A close up of a sign

AI-generated content may be incorrect.

**Department of Computer Science and Engineering**

**School of Computer Science and Engineering**

**Manipal University Jaipur**

**Jaipur, Rajasthan - 303007**

Internship Project Report

On

**Machine Learning Capstone Project**

Submitted to



By

Anwesha Singh

anweshasingh0611@gmail.com

(B.Tech. 2023-2027)

**CONTENTS**

|  |  |
| --- | --- |
|  | Page |
| Introduction | 3 |
| Exploratory Data Analysis | 4-6 |
| Roadmap | 7 |
| Uploading Dataset | 8 |
| Cleaning Dataset | 9-11 |
| Train-Test split and Training | 12-14 |
| Feature Importance | 15-17 |
| Model Evaluation | 18-20 |
| Comparing accuracy | 21 |
| Testing | 22-23 |
| Conclusion | 24 |

1. **INTRODUCTION**

**1.1 Project Overview:** Business Problem: To ensure there is no discrimination between employees, it is imperative for the Human Resources department of Company X to maintain a salary range for each employee with a similar profile. Apart from the existing salary, a considerable number of factors, such as an employee’s experience and other abilities, are evaluated during interviews. Given the data related to individuals who applied to Company X, models can be built that automatically determine the salary to be offered if a prospective candidate is selected. This model seeks to minimize human judgment in salary decisions.

**1.2 Goal & Objective:** The objective of this exercise is to build a model, using historical data, that will determine the salary to be offered to an employee, minimizing manual judgment in the selection process. The approach aims to be robust and eliminate any discrimination in salary among employees with similar profiles.

Upon reading the Problem-Statement, this seems like a **supervised regression** problem, where the goal is to predict the **salary (continuous variable)** of a candidate based on features like experience, skills, etc.

1. **EXPLORATORY DATA ANALYSIS**

Step in which we analyze, understand, and summarize the characteristics of the dataset. It is the first step before applying any ML model.



2.1 Snip-it of the dataset

|  |  |  |  |
| --- | --- | --- | --- |
| **COLUMN NAME** | **CONTENT** | **DATATYPE** | **USE** |
| IDX | Index | ID | Ignore |
| Applicant\_ID | Unique ID for each applicant | Categorical | Might be helpful |
| Total\_Experience | Years of work experience | Numerical | Strongly helpful in salary prediction |
| Total\_Experience\_in  \_field\_applied | Experience for the role applied | Numerical | Important |
| Department | Department to which the role belongs to | Categorical | Used to assign department |
| Role | Applied job role | Categorical | Significant impact on salary |
| Industry | Industry where applicant previously worked at | Categorical | Might be helpful |
| Organization | Previous company | Categorical | Might be helpful |
| Designation | Designation at previous job | Categorical | Decides seniority |
| Education | Education level | Categorical | Important influence |
| Graduation\_  Specialization | Degree subject | Categorical | Might have a minor impact |
| University\_Grad | Undergraduate university | Categorical | Useful if elite universities are favoured |
| Passing\_Year\_  Of\_Graduation | Year of graduation | Numerical | Little influence |
| PG\_Specialization | PG degree subject | Categorical | Might have some influence |
| University\_PG | PG university | Categorical | Useful if elite universities are favoured |
| Passing\_Year  \_Of\_PG | Year PG degree was completed | Numerical | Little influence |
| PHD\_Specialization | PHD subject | Categorical |  |
| University\_PHD | PHD university | Categorical | Useful if elite universities are favoured |
| Passing\_Year  \_Of\_PHD | Year of PHD completion | Numerical |  |
| Current\_Location | Where applicant is currently located | Categorical | Might have impact |
| Preferred\_  Location | Location where applicant wants to work | Categorical | Impacts applicant’s willingness to work |
| Current\_CTC | Current salary | Numerical | Greatly impacts |
| Inhand\_Offer | If they have any other offers | Binary | Influences applicant’s salary demand |
| Last\_Appraisal\_  Rating | Performance rating at previous organization | Categorical |  |
| No\_Of\_Companies  \_Worked | Number of companies the applicant has worked at | Numerical |  |
| Number\_Of\_  Publications | Research papers of publications | Numerical | Might influence |
| Certifications | Number of certifications held | Numerical | Might help when determining skills |
| International\_degree  \_any | If an applicant has any international degrees | Binary | Might influence |
| Expected\_CTC | Target variable | Numerical |  |

