INNOMATICS®
RESEARCH LABS

INNOVATION. AUTOMATION. ANALYTICS

PROJECT ON

Predicting Student Placements Using Machine Learning

Done By: Anuradha K

About me

- **Background ? (B-tech or M-tech)**

B.Arch Graduate

- **Why you want to learn Data Science**

I enjoy learning new things, and data science is a field that's always growing and changing. I want to add new skills like data analysis, programming, and machine learning to my skill set.

In architecture, we often make choices based on design principles. I'm excited to learn how to make decisions based on data, which can provide more precise and impactful results.

- **Any work experience**

No, work experience

- **Share your linkedin and github profile urls**

Anuradha Kilaparthi | LinkedIn
anuradhak0801 (Anuradha K) (github.com)

PROBLEM STATEMENT AND USE CASE DOMAIN

Problem Statement:
Educational institutions and career placement cells struggle to efficiently predict which students are more likely to secure a job offer. Without data-driven insights, career counselors and recruiters rely on manual assessments, which may not always be accurate. This results in missed opportunities for students and inefficient hiring for companies.

Use Case Domain:
This project falls under the **Education Technology (EdTech)** and **HR Tech (Recruitment Analytics)** domains, benefiting **Universities & Colleges, Placement & Training Departments, Corporate Recruiters & HR Departments, EdTech & AI-Based Career Platforms.**

Data
The dataset contains the following columns:

1. StudentID	7. SoftSkillsRating
2. CGPA	8. ExtracurricularActivities
3. Internships	9. Placement Training
4. Projects	10. SSC_Marks
5. Workshops/Certifications	11. HSC_Marks
6. AptitudeTestScore	12. PlacementStatus

OBJECTIVE AND MODEL BUILDING PROCESS

Objective:

To develop a machine learning model that predicts whether a student will be placed based on academic performance, internships, projects, certifications, soft skills, and other relevant factors. This will help educational institutions improve placement rates, assist recruiters in shortlisting candidates efficiently, and provide students with personalized career guidance.

Data Preprocessing:

1. **Handled Missing Values:** Checked for null values and imputed or removed them.
2. **Detected & Treated Outliers:** Used IQR method & box plots to identify and handle outliers.
3. **Removed Unwanted Columns:** Dropped columns that do not contribute to prediction.
4. **Encoded Categorical Features:** Applied Label Encoding to convert categorical variables into numerical form.
5. **Feature Scaling:** Applied **StandardScaler** to normalize numeric values.
6. **Polynomial Features:** Applied Polynomial Features (Degree = 2) to capture non-linear relationships.

OBJECTIVE AND MODEL BUILDING PROCESS

EDA:

Visualizing Target Distribution:

- Used sns.countplot() and df["PlacementStatus"].value_counts() to check **class imbalance**.
- Plotted **bar graphs** to analyze placed vs. non-placed students.

Analyzing Feature Distributions:

- Used sns.histplot() for **CGPA, Aptitude Test Scores, SSC & HSC Marks**.
- Identified **skewness and outliers**.

Detecting Outliers:

- Used **box plots** (sns.boxplot()) for numerical features like **CGPA, SSC, and HSC Marks**.
- Applied **IQR (Interquartile Range) method** to find outliers.

Feature Relationships & Correlation:

- Used sns.heatmap(df.corr()) to visualize **feature correlations**.
- Helped identify **unnecessary features** and their impact on placement status.

Comparing Placement vs. Key Factors:

