Pràctica 2: Neteja i anàlisi de les dades

Autor: Stefany Chóez Bolaños i Daniel Panadero Espinosa

Decembre 2021

Descripción del dataset

El dataset que utilitzarem en aquesta pràctica és Titanic: Machine Learning from Disaster (https://www.kaggle.com/c/titanic).

A continuació, es descriuen les variables del dataset:

PassengerId Id del passatger.

Name Noms dels passatgers.

Sex Factor de gènere (Masculí o Femení).

Age Valor numèric amb l'edat de les persones al dia de l'enfonsament.

Pclass Factor de la classe dels passatgers o el tipus de servei de la tripulació.

Embarked Factor del port d'embarcament

Ticket Valor numèric del número de tiquet.

Fare Valor numèric que representa el preu del tiquet.

SibSp Nombre de germans/cònjuges a bord del Titanic

Parch Nombre de pares/fills a bord del Titanic

Survived Factor que representa si la persona ha sobreviscut o no.

Cabin Valor numèric del nombre de la cabina.

A partir d'aquest conjunt de dades es planteja la problemàtica de determinar quines variables influeixen més a l'hora de sobreviure en l'embarcament. A més a més, es crearà un model de classificació per permetre predir quina persona sobreviu o quina no, en funció de les seves característiques.

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

Carreguem el fitxer de dades.

```
titanic_original <- read.csv('../Dataset/train.csv',stringsAsFactors = FALSE)
files=dim(titanic_original)[1]
files
```

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

Veiem que tenim 891 registres que es corresponen als viatgers i tripulació del Titànic i 12 variables que els caracteritzen.

Verifiquem l'estructura del joc de dades principal.

```
str(titanic_original)
```

```
## '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" ...
```

Passem les variables de tipus caràcter i numèric a tipus factor, ja que són classes.

```
titanic_original$Survived <- as.factor(titanic_original$Survived)
titanic_original$Pclass <- as.factor(titanic_original$Pclass)
titanic_original$Sex <- as.factor(titanic_original$Sex)
```

Anem ara a treure estadístiques bàsiques.

```
summary(titanic_original)
```

```
##   PassengerId    Survived Pclass      Name              Sex
##  Min.   :  1.0   0:549    1:216   Length:891         female:314
##  1st Qu.:223.5   1:342    2:184   Class :character   male  :577
##  Median :446.0            3:491   Mode  :character
##  Mean   :446.0
##  3rd Qu.:668.5
##  Max.   :891.0
##
##       Age            SibSp           Parch           Ticket
##  Min.   : 0.42   Min.   :0.000   Min.   :0.0000   Length:891
##  1st Qu.:20.12   1st Qu.:0.000   1st Qu.:0.0000   Class :character
##  Median :28.00   Median :0.000   Median :0.0000   Mode  :character
##  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
##       Fare           Cabin             Embarked
##  Min.   :  0.00   Length:891         Length:891
##  1st Qu.:  7.91   Class :character   Class :character
##  Median : 14.45   Mode  :character   Mode  :character
##  Mean   : 32.20
##  3rd Qu.: 31.00
##  Max.   :512.33
##
```

De la informació mostrada destaquem que el passatger més jove tenia 2 mesos i el més gran 80 anys. La mitjana d'edat la tenien en 29,7 anys. També podem observar el que es va pagar pel bitllet. Sibsp i parch també mostren dades interessants el viatger amb qui més familiar viatjava eren 8 germans o dona i 6 fills o pare/mare.

Eliminem les variables PassengerId, Name, Ticket i Cabin perquè no ens aporta informació rellevant per la nostra anàlisi.

```
titanic <- titanic_original[, c(2,3,5:8,10,12)]
```

Neteja de les dades.

Les dades amb zeros o elements buits.

Estadístiques de valors buits.

```
colSums(is.na(titanic))
```

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

```
colSums(titanic=="")
```

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

Assignem valor "NA" per als valors buits de la variable "Embarked".

```
titanic$Embarked[titanic$Embarked==""] <- "NA"
titanic$Embarked <- as.factor(titanic$Embarked)
```

Mostrem gràficament els valors NA:

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

```
## Loading required package: VIM
```

```
## Loading required package: colorspace
```

```
## Loading required package: grid
```

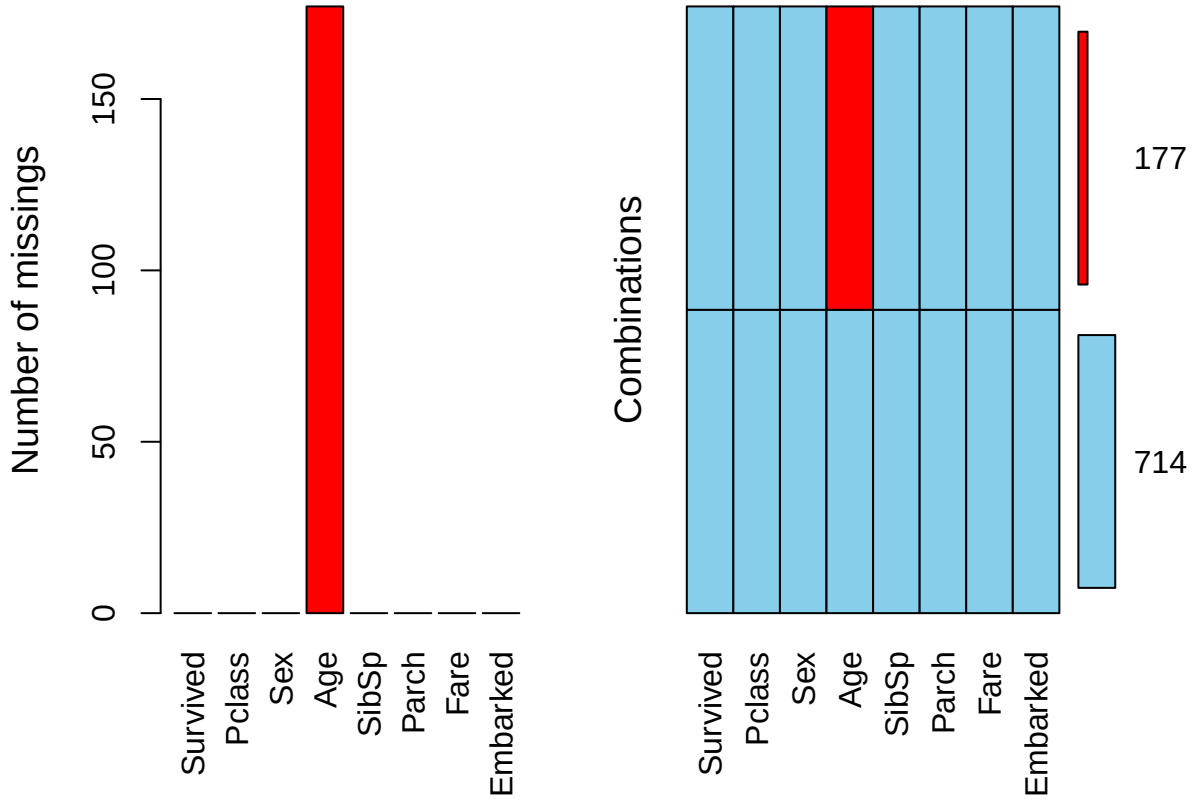
```
## VIM is ready to use.
```

```
## Suggestions and bug-reports can be submitted at: https://github.com/statistikat/VIM/issues
```

```
##
## Attaching package: 'VIM'
```

```
## The following object is masked from 'package:datasets':
##
##     sleep
```

```
aggr(titanic, prop = F, numbers = T)
```



Podem veure que només tenim una variable amb valors NA que és la variable "Age" i es pot observar que en total són 177 valors NA.

