FML_Assignment3

Krishna Krupa Singamshetty

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."

###Loading the required libraries

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

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

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

```
library(klaR)
```

```
## Loading required package: MASS
```

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following object is masked from 'package:MASS':
##
##     select
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

###Importing and reading the data

```
accidents <- read.csv("C:/Users/Krupa shetty/Downloads/accidentsFull.csv")

###Creating a new coulmn INJURY to classify MAX_SEV_IR, if MAX_SEV_IR is greater than 0, then injury is
accidents$INJURY = ifelse(accidents$MAX_SEV_IR>0,"yes","no")
head(accidents)
```

```
##   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
##   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
##   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
##   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
```

```
# 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
```

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?

It is not possible to predict if there would be any injury or not when an accident is reported with no sufficient information, but by using dataset that is historic data, we can say that if the probability of getting injured is greater than probability of not getting injured, then we can say that there are more chances to get injured whereas if probability of not getting injured is greater than probability of getting injured, then we can say that there is less likely chances that accident has a injury.

```
###probability of injury is yes
yes <- accidents%>% filter(accidents$INJURY=="yes")%>%summarise(count= n())
yes
```

```
##   count
## 1 21462
```

```
is_yes<-  yes/nrow(accidents)
is_yes
```

```
##       count
## 1 0.5087832
```

```
No <- accidents%>% filter(accidents$INJURY=="no")%>%summarise(count= n())
No
```

```
##   count
## 1 20721
```

```
is_no<-  yes/nrow(accidents)
is_no
```

```
##       count
## 1 0.5087832
```

Therefore, the probability of getting injury is 0.5087832 which is greater than probability og not getting injured 0.4912168, so we can say that if any accident is reported we can assume that there is an injury.

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.

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

```
dt1 = ftable(accidents24)
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 = ftable(accidents24[,-1]) # print table only for conditions
dt2
```

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

```r
# 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
```

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

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

#therefore, the conditional probabilities for 6 predictors are 1.Probability of Injury= yes given Weather = 1 and Traffic = 0 is 0.6666667 2.Probability of Injury= yes given Weather = 1 and Traffic = 1 is 0.0000000 3.Probability of Injury= yes given Weather = 1 and Traffic = 2 is 0.0000000 4.Probability of Injury= yes given Weather = 2 and Traffic = 0 is 0.1818182 5.Probability of Injury= yes given Weather = 2 and Traffic = 1 is 0.0000000 6.Probability of Injury= yes given Weather = 2 and Traffic = 2 is 1.0000000 7.Probability of Injury= no given Weather = 1 and Traffic = 0 is 0.3333333 8.Probability of Injury= no given Weather = 1 and Traffic = 1 is 1.0000000 9.Probability of Injury= no given Weather = 1 and Traffic = 2 is 1.0000000 10.Probability of Injury= no given Weather = 2 and Traffic = 0 is 0.8181818 11.Probability of Injury= no given Weather = 2 and Traffic = 1 is 1.0000000 12.Probability of Injury= no given Weather = 2 and Traffic = 2 is 0.0000000

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

```r
# Define a vector to store the classification results
classification_results <- character(24)

# Assign classifications based on the probabilities and a cutoff of 0.5
for (i in 1:24) {
    if (accidents24$WEATHER_R[i] == "1") {
        if (accidents24$TRAF_CON_R[i] == "0") {
            classification_results[i] = ifelse(p1 > 0.5, "Yes", "No")
        } else if (accidents24$TRAF_CON_R[i] == "1") {
            classification_results[i] = ifelse(p3 > 0.5, "Yes", "No")
        } else {
            classification_results[i] = ifelse(p5 > 0.5, "Yes", "No")
        }
    } else {
        if (accidents24$TRAF_CON_R[i] == "0") {
            classification_results[i] = ifelse(p2 > 0.5, "Yes", "No")
```

```
        } else if (accidents24$TRAF_CON_R[i] == "1") {
            classification_results[i] = ifelse(p4 > 0.5, "Yes", "No")
        } else {
            classification_results[i] = ifelse(p6 > 0.5, "Yes", "No")
        }
    }
}

# Print the classification results
cat("Classification Results based on Exact Bayes:\n")
```

```
## Classification Results based on Exact Bayes:
```

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

```
## Yes No No No Yes No No Yes No No No No 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.

