Práctica 2. Tipología y ciclo de vida de los datos

Alicia Julián Beltrán

5 de enero de 2019

Ejercicio 1

Descripción del dataset

El conjunto de datos seleccionado es el dataset World Happiness Report (https://www.kaggle.com/unsdsn/world-happiness (https://www.kaggle.com/unsdsn/world-happiness)).
El dataset mide la felicidad basándose en seis factores: producción económica o PIB, apoyo social o familia, esperanza de vida, libertad, confianza en el gobierno o ausencia de corrupción y generosidad.
Los campos que tiene el dataset son:
- **Country**. Nombre del país.
- **Happiness Rank**. Rango del país en función de la puntuación en felicidad.
- **Happiness Score**. Métrica obtenida preguntando a las personas de la muestra cómo calificarían su felicidad, basándose en una escala del 1 al 10, donde el 10 es la puntuación más alta, es decir, el mayor nivel de felicidad.
- **Standard Error**. Error estándar de la puntuación de la felicidad.
- **Economy (GDP per Capita)**. La medida en que el PIB contribuye al cálculo de la puntuación de la felicidad.
- **Family**. La medida en que la familia contribuye al cálculo de la puntuación de la felicidad.
- **Health (Life Expectancy)**. La medida en que la esperanza de vida contribuye al cálculo de la puntuación de la felicidad.
- **Freedom**. La medida en que la libertad contribuye al cálculo de la puntuación de la felicidad.
- **Trust (Government Corruption)**. La medida en que la confianza en el gobierno contribuye al cálculo de la puntuación de la felicidad.
- **Generosity**. La medida en que la generosidad contribuye al cálculo de la puntuación de la felicidad.
- **Dystopia Residual**. La medida en que distopía residual contribuye al cálculo de la puntuación de la felicidad.

El dataset es importante para identificar aquellos factores que hacen que la población sea más feliz.

```
> ##Establecemos el directorio de trabajo
> setwd("C:/Users/ali00/Google Drive/UOC_2018_2019/Tipologia_CV_Datos/PRACT2/")
> ##Verificamos que el directorio de trabajo se ha cambiado correctamente
> getwd()
```

```
[1] "C:/Users/ali00/Google Drive/UOC_2018_2019/Tipologia_CV_Datos/PRACT2"
```

Ejercicio 2

Integración de los datos

En primer lugar, vamos a importar e integrar los 3 ficheros.

```
> ##Importamos el fichero 2015.csv en el dataset Happiness2015
> happiness2015=read.csv2("2015.csv",header=TRUE,sep=",")
> #Añadimos una columna para el año de los datos
> happiness2015$Year=2015
>
> ##Importamos el fichero 2016.csv en el dataset Happiness2016
> happiness2016=read.csv2("2016.csv",header=TRUE,sep=",")
> #Añadimos una columna para el año de los datos
> happiness2016$Year=2016
>
> ##Importamos el fichero 2017.csv en el dataset Happiness2015
> happiness2017=read.csv2("2017.csv",header=TRUE,sep=",")
> #Añadimos una columna para el año de los datos
> happiness2017$Year=2017
```

Una vez importados, vamos a ver la estructura de cada uno de ellos con el objetivo de integrarlos.

```
> #Describimos el dataset del año 2015
> ##Número de campos
> ncol(happiness2015)
```

```
[1] 13
```

```
> #Estructura
> str(happiness2015)
```

```
'data.frame':   158 obs. of  13 variables:
 $ Country                      : Factor w/ 158 levels "Afghanistan",..: 136 59 38
106 25 46 100 135 101 7 ...
 $ Region                       : Factor w/ 10 levels "Australia and New Zealan
d",..: 10 10 10 10 6 10 10 10 1 1 ...
 $ Happiness.Rank               : int  1 2 3 4 5 6 7 8 9 10 ...
 $ Happiness.Score              : Factor w/ 157 levels "2.839","2.905",..: 157 156
155 154 153 152 151 150 149 148 ...
 $ Standard.Error               : Factor w/ 153 levels "0.01848","0.01866",..: 20 1
01 17 51 27 11 6 12 19 57 ...
 $ Economy..GDP.per.Capita.     : Factor w/ 158 levels "0","0.0153","0.01604",..: 1
52 137 140 154 141 135 143 144 130 145 ...
 $ Family                       : Factor w/ 158 levels "0","0.13995",..: 155 158 15
6 153 152 150 139 142 151 148 ...
 $ Health..Life.Expectancy.     : Factor w/ 157 levels "0","0.04776",..: 149 151 12
9 133 141 135 138 144 142 148 ...
 $ Freedom                      : Factor w/ 158 levels "0","0.07699",..: 157 144 15
2 158 146 151 139 155 148 153 ...
 $ Trust..Government.Corruption.: Factor w/ 157 levels "0","0.00227",..: 151 99 15
4 144 142 150 139 153 152 143 ...
 $ Generosity                   : Factor w/ 158 levels "0","0.00199",..: 116 145 13
0 133 146 94 151 134 150 144 ...
 $ Dystopia.Residual            : Factor w/ 158 levels "0.32858","0.65429",..: 126
135 123 119 117 131 120 110 99 100 ...
 $ Year                         : num  2015 2015 2015 2015 2015 ...
```

```
> #Describimos el dataset del año 2016
> ##Número de campos
> ncol(happiness2016)
```

```
[1] 14
```

```
> #Estructura
> str(happiness2016)
```

```
'data.frame':   157 obs. of  14 variables:
 $ Country                      : Factor w/ 157 levels "Afghanistan",..: 38 135 58
104 45 26 98 99 7 134 ...
 $ Region                       : Factor w/ 10 levels "Australia and New Zealan
d",..: 10 10 10 10 10 6 10 1 1 10 ...
 $ Happiness.Rank               : int  1 2 3 4 5 6 7 8 9 10 ...
 $ Happiness.Score              : Factor w/ 154 levels "2.905","3.069",..: 154 153
152 151 150 149 148 147 146 145 ...
 $ Lower.Confidence.Interval    : Factor w/ 154 levels "2.732","2.936",..: 154 153
149 152 151 150 148 147 146 145 ...
 $ Upper.Confidence.Interval    : Factor w/ 154 levels "3.078","3.202",..: 153 152
154 151 150 149 147 148 146 145 ...
 $ Economy..GDP.per.Capita.     : Factor w/ 157 levels "0","0.05661",..: 141 151 13
8 153 136 139 146 131 142 145 ...
 $ Family                       : Factor w/ 157 levels "0","0.10419",..: 154 152 15
7 149 151 144 120 156 147 142 ...
 $ Health..Life.Expectancy.     : Factor w/ 156 levels "0","0.03824",..: 127 150 15
1 129 134 140 135 141 149 142 ...
 $ Freedom                      : Factor w/ 157 levels "0","0.00589",..: 151 154 14
5 156 149 150 141 152 148 153 ...
 $ Trust..Government.Corruption.: Factor w/ 156 levels "0","0.00322",..: 153 151 10
7 147 150 141 138 152 144 149 ...
 $ Generosity                   : Factor w/ 157 levels "0","0.02025",..: 129 110 14
7 131 98 142 146 151 145 133 ...
 $ Dystopia.Residual            : Factor w/ 157 levels "0.81789","0.91681",..: 127
122 134 118 132 123 124 98 106 107 ...
 $ Year                         : num  2016 2016 2016 2016 2016 ...
```

```
> #Describimos el dataset del año 2017
> ##Número de campos
> ncol(happiness2017)
```

```
[1] 13
```

```
> #Estructura
> str(happiness2017)
```

```
'data.frame':   155 obs. of  13 variables:
 $ Country                      : Factor w/ 155 levels "Afghanistan",..: 105 38 58
133 45 99 26 100 132 7 ...
 $ Happiness.Rank               : int  1 2 3 4 5 6 7 8 9 10 ...
 $ Happiness.Score              : Factor w/ 151 levels "2.69300007820129",..: 151 1
50 149 148 147 146 145 144 143 143 ...
 $ Whisker.high                 : Factor w/ 155 levels "2.86488426923752",..: 154 1
53 155 152 151 150 149 148 146 147 ...
 $ Whisker.low                  : Factor w/ 155 levels "2.52111588716507",..: 155 1
54 151 153 152 150 148 149 147 146 ...
 $ Economy..GDP.per.Capita.     : Factor w/ 155 levels "0","0.022643184289336
2",..: 150 138 137 149 134 144 136 129 143 139 ...
 $ Family                       : Factor w/ 155 levels "0","0.396102607250214",..:
149 153 155 147 150 120 139 151 138 146 ...
 $ Health..Life.Expectancy.     : Factor w/ 155 levels "0","0.0055647538974881
2",..: 128 125 141 150 131 134 142 136 140 145 ...
 $ Freedom                      : Factor w/ 155 levels "0","0.014995855279266
8",..: 154 151 152 150 149 139 146 148 147 142 ...
 $ Generosity                   : Factor w/ 155 levels "0","0.010164656676352",..:
130 127 145 107 85 143 141 151 132 146 ...
 $ Trust..Government.Corruption.: Factor w/ 155 levels "0","0.0043879006989300
3",..: 145 152 117 148 149 140 141 150 151 144 ...
 $ Dystopia.Residual            : Factor w/ 155 levels "0.3779137134552",..: 127 13
1 132 126 136 129 120 104 112 106 ...
 $ Year                         : num  2017 2017 2017 2017 2017 ...
```

```
> #Nos quedamos con las columnas que tienen en común los 3 datasets
> columnas=c("Year","Country", "Happiness.Rank", "Happiness.Score","Economy..GDP.pe
r.Capita.", "Health..Life.Expectancy.", "Family", "Freedom", "Trust..Government.Cor
ruption.", "Generosity", "Dystopia.Residual")
>
> #Integramos los tres datasets en uno único, quedándonos con las columnas comunes
a los tres
> happiness=rbind(subset(happiness2015, select = columnas),subset(happiness2016, se
lect = columnas),subset(happiness2017, select = columnas))
>
> ##Mostramos los campos que tiene el dataset
> colnames(happiness)
```

```
 [1] "Year"                         "Country"
 [3] "Happiness.Rank"               "Happiness.Score"
 [5] "Economy..GDP.per.Capita."     "Health..Life.Expectancy."
 [7] "Family"                       "Freedom"
 [9] "Trust..Government.Corruption." "Generosity"
[11] "Dystopia.Residual"
```

```
> ##Número de campos
> ncol(happiness)
```

```
[1] 11
```

```
> #Número de observaciones
> nrow(happiness)
```

```
[1] 470
```

