IS6052

Predictive Analytics

Individual Assignment

Full Name: Weixi Wang

Student Number: 124103176

Submission Data: 2/12/2024

Contents

Introduction

Background and Problem Statement

In this study and report, we need to explain which model is the most appropriate for the Irish property data forecasting study and which is the most optimal for our data set. In the context of Irish property data, we can see that because of the imperfections in the underlying data, the predictive analysis will be inaccurate because we need to use Python software to clean and predict the analysis of the data in this research and use the EDA method to analyze our data.

Dataset Description and EDA

EDA serves the fundamental purpose of scrutinizing data before making any assumptions. It aids in pinpointing evident discrepancies, comprehending data patterns more thoroughly, pinpointing outliers or irregular occurrences, and unveiling intriguing relationships among variables. Data scientists utilize exploratory analysis to ensure the validity and relevance of their findings to desired business objectives. **(Role of EDA in data science, 2024)**

Dataset Summary

This dataset, provided by the professor, contains tens of thousands of properties and eleven relevant features to describe the variables or attributes of the properties (the "ID" column does not apply to the study and analysis of this dataset and can be ignored), obviously, it is more scientific and efficient to process and analyze it using python programming software. therefore, the processing and analysis using Python programming software is more scientific and efficient.

Characterization

From the 11 columns of the dataset, the columns "property scope," "location," "price-per-sqft-$," etc., are the most relevant to the study of this dataset. "Other columns are the ones that have a significant impact on the prediction or assessment of future real estate trends.

Initial Exploratory Analysis of Data

We formally enter the EDA methodology from this stage onwards, first roughly exploring the data performance of the relevant feature columns through various graphical visualizations, laying the foundation for deeper analysis at a later stage.

Analysis of property prices per square meter

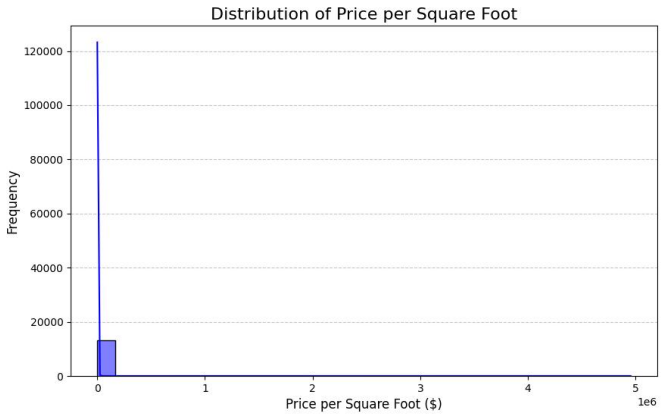


Fig. 1

Looking at the histogram distribution of prices per square foot for properties, there is a skewed distribution in the data, with many properties with prices per square foot concentrated in the lower range, but there are also some extremely high values, which may be outliers in the data or a very small number of high-end properties. These high values cause the graph to appear heavily skewed, resulting in most of the data points being concentrated almost to the left. To further analyze and improve the performance of the model, these outliers may need to be processed, or data transformed to better capture the distribution of prices within the normal range.

Distribution of BER Ratings

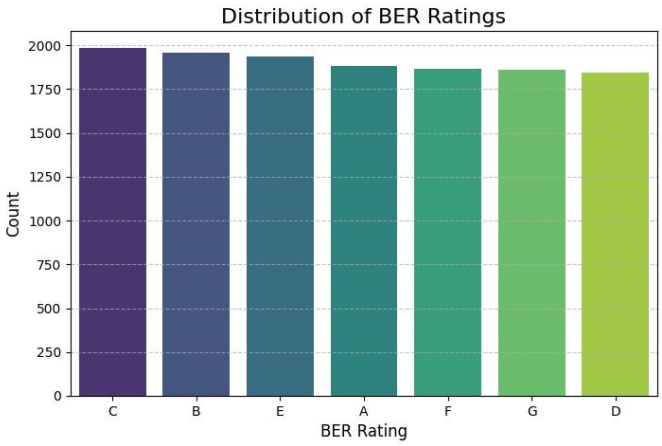


Fig. 2

The data in the BER column shows a relatively even distribution of the number of properties across ratings, with no significant deviations. The number of properties with a rating of C is slightly higher than the other ratings, but overall, all ratings (from A to G) appear in the data with close frequency. Suggesting that the energy efficiency of properties is relatively evenly distributed may mean that most of the energy efficiency ratings for properties in this dataset are more spread out. For analyzing house prices or home purchase decisions, an even distribution of BER ratings may not have too extreme an impact.

