Problemas propuestos de Regresión Logística

Francisco Javier Mercader Martínez

Problema 1

El fichero **processed.cleveland.data**, continene los datos correspondientes a un estudio sobre enfermedad cardíaca por *Cleveland Clinic Foundation*.

El fichero contiene un total de 14 columnas, correspondientes a las siguientes variables: *age, sex, cp, trestbps, chol, fbs, restecg, thalach, exang, oldpeak, slope, ca, thal* y *num*. La variable "num" toma valores 0, 1, 2, 3 y 4, indicando el tipo de anomalía cardíaca. El valor 0 indica ausencia de enfermedad, mientras que el resto de valores indican algún tipo de anomalía. Para la descripción detallada de cada variable, puede consultarse el fichero **heart-disease.names**.

Se desea realizar un análisis de Regresión Logística con el fin de predecir la presencia (o no) de enfermedad cardíaca en función del resto de variables (predictores). Se pide:

1) Importar los datos del fichero **processed.cleveland.data** y poner el nombre de cada variable como se indica en el enunciado. Sustituir la variable "num" por una nueva variable llamada "disease" que valga 0 si no hay enfermedad y que valga 1 cuando haya anomalía cardiaca.

```
mydata <- read.table("../../data/processed.cleveland.data", sep = ",", dec = '.', header
 ↪  = FALSE)
colnames(mydata) <- c("age", "sex", "cp", "trestbps", "chol", "fbs",
                      "restecg", "thalach", "exang", "oldpeak", "slope", "ca",
                      "thal", "num")

# Crear la nueva variable 'disease'
mydata$disease[mydata$num == 0] <- 0
mydata$disease[mydata$num > 0] <- 1

library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
mydata <- select(mydata, -num)
```

De esta forma elimino la columna `num` para que `disease` la sustituya

2) Eliminar todas las filas que tengan algún valor perdido. **Importante:** confirmar primero si todas las variables son de tipo numérico para identificar adecuadamente los valores perdidos.

```r
str(mydata)
```

```
## 'data.frame':    303 obs. of  14 variables:
##  $ age     : num  63 67 67 37 41 56 62 57 63 53 ...
##  $ sex     : num  1 1 1 1 0 1 0 0 1 1 ...
##  $ cp      : num  1 4 4 3 2 2 4 4 4 4 ...
##  $ trestbps: num  145 160 120 130 130 120 140 120 130 140 ...
##  $ chol    : num  233 286 229 250 204 236 268 354 254 203 ...
##  $ fbs     : num  1 0 0 0 0 0 0 0 0 1 ...
##  $ restecg : num  2 2 2 0 2 0 2 0 2 2 ...
##  $ thalach : num  150 108 129 187 172 178 160 163 147 155 ...
##  $ exang   : num  0 1 1 0 0 0 0 1 0 1 ...
##  $ oldpeak : num  2.3 1.5 2.6 3.5 1.4 0.8 3.6 0.6 1.4 3.1 ...
##  $ slope   : num  3 2 2 3 1 1 3 1 2 3 ...
##  $ ca      : chr  "0.0" "3.0" "2.0" "0.0" ...
##  $ thal    : chr  "6.0" "3.0" "7.0" "3.0" ...
##  $ disease : num  0 1 1 0 0 0 1 0 1 1 ...
```

```r
# Las columnas ca y thal son de tipo chr
mydata$ca <- as.numeric(mydata$ca)
```

```
## Warning: NAs introducidos por coerción
```

```r
mydata$thal <- as.numeric(mydata$thal)
```

```
## Warning: NAs introducidos por coerción
```

```r
# Como nos da el aviso de que hay valores NA's, vamos a eliminar las filas que
# los contienen

mydata <- na.omit(mydata)

# El DataFrame orginal tenía 303 filas y ahora hay 297.
summary(mydata)
```

```
##       age             sex               cp           trestbps    
##  Min.   :29.00   Min.   :0.0000   Min.   :1.000   Min.   : 94.0  
##  1st Qu.:48.00   1st Qu.:0.0000   1st Qu.:3.000   1st Qu.:120.0  
##  Median :56.00   Median :1.0000   Median :3.000   Median :130.0  
##  Mean   :54.54   Mean   :0.6768   Mean   :3.158   Mean   :131.7  
##  3rd Qu.:61.00   3rd Qu.:1.0000   3rd Qu.:4.000   3rd Qu.:140.0  
##  Max.   :77.00   Max.   :1.0000   Max.   :4.000   Max.   :200.0  
##       chol            fbs            restecg          thalach     
##  Min.   :126.0   Min.   :0.0000   Min.   :0.0000   Min.   : 71.0  
##  1st Qu.:211.0   1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:133.0  
##  Median :243.0   Median :0.0000   Median :1.0000   Median :153.0  
##  Mean   :247.4   Mean   :0.1448   Mean   :0.9966   Mean   :149.6  
##  3rd Qu.:276.0   3rd Qu.:0.0000   3rd Qu.:2.0000   3rd Qu.:166.0  
##  Max.   :564.0   Max.   :1.0000   Max.   :2.0000   Max.   :202.0  
##      exang           oldpeak          slope             ca        
##  Min.   :0.0000   Min.   :0.000   Min.   :1.000   Min.   :0.0000  
##  1st Qu.:0.0000   1st Qu.:0.000   1st Qu.:1.000   1st Qu.:0.0000  
##  Median :0.0000   Median :0.800   Median :2.000   Median :0.0000  
##  Mean   :0.3266   Mean   :1.056   Mean   :1.603   Mean   :0.6768  
##  3rd Qu.:1.0000   3rd Qu.:1.600   3rd Qu.:2.000   3rd Qu.:1.0000
```

```
##  Max.   :1.0000   Max.   :6.200   Max.   :3.000   Max.   :3.0000
##       thal           disease
##  Min.   :3.000   Min.   :0.0000
##  1st Qu.:3.000   1st Qu.:0.0000
##  Median :3.000   Median :0.0000
##  Mean   :4.731   Mean   :0.4613
##  3rd Qu.:7.000   3rd Qu.:1.0000
##  Max.   :7.000   Max.   :1.0000
```

