Práctica 2 - Tipología y ciclo de vida de los datos

Nicola Bafundi

23/05/2020

Table of Contents

El objetivo de esta práctica es, a partir de un dataset elegido, identificar los datos relevantes para un proyecto analítico y usar las herramientas de integración, limpieza, validación y análisis de las mismas.

1. Descripción del dataset elegido.

El dataset elegido fue el obtenido en la primera práctica de la asignatura Tipología y ciclo de vida de los datos.

El conjunto de datos contiene la información de la evolución de los casos detectados de COVID-19 por día para 8 países diferentes repartidos por todo el mundo. En concreto, se dispone de los registros de casos activos totales acumulados, de casos nuevos detectados, de decesos y de recuperaciones de COVID-19. Además, se recogen también el nivel de diversas partículas contaminantes en el aire para una de las ciudades más pobladas de los países seleccionados.

Los 8 países con su ciudad correspondiente son los siguientes:

- España (Madrid)
- Argentina (Buenos Aires)
- Alemania (Berlín)
- Inglaterra (Londres)
- Italia (Roma)
- China (Pekín)
- Francia (París)
- Estados Unidos (Nueva York)

Las partículas seleccionadas para determinar el nivel de calidad del aire son las siguientes:

- PM2,5: partículas de 2,5 μm de diámetro o menor que pueden incluir sustancias químicas orgánicas, polvo, hollín y metales
- PM10: partículas sólidas o líquidas de polvo, cenizas, hollín, partículas metálicas, cemento o polen, dispersas en la atmósfera, y cuyo diámetro varía entre 2,5 y 10 μm
- O3 (Ozono): gas tóxico que a concentraciones elevadas puede tener efectos en la salud humana, afectando principalmente al aparato respiratorio e irritando las mucosas, pudiendo llegar a producir afecciones pulmonares.
- NO2 (dióxido de nitrógeno): compuesto químico gaseoso de color marrón amarillento formado por la combinación de un átomo de nitrógeno y dos de oxígeno. Es un gas tóxico e irritante.
- SO2 (dióxido de azufre): es un gas que se origina sobre todo durante la combustión de carburantes fósiles que contienen azufre (petróleo, combustibles sólidos). Tiene efectos adversos sobre la salud.

- CO (monóxido de carbono): es un gas tóxico, inodoro, incoloro e insípido, parcialmente soluble en agua, alcohol y benceno, resultado de la oxidación incompleta del carbono durante el proceso de combustión.

A continuación procedemos a leer el conjunto de datos obtenido:

```
data<-read.csv("./COVID19_Pollution_Dataset.csv",header=T,sep=",",na.strings = "null")

summary(data)
```

```
##        X              Country               City             Date
##  Min.   :  0.0   China    : 80   Beijing     : 80   01-03-2020:  8
##  1st Qu.:117.8   Argentina: 56   Berlin      : 56   01-04-2020:  8
##  Median :235.5   France   : 56   Buenos Aires: 56   02-03-2020:  8
##  Mean   :235.5   Germany  : 56   London      : 56   02-04-2020:  8
##  3rd Qu.:353.2   Italy    : 56   Madrid      : 56   03-03-2020:  8
##  Max.   :471.0   Spain    : 56   New York    : 56   03-04-2020:  8
##                  (Other)  :112   (Other)     :112   (Other)   :424
##   Active.Cases    Daily.New.Cases   Daily.New.Deaths Newly.Recovered
##  Min.   :     0   Min.   :    0.0   Min.   :   0.0   Min.   :   -1.0
##  1st Qu.:    78   1st Qu.:   27.5   1st Qu.:   0.0   1st Qu.:    0.0
##  Median :  2837   Median :  391.0   Median :  10.0   Median :   41.5
##  Mean   : 25528   Mean   : 2713.2   Mean   : 173.7   Mean   :  674.6
##  3rd Qu.: 33285   3rd Qu.: 3616.5   3rd Qu.: 143.8   3rd Qu.:  988.5
##  Max.   :456815   Max.   :34196.0   Max.   :2035.0   Max.   :10219.0
##                   NA's   :25                         NA's   :112
##      PM2.5             PM10             O3              NO2
##  Min.   : 11.00   Min.   :  2.00   Min.   : 5.00   Min.   : 1.00
##  1st Qu.: 34.00   1st Qu.: 14.00   1st Qu.:21.00   1st Qu.: 8.00
##  Median : 46.00   Median : 20.00   Median :27.00   Median :14.00
##  Mean   : 59.43   Mean   : 25.14   Mean   :26.17   Mean   :16.38
##  3rd Qu.: 68.00   3rd Qu.: 31.00   3rd Qu.:32.00   3rd Qu.:24.00
##  Max.   :261.00   Max.   :119.00   Max.   :55.00   Max.   :47.00
##  NA's   :62       NA's   :74       NA's   :66      NA's   :68
##       SO2               CO
##  Min.   : 0.000   Min.   : 0.000
##  1st Qu.: 0.000   1st Qu.: 0.000
##  Median : 1.000   Median : 0.000
##  Mean   : 2.163   Mean   : 2.782
##  3rd Qu.: 3.000   3rd Qu.: 5.000
##  Max.   :21.000   Max.   :23.000
##  NA's   :134      NA's   :192
```

```
str(data)
```

```
## 'data.frame':    472 obs. of  14 variables:
##  $ X              : int  0 1 2 3 4 5 6 7 8 9 ...
##  $ Country        : Factor w/ 8 levels "Argentina","China",..: 6 6 6 6 6 6 6 6 6 6 ...
##  $ City           : Factor w/ 8 levels "Beijing","Berlin",..: 5 5 5 5
```

```
5 5 5 5 5 5 ...
##  $ Date            : Factor w/ 80 levels "01-02-2020","01-03-2020",..:
39 41 43 45 47 49 51 54 57 60 ...
##  $ Active.Cases    : int  0 0 0 0 0 0 0 0 0 1 ...
##  $ Daily.New.Cases : int  NA 0 0 0 0 0 0 0 0 1 ...
##  $ Daily.New.Deaths: int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Newly.Recovered : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ PM2.5           : int  79 80 73 57 44 43 67 59 68 56 ...
##  $ PM10            : int  34 28 21 17 21 33 30 34 30 33 ...
##  $ O3              : int  20 24 14 19 26 13 23 23 30 27 ...
##  $ NO2             : int  26 24 23 25 29 41 41 43 34 39 ...
##  $ SO2             : int  2 2 2 3 3 4 4 4 3 3 ...
##  $ CO              : int  0 0 0 0 0 0 0 0 0 0 ...
```

Como se puede ver en la captura anterior, el dataset estará formado por los siguiente campos:

- Country: Nombre del país, en inglés, empezando en mayúscula.
- City: Nombre de la ciudad, en inglés, empezando en mayúscula.
- Date: fecha en formato DD/MM/YYYY.
- Active Cases: número entero que indica los casos activos totales registrados de COVID-19 en el día en concreto.
- Daily New cases; número entero que indica el incremento de casos positivos en el día en concreto.
- Daily New Deaths: número entero que indica la cantidad de muertes por COVID-19 registradas el día en concreto.
- Newly recovered: número entero que indica la cantidad de pacientes recuperados de COVID-19 ese día.
- PM2.5: medición en ug/m3 de partículas de 2,5 um de diámetro o menor.
- PM10: medición en ug/m3 de partículas de 10 um de diámetro o menor.
- O3: medición en ug/m3 de moléculas de ozono.
- NO2: medición en ug/m3 de moléculas de dióxido de nitrógeno.
- SO2: medición en ug/m3 de moléculas de dióxido de azufre.
- CO: medición en ug/m3 de moléculas de monóxido de carbono.

Cabe destacar que los datos recogen la información desde que el COVID-19 llega al país seleccionado hasta el 10/04/2020, fecha en la que se recogieron los datos.

Como se puede ver, el conjunto de datos relaciona la evolución del avance del COVID-19 en diversos países (medida en función de los nuevos casos de infectados, decesos y recuperados) con el nivel de calidad del aire de las ciudades más importantes.

La principal pregunta que se pretende responder es si el incremento de los casos en COVID-19 en un país se corresponde con un descenso en la contaminación de una de sus ciudades más importantes.

2. Integración y selección de los datos de interés a analizar

Debido a que los datos contienen información de 8 países diferentes, para limitar el análisis se decide escoger solamente uno de los países que marcaran los pasos para el estudio del resto de países en otra ocasión. Se decide España como país más interesante a elegir para el análisis, debido a que es el país dónde vive el autor de este estudio.

