FML_Assignment_3_NaiveBayes

Sri Naga Dattu Gummadi

2023-10-13

Summary

It is noticed that When an accident has just been reported and no additional information is given, it is assumed that injuries may have occurred (INJURY = Yes). This assumption is established in order to properly depict the accident's maximum amount of damage, MAX_SEV_IR. If MAX_SEV_IR is 1 or 2, then an injury has occurred, per the instructions (INJURY = Yes). On the other side, if MAX_SEV_IR equals 0, it indicates that there isn't any inferred damage (damage = No). As a result, until fresh information indicates otherwise, it is reasonable to assume that there was some level of injury caused by the accident when there is a lack of further information about it.

- There are "20721 NO and yes are 21462" in total. To create a new data frame with 24 records and only 3 variables (Injury, Weather, and Traffic), the following procedures were carried out:

With the factors of traffic, weather, and injuries, a pivot table was created. In this stage, the data had to be set up in a tabular format with these specific columns.

- Because it would not be used in the analysis that would come next, the variable Injury was removed from the data frame.

The likelihood of an injury occurring was calculated using Bayes probabilities for each of the first 24 elements in the data frame. Accidents that were categorized with a 0.5 cutoff. - Using the probabilities generated in Step 3, each accident was categorized as either likely or not likely to cause injuries based on a 0.5 cutoff criterion. WEATHER_R and TRAF_CON_R were each set to 1 to determine the naive bayes conditional probability of harm. The results are as follows.

-The likelihood of an injury is zero.

The chance is 1 if there is no damage.

The Bayes model was tested against the sample data (24 observations) using the Naive Bayes approach. Although yes/no is used as a classifier in both models, they don't match at the observation level. The situation for ordering was the same.

The assumption of conditional independent probability, which Naive Bayes uses, requires that each feature be considered separately, which Bayes does not do.

Using the same Naive Bayes model, which had a 60% training set, produced results with a 52.03% accuracy rate, a sensitivity (TPR) of 15.61%, and a specificity (TPR) of 87.27%. The remaining 40% went toward the validation data set, which produced an error rate of 47.43%.

Problem Statement

The file accidentsFull.csv contains information on 42,183 actual automobile accidents in 2001 in the United States that involved one of three levels of injury: NO INJURY, INJURY, or FATALITY. For each accident, additional information is recorded, such as day of week, weather conditions, and road type. A firm might be interested in developing a system for quickly classifying the severity of an accident based on initial reports and associated data in the system (some of which rely on GPS-assisted reporting).

Our goal here is to predict whether an accident just reported will involve an injury (MAX_SEV_IR = 1 or 2) or will not (MAX_SEV_IR = 0). For this purpose, create a dummy variable called INJURY that takes the value "yes" if MAX_SEV_IR = 1 or 2, and otherwise "no."

Q1. 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?

```
library(e1071)
library(ggplot2)
library(caret)
```

```
## Loading required package: lattice
```

```
accidents <- read.csv("/Users/srinagadattugummadi/Downloads/accidentsFull.csv")
accidents$INJURY = ifelse(accidents$MAX_SEV_IR>0,"yes","no")

# 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
```

```
table(accidents$INJURY)
```

```
##
##    no   yes
## 20721 21462
```

Given the frequency of Injury is yes are higher from the given dataset, if an accident is just reported there is a likely chance that the Injury is yes. (CHANGE THIS STATEMENT)

Q2. 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.

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

```
##   INJURY WEATHER_R TRAF_CON_R
```

```
## 1    yes         1          0
## 2     no         2          0
## 3     no         2          1
## 4     no         1          1
## 5     no         1          0
## 6    yes         2          0
```

#Creating 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.

```
pivot_table1 <- ftable(accidents24)
pivot_table2 <- ftable(accidents24[,-1])

pivot_table1
```

```
##                  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
```

```
pivot_table2
```

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

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 12 records. Use all three variables in the pivot table as rows/columns.

