

ANALISIS DISCRIMINANTE

SERGIO ADRIAN ORTIZ ORTEGA

2022-05-14

```
#LDA
```

```
##Instalar paqueterias
```

```
library(MASS)
library(tidyverse)
library(readxl)
```

cargamos los datos

```
tension<- read_excel("tenso.xlsx")
```

Exploracion de matriz

```
dim(tension)
```

```
## [1] 50 13
```

```
str(tension)
```

```
## tibble [50 x 13] (S3: tbl_df/tbl/data.frame)
## $ edad : num [1:50] 75 55 65 50 65 90 75 60 65 80 ...
## $ anemia : num [1:50] 0 0 0 1 1 1 1 0 1 ...
## $ diabetes : num [1:50] 0 0 0 0 1 0 0 1 0 0 ...
## $ Alta_presión_sanguínea : num [1:50] 1 0 0 0 0 1 0 0 0 1 ...
## $ sexo : num [1:50] 1 1 1 1 0 1 1 1 0 1 ...
## $ fuma : num [1:50] 0 0 1 0 0 1 0 1 0 1 ...
## $ MUERTE_EVENTO : num [1:50] 1 1 1 1 1 1 1 1 1 1 ...
## $ creatinina_fosfoquinasa: num [1:50] 582 7861 146 111 160 ...
## $ fracción_de_eyeción : num [1:50] 20 38 20 20 20 40 15 60 65 35 ...
## $ plaquetas : num [1:50] 265000 263358 162000 210000 327000 ...
## $ suero_creatinina : num [1:50] 1.9 1.1 1.3 1.9 2.7 2.1 1.2 1.1 1.5 9.4 ...
## $ suero_sodio : num [1:50] 130 136 129 137 116 132 137 131 138 133 ...
## $ tiempo : num [1:50] 4 6 7 7 8 8 10 10 10 10 ...
```

```
colnames(tension)
```

```
## [1] "edad" "anemia"
## [3] "diabetes" "Alta_presión_sanguínea"
## [5] "sexo" "fuma"
## [7] "MUERTE_EVENTO" "creatinina_fosfoquinasa"
## [9] "fracción_de_eyeción" "plaquetas"
## [11] "suero_creatinina" "suero_sodio"
## [13] "tiempo"
```

```
##transformar las variables para que funcionen con el codigo
fuma<-factor(tension$fuma,
             levels= c("1","0"))
anemia<-factor(tension$anemia,
              levels= c("1","0"))
sexo<-factor(tension$sexo,
            levels= c("1","0"))
diabetes<-factor(tension$diabetes,
               levels= c("1","0"))
Alta_presión_sanguínea<-factor(tension$Alta_presión_sanguínea,
                              levels= c("1","0"))
MUERTE_EVENTO<-factor(tension$MUERTE_EVENTO,
                     levels= c("1","0"))
edad<-as.numeric(tension$edad,strict = TRUE)
creatinina_fosfoquinasa<-as.numeric(tension$creatinina_fosfoquinasa,strict = TRUE)
fracción_de_eyeción<-as.numeric(tension$fracción_de_eyeción,strict = TRUE)
plaquetas<-as.numeric(tension$plaquetas,strict = FALSE)
suero_creatinina<-as.numeric(tension$suero_creatinina,strict = TRUE)
suero_sodio<-as.numeric(tension$suero_sodio,strict = TRUE)
tiempo<-as.numeric(tension$tiempo,strict = TRUE)
```

ya con los datos convertidos como requerimos los transformamos de nuevo a una matrix

```
Z<-data.frame(edad, anemia, diabetes,Alta_presión_sanguínea,sexo,fuma,
              MUERTE_EVENTO,creatinina_fosfoquinasa,fracción_de_eyeción,
              plaquetas,suero_creatinina,suero_sodio,tiempo)
```

##Se define la matriz de datos y la variable respta con las clasificaciones.

```
x<-Z[,8:13]
y<-Z[,5]
```

##Definir como n y p el numero de flores y variables

```
n<-nrow(x)
p<-ncol(x)
```

##Se aplica el Analisis discriminante lineal (LDA) Cross validation (cv): clasificacion optima

```
lda.iris<-lda(y~.,data=x, CV=TRUE)
```

##contiene las clasificaciones hechas por CV usando LDA.

```
lda.iris$class
```

```
## [1] 1 1 1 1 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1
## [39] 1 1 1 1 1 1 1 1 1 0 1 1
## Levels: 1 0
```

##Creacion de la tabla de clasificaciones buenas y malas

```
table.lda<-table(y,lda.iris$class)
table.lda
```

```
##
## y    1  0
##    1 31  3
##    0 16  0
```

