Universitat Autònoma de Barcelona

Facultat de Ciències i Biociències

Anàlisi de Dades Complexes World Happiness report

Autor:
Carla Ardiaca: 1633638

Contents

1 Abstract

The purpose of this study of the World Happiness Report is studying the correlation between the variables of the Report as GDP per capita, the freedom of the population, the healthy expectancy of life... With this we want to answer the question "What make us happy?" using methods as bootstrap and backward selection of lineal models in our dataset done with a survey that was answered by 130.000 people. Our analysis shows a strong correlation between the Score of Happiness and the GDP per capita, Freedom to make life choices and Social Support. So we conclude that happiness is mostly achieved with a high value of these variables.

2 Introduction

The main objective in our life should be always the same: be as happy as we can. But, how do we achieve this happiness? What countries are happier? Which are the things that make us happier? In this report, I will try to give an answer to these questions.

In order to do this, I have chosen a dataset which contains 156 different countries ordered by their happiness score. The dataset also contains other important parameters as it can be the generosity of each country, the freedom to make life choices, its social support, the GDP per capita of the country or how it is perceived the corruption in it.

I have chosen this Dataset because I really love to travel around the word and go to different countries. Since I was a child, I have been interested in Geography and what makes a country better than another talking about its culture and style of life, so, this is a perfect opportunity to know what is better for a country.

I really want to study what happen with all the parameters of the dataset, but I want to focus more on one. It is said that money makes us happy. With this analysis, I want to prove if this hypothesis is really true and if this is the only thing we have to take into account when we talk about happiness or the other parameters are also important.

3 Description Dataset

The Dataset I have chosen is called World Happiness Report, and it is a Dataset with 156 rows (each row is a different country) with the happiness scored according to economic production, social support, etc.

The dataset has a total of 9 columns. The first column lets us now the Overall rank (it sorts the countries by their happiness score from 1 to 156). The second column contains the name of the country or the region and the third one the score of happiness of each country. The other 6 columns are the different parameters that affect the happiness score:

- GDP per capita
- Social support
- Healthy life expectancy
- Freedom to make life choices
- Generosity
- Perception of corruption

It seems strange to be able to measure the happiness of a country, so, how has been the happiness measured in this Dataset? The data of the World Happiness Report has been obtained from the Gallup World Poll (GWP) surveys. In this survey they ask the users to to think of a ladder, where a 10 is the best possible life they can have and 0 is the worst one. The number of people surveyed is more or less 1.000 people for each country, so 156.000 people have been surveyed to do this Dataset of the World Happiness Report of 2019.

Dystopia is an imaginary country that has on the least happy people. We "create" Dystopia to establish a benchmark against which all countries can be favourably compared (there is no country with worse values than Dystopia). The 6 columns mentioned before estimate the extent to which each of the six parameters contribute to making life evaluations higher in each country than they are in Dystopia. With this imaginary country, no variable will be negative. The Dystopia has no impact on the total score reported for each country, but they do explain why some countries rank higher than others.

How have been measured the different variables of the Dataset?

- **GDP per capita:** It has been taken from the World Development Indicators released by the World Bank. The equation uses the natural log of GDP per capita, as this form fits the data significantly better.

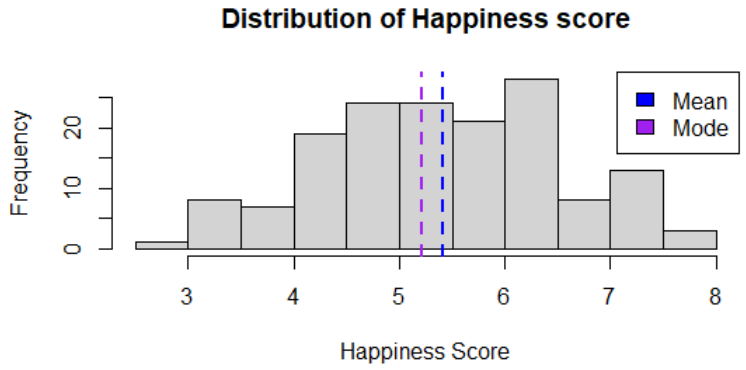
- **Social support:** Social Support refers of the support provided by members of social networks to an individual to the questions "If you were in trouble, do you have relatives or friends you can count on to help you whenever you need them, or not?"
- **Healthy life expectancy:** It has been taken from the World Health Organization Global Health Observatory data.
- **Freedom to make life choices:** is the national average of a dichotomous response (0 o 1) in the GWP survey to the question "Are you satisfied or dissatisfied with your freedom to choose what you do with your life?"
- **Generosity:** is the residual of regressing the national average of GWP responses to the question "Have you donated money to a charity in the past month?" on GDP per capita.
- **Perception of corruption:** is the national average of a dichotomous response (0 o 1) in the GWP survey to the questions "Is corruption widespread throughout the government or not?" and "Is corruption widespread within businesses or not?"

  Trough all this work, I will be working with all these variables (the six last columns) and the score of happiness. The Overall Rank and the Country of its row won't be useful for analysing the dataset although, to do some graphics, they may be used.

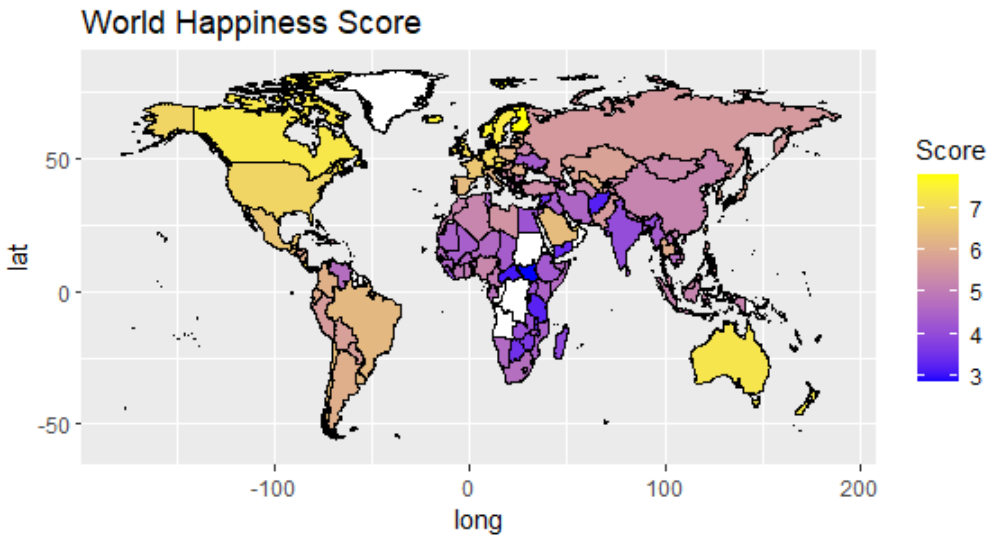
4 Analysis

4.1 Initial observations of the Happiness around the World

First of all, let's see how Happiness is distributed with a histogram of the score of happiness around the world.