2.2 Data Analysis

**2.3 MAJOR OBSERVATIONS IN THE DATASET**

* There are many categorical features in the given dataset.
* Some features (like Organization, University\_Grad) are likely to have a very high cardinality.
* Strong predictors are- Total\_Experience, Current\_CTC, Certifications
* There are a lot of NA values or missing values.

1. **ROADMAP**

`

Testing

Evaluation

Modelling

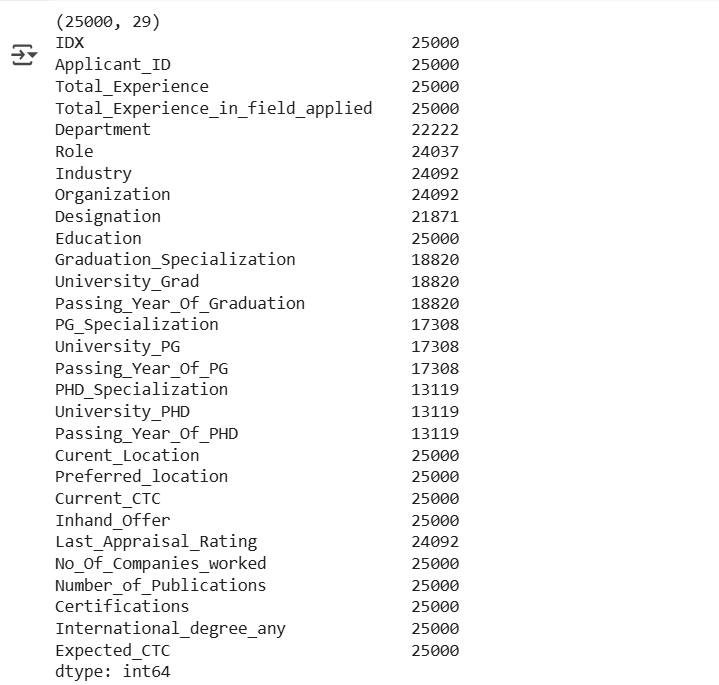
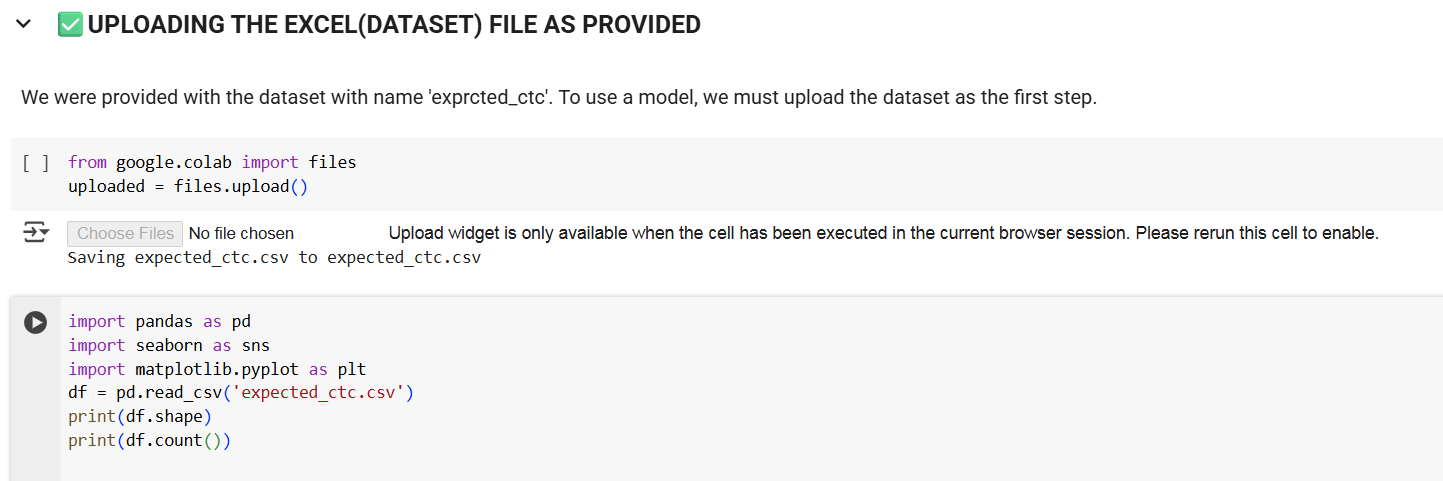
EDA

