

Modelo de Regresión Logística

Emmanuel Naranjo Blanco – A00835704; Fernanda Martínez Valles – A01722279.

Tecnológico de Monterrey – Campus Monterrey

Fecha de entrega: 30/11/2023

Profesores: Dr. Alexander Garrido Ríos

INTRODUCCIÓN

En este informe, abordaremos el desafío de clasificar el género de individuos en función de su peso y altura utilizando la técnica de regresión logística.

Este enfoque estadístico nos permite modelar la relación entre las variables predictoras (peso y altura) y la variable de respuesta (género) a través de la estimación de los coeficientes del modelo (b_0 , b_1 y b_2).

El objetivo es comprender cómo estas características antropométricas influyen en la probabilidad de pertenecer a una categoría de género específica. El proceso incluye la obtención del conjunto de datos, el entrenamiento del clasificador logístico y la realización de predicciones basadas en dicho modelo para un nuevo conjunto de datos. En otras palabras, utilizaremos un dataset de 10 000 instancias de tipo numérico que contiene 3 atributos; 2 atributos serán predictores: “Height” y “Weight”, para clasificar “Gender” (atributo de tipo texto) en “Male” o “Female”.

Para visualizar de manera efectiva la relación entre las variables, se generará una gráfica que represente la línea de decisión del modelo, proporcionando una representación visual de cómo se establece la frontera de clasificación en función de las dimensiones físicas de cada individuo. Este enfoque no solo contribuirá a la comprensión de las dinámicas subyacentes, sino que también permitirá evaluar la eficacia del modelo en la tarea de clasificación categórica.

Importación de Librerías y Dataset

Inicialmente se instalan los siguientes paquetes y se procede con cargar la base de datos.

```
library("readxl")
## Warning: package 'readxl' was built under R version 4.3.2
library("ggplot2")
## Warning: package 'ggplot2' was built under R version 4.3.2
library("caret")
## Warning: package 'caret' was built under R version 4.3.2
## Loading required package: lattice
library("skimr")
## Warning: package 'skimr' was built under R version 4.3.2
data<-read_excel("01_heights_weights_genders-1-1.xlsx")
```

Visualización de los Atributos y sus Datos

Distribución de los datos

Al observar el conjunto de datos podemos notar un resumen de estos, donde podemos ver que “Gender” es de tipo texto y contiene 10 000 instancias, y un resumen de la distribución estadística de las variables numéricas “Height” y “Weight”.

Mediante skim podemos observar cómo están distribuidos los datos, donde podemos notar que no hay missing values, por lo que para efectos del presente análisis no se empleará imputación de datos.

```
summary(data)
##      Gender      Height      Weight
## Length:10000   Min.   :54.26   Min.   : 64.7
## Class :character 1st Qu.:63.51   1st Qu.:135.8
## Mode  :character Median :66.32   Median :161.2
##              Mean  :66.37   Mean   :161.4
##              3rd Qu.:69.17   3rd Qu.:187.2
##              Max.   :79.00   Max.   :270.0
skim(data)
```

Data summary

Name data
 Number of rows 10000
 Number of columns 3

Column type frequency:



character 1
 numeric 2

Group variables None

Variable type: character

| skim_variable | n_missing | complete_rate | min | max | empty | n_unique | whitespace |
|---------------|-----------|---------------|-----|-----|-------|----------|------------|
| Gender | 0 | 1 | 4 | 6 | 0 | 2 | 0 |

