Pràctica 2

Guillem Campo, Aleix Yébenes

Juny 2022

Descripció de la pràctica

L'objectiu d'aquesta activitat serà el tractament d'un dataset, que pot ser el creat a la pràctica 1 o bé qualsevol dataset lliure disponible a Kaggle (https://www.kaggle.com). Alguns exemples de dataset amb els que podeu treballar són:

- Red Wine Quality (https://www.kaggle.com/uciml/red-wine-quality-cortez-et-al-2009).
- Titanic: Machine Learning from Disaster (https://www.kaggle.com/c/titanic).

L'últim exemple correspon a una competició activa a Kaggle de manera que, opcionalment, podeu aprofitar el treball realitzat durant la pràctica per entrar en aquesta competició.

Seguint les principals etapes d'un projecte analític, les diferents tasques a realitzar (i justificar) són les següents:

1. Descripció del dataset. Perquè és important i quina pregunta/problema pretén respondre?
2. Integració i selecció de les dades d'interès a analitzar. Pot ser el resultat d'addicionar diferents datasets o una subselecció útil de les dades originals, en base a l'objectiu que es vulgui aconseguir.
3. Neteja de les dades. 3.1. Les dades contenen zeros o elements buits? Gestiona cadascun d'aquests casos. 3.2. Identifica i gestiona els valors extrems.
4. Anàlisi de les dades. Tipologia i cicle de vida de les dades Pràctica 2 pàg 2 4.1. Selecció dels grups de dades que es volen analitzar/comparar (p. e., si es volen comparar grups de dades, quins són aquests grups i quins tipus d'anàlisi s'aplicaran?). 4.2. Comprovació de la normalitat i homogeneïtat de la variància. 4.3. Aplicació de proves estadístiques per comparar els grups de dades. En funció de les dades i de l'objectiu de l'estudi, aplicar proves de contrast d'hipòtesis, correlacions, regressions, etc. Aplicar almenys tres mètodes d'anàlisi diferents.
5. Representació dels resultats a partir de taules i gràfiques. Aquest apartat es pot respondre al llarg de la pràctica, sense la necessitat de concentrar totes les representacions en aquest punt de la pràctica.
6. Resolució del problema. A partir dels resultats obtinguts, quines són les conclusions? Els resultats permeten respondre al problema?
7. Codi: Cal adjuntar el codi, preferiblement en R, amb el que s'ha realitzat la neteja, anàlisi i representació de les dades. Si ho preferiu, també podeu treballar en Python.

Descripció del dataset

L'enfonsament del Titanic és un dels naufragis més famosos de la història. El 15 d'abril de 1912, durant el seu viatge inaugural, el RMS Titanic, àmpliament considerat "inenfonsable", es va enfonsar després de xocar amb un iceberg. Malauradament, no hi havia suficients bots salvavides per a tots els que estaven a bord, cosa que va provocar la mort de 1.502 dels 2.224 passatgers i la tripulació. Tot i que hi va haver

algun element de sort involucrat en la supervivència, sembla que alguns grups de persones tenien més probabilitats de sobreviure que altres. En base a això volem saber: "quin tipus de persones tenien més probabilitats de sobreviure?" utilitzant dades de passatgers. Això ens pot donar certa informació envers a propers aconteixements semblants i com es podria actuar amb aquesta informació, trobar tipus de passatgers susceptibles a perdre la vida o a sobreviure. El conjunt de dades amb el que treballarem s'ha obtingut mitjançant un enllaç a kaggle i esta dividit en dues parts, les dades de train i les dades de test. Les dues parts contenen les mateixes variables, exceptuant que test no té la variable de si els passatgers van sobreviure o no.

```
#https://cran.r-project.org/web/packages/ggplot2/index.html
if (!require('ggplot2')) install.packages('ggplot2'); library('ggplot2')
#https://cran.r-project.org/web/packages/dplyr/index.html
if (!require('dplyr')) install.packages('dplyr'); library('dplyr')
if (!require('VIM')) install.packages('VIM'); library('VIM')
if (!require('epiDisplay')) install.packages('epiDisplay'); library('epiDisplay')
if (!require('randomForest')) install.packages('randomForest'); library('randomForest')
if (!require('pROC')) install.packages('pROC'); library('pROC')
```

Primer es llegeix el dataset 'train.csv', que és el set d'entrenament, per posteriorment llegir 'test.csv'i provar el model predictiu.

```
dades <- read.csv('train.csv',stringsAsFactors = FALSE)
filas=dim(dades)[1]
```

```
dim(dades)
```

```
## [1] 891  12
```

El dataset d'entrenament està format per 891 files amb 12 columnes, amb els següents tipus de dades:

```
str(dades)
```

```
## 'data.frame':    891 obs. of  12 variables:
##  $ PassengerId: int  1 2 3 4 5 6 7 8 9 10 ...
##  $ Survived   : int  0 1 1 1 0 0 0 0 1 1 ...
##  $ Pclass     : int  3 1 3 1 3 3 1 3 3 2 ...
##  $ Name       : chr  "Braund, Mr. Owen Harris" "Cumings, Mrs. John Bradley (Florence Briggs Thayer)"
##  $ Sex        : chr  "male" "female" "female" "female" ...
##  $ Age        : num  22 38 26 35 35 NA 54 2 27 14 ...
##  $ SibSp      : int  1 1 0 1 0 0 0 3 0 1 ...
##  $ Parch      : int  0 0 0 0 0 0 0 1 2 0 ...
##  $ Ticket     : chr  "A/5 21171" "PC 17599" "STON/O2. 3101282" "113803" ...
##  $ Fare       : num  7.25 71.28 7.92 53.1 8.05 ...
##  $ Cabin      : chr  "" "C85" "" "C123" ...
##  $ Embarked   : chr  "S" "C" "S" "S" ...
```

Els atributs del dataset són els següents:

- PassengerId: Número de passatger
- Survived: Sobreviscut (valor booleà)
- Pclass: Classe (1 = primera, 2 = segona, 3 = tercera)

- Name: Nom del passatger
- Sex: Gènere del passatger
- Age: Edat del passatger
- SibSp: Nombre de germans/es a bord del Titanic
- Parch: Nombre de pares o fills a bord del Titanic
- Ticket: Número de tiquet
- Fare: Preu del bitllet del passatger
- Cabin: Número de camarot
- Embarked: Port on ha embarcat (C = Cherbourg, Q = Queenstown, S = Southampton)

```
summary(dades)
```

```
##   PassengerId       Survived          Pclass          Name
##  Min.   :  1.0   Min.   :0.0000   Min.   :1.000   Length:891
##  1st Qu.:223.5   1st Qu.:0.0000   1st Qu.:2.000   Class :character
##  Median :446.0   Median :0.0000   Median :3.000   Mode  :character
##  Mean   :446.0   Mean   :0.3838   Mean   :2.309
##  3rd Qu.:668.5   3rd Qu.:1.0000   3rd Qu.:3.000
##  Max.   :891.0   Max.   :1.0000   Max.   :3.000
##
##      Sex                 Age            SibSp           Parch
##  Length:891         Min.   : 0.42   Min.   :0.000   Min.   :0.0000
##  Class :character   1st Qu.:20.12   1st Qu.:0.000   1st Qu.:0.0000
##  Mode  :character   Median :28.00   Median :0.000   Median :0.0000
##                     Mean   :29.70   Mean   :0.523   Mean   :0.3816
##                     3rd Qu.:38.00   3rd Qu.:1.000   3rd Qu.:0.0000
##                     Max.   :80.00   Max.   :8.000   Max.   :6.0000
##                     NA's   :177
##     Ticket               Fare           Cabin             Embarked
##  Length:891         Min.   :  0.00   Length:891         Length:891
##  Class :character   1st Qu.:  7.91   Class :character   Class :character
##  Mode  :character   Median : 14.45   Mode  :character   Mode  :character
##                     Mean   : 32.20
##                     3rd Qu.: 31.00
##                     Max.   :512.33
##
```

Integració i selecció de les dades d'interès a analitzar.

