PREDICTING REAL ESTATE PRICES IN MONTREAL USING MACHINE LEARNING TREE-BASED MODELS

Maria Kędzierska
Faculty of Mathematics and Information Technology
Warsaw University of Technology
Warsaw, Poland
https://github.com/kaluskam

Szymon Gut
Faculty of Mathematics and Information Technology
Warsaw University of Technology
Warsaw, Poland
https://github.com/Szymon-Gut

Wiktor Jakubowski
Faculty of Mathematics and Information Technology
Warsaw University of Technology
Warsaw, Poland
https://github.com/WJakubowsk

Maciej Orsłowski
Faculty of Mathematics and Information Technology
Warsaw University of Technology
Warsaw, Poland
https://github.com/maciejors

June 9, 2022

ABSTRACT

Our goal is to utilize Machine Learning Models in analyzing real estate market in Montreal, Canada. The problem we are dealing with is to predict the value of real assets in one of the biggest and most popular cities in Canada, based on collected data. To do that, we use the *R* programming language with its ML packages such as *caret* and we use several tree-based Machine Learning models: Random Forrest, Extreme Gradient Boost (XGBoost), CatBoost and Light Gradient Boosting Machine (Light GBM). The performance of these models is examined with several metrics to ensure great results. Moreover, we value the explainability of ML models, becuase it is important not to treat them as *black boxes* in order to avoid unwanted insights. Because of that, we inspect the best models' behaviour using techniques of eXplained Artificial Intelligence (XAI), implemented in *DALEX* package for *R*. The outcome of this case study is ML explainable models, that can predict prices in Montreal with decent results. In short, this article walks you through from raw data and zero insights, to Machine Learning models that can utilize this data and give explainable predictions on real estate in the second biggest city of Canada, Montreal.

Keywords Machine Learning · Explainable Artificial Intelligence · price prediction · housing market · tree-based models

1 Introduction

Machine Learning is an indispensable part of the twenty-first century technology's development. Machine Learning methods are helpful in many fields of our daily life. Therefore we decided to use these methods for solving our problem, which is the real estate price predictions in Montreal, Canada. The task is particularly interesting, because it involves applying AI methods in real life problems. We figured the research about prices of houses and other properties in the second biggest agglomeration in Canada should raise interesting insights of housing market operations. We gained some information about that from Richard Green [1999].

In order to predict the value of real estate properly, we want to utilize several Machine Learning models implemented in *R* programming language: Random Forrest, Extreme Gradient Boost (XGBoost), CatBoost and Light Gradient Boosting Machine (Light GBM). Note that all of them are tree-based, meaning they use a series of if-then rules to generate predictions from one or more decision trees.

As a result, we want to obtain stable models that have been inspected by us using several evaluation metrics. Moreover, we will put an impact on explainability of our models. We do not want them to be *blackboxes*, that make their decisions based on irrational conclusions from human point of view. To avoid that, these models will be explained with explainable Artificial Intelligence methods.

If this sounds interesting enough, let us immerse deeper into our project.

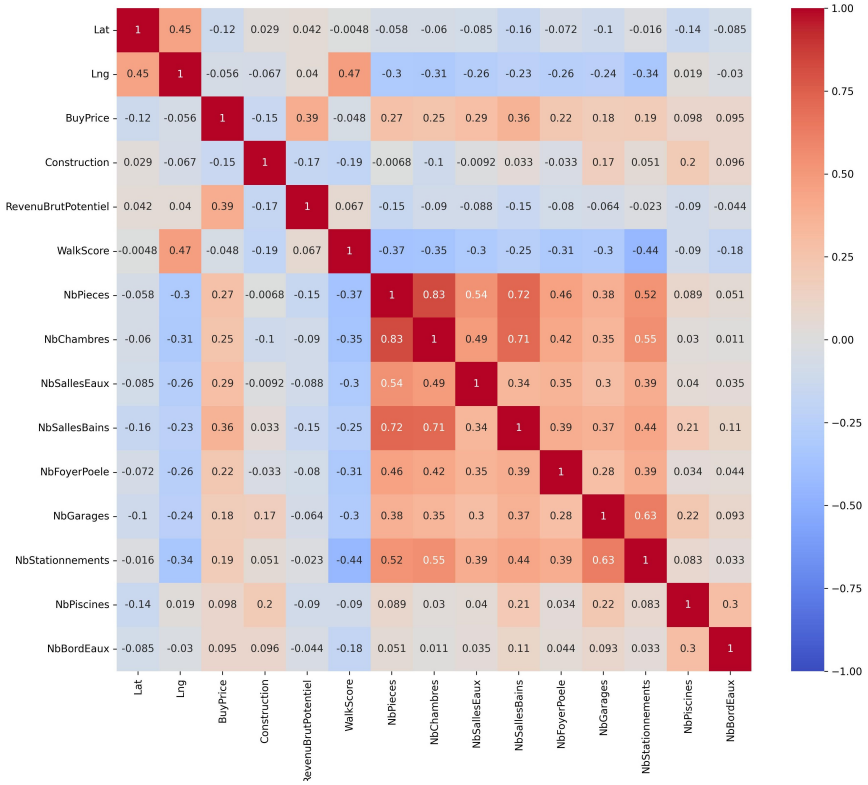
2 Data

The data has been obtained by the authors of Nissan Pow [2014] by scraping the relevant websites and is now available on their Github. The dataset consists of six data frames. Each of them contains information related to Montreal.

