Melbourne Housing Business Analytics

Milan Mitrovic | s4663796

Contents

INTRODUCTION

In the fiercely competitive realm of real estate consulting, securing a distinct advantage is paramount for success, the proposed solution: harnessing the power of business analytics & data mining. Employed as a business analyst in a real estate consulting firm with the goal of maximizing the firm's organisational performance with the provided Melbourne housing dataset to be used for leverage with business analytics. This report's objective is to conduct our business analysis task, gather the key insights from our business analysis task, and use these key insights to optimize the firm's organisational performance. The business analysis task used in this report will be predicting the price of properties, based on the features of the property using regression. We will determine the best suited algorithms for doing such a task, which in turn will drive numerous business benefits and have significant positive implications for our real estate consulting firm. We will utilize multiple machine learning algorithms to predict the price of properties based on given (Melbourne housing data) and unseen data. Machine learning algorithms will play a crucial role in price prediction by leveraging the Melbourne housing dataset to identify patterns, features, and risk factors associated with price fluctuation in properties, enabling the forecast of a properties price. This business analysis task will primarily have key benefits for decision-making when it comes to property investments; along with providing other substantial organisational improvements. Ultimately, the business analysis task will help our firm find opportunities in data, predict trends that forecast future opportunities, and aid in selecting a course of action that optimizes the firm's allocation of resources to maximize value and performance.

BUSINESS ANALYSIS TASK

Regression analysis will be conducted to predict property prices based on the features within the provided Melbourne housing dataset. For example, regression analysis would be used to predict the price of a property based on its features such as the number of rooms, bathrooms, distance from CBD, land size, building area, etc. By feeding regression models all of the associated features of properties within the dataset, it will subsequently identify patterns in property pricing over time and predict future price movements; the main benefit with this is the valuable key insights you could extract and in turn make informed decisions about property investments. Additionally, you could utilize these key insights to identify what exact features influence property prices the most: this information is also beneficial to the cause of property valuation and pricing strategies. Essentially, regression analysis allows our real estate consulting firm to maximize returns, optimize strategies, and overall enhance our real estate consulting firm's organizational performance.

By finishing this business analysis task these business benefits are expected:

Regression Business Benefits:

1. **Informed Investment Decisions**: regression property price prediction models provide valuable insights for buyers and sellers, facilitating informed investment decisions with accurate estimations of property values.
2. **Market Trend Analysis and Forecasting**: by analyzing our data, regression models reveal past price patterns and forecast future market shifts, empowering our real estate consulting firm to foresee changes and adjust strategies accordingly.
3. **Identification of Key Features Influencing Price**: regression analysis identifies the most influential features impacting property prices, allowing our real estate consulting firm to focus on optimizing these areas to boost returns and profitability.
4. **Pricing Strategy Optimization:** garnering insights into property pricing trends facilitates the firm to establish competitive and appealing property prices. By analyzing the features influencing property prices, the firm can formulate data-driven pricing strategies that draw in buyers and increase profit margins.
5. **Optimizing Resource Allocation:** predictive insights aid in optimizing resource allocation by identifying the most valuable property features and regions. This improves the efficiency of marketing efforts, ensuring that resources are targeted towards the most promising opportunities.
6. **Enhanced Operational Efficiency:** predictive models entail automation of pricing predictions and market trend analysis; streamlining operations, greatly reducing the time and effort required for manual analysis. This improves operational efficiency, allowing the firm to focus more on strategic initiatives.
7. **Risk Management:** predictive models can pinpoint potential risks by identifying market volatility and unstable property investments. This facilitates the development of risk mitigation strategies, enabling the firm to manage uncertainties and safeguard client investments.
8. **Competitive Advantage:** employing regression analysis for price prediction and market analysis provides a significant edge over competitors that don’t utilize data-driven methodologies. This advantage leads to improved market positioning and increased client acquisition rates.

In essence, condudcting regression analysis on the Melbourne housing dataset offers numerous benefits to the real estate consulting firm, including informed investment decisions, market trend forecasting, identification of key price-influencing features, optimized pricing strategies, enhanced resource allocation, improved operational efficiency, effective risk management, and competitive advantage. These benefits enable the firm to maximize returns, mitigate risks, and maintain a leading position in the real estate market.

METHODOLOGIES/DATA PRE-PROCESSING

Data mining algorithms used to solve the business analysis task:

1. **Linear Regression** – linear regression is a supervised machine learning algorithm, which is used for predicting numerical target features. It models the relationship between the dependent variable (target feature/y) and the independent variable (all other features/X). It presumes a linear relationship between the target feature and all the other features. This linear relationship describes how changes in the features are associated with changes in the target feature. In this context our target feature would be Price and our independent variables would be every other feature within the Melbourne housing dataset. Subsequently meaning we are employing multiple linear regression as our type of linear regression, as we are using multiple features (independent variables) to predict Price (target feature). This is the equation for multiple linear regression, which represents the relationship in a straight line:

$$y = b_0 + b_1 x_1 \ b_2 x_2 + \ldots + b_n x_n$$

- y represents the price target feature.
- x_1, x_2, etc. represents all features besides Price used for the prediction: (Rooms, Bathroom, Distance, etc.).
- b_0 represents the intercept form (It represents the predicted value of the dependent variable y (Price) when all the independent variables X (Rooms, Bathrooms, Distance, etc.) are zero.
- $b_{1,}$ b_2 represent the coefficients/slope for each of the features besides price. b_1 represents the change in the dependent variable *y* (Price) for a one-unit change in the independent variable $X_{1,}$ $X_{2,}$ and so on (Rooms, Bathrooms, Distance, etc.) holding all other variables constant. For example, if b_1 is the coefficient for Rooms, then b_1 signals how much the price of a property is expected to change with an additional room. The same process applies for every other feature in X.

2. **Gradient Boosting Regression (GBR)** – GBR is a supervised machine learning algorithm utilized to predict numerical target features. In the context of the Melbourne housing dataset, GBR aims to predict the target variable (Price), based on the independent variables encompassed within the dataset (all other features). GBR functions by iteratively constructing an ensemble of decision trees that collectively minimize the prediction errors between the forecasted property prices and their actual price. The gradient boosting process focuses on learning the optimal combination of trees by iteratively minimizing a loss function, typically the mean squared error, between the predicted prices and the actual prices of the properties.

3. **Random Forest Regression** – random forest regression is a supervised machine learning algorithm, employed for predicting numerical target features. It works by constructing a tree-like structure, which means it moves like a flow chart, separating each data point into more and more similar groups until it reaches a defined limit, running many individual decision trees at once and makes a prediction based on the most popular result. Again, it predicts the dependent variable (Price) which is our target feature, by leveraging the independent variables (all other features). The property price prediction is the output of the typical average of every decision tree.

I select all the models above to solve the business analysis task because they are designed to work specifically for regression analysis - predicting a numerical target feature (property price), based on the property's characteristics (all other features besides price) The models are perfectly compatible and compliant with my business analysis task. Each model has its own unique strengths:

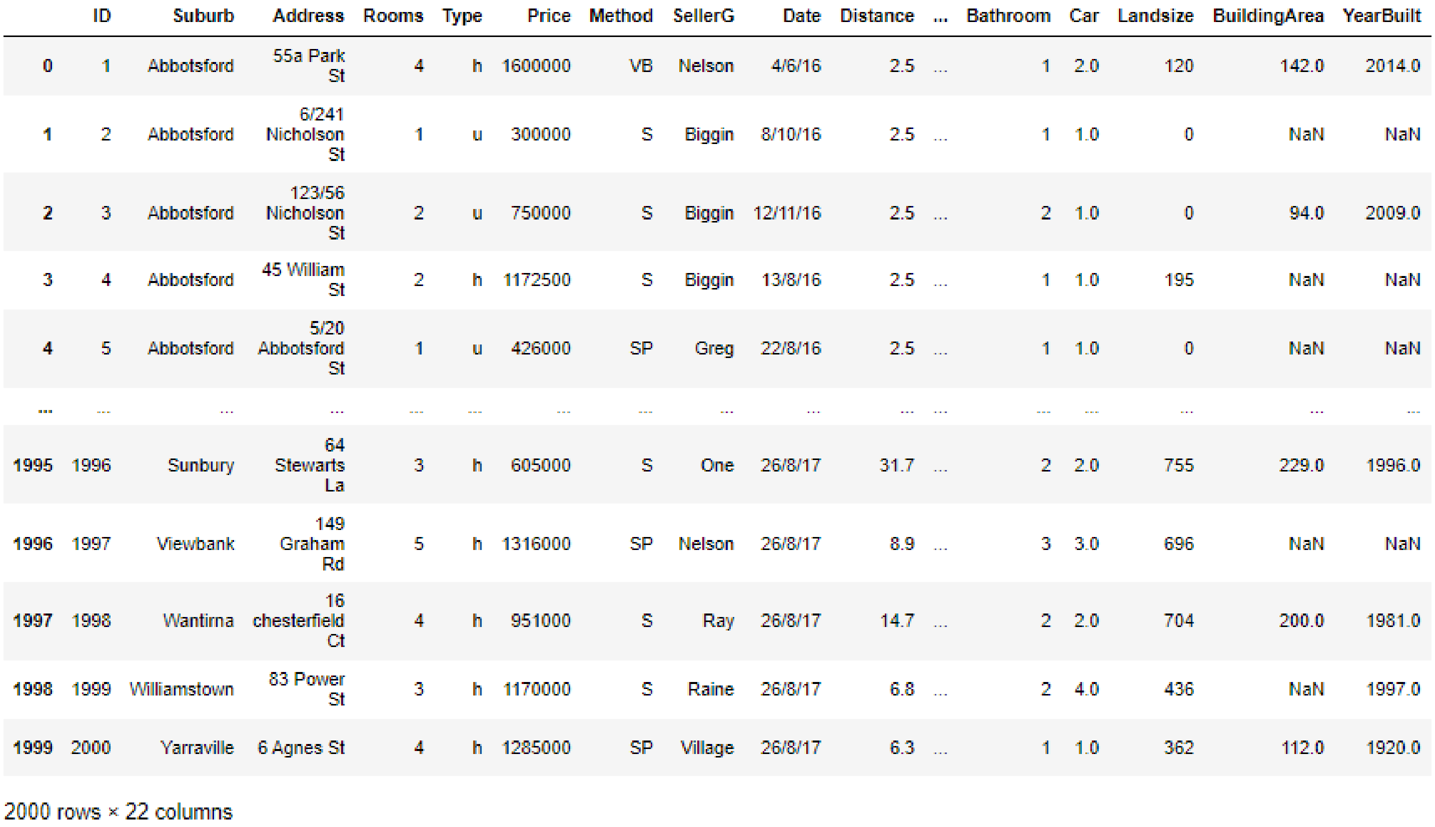
- **Linear Regression**: Could reveal the simple linear trends in property prices with all features (coefficients), such as the increase in price per additional room.
- **Random Forest Regression**: Can highlight the relative importance of all features to property prices, whilst providing accurate price predictions.
- **Gradient Boosting Regression (GBR)**: Great at capturing complex patterns in data. It learns from its mistakes, prioritizes important features, and gives precise predictions, making it great for understanding what drives property prices.

