comp.14    Comp.15    Comp.16    Comp.17   Comp.18     Comp.19     Comp.20      Comp.21
Standard deviation     0.17266797 0.16830175 0.15729883 0.13847945 0.13437139 0.12941399 0.1154637 0.097222399 0.066009041 8.215709e-03
Proportion of Variance 0.02270953 0.02157555 0.01884671 0.01460681 0.01375302 0.01275696 0.0101549 0.007199747 0.003318879 5.141321e-05
Cumulative Proportion  0.89771937 0.91929492 0.93814163 0.95274844 0.96650146 0.97925842 0.9894133 0.996613069 0.999931948 9.999834e-01
                           Comp.22      Comp.23
Standard deviation     4.673812e-03 5.587935e-09
Proportion of Variance 1.663899e-05 2.378413e-17
Cumulative Proportion  1.000000e+00 1.000000e+00
```



Conclusion

For the scope of this project, we were asked to choose a data set, we decided to use a big data set with too many variables. We decided to use this data set because we wanted to approach a real database as much as possible. From the start the data set was complex with many variables that were coded that we decided to take out as part of the CRISP-DM, we iterated many times throughout the Data understanding phase and Data Preparation Phase. For the scope of this project, we believe that the goals were presumptuous, being the data set not appropriate for this kind of project. When trying to understand each variable and preparing the data set took longer than we believed at the start of the project and we runed out of time to perform a robust exploratory data analysis. We had time to analyse some correlations and show some insides like the increase of the casualties of the attacks over time, but we did not extract as many insides from the data as we expected. As well we did not have time to include in our report the explanation about the distributions.

References

DataCamp. 2022. *Principal Component Analysis in R Tutorial*. [Online]. https://www.datacamp.com/tutorial/pca-analysis-r [Accessed 3 December 2023].

START (National Consortium for the Study of Terrorism and Responses to Terrorism). 2022. *Global Terrorism Database, 1970 - 2020* [data file]. https://www.start.umd.edu/gtd [Accessed 3 December 2023].

Appendix

R Files, Data set and extra materials

R script:

```
#PREPARATION
setwd("~/College Classes/CCT/Data Exploration & Preparation/CA1")
#Loading the data set
data <- read.csv("./STATS.Terrorism/globalterrorismdb_0522dist.csv")
View(data)
variables <- colnames(data)
summary(data)
variables

#install.packages("dplyr")
library(dplyr)

#Cleaning of explanatory text variables
data <- select(data, -location, -summary,-motive, -weapdetail, -propcomment, -addnotes, -scite1, -scite2, -scite3)

#Creating a single column for date yyyy/mm/dd
date <- as.Date(paste(data$iyear, data$imonth, data$iday, sep = "-"))
data <- data %>%
  mutate(date = date) %>%
  select(eventid, date, everything())
data <- select(data, -iyear, -imonth, -iday)
data <- select(data, -approxdate)

#Cleaning variables that are not part of our analysis
data <- select(data, -latitude, -longitude, -specificity, -nkillus, -nwoundus, -nhostkidus, -ransomamtus, -ransomamt, -ransompaidus, -dbsource)

#Replace empty spaces with NA
data <- data %>% mutate_at(-c(1, 2), ~ ifelse(. == "", NA, .))

#Remove columns with more than 80% missing values excluding some columns we want to keep
columns_to_exclude <- c("eventid", "propextent_txt", "propvalue", "nkillter", "nwoundte")
data_subframe <- select(data, columns_to_exclude)
data_cleaned <- data %>%
  select_if(~ sum(!is.na(.)) / length(.) > 0.8)
View(data_subframe)
View(data_cleaned)
View(data)
data_merge <- merge(data_cleaned, data_subframe, by = "eventid", all = TRUE)
View(data_merge)
```

```
data <- data_merge

#Remove columns codes
data_cleaned <- select(data, -country, -region,-attacktype1, -targtype1, -targsubtype1, -natlty1, -weaptype1, -weapsubtype1)
data <- data_cleaned

