Construcció de models multivariants

Sergi Cozar Badia

2023-01-20

Exercici 1:

Carreguem les dades i les visualitzem:

```
socsupport <- read.csv("socsupport.csv", row.names = 1)
head(socsupport)
```

Seleccionem les columnes:

```
dat0 = select(socsupport, c(10,11,12,13,14,15,16,17,18,19))
```

Fem un sapply per veure si hi ha valors NaN:

```
sapply(dat0, function(x) sum(is.na(x)))
```

Eliminem les files que tinguin valors NaN, posteriorment comprovem que ja no n'hi hagi:

```
dat01 <- na.omit(dat0)
sapply(dat01, function(x) sum(is.na(x)))
```

```
## emotionalsat     tangible  tangiblesat       affect    affectsat          psi
##            0            0            0            0            0            0
##       psisat     esupport     psupport   supsources
##            0            0            0            0
```

Observem quantes files hem eliminat:

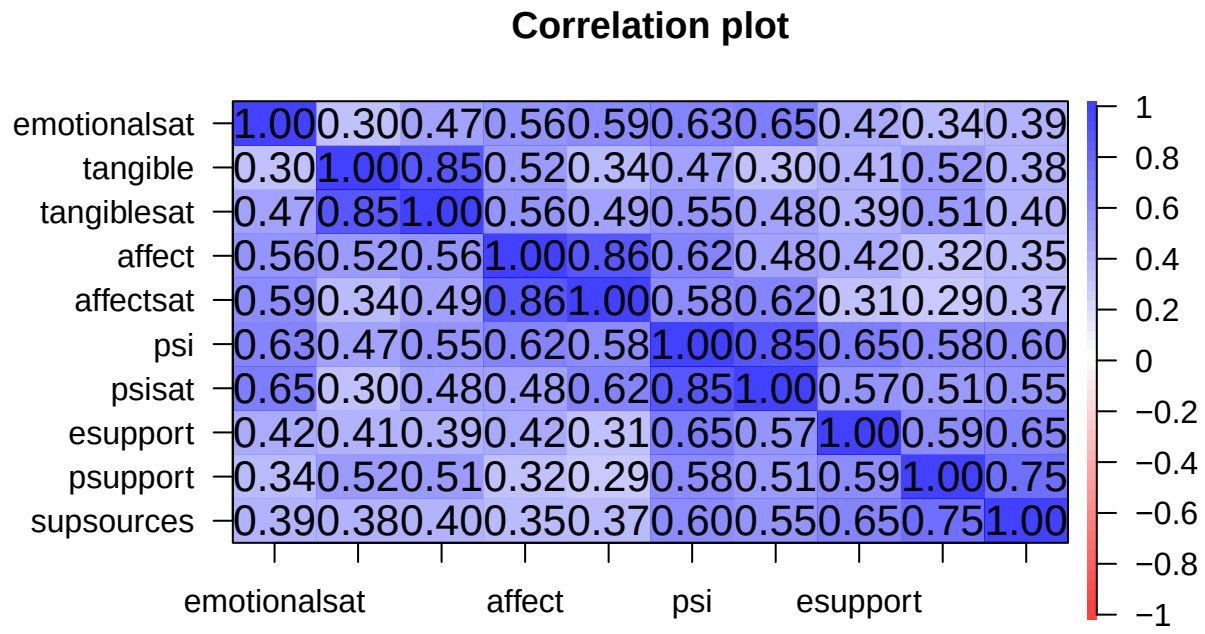
```
dim(dat0) # Abans d'eliminar dades.
dim(dat01) # Després d'eliminar dades.
```

```
## [1] 95 10
## [1] 90 10
```

Observem que només hem eliminat 5 files, i que, per tant, no ens afectarà molt.

Observem la correlació:

```
R = cor(dat01)
cor.plot(R)
```



Quan tenim correlacions tan altes, com en aquest cas, algunes de 0.85, 0.86, 0.75, etc. entrem en el problema de la correlació entre les variables independents, una solució seria treure aquelles que estan molt interrelacionades o, per una altra banda, aplicar PCA i reduir la dimensió.