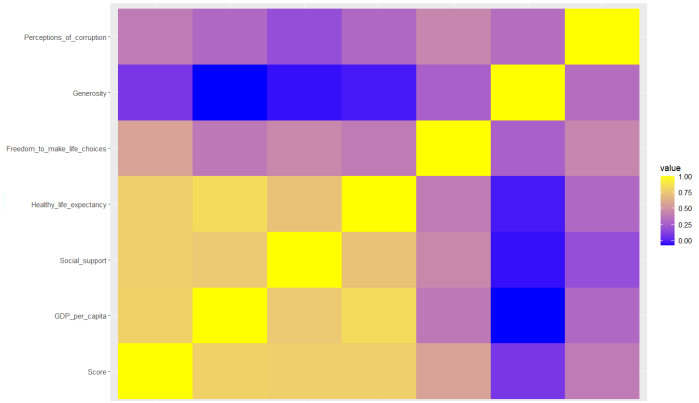


How is Happiness distributed around the world? To answer this question I have decided to do a plot of a world map and paint the countries with different colors according them score value. White countries aren't in this Dataset or haven't been detected for the "map_data" function of R (as it happens with the country Congo (Kinshasa) for example).



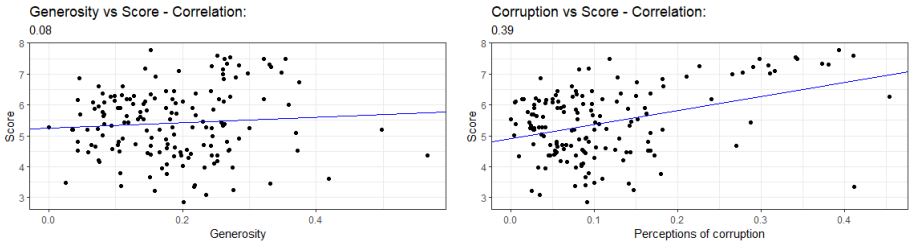
4.2 Exploring Happiness data correlation

In this dataset analysis we are focused on studying the correlations between the different variables to know which have more effect on the score of happiness. For this reason the first thing I have done is do a correlation plot between all the variables in the dataset.



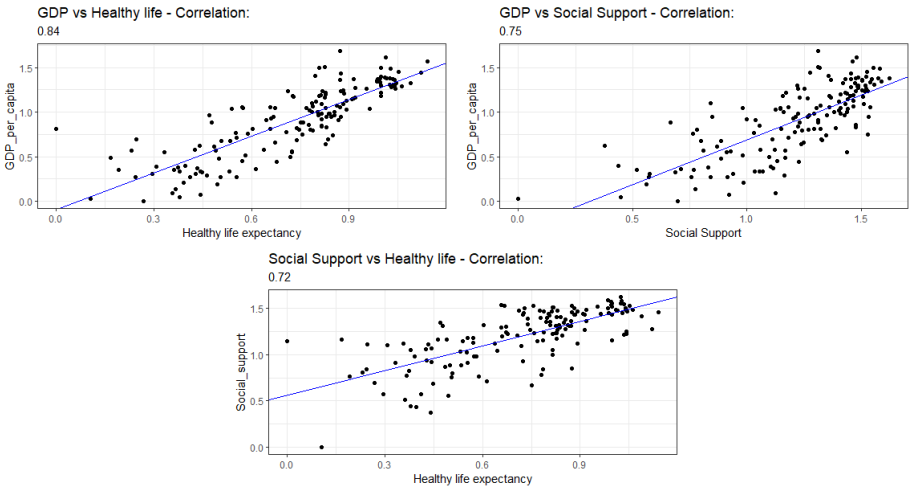
This correlation plot shows us that the variables as "GDP per capita", "social support" and "health life expectancy" have a higher correlation than the others with the score of happiness. We clearly also see that "generosity" has less relationship with the score so, it is very probable that generosity is a non-significant variable.

The following graphics show us the correlation between the score of happiness and the other variables of the Dataset.



The results of this graphic support the results of the correlation plot, we can see that "GDP per capita", "Social Support", "Healthy life expectancy" and "Freedom to make life choices" correlate better than "Generosity" and "Perception of Corruption".

As it is seen in the correlation plot, "GDP per capita", "healthy life expectancy" and "social support" have a high correlation between them. The forthcoming plots let us see the correlation between these three variables.



After doing these plots, we clearly see that these three variables are correlated each other and they go hand in hand.

4.3 Backward selection

Backward selection consists of eliminating the variables that are not significant for our model. We start with the full model (this model contains all the variables in the dataset). We remove the least informative variable that is the most non-significant variables that has the highest p-value (bigger than 0.05). Once we have removed this variable, if there are more non-significant parameters we do all these steps. If not, we stop and we achieve a model where all the variables in it are important.

Before to do any other analysis I will use backward selection to remove all this non-significant variables, thus this will do our further analysis more accurate it and easier.

4.4 Bootstrap method

In the second part of the analysis, I will use bootstrap method. This method allows us to estimate the sampling distribution of a statistic, in our case, we will calculate 95% confidence intervals. What bootstrap let us do is resembling generating a large number of bootstrap samples. For each sample we can calculate the mean, the standard deviation... Thanks to this we can have more accurate values with the confidence intervals. The number or iterations that will be done is 10.000.

4.4.1 Parametric bootstrap

First of all, we must decide the distribution of our sample. To do this I have looked the histogram above, and I have assumed that was a Normal distribution. Here, the estimated parameters are estimated from our original sample of the Dataset. Parametric bootstrap is normally more efficient and matches better the distribution chosen.The number or iterations that will be done is 10.000.

It is important to mention that as my response variance goes to 0 to 10, and, for this reason it is a limited response variable so, Normal Distribution may not fit perfectly. To know if it fits our data we can use Shapiro Test that it is used to know if our data is normally distributed or not. Our null hypothesis is that the it is Normally Distributed so, if p-value is greater than 0.05 we accept the null hypothesis. Shapiro test in R:

shapiro.test(data_hap$Score)
Shapiro-Wilk normality test
data: data_hap$Score
W = 0.9872, **p-value = 0.1633**

As p-value is higher than 0.05, we conclude that our data, in this case, the score of happiness is Normally distributed, so this is the distribution that we will assume in the parametric bootstrap.

4.4.2 Non-parametric bootstrap

With non-parametric bootstrap we don't assume any distribution of the sample to do the bootstrap method. In this method bootstrap samples are generated with resembling the original sample. This method is more flexible and robust and can be used when we don't know the distribution of the data.