Preprocessing

1. **UPLOADING THE EXCEL(DATASET) FILE AS PROVIDED**

We were provided with the dataset with name 'exprcted\_ctc'. To use a model, we must upload the dataset as the first step.

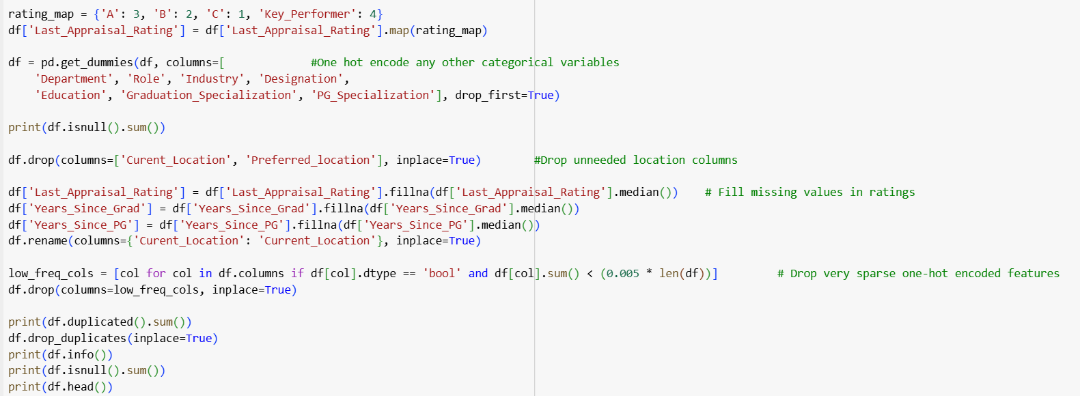
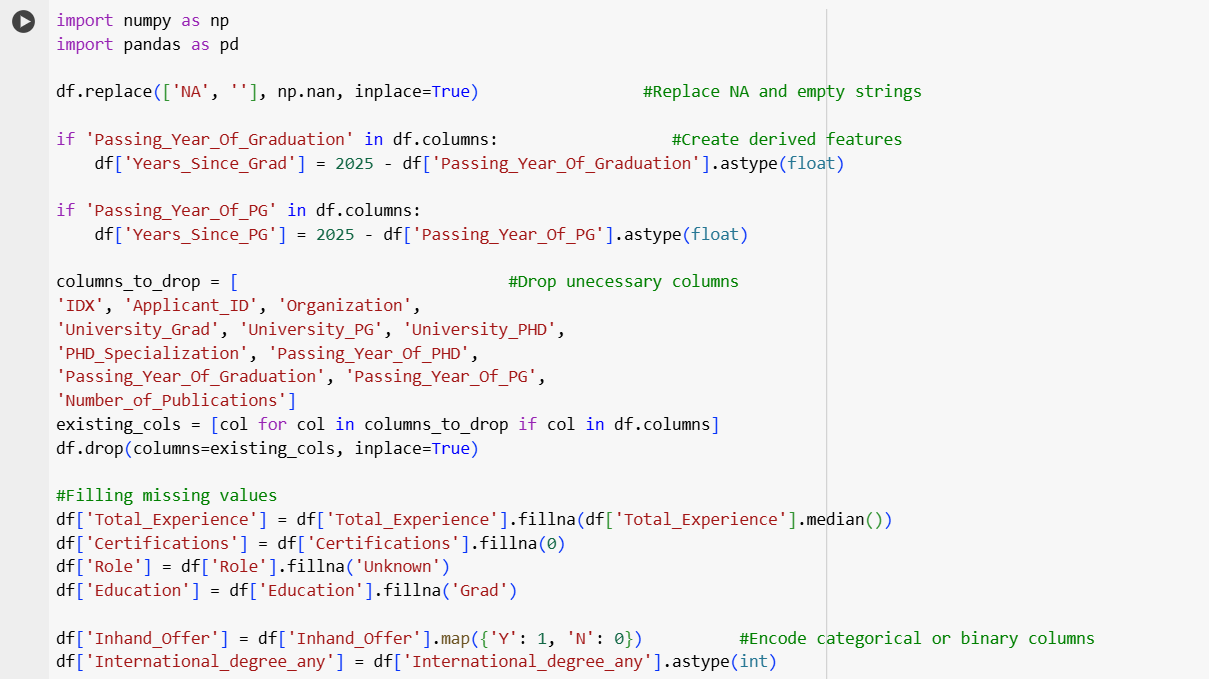
This step is the same regardless the model you use.

