Assignment – 2 (REPORT)

Campus Placement Prediction Using Machine Learning

Report by : Athulya Jayan

Student ID: C0936177

1. Dataset Overview

The dataset used for this project was sourced from Kaggle (https://www.kaggle.com/c/ml-with-python-course-project/data?select=train.csv) as part of a machine learning coursework. It contains academic and employment records of students, with the target variable being "status" (Placed/Not Placed).

The original dataset had 215 entries and 15 columns. After preprocessing:

"sl_no" (a simple identifier) was removed as it had no predictive value.

"salary" was excluded due to missing values and potential target leakage (since salary is only relevant for placed students).

The final dataset included 13 columns (mix of categorical and numerical data) with no missing values.

2. Data Preprocessing

Key preprocessing steps:

1. Removed irrelevant columns: "sl_no" and "salary".

2. Confirmed no missing values in the remaining data.

3. Encoded categorical variables using label encoding.

4. Scaled numerical variables to standardize their ranges.

5. Split the dataset into 70% training and 30% testing subsets.

3. Exploratory Data Analysis (EDA)

Key insights from EDA:

1. The dataset was moderately imbalanced: 148 placed vs 67 not placed.
2. Visualized distributions of key numerical variables:
3. Secondary school percentage (ssc_p)
4. Higher secondary percentage (hsc_p)
5. Degree percentage (degree_p)

6. Employability test scores (etest_p)
7. Used count plots and bar plots to analyze categorical variables and their relationship with placement status.

Additional improvements included more detailed distribution plots to better understand academic performance trends among placed/unplaced students.

4. Model Selection and Tuning

Four machine learning models were selected for their complementary strengths in structured tabular data:

1. Logistic Regression

   1. Simple, interpretable, and effective for binary classification.
   2. Assumes a linear relationship between features and log-odds of placement.

2. Random Forest Classifier

   1. Handles numerical and categorical data well.
   2. Captures nonlinear relationships and feature interactions.
   3. Robust to outliers and noise.

3. Support Vector Machine (SVM)

   1. Maximizes decision margin between classes.
   2. Effective for high-dimensional data and complex boundaries (using kernel tricks).

4. Voting Classifier

   1. Combined predictions from Logistic Regression, Random Forest, and SVM.
   2. Used hard voting (majority vote) for final predictions.

Hyperparameter tuning was done using Grid Search with cross-validation to optimize performance. Evaluation metrics included
accuracy, precision, recall, and F1-score.

5. Model Evaluation

Performance summary on the test dataset:

Model	Accuracy	Precision	Recall	F1-Score
Logistic Regression	75.38%	79.59%	86.67%	82.98%
Random Forest	93.85%	93.62%	97.78%	95.65%
SVM	69.23%	69.23%	100.0%	81.80%
Voting Classifier	84.62%	83.02%	97.78%	89.80%

Model Performance Insights:

Among the models that I have tested, Random Forest emerged as the most powerful and efficient. When properly fine-tuned, it delivered outstanding results with the highest accuracy rate of 93.85% and an F1 Score of 95.65%, indicating that it excels in striking a balance between precision and recall as well. This means that the model has not only captures the majority of students who actually get placed but is also quite accurate in those predictions as well.

The Voting Classifier also showcased then strong overall performance. Although its accuracy (84.62%) was slightly lower than that of Random Forest, it demonstrated excellent recall (97.78%), nearly matching Random Forest in identifying placed students. It maintained a solid precision score of (83.02%) and a high F1 Score of 89.80%, reflecting a well-rounded and dependable model. The Voting Classifier benefits from combining the strengths of multiple algorithms, resulting in a more balanced predictive outcome.

Support Vector Machine (SVM) behaved quite differently. It achieved a perfect recall score (100%), meaning it successfully identified all students who were placed. However, its precision was low (69.23%), suggesting a tendency to overpredict placements. As a result, while it didn’t miss any placed students, it frequently mislabeled unplaced students as placed. This led to a relatively low accuracy (69.23%) and F1 Score (81.80%), making it less reliable overall.

Logistic Regression, despite being a simpler model, provided a reasonable benchmark. It achieved an accuracy of 75.38%, with an F1 Score of 82.98%, which is decent for a baseline model. Its performance in precision and recall was satisfactory, though it was clearly surpassed by both the Voting Classifier and Random Forest in every key metric.

Conclusion

Random Forest is the most effective model, leading in accuracy, precision, recall, and F1 Score.

Voting Classifier is a strong alternative, offering robust recall and a good overall balance.

SVM while perfect in recall, lacks precision and may lead to over-classification.

Logistic Regression performs adequately but is better suited as a comparison model rather than the final choice.

Future Work:

1. Address class imbalance using techniques like SMOTE or weighted classes.
2. Experiment with additional models (e.g., XGBoost, Neural Networks).
3. Collect more data to improve generalization.