Dataset Description

The dataset selected for conducting the business analysis task is the Melbourne housing dataset, encompassing 22 features: ID, Rooms, Price, Method, Type, Seller, Date, Distance, Region Name, Property Count, Bedroom, Bathroom, Car, Land Size, Building Area, Council Area, Suburb, Address, Year Built, Latitude, and Longitude. The 'Price' feature serves as the numeric target variable for regression analysis, representing the price of a property. By utilizing supervised machine learning regression algorithms such as linear regression, random forest regression, and Gradient Boosting Regression (GBR), we aim to predict property prices by leveraging the relationships between the target variable (Price) and the independent variables (all other features) in the dataset. Through training these models with the provided data, we enable them to learn and capture the underlying patterns and relationships, facilitating predictions of property pricing on unseen data.

Data Cleaning

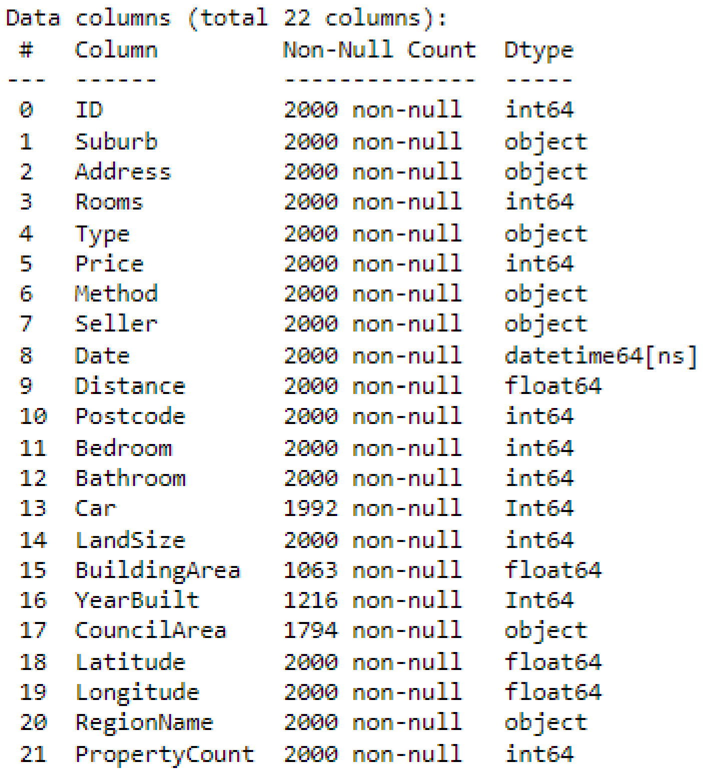
Before starting the modelling phase, we need to prepare and clean the data, as dirty data can lead to misleading results and inaccurate conclusions.



	ID	Suburb	Address	Rooms	Type	Price	Method	SellerG	Date	Distance	...	Bathroom	Car	Landsize	BuildingArea	YearBuilt
0	1	Abbotsford	55a Park St	4	h	1600000	VB	Nelson	4/6/16	2.5	...	1	2.0	120	142.0	2014.0
1	2	Abbotsford	6/241 Nicholson St	1	u	300000	S	Biggin	8/10/16	2.5	...	1	1.0	0	NaN	NaN
2	3	Abbotsford	123/56 Nicholson St	2	u	750000	S	Biggin	12/11/16	2.5	...	2	1.0	0	94.0	2009.0
3	4	Abbotsford	45 William St	2	h	1172500	S	Biggin	13/8/16	2.5	...	1	1.0	195	NaN	NaN
4	5	Abbotsford	5/20 Abbotsford St	1	u	426000	SP	Greg	22/8/16	2.5	...	1	1.0	0	NaN	NaN
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
1995	1996	Sunbury	64 Stewarts La	3	h	605000	S	One	26/8/17	31.7	...	2	2.0	755	229.0	1996.0
1996	1997	Viewbank	149 Graham Rd	5	h	1316000	SP	Nelson	26/8/17	8.9	...	3	3.0	696	NaN	NaN
1997	1998	Wantirna	16 chesterfield Ct	4	h	951000	S	Ray	26/8/17	14.7	...	2	2.0	704	200.0	1981.0
1998	1999	Williamstown	83 Power St	3	h	1170000	S	Raine	26/8/17	6.8	...	2	4.0	436	NaN	1997.0
1999	2000	Yarraville	6 Agnes St	4	h	1285000	SP	Village	26/8/17	6.3	...	1	1.0	362	112.0	1920.0

2000 rows × 22 columns

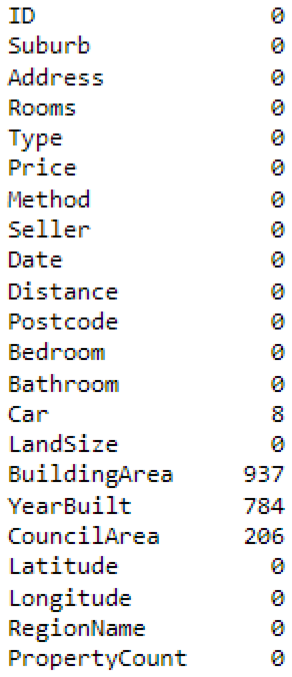
Checking the dimensions of the dataset (2000 rows, 22 columns), as well as removing duplicates in case they were present (there weren't any).



```
Data columns (total 22 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   ID             2000 non-null   int64
 1   Suburb         2000 non-null   object
 2   Address        2000 non-null   object
 3   Rooms          2000 non-null   int64
 4   Type           2000 non-null   object
 5   Price          2000 non-null   int64
 6   Method         2000 non-null   object
 7   Seller         2000 non-null   object
 8   Date           2000 non-null   datetime64[ns]
 9   Distance       2000 non-null   float64
 10  Postcode       2000 non-null   int64
 11  Bedroom        2000 non-null   int64
 12  Bathroom       2000 non-null   int64
 13  Car            1992 non-null   Int64
 14  LandSize       2000 non-null   int64
 15  BuildingArea   1063 non-null   float64
 16  YearBuilt      1216 non-null   Int64
 17  CouncilArea    1794 non-null   object
 18  Latitude       2000 non-null   float64
 19  Longitude      2000 non-null   float64
 20  RegionName     2000 non-null   object
 21  PropertyCount  2000 non-null   int64
```

Changing the names of certain columns (SellerG to Seller, Bedroom2 to Bedroom, Landsize to LandSize, Regionname to RegionName, and Propertycount to PropertyCount, Lattitude to Latitude, Longtitude to Longitude) to correct spelling issues and follow the proper naming convention of this dataset (pascal case). Also, changing the data types of certain features to its appropriate data types: Date from object to datetime64, Car from float to integer, and YearBuilt from float to integer. I changed the data type of YearBuilt to its suitable data type integer as years are whole numbers and should not be represented as floating-point numbers, increasing conciseness, and ensuring consistency. For Date I changed it to its appropriate data type method of datetime64, this is done so we have the option to do time-based analysis, which was not possible with the previous object data type, there is also increased clarity. Finally, the Car feature represents the count of parking spots, it should ideally be an integer data type since you cannot have a fraction of a parking spot.

There also appears to be null values in Car, BuildingArea, YearBuilt, and CouncilArea. We will have to deal with these null values.



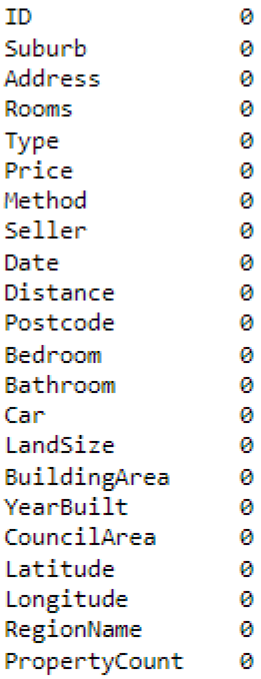
```
ID                 0
Suburb             0
Address            0
Rooms              0
Type               0
Price              0
Method             0
Seller             0
Date               0
Distance           0
Postcode           0
Bedroom            0
Bathroom           0
Car                8
LandSize           0
BuildingArea     937
YearBuilt        784
CouncilArea      206
Latitude           0
Longitude          0
RegionName         0
PropertyCount      0
```

8 null values in Car, 937 null values in BuildingArea, 784 null values in YearBuilt and 206 null values in CouncilArea. There are many missing values present. Significant missing values in BuildingArea, YearBuilt, and CouncilArea. Insignificant missing values in Car. I deal with the insignificant and significant null values as follows:

- Insignificant null values: I deal with the insignificant null values in Car by removing the 8 rows which contain null values. I decided to do this as it was only 8 rows removed in total, which equates to 0.4% of the entire dataset: insignificant, maintaining data integrity and quality – removing nulls.

