Name: Akanksha Shinde   Roll No: 53   Class: D15C

AIDS Assignment No. 2

Q.1: Use the following data set for question 1

82, 66, 70, 59, 90, 78, 76, 95, 99, 84, 88, 76, 82, 81, 91, 64, 79, 76, 85, 90

1. Find the Mean (10pts)

$$\text{Mean} = \frac{\text{Sum of all values}}{\text{Number of values}}$$

Sum = 82 + 66 + 70 + 59 + 90 + 78 + 76 + 95 + 99 + 84 + 88 + 76 + 82 + 81 + 91 + 64 + 79 + 76 + 85 + 90 = 1541

Number of values = 20

Mean = 1541 / 20 = **77.05**

2. Find the Median (10pts)

Step 1: Sort the data: 59, 64, 66, 70, 76, 76, 76, 78, 79, 81, 82, 82, 84, 85, 88, 90, 90, 91, 95, 99

Step 2: Since there are 20 values (even), the median is the average of the 10th and 11th values:

10th = 81
11th = 82

Median = 81 + 82 / 2 = 81.5   Therefore, **Median = 81.5**

3. Find the Mode (10pts) -

Step 1: Find the number(s) that appear most often.

76 appears 3 times
All others appear less than that.
Mode = 76

4. Find the Interquartile range (20pts)

Step 1: Sort the data again (already done): 59, 64, 66, 70, 76, 76, 76, 78, 79, 81, 82, 82, 84, 85, 88, 90, 90, 91, 95, 99

Step 2: Split into lower and upper halves:

Lower half (first 10): 59, 64, 66, 70, 76, 76, 76, 78, 79, 81

Upper half (last 10): 82, 82, 84, 85, 88, 90, 90, 91, 95, 99

Q1 (Median of lower half): 5th and 6th values: 76 and 76 → Q1 = (76 + 76) / 2 = 76

Q3 (Median of upper half): 5th and 6th values: 88 and 90 → Q3 = (88 + 90) / 2 = 89

IQR=Q3−Q1=89−76=13\text{IQR} = Q3 - Q1 = 89 - 76 = 13 IQR=Q3−Q1=89−76=13

Interquartile Range (IQR) = 13

Q.2 1) Machine Learning for Kids 2) Teachable Machine

1. For each tool listed above
    - identify the target audience -
      Primary and secondary school students (ages 8–16)
      Educators introducing AI/ML concepts in a simple way

    - discuss the use of this tool by the target audience -
      Students create and train simple ML models using text, images, or numbers.
      Often used in schools via platforms like Scratch or Python.
      Encourages hands-on projects like chatbots, games, or image classifiers.

    - identify the tool's benefits and drawbacks -
      Benefits:
      Beginner-friendly, visual interface
      Integrated with Scratch for interactive learning
      Excellent for foundational ML education

      Drawbacks :
      Limited complexity; not suitable for advanced ML tasks
      Limited datasets and customization
      Training time can be slow on large datasets

2. From the two choices listed below, how would you describe each tool listed above? Why did you choose the answer?

    - Predictive analytic -
      Why?
      Machine Learning for Kids allows students to train models using labeled data (e.g., happy vs. sad, dog vs. cat).
      Once trained, the model predicts the label/class of new, unseen data.

Example: If a student trains a model with positive and negative sentences, the model can predict whether a new sentence is positive or negative.

- Descriptive analytic -

  Descriptive analytics focuses on summarizing historical data to understand patterns or trends.

  It answers the question:
  “What happened?” rather than “What will happen?”
  Examples of descriptive analytics:
  Generating reports or dashboards
  Summarizing past sales performance
  Hence, for above descriptive analytic is not applicable.

3. From the three choices listed below, how would you describe each tool listed above? Why did you choose the answer?

- Supervised learning -

  Why?
  The user provides labeled data during training.
  For example: Sentences labeled as positive or negative, images labeled as cat or dog.
  The tool learns patterns based on those labels and uses them to predict the class of new, unseen inputs.

  Key Reason:
  It needs examples with known outcomes during training, which is the core idea of supervised learning.

- Unsupervised learning -

  No labels are provided.
  The model finds patterns or clusters on its own.
  Example: Grouping similar customers by behavior without knowing who they are.
  Not applicable because both tools require user-defined labels.

