PRACTICA 2 - NETEJA I VALIDACIO DE DADES
POL CASELLAS i CARLES RIVAS

---

1. DescripciÃƒÂ³ del dataset. PerquÃƒÂ¨ ÃƒÂ©s important, quina pregunta/problema pretÃƒÂ©n respondre?

El dataset original ÃƒÂ©s el dataset de dades dels passatgers del TitÃƒÂ nic, que sÃ¢â‚¬â„¢utilitza en la competiciÃƒÂ³ Ã¢â‚¬Å“Titanic: Machine Learning from DisasterÃ¢â‚¬Â. Esta composat per dos fitxers, un de training (train.csv) amb 891 registres i un de test (test.csv) amb 418 registres. LÃ¢â‚¬â„¢objectiu del dataset de la competiciÃƒÂ³ es generar un model de ML per predir la supervivÃƒÂ¨ncia a lÃ¢â‚¬â„¢enfonsament del vaixell, en base a la informaciÃƒÂ³ disponible dels passatgers.

El diccionari de dades del dataset original ÃƒÂ©s :

Variable	DescripciÃƒÂ³	Tipus	Valors
PassangerId	Identificador de passatger	Enter	
Survival	Identificador de supervivÃƒÂ¨ncia	Factor	0 = No, 1 = Yes
Pclass	Classe del passatge	Factor	1 = 1a, 2 = 2ona, 3 = 3Ã‚Âª
Name	Nom del passatger	Text	
Sex	Sexe	Factor	1 = male, 2 = female
Age	Edat en anys	Nombre	
Sibsp	Nombre de germans i dones embarcats	Enter	
Parch	Nombre de pares i fills embarcats	Enter	
Ticket	Nombre de Ticket	Text	
Fare	Tarifa del passatge	Nombre	
Cabin	Nombre de cabina	Text	
Embarked	Port dÃ¢â‚¬â„¢Embarcament	Factor	C = Cherbourg Q = Queenstown S = Southampton

2. IntegraciÃƒÂ³ i selecciÃƒÂ³ de les dades dÃ¢â‚¬â„¢interÃƒÂ s a analitzar.

Es disposa de 2 fitxers, un fitxer dÃ¢â‚¬â„¢entrenament (que disposa de les dades de supervivÃƒÂ¨ncia) i un fitxer de test (sense les dades de supervivÃƒÂ¨ncia). Es llegeixen les dades i sÃ¢â‚¬â„¢integren en un ÃƒÂºnic dataset.

```
library('dplyr')
dades <- bind_rows(read.csv('train.csv', stringsAsFactors = F), read.csv('test.csv', stringsAsFacto
```

Es crea un dataset que integra els dos fitxers anomenat dades. La comanda dim(dades) mostra el nombre dÃ¢â‚¬â„¢observacions i atributs de la mostra, que en aquest cas sÃƒÂ³n 1309 observacions i 12 atributs. El camp **Survived** nomÃƒÂ©s estÃƒÂ informat per les entrades del fitxer dÃ¢â‚¬â„¢entrenament (fins la posiciÃƒÂ³ 891). En la resta dÃ¢â‚¬â„¢entrades, el valor en N.A.

```
dim (dades)
```

```
## [1] 1309   12
```

La comanda str(dades) presenta de forma compacta lÃ¢â‚¬â„¢estructura interna del dataset.

```
str(dades)
```

```
## 'data.frame':    1309 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 Thay
##  $ 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" ...
```

Amb la comanda

```
factors<-c('Sex','Embarked')
dades[factors] <- lapply(dades[factors], function(x) as.factor(x))
```

sÃ¢â‚¬â„¢indica al dataset que els camps **Pclass**, **Sex** i **Embarked** sÃƒÂ³n factors. Al aplicar de nou la funciÃƒÂ³ str (dades) es pot observar que ara es consideren factors i es mostren els possibles valors que admeten.

```
str(dades)
```

```
## 'data.frame':    1309 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 Thay
##  $ Sex        : Factor w/ 2 levels "female","male": 2 1 1 1 2 2 2 2 1 1 ...
##  $ 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   : Factor w/ 4 levels "","C","Q","S": 4 2 4 4 4 3 4 4 4 2 ...
```

La comanda summary(dades) mostra una visiÃƒÂ³ general dels atributs i en el cas de factors, es presenten els valors possibles i el nombre dÃ¢â‚¬â„¢ocurrÃƒÂ¨ncies per cada valor. A mÃƒÂ©s, es pot observar que hi ha diversos valors mal informats o buits.

```
summary (dades)
```

```
##   PassengerId      Survived          Pclass          Name
##  Min.   :   1   Min.   :0.0000   Min.   :1.000   Length:1309
##  1st Qu.: 328   1st Qu.:0.0000   1st Qu.:2.000   Class :character
##  Median : 655   Median :0.0000   Median :3.000   Mode  :character
##  Mean   : 655   Mean   :0.3838   Mean   :2.295
##  3rd Qu.: 982   3rd Qu.:1.0000   3rd Qu.:3.000
##  Max.   :1309   Max.   :1.0000   Max.   :3.000
##                 NA's   :418
##      Sex            Age            SibSp            Parch
##  female:466   Min.   : 0.17   Min.   :0.0000   Min.   :0.000
##  male  :843   1st Qu.:21.00   1st Qu.:0.0000   1st Qu.:0.000
##               Median :28.00   Median :0.0000   Median :0.000
##               Mean   :29.88   Mean   :0.4989   Mean   :0.385
##               3rd Qu.:39.00   3rd Qu.:1.0000   3rd Qu.:0.000
##               Max.   :80.00   Max.   :8.0000   Max.   :9.000
##               NA's   :263
##     Ticket               Fare            Cabin           Embarked
##  Length:1309        Min.   :  0.000   Length:1309         :  2
##  Class :character   1st Qu.:  7.896   Class :character   C:270
##  Mode  :character   Median : 14.454   Mode  :character   Q:123
##                     Mean   : 33.295                      S:914
##                     3rd Qu.: 31.275
##                     Max.   :512.329
##                     NA's   :1
```

3. Neteja de les dades.

3.1 Les dades contenen zeros o elements buits? Com gestionaries aquests casos?

Es pot observar que hi ha 418 valors no informats de Survived que corresponen als valors del fitxer de test, 263 registres no presents a **Age**, 1 registre que no informa el **Fare** i 2 registres que no indiquen el camp Embarked. Addicionalment hi ha diversos registres de **Fare** amb valor 0. Es procedeix a aplicar diverses tÃƒÂ¨cniques per completar aquestes dades.

