Data Science: London Housing market

Hugo-Paul Damestoy

Table of Contents

[Summary 3](#_Toc58783271)

[Introduction 3](#_Toc58783272)

[Body of the report 4](#_Toc58783273)

[Exploratory Data Analysis: 4](#_Toc58783274)

[Tuning models 7](#_Toc58783275)

[Importance of variables 8](#_Toc58783276)

[Stacking and my final decision 11](#_Toc58783277)

[Elizabeth Line (Crossrail) 11](#_Toc58783278)

[Conclusion and Recommendations 12](#_Toc58783279)

[Appendices 13](#_Toc58783280)

# Summary

This report focuses on creating the most efficient algorithms to predict the price of houses from the out-of-sample data, in order to be able to identify the best houses to invest in. To create the best and more robust algorithms it was necessary to tune some of them and to stack them in order to reach the lowest value of RMSE possible. First thing to do was to separate my training set in 2 parts, to create training data where I will train my algorithms and validation data where I will check my results and look for overfitting. Once that done, I start by visualizing the variables and try to understand which one seems to have the more impact on price. Then I tune 4 models (linear regression, tree, knn and random forest) in order to reach the lowest RMSE in each of these one. Once I got robust solution in each of these models, I stack them, to create an even better algorithm. I will use the best one to predict the real value of the houses in the out of sample set in order to reach the highest profit possible. Here we do not take into account the time value of money for the sake of simplicity.

# Introduction

During the last 20 years, the average price for a square meter in London has plummet to a record of almost 7,000£, an increase of more than 300%. In one hand it creates a huge gap between the Londoner purchasing power and the price of houses. In another hand it makes this market as a really attractive one to investors. For them, it’s really interesting to be able to judge/predict the real prices of houses using some relevant analytical tools. During this project I will aim to predict values from the London housing market, enabling investors to make the maximum profit out of this situation. Our training dataset comprises information about 14,000 houses that have been sold in 2019 in London taken from Land Registry’s Price Paid Data which is a data set which collect information about the property sale in England and Wales since 1995. Then some technical (total\_floor\_area, distance to closest station) and geographical (postcode) information were added to this data set. All these information’s will help us understand the impact of different variable such as localization, size, type on the price of each houses. We will be using our second data set “out of sample” which is composed of 2,000 rows, each having exactly the same information (variable) about houses, just here I do not now the actual price but only the asking one and here I will use my solution in order to predict the actual price.

Through this project, I developed 4 solution (Linear regression, random Forrest, knn, tree, actually 5 by stacking them) in order to predict the actual prices of each houses in the out of sample set. In my conclusion, I will further elaborate how can I use these insights/algorithms in order to take the best decision.

# Body of the report

## Exploratory Data Analysis:

As in any other analysis, my first step was to clean the data and to start to understand them. In order to do so I visualized data by using plots to comprehend and evaluate information in a more suitable manner. First things that I did was to check how sparse was the data and if there are any outliers. From the following graph we can see that the distribution is left skewed and that we have some high values, but they all seem to be actual real values, so we keep them. Note: for the purpose of visualisation : I don’t show maximum values, most expensive house has been sold for more than 10 000 000£.

Chart, histogram

Description automatically generated

Then I started to try to understand the impact on each variable on price. To do so I try to answer question like that: Can I visualize a pattern between price and floor size? Between price and district? Here by looking at the following graphs, at the regression line for floor area and at the different boxplot for district we can observe an impact between these variables and price.

Chart, box and whisker chart

Description automatically generatedChart, scatter chart

Description automatically generated

Once I’ve done that kind of graph for a majority of variables, I started to understand how each of them impacted price and I start thinking of interaction terms that can help me have the better model, if some of these variables can have an explicability power multiplicated if we treat them together. Here I found some really important trends. Next graph represents the combined impact of district and total floor area on prices. We observe that the increase in total floor area has a greater impact on price if the district from which the house is coming from is an expensive district, which is perfectly logical. The coefficient of the regression line is much higher for district like Kensington and Westminster than for district like Croydon. I also used some interaction between floor are and property type and between London zone and floor area.

Chart, scatter chart

Description automatically generated

I also did a correlation table in order to understand how each numeric variable influence price but also how each of them influences each other. It permits me to identify strong positive correlation between some variables like total floor area and number of rooms which is perfectly logical, but some were not that logical like a score of 0.53 between price and current co2 emission. I mainly used it in order to perform my first regression model.