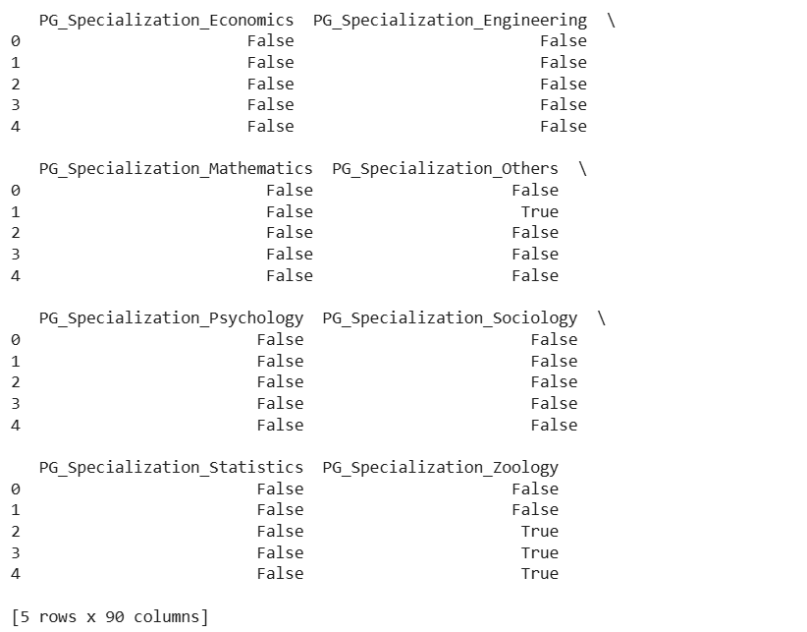
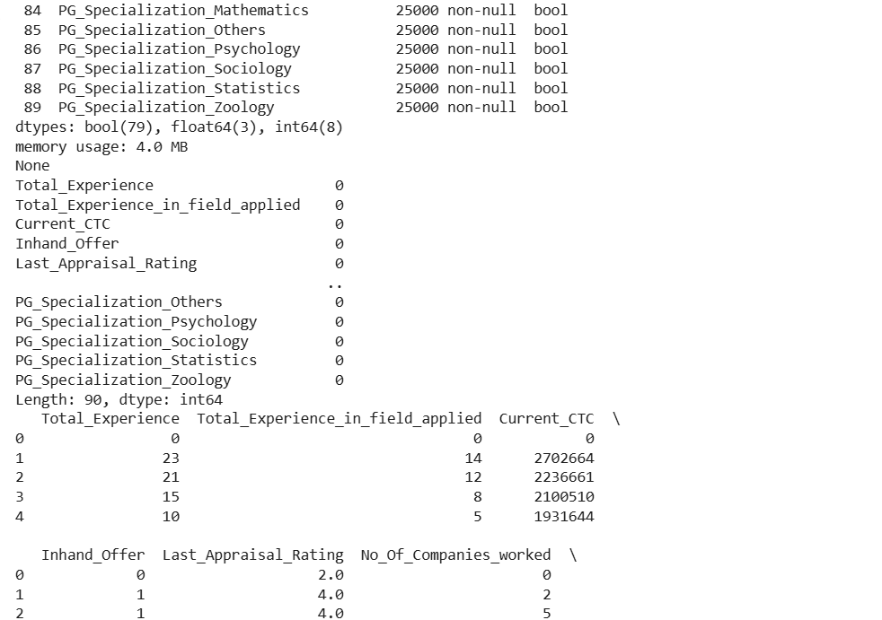
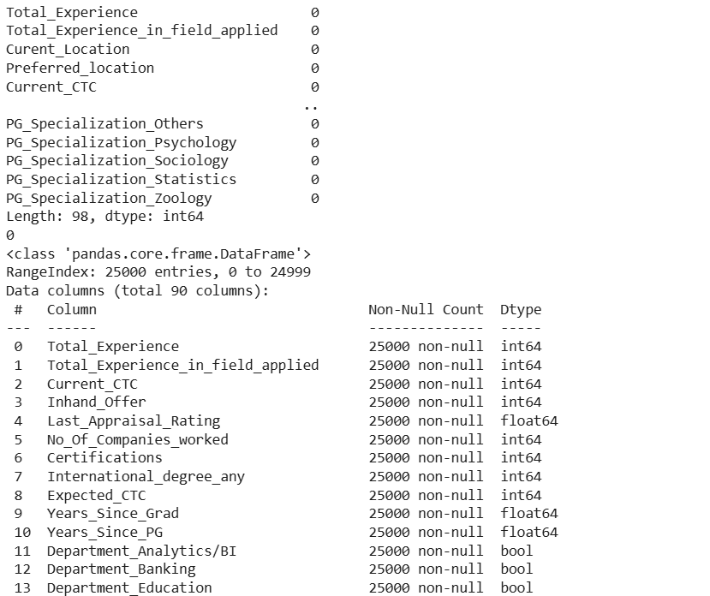


