ISYE 6740 - Computational Data Analysis /
Machine Learning I, Spring 2023
Project Report

Comparison of single, ensemble and neural network machine learning algorithms for residential property prices in Melbourne

Bradley Wallace, GTID# 903564466
bwallace35@gatech.edu

1 INTRODUCTION AND PROBLEM STATEMENT

The affordability of residential housing has been a growing concern in Australia's major cities, such as Sydney, Melbourne and Brisbane, for the past half-a-decade. Chowdhury, I. (2022), Professor at Australia's #1 School of Politics and International Relations at ANU, states that affordable housing being available to citizens is a hallmark of developed society; yet, Australia's housing system has been failing new generations of Australians. Furthermore, this concern has deepened into a crisis as inflation and cost of living outpace wage growth in Australia in a period of economic uncertainty post the COVID crisis of 2020-2021. To further demonstrate the pressures placed on Australian cities to provide and develop housing for the population, as of 2021, 86.36% of Australia's 25.69 million population reside in its cities, with 5.312 million and 5.078 million residing in greater metropolitan area of Sydney and Melbourne, respectively, as Australia's two largest cities (Australian Bureau of Statistics, et. al., 2023). Furthermore, in the past 10 years, seven local government areas (councils) in Melbourne's greenfield suburbs accounted for 50% of Victoria's (one of the Australia's six states) population growth (Visontay, E., 2023). In addition, Australia is set to receive 650,000 migrants across this financial year and the next, further adding to pressures on the housing market (Visontay, E., 2023).

To enhance the decision-making of these new renters and home buyers, it is necessary to provide them with easy-to-access tools that can give them an understanding of the housing market landscape. Many existing online housing price or housing value estimate tools typically require users to enter a valid property address along with housing characteristic information in order to receive a valuation. These services are typically offered from banks, other mortgage (home loan) lenders, large realtor businesses or large online listing platforms. Typically, such services are offered with further business in-mind, such as the financing of new home loans, refinancing of an existing house for free capital or selling an existing house. In addition, online listing platforms will usually return recent listings or recently-closed house sales based on search options. It is also noted that online housing databases such as onthehouse.com allow users to look up the historical values of real properties with ease. What these existing platforms lack is the ability to infer a property price based on

housing characteristics, this is usually left to professional (or amateur) judgement based on historical house and listings. Therefore, this study proposes to develop efficient and accurate prediction models to infer expected house prices based on housing characteristics and location. These model would be able to consider longer time periods of historical house sales to infer house prices, when compared with human judgement and inference which would typically only look at more recent sale history.

To better understand any key drivers of residential housing prices and develop machine learning-backed tools to assist potentially-disadvantaged homebuyers, the key objectives of this study is to undertake statistical analysis of housing sale data in Melbourne and surrounds, Australia, and develop machine learning models to predict housing prices based on housing characteristics such as age, size, number of rooms, distance from CBD, region (spatial) and temporal information that may follow market trends.

Statistical analysis of housing sale data will be undertaken using a range of data exploration techniques and application of domain expertise and knowledge. Then, a range of single, ensemble and neural network machine learning regression model architectures will be developed to predict Melbourne house prices from the Melbourne housing sale data. K-fold Cross Validation and multi-processing will be used to tune model hyperparameters to achieve robustness and high performance in the various regression models. The best-performing regression models will then be further assessed using Monte Carlo Cross Validation to analyze their sensitivity to data variation (i.e. another measure of model robustness) and statistical tests will be performed to determine if there is definitively-best regression model for housing price prediction in Melbourne, Australia. Lastly, there will be a high-level comparison of the behavior (regression solution space) of the different types of machine learning regression models developed to better understand how the models make predictions and identify any relationships with the characteristics describing houses and housing sales.

2 DATA SOURCE

Housing sale data for Melbourne, Australia, was scraped from publicly available results posted each week from *Domain.com.au*, the second largest online real-estate marketing platform in Australia. The dataset spans from 2016-03-09 to 2018-02-24 and contains 34,857 data points (individual house sales) with 20 descriptive characteristics (described below) and the houses' price at sale date. Further information on the dataset can be found at the following link, where some preliminary data cleaning was performed by the original data scraper (Pino, T., 2018).

The 20 descriptors and potential predictor variables for modeling are described in **Table 1** below.

Table 1. **Melbourne Property Sale dataset descriptors and sale price, variable descriptions**

Suburb:	Property Suburb	**SellerG:**	Real Estate Agent
Address	Property Address	**Date:**	Date of sale
Rooms	No. of rooms, listed	**Distance:**	Distance from Melbourne CBD in Kms
Type:	**br** - bedroom(s); **h** - house, cottage, villa, semi, terrace; **u** - unit, duplex; **t** - townhouse; **dev site** - development site; **o res** - other residential	**Postcode:**	Postal Code, this is a bounded postal service area typically composed of several suburbs
Price:	Price in Australian dollars	**Bedroom2:**	No. of Bedrooms, from different source
Method:	Method of sale **S** - property sold; **SP** - property sold prior; **PI** - property passed in; **PN** - sold prior not disclosed; **SN** - sold not disclosed; **NB** - no bid; **VB** - vendor bid; **W** - withdrawn prior to auction; **SA** - sold after auction; **SS** - sold after auction price not disclosed; **N/A** - price or highest bid not available.	**Bathroom**:	Number of Bathrooms
		CouncilArea:	Governing council (local government) for the area. These are typically composed of many (15+) suburbs.
Car:	Number of on-property car parking spaces (In Australia, this is not always a garaged or undercover parking space)	**Latitude:**	Latitude in decimal format

Table 1. **Melbourne Property Sale dataset descriptors and sale price, variable descriptions**

Landsize:	Land size in sq. meters	**Longitude:**	Longitude in decimal format
BuildingArea:	Building Size in sq. meters	**RegionName:**	General Region (West, North West, North, North East, and so on)
YearBuilt:	Year the house was built	**PropertyCount:**	Number of properties that exist in the Suburb i.e. size of Suburb

The extent of the greater metropolitan area of Melbourne in Victoria, Australia, (and thus the data collection) can be seen as the darker-shaded area in **Figure 1** below. The respective local government body can be seen as the individual names in each sub-area of the darker-shaded area.

Figure 1 — Map of greater metropolitan area of Melbourne and respective councils (Municipal Association of Victoria, 2023)

3 METHODOLOGY

The methods used in the analysis and modeling of this study include (in this general order): Distribution Analysis, Exponential Smoothing, Dummy and Temporal Variable Encoding, Outlier Removal, Correlation Analysis, ElasticNet Linear Regression, K-Nearest Neighbors (KNN), Support Vector Machine Regression (SVR), Random Forest, Adaptive Boosting (AdaBoost), Gradient Boosting, Neural Network Regression, Shapiro-Wilk normality Test, Wilcoxon signed-rank Test and Student's t-Test and Principal Component Analysis (PCA, for dimensionality reduction and visualization). Model hyperparameter tuning to develop robust and high-performance models utilizes K-Fold Cross Validation and the robustness (or sensitivity to variation of the data) of the best-performing supervised learning architectures is further explored using Monte Carlo CV.

The methodology was implemented as per Python code in the *Attachments*. In particular, the various supervised learning regression models are all implemented using the respected machine learning library in Python, *scikit-learn*. Statistical tests are implemented using *statsmodels.api*. Furthermore, other Python libraries are utilized: *pandas* and *numpy* are used for matrix, array and table operations, and *matplotlib.pyplot* and *seaborn* are used for plots and visualizations. Finally, the *multiprocessing* package is used to optimize model training times through utilization of additional processing cores.

