PRA2 - Neteja i anàlisi de les dades

Autor: Jordi Puig Ovejero

Desembre 2020

Carrega de les llibreries que es necessiten

```
library(ggpubr)
library(ggplot2)
library(arules)
library(dplyr)
library(factoextra)
library(FactoMineR)
library(nortest)
library(plyr)
library(randomForest)
library(caret)
library(VIM)
library(DescTools)
```

Descripció del dataset. Perquè és important i quina pregunta/problema pretén respondre?

Per a realitzar un estudi he triat un dataset de marqueting bancari.

Les dades estan relacionades amb campanyes de màrqueting directe d'una institució bancària portuguesa. Aquestes campanyes, basades en trucades de telèfon, buscaven clients que contractessin un dipòsit a termini. Sovint, es requeria més d'una trucada amb el mateix client per a concretar la transacció.

Aquest dataset permet respondre a la pregunta per predir si un client subscriurà ("yes"/"no") un dipòsit a termini a partir d'una sèrie de dades del client (age, job, marital, education…). Si tenim una bona segmentació dels clients i fem una bona predicció el banc pot fer campanyes molt dirigides a obtenir resultats positius.

Apart d'aquesta pregunta predictiva, també volem saber si hi ha alguna relació entre la quantitat de diners que els usuaris tenen al banc amb altre informació emmagatzemada (estudis, estat civil, …).

Durant l'exercici direm que un usuari ha convertit, és a dir, tenim una conversió, si y = 'yes'. En cas contrari, direm que no ha convertit. El concepte **conversió** sortirà durant tot l'exercici.

El fitxer on es troba el data set és, **bank-full.csv**:

- Nombre d'instancies: 45211
- Nombre d'instancies: 16 + atribut de sortida (total 17)

Definició dels atributs

Dades del client:

- 1 - age (numeric)
- 2 - job : tipus de feina (categorical):
  - 'admin.'
  - 'unknown'
  - 'unemployed'
  - 'management'
  - 'housemaid'
  - 'entrepreneur'
  - 'student'
  - 'blue-collar'
  - 'self-employed'
  - 'retired'
  - 'technician'
  - 'services'
- 3 - marital : estat civil (categorical):
  - 'divorced': significa divorciat o vidu
  - 'married'
  - 'single'
- 4 - education (categorical):
  - 'primary'
  - 'seconday'
  - 'tertiary'
  - 'unknown'
- 5 - default: té crèdit per defecte? (categorical: 'no','yes')
- 6 - balance: mitja de saldo anual, en euros (numeric)
- 7 - housing: té préstec d'habitatge? (categorical: 'no','yes')
- 8 - loan: té préstec personal? (categorical: 'no','yes')

Atributs relacionats amb el darrer contacte de la campanya actual:

- 9 - contact: com ha estat la comunicació (categorical: 'cellular','telephone','unknown')
- 10 - month: darrer contacte, mes de l'any (categorical: 'jan', 'feb', 'mar', …, 'nov', 'dec')
- 11 - day: darrer contacte, dia del més (numeric)
- 12 - duration: darrer contacte, en segons (numeric).

Altres atributs:

- 13 - campaign: nombre de contactes realitzats durant aquesta campanya i per a aquest client (numeric, inclou el darrer contacte)

- 14 - pdays: nombre de dies que han passat des de el darrer contacte d'una campanya anterior (numeric; -1 significa que no ha estat contactat previament)
- 15 - previous: nombre de contactes realitzats abans d'aquesta campanya i per a aquest client (numeric)
- 16 - poutcome: resultat de la campanya de màrqueting anterior (categorical: 'unknown','other','failure','success')

Variable de sortida (objectiu desitjat, ens indica si el registre ha convertit o no)

- 17 - y - El client ha subscrit el dipòsit a termini? (binary: 'yes','no')

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

Carreguem les dades i fem un summary. Els elements categòrics estan carregats com a *factor* des d'un inici. Per les variables categòriques prefereixo treballar com a factor que com character. Ens facilita les coses per ordenació o per veure les diferents categories que hi ha.

```
bank <- read.csv('bank-full.csv',stringsAsFactors = TRUE, sep = ';')
attach(bank) # ens permet referenciar les columnes de bank sense haver de
especificar el dataset.
summary(bank)
```

```
##       age                 job           marital          education
##  Min.   :18.00   blue-collar:9732   divorced: 5207   primary  : 6851
##  1st Qu.:33.00   management :9458   married :27214   secondary:23202
##  Median :39.00   technician :7597   single  :12790   tertiary :13301
##  Mean   :40.94   admin.     :5171                    unknown  : 1857
##  3rd Qu.:48.00   services   :4154
##  Max.   :95.00   retired    :2264
##                  (Other)    :6835
##  default        balance       housing      loan            contact
##  no :44396   Min.   : -8019   no :20081   no :37967   cellular :29285
##  yes:  815   1st Qu.:    72   yes:25130   yes: 7244   telephone: 2906
##              Median :   448                           unknown  :13020
##              Mean   :  1362
##              3rd Qu.:  1428
##              Max.   :102127
##
##       day            month          duration         campaign
##  Min.   : 1.00   may    :13766   Min.   :   0.0   Min.   : 1.000
##  1st Qu.: 8.00   jul    : 6895   1st Qu.: 103.0   1st Qu.: 1.000
##  Median :16.00   aug    : 6247   Median : 180.0   Median : 2.000
##  Mean   :15.81   jun    : 5341   Mean   : 258.2   Mean   : 2.764
##  3rd Qu.:21.00   nov    : 3970   3rd Qu.: 319.0   3rd Qu.: 3.000
##  Max.   :31.00   apr    : 2932   Max.   :4918.0   Max.   :63.000
##                  (Other): 6060
##      pdays          previous           poutcome       y
##  Min.   : -1.0   Min.   :  0.0000   failure: 4901   no :39922
##  1st Qu.: -1.0   1st Qu.:  0.0000   other  : 1840   yes: 5289
##  Median : -1.0   Median :  0.0000   success: 1511
##  Mean   : 40.2   Mean   :  0.5803   unknown:36959
##  3rd Qu.: -1.0   3rd Qu.:  0.0000
##  Max.   :871.0   Max.   :275.0000
##
```

Amb aquesta informació podem dir que:

- La mitja d'edat està quasi en els 41 anys i el 3Q en els 48. Per tant intueixo que gran part de la mostra està en una franja d'edat relativament jove.
- Tenim valors 'unknown' a education però no és molt significatiu.
- A balance veiem possibles valors atípics (outliers).
- Tenim molts valors unkwown en el registre contact. Així que segurament no els esborrarem i farem servir com una categoria més o els haurem d'inferir.
- Al mes de maig és on tenim més mostra.
- La mitja de la duració de la trucada és d'uns 3 minuts. 1Q = 1.6 minuts aprox. i 3Q = 5.3 minunts. Tenim alguns valors atípics amb una trucada de 82 minuts.
- campaign: sembla que tenim outliers ja que el 3Q està a 3 i tenim un 63 (63 trucades durant aquesta campanya)
- previous: el mateix passar amb previous (contactes previs a la campanya).
- poutcome: té molts valors unknown que no esborrarem per no reduïr tant la mostra i ens serviran com una categoria pròpia segurament.

Neteja de les dades.

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

```
colSums(is.na(bank))
```

```
##       age       job   marital education   default   balance   housing
loan
##         0         0         0         0         0         0         0
0
##   contact       day     month  duration  campaign     pdays  previous
poutcome
##         0         0         0         0         0         0         0
0
##         y
##         0
```

No tenim valors nulls però hi ha 'unknown', com hem vist en l'anàlisi exploratori, en alguns atributs categòrics que ara tractarem.

Tenim 3 opcions per aquests valors:

1. Eliminar els registres
2. Assignar per un valor estimat
3. No fer res i tractar-los com a una categoria.

Optarem per la solució 3 segons la proporció d'elements de la mostra ja que:

- Els valors 'unknown' poden representar per si mateix una categoria única.
- Pot haver-hi una diferència important de dades si eliminem els 'unknown'.
- Tot el que ha causat el camp 'unknown' pot estar relacionat amb el resultat.

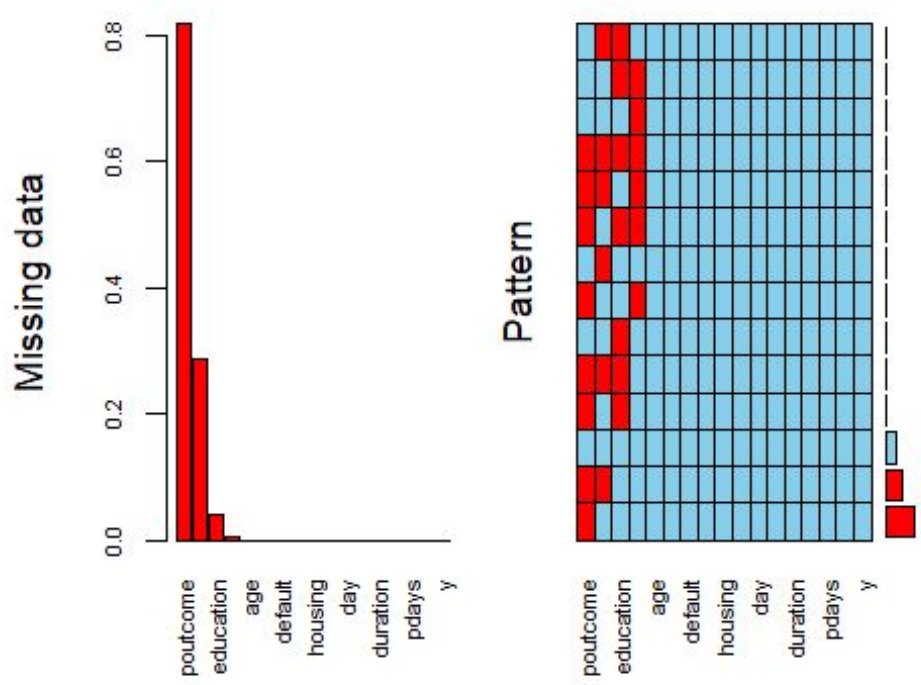
Els transformem a null en una variable temporal i visualitzem gràficament la quantitat.

```
bank.unknown <- bank
bank.unknown[bank.unknown=="unknown"] <- NA
colSums(is.na(bank.unknown))
```

```
##       age       job   marital education   default   balance   housing
loan
##         0       288         0      1857         0         0         0
0
##   contact       day     month  duration  campaign     pdays  previous
poutcome
##     13020         0         0         0         0         0         0
```

```
36959
##         y
##         0
```

```
aggr(bank.unknown, numbers=TRUE, sortVars=TRUE, labels=names(bank.unknown),
cex.axis=.7, gap=3, ylab=c("Missing data","Pattern"))
```




```
##
##  Variables sorted by number of missings:
##   Variable       Count
##   poutcome 0.817478047
##    contact 0.287983013
##  education 0.041074075
##        job 0.006370131
##        age 0.000000000
##    marital 0.000000000
##    default 0.000000000
##    balance 0.000000000
##    housing 0.000000000
##       loan 0.000000000
##        day 0.000000000
##      month 0.000000000
##   duration 0.000000000
##   campaign 0.000000000
##      pdays 0.000000000
##   previous 0.000000000
##          y 0.000000000
```

Veiem on tenim els valors 'unknown' i com sabiem de l'anàlisi exploratori previ, en tenim molts a poutcome, contact i en menor mesura a education i job.

