Solución PRA2 – ¿Cómo realizar la limpieza y análisis de datos?

Tipología y Ciclo de Vida de los Datos

David Esparcia

Rubén Revuelta

ÍNDICE

1. Descripción del dataset

El conjunto de datos recogido en Kaggle pretende recoger la información necesaria para poder predecir y anticipar posibles ataques al corazón. En concreto, el conjunto está compuesto por un total de 303 registros y 14 atributos divididos en 13 características y 1 variable objetivo.

```
dim(data)
```

```
[1] 303  14
```

A continuación, se procede con la descripción de los distintos atributos que forman el conjunto de datos a estudiar.

- **Age**: edad del paciente sobre el que se recogen los datos.
- **Sex**: sexo del paciente sobre el que se recogen los datos.
- **Cp:** tipo de dolor de pecho. Se trata de una variable categórica que puede tomar los siguientes valores.
  - 0 = Angina típica.
  - 1 = Angina atípica.
  - 2 = Dolor no asociado a una angina.
  - 3 = Asintomático.
- **Trtbps**: presión arterial en estado de reposo (medida en mm Hg).
- **Chol**: Colesterol medido en mg/dl obtenido a través del sensor IMC.
- **Fbs**: Indica la existencia de glucemia en ayunas, ligado a posibles indicios de diabetes entre otros. Tomará un valor u otro en función de si es mayor a 120 mg/dl.
  - 1 = Verdadero
  - 0 = Falso
- **Rest_ecg:** Resultados del electrocardiograma en reposo.
  - 0 = Normal.
  - 1 = Anormalidad en la onda ST-T.
  - 2 = Hipertrofia ventricular izquierda.
- **Thalachh:** Frecuencia cardiaca máxima alcanzada.
- **Oldpeak:** Pico anterior.
- **Slp:** pendiente del segmento ST mostrada en el electrocardiograma. Puede tomar tres valores.
  - Ascendente
  - Plana
  - Descendente
- **Thall:** Resultado del test de estrés de Thalium.
- **Exng:** Indica si el ejercicio induce la angina.
  - 1 = Sí.
  - 0 = No.
- **Caa:** número de vasos principales, de 0 a 3.
- **Target:** variable objetivo del análisis, indica de forma categórica la probabilidad de sufrir un ataque al corazón.
  - 0 = menor probabilidad de sufrir un ataque al corazón.
  - 1 = mayor probabilidad de sufrir un ataque al corazón.

En resumen, el conjunto de datos dispone de un total de **5 variables continuas** y **9 variables categóricas** distribuidas de la siguiente forma:

- **Variables continuas:** age, trtbps, chol, thalachh, oldpeak.
- **Variables categóricas:** sex, cp, fbs, restecg, exng, slp, caa, thall, output.

2. Integración y selección

La realidad es que, en primera instancia, sin realizar ningún tipo de análisis previo como pudiera ser la distribución de la varianza o análisis de correlación, todas las variables pueden ser de interés al presentar cierta relación con el objetivo de estudio.

3. Limpieza de los datos

3.1. Valores nulos, vacíos y duplicados.

La primera acción de limpieza de datos que se lleva a cabo es la detección de valores nulos o vacíos en el conjunto de datos. Comenzamos buscando por valores nulos.

```
#Existencia de registros con valores nulos en alguno de sus atributos
colSums(is.na(data))
```

```
     age      sex       cp   trtbps     chol      fbs  restecg thalachh     exng
oldpeak      slp      caa    thall   output
       0        0        0        0        0        0        0        0        0
0        0        0        0        0
```

Tras observar que el conjunto de datos no dispone de valores nulos, buscamos registros cuyos atributos contengan valores vacíos.

```
#Existencia de registros con valores vacíos en alguno de sus atributos
colSums(data=="")
```

```
     age      sex       cp   trtbps     chol      fbs  restecg thalachh     exng
oldpeak      slp      caa    thall   output
       0        0        0        0        0        0        0        0        0
0        0        0        0        0
```

De igual forma, no se encuentran registros que presenten este tipo de valores. En tercer lugar, se realiza una búsqueda de posibles registros duplicados. Esto es posible a través de la función *duplicated()* presente en R.

```
duplicated(data)
```

```
  [1] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
 FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
 [22] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
 FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
 [43] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
 FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
 [64] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
 FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
 [85] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
 FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[106] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
 FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[127] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
 FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[148] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
 FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE
[169] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
 FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[190] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
 FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[211] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
 FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[232] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
 FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
```

Esta vez, la prueba realizada nos indica que si existe un valor duplicado por lo que habría que proceder a su eliminación. Para eliminar el valor duplicado utilizamos la función *distinct()* de la librería *dplyr*.

```
data <- data %>% distinct()
dim(data)
```

```
[1] 302  14
```

Tras el borrado de duplicados se observa como el conjunto de datos pasa de tener 303 registros a tener uno menos, 302 registros.