Steps in the methodology are summarized as follows:

3.1 Data Preparation and Exploratory Data Analysis

1. Preliminary variable selection, removal of redundant predictor variables using preliminary summary tables of the dataset and application of real estate, engineering and data science domain knowledge.
   Five predictor variables 'Address', 'Latitude', 'Longitude' and 'Postcode' were removed because there are several other spatial predictor variables such as 'Suburb', 'Distance', 'CouncilArea' and 'RegionName'. 'Postcode' is of particular note because it has 209 unique categorical levels and therefore would create an additional 209 dummy variables. A decision was made to keep 'Suburb' rather than 'Postcode' as a predictor variable to describe the local area of a datapoint. Similarly, 'SellerG' (realtor who conducted the sale) is removed because the study objective is to predict sale price using housing characteristics and spatial-temporal information.
2. Addressing missing values (NaNs) in 'Price' and categorical data columns through deletion
   This initial removal of missing values reduced the dataset from 34857 datapoints to 27244.
3. Preliminary outlier removal through reducing the dataset to reasonably sized property and houses (less than 10,000 square meters) as the study is interested in residential houses.

This preliminary outlier removal (or narrowing of the dataset) reduced the dataset from 27244 datapoints to 9366.

4. Analysis of average house sale price over time in the data set to assess any upward or downward trends, also using Exponential Smoothing and forecasting.
5. Encoding date of sale as a continuous variable such that there is a predictor variable that can capture decreasing, stable or increasing housing sale prices over time.
   A continuous integer variable, 'SaleDate_MonthsFrom2015EOY', was encoded in the dataset to replace 'Date' (date of sale) as datetime cannot be interpreted by the machine learning models. This measures the time in months from End of Year 2015, where the modeling period will begin.
6. Distribution analysis of predictor variables using frequency and versus house sale price, to detect outliers and redundant predictors that offer no variability and explanatory power.
7. Post-distribution analysis removal of outliers.
   Initial distribution analysis revealed large voids in the x-axis in histogram plots of 'YearBuilt', 'BuildingArea', 'Landsize' and 'Price'. Therefore, it was necessary to prune the dataset to remove the outliers that fell far outside the observable normal and log-normal distributions ('YearBuilt' appears to be the multi-modal). An outlier at 'YearBuilt' = 1196 was removed, and 'BuildingArea' (and by proxy 'Landsize') was further restricted to less than 2000 square meters. Furthermore, maximum 'Price' in the dataset was restricted from $9million to $7million. This process reduced the extent of the dataset from 9366 datapoints to 9358.
8. Encoding of dummy variables for continuous predictor variables with a high occurrence of missing values.
9. Encoding of dummy variables for multi-level categorical variables.
   Encoding of dummy variables for high amounts of missing values in continuous variables, as well as dummy variables for each level in multi-level categorical variables resulted in increasing the number of predictor variables in the dataset from 15 to 381.
10. Correlation analysis, noting it only captures linear relationships.
    Post-correlation analysis removal of redundant predictor variables.
    'Method' of sale was poorly-correlated with other variables and deemed unnecessary to include in final model training.
11. Noting that there may be time-scale effects (such as trends) and trade-offs between with computational effort and model performance (longer time periods mean more data to train on), split the data into previous 12-month, 24-month and 36-month data (i.e. all the available data).

3.2 Predictive Machine Learning Modeling Development

For 12-month, 24-month and 36-month data:

1. Normalize float and int values in the dataset on a 0-1 scale.
2. Drop 'Date' data from the dataset as it is no longer required

3. Split the dataset into an 80% training set and a 20% testing set for model training using 5-Fold CV. Justification for this split are as follows: most of the data (80%) is used for training the models and a reasonable remaining portion (20%) is reserved for evaluating the models' performances; random split reduces potential order bias that may be present in the data set; and the 80/20 split is pedagogical standard that is repeated many times in our course work. It is also commonly seen across studies and is considered an industry standard.
4. Perform 5-Fold CV Grid Search on a range of hyperparameters for seven different regression models on the training set and evaluate their performance on the testing set. The metrics for evaluation of model performance is $\mathbf{r^2}$ (the coefficient of determination) and **mean squared error (MSE)**. Ideally, the best $\mathbf{r^2}$ and **MSE** for each supervised learning regression model architecture will have the same hyperparameters. The seven regression model architectures, and their associated tuning hyperparameters are as follows:
    1. **ElasticNet Linear Regression**, **l-1** ratio (**l-1**: LASSO, **l-2**: Ridge);
    2. **K-Nearest Neighbors (KNN)**, k number of neighbors;
    3. **Support Vector Machine Regression (SVR)**, C regularization parameter, ε margin of tolerance;
    4. **Random Forest Regressor**, number of trees and minimum samples per leaf (i.e. model complexity);
    5. **AdaBoost Regressor**, number of estimators and learning rate;
    6. **Gradient Boosting Regressor**, number of estimators and learning rate; and
    7. **Neural Network**, α penalty term and hidden layer architectures (single layer or dense multi-layer)
5. Based on the 5-CV Grid Search results for the models trained on 12, 24 and 36-month datasets, find the most stable and 'optimal' hyperparameters for each of the seven regression models, tabulate the results and retrain the final models using those 'optimal' hyperparameters values which yield the best performance.
6. For completeness, retrain the best performing models of all model architectures and summarize their performance using metrics: $\mathbf{r^2}$, **MSE** and **MAE** (Mean Absolute Error). Compare their performance to a null hypothesis predictive model that assumes all predictions are simply the average sale price for the entire dataset.
7. If the two or more models have comparable performance it is necessary to undertake Monte Carlo CV to further test model robustness, comparing their performance using $\mathbf{r^2}$. Using these results, it is necessary to perform further statistical tests to determine which of the remaining regression models is definitively the best. Wilcoxon signed-rank Test and Student's t-Test will be used to determine if the regression models both model the same underlying distribution or if they are different. If statistical tests reject the null hypothesis and determine that any comparable models

are truly different, it is then prudent to accept the model with the highest $\mathbf{r^2}$ and (hopefully) lowest **MSE** and **MAE** as the best model.

3.3 Solution Space Visualization for Regression Models - PCA 2-Dimension Approximations

To facilitate understanding of the seven regression models' performance, the solution spaces of each model architectures will be visualized over a grid in PCA 2-dimension space using a simplified dataset. To achieve this, the following steps will be taken:

1. The dataset will be simplified from 381 predictor variables to 19 predictor variables by removing the very large amounts of 'Suburb', 'RegionName' and 'CouncilArea' categorical dummy variables.
2. The predictor portion of the dataset will be projected into the first two principal directions of a Principal Component Analysis (PCA) i.e. the predictor variables will be combined through PCA dimensionality reduction into 2 dimensions. This will then be merged back with original dataset and true 'Price' values.
3. Scatterplots of the PCA 2-dimensional data will be explored to understand if the 2 principal directions make logical sense in terms of the original variables.
4. The seven regression models: Elastic Net, KNN, SVR, Random Forest, AdaBoost, Gradient Boosting and Neural Net, will be re-trained on the projected predictor training data and a single response variable ('Price') using hyperparameters obtained from previous tuning.
5. The models trained on the 2-dimensional predictor data will then predict the response variable 'price' over a grid at equal intervals e.g. from –6 to 16 in the X-direction and –6 to 6 in the Y-direction, at 100 intervals each, which results in 10,000 equally-spaced data points.
6. Each model's solution space (grid predictions in 2-D) will then be compared to the true values in order to intuitively appreciate and potentially explain the observed performance of each model architecture.

3.4 Particular methodology comments

The 2-dimensional projections of each classification model's decision boundary is not strictly the same as the multi-dimensional models for which testing error (misclassification rate) is calculated. They are as close-as-possible approximations for the purpose of visualizations and facilitating understanding.