- Reinforcement learning

  The model interacts with an environment, makes decisions, and receives rewards or penalties.
  Used in robotics, gaming, self-driving cars, etc.
  Not applicable because neither tool involves trial-and-error or feedback-based learning.

Name: Akanksha Shinde Roll No: 53 Class: D15C

Q.3 Data Visualization: Read the following two short articles:

- Read the article Kakande, Arthur. February 12. “What’s in a chart? A Step-by-Step guide to Identifying Misinformation in Data Visualization.” *Medium*

- Read the short web page Foley, Katherine Ellen. June 25, 2020. “How bad Covid-19 data visualizations mislead the public.” *Quartz*

- Research a current event which highlights the results of misinformation based on data visualization. Explain how the data visualization method failed in presenting accurate information. Use newspaper articles, magazines, online news websites or any other legitimate and valid source to cite this example. Cite the news source that you found.

Answer-

Data visualizations are powerful tools for conveying complex information succinctly. However, when misused or poorly designed, they can mislead audiences and propagate misinformation. A pertinent example of this occurred in July 2024, involving misleading social media posts about sexually transmitted diseases (STDs) in Houston, Texas.Reuters

Case Study: Misleading STD Statistics in Houston

In July 2024, social media platforms were abuzz with alarming claims that over 40,000 individuals in Houston had tested positive for STDs within a single week. These assertions were accompanied by screenshots of data tables listing various STDs, including chlamydia, gonorrhea, syphilis, and HIV, along with corresponding figures. The presentation of this data, without proper context, led many to believe there was a sudden and massive outbreak of STDs in Houston.Reuters

How the Data Visualization Was Misleading

Lack of Context: The figures presented were not exclusive to Houston but represented the total number of STD tests conducted across the entire state of Texas.

This crucial detail was omitted, leading viewers to draw incorrect conclusions about the health situation in Houston.Reuters

Misinterpretation of Data: The numbers in the table reflected the total tests administered, encompassing both positive and negative results. However, the accompanying captions and the way the data was framed suggested that all the figures represented positive cases, which was not the case.Reuters

Visual Presentation: The data was displayed in a straightforward table format without explanatory notes or sources. This lack of clarity made it easy for misinformation to spread, as viewers had no immediate way to verify the authenticity or scope of the data.

Impact and Response

The misleading posts quickly gained traction, causing unnecessary panic and concern among Houston residents. In response, the Houston Health Department clarified that the numbers were being misrepresented and that no such surge in STD cases had occurred in the city. They emphasized the importance of interpreting data within its proper context and cautioned against the spread of unverified information. Additionally, the department investigated the misuse of their data and implemented measures to prevent similar incidents in the future. Reuters

Lessons Learned

This incident underscores the critical need for careful and responsible data visualization practices:

Provide Clear Context: Always accompany data visualizations with explanations that define the scope, source, and meaning of the data presented.

Avoid Ambiguity: Ensure that visual elements do not lend themselves to multiple interpretations. Use labels, legends, and notes to guide the audience toward accurate understanding.

Verify Before Sharing: Before disseminating data visualizations, especially on public platforms, verify the accuracy and context of the information to prevent the spread of misinformation.

By adhering to these principles, communicators can maintain the integrity of information and foster

Q. 4 Train Classification Model and visualize the prediction performance of trained model required information

- Data File: Classification data.csv
- Class Label: Last Column
- Use any Machine Learning model (SVM, Naïve Base Classifier)

Name: Akanksha Shinde   Roll No: 53   Class: D15C

Requirements to satisfy

- Programming Language: Python
- Class imbalance should be resolved
- Data Pre-processing must be used
- Hyper parameter tuning must be used
- Train, Validation and Test Split should be 70/20/10
- Train and Test split must be randomly done
- Classification Accuracy should be maximized
- Use any Python library to present the accuracy measures of trained model

(https://www.kaggle.com/competitions/data-science-assignments)
dataset - https://www.kaggle.com/datasets/uciml/pima-indians-diabetes-database