- listings.csv - key attributes for real estates sold in Montreal, e.g. price, number of rooms or information if estate has garrage.
- policeCoord.csv and firestatsions.csv - coordinates of the police and fire departments.
- montreal_hpi.csv - Montreal House Price Index.
- sociodemo.csv - demographical data.
- extradata.csv - extension to listings.csv, contains mainly information about locations available in proximity.

The core data frame is 'listings.csv', as it contains key attributes for real estates sold in Montreal. During data exploratory analysis we have discovered that some columns contain largely nulls, so we decided to drop them. Additionally, columns with the same value for all records were also rejected as they did not provide any information. The variable we want to predict is **BuyPrice**, we observed that some records contained the price per foot or square meter while the rest contained the total price of the property. Due to the lack of information about the living area, we were forced to drop the records, which did not represent the total price. Fortunately, there were not many such records. As shows the correlation matrix 2 the **BuyPrice** is mostly correlated with the potential gross income, the number of bathrooms, bedrooms, and rooms.

Figure 1: Correlation matrix for columns from listings data frame.



The next data frames that were analyzed were "policeCoord.csv" and "firestations.csv". These data frames only contained the coordinates of the police and fire departments. It seems useful to calculate the distance to the nearest police and fire station for each property, according to Axel Viktor Heymana [2019].

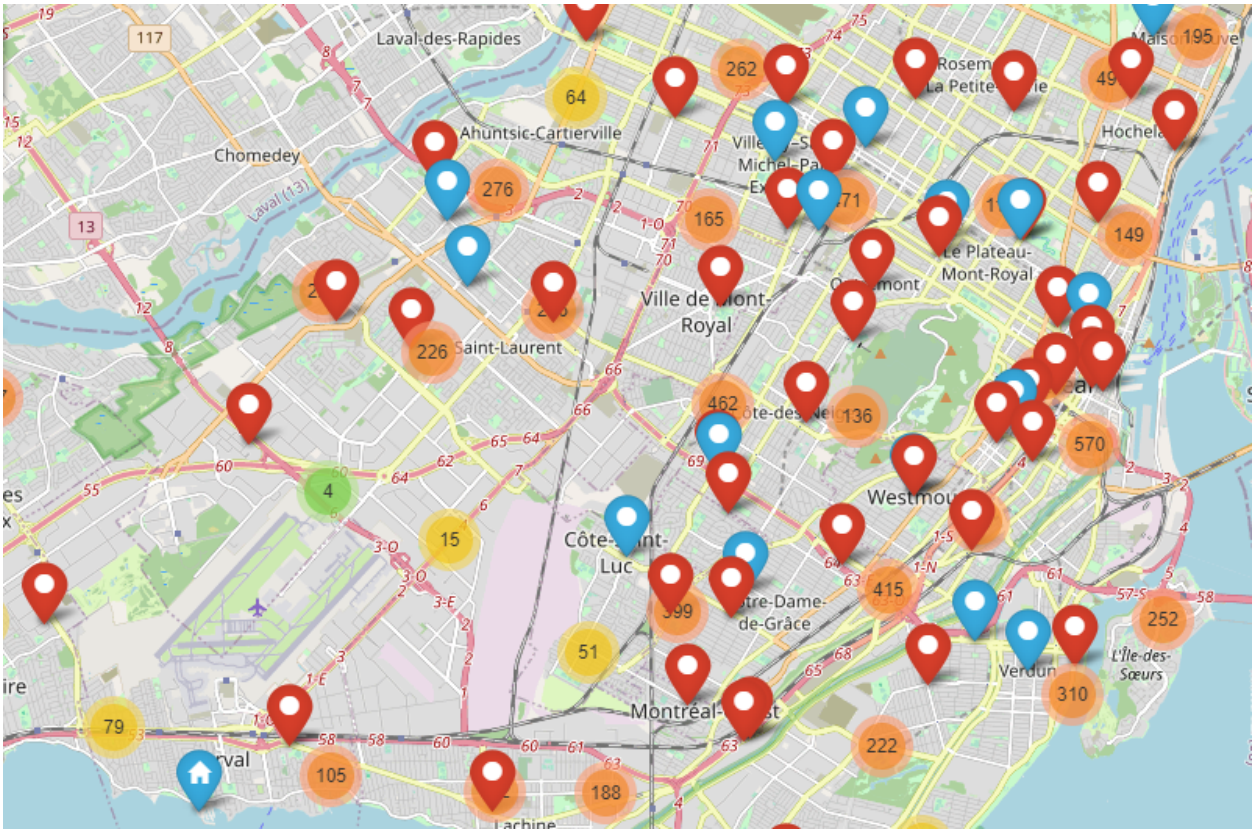


Figure 2: Montreal map with red markers for fire stations and blue markers for police stations.

Next, we examined the file "montreal_hpi.csv" which contained the house price index measures. Due to the significant correlation (greater than 0.99) between the variables, we concluded that only one column should be used from the mentioned dataframe.

In order to add socio-demographic data, we have analyzed 'sociodemo.csv'. A data frame with information about population, nearby schools, average income, etc.

The last data frame was 'extra_data.csv' which provided additional information for some properties in 'listings.csv'. Unfortunately, the vast majority of its contents were null values. However, some columns, such as LivingArea, found themselves truly relevant.

