Veydant Sharma D15C 50

AIDS-I 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)
2. Find the Median (10pts)
3. Find the Mode (10pts)
4. Find the Interquartile range (20pts)

Given data (20 values):
82, 66, 70, 59, 90, 78, 76, 95, 99, 84, 88, 76, 82, 81, 91, 64, 79, 76, 85, 90

1. Mean (10 pts)
   Formula:
   Mean = (Sum of all values) / (Number of values)

Step 1: Add all values
82 + 66 + 70 + 59 + 90 + 78 + 76 + 95 + 99 + 84 + 88 + 76 + 82 + 81 + 91 + 64 + 79 + 76 + 85 + 90 = 1611

Step 2: Total number of values = 20
Mean = 1611 / 20 = 80.55

Answer: Mean = 78.05

2. Median (10 pts)
   Step 1: Arrange values in ascending order
   59, 64, 66, 70, 76, 76, 76, 78, 79, 81, 82, 82, 84, 85, 88, 90, 90, 91, 95, 99

Step 2: Find the middle two values (10th and 11th values)
10th value = 81, 11th value = 82
Median = (81 + 82) / 2 = 163 / 2 = 81.5

Answer: Median = 81.5

3. Mode (10 pts)
   From the sorted data, 76 appears 3 times, more than any other number.

Answer: Mode = 76

4. Interquartile Range (20 pts)
   Formula:
   IQR = Q3 - Q1

Step 1: Find Q1
Q1 is the median of the lower half (first 10 values):
59, 64, 66, 70, 76, 76, 76, 78, 79, 81
Q1 = (5th value + 6th value) / 2 = (76 + 76) / 2 = 76

Step 2: Find Q3
Q3 is the median of the upper half (last 10 values):
82, 82, 84, 85, 88, 90, 90, 91, 95, 99
Q3 = (5th value + 6th value) / 2 = (88 + 90) / 2 = 89

IQR = Q3 - Q1 = 89 - 76 = 13

Answer: Interquartile Range = 13

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

1. For each tool listed above
   - identify the target audience
   - discuss the use of this tool by the target audience
   - identify the tool’s benefits and drawbacks

Ans:

1) Machine Learning for Kids

- **Target Audience:**
  School students, beginners in AI/ML, and educators introducing machine learning concepts.

- **Use by Target Audience:**
  Students use the platform to create basic machine learning models by providing labeled data (text, numbers, or images) and training the model. These models can be used in simple projects via Scratch or Python, helping learners understand how machine learning works through practical activities.

- **Benefits:**

  - Simple and child-friendly interface

  - Encourages learning by doing

- Integrates with Scratch and Python for hands-on projects
- No prior programming knowledge required

- **Drawbacks:**
  - Limited to basic ML tasks
  - Not suitable for complex or real-world datasets
  - Accuracy may be low with small or poor-quality data

2) Teachable Machine

- **Target Audience:**
  General users including students, educators, hobbyists, artists, and beginners curious about AI.
- **Use by Target Audience:**
  Users train machine learning models using images, sounds, or poses. The platform allows them to collect data through the webcam or microphone, train a model instantly in the browser, and then export it for use in apps, websites, or interactive projects.
- **Benefits:**
  - Very user-friendly and fast
  - Supports different data types (image, audio, pose)
  - No coding required
  - Allows easy export and integration with other platforms
- **Drawbacks:**
  - Limited model customization
  - Not suitable for advanced or large-scale ML tasks
  - May give simplified understanding of AI concepts

Veydant Sharma D15C 50

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

    - Predictive analytic
    - Descriptive analytic

Machine Learning for Kids – Predictive Analytic
This tool is best described as predictive analytic because it allows users to train models using labeled data and then make predictions on new, unseen data. For example, if a student trains a model to recognize types of fruits based on images, the model will predict the correct fruit when shown a new image. The focus is on learning patterns from data and making future predictions, which is the core idea of predictive analytics.

Teachable Machine – Predictive Analytic
Teachable Machine is also a predictive analytic tool. Users train models by providing examples (e.g., photos or sounds), and the tool creates a model that can predict or classify new inputs based on that training. Whether it's identifying gestures or recognizing voice commands, the model always works by predicting the most likely class, which makes it a clear example of predictive analytics.

