Artificial Intelligence and Machine Learning Laboratory

Reg no:24MAI0104
Name: GOWTHAM.J

TITLE : (DATA PREPROCESSING)

1.Data preprocessing is a crucial step before building a machine learning model, as it ensures the data is clean, well-structured, and ready for analysis.

DESCRIPTION :

Data preprocessing is a fundamental step in the machine learning pipeline that involves transforming raw data into a clean, structured format suitable for building models. The quality and relevance of the data you feed into your model directly impact its performance, making data preprocessing crucial for achieving accurate and reliable results.

Key Steps in Data Preprocessing

1. Data Cleaning:

- **Handling Missing Values**: Real-world data often has missing values, which can lead to incorrect model predictions. Common techniques include:
  - **Removing Missing Data**: Dropping rows or columns with missing values if they are not significant.
  - **Imputing Missing Data**: Replacing missing values with a mean, median, mode, or using advanced methods like forward/backward filling.
- **Removing Duplicates**: Ensuring that no duplicate rows exist in the dataset, as they can bias the model.

2. Data Transformation:

- **Normalization/Standardization**: Scaling features to a similar range or distribution is crucial when the model assumes that all input features have the same scale (e.g., in algorithms like KNN, SVM).
  - **Normalization**: Scaling data to a range of [0, 1].
  - **Standardization**: Rescaling data to have a mean of 0 and a standard deviation of 1.
- **Encoding Categorical Variables**: Converting categorical data into numerical format, which is essential for algorithms that require numerical input.
  - **Label Encoding**: Assigning a unique integer to each category.
  - **One-Hot Encoding**: Creating binary columns for each category.

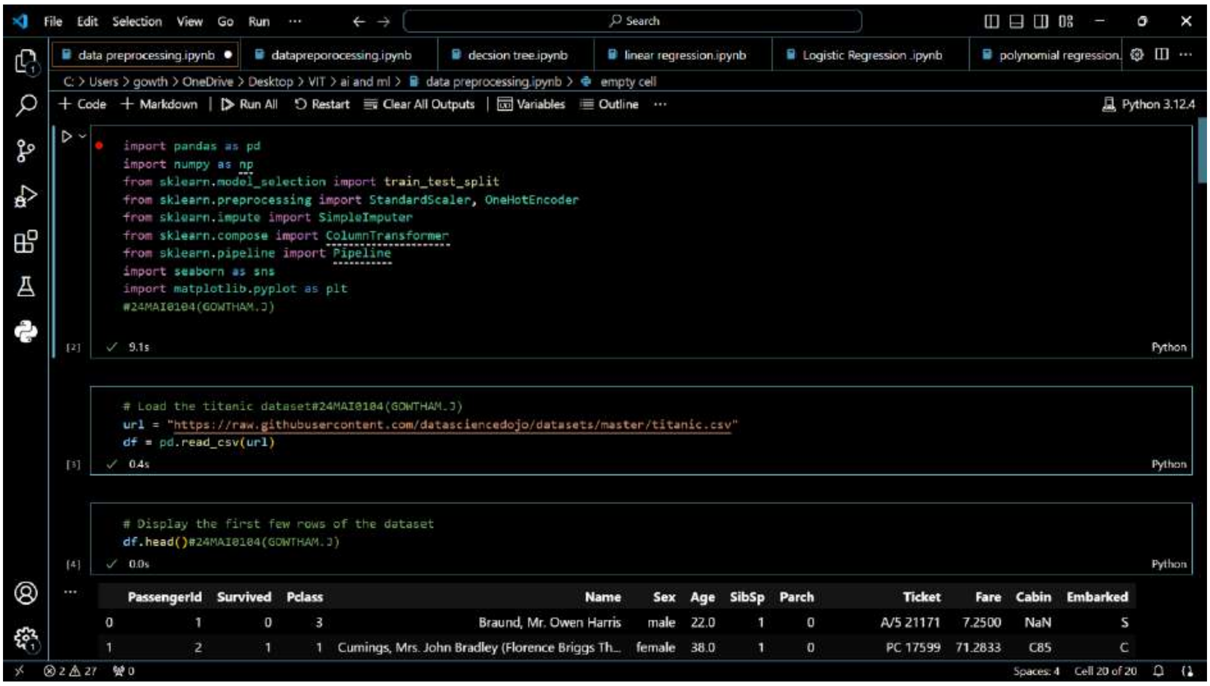
3. Feature Engineering:

- **Creating New Features**: Generating new features that might help the model better capture the underlying patterns in the data. For example, creating interaction terms, polynomial features, or aggregating data over time periods.
- **Feature Selection**: Choosing the most relevant features to improve model performance and reduce overfitting. Techniques include:
    - **Correlation Analysis**: Identifying and removing highly correlated features.
    - **Feature Importance**: Using models like Random Forest to identify the most important features.

4. Splitting the Dataset:

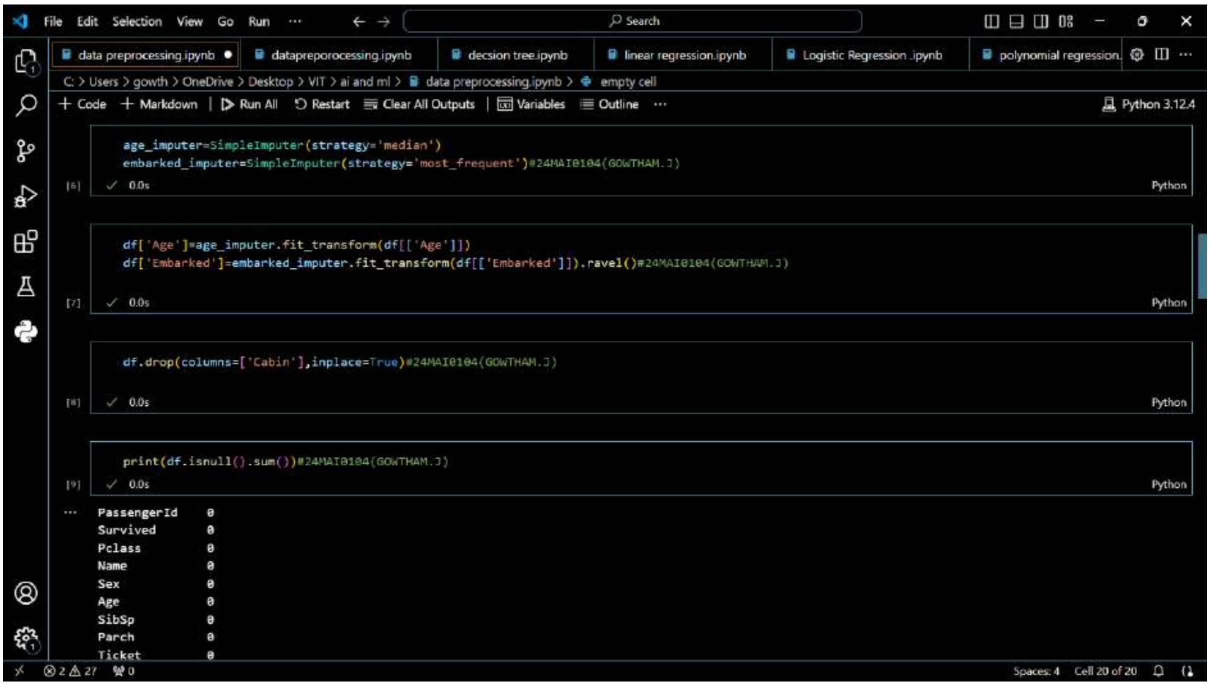
- **Training and Testing Split**: Dividing the dataset into training and testing sets to evaluate the model's performance. This step helps in understanding how the model generalizes to unseen data.
- **Cross-Validation**: A technique to ensure that the model's performance is robust and not overly dependent on a single train-test split.

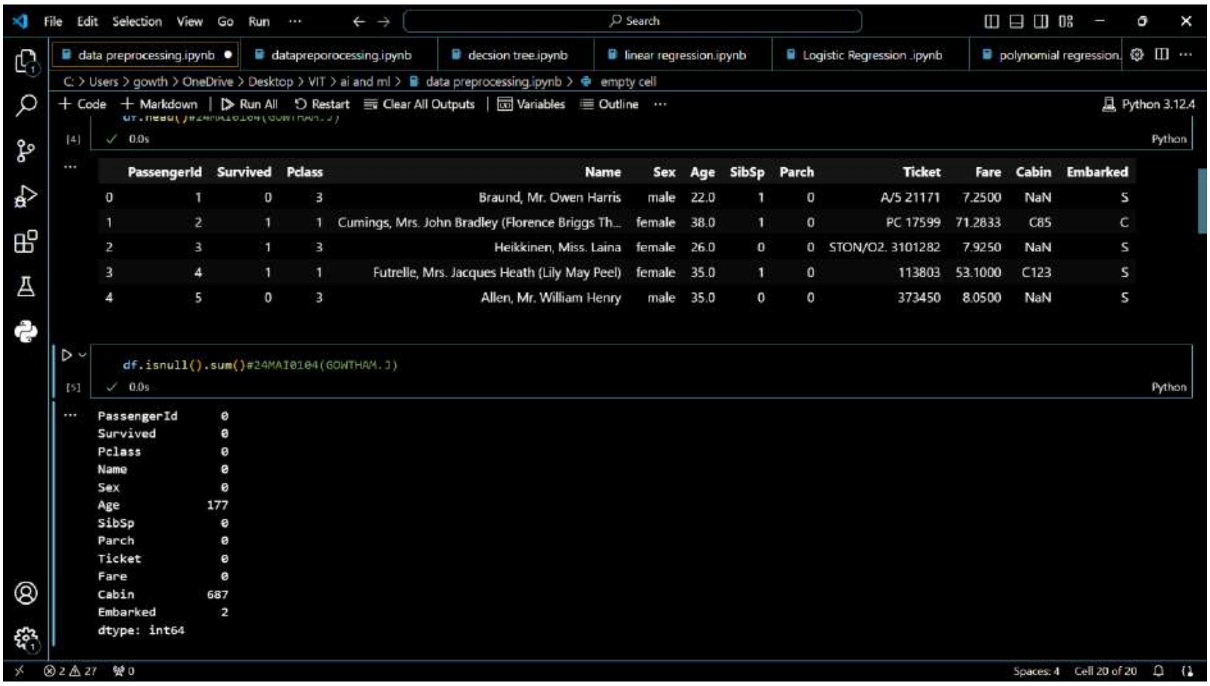
SOURCE CODE AND RESULT :



Artificial Intelligence and Machine Learning Laboratory

Reg no:24MAI0104
Name: GOWTHAM.J





Reg no:24MAI0104

Name: GOWTHAM.J


```python
encoder=OneHotEncoder (sparse_output=False, drop='first')
encoded_sex_embarked=encoder.fit_transform (df [['Sex', 'Embarked']])
encoded_df = pd.DataFrame (encoded_sex_embarked,
columns=encoder.get_feature_names_out(['Sex', 'Embarked']))
```

[12] ✓ 0.0s

```python
df.drop(columns=['Sex','Embarked'],inplace=True)
df=pd.concat([df,encoded_df],axis=1)
```