Property Scope Distribution

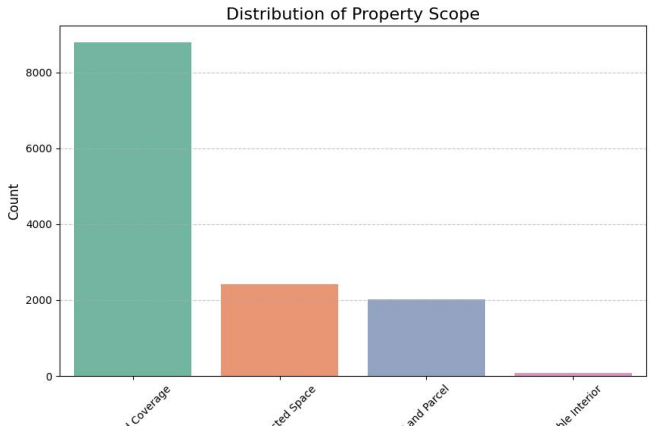


Fig. 3

“Extended Coverage” dominates the bar chart and is the most common property type compared to the other three. The “Usable Interior” type is the least common and has

the highest variability in the data compared to the other three types. This unbalanced distribution may affect the performance of the model, so in subsequent modeling, these categories may need to be characterized or otherwise adjusted to ensure that the impact of each type is fairly represented.

Relationship Between Location and Price-per-sqft

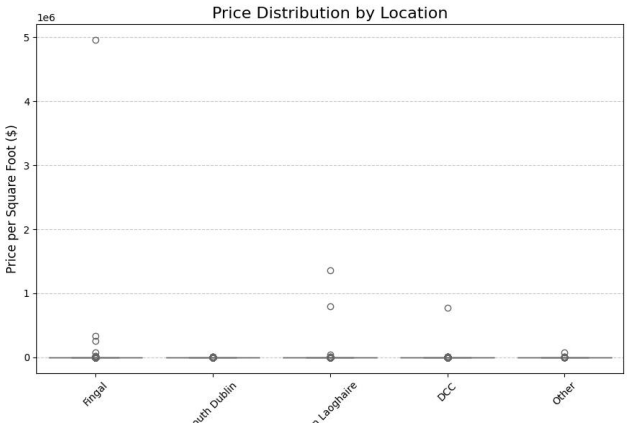


Fig. 4

The distribution of data in the “Location” box plot shows that Fingal, South Dublin, Dun Laoghaire, and DCC have a relatively concentrated distribution of prices per square foot, with some outliers, suggesting that some properties in these areas are more expensive. In contrast, prices in other places are cheaper and more evenly distributed, with less overall price volatility. Despite the differences in the distribution of prices across regions, price extremes (outliers) occur in multiple regions and need to be dealt with in the modeling to help improve the accuracy of the model.

Correlation Between Total_sqft and Price-per-sqft

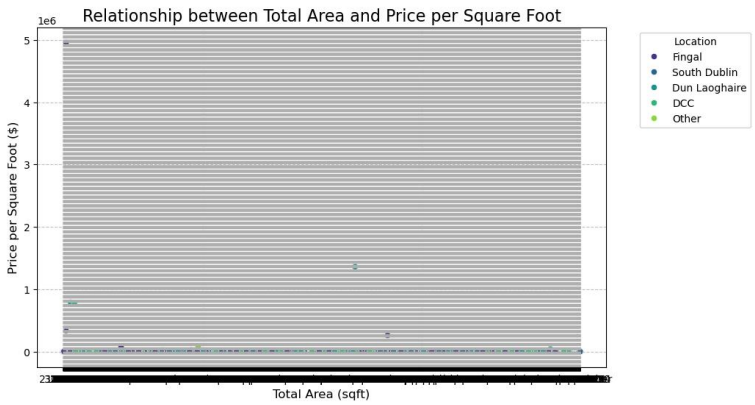


Fig. 5

To visualize the extent of the relationship between the two, a scatter plot was created to observe and analyze. However, the relationship between (total_sqft) and price per square foot (price-per-sqft-$) is not obvious. Most of the data points are concentrated in the lower price ranges and there is some overlap in the distribution between the different regions (e.g. Fingal, South Dublin, Dun Laoghaire, DCC, and Other), suggesting that the region does not significantly influence price per sqft. Although prices are generally lower for properties with larger total square footage, there are a few outliers.

