

Naive Bayes

W. Grossmann

```
# Naive Bayes Demonstration
# -----
# Used Libraries
# -----
library("e1071")          # naiveBayes

## Warning: package 'e1071' was built under R version 3.2.5

library("klaR")          # NaiveBayes

## Warning: package 'klaR' was built under R version 3.2.5

## Loading required package: MASS

## Warning: package 'MASS' was built under R version 3.2.5

library("class")
library("MASS")
library("foreign")

# -----
# Data
# -----

demo1<- read.csv("SimpleCRM1.csv",header=T,sep=";")
demo1

##      DurCRM Sales User.Type UseS
## 1         10    12         1     1
## 2         24    36         0     1
## 3         28    48         0     1
## 4         45    20         1     1
## 5         30    34         1     1
## 6          3    21         1     1
## 7          1     5         0     0
## 8         23    23         0     0
## 9         12    49         0     0
## 10        35    12         1     0
## 11        33    15         0     0
## 12        12    25         1    NA

attach(demo1)
#-----
# Naive Bayes without normal density
#-----
predictors<-demo1[,-4]
```

```

res <- NaiveBayes(factor(UseS) ~ ., data = predictors)
res

## $apriori
## grouping
##      0      1
## 0.4166667 0.5000000
##
## $tables
## $tables$DurCRM
##      [,1]      [,2]
## 0 20.80000 14.35967
## 1 23.33333 15.01555
##
## $tables$Sales
##      [,1]      [,2]
## 0 20.8 17.03526
## 1 28.5 13.17194
##
## $tables$User.Type
##      [,1]      [,2]
## 0 0.2000000 0.4472136
## 1 0.6666667 0.5163978
##
##
## $levels
## [1] "0" "1"
##
## $call
## NaiveBayes.default(x = X, grouping = Y)
##
## $x
##      DurCRM Sales User.Type
## 1      10    12         1
## 2      24    36         0
## 3      28    48         0
## 4      45    20         1
## 5      30    34         1
## 6       3    21         1
## 7       1     5         0
## 8      23    23         0
## 9      12    49         0
## 10     35    12         1
## 11     33    15         0
## 12     12    25         1
##
## $usekernel
## [1] FALSE
##
## $varnames
## [1] "DurCRM" "Sales" "User.Type"

```

```

##
## attr(,"class")
## [1] "NaiveBayes"

predict(res,demo1[,-4])

## $class
## [1] 1 0 0 1 1 1 0 0 0 1 0 1
## Levels: 0 1
##
## $posterior
##           0           1
## [1,] 0.2931546 0.7068454
## [2,] 0.5553987 0.4446013
## [3,] 0.5562670 0.4437330
## [4,] 0.1400834 0.8599166
## [5,] 0.1232088 0.8767912
## [6,] 0.2088179 0.7911821
## [7,] 0.8580480 0.1419520
## [8,] 0.6340582 0.3659418
## [9,] 0.6035594 0.3964406
## [10,] 0.2352998 0.7647002
## [11,] 0.6890942 0.3109058
## [12,] 0.1764323 0.8235677

predict(res, demo1, type= "raw")

## $class
## [1] 1 0 0 1 1 1 0 0 0 1 0 1
## Levels: 0 1
##
## $posterior
##           0           1
## [1,] 0.2931546 0.7068454
## [2,] 0.5553987 0.4446013
## [3,] 0.5562670 0.4437330
## [4,] 0.1400834 0.8599166
## [5,] 0.1232088 0.8767912
## [6,] 0.2088179 0.7911821
## [7,] 0.8580480 0.1419520
## [8,] 0.6340582 0.3659418
## [9,] 0.6035594 0.3964406
## [10,] 0.2352998 0.7647002
## [11,] 0.6890942 0.3109058
## [12,] 0.1764323 0.8235677

pred<-predict(res,predictors)
pred<-predict(res,demo1)
ls(pred)

## [1] "class"      "posterior"

table(pred$class, demo1[,4])

