STOCKHOLM SCHOOL OF ECONOMICS
Department of Economics
5350 Masters thesis in economics
Academic year 2018-2019

An Ensemble Machine Learning Approach to Estimating Swiss Home Prices

Andreas Thürlimann (41215)

Abstract

The price of a home is usually estimated by professional appraisers or by automated home valuation models, also known a hedonic pricing models. In the past, these models were largely based on linear regression models, however there is a trend towards using machine learning algorithms to estimate home prices. This thesis sheds light on automated home valuation models in Switzerland by developing a hedonic pricing model using machine learning algorithms and comparing it to the FPRE semiparametric regression model that is being used commercially to estimate home prices in Switzerland. The author found that a hedonic pricing model developed using a meta-learner trained on the predictions of several machine learning models was more accurate at estimating condominium prices in Switzerland than the FPRE model.

Keywords: Hedonic Pricing Model, Machine Learning, Real Estate, Stacked Generalization
JEL: C55, R31

Supervisor: Abhijeet Singh
Date Submitted: 10.12.2018
Date Examined: 08.01.2019
Discussant: Jennifer Olsen
Examiner: Maria Perrotta Berlin

Acknowledgements

I would hereby like to first thank my thesis advisor at the University of St. Gallen Prof. Dr. Roland Füss for supervising me in this thesis and arranging for me to work with data at Fahrländer Partner Raumentwicklung. I would also like to thank Dr. Stefan Fahrländer for providing the necessary data for this thesis and supporting me throughout the thesis. Moreover, I am grateful to Fahrländer Partner Raumentwicklung for offering me a place to work on my thesis and to its many employees for providing me with valuable feedback.

Contents

1 Introduction **1**

2 Background **3**

2.1 The Traditional Real Estate Appraisal 3

2.1.1 Importance of Real Estate Appraisals 3

2.1.2 The Three Types of Real Estate Appraisals 3

2.1.3 Real Estate Appraisal Bias 4

2.2 Linear Hedonic Pricing Method 5

2.3 Machine Learning Methods 6

3 Data **8**

3.1 Description of the FPRE Data 8

3.2 Feature Engineering 10

3.2.1 Sale Price 10

3.2.2 Location Levels 11

3.2.3 Geographic Coordinates 12

4 Model **13**

4.1 Benchmark Model: Semiparametric Model from Fahrländer (2006) 13

4.2 Model Development 14

4.2.1 Model Setup 14

4.2.2 Data Splitting 15

4.2.3 Hyperparameter Tuning 17

4.2.4 Performance Evaluation . . . 17

4.3 Base Models . . . 19

4.3.1 Linear regression . . . 19

4.3.2 Random Forest . . . 19

4.3.3 Gradient Boosted Machines . . . 21

4.3.4 Neural Networks . . . 22

4.4 Stacked Generalization . . . 25

5 Results **27**

5.1 Model Performance . . . 27

5.2 Discussion . . . 28

5.3 Limitations and Areas for Improvements . . . 29

6 Conclusion **31**

Bibliography **32**

Appendices **35**

A Variable Descriptions . . . 35

A.1 Ratings . . . 35

A.2 Condominium Type . . . 35

A.3 Second Home List Variable . . . 36

B Number of Observations by ObjOrt, Municipality, Kanton and Region . . . 36

List of Tables

1 Summary statistics (FPRE data) 9

2 Random Forest Hyperparameter Space 21

3 Gradient Boosting Hyperparameter Space 22

4 Neural Network Hyperparameter Space 25

5 Elastic Net Hyperparameter Space 26

6 Model Performance Evaluated on the Test Set 27

7 Description of the Ratings 35

8 Description of the Condominium Type 35

9 Description of the Second Home List 36

10 Number of Observations by Canton 37

List of Figures

1 Number of Observations by Objort 9

2 Distribution of Sale Prices 10

3 Regression Tree Partitions using only Longitude and Latitude Values 12

4 Model setup 15

5 Expanding window 16

6 Example of a Regression Tree 20

7 Example of a Feed-Forward Neural Network with a Single Hidden Layer 23

8 Distribution of Ratings 29

9 MAPE by Rating 29

10 Absolute Changes in the FPRE Home Price Index 30

11 MAPE by Year and Quarter Sold 30

1 Introduction

In Switzerland, real estate appraisals are largely carried out by licensed appraisers who have undergone an extensive program and have passed a federal exam. These appraisers analyze information on a home's attributes, its neighbourhood and current market conditions to produce an estimate of the home's value. Gathering and interpreting this information can oftentimes be a lengthy process and can lead to costly appraisal fees. A less expensive alternative is to use automated hedonic pricing models that employ statistical methods to produce an instantaneous estimate of a home's value based on the home's attributes. In the past, hedonic pricing models were largely based on linear regression models that in some form also try to model spatial dependencies among home prices. For example, (Fahrländer, 2006) developed a semiparametric regression model that is currently being used commercially to estimate the value of condominiums and single-family homes in Switzerland. In recent years, machine learning methods have gained a lot of hype and are being used in a variety of different applications including real estate pricing. The most prominent example of using machine learning models to estimate home prices is the Zillow home valuation model, or *Zestimate*, which produces home value estimates for more than 110 million U.S. homes and has achieved a surprisingly low nationwide median absolute error rate of 4.3 percent (Zillow, 2018). To achieve such results, Zillow uses a meta-model that combines millions of statistical models including machine learning models (CBS, 2017). Researchers have also experimented with machine learning models to estimate home prices. Peterson & Flanagan (2009) were the first to show that neural networks produce significantly more accurate home price estimates than linear hedonic models. Antipov & Pokryshevskaya (2012), Pow et al. (2016) and Kok et al. (2017) have used other popular machine learning techniques to estimate home prices in several different countries. Unfortunately, they were all trained on different data sets, which makes it unclear which algorithm is most accurate for estimating home prices. Moreover, they neither employ stacked generalization to boost performance, nor do they compare model performance to hedonic pricing models that are being used in the industry today.

This thesis sheds more light on the practice of automated real estate valuation models by utilizing data from Fahrländer Partner Raumentwicklung (FPRE) to develop a model for estimating condominium prices for Switzerland. Moreover, the semiparametric regression model from Fahrländer (2006) is used as a benchmark to compare model performance. This dataset is based on the arm's length transactions of condominiums in Switzerland and includes variables like the sale price, location, home attributes such as size and number of bedrooms, as well as micro-location ratings. With this data, popular

machine learning models were trained; e.g linear regression, random forests, gradient boosted machines and neural networks. Furthermore, we implement stacked generalization, which trains a meta-model on the predictions of the other machine learning models in order to boost accuracy. Because of the time dimension in the data, hyperparameter tuning was conducted using the expanding window method, where model selection is based on one quarter ahead home price prediction errors.

This paper serves several purposes. Firstly, by training multiple machine learning models, one can compare the individual machine learning model performance and test whether stacked generalization significantly improves accuracy. Secondly, the model's performance can be compared to semiparametric model from (Fahrländer, 2006), which serves as a good proxy for real estate appraisal accuracy today. Thirdly, the model's performance can be explored to see where the limitations may lie.

The first part of the paper gives a brief overview of the various real estate appraisal techniques. The second part of the paper describes data cleaning and feature engineering processes used. The third part explains the model development in detail. The last part of the paper shows the results and draws conclusions.

2 Background

2.1 The Traditional Real Estate Appraisal

2.1.1 Importance of Real Estate Appraisals

Knowing the market value, meaning the most probable sale price of a home is important for many parties besides buyer and sellers. For instance, mortgage lenders have an interest in knowing the value of the home, because it serves as repayment in case the homeowner defaults. Tax agencies also want to know the value of the home to determine the amount of property tax that is owed. Even insurance holders want to know the value of a home in order to determine the premium for property insurance (Brueggeman & Fisher, 2010, p. 201). As it is difficult to establish an accurate market value of a home, parties often rely on professional real estate appraisers to determine a home's value. To produce an appraisal of a home's value they rely on three main appraisal methods.

2.1.2 The Three Types of Real Estate Appraisals

There are three main types of real estate used today; the sales comparison approach, the cost approach, and the income approach. The most popular real estate appraisal method is the sales comparison approach. The rational behind this method is that an investor would never pay more for a home than what other buyers have recently paid for a comparable home. When using the sales comparison approach the appraiser first collects data on nearby recently sold homes and selects the three to four most similar homes to the home being appraised. Than the appraiser makes modifications to the sales prices of the comparables by adjusting for differences (e.g. size, age, quality, number of bathrooms, location, etc.). When making these adjustments the appraiser sometimes has to rely on his/her own judgment to determine for instance the added value of having an additional bathroom[1]. In the end, the appraiser calculates the average adjusted sales price of the comparables and uses it as an estimate for the sale price. The sale comparison approach works particularly well when there are many similar recently sold properties in that neighborhood. (Brueggeman & Fisher, 2010, pp. 201-205)

The second approach, the cost approach, lies on the fundamental idea that an investor would not pay more for a home than what it would cost to buy the land, and build the home. Thus, the appraiser

[1]Nowadays, adjustments in value are largely guided by hedonic pricing models

performs two separate calculations. First, the appraiser estimates the value of the land using the sales comparison approach for example. Second, the appraiser estimates the overall costs it would need to build that home today and subtracts depreciation, physical deterioration, functional obsolescence or external obsolesce from its estimate. Because it is sometimes difficult to put an accurate value on depreciation, this method is better suited for more newly constructed buildings (Brueggeman & Fisher, 2010, p. 207).

The third approach, the income approach, is best suited to estimate the value of income producing properties like apartment buildings and commercial real estate. Like the discounted cash flow method, the values of the property is equal to the sum of discounted rents minus expenses. In other words, the appraisers makes assumptions about the future rental income and expenses, and discounts them by a certain discount factor (Brueggeman & Fisher, 2010, p. 208).