3 Selected models

After the exact exploratory data analysis (EDA) and exploring machine learning models that could be useful in our task from Quang Truong [2020], we started next step which was building and testing different models. We have built several RandomForest, XGBoost, LightGBM and CatBoost models. Our models were trained using Grid Search and Random Search. Then were selected only the best models based on RMSE. For each alghorithm mentioned before we have selected five models with different hyperparameters which have reached the best scores. Then we have tried Principal Component Analysis (PCA) to reduce dimension of the dataset and this significantly improved scores of our models. Below is a table with the best scores on test data sets from the four selected algorithms.

RMSE	RandomForest	XGBoost	LightGBM	CatBoost
with PCA	87 740.23	360.06	74 396.47	46 023.39
without PCA	334 091.74	228890.90	282 358.90	246 424.10

Table 1: Results for the best models for each model type.

All hyperparameters were set to avoid overfitting and underfitting so models presented above have similar scores on training and test data sets. Then we selected the best algorithm for our models, which, as we can see in the table above, turned out to be XGBoost. Below we can see the best 3 models built on XGboost's algorithm and their hyperparameters.

Model	RMSE train	RMSE test	booster	eta	max_depth	gamma	min_child_weight	subsample	colsample_bytree
xgboost 1	101.90	360.07	gbtree	0.05	7	0	1	1	1
xgboost 2	430.97	881.18	gbtree	0.09	8	4	1	1	1
xgboost 3	808.91	1 477.11	gbtree	0.12	6	6	1	1	1

Table 2: Results for the best models used on PCA data with their parameters.

Next, we created residual plots to check how our predictions looked like. What really satisfied us is that almost all predictions are very accurate. Residual plots 3 are presented below.

Figure 3: Residuals plot for Xgboost 1.

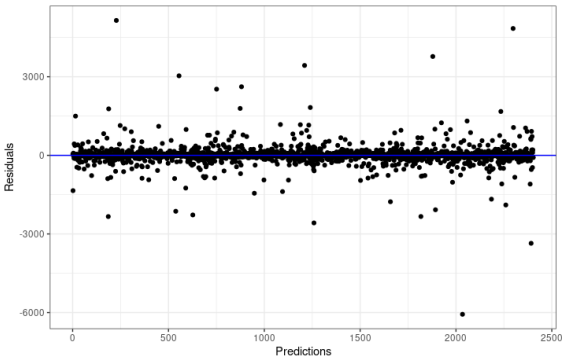


Figure 4: Residuals plot for Xgboost 2.

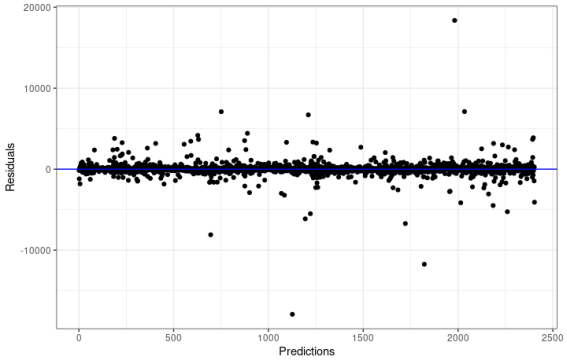
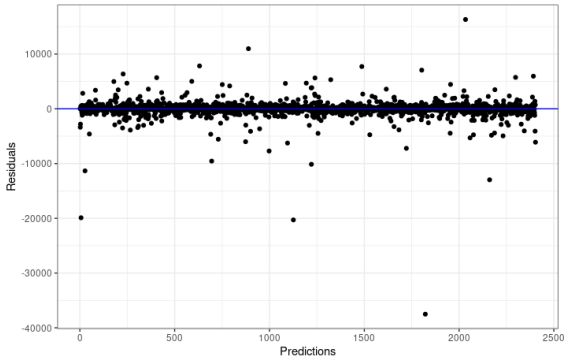


Figure 5: Residuals plot for Xgboost 3.



So as we can see these three XGBoost models were very accurate and reached very low root mean squared error (RMSE). But how about the other metrics? That was our next step in choosing the most accurate model. We have measured the predictions using mean absolute error (MAE), R-squared (R^2) and the mentioned before RMSE. In the following formulas, y_i stands for real value of i-th observation in dataset and $\hat{y}_i$ stands for the model's prediction of this value. The results are presented in a table below.

All three models achieve very good metrics and will also be further analyzed using eXplainable Artificial Intelligence (XAI) methods. But there is one change that must be performed. As we have used the PCA method to reduce dimension

Model name	RMSE test	MAE test	R2
xgboost1	360.07	135.74	0.9998625
xgboost2	881.18	289.28	0.9995856
xgboost3	1 477.11	520.25	0.9978164

Table 3: Metrics evaluation of the best models on PCA data.