3.2. Gestión de *outliers*

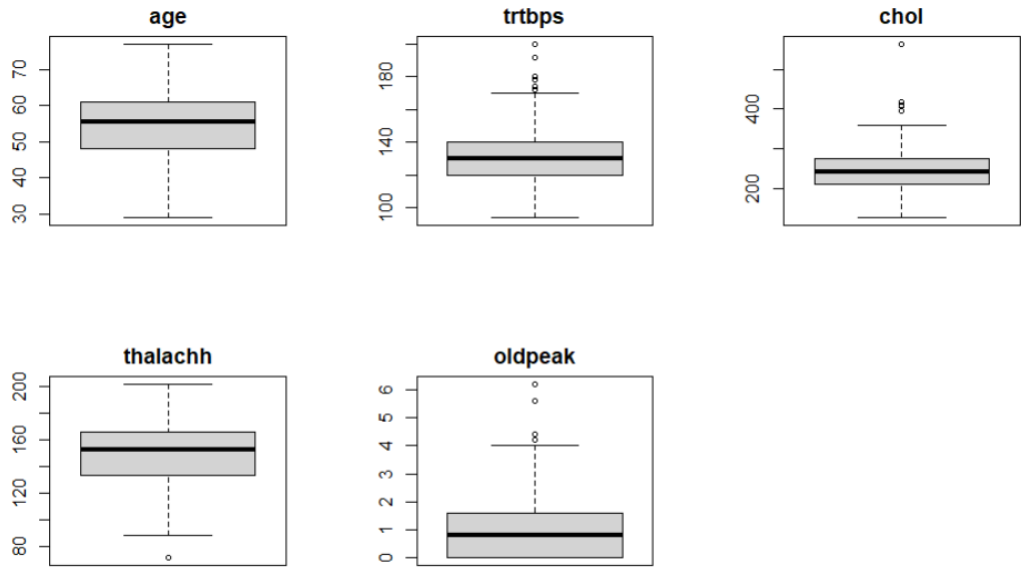
Una primera aproximación para la detección de outliers en nuestras variables continuas la podemos realizar aplicando la función *summary()* a nuestro conjunto de datos.

```
variables_continuas <- data[,c("age","trtbps","chol","thalachh","oldpeak")]
summary(variables_continuas)
```

```
      age            trtbps           chol         thalachh        oldpeak      
 Min.   :29.00   Min.   : 94.0   Min.   :126.0   Min.   : 71.0   Min.   :0.000  
 1st Qu.:48.00   1st Qu.:120.0   1st Qu.:211.0   1st Qu.:133.2   1st Qu.:0.000  
 Median :55.50   Median :130.0   Median :240.5   Median :152.5   Median :0.800  
 Mean   :54.42   Mean   :131.6   Mean   :246.5   Mean   :149.6   Mean   :1.043  
 3rd Qu.:61.00   3rd Qu.:140.0   3rd Qu.:274.8   3rd Qu.:166.0   3rd Qu.:1.600  
 Max.   :77.00   Max.   :200.0   Max.   :564.0   Max.   :202.0   Max.   :6.200  
```

Sin embargo, la forma más eficiente de detectar posibles outliers en nuestras variables continuas es a través de box-plots.

```
par(mfrow=c(2,3))
for(i in 1:ncol(variables_continuas)) {
  boxplot(variables_continuas[,i], main = colnames(variables_continuas)[i], width =
100)
}
```