+ Compute the exact Bayes conditional probabilities of an injury (INJURY = Yes) given the six possible combinations of the predictors.

```
#Injury=yes
p1 = pivot_table1[3,1]/pivot_table2[1,1] #Injury, Weather = 1, Traf = 0
p2 = pivot_table1[4,1]/pivot_table2[2,1] #Injury, Weather = 2, Traf = 0
p3 = pivot_table1[3,2]/pivot_table2[1,2] #Injury, Weather = 1, Traf = 1
p4 = pivot_table1[4,2]/pivot_table2[2,2] #Injury, Weather = 2, Traf = 1
p5 = pivot_table1[3,3]/pivot_table2[1,3] #Injury, Weather = 1, Traf = 2
p6 = pivot_table1[4,3]/pivot_table2[2,3] #Injury, Weather = 2, Traf = 2
```

```
print(c(p1,p2,p3,p4,p5,p6))
```

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

2. Let us now compute

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

```
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$TRAF_CON_R[i]=="0"){
        prob.inj[i] = p2
      }
      else if (accidents24$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

prob.inj <- rep(0,24)

head(accidents24,n=24)
```

```
##    INJURY WEATHER_R TRAF_CON_R  prob.inj
## 1     yes         1          0 0.6666667
## 2      no         2          0 0.1818182
## 3      no         2          1 0.0000000
## 4      no         1          1 0.0000000
## 5      no         1          0 0.6666667
## 6     yes         2          0 0.1818182
## 7      no         2          0 0.1818182
## 8     yes         1          0 0.6666667
```

```
## 9      no         2          0 0.1818182
## 10     no         2          0 0.1818182
## 11     no         2          0 0.1818182
## 12     no         1          2 0.0000000
## 13    yes         1          0 0.6666667
## 14     no         1          0 0.6666667
## 15    yes         1          0 0.6666667
## 16    yes         1          0 0.6666667
## 17     no         2          0 0.1818182
## 18     no         2          0 0.1818182
## 19     no         2          0 0.1818182
## 20     no         2          0 0.1818182
## 21    yes         1          0 0.6666667
## 22     no         1          0 0.6666667
## 23    yes         2          2 1.0000000
## 24    yes         2          0 0.1818182
```

```
print(c(p1,p2,p3,p4,p5,p6))
```

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

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

```
##    INJURY WEATHER_R TRAF_CON_R  prob.inj pred.prob
## 1     yes         1          0 0.6666667       yes
## 2      no         2          0 0.1818182        no
## 3      no         2          1 0.0000000        no
## 4      no         1          1 0.0000000        no
## 5      no         1          0 0.6666667       yes
## 6     yes         2          0 0.1818182        no
## 7      no         2          0 0.1818182        no
## 8     yes         1          0 0.6666667       yes
## 9      no         2          0 0.1818182        no
## 10     no         2          0 0.1818182        no
## 11     no         2          0 0.1818182        no
## 12     no         1          2 0.0000000        no
## 13    yes         1          0 0.6666667       yes
## 14     no         1          0 0.6666667       yes
## 15    yes         1          0 0.6666667       yes
## 16    yes         1          0 0.6666667       yes
## 17     no         2          0 0.1818182        no
## 18     no         2          0 0.1818182        no
## 19     no         2          0 0.1818182        no
## 20     no         2          0 0.1818182        no
## 21    yes         1          0 0.6666667       yes
## 22     no         1          0 0.6666667       yes
## 23    yes         2          2 1.0000000       yes
## 24    yes         2          0 0.1818182        no
```

Q2.3 Compute manually the naive Bayes conditional probability of an injury given WEATHER_R = 1 and TRAF_CON_R = 1.

```
accidents24$INJURYnum = ifelse(accidents24$INJURY=="yes",1,0)

#Injury = Yes

# Calculating the probability of an injury
probability_Injury_Yes <- sum(accidents24$INJURYnum == 1) / nrow(accidents24)
probability_Injury_Yes
```

```
## [1] 0.375
```

```
# Calculating the probability of WEATHER_R = 1 given INJURY = 1
probability_Injury_Yes_WR <- sum(accidents24$WEATHER_R == 1 & accidents24$INJURYnum == 1) / sum(acciden
probability_Injury_Yes_WR
```

```
## [1] 0.6666667
```

```
# Calculating the probability of TRAF_CON_R = 1 given INJURY = 1
probability_InjuryYes_TR <- sum(accidents24$TRAF_CON_R == 1 & accidents24$INJURYnum == 1) / sum(acciden
probability_InjuryYes_TR
```