#Remove columns that are not part of our analysis
data_cleaned <- select(data, -extended, -vicinity, -doubtterr, -ishostkid, -multiple, -guncertain1, -individual, -INT_LOG, -INT_IDEO, -INT_MISC, -INT_ANY)
data <- data_cleaned

#Decode variables
data_cleaned$crit1 <- as.logical(data$crit1)
data_cleaned$crit2 <- as.logical(data$crit2)
data_cleaned$crit3 <- as.logical(data$crit3)
data_cleaned$success <- as.logical(data$success)
data_cleaned$suicide <- as.logical(data$suicide)
data_cleaned$property <- ifelse(data$property == 1, TRUE, ifelse(data$property == 0, FALSE, "Unknown"))
data <- data_cleaned
summary(data)

# nwound & nkill NA to 0
data$nkill[is.na(data$nkill)] <- 0
data$nkillter[is.na(data$nkillter)] <- 0
data$nwound[is.na(data$nwound)] <- 0
data$nwoundte[is.na(data$nwoundte)] <- 0
data$propvalue[is.na(data$propvalue)] <- 0

#Merge of wound and killed columns
data$total_wound <- data$nwound + data$nwoundte
data$total_kill <- data$nkill + data$nkillter
data <- select(data, -nwound, -nwoundte, -nkillter, -nkill)
count_0_kill <- table(data$total_kill)["0"]
count_0_wound <- table(data$total_wound)["0"]

#Data propvalue -99 to 0 and counting them
#http://127.0.0.1:41253/graphics/plot_zoom_png?width=1920&height=1009
count_prop_unknown <- table(data$propvalue)["-99"]
data$propvalue <- ifelse(data$propvalue == -99, 0, data$propvalue)
count_0_prop <- table(data$propvalue)["0"]

#Checking for outliers
library(ggplot2)
ggplot(data = data, mapping = aes(y = total_wound)) + geom_boxplot()
ggplot(data = data, mapping = aes(y = total_kill)) + geom_boxplot()
ggplot(data = data, mapping = aes(y = propvalue)) + geom_boxplot()
```

```
summary(data)
propValue_max <- data[which.max(data$propvalue), ]
total_kill_max <- data[which.max(data$total_kill), ]
total_wound_max <- data[which.max(data$total_wound), ]

#Export the Data set
write.csv(data, "./STATS.Terrorism/globalterrorism_prepared.csv", row.names=FALSE)


#EDA
#Reload the Data set
data <- read.csv("./STATS.Terrorism/globalterrorism_prepared.csv")

year <- as.POSIXct(data$date, format = "%Y-%m-%d")
year <- format(year, format="%Y")
decade <- floor(as.numeric(year) / 10) * 10
data$decade <- decade

data$casualties <- data$total_kill + data$total_wound

library(ggplot2)
library(dplyr)
library(reshape2)

#Min-Max normalization
#library(caret)
#library(lattice)

#process <- preProcess(as.data.frame(data$propvalue), method=c("range"))
#norm_scale <- predict(process, as.data.frame(data$propvalue))
#data$propvalue2 <- norm_scale

#process <- preProcess(as.data.frame(data$total_kill), method=c("range"))
#norm_scale <- predict(process, as.data.frame(data$total_kill))
#data$total_kill2 <- norm_scale

#process <- preProcess(as.data.frame(data$total_wound), method=c("range"))
#norm_scale <- predict(process, as.data.frame(data$total_wound))
#data$total_wound2 <- norm_scale

#Plots
ggplot(data = data, mapping = aes(x = region_txt, y = casualties, fill = decade)) +
  labs(title = "Terrorist attacks by region overtime", x = "Region") +
  geom_bar(position="stack", stat="identity") +
  theme(axis.text.x=element_text(angle=90,hjust=1,vjust=0.5))

ggplot(data = data, mapping = aes( x = attacktype1_txt, y = casualties, fill = decade)) +
  labs(title = "Type of attack vs casualties overtime",) +
```

