Out-of-Sample House Price Prediction by Hedonic Price Models and Machine Learning Algorithms

In an illiquid market like the real estate market, market values are not readily available. Transactions are scarce and do not always reflect market value. As a consequence, appraisal values play an important role to inform agents in decision making, financial reporting and for property taxes. For example, appraisal values are used for property investment decisions and for providing mortgage loans. In a recent report De Nederlandsche Bank raises concerns about the quality and independency of appraisal values (Van der Molen and Nijskens, 2019). The authors show that one third of all appraisal values exactly match the transaction price, and in almost 60% the appraisal value is higher than the transaction price. Automated valuation models (AVMs) are less prone to potential client influence. However, in order to be accepted by a broad audience, AVMs need to be transparent, robust, explainable and they need to provide reliable predictions. In this research we address these issues. We compare traditional hedonic price models to more advanced machine learning algorithms and analyse the accuracy of out-of-sample predictions and variable importance. The research is based on almost all residential transaction prices in the Netherlands in 2017.

Authors: Jeroen Beimer & Marc Francke

Methodology

Literature overview

There is some literature comparing the accuracy of model values based on traditional hedonic price models and machine learning algorithms. Early studies from the nineties do not find evidence of better model performance for machine learning algorithms, see for example Lenk et al. (1997) and Worzala et al. (1995). Since then, computer power and the availability of ready-to-use machine learning software packages and large databases have increased enormously. Later studies find some proof for machine learning algorithms to perform better than hedonic price models, see for example Antipov and Pokryshevskaya (2012), Zurada et al. (2011) and Kok et al. (2017). Although it is fair to say that most of these studies only use the most basic linear hedonic price models as a benchmark. They do not take into account non-linear relations and spatial and temporal dependencies of transaction prices, and the latter two are hard to deal with by machine learning algorithms. McCluskey et al. (2013) find that a spatial hedonic price model performs better than an artificial neural network, which is an example of a machine learning algorithm.

Added value to existing literature

This research adds to the existing literature by using a large dataset. Most studies use small datasets, in a specific city or neighbourhood. This research uses a large database for the Netherlands. Furthermore, both single family homes as well as apartments are included in the dataset. Earlier studies often focus on only one type. Finally, a comparative study of different house price models – including machine learning algorithms – is unique for the Netherlands.

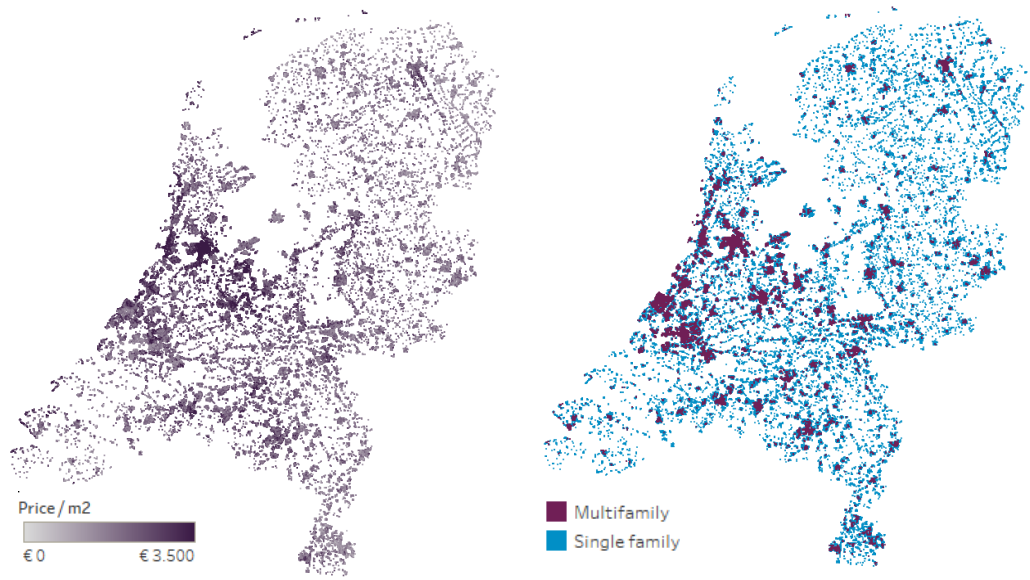
Data description

The dataset being used is retrieved from MVGM, covering around 95% of all transactions of residential real estate in the Netherlands in

