Contents

[Introduction 2](#_Toc182666157)

[Characterisation of the dataset 3](#_Toc182666158)

[Pre-processing 3](#_Toc182666159)

[Hyperparameter Tuning 4](#_Toc182666160)

[Results 4](#_Toc182666161)

[Conclusion 7](#_Toc182666162)

[References 8](#_Toc182666163)

[Github Link 8](#_Toc182666164)

# Introduction

It is well known that the global real estate market has experienced significant changes in property prices due to economic, social and political factors. This has led to an interest in predictive property price analysis, both for buyers and sellers. For this study, a database of some properties in London that are for sale in the year 2024 has been obtained. The objective of this analysis is to predict the price of each property in terms of its characteristics, such as the number of bathrooms, bedrooms, size, location, etc.

Regression models such as linear regression, decision tree and random forest will be applied to predict the price of a new property, and with this it will be possible to justify which techniques are more effective for price analysis, thus building a basis for future work.

Practicing with real data will contribute to my analytical development and allow me to demonstrate skills in predictive analytics. In addition, working with a dataset based on the real estate market will strengthen my logic in implementing models that study data patterns using regression concepts.

# Characterisation of the dataset

The dataset is based on 9 columns which talk about the date when the houses were published, the description and characterisation of the property, the type of property, its size in feet, bathrooms and rooms available and finally its price, and counts 1019 rows. The dataset had different null values in some variables and also the data were dispersed in terms of square meters and price which resulted in a certain number of outliers in the dataset. Some columns were biased while others had a normal distribution. The relationship of variables shown in the correlation matrix indicated a vague relationship between different variables.

# Pre-processing

**Data Cleaning**

* Null data was removed from the addedOn, sizeSqFeetMax, bedrooms and bathrooms columns.
* Because the price was in text format, the ‘£’ symbol was removed from the Price column and the column was converted to float type.
* The outliers were not eliminated as they could have a high significance in terms of values related to very large or very small houses.

**Data Transformation**

* Dummy values were obtained from the propertyType column and then transformed to Boolean, thus having more variables to contribute to the model in order to improve accuracy.
* Logarithmic transformation was applied to the price variable to improve the linear relationship.
* Finally, the StandardScaler was applied to the bathrooms, bedrooms, and sizeSqFeetMax variables to reduce their variability and improve the performance of the model

**Model under-fitting by 10%, 15% and 25% with respect to the model training data.**

The model has been tested with at least 10%, 15% and 25% of the data. Due to this under-fitting in the training data, it is observed that the model may not adequately learn the relationships between variables, as the model is less able to capture general patterns. This leads to more inaccurate predictions, which is reflected in a higher MSE and a lower R2 score.

# Hyperparameter Tuning

Model parameter tuning is important as it allows the model to better fit the characteristics of the data. They are set before training the model to control for overfitting or underfitting, thereby improving performance metrics such as R-squared, mean squared error, etc.

Tools such as GridSearchCV can help us find the best parameters to fit the model, by testing all possible combinations of the predefined model parameters. This improves the generalisation of the data, making the model able to learn patterns from the data, but not as many as to memorise irrelevant data.

For the case of this project, the housing prices in London dataset, the GridSearchCV was used to find the best parameters for the random forest model, using the following defined parameters.

A computer code with red and green text

Description automatically generated

Providing the following result.



# Results

**Before GridSearchCV usage**

After applying different types of models such as linear regression, decision tree, random forest, KNN and support vector regression, the following results have been obtained.

A screenshot of a computer code

Description automatically generated

The best performing model was the random forest model, capturing 54.7% which indicates that it is able to explain just over half of the variation in the data, the remainder of which may be due to factors not included in the data or simply noise. Furthermore, analysing that the MSE is lower than the MAE, it can be concluded that there are no large errors in the model on a frequent basis.

Applying a 5-fold cross validation on random forest model, the following mean and standard deviation score results could be observed.

A number with numbers and circles

Description automatically generated with medium confidence

The mean score indicates that the model has a moderate relationship with the data, and has a low standard deviation, which indicates that the model is not being very sensitive to different subsets of the data during cross-validation, the model performance is consistent across folds, which is a good sign of stability.

**After GridSearchCV usage**

Using the best parameters previously mentioned, the following results have been obtained.

A number on a white background

Description automatically generated

GridSearchCV has had a positive impact on the model by optimising the hyperparameters, improving the performance of the model by being 7% more accurate in the variance of the data (R2 result) and lower MSE than before which means that the predicted data is closer to the real data.

By applying a cross-validation to our new model the following results have been obtained.

A number of numbers on a white background

Description automatically generated

This could indicate that, although the model has improved in terms of accuracy (Mean CV Score), it now has slightly more variability in its performance between folds. However, this difference is not very large, suggesting that the variability is still quite low and that the model remains relatively consistent.

# Conclusion

The analysis of London property prices allowed the application of pre-processing techniques, regression models and hyperparameter optimisation. Initially, the Random Forest model explained 54.7% of the variation in the data, but after fitting hyperparameters with GridSearchCV, its performance improved to 62.5% (R²) with a lower MSE, indicating more accurate predictions without overfitting the data.

Cross-validation showed consistency in model performance, with a low standard deviation between folds, confirming its stability and generalisability. This demonstrates that proper preprocessing and hyperparameter tuning are fundamental to optimise predictive models and achieve more accurate and reliable results.

It has also been seen that for this dataset the relationship of house price to size square feet is at least 73%, which is arguably the variable that most affects the prediction of our model.

# References

Kanchana1990 (2024). *Real Estate Data London 2024*. [online] Kaggle.com. Available at: <https://www.kaggle.com/datasets/kanchana1990/real-estate-data-london-2024>.

Mitchell, T. (1997). Machine Learning. [online] Available at: <https://www.cin.ufpe.br/~cavmj/Machine%20-%20Learning%20-%20Tom%20Mitchell.pdf>.

Molin, S. (2021). *Google Books*. [online] Google.ie. Available at: https://www.google.ie/books/edition/Hands\_On\_Data\_Analysis\_with\_Pandas/Eh4sEAAAQBAJ?hl=en&gbpv=1&dq=pandas+and+data+preparation&printsec=frontco

# Github Link

https://github.com/CCT-Dublin/ca1assignment-50-AntonioGiambra