Campus Placement Analysis and Model Evaluation Report

1. Dataset Description and Preprocessing

Dataset Overview:

- The Campus Placement dataset is designed to analyze the factors influencing the campus recruitment outcomes of students. It aims to predict whether a student gets placed based on various academic, demographic, and employability factors.
- The dataset contains 15 features, each representing a different aspect of the student's profile:
    - Academic Scores:
        - ssc_p: Percentage obtained in Secondary Education (10th grade).
        - hsc_p: Percentage obtained in Higher Secondary Education (12th grade).
        - degree_p: Percentage obtained in the undergraduate degree.
        - mba_p: Percentage obtained in the MBA degree.
        - etest_p: Employability test score percentage, assessing student's aptitude.
    - Demographic Information:
        - gender: Gender of the student (Male/Female).
        - ssc_b: Board of Education for Secondary Education (Central/Other).
        - hsc_b: Board of Education for Higher Secondary Education (Central/Other).
        - hsc_s: Specialization in Higher Secondary Education (Science/Commerce/Arts).
        - degree_t: Type of undergraduate degree (Science/Commerce/Arts).
        - specialisation: MBA specialization (Marketing & Finance or Marketing & HR).
        - workex: Indicates if the student has prior work experience (Yes/No).
    - Target Variable:
        - status: Indicates whether the student was placed ('Placed') or not ('Not Placed').
        - salary: The offered salary for students who were placed. Not used as a feature since we are predicting placement status.

2. Preprocessing Steps:

- Dropped Irrelevant Columns:
    - Removed sl_no as it is merely an identifier with no predictive value.
    - Dropped salary because it is not relevant when predicting placement status and is only available for placed students.
- Encoding the Target Variable:
    - Converted the status column to binary values (1 for 'Placed', 0 for 'Not Placed') to make it suitable for binary classification.
- Encoding Categorical Variables:
    - Applied One-Hot Encoding to binary features (gender, ssc_b, hsc_b) and multi-class categorical features (degree_t, specialisation, workex) to convert them into numerical format.
- Scaling Numerical Features:
    - Used StandardScaler for features like ssc_p, hsc_p, degree_p, etest_p, and mba_p to standardize them. This ensures that all numerical features are on a similar scale, which is essential for models sensitive to feature magnitude.
- Handling Class Imbalance:
    - The dataset exhibited class imbalance, with a higher number of 'Placed' students compared to 'Not Placed'. To balance this, SMOTE (Synthetic Minority Over-sampling Technique) was applied to the training set, creating synthetic samples for the minority class (Not Placed). This ensures that the models do not become biased towards the majority class.

---

3. Model Selection and Explanation

Models Chosen:

- Logistic Regression:
    - Why Chosen: Logistic Regression is a fundamental binary classification model that provides interpretability through coefficients. It is suitable when the relationship between features and the target is linear and when transparency in decision-making is important.

- Decision Tree:
    - Why Chosen: Decision Trees can model complex, non-linear relationships and do not require feature scaling. They are easy to visualize and can highlight important features through splits, making them suitable for understanding decision paths.
- Random Forest:
    - Why Chosen: This ensemble method reduces the risk of overfitting by averaging predictions across multiple trees, offering robustness and generalization to unseen data. It is particularly useful when feature interactions are complex.
- Voting Classifier:
    - Why Chosen: Combines the strengths of multiple models (Logistic Regression, Decision Tree, and Random Forest) by using a majority voting approach. This helps improve generalization and balance between precision and recall, making it suitable for datasets where both false positives and false negatives need to be minimized.

Training Methodology:

- Each model was trained on the balanced dataset obtained using SMOTE.
- Hyperparameter tuning was performed using GridSearchCV to identify the best parameters for each model, ensuring optimal performance.
- The performance of each model was evaluated on a separate test set to assess generalization capabilities.