Variable type: numeric

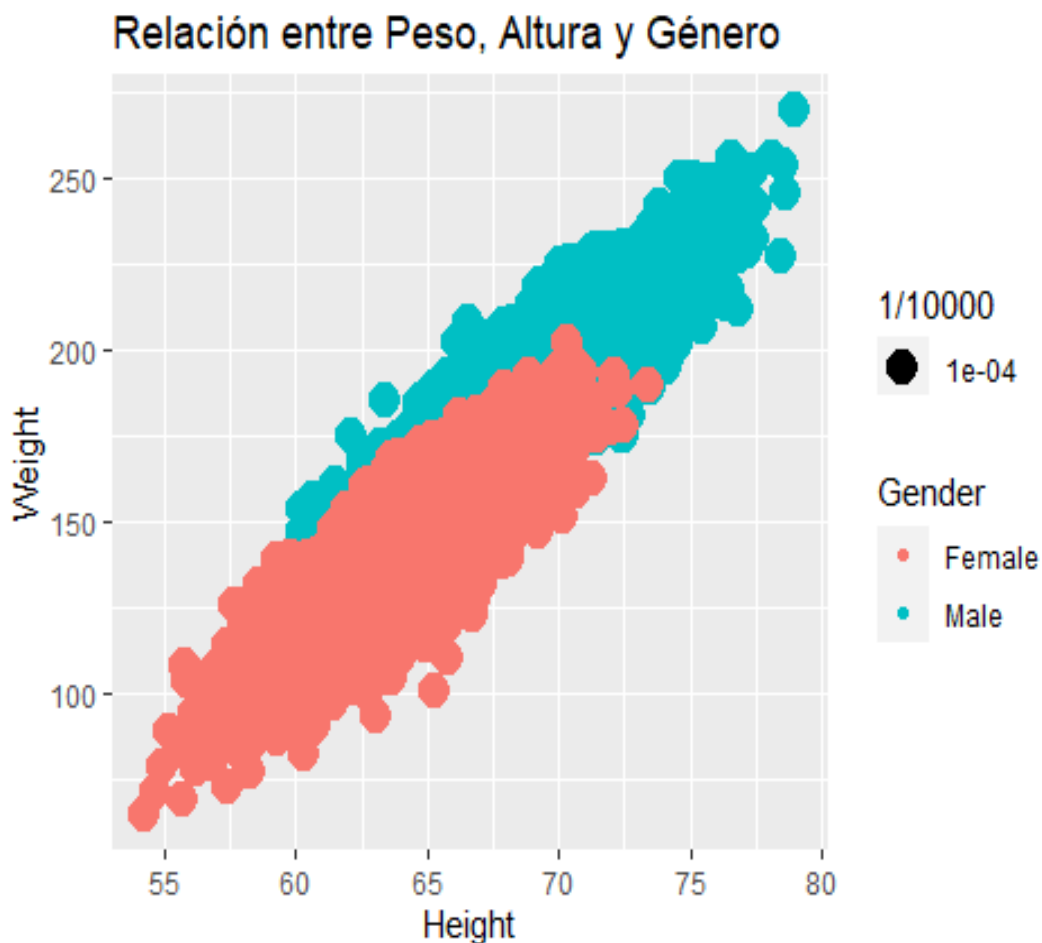
| skim_variable | n_missing | complete_rate | mean | sd | p0 | p25 | p50 | p75 | p100 | hist |
|---------------|-----------|---------------|--------|-------|-------|--------|--------|--------|--------|---|
| Height | 0 | 1 | 66.37 | 3.85 | 54.26 | 63.51 | 66.32 | 69.17 | 79.00 |  |
| Weight | 0 | 1 | 161.44 | 32.11 | 64.70 | 135.82 | 161.21 | 187.17 | 269.99 |  |

Al representar gráficamente los datos, se puede notar una relación entre el peso y la altura con el género.

El siguiente gráfico infiere el tener naturaleza lineal, la clasificación “Male” tiende a ubicarse mayormente en los valores más altos de “Height” y “Weight” la clasificación “Female” se centran en rangos más bajos. Lo cual respalda la elección de utilizar estas variables como predictores en nuestro modelo de regresión logística.

```
# Crear el gráfico con título
plot <- ggplot(data=data, aes(x=Height, y=Weight, col=Gender)) +
  geom_point(aes(size = 1/10000)) +
  xlab("Height") + ylab("Weight") +
  scale_color_discrete(name = "Gender") +
  labs(title = "Relación entre Peso, Altura y Género")

# Mostrar el gráfico
plot
```



Hipótesis de Regresión Logística

Dado que nuestro conjunto de datos cuenta con dos atributos: “Height” y “Weight”, definiremos la hipótesis de regresión logística de la siguiente manera:

- $h(x) = g(B_0 + B_1 \cdot \text{Height} + B_2 \cdot \text{Weight})$

donde,

- $h(x)$ representa la hipótesis de regresión logística.
- $g()$ es la función sigmoide, que transforma la entrada en un valor entre 0 y 1.

El umbral se establecerá en $g(z) = 0.5$, lo que significa que si el resultado de la hipótesis supera este valor, la predicción será clasificada como 1 o TRUE.

