Data and Artificial Intelligence
Cyber Shujaa Program

Week 7 Assignment
Linear Regression Model

Student Name: Faith Jeptoo

Student ID: CS-DA02-25005

Table of Contents

1. Introduction

2. Dataset Exploration

3. Data Preparation

4. Model Training

5. Model Evaluation

6. Visualizations

7. Conclusion

Introduction

This project focuses on implementing a Simple Linear Regression model to understand and predict a dependent variable based on one independent variable.

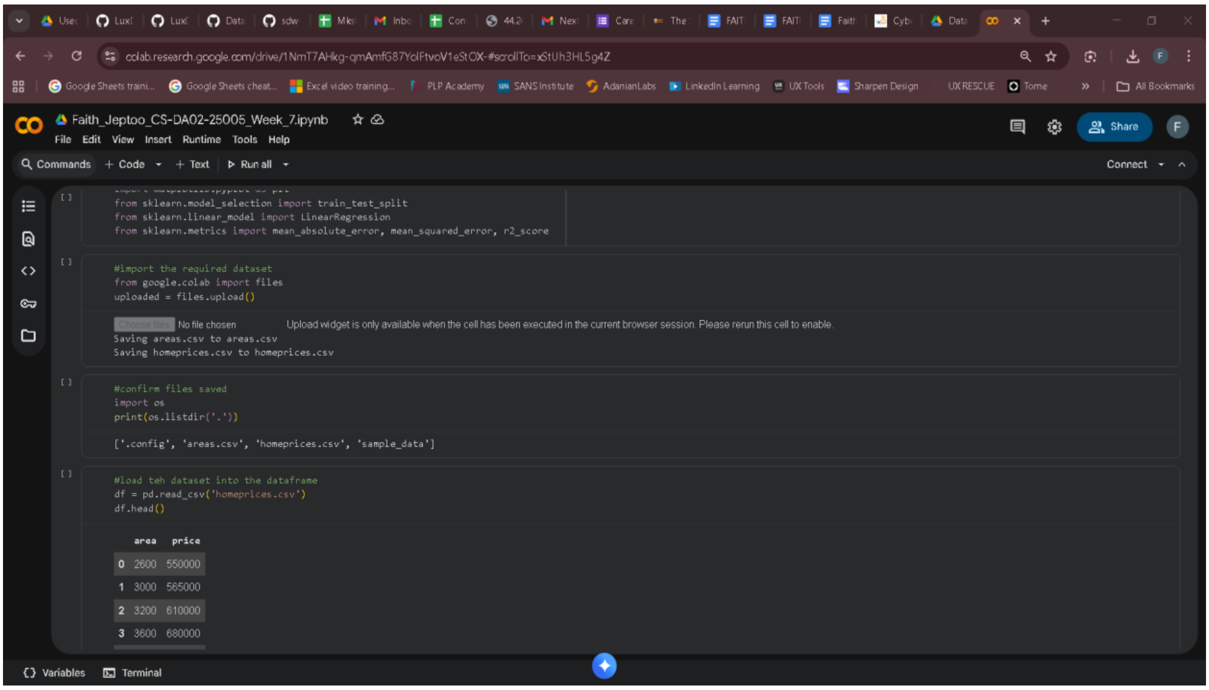
Linear regression is one of the simplest and most widely used statistical algorithms in data science. It helps in establishing a relationship between variables by fitting a straight line to the data.

In this assignment, the primary objective was to explore the dataset, prepare data, train a regression model, evaluate model performance, and visualize results

Task Completion

1 Dataset Exploration

In this section, the dataset was loaded and examined to understand its structure, size, and features. Summary statistics were generated to analyze the distribution of values, and missing values were checked. Visualization through a scatter plot was used to understand the relationship between the independent and dependent variables before applying the regression model


colab.research.google.com/drive/1NmT7AHkg-qmAmfG87YolFtvoV1eStOX-#scrollTo=xStUh3HL5g4Z
Faith_Jeptoo_CS-DA02-25005_Week_7.ipynb
File Edit View Insert Runtime Tools Help
Commands + Code + Text ▷ Run all
Share
Connect
Variables
Terminal

```
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score
```

```
#import the required dataset
from google.colab import files
uploaded = files.upload()
```

```
Choose files No file chosen     Upload widget is only available when the cell has been executed in the current browser session. Please rerun this cell to enable.
Saving areas.csv to areas.csv
Saving homeprices.csv to homeprices.csv
```

```
#confirm files saved
import os
print(os.listdir('.'))
```