Eliminem els 'unknown' de job i education perquè la proporció és petita, en canvi, amb contact i poutcome podríem fer una assignació estimada o deixar-los com a categoria pròpia, ja que tenim quasi un 30% en contact i més d'un 80% en poutcome.

Eliminem els de education i job i deixem els de contact i poutcome com a categoria pròpia.

```
# eliminem els unknown de job i de education
total.rows <- nrow(bank);
bank.clean <-subset(bank, education != "unknown")
bank.clean <- subset(bank.clean, job != "unknown")

# eliminem categories buides
bank.clean <- droplevels(bank.clean)

rows <- nrow(bank.clean);
(rows / total.rows) * 100
```

```
## [1] 95.53648
```

Eliminant els 'unknown' de education i job encara tenim més del 95.5% de la mostra.

Identificació i tractament de valors extrems. (outliers)

Ara anem a veure els valors atípics. Per a trobar valors extrems anem a aplicar la idea dels IQR (interquartile ranges):

- referència1:
- referència2:

Per a una determinada variable contínua, els outliers són aquelles observacions que es troben fora de 1.5 * IQR, on IQR, el "Inter Quartile Range" és la diferència entre el Q3 i el Q1:

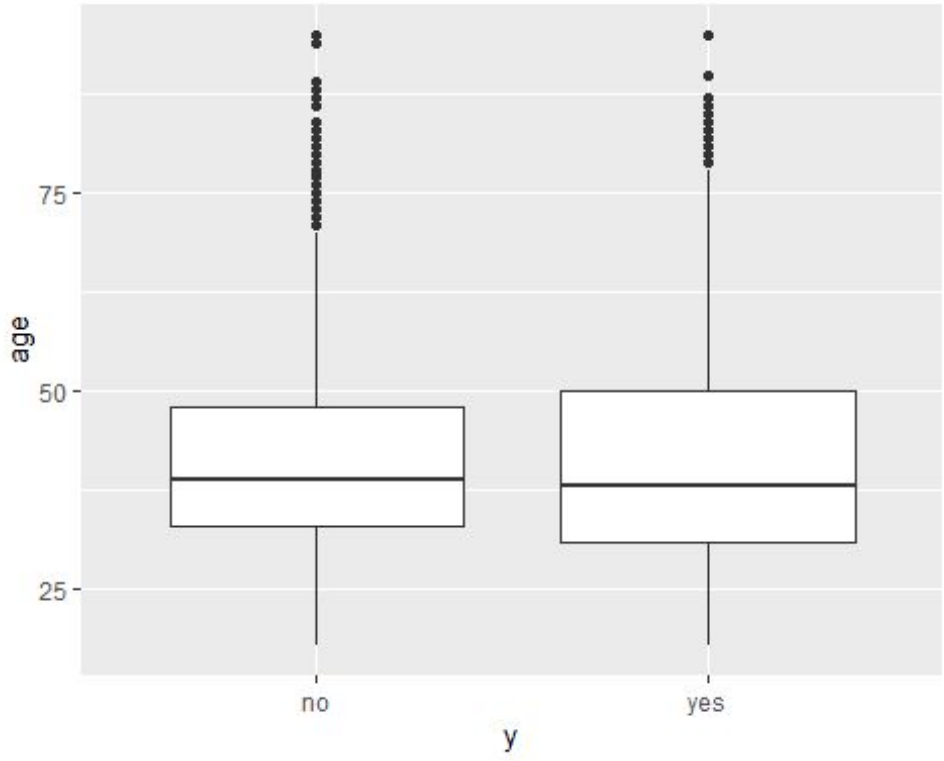
- Interquartile range, IQR = Q3 - Q1
- lower = Q1 - 1.5 * IQR
- Upper = Q3 + 1.5 * IQR

Funció per treure els límits dels valors atípics

```
outliersLimits <- function(x) {
    limits <- c("above", "under")
    limits$above <- quantile(x, 0.75, type=6) + 1.5 * (quantile(x, 0.75,
type=6) - quantile(x, 0.25, type=6))
    limits$under <- quantile(x, 0.25, type=6) - 1.5 * (quantile(x, 0.75,
type=6) - quantile(x, 0.25, type=6))
  return(limits)
}
```

Valors atípics a age

```
ggplot(data = bank.clean ,aes(x=y,y=age))+geom_boxplot()
```



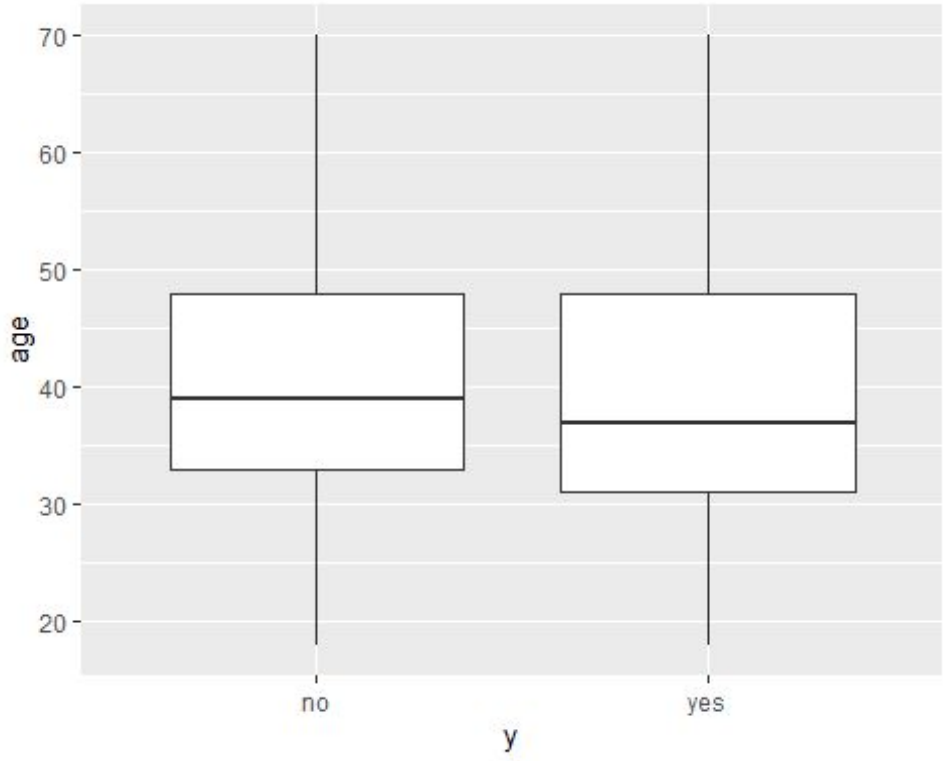

```
# podem veure els límits de age
limits <- outliersLimits(bank.clean$age)
paste("Límit inferior:", limits$under)
```

```
## [1] "Límit inferior: 10.5"
```

```
paste("Límit superior:", limits$above)
```

```
## [1] "Límit superior: 70.5"
```

```
# treiem aquest valors atípics
bank.clean <-subset(bank.clean, age >= limits$under)
bank.clean <-subset(bank.clean, age <= limits$above)

ggplot(data = bank.clean,aes(x=y,y=age))+geom_boxplot()
```

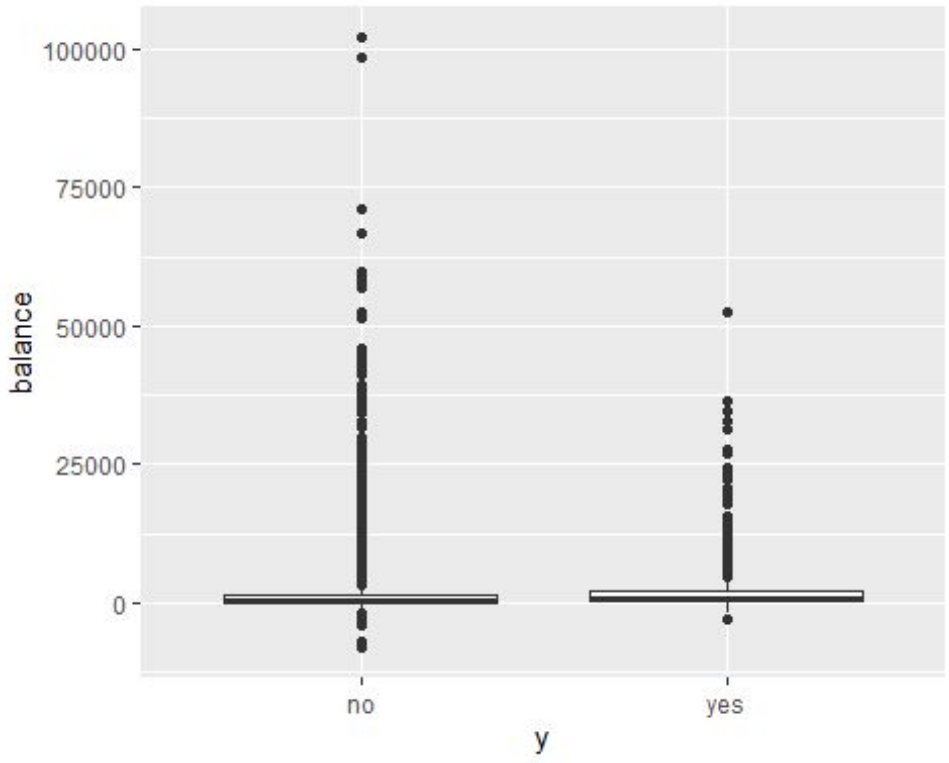



```
total.rows <- nrow(bank.clean);
rows <- nrow(bank.clean)
(rows / total.rows) * 100

## [1] 100
```

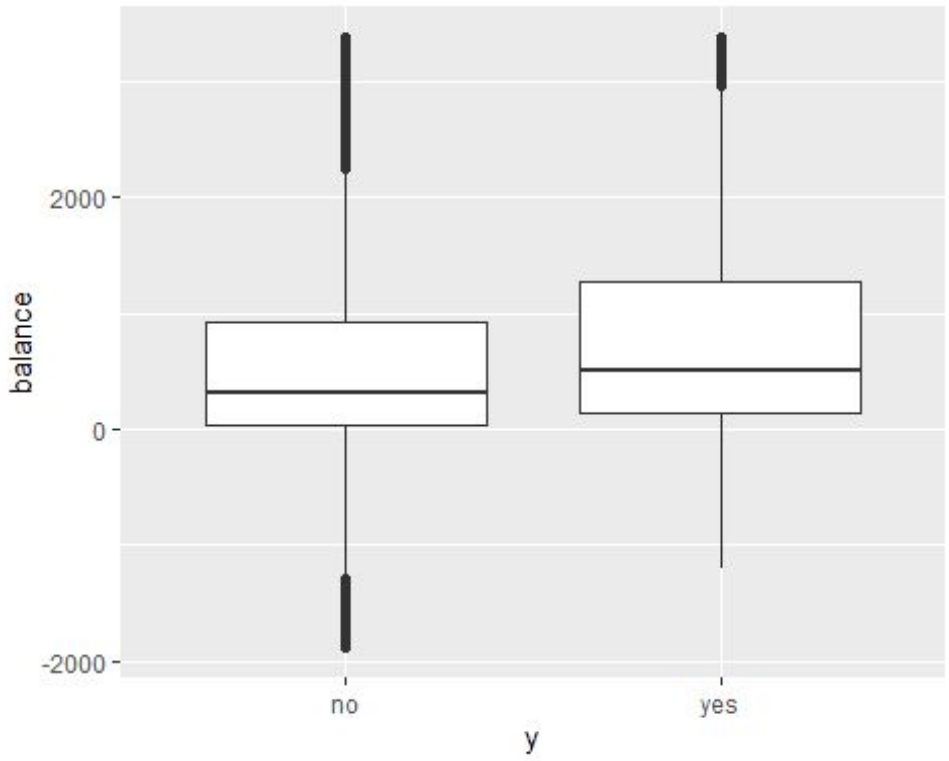
Amb graf boxplot podem preveure els valors atípics abans de ser tractats i posteriorment, on és veu la mostra molt més compactada.