Com que el dataset de Kaggle proporciona totes les dades disponibles dels passatgers, no s'ha contemplat cap dataset addicional.

S'ha seleccionat un subset de les dades originals per ometre d'entrada els atributs PassengerId i Ticket, ja que no proporcionen cap informació que es pugui relacionar amb la supervivència dels passatgers. En el següent apartat s'avalua si es prescindeix d'algun altre atribut.

```
dades<-dades[,-c(1,9)]
```

Preprocés

En primer lloc s'avalua si hi ha valors buits o NA.

```
colSums(is.na(dades))
```

```
## Survived   Pclass     Name      Sex      Age    SibSp    Parch     Fare
##        0        0        0        0      177        0        0        0
##    Cabin Embarked
##        0        0
```

```
colSums(dades=="")
```

```
## Survived   Pclass     Name      Sex      Age    SibSp    Parch     Fare
##        0        0        0        0       NA        0        0        0
##    Cabin Embarked
##      687        2
```

Principalment hem vist que la variable Age contenia molts valors NA, per ser més precissos contenia 177 valors NA de 891 registres.

Vist això, hem de decidir que fer amb aquests valors, els podriem eliminar pero perdriem registres importants que ens donen informació valuosa. De manera que s'ha optat per implementar un mètode d'imputació de valors basat en la similitud o diferencia entre els registres, anomenat "k-NN-imputation" o k veins més propers. Hem escollit aquest mètode ja que els registres guarden certa relació, tot i que sempre es millor treballar amb dades aproximades que valors buits, ja que tindrem menys marge d'error.

```
suppressWarnings(suppressMessages(library(VIM)))
dades$Age<-kNN(dades)$Age
```

```
summary(dades[,"Age"])
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.42   21.00   28.50   29.77   39.00   80.00
```

Com es pot observar, no hi ha outliers (ja que les edats estan compreses entre 0 i 80), de manera que no es realitza cap operació addicional amb aquest atribut.

```
colSums(is.na(dades))
```

```
## Survived   Pclass     Name      Sex      Age    SibSp    Parch     Fare
##        0        0        0        0        0        0        0        0
##    Cabin Embarked
##        0        0
```

```
colSums(dades=="")
```

```
## Survived   Pclass     Name      Sex      Age    SibSp    Parch     Fare
##        0        0        0        0        0        0        0        0
##    Cabin Embarked
##      687        2
```

Podem veure que ara la variable Age ja no te valors buits ni valors NA, però s'ens afegeix un altre problema. La variable Cabin té 687 registres buits, de manera que obviarem aquesta variable ja que es una variable que no es pot aproximar o predir, perquè no disposem de la informació necessària per fer-ho.

```
dades<-dades[,-c(9)]
```

Per acabar amb el preprocés, es transforma la variable dicotòmica Survived de (0,1) a ("No","Yes"), i es creen dues variables per estudiar l'edat: el segment d'edat (de 10 en 10 anys) i la variable binària de major o menor d'edat (Adult = 0, 1).

```
dades$Survived[(dades$Survived==1)] <- "Yes"
dades$Survived[(dades$Survived==0)] <- "No"
dades["segment_edat"] <- cut(dades$Age,
                             breaks = c(0,10,20,30,40,50,60,70,100),
                             labels = c("0-9", "10-19", "20-29", "30-39","40-49","50-59","60-69","70-79
dades["Adult"] <- cut(dades$Age, breaks = c(0,17.5,100), labels = c(0,1))
```

A continuació es previsualitzen les primeres files de les dades després del preprocés i s'escriuen en un nou fitxer .csv

```
head(dades)
```

```
##   Survived Pclass                                                Name    Sex
## 1       No      3                             Braund, Mr. Owen Harris   male
## 2      Yes      1 Cumings, Mrs. John Bradley (Florence Briggs Thayer) female
## 3      Yes      3                              Heikkinen, Miss. Laina female
## 4      Yes      1        Futrelle, Mrs. Jacques Heath (Lily May Peel) female
## 5       No      3                            Allen, Mr. William Henry   male
## 6       No      3                                    Moran, Mr. James   male
##   Age SibSp Parch    Fare Embarked segment_edat Adult
## 1  22     1     0  7.2500        S        20-29     1
## 2  38     1     0 71.2833        C        30-39     1
## 3  26     0     0  7.9250        S        20-29     1
## 4  35     1     0 53.1000        S        30-39     1
## 5  35     0     0  8.0500        S        30-39     1
## 6  33     0     0  8.4583        Q        30-39     1
```

```
write.csv(dades,"dades_clean.csv")
```

Exploració de les dades

Per a un coneixement major sobre les dades, que permeti el seu posterior anàlisi, s'utilitzen eines de visualització com ggplot i grid:

```
if(!require(grid)){
    install.packages('grid', repos='http://cran.us.r-project.org')
    library(grid)
}
if(!require(gridExtra)){
    install.packages('gridExtra', repos='http://cran.us.r-project.org')
    library(gridExtra)
}

if (!require('corrplot')) install.packages('corrplot'); library('corrplot')
```

En primer lloc, s'analitza la distribució d'algunes de les variables més rellevants: sexe, grup d'edat, classe i supervivència.

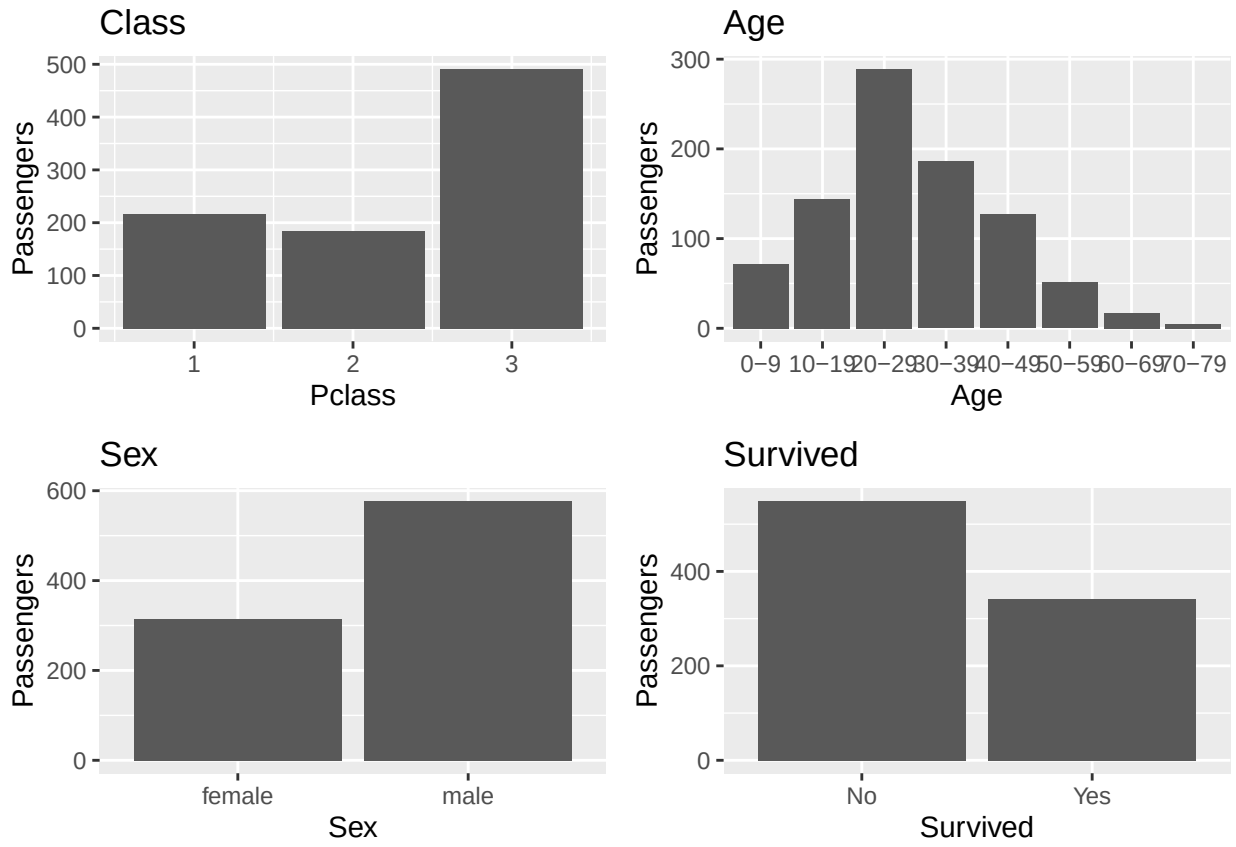
```
plotbyClass<-ggplot(dades,aes(Pclass))+geom_bar() +labs(x="Pclass", y="Passengers")+
  guides(fill=guide_legend(title=""))+ scale_fill_manual(values=c("blue","#008000"))+ggtitle("Class")

plotbyAge<-ggplot(dades,aes(segment_edat))+geom_bar() +labs(x="Age", y="Passengers")+
  guides(fill=guide_legend(title=""))+ scale_fill_manual(values=c("blue","#008000"))+ggtitle("Age")

plotbySex<-ggplot(dades,aes(Sex))+geom_bar() +labs(x="Sex", y="Passengers")+
  guides(fill=guide_legend(title=""))+ scale_fill_manual(values=c("blue","#008000"))+ggtitle("Sex")

plotbySurvived<-ggplot(dades,aes(Survived))+geom_bar() +labs(x="Survived", y="Passengers")+
  guides(fill=guide_legend(title=""))+ scale_fill_manual(values=c("blue","#008000"))+ggtitle("Survived")

grid.arrange(plotbyClass,plotbyAge,plotbySex,plotbySurvived,ncol=2)
```