4 EVALUATION AND FINAL RESULTS

4.1 Time-series analysis

In the data exploration phase of the study, monthly-moving average of 'Sale' price and Exponential Smoothing was used to determine if there were any trends in the response variable 'Price'. This was of particular importance because it helps to guide the feasibility of developing machine learning models using historical data to predict a variable ('Price') that has over the long-term typically trended upwards in entire history of modern society. This was achieved by plotting monthly-averages of house sale prices and applying the Exponential Smoothing method to the time series. In addition, a 6-month forecast after the dataset period (i.e. until May 2019) was modeled by the Exponential Smoothing method. This plot can be viewed in **Figure** 2 below. Exponential Smoothing revealed no clear upward or downward trend in the movement of average house sale price. The smoothing process shows that the dataset is 'noisy' and oscillating around an average sale price of $1,080,000. However, there is a steep decline in average house sale price at the end of the dataset period which causes the Exponential Smoothing method to forecast the average house sale price as $1,000,000 in the next 6 months after the dataset period (i.e. until May 2019), being lower than the previously-observed 'noisy' average sale price of $1,080,000.

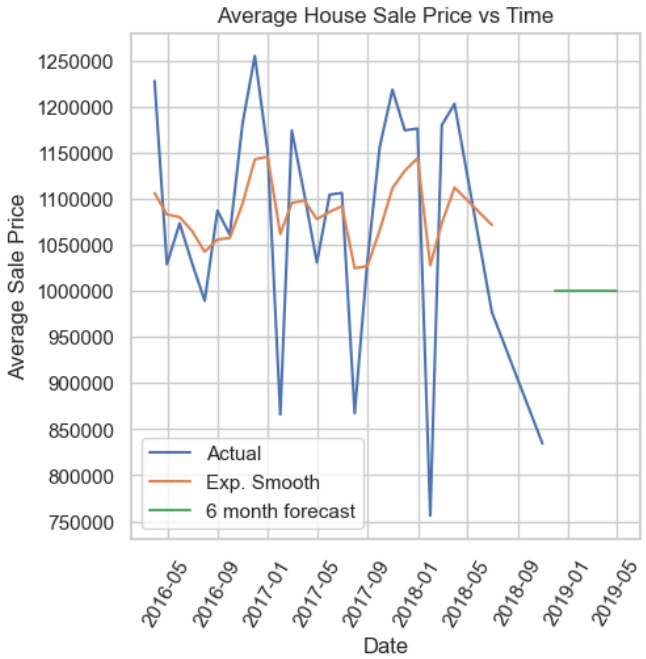
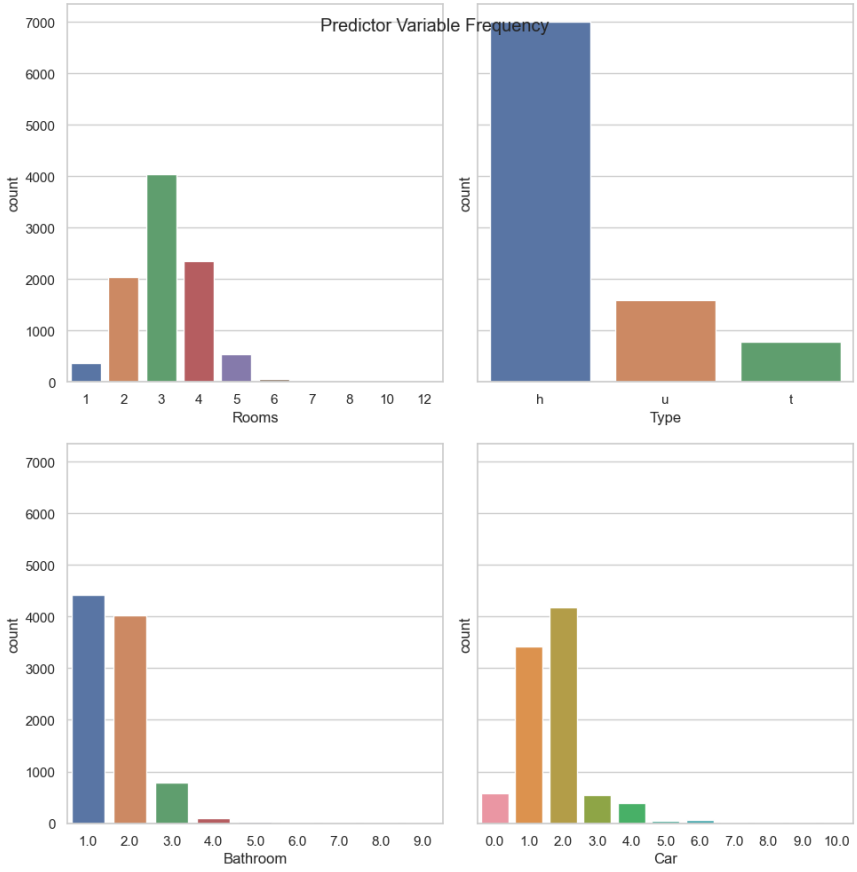


Figure 2 — Monthly-average house sale price and Exponential Smoothing with 6-month forward forecast

Although housing prices appear stable over the provided dataset period, it was determined as necessary to encoded date of sale as continuous integer variables (in months) as a new predictor variable. In that way, the modeling architecture and methodology can be preserved for any updates to the data and capture any house price trends in the data.

4.2 Distribution analyses

Distribution analyses of all predictor variables was undertaken. First, with histogram plots noting predictor value frequency, and then in comparison to sale price using scatterplots for



continuous variables and boxplots for categorical variables. The full extent of distribution analyses can be viewed in *Appendix*, however some sample plots are viewable below (refer **Figure 3** and **4**).

Figure 3 — Sample of Distribution Analyses for Predictor Variables #1

For the most part, these plots facilitated outlier removal and guided restriction of the dataset to reasonable maximums. For example, the dataset originally included multi-million-dollar penthouses, mansions and rural estates, but distribution analyses facilitated the decision to exclude such properties from further cleaned version of the dataset.

In saying that, it is important to note that many predictor variables are normal or log-normal in appearance, including house price, number of rooms, number of bathrooms, car parking spaces, land size, building size, distance from the CBD.

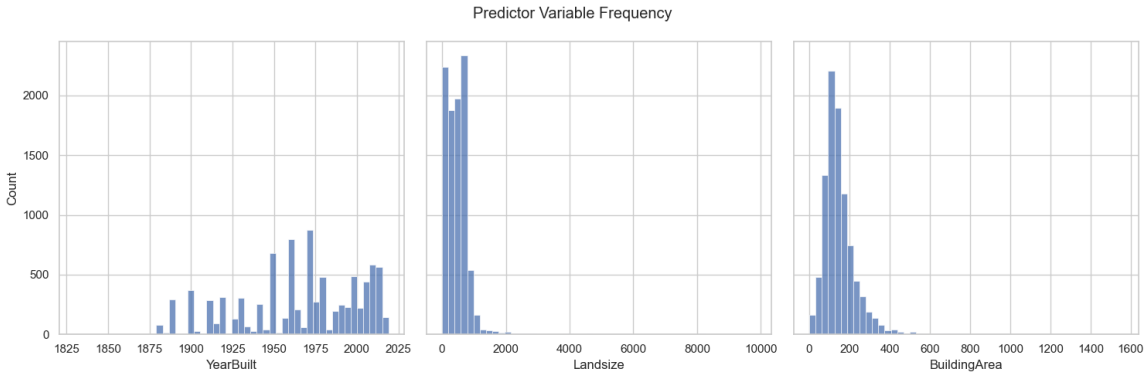


Figure 4 — Sample of Distribution Analyses for Predictor Variables #1

Stevens, C. F. (1974) states that the demand for domestic goods of varying amounts or features are typically log-normally distributed. Coupled with the fact that many human behaviors tend to exhibit a normally-distributed phenomenon, it makes logical sense that demand for housing (and therefore price), housing sizes, the number of bedrooms and carparking spaces, and distance from the CBD might exhibit normal or log-normal distributions in a sufficiently large dataset. Log-normal distributions are left-skewed normal distributions with long gradual right-tails that tend to arise because there is a minimum threshold required. For example, encapsulated in this dataset, there are minimum house prices, legislated minimum land sizes, legislated minimum building size and legislative requirements to have at least one toilet in a house, which likely causes all related predictor variables to exhibit log-normal behavior. Furthermore, house sale price appears to be normally-distributed around a mean centered on fairly large family houses of 5-6 bedrooms and 4-6 bathrooms.