Overall, the total square footage of properties did not show a clear linear relationship, so further exploration of the impact of other factors on price may be needed.

Data Preparation and Feature Engineering

A. Data Cleaning

Clean data provides the foundation for data analysis, making it easier to gain insights from data. It is important to ensure data records are accurate and up-to-date in order to deliver reliable data analytical results. (**Sharma, R., 2024)**

The processing of missing values is based on the principle of categorization, first, checking and counting the missing values, statistically visible missing values exist in 5 columns, in which the balcony column has the most missing values up to 609 items, and the location column has at least only 1 missing value. In this case, the missing values were processed according to the type of filling, in which the two columns bath and balcony will be filled with missing values to 0, for the following reasons:

1. Logical reason for populating with 0

a. To avoid sensitivity to missing values during machine learning

Machine learning models usually cannot handle missing values directly, so they must be populated, and populating them with 0 maintains the numeric format of the fields so that the model can read and learn these features correctly.

b. Compared to other padding methods

Comparison with padding to mean or median:

A mean or median fill can mislead the model into thinking that all properties have at least a certain number of bathrooms or balconies.

Comparison with fill as 'unknown' or categorical codes:

A fill of 'unknown' cannot be quantified, leading the model to fail to understand what it means.

2. The effect of a fill of 0 on subsequent analysis

a. Improved accuracy of model predictions

For categorical models (e.g. 'whether to buy a property'), the number of balconies and bathrooms are important influences:

If the number of balconies is zero, this may affect the willingness to buy (e.g., households are more likely to choose a property with a balcony).

Filling in 0 helps the model to more accurately differentiate the impact of having a balcony/bathroom on the purchase decision.

For regression models (e.g. 'house price forecasting'), the lack of a balcony or bathroom usually reduces the value of a property. Filling with zeros allows the model to incorporate these features as negative effects in the price prediction.

b. Maintain data consistency and feature integrity

Retaining missing values may cause some records to be ignored in the analysis, resulting in a smaller training set and reducing the model's ability to generalize.

Filling to 0 ensures that each record contains all features, improving data integrity and consistency.

c. Support for feature engineering and new feature construction

Using the populated bath and balcony features, derived features such as “number of bathrooms per bedroom” and “ratio of balconies to bedrooms” can be constructed to further enhance the model's expressiveness.

d. Improve model robustness

Filling 0 makes the model more robust and avoids the instability caused by improper handling of missing values. For example, properties with 0 number of balconies should not create unreasonable noise in the model predictions.

For the other columns of the missing values to fill the way, the uniform use of “Unknown” way to fill, the same way to fill the reasons for this way combined with the later model analysis of the key points of the factors are as follows:

1. Maintaining consistency of categorical variables

Categorical variables are often required to participate in model training with specific categorical values. Deleting the missing values would result in a loss of sample information, whereas filling with Unknown preserves the integrity of the data.

2. Avoid False Assumptions

Using plurality fill for categorical variables can introduce false assumptions. Using Unknown explicitly states that this data is uncertain and avoids misclassification.

Translated with www.DeepL.com/Translator (free version)

a. Impact of the classification model

The Unknown category may become an important feature to help the model distinguish the target category.

b. Impact of Regression Modeling

In house price prediction, properties with location = Unknown may present a unique price distribution, providing additional information for model fitting.

c. Prevention of bias

Filling in the Unknown prevents analytical bias caused by incorrect assumptions. For example, filling in a missing location as DCC may over-bias the model towards that area.

B. Handle the Outliers

When using the IQR method to detect outliers in the “price-per-sqft-$” column, the outliers are calculated to be in the range of values less than -23.67 or greater than 1336.59.