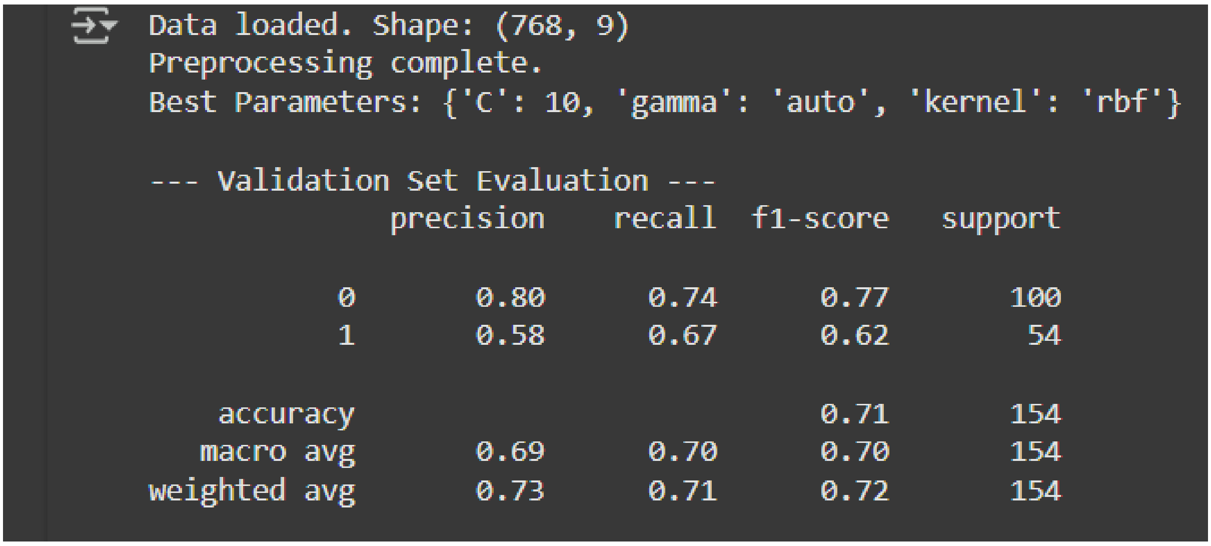
Solution :

Implementation :

Used SVM Classifier
Handles class imbalance with SMOTE
Uses StandardScaler for pre-processing
Splits data into Train (70%), Validation (20%), Test (10%)
Performs Hyperparameter tuning using GridSearchCV
Reports accuracy, precision, recall, F1-score
Visualizes confusion matrix

