**Machine Learning – Assignment 2**

**Ganesh G (25MML0043)**

**Link:** https://drive.google.com/drive/folders/1HIDK8\_eav175ae1AAMzFhqMIeCRqa6cM?usp=sharing

1. **Decision Tree**

**AIM:**

To understand and apply decision tree algorithms for multi-class classification problems by building, evaluating, and interpreting a decision tree model that predicts prescribed drug types based on patient medical and demographic features.

**INTRODUCTION:**

A decision tree is a supervised machine learning algorithm used for classification tasks, among others. It works by recursively splitting the data based on feature values into subsets, aiming to create groups as homogeneous as possible with respect to the target variable. This splitting process forms a tree-like structure with a root node, internal decision nodes representing feature-based tests, branches showing the outcomes of these tests, and leaf nodes which provide the classification labels or predictions.

**REAL WORLD APPLICATION:**

**1. Healthcare**

* Disease diagnosis: Doctors use decision trees to determine whether a patient has a certain disease based on symptoms, test results, and medical history.
* Example: Predicting if a patient has diabetes using age, BMI, and glucose level.

**2. Finance**

* Loan approval: Banks apply decision trees to decide whether to approve a loan by analyzing income, employment history, credit score, and repayment history.
* Fraud detection: Identifying fraudulent transactions by analyzing transaction amount, location, and spending patterns.

**3. Retail & Marketing**

* Customer segmentation: Retailers use decision trees to classify customers into groups (e.g., high spenders, bargain seekers) for targeted marketing.
* Churn prediction: Telecom companies predict whether a customer will leave based on usage data and complaints.

**4. Manufacturing**

* **Quality control**: Used to predict whether a product will pass or fail inspection based on features like material quality, temperature, or machine settings.

**5. Education**

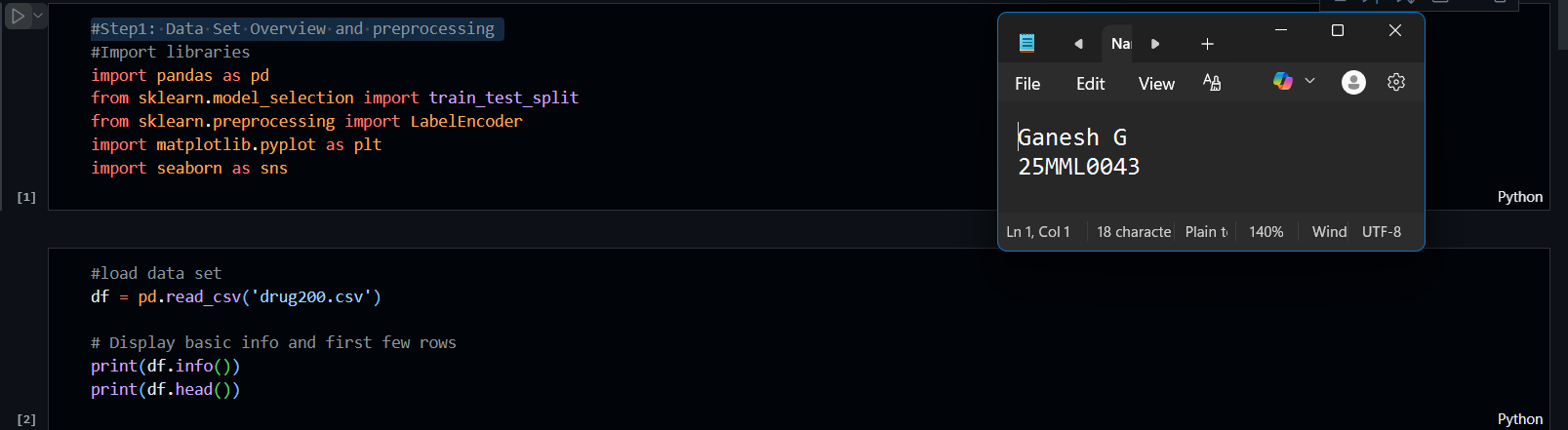
* **Student performance prediction**: Schools use decision trees to predict whether a student will pass/fail based on attendance, study hours, and past grades.

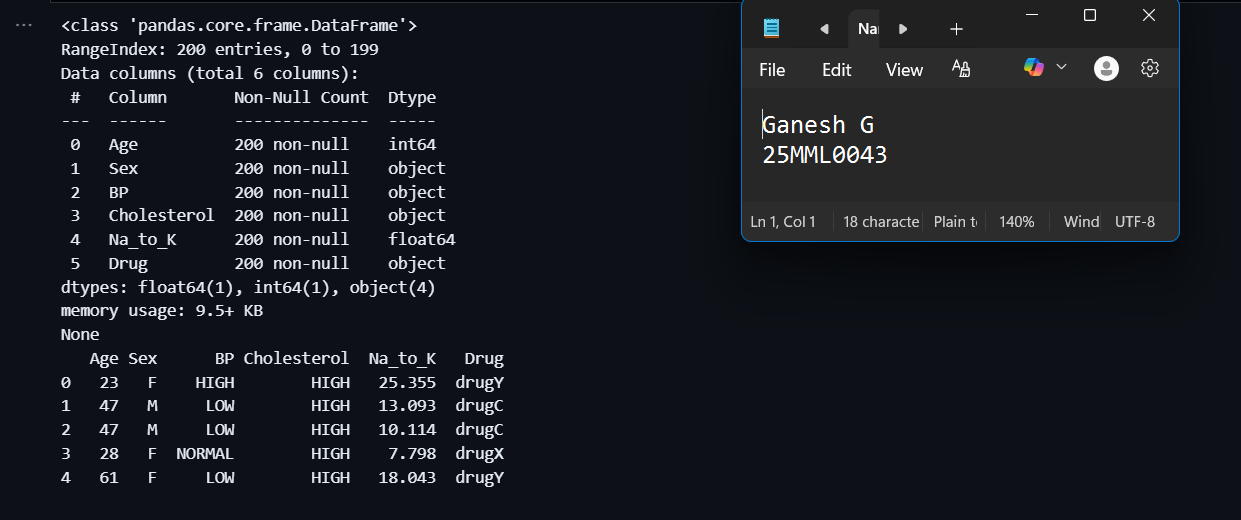
**ALGORITHM:**

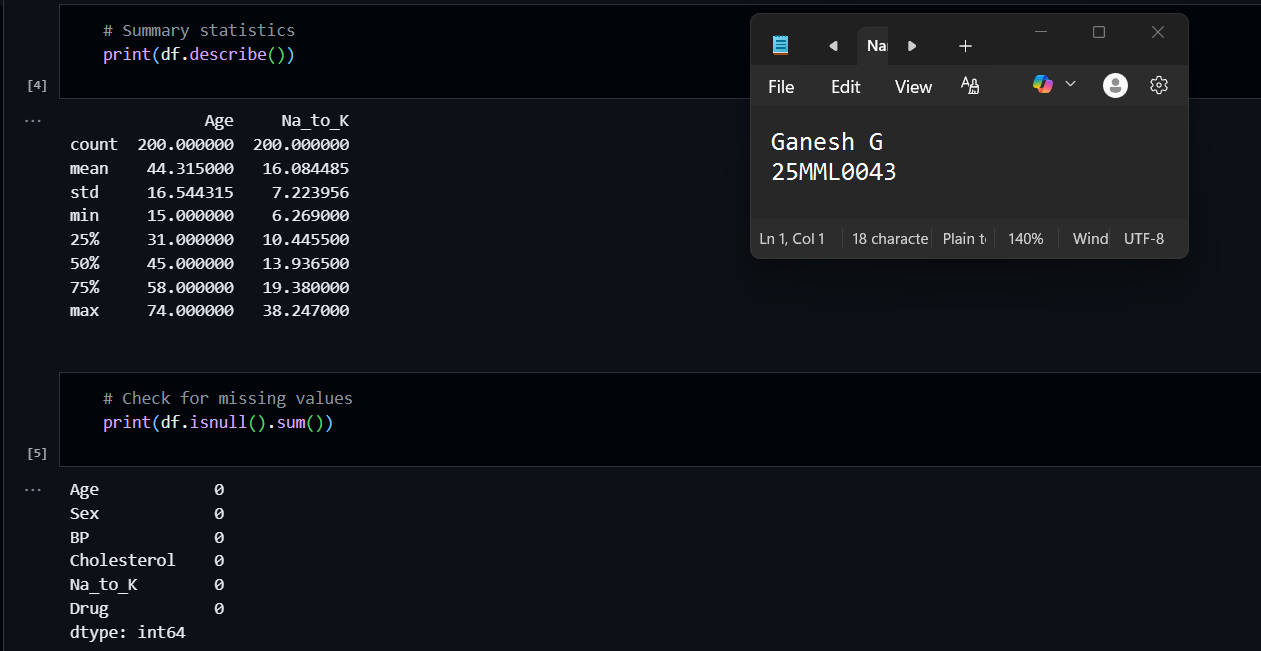
1. **Start with the full dataset** as the root node.
2. **Check stopping criteria**
   * If all data points belong to the same class → make this a **leaf node** with that class.
   * If no features remain → make this a **leaf node** with the majority class.
3. **Select the best feature to split**
   * Calculate a **splitting criterion** (such as *Information Gain*, *Gain Ratio*, or *Gini Index*) for each feature.
   * Choose the feature that best separates the data.
4. **Split the dataset** into subsets based on the chosen feature’s values.
5. **Create child nodes** for each subset.
6. **Repeat steps 2–5** recursively for each child until:
   * Maximum depth is reached, OR
   * All nodes are pure (only one class), OR
   * No improvement in splitting criterion.
7. **Label leaf nodes** with the majority class (if pure class not possible).

