TP8 : Régression Logistique

DJEBALI Wissam

6 mars 2018

Packages : *ggplot2, reshape2, questionr, effects, ROCR*

Description

https://www.listendata.com/2016/02/Logistic-Regression-with-R.html

Méthode utilisée pour prédire le résultat d'une variable qualitative, type binaire (oui/non, malade/pas malade, etc.), qui dépend d'une ou plusieurs autres variables quantitatives qui sont indépendantes entre elles.

La régression logistique est basée sur l'estimation du maximum de vraisemblance (**Maximum Likelihood(ML) Estimation**) : qui veut que les coefficients β_i devant les variables soient choisies de telle sorte qu'ils maximisent la probabilité de Y sachant $X = (X_i)_{i=1,...,k}$ (maximum de vraisemblance). Lors de la recherche avec **ML**, l'ordinateur à travers différentes itérations essaye plusieurs solutions jusqu'à obtenir le maximum de vraisemblance. Le **score de Fisher (Fisher Scoring)** est la plus célébre des méthodes itératives pour estimer les paramètres de la régression.

$$logit(p) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + ... + \beta_k X_k$$

où $logit(p) = log_e(\frac{p}{1-p})$ avec comme équation :

$$p = \frac{1}{1 + e^{-(\beta_0 + \beta_1 X_1 + \beta_2 X_2 + ... + \beta_k X_k)}}$$

p: probabilité que la variable soit égale à "succès" ou à un "événement"

Indicateurs de Performance Importants

1) **Pourcentage de Concordants(Percent of Concordant)**

Pourcentage des paires où les observations avec la modalité désirée(l'événement désirée ;ex : malade, positif, spam...) ont une plus grande probabilité de prediction que les observations avec la modalité opposée (opposé de l'événement désiré; ex : non malade, négatifs, non spam)

Règle : Plus le pourcentage de paires concordantes est élevé meilleur le modèle est adapté. Au delà de 80% le modèle est considéré comme un bon modèle.

2) **Pourcentage de Discordants(Percent Discordant)**

Poourcentage de paires où les observations avec la modalité désirée ont une probabilté de prédiction plus petite que les observations avec la modalité opposée.

3) **Pourcentage d'Ex aequo(Percent Tied)**

Pourcentage de paires où les observations avec la modalité désirée ont la même probabilité que les observations avec la modalité opposée.

4) **Aire sous la courbe ROC (Area under curve (c statistics))**

Compris entre 0.5 et 1, où 0.5 correspond à un modèle qui prédit de façon aléatoire les réponses et 1 à un modèle qui prédit parfaitement les réponses.

$$C = \text{Area under Curve} = \%concordant + (0.5 \times \%tied)$$

Le modèle sera jugé de :

- .90-1 = excellent (A)
- .80-.90 = bon (B)
- .70-.80 = équitable (normal ni trop bon ni trop mauvais) (C)
- .60-.70 = pauvre (D)
- .50-.60 = échec (E)

5) **Matrice de confusion [Classification Table (Confusion Matrix)]**

Sensitivité (Taux de Vrai Positifs)[Sensitivity (True Positive Rate)] % des observations pour lesquelles on a prédit la modalité postive sachant que celle-ci est la modalité correct.

$$Sensitivity = \frac{VRAIPOS}{VRAIPOS + FAUXNEG}$$

Spécificité (Taux de Vrai Négatifs) [Specificity (True Negative Rate)]

% des observations pour lesquelles on a prédit la modalité négative sachant que celle-ci est la modalité correct.

$$Spécificté = \frac{VRAINEG}{VRAINEG + FAUXPOS}$$

Précision [Correct (Accuracy)]=$\frac{\text{Nombre de prédiction correct (VRAI POS + VRAI NEG)}}{\text{Nombre d'observations de l'échantillon}}$

"Cut-off" optimisé sur la base d'un compro- mis entre sensibilité et spécificité

Exemple : Données maladie cardiaque

```
#Read Data File
maladcoeur <-read.delim("C:/Users/DJEBALI/Documents/M2_ISIFAR/Data_Mining/maladcoeur.txt")

mydata<-maladcoeur[,-1]

#Summary
summary(mydata)
```

```
##       sbp           tobacco             ldl           adiposity
##  Min.   :101.0   Min.   : 0.0000   Min.   : 0.980   Min.   : 6.74
##  1st Qu.:124.0   1st Qu.: 0.0525   1st Qu.: 3.283   1st Qu.:19.77
##  Median :134.0   Median : 2.0000   Median : 4.340   Median :26.11
##  Mean   :138.3   Mean   : 3.6356   Mean   : 4.740   Mean   :25.41
##  3rd Qu.:148.0   3rd Qu.: 5.5000   3rd Qu.: 5.790   3rd Qu.:31.23
##  Max.   :218.0   Max.   :31.2000   Max.   :15.330   Max.   :42.49
##      typea          obesity          alcohol            age
##  Min.   :13.0   Min.   :14.70   Min.   :  0.00   Min.   :15.00
```

```
##  1st Qu.:47.0   1st Qu.:22.98   1st Qu.:  0.51   1st Qu.:31.00
##  Median :53.0   Median :25.80   Median :  7.51   Median :45.00
##  Mean   :53.1   Mean   :26.04   Mean   : 17.04   Mean   :42.82
##  3rd Qu.:60.0   3rd Qu.:28.50   3rd Qu.: 23.89   3rd Qu.:55.00
##  Max.   :78.0   Max.   :46.58   Max.   :147.19   Max.   :64.00
##       chd
##  Min.   :0.0000
##  1st Qu.:0.0000
##  Median :0.0000
##  Mean   :0.3463
##  3rd Qu.:1.0000
##  Max.   :1.0000
```

```
#Proportion de malade et non malade
freq(mydata$chd)
```

```
##     n    % val%
## 0 302 65.4 65.4
## 1 160 34.6 34.6
```

```
#ou en utilisant :
table(mydata[,9])
```

```
##
##   0   1
## 302 160
```

```
# Corrélation entre les variables
cc=cor(mydata)

#Affichage des correlation entre les variables avec la fonction melt
melt(cc)
```

```
##         Var1      Var2       value
## 1        sbp       sbp  1.00000000
## 2    tobacco       sbp  0.21224652
## 3        ldl       sbp  0.15829633
## 4  adiposity       sbp  0.35650008
## 5      typea       sbp -0.05745431
## 6    obesity       sbp  0.23806661
## 7    alcohol       sbp  0.14009559
## 8        age       sbp  0.38877060
## 9        chd       sbp  0.19235411
## 10       sbp   tobacco  0.21224652
## 11   tobacco   tobacco  1.00000000
## 12       ldl   tobacco  0.15890546
## 13 adiposity   tobacco  0.28664037
## 14     typea   tobacco -0.01460788
## 15   obesity   tobacco  0.12452941
## 16   alcohol   tobacco  0.20081339
## 17       age   tobacco  0.45033016
## 18       chd   tobacco  0.29971754
## 19       sbp       ldl  0.15829633
## 20   tobacco       ldl  0.15890546
## 21       ldl       ldl  1.00000000
## 22 adiposity       ldl  0.44043175
```