- Per intentar determinar el valor de la tarifa (**Fare**) que manca, el primer que es fa ÃƒÂ©s identificar el registre mal informat

```
IdNoFare <- dades$PassengerId[is.na(dades$Fare)]
dades[IdNoFare,]
```

```
##      PassengerId Survived Pclass               Name  Sex  Age SibSp Parch
## 1044        1044       NA      3 Storey, Mr. Thomas male 60.5     0     0
##      Ticket Fare Cabin Embarked
## 1044   3701   NA              S
```

SÃ¢â‚¬â„¢observa que es tracta dÃ¢â‚¬â„¢un passatger de classe 3 i que va embarcar a Ã¢â‚¬Ëœ SÃ¢â‚¬â„¢. Es calcula el preu mitja de la tarifa dÃ¢â‚¬â„¢aquest tipus de clients i sÃ¢â‚¬â„¢assigna a la tarifa del passatger no informat. El cÃƒÂ lcul de la mitjana no tÃƒÂ© en compte els valors NA (parÃƒÂ metre na.rm = TRUE)

```
dades$Fare[is.na(dades$Fare)] <- median(dades[dades$Pclass == '3' & dades$Embarked == 'S', ]$Fare,
dades[IdNoFare,]
```

```
##      PassengerId Survived Pclass               Name  Sex  Age SibSp Parch
## 1044        1044       NA      3 Storey, Mr. Thomas male 60.5     0     0
##      Ticket Fare Cabin Embarked
## 1044   3701 8.05              S
```

El valor assignat ÃƒÂ©s 8,05 $

- Per intentar determinar el valor de lÃ¢â‚¬â„¢Embarked que manca, el primer que es fa ÃƒÂ©s identificar els registres mal informats

```
IdNoEmbarked <- dades$PassengerId[dades$Embarked == '']
dades[IdNoEmbarked,]
```

```
##     PassengerId Survived Pclass                                      Name
## 62           62        1      1                       Icard, Miss. Amelie
## 830         830        1      1 Stone, Mrs. George Nelson (Martha Evelyn)
##        Sex Age SibSp Parch Ticket Fare Cabin Embarked
## 62  female  38     0     0 113572   80   B28
## 830 female  62     0     0 113572   80   B28
```

SÃ¢â‚¬â„¢observa que es tracta de passatgers de classe 1 amb una tarifa de 80 $. Per determinar el port dÃ¢â‚¬â„¢embarcament es calcula la tarifa mitjana per a la classe 1, per cada tipus dÃ¢â‚¬â„¢embarcament possible.

```
median(dades[dades$Pclass == '1' & dades$Embarked == 'S', ]$Fare, na.rm = TRUE)
```

```
## [1] 52
```

```
median(dades[dades$Pclass == '1' & dades$Embarked == 'Q', ]$Fare, na.rm = TRUE)
```

```
## [1] 90
```

```
median(dades[dades$Pclass == '1' & dades$Embarked == 'C', ]$Fare, na.rm = TRUE)
```

```
## [1] 76.7292
```

SÃ¢â‚¬â„¢observa que la tarifa mitjana mÃƒÂ©s propera a la dels registres no informats ÃƒÂ©s 76,73 $ corresponent a lÃ¢â‚¬â„¢embarcament Ã¢â‚¬Ë œCÃ¢â‚¬â„¢. SÃ¢â‚¬â„¢assigna aquest valor als registres mal informats.

```
dades$Embarked[c(IdNoEmbarked)] <- 'C'
```

- Donat que el nombre de valors **Age** no informats ÃƒÂ©s elevat, la opciÃƒÂ³ dÃ¢â‚¬â„¢esborrar aquests valors suposaria la pÃƒÂ¨rdua de molts registres que si contenen altres valors i que ÃƒÂ©s interessant conservar. Per aquest motiu, sÃ¢â‚¬â„¢utilitzarÃƒÂ la llibreria mice (Multivariate Imputations by Chained Equations) que implementa un mÃƒÂ¨tode per tractar valors no informats, creant mÃƒÂºltiples imputacions per dades no informades multivariable. Cada variable incompleta es pot imputar per un model separat, permetent la imputaciÃƒÂ³ de barreges de dades de diversos tipus (categÃƒÂ²riques, continues, binaries, …) i mantenir la coherÃƒÂ¨ncia entre les imputacions mitjanÃƒÂ§ant la implantaciÃƒÂ³ passiva.

SÃ¢â‚¬â„¢aplica la funciÃƒÂ³ mice a un subconjunt dels camps del dataset, no es contemplen els camps **PassengerId**, **Name**, **Ticket**, **Cabin**, **Survived** perquÃƒÂ¨ no aporten res al model i es simplifiquen els cÃƒÂ lculs

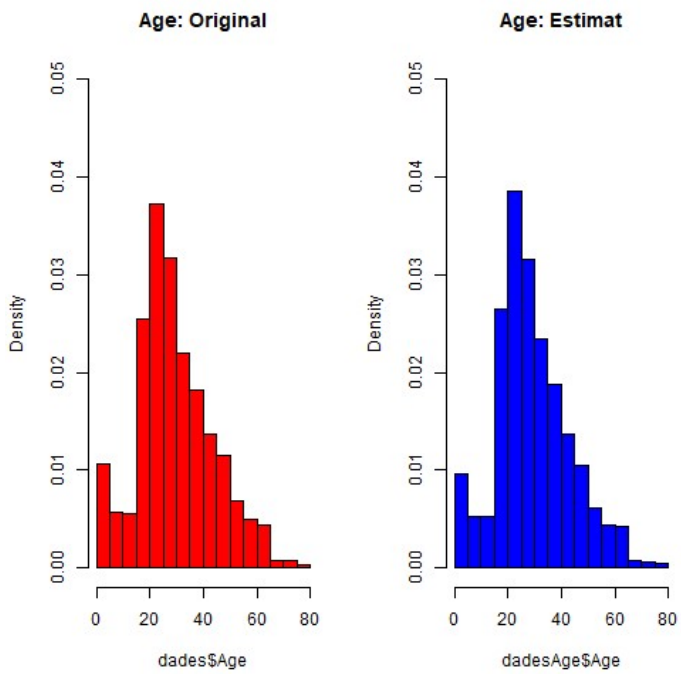
```
library('mice')
library('randomForest')
set.seed(1000)
dadesAge <- complete(mice(dades[, !names(dades) %in% c('PassengerId','Name', 'Ticket', 'Cabin', 'Su
```

```
##
##  iter imp variable
##   1   1  Age
##   1   2  Age
##   1   3  Age
##   1   4  Age
##   1   5  Age
##   2   1  Age
##   2   2  Age
##   2   3  Age
##   2   4  Age
##   2   5  Age
##   3   1  Age
##   3   2  Age
##   3   3  Age
##   3   4  Age
##   3   5  Age
##   4   1  Age
##   4   2  Age
##   4   3  Age
##   4   4  Age
##   4   5  Age
##   5   1  Age
##   5   2  Age
##   5   3  Age
##   5   4  Age
##   5   5  Age
```

```
## Warning: Number of logged events: 25
```

MitjanÃƒÂ§ant la comparaciÃƒÂ³ visual de plots sobre lÃ¢â‚¬â„¢**Age** original i el estimat, es pot verificar que la assignaciÃƒÂ³ de les edats no informades no introdueix biaixos en les dades.

```
par(mfrow=c(1,2))
hist(dades$Age, freq=F, main='Age: Original', col='red', ylim=c(0,0.05))
hist(dadesAge$Age, freq=F, main='Age: Estimat', col='blue', ylim=c(0,0.05))
```




```
dades$Age <- dadesAge$Age
summary (dades)
```

```
##   PassengerId      Survived          Pclass          Name
##  Min.   :   1   Min.   :0.0000   Min.   :1.000   Length:1309
##  1st Qu.: 328   1st Qu.:0.0000   1st Qu.:2.000   Class :character
##  Median : 655   Median :0.0000   Median :3.000   Mode  :character
##  Mean   : 655   Mean   :0.3838   Mean   :2.295
##  3rd Qu.: 982   3rd Qu.:1.0000   3rd Qu.:3.000
##  Max.   :1309   Max.   :1.0000   Max.   :3.000
##                 NA's   :418
##      Sex            Age            SibSp            Parch
##  female:466   Min.   : 0.17   Min.   :0.0000   Min.   :0.000
##  male  :843   1st Qu.:21.00   1st Qu.:0.0000   1st Qu.:0.000
##               Median :28.00   Median :0.0000   Median :0.000
##               Mean   :29.74   Mean   :0.4989   Mean   :0.385
##               3rd Qu.:38.00   3rd Qu.:1.0000   3rd Qu.:0.000
##               Max.   :80.00   Max.   :8.0000   Max.   :9.000
##
##     Ticket               Fare            Cabin           Embarked
##  Length:1309        Min.   :  0.000   Length:1309         :  0
##  Class :character   1st Qu.:  7.896   Class :character   C:272
##  Mode  :character   Median : 14.454   Mode  :character   Q:123
##                     Mean   : 33.276                      S:914
##                     3rd Qu.: 31.275
##                     Max.   :512.329
##
```

