

Práctica 2: ¿Cómo realizar la limpieza y análisis de datos?

Tipología y ciclo de vida de los datos

Autores: Ouassim Aouattah Akandouch, Juan Andrés Dávila

Enero 2023

Contents

Descripción del dataset ¿Por qué es importante y qué pregunta/problema pretende responder?	2
Objetivos	2
Integración y selección	3
Limpieza de los datos	4
Valores nulos o vacíos	4
Valores extremos	5
Análisis de los datos	9
Selección de los grupos de datos que se quieren analizar/comparar	9
Comprobación de la normalidad y homogeneidad de la varianza	10
Aplicación de pruebas estadísticas para comparar los grupos de datos.	12
5. Representación de los resultados a partir de tablas y gráficas.	20
6. Resolución del problema. A partir de los resultados obtenidos,	21
Conclusiones	21
7 Código.	21
8 Video.	21
Bibliografía	21

Descripción del dataset ¿Por qué es importante y qué pregunta/problema pretende responder?

Este dataset, llamado *Heart Attack Analysis & Prediction Dataset*, contiene información sobre pacientes que han sufrido o podrían sufrir un ataque al corazón. Incluye atributos como la edad y el sexo del paciente, si ha experimentado angina inducida por ejercicio, el número de vasos principales, el tipo de dolor en el pecho, la presión arterial en reposo, el colesterol, los resultados electrocardiográficos en reposo, la frecuencia cardíaca máxima lograda, y finalmente si tiene más o menos probabilidades de sufrir un ataque al corazón.

Este dataset es importante ya que el ataque al corazón es una de las principales causas de mortalidad a nivel mundial. El análisis de estos datos puede ayudar a identificar patrones y factores de riesgo relacionados con los ataques cardíacos, lo que permitiría a los médicos y científicos de la salud tomar medidas preventivas y mejorar los tratamientos. Se puede responder preguntas como ¿Qué factor o variables son los de mayor incidencia cuando una persona sufre de un ataque al corazón? o ¿Las mujeres son más propensas a tener este tipo de enfermedades? ¿es el colesterol un factor más determinante para un ataque cardíaco que la edad de la persona?

Revisamos la descripción de las variables contenidas en el fichero y los tipos de cada una y a continuación Construimos un pequeño diccionario de datos utilizando la documentación auxiliar.

- **target**: Probabilidades de sufrir un ataque al corazón (0 = menor probabilidad, 1 = mayor probabilidad).
- **Age**: Edad del paciente (en años).
- **trtbps**: Presión arterial en reposo (en mm Hg).
- **cp**: Tipo de dolor en el pecho.
 - Valor 1: Angina típica.
 - Valor 2: Angina atípica.
 - Valor 3: Dolor no anginal.
 - Valor 4: Asintomático.
- **thalach**: Frecuencia cardíaca máxima alcanzada (en latidos por minuto).
- **Sex**: Sexo del paciente (1 = hombre, 0 = mujer).
- **exang**: Angina inducida por ejercicio (1 = sí, 0 = no).
- **caa**: Número de vasos principales (valores posibles: 0-3)
- **chol**: Colesterol (en mg/dl) obtenido a través del sensor de índice de masa corporal (IMC).
- **fbs**: Azúcar en sangre en ayuno > 120 mg/dl (1 = verdadero, 0 = falso).
- **rest_ecg**: Resultados electrocardiográficos en reposo.
 - Valor 0: Normal.
 - Valor 1: Anormalidades en las ondas ST-T (inversiones de onda T y/o elevación o depresión ST > 0.05 mV).
 - Valor 2: Hipertrofia ventricular izquierda probable o definitiva según los criterios de Estes.

Objetivos

Al analizar el dataset “Heart Attack Analysis & Prediction Dataset”, se pueden establecer varios objetivos metodológicos:

1. Análisis exploratorio de los datos: estudiar las variables y los patrones de los datos para obtener una comprensión general de los datos y detectar posibles problemas o outliers.

2. Identificar factores de riesgo: utilizar técnicas estadísticas para identificar variables que estén relacionadas con un mayor riesgo de sufrir un ataque al corazón.
3. Desarrollar modelos de predicción: utilizar algoritmos de aprendizaje automático para desarrollar modelos que puedan predecir el riesgo de sufrir un ataque al corazón en base a los datos de los pacientes.
4. Evaluar el rendimiento de los modelos: medir la precisión y el rendimiento de los modelos desarrollados para evaluar su capacidad para predecir el riesgo de sufrir un ataque al corazón.
5. Identificar las variables más importantes: Utilizar técnicas de selección de características para identificar las variables más importantes en la predicción de riesgo de un ataque cardíaco.
6. Utilizar el modelo para la toma de decisiones: Utilizar el modelo generado para la toma de decisiones clínicas y asesoramiento a pacientes.

Integración y selección

Como se indicó en el apartado anterior existen muchas variables con las cuales se pueden trabajar para poder realizar un análisis por lo que empezaremos cargando el fichero de datos.

```
path = 'heart.csv'
df <- read.csv(path, row.names=NULL)
```

