Regressão logistica

Para essa análise de regressão logística serão utilizados dados sobre diabetes

```
dados <- read.csv("pima-indians-diabetes.csv")
names(dados) <- c("number_preg","plasma_gluc","diastolic_pressure","skin_thick","insulin","bmc","diabet
head(dados)
```

```
##   number_preg plasma_gluc diastolic_pressure skin_thick insulin  bmc
## 1           1          85                 66         29       0 26.6
## 2           8         183                 64          0       0 23.3
## 3           1          89                 66         23      94 28.1
## 4           0         137                 40         35     168 43.1
## 5           5         116                 74          0       0 25.6
## 6           3          78                 50         32      88 31.0
##   diabet_pedigree_func age diagno
## 1                0.351  31      0
## 2                0.672  32      1
## 3                0.167  21      0
## 4                2.288  33      1
## 5                0.201  30      0
## 6                0.248  26      1
```

O conjunto de dados será separado em uma parte para treinar o modelo e outra parte para testar o modelo

```
indices <- sample(1:nrow(dados),size = floor(nrow(dados)*.8))
treino <- dados[indices,]
teste <- dados[-indices,]
```

Ajustando o modelo para o conjunto de treino

```
modelo <- glm(diagno~.,data = treino,family = "binomial")
summary(modelo)
```

```
##
## Call:
## glm(formula = diagno ~ ., family = "binomial", data = treino)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.6224  -0.7441  -0.4112   0.7254   2.8173
##
## Coefficients:
##                       Estimate Std. Error z value Pr(>|z|)
## (Intercept)          -8.667140   0.841264 -10.303  < 2e-16 ***
## number_preg           0.113644   0.036678   3.098 0.001946 **
## plasma_gluc           0.034019   0.004165   8.168 3.14e-16 ***
## diastolic_pressure   -0.015351   0.006044  -2.540 0.011085 *
## skin_thick            0.003619   0.007937   0.456 0.648460
## insulin              -0.001685   0.001021  -1.651 0.098783 .
## bmc                   0.106168   0.017710   5.995 2.04e-09 ***
## diabet_pedigree_func  1.126425   0.339341   3.319 0.000902 ***
## age                   0.014028   0.010507   1.335 0.181852
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 790.57  on 612  degrees of freedom
## Residual deviance: 578.66  on 604  degrees of freedom
## AIC: 596.66
##
## Number of Fisher Scoring iterations: 5
```

Algumas das variáveis não foram consideradas significativas e será feita uma seleção de variáveis.

```r
selecao <- step(modelo,scope = list(lower=~1,upper=~.))
```

```
## Start:  AIC=596.66
## diagno ~ number_preg + plasma_gluc + diastolic_pressure + skin_thick +
##     insulin + bmc + diabet_pedigree_func + age
##
##                         Df Deviance    AIC
## - skin_thick             1   578.87 594.87
## - age                    1   580.43 596.43
## <none>                       578.66 596.66
## - insulin                1   581.38 597.38
## - diastolic_pressure     1   585.31 601.31
## - number_preg            1   588.51 604.51
## - diabet_pedigree_func   1   590.22 606.22
## - bmc                    1   621.67 637.67
## - plasma_gluc            1   663.85 679.85
##
## Step:  AIC=594.87
## diagno ~ number_preg + plasma_gluc + diastolic_pressure + insulin +
##     bmc + diabet_pedigree_func + age
##
##                         Df Deviance    AIC
## - age                    1   580.54 594.54
## <none>                       578.87 594.87
## - insulin                1   581.46 595.46
## + skin_thick             1   578.66 596.66
## - diastolic_pressure     1   585.32 599.32
## - number_preg            1   588.79 602.79
## - diabet_pedigree_func   1   590.83 604.83
## - bmc                    1   628.50 642.50
## - plasma_gluc            1   665.78 679.78
##
## Step:  AIC=594.54
## diagno ~ number_preg + plasma_gluc + diastolic_pressure + insulin +
##     bmc + diabet_pedigree_func
##
##                         Df Deviance    AIC
## <none>                       580.54 594.54
## + age                    1   578.87 594.87
## - insulin                1   583.68 595.68
## + skin_thick             1   580.43 596.43
## - diastolic_pressure     1   586.02 598.02
## - diabet_pedigree_func   1   592.95 604.95
## - number_preg            1   600.19 612.19
## - bmc                   1   628.86 640.86
## - plasma_gluc           1   677.61 689.61
```

```
summary(selecao)
```

```
## 
## Call:
## glm(formula = diagno ~ number_preg + plasma_gluc + diastolic_pressure + 
##     insulin + bmc + diabet_pedigree_func, family = "binomial", 
##     data = treino)
## 
## Deviance Residuals: 
##     Min       1Q   Median       3Q      Max  
## -2.6588  -0.7592  -0.4254   0.7288   2.8613  
## 
## Coefficients:
##                       Estimate Std. Error z value Pr(>|z|)    
## (Intercept)          -8.470324   0.824060 -10.279  < 2e-16 ***
## number_preg           0.138096   0.031716   4.354 1.34e-05 ***
## plasma_gluc           0.034745   0.004008   8.669  < 2e-16 ***
## diastolic_pressure   -0.013359   0.005779  -2.312 0.020803 *  
## insulin              -0.001620   0.000916  -1.769 0.076910 .  
## bmc                   0.106602   0.016934   6.295 3.07e-10 ***
## diabet_pedigree_func  1.160021   0.337006   3.442 0.000577 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## (Dispersion parameter for binomial family taken to be 1)
## 
##     Null deviance: 790.57  on 612  degrees of freedom
## Residual deviance: 580.54  on 606  degrees of freedom
## AIC: 594.54
## 
## Number of Fisher Scoring iterations: 5
```

