FML_Assignment_3

Sushma Palancha

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).

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?

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
```

```
Accidentsfull <- read.csv("~/Documents/FML/FML ASSIGNMENT 3/accidentsFull (1).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

```r
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

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

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

```
No_percentage <- (no_count/(yes_count+no_count))*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 %"
```

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
```

```r
dt1 <- ftable(Accidents24)
dt2 <- ftable(Accidents24[,-1]) # print table only for conditions
dt1
```

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

```r
dt2
```

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

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

```r
# Injury = yes
pivot1 = dt1[3,1] / dt2[1,1] # Injury, Weather=1 and Traf=0
pivot2 = dt1[4,1] / dt2[2,1] # Injury, Weather=2, Traf=0
pivot3 = dt1[3,2] / dt2[1,2] # Injury, W=1, T=1
pivot4 = dt1[4,2] / dt2[2,2] # I, W=2,T=1
pivot5 = dt1[3,3] / dt2[1,3] # I, W=1,T=2
pivot6 = dt1[4,3]/ dt2[2,3] #I,W=2,T=2

# Injury = no
no1 = dt1[1,1] / dt2[1,1] # Weather=1 and Traf=0
no2 = dt1[2,1] / dt2[2,1] # Weather=2, Traf=0
no3 = dt1[1,2] / dt2[1,2] # W=1, T=1
no4 = dt1[2,2] / dt2[2,2] # W=2,T=1
no5 = dt1[1,3] / dt2[1,3] # W=1,T=2
no6 = 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:"
```

```r
print(c(pivot1,pivot2,pivot3,pivot4,pivot5,pivot6))
```

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

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

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

```
print(c(no1,no2,no3,no4,no5,no6))
```

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

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

```
probability.injury <- 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"){
        probability.injury[i] = no1
      }
      else if (Accidents24$TRAF_CON_R[i]=="1") {
        probability.injury[i] = no3
      }
      else if (Accidents24$TRAF_CON_R[i]=="2") {
        probability.injury[i] = no5
      }
    }
    else {
      if (Accidents24$TRAF_CON_R[i]=="0"){
        probability.injury[i] = no2
      }
      else if (Accidents24$TRAF_CON_R[i]=="1") {
        probability.injury[i] = no4
      }
      else if (Accidents24$TRAF_CON_R[i]=="2") {
        probability.injury[i] = no6
      }
    }
  }
```

```
## [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$probability.injury <- probability.injury
#Classify using the threshold of 0.5.
Accidents24$pred.probability <- ifelse(Accidents24$probability.injury>0.5, "yes", "no")
#Print the resulting dataframe
head(Accidents24, 10)
```

```
##    INJURY WEATHER_R TRAF_CON_R probability.injury pred.probability
## 1     yes         1          0          0.3333333               no
## 2      no         2          0          0.8181818              yes
## 3      no         2          1          1.0000000              yes
## 4      no         1          1          1.0000000              yes
## 5      no         1          0          0.3333333               no
## 6     yes         2          0          0.8181818              yes
## 7      no         2          0          0.8181818              yes
## 8     yes         1          0          0.3333333               no
## 9      no         2          0          0.8181818              yes
## 10     no         2          0          0.8181818              yes
```

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

```r
#creating a naive bayes model
naivebayes_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_naivebayes <- predict(naivebayes_model, newdata = Data, type = "raw")
injury_probability_naivebayes <- prob_naivebayes[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_probability_naivebayes, "\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
naivebayes_model_24 <- naiveBayes(INJURY ~ WEATHER_R + TRAF_CON_R, data = Accidents24)
# Predict using the naive Bayes model with the same data
naivebayes_predictions_24 <- predict(naivebayes_model_24, Accidents24)
# Extract the probability of "Yes" class for each record
injury_prob_naivebayes_24 <- attr(naivebayes_predictions_24, "probabilities")[, "yes"]
# Create a vector of classifications based on a cutoff of 0.5
classification_results_naivebayes_24 <- ifelse(injury_prob_naivebayes_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:
```

```r
cat(classification_results_naivebayes_24, sep = " ")
# Check if the resulting classifications are equivalent to the exact Bayes classification
equivalent_classifications <- classification_results_naivebayes_24 == Accidents24$pred.prob
# Check if the ranking (= ordering) of observations is equivalent
equivalent_ranking <- all.equal(injury_prob_naivebayes_24, as.numeric(Accidents24["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(equivalent_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(1)

# splitting the data
trainingindicate <- createDataPartition(Accidents24$INJURY, p = 0.6, list = FALSE)
trainingdata <- Accidents24[trainingindicate, ]
validdata <- Accidents24[-trainingindicate, ]

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

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

validdata$INJURY <- factor(validdata$INJURY,levels = levels(naivebayes_predictions_24))

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

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

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