3) Pasar a tipo factor las variables que por naturaleza sean de tipo categórico.

```
mydata$sex <- factor(mydata$sex)
mydata$cp <- factor(mydata$cp)
mydata$fbs <- factor(mydata$fbs)
mydata$restecg <- factor(mydata$restecg)
mydata$exang <- factor(mydata$exang)
mydata$slope <- factor(mydata$slope)
mydata$ca <- factor(mydata$ca)
mydata$thal <- factor(mydata$thal)
mydata$disease <- factor(mydata$disease)

summary(mydata)
```

```
##       age        sex     cp         trestbps          chol       fbs
##  Min.   :29.00   0: 96   1: 23   Min.   : 94.0   Min.   :126.0   0:254
##  1st Qu.:48.00   1:201   2: 49   1st Qu.:120.0   1st Qu.:211.0   1: 43
##  Median :56.00           3: 83   Median :130.0   Median :243.0
##  Mean   :54.54           4:142   Mean   :131.7   Mean   :247.4
##  3rd Qu.:61.00                   3rd Qu.:140.0   3rd Qu.:276.0
##  Max.   :77.00                   Max.   :200.0   Max.   :564.0
##  restecg    thalach      exang      oldpeak      slope   ca      thal
##  0:147   Min.   : 71.0   0:200   Min.   :0.000   1:139   0:174   3:164
##  1:  4   1st Qu.:133.0   1: 97   1st Qu.:0.000   2:137   1: 65   6: 18
##  2:146   Median :153.0           Median :0.800   3: 21   2: 38   7:115
##          Mean   :149.6           Mean   :1.056           3: 20
##          3rd Qu.:166.0           3rd Qu.:1.600
##          Max.   :202.0           Max.   :6.200
##  disease
##  0:160
##  1:137
##
##
##
##
```

4) Dividir el conjunto de datos en entrenamiento y prueba (70% entrenamiento, 30% prueba). Tomar semilla 123.

```
set.seed(123)

indice_entrenamiento <- sample(1:nrow(mydata), 0.7 * nrow(mydata))

# Conjunto de entrenamiento
train_data <- mydata[indice_entrenamiento, ]

# Conjunto de test
```

```r
test_data <- mydata[-indice_entrenamiento, ]
```

5) Con los datos de entrenamiento, obtener el modelo ajustado de Regresión Logística usando todos los predictores. ¿Son todos los predictores significativos?

```r
modelo_ajustado <- glm(disease ~ ., data = train_data, family = "binomial")
summary(modelo_ajustado)
```

```
##
## Call:
## glm(formula = disease ~ ., family = "binomial", data = train_data)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -9.633409   4.209619  -2.288 0.022113 *
## age         -0.019917   0.032526  -0.612 0.540319
## sex1         1.909221   0.745670   2.560 0.010455 *
## cp2          1.550798   0.935746   1.657 0.097462 .
## cp3          0.035539   0.831372   0.043 0.965903
## cp4          2.282459   0.804829   2.836 0.004569 **
## trestbps     0.032780   0.014958   2.191 0.028419 *
## chol         0.005643   0.005920   0.953 0.340477
## fbs1        -1.096125   0.821602  -1.334 0.182161
## restecg1     1.188419   3.934905   0.302 0.762637
## restecg2     0.449819   0.500431   0.899 0.368726
## thalach     -0.010109   0.015991  -0.632 0.527273
## exang1       0.493119   0.582936   0.846 0.397596
## oldpeak      0.782926   0.314411   2.490 0.012769 *
## slope2       1.221948   0.615551   1.985 0.047130 *
## slope3       1.095381   1.075361   1.019 0.308385
## ca1          2.627145   0.661726   3.970 7.18e-05 ***
## ca2          3.097165   0.971737   3.187 0.001436 **
## ca3          3.895965   2.055422   1.895 0.058032 .
## thal6        0.256946   0.986484   0.260 0.794504
## thal7        1.905655   0.566459   3.364 0.000768 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 284.83  on 206  degrees of freedom
## Residual deviance: 120.47  on 186  degrees of freedom
## AIC: 162.47
##
## Number of Fisher Scoring iterations: 7
```

Los predictores con un valor p pequeño (generalmente menor a 0.05) son considerados significativos. En este caso, los predictores significativos son:

- (Intercept)
- cp
- exang1
- ca
- thal

Estos predictores tienen un valor p menor a 0.05, lo que indica que hay una fuerte evidencia de que estos predictores tienen un efecto significativo en la variable de respuesta `disease`.

6) Obtener las predicciones para los datos del conjunto de prueba, es decir, la probabilidad predicha de padecer cardíaca para cada individuo del conjunto de testeo.

```
predictions <- predict(modelo_ajustado, newdata = test_data, type = "response")
```

7) Veamos ahora el problema de Regresión Logística como un problema de clasificiación. Usando las predicciones del apartado anterior y tomando como punto de corte la probabilidad de 0.5, obtener la clase predicha para los individuos del conjunto de prueba. Medir la eficiencia del modelo calculando la matriz de confusión, accuracy, sensibilidad y especificidad.

```
predic_grupos <- ifelse(predictions > 0.5, 1, 0)

matriz_confusion <- table(test_data$disease, predic_grupos)

VP <- matriz_confusion[2, 2]
FN <- matriz_confusion[2, 1]
VN <- matriz_confusion[1, 1]
FP <- matriz_confusion[1, 2]

sensibilidad <- VP/(VP+FN)
especificidad <- VN/(VN+FP)
accuracy <- (VP/VN)/(VP+FP+VN+FN)
paste("Accuracy =", accuracy)
```

```
## [1] "Accuracy = 0.00873015873015873"
```

```
paste("Sensibilidad =", sensibilidad)
```

```
## [1] "Sensibilidad = 0.75"
```

```
paste("Especificidad =", especificidad)
```

```
## [1] "Especificidad = 0.91304347826087"
```

8) Para los datos del conjunto de prueba, obtener la curva ROC del método de clasificación, calcular el AUC (área bajo la curva) e interpretar el resultado.

```
library("pROC")
```