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

```
#Injury=No
# Calculating the probability of an injury
probability_Injury_No <- sum(accidents24$INJURYnum == 0) / nrow(accidents24)
probability_Injury_No
```

```
## [1] 0.625
```

```
# Calculating the probability of WEATHER_R = 1 given INJURY = 1
probability_InjuryNo_WR <- sum(accidents24$WEATHER_R == 1 & accidents24$INJURYnum == 0) / sum(accidents
probability_InjuryNo_WR
```

```
## [1] 0.3333333
```

```
# Calculating the probability of TRAF_CON_R = 1 given INJURY = 1
probability_InjuryNo_TR <- sum(accidents24$TRAF_CON_R == 1 & accidents24$INJURYnum == 0) / sum(accident
probability_InjuryNo_TR
```

```
## [1] 0.1333333
```

```
# Calculating the naive Bayes conditional probability of an injury given WEATHER_R = 1 and TRAF_CON_R =
probability_Injury_Yes <- probability_Injury_Yes * probability_Injury_Yes_WR * probability_InjuryYes_TR
probability_Injury_Yes
```

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

```
probability_Injury_No <- probability_Injury_No * probability_InjuryNo_WR * probability_InjuryNo_WR
probability_Injury_No
```

```
## [1] 0.06944444
```

```
Naive_Bayes <- (probability_Injury_Yes)/(probability_Injury_Yes+probability_Injury_No)
Naive_Bayes
```

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

Q2.4 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?

```
Naive_Bayes <- naiveBayes(INJURY ~ TRAF_CON_R + WEATHER_R, data = accidents24)
nbt <- predict(Naive_Bayes, newdata = accidents24, type = "raw")
nbt
```

```
##              no         yes
##  [1,] 0.4285714 0.571428571
##  [2,] 0.7500000 0.250000000
##  [3,] 0.9977551 0.002244949
##  [4,] 0.9910803 0.008919722
##  [5,] 0.4285714 0.571428571
##  [6,] 0.7500000 0.250000000
##  [7,] 0.7500000 0.250000000
##  [8,] 0.4285714 0.571428571
##  [9,] 0.7500000 0.250000000
## [10,] 0.7500000 0.250000000
## [11,] 0.7500000 0.250000000
## [12,] 0.3333333 0.666666667
## [13,] 0.4285714 0.571428571
## [14,] 0.4285714 0.571428571
## [15,] 0.4285714 0.571428571
## [16,] 0.4285714 0.571428571
## [17,] 0.7500000 0.250000000
## [18,] 0.7500000 0.250000000
## [19,] 0.7500000 0.250000000
## [20,] 0.7500000 0.250000000
## [21,] 0.4285714 0.571428571
## [22,] 0.4285714 0.571428571
## [23,] 0.6666667 0.333333333
## [24,] 0.7500000 0.250000000
```

```r
accidents24$nbpred.prob  <- nbt[,2]
accidents24$nb.preb.prob <- ifelse(accidents24$nbpred.prob>0.5,"yes","no")
accidents24
```

```
##    INJURY WEATHER_R TRAF_CON_R  prob.inj pred.prob INJURYnum nbpred.prob
## 1     yes         1          0 0.6666667       yes         1 0.571428571
## 2      no         2          0 0.1818182        no         0 0.250000000
## 3      no         2          1 0.0000000        no         0 0.002244949
## 4      no         1          1 0.0000000        no         0 0.008919722
## 5      no         1          0 0.6666667       yes         0 0.571428571
## 6     yes         2          0 0.1818182        no         1 0.250000000
## 7      no         2          0 0.1818182        no         0 0.250000000
## 8     yes         1          0 0.6666667       yes         1 0.571428571
## 9      no         2          0 0.1818182        no         0 0.250000000
## 10     no         2          0 0.1818182        no         0 0.250000000
## 11     no         2          0 0.1818182        no         0 0.250000000
## 12     no         1          2 0.0000000        no         0 0.666666667
## 13    yes         1          0 0.6666667       yes         1 0.571428571
## 14     no         1          0 0.6666667       yes         0 0.571428571
## 15    yes         1          0 0.6666667       yes         1 0.571428571
## 16    yes         1          0 0.6666667       yes         1 0.571428571
## 17     no         2          0 0.1818182        no         0 0.250000000
## 18     no         2          0 0.1818182        no         0 0.250000000
## 19     no         2          0 0.1818182        no         0 0.250000000
## 20     no         2          0 0.1818182        no         0 0.250000000
## 21    yes         1          0 0.6666667       yes         1 0.571428571
## 22     no         1          0 0.6666667       yes         0 0.571428571
## 23    yes         2          2 1.0000000       yes         1 0.333333333
## 24    yes         2          0 0.1818182        no         1 0.250000000
##    nb.preb.prob
## 1           yes
## 2            no
## 3            no
## 4            no
## 5           yes
## 6            no
## 7            no
## 8           yes
## 9            no
## 10           no
## 11           no
## 12          yes
## 13          yes
## 14          yes
## 15          yes
## 16          yes
## 17           no
## 18           no
## 19           no
## 20           no
## 21          yes
## 22          yes
## 23           no
```

```
## 24           no
```

```
#Classification results and ranking of the observations are not similar.
```

