Lab 12: Naïve Bayes Classifier

Problem statement:

Task1: Use iris dataset which is available with R by default data (iris) and perform following operations:

- Develop Naïve Bayes classifier (Dependent variable (Species/Class) and rest of all are independent features)
- Observe the priori probabilities of all available classes.
- Observe the Conditional probabilities of all the classes against each and every independent features.
- Measured and display the confusion matrix
- Calculate the accuracy of model

Source Code:

```
#Author: Ashish Upadhyay
#Branch: Computer Science and Engineering
#Semester: 6th
#Dr. SP Mukherjee International Institute of Information Technology, Naya Raipur
#Subject: Machine Learning Lab 12
#Task: Naive Bayes Implementation

setwd("C:/Users/Ashish Upadhyay/Documents/Semester6/MachineLearning/Lab Programs")
getwd()
d <- read.csv("iris.csv")
head(d)
nrow(d)
summary(d)

#converting as a factor to class
d$class=factor(d$class)
#Finding structure of iris data
str(d)

# Creating table for class variable
table(d$class)

sample_iris=sample(150,110,replace = FALSE)

#creating training and test dataset
iris_training=d[sample_iris,]
iris_test=d[-sample_iris,]
#creating levels
iris_training_labels=d[sample_iris,]$class
iris_test_labels=d[-sample_iris,]$class

table(iris_training$class)
table(iris_test$class)

library(e1071)
iris_classifier=naiveBayes(class ~ ., data = iris_training)
class(iris_classifier)
print(iris_classifier)
```

```
summary(iris_classifier)
#Evaluvating model performance
iris_test_pred=predict(iris_classifier,iris_test)
iris_test_pred

#install.packages("gmodels")
#library(gmodels)
conf_matrix <- table(iris_test_pred, iris_test$class)
conf_matrix
accuracy <- sum(diag(conf_matrix)) / sum(conf_matrix)
accuracy
```