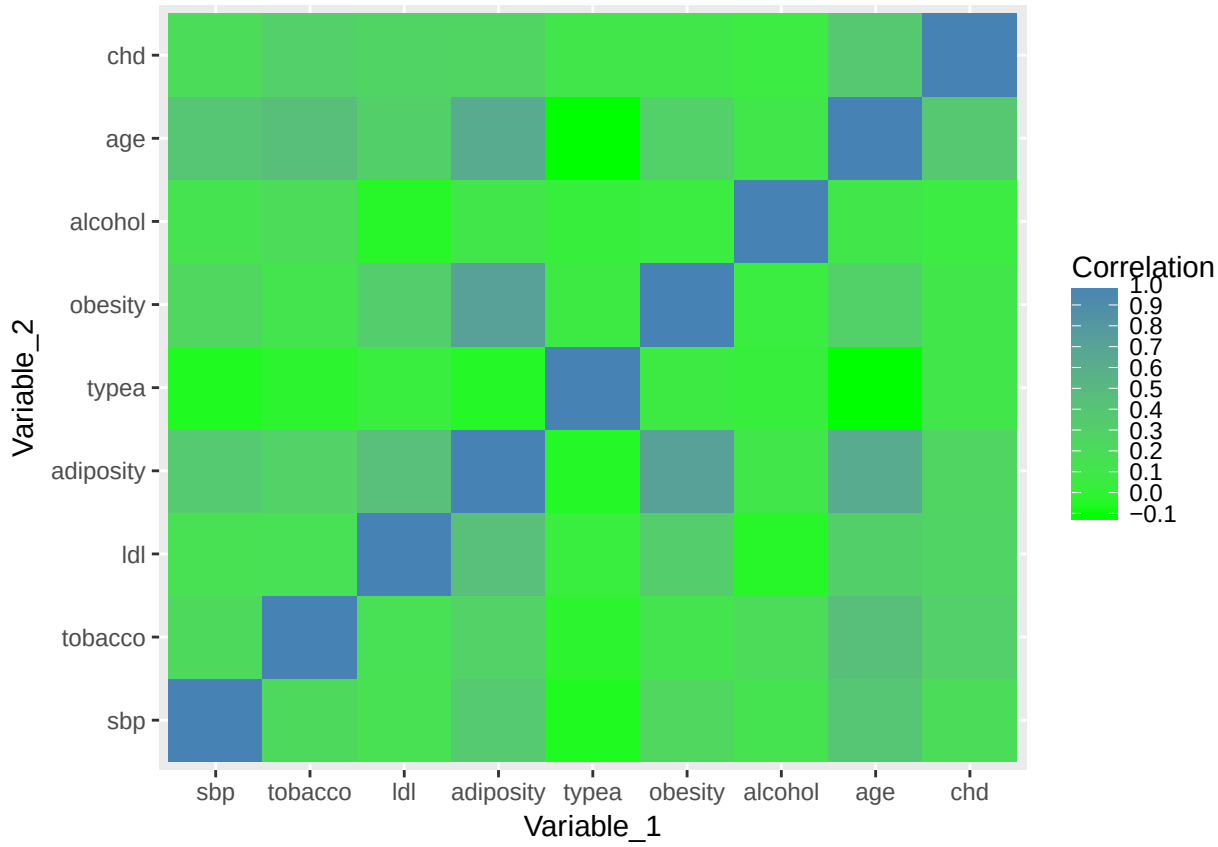
```
## 23     typea       ldl  0.04404758
## 24   obesity       ldl  0.33050586
## 25   alcohol       ldl -0.03340340
## 26       age       ldl  0.31179923
## 27       chd       ldl  0.26305268
## 28       sbp adiposity  0.35650008
## 29   tobacco adiposity  0.28664037
## 30       ldl adiposity  0.44043175
## 31 adiposity adiposity  1.00000000
## 32     typea adiposity -0.04314364
## 33   obesity adiposity  0.71655625
## 34   alcohol adiposity  0.10033013
## 35       age adiposity  0.62595442
## 36       chd adiposity  0.25412139
## 37       sbp     typea -0.05745431
## 38   tobacco     typea -0.01460788
## 39       ldl     typea  0.04404758
## 40 adiposity     typea -0.04314364
## 41     typea     typea  1.00000000
## 42   obesity     typea  0.07400610
## 43   alcohol     typea  0.03949794
## 44       age     typea -0.10260632
## 45       chd     typea  0.10315583
## 46       sbp   obesity  0.23806661
## 47   tobacco   obesity  0.12452941
## 48       ldl   obesity  0.33050586
## 49 adiposity   obesity  0.71655625
## 50     typea   obesity  0.07400610
## 51   obesity   obesity  1.00000000
## 52   alcohol   obesity  0.05161957
## 53       age   obesity  0.29177713
## 54       chd   obesity  0.10009508
## 55       sbp   alcohol  0.14009559
## 56   tobacco   alcohol  0.20081339
## 57       ldl   alcohol -0.03340340
## 58 adiposity   alcohol  0.10033013
## 59     typea   alcohol  0.03949794
## 60   obesity   alcohol  0.05161957
## 61   alcohol   alcohol  1.00000000
## 62       age   alcohol  0.10112465
## 63       chd   alcohol  0.06253068
## 64       sbp       age  0.38877060
## 65   tobacco       age  0.45033016
## 66       ldl       age  0.31179923
## 67 adiposity       age  0.62595442
## 68     typea       age -0.10260632
## 69   obesity       age  0.29177713
## 70   alcohol       age  0.10112465
## 71       age       age  1.00000000
## 72       chd       age  0.37297334
## 73       sbp       chd  0.19235411
## 74   tobacco       chd  0.29971754
## 75       ldl       chd  0.26305268
## 76 adiposity       chd  0.25412139
```

```
## 77     typea       chd  0.10315583
## 78   obesity       chd  0.10009508
## 79   alcohol       chd  0.06253068
## 80       age       chd  0.37297334
## 81       chd       chd  1.00000000
```