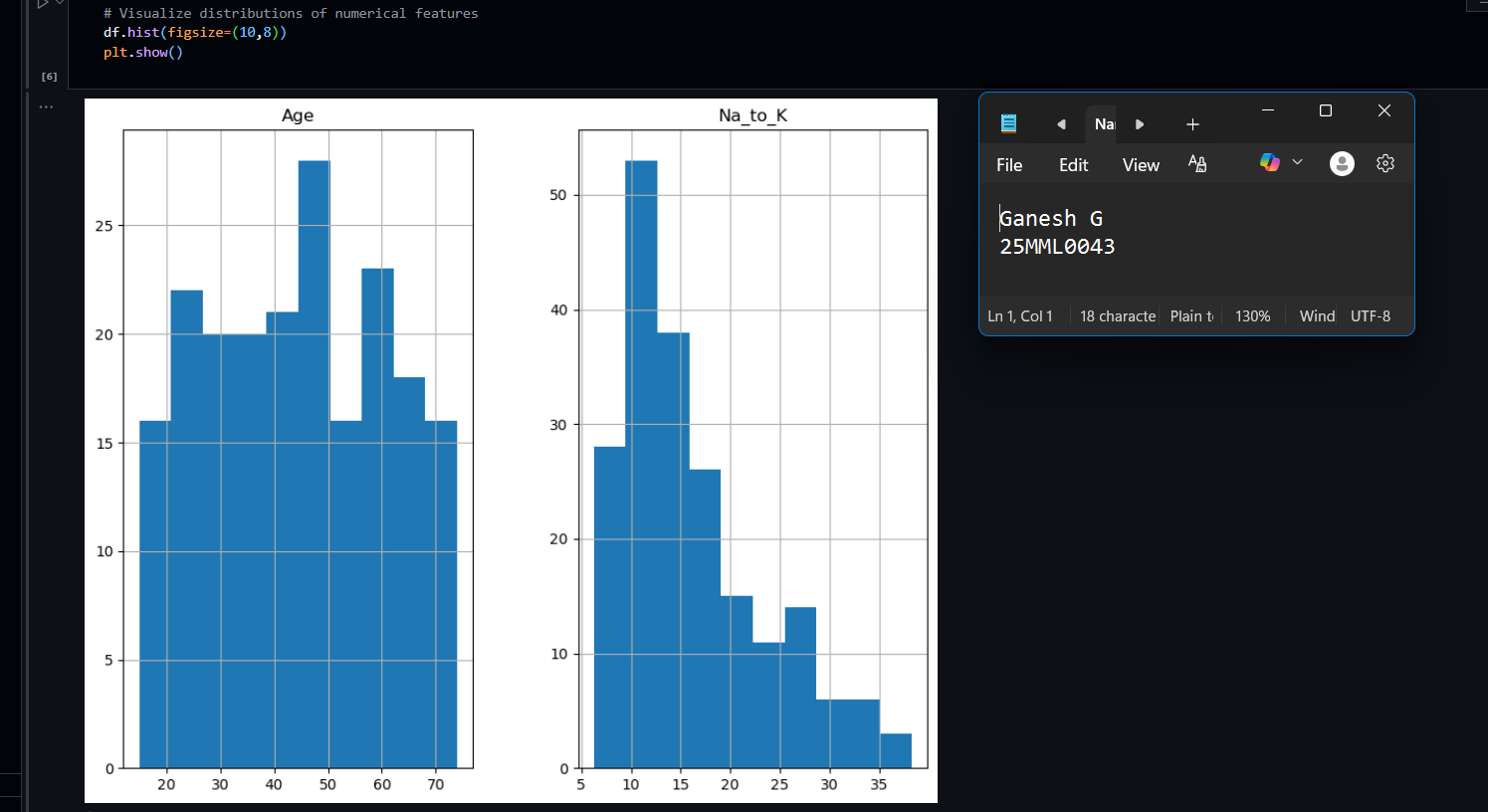
**Implementation and results:**

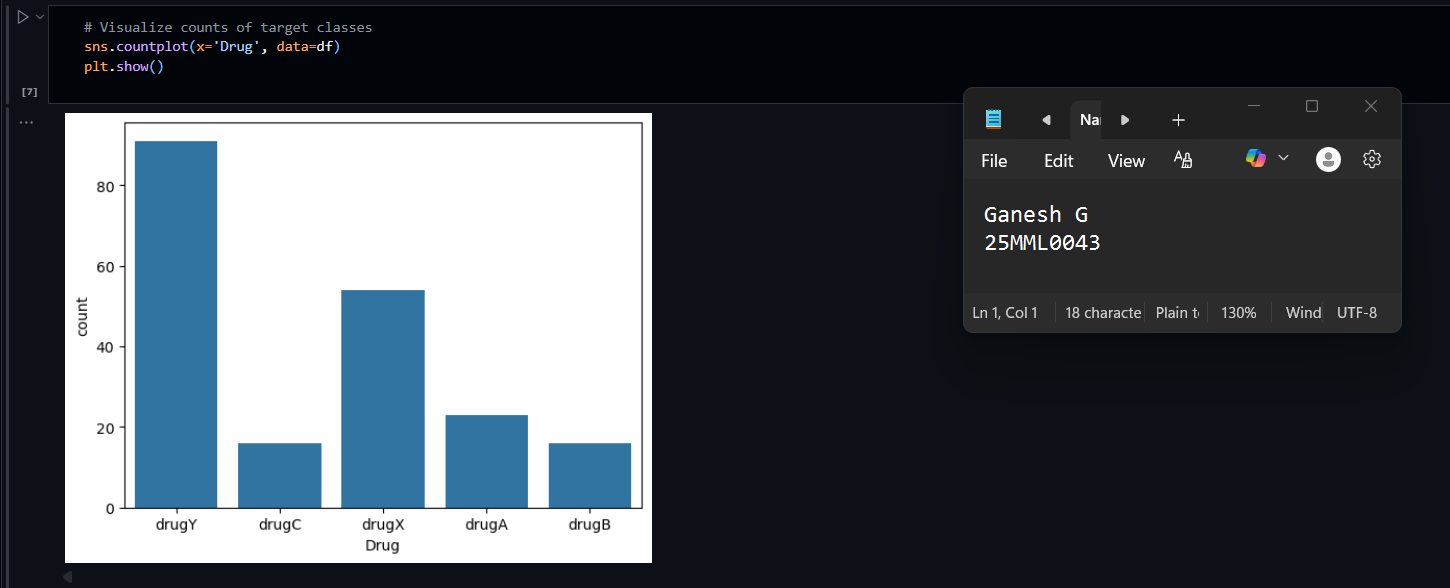
**Step1: Data Set Overview and preprocessing**

