# Proyecto Redes Neuronales

Alvarado Serrano Marco Antonio, Herrera Flores Aldo

05/06/2020

## Introducción

### COVID-19

La COVID-19 (acrónimo del inglés coronavirus disease 2019), también conocida como coronavirus e incorrectamente como neumonía por coronavirus, es una enfermedad infecciosa causada por el virus SARS-CoV-2.

Produce síntomas similares a los de la gripe, entre los que se incluyen fiebre, tos seca, disnea, mialgia y fatiga. En casos graves se caracteriza por producir neumonía, síndrome de dificultad respiratoria aguda, sepsis y choque séptico que conduce a cerca de 3,75 % de los infectados a la muerte según la OMS. Los síntomas aparecen entre dos y catorce días, con un promedio de cinco días, después de la exposición al virus. Existe evidencia limitada que sugiere que el virus podría transmitirse uno o dos días antes de que se tengan síntomas, ya que la viremia alcanza un pico al final del período de incubación. No existe tratamiento específico; las medidas terapéuticas principales consisten en aliviar los síntomas y mantener las funciones vitales.

### Shinny

Shiny es un paquete R que facilita la creación de aplicaciones web interactivas directamente desde R. Puede alojar aplicaciones independientes en una página web o incrustarlas en documentos R Markdown o crear paneles . También puede ampliar sus aplicaciones Shiny con temas CSS , widgets html y acciones de JavaScript.

### Redes Neuronales

Las redes neuronales artificiales son un modelo computacional vagamente inspirado en el comportamiento observado en su homólogo biológico. Consiste en un conjunto de unidades, llamadas neuronas artificiales, conectadas entre sí para transmitirse señales. La información de entrada atraviesa la red neuronal, donde se somete a diversas operacionesl, produciendo unos valores de salida.

Estos sistemas aprenden y se forman a sí mismos, en lugar de ser programados de forma explícita, y sobresalen en áreas donde la detección de soluciones o características es difícil de expresar con la programación convencional. Para realizar este aprendizaje automático, normalmente, se intenta minimizar una función de pérdida que evalúa la red en su total. Los valores de los pesos de las neuronas se van actualizando buscando reducir el valor de la función de pérdida. Este proceso se realiza mediante la propagación hacia atrás.

El objetivo de la red neuronal es resolver los problemas de la misma manera que el cerebro humano, aunque las redes neuronales son más abstractas. Las redes neuronales actuales suelen contener desde unos miles a unos pocos millones de unidades neuronales.

## Objetivos

Dado que las redes neuronales se han utilizado para resolver una amplia variedad de tareas, como la visión por computador y el reconocimiento de voz, que son difíciles de resolver usando la ordinaria programación basado en reglas. El objetivo principal de este proyecto es crear una red neuronal para hacer clasificación

entre las personas que ya fueron diagnosticadas con el virus y saber cuáles de ellas son más propensas a llegar al caso letal de fallecimiento dada su situación médica y física.

Como objetivo secundario, se plantea la creación de una shinny app multipropósito para el análisis de datos, la cual incluye un módulo para hacer clasificación con una red neuronal de arquitectura customizable mediante la interfaz gráfica de la misma app. Con ella se podrá replicar el mismo análisis del objetivo principal.

## Materiales

Se utilizará la base de datos de COVID proporcionada por el gobierno de la Ciudad de México. (El link se anexa al final del documento)

## Resultados

### Carga y limpia de la base de datos.

Para este proceso, primero cargaremos la base de datos que mencionamos anteriormente y aplicamos algunas funciones para obtener las variables de interés para nuestro estudio, así como funciones de limpieza y modificación. También cargaremos los diccionarios de las variables para poder ayudarnos luego con ellas.

```
setwd("C:/Users/Marco/Downloads/GLM/Proyecto/")
datosCovid <- read_csv("200606COVID19MEXICO.csv")
catSex <- read_xlsx("diccionario_datos_covid19/Catalogos_0412.xlsx", sheet = 3)
catSi_No <- read_xlsx("diccionario_datos_covid19/Catalogos_0412.xlsx", sheet = 5)
datosCovid <- datosCovid[-c(1,2,3,7)]

#Convertimos las columnas de clase character a factor
datosCovid <-  EDAS::cambio_prop(datosCovid,
                                 columns = which(sapply(datosCovid, class) != "Date"),
                                 prop = "factor")

#Modificamos la variable edad para dividirla en 5 grupos
datosCovid$EDAD <- as.factor(case_when(as.numeric(datosCovid$EDAD)  < 21 ~ 0,
                                       as.numeric(datosCovid$EDAD)  %in%  21:30 ~ 1,
                                       as.numeric(datosCovid$EDAD)  %in%  31:40 ~ 2,
                                       as.numeric(datosCovid$EDAD)  %in%  41:50 ~ 3,
                                       as.numeric(datosCovid$EDAD)  %in%  51:60 ~ 4,
                                       as.numeric(datosCovid$EDAD)  > 60 ~ 5))

# Agregamos una variable auxiliar para poder modificar inmediatamente la variable 'DEFUNCION' y que ést
datosCovid <- datosCovid %>%
  mutate(FECHA_DEF = as.factor(if_else(!is.na(FECHA_DEF),1,2))) %>%
  rename(DEFUNCION=FECHA_DEF)

#Seleccionamos las variables de interés, limpiamos NA's y eliminamos los levels que no necesitamos
datosCovidG <- datosCovid %>%
  select(INTUBADO, SEXO, EDAD, HIPERTENSION, DIABETES, OBESIDAD, EPOC,
         ASMA, TABAQUISMO, INMUSUPR , NEUMONIA, DEFUNCION, RESULTADO) %>%
  filter(INTUBADO %in% c("1","2"), RESULTADO == "1", HIPERTENSION %in% c("1","2"),
         DIABETES %in% c("1","2"), OBESIDAD %in% c("1","2"), EPOC %in% c("1","2"),
         ASMA %in% c("1","2"), TABAQUISMO %in% c("1","2"),
         INMUSUPR %in% c("1","2"), NEUMONIA %in% c("1","2"))  %>%
  select(-RESULTADO) %>%
  droplevels()
```