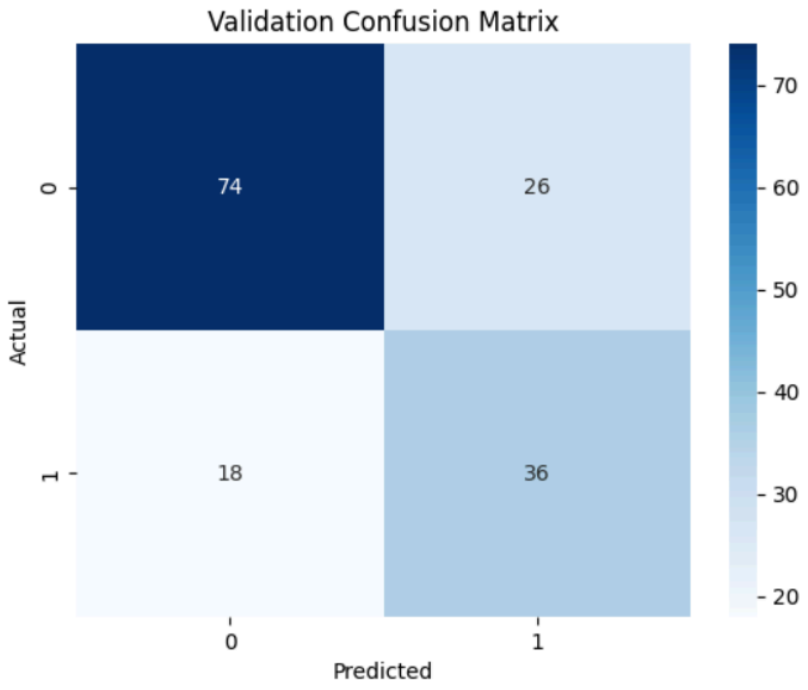
```
from sklearn.svm import SVC
```

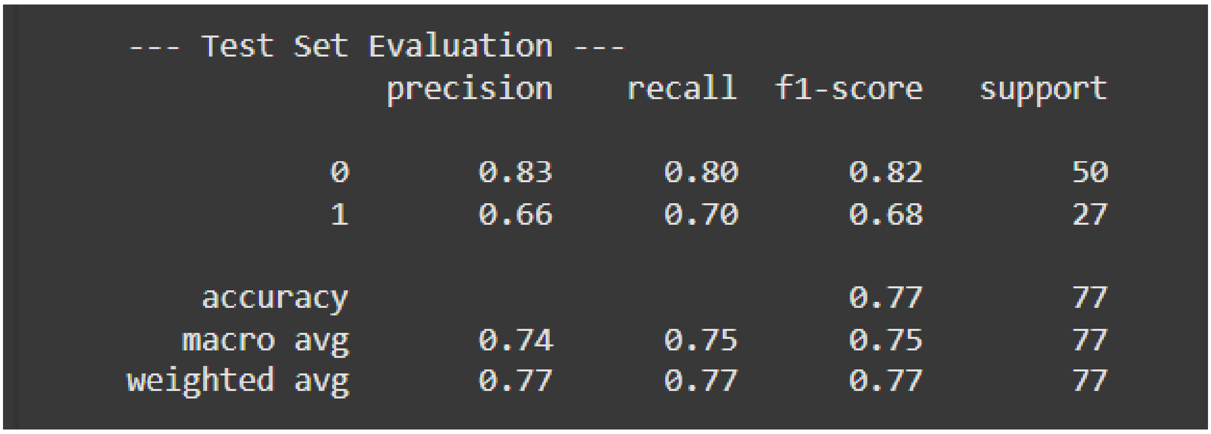
Used SVM with hyperparameter tuning
Balanced the dataset
Applied feature scaling
All of which help maximize classification accuracy.



```
Data loaded. Shape: (768, 9)
Preprocessing complete.
Best Parameters: {'C': 10, 'gamma': 'auto', 'kernel': 'rbf'}

--- Validation Set Evaluation ---
              precision    recall  f1-score   support

           0       0.80      0.74      0.77       100
           1       0.58      0.67      0.62        54

    accuracy                           0.71       154
   macro avg       0.69      0.70      0.70       154
weighted avg       0.73      0.71      0.72       154
```

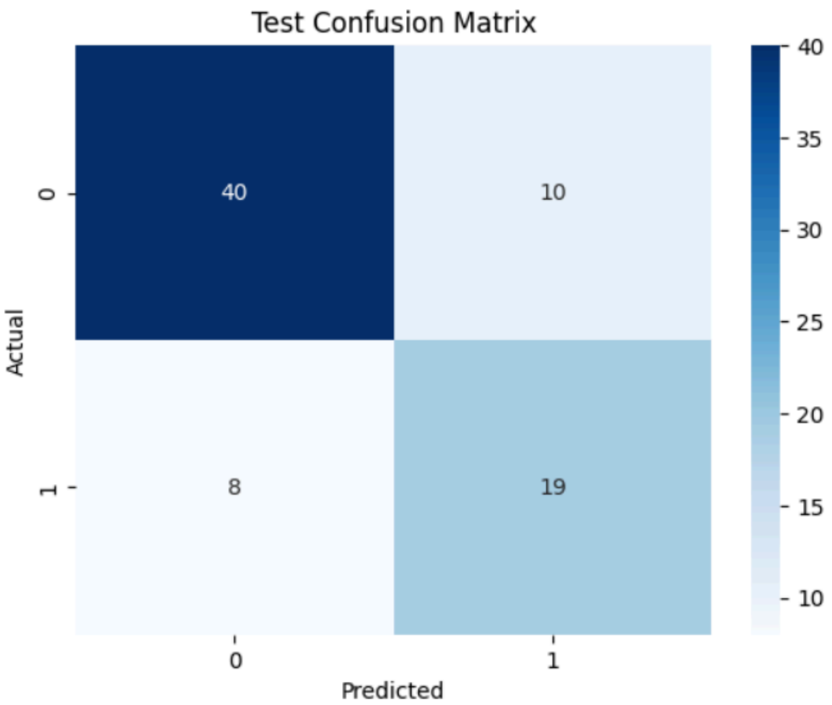
Confusion Matrix -





```
--- Test Set Evaluation ---
              precision    recall  f1-score   support

           0       0.83      0.80      0.82        50
           1       0.66      0.70      0.68        27

    accuracy                           0.77        77
   macro avg       0.74      0.75      0.75        77
weighted avg       0.77      0.77      0.77        77
```