[13] ✓ 0.0s

```python
df.head()
```

[14] ✓ 0.0s

	PassengerId	Survived	Pclass	Name	Age	SibSp	Parch	Ticket	Fare	Sex_male	Embarked_Q	Embarked_S
0	1	0	3	Braund, Mr. Owen Harris	22.0	1	0	A/5 21171	7.2500	1.0	0.0	1.0
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	38.0	1	0	PC 17599	71.2833	0.0	0.0	0.0
2	3	1	3	Heikkinen, Miss. Laina	26.0	0	0	STON/O2. 3101282	7.9250	0.0	0.0	1.0
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	35.0	1	0	113803	53.1000	0.0	0.0	1.0
4	5	0	3	Allen, Mr. William Henry	35.0	0	0	373450	8.0500	1.0	0.0	1.0

```
SibSp       0
Parch       0
Ticket      0
Fare        0
Embarked    0
dtype: int64
```

```python
df.head()#24MAI0104(GOWTHAM.J)
```

[10] ✓ 0.0s

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Embarked
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	S
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	C
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	S
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	S
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	S

```python
df.shape
```

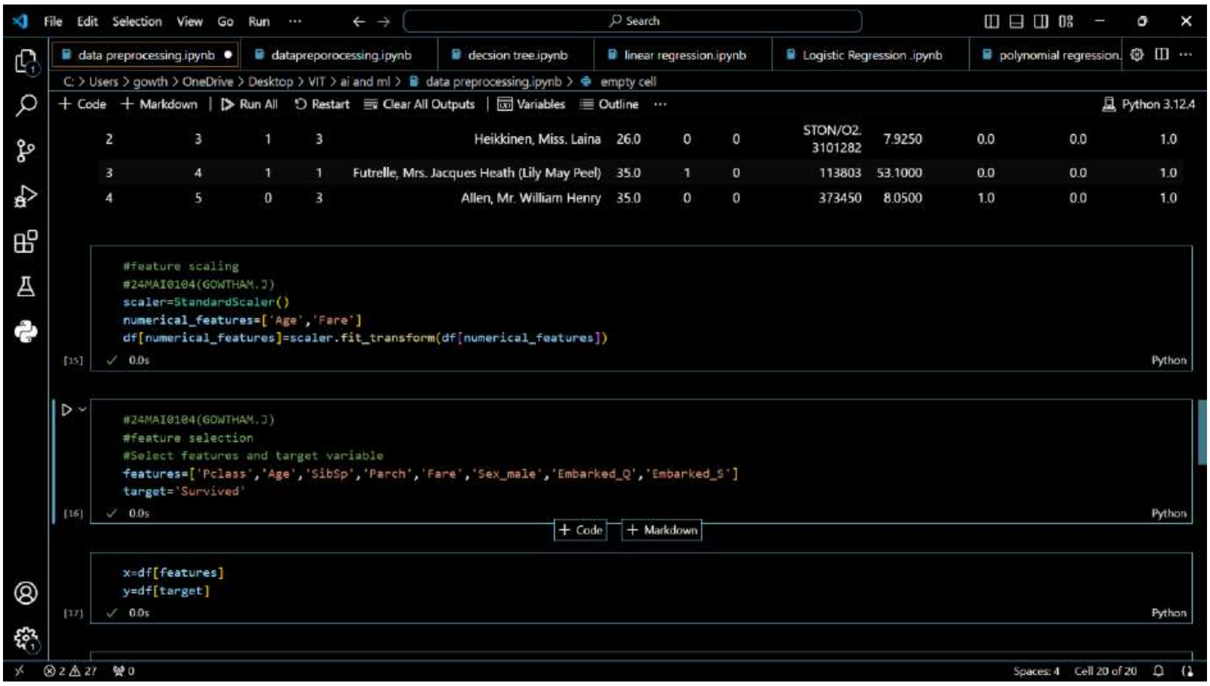
[11] ✓ 0.0s

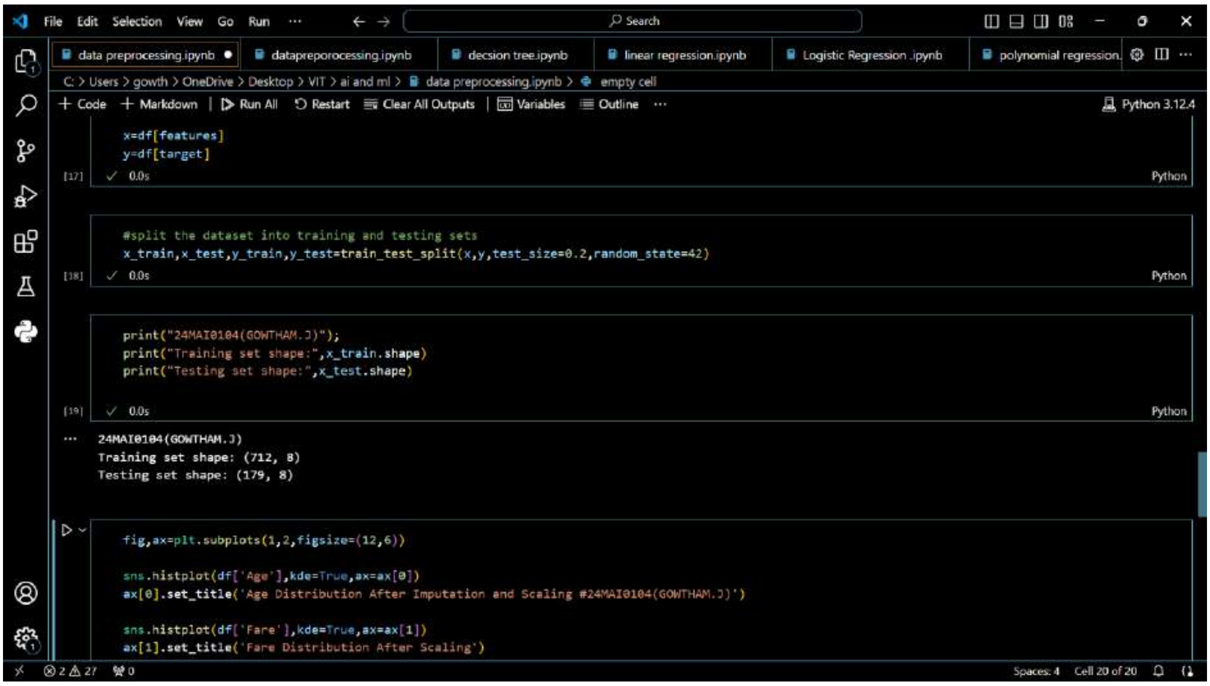
```
(891, 11)
```

```python
encoder=OneHotEncoder (sparse_output=False, drop='first')
```

Artificial Intelligence and Machine Learning Laboratory

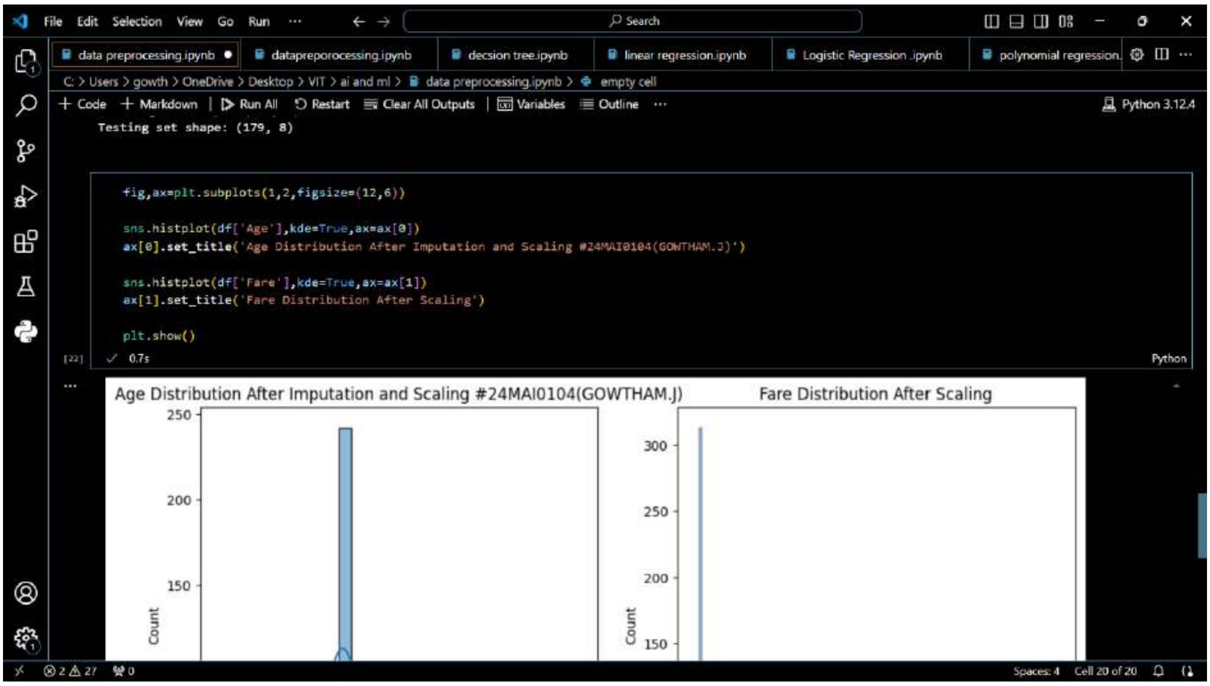
Reg no:24MAI0104
Name: GOWTHAM.J

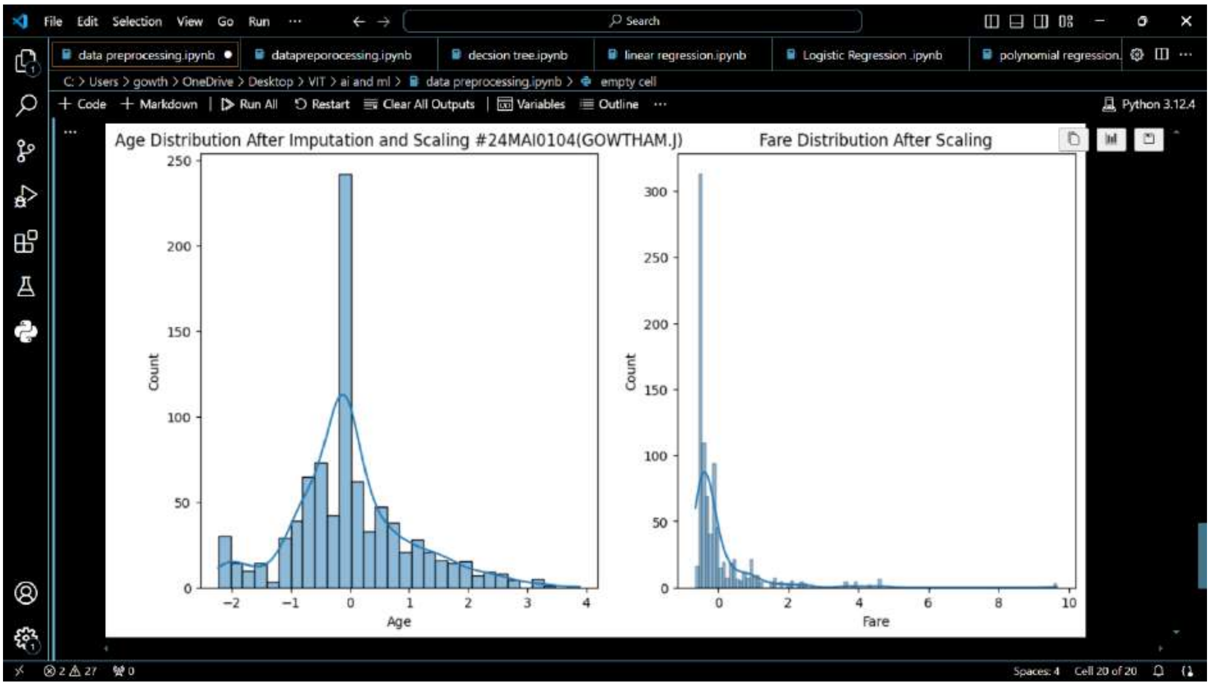




Artificial Intelligence and Machine Learning Laboratory

Reg no:24MAI0104
Name: GOWTHAM.J





Reg no:24MAI0104
Name: GOWTHAM.J

CONCULSION :

Data preprocessing is essential for preparing data for machine learning:

1. Clean data by handling missing values, duplicates, and outliers.
2. Transform data by normalizing, encoding categorical variables, and engineering features.
3. Integrate data sources and reduce dimensionality.
4. Split data for training and testing to ensure model performance.
5. It improves model accuracy and reliability by optimizing data quality and structure.