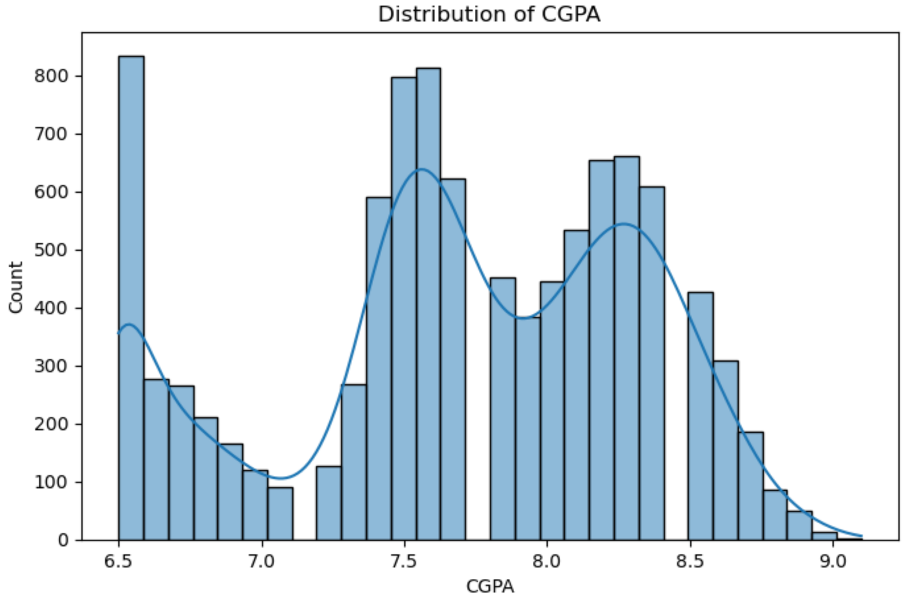
- **Internships & Projects vs. Placement:** Assessed real-world experience impact.
- **Placement Training vs. Placement:** Evaluated importance of training.

OBJECTIVE AND MODEL BUILDING PROCESS

Model Building & Hyperparameter Tuning:
1. Split Data into Training & Testing Sets (80% training, 20% testing).
2. **Optimized Model Performance:** Used GridSearchCV for hyperparameter tuning.

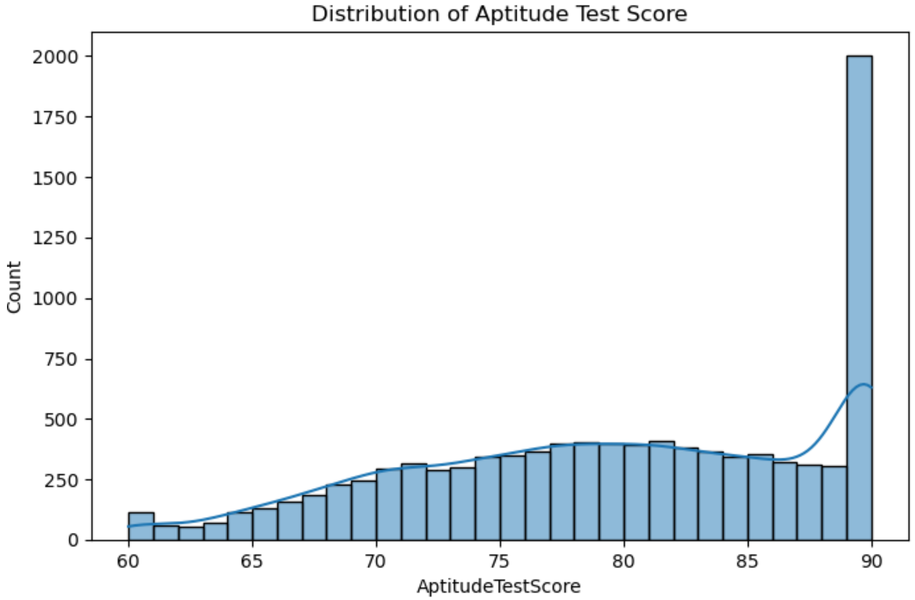
Model Evaluation & Performance Metrics:
1. **Accuracy Score:** Measures overall correctness.
2. **ROC-AUC Score:** Evaluates classification performance for imbalanced classes.
3. **Confusion Matrix:** Analyses false positives and false negatives.
4. **Precision, Recall & F1 Score:** Assesses how well the model predicts placed vs. non-placed students.

EDA



Key Observations

- The histogram with KDE shows the overall distribution of students' CGPA.
- A significant number of students have CGPA between 7 and 8.5, which could indicate that this range is common among students.