Il existe une forte corrélation positive entre les variables:
adiposity et obesity, adiposity et age.
On peut justifier la corrélation entre adiposity et obesity par le fait que l'obésity et du fait de l'accumulation de graisse dans le corp.

`r plot(mydata)` On constate qu'avec l'age le taux de graisse dans les cellules augmente.

```
#Matrice de corrélation
cc.liste = melt(cc)

# On doit changer le nom des étiquettes des variables pour avoir le graphe avec la corrélation
names(cc.liste)=c("Variable_1","Variable_2","Correlation")

#Permet un affichage des corrélations
graph <- ggplot(cc.liste, aes(Variable_1, Variable_2, fill=Correlation)) + geom_tile(aes(fill=Correlati
graph
```




```
# Séparation des données en données d'apprentissage(training) (70%) et données test(validation) (30%)
dt = sort(sample(nrow(mydata),nrow(mydata)*.7))
train<-mydata[dt,]
val<-mydata[-dt,]
```

```r
# Vérification du nombre de ligne dans les données d'apprentissage et les données test
nrow(train)
```

```
## [1] 323
```

```r
nrow(val)
```

```
## [1] 139
```

```r
#Execution de la Régression logistique
mylogistic <- glm(chd ~ ., data = train, family = "binomial")
summary(mylogistic)$coefficient
```

```
##                  Estimate  Std. Error     z value     Pr(>|z|)
## (Intercept) -4.9158327312 1.481645055 -3.31782077 0.0009072268
## sbp          0.0050364699 0.007209845  0.69855454 0.4848304617
## tobacco      0.0808205043 0.032522212  2.48508632 0.0129520023
## ldl          0.2483325857 0.070375639  3.52867257 0.0004176495
## adiposity    0.0431328696 0.033681515  1.28060953 0.2003308509
## typea        0.0239512968 0.013665900  1.75263224 0.0796651537
## obesity     -0.0699428952 0.049091883 -1.42473440 0.1542339902
## alcohol      0.0003577675 0.005873796  0.06090907 0.9514316245
## age          0.0345062681 0.014046441  2.45658441 0.0140264848
```

On regarde la p-value, si c'est <0.05 alors on rejette l'hypothèse de l'indépendance donc les variables sont liées.

```r
#Stepwise Logistic Regression pour réduire le nb de variables dans le modèle
mylogit = step(mylogistic)
```

```
## Start:  AIC=364.74
## chd ~ sbp + tobacco + ldl + adiposity + typea + obesity + alcohol +
##     age
##
##             Df Deviance    AIC
## - alcohol    1   346.74 362.74
## - sbp        1   347.23 363.23
## - adiposity  1   348.41 364.41
## <none>           346.74 364.74
## - obesity    1   348.84 364.84
## - typea      1   349.89 365.89
## - age        1   352.89 368.89
## - tobacco    1   353.50 369.50
## - ldl        1   360.71 376.71
##
## Step:  AIC=362.74
## chd ~ sbp + tobacco + ldl + adiposity + typea + obesity + age
##
##             Df Deviance    AIC
## - sbp        1   347.26 361.26
## - adiposity  1   348.42 362.42
## <none>           346.74 362.74
## - obesity    1   348.85 362.85
## - typea      1   349.89 363.89
## - age        1   352.94 366.94
## - tobacco    1   353.96 367.96
## - ldl        1   360.88 374.88
```

```
##
## Step:  AIC=361.26
## chd ~ tobacco + ldl + adiposity + typea + obesity + age
##
##             Df Deviance    AIC
## - adiposity  1   349.13 361.13
## <none>           347.26 361.26
## - obesity    1   349.30 361.30
## - typea      1   350.38 362.38
## - age        1   354.42 366.42
## - tobacco    1   354.60 366.60
## - ldl        1   361.42 373.42
##
## Step:  AIC=361.13
## chd ~ tobacco + ldl + typea + obesity + age
##
##           Df Deviance    AIC
## - obesity  1   349.50 359.50
## <none>         349.13 361.13
## - typea    1   351.98 361.98
## - tobacco  1   356.27 366.27
## - age      1   365.49 375.49
## - ldl      1   367.29 377.29
##
## Step:  AIC=359.5
## chd ~ tobacco + ldl + typea + age
##
##           Df Deviance    AIC
## <none>         349.50 359.50
## - typea    1   352.26 360.26
## - tobacco  1   356.73 364.73
## - age      1   365.49 373.49
## - ldl      1   367.46 375.46
```

```
# On passe de 9 variables dans le modèle à 4 variables pour expliquer la variable chd

 #Logistic Regression Coefficient
summary.coeff0=summary(mylogit)$coefficient

 #Calculating Odd Ratios
OddRatio = exp(coef(mylogit))
OddRatio
```

```
## (Intercept)     tobacco         ldl       typea         age
## 0.004753935 1.084252124 1.300408114 1.022632644 1.045887895
```

Odd Ratio du Tabac(Valeur exponentielle de l'estimation du Tabac)=1.09 se traduit par une augmentation d'une unité de consommation de tabac les chance d'être atteint d'une maladie cardiaque augmente d'un facteur de 1.09

```
summary.coeff=cbind(Variable=row.names(summary.coeff0), OddRatio, summary.coeff0)
row.names(summary.coeff) = NULL
summary.coeff
```