Vemos los nombres de las variables:

```r
names(datosCovidG)
```

```
##  [1] "INTUBADO"     "SEXO"         "EDAD"         "HIPERTENSION" "DIABETES"
##  [6] "OBESIDAD"     "EPOC"         "ASMA"         "TABAQUISMO"   "INMUSUPR"
## [11] "NEUMONIA"     "DEFUNCION"
```

Donde todas, a excepción de 'EDAD', son variables binarias y: * INTUBADO: Indica si la persona fue intubada. * SEXO: Indica el sexo de la persona (masculino/femenino) * EDAD: Indica a qué grupo de edad pertenece la persona. * HIPERTENSION: Indica si la persona tiene diagnóstico de hipertensión. * DIABETES: Indica si la persona tiene diagnóstico de diabetes. * OBESIDAD: Indica si la persona tiene diagnóstico de sobrepeso. * EPOC: Indica si el paciente tiene un diagnóstico de EPOC (Enfermedad pulmonar obstructiva crónica). * ASMA: Indica si la persona tiene diagnóstico de asma. * TABAQUISMO: Indica si la persona tiene hábito de tabaquismo. * INMUSUPR: Indica si la persona presenta inmunosupresión. * NEUMONIA: Indica si la persona tiene un diagnóstico neumonía. * DEFUNCION: Indica si la persona falleció.

### Modificamos la base a conveniencia

Lo que hacemos a continuación es agregar una nueva variable auxiliar llamada 'aux' para posteriormente hacer conteos, aunque esto parezca un paso innecesario, en realidad es muy útil para evitar posteriores errores en la función que crearemos para hacer gráficas. Además haremos 'paquetitos' para cada variable explicativa y donde cada uno de ellos contiene las variables 'DEFUNCION', 'aux' y 'value'. Donde:

- DEFUNCION: Indica si la persona falleció.
- aux: Es la variable auxiliar de conteo.
- value: Indica a qué grupo pertenece (Si/No, masculino/femenino, grupo de edad)

```r
datosCovidGN <- datosCovidG  %>%
  mutate(aux = 1) %>%
  gather(key="key", value = "value", -c(DEFUNCION,aux)) %>%
  group_by(value,key) %>%
  group_by(key) %>%
  nest()
```

### Creamos la función auxiliar para las gráficas

```r
graph2 <- function(data,b, labelsSF){
  data1 <- data[[2]][[b]] %>%
    group_by(DEFUNCION,value) %>%
    summarise(NAC = sum(aux)) %>%
    mutate(aux = sum(NAC), prop = NAC/aux)

  g1 <- data1 %>%
    ggplot(aes(x= DEFUNCION, y= NAC,fill=value)) +
    geom_col(position="dodge") +
    theme(axis.text.x = element_text(angle = 45, size = 20),
        axis.text.y = element_text(size = 20),
        axis.text=element_text(size=20, face="bold"),
        axis.title=element_text(size=25,face="bold"),
        title = element_text(size = 20, face = "bold"),
        legend.text = element_text(size = 25),
        legend.title = element_text(size=30, face = "bold"))  +
    xlab("DEFUNCION") +
    ylab(label = "Count") +
    labs(title = paste("Numero de personas muertas dado", data[[1]][b])) +
```

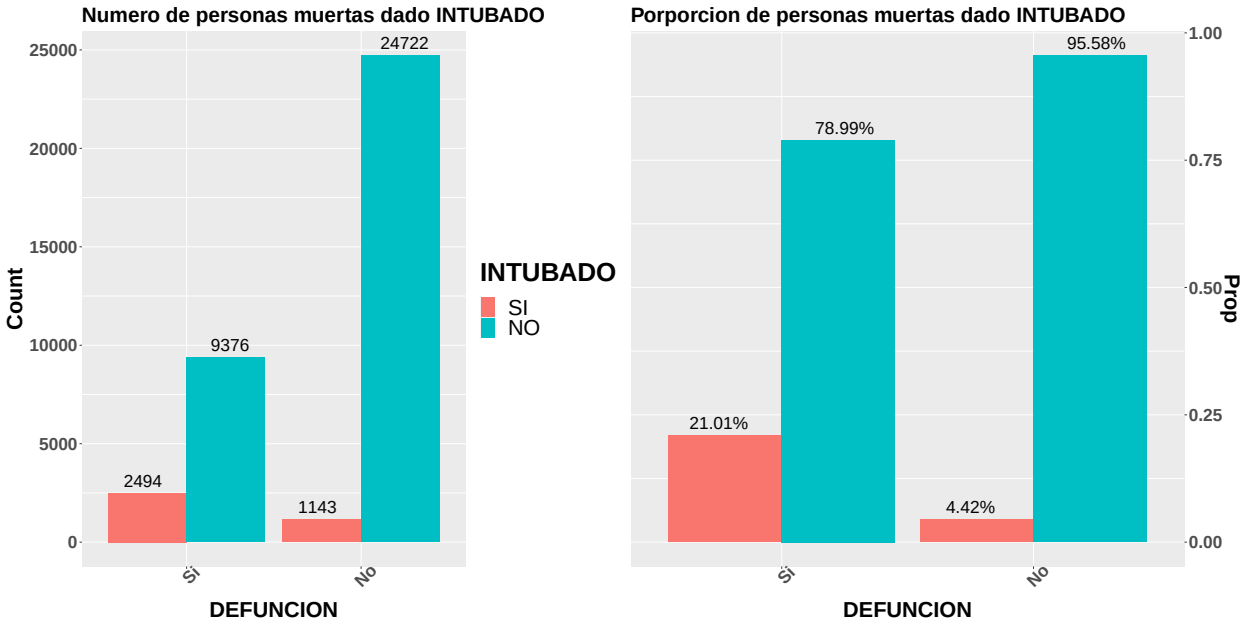
```
    scale_fill_discrete(name = data[[1]][b], labels = labelsSF) +
    geom_text(aes(y=NAC, label =  NAC),
              size=7, alpha=1, position = position_dodge(1), vjust=-.5) +
    scale_x_discrete(labels= c("Si", "No"))

  g2 <- data1 %>%
    ggplot(aes(x= DEFUNCION, y= prop,fill=value)) +
    geom_col(position="dodge") +
    theme(axis.text.x = element_text(angle = 45, size = 20),
        axis.text.y = element_text(size = 20),
        axis.text=element_text(size=20, face="bold"),
        axis.title=element_text(size=25,face="bold"),
        title = element_text(size = 20, face = "bold"))  +
    xlab("DEFUNCION") +
    ylab(label = "Prop") +
    labs(title = paste("Porporcion de personas muertas dado", data[[1]][b])) +
    guides(fill=F) +
    scale_y_continuous(position = "right") +
    geom_text(aes(y=prop, label =  paste0(round(prop,4)*100,"%")),
              size=7, alpha=1, position = position_dodge(1), vjust=-.5) +
    scale_x_discrete(labels= c("Si", "No"))

  grid.arrange(g1, g2, ncol=2)
}
```