```
  geom_bar(position="stack", stat="identity") +
  theme(axis.text.x=element_text(angle=90,hjust=1,vjust=0.5))

ggplot(data = data, mapping = aes( x = targtype1_txt, y = casualties, fill = decade)) +
  labs(title = "Target type casualties overtime",) +
  geom_bar(position="stack", stat="identity") +
  theme(axis.text.x=element_text(angle=90,hjust=1,vjust=0.5))

summary(data)

#Create mode function https://www.tutorialspoint.com/r/r_mean_median_mode.htm
getMode <- function(v) {
  uniqv <- unique(v)
  uniqv[which.max(tabulate(match(v, uniqv)))]
}

getSecondMode <- function(v) {
  uniqv <- unique(v)
  freq <- tabulate(match(v, uniqv))
  second_max_index <- which(freq == sort(freq, decreasing = TRUE)[2])
  uniqv[second_max_index]
}

#Get measures of center
getMode(data$country_txt)
getMode(data$region_txt)
getMode(data$provstate)
getSecondMode(data$city)
getMode(data$attacktype1_txt)
getMode(data$targtype1_txt)
getMode(data$target1)
getMode(data$natlty1_txt)
getSecondMode(data$gname)
getMode(data$weaptype1_txt)
getMode(data$weapsubtype1_txt)
data$property <- as.logical(data$property)
getSecondMode(data$propextent_txt)
getMode(data$total_kill)
getMode(data$total_wound)
summary(data$property)
summary(data$crit1)
summary(data$crit2)
summary(data$crit3)
summary(data$success)
summary(data$suicide)
summary(data$total_wound)
summary(data$propvalue)
summary(data$total_kill)
```

```
colnames(data)

#Correlation between numeric variables
#https://www.geeksforgeeks.org/how-to-calculate-correlation-between-multiple-variables-in-r/
cormat <- cor(data[, c("total_kill", "total_wound", "propvalue")])
melted_cormat <- melt(cormat)

# Heatmap http://www.sthda.com/english/wiki/ggplot2-quick-correlation-matrix-heatmap-r-software-and-data-visualization
# https://stackoverflow.com/questions/14290364/create-heatmap-with-values-from-matrix-in-ggplot2

ggplot(melted_cormat, aes(Var1, Var2)) +
  geom_tile(aes(fill = value)) +
  geom_text(aes(label = round(value, 1))) +
  scale_fill_gradient(low = "deepskyblue", high = "dodgerblue4")

#sort by casualties, show the most affected countries
Casualties_by_country <- data %>%
  group_by(country_txt) %>%
  summarise(casualties=sum(casualties))

Casualties_by_country_decade <- data %>%
  mutate(decade = as.character(decade)) %>%
  group_by(country_txt, decade) %>%
  summarise(casualties = sum(casualties))

sorted_casualties_by_country <- Casualties_by_country[order(Casualties_by_country$casualties, decreasing=TRUE),]
slice <- slice(sorted_casualties_by_country, 1:10)

sorted_casualties_by_country_decade <- Casualties_by_country_decade[order(Casualties_by_country_decade$casualties, decreasing=TRUE),]
slice2 <- sorted_casualties_by_country_decade[sorted_casualties_by_country_decade$country_txt %in% slice$country_txt, ]

ggplot(data = slice2, aes(fill=as.character(decade), y=casualties, x=country_txt)) +
  labs(title = "The 10 most affected countries by casualties overtime", x = "Country", fill = "Decade") +
  geom_bar(position="stack", stat="identity") + scale_fill_brewer(palette = "set2")

library(psych)
description <- describe(data)

#coding region
region <- unique(data$region_txt)
code_region <- c(1:12)
names(code_region) = region
data$region <- code_region[data$region_txt]
```

```
#coding country
country <- unique(data$country_txt)
code_country <- c(1:204)
names(code_country) = country
data$country <- code_country[data$country_txt]

#coding attacktype
attacktype <- unique(data$attacktype1_txt)
code_attacktype <- c(1:9)
names(code_attacktype) = attacktype
data$attacktype <- code_attacktype[data$attacktype1_txt]