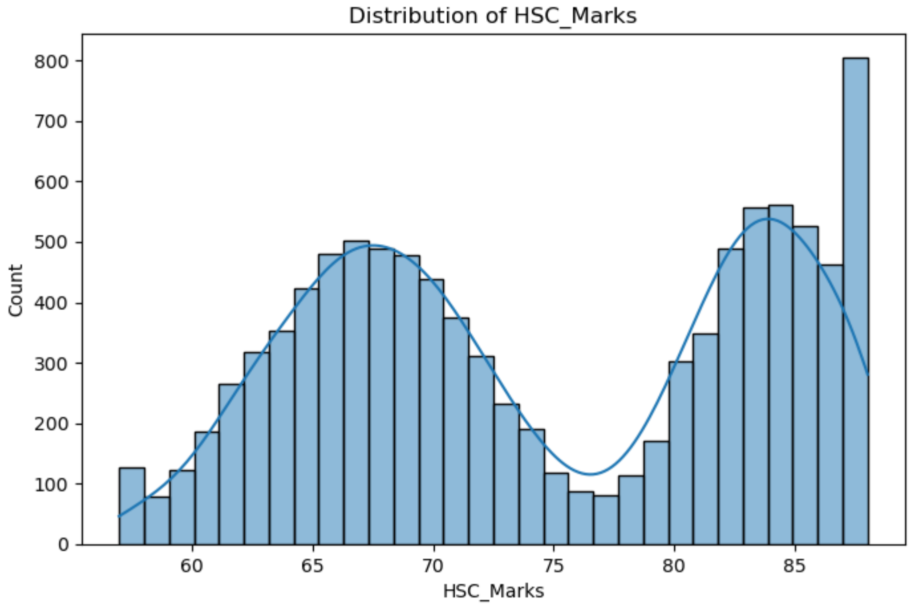


Key Observations

- The distribution appears right-skewed, with a peak at around 90, indicating many students scored 90.
- There is a gradual increase in frequency between 60-85, followed by a sharp spike at 90.

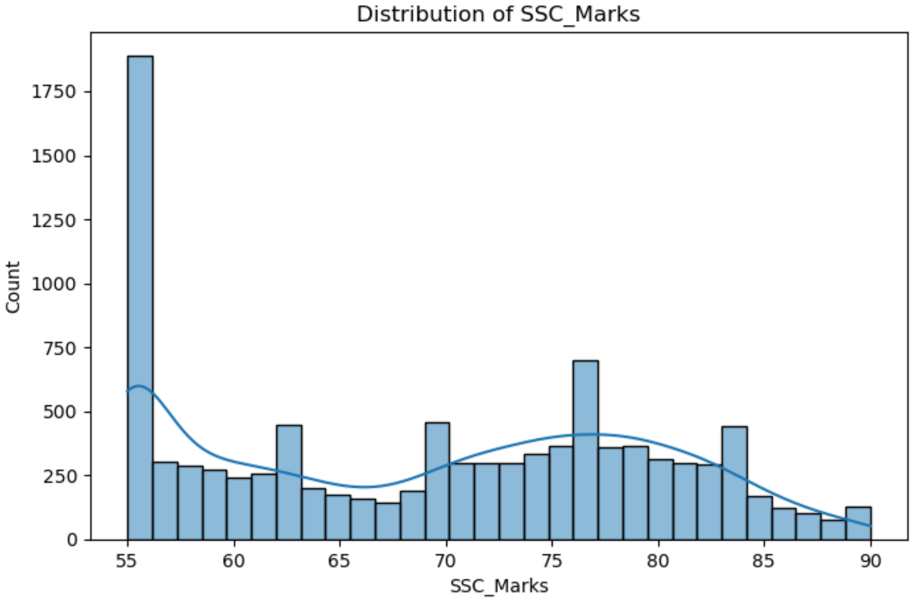
INNOMATICS
RESEARCH LABS

EDA



Key Observations

- The distribution is bimodal, with peaks around 65-70 and another around 80-85.
- This suggests that there might be two distinct groups of students, one group scoring in the mid-60s and another in the 80s.