Ejercicio 3

Limpieza de los datos En primer lugar, vamos a verificar si el tipo de cada una de las variables es correcto. Todas las variables, excepto Year, Country y Happiness.Rank, deberán ser numéricas.

```
> #Comprobamos el tipo de cada variable
> tipo <- sapply(happiness,class)
> kable(data.frame(variable=names(tipo),tipo=as.vector(tipo)))
```

variable	tipo
Year	numeric
Country	factor
Happiness.Rank	integer
Happiness.Score	factor
Economy..GDP.per.Capita.	factor
Health..Life.Expectancy.	factor
Family	factor
Freedom	factor
Trust..Government.Corruption.	factor
Generosity	factor
Dystopia.Residual	factor

```
> #Asignamos el tipo correcto en aquellas que no coincide
> happiness[1]=lapply(happiness[1],as.factor)
> happiness[3]=lapply(happiness[3],as.factor)
>
> for (i in 4:11){
+   happiness[,names(tipo[i])]=as.numeric(format(happiness[,names(tipo[i])], decima
l.mark="."))
+ }
>
> #Verificamos la correcta asignación
> tipo <- sapply(happiness,class)
> kable(data.frame(variable=names(tipo),tipo=as.vector(tipo)))
```

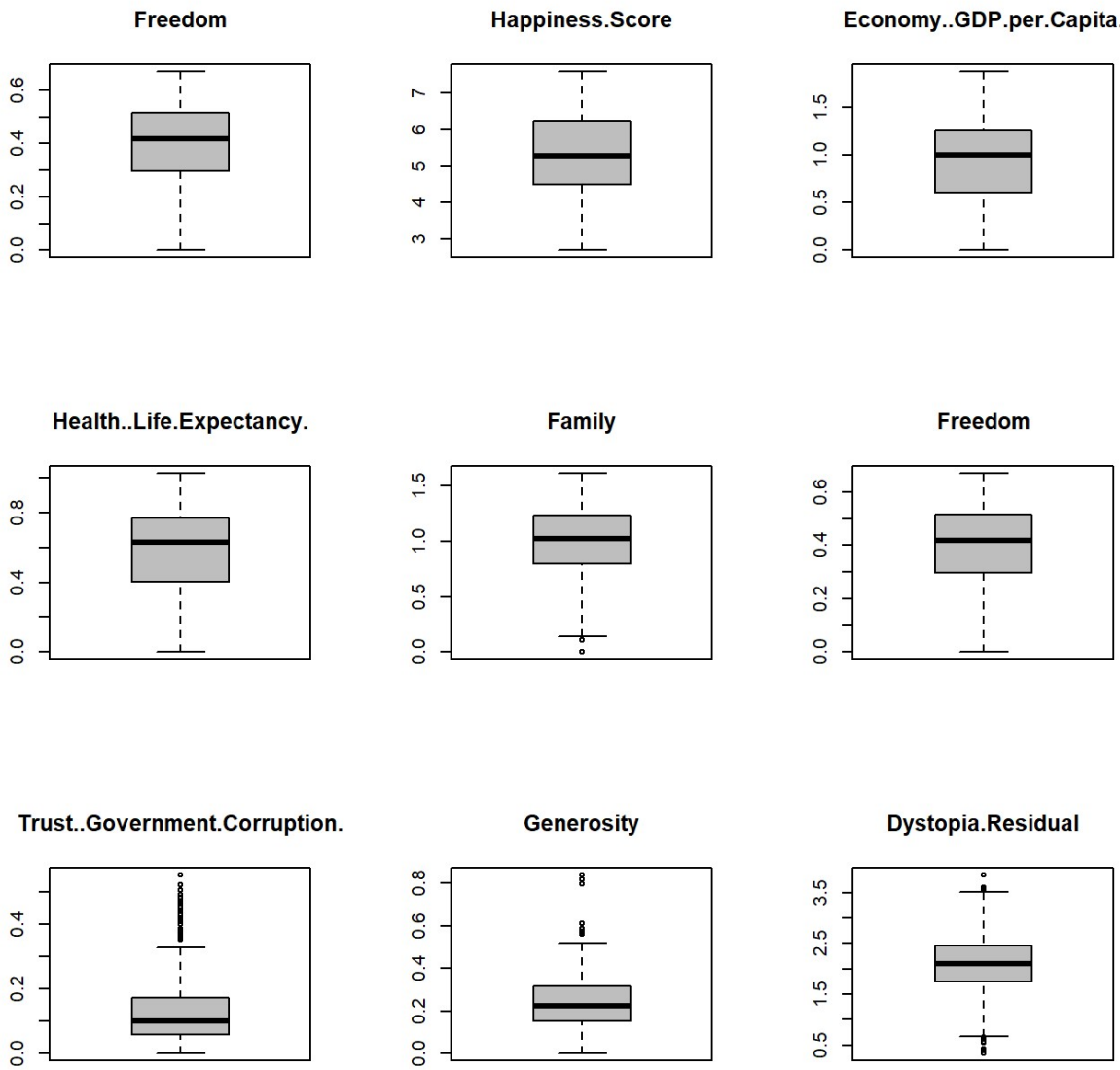
variable	tipo
Year	factor
Country	factor
Happiness.Rank	factor
Happiness.Score	numeric
Economy..GDP.per.Capita.	numeric
Health..Life.Expectancy.	numeric
Family	numeric
Freedom	numeric
Trust..Government.Corruption.	numeric
Generosity	numeric
Dystopia.Residual	numeric

Vamos a verificar si tenemos elementos vacíos o valores atípicos.

```
> ##Elementos vacíos
> sapply(happiness, function(x) sum(is.na(x)))
```

```
                          Year                        Country
                             0                              0
               Happiness.Rank               Happiness.Score
                             0                              0
     Economy..GDP.per.Capita.      Health..Life.Expectancy.
                             0                              0
                        Family                        Freedom
                             0                              0
Trust..Government.Corruption.                     Generosity
                             0                              0
            Dystopia.Residual
                             0
```

```
> # Activa el parámetro para dibujar nueve gráficos por página
> par(mfrow=c(3,3))
> # Como no existen elementos vacíos, vamos a verificar los valores atípicos para c
ada una de las variables cuantitativas
> boxplot(happiness$Freedom, main="Freedom",col="gray")
> for (i in 4:11){
+   boxplot(happiness[names(tipo[i])], main=names(happiness[names(tipo[i])]),col="g
ray")
+ }
```



La primera variable en la que observamos valores atípicos es Family. Los valores atípicos para la variable son los siguientes:

```
> indices = which(happiness$Family %in%
+   boxplot.stats(happiness[,"Family"])$out)
>   cat("Family:", toString(indices), "\n" )
```

```
Family: 148, 311, 312, 313, 470
```

```
> #Comprobamos los valores correspondientes
> happiness[indices,"Family"]
```

```
[1] 0.00000 0.10419 0.11037 0.00000 0.00000
```

Vemos que tenemos valores a 0 en las posiciones 148, 313 y 470. Para esas posiciones, vamos a imputar los valores a partir de los k-vecinos más cercanos usando la distancia de Gower con la información de todas las variables.

```
> library(VIM)
> perdidos = c(148,313,470)
> perdidos
```

```
[1] 148 313 470
```

```
> happiness$Family[perdidos] = kNN(happiness)$Family
>
> #Volvemos a revisar los valores atípicos para verificar que ya no tenemos valore
s perdidos
> indices = which(happiness$Family %in%
+   boxplot.stats(happiness[,"Family"])$out)
>   cat("Family:", toString(indices), "\n" )
```

```
Family: 158, 311, 312
```

```
> #Comprobamos los valores correspondientes
> happiness[indices,"Family"]
```

```
[1] 0.13995 0.10419 0.11037
```

Otras de las variables en la que podemos ver valores atípicos son Trust..Government.Corruption y Generosity. Los valores atípicos para las variables son los siguientes:

```
> #Variable Trust..Government.Corruption
> indicesT = which(happiness$Trust..Government.Corruption. %in%
+   boxplot.stats(happiness[,"Trust..Government.Corruption."])$out)
>   cat("Trust..Government.Corruption.:", toString(indicesT), "\n" )
```

```
Trust..Government.Corruption.: 1, 3, 4, 6, 8, 9, 10, 17, 20, 24, 28, 72, 91, 130, 1
54, 159, 160, 162, 163, 166, 168, 178, 180, 186, 194, 255, 310, 317, 319, 320, 32
3, 324, 341, 350, 466
```

```
> #Comprobamos los valores correspondientes
> happiness[indicesT,"Trust..Government.Corruption."]
```

```
 [1] 0.4197800 0.4835700 0.3650300 0.4137200 0.4384400 0.4292200 0.3563700
 [8] 0.3779800 0.3858300 0.4921000 0.5220800 0.3712400 0.3992800 0.3833100
[15] 0.5519100 0.4445300 0.4120300 0.3577600 0.4100400 0.4190400 0.4086700
[22] 0.3532900 0.4698700 0.3556100 0.4804900 0.3679400 0.5052100 0.4007701
[29] 0.3670073 0.3826115 0.3828167 0.3843987 0.4643078 0.4392993 0.4552200
```

```
> #Variable Generosity
> indicesG = which(happiness$Generosity %in%
+   boxplot.stats(happiness[,"Generosity"])$out)
>   cat("Generosity:", toString(indicesG), "\n" )
```

```
Generosity: 34, 129, 188, 191, 237, 277, 342, 347, 396, 429
```

```
> #Comprobamos los valores correspondientes
> happiness[indicesG,"Generosity"]
```

```
 [1] 0.5763000 0.7958800 0.5623700 0.5869600 0.5652100 0.8197100 0.5747306
 [8] 0.5721231 0.6117046 0.8380752
```

En ambos casos, no podemos saber si son valores correctos o incorrectos, por lo que los mantenemos.

Una vez tratados los datos, los exportamos a un fichero csv:

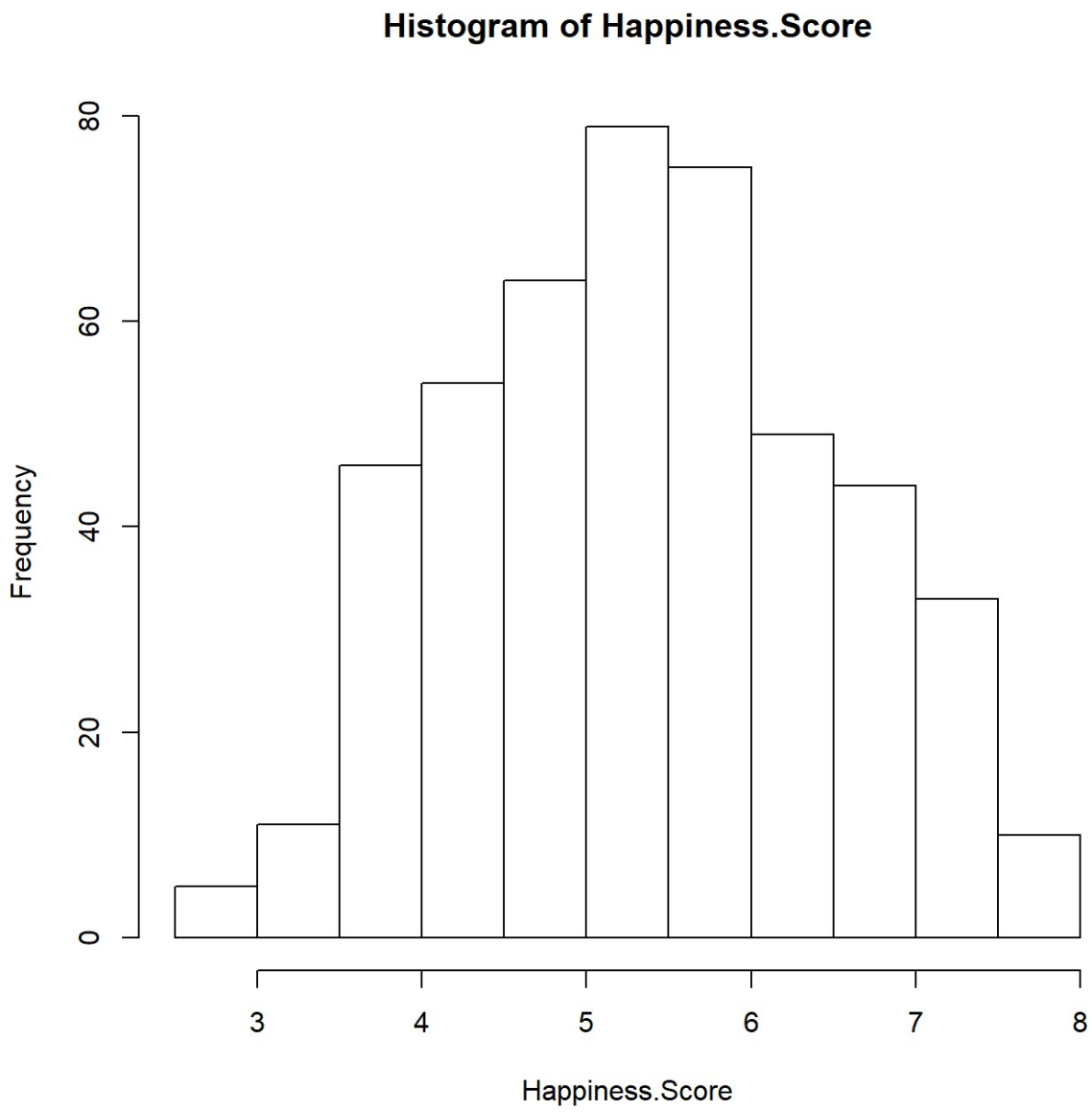
```
> write.csv(happiness, "Happiness_clean.csv")
```

Ejercicio 4

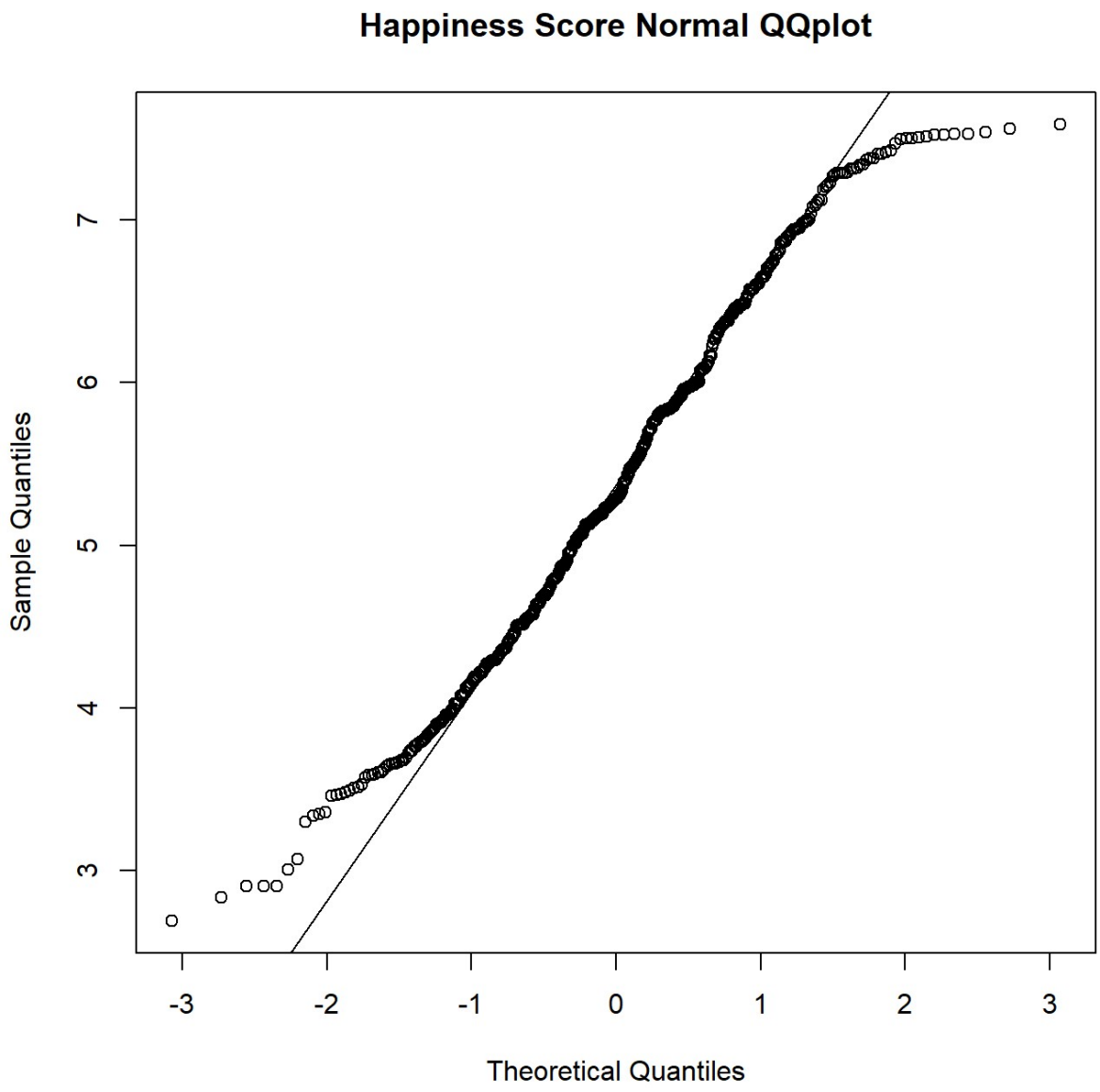
Análisis de los datos

Para hacer el estudio, nos vamos a quedar con las seis variables que miden la felicidad: Economy..GDP.per.Capita., Family, Health..Life.Expectancy., Freedom, Generosity y Trust..Government.Corruption. Para comprobar la normalidad, vamos a dibujar un histograma y un qq plot para ver de forma gráfica la simetría de la distribución para cada una de las seis variables.

```
> # Dibuja un histograma
> with(happiness, hist(Happiness.Score) )
```




```
>  # Dibuja un qq plot
> with(happiness, qqnorm(happiness$Happiness.Score, main="Happiness Score Normal QQ
plot" ));with(happiness, qqline(happiness$Happiness.Score) )
```

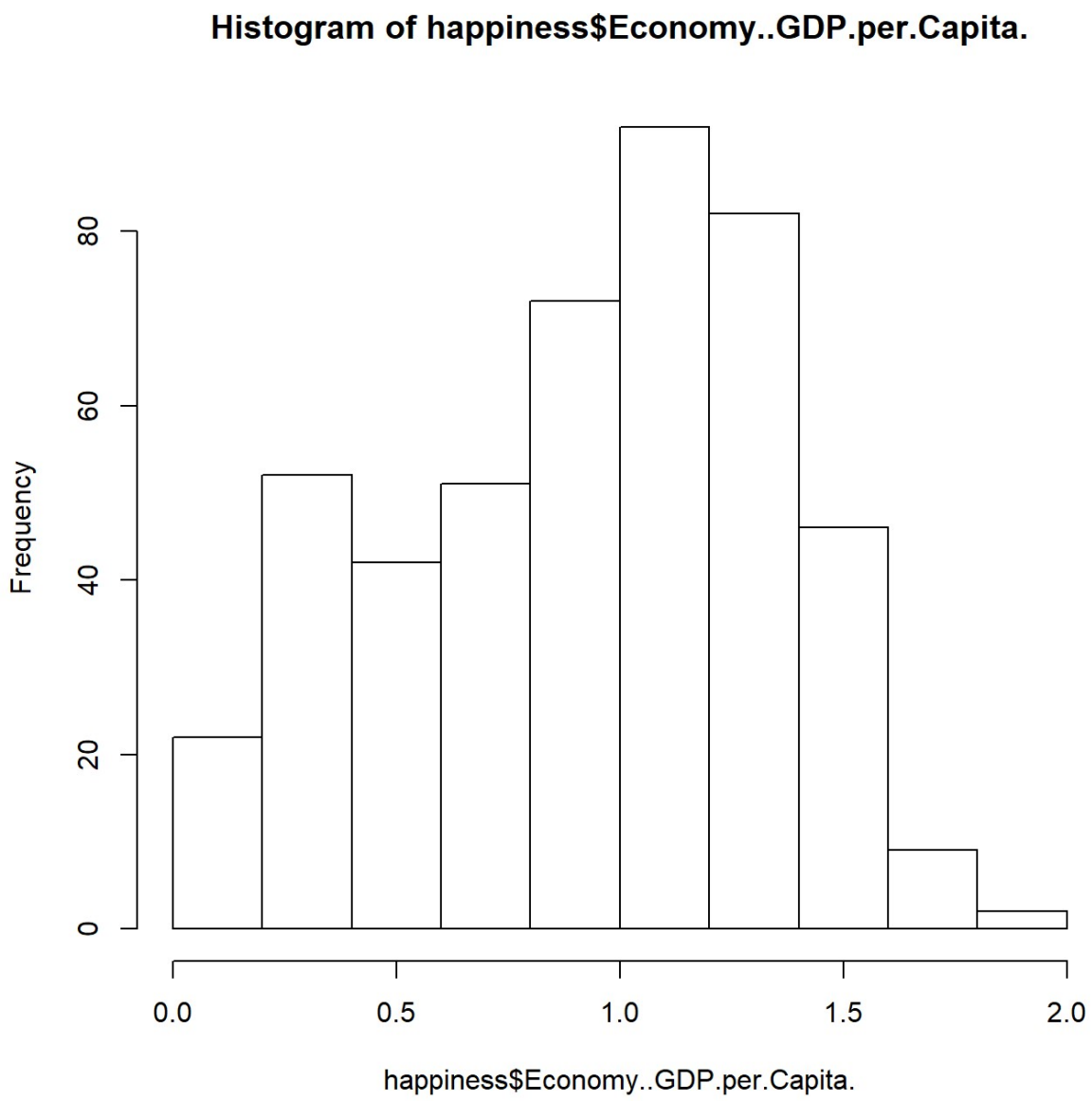



```
> # Efectúa el test de normalidad de Shapiro-Wilks
> with(happiness, shapiro.test(Happiness.Score) )
```