Valors atípics a balance

```
ggplot(data = bank.clean ,aes(x=y,y=balance))+geom_boxplot()
```




```
total.rows <- nrow(bank.clean);

# podem veure els límits de balance
limits <- outliersLimits(bank.clean$balance)
paste("Límit inferior:", limits$under)

## [1] "Límit inferior: -1923"

paste("Límit superior:", limits$above)

## [1] "Límit superior: 3389"

# treiem aquest valors atípics
bank.clean <-subset(bank.clean, balance >= limits$under)
bank.clean <-subset(bank.clean, balance <= limits$above)

ggplot(data = bank.clean,aes(x=y,y=balance))+geom_boxplot()
```




```
rows <- nrow(bank.clean)
(rows / total.rows) * 100

## [1] 89.46187
```

Les mostres queden molt més compactades eliminant aquests valors extrems.

```
total.rows <- nrow(bank);
rows <- nrow(bank.clean);
(rows / total.rows) * 100

## [1] 84.60994
```

Finalment hem calculat quin percentatge de la mostra hem eliminat amb els valors atípics i el 'unkwown' i ens ha quedat un 84.61% del total.

Treballarem el model bank.clean on hem tret alguns unknown i valors atípics.

Reducció de la dimensionalitat

Segurament amb una mostra menor d'atributs podem tenir un model similar però reduint les dimensions. Detectem i eliminem aquells atributs poc rellevants o redundant. Agafem els components que aportin més variança al total.

Dades quantitatives. Principal Component Analysis (PCA)

La funció prcom treballa amb dades quantitatives, per tant agafem només aquells atributs que són numèrics.

```
bank.pca <- prcomp(bank.clean[,c(1,6,10,12:15)], center = TRUE, scale = TRUE)
summary(bank.pca)
```

```
## Importance of components:
##                           PC1    PC2    PC3    PC4    PC5    PC6     PC7
## Standard deviation     1.2288 1.0771 1.0370 0.9936 0.9492 0.9050 0.73984
## Proportion of Variance 0.2157 0.1657 0.1536 0.1410 0.1287 0.1170 0.07819
## Cumulative Proportion  0.2157 0.3814 0.5351 0.6761 0.8048 0.9218 1.00000
```

El resultat no ens ajuda gaire ja que amb PC5 només tenim un 80% de la variança i si treiem un component en tenim un 92%.

No treurem doncs cap dels atributs.

Exportació de dades netejades

Un cop hem netejat les dades les anem a emmagatzemar físicament en un arxiu csv.

```
write.csv(bank.clean, "bank_clean.csv")
```

Anàlisi de les dades.

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

Anem a seleccionar els grups del nostre conjunt de dades per a realitzar anàlisi i realitzar les comparatives.

Agrupació per estudis (education)

```
bank.education.primary <- bank.clean[bank.clean$education == "primary",]
bank.education.secondary <- bank.clean[bank.clean$education == "secondary",]
bank.education.tertiary <- bank.clean[bank.clean$education == "tertiary",]
```

Agrupació per estat civil (marital)

```
bank.marital.divorced <- bank.clean[bank.clean$marital == "divorced",]
bank.marital.married <- bank.clean[bank.clean$marital == "married",]
bank.marital.single <- bank.clean[bank.clean$marital == "single",]
```

Agrupació per si te prestec d'habitatge (housing)

```
bank.housing.yes <- bank.clean[bank.clean$housing == "yes",]
bank.housing.no <- bank.clean[bank.clean$housing == "no",]
```

Agrupació per si té un crèdit per defecte (default)

```
bank.default.yes <- bank.clean[bank.clean$default == "yes",]
bank.default.no <- bank.clean[bank.clean$default == "no",]
```

Després veurem si fem servir totes les variables que hem generat.

Comprovació de la normalitat i homogeneïtat de la variància.

Normalitat

Per a comprovar que les dades del dataset **(les numèriques)** provenen d'una mostra distribuida amb normalitat farem servir la prova de normalitat d'Anderson-Darling i Shapiro-Wilk.

Per un conjunt de dades d'una mostra, avaluem si provenen d'un distribució específica (el nostre cas distribució normal). El que farem és que per cada un dels atributs de la mostra fer el test per si el p-valor és superior al valor prefixat de α = 0,05. Si es cumpleix, podem dir que tenim una distribució normal.

Test de Anderson-Darling:

```
alpha = 0.05
col.names = colnames(bank.clean)
```

```
for (i in 1:ncol(bank.clean)) {
  if (is.integer(bank.clean[,i]) | is.numeric(bank.clean[,i])) {
    p_val = ad.test(bank.clean[,i])$p.value
    if (p_val < alpha) {
      cat("La variable", col.names[i],"NO segueix una distribució normal.\n")
    } else {
      cat("La variable", col.names[i],"SI segueix una distribució normal.\n")
    }
  }
}
```

```
## La variable age NO segueix una distribució normal.
## La variable balance NO segueix una distribució normal.
## La variable day NO segueix una distribució normal.
## La variable duration NO segueix una distribució normal.
## La variable campaign NO segueix una distribució normal.
## La variable pdays NO segueix una distribució normal.
## La variable previous NO segueix una distribució normal.
```

Cap de les variables estudiades sembla seguir una distribució normal.

Provarem com hem comentat també el test de Shapiro-Wilk estudiat als apunts. Es considera un dels més potents per a estudiar la normalitat. La dinàmica es similar, assumint com a hipòtesi nul·la que la mostra està distribuïda normalment, si el p-valor és menor que α = 0,05, es rebutja la hipòtesi i es conclou que les dades no provenen d'una distribució normal.

Aquest test només treballa amb menys de 5000 elements per tant agafarem una mostra aleatòria d'aquesta magnitud (10%) . La nostre mostra inicial és molt més gran.

```
random.data <- sample(1:nrow(bank.clean), 0.10 * nrow(bank.clean))
data.test <- bank.clean[random.data,]

alpha = 0.05
col.names = colnames(bank.clean)

for (i in 1:ncol(data.test)) {
  if (is.integer(data.test[,i]) | is.numeric(data.test[,i])) {
    p_val = shapiro.test(data.test[,i])
    if (p_val$p.value < alpha) {
      cat("La variable", col.names[i],"NO segueix una distribució normal.\n")
    } else {
      cat("La variable", col.names[i],"SI segueix una distribució normal.\n")
    }
  }
}
```

```
## La variable age NO segueix una distribució normal.
## La variable balance NO segueix una distribució normal.
## La variable day NO segueix una distribució normal.
```

```
## La variable duration NO segueix una distribució normal.
## La variable campaign NO segueix una distribució normal.
## La variable pdays NO segueix una distribució normal.
## La variable previous NO segueix una distribució normal.
```

Amb els dos test concluim que els valors de les mostres no provenen d'una mostra amb una distribució normal.

Homogeneïtat de la variància

Ara comprovarem l'homoscedasticitat o igualtat entre les variàncies dels grups que comparem. Anem a aplicar el test de Fligner-Killeen que es fa servir quan les dades no segueixen la condició de normalitat, com es el nostre cas. La hipòtesi nul·la assumeix igualtat de variàncies en els grups de dades. Així p-valors inferiors a 0,05 indicaran que variàncies diferents (heteroscedasticitat).

Farem la prova amb els clients que tenen estat civil 'divorced', 'married' o single respecte al saldo en el banc.

```
fligner.test(balance ~ marital, data = bank.clean)

##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  balance by marital
## Fligner-Killeen:med chi-squared = 85.933, df = 2, p-value < 2.2e-16
```

El p-valor és inferior a 0,05. Per tant, concluim que les variàncies son heterogènies.

Fem el mateix, però ara amb el nivell d'estudis i el saldo al banc.

```
fligner.test(balance ~ education, data = bank.clean)

##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  balance by education
## Fligner-Killeen:med chi-squared = 204.39, df = 2, p-value < 2.2e-16
```

Ens passa el mateix.

Finalment, fem aquesta mateixa prova amb la variable balance i la variable de sortida 'y' (contracte o no un prèstec).

```
fligner.test(balance ~ y, data = bank.clean)

##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  balance by y
## Fligner-Killeen:med chi-squared = 154.7, df = 1, p-value < 2.2e-16
```

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.

Quines variables influeixen més en la quantitat de diners al banc? (Correlació)

Amb la correlació volem mesurar l'impacte que té una variable sobre una altre. El coeficient pot prendre els valors entre 1 i -1, on els extrems indiquen una relació perfecta i el 0 indica que no tenim relació. Els signe negatiu ens indica el valor elevat d'una están relacionats amb valors petits dels altres i el signe positiu que van de la mà en quant a valors grans o petits.