- En el camp **Fare** sÃ¢â‚¬â„¢observen diversos valors a 0 que es considera un valor fals, que correspon a valors no informats. Per estimar-los sÃ¢â‚¬â„¢utilitza lÃ¢â‚¬â„¢algoritme kNN. En primer lloc es substitueix els valors 0 per NA, ja que lÃ¢â‚¬â„¢algoritme nomÃƒÂ©s estima els valors NA del dataset. A continuaciÃƒÂ³ sÃ¢â‚¬â„¢aplica lÃ¢â‚¬â„¢algoritme i sÃ¢â‚¬â„¢obtenen valors pels valors no informats de **Fare**.

```
# Els valors 0 de Fare es converteixen en NA i es dedueixen amb l'algoritme KNN
dades$Fare[dades$Fare == 0] <- NA

# Es Comprova si hi ha NA,s a Fare
sapply(dades, function(x) sum(is.na(x)))
```

```
## PassengerId    Survived      Pclass        Name         Sex         Age
##           0         418           0           0           0           0
##       SibSp       Parch      Ticket        Fare       Cabin    Embarked
##           0           0           0          17           0           0
```

SÃ¢â‚¬â„¢observa que al **Fare** hi ha 17 registres no informats.

```
library(VIM)

# S'aplica l'algoritme kNN
dades.kNN <- kNN(dades[, !names(dades) %in% c('PassengerId','Name', 'Ticket', 'Cabin', 'Survived')]

# Es comprova si hi ha NA,s a Fare estimat
sapply(dades.kNN, function(x) sum(is.na(x)))
```

```
##       Pclass          Sex          Age        SibSp        Parch
##            0            0            0            0            0
##         Fare     Embarked   Pclass_imp      Sex_imp      Age_imp
##            0            0            0            0            0
##    SibSp_imp    Parch_imp     Fare_imp Embarked_imp
##            0            0            0            0
```

```
# Es comprova si hi ha zeros a Fare estimat
length(dades.kNN$Fare[dades.kNN$Fare == 0])
```

```
## [1] 0
```

```
# S'assignen els valors de Fare estimats amb kNN al dataset original
dades$Fare = dades.kNN$Fare
```

- Per millorar el dataset es planteja la creaciÃƒÂ³ de nous camps a partir de les dades disponibles :

En primer lloc es crea el camp **TamanyFamilia** format a partir de la suma dels camps **SibSp** i **Parch** (cÃƒÂ²njuges, germans, pares i fills) mÃƒÂ©s el propi passatger.

```
# CreaciÃƒÂ³ del camp TamanyFamilia
dades$TamanyFamilia <- dades$SibSp + dades$Parch + 1
```

A partir dÃ¢â‚¬â„¢aquest camp es crea el camp **TipusFamilia**, de tipus factor, format a partir de rangs de nombre de membres de la famÃƒÂlia embarcats.

```
# CreaciÃƒÂ³ del camp TipusFamilia
dades$TipusFamilia[dades$TamanyFamilia == 1] <- 'solitari'
dades$TipusFamilia[dades$TamanyFamilia < 5 & dades$TamanyFamilia > 1] <- 'petita'
dades$TipusFamilia[dades$TamanyFamilia > 4] <- 'nombrosa'

# ConversiÃƒÂ³ a factor
dades$TipusFamilia <- as.factor(dades$TipusFamilia)
```

TambÃƒÂ© a partir del camp Age es crea el camp **TipusEdat**, de tipus factor, format a partir de rangs dÃ¢â‚¬â„¢edats.

```
# Es crea el camp TipusEdat en funciÃƒÂ³ de franges d'edat i es converteix en factor

dades$TipusEdat[dades$Age < 18] <- 'Menor'
dades$TipusEdat[dades$Age >= 18 & dades$Age <= 65] <- 'Adult'
dades$TipusEdat[dades$Age > 65] <- 'AnciÃƒÂ '
dades$TipusEdat <- as.factor(dades$TipusEdat)
```

En funciÃ³ del tipus de metodologies i algoritmes a utilitzar es pot prioritzar l'Ãºs d'atributs numÃ¨rics com en el cas d'**Age** o **TamanyFamilia**, o be, atributs de tipus factor com sÃ³n **TipusEdat** o **TipusFamilia**, calculats a partir dels camps numÃ¨rics. TambÃ© s'ha descartat l'Ãºs del camp **Ticket** per que presenta una estructura bastant variable, no segueix un patrÃ³ i correspon en gran part a codis de ticket autonumÃ¨rics que s'han considerats que aporten poca informaciÃ³ per a la questiÃ³ plantejada. El camp **Cabin** tambÃ© s'ha descartat perquÃ©, si be era un camp que potencialment podria aportar informaciÃ³ rellevant respecte a la supervivÃ¨ncia en funciÃ³ de les cabines, per exemple, degut a la seva ubicaciÃ³, el nombre de valors no informats Ã©s molt elevat (mÃ©s de 4/5 parts) el que fa poc recomanble el completar els valors no informats mitjanÃ§ant les tÃ¨cniques aplicades en altres casos. Finalment s'ha utilitzar el camp TamanyFamilia, enlloc dels camps SibSp i Parch, ja que n'Ã©s l'agrupaciÃ³ d'ambos, reduint la dimensionalitat del dataset. S'han mantinut els camps **PassengerId** i **Name** per claredat ja que no aporten valor a la identificaciÃ³ de la supervivÃ¨ncia.

Per tant el dataset final, integrat i filtrat ÃƒÂ©s el segÃƒÂ¼ent :

```
#dataset final integrat i filtrat
dades = dades[, !names(dades) %in% c('SibSp','Parch', 'Ticket', 'Cabin', 'TipusFamilia', "TipusEdat
summary (dades)
```

```
##   PassengerId      Survived          Pclass          Name
##  Min.   :   1   Min.   :0.0000   Min.   :1.000   Length:1309
##  1st Qu.: 328   1st Qu.:0.0000   1st Qu.:2.000   Class :character
##  Median : 655   Median :0.0000   Median :3.000   Mode  :character
##  Mean   : 655   Mean   :0.3838   Mean   :2.295
##  3rd Qu.: 982   3rd Qu.:1.0000   3rd Qu.:3.000
##  Max.   :1309   Max.   :1.0000   Max.   :3.000
##                 NA's   :418
##      Sex           Age             Fare         Embarked TamanyFamilia
##  female:466   Min.   : 0.17   Min.   :  3.171    :  0    Min.   : 1.000
##  male  :843   1st Qu.:21.00   1st Qu.:  7.925   C:272    1st Qu.: 1.000
##               Median :28.00   Median : 14.500   Q:123    Median : 1.000
##               Mean   :29.74   Mean   : 33.644   S:914    Mean   : 1.884
##               3rd Qu.:38.00   3rd Qu.: 31.387            3rd Qu.: 2.000
##               Max.   :80.00   Max.   :512.329            Max.   :11.000
##
```

El diccionari de dades del dataset final, integrat i filtrat ÃƒÂ©s el segÃƒÂ¼ent :