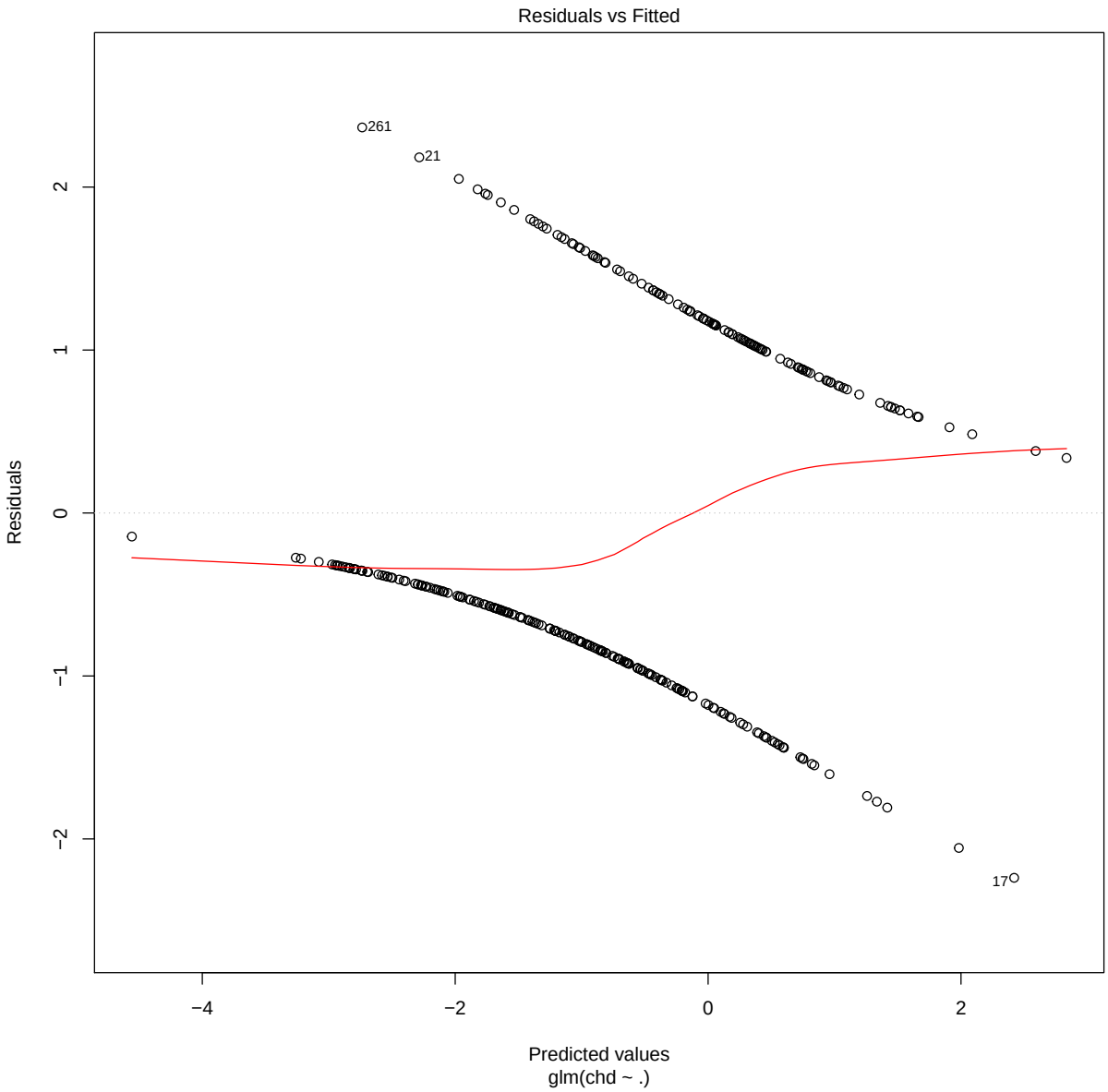
```
##      Variable      OddRatio              Estimate
## [1,] "(Intercept)" "0.00475393545902231" "-5.348782486278"
```

