Regresión Logística - diabetes

D. S. Fernández del Viso

2024-12-20

#### paquetes

# Cargar paquetes  
library(tidyverse)

## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──  
## ✔ dplyr 1.1.4 ✔ readr 2.1.5  
## ✔ forcats 1.0.0 ✔ stringr 1.5.1  
## ✔ ggplot2 3.5.1 ✔ tibble 3.2.1  
## ✔ lubridate 1.9.3 ✔ tidyr 1.3.1  
## ✔ purrr 1.0.2   
## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::lag() masks stats::lag()  
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors

## Regresión Logística

### Cargar datos

# Cargar datos desde archivo .csv  
diabetes <- read.csv("diabetes.csv")  
head(diabetes)

## Pregnancies Glucose BloodPressure SkinThickness Insulin BMI  
## 1 6 148 72 35 0 33.6  
## 2 1 85 66 29 0 26.6  
## 3 8 183 64 0 0 23.3  
## 4 1 89 66 23 94 28.1  
## 5 0 137 40 35 168 43.1  
## 6 5 116 74 0 0 25.6  
## DiabetesPedigreeFunction Age Outcome  
## 1 0.627 50 1  
## 2 0.351 31 0  
## 3 0.672 32 1  
## 4 0.167 21 0  
## 5 2.288 33 1  
## 6 0.201 30 0

### Filtrar datos para no tener valores de 0 en las variables predictoras

# Filtrar datos para no tener valores de 0 en las variables predictoras  
diabetes\_filt <- diabetes %>%  
 filter(Glucose != 0 & BloodPressure != 0 & SkinThickness != 0 & BMI != 0 & Insulin != 0)

### Crear un modelo de regresión logística

# Crear un modelo de regresión logística  
modelo <- glm(Outcome ~ Glucose + BloodPressure + SkinThickness + BMI + Insulin,   
 data = diabetes\_filt, family = binomial)  
summary(modelo)

##   
## Call:  
## glm(formula = Outcome ~ Glucose + BloodPressure + SkinThickness +   
## BMI + Insulin, family = binomial, data = diabetes\_filt)  
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -8.8117401 1.1086601 -7.948 1.89e-15 \*\*\*  
## Glucose 0.0413484 0.0056883 7.269 3.62e-13 \*\*\*  
## BloodPressure 0.0089356 0.0110292 0.810 0.418   
## SkinThickness 0.0258049 0.0163536 1.578 0.115   
## BMI 0.0453530 0.0255624 1.774 0.076 .   
## Insulin -0.0006312 0.0012937 -0.488 0.626   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 498.10 on 391 degrees of freedom  
## Residual deviance: 368.48 on 386 degrees of freedom  
## AIC: 380.48  
##   
## Number of Fisher Scoring iterations: 5

### Calcular los odds ratios

# Calcular los odds ratios con intervalos de confianza  
exp(cbind(OR = coef(modelo), confint(modelo)))

## Waiting for profiling to be done...

## OR 2.5 % 97.5 %  
## (Intercept) 0.0001489738 1.522795e-05 0.001189458  
## Glucose 1.0422151177 1.031088e+00 1.054408759  
## BloodPressure 1.0089756471 9.875917e-01 1.031368052  
## SkinThickness 1.0261407265 9.937361e-01 1.059718598  
## BMI 1.0463971494 9.958267e-01 1.101144012  
## Insulin 0.9993690417 9.968442e-01 1.001936703

# Generate predicted probabilities  
diabetes\_filt$predicted\_prob <- predict(modelo, type = "response")  
  
# Plot the data with the logistic regression curve  
ggplot(diabetes, aes(x = Glucose, y = Outcome)) +  
 geom\_point(alpha = 0.5) + # Scatter plot of actual data  
 geom\_smooth(method = "glm", method.args = list(family = "binomial"), se = TRUE, color = "blue") +  
 labs(title = "Logistic Regression with Smooth Line",  
 x = "Glucose",  
 y = "Probability of Outcome") +  
 theme\_minimal()

## `geom\_smooth()` using formula = 'y ~ x'