5 Results of the analysis of the dat

5.1 We build our lineal model

We do a lineal regression to know how the six different variables affect the score of happiness:

```
Call:
lm(formula = Score ~ 1 + GDP_per_capita + Social_support + Healthy_life_expectancy +
    Freedom_to_make_life_choices + Generosity + Perceptions_of_corruption,
    data = data_hap)

Residuals:
     Min       1Q   Median       3Q      Max
-1.75304 -0.35306  0.05703  0.36695  1.19059

Coefficients:
                             Estimate Std. Error t value Pr(>|t|)
(Intercept)                    1.7952     0.2111   8.505 1.77e-14 ***
GDP_per_capita                 0.7754     0.2182   3.553 0.000510 ***
Social_support                 1.1242     0.2369   4.745 4.83e-06 ***
Healthy_life_expectancy        1.0781     0.3345   3.223 0.001560 **
Freedom_to_make_life_choices   1.4548     0.3753   3.876 0.000159 ***
Generosity                     0.4898     0.4977   0.984 0.326709
Perceptions_of_corruption      0.9723     0.5424   1.793 0.075053 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.5335 on 149 degrees of freedom
Multiple R-squared:  0.7792,    Adjusted R-squared:  0.7703
F-statistic: 87.62 on 6 and 149 DF,  p-value: < 2.2e-16
```

So, our lineal regression follows the equation:

$$\textbf{Score} = 1.8 + 0.78\text{*GDP} + 1.12\text{*Social\_support} + 1.08\text{*Health} + 1.45\text{*Freedom} + 0.49\text{*Generosity} + 0.97\text{*Corruption}$$

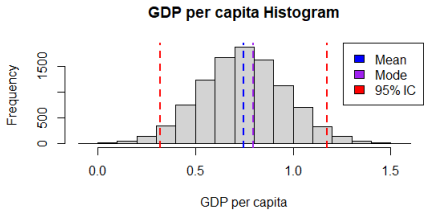
We see that generosity is a non-significant predictor of happiness at the 0.1 cutoff, while perceptions of corruption is a non-significant at the 0.05 cutoff level. We apply backward selection to remove non-significant variables, first we remove generosity that has the highest p-value. We call again "lm" function and we see that all the other variables are significant, so we finish backward selection method.

The following step to analyse our dataset is Bootstrap method that will be done without the generosity variable because, as we saw before, it is a non-significant variable.

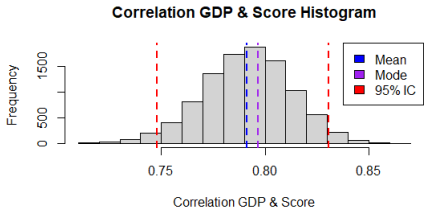
5.2 Results with Bootstrap method

5.2.1 Parametric Bootstrap:

- **GDP per capita:**

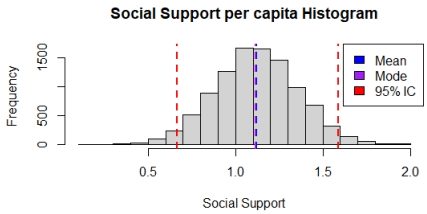


IC 95% [0.322,1.175]

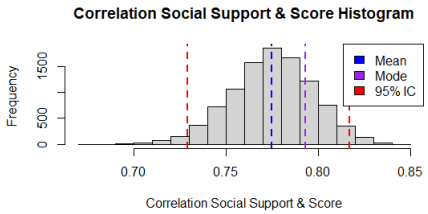


IC 95% [0.748, 0.831]

- **Social support:**

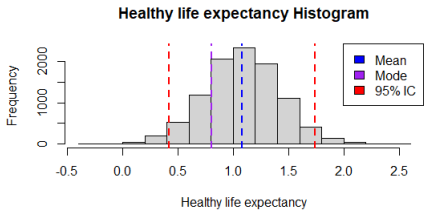


IC 95% [0.666, 1.587]

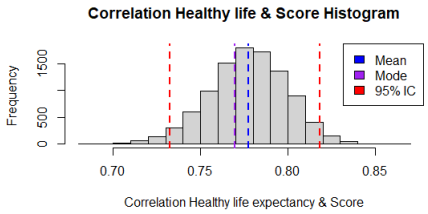


IC 95% [0.729, 0.817]

- **Healthy life expectancy:**

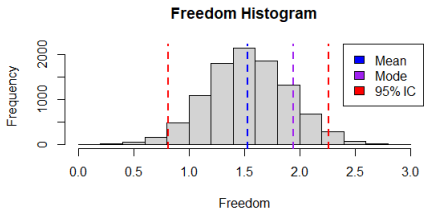


IC 95% [0.418, 1.741]

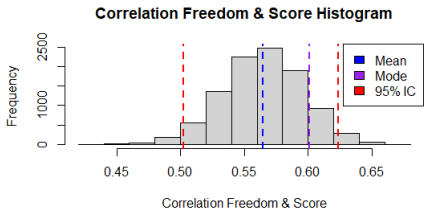


IC 95% [0.732, 0.818]

- **Freedom to make life choices:**

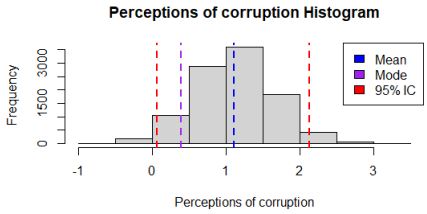


IC 95% [0.815, 2.264]

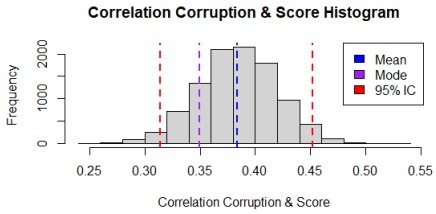


IC 95% [0.502, 0.6239]

- **Perceptions of corruption:**



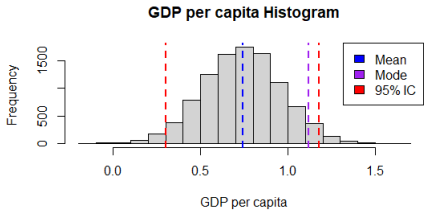
IC 95% [0.068, 2.133]



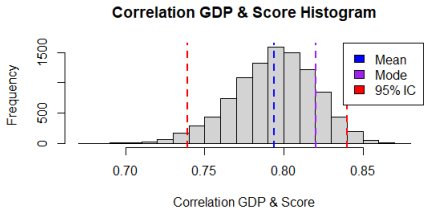
IC 95% [0.314, 0.451]

5.2.2 Non-Parametric Bootstrap:

- **GDP per capita:**

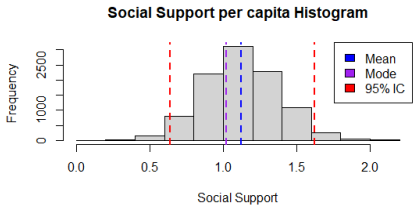


IC 95% [0.304, 1.179]

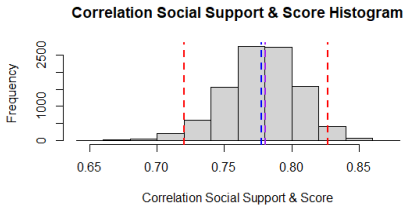


IC 95% [0.739, 0.840]

- **Social support:**

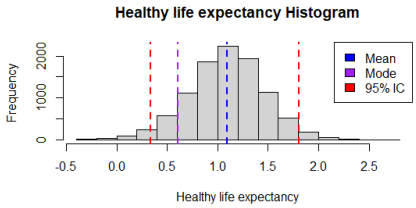


IC 95% [0.639, 1.623]

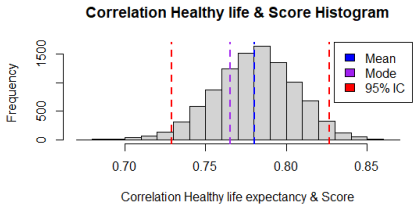


IC 95% [0.720, 0.827]

