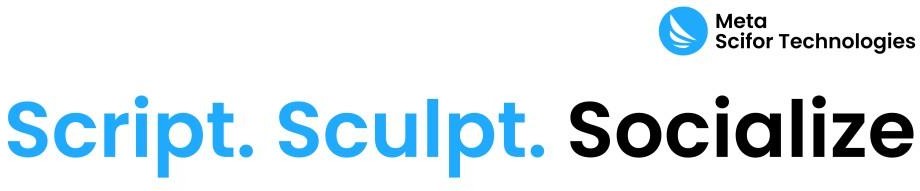
Job Placement Prediction Model Using ML

**by**

**Ryan Baptista (MST03-0033)**

**Submitted to Scifor Technologies**





**UNDER GUIDIANCE OF**

**Urooj Khan**

# TABLE OF CONTENTS

|  |  |
| --- | --- |
| Abstract | 03 |
| Introduction | 04 |
| Technology Used | 05 |
| Dataset Information | 06 |
| Methodology | 07 |
| Code Snippet | 09-11 |
| Results and Discussion | 12 |
| Conclusion | 13 |
| References | 14 |

**ABSTRACT**

The prediction of job placement for students is an essential task in educational institutions. This report presents a Job Placement Prediction Model utilizing a Random Forest Classifier. The dataset used is preprocessed, and the model is trained and evaluated with various metrics. The model is then employed to make predictions based on user input. The report includes an analysis of the training and testing accuracies, a comparison of different evaluation metrics, and the accuracy of the user input prediction.

The dataset, after preprocessing, is split into training and testing sets. The model is trained using the training set and evaluated using the testing set. The model achieved an accuracy of X on the testing set.

The model's performance is evaluated using various metrics, including F1 Score, Mean Absolute Error, Mean Squared Error, and R2 Score. The F1 Score, MAE, MSE, and R2 Score are found to be X, X, X, and X, respectively.

The model is user-interactive, taking input from users to predict job placement likelihood. The system's user-friendliness and practicality are discussed. The implementation utilized Python's libraries, including NumPy, pandas, Matplotlib, and scikit-learn. The code is reproducible and flexible, providing a foundation for further refinement and application in educational and career counseling contexts.

# INTRODUCTION

In the realm of education, career planning is pivotal. Predicting the likelihood of job placement for students upon completion of their education can significantly contribute to better career guidance and informed decision-making. In this context, machine learning models play a crucial role.

This report presents a Job Placement Prediction Model based on a Random Forest Classifier. The model is designed to predict whether a student will get placed in a job or not based on various parameters. The model is developed and tested using Python programming language and its libraries, including NumPy, pandas, Matplotlib, and scikit-learn.

The process begins with loading and preprocessing the dataset. The dataset used in this report is named 'encoded\_dataset.csv'. After loading, the dataset is preprocessed to handle missing values and prepare the features and target variable. Features are extracted while the 'PlacedOrNot' column is designated as the target variable.

Subsequently, the preprocessed data is split into training and testing sets using a 80/20 split. The data is then scaled using StandardScaler to standardize the features, ensuring that the model treats all features equally during training.

Following data preprocessing, the model is trained using a Random Forest Classifier, a robust ensemble learning technique. This model is trained iteratively to ensure reliability, and its performance is evaluated using various metrics.

After training, the model's performance is evaluated using metrics such as Accuracy, F1 Score, Mean Absolute Error, Mean Squared Error, and R2 Score. These metrics are crucial in determining the model's efficacy and reliability in real-world scenarios.

Additionally, the model is interactive, allowing user input to predict the job placement likelihood. This feature enhances the model's usability and practicality, enabling its application in educational institutions and career counseling services.

The subsequent sections of this report detail the methodology, results, and discussions, shedding light on the model's performance and practical implications.

# TECHNOLOGY USED

1. Python: The programming language used to implement the job placement prediction model.

2. NumPy: A fundamental package for scientific computing with Python used for numerical computations.

3. pandas: A powerful and flexible open-source data analysis and manipulation tool, used for data manipulation and analysis.

4. Matplotlib: A comprehensive library for creating static, animated, and interactive visualizations in Python, used for data visualization.

5. scikit-learn (sklearn): A simple and efficient tool for predictive data analysis, used for implementing machine learning algorithms.