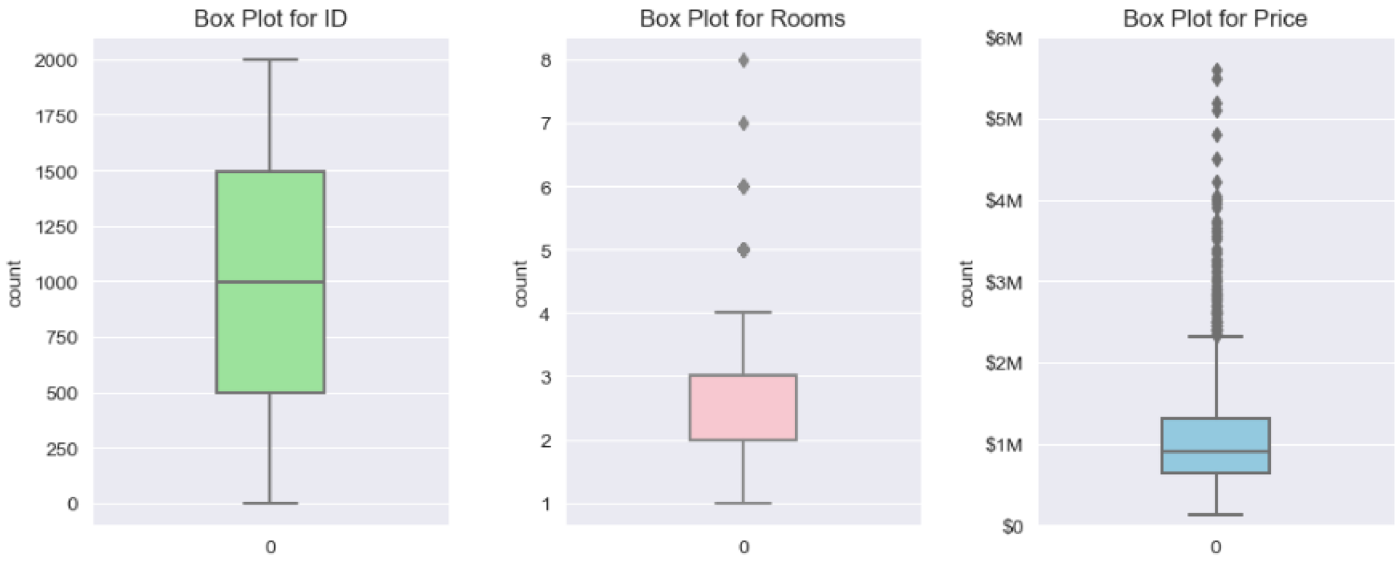
- Significant null values: I deal with the significant null values in BuildingArea by replacing them with the median values from its column. I do this to preserve the 937 rows or approximately 47% of the dataset – which is significant. I choose median here specifically because its robust to outliers and

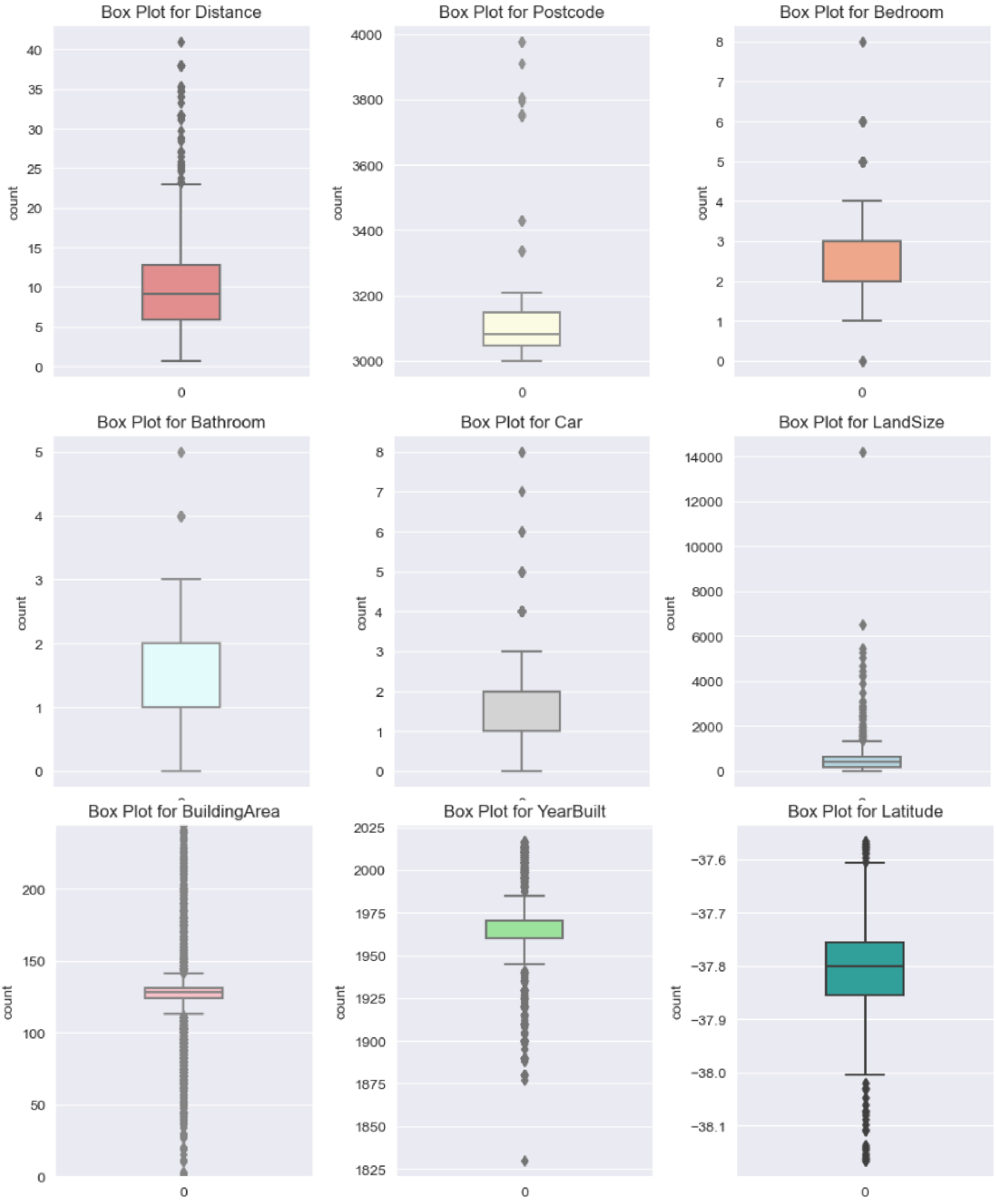
BuildingArea contains outliers; overall, this approach eliminates the null values, increasing data integrity and quality. For YearBuilt we replaced the null values with the mode of its column. This is done to deal with the nulls, maintain data integrity and quality, as well as preserving 784 rows or around 39.4% of the dataset. Finally, for CouncilArea we also replaced the 206 null values with the mode of itself. Again, this is done to preserve the data frame (10.35% of the data frame) and enhance the data integrity and quality of the dataset by getting rid of the nulls. We opt for mode here because CouncilArea is a nominal attribute, and the mode is the most appropriate measure here, as mean or median is not applicable to nominal attributes. This process was a necessity as the null values would have distorted the data – negatively impacting the models performance.



```
ID               0
Suburb           0
Address          0
Rooms            0
Type             0
Price            0
Method           0
Seller           0
Date             0
Distance         0
Postcode         0
Bedroom          0
Bathroom         0
Car              0
LandSize         0
BuildingArea     0
YearBuilt        0
CouncilArea      0
Latitude         0
Longitude        0
RegionName       0
PropertyCount    0
```

Now that I've addressed the null values in the dataset, it's time to investigate the outliers in all numerical columns. This will be achieved by data visualization techniques — specifically boxplots. Boxplots displays several key descriptive statistics, including the median, quartiles, and potential outliers, in a concise and easy-to-understand manner. However, I will only be focusing on outliers here as this is a part of the data cleaning process.