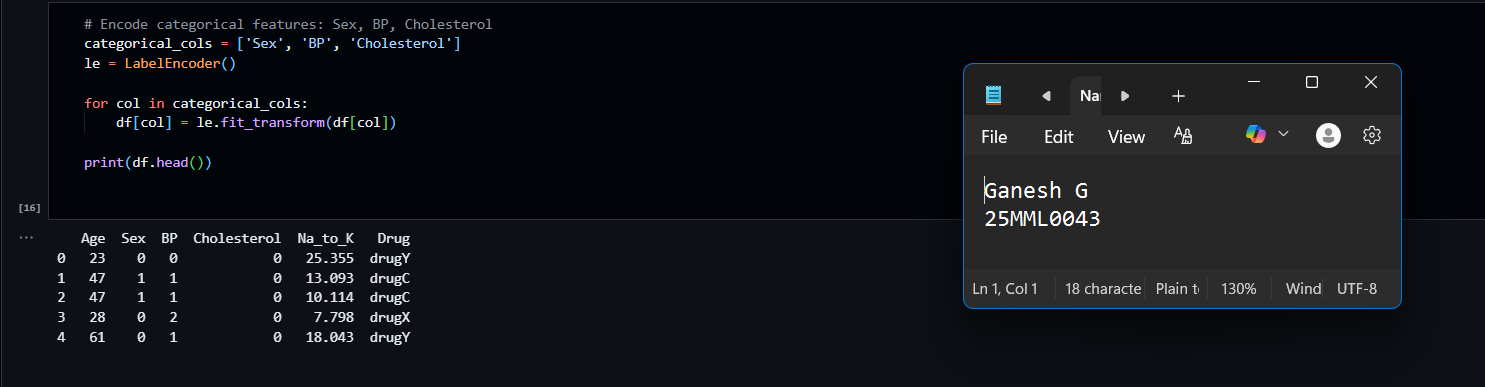
****

****

****

****

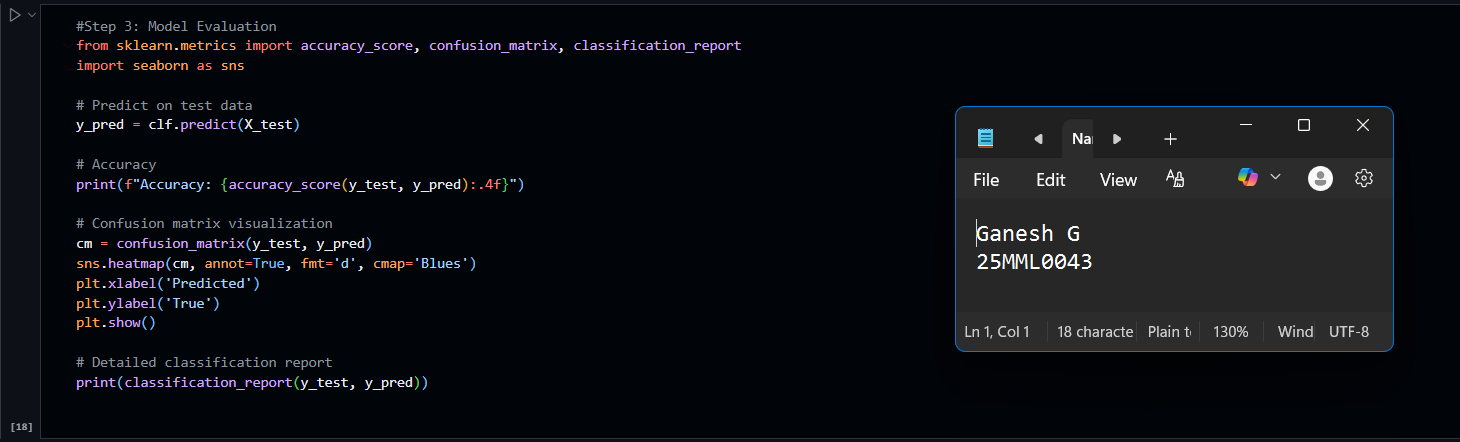
****

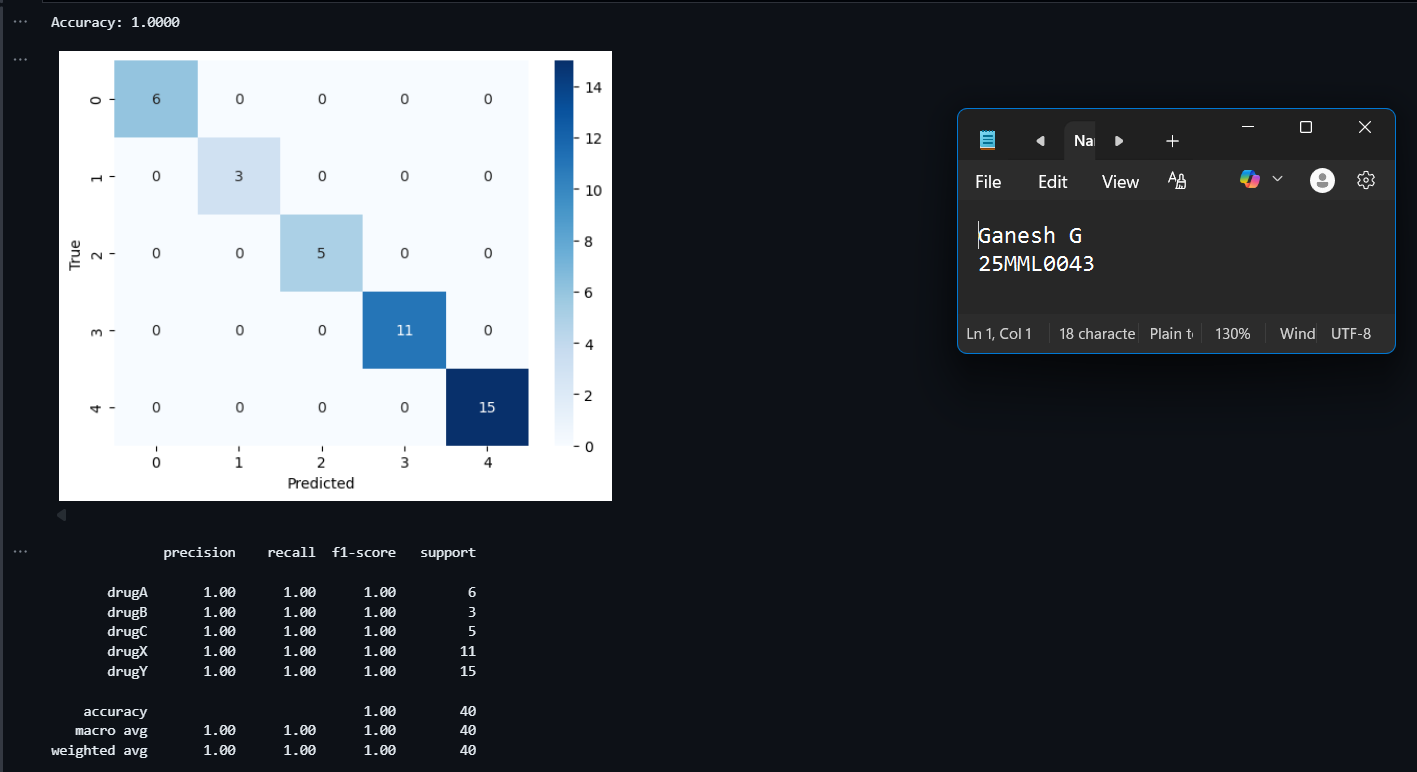
****

**Step 2: Decision Tree Model Building**

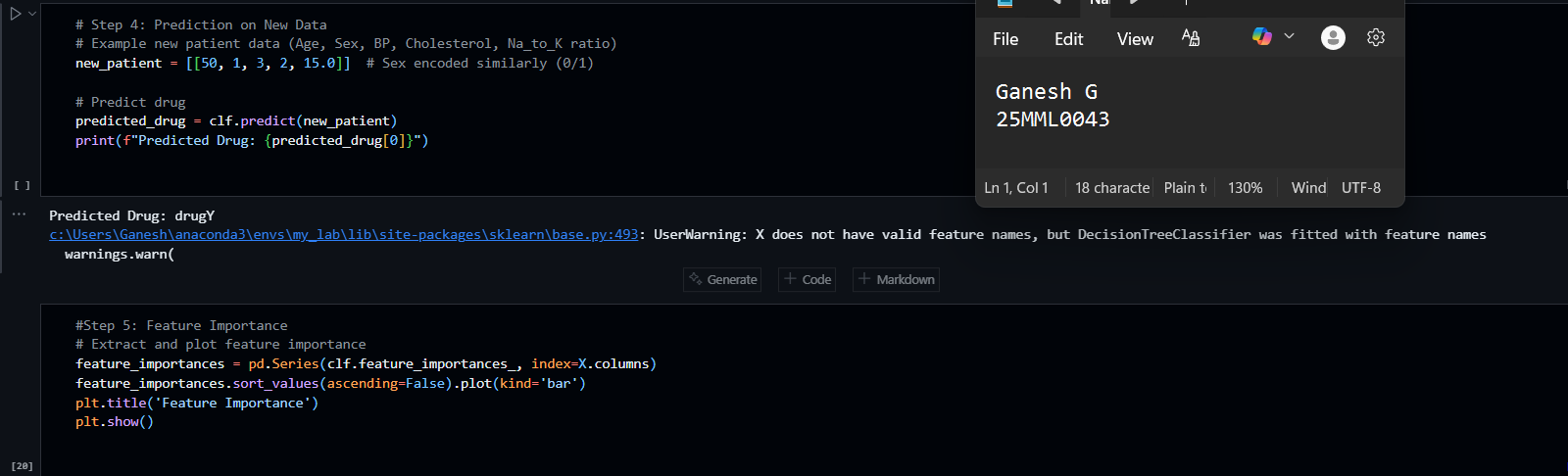
****

**Step 3: Model Evaluation**

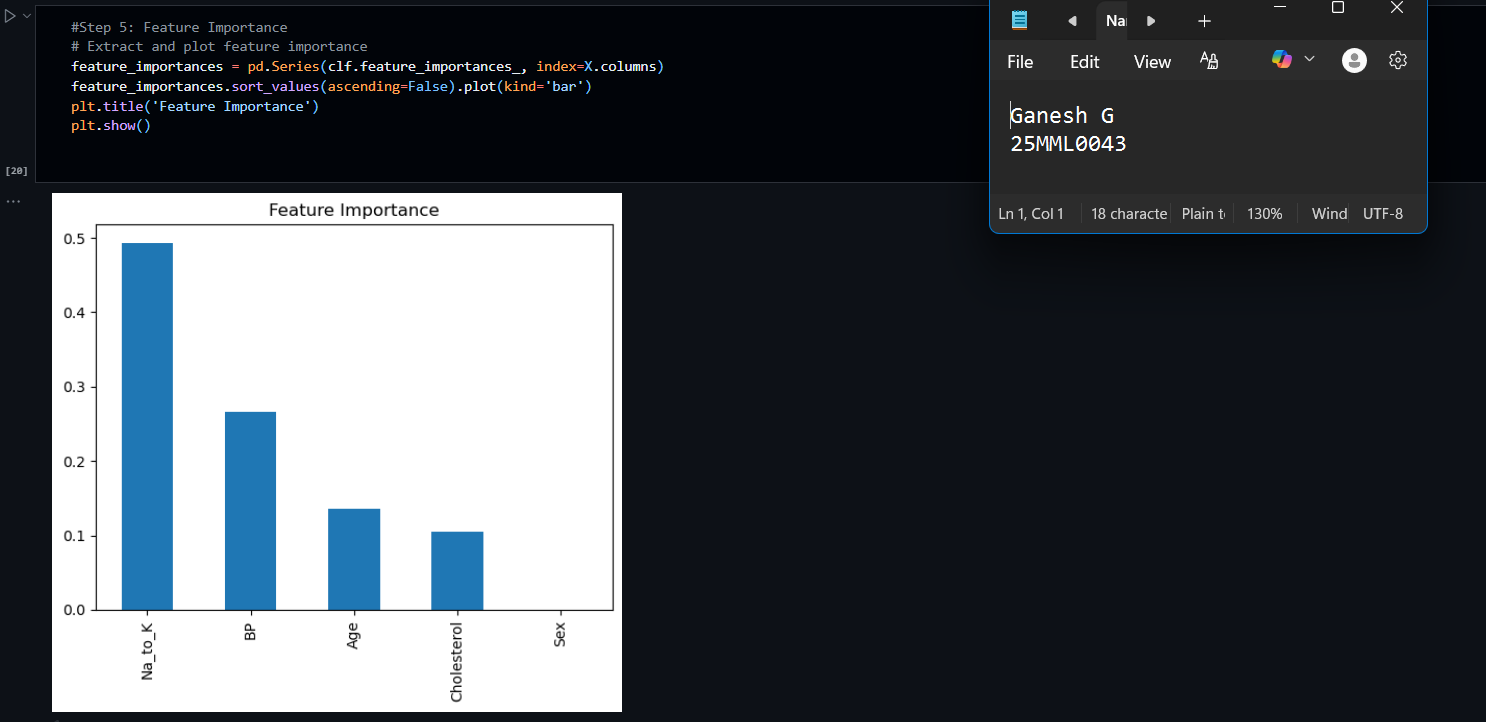
****

****

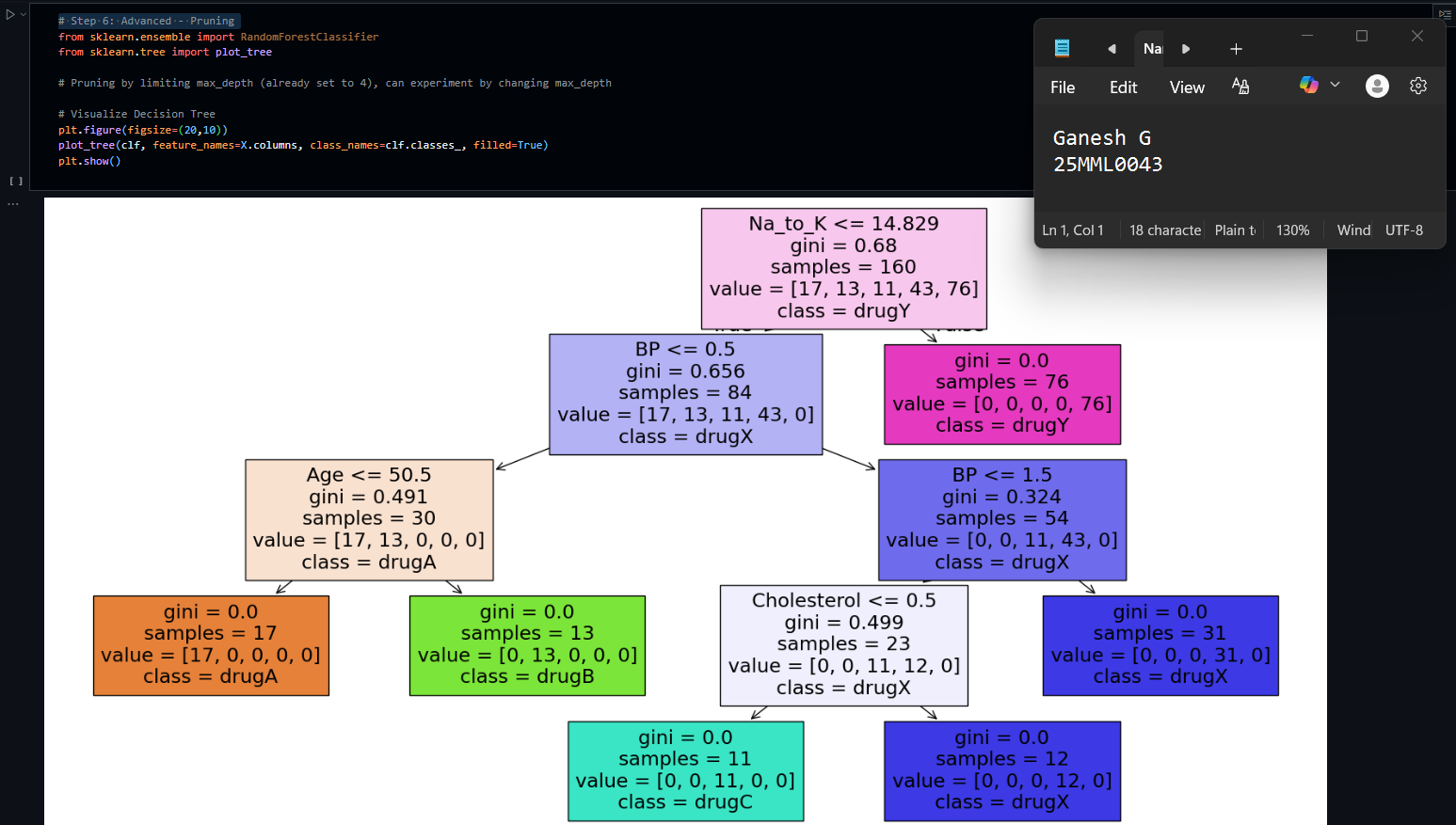
**Step 4: Prediction on New Data**

****

**Step 5: Feature Importance**

****

**Step 6: Advanced - Pruning**

****

1. **Random Forest**

**AIM:**

The aim for analyzing a college student placement dataset typically involves predicting whether a student will be placed based on various features like academic performance, skills, internships, work experience, and other factors. Common goals include:

* Predicting the likelihood of a student getting placed using classification algorithms.
* Identifying the important factors influencing placement success.
* Evaluating model performance with accuracy, precision, recall, and F1-score.
* Helping students and institutions improve preparation and strategies for campus recruitment.

