Graz Economics Papers – GEP

GEP 2020–02

Metrics for Measuring the Performance of Machine Learning Prediction Models: An Application to the Housing Market

Miriam Steurer and Robert J. Hill

February 2020

Department of Economics
Department of Public Economics
University of Graz

Metrics for Evaluating the Performance of Automated Valuation Models

Miriam Steurer and Robert J. Hill
Department of Economics, University of Graz,
Universitätsstrasse 15/F4, 8010 Graz, Austria:
miriam.steurer@uni-graz.at, robert.hill@uni-graz.at

January 15, 2020

Abstract:
Automated Valuation Models (AVMs) based on Machine Learning (ML) algorithms are widely used for the prediction of house prices. While there is consensus in the literature that cross-validation (CV) should be used for model selection in this context, the question of which performance metrics to use is generally neglected. Here we collect the most commonly used metrics from the AVM literature and elsewhere, and evaluate them with respect to two symmetry conditions: symmetry with respect to prediction error rates and symmetry with respect to the treatment of actual and predicted values. While none of the commonly used metrics satisfy both conditions, we propose a number of new metrics that do. We also show how popular existing metrics can be altered so that they adhere to these conditions. To illustrate our findings we compare the performance of 5 ML-based AVMs and find, that the most popular metrics in the AVM literature can generate misleading results. A different picture emerges when the full set of metrics is considered, and especially when we focus on four key metrics with the best symmetry properties.
(JEL. C45; C53)

Keywords: Machine learning; Performance metric; Automated valuation model (AVM); Appraisal; Prediction error; Model selection

We acknowledge financial support for this project from the Austrian Research Promotion Agency (FFG), grant #10991131, and thank ZTdatenforum (http://www.zt.co.at) for providing us with the dataset.

1 Introduction

While parametric models remain the gold standard when it comes to understanding the structure of the world around us, data-driven semi- or non-parametric models – collectively often referred to as Machine Learning (ML) models – generally outperform their parametric counterparts at short-term out of sample prediction.[1] Driven by the development of new methods, increased computing power, and the emergence of big data, the last two decades have brought about a huge growth in new ML methods. A distinction can be drawn between those ML methods that predict numerical values and those that classify observations into different groups. Our focus here is the former task - in particular we are interested in how researchers can judge the relative performance of various Automated Valuation Models (AVMs) of the real estate markets.[2]

In the real-estate context, the objective of an AVM is to predict the price of apartments or houses. The traditional benchmark for AVMs is the hedonic model, where price (or log price) is assumed to depend on available characteristics in an additive way. This model has a number of benefits: it can be easily estimated via least-squares regression, it is well grounded in economic theory, and its output has an intuitive interpretation (total price is dependent on the shadow prices of the individual characteristics). However, due to their superior performance with regard to price predictions, most AVMs use some type of ML technique instead of hedonic models (see e.g., Schultz, Wersing, and Werwatz, 2014). When it comes to ML methods, users can choose from a large and continually expanding list of different approaches such as Random Forests, Quantile Regression, LASSO Regression, Adaptive Regression Splines, and Neural Nets, to name but a few. These methods can be parametric, semi-parametric, and non-parametric in nature.

Model selection between parametric models is often centered around variations on the Akaike

[1] This point has been stressed both in the academic literature (see e.g. Varian, 2014) as well as in practical applications (e.g., the winning entries of Kaggle competitions (www.kaggle.com/competitions) almost invariably use ML methods).

[2] Sometimes the term “Mass Appraisal” is used instead of AVM in the literature.

Information Criterion (AIC) (Akaike, 1973), where to avoid overfitting the use of more parameters is penalized. On the other hand, model selection for ML models is generally done via cross-validation (CV) (Yang, 2007).[3] CV can also be used to compare parametric and non-parametric models. Indeed, Yang (2007) shows that under certain conditions CV is a consistent criterion in this regard (in the sense of selecting the better procedure with probability approaching 1 in large samples). But using CV for model selection does not answer the question of which performance metrics should be used at each stage in the validation process. We attempt to present an answer to this question here, and in the process rationalize the relevant literature.

We begin by proposing two symmetry conditions for performance metrics: symmetry with respect to the prediction error rates and symmetry in the treatment of actual and predicted values. We then survey a range of metrics that have been proposed in different strands of the literature and bring them together using a common notation to allow direct comparison. These metrics are then classified into four classes based on their performance relative to the symmetry conditions. We show that none of the metrics generally used satisfy both conditions. Next, we illustrate how to transform metrics so that they adhere to these notions of symmetry. In total, we consider 49 different performance metrics that could be used to evaluate model performance.