Primero se seleccionan los datos del dataset:

```
dataSpain <- data[which(data$Country == "Spain"),]
nrow(dataSpain)

## [1] 56

head(dataSpain)

##   X Country   City       Date Active.Cases Daily.New.Cases
## 1 0   Spain Madrid 15-02-2020            0              NA
## 2 1   Spain Madrid 16-02-2020            0               0
## 3 2   Spain Madrid 17-02-2020            0               0
## 4 3   Spain Madrid 18-02-2020            0               0
## 5 4   Spain Madrid 19-02-2020            0               0
## 6 5   Spain Madrid 20-02-2020            0               0
##   Daily.New.Deaths Newly.Recovered PM2.5 PM10 O3 NO2 SO2 CO
## 1                0               0    79   34 20  26   2  0
## 2                0               0    80   28 24  24   2  0
## 3                0               0    73   21 14  23   2  0
## 4                0               0    57   17 19  25   3  0
## 5                0               0    44   21 26  29   3  0
## 6                0               0    43   33 13  41   4  0

tail(dataSpain)

##     X Country   City       Date Active.Cases Daily.New.Cases
## 51 50   Spain Madrid 05-04-2020        80925            5478
## 52 51   Spain Madrid 06-04-2020        82897            5029
## 53 52   Spain Madrid 07-04-2020        84689            5267
## 54 53   Spain Madrid 08-04-2020        85407            6278
## 55 54   Spain Madrid 09-04-2020        85610            5002
## 56 55   Spain Madrid 10-04-2020        86524            5051
##    Daily.New.Deaths Newly.Recovered PM2.5 PM10 O3 NO2 SO2 CO
## 51              694            3861    42   14 34   5   2  0
## 52              700            2357    45   17 26   8   1  0
## 53              704            2771    56   26 24  13   0  0
## 54              747            4813    81   19 30  12   0  0
```

```
## 55              655            4144    61   18 29   9   1  0
## 56              634            3503    56   16 22   6   0  0
```

En el dataset obtenido, tenemos los datos para 56 días de España Se tienen datos hasta el día 14/04/2020, día en que se realizó la extracción. Debido a la poca cantidad de datos a analizar, se ha decidido volver a realizar la extracción hasta el día 01/06/2020. A continuación, se importan los nuevos datos obtenidos:

```
data<-read.csv("./COVID19_Pollution_Dataset - Updated.csv",header=T,sep=","
,na.strings = "null")

#Los separamos por países:

dataSpain <- data[which(data$Country == "Spain"),]
nrow(dataSpain)

## [1] 107

tail(dataSpain)

##       X Country   City       Date Active.Cases Daily.New.Cases
## 102 101   Spain Madrid 26-05-2020        59264             859
## 103 102   Spain Madrid 27-05-2020        59773             510
## 104 103   Spain Madrid 28-05-2020        60909            1137
## 105 104   Spain Madrid 29-05-2020        61565             658
## 106 105   Spain Madrid 30-05-2020        62225             664
## 107 106   Spain Madrid 31-05-2020        62424             201
##     Daily.New.Deaths Newly.Recovered PM2.5 PM10 O3 NO2 SO2 CO
## 102              280               0    81   30 34  10   3  0
## 103                1               0    66   26 38  10   3  0
## 104                1               0    67   23 40  10   3  0
## 105                2               0    60   31 47  12   3  0
## 106                4               0    65   31 51  13   3  0
## 107                2               0    69   34 46   7   3  0
```

Como se puede ver, después de la nueva extracción de los datos, aumentamos el número de observaciones. Estos son los datos originales a partir de los cúales se trabajará.

Volvemos a ver la estructura de los datos:

```
summary(dataSpain)

##        X          Country         City             Date
##  Min.   :  0.0   China:  0   Beijing:  0   01-03-2020:  1
##  1st Qu.: 26.5   Spain:107   Madrid :107   01-04-2020:  1
##  Median : 53.0                             01-05-2020:  1
##  Mean   : 53.0                             02-03-2020:  1
##  3rd Qu.: 79.5                             02-04-2020:  1
##  Max.   :106.0                             02-05-2020:  1
##                                            (Other)   :101
```

```
##   Active.Cases    Daily.New.Cases  Daily.New.Deaths  Newly.Recovered
##  Min.   :     0   Min.   :-372.0   Min.   :-1915.0   Min.   :    0
##  1st Qu.:  3888   1st Qu.: 472.2   1st Qu.:   14.5   1st Qu.:    0
##  Median : 58598   Median :2020.0   Median :  192.0   Median : 1104
##  Mean   : 47907   Mean   :2702.9   Mean   :  266.0   Mean   : 1858
##  3rd Qu.: 77165   3rd Qu.:4246.2   3rd Qu.:  487.5   3rd Qu.: 3285
##  Max.   :100106   Max.   :8271.0   Max.   :  961.0   Max.   :18368
##                   NA's   :1        NA's   :5         NA's   :1
##      PM2.5            PM10            O3              NO2
##  Min.   : 23.00   Min.   : 8.00   Min.   :13.00   Min.   : 2.0
##  1st Qu.: 40.00   1st Qu.:13.00   1st Qu.:27.00   1st Qu.: 8.0
##  Median : 49.00   Median :16.00   Median :31.00   Median :11.0
##  Mean   : 51.62   Mean   :20.13   Mean   :31.49   Mean   :13.5
##  3rd Qu.: 61.50   3rd Qu.:26.00   3rd Qu.:35.00   3rd Qu.:16.5
##  Max.   :109.00   Max.   :65.00   Max.   :51.00   Max.   :43.0
##
##       SO2              CO
##  Min.   :0.000   Min.   :0
##  1st Qu.:1.000   1st Qu.:0
##  Median :2.000   Median :0
##  Mean   :2.019   Mean   :0
##  3rd Qu.:3.000   3rd Qu.:0
##  Max.   :4.000   Max.   :0
##
```

```
str(dataSpain)
```

```
## 'data.frame':    107 obs. of  14 variables:
##  $ X              : int  0 1 2 3 4 5 6 7 8 9 ...
##  $ Country        : Factor w/ 2 levels "China","Spain": 2 2 2 2 2 2 2
2 2 2 ...
##  $ City           : Factor w/ 2 levels "Beijing","Madrid": 2 2 2 2 2
2 2 2 2 2 ...
##  $ Date           : Factor w/ 131 levels "01-02-2020","01-03-2020",..
: 57 61 65 69 73 77 81 86 91 96 ...
##  $ Active.Cases    : int  0 0 0 0 0 0 0 0 0 1 ...
##  $ Daily.New.Cases : int  NA 0 0 0 0 0 0 0 0 1 ...
##  $ Daily.New.Deaths: int  NA NA NA NA NA 0 0 0 0 0 ...
##  $ Newly.Recovered : int  NA 0 0 0 0 0 0 0 0 0 ...
##  $ PM2.5          : int  79 80 73 57 44 43 67 59 68 56 ...
##  $ PM10           : int  34 28 21 17 21 33 30 34 30 33 ...
##  $ O3             : int  20 24 14 19 26 13 23 23 30 27 ...
##  $ NO2            : int  26 24 23 25 29 41 41 43 34 39 ...
##  $ SO2            : int  2 2 2 3 3 4 4 4 3 3 ...
##  $ CO             : int  0 0 0 0 0 0 0 0 0 0 ...
```

De entre los campos originales, excluiremos los siguientes:

- “X”, este campo es el identificador de la fila para la extracción, para el análisis no es necesario.

- “City”, debido a que solamente se tienen los datos para una ciudad de cada país (Madrid y Beijing)
- “CO”, el campo que indica la cantidad de monóxido de carbono en el aire, en el caso de España no es relevante ya que siempre es “0”.
- “Country”, el campo que indica el país es relevante para distinguir las observaciones. Pero como solamente analizaremos las de España, se puede eliminar En el caso de ya tenerlas separado se excluirá.

Otros campos que se podrían excluir pero se mantienen por su posibilidad de ser interesantes para un análisis

- “Active Cases”, campo que indica la cantidad de casos activos totales en el país durante un día determinado. A pesar de que es un parámetro que se puede obtener a partir de los nuevos casos, recuperaciones y muertes diarias, puede ser interesante mantenerlo para compararlo con los datos de contaminación.

Asimismo, se puede ver que para algunos campos se obtienen valores perdidos (“NA’s”).

```
dataSpain$X <- NULL
dataSpain$City <- NULL
dataSpain$CO <-NULL
dataSpain$Country <- NULL
```

A continuación, se procede con la limpieza de los datos.

3. Limpieza de los datos

Antes de realizar el análisis, se buscarán aquellos valores extremos que puedan afectar significativamente al análisis de los datos así como la estrategia de tratamiento de datos perdidos.

Pero antes de todo, se realizará el cambio de tipo para la variable “Fecha” para convertirla en el tipo “Date”

```
dataSpain$Date <- as.Date(dataSpain$Date, "%d-%m-%Y")
```

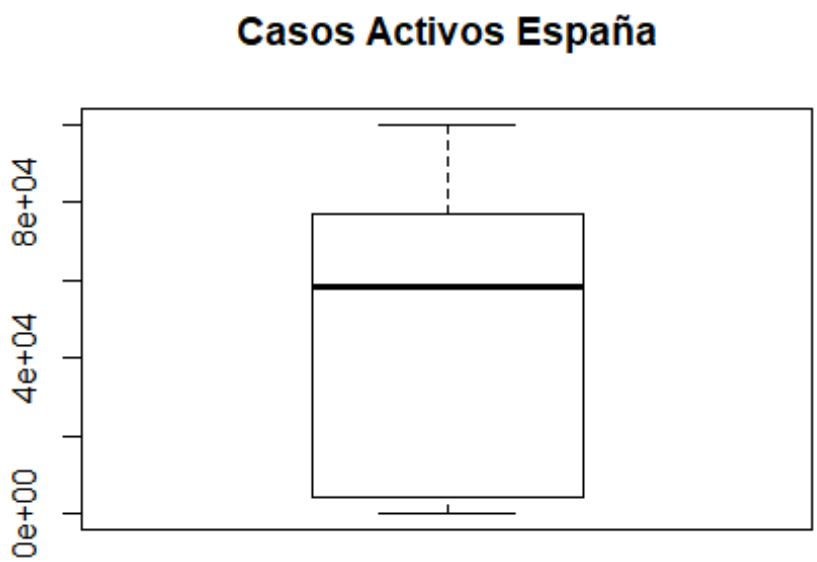
Valores extremos

Con los datos ya cambiados, procedemos a verificar los valores extremos de cada variable numérica para cada país. Para ello, utilizaremos la función boxplot, que nos permite identificar gráficamente cuales son los valores extremos.