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

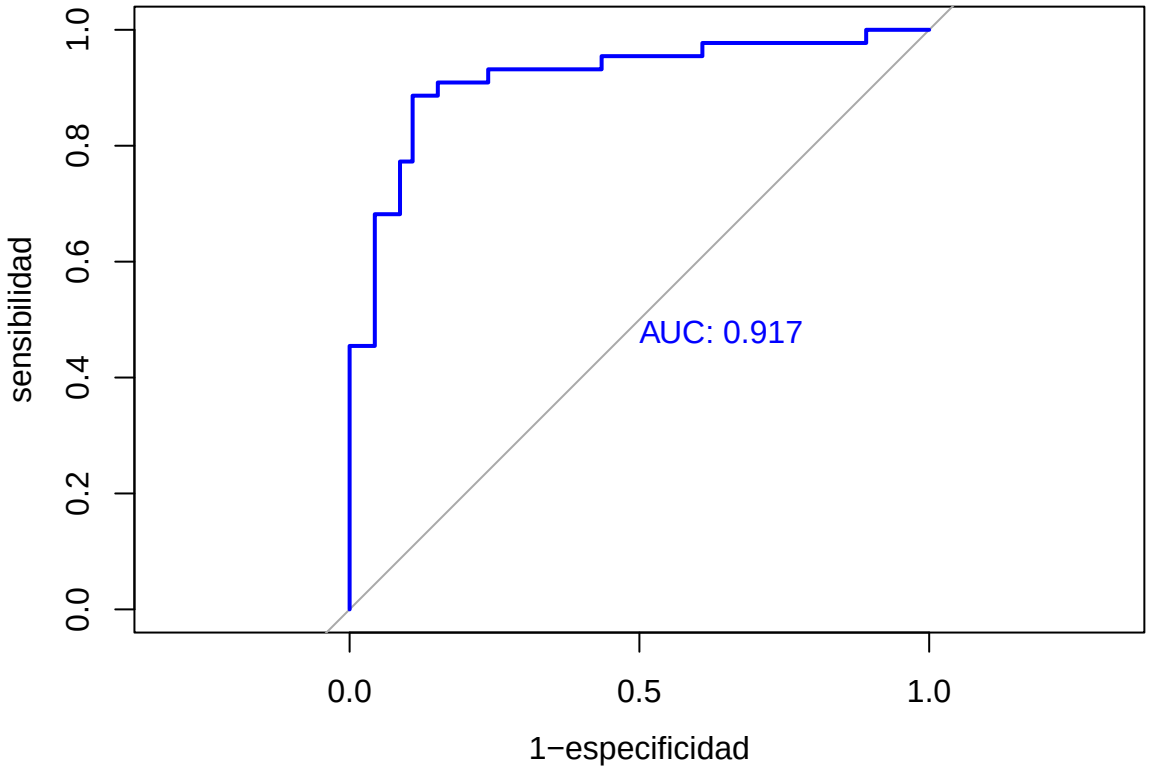
```
##
## Attaching package: 'pROC'
```

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

```
roc(test_data$disease, predictions, plot = TRUE,
    legacy.axes = TRUE, percent = FALSE,
    xlab = "1-especificidad", ylab = "sensibilidad",
    col = "blue", lwd = 2, print.auc = TRUE)
```

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

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




```
##
## Call:
## roc.default(response = test_data$disease, predictor = predictions,     percent = FALSE, plot = TRUE,
##
## Data: predictions in 46 controls (test_data$disease 0) < 44 cases (test_data$disease 1).
## Area under the curve: 0.917
```