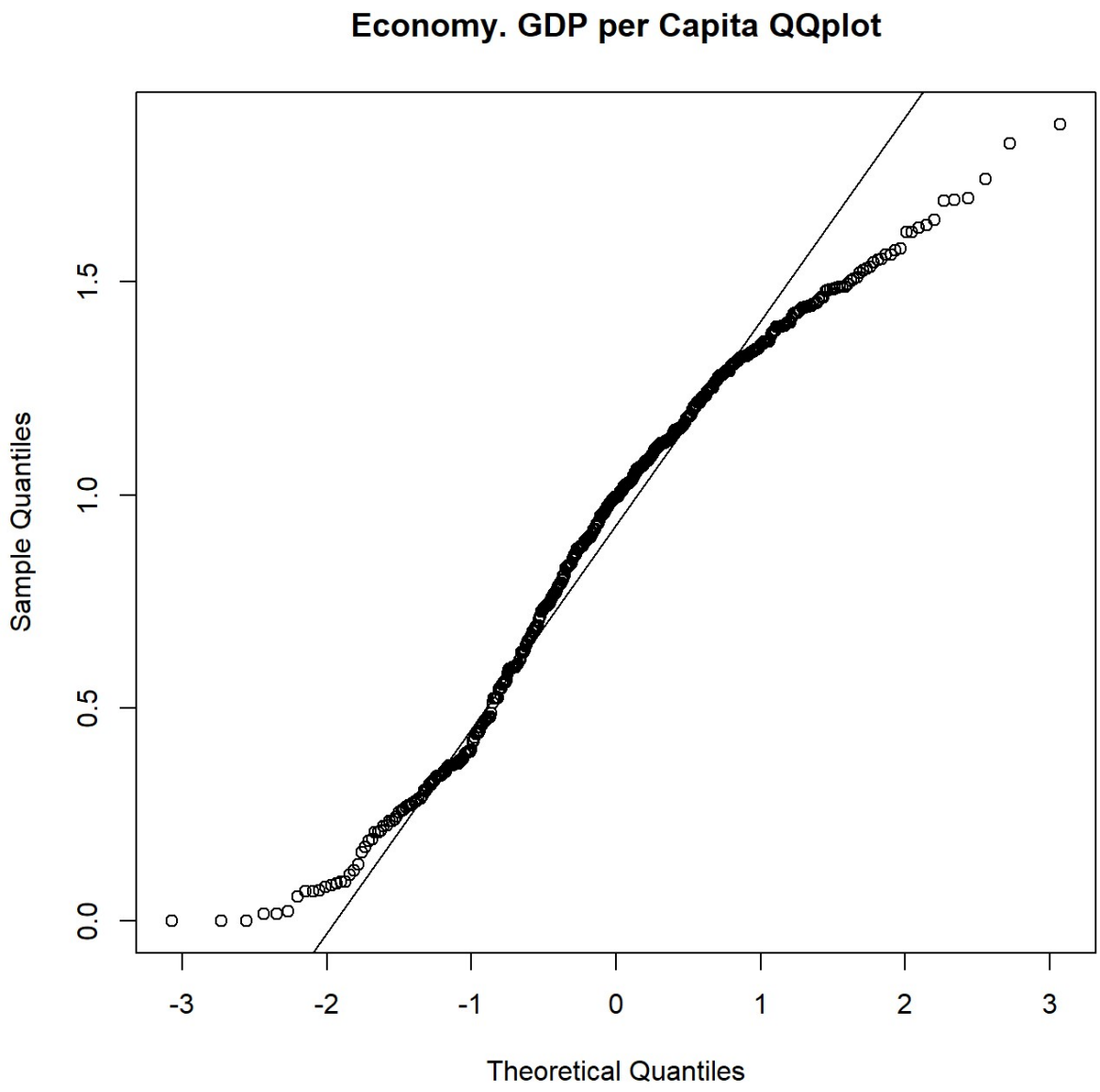
```
    Shapiro-Wilk normality test

data:  Happiness.Score
W = 0.98179, p-value = 1.269e-05
```

Se observa que la variable Happiness Score no sigue una distribución normal. El histograma no es simétrico y en el qq-plot los puntos no se sitúan sobre la línea. El p valor del contraste de normalidad nos indica que hay que rechazar la hipótesis nula y que la media no es un buen descriptor para la puntuación de la felicidad.

```
> # Dibuja un histograma
> with(happiness, hist(happiness$Economy..GDP.per.Capita.) )
```




```
>  # Dibuja un qq plot
> with(happiness, qqnorm(happiness$Economy..GDP.per.Capita., main="Economy. GDP pe
r Capita QQplot" ));with(happiness, qqline(happiness$Economy..GDP.per.Capita.) )
```



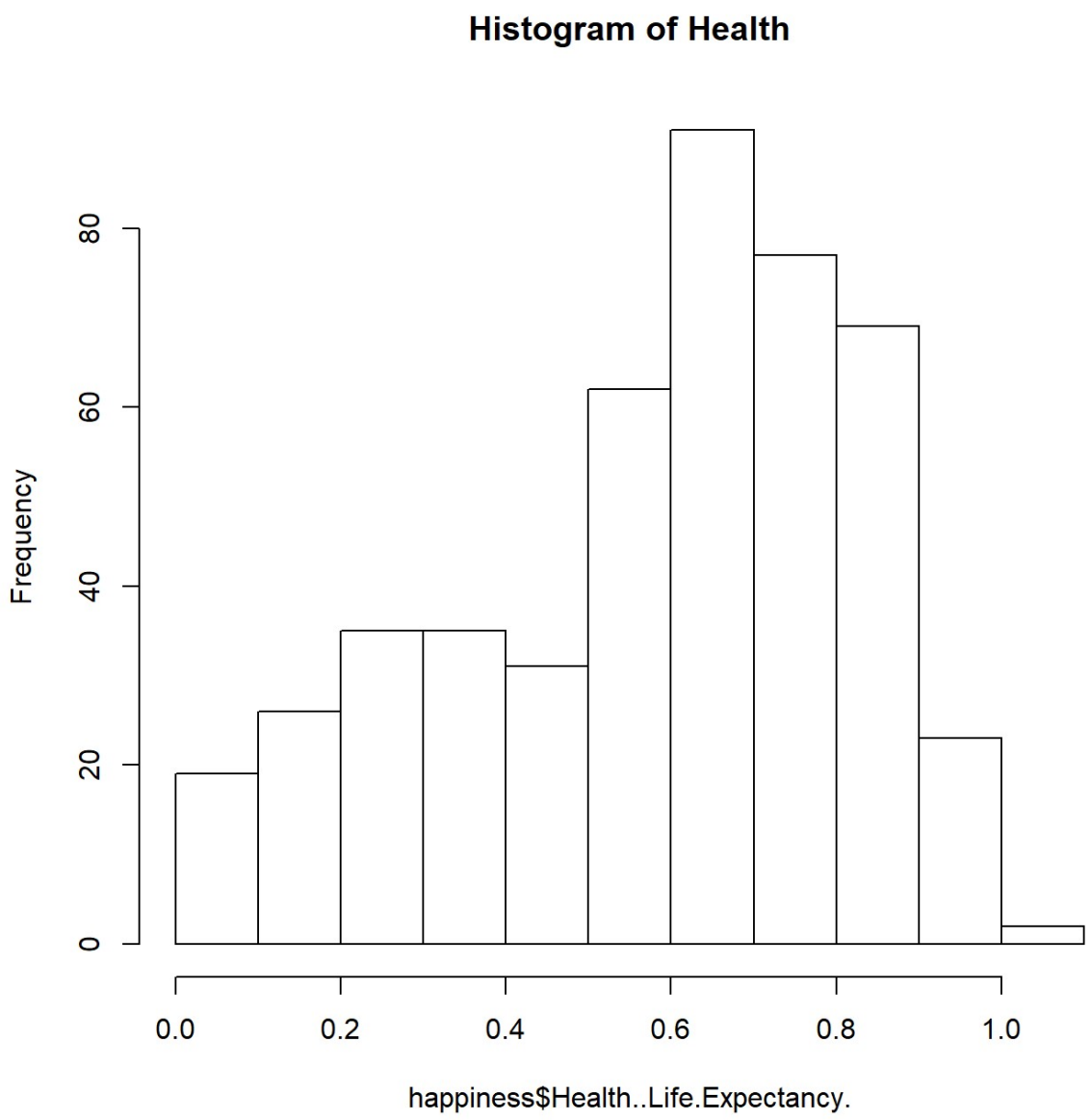

```
> # Efectúa el test de normalidad de Shapiro-Wilks
> with(happiness, shapiro.test(happiness$Economy..GDP.per.Capita.) )
```

```
    Shapiro-Wilk normality test

data:  happiness$Economy..GDP.per.Capita.
W = 0.97131, p-value = 5.752e-08
```

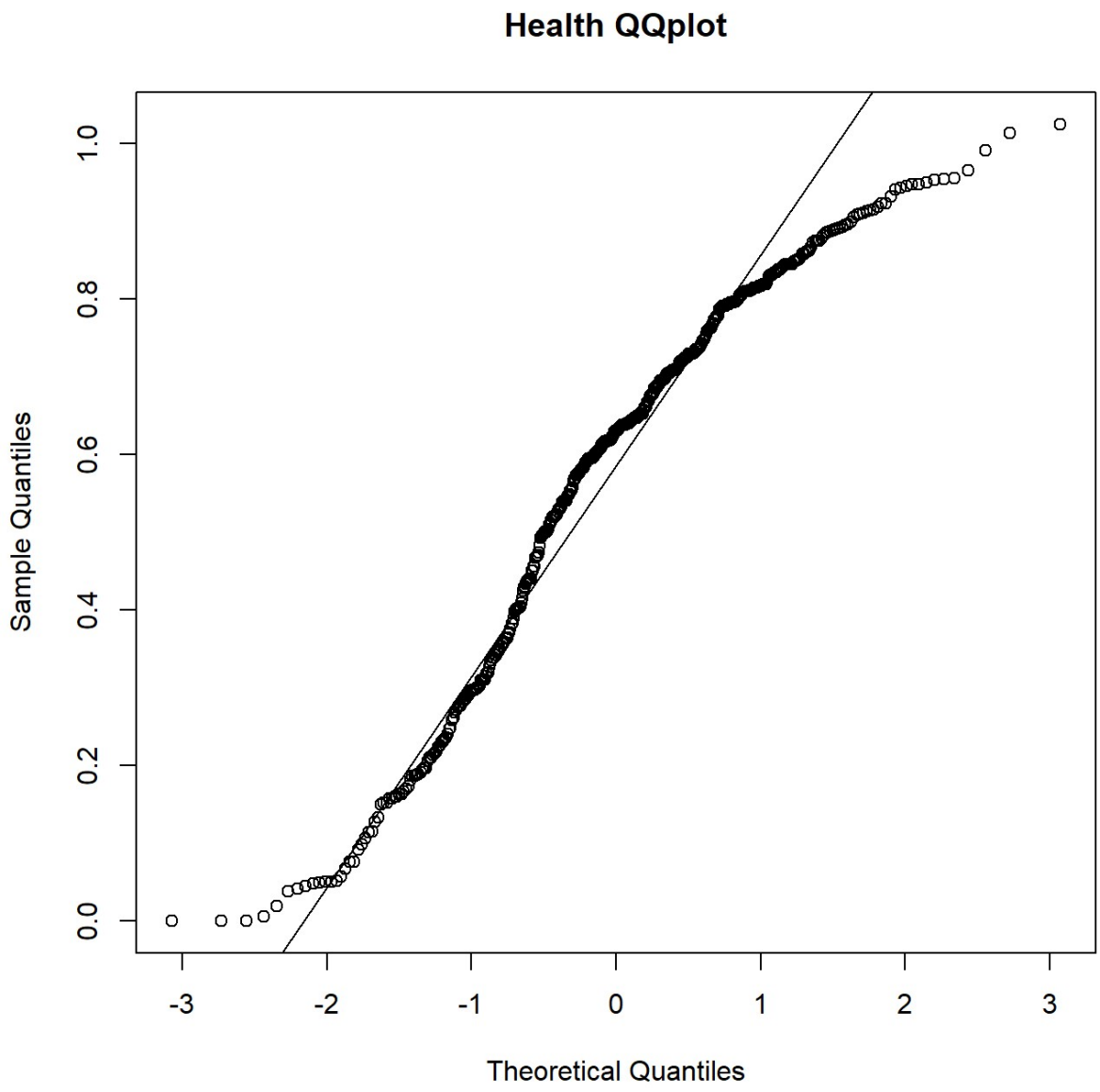
En este caso, se ve más claramente que la variable Economy..GDP.per.Capita. no sigue una distribución normal. El histograma no es simétrico y en el qq-plot los puntos no se sitúan sobre la línea. El p valor del contraste de normalidad nos indica que hay que rechazar la hipótesis nula.