Artificial Intelligence and Machine Learning Laboratory

Reg no:24MAI0104
Name: GOWTHAM.J

TITLE（LINEAR REGRESSION MODEL）:

2. Predicting gold prices using the **LINEAR REGRESSION MODEL** involves data collection, preprocessing, model training, and evaluation. Below is a detailed example using Python and Scikit-learn to predict gold prices.

DESCRIPTION :

Linear Regression is one of the most fundamental and widely used algorithms in machine learning and statistics. It is a method for modeling the relationship between a dependent variable (often called the target or response variable) and one or more independent variables (also known as features or predictors).

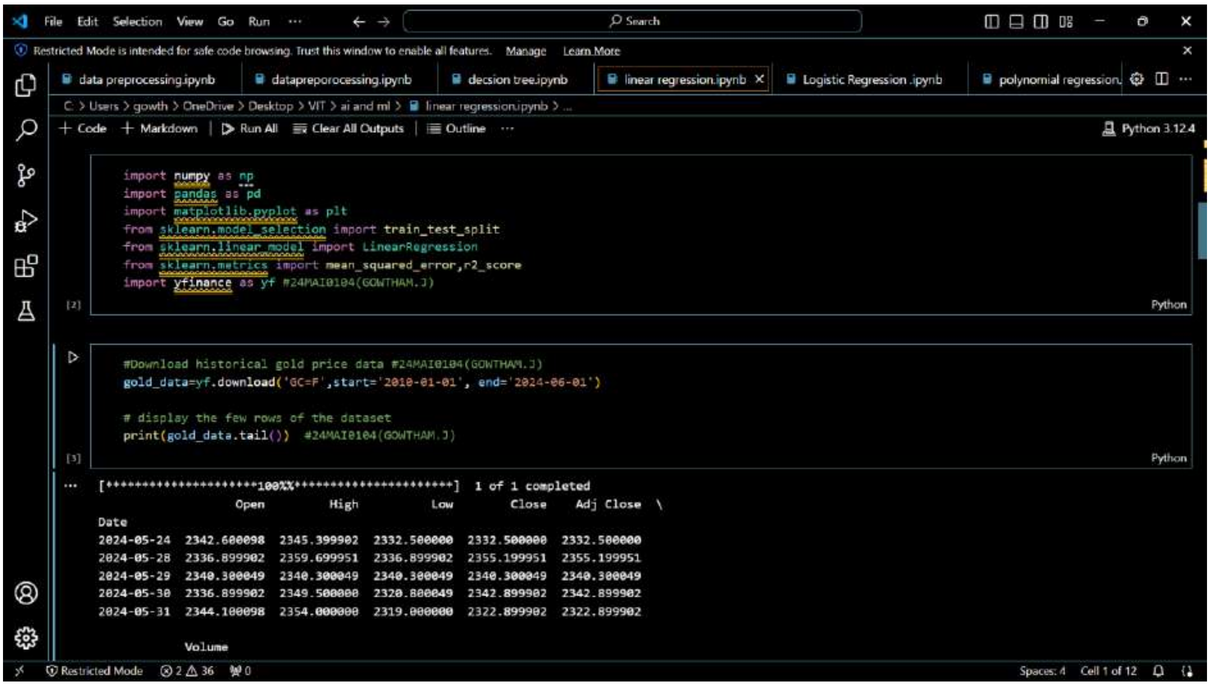
In this project, we aim to predict the future prices of gold using a Linear Regression model. The main objective is to build a model that can effectively learn the relationship between historical gold prices and other related features to forecast the price of gold for the next day or a specified future date.

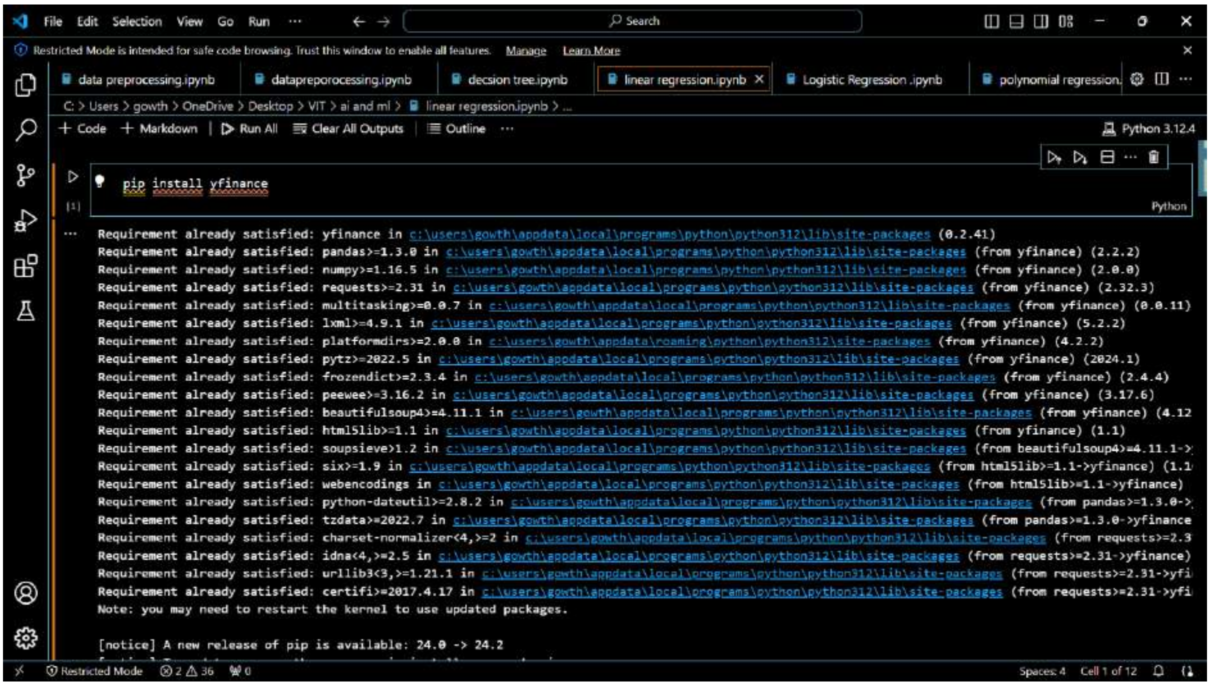
Steps Involved:

1. **Data Collection**: We will begin by gathering historical gold price data, which typically includes daily prices over a specific period. This data can be sourced from financial platforms like Yahoo Finance .
2. **Data Preprocessing**: The collected data may contain missing values, noise, or irrelevant information. We will clean the data by handling missing values, converting date formats, and selecting relevant features for the model.
3. **Feature Selection**: To predict the future price of gold, we will choose features that may influence gold prices. In a simple case, we might use the previous day's price as a predictor. For a more complex model, additional features like moving averages, economic indicators, or commodity prices may be considered.
4. **Splitting the Data**: The dataset will be split into training and testing sets. The training set will be used to train the model, while the testing set will help evaluate its performance.
5. **Model Training**: We will train a Linear Regression model on the training data. The model will learn the relationship between the selected features and the target variable (future gold price).
6. **Model Evaluation**: After training, the model's performance will be evaluated on the test set using metrics like Mean Squared Error (MSE) and R^2 score. These metrics will help us understand how well the model predicts future gold prices.

7. **Prediction and Visualization**: Finally, we will use the trained model to make predictions on unseen data. The actual vs. predicted prices will be visualized to assess the model's accuracy.

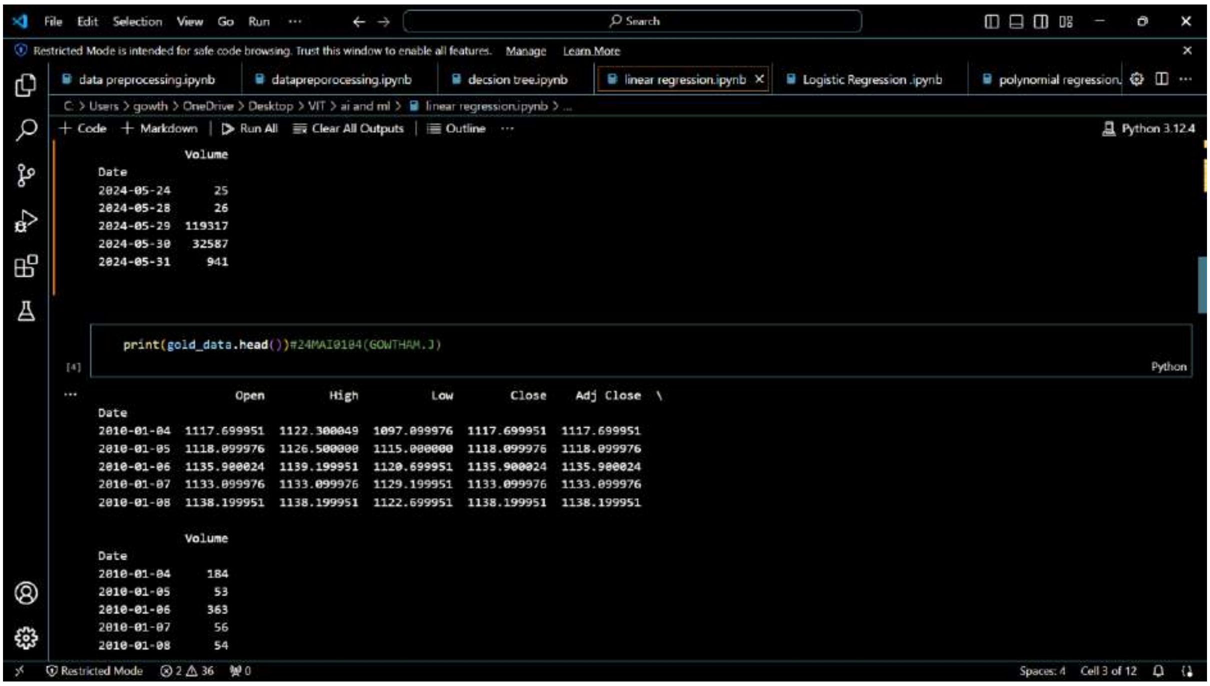
SOURCE CODE AND RESULT :

