FML_Assignment_3

Ganesh

2023-10-15

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).

TOur 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?

Data Input and Cleaning

Load the required libraries and read the input file

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

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

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

```
accidents <- read.csv("accidentsFull.csv") #loding the data to R from the saved folder
head(accidents, 15)
```

```
##    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
##    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
##    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
##    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
## 11          0          0
## 12          0          0
## 13          0          1
## 14          0          0
## 15          0          1
```

Create a pivot table that examines INJURY as a function of the two predictors for these 24 records.

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

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

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
```

Predict based on the majority class

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

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

```r
Yes_percentage<- (count_yes/(count_yes+count_no))*100
print(paste("The percentage of Accident being INJURY is:", round(Yes_percentage,2),"%"))
```

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

```r
No_percentage <- (count_no/(count_yes+count_no))*100
print(paste("The percentage of Accident being NO INJURY is:", round(No_percentage,2), "%"))
```

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

#Explanation for prediction of the new accident : Injury = Yes

#The forcast should be INJURY = Yes if an accident has just been reported and since no fresh information is available. This is because50.88 of accidents in the sample had injuries as a result. Consequently, there's an inadequate information in favour of injuries being in an accident as opposed to not. This is only a vaticination, after all, and there's no assurance that anyone will be hurt in the collision. Making a more precise protuberance would bear further details, similar as the extent of the vehicles' damage and the number of injured persons.

When an accident has just been reported, and there is no additional information available, it is difficult to make a precise prediction regarding the outcome of the accident. However, you can make a general and conservative assumption based on historical data and common knowledge.

In this scenario, a conservative prediction would be to assume that there is a possibility of injury (“INJURY” = “Yes”). This assumption is based on the fact that many automobile accidents do result in injuries, even if they are minor. It acknowledges the potential risk of injury without making any specific claims about the severity.

Again, it’s essential to emphasize that this is a very general and cautious assumption, and actual outcomes can vary widely based on numerous factors, including the specific circumstances of the accident. To make more accurate predictions, additional data and, possibly, advanced predictive modeling techniques would be necessary.

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.***

```
accidents_24 <- accidents[1:24,c("INJURY","WEATHER_R","TRAF_CON_R")]
head(accidents_24)
```

```
##   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
```

```
GK1 <- ftable(accidents_24)
GK2 <- ftable(accidents_24[,-1]) # print table only for conditions

GK1
```

```
##                   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
```

```
GK2
```

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

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

```
# Injury = yes
r1 = GK1[3,1] / GK2[1,1] # Injury, Weather=1 and Traf=0
r2 = GK1[4,1] / GK2[2,1] # Injury, Weather=2, Traf=0
r3 = GK1[3,2] / GK2[1,2] # Injury, W=1, T=1
r4 = GK1[4,2] / GK2[2,2] # I, W=2,T=1
r5 = GK1[3,3] / GK2[1,3] # I, W=1,T=2
r6 = GK1[4,3]/ GK2[2,3] #I,W=2,T=2

# Injury = no
m1 = GK1[1,1] / GK2[1,1] # Weather=1 and Traf=0
m2 = GK1[2,1] / GK2[2,1] # Weather=2, Traf=0
m3 = GK1[1,2] / GK2[1,2] # W=1, T=1
m4 = GK1[2,2] / GK2[2,2] # W=2,T=1
m5 = GK1[1,3] / GK2[1,3] # W=1,T=2
m6 = GK1[2,3] / GK2[2,3] # W=2,T=2
# Print the conditional probabilities
print("Conditional Probabilities given Injury = Yes:")
```

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

```
print(c(r1,r2,r3,r4,r5,r6))
```

```
## [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(m1,m2,m3,m4,m5,m6))
```

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

b. 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(accidents_24$WEATHER_R[i],accidents_24$TRAF_CON_R[i]))
    if (accidents_24$WEATHER_R[i] == "1") {
      if (accidents_24$TRAF_CON_R[i]=="0"){
        prob.inj[i] = r1
      }
      else if (accidents_24$TRAF_CON_R[i]=="1") {
        prob.inj[i] = r3
      }
      else if (accidents_24$TRAF_CON_R[i]=="2") {
        prob.inj[i] = r5
      }
```

```r
    }
    else {
      if (accidents_24$TRAF_CON_R[i]=="0"){
        prob.inj[i] = r2
      }
      else if (accidents_24$TRAF_CON_R[i]=="1") {
        prob.inj[i] = r4
      }
      else if (accidents_24$TRAF_CON_R[i]=="2") {
        prob.inj[i] = r6
      }
    }
  }
```

```
## [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
```

```
#Adding a new column with the probability
accidents_24$prob.inj <- prob.inj
#Classify using the threshold of 0.5.
accidents_24$pred.prob <- ifelse(accidents_24$prob.inj>0.5, "yes", "no")
#Print the resulting dataframe
head(accidents_24, 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
```