The method of choice largely depends on the property being appraised. In the appraisal industry, it is often common practice to use several appraisal methods to come to a final appraisal. For, example one could consider weighted average, where each method is given equal weight or more reliable methods are given larger weights.

2.1.3 Real Estate Appraisal Bias

As mentioned above, the real estate appraiser has different methods to choose from depending on the home being appraised as well as the personal preference of the appraiser. Unfortunately, a large part of the process to produce appraisals still must rely on the appraiser's judgment. His or her appraisals can be subject to human biases and errors.

One notable source of bias, stems from the anchoring bias. Clayton et al. (2001, p. 337) found evidence that appraisers valuing the same property in consecutive periods tend to anchor onto their previous appraised values, causing appraisals to be too high in falling markets and too low in rising markets. This systemic bias helps partially explain why we observe smoother returns in appraisal-based indices compared to transaction-based indices Clayton et al. (2001, p. 359). Moreover, Julian Diaz & J. Andrew (1997, p. 256) found that appraisers operating in unfamiliar geographic areas tend to anchor on previous appraisals of other experts.

Another source of appraisal bias can be influenced by incentives. For example, during the U.S. housing bubble 2003-2007 many residential mortgage originators, whose compensation is based on granting

loans, pressured real estate appraisers to inflate their home appraisals that were being used as collateral in order to increase chance of getting the loan approved. Such behavior is fully rational when one considers that at the time they could easily pass on the increased default risk by selling the mortgages to unknowing secondary mortgage market agents (Diaz III & Hansz, 2010, p. 3). This helps explain the heavily upward biased appraisals during that period and may have been an important factor causing the real estate bubble.

2.2 Linear Hedonic Pricing Method

The linear hedonic housing pricing method was the first approach towards using large amounts of data and statistical methods to produce an objective estimate (free of any biases or errors) of a home's value. It is fairly similar to the sales comparison approach, since it also involves estimating the price based on prices of previously sold homes. The fundamental idea behind this method is that the value of home is determined by a combination of home features that a consumer finds desirable. Such features can include structural variables (size, age, number of bathrooms), location variables (quality of schools, tax rate) and environmental variables (air quality, noise level, number of green space) (Jens & Henry, 2015, p. 5). The simplest way of identifying the value that each feature adds to the price is through a linear regression model, also referred in the literature as a linear hedonic model

$$y = X\beta + \epsilon \tag{1}$$

where y is a $n \times 1$ vector of home prices and X is a $n \times m$ matrix of home attributes. The main benefit of linear hedonic models is its interpretability, as the coefficients β measure the average changes in the price when the corresponding variable changes by one. This feature is especially useful when trying to explain home value estimates to clients. The disadvantage of linear models is that they tend to overfit, therefore performing very poorly when predicting home prices out-of-sample.

In the real pricing, location is considered to be the most important driver of the value of the home, following the real estate mantra: "location, location, location". In real estate data one observes that home prices largely depend on the home prices of its surrounding neighbours. A possible way of dealing with spatial dependencies in home prices is through the use of spatial models. The simplest

one is the spatial auto-regressive model (SAR).

$$y = \rho W y + X\beta + \epsilon \tag{2}$$

SAR introduces a $n \times 1$ vector of spatial auto-correlations ρ and a $n \times n$ spatial weighting matrix W. Furthermore, it might be the case that the covariates X in a region such as a neighbourhoods crime rate may also affect the home price y in that region. The spatial Durbin model (SDM) accounts for dependencies in the covariates by introducing the term $WX\theta$ to the SAR model:

$$y = \rho W y + X\beta + WX\theta + \epsilon \tag{3}$$

2.3 Machine Learning Methods

Machine learning models are highly flexible statistical models that use data, oftentimes high-dimensional data, to make predictions. These types of models are better at fitting non-linear relationships and interaction among features in the data. They oftentimes include "regularization" methods or tuning parameters to help them to control for overfitting, which leads to more accurate out-of-sample predictions (Gu et al., 2018).

In recent years, machine learning has gained a lot of attention and is used in a variety of applications. In finance, (Gu et al., 2018) found more accurate measures asset price risk premia by using machine learning algorithms to model complex relationships between expected asset returns and its risk factors. In health care, (Gorczyca et al., 2017) train several machine learning algorithms and combines them using stacked generalization to improve the mortality risk predictions to better allocate care for trauma patients. Moreover, in the field of economics machine learning algorithms are being used to collect new sources of economic data, such as in the case where satellite data is used to measure economic development in African countries (Jean et al., 2016).

Interestingly, machine learning methods are also being used to price real estate. As the real estate data is high-dimensional, relationships are often non-linear and there are millions of possible interactions between home features, machine learning algorithms are a great way of making use of data to form predictions on the sales prices of homes. Many academics have already touched upon this field. For example, Peterson & Flanagan (2009) use neural networks to price residential real estate in Wake County, USA and showed that neural networks produce significantly more accurate out-of-sample

estimates of sales prices than linear hedonic models. Antipov & Pokryshevskaya (2012) use linear regression, k-nearest neighbours, neural networks, random forests, and gradient boosted machine to price apartments in Saint-Petersburg, Russia. They found that random forests produce the best estimates. Pow et al. (2016) employ support vector regressions, k-nearest neighbours and random forests to predict real estate prices in Montreal. They found that an ensemble of the k-nearest neighbour and random forest model produces the most accurate estimates (Pow et al., 2016, p. 6). Kok et al. (2017) use random forest and gradient boosted machines to predict home prices in the U.S. with a focus on states like California, Florida and Texas. They found that gradient boosting works best.

3 Data

3.1 Description of the FPRE Data

The data used for building the real estate valuation model is provided by FPRE. The data is collected on a quarterly basis from important players in the Swiss residential real estate market. They include banks and insurance, brokerage, trust, and architecture firms, as well as developers and consulting firms. The data set is fairly large (182'969 observations), covers most of Switzerland and spans the years 2000 to 2017. In addition to the sale prices, the homes are also described by a handful of home specific features namely year and quarter sold, year of construction, living area, number of rooms, renovation level, condominium type and a vacation home indicator. The data set also includes an indicator specifying whether the vacation home is in a town on the second home list in a inexpensive, expensive or tourist hostspot area. This relates to the law that was passed in Switzerland in 2012, and implemented of the 1st of January 2016, which limits the construction of second homes in certain municipalities.

Unfortunately, due to privacy concerns[2], the homes in this data set are not geocoded (information is only provided on the name of the town and district in which the home is located) and little information is provided about certain home features. Instead the homes are given ratings on their micro-location (value of the location), condition and standard. To produce these ratings, the data providers follow certain guidelines and largely base their ratings on observable factors in order to produce uniform ratings across all providers and minimize the subjective influence of the home assessors. These factors are then combined to give a rating between 1 (worst) and 5 (best) in increments of 0.5. The micro-location ratings are based on a handful of observable factors such as sunlight exposure, view, the image of the district, distance to services and leisure activities, access to public transport, road links and noise levels[3]. Condition is determined by a rather subjective view of the condition of the facade and the construction and whether the property was recently renovated. The standard rating is based on the observable factors such as room height, number of bathrooms, energy standard, heating unit and heat distribution as well as subjective factors such as room division, wall coverings and quality of its kitchen appliances. As previously mentioned, the data only provides the information on the town and district in which the home is located. FPRE uses this information to split the the towns and districts

[2]Swiss banks are not allowed to provide any information that could hurt the anonymity of its clients, as this would be a violation of the bank secrecy laws.

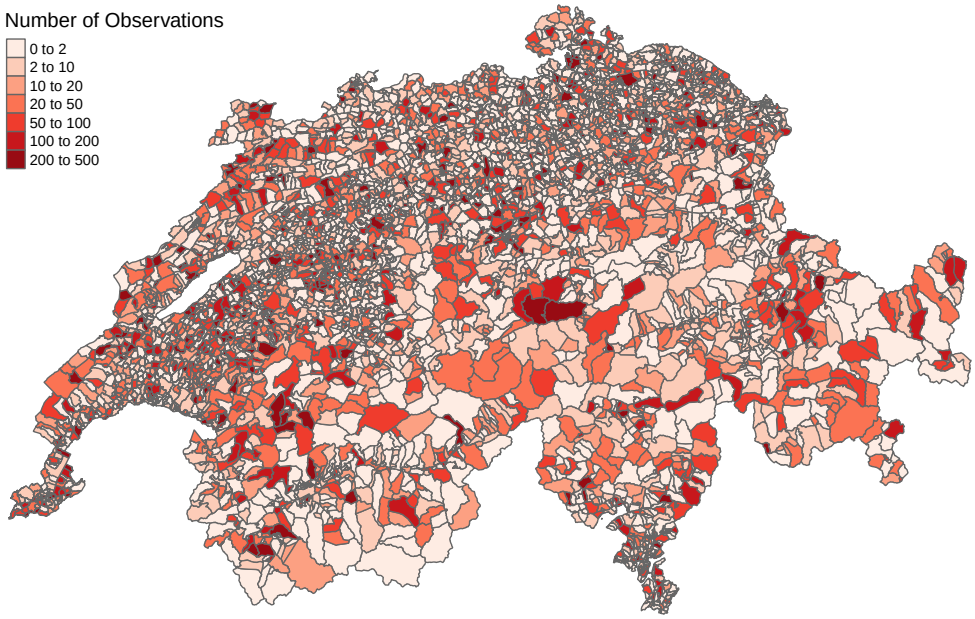
[3]For a more detailed description of the rating methodology please refer to: https://www.fpre.ch/en/products/micro-rating/ & https://tools.fpre.ch/Data/Erfassungsformulare/Form_ewg_CH-EN.pdf

Table 1: Summary statistics (FPRE data)

	Min	Max	Median	Mean
Sale price (in CHF)	56'000	11'850'000	586'000	705'454
Year and quarter sold	2000 Q1	2017 Q4	2010 Q4	2010 Q1
Year of construction	1920	2017	2003	1995
Living area (in m^2)	20	350	110	112
Number of rooms	1.0	20.0	4.5	4.08
Condominium type (categorical)	1	14	3	3.14
Second home (categorical)	1	7	1	1.17
Renovation level	0	4	0	0.26
Micro-location	1.0	5.0	3.5	3.61
Condition	1.0	5.0	5.0	4.23
Standard	1.0	5.0	3.0	3.35

into 5'430 so called Objorts as opposed to only 2'222 municipalities. In other words, Objorts can be thought of as either boroughs, towns or districts in large cities. When looking at the data, one can see that many condominiums are sold in densely populated area and tourist regions, while in rural areas tend to have fewer condominium sales, since people tend to buy more single-family homes in those areas.