2017. The dataset includes various housing and locational characteristics including geographical keys, which allow to match with other data sources. Housing characteristics include, amongst others, the number of (sleeping) rooms, living space and plot area in square meters, construction year, energy label, house type, the presence and type of garage and storage, maintenance level, both inside and outside, and the address including coordinates. Locational data describes the location in various dimensions such as city centre or neighbourhood, next to a busy road or adjacent to a park or forest. The dataset is enriched with additional variables on districts and neighbourhoods from Statistics Netherlands, ABF Research, Locatus and the Ministry of Internal Affairs. These include demographic and social factors, such as the composition of the population based on age cohorts, but also forecasts on demographic growth, the number of retail units within a specific area and the liveability score.

The total dataset contains 235,603 records, including non-residential transactions of land, mobile homes, caravans and houses for recreational purposes. Some records are missing geographical keys and construction year. These records have been removed. The transactions within our dataset have the date of legal transfer in 2017. In our analysis we use the date when the sale contract was signed as transaction date. On average the difference between these two dates is 2 to 3 months. We only consider transactions where both the date of signing the sale contract and the legal transfer date is in 2017. Finally, exceptionally large (small) or expensive (cheap) houses are being excluded from the dataset. We end up with 163,699 transactions. Figure 1 provides maps of transaction prices per square meter and the distribution of single family homes and apartments.

FIGURE 1 ▸ TRANSACTION PRICE PER M^2 (left) AND DISTRIBUTION OF SEGMENTS (right)



Source: MVGM

The pre-processing part includes the transformation of string variables (house type, location, energy label, maintenance level and urbanity) to dummy variables. We created quarterly time dummy variables per Corop-region[1] to accommodate regional house price changes. The final step is normalizing the dependent variables by means of a min-max scaler such that the normalized variables have a value between 0 and 1. Some machine learning algorithms require this kind of normalization.

Test and training set

Model calibration is done by the open-source software Python, which contains several useful libraries for linear regression and machine learning algorithms. The dataset is split into a training set, used for model training, and a test set for evaluating out-of-sample prediction. The same split is applied to all models. It is common practice to use 75% of the data for model training. However, due to the large dataset and hardware performance, 50% of the data is used for training and the other 50% for testing. In order to optimise the model, it is tested within the training set by cross-validation; the training set is repeatedly split and multiple models are trained. A 5-fold cross-validation is applied, so the training set is partitioned into 5 parts of equal size. Cross-validation reduces potentially model over- and underfitting. A drawback of cross-validation is an increase in computation time.

Variable and observation subsets

Machine learning algorithms are perceived as 'black boxes' and that variable (feature) importance is opaque. In order to get a sense for the impact of different variables, the modelling is executed on three overlapping subsets of variables. Subset A contains the smallest number of variables, covering only some property characteristics, such as the total surface, plot area and number of rooms. The second subset, B, adds property variables such as construction year, dwelling type, energy label and maintenance level. The final subset, C, adds both micro and macro locational variables. Comparing the results of the different variable subsets provides some insight in the importance of the different variables. For example, comparing the model performance between variable subsets C and B gives some insight in the importance of the locational variables.

Methods

We use seven different methods to calibrate house prices which can be divided into three different groups. The first group contains linear models: the hedonic price models, Ridge and Lasso regression and support vector machines (SVM). The second group consists of tree-based models: decision tree and random forest. The third group comprises the neural network model, a deep learning model.

Linear models

The basic hedonic price model is a linear model with the transformation of the house price as dependent variable, and housing, locational and time characteristics as independent variables. In this study we use a basic and simple hedonic price model that can be estimated by ordinary least squares. Ridge regression is very similar to the linear model apart from the fact that it uses regularisation in order to avoid multicollinearity and overfitting. Regularisation shrinks the coefficients of the explanatory variables and can be viewed as a restricted linear model. Lasso regression is almost similar to Ridge but allows for zero coefficients and therefore may exclude independent variables. Both Ridge and Lasso are powerful in small datasets. The SVM is more complex than the aforementioned ones. Next to regularisation, SVMs are more flexible as they allow for interactions between the independent variables. Additionally, it is possible to have non-linear relationships between variables. In this study we use a linear kernel, because it works better than the non-linear one and it requires less computation time.