Verificamos la estructura del juego de datos principal. Mostramos el número de columnas que tenemos y ejemplos de los contenidos de las filas.

```
structure = str(df)
```

```
## 'data.frame': 303 obs. of 14 variables:
## $ age : int 63 37 41 56 57 57 56 44 52 57 ...
## $ sex : int 1 1 0 1 0 1 0 1 1 1 ...
## $ cp : int 3 2 1 1 0 0 1 1 2 2 ...
## $ trtbps : int 145 130 130 120 120 140 140 120 172 150 ...
## $ chol : int 233 250 204 236 354 192 294 263 199 168 ...
## $ fbs : int 1 0 0 0 0 0 0 0 1 0 ...
## $ restecg : int 0 1 0 1 1 1 0 1 1 1 ...
## $ thalachh: int 150 187 172 178 163 148 153 173 162 174 ...
## $ exng : int 0 0 0 0 1 0 0 0 0 0 ...
## $ oldpeak : num 2.3 3.5 1.4 0.8 0.6 0.4 1.3 0 0.5 1.6 ...
## $ slp : int 0 0 2 2 2 1 1 2 2 2 ...
## $ caa : int 0 0 0 0 0 0 0 0 0 0 ...
## $ thall : int 1 2 2 2 2 1 2 3 3 2 ...
## $ output : int 1 1 1 1 1 1 1 1 1 1 ...
```

```
summary(df)
```

```
##      age      sex      cp      trtbps
## Min.   :29.00  Min.   :0.0000  Min.   :0.000  Min.   : 94.0
## 1st Qu.:47.50  1st Qu.:0.0000  1st Qu.:0.000  1st Qu.:120.0
## Median :55.00  Median :1.0000  Median :1.000  Median :130.0
## Mean   :54.37  Mean   :0.6832  Mean   :0.967  Mean   :131.6
## 3rd Qu.:61.00  3rd Qu.:1.0000  3rd Qu.:2.000  3rd Qu.:140.0
## Max.   :77.00  Max.   :1.0000  Max.   :3.000  Max.   :200.0
##      chol      fbs      restecg      thalachh
## Min.   :126.0  Min.   :0.0000  Min.   :0.0000  Min.   : 71.0
## 1st Qu.:211.0  1st Qu.:0.0000  1st Qu.:0.0000  1st Qu.:133.5
## Median :240.0  Median :0.0000  Median :1.0000  Median :153.0
## Mean   :246.3  Mean   :0.1485  Mean   :0.5281  Mean   :149.6
## 3rd Qu.:274.5  3rd Qu.:0.0000  3rd Qu.:1.0000  3rd Qu.:166.0
## Max.   :564.0  Max.   :1.0000  Max.   :2.0000  Max.   :202.0
##      exng      oldpeak      slp      caa
## Min.   :0.0000  Min.   :0.00  Min.   :0.000  Min.   :0.0000
## 1st Qu.:0.0000  1st Qu.:0.00  1st Qu.:1.000  1st Qu.:0.0000
## Median :0.0000  Median :0.80  Median :1.000  Median :0.0000
## Mean   :0.3267  Mean   :1.04  Mean   :1.399  Mean   :0.7294
## 3rd Qu.:1.0000  3rd Qu.:1.60  3rd Qu.:2.000  3rd Qu.:1.0000
## Max.   :1.0000  Max.   :6.20  Max.   :2.000  Max.   :4.0000
##      thall      output
## Min.   :0.000  Min.   :0.0000
## 1st Qu.:2.000  1st Qu.:0.0000
## Median :2.000  Median :1.0000
## Mean   :2.314  Mean   :0.5446
## 3rd Qu.:3.000  3rd Qu.:1.0000
## Max.   :3.000  Max.   :1.0000
```

Vemos que tenemos **14** variables y **303** registros.

Limpieza de los datos

Valores nulos o vacíos

Un paso esencial será la limpieza de datos, mirando si hay valores vacíos o nulos.

```
colSums(is.na(df))
```

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

```
colSums(df == "")
```

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

Vemos que no hay valores nulos en los datos ni existen campos llenos de espacios en blanco. Procedemos a convertir a factores las variables correspondientes a este tipo.

```
df$sex <- factor(df$sex, levels = c(0, 1), labels = c("Female", "Male"))
df$exng <- factor(df$exng, levels = c(0, 1), labels = c("No", "Yes"))
df$cp <- factor(df$cp, levels = c(0, 1, 2, 3), labels = c("Typical Angina", "Atypical Angina", "Non-ang"))
df$fbs <- factor(df$fbs, levels = c(0, 1), labels = c("False", "True"))
df$restecg <- factor(df$restecg, levels = c(0, 1, 2), labels = c("Normal", "ST-T wave abnormality", "Le"))
df$thall <- factor(df$thall)
```

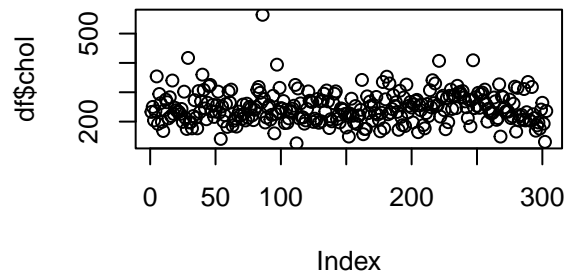
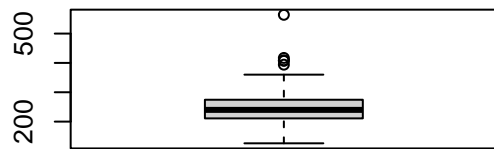
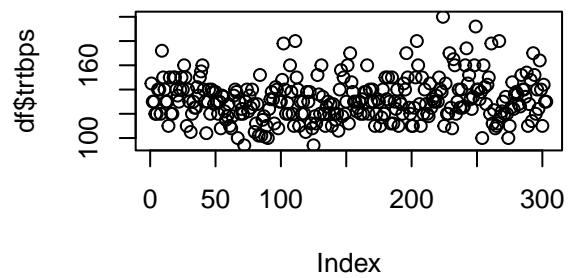
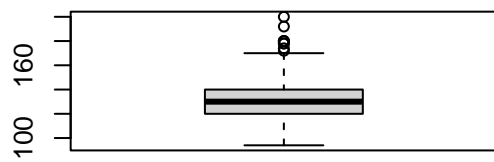
Valores extremos