Para avaliar o poder preditivo do modelo

```
(tabela <- table(selecao$fitted.values>.5,treino$diagno))
```

```
## 
##           0   1
##   FALSE 361  95
##   TRUE   40 117
```

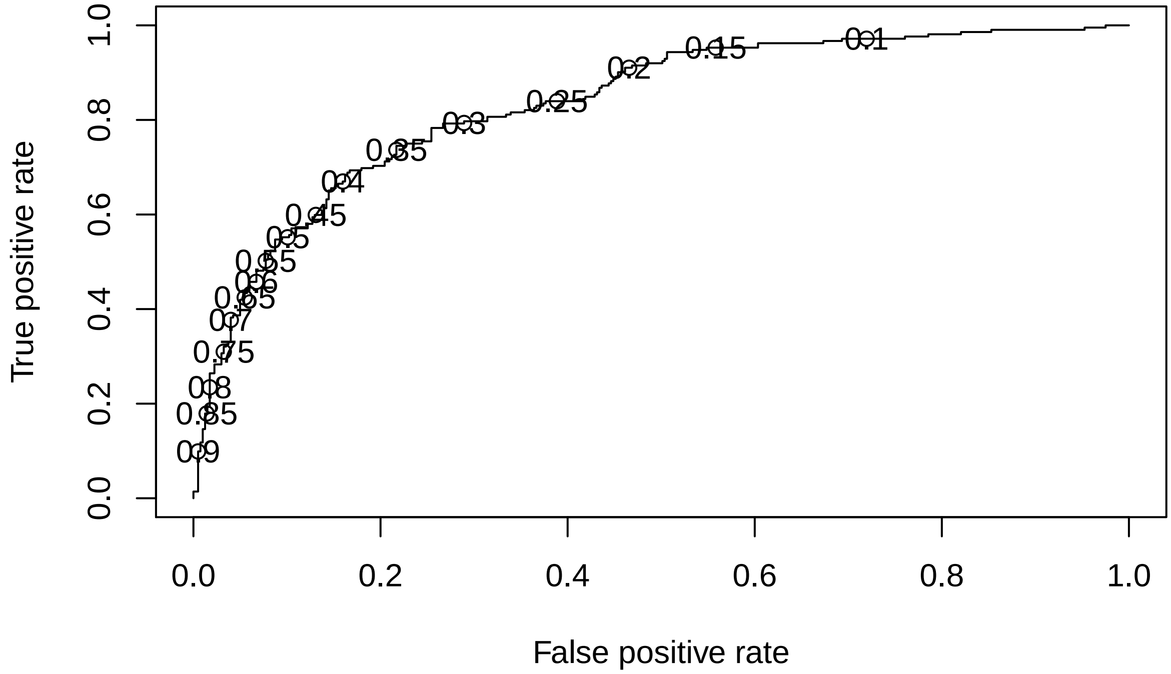
Para avaliar a acurácia

```
sum(diag(tabela))/sum(tabela)
```

```
## [1] 0.7797716
```

Será que o valor limite a ser considerado deve ser o valor de .5?

```
result <- predict(selecao,treino,type = "response")
rocpred <- prediction(result,treino$diagno)
rocperf <- performance(rocpred,"tpr","fpr")
plot(rocperf,print.cutoffs.at=seq(.1,.9,by=.05))
```



Caso seja utilizado o valor de corte como .3

```
(tabela3 <- table(selecao$fitted.values>.3,treino$diagno))
```

```
##
##           0   1
##   FALSE 285  44
##   TRUE  116 168
```

```
sum(diag(tabela3))/sum(tabela3)
```

```
## [1] 0.7389886
```

Avaliando o desempenho do modelo considerando os dados de teste

```
predito <- predict(selecao,teste,type="response")
table(predito>.3,teste$diagno)
```

```
##
##          0  1
##   FALSE 65  9
##   TRUE  34 46
```