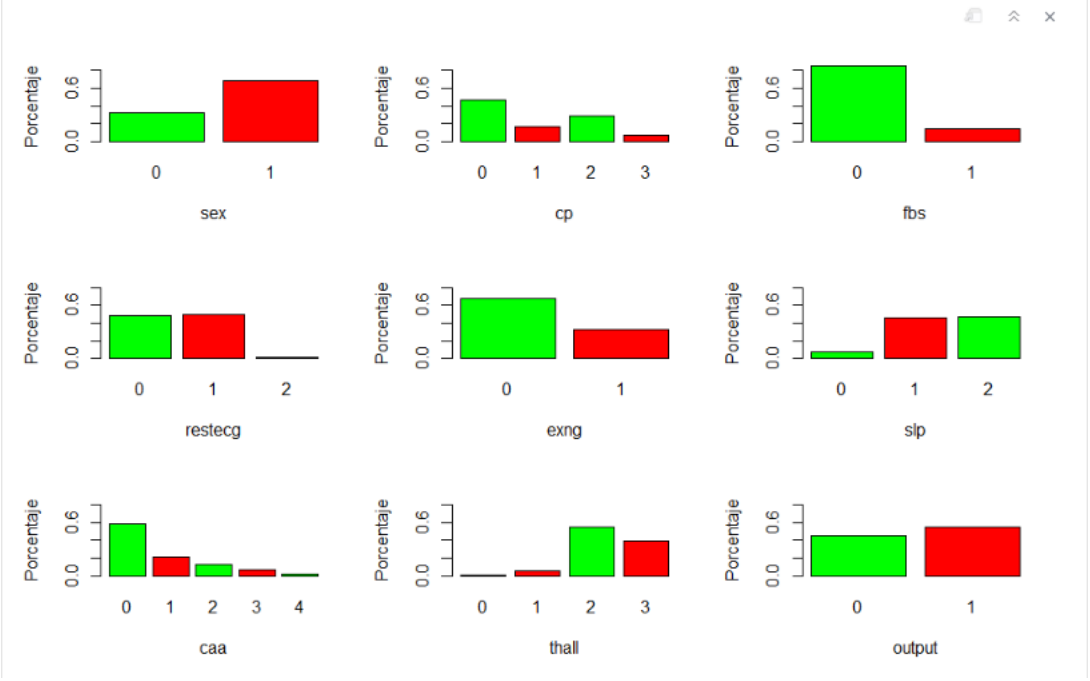
A través de los diagramas de cajas observamos ciertos candidatos a *outliers* en algunas de las variables como pueden ser *chol o trtbps*. Sin embargo, analizando la naturalidad de estas y entendiendo el problema e información que representan, se determina que se tratan de valores que pueden ser perfectamente válidos. Por ejemplo, un colesterol mayor de 500 se considera valores altos de colesterol.

Apoyando este análisis, a través de la comparación con la media, ninguno de los posibles *outliers* llegaría si quiera a ser *fringelie* al no superar 3 veces la desviación estándar de la media.

Por otro lado, la detección de valores anómalos en las variables categóricas la podemos llevar a cabo a través de su representación por medio de histogramas.

```
variables_categoricas <- data[, c("sex", "cp", "fbs", "restecg", "exng", "slp",
"caa", "thall", "output")]

par(mfrow=c(3,3))
for(i in 1:ncol(variables_categoricas)) {
  counts <- table(variables_categoricas[,i])
  barplot(prop.table(counts),col=c("green","red"), ,xlab =colnames
(variables_categoricas)[i], ylab = "Porcentaje",ylim=c(0,0.8))
}
```



Como observamos, tras comprobar todos los histogramas, todas las variables categóricas toman valores dentro del rango para el cual se encuentran definidas.

4. Análisis de los datos

Como anteriormente hemos visto una visión general del dataset con las funciones summary y str, en este apartado nos centraremos en un análisis inferencial mediante la comprobación de grupos y de la normalidad utilizado diferentes pruebas y métodos analíticos.