A continuación, analizaremos los posibles valores extremos de nuestro juego de datos.

```
par(mfrow = c(2,2))

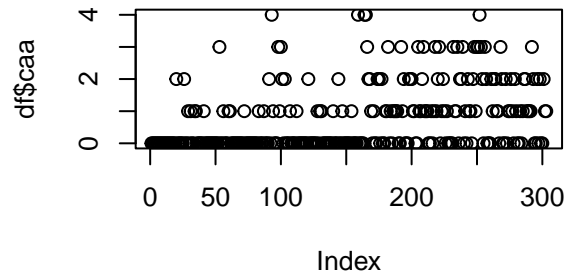
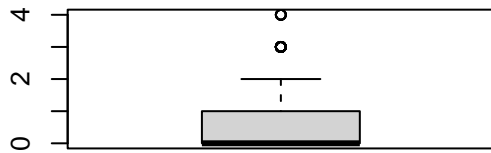
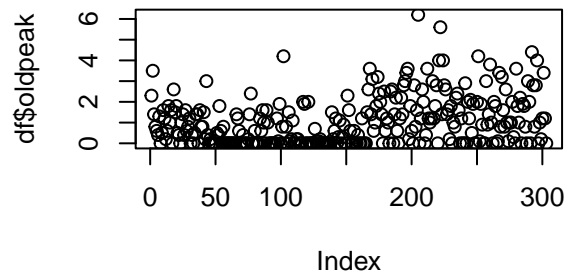
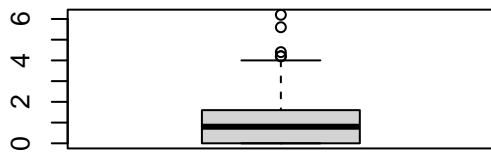
boxplot(df$trtbps)
plot(df$trtbps, )

boxplot(df$chol)
plot(df$chol, )
```



```
boxplot(df$oldpeak)
plot(df$oldpeak, )

boxplot(df$caa)
plot(df$caa, )
```



```
# Cálculo del rango intercuartil (iqr) para cada variable
iqr_trtbps <- quantile(df$trtbps, 0.75) - quantile(df$trtbps, 0.25)
iqr_chol <- quantile(df$chol, 0.75) - quantile(df$chol, 0.25)
iqr_oldpeak <- quantile(df$oldpeak, 0.75) - quantile(df$oldpeak, 0.25)
iqr_caa <- quantile(df$caa, 0.75) - quantile(df$caa, 0.25)

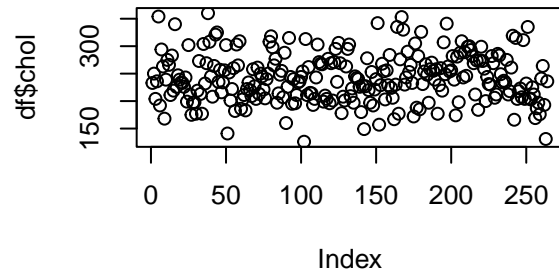
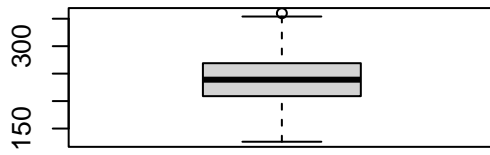
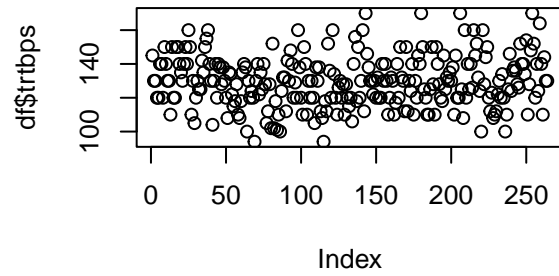
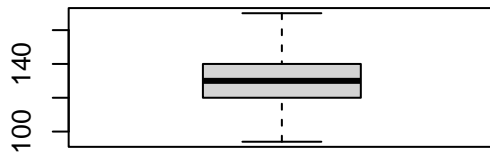
# Cálculo de los límites superior e inferior para cada variable
upper_trtbps <- quantile(df$trtbps, 0.75) + 1.5*iqr_trtbps
lower_trtbps <- quantile(df$trtbps, 0.25) - 1.5*iqr_trtbps
upper_chol <- quantile(df$chol, 0.75) + 1.5*iqr_chol
lower_chol <- quantile(df$chol, 0.25) - 1.5*iqr_chol
upper_oldpeak <- quantile(df$oldpeak, 0.75) + 1.5*iqr_oldpeak
lower_oldpeak <- quantile(df$oldpeak, 0.25) - 1.5*iqr_oldpeak
upper_caa <- quantile(df$caa, 0.75) + 1.5*iqr_caa
lower_caa <- quantile(df$caa, 0.25) - 1.5*iqr_caa

# Eliminar los valores atípicos
df <- df[df$trtbps <= upper_trtbps & df$trtbps >= lower_trtbps,]
df <- df[df$chol <= upper_chol & df$chol >= lower_chol,]
df <- df[df$oldpeak <= upper_oldpeak & df$oldpeak >= lower_oldpeak,]
df <- df[df$caa <= upper_caa & df$caa >= lower_caa,]
```

```
par(mfrow = c(2,2))

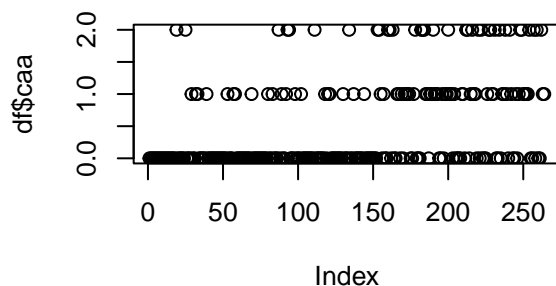
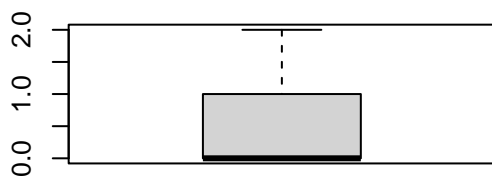
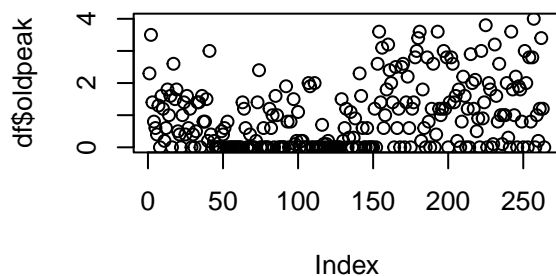
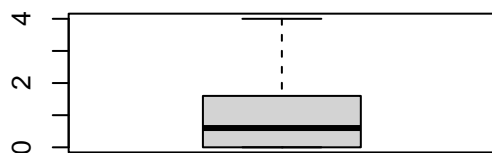
boxplot(df$trtbps)
```

```
plot(df$trtbps, )
boxplot(df$chol)
plot(df$chol, )
```



```
boxplot(df$oldpeak)
plot(df$oldpeak, )

boxplot(df$caa)
plot(df$caa, )
```

Análisis de los datos

Selección de los grupos de datos que se quieren analizar/comparar

En nuestro análisis compararemos los pacientes que sufrieron un ataque cardíaco con los que no. Lo haremos mediante la selección de dos grupos de datos basados en el valor de la variable **output**: pacientes con valor de **output** igual a 1 (pacientes que sufrieron un ataque cardíaco) y pacientes con valor de **output** igual a 0 (pacientes que no sufrieron un ataque cardíaco). Con estos dos grupos seleccionados, podremos aplicar diferentes métodos de análisis estadístico para comparar las diferencias entre los grupos en las diferentes variables del dataset.

- **Correlaciones:** mostraremos las correlaciones y distribuciones de algunas variables para ver como se relacionan entre sí.
- **Prueba t-Student:** utilizaremos una prueba t de student para comparar el promedio de la edad entre los pacientes que sufrieron un ataque cardíaco y los que no.
- **Regresión logística:** generaremos una regresión logística para analizar la relación entre las variables y el riesgo de sufrir un ataque cardíaco.