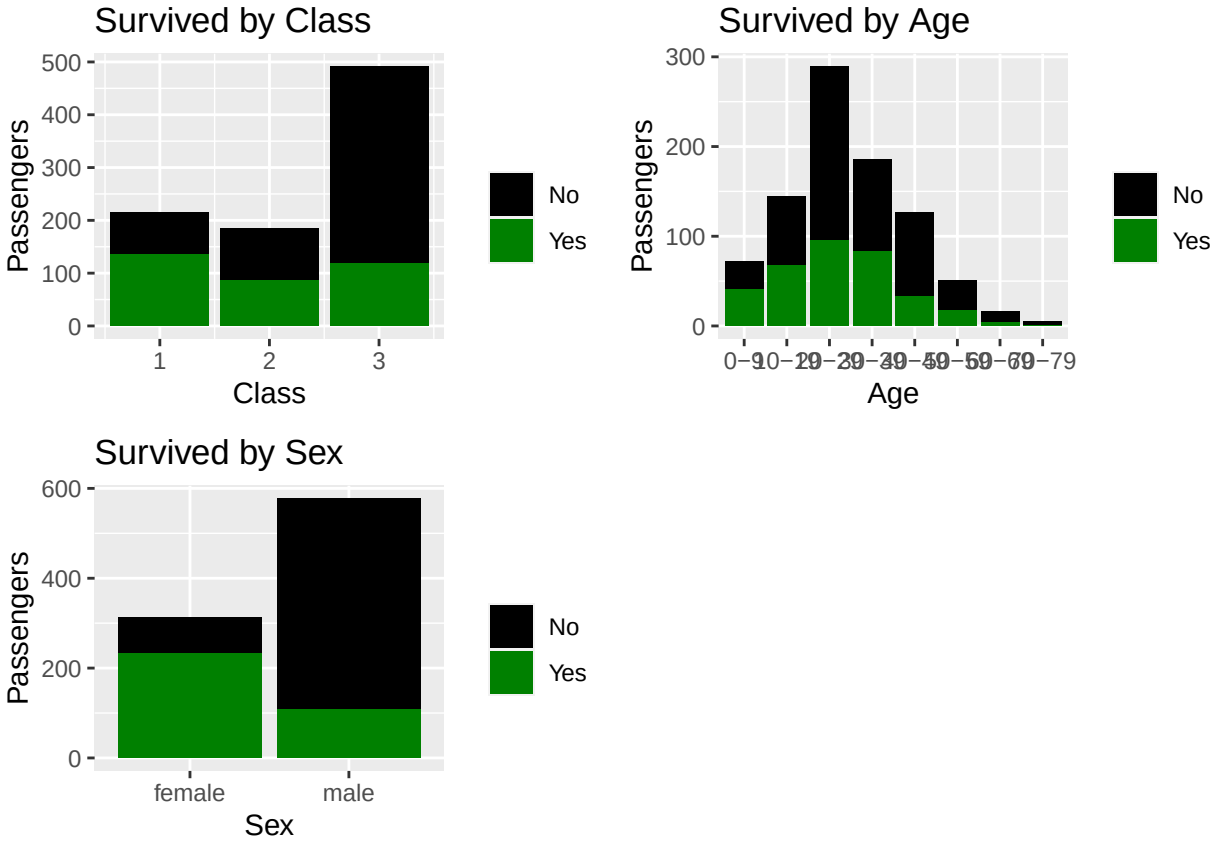


Com es pot veure, la major part dels passatgers eren homes, viatjaven en tercera classe, tenien una edat d'uns 30 anys i no van sobreviure.

És interessant veure, sobre els mateixos gràfics, la proporció de passatgers que van sobreviure respecte el valor de cada atribut, ja que dona una primera aproximació als determinants de la supervivència dels passatgers:

```
grid.newpage()
plotbyClass<-ggplot(dades,aes(Pclass,fill=Survived))+geom_bar() +labs(x="Class", y="Passengers")+
  guides(fill=guide_legend(title=""))+ scale_fill_manual(values=c("black","#008000"))+ggtitle("Survived
```

```
plotbyAge<-ggplot(dades,aes(segment_edat,fill=Survived))+geom_bar() +labs(x="Age", y="Passengers")+
  guides(fill=guide_legend(title=""))+ scale_fill_manual(values=c("black","#008000"))+ggtitle("Survived
plotbySex<-ggplot(dades,aes(Sex,fill=Survived))+geom_bar() +labs(x="Sex", y="Passengers")+
  guides(fill=guide_legend(title=""))+ scale_fill_manual(values=c("black","#008000"))+ggtitle("Survived
plotbyEmbarked<-ggplot(dades,aes(Embarked,fill=Survived))+geom_bar() +labs(x="Embarked", y="Passengers"
  guides(fill=guide_legend(title=""))+ scale_fill_manual(values=c("black","#008000"))+ggtitle("Survived
grid.arrange(plotbyClass,plotbyAge,plotbySex,ncol=2)
```



Aquests gràfics evidencien fets coneguts popularment, com que es va prioritzar el salvament de dones i nens, i que els passatgers amb més supervivència van ser els de les classes superiors.