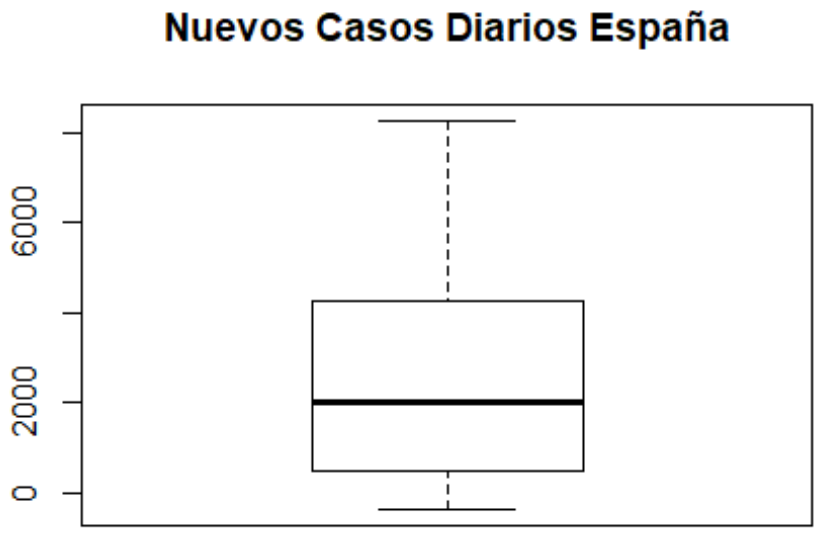
Consideraremos outliers, todos aquellos valores que se encuentre fuera del rango determinado por el boxplot, es decir aquellos cuyo valor este por encima o por debajo de la distancia entre los percentiles 25% y 75% de la distribución (rango intercuantílico) por 1.5.

Procedemos a mostrar las funciones boxplot para las variables:

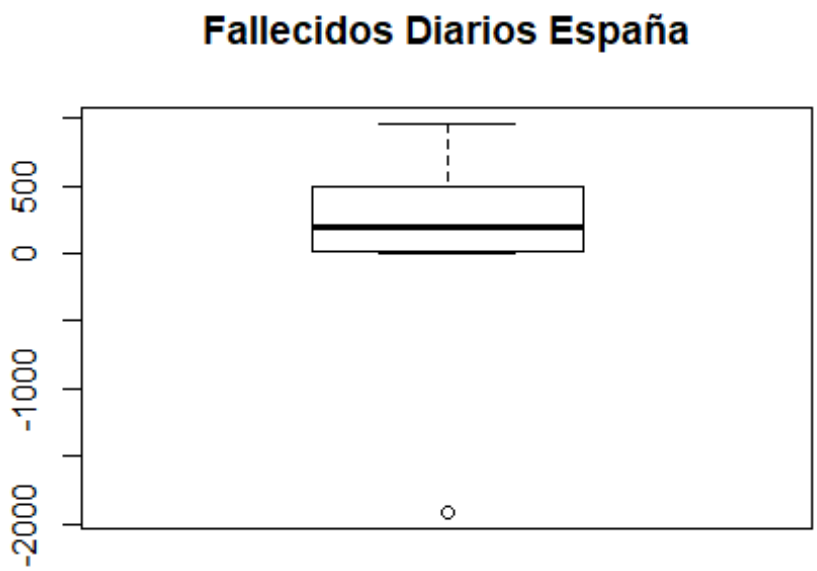
```
boxplot(dataSpain$Active.Cases)
title("Casos Activos España")
```



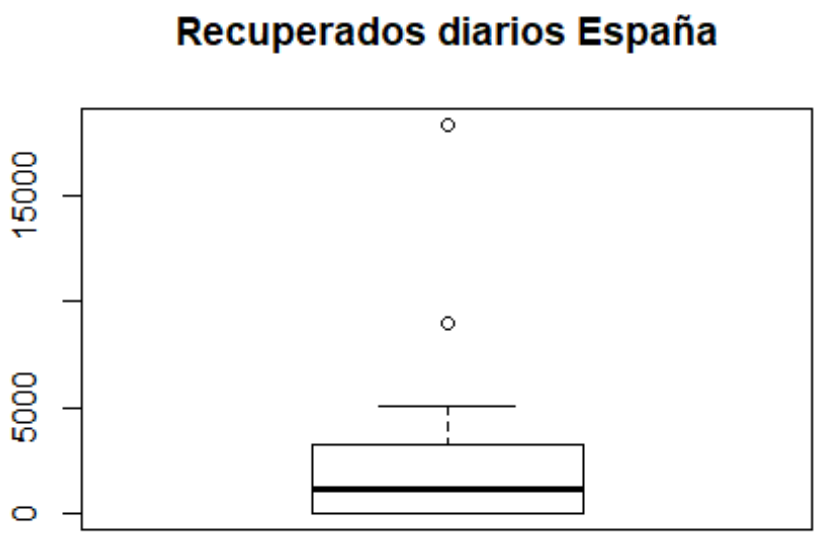

```r
boxplot(dataSpain$Daily.New.Cases)
title("Nuevos Casos Diarios España")
```



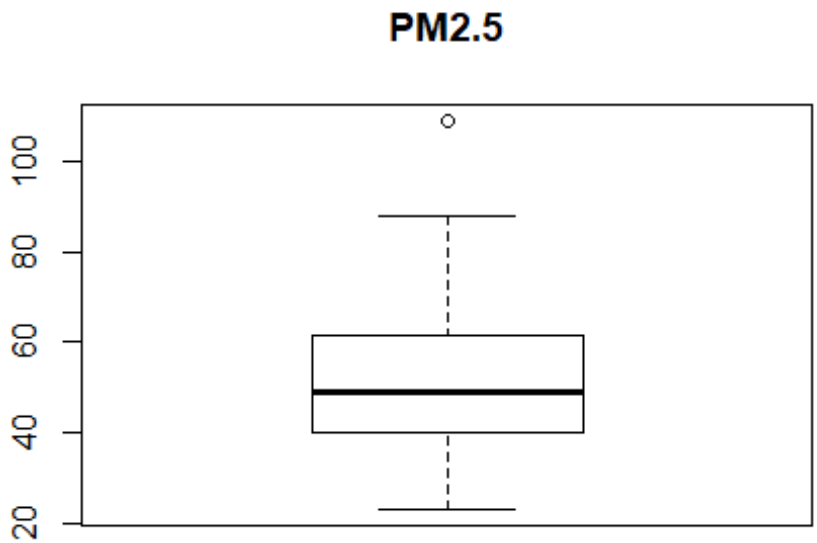
```r
boxplot(dataSpain$Daily.New.Deaths)
title("Fallecidos Diarios España")
```



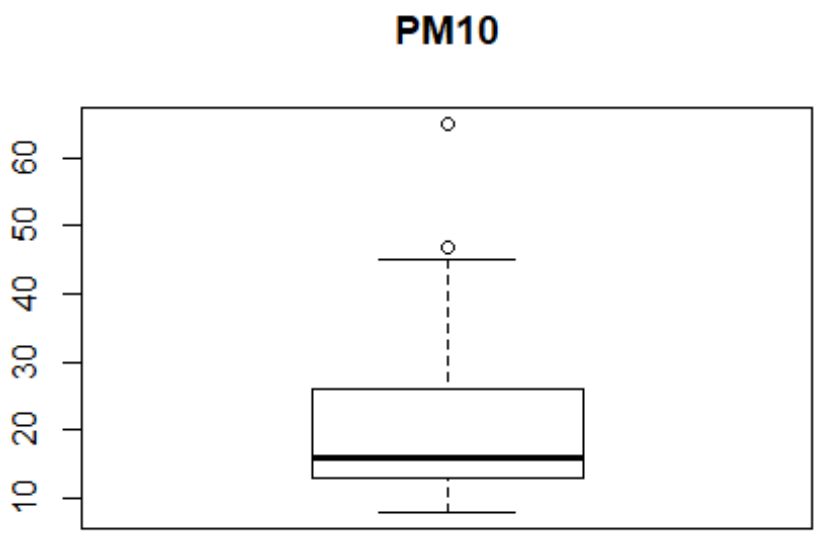

```r
boxplot(dataSpain$Newly.Recovered)
title("Recuperados diarios España")
```



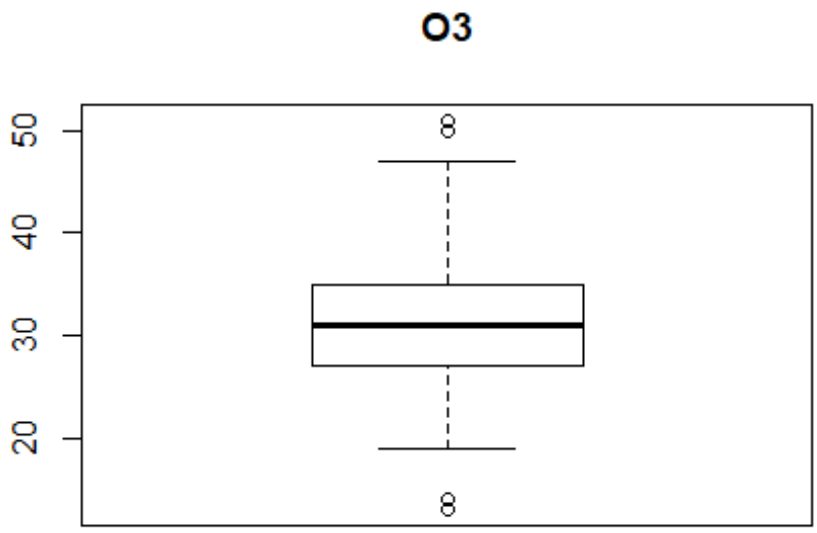

```r
boxplot(dataSpain$PM2.5)
title("PM2.5")
```