1. **CLEANING THE DATASET**

Fixing, or removing incorrect, incomplete, or irrelevant data before training the model.

* Replace null values or empty strings
* Create derived features as required
* Dropping any unnecessary columns
* Filling any missing values
* Encode categorical or binary columns (there are many categorical features.)
* One hot encode other categorical variables





1. **TRAIN-TEST SPLIT AND TRAINING THE MODEL**

Dividing the dataset into two parts:

* **Training set-** I used 80% of dataset to train the model.
* **Testing set-** I used 20% of dataset to do the testing of the model.

It helps when checking if the model generalizes well and if it avoids overfitting.

**CHOOSING THE MODEL**

Based on the Exploratory Data Analysis, this problem comes out to be a **regression** problem.

I will use three models on the dataset and compare each one’s accuracy and give the output based on the most accurate one.

I’ll be using the following three models:

* **Linear Regression-** A supervised ML model used when predicting continuous value with the help of the linear relationship between input features and target.

It tries to find the best fitting straight line which helps minimize the difference between actual and predicted values.

* **Random Forest Regressor-** A supervised ML model which uses an ensemble of decision trees to predict continuous values. It picks out random subsets of data from the dataset and takes the average of the predictions.
* **Gradient Boosting Regressor-** A supervised ML model that makes an ensemble of weaker models in a sequential manner. Each model in the ensemble rectifies the errors of the previous model.

