Aim : Predicting Housing Prices using K-Nearest Neighbors (KNN) Algorithm

Objective: The objective of this lab is to introduce students to the K-Nearest Neighbors (KNN) algorithm and its application in predicting housing prices. Students will gain hands-on experience implementing KNN for regression tasks and evaluating the model's performance.

Prerequisites:

- Basic understanding of Python programming.

- Familiarity with basic concepts of machine learning.

- Understanding of regression and supervised learning.

Tools and Libraries:

- Python (3.x recommended)

- Jupyter Notebook

- NumPy

- Pandas

- Scikit-learn

- Matplotlib

- Seaborn

Lab Outline:

1. Introduction to Regression and Predicting Housing Prices:

- Briefly explain the concept of regression and its application in predicting numeric values.

- Discuss the importance of predicting housing prices in real estate and finance.

2. Overview of K-Nearest Neighbors (KNN) Algorithm:

- Explain the KNN algorithm for regression tasks.

- Discuss how KNN predicts the target variable based on the average or weighted average of its k-nearest neighbors.

3. Installing Required Libraries:

- Instruct students to install necessary Python libraries using the following commands in a Jupyter Notebook:

```python
!pip install numpy pandas scikit-learn matplotlib seaborn
```

4. Loading and Exploring the Dataset:

- Provide a housing dataset (e.g., Boston Housing Dataset) and guide students through loading and exploring the data using Pandas.

- Discuss the features and target variable in the dataset.

5. Preprocessing the Data:

- Discuss the importance of data preprocessing in machine learning.

- Guide students through handling missing values, scaling features, and splitting the data into training and testing sets.

6. Implementing K-Nearest Neighbors (KNN) Algorithm:

- Walk students through the implementation of the KNN algorithm using Scikit-learn.

- Discuss hyperparameters such as the number of neighbors (k) and the distance metric.

7. Training the KNN Model:

- Instruct students on how to train the KNN model using the training data.

- Discuss the role of the k-nearest neighbors in predicting housing prices.

8. Evaluating the Model:

- Teach students how to evaluate the performance of the KNN model on the testing data.

- Introduce metrics such as Mean Absolute Error (MAE) or Root Mean Squared Error (RMSE) for regression tasks.

9. Tuning Hyperparameters:

- Discuss the importance of hyperparameter tuning in improving model performance.

- Guide students through experimenting with different values of k and evaluating the impact on the model.

10. Visualizing Predictions:

- Use Matplotlib and Seaborn to visualize the predicted housing prices against the actual prices.

- Discuss the strengths and limitations of the KNN model.

11. Conclusion and Discussion:

- Summarize the key concepts covered in the lab.

- Discuss potential challenges and considerations when applying the KNN algorithm to real-world housing price prediction scenarios.

12. Additional Challenges (Optional):

- Pose additional challenges for students to further enhance their understanding and skills, such as experimenting with different distance metrics or exploring alternative regression algorithms.

Resources:

- Provide additional resources, such as relevant research papers, online tutorials, and documentation for further exploration.

Assessment:

- Evaluate students based on their understanding of the KNN algorithm, successful implementation, and effective evaluation of the model's performance.

- Encourage students to submit their Jupyter Notebooks along with a brief report discussing their observations, insights, and any improvements made to the model.

Result/Conclusion: By following this lab content, students should gain practical experience in using the K-Nearest Neighbors algorithm for predicting housing prices, understanding its strengths, and exploring ways to improve its performance.

Frequently Asked Questions (FAQ)

1) How does the K-Nearest Neighbors (KNN) algorithm work in the context of predicting housing prices?

2) What factors or features are typically considered when using KNN for predicting housing prices?

3) How do you handle missing or noisy data when using the KNN algorithm for housing price prediction?

4) Can you explain the impact of the "k" parameter on the performance of the KNN algorithm in predicting housing prices?

5) What are the limitations of the KNN algorithm in predicting housing prices, and how can they be addressed?