PRÁCTICA

Actividad 1. Tratamiento de datos con R

- Módulo: Big Data Aplicado
- Unidad de trabajo: UNIDAD 1. Almacenamiento de datos
- Alumno:Alvaro Lucio-Villegas de Cea

Índice

Enunciado

Toma el archivo CSV adjunto y realiza las tareas de limpieza y transformación que veas oportunas, indicando el porqué de cada una de ellas. Por ejemplo.

- Con tal script detectó tal problema
- He ejecutado tal script para eliminar/arreglar/modificar/incluir tal columna.

Realiza una investigación sobre la importancia de las diferentes variables.

- Compara los resultados ofrecidos por dos algoritmos diferentes que permiten seleccionar características.
- ¿Podríamos identificar en qué franja de edad se produjeron más contagios?¿y más muertes?. Dicho de otra forma, ¿Qué variable/característica es más importante para el número de infectados y de muertos ?

Añade alguna representación gráfica de la totalidad o parte de los datos que según tu opinión, ofrezca una manera clara y rápida de obtener información a partir de los mismos. Justifica tu respuesta.

Ejercicio

Extracción y Limpieza

Tratamiento de los datos y limpieza.

```r
library(dplyr)
library(corrplot)
library(ggplot2)

#Carga de Datos
df <- read.csv("COVID-19-2.csv",header = TRUE, sep=";")

#Eliminación de Columnas
df <- df %>% select(-Eli1,-Eli2,-Eli3,-Eli4,-Eli5,-Eli6,-Eli7.)

#Remplazo de valores erróneos
df <- df %>% mutate_all(~ifelse(. == "...", 0, .))

#Conversión de valores nulos a 0
df$NRecuperados[is.na(df$NRecuperados)] <- 0

#Transformación de una cadena de String para su futura conversión a Int
df$Proporcion <- gsub(",", ".", as.character(df$Proporcion))

#División de DF sin fecha para el análisis de correlación.
dfSinFecha <- df %>% select(-Fecha)

#Dejamos el Df Principal con solo la columna Fecha
Fecha<-df$Fecha

#Convertimos las columnas string a numéricas
dfSinFecha <- dfSinFecha %>% mutate_if(is.character, as.numeric)

#Volvemos a rellenar los valores NA a 0
dfSinFecha[is.na(dfSinFecha)]<-0


#Concatenación de los dos DataFrames para tener uno completo ya limpio
df <- cbind(Fecha, dfSinFecha)
```