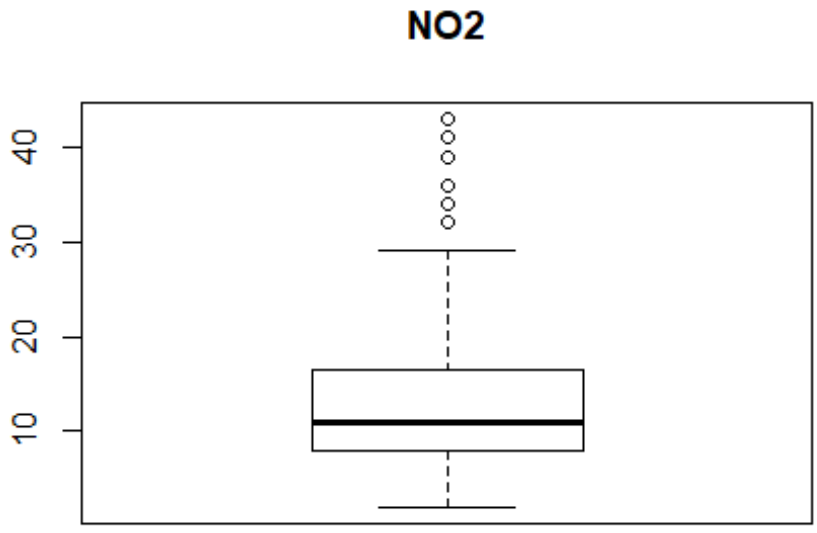
```r
boxplot(dataSpain$PM10)
title("PM10")
```



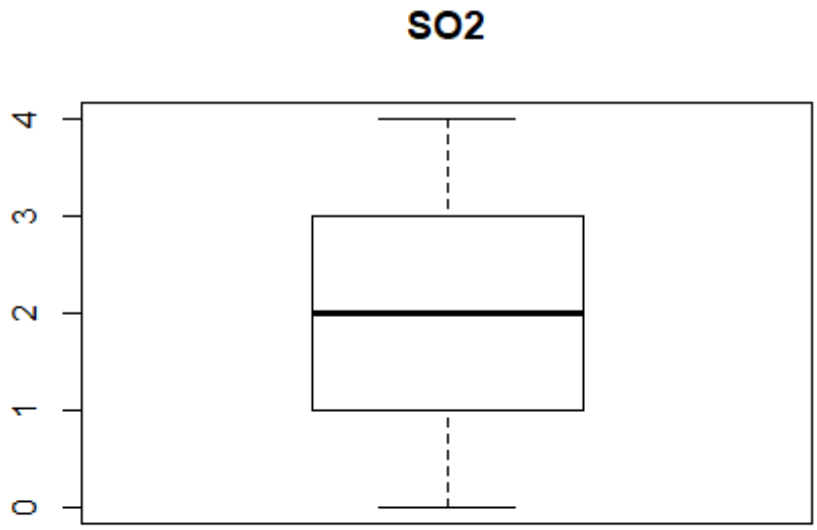

```r
boxplot(dataSpain$O3)
title("O3")
```




```r
boxplot(dataSpain$NO2)
title("NO2")
```



```r
boxplot(dataSpain$SO2)
title("SO2")
```



Detectamos Outliers para las siguientes variables:

- Fallecidos Diarios
- Recuperados diarios
- PM2.5
- PM10
- O3
- NO2

Procedemos a ver individualmente para los casos de Fallecidos y recuperaciones diarias para ver si los consideramos en el análisis, eliminamos la observación o bien la consideramos NA.

Para los outliers detectados en las variables de contaminación, se decide mantenerlos ya que pueden ser significativos para los análisis. Asimismo los valores extremos tampoco parecen ser “elevadamente extremos” o “raros” como para indicar que son un error.

Procedemos con los Fallecidos Diarios:

```r
bx <- boxplot(dataSpain$Daily.New.Deaths ,plot = FALSE)
bx$out
```

```
## [1] -1915
```

```
dataSpain[which(dataSpain$Daily.New.Deaths == bx$out ),]
```

```
##           Date Active.Cases Daily.New.Cases Daily.New.Deaths
## 101 2020-05-25        58685            -372            -1915
##     Newly.Recovered PM2.5 PM10 O3 NO2 SO2
## 101               0    78   45 33  10   3
```

```
#Ponemos como NA

dataSpain[which(dataSpain$Daily.New.Deaths == bx$out ),"Daily.New.Deaths"
] <- NA

#Se verifica si hay algún otro valor inferior a cero para los nuevos caso
s diarios

nrow(dataSpain[which(dataSpain$Daily.New.Cases < 0 ),])
```

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

```
# Se considera también como NA.
dataSpain[which(dataSpain$Daily.New.Cases < 0 ),"Daily.New.Cases"] <- NA
```

Como se puede ver en este caso el número de fallecidos diarios es negativo. Esto se debe a un error en el recuento de fallecidos del Ministerio de Sanidad debido a las validaciones posteriores de los datos enviados por las comunidades autónomas. ver el siguiente link con la noticia:

https://www.lasprovincias.es/sociedad/salud/fernando-simon-explica-2000-muertos-menos-coronavirus-20200525192539-nt.html?ns_campaign=jaqueton&ref=https:%2F%2Ft.co%2FZGELgdnWOT%3Famp%3D1

Se considera como dato perdido el valor de ese día pero, cabe indicar que el valor de nuevos casos diarios es negativo, ocasionado también por la misma casuística que el de fallecimientos.

Continuamos con las recuperaciones diarias:

```
bx <- boxplot(dataSpain$Newly.Recovered ,plot = FALSE)
bx$out
```

```
## [1] 18368  9026
```

```
dataSpain[which(dataSpain$Newly.Recovered %in% bx$out ),]
```

```
##           Date Active.Cases Daily.New.Cases Daily.New.Deaths
## 70  2020-04-24        88111            6740              367
## 75  2020-04-29        79695            4771              453
##     Newly.Recovered PM2.5 PM10 O3 NO2 SO2
## 70            18368    53   16 34   8   1
## 75             9026    39   12 30  10   2
```

```r
summary(dataSpain$Newly.Recovered)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##       0       0    1104    1858    3285   18368       1
```

```r
#Ponemos como NA

dataSpain[which(dataSpain$Newly.Recovered == 18368 ),"Newly.Recovered"] <
- NA
```

En este caso, podemos ver que los valores extremos son para esos días en que se registraron valores más altos de lo normal para las recuperaciones. El caso del día 24/04/2020 puede ser debido a la acumulación de recuperaciones durante los días previos que no fueron registradas previamente. Por esta razón y porque es bastante distante respecto al resto de observaciones, se considerará como NA para evitar su influencia dentro de la muestra. A pesar de ello, sí se decide mantener el otro outlier.

Datos perdidos

A continuación, procedemos a ver los datos perdidos (NA) en el subconjunto de España:

```r
summary(dataSpain)
```

```
##       Date              Active.Cases     Daily.New.Cases Daily.New.Death
s
##  Min.   :2020-02-15   Min.   :     0   Min.   :   0    Min.   :  0.0
##  1st Qu.:2020-03-12   1st Qu.:  3888   1st Qu.: 482    1st Qu.: 19.0
##  Median :2020-04-08   Median : 58598   Median :2086    Median :193.0
##  Mean   :2020-04-08   Mean   : 47907   Mean   :2732    Mean   :287.5
##  3rd Qu.:2020-05-04   3rd Qu.: 77165   3rd Qu.:4258    3rd Qu.:499.0
##  Max.   :2020-05-31   Max.   :100106   Max.   :8271    Max.   :961.0
##                                        NA's   :2       NA's   :6
##  Newly.Recovered     PM2.5             PM10             O3
##  Min.   :   0    Min.   : 23.00   Min.   : 8.00   Min.   :13.00
##  1st Qu.:   0    1st Qu.: 40.00   1st Qu.:13.00   1st Qu.:27.00
##  Median :1013    Median : 49.00   Median :16.00   Median :31.00
##  Mean   :1701    Mean   : 51.62   Mean   :20.13   Mean   :31.49
##  3rd Qu.:3282    3rd Qu.: 61.50   3rd Qu.:26.00   3rd Qu.:35.00
##  Max.   :9026    Max.   :109.00   Max.   :65.00   Max.   :51.00
##  NA's   :2
##       NO2             SO2
##  Min.   : 2.0   Min.   :0.000
##  1st Qu.: 8.0   1st Qu.:1.000
##  Median :11.0   Median :2.000
##  Mean   :13.5   Mean   :2.019
##  3rd Qu.:16.5   3rd Qu.:3.000
##  Max.   :43.0   Max.   :4.000
##
```

Como se puede observar en el resumen, hay tres variables con valores perdidos. Realizaremos la imputación de los valores perdidos con la función KNN (K - Nearest Neighbours) de la librería VIM.

Para ello, solamente utilizaremos las variables casos activos, casos diarios, fallecidos diarios y recuperados diarios.

```
library("VIM")

#Seleccionamos el dataset a partir de los datos indicados
quant.dataSpain <- dataSpain[,c(2:5)]

#Con la librería VIM cargada, ejecutamos la imputación de valores para k
= 5 utilizando la función kNN que utiliza la distancia de Gower.

quant.input <- kNN(quant.dataSpain, k=5)

#Vemos cuales son los valores imputados y los comparamos con los original
es

idxInput <- which(quant.input$Daily.New.Cases_imp == TRUE | quant.input$D
aily.New.Deaths_imp == TRUE | quant.input$Newly.Recovered_imp)

quant.input[idxInput,1:4]
```

```
##     Active.Cases Daily.New.Cases Daily.New.Deaths Newly.Recovered
## 1              0               0                0               0
## 2              0               0                0               0
## 3              0               0                0               0
## 4              0               0                0               0
## 5              0               0                0               0
## 70         88111            6740              367            3944
## 101        58685               0               50               0
```

```
quant.dataSpain[idxInput,]
```

```
##     Active.Cases Daily.New.Cases Daily.New.Deaths Newly.Recovered
## 1              0              NA               NA              NA
## 2              0               0               NA               0
## 3              0               0               NA               0
## 4              0               0               NA               0
## 5              0               0               NA               0
## 70         88111            6740              367              NA
## 101        58685              NA               NA               0
```

```
#Asignamos los valores imputados al dataSpain

quant.dataSpain[idxInput,] <- quant.input[idxInput,1:4]

dataSpain[,c(2:5)] <- quant.dataSpain
```

Commparando los valores imputados, se observa que el valor de casos diarios para la fila 101 es 0, que en contraste con las observaciones anteriores y posteriores, 0 no es el valor adecuado para la fila, por eso se decide recalcular el valor de ese campo utilizando la fórmula que los casos activos = casos activos del dia anterior + casos nuevos diarios - fallecimientos diarios - recuperaciones diarias:

```
dataSpain[101,"Daily.New.Cases"] <- dataSpain[101,"Active.Cases"] - dataS
pain[100,"Active.Cases"] + dataSpain[101,"Daily.New.Deaths"] + dataSpain[
101,"Newly.Recovered"]

dataSpain[99:105,]
```

```
##           Date Active.Cases Daily.New.Cases Daily.New.Deaths
## 99  2020-05-23        56734             466               50
## 100 2020-05-24        57142             482               74
## 101 2020-05-25        58685            1593               50
## 102 2020-05-26        59264             859              280
## 103 2020-05-27        59773             510                1
## 104 2020-05-28        60909            1137                1
## 105 2020-05-29        61565             658                2
##     Newly.Recovered PM2.5 PM10 O3 NO2 SO2
## 99                0    76   40 45  10   2
## 100               0   109   28 42   3   3
## 101               0    78   45 33  10   3
## 102               0    81   30 34  10   3
## 103               0    66   26 38  10   3
## 104               0    67   23 40  10   3
## 105               0    60   31 47  12   3
```

Ahora el valor imputado tiene más sentido.