Question 3

#3. Let us now return to the entire dataset. Partition the data into training (60%) and validation (40%).

```
set.seed(100)
training_set<- sample(row.names(accidents), 0.6*dim(accidents)[1])
validation_set <- setdiff(row.names(accidents), training_set)
training.df <- accidents[training_set,]
validation.df <- accidents[validation_set,]

for (i in c(1:dim(training.df)[2])){
training.df[,i] <- as.factor(training.df[,i])
}

for (i in c(1:dim(validation.df)[2])){
validation.df[,i] <- as.factor(validation.df[,i])
}

accidents <- rbind(training.df,validation.df)
head(accidents)
```

```
##       HOUR_I_R ALCHL_I ALIGN_I STRATUM_R WRK_ZONE WKDY_I_R INT_HWY LGTCON_I_R
## 16887        0       2       1         1        0        0       0          1
## 3696         0       2       1         1        0        1       0          3
## 31705        0       2       1         1        0        1       1          1
## 24270        1       2       2         0        0        0       0          1
## 11159        0       1       1         1        0        1       1          2
## 26116        0       2       1         0        0        0       0          1
##       MANCOL_I_R PED_ACC_R RELJCT_I_R REL_RWY_R PROFIL_I_R SPD_LIM SUR_COND
## 16887          2         0          0         1          0      35        2
## 3696           0         0          0         0          1      65        1
## 31705          0         0          0         0          0      75        1
## 24270          2         0          1         1          0      45        1
## 11159          0         0          0         0          0      70        1
## 26116          2         0          1         1          0      45        1
##       TRAF_CON_R TRAF_WAY VEH_INVL WEATHER_R INJURY_CRASH NO_INJ_I
## 16887          0        1        2         1            0        0
## 3696           0        2        1         1            0        0
## 31705          0        2        1         1            0        0
## 24270          2        3        2         1            0        0
## 11159          0        2        2         1            0        0
## 26116          1        2        2         1            1        1
##       PRPTYDMG_CRASH FATALITIES MAX_SEV_IR INJURY
## 16887              1          0          0     no
## 3696               1          0          0     no
## 31705              1          0          0     no
```

```
## 24270              1          0          0     no
## 11159              1          0          0     no
## 26116              0          0          1    yes
```