**EDA**

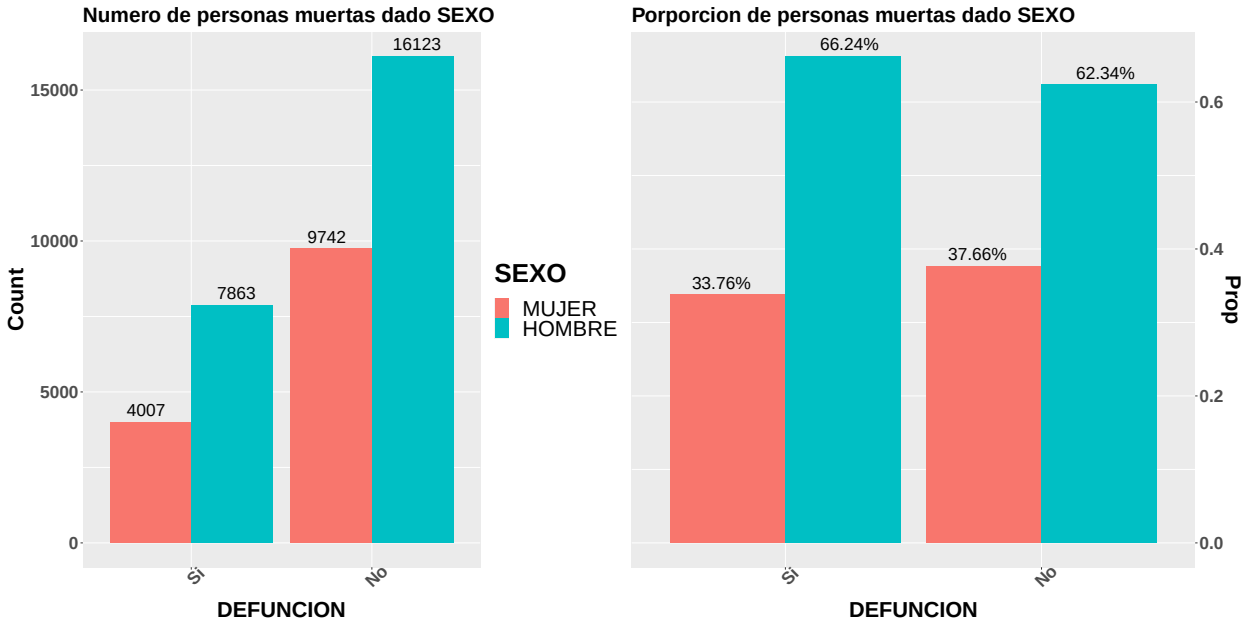
```
graph2(data = datosCovidGN, b = 1, labelsSF = catSi_No$DESCRIPCIÓN)
```




En esta primera gráfica, podemos notar una amplia diferencia entre la mortalidad de aquellas personas que fueron intubadas, y las que no. Vemos que si la persona ya fue diagnosticado por corinavirus, hay un 50% más personas que lleguaron al caso letal de fallecimiento debido a que no fueron intubadas, comparado con aquellas que sí fueron intubadas. Por otro lado, vemos que aquellas personas que no fallecieron, el 95% no

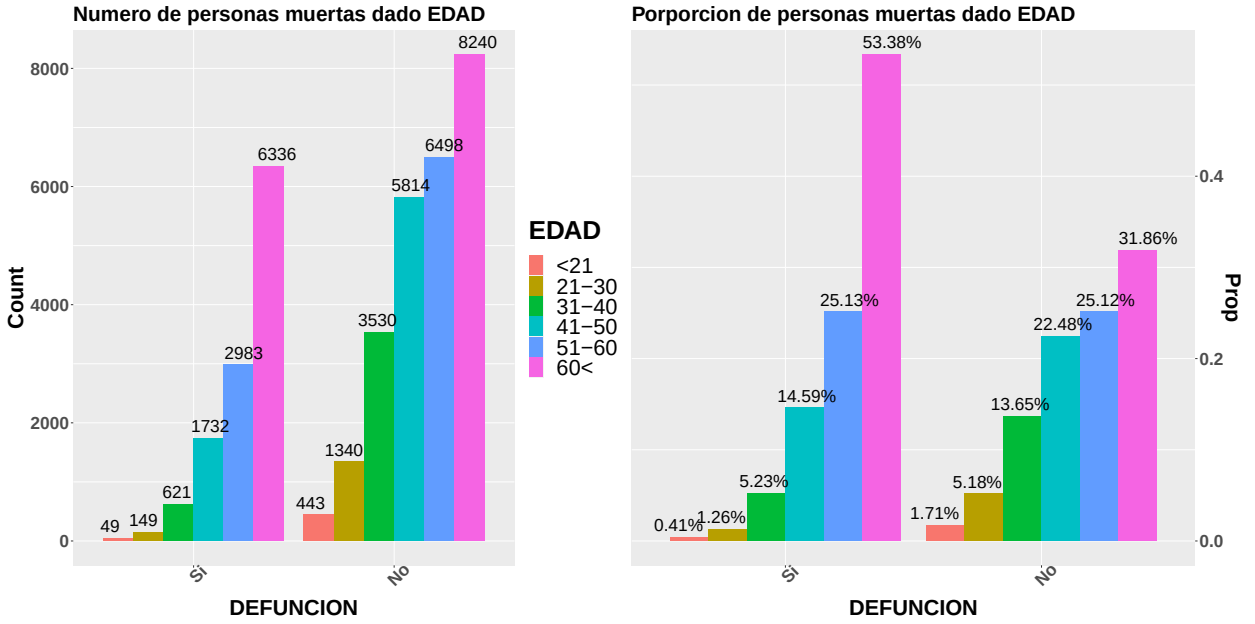
fueron intubadas, esto se debe a que posiblemete éstas solo presentaron síntomas leves, o pertenecieron a la población asintomática, y no requirieron de mayor atención.

```
graph2(data = datosCovidGN, b = 2, labelsSF = catSex$DESCRIPCIÓN)
```



