PRÁCTICA 2

INSUFICIENCIA CARDIACA

Alumna: Melanie Valverde Varas

DESCRIPCIÓN DE LA PRÁCTICA

OBJETIVOS DE LA PRÁCTICA

Los objetivos que se persiguen mediante el desarrollo de esta actividad son los siguientes:

- Aplicar los conocimientos adquiridos en la parte teórica de la asignatura y mejorar la capacidad de resolución de problemas orientados a casos reales.
- Saber identificar los datos más importantes y procesarlos de manera correcta.
- Desarrollar la capacidad de búsqueda y de gestión.

INTRODUCCIÓN AL DATASET

Analizaremos un conjunto de datos de 299 pacientes con insuficiencia cardiaca, recopilado en el 2015. Se han extraído los datos de 105 mujeres y 194 hombres de entre 40 y 95 años. El dataset contiene además información sobre 13 características físicas de cada una de las personas.

Feature	Explanation	Measurement	Range
Age	Edad del paciente	Años	[40,..., 95]
Anaemia	Disminución de los globulos rojos	Booleano	0, 1
High blood pressure	Si el paciente tiene hipertensión	Booleano	0, 1
Creatinine phosphokinase (CPK)	Nivel de la enzima CPK en la sangre	mcg/L	[23,..., 7861]
Diabetes	Si el paciente tiene diabetes	Booleano	0, 1
Ejection fraction	% de sangre que sale del corazón cada vez que se contrae	Porcentaje	[14,..., 80]

Sex	Hombre o mujer	Binario	0(mujer), 1(hombre)
Platelets	Plaquetas en sangre	kiloplatelets/mL	[25.01,..., 850.00]
Serum creatinine	Nivel de creatinina en sangre	mg/dL	[0.50,..., 9.40]
Serum sodium	Nivel de sodio en la sangre	mEq/L	[114,..., 148]
Smoking	Si el paciente es fumador	Booleano	0, 1
Time	Periodo de seguimiento	Días	[4,...,285]
(target) death event	Si el paciente fallece durante el tiempo de seguimiento.	Booleano	0, 1

Estadísticas generales:
Sobrevivientes: 203 pacientes (value: 0)
Fallecidos: 96 (value: 1)

OBJETIVOS DE LA INVESTIGACIÓN

El objetivo de este proyecto es saber los principales atributos del dataset, los que más información nos aportan con la finalidad de conseguir un modelo adecuado que nos permita predecir aquellos pacientes que se encuentren en peligro.

LIMPIEZA DE DATOS

En primer lugar, obtenemos la información del dataset, en este caso desde la web https://archive.ics.uci.edu y procedemos a hacer la lectura del documento:

```
## Cargamos el juego de datos

heart_failure <- read.csv('https://archive.ics.uci.edu/ml/machine-learning-databases/00519/heart_failure_clinical_records_dataset.csv',stringsAsFactors = FALSE, header = T)

sapply(heart_failure, function(x) class(x))
```

Tipos de datos:

```
##                       age                   anaemia creatinine_phosphokinase
##                 "numeric"                 "integer"                 "integer"
##                  diabetes         ejection_fraction       high_blood_pressure
##                 "integer"                 "integer"                 "integer"
##                 platelets          serum_creatinine              serum_sodium
##                 "numeric"                 "numeric"                 "integer"
##                       sex                   smoking                      time
##                 "integer"                 "integer"                 "integer"
##                     death
##                 "integer"
```

SELECCIÓN DE INFORMACIÓN DE INTERÉS

En este caso dejaremos todos los atributos, ya que forman parte de las características físicas que pueden influir en la gravedad de esta enfermedad.

ELEMENTOS INCONSISTENTES

```
sum(is.na(heart_failure))

colSums(is.na(heart_failure)
```

```
##                      age                  anaemia creatinine_phosphokinase
##                        0                        0                        0
##                 diabetes         ejection_fraction      high_blood_pressure
##                        0                        0                        0
##                platelets          serum_creatinine             serum_sodium
##                        0                        0                        0
##                      sex                  smoking                     time
##                        0                        0                        0
##                    death
##                        0
```

```
colSums(heart_failure=="")
```

```
##                      age                  anaemia creatinine_phosphokinase
##                        0                        0                        0
##                 diabetes         ejection_fraction      high_blood_pressure
##                        0                        0                        0
##                platelets          serum_creatinine             serum_sodium
##                        0                        0                        0
##                      sex                  smoking                     time
##                        0                        0                        0
##                    death
##                        0
```

```
colSums(heart_failure==" ?")
```

```
##                      age                  anaemia creatinine_phosphokinase
##                        0                        0                        0
##                 diabetes         ejection_fraction      high_blood_pressure
##                        0                        0                        0
##                platelets          serum_creatinine             serum_sodium
##                        0                        0                        0
##                      sex                  smoking                     time
##                        0                        0                        0
##                    death
##                        0
```