Variable	DescripciÃƒÂ³	Tipus	Valors
PassangerId	Identificador de passatger	Enter	
Survival	Identificador de supervivÃƒÂ¨ncia	Factor	0 = No, 1 = Yes
Pclass	Classe del passatge	Factor	1 = 1a, 2 = 2ona, 3 = 3Ã‚Âª
Name	Nom del passatger	Text	
Sex	Sexe	Factor	1 = male, 2 = female
Age	Edat en anys	Nombre	
Fare	Tarifa del passatge	Nombre	
Embarked	Port dÃ¢â‚¬â„¢Embarcament	Factor	C = Cherbourg Q = Queenstown S = Southampton
TamanyFamilia	Nombre de familiars embarcats	Enter	

3.2. IdentificaciÃƒÂ³ i tractament de valors extrems.

Identifiquem els valors extrems de les variables **Age** i **Fare** a partir de la funciÃƒÂ³ boxplot:

```
outliers_age <- boxplot.stats(dades$Age)$out
outliers_fare <- boxplot.stats(dades$Fare)$out
outliers_age
```

```
##  [1] 66.0 65.0 65.0 71.0 70.5 70.0 65.0 64.0 65.0 71.0 64.0 80.0 70.0 70.0
## [15] 74.0 67.0 76.0 64.0 64.0 80.0 64.0
```

```
outliers_fare
```

```
##   [1]  71.2833 263.0000 146.5208  82.1708  76.7292  80.0000  83.4750
##   [8]  73.5000 263.0000  77.2875 247.5208  73.5000  77.2875  79.2000
##  [15]  66.6000  69.5500  69.5500 146.5208  69.5500 113.2750  76.2917
##  [22]  90.0000  83.4750  90.0000  79.2000  86.5000 512.3292  79.6500
##  [29] 153.4625 135.6333  77.9583  78.8500  91.0792 151.5500 247.5208
##  [36] 151.5500 110.8833 108.9000  83.1583 262.3750 164.8667 134.5000
##  [43]  69.5500 135.6333 153.4625 133.6500  66.6000 134.5000 263.0000
##  [50]  75.2500  69.3000 135.6333  82.1708 211.5000 227.5250  73.5000
##  [57] 120.0000 113.2750  90.0000 120.0000 263.0000  81.8583  89.1042
##  [64]  91.0792  90.0000  78.2667 151.5500  86.5000 108.9000  93.5000
##  [71] 221.7792 106.4250  71.0000 106.4250 110.8833 227.5250  79.6500
##  [78] 110.8833  79.6500  79.2000  78.2667 153.4625  77.9583  69.3000
##  [85]  76.7292  73.5000 113.2750 133.6500  73.5000 512.3292  76.7292
##  [92] 211.3375 110.8833 227.5250 151.5500 227.5250 211.3375 512.3292
##  [99]  78.8500 262.3750  71.0000  86.5000 120.0000  77.9583 211.3375
## [106]  79.2000  69.5500 120.0000  93.5000  90.0000  80.0000  83.1583
## [113]  69.5500  89.1042 164.8667  69.5500  83.1583  82.2667 262.3750
## [120]  76.2917 263.0000 262.3750 262.3750 263.0000 211.5000 211.5000
## [127] 221.7792  78.8500 221.7792  75.2417 151.5500 262.3750  83.1583
## [134] 221.7792  83.1583  83.1583 247.5208  69.5500 134.5000 227.5250
## [141]  73.5000 164.8667 211.5000  71.2833  75.2500 106.4250 134.5000
## [148] 136.7792  75.2417 136.7792  82.2667  81.8583 151.5500  93.5000
## [155] 135.6333 146.5208 211.3375  79.2000  69.5500 512.3292  73.5000
## [162]  69.5500  69.5500 134.5000  81.8583 262.3750  93.5000  79.2000
## [169] 164.8667 211.5000  90.0000 108.9000
```

Si observem els valors extrems identificats de les variables **Age** i **Fare**, sÃƒÂ³n valors coherents i que no tenen perquÃƒÂ© suposar un error en la mostra de dades. No obstant, si volguesim eliminar els valors ho fariem mitjanÃƒÂ§ant el codi segÃƒÂ¼ent:

```
#dades<-dades[-which(dades$Age %in% outliers_age),]
#dades<-dades[-which(dades$Fare %in% outliers_fare),]
```

#4. AnÃƒÂ lisi de les dades.

##4.1. SelecciÃƒÂ³ dels grups de dades que es volen analitzar/comparar (planificaciÃƒÂ³ dels anÃƒÂ lisis a aplicar).

##4.2. ComprovaciÃƒÂ³ de la normalitat i homogeneÃƒÂ¯tat de la variÃƒÂ ncia.

##4.3. AplicaciÃƒÂ³ de proves estadÃƒÂstiques per comparar els grups de dades. En funciÃƒÂ³ de les dades i de lÃ¢â‚¬â„¢objectiu de lÃ¢â‚¬â„¢estudi, aplicar proves de contrast dÃ¢â‚¬â„¢hipÃƒÂ²tesis, correlacions, regressions, etc. Aplicar almenys tres mÃƒÂ¨todes dÃ¢â‚¬â„¢anÃƒÂ lisi diferents.

En l'apartat anterior, hem integrat els conjunts d'entrenament i test, aquest fet ens ha estat ÃƒÂºtil per a obtenir una major mostra per omplir amb major confianÃƒÂ§a els buits o zeros del conjunt. No obstant aixÃƒÂ², en aquest apartat ens centrarem en la variable **Survived**. Aquesta variable ens indica si un individuo va sobreviure o no a l'accident del titÃƒÂ nic, nomÃƒÂ©s la tenim informada per a les dades del conjunt d'entrenament, per tant, nomÃƒÂ©s utilitzarem les dades d'aquest.

```
titanic_train <- dades[!is.na(dades$Survived),]
titanic_test <- dades[is.na(dades$Survived),]

write.csv(dades, file = "dadesValidated.csv")
write.csv(titanic_train, file = "trainValidated.csv")
write.csv(titanic_test, file = "testValidated.csv")
```

HipÃƒÂ²tesi nulÃ‚Â·la i alternativa

En aquest cas d'estudi, prenem per a hipÃƒÂ²tesi nulÃ‚Â·la que la mitjana de les variables numÃƒÂ¨riques **Age, TamanyFamilia, Fare, Pclass** ÃƒÂ©s independent a la variable **Survived**. Ãƒâ€°s a dir, les mitjanes de les variables anteriors pels passatgers sobrevivents i les que no sÃƒÂ³n iguals. $H_0 : \mu_1 = \mu_0$

Com a hipÃƒÂ²tesi alternativa, es tÃƒÂ© que la mitjana de cadascuna de les variables dels passatgers sobrevivents i la dels no sobrevivents ÃƒÂ©s diferent. $H_1 : \mu_1 \neq \mu_0$

On μ_1 ÃƒÂ©s la mitjana de cadascuna de les variables numÃƒÂ¨riques pels passatgers supervivents i μ_2 ÃƒÂ©s la mitjana de la variable d'estudi pels passatgers no supervivents.

AssumpciÃƒÂ³ de normalitat

Comprovar si es compleix l'assumpciÃƒÂ³ de normalitat en les dades. Per a fer-ho, s'aplica el test Shapiro-Wilk. Si el p-valor del test Shapiro Wilk ÃƒÂ©s superior a 0.05, implica que la distribuciÃƒÂ³ de les dades no ÃƒÂ©s significativament diferent de la distribuciÃƒÂ³ normal. Ãƒâ€°s a dir, podem assumir normalitat.

```
shapiro.test(titanic_train$Age)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  titanic_train$Age
## W = 0.98099, p-value = 0.000000002263
```

```
shapiro.test(titanic_train$TamanyFamilia)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  titanic_train$TamanyFamilia
## W = 0.61508, p-value < 2.2e-16
```

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

```
##
##  Shapiro-Wilk normality test
##
## data:  titanic_train$Fare
## W = 0.51848, p-value < 2.2e-16
```

```
shapiro.test(titanic_train$Pclass)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  titanic_train$Pclass
## W = 0.71833, p-value < 2.2e-16
```

```
length(titanic_train$Pclass)
```

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

Com que el p-valor ÃƒÂ©s inferior a 0,05 en tots els casos, es rebutja la hipÃƒÂ²tesi nulÃ‚Â·la del test de Shapiro-Wilk que confirma l'assumpciÃƒÂ³ de normalitat en les dades.

