Week 7 - Decision Tree and Random Forest

DS3010: Introduction to Machine Learning Lab

Timing: 02:00 PM to 04:30 PM Max Marks: 5

Instructions

1. Submit one .ipynb file containing all answers. The name should be [student_name] _[rollno]_[lab].ipynb
2. Write the questions in separate text blocks before the answers.
3. Outputs for all sub-questions should be given and the code should be executable.
4. Write justifications for your choices where needed.
5. Ensure that all plots include clear labels and legends for better interpretation.
6. Use of generative AI tools (such as ChatGPT, Gemini, etc.) is strictly prohibited. Any submission found to contain AI-generated or plagiarized content will receive a score of zero, and disciplinary action.

Your goal is to predict the **Air Quality Class (Good, Moderate, Poor)** using environmental pollutant data and compare the interpretability and performance of **Decision Tree** and **Random Forest** model.

1 Data Preprocessing (0.5 Mark)

1. Load the dataset `air_quality_index_data.csv`.
2. Check for missing values and handle them appropriately (e.g., imputation with median or mean).
3. Encode the categorical column `CityType` using label or one-hot encoding.
4. Split the dataset into train (80%) and test (20%) subsets.
5. Visualize the distribution of `AirQualityClass` using a bar or pie chart.

2 Decision Tree Classifier (1.25 Mark)

1. Train a `DecisionTreeClassifier` to predict `AirQualityClass`.
2. Print the classification report (precision, recall, f1-score) for both train and test data.
3. Display the depth and number of leaves of the trained tree.
4. Visualize the Decision Tree structure using `plot_tree()` .

3 Random Forest Classifier (1.25 Mark)

1. Train a `RandomForestClassifier` on the same dataset.
2. Display the classification report for both train and test data.
3. Compute and visualize feature importances as a horizontal bar chart.
4. Compare model performance (accuracy/F1-score) with the Decision Tree model.
5. Discuss any overfitting behavior observed.

4 Hyperparameter Tuning (1 Marks)

1. Use `GridSearchCV` for Decision Tree with a parameter grid including: `criterion`, `max_depth`, `min_samples_split`, `min_samples_leaf`, and `max_features`.
2. Use `RandomizedSearchCV` for Random Forest with parameters: `n_estimators`, `max_depth`, `min_samples_split`, `min_samples_leaf`, and `bootstrap`.
3. Print the best parameters and retrain both models using them.
4. Evaluate tuned models on the test data and compare metrics with the base models.

5 Model Interpretation and Discussion (1 Marks)

1. Explain the main advantage of using a Random Forest over a single Decision Tree. In your answer, discuss how ensemble learning helps reduce overfitting and when a single Decision Tree might still be preferable.