Práctica 2: LIMPIEZA Y VALIDACIÓN DE LOS DATOS

Alberto Lucas Navío

1. Detalles de la actividad:

Esta actividad consiste un llevar a cabo un caso práctico de tratamiento de datos. Como parte de este tratamiento se incluye, la limpieza del conjunto de datos, seleccionar los datos relevantes para llevar a cabo un análisis, así como utilizar de forma eficiente las herramientas de integración, limpieza, validación y análisis de los datos.

2. Resolución:

2.1. Descripción del dataset:

El dataset elegido para llevar a cabo el proyecto se denomina *"Countries of the World"* y se puede encontrar en Kaggle a través de este enlace: https://www.kaggle.com/fernandol/countries-of-the-world. Este conjunto de datos contiene indicadores económicos como tasa de migración, producto interior bruto per cápita, etc., sobre prácticamente todos los países del mundo. A continuación, describiremos el conjunto de variables que forman este dataset.

Este conjunto de datos se ha enriquecido con datos adicionales extraídos de otro dataset, denominado *"Country_profile_variables".* Este conjunto de datos se puede encontrar también en Kaggle, concretamente: https://www.kaggle.com/sudalairajkumar/undata-country-profiles. Concretamente, hemos escogido una variable de este dataset, que es la tasa de desempleo de cada país, y lo hemos agregado al conjunto datos anterior.

- Country: nombre del país.
- Region: region geográfica en la que se encuentra el país en cuestión
- Population: número de habitantes del país.
- Area (sq. mi.): superficie del país en millas cuadradas.
- Pop. Density (per sq.mi.): densidad de población del país.
- Coastline (coast/area ratio): el porcentaje de línea de costa del país con relación al área total del mismo.
- Net migration: la tasa de migración del país. Será un valor positivo si el país en cuestión tiene más inmigrantes que emigrantes. Aquellas observaciones con un valor negativo para esta variable, experimentan más emigración que inmigración.
- Infant mortality (per 1000 births): número de muertes por cada 1000 niños nacidos en un año.
- GDP_MILLION: product interior bruto del país.
- GDP($ per capita): producto interior bruto del país dividido por el número de habitantes.
- Literacy (%): porcentaje de personas del país que saben leer y escribir.
- Phones (per 1000): número total de teléfonos móviles en uso por cada 1000 habitantes.
- Arable (%): porcentaje del territorio que puede ser utilizado para el cultivo.

- Crops (%): índice que mide la producción agrícola del país.
- Other (%)
- Climate: tipo de clima del país en cuestión
- Birthrate: número de nacimientos por 1000 habitantes en el país.
- Deathrate: el ratio de muertes por 1000 habitantes en el país.
- Agriculture: porcentaje del producto interior bruto generado por el sector primario.
- Industry: porcentaje del producto interior bruto generado por el sector secundario.
- Service: procentaje del product interior bruto generado por el sector terciario.
- Unemployment: porcentaje de población activa que no tiene trabajo.

Partimos pues de un conjunto de datos con 227 observaciones y 22 atributos para cada observación.

2.2. Importancia y objetivos de los análisis:

A partir de este conjunto de datos, vamos a intentar extraer conclusiones acerca de qué factores económicos de un país tienen más influencia sobre fenómenos como la migración. Así mismo, vamos a comprobar si existe una relación estadística entre el producto interior bruto de un país y su nivel de alfabetización o la esperanza de vida de un país.

Este tipo de análisis tienen gran relevancia en una gran diversidad de ámbitos. A nivel profesional se podría asumir que historiadores, geógrafos y cualquier persona que trabaje en el ámbito de las Ciencias Sociales serían los principales receptores de un análisis de este tipo. Sin embargo, este análisis también puede ser de interés para cualquier persona interesada en entender mejor el mundo en el que vive, independientemente de su profesión.