Box Plot for Distance

Box Plot for Postcode

Box Plot for Bedroom

Box Plot for Bathroom

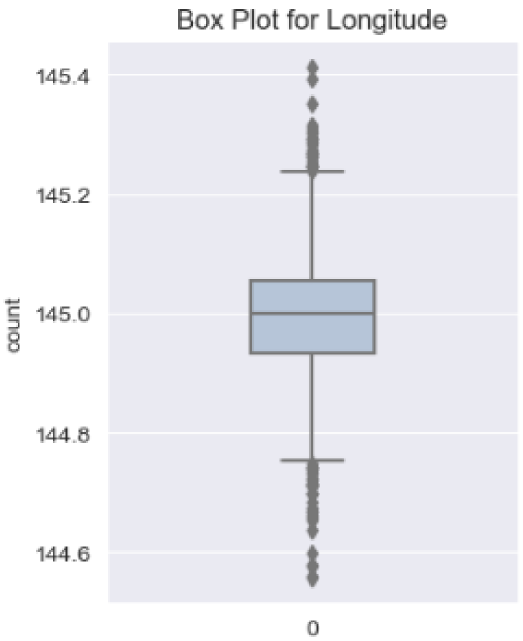
Box Plot for Car

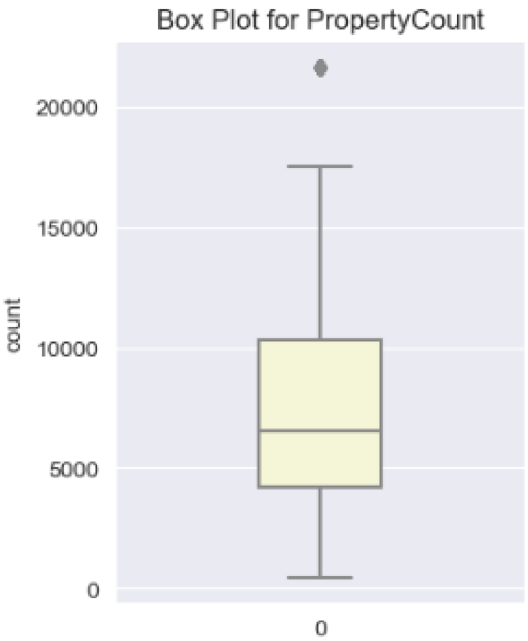
Box Plot for LandSize

Box Plot for BuildingArea

Box Plot for YearBuilt

Box Plot for Latitude

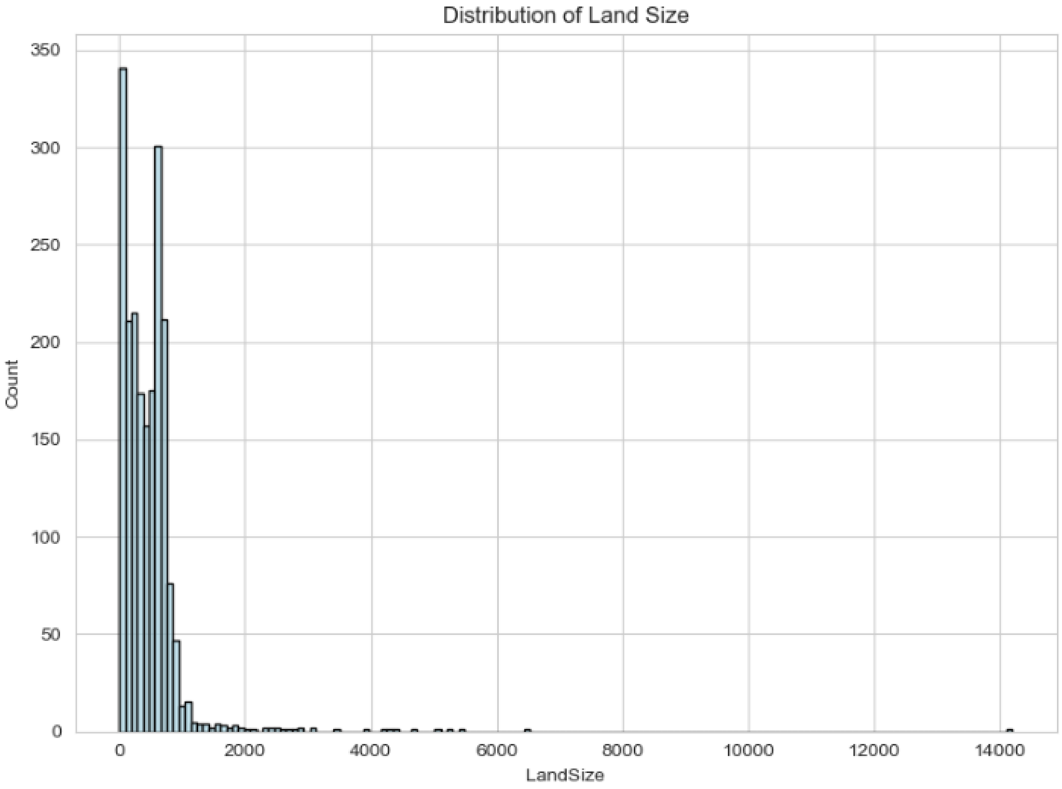


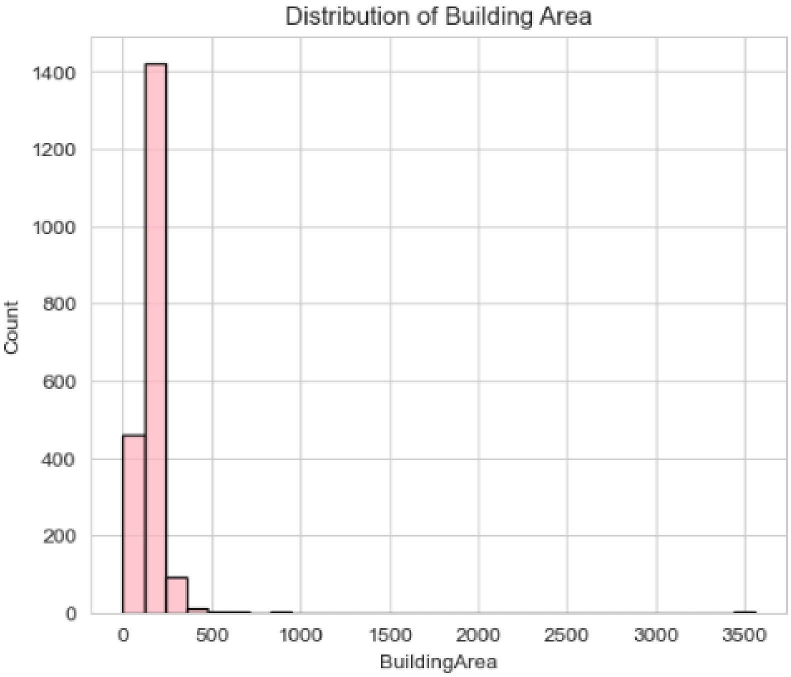


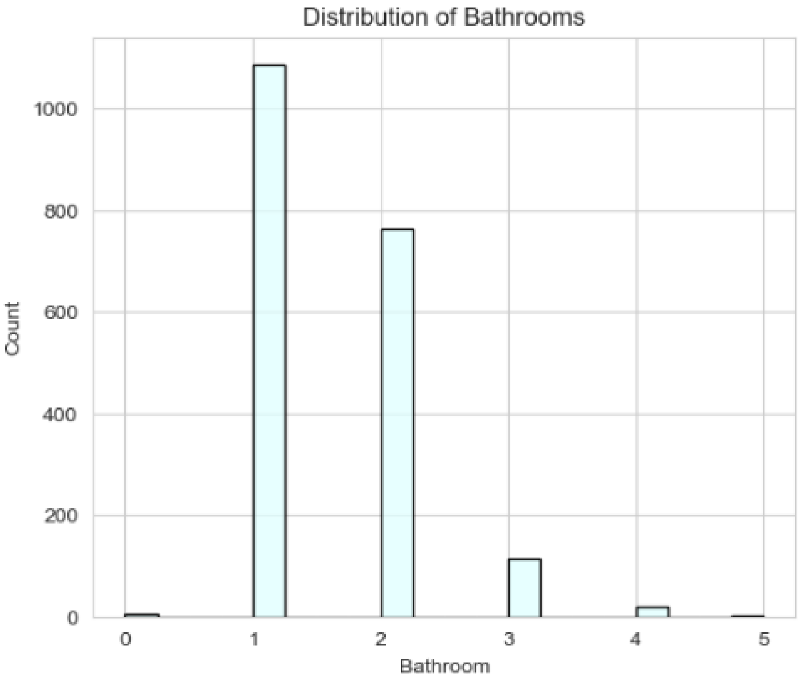
(Rooms, Bedroom, Bathroom, Car, YearBuilt boxplot medians aren't showing properly because their medians and Q1/Q3 scores are equal – Q1/Q3 is blocking the median).

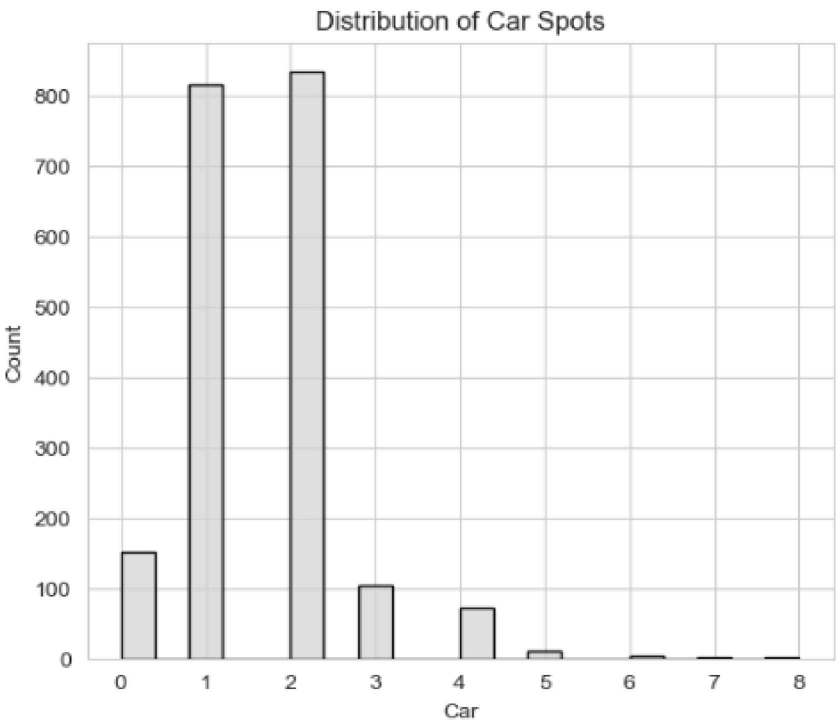
Clearly, apart from ID there are outliers present in every other attribute. The outliers in Rooms, Price, Distance, Postcode, YearBuilt, Latitude, Longitude, and PropertyCount all appear to be valid extreme values as they are realistic. Bedroom, Bathroom, and Car upper outliers also appear to be valid extreme values as 8 bedrooms, 5 bathrooms, and 8 car spots are feasible. We can conclude that the outliers covered so far are valid values that aren't errors. However, the extreme upper outliers for LandSize and BuildingArea are questionable, at a land size of 14196 square meters and a building size of 3558 square meters (which is not visible in the BuildingArea boxplot because the disparity between the max value/outlier and the other corresponding values of the boxplot are so vast the boxplot ceases to display correctly if we were to plot y ticks to 3600). The LandSize extreme upper outlier (14196m^2) is actually a valid extreme value – I verified the addresses land size (52/73 River St, Richmond). However, the BuildingArea extreme upper outlier (3558m^2) is a data entry error, I checked the address (186 Queens Pde, Fitzroy North) and validated the building size at 145m^2 – I will deal with this outlier once we have finalized the outlier/0-value validation process.

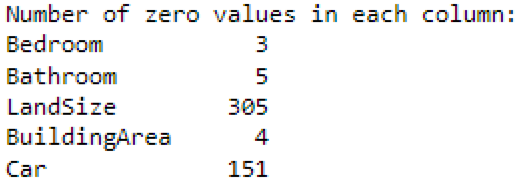
Bedroom and BuildingArea seem to contain 0-value outliers which doesn't make sense and needs more investigating. The 0-values for Bathroom, Car, and LandSize are not showing up because the boxplots lower whisker range starts at 0 for these features, so these 0-values are not being displayed as lower outliers by default because of this. This is a problem because we can see these columns contain 0-values as minimum values which are clearly errors. We will display these particular features (Bathroom, Car, LandSize) in histograms instead to avoid this, so we can better visualize the errors in box plot as visualizing the highest upper outlier in BuildingArea, which wasn't possible with the box plot for BuildingArea.


Distribution of Land Size


Distribution of Building Area


Distribution of Bathrooms





```
Number of zero values in each column:
Bedroom            3
Bathroom           5
LandSize         305
BuildingArea       4
Car              151
```

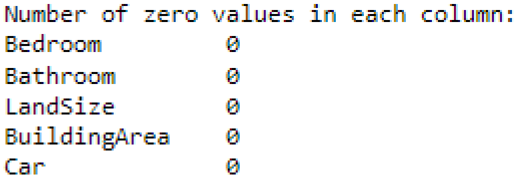
Now that we can visualize the extreme upper outlier in BuildingArea and 0-values in Bathrooms, Car, and LandSize we will begin with the final stages of the data cleaning process. By observing the above histograms or description of the zero values in each column: we can ascertain the number of 0-values in each column. Insignificant amounts in Bedroom (3), Bathroom (5), and BuildingArea (4). Significant amounts in LandSize (305) and Car (151).

We deal with the described issues above accordingly:

- **Insignificant outlier values/0-values:** We begin by removing the 3558m^2 extreme upper outlier we identified as an error from BuildingArea, along with its lower outlier 0-values, as well as the remaining 0-values in Bedroom, Bathroom. We do this because these 0-values are junk values which would skew the data; negatively impacting the data analysis. Essentially, we are removing noise as it is impossible for a house in urban Melbourne to have a 0 square meter building size, along with 0 bedrooms. The same way it is impossible for a house, unit, or townhouse to have 0 bathrooms – properties are expected to have at least 1 bathroom as bathrooms are considered essential features in properties, as they provide necessary facilities for personal hygiene and sanitation. Therefore, we can conclude these 0-values/outliers are data entry errors, and as they only total for 13 rows (less than 1% of the dataset) we go with the practical way of removing them.