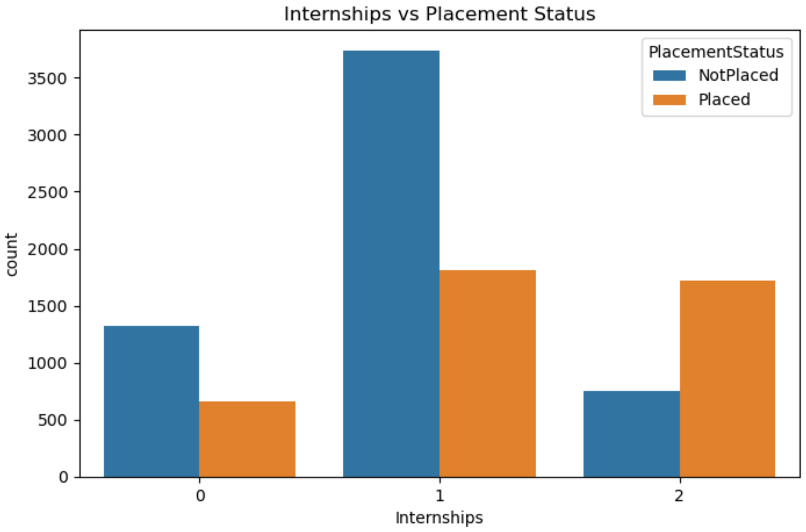


Key Observations

- This distribution is somewhat bimodal, with a peak near 55-60 and another around 75.
- The high number of students scoring near 55 suggests a higer number of lower-performing students.

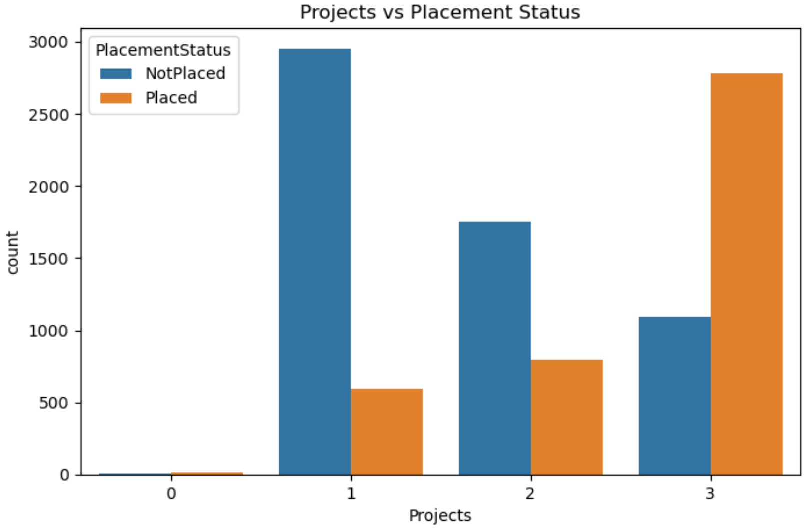
INNOMATICS RESEARCH LABS

EDA



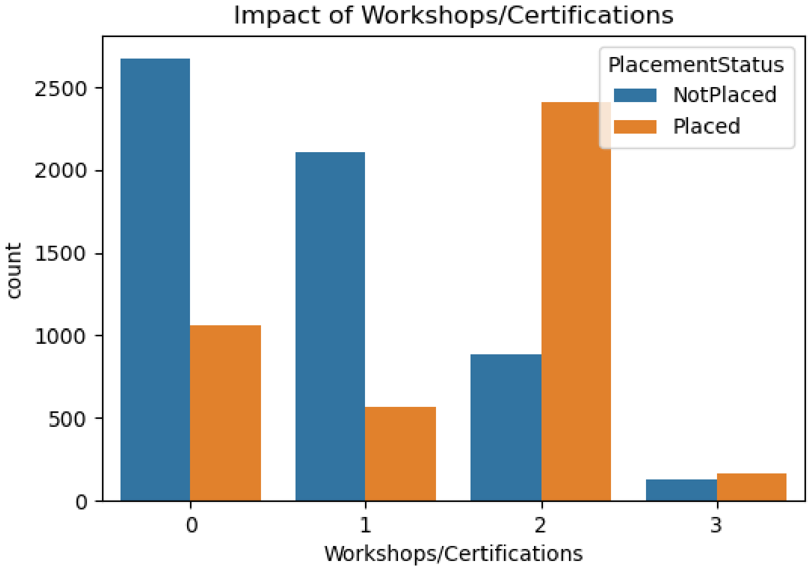
Key Observations

- The plot show that having at least one internship increases the probability of being placed.
- Students with no internships have a higher chance of not being placed.