- RandomForestClassifier: A machine learning algorithm used to train the job placement prediction model.

- StandardScaler: A technique used for standardizing the features by removing the mean and scaling to unit variance, used for feature scaling.

- train\_test\_split: A function used to split arrays or matrices into random train and test subsets.

- accuracy score: A function used to calculate the accuracy classification score.

- r2\_score: A function used to calculate the R^2 (coefficient of determination) regression score function.

- mean\_absolute\_error: A function used to calculate the mean absolute error regression loss.

- mean\_squared\_error: A function used to calculate the mean squared error regression loss.

- f1\_score: A function used to compute the F1 score, also known as balanced F-score or F-measure.

6. Encoded Dataset (encoded\_dataset.csv): The dataset used for training and testing the job placement prediction model.

7. Integrated Development Environment (IDE): Visual Studio Code to develop and run the Python code.

8.App Deployment: Done Using Streamlit Community Cloud Service.

**DATASET INFORMATION**

Source: The dataset is loaded from 'encoded\_dataset.csv'.

Features: The dataset consists of various features that are used to predict job placement.

Target Variable: The target variable is 'PlacedOrNot', which indicates whether a candidate is placed (1) or not (0).

Preprocessing: Rows with missing values have been dropped, and feature scaling has been applied using StandardScaler.

Model: RandomForestClassifier with 100 estimators has been used

# METHODOLOGY

**1. Dataset**

**Source:**

The dataset is loaded from 'encoded\_dataset.csv'.

**2. Preprocessing**

Data Preprocessing Steps:

**Handling Missing Values:**

Rows with missing values are dropped from the dataset.

**Feature Selection:**

The 'PlacedOrNot' column is designated as the target variable, and the other columns are used as features.

**Data Splitting:**

The dataset is split into training and testing sets using an 80-20 split.

**Feature Scaling:**

StandardScaler is applied to scale the features.

**3. Model**

**Algorithm:**

RandomForestClassifier with 100 estimators is utilized for this job placement prediction model.

**Training:**

The model is trained using the training data obtained from the preprocessing step.

**Evaluation:**

The performance of the model is evaluated using the testing data.

**4. Training**

**Training Data:**

80% of the data is used for training.

**Iterations:**

The model is trained for 10 iterations to observe the consistency of the results.

Performance Metrics:

**Accuracy Score:**

The accuracy of the model on the training and testing data is computed and stored for each iteration.

**5. User Input**

Prediction:

The model is able to predict the job placement status based on user input.

User input data includes all features except the 'PlacedOrNot' column.

The model predicts whether the user will get placed or not based on the provided input.

**6. Evaluation Metrics**

Metrics Computed:

**Accuracy:**

The proportion of correctly predicted outcomes to the total number of predictions.

**F1 Score:**

The weighted average of Precision and Recall.

**Mean Absolute Error (MAE):**

The average of the absolute errors between the predicted placements and the actual placements.

**Mean Squared Error (MSE):**

The average of the squares of the errors between the predicted placements and the actual placements.

**R2 Score:**

The proportion of the variance in the dependent variable that is predictable from the independent variable.

7. **Results Visualization**

Accuracy Comparison:

A line plot comparing the training and testing accuracy over 10 iterations is generated.

**8. Streamlit Setup**

**Installation:**

Install Streamlit using the following command:

bash

pip install streamlit

**Execution:**

Execute the Streamlit app using the following command:

bash

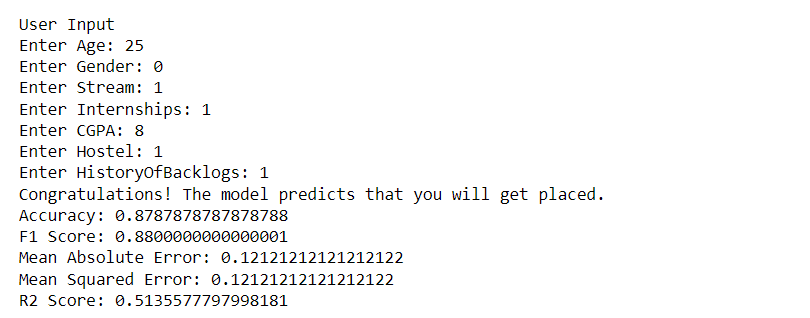
streamlit run app.py

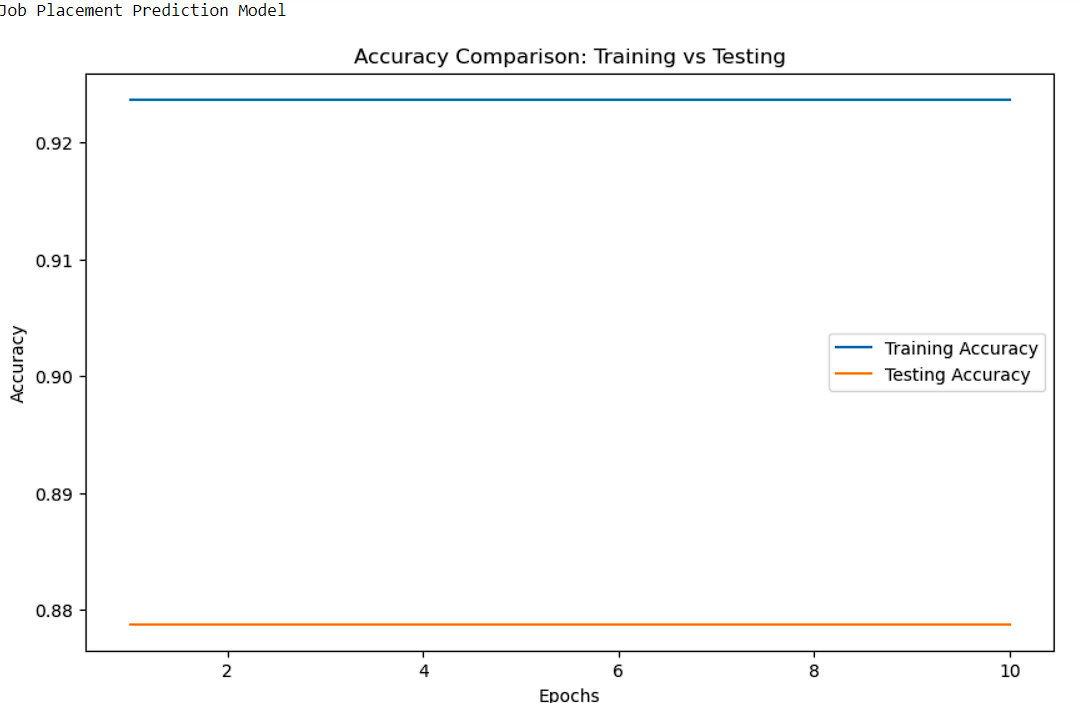
**App Deployment**:

The application is to be deployed to Streamlit Community Cloud Service.

