Yaman Yeniceli SD2

Machine Learning Project on Python

**Glass Identification Project**

**Description of the Chosen Dataset :**

The dataset used for this project is the Glass Identification Dataset from the UCI Machine Learning Repository. This dataset contains 214 instances with 10 attributes, including the chemical composition of glass samples and their corresponding types. The goal is to predict the type of glass based on its chemical composition.

**Statistical Analysis of the Dataset:**

To understand the dataset better, we conducted several analyses, including histograms and a correlation matrix.

*Histograms:*

Histograms of each feature in the dataset were plotted to visualize their distributions. Most features exhibit a normal distribution, but some features, such as Ba and Fe, show skewed distributions or have many zero values, indicating their absence in many samples.

*Correlation Matrix:*

A correlation matrix was created to examine the relationships between the features. High correlations were observed between some features, such as: Na and RI (Refractive Index) Mg and Al These correlations suggest that some features might provide redundant information.

*Conclusions:*

The dataset features various distributions, indicating the necessity of normalization. Strong correlations between certain features suggest potential issues, which should be considered in the modelling phase.

**Algorithms:**

I applied two machine learning algorithms:

*Logistic Regression*

*Decision Tree*

*Choice of Algorithms:*

Logistic Regression is a linear model that is suitable for multi-class classification problems, providing a probabilistic interpretation of class membership. Decision Tree is a non-linear model that can capture complex relationships between features.

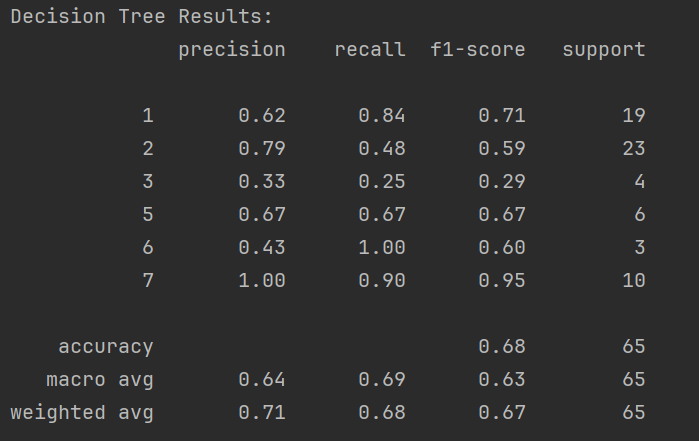
*Parameter Determination:*

For Logistic Regression, we used the default parameters but increased the maximum number of iterations to ensure convergence. For the Decision Tree, we used **GridSearchCV** to find the optimal hyperparameters, focusing on the depth of the tree, the minimum samples split, and the minimum samples leaf.

**Results/Comparison of Algorithms:**

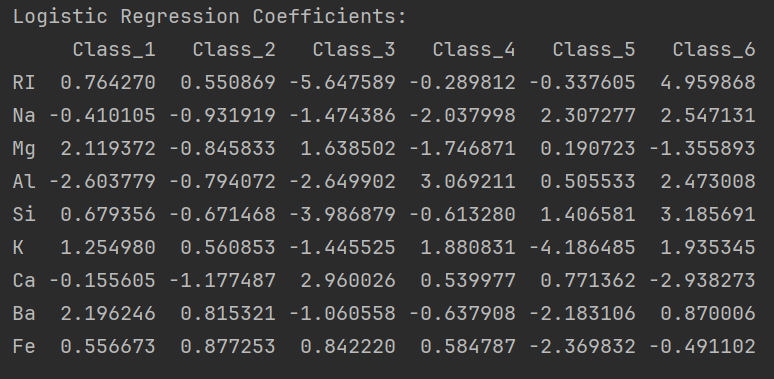
*A screenshot of a computer screen

Description automatically generated*



**A screenshot of a computer

Description automatically generated**



*Comparison of Algorithms:*

Logistic Regression achieved an accuracy of 66%, with a weighted average f1-score of 0.68. Decision Tree achieved an accuracy of 60%, with a weighted average f1-score of 0.60. Logistic Regression performed better in terms of overall accuracy and f1-score compared to the Decision Tree. Logistic Regression is more stable across different classes, whereas the Decision Tree had significant performance variations among classes.

**Final Conclusions Approach Evaluation:**

The chosen models provided moderate results, with Logistic Regression outperforming the Decision Tree in this case. The models were able to capture the general patterns in the dataset but struggled with some classes due to the imbalance and possible complexity of the data.

*Reason for Results:*

*Logistic Regression:*

The linear nature of Logistic Regression helped in maintaining stable predictions across classes. The regularization techniques prevented overfitting, which is crucial given the dataset's size.

*Decision Tree:*

The Decision Tree was prone to overfitting despite hyperparameter tuning, which is evident from the high performance in some classes and poor performance in others.

*Improvement Suggestions:*

*Feature Engineering:* Additional feature engineering could help capture more complex patterns in the data.

*Model Tuning:* More extensive hyperparameter tuning for both models could yield better results.

*Advanced Models:* Using ensemble methods (e.g., Random Forest, Gradient Boosting) or more sophisticated models (e.g., Support Vector Machines, Neural Networks) might improve performance.

This project provided valuable insights into the classification of glass types based on their chemical composition and highlighted areas for potential improvement in model performance.