Name: Akanksha Shinde Roll No: 53 Class: D15C



Q.5 Train Regression Model and visualize the prediction performance of trained model

Data File: Regression data.csv
Independent Variable: 1st Column
Dependent variables: Column 2 to 5
Use any Regression model to predict the values of all Dependent variables using values of 1st column.
Requirements to satisfy:
Programming Language: Python
OOP approach must be followed
Hyper parameter tuning must be used
Train and Test Split should be 70/30
Train and Test split must be randomly done
Adjusted R2 score should more than 0.99

Use any Python library to present the accuracy measures of trained model
https://github.com/Sutanoy/Public-Regression-Datasets

https://raw.githubusercontent.com/selva86/datasets/master/BostonHousing.cs v

URL:
https://archive.ics.uci.edu/ml/machine-learning-databases/00477/Real%20estate%20valuation%20data%20set.xlsx
(Refer any one)

Answer -

1. Dataset Loading
The dataset was loaded using

2. Data Splitting (Train/Validation/Test = 70/20/10)
The dataset was split randomly using train_test_split():
70% Training
20% Validation
10% Testing

3. Handling Class Imbalance using SMOTE
SMOTE (Synthetic Minority Oversampling Technique) was applied only to the training set to synthetically generate samples of the minority class and balance the dataset.
This ensures that the classifier doesn't become biased toward the majority class.

4. Data Preprocessing with StandardScaler
Feature scaling was performed using StandardScaler to normalize the input features.
The scaler was fit on the training set and then applied to validation and test sets to prevent data leakage.

5. Model Selection: Support Vector Machine (SVM)
The classification model used was SVM (Support Vector Classifier) from sklearn.svm.

SVM was chosen due to its effectiveness in high-dimensional spaces and support for kernel trick.

6. Hyperparameter Tuning with GridSearchCV
GridSearchCV was used to find the best combination of hyperparameters.

The hyperparameters tuned include -
C: Regularization parameter
kernel: Type of SVM kernel (linear, rbf)
gamma: Kernel coefficient for 'rbf'

7. Model Evaluation
The best model from GridSearchCV was used for evaluation on Validation and Test sets.

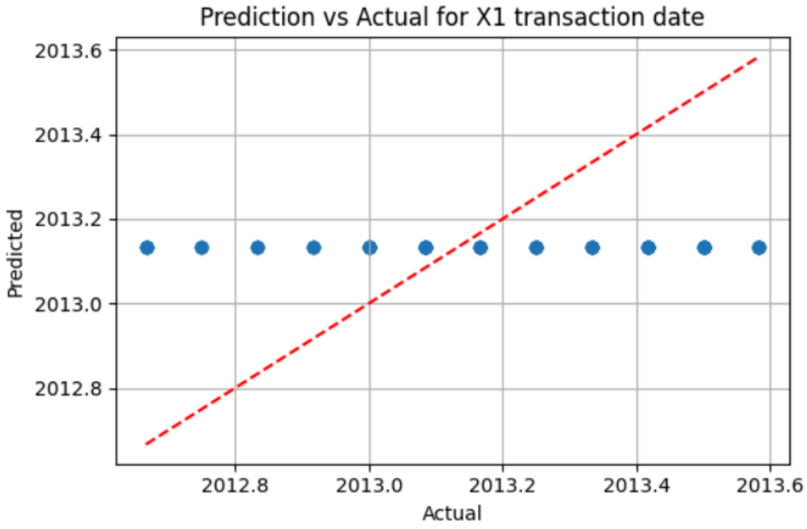
classification_report was used to display accuracy, precision, recall, and F1-score.
confusion_matrix was visualized using a Seaborn heatmap for better understanding of predictions.