In terms of house types sold, it was noted that approximately 75% of sales in the dataset were houses, 17% of sales were units or apartments and 8% of sales were townhouses. Further, house prices are higher in the southern metropolitan region of Melbourne. In addition, house prices spike upwards for houses with 7 car parking spaces (perhaps these are mansions).

It was also observed that the year a house was built ('YearBuilt') appeared multimodal (refer Figure **4**) and this makes logical sense given there are periods of population boom and housing development thoroughout the 20th century.

4.3 Correlation analysis

Correlation coefficients are indicators of the strength of the linear relationship between variables. Therefore, a correlation analysis using a correlation matrix is a highly useful tool for undertaking preliminary data analysis prior to developing the multiple regression models. **Figure 5** depicts the strength of correlation coefficients between all the predictor variables (note that suburb and Council area are omitted) and response variable ('Price'),

where a light is a strong positive correlation and dark is a strong negative (or inverse) correlation.

The correlation analysis can be summarized as follows and generally follows logical relationships between the housing characteristics (e.g. a larger house size is more likely to have more rooms, bedrooms and car parking spaces):

- 'Price' is positively correlated with 'Rooms', 'Type_h' (house), 'Bedroom2', 'Bathroom', 'BuildingArea' and 'Southern Metro Area'. Meaning, an increasing number of rooms, whether the property for sale is a house and whether it is in the southern metro area increase the housing price.
- 'Price' is negatively-correlated with 'Type_u' (unit). Meaning if the property for sale is a unit or apartment, it will command a lower sale price.
- There are weaker correlations of 'Price' with 'Car' (parking spaces), 'Landsize' and 'Western Metro Area', highlighting that these are less of factor when purchasing prices are determined by market demand.
- 'Distance' (from CBD) is reasonably positively-correlated with the number of rooms, whether a property is a house and the number of carparking spaces. It is also inversely-correlated with whether the property is a unit. This suggests that there are more units (high density residential development) closer to the CBD and houses are located further away from the CBD in suburban areas. This also infers that units have less carparking spaces potentially due to being closer to the CBD.
- 'Rooms' and 'Bedrooms' have a very-close relationship at 0.96 correlation. Please note that bedroom counts do not typically include counts of study rooms or other smaller rooms, but a room count for a house will include. This explains this slight variation.

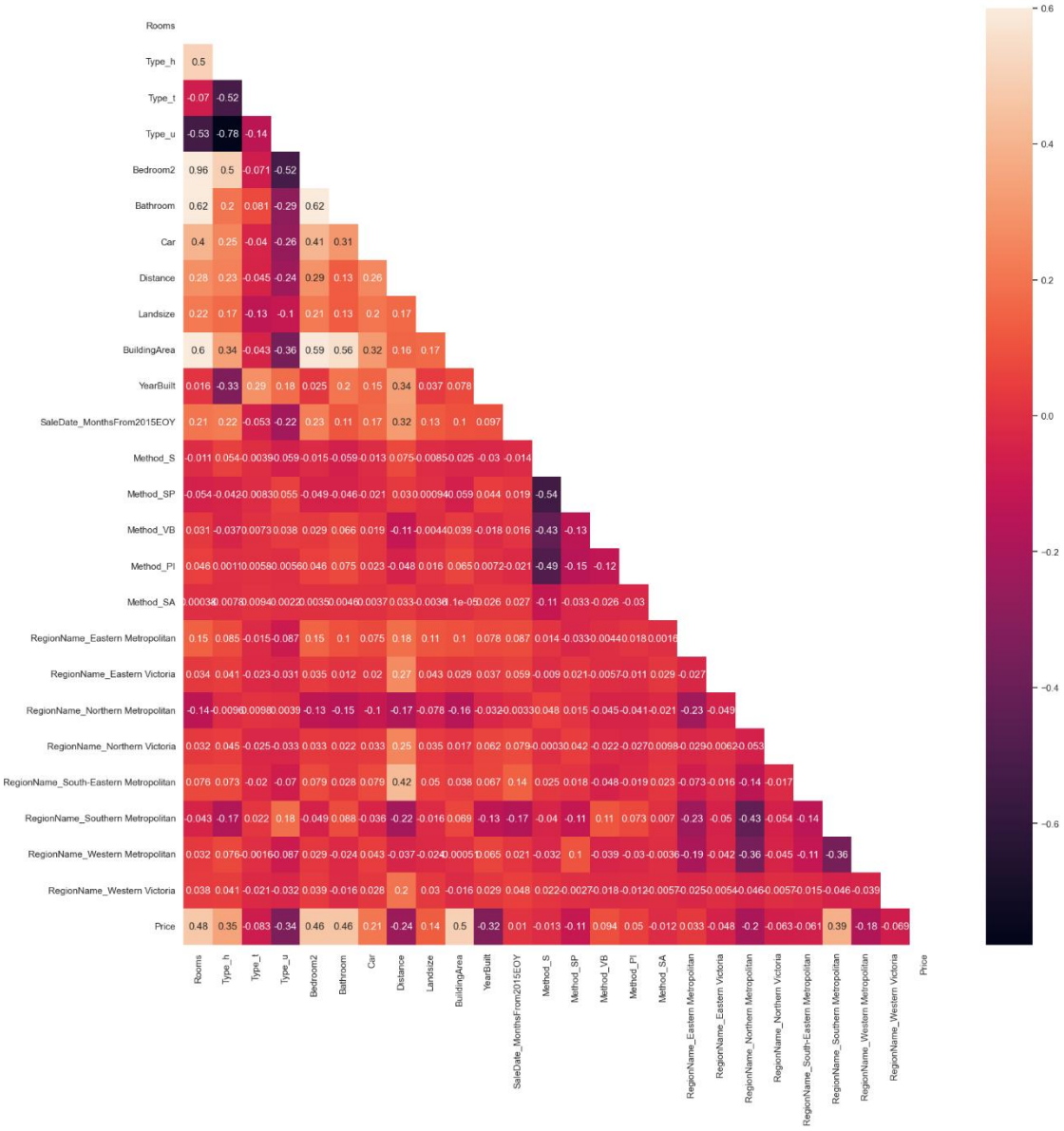


Figure 5 — Correlation matrix of the 7 predictor and 1 response ('mpg01') variables

- Many of the rooms are well-correlated with one another, and this makes logical sense as a larger home will typically have coincidingly more bedrooms, rooms and bathrooms.
- The number of rooms is positively-correlated with a property being a house and inversely-correlated with it being a unit, which makes logical sense.
- Houses and units are strongly inversely-correlated with one another, if a property is not a house, it is most likely a unit (unless it is townhouse), which makes logical sense.
- The correlation analysis informed decision-making to remove 'Method' (of sale) from the final variable selection, as 'Method' is poorly-correlated with other variables and therefore has little explanatory power.