No obstant aixÃƒÂ², com que es tÃƒÂ© que $N = 891$ com a conseqÃƒÂ¼ÃƒÂ¨ncia del teorema del lÃƒÂmit central, es pot considerar que les dades segueixen una distribuciÃƒÂ³ normal.

HomogeneÃƒÂ¯tat de la variÃƒÂ ncia

Les variÃƒÂ ncies sÃƒÂ³n desconegudes. A continuaciÃƒÂ³, per a decidir si apliquem variÃƒÂ ncies iguals o diferents, apliquem el test de Fligner-Killena.

```
library(car)

fligner.test(Age ~ Survived, data = titanic_train)
```

```
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  Age by Survived
## Fligner-Killeen:med chi-squared = 1.7343, df = 1, p-value = 0.1879
```

```
fligner.test(TamanyFamilia ~ Survived, data = titanic_train)
```

```
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  TamanyFamilia by Survived
## Fligner-Killeen:med chi-squared = 19.647, df = 1, p-value =
## 0.000009317
```

```
fligner.test(Fare ~ Survived, data = titanic_train)
```

```
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  Fare by Survived
## Fligner-Killeen:med chi-squared = 93.468, df = 1, p-value <
## 2.2e-16
```

```
fligner.test(Pclass ~ Survived, data = titanic_train)
```

```
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  Pclass by Survived
## Fligner-Killeen:med chi-squared = 23.648, df = 1, p-value =
## 0.000001157
```

El p-valor ÃƒÂ©s superior a 0,05 en la variable age i per tant en ella podem assumir la igualtat de variÃƒÂ ncies. No obstant aixÃƒÂ², per la resta de variables el p-valor ÃƒÂ©s inferior a 0,05 i per tant s'assumeix la no igualtat de variÃƒÂ ncies.

Per tant, apliquem test t de dues mostres independents per a la diferÃƒÂ¨ncia de mitjanes, variÃƒÂ ncies desconegudes i

iguals en el cas de l'edat i apliquem test de wilcox en el cas de la mida de la famÃƒÂlia, la tarifa i la classe.

```
t.test(Age ~ Survived,data = titanic_train, var.equal=TRUE)
```

```
##
##  Two Sample t-test
##
## data:  Age by Survived
## t = 1.5319, df = 889, p-value = 0.1259
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.420076  3.408488
## sample estimates:
## mean in group 0 mean in group 1
##        30.38798        28.89377
```

```
wilcox.test(TamanyFamilia ~ Survived,data = titanic_train, var.equal=TRUE)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  TamanyFamilia by Survived
## W = 77659, p-value = 0.0000007971
## alternative hypothesis: true location shift is not equal to 0
```

```
wilcox.test(Fare ~ Survived,data = titanic_train, var.equal=TRUE)
```

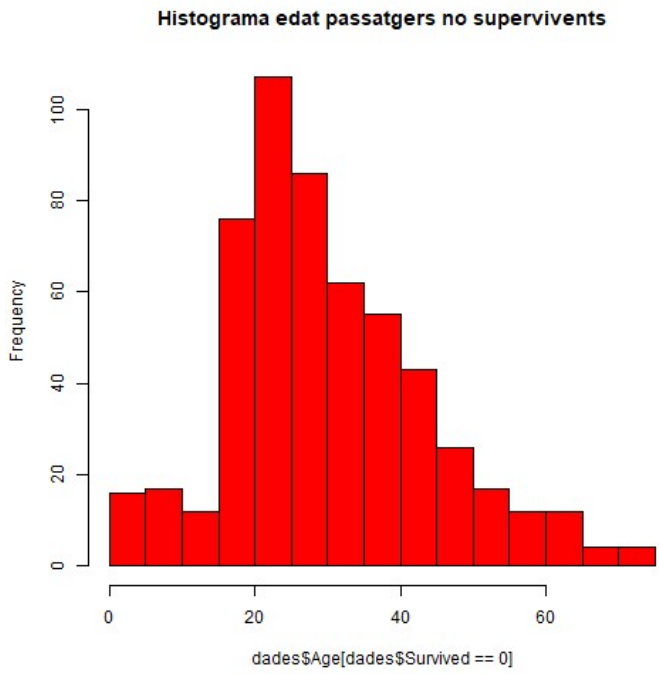
```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  Fare by Survived
## W = 59732, p-value < 2.2e-16
## alternative hypothesis: true location shift is not equal to 0
```

```
wilcox.test(Pclass ~ Survived,data = titanic_train, var.equal=TRUE)
```

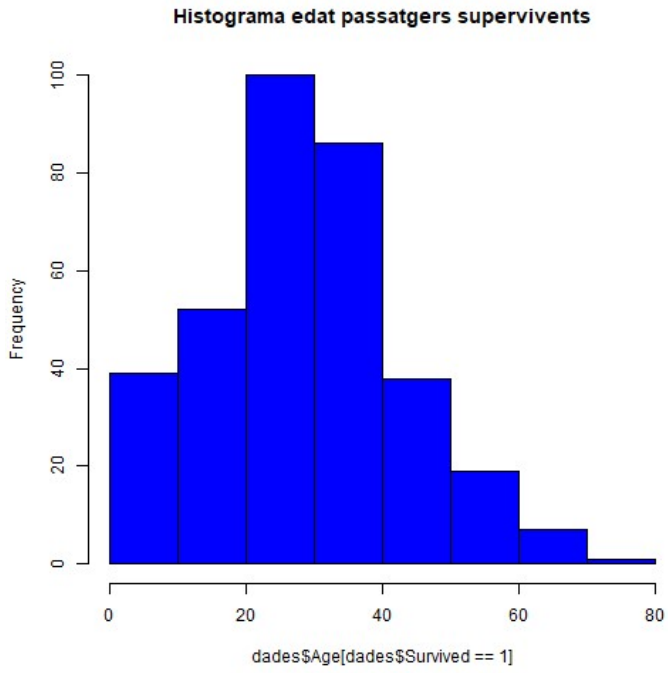
```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  Pclass by Survived
## W = 127940, p-value < 2.2e-16
## alternative hypothesis: true location shift is not equal to 0
```

El valor-p ÃƒÂ©s superior a 0,05 en el cas de l'edat i per tant s'accepta la hipÃƒÂ²tesi H_0 i s'afirma que hi ha diferÃƒÂ¨ncia estadÃƒÂstica en la mitjana d'edat en els dos grups. A continuaciÃƒÂ³ es mostren dos histogrames, un amb l'edat dels supervivents i l'altre amb els no supervivents.

```
hist(dades$Age[dades$Survived == 0], main = "Histograma edat passatgers no supervivents", col = "re
```




```
hist(dades$Age[dades$Survived == 1], main = "Histograma edat passatgers supervivents", col = "blue'
```



En els dos histogrames anteriors, es veu com la majoria de passatgers amb edat inferior a 20 anys van sobreviure al titÃƒÂ nic. Pel que fa a les altres franges d'edat, no s'aprecien diferencies.

En canvi, el valor-p ÃƒÂ©s inferior a 0,05 en el cas del dimensiÃƒÂ³ de la famÃƒÂlia, la tarifa i la classe, per tant es rebutja la hipÃƒÂ²tesi H_0 i s'accepta la hipÃƒÂ²tesi alternativa H_1 que afirmar que no hi ha diferÃƒÂ¨ncia estadÃƒÂstica de la mitjana de les dues variables en els dos grups.

