Assignment 3

Priyanka Jonnala

2023-10-16

Summary

1.Using the information in this dataset, if an accident has just been reported and no further information is available, what should the prediction be? (INJURY = Yes or No?) Why?

Ans : Using the information in the dataset, the prediction is *Injury = Yes*. This is beacuse Injury = yes is 0.50878 and Injury = No is 0.49121. Since the probability of Injury = yes is greater than the Injury = No, I predict that Injury = Yes.

2.Select the first 24 records in the dataset and look only at the response (INJURY) and the two predictors WEATHER_R and TRAF_CON_R. Create a pivot table that examines INJURY as a function of the two predictors for these 24 records. Use all three variables in the pivot table as rows/columns. 1.Compute the exact Bayes conditional probabilities of an injury (INJURY = Yes) given the six possible combinations of the predictors.

Ans : Probability of predictions

TRAF_CON_R = 1 and WEATHER_R = 0.6666667

TRAF_CON_R = 0 and WEATHER_R = 2, 0.1818182.

TRAF_CON_R = 1 and WEATHER_R = 1 0.0000000

TRAF_CON_R = 1 and WEATHER_R = 2 0.0000000

TRAF_CON_R = 2 and WEATHER_R = 1 0.0000000

TRAF_CON_R = 2 and WEATHER_R = 2 1.0000000

2.2.Classify the 24 accidents using these probabilities and a cutoff of 0.5.

Ans : [1] 0.6666667 0.1818182 0.0000000 0.0000000 0.6666667 0.1818182 [7] 0.1818182 0.6666667 0.1818182 0.1818182 0.1818182 0.0000000 [13] 0.6666667 0.6666667 0.6666667 0.6666667 0.1818182 0.1818182 [19] 0.1818182 0.1818182 0.6666667 0.6666667 1.0000000 0.1818182

Qunatitative Predictions:

"yes" "no" "no" "no" "yes" "no" "no" "yes" "no" "no" "no" "no" "yes" "yes" "yes" "yes" "no" "no" "no" "no" "yes" "yes" "yes" "no"

2.3.Compute manually the naive Bayes conditional probability of an injury given WEATHER_R = 1 and TRAF_CON_R = 1.

Ans: The result of naive bayes conditional probability of an injury given WEATHER_R = 1 and TRAF_CON_R =1 is "0".

2.4.Run a naive Bayes classifier on the 24 records and two predictors. Check the model output to obtain probabilities and classifications for all 24 records. Compare this to the exact Bayes classification. Are the resulting classifications equivalent? Is the ranking (= ordering) of observations equivalent?

Ans : Refer to line 160 to 176.

3.Let us now return to the entire dataset. Partition the data into training (60%) and validation (40%). 1.Run a naive Bayes classifier on the complete training set with the relevant predictors (and INJURY as the response). Note that all predictors are categorical. Show the confusion matrix.

Confusion Matrix and Statistics

```
      Reference
```

Prediction no yes no 5097 7405 yes 4230 8577

```
           Accuracy : 0.5403
```

3.2.What is the overall error of the validation set?