Como podemos observar nuestro dataset, no contiene datos nulos, ni vacíos. Por lo tanto, podemos continuar con la investigación. Cabe resaltar que, en el caso de haber encontrado datos inconsistentes, la solución más práctica sería reemplazarlos por la media de los valores o simplemente eliminarlos de nuestro dataset.

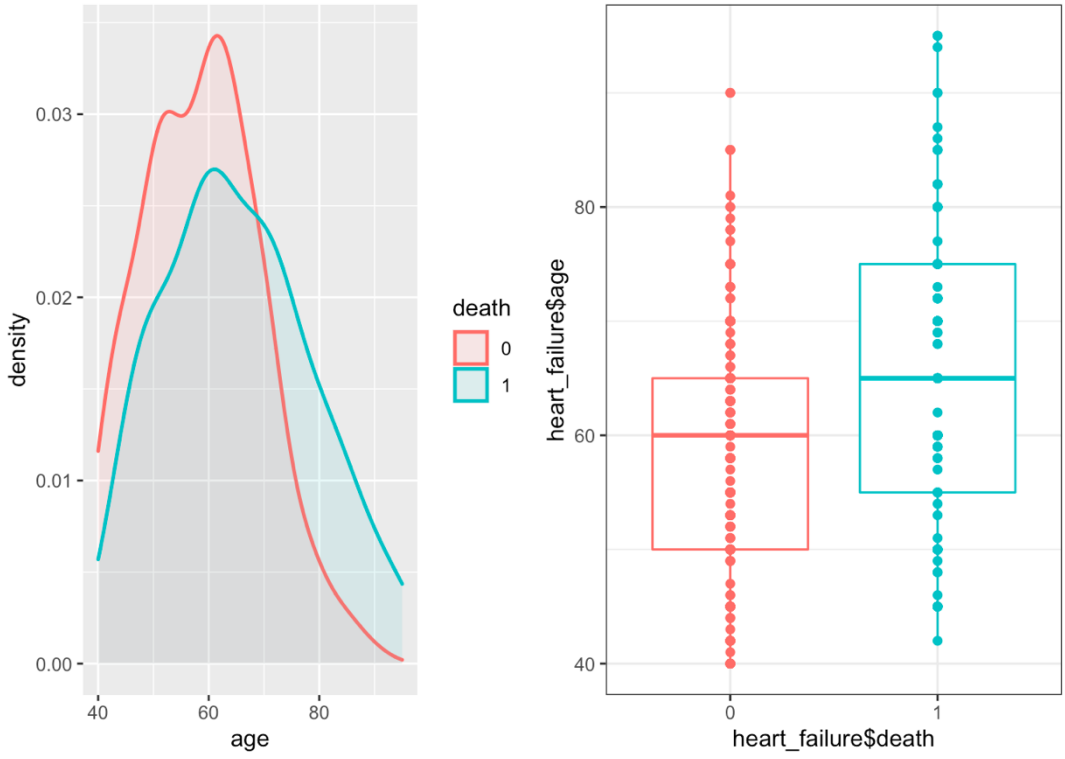
VALORES EXTREMOS

Recogemos los valores extremos para cada uno de los atributos que estamos estudiando mediante las siguientes funciones y pasaremos a crear el gráfico de la caja y los bigotes para cada uno con el objetivo de visualizar mejor los outliers y extraer más información.