9) Repetir el análisis (apartado 5 y siguientes) pero aplicando primero los métodos de selección de regresores, con el fin de proponer un modelo más parsimonioso.

```
modelo_cte <- glm(disease ~ 1, data = mydata, family = "binomial")
modelo_backward <- step(modelo_ajustado, direction = "backward")
```

```
## Start:  AIC=162.47
## disease ~ age + sex + cp + trestbps + chol + fbs + restecg +
##     thalach + exang + oldpeak + slope + ca + thal
##
##            Df Deviance    AIC
## - restecg   2   121.35 159.35
## - age       1   120.85 160.85
## - thalach   1   120.88 160.88
## - exang     1   121.18 161.18
## - chol      1   121.39 161.40
## - fbs       1   122.35 162.35
## <none>          120.47 162.47
## - slope     2   124.61 162.61
## - trestbps  1   125.62 165.62
```

```
## - oldpeak   1   127.26 167.26
## - sex       1   127.95 167.95
## - thal      2   134.09 172.09
## - cp        3   136.61 172.61
## - ca        3   149.66 185.66
##
## Step:  AIC=159.35
## disease ~ age + sex + cp + trestbps + chol + fbs + thalach +
##     exang + oldpeak + slope + ca + thal
##
##            Df Deviance    AIC
## - age       1   121.68 157.68
## - thalach   1   121.91 157.91
## - exang     1   122.01 158.01
## - chol      1   122.47 158.47
## - fbs       1   123.12 159.12
## <none>          121.35 159.35
## - slope     2   126.00 160.00
## - trestbps  1   126.99 162.99
## - oldpeak   1   128.22 164.22
## - sex       1   129.18 165.18
## - thal      2   134.34 168.34
## - cp        3   137.26 169.26
## - ca        3   151.00 183.00
##
## Step:  AIC=157.68
## disease ~ sex + cp + trestbps + chol + fbs + thalach + exang +
##     oldpeak + slope + ca + thal
##
##            Df Deviance    AIC
## - thalach   1   122.03 156.03
## - exang     1   122.35 156.35
## - chol      1   122.75 156.75
## - fbs       1   123.54 157.54
## <none>          121.68 157.68
## - slope     2   126.15 158.15
## - trestbps  1   126.99 160.99
## - oldpeak   1   129.66 163.66
## - sex       1   129.90 163.90
## - thal      2   134.41 166.41
## - cp        3   137.95 167.95
## - ca        3   151.65 181.65
##
## Step:  AIC=156.03
## disease ~ sex + cp + trestbps + chol + fbs + exang + oldpeak +
##     slope + ca + thal
##
##            Df Deviance    AIC
## - chol      1   123.08 155.08
## - exang     1   123.10 155.10
## - fbs       1   123.93 155.93
## <none>          122.03 156.03
## - slope     2   127.29 157.29
## - trestbps  1   127.19 159.19
```

```
## - sex       1   130.01 162.01
## - oldpeak   1   130.04 162.04
## - thal      2   134.88 164.88
## - cp        3   139.69 167.69
## - ca        3   154.46 182.46
##
## Step:  AIC=155.08
## disease ~ sex + cp + trestbps + fbs + exang + oldpeak + slope +
##     ca + thal
##
##            Df Deviance    AIC
## - exang     1   124.10 154.10
## - fbs       1   124.91 154.91
## <none>          123.08 155.08
## - slope     2   129.03 157.03
## - trestbps  1   128.96 158.96
## - sex       1   130.17 160.17
## - oldpeak   1   131.32 161.32
## - thal      2   136.53 164.53
## - cp        3   141.59 167.59
## - ca        3   155.44 181.44
##
## Step:  AIC=154.1
## disease ~ sex + cp + trestbps + fbs + oldpeak + slope + ca +
##     thal
##
##            Df Deviance    AIC
## - fbs       1   125.63 153.63
## <none>          124.10 154.10
## - slope     2   130.00 156.00
## - trestbps  1   129.87 157.87
## - sex       1   130.62 158.62
## - oldpeak   1   134.13 162.13
## - thal      2   139.03 165.03
## - cp        3   146.29 170.29
## - ca        3   156.05 180.05
##
## Step:  AIC=153.63
## disease ~ sex + cp + trestbps + oldpeak + slope + ca + thal
##
##            Df Deviance    AIC
## <none>          125.63 153.63
## - slope     2   131.02 155.02
## - trestbps  1   130.29 156.29
## - sex       1   131.78 157.78
## - oldpeak   1   135.92 161.92
## - thal      2   140.07 164.07
## - cp        3   149.75 171.75
## - ca        3   156.13 178.13
```

```
modelo_forward <- step(modelo_cte, direction = "forward", scope =
 ↪  formula(modelo_ajustado))
```

```
## Start:  AIC=411.95
```

```
## disease ~ 1
##
##            Df Deviance    AIC
## + thal      2   323.39 329.39
## + cp        3   328.75 336.75
## + ca        3   333.93 341.93
## + oldpeak   1   350.48 354.48
## + thalach   1   351.97 355.97
## + exang     1   355.48 359.48
## + slope     2   365.16 371.16
## + sex       1   386.12 390.12
## + age       1   394.25 398.25
## + restecg   2   400.28 406.28
## + trestbps  1   402.88 406.88
## <none>          409.95 411.95
## + chol      1   408.03 412.03
## + fbs       1   409.94 413.94
##
## Step:  AIC=329.39
## disease ~ thal
##
##            Df Deviance    AIC
## + ca        3   268.65 280.65
## + cp        3   275.86 287.86
## + thalach   1   288.27 296.27
## + oldpeak   1   294.27 302.27
## + exang     1   295.84 303.84
## + slope     2   300.98 310.98
## + age       1   312.61 320.61
## + restecg   2   311.48 321.48
## + sex       1   320.64 328.64
## + trestbps  1   320.68 328.68
## + chol      1   320.89 328.89
## <none>          323.39 329.39
## + fbs       1   322.99 330.99
##
## Step:  AIC=280.65
## disease ~ thal + ca
##
##            Df Deviance    AIC
## + cp        3   230.40 248.40
## + exang     1   245.07 259.07
## + thalach   1   246.15 260.15
## + oldpeak   1   247.64 261.64
## + slope     2   246.33 262.33
## + restecg   2   259.55 275.55
## + sex       1   265.29 279.29
## + fbs       1   265.81 279.81
## + trestbps  1   266.15 280.15
## <none>          268.65 280.65
## + chol      1   267.97 281.97
## + age       1   268.36 282.36
##
## Step:  AIC=248.4
```

```
## disease ~ thal + ca + cp
##
##            Df Deviance    AIC
## + oldpeak   1   213.32 233.32
## + slope     2   212.54 234.54
## + thalach   1   221.44 241.44
## + exang     1   223.23 243.23
## + trestbps  1   226.04 246.04
## + restecg   2   224.44 246.44
## + sex       1   226.74 246.74
## <none>          230.40 248.40
## + chol      1   229.38 249.38
## + age       1   230.00 250.00
## + fbs       1   230.12 250.12
##
## Step:  AIC=233.32
## disease ~ thal + ca + cp + oldpeak
##
##            Df Deviance    AIC
## + slope     2   204.79 228.79
## + thalach   1   209.22 231.22
## + exang     1   209.42 231.42
## + sex       1   209.88 231.88
## + trestbps  1   209.97 231.97
## + restecg   2   208.94 232.94
## <none>          213.32 233.32
## + chol      1   212.72 234.72
## + age       1   213.05 235.05
## + fbs       1   213.17 235.17
##
## Step:  AIC=228.79
## disease ~ thal + ca + cp + oldpeak + slope
##
##            Df Deviance    AIC
## + sex       1   198.20 224.20
## + trestbps  1   201.03 227.03
## + exang     1   201.93 227.93
## <none>          204.79 228.79
## + thalach   1   203.34 229.34
## + restecg   2   201.52 229.52
## + chol      1   204.48 230.48
## + fbs       1   204.63 230.63
## + age       1   204.79 230.79
##
## Step:  AIC=224.2
## disease ~ thal + ca + cp + oldpeak + slope + sex
##
##            Df Deviance    AIC
## + trestbps  1   192.57 220.57
## + exang     1   194.97 222.97
## + thalach   1   195.92 223.92
## <none>          198.20 224.20
## + chol      1   196.71 224.71
## + restecg   2   195.02 225.02
```

```
## + age       1   197.94 225.94
## + fbs       1   197.97 225.97
##
## Step:  AIC=220.57
## disease ~ thal + ca + cp + oldpeak + slope + sex + trestbps
##
##           Df Deviance    AIC
## + exang    1   189.76 219.76
## + thalach  1   189.78 219.78
## <none>         192.57 220.57
## + chol     1   191.53 221.53
## + fbs      1   191.85 221.85
## + restecg  2   190.51 222.51
## + age      1   192.56 222.56
##
## Step:  AIC=219.76
## disease ~ thal + ca + cp + oldpeak + slope + sex + trestbps +
##     exang
##
##           Df Deviance    AIC
## + thalach  1   187.74 219.74
## <none>         189.76 219.76
## + chol     1   188.74 220.74
## + fbs      1   188.85 220.85
## + age      1   189.75 221.75
## + restecg  2   187.77 221.77
##
## Step:  AIC=219.74
## disease ~ thal + ca + cp + oldpeak + slope + sex + trestbps +
##     exang + thalach
##
##           Df Deviance    AIC
## <none>         187.74 219.74
## + chol     1   186.40 220.40
## + fbs      1   186.88 220.88
## + age      1   187.29 221.29
## + restecg  2   185.76 221.76
```

```
modelo_stepwise <- step(modelo_cte, direction = "both", scope = formula(modelo_ajustado))
```

```
## Start:  AIC=411.95
## disease ~ 1
##
##            Df Deviance    AIC
## + thal      2   323.39 329.39
## + cp        3   328.75 336.75
## + ca        3   333.93 341.93
## + oldpeak   1   350.48 354.48
## + thalach   1   351.97 355.97
## + exang     1   355.48 359.48
## + slope     2   365.16 371.16
## + sex       1   386.12 390.12
## + age       1   394.25 398.25
## + restecg   2   400.28 406.28
```

```
## + trestbps  1   402.88 406.88
## <none>          409.95 411.95
## + chol      1   408.03 412.03
## + fbs       1   409.94 413.94
##
## Step:  AIC=329.39
## disease ~ thal
##
##            Df Deviance    AIC
## + ca        3   268.65 280.65
## + cp        3   275.86 287.86
## + thalach   1   288.27 296.27
## + oldpeak   1   294.27 302.27
## + exang     1   295.84 303.84
## + slope     2   300.98 310.98
## + age       1   312.61 320.61
## + restecg   2   311.48 321.48
## + sex       1   320.64 328.64
## + trestbps  1   320.68 328.68
## + chol      1   320.89 328.89
## <none>          323.39 329.39
## + fbs       1   322.99 330.99
## - thal      2   409.95 411.95
##
## Step:  AIC=280.65
## disease ~ thal + ca
##
##            Df Deviance    AIC
## + cp        3   230.40 248.40
## + exang     1   245.08 259.08
## + thalach   1   246.15 260.15
## + oldpeak   1   247.64 261.64
## + slope     2   246.33 262.33
## + restecg   2   259.54 275.54
## + sex       1   265.29 279.29
## + fbs       1   265.81 279.81
## + trestbps  1   266.15 280.15
## <none>          268.65 280.65
## + chol      1   267.97 281.97
## + age       1   268.36 282.36
## - ca        3   323.39 329.39
## - thal      2   333.93 341.93
##
## Step:  AIC=248.4
## disease ~ thal + ca + cp
##
##            Df Deviance    AIC
## + oldpeak   1   213.32 233.32
## + slope     2   212.54 234.54
## + thalach   1   221.44 241.44
## + exang     1   223.23 243.23
## + trestbps  1   226.04 246.04
## + restecg   2   224.44 246.44
## + sex       1   226.74 246.74
```

```
## <none>           230.40 248.40
## + chol      1    229.38 249.38
## + age       1    230.00 250.00
## + fbs       1    230.12 250.12
## - cp        3    268.65 280.65
## - ca        3    275.86 287.86
## - thal      2    274.00 288.00
##
## Step:  AIC=233.32
## disease ~ thal + ca + cp + oldpeak
##
##            Df Deviance    AIC
## + slope     2   204.79 228.79
## + thalach   1   209.22 231.22
## + exang     1   209.42 231.42
## + sex       1   209.88 231.88
## + trestbps  1   209.97 231.97
## + restecg   2   208.94 232.94
## <none>          213.32 233.32
## + chol      1   212.72 234.72
## + age       1   213.05 235.05
## + fbs       1   213.17 235.17
## - oldpeak   1   230.40 248.40
## - thal      2   243.69 259.69
## - cp        3   247.64 261.64
## - ca        3   253.68 267.68
##
## Step:  AIC=228.79
## disease ~ thal + ca + cp + oldpeak + slope
##
##            Df Deviance    AIC
## + sex       1   198.20 224.20
## + trestbps  1   201.03 227.03
## + exang     1   201.93 227.93
## <none>          204.79 228.79
## + thalach   1   203.34 229.34
## + restecg   2   201.52 229.52
## + chol      1   204.48 230.48
## + fbs       1   204.63 230.63
## + age       1   204.79 230.79
## - slope     2   213.32 233.32
## - oldpeak   1   212.54 234.54
## - thal      2   233.70 253.70
## - cp        3   238.09 256.10
## - ca        3   246.35 264.35
##
## Step:  AIC=224.2
## disease ~ thal + ca + cp + oldpeak + slope + sex
##
##            Df Deviance    AIC
## + trestbps  1   192.57 220.57
## + exang     1   194.97 222.97
## + thalach   1   195.92 223.92
## <none>          198.20 224.20
```

```
## + chol      1   196.71 224.71
## + restecg   2   195.02 225.02
## + age       1   197.94 225.94
## + fbs       1   197.97 225.97
## - sex       1   204.79 228.79
## - oldpeak   1   204.88 228.88
## - slope     2   209.88 231.88
## - thal      2   215.38 237.38
## - cp        3   233.53 253.53
## - ca        3   238.39 258.39
##
## Step:  AIC=220.57
## disease ~ thal + ca + cp + oldpeak + slope + sex + trestbps
##
##            Df Deviance    AIC
## + exang     1   189.76 219.76
## + thalach   1   189.78 219.78
## <none>          192.57 220.57
## + chol      1   191.53 221.53
## + fbs       1   191.85 221.85
## + restecg   2   190.51 222.51
## + age       1   192.56 222.56
## - oldpeak   1   197.85 223.85
## - trestbps  1   198.20 224.20
## - sex       1   201.03 227.03
## - slope     2   205.23 229.23
## - thal      2   208.00 232.00
## - cp        3   231.56 253.56
## - ca        3   234.44 256.44
##
## Step:  AIC=219.76
## disease ~ thal + ca + cp + oldpeak + slope + sex + trestbps +
##     exang
##
##            Df Deviance    AIC
## + thalach   1   187.74 219.74
## <none>          189.76 219.76
## - exang     1   192.57 220.57
## + chol      1   188.74 220.74
## + fbs       1   188.85 220.85
## + age       1   189.75 221.75
## + restecg   2   187.77 221.77
## - oldpeak   1   194.31 222.31
## - trestbps  1   194.97 222.97
## - sex       1   198.56 226.56
## - slope     2   201.14 227.14
## - thal      2   203.21 229.21
## - cp        3   218.11 242.11
## - ca        3   231.23 255.23
##
## Step:  AIC=219.74
## disease ~ thal + ca + cp + oldpeak + slope + sex + trestbps +
##     exang + thalach
##
```

```
##            Df Deviance    AIC
## <none>          187.74 219.74
## - thalach   1   189.76 219.76
## - exang     1   189.78 219.78
## + chol      1   186.40 220.40
## + fbs       1   186.88 220.88
## + age       1   187.29 221.29
## - oldpeak   1   191.66 221.66
## + restecg   2   185.76 221.76
## - trestbps  1   193.48 223.48
## - slope     2   195.94 223.94
## - sex       1   197.45 227.45
## - thal      2   200.93 228.93
## - cp        3   212.80 238.80
## - ca        3   225.74 251.74
```

```r
modelo_reducido <- glm(disease ~ thal + ca + oldpeak + cp + trestbps + sex + slope, data
 ↪  = train_data, family = "binomial")

predictions2 <- predict(modelo_reducido, newdata = test_data, type = "response")
predict_group_2 <- ifelse(predictions2 > 0.5, 1, 0)

# Matriz de confusión
matriz_confusion2 <- table(test_data$disease, predict_group_2)

VP2 <- matriz_confusion2[2, 2]
FN2 <- matriz_confusion2[2, 1]
VN2 <- matriz_confusion2[1, 1]
FP2 <- matriz_confusion2[1, 2]

sensibilidad2 <- VP2/(VP2+FN2)
especificidad2 <- VN2/(VN2+FP2)
accuracy2 <- (VP2/VN2)/(VP2+FP2+VN2+FN2)
paste("Accuracy =", accuracy2)
```

```
## [1] "Accuracy = 0.00899470899470899"
```

```r
paste("Sensibilidad =", sensibilidad2)
```

```
## [1] "Sensibilidad = 0.772727272727273"
```