Figure 1: Number of Observations by Objort

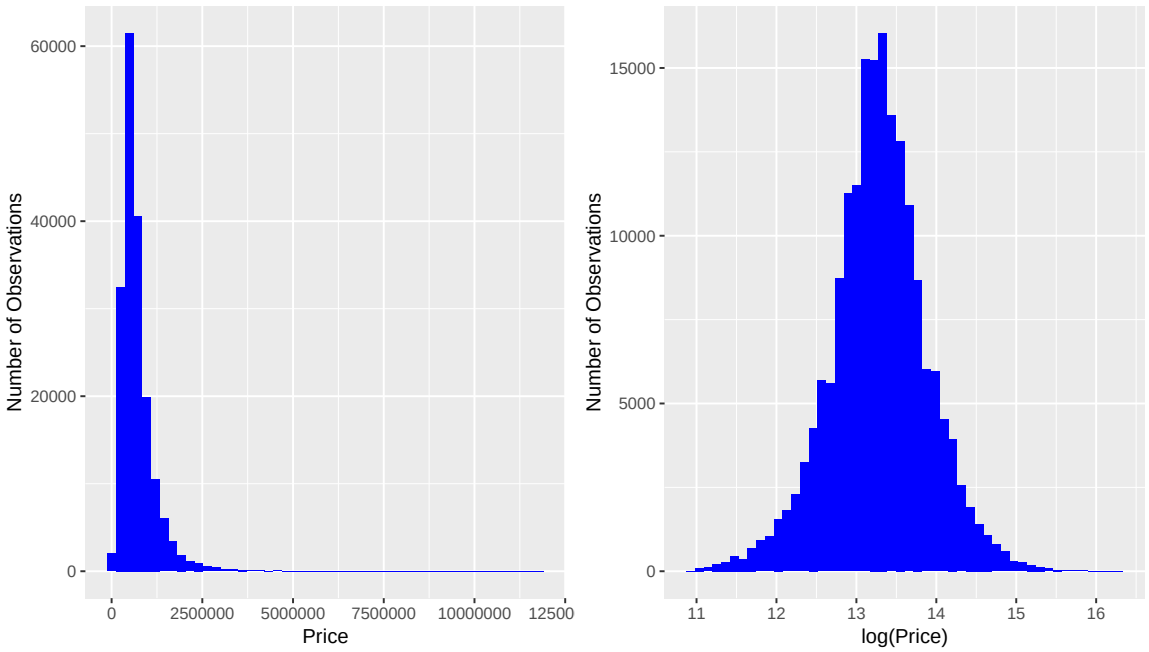


3.2 Feature Engineering

3.2.1 Sale Price

In the hedonic pricing literature, It is common practice to log transform the sale price (source). For linear models, it makes the distribution of the sale prices seem more normal and helps to satisfy the normally distributed residuals assumption for linear regression models. In the his paper, the most

Figure 2: Distribution of Sale Prices



important reason for log transforming the prices is due to how it affects the loss function of the machine learning algorithms. For all regression problems, the mean squares error is used as the loss function which one tries to minimize. However, in the case of real estate hedonic real estate models, we are rather interested in minimizing the squared percentage error rather than the squared error. By taking the log, we can approximate the mean squared percentage error loss function.

$$[\log \hat{y} - \log y]^2 \approx [\frac{\hat{y} - y}{y}]^2 \tag{4}$$

3.2.2 Location Levels

As mentioned in section 3.1, the number of observations per Objorts are not evenly distributed. This can pose several problems for machine learning models. In linear regression models, this will cause high standard errors in location dummy coefficients with few observations. Tree based models (random forest and gradient boosting) will not consider these location dummies for making splits, because it would cause the number of observations after the split to fall below the minimum number. Moreover, home prices for homes in Objorts that where not used during training will be improperly estimated.

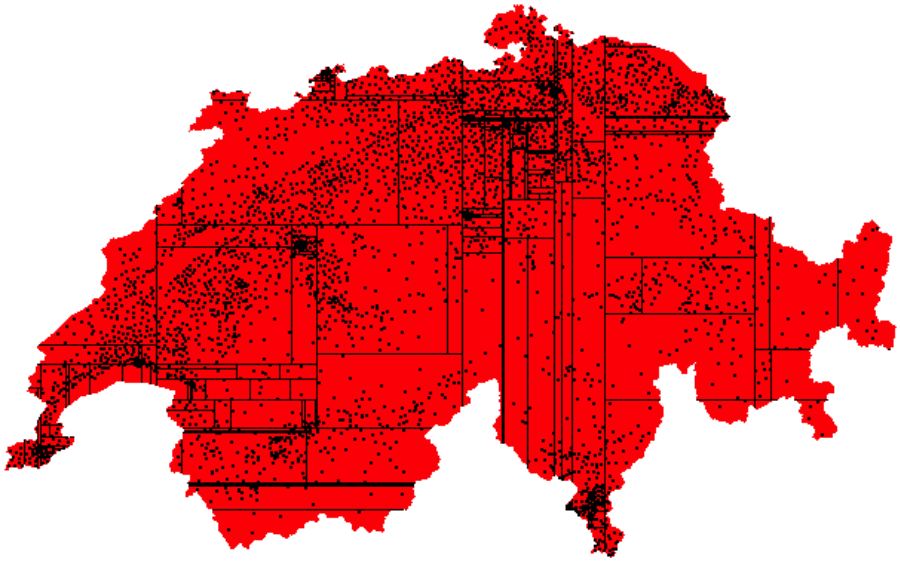
For this reason, higher location levels namely the municipality, MS-region and cantonal level as well FPRE regional level, which divides Switzerland into eight regions are also included. Furthermore, FPRE also assign the Objorts four location types (nucleated village, district in a city, municipality, small town within a municipality) as well as five region types (top location, location in a large and small metropolitan area, tourist spot, rural region).[4] Assigning such location level allows regression models to also estimate the dummy variables for higher location levels that can be used for predict home prices for homes in Objorts that have not been seen during training. Furthermore, tree based model to use these level as split points for Objorts with few observations in the training data. In other words, they can borrow observations from towns within the same region or from location with similar characteristics.

[4]The distribution of observations by municipality, canton, region and type can be found in the appendix

3.2.3 Geographic Coordinates

As home prices in Objorts also spatially depend on home prices in distant Objorts, it would have been useful to have access to the geographic coordinates of the homes. As a substitute, artificial coordinates of these Objorts were created by using the centroids of the Objort area to add geographic information to the algorithms. In fact, tree based models can use the longitude and latitude values to partition the regions themselves and use the sample mean in the regions as predictions. This is helpful for municipalities with few observations, as the algorithm can borrow observations from its neighbours to produce an estimate. Visually, this will look like the following:

Figure 3: Regression Tree Partitions using only Longitude and Latitude Values



4 Model

4.1 Benchmark Model: Semiparametric Model from Fahrländer (2006)

For a benchmark model, the semi-parametric regression model from (Fahrländer, 2006) was used, as it is being used commercially in the Swiss real estate industry today. It is used to help clients to appraise condominiums and single-family homes, and to estimate the rental incomes of rental properties. Therefore, this model serves as a good proxy to the industry standard of hedonic pricing models today. Over the years the model has evolved, however the fundamental idea has remained the same. For instance, the model is now only fitted using data collected in last six quarters, since the home feature coefficients can vary over time, and the fixed effects in equation (5) are estimated using historic data.

The model is built in three steps. In the first step Fahrländer (2006) fits a linear fixed effects model of the form

$$\begin{aligned} ln(price_i) = {} & \alpha_j \cdot Objort_i + \beta_{size} \cdot ln(size_i) + \beta_{yb} \cdot year_built_i + \beta_{yb2} \cdot year_built_i^2 \\ & + \beta_{cond}|_9 \cdot cond_i + \beta_{stand}|_9 \cdot stand_i + \beta_{micro}|_9 \cdot micro_i + \beta_{renov}|_9 \cdot renov \\ & + \beta_{second}|_7 \cdot second_i + \beta_{yq}|_6 \cdot yearqu_i + \beta_{sm}|_6 \cdot stand_i \cdot micro_i \cdot I(rating_i \geq 4) \\ & + \beta_{MSReg}|_{106} \cdot MSReg \cdot I(yearqu_i \in last_quarter) + \epsilon_i \end{aligned} \tag{5}$$

where the coefficients are estimated via robust regression with Huber's ψ-function in order to control for outliers. Size and year built represent numerical variables, while the all others represent categorical variables. The coefficient β_{sm} captures the interaction between standard and micro-location for all above 4. The coefficients β_{MSReg} is a categorical variable for all MS regions that have observations in their region, which helps capture the most recently estimated price level in the respective MS region. The fixed effects α_j estimate the price level of the Objort, however fixed effects with few observations tend to have high variance. In a second step, the estimated fixed effects α_j are used as a dependent variable in a second regression

$$\alpha_{j,t} = \beta_0 + s(dt1_j, dt2_j, dt3_j) + \beta_1 \cdot \hat{\alpha}_{j,t-1} + \epsilon_j \tag{6}$$

with an intercept, a regression spline, which is based on the first, second and third principal component that are obtained from of a $p \times p$ matrix of driving times between the p number unique of Objorts

in the data set and the fixed effects that were estimated in the previous quarter. These estimated coefficients are then used to re-estimate the fixed effects, which should produce better fixed effects estimates. In the third step, the regression model is re-estimated using the predicted fixed effects $\hat{\alpha}_i$ as inputs, where $X\beta$ represents the covariates in the 1. step.