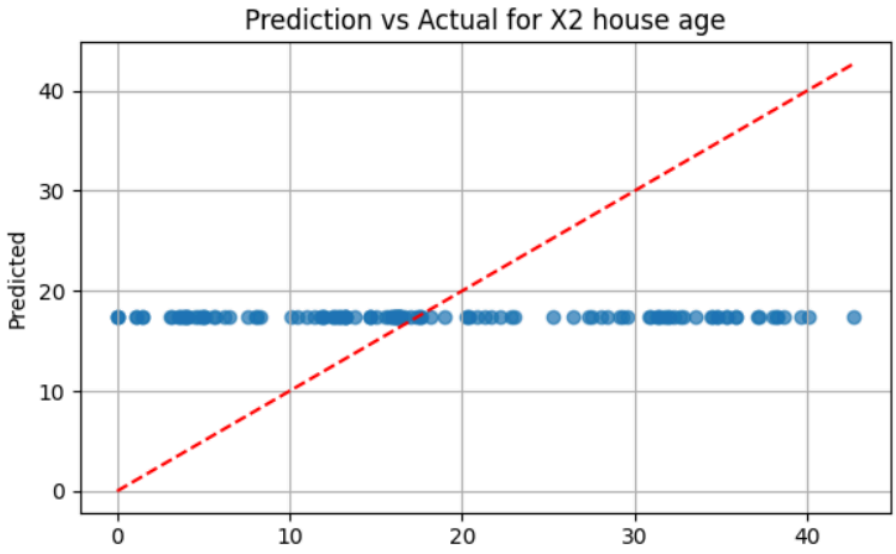
8. Object-Oriented Programming (OOP) Structure
The entire workflow was encapsulated inside a Python class ClassificationPipeline.

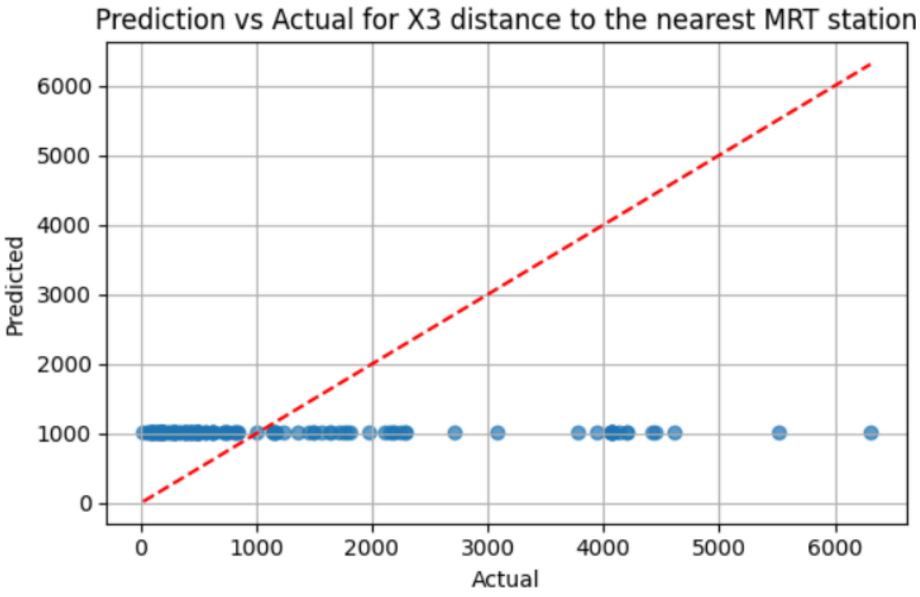
The class included the following methods -