- **Significant 0-values:** We decided to deal with the significant 0-values in LandSize and Car by replacing them with their own median. This is mainly done to preserve the dataset: 456 rows in total – approximately 23% of the total dataset. However, we also deal with the junk 0-values at the same time. In urban areas like Melbourne, Australia, it's highly unlikely for properties such as houses, units, or townhouses to have zero car spots, as parking spots are an essential amenity. Therefore, these 0-values in the Car column are data entry errors. Again, the same applies to LandSize: land size is a fundamental attribute of properties, it’s impossible for properties such as houses, units, or townhouses to have zero land size, as even the smallest properties would have some land size. - 0-values in the LandSize column are data entry errors.

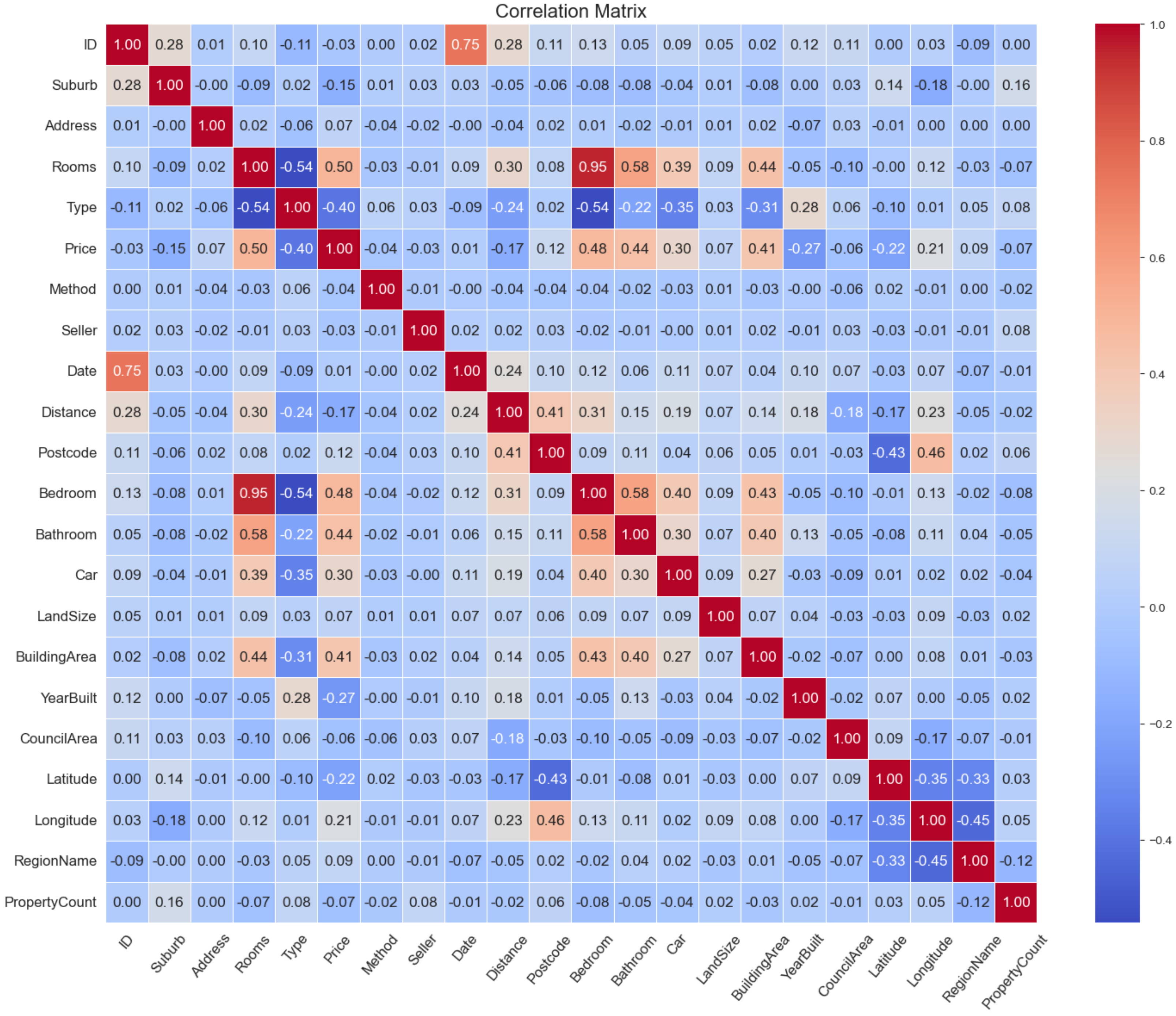
Overall, by dealing with these 0-values, we improve the data quality and integrity of the dataset, in addition we rectify the data entry 0-value errors (noise) which would have otherwise skewed the data, adversely affecting the modelling performance and the quality of the dataset.



```
Number of zero values in each column:
Bedroom         0
Bathroom        0
LandSize        0
BuildingArea    0
Car             0
```

Updated number of zero values in each column, we can clearly observe the outliers/0-values which were errors are removed. Now that the data cleaning process is over, we can now move on to the feature engineering phase.

Feature Engineering/Selection



Correlation Matrix

	ID	Suburb	Address	Rooms	Type	Price	Method	Seller	Date	Distance	Postcode	Bedroom	Bathroom	Car	LandSize	BuildingArea	YearBuilt	CouncilArea	Latitude	Longitude	RegionName	PropertyCount
ID	1.00	0.28	0.01	0.10	-0.11	-0.03	0.00	0.02	0.75	0.28	0.11	0.13	0.05	0.09	0.05	0.02	0.12	0.11	0.00	0.03	-0.09	0.00
Suburb	0.28	1.00	-0.00	-0.09	0.02	-0.15	0.01	0.03	0.03	-0.05	-0.06	-0.08	-0.08	-0.04	0.01	-0.08	0.00	0.03	0.14	-0.18	-0.00	0.16
Address	0.01	-0.00	1.00	0.02	-0.06	0.07	-0.04	-0.02	-0.00	-0.04	0.02	0.01	-0.02	-0.01	0.01	0.02	-0.07	0.03	-0.01	0.00	0.00	0.00
Rooms	0.10	-0.09	0.02	1.00	-0.54	0.50	-0.03	-0.01	0.09	0.30	0.08	0.95	0.58	0.39	0.09	0.44	-0.05	-0.10	-0.00	0.12	-0.03	-0.07
Type	-0.11	0.02	-0.06	-0.54	1.00	-0.40	0.06	0.03	-0.09	-0.24	0.02	-0.54	-0.22	-0.35	0.03	-0.31	0.28	0.06	-0.10	0.01	0.05	0.08
Price	-0.03	-0.15	0.07	0.50	-0.40	1.00	-0.04	-0.03	0.01	-0.17	0.12	0.48	0.44	0.30	0.07	0.41	-0.27	-0.06	-0.22	0.21	0.09	-0.07
Method	0.00	0.01	-0.04	-0.03	0.06	-0.04	1.00	-0.01	-0.00	-0.04	-0.04	-0.04	-0.02	-0.03	0.01	-0.03	-0.00	-0.06	0.02	-0.01	0.00	-0.02
Seller	0.02	0.03	-0.02	-0.01	0.03	-0.03	-0.01	1.00	0.02	0.02	0.03	-0.02	-0.01	-0.00	0.01	0.02	-0.01	0.03	-0.03	-0.01	-0.01	0.08
Date	0.75	0.03	-0.00	0.09	-0.09	0.01	-0.00	0.02	1.00	0.24	0.10	0.12	0.06	0.11	0.07	0.04	0.10	0.07	-0.03	0.07	-0.07	-0.01
Distance	0.28	-0.05	-0.04	0.30	-0.24	-0.17	-0.04	0.02	0.24	1.00	0.41	0.31	0.15	0.19	0.07	0.14	0.18	-0.18	-0.17	0.23	-0.05	-0.02
Postcode	0.11	-0.06	0.02	0.08	0.02	0.12	-0.04	0.03	0.10	0.41	1.00	0.09	0.11	0.04	0.06	0.05	0.01	-0.03	-0.43	0.46	0.02	0.06
Bedroom	0.13	-0.08	0.01	0.95	-0.54	0.48	-0.04	-0.02	0.12	0.31	0.09	1.00	0.58	0.40	0.09	0.43	-0.05	-0.10	-0.01	0.13	-0.02	-0.08
Bathroom	0.05	-0.08	-0.02	0.58	-0.22	0.44	-0.02	-0.01	0.06	0.15	0.11	0.58	1.00	0.30	0.07	0.40	0.13	-0.05	-0.08	0.11	0.04	-0.05
Car	0.09	-0.04	-0.01	0.39	-0.35	0.30	-0.03	-0.00	0.11	0.19	0.04	0.40	0.30	1.00	0.09	0.27	-0.03	-0.09	0.01	0.02	0.02	-0.04
LandSize	0.05	0.01	0.01	0.09	0.03	0.07	0.01	0.01	0.07	0.07	0.06	0.09	0.07	0.09	1.00	0.07	0.04	-0.03	-0.03	0.09	-0.03	0.02
BuildingArea	0.02	-0.08	0.02	0.44	-0.31	0.41	-0.03	0.02	0.04	0.14	0.05	0.43	0.40	0.27	0.07	1.00	-0.02	-0.07	0.00	0.08	0.01	-0.03
YearBuilt	0.12	0.00	-0.07	-0.05	0.28	-0.27	-0.00	-0.01	0.10	0.18	0.01	-0.05	0.13	-0.03	0.04	-0.02	1.00	-0.02	0.07	0.00	-0.05	0.02
CouncilArea	0.11	0.03	0.03	-0.10	0.06	-0.06	-0.06	0.03	0.07	-0.18	-0.03	-0.10	-0.05	-0.09	-0.03	-0.07	-0.02	1.00	0.09	-0.17	-0.07	-0.01
Latitude	0.00	0.14	-0.01	-0.00	-0.10	-0.22	0.02	-0.03	-0.03	-0.17	-0.43	-0.01	-0.08	0.01	-0.03	0.00	0.07	0.09	1.00	-0.35	-0.33	0.03
Longitude	0.03	-0.18	0.00	0.12	0.01	0.21	-0.01	-0.01	0.07	0.23	0.46	0.13	0.11	0.02	0.09	0.08	0.00	-0.17	-0.35	1.00	-0.45	0.05
RegionName	-0.09	-0.00	0.00	-0.03	0.05	0.09	0.00	-0.01	-0.07	-0.05	0.02	-0.02	0.04	0.02	-0.03	0.01	-0.05	-0.07	-0.33	-0.45	1.00	-0.12
PropertyCount	0.00	0.16	0.00	-0.07	0.08	-0.07	-0.02	0.08	-0.01	-0.02	0.06	-0.08	-0.05	-0.04	0.02	-0.03	0.02	-0.01	0.03	0.05	-0.12	1.00

This is a correlation matrix between all features within the dataset. In every row you can observe said features correlation against every other feature. The closer the number is to 1 (or -1), the stronger the positive or negative correlation is. I won't go through every feature here as there is a lot, however we will target the most important feature in this dataset, which is related to our business analysis task, our target feature – Price.