4.4 Seven Regression Models Performance

5-Fold CV Results with 'optimal' hyperparameters for the seven regression models for the 12-month, 24-month and 36-month (entire) datasets can be viewed in **Table 2** below. **MSE** and $\mathbf{r^2}$ were used to assess their performance. The model architectures are: Elastic Net Linear Regression, K-Nearest Neighbors (KNN), Support Vector Machine Regression, Random Forest, Adaptive Boosting (AdaBoost), Gradient Boosting and Neural Network Regression

Table 2 — Optimal hyperparameter results 7 classification models, for 5-Fold CV

```
###### 12 Month Data, Machine Learning Model Performance #####
```

	Reg Model Type	MSE best params	MSE	r2 best params	r2
0	ElasticNet	{'l1_ratio': 0}	0.010390	{'l1_ratio': 0}	0.182830
1	KNN	{'n_neighbors': 7}	0.005797	{'n_neighbors': 7}	0.544111
2	SVM	{'C': 10, 'epsilon': 0.1}	0.005014	{'C': 10, 'epsilon': 0.1}	0.605701
3	RandomForest	{'max_features': 'sqrt', 'min_samples_leaf': 1...	0.002682	{'max_features': 'sqrt', 'min_samples_leaf': 1...	0.784221
4	AdaBoost	{'learning_rate': 0.1, 'n_estimators': 100}	0.004537	{'learning_rate': 0.1, 'n_estimators': 100}	0.653059
5	GradientBoost	{'learning_rate': 0.1, 'n_estimators': 400}	0.002496	{'learning_rate': 0.1, 'n_estimators': 400}	0.801822
6	NeuralNet	{'alpha': 0.35, 'hidden_layer_sizes': (377,)}	0.003803	{'alpha': 0.1, 'hidden_layer_sizes': (200,)}	0.699728

```
###### 24 Month Data, Machine Learning Model Performance #####
```

	Reg Model Type	MSE best params	MSE	r2 best params	r2
0	ElasticNet	{'l1_ratio': 0}	0.011095	{'l1_ratio': 0}	0.145940
1	KNN	{'n_neighbors': 9}	0.005557	{'n_neighbors': 9}	0.572229
2	SVM	{'C': 10, 'epsilon': 0.1}	0.004163	{'C': 10, 'epsilon': 0.1}	0.679561
3	RandomForest	{'max_features': 'sqrt', 'min_samples_leaf': 1...	0.002238	{'max_features': 'sqrt', 'min_samples_leaf': 1...	0.821590
4	AdaBoost	{'learning_rate': 0.1, 'n_estimators': 100}	0.004511	{'learning_rate': 0.1, 'n_estimators': 100}	0.656653
5	GradientBoost	{'learning_rate': 0.1, 'n_estimators': 400}	0.001975	{'learning_rate': 0.1, 'n_estimators': 400}	0.848588
6	NeuralNet	{'alpha': 0.001, 'hidden_layer_sizes': (377,)}	0.002474	{'alpha': 0.001, 'hidden_layer_sizes': (377,)}	0.789934

```
###### 36 Month Data, Machine Learning Model Performance #####
```

	Reg Model Type	MSE best params	MSE	r2 best params	r2
0	ElasticNet	{'l1_ratio': 0}	0.009258	{'l1_ratio': 0}	0.161331
1	KNN	{'n_neighbors': 7}	0.004776	{'n_neighbors': 7}	0.567334
2	SVM	{'C': 10, 'epsilon': 0.1}	0.003582	{'C': 10, 'epsilon': 0.1}	0.675544
3	RandomForest	{'max_features': 'sqrt', 'min_samples_leaf': 1...	0.001788	{'max_features': 'sqrt', 'min_samples_leaf': 1...	0.835403
4	AdaBoost	{'learning_rate': 0.1, 'n_estimators': 100}	0.003827	{'learning_rate': 0.1, 'n_estimators': 100}	0.649904
5	GradientBoost	{'learning_rate': 0.25, 'n_estimators': 400}	0.001740	{'learning_rate': 0.25, 'n_estimators': 400}	0.837314
6	NeuralNet	{'alpha': 0.001, 'hidden_layer_sizes': (377,)}	0.002148	{'alpha': 0.001, 'hidden_layer_sizes': (377,)}	0.802874

'Optimal' hyperparameters are very stable across all three variations of the dataset. The optimal number of neighbors for the KNN algorithm increases from 7 to 9 for the 24-month dataset, but returns to 7 neighbors in the 12 and 36-month datasets. Similarly, the number of trees in the Random Forest is 150 for the 24-month data, but 200 trees for the 12 and 36-month datasets. Random Forest is always 'optimal' at having leaves of 1 sample. It is noted that AdaBoost favors a lower number of estimators, whilst Gradient Boosting favors a higher number of estimators. It is also noted that the Neural Network architecture

favors a large single-layer as opposed to multiple dense layers of small number per layer, but equivalent magnitude of interconnectivity. Please note the number of Neural Network nodes in a single layer is capped at the number of predictor variables.

It should be noted that there is slight variability amongst runs for ensemble methods and the neural network model, but is overall fairly stable and will yield similar results to the above in **Table 2**. This may be due to these models' complexity, as they generally converge to similar solutions despite the split of training and test data being fixed for each run. On the other hand, Elastic Net, KNN and SVM Regression, the single non-ensemble methods, converge to the same solution each run.

4.5 Regression Model Shortlisting

It was found that the Random Forest, Gradient Boosting and Neural Network regression models yielded the best performance and using the 36-month dataset. Thus, these models selected for further analysis.

Retraining these models yielded the following $\mathbf{r^2}$, **MSE** and **MAE** in **Table 3**.

Table 3 — Final Model Performance

	Reg Model Type	r2	mean sq error	mean abs error
0	RandomForest	0.833484	0.001838	0.025556
1	GradientBoost	0.837977	0.001789	0.025234
2	NeuralNet	0.795596	0.002256	0.031272
3	Assumed Average Price	-0.000001	0.011039	0.076108

The three models were compared to a model that simply assumes that all predictions are the average house price in the dataset. This basic straight-line model has a coefficient of determination ($\mathbf{r^2}$) of zero as it does not account for the variability observed in the data. In terms of $\mathbf{r^2}$, these three models are able to explain 80% or more of the variation in the price of a house, based on the variability observed in the predictor variables. Curiously, all three models **MAE** is still within a similar order of magnitude to basic straight-line model. Converting these error terms back from a 0-1 scaling to the true values, in addition to converting **MSE** to **RMSE** (Root Mean Squared Error), reveals that these models make average predictive errors in the realm of $400,000 to $1.2million AUD. Refer **Table 4**.

Table 4 — Final Model Performance in real $AUD

Final Models - 36 Month		
	RMSE (0-1)	**RMSE ($AUD)**
RandomForest	0.001838	$ 404,051.15
GradientBoost	0.001789	$ 400,386.87
NeuralNet	0.00256	$ 453,248.74
Average , global	0.011039	$ 800,169.47
	MAE (0-1)	**MAE ($AUD)**
RandomForest	0.025556	$ 1,149,163.89
GradientBoost	0.025234	$ 1,142,729.23
NeuralNet	0.031272	$ 1,257,287.02
Average, global	0.076108	$ 1,888,059.24

As the model performance between the three final models is fairly comparable, it is necessary to perform Monte Carlo Cross Validation to assess their robustness to large multiples of resampling of the data, with respect to $\mathbf{r^2}$. Furthermore, this allows the usage of statistical tests to determine if the models are modeling different underlying distributions i.e. they are truly different. The results of Monte Carlo 50-CV can be viewed in **Table 5** below.

Table 5 — Final Model $\mathbf{r^2}$ Monte Carlo 50-CV

	RandomForest	GradientBoost	NeuralNet
mean	0.832328	0.838993	0.779316
std	0.011589	0.010047	0.032930
variance	0.000134	0.000101	0.001084

Under Monte Carlo 50-CV, it is noted that the Random Forest and Gradient Boosting regression models have $\mathbf{r^2}$ variances one order of magnitude lower than the Neural Network model, suggesting higher robustness. It is also noted that Random Forest and Gradient Boosting regression models have higher mean $\mathbf{r^2}$ values.

To test if the differences between model results are statistically significant at 5% confidence, paired tests were used, including the Wilcoxon signed-rank Test (a non-parametric test) and Student's t-Test, whose null hypotheses are that the two distributions are the same. In order to use the t-Test, which has stronger statistical inference, it is necessary to demonstrate the data being compared is normally-distributed. This can be achieved using the Shapiro-Wilk test, whose null hypothesis is that the data is normally-distributed. It was found that the Random Forest and Gradient Boosting models 50-CV results were normally-distributed and thus could be compared using Student's t-Test. A summary of the statistical tests described above can be seen below in **Table 6**.