Análisis de Correlación

Tipos de Métodos

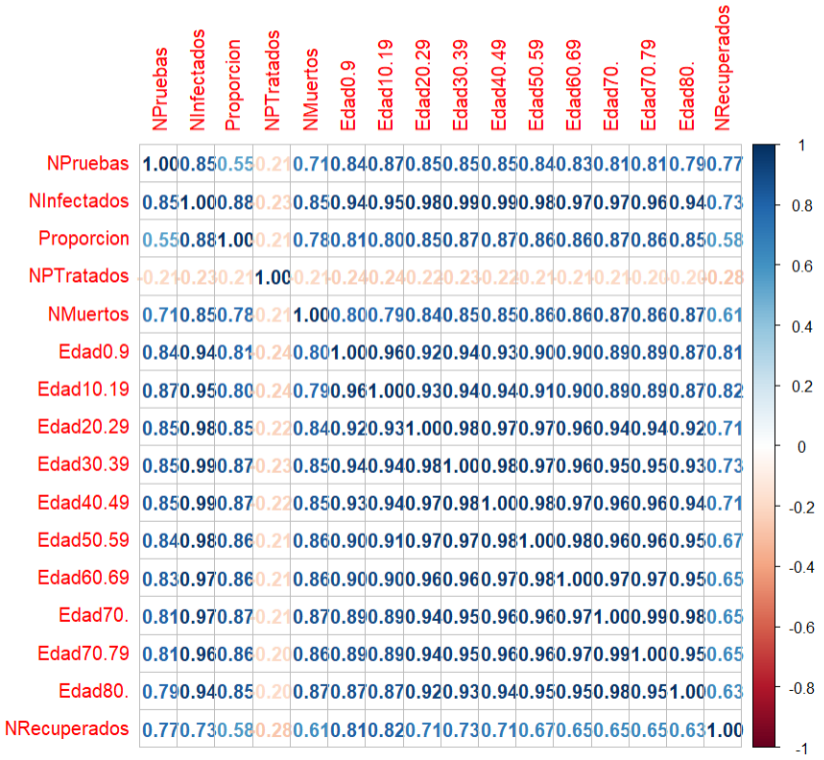
Nombre	Descripción
pearson	Este es el método predeterminado en cor() y calcula la correlación de Pearson. Esta metodología asume una distribución normal y linealidad en las variables. La correlación de Pearson es un valor comprendido entre -1 y 1 que indica el grado de relación lineal entre las variables. Un valor de 1 indica una relación lineal perfectamente positiva, un valor de -1 indica una relación lineal perfectamente negativa y un valor de 0 indica la ausencia de relación lineal.
spearman	Este método calcula la correlación de Spearman. Esta metodología es no paramétrica y no requiere la distribución normal de las variables. La correlación de Spearman es un valor comprendido entre -1 y 1 que indica el grado de relación monotónica entre las variables.
kendall	Este método calcula la correlación de Kendall. Al igual que la correlación de Spearman, esta metodología es no paramétrica y no requiere la distribución normal de las variables. La correlación de Kendall es un valor comprendido entre -1 y 1 que indica el grado de relación monotónica entre las variables.
cov	Este método calcula la covarianza entre las variables. La covarianza es una medida de la relación lineal entre las variables, pero no está normalizada y por lo tanto no está limitada a valores comprendidos entre -1 y 1.

Código:

```
#Tratamiento para obtener el p value de correlación en el dataSet
dfSinTra<- dfSinFecha %>% select(-NPTratados)
chisq.test(dfSinTra)



#Generamos una matriz de correlación para poder visualizarla en un grafo.
#Se pueden usar 3 métodos de correlación el spearman, pearson, kendall y cov.
corr_matrix <- cor(dfSinFecha, method = "spearman")

#Lo mostramos en un grafo de correlación.
corrplot(corr_matrix, method = "number")
```

Resultado:

```
        Pearson's Chi-squared test

data:  dfSinTra
X-squared = 8653189, df = 9856, p-value < 2.2e-16
```



Conclusiones

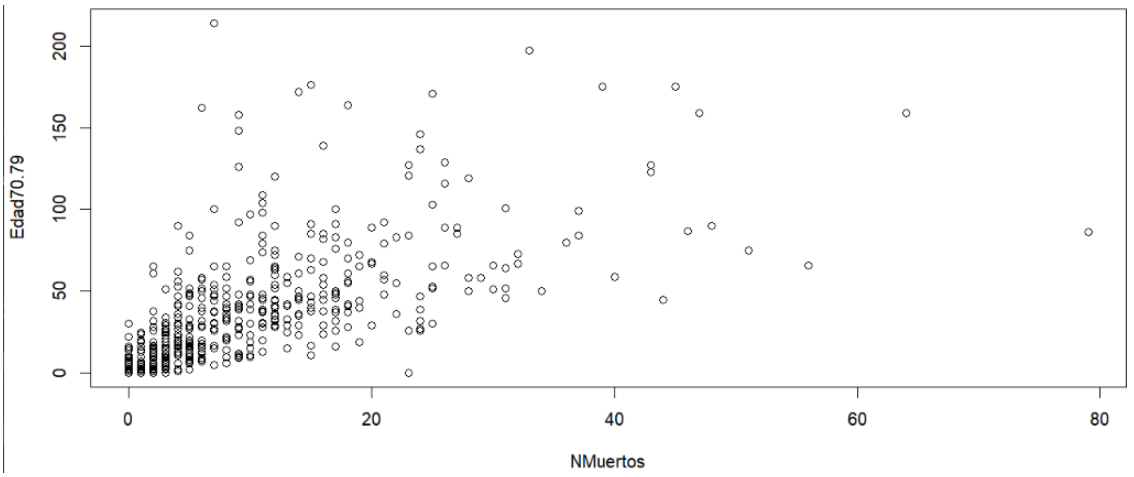
Gracias a la matriz de correlación podemos observar que las edades con mayores contagios son de 30-49 años muy seguido de 20-29 y de 50-59.