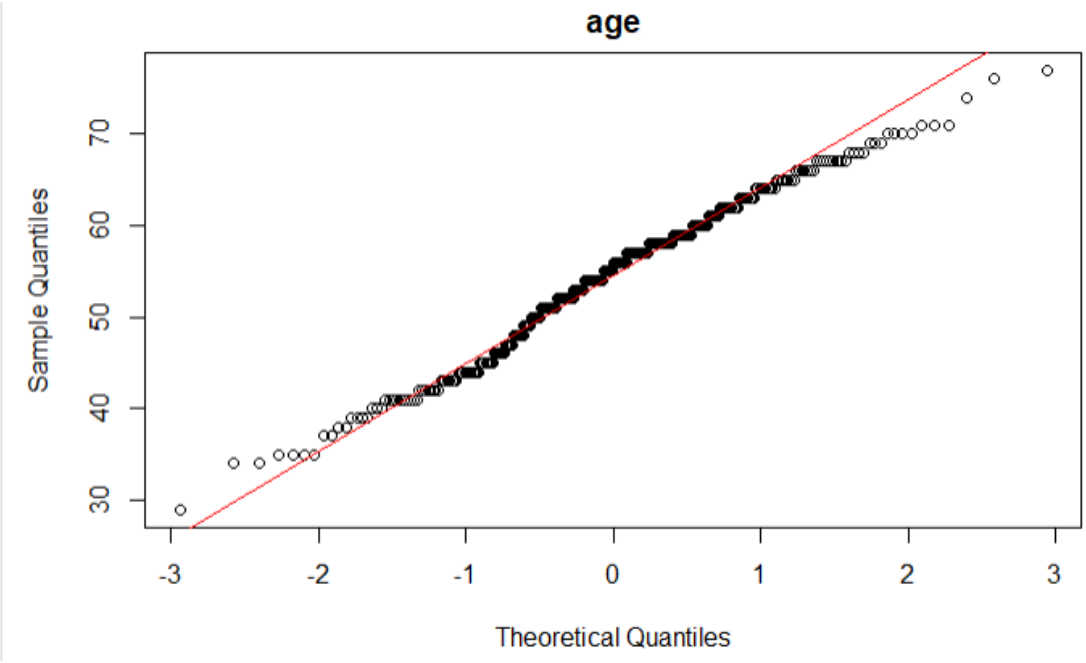
4.1. Selección de grupos a analizar y comparar

A continuación, seleccionamos diferentes grupos a analizar y comparar, aplicando diferentes tests en función del tipo de variable a tratar.

4.2. Comprobación de la normalidad y homogeneidad de la varianza

Primero comprobamos la normalidad de las variables continuas mediante el test de Shapiro-Wilk y un gráfico QQ plot, para ver visualmente si los resultados del test nos cuadran con la distribución que sigue cada variable.

```
```{r ,eval=TRUE,echo=TRUE}
qqnorm(variables_continuas$age, main='age')
qqline(variables_continuas$age, col="red")
shapiro.test(variables_continuas$age)
```
```




```
        Shapiro-Wilk normality test

data:  variables_continuas$age
W = 0.98664, p-value = 0.006745
```

Adjuntamos solo un ejemplo, aunque hemos hecho el mismo proceso para todas las variables numéricas.

Vemos que en todos los casos se rechaza la hipótesis nula de normalidad en la distribución, ya que todos los p-value son menores al nivel de significancia asumido (α=0,05).

Sería posible normalizar todos los datos, sin embargo, creo que es interesante tener una visión real de algunas variables como el colesterol o la edad. Teniendo en cuenta que el número de observaciones es mayor a 30 y que en la mayoría de los gráficos

QQ plot tampoco vemos una gran desviación, podríamos asumir que la distribución es normal tal y como dice el teorema central del límite.

A continuación, vamos a estudiar la homogeneidad de las varianzas para las variables *chol* y *cp*, que corresponden al colesterol y al tipo de dolor en el pecho, utilizando un test de Fligner-Killeen con una hipótesis nula en la cual ambas varianzas son iguales.

```{r, eval=TRUE, echo=TRUE}
fligner.test(output ~ chol, data = data)
```

```
        Fligner-Killeen test of homogeneity of variances

data:  output by chol
Fligner-Killeen:med chi-squared = 146.09, df = 151, p-value = 0.5977
```

```{r, eval=TRUE, echo=TRUE}
fligner.test(output ~ cp, data = data)
```

```
        Fligner-Killeen test of homogeneity of variances

data:  output by cp
Fligner-Killeen:med chi-squared = 2.7653, df = 3, p-value = 0.4292
```

En ninguno de los dos casos se rechaza la hipótesis nula ya que el p-valor es superior a 0.05, por lo que podemos decir que las varianzas son homogéneas.

4.3. Aplicación de pruebas estadísticas para comparar los grupos de datos

Teniendo en cuenta que el objetivo del estudio es verificar que variables tienen más influencia a la hora de provocar un ataque al corazón, vamos a realizar diferentes pruebas estadísticas que nos permitan sacar conclusiones del dataset.

La primera prueba que vamos a aplicar es el test chi2 para ver si existen diferencias significativas a la hora de tener un ataque al corazón entre hombres y mujeres.