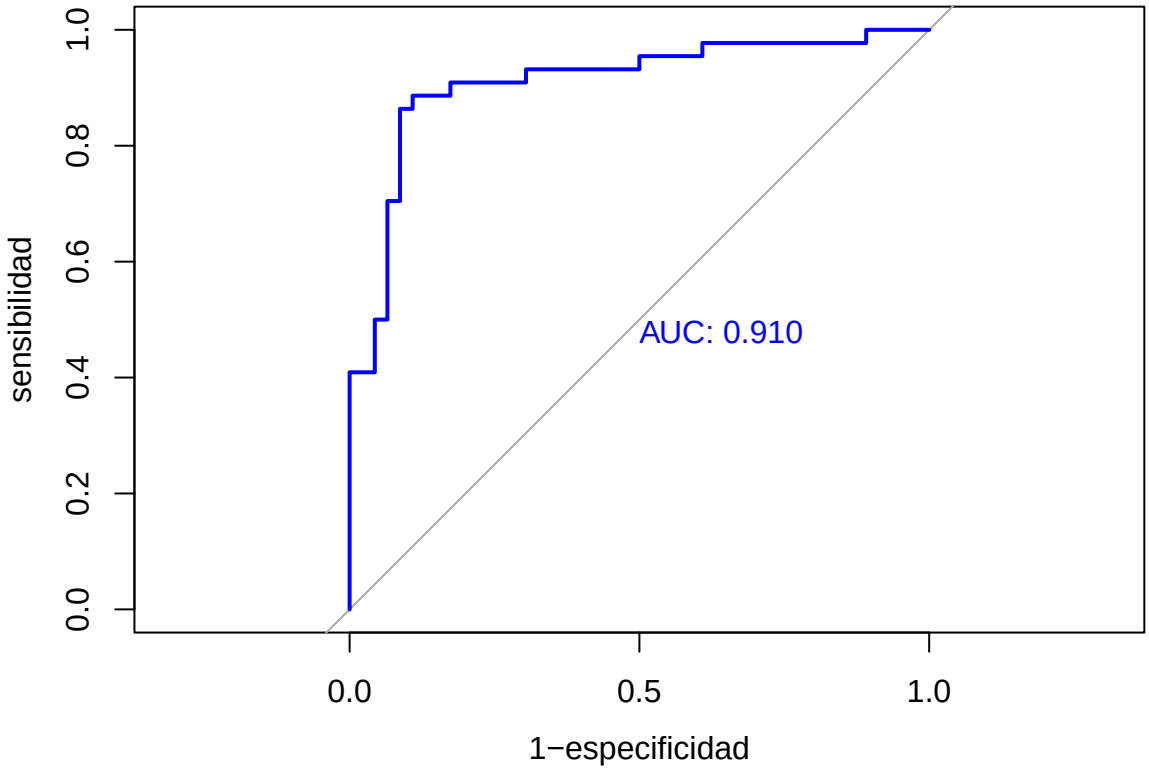
```r
paste("Especificidad =", especificidad2)
```

```
## [1] "Especificidad = 0.91304347826087"
```

```r
# Curva ROC
library("pROC")
roc(test_data$disease, predictions2, plot = TRUE,
    legacy.axes = TRUE, percent = FALSE,
    xlab = "1-especificidad", ylab = "sensibilidad",
    col = "blue", lwd = 2, print.auc = TRUE)
```