---

4. Model Performance Evaluation

Evaluation Metrics:

- Accuracy: Measures the overall percentage of correct predictions.
- Precision: Focuses on the proportion of true positives among all positive predictions, indicating the quality of positive predictions.
- Recall: Measures the ability to correctly identify actual positive cases, i.e., the proportion of placed students correctly identified.
- F1 Score: A balance between precision and recall, providing a single measure of a model's performance.

Performance Summary:

- Logistic Regression:
    - Achieved an accuracy of 0.8372, with high precision (0.9) and a strong F1 score (0.8852).
    - The model's interpretability makes it suitable for decision-making in educational institutions where understanding the impact of each factor is important.
- Decision Tree:
    - Showed the highest recall (0.9032), making it effective in identifying most of the placed students.
    - However, it had a slightly lower accuracy (0.814) and F1 score compared to Logistic Regression, indicating a trade-off between recall and precision.
- Random Forest:
    - Despite being an ensemble method, it struggled with precision (0.8438), which led to lower overall accuracy (0.7907).
    - The model may have overfitted due to its complexity, making it less generalizable to new data.
- Voting Classifier:
    - Demonstrated a balanced performance with an accuracy of 0.8437, a precision of 0.875, and a recall of 0.9032.
    - Its F1 score (0.8889) indicates that it effectively balances precision and recall, making it a competitive option alongside Logistic Regression.

---

5. Before vs. After Hyperparameter Tuning Comparison

Model Comparisons:

- Logistic Regression:
    - Performance metrics remained largely consistent before and after tuning, suggesting that the default settings were already optimal for this problem.
- Decision Tree:

    - Hyperparameter tuning led to improved precision and recall, indicating better model performance. The tuned model was better at balancing true positives and minimizing false positives.
- Random Forest:
    - The tuned model experienced a slight decrease in accuracy and precision, likely due to overfitting to the training data. This suggests that while the model performed well during training, it struggled with generalization when applied to the test data.

---

6. Conclusion

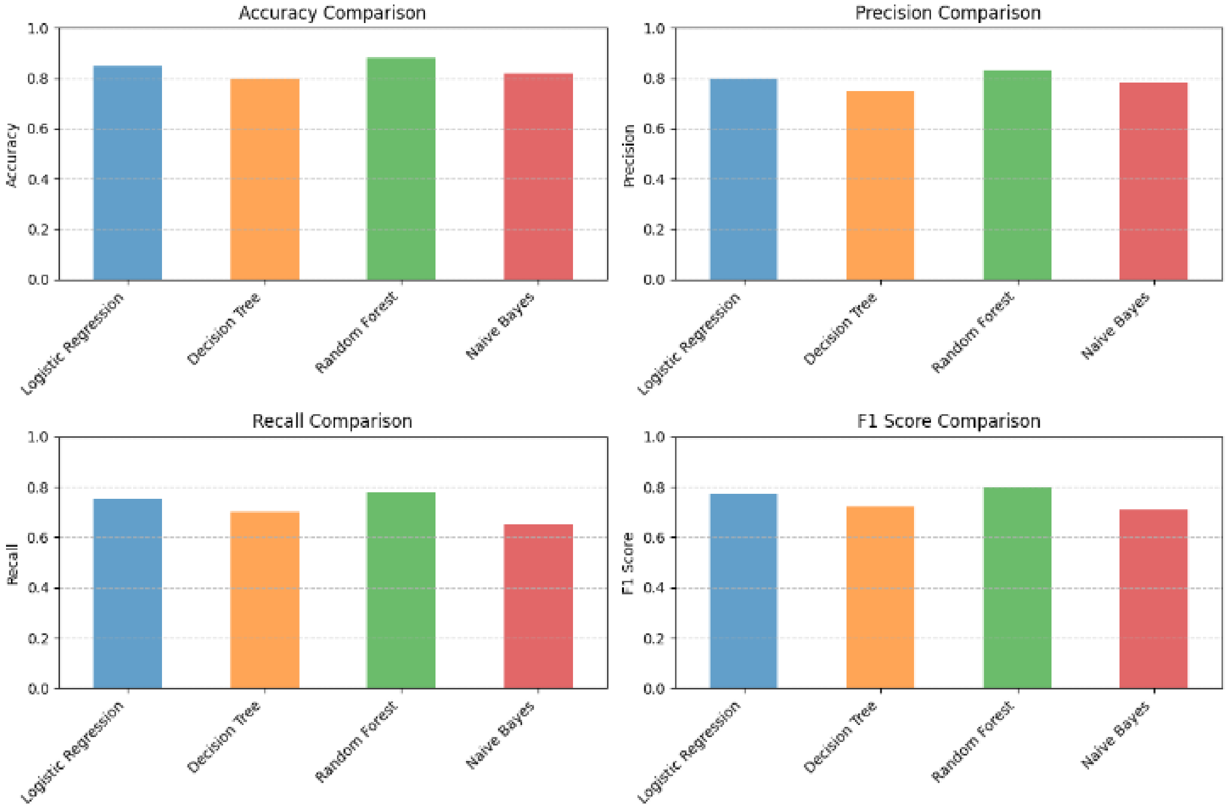
- Best Model:
    - Logistic Regression is the top performer in terms of accuracy and F1 score, making it suitable for situations where interpretability and precision are critical.
    - Voting Classifier provides a balanced solution with high recall, making it ideal for scenarios where identifying all potential candidates is more important, such as minimizing the risk of missing qualified students.
- Recommendations:
    - Educational institutions could leverage Logistic Regression for interpreting factors that influence student placement outcomes.
    - If the goal is to ensure that all potential students are identified (minimizing false negatives), the Voting Classifier is recommended due to its balanced recall.