Como primer punto a analizar vamos a verificar si el nivel de colesterol medido en miligramos por decilitro mg/dl tiene una relacion más directa que la edad con la probabilidad de sufrir un ataque cardiaco que la Para ellos creamos un nuevo dataset con estas dos variables.

Comprobación de la normalidad y homogeneidad de la varianza

Como primer punto a analizar vamos a verificar si existe una diferencia estadística en el colesterol de hombres y mujeres que tienen una mayor probabilidad de sufrir un ataque al corazón Para ellos creamos un nuevo dataset con estas dos variables filtrando solo los casos que si tenían posibilidad de un ataque.

```
test <- df[,names(df) %in% c("chol","sex", "output")]
test <- test %>% filter(output == 1)
structure= str(test)
```

```
## 'data.frame': 152 obs. of 3 variables:
## $ sex : Factor w/ 2 levels "Female","Male": 2 2 1 2 1 2 1 2 2 2 ...
## $ chol : int 233 250 204 236 354 192 294 263 168 239 ...
## $ output: int 1 1 1 1 1 1 1 1 1 1 ...
```

Vamos a comprobar la normalidad la variable chol (colesterol).

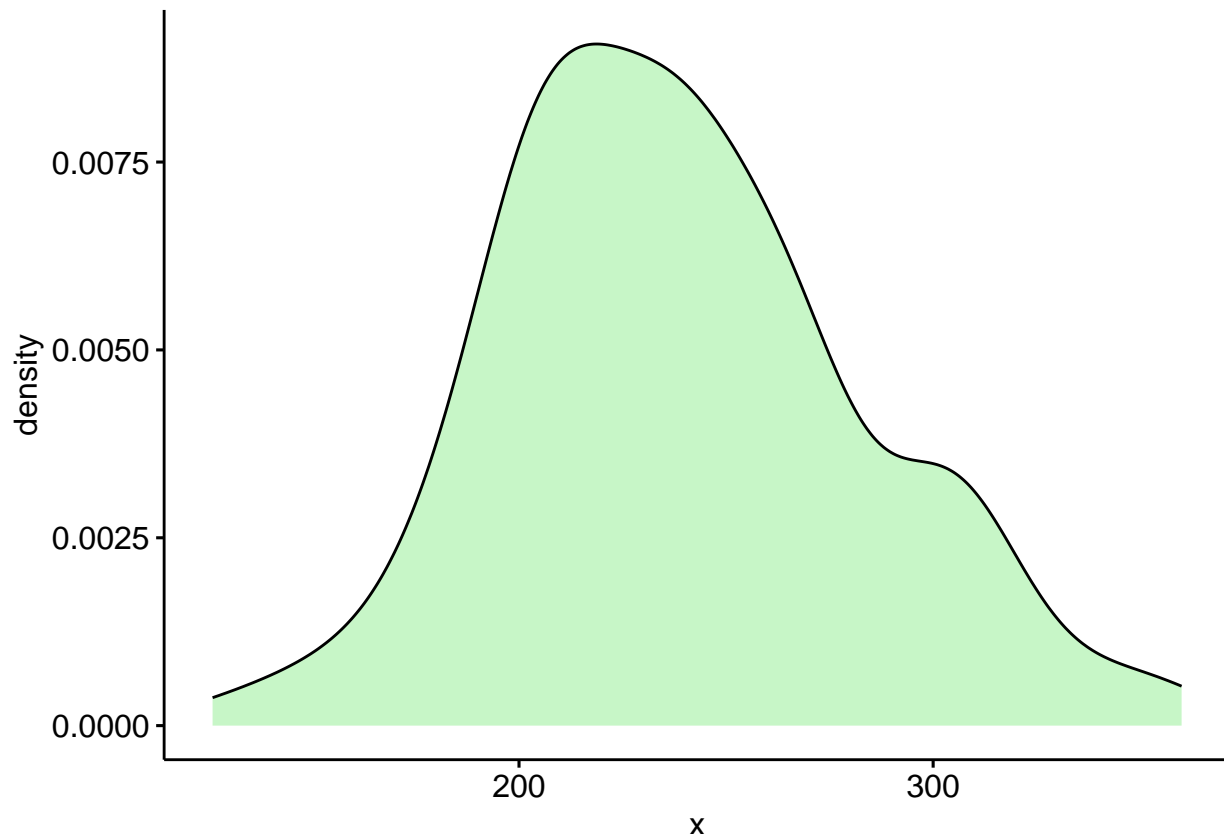
```
ks.test(test$chol, pnorm, mean(test$chol), sd(test$chol))
```

```
##
## Asymptotic one-sample Kolmogorov-Smirnov test
##
## data: test$chol
## D = 0.062247, p-value = 0.5979
## alternative hypothesis: two-sided
```

```
shapiro.test(test$chol)
```

```
##
## Shapiro-Wilk normality test
##
## data: test$chol
## W = 0.98514, p-value = 0.1023
```

```
grid.arrange((ggdensity(test$chol, fill = "lightgreen")),ncol=1)
```



Si nuestra hipótesis nula es que la población está distribuida normalmente, si el p-valor es menor a una significancia de 0.05, entonces rechazamos la hipótesis y concluimos que los datos no cuentan con una distribución normal. Para nuestra variable colesterol según Kolmogorov-Smirnov siguen una distribución normal mientras que para el test de Shapiro-Wilk se rechaza la hipótesis nula y considera que no es así. Sin embargo por el teorema del límite central, podemos considerar que los datos sí siguen una distribución normal. Podemos apoyar dichos resultados igualmente a través de un gráfico de densidad, podemos ver la distribución de valores para la variable peso y demostrar que toma la forma de campana esperada que indica normalidad.

A continuación para comprobar la homocedasticidad o homogeneidad de la varianza vamos a utilizar el test de Levene y el de Fligner-Killeen que se usa cuando los datos siguen una distribución normal. Lo haremos comprobando nuestras dos variables colesterol y edad con la siguiente hipótesis. H_0 : colesterol = genero
 H_A : colesterol \neq genero

```
leveneTest(chol ~ sex, data = test)
```

```
## Levene's Test for Homogeneity of Variance (center = median)
##      Df F value Pr(>F)
## group 1  6.0507 0.01504 *
##      150
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
fligner.test(chol ~ sex, data = test)
```

```
##
```

```
## Fligner-Killeen test of homogeneity of variances
##
## data: chol by sex
## Fligner-Killeen:med chi-squared = 6.2834, df = 1, p-value = 0.01219
```

Dado que el p-value es mayor que la significancia 0.05, no podemos rechazar la hipótesis nula, H_0 . La prueba de Levene no es estadísticamente significativa y puede asumir que la varianza de la población de hombres y mujeres es homogénea y puede asumir con seguridad la misma varianza.