- Rooms, Bedroom, Bathroom, BuildingArea, Car, Longitude; have a positive correlation with Price — with rooms having the highest rate of positive correlation at 0.50.
- Address, Postcode, LandSize, RegionName; have a weak/low positive correlation with Price.
- ID, Method, Seller, and Date, CouncilArea have no correlation with Price.
- CouncilArea, PropertyCount; have a weak/low negative correlation with Price.
- Type, YearBuilt, Latitude, Distance, Suburb; have a negative correlation with Price – with Type having the highest rate of negative correlation at -0.40.

For feature selection we are going to remove irrelevant features which would hinder the models performance:

- **ID -** the column ID is not needed: the feature is irrelevant as it does not provide any valuable information for analysis nor is it beneficial for the modelling process. There is no correlation with our target feature Price.

- **Address -** unique identifier for every property and does not provide useful information for predicting the price in a numerical sense that supervised machine learning models can interpret, it is similar to ID in that sense. Also, non-existent correlation with the target feature Price.

- **Method -** this feature indicates the method of sale (property sold, sold after auction, etc.). The effect of this feature on the price of a property would be extremely minimal, if any, as it is more related to the sale procedure rather than the property's value, which is evident in the correlation between our target feature Price and Method – there is none.

- **Seller -** The real estate agent's name is irrelevant towards the prediction of a property's price, as it does not directly affect the property's value. Which is evident when looking at the correlation between Price and Seller (-0.03): no correlation at all.

- **Date –** doesn't provide enough relevant predictive information for the current price of a property in this context. The date a property was sold can introduce noise if not properly engineered, skewing results of the model. There also is no correlation present between Date and Price.

- **CouncilArea –** this feature is irrelevant when it comes to the prediction of a property's price, as the information it provides is not significant or influential to the price of a property at all. This is backed by the correlation between it and Price. (-0.06 – basically no correlation).

Overall, this is mainly done to improve the models performance. By refining our dataset in this way, we are ensuring that only the most relevant and useful information is used to build our regression models - maximizing our models evaluation metrics.

MODELLING/RESULTS

Before we start the modelling phase, I want to emphasize the difference and importance of the training and testing set scores and the evaluation metrics our models are measured in. The testing set metrics represent the model's performance on predicting unseen data, simulating real-world scenarios on predicting property prices — more importance placed on testing set. However, the training set metrics indicate the performance on how well the model fits the training data. Essentially, the scores indicate how accurately the model can predict the prices of properties within the original dataset. Less importance on the training set since it does not represent real-world scenarios however, it serves as a good benchmark to see if the model adequately fits the training data; if the scores are subpar the model shouldn't even be tested on unseen data — because you are certain that the model is not trained well enough.

Evaluation Metrics:

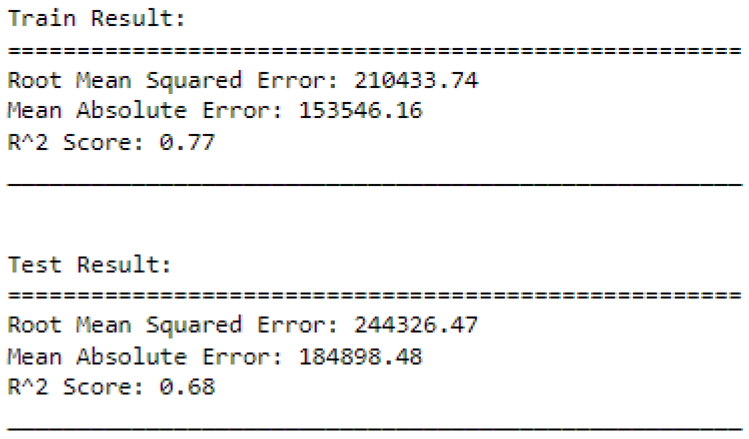
- **Root Mean Squared Error (RMSE)**: RMSE is used to benchmark regression models, it represents the average weight of errors between predicted values and true values of the target feature (Price). It is computed by the square root of the average squared differences between predicted and true values. RMSE also penalizes large errors compared to other regression metrics. Basically, RMSE is the measure of how accurate the model's average target feature predictions are compared to the actual values. This process calculates the average errors in the same units as the target feature. The lower the RMSE, the better – 0 indicates perfect predictions from the model. In our case, it is evaluating the model's average weight of errors between predicted property prices and true property prices. Basically, RMSE represents the average difference between the actual prices of properties and the prices predicted by our models.

- **Mean Absolute Error (MAE)**: MAE is a metric used to evaluate a regression model's predictive capabilities; like RMSE it measures the average magnitude of errors between predicted values and true values of the target feature (Price), however, unlike RMSE it does not square the errors. It is computed by obtaining the absolute average of the differences between predicted and true values. Also, unlike RMSE, MAE does not penalize large errors, instead it treats all errors equally no matter the magnitude. Again, like RMSE the lower the MAE values the better, with zero indicating perfect predictions by the model. In this context, MAE is measuring the model's absolute average weight of errors between predicted property prices and true property prices. This procedure calculates the absolute average errors in the same units as the target feature or Price.

- **R² Score**: the R² score measures the proportion of variance in the target feature (Price) that is predictable from all the other features in the regression model. The scores range from 0 to 1, where 1 indicates a perfect fit, meaning that the model explains all the variability in the target feature. A score of 0 indicates that the model does not explain any variability beyond the mean of the target feature. Scores between 0 and 1 represent the proportion of variability explained by the model relative to the total variability in the target feature. Therefore, a score less than 0.5 suggests that the model explains less variability than simply predicting the mean (average) of the target feature. In summary, the R² score provides insight into how well the regression model fits the training data, with higher scores indicating a better fit.

In summary, RMSE and MAE measure the accuracy of predictions, while R² score measures the regression models quality of fit to the data. These metrics provide valuable insights into different aspects of the model's performance.

Now that I have emphasized the differences between the scoring of training and testing sets, as well as covering the evaluation metrics of our models and what they mean, we can go ahead and start with the modelling phase.



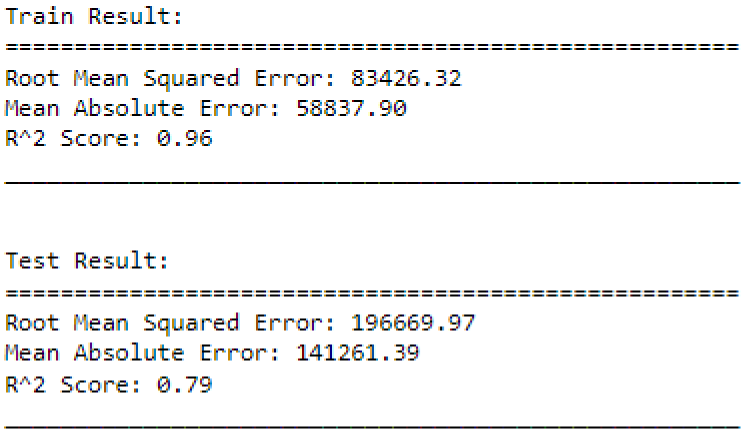
```
Train Result:
=====================================================
Root Mean Squared Error: 210433.74
Mean Absolute Error: 153546.16
R^2 Score: 0.77
_____________________________________________________


Test Result:
=====================================================
Root Mean Squared Error: 244326.47
Mean Absolute Error: 184898.48
R^2 Score: 0.68
_____________________________________________________
```

Linear Regression Model:

- Train Results: The root mean squared error (RMSE) is $210,433.74. This indicates that, on average, the difference between the predicted prices and the actual prices is approximately $210,433.74. Meanwhile, MAE is better at an absolute average of $153,546.16, which means that our model's predictions deviate from the actual prices by an average of $153,546.16. R^2 score is at 0.77 which indicates a good model fit, capturing 77% of the variance in property prices.

- Test Results: The RMSE, standing at $244,326.47, signifies the average disparity between predicted and actual prices. Conversely, the MAE, at an average of $184,898.48, offers a better representation of the model's prediction deviations from actual prices. With an R^2 score of 0.68, our model demonstrates a moderate fit, capturing 68% of the variance in property prices.



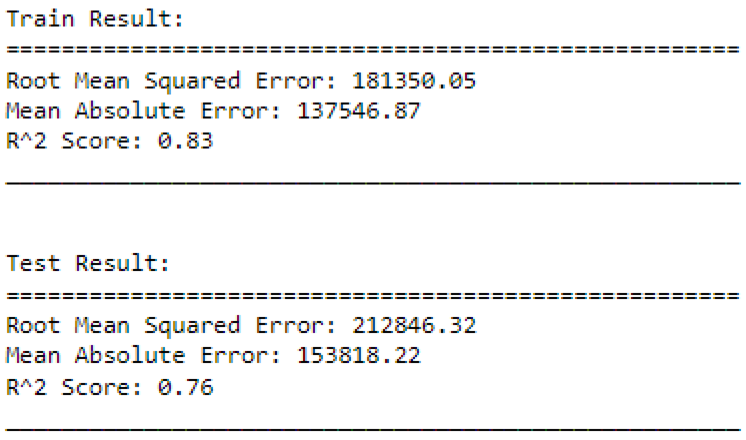
```
Train Result:
=====================================================
Root Mean Squared Error: 83426.32
Mean Absolute Error: 58837.90
R^2 Score: 0.96
_____________________________________________________


Test Result:
=====================================================
Root Mean Squared Error: 196669.97
Mean Absolute Error: 141261.39
R^2 Score: 0.79
_____________________________________________________
```

Random Forest Regression Model:

- Train Results: The RMSE of $83,426.32 represents the average discrepancy between predicted and actual prices. Conversely, the MAE, averaging $58,837.90, offers a more accurate gauge of prediction deviations from actual prices. With an impressive R^2 score of 0.96, our model demonstrates an excellent fit, possessing 96% of the variability in property prices.

- Test Results: The RMSE, amounting to $196,669.97, indicates the average gap between predicted and actual prices. On the other hand, the MAE, averaging $141,261.39, provides a more accurate measure of prediction deviations from actual prices. With an R^2 score of 0.79, our model shows a good fit, representing 79% of the variability in property prices.