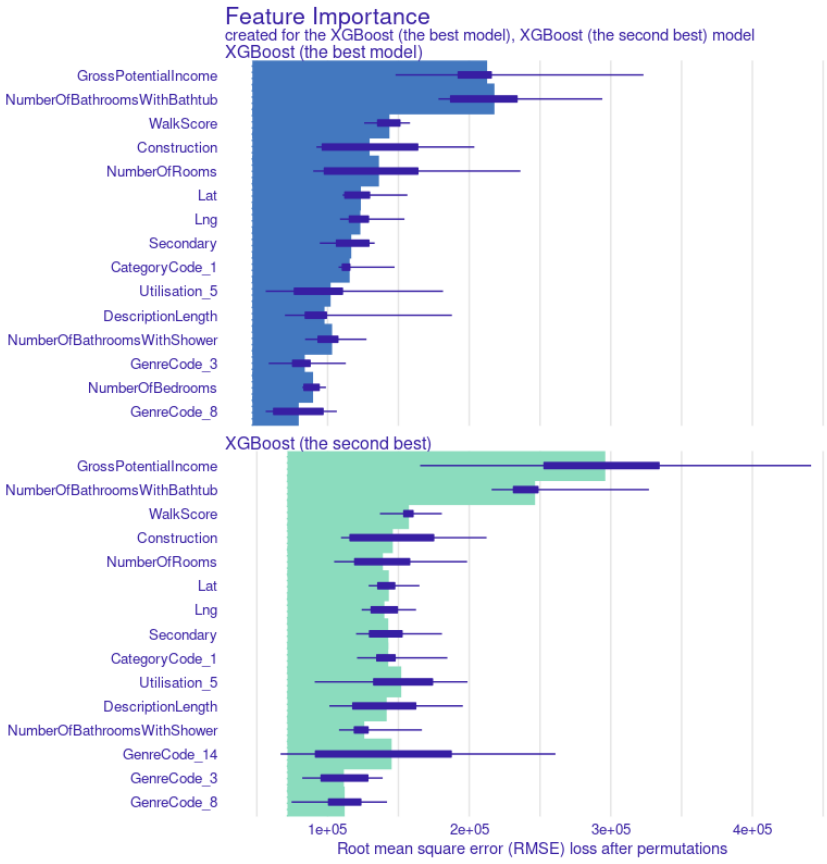
of our data sets we have lost the interpretability of our data. So we need to decide what is more important: increasing the accuracy of the prediction or preserving the interpretability and an insight into the influence of individual variables on the predictions. In the next steps, where we are analyzing our models using XAI methods, we are using the data set without PCA.

4 Explainability of models - XAI

In this section, we are taking a closer look at the inside of two of our best XGBoost models. In order to do so, we are going to use several methods to interpret the predictions of our models. All the visualizations were generated with DALEX XAI Biecek [2018] package in R language. For start, let us focus on the global overview of our algorithms, then we are going to go on to the local perspective.

4.1 Permutational feature importance

The very first thing that comes to mind when analyzing a model is inspecting which variables have the greatest impact on model's predictions. Not only should this give us a sense of understanding of our model, but we also can detect some anomalies that will just look strange for a human eye. Below, there are two plots, for xgboost1 and xgboost2 respectively, showing feature importance of the model's predictions.

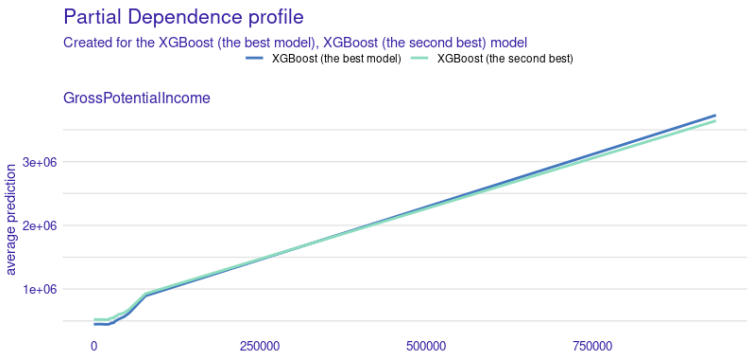


We can see that both of the inspected models have quite similar order of the most important features. There are two variables that stand out from the rest: *GrossPotentialIncome*, which is the maximum income the property can produce and *NumberOfBathroomsWithBathHub* which meaning is self-explainable. Other significant features are also *WalkScore* - metric describing how well-located the property is (and not car-dependent to go somewhere) as well as *Construction* (year of construction) and *NumberOfRooms*. From what we see, we can deduce that these features could in fact be substantial to the predictions. It makes sense to take the number of bathrooms, location or the potential income the property can produce when we estimate its value. Now that we learned about significant features for our models, let us find out, how these variables affect the average prediction. In order to do so, we will use two approaches - Partial Dependence Profile (PDP) and Accumulated Local Effects (ALE).

4.2 Partial Dependence Profile

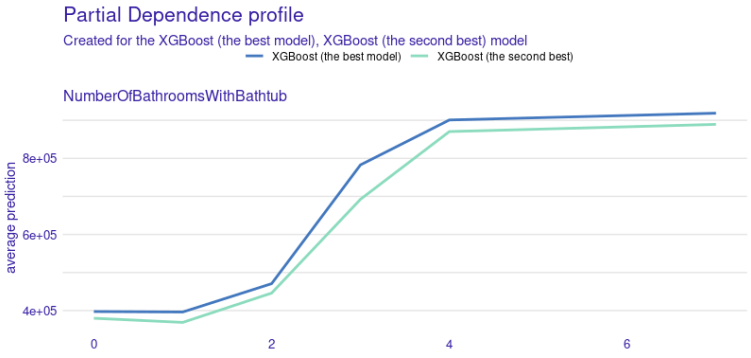
Partial Dependence Plots are used to depict the functional relationship between a small number of input variables and predictions. They show how the predictions partially depend on values of the input variables of interest. Let us see the impact of three variables: *GrossPotentialIncome*, *NumberOfBathroomsWithBathHub* and *Construction*.