Aplicación de pruebas estadísticas para comparar los grupos de datos.

Prueba t-Student

Ya que la normalidad y la homocedasticidad se cumplieron, ahora podemos aplicar pruebas por contraste de hipótesis de tipo paramétrico como la prueba t Student

```
t.test(chol ~ sex, data = test)
```

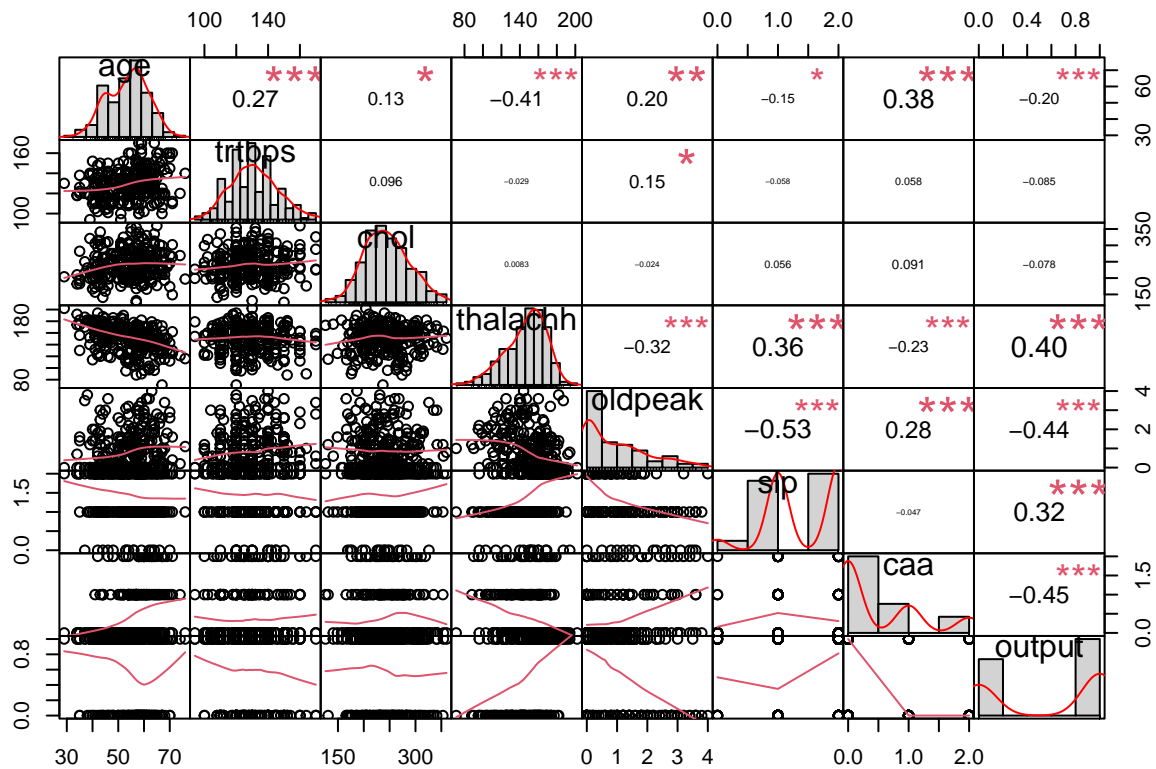
```
##
## Welch Two Sample t-test
##
## data: chol by sex
## t = 2.0051, df = 125.27, p-value = 0.04711
## alternative hypothesis: true difference in means between group Female and group Male is not equal to
## 95 percent confidence interval:
## 0.1889702 28.9931026
## sample estimates:
## mean in group Female mean in group Male
## 246.8529 232.2619
```

El p-valor 0.003927 obtenido de esta prueba t de student es mayor al nivel de significancia, por ende no se observan diferencias estadísticamente significativas entre el sexo del paciente para la variable colesterol.

Correlaciones

La librería PerformanceAnalytics en R incluye una función llamada chart.Correlation, la cual permite generar un gráfico de relación entre variables con elementos como histogramas, funciones de densidad, líneas de regresión suavizadas y coeficientes de correlación con indicadores de significancia estadística (representado por estrellas, donde ausencia de estrellas indica que la variable no es estadísticamente significativa, mientras que una, dos y tres estrellas indican significancia estadística al nivel del 10%, 5% y 1%, respectivamente).

```
if(!require("PerformanceAnalytics")) install.packages("PerformanceAnalytics"); library(PerformanceAnalytics)
df_num <- df[,sapply(df,is.numeric)]
df_num <- df_num[complete.cases(df_num),]
chart.Correlation(df_num, histogram = TRUE, method = "pearson")
```



Regresión logística

Queremos analizar la relación entre las variables y el riesgo de sufrir un ataque cardíaco. Para ello utilizaremos una regresión logística, tomando como variable dependiente **output** i un conjunto de variables explicativas de la base de datos, que iremos eliminando o añadiendo según creamos conveniente. Para poder estimar de forma más objetiva la precisión del modelo, separaremos el conjunto de datos en dos partes: el conjunto de entrenamiento (training) y el conjunto de prueba (testing). Ajustaremos el modelo de regresión logística con el conjunto de entrenamiento, y evaluaremos la precisión con el conjunto de prueba.