Realitzem un anàlisi per determinar quines variables tenen més impacte sobre la quantitat de diners que els clients tenen al banc. Com que les dades que tenim no segueixen una distribució normal farem servir el coeficient de correlació de **Spearman**.

```
# passem totes les variables a numèriques
bank.clean.tmp <- bank.clean
for (colname in colnames(bank.clean.tmp)) {
  bank.clean.tmp[colname] <- lapply(bank.clean.tmp[colname],as.integer)
}

# creem una matriu amb dos columnes (estimate, p-value)
corr_matrix <- matrix(nc = 2, nr = 0)
colnames(corr_matrix) <- c("estimate", "p-value")

# calculem el coeficient de correlació per a cada variable quantitativa
respecte al camp "balance"
for (i in 1:(ncol(bank.clean.tmp) - 1)) {
  if (is.integer(bank.clean.tmp[,i]) | is.numeric(bank.clean.tmp[,i])) {
    spearman_test = cor.test(as.numeric(bank.clean.tmp[,"balance"])
,bank.clean.tmp[,i] , method = "spearman")
    corr_coef = spearman_test$estimate
    p_val = spearman_test$p.value

    # afegim el parell de valors (estimate i p.value) a la matriu
    pair = matrix(ncol = 2, nrow = 1)
    pair[1][1] = corr_coef
    pair[2][1] = p_val
    corr_matrix <- rbind(corr_matrix, pair)
    rownames(corr_matrix)[nrow(corr_matrix)] <- colnames(bank.clean.tmp)[i]
  }
}

print(corr_matrix)
```

```
##                estimate       p-value
## age         0.057046554  5.974226e-29
## job         0.015678261  2.165800e-03
## marital     0.029633830  6.752018e-09
## education   0.059804654  1.174773e-31
## default    -0.168224732 8.614635e-241
## balance     1.000000000  0.000000e+00
## housing    -0.039780065  7.073495e-15
## loan       -0.105263318  1.094021e-94
## contact    -0.027361052  8.688941e-08
## day        -0.009885936  5.317310e-02
## month       0.003485406  4.954486e-01
## duration    0.040040173  4.722417e-15
## campaign   -0.033713554  4.238758e-11
## pdays       0.070511719  2.290827e-43
## previous    0.077778941  2.075039e-52
## poutcome   -0.074370441  4.666118e-48
```

Segons els valors més propers a 1 o -1 podem determinar quines tenen més relació. Les que tenen més relació amb els diners al banc són loan(té prestec personal) i sobretot default(té un crèdit).

Quines variables influeixen més si contractaran un dipòsit a termino o no? (Correlació)

Anem a realitzar un estudi similar però ara amb la variable de sortida 'y', que ens diu si es va a contractar un dipòsit a termini.

```
# passem totes les variables a numèriques
bank.clean.tmp <- bank.clean
for (colname in colnames(bank.clean.tmp)) {
  bank.clean.tmp[colname] <- lapply(bank.clean.tmp[colname],as.integer)
}

# creem una matriu amb dos columnes (estimate, p-value)
corr_matrix <- matrix(nc = 2, nr = 0)
colnames(corr_matrix) <- c("estimate", "p-value")

# calculem el coeficient de correlació per a cada variable quantitativa
respecte al camp "balance"
for (i in 1:(ncol(bank.clean.tmp) - 1)) {
  if (is.integer(bank.clean.tmp[,i]) | is.numeric(bank.clean.tmp[,i])) {
    spearman_test = cor.test(as.numeric(bank.clean.tmp[,"y"])
,bank.clean.tmp[,i] , method = "spearman")
    corr_coef = spearman_test$estimate
    p_val = spearman_test$p.value

    # afegim el parell de valors (estimate i p.value) a la matriu
    pair = matrix(ncol = 2, nrow = 1)
```

```
    pair[1][1] = corr_coef
    pair[2][1] = p_val
    corr_matrix <- rbind(corr_matrix, pair)
    rownames(corr_matrix)[nrow(corr_matrix)] <- colnames(bank.clean.tmp)[i]
  }
}

print(corr_matrix)
```

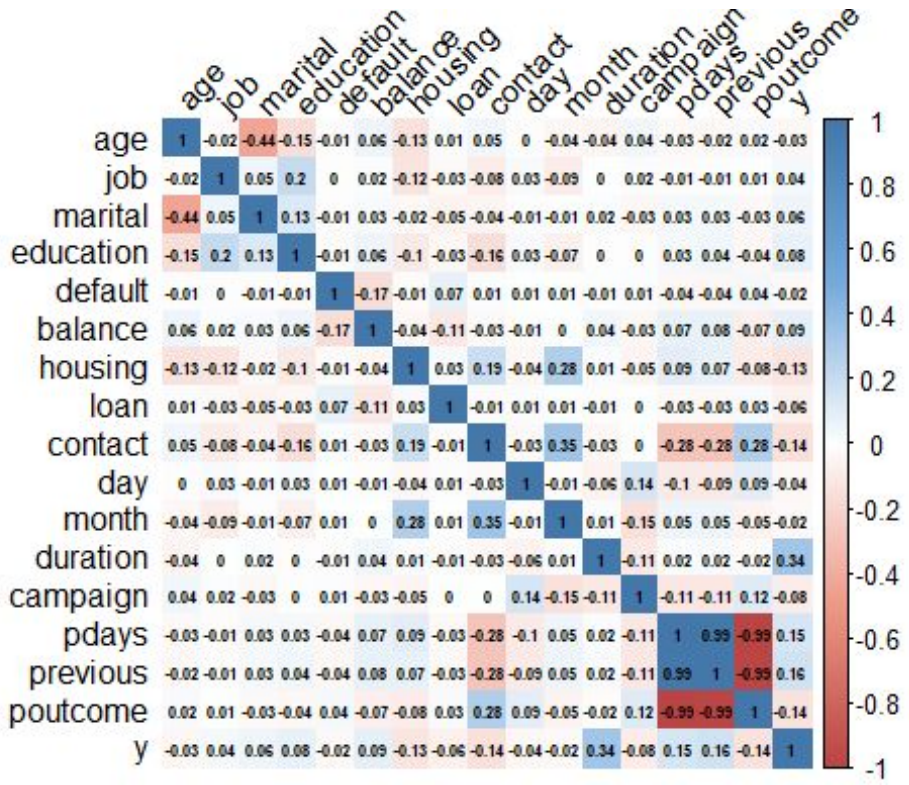
```
##              estimate       p-value
## age       -0.03473653  1.077654e-11
## job        0.03738466  2.590166e-13
## marital    0.05803506  6.620720e-30
## education  0.07956279  9.150336e-55
## default   -0.02156086  2.472390e-05
## balance    0.08772125  3.186323e-66
## housing   -0.12601660 3.536553e-135
## loan      -0.06353381  1.618283e-35
## contact   -0.14106732 3.180910e-169
## day       -0.03603133  1.799483e-12
## month     -0.02207253  1.578693e-05
## duration   0.33852615  0.000000e+00
## campaign  -0.08095048  1.235553e-56
## pdays      0.14591875 4.788306e-181
## previous   0.16108524 1.064216e-220
## poutcome  -0.13726344 3.034367e-160
```

Comparant l'atribut 'y' amb la resta, veiem una relació amb housing, contact, pdays, previous, poutcome i sobretot **duration**, as a dir, la durada de la trucada.

Matriu de correlació visual (Correlació)

Amb la matriu de correlació veiem la relació de cada atribut amb els altres.

```
bank.clean.tmp <- bank.clean
for (colname in colnames(bank.clean.tmp)) {
  bank.clean.tmp[colname] <- lapply(bank.clean.tmp[colname],as.integer)
}
corr.mat <- cor(bank.clean.tmp, method = "spearman")
# visualize it
library(corrplot)
col <- colorRampPalette(c("#BB4444", "#EE9988", "#FFFFFF", "#77AADD",
"#4477AA"))
corrplot(corr.mat, method="color", col=col(200),
         addCoef.col = "black",
         tl.col="black", tl.srt=45,
         insig = "blank",
         number.cex=0.5)
```



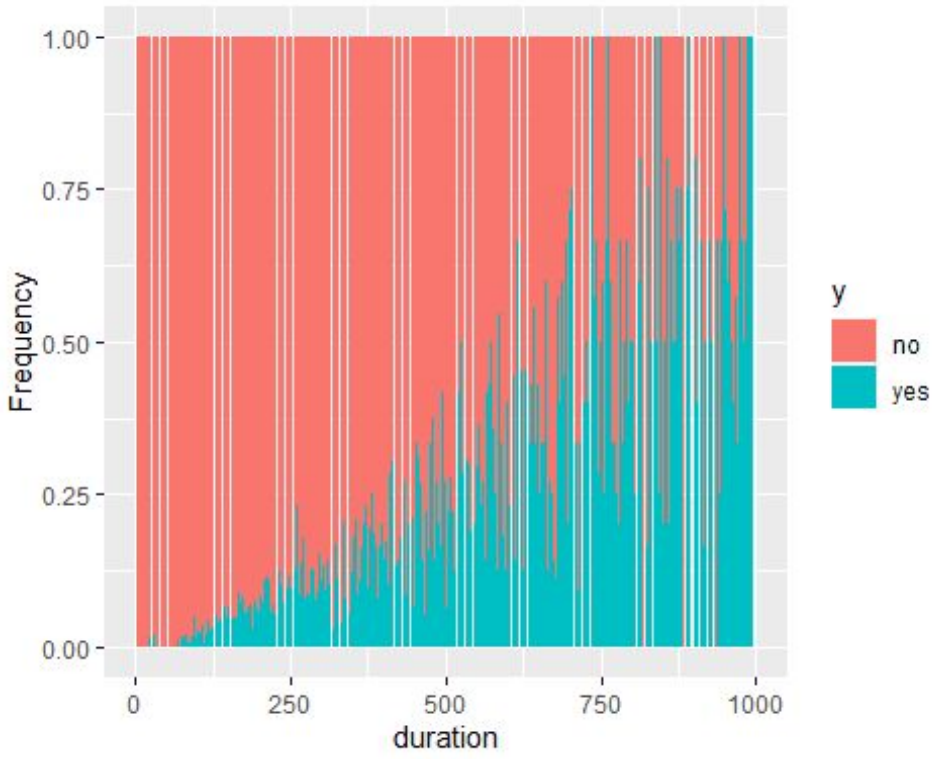
Apart de les relacions que hem comentat abans podem veure una força relació significativa entre:

- **marital - age**
- marital - education
- education - age
- housing - age

Hi han altres relacions fortes però que tenen a veure amb atributs de com s'han fet les campanyes de marketing anteriors (entre elles) i no a les dades dels clients.

Conversió de 'y' en funció de duration

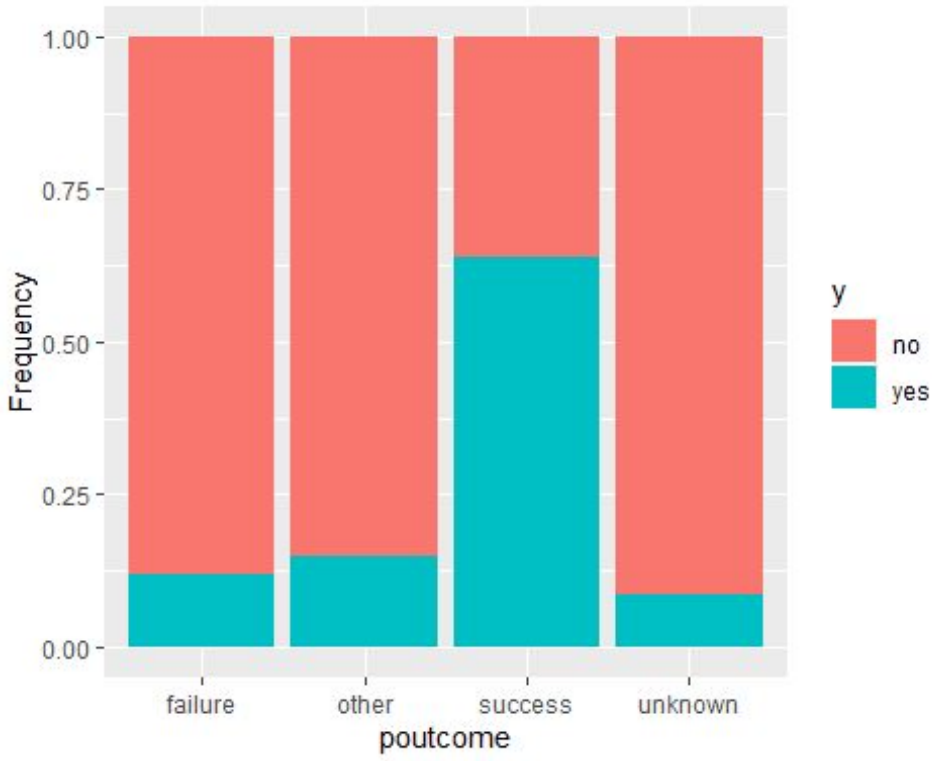
```
ggplot(data=bank.clean,aes(x=duration,fill=y)) + geom_bar(position="fill") +
ylab("Frequency") + scale_x_continuous(limits = c(0, 1000))
```



L'evidència sembla clara. Com més temps dura la trucada, més possibilitats de que el client contracti un dipòsit a termini.