- **Healthy life expectancy:**

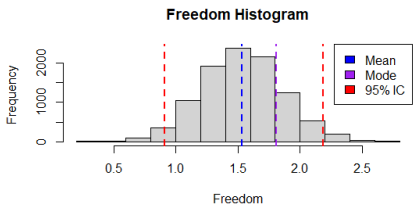


IC 95% [0.337, 1.809]

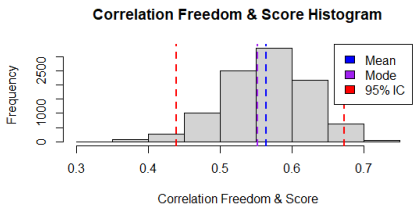


IC 95% [0.729, 0.826]

- **Freedom to make life choices:**

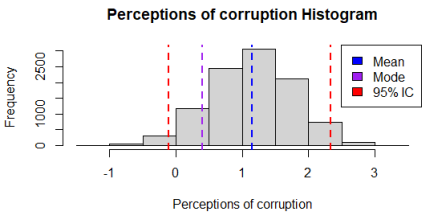


IC 95% [0.910, 2.185]

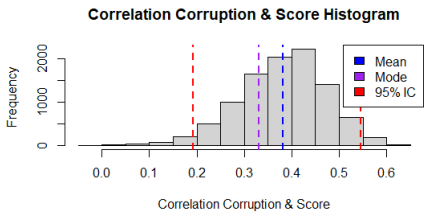


IC 95% [0.440 0.673]

- **Perceptions of corruption:**



IC 95% [-0.11, 2.327]



IC 95% [0.191 0.545]

5.2.3 Predictions of imaginary countries

As another result I will be playing with the values of variables to make predictions about hypothetical countries.

- **Best values found on the dataset for the variables:**
  - GDP per capita: 1.68
  - Social support: 1.62
  - Health life expectancy: 1.14
  - Freedom to make life choices: 0.63
  - Perceptions of corruption: 0.57
  - Generosity: 0.43

  Total Score Happiness: 7.83

  This prediction has been done with the best values of each variable in the Dataset. They have been found using the max function in R of all the variables. With the best values for the parameters we achieve a total Score of 7.83, a Score that is slightly better than Finland, the happiest country in the Dataset.

- **Average values found on the dataset for the variables:**
  - GDP per capita: 0.91
  - Social support: 1.21
  - Health life expectancy: 0.72
  - Freedom to make life choices: 0.39
  - Perceptions of corruption: 0.11
  - Generosity: 0.18

  Total Score Happiness: 5.40

  This prediction has been done with the average of the values of each variable in the Dataset. With these values for the parameters we achieve a total Score of 5.4, the average Score of all the Dataset.

- **Good values for significant variables and average values for non-significant variables:**
  - GDP per capita: 1.68
  - Social support: 1.62
  - Health life expectancy: 1.14
  - Freedom to make life choices: 0.63

- Perceptions of corruption: 0.11
- Generosity: 0.37

Total Score Happiness: 7.33

We assume that Generosity and Perception of Corruptions are non-significant parameters and the other variables are significant. We can see that non-significant variables aren't as important as the others in order to calculate the value of the Score.

- **Average values for significant variables and good values for non-significant variables:**
  - GDP per capita: 0.91
  - Social support: 1.21
  - Health life expectancy: 0.72
  - Freedom to make life choices: 0.39
  - Perceptions of corruption: 0.57
  - Generosity: 0.43

  Total Score Happiness: 5.86

  We assume that Generosity and Perception of Corruptions are non-significant values. We see that the values of significant variables are important to have a high Score.

- **Happiness score with the highest GDP value and the least happiest country values for the other variables:**
  - GDP per capita: 1.30
  - Social support: 0.58
  - Health life expectancy: 0.30
  - Freedom to make life choices: 0.01
  - Perceptions of corruption: 0.09
  - Generosity: 0.20

  Total Score Happiness: 4.09

  This is the result of the Score of Happiness with the highest GDP per capita (the money) of all the Dataset and the other values are from the least happy country (South Sudan, that had a Score of 2.85). We can see that money gives us more happiness, but the others parameters are also important because we don't even reach the average Score of happiness (5.40) of the different countries.

6 Conclusions and discussion

The first of all, I must underline that our response variable, as it is a number from 0 to 10, it is a limited response variable and may not follow a completely Normal Distribution. In this work, I have decided to use this distribution because it was the most similar and Shapirto test accepts it as a Normal Distribution. This may affect on the results and perhaps non-parametric bootstrap gives us more accurate results than the lineal model and the parametric bootstrap.

With the initial analysis we see than the level of happiness is different around the World and the levels of happiness are similar in each continent. Exploring the Happiness data correlation and the graphics of that section, we see a clear evidence that "GDP per capita", "Social Support" and "Healthy Life Expectancy" affect directly to the happiness of the people.

With a linear regression and backward selection, we discard Generosity as a variable that produces Happiness. Parametric bootstrapping it is used to corroborate the results of our lineal model so, we use it as a validation. With this validation, we see then if our sample follows a Normal Distribution, the intervals of confidence matches with our lineal model. With this method, we see that, except generosity variable all the other affect on our Happiness Score and "GDP", "Social Support" and "Healthy Life Expectancy" are the most important ones.

Non-parametric bootstrap method give us results slightly different. We conclude that this method is perhaps more reliable as it doesn't follow any distribution and fits better with our data. With this analysis, looking the confidence intervals of the variables, we deduce that either "Generosity" and "Perceptions of Corruption" are non-significant variables (the 95% IC for corruption is between [-0.11,2.3727]). In parametric bootstrap, "Perception of corruption" was a significant variable but not too much (their values aren't higher than 0.05 but it is closer), so it makes sense these results of non-parametric bootstrap. So "GDP per capita", "Social Support", "Healthy life expectancy" and "Freedom to make life choices" affect directly and correlate with the Happines Score of a country being the "Freedom" variable the less important of them.

The results of bootstrap seem to be consistent with the reality because countries with more GDP will normally have a better sanity system so health life expectancy will be also high. At the same time, it is normal that "Social Support" with a high GDP per capita and healthy life expectancy also increases.

To conclude and answer the initial questions we achieve the happiness when our GDP per capita and our level of health and Social Support are high enough. Having a life without money preoccupations, with a good level of health and feeling supported by others are the keys of being happy. The happiest countries are Findland and Denmark and we can see that their GDP, Health and Social Support are higher than other countries.

As I have said in the introduction, a lot of people say that money (GDP per capita in our study) make us happy and I wanted to prove if this hypothesis is really true. With this study we can conclude that money make us happy but it isn't the only important parameter altough it has a big correlation with the other important variables (Social Support and Health Life Expectancy). So, if we want to be happy, money can help us but we also have to consider the other parameters as be could see in the last prediction done in the previous section.