Generación de los conjuntos de entrenamiento y de test Generaremos los conjuntos de datos para entrenar el modelo (training) y para testarlo (testing). En este caso, fijaremos el tamaño de la muestra de entrenamiento a un 80% del original.

```
if(!require("caret")) install.packages("caret"); library(caret)

set.seed(888)

# Crear partición de los datos
partition <- createDataPartition(df$output, p = 0.8, list = FALSE)

# Generar conjunto de entrenamiento
training_data <- df[sample(partition), ]

# Generar conjunto de prueba
testing_data <- df[-sample(partition), ]
```

Estimación del modelo con el conjunto de entrenamiento e interpretación Procedemos a seleccionar las variables explicativas y a entrenar el modelo de regresión. Empezamos seleccionando todos los atributos del conjunto de datos.

```
# Seleccionar variables explicativas
explanatory_vars <- names(training_data)[!names(training_data) %in% c("")]

# Estimar modelo de regresión logística
log_model <- glm(output ~ ., data = training_data[, explanatory_vars], family = binomial)

summary(log_model)
```

```
##
## Call:
## glm(formula = output ~ ., family = binomial, data = training_data[,
##      explanatory_vars])
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -2.8812  -0.1901   0.1482   0.5308   2.8725
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    -2.526e+00  4.112e+00  -0.614  0.539018
## age              2.848e-02  3.068e-02   0.928  0.353262
## sexMale         -2.089e+00  6.774e-01  -3.084  0.002044 **
## cpAtypical Angina  9.735e-01  6.473e-01   1.504  0.132605
## cpNon-anginal pain  2.289e+00  6.712e-01   3.409  0.000651 ***
## cpAsymptomatic    2.448e+00  8.239e-01   2.971  0.002968 **
## trtbps          -2.095e-02  1.574e-02  -1.331  0.183184
## chol            -4.605e-03  6.301e-03  -0.731  0.464889
## fbsTrue          4.511e-01  8.227e-01   0.548  0.583465
## restecgST-T wave abnormality  6.658e-01  4.683e-01   1.422  0.155117
## restecgLeft ventricular hypertrophy  1.205e+01  1.455e+03   0.008  0.993394
## thalachh         2.614e-02  1.526e-02   1.713  0.086719 .
## exngYes         -5.089e-01  5.815e-01  -0.875  0.381491
## oldpeak         -6.838e-01  3.066e-01  -2.231  0.025714 *
## slp              7.165e-01  4.384e-01   1.634  0.102202
## caa            -1.740e+00  4.128e-01  -4.216  2.48e-05 ***
## thall1          4.066e+00  2.530e+00   1.607  0.108023
## thall2          2.432e+00  2.320e+00   1.048  0.294704
## thall3          8.337e-01  2.324e+00   0.359  0.719756
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 285.52  on 211  degrees of freedom
## Residual deviance: 130.80  on 193  degrees of freedom
## AIC: 168.8
##
```

```
## Number of Fisher Scoring iterations: 14
```

Los puntos clave de los resultados de una regresión logística son los coeficientes de regresión, los valores de p y las medidas de ajuste del modelo.

1. Los coeficientes de regresión: Estos indican la dirección y magnitud de la relación entre cada variable independiente y la variable dependiente. Un coeficiente positivo indica una relación positiva entre la variable independiente y la variable dependiente, mientras que un coeficiente negativo indica una relación negativa. En este caso, un ejemplo de esto es que `cpNon-anginal pain` tiene un coeficiente positivo de 2.289 y significativamente alto, esto indica que tiene una relación positiva con la probabilidad de sufrir un ataque cardíaco.
2. Los valores de p: Estos indican la probabilidad de que el coeficiente de regresión sea cero (es decir, que no haya relación entre la variable independiente y la variable dependiente). Un valor de p menor que el nivel de significación establecido (generalmente 0.05) indica que es poco probable que el coeficiente sea cero y, por lo tanto, que hay una relación estadísticamente significativa entre la variable independiente y la variable dependiente. En este caso, por ejemplo, el valor de p de 'cpNon-anginal pain' es muy bajo, es decir, 0.000651, lo que indica que es muy probable que exista una relación entre 'cpNon-anginal pain' y la probabilidad de sufrir un ataque cardíaco.
3. Medidas de ajuste del modelo: Estas medidas indican cómo bien el modelo se ajusta a los datos. La desviación residual y el AIC son dos medidas que nos indican qué tan bien se ajusta un modelo a los datos. Una desviación residual y AIC bajos indican que el modelo tiene un mejor ajuste.

Usaremos el método VIF (Variance Inflation Factor) para estudiar la presencia o no de colinealidad en nuestro modelo de regresión. Un VIF alto indica una alta correlación entre una variable independiente y las demás variables independientes en el modelo. Un valor de VIF superior a 5 suele ser considerado como indicativo de un problema de multicolinealidad.

```
fiv <- car::vif(log_model)
fiv
```

```
##           GVIF Df GVIF^(1/(2*Df))
## age      1.618354 1      1.272145
## sex      1.539250 1      1.240665
## cp       2.207040 3      1.141042
## trtbps   1.321937 1      1.149755
## chol     1.166049 1      1.079837
## fbs      1.239088 1      1.113143
## restecg  1.129608 2      1.030937
## thalachh 1.666627 1      1.290979
## exng     1.185293 1      1.088712
## oldpeak  1.703087 1      1.305024
## slp      1.606707 1      1.267560
## caa      1.528599 1      1.236365
## thall    1.945063 3      1.117264
```

Los resultados indican que ninguna de las variables independientes del modelo tiene un VIF alto. Esto sugiere que no hay un problema significativo de multicolinealidad entre las variables independientes del modelo.

Como vimos en el primer modelo, algunas variables no eran estadísticamente significativas (es decir, cuyo valor p sea menor a 0.05), pero como las consideramos importantes en el contexto del análisis, en vez de eliminarlas de dicho modelo entrenaremos uno adicional sin ellas para luego compararlas entre sí.

```
# Seleccionar variables explicativas
explanatory_vars <- names(training_data)[!names(training_data) %in% c("age", "trtbps", "fbs", "restecg")]

# Estimar modelo de regresión logística
log_model_2 <- glm(output ~ ., data = training_data[, explanatory_vars], family = binomial)

summary(log_model_2)
```

```
##
## Call:
## glm(formula = output ~ ., family = binomial, data = training_data[,
##     explanatory_vars])
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -2.5818  -0.2502   0.2024   0.5117   2.6803
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    -2.56728     2.92087  -0.879  0.379433
## sexMale        -1.84280     0.61411  -3.001  0.002693 **
## cpAtypical Angina  0.96459     0.61962   1.557  0.119530
## cpNon-anginal pain 2.07671     0.60310   3.443  0.000574 ***
## cpAsymptomatic    2.12331     0.75432   2.815  0.004880 **
## thalachh         0.02304     0.01283   1.796  0.072423 .
## exngYes         -0.49020     0.54890  -0.893  0.371831
## oldpeak        -0.85320     0.26967  -3.164  0.001557 **
## caa            -1.42688     0.34793  -4.101  4.11e-05 ***
## thall1         3.39969     2.54490   1.336  0.181588
## thall2         1.98943     2.37605   0.837  0.402432
## thall3         0.51445     2.37218   0.217  0.828312
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 285.52  on 211  degrees of freedom
## Residual deviance: 139.18  on 200  degrees of freedom
## AIC: 163.18
##
## Number of Fisher Scoring iterations: 6
```

Como observamos, el primer modelo tiene una desviación residual menor (se ajusta mejor) aunque lo hace a costa de un mayor AIC (utiliza más variables, es menos parsimonioso).

