FML Assignment\_3

Sri Hari Vadhry

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? A. There is 50.88% chance that an injury has been occurred. This is because of the reported cases as 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. a.Compute the exact Bayes conditional probabilities of an injury (INJURY = Yes) given the six possible combinations of the predictors. A. Following are the Bayes probability of INJURY = YES

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

1. Classify the 24 accidents using these probabilities and a cutoff of 0.5. A. Using prob and cutoff of 0.5 the below are the predictions of 24 accidents Qualitative predictions- 0.6666667 0.1818182 0.0000000 0.0000000 0.6666667 0.1818182 0.1818182 0.6666667 0.1818182 0.1818182 0.1818182 0.0000000 0.6666667 0.6666667 0.6666667 0.6666667 0.1818182 0.1818182 0.1818182 0.1818182 0.6666667 0.6666667 1.0000000 0.1818182 Quantitative 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. Compute manually the naive Bayes conditional probability of an injury given WEATHER\_R = 1 and TRAF\_CON\_R = 1 A. The result of Naive Bayes when Weather =1 and Traf\_Con=1 is 0

d.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?

A. As we have applied Naive Bayes Classifier to the 24 records and 2 predictors along with the check of models output it has been understood that resultant classifications and rank are distant.

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. A.  
Reference Prediction no yes no 3203 5016 yes 2862 5793, Accuracy =0.5331

1. What is the overall error of the validation set? A. Overall validation Error is 0.4668

# 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.”

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?
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.
* Classify the 24 accidents using these probabilities and a cutoff of 0.5.
* Compute manually the naive Bayes conditional probability of an injury given WEATHER\_R = 1 and TRAF\_CON\_R = 1.
* 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?

1. 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.
* What is the overall error of the validation set?

## Data Input and Cleaning

# Load the required libraries

library(e1071)  
library(caret)

## Loading required package: ggplot2

## Loading required package: lattice

# Importing the data

accidents <- read.csv("C:/Users/Hari/Downloads/accidentsFull.csv")  
  
dim(accidents)

## [1] 42183 24

accidents$INJURY = ifelse(accidents$MAX\_SEV\_IR>0,"yes","no")

# Coverting variables into 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

## Questions

1. 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.
* Classify the 24 accidents using these probabilities and a cutoff of 0.5.
* Compute manually the naive Bayes conditional probability of an injury given WEATHER\_R = 1 and TRAF\_CON\_R = 1.
* 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?

accidents24 <- accidents[1:24,c("INJURY","WEATHER\_R","TRAF\_CON\_R")]  
#head(accidents24)

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

dt2

## TRAF\_CON\_R 0 1 2  
## WEATHER\_R   
## 1 9 1 1  
## 2 11 1 1

1. 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 = 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

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

## [1] 0.3333333 0.8181818 1.0000000 1.0000000 1.0000000 0.0000000

1. 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$prob.inj <- prob.inj  
accidents24$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.

Prob\_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])  
Prob\_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])  
ProbIY = (dt1[3,1]+dt1[3,2]+dt1[3,3]+dt1[4,1]+dt1[4,2]+dt1[4,3])/24  
Prob\_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])  
Prob\_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])  
ProbIN = (dt1[1,1]+dt1[1,2]+dt1[1,3]+dt1[2,1]+dt1[2,2]+dt1[2,3])/24  
  
Prob\_IY\_W1.T1= (Prob\_W1\_IY\*Prob\_T1\_IY\*ProbIY)/((Prob\_W1\_IY\*Prob\_T1\_IY\*ProbIY)+(Prob\_W1\_IN\*Prob\_T1\_IN\*ProbIN))  
Prob\_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?

library(klaR)

## Loading required package: MASS

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

library(klaR)  
  
formula <- INJURY ~ TRAF\_CON\_R + WEATHER\_R  
  
accidents24$INJURY <- as.factor(accidents24$INJURY)  
  
nb2 <- NaiveBayes(formula,data = accidents24)  
  
predict(nb2, newdata = accidents24[,c("INJURY", "WEATHER\_R", "TRAF\_CON\_R")])

## $class  
## 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20   
## yes no no no yes no no yes no no no yes yes yes yes yes no no no no   
## 21 22 23 24   
## yes yes no no   
## Levels: no yes  
##   
## $posterior  
## 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

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

a.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.

set.seed(1)  
  
train\_df <- sample(row.names(accidents),0.6\*dim(accidents)[1])  
valid\_df <- setdiff(row.names(accidents),train\_df)  
  
  
  
train.df <- accidents[train\_df,]  
valid.df <- accidents[valid\_df,]  
  
# Defining a variable to be used  
  
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")  
  
naivepred <- naiveBayes(INJURY~.,data = train.df[,vars])  
naivepred

##   
## 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

1. What is the overall error of validation set?

confusionMatrix = confusionMatrix(valid.df$INJURY, predict(naivepred, valid.df[, vars]), positive = "yes")  
  
print(confusionMatrix)

## 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   
##

# Calculated Overall Error

overall\_error\_rate = 1 - confusionMatrix$overall["Accuracy"]  
  
cat("Overall Error", overall\_error\_rate)

## Overall Error 0.4668721