Una vez con los valores imputados, procedemos a hacer un resumen de los datos:

```
summary(dataSpain)
```

```
##       Date             Active.Cases    Daily.New.Cases  Daily.New.Deat
hs
##  Min.   :2020-02-15   Min.   :     0   Min.   :   0.0   Min.   :  0.0
##  1st Qu.:2020-03-12   1st Qu.:  3888   1st Qu.: 475.5   1st Qu.:  5.5
##  Median :2020-04-08   Median : 58598   Median :1954.0   Median :184.0
##  Mean   :2020-04-08   Mean   : 47907   Mean   :2696.0   Mean   :271.9
##  3rd Qu.:2020-05-04   3rd Qu.: 77165   3rd Qu.:4234.5   3rd Qu.:446.5
##  Max.   :2020-05-31   Max.   :100106   Max.   :8271.0   Max.   :961.0
##  Newly.Recovered     PM2.5             PM10             O3
##  Min.   :   0    Min.   : 23.00   Min.   : 8.00   Min.   :13.00
##  1st Qu.:   0    1st Qu.: 40.00   1st Qu.:13.00   1st Qu.:27.00
##  Median :1013    Median : 49.00   Median :16.00   Median :31.00
##  Mean   :1706    Mean   : 51.62   Mean   :20.13   Mean   :31.49
##  3rd Qu.:3284    3rd Qu.: 61.50   3rd Qu.:26.00   3rd Qu.:35.00
##  Max.   :9026    Max.   :109.00   Max.   :65.00   Max.   :51.00
##       NO2             SO2
```

```
##  Min.   : 2.0   Min.   :0.000
##  1st Qu.: 8.0   1st Qu.:1.000
##  Median :11.0   Median :2.000
##  Mean   :13.5   Mean   :2.019
##  3rd Qu.:16.5   3rd Qu.:3.000
##  Max.   :43.0   Max.   :4.000
```

Ya no tenemos valores perdidos en los datos y los podemos considerar como limpios. A continuación, procedemos a exportarlos:

```
write.csv(dataSpain,"./COVID19_Pollution_Dataset_Clean.csv")
```

4. Análisis de los datos y representación gráfica de los resultados

Después de tratar los valores extremos y perdidos en los datos, procedemos a realizar la fase de análisis de los datos. Primero de todo, seleccionaremos los grupos de datos a analizar. De momento, solamente se excluirá la variable fecha que es la única variable "categórica" de la que se dispone. En función de los resultados de los test que se aplicaran a los datos se formarán diversos grupos de datos.

```
ds <- dataSpain
ds$Date <- NULL
```

Normalidad

A continuación, comprobaremos si los datos obtenidos siguen una distribución normal. Para ello, aplicaremos el test de shapiro-wilk, en el que se asume como hipótesis nula que la población está distribuida normalmente, si el p-valor es menor al nivel de significancia (0.05), entonces la hipótesis nula es rechazada y se concluye que los datos no cuentan con una distribución normal.

```
shapiro.test(ds$Active.Cases)

##
##  Shapiro-Wilk normality test
##
## data:  ds$Active.Cases
## W = 0.86687, p-value = 2.335e-08

shapiro.test(ds$Daily.New.Cases)

##
##  Shapiro-Wilk normality test
##
## data:  ds$Daily.New.Cases
## W = 0.88852, p-value = 2.007e-07

shapiro.test(ds$Daily.New.Deaths)

##
##  Shapiro-Wilk normality test
##
## data:  ds$Daily.New.Deaths
## W = 0.8601, p-value = 1.246e-08

shapiro.test(ds$Newly.Recovered)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  ds$Newly.Recovered
## W = 0.82233, p-value = 5.105e-10
```

Para los datos relacionados con el COVID-19 observamos que no siguen una distribución normal, ya que todos los p-valores son inferiores a 0.05

```
shapiro.test(ds$PM2.5)

##
##  Shapiro-Wilk normality test
##
## data:  ds$PM2.5
## W = 0.97016, p-value = 0.01644

shapiro.test(ds$PM10)

##
##  Shapiro-Wilk normality test
##
## data:  ds$PM10
## W = 0.82022, p-value = 4.329e-10

shapiro.test(ds$O3)

##
##  Shapiro-Wilk normality test
##
## data:  ds$O3
## W = 0.97954, p-value = 0.09793

shapiro.test(ds$NO2)

##
##  Shapiro-Wilk normality test
##
## data:  ds$NO2
## W = 0.83823, p-value = 1.848e-09

shapiro.test(ds$SO2)

##
##  Shapiro-Wilk normality test
##
## data:  ds$SO2
## W = 0.87791, p-value = 6.792e-08
```

De los valores relacionados con la contaminación, solamente podemos decir que puede seguir una distribución normal es la variable O3, el ozono.

Conocer si una variable sigue una distribución normal o no, nos permitirá aplicar un test o otro durante el análisis.

Homocedasticidad

A continuación, comprobaremos si las variables son homogéneas, es decir, tienen la misma varianza. Para ello, como la mayoría de los datos no siguen una distribución no normal, se utilizará el test de Fligner-Killeen. Para ello, comprobaremos la homocedasticidad de la variable Casos Activos con las variables con los datos de contaminación.

```
fligner.test(Active.Cases ~ PM2.5, data = ds)

##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  Active.Cases by PM2.5
## Fligner-Killeen:med chi-squared = 46.663, df = 48, p-value =
## 0.5277

fligner.test(Active.Cases ~ PM10, data = ds)

##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  Active.Cases by PM10
## Fligner-Killeen:med chi-squared = 29.217, df = 29, p-value =
## 0.4538

fligner.test(Active.Cases ~ O3, data = ds)

##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  Active.Cases by O3
## Fligner-Killeen:med chi-squared = 37.954, df = 30, p-value =
## 0.1509

fligner.test(Active.Cases ~ NO2, data = ds)

##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  Active.Cases by NO2
## Fligner-Killeen:med chi-squared = 44.322, df = 31, p-value =
## 0.0572

fligner.test(Active.Cases ~ SO2, data = ds)

##
##  Fligner-Killeen test of homogeneity of variances
##
```

```
## data:  Active.Cases by SO2
## Fligner-Killeen:med chi-squared = 7.4448, df = 4, p-value = 0.1142
```

En este caso, la hipótesis nula es que se asume la igualdad de varianzas entre las variables, por lo que p-values inferiores al nivel de significancia (0.05) indican heterocedasticidad. En los resultados de las pruebas de homocedasticidad de las variables con datos de contaminación con los casos activos, obtenemos un p-value mayor al nivel de significancia por tanto podemos asumir la igualdad de varianzas entre las variables.

Correlaciones

A continuación, continuamos con el análisis de la correlatividad entre variables.

```
cor.res <- cor(ds, method = "spearman")
cor.res
```

```
##                   Active.Cases Daily.New.Cases Daily.New.Deaths
## Active.Cases        1.00000000       0.7149131        0.7428286
## Daily.New.Cases     0.71491305       1.0000000        0.9459618
## Daily.New.Deaths    0.74282857       0.9459618        1.0000000
## Newly.Recovered     0.72425016       0.7192525        0.7095728
## PM2.5              -0.09692362      -0.3173681       -0.2935634
## PM10               -0.37281470      -0.5504486       -0.5494179
## O3                  0.37972021       0.1847274        0.2268293
## NO2                -0.72053557      -0.5073285       -0.5146223
## SO2                -0.28069811      -0.4301789       -0.3810355
##                   Newly.Recovered       PM2.5       PM10          O3
## Active.Cases            0.7242502 -0.09692362 -0.3728147  0.37972021
## Daily.New.Cases         0.7192525 -0.31736812 -0.5504486  0.18472742
## Daily.New.Deaths        0.7095728 -0.29356344 -0.5494179  0.22682932
## Newly.Recovered         1.0000000 -0.32179219 -0.5196533  0.18050409
## PM2.5                  -0.3217922  1.00000000  0.5251895  0.01531684
## PM10                   -0.5196533  0.52518951  1.0000000 -0.12948473
## O3                      0.1805041  0.01531684 -0.1294847  1.00000000
## NO2                    -0.6174313  0.01315704  0.3170086 -0.39021380
## SO2                    -0.3529459  0.14333820  0.2943190 -0.05410671
##                             NO2         SO2
## Active.Cases        -0.72053557 -0.28069811
## Daily.New.Cases     -0.50732854 -0.43017888
## Daily.New.Deaths    -0.51462226 -0.38103550
## Newly.Recovered     -0.61743131 -0.35294591
## PM2.5                0.01315704  0.14333820
## PM10                 0.31700855  0.29431898
## O3                  -0.39021380 -0.05410671
## NO2                  1.00000000  0.32544975
## SO2                  0.32544975  1.00000000
```