Output:

```
> #Author: Ashish Upadhyay
> #Branch: Computer Science and Engineering
> #Semester: 6th
> #Dr. SP Mukherjee International Institute of Information Technology, Naya Raipur
> #Subject: Machine Learning Lab 12
> #Task: Naive Bayes Implementation
>
> setwd("C:/Users/Ashish Upadhyay/Documents/Semester6/MachineLearning/Lab Programs")
> getwd()
[1] "C:/Users/Ashish Upadhyay/Documents/Semester6/MachineLearning/Lab Programs"
> d <- read.csv("iris.csv")
> head(d)
  length_sepal width_sepal length_petal width_petal      class
1          5.1         3.5          1.4         0.2 Iris-setosa
2          4.9         3.0          1.4         0.2 Iris-setosa
3          4.7         3.2          1.3         0.2 Iris-setosa
4          4.6         3.1          1.5         0.2 Iris-setosa
5          5.0         3.6          1.4         0.2 Iris-setosa
6          5.4         3.9          1.7         0.4 Iris-setosa
> nrow(d)
[1] 150
> summary(d)
  length_sepal    width_sepal     length_petal    width_petal
 Min.   :4.300   Min.   :2.000   Min.   :1.000   Min.   :0.100
 1st Qu.:5.100   1st Qu.:2.800   1st Qu.:1.600   1st Qu.:0.300
 Median :5.800   Median :3.000   Median :4.350   Median :1.300
 Mean   :5.843   Mean   :3.054   Mean   :3.759   Mean   :1.199
 3rd Qu.:6.400   3rd Qu.:3.300   3rd Qu.:5.100   3rd Qu.:1.800
 Max.   :7.900   Max.   :4.400   Max.   :6.900   Max.   :2.500
         class
 Iris-setosa    :50
 Iris-versicolor:50
 Iris-virginica :50



>
> #converting as a factor to class
> d$class=factor(d$class)
> #Finding structure of iris data
> str(d)
```

```
'data.frame':          150 obs. of  5 variables:
 $ length_sepal: num  5.1 4.9 4.7 4.6 5 5.4 4.6 5 4.4 4.9 ...
 $ width_sepal : num  3.5 3 3.2 3.1 3.6 3.9 3.4 3.4 2.9 3.1 ...
 $ length_petal: num  1.4 1.4 1.3 1.5 1.4 1.7 1.4 1.5 1.4 1.5 ...
 $ width_petal : num  0.2 0.2 0.2 0.2 0.2 0.4 0.3 0.2 0.2 0.1 ...
 $ class       : Factor w/ 3 levels "Iris-setosa",..: 1 1 1 1 1 1 1 1 1 1 ...
>
> # Creating table for class variable
> table(d$class)

   Iris-setosa Iris-versicolor  Iris-virginica
            50             50             50
>
> sample_iris=sample(150,110,replace = FALSE)
>
> #creating training and test dataset
> iris_training=d[sample_iris,]
> iris_test=d[-sample_iris,]
> #creating levels
> iris_training_labels=d[sample_iris,]$class
> iris_test_labels=d[-sample_iris,]$class
>
> table(iris_training$class)

   Iris-setosa Iris-versicolor  Iris-virginica
            35             36             39
> table(iris_test$class)

   Iris-setosa Iris-versicolor  Iris-virginica
            15             14             11
>
> library(e1071)
> iris_classifier=naiveBayes(class ~ ., data = iris_training)
> class(iris_classifier)
[1] "naiveBayes"
> print(iris_classifier)

Naive Bayes Classifier for Discrete Predictors

Call:
naiveBayes.default(x = X, y = Y, laplace = laplace)

A-priori probabilities:
Y
   Iris-setosa Iris-versicolor  Iris-virginica
     0.3181818      0.3272727      0.3545455

Conditional probabilities:
                 length_sepal
Y                     [,1]     [,2]
  Iris-setosa     5.014286 0.3614862
  Iris-versicolor 5.913889 0.5259836
  Iris-virginica  6.684615 0.5828869

                 width_sepal
```

```
Y                 [,1]     [,2]
 Iris-setosa     3.431429 0.3771187
 Iris-versicolor 2.769444 0.3087481
 Iris-virginica  3.007692 0.3351234

             length_petal
Y                 [,1]     [,2]
 Iris-setosa     1.465714 0.1781322
 Iris-versicolor 4.255556 0.4494088
 Iris-virginica  5.617949 0.5305765

             width_petal
Y                 [,1]     [,2]
 Iris-setosa     0.2285714 0.1045197
 Iris-versicolor 1.3277778 0.1830084
 Iris-virginica  2.0461538 0.2798930

> summary(iris_classifier)
      Length Class  Mode
apriori 3     table  numeric
tables  4     -none- list
levels  3     -none- character
call    4     -none- call
> #Evaluvating model performance
> iris_test_pred=predict(iris_classifier,iris_test)
> iris_test_pred
 [1] Iris-setosa     Iris-setosa     Iris-setosa     Iris-setosa     Iris-setosa
 [6] Iris-setosa     Iris-setosa     Iris-setosa     Iris-setosa     Iris-setosa
[11] Iris-setosa     Iris-setosa     Iris-setosa     Iris-setosa     Iris-setosa
[16] Iris-versicolor Iris-versicolor Iris-versicolor Iris-versicolor Iris-versicolor
[21] Iris-versicolor Iris-virginica  Iris-versicolor Iris-versicolor Iris-versicolor
[26] Iris-versicolor Iris-versicolor Iris-versicolor Iris-versicolor Iris-virginica
[31] Iris-versicolor Iris-virginica  Iris-virginica  Iris-virginica  Iris-virginica
[36] Iris-virginica  Iris-versicolor Iris-virginica  Iris-virginica  Iris-virginica
Levels: Iris-setosa Iris-versicolor Iris-virginica
>
> #install.packages("gmodels")
> #library(gmodels)
> conf_matrix <- table(iris_test_pred, iris_test$class)
> conf_matrix

iris_test_pred    Iris-setosa Iris-versicolor Iris-virginica
 Iris-setosa           15            0            0
 Iris-versicolor        0           13            2
 Iris-virginica         0            1            9
> accuracy <- sum(diag(conf_matrix)) / sum(conf_matrix)
> accuracy
[1] 0.925
```