



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

Experiment2.2

Student Name: Kanishk Soni
Branch: BE-CSE
Semester: 6th
Subject Code: 20CSP-376

UID: 20BCS9398
Section/Group: DM-708/B
Subject Name: Data Mining Lab

1. Aim:

To perform classification using Bayesian classification algorithm using R

Objective:

- The objective here is to determine the likelihood of an event A happening given B happens.
- To evaluate the accuracy and performance of the naïve bayes algorithm.
- Bayes theorem gives the conditional probability of an event A given another event B has occurred.

2. Code and Output:

- **PROGRAM**

```
# Naive Bayes

# Importing the dataset
dataset = read.csv('Social_Network_Ads.csv')
dataset = dataset[3:5]

# Encoding the target feature as factor
dataset$Purchased = factor(dataset$Purchased, levels = c(0, 1))

# Splitting the dataset into the Training set and Test set
library(caTools)
split = sample.split(dataset$Purchased, SplitRatio = 0.75)
training_set = subset(dataset, split == TRUE)
test_set = subset(dataset, split == FALSE)

# Feature Scaling
training_set[-3] = scale(training_set[-3])
test_set[-3] = scale(test_set[-3])
```



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

```
# Fitting Naive Baiyes Classifier to the Training set
```

```
library(e1071)
```

```
classifier = naiveBayes(x = training_set[-3],  
                        y = training_set$Purchased)
```

```
print(classifier)
```

```
# Predicting train set results
```

```
y_pred_train = predict(classifier, newdata = training_set[-3])
```

```
# Making the Confusion Matrix for training set
```

```
cm_train = table(training_set[, 3], y_pred_train)
```

```
print(cm_train)
```

```
#Accuracy on training data
```

```
accuracy_train <- sum(diag(cm_train))/sum(cm_train)
```

```
cat("\nAccuracy on training set: ", accuracy_train)
```

```
# Predicting the Test set results
```

```
y_pred_test = predict(classifier, newdata = test_set[-3])
```

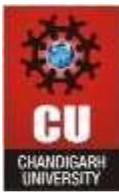
```
# Making the Confusion Matrix for testing set
```

```
cm_test = table(test_set[, 3], y_pred_test)
```

```
# Accuracy on test data
```

```
accuracy_test <- sum(diag(cm_test))/sum(cm_test)
```

```
cat("\nAccuracy on test set: ", accuracy_test)
```



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

- OUTPUT

```
Console Terminal Background Jobs
R 4.2.2 · C:/Users/hp/Downloads/
1 0.4416656 1.2549938

>
> # Predicting train set results
> y_pred_train = predict(Classifier, newdata = training_set[-3])
>
> # Making the Confusion Matrix for training set
> cm_train = table(training_set[, 3], y_pred_train)
>
> print(cm_train)
  y_pred_train
    0      1
0 181    12
1   17    90
>
> #Accuracy on training data
> accuracy_train <- sum(diag(cm_train))/sum(cm_train)
>
> cat("\nAccuracy on training set: ", accuracy_train)

Accuracy on training set: 0.9033333>
> # Predicting the Test set results
> y_pred_test = predict(Classifier, newdata = test_set[-3])
>
> # Making the Confusion Matrix for testing set
> cm_test = table(test_set[, 3], y_pred_test)
>
> # Accuracy on test data
> accuracy_test <- sum(diag(cm_test))/sum(cm_test)
> cat("\nAccuracy on test set: ", accuracy_test)

Accuracy on test set: 0.87
> |
```



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

Environment	History	Connections	Tutorial
Import Dataset 171 MiB			
R Global Environment			
Data			
classifier	List of 5		
dataset	400 obs. of 3 variables		
Social_Network_Ads	400 obs. of 5 variables		
test_set	100 obs. of 3 variables		
training_set	300 obs. of 3 variables		
Values			
accuracy_test	0.87		
accuracy_train	0.9033333333333333		
cm_test	'table' int [1:2, 1:2] 59 8 5 28		
cm_train	'table' int [1:2, 1:2] 181 17 12 90		
split	logi [1:400] FALSE TRUE TRUE TRUE TRUE FALSE ...		
y_pred_test	Factor w/ 2 levels "0","1": 1 1 1 1 1 1 2 1 1 ...		
y_pred_train	Factor w/ 2 levels "0","1": 1 1 1 1 2 1 1 1 1 2 ...		

```
1 # Naive Bayes
2
3 # Importing the dataset
4 dataset = read.csv('Social_Network_Ads.csv')
5 dataset = dataset[3:5]
6
7 # Encoding the target feature as factor
8 dataset$Purchased = factor(dataset$Purchased, levels = c(0, 1))
9
10 # Splitting the dataset into the Training set and Test set
11 library(caTools)
12 split = sample.split(dataset$Purchased, SplitRatio = 0.75)
13 training_set = subset(dataset, split == TRUE)
14 test_set = subset(dataset, split == FALSE)
15
16 # Feature Scaling
17 training_set[-3] = scale(training_set[-3])
18 test_set[-3] = scale(test_set[-3])
19
20 # Fitting Naive Bayes Classifier to the Training set
21 fitNaiveBayes = naiveBayes(training_set[, 1:2], training_set[, 3])
22
23 # Predicting the Test set results
24 y_pred = predict(fitNaiveBayes, test_set[, 1:2])
25
26 # Confusion Matrix
27 cm = table(y_pred, test_set[, 3])
28
29 # Accuracy
30 accuracy = sum(diag(cm)) / sum(cm)
31
32 # Print the confusion matrix and accuracy
33 print(cm)
34 print(accuracy)
```

test_set	100 obs. of 3 variables
training_set	300 obs. of 3 variables

accuracy_test	0.87
accuracy_train	0.9033333333333333
cm_test	'table' int [1:2, 1:2] 59 8 5 28
cm_train	'table' int [1:2, 1:2] 181 17 12 90
split	logi [1:400] FALSE TRUE TRUE TRUE TRUE FALSE ...
y_pred_test	Factor w/ 2 levels "0","1": 1 1 1 1 1 1 2 1 1 ...
y_pred_train	Factor w/ 2 levels "0","1": 1 1 1 1 2 1 1 1 1 2 ...

```
R 4.2.2 - C:\Users\hp\Downloads\
trying URL 'https://cran.rstudio.com/bin/windows/contrib/4.2/caTools_1.18.2.zip'
Content type 'application/zip' length 246175 bytes (240 KB)
downloaded 240 KB

package 'caTools' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
C:\Users\hp\AppData\Local\Temp\Rtmpm4w6v\downloaded_packages
> library(caTools)
Warning message:
package 'caTools' was built under R version 4.2.3
> split = sample.split(dataset$Purchased, SplitRatio = 0.75)
> training_set = subset(dataset, split == TRUE)
> test_set = subset(dataset, split == FALSE)
> training_set[-3] = scale(training_set[-3])
> test_set[-3] = scale(test_set[-3])
>
> # Fitting Naive Bayes Classifier to the Training set
> fitNaiveBayes = naiveBayes(training_set[, 1:2], training_set[, 3])
> library(a1071)
```