Matriu de correlació diferent de la matriu identitat:

```
cortest.bartlett(cor(dat01), n=dim(dat01))
```

```
## $chisq
## [1] 748.45836  42.64301
##
## $p.value
## [1] 2.788204e-128  5.723134e-01
##
## $df
## [1] 45
```

Observem que el pvalor és menor a una alfa de l'1%, per tant, hi ha correlació entre variables.

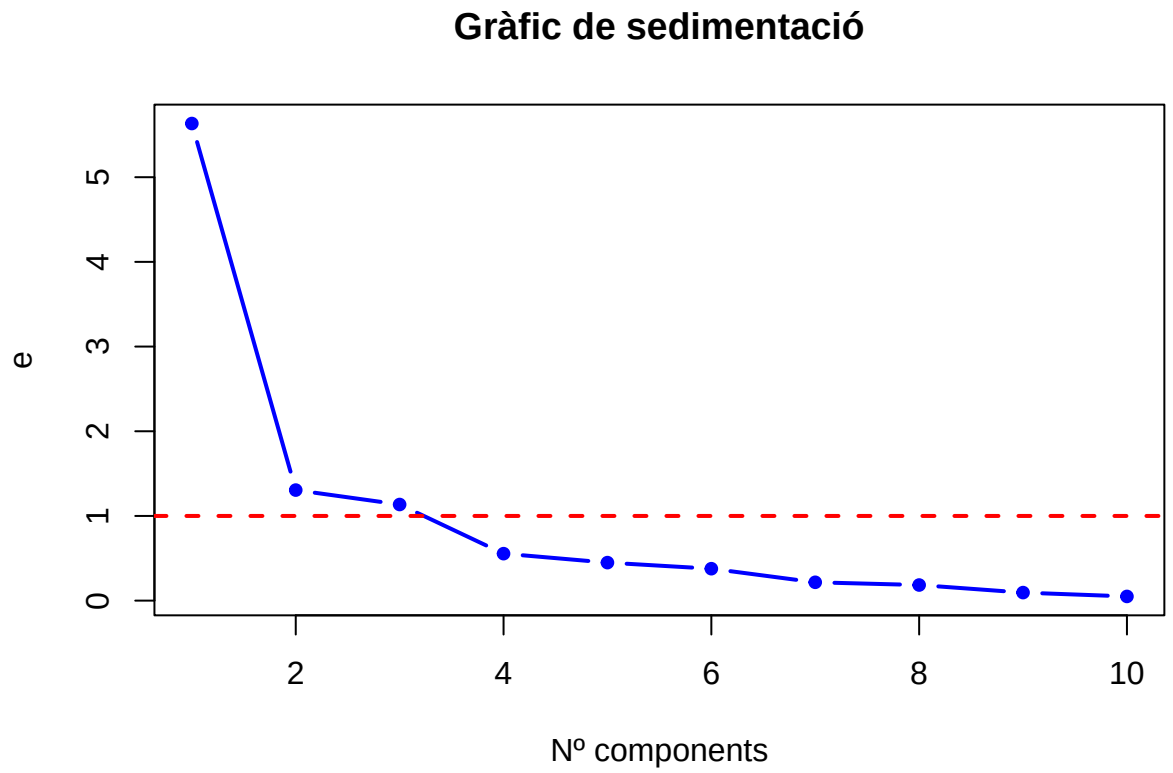
Normalitat multivariada:

```
mshapiro.test(t(dat01))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  Z
## W = 0.84822, p-value = 3.663e-08
```

Nombre de components:

```
e = eigen(R)$val
plot(e, type="b", pch=20, col="blue", lwd=2,
     main="Gràfic de sedimentació", xlab="Nº components")
abline(h=1, lwd=2, col="red", lty = 2)
```



Observem que a partir de la quarta component comença a sedimentar. Veiem que a partir de la tercera pca, obtenim un 80% de la variància.