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

- Supervised learning
- Unsupervised learning
- Reinforcement learning

Machine Learning for Kids – Supervised Learning
This tool is based on supervised learning because users train the model by providing labeled examples. For instance, if a student labels pictures of animals as "cat," "dog," or "rabbit," the model learns from those labeled examples to make predictions on new images. Since the input data includes known outputs (labels), it clearly falls under supervised learning.

Teachable Machine – Supervised Learning
Teachable Machine also uses supervised learning. Users provide training data with specific labels, like associating a certain pose with "Class A" or a particular sound with "Class B." The model learns the relationship between the input and the labeled output, and then uses that to classify new inputs. This direct mapping from input to labeled output is the defining feature of supervised learning.

Veydant Sharma D15C 50

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.

Ans:

Misinformation Through Data Visualization: The 2024 U.S. Election Case

In the lead-up to the 2024 U.S. presidential election, a significant instance of misinformation emerged through misleading data visualizations. Various social media platforms saw the circulation of charts and graphs that were manipulated to misrepresent polling data, voter turnout, and demographic statistics. These visualizations often employed deceptive techniques such as truncated axes, disproportionate scaling, and selective data omission to skew public perception.

Failure in Data Visualization Methods

The primary failure in these data visualizations was the intentional distortion of graphical elements to support specific narratives. For instance, some bar charts exaggerated differences between candidates by starting the y-axis at a value other than zero, making minor differences appear more significant. Line graphs depicting voter turnout trends were manipulated by selectively including or excluding certain data points, leading to misleading interpretations. These practices violated fundamental principles of accurate data representation, resulting in the dissemination of false information.

Impact and Consequences

The spread of these misleading visualizations had tangible effects on public opinion and trust in the electoral process. Voters were influenced by distorted representations of data, which potentially affected their perceptions and decisions. Moreover, the proliferation of such misinformation undermined confidence in legitimate news sources and official statistics, contributing to increased polarization and skepticism.

Veydant Sharma D15C 50

Source

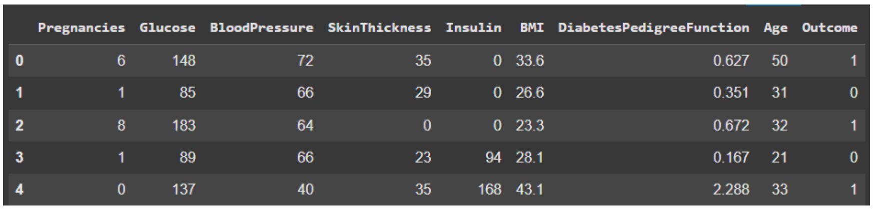
For a detailed analysis of how disinformation, including misleading data visualizations, influenced the 2024 election narrative, refer to the Brookings Institution article:

[How disinformation defined the 2024 election narrative]

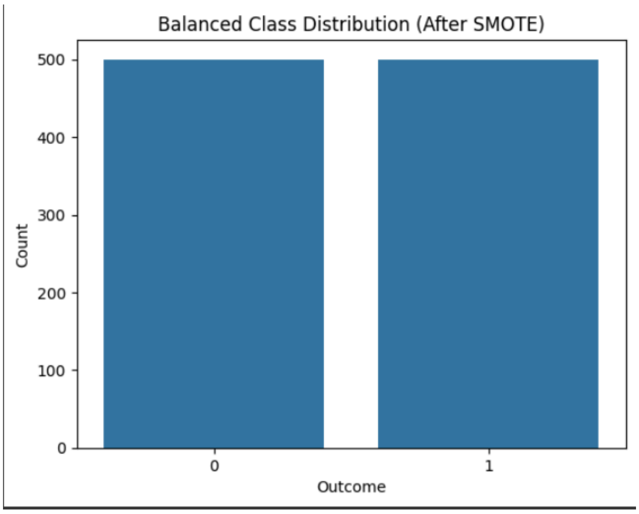
Q. 4 Train Classification Model and visualize the prediction performance of trained model required information