Trees

Decision tree regression models differ from linear models. Decision trees are based on so-called

if-then-else questions. A drawback of decision trees is that they tend to overfit, although this can be minimalised by limiting the tree depth. The generalization performance is relatively weak due to this probability of overfitting. On the other hand, decision trees are easy to understand and they can handle different types of data. An improved version is a random forest model, an ensemble of decision trees. An ensemble combines different machine learning algorithms to create even more powerful ones. Random forests are a collection of decision trees, whereby trees differ slightly from each other. By averaging the results of all decision trees, it is possible to reduce overfitting. This is called bootstrap aggregating or bagging. Random forests are very flexible and can handle many different types of data. A disadvantage is that computation time is substantially more compared to linear models and decision trees.

Neural networks

Neural networks exist for quite some time, but these models have become popular over the last years due to the increase of computation power. A neural network system is an artificial intelligence model that replicates the human brain's learning process. This process is relatively unknown, but learning occurs through repetition. A neural network trains itself with historical combinations of data input and output. There is a variety of neural network models and for this analysis the multilayer perceptrons (MLP) is used, or the feed-forward neural network. This model has some linkages with a logistic regression. Each independent variable determines the total output based on their weights, or coefficients. The output is simply the weighted sum of all these inputs. In a MLP, this process is repeated multiple times, only with an intermediate step. This intermediate step, a hidden layer, combines all these weighted input to arrive at the final result.

Evaluation criteria

For the model evaluation we use four evaluation criteria: the R-squared (R^2), root mean squared error (RMSE), mean absolute error (MAE) and the mean absolute percentage error (MAPE). The selection of these criteria is based on their applicability for all models, and interpretability. They are commonly used in similar studies.

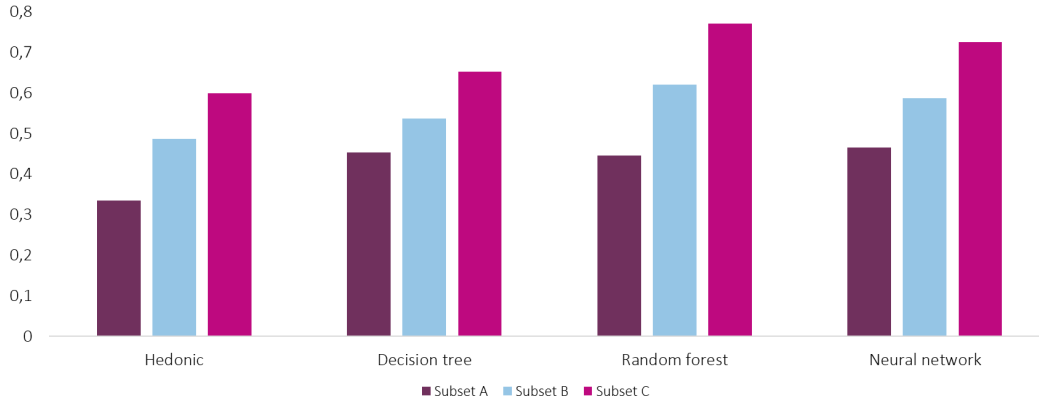
Results

The performances of Ridge and Lasso are almost similar to the hedonic price model based on all four evaluation criteria. The same holds for the SVM with linear kernel. For that reason we only discuss the results from the hedonic price model in the remainder of this paper.

The tendency amongst the four evaluation criteria is similar, whereby each individual criterion leads to the same conclusion. There are no contrasting results from the evaluation criteria and for that reason the focus of the results is on the R-squared since it is intuitive and easy to visualise.

Figure 2 shows the R-squared per model for the different subsets of independent variables (A, B, C). The R-squared of the hedonic price model is 0.33, 0.49, and 0.60 for subset A, B, and C, respectively. The same pattern is visible for the other algorithms. The inclusion of additional property characteristics like maintenance, energy label, building year and property type and in particular locational factors are important features to predict house prices. The R-squared of the random forest increases from 0.44 to 0.77 and that of the neural network from 0.47 to 0.73. Remarkably, the addition of more explanatory variables has a larger positive effect in the random forest than for the neural network. This might be due to the relative basic tuning of the neural network.