*Chart, scatter chart

Description automatically generated*

Last thing I want to talk about in this part, is to transform variables like London zone in factorial variables rather than numeric, which from me and the following graph would have made perfect sense. But the effect of that changes was only beneficial in the regression model, so I choose to keep this variable as a numeric one.

Chart, box and whisker chart

Description automatically generated

The next step was to build the model, tuning the parameter and engineering features by adding extra variables which would lead to improve my model performance

## Tuning models

Now that I have a good understanding of variables, I start to build my 4 models. I want to be able to test my data on a part of the testing set, so I start by seeding my data and then created a 75%-25% split of the data from the train set, 75% of them are now my train data and 25% of them are now my validation data.

I already know that I’m going to stack my 4 models, so I should use the same control parameter for all of them. I start by fixing my control parameter, cross validation with 10 folds and without repetition, I will only keep the result for the best model and predict only using the best model. My first model is going to be a Linear regression here I’m going to follow a forward selection which means I will start by adding the variables which has the highest positive or negative correlation with my variable of interest. Each time I add a new variable I look at the p-value or t\_stat. If the p\_value is inferior to 0,05 (can also check if the absolute value of t-stat is superior to 2) and the adjust R squared increase, I keep this value and continue doing that. Then we start to use the factorial variable, starting by the one which seems to have the more impact based on our visualisation, and then I try to add some interaction term. I also used the graph which shows the importance of variable (going to speak about this one in the next part) in order to make my choices. Once I reach the highest R squared and I minimised the variables which are not significant in my model, I use my model to predict the prices of houses in my validation set. In this step I’m looking for a potential overfitting. However, the adjust R Squared (or you can look at the RMSE which should be the lowest possible) is even higher in my validation set which means that my model seems to perform even better in my validation set (check appendix).

My second model is a tree, in this one I follow the same steps in order to add variables in my model. However here I have others parameter to tune, the most important one is the complexity parameter which is mainly the minimum level of information that a split must have in order to be possible. Here we should take care of overfitting, a small cp will permit a lot of split and then might lead to overfitting, a high cp will lead to few splits decreasing the model performance. We can also in this model tune parameter like min node size, max node size, depth etc…To find the best cp for my model I try multiple of them, and then choose the one that lead to the highest R squared in my model. My model is not overfitting as the R squared is a bit higher in the validation data.

My third model is a knn, here it’s the same principle as the tree, but now the parameter that I want to tune is K which is the number of neighbour that we use in order to assign a price to an observation. I follow the same principle to add variables and I reach a lower RMSE when k=5. Thanks to the validation set I can conclude that with a negative difference in adjusted R squared of 0.2 my model does not overfit or only overfit by a small proportion.

My last model is a random forest, which is a much more complicated model to tune, because you now have 3 parameters to tune mtry, splitrule and min. node. Size. Random forest is mainly a forest composed of many trees (second model that I used). As I said tuning random forest is much more complicated than other model and it’s also much more time-consuming because it’s taking a long time to compute and run the model. After, a bit more than 20 tries I arrive at one model which is the best one of my 4ths, using mtry=3, splitrule =” variance” and min. node. Size =5, I used also fewer variables in this one cause adding too much of them or interaction terms was leading to a decrease the adjusted R squared or overfitting in the validation set.

## Importance of variables

The importance of variables is different in our model, so I will treat the importance variable of each model separately and then try to draw a conclusion:

Linear Regression: In this model the variable which has the highest importance seems to be average income. However, I feel that the variable with the most explicability power is floor area or the interaction between total\_floor\_area and district, which is even more true when we are speaking about expensive areas, we can see that 3 of these variables are present in our top 10. The main explicative variable is total floor area London zone and district (8 variables out of 10 are a component or interaction of them) plus average income which are not surprising. CO2\_emissions potential is a much more surprising candidate; his place here might be dur to his strong correlation with total floor area (present in 6 of these variables).

Chart

Description automatically generated

Tree: In this model, we have almost the same variable which are really important, but the new top one is somewhat surprising: longitude. However, if I think of it, the centre of London is mainly vertical on a map, which means in order to determine where a price should be expensive a tree can perfectly say between this longitude and this one price should be higher (because it’s at the centre of London)

Chart

Description automatically generated

KNN: In this model it’s quite surprising that we don’t see any of our interaction variable even if they increase our adjust R squared. Total floor area is still the most explainable variable, but now it’s mainly variables with the highest correlation with total floor area such as co2 emission or number habitable rooms which have a lot of importance in this model.