Es así como, el clasificador de regresión logística predecirá “Male” si la siguiente condición es verdadera (TRUE / 1):

- $B_0 + B_1 \cdot \text{Height} + B_2 \cdot \text{Weight} \geq 0$

Construcción del Modelo de Machine Learning

Ahora se procede a categorizar Gender, de lo cual se ve que R ha asignado 0 a femenino y 1 a masculino.

```
# Convertir "Gender" en un factor o categórico en lugar de numérico o cadena.
data$Gender <- as.factor(data$Gender)

# Ver las categorías llamando a la función contrasts()
contrasts(data$Gender)

##           Male
## Female      0
## Male        1
```

Con este conjunto de datos bien estructurado, avanzaremos a la fase de entrenamiento, explorando cómo la altura y el peso influyen en la clasificación de género y, finalmente, evaluando la capacidad predictiva del modelo resultante.

Training & Testing

Para verificar y probar el rendimiento de nuestro modelo, primero tenemos que dividir nuestros datos en conjuntos de entrenamiento y de prueba.

De la cual emplearemos la función “createDataPartition()” para dividir los datos en conjuntos separados. Aquí, dividimos el 60% de los datos para el entrenamiento y el 40% restante para las pruebas.

- createDataPartition() se encarga de dividir los datos de manera aleatoria, asegurando que la proporción de “Gender” se mantenga en ambos conjuntos “training” y “testing”.

```
# train es un vector que contiene los índices de las observaciones  
seleccionadas para formar el conjunto de entrenamiento de forma aleatoria  
train <- createDataPartition(y = data$Gender, p= .60, list = FALSE)  
  
# subset de entrenamiento  
training <- data[train,]  
  
# subset de prueba  
testing <- data[-train,]
```

Comprobaremos cuántas observaciones hay almacenadas en los conjuntos de entrenamiento y de prueba llamando a la función dim(). Como se mencionó anteriormente, se eligieron el 60% de los datos para el entrenamiento, esto se puede observar en las 6000 instancias para el dataframe ‘training’; y el restante 40% se dirigió al dataframe para la validación llamado ‘testing’, donde se ubican los 4000 datos restantes.

Para entrenamiento:

```
dim(training)  
## [1] 6000    3
```

Para Prueba:

```
dim(testing)  
## [1] 4000    3
```

Entrenamiento: Ajustar Modelo de Regresión Logística

Los datos para utilizar se ubican dentro del set 'training', que representa el 60% de los datos. Para implementar el modelo de regresión logística se creó una nueva variable 'logistic' en la cual se empleó el modelo con uso de la función 'glm' que relaciona Height y Weight con el factor Gender. Ambos atributos son predictores significativos evidenciado por la función 'summary' en la cual se muestra un resumen general del modelo..

Se puede observar en el resumen del modelo que ambas variables son significativas para el modelo (tienen un código de significancia de 0). A partir del análisis se obtiene un modelo de regresión logística en el que los 'Estimates' se presentan los coeficientes del modelo.

El modelo de regresión logística para el set de entrenamiento queda de la siguiente manera:

$$h(x) = g(13956 - (0.4999 \cdot \text{Height}) + (0.1972 \cdot \text{Weight}))$$

```
# Predict "Gender" using all predictors
logistic <- glm(Gender ~ Height+Weight, data=training, family= "binomial")

# Analysis of the model
summary(logistic)

##
## Call:
## glm(formula = Gender ~ Height + Weight, family = "binomial",
##      data = training)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  1.122431   1.745803   0.643    0.52
## Height      -0.504789   0.038178 -13.222 <2e-16 ***
## Weight       0.200821   0.006761  29.705 <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 8317.8  on 5999  degrees of freedom
## Residual deviance: 2474.8  on 5997  degrees of freedom
## AIC: 2480.8
##
## Number of Fisher Scoring iterations: 7
```

Analysis of the machine learning model: Testing

Ya tenemos un modelo ajustado de los datos para hacer predicciones. Ahora procederemos a responder la pregunta:

- ¿cómo se comporta nuestro modelo con los datos de prueba?

Lo haremos construyendo una matriz de confusión que muestre la tasa de éxito de las predicciones de nuestro modelo sobre los datos de prueba que creamos anteriormente.

Para esto, empleamos la función `predict()`, la cual realiza la predicción de “Gender” basándose en “Height” y “Weight” del conjunto de pruebas “testing”.

```
# Probabilidades de pertenecer a "Male"  
# El resultado se almacena en data.testing  
data.testing = predict(logistic, testing, type="response")
```

En este paso se crea un vector “data.predicha” que inicialmente está lleno con la etiqueta “Female”.