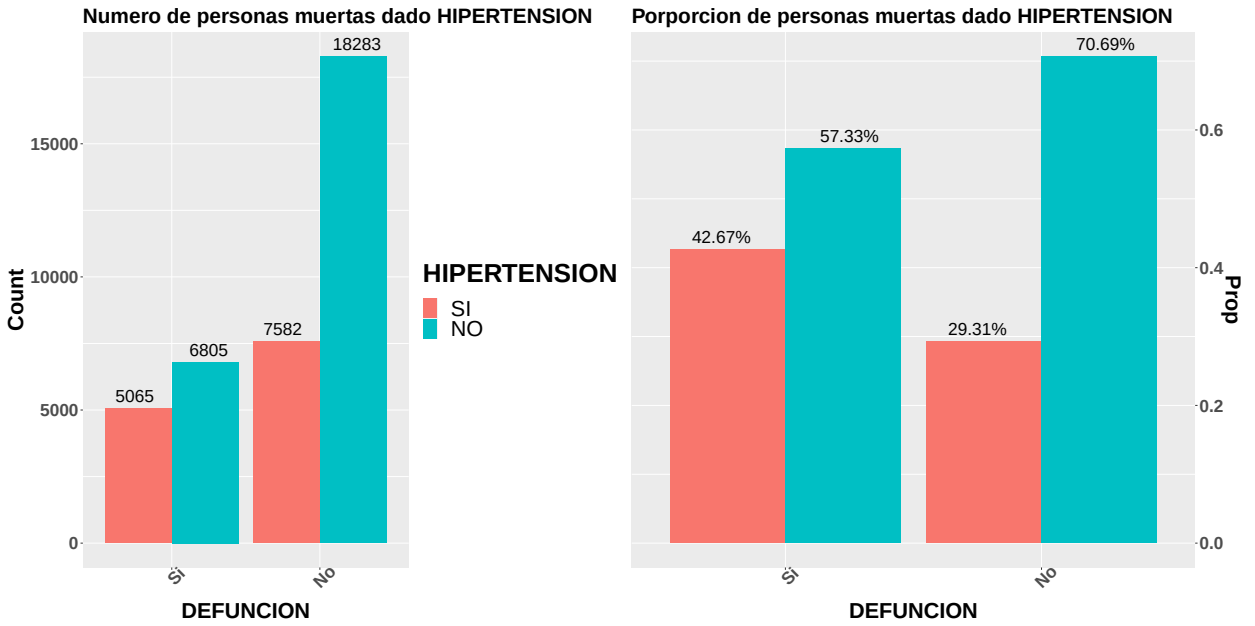

Para este caso podemos ver que las personas que murieron son en su mayoría hombres, esto se debe posiblemente a que este grupo prefiere no asistir al doctor o a que son más descuidados con su salud. Además, implicitamente vemos que la población que más contrae el virus son los mismos hombres.

```
graph2(data = datosCovidGN, b = 3,
       labelsSF = c("<21", "21-30", "31-40", "41-50", "51-60", "60<" ))
```