**Introduction:**

The primary objective of this lab is to develop a model that predicts whether a student will be placed in campus recruitment based on various academic and non-academic attributes. Campus placements are a vital milestone for students and institutions as they reflect the outcome of academic efforts and influence the reputation of educational institutes. By analyzing historical placement data using machine learning classification algorithms like Decision Tree, Logistic Regression, or Naive Bayes, this lab aims to forecast placement likelihood for current students. Predictive modeling helps the placement cell identify potential candidates early and focus on improving their skills to maximize placement chances. This system also provides insights into key factors affecting placement success, enabling data-driven decisions to enhance student career outcomes and institutional placement performance. The lab involves data pre-processing, training machine learning models, and evaluating their accuracy for reliable placement prediction.

**Real World Application:**

**1. Healthcare**

* Disease prediction: Used to predict whether a patient has a disease (like diabetes, cancer, or heart disease) based on lab test values, medical history, and lifestyle.
* Example: Classifying breast cancer tumors as *benign* or *malignant* (Random Forest performs well on medical datasets).

**2. Finance**

* **Credit scoring & loan approval**: Banks use Random Forests to assess whether a person is a good or risky borrower by analyzing income, spending habits, and credit history.
* **Fraud detection**: Identifying unusual transactions that could be fraud.

**3. E-commerce & Retail**

* **Recommendation systems**: Predict what products a customer might be interested in, based on past purchases and browsing behavior.
* **Customer churn prediction**: Telecom or subscription-based businesses predict which customers are likely to leave, so they can offer retention plans.

**4. Cybersecurity**

* Intrusion detection systems: Random Forests detect unusual patterns in network traffic that may signal hacking or malware attacks.

**5. Text & Sentiment Analysis**

* Spam email detection: Filtering spam vs. non-spam emails.
* Sentiment classification: Analyzing whether a product review is positive or negative.

**ALGORITHM:**

**Input**: Training dataset D with n samples and m features.

**For each tree (repeat K times):**

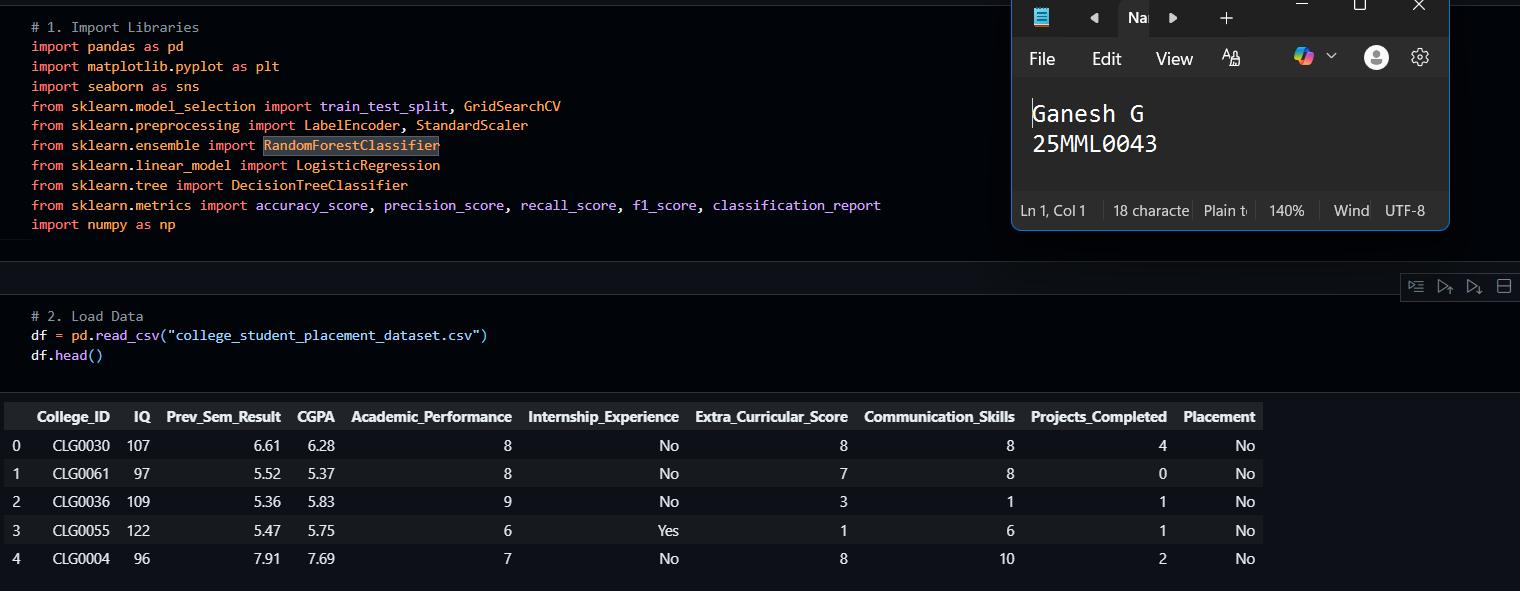
* **Bootstrap sampling**: Randomly select a subset of the data (with replacement) from D.
* **Feature selection**: At each split in the tree, choose a random subset of features (instead of all features).
* **Build a decision tree** using the chosen data and feature subset (often grown fully, without pruning).

**Aggregate results**:

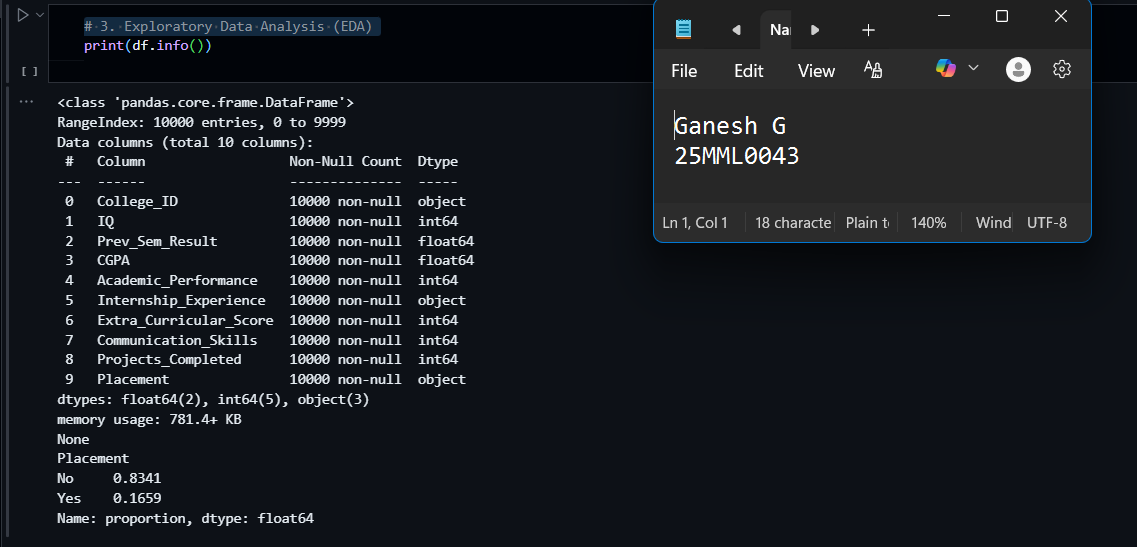
* **Classification**: Each tree votes for a class. The class with the most votes is the final prediction.
* **Regression**: Take the average of all tree predictions