2.3. Limpieza de datos

El primer paso del proyecto será importar el conjunto de datos con el que vamos a trabajar. Éste se encuentra en un fichero CS, que importamos a la interfaz de R en formato data.frame, mediante el comando read.csv.

```
> #Importamos los datos
> country_data = read.csv("initial_dataset.csv", header=TRUE, dec=",")
> head(country_data)
        Country                               Region Population Area..sq..mi.. Pop..Density..per.sq..mi.. Coastline..coast.area.ratio. Net.migration Infant.mortality..per.1000.births.
1   Afghanistan        ASIA (EX. NEAR EAST)            31056997        647500                    48.0                        0.00        23.06                            163.07
2       Albania   EASTERN EUROPE                     3581655         28748                   124.6                        1.26        -4.93                             21.52
3       Algeria   NORTHERN AFRICA                   32930091       2381740                    13.8                        0.04        -0.39                             31.00
4 American Samoa  OCEANIA                              57794           199                   290.4                       58.29       -20.71                              9.27
5       Andorra   WESTERN EUROPE                       71201           468                   152.1                        0.00         6.60                              4.05
6        Angola   SUB-SAHARAN AFRICA                12127071       1246700                     9.7                        0.13         0.00                            191.19
  GDP_MILLION_. GDP....per.capita. Literacy.... Phones..per.1000. Arable.... Crops.... Other.... Climate Birthrate Deathrate Agriculture Industry Service Unemployment
1             0               700         36.0              3.2       12.13      0.22      87.65       1     46.60     20.34       0.380    0.240   0.380           35
2          4100              4500         86.5             71.2       21.09      4.42      74.49       3     15.11      5.22       0.232    0.188   0.579           14
3        108700              6000         70.0             78.1        3.22      0.25      96.53       1     17.14      4.61       0.101    0.600   0.298         11,7
4           #N/A              8000         97.0            259.5       10.00     15.00      75.00       2     22.46      3.27          NA       NA      NA         29,8
5          1000             19000        100.0            497.2        2.22      0.00      97.78       3      8.71      6.25          NA       NA      NA          3,7
6           #N/A              1900         42.0              7.8        2.41      0.24      97.35      NA     45.11     24.20       0.096    0.658   0.246          #N/A
>
```

Observamos qué tipo asigna R a cada variable. Como se puede ver disponemos de varias variables numéricas, así como algunas variables a las que R les ha asignado el tipo factor.

```
> ## Tipo de dato asignado a las variables del dataframe
> sapply(country_data, function(x) class(x))
                            Country                              Region
                           "factor"                            "factor"
                        Population                        Area..sq..mi..
                           "integer"                           "integer"
        Pop..Density..per.sq..mi..       Coastline..coast.area.ratio.
                           "numeric"                           "numeric"
                      Net.migration Infant.mortality..per.1000.births.
                           "numeric"                           "numeric"
                      GDP_MILLION_.                   GDP....per.capita.
                           "integer"                           "integer"
                       Literacy....                    Phones..per.1000.
                           "numeric"                           "numeric"
                         Arable....                            Crops....
                           "numeric"                           "numeric"
                         Other....                             Climate
                           "numeric"                           "numeric"
                         Birthrate                           Deathrate
                           "numeric"                           "numeric"
                       Agriculture                            Industry
                           "numeric"                           "numeric"
                           Service                        Unemployment
                           "numeric"                           "numeric"
>
```

2.3.1. Selección de datos de interés:

Vamos a iniciar el tratamiento de los datos, seleccionando los atributos relevantes para nuestro análisis. Para el análisis que vamos a realizar podemos prescindir de varios atributos que están más relacionados con factores geográficos de un país que con indicadores estrictamente económicos. Así pues, prescindiremos de los siguientes atributos*: pop. Density, coastline, phones (per 1000), arable, crops* and *climate*.

```
> country_data <- country_data[, -(5:6)]

> country_data <- country_data[, -(10:14)]
```

El resultado por tanto, es un conjunto de datos con 227 observaciones y 15 atributos. Una vez seleccionados los atributos, es posible tratar los valores ausentes del dataset.

2.3.2. Tratamiento de nulos y ceros:

El siguiente paso del proyecto será el tratamiento de nulos y ceros de nuestro conjunto de datos. Antes de nada, es necesario comprobar el número de valores nulos de cada variable. Así mismo, hay que considerar si es posible que una variable contenga el valor 0, o si este valor no forma parte del dominio del atributo. En este caso habría que tratar este 0 como un valor nulo.