The overall error of the validation set is 0.4668721

Load the required libraries and read the input file

```r
library(e1071)
library(caret)
```

```
## Loading required package: ggplot2

## Loading required package: lattice
```

```r
accidents <- read.csv("C:\\Users\\priya\\OneDrive\\Desktop\\FML\\Assignment 3\\accidentsFull.csv")
accidents$INJURY = ifelse(accidents$MAX_SEV_IR>0,"yes","no")
```

```r
# Convert variables to factor
for (i in c(1:dim(accidents)[2])){
  accidents[,i] <- as.factor(accidents[,i])
}
head(accidents,n=24)
```

```
##    HOUR_I_R ALCHL_I ALIGN_I STRATUM_R WRK_ZONE WKDY_I_R INT_HWY LGTCON_I_R
## 1         0       2       2         1        0        1       0          3
## 2         1       2       1         0        0        1       1          3
## 3         1       2       1         0        0        1       0          3
## 4         1       2       1         1        0        0       0          3
## 5         1       1       1         0        0        1       0          3
## 6         1       2       1         1        0        1       0          3
## 7         1       2       1         0        0        1       1          3
## 8         1       2       1         1        0        1       0          3
## 9         1       2       1         1        0        1       0          3
## 10        0       2       1         0        0        0       0          3
## 11        1       2       1         0        0        1       0          3
## 12        1       2       1         1        0        1       0          3
## 13        1       2       1         1        0        1       0          3
## 14        1       2       2         0        0        1       0          3
## 15        1       2       2         1        0        1       0          3
## 16        1       2       2         1        0        1       0          3
## 17        1       2       1         1        0        1       0          3
## 18        1       2       1         1        0        0       0          3
```

```
## 19         1        2        1          1         0         1        0          3
## 20         1        2        1          0         0         1        0          3
## 21         1        2        1          1         0         1        0          3
## 22         1        2        2          0         0         1        0          3
## 23         1        2        1          0         0         1        0          3
## 24         1        2        1          1         0         1        9          3
##    MANCOL_I_R PED_ACC_R RELJCT_I_R REL_RWY_R PROFIL_I_R SPD_LIM SUR_COND
## 1           0         0          1         0          1      40        4
## 2           2         0          1         1          1      70        4
## 3           2         0          1         1          1      35        4
## 4           2         0          1         1          1      35        4
## 5           2         0          0         1          1      25        4
## 6           0         0          1         0          1      70        4
## 7           0         0          0         0          1      70        4
## 8           0         0          0         0          1      35        4
## 9           0         0          1         0          1      30        4
## 10          0         0          1         0          1      25        4
## 11          0         0          0         0          1      55        4
## 12          2         0          0         1          1      40        4
## 13          1         0          0         1          1      40        4
## 14          0         0          0         0          1      25        4
## 15          0         0          0         0          1      35        4
## 16          0         0          0         0          1      45        4
## 17          0         0          0         0          1      20        4
## 18          0         0          0         0          1      50        4
## 19          0         0          0         0          1      55        4
## 20          0         0          1         1          1      55        4
## 21          0         0          1         0          0      45        4
## 22          0         0          1         0          0      65        4
## 23          0         0          0         0          0      65        4
## 24          2         0          1         1          