Obtenim ara una matriu de percentatges de freqüència. Veiem, per exemple que la probabilitat de sobreviure si es va embarcar en "C" és d'un 55.35%, o si es va embarcar en "Q" és d'un 38.96%

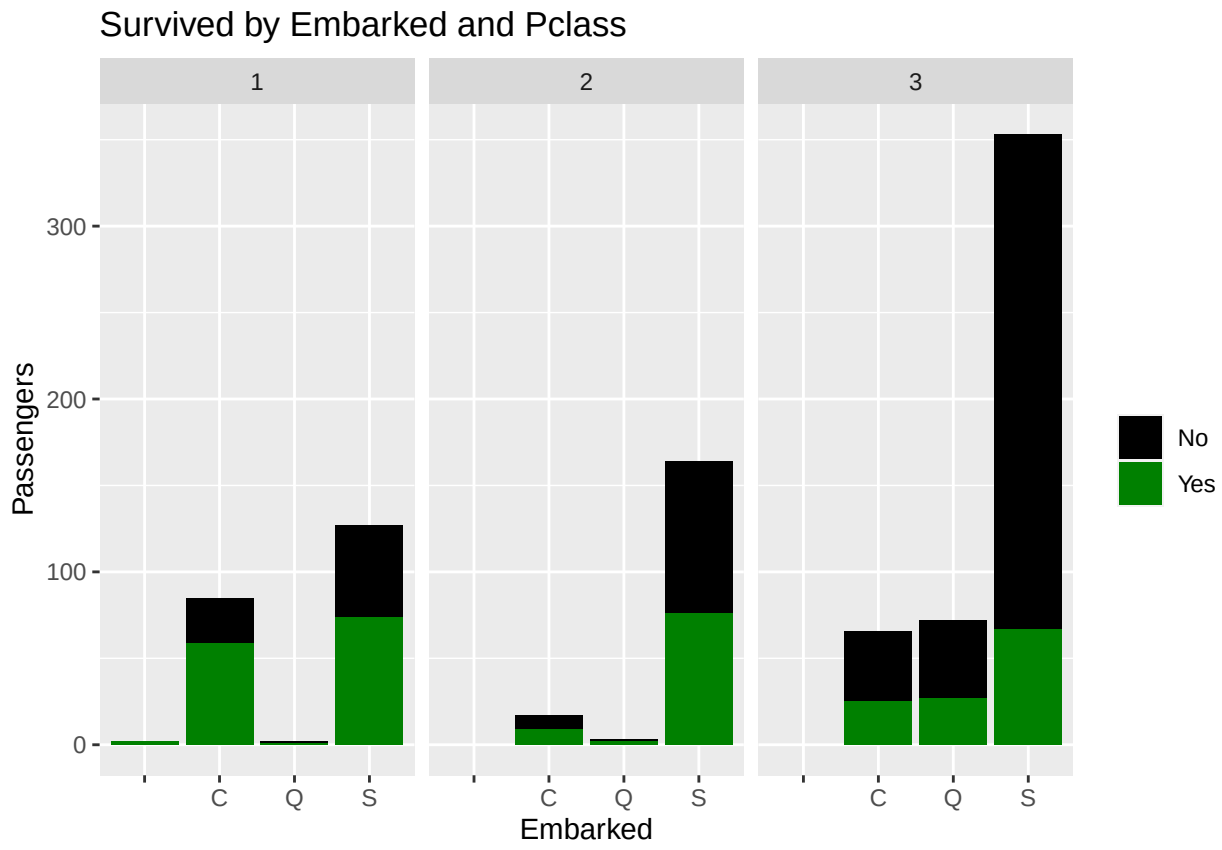
```
t<-table(dades[1:filas,]$Embarked,dades[1:filas,]$Survived)
for (i in 1:dim(t)[1]){
    t[i,]<-t[i,]/sum(t[i,])*100
}
t
```

```
##
##            No       Yes
##       0.00000 100.00000
##   C  44.64286  55.35714
##   Q  61.03896  38.96104
##   S  66.30435  33.69565
```

Vegem ara com en un mateix gràfic de freqüències podem treballar amb 3 variables: Embarked, Survived i class.

Mostrem el gràfic d'embarcats per Pclass:

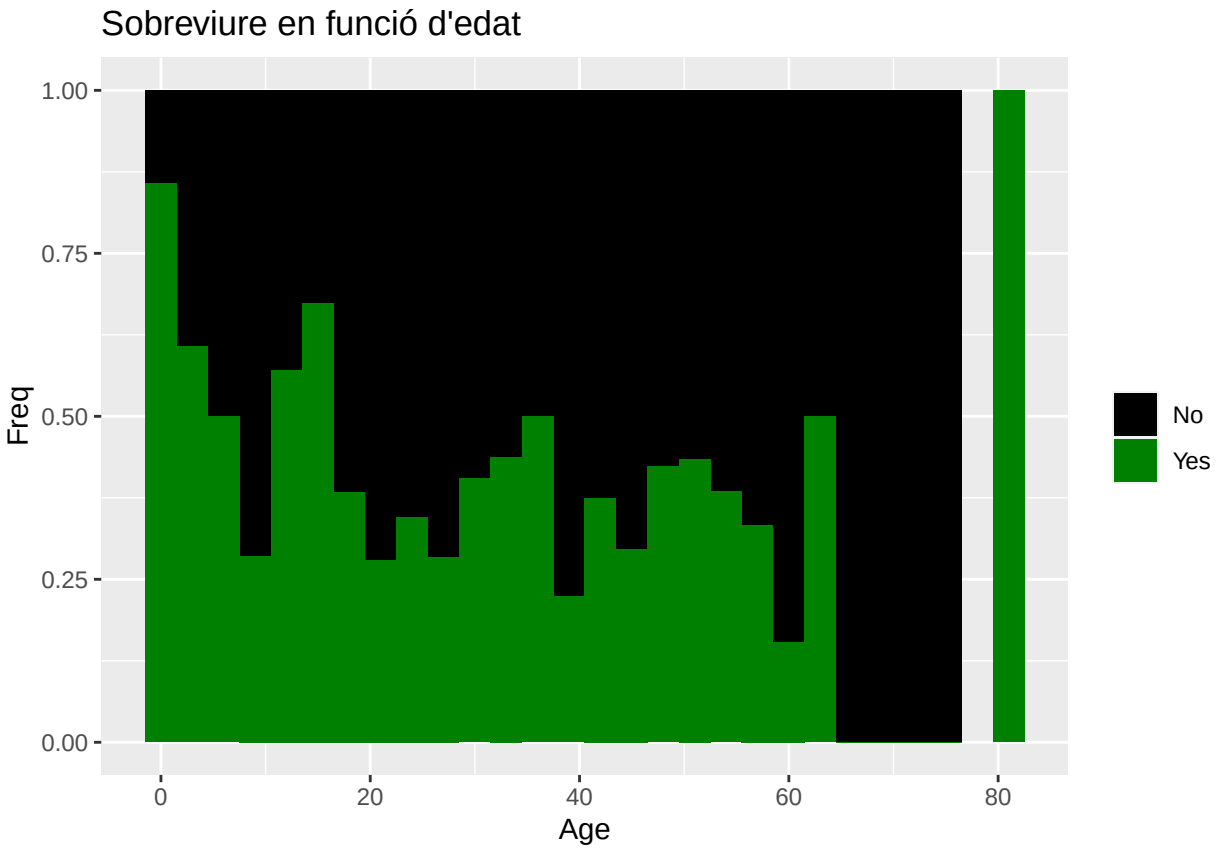
```
plotbyEmbarkedc<-ggplot(dades,aes(Embarked,fill=Survived))+geom_bar()+facet_wrap(~Pclass) +
  labs(x="Embarked", y="Passengers")+ guides(fill=guide_legend(title=""))+ scale_fill_manual(values=c("
plotbyEmbarkedc
```



Aquí ja podem extreure molta informació. Es pot apreciar com els passatgers de 1a classe van sobreviure més en comparació amb la 2a i 3a classe. O que a Southampton hi havia el percentatge de gent més pobre ja que la 3a classe predomina allà.

```
ggplot(data = dades[!is.na(dades[1:filas,]$Age),],aes(x=Age,fill=Survived))+
  geom_histogram(binwidth = 3,position="fill")+ylab("Freq")+ guides(fill=guide_legend(title=""))+
  scale_fill_manual(values=c("black","#008000"))+ggtitle("Sobreviure en funció d'edat")
```

```
## Warning: Removed 2 rows containing missing values (geom_bar).
```



Sembla que els nens varen tenir més possibilitat de salvar-se. La gent de 80 anys no la comptarem com que tenien més possibilitat de salvar-se ja que es tracta d'un sol registre, i com podem veure a partir dels 60 i pocs ja no sobrevivia cap passatger.