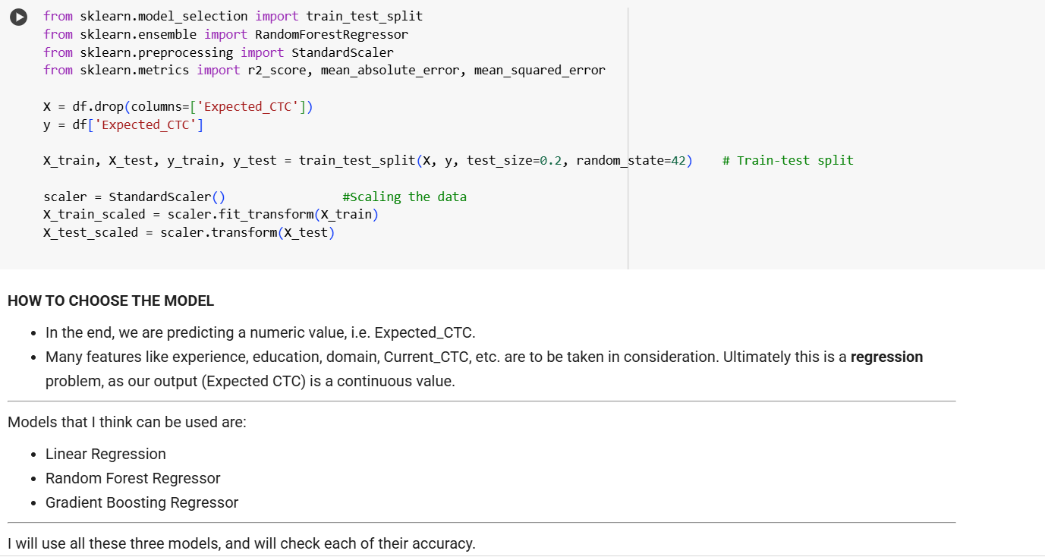
**6.1 Linear Regression**

A screenshot of a computer program

AI-generated content may be incorrect.A screenshot of a computer

AI-generated content may be incorrect.

**6.2 Random Forest Regressor**

A computer screen shot of a computer program

AI-generated content may be incorrect.

**6.3 Gradient Boosting Regressor**

A screenshot of a computer program

AI-generated content may be incorrect.A screenshot of a computer code

AI-generated content may be incorrect.

1. **FEATURE IMPORTANCE**

Helps us determine which input columns have the biggest impact on model’s predictions.

Useful in selecting the features for predictions. It also helps improve model’s performance and accuracy.

* 1. **Linear Regression**

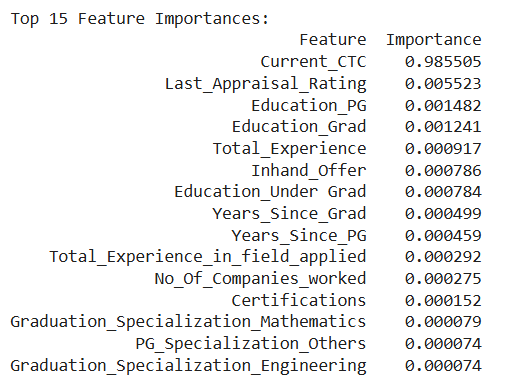
A screen shot of a computer code

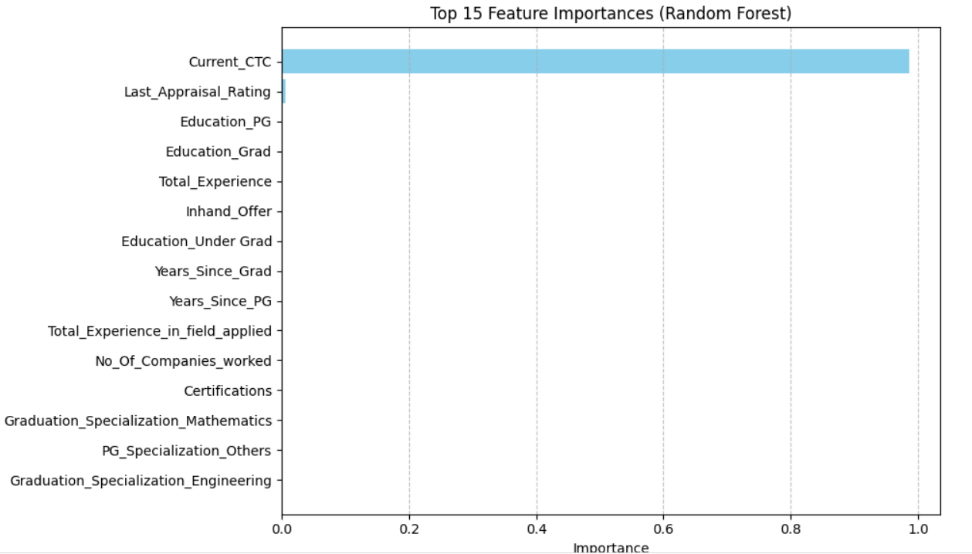
AI-generated content may be incorrect.A graph with blue squares

AI-generated content may be incorrect.