Utilitzem la funció kNN de la llibreria VIM per imputar els valors de NA de la variable "Age" i utilitzem la resta de variables per calcular els valors amb un K igual a 3.

```
titanic<- kNN( titanic, variable="Age",dist_var=c("Survived", "Pclass", "Sex", "SibSp", "Parch",
                                                  "Fare", "Embarked"), k=3, imp_var = FALSE)
```

Com podem observar, ja no tenim cap variable amb valors NA o buits:

```
colSums(is.na(titanic))
```

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

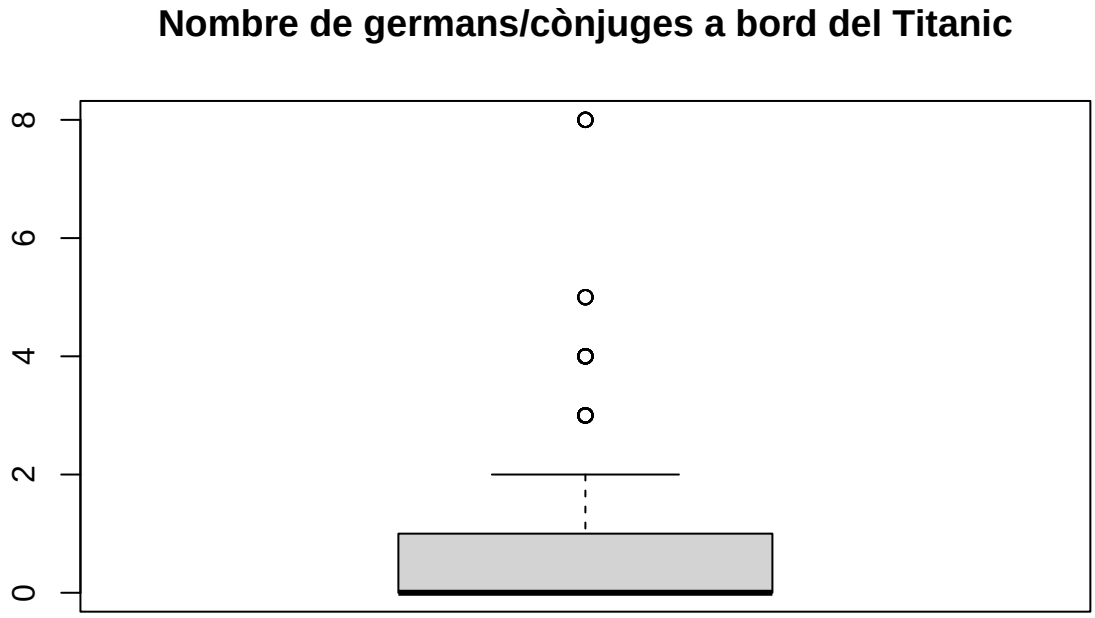
```
colSums(titanic=="")
```

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

Identificació i tractament de valors extrems.

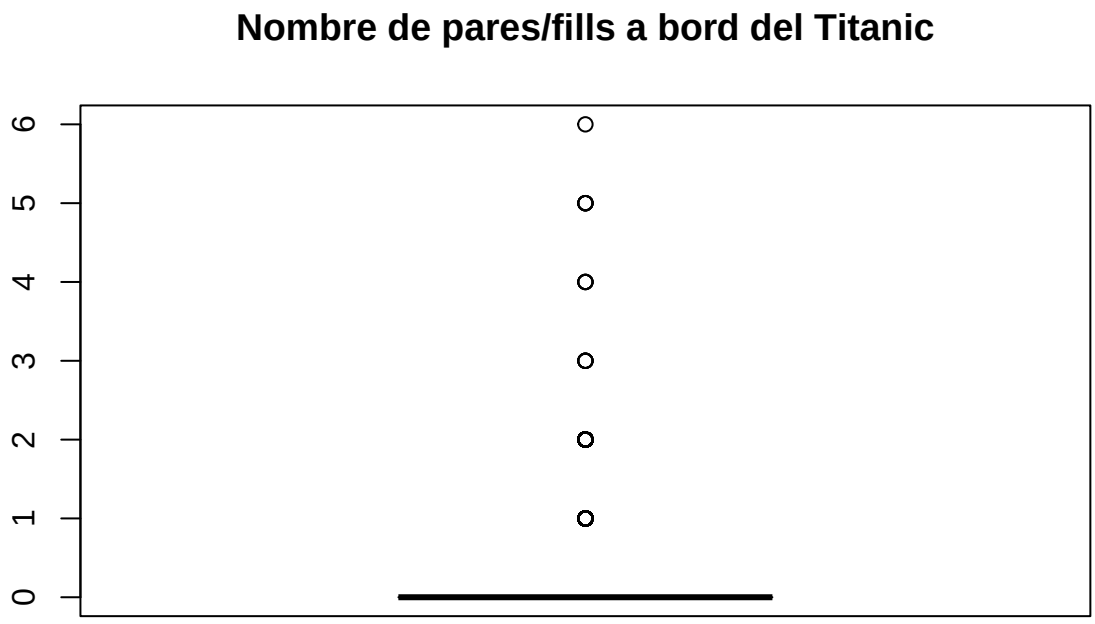
Realitzarem gràfics de les variables de tipus numèrics.

```
boxplot(titanic$SibSp,main="Nombre de germans/cònjuges a bord del Titanic")
```



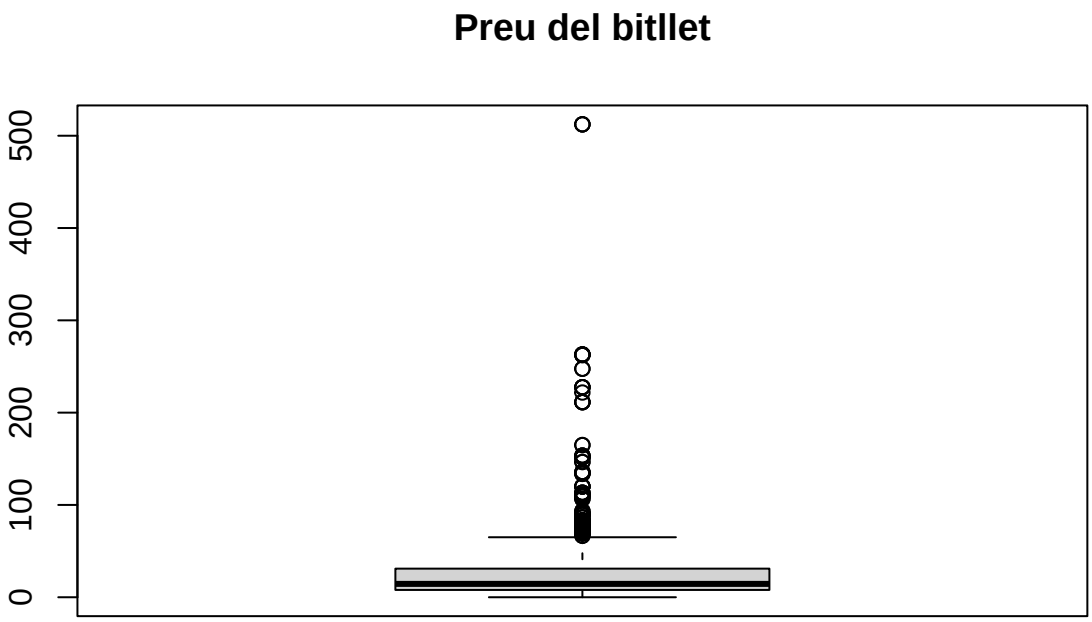
Com es pot observar, la majoria tenen entre 0 i 2 germans/cònjuges. De 3 a 8 són valors outliers, però són rellevants per l'anàlisi.

```
boxplot(titanic$Parch,main="Nombre de pares/fills a bord del Titanic")
```