Cálculo de las OR (Odds-Ration) Las odds ratios (OR) son una medida del efecto relativo de una variable explicativa sobre la variable dependiente en un modelo de regresión logística. Una OR mayor que 1 indica que la variable explicativa tiene un efecto positivo en la variable dependiente, mientras que una OR menor que 1 indica un efecto negativo.

```
# Obtener coeficientes del modelo
coefs <- coef(log_model)
```



```
# Calcular factores de riesgo o protección
risk_factors <- exp(coefs)

# Imprimir factores de riesgo o protección
risk_factors
```

```
##              (Intercept)              age
##          7.999867e-02          1.028889e+00
##              sexMale          cpAtypical Angina
##          1.237987e-01          2.647324e+00
##          cpNon-anginal pain          cpAsymptomatic
##          9.860146e+00          1.156429e+01
##              trtbps              chol
##          9.792699e-01          9.954057e-01
##              fbsTrue          restecgST-T wave abnormality
##          1.570060e+00          1.945987e+00
##          restecgLeft ventricular hypertrophy          thalachh
##          1.711562e+05          1.026488e+00
##              exngYes              oldpeak
##          6.011512e-01          5.046832e-01
##              slp              caa
##          2.047305e+00          1.754768e-01
##              thall1              thall2
##          5.834936e+01          1.137594e+01
##              thall3
##          2.301811e+00
```

En el caso observado, se puede ver que la OR para la variable `cpNon-anginal pain` es de 10.94, lo que significa que tener angina atípica tiene 10.94 veces más probabilidades de tener un ataque cardíaco en comparación con tener una angina típica.

```
# Obtener predicciones del modelo en el conjunto de prueba
predictions <- predict(log_model, newdata = testing_data, type = "response")

# Crear nueva variable con predicciones del modelo
testing_data$prediction <- ifelse(predictions >= 0.5, 1, 0)

# Comparar predicciones del modelo con variable dependiente del conjunto de prueba
confusion_matrix <- table(testing_data$prediction, testing_data$output)

confusion_matrix
```

Matriz de confusión

```
##
##      0  1
## 0 22  2
## 1  5 23
```

```
# Calcular tasa de aciertos del modelo
accuracy <- mean(testing_data$prediction == testing_data$output)

# Imprimir tasa de aciertos del modelo
print(accuracy)
```

```
## [1] 0.8653846
```

```
sensitivity(confusion_matrix)
```

```
## [1] 0.8148148
```

```
specificity(confusion_matrix)
```

```
## [1] 0.92
```

Los resultados indican que el modelo de regresión logística tiene una tasa de aciertos del 87%. Esto quiere decir que, de las predicciones del modelo, el 87% son correctas.

Para entender mejor estos resultados, podemos observar la tabla de contingencia que se ha obtenido al comparar la variable prediction con la variable dependiente (output) del conjunto de prueba. Esta tabla nos muestra el número de veces que se ha dado cada combinación de valores en ambas variables.

```
# Seleccionar variables explicativas
explanatory_vars <- names(df)[!names(df) %in% c("output")]

# Estimar modelo de regresión logística
patients <- df[c(12, 200), explanatory_vars]

patients
```

Predicción

```
##      age sex      cp trtbps chol  fbs      restecg thalachh
## 13   49 Male Atypical Angina   130 266 False ST-T wave abnormality   171
## 220  48 Male Typical Angina   130 256 True      Normal   150
##      exng oldpeak slp caa thall
## 13   No      0.6   2   0      2
## 220  Yes      0.0   2   2      3
```

```
# Hacer la predicción
prediction_patients <- predict(log_model, patients, interval = "confidence", type = "response")

# Mostrar el resultado de la predicción
prediction_patients
```

```
##           13           220
## 0.91656261 0.01098676
```

En el caso del paciente 12, la probabilidad de tener un ataque cardíaco es de 0.917 o 91.7%. Esto significa que según el modelo, hay un 92.5% de posibilidades de que ese paciente sufra un ataque cardíaco. En el caso del paciente 200, la probabilidad es de 0.011 o 1%. Esto significa que según el modelo, hay un 1% de posibilidades de que ese paciente sufra un ataque cardíaco.

Estas probabilidades no deben interpretarse como una predicción absoluta, sino como una herramienta para identificar pacientes con un mayor riesgo de sufrir un ataque cardíaco y tomar medidas preventivas.

Bondad de ajuste Evaluaremos la bondad del ajuste de nuestro modelo mediante la devianza. Para poder considerar que el modelo es bueno, la devianza residual debe ser menor que la devianza nula.

```
log_model$deviance
```

```
## [1] 130.8024
```

```
log_model$null.deviance
```

```
## [1] 285.5183
```

```
if (log_model$deviance < log_model$null.deviance) {  
  print("El modelo log_model tiene un buen ajuste")  
} else {  
  print("El modelo log_model no tiene un buen ajuste")  
}
```

```
## [1] "El modelo log_model tiene un buen ajuste"
```

También podemos evaluar la eficacia dl modelo utilizando el test de chi-cuadrado. Si la probabilidad asociada al estadístico del contraste es mayor o igual a 0.05, como en nuestro caso (**prob** = 1), entonces podemos decir que el modelo log_model es eficaz.

```
chisq_observed <- log_model$null.deviance - log_model$deviance  
prob <- pchisq(chisq_observed, df = log_model$df.residual, lower.tail = FALSE)  
  
prob
```