```r
print(confusion_matrix)
```

```
##
## predictions_valid no yes
##               no   4   1
##               yes  2   2
```

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

```r
#Calculating the overall error rate
overall_errorrate <- 1 - sum(diag(confusion_matrix)) / sum(confusion_matrix)

cat("The overall error rate is:", overall_errorrate)
```

```
## The overall error rate is: 0.3333333
```

```
overr <- 1 - sum(predictions_valid == validdata$INJURY) / nrow(validdata)
overr
```

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

#Summary If an accident is recently recorded and no further information is given, it is assumed that there may be injuries (INJURY = Yes). This assumption is designed to accurately depict the maximum damage caused by the accident, MAX_SEV_IR. INJURY = Yes indicates that there has been an injury if MAX_SEV_IR is 1 or 2, per the guidelines. If, however, MAX_SEV_IR = 0, then INJURY = No, indicating that no inferred injury exists. Therefore, until fresh information proves otherwise, it is reasonable to assume that there was some injury caused when there is a lack of additional information on the accident. • "20721 NO and yes are 21462" is the total. To create a new data frame with 24 records and just 3 variables (traffic, weather, and injury), the following three steps were taken: a pivot table was set up with the variables traffic, weather, and injury. During this phase, the data had to be organised in a tabular style with these specific columns. • The variable Injury wouldn't be used in the ensuing analysis, it was removed from the data frame.

Bayes probabilities were computed to ascertain the probability of an injury occurring for each of the first 24 elements in the data frame. Incidents that were categorised using a 0.5 threshold. . Using the probabilities obtained in Step 3, each accident was categorised as either probable or unlikely to result in injuries based on a 0.5 cutoff criterion. The naive bayes conditional probability of harm was computed with WEATHER_R and TRAF_CON_R set to 1, respectively. The results are as follows. The chance of a harm occurring is zero. In the event of no harm, the probability is 1. The following outcomes arise from the exact Bayes classification and the predictions of the Naive Bayes model: [1]yes no no yes yes no no yes no no no yes yes yes yes yes no no no no [21]yes yes no no Levels: no yes [1]yes no no yes yes no no yes no no no yes yes yes yes yes no no no no [21]yes yes no no Levels: no yes The records in this case are categorised as "yes" or "no." Most importantly, we see that there are points at which the values in both categories are the same, indicating that there is agreement between the two classes regarding the order or ranking of the observations. This suggests that both classes understand the data and evaluate the components in a comparable way. In the next stage, the entire dataset is divided into two sets: a training set (which will include 60% of the data) and a validation set (40% of the data). This will be done by splitting the dataset and then using the training data to train the model. Metrics including accuracy, precision, recall, and F1-score will be used for a comprehensive analysis in order to assess the model's performance and ability to predict future accidents. The entire dataset will be used for this purpose. After the data frame is segmented, the following step is to normalise the data. This normalisation process allows for more accurate decision-making by ensuring that every segment is represented as a single row. For comparisons to be valid, the traits under study must have stable levels and be either integer or numeric values. It also ensures that operations on the data yield meaningful and reliable results for use in decision-making. This consistency in attribute levels and data types helps to prevent analytical errors. Furthermore, you stated that the validation set's overall error rate, reported in decimals, is roughly 0.47. This implies that the Naive Bayes classifier uses this dataset with a fair degree of accuracy and performance. The statistics and confusion matrix for your classification model are as follows: *Accuracy: 50% of the predictions made by your model are accurate, with an accuracy of 0.5.* Sensitivity: 0.15635 is the sensitivity, commonly referred to as the true positive rate or recall. This indicates that 15.635% of the time, your model properly detects positive cases (such as injuries). *Specificity: Specificity is 0.8708, meaning that 87.08% of the time your model successfully detects negative situations (i.e., no injuries). Overall, the model appears to be working well, but it may not be very excellent at correctly predicting injuries, particularly when the injuries are positive. Although the Naive Bayes method performs well, it oversimplifies the assumption of variable independence, which is not always true. Think about the specific findings and their implications in relation to your objectives and dataset.