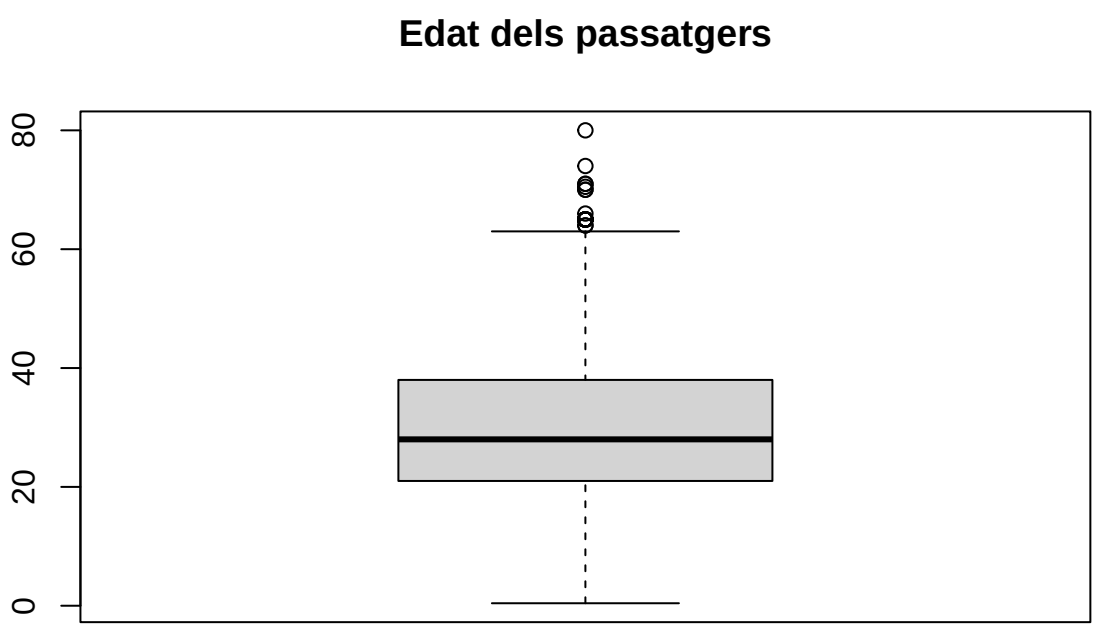
Com es pot observar, la majoria no tenen pares o fills. De 1 a 6 són valors outliers, però són rellevants per l'anàlisi.

```
boxplot(titanic$Fare,main="Preu del bitllet")
```



Com es pot observar, la mitjana son 32 de preu de bitllet i que de 100 a 500 tenim valors outliers.

```
boxplot(titanic$Age,main="Edat dels passatgers")
```



Com es pot observar, la mitjana son 30 anys i que la majoria d'edats està entre 20 i 40 anys. Entre 60 i 80 anys tenim valors outliers.

Anàlisi de les dades.

Selecció dels grups de dades que es volen analitzar/comparar (planificació dels anàlisis a aplicar).

A continuació, se seleccionen els grups dins del nostre conjunt de dades que poden resultar interessants per a analitzar i/o comparar.

```
# Agrupació per tipus de classe
titanic.class1 <- titanic[titanic$Pclass == "1",]
titanic.class2 <- titanic[titanic$Pclass == "2",]
titanic.class3 <- titanic[titanic$Pclass == "3",]
```

```
# Agrupació per port d'embarcació
titanic.Cherbourg <- titanic[titanic$Embarked == "C",]
titanic.Queenstown <- titanic[titanic$Embarked == "Q",]
titanic.Southampton <- titanic[titanic$Embarked == "S",]
```

```
# Agrupació per sexe
titanic.male <- titanic[titanic$Sex== "male",]
titanic.female <- titanic[titanic$Sex == "female",]
```

Comprovació de la normalitat i homogeneïtat de la variància

Per la comprovació de normalitat de les variables quantitatives utilitzarem les proves de normalitat de Kolmogorov-Smirnov i de Shapiro-Wilk, sent aquesta última més robusta.

```r
# Tests de la variable Age
ks.test(titanic$Age, pnorm, mean(titanic$Age),sd(titanic$Age))
```

```
## Warning in ks.test(titanic$Age, pnorm, mean(titanic$Age), sd(titanic$Age)): ties
## should not be present for the Kolmogorov-Smirnov test
```

```
##
##  One-sample Kolmogorov-Smirnov test
##
## data:  titanic$Age
## D = 0.074501, p-value = 0.0001013
## alternative hypothesis: two-sided
```

```r
shapiro.test(titanic$Age)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  titanic$Age
## W = 0.97352, p-value = 1.195e-11
```

```r
# Tests de la variable SibSp
ks.test(titanic$SibSp, pnorm, mean(titanic$SibSp),sd(titanic$SibSp))
```

```
## Warning in ks.test(titanic$SibSp, pnorm, mean(titanic$SibSp),
## sd(titanic$SibSp)): ties should not be present for the Kolmogorov-Smirnov test
```

```
##
##  One-sample Kolmogorov-Smirnov test
##
## data:  titanic$SibSp
## D = 0.36473, p-value < 2.2e-16
## alternative hypothesis: two-sided
```

```r
shapiro.test(titanic$SibSp)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  titanic$SibSp
## W = 0.51297, p-value < 2.2e-16
```

```r
# Tests de la variable Parch
ks.test(titanic$Parch, pnorm, mean(titanic$Parch),sd(titanic$Parch))
```

```
## Warning in ks.test(titanic$Parch, pnorm, mean(titanic$Parch),
## sd(titanic$Parch)): ties should not be present for the Kolmogorov-Smirnov test
```

```
##
##  One-sample Kolmogorov-Smirnov test
##
## data:  titanic$Parch
## D = 0.44298, p-value < 2.2e-16
## alternative hypothesis: two-sided
```

```r
shapiro.test(titanic$Parch)
```

```
##
##  Shapiro-Wilk normality test
```

```
##
## data:  titanic$Parch
## W = 0.53281, p-value < 2.2e-16
```

```
# Tests de la variable Fare
ks.test(titanic$Fare, pnorm, mean(titanic$Fare),sd(titanic$Fare))
```

```
## Warning in ks.test(titanic$Fare, pnorm, mean(titanic$Fare), sd(titanic$Fare)):
## ties should not be present for the Kolmogorov-Smirnov test
```

```
##
##  One-sample Kolmogorov-Smirnov test
##
## data:  titanic$Fare
## D = 0.28185, p-value < 2.2e-16
## alternative hypothesis: two-sided
```

```
shapiro.test(titanic$Fare)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  titanic$Fare
## W = 0.52189, p-value < 2.2e-16
```

Mirem si el p-valor és més gran que el nivell de significació alfa = 0.05. Si és més gran, acceptem la hipòtesi nul · la de normalitat i si és més petit, la rebutgem i podríem dir que no hi ha normalitat.