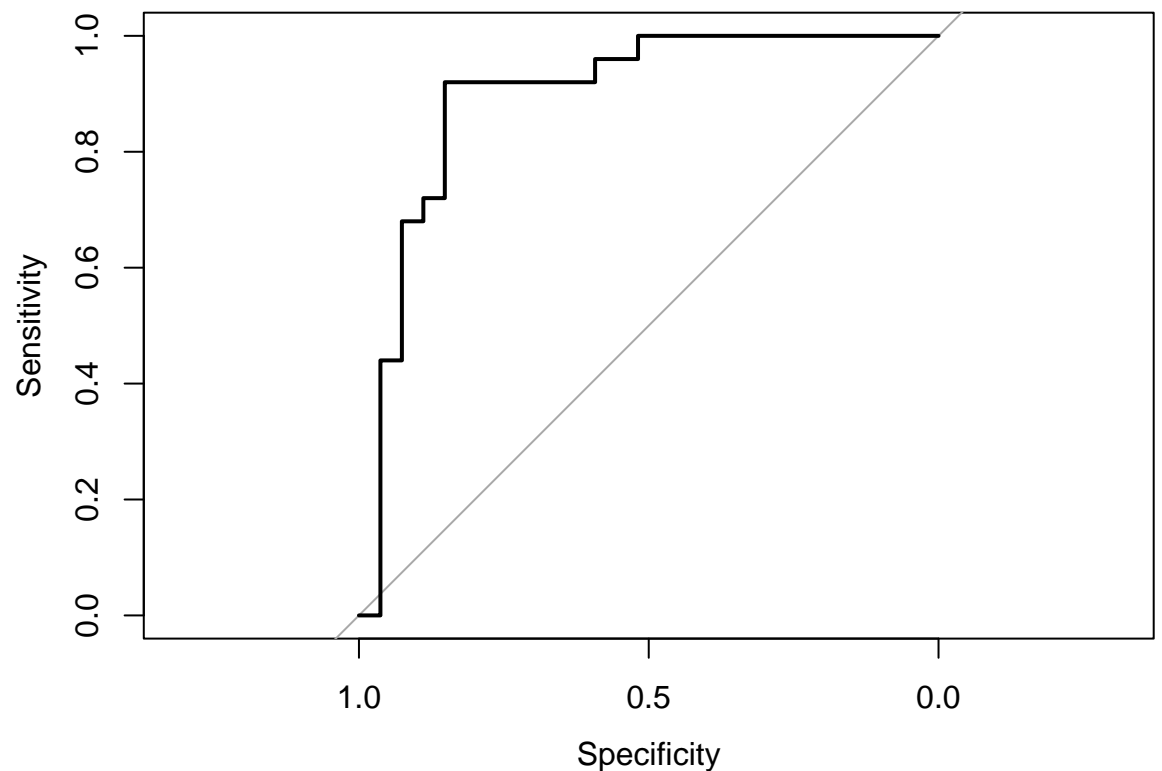
```
## [1] 0.9803124
```

```
if (prob >= 0.05) {  
  print("El modelo log_model es eficaz")  
} else {  
  print("El modelo log_model no es eficaz")  
}
```

```
## [1] "El modelo log_model es eficaz"
```

```
if(!require("pROC")) install.packages("pROC"); library(pROC)

roc_curve <- roc(testing_data$output, predictions)
plot(roc_curve)
```



Curva ROC

```
area_under_curve <- auc(roc_curve)
print(paste("Área debajo de la curva:", area_under_curve))
```

```
## [1] "Área debajo de la curva: 0.896296296296296"
```

El área debajo de la curva (AUC) es una medida de la eficacia del modelo de regresión logística `log_model` para predecir la variable dependiente `output`. Un AUC de 1 indica un modelo perfecto, mientras que un AUC de 0.5 indica un modelo no mejor que el azar.

En este caso, el área debajo de la curva nos indica que el modelo **log_model** es muy eficaz para predecir la variable dependiente `output`. Este resultado es consistente con los resultados de las medidas de sensibilidad y especificidad que obtuvimos anteriormente.

5. Representación de los resultados a partir de tablas y gráficas.

Los resultados graficos se representan en los apartados anteriores.

6. Resolución del problema. A partir de los resultados obtenidos,

Conclusiones

En resumen, el análisis realizado en este dataset se enfocó en el diagnóstico de un ataque cardíaco, utilizando una variedad de variables, incluyendo edad, sexo, tipo de dolor en el pecho, presión arterial, colesterol, glicemia, resultados de electrocardiografía, ritmo cardíaco máximo, entre otros.

Para empezar, se realizó un análisis exploratorio de los datos, que permitió entender mejor la distribución de las variables y detectar posibles valores atípicos. Seguidamente, se aplicaron diferentes técnicas estadísticas para analizar los datos, como análisis de correlación, regresión logística, curva ROC y VIF.

En cuanto a los resultados obtenidos, se pudo observar que algunas variables como el sexo, el tipo de dolor en el pecho, la presión arterial, el ritmo cardíaco máximo y el número de vasos principales tienen una relación significativa con el diagnóstico de un ataque cardíaco. También se observó que algunas variables no tienen una relación significativa con el diagnóstico, como la glicemia y los resultados de electrocardiografía.

Analizando a través de una prueba t student también se pudo determinar que no existían diferencias estadísticamente significativas entre el sexo del paciente con la variable colesterol cuando una persona tiene una probabilidad alta de un ataque.

Por otro lado, el modelo de regresión logística proporcionó resultados precisos para la predicción del diagnóstico, como se pudo observar en la curva ROC. Sin embargo, se recomendaría eliminar algunas variables poco significativas antes de aplicar el modelo para mejorar la precisión.

En conclusión, el análisis realizado en este dataset proporciona una comprensión valiosa sobre los factores que contribuyen al diagnóstico de un ataque cardíaco, y puede ser utilizado para desarrollar un sistema de diagnóstico automatizado o mejorar el proceso de diagnóstico actual. Sin embargo, sería recomendable realizar más investigaciones con conjuntos de datos más grandes y variados para validar estos resultados.

7 Código.

Ver el código en el repositorio: <https://github.com/XnetLoL/TD.PRA2>

8 Video.

Enlace al video en google drive: https://drive.google.com/file/d/1NNcMX8hSdNDo_SEJ0_kgG0_50i5jqRWp/view?usp=sharing

Bibliografía

- Calvo M, Subirats L, Pérez D (2019). *Introducción a la limpieza y análisis de los datos*. Editorial UOC.
- Julià Minguiellón Alfonso, Ramon Caihuelas Quiles. (2021). *Proceso de minería de datos*. Editorial FUOC.
- RASHIK RAHMAN (2021). *Heart Attack Analysis & Prediction Dataset*. <https://www.kaggle.com/datasets/rashikrahmanpritom/heart-attack-analysis-prediction-dataset?resource=download&select=heart.csv>









Contribuciones	Firmas	
Investigación previa	Integrante 1: 	Integrante 2: 
Redacción de las respuestas	Integrante 1: 	Integrante 2: 
Desarrollo del código	Integrante 1: 	Integrante 2: 
Participación en el vídeo	Integrante 1: 	Integrante 2: 

Figure 1: Contribuciones