# Topic: Naive Bayes

**Name: Silpa Batch Id:**  **Topic: Naïve Bayes**

**Problem statement:**

1.) Prepare a classification model using Naive Bayes for Salary dataset, train and test datasets are given separately use both datasets for model building.

Business objective:

By implementing Navie Bayes for salary dataset predictive model is built. With this model which category employee receives salary above 50k.

Step1:

#Data preprocessing

loading of data sets of both train and test data separately and then EDA is done on the data.

checking for Nan values in data by isnull() function. This data contains no null values . the first few rows of data is viewed by head() function in pandas. later removal of few columns.

The categorical data is transformed in to binary by using get\_dummies function

\*salary\_train=pd.get\_dummies(salary\_train,columns=['workclass','occupation','sex'])

the whole data of both test and train are split into input and target data

\*salary\_train\_input=salary\_train.iloc[:,0:26]

\*salary\_train\_target=salary\_train.iloc[:,[6]]

\*salary\_test\_target=salary\_test.iloc[:,[6]]

\*salary\_test\_input=salary\_test.iloc[:,0:26]

Step2:

#Model building

Classification model is built by Navie bayes algorithm ,for that GaussianNB module is imported from sklearn.navie\_bayes .

\*from sklearn.naive\_bayes import GaussianNB

after model is built it is fitted to train data in order to train it

\*model.fit(salary\_train\_input,salary\_train\_target.values.ravel())

#Accuracy is measured by using score by testing model on test data

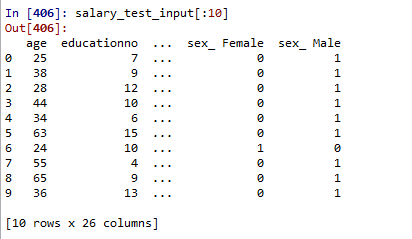
\*model.fit(salary\_train\_input,salary\_train\_target.values.ravel())

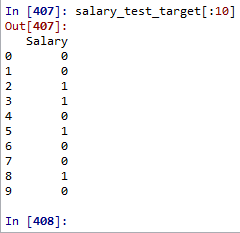


##checking of first 10 samples of test data

\*salary\_test\_input[:10]

\*salary\_test\_target[:10]





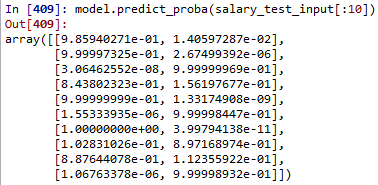
Step3:

##Prediction

prediction is determined by using predict function .predicted values are compared to the test values.



by propability



Summary:

By checking both test and predict values the model is of good.

2. Problem Statement: -

This dataset contains information of users in social network. This social network has several business clients which can put their ads on social network and one of the Client has a car company who has just launched a luxury SUV for ridiculous price. Build the Bernoulli Naïve Bayes model using this dataset and classify which of the users of the social network are going to purchase this luxury SUV.

Purchased: - 1 and Not Purchased: - 0

Business objective:

By implementing Navie Bayes for salary dataset predictive model is built. With this model which category employee receives salary above 50k.

Step1:

#Data preprocessing

EDA is done on the data.

checking for Nan values in data by isnull() function. This data contains no null values . the first few rows of data is viewed by head() function in pandas. later removal of few columns.

The categorical data is transformed in to binary by using get\_dummies function

\*car\_ad=pd.get\_dummies(car\_ad,columns=['Gender'])

the whole data of both test and train are split into input and target data

\*car\_ad\_target=car\_ad.iloc[:,[2]]

\*car\_ad\_input=car\_ad.drop(['Purchased'],axis=1)

Data is split in to train and test data by using train\_test\_split module from sklearn.model\_selection

from sklearn.model\_selection import train\_test\_split

x\_train, x\_test, y\_train, y\_test = train\_test\_split(car\_ad\_input,car\_ad\_target, test\_size = 0.2)

Step2:

#Model building

Classification model is built by Navie bayes algorithm ,for that GaussianNB module is imported from sklearn.navie\_bayes .

\*from sklearn.naive\_bayes import GaussianNB

after model is built it is fitted to train data in order to train it

\*model.fit(salary\_train\_input,salary\_train\_target.values.ravel())

#Accuracy is measured by using score by testing model on test data

\* model.fit(x\_train, y\_train)

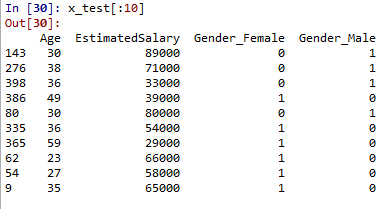
and score is obtained by

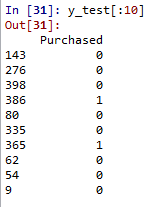


##checking of first 10 samples of test data

x\_test[:10]

y\_test[:10]





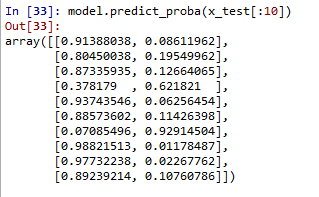
Step3:

##Prediction

prediction is determined by using predict function .predicted values are compared to the test values



by probability



Problem Statement: -

In this case study you have been given with tweeter data collected from an anonymous twitter handle, with the help of Naïve Bayes algorithm predict a given tweet is Fake or Real about real disaster occurring.

Business objective:

By implementing Navie Bayes for salary dataset predictive model is built. With this model which tweets are real and which are fake.

Step1:

#Data preprocessing

The given data mostly contains textual data, so text mining concepts are used. data is preprocessed by using preprocessing techniques of tokenization and stemming techniques. Data is cleaned by using customized cleaning\_text() function. Later it applied on model and checked its functionality.

\*cleaning\_text("Residents Return To Destroyed Homes As Washington Wildfire Burns on http://t.co/UcI8stQUg1")



the same way above function is applied on textual data column and then data is splitted in to both test and train sets

from sklearn.model\_selection import train\_test\_split

X\_train, X\_test, y\_train, y\_test = train\_test\_split(twitter\_data.text,twitter\_data.target)

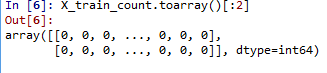
Step2:

#Model building

Before building model textual data is converted to count matrix by word frequencies found by count vectoriser

from sklearn.feature\_extraction.text import CountVectorizer

X\_train\_count = v.fit\_transform(X\_train.values)



Classification model is built by Navie bayes algorithm ,for that multinomialNB module is imported from sklearn.navie\_bayes .

\*from sklearn.naive\_bayes import multinomialNB

after model is built it is fitted to train data in order to train it

\*model.fit(X\_train\_count,y\_train)



#Accuracy is measured by using score by testing model on test data

and score is obtained by

X\_test\_count = v.transform(X\_test)

model.score(X\_test\_count, y\_test)



Summary:

By checking both test and predict values the model is of good.