En el nostre cas tots els tests ens han donat inferiors al nivell de significació, per tant, rebutgem la hipòtesi nul · la i afirmem que les variables quantitatives no són normals, amb un nivell de confiança del 95%.

A continuació realitzarem el test de Fligner-Killeen per comprovar l'homogeneïtat de la variància.

```
# Test de la variable Fare - Sex
fligner.test(Fare ~ Sex, data = titanic)
```

```
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  Fare by Sex
## Fligner-Killeen:med chi-squared = 55.949, df = 1, p-value = 7.436e-14
```

```
# Test de la variable Fare - Pclass
fligner.test(Fare ~ Pclass, data = titanic)
```

```
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  Fare by Pclass
## Fligner-Killeen:med chi-squared = 365.8, df = 2, p-value < 2.2e-16
```

```
# Test de la variable Fare - Embarked
fligner.test(Fare ~ Embarked, data = titanic)
```

```
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  Fare by Embarked
## Fligner-Killeen:med chi-squared = 137.46, df = 3, p-value < 2.2e-16
```

En tots els casos obtenim un p-valor inferior a 0,05, per tant, rebutgem la hipòtesi nul · la i podem dir que les mostres no són homogènies, amb un 95% de confiança.

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.

Van sobreviure més les dones que els homes? La primera prova estadística que s'aplicarà consistirà en un test per a la diferència de dues proporcions per determinar si van sobreviure més les dones que els homes. Per fer-ho compararem les proporcions de les dues mostres.

$$H_0 : p_F < p_M H_1 : p_F >= p_M$$

On p és la proporció de passatgers que van sobreviure.

```
xf<-sum(titanic.female$Survived==1)
nf<-nrow(titanic.female)
xm<-sum(titanic.male$Survived==1)
nm<-nrow(titanic.male)
prop.test(c(xf,xm),c(nf,nm), alternative="greater")
```

```
##
##  2-sample test for equality of proportions with continuity correction
##
## data:  c(xf, xm) out of c(nf, nm)
## X-squared = 260.72, df = 1, p-value < 2.2e-16
## alternative hypothesis: greater
## 95 percent confidence interval:
##  0.5020113 1.0000000
## sample estimates:
##    prop 1    prop 2
## 0.7420382 0.1889081
```

El valor p és menor que alfa=0.05, estem en la zona de no acceptació de la hipòtesi nul · la. Per tant, podem afirmar que les diferències de proporcions són significativament diferents amb un nivell de confiança del 95%. Podem dir que van sobreviure més dones que homes.

Van sobreviure més els de primera classe que la resta? La segona prova estadística que s'aplicarà consistirà en un test per a la diferència de dues proporcions per determinar si van sobreviure més els de primera classe que la resta. Per fer-ho compararem les proporcions de les dues mostres.

$$H_0 : p_{1ra} < p_{23ra} H_1 : p_{1ra} >= p_{23ra}$$

On p és la proporció de passatgers que van sobreviure.

```
x1<-sum(titanic.class1$Survived==1)
n1<-nrow(titanic.class1)
x23<-sum(titanic.class2$Survived==1)+sum(titanic.class3$Survived==1)
n23<-nrow(titanic.class2)+nrow(titanic.class3)
prop.test(c(x1,x23),c(n1,n23), alternative="greater")
```

```
##
##  2-sample test for equality of proportions with continuity correction
##
## data:  c(x1, x23) out of c(n1, n23)
```

```
## X-squared = 71.466, df = 1, p-value < 2.2e-16
## alternative hypothesis: greater
## 95 percent confidence interval:
##  0.2599815 1.0000000
## sample estimates:
##    prop 1    prop 2
## 0.6296296 0.3051852
```

El valor p és menor que alfa=0.05, estem en la zona de no acceptació de la hipòtesi nul · la. Per tant, podem afirmar que les diferències de proporcions són significativament diferents amb un nivell de confiança del 95%. Podem dir que van sobreviure més els de 1a classe que la resta.

Realitzarem les correlacions entre diferents variables numèriques. Utilitzarem el coeficient de correlació de "Spearman", ja que no tenim variables amb distribució normal.

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

```
## Loading required package: tidyverse
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --
```

```
## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.5     v dplyr   1.0.7
## v tidyr   1.1.4     v stringr 1.4.0
## v readr   2.0.2     v forcats 0.5.1
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
# Correlació entre l'edat i el preu del bitllet.

cor.test(x = titanic$Age, y = titanic$Fare, method = "spearman")
```

```
## Warning in cor.test.default(x = titanic$Age, y = titanic$Fare, method =
## "spearman"): Cannot compute exact p-value with ties
```

```
##
##  Spearman's rank correlation rho
##
## data:  titanic$Age and titanic$Fare
## S = 105276223, p-value = 0.00138
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##       rho
## 0.1070051
```

```
# Correlació entre l'edat i la variable SibSp (germans/conjuges).

cor.test(x = titanic$SibSp, y = titanic$Age, method = "spearman")
```

```
## Warning in cor.test.default(x = titanic$SibSp, y = titanic$Age, method =
## "spearman"): Cannot compute exact p-value with ties
```

```
##
##  Spearman's rank correlation rho
##
## data:  titanic$SibSp and titanic$Age
## S = 142428654, p-value = 3.544e-10
## alternative hypothesis: true rho is not equal to 0
```

```
## sample estimates:
##        rho
## -0.2081366
```

```
# Correlació entre l'edat i la variable Parch (pares/fills).

cor.test(x = titanic$Parch, y = titanic$Age, method = "spearman")
```

```
## Warning in cor.test.default(x = titanic$Parch, y = titanic$Age, method =
## "spearman"): Cannot compute exact p-value with ties
```

```
##
##  Spearman's rank correlation rho
##
## data:  titanic$Parch and titanic$Age
## S = 150261279, p-value < 2.2e-16
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##        rho
## -0.2745761
```

```
# Correlació el preu del bitllet i la variable Parch (pares/fills).

cor.test(x = titanic$Parch, y = titanic$Fare, method = "spearman")
```

```
## Warning in cor.test.default(x = titanic$Parch, y = titanic$Fare, method =
## "spearman"): Cannot compute exact p-value with ties
```

```
##
##  Spearman's rank correlation rho
##
## data:  titanic$Parch and titanic$Fare
## S = 69547095, p-value < 2.2e-16
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##       rho
## 0.4100738
```

```
# Correlació el preu del bitllet i la variable SibSp (pares/fills).

cor.test(x = titanic$SibSp, y = titanic$Fare, method = "spearman")
```

```
## Warning in cor.test.default(x = titanic$SibSp, y = titanic$Fare, method =
## "spearman"): Cannot compute exact p-value with ties
```

```
##
##  Spearman's rank correlation rho
##
## data:  titanic$SibSp and titanic$Fare
## S = 65180502, p-value < 2.2e-16
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##      rho
## 0.447113
```

En tots els casos el p-valor és significatiu. Entre l'edat i preu del bitllet hem obtingut 0,1, per tant, podríem dir que no hi ha correlació. L'edat i les 2 variables de famílies, hem obtingut 0,21 i 0,27 el qual no indicaria correlació. Per acabar, entre el bitllet i les 2 variables de famílies, hem obtingut una correlació positiva de

0,41 i 0,45, el qual podria indicar una correlació baixa.