```

```

##
##      0 1
##     0 4 2
##     1 1 4

#-----
#NaiveBayes with kernel density estimation
#-----

predictors<-demo1[,-4]
res1 <- NaiveBayes(factor(UseS) ~ ., data = predictors,
                   usekernel = TRUE)
pred1<-predict(res1,predictors)
pred1<-predict(res1,demo1)
ls(pred1)

## [1] "class"      "posterior"

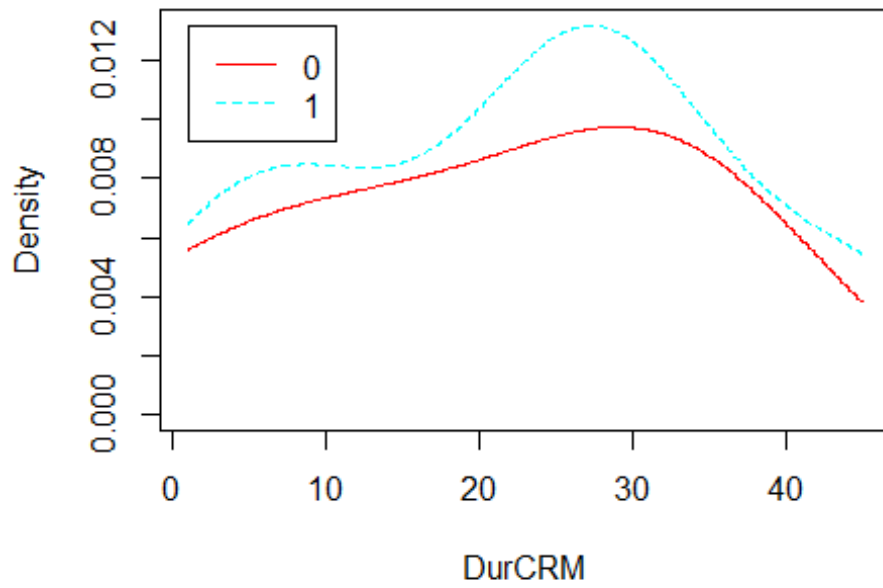
table(pred1$class, demo1[,4])

##
##      0 1
##     0 4 1
##     1 1 5

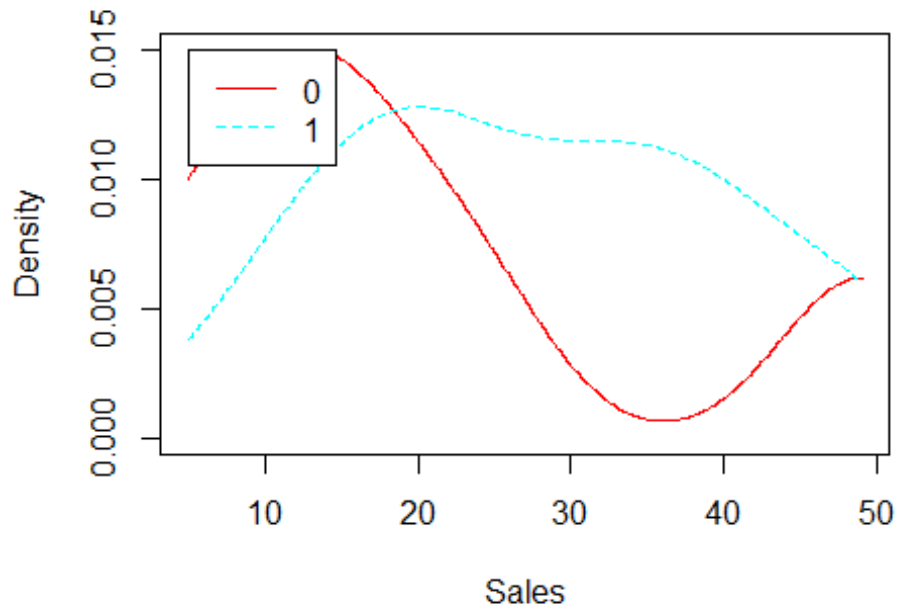
plot(res1)

```

Naive Bayes Plot



Naive Bayes Plot



Naive Bayes Plot