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

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



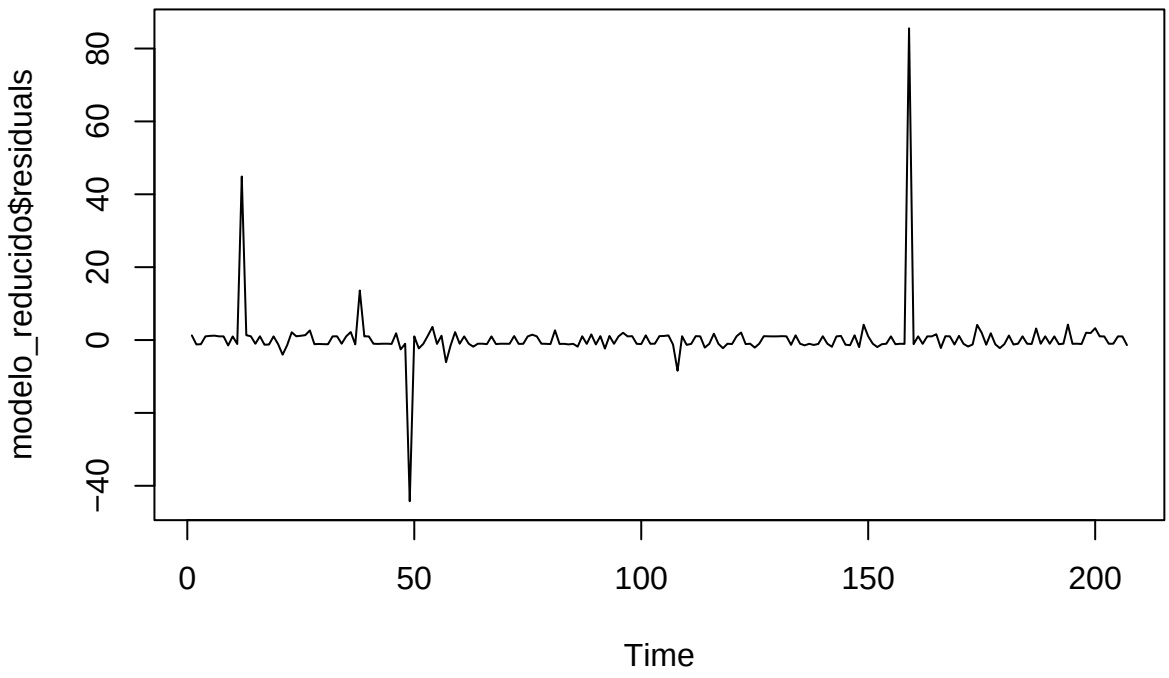

```
##
## Call:
## roc.default(response = test_data$disease, predictor = predictions2,     percent = FALSE, plot = TRUE
##
## Data: predictions2 in 46 controls (test_data$disease 0) < 44 cases (test_data$disease 1).
## Area under the curve: 0.9101
```

```
rms::vif(modelo_reducido)
```

```
## Warning in .recacheSubclasses(def@className, def, env): undefined subclass
## "ndiMatrix" of class "replValueSp"; definition not updated
```

```
##    thal6    thal7      ca1      ca2      ca3  oldpeak      cp2      cp3
## 1.265071 1.279986 1.422441 1.292394 1.045100 1.608831 2.448666 2.207648
##      cp4 trestbps     sex1   slope2   slope3
## 2.973619 1.173107 1.423773 1.668836 1.421255
```

```
ts.plot(modelo_reducido$residuals)
```



Problema 2

¿Qué sucede en el problema anterior si no se realiza el apartado 4? Es decir, qué sucede si no separamos el conjunto de datos en dos subconjuntos de entrenamiento y prueba.