Models classificadors A continuació es crearan diferents models per resoldre el problema de classificació. Provarem un model bayesià, un random forest i SVM (màquina de vectors de suport).

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

```
## Loading required package: caret
```

```
## Warning: package 'caret' was built under R version 4.1.2
```

```
## Loading required package: lattice
```

```
##
## Attaching package: 'caret'
```

```
## The following object is masked from 'package:purrr':
##
##     lift
```

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

```
## Loading required package: rminer
```

```
## Warning: package 'rminer' was built under R version 4.1.2
```

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

```
## Loading required package: naivebayes
```

```
## Warning: package 'naivebayes' was built under R version 4.1.2
```

```
## naivebayes 0.9.7 loaded
```

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

```
## Loading required package: LiblineaR
```

```
## Warning: package 'LiblineaR' was built under R version 4.1.2
```

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

```
## Loading required package: pROC
```

```
## Warning: package 'pROC' was built under R version 4.1.2
```

```
## Type 'citation("pROC")' for a citation.
```

```
##
## Attaching package: 'pROC'
```

```
## The following object is masked from 'package:colorspace':
##
##     coords
```

```
## The following objects are masked from 'package:stats':
##
##     cov, smooth, var
```

```
# Model Naive Bayes
set.seed(4)
h1<-holdout(titanic$Survived,ratio=7/10,mode="stratified")
titanic_train1<-titanic[h1$tr,]
titanic_test1<-titanic[h1$ts,]
```

```r
train_control1<- trainControl(method="cv", number=4)
modBayes<-train(Survived~., data=titanic_train1, method="naive_bayes", trControl = train_control1)
predBayes1 <- predict(modBayes, newdata=titanic_test1)
predBayes2 <- predict(modBayes, newdata=titanic_test1,type = "prob")
confusionMatrix(predBayes1,titanic_test1$Survived,positive="1")
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0 118  19
##          1  47  84
##
##                Accuracy : 0.7537
##                  95% CI : (0.6976, 0.8041)
##     No Information Rate : 0.6157
##     P-Value [Acc > NIR] : 1.176e-06
##
##                   Kappa : 0.5049
##
##  Mcnemar's Test P-Value : 0.000889
##
##             Sensitivity : 0.8155
##             Specificity : 0.7152
##          Pos Pred Value : 0.6412
##          Neg Pred Value : 0.8613
##              Prevalence : 0.3843
##          Detection Rate : 0.3134
##    Detection Prevalence : 0.4888
##       Balanced Accuracy : 0.7653
##
##        'Positive' Class : 1
##
```

```r
# Model Random Forest
set.seed(7)
h2<-holdout(titanic$Survived,ratio=7/10,mode="stratified")
titanic_train2<-titanic[h2$tr,]
titanic_test2<-titanic[h2$ts,]
train_control2<- trainControl(method="cv", number=4)
modForest<-train(Survived~., data=titanic_train2, method="rf", trControl = train_control2)
predForest1 <- predict(modForest, newdata=titanic_test2)
predForest2 <- predict(modForest, newdata=titanic_test2, type = "prob")
confusionMatrix(predForest1,titanic_test2$Survived,positive="1")
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0 148  23
##          1  17  80
##
##                Accuracy : 0.8507
##                  95% CI : (0.8024, 0.8912)
```

```
##     No Information Rate : 0.6157
##     P-Value [Acc > NIR] : <2e-16
##
##                   Kappa : 0.6811
##
##  Mcnemar's Test P-Value : 0.4292
##
##             Sensitivity : 0.7767
##             Specificity : 0.8970
##          Pos Pred Value : 0.8247
##          Neg Pred Value : 0.8655
##              Prevalence : 0.3843
##          Detection Rate : 0.2985
##    Detection Prevalence : 0.3619
##       Balanced Accuracy : 0.8368
##
##        'Positive' Class : 1
##
```

```
# Model Màquines de suport vectorial
set.seed(8)
h<-holdout(titanic$Survived,ratio=7/10,mode="stratified")
titanic_train<-titanic[h$tr,]
titanic_test<-titanic[h$ts,]
train_control<- trainControl(method="cv", number=4)
modSVM<-train(Survived~., data=titanic_train, method="svmLinearWeights2", trControl = train_control)
predSVM1 <- predict(modSVM, newdata=titanic_test)
confusionMatrix(predSVM1,titanic_test$Survived,positive="1")
```

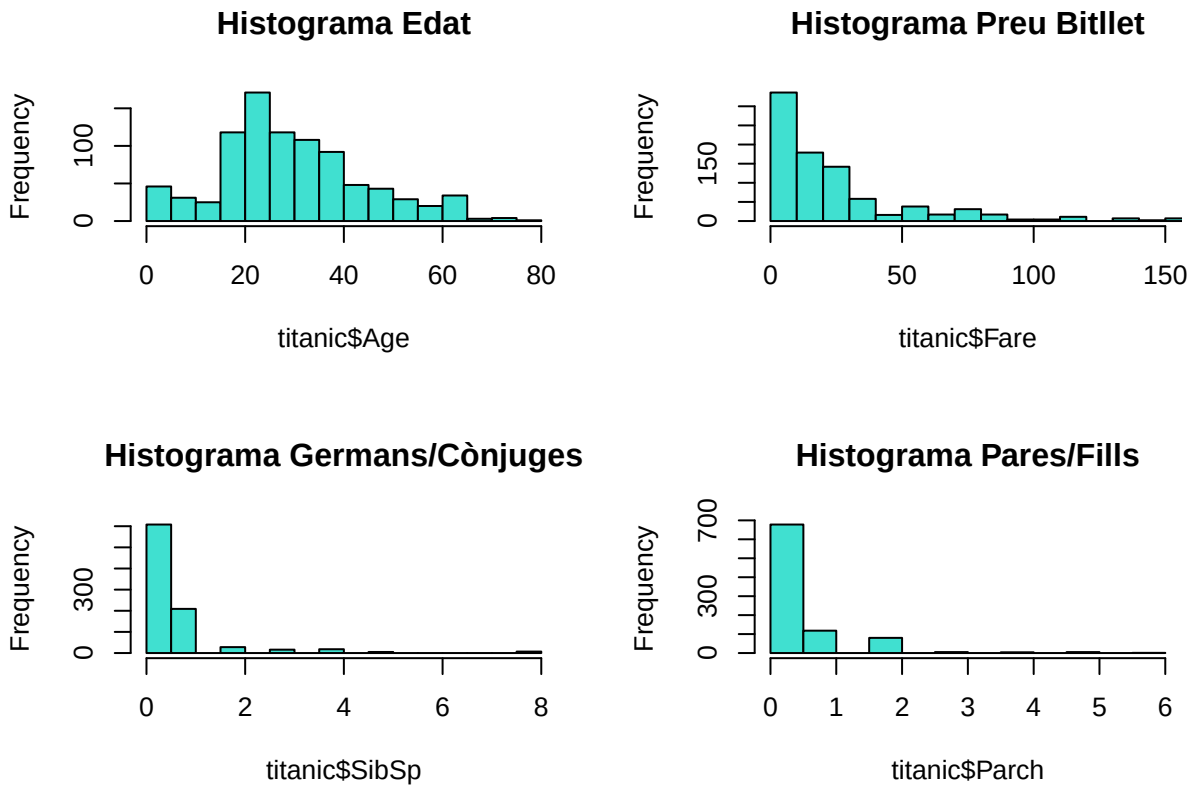
```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0 157  30
##          1   8  73
##
##                Accuracy : 0.8582
##                  95% CI : (0.8106, 0.8977)
##     No Information Rate : 0.6157
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.6879
##
##  Mcnemar's Test P-Value : 0.0006577
##
##             Sensitivity : 0.7087
##             Specificity : 0.9515
##          Pos Pred Value : 0.9012
##          Neg Pred Value : 0.8396
##              Prevalence : 0.3843
##          Detection Rate : 0.2724
##    Detection Prevalence : 0.3022
##       Balanced Accuracy : 0.8301
##
##        'Positive' Class : 1
```

