

Practical 6

AIM: IMPLEMENT SIMPLE NAVIE BAYESCLASSIFICATIONALGORITHM. USING PYTHON ON IRIS.CSV DATASET.

REQUIREMENT:

AnacondaInstaller

Windows10OS

JupyterNotebook

THEORY:

Naïve Bayes Classifier:

- Naïve Bayes is a **supervised learning algorithm** used for **classification problems**.
 - It is based on **Bayes' Theorem** and works on **probability**.
 - It is simple, fast, and effective, especially for **text classification** (spam detection, sentiment analysis).
 - It assumes that all features are **independent** of each other.
-

Why it is called Naïve Bayes:

- **Naïve:** Assumes features are independent.
 - **Bayes:** Based on Bayes' Theorem (probability concept).
-

Advantages:

- Fast and easy to implement
 - Works well with small datasets
 - Suitable for multi-class problems
 - Performs well with categorical data
-

Disadvantages:

- Assumes feature independence (not always true)
- Cannot handle unseen categories well (zero frequency problem)
- Probability predictions may not always be accurate

Applications of Naïve Bayes:

- Used for **real-time predictions** due to its speed
 - Useful for **multi-class classification problems**
 - Applied in **spam email filtering** (e.g., Gmail)
 - Used in **sentiment analysis** (positive/negative reviews)
 - Helps in **recommendation systems**
-

Types of Naïve Bayes Classifier:

- **Bernoulli Naïve Bayes:** Works with **binary data** (True/False, Yes/No)
- **Multinomial Naïve Bayes:** Used for **text classification** based on word frequency
- **Gaussian Naïve Bayes:** Used for **continuous data** assuming normal distribution

LibrariesUsed:

Pandas:

Sklearn

CONCLUSION:

In this experiment we have studied about Navie bayes algorithm and implemented it an the iris dataset for spam filtration