4.2.1 Gross Potential Income



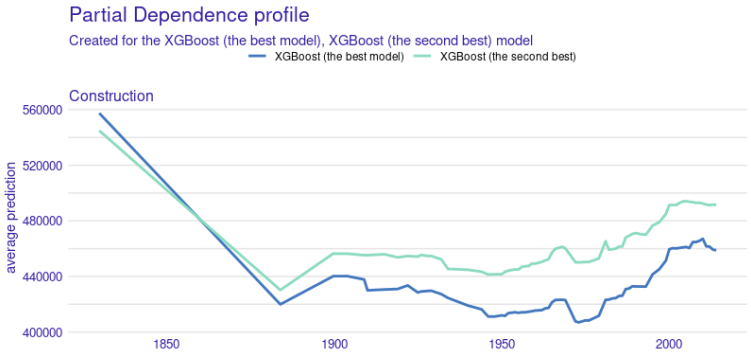
The results should not worry us and it is not surprising - properties that are more profitable (according to the Gross Potential Income index) are better priced by the models. There is a clear, linear correlation between the average prediction of price, and the GPI index.

4.2.2 Number of Bathrooms With Bathtub



The plot shows us, that the more bathrooms property has, the more expensive it is. In our opinion, it is a case not only because it is purely more comfortable to have more than one bathroom, but because it also indicates that the property is bigger - you cannot imagine a bungalow with an area of $100m^2$ having 4 bathrooms in it.

4.2.3 Year of Construction

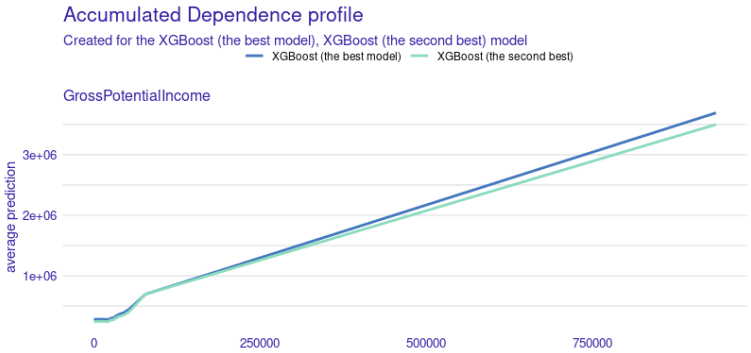


We can observe a steady fall in the preditced value of a property, when the year of construction of a real estate was later up until 1880. It can be caused by a small amount of data in these years, so the outcome is not that accurate. From 1880 onwards, the price is fluctuating, but it keeps growing, which should be a good sign as mainly newer properties are getting sold quicker and there should be a sign of inflation.

4.3 Accumulated Local Effects

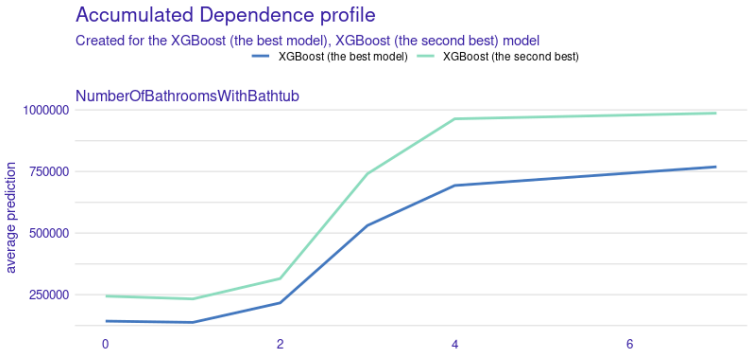
ALE plots describe how features influence the prediction of a machine learning model on average. They are a more reliable and unbiased alternative to partial dependence plots (PDPs). Again, we will analyze the influence of three variables: *GrossPotentialIncome*, *NumberOfBathroomsWithBathHub* and *Construction* on the models' average prediction .

4.3.1 Gross Potential Income



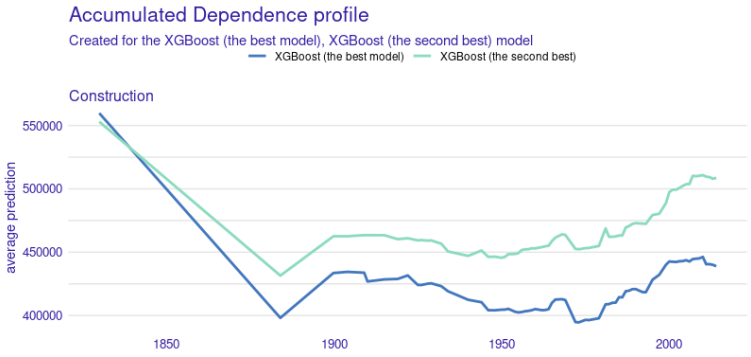
The result and insights are similar to the ones from the PDP plot - properties that are more profitable (according to the GPI index), are better priced according to the models. There is a clear, linear correlation between average prediction of price, and the GPI.

