Chance of Admission Prediction

# 🎯 Objective

To build a Machine Learning regression model that predicts the probability of a student’s admission to a graduate program based on academic performance and profile features such as GRE, TOEFL, CGPA, and research experience.

# 📊 Dataset Overview

The dataset was sourced from the YBI Foundation and consists of 400 records, each representing a student application. The dataset includes the following features:

- GRE Score  
- TOEFL Score  
- University Rating  
- SOP (Statement of Purpose strength)  
- LOR (Letter of Recommendation quality)  
- CGPA  
- Research (0 or 1)  
- Chance of Admit (target variable)

# 🧰 Technologies and Libraries Used

The following libraries were used in this project:  
- pandas  
- numpy  
- matplotlib  
- seaborn  
- sklearn (model\_selection, linear\_model, metrics)

# 🧹 Data Preprocessing

Column names were cleaned and unnecessary columns were dropped. No missing values were found. Features (X) and target (y) were defined.

# 📈 Exploratory Data Analysis

A correlation heatmap was plotted to identify important features. CGPA, GRE, and TOEFL Scores were found to have the highest correlation with the target variable.

# 🤖 Model Training

The dataset was split into training and test sets using an 80-20 split. A Linear Regression model was trained using the training data.

# 📏 Model Evaluation

The model's performance was evaluated using R² Score and Mean Squared Error (MSE):  
- R² Score: 0.8212  
- MSE: 0.0046  
A scatter plot of actual vs predicted values was also generated to visually inspect performance.

# 🔮 Prediction Example

Sample Student Profile:

- GRE Score: 325  
- TOEFL Score: 112  
- University Rating: 4  
- SOP: 4.5  
- LOR: 4  
- CGPA: 9.1  
- Research: 1

Predicted Chance of Admission: 0.80

# 📘 Conclusion

The Linear Regression model effectively predicted admission chances using key academic and profile features. CGPA, GRE, and TOEFL scores were found to be the most impactful. The model achieved a high R² score, indicating strong predictive performance.

# 🔧 Future Improvements

- Use advanced models like Ridge, Lasso, or Random Forest.  
- Perform cross-validation.  
- Apply hyperparameter tuning.  
- Deploy the model using a web application (Flask/Streamlit).