Comprobamos pues, el número de nulos que hay por cada variable, como se puede ver a continuación:

```
> sapply (country_data, function(x) sum(is.na(x)))
                         Country                           Region
                               0                                0
                      Population                    Area..sq..mi..
                               0                                0
                   Net.migration Infant.mortality..per.1000.births.
                               0                                3
                   GDP_MILLION_.                GDP....per.capita.
                             142                                0
                    Literacy....                        Birthrate
                              18                                3
                      Deathrate                      Agriculture
                               4                               15
                        Industry                          Service
                              16                               15
                    Unemployment
                              37
```

Una vez comprobado el número de valores nulos, hay que decidir qué estrategia es la más adecuada para tratar con estos. Se podrían las observaciones con valores nulos. Sin embargo, este dataset sólo contiene 227 observaciones. Por tanto, eliminar las observaciones con valores nulos supondría una pérdida de información importante.

Por otra parte, hay variables con gran cantidad de valores nulos y otras con muy pocos. Por lo tanto, utilizaremos una estrategia diferente para tratar los valores nulos de cada variable. Así pues, vamos analizando atributo por atributo:

- GDP_MILLION_$:

  Para este atributo, no disponemos del valor de 142 observaciones. Además, en 35 observaciones el valor es 0. Sin embargo, 0 no puede formar parte del dominio de este atributo. Por tanto, podemos decir que tenemos 172 valores nulos para esta variable. Es decir, más del 75% de observaciones son valores nulos.

  Como hay tantos valores ausentes, no sería muy efectivo aplicar técnicas de imputación de valores como la imputación basada en K vecinos. Sin embargo, podemos calcular este atributo fácilmente a partir de otras variables de nuestro conjunto de datos. Concretamente, vamos a multiplicar el producto interior bruto per cápita de un país por su número de habitantes para obtener el producto interior bruto del país.

  Así pues, añadimos una nueva columna a nuestro *dataframe*, que será igual al producto de las columnas GDP ($ per capita) y population. Una vez hecho esto, eliminamos el atributo GDP_MILLION_$:

```
> country_data$GDP <- as.integer((country_data$GDP....per.capita. * country_data$Population)/ 1000000)
>
```

```
>
> country_data <- country_data[,-(7)]
>
```

- GDP ($ per capita):

  Sólo 1 valor ausente. Como es una única observación, no supone un esfuerzo excesivo hallar el valor del este atributo. Por tanto, podemos buscar el valor de la misma e incluirlo en nuestro dataset. Esta forma de tratar los valores nulos nos permite completar nuestro conjunto de datos de la forma más precisa, lo cual nos permitirá realizar análisis más precisos.

  Además, es sencillo encontrar la información que necesitamos y añadir el valor a la observación correspondiente.

```
> country_data$Country[224]
[1] Western Sahara
227 Levels: Afghanistan Albania Algeria American Samoa Andorra Angola ... Zimbabwe
>
> country_data$GDP....per.capita.[224]
[1] NA
>
> country_data$GDP....per.capita.[224]<- 2500
> country_data$GDP....per.capita.[224]
[1] 2500
```

- Literacy: 18 observaciones sin este valor.

  En este caso vamos a utilizar el método de kNN, del paquete *VIM,* un método de imputación de valores basado en la similitud entre las observaciones. Es decir, utilizamos una técnica para predecir el valor del atributo de una observación, a partir del valor de las observaciones más parecidas.

```
> suppressWarnings (suppressMessages(library(VIM)))
```

  Con esto asignamos valores a las observaciones que no tienen valor en nuestro conjunto de datos.

```
> country_data$Literacy_rate <- kNN(country_data)$Literacy_rate
```

- Birthrate: 3 observaciones sin este valor.

  De nuevo, como sólo hay 3 observaciones sin valor, es razonable buscar estos valores y añadirlos al conjunto de datos.

```
> country_data$Birthrate[182]
[1] NA
>
> country_data$Birthrate[182]<- 9.2
> country_data$Birthrate[182]
[1] 9.2
>
```

```
>
> country_data$Birthrate[222]<- 5.5
> country_data$Birthrate[224]<- 28.9
>
```

- Deathrate: 4 observaciones sin este valor.

En este caso hay 4 valores nulos. Aplicamos de nuevo la técnica anterior de encontrar la información que necesitamos y aplicarla a nuestro conjunto de datos.

```
> country_data$Deathrate[48]
[1] NA
>
> country_data$Deathrate[48]<- 8.6
> country_data$Deathrate[48]
[1] 8.6
>
```

```
> country_data$Deathrate[48]
[1] NA
>
> country_data$Deathrate[48]<- 8.6
> country_data$Deathrate[48]
[1] 8.6
>
> country_data$Deathrate[222]<- 5.5
> country_data$Deathrate[224]<- 11.49
>
```

```
> country_data$Deathrate[182]
[1] NA
>
> country_data$Deathrate[182]<- 13.2
> country_data$Deathrate[182]
[1] 13.2
>
```

- Agriculture, industry, service: 15 observaciones sin valor.