4.3.2 Number of Bathroms With BathTub



The plot shows us that in fact the more bathrooms property has, the more expensive it is - just as on the PDP plot. However, the difference of price between real estates with fewer bathrooms and properties with a higher number of them is significantly greater - from under 250.000$ with one bathroom we jump to 750.000$ so the difference is bigger than in the PD plot.

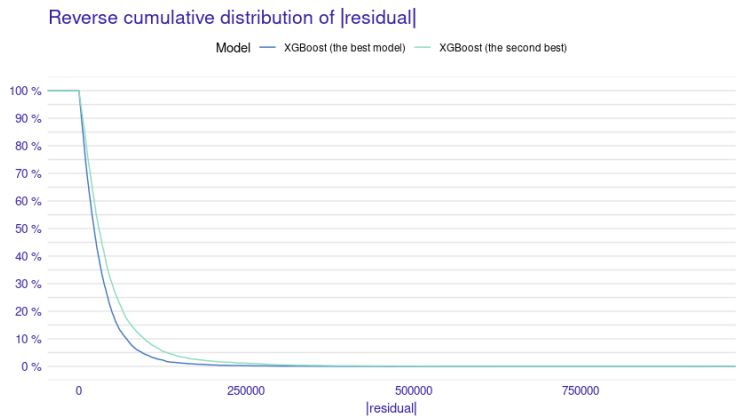
4.3.3 Year of Construction



This particular one is interesting. The second best model (the lime one on the plot) shows a similar behaviour to the PD plot. But the best xgboost model predicted that on average, the price of real estate decreases with the increase of construction year, in the period from 1900 until 1980.

4.4 Models' performance

The last part of our models' global analysis is their performance. Let us take a look at the distribution of the residuals, i.e. the absolute value of real price of property minus the predicted value.



We can see that generally the model fits well to the data. There are less than 5% of records with an error greater than 100.000$ CAD for both models. For the record, the average price of real estate prices in our dataset lay between 450.000$ CAD and 600.000$ CAD.

We have analyzed our models globally. Now it is time to see, what is going on in the smaller scale. For a local inspection, we will use Break Down and Ceteris Paribus plots.

4.5 Break Down profiles

Break Down plots show how the contributions attributed to individual explanatory variables change the mean value of prediction. We will use two different observations from our dataset and we will explore the behaviour of our two models in these two cases. Let us start with the first xgboost model and see how it performs on our two observations.

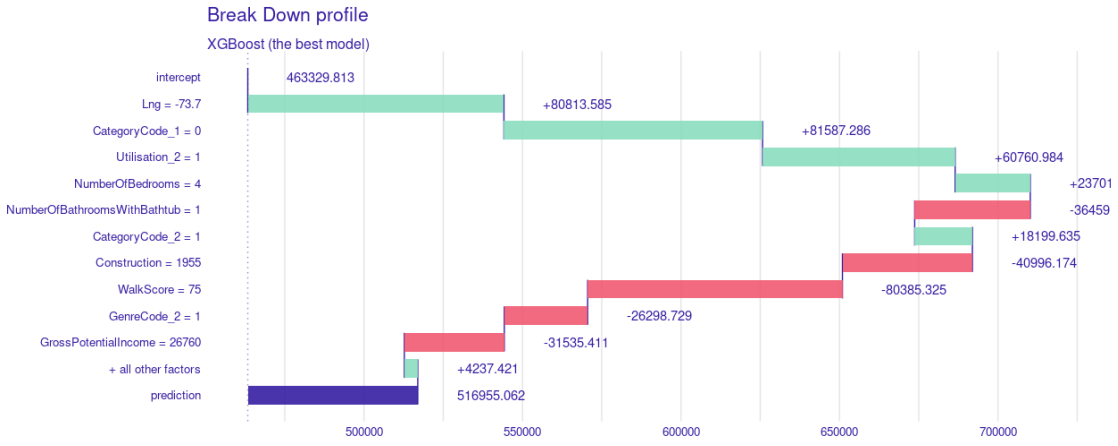


Figure 6: First observation's profile

The most important features for this observation are *Longitude* coordinates and *CategoryCode1* which corresponds to the category of the property - here it is a flag of small condominium which should lower the price.

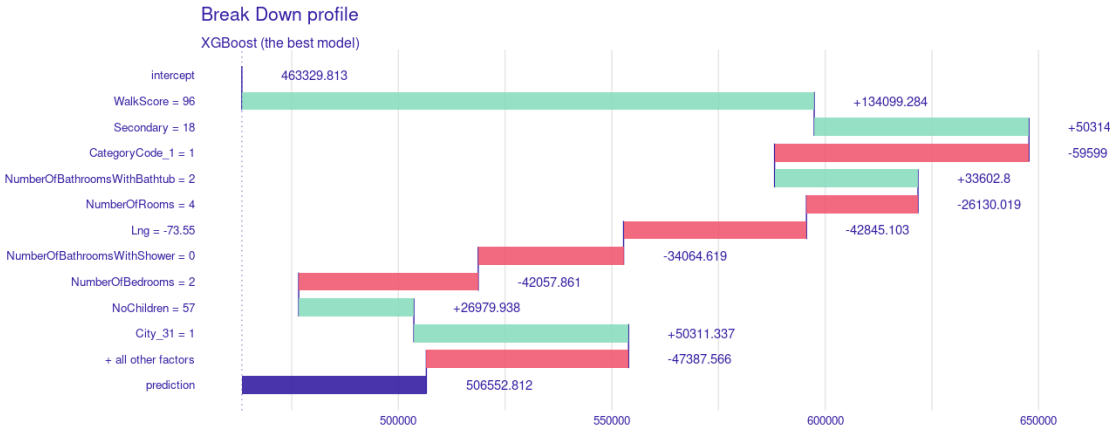


Figure 7: Second observation's profile

In this example, the most significant variables are *WalkScore* and *Secondary* - the metric describing the presence of secondary schools in the near location.