Este tipo de prueba es útil para comprobar si hay diferencias significativas entre variables categóricas.

```
```{r, eval=TRUE, echo=TRUE}
sex.result = table(data$output, data$sex)
colnames(sex.result) <- c('Mujer', 'Hombre')
rownames(sex.result) <- c('No', 'Si')
sex.result
chisq.test(sex.result)

```
```

```
   Mujer Hombre
No    24    114
Si    72     92

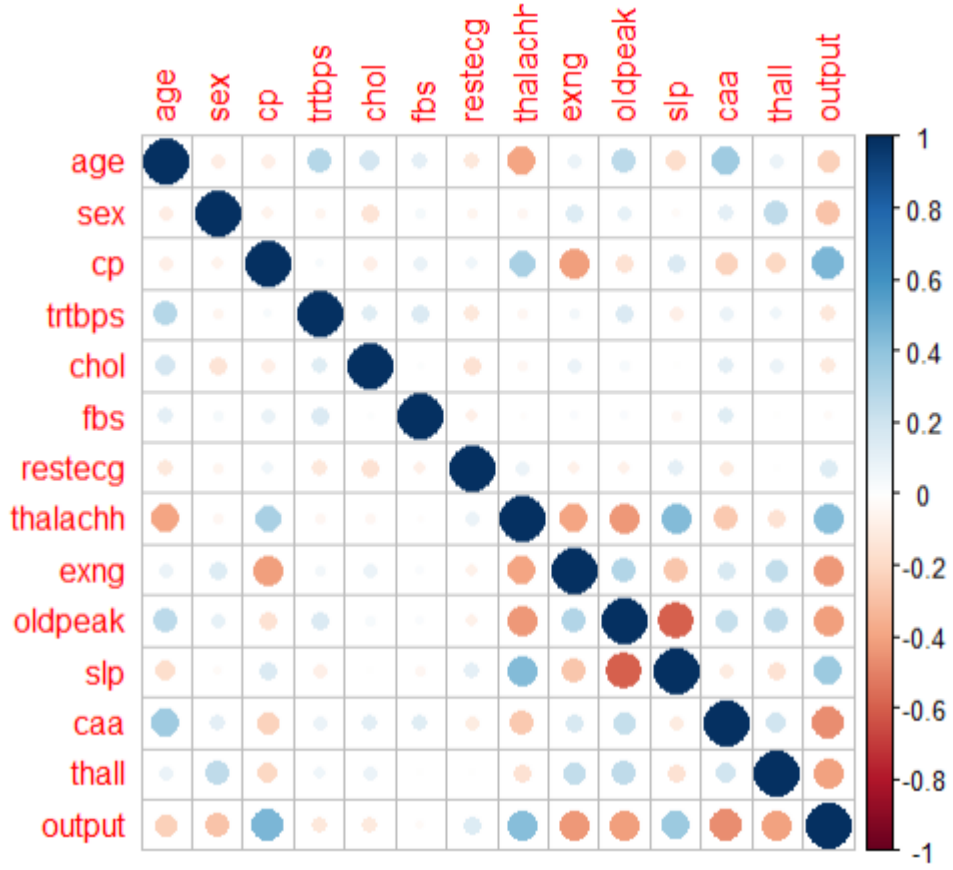
        Pearson's Chi-squared test with Yates' continuity correction

data:  sex.result
X-squared = 23.084, df = 1, p-value = 1.551e-06
```

El p-value es menor a 0.05 y por tanto podemos decir que existen diferencias significativas entre hombres y mujeres.

Seguidamente vamos a ver las correlaciones entre las diversas variables para poder identificar cuales pueden estar más correlacionadas con el hecho de padecer un ataque al corazón. Como anteriormente hemos visto que los datos en general no seguían una distribución normal, aplicamos el test de Spearman para calcular las correlaciones entre pares de variables.

```
```{r, eval=TRUE, echo=TRUE}
# Primero vemos una representación gráfica de las correlaciones
library(corrplot)
corr.resultados <- cor(data, method='spearman')
corrplot(corr.resultados,method='circle')
```
```



Podemos ver que las variables más correlacionadas con sufrir un ataque al corazón (output) son:

- Cp: tipo de dolor de pecho
- Thalachh: frecuencia cardíaca máxima
- Slp: pendiente del segmento ST
- Exng: angina inducida por ejercicio
- Oldpeak: alteración del segmento ST
- Caa: número de vasos principales

Teniendo esto en cuenta, la última prueba estadística a realizar va a ser un modelo de regresión logística ya que estamos tratando de explicar las variables que más afectan al resultado de una variable categórica (*output*).