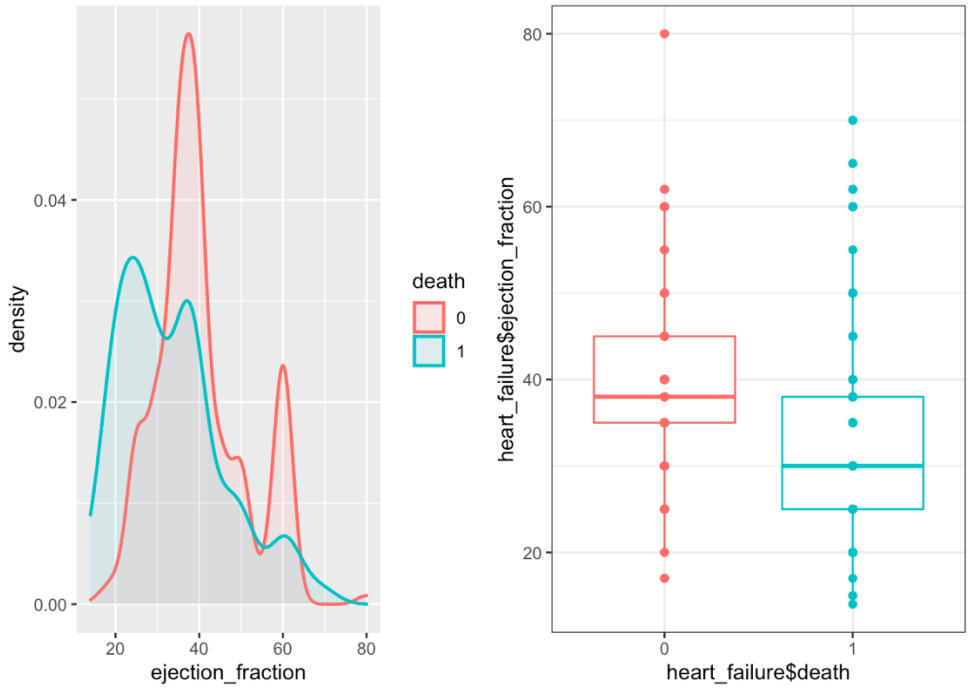
```
boxplot.stats(heart_failure$age)$out

a1 <- ggplot(heart_failure, aes(age, fill=death, colour = death))+geom_density(alpha=0.1, size=0.8)

a2 <- ggplot(data = heart_failure, aes(x = heart_failure$death, y = heart_failure$age, colour = heart_failure$death)) +
  geom_boxplot() +
  geom_point() +
  theme_bw() +
  theme(legend.position = "none")

grid.arrange(a1, a2, ncol = 2)
```