Utilizando la función cor(), obtenemos el coeficiente de correlación entre dos variables. El coeficiente de correlación puede tomar valores entre -1 y 1 donde los

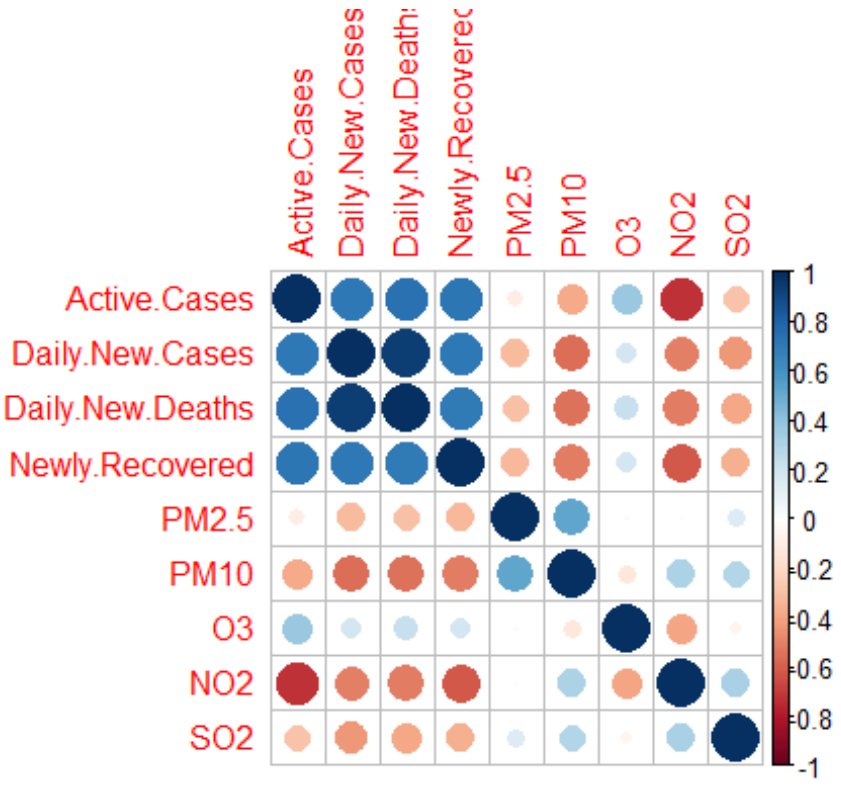
extremos indican una correlación perfecta y el 0 indica la ausencia de correlación. Si el signo es negativo significa que ha medida que crece una variable la otra disminuye, en cambio si el signo es positivo ambas variables tienden a incrementar simultáneamente.

Debido a que la mayoría de distribuciones no siguen una distribución normal, se ha calculado el coeficiente de correlación utilizando el método de spearman. A continuación, se detallan las conclusiones del análisis:

- Podemos ver que los datos relacionados con los contagiados tienen una dependencia positiva entre ellos, si aumenta uno también tiende a aumentar el otro.
- Entre las variables de contaminación no parece haber mucha correlatividad entre ellos.
- Existe un índice de correlatividad negativo significativo entre el indicador de contaminación por NO2 y las variables de los contagiados.
- Existe un índice de correlatividad negativo entre el indicado de contaminación por partículas PM10 y las variables de los contagiados, aunque en menor medida que para el NO2.

A continuación procedemos a ver la matriz de correlación de forma visual:

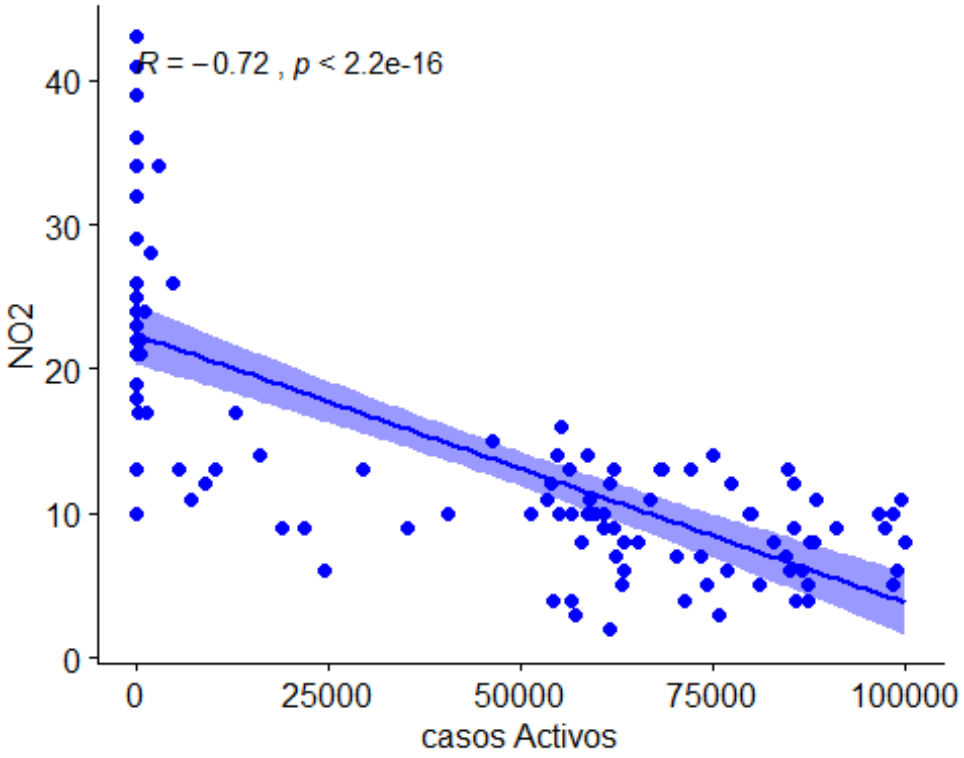
```
corrplot::corrplot(cor.res,method = "circle")
```



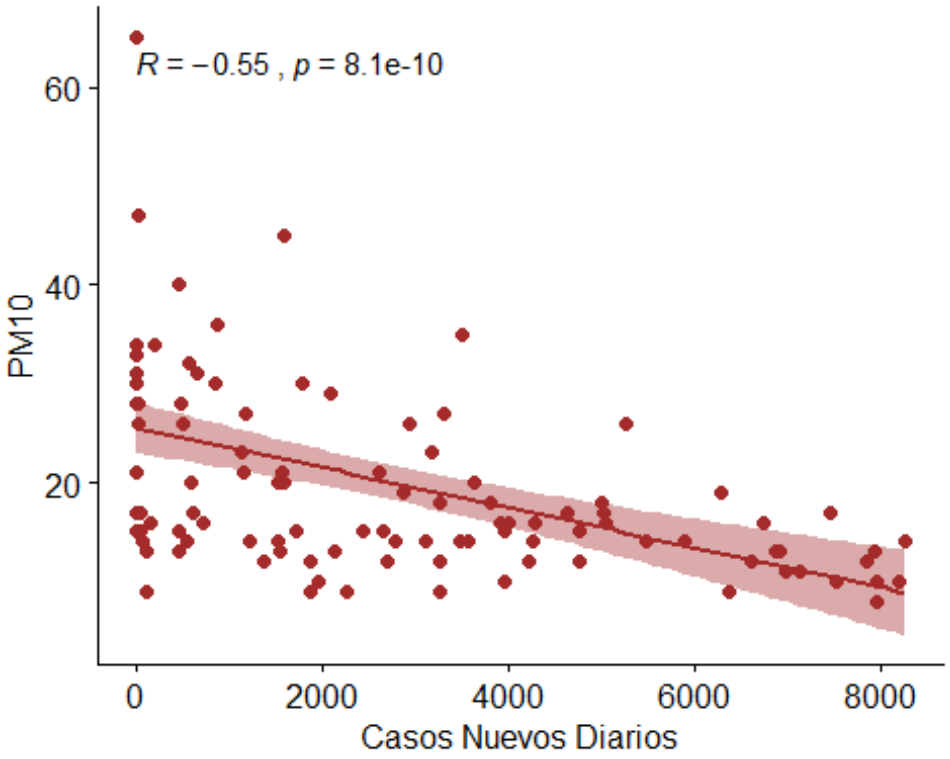
Utilizando el gráfico de correlación, podemos observar más fácilmente el nivel de correlatividad entre variables.

A continuación procedemos a ver el nivel de significancia de la correlación de los casos activos con el NO2 y de los casos nuevos diarios con PM10: Para ello, lo haremos mostrando gráficamente el resultado utilizando la función ggscatter de la librería ggpubr:

```
library("ggpubr")
ggscatter(ds,x = "Active.Cases", y = "NO2", color = "blue",
          add = "reg.line", conf.int = TRUE, cor.coef = TRUE, cor.method
= "spearman",
          xlab = "casos Activos", ylab = "NO2")
```




```
library("ggpubr")
ggscatter(ds,x = "Daily.New.Cases", y = "PM10", color = "brown",
          add = "reg.line", conf.int = TRUE, cor.coef = TRUE, cor.method
= "spearman",
          xlab = "Casos Nuevos Diarios", ylab = "PM10")
```



Como podemos observar, el nivel de correlación entre las variables es significativo (p-value < 0.05). También vemos gráficamente como cuando había 0 casos activos, se produjeron los valores más altos de NO2 en el dataset y a medida que los casos activos fueron aumentando el nivel de NO2 no subió de los 20 ug/m3. Esto es debido a que cuando se tenía un número importante de casos activos de COVID-19 se proclamó el estado de alarma en el estado español, limitando la circulación de personas y por ende se redujo la emisión de gases como el NO2.