Since the price-per-square-foot is usually positive, the negative range of the lower bound can be ignored. The actual outliers are concentrated in properties larger than 1336.59. A total of 1,288 outliers were detected, suggesting that there are some extremely high-priced properties in the data that may have implications for subsequent analysis and modeling. Dealing with these outliers (e.g., removing or replacing them) can help improve the robustness of the model.

```
Cleaned dataset saved to 'cleaned_data_before_outlier_handling.xlsx'.

Outlier bounds for 'price-per-sqft-$': -23.67 to 1336.59

Number of outliers detected:
1288

Dataset with outliers saved to 'cleaned_data_before_outlier_adjustment.xlsx'.

Number of outliers remaining after adjustment:
0
```

Fig. 6

Outlier visualization

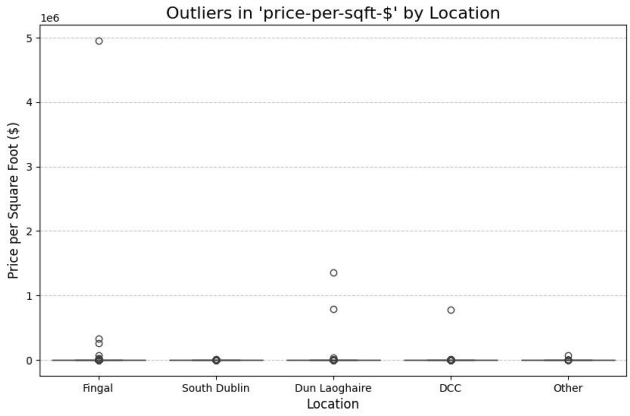


Fig.7

The price-per-sqft-$ column has outliers in all regions of the box plot. In Fingal and Other, the outliers are significant, with very few properties having prices above the normal range.

There are very few properties above the normal range, but there are some extremely overpriced properties. These outliers may have a negative impact on the data analysis and modeling, and therefore need to be addressed (e.g., removed or replaced) in subsequent modeling to improve the accuracy and robustness of the model.

Outliers that are outside the normal range are adjusted to boundary values using upper and lower boundary values. After adjustment, the number of outliers remaining is zero, indicating that all outliers have been effectively dealt with and the data has become more reasonable and suitable for subsequent analysis.

Ranges in the total_sqft column are processed before new features are added to the model; for ranges containing “-”, the function splits them and calculates the average of the ranges, and for single values, it converts them directly to floating point numbers.

For a single value, it is directly converted to a floating-point number. This ensures that the data format of the total_sqft column is consistent, which is convenient for subsequent analysis.

C. Feature Engineering

The quality and relevance of features are crucial for empowering machine learning algorithms and obtaining valuable insights. Overall, feature engineering is used to:

Improve model performance: As previously noted, features are key to the optimal performance of machine learning models. Features can be thought of as the recipe and the output of the model can be thought of as the meal

Lessen computational costs: When done right, feature engineering results in reduced computational requirements, like storage, and can improve the user experience by reducing latency

Improve model interpretability: Speaking to a human's ability to predict a machine learning model's outcome, well-chosen features can assist with interpretability by helping explain why a model is marking certain predictions

Feature engineering will ultimately determine if a predictive model succeeds or fails. A popular example of feature engineering is the Titanic Competition, which challenges users who are part of an online community to use feature engineering and machine learning models to predict which passengers will survive the Titanic's sinking. (**Milwaukee School of Engineering, 2024)**

Add new features

Multiple meaningful features were added to the dataset using the "add_features" function. The average bedroom size and average bathroom size were calculated for each property by extracting the numbers in the size column and converting them to float values. In addition, the ratio of the number of balconies to the number of bedrooms was calculated, as well as the energy rating scores based on the BER scores, and the total price of the property (total square footage × price per square foot) was calculated. These new features provide more contextual information to the dataset and help improve the quality of subsequent analysis and modeling.

Numerical treatment of features

Using the “process_features” function, several category variables and string columns are converted to numeric variables for subsequent analysis and modeling. bedroom values in the size column are extracted and converted to floats, the buying or not buying column is converted to a binary variable (0, 1, 2), and BER is mapped to a numeric score. The Renovation needed and outlier columns were also converted to binary variables (0 and 1). After processing, the original category columns are removed, and the dataset becomes tidier and more suitable for machine-learning model inputs.

Predictive Analytics

Predictive analytics help us understand possible future occurrences by analyzing the past. At its core, predictive analytics includes a series of statistical techniques (including machine learning, predictive modeling, and data mining) and uses statistics (both historical and current) to estimate, or predict, future outcomes. **(Halton, C., 2024)**

Load to view two datasets

Fig.8

one before the outlier processing (df1) and the other after the outlier processing (df2). The number of rows and columns of both datasets are displayed, which helps to confirm that the datasets are sized and processed correctly. With this shape information, the performance of the two datasets during model training can be further analyzed and compared.