---

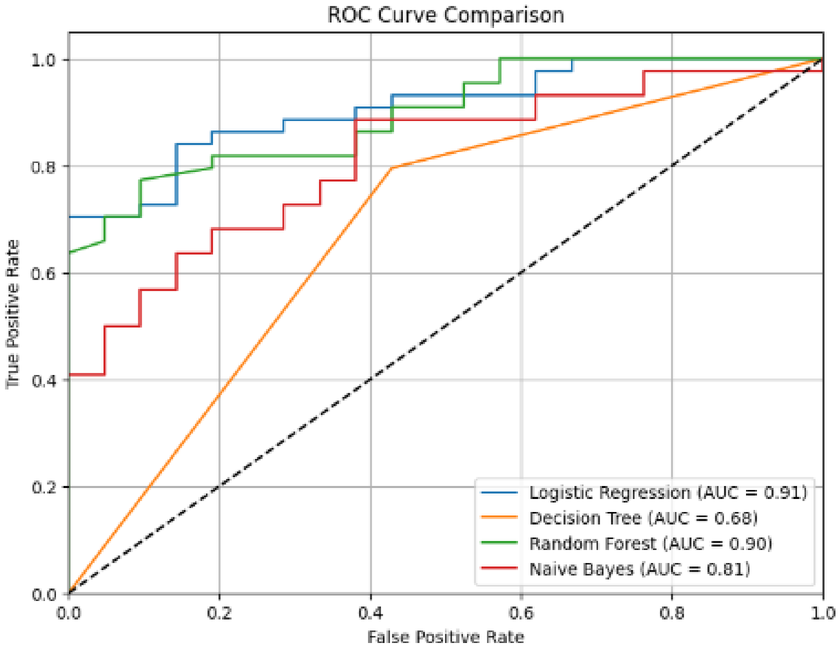
7. Appendix: Code and Visualization

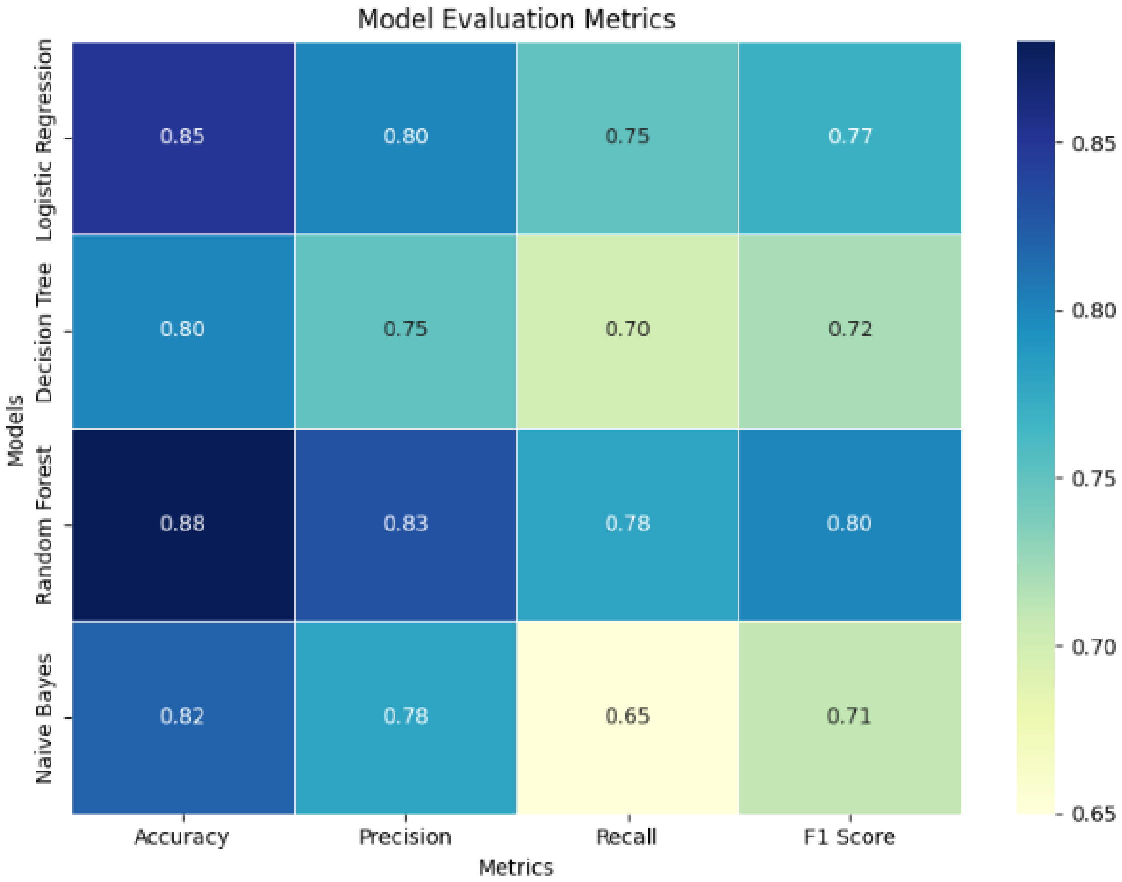
- Code Snippets:
    - Data Preprocessing: Includes code for dropping irrelevant columns, encoding variables, scaling, and handling class imbalance using SMOTE.
    - Model Training: Code snippets for training models, performing hyperparameter tuning with GridSearchCV, and evaluating models.
- Visualizations:

- Bar Plots: Graphs comparing model performance metrics (accuracy, precision, recall, F1 score) for each model before and after tuning.



- Confusion Matrices: Heatmaps showing true positive, true negative, false positive, and false negative counts for each model.



Model Evaluation Metrics
Models
Logistic Regression
Decision Tree
Random Forest
Naive Bayes
Metrics
Accuracy
Precision
Recall
F1 Score
0.85
0.80
0.75
0.77
0.80
0.75
0.70
0.72
0.88
0.83
0.78
0.80
0.82
0.78
0.65
0.71
0.85
0.80
0.75
0.70
0.65