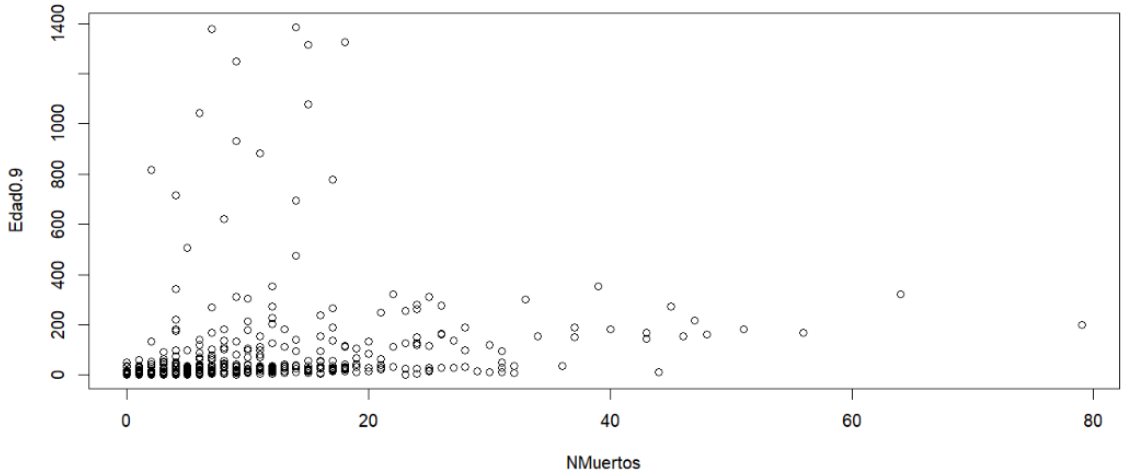
También se puede observar que la mayor tasa de defunción es de personas mayores de 80 años.Otra información que podemos apreciar es que ha habido mayor tasa de tratamiento en niños menores y esto ha ocasionado que su índice de muerte sea mucho menor que el resto.

Otros grafos

Scatter plot de los muertos según la edad.

```
#Tambien podemos visualizar por edades el numero de muertes
plot(df$NMuertos, df$Edad70.79, xlab = "NMuertos", ylab = "Edad70.79")
plot(df$NMuertos, df$Edad0.9, xlab = "NMuertos", ylab = "Edad0.9")
```