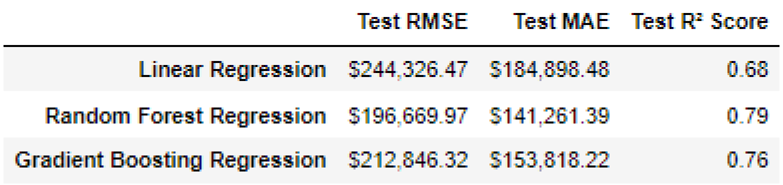
```
Train Result:
=====================================================
Root Mean Squared Error: 181350.05
Mean Absolute Error: 137546.87
R^2 Score: 0.83
_____________________________________________________


Test Result:
=====================================================
Root Mean Squared Error: 212846.32
Mean Absolute Error: 153818.22
R^2 Score: 0.76
_____________________________________________________
```

Gradient Boosting Regression (GBR) Model:

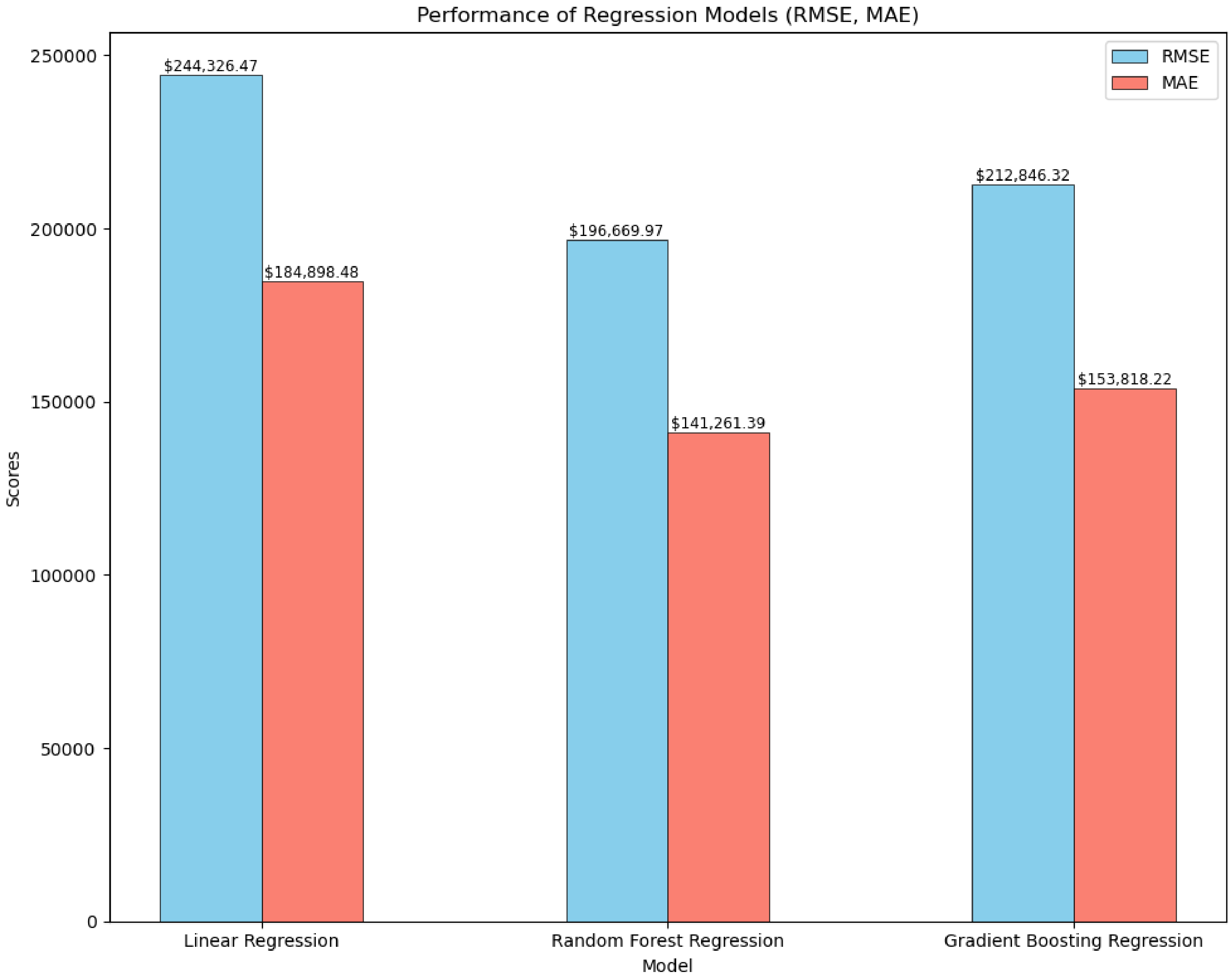
- Train Results: The RMSE, at $181,350.05, indicates the average difference between the predicted and actual prices. Meanwhile, the MAE, with an average of $137,546.87, provides a more precise measure of prediction errors from the actual prices. With an impressive R² score of 0.83, our model exhibits a high level of fit, highlighting 83% of the variability in property prices.

- Test Results: The RMSE, totaling $212,846.32, reflects the average difference between predicted and actual prices. In contrast, the MAE, with an average of $153,818.22, gives a more precise indication of prediction errors from actual prices. An R² score of 0.76 demonstrates that our model has a good fit, accounting for 76% of the variability in property prices.

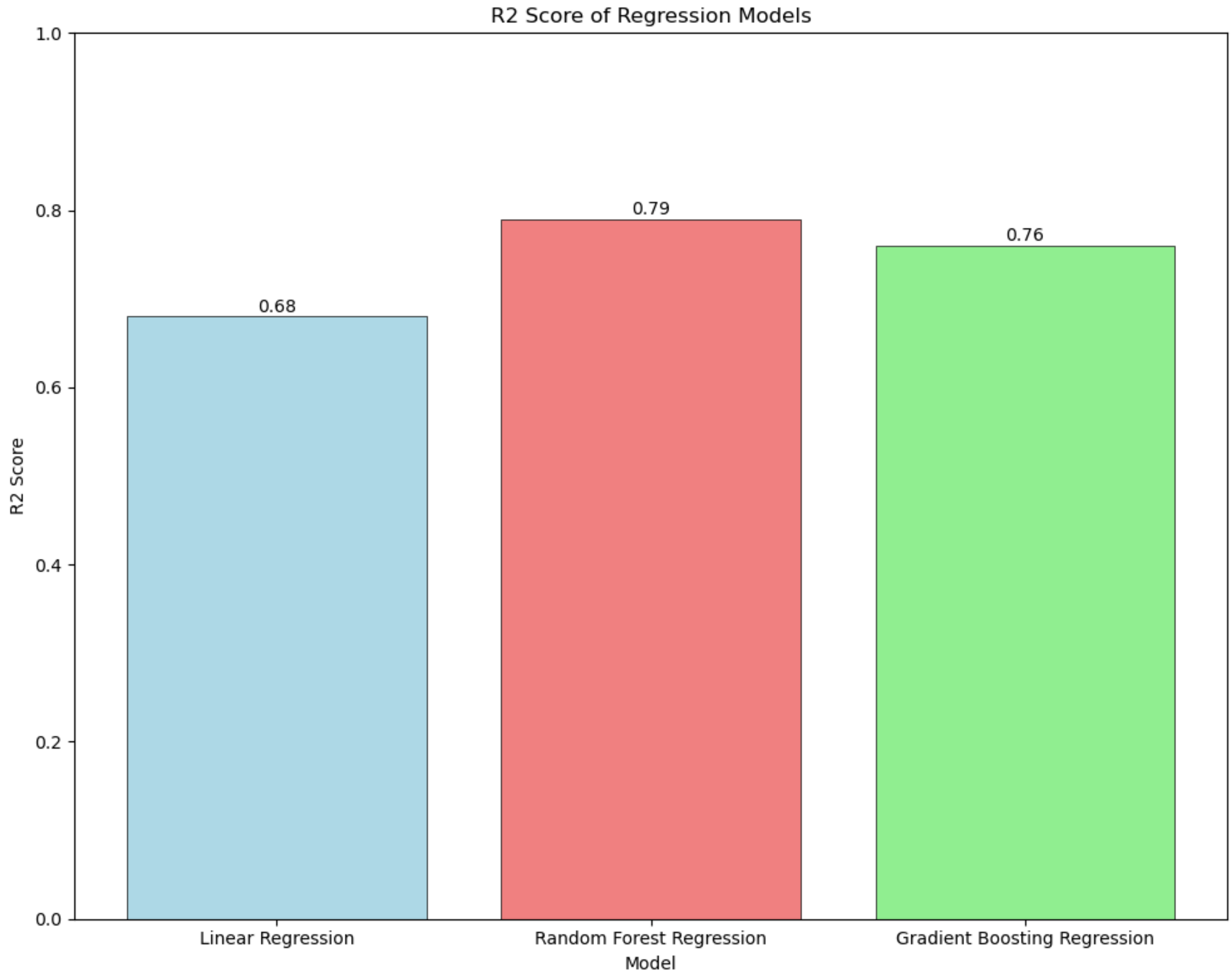


	Test RMSE	Test MAE	Test R² Score
Linear Regression	$244,326.47	$184,898.48	0.68
Random Forest Regression	$196,669.97	$141,261.39	0.79
Gradient Boosting Regression	$212,846.32	$153,818.22	0.76

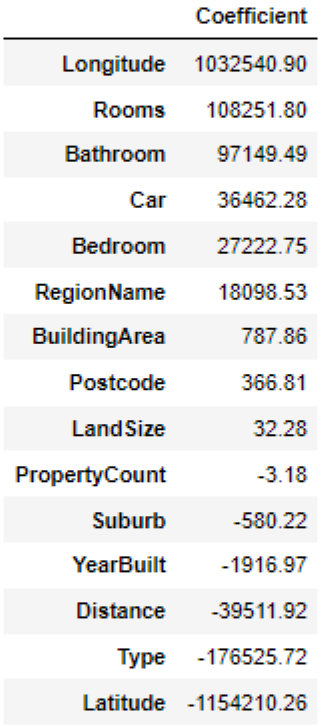
The above table is a summarization of the results of the modelling phase. However, we are only going to be focusing on test results, as emphasized before, the test results are far more important as they represent real-world scenarios were we are conducting our models to predict property prices in unseen data. By observing the table above, we can ascertain that random forest regression is currently the best performing model, containing the best scores in every metric. Let’s visualize this in another appealing way and discuss the results further.