  Estos tres atributos van a ser reemplazados por un único atributo que describa simplemente cuál es el sector principal en la economía del país. Para nuestro análisis no es necesario disponer del porcentaje de cada sector en la economía de cada país. Así pues, primero se reemplazarán los atributos y luego se insertarán los valores de forma manual.

- Unemployment: 37 observaciones sin valor

  Como se ha hecho anteriormente, reemplazamos los valores nulos utilizando Knn.

```
> country_data$Unemployment_rate <- kNN(country_data)$Unemployment_rate
```

- Net_migration: 3 observaciones sin valor.

  De nuevo para este atributo, disponemos sólo de 3 observaciones sin valor. Cabe destacar que hay valores 0, pero en este caso, el valor 0 sí que forma parte del dominio del atributo. Como hemos hecho anteriormente, vamos a encontrar la información necesaria e introducir los valores para la observación correspondiente.

```
> country_data$Net.migration[48]
[1] NA
>
> country_data$Net.migration[48] <- -2.2
> country_data$Net.migration[48]
[1] -2.2
>
```

```
> country_data$Net.migration[222]
[1] NA
>
> country_data$Net.migration[222]<- -4.6
> country_data$Net.migration[222]
[1] -4.6
>
```

```
> country_data$Net.migration[224]
[1] NA
>
> country_data$Net.migration[224]<- 5.4
> country_data$Net.migration[224]
[1] 5.4
>
```

- Infant_mortality: 3 observaciones sin valor.

De nuevo, sólo 3 observaciones con valores nulos. Como en los casos anteriores buscamos la información necesaria y la introducimos en el dataframe.

```
> country_data$Infant.mortality[48]<- 12.6
> country_data$Infant.mortality[222]<- 4.3
> country_data$Infant.mortality[224]<- 50.5
>
```

2.3.3. Reducción de dimensionalidad

Una vez hemos tratado los valores ausentes, vamos a llevar a cabo un ejercicio de reducción de dimensionalidad. Disponemos de tres atributos que indican el porcentaje de importancia que tiene cada sector económico.

Vamos a crear un nuevo atributo que indique el cuál es el sector más importante en la economía de un país, que va a reemplazar las tres variables que indican el peso de cada sector económico.

```
> country_data$Main_sector <- ifelse ((country_data$Agriculture > country_data$Industry) & (country_data$Agriculture > country_data$Service), "Primario",
+ ifelse((country_data$Industry > country_data$Agriculture) & (country_data$Industry > country_data$Service), "Secundario",
+ ifelse ((country_data$Service > country_data$Agriculture) & (country_data$Service > country_data$Industry), "Terciario", "Desconocido")))
>
```

Una vez hecho esto, hay varias observaciones para las que no tenemos valor. Sin embargo, es posible encontrar cuál es el sector más importante en la economía de cada país y completar nuestro conjunto de datos.

```
>
> country_data$Main_sector[4] <- "Primario"
> country_data$Main_sector[5] <- "Terciario"
> country_data$Main_sector[79] <- "Terciario"
> country_data$Main_sector[81] <- "Secundario"
> country_data$Main_sector[84] <- "Terciario"
> country_data$Main_sector[135] <- "Primario"
> country_data$Main_sector[139] <- "Terciario"
> country_data$Main_sector[141] <- "Terciario"
> country_data$Main_sector[145] <- "Terciario"
> country_data$Main_sector[154] <- "Secundario"
> country_data$Main_sector[172] <- "Primario"
> country_data$Main_sector[175] <- "Terciario"
> country_data$Main_sector[178] <- "Terciario"
> country_data$Main_sector[209] <- "Terciario"
> country_data$Main_sector[222] <- "Primario"
> country_data$Main_sector[224] <- "Primario"
>
```

Una vez hecho esto, podemos eliminar los atributos “agriculture, industry, service” ya que una vez creado el atributo “Main_sector”, esta información resulta redundante para nuestro análisis.

```
>
> country_data <- country_data[,-(10:12)]
```