Table 6 — Statistical Tests demonstrating Gradient Boosting is statistically-significant

Shapiro-Wilk normality Test			
Model	*p-value*	*Outcome (p-value <0.05: H,a)*	
RandomForest	0.0627	H,0: normally-distributed	
GradientBoost	0.6298	H,0: normally-distributed	
NeuralNet	5.26E-10	H,a: not normally-distributed	
Wilcoxon signed rank Test			
Model 1	*Model 2*	*p-value*	*Outcome (p-value <0.05: H,a)*
RandomForest	GradientBoost	0.000529	H,a: models are different
RandomForest	NeuralNet	1.78E-15	H,a: models are different
GradientBoost	NeuralNet	1.78E-15	H,a: models are different
Student's t-Test			
Model 1	*Model 2*	*p-value*	*Outcome (p-value <0.05: H,a)*
RandomForest	GradientBoost	0.002746999	H,a: models are different

The statistical tests demonstrate that the Gradient Boosting model on the 36-month data is definitively the best model amongst the various regression models developed.

4.6 Final Regression Model and Closing Remarks

Predictions can now be performed using the final model of a Gradient Boosting Regression model, with hyperparameters of a learning rate = 0.25, and 400 estimators.

The predicted price for a new house (or custom house) can be calculated using the co . de in the *Attachments*:

$$\boldsymbol{rescale\_price}(\boldsymbol{gradboost\_final\_model.predict}(\boldsymbol{x}))$$

where $\boldsymbol{x}$ is the new data point being used to predict a price form.

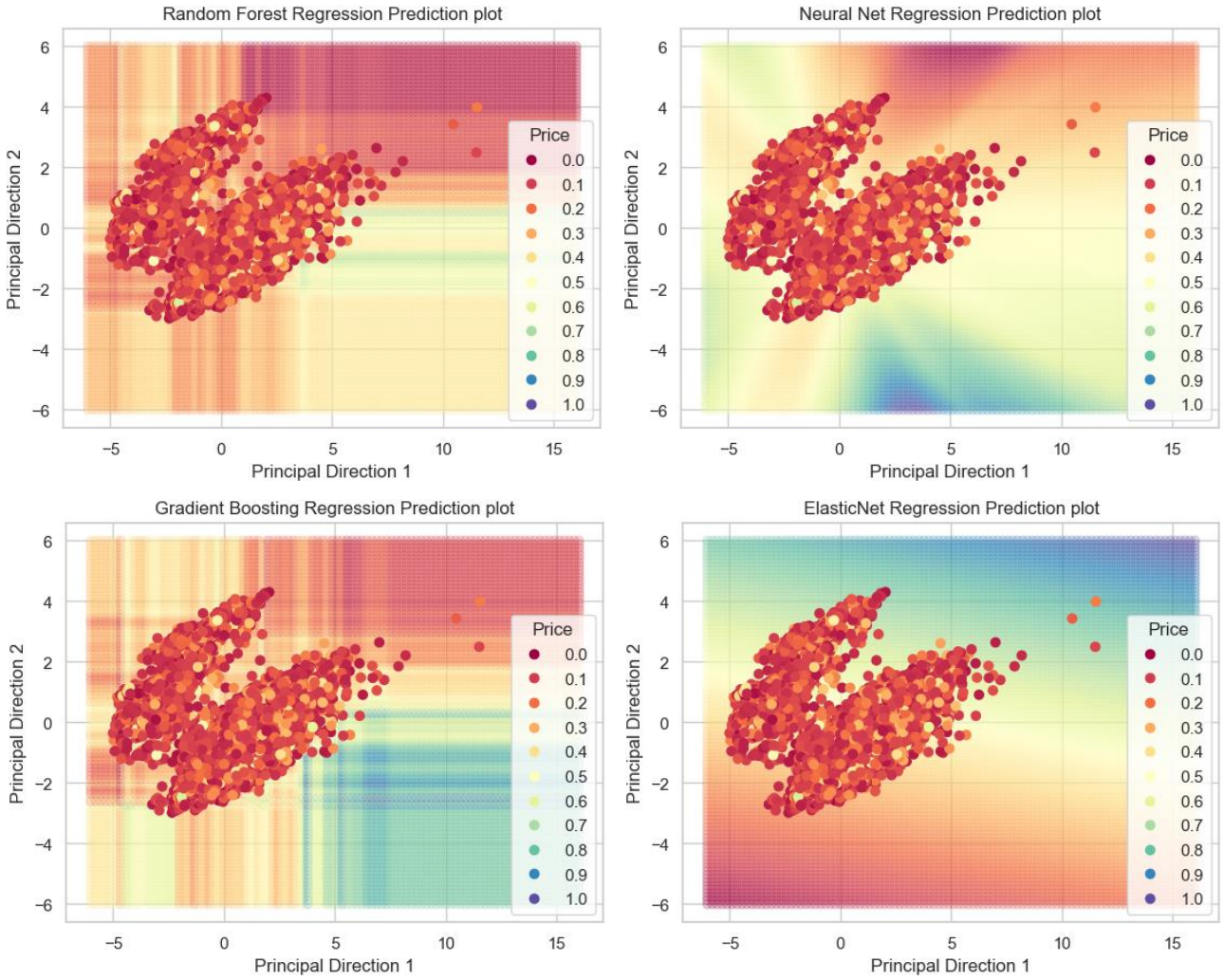
This Gradient Boosting Regression model is a high performing model with a $\mathbf{r}^2$ value of 0.838, meaning it is able to describe 83.8% of the variability in price using the variability in the predictor variables.

4.7 Visualization of Solution Spaces

As three of the regression models have similar predictive performance in the same order of magnitude, it is useful to visualize their solution spaces using two principal directions in an attempt to understand model behavior. Please refer to **Figure 6** below. A thorough analysis of the projection of predictors into PCA 2-dimensional space and solution space visualization for all seven regression models can be viewed in the *Attachments*.

In PCA 2-dimensional space, the true values in the dataset becomes split into two clusters resembling a 'butterfly' in a diagonal alignment. The true values are predominantly houses on the lower-range of the prices in the 36-month dataset. It can be seen that the Gradient Boosting model attempts to place the cheapest houses in the top-right corner of the plot and the most expensive properties in the lower right-hand corner of the plot, with low-to-mid valued properties in the center of the plot. It can be seen that the Random Forest model exhibits similar behavior, but lacks a region of high-value properties in visible section. Boosting algorithms and Random Forest tend to exhibit similar behaviors in two dimensional spaces as they are ensemble methods composed of hundreds of basic (often binary) models. This explains why the solution space takes on a kind-of woven pattern, where each strand represents a basic model or tree. On the other hand, the Neural Network model exhibits a more unique solution space, with mid-value properties in the middle of the plot, low value houses in the middle-top and top-right of the plot, and the most expensive properties in the middle-bottom and bottom-right of the plot. Similarities in the solution spaces can been identified and this helps to understand why these three models edge out above the other four model architectures. For comparison, the poorly-performing Elastic Net model's solution space displays the opposite behavior with inversed relationships.

