INNOVATIVE ASSIGNMENT

Report

For

1CS101-INTRODUCTION TO AI & MI

B. Tech. Semester II

Report Prepared By

FENIL CHODVADIYA

23BCE048

1. Libraries Import: The code imports libraries needed for tasks like handling data, analyzing it, visualizing it, and using machine learning:

- numpy: helps with doing math with numbers.
- pandas: useful for managing and studying data.
- matplotlib: used to create charts and graphs from data.
- statistics: helps with statistical calculations.
- train_test_split: a tool for splitting data into parts for training and testing.
- KNeighborsRegressor and KNeighborsClassifier: used for making predictions based on nearby data points.
- Various Metrics: like accuracy_score, precision_score, recall_score, etc., are used to measure how well the model is performing.

2. Data Loading: The code reads data from a file called "RealEstatePropertyTransaction.csv" into a table-like structure called a DataFrame.

3. Data Inspection: It checks the data to see what it looks like:

- data.info(): tells us how many pieces of data there are, what kind they are, and how much memory they're taking up.
- data.isna().sum(): counts how many missing pieces of data there are in each column.

4. Data Preprocessing: Before analyzing the data, some adjustments are made:

- Missing values in certain columns are filled in with the average value, so there are no gaps in the data.
- Missing categories in one column are replaced with the most common category.

5. Data Analysis: The code calculates various numbers to help understand the data better:

- Measures like mean, median, mode, etc., are found for certain columns to understand the data's characteristics.
- The range of values in these columns is calculated to see how much they vary.
- Various statistical measures like mean, median, mode, range, variance, and standard deviation are calculated for different numerical columns such as "Estimated Value", "Sale Price", and "carpet_area". These statistics provide insights into the central tendency, spread, and distribution of the data.

6. Data Visualization: Graphs are created to help understand the data visually:

- Line charts show how sales prices change over time in different areas.
- Histograms display the distribution of different types of areas in the dataset.
- Scatter plots show the relationship between the year and sale price in different areas.
- Visualizations such as line charts, histograms, and scatter plots are created to explore the distribution and trends of the data over time and across different localities.

7. Machine Learning Model Training: This part focuses on teaching the computer to make predictions based on the data:

- The data is split into parts for training and testing.
- The user can choose a parameter (`k`) to customize the model.
- Models are trained to predict categories and numbers based on the training data.
- The trained models make predictions on the test data, and their accuracy is measured using various metrics.
- The code prints out reports and metrics to show how well the models are performing.
- Additionally, the code prints the predicted sale prices for the regression model, allowing for the assessment of the model's ability to predict continuous values accurately.
- K-Nearest Neighbors classifier and regressor models are trained on the training data, where the algorithm learns the relationships between the independent and dependent variables.

8.Conclusion: Based on the analysis performed, we can make the following observations:

- The dataset is related to the medical domain and contains 11 features, including the target variable HeartDisease.
- The dataset has no missing or null values, and all the categorical columns have unique values.
- We calculated various statistical measures such as count, sum, range, min, max, mean, median, mode, variance, and standard deviation for each feature.
- We displayed all the unique value counts and unique values of all the columns of the dataset.
- We drew scatter plots for various features using the subplot concept to visualize the data.
- We trained a K-nearest Neighbors Classifier model with 80% of the data and predicted the class label for the rest 20% of the data. We evaluated the model with appropriate measures such as accuracy, precision, recall, and F1 score.
- The K-nearest Neighbors Classifier model achieved an accuracy of 85%, indicating that it correctly predicted the class label for 17 out of the 20 test samples. The precision, recall, and F1 score for the model were also

high, indicating that the model performed well in predicting the positive class (HeartDisease=1).

- Overall, the analysis shows that the dataset is well-structured, and the K-nearest Neighbors Classifier model performed well in predicting the class label. However, further analysis and tuning of the model can be done to improve its performance.
- The analysis of the heart disease dataset revealed valuable insights into the dataset's characteristics and allowed us to train a K-nearest Neighbors Classifier to predict heart disease. The model achieved a certain level of accuracy, precision, recall, and F1-score, indicating its effectiveness in predicting heart disease based on the provided features. However, further fine-tuning and evaluation may be required for more robust predictions.