We can see that in the first case, *CategoryCode1* increases the prediction while it decreases the outcome in the second observation. Also, we can see that the *WalkScore* metric influences both results differently. When it is slightly lower - 75 - it decreases the prediction drastically. On the other hand, it hugely elevates the prediction in the other example where it is considerably higher - 96 out of 100.

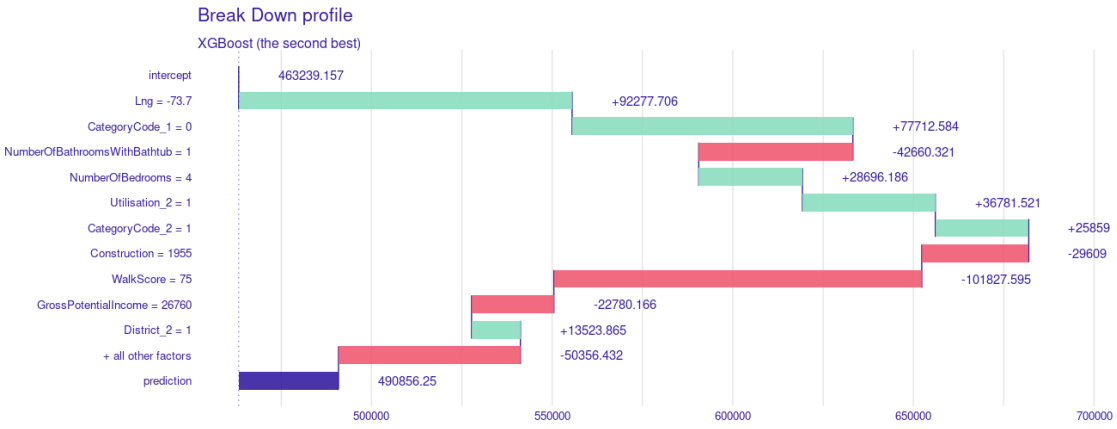


Figure 8: First observation's profile

The most important features for this observation are *Longitude* coordinates and *WalkScore*.

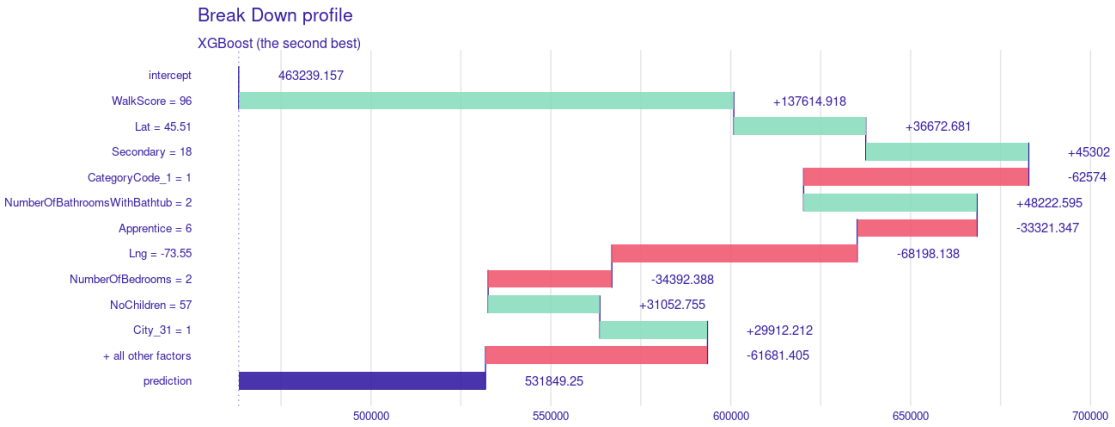


Figure 9: Second observation's profile

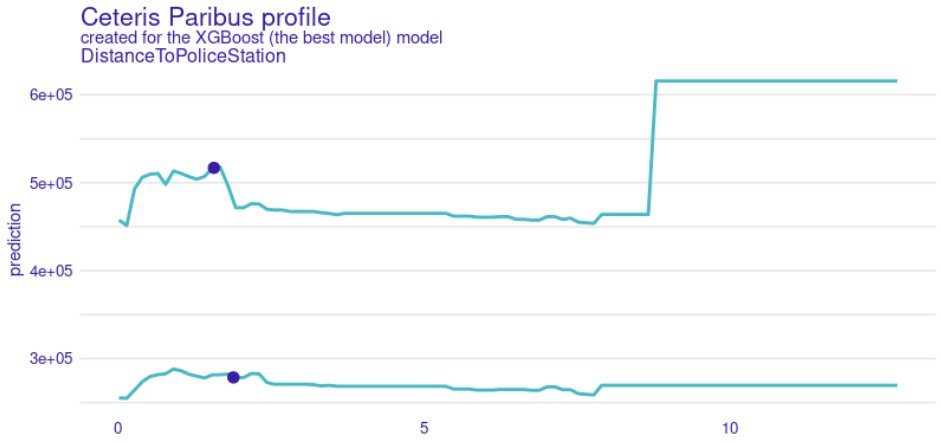
In this example, the most significant variables are *WalkScore* and *Longitute*.