3.1 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.

```
training_set<- naiveBayes(INJURY ~ TRAF_CON_R + WEATHER_R, data = training.df)
training_set
```

```
##
## Naive Bayes Classifier for Discrete Predictors
##
## Call:
## naiveBayes.default(x = X, y = Y, laplace = laplace)
##
## A-priori probabilities:
## Y
##        no       yes
## 0.4936189 0.5063811
##
## Conditional probabilities:
##      TRAF_CON_R
## Y             0         1         2
##   no  0.6599696 0.1881854 0.1518450
##   yes 0.6187578 0.2219881 0.1592541
##
##      WEATHER_R
## Y             1         2
##   no  0.8455935 0.1544065
##   yes 0.8714888 0.1285112
```

```
nbt.train.test <- predict(training_set, newdata = training.df,type = "raw")
head(nbt.train.test)
```

```
##             no       yes
## [1,] 0.5021974 0.4978026
## [2,] 0.5021974 0.4978026
## [3,] 0.5021974 0.4978026
## [4,] 0.4741904 0.5258096
## [5,] 0.5021974 0.4978026
## [6,] 0.4450018 0.5549982
```

```
training.df$nbpred.prob.full <- nbt.train.test[,2]
head(training.df)
```

```
##       HOUR_I_R ALCHL_I ALIGN_I STRATUM_R WRK_ZONE WKDY_I_R INT_HWY LGTCON_I_R
```

```
## 16887        0       2       1         1        0        0       0          1
## 3696         0       2       1         1        0        1       0          3
## 31705        0       2       1         1        0        1       1          1
## 24270        1       2       2         0        0        0       0          1
## 11159        0       1       1         1        0        1       1          2
## 26116        0       2       1         0        0        0       0          1
##       MANCOL_I_R PED_ACC_R RELJCT_I_R REL_RWY_R PROFIL_I_R SPD_LIM SUR_COND
## 16887          2         0          0         1          0      35        2
## 3696           0         0          0         0          1      65        1
## 31705          0         0          0         0          0      75        1
## 24270          2         0          1         1          0      45        1
## 11159          0         0          0         0          0      70        1
## 26116          2         0          1         1          0      45        1
##       TRAF_CON_R TRAF_WAY VEH_INVL WEATHER_R INJURY_CRASH NO_INJ_I
## 16887          0        1        2         1             0        0
## 3696           0        2        1         1             0        0
## 31705          0        2        1         1             0        0
## 24270          2        3        2         1             0        0
## 11159          0        2        2         1             0        0
## 26116          1        2        2         1             1        1
##       PRPTYDMG_CRASH FATALITIES MAX_SEV_IR INJURY nbpred.prob.full
## 16887              1          0          0     no        0.4978026
## 3696               1          0          0     no        0.4978026
## 31705              1          0          0     no        0.4978026
## 24270              1          0          0     no        0.5258096
## 11159              1          0          0     no        0.4978026
## 26116              0          0          1    yes        0.5549982
```

```
training.df$nbpred.prob.full.c <- ifelse(training.df$nbpred.prob.full>0.5, "yes","no")
training.df$nbpred.prob.full.c <- factor(training.df$nbpred.prob.full.c)
head(training.df)
```

```
##       HOUR_I_R ALCHL_I ALIGN_I STRATUM_R WRK_ZONE WKDY_I_R INT_HWY LGTCON_I_R
## 16887        0       2       1         1        0        0       0          1
## 3696         0       2       1         1        0        1       0          3
## 31705        0       2       1         1        0        1       1          1
## 24270        1       2       2         0        0        0       0          1
## 11159        0       1       1         1        0        1       1          2
## 26116        0       2       1         0        0        0       0          1
##       MANCOL_I_R PED_ACC_R RELJCT_I_R REL_RWY_R PROFIL_I_R SPD_LIM SUR_COND
## 16887          2         0          0         1          0      35        2
## 3696           0         0          0         0          1      65        1
## 31705          0         0          0         0          0      75        1
## 24270          2         0          1         1          0      45        1
## 11159          0         0          0         0          0      70        1
## 26116          2         0          1         1          0      45        1
##       TRAF_CON_R TRAF_WAY VEH_INVL WEATHER_R INJURY_CRASH NO_INJ_I
## 16887          0        1        2         1             0        0
## 3696           0        2        1         1             0        0
## 31705          0        2        1         1             0        0
## 24270          2        3        2         1             0        0
## 11159          0        2        2         1             0        0
## 26116          1        2        2         1             1        1
##       PRPTYDMG_CRASH FATALITIES MAX_SEV_IR INJURY nbpred.prob.full
```

```
## 16887              1          0          0     no        0.4978026
## 3696               1          0          0     no        0.4978026
## 31705              1          0          0     no        0.4978026
## 24270              1          0          0     no        0.5258096
## 11159              1          0          0     no        0.4978026
## 26116              0          0          1    yes        0.5549982
##       nbpred.prob.full.c
## 16887                 no
## 3696                  no
## 31705                 no
## 24270                yes
## 11159                 no
## 26116                yes
```

```
ConfusionMatrix <- confusionMatrix(training.df$nbpred.prob.full.c, training.df$INJURY)
ConfusionMatrix
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   no  yes
##        no  8474 8130
##        yes 4019 4686
##
##                Accuracy : 0.52
##                  95% CI : (0.5138, 0.5261)
##     No Information Rate : 0.5064
##     P-Value [Acc > NIR] : 7.821e-06
##
##                   Kappa : 0.0438
##
##  Mcnemar's Test P-Value : < 2.2e-16
##
##             Sensitivity : 0.6783
##             Specificity : 0.3656
##          Pos Pred Value : 0.5104
##          Neg Pred Value : 0.5383
##              Prevalence : 0.4936
##          Detection Rate : 0.3348
##    Detection Prevalence : 0.6561
##       Balanced Accuracy : 0.5220
##
##        'Positive' Class : no
##
```