CorrelaciÃƒÂ³

Per a comprovar la correlaciÃƒÂ³ entre variables, tan sols comparem les variables numÃƒÂ¨riques. Per tant, per a aquesta prova seleccionem tan sols les variables **Age**, **TamanyFamilia**, **Fare** i **Pclass**. En primer lloc calculem la matriu de correlaciÃƒÂ³ entre les variables.

```
titanic_cor <- data.frame(titanic_train$Age, titanic_train$TamanyFamilia, titanic_train$Fare, titar
cor(titanic_cor)
```

```
##                            titanic_train.Age titanic_train.TamanyFamilia
## titanic_train.Age                  1.0000000                 -0.31425111
## titanic_train.TamanyFamilia       -0.3142511                  1.00000000
## titanic_train.Fare                 0.0875904                  0.21254320
## titanic_train.Pclass              -0.3543361                  0.06599691
##                            titanic_train.Fare titanic_train.Pclass
## titanic_train.Age                   0.0875904          -0.35433611
## titanic_train.TamanyFamilia         0.2125432           0.06599691
## titanic_train.Fare                  1.0000000          -0.55989875
## titanic_train.Pclass               -0.5598987           1.00000000
```

Estudiem doncs els casos amb major correlaciÃƒÂ³:

-Variables **Pclass** i **Fare**: correlaciÃƒÂ³ de -0.5598987. -Variables **Pclass** i **Age**: correlaciÃƒÂ³ de -0.3543361. -Variables **Age** i **TamanyFamilia**: correlaciÃƒÂ³ de -0.3142511.

Abans de comprovar la correlaciÃƒÂ³ entre els parells de variables anteriors, per a decidir quin ÃƒÂ©s el mÃƒÂ¨tode mÃƒÂ©s adequat a aplicar, cal comprovar si es compleix l'assumpciÃƒÂ³ de normalitat en les dades. Per a fer-ho, tal com hem fet anteriorment, s'aplica el test Shapiro-Wilk.

```
shapiro.test(titanic_train$Pclass)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  titanic_train$Pclass
## W = 0.71833, p-value < 2.2e-16
```

```
shapiro.test(titanic_train$Age)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  titanic_train$Age
## W = 0.98099, p-value = 0.000000002263
```

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

```
##
##  Shapiro-Wilk normality test
##
## data:  titanic_train$Fare
## W = 0.51848, p-value < 2.2e-16
```

```
shapiro.test(titanic_train$TamanyFamilia)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  titanic_train$TamanyFamilia
## W = 0.61508, p-value < 2.2e-16
```

Com que en tots els casos, el p-valor ÃƒÂ©s inferior a 0,05 es rebutja la hipÃƒÂ²tesi nulÃ‚Â·la del test de Shapiro-Wilk que confirma l'assumpciÃƒÂ³ de normalitat en les dades. Per tant, per a estudiar la correlaciÃƒÂ³ entre els parells de variables estudiats aplicarem el test de Spearman.

```
cor.test(titanic_train$Pclass,titanic_train$Fare, method="spearman")
```

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

```
##
##  Spearman's rank correlation rho
##
## data:  titanic_train$Pclass and titanic_train$Fare
## S = 203770000, p-value < 2.2e-16
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##        rho
## -0.7284641
```

```
cor.test(titanic_train$Pclass,titanic_train$Age, method="spearman")
```

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

```
##
##  Spearman's rank correlation rho
##
## data:  titanic_train$Pclass and titanic_train$Age
## S = 158370000, p-value < 2.2e-16
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##        rho
## -0.3433576
```

```
cor.test(titanic_train$TamanyFamilia,titanic_train$Age, method="spearman")
```

```
## Warning in cor.test.default(titanic_train$TamanyFamilia,
## titanic_train$Age, : Cannot compute exact p-value with ties
```

```
##
##  Spearman's rank correlation rho
##
## data:  titanic_train$TamanyFamilia and titanic_train$Age
## S = 145000000, p-value = 3.684e-12
## alternative hypothesis: true rho is not equal to 0
## sample estimates:
##        rho
## -0.2299892
```

En els tres casos, el p-valor ÃƒÂ©s significatiu. Per tant, podem observar doncs una correlaciÃƒÂ³ prou significativa entre les parelles de variables estudiades, mÃƒÂ©s concretament:

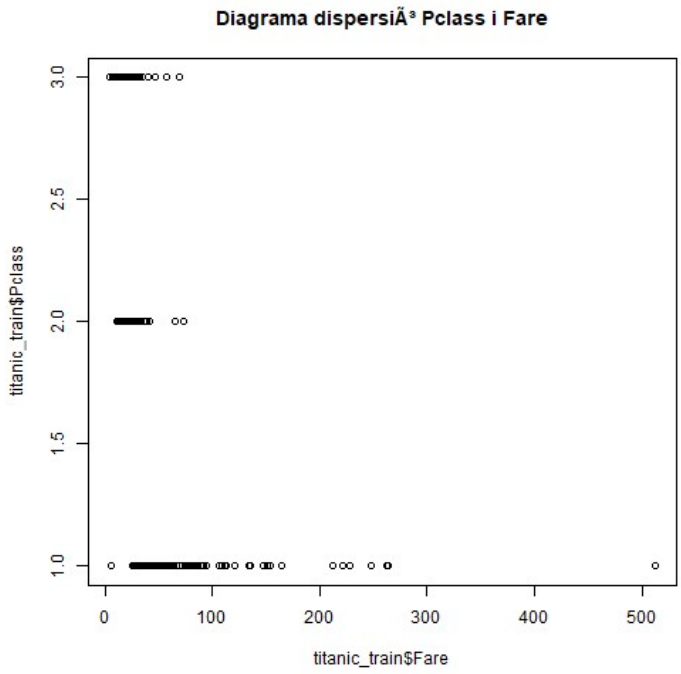
-Variables **Pclass** i **Fare**: correlaciÃƒÂ³ de -0.7284641 -Variables **Pclass** i **Age**: correlaciÃƒÂ³ de -0.3433576 -Variables **Age** i **TamanyFamilia**: correlaciÃƒÂ³ de -0.2299892

La primera resulta totalment comprensible ja que la tarifa sol anar marcada principalment per a la classe, on la primera ÃƒÂ©s la que tÃƒÂ© tarifa mÃƒÂ©s alta i la tercera la mÃƒÂ©s baixa, ÃƒÂ©s a dir correlaciÃƒÂ³ negativa.

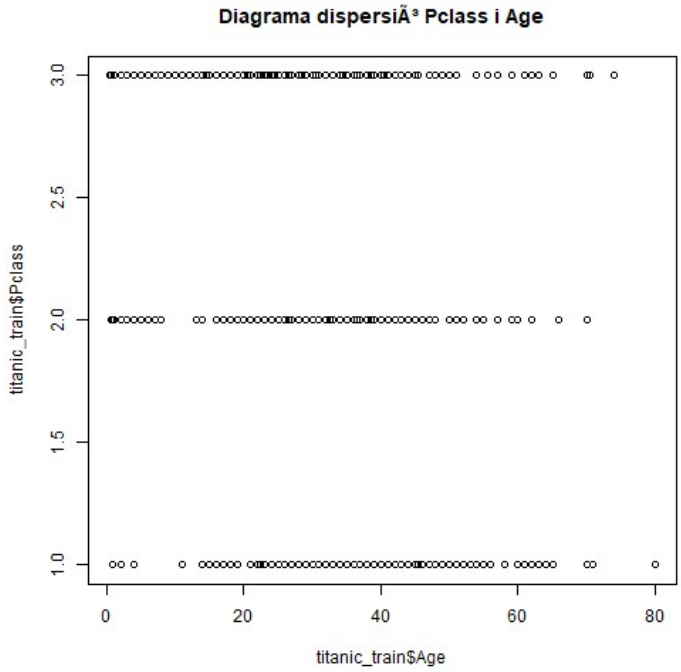
El segon resultat tambÃƒÂ© sembla coherent, ja que com mÃƒÂ©s edat tÃƒÂ© una persona, normalment acostuma a tenir major poder adquisitiu i mÃƒÂ©s comoditat busca, per tant, selecciona classes superiors.