0      55        4
##    TRAF_CON_R TRAF_WAY VEH_INVL WEATHER_R INJURY_CRASH NO_INJ_I PRPTYDMG_CRASH
## 1           0        3        1         1            1        1              0
## 2           0        3        2         2            0        0              1
## 3           1        2        2         2            0        0              1
## 4           1        2        2         1            0        0              1
## 5           0        2        3         1            0        0              1
## 6           0        2        1         2            1        1              0
## 7           0        2        1         2            0        0              1
## 8           0        1        1         1            1        1              0
## 9           0        1        1         2            0        0              1
## 10          0        1        1         2            0        0              1
## 11          0        1        1         2            0        0              1
## 12          2        1        2         1            0        0              1
## 13          0        1        4         1            1        2              0
## 14          0        1        1         1            0        0              1
## 15          0        1        1         1            1        1              0
## 16          0        1        1         1            1        1              0
## 17          0        1        1         2            0        0              1
## 18          0        1        1         2            0        0              1
## 19          0        1        1         2            0        0              1
## 20          0        1        1         2            0        0              1
## 21          0        3        1         1            1        1              0
## 22          0        3        1         1            0        0              1
```

```
## 23          2        2        1         2            1        2              0
## 24          0        2        2         2            1        1              0
##    FATALITIES MAX_SEV_IR INJURY
## 1           0          1    yes
## 2           0          0     no
## 3           0          0     no
## 4           0          0     no
## 5           0          0     no
## 6           0          1    yes
## 7           0          0     no
## 8           0          1    yes
## 9           0          0     no
## 10          0          0     no
## 11          0          0     no
## 12          0          0     no
## 13          0          1    yes
## 14          0          0     no
## 15          0          1    yes
## 16          0          1    yes
## 17          0          0     no
## 18          0          0     no
## 19          0          0     no
## 20          0          0     no
## 21          0          1    yes
## 22          0          0     no
## 23          0          1    yes
## 24          0          1    yes
```

```r
p_yes <- mean(accidents$INJURY == "yes")
p_no <- mean(accidents$INJURY == "no")
p_yes
```

```
## [1] 0.5087832
```

```r
p_no
```

```
## [1] 0.4912168
```

```r
accidents24 <- accidents[1:24,c("INJURY","WEATHER_R","TRAF_CON_R")]
#head(accidents24)
```

```r
dt1 <- ftable(accidents24)
dt2 <- ftable(accidents24[,-1]) # print table only for conditions
dt1
```

```
##                  TRAF_CON_R 0 1 2
## INJURY WEATHER_R
## no     1                    3 1 1
##        2                    9 1 0
## yes    1                    6 0 0
##        2                    2 0 1
```

```r
dt2
```

```
##           TRAF_CON_R  0  1  2
## WEATHER_R
## 1                     9  1  1
## 2                    11  1  1
```

```r
# Injury = yes
p1 = dt1[3,1] / dt2[1,1] # Injury, Weather=1 and Traf=0
p2 = dt1[4,1] / dt2[2,1] # Injury, Weather=2, Traf=0
p3 = dt1[3,2] / dt2[1,2] # Injury, W=1, T=1
p4 = dt1[4,2] / dt2[2,2] # I, W=2,T=1
p5 = dt1[3,3] / dt2[1,3] # I, W=1,T=2
p6 = dt1[4,3]/ dt2[2,3] #I,W=2,T=2

# Injury = no
n1 = dt1[1,1] / dt2[1,1] # Weather=1 and Traf=0
n2 = dt1[2,1] / dt2[2,1] # Weather=2, Traf=0
n3 = dt1[1,2] / dt2[1,2] # W=1, T=1
n4 = dt1[2,2] / dt2[2,2] # W=2,T=1
n5 = dt1[1,3] / dt2[1,3] # W=1,T=2
n6 = dt1[2,3] / dt2[2,3] # W=2,T=2
print(c(p1,p2,p3,p4,p5,p6))
```

```
## [1] 0.6666667 0.1818182 0.0000000 0.0000000 0.0000000 1.0000000
```

```r
print(c(n1,n2,n3,n4,n5,n6))
```

```
## [1] 0.3333333 0.8181818 1.0000000 1.0000000 1.0000000 0.0000000
```

