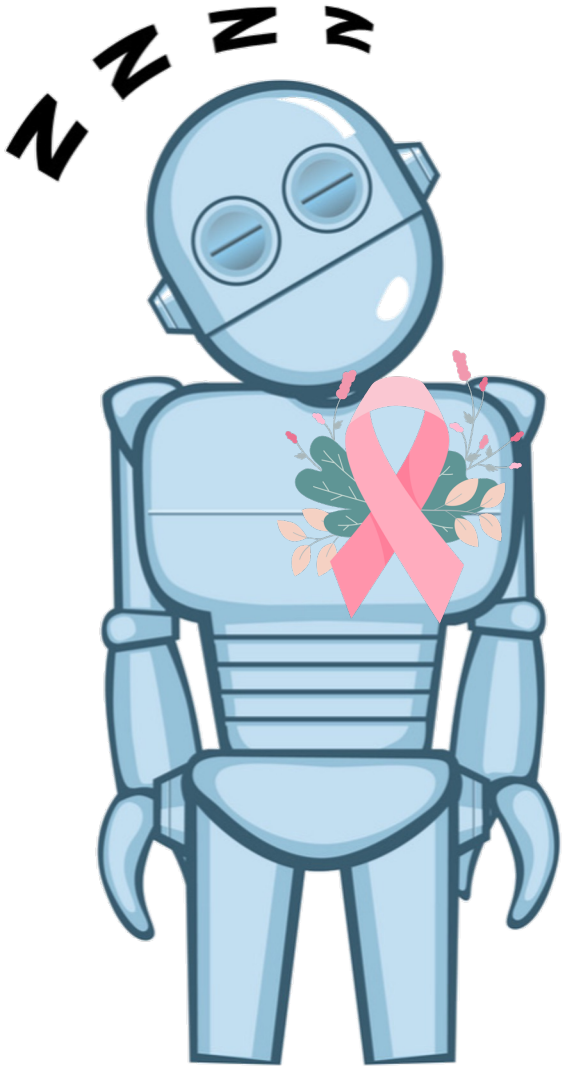

Laboratorio No 01 – Cáncer de Mama

Lazy Learning Lab

Classification Using Nearest Neighbors

Creado por: Ing. Marcela Parra, Mg

Paso 01 – Recopilación de Datos

- Se utilizará el conjunto de datos **"Breast Cancer Wisconsin Diagnostic"** del Repositorio de aprendizaje automático de la UCI, que está disponible en **http://archive.ics.uci.edu/ml.**

- Estos datos fueron **donados por investigadores** de la **Universidad de Wisconsin** e incluyen **mediciones de imágenes digitalizadas** de aspiración con aguja fina de una masa mamaria.

LABORATORIO 01

Paso 01 – Recopilación de Datos

Los datos sobre el cáncer de mama incluyen **569 ejemplos de biopsias de cáncer**, cada uno con **32 características**. El diagnóstico se codifica como **M para indicar maligno o B para indicar benigno**. Algunas mediciones son:

- Radio
- Textura
- Perímetro
- Área
- Suavidad
- Compacidad
- Concavidad
- Puntos cóncavos
- Simetría
- Dimensión fractal

LABORATORIO 01

Paso 02. Instalación de librerías

Se instalarán las siguientes librerías y se llamarán posteriormente.

```
# Instalar las librerías
install.packages("tidyverse")
install.packages("class")
install.packages("gmodels")
install.packages("caret")


# Llamar a las librerías
library(tidyverse)
library(class)
library(gmodels)
library(caret)
```

LABORATORIO 01

Paso 03. Preparación de los Datos

Para este análisis, se utilizará el conjunto de datos que se encuentra en el sitio web de la **editorial Oreilly**. Descargamos el **archivo .csv** y lo guardamos en un **data frame.**

```
# Descargamos el archivo .csv

download.file("https://resources.oreilly.com/examples/9781784393908/raw/ac9fe41596dd42fc3877cfa8ed410dd346c43548/Machine%20Learning%20with%20R,%20Second%20Edition_Code/Chapter%2003/wisc_bc_data.csv",
destfile = "wisc_bc_data.csv")

wisc_data <- read.csv(file = "wisc_bc_data.csv")
str(wisc_data)
```

