DATE:

EXPERIMENT-7

Aim:

Write a program to implement the Naive Bayesian classifier for the given dataset and compute the accuracy of the classifier.

Requirements:

Algorithm:

1. Naive Bayes algorithm is a supervised learning algorithm, which is based on Bayes theorem and used for solving classification problems.
2. It is a probabilistic classifier, which means it predicts on the basis of the probability of an object.

Dataset:

1. 'user_data' contains the information of various users obtained from the social networking sites. There is a car making company that has recently launched a new car. So the company wanted to check whether the user can purchase a car or not.
2. For this problem, we will build a Machine Learning model using the Naive Bayes algorithm.

Importing the libraries :

1. pandas is imported to read the csv file.
2. From sklearn.naive_bayes we should import GaussianNB ,it is used to fit it to the training dataset.
3. from sklearn.metrics we should import confusion_matrix ,it is used to test the accuracy of the result.

Procedure:

1. Data Pre-processing step: In this step, we will pre-process/prepare the data so that we can use it efficiently in our code.
2. Fitting Naive Bayes to the Training set: After the pre-processing step, now we will fit the Naive Bayes model to the Training set.

3. Predicting the test result: Now we will predict the test set result. For this, we will create a new predictor variable y_red, and will use the predict function to make the predictions.
4. Test accuracy of the result (Creation of Confusion matrix): Now we will check the accuracy of the Naive Bayes classifier using the Confusion matrix.

Code:

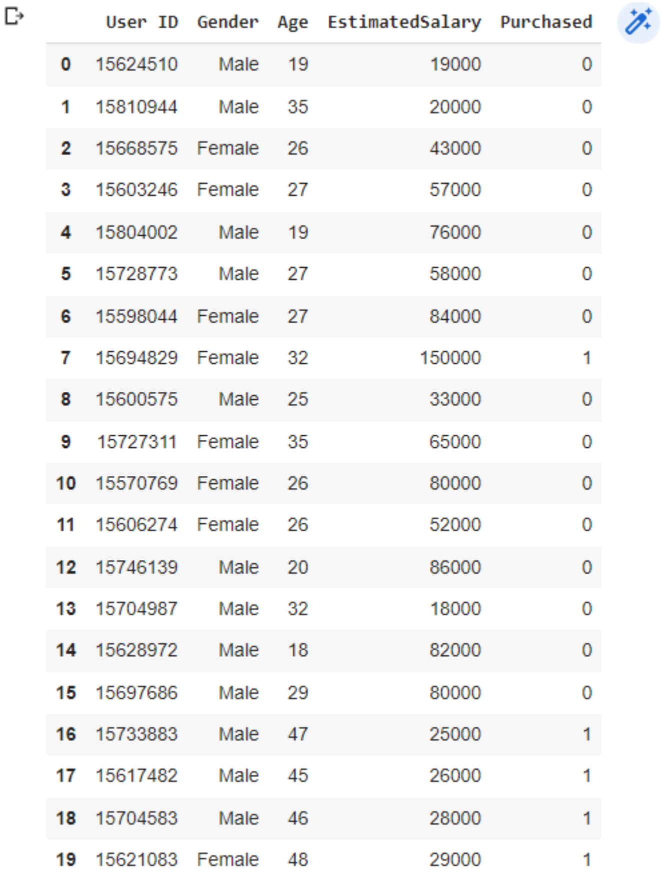
```
import pandas as pd
import numpy as nm

# Importing the dataset
dataset = pd.read_csv('user_data.csv')
dataset
x = dataset.iloc[:, [2, 3]].values
y = dataset.iloc[:, 4].values

# Splitting the dataset into the Training set and Test set
from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test = train_test_split(x, y, test_size = 0.25, random_state = 0)

# Feature Scaling
from sklearn.preprocessing import StandardScaler
sc = StandardScaler()
x_train = sc.fit_transform(x_train)
x_test = sc.transform(x_test)
```

Output:



	User ID	Gender	Age	EstimatedSalary	Purchased
0	15624510	Male	19	19000	0
1	15810944	Male	35	20000	0
2	15668575	Female	26	43000	0
3	15603246	Female	27	57000	0
4	15804002	Male	19	76000	0
5	15728773	Male	27	58000	0
6	15598044	Female	27	84000	0
7	15694829	Female	32	150000	1
8	15600575	Male	25	33000	0
9	15727311	Female	35	65000	0
10	15570769	Female	26	80000	0
11	15606274	Female	26	52000	0
12	15746139	Male	20	86000	0
13	15704987	Male	32	18000	0
14	15628972	Male	18	82000	0
15	15697686	Male	29	80000	0
16	15733883	Male	47	25000	1
17	15617482	Male	45	26000	1
18	15704583	Male	46	28000	1
19	15621083	Female	48	29000	1

```
# Fitting Naive Bayes to the Training set

from sklearn.naive_bayes import GaussianNB

classifier = GaussianNB()

classifier.fit(x_train, y_train)


# Predicting the Test set results

y_pred = classifier.predict(x_test)

y_pred
```

Output:

```
array([1, 0, 1, 0, 0])
```

```
# Making the Confusion Matrix

from sklearn.metrics import confusion_matrix

cm = nm.array(confusion_matrix(y_test, y_pred))

cm
```

Output:

```
array([[3, 0],
        [0, 2]])
```

Result:

The above Naive Bayesian classifier program is successfully executed and successfully computed the accuracy of the classifier.