2. Let us now compute Classify the 24 accidents using these probabilities and a cutoff of 0.5.

```r
prob.inj <- rep(0,24)

for (i in 1:24) {
  print(c(accidents24$WEATHER_R[i],accidents24$TRAF_CON_R[i]))
    if (accidents24$WEATHER_R[i] == "1") {
      if (accidents24$TRAF_CON_R[i]=="0"){
        prob.inj[i] = p1
      }
      else if (accidents24$TRAF_CON_R[i]=="1") {
        prob.inj[i] = p3
      }
      else if (accidents24$TRAF_CON_R[i]=="2") {
        prob.inj[i] = p5
      }
    }
    else {
      if (accidents24$TRAF_CON_R[i]=="0"){
        prob.inj[i] = p2
      }
```

```r
      else if (accidents24$TRAF_CON_R[i]=="1") {
        prob.inj[i] = p4
      }
      else if (accidents24$TRAF_CON_R[i]=="2") {
        prob.inj[i] = p6
      }
    }
  }
```

```
## [1] 1 0
## Levels: 1 2 0
## [1] 2 0
## Levels: 1 2 0
## [1] 2 1
## Levels: 1 2 0
## [1] 1 1
## Levels: 1 2 0
## [1] 1 0
## Levels: 1 2 0
## [1] 2 0
## Levels: 1 2 0
## [1] 2 0
## Levels: 1 2 0
## [1] 1 0
## Levels: 1 2 0
## [1] 2 0
## Levels: 1 2 0
## [1] 2 0
## Levels: 1 2 0
## [1] 2 0
## Levels: 1 2 0
## [1] 1 2
## Levels: 1 2 0
## [1] 1 0
## Levels: 1 2 0
## [1] 1 0
## Levels: 1 2 0
## [1] 1 0
## Levels: 1 2 0
## [1] 1 0
## Levels: 1 2 0
## [1] 2 0
## Levels: 1 2 0
## [1] 2 0
## Levels: 1 2 0
## [1] 2 0
## Levels: 1 2 0
## [1] 2 0
## Levels: 1 2 0
## [1] 1 0
## Levels: 1 2 0
## [1] 1 0
## Levels: 1 2 0
```

```
## [1] 2 2
## Levels: 1 2 0
## [1] 2 0
## Levels: 1 2 0
```

```
accidents24$prob.inj <- prob.inj
accidents24$prob.inj
```

```
##  [1] 0.6666667 0.1818182 0.0000000 0.0000000 0.6666667 0.1818182 0.1818182
##  [8] 0.6666667 0.1818182 0.1818182 0.1818182 0.0000000 0.6666667 0.6666667
## [15] 0.6666667 0.6666667 0.1818182 0.1818182 0.1818182 0.1818182 0.6666667
## [22] 0.6666667 1.0000000 0.1818182
```

```
accidents24$pred.prob <- ifelse(accidents24$prob.inj>0.5, "yes", "no")
accidents24$pred.prob
```

```
##  [1] "yes" "no"  "no"  "no"  "yes" "no"  "no"  "yes" "no"  "no"  "no"  "no"
## [13] "yes" "yes" "yes" "yes" "no"  "no"  "no"  "no"  "yes" "yes" "yes" "no"
```