Anàlisi de dades

Contrast d'hipòtesi

En primer lloc, es fa un contrast amb la següent hipòtesi:

$$H_0 : \mu_{homes} = \mu_{dones}$$
$$H_1 : \mu_{homes} < \mu_{dones}$$

El que es vol mirar en aquest contrast és si la supervivència en homes i dones va ser o no la mateixa. Primer es fa un test sobre la variància:

```
dades$Survived[(dades$Survived=="Yes")] <- 1
dades$Survived[(dades$Survived=="No")] <- 0
dades$Survived <- as.numeric(dades$Survived)

var.test(dades$Survived[dades$Sex=="male"], dades$Survived[dades$Sex=="female"] )
```

```
##
```

```
##  F test to compare two variances
##
## data:  dades$Survived[dades$Sex == "male"] and dades$Survived[dades$Sex == "female"]
## F = 0.7993, num df = 576, denom df = 313, p-value = 0.02218
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
##  0.6558632 0.9685628
## sample estimates:
## ratio of variances
##           0.799295
```

S'accepta la hipòtesi d'igualitat de variances en les dues poblacions. Per tant, s'aplica un test de dues mostres independents sobre la mitjana amb variança desconeguda igual.

```
t.test(dades$Survived[dades$Sex=="male"], dades$Survived[dades$Sex=="female"],alternative="less")
```

```
##
##  Welch Two Sample t-test
##
## data:  dades$Survived[dades$Sex == "male"] and dades$Survived[dades$Sex == "female"]
## t = -18.672, df = 584.43, p-value < 2.2e-16
## alternative hypothesis: true difference in means is less than 0
## 95 percent confidence interval:
##        -Inf -0.5043259
## sample estimates:
## mean of x mean of y
## 0.1889081 0.7420382
```

Amb el p-value obtingut ($p<0.05$), es descarta la hipòtesi nul · la, i es conclou, tal i com se suposava en la secció de visualització de les dades, que la supervivència en homes va ser menor que en dones.

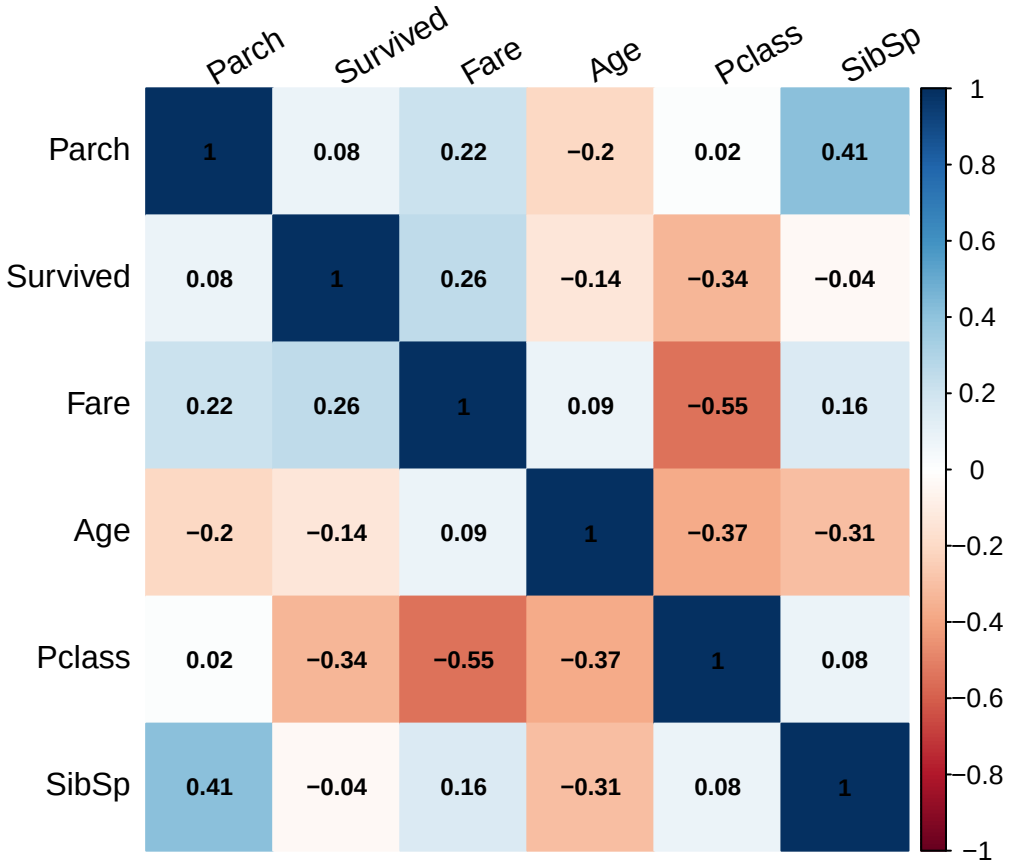
Anàlisi de correlacions

Matriu de correlacions:

```
nums <- unlist(lapply(dades, is.numeric))

res<-cor(dades[,nums])

corrplot(res,method="color",tl.col="black", tl.srt=30, order = "AOE",
         number.cex=0.75,sig.level = 0.01, addCoef.col = "black")
```



Podem veure com les variables més correlacionades son Pclass-Fare i Sibsp-Parch. Tot i així no és una correlació molt gran.

Regressió logística

Apliquem la regressió logística per tal de predir les probabilitats de supervivència dels passatgers:

```
logit_model_1 <- glm(formula=Survived~Pclass+Sex+Adult, data=dades, family=binomial)
summary(logit_model_1)
```

```
##
## Call:
## glm(formula = Survived ~ Pclass + Sex + Adult, family = binomial,
##     data = dades)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.6468  -0.6561  -0.3897   0.7110   2.2871
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)   4.5852     0.4045  11.334  < 2e-16 ***
## Pclass       -1.1129     0.1131  -9.844  < 2e-16 ***
## Sexmale      -2.5367     0.1867 -13.589  < 2e-16 ***
## Adult1       -1.2492     0.2408  -5.188 2.13e-07 ***
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1186.66  on 890  degrees of freedom
## Residual deviance:  799.67  on 887  degrees of freedom
## AIC: 807.67
##
## Number of Fisher Scoring iterations: 5
```

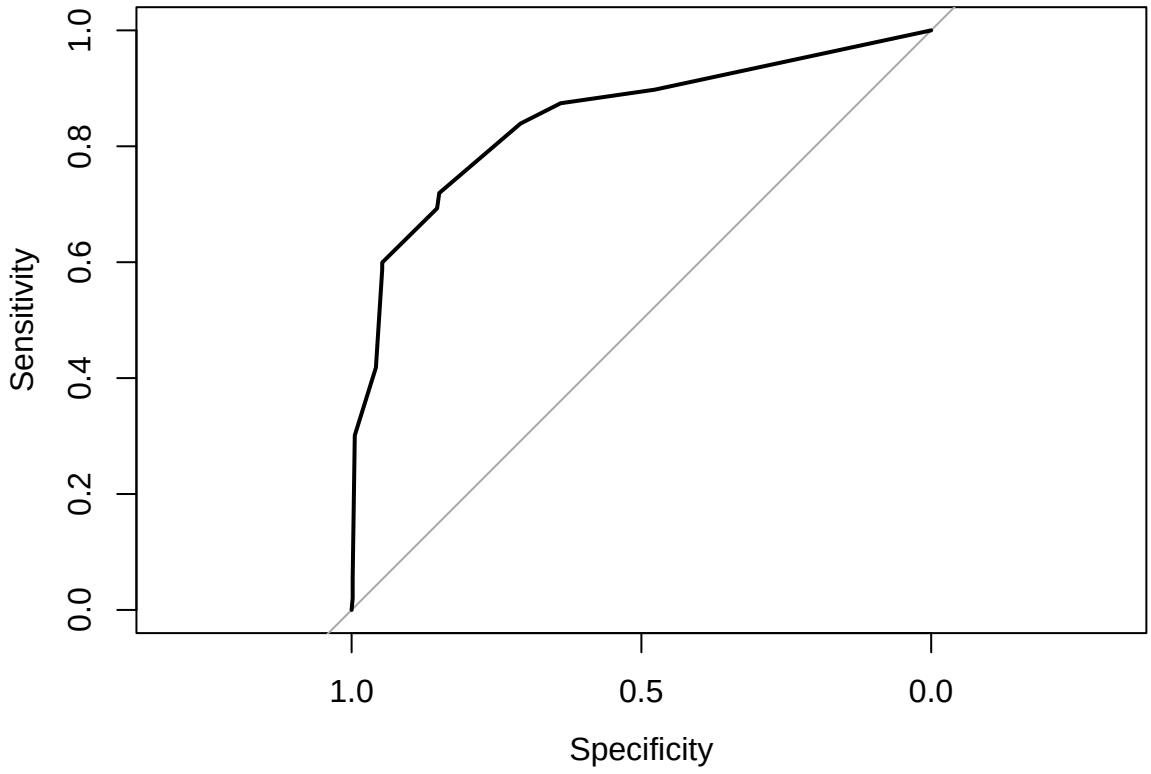
Fem una predicció de les dades de train per analitzar la corba roc i saber si el nostre model discrimina be les dades.

```
pr = predict(logit_model_1, dades, type="response")
r=roc(dades$Survived,pr, data=dades)
```

```
## Setting levels: control = 0, case = 1
```

```
## Setting direction: controls < cases
```

```
plot(r)
```



```
auc(r)
```

```
## Area under the curve: 0.8481
```