Conversió de 'y' en funció de poutcome

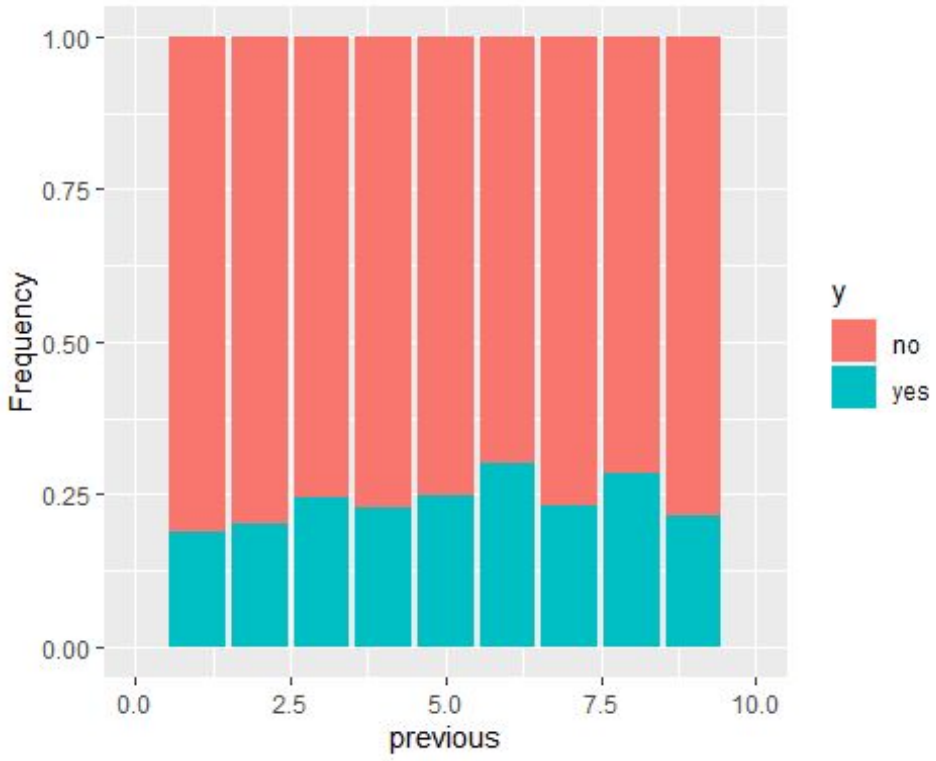
```
ggplot(data=bank.clean, aes(x= poutcome, fill = y)) +
geom_bar(position="fill") + ylab("Frequency")
```



Si en una campanya anterior havíen contractat un dipòsit tenen un percentatge molt més alt de possibilitats de contractar.

Conversió de 'y' en funció de previous

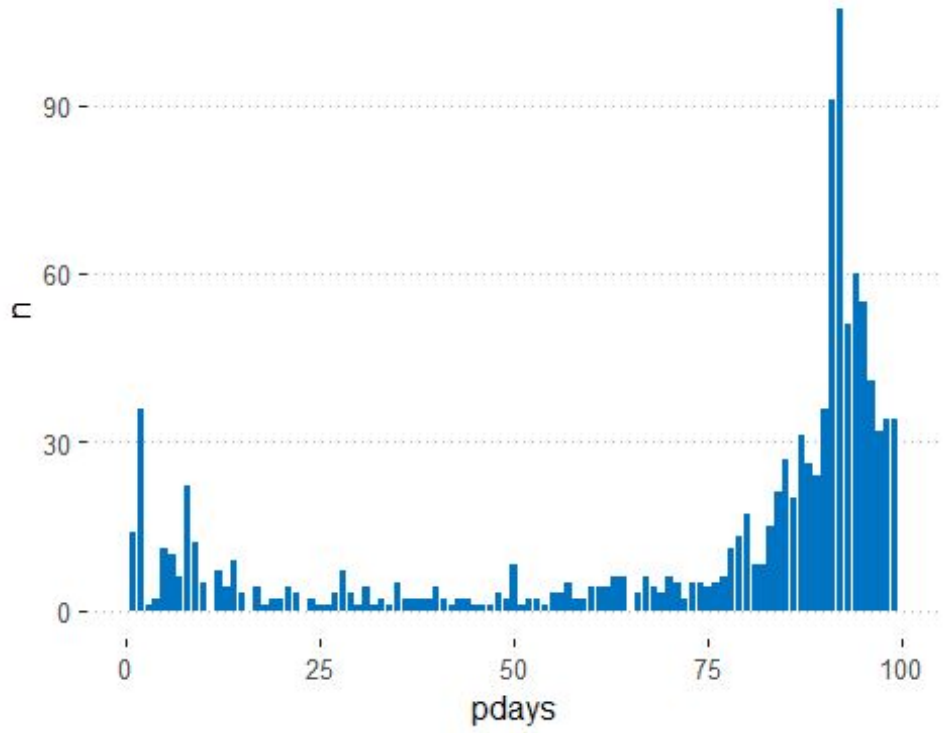
```
ggplot(data=bank.clean,aes(x=previous,fill=y)) + geom_bar(position="fill") +
ylab("Frequency") + scale_x_continuous(limits = c(0, 10))
```



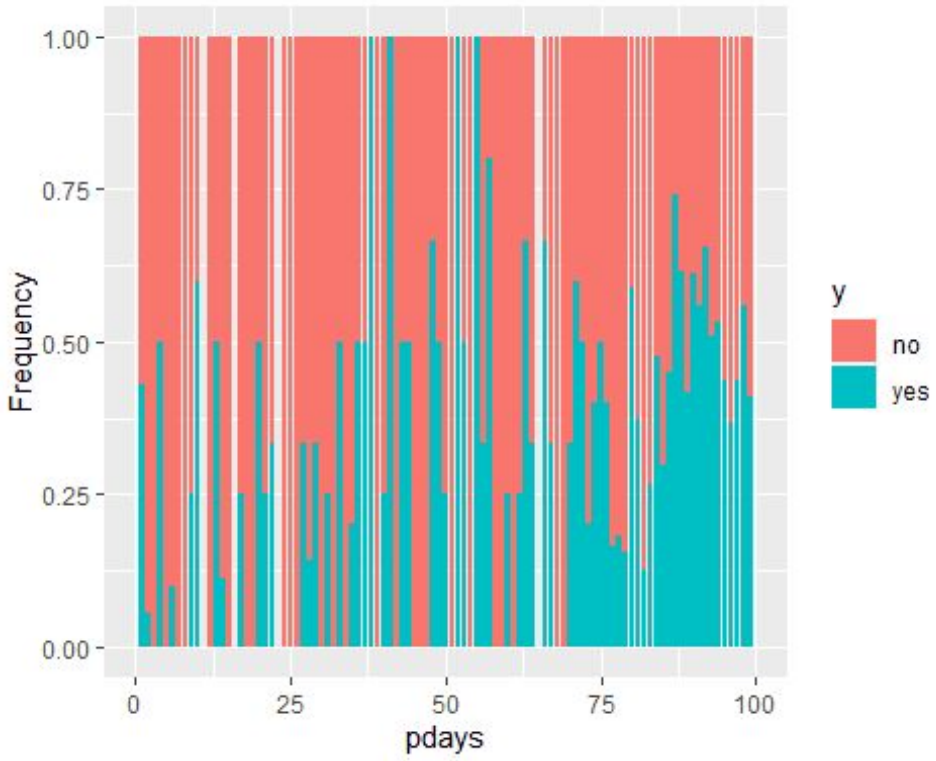
Amb el nombre de contactes previs, encara que la correlació ens digui que tenen una relació directe, no seria capaç de trobar un patró de cara a definir el nombre de trucades a realitzar.

Conversió de 'y' en funció de pdays

```
pdays.groups <- bank.clean %>% group_by(pdays) %>% dplyr::summarise(n = n())
ggplot(pdays.groups, aes(x = pdays, y = n)) + geom_bar(fill = "#0073C2FF",
stat = "identity")  + theme_pubclean() + scale_x_continuous(limits = c(0,
100))
```



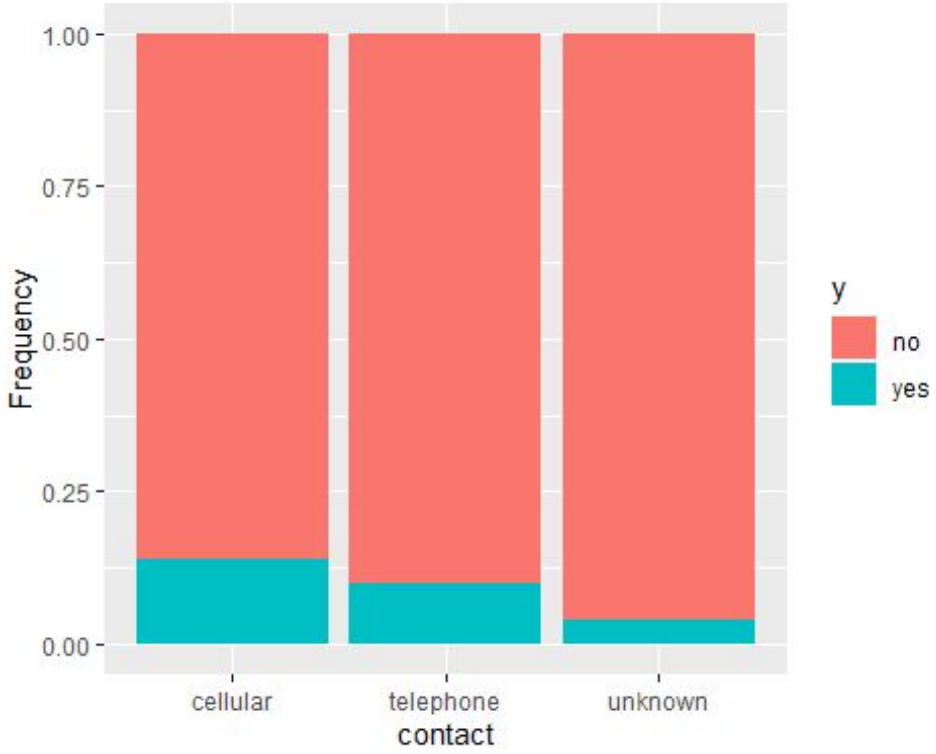

```
ggplot(data=bank.clean,aes(x=pdays,fill=y)) + geom_bar(position="fill") +
ylab("Frequency") + scale_x_continuous(limits = c(0, 100))
```



En aquest cas hem tret tant una gràfica numèrica absoluta com normalitzada. Si que sembla que tenim un patró que si fa poc que hem trucat tenim menys possibilitats de contractar. Necessitariem més mostra en les franjes de menys dies per a tenir una certesa.