Compute manually the naive Bayes conditional probability of an injury given WEATHER_R = 1 and TRAF_CON_R = 1.

```
P_W1_IY = (dt1[3,1]+dt1[3,2]+dt1[3,3])/(dt1[3,1]+dt1[3,2]+dt1[3,3]+dt1[4,1]+dt1[4,2]+dt1[4,3])
P_T1_IY = (dt1[3,2]+dt1[4,2])/(dt1[3,1]+dt1[3,2]+dt1[3,3]+dt1[4,1]+dt1[4,2]+dt1[4,3])
PIY     = (dt1[3,1]+dt1[3,2]+dt1[3,3]+dt1[4,1]+dt1[4,2]+dt1[4,3])/24
P_W1_IN = (dt1[1,1]+dt1[1,2]+dt1[1,3])/(dt1[1,1]+dt1[1,2]+dt1[1,3]+dt1[2,1]+dt1[2,2]+dt1[2,3])
P_T1_IN = (dt1[1,2]+dt1[2,2])/(dt1[1,1]+dt1[1,2]+dt1[1,3]+dt1[2,1]+dt1[2,2]+dt1[2,3])
PIN    = (dt1[1,1]+dt1[1,2]+dt1[1,3]+dt1[2,1]+dt1[2,2]+dt1[2,3])/24

P_IY_W1.T1= (P_W1_IY*P_T1_IY*PIY)/((P_W1_IY*P_T1_IY*PIY)+(P_W1_IN*P_T1_IN*PIN))
P_IY_W1.T1
```

```
## [1] 0
```

Run a naive Bayes classifier on the 24 records and two predictors. Check the model output to obtain probabilities and classifications for all 24 records. Compare this to the exact Bayes classification. Are the resulting classifications equivalent? Is the ranking (= ordering) of observations equivalent?

```
nb <- naiveBayes(INJURY ~ TRAF_CON_R + WEATHER_R,
                 data = accidents24)

nbt <- predict(nb, newdata = accidents24,type = "raw")
accidents24$nbpred.prob <- nbt[,2] # Transfer the "Yes" nb prediction
print(nb)
```

```
##
## Naive Bayes Classifier for Discrete Predictors
##
## Call:
## naiveBayes.default(x = X, y = Y, laplace = laplace)
##
```

```
## A-priori probabilities:
## Y
##    no   yes
## 0.625 0.375
##
## Conditional probabilities:
##      TRAF_CON_R
## Y              0          1          2
##   no  0.80000000 0.13333333 0.06666667
##   yes 0.88888889 0.00000000 0.11111111
##
##      WEATHER_R
## Y             1         2
##   no  0.3333333 0.6666667
##   yes 0.6666667 0.3333333
```