Luego, se actualizan las etiquetas a “Male” para aquellas observaciones donde la probabilidad predicha (`data.testing`) es mayor que 0.5, indicando que el modelo ha clasificado la observación como “Male”.

```
data.predicha = rep("Female", dim(training)[1])  
data.predicha[data.testing > .5] = "Male"
```


Finalmente, se utiliza la función `table` para construir la matriz de confusión comparando las predicciones (`data.predicha`) con las verdaderas etiquetas de género en el conjunto de entrenamiento (`training`).

La diagonal de la matriz de confusión representa el número de predicciones correctas, mientras que las celdas fuera de la diagonal indican las predicciones incorrectas. Es decir,

- **1067** observaciones se clasificaron correctamente como “Female”
- **1942** observaciones se clasificaron correctamente como “Male”
- **1933** observaciones se clasificaron incorrectamente como “Male”, pero en realidad eran “Female”. Estas son falsos positivos.
- **1058** observaciones se clasificaron incorrectamente como “Female”, pero en realidad eran “Male”. Estos son falsos negativos.

```
table(data.predicha, training$Gender)
```

```
##  
## data.predicha Female Male  
##      Female    1065 1064  
##      Male     1935 1936
```

Porcentaje de éxito de nuestras predicciones

La expresión `data.predicha == training$Gender`, crea un vector de valores lógicos (TRUE si la predicción es correcta, FALSE si es incorrecta).

La función `mean()` se utiliza para calcular el promedio de este vector de valores lógicos, lo que proporciona el porcentaje de éxito de las predicciones, siendo para nuestro modelo de entrenamiento de un 50.15%.

Esto es, alrededor de la mitad de las predicciones son precisas según las verdaderas etiquetas de género en el conjunto de entrenamiento. Es importante evaluar el modelo en un conjunto de prueba independiente para así obtener una estimación más realista del rendimiento.

```
mean(data.predicha == training$Gender)
## [1] 0.5001667
```

Tasa de error

El complemento del porcentaje de éxito, lo que da como resultado la tasa de error. El cual es de un 49.85%.

```
1 - mean(data.predicha == training$Gender)
## [1] 0.4998333
```

LR Model Dataset Completo

Ahora bien, procederemos a emplear el modelo de Regresión Logística para todo el conjunto de datos.

```
# Predecir Gender en función de todas las demás variables predictoras presentes
logistic <- glm(Gender ~ ., data=data, family= "binomial")
```

Del cual a continuación podemos ver un resumen estadístico. Al analizar el modelo de regresión lineal empleado en la totalidad de los datos, se puede observar que ambas variables son significativas para el modelo, al presentar un código de significancia de 0. A partir de este análisis se obtiene un modelo de regresión logística en el que los 'Estimates' representan los coeficientes del modelo.

Al presentar un AIC de valor alto, se sugiere que el modelo puede no ser el más adecuado para describir los datos. Cuando se comparan modelos con AIC, se prefiere el valor más bajo. Un valor de AIC más bajo indica que el modelo proporciona un buen ajuste a los datos con un número mínimo de parámetros.

```
# Analysis of the model
summary(logistic)

##
## Call:
## glm(formula = Gender ~ ., family = "binomial", data = data)
##
```

```
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.69254    1.32846   0.521   0.602
## Height      -0.49262    0.02896 -17.013 <2e-16 ***
## Weight       0.19834    0.00513  38.663 <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 13862.9  on 9999  degrees of freedom
## Residual deviance:  4182.6  on 9997  degrees of freedom
## AIC: 4188.6
##
## Number of Fisher Scoring iterations: 7
```

La función `coef(logistic)` se utiliza para obtener los coeficientes estimados del modelo. Estos coeficientes representan la magnitud y dirección de la relación entre cada variable predictora y la probabilidad de pertenecer a la categoría “Male”.

De esta manera, el modelo de regresión logística para el dataset de completo queda de la siguiente manera:

$$h(x) = g(0.6925431 - (0.4926200 \cdot \text{Height}) + (0.1983404 \cdot \text{Weight}))$$

```
# Imprimir coeficientes (valores estimados)
```

```
cat("Coeficientes estimados:\n")
```

```
## Coeficientes estimados:
```

```
print(coef(logistic))
```

```
## (Intercept)      Height      Weight
##   0.6925431   -0.4926200    0.1983404
```

Gráfica del modelo obtenido

Ahora generaremos una gráfica que visualiza el modelo de regresión logística en función de las variables “Height” y “Weight”.