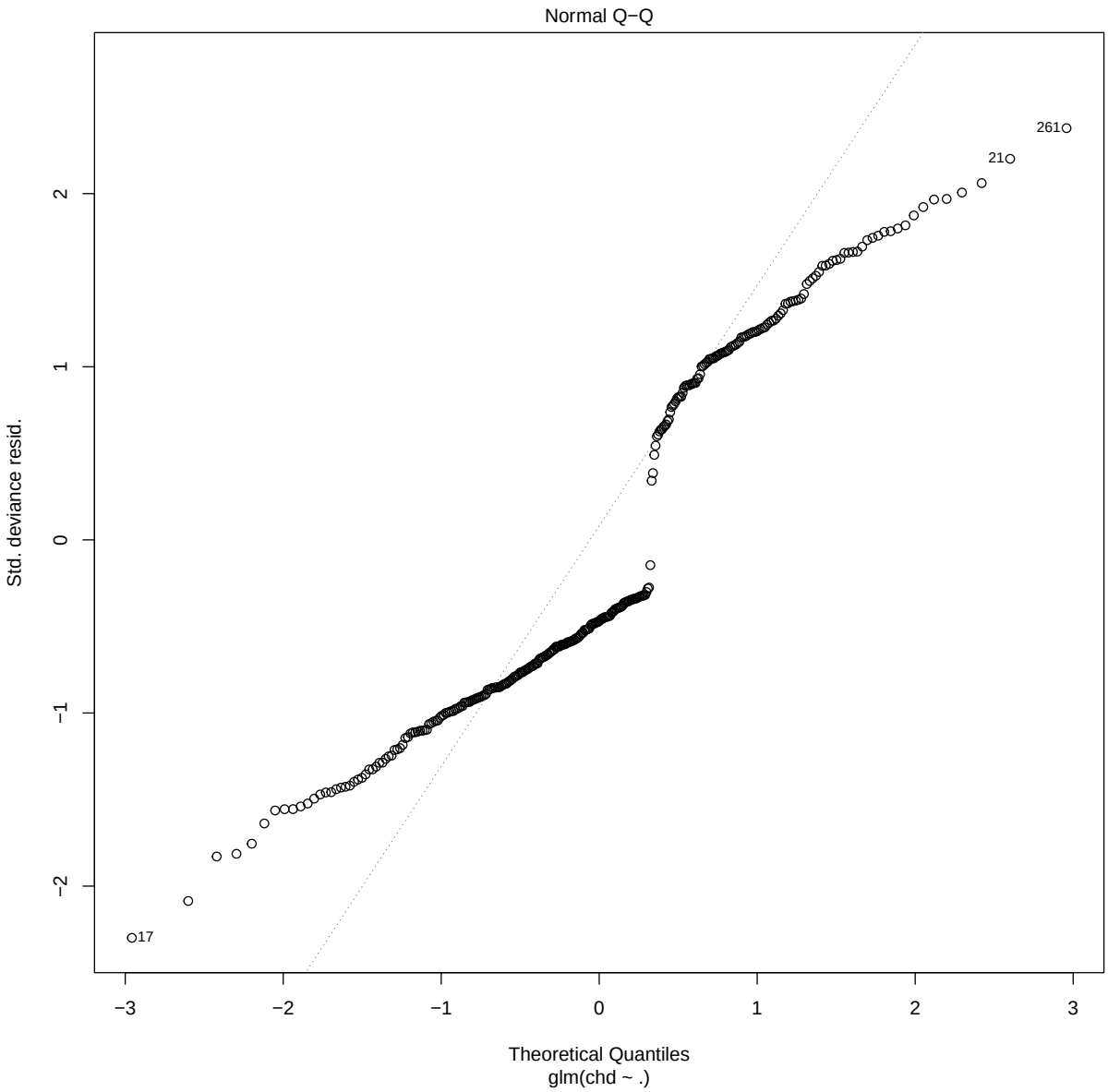
```
## [2,] "tobacco"     "1.08425212407707"    "0.0808904627603074"
## [3,] "ldl"         "1.30040811434585"    "0.262678149312677"
## [4,] "typea"       "1.02263264350251"    "0.0223803252181329"
## [5,] "age"         "1.04588789534132"    "0.0448661852732278"
##      Std. Error           z value             Pr(>|z|)
## [1,] "1.01120907148924"   "-5.28949219017654" "1.22656419115399e-07"
## [2,] "0.0314591056594106" "2.57128933149153"  "0.0101320638915589"
## [3,] "0.0656730030529708" "3.99978891022853"  "6.33990079093851e-05"
## [4,] "0.0136289729433314" "1.64211384901777"  "0.100566412114574"
## [5,] "0.0116138593355661" "3.86315900484781"  "0.000111930099208469"
```

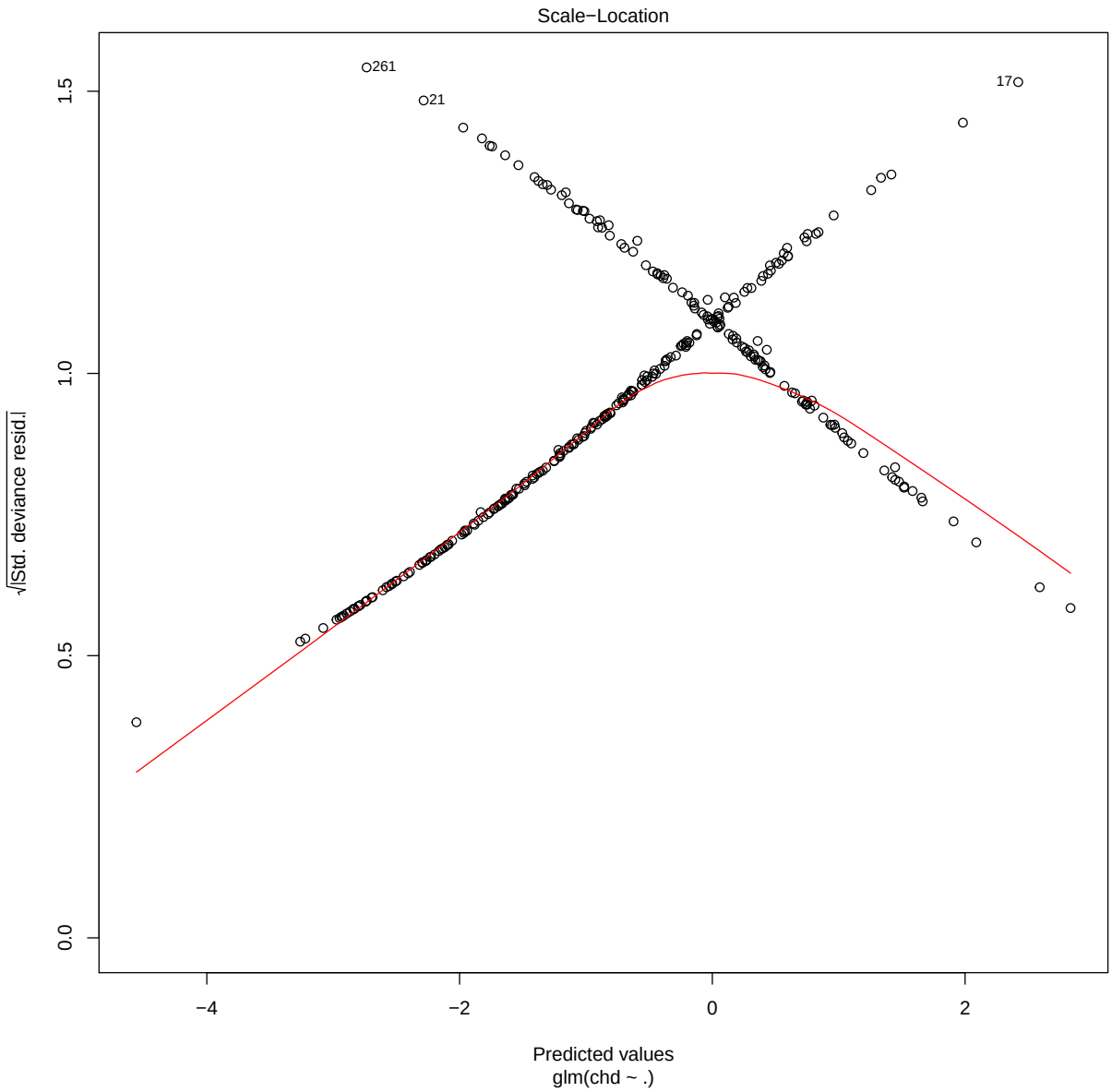
```r
 #R Function : Standardized Coefficients
stdz.coff <- function (regmodel)
{ b <- summary(regmodel)$coef[-1,1]
  sx <- sapply(regmodel$model[-1], sd)
  beta <-(3^(1/2))/pi * sx * b
  return(beta)
}

std.Coeff = data.frame(Standardized.Coeff = stdz.coff(mylogit))
std.Coeff = cbind(Variable=row.names(std.Coeff), std.Coeff)
row.names(std.Coeff) = NULL

 #Final Summary Report
final = merge(summary.coeff, std.Coeff, by = "Variable", all.x = TRUE)

final
```