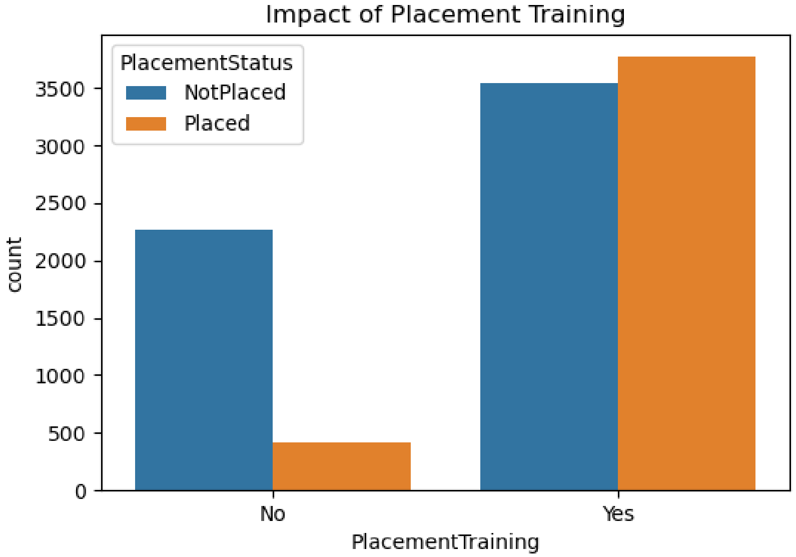
Key Observations

- There is a clear relation between the number of projects and placement status.
- Students who have done three projects have the highest placement rate.



Key Observations

- Students with 2 or more workshops/certifications have a higher chance of being placed.
- The majority of students with 0 or 1 certification remain unplaced.