Generem la nova matriu de dades:

```
facto = principal(dat01, nfactors = 3, rotate = "none")
facto$loadings
```

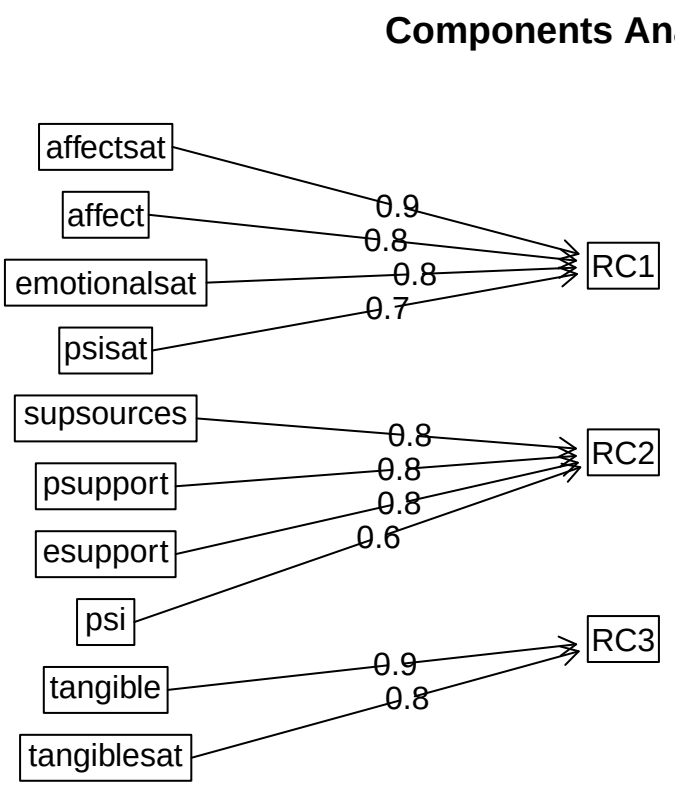
```
##
## Loadings:
##              PC1    PC2    PC3
## emotionalsat  0.717 -0.309 -0.248
## tangible      0.670  0.101  0.692
## tangiblesat   0.753         0.554
## affect        0.759 -0.483  0.122
## affectsat     0.731 -0.554
## psi           0.880        -0.216
## psisat        0.813        -0.383
## esupport      0.721  0.373 -0.191
## psupport      0.718  0.539
## supsources    0.723  0.474 -0.184
##
##                   PC1   PC2   PC3
## SS loadings    5.634 1.305 1.135
## Proportion Var 0.563 0.131 0.114
## Cumulative Var 0.563 0.694 0.807
```

```
facto.rota=principal(dat01, nfactors = 3, rotate = "varimax")
facto.rota$loadings #
```

```
##
## Loadings:
##              RC1   RC2   RC3
## emotionalsat 0.753 0.309
## tangible     0.152 0.260 0.920
## tangiblesat  0.348 0.257 0.830
## affect       0.791       0.435
## affectsat    0.885       0.238
## psi          0.624 0.626 0.198
## psisat       0.681 0.590
## esupport     0.258 0.777 0.159
## psupport           0.813 0.379
## supsources   0.186 0.847 0.169
##
##                   RC1   RC2   RC3
## SS loadings    3.080 2.967 2.027
## Proportion Var 0.308 0.297 0.203
## Cumulative Var 0.308 0.605 0.807
```

Veiem com afecta cada variable a les components

```
fa.diagram(facto.rota)
```




```
facto.rota$communality
```

```
## emotionalsat     tangible  tangiblesat       affect    affectsat          psi
##    0.6710454    0.9377534    0.8754351    0.8236304    0.8483804    0.8210370
##       psisat     esupport     psupport   supsources
##    0.8116106    0.6954603    0.8087529    0.7808154
```

Observem que les variables queden ordenats per ordre d'importància, en la PR1, són aquelles que el seu valor supera el 0,7, Pr2 és la segona i Pr3 la tercera, per tant, les variables que tenen més pes queden agrupades dins de Pr1.