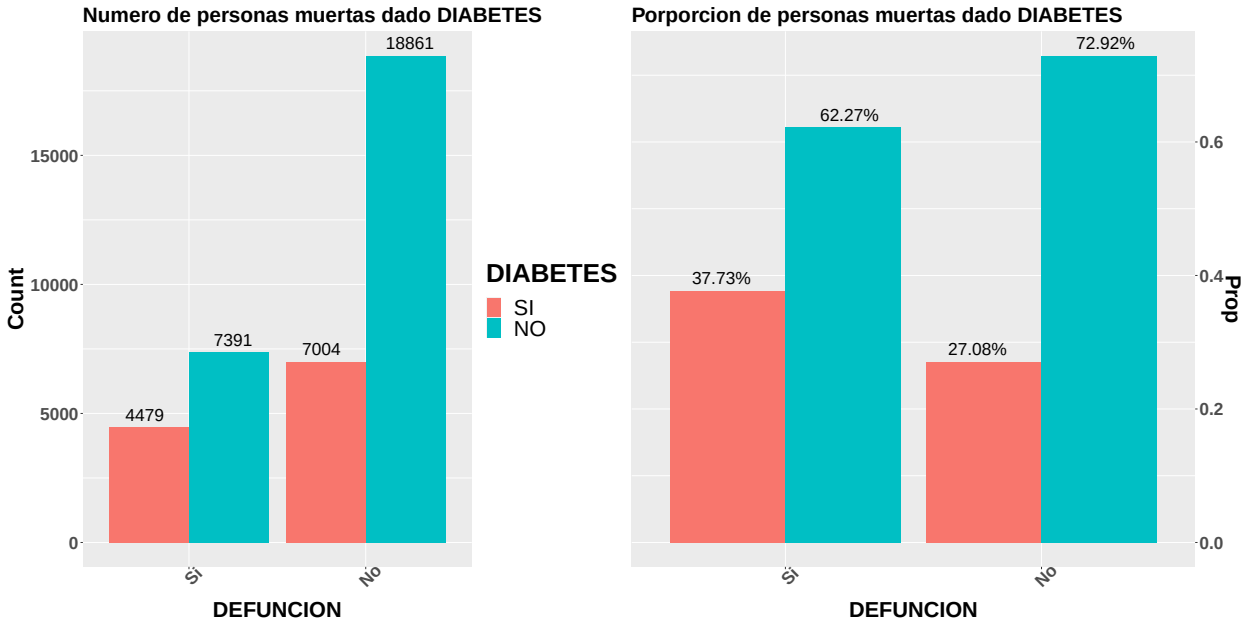



La gráfica por grupos de edad nos muestra que las personas pertenecientes al grupo 5 (más de 60 años) tienen una mayor mortalidad a diferencia de los demás grupos, esto debido tal vez a la debilidad de su sistema inmunológico y la poca resistencia que tienen sus funciones vitales.

```
graph2(data = datosCovidGN, b = 4, labelsSF = catSi_No$DESCRIPCIÓN)
```

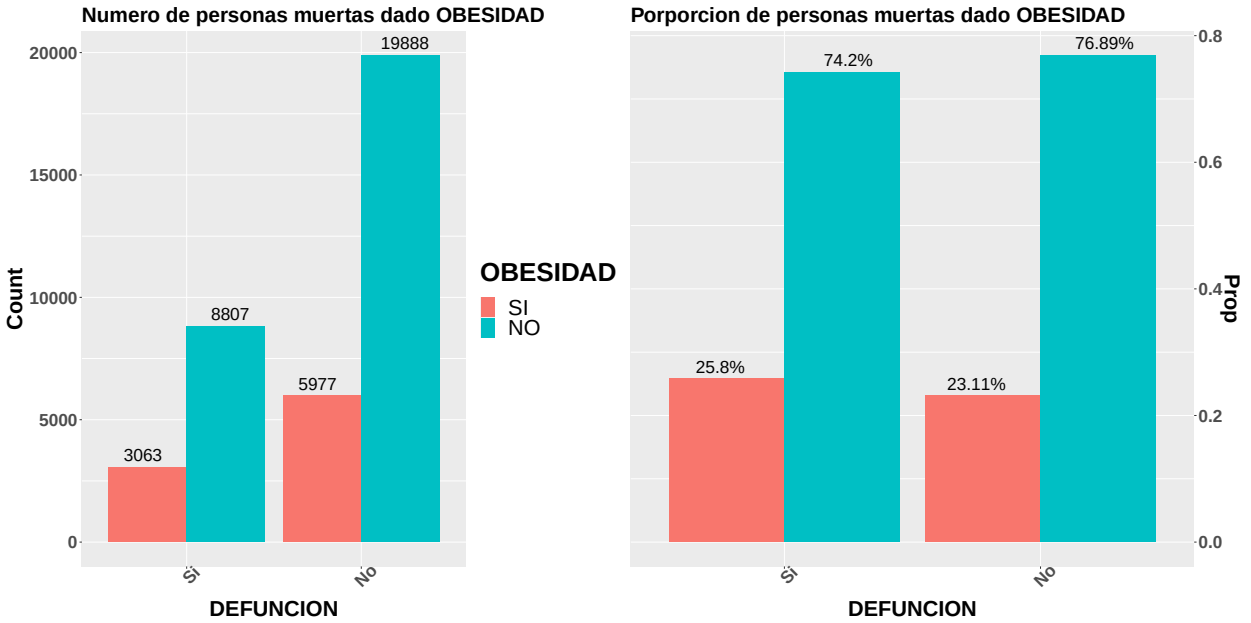



```
graph2(data = datosCovidGN, b = 5, labelsSF = catSi_No$DESCRIPCIÓN)
```

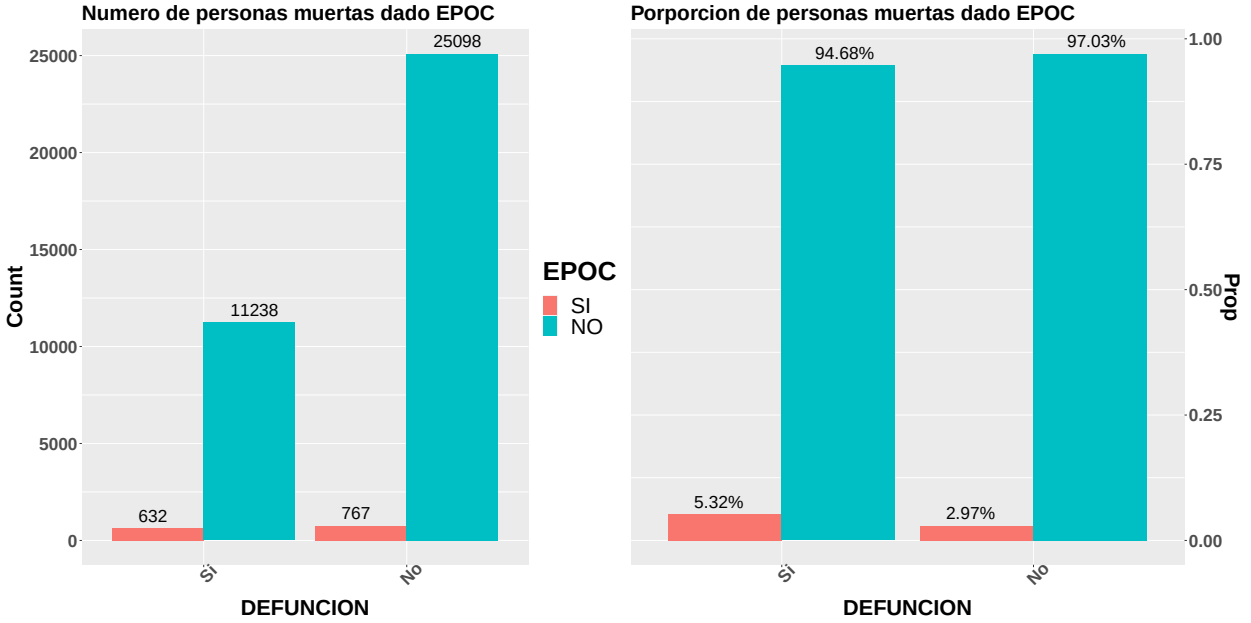



En estas dos gráficas nos muestran que hay una poca distinción entre las personas que fallecieron, si se cuenta con alguna de estas dos enfermedades o no.