#outliers

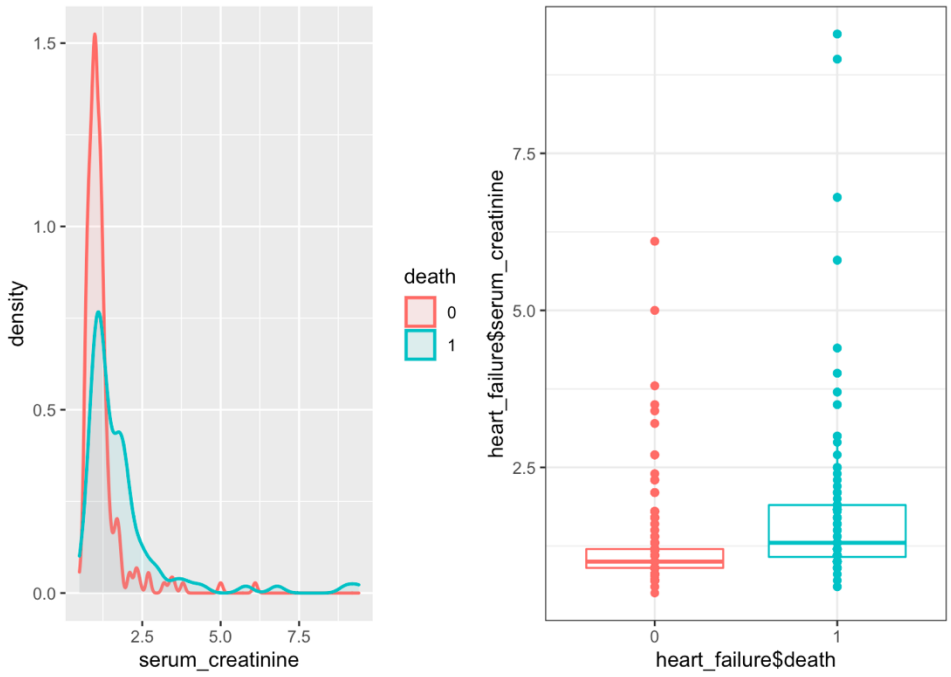
```
## integer(0)
```



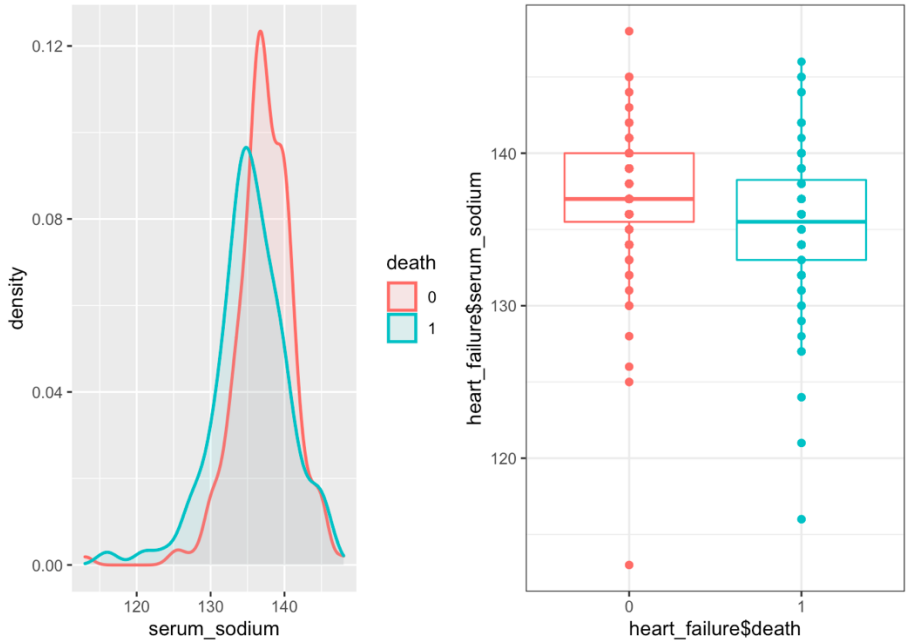
```
## [1] 80 70
```



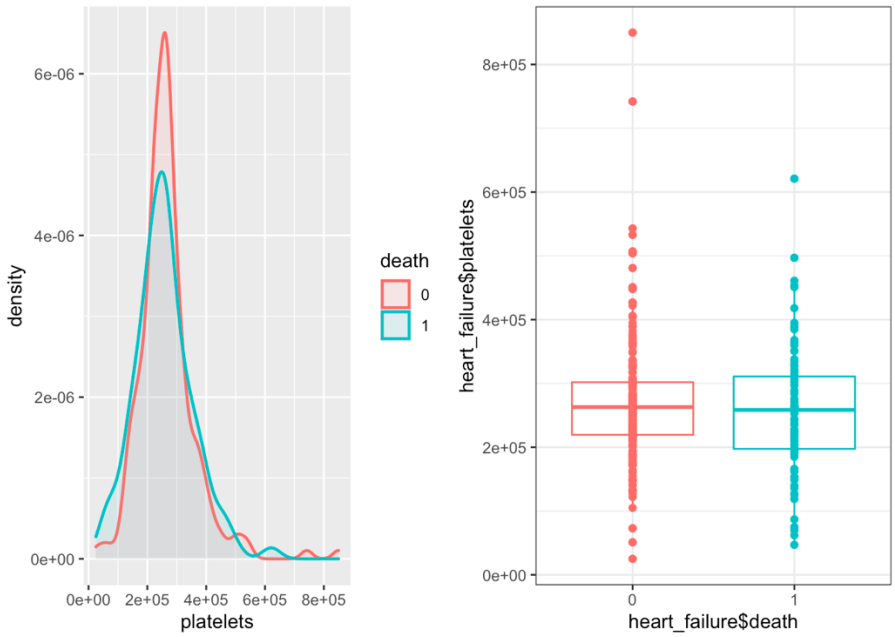

```
##  [1] 2.7 9.4 4.0 5.8 3.0 3.5 2.3 3.0 4.4 6.8 2.2 2.7 2.3 2.9 2.5 2.3 3.2 3.7 3.4
## [20] 6.1 2.5 2.4 2.5 3.5 9.0 5.0 2.4 2.7 3.8
```



