Project Title: Comparative Analysis of Supervised Learning Algorithms

Project Overview:

This project aims to provide students with hands-on experience in applying various supervised learning algorithms to a selected dataset, performing feature selection, tuning hyperparameters, and evaluating model performance using appropriate metrics. Students will compare at least three different supervised learning algorithms to understand their strengths, weaknesses, and applicability to specific types of data and problems.

Objectives:

- Select the dataset of your choice for supervised problems.
- Understand and apply different supervised learning algorithms.
- Perform feature selection to improve model performance.
- Tune hyperparameters to optimize models.
- Evaluate and compare model performance using various metrics.

Deliverables:

Final Report (50%): Comprehensive documentation of the project, including final feature selection methods, hyperparameter tuning techniques, model evaluation, and a comparative analysis of the algorithms.
Presentation (40%): A 20-minute presentation summarizing the project findings, challenges faced, and key takeaways.
Code Repository (10%): A Git repository containing all code used in the project, including data preprocessing, model training, and evaluation scripts.

Submission Format:

Submit a Report, Powerpoint Presentation, and Github code repository link via Canvas

Project Phases:

Phase 1: Dataset Selection and Preprocessing

- Select a dataset suitable for supervised learning tasks.
- Perform data cleaning, normalization, and splitting into training and testing sets.

Phase 2: Exploratory Data Analysis (EDA) and Feature Selection

- Conduct EDA to understand the dataset's characteristics.
- Apply feature selection techniques to identify relevant features for modeling.

Phase 3: Model Implementation and Baseline Evaluation

- Implement at least three supervised learning algorithms (e.g., SVM, Decision Trees, Neural Networks).
- Evaluate baseline models using initial features without hyperparameter tuning.

Phase 4: Hyperparameter Tuning

- Use techniques like grid search or random search to find optimal hyperparameters for each model.
- Re-evaluate models with tuned hyperparameters to assess performance improvements.

Phase 5: Model Evaluation and Comparative Analysis

- Evaluate models using metrics such as accuracy, precision, recall, F1 score, and ROC-AUC.
- Compare the performance, computational efficiency, and applicability of each algorithm.

Phase 6: Conclusion and Recommendations

- Summarize findings and insights gained from the comparative analysis.
- Provide recommendations on the most suitable algorithms for the dataset and problem type.

Evaluation Metrics:

- Clarity and thoroughness of the project proposal and reports.
- Effectiveness of feature selection and hyperparameter tuning strategies.
- Accuracy and depth of model evaluation and comparative analysis.
- Quality and organization of the presentation and code.

Recommended Tools and Libraries:

- Programming Language: Python
- Libraries: scikit-learn, pandas, numpy, matplotlib, seaborn, polars
- Environment: Jupyter Notebook or Google Colab

Submission Date:

April 10 (by Midnight)