$$ln(price_i) = \delta_j \hat{\alpha}_i + X\beta + \epsilon_i \tag{7}$$

4.2 Model Development

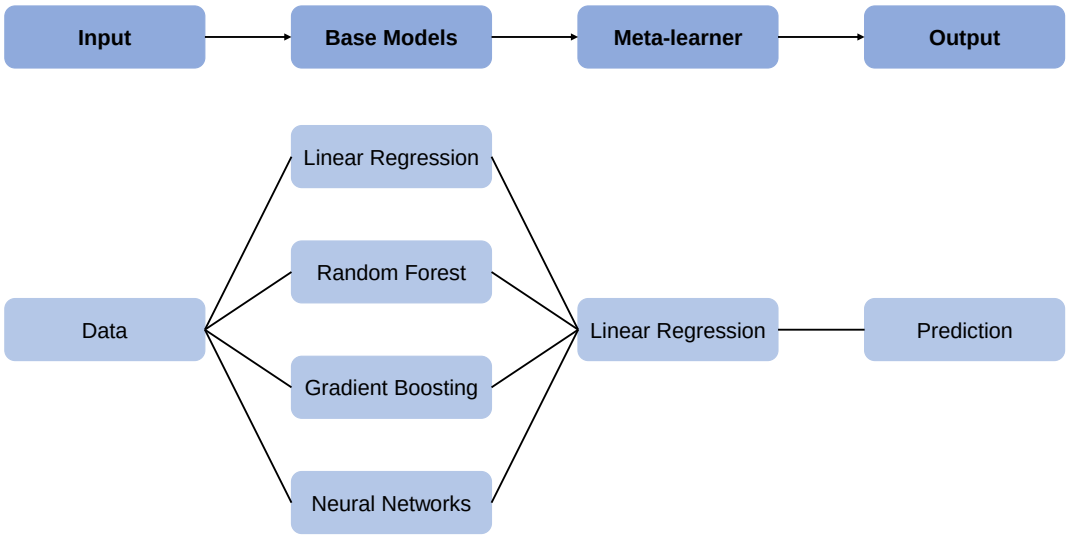
4.2.1 Model Setup

In this paper, data provided by FPRE is used to develop an automated valuation model to predict the sales prices of Swiss homes. This model starts off by training a handful of models, which will later be explained in more detail, namely a linear regression model, 50 random forest models, 200 gradient boosted machines and 20 neural networks using different hyperparameter values[5]. Moreover, stacked generalization is implemented to combine the models in order to boost accuracy. Like in real estate appraisals, the best appraisal is oftentimes a combination of appraisals. In stacked generalizations, one further step is taken by training a linear regression model on the predictions of the base model (Wolpert, 1992). Figure 5 shows a depiction of the model setup.

The estimated coefficients β_j in the meta-learner assign weights β_j to base model predictions $\hat{y}_{ij}$. The final prediction is simply the weighted sum of base model predictions $\hat{y}_i = \sum_{j=1}^{p} \beta_j \hat{y}_{ij}$. This diverse mix of base models helps to improve accuracy, as each base model is trained a little differently and learns some part of the problem so that by combining the models part of their prediction errors cancel each other out (Swamynathan, 2017, p. 244).

[5]The number of trained models is chosen somewhat arbitrarily, however they tend to reflect similar computing times

Figure 4: Model setup



4.2.2 Data Splitting

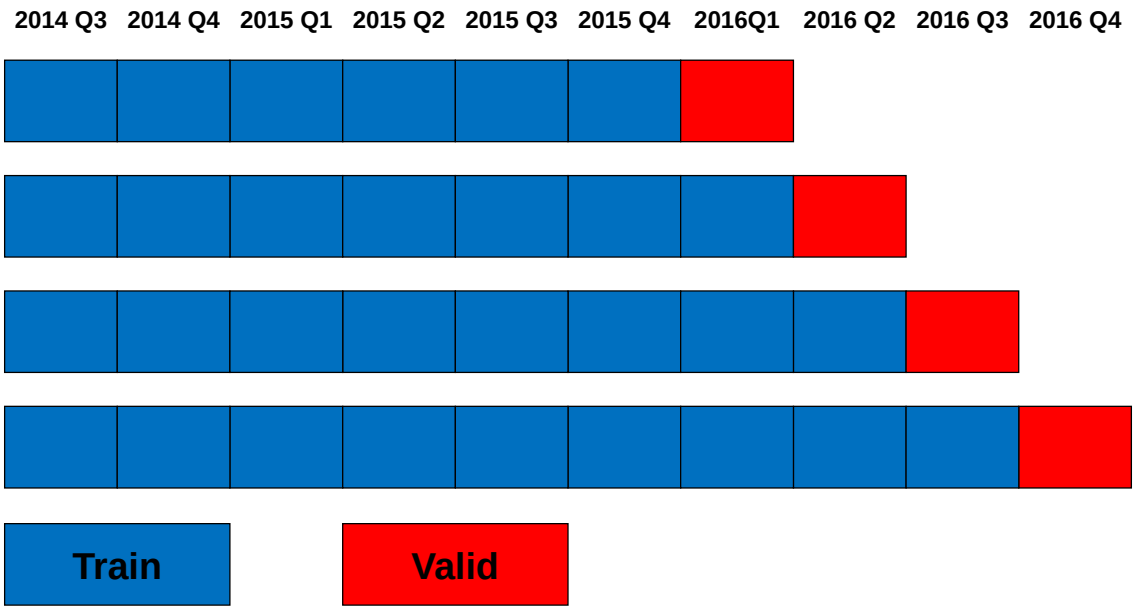
In most machine learning problems, the data is randomly split into three sets, independent from any features; a training set where the models are trained, a validation set where the prediction error is estimated for model selection and a test set where model performance is measured for the chosen model[6] (James et al., 2017, p.196). However, if data is sparse, one can also resort to k-fold cross-validation where one splits the training and validation set into k equal-sized parts and uses the k th part as a validation set and the other parts as a training set. This process is repeated until each part has been used as a validation set and sum of all k sets form the final validation set.

In the case of real estate valuation, one cannot simply divide the data into random parts, because it would be unrealistic to use data from the future to determine the price of a home today. Therefore, this paper uses the expanding window method[7], where one split our data based on year and quarter sold. We initially split the data into a training (2000 Q1 - 2014 Q4), validation (2015 Q1 - 2016 Q4) and test set (2017 Q1 - 2017 Q4).

[6]The test set was kept separate from the training and validation set and was only brought in for model evaluation at the very end. In no way did the test set influence the hyperparameter or feature selection

[7]In the literature also referred to as forward chaining or nested cross-validation

Figure 5: Expanding window



During the training and validation phase, a model is trained on the training set and makes an out-of-sample prediction of home prices using home features of homes sold one quarter in the future. The prediction at $t+1$ is given by adding the home features at time $t+1$ to the model trained at time t using only information $\mathcal{F}_t$ up to time t.

$$\hat{y}_{i,t+1} = f_t(x_{i,t+1} \mid \mathcal{F}_t) \tag{8}$$

This ensures that no data leakage occurs, meaning that no future home prices are used to determine the price of a home today. Once predictions are made, the training set is expanded by one quarter[8] and the procedure is reiterated until the last quarter is reached in the validation set. The collection of out-of-sample predictions in the validation sets serve two distinct purposes: one is that the predictions are used to calculate the validation error in order to determine the base model's optimal hyperparameter values, the other is that these predicted home prices also serve as input for the meta-learner training, while the observed prices serve as output variables. By running an algorithm on the meta-learner set, the meta-learner model looks for ways to combine the base-model prediction in order to better predict the observed prices. For meta-leaner training, the expanding window method was not employed,

[8]In the time series literature, the training data is often shifted one period forward to ensure that the training sets are of equal size. In this paper, the author decided to expand rather than shift the training data sets, in order to feed the model as much information as possible

because time is no longer a factor and most importantly, the meta-learner training set would have otherwise been to small. Instead k-fold cross-validation is used to take full-advantage of the rather small meta-learner set.

During the testing phase, the base models are trained, one quarter ahead prediction are formed, and the tuned meta-learner is applied to form a final prediction. This procedure repeated until the last observation in the test set is reached. The prediction of the meta-learner model and the tuned base models are stored, the errors are averaged, results are presented and conclusions are drawn.

4.2.3 Hyperparameter Tuning

Most machine learning model contain set of hyperparameter values, which can be thought of as constant parameters that need to be chosen before training. Each different set of hyperparameters values affects how the machine learning algorithm will fit the data and consequently will affect the accuracy of the model. To find these optimal set of hyperparameter values, the author uses random search, which trains models using a random combination instead of manual grid search, which trains models using all possible combinations of hyper-parameters for two reason (Swamynathan, 2017, p. 246). First, Bergstra & Bengio (2012, p. 1) have shown that random search finds better hyperparameter values than manual grid search in the same time, because oftentimes certain hyperparameters have limited effect on model accuracy. Second, by randomly drawing hyperparameters from a large space instead of manually drawing them from a small space, one can train more diverse models, which should help improve accuracy in the meta-learner model.