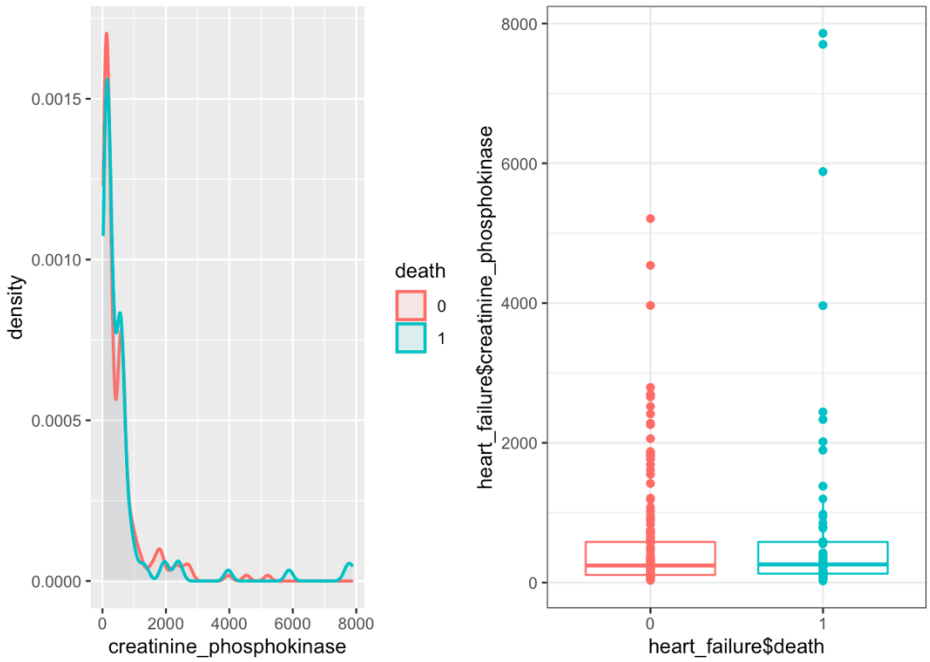
```
## [1] 116 121 124 113
```




```
##  [1] 454000  47000 451000 461000 497000 621000 850000 507000 448000  75000

## [11]  70000  73000 481000 504000  62000 533000  25100 451000  51000 543000

## [21] 742000
```



```
##  [1] 7861 2656 1380 3964 7702 5882 5209 1876 1808 4540 1548 1610 22
61 1846 2334

## [16] 2442 3966 1419 1896 1767 2281 2794 2017 2522 2695 1688 1820 20
60 2413
```



Cómo podemos ver existen valores extremos, es decir valores por debajo del mínimo y por encima del máximo, sin embargo al tratarse de características físicas médicas, como por ejemplo el número de plaquetas en sangre se pueden dar por aprobados.

ANÁLISIS DE DATOS

SELECCIÓN DE LOS DATOS A ESTUDIAR

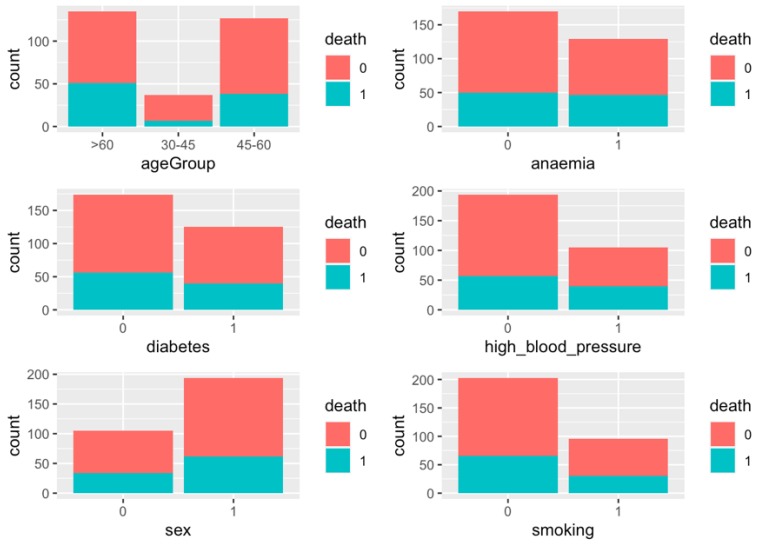
Hemos hecho una división entre el estudio de atributos binarios y atributos cuantitativos. A continuación, extraemos las estadísticas de los atributos binarios en relación con la clase death:

1. Normalizamos los datos de age para obtener una vista más comprensible:

```
heart_failure$age <- as.integer(heart_failure$age)

heart_failure<- heart_failure %>% mutate(ageGroup = ifelse(age <=30, "<=30",
                                                  ifelse(age>30 & age <=45, "30-45",
                                                         ifelse(age>45 & age <=60,"45-60",
                                                                ">60"))))

heart_failure$ageGroup <- as.factor(heart_failure$ageGroup)
```

