FML ASSIGNMENT 3

JYOTHIRMAI MOPARTHI

2023-10-16

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?

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

Summary

I've kept the data in a data frame called Accident_data, so we can access it whenever we want. I then made a dummy variable called Injury to represent the severity of the injury. If there is more than one or two, we assume there is some form of injury. If it is less than zero, we consider there to be no injury.

1. Injury is one of the types of variables we have, and it has classifiers like yes or no. Since we only know that an accident was reported, we would expect it to result in injuries. This is due to the fact that more records have the annotation “Injury=yes” than “No,” indicating a higher probability of an accident.

2: Using the two predictive parameters WEATHER_R and TRAF_CON_R, we will select the top 24 entries from the dataset. The dataset is contained in the “Sub_accident_data” field. In order to better understand the data, we might put the information into a pivot table and arrange it according to the amount of traffic and the weather. The pivot table looks like this: 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
```

#Bayes Theorem : P(A/B) = (P(B/A)P(A))/P(B) where P(A),P(B) are events and P(B) not equal to 0.

We could determine the probability that one of the six injury predictors would be positive. The following values are what we obtained for various combinations.

P(INJURY = Yes | WEATHER_R = 1 and TRAF_CON_R = 0): 0.6666667

P(INJURY = Yes | WEATHER_R = 2 and TRAF_CON_R = 0): 0.1818182

P(INJURY = Yes | WEATHER_R = 2 and TRAF_CON_R = 2): 1

The other 3 combinations pf probability of injury=yes is 0.

P(INJURY = Yes | WEATHER_R = 1 and TRAF_CON_R = 1): 0

P(INJURY = Yes | WEATHER_R = 1 and TRAF_CON_R = 2): 0

P(INJURY = Yes | WEATHER_R = 2 and TRAF_CON_R = 1): 0

The first and most important thing to accomplish is to take note of the fact that both of these classes show “yes” at the same indices. This shows that the Ranking (= Ordering) of the observations is reliable.

-If the rank is same, both categories understand the data similarly and give each factor the same amount of weight. In this case, the models consistently make decisions on the importance of the data pieces.

In conclusion, this evaluation was based on a subgroup with only three features. To assess the model’s overall performance and equivalence, it would typically be evaluated on a dataset as a whole. The common evaluation measures, such as recall, accuracy, and precision, as well as the F1-score, which provides a more thorough insight of models

-We now divide all of our data into two sets: a training set (60%) and a validation set (40%). Following the analysis of the sets, we use the training data to train the model in order to use the information to identify future crashes (new or unseen data).

-Validation Set: This set is used to validate the data it includes, using a reference as the training set, so that we may know how effectively our model is trained when they get unknown data (new data). Given the training set, it categorizes the validation set.

-We normalize the data to put all of the data on the same line after partitioning the data frame. We operate on this normalized data to provide precise numbers that we utilize in our decision-making.

-It is crucial that the characteristics being compared be numbers or integers and have the same levels to prevent errors.

Data Input and Cleaning

Load the required libraries and read the input file

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

```
## Loading required package: ggplot2
```

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

```r
AccidentsFull <- read.csv("AccidentsFull.csv")
#Exploring the data given in the data-set file by using some predefined operations in R
head(AccidentsFull, 10)
```

```
##    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
##    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
##    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
##    FATALITIES MAX_SEV_IR
## 1           0          1
## 2           0          0
## 3           0          0
## 4           0          0
## 5           0          0
## 6           0          1
## 7           0          0
## 8           0          1
## 9           0          0
## 10          0          0
```

#Creating a new variable i.e„ “INJURY” based on the values in MAX_SEV_IR

```
AccidentsFull$INJURY = ifelse(AccidentsFull$MAX_SEV_IR>0,"yes","no")
yes_no_counts <- table(AccidentsFull$INJURY)
yes_no_counts
```

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

#Convert variables to factor

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

Predict based on the majority class

```
yes_count <- yes_no_counts["yes"]
no_count <- yes_no_counts["no"]
prediction <- ifelse((yes_count > no_count), "Yes", "No")
print(paste("Prediction of the new accident: INJURY =", prediction))
```

```
## [1] "Prediction of the new accident: INJURY = Yes"
```

```
Yes_percent <- (yes_count/(yes_count+no_count))*100
print(paste("The percentage of Accident being INJURY is:", round(Yes_percent,2),"%"))
```

```
## [1] "The percentage of Accident being INJURY is: 50.88 %"
```

```
No_percent <- (no_count/(yes_count+no_count))*100
print(paste("The percentage of Accident being NO INJURY is:", round(No_percent,2), "%"))
```

```
## [1] "The percentage of Accident being NO INJURY is: 49.12 %"
```

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.