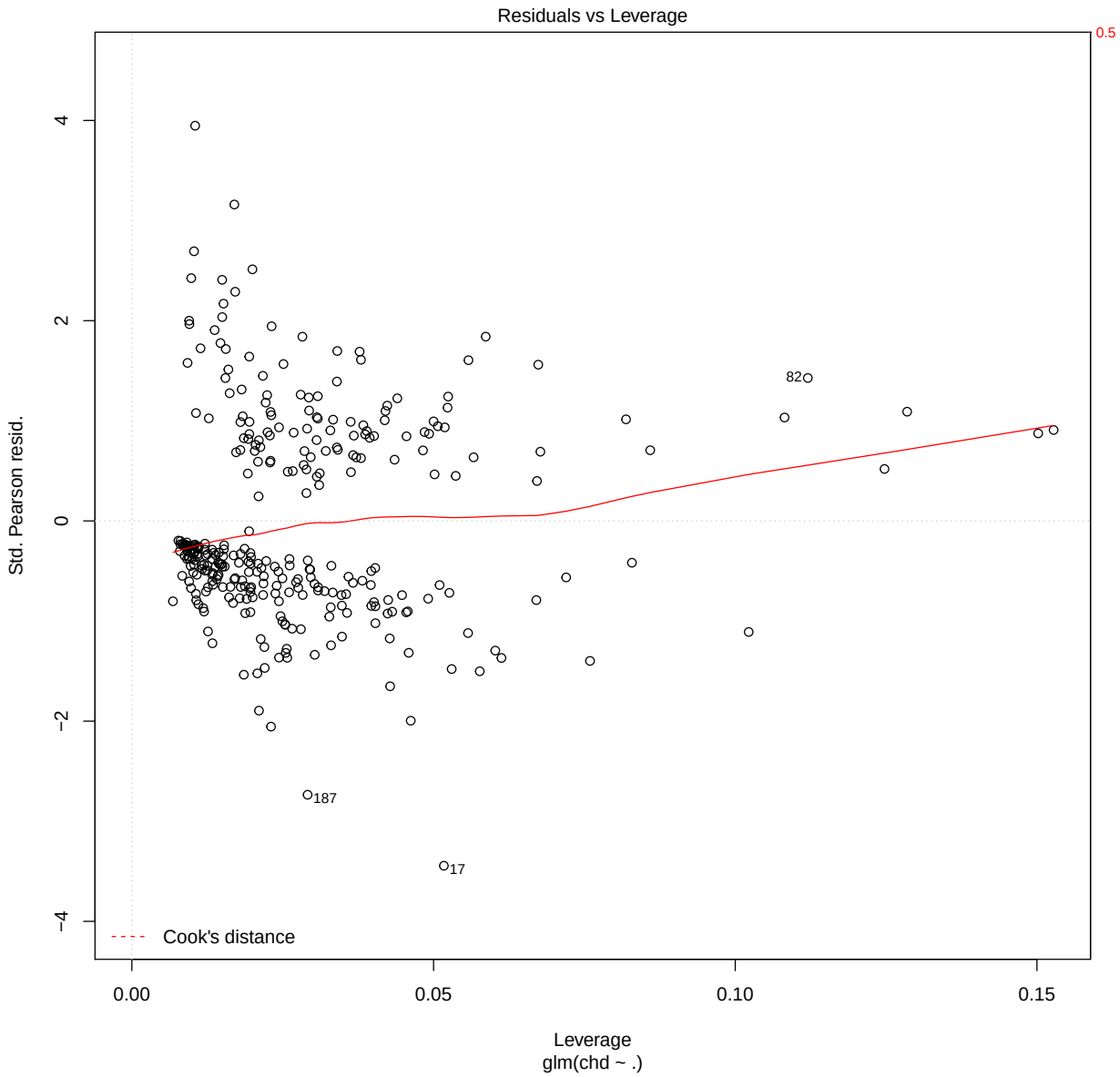
```
##      Variable            OddRatio           Estimate         Std. Error
## 1 (Intercept) 0.00475393545902231    -5.348782486278   1.01120907148924
## 2         age    1.04588789534132 0.0448661852732278 0.0116138593355661
## 3         ldl    1.30040811434585  0.262678149312677 0.0656730030529708
## 4     tobacco    1.08425212407707 0.0808904627603074 0.0314591056594106
## 5       typea    1.02263264350251 0.0223803252181329 0.0136289729433314
##             z value             Pr(>|z|) Standardized.Coeff
## 1 -5.28949219017654 1.22656419115399e-07                NA
## 2  3.86315900484781 0.000111930099208469         0.3589154
## 3  3.99978891022853 6.33990079093851e-05         0.3190484
## 4  2.57128933149153   0.0101320638915589         0.2043657
## 5  1.64211384901777    0.100566412114574         0.1217356
```

```r
plot(mylogistic)
```