Eliminamos también el atributo “Infant.Mortality.per1000births”. Como disponemos de un atributo denominado “Infant.Mortality” que mide el mismo factor, es redundante conservar las dos variables. Eliminamos por lo tanto una de ellas, como se ve a continuación.

```
>
> country_data <- country_data[,-(6)]
```

- **Reemplazar los nombres de las columnas**

  Una última operación que vamos a realizar antes de finalizar la fase de limpieza de los datos es la de cambiar el nombre de las columnas, para facilitar el uso de las variables durante la fase de análisis.

```
>
> colnames(country_data) <- c("Country", "Region", "Population", "Area", "Net_migration", "GDP_per_capita", "Literacy_rate", "Birthrate",
+ "Deathrate", "Unemployment_rate", "Infant_mortality", "GDP", "Main_sector")
>
```

Una vez hemos tratado los valores nulos de cada atributo, comprobamos que efectivamente nuestro conjunto de datos ya no tiene valores nulos.

```
> sapply(country_data, function(x) sum(is.na(x)))
          Country            Region        Population              Area      Net_migration
                0                 0                 0                 0                 0
   GDP_per_capita     Literacy_rate         Birthrate         Deathrate Unemployment_rate
                0                 0                 0                 0                 0
 Infant_mortality               GDP       Main_sector
                0                 0                 0
> |
```

Una vez completada esta operación, procedemos a tratar los valores extremos.

2.3.4. Valores extremos

Los valores extremos son aquellos valores que se encuentran muy alejados de la distribución normal de una variable. Estos valores pueden aparecer por distintos motivos y pueden ser indicadores de que hay algún error en los datos que puede afectar al resultado de los análisis.

Vamos a aplicar la función *boxplot.stats()* de R, en aquellas variables en que los valores extremos puedan ser indicadores. Es decir, obteniendo los valores extremos para la variable población se pueden extraer muchas conclusiones, ya que estamos comparando países con poblaciones muy diferentes. Sin embargo, con atributos como la tasa de migración, los valores extremos sí que pueden ser relevantes.

Vemos a continuación el cálculo de los valores extremos para nuestro conjunto de datos:

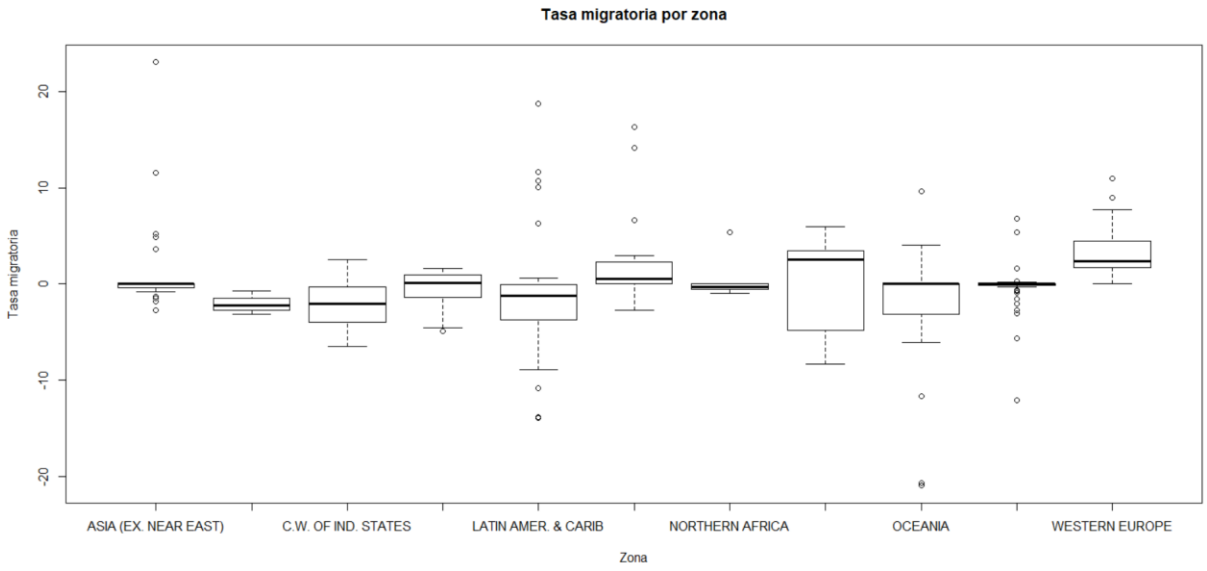
```
> boxplot.stats(country_data$Net_migration)$out
 [1]  23.06  -4.93 -20.71   6.60  10.76  -6.15  -6.47  -4.90  10.01  -4.58   5.96 -12.07  18.75 -13.87
[15]  -8.58   6.27  -4.70  -8.37 -13.92   5.24   4.99   5.36  -4.92   6.59  14.18   4.85   8.97   4.86
[29]  -6.04   6.78  -4.87 -20.99   7.75   4.05   9.61  16.29  -7.11  -4.86  -7.64 -11.70  10.98  -5.69
[43]  11.53   5.37  -8.81   4.05 -10.83  11.68  -8.94  -4.60   5.40
>
> boxplot.stats(country_data$GDP_per_capita)$out
[1] 55100 37800 37800
>
> boxplot.stats(country_data$Literacy_rate)$out
[1] 36.0 26.6 35.9 17.6 31.4 37.8
>
> boxplot.stats(country_data$Birthrate)$out
numeric(0)
>
> boxplot.stats(country_data$Deathrate)$out
 [1] 20.34 24.20 29.50 18.65 19.31 28.71 23.10 19.33 21.35 18.86 20.91 23.03 22.00 29.74 19.93 21.84
>
> boxplot.stats(country_data$Unemployment_rate)$out
 [1] 35.0 29.8 77.0 60.0 28.0 26.7 33.5 40.6 40.0 30.6 28.1 30.0 36.0 29.8 28.1 48.0 27.6 28.0 50.0 26.4
[21] 26.7 27.0 95.0
>
> boxplot.stats(country_data$Infant_mortality)$out
[1] 163.07 191.19 128.87 130.79 143.64
> |
```