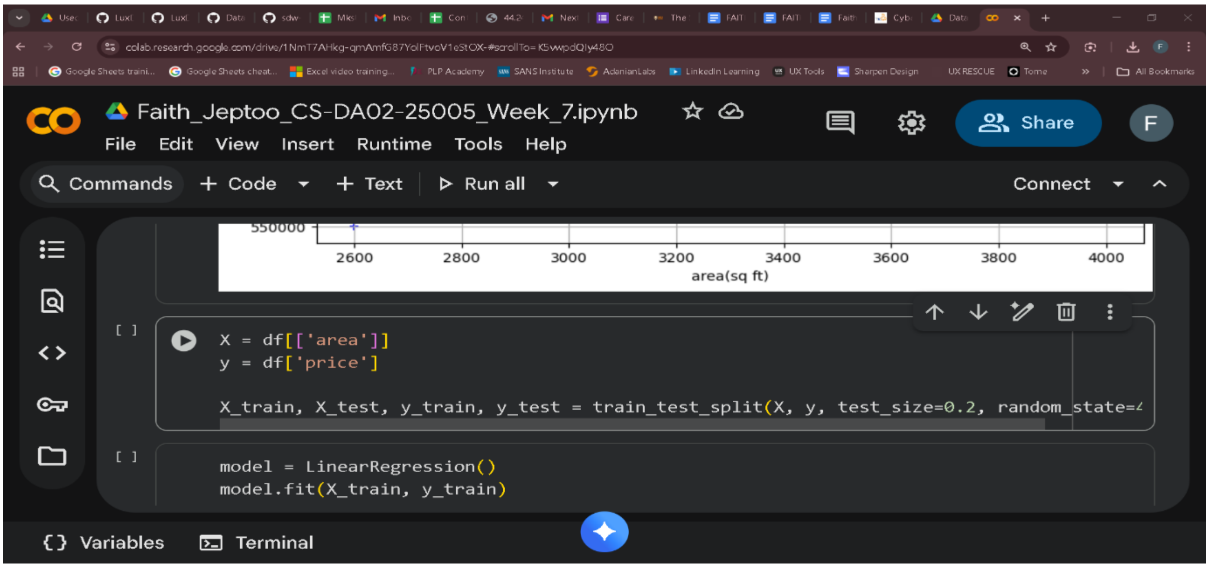
```
['.config', 'areas.csv', 'homeprices.csv', 'sample_data']
```

```
#load teh dataset into the dataframe
df = pd.read_csv('homeprices.csv')
df.head()
```

```
   area   price
0  2600  550000
1  3000  565000
2  3200  610000
3  3600  680000
```

2 Data Preparation

The dataset was checked for missing values, and the data was cleaned accordingly. The independent and dependent variables were selected based on the dataset. The dataset was then split into training and testing sets to ensure the model could be evaluated objectively.