LABORATORIO 01

Paso 03. Preparación de los Datos

```
# Eliminamos el ID y no es util y puede dar un sobrajuste la
eliminamos
wisc_data <- wisc_data[,-1]


# La variable diagnóstico, es de particular interés, ya que es el
resultado que esperamos predecir.
# table(wisc_data$diagnosis)
```

LABORATORIO 01

Paso 03. Preparación de los Datos

```
# Daremos valores B y M etiquetas más informativas usando el
parámetro etiquetas:
wisc_data <- mutate(wisc_data,diagnosis =
fct_recode(wisc_data$diagnosis,"Benigno" = "B","Maligno" = "M"))


# Miremos la salida de prop.table(), se nota que los valores han
sido etiquetados como Benignos y Malignos
round(prop.table(table(wisc_data$diagnosis)) * 100, 1)


#Las 30 características restantes son todas numéricas
summary(wisc_data[c("radius_mean", "area_mean", "smoothness_mean")])
```

LABORATORIO 01

Paso 03. Preparación de los Datos

```
#Normalizamos las variables
normalize <- function(x){return ((x - min(x))/(max(x) - min(x)))}
wisc_data_n <- as.data.frame(lapply(wisc_data[2:31], normalize))


#Para confirmar que la transformación se aplicó correctamente, veamos
las estadísticas resumidas de una variable:
summary(wisc_data_n$area_mean)
```

LABORATORIO 01

Paso 03. Preparación de los Datos

```
#Usaremos los primeros 469 registros para el conjunto de datos de
entrenamiento y los 100 restantes para simular nuevos pacientes
wisc_training <- wisc_data_n[1:469,]
wisc_test <- wisc_data_n[470:569,]

# Dividimos en los conjuntos de datos de entrenamiento y prueba con
etiquetas
wisc_training_labels <- wisc_data[1:469,1]
wisc_test_labels <- wisc_data[470:569,1]
```

LABORATORIO 01

Paso 04. Entrenar el Modelo

```
# Paso 04 - Entrenar el modelo
##########################################
#Como nuestros datos de entrenamiento incluyen 469 instancias,
podríamos probar con k = 21, un número impar aproximadamente igual a
la raíz cuadrada de 469.


wisc_test_predicted <- knn(wisc_training, wisc_test, cl =
wisc_training_labels, k = 21)
wisc_test_predicted
```

LABORATORIO 01

Paso 05. Evaluar el Problema

```
# Paso 05 - Evaluar el modelo
##############################################
#Estadístico de Prueba Chi cuadrado
CrossTable(x = wisc_test_labels, y = wisc_test_predicted,
prop.chisq=FALSE)


#Tabla Cruzada
confusionMatrix(data = wisc_test_predicted, reference =
wisc_test_labels)
```

LABORATORIO 01

Paso 06. Mejorar el Modelo

```
#Cambiamos todas las funciones con excepción del diagnóstico y
almacena el resultado como un marco de datos en la variable
wisc_data_z.
wisc_data_z <- as.data.frame(scale(wisc_data[2:31]))


#Confirmar que la transformación se aplicó correctamente, podemos
mirar las estadísticas resumidas:
summary(wisc_data_z$area_mean)


#Compararemos las etiquetas previstas con las etiquetas reales usando
CrossTable()
wisc_training_z <- wisc_data_z[1:469,]
wisc_test_z <- wisc_data_z[470:569,]
wisc_test_predicted_z <- knn(wisc_training_z, wisc_test_z, cl =
wisc_training_labels, k = 21)
confusionMatrix(data = wisc_test_predicted_z, reference =
wisc_test_labels)
```

LABORATORIO 01

Resumen

https://resources.oreilly.com/examples/9781784393908/-/blob/master/Machine%20Learning%20with%20R%2C%20Second%20Edition_Code/Chapter%2002/MLwR_v2_02.r