Puede intentar repetir todo el ejercicio en este nuevo escenario y ver qué sucede con las medidads de bondad del ajuste o medidas de eficiencia del método clasificador.

Repetimos todo el proceso hasta salvo el apartado 4.

```
mydata <- read.table("../../data/processed.cleveland.data", sep = ",", dec = '.', header
 ↪  = FALSE)
colnames(mydata) <- c("age", "sex", "cp", "trestbps", "chol", "fbs",
                      "restecg", "thalach", "exang", "oldpeak", "slope", "ca",
                      "thal", "num")

# Crear la nueva variable 'disease'
mydata$disease[mydata$num == 0] <- 0
mydata$disease[mydata$num > 0] <- 1

library(dplyr)
mydata <- select(mydata, -num)
```

```
str(mydata)
```

```
## 'data.frame':    303 obs. of  14 variables:
##  $ age     : num  63 67 67 37 41 56 62 57 63 53 ...
```

```
##  $ sex     : num  1 1 1 1 0 1 0 0 1 1 ...
##  $ cp      : num  1 4 4 3 2 2 4 4 4 4 ...
##  $ trestbps: num  145 160 120 130 130 120 140 120 130 140 ...
##  $ chol    : num  233 286 229 250 204 236 268 354 254 203 ...
##  $ fbs     : num  1 0 0 0 0 0 0 0 0 1 ...
##  $ restecg : num  2 2 2 0 2 0 2 0 2 2 ...
##  $ thalach : num  150 108 129 187 172 178 160 163 147 155 ...
##  $ exang   : num  0 1 1 0 0 0 0 1 0 1 ...
##  $ oldpeak : num  2.3 1.5 2.6 3.5 1.4 0.8 3.6 0.6 1.4 3.1 ...
##  $ slope   : num  3 2 2 3 1 1 3 1 2 3 ...
##  $ ca      : chr  "0.0" "3.0" "2.0" "0.0" ...
##  $ thal    : chr  "6.0" "3.0" "7.0" "3.0" ...
##  $ disease : num  0 1 1 0 0 0 1 0 1 1 ...
```

```
# Las columnas ca y thal son de tipo chr
mydata$ca <- as.numeric(mydata$ca)
```

```
## Warning: NAs introducidos por coerción
```

```
mydata$thal <- as.numeric(mydata$thal)
```

```
## Warning: NAs introducidos por coerción
```

```
# Como nos da el aviso de que hay valores NA's, vamos a eliminar las filas que
# los contienen

mydata <- na.omit(mydata)

# El DataFrame orginal tenía 303 filas y ahora hay 297.
summary(mydata)
```

```
##       age             sex               cp           trestbps
##  Min.   :29.00   Min.   :0.0000   Min.   :1.000   Min.   : 94.0
##  1st Qu.:48.00   1st Qu.:0.0000   1st Qu.:3.000   1st Qu.:120.0
##  Median :56.00   Median :1.0000   Median :3.000   Median :130.0
##  Mean   :54.54   Mean   :0.6768   Mean   :3.158   Mean   :131.7
##  3rd Qu.:61.00   3rd Qu.:1.0000   3rd Qu.:4.000   3rd Qu.:140.0
##  Max.   :77.00   Max.   :1.0000   Max.   :4.000   Max.   :200.0
##       chol            fbs            restecg          thalach
##  Min.   :126.0   Min.   :0.0000   Min.   :0.0000   Min.   : 71.0
##  1st Qu.:211.0   1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:133.0
##  Median :243.0   Median :0.0000   Median :1.0000   Median :153.0
##  Mean   :247.4   Mean   :0.1448   Mean   :0.9966   Mean   :149.6
##  3rd Qu.:276.0   3rd Qu.:0.0000   3rd Qu.:2.0000   3rd Qu.:166.0
##  Max.   :564.0   Max.   :1.0000   Max.   :2.0000   Max.   :202.0
##      exang           oldpeak          slope             ca
##  Min.   :0.0000   Min.   :0.000   Min.   :1.000   Min.   :0.0000
##  1st Qu.:0.0000   1st Qu.:0.000   1st Qu.:1.000   1st Qu.:0.0000
##  Median :0.0000   Median :0.800   Median :2.000   Median :0.0000
##  Mean   :0.3266   Mean   :1.056   Mean   :1.603   Mean   :0.6768
##  3rd Qu.:1.0000   3rd Qu.:1.600   3rd Qu.:2.000   3rd Qu.:1.0000
##  Max.   :1.0000   Max.   :6.200   Max.   :3.000   Max.   :3.0000
##       thal          disease
##  Min.   :3.000   Min.   :0.0000
##  1st Qu.:3.000   1st Qu.:0.0000
```

```
##  Median :3.000   Median :0.0000
##  Mean   :4.731   Mean   :0.4613
##  3rd Qu.:7.000   3rd Qu.:1.0000
##  Max.   :7.000   Max.   :1.0000
```

```
mydata$sex <- factor(mydata$sex)
mydata$cp <- factor(mydata$cp)
mydata$fbs <- factor(mydata$fbs)
mydata$restecg <- factor(mydata$restecg)
mydata$exang <- factor(mydata$exang)
mydata$slope <- factor(mydata$slope)
mydata$ca <- factor(mydata$ca)
mydata$thal <- factor(mydata$thal)
mydata$disease <- factor(mydata$disease)

summary(mydata)
```

```
##       age        sex     cp         trestbps          chol       fbs
##  Min.   :29.00   0: 96   1: 23   Min.   : 94.0   Min.   :126.0   0:254
##  1st Qu.:48.00   1:201   2: 49   1st Qu.:120.0   1st Qu.:211.0   1: 43
##  Median :56.00           3: 83   Median :130.0   Median :243.0
##  Mean   :54.54           4:142   Mean   :131.7   Mean   :247.4
##  3rd Qu.:61.00                   3rd Qu.:140.0   3rd Qu.:276.0
##  Max.   :77.00                   Max.   :200.0   Max.   :564.0
##  restecg    thalach      exang      oldpeak      slope   ca      thal
##  0:147   Min.   : 71.0   0:200   Min.   :0.000   1:139   0:174   3:164
##  1:  4   1st Qu.:133.0   1: 97   1st Qu.:0.000   2:137   1: 65   6: 18
##  2:146   Median :153.0           Median :0.800   3: 21   2: 38   7:115
##          Mean   :149.6           Mean   :1.056           3: 20
##          3rd Qu.:166.0           3rd Qu.:1.600
##          Max.   :202.0           Max.   :6.200
##  disease
##  0:160
##  1:137
##
##
##
##
```

```
modelo_ajustado <- glm(disease ~ ., data = mydata, family = "binomial")
summary(modelo_ajustado)
```

```
##
## Call:
## glm(formula = disease ~ ., family = "binomial", data = mydata)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -6.253978   2.960399  -2.113 0.034640 *
## age         -0.023508   0.025122  -0.936 0.349402
## sex1         1.670152   0.552486   3.023 0.002503 **
## cp2          1.448396   0.809136   1.790 0.073446 .
## cp3          0.393353   0.700338   0.562 0.574347
## cp4          2.373287   0.709094   3.347 0.000817 ***
## trestbps     0.027720   0.011748   2.359 0.018300 *
```

```
## chol         0.004445   0.004091   1.087 0.277253
## fbs1        -0.574079   0.592539  -0.969 0.332622
## restecg1     1.000887   2.638393   0.379 0.704424
## restecg2     0.486408   0.396327   1.227 0.219713
## thalach     -0.019695   0.011717  -1.681 0.092781 .
## exang1       0.653306   0.447445   1.460 0.144267
## oldpeak      0.390679   0.239173   1.633 0.102373
## slope2       1.302289   0.486197   2.679 0.007395 **
## slope3       0.606760   0.939324   0.646 0.518309
## ca1          2.237444   0.514770   4.346 1.38e-05 ***
## ca2          3.271852   0.785123   4.167 3.08e-05 ***
## ca3          2.188715   0.928644   2.357 0.018428 *
## thal6       -0.168439   0.810310  -0.208 0.835331
## thal7        1.433319   0.440567   3.253 0.001141 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 409.95  on 296  degrees of freedom
## Residual deviance: 183.10  on 276  degrees of freedom
## AIC: 225.1
##
## Number of Fisher Scoring iterations: 6
```

```
predictions <- predict(modelo_ajustado, data = mydata, type = "response")
```

```
predic_grupos <- ifelse(predictions > 0.5, 1, 0)

matriz_confusion <- table(mydata$disease, predic_grupos)

VP <- matriz_confusion[2, 2]
FN <- matriz_confusion[2, 1]
VN <- matriz_confusion[1, 1]
FP <- matriz_confusion[1, 2]

sensibilidad <- VP/(VP+FN)
especificidad <- VN/(VN+FP)
accuracy <- (VP/VN)/(VP+FP+VN+FN)
paste("Accuracy =", accuracy)
```

```
## [1] "Accuracy = 0.00258290669249573"
```

```
paste("Sensibilidad =", sensibilidad)
```