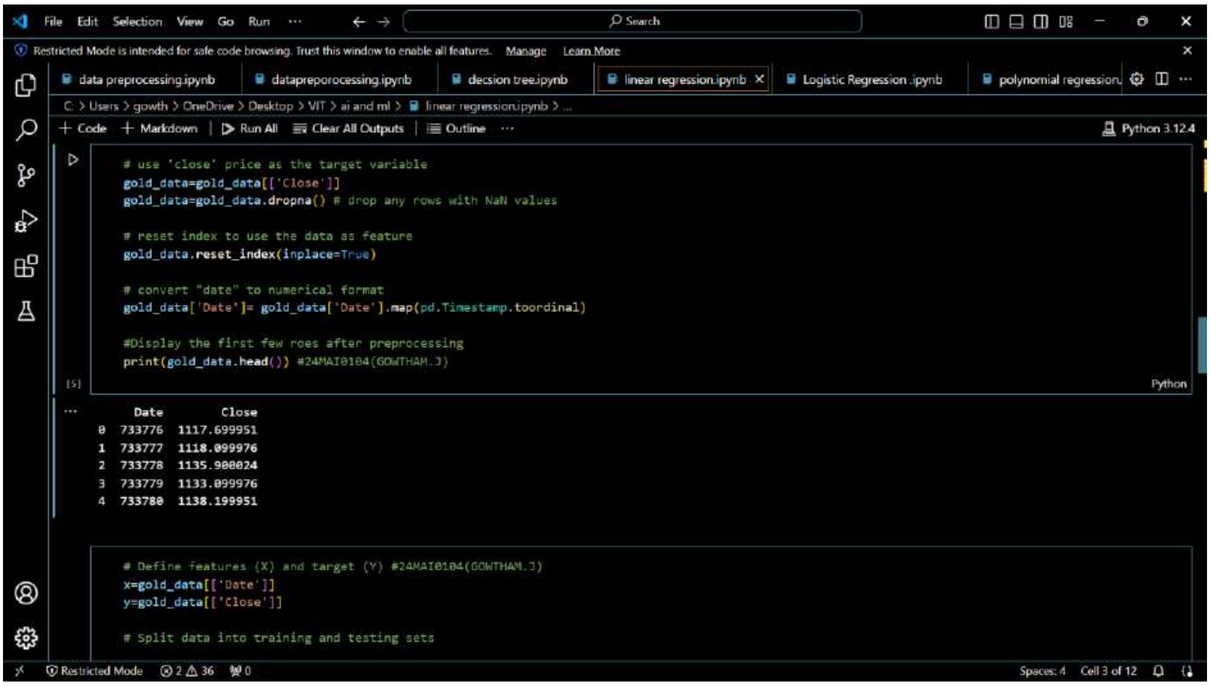




Artificial Intelligence and Machine Learning Laboratory

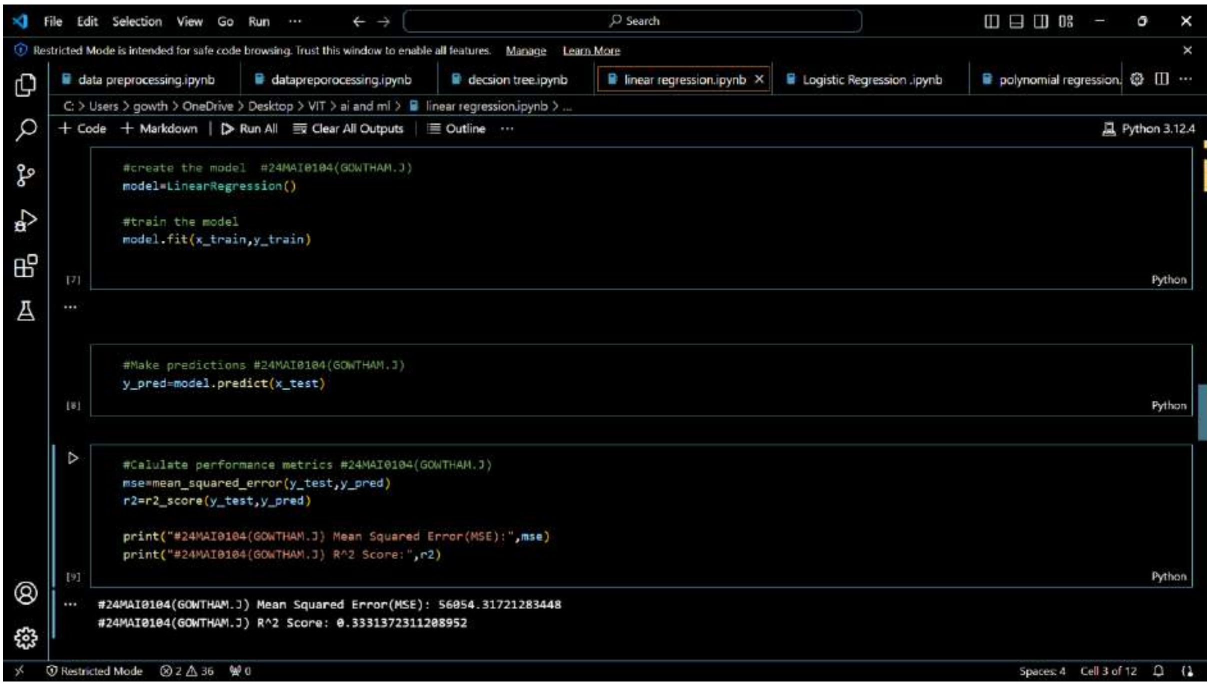
Reg no:24MAI0104
Name: GOWTHAM.J

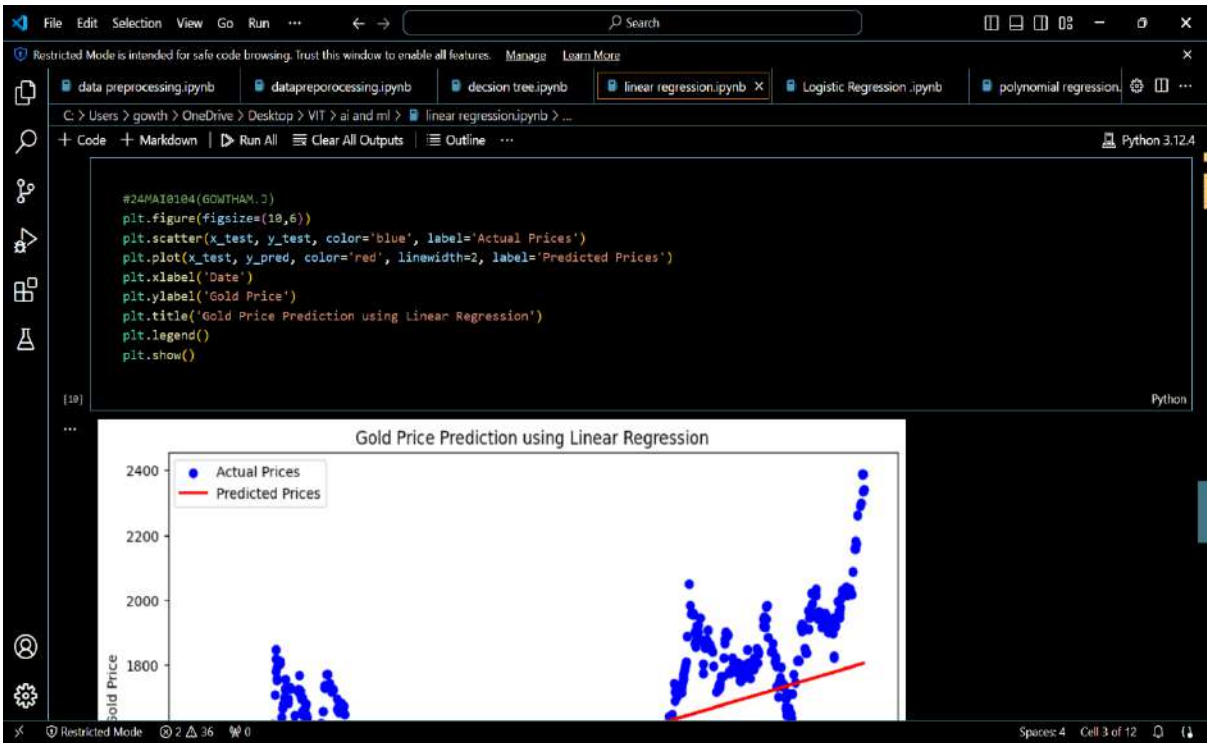




Artificial Intelligence and Machine Learning Laboratory

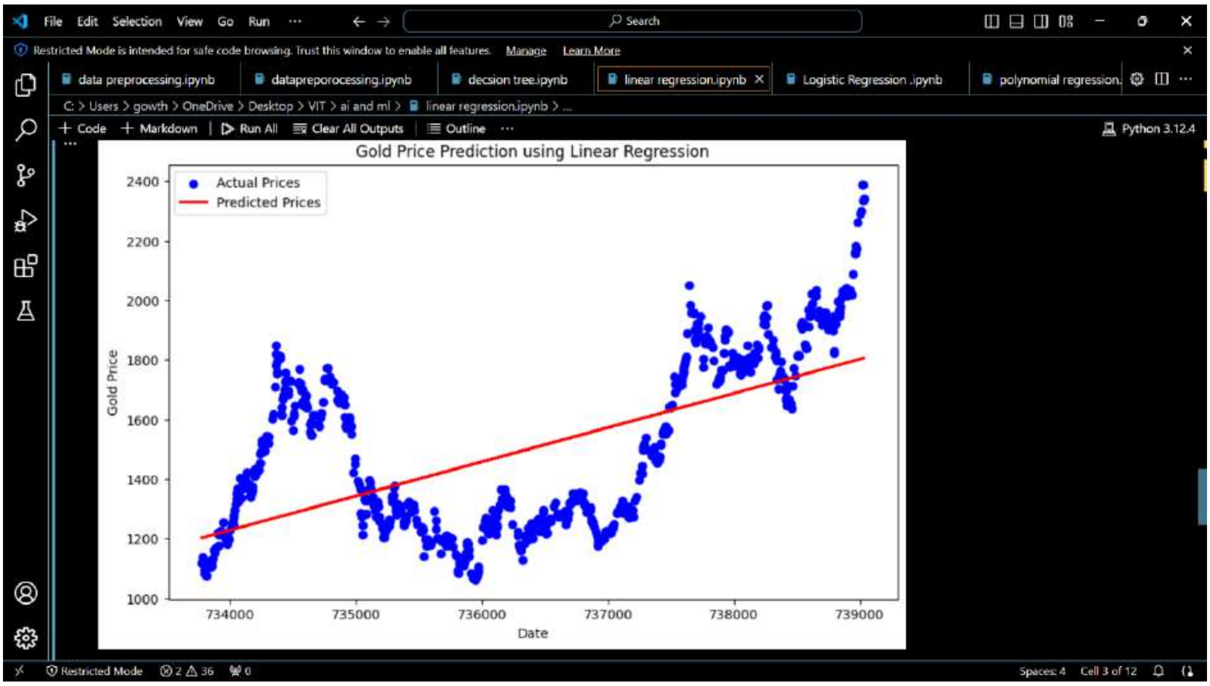
Reg no:24MAI0104
Name: GOWTHAM.J

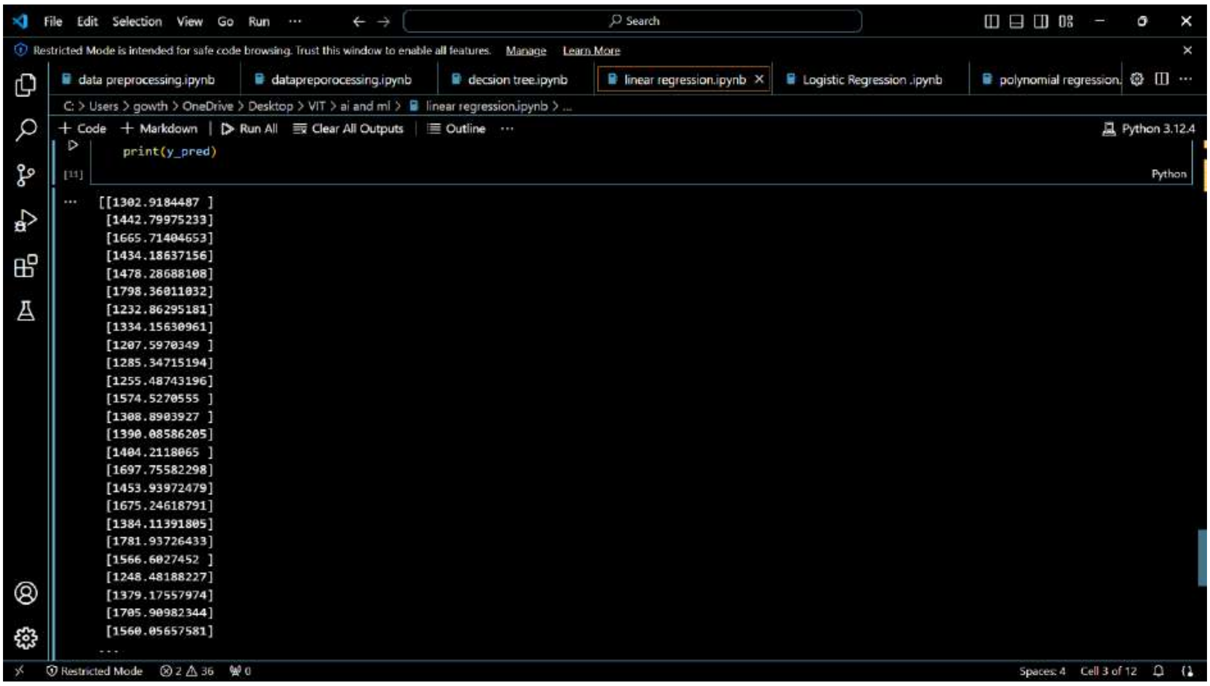


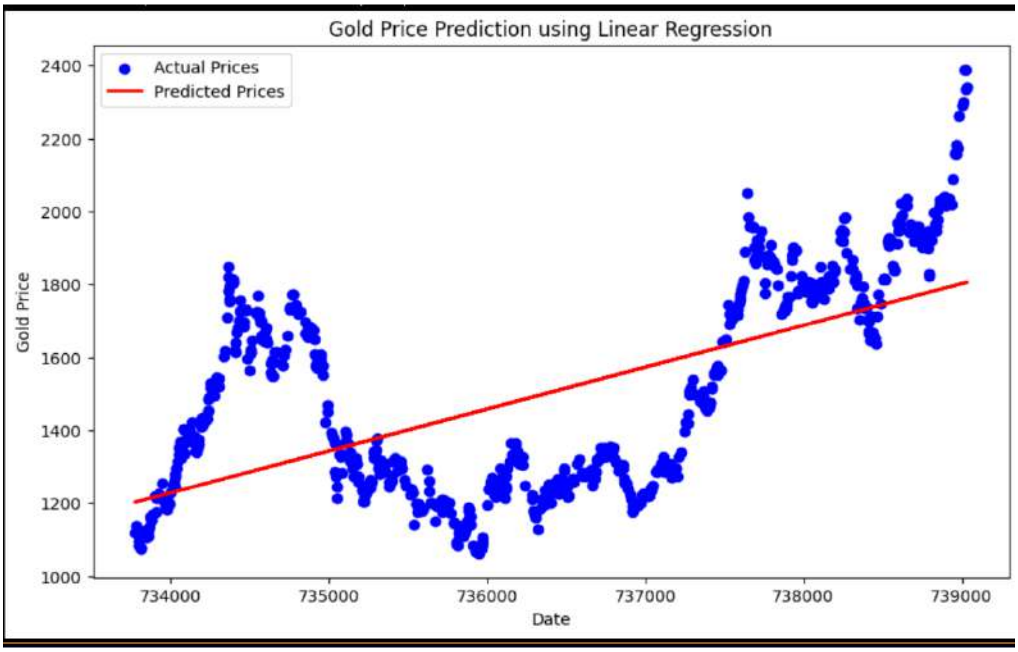


Artificial Intelligence and Machine Learning Laboratory

Reg no:24MAI0104
Name: GOWTHAM.J







CONCLUSION :

In this project, we successfully developed a Linear Regression model to predict gold prices based on historical data. The steps involved collecting and preprocessing the data, selecting relevant features, training the model, and evaluating its performance.

Key Takeaways:

- **Model Performance**: The Linear Regression model provided a basic but effective method for predicting gold prices. The evaluation metrics, such as Mean Squared Error (MSE) and R^2 score, helped us understand the accuracy and reliability of our predictions.
- **Simplicity and Interpretability**: Linear Regression is a straightforward algorithm that offers easy interpretability.

Artificial Intelligence and Machine Learning Laboratory

Reg no:24MAI0104
Name: GOWTHAM.J

TITLE : (POLYNOMIAL REGRESSION)

3. Predicting house prices using polynomial regression can be accomplished using publicly available datasets. One of the most popular datasets for this purpose is the **Ames Housing dataset**.

This dataset is comprehensive and provides detailed information about houses in Ames, Iowa, which can be used for various regression tasks, including polynomial regression.

DESCRIPTION :