Let us use caret

```r
library(klaR)
```

```
## Loading required package: MASS
```

```r
nb2 <- train(INJURY ~ TRAF_CON_R + WEATHER_R,
      data = accidents24, method = "nb")
```

```
## Warning: model fit failed for Resample01: usekernel=FALSE, fL=0, adjust=1 Error in NaiveBayes.defaul
##   Zero variances for at least one class in variables: TRAF_CON_R1

## Warning: model fit failed for Resample02: usekernel=FALSE, fL=0, adjust=1 Error in NaiveBayes.defaul
##   Zero variances for at least one class in variables: TRAF_CON_R1

## Warning: model fit failed for Resample03: usekernel=FALSE, fL=0, adjust=1 Error in NaiveBayes.defaul
##   Zero variances for at least one class in variables: TRAF_CON_R1

## Warning: model fit failed for Resample04: usekernel=FALSE, fL=0, adjust=1 Error in NaiveBayes.defaul
##   Zero variances for at least one class in variables: TRAF_CON_R1, TRAF_CON_R2

## Warning: model fit failed for Resample05: usekernel=FALSE, fL=0, adjust=1 Error in NaiveBayes.defaul
##   Zero variances for at least one class in variables: TRAF_CON_R1, TRAF_CON_R2, WEATHER_R2

## Warning: model fit failed for Resample06: usekernel=FALSE, fL=0, adjust=1 Error in NaiveBayes.defaul
##   Zero variances for at least one class in variables: TRAF_CON_R1, TRAF_CON_R2

## Warning: model fit failed for Resample07: usekernel=FALSE, fL=0, adjust=1 Error in NaiveBayes.defaul
##   Zero variances for at least one class in variables: TRAF_CON_R1, TRAF_CON_R2

## Warning: model fit failed for Resample08: usekernel=FALSE, fL=0, adjust=1 Error in NaiveBayes.defaul
##   Zero variances for at least one class in variables: TRAF_CON_R1

## Warning: model fit failed for Resample09: usekernel=FALSE, fL=0, adjust=1 Error in NaiveBayes.defaul
##   Zero variances for at least one class in variables: TRAF_CON_R1, TRAF_CON_R2
```

```
## Warning: model fit failed for Resample10: usekernel=FALSE, fL=0, adjust=1 Error in NaiveBayes.defaul
##   Zero variances for at least one class in variables: TRAF_CON_R1, TRAF_CON_R2

## Warning: model fit failed for Resample11: usekernel=FALSE, fL=0, adjust=1 Error in NaiveBayes.defaul
##   Zero variances for at least one class in variables: TRAF_CON_R1

## Warning: model fit failed for Resample12: usekernel=FALSE, fL=0, adjust=1 Error in NaiveBayes.defaul
##   Zero variances for at least one class in variables: TRAF_CON_R1, TRAF_CON_R2

## Warning: model fit failed for Resample13: usekernel=FALSE, fL=0, adjust=1 Error in NaiveBayes.defaul
##   Zero variances for at least one class in variables: TRAF_CON_R1, TRAF_CON_R2

## Warning: model fit failed for Resample14: usekernel=FALSE, fL=0, adjust=1 Error in NaiveBayes.defaul
##   Zero variances for at least one class in variables: TRAF_CON_R1

## Warning: model fit failed for Resample15: usekernel=FALSE, fL=0, adjust=1 Error in NaiveBayes.defaul
##   Zero variances for at least one class in variables: TRAF_CON_R1

## Warning: model fit failed for Resample16: usekernel=FALSE, fL=0, adjust=1 Error in NaiveBayes.defaul
##   Zero variances for at least one class in variables: TRAF_CON_R1, TRAF_CON_R2

## Warning: model fit failed for Resample17: usekernel=FALSE, fL=0, adjust=1 Error in NaiveBayes.defaul
##   Zero variances for at least one class in variables: TRAF_CON_R1

## Warning: model fit failed for Resample18: usekernel=FALSE, fL=0, adjust=1 Error in NaiveBayes.defaul
##   Zero variances for at least one class in variables: TRAF_CON_R1, TRAF_CON_R2

## Warning: model fit failed for Resample19: usekernel=FALSE, fL=0, adjust=1 Error in NaiveBayes.defaul
##   Zero variances for at least one class in variables: TRAF_CON_R1

## Warning: model fit failed for Resample20: usekernel=FALSE, fL=0, adjust=1 Error in NaiveBayes.defaul
##   Zero variances for at least one class in variables: TRAF_CON_R1, TRAF_CON_R2

## Warning: model fit failed for Resample21: usekernel=FALSE, fL=0, adjust=1 Error in NaiveBayes.defaul
##   Zero variances for at least one class in variables: TRAF_CON_R1, TRAF_CON_R2

## Warning: model fit failed for Resample22: usekernel=FALSE, fL=0, adjust=1 Error in NaiveBayes.defaul
##   Zero variances for at least one class in variables: TRAF_CON_R1

## Warning: model fit failed for Resample23: usekernel=FALSE, fL=0, adjust=1 Error in NaiveBayes.defaul
##   Zero variances for at least one class in variables: TRAF_CON_R1

## Warning: model fit failed for Resample24: usekernel=FALSE, fL=0, adjust=1 Error in NaiveBayes.defaul
##   Zero variances for at least one class in variables: TRAF_CON_R1, TRAF_CON_R2

## Warning: model fit failed for Resample25: usekernel=FALSE, fL=0, adjust=1 Error in NaiveBayes.defaul
##   Zero variances for at least one class in variables: TRAF_CON_R1, TRAF_CON_R2

## Warning in nominalTrainWorkflow(x = x, y = y, wts = weights, info = trainInfo,
## : There were missing values in resampled performance measures.

## Warning in train.default(x, y, weights = w, ...): missing values found in
## aggregated results
```

```r
predict(nb2, newdata = accidents24[,c("INJURY", "WEATHER_R", "TRAF_CON_R")])
```

```
##  [1] no no no no no no no no no no no no no no no no no no no no no no no no
## Levels: no yes
```

```r
predict(nb2, newdata = accidents24[,c("INJURY", "WEATHER_R", "TRAF_CON_R")],
                                    type = "raw")
```

```
##  [1] no no no no no no no no no no no no no no no no no no no no no no no no
## Levels: no yes
```