Revisando estos datos, podemos ver que la mayoría de atributos contienen valores extremos que son perfectamente posibles. Sin embargo, el atributo "*Net_migration*" contiene una gran cantidad de valores extremos, por lo que vamos a estudiar con más detalle la distribución de datos de esta variable.

Para ello, vamos a representar los valores de la tasa migratoria de cada país dividida por regiones, ya que esto nos permitirá ver más fácilmente si hay algún valor erróneo.

```
>
> boxplot (country_data$Net_migration ~ country_data$Region, data = country_data, main = "Tasa migratoria por zona", xlab = "Zona", ylab="Tasa migratoria")
>
```

Así pues, representamos estos valores en un boxplot para cada región.



El primer valor que resulta llamativo es el país de Asia que tiene una tasa de migración positiva superior a 20. Inicialmente, se ha considerado que este dato es erróneo. Sin embargo, este dato se corresponde con Afganistán y es resultado de que muchos refugiados que emigraron por la guerra han regresado al país. Por tanto, este valor es correcto.

El resto de valores extremos se debe principalmente a países con unas circunstancias especiales como las Islas Vírgenes Británicas en Sudamérica, Singapur en Asia y otros territorios como colonias y países con tratados migratorios especiales.

No obstante, hay un valor que es erróneo es el de Western Sahara. Vemos como hay un valor extremo en la región de África del norte. Éste se corresponde con Western Sahara y se debe a que el valor introducido anteriormente para este país pertenece a un año distinto al del resto de países. Por tanto, nos aseguramos de averiguar el valor correcto y sustituimos el valor anterior por el dato correcto. No vamos a realizar ningún cambio sobre el resto de los datos.

- **Western Sahara es de -3.5**

Con esto concluye la fase de preparación de los datos. En este momento nuestro conjunto de datos está preparado para realizar los análisis correspondientes.