7 Bibliography

- Dataset World Happiness Report: https://www.kaggle.com/datasets/unsdsn/world-happiness?select=2017.csv
- Information about the Dataset: https://worldhappiness.report/
- Bootstrap method: https://stats.stackexchange.com/questions/199798/through-an-example-what-is-parametric-and-non-parametric-boostrap
- Notes of power 5 and 6 on Campus Virtual Anàlisi de Dades Complexes: https://e-aules.uab.cat/2022-23/course/view.php?id=20233
- Dataset modify to do this assigment: upload it in the Campus Virtual Assigment.

8 Appendix with the R scripts and data used for the analysis

This is the R script I have used to do this study. In the final assigment task I have upload the R code because here, wanting to make the understandable and fit it to the pdf, I have added some blank spaces that can make the code go wrong when executing it.

```
library(reshape2);   library(ggplot2);    library(dplyr)
library(purrr); library(ggcorrplot) # for correlation plots

#Read Datset
data_hap <-read.csv ("C:/Users/Carla/OneDrive/Escritorio/MatCAD/2n/2n semestre/
Anàlisi de Dades Complexa/treball final/archive/2019.csv")
summary(data_hap)
shapiro.test(data_hap$Score)

hap<- select(data_hap,Country_or_region, Score)
world <- world %>% filter(region != "Antarctica")
world <- fortify(world)

ggplot() +  geom_map(data=world, map=world, aes(x=long, y=lat, group=group,
        map_id=region), fill="white", colour="black") +  geom_map(data=hap,
        map=world, aes(fill=Score, map_id=Country_or_region),colour="black") +
  scale_fill_continuous(low="blue", high="yellow", guide="colorbar") +
  labs(title = "World Happiness Score")

#Correlation Matrix
numeric_data=select(data_hap,Score,GDP_per_capita,Social_support,Healthy_life_
expectancy,Freedom_to_make_life_choices,Generosity,Perceptions_of_corruption)
cormat <- round(x = cor(numeric_data), digits = 2)

melted_corr_mat <- melt(cormat)
ggplot(data = melted_corr_mat, aes(x=Var1, y=Var2,fill=value)) +
geom_tile() +
scale_fill_gradient(low = "blue", high = "yellow", guide = "colorbar")

#Lineal Model lm function
fit <- lm(Score~1+GDP_per_capita+Social_support+Healthy_life_expectancy+Freedom_
to_make_life_choices+Generosity+Perceptions_of_corruption, data = data_hap)
summary(fit)
confint(fit) #confidence intervals for the lineal model

#we apply backward selection and delete "generosity" of our model
fit1 <- lm(Score~1+GDP_per_capita+Social_support+Healthy_life_expectancy+
```

```
Freedom_to_make_life_choices+Perceptions_of_corruption, data = data_hap)
summary(fit1) #we'll work with this model
confint(fit1) #confidence intervals for the lineal model

#gdp
model <- lm(data_hap$Score~ data_hap$GDP_per_capita, data = data_hap)
cor_coef<-cor(data_hap$Score,data_hap$GDP_per_capita)
p<- ggplot(data_hap, aes(x=GDP_per_capita, y=Score)) + geom_point()+ theme_bw() +
xlab("GDP per capita") + ggtitle("GDP vs Score - Correlation: ", round(cor_coef,2))
p=p + geom_abline(intercept = coef(model)[1], slope = coef(model)[2], color="blue")
p

#social support
model <- lm(data_hap$Score~ data_hap$Social_support, data = data_hap)
cor_coef<-cor(data_hap$Score,data_hap$Social_support)
p<- ggplot(data_hap, aes(x=Social_support, y=Score)) + geom_point()+ theme_bw() +
xlab("Social support") + ggtitle("Social Support vs Score - Correlation: ",
round(cor_coef,2))
p=p + geom_abline(intercept = coef(model)[1], slope = coef(model)[2], color="blue")
p

#Healthy_life_expectancy
model <- lm(data_hap$Score~ data_hap$Healthy_life_expectancy, data = data_hap)
cor_coef<-cor(data_hap$Score,data_hap$Healthy_life_expectancy)
p<- ggplot(data_hap, aes(x=Healthy_life_expectancy, y=Score)) + geom_point()+ theme_bw()
+ xlab("Healthy life expectancy") + ggtitle("Healthy life vs Score - Correlation: ",
round(cor_coef,2))
p=p + geom_abline(intercept = coef(model)[1], slope = coef(model)[2], color="blue")
p

#Freedom_to_make_life_choices
model <- lm(data_hap$Score~ data_hap$Freedom_to_make_life_choices, data = data_hap)
cor_coef<-cor(data_hap$Score,data_hap$Freedom_to_make_life_choices)
p<- ggplot(data_hap, aes(x=Freedom_to_make_life_choices, y=Score)) + geom_point()
+ theme_bw() + xlab("Freedom to make life choices") + ggtitle("Freddom vs Score -
Correlation: ", round(cor_coef,2))
p=p + geom_abline(intercept = coef(model)[1], slope = coef(model)[2], color="blue")
p

#Generosity
model <- lm(data_hap$Score~ data_hap$Generosity, data = data_hap)
cor_coef<-cor(data_hap$Score,data_hap$Generosity)
p<- ggplot(data_hap, aes(x=Generosity, y=Score)) + geom_point()+ theme_bw() +
xlab("Generosity") + ggtitle("Generosity vs Score - Correlation: ", round(cor_coef,2))
p=p + geom_abline(intercept = coef(model)[1], slope = coef(model)[2], color="blue")
p
```

```
#Perceptions_of_corruption
model <- lm(data_hap$Score~ data_hap$Perceptions_of_corruption, data = data_hap)
cor_coef<-cor(data_hap$Score,data_hap$Perceptions_of_corruption)
p<- ggplot(data_hap, aes(x=Perceptions_of_corruption, y=Score)) + geom_point()+
theme_bw() + xlab("Perceptions of corruption") + ggtitle("Corruption vs Score
- Correlation: ", round(cor_coef,2))
p=p + geom_abline(intercept = coef(model)[1], slope = coef(model)[2], color="blue")
p

#GDP vs Health
model <- lm(data_hap$GDP_per_capita ~ data_hap$Healthy_life_expectancy, data = data_hap)
cor_coef<-cor(data_hap$GDP_per_capita,data_hap$Healthy_life_expectancy)
p<- ggplot(data_hap, aes(x=Healthy_life_expectancy, y=GDP_per_capita)) + geom_point()
+ theme_bw() + xlab("Healthy life expectancy") + ggtitle("GDP vs Healthy life -
Correlation: ", round(cor_coef,2))