Polynomial regression is a form of regression analysis that models the relationship between the independent variable xxx and the dependent variable Y as an N-th degree polynomial. Here's a brief overview:

Polynomial Regression Overview

- **Objective**: To model the relationship between a feature and the target variable as a polynomial function, which allows for capturing non-linear relationships.
- **Mathematical Form**:

  y=β0+β1x+β2x2+β3x3+…+βnx^n+ϵ

Where y is the dependent variable, x is the independent variable, βi are the coefficients, and ϵ epsilonϵ is the error term.

- **Advantages**:
  - Can model non-linear relationships between features and target variables.
  - Flexible in fitting the data, making it useful for capturing complex patterns.
- **Disadvantages**:
  - Risk of overfitting, especially with high-degree polynomials.
  - Can become computationally expensive and harder to interpret as the degree increases.
- **Applications**:
  - Useful for regression tasks where the relationship between variables is not linear.

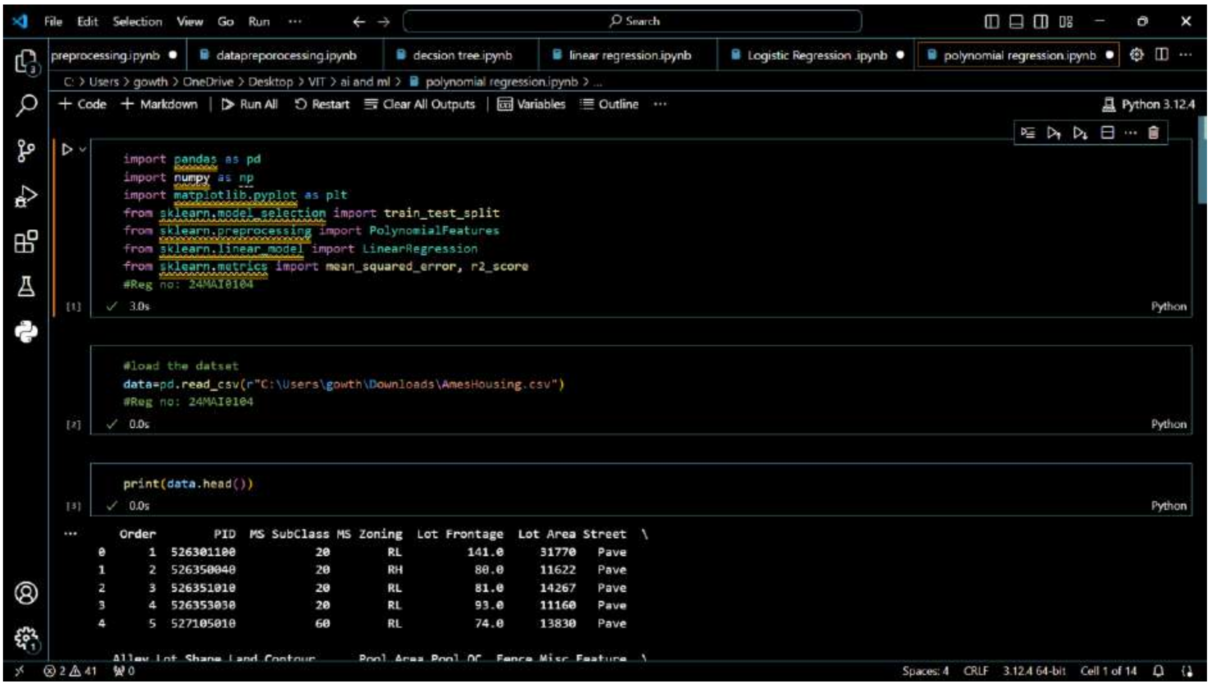
- Commonly applied in scenarios like predicting house prices, where relationships between features (e.g., square footage, number of rooms) and target variables (e.g., price) can be complex.

Example with the Ames Housing Dataset:

1. **Feature Selection**: Choose features relevant to house prices (e.g., square footage, number of bedrooms).
2. **Polynomial Features**: Generate polynomial features from selected features (e.g., square footage squared, cubic terms).
3. **Model Training**: Fit a polynomial regression model using these features.
4. **Evaluation**: Assess model performance using metrics like Mean Squared Error (MSE) and adjust the polynomial degree to balance bias and variance.