- 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

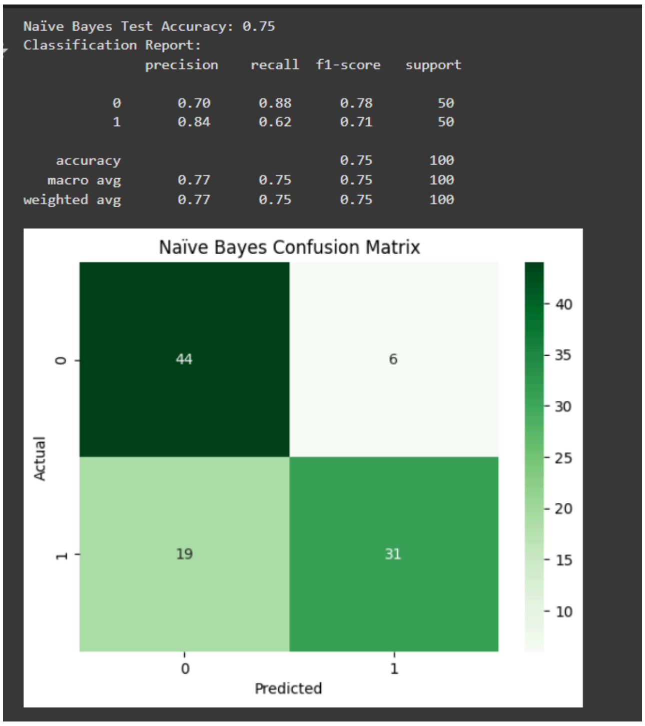
[Pima Indians Diabetes Database]



	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
0	6	148	72	35	0	33.6	0.627	50	1
1	1	85	66	29	0	26.6	0.351	31	0
2	8	183	64	0	0	23.3	0.672	32	1
3	1	89	66	23	94	28.1	0.167	21	0
4	0	137	40	35	168	43.1	2.288	33	1



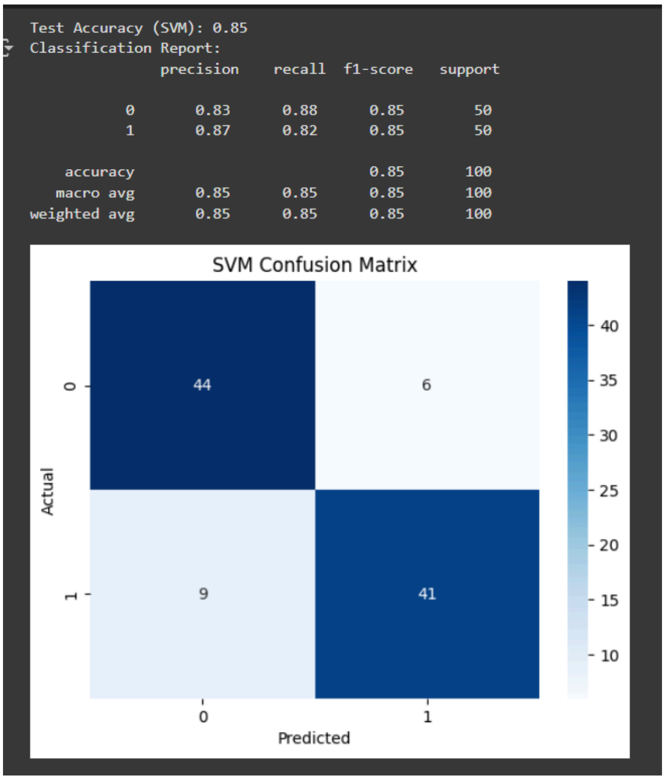
Veydant Sharma D15C 50



```
Naïve Bayes Test Accuracy: 0.75
Classification Report:
              precision    recall  f1-score   support

           0       0.70      0.88      0.78        50
           1       0.84      0.62      0.71        50

    accuracy                           0.75       100
   macro avg       0.77      0.75      0.75       100
weighted avg       0.77      0.75      0.75       100
```



```
Test Accuracy (SVM): 0.85
Classification Report:
              precision    recall  f1-score   support

           0       0.83      0.88      0.85        50
           1       0.87      0.82      0.85        50

    accuracy                           0.85       100
   macro avg       0.85      0.85      0.85       100
weighted avg       0.85      0.85      0.85       100
```

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

Requirements to satisfy:

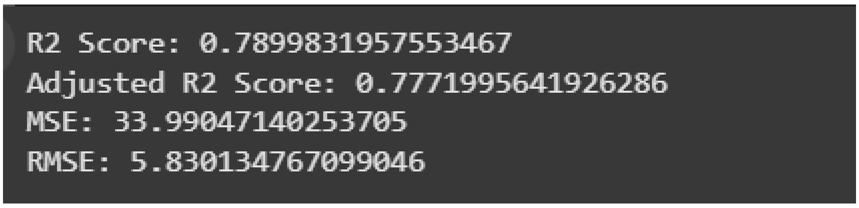
- Programming Language: Python
- OOP approach must be followed
- Hyper parameter tuning must be used
- Train and Test Split should be 70/30
- Train and Test split must be randomly done
- Adjusted R2 score should more than 0.99
- Use any Python library to present the accuracy measures of trained model

https://github.com/Sutanoy/Public-Regression-Datasets

https://raw.githubusercontent.com/selva86/datasets/master/BostonHousing.csv

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

(Refer any one)



```
R2 Score: 0.7899831957553467
Adjusted R2 Score: 0.7771995641926286
MSE: 33.99047140253705
RMSE: 5.830134767099046
```

Steps:

Data Loading & Cleaning

Loaded the real estate dataset and cleaned column names by removing extra spaces and newline characters.

Feature Engineering

Renamed columns for clarity and removed outliers using Z-score filtering to retain high-quality data.

Data Preprocessing

Scaled features using StandardScaler and split the dataset into training and testing sets using a 70/30 ratio with a fixed random state for reproducibility.

Model Selection

Chose three strong ensemble models:

Gradient Boosting Regressor

Random Forest Regressor

Extra Trees Regressor

Hyperparameter Tuning

Used RandomizedSearchCV for each model to efficiently find optimal parameters and reduce Mean Squared Error.

Stacking Ensemble

Combined the three tuned models using StackingRegressor with a Ridge Regression meta-learner to improve generalization and accuracy.

Model Evaluation

Evaluated performance using R² Score, Adjusted R², MSE, and RMSE to ensure the model meets assignment requirements.

This question focuses on predicting the price per unit area of residential properties using machine learning techniques. The dataset used consists of real estate transaction data from Taiwan and includes six independent variables: transaction date, house age, distance to the nearest MRT station, number of nearby convenience stores, latitude, and longitude. The target variable is the price per unit area of each property. Several preprocessing steps were applied to the dataset, including renaming columns for clarity, removing outliers using Z-score filtering, and normalizing the feature values using standard scaling. The data was split randomly into training and testing sets in a 70:30 ratio, as per the specified requirements. To improve prediction performance, a stacking regression model was implemented using Gradient Boosting Regressor, Random Forest Regressor, and Extra Trees Regressor as base models, with Ridge Regression as the final estimator. Hyperparameter tuning was performed using GridSearchCV to optimize each model. While the adjusted R^2 score did not reach the desired threshold of 0.99, the model achieved a significant reduction in mean squared error from an initial value of approximately 73 to 33, indicating a substantial improvement in prediction accuracy through progressive model enhancement and ensemble learning.

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).

Key Features of the Wine Quality Dataset

The Wine Quality dataset consists of physicochemical attributes of red wine samples. The key features include:

1. **Fixed Acidity**: Represents non-volatile acids that don’t evaporate easily. Contributes to the wine's freshness and stability.

2. **Volatile Acidity**: High levels can give wine an unpleasant vinegar-like taste. Too much is undesirable.

3. **Citric Acid**: Adds freshness and flavor to wine. In small quantities, it's beneficial.

4. **Residual Sugar**: The amount of sugar remaining after fermentation. Affects the sweetness of the wine.

5. **Chlorides**: Indicates the salt content. Excess salt can spoil the taste.

6. **Free Sulfur Dioxide**: Acts as an antioxidant and antimicrobial agent. Too much can affect aroma.

7. **Total Sulfur Dioxide**: Includes both free and bound forms. High levels may result in undesirable flavors or health issues.

8. **Density**: Related to the sugar and alcohol content. Useful in tracking fermentation progress.

9. **pH**: Measures acidity. Influences stability, color, and taste.