Data partitioning, partitioning the training and test sets

By populating the numeric column with the missing value of median and the categorical column with the missing value of plural. At the same time, it removes unnecessary columns (e.g., ID, property_scope, availability, location) and puts the

target variables (buying_or_not_buying and price-per-sqft-$) as categorical and regression target variables, respectively. Then, the feature sets (X1 and X2) are separated from the target variables (y1_class, y2_class, y1_reg, and y2_reg), and the dataset is partitioned into a training set and a test set for classification and regression models, respectively.

Two classification models, Random Forest, and Logistic Regression, were evaluated by Grid Search, which set hyperparameter ranges for each model, and cross-validation (cv=5) was used to find the best model hyperparameter combinations. For each model, the model is trained using the best hyperparameters and evaluated using a test set, outputting a classification report and a confusion matrix. Visualization of the confusion matrix helps to visualize the accuracy, recall, and precision of the model predictions.

Grid Search was also used to evaluate two kinds of regression models: Random Forest Regressor) and Linear Regression). Cross-validation was also used to find the best combination of parameters During training, the model was trained using the best hyperparameters, and the regression model was evaluated using a test set that outputs the Mean Squared Error (MSE) and R-squared (R^2). In addition, the relationship between predicted and actual values is shown in a scatter plot to help visualize how well the model fits.

Prediction results

1) Classification model analytics results

Evaluate the performance of the classification model for both datasets (Dataset 1 and Dataset 2) and output a classification report and visualization of the confusion matrix. This allows comparison of the differences in model performance between the two datasets in the classification task.

The results of the classification model evaluation on the two datasets (Dataset 1 and Dataset 2) show different model performances, in particular different classification results, especially in the case of category imbalance. The following is a more detailed analysis:

Random Forest:

Precision: It correctly predicts most of the samples in the non-purchase category. However, for category 1 (purchased properties), the precision drops to 0.49 (Dataset 1) and 0.48 (Dataset 2), and while the model can identify some of the Category 1

samples, it performs poorly in identifying Category 1. Combined with the F1-Score, further demonstrates the model's inadequacy in recognizing category 1.

Logistic Regression:

Accuracy: In both Dataset 1 and Dataset 2, the accuracy of Logistic Regression is 0.68, indicating that category 0, it is more stable in its prediction. Although it works better on the prediction of category 0, the F1-Score score is 0 and the recall is 0, indicating that the model predicts category 1 extremely poorly and the overall performance is limited.

2) Regression model analytics results

Random Forest algorithm is a powerful tree learning technique in Machine Learning. It works by creating a number of Decision Trees during the training phase. Each tree is constructed using a random subset of the data set to measure a random subset of features in each partition. This randomness introduces variability among individual trees, reducing the risk of overfitting and improving overall prediction performance. **(GeeksforGeeks, 2024)**

Linear Regression is a key data science tool for predicting continuous outcomes. This guide explains its principles, uses, and how to implement it in Python with real data. It covers simple and multiple linear regression, highlighting their importance, limitations, and practical examples. **(Mali, K., 2021)**

Dataset 1 - Regression Model Evaluation Results:

Random Forest

Mean Square Error (MSE): 5242031845, a higher MSE indicates that the model has a large error in prediction. R-squared (R^2):

0.17, indicating that the model is a poor fit, explaining only 17% of the data variability and that the model has limited predictive power.

Linear Regression

Mean Square Error: 656897432.66, although the MSE is low, this value is still large, implying that the model's predictions are still highly biased. R^2: 0.90, a better performance, the linear regression can explain about 90% of the data variability, indicating that the model fits the data relatively well.

Dataset 2 - Regression Model Evaluation Results:

Random Forest

Mean Square Error: 237.98, MSE is significantly smaller, indicating that the model has a small prediction error. R^2: 0.997, indicating that the model is very effective in explaining data variability and fits the data almost perfectly.

Linear Regression

Mean Square Error: 26871.68, with a low MSE, indicates that the linear regression has a good fit, but still has some error. R^2: 0.69, indicating that linear regression is able to explain about 69% of the variability in the data, which is a good performance, but not yet at the level of Random Forest.

Conclusion

The random forest regression model is more sensitive to outliers but is a more suitable regression method for this task as it can better fit complex relationships after data cleaning.