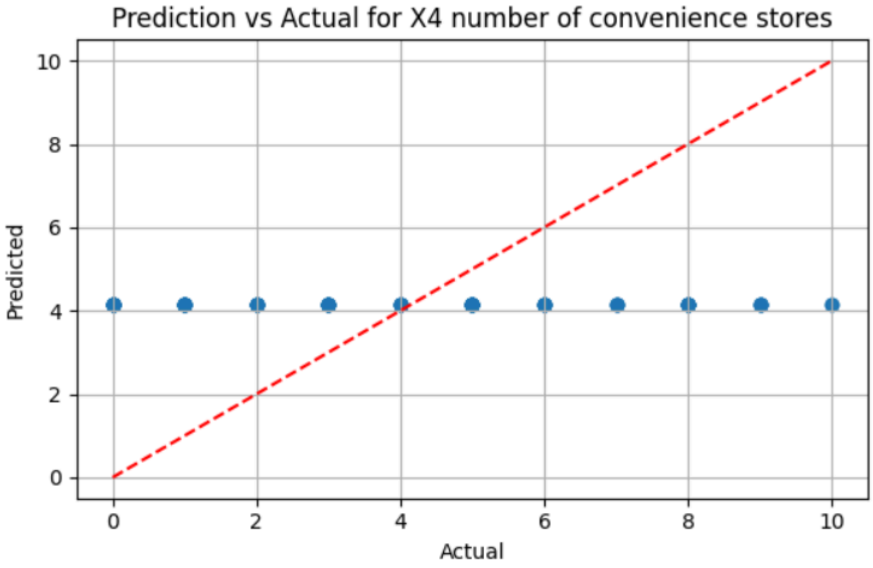
load_data() – Load and prepare features and labels
preprocess() – Split, balance, and scale data
train_model() – Hyperparameter tuning using GridSearchCV
evaluate_model() – Report and visualize performance on val/test sets

Result -




Prediction vs Actual for X2 house age
Predicted
0
10
20
30
40
0
10
20
30
40


Prediction vs Actual for X3 distance to the nearest MRT station
Predicted
0
1000
2000
3000
4000
5000
6000
0
1000
2000
3000
4000
5000
6000
Actual


Prediction vs Actual for X4 number of convenience stores
Predicted
0
2
4
6
8
10
0
2
4
6
8
10
Actual

Q.6 What are the key features of the wine quality data set? Discuss the importance of each feature in predicting the quality of wine? How did you handle missing data in the wine quality data set during the feature engineering process? Discuss the advantages and disadvantages of different imputation techniques. (Refer dataset from Kaggle).

Solution:

Following are the key Features of the Wine Quality Dataset

This table provides a data dictionary for a wine quality dataset, outlining key physicochemical features influencing wine taste and quality. Each row describes a feature—such as *fixed acidity*, *citric acid*, or *alcohol*—and explains its relevance to wine's stability, flavor, or overall quality. The target variable, *quality*, is an integer score ranging from 0 to 10, determined by sensory evaluations.

Importance of Each Feature in Predicting Wine Quality

1. **Alcohol**: Positively correlated with quality – stronger wines are generally preferred.

2. **Volatile Acidity**: Negative impact – higher levels usually reduce quality.

3. **Sulphates**: Moderate positive impact – improves preservation and taste.

4. **Citric Acid**: Enhances freshness and contributes to a crisp taste.

5. **Residual Sugar**: Small influence – only certain wine types benefit from high sugar.

6. **Total/Free SO_2**: Affects preservation but excessive values degrade taste.

7. **Fixed Acidity & pH**: Interact with other acids; control wine's sharpness and stability.

Feature importance can be quantitatively analyzed using feature importance plots (e.g., using Random Forest or SHAP values)

Handling Missing Data During Feature Engineering -

Mean/Median/Mode Imputation: Replace missing values with the column's mean, median, or mode.

KNN Imputation: Use similar (nearest neighbor) data points to fill in missing values.

Multivariate Imputation (e.g., MICE): Use regression models to estimate missing values using other features.

Model-Based Imputation (e.g., Regression): Predict missing values with a regression or ML model.

Drop Rows: Remove any rows that contain missing data.

Advantages -

Mean/Median/Mode Imputation: Simple, fast
KNN Imputation: Preserves data patterns
Multivariate Imputation (e.g., MICE): Captures inter-feature relationships
Model-Based (e.g., Regression): More accurate if correlations are strong
Drop Rows: Simple

Disadvantages -

Mean/Median/Mode Imputation: Can distort variance; doesn’t preserve relationships
KNN Imputation: Slow with large datasets; sensitive to outliers
Multivariate Imputation (e.g., MICE): Computationally intensive
Model-Based (e.g., Regression): Risk of overfitting
Drop Rows: Can lose valuable data and reduce sample size.