##

Representació dels resultats a partir de taules i gràfiques.

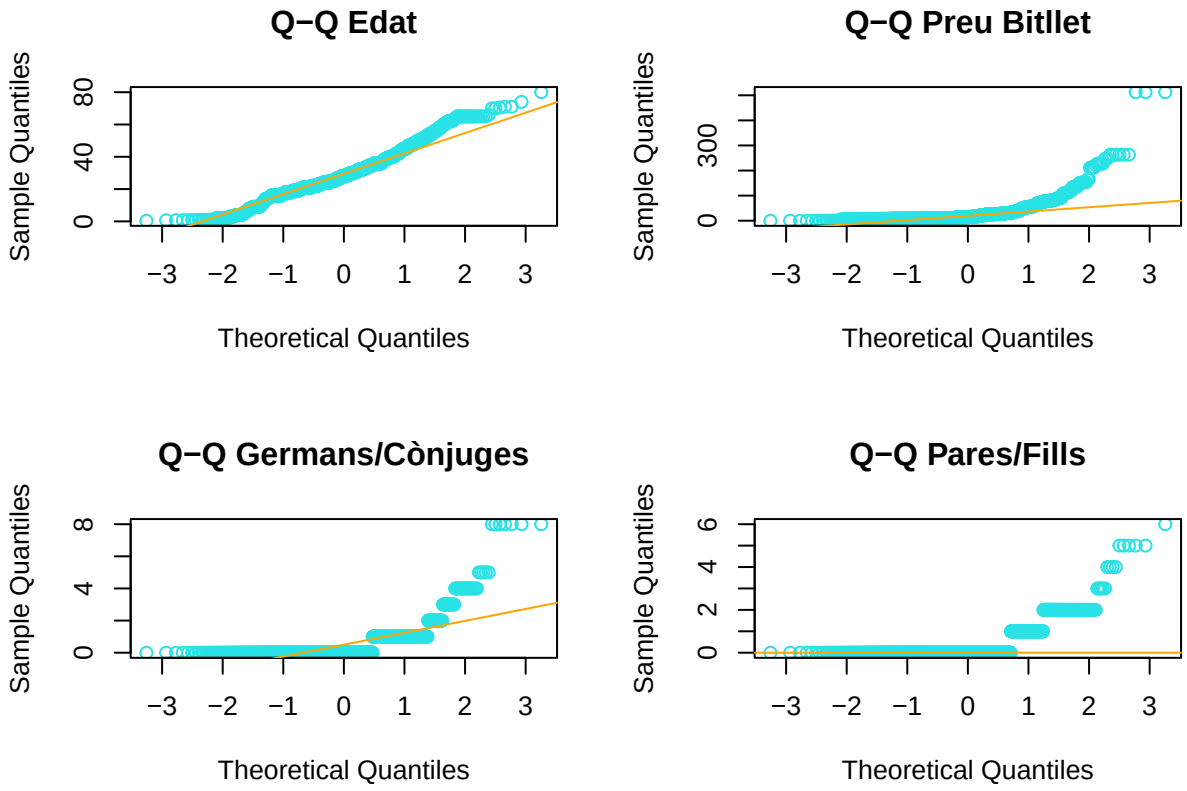
A continuació es mostraran els histogrames i els gràfics Q-Q (quantils teòrics), per comprovar la normalitat gràficament.

```
par(mfrow=c(2,2))
# Histograma Age
hist(titanic$Age, col = "turquoise", breaks = 20, main = "Histograma Edat")
# Histograma Fare
hist(titanic$Fare, col = "turquoise", breaks= 50, xlim = c(0,150), main = "Histograma Preu Bitllet")
# Histograma SibSp
hist(titanic$SibSp, col = "turquoise", breaks = 15, main = "Histograma Germans/Cònjuges")
# Histograma Parch
hist(titanic$Parch, col = "turquoise", breaks = 15, main = "Histograma Pares/Fills")
```



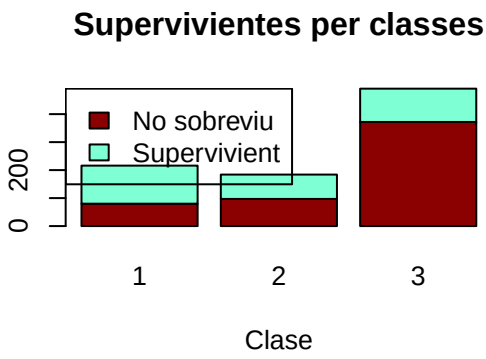

```
# Gràfic Q-Q Age
qqnorm(titanic$Age, main="Q-Q Edat", col = 5)
qqline(titanic$Age,col="orange")
# Gràfic Q-Q Fare
qqnorm(titanic$Fare, main="Q-Q Preu Bitllet", col = 5)
qqline(titanic$Fare,col="orange")
# Gràfic Q-Q SibSp
qqnorm(titanic$SibSp, main="Q-Q Germans/Cònjuges", col = 5)
qqline(titanic$SibSp,col="orange")
# Gràfic Q-Q Parch
```

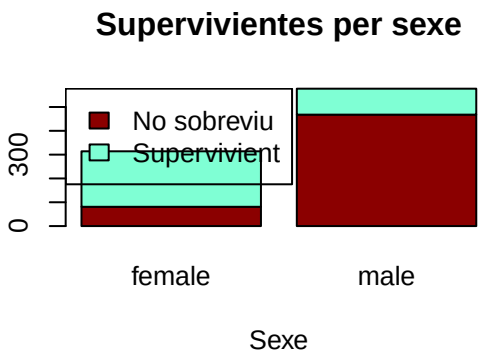
```
qqnorm(titanic$Parch, main="Q-Q Pares/Fills", col = 5)
qqline(titanic$Parch,col="orange")
```

