Practical No.6

Title: Write a program to implement the naïve Bayesian classifier for a sample training data set stored as a .CSV file. Compute the accuracy of the classifier, considering few test data sets.

Objective

The objective of this practical is to implement a Naïve Bayes Classifier using a dataset stored in a .CSV file. The model will be trained on labeled data and tested with new observations. The accuracy of the classifier will be computed to evaluate its performance.

Introduction to Naïve Bayes Classifier

The Naïve Bayes Classifier is a probabilistic classification algorithm based on Bayes' Theorem. It assumes that the features are conditionally independent given the class label, which simplifies probability computations.

Bayes' Theorem:

P(C|X)=P(X|C)P(C)P(X)P(C|X)=\frac{P(X|C)
P(C)}{P(X)}P(C|X)=P(X)P(X|C)P(C)

where:

- P(C|X)P(C|X)P(C|X) is the posterior probability (probability of class CCC given features XXX).
- P(X|C)P(X|C)P(X|C) is the likelihood (probability of features XXX given class CCC).
- P(C)P(C)P(C) is the prior probability (probability of class CCC occurring).
- P(X)P(X)P(X) is the evidence (total probability of features XXX).

Types of Naïve Bayes Classifiers

1. Gaussian Naïve Bayes: Assumes features follow a normal distribution.
2. Multinomial Naïve Bayes: Suitable for text classification problems.
3. Bernoulli Naïve Bayes: Used for binary feature datasets.

This classifier is widely used in spam filtering, sentiment analysis, medical diagnosis, and text classification due to its efficiency and speed.

Dataset Description

The dataset is stored in a .CSV file and consists of multiple features (independent variables) and a target variable (dependent variable) representing the class labels. The data should be well-structured, with rows representing individual observations and columns representing attributes.

Implementation Steps

Step 1: Load and Explore the Dataset

- Read the dataset from a .CSV file.
- Display dataset information, including column names, data types, and summary statistics.
- Check for and handle missing values appropriately.

Step 2: Data Preprocessing

- Convert categorical variables into numerical format using encoding techniques if necessary.
- Normalize numerical features if required for better model performance.
- Split the dataset into training data (80%) and testing data (20%).

Step 3: Train the Naïve Bayes Classifier

- Select an appropriate Naïve Bayes model (Gaussian, Multinomial, or Bernoulli).
- Train the model using the training dataset by fitting the feature variables to the target variable.

Step 4: Make Predictions

- Use the trained model to predict class labels for the test dataset.
- Obtain class probabilities for further analysis.

Step 5: Compute Model Accuracy

The accuracy of the model is computed by comparing predicted labels with actual labels. Several evaluation metrics are used:

Accuracy Score

- Measures the proportion of correctly classified instances out of the total observations.

Accuracy=Correct PredictionsTotal PredictionsAccuracy = \frac{Correct\ Predictions}{Total\ Predictions}Accuracy=Total PredictionsCorrect Predictions

Confusion Matrix

- Summarizes the correct and incorrect classifications, showing:
    - True Positives (TP) – Correctly predicted positive instances.
    - True Negatives (TN) – Correctly predicted negative instances.
    - False Positives (FP) – Incorrectly predicted positive instances.
    - False Negatives (FN) – Incorrectly predicted negative instances.

Precision, Recall, and F1-Score

- Precision: Measures how many of the predicted positive instances are actually positive.
- Recall (Sensitivity): Measures how many of the actual positive instances were correctly classified.
- F1-Score: A balance between precision and recall for an overall assessment of model performance.

ROC Curve and AUC Score

- The ROC (Receiver Operating Characteristic) curve plots the True Positive Rate (TPR) against the False Positive Rate (FPR).
- The AUC (Area Under the Curve) score quantifies the model's ability to distinguish between classes.

Conclusion

This practical demonstrates the implementation of the Naïve Bayes Classifier for classification tasks using a dataset stored in a .CSV file. The model was trained and evaluated using accuracy, confusion matrix, precision, recall, F1-score, and AUC score.