A continuación, se comprueba de la misma manera que el resto de variables no siguen una distribución normal.

```
> # Dibuja un histograma
> with(happiness, hist(happiness$Health..Life.Expectancy., main = "Histogram of Hea
lth") )
```



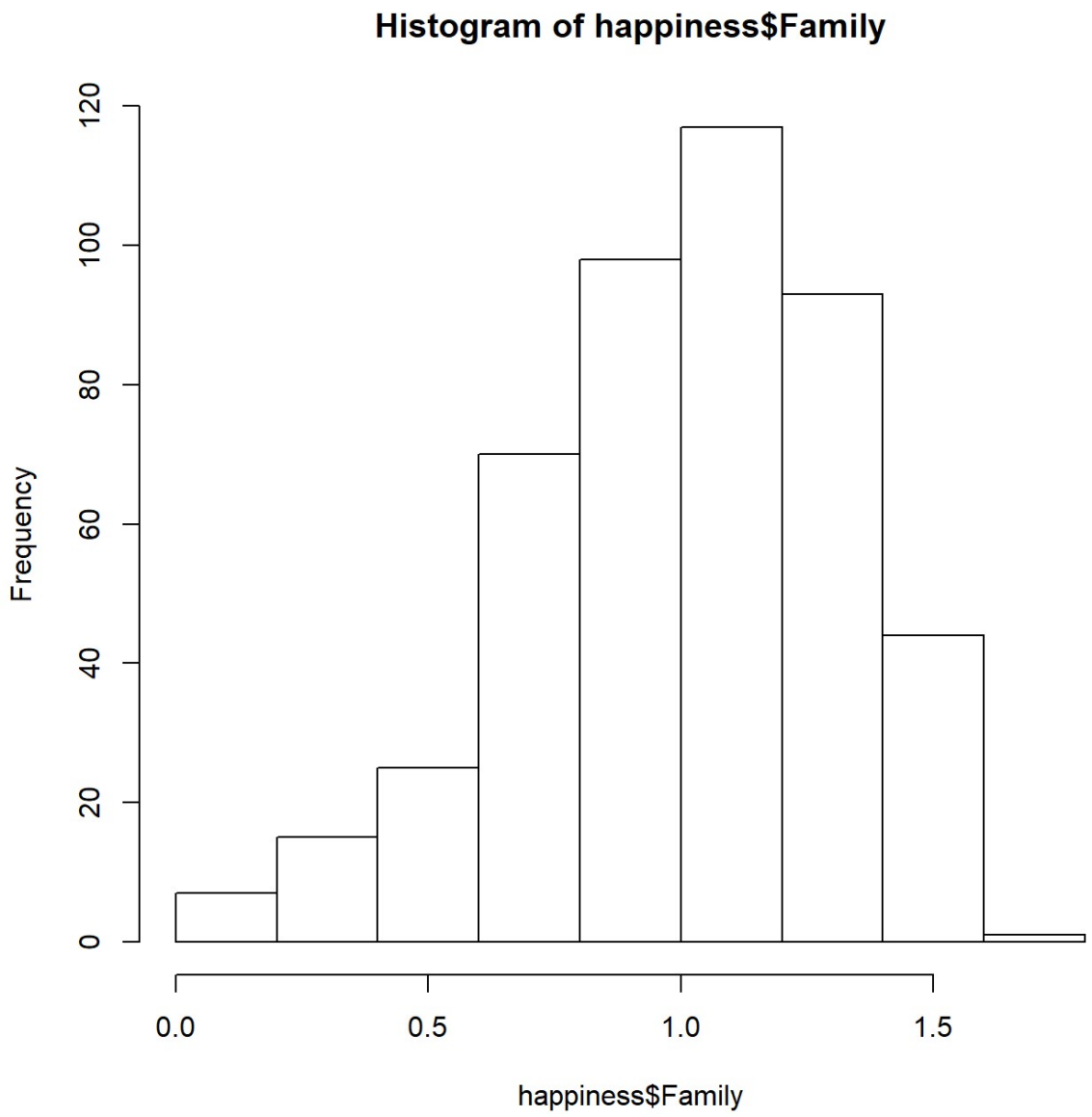

```
>  # Dibuja un qq plot
> with(happiness, qqnorm(happiness$Health..Life.Expectancy., main="Health QQplot"
));with(happiness, qqline(happiness$Health..Life.Expectancy.) )
```



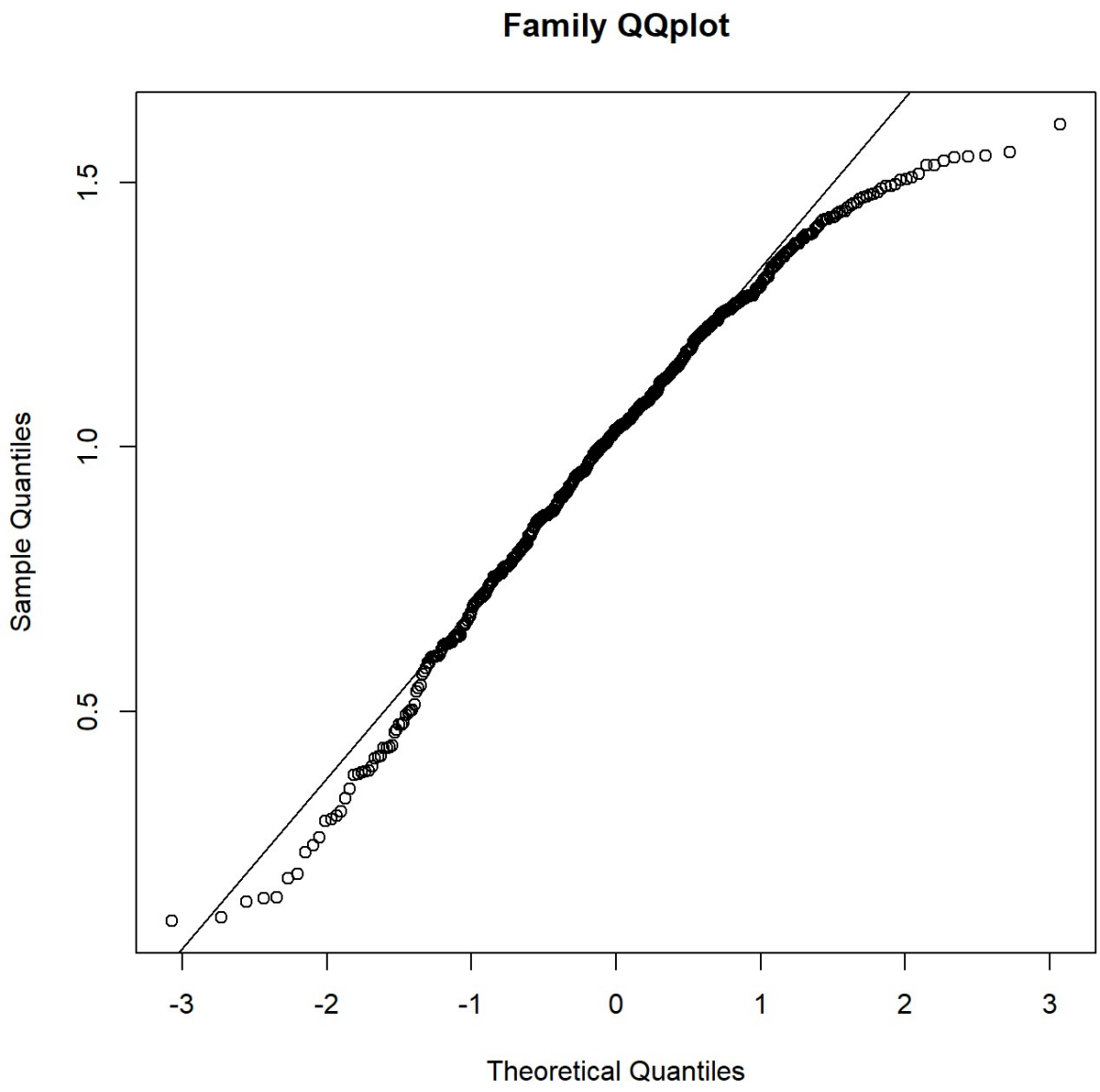

```
> # Efectúa el test de normalidad de Shapiro-Wilks
> with(happiness, shapiro.test(happiness$Health..Life.Expectancy.) )
```

```

    Shapiro-Wilk normality test

data:  happiness$Health..Life.Expectancy.
W = 0.95306, p-value = 4.554e-11
```

```
> # Dibuja un histograma
> with(happiness, hist(happiness$Family) )
```