Extraemos los datos:

```
#convertimos vaiables binarias a factor
newattr <- c("anaemia",
             "diabetes",
             "high_blood_pressure",
             "sex",
             "smoking",
             "death")
heart_failure[newattr] <- lapply(heart_failure[newattr], factor)

g1 <- ggplot(heart_failure, aes(ageGroup)) + geom_bar(aes(fill = death))
g2 <- ggplot(heart_failure, aes(anaemia)) + geom_bar(aes(fill = death))
g3 <- ggplot(heart_failure, aes(diabetes)) + geom_bar(aes(fill = death))
g4 <- ggplot(heart_failure, aes(high_blood_pressure)) + geom_bar(aes(fill = death))
g5 <- ggplot(heart_failure, aes(sex)) + geom_bar(aes(fill = death))
g6 <- ggplot(heart_failure, aes(smoking)) + geom_bar(aes(fill = death))
grid.arrange(g1, g2,g3, g4, g5, g6, nrow = 3)
```



Podemos observar por ejemplo que el número de pacientes fallecidos de sexo femenino es mucho mayor a la del sexo masculino, teniendo en cuenta el número de muestra que poseen cada uno. Lo mismo sucede en el caso de los fumadores, el porcentaje de pacientes fumadores fallecidos es más alto que de los que no lo son.

COMPROBACIÓN DE LA NORMALIDAD Y HOMOGENEIDAD

Comprobamos la distribución:

```
alpha = 0.05
col.names = colnames(heart_failure)
for (i in 1:ncol(heart_failure)) {
  if (i == 1) cat("Variables que no siguen una distribución normal:\n")
    if (is.integer(heart_failure[,i]) | is.numeric(heart_failure[,i])) {
      p_val = ad.test(heart_failure[,i])$p.value
      if (p_val < alpha) {
        cat(col.names[i])
        # Format output
        if (i < ncol(heart_failure) - 1) cat(", ")
        if (i %% 3 == 0) cat("\n") }
} }
```

```
## Variables que no siguen una distribución normal:
## age, anaemia, creatinine_phosphokinase,
## diabetes, ejection_fraction, high_blood_pressure,
## platelets, serum_creatinine, serum_sodium,
## sex, smoking, time,
## death
```

Como hemos comprobado que nuestras variables no tienen una distribución normal podemos aplicar el test de Fligner-Killen para estudiar la homogeneidad de varianzas. Esta prueba define la hipótesis nula como la igualdad de varianzas entre atributos.

```
a <- heart_failure[heart_failure$smoking == 0,"death"]
b <- heart_failure[heart_failure$smoking == 1,"death"]
fligner.test(x = list(a,b))
```

```
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  list(a, b)
## Fligner-Killeen:med chi-squared = 0.047485, df = 1, p-value = 0.8275
```

```
a1 <- heart_failure[heart_failure$sex == 0,"death"]
b1 <- heart_failure[heart_failure$sex == 1,"death"]
fligner.test(x = list(a1,b1))
```

```
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  list(a1, b1)
## Fligner-Killeen:med chi-squared = 0.0055521, df = 1, p-value = 0.9406
```

Como hemos obtenido un p-value > 0,05, podemos aceptar la hipótesis de que las varianzas de ambas muestras son homogéneas.

PRUEBAS ESTADÍSTICAS

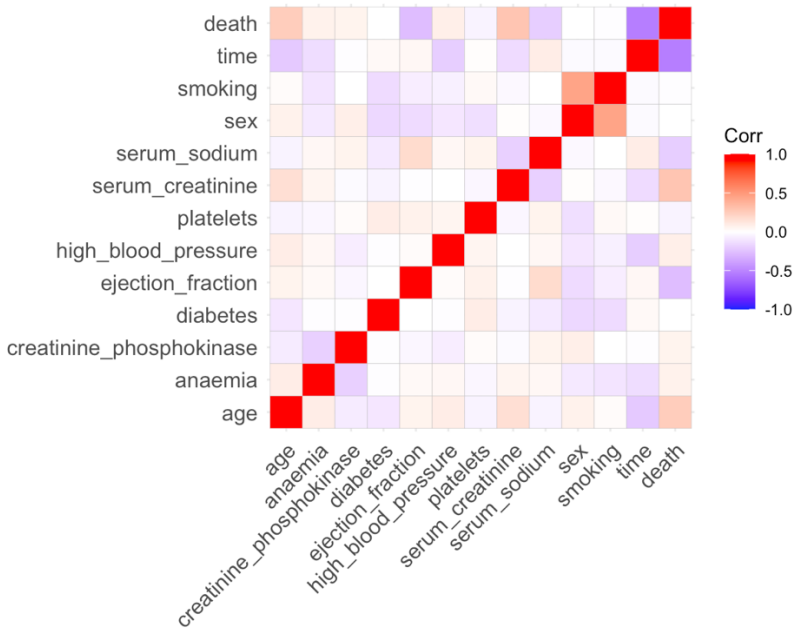
Para averiguar la relación de los atributos con el atributo death, vamos a calcular la correlación de cada uno de estos.

```
#CORRELACION
r <- cor(df, use="complete.obs")
round(r,2)
```