Comuns:

```
facto.rota$communality
```

```
## emotionalsat     tangible  tangiblesat       affect    affectsat          psi
##    0.6710454    0.9377534    0.8754351    0.8236304    0.8483804    0.8210370
##       psisat     esupport     psupport   supsources
##    0.8116106    0.6954603    0.8087529    0.7808154
```

Puntuacions:

```
head(facto.rota$scores)
```

```
##          RC1         RC2        RC3
## 1  0.8841881  0.04939991  0.0989084
## 2 -1.3099128 -0.35923818 -1.2503519
## 3  0.2029895  0.67635050 -0.1056233
## 4 -1.0815869  1.63915616  0.4712860
## 5 -0.4473746 -0.49181642 -1.2950867
## 6 -0.3523818  0.45296026 -0.1927572
```

Exercici 2:

Llegim i visualitzem les dades:

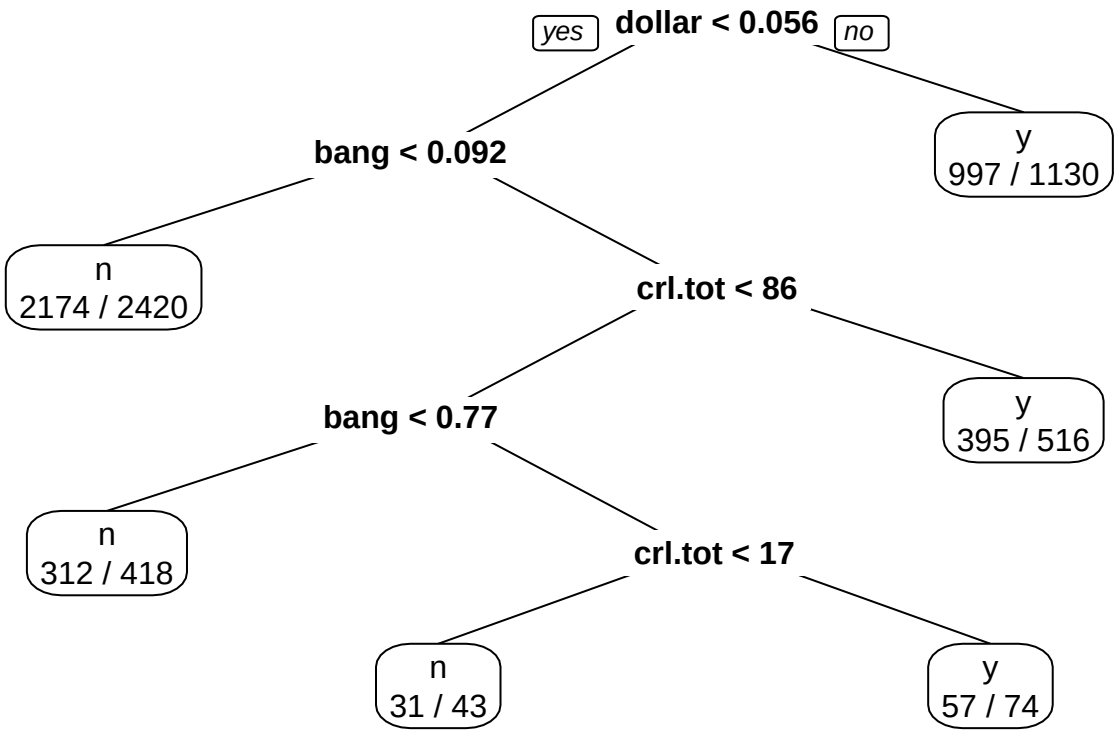
```
dat1 <- read.csv("spam7.csv", row.names = 1)
head(dat1)

summary(dat1)
```

Fem un arbre amb totes les variables:

```
spam.rpart <- rpart(yesno ~ crl.tot +
                      dollar +
                      bang +
                      money +
                      n000 +
                      make,
                    data=dat1)

prp(spam.rpart, extra=2)
```



Fem una taula de sensibilitat i especificitat:

```
tab1 <- table(predict(spam.rpart, type = "class"), dat1$yesno)
diag(tab1) / colSums( tab1 )
```

```
##         n         y
## 0.9027977 0.7992278
```

```
sum(diag(tab1)) / sum(tab1)
```

```
## [1] 0.8619865
```

Observem que la sensibilitat, o el bé que detecti els casos que no són spam (n), veiem que és de 0.90. En el cas de la (y), és el bé que detecta els casos que si són spam, i veiem que és de 0,79. Finalment, ens dona "l'acuraci" que és de 0.86, és a dir, lo bé que encerta en general.