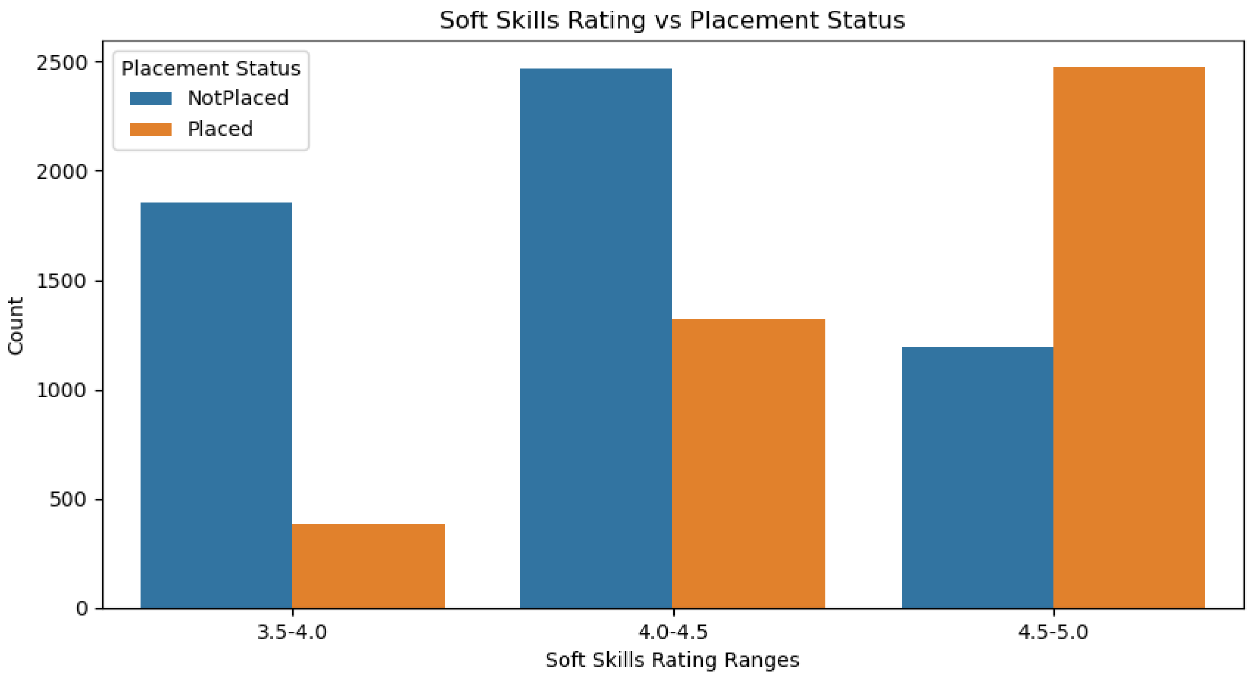


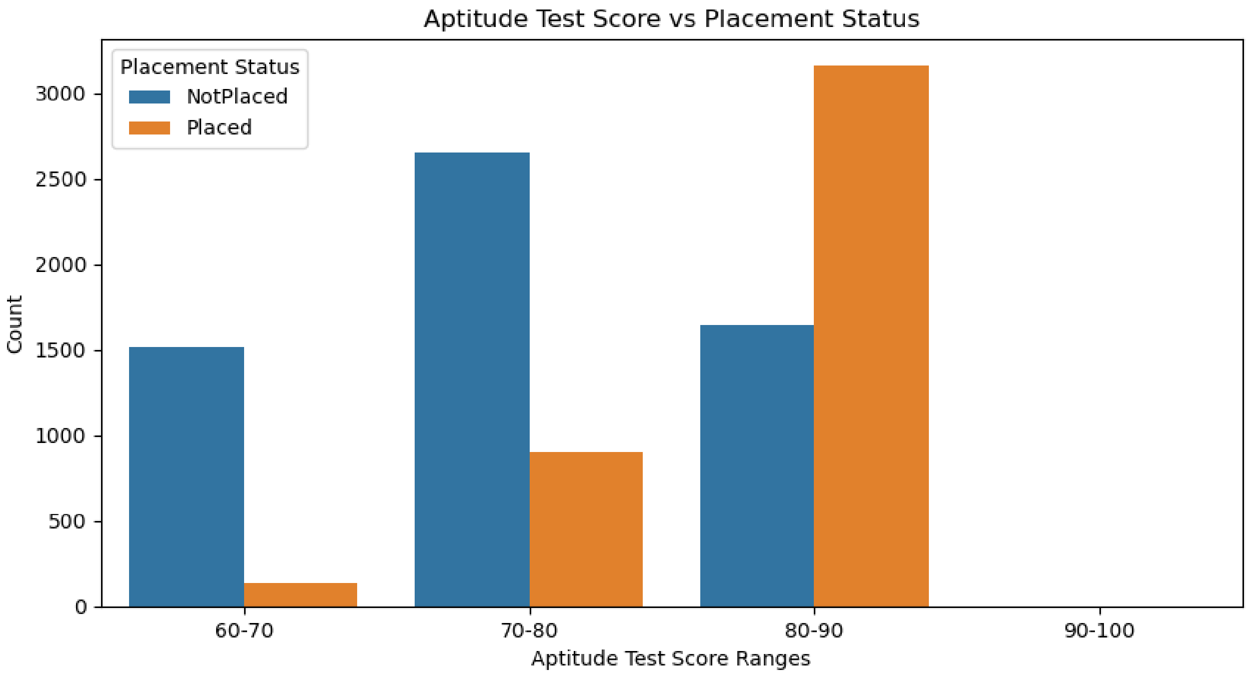
Key Observations

- This bar plot indicates that students who underwent placement training had a much higher placement rate than those who did not.

INNOMATICS RESEARCH LABS

EDA





Key Observations

- Low Soft Skills Ratings (3.5-4.0): Majority are Not Placed, which means lower soft skills may negatively impact placement chances.
- High Ratings (4.5-5.0): The number of Placed students is higher than Not Placed, suggesting that strong soft skills significantly improve placement chances.

Key Observations

- Students with scores between 60-80: More students are Not Placed compared to Placed.
- Scores between 80-90: There is a significant increase in placements, indicating that students with higher aptitude scores are more likely to be placed.

MODEL BUILDING

DATA:

- The dataset contains the following columns:

1. StudentID	7. SoftSkillsRating
2. CGPA	8. ExtracurricularActivities
3. Internships	9. Placement Training
4. Projects	10. SSC_Marks
5. Workshops/Certifications	11. HSC_Marks
6. AptitudeTestScore	12. PlacementStatus

- We use train_test_split to **split** the data.

Train Data: The data contains all the columns mentioned above. It is used for training and validation of the model.

Test Data: The data contains all the columns except the PlacementStatus column. We have to predict the placement status of each student using all the features of the test data.