p=p + geom_abline(intercept = coef(model)[1], slope = coef(model)[2], color="blue")
p

#GDP vs Social Support
model <- lm(data_hap$GDP_per_capita ~ data_hap$Social_support, data = data_hap)
cor_coef<-cor(data_hap$GDP_per_capita,data_hap$Social_support)
p<- ggplot(data_hap, aes(x=Social_support, y=GDP_per_capita)) + geom_point()+ theme_bw()
+ xlab("Social Support") + ggtitle("GDP vs Social Support - Correlation: ", round(cor_coef,2
p=p + geom_abline(intercept = coef(model)[1], slope = coef(model)[2], color="blue")
p

#Health vs Social
model <- lm(data_hap$Social_support ~ data_hap$Healthy_life_expectancy, data = data_hap)
cor_coef<-cor(data_hap$Social_support,data_hap$Healthy_life_expectancy)
p<- ggplot(data_hap, aes(x=Healthy_life_expectancy, y=Social_support)) + geom_point()
+ theme_bw() + xlab("Healthy life expectancy")+
ggtitle("Social Support vs Healthy life - Correlation: ", round(cor_coef,2))
p=p + geom_abline(intercept = coef(model)[1], slope = coef(model)[2], color="blue")
p

mean_score<-mean(data_hap$Score);
meand_GDP<-mean(data_hap$GDP_per_capita)
mean_gene = mean(data_hap$Generosity)
sd_score<-sd(data_hap$Score);
sd_score_GDP<-sd(data_hap$GDP_per_capita);
sd_generosity = sd(data_hap$Generosity)

getmode <- function(v) {uniqv <- unique(v); uniqv[which.max(tabulate(match(v, uniqv)))]}

moda <- getmode(data_hap$Score)
```

```
hist(data_hap$Score, xlab="Happiness Score", ylab="Frequency",
    main="Distribution of Happiness score")
abline(v = mean_score, col = c("blue"), lwd = 2, lty = 2:3)
abline(v = moda, col = c("purple"), lwd = 2, lty = 2:3)
legend(x = "topright", legend = c("Mean", "Mode"), fill = c("blue", "purple"))

#correlation coefficient between the variable scorre and gdp per capita
cor_coef<-cor(data_hap$Score,data_hap$GDP_per_capita)

#Generate the parametric bootstrap replicates and calculate the statistics of interest.
iter<-10000 ; #number of simulations
n<-length(data_hap$Score) #number of countries

slope_gdp<-numeric(iter);
slope_social<-numeric(iter);
slope_healthy<-numeric(iter);
slope_freedom<-numeric(iter);
slope_corruption<-numeric(iter);
slope_generosity<-numeric(iter);
interc<-numeric(iter)

corr_gdp<-numeric(iter);
corr_social<-numeric(iter);
corr_healthy<-numeric(iter);
corr_freedom<-numeric(iter);
corr_corruption<-numeric(iter);
#corr_generosity<-numeric(iter);

rsales<-numeric(n)
a <- lm(Score~1+GDP_per_capita+Social_support+Healthy_life_expectancy+
    Freedom_to_make_life_choices+Perceptions_of_corruption, data = data_hap)
b=summary(a)

for(i in 1:10000){
  error<-rnorm(n,0,b$sigma)
  rscore <-a$coefficient[1] + a$coefficient[2]*data_hap$GDP_per_capita +
    a$coefficient[3]*data_hap$Social_support +
    a$coefficient[4]*data_hap$Healthy_life_expectancy +
    a$coefficient[5]*data_hap$Freedom_to_make_life_choices +
    a$coefficient[6]*data_hap$Perceptions_of_corruption +
    error

  nl_nova<-lm(rscore~GDP_per_capita+Social_support+Healthy_life_expectancy+
    Freedom_to_make_life_choices+Perceptions_of_corruption, data = data_hap)
```

```
  interc[i]<-nl_nova$coefficient[1]
  slope_gdp[i]<-nl_nova$coefficient[2] ;
  slope_social[i]<-nl_nova$coefficient[3] ;
  slope_healthy[i]<-nl_nova$coefficient[4] ;
  slope_freedom[i]<-nl_nova$coefficient[5] ;
  slope_corruption[i]<-nl_nova$coefficient[6] ;
  #slope_generosity[i]<-nl_nova$coefficient[6] ;

  corr_gdp[i]<-cor(rscore,data_hap$GDP_per_capita)
  corr_social[i]<-cor(rscore,data_hap$Social_support)
  corr_healthy[i]<-cor(rscore,data_hap$Healthy_life_expectancy)
  corr_freedom[i]<-cor(rscore,data_hap$Freedom_to_make_life_choices)
  corr_corruption[i]<-cor(rscore,data_hap$Perceptions_of_corruption)
  #corr_generosity[i]<-cor(rscore,data_hap$Generosity)
}

#Calculate confidence intervals
#gdp: variable and correlation
IC=quantile(slope_gdp, probs = c(0.025, 0.975))
IC
hist(slope_gdp, xlab="GDP per capita", ylab="Frequency",
    main="GDP per capita Histogram")
abline(v = IC[1], col = c("red"), lwd = 2, lty = 2:3)
abline(v = IC[2], col = c("red"), lwd = 2, lty = 2:3)

moda <- getmode(slope_gdp)
meand_GDP<-mean(slope_gdp)
abline(v = meand_GDP, col = c("blue"), lwd = 2, lty = 2:3)
abline(v = moda, col = c("purple"), lwd = 2, lty = 2:3)
legend(x = "topright", legend = c("Mean", "Mode", "95% IC"),
    fill = c("blue", "purple", "red"))

IC_c=quantile(corr_gdp, probs = c(0.025, 0.975))
IC_c
hist(corr_gdp, xlab="Correlation GDP & Score", ylab="Frequency",
    main="Correlation GDP & Score Histogram")
abline(v = IC_c[1], col = c("red"), lwd = 2, lty = 2:3)
abline(v = IC_c[2], col = c("red"), lwd = 2, lty = 2:3)

moda <- getmode(corr_gdp)
meand_GDP<-mean(corr_gdp)
abline(v = meand_GDP, col = c("blue"), lwd = 2, lty = 2:3)
abline(v = moda, col = c("purple"), lwd = 2, lty = 2:3)
legend(x = "topright", legend = c("Mean", "Mode", "95% IC"),
    fill = c("blue", "purple", "red"))
```

```
#social support: variable and correlation
IC=quantile(slope_social, probs = c(0.025, 0.975))
IC
hist(slope_social, xlab="Social Support", ylab="Frequency",
    main="Social Support per capita Histogram")
abline(v = IC[1], col = c("red"), lwd = 2, lty = 2:3)
abline(v = IC[2], col = c("red"), lwd = 2, lty = 2:3)

moda <- getmode(slope_social)
meand_social<-mean(slope_social)
abline(v = meand_social, col = c("blue"), lwd = 2, lty = 2:3)
abline(v = moda, col = c("purple"), lwd = 2, lty = 2:3)
legend(x = "topright", legend = c("Mean", "Mode", "95% IC"),
    fill = c("blue", "purple", "red"))

IC_c=quantile(corr_social, probs = c(0.025, 0.975))
IC_c
hist(corr_social, xlab="Correlation Social Support & Score", ylab="Frequency",
    main="Correlation Social Support & Score Histogram")
