Title : Naive- Bayes Algorithm

Problem Statement :

Download Pima Indians Diabetes dataset. Use Naive Bayes algorithm for classification.

1. load the data into csv file and split it into training and test datasets.
2. Summarize properties in the training dataset so that we can calculate probability and make predictions.
3. Classify samples from the test dataset and a summarized training dataset.

Objective : - To learn classification algorithms like Naives-Bayes
- To implement such algorithms to predict data.

Outcomes : We will be able to -
- learn classification algorithms.
- make predictions using the training datasets.

S/W & H/W requirements : - OS : Fedora / Ubuntu
- Python & Libraries

Theory :

A. Baye's Theorem :

- It is a way of finding a probability, when we know certain other probabilities.
- Formula :

$$P(A/B) = \frac{P(A) \cdot P(B/A)}{P(B).}$$

where,

P(A/B) ≡ how often A happens given that B happens

P(B/A) ≡ how often B happens given that A happens.

P(A) ≡ how likely A is on its own.

P(B) ≡ how likely B is on its own.

Example:

If dangerous fires are rare (1%) but smoke is fairly common (10%) due to barbeques, and 90% of dangerous fires make smoke then,

$$P(fire/smoke) = \frac{P(fire) \cdot P(smoke/fire)}{P(smoke)}$$

$$= \frac{0.01 \times 0.9}{0.1} = 9\%$$

∴ Probability of dangerous fire when there is smoke = 9%

B] Naive-Bayes classification:

- It is a simple, yet effective and commonly used, machine learning classifier.
- It is a probabilistic classifier that makes classifications using the maximum Aposteriori decision rule in a Bayesian setting. It can be represented using a very simple Bayesian network.
- It is especially popular for text classification and is a traditional solution for problem such as spam detection.

c] Applications :

1. Real time prediction :

Naive-Bayes is an eager learning classifier and it is very fast. Thus, it could be used to make predictions in real time.

2. Multi-class prediction :

This algorithm is also well known for multi-class prediction feature. Here, we can predict the probability of multi-classes of target variable.

Test case :

Input : Diabetes dataset

Output : Confusion Matrix

	0	1
0	125	37
1	25	43

Accuracy : 0.7304

Test set was 30% of the dataset and 73% of predicted values were obtained correctly.

Conclusion :

Thus, we successfully learnt and implemented Naive-Bayes classification algorithm.