Fem el model logístic:

Abans de res, és necessari que "si i no", siguin 1 o 0, per tant, efectuem el canvi:

```
table(dat1$yesno)
dat1$yn <- 1*( dat1$yesno == "y" )
head(dat1)

spam.logis <- glm(yn ~ crl.tot +
                    dollar +
                    bang +
                    money +
                    n000 +
                    make,
                  data = dat1,
                  family = binomial)
```

```
## Warning: glm.fit: algorithm did not converge
```

```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```
##
##    n    y
## 2788 1813
##   crl.tot dollar  bang money n000 make yesno yn
## 1     278  0.000 0.778  0.00 0.00 0.00     y  1
## 2    1028  0.180 0.372  0.43 0.43 0.21     y  1
## 3    2259  0.184 0.276  0.06 1.16 0.06     y  1
## 4     191  0.000 0.137  0.00 0.00 0.00     y  1
## 5     191  0.000 0.135  0.00 0.00 0.00     y  1
## 6      54  0.000 0.000  0.00 0.00 0.00     y  1
```

```
summary(spam.logis)
```

Observem els resultats, i podem deduir que com més grans siguin els números de la columna "Estimate" relacionada amb les variables "ctr.tot, dollar, bang, money i n000", les probabilitats que siguin spam augmenten. Veiem que "make" no surt marcada, i per tant es podria considerar treure-la del model.

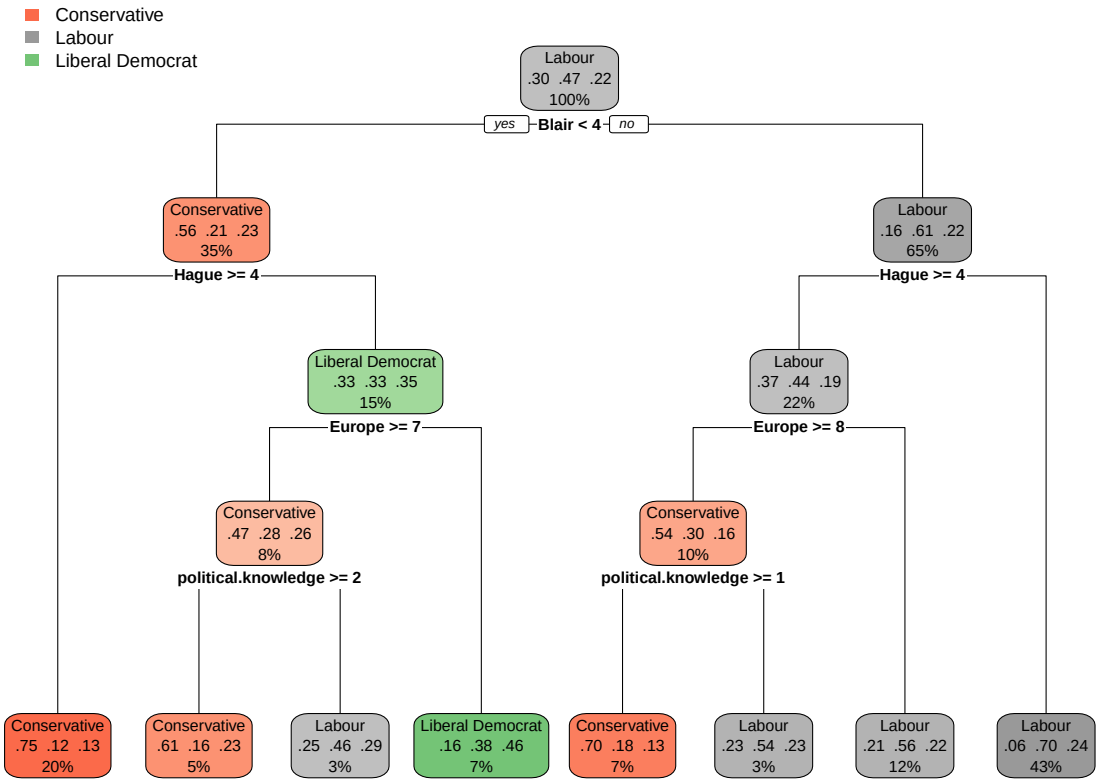
Exercici 3:

Llegim i visualitzem les dades:

```
dat3 <- read.csv("BEPS.csv", row.names = 1)
head(dat3)
summary(dat3)
table(dat3$vote)
```

Fem el dendrograma de les dades:

```
fit <- rpart(vote~., data = dat3, method = 'class',)
rpart.plot(fit)
```



Hem fet un model classificador/arbre de clarificació el qual surt exposat en el dendrograma. Quan és de color roig, identifica que la predicció de les dades d'aquests individus amb aquests perfils seran que votarà conservadors.

Observem que en el cas dels votants conservadors, per la part esquerra del dendrograma, els votants conservadors consideren el "Hague" si és més gran o igual que 4. En el cas de "political knowledge" el consideren si és més gran o igual que 2.

Veiem que en el cas dels votants laboristes, per la part dreta del dendrograma, els votants conservadors consideren el "political knowledge" si és superior o igual que 1.

Exercici 4:

Llegim i visualitzem les dades:

```
dat4 <- read.csv("FirstYearGPA.csv", row.names = 1)
head(dat4)
```

```
summary(dat4)
```

```
##       GPA             HSGPA            SATV            SATM
##  Min.   :1.930   Min.   :2.340   Min.   :260.0   Min.   :430.0
##  1st Qu.:2.745   1st Qu.:3.170   1st Qu.:565.0   1st Qu.:580.0
##  Median :3.150   Median :3.500   Median :610.0   Median :640.0
##  Mean   :3.096   Mean   :3.453   Mean   :605.1   Mean   :634.3
##  3rd Qu.:3.480   3rd Qu.:3.760   3rd Qu.:670.0   3rd Qu.:690.0
##  Max.   :4.150   Max.   :4.000   Max.   :740.0   Max.   :800.0
##       Male              HU              SS            FirstGen
##  Min.   :0.0000   Min.   : 0.00   Min.   : 0.000   Min.   :0.0000
##  1st Qu.:0.0000   1st Qu.: 8.00   1st Qu.: 3.000   1st Qu.:0.0000
##  Median :0.0000   Median :13.00   Median : 6.000   Median :0.0000
##  Mean   :0.4658   Mean   :13.11   Mean   : 7.249   Mean   :0.1142
##  3rd Qu.:1.0000   3rd Qu.:17.00   3rd Qu.:11.000   3rd Qu.:0.0000
##  Max.   :1.0000   Max.   :40.00   Max.   :21.000   Max.   :1.0000
##      White       CollegeBound
##  Min.   :0.00   Min.   :0.0000
##  1st Qu.:1.00   1st Qu.:1.0000
##  Median :1.00   Median :1.0000
##  Mean   :0.79   Mean   :0.9224
##  3rd Qu.:1.00   3rd Qu.:1.0000
##  Max.   :1.00   Max.   :1.0000
```

Observem, després de fer el summary, que el GPA no té molt de sentit, ja que dona per sobre de 4. Realment les notes del GPA van de 0 a 4.