```
Accidents24 <- AccidentsFull[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
```

```
dt1 <- ftable(Accidents24)
dt2 <- ftable(Accidents24[,-1]) # print table only for conditions
print("Table with all three variables:")
```

```
## [1] "Table with all three variables:"
```

```
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
```

```
print("Table without the first variable (INJURY):")
```

```
## [1] "Table without the first variable (INJURY):"
```

```
dt2
```

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

#2(1). 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 the conditional probabilities
print("Conditional Probabilities given Injury = Yes:")
```

```
## [1] "Conditional Probabilities given Injury = Yes:"
```

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

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

```
print("Conditional Probabilities given Injury = No:")
```

```
## [1] "Conditional Probabilities given Injury = No:"
```

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

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

#2(2). 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
```

```r
#Adding a new column with the probability
Accidents24$prob.inj <- prob.inj
#Classify using the threshold of 0.5.
Accidents24$pred.prob <- ifelse(Accidents24$prob.inj>0.5, "yes", "no")
#Print the resulting dataframe
head(Accidents24, 10)
```

```
##    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
```

#2(3). Compute manually the naive Bayes conditional probability of an injury given WEATHER_R = 1 and TRAF_CON_R = 1.

```
#loading the library
library(e1071)

#ceating a naive bayes model
Naive_Bayes_Model <- naiveBayes(INJURY ~ WEATHER_R + TRAF_CON_R, data = Accidents24)

#Identify the data that we wish to use to calcul
Data <- data.frame(WEATHER_R = "1", TRAF_CON_R = "1")

# Predict the probability of "Yes" class
Prob_Naive_Bayes <- predict(Naive_Bayes_Model, newdata = Data, type = "raw")
injury_prob_naive_bayes <- Prob_Naive_Bayes[1, "yes"]

# Print the probability
cat("Naive Bayes Conditional Probability for WEATHER_R = 1 and TRAF_CON_R = 1:\n")
```

```
## Naive Bayes Conditional Probability for WEATHER_R = 1 and TRAF_CON_R = 1:
```

```
cat(injury_prob_naive_bayes, "\n")
```

```
## 0.008919722
```

#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?

```
# Load the e1071 library for naiveBayes
library(e1071)

# Create a naive Bayes model for the 24 records and two predictors
nb_model_24 <- naiveBayes(INJURY ~ WEATHER_R + TRAF_CON_R, data = Accidents24)

# Predict using the naive Bayes model with the same data
Naive_Bayes_Predictions_24 <- predict(nb_model_24, Accidents24)

# Extract the probability of "Yes" class for each record
injury_prob_naive_bayes_24 <- attr(Naive_Bayes_Predictions_24, "probabilities")[, "yes"]

# Create a vector of classifications based on a cutoff of 0.5
classification_results_naive_bayes_24 <- ifelse(injury_prob_naive_bayes_24 > 0.5, "yes", "no")

# Print the classification results
cat("Classification Results based on Naive Bayes for 24 records:\n")
```

```
## Classification Results based on Naive Bayes for 24 records:
```

```
cat(classification_results_naive_bayes_24, sep = " ")

# Check if the resulting classifications are equivalent to the exact Bayes classification
equivalent_classifications <- classification_results_naive_bayes_24 == Accidents24$pred.prob

# Check if the ranking (= ordering) of observations is equivalent
equivalent_ranking <- all.equal(injury_prob_naive_bayes_24, as.numeric(Accidents24["yes", , ]))
cat("Are the classification results are equivalent?", "\n")
```

```
## Are the classification results are equivalent?
```

```
print(all(equivalent_classifications))
```

```
## [1] TRUE
```

```
cat("are the ranking of observations are equivalent?", "\n")
```

```
## are the ranking of observations are equivalent?
```

```
print(equivalent_ranking)
```

```
## [1] "target is NULL, current is numeric"
```

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

#i. 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(123)

# splitting the data
training_indices <- createDataPartition(AccidentsFull$INJURY, p = 0.6, list = FALSE)
training_data <- AccidentsFull[training_indices, ]
valid_data <- AccidentsFull[-training_indices, ]

# training the naive bayes
naive_bayes_model <- naiveBayes(INJURY ~ WEATHER_R + TRAF_CON_R, data = training_data)

# generating predicitions on validation data
predictions_valid <- predict(naive_bayes_model, newdata = valid_data)

# creating a confusion matrix
confusion_matrix <- table(predictions_valid, valid_data$INJURY)

# Print the confusion matrix
print("The confusion matrix is:")
```

```
## [1] "The confusion matrix is:"
```

```
print(confusion_matrix)
```

```
##
## predictions_valid   no  yes
##               no  1294 1064
##               yes 6994 7520
```

#ii. What is the overall error of the validation set?

```
#Calculating the overall error rate
overall_error_rate <- 1 - sum(diag(confusion_matrix)) / sum(confusion_matrix)
cat("The overall error rate is:", overall_error_rate)
```

```
## The overall error rate is: 0.477596
```