Figure 6 — 2D Visualization of Solution spaces for Random Forest, Gradient Boosting, Neural Network and Elastic Net models



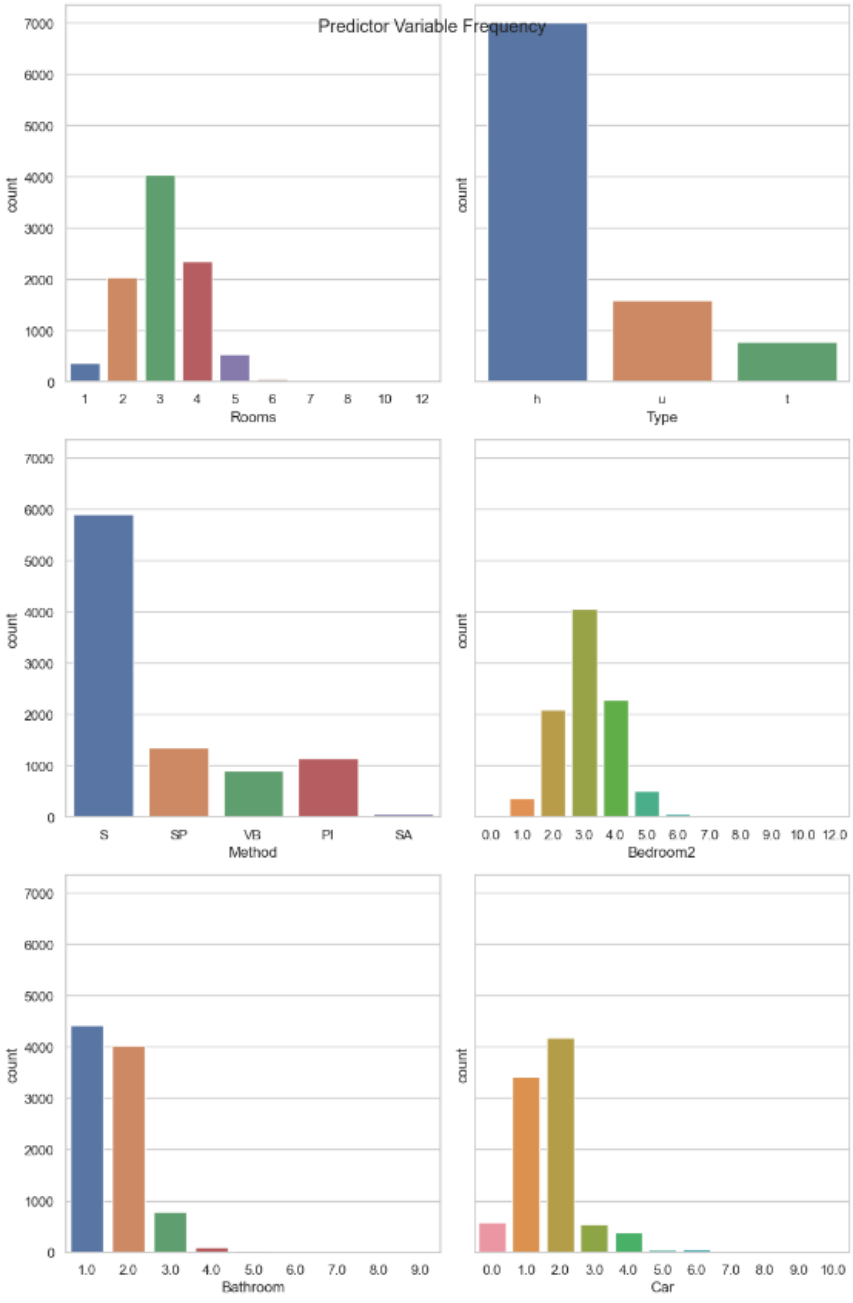
5 REFERENCE

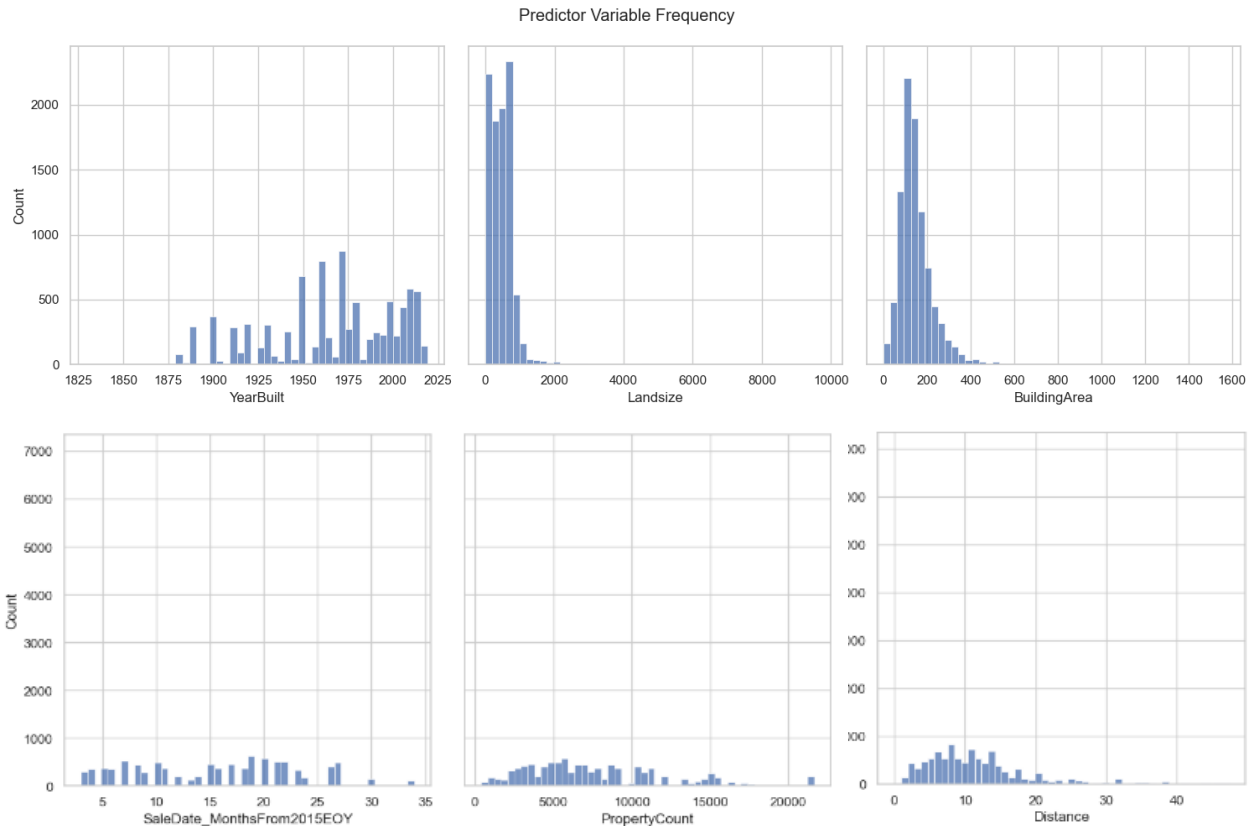
1. Australian Bureau of Statistics, O'Neill, A. (2023) *Australia: Degree of urbanization 1960-2021, Statista*. Available at: https://www.statista.com/statistics/260498/degree-of-urbanization-in-australia/(Accessed: April 21, 2023).
2. Chowdhury, I. (2022) *Housing Affordability Crisis: The elephant in the room Stomping Australians, The Ethics Centre*. Available at: https://tinyurl.com/2p9h3das (Accessed: April 21, 2023).
3. Municipal Association of Victoria (2023) *Victorian councils map, Vic Councils*. Vic Councils. Available at: https://www.viccouncils.asn.au/find-your-council/council-map (Accessed: April 25, 2023).
4. Pino, T. (2018) *Melbourne housing market, Kaggle*. Available at: https://www.kaggle.com/datasets/anthonypino/melbourne-housing-market?select=Melbourne_housing_FULL.csv (Accessed: April 12, 2023).