Mediante la función “expand.grid” crearemos un conjunto de datos llamado “plot_data”, que cubre todo el rango de valores observados en las instancias “Height” y “Weight” del conjunto de datos original “data”.

```
plot_data <- expand.grid(  
  Height = seq(min(data$Height), max(data$Height)),  
  Weight = seq(min(data$Weight), max(data$Weight))  
)
```

Ahora realizaremos la predicción de probabilidades utilizando el modelo de regresión logística ajustado.

Se predice la probabilidad de pertenecer a la clase “Male” para cada combinación de instancias de Height y Weight en el conjunto de datos plot_data.

Esta información se almacena en la columna “Probability” de “plot_data”.

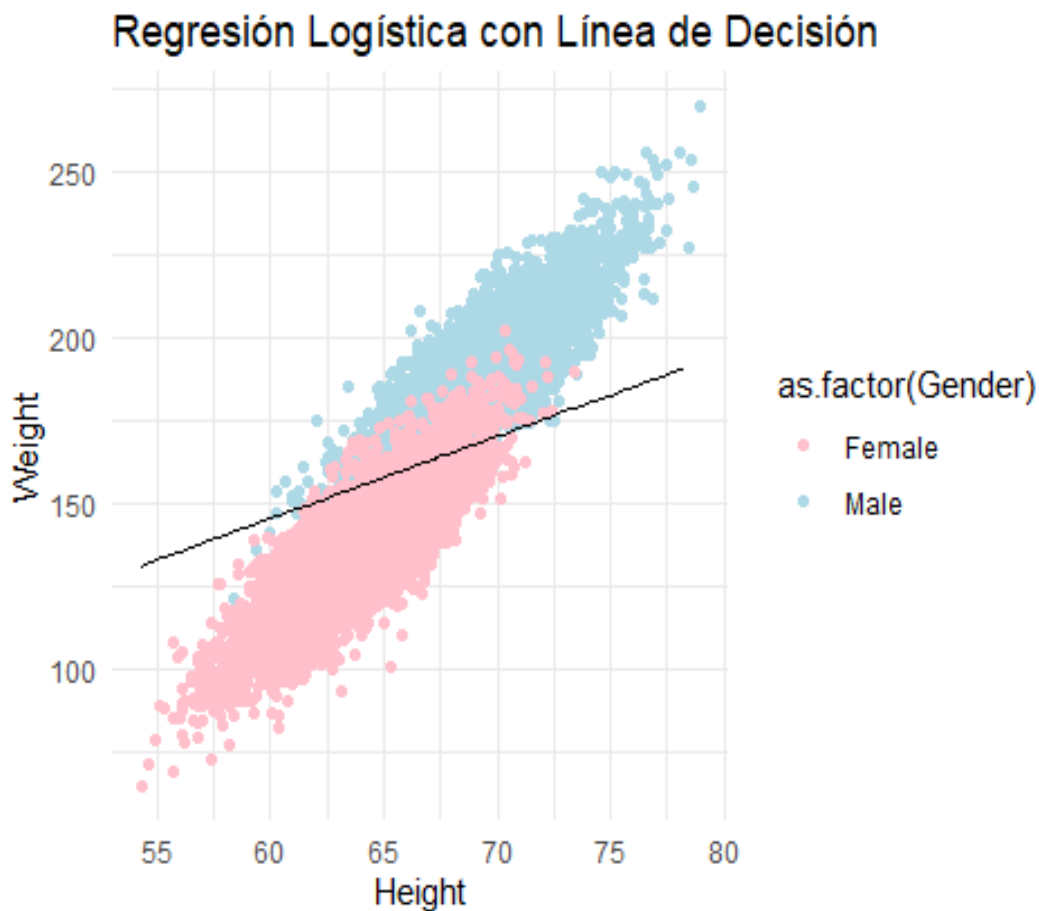
Esta columna de probabilidades posteriormente se utiliza en la creación de la gráfica para visualizar cómo el modelo de regresión logística clasifica las observaciones en función de las variables Height y Weight.

En la gráfica final, la función geom_contour utiliza estas probabilidades para trazar líneas de contorno, lo que representa visualmente la región donde el modelo asigna una mayor probabilidad de pertenecer a la clase “Male”. La línea de contorno puede interpretarse como la línea de decisión del modelo, donde por encima de la línea se predice la clase positiva y por debajo se predice la clase negativa.