#chisq test to evaluate the relationship between two categorical variables
#Create a contingency table
chi_country_attacktype<- table(data$country_txt, data$attacktype1_txt)
#mytable2 <- table(data$country, data$attacktype)
chi_country_attacktype<- table(data$country_txt, data$attacktype1_txt)
#Perform chi-squared test
chisq.test(chi_country_attacktype)
#chisq.test(mytable2)

chi_country_txt_gname<- table(data$gname, data$country_txt)
chisq.test(chi_country_txt_gname)

chi_attacktype_target1<- table(data$attacktype1_txt, data$target1)
chisq.test(chi_attacktype_target1)

chi_attacktype_target1<- table(data$attacktype1_txt, data$target1)
chisq.test(chi_attacktype_target1)


#PCA https://www.datacamp.com/tutorial/pca-analysis-r
str(data)
#coding natlty1_txt
natlty1_txt  <- unique(data$natlty1_txt)
code_natlty1_txt <- c(1:length(natlty1_txt))
names(code_natlty1_txt) = natlty1_txt
data$natlty1 <- code_natlty1_txt[data$natlty1_txt]

#coding propextent_txt
propextent_txt  <- unique(data$propextent_txt)
code_propextent_txt <- c(1:length(propextent_txt))
names(code_propextent_txt) = propextent_txt
data$propextent <- code_propextent_txt[data$propextent_txt]

#coding weapsubtype1_txt
weapsubtype1_txt  <- unique(data$weapsubtype1_txt)
code_weapsubtype1_txt <- c(1:length(weapsubtype1_txt))
```

```
names(code_weapsubtype1_txt) = weapsubtype1_txt
data$weapsubtype1_txt <- code_weapsubtype1_txt[data$weapsubtype1_txt]

#coding weaptype1_txt
weaptype1_txt  <- unique(data$weaptype1_txt)
code_weaptype1_txt <- c(1:length(weaptype1_txt))
names(code_weaptype1_txt) = weaptype1_txt
data$weaptype1 <- code_weaptype1_txt[data$weaptype1_txt]

#coding gname
gname  <- unique(data$gname)
code_gname <- c(1:length(gname))
names(code_gname) = gname
data$code_gname <- code_gname[data$gname]

#coding target1
target1   <- unique(data$target1)
code_target1 <- c(1:length(target1))
names(code_target1) = target1
data$code_target1 <- code_target1[data$target1]

#coding targsubtype1_txt
targsubtype1_txt   <- unique(data$targsubtype1_txt)
code_targsubtype1_txt <- c(1:length(targsubtype1_txt))
names(code_targsubtype1_txt) = targsubtype1_txt
data$targsubtype1 <- code_targsubtype1_txt[data$targsubtype1_txt]

#coding targtype1_txt
targtype1_txt  <- unique(data$targtype1_txt)
code_targtype1_txt <- c(1:length(targtype1_txt))
names(code_targtype1_txt) = targtype1_txt
data$targtype1 <- code_targtype1_txt[data$targtype1_txt]

data_cleaned <- select(data, -natlty1_txt, -region_txt, -date, -country_txt, -provstate, -
attacktype1_txt, -propextent_txt, -weaptype1_txt, -gname, -target1, -targsubtype1_txt, -
targtype1_txt, -city)

str(data_cleaned)

#take out binary variables
data_cleaned <- select(data, -crit1, -crit2, -crit3, -success, -suicide, -property)

PCA2 <- prcomp(na.omit(data_cleaned), scale = TRUE, center = TRUE, tol = 0)
PCA2
summary(PCA2)

plot(PCA2, main="",col="dodgerblue3")
title(main="Principal Components Importance for Students")
```

```
install.packages("corrr")
library(corrr)
install.packages("ggcorrplot")
library(ggcorrplot)
numerical_data <- na.omit(data_cleaned)
data_normalized <- scale(numerical_data)
corr_matrix <- cor(data_normalized)
ggcorrplot(corr_matrix)
data.pca <- princomp(corr_matrix)
summary(data.pca)
data.pca$loadings[, 1:7]

library(factoextra)
fviz_eig(data.pca, addlabels = TRUE)
```