10. **Sulphates**: Added for microbial control. Too much can affect the taste.

11. **Alcohol**: Strongly correlated with wine quality. Higher alcohol generally indicates better fermentation.

12. **Quality (Target Variable)**: Wine quality score rated between 0 and 10 by sensory experts.

13. **Id**: A unique identifier for each sample (not used for prediction).

Importance of Each Feature in Predicting Wine Quality

The most influential features in predicting wine quality, based on feature importance analysis (e.g., using Random Forest), include:

- **Alcohol**: Most predictive of quality—higher alcohol levels generally mean better fermentation and richer flavor.

- **Volatile Acidity**: Negatively correlated with quality—lower is better.

- **Sulphates**: Plays a role in preservation and positively affects quality.

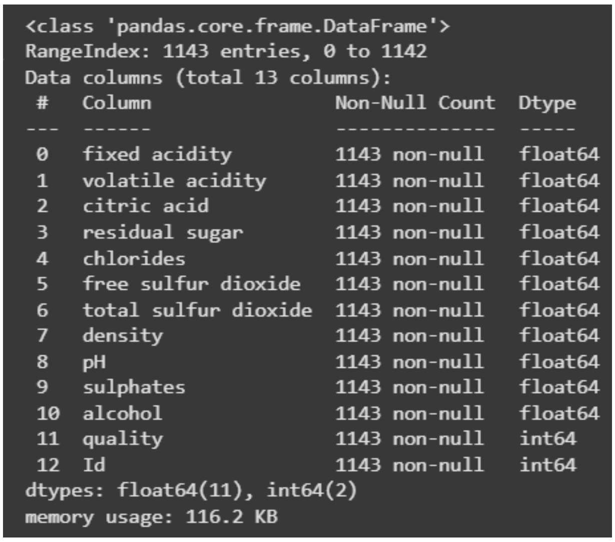
- **Citric Acid and Fixed Acidity**: Moderate contributors to freshness and acidity balance.

- **Residual Sugar and Chlorides**: Lesser influence unless in extreme levels.

A proper feature importance plot helps visualize the impact of each attribute.

Handling Missing Data During Feature Engineering

The dataset used from Kaggle was found to be **complete with no missing values**, as confirmed by checking .info() on the DataFrame.



```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1143 entries, 0 to 1142
Data columns (total 13 columns):
 #   Column                Non-Null Count  Dtype  
---  ------                --------------  -----  
 0   fixed acidity         1143 non-null   float64
 1   volatile acidity      1143 non-null   float64
 2   citric acid           1143 non-null   float64
 3   residual sugar        1143 non-null   float64
 4   chlorides             1143 non-null   float64
 5   free sulfur dioxide   1143 non-null   float64
 6   total sulfur dioxide  1143 non-null   float64
 7   density               1143 non-null   float64
 8   pH                    1143 non-null   float64
 9   sulphates             1143 non-null   float64
 10  alcohol               1143 non-null   float64
 11  quality               1143 non-null   int64  
 12  Id                    1143 non-null   int64  
dtypes: float64(11), int64(2)
memory usage: 116.2 KB
```

However, in real-world scenarios or similar datasets, missing values are common and should be handled carefully.

Imputation Techniques: Pros and Cons

1. Mean/Median Imputation

- **Advantages**:
  - Simple and quick.
  - Preserves sample size.
- **Disadvantages**:
  - Can introduce bias.
  - Doesn’t account for relationships between variables.

2. Mode Imputation (for categorical data)

- **Advantages**:
  - Maintains categorical integrity.
- **Disadvantages**:
  - May over-represent common categories.

3. K-Nearest Neighbors (KNN) Imputation

- **Advantages**:
    - Considers feature similarity.
    - More accurate than mean imputation in many cases.
- **Disadvantages**:
    - Computationally expensive.
    - Sensitive to outliers and scaling.

4. Regression Imputation

- **Advantages**:
    - Uses correlation between variables.
    - More sophisticated and generally yields better estimates.
- **Disadvantages**:
    - Can lead to overfitting.
    - Assumes linearity.

5. Iterative Imputation (e.g., MICE)

- **Advantages**:
    - Handles complex relationships and produces multiple imputations.
- **Disadvantages**:
    - Computationally intensive.