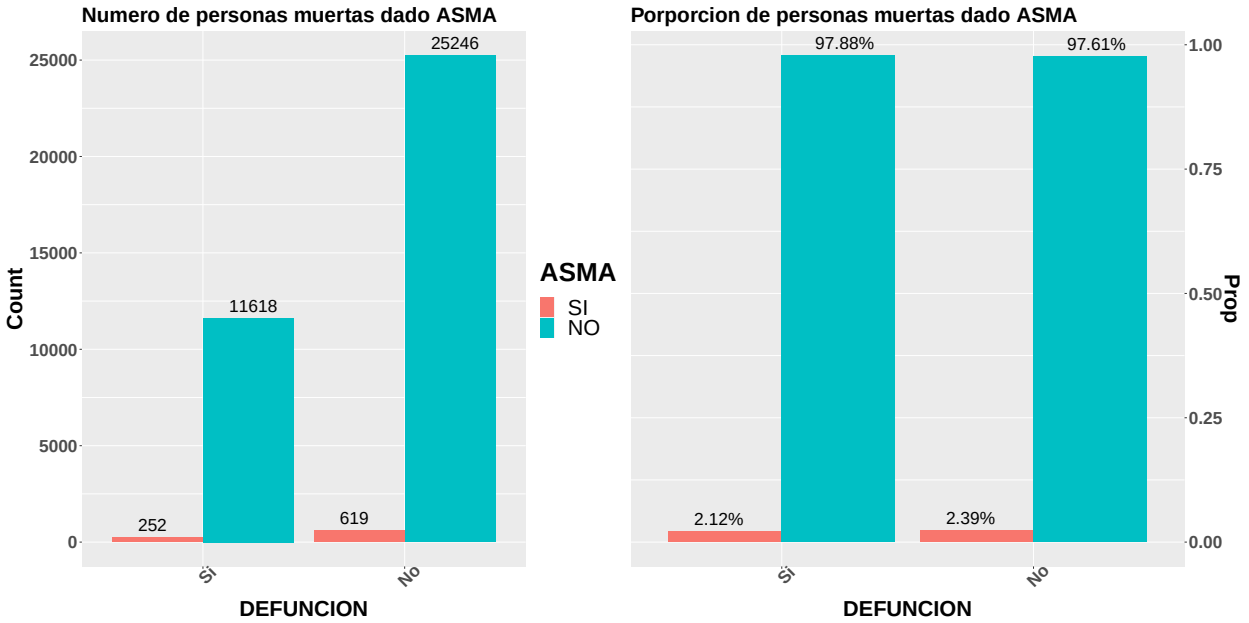
```
graph2(data = datosCovidGN, b = 6, labelsSF = catSi_No$DESCRIPCIÓN)
```



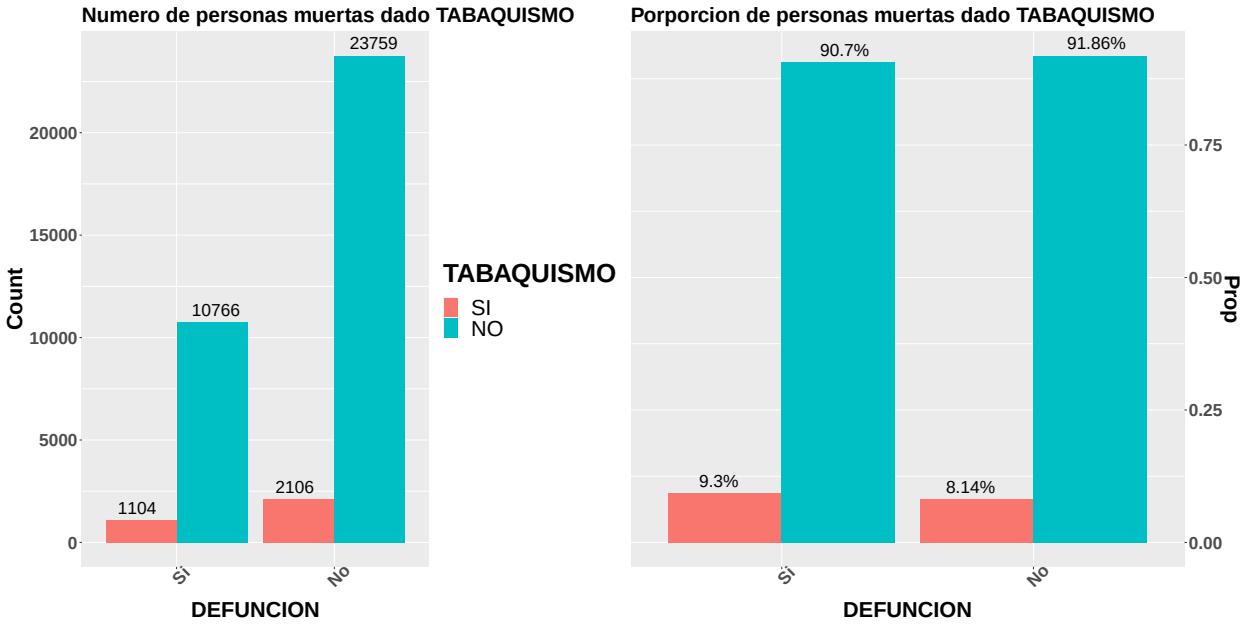

```
graph2(data = datosCovidGN, b = 7, labelsSF = catSi_No$DESCRIPCIÓN)
```