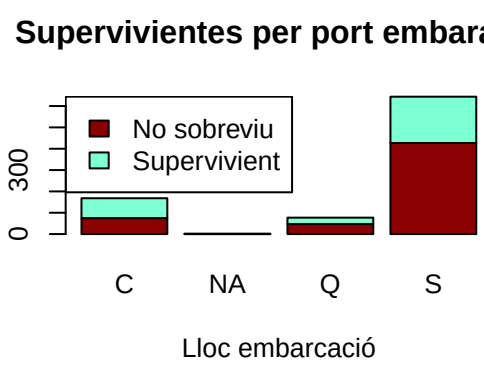


També farem els gràfics de la resta de variables, utilitzant gràfics de barres. Cada variable es compara amb la variable objectiu de si sobreviu o no.

```
par(mfrow=c(2,2))
#Classes
barplot(table(titanic[,c(1,2)]), main = "Supervivientes per classes", xlab = "Clase",
        col = c("red4","aquamarine"))
legend("topleft", c("No sobreviu","Supervivient"), fill = c("red4","aquamarine"))
#Sexe
barplot(table(titanic[,c(1,3)]), main = "Supervivientes per sexe", xlab = "Sexe",
        col = c("red4","aquamarine"))
legend("topleft", c("No sobreviu","Supervivient"), fill = c("red4","aquamarine"))
#Embarked
barplot(table(titanic[,c(1,8)]), main = "Supervivientes per port embaració",
        xlab = "Lloc embarcació",col = c("red4","aquamarine"))
legend("topleft", c("No sobreviu","Supervivient"), fill = c("red4","aquamarine"))
```







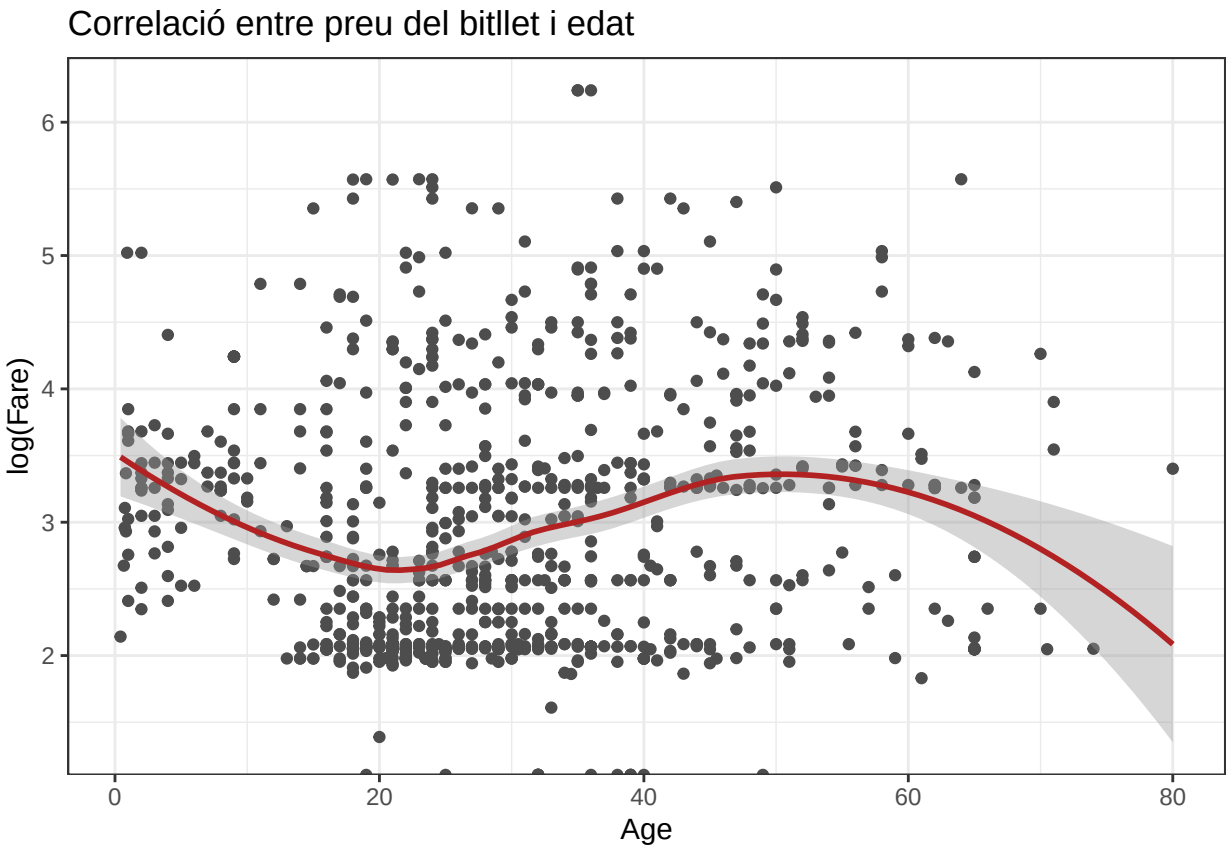
Correlacions

A continuació farem el gràfic de correlació entre el preu del bitllet i l'edat del passatger.

```
ggplot(data = titanic, aes(x = Age, y = log(Fare))) + geom_point(color = "gray30") +
  geom_smooth(color = "firebrick") + theme_bw() +ggtitle("Correlació entre preu del bitllet i edat")
```

```
## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```

```
## Warning: Removed 15 rows containing non-finite values (stat_smooth).
```



Podem veure que no hi ha correlació entre el preu del bitllet i l'edat.

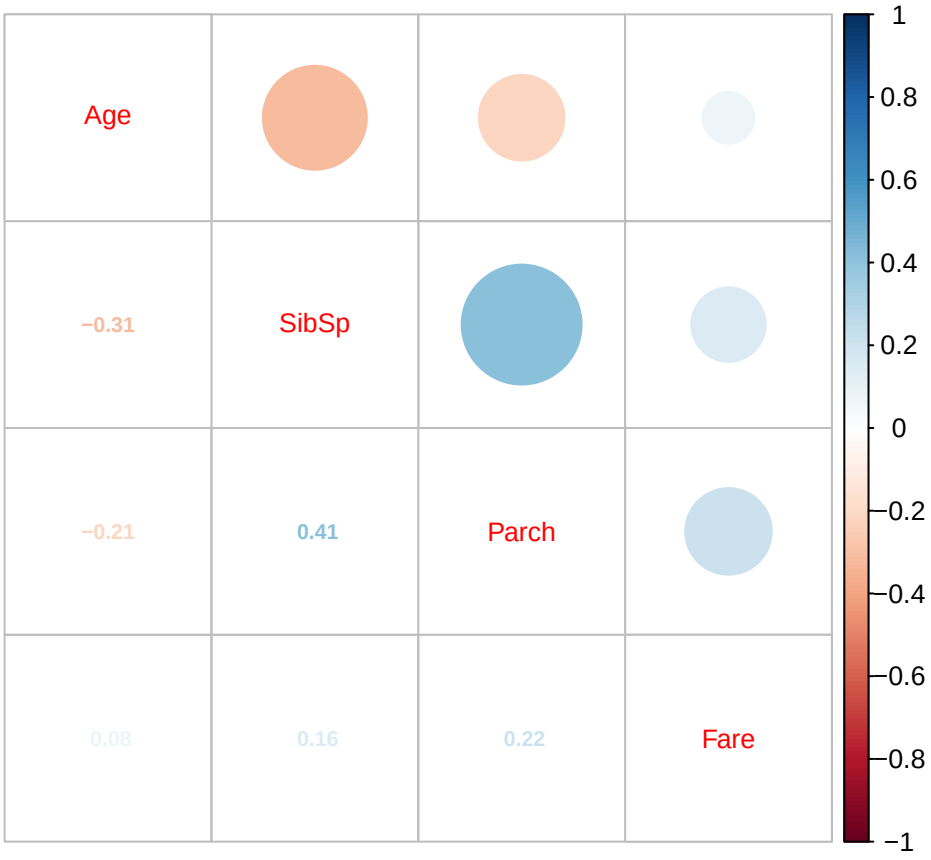
A continuació mostrarem les correlacions de les variables numèriques amb gràfics.

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