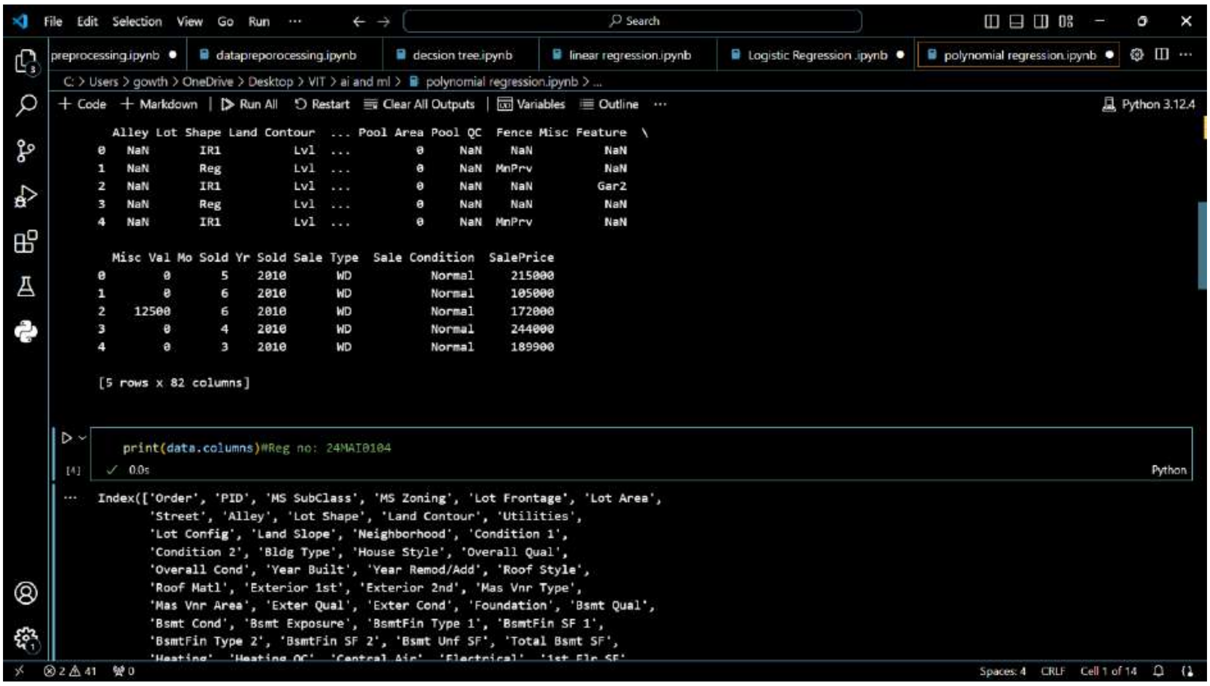
Polynomial regression can effectively capture intricate relationships in datasets like the Ames Housing dataset, providing insights into how various features impact house prices.

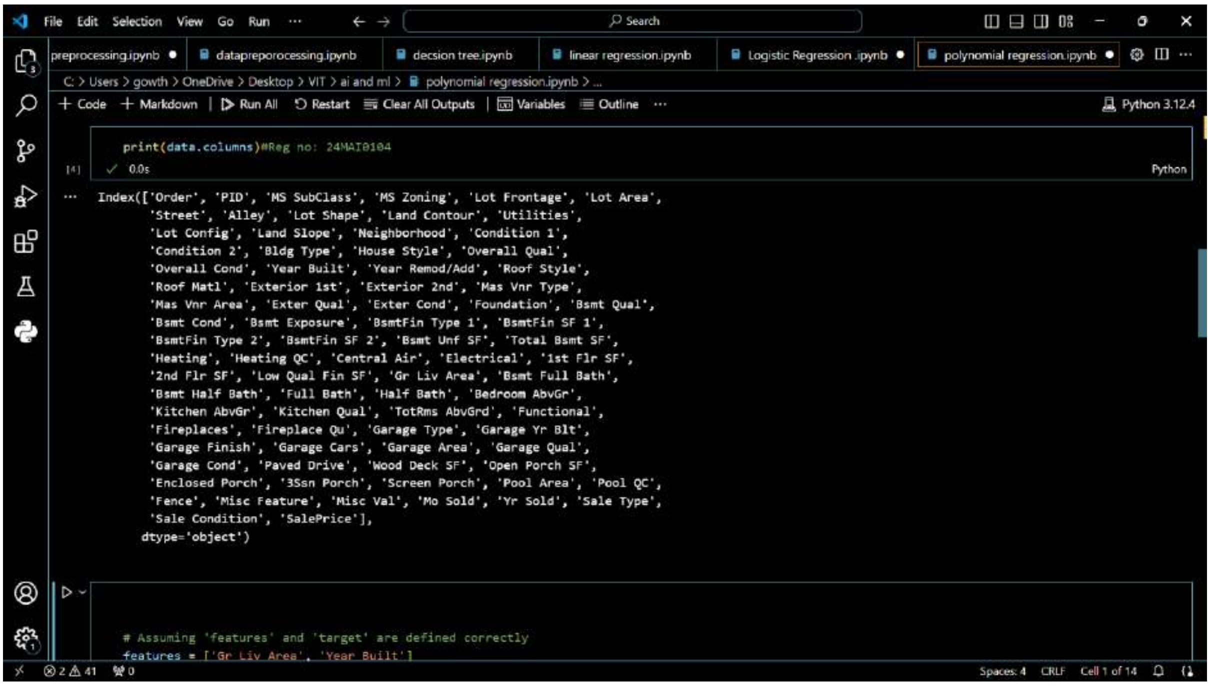
SOURCE CODE AND RESULT:



Artificial Intelligence and Machine Learning Laboratory

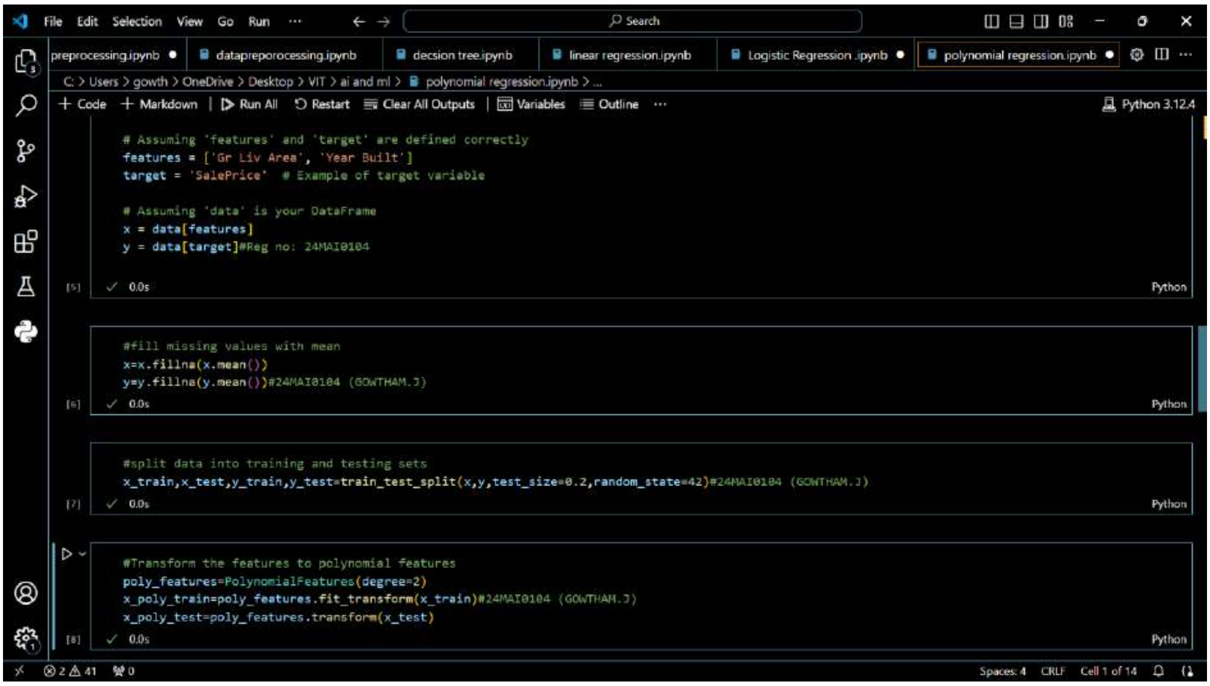
Reg no:24MAI0104
Name: GOWTHAM.J

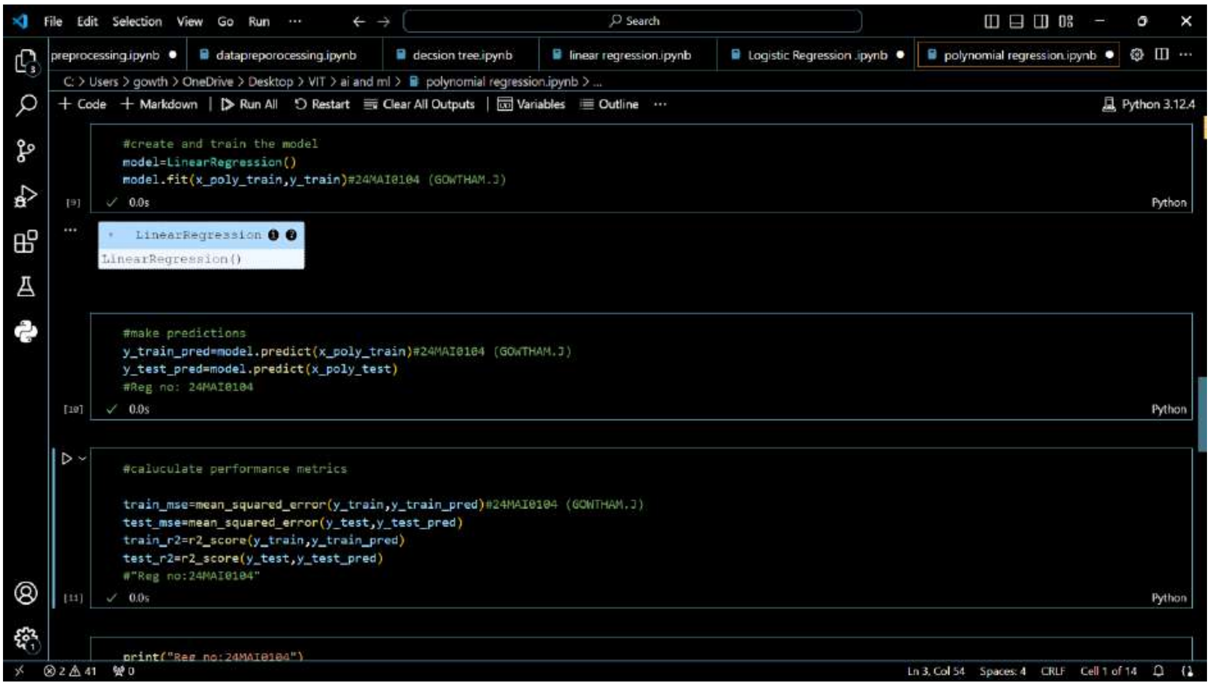




Artificial Intelligence and Machine Learning Laboratory

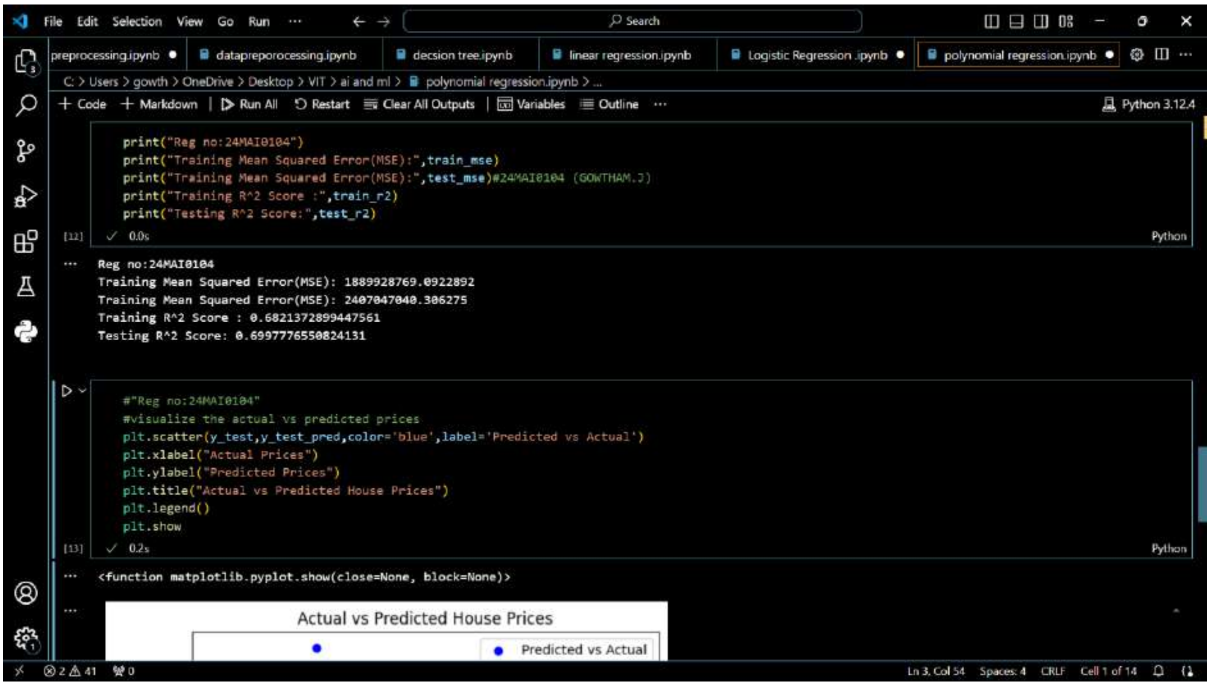
Reg no:24MAI0104
Name: GOWTHAM.J

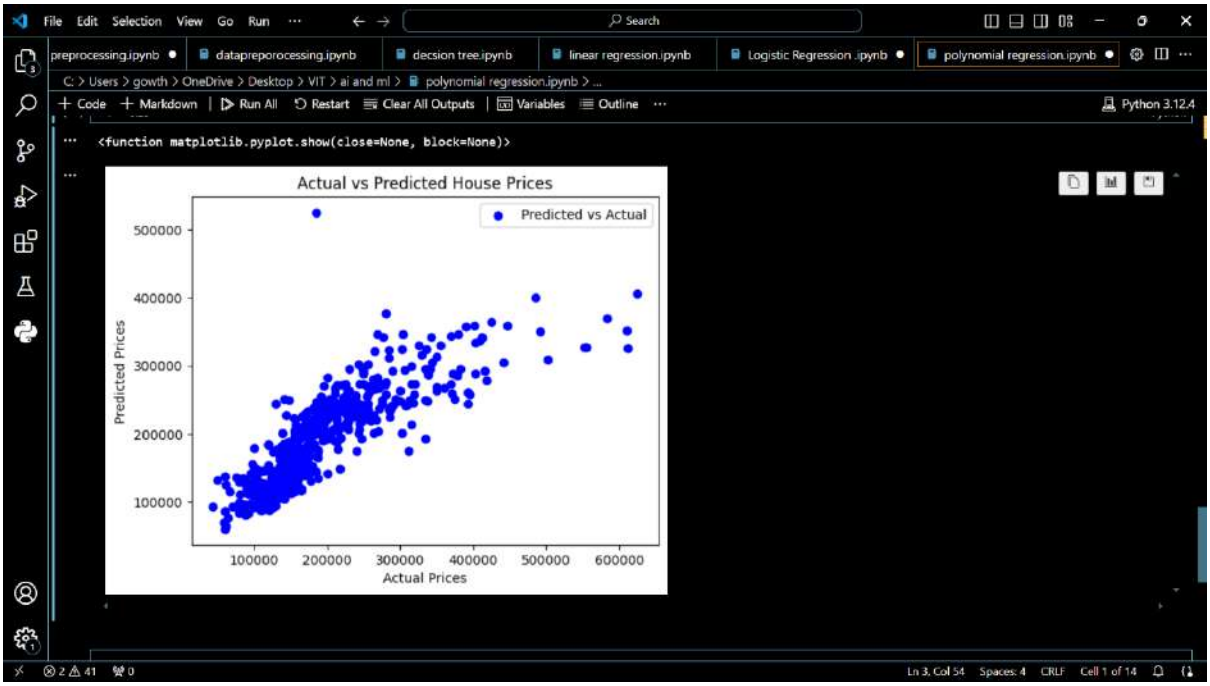




Artificial Intelligence and Machine Learning Laboratory

Reg no:24MAI0104
Name: GOWTHAM.J





CONCULSION :

When using polynomial regression for predicting house prices, here's a concise consultation guide:

1. Data Preparation

- **Feature Selection**: Identify relevant features such as square footage, number of bedrooms, and age of the house.
- **Preprocessing**: Handle missing values, normalize or standardize features, and create polynomial features.

2. Polynomial Degree

- **Choosing Degree**: Start with a low-degree polynomial (e.g., quadratic) and incrementally increase the degree to capture more complexity.
- **Avoid Overfitting**: Use cross-validation to find the optimal polynomial degree that balances model complexity and generalization.

3. Model Training

- **Train-Test Split**: Divide the dataset into training and testing sets to evaluate model performance.
- **Fit Model**: Train the polynomial regression model on the training set.

4. Evaluation

- **Performance Metrics**: Evaluate using metrics like Mean Squared Error (MSE) or R^2 score.
- **Visual Inspection**: Plot predictions versus actual values to visually inspect fit quality.

5. Refinement

- **Feature Engineering**: Experiment with different polynomial terms and interaction features.
- **Regularization**: Consider using regularization techniques (e.g., Ridge or Lasso) to mitigate overfitting.

6. Deployment

- **Scalability**: Ensure the model performs well on new, unseen data before deploying.
- **Interpretability**: While polynomial models can be complex, aim to interpret the impact of key features on predictions.

Artificial Intelligence and Machine Learning Laboratory

Reg no:24MAI0104
Name: GOWTHAM.J

TITLE: (Logistic Regression)

4.Logistic Regression is a popular algorithm for binary classification tasks. A simple example of classifying images using Logistic Regression.

use the MNIST dataset, which contains images of handwritten digits and classify whether the digit is a '0' or a '1'.

DESCRIPTION :

Logistic Regression is widely used for binary classification tasks, where the goal is to predict one of two possible outcomes. In this example, we'll use Logistic Regression to classify images from the MNIST dataset, which consists of handwritten digits. Specifically, we'll focus on classifying whether a given digit is a '0' or a '1'.

Dataset:

The MNIST dataset is a well-known dataset in machine learning, comprising 70,000 images of handwritten digits (0-9). Each image is 28x28 pixels, and each pixel has a grayscale value between 0 and 255. For this binary classification task, we'll filter the dataset to include only the images labeled as '0' or '1'.

Problem Statement:

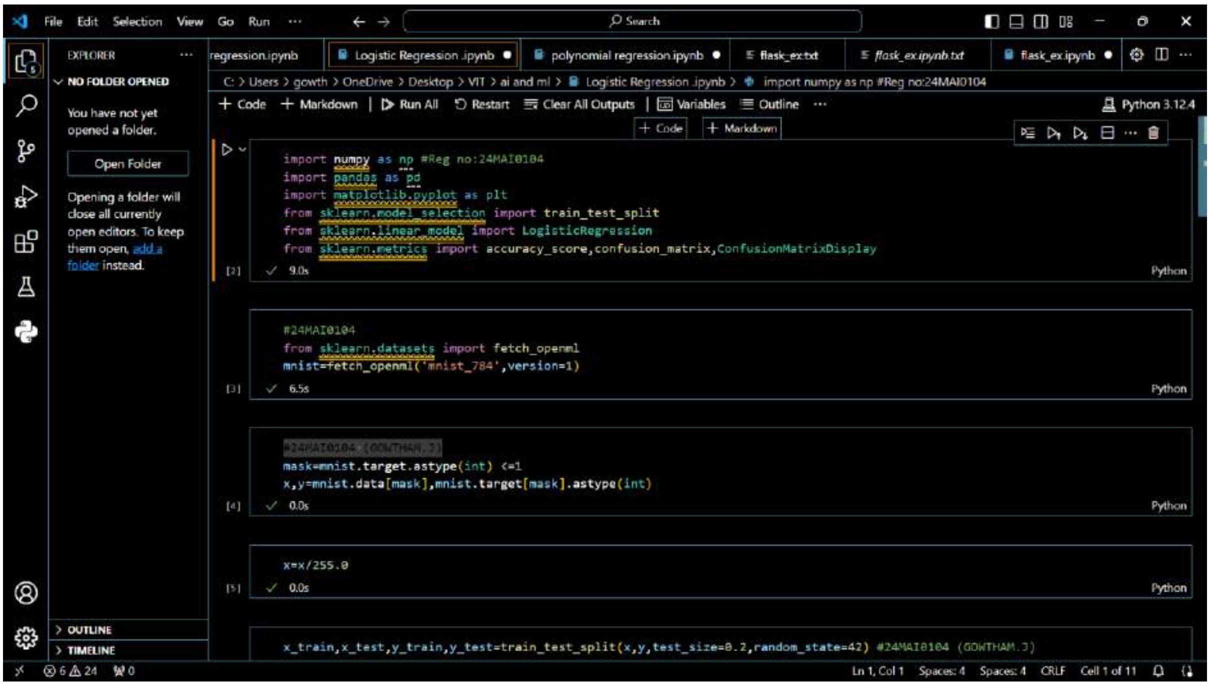
Given an image of a handwritten digit, the task is to determine whether the digit is a '0' or a '1'. Logistic Regression will be used to model this binary classification problem.

Steps:

1. Data Preprocessing:
    - Data Selection: Filter the MNIST dataset to include only images labeled as '0' or '1'.
    - Flattening the Images: Each 28x28 image will be flattened into a 784-dimensional vector (since 28x28 = 784). This will serve as the input feature vector for Logistic Regression.
2. Model Training:
    - Logistic Regression Model: Initialize and train a Logistic Regression model using the preprocessed feature vectors.
    - Training Process: The model will learn the optimal coefficients by minimizing the difference between the predicted probabilities and the actual labels using a technique called Maximum Likelihood Estimation (MLE).