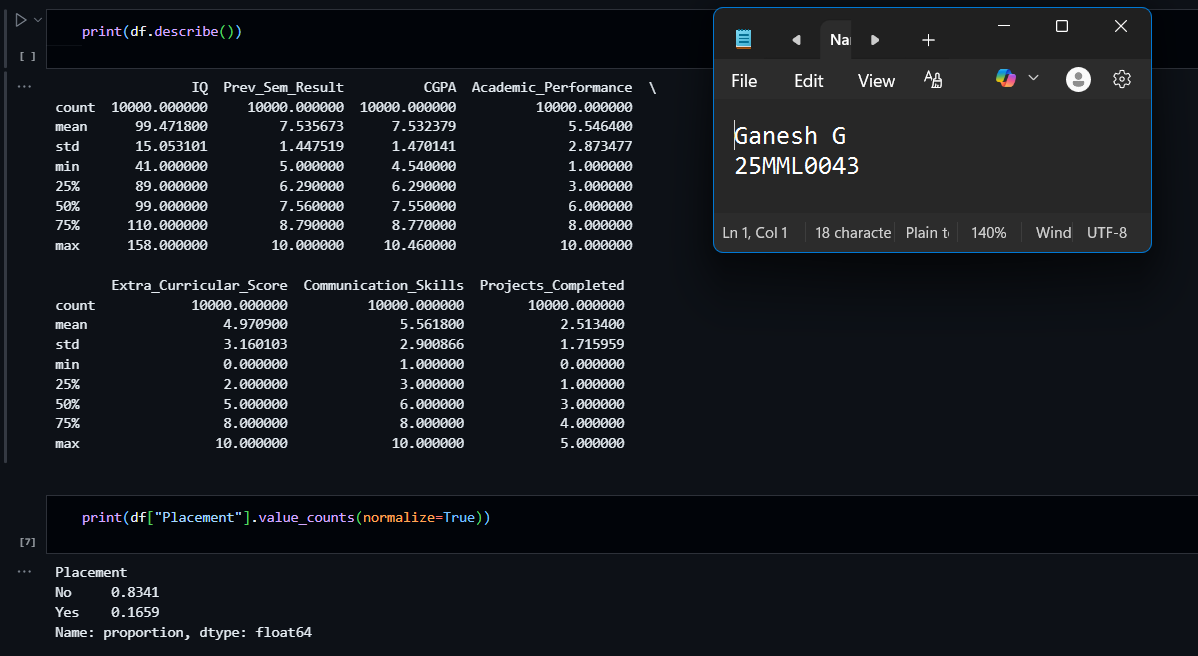
**Implementation and results:**

**Import libraries and Load dataset**

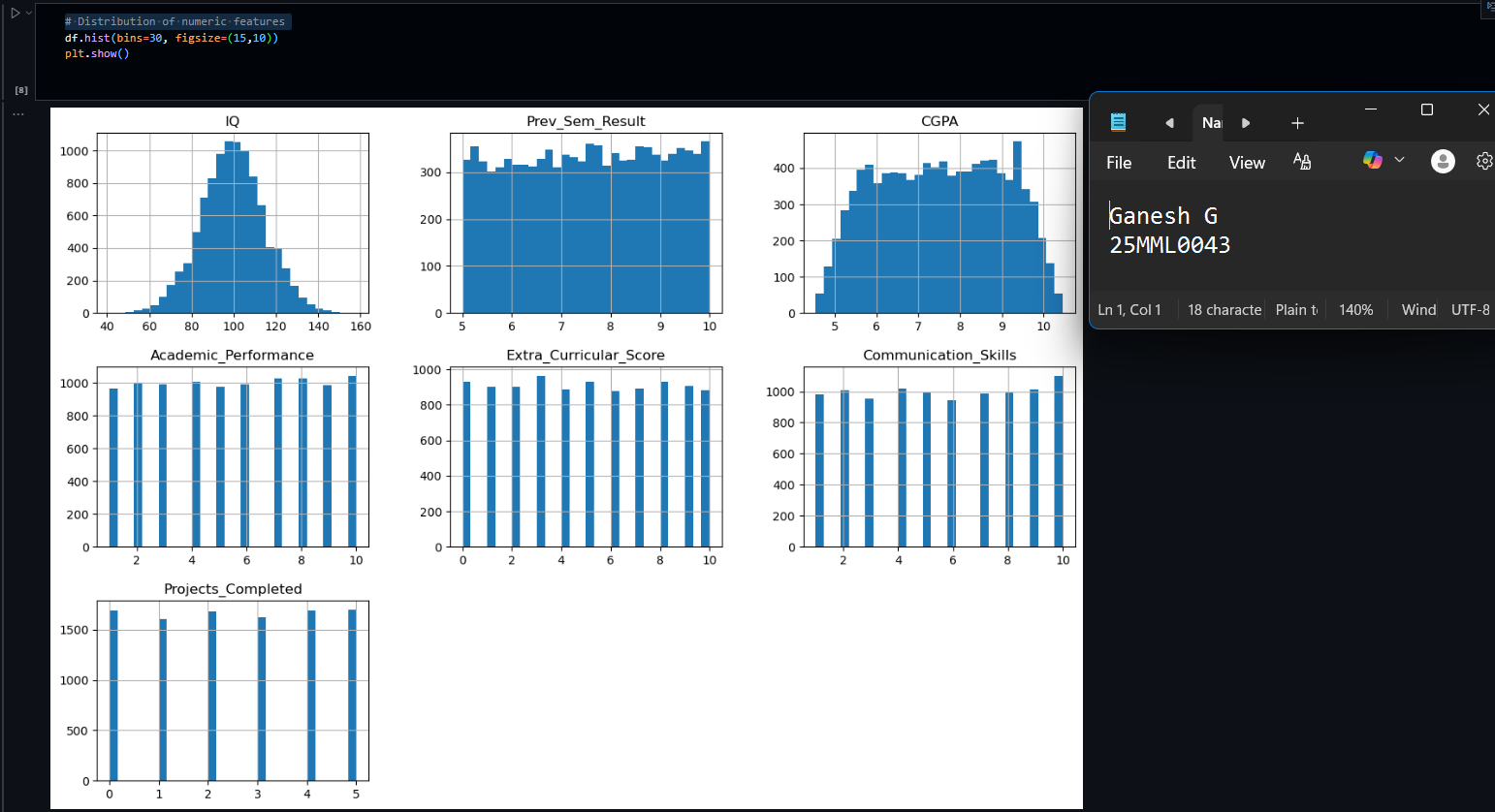
****

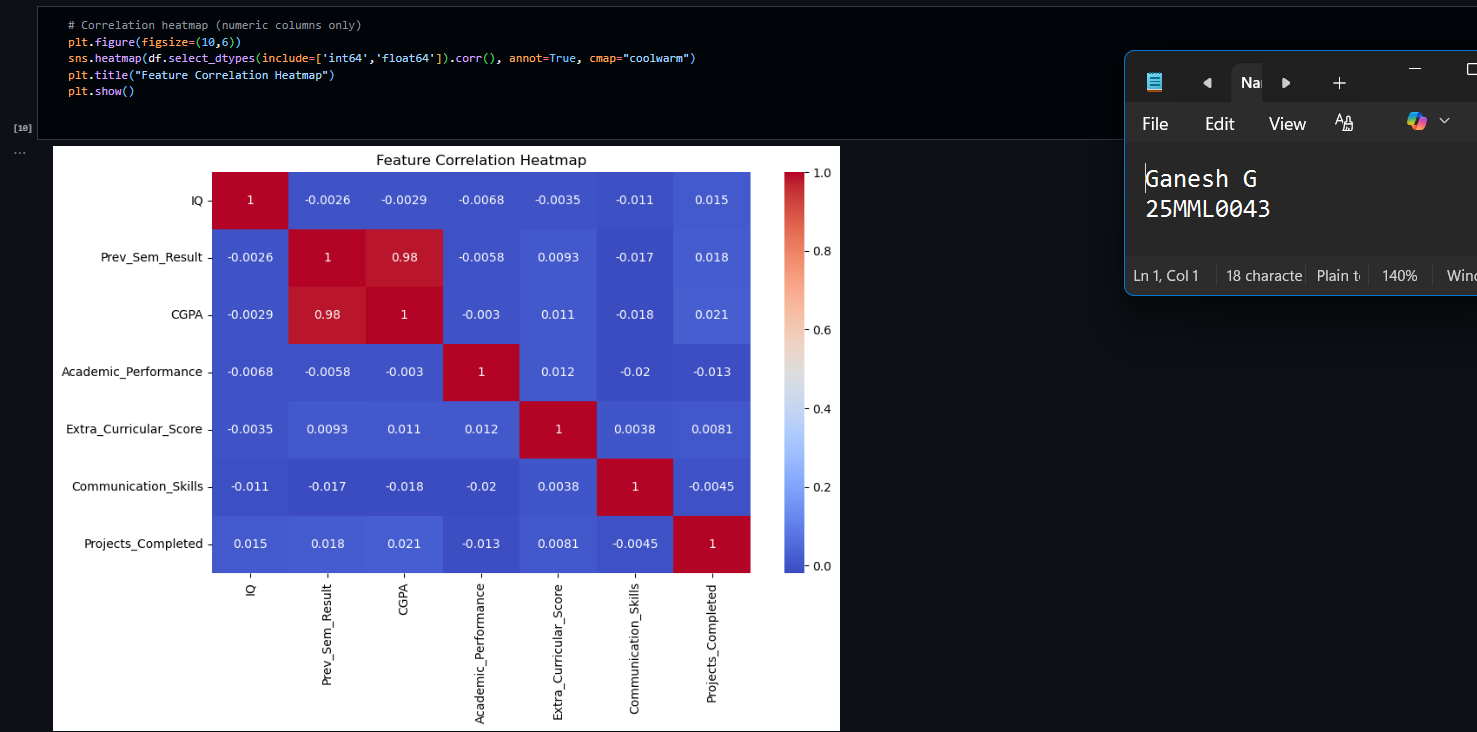
**Exploratory Data Analysis (EDA)**

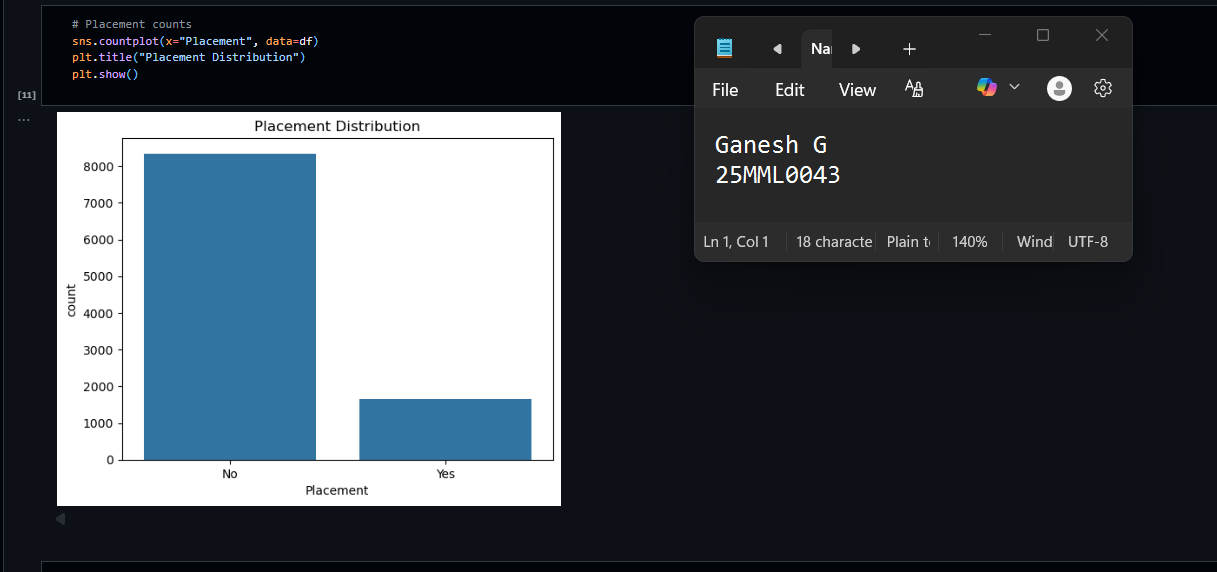