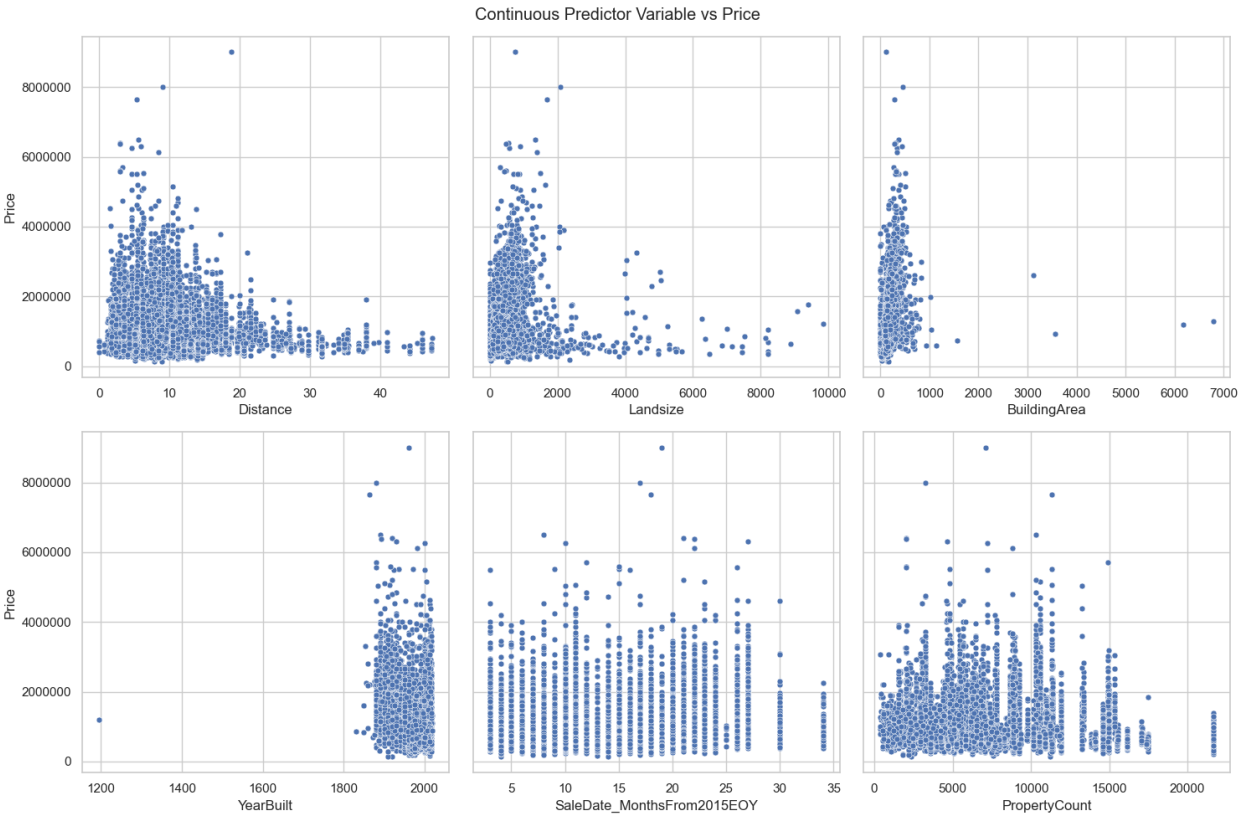
5. Stevens, C. F. (1974). *On the Variability of Demand for Families of Items*. Operational Research Quarterly (1970-1977), 25(3), 411–419. https://doi.org/10.2307/3007928
6. Visontay, E. (2023) *Living with density: Will Australia's housing crisis finally change the way its cities work?*, *The Guardian*. Guardian News and Media. Available at: https://tinyurl.com/y3xapv5m (Accessed: April 22, 2023).

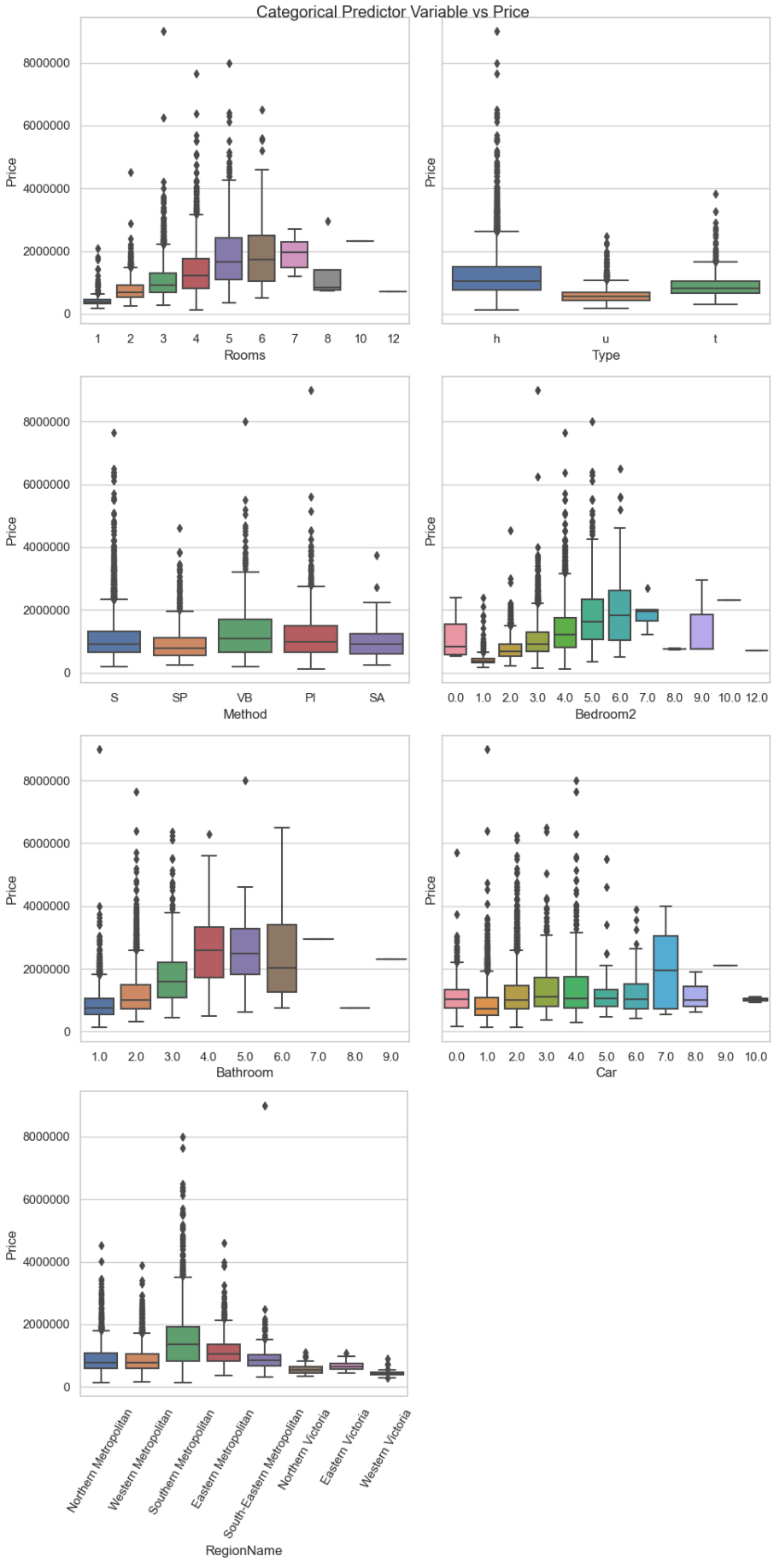
6 APPENDIX

6.1 Distribution Analyses




Predictor Variable Frequency
Count
YearBuilt
Landsize
BuildingArea
SaleDate_MonthsFrom2015EOY
PropertyCount
Distance


Continuous Predictor Variable vs Price
Price
Distance
Landsize
BuildingArea
YearBuilt
SaleDate_MonthsFrom2015EOY
PropertyCount


Categorical Predictor Variable vs Price
Price
Rooms
Type
Method
Bedroom2
Bathroom
Car
RegionName
Northern Metropolitan
Western Metropolitan
Southern Metropolitan
Eastern Metropolitan
South-Eastern Metropolitan
Northern Victoria
Eastern Victoria
Western Victoria