Podem veure que l'Area under the curve: 0.8481, seguint la regla sabem que si 0,6 <= AUROC < 0,8, podem dir que el model no acaba de discriminar de manera gaire adequada.

```
new <- read.csv('test.csv',stringsAsFactors = FALSE)
summary(new)
```

```
##   PassengerId         Pclass          Name               Sex
##  Min.   : 892.0   Min.   :1.000   Length:418         Length:418
##  1st Qu.: 996.2   1st Qu.:1.000   Class :character   Class :character
##  Median :1100.5   Median :3.000   Mode  :character   Mode  :character
##  Mean   :1100.5   Mean   :2.266
##  3rd Qu.:1204.8   3rd Qu.:3.000
##  Max.   :1309.0   Max.   :3.000
##
##       Age            SibSp            Parch           Ticket
##  Min.   : 0.17   Min.   :0.0000   Min.   :0.0000   Length:418
##  1st Qu.:21.00   1st Qu.:0.0000   1st Qu.:0.0000   Class :character
##  Median :27.00   Median :0.0000   Median :0.0000   Mode  :character
##  Mean   :30.27   Mean   :0.4474   Mean   :0.3923
##  3rd Qu.:39.00   3rd Qu.:1.0000   3rd Qu.:0.0000
##  Max.   :76.00   Max.   :8.0000   Max.   :9.0000
##  NA's   :86
##       Fare            Cabin             Embarked
##  Min.   :  0.000   Length:418         Length:418
##  1st Qu.:  7.896   Class :character   Class :character
##  Median : 14.454   Mode  :character   Mode  :character
##  Mean   : 35.627
##  3rd Qu.: 31.500
##  Max.   :512.329
##  NA's   :1
```

Principalment hem vist que la variable Age contenia molts valors NA.

Vist això, hem de decidir que fer amb aquests valors, els podriem eliminar pero perdriem registres importants que ens donen informació valuosa. De manera que s'ha optat per implementar un mètode d'imputació de valors basat en la similitud o diferencia entre els registres, anomenat "k-NN-imputation" o k veins més propers. Hem escollit aquest mètode ja que els registres guarden certa relació, tot i que sempre es millor treballar amb dades aproximades que valors buits, ja que tindrem menys marge d'error.

```
suppressWarnings(suppressMessages(library(VIM)))
new$Age<-kNN(new)$Age
```

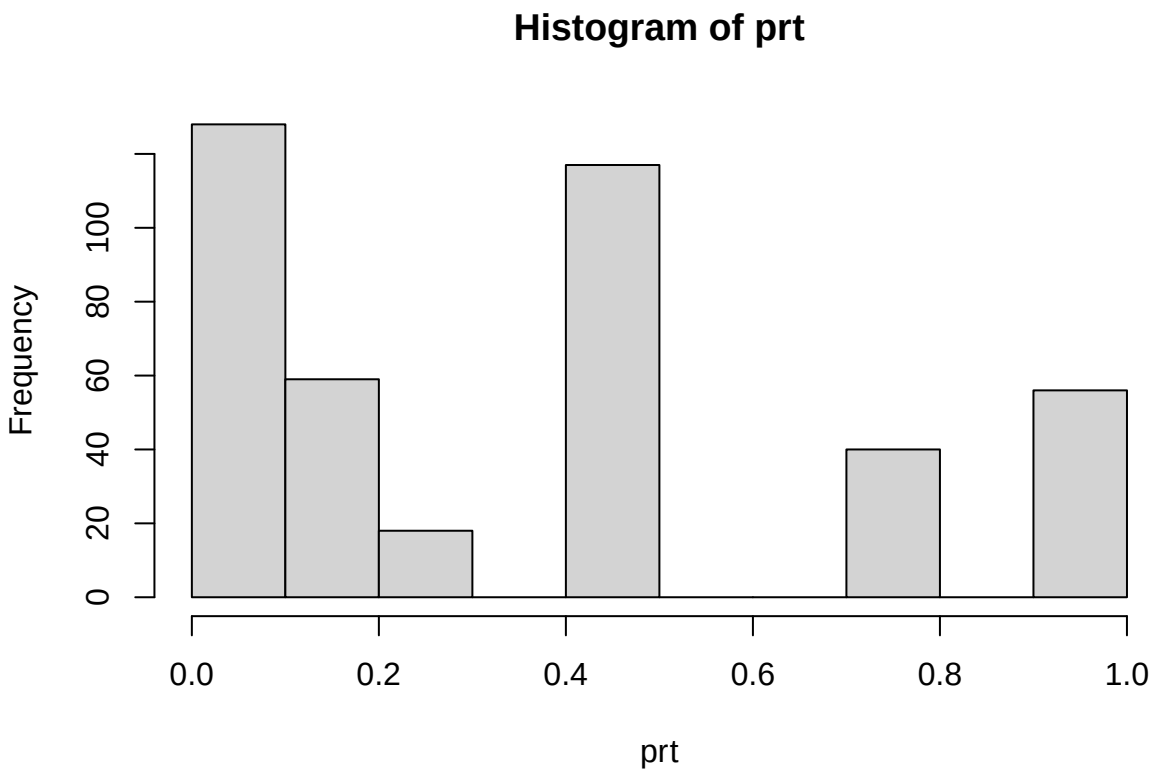
```
summary(new)
```

```
##   PassengerId         Pclass          Name               Sex
##  Min.   : 892.0   Min.   :1.000   Length:418         Length:418
##  1st Qu.: 996.2   1st Qu.:1.000   Class :character   Class :character
```

```
##  Median :1100.5   Median :3.000   Mode  :character   Mode  :character
##  Mean   :1100.5   Mean   :2.266
##  3rd Qu.:1204.8   3rd Qu.:3.000
##  Max.   :1309.0   Max.   :3.000
##
##       Age             SibSp            Parch           Ticket
##  Min.   : 0.17   Min.   :0.0000   Min.   :0.0000   Length:418
##  1st Qu.:22.00   1st Qu.:0.0000   1st Qu.:0.0000   Class :character
##  Median :27.00   Median :0.0000   Median :0.0000   Mode  :character
##  Mean   :29.46   Mean   :0.4474   Mean   :0.3923
##  3rd Qu.:36.00   3rd Qu.:1.0000   3rd Qu.:0.0000
##  Max.   :76.00   Max.   :8.0000   Max.   :9.0000
##
##       Fare            Cabin             Embarked
##  Min.   :  0.000   Length:418         Length:418
##  1st Qu.:  7.896   Class :character   Class :character
##  Median : 14.454   Mode  :character   Mode  :character
##  Mean   : 35.627
##  3rd Qu.: 31.500
##  Max.   :512.329
##  NA's   :1
```

```
new["segment_edat"] <- cut(new$Age,
                           breaks = c(0,10,20,30,40,50,60,70,100),
                           labels = c("0-9", "10-19", "20-29", "30-39","40-49","50-59","60-69","70-79"))
new["Adult"] <- cut(new$Age, breaks = c(0,17.5,100), labels = c(0,1))
```

```
prt = predict(logit_model_1, new, type="response")
hist(prt)
```



Podem veure que en el conjunt de test hi ha més probabilitats de morir que de sobreviure, i es concentra un gran nombre de passatgers entre el 40% i 50% de possibilitats de sobreviure.