The linear regression model performs well with global linear relationships but is weaker with outliers or nonlinear relationships.

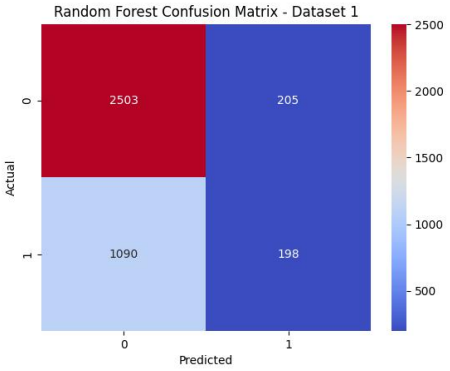
Data cleaning (especially outlier handling) significantly affects the performance of the regression model.The regression results of Dataset 2 are much better than those of Dataset 1, suggesting that reasonable handling of outliers can improve the explanatory power and prediction accuracy of the model.

The random forest model is suitable for predicting complex price trends or analyzing home-buying behavior in the Irish property market, but it needs to be coupled with data cleaning and feature engineering.

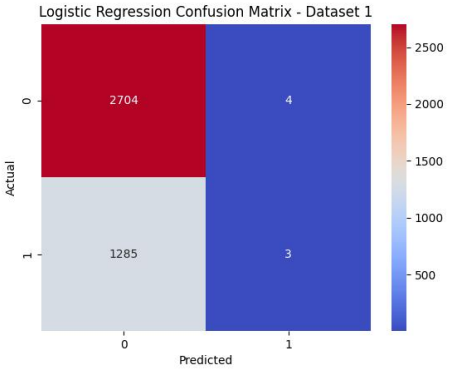
Visualization

Classification model

```
Classification Models Evaluation for Dataset 1

Training Random Forest...

Random Forest Classification Report:
              precision    recall  f1-score   support

           0       0.70      0.92      0.79      2708
           1       0.49      0.15      0.23      1288

    accuracy                           0.68      3996
   macro avg       0.59      0.54      0.51      3996
weighted avg       0.63      0.68      0.61      3996
```

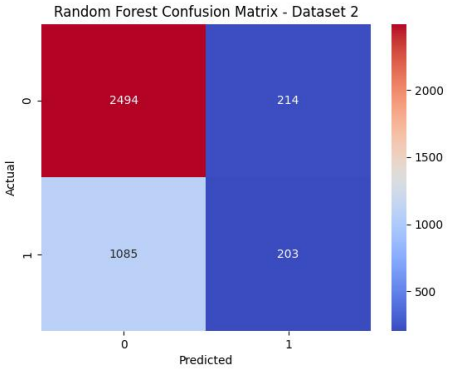



```
Training Logistic Regression...

Logistic Regression Classification Report:
              precision    recall  f1-score   support

           0       0.68      1.00      0.81      2708
           1       0.43      0.00      0.00      1288

    accuracy                           0.68      3996
   macro avg       0.55      0.50      0.41      3996
weighted avg       0.60      0.68      0.55      3996
```

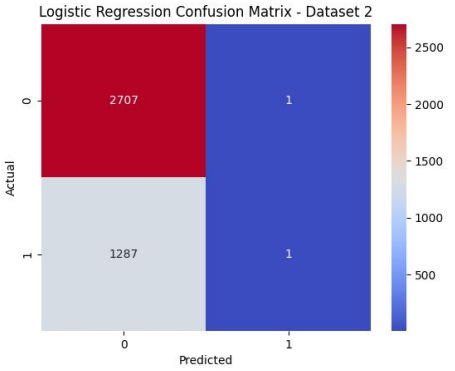


Dataset 2

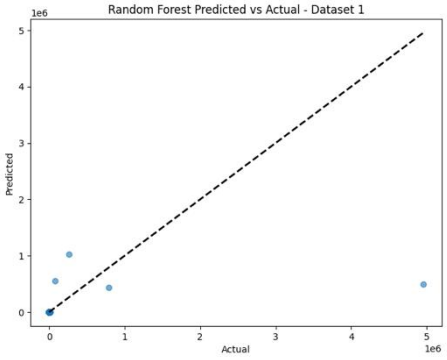
```
Classification Models Evaluation for Dataset 2

Training Random Forest...

Random Forest Classification Report:
              precision    recall  f1-score   support

           0       0.70      0.92      0.79      2708
           1       0.49      0.16      0.24      1288

    accuracy                           0.67      3996
   macro avg       0.59      0.54      0.52      3996
weighted avg       0.63      0.67      0.61      3996
```