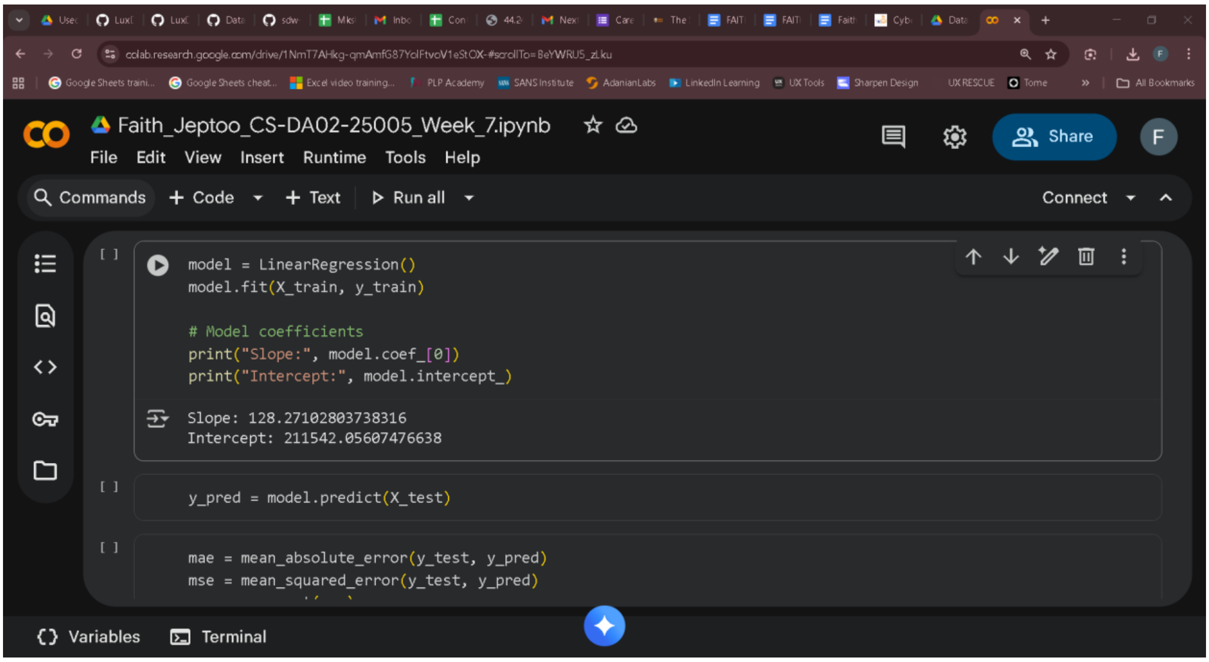

colab.research.google.com/drive/1NmT7AHkg-qmAmfG87YolFtvoV1eStOX-#scrollTo=K5vwpdQlyl48O
Faith_Jeptoo_CS-DA02-25005_Week_7.ipynb
File Edit View Insert Runtime Tools Help
Commands + Code + Text ▷ Run all
Share
Connect
550000
2600
2800
3000
3200
3400
3600
3800
4000
area(sq ft)
Variables
Terminal

```
X = df[['area']]
y = df['price']

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=
```

```
model = LinearRegression()
model.fit(X_train, y_train)
```

3. Model Training

The Simple Linear Regression model was trained using the training data. The LinearRegression() class from Scikit-learn was used to fit the model. Once the model was trained, the slope (coefficient) and intercept were extracted to understand the linear equation used for prediction.



colab.research.google.com/drive/1NmT7AHkg-qmAmfG87YolFtvoV1eStOX-#scrollTo=8eYWRU5_zLku
Faith_Jeptoo_CS-DA02-25005_Week_7.ipynb
File Edit View Insert Runtime Tools Help
Share
Commands + Code + Text Run all
Connect
Variables Terminal

```
model = LinearRegression()
model.fit(X_train, y_train)

# Model coefficients
print("Slope:", model.coef_[0])
print("Intercept:", model.intercept_)
```

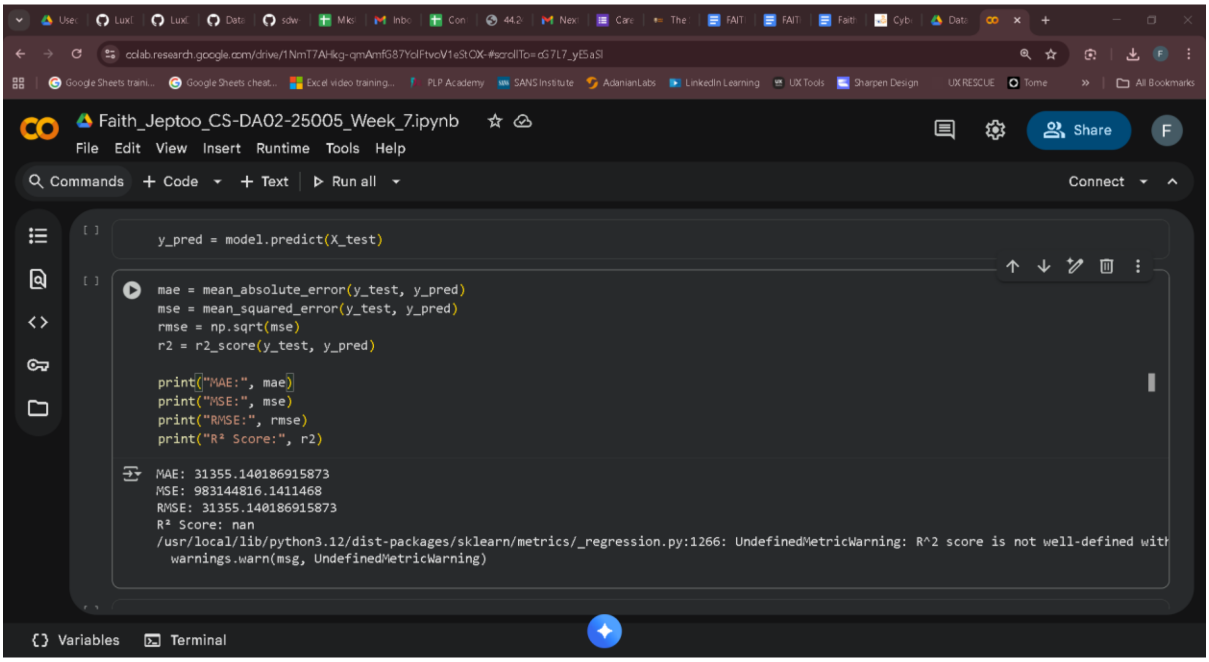
```
Slope: 128.27102803738316
Intercept: 211542.05607476638
```

```
y_pred = model.predict(X_test)
```

```
mae = mean_absolute_error(y_test, y_pred)
mse = mean_squared_error(y_test, y_pred)
```