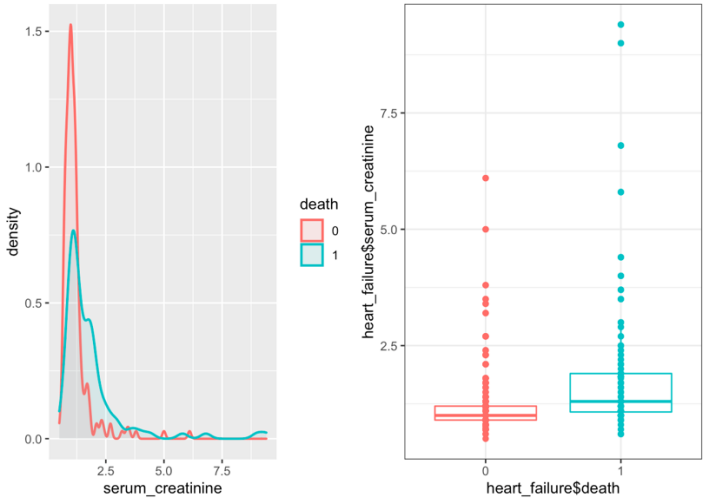
```
##                              death
## age                           0.25
## anaemia                       0.07
## creatinine_phosphokinase      0.06
## diabetes                      0.00
## ejection_fraction            -0.27
## high_blood_pressure           0.08
## platelets                    -0.05
## serum_creatinine              0.29
## serum_sodium                 -0.20
## sex                           0.00
## smoking                      -0.01
```

```
## time                    -0.53
## death                    1.00
```

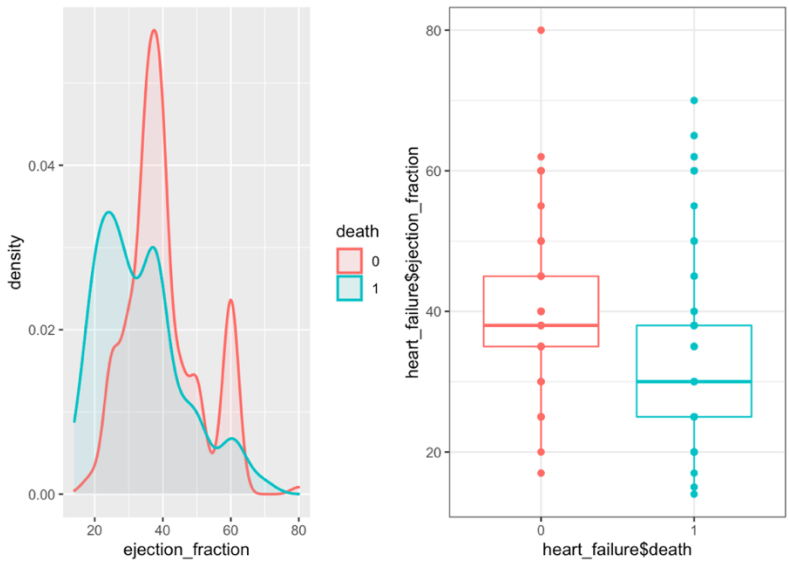
```
library(ggcorrplot)
ggcorrplot(r)
```



Podemos observar gracias al gráfico y a los resultados que el atributo más relacionado con la clase death, es el serum_creatinine, el cual indica el nivel de creatinina en la sangre. Como hemos podido ver además en la gráfica de la caja y bigotes: En esta relación, las personas fallecidas poseen un alto nivel de creatinina, cuanto más incrementa este valor, mayor es la gravedad del enfermo.



También podemos sacar la correlación negativa, aquí encontramos a los atributos del tiempo y ejection_fraction, que indica el porcentaje de sangre que sale cuando el corazón se contrae. El primero lo vamos a omitir, porque el tiempo de seguimiento, es algo relativo, pero el segundo es un factor médico que aporta más información. Podemos observar, como los pacientes con un valor bajo de esta medida tienen más probabilidades de fallecer.