Modelització predictiva

```
dades$Survived[(dades$Survived==1)] <- "Yes"
dades$Survived[(dades$Survived==0)] <- "No"
```

Per a la futura avaluació del random forest, és necessari dividir el conjunt de dades en un conjunt d'entrenament i un conjunt de prova. El conjunt d'entrenament és el subconjunt del conjunt original de dades utilitzat per a construir un primer model; i el conjunt de prova, el subconjunt del conjunt original de dades utilitzat per a avaluar la qualitat del model.

```
set.seed(666)
y <- dades[,1]
X <- dades[,c(2,4,5,6,7,8,9,11)]
```

De manera dinàmica podem definir una manera de separar les dades en funció d'un paràmetre, en aquest cas del "split_prop". Definim un paràmetre que controla el split de manera dinàmica en el test.

```
split_prop <- 4
indexes = sample(1:nrow(dades), size=floor(((split_prop-1)/split_prop)*nrow(dades)))
trainX<-X[indexes,]
trainy<-y[indexes]
```

```
testX<-X[-indexes,]
testy<-y[-indexes]
```

Després d'una extracció aleatòria de casos és altament recomanable efectuar una anàlisi de dades mínim per a assegurar-nos de no obtenir classificadors esbiaixats pels valors que conté cada mostra.

```
summary(trainX)
```

```
##      Pclass          Sex                 Age            SibSp
##  Min.   :1.000   Length:668         Min.   : 0.67   Min.   :0.0000
##  1st Qu.:1.750   Class :character   1st Qu.:21.00   1st Qu.:0.0000
##  Median :3.000   Mode  :character   Median :28.50   Median :0.0000
##  Mean   :2.293                      Mean   :29.64   Mean   :0.5269
##  3rd Qu.:3.000                      3rd Qu.:39.00   3rd Qu.:1.0000
##  Max.   :3.000                      Max.   :74.00   Max.   :8.0000
##      Parch             Fare           Embarked         Adult
##  Min.   :0.0000   Min.   :  0.000   Length:668         0:107
##  1st Qu.:0.0000   1st Qu.:  7.925   Class :character   1:561
##  Median :0.0000   Median : 14.500   Mode  :character
##  Mean   :0.3817   Mean   : 33.042
##  3rd Qu.:0.0000   3rd Qu.: 31.387
##  Max.   :5.0000   Max.   :512.329
```

```
summary(trainy)
```

```
##    Length     Class      Mode
##       668 character character
```

```
summary(testX)
```

```
##      Pclass          Sex                 Age            SibSp
##  Min.   :1.000   Length:223         Min.   : 0.42   Min.   :0.0000
##  1st Qu.:2.000   Class :character   1st Qu.:21.00   1st Qu.:0.0000
##  Median :3.000   Mode  :character   Median :29.00   Median :0.0000
##  Mean   :2.354                      Mean   :30.14   Mean   :0.5112
##  3rd Qu.:3.000                      3rd Qu.:40.00   3rd Qu.:1.0000
##  Max.   :3.000                      Max.   :80.00   Max.   :8.0000
##      Parch             Fare           Embarked         Adult
##  Min.   :0.0000   Min.   :  0.000   Length:223         0: 38
##  1st Qu.:0.0000   1st Qu.:  7.896   Class :character   1:185
##  Median :0.0000   Median : 13.000   Mode  :character
##  Mean   :0.3812   Mean   : 29.695
##  3rd Qu.:0.0000   3rd Qu.: 27.900
##  Max.   :6.0000   Max.   :512.329
```

```
summary(testy)
```

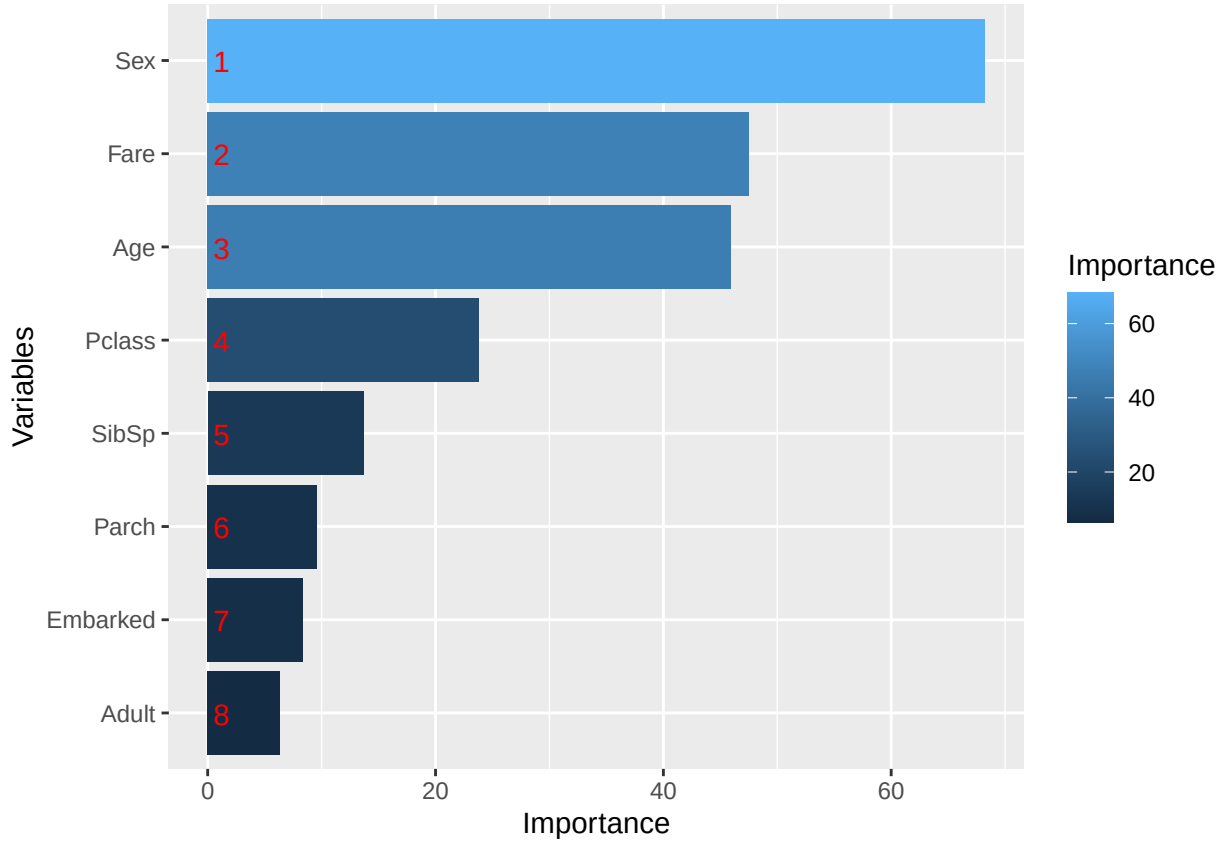
```
##    Length     Class      Mode
##       223 character character
```

Es crea el random forest usant les dades d'entrenament .

```
set.seed(754)
randForest <- randomForest(factor(trainy) ~ Pclass + Sex + Age + SibSp + Parch +
                                            Fare + Embarked + Adult,
                                            data = trainX)
```

Comprovem la importància de les variables :

```
importance    <- importance(randForest)
varImportance <- data.frame(Variables = row.names(importance),
                            Importance = round(importance[ ,'MeanDecreaseGini'],2))
rankImportance <- varImportance %>%
  mutate(Rank = paste0(dense_rank(desc(Importance))))
ggplot(rankImportance, aes(x = reorder(Variables, Importance),
    y = Importance, fill = Importance)) +
  geom_bar(stat='identity') +
  geom_text(aes(x = Variables, y = 0.5, label = Rank),
    hjust=0, vjust=0.55, size = 4, colour = 'red') +
  labs(x = 'Variables') +
  coord_flip()
```



És intereessant com el Sexe es la variable més important del dataset, i una de les variables que pensavem que tindria molta importància com es Pclass estigui a la 4a posició.