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

```
library(e1071) #loading the library

#ceating a naive bayes model
naive_bayes <- naiveBayes(INJURY ~ WEATHER_R + TRAF_CON_R, data = accidents_24)

#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
naive_bayes_prob <- predict(naive_bayes, newdata = Data, type = "raw")
injury_prob_naive_bayes <- naive_bayes_prob[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:
```

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

```
## 0.008919722
```

iV. 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?

```r
# Create a naive Bayes model for the 24 records and two predictors
nb_model24 <- naiveBayes(INJURY ~ WEATHER_R + TRAF_CON_R, data = accidents_24)

# Predict using the naive Bayes model with the same data
naive_bayes_predictions24 <- predict(nb_model24, accidents_24)

# Extract the probability of "Yes" class for each record
injury_prob_naive_bayes24 <- attr(naive_bayes_predictions24, "probabilities")[, "yes"]

# Create a vector of classifications based on a cutoff of 0.5
classification_results_naive_bayes24 <- ifelse(injury_prob_naive_bayes24 > 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:
```

```r
cat(classification_results_naive_bayes24, sep = " ")

# Check if the resulting classifications are equivalent to the exact Bayes classification
equivalent_classifications <- classification_results_naive_bayes24 == accidents_24$pred.prob

# Check if the ranking (= ordering) of observations is equivalent
equal_ranking <- all.equal(injury_prob_naive_bayes24, as.numeric(accidents_24["yes", , ]))
cat("Are the classification results are equivalent?", "\n")
```

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

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

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

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

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

```r
print(equal_ranking)
```

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

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.

```r
set.seed(123)

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

#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:"
```

```r
print(confusion_matrix)
```

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

What is the overall error of the validation set?

```r
#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
```

Analysis Summary:

1. Exact Bayes vs. Naive Bayes:

a.Both precise Bayes calculations and the use of a naïve Bayes classifier on a subset of the data were provided in the code.

b. A naive Bayes classifier was trained on a subset of 24 data, and the exact Bayes probability calculations were done manually.

2. **Comparison of Classifications:**

   a. The classifications obtained using the exact Bayes computations and the naïve Bayes model were contrasted.

   b. The code examined if the generated categories and the ranking (ordered) of the observations were equal.

3. **Naive Bayes on Entire Dataset:**

   a. The code divided the dataset into training (60%) and validation (40%) sets in order to expand the analysis to the complete collection of data.

   b. The entire training set, including the predictors WEATHER_R and TRAF_CON_R, was used to train a naïve Bayes classifier with INJURY as the response variable. In order to assess how well the model performed on the validation set, the confusion matrix was created.

4. **Overall Error Rate:**

   a. The proportion of misclassified cases was used to calculate the total error rate of the naive Bayes classifier on the validation set.

Conclusions :

1. **Comparing Exact Bayes and Naive Bayes:**

   a. To evaluate the effectiveness of the naive Bayes model, it is helpful to compare the precise Bayes and naive Bayes classifications. The naive Bayes assumptions may not be seriously broken if the classifications are equal.

2. **Naive Bayes on the Entire Dataset:**

   a. A naive Bayes classifier's performance can be better understood by training it on the complete dataset. As a result, the model can gain knowledge from a bigger sample of data.

3. **Model Evaluation with Confusion Matrix:**

   a. The confusion matrix is a useful tool for assessing how well the model is working. Regarding true positives, true negatives, false positives, and false negatives, it offers insights.

4. **Overall Error Rate:**

   a. The validation set's overall error rate quantifies the model's precision. Understanding how well the model generalises to fresh, untested data is crucial.

5. Considerations for Improvement:

a. Tuning hyperparameters, looking into more predictors, and figuring out how data preprocessing processes affect model performance are all possible areas for further investigation.

In conclusion, the research offers a strong framework for analysing the naive Bayes classifier's performance on the provided dataset. A thorough review is made possible by the comparison with exact Bayes, evaluation metrics, and suggestions for improvement.