```
data_x = accidents24[accidents24$WEATHER_R == "1" & accidents24$TRAF_CON_R == "1", ]

pro_yes = sum(data_x$INJURY == "yes") / nrow(data_x)

pro_yes
```

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

```
#The probability of injury =yes and WEATHER_R = 1 and TRAF_CON_R = 1 is 0. This means the model is esim
```

```
data_1 = data.frame(WEATHER_R = "1", TRAF_CON_R = "1")

naive_bayes_pred = naiveBayes(INJURY ~ WEATHER_R + TRAF_CON_R, data = accidents24)

prediction = predict(naive_bayes_pred, newdata =data_1, type = "raw")

prediction
```

```
##             no         yes
## [1,] 0.9910803 0.008919722
```

```
#Therefore, the probability of injury is yes is 0.008919722 and for injury is no is 0.9910803 for the g
```

-> 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?

```
Tnaive_bayes= naiveBayes(INJURY ~ TRAF_CON_R + WEATHER_R,
                 data = accidents24)

pred_nb = predict(Tnaive_bayes, newdata = accidents24)
# accidents24$nbpred.prob = nbt[,2] # Transfer the "Yes" nb prediction
pred_nb
```

```
##  [1] yes no  no  no  yes no  no  yes no  no  no  yes yes yes yes yes no  no  no
## [20] no  yes yes no  no
## Levels: no yes
```

```
cutoff = 0.5

exact_bayes = ifelse(c(p1, p2, p3, p4, p5, p6) > cutoff, "yes", "no")

result = data.frame(
  "Exact_Bayes" = exact_bayes,
  "Naive_Bayes_Probability" = pred_nb

)

classifications = exact_bayes == pred_nb

ranking = order(-as.numeric(c(p1, p2, p3, p4, p5, p6))) == order(-as.numeric(pred_nb))

cat("Are the resulting classifications equivalent? ", all(classifications), "\n")
```

```
## Are the resulting classifications equivalent?  FALSE
```

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

```
## Is the ranking of observations equivalent?  FALSE
```

The result of exact bayes classification and naive bayes doesnt match for all the data and the order that is ranking is also not equivalent for exact bayes and naive bayes. This means that even if both methods are similar, there is difference in their result. This might be because of the dataset and their variables.

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?

```
#Setting the seed as to get the same result everytime
set.seed(123)

#read the data
accidents_new = read.csv("C:/Users/Krupa shetty/Downloads/accidentsFull.csv")

accidents_new$INJURY = ifelse(accidents_new$MAX_SEV_IR>0,1,0)
```

```r
for (i in c(1:dim(accidents_new)[2])){
  accidents[,i] = as.factor(accidents_new[,i])
}


# index_split = createDataPartition(accidents_new$INJURY, p = 0.6, list = FALSE)
#
# train_data = accidents_new[index_split, ]
#
# validation_data = accidents_new[-index_split, ]

#splitting the data
t_split = sample(row.names(accidents_new), 0.6*dim(accidents_new)[1])

v_split = setdiff(row.names(accidents_new), t_split)

T_data = accidents_new[t_split,]

V_data = accidents_new[v_split,]

#naive bayes
naive_model = naiveBayes(INJURY ~ ., data = T_data)

naive_predictions = predict(naive_model, V_data)


##confusion matrix
confusion_matrix_nb = confusionMatrix(naive_predictions, as.factor(V_data$INJURY))

print(confusion_matrix_nb)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    0    1
##          0 8306  142
##          1    4 8422
##
##                Accuracy : 0.9913
##                  95% CI : (0.9898, 0.9927)
##     No Information Rate : 0.5075
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.9827
##
##  Mcnemar's Test P-Value : < 2.2e-16
##
##             Sensitivity : 0.9995
##             Specificity : 0.9834
##          Pos Pred Value : 0.9832
##          Neg Pred Value : 0.9995
##              Prevalence : 0.4925
##          Detection Rate : 0.4922
```

```
##    Detection Prevalence : 0.5007
##       Balanced Accuracy : 0.9915
##
##        'Positive' Class : 0
##
```

```
## Total error that is overall error
Total_error = 1 - confusion_matrix_nb$overall["Accuracy"]

Total_error
```

```
##    Accuracy
## 0.008652365
```

```
#The total error is 0.008652365 which ia almost equal to zero , the model performs good in case of pred
```