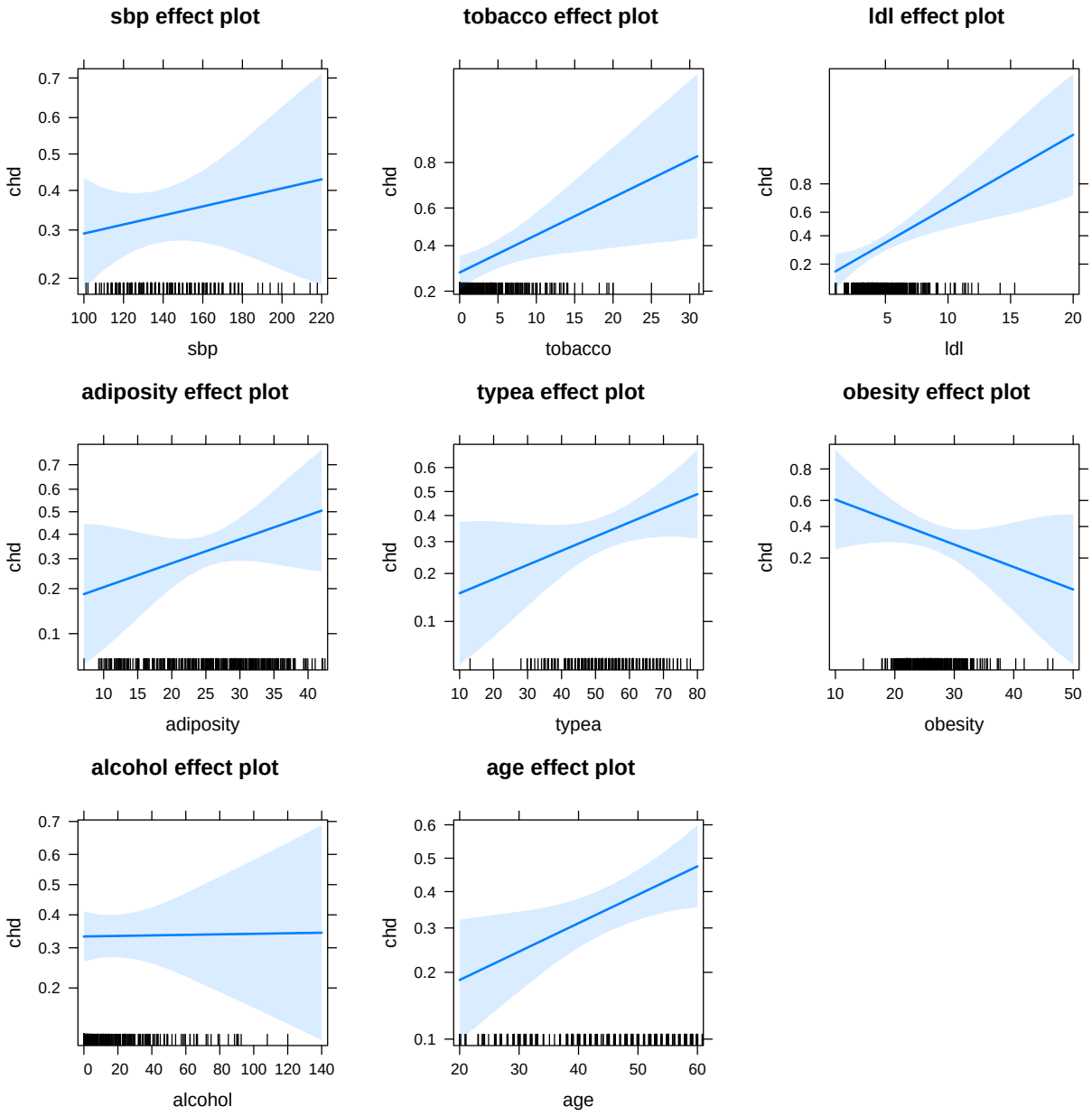

Residuals vs Fitted
261
21
17
Residuals
2
1
0
−1
−2
−4
−2
0
2
Predicted values
glm(chd ~ .)


Normal Q−Q
Std. deviance resid.
Theoretical Quantiles
glm(chd ~ .)
261
21
17
−3
−2
−1
0
1
2
3
−2
−1
0
1
2


Scale−Location
261
21
17
1.5
1.0
0.5
0.0
√|Std. deviance resid.|
−4
−2
0
2
Predicted values
glm(chd ~ .)




```
#L'extension effects propose une représentation graphique résumant les
#effets de chaque variable du modèle
plot(allEffects(mylogistic))
```



Prediction

On fait la prédiction sur l'échantillon test et par la suite verifier si notre réression est correcte.

```
pred = predict(mylogit,val, type = "response")
finaldata = cbind(val, pred)

# Score de performance du model
pred_val <-prediction(pred ,finaldata$chd)

 # Maximum Accuracy and prob. cutoff against it
acc.perf <- performance(pred_val, "acc")
ind = which.max(slot(acc.perf,"y.values")[[1]])
```

```r
acc = slot(acc.perf,"y.values")[[1]][ind]
cutoff = slot(acc.perf,"x.values")[[1]][ind]


# Print Results
print(c(accuracy= acc, cutoff = cutoff))
```