MODELO DE REGRESIÓN LOGÍSTICA

Generalmente no se sugiere la regresión lineal para la clasificación binaria porque podría generar valores por debajo de 0 o por encima de 1. Por esta razón decidimos realizar el estudio con el modelo de regresión logística, que por el contrario es recomendada para los casos de clasificación binaria.

```
library(boot)
df <- heart_failure[, -14]
## 80% de las muestras
smp_size <- floor(0.8 * nrow(df))
set.seed(123)
train_i <- sample(seq_len(nrow(df)), size = smp_size)
## definimos los conjuntos de test y entrenamiento
train <- df[train_i, ]
test  <- df[-train_i, ]
# usamos la función glm para generar el modelo
```

```
model <- glm(death ~ ., data = train, family = binomial)

# extract the model summary

summary(model)
```

```
##
## Call:
## glm(formula = death ~ ., family = binomial, data = train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.4644  -0.4677  -0.1528   0.3352   2.2098
##
## Coefficients:
##                             Estimate Std. Error z value Pr(>|z|)
## (Intercept)               1.169e+01  7.018e+00   1.666  0.09571 .
## age                       5.644e-02  1.978e-02   2.853  0.00433 **
## anaemia1                  7.019e-02  4.300e-01   0.163  0.87033
## creatinine_phosphokinase  2.160e-04  2.115e-04   1.022  0.30699
## diabetes1                 3.593e-01  4.274e-01   0.841  0.40061
## ejection_fraction        -8.354e-02  1.998e-02  -4.180 2.91e-05 ***
## high_blood_pressure1      6.363e-02  4.240e-01   0.150  0.88070
## platelets                -1.639e-06  2.230e-06  -0.735  0.46247
## serum_creatinine          8.572e-01  2.154e-01   3.980 6.89e-05 ***
## serum_sodium             -7.817e-02  4.851e-02  -1.611  0.10708
## sex1                     -4.641e-01  4.953e-01  -0.937  0.34881
## smoking1                 -1.971e-01  4.933e-01  -0.400  0.68948
## time                     -2.723e-02  4.231e-03  -6.436 1.23e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 301.89  on 238  degrees of freedom
## Residual deviance: 152.07  on 226  degrees of freedom
## AIC: 178.07
##
## Number of Fisher Scoring iterations: 6
```

A continuación, vamos a ver qué valor obtendríamos de un modelo que considera los atributos más significativos, en este caso: ejection fraction y serum_creatine, que son coinciden con los atributos que comentamos en el apartado anterior.

```
model2 <- glm(data = train, death ~ serum_creatinine + ejection_fraction, family = "binomial")

summary(model2)
```

Aplicamos el modelo chi cuadrado para comparar la distribución observada de los datos con una distribución esperada de los datos.

```
dev2 <- model2$deviance

nullDev2 <- model2$null.deviance

modelChi2 <- nullDev2 - dev2

modelChi2
```

Como obtenemos un valor positivo quiere decir que la devianza del modelo es menor que la devianza nula por lo que las variables del modelo mejoran la predicción de la variable respuesta. Sin embargo, es un valor alto, lo cual implica que existan residuos sin explicación.

```
## [1] 44.29762
```

Ahora calcularemos la significación

```
chidf2 <- model2$df.null - model2$df.residual
chisq.prob2 <- 1 - pchisq(modelChi2, chidf2)
chisq.prob2
```

Como el valor es < 0.05, quiere decir que nuestro modelo es significativamente bueno prediciendo si un paciente sobrevive o no en función de estos 2 atributos, el porcentaje de sangre que sale del corazón cada vez que se contrae y la cantidad de creatina sérica en sangre.

```
## [1] 2.403771e-10
```

CONCLUSIONES

La evaluación de este dataset, nos ha permitido verificar mediante diferentes pruebas analíticas, cuáles eran las variables más significativas, es decir, que influyen sobre el fallecimiento de una persona con insuficiencia cardiaca, también hemos podido gracias a las mismas obtener un modelo predictivo, con la finalidad de poder prevenir a través de características específicas si el paciente se encuentra en estado de gravedad.

Hemos realizado la limpieza de datos, corroborando que no existan datos nulos o inconsistentes que puedan influir en nuestros resultados, a continuación, hemos comprobado la existencia de outliers, pero que finalmente son aceptables ya que son valores que perfectamente pueden darse en el mundo real.