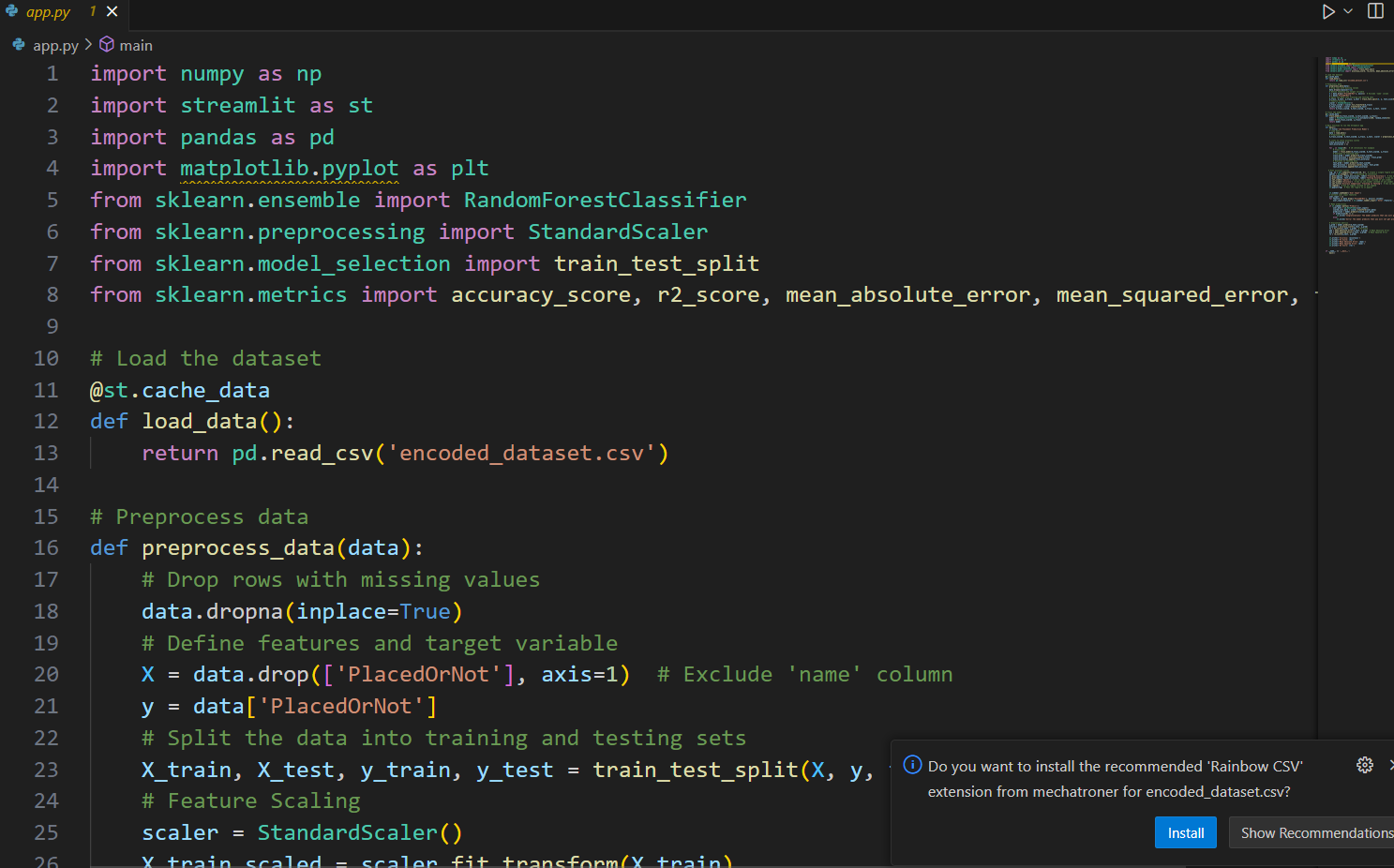
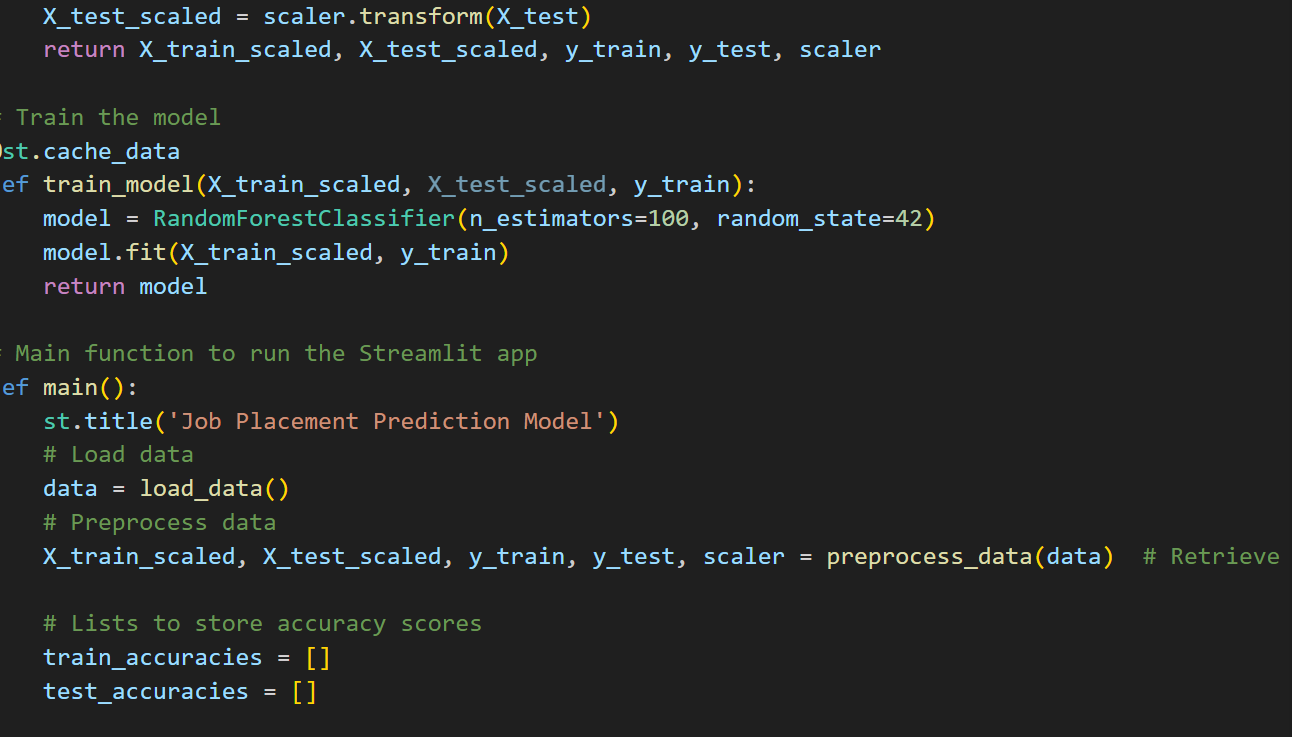
# CODE SNIPPET