4.2.4 Performance Evaluation

To evaluate model performance, there are a handful of metrics one can choose from. One of the most widely reported metrics for regression problems is the the mean squared error (MSE). For machine learning model, the MSE is the loss function of choice for most regression problems. As explained in section 3.1, when one log transform the sale prices one can approximately transform the loss function equal to the mean squared percentage error (MSPE). The optimal home valuation model should minimize the the mean squared percentage error rather the mean squared error for several reasons. One is more interested in an estimate with the smallest relative error, the other is that the model is sensitive to the composition of homes in the validation set. For model evaluation, the MSE or the

MSPE are difficult to interpret. The mean absolute percentage error offers a lot more interpretability, since it shows how many percent the estimate is off from the sale price on average. Additionally, the median absolute percentage error (MdAPE) was also reported, since the distribution of absolute errors is positively skewed and thus the MdAPE is less sensitive to outliers.

$$\begin{aligned} \text{MSE} &= \frac{1}{N}\sum_{i=1}^{n}(\log \hat{y}_i - \log y_i)^2 \\ \text{MAPE} &= \frac{1}{N}\sum_{i=1}^{n}\left|\frac{\hat{y}_i - y_i}{y_i}\right| \\ \text{MdAPE} &= \text{median}\left|\frac{\hat{y}_i - y_i}{y_i}\right| \end{aligned} \tag{9}$$

4.3 Base Models

4.3.1 Linear regression

Linear regression is one the most simple statistical model. Its main advantage is that the coefficients[9] are highly interpretable, which comes at a cost of assuming certain functional form. The functional form was chosen similar to the benchmark model, however the interaction between standard and micro-location, as well as the interaction between the MS region and the last quarter were not used, as they did not increase the performance in the validation set. In this paper, three linear regression models using different location dummies were fitted.

$$\begin{aligned} ln(price_i) = {} & \delta_j \cdot Objort_i + \beta_{size} \cdot ln(size_i) + \beta_{yb} \cdot year_built_i + \beta_{yb2} \cdot year_built_i^2 \\ & + \beta_{cond}|_9 \cdot cond_i + \beta_{stand}|_9 \cdot stand_i + \beta_{micro}|_9 \cdot micro_i + \beta_{renov}|_9 \cdot renov \\ & + \beta_{second}|_7 \cdot second_i + \beta_{yq}|_6 \cdot yearqu_i + \epsilon_i \end{aligned} \tag{10}$$

$$\begin{aligned} ln(price_i) = {} & \delta_k municipality_i + \beta_{size} \cdot ln(size_i) + \beta_{yb} \cdot year_built_i + \beta_{yb2} \cdot year_built_i^2 \\ & + \beta_{cond}|_9 \cdot cond_i + \beta_{stand}|_9 \cdot stand_i + \beta_{micro}|_9 \cdot micro_i + \beta_{renov}|_9 \cdot renov \\ & + \beta_{second}|_7 \cdot second_i + \beta_{yq}|_6 \cdot yearqu_i + \epsilon_i \end{aligned} \tag{11}$$

$$\begin{aligned} ln(price_i) = {} & \delta_f ms_region_i + \beta_{size} \cdot ln(size_i) + \beta_{yb} \cdot year_built_i + \beta_{yb2} \cdot year_built_i^2 \\ & + \beta_{cond}|_9 \cdot cond_i + \beta_{stand}|_9 \cdot stand_i + \beta_{micro}|_9 \cdot micro_i + \beta_{renov}|_9 \cdot renov \\ & + \beta_{second}|_7 \cdot second_i + \beta_{yq}|_6 \cdot yearqu_i + \epsilon_i \end{aligned} \tag{12}$$

The coefficients in equation (10) were used to estimate most home prices, while the coefficient in equation (11) and (12) are used to estimate home prices in Objorts and municipalities that have not been used during training.

4.3.2 Random Forest

The random forest algorithm was first published by Ho (1995) and can be used for both classification and regression problems. The algorithm starts off by training regression trees $\hat{f}^b(x_i)$ in order to predict the housing prices. At each branch, the regression tree algorithm chooses a feature and split point

[9]The coefficients are not reported, as otherwise someone could carry out their own home appraisal

that minimizes the loss function in each region R_m after the split occurs, where the prediction in each region is equal to the average value

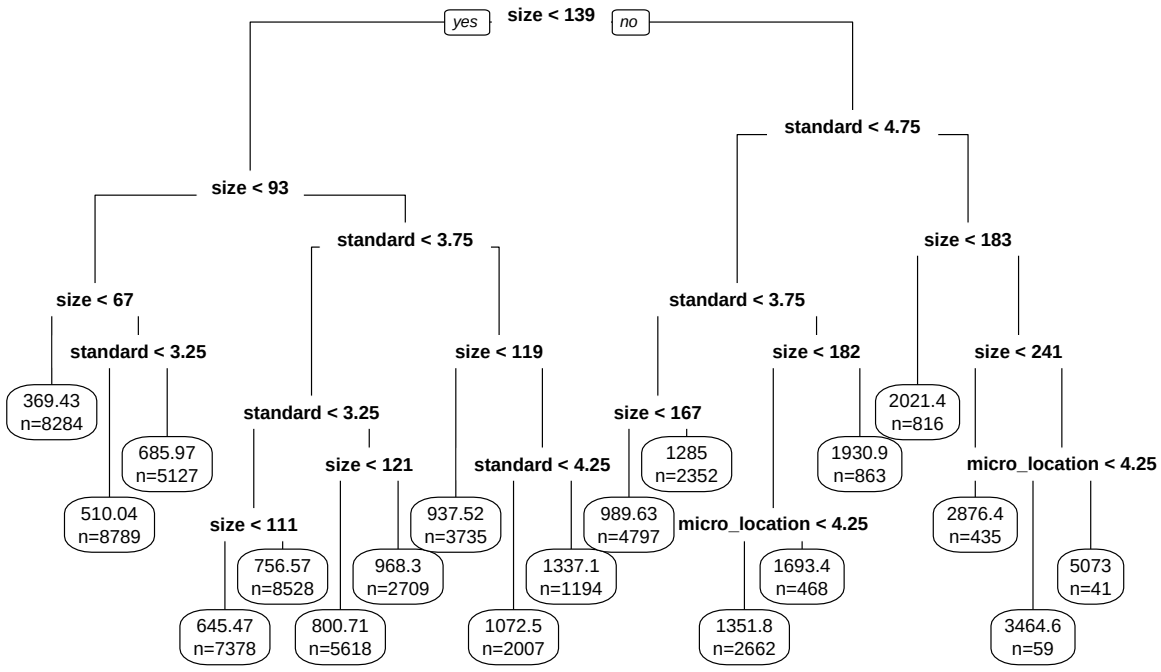
$$\begin{gathered} R_1(j,s) = \{X \mid X_j \leq s\} \text{ and } R_2(j,s) = \{X \mid X_j > s\} \\ \min_{j,s} [\min_{\hat{y}_{R_1}} \sum_{x_i \in R_1} (y_i - \hat{y}_{R_1}) + \min_{\hat{y}_{R_2}} \sum_{x_i \in R_2} (y_i - \hat{y}_{R_2})] \end{gathered} \tag{13}$$

where the prediction $\hat{y}_{R_m}$ in each region corresponds to the average value of y_i in the the region R_m (Hastie et al., 2009, p. 307).

$$\hat{y}_{R_m} = \frac{1}{I(x_i \in R_m)} \sum_{m=1}^{M} y_i I(x_i \in R_m) \tag{14}$$

The regression tree continues splitting the regions until it reaches a stopping criteria such as the maximum tree depth or a minimum number of observation in a leaf, which keep the tree from overfitting the data (Hastie et al., 2009, p. 307). Visually, a regression tree might also look like the following[10].

Figure 6: Example of a Regression Tree



Although individual regression trees have low bias they do suffer from high variance making them very imprecise on their own (James et al., 2017, p.316). In order to reduce variance, the random

[10]In the regression tree above I restricted the number of features considered to three to make it more visible

forest algorithm aggregates a collection of diverse regression trees. Diversity in the trees is introduced twofold: First, bootstrapped samples of the data are drawn before growing each tree and second, at each node only a random subset of features is considered before making a split. The final prediction is based on the average prediction of all regression trees.

$$\hat{y}_i = \frac{1}{B} \sum_{b=1}^{B} \hat{f}^b(x_i) \tag{15}$$

Aggregating trees cause the variance to converge to a certain limit (James et al., 2017, p.322). In this paper, the validation error stops to decreases when more than 150 trees are grown. Once the number of trees is fixed to 150, only three hyperparameters remain. The minimum number of observations in each leaf and the maximum tree depth control the size of the tree, while the fraction of features samples control the diversity of the trees.

Table 2: Random Forest Hyperparameter Space

Minimum number of obs. in a leaf	{1, 2, ... ,74, 75}
Maximum tree depth	{1, 2, ... , 49, 50}
Percent of variables used in each split	{30, 31, ..., 79, 80}

4.3.3 Gradient Boosted Machines

The Gradient Boosting algorithm is fairly similar to the random forest algorithm, as it also involves training regression trees. However, instead of training the regression tree independently from each other, the gradient boosting algorithm trains them sequentially (James et al., 2017, p. 321). In the literature, these regression are referred to as weak-learners[11], whose error rate is slightly better than chance (James et al., 2017, p. 299). The GBM algorithm is trained via gradient descent, which tries to minimize the loss function by gradually updating the models parameters until it reaches a minimum.

Another way of thinking about gradient boosting is the following: First, the algorithm first fits a small tree to to fit the prices. Second, it computes the residual $r_i = y_i - f(x_i)$ and fits another small tree $h(x_i)$ on the residuals. Finally, the fitted tree is multiplied by a predefined learning rate earning rate $\lambda \in [0, 1]$, which controls the weighting of new trees that are added to the model, and added to the

[11] Weak learners can also be linear model, however trees tend to work better for gradient boosted machines

initial tree.

$$\begin{aligned} \hat{y}_i &= f(x_i) \\ \hat{r}_i &= h(x_i) \\ f(x) &\leftarrow f(x) + \lambda h(x) \end{aligned} \tag{16}$$

This process is repeated until a specified number of trees are grown. The power of the GBM algorithm lies on the ability to improve upon its previous prediction, because observations where the algorithm performed poorly are given higher weights through larger residuals r_i.