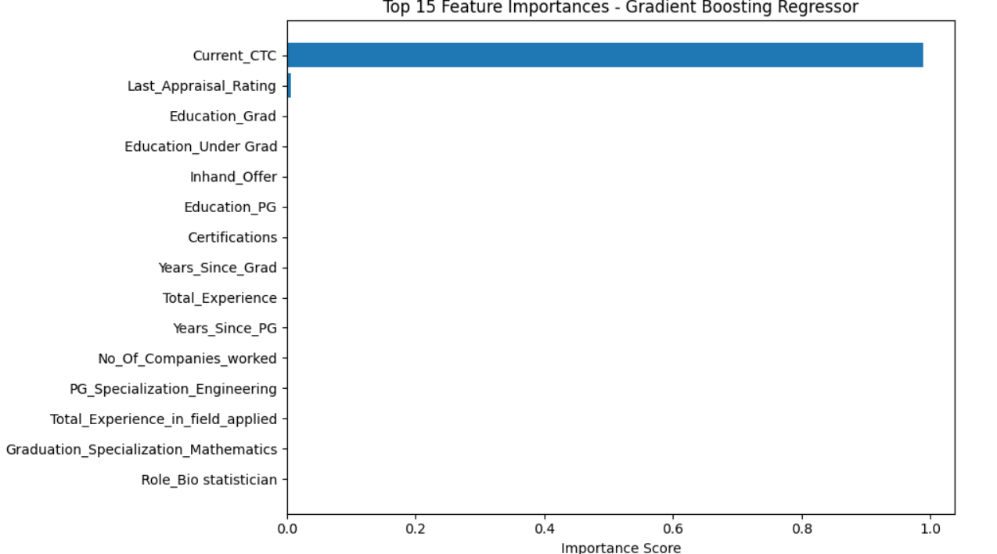
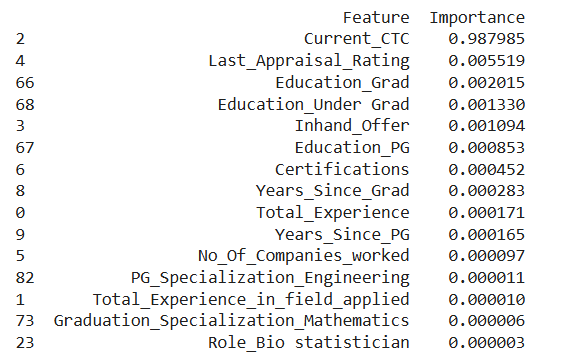
* 1. **Random Forest Regressor**







* 1. **Gradient Boosting Regressor**



1. **MODEL EVALUATION**

Check how well the model performs after training on unseen test data. It helps measure the accuracy and errors.

We often use following metrics to evaluate a model.

**MSE:**

 Measures the average squared difference between actual and predicted values.

Useful for understanding how far predictions are from actual values.

A black and white background with black text

AI-generated content may be incorrect.

**Lower MSE = better model**

**R² Score**:

Coefficient of Determination. Used in regression to measure how well a model explains variances in the target column.

A math equations on a white background

AI-generated content may be incorrect.

Ranges between 0 to 1 (or negative if very poor).

Tells how good your model is compared to a baseline (mean) model.

R² = 1 means perfect prediction, R² = 0 means predictions are no better than guessing the mean.