```
## Loading required package: corrplot
```

```
## corrplot 0.90 loaded
```

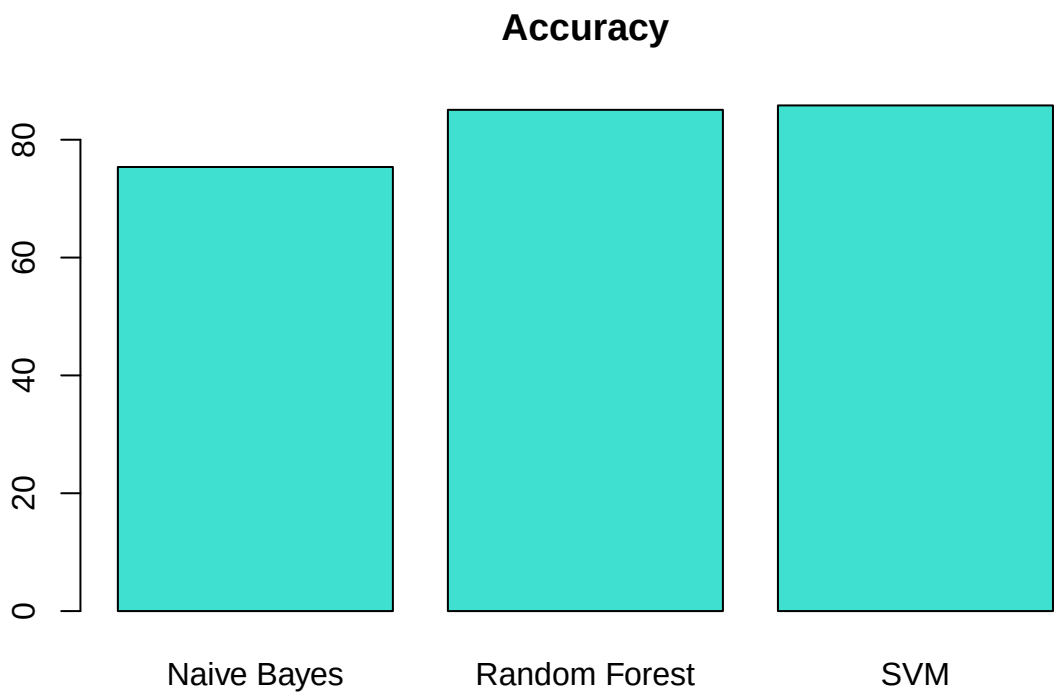
```
corr.res<-cor(titanic[c('Age','SibSp','Parch','Fare')])
corrplot.mixed(corr.res,upper="circle",number.cex=.7,tl.cex=.8)
```



Accuracy Models

A continuació mostrarem les accuracy dels models en gràfic de barres.

```
par(mfrow=c(1,1))
bayes<- c(75.37)
random<-c(85.07)
SVM<-c(85.82)
df <- data.frame(bayes,random,SVM)
names(df) <- c('Naive Bayes','Random Forest','SVM')
plot<-barplot(as.matrix(df), col = 'turquoise', main = "Accuracy")
```



El millor model que hem obtingut ha estat el de SVM, amb un accuracy del 85,82%.

Curves ROC

A continuació mostrarem les corbes ROC del model bayesià i Random Forest.

```
par(mfrow=c(2,2))
# Corba ROC Bayesià:
plot.roc(titanic_test1$Survived , predBayes2$"1", )
```

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

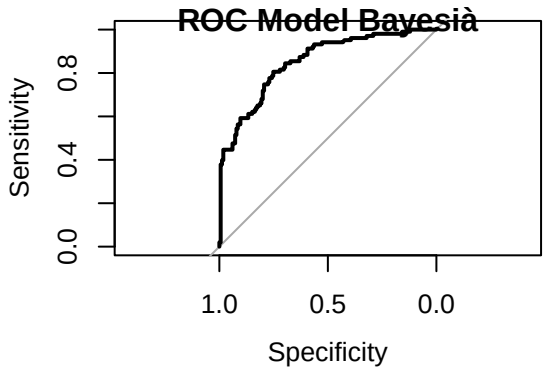
```
## Setting direction: controls < cases
```

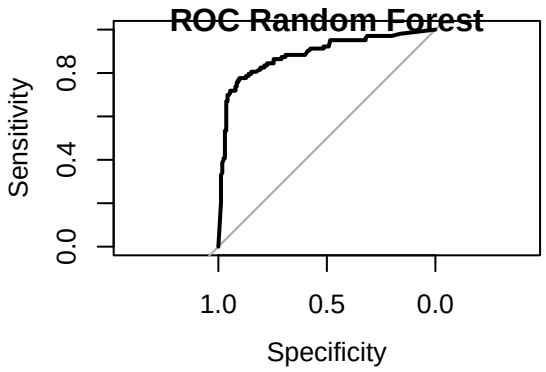
```
title("ROC Model Bayesià")

# Corba ROC Random Forest:
plot.roc(titanic_test2$Survived , predForest2$"1")
```

```
## Setting levels: control = 0, case = 1
## Setting direction: controls < cases
```

```
title("ROC Random Forest")
```





Resolució del problema. A partir dels resultats obtinguts, quines són les conclusions? Els resultats permeten respondre al problema?

Primer hem fet una anàlisi de les dades, imputació, selecció de les variables més importants i detecció d'outliers.

Un cop teníem les dades "netes", hem pogut comprovar que les dones teníem més possibilitats de sobreviure que els homes. També si eres de primera classe. El preu del bitllet no estava relacionat amb l'edat.

Finalment, hem creat 3 models que amb les característiques del passatger pot dir si sobreviurà o no. Destacar que amb menys de 1000 observacions, hem pogut obtenir nivells de "accuracy" elevats, el qual el millor ha estat el de SVM (màquina de vectors de suport).

Dataset Final

```
write.csv(titanic,"../Dataset/Titanic_Final.csv", row.names = F)
```

Per últim la taula de contribucions del treball.

Contribucions	Firma
Investigació prèvia	DPE, SCB
Redacció de les respostes	DPE, SCB
Desenvolupament codi	DPE, SCB