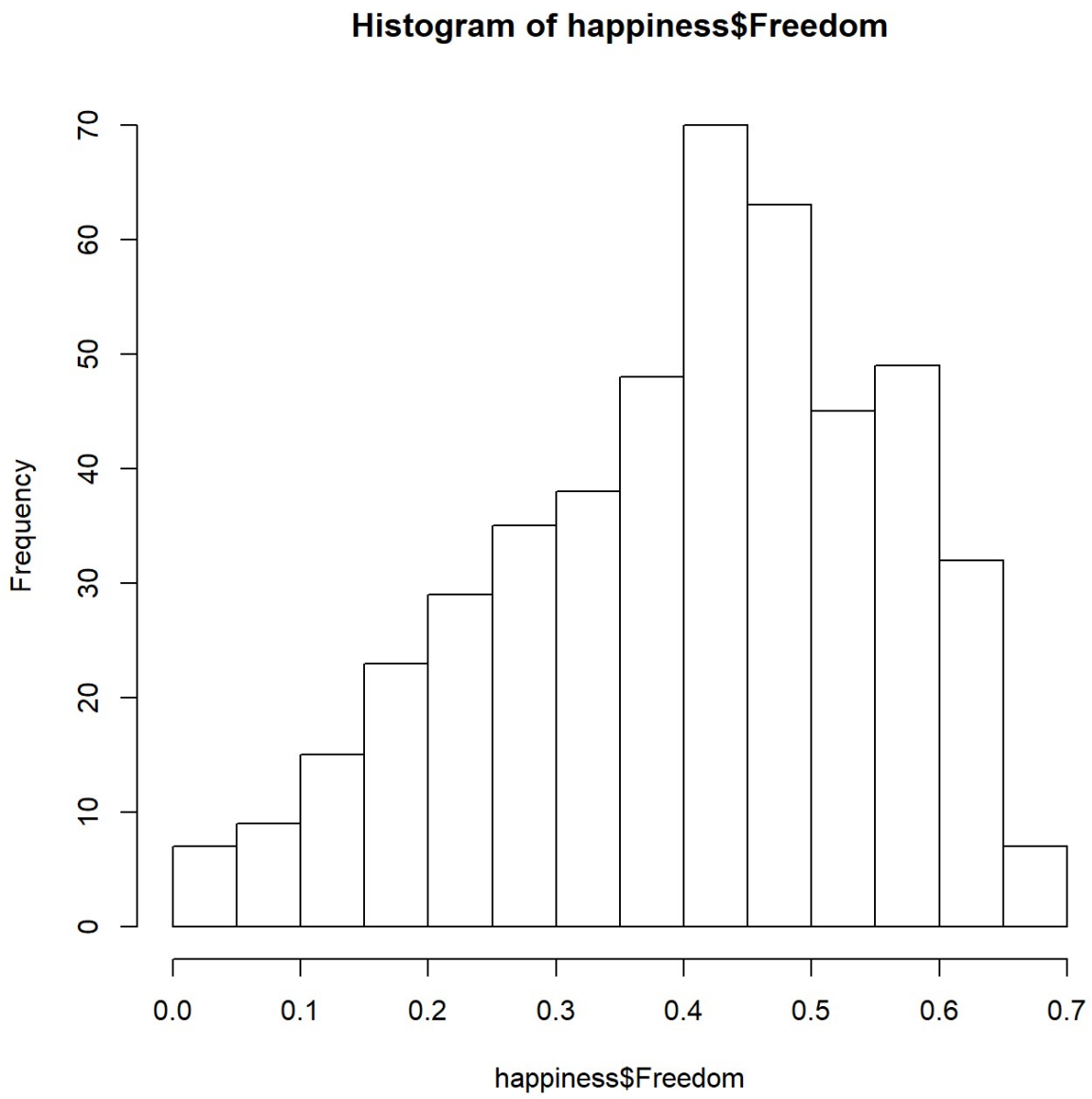

```
>  # Dibuja un qq plot
> with(happiness, qqnorm(happiness$Family, main="Family QQplot" ));with(happiness,
qqline(happiness$Family) )
```



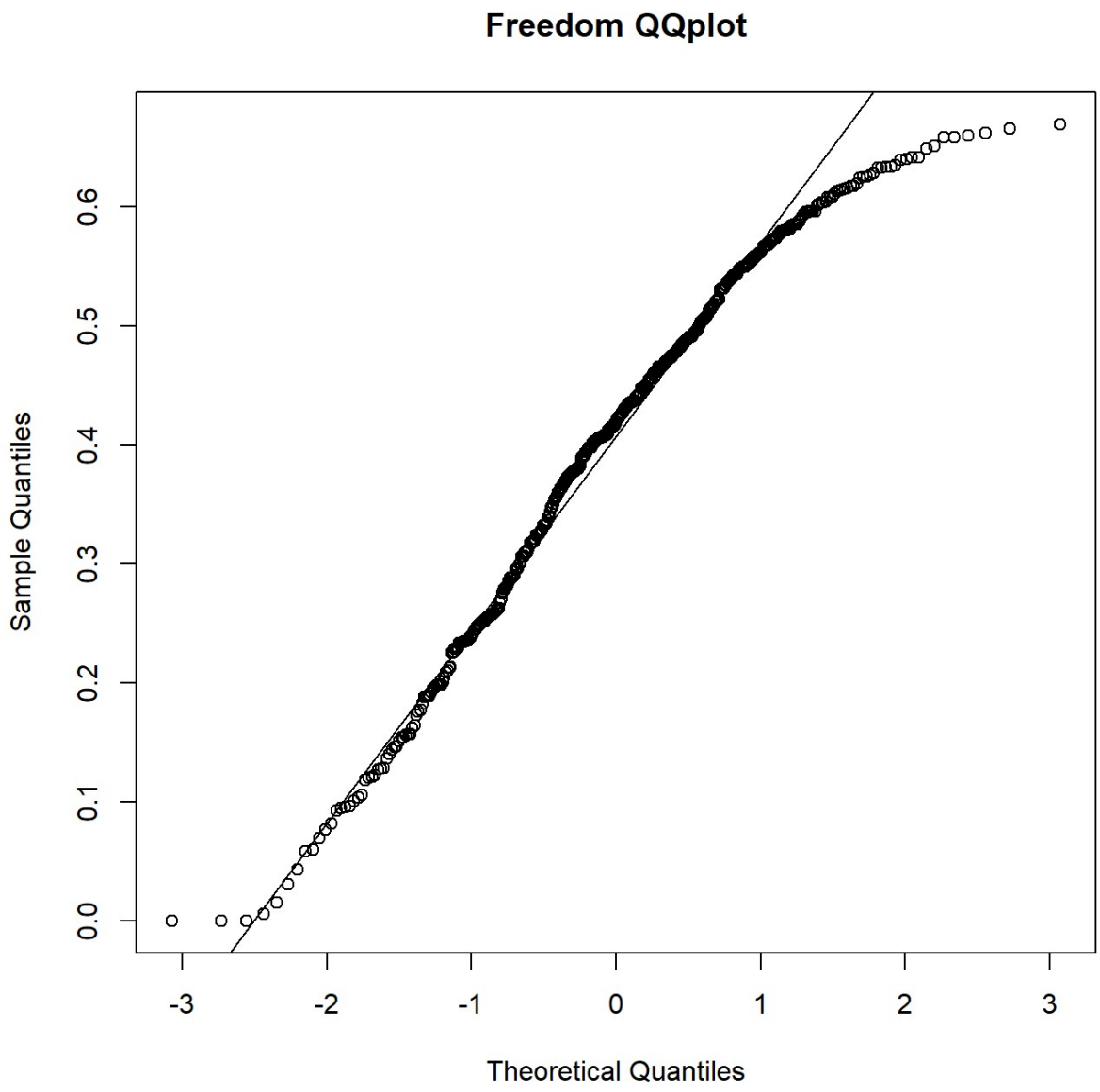

```
> # Efectúa el test de normalidad de Shapiro-Wilks
> with(happiness, shapiro.test(happiness$Family) )
```

```

    Shapiro-Wilk normality test

data:  happiness$Family
W = 0.97788, p-value = 1.451e-06
```

```
> # Dibuja un histograma
> with(happiness, hist(happiness$Freedom) )
```



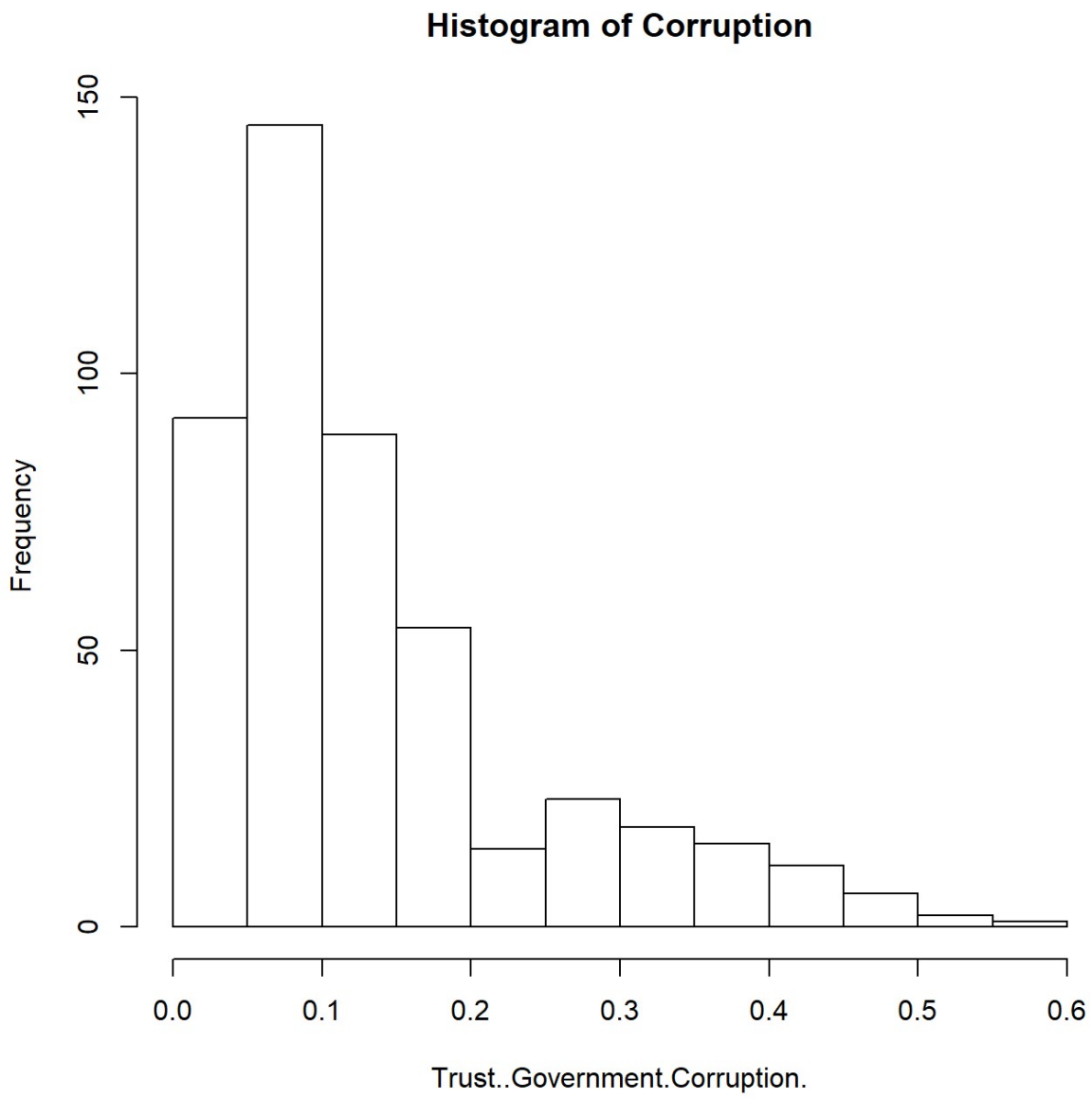

```
>  # Dibuja un qq plot
> with(happiness, qqnorm(happiness$Freedom, main="Freedom QQplot" ));with(happines
s, qqline(happiness$Freedom) )
```