Asimismo, también podemos ver que con el incremento de los casos nuevos de contagio diarios la reducción de partículas de 10 um de diámetro o menor por metro cúbico tiende a reducirse.

Modelo no supervisado: Kmeans

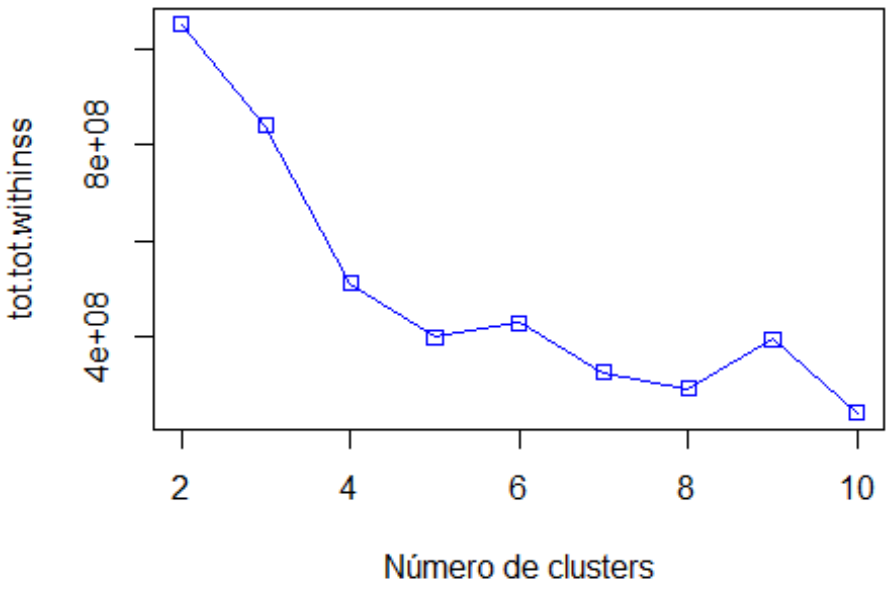
A continuación, intentaremos aplicar un modelo no supervisado, el k-means.

El algoritmo K-means permite agrupar en k clusters las diferentes observaciones del conjunto de datos en función de la media.

Para medir la distancia de la media entre las diferentes observaciones utilizaremos el método de Euclides y para determinar el número de k adecuado utilizaremos la regla de codo. Para ello, probaremos valores de k del 2 al 10 y verificaremos cual sería el que mejor resultado daría en función de aquel que ofrece la menor suma de los cuadrados de las distancias de los puntos de cada grupo con respecto a su centro (withinss).

Para el cálculo, se utilizará la función Kmeans de la librería “amap”.

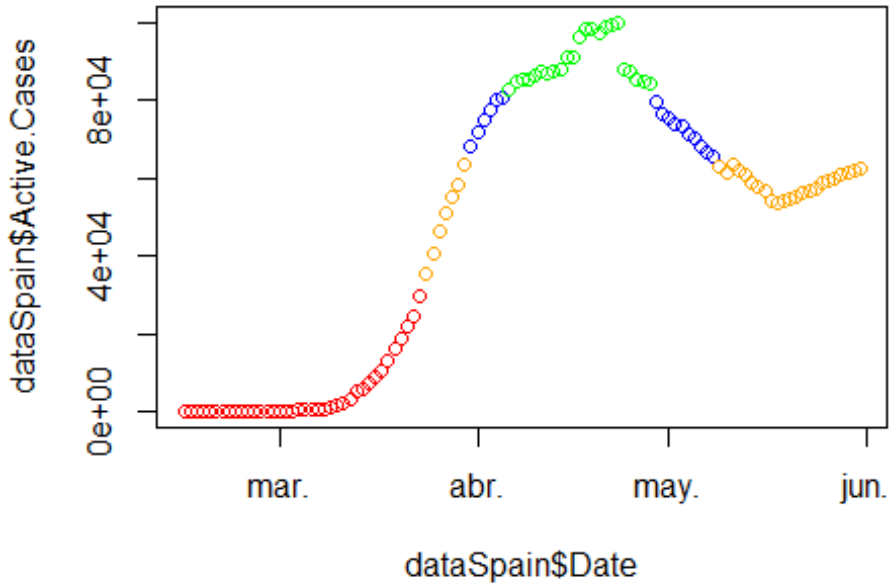
```
library(amap)
set.seed(8)

resultados <- rep(0, 10)
for (i in c(2:10))
{
  fit          <- Kmeans(ds, centers = i, method = "euclidean")
  resultados[i] <- sum(fit$withinss)
}
plot(2:10,resultados[2:10],type="o",col="blue",pch=0,xlab="Número de clus
ters",ylab="tot.tot.withinss")
```



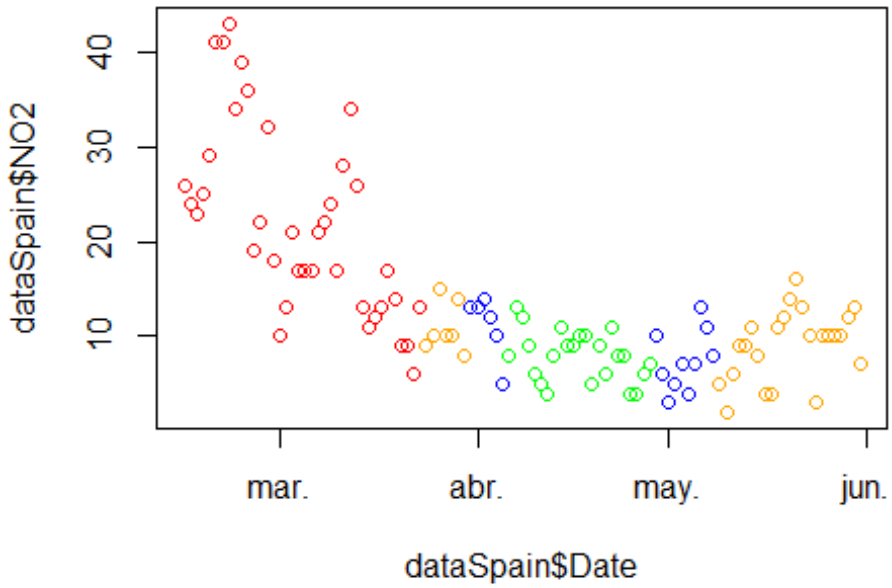
Observamos que, aproximadamente, utilizando la regla del codo, la curva se empieza estabilizar a partir del número 4. Se decide escoger este número cómo óptimo de k.

Procedemos a utilizar k = 4 para calcular los clústeres con el k-means.

```
set.seed(8)

K6Eu <- Kmeans(ds,centers = 4, method = "euclidean")

plot(dataSpain$Date, dataSpain$Active.Cases,col = c("red", "blue", "green
", "orange")[K6Eu$cluster])
```




```r
plot(dataSpain$Date, dataSpain$NO2,col = c("red", "blue", "green", "orang
e")[K6Eu$cluster])
```



Después de calcular los grupos, los mostramos de forma gráfica identificándolos por colores.

Vemos que el primer grupo (rojo) viene determinado por los días en que la pandemia estaba en sus inicios, con pocos contagios pero en aumento y con un índice alto de contaminación por NO2. El segundo grupo esta formado por aquellos días en que los contagios incrementaban en gran medida de un día para otro y por los últimos días recogidos en el dataset, en el que la curva ya había pasado el pico y comenzaba la "normalidad" en las ciudades españolas, como Madrid. Asimismo el indicador de NO2 comenzaba a subir de nuevo El tercer grupo esta formado por los días en que la curva estaba llegando a su pico y cuando justo la había pasado. Los niveles de NO2 en ese punto ya eran bajos. Finalmente, el cuarto grupo está formado por los días en que los casos activos en España estaban su punto más álgido y el nivel de NO2 era bastante bajo en Madrid.

Mediante esta agrupación podemos ver también como los niveles altos de NO2 se agrupan con los pocos casos activos de COVID-19 y viceversa.

Asimismo, parece que el principal criterio del algoritmo para separar los grupos fue la dimensión de casos activos.

Modelo de regresión linear

Seguimos con el análisis de los datos mediante la creación de un modelo de regresión linear que nos permita determinar el nivel de contaminación por NO2 durante los días de la pandemia en función del resto de datos obtenidos que más están correlacionados en nivel de absoluto con esta variable.

Para ello, utilizaremos la función lm() de la librería stats de R.

Obtenemos el modelo de regresión linear del nivel de contaminación de NO2 a partir de los casos activos de COVID-19.

```
linear.model1 <- lm(NO2 ~ Active.Cases, data = ds)
summary(linear.model1)
```

```
##
## Call:
## lm(formula = NO2 ~ Active.Cases, data = ds)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -12.3924  -3.7116  -0.5818   3.3047  20.5924
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2.241e+01  1.033e+00   21.68   <2e-16 ***
## Active.Cases -1.860e-04  1.743e-05  -10.67   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 6.295 on 105 degrees of freedom
## Multiple R-squared:  0.5202, Adjusted R-squared:  0.5157
## F-statistic: 113.9 on 1 and 105 DF,  p-value: < 2.2e-16
```

Con este modelo, se obtiene un coeficiente de R-squared de 0.52 por lo que podemos decir la calidad del modelo es media. Adicionalmente, observamos que el coeficiente de la variable de casos activos es negativa, lo que indica que a medida que aumente este valor, el valor de NO2 disminuye. Por último cabe indicar que el p-value es menor al nivel de significancia 0.05.