In gradient boosting, oftentimes it is common to introduce an annealing learning rate $\alpha \in [0, 1]$, which gradually decreases the learning rate throughout training. This helps the GBM algorithm avoid missing out on a global minimum. Other hyperparameters include the random fraction of features considered at each split, similar to the random forest, and the random fraction of data sampled before growing a tree.

Table 3: Gradient Boosting Hyperparameter Space

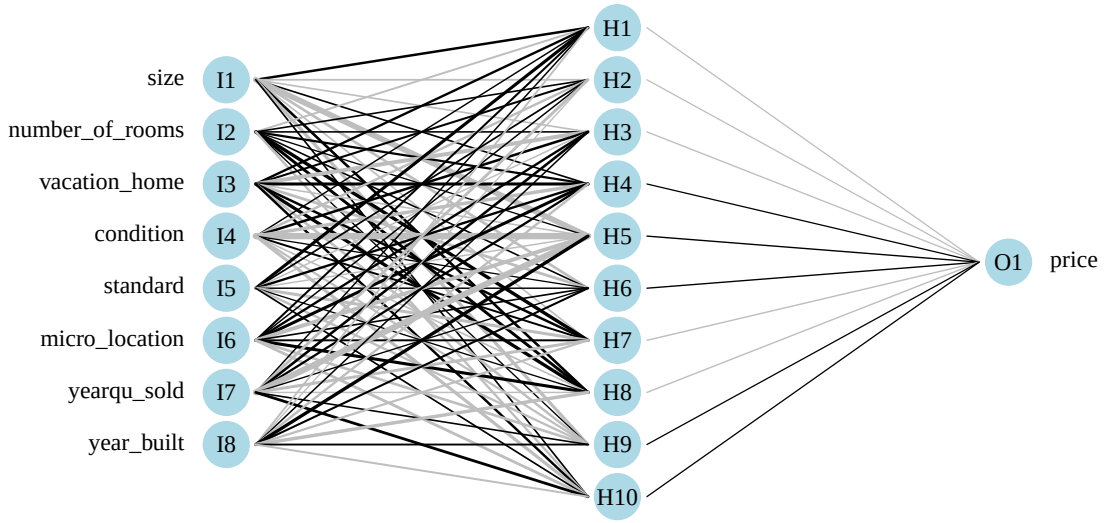
Number of trees	{10, 11, ... ,99, 100}
Minimum number of obs. in a leaf	{1, 2, ... ,74, 75}
Maximum tree depth	{1, 2, ... , 14, 15}
Percent of variables used in each split	{30, 31, ..., 79, 80}
Learning rate	{0, 0.001, ..., 0.499, 0.5}
Learning rate annealing	{0.85, 0.851, ..., 0.999, 1}
Sampling rate	{0.7, 0.701, ..., 0.899, 0.9}

4.3.4 Neural Networks

The term neural network describes a large class of algorithms that are loosely based on the biological brain. Neural networks are a great way of modelling non-linearities in the data and thus is the model of choice for tasks such as image and speech recognition. Furthermore, they are especially good at extracting linear combinations of the features.

In this paper, feed-forward neural networks were used.

Figure 7: Example of a Feed-Forward Neural Network with a Single Hidden Layer



A single layer neural network can be thought of simply a two-stage regression problem, where the derived features H_m in the hidden layer are derived from a weighted linear combination of inputs and the output[12] y is modelled through an activation function $\sigma(x)$ of a weighted linear combination of H_m. In this paper, rectifier linear unit activation function was chosen$ReLU(x) = max\{x, 0\}$, which allows the the neural networks to model non-linearities.

$$\begin{aligned} H_m &= \sigma(\alpha_{0m} + \alpha_m^T X), m = 1, ..., M \\ \hat{y} &= f(X) = \beta_0 + \beta^T H \end{aligned} \tag{17}$$

Similar to GBM, neural networks are trained by minimizing the loss function via stochastic gradient descent, where only a fraction of the data is used to compute the gradient. In the literature, this process is referred to as back propagation, which works in a forward and backward manner. The algorithm starts off by initializing the weights w $\{\alpha, \beta\}$ and biases b $\{\alpha_0, \beta_0\}$ by randomly selecting values close to zero. Then makes a prediction using the current weights, "back-propagates" the error and updates weights by taking steps equal to the negative gradient multiplied by a constant learning rate α.

$$\begin{aligned} w_{j+1} &:= w_j - \alpha \frac{\partial L(w_j)}{\partial w_j} \\ b_{j+1} &:= b_j - \alpha \frac{\partial L(b_j)}{\partial b_j} \end{aligned} \tag{18}$$

[12]Inputs and outputs are standardized before running the neural network algorithm

The learning rate α control the size of the updates. Specifying the learning rate can be quite tricky. On the one hand, a learning rate set too small can lead to slow convergence (high computation time). On the other hand, a learning rate too large can hinder convergence and cause the parameter values to oscillate back and forth around the minima. One way to tackle this problem is to add optional hyperparameters to the neural networks such as learning rate annealing and momentum training. Specifying the annealing learning rate will gradually reduce the learning rate throughout training in order to "freeze" the parameters into a local minimum, while momentum modifies back-propagation by allowing prior parameter iterations to influence the current update, which helps speed up the training (Candel & Parmar, 2015, pp.15 - 16). The inclusion of annealing learning rate and momentum can lead to more accurate models, however it requires tuning up to 7 additional hyperparameters (Candel & Parmar, 2015, p.15).

For this paper, the author employs the ADADELTA adaptive learning algorithm introduced by Zeiler (2012), which is well suited for sparse datasets. It combines the benefits of a learning rate annealing and momentum training with only two hyperparameters ϵ and ρ. The hyperparameter ϵ affects gradually decrease in the learning rate, while ρ relates how much weight is given to previous step sizes, similar to the momentum training.

In order to avoid overfitting, early stopping, as well as l_1 and dropout regularization techniques were used. Early stooping causes the neural networks to stop training once the mean absolutes error increases in the validation set. The l_1 regularization technique adds a penalty term to the loss function , penalizing the sum of absolute weights and biases multiplied by the l_1 hyperparameter λ_1. Like in the lasso, this causes many weights to shrink towards zero. The dropout rate, the probability that a neuron suppresses its activation during forward propagation at each training example is also used. The dropout rates vary for input layer and hidden layers and tend to be higher for the hidden layers. The number of hidden layers was chosen to be one , two or three and number of hidden units is constant for all hidden layers and varies between 10 and 1'000. We also implemented early stopping, which stops the neural network algorithm from updating the weights once the validation error starts to increase.

Table 4: Neural Network Hyperparameter Space

Number of hidden units	{10, 11, ... , 99, 100}
Number of hidden layers	{1, 2, 3}
Dropout ratio	{0, 0.01, ... , 0.32, 0.33}
$l1$ penalty	$\{10^{-10}, 10^{-9}, \ldots, 10^{-5}, 10^{-4}\}$
ρ	{0.9, 0.901, ..., 0.998, 0.999}
ϵ	$\{10^{-10}, 10^{-9}, \ldots, 10^{-5}, 10^{-4}\}$

4.4 Stacked Generalization

Stacked generalization also referred to as stacking, ensemble learning or blending is way of combining machine learning model predictions in order to improve prediction accuracy beyond the level obtained by any model alone (Günes et al., 2017, p.1). Stacked generalization is especially popular in data science competitions, where winning or loosing the competition often depends on a fraction of a percentage point in the error rate. One of the most famous competition is the one million dollar Netflix Prize, where a handful of teams competed to improve its movie recommendation algorithm. The wining team, *BellKor Pragmatic Chaos*, published its solution to winning the Netflix Prize (Bell et al., 2008). One of its key ingredient was in fact stacked generalization.

In the literature there are several stacking implementations. Stacking usually involves training a model on the predictions of other models. This idea can also be expanded to include multiple levels, where in each level a model is trained on the model predictions of the lower level. Furthermore, one can also only include various subsets of observation, features and models in each level (Günes et al., 2017, p.6). While such multi-level stacking methods can lead to incremental gains in model accuracy, they are computationally expensive. For this reason, the author follow the single level stacking approach similar to Gorczyca et al. (2017), where one trains several base models using different machine learning methods and hyperparameter values, and trains meta-learner model on top of their predictions.

For the meta-learner model the elastic net regression was chosen, since it helps choose the best predictors (best base models) for making a prediction. The elastic net is a shrinkage method similar to the ridge and lasso regression, which reduce variance at a cost of higher bias (Hastie et al., 2009, p. 73). It combines the ridge and lasso penalty terms, which penalize large coefficients. The vector of coefficients $\beta = (\beta_1, ..., \beta_n)^T$ is estimated by solving the following minimization problem

$$\hat{\beta} = \arg\min_{\beta} \Big\{ \sum_{i=1}^{n} (y_i - \beta_0 - \sum_{j=1}^{p} x_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{p} (\alpha\beta_j^2 + (1-\alpha) \mid \beta_j \mid) \Big\} \tag{19}$$

where the final prediction is

$$\hat{y} = X\hat{\beta} \tag{20}$$

The introduction of the penalty term λ causes the model to penalize large coefficients and shrinks them towards zero. The hyperparameter λ determines how much weight is given to the size of the coefficients, while the hyperparmeter α distributes the weight that is imposed on the ridge and lasso penalty term. As a results the elastic net eliminates unimportant features from the equation, since they are shrunk to zero. Hyperparameter tuning was done through 5-fold cross-validation and by randomly choosing hyperparameter values from the following hyperparameter space.