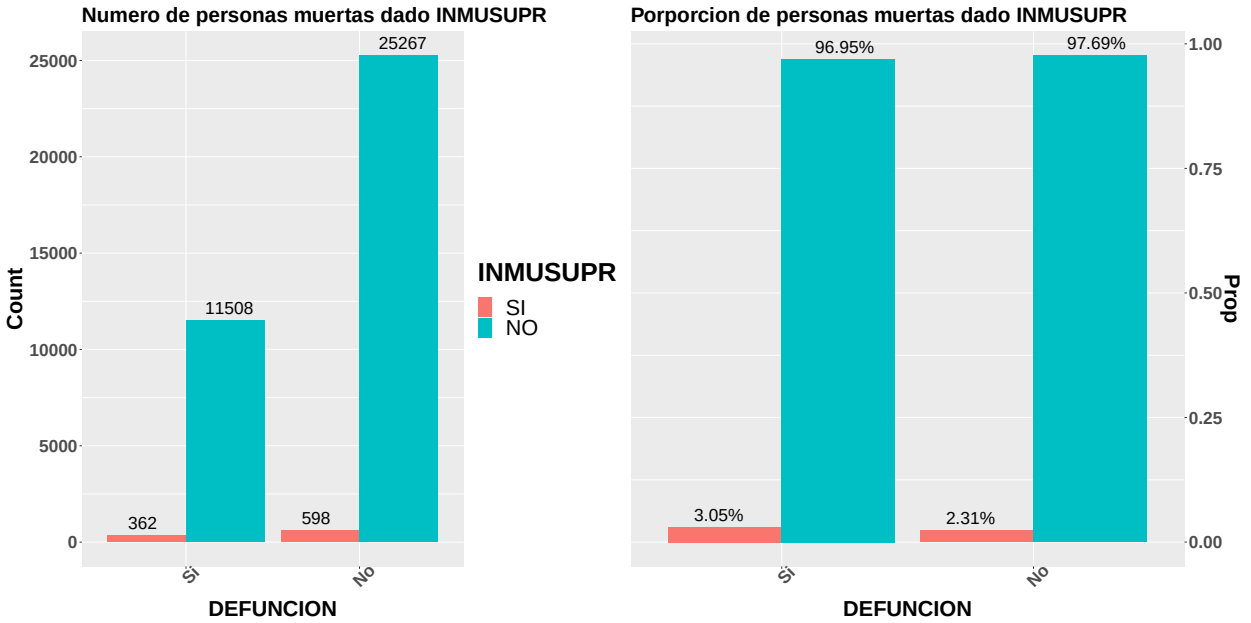

```
graph2(data = datosCovidGN, b = 8, labelsSF = catSi_No$DESCRIPCIÓN)
```




```
graph2(data = datosCovidGN, b = 9, labelsSF = catSi_No$DESCRIPCIÓN)
```

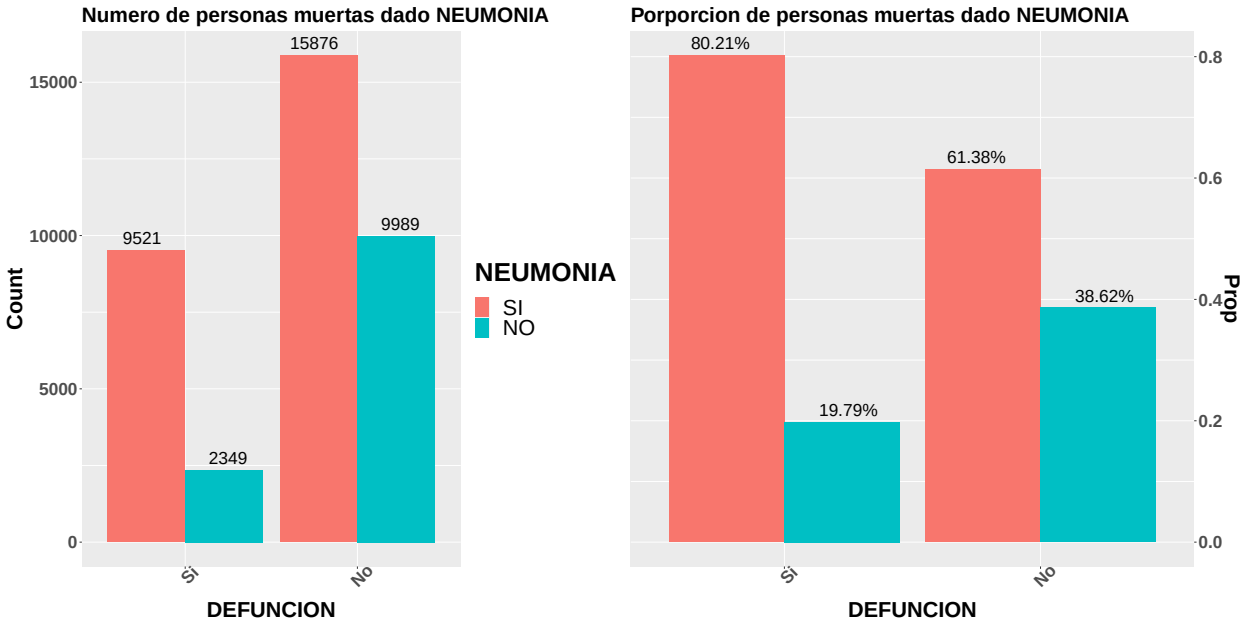



```
graph2(data = datosCovidGN, b = 10, labelsSF = catSi_No$DESCRIPCIÓN)
```




Las anteriores gráficas nos revelan que las personas que cuentan con alguna enfermedad mencionada no son más propensos a fallecer, sin embargo, vemos que hay un gran sesgo hacia este grupo de personas.

```
graph2(data = datosCovidGN, b = 11, labelsSF = catSi_No$DESCRIPCIÓN)
```




Finalmente, en esta gráfica vemos que el factor de neumonía toma un gran papel para saber si una persona va a fallecer o no, dado que hay casi un 60% más de personas que murieron dado que tenian neumonia en comparación a las que no.

