Assignment: Ensemble Learning and Tree-Based Algorithms

Objective: To understand and implement various tree-based algorithms and ensemble methods, evaluate their performance, and interpret their results.

Part 1: Decision Trees

1. **Dataset Selection**: Choose a dataset suitable for classification. This could be from UCI Machine Learning Repository, Kaggle, or any other source.
2. **Data Exploration**: Conduct a brief exploratory data analysis (EDA) to understand the dataset's features and target variable.
3. **Decision Tree Implementation**: Train a decision tree classifier. Visualize the tree and interpret the results.
4. **Evaluation**: Use appropriate metrics (e.g., accuracy, precision, recall, F1-score) to evaluate the model's performance on a test set.

Part 2: Bagging

1. **Bootstrap Sampling**: From the dataset used in Part 1, create 10 bootstrap samples.
2. **Model Training**: Train a decision tree on each bootstrap sample.
3. **Aggregation**: Aggregate predictions from all trees to make the final prediction.
4. **Evaluation**: Compare the bagging ensemble's performance against the single decision tree from Part 1.

Part 3: Random Forest

1. **Model Implementation**: Train a Random Forest classifier on the dataset.
2. **Feature Importance**: List the top 5 features based on their importance.
3. **Evaluation**: Compare the Random Forest's performance against the models from previous parts.

Part 4: Gradient Boosting

1. **XGBoost Implementation**: Train an XGBoost classifier. Adjust hyperparameters like **learning_rate**, **max_depth**, and **n_estimators**.
2. **Feature Importance with XGBoost**: Identify and visualize the importance of features in the trained model.
3. **Evaluation**: Compare the XGBoost model's performance against the other models.

Part 5: Comparative Analysis

1. **Performance Metrics**: Tabulate the performance metrics (accuracy, precision, recall, F1-score) of all models side by side.

   Example:

	accuracy	precision	recall	F1-score
Decision Trees	0.85	0.85	0.85	0.85
Bagging	0.85	0.85	0.85	0.85
Random Forest	0.85	0.85	0.85	0.85
Gradient Boosting	0.85	0.85	0.85	0.85

2. **Discussion**: Write a brief report discussing the performance of each model, situations where each model might be preferred, and any challenges faced during implementation.

Submission Guidelines:

- Submit the code file (Jupyter notebooks) with detailed comments and visualizations like tree diagrams, feature importance plots, and performance metric graphs and metric TAB.