Además, en este gráfico que representa el modelo de regresión logística se puede observar la razón de la tasa de error. Es decir, se puede ver claramente que los datos referentes a cada variable (male y female) sobrepasan la línea de decisión.

```
plot_data$Probability <- predict(logistic, newdata=plot_data,  
type="response")  
  
# Crear la gráfica  
ggplot(data, aes(x = Height, y = Weight, color = as.factor(Gender))) +  
  geom_point() +  
  geom_contour(data = plot_data, aes(z = Probability), color = "black", bins  
= 2) +  
  scale_color_manual(values = c("pink", "lightblue")) +  
  labs(title = "Regresión Logística con Línea de Decisión") +  
  theme_minimal()
```

```
View(plot_data)
```



Let's see how well the model does in predicting Gender

Para esta sección de la tarea, se creó un nuevo dataframe empleando los valores a evaluar propuestos por el profesor.

Utilizando este nuevo dataframe, se predice el género de cada conjunto de datos por medio de regresión logística. Estas predicciones posteriormente se utilizan para clasificar los conjuntos de datos en 'male' si la probabilidad es mayor a 0.5, y en 'female' si es menor a este valor. Posteriormente, se implementan los resultados en un nuevo dataset demostrado en la parte inferior.

```
predicciones <- data.frame(Height = c(68, 62, 70, 80, 45), Weight = c(175,
130, 130, 190, 200))

# Predecir Las probabilidades de género para Los nuevos datos
probabilidades <- predict(logistic, newdata = predicciones, type =
"response")

# Clasificar en función de Las probabilidades (por ejemplo, usando un umbral
del 0.5)
clasificacion <- ifelse(probabilidades >= 0.5, "Male", "Female")

# Crear un dataframe con Los resultados
resultados <- data.frame(Altura = predicciones$Height, Peso =
predicciones$Weight, Genero_Predicho = clasificacion, Probabilidad =
probabilidades)

# Imprimir Los resultados
print(resultados)

##   Altura  Peso  Genero_Predicho  Probabilidad
## 1     68   175             Male 0.8703385790
## 2     62   130             Female 0.0168627255
## 3     70   130             Female 0.0003331437
## 4     80   190             Female 0.2626233028
## 5     45   200             Male 0.9999999874
```

Para el análisis de cada uno de estos casos, es importante recalcar que las probabilidades son basadas en la clasificación del hombre, es decir, una alta probabilidad indica que los datos son clasificados como 'male'; y una baja probabilidad indica que los datos son categorizados como 'female'.

- En el primer caso, cuando una persona tiene una altura de 68 pulgadas y un peso de 175 libras, existe una probabilidad de 87% (probabilidad alta) de que es un hombre.
- Cuando una persona tiene una altura de 62 pulgadas y un peso de 62 libras, existe una probabilidad de 0.02% (probabilidad baja) de que es un hombre, por lo tanto, se categriza como mujer.
- Cuando una persona tiene una altura de 70 pulgadas y un peso de 70 libras, existe una probabilidad de 0.0003% (probabilidad baja) de que es un hombre, por lo tanto, se categriza como mujer.
- Cuando una persona tiene una altura de 80 pulgadas y un peso de 80 libras, existe una probabilidad de 0.26% (probabilidad baja) de que es un hombre, por lo tanto, se categriza como mujer.
- Cuando una persona tiene una altura de 45 pulgadas y un peso de 45 libras, existe una probabilidad de 99% (probabilidad alta) de que es un hombre.

CONCLUSIÓN

En conclusión, se recapitulará brevemente lo realizado a lo largo de esta actividad. En primera instancia, se analizaron brevemente los datos con el fin de asegurarnos de que no existieran missing values en el dataset. Posteriormente, se declaró el atributo 'Género' como un factor, y se establecieron los datos correspondientes al set de entrenamiento (60% de los datos y validación (40% de los datos) con el fin de probar el buen funcionamiento del modelo de regresión logística. Aunque el porcentaje de error del modelo de regresión muestra un porcentaje de 50%, decidimos proceder con el modelo ya que los datos originales se empalman y sobre pasan la línea de referencia establecida por el modelo que separa ambos géneros; por lo tanto este modelo no es el mejor, pero se podría mejorar por medio de la implementación de más atributos en relación a la variable de salida. Sin embargo, procedimos con el modelo, y aunque las predicciones no son perfectas, dan un resultado favorable.