Conversió de 'y' en funció de contact

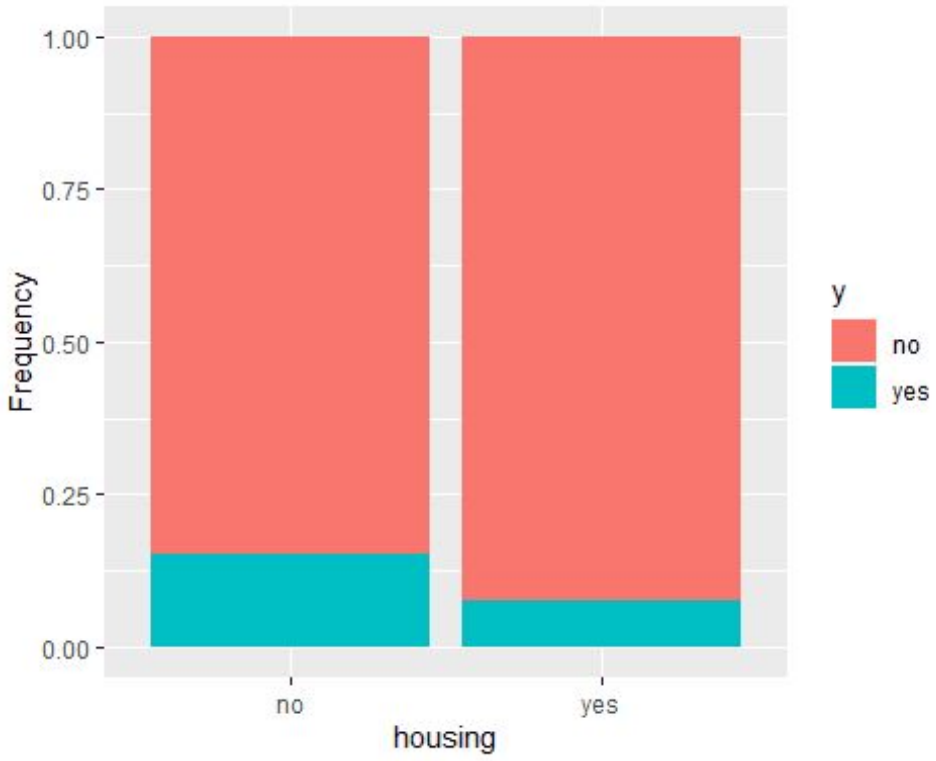
```
ggplot(data=bank.clean, aes(x= contact, fill = y)) +
geom_bar(position="fill") + ylab("Frequency")
```



La proporció del mòbil respecte al telèfon es millor i el pitjor són els unknown.

Conversió de 'y' en funció de housing

```
ggplot(data=bank.clean, aes(x= housing, fill = y)) +
geom_bar(position="fill") + ylab("Frequency")
```



Podem concluir que si tenen un prèstec d'habitatge serà més complicat que contractin un crèdit a termini.

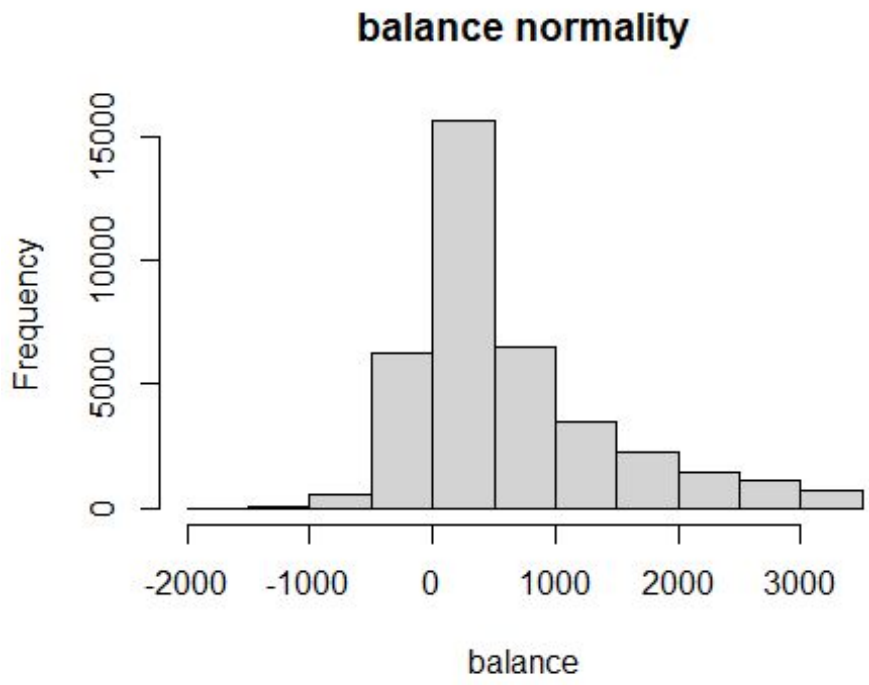
Es tenen més diners al banc si NO es té un crèdit per defecte? (Contrast d'hipòtesi)

Anem a realitzar un altre tipus de prova estadística, el contrast d'hipòtesi. Aquest contrast el farem sobre dues mostres per determinar si el fet de no tenir un crèdit per defecte (default = no) influeix alhora de tenir més diners al banc. Per a fer-ho, tenim una mostra amb els usuaris amb crèdit per defecte i una altre sense.

Comprovam la normalitat de la distribució del camp balance (ja sabiem d'abans que no segueix aquesta distribució). Amb el test Anderson-Darling (mostres grans) tornem a comprovar que no és una distribució normal, ja que p-value < 0.05

```
ad.test(bank.clean$balance)

##
##  Anderson-Darling normality test
##
## data:  bank.clean$balance
## A = 1877.5, p-value < 2.2e-16

hist(bank.clean$balance,
  main = "balance normality",
```

```
  xlab = "balance"
)
```



I ara amb Fligner-Killeen rebutjem la homogeneïtat en la variança.

```
fligner.test(balance ~ default, data = bank.clean)
```

```
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  balance by default
## Fligner-Killeen:med chi-squared = 256.46, df = 1, p-value < 2.2e-16
```

Com que les dades no segueixen una distribució normal i no tenim una homogeneïtat en les variancies, no aplicarem la t de Student. Fem servir el mètode de Wilcoxon i Mann-Whitney encara que **perdem potència estadísctica**.

Fem el contrast d'hipòtesi de si el subconjunt de dades sense crèdit bancari té un balance diferent amb els que si el tenen.

- H0 : els 2 grups son similars
- H1 : els 2 grups son diferents

```
wilcox.test(bank.default.yes$balance, bank.default.no$balance)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
```

```
## data:  bank.default.yes$balance and bank.default.no$balance
## W = 4430308, p-value < 2.2e-16
## alternative hypothesis: true location shift is not equal to 0
```

I podem concloure que rebutjem la hipòtesi nul·la amb el valor de p-value < 0.05 de que no hi han diferències significatives entre els balance que tenen un crèdit bancari (default) dels que no. Si que existeixen aquestes diferències.

Altre opció és amb el alternative less, i canviem la hipotesi a

- H0 : μ1 − μ2 = 0
- H1 : μ1 − μ2 < 0

on μ1 és la mitja de població que tè un crèdit i μ2 és la mitja que no en té. Fem servir una $\alpha = 0,05$.

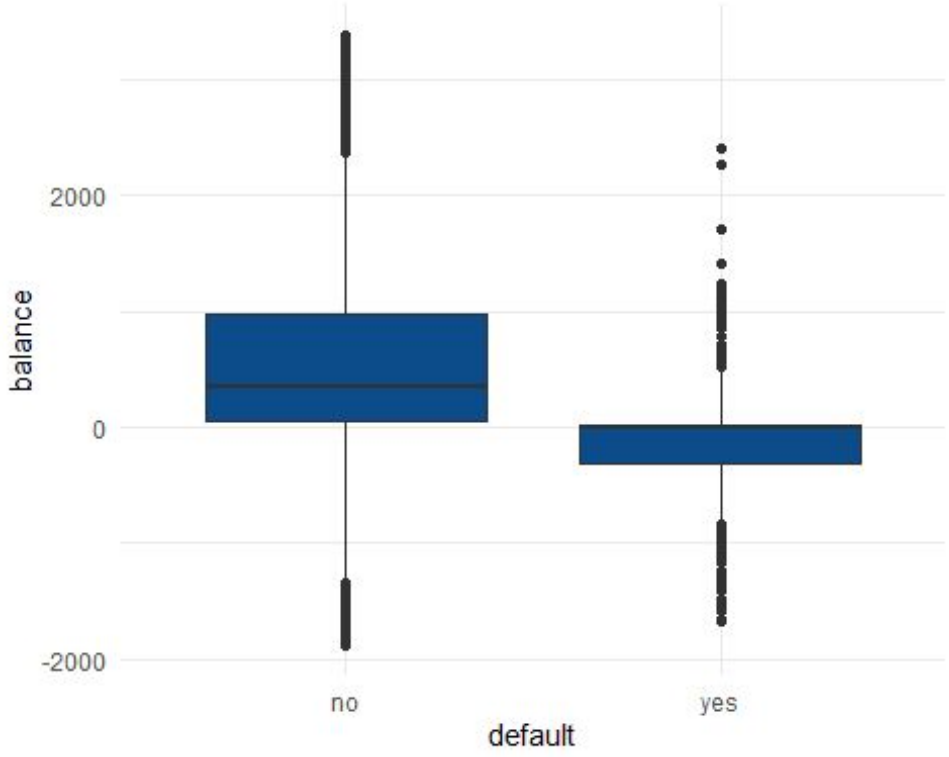
```
wilcox.test(bank.default.yes$balance - bank.default.no$balance, alternative = "less")
```

```
## Warning in bank.default.yes$balance - bank.default.no$balance: longitud de
## objeto mayor no es múltiplo de la longitud de uno menor

##
##  Wilcoxon signed rank test with continuity correction
##
## data:  bank.default.yes$balance - bank.default.no$balance
## V = 59757238, p-value < 2.2e-16
## alternative hypothesis: true location is less than 0
```

Ja que el valor del p-value és < 0.05 podem concloure que tenir un crèdit bancari pot indicar que és tenen menys diners de mitja anual al banc.