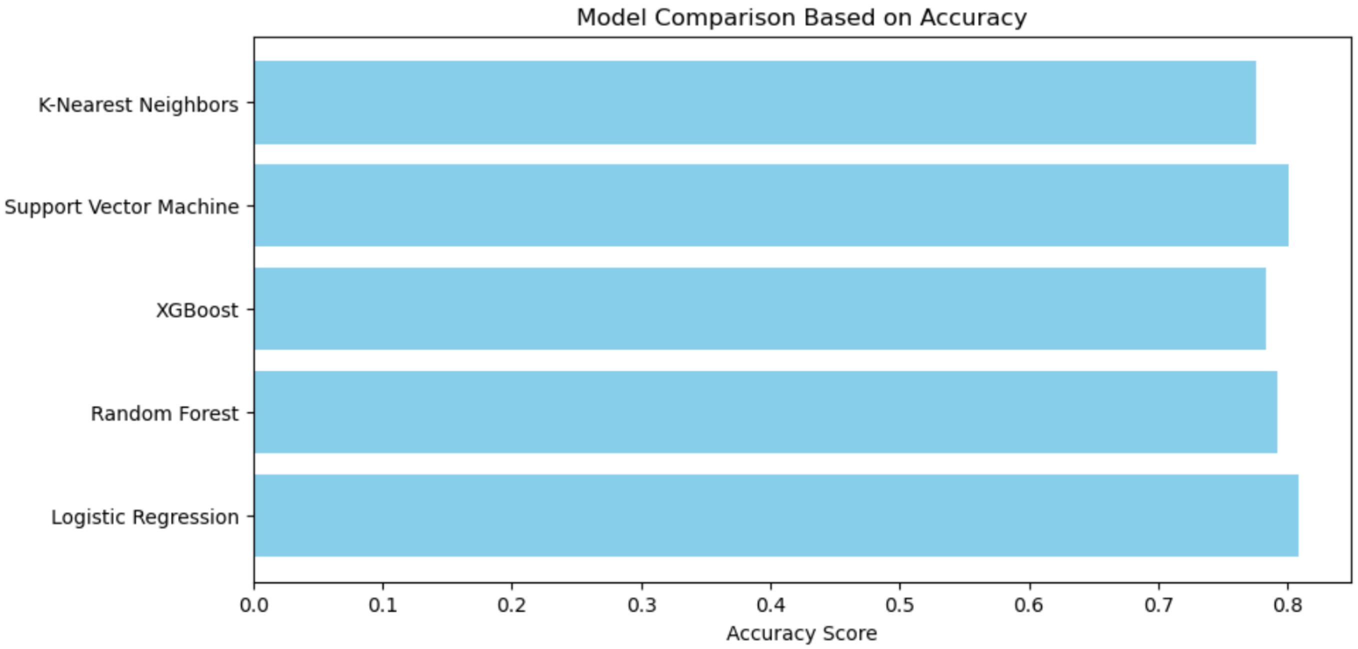
- After the data is split, we **scale** the data to normalize the numerical values.
- Then we apply **polynomial features** (degree = 2) to capture non-linear relationships.

MODEL BUILDING

- Now the model is ready for training.

MODELS:

- Tried various algorithms like Logistic Regression, KNN, SVM, XGBoost, Random Forest.

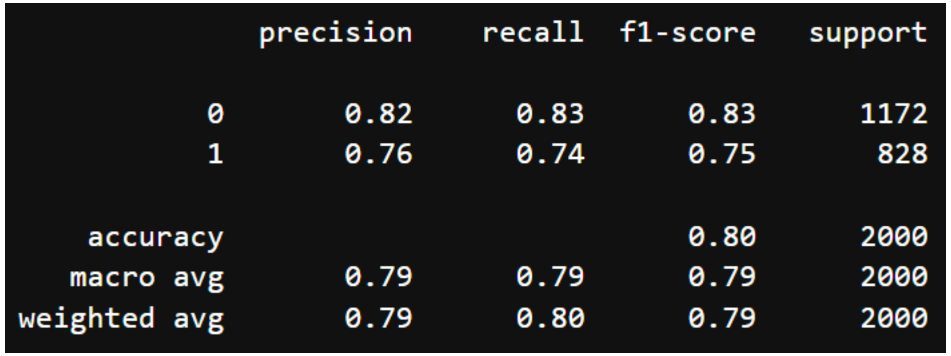


- Logistic Regression has the highest Accuracy Score, hence chose to train the model with the same.
- To improve the model's performance, we used GridSearchCV for finding the best hyperparameters, and trained the model with those parameters.

