

```

#importing libraries

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt


#importing datasets

dataset = pd.read_csv("bank.csv", sep = ";")

X = dataset.iloc[:, :-1].values
Y = dataset.iloc[:, 16].values


#using encoding

from sklearn.preprocessing import LabelEncoder ,OneHotEncoder

labelencoder_X = LabelEncoder()

X[:,1] = labelencoder_X.fit_transform(X[:,1])
X[:,2] = labelencoder_X.fit_transform(X[:,2])
X[:,3] = labelencoder_X.fit_transform(X[:,3])
X[:,4] = labelencoder_X.fit_transform(X[:,4])
X[:,6] = labelencoder_X.fit_transform(X[:,6])
X[:,7] = labelencoder_X.fit_transform(X[:,7])
X[:,8] = labelencoder_X.fit_transform(X[:,8])
X[:,10] = labelencoder_X.fit_transform(X[:,10])
X[:,15] = labelencoder_X.fit_transform(X[:,15])


Y = labelencoder_X.fit_transform(Y)

onehotencoder = OneHotEncoder(categorical_features= [1,2,3,4,6,7,8,10,15])

X = onehotencoder.fit_transform(X).toarray()

```

```
#splitting dataset into training and testing dataset
from sklearn.model_selection import train_test_split
X_train, X_test, Y_train, Y_test = train_test_split(X,Y, test_size = .1, random_state = 1)

#normalizing
from sklearn.preprocessing import StandardScaler
sc_X = StandardScaler()
X_train = sc_X.fit_transform(X_train)
X_test = sc_X.fit_transform(X_test)

#fitting of the naive bayes to training set
from sklearn.naive_bayes import GaussianNB
classifier = GaussianNB()
classifier.fit(X_train,Y_train)

#predict
Y_pred = classifier.predict(X_test)

#making confusion matrix
from sklearn.metrics import confusion_matrix
cm = confusion_matrix(Y_test,Y_pred)

# confusion matrix
```

C:\Users\shubham b thorat\ML_self\naive_bayes - Spyder (Python 3.6)

File Edit Search Source Run Debug Consoles Projects Tools View Help

Project explorer

- naive_bayes
 - ad5.csv
 - bank.csv
 - salary.csv
 - untitled0.py

Editor - C:\Users\shubham b thorat\ML_self\naive_bayes\untitled1.py

```
15 X = dataset.iloc[:, :-1].values
16 Y = dataset.iloc[:, 16].values
17
18 #using encoding
19 from sklearn.preprocessing import LabelEncoder, OneHotEncoder
20 labelencoder_X = LabelEncoder()
21 X[:, 1] = labelencoder_X.fit_transform(X[:, 1])
22 X[:, 2] = labelencoder_X.fit_transform(X[:, 2])
23 X[:, 3] = labelencoder_X.fit_transform(X[:, 3])
24 X[:, 4] = labelencoder_X.fit_transform(X[:, 4])
25 X[:, 6] = labelencoder_X.fit_transform(X[:, 6])
26 X[:, 7] = labelencoder_X.fit_transform(X[:, 7])
27 X[:, 8] = labelencoder_X.fit_transform(X[:, 8])
28 X[:, 10] = labelencoder_X.fit_transform(X[:, 10])
29 X[:, 15] = labelencoder_X.fit_transform(X[:, 15])
30
31 Y = labelencoder_X.fit_transform(Y)
32 onehotencoder = OneHotEncoder()
33 X = onehotencoder.fit_transform(X).toarray()
34
35
36 #splitting dataset into training and testing dataset
37 from sklearn.model_selection import train_test_split
38 X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size = 0.3, random_state = 1)
39
40 #normalizing
41 from sklearn.preprocessing import StandardScaler
42 sc_X = StandardScaler()
43 X_train = sc_X.fit_transform(X_train)
44 X_test = sc_X.fit_transform(X_test)
45
46 #fitting of the naive bayes
47 from sklearn.naive_bayes import GaussianNB
48 classifier = GaussianNB()
49 classifier.fit(X_train, Y_train)
50
51 #predict
52 Y_pred = classifier.predict(X_test)
53
54 #making confusion matrix
55 from sklearn.metrics import confusion_matrix
56 cm = confusion_matrix(Y_test, Y_pred)
```

cm - Numpy array

	0	1
0	355	43
1	30	25

Format Resize Background color OK Cancel

Variable explorer

Name	Type	Size	Value
Y	int64	(4521,)	[0 0 0 ..., 0 0 0]
Y_pred	int64	(453,)	[0 0 0 ..., 0 0 0]
Y_test	int64	(453,)	[0 0 0 ..., 0 0 0]
Y_train	int64	(4068,)	[1 0 0 ..., 1 0 0]
cm	int64	(2, 2)	[[355 43] [30 25]]

dataset: DataFrame (4521, 17) Column names: age, job, marital, education, default, balance, housing, ...

Python console

```
...: #splitting dataset into training and testing dataset
...: from sklearn.model_selection import train_test_split
...: X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size = 0.3, random_state = 1)
...:
...: #normalizing
...: from sklearn.preprocessing import StandardScaler
...: sc_X = StandardScaler()
...: X_train = sc_X.fit_transform(X_train)
...: X_test = sc_X.fit_transform(X_test)
...:
...: #fitting of the naive bayes to training set
...: from sklearn.naive_bayes import GaussianNB
...: classifier = GaussianNB()
...: classifier.fit(X_train, Y_train)
...:
...: #predict
...: Y_pred = classifier.predict(X_test)
...:
...: #making confusion matrix
...: from sklearn.metrics import confusion_matrix
...: cm = confusion_matrix(Y_test, Y_pred)
```

In [2]:

Permissions: RW End-of-lines: CRLF Encoding: UTF-8 Line: 32 Column: 75 Memory: 48 %

ENG 11:34 PM 7/9/2018