abline(v = IC_c[1], col = c("red"), lwd = 2, lty = 2:3)
abline(v = IC_c[2], col = c("red"), lwd = 2, lty = 2:3)

moda <- getmode(corr_social)
meand_GDP<-mean(corr_social)
abline(v = meand_GDP, col = c("blue"), lwd = 2, lty = 2:3)
abline(v = moda, col = c("purple"), lwd = 2, lty = 2:3)
legend(x = "topright", legend = c("Mean", "Mode", "95% IC"),
    fill = c("blue", "purple", "red"))

#healthy support: variable and correlation
IC=quantile(slope_healthy, probs = c(0.025, 0.975))
IC
hist(slope_healthy, xlab="Healthy life expectancy", ylab="Frequency",
    main="Healthy life expectancy Histogram")
abline(v = IC[1], col = c("red"), lwd = 2, lty = 2:3)
abline(v = IC[2], col = c("red"), lwd = 2, lty = 2:3)

moda <- getmode(slope_healthy)
mean<-mean(slope_healthy)
abline(v = mean, col = c("blue"), lwd = 2, lty = 2:3)
abline(v = moda, col = c("purple"), lwd = 2, lty = 2:3)
legend(x = "topright", legend = c("Mean", "Mode", "95% IC"),
    fill = c("blue", "purple", "red"))

IC_c=quantile(corr_healthy, probs = c(0.025, 0.975))
IC_c
```

```
hist(corr_healthy, xlab="Correlation Healthy life expectancy & Score", ylab="Frequency",
    main="Correlation Healthy life & Score Histogram")
abline(v = IC_c[1], col = c("red"), lwd = 2, lty = 2:3)
abline(v = IC_c[2], col = c("red"), lwd = 2, lty = 2:3)

moda <- getmode(corr_healthy)
mean<-mean(corr_healthy)
abline(v = mean, col = c("blue"), lwd = 2, lty = 2:3)
abline(v = moda, col = c("purple"), lwd = 2, lty = 2:3)
legend(x = "topright", legend = c("Mean", "Mode", "95% IC"),
    fill = c("blue", "purple", "red"))

#freedom: variable and correlation
IC=quantile(slope_freedom, probs = c(0.025, 0.975))
IC
hist(slope_freedom, xlab="Freedom", ylab="Frequency",
    main="Freedom Histogram")
abline(v = IC[1], col = c("red"), lwd = 2, lty = 2:3)
abline(v = IC[2], col = c("red"), lwd = 2, lty = 2:3)

moda <- getmode(slope_freedom)
mean<-mean(slope_freedom)
abline(v = mean, col = c("blue"), lwd = 2, lty = 2:3)
abline(v = moda, col = c("purple"), lwd = 2, lty = 2:3)
legend(x = "topright", legend = c("Mean", "Mode", "95% IC"),
    fill = c("blue", "purple", "red"))

IC_c=quantile(corr_freedom, probs = c(0.025, 0.975))
IC_c
hist(corr_freedom, xlab="Correlation Freedom & Score", ylab="Frequency",
    main="Correlation Freedom & Score Histogram")
abline(v = IC_c[1], col = c("red"), lwd = 2, lty = 2:3)
abline(v = IC_c[2], col = c("red"), lwd = 2, lty = 2:3)

moda <- getmode(corr_freedom)
meand_GDP<-mean(corr_freedom)
abline(v = meand_GDP, col = c("blue"), lwd = 2, lty = 2:3)
abline(v = moda, col = c("purple"), lwd = 2, lty = 2:3)
legend(x = "topright", legend = c("Mean", "Mode", "95% IC"),
    fill = c("blue", "purple", "red"))

IC=quantile(slope_corruption, probs = c(0.025, 0.975))
IC
hist(slope_corruption, xlab="Perceptions of corruption", ylab="Frequency",
    main="Perceptions of corruption Histogram")
abline(v = IC[1], col = c("red"), lwd = 2, lty = 2:3)
```

```
abline(v = IC[2], col = c("red"), lwd = 2, lty = 2:3)

moda <- getmode(slope_corruption)
meand_GDP<-mean(slope_corruption)
abline(v = meand_GDP, col = c("blue"), lwd = 2, lty = 2:3)
abline(v = moda, col = c("purple"), lwd = 2, lty = 2:3)
legend(x = "topright", legend = c("Mean", "Mode", "95% IC"),
    fill = c("blue", "purple", "red"))

IC_c=quantile(corr_corruption, probs = c(0.025, 0.975))
IC_c
hist(corr_corruption, xlab="Correlation Corruption & Score", ylab="Frequency",
    main="Correlation Corruption & Score Histogram")
abline(v = IC_c[1], col = c("red"), lwd = 2, lty = 2:3)
abline(v = IC_c[2], col = c("red"), lwd = 2, lty = 2:3)

moda <- getmode(corr_corruption)
meand_GDP<-mean(corr_corruption)
abline(v = meand_GDP, col = c("blue"), lwd = 2, lty = 2:3)
abline(v = moda, col = c("purple"), lwd = 2, lty = 2:3)
legend(x = "topright", legend = c("Mean", "Mode", "95% IC"),
    fill = c("blue", "purple", "red"))

#non-parametric boostrap
n<-length(data_hap$Score) #number of countries
iter=10000

slope_gdp<-numeric(iter);
slope_social<-numeric(iter);
slope_healthy<-numeric(iter);
slope_freedom<-numeric(iter);
slope_corruption<-numeric(iter);
slope_generosity<-numeric(iter);
interc<-numeric(iter)

corr_gdp<-numeric(iter);
corr_social<-numeric(iter);
corr_healthy<-numeric(iter);
corr_freedom<-numeric(iter);
corr_corruption<-numeric(iter);

hap<- select(data_hap,Score, GDP_per_capita, Social_support,Healthy_life_expectancy,
    Freedom_to_make_life_choices,Perceptions_of_corruption  )
```

```
for(i in 1:10000){
  yb <- hap[sample(n, n, replace = TRUE ), ]
  fitb<-lm(Score~GDP_per_capita+Social_support+Healthy_life_expectancy+
    Freedom_to_make_life_choices+Perceptions_of_corruption, data = yb)

  interc[i] <- summary(fitb)$coefficients[1]
  slope_gdp[i]<-summary(fitb)$coefficients[2]
  slope_social[i]<-summary(fitb)$coefficients[3]
  slope_healthy[i]<-summary(fitb)$coefficients[4]
  slope_freedom[i]<-summary(fitb)$coefficients[5]
  slope_corruption[i]<-summary(fitb)$coefficients[6]
  #slope_generosity[i]<-summary(fitb)$coefficients[6]

  corr_gdp[i]<-cor(yb$Score , yb$GDP_per_capita)
  corr_social[i]<-cor(yb$Score , yb$Social_support)
  corr_healthy[i]<-cor(yb$Score , yb$Healthy_life_expectancy)
  corr_freedom[i]<-cor(yb$Score , yb$Freedom_to_make_life_choices)
  corr_corruption[i]<-cor(yb$Score , yb$Perceptions_of_corruption)
  #corr_generosity[i]<-cor(rscore,data_hap$Generosity)
}
#Calculate confidence intervals
#gdp: variable and correlation
IC=quantile(slope_gdp, probs = c(0.025, 0.975))
IC
hist(slope_gdp, xlab="GDP per capita", ylab="Frequency",
    main="GDP per capita Histogram")