Chart

Description automatically generated

Random Forest: Total floor area is again the most explicative variable here, then its London zone followed by longitude average income and latitude, property type has only a small explicability power. All of these variables are not too surprising

Chart

Description automatically generated

In general variables which has the most importance to predict prices are in that order total floor area, district average income, and in London zone. We can add to that, in different models, some interaction term and variables which have high correlation with total floor area

## Stacking and my final decision

Then I stack my 4 models, using the caretEnsemble library, looking at the p-value of my stack model I found that my tree model(rpart) doesn’t have enough power and it is not significant. I stack my model again without the tree and here I reach my best solution which has the lowest RMSE. Using the validation set, I can conclude that there is no overfitting. I used my last model to predict the value of the house and then use a function ((predicted price-asking price)/asking price\*100) in order to compute the predicted profit in % of investment. I select the top 200 houses with the best predicted profit.

# Elizabeth Line (Crossrail)

When I read about the Elizabeth line, my first idea was to really understand if the actual time of transport to the centre of London is an actual powerful variable to predict the price of houses. In order to check that I will create a variable which compute the average time from one house to one of the main tube stations (the closest one) in central London. I give a not exhaustive list in the following brackets (King’s cross, Victoria and Oxford circus). This new variable will replace the variable about transportation like distance to station or number of tube rails. I will use this variable in my model in order to determine if the actual time to the centre has a real impact on the price of a house. If it does, I will try to compute a variable which shows the gain of time between the precedent variable calculated and the time to go to one of the main tube stations without using Crossrail (this difference will mainly be 0 for district where Crossrail is not present). I then will try to predict prices using these 2 variables: time to main station + difference.

Another idea if time to main station is significant will be to look for district/county/postcode where Crossrail pass and that have one of the following conditions:

-Lot of houses with high total floor area

-average income higher than what expected

As they were the 2 most explicative and not geographical variable in our model, and due to the fact that Crossrail is reducing the constraint of geo-localisation, these zones might be the first one to experience an increase on price.

# Conclusion and Recommendations

In this report I show you how I reach a decent solution on which house investors should invest in term of optimal profit return in term of %. I did that using a linear regression, a tree, a knn and a random forest and by then stacking them. My best model is the one stacking knn, random forest and linear regression. But I reach good solution using only linear regression or Random Forrest: adjust R squared superior to 0.83. That’s mainly due to the fact I was able to train my model using explicative variable such as floor area district or average income and that I created interacting variables from these already powerful variables.

I want to make clear that here, I choose the 200 houses based on an optimal %profit return and not absolute values. Technically it takes in account the assumption that: you, investors want to minimise risk and you care about the maximum amount invested in these houses. If you actually don’t care about that amount and want to make the maximum profit in £ you should invest in the houses with the highest return in £.

I give you here a powerful tool to choose wisely the houses, but risk still exist. The adjusted R squared of my model reach 0.8634 on the train data and 0.8669 on the validation data which leads to 0.75\*0.8634+0.25\*0.8669= 0.8643. That means that my model explains 86.43% of the data, which is in statistics a high score, but incertitude always exists, and I cannot guarantee you that by investing in this 200 houses you will reach the 68.99% return predicted. Moreover, all of this reasoning takes the assumption that the training set is representative of the out of sample set. Nevertheless, this solution is robust and should lead you to an interesting return on investment.

# Appendices

**Score:**

**Linear Regression:**

Text

Description automatically generated

Graphical user interface, application

Description automatically generated

**Tree:**

Table

Description automatically generated

A picture containing application

Description automatically generated

**KNN:**

**Text, letter

Description automatically generated**

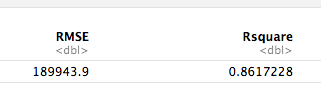
**Graphical user interface, application

Description automatically generated**

**Random Forrest:**

**Text

Description automatically generated**

****

**STACK (4models)Table

Description automatically generated**

**A picture containing chart

Description automatically generated**

**Stack (3models)**

**Table

Description automatically generated**

**A picture containing chart

Description automatically generated**