INNOMATICS RESEARCH LABS

MODEL BUILDING

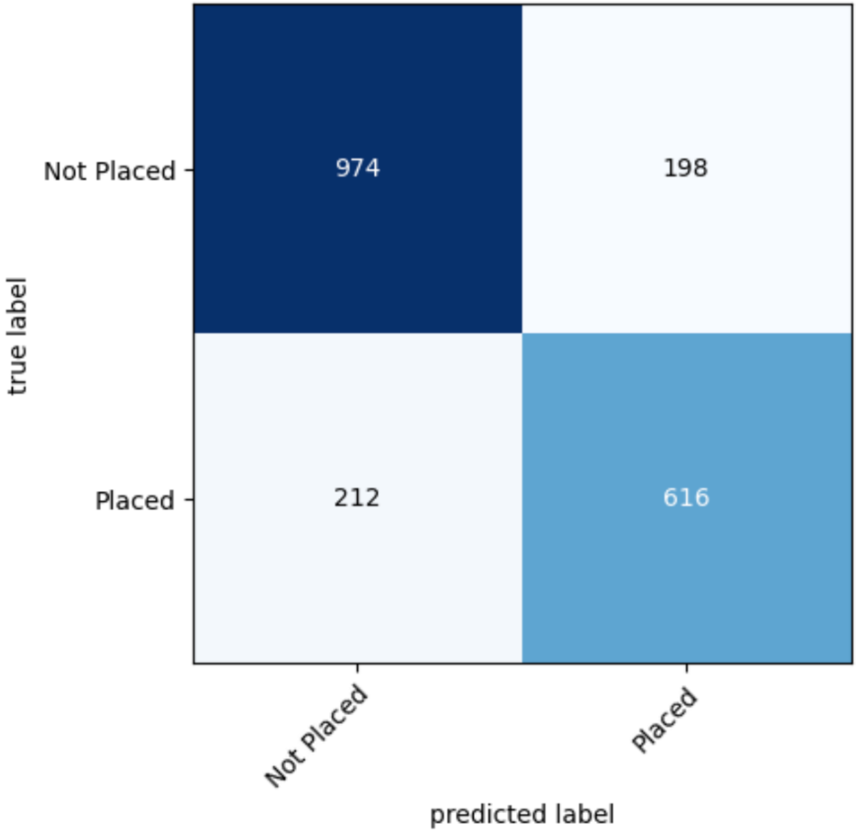
METRICS:
The metrics used to evaluate the model are Accuracy Score, Confusion Matrix, Precision, Recall, F1 Score and AUC-ROC Score.



```
              precision    recall  f1-score   support

           0       0.82      0.83      0.83      1172
           1       0.76      0.74      0.75       828

    accuracy                           0.80      2000
   macro avg       0.79      0.79      0.79      2000
weighted avg       0.79      0.80      0.79      2000
```



Precision: It measures the accuracy of positive predictions by checking how many of the predicted positive cases are actually positive.
Higher precision means fewer false positives (FP).
Recall: It measures the model’s ability to identify all actual positive cases.
Higher recall means fewer false negatives (FN).
F1-Score: It is the **harmonic mean** of precision and recall.
It is useful when there is an imbalance between false positives and false negatives, as it **provides a balanced evaluation**.

Key Business Question

What factors influence a student's placement status, and how can we predict whether a student will be placed based on academic performance, skills, and other attributes?

Conclusion (Key finding overall)

- Students with higher aptitude scores and better soft skills have a higher chance of getting placed.
- Doing more projects increases the likelihood of placement.
- Attending workshops and earning certifications improves placement opportunities.
- Using polynomial features improved the model's accuracy to 80%.
- These insights can help students focus on key skills to boost their employability.

Your Experience/Challenges Working on Machine Learning Project

- **Learned New Concepts -** I learned about data preprocessing (techniques like scaling, polynomial features, etc.), feature engineering, and model evaluation.
- **Choosing the Right Model -** I tried different models to find the best one for accuracy.
- **Improving Model Performance -** Using polynomial features and scaling helped improve results. Also tuning the hyperparameters using GridSearchCV helped improve model's performance.
- **Understanding Business Impact -** The project showed how data helps in predicting placements.
- **Learning Experience -** I gained hands-on experience with Python tools (like pandas, matplotlib and scikit-learn) for working with data, creating visualizations and building the model.

Q&A

THANK YOU