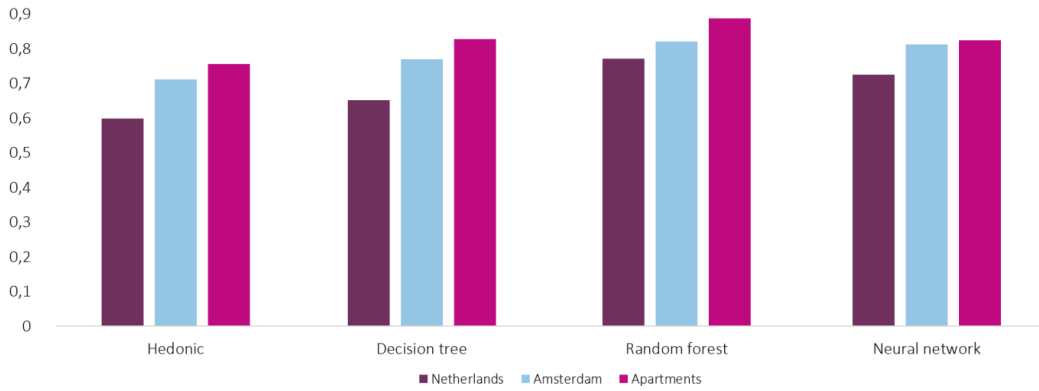
FIGURE 2 ▸ **R-SQUARED RESULTS FOR THE THREE DIFFERENT RESTRICTED VARIABLE INCLUSION MODELS**



Note: dataset includes all Dutch transactions (observation subset I)

Figure 3 shows the R-squared for the various subsets of observations, the Netherlands, Amsterdam and apartments, using all property and locational variables. Within all models, the R-squared is highest for apartments, and lowest for the Netherlands. And the random forest is uniformly performing better than the other algorithms. Differences between the algorithms are relatively small for apartments compared to the Netherlands: The R-squared ranges from 0.76 for the hedonic price model to 0.89 for the random forest. The range in R-squared for the Netherlands is 0.60 for the hedonic price model to 0.77 for the random forest. The former implies that the apartment sector is more homogeneous, but also that relationships between the independent variables are more linear for apartments.

FIGURE 3 ▸ **R-SQUARED RESULTS FOR THE DIFFERENT DATA SUBSETS**



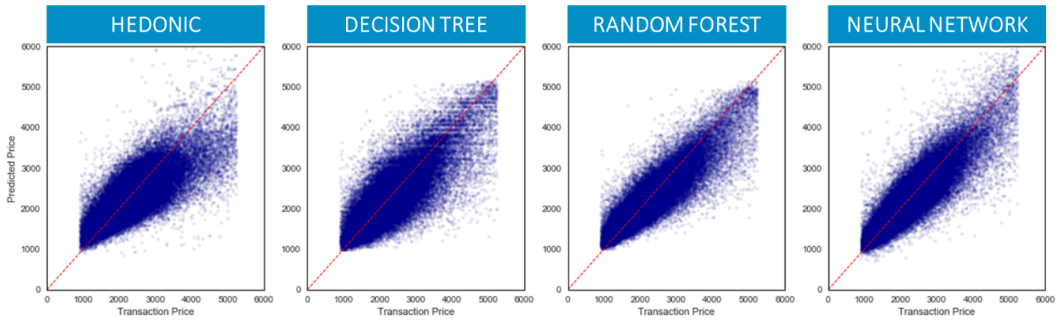
Note: dataset includes all variables, no restrictions (subset C)

Figure 4 displays scatter plots of transaction prices versus out-of-sample model predictions in EUR per square meter for the different methods, using all transactions and the largest set of independent variables (C). Ideally all dots should be on or as close to the red dotted lined, indicating that predictions are (nearly) similar to the price. For all methods there is a clear positive relation between price and prediction, but there are some differences.

Both the hedonic price model and the decision tree have larger errors since the scattered cloud is more curved than for the random forest and neural network. The latter two show a cloud which is flat and oblong. Additionally, the cloud of the hedonic price model looks non-linear, whereby dwellings with prices above EUR 3,500 per sq.m are undervalued. This is not the case for the other algorithms. Apparently, there are some relations that cannot be captured well in the basic linear hedonic price model.