Eliminem dades que no pertoquen, en aquest cas, eliminarem només les dades que el GPA sigui superior a 4:

```
dat4_2 <- subset(dat4, GPA <= 4)
summary(dat4_2)
```

```
##       GPA             HSGPA            SATV            SATM
##  Min.   :1.930   Min.   :2.340   Min.   :260.0   Min.   :430.0
##  1st Qu.:2.743   1st Qu.:3.165   1st Qu.:562.5   1st Qu.:580.0
##  Median :3.145   Median :3.500   Median :610.0   Median :640.0
##  Mean   :3.091   Mean   :3.450   Mean   :604.4   Mean   :633.9
##  3rd Qu.:3.475   3rd Qu.:3.757   3rd Qu.:670.0   3rd Qu.:690.0
##  Max.   :4.000   Max.   :4.000   Max.   :740.0   Max.   :800.0
##       Male              HU              SS            FirstGen
##  Min.   :0.0000   Min.   : 0.00   Min.   : 0.000   Min.   :0.0000
##  1st Qu.:0.0000   1st Qu.: 8.00   1st Qu.: 3.000   1st Qu.:0.0000
##  Median :0.0000   Median :13.00   Median : 6.000   Median :0.0000
##  Mean   :0.4679   Mean   :13.13   Mean   : 7.268   Mean   :0.1147
##  3rd Qu.:1.0000   3rd Qu.:17.00   3rd Qu.:11.000   3rd Qu.:0.0000
##  Max.   :1.0000   Max.   :40.00   Max.   :21.000   Max.   :1.0000
##      White        CollegeBound
##  Min.   :0.000   Min.   :0.000
##  1st Qu.:1.000   1st Qu.:1.000
##  Median :1.000   Median :1.000
##  Mean   :0.789   Mean   :0.922
##  3rd Qu.:1.000   3rd Qu.:1.000
##  Max.   :1.000   Max.   :1.000
```

Observem que ara, efectivament, el GPA no està per sobre de 4.

Observem quantes dades hem eliminat:

```
dim(dat4) # Abans d'eliminar dades.
```

```
## [1] 219  10
```

```
dim(dat4_2) # Després d'eliminar dades.
```

```
## [1] 218  10
```

Ens adonem que només hem eliminat un cas, per tant, podem pensar que era una dada equivocada. També és molt positiu, ja que en estar eliminant només una dada i no moltes, no ens afectarà molt.

Fem el model, en tenir les dades contínues, fem una regressió estàndard:

```
fit4 <- glm(GPA ~ . , data = dat4_2)
summary(fit4)
```

```
##
## Call:
## glm(formula = GPA ~ ., data = dat4_2)
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)   6.022e-01  3.480e-01   1.730  0.08509 .
## HSGPA         4.812e-01  7.422e-02   6.483 6.43e-10 ***
## SATV          5.404e-04  3.923e-04   1.377  0.16984
## SATM          5.568e-05  4.415e-04   0.126  0.89977
## Male          5.757e-02  5.678e-02   1.014  0.31176
## HU            1.682e-02  3.954e-03   4.253 3.18e-05 ***
## SS            8.124e-03  5.535e-03   1.468  0.14364
## FirstGen     -7.032e-02  8.810e-02  -0.798  0.42564
## White         1.956e-01  6.949e-02   2.815  0.00535 **
## CollegeBound  1.523e-02  9.962e-02   0.153  0.87866
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for gaussian family taken to be 0.1447558)
##
##     Null deviance: 46.118  on 217  degrees of freedom
## Residual deviance: 30.109  on 208  degrees of freedom
## AIC: 209.09
##
## Number of Fisher Scoring iterations: 2
```

Observem que les variables més significatives són HSGPA (Notes de l'institut), HU (quantitat d'hores que s'ha fet humanitats) i white. M'ha sorprès negativament que surti t'han destacat la variable "white", et fa pensar sobre els privilegis que es poden tenir a EEUU, depenent de la teva ètnia.

En canvi, m'ha sorprès molt positivament que no sigui significatiu el sexe (que no siguis sempre significatiu, no vol dir que sigui dolent), això vol dir, que les notes GPA no es veuen afectades per aquesta raó.

També és sorprenent que no siguin significatives algunes variables com sATV (Expressió oral i escrita) o SATM (Matemàtiques), et dona a penar que l'itinerari no està tan marcat com aquí i que influeix molt que l'educació sigui privada.

Observem que, tot i no estar marcada com a significativa, la variable FirstGen, observem que es negativa, i això ens explica que penalitza aquelles generacions que són les primeres que estudien (dins de la seva família).

Finalment, cal recalcar que totes les variables que estan al model ens aporten informació molt important (com sexe o FirstGen), i no només les que estan marcades com a significatives, no ens hem de quedar només en les més significatives, sinó que totes les variables afecten.