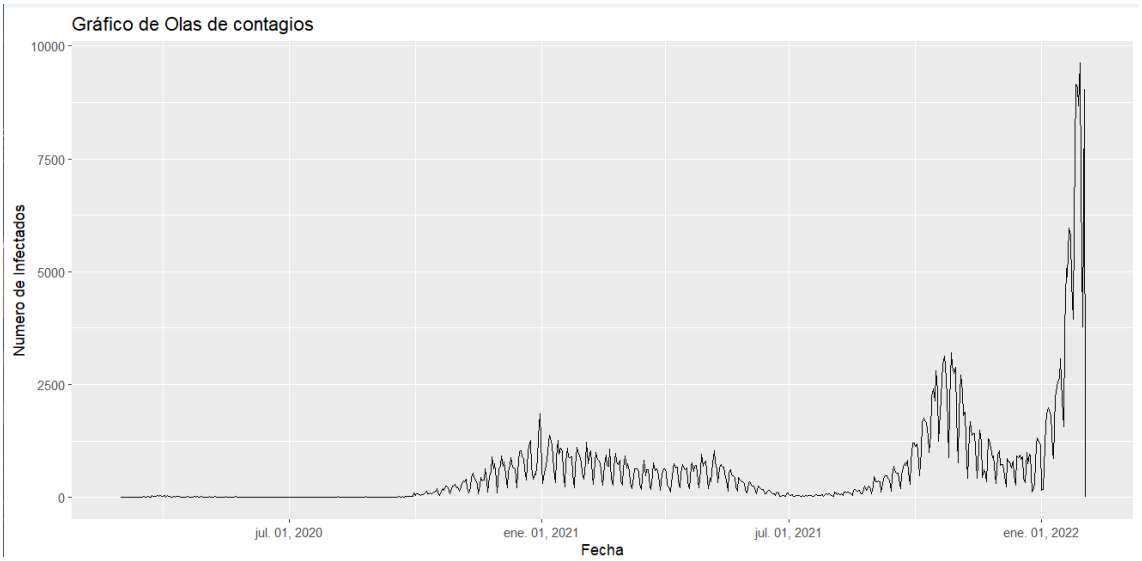


Distintos tipos de grafos por fecha:

Comandos y salidas

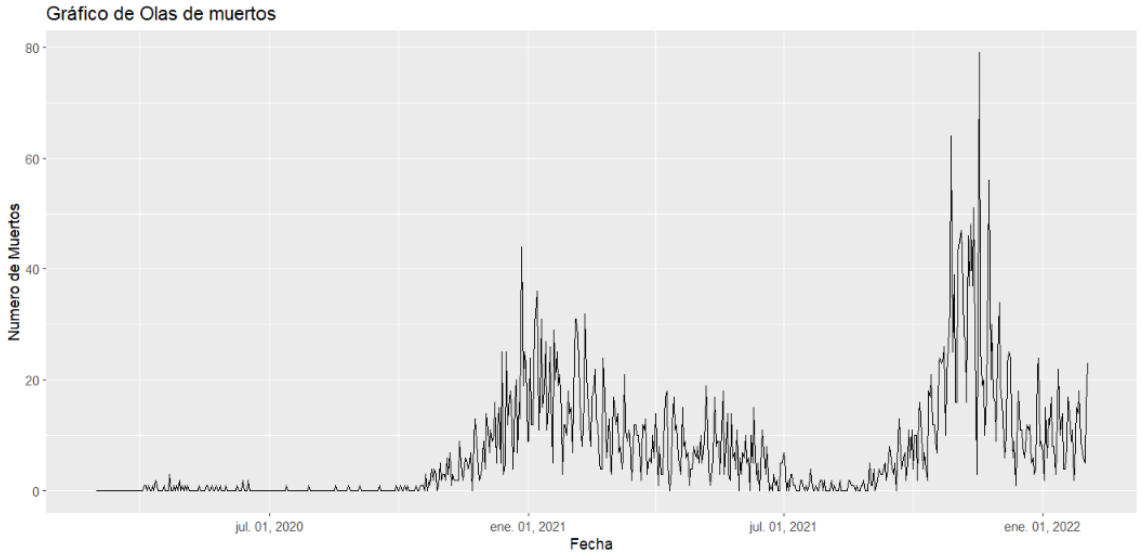
Por infectados:

```
#Tambien vamos a realizar una visualización de las diferentes olas de la pandemia.

#Sustitución de . final de los valores para su conversión.
df$Fecha <- gsub("\\.$", "", df$Fecha)
#Conversión de los datos de tipo char a Date
df$Fecha  <- as.Date(df$Fecha , format = "%Y.%m.%d")

#Creación de grafo de fechas y cantidad de infectados
ggplot(df, aes(Fecha, NInfectados)) +
  geom_line() +
  scale_x_date(date_labels = "%b %d, %Y") +
  ggtitle("Gráfico de Olas de contagios") +
  xlab("Fecha") +
  ylab("Numero de Infectados")
```



Por muertos:

```
#Creación de grafo de fechas y cantidad de muertos
ggplot(df, aes(Fecha, NMuertos)) +
  geom_line() +
  scale_x_date(date_labels = "%b %d, %Y") +
  ggtitle("Gráfico de Olas de muertos") +
  xlab("Fecha") +
  ylab("Numero de Muertos")
```



Por Pacientes tratados:

```
#Creación de grafo de fechas y cantidad de Pacientes Tratados
ggplot(df, aes(Fecha, NPTratados)) +
  geom_line() +
  scale_x_date(date_labels = "%b %d, %Y") +
  ggtitle("Gráfico de Olas de Tratados") +
  xlab("Fecha") +
  ylab("Nº Pacientes Tratados")
```