This grouped bar plot displays the test metrics of RMSE and MAE. recall of every model. Visualizing a clearer picture of the best performing models in this regard. We can observe that overall, Random Forest Regression is the best performing model: $196,669.97 RMSE and $141.261.39 MAE. In contrast, Linear Regression is the worst performing model at $244,326.47 RMSE and $184,898.48 MAE. Obviously, there is more left to be desired with these results. Despite following the best protocols and handling the null/0-values/outliers/data entry errors and conducting feature engineering/selection the scores are good but there is room for improvement. For the scores to improve these issues would need to be resolved: potential overfitting: where the models are fitting the noise in the training data rather than capturing the underlying patterns due to the vast number of outliers, even if they are valid values. Or, because our target feature Price is skewed somewhat, this can be observed in this chart as MAE's scores are significantly better than RMSE; this is because MAE is robust to outliers where RMSE is not. Another factor could be the general quality of this data set, it needs more samples which are less skewed, and possibly new features such as distance to a shopping mall or beach, which has a high correlation with our target feature Price. Anyway, for obvious reasons I would recommend the Random Forest Regression model for property price prediction with a prefix on MAE to conduct the predictions. Evidently, it's the most ideal as its absolute average is $141.261.39, which subsequently means that our Random Forest Regression model's predictions deviate from the actual prices of property by an average of $$141.261.39. However, while this may seem like a large absolute error, it is relatively small compared to the range of prices within the dataset. Considering the mode for Price is around $1,000,000, the MAE represents almost one-tenth of a million dollars, indicating that on average, the model's predictions deviate from the actual prices by this fraction of the price range.

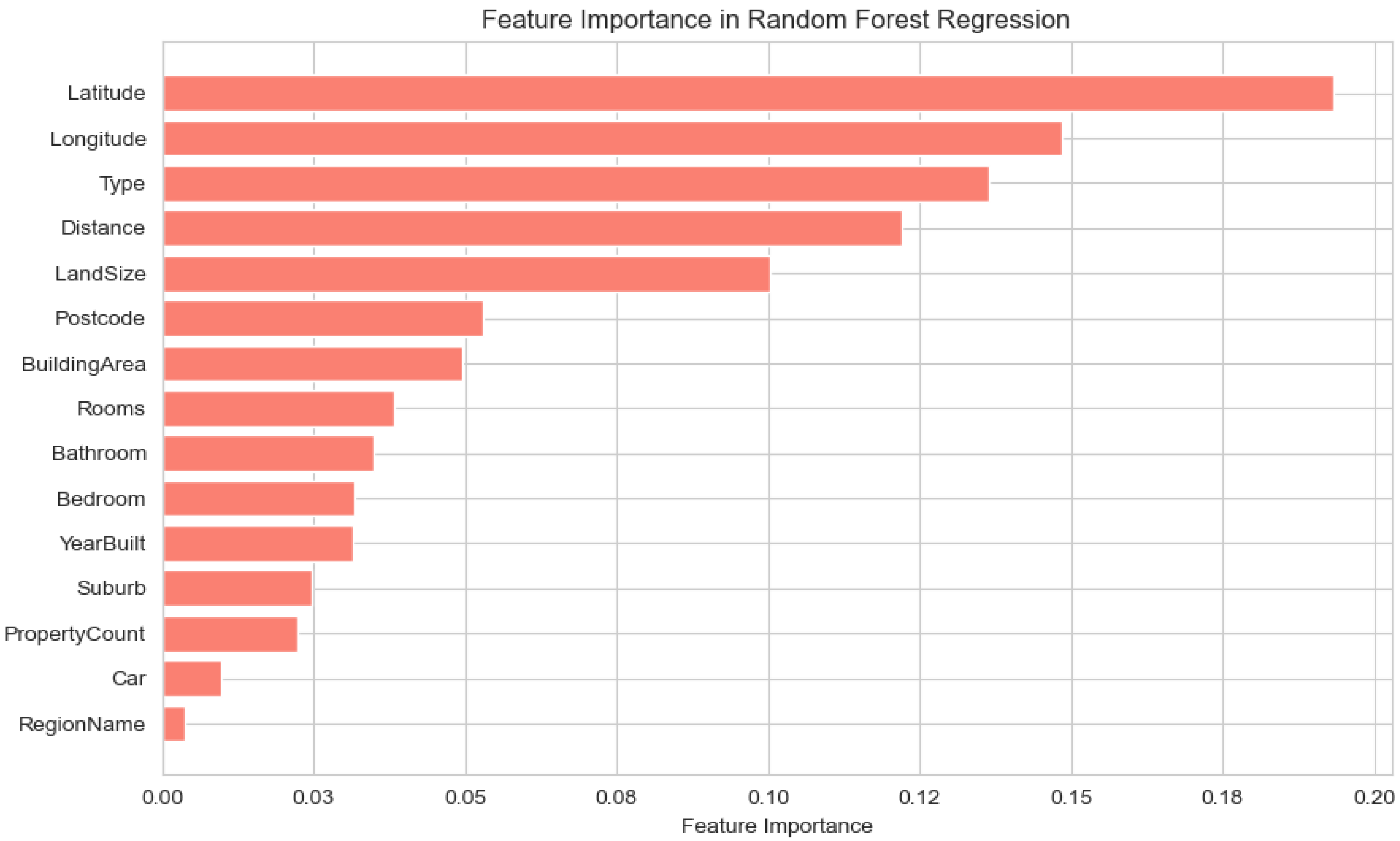


The above bar plot displays the R_2 scores of every model. Depicting a more visually appealing graph of the best performing models in this regard. We can observe Random Forest Regression is the best performing model: boasting 0.79 R_2 score. In contrast, Linear Regression is the worst performing model at 0.68 R_2 score. Overall, these scores are quite good with the lowest score being near 0.70 which indicates a good fit, and the other scores being closer to 0.80 representing a high fit, with our best model capturing exactly 79% of variance in the target feature (Price). Essentially, this means that the model's predictions of property prices are quite close to the actual prices, with only 21% of the variance remaining unexplained. This indicates a relatively high level of accuracy and predictive power in the model. It also shows that our Random Forest Regression model is very good at capturing underlying patterns and trends in the data.



	Coefficient
Longitude	1032540.90
Rooms	108251.80
Bathroom	97149.49
Car	36462.28
Bedroom	27222.75
RegionName	18098.53
BuildingArea	787.86
Postcode	366.81
LandSize	32.28
PropertyCount	-3.18
Suburb	-580.22
YearBuilt	-1916.97
Distance	-39511.92
Type	-176525.72
Latitude	-1154210.26

This table contains the coefficients of the features (independent variables) obtained from the Linear Regression model. The coefficients will tell you about the average increase in the target variable (Price) for a one-unit increase in the independent variable, holding all other independent variables unchanged (all other features besides Price). It is displayed in descending order. Surprisingly, Longitude has the highest positive coefficient, indicating it has the highest impact on price prediction. A one unit increase to longitude boosts the average predicted price of a property by $1,032,540.90 dollars. Second up is Rooms, whereby a one unit increase to rooms increases the average predicted price of a property by $108,251.80. Basically, a one unit increase in the context of Rooms means adding another room to a property. Features with higher positive coefficients contribute more to increasing the predicted price, while features with higher negative coefficients contribute more to decreasing the predicted price. Features with coefficients close to zero have minimal impact on the predicted price. Conversely, Latitude has the most negative impact on price, with an increment in latitude implicating the average property price decreasing by $1,154,210.26. Basically, this table represents feature importance towards price for the linear regression model.



The above bar plot represents feature importance towards predicting property prices in our best model: Random Forest Regression. The features are structured in descending order of importance. The scores range between 0 and 1, where a score closer to 1 indicates the highest importance and a score closer to 0 indicates the lowest importance. Higher scores show that a feature has a greater influence on predicting the price of a property. Conversely, lower scores suggest that the feature has less influence. The most important feature for our best model when it comes to predicting Price is Latitude, with a feature importance score of 0.19. This means that 19% of the model's predictive power for property prices is derived from the Latitude feature. In contrast, RegionName has the lowest importance score of 0, indicating that it does not contribute to the model's predictive power for our target feature (Price). In summary, this plot shows how much each feature contributes to the model's ability to accurately predict property prices, with higher scores indicating more significant contributions.

Business Management Suggestions:

- **Informed Investment Decisions**: Utilize the Random Forest Regression model with the prefix of mean absolute error (MAE) for property price prediction to provide key insights for clients, facilitating informed investment decisions with accurate predictions of property values.

- **Key Features Influencing Price**: Observing feature importance in Random Forest Regression model, we now know that Latitude and Longitude influence property prices the most, order our real estate consulting firm to focus solely on properties with favourable locations. This approach is guaranteed to maximize returns and profitability by far.

- **Market Trend Analysis and Forecasting**: Understand market trends and predict market shifts by analysing the Linear Regression model's positive coefficients over time, we understand that as Longitude, Rooms, and Bathrooms tend to increase, property prices also tend to increase simultaneously. Also, the magnitude of the positive coefficient strength is significant towards property prices, indicating a very strong relationship.

- **Maximize Resource Allocation**: Predictive insights found in our models sufficiently help optimizing resource allocation by knowing the most valuable property features such as Latitude, Longitude (location of property), or ample rooms and bathrooms. This will drastically improve the efficiency of marketing efforts, ensuring that all available resources are targeted towards properties in good locations that have many rooms and bathrooms.

- **Improve Operational Efficiency**: Use Random Forest Regression model to automate property price predictions; streamlining the real estate consulting firm operations as manual analysis is not required - drastically reducing the time and effort wasted on manual analysis. This improves overall operational efficiency, allowing the firm to focus more on strategic initiatives.

- **Risk Management**: Linear Regression model has pinpointed risks in properties by analysing the negative coefficients of property prices. Avoid properties that are small in unfavourable locations. This facilitates the development of risk mitigation strategies towards property investing, enabling the firm to safeguard client investments.

CONCLUSION

In this report, we addressed the business analysis task of predicting property prices using regression to support informed investment decisions for a real estate consulting firm. I employed various supervised machine learning regression models, including Linear Regression, Random Forest Regression, and Gradient Boosting Regression to analyse key factors influencing property prices and forecast market trends. The methodology involved extensive data pre-processing, including handling nulls, zero-values, outliers, and feature selection, to ensure the robustness and optimization of our models. Random Forest Regression model was by far the best model in terms of all evaluation metrics (RMSE, MAE, R^2) and predictive power. I utilized advanced regression techniques and feature importance analysis to identify the most significant predictors of property prices, such as the features Longitude, Latitude, Rooms, and Bathrooms. Through vigorous regression analysis, we uncovered valuable insights into market trends and identified optimal resource allocation to maximize return on investments. The findings emphasized the importance of focusing on properties with favourable locations with an ample number of rooms and bathrooms. Overall, our approach provides a comprehensive framework for real estate consulting firms to leverage business analytics and regression modelling techniques to make informed decisions and capitalize on market opportunities in the ever-dynamic real estate market.