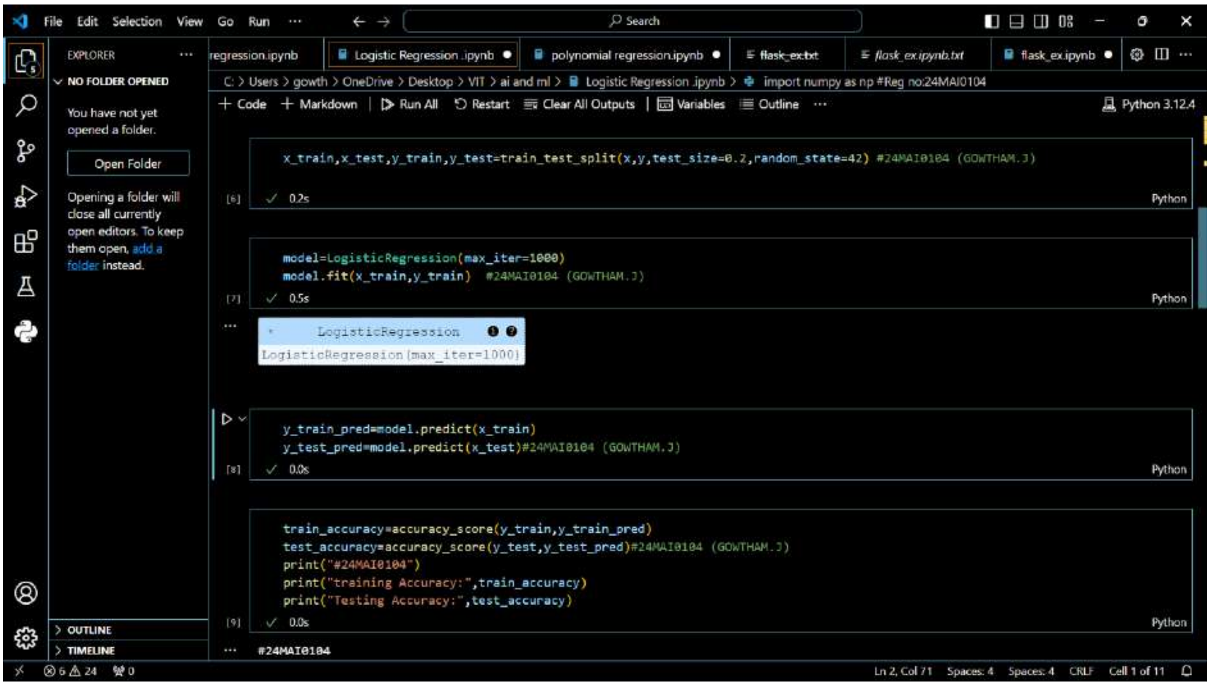
3. Prediction:
    - Probability Output: For a given input image, the model will output a probability indicating the likelihood that the image represents a '1'.
    - Decision Rule: If the probability is greater than 0.5, the model predicts '1'; otherwise, it predicts '0'.
4. Evaluation:
    - Accuracy: The accuracy of the model can be evaluated on a separate test set of '0' and '1' images to determine how well the model generalizes to unseen data.
    - Confusion Matrix: This will give insights into how many '0's and '1's were correctly or incorrectly classified.

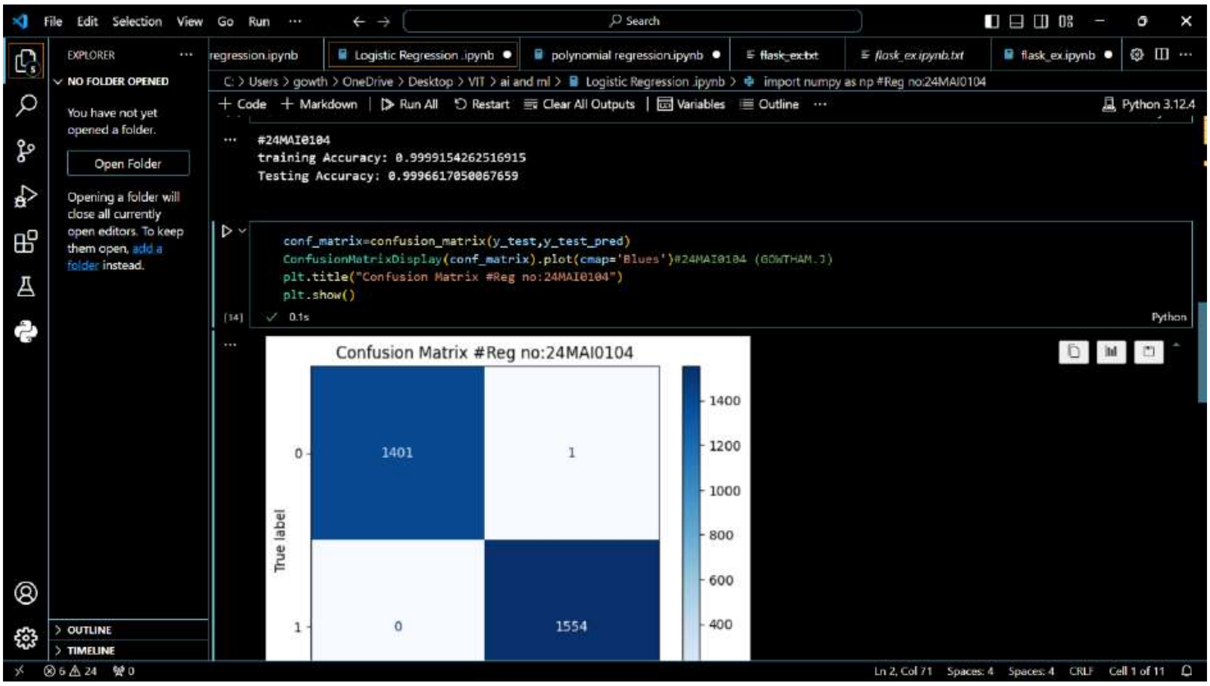
SOURCE CODE AND RESULT :



Artificial Intelligence and Machine Learning Laboratory

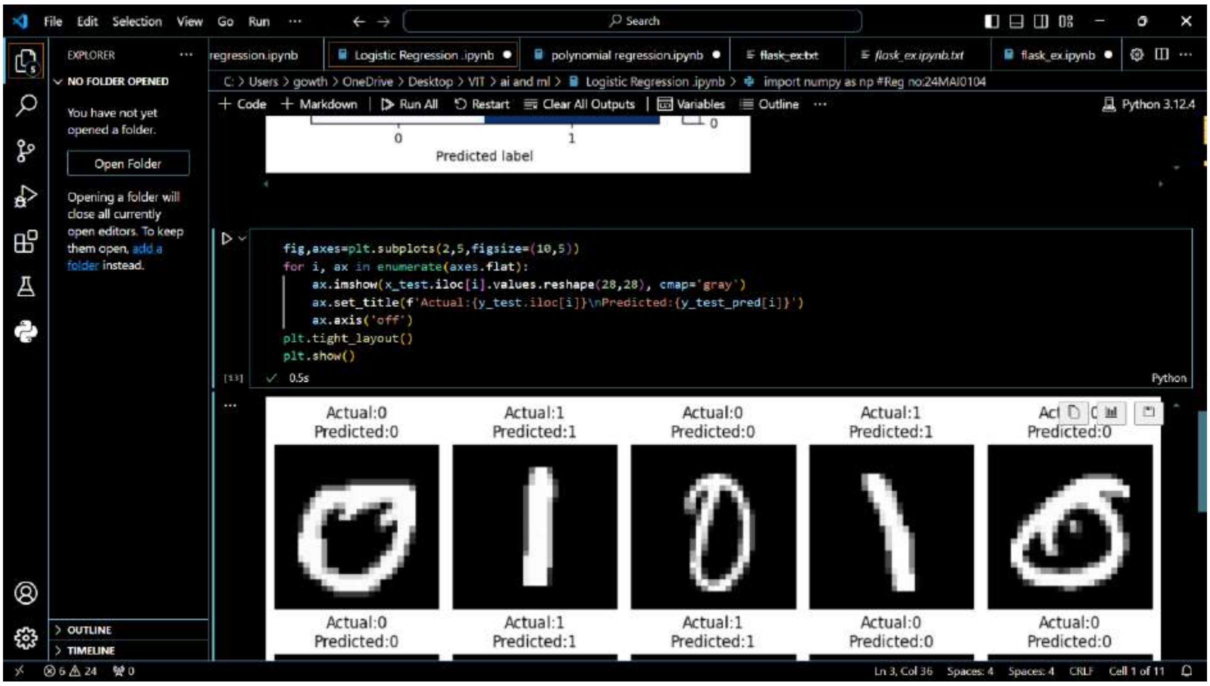
Reg no:24MAI0104
Name: GOWTHAM.J

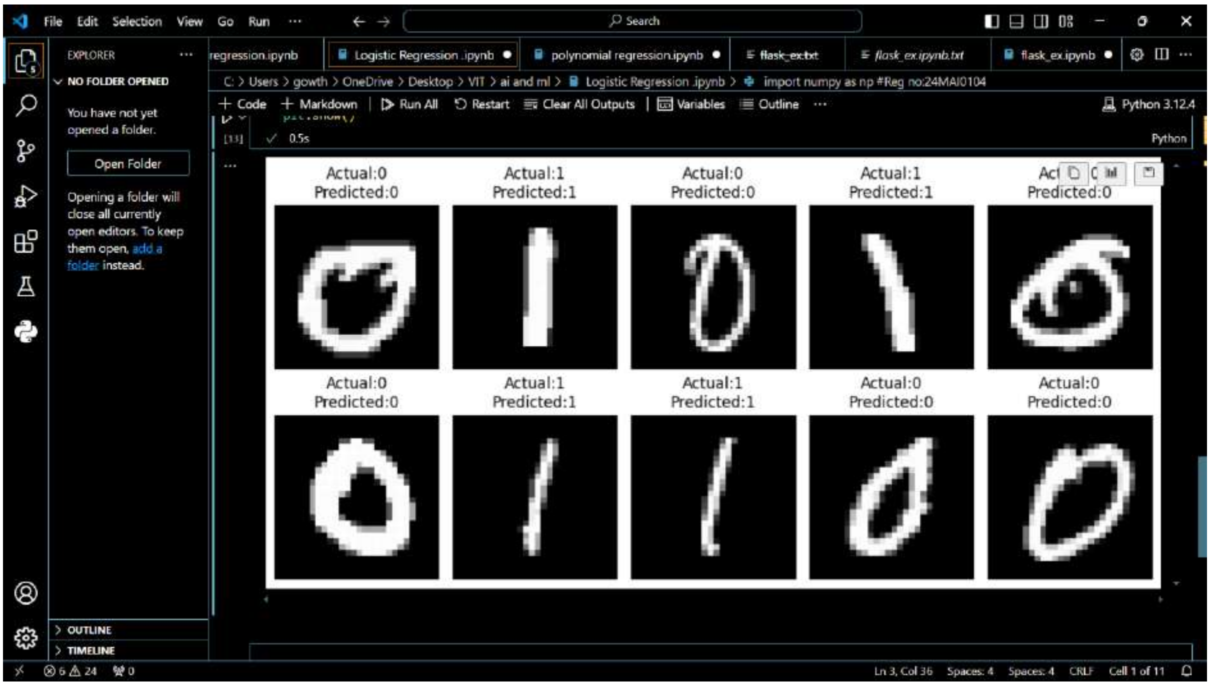




Artificial Intelligence and Machine Learning Laboratory

Reg no:24MAI0104
Name: GOWTHAM.J





CONCLUSION :

In conclusion, using Logistic Regression for classifying images of handwritten digits '0' and '1' from the MNIST dataset proves to be a straightforward yet effective approach for binary classification tasks. This method leverages the simplicity and interpretability of Logistic Regression while demonstrating its application in the context of image classification.

Key Points:

1. Model Performance: Logistic Regression provides a reasonable baseline performance for binary classification tasks, achieving decent accuracy in distinguishing between '0' and '1' digits based on pixel intensity values.
2. Interpretability: The coefficients learned by Logistic Regression offer insights into which pixels contribute positively or negatively to the likelihood of a digit being classified as '1' versus '0'. This interpretability is valuable for understanding the model's decision-making process.
3. Implementation Ease: Logistic Regression is relatively easy to implement and computationally efficient, making it suitable for initial experimentation and as a baseline comparison against more complex models.