Zooming in into the random forest and neural network shows undervaluation since more dots lie below the red line than above, in specific for the random forest. The latter one has more deviations in the higher price segment. Compared to random forest, the neural network has a relatively large amount of overvaluation in the higher segment. Most likely, this is the reason why random forest performs overall better than neural network.

FIGURE 4 ▸ VALUE PREDICTIONS VERSUS THE ACTUAL TRANSACTION PRICE (EUR per sq.m)

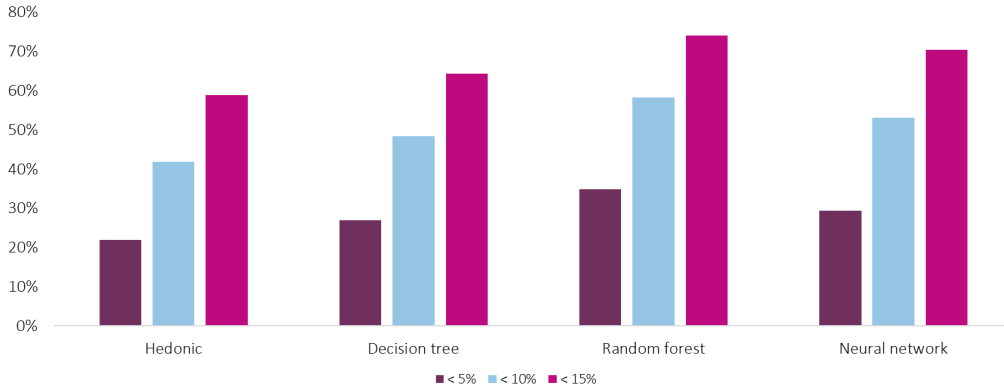


Note: dataset includes all variables, no restrictions (subset C) and all Dutch transactions (observation subset I)

Figure 5 displays the accuracy of the out-of-sample predictions for the different algorithms based on different bandwidths, using all transactions and the largest set of independent variables (C). In 22% (60%) of the cases the absolute relative price difference between prediction and price is less than 5% (15%) for the hedonic price model. Similar to previous results, the random forest gives the best performance with almost 75% of the cases within 15% absolute relative difference. The neural network comes close to the random forest.

A 15% deviation might be acceptable for the valuation of a large portfolio of properties (for example underlying properties of a mortgage portfolio) – where individual deviations cancel out – but is quite substantial for individual valuations. The appropriate bandwidth depends on the valuation purpose.

FIGURE 5 ▸ ACCURACY OF PREDICTIONS BASED ON DIFFERENT BANDWIDTH SIZES

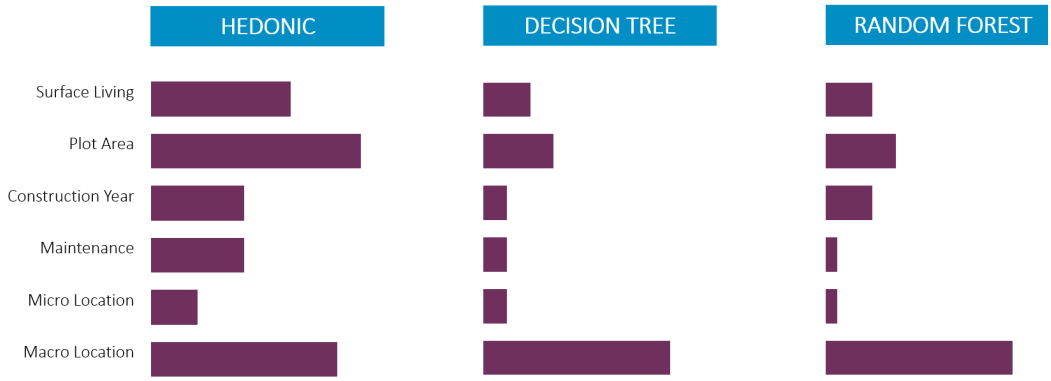


Note: dataset includes all variables, no restrictions (subset C) and all Dutch transactions (observation subset I)

Figure 6 shows the impact of variables on model values. Since there are multiple independent variables, some of them are combined. This aggregation is based on heuristics and denominated in absolute terms, enabling a reliable comparison between models. For example, the random forest displays feature importance in positive numbers only. For the neural network it is not possible to retrieve the impact of variables.