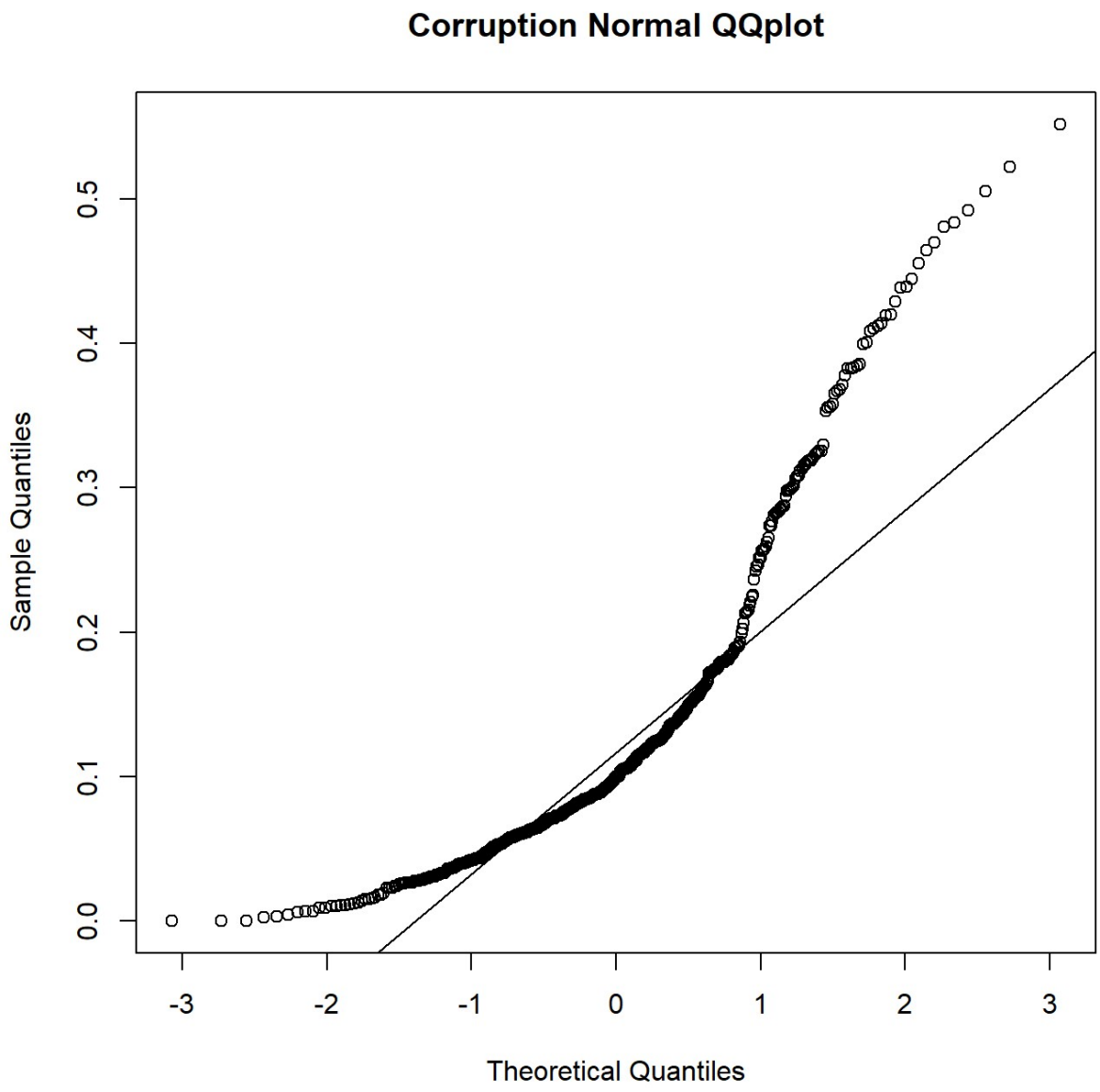

```
> # Efectúa el test de normalidad de Shapiro-Wilks
> with(happiness, shapiro.test(happiness$Freedom) )
```

```

    Shapiro-Wilk normality test

data:  happiness$Freedom
W = 0.97297, p-value = 1.246e-07
```

```
> # Dibuja un histograma
> with(happiness, hist(Trust..Government.Corruption.,  main="Histogram of Corruptio
n"))
```



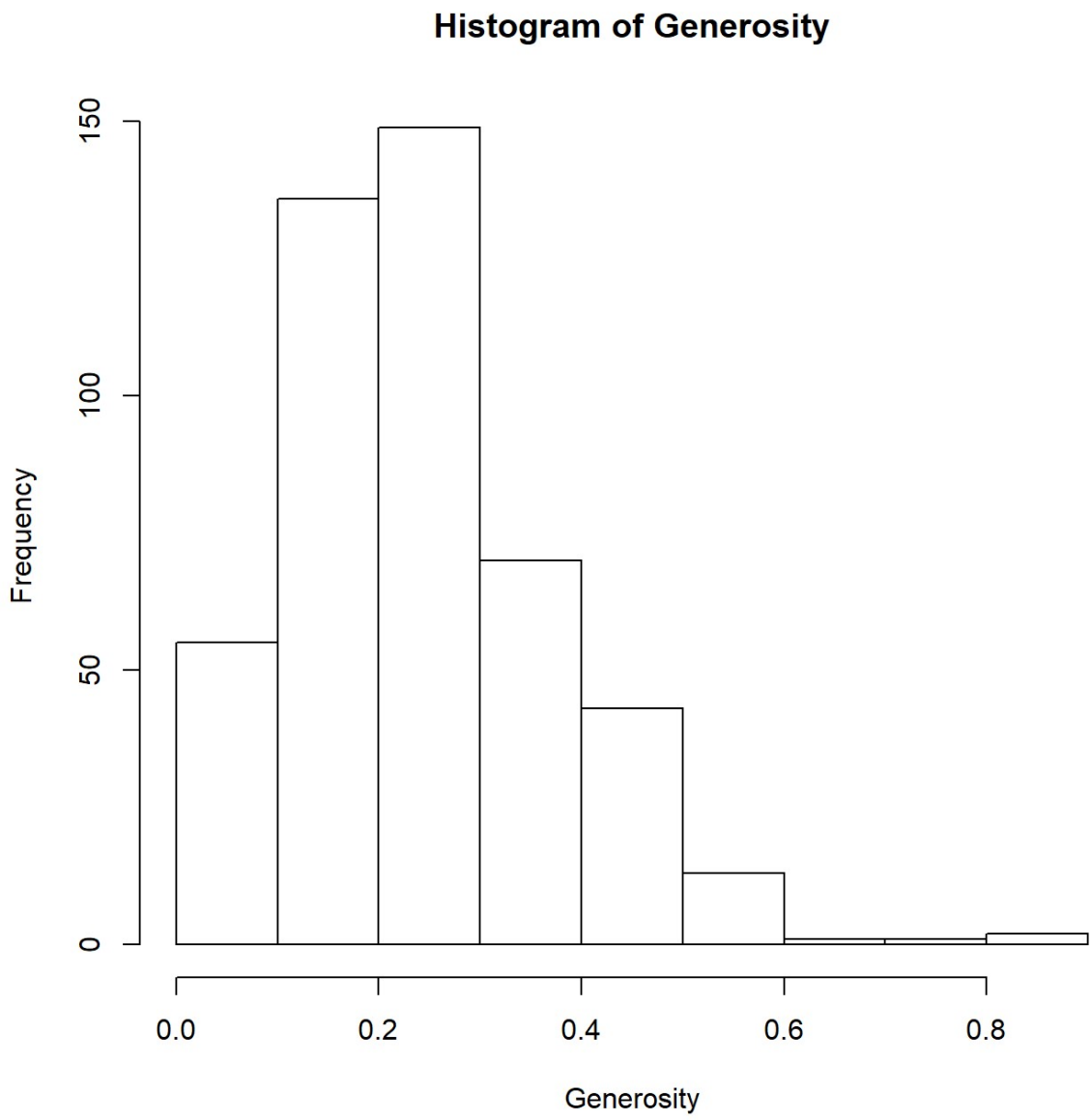

```
>  # Dibuja un qq plot
> with(happiness, qqnorm(happiness$Trust..Government.Corruption., main="Corruption
Normal QQplot" ));with(happiness, qqline(happiness$Trust..Government.Corruption.) )
```



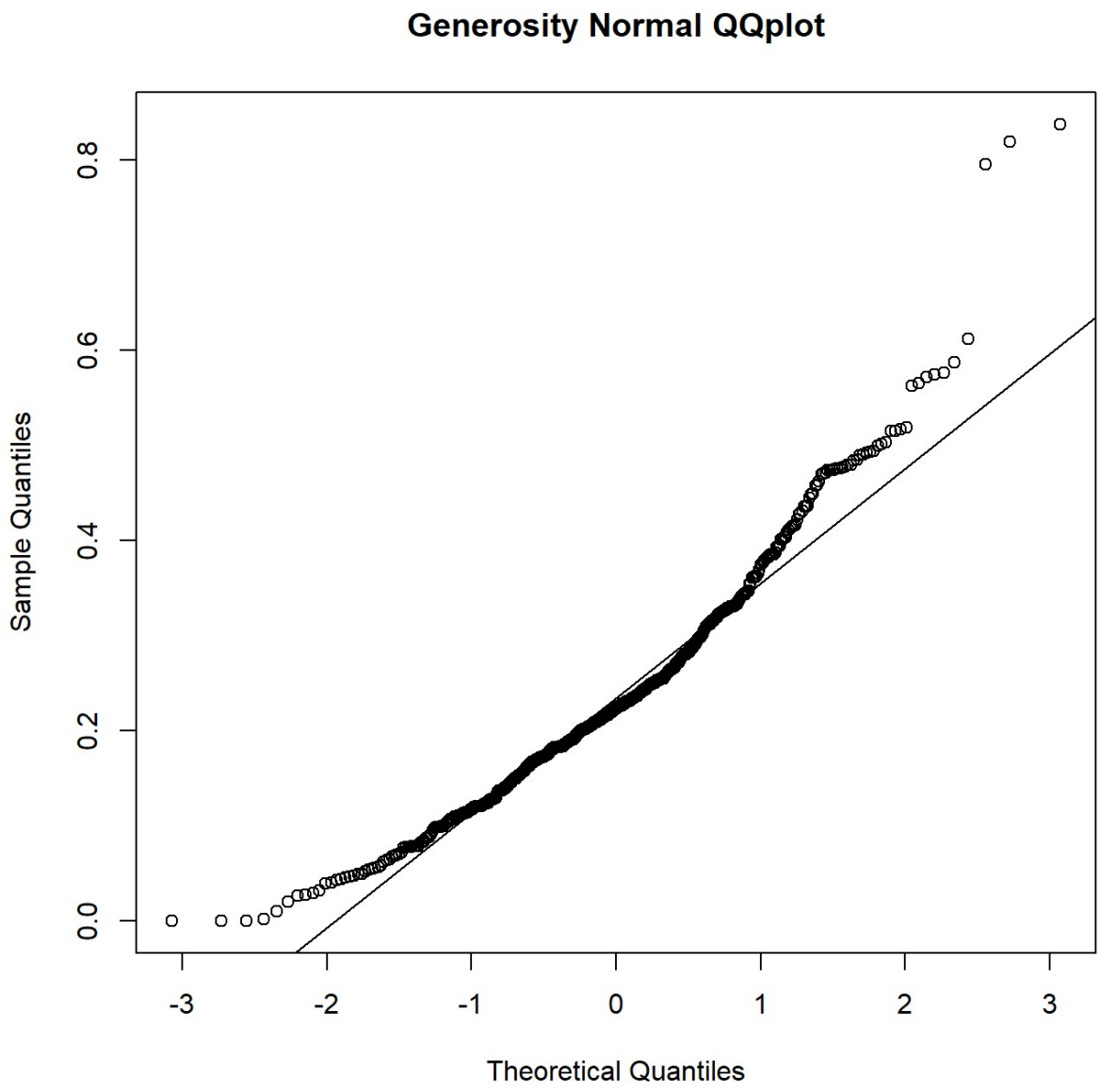

```
> # Efectúa el test de normalidad de Shapiro-Wilks
> with(happiness, shapiro.test(Trust..Government.Corruption.) )
```

```

    Shapiro-Wilk normality test

data:  Trust..Government.Corruption.
W = 0.85134, p-value < 2.2e-16
```

```
> # Dibuja un histograma
> with(happiness, hist(Generosity) )
```




```
>  # Dibuja un qq plot
> with(happiness, qqnorm(happiness$Generosity, main="Generosity Normal QQplot" ));w
ith(happiness, qqline(happiness$Generosity) )
```




```
> # Efectúa el test de normalidad de Shapiro-Wilks
> with(happiness, shapiro.test(Generosity) )
```

```
    Shapiro-Wilk normality test

data:  Generosity
W = 0.95165, p-value = 2.84e-11
```