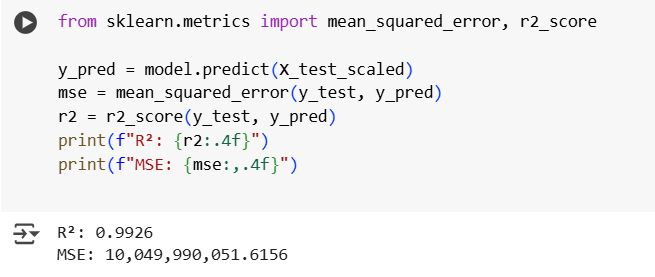
**MAE:**

Used in regression tasks to measure average absolute difference between the actual values and the predicted values.

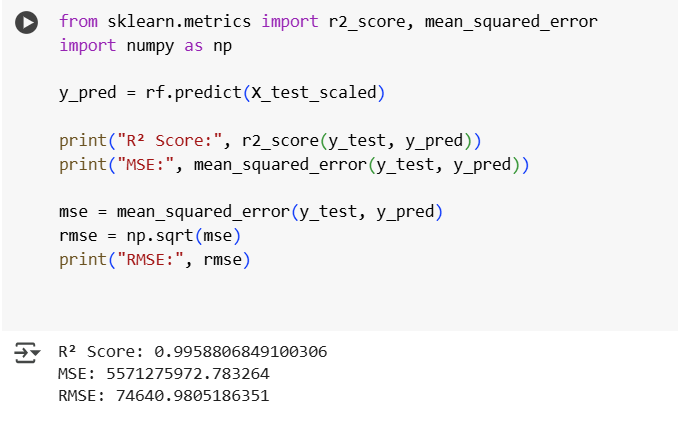
A black text on a white background

AI-generated content may be incorrect.

* 1. **Linear Regressor Model**



* 1. **Random Forest Regressor**



* 1. **Gradient Boosting Regressor**

A screenshot of a computer code

AI-generated content may be incorrect.

1. **COMPARING THE ACCURACY OF EACH MODEL**

* Model evaluation output for **‘Linear Regression’** model:

**R²:** 0.9926

**MSE:** 10,049,990,051.6156

* Model evaluation output for **‘Random Forest Regressor’** model:

**R²:** 0.9958806849100306

**MSE:** 5571275972.783264

* Model evaluation output for **‘Gradient Boosting Regressor’** model:

**R²:** 0.9954361344875179

**MSE:** 6172519877.059157

Comparing these models based on these outputs:

**Random Forest Regressor** has the highest R2 score, which is it experiences about **99.59%** variances in salaries.

It also has **the lowest Mean Squared Error**, which means that it has the lowest average prediction error among the others.

**Hence, the outputs from Random Forest Regressor model are the most accurate and dependable.**

1. **TESTING THE MODEL**

I created a dictionary with a random candidate's entries to test the model.

* **Linear Regressor**

Expected Salary (CTC): ₹1,169,836.46

A screenshot of a computer program

AI-generated content may be incorrect.

* **Random Forest Regressor**

Expected Salary (CTC): ₹1,214,645.26 (Most likely, most accurate)



* **Gradient Booster Regressor**

Expected Salary (CTC): ₹1,085,973.99

A screenshot of a computer program

AI-generated content may be incorrect.

1. **CONCLUSION**

Throughout this project, the goal was to make a ML model predicting expected salary for applicants applying for a job at some company X, while minimizing manual judgement and eliminating any discrimination in salary among applicants with similar qualifications.

To achieve this, I implemented three regression models:

* Linear Regression
* Random Forest Regressor
* Gradient Boosting Regressor

Dataset was cleaned and preprocessed thoroughly and then each model was trained.

Each model was evaluated on metrics such as R2 score and Mean Squared Error (MSE). The **Random Forest Regressor model** showed the best and most accurate results and performed the best, with the R2 score of 0.9959 and lowest MSE, which signifies that it was the most accurate.

Hence, if we predict salaries using Random Forest regressor we can significantly ensure reduced discrimination and accuracy. This model can assist the company to get the job done while ensuring transparency.

Link to my code repository on github:

<https://github.com/Anwesha-code/LaunchEd_CapstoneProject>