4. Model Evaluation

To assess the performance of the trained model, several evaluation metrics were used, including Mean Absolute Error (MAE), Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and the R² Score. These metrics help determine how accurately the model predicts real values.



colab.research.google.com/drive/1NmT7AHkg-qmAmfG87YolFtvoV1eStOX-#scrollTo=dG7L7_yE5aSl
Faith_Jeptoo_CS-DA02-25005_Week_7.ipynb
File Edit View Insert Runtime Tools Help
Share
Commands + Code + Text Run all
Connect
Variables Terminal

```
y_pred = model.predict(X_test)
```

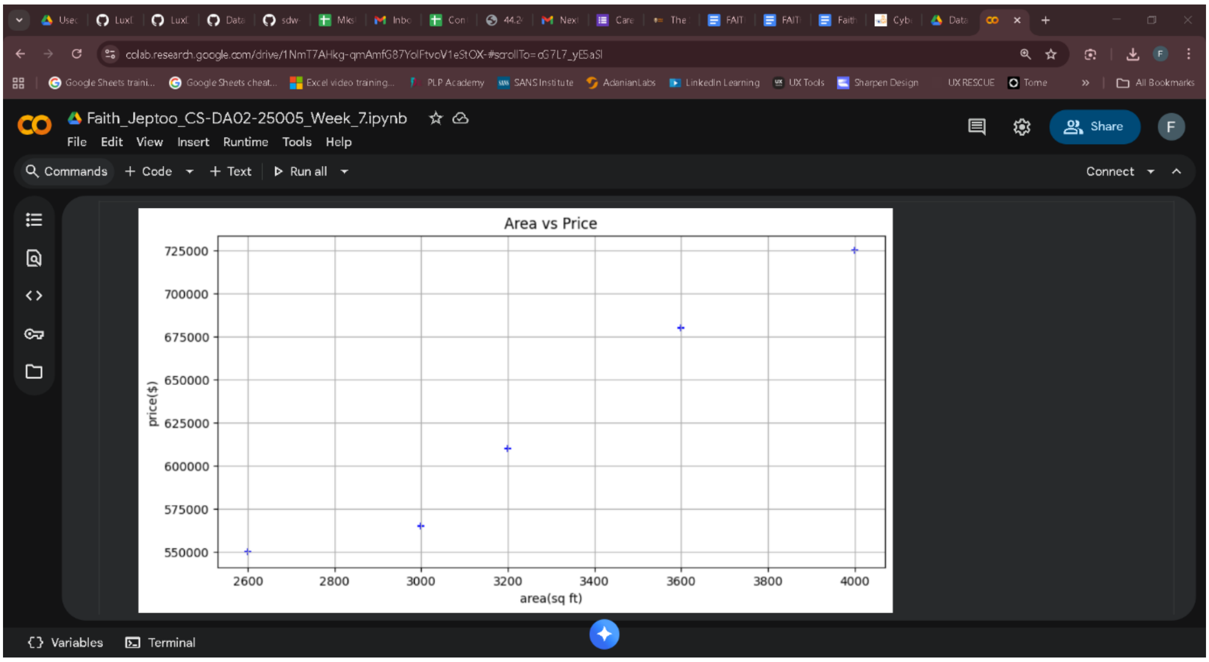
```
mae = mean_absolute_error(y_test, y_pred)
mse = mean_squared_error(y_test, y_pred)
rmse = np.sqrt(mse)
r2 = r2_score(y_test, y_pred)

print("MAE:", mae)
print("MSE:", mse)
print("RMSE:", rmse)
print("R² Score:", r2)
```

```
MAE: 31355.140186915873
MSE: 983144816.1411468
RMSE: 31355.140186915873
R² Score: nan
/usr/local/lib/python3.12/dist-packages/sklearn/metrics/_regression.py:1266: UndefinedMetricWarning: R^2 score is not well-defined with
  warnings.warn(msg, UndefinedMetricWarning)
```