Prueba de homogeneidad de varianzas Para ver la homogeneidad de varianzas, se utiliza un test de Fligner-Killeen.

```
> fligner.test(Happiness.Score~Economy..GDP.per.Capita., data = happiness)
```

```
    Fligner-Killeen test of homogeneity of variances

data:  Happiness.Score by Economy..GDP.per.Capita.
Fligner-Killeen:med chi-squared = 379.17, df = 466, p-value =
0.9987
```

```
> fligner.test(Happiness.Score~Family, data = happiness)
```

```
    Fligner-Killeen test of homogeneity of variances

data:  Happiness.Score by Family
Fligner-Killeen:med chi-squared = 468.46, df = 466, p-value =
0.4593
```

```
> fligner.test(Happiness.Score~Health..Life.Expectancy., data = happiness)
```

```
    Fligner-Killeen test of homogeneity of variances

data:  Happiness.Score by Health..Life.Expectancy.
Fligner-Killeen:med chi-squared = 418.33, df = 465, p-value =
0.941
```

```
> fligner.test(Happiness.Score~Freedom, data = happiness)
```

```
    Fligner-Killeen test of homogeneity of variances

data:  Happiness.Score by Freedom
Fligner-Killeen:med chi-squared = 447.18, df = 461, p-value =
0.6692
```

```
> fligner.test(Happiness.Score~Generosity, data = happiness)
```

```
    Fligner-Killeen test of homogeneity of variances

data:  Happiness.Score by Generosity
Fligner-Killeen:med chi-squared = 311.32, df = 467, p-value = 1
```

```
> fligner.test(Happiness.Score~Trust..Government.Corruption., data = happiness)
```

```
    Fligner-Killeen test of homogeneity of variances

data:  Happiness.Score by Trust..Government.Corruption.
Fligner-Killeen:med chi-squared = 425.98, df = 465, p-value =
0.9024
```

Excepto en el caso de la variable Family, para el resto podemos aceptar la hipótesis de que las varianzas son homogéneas al obtener un p-valor superior a 0,05.

Pruebas estadísticas Queremos ver, de forma general, cuáles son las variables con mayor relación con la felicidad.

```
> #Utilizamos un modelo de regresión múltiple sobre las variables que influyen en l
a felicidad.
> modelo = lm(happiness$Happiness.Score ~ ., data=happiness[,5:10])
> summary(modelo)
```

```
Call:
lm(formula = happiness$Happiness.Score ~ ., data = happiness[,
    5:10])

Residuals:
    Min      1Q  Median      3Q     Max
-1.7158 -0.3305 -0.0111  0.3768  1.6368

Coefficients:
                             Estimate Std. Error t value Pr(>|t|)
(Intercept)                    2.2166     0.1024  21.643  < 2e-16 ***
Economy..GDP.per.Capita.       1.0569     0.1131   9.346  < 2e-16 ***
Health..Life.Expectancy.       1.2783     0.1799   7.104 4.60e-12 ***
Family                         0.6103     0.1066   5.724 1.87e-08 ***
Freedom                        1.5398     0.2266   6.797 3.31e-11 ***
Trust..Government.Corruption.  0.8321     0.2795   2.977  0.00306 **
Generosity                     0.3712     0.2166   1.714  0.08719 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.5611 on 463 degrees of freedom
Multiple R-squared:  0.7596,    Adjusted R-squared:  0.7565
F-statistic: 243.9 on 6 and 463 DF,  p-value: < 2.2e-16
```

Vemos que la puntuación de la felicidad depende sobre todo de las variables Economy..GDP.per.Capita., Health..Life.Expectancy., Family y Freedom. Además, el test parcial sobre la variable Generosity tiene un p-valor > 0.05 por lo que no podemos suponer que la felicidad dependa de la generosidad. Como el valor de R2 es bastante alto, podemos suponer que tenemos bondad en el ajuste.

Podemos ver la evolución en los años

```
> #Para el año 2015
> happiness2015=happiness[which(happiness$Year==2015),]
> modelo2015 = lm(happiness2015$Happiness.Score ~ ., data=happiness2015[,5:10])
> summary(modelo2015)
```

```
Call:
lm(formula = happiness2015$Happiness.Score ~ ., data = happiness2015[,
    5:10])

Residuals:
     Min       1Q   Median       3Q      Max
-1.37567 -0.31918 -0.03547  0.37029  1.49919

Coefficients:
                             Estimate Std. Error t value Pr(>|t|)
(Intercept)                    1.7960     0.1968   9.126 4.24e-16 ***
Economy..GDP.per.Capita.       0.8615     0.2206   3.905 0.000142 ***
Health..Life.Expectancy.       1.0974     0.3179   3.452 0.000722 ***
Family                         1.4381     0.2284   6.298 3.13e-09 ***
Freedom                        1.1698     0.3939   2.970 0.003463 **
Trust..Government.Corruption.  0.8543     0.4386   1.948 0.053301 .
Generosity                     0.4114     0.3913   1.051 0.294796
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.5516 on 151 degrees of freedom
Multiple R-squared:  0.7768,    Adjusted R-squared:  0.7679
F-statistic: 87.59 on 6 and 151 DF,  p-value: < 2.2e-16
```

```
> #Para el año 2016
> happiness2016=happiness[which(happiness$Year==2016),]
> modelo2016 = lm(happiness2016$Happiness.Score ~ ., data=happiness2016[,5:10])
> summary(modelo2016)
```

```
Call:
lm(formula = happiness2016$Happiness.Score ~ ., data = happiness2016[,
    5:10])

Residuals:
     Min       1Q   Median       3Q      Max
-1.55205 -0.26937 -0.00426  0.31909  1.50449

Coefficients:
                             Estimate Std. Error t value Pr(>|t|)
(Intercept)                    2.2148     0.1664  13.309  < 2e-16 ***
Economy..GDP.per.Capita.       0.8668     0.2189   3.960 0.000116 ***
Health..Life.Expectancy.       1.4418     0.3602   4.003 9.81e-05 ***
Family                         0.9590     0.2279   4.208 4.42e-05 ***
Freedom                        1.6056     0.4022   3.992 0.000102 ***
Trust..Government.Corruption.  0.8214     0.4786   1.716 0.088219 .
Generosity                     0.2398     0.3733   0.642 0.521588
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.5539 on 150 degrees of freedom
Multiple R-squared:  0.7736,    Adjusted R-squared:  0.7646
F-statistic: 85.45 on 6 and 150 DF,  p-value: < 2.2e-16
```

```
> #Para el año 2017
> happiness2017=happiness[which(happiness$Year==2017),]
> modelo2017 = lm(happiness2017$Happiness.Score ~ ., data=happiness2017[,5:10])
> summary(modelo2017)
```

```
Call:
lm(formula = happiness2017$Happiness.Score ~ ., data = happiness2017[,
    5:10])

Residuals:
     Min       1Q   Median       3Q      Max
-1.52300 -0.23246 -0.01927  0.27793  1.25706

Coefficients:
                              Estimate Std. Error t value Pr(>|t|)
(Intercept)                     1.7208     0.1991   8.641 8.33e-15 ***
Economy..GDP.per.Capita.        0.8430     0.2054   4.105 6.67e-05 ***
Health..Life.Expectancy.        1.3332     0.3258   4.093 6.99e-05 ***
Family                          1.0527     0.2078   5.065 1.20e-06 ***
Freedom                         1.5175     0.3471   4.373 2.30e-05 ***
Trust..Government.Corruption.   0.7916     0.4909   1.612    0.109
Generosity                      0.3621     0.3341   1.084    0.280
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.5068 on 148 degrees of freedom
Multiple R-squared:  0.8071,    Adjusted R-squared:  0.7993
F-statistic: 103.2 on 6 and 148 DF,  p-value: < 2.2e-16
```

Vemos que la puntuación de la felicidad depende sobre todo de las variables Economy..GDP.per.Capita., Health..Life.Expectancy., Family y Freedom. Además, revisando los datos de los 3 años, vemos que el p-valor para las variables Generosity y Trust..Government.Corruption. es superior a 0.05 por lo que no podemos suponer que la felicidad dependa de esas variables.

Ejercicio 5

Mediante la creación de gráficos de dispersión, podemos ver gráficamente los resultados anteriores.

```
> # Dibujamos cuatro gráficos por página
> par(mfrow=c(3,2))
> plot(happiness$Happiness.Score, happiness$Economy..GDP.per.Capita., xlab = "Felic
idad", ylab = "PIB", title("Relación felicidad-economía"))
> plot(happiness$Happiness.Score, happiness$Family, xlab = "Felicidad", ylab = "Fam
ilia", title("Relación felicidad-familia"))
> plot(happiness$Happiness.Score, happiness$Health..Life.Expectancy., xlab = "Healt
h..Life.Expectancy.", ylab = "Salud", title("Relación felicidad-salud"))
> plot(happiness$Happiness.Score, happiness$Freedom, xlab = "Felicidad", ylab = "Li
bertad", title("Relación felicidad-libertad"))
> plot(happiness$Happiness.Score, happiness$Trust..Government.Corruption., xlab =
"Felicidad", ylab = "Trust..Government.Corruption.", title("Relación felicidad-corr
upción"))
> plot(happiness$Happiness.Score, happiness$Generosity, xlab = "Felicidad", ylab =
"Generosity", title ("Relación felicidad-generosidad"))
```