Les variables **Age** i **TamanyFamilia** tenen una petita correlaciÃƒÂ³ negativa que tambÃƒÂ© tÃƒÂ© una possible explicaciÃƒÂ³, ja que per exemple els nens, no viatjaran sols. En canvi, els adults sÃƒÂ que podrien viatjar sols. Finalment, observem els diagrames de dispersiÃƒÂ³ de les 3 parelles anteriors.

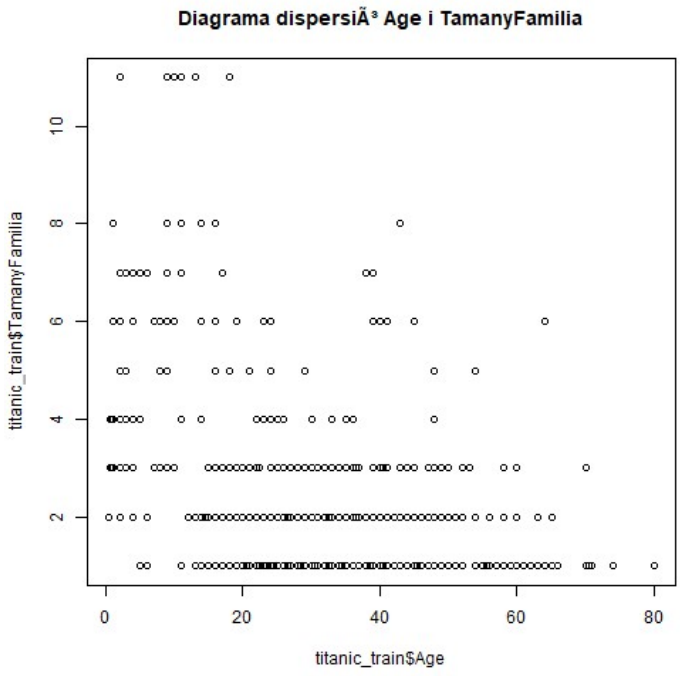
```
plot(titanic_train$Fare, titanic_train$Pclass, main="Diagrama dispersiÃƒÂ³ Pclass i Fare")
```




```
plot(titanic_train$Age, titanic_train$Pclass, main="Diagrama dispersiÃƒÂ³ Pclass i Age")
```




```
plot(titanic_train$Age, titanic_train$TamanyFamilia, main="Diagrama dispersiÃƒÂ³ Age i TamanyFamili
```



RegressiÃƒÂ³ logÃƒÂstica

En el nostre cas, el conjunt d'entrenament i de test ja venen definits per necesitat, ja que en aquest segon no hi ha la categoritzaciÃƒÂ³ de si un passatger a sobreviscut o no. En cas que no hi haguÃƒÂ©s un conjunt d'entrenament definit i per a guanyar una major eficiencia en l'algoritme predictiu, una bona estrategia seria dividir de diferents maneres els conjunts d'entrenament i test, comparant mitjanes o desviacions tipiques per tal d'aconseguir un model mes robust. Una tecnica molt comuna que permet crear diferents conjunts d'entrenaments i tests per tal d'optimitzar el model ÃƒÂ©s el K-Fold.

En primer lloc cal tractar les variables categÃƒÂ²riques. Usant el que sÃ¢â‚¬â„¢anomena variables dummy podrem calcular el model lineal. Per a fer-ho, especificarem quina ÃƒÂ©s la categoria de referÃƒÂ¨ncia amb la funciÃƒÂ³ relevel(). En la variable **Sex**, la categoria de referÃƒÂ¨ncia ÃƒÂ©s Ã¢â‚¬Å“femaleÃ¢â‚¬Â i per a la variable **Embarked** la categoria de referÃƒÂ¨ncia ÃƒÂ©s “S”

```
titanic_train$sexR <- relevel(titanic_train$Sex, ref = "female")
titanic_train$EmbarkedR <- relevel(titanic_train$Embarked, ref = "S")
```

Apliquem diversos models de regressiÃƒÂ³ logÃƒÂstica combinant diferents variables del conjunt de dades per a buscar el millor model de regressiÃƒÂ³.

```
model1<- glm(Survived ~ sexR+TamanyFamilia+Age+EmbarkedR+Pclass,data=titanic_train)
summary(model1)
```

```
##
## Call:
## glm(formula = Survived ~ sexR + TamanyFamilia + Age + EmbarkedR +
##     Pclass, data = titanic_train)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -1.00624  -0.21724  -0.07228   0.22000   0.98801
##
## Coefficients:
##                 Estimate Std. Error t value   Pr(>|t|)
## (Intercept)     1.323701   0.066827  19.808    < 2e-16 ***
## sexR[T.male]   -0.509715   0.027997 -18.206    < 2e-16 ***
## TamanyFamilia  -0.031486   0.008508  -3.701   0.000228 ***
## Age            -0.005025   0.001023  -4.910 0.00000109 ***
## EmbarkedR[T.C]  0.069263   0.033884   2.044   0.041235 *
## EmbarkedR[T.Q]  0.066257   0.047435   1.397   0.162831
## Pclass         -0.181467   0.017563 -10.332    < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 0.1446284)
##
##     Null deviance: 210.73  on 890  degrees of freedom
## Residual deviance: 127.85  on 884  degrees of freedom
## AIC: 814.69
##
## Number of Fisher Scoring iterations: 2
```

```
model2<- glm(Survived ~ sexR+Age+Pclass,data=titanic_train)
summary(model2)
```

```
##
## Call:
## glm(formula = Survived ~ sexR + Age + Pclass, data = titanic_train)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -1.05489  -0.23720  -0.07882   0.21561   0.97773
##
## Coefficients:
##                Estimate Std. Error t value  Pr(>|t|)
## (Intercept)   1.2454421  0.0571957  21.775   < 2e-16 ***
## sexR[T.male] -0.4994160  0.0275270 -18.143   < 2e-16 ***
## Age          -0.0038999  0.0009843  -3.962 0.0000803 ***
## Pclass       -0.1827548  0.0167452 -10.914   < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 0.1476099)
##
##     Null deviance: 210.73  on 890  degrees of freedom
## Residual deviance: 130.93  on 887  degrees of freedom
## AIC: 829.89
##
## Number of Fisher Scoring iterations: 2
```

```
model3<- glm(Survived ~ sexR+EmbarkedR+Pclass,data=titanic_train)
summary(model3)
```

```
##
## Call:
## glm(formula = Survived ~ sexR + EmbarkedR + Pclass, data = titanic_train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.9774  -0.2303  -0.0772   0.2609   0.9228
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)    1.04527    0.04345  24.056   <2e-16 ***
## sexR[T.male]  -0.50886    0.02755 -18.469   <2e-16 ***
## EmbarkedR[T.C]  0.08515    0.03428   2.484   0.0132 *
## EmbarkedR[T.Q]  0.06059    0.04776   1.268   0.2050
## Pclass        -0.15307    0.01645  -9.307   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 0.1491811)
##
##     Null deviance: 210.73  on 890  degrees of freedom
## Residual deviance: 132.17  on 886  degrees of freedom
## AIC: 840.32
##
## Number of Fisher Scoring iterations: 2
```

```
model4<- glm(Survived ~ sexR+TamanyFamilia+Pclass,data=titanic_train)
summary(model4)
```

```
##
## Call:
## glm(formula = Survived ~ sexR + TamanyFamilia + Pclass, data = titanic_train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.9495  -0.2526  -0.1091   0.2047   1.0192
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)    1.125021   0.043239  26.019  < 2e-16 ***
## sexR[T.male]  -0.532079   0.027951 -19.036  < 2e-16 ***
## TamanyFamilia -0.021392   0.008226  -2.600  0.00946 **
## Pclass        -0.154148   0.015688  -9.826  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 0.1490855)
##
##     Null deviance: 210.73  on 890  degrees of freedom
## Residual deviance: 132.24  on 887  degrees of freedom
## AIC: 838.76
##
## Number of Fisher Scoring iterations: 2
```

```
model5<- glm(Survived ~ sexR+Age,data=titanic_train)
summary(model5)
```

```
##
## Call:
## glm(formula = Survived ~ sexR + Age, data = titanic_train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -0.7450  -0.1894  -0.1879   0.2580   0.8142
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)   0.7391612  0.0356136  20.755   <2e-16 ***
## sexR[T.male] -0.5534355  0.0288230 -19.201   <2e-16 ***
## Age           0.0001031  0.0009723   0.106    0.916
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 0.1672435)
##
##     Null deviance: 210.73  on 890  degrees of freedom
## Residual deviance: 148.51  on 888  degrees of freedom
## AIC: 940.16
##
## Number of Fisher Scoring iterations: 2
```