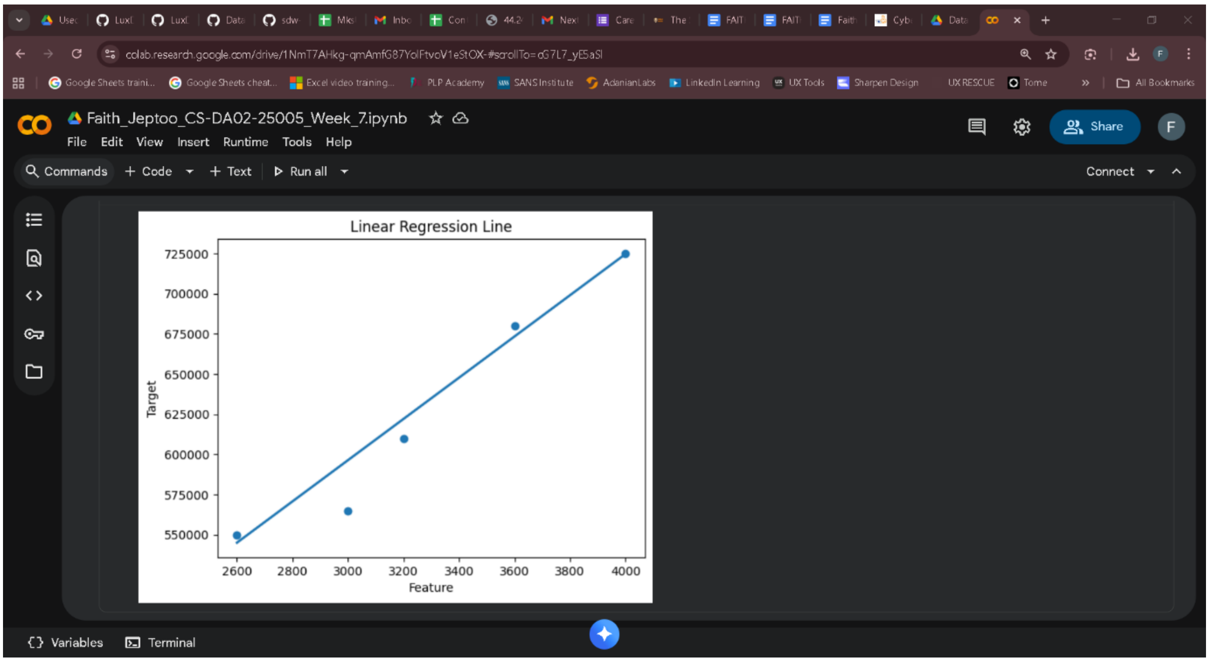
5. Visualizations

Visualizations were created to understand the relationship between variables and validate the performance of the regression model. A scatter plot was used to display the dataset. A regression line was then plotted to show how well the model fits the data visually.

i. Scatter plot



ii. Linear Regression Line



6. Conclusion

This assignment provided hands-on experience in working with a real dataset and applying a simple linear regression model. Through the steps of data exploration, preparation, model training, and evaluation, I gained a deeper understanding of how regression works and how model performance is assessed. Visualization further strengthened the interpretation of results. This knowledge forms a fundamental base for more advanced machine learning models.

4. Link to Google Colab

https://colab.research.google.com/drive/1NmT7AHkg-qmAmfG87YolFtvoV1eStOX-?usp=sharing