Table 5: Elastic Net Hyperparameter Space

Penalty term λ	$\{10^{-10}, 10^{-9}, \ldots, 10^{-2}, 10^{-1}\}$
Weight α	$\{0, 0.01, \ldots, 0.99, 1\}$

5 Results

5.1 Model Performance

As mentioned in chapter 4, one linear regression model, 200 gradient boosted machines, 50 random forests and 20 neural networks were trained. The validation set (2015 Q1 to 2016 Q4) was used to find the optimal hyperparameters of the the base models and meta-learner model. These tuned models were then used to report the one quarter out-of-sample performance for 14'915 condominium sales in the test period (2017 Q1 to 2017 Q4), which is presented in table 6. Although all models were trained using the same time-period, not all features were included in certain algorithms. In linear models, the longitude and latitude variable were not considered, because a non-spatial linear model would not be able to model such highly non-linear data and the higher location levels were only used in separate regression model. For neural networks, the higher level locations were not included, as it would have considerably increased computation time.

Table 6: Model Performance Evaluated on the Test Set

	MSE	MAPE	MdAPE	<1%	<5%	<10%	<15%
Meta-Learner	**3.17**	**13.49**	**9.68**	**5.33**	**27.03**	**51.40**	**69.15**
FPRE Model	3.54	14.18	10.52	5.08	25.60	48.07	65.33
Gradient Boosting	3.77	14.37	10.28	5.47	26.97	49.08	65.46
Random Forest	4.34	15.07	11.41	4.72	23.62	44.77	61.71
Linear Regression	4.15	15.74	11.77	4.58	22.7	43.43	60.84
Neural Networks	6.72	19.49	15.42	3.49	17.34	33.8	48.84

Table 6 shows that the gradient boosted model is the clear winner regarding all base models. Second in line is the random forest model, which has 0.67 percentage points lower MAPE than the standard linear regression model. Surprisingly, neural networks did not perform that well. This is most likely due to the Objort variable, a large categorical variable, that significantly increased computation, since it increased the number of weights by 5'430 times the number of hidden units in the first hidden layer. Therefore, one could not train enough neural networks models to find the optimal hyperparameters. To improve performance, one would either need more computational power or alter the neural network architecture.

The results suggest that tree based models such as the random forest and gradient boosted machines work really well at estimating home prices. They are able to handle the large categorical Objort variable, missing values and irrelevant features, and are robust to outliers (Hastie et al., 2009, p. 351).

Surprisingly, none of the base models are able to outperform the FPRE Model on their own, except the gradient boosted model in terms of MdAPE. However, when the models are combined using a meta-learner the picture changes. Now the meta-model is able to noticeably outperform the FPRE model in all performance metrics. On average the meta-learner models home price estimates are off by 13.49 percent. In other words, the meta-learner home price estimate is on average 4.9 percent closer to the true sales price than the FPRE model. The MdAPE, which shows that half of all home estimates are within 9.68 percent of the true sales price, is considerably lower than the MAPE due to the presence of outliers. In fact, 3.5 percent of home price estimates in the test set have an absolute percentage error greater than 50 percent. However, when analyzing these outliers, no noticeable patterns could be found, suggesting that these high errors are either caused by unreported or misreported home features.

5.2 Discussion

As previously mentioned, this paper shows that the meta-learner model outperforms the FPRE model in terms of model accuracy. However, this does not necessarily mean that the meta-learner model is a better suited hedonic pricing model than the FPRE model. The FPRE model has some advantages over the meta-model that need to be considered.

First, the FPRE model is a linear regression model, which means that the model coefficients are interpretive. Unfortunately, most machine learning models do not possess that property. Interpreting neural networks is almost impossible and understanding how tree based models produce their home estimates requires one to examine hundreds of trees. Second, the FPRE model is considerably more computationally efficient, as it can be trained in less than a minute, while the meta-learner models require training several machine learning models using different hyperparameter values to produce a home estimate, taking hours depending on the computer. A faster alternative may be to only select the best base models and combining them with a linear regression model. However, this would only lead to a MAPE of 14.06 percent.

While lower computation is not necessarily a huge advantage for real estate appraisal, since one only needs to train the model once every quarter, having an interpretative model has several benefits. For instance, an interpretive model allows one to explain to clients how exactly the model produces these home estimates. Moreover, it is easier to reach regulatory requirements. Therefore, one really needs to consider whether having a more accurate and less interpretive model is a better choice.

5.3 Limitations and Areas for Improvements

The meta-learner model is not equally accurate for all homes. Like in the sales comparison approach, the model performs well for homes with many comparable homes. Figure 9 and 10 perfectly highlight this problem, as they show a clear negative relationship between the number of homes and the MAPE in that rating class. Linear regression model, where the ratings were coded as dummies, saw highly variable coefficients for rating dummies with fewer observations. Tree based models, where the ratings were coded as numerical variables, can also borrow observations from their neighbouring rating classes to improve their predictions. However, this also causes them to overvalue and undervalue homes on the left and right tail of the micro-location and standard ratings distribution. Most likely, the model would have performed better if the data set also included the variables needed to produce these ratings.

Figure 8: Distribution of Ratings

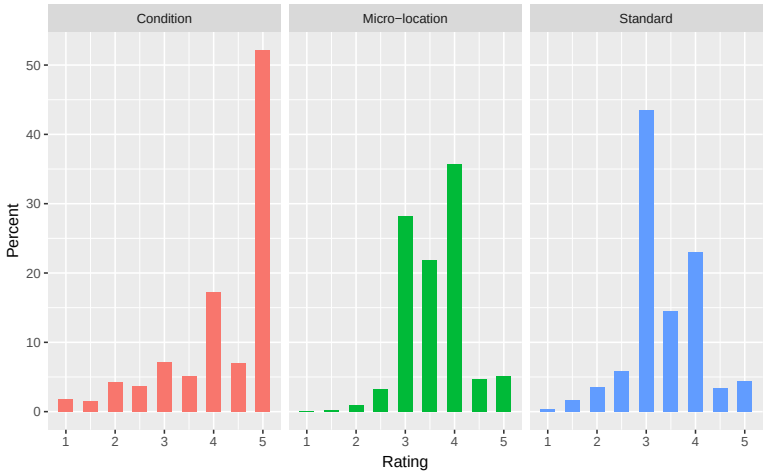
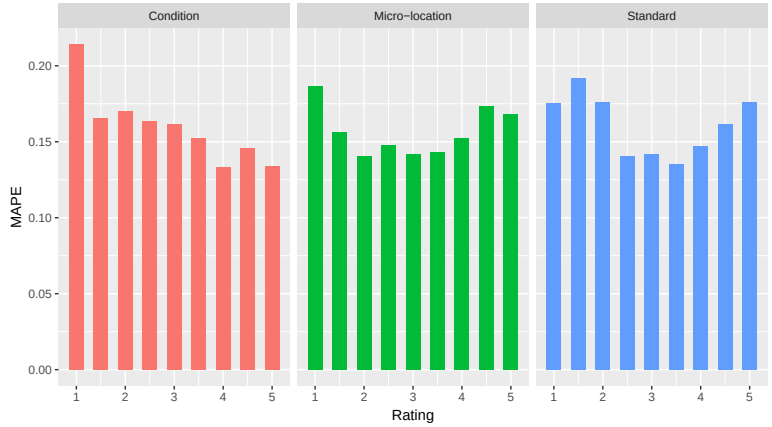


Figure 9: MAPE by Rating



Another area where the model seems to not perform that well is in predicting home prices in the future. The meta-learner model estimates home prices as if they had been sold in the previous quarter. In other words, the model is not able to capture any time trend, therefore large changes in the real estate will cause biased estimates. Figure 11 and 12 shows a positive relationship between the absolute percentage changes in the FPRE index and the MAPE. In fact the period with the highest MAPE also saw the largest percentage increase in the home price index. To capture this time trend, several

Figure 10: Absolute Changes in the FPRE Home Price Index

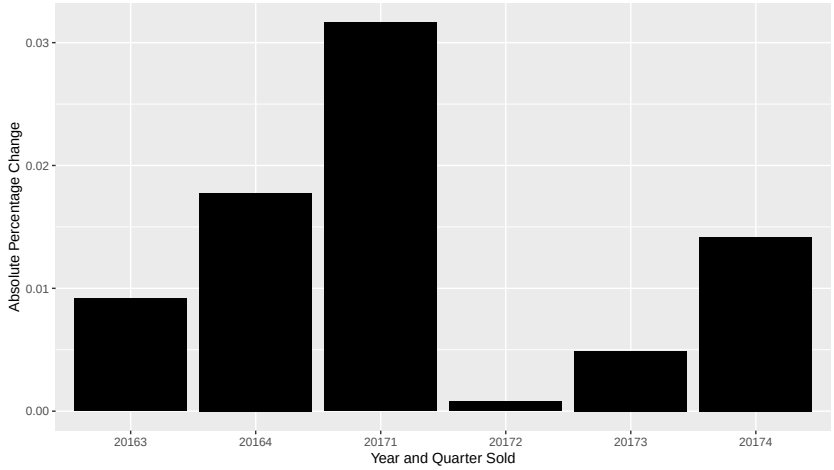
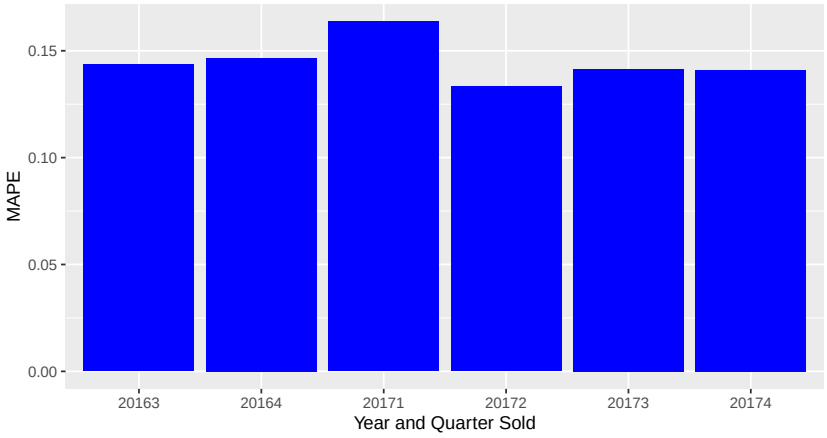


Figure 11: MAPE by Year and Quarter Sold



possible methods were utilized, however with little success. First, an auto-regressive model of order one was fitted on the FPRE home price index in order to forecast the quarterly percentage changes in the index value. These percentage changes were either included as features in the model or were used to adjust the predicted home prices. The first method did not improve the accuracy at all, while the second method actually reduced accuracy. Perhaps, one would have to somehow alter the machine learning model to also capture the dynamic setting or include better predictors as features.