Para empezar en el modelo hemos incluido las variables con más correlación y podemos ver que todas son estadísticamente significativas, ya que el PR(>|z1) es más pequeño que 0.05.

```
```{r, eval=TRUE, echo=TRUE}
m1 = glm(formula = output~sex+cp+thalachh+exng+oldpeak+caa+thall, family='binomial', data=data)
summary(m1)
```
```

```
Call:
glm(formula = output ~ sex + cp + thalachh + exng + oldpeak +
    caa + thall, family = "binomial", data = data)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.3934  -0.4499   0.1940   0.5770   2.4817

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  0.56332    1.48757   0.379 0.704921
sex         -1.39706    0.40569  -3.444 0.000574 ***
cp           0.77521    0.17488   4.433 9.30e-06 ***
thalachh     0.02301    0.00883   2.606 0.009174 **
exng        -1.04746    0.38976  -2.687 0.007200 **
oldpeak     -0.72752    0.18284  -3.979 6.92e-05 ***
caa         -0.76579    0.18400  -4.162 3.16e-05 ***
thall       -0.88520    0.27507  -3.218 0.001290 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 416.42  on 301  degrees of freedom
Residual deviance: 221.91  on 294  degrees of freedom
AIC: 237.91

Number of Fisher Scoring iterations: 5
```

Para comprobar como de bueno es el ajuste del modelo, tenemos que fijarnos en el coeficiente de Akaike (AIC). CUanto menor sea el valor mejor será el modelo.

Para verificar que estamos incluyendo las variables adecuadas, vamos a probar a añadir alguna variable más y ver cómo se comporta el nuevo modelo.

```
```{r, eval=TRUE, echo=TRUE}
m2 = glm(formula = output~age+sex+cp+thalachh+exng+oldpeak+slp+caa+thall, family='binomial', data=data)
summary(m2)
```
```

```
Call:
glm(formula = output ~ age + sex + cp + thalachh + exng + oldpeak +
    slp + caa + thall, family = "binomial", data = data)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.4276  -0.4473   0.1883   0.5783   2.6082

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  1.661322   2.308555   0.720 0.471748
age         -0.019873   0.021842  -0.910 0.362903
sex         -1.543728   0.425730  -3.626 0.000288 ***
cp           0.810367   0.178262   4.546 5.47e-06 ***
thalachh     0.016981   0.009664   1.757 0.078877 .
exng        -1.021593   0.394322  -2.591 0.009577 **
oldpeak     -0.561566   0.205442  -2.733 0.006267 **
slp          0.584345   0.339234   1.723 0.084971 .
caa         -0.784304   0.194820  -4.026 5.68e-05 ***
thall       -0.887433   0.274246  -3.236 0.001213 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 416.42  on 301  degrees of freedom
Residual deviance: 218.14  on 292  degrees of freedom
AIC: 238.14

Number of Fisher Scoring iterations: 6
```

Vemos que pese a no haber mucha diferencia en el valor de AIC, en este modelo hay variables que no son estadísticamente significativas por lo que no nos servirían para explicar la variable dependiente.

5. Representación de los resultados

Hemos ido incluyendo las diversas tablas y gráficos a lo largo de la práctica.

6. Resolución del problema

Como conclusión, y según lo visto en los resultados, podríamos extraer que las variables que más ayudan a explicar la posibilidad de tener un ataque al corazón serían:

- Sexo: en este dataset hemos visto que las mujeres tienen más posibilidades de padecer un ataque al corazón
- CP: tipo de dolor de pecho
- Frecuencia cardíaca máxima
- Angina inducida por el ejercicio
- Alteraciones en el segmento ST
- El número de vasos sanguíneos principales (0-3) coloreados por fluoroscopia
- Tipo de corazón (normal o con problemas anteriores)

Parece que los resultados obtenidos tienen lógica y entrar dentro de las variables que podríamos imaginar que ayudan a explicar la posibilidad de ataques al corazón.

Sin embargo, no estamos muy seguros de que los resultados permitan responder al problema ya que hemos visto que no existía mucha diferencia entre el modelo elegido y otros que incluían más variables, por lo que pensamos que quizá harían falta más observaciones y variables para poder extraer conclusiones más concretas.

7. Participación

Contribuciones	Firma
Investigación previa	D.E / R.R
Redacción de las respuestas	D.E / R.R
Desarrollo del código	D.E / R.R
Participación en el vídeo	D.E / R.R