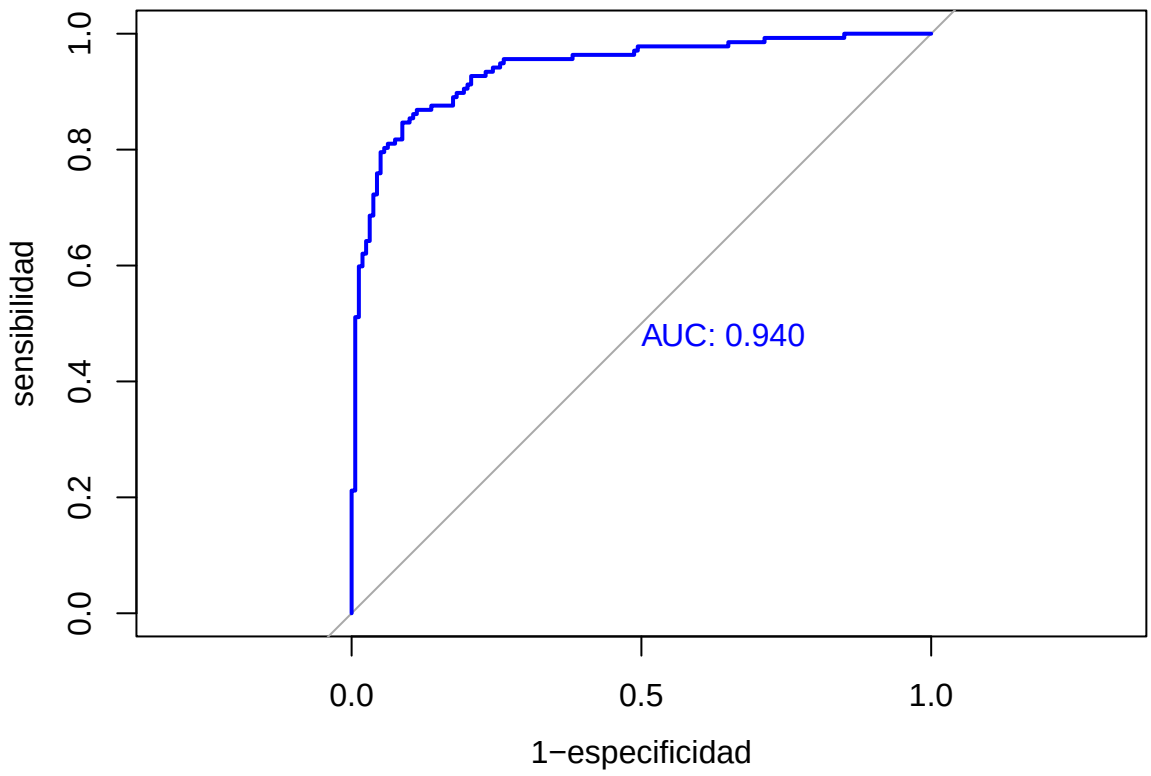
```
## [1] "Sensibilidad = 0.817518248175182"
```

```
paste("Especificidad =", especificidad)
```

```
## [1] "Especificidad = 0.9125"
```

```
library("pROC")
roc(mydata$disease, predictions, plot = TRUE,
    legacy.axes = TRUE, percent = FALSE,
    xlab = "1-especificidad", ylab = "sensibilidad",
    col = "blue", lwd = 2, print.auc = TRUE)
```

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




```
##
## Call:
## roc.default(response = mydata$disease, predictor = predictions,     percent = FALSE, plot = TRUE, leg
##
## Data: predictions in 160 controls (mydata$disease 0) < 137 cases (mydata$disease 1).
## Area under the curve: 0.9395
```