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

```
validation <- naiveBayes(INJURY ~ TRAF_CON_R + WEATHER_R, data = validation.df)
head(validation)
```

```
## $apriori
## Y
##   no  yes
```

```
## 8228 8646
##
## $tables
## $tables$TRAF_CON_R
##      TRAF_CON_R
## Y             0         1         2
##   no  0.6573894 0.1894750 0.1531356
##   yes 0.6242193 0.2161693 0.1596114
##
## $tables$WEATHER_R
##      WEATHER_R
## Y             1         2
##   no  0.8337385 0.1662615
##   yes 0.8752024 0.1247976
##
##
## $levels
## [1] "no"  "yes"
##
## $isnumeric
## TRAF_CON_R  WEATHER_R
##      FALSE      FALSE
##
## $call
## naiveBayes.default(x = X, y = Y, laplace = laplace)
```

```
validation <- predict(validation, newdata = validation.df,type = "raw")
head(validation)
```

```
##             no       yes
## [1,] 0.5263531 0.4736469
## [2,] 0.4884235 0.5115765
## [3,] 0.5717733 0.4282267
## [4,] 0.4884235 0.5115765
## [5,] 0.5717733 0.4282267
## [6,] 0.5717733 0.4282267
```

```
validation.df$nbpred.prob.full <- validation[,2]
head(validation.df)
```

```
##    HOUR_I_R ALCHL_I ALIGN_I STRATUM_R WRK_ZONE WKDY_I_R INT_HWY LGTCON_I_R
## 3         1       2       1         0        0        1       0          3
## 5         1       1       1         0        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
## 11        1       2       1         0        0        1       0          3
##    MANCOL_I_R PED_ACC_R RELJCT_I_R REL_RWY_R PROFIL_I_R SPD_LIM SUR_COND
## 3           2         0          1         1          1      35        4
## 5           2         0          0         1          1      25        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
```

```
## 11          0         0          0         0          1      55        4
##    TRAF_CON_R TRAF_WAY VEH_INVL WEATHER_R INJURY_CRASH NO_INJ_I PRPTYDMG_CRASH
## 3           1        2        2         2            0        0              1
## 5           0        2        3         1            0        0              1
## 7           0        2        1         2            0        0              1
## 8           0        1        1         1            1        1              0
## 9           0        1        1         2            0        0              1
## 11          0        1        1         2            0        0              1
##    FATALITIES MAX_SEV_IR INJURY nbpred.prob.full
## 3           0          0     no        0.4736469
## 5           0          0     no        0.5115765
## 7           0          0     no        0.4282267
## 8           0          1    yes        0.5115765
## 9           0          0     no        0.4282267
## 11          0          0     no        0.4282267
```

```
validation.df$nbpred.prob.full.c <- ifelse(validation.df$nbpred.prob.full>0.5, "yes","no")
validation.df$nbpred.prob.full.c <- factor(validation.df$nbpred.prob.full.c)
head(validation.df)
```

```
##    HOUR_I_R ALCHL_I ALIGN_I STRATUM_R WRK_ZONE WKDY_I_R INT_HWY LGTCON_I_R
## 3         1       2       1         0        0        1       0          3
## 5         1       1       1         0        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
## 11        1       2       1         0        0        1       0          3
##    MANCOL_I_R PED_ACC_R RELJCT_I_R REL_RWY_R PROFIL_I_R SPD_LIM SUR_COND
## 3           2         0          1         1          1      35        4
## 5           2         0          0         1          1      25        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
## 11          0         0          0         0          1      55        4
##    TRAF_CON_R TRAF_WAY VEH_INVL WEATHER_R INJURY_CRASH NO_INJ_I PRPTYDMG_CRASH
## 3           1        2        2         2            0        0              1
## 5           0        2        3         1            0        0              1
## 7           0        2        1         2            0        0              1
## 8           0        1        1         1            1        1              0
## 9           0        1        1         2            0        0              1
## 11          0        1        1         2            0        0              1
##    FATALITIES MAX_SEV_IR INJURY nbpred.prob.full nbpred.prob.full.c
## 3           0          0     no        0.4736469                 no
## 5           0          0     no        0.5115765                yes
## 7           0          0     no        0.4282267                 no
## 8           0          1    yes        0.5115765                yes
## 9           0          0     no        0.4282267                 no
## 11          0          0     no        0.4282267                 no
```

```
# showing confusion matrix
ConfusionMatrix <- confusionMatrix(validation.df$nbpred.prob.full.c, validation.df$INJURY)$overall[1]
ConfusionMatrix
```

```
##  Accuracy
```

```
## 0.5295129
```

```
# calculating the overall error of validation set
cal_Validation_Error <- (1-ConfusionMatrix)
cal_Validation_Error
```

```
##  Accuracy
## 0.4704871
```