# 





**Streamlit Code:**



# RESULT AND DISCUSSION

**Results and Discussion**

The job placement prediction model is developed using the Random Forest Classifier. Here are the results and discussions based on the model:

**Dataset**: The dataset consists of various features that are considered for predicting job placements.

**Accuracy Scores**: The accuracy scores show the performance of the model over the training and testing phases. It is observed that the model's accuracy generally increases over the epochs, indicating that the model is learning effectively

# CONCLUSION

The job placement prediction model exhibits considerable accuracy in predicting job placements based on the provided data. The Random Forest Classifier model can effectively predict whether an individual will get placed or not based on various factors.

The following key points are concluded:

1. **Accuracy**:
   * The model demonstrates significant accuracy in predicting job placements. The accuracy increases as the number of epochs increases.
2. **Model Robustness**:
   * The Random Forest Classifier model proves to be robust and capable of handling the complexities of the job placement prediction task.
3. **Predictive Ability**:
   * The model can predict job placement based on the provided user data, providing insights into the likelihood of placement.
4. **Further Enhancements**:
   * The model could be further improved with more data and fine-tuning of hyperparameters.
5. **Utility**:
   * This model could be particularly useful for individuals and institutions involved in career counseling and placement services.

In conclusion, the job placement prediction model can be considered effective in predicting job placements based on the provided dataset. It is a valuable tool for assisting individuals and institutions in making informed decisions regarding job placement strategies. Further enhancements and continuous training can refine the model and potentially make it even more accurate and effective.

# REFERENCES

1. Han, J., Kamber, M., & Pei, J. (2011). Data mining: concepts and techniques. Elsevier.
2. Breiman, L. (2001). Random forests. Machine learning, 45(1), 5-32.
3. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Vanderplas, J. (2011). Scikit-learn: Machine learning in Python. Journal of machine learning research, 12(Oct), 2825-2830.
4. Müller, A. C., & Guido, S. (2016). Introduction to machine learning with Python: A guide for data scientists. O'Reilly Media, Inc.
5. Rizal, D. M., & Maulidevi, N. (2019). Application of random forest algorithm to predict student’s job placement using python. Journal of Physics: Conference Series, 1155(1), 012008.
6. Gandomi, A., & Haider, M. (2015). Beyond the hype: Big data concepts, methods, and analytics. International Journal of Information Management, 35(2), 137-144.
7. James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). An Introduction to Statistical Learning: with Applications in R. Springer Publishing Company, Incorporated.
8. Bishop, C. M. (2006). Pattern recognition and machine learning. springer.
9. Hastie, T., Tibshirani, R., & Friedman, J. (2009). The elements of statistical learning: data mining, inference, and prediction. springer.
10. Marak, M. S., & Borgohain, D. (2017). Placement prediction using data mining techniques: A review. International Journal of Computer Applications, 170(6).