The results show that the impact of variables differ per model. In the hedonic price model, the variables surface of the living area and the plot area have a substantial impact. Other dwelling characteristics, such as construction year and maintenance level have less impact. This is in line with previous results, where the R-squared increases as the model is being extended. The impact of the macro locational factors is substantial. Not only, the coefficients of the Corop region dummy variables are important, but also the expected population growth per municipality.

FIGURE 6 ▸ IMPORTANCE OF VARIABLES



Note: dataset includes all variables, no restrictions (subset C) and all Dutch transactions (observation subset I)

The results for the decision tree and neural network are rather similar, which is not surprising since they are both tree-originated models. The results also are in line with the hedonic price model with the difference that the impact for the dwelling characteristics is less. Again, macro locational variables are dominant in the determination of house prices.

Conclusion

The goal of this study is to compare out-of-sample house price predictions based on hedonic price models and machine learning algorithms. In our application on house prices in the Netherlands in 2017 machine learning algorithms have larger explanatory power and provide lower errors. It is fair to note that the applied hedonic price model is relatively simple and only takes into account linear relationships. Moreover, spatial heterogeneity and dependence are not taken into account. The random forest provides the best overall performance, for all subsets of independent variables and observations. Random forest and other machine learning algorithms can easily deal with non-linear relations. For the hedonic price model, decision tree and random forest we have shown the importance of the different features, enlarging the transparency of the methods. For neural networks this is not (easily) possible, making it a more black-box method. All methods show that locational factors are the most important value drivers. The analysis on subsets of observations – Amsterdam and apartments only – have higher explanatory power and lower errors. This implies that the model for the Netherlands could be improved by adding more granular locational features.

ABOUT THE AUTHORS

Jeroen Beimer MSc MBA works as a senior real estate investment strategist in the Research & Strategic Advisory department at Bouwinvest Real Estate Investors. With this research, Beimer completed his MBA Big Data & Business Analytics course at the Amsterdam Business School in 2018. **Prof. dr. Marc Francke** is professor of real estate valuation at the University of Amsterdam and head of real estate research at Ortec Finance. Francke has supervised Beimer's MBA research.

FOOTNOTE

1 Corop is an administrative region in the Netherlands compared to NUTS 3 level in the European Union. There are 40 Corop-regions and its geographic layer is between province and municipality level.

REFERENCES

- Antipov, E.A., Pokryshevskaya, E.B. (2012). Mass Appraisal of Residential Apartments: An Application of Random Forest for Valuation and a CART-based Approach for Model Diagnostics. *Expert Systems with Applications 39(2)*, pp. 1772-1778.
- Kok, N., Koponen, E-L., Martinez-Barbosa, C.A. (2017). Big Data in Real Estate? From Manual Appraisal to Automated Valuation. *The Journal of Portfolio Management, Special Real Estate Issue* pp. 202-211.
- Lenk, M.M., Worzala, E.M., Silva, A. (1997). High-tech valuation: should artificial neural networks bypass the human valuer? *Journal of Property Valuation and Investment 15(1)* pp. 8-26.
- McCluskey, W.J., McCord, M., Davis, P.T., Haran, M., McIlhatton, D. (2013). Prediction Accuracy in Mass Appraisal: A Comparison of Modern Approaches, *Journal of Property Research 30(4)* pp. 239- 265.
- Van der Molen, R., Nijskens, R. (2019), De kwaliteit en onafhankelijkheid van woningtaxaties, Occasional Studies, Volume 17-1, De Nederlandsche Bank.
- Worzala, E., Lenk, M., Silva, A. (1995). An Exploration of Neural Networks and its Application to Real Estate Valuation. *Journal of Real Estate Research 10(2)* pp. 185-201.
- Zurada, J., Levitan, A.S., Guan, J. (2011). A Comparison of Regression and Artificial Intelligence Methods in a Mass Appraisal Context. *Journal of Real Estate Research 33(3)*, pp. 349-387.