Department of Computer Engineering
BE Computer-B (2025-26 Sem I)
LP-III Machine Learning
Practical Assignment 4: K-Nearest Neighbors

[CO1, CO3, BT: L3 (Apply)] [Max Marks: 10]

Problem Definition: 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/diabetes

Learning Outcomes:

By completing this practical, students will be able to:

Learning Outcome	**Bloom's Taxonomy Level (BT)**
Understand the working principle of KNN algorithm	Understand (Level 2)
Apply KNN for classification tasks	Apply (Level 3)
Implement KNN on real-world diabetes dataset	Apply (Level 3)
Compute evaluation metrics: accuracy, error rate, precision, recall	Analyze (Level 4)
Interpret confusion matrix results	Evaluate (Level 5)
Optimize KNN parameters (k-value)	Create (Level 6)

Software / Hardware Requirements

Software:

- Python 3.x
- Libraries: Pandas, NumPy, Scikit-learn, Matplotlib, Seaborn
- Jupyter Notebook / Google Colab

Hardware:

- Minimum 4 GB RAM
- Intel i3 or higher
- Stable internet connection (for data access)
- Recommended: GPU support for faster training

Theory:

The aim of this experiment is to implement the K-Nearest Neighbors algorithm on the PIMA Indians Diabetes Dataset to classify whether a patient is diabetic or not based on given medical parameters. The dataset contains diagnostic measurements such as glucose level, blood pressure, insulin level, body mass index (BMI), and others. The model will be trained and tested to evaluate its classification performance using metrics like confusion matrix, accuracy, error rate, precision, and recall.

1. Dataset Description:

The **PIMA Indians Diabetes Dataset** contains medical diagnostic measurements collected from female Pima Indian patients aged **21 years or older**. The goal is to use these features to predict whether a patient has diabetes (1) or not (0).

Attributes in diabetes.csv:

Column Name	Description	Type / Unit	Example Value
Pregnancies	Number of times pregnant	Integer	6
Glucose	Plasma glucose concentration (2 hours in an oral glucose tolerance test)	mg/dL	148
BloodPressure	Diastolic blood pressure	mm Hg	72
SkinThickness	Triceps skin fold thickness	mm	35
Insulin	2-hour serum insulin	µU/mL	0
BMI	Body Mass Index	kg/m²	33.6
DiabetesPedigreeFunction	Diabetes pedigree function (genetic influence)	Numeric	0.627
Age	Age of the patient	Years	50
Outcome	Class variable: 0 = No Diabetes, 1 = Diabetes	Binary	1

Data Characteristics:

- All patients are female of Pima Indian heritage.
- Missing values are sometimes encoded as **0** in Glucose, BloodPressure, SkinThickness, Insulin, and BMI — these require preprocessing.
- Dataset size: **768 rows × 9 columns**.

Acknowledgement:

The dataset is referred from Kaggle: https://www.kaggle.com/datasets/abdallamahgoub/diabetes

2. K-Nearest Neighbors (KNN)

KNN is a **non-parametric**, **instance-based** supervised machine learning algorithm used for classification and regression tasks. In classification, it predicts the class label based on the **majority vote of its k-nearest neighbors** in the feature space.

Working Principle:

1. Choose the number of neighbors **k**.
2. Calculate the distance between the query point and all points in the training dataset (commonly **Euclidean distance**).
3. Select the **k** nearest neighbors based on the smallest distances.
4. Assign the most frequent class among the neighbors to the query point.

Mathematical Model

Euclidean Distance Formula:

For two points $X=(x_1,x_2,...,x_n)$ and $Y=(y_1,y_2,...,y_n)$:

$$d(X,Y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$$

Where:

- nnn = number of features
- x_i,y_i = values of feature i for X and Y

Model Evaluation Metrics:

1. **Accuracy**: {TP + TN}/{TP + TN + FP + FN}
2. **Confusion Matrix**:

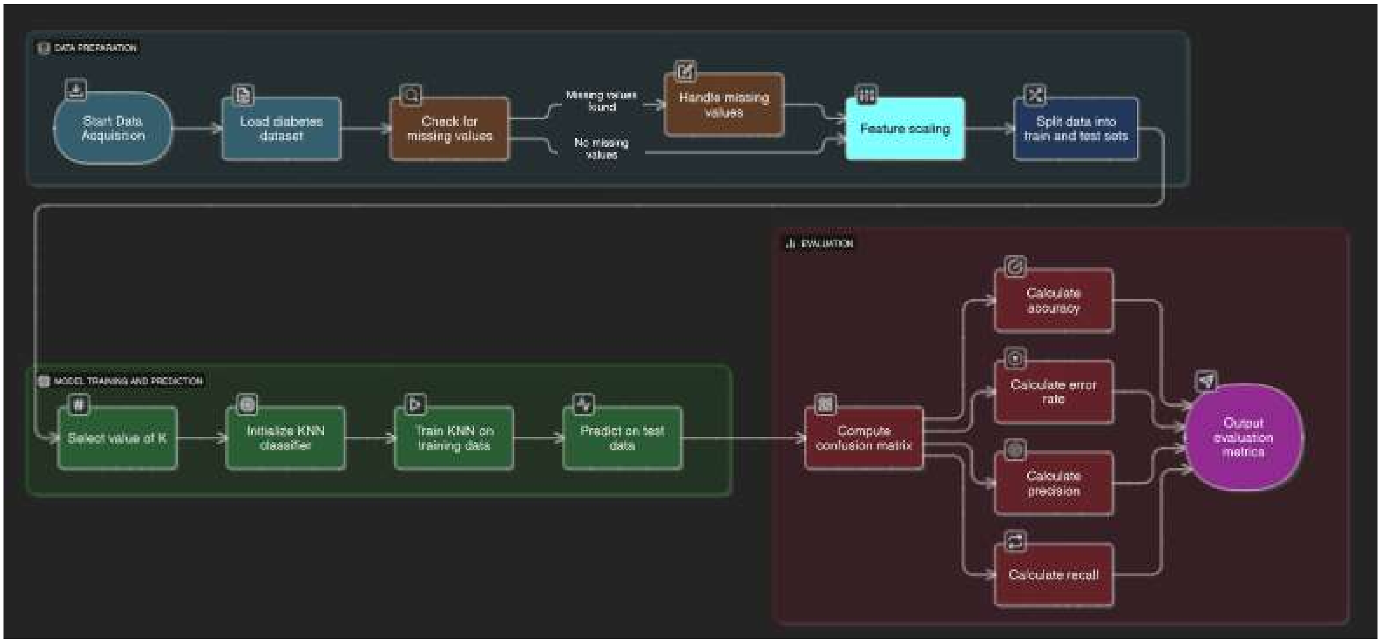
Actual / Predicted	Positive	Negative
Positive	TP	FN
Negative	FP	TN

3. **Error Rate:** 1-Accuracy
4. **Precision**: {TP}/{TP + FP}
5. **Recall**: {TP}/{TP + FN}
6. **F1 Score**: Harmonic mean of precision and recall ((2 * P*R) / (P + R))

Algorithm

1. Load the dataset (`diabetes.csv`).
2. Preprocess the data (handle missing values if needed, normalize features).
3. Split the dataset into training and testing sets.
4. Select value of **k** (e.g., k=5).
5. Train the **KNN classifier** on the training data.
6. Predict output on the test set.
7. Compute **confusion matrix**, **accuracy**, **error rate**, **precision**, and **recall**.
8. Display results and interpret.

Flow Chart



Result:

After running the implementation:

- A **confusion matrix** will be generated.
- Accuracy, error rate, precision, and recall values will be calculated.
- The classification performance of KNN will be evaluated on the diabetes dataset.

Metric	Value
Accuracy	
Error Rate	
Precision	
Recall	
F1-Score	

(Insert Screenshots: Accuracy, Confusion Matrix, Accuracy/loss curve from training)

Conclusion

The KNN classifier is effective for medical diagnosis tasks such as predicting diabetes from clinical measurements. Choosing an appropriate value of k and proper preprocessing of features significantly affects the accuracy. The computed evaluation metrics help determine the classifier's reliability in real-world medical applications.

Viva Questions

Question	BT Level
What is the principle behind KNN?	Understand (L2)

Why is KNN called a lazy learner?	Understand (L2)
How do you choose the value of k?	Apply (L3)
What is the role of distance metrics in KNN?	Analyze (L4)
Explain the impact of feature scaling on KNN performance.	Analyze (L4)
Differentiate between KNN and other classification algorithms.	Evaluate (L5)
How can KNN be optimized for large datasets?	Create (L6)

Rubrics for Evaluation (Example)

Criteria	**Best**	**Good**	**Average**	**Poor**
Correctness of Program / Demonstration of Program [4]	Perfect [4]	Can be better [3]	Satisfied [2]	Poor [0]
File Submission/Documentation [2]	Perfect [2]	Can be better [1.5]	Satisfied [1]	Poor [0]
Timely Submission [2]	On Time [2]	After 1 week [1.5]	After 1.5 week [1]	At the end of semester [0]
Viva [2]	Perfect [2]	Can be better [1.5]	Satisfied [1]	Poor [0]

Dr. Nikita Singhal
(Subject Incharge)

DR SR Dhore
(HoD)