3.Let us now return to the entire dataset. Partition the data into training (60%) and validation (40%). Run a naive Bayes classifier on the complete training set with the relevant predictors (and INJURY as the response). Note that all predictors are categorical. Show the confusion matrix.

```r
set.seed(1)
train.index <- sample(c(1:dim(accidents)[1]), dim(accidents)[1]*0.6)
train.df <- accidents[train.index,]
valid.df <- accidents[-train.index,]
#defining a variable to be used here
vars <- c("INJURY", "HOUR_I_R",  "ALIGN_I" ,"WRK_ZONE",  "WKDY_I_R",
          "INT_HWY",  "LGTCON_I_R", "PROFIL_I_R", "SPD_LIM", "SUR_COND",
          "TRAF_CON_R",   "TRAF_WAY",   "WEATHER_R")

nbTotal <- naiveBayes(INJURY~.,data = train.df[,vars])
nbTotal
```

```
##
## Naive Bayes Classifier for Discrete Predictors
##
## Call:
## naiveBayes.default(x = X, y = Y, laplace = laplace)
##
## A-priori probabilities:
## Y
##        no       yes
## 0.4939745 0.5060255
##
## Conditional probabilities:
##      HOUR_I_R
## Y             0         1
##   no  0.5689490 0.4310510
##   yes 0.5703131 0.4296869
##
##      ALIGN_I
## Y             1         2
##   no  0.8712206 0.1287794
##   yes 0.8652300 0.1347700
##
##      WRK_ZONE
## Y              0          1
##   no  0.97664374 0.02335626
```

```
##   yes 0.97727805 0.02272195
##
##      WKDY_I_R
## Y            0         1
##   no  0.2194049 0.7805951
##   yes 0.2381510 0.7618490
##
##      INT_HWY
## Y               0            1            9
##   no  0.8513837786 0.1481362982 0.0004799232
##   yes 0.8593737800 0.1397673147 0.0008589053
##
##      LGTCON_I_R
## Y            1         2         3
##   no  0.6870101 0.1251000 0.1878899
##   yes 0.7014914 0.1096275 0.1888811
##
##      PROFIL_I_R
## Y            0         1
##   no  0.7531595 0.2468405
##   yes 0.7633326 0.2366674
##
##      SPD_LIM
## Y               5           10           15           20           25
##   no  0.0000799872 0.0004799232 0.0043992961 0.0085586306 0.1121420573
##   yes 0.0001561646 0.0003123292 0.0040602795 0.0039041149 0.0906535488
##      SPD_LIM
## Y              30           35           40           45           50
##   no  0.0860662294 0.1896496561 0.0962246041 0.1553351464 0.0407934730
##   yes 0.0860466932 0.2123057703 0.1068946670 0.1574139143 0.0394315609
##      SPD_LIM
## Y              55           60           65           70           75
##   no  0.1590145577 0.0355143177 0.0645496721 0.0409534474 0.0062390018
##   yes 0.1549152807 0.0430233466 0.0621535098 0.0311548372 0.0075739830
##
##      SUR_COND
## Y              1           2           3           4           9
##   no  0.774196129 0.176931691 0.016717325 0.028155495 0.003999360
##   yes 0.815725775 0.151245413 0.010697275 0.016709612 0.005621926
##
##      TRAF_CON_R
## Y            0         1         2
##   no  0.6566149 0.1902096 0.1531755
##   yes 0.6213009 0.2191770 0.1595221
##
##      TRAF_WAY
## Y             1          2          3
##   no  0.57998720 0.36690130 0.05311150
##   yes 0.56063090 0.39743890 0.04193019
##
##      WEATHER_R
## Y            1         2
##   no  0.8390657 0.1609343
##   yes 0.8744437 0.1255563
```

```r
#generating the confusion matrix using the train.df, the prediction and the classes
confusionMatrix(train.df$INJURY, predict(nbTotal, train.df[, vars]), positive = "yes")
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   no  yes
##        no  5097 7405
##        yes 4230 8577
##
##                Accuracy : 0.5403
##                  95% CI : (0.5341, 0.5464)
##     No Information Rate : 0.6315
##     P-Value [Acc > NIR] : 1
##
##                   Kappa : 0.0776
##
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 0.5367
##             Specificity : 0.5465
##          Pos Pred Value : 0.6697
##          Neg Pred Value : 0.4077
##              Prevalence : 0.6315
##          Detection Rate : 0.3389
##    Detection Prevalence : 0.5060
##       Balanced Accuracy : 0.5416
##
##        'Positive' Class : yes
##
```

What is the overall error of the validation set?

```r
confusion_matrix= confusionMatrix(valid.df$INJURY, predict(nbTotal, valid.df[, vars]), positive = "yes")
print(confusion_matrix)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   no  yes
##        no  3203 5016
##        yes 2862 5793
##
##                Accuracy : 0.5331
##                  95% CI : (0.5256, 0.5407)
##     No Information Rate : 0.6406
##     P-Value [Acc > NIR] : 1
##
##                   Kappa : 0.0594
##
##  Mcnemar's Test P-Value : <2e-16
##
```

```
##             Sensitivity : 0.5359
##             Specificity : 0.5281
##          Pos Pred Value : 0.6693
##          Neg Pred Value : 0.3897
##              Prevalence : 0.6406
##          Detection Rate : 0.3433
##    Detection Prevalence : 0.5129
##       Balanced Accuracy : 0.5320
##
##        'Positive' Class : yes
##
```

```r
#Calculated overall error
overall_error <- 1 - confusion_matrix$overall["Accuracy"]
cat("overall error of the validation set:", overall_error, "\n")
```

```
## overall error of the validation set: 0.4668721
```