ASSIGNMENT 5

TITLE: Implement K-Nearest Neighbors algorithm on diabetes.csv dataset. Compute confusion matrix, accuracy, error rate, precision and recall on the given dataset.

Dataset link : https://www.kaggle.com/datasets/abdallamahgoub/diabete

Introduction

K-Nearest Neighbors (K-NN) is a simple and effective classification algorithm used for both supervised and unsupervised machine learning tasks. In this lab, we will implement the K-NN algorithm to classify diabetes outcomes using the diabetes dataset.

Prerequisites

Before starting this lab, ensure you have the following installed:

- Python (3.6 or higher)
- Jupyter Notebook (optional but recommended)
- Required Python libraries: pandas, scikit-learn (sklearn)

Dataset

Download the diabetes dataset from this Kaggle link and save it as **diabetes.csv** in your working directory.

Theory

K-Nearest Neighbors (K-NN)

K-NN is a classification algorithm that works based on the assumption that data points with similar features tend to belong to the same class. It makes predictions by identifying the K-nearest data points to a given test point and assigning the class label that is most common among those neighbors.

Evaluation Metrics

1. **Confusion Matrix**: A confusion matrix is a table used to evaluate the performance of a classification algorithm. It provides information about the true positives, true negatives, false positives, and false negatives.
2. **Accuracy**: Accuracy measures the percentage of correctly classified instances out of the total instances in the dataset.
3. **Error Rate**: Error rate is the complement of accuracy. It measures the percentage of incorrectly classified instances.

4. **Precision**: Precision is the ratio of true positives to the total predicted positives. It measures the ability of the model to avoid false positive predictions.

5. **Recall**: Recall is the ratio of true positives to the total actual positives. It measures the ability of the model to find all relevant instances.

IMPLEMENTATION STEPS:

- Load and Explore the Dataset
- Preprocess the Data

  Handle missing values if any.

  Split the data into features (X) and the target variable (y).
- Split the Data
- Create and Train the K-NN Model
- Make Predictions
- Evaluate the Model

Conclusion

In this lab, you have learned how to implement the K-Nearest Neighbors algorithm on the diabetes dataset. You also learned about important evaluation metrics such as the confusion matrix, accuracy, error rate, precision, and recall to assess the model's performance.