Procedimos a añadir la variable O3 al modelo:

```
linear.model2 <- lm(NO2 ~ Active.Cases + O3, data = ds)
summary(linear.model2)
```

```
##
## Call:
## lm(formula = NO2 ~ Active.Cases + O3, data = ds)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -12.4665  -4.2102  -0.4762   3.1187  19.5736
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2.887e+01  2.700e+00  10.691  < 2e-16 ***
## Active.Cases -1.654e-04  1.878e-05  -8.809 3.05e-14 ***
## O3           -2.366e-01  9.176e-02  -2.578   0.0113 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.132 on 104 degrees of freedom
## Multiple R-squared:  0.5491, Adjusted R-squared:  0.5404
## F-statistic: 63.32 on 2 and 104 DF,  p-value: < 2.2e-16
```

Después de añadir la variable O3, observamos que el coeficiente R-squared ajustado aumenta ligeramente a 0.54 por lo que aumenta la calidad del modelo. También para la variable O3, en caso de que aumente su valor hace disminuir el valor del indicador de NO2.

Procedemos a añadir la tercera variable que está más correlacionada con la variable NO2, el indicador de S02:

```
linear.model3 <- lm(NO2 ~ Active.Cases + O3 + SO2, data = ds)
summary(linear.model3)
```

```
##
## Call:
## lm(formula = NO2 ~ Active.Cases + O3 + SO2, data = ds)
##
```

```
## Residuals:
##      Min       1Q   Median       3Q      Max
## -11.3788  -3.9461   0.5997   3.2823  14.2859
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)   2.184e+01  2.800e+00   7.800 5.26e-12 ***
## Active.Cases -1.395e-04  1.766e-05  -7.898 3.22e-12 ***
## O3           -2.702e-01  8.282e-02  -3.262   0.0015 **
## SO2           3.389e+00  6.714e-01   5.048 1.93e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 5.517 on 103 degrees of freedom
## Multiple R-squared:  0.6385, Adjusted R-squared:  0.628
## F-statistic: 60.64 on 3 and 103 DF,  p-value: < 2.2e-16
```

Con este modelo, la calidad del modelo ha aumentado notablemente respecto a los modelos anteriores con un coeficiente R-squared ajustado de 0.62. Para el caso del indicador de SO2, si este aumenta en una unidad el indicado de NO2 aumenta en 3,39. Asimismo, todas las variables son significativas en el modelo con un p-value inferior a 0.05.

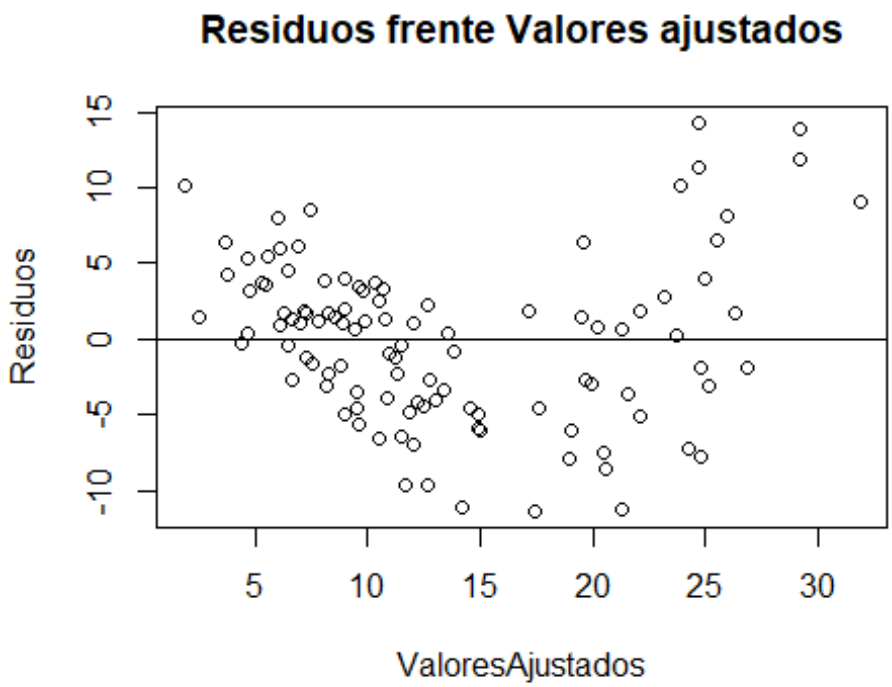
Procedemos a realizar una diagnosis del modelo para verificar su índice de acierto gráficamente:

Diagnosis del modelo

A continuación procederemos para hacer un diagnosis mediante gráficos del modelo que relaciona el valor de NO2 con los casos activos de COVID-19, el O3 y el SO2.

Primero haremos uno con los valores uno con los valores ajustados frente a los residuos (que nos permitirá ver si la varianza es constante).

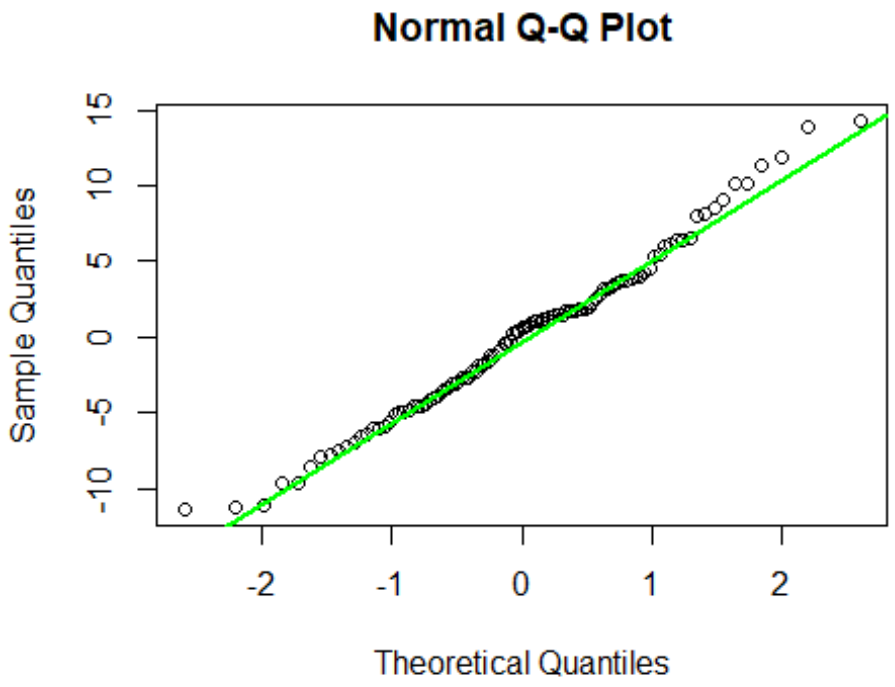
```
#Predecimos los valores
ValoresAjustados <- predict(linear.model3,ds)

#Eliminamos las etiquetas
names(ValoresAjustados)<-NULL

#Calculamos los residuos
Residuos <-  ds$NO2 - ValoresAjustados



plot(ValoresAjustados,Residuos,title("Residuos frente Valores ajustados")
)
abline(h = 0)
```



Por lo que vemos en el gráfico, la mayoría de los valores se ajustan con un error entre +5 y -5 del indicado de NO2.

A continuación, realizaremos un gráfico cuantil-cuantil para comparar los residuos del modelo con los valores de una variable que se distribuye normalmente(QQ plot).

```
qqnorm(Residuos)
qqline(Residuos, col = "green", lwd = 2)
```



En este gráfico vemos la distribución de los residuos. Para tener una distribución normal los valores deberían de estar en sintonía con la recta de color verde marcada en el gráfico. Podemos ver como la gran mayoría de los valores están en línea a excepción de unos pocos valores con sobrepasan los +10 y -10 unidades de error. A excepción de estos valores, podría decirse que los residuos siguen una distribución normal en la que la varianza es constante.

5. Resolución del problema - Conclusiones

Utilizando el dataset con los datos de COVID-19 de 8 países y con los datos de contaminación atmosférica de sus correspondientes capitales, se pretendía verificar si la contaminación en las ciudades más pobladas disminuyó con el inicio de la pandemia. Se decidió seleccionar los datos de uno de los países, España para reducir el alcance y establecer los pasos a seguir para el posterior análisis del resto de países.

Primero de todo, se actualizaron los datos del país elegido, ampliando los datos a analizar. Asimismo, se descartaron las variables que no serían significantes para el análisis.

Seguidamente, se realizó un análisis variable por variable, modificando el tipo de datos que fuera necesario e identificando los valores extremos y los ouliers. Para cada valor extremo identificado utilizando la función boxplot, se analizó y buscó las posibles causas y tomó la decisión de eliminar (considerar como dato perdido) o mantener.

Para los valores perdidos identificados, se utilizó el algoritmo K-Nearest-Neighbors para realizar la imputación de los valores y finalizar la limpieza de los datos.

Antes de iniciar el análisis, se realizaron las comprobaciones previas de normalidad y homocedasticidad para los datos. Utilizando el test de shapiro-wilk se obtuvo que ninguna de las variables seguía la distribución normal y utilizando el test de Fligner-Killeen se obtuvieron p-valores menores al nivel de significancia 0.05, por lo que se asumió igualdad de varianza entre las variables.

Después de aplicar los test, se realizó un análisis de la correlatividad entre las variables, del que se pudo observar gráficamente que a medida que los casos activos de COVID-19 aumentaban, el nivel de contaminación de NO2 y de PM10 se reducía.

Seguidamente, se aplicó el algoritmo k-means, un modelo no supervisado, entre los que se agruparon los valores bajos de casos activos con los valores altos de contaminación por NO2, y los días con más casos activos con los días de menor contaminación por N02.

Por último, se intentó crear un modelo de regresión linear para predecir el nivel de contaminación de NO2 en función de los parámetros que estaban más correlacionados con la variable, entre los que se encontraban los casos activos.

Por lo tanto, podemos concluir que el incremento de casos activos de COVID-19 contribuyó en la reducción de diversas sustancias contaminantes como el N02 y las PM10. Esto es debido a las medidas de restricción de movilidad ciudadana instaurada por el Gobierno Español para frenar el avance del virus.

Contribuciones	Firma
Investigación Previa	N.B.A.
Redacción de las respuestas.	N.B.A.
Desarrollo código.	N.B.A.