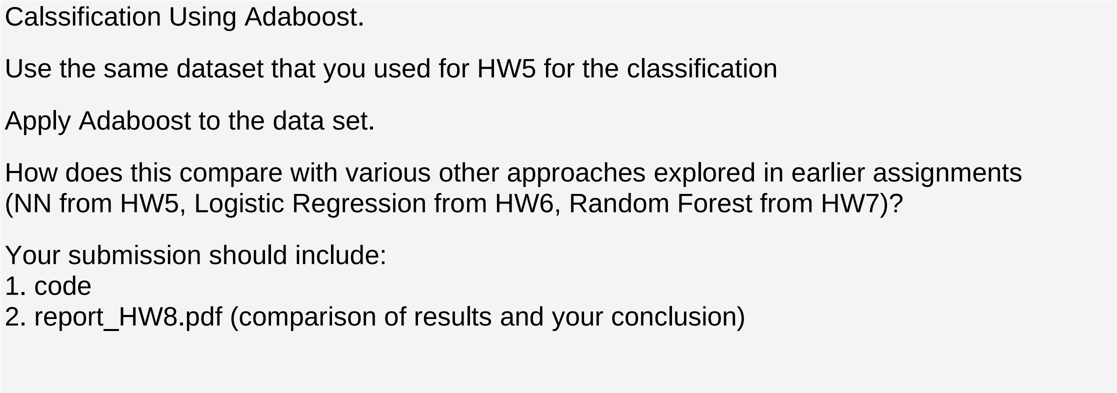
Brijesh verma
SUID: 826257344

CIS 662: Intro to Machine Learning and Algorithms

HW-8

1. Problem Statement:



Calssification Using Adaboost.

Use the same dataset that you used for HW5 for the classification

Apply Adaboost to the data set.

How does this compare with various other approaches explored in earlier assignments (NN from HW5, Logistic Regression from HW6, Random Forest from HW7)?

Your submission should include:
1. code
2. report_HW8.pdf (comparison of results and your conclusion)

2. Solution Steps:

1. Data Preprocessing:

a. **Data Loading:** Loaded the dataset ('31-40.csv') using the Pandas library.
b. **Data Cleaning:** Cleaned the 'cit_2017' column by removing commas and converting it to integers.
c. **Feature Engineering:** Created a new column 'Ratio' based on the division of 'cit_2022' by 'cit_2021'.
d. **Categorization:** Classified the 'Ratio' values into 'Low', 'Medium', or 'High' using predefined criteria.
e. **Feature Selection:** Extracted input features (X) and target labels (y) from the dataset.
f. **Label Encoding:** Encoded target labels into numerical values (0, 1, 2) representing Low, Medium, and High categories.
g. **Feature Scaling:** Applied Min-Max scaling to normalize the input features.
h. **Data Split:** Split the dataset into training and test sets (80-20 split).

2. Model Building:
Created an AdaBoost classifier with 800 estimators and a learning rate of 0.5.

3. Model Training:
Trained the AdaBoost classifier on the training set.

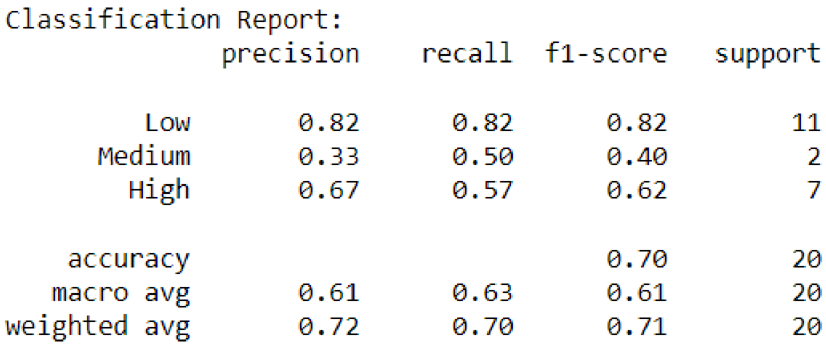
4. Model Evaluation:

a. Evaluated the model on the test set.

b. Calculated and displayed accuracy, confusion matrix, and classification report.

3. Results:

The AdaBoost classifier achieved an accuracy of 70.00% on the test set. The confusion matrix and classification report provide additional insights into the model's performance, indicating precision, recall, and F1-score for each category.



```
Classification Report:
              precision    recall  f1-score   support

         Low       0.82      0.82      0.82        11
      Medium       0.33      0.50      0.40         2
        High       0.67      0.57      0.62         7

    accuracy                           0.70        20
   macro avg       0.61      0.63      0.61        20
weighted avg       0.72      0.70      0.71        20
```

Comparison with Previous Approaches:

HW - 5 (Neural Network):

- Achieved an accuracy of 80.00% using a 6-6-3 neural network.
- More complex model architecture.
- Higher accuracy compared to AdaBoost.

HW - 6 (Logistic Regression):

- Achieved an accuracy of 60.00% using logistic regression.
- Simpler model architecture.
- Lower accuracy compared to AdaBoost.

HW - 7 (Random Forest):

- Achieved an accuracy of 65.00% using Random Forest.
- Lower accuracy compared to AdaBoost.

4. Conclusion:

- AdaBoost achieved a moderate accuracy of 70.00% in classifying individuals based on the given criteria.
- The choice between AdaBoost and other approaches depends on factors such as model complexity, interpretability, and specific goals.
- Further experimentation and fine-tuning of hyperparameters may enhance the performance of AdaBoost or other models.

- Careful consideration of dataset characteristics and problem requirements is essential for selecting the most suitable classification approach

References:

https://chat.openai.com/,
https://keras.io
https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.train_test_split.html
https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.AdaBoostClassifier.html
https://towardsdatascience.com/adaboost-in-7-simple-steps-a89dc41ec4