****

****

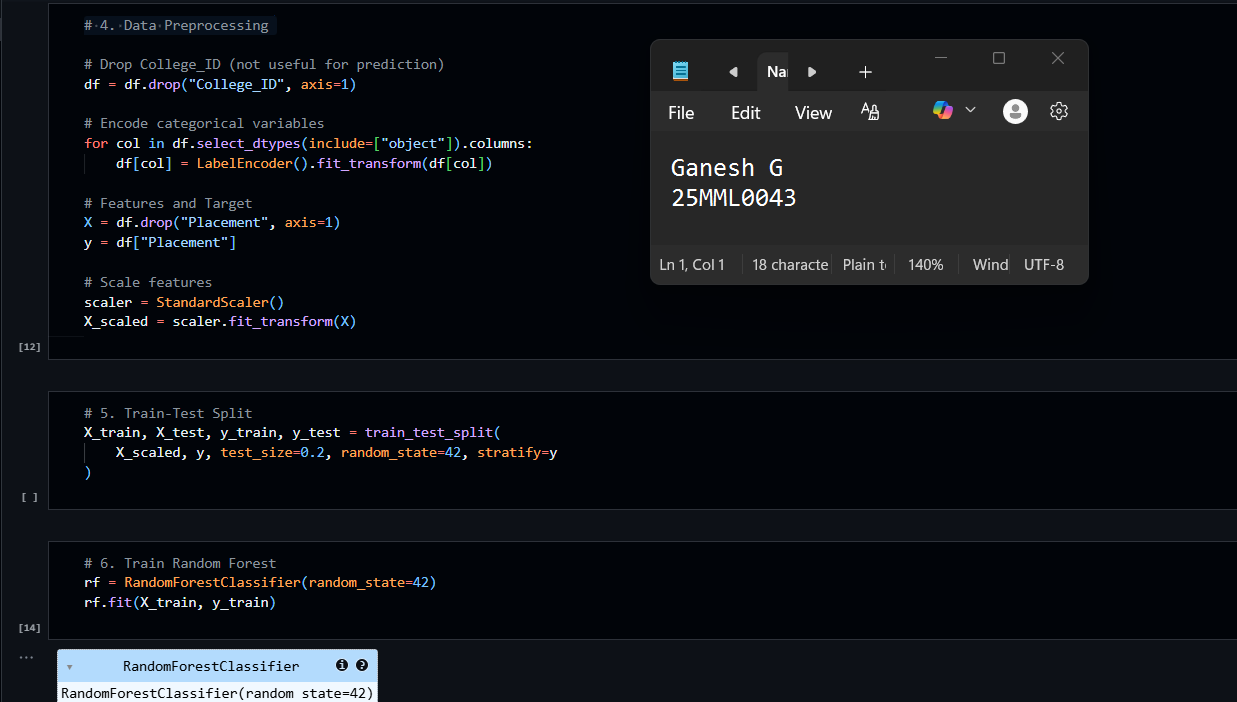
**Distribution of numeric features**

****

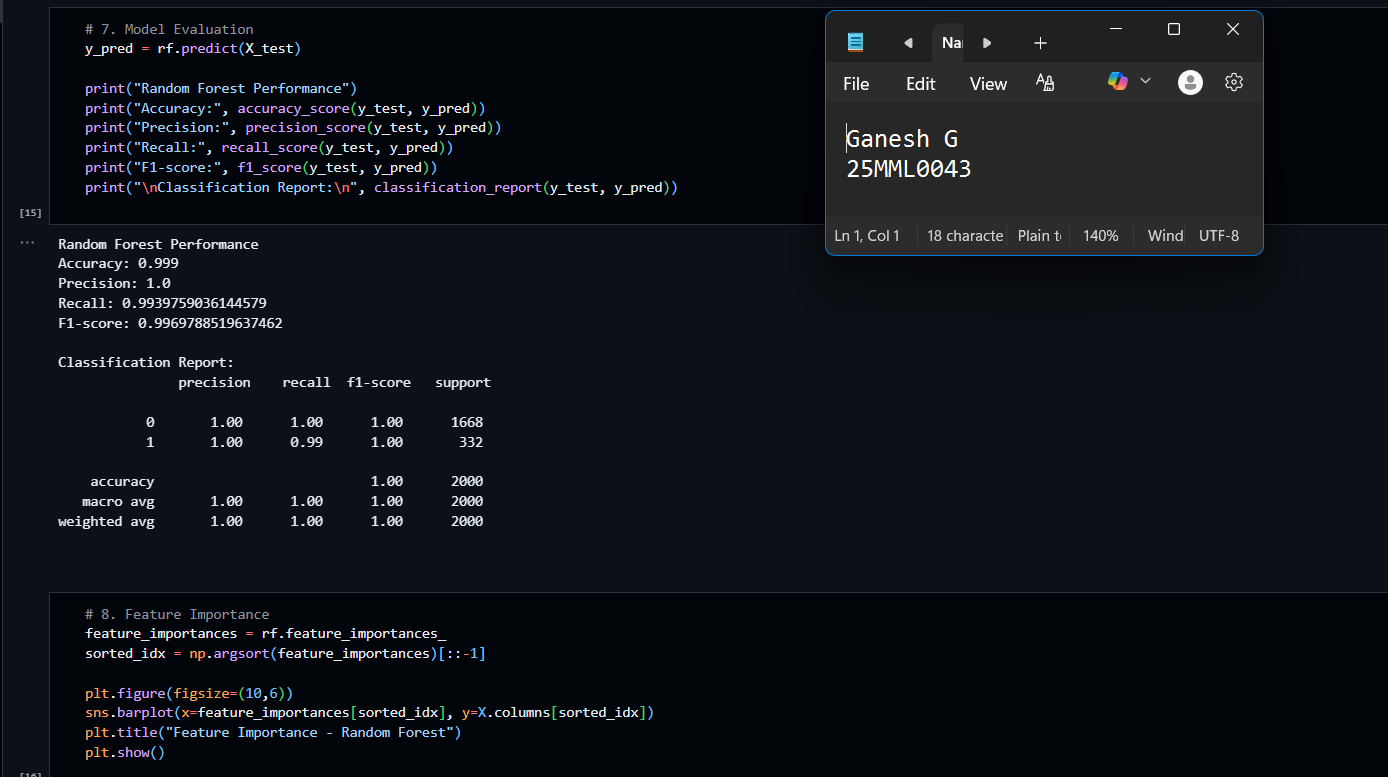
****

****

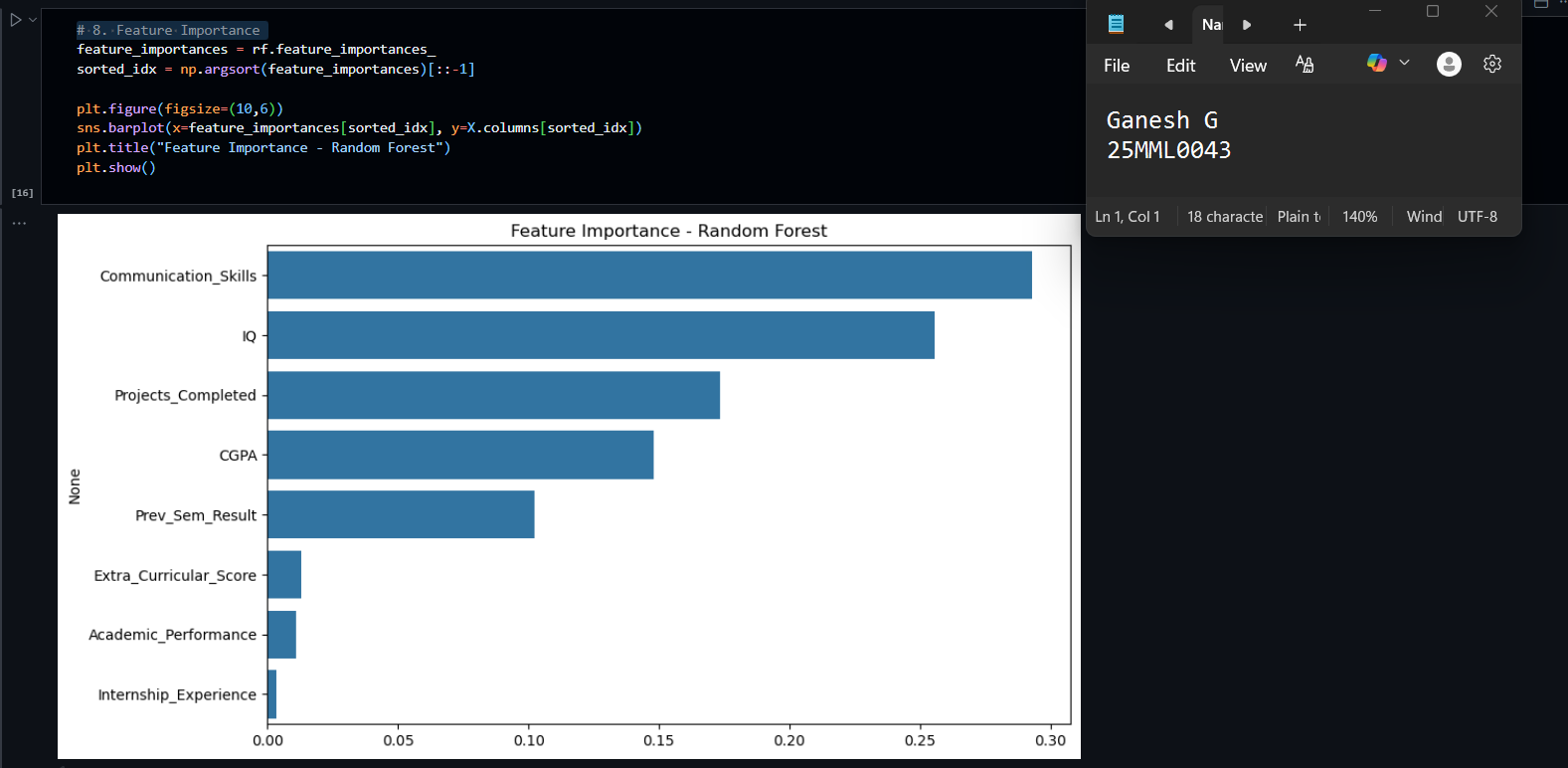
**Data Preprocessing**

****

**Model Evaluation**

****

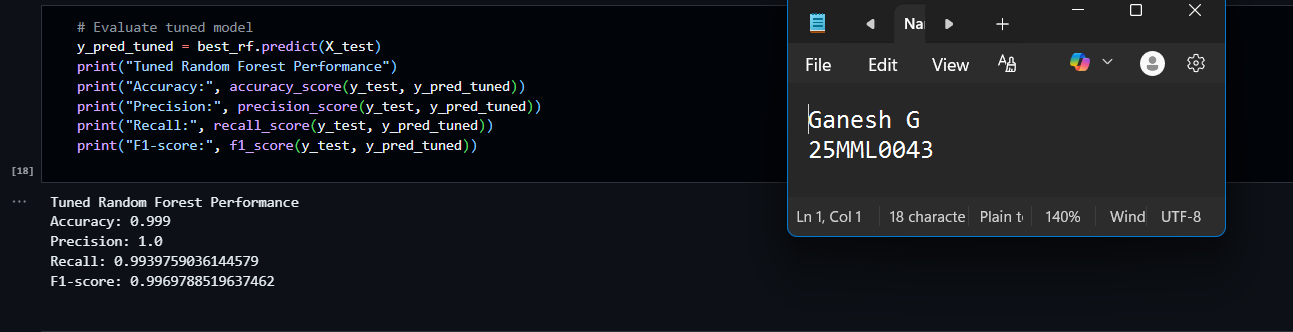