2.3.5. Exportación de los datos preprocesados

Una vez finalizada la fase de integración, validación y limpieza del conjunto de datos, procedemos a exportar un fichero con el conjunto de datos generado. El fichero se denominará "country_data.csv", y se generará mediante la función *write.csv().*

```
> write.csv(country_data, "country_data.csv")
>
```

2.4. Análisis de los datos

2.4.1. Selección de grupos de datos a analizar

```
> country_data.asia <- country_data[country_data$Region == "ASIA (EX. NEAR EAST)",]
> country_data.europa_este <- country_data[country_data$Region == "EASTERN EUROPE                     ",
> country_data.africa_norte <- country_data[country_data$Region == "NORTHERN AFRICA",]
> country_data.oceania <- country_data[country_data$Region == "OCEANIA",]
> country_data.europa <- country_data[country_data$Region == "WESTERN EUROPE",]
> country_data.africa <- country_data[country_data$Region == "SUB-SAHARAN AFRICA",]
> country_data.sudamerica <- country_data[country_data$Region == "LATIN AMER. & CARIB",]
> country_data.medio_oriente <- country_data[country_data$Region == "NEAR EAST",]
> country_data.norteamerica <- country_data[country_data$Region == "NORTHERN AMERICA",]
>
```

2.4.2. Comprobación de la normalidad y homogeneidad de la varianza

```
> library (nortest)
>
> alpha = 0.05
> col.names = colnames(country_data)
>
> for (i in 1:ncol(country_data)){
+    if (i == 1) cat("Variables que no siguen una distribución normal:\n")
+    if (is.integer (country_data[,i]) | is.numeric (country_data[,i])) {
+        p_val = ad.test(country_data[,i])$p.value
+        if(p_val < alpha) {
+             cat(col.names[i])
+
+
+             if (i < ncol(country_data) - 1) cat(",")
+             if (i %% 3 == 0) cat("\n")
+           }
+        }
+ }
Variables que no siguen una distribución normal:
Population,
Area,Net_migration,GDP_per_capita,
Literacy_rate,Birthrate,Deathrate,
Unemployment_rate,Infant_mortality,GDP
>
```

```
> fligner.test (GDP_per_capita ~ Net_migration, data = country_data)

        Fligner-Killeen test of homogeneity of variances

data:  GDP_per_capita by Net_migration
Fligner-Killeen:med chi-squared = 178.59, df = 158, p-value = 0.1254
```

```
> fligner.test (Birthrate ~ Deathrate, data = country_data)

        Fligner-Killeen test of homogeneity of variances

data:  Birthrate by Deathrate
Fligner-Killeen:med chi-squared = 225.88, df = 203, p-value = 0.1296
```

2.5. Pruebas estadísticas

2.5.1. Variables con mayor influencia sobre la tasa de mortalidad y sobre la tasa de migración

CALCULAR EL COEFICIENTE DE CORRELACIÓN DE CADA VARIABLE CON LA TASA DE MIGRACIÓN

```
> corr_matrix <- matrix(nc = 2, nr = 0)
> colnames(corr_matrix) <- c("estimate", "p-value")
```

```
>
>  # Calculamos el coeficiente de correlación para cada variable cuantitativa para el atributo "Net_migration"
> for (i in 1:(ncol(country_data)-1)) {
+      if (is.integer(country_data[,i]) | is.numeric(country_data[,i])) {
+           spearman_test = cor.test(country_data[,i], country_data[,5], method = "spearman")
+           corr_coef = spearman_test$estimate
+           p_val = spearman_test$p.value
+
+ # Añadimos las filas a la matriz
+
+ pair = matrix(ncol = 2, nrow = 1)
+ pair[1][1] = corr_coef
+ pair[2][1] = p_val
+ corr_matrix <- rbind(corr_matrix, pair)
+ rownames(corr_matrix)[nrow(corr_matrix)] <- colnames(country_data)[i]
+ }
+ }
```

	estimate	p-value
Population	-0.04040601	5.447375e-01
Area	-0.10466235	1.158315e-01
Net_migration	1.00000000	0.000000e+00
GDP_per_capita	0.37957537	3.421653e-09
Literacy_rate	0.16799696	1.123832e-02
Birthrate	-0.21022586	1.444739e-03
Deathrate	0.02899192	6.639347e-01
Unemployment_rate	-0.22856050	5.191911e-04
Infant_mortality	-0.34385174	1.066561e-07
GDP	0.11084859	9.570932e-02

2.5.2. Dar respuesta a las siguientes preguntas:

- **¿Hay una relación directa entre el producto interior bruto per cápita de un país y la tasa de migración?**
- **¿Es la tasa de mortalidad inversamente proporcional al producto interior bruto per capita?**
- **¿Se puede demostrar estadísticamente que el índice de alfabetización influye sobre la tasa de desempleo de un país?**

2.5.3. Modelo de regresión lineal

- **¿Se puede predecir la tasa de migración de un país a partir de sus indicadores económicos?**