### Creamos la red neuronal

Como paso intermedio, modificaremos las variables explicativas para convertirlas a numéricas, ya que la función así lo requiere, y así llegamos al paso final que es crear la red.

```
datosMod <- datosCovidG %>%
  mutate_at(c("INTUBADO", "SEXO", "EDAD", "HIPERTENSION", "DIABETES", "OBESIDAD", "EPOC",
         "ASMA", "TABAQUISMO", "INMUSUPR" , "NEUMONIA"),as.numeric)
```

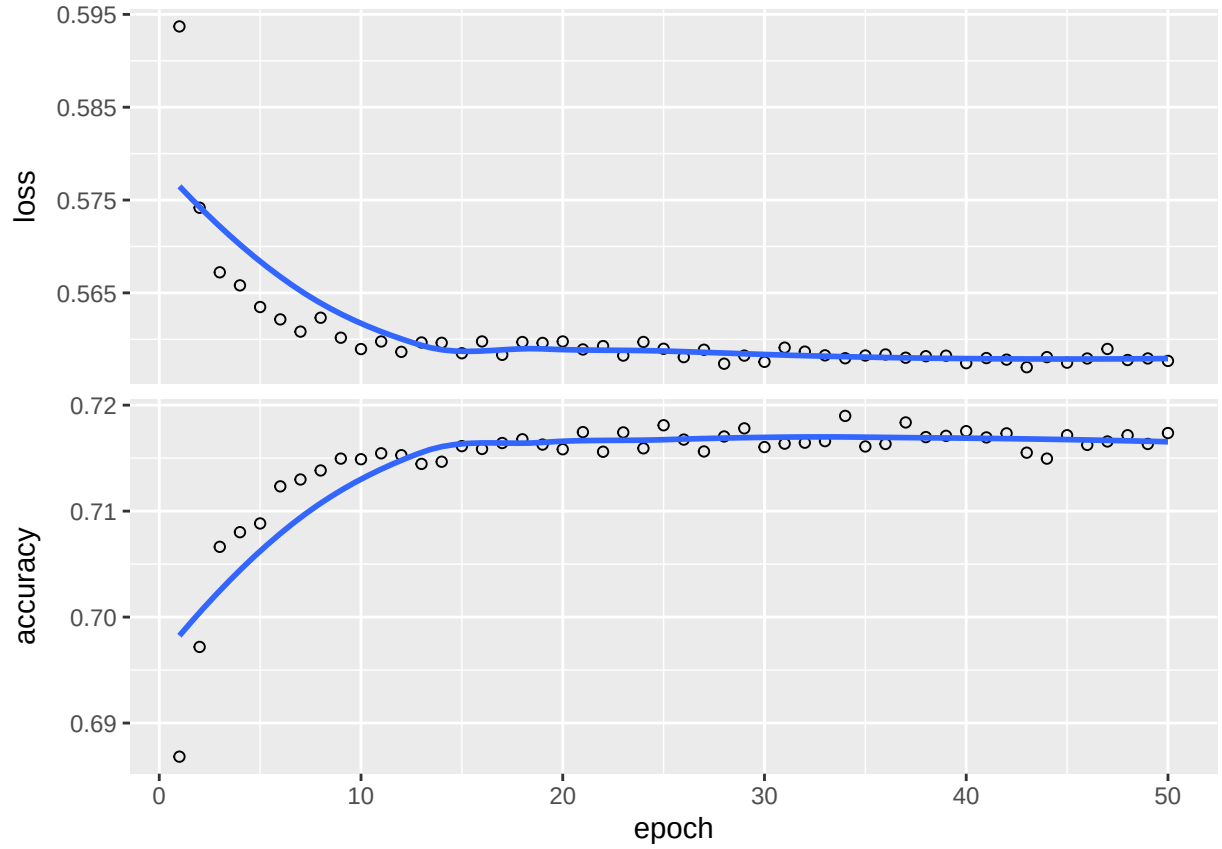
Para este paso final, utilizaremos un paquete que creamos previamente (EDAS) y que se utiliza en la shinny app, para ello, sólo tenemos que llamar la función con los parámetros necesarios.

Se utilizará una red neuronal con 3 capas ocultas de 100, 50 y 30 neuronas cada una y funciones de activación relu, tanh y softmax respectivamente, así como función de salida softplus. Además, se utilizarán 50 epochs y un batchsize de 32. La red se compiló con optimizador Adam y métrica de precisión. Se usará el 90% de los datos para el entrenamiento y el 10% para prueba.

```
set.seed(1248)
mod <- redes(data = datosMod, var_est = "DEFUNCION", vPerc = 90, unitsE = c(100,50,30),
      activationE = c("relu", "tanh", "softmax", "softplus"), lDrop = c(50,40,20), epochsE = 50,
      batch_sizeE = 32, validation_splitE = 0)
```

```
plot(mod$plt)
```

```
## `geom_smooth()` using formula 'y ~ x'
```




```
mod$ptest
```

```
## $loss
## [1] 0.5513823
##
## $accuracy
## [1] 0.7270299
```

Con la red obtenemos un 72.7 de presición. Vemos cómo queda la matríz de consufusión.

```
mod$predT
```

```
##           Actual
## Predicted    0    1
##         0  250   97
##         1  925 2472
```

Vemos que se clasifican de manera decente los casos de fallecimiento (0) con un 72.05% de presición y para no fallecimiento (1) con un 72.77%

## Conclusiones

Vemos que la red funciona de buena manera para la clasificación de fallecimientos por el virus COVID-19, teniendo una buena presición y necesitando de pocas capas ocultas en la red.

En cuanto a la shinny app, se decidió no incluirla en el reporte debido al denso código que hay detrás de ella, sin embargo, podemos concluir que es una buena herramienta para aquellas personas que quieren incursionar en el mundo de las redes neuronales artificiales y que saben sobre la teoría detrás de ella pero que no saben cómo programarlas. Además, es una manera cómoda de crear modelos de manera rápida, fácil y sencilla.

## Bibliografía


https://es.wikipedia.org/wiki/COVID-19

https://shiny.rstudio.com/

https://es.wikipedia.org/wiki/Red_neuronal_artificial

Base de datos COVID-19

https://datos.gob.mx/busca/dataset/informacion-referente-a-casos-covid-19-en-mexico