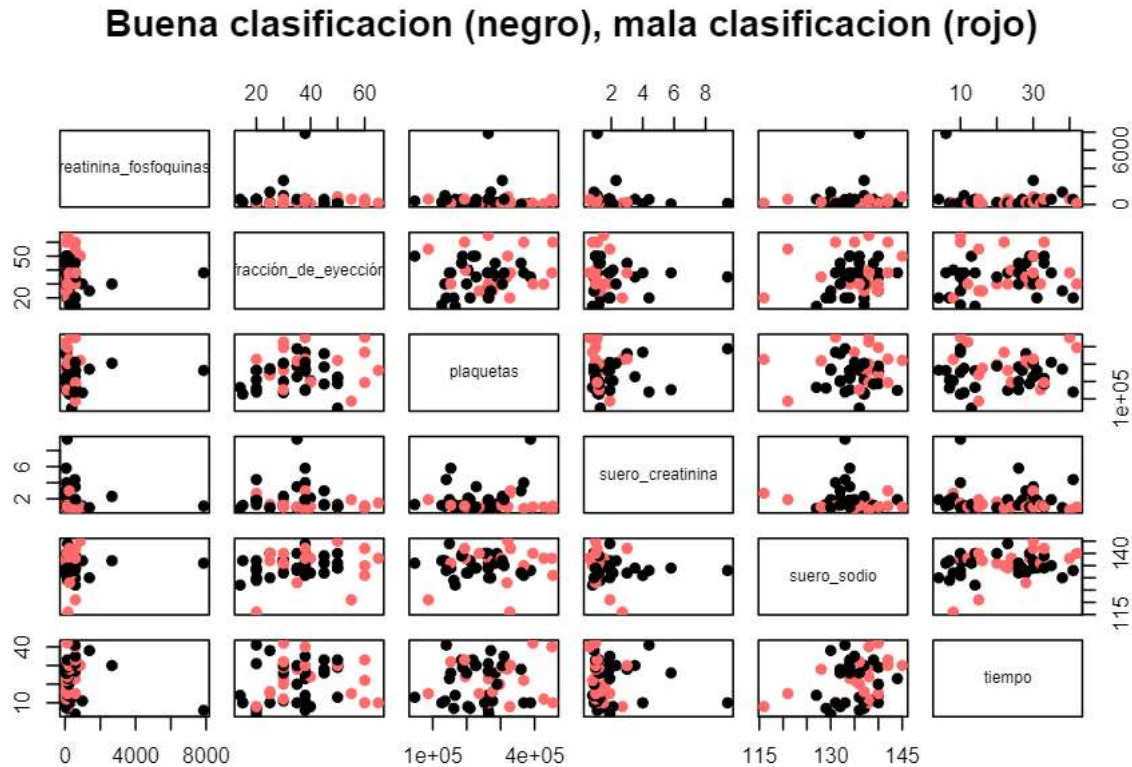
```
##Proporcion de errores
```

```
mis.lda<- n-sum(y==lda.iris$class)
mis.lda/n
```

```
## [1] 0.38
```

```
##scatter plot Buenas clasificaciones en negro y malas en rojo
```

```
col.lda.iris<-c("indianred1","black")[1*(y==lda.iris$class)+1]
pairs(x,main="Buena clasificacion (negro), mala clasificacion (rojo)",
      pch=19,col=col.lda.iris)
```



```
##Probabilidad de pertenencia a uno de los tres grupos
```

```
lda.iris$posterior
```

```
##          1          0
## 1  0.7162268 0.2837732343
## 2  0.9993613 0.0006386579
## 3  0.7236511 0.2763489483
## 4  0.7203173 0.2796827199
## 5  0.9266149 0.0733851406
## 6  0.7299487 0.2700512520
## 7  0.7389435 0.2610565120
## 8  0.2938028 0.7061972092
## 9  0.7817438 0.2182561953
## 10 0.9127947 0.0872053191
## 11 0.6944506 0.3055493799
## 12 0.6197455 0.3802545399
## 13 0.7762392 0.2237607842
## 14 0.6185890 0.3814109993
```

```
## 15 0.5831465 0.4168535242
## 16 0.7783054 0.2216946452
## 17 0.6053724 0.3946276071
## 18 0.6975204 0.3024796443
## 19 0.7062564 0.2937435513
## 20 0.9764318 0.0235681936
## 21 0.6652450 0.3347549740
## 22 0.6413524 0.3586476213
## 23 0.5807226 0.4192774023
## 24 0.4043160 0.5956840029
## 25 0.7030756 0.2969244048
## 26 0.7215117 0.2784883218
## 27 0.6784129 0.3215871410
## 28 0.5906298 0.4093702195
## 29 0.8779285 0.1220714699
## 30 0.6560407 0.3439593175
## 31 0.6791539 0.3208461363
## 32 0.5995452 0.4004548013
## 33 0.6195024 0.3804975762
## 34 0.6019449 0.3980551422
## 35 0.6224993 0.3775006781
## 36 0.7776136 0.2223863683
## 37 0.5865436 0.4134563974
## 38 0.6562886 0.3437113714
## 39 0.7728158 0.2271841598
## 40 0.7252998 0.2747002282
## 41 0.6620036 0.3379963648
## 42 0.7427175 0.2572824844
## 43 0.6698249 0.3301751406
## 44 0.5852029 0.4147971131
## 45 0.7632513 0.2367486884
## 46 0.6128440 0.3871559525
## 47 0.5953065 0.4046935406
## 48 0.2838103 0.7161897130
## 49 0.8332413 0.1667587446
## 50 0.5426934 0.4573066161
```

Grafico de probabilidades

```
plot(1:n, lda.iris$posterior[,1],
     main="Probabilidades a posteriori",
     pch=20, col="blue",
     xlab="Numero de observaciones", ylab="Probabilidades")
points(1:n, lda.iris$posterior[,2],
       pch=20, col="green")
```

Probabilidades a posteriori