Si revisem visualment:

```
ggplot(bank.clean) +
  aes(x = default, y = balance) +
  geom_boxplot(fill = "#0c4c8a") +
  theme_minimal()
```



Comprovem que els balance son totalment superiors si NO tenen crèdit per defecte (default).

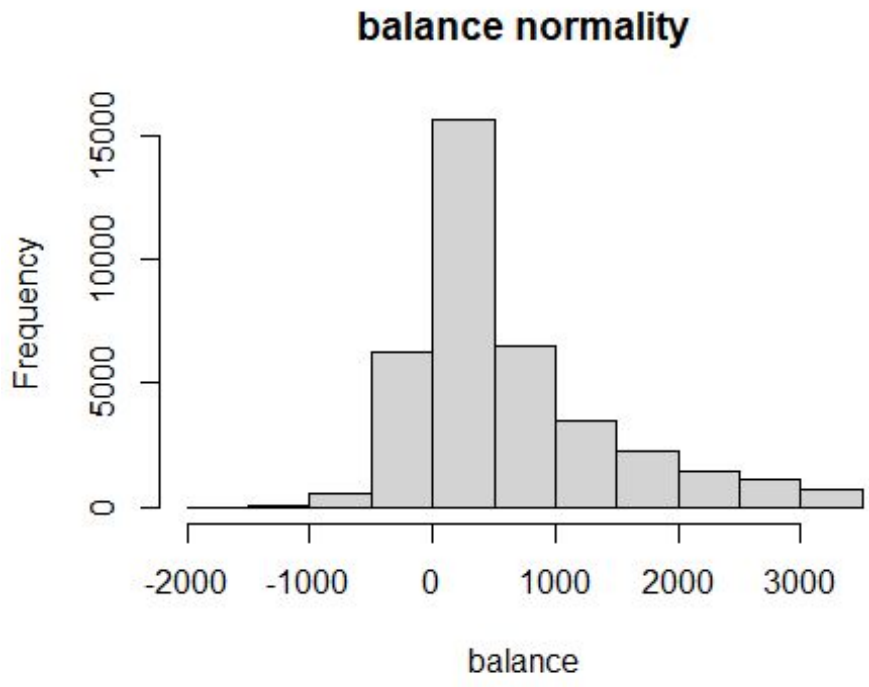
Es tenen més diners al banc si NO es té un prestec d'habitatge? (Contrast d'hipòtesi)

Plantejament similar al exercici anterior on ja sabem que el camp balance no seguiex una distribució normal.

```
ad.test(bank.clean$balance)
```

```
##
##  Anderson-Darling normality test
##
## data:  bank.clean$balance
## A = 1877.5, p-value < 2.2e-16
```

```
hist(bank.clean$balance,
  main = "balance normality",
  xlab = "balance"
)
```



I ara amb Fligner-Killeen rebutjem la homogeneïtat en la variança amb el camp housing.

```
fligner.test(balance ~ housing, data = bank.clean)
```

```
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  balance by housing
## Fligner-Killeen:med chi-squared = 54.159, df = 1, p-value = 1.849e-13
```

Altre cop fem servir Wilcoxon test per a fer el contrast d'hipòtesi de si el subconjunt de dades sense hipoteca té un balance diferent amb els que si tenen hipoteca.

- H0 : els 2 grups son similars
- H1 : els 2 grups son diferents

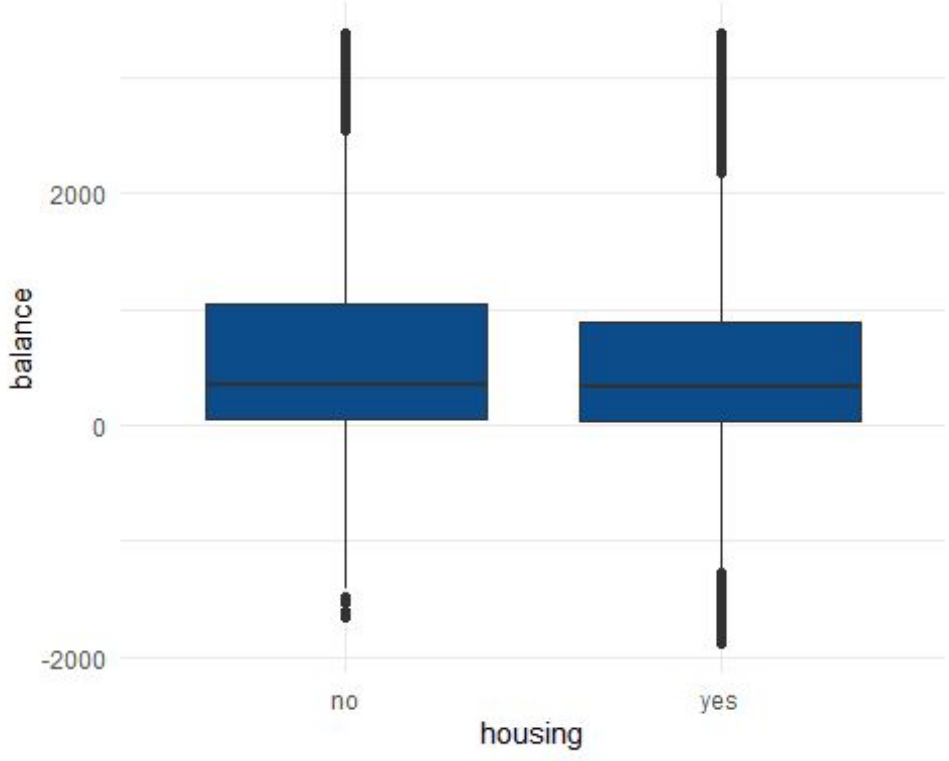
```
wilcox.test(bank.housing.yes$balance, bank.housing.no$balance)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  bank.housing.yes$balance and bank.housing.no$balance
## W = 170386078, p-value = 7.239e-15
## alternative hypothesis: true location shift is not equal to 0
```

I podem concloure que rebutjem la hipòtesi nul·la amb el valor de p-value < 0.05 de que no hi han diferències significatives entre els balance que tenen hipoteca (housing) i els que no en tenen.

Si revisem visualment:

```
ggplot(bank.clean) +
  aes(x = housing, y = balance) +
  geom_boxplot(fill = "#0c4c8a") +
  theme_minimal()
```



Els balance son relativament superiors si NO té hipoteca, encara que no es tan evident com en l'exemple anterior.

Es tenen més diners al banc si es tenen estudis primary i tertiary? (Contrast d'hipòtesi)

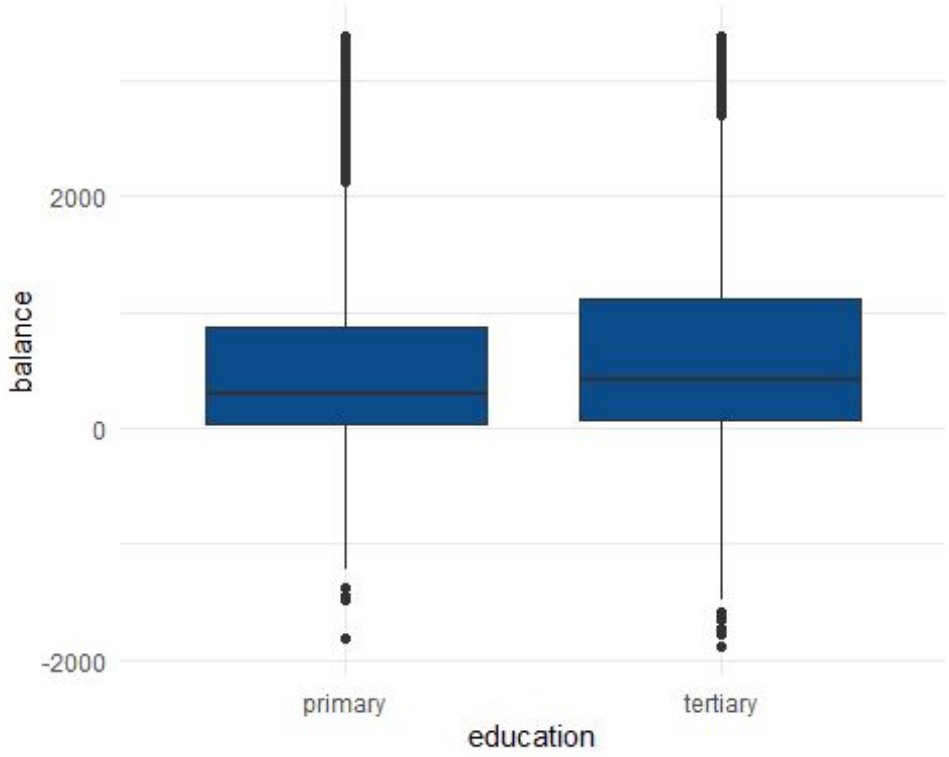
Igual que abans anem a comprovar si els sous son diferents segons els estudis i ja sabem que no compleixen els requeriments per aplicar una t de Student. Anem directament a aplicar Wilcoxon test.

```
wilcox.test(balance ~ education, data = bank.clean, subset = education %in%
c('primary', 'tertiary'))

##
##  Wilcoxon rank sum test with continuity correction
##
```

```
## data:  balance by education
## W = 30716003, p-value < 2.2e-16
## alternative hypothesis: true location shift is not equal to 0
```

Tornem a rebutjar la hipòtesi nul·la i deduim que les mostres tenen balance diferents si tenen estudis diferents.

```
ggplot(subset(bank.clean, education != "secondary")) +
  aes(x = education, y = balance) +
  geom_boxplot(fill = "#0c4c8a") +
  theme_minimal()
```



Predicció mitjançant regressió logística

És un tipus d'anàlisi de regressió que ens permet predir un resultat dicotòmic, a partir de variables independents. El que fa és estimar la probabilitat d'ocurrència d'una de les categories de sortida (y = yes , y = no) basant-se en una funció logística.