Una vegada tenim el model, podem comprovar la seva qualitat predient la classe per a les dades de prova que ens hem reservat al principi.

```
predicted_model  <- predict(randForest, testX)
print(sprintf("La precisión del árbol es: %.4f %%",100*sum(predicted_model == testy) / length(predicted
```

```
## [1] "La precisión del árbol es: 85.2018 %"
```

Quan hi ha poques classes, la qualitat de la predicció es pot analitzar mitjançant una matriu de confusió que identifica els tipus d'errors comesos.

```
mat_conf<-table(testy,Predicted=predicted_model)
mat_conf
```

```
##      Predicted
## testy  No Yes
##   No  119   8
##   Yes  25  71
```

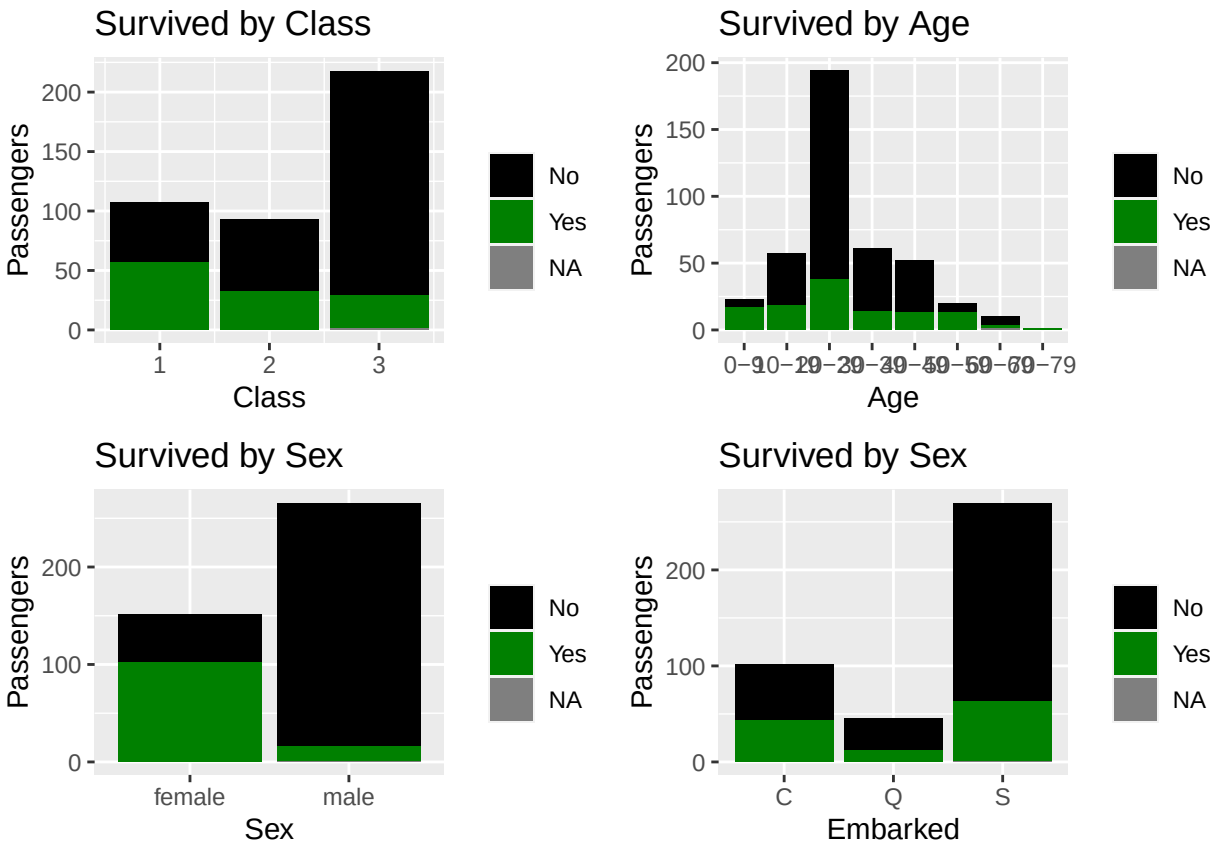
Fem la predicció del dataset de test, per tal de predir la variable survival.

```
predicted_model  <- predict(randForest, new)
new['Survived'] <- predicted_model
head(new)
```

```
##   PassengerId Pclass                                          Name    Sex  Age
## 1         892      3                              Kelly, Mr. James   male 34.5
## 2         893      3              Wilkes, Mrs. James (Ellen Needs) female 47.0
## 3         894      2                     Myles, Mr. Thomas Francis   male 62.0
## 4         895      3                               Wirz, Mr. Albert   male 27.0
## 5         896      3 Hirvonen, Mrs. Alexander (Helga E Lindqvist) female 22.0
## 6         897      3                     Svensson, Mr. Johan Cervin   male 14.0
##   SibSp Parch  Ticket    Fare Cabin Embarked segment_edat Adult Survived
## 1     0     0  330911  7.8292              Q        30-39     1       No
## 2     1     0  363272  7.0000              S        40-49     1       No
## 3     0     0  240276  9.6875              Q        60-69     1       No
## 4     0     0  315154  8.6625              S        20-29     1       No
## 5     1     1 3101298 12.2875              S        20-29     1       No
## 6     0     0    7538  9.2250              S        10-19     0       No
```

```
plotbyClass<-ggplot(new,aes(Pclass,fill=Survived))+geom_bar() +labs(x="Class", y="Passengers")+
  guides(fill=guide_legend(title=""))+ scale_fill_manual(values=c("black","#008000"))+ggtitle("Survived
plotbyAge<-ggplot(new,aes(segment_edat,fill=Survived))+geom_bar() +labs(x="Age", y="Passengers")+
  guides(fill=guide_legend(title=""))+ scale_fill_manual(values=c("black","#008000"))+ggtitle("Survived
plotbySex<-ggplot(new,aes(Sex,fill=Survived))+geom_bar() +labs(x="Sex", y="Passengers")+
  guides(fill=guide_legend(title=""))+ scale_fill_manual(values=c("black","#008000"))+ggtitle("Survived
plotbyEmbarked<-ggplot(new,aes(Embarked,fill=Survived))+geom_bar() +labs(x="Embarked", y="Passengers")+
  guides(fill=guide_legend(title=""))+ scale_fill_manual(values=c("black","#008000"))+ggtitle("Survived

grid.arrange(plotbyClass,plotbyAge,plotbySex,plotbyEmbarked,ncol=2)
```



Amb aquests gràfics podem veure la distribució de gent que va sobreviure o no en funció d'altres variables, i apreciem que segueixen una distribució molt semblant al observat en el dataset de train.

Conclusions

Amb els anàlisis realitzats, s'extreuen les següents conclusions:

- La ratio de supervivència de les dones va ser significativament superior a la dels homes, segons el contrast d'hipòtesi realitzat.
- El model de regressió logística obtingut no acaba de discriminar de manera gaire adequada.
- El factor més determinant en la supervivència dels passatgers del Titanic va ser el gènere, i també són significatius l'edat, preu del bitllet i classe (aquests dos últims estan correlacionats segons s'ha vist a la matriu de correlació).
- L'arbre obtingut té una precisió del 85%.