We can see that in the second case, small change in *Longitute* can shift the prediction considerably. In the first observation, the longitude of -73.7 increases the prediction while the -73.55 value in second observation shrinks the value of a prediction noticeably. Also, we can see that the *WalkScore* metric influences both results differently. If its slightly lower - 75 - it decreases the prediction drastically. On the other hand, it hugely elevates the prediction in the other example when it is considerably higher - 96 out of 100 - just as in the first example.

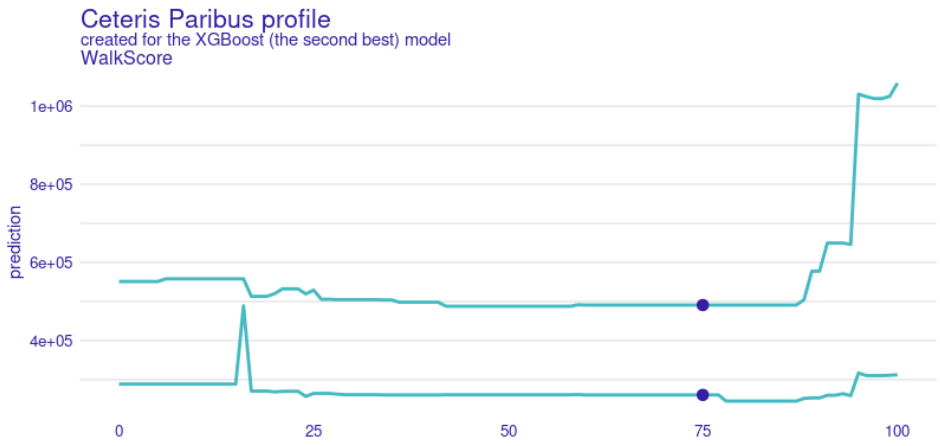
4.6 Ceteris Paribus profile

Ceteris Paribus decomposition profiles are designed to show model responses around a single point in the feature space. Essentially, we take the observation and analyze how the prediction is changing provided that other features' values in this observation stay untouched.

The first figure shows the analysis of the first model based on aforementioned two observations. In this part, we will examine the influence of *DistanceToPoliceStation* variable on the model's outcome.



The second CP plot concerns the behaviour of the second model based on two observations. Here, we will inspect the influence of *WalkScore* feature on the model's outcome.



We can see that the first observation behaves differently from the second one in both models. It reacts more abruptly with the increase of the variable's value. The second one does not show any significant reaction.

4.7 Conclusions from XAI

Clearly the most important features are *GrossPotentialIncome*, *NumberOfBathroomsWithBathtub* and *Walkscore* which makes sense, because these conclusions are in line with our intuition. In the city as big as Montreal, the ability to access facilities on foot is pretty convenient. Also, the number of bathrooms corresponds to more comfort and greater area of property. Variables related to location (i.e. *Lat* and *Lng*) also play a significant role in predictions, however they are not as important as the ones mentioned before. Interestingly, even though *CategoryCode_1* is not among the most important features on the permutational feature importance plot, it plays a big role in shaping the individual predictions which we have shown above.

5 Summary

Exploratory Data Analysis gave us initial insight into the data. If we compare these insights with the intuition gained from the XAI methods we can see that they are alike. We learned from the XAI methods that the most important features for forecasting real estate prices were the GPI index, the number of bathrooms with bathtubs and the Walkscore closely following them. This is consistent for the GPI and the number of baths with the results read from the correlation matrix, however, if we chose not to explain our models and based our understanding only on the correlation matrix, we would miss an important factor such as the Walkscore.

When it comes to training and selecting models, after training 20 different models (4 types with 5 versions with different parameters each), we have noticed that the XGBoost models inarguably outperformed the other models. Therefore we have chosen only XGBoost models for further analysis with the XAI methods. We believe that the two best of them, which had the RMSE lower than 1000$ CAD on the test dataset after PCA, are perfectly suited for predicting real estate prices in Montreal.

References

Patrick H. Hendershott Richard Green. Age, housing demand, and real house prices. 1999. URL `https://www.sciencedirect.com/science/article/pii/016604629602128X`.

Liu (Dave) Liu Nissan Pow, Emil Janulewicz. Applied machine learning project 4 prediction of real estate property prices in montréal. 2014. URL `https://www.readkong.com/page/applied-machine-learning-project-4-prediction-of-real-4501082`.

Dag Einar Sommervoll Axel Viktor Heymana. House prices and relative location. 2019. URL `https://www.sciencedirect.com/science/article/pii/S0264275118312241`.

Hy Dang Bo Mei Quang Truong, Minh Nguyen. Housing Price Prediction via Improved Machine Learning Techniques. 2020. URL `https://www.sciencedirect.com/science/article/pii/S1877050920316318`.

Przemyslaw Biecek. DALEX: Explainers for Complex Predictive Models in R. *Journal of Machine Learning Research*, 19(84):1–5, 2018. URL https://jmlr.org/papers/v19/18-416.html.