Per a fer-ho, partim les dades en un 70% per a entrenament i l'altre 30% per a test.

```
set.seed(22)
# realitzem la partició
partition <- createDataPartition(bank.clean$y, p = 0.7, list = FALSE)
bank.train <- bank.clean[partition, ]
bank.test <- bank.clean[-partition, ]

# executem l'entrenament 10 cops per validar el model a partir de la qualitat
```

```r
basada en el AIC
control <- trainControl(method = "cv", number = 10, classProbs = TRUE)

# entrenem el model i treiem els resultats
log.trained <- train(y ~., data = bank.train, method = "glm", trControl =
control)
summary(log.trained)
```

```
##
## Call:
## NULL
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -4.9397  -0.3580  -0.2449  -0.1497   3.5060
##
## Coefficients:
##                      Estimate Std. Error z value Pr(>|z|)
## (Intercept)        -2.514e+00  2.505e-01 -10.038  < 2e-16 ***
## age                -5.704e-03  3.109e-03  -1.835 0.066561 .
## `jobblue-collar`   -2.416e-01  9.480e-02  -2.549 0.010801 *
## jobentrepreneur    -3.521e-01  1.644e-01  -2.141 0.032281 *
## jobhousemaid       -5.018e-01  1.834e-01  -2.737 0.006209 **
## jobmanagement      -1.816e-01  9.786e-02  -1.855 0.063545 .
## jobretired          7.052e-02  1.426e-01   0.495 0.620793
## `jobself-employed` -1.328e-01  1.433e-01  -0.927 0.354181
## jobservices        -2.504e-01  1.100e-01  -2.276 0.022867 *
## jobstudent          5.374e-01  1.507e-01   3.567 0.000361 ***
## jobtechnician      -2.097e-01  9.050e-02  -2.317 0.020492 *
## jobunemployed      -4.389e-02  1.433e-01  -0.306 0.759419
## maritalmarried     -2.077e-01  7.988e-02  -2.600 0.009331 **
## maritalsingle       8.898e-03  9.005e-02   0.099 0.921289
## educationsecondary  2.529e-01  8.886e-02   2.846 0.004423 **
## educationtertiary   4.049e-01  1.046e-01   3.872 0.000108 ***
## defaultyes         -3.150e-02  2.194e-01  -0.144 0.885808
## balance             1.321e-04  2.830e-05   4.668 3.04e-06 ***
## housingyes         -7.060e-01  5.825e-02 -12.119  < 2e-16 ***
## loanyes            -3.858e-01  7.652e-02  -5.042 4.60e-07 ***
## contacttelephone   -1.967e-01  1.097e-01  -1.793 0.073043 .
## contactunknown     -1.500e+00  9.465e-02 -15.849  < 2e-16 ***
## day                 1.060e-02  3.386e-03   3.129 0.001752 **
## monthaug           -5.829e-01  1.075e-01  -5.422 5.88e-08 ***
## monthdec            7.793e-01  2.633e-01   2.960 0.003075 **
## monthfeb           -1.259e-01  1.220e-01  -1.032 0.301943
## monthjan           -1.286e+00  1.683e-01  -7.643 2.13e-14 ***
## monthjul           -7.023e-01  1.041e-01  -6.748 1.49e-11 ***
## monthjun            6.215e-01  1.245e-01   4.993 5.95e-07 ***
## monthmar            1.873e+00  1.679e-01  11.158  < 2e-16 ***
## monthmay           -3.208e-01  9.703e-02  -3.306 0.000945 ***
## monthnov           -8.950e-01  1.193e-01  -7.500 6.36e-14 ***
```

```
## monthoct             8.903e-01  1.497e-01   5.947 2.73e-09 ***
## monthsep             1.126e+00  1.679e-01   6.708 1.98e-11 ***
## duration             4.239e-03  8.517e-05  49.767  < 2e-16 ***
## campaign            -9.693e-02  1.377e-02  -7.038 1.95e-12 ***
## pdays               -1.066e-05  3.963e-04  -0.027 0.978545
## previous             4.057e-02  1.228e-02   3.303 0.000957 ***
## poutcomeother        1.021e-01  1.201e-01   0.850 0.395068
## poutcomesuccess      2.185e+00  1.110e-01  19.678  < 2e-16 ***
## poutcomeunknown     -1.356e-01  1.282e-01  -1.058 0.289954
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 18329  on 26777  degrees of freedom
## Residual deviance: 12068  on 26737  degrees of freedom
## AIC: 12150
##
## Number of Fisher Scoring iterations: 6
```

Podem veure la importància de cada una de les variables en la funció logarítmica que s'ha generat. Per cada valor de cada categoria ha generat un atribut propi per a la funció.

```
log_imp <- varImp(log.trained, scale = FALSE, competes = FALSE)
log_imp
```

```
## glm variable importance
##
##   only 20 most important variables shown (out of 40)
##
##                    Overall
## duration           49.767
## poutcomesuccess    19.678
## contactunknown     15.849
## housingyes         12.119
## monthmar           11.158
## monthjan            7.643
## monthnov            7.500
## campaign            7.038
## monthjul            6.748
## monthsep            6.708
## monthoct            5.947
## monthaug            5.422
## loanyes             5.042
## monthjun            4.993
## balance             4.668
## educationtertiary   3.872
## jobstudent          3.567
## monthmay            3.306
```

```
## previous            3.303
## day                 3.129
```

Com ja havíem vist en estudis anteriors la variable que te més pes és 'duration', seguit de 'poutcome' quan es success. El tercer és 'contact' quan és unknown **(d'aquí la importancia de deixar de vegades els null o valors desconeguts)**.

I ara realitzen la predicció i treiem una matriu de confusió per a veure la qualitat de la predicció del model.

```
pred_log <- predict(log.trained, newdata = bank.test)
confusionMatrix(pred_log, bank.test$y)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    no   yes
##        no  10030   814
##        yes   207   424
##
##                Accuracy : 0.911
##                  95% CI : (0.9057, 0.9162)
##     No Information Rate : 0.8921
##     P-Value [Acc > NIR] : 1.104e-11
##
##                   Kappa : 0.4108
##
##  Mcnemar's Test P-Value : < 2.2e-16
##
##             Sensitivity : 0.9798
##             Specificity : 0.3425
##          Pos Pred Value : 0.9249
##          Neg Pred Value : 0.6719
##              Prevalence : 0.8921
##          Detection Rate : 0.8741
##    Detection Prevalence : 0.9450
##       Balanced Accuracy : 0.6611
##
##        'Positive' Class : no
##
```

Tenim una molt bona predicció > 91%, però prediu molt millor els 'no', quan volem saber si un client no contractarà un crèdit bancari que els 'yes'. Tenim un alt nombre de falsos positius.

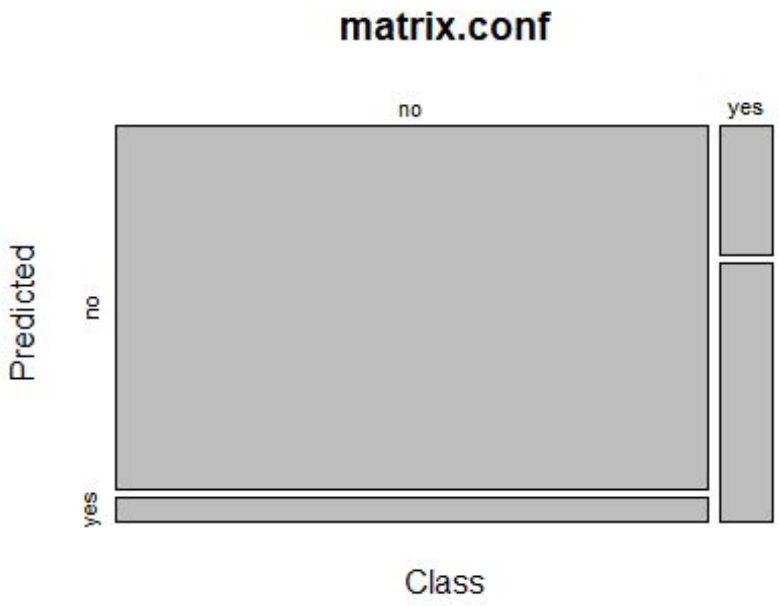
Predicció mitjançant classificació (Random Forest)

El que anem a fer en aquest apartat és una predicció mitjançant un algoritme de classificació. En tenim de diferents tipus, i en aquest cas farem servir Random Forest.

Un Random Forest és un conjunt d'arbres de decisió combinats amb bagging. A l'usar bagging, el que en realitat està passant, és que diferents arbres veuen diferents porcions de les dades. Cap arbre veu totes les dades d'entrenament. Això fa que cada arbre s'entreni amb diferents mostres de dades per a un mateix problema. D'aquesta manera, al combinar els seus resultats, uns errors es compensen amb altres i tenim una predicció que generalitza millor.

Per a realitzar l'examen dividim el conjunt de dades en un 70% d'entrenament i l'altre 30% per a fer proves.

```
set.seed(22)

# separem el 70% de conjunt d'entrenament del 30% per a avaluar
random.data <- sample(1:nrow(bank.clean), 0.70 * nrow(bank.clean))

train <- bank.clean[random.data,]
test <- bank.clean[-random.data,]

# en les dades d'entrenament, separem les dades per a generar l'arbre de la
dada objectiu (target)
train.x <- train[,1:16]
train.y <- train[,17]

# en les dades de test, separem les dades per a generar l'arbre de la dada
objectiu (target)
test.x <- test[,1:16]
test.y <- test[,17]

train_rf <- randomForest(y~.,data = train, ntree = 50)

predicted.model <- predict(train_rf,test.x)


print(sprintf("La precissió de l'arbre és del: %.2f
%%",100*sum(predicted.model == test.y) / length(predicted.model)))

## [1] "La precissió de l'arbre és del: 91.46 %"

matrix.conf <- table(Class=predicted.model,Predicted=test.y)
percent.correct <- 100 * sum(diag(matrix.conf)) / sum(matrix.conf)
print(sprintf("L'error de classificació és: %.2f %%",100 - percent.correct))

## [1] "L'error de classificació és: 8.54 %"

mosaicplot(matrix.conf)
```




```
confusionMatrix(predicted.model,test.y, dnn = c("Prediction"))
```

```
## Confusion Matrix and Statistics
##
##           NA
## Prediction   no  yes
##        no  9883  670
##        yes  310  613
##
##                Accuracy : 0.9146
##                  95% CI : (0.9093, 0.9197)
##     No Information Rate : 0.8882
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.5099
##
##  Mcnemar's Test P-Value : < 2.2e-16
##
##             Sensitivity : 0.9696
##             Specificity : 0.4778
##          Pos Pred Value : 0.9365
##          Neg Pred Value : 0.6641
##              Prevalence : 0.8882
##          Detection Rate : 0.8612
##    Detection Prevalence : 0.9196
##       Balanced Accuracy : 0.7237
```

```
##
##        'Positive' Class : no
##
```

- Accuracy: 0.9108
- Prediction ‘no’: 0.9263
- Prediction ‘yes’: 0.6496

Tenim altre cop molt bona predicció > 90% però i predicció del ‘no’ també és força baixa 0.6496. Tenim altre cop força falsos positius.

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

Com hem vist durant tot l'exercici hem realitzat diverses proves tant de anàlisi exploratori, neteja de les dades, proves estadístiques i finalment proves amb algoritmes supervisats.

El que hem analitzat principalment ha estat amb la correlació, quins atributs tenen més impacte en els atributs 'balance' (saldo mitjà anual) i 'y' (si contracten dipòsit a termini o no).

A més hem realitzat una regressió logarítmica per a predir el probable comportament dels usuaris als quals se'ls farà una campanya de marketing. Entre altres variables podem dir que el temps a la trucada és rellevant, però això també és un factor que no es pot saber a priori. Altres factors important alhora de decidir per trucar a uns clients o uns altres seria: si ja han contractat prèviament un crèdit o si tenen un préstec d'habitatge.

També amb el contrast d'hipòtesi hem comprovat que no tenir crèdit per defecte, no tenir hipoteca o tenir més estudis influeix en la quantitat de diners de mitja anual que es té en el banc.

En quant a la neteja de dades, remarcar que incloure uns valors nulls com a categoria pròpia ens ha ajudat a tenir millors prediccions en els algoritmes de classificació.