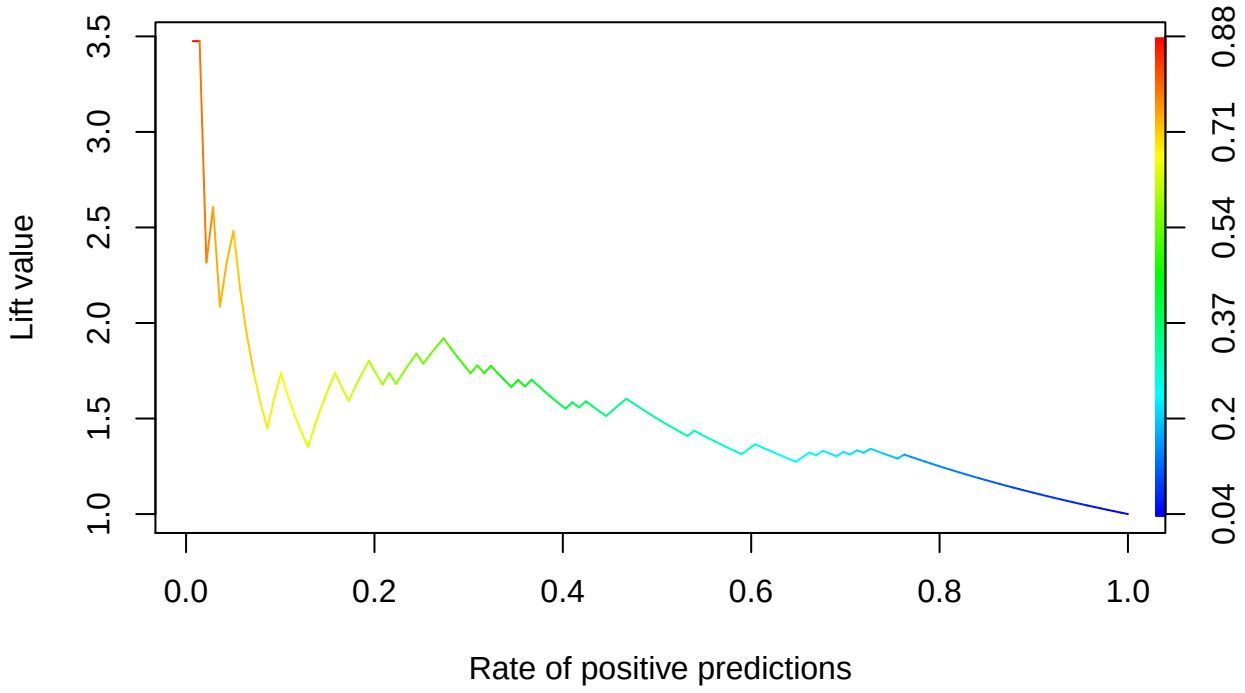
```
##  accuracy cutoff.10
## 0.7410072 0.5242063
```

```r
 # Calcul de l'Area under Curve AUC
perf_val <- performance(pred_val,"auc")
paste(perf_val@y.name,' : ' ,perf_val@y.values)
```

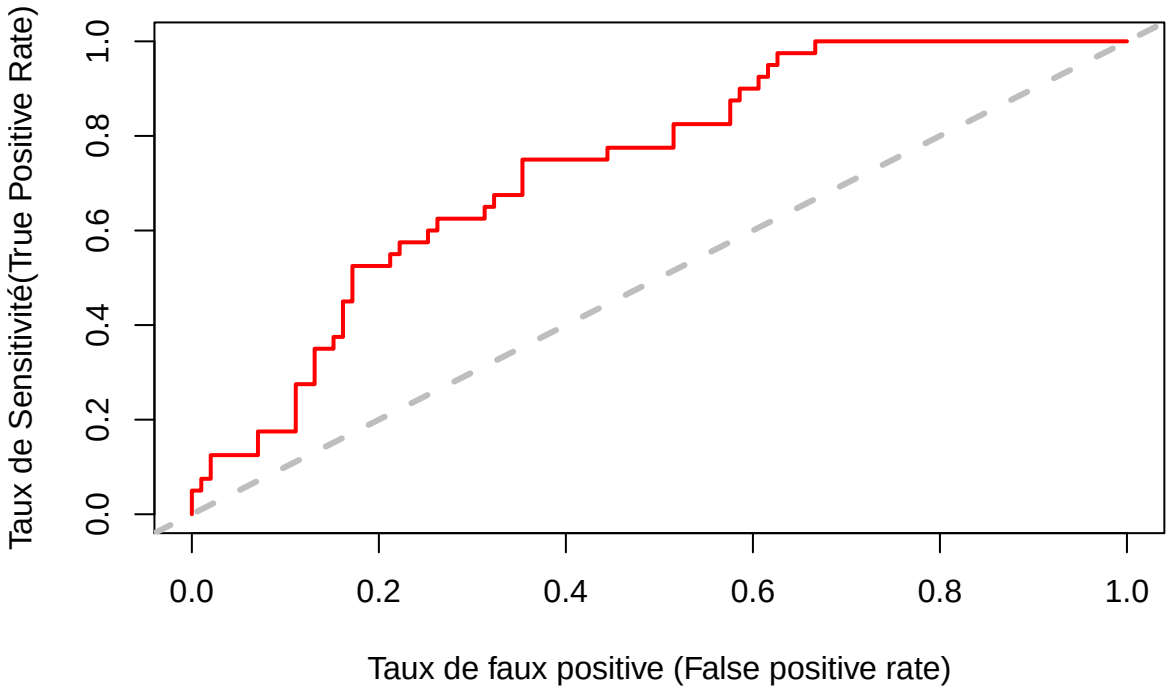
```
## [1] "Area under the ROC curve  :  0.736111111111111"
```

$AUC = 0.74$ de ce fait on a un modèle équitable (normal ni trop bon ni trop mauvais)

```r
 # Plotting Lift curve
plot(performance(pred_val, measure="lift", x.measure="rpp"), colorize=TRUE)
```




```r
 # Affochage de la courbe ROC
perf_val2 <- performance(pred_val, "tpr", "fpr")
plot(perf_val2, col = "red", lwd = 2,xlab='Taux de faux positive (False positive rate)',ylab='Taux de S
abline(a=0,b=1,lwd=3,lty=2,col="gray")
```



C'est une mèthode qui permet de comparer plusieurs mèthodes de classification binaire.

```
 #Calcule de la Statistique KS (KS statistics)
ks1.tree <- max(attr(perf_val2, "y.values")[[1]] - (attr(perf_val2, "x.values")[[1]]))
ks1.tree
```

```
## [1] 0.3964646
```

Comment interpréter ces indicateurs

Un "bon" modèle doit présenter des valeurs faibles de taux d'erreur et de taux de faux positifs (proche de 0) ; des valeurs élevées de sensibilité,précision et spécificité (proche de 1).

Le taux d'erreur est un indicateur symétrique, il donne la même importance aux faux positifs (c) et aux faux négatifs (b).

La sensibilité et la précision sont asymétriques, ils accordent un rôle particulier aux positifs.

Enfin, en règle générale, lorsqu'on oriente l'apprentissage de manière à améliorer la sensibilité, on dégrade souvent la précision et la spécificité. Un modèle qui serait meilleur que les autres sur ces deux groupes de critères antinomiques est celui qu'il faut absolument retenir.

Amélioration du modèle

On peut utiliser la fonction *step* avec l'***AIC*** pour diminuer le nombre de variables dans le modèle et ainsi par la suite avoir en général de meilleurs résultats.