6 Conclusion

This paper showed how different machine learning models compared to one another in predicting home prices. It is shown that only a hedonic pricing model developed using several machine learning models and combined through stacked generalization was more accurate at estimating condominium prices in Switzerland than the FPRE model. This highlights the importance of training and combining several machine learning algorithms to improve predictive performance, as otherwise one is not taking full advantage of the predictive power of the data. Unfortunately, a hedonic pricing model developed using machine learning algorithms is a black box, which is good for purely predictive tasks, however it is not able to describe step-by-step how the model produces these predictions. Because clients and regulators require some sort of interpretability and the difference in accuracy between the meta-learner model and FPRE model is quite small, the FPRE model remains a viable hedonic pricing model for Switzerland today.

This may however change in the future. As the paper shows, stacking several machine learning models considerably increased accuracy. Therefore, the author would also expect that training more base models and finding ways to decrease computation for the neural networks would lead to a gain in meta-model accuracy. However, in order to reach accuracy levels similar to the Zestimate, data providers would need to provide more information about the home. For instance, the author expects that by including the geographic coordinates in the data one would see an increase in model accuracy, because one could gather more valuable information on the locations and model spatial dependencies. Another way of improving models is to also tweak the machine learning models in order to capture the time trend in real estate prices.

Perhaps in the future, hedonic pricing models will improve to the point that they will be able to supplement or even replace traditional real estate appraisals. This will bring several benefits to the real estate market such as lower appraisal fees, more transparency through low cost instantaneous appraisals and a lower risk of future housing bubbles.

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*, 1772 - 1778. Retrieved from https://www.sciencedirect.com/science/article/pii/S0957417411011894

Bell, R. M., Koren, Y., & Volinsky, C. (2008). The bellkor 2008 solution to the netflix prize. Retrieved from https://www.netflixprize.com/assets/ProgressPrize2008$_{B}ellKor.pdf$

Bergstra, J., & Bengio, Y. (2012). Random search for hyper-parameter optimization. *Journal of Machine Learning Research*, *13*, 281–305. Retrieved from http://dl.acm.org/citation.cfm?id=2188385.2188395

Brueggeman, W. B., & Fisher, J. D. (2010). *Real estate finance and investments.* New York McGraw-Hill, 2010.

Candel, A., & Parmar, V. (2015). Deep learning with h2o.

CBS. (2017, May 24). *Zillow ceo on 1 million competition to improve "zestimates".* Retrieved 2018-12-04, from https://cbsn.ws/2G1a6z3

Clayton, J., Geltner, D., & Hamilton, S. W. (2001). Smoothing in commercial property valuations: Evidence from individual appraisals. *Real Estate Economics*, *29*(3), 337. Retrieved from https://onlinelibrary.wiley.com/doi/epdf/10.1111/1080-8620.00014

Diaz III, J., & Hansz, J. A. (2010). A taxonomic field investigation into induced bias in residential real estate appraisals. *International Journal of Strategic Property Management*, *14*(1), 3-17. Retrieved from https://www.tandfonline.com/doi/pdf/10.3846/ijspm.2010.02?needAccess=true

Fahrländer, S. (2006). Semiparametric construction of spatial generalized hedonic models for private properties. *Swiss Journal of Economics and Statistics*, *142*(4), 501-528. Retrieved from https://www.fpre.ch/wp-content/uploads/2006-IV-3.pdf

Gorczyca, M. T., Toscano, N. C., & Cheng, J. D. (2017). The trauma severity model: An ensemble learning approach to risk prediction. *bioRxiv*. Retrieved from https://www.biorxiv.org/content/early/2017/10/31/210575

Gu, S., Kelly, B. T., & Xiu, D. (2018). Empirical asset pricing via machine learning. *Swiss Finance Institute Research Paper No. 18-71*. Retrieved from https://ssrn.com/abstract=3281018

Günes, F., Russ, Wolfinger, & Tan, P.-Y. (2017). Stacked ensemble models for improved prediction accuracy. *SAS Institute Inc.* Retrieved from http://support.sas.com/resources/papers/proceedings17/SAS0437-2017.pdf

Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The elements of statistical learning. data mining, inference, and prediction.* New York Springer, 2009.

Ho, T. K. (1995). Random decision forests. In *Proceedings of 3rd international conference on document analysis and recognition* (Vol. 1, p. 278-282).

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2017). *An introduction to statistical learning with applications in r.* New York Springer, 2017.

Jean, N., Burke, M., Xie, M., Davis, W. M., Lobell, D. B., & Ermon, S. (2016). Combining satellite imagery and machine learning to predict poverty. *Science*, *353*(6301), 790-794. Retrieved from http://science.sciencemag.org/content/353/6301/790 doi: 10.1126/science.aaf7894

Jens, K., & Henry, W. (2015). Estimating the value of urban green space: A hedonic pricing analysis of the housing market in cologne, germany. *SFB 649 Discussion Papers*. Retrieved from http://sfb649.wiwi.hu-berlin.de/papers/pdf/SFB649DP2015-002.pdf

Julian Diaz, I., & J. Andrew, H. (1997). How valuers use the value opinions of others. *Journal of Property Valuation and Investment*(3), 256. Retrieved from https://www.emeraldinsight.com/doi/full/10.1108/14635789710184970

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 2017*, *43*(6), 202-211. Retrieved from https://doi.org/10.3905/jpm.2017.43.6.202

Peterson, S., & Flanagan, A. B. (2009). Neural network hedonic pricing models in mass real estate appraisal. *Journal of Real Estate Research*, *31*(2), 147 - 164. Retrieved from https://ideas.repec.org/a/jre/issued/v31n22009p147-164.html

Pow, N., Janulewicz, E., & Liu, D. (2016). Applied machine learning project 4: Prediction of real estate property prices in montral. Retrieved from https://goo.gl/amjgjC

Swamynathan, M. (2017). *Mastering machine learning with python in six steps : A practical implementation guide to predictive data analytics using python.* Apress.

Wolpert, D. H. (1992). Stacked generalization. *Neural Networks*, *5*, 241-259. Retrieved from http://www.machine-learning.martinsewell.com/ensembles/stacking/Wolpert1992.pdf

Zeiler, M. D. (2012). Adadelta: An adaptive learning rate method. Retrieved from https://arxiv.org/abs/1212.5701

Zillow. (2018). *What is a zestimate home value?* Retrieved 2018-12-04, from https://www.zillow.com/zestimate/

Appendix

A Variable Descriptions

A.1 Ratings

Table 7: Description of the Ratings

Rating	Micro-location	Condition	Standard
5.0	best location	as new	luxurious
4.5	very good location	good-as new	elevated-luxurious
4.0	average-good location	good	elevated
3.5	average location	intact-good	average-elevated
3.0	location with minor disadvantages	intact	average
2.5	location with disadvantages	requiring renovation-intact	out of date-average
2.0	location with disadvantages	requiring renovation	out of date
1.5	unfavourable location	strongly requiring renovation	modest-out of date
1.0	unsuitable location	urgently requiring renovation	modest

A.2 Condominium Type

Table 8: Description of the Condominium Type

Type	Description
1	Apartment on the ground floor
2	Apartment on the 1st floor
3	Apartment on the 2nd floor
4	Apartment on the 3rd floor
5	Apartment on the 4th floor
6	Apartment on the 5th floor
7	Apartment on the 6th+ floor
8	Apartment in the attic
9	Attic
10	Duplex apartment
11	Stepped apartments
12	Apartment in the basement
13	Special object
14	Unknown

A.3 Second Home List Variable

Table 9: Description of the Second Home List

Type	Description
1	Not a vacation home
2	Vacation home in a inexpensive tourist region on the second home list
3	Vacation home in a inexpensive tourist region not on the second home list
4	Vacation home in an expensive tourist region on the second home list
5	Vacation home in an expensive tourist region not on the second home list
6	Vacation home in a tourist hotspot region on the second home list
7	Vacation home in a tourist hotspot not on the second home list

B Number of Observations by ObjOrt, Municipality, Kanton and Region

Table 10: Number of Observations by Canton

Canton	Total	Percent
Zurich	28'982	15.84
Vaud	27'569	15.07
Bern	16'853	9.21
Valais	14'940	8.17
St. Gallen	13'658	7.46
Aargau	10'494	5.74
Geneva	9'928	5.43
Ticino	9'685	5.29
Graubünden	9'332	5.10
Lucerne	6'750	3.69
Basel Landschaft	4'881	2.67
Fribourg	4'868	2.66
Thurgau	4'530	2.48
Schwyz	3'203	1.75
Solothurn	2'989	1.63
Basel Stadt	2'901	1.59
Neuchâtel	2'677	1.46
Zug	2'663	1.46
Appenzell Ausserrhoden	1'433	0.78
Schaffhausen	1'360	0.74
Nidwalden	962	0.53
Obwalden	653	0.36
Jura	643	0.35
Uri	399	0.22
Glarus	397	0.22
Appenzell Innerrhoden	219	0.12
Switzerland	182'969	100.00