**Feature Importance**

****

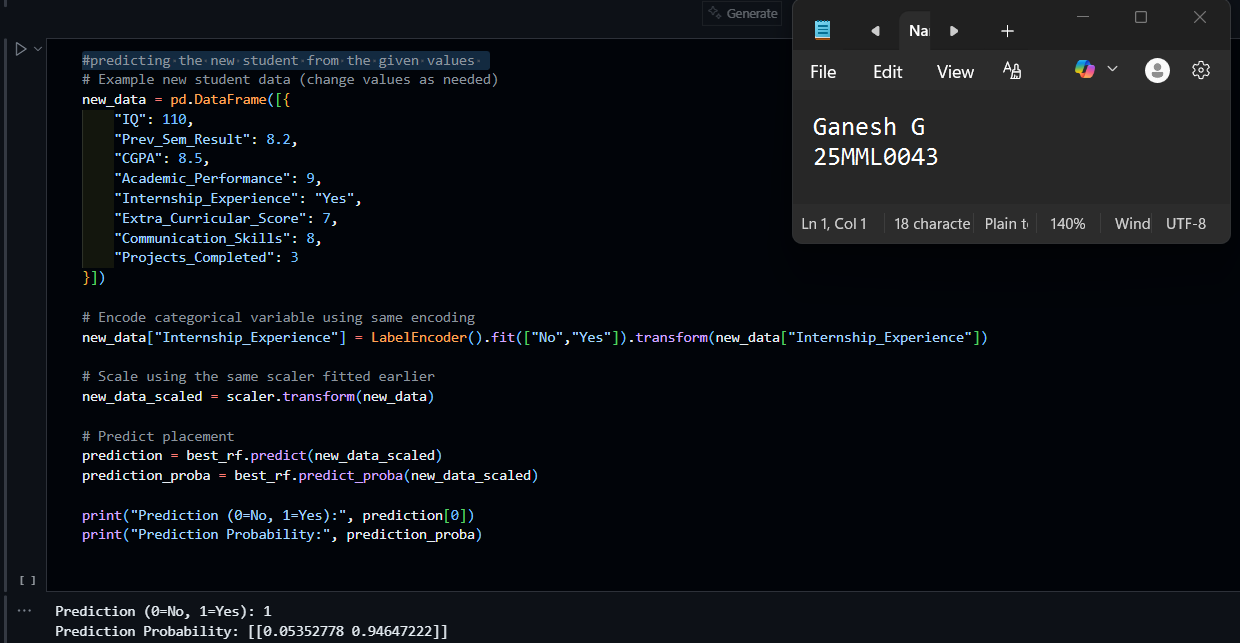
**Hyperparameter Tuning**

****

**Evaluation of Tuned model**

****

**predicting the new student from the given values**

****

**Models Comparison**

****