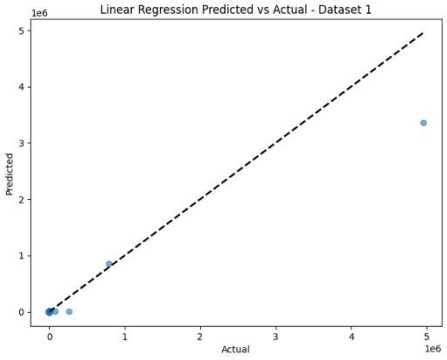


```
Training Logistic Regression...

Logistic Regression Classification Report:
              precision    recall  f1-score   support

           0       0.68      1.00      0.81      2708
           1       0.50      0.00      0.00      1288

    accuracy                           0.68      3996
   macro avg       0.59      0.50      0.40      3996
weighted avg       0.62      0.68      0.55      3996
```



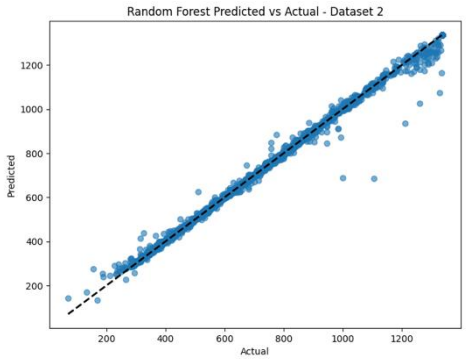
Regression Model

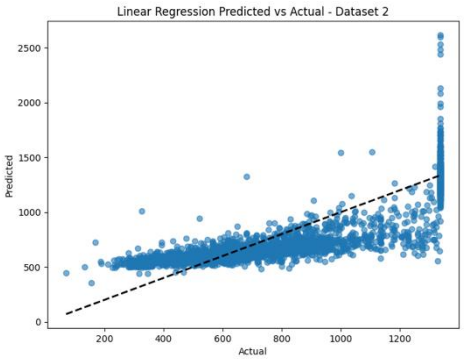





```
Training Random Forest...

Random Forest Regression Performance:
Mean Squared Error: 5220518701.914054
R-squared: 0.17281763439705666
```

```
Training Linear Regression...

Linear Regression Regression Performance:
Mean Squared Error: 656897432.5733438
R-squared: 0.8959157119702884
```






```
Training Random Forest...

Random Forest Regression Performance:
Mean Squared Error: 240.21485528924632
R-squared: 0.9972220607118695
```

```
Training Linear Regression...

Linear Regression Regression Performance:
Mean Squared Error: 26871.684618317013
R-squared: 0.6892452452634912
```

References

GeeksforGeeks (2024). *Random forest algorithm in machine learning*. [online] GeeksforGeeks. Available at: https://www.geeksforgeeks.org/random-forest-algorithm-in-machine-learning/.

Halton, C. (2024). *Predictive Analytics Definition*. [online] Investopedia. Available at: https://www.investopedia.com/terms/p/predictive-analytics.asp.

Kowieski, J. (2022). *What is Data Cleaning? Examples and How to Clean Your Data*. [online] ThoughtSpot. Available at: https://www.thoughtspot.com/data-trends/data-science/what-is-data-cleaning-and-how-to-keep-your-data-clean-in-7-steps.

Mali, K. (2021). *Linear Regression | Everything you need to Know about Linear Regression*. [online] Analytics Vidhya. Available at: https://www.analyticsvidhya.com/blog/2021/10/everything-you-need-to-know-about-linear-regression/.

Online Degree Programs | Milwaukee School of Engineering. (2024). *The Importance of Feature Engineering in Machine Learning*. [online] Available at: https://online.msoe.edu/engineering/blog/importance-of-feature-engineering-in-machine-learning.

Sharma, R. (2024). *Role of EDA in data science - The Pythoneers - Medium*. [online] Medium. Available at: https://medium.com/pythoneers/role-of-eda-in-data-science-f8b6fa9bf462.

AI supervisory

https://chatgpt.com/share/674d03a6-56ac-8008-8575-9466faa5098a