Dels models anteriors (i tots els provats) seleccionem el **model1**, ja que ÃƒÂ©s el model que tÃƒÂ© major bondat per a tenir una menor AIC. A partir d'aquest model predim la probabilitat de sobreviure de cadascÃƒÂºn dels registres del conjunt de test:

```
titanic_test$sexR <- relevel(titanic_test$Sex, ref = "female")
titanic_test$EmbarkedR <- relevel(titanic_test$Embarked, ref = "S")

titanic_test$ProbSurvived <- predict(model1, titanic_test)
```

Mostrem la taula que contÃƒÂ© el resultat de les prediccions del conjunt de test:

```
library(kableExtra)
kable(head(titanic_test), format = 'markdown')
```

	PassengerId	Survived	Pclass	Name	Sex	Age	Fare	Embarked	TamanyFamilia	sexR	EmbarkedR	ProbSurvived
892	892	NA	3	Kelly, Mr. James	male	34.5	7.8292	Q	1	male	Q	0.1310039
893	893	NA	3	Wilkes, Mrs. James (Ellen Needs)	female	47.0	7.0000	S	2	female	S	0.4801679
894	894	NA	2	Myles, Mr. Thomas Francis	male	62.0	9.6875	Q	1	male	Q	0.1742924
895	895	NA	3	Wirz, Mr. Albert	male	27.0	8.6625	S	1	male	S	0.1024324
896	896	NA	3	Hirvonen, Mrs. Alexander (Helga E Lindqvist)	female	22.0	12.2875	S	3	female	S	0.5742986
897	897	NA	3	Svensson, Mr. Johan Cervin	male	14.0	9.2250	S	1	male	S	0.1677532

```
write.csv(titanic_test, file = "testPrediction.csv")
```

5. RepresentaciÃƒÂ³ dels resultats a partir de taules i grÃƒÂfiques.

Al llarg d'aquesta prÃƒÂctica s'han mostrat diferents grÃƒÂfiques o taules per a mostrar els resultats.

6. ResoluciÃƒÂ³ del problema. A partir dels resultats obtinguts, quines sÃƒÂ³n les conclusions? Els resultats permeten respondre al problema?

6.1.- ProcÃ©s de neteja i validaciÃ³

Es partia de dos conjunts de dades relatives als passatgers del Titanic. Un conjunt de training (on sâ€™informava el camp Survived) i un conjunt de test (on no sâ€™informava el camp Survived). El dataset sâ€™utilitza per generar models de predicciÃ³ de la probabilitat de supervivÃ¨ncia en funciÃ³ de les caracterÃstiques del passatger. Per a la fase de neteja i validaciÃ³ de les dades sâ€™ha optat per integrar els dos fitxers per disposar de mÃ©s registres que permetessin aplicar tÃ¨cniques per completar camps no informats amb major precisiÃ³. Dâ€™una anÃ lisi preliminar sâ€™ha comprovat que el conjunt de dades presentava nombrosos camps no informats (en especial el camp Age). Sâ€™ha procedit a analitzar aquests camps i aplicar diverses tÃ¨cniques per completar-los.

El camp Fare (Tarifa) presentava un valor no informat. Per deduir-lo sâ€™han obtingut tots els registres que compartien classe i origen dâ€™embarcament, sâ€™ha obtingut la mitja i sâ€™ha assignat al valor no informat.

Per al camp Embarket (Origen dâ€™embarcament) , sâ€™han obtingut els registres no informats i sâ€™observa que tots ells comparteixen tarifa i classe. Sâ€™ha obtingut la mitjana de la tarifa pels passatgers de classe 1 per cada port dâ€™origen i sâ€™ha assignat el port origen que tenia una tarifa mitjana mÃ©s propera.

En el cas del camp Age, donat que hi havia molts camps no informats i era important conservar-los, sâ€™ha optat per aplicar la metodologia Mice (Multivariate Imputations by Chained Equations).

Finalment pel camp Fare, sâ€™han substituÃ¯t els valors 0 per valors no informats (per requeriment de lâ€™algoritme kNN) i sâ€™ha utilitzat lâ€™algoritme kNN per estimar-los.

Sâ€™han creat camps nous com TamanyFamilia que Ã©s la agrupaciÃ³ dels camps SibSp i Parch, el que ha permÃ¨s reduir la dimensiÃ³ del dataset i sâ€™han generat els camps factor TipusFamilia i TipusEdat per si eren interessants per lâ€™aplicaciÃ³ de certs mÃ¨todes.

El dataset final esta format pels camps PassangerId, Survival, Pclass, Name, Sex , Age, Fare, Embarked i TamanyFamilia.

6.2.- AnÃƒÂlisi de dades

Fent l'anÃƒÂlisi de dades hem arribat a la conclusiÃƒÂ³, mitjanÃƒÂ§ant tests d'hipÃƒÂ²tesis, que en general els passatgers del titÃƒÂnic menor de vint anys tenien un major ÃƒÂndex de supervivÃƒÂ¨ncia. MitjanÃƒÂ§ant aquesta mateixa prova hem vist que ni per la mida de la famÃƒÂlia, ni per la tarifa, ni per la classe en la qual es viatjava, s'aprecien diferencies en l'ÃƒÂndex de supervivÃƒÂ¨ncia.

Aplicant un test de correlaciÃƒÂ³, hem detectat correlaciÃƒÂ³ negativa entre el preu del bitllet i la classe que es viatja, l'edat del passatger i la classe en la qual viatge o bÃƒÂ© l'edat del passatger i el nombre de familiars a bord.

Finalment i mitjanÃƒÂ§ant un algoritme de regressiÃƒÂ³ logistica, hem escollit un model entre els diferents que hem creat que ens a permÃƒÂ©s predir la probabilitat de supervivencia dels individuos del conjunt de test.

Contribucions	Firma
InvestigaciÃƒÂ³ prÃƒÂ¨via	PC CR
RedacciÃƒÂ³ de les respostes	PC CR
Desenvolupament codi	PC CR