abline(v = IC[1], col = c("red"), lwd = 2, lty = 2:3)
abline(v = IC[2], col = c("red"), lwd = 2, lty = 2:3)
moda <- getmode(slope_gdp)
meand_GDP<-mean(slope_gdp)
abline(v = meand_GDP, col = c("blue"), lwd = 2, lty = 2:3)
abline(v = moda, col = c("purple"), lwd = 2, lty = 2:3)
legend(x = "topright", legend = c("Mean", "Mode", "95% IC"),
    fill = c("blue", "purple", "red"))

IC_c=quantile(corr_gdp, probs = c(0.025, 0.975))
IC_c
hist(corr_gdp, xlab="Correlation GDP & Score", ylab="Frequency",
    main="Correlation GDP & Score Histogram")
abline(v = IC_c[1], col = c("red"), lwd = 2, lty = 2:3)
abline(v = IC_c[2], col = c("red"), lwd = 2, lty = 2:3)

moda <- getmode(corr_gdp)
meand_GDP<-mean(corr_gdp)
abline(v = meand_GDP, col = c("blue"), lwd = 2, lty = 2:3)
abline(v = moda, col = c("purple"), lwd = 2, lty = 2:3)
```

```
legend(x = "topright", legend = c("Mean", "Mode", "95% IC"),
    fill = c("blue", "purple", "red"))

#social support: variable and correlation
IC=quantile(slope_social, probs = c(0.025, 0.975))
IC
hist(slope_social, xlab="Social Support", ylab="Frequency",
    main="Social Support per capita Histogram")
abline(v = IC[1], col = c("red"), lwd = 2, lty = 2:3)
abline(v = IC[2], col = c("red"), lwd = 2, lty = 2:3)

moda <- getmode(slope_social)
meand_social<-mean(slope_social)
abline(v = meand_social, col = c("blue"), lwd = 2, lty = 2:3)
abline(v = moda, col = c("purple"), lwd = 2, lty = 2:3)
legend(x = "topright", legend = c("Mean", "Mode", "95% IC"),
    fill = c("blue", "purple", "red"))

IC_c=quantile(corr_social, probs = c(0.025, 0.975))
IC_c
hist(corr_social, xlab="Correlation Social Support & Score", ylab="Frequency",
    main="Correlation Social Support & Score Histogram")
abline(v = IC_c[1], col = c("red"), lwd = 2, lty = 2:3)
abline(v = IC_c[2], col = c("red"), lwd = 2, lty = 2:3)

moda <- getmode(corr_social)
meand_GDP<-mean(corr_social)
abline(v = meand_GDP, col = c("blue"), lwd = 2, lty = 2:3)
abline(v = moda, col = c("purple"), lwd = 2, lty = 2:3)
legend(x = "topright", legend = c("Mean", "Mode", "95% IC"),
    fill = c("blue", "purple", "red"))

#healthy support: variable and correlation
IC=quantile(slope_healthy, probs = c(0.025, 0.975))
IC
hist(slope_healthy, xlab="Healthy life expectancy", ylab="Frequency",
    main="Healthy life expectancy Histogram")
abline(v = IC[1], col = c("red"), lwd = 2, lty = 2:3)
abline(v = IC[2], col = c("red"), lwd = 2, lty = 2:3)

moda <- getmode(slope_healthy)
mean<-mean(slope_healthy)
abline(v = mean, col = c("blue"), lwd = 2, lty = 2:3)
abline(v = moda, col = c("purple"), lwd = 2, lty = 2:3)
legend(x = "topright", legend = c("Mean", "Mode", "95% IC"),
    fill = c("blue", "purple", "red"))
```

```
IC_c=quantile(corr_healthy, probs = c(0.025, 0.975))
IC_c
hist(corr_healthy, xlab="Correlation Healthy life expectancy & Score", ylab="Frequency",
    main="Correlation Healthy life & Score Histogram")
abline(v = IC_c[1], col = c("red"), lwd = 2, lty = 2:3)
abline(v = IC_c[2], col = c("red"), lwd = 2, lty = 2:3)

moda <- getmode(corr_healthy)
mean<-mean(corr_healthy)
abline(v = mean, col = c("blue"), lwd = 2, lty = 2:3)
abline(v = moda, col = c("purple"), lwd = 2, lty = 2:3)
legend(x = "topright", legend = c("Mean", "Mode", "95% IC"),
    fill = c("blue", "purple", "red"))

#freedom: variable and correlation
IC=quantile(slope_freedom, probs = c(0.025, 0.975))
IC
hist(slope_freedom, xlab="Freedom", ylab="Frequency",
    main="Freedom Histogram")
abline(v = IC[1], col = c("red"), lwd = 2, lty = 2:3)
abline(v = IC[2], col = c("red"), lwd = 2, lty = 2:3)

moda <- getmode(slope_freedom)
mean<-mean(slope_freedom)
abline(v = mean, col = c("blue"), lwd = 2, lty = 2:3)
abline(v = moda, col = c("purple"), lwd = 2, lty = 2:3)
legend(x = "topright", legend = c("Mean", "Mode", "95% IC"),
    fill = c("blue", "purple", "red"))

IC_c=quantile(corr_freedom, probs = c(0.025, 0.975))
IC_c
hist(corr_freedom, xlab="Correlation Freedom & Score", ylab="Frequency",
    main="Correlation Freedom & Score Histogram")
abline(v = IC_c[1], col = c("red"), lwd = 2, lty = 2:3)
abline(v = IC_c[2], col = c("red"), lwd = 2, lty = 2:3)

moda <- getmode(corr_freedom)
meand_GDP<-mean(corr_freedom)
abline(v = meand_GDP, col = c("blue"), lwd = 2, lty = 2:3)
abline(v = moda, col = c("purple"), lwd = 2, lty = 2:3)
legend(x = "topright", legend = c("Mean", "Mode", "95% IC"),
    fill = c("blue", "purple", "red"))

IC=quantile(slope_corruption, probs = c(0.025, 0.975))
IC
```

```
hist(slope_corruption, xlab="Perceptions of corruption", ylab="Frequency",
    main="Perceptions of corruption Histogram")
abline(v = IC[1], col = c("red"), lwd = 2, lty = 2:3)
abline(v = IC[2], col = c("red"), lwd = 2, lty = 2:3)

moda <- getmode(slope_corruption)
meand_GDP<-mean(slope_corruption)
abline(v = meand_GDP, col = c("blue"), lwd = 2, lty = 2:3)
abline(v = moda, col = c("purple"), lwd = 2, lty = 2:3)
legend(x = "topright", legend = c("Mean", "Mode", "95% IC"),
    fill = c("blue", "purple", "red"))

IC_c=quantile(corr_corruption, probs = c(0.025, 0.975))
IC_c
hist(corr_corruption, xlab="Correlation Corruption & Score", ylab="Frequency",
    main="Correlation Corruption & Score Histogram")
abline(v = IC_c[1], col = c("red"), lwd = 2, lty = 2:3)
abline(v = IC_c[2], col = c("red"), lwd = 2, lty = 2:3)

moda <- getmode(corr_corruption)
meand_GDP<-mean(corr_corruption)
abline(v = meand_GDP, col = c("blue"), lwd = 2, lty = 2:3)
abline(v = moda, col = c("purple"), lwd = 2, lty = 2:3)
legend(x = "topright", legend = c("Mean", "Mode", "95% IC"),
    fill = c("blue", "purple", "red"))
```