Next, we illustrate the performance of these metrics by applying them to five different parametric, semi-parametric and non-parametric models to predict house prices based on transaction data for the city of Graz in Austria for the 2014-2017 period. In particular, we train the following models: a (hedonic) linear regression model which serves as our parametric benchmark model, a Random Forest model, a model with Multivariate Adaptive regression splines (MARS), a quantile regression model with LASSO penalites, and a simple Neural net model. We apply CV twice. First, a particular model type is calibrated to find the best hyper-

[3] CV refers to various types of out-of-sample testing techniques (e.g. delete-1 CV, k-fold CV, etc.) that to prevent overfitting split the dataset (once or multiple times) and then use part of it to fit a particular model and the rest of the data to measure its performance. If the dataset has been split into k parts, the overall test error is estimated by taking the average test error across k trials (Goodfellow, Bengio, and Courville, 2016).

parameters.[4] Second, similar to online data analysis competitions, we use a variation of CV – the hold-out sample – to choose the "winning" model amongst the five competitors using the 49 different metrics that we consider.[5] We illustrate how the ranking of model performance varies strongly depending on which metrics are used. Our final contribution is to suggest a short list of four metrics for general model evaluation, that differ from the standard ones used in the housing valuation literature. These four metrics all belong to the best performing class with respect to our symmetry criteria.

Although the focus of our analysis is on the housing market, these metrics and the classification system discussed in this paper should be useful in all situations where a choice between different regression models needs to be made.

2 Symmetry conditions

2.1 Some general comments

The interdisciplinary nature of the subject has made it hard for any consensus to emerge over the properties that performance metrics should satisfy when used to judge the performance of AVMs. Additionally, the terminology used varies widely across articles. In what follows, we try and use the most common terminology. Given that our focus is on the prediction of real estate prices, we will call our realized values p_n and our predictions $\hat{p}_n$, where n indexes the real estate units in the dataset. To ease interpretation and comparability we formulate (or re-formulate) all metrics so that lower absolute values indicate better model performance.

Most of the performance metrics suggested in the literature to measure predictive performance fall into two categories: they are either ratio-based or difference-based. Ratio-based metrics center around the ratio of "true" output values relative to predicted values. We refer to the ratio $p_n/\hat{p}_n$ as the prediction error ratio. With difference-based metrics the difference between

[4] For this stage we use 10-fold CV, which is often recommended for this task (see Breiman and Spector, 1992, and Hastie, Tibshirani, and Friedman, 2009).

[5] See for example: www.kaggle.com/competitions.

realized values and predicted values is used to measure accuracy, either directly $(p_n - \hat{p}_n)$ or in squared form $(p_n - \hat{p}_n)^2$.

It is important to consider what properties metrics should ideally satisfy. Here we define two symmetry conditions that we argue are useful in this regard. That is not to say that metrics that do not satisfy these symmetry conditions should not be used, but rather that the user should be aware of their lack of symmetry. Furthermore we show how ratio and difference based metrics that do not satisfy these symmetry conditions can be made symmetric.

2.2 Symmetric treatment of prediction error ratios

For AVMs it is generally more accurate to express prediction errors in ratios rather than levels. For example, in most contexts, a 10,000 dollar prediction error on a house that sells for 1 million dollars is considered less bad than the same prediction error on a house that sells for 100,000 dollars. Also it is naturally appealing that the unit of measurement (e.g. whether prices are measured in Euros or US-Dollars) should not matter for the performance of the model. This principle can be expressed as follows: Suppose that for two observations m and n the prediction error ratios are the same: i.e., $\hat{p}_m/p_m = \hat{p}_n/p_n$. In this case, the predictions $\hat{p}_m$ and $\hat{p}_n$ should be viewed as equally accurate.

An implication of this principle is formulated in the following condition:

Condition 1: *When $\hat{p}_m/p_m = \hat{p}_n/p_n$, replacing observation m with a duplicate of observation n (or replacing observation n with a duplicate of observation m) has no impact on the performance metric.*

More formally, for a performance metric $M(\cdot)$ this condition can be written as follows:

$$\begin{aligned} \hat{p}_m/p_m = \hat{p}_n/p_n \Rightarrow M(p_1, \ldots, p_m, p_n, \ldots, p_N; \hat{p}_1, \ldots, \hat{p}_m, \hat{p}_n, \ldots, \hat{p}_N) \\ = M(p_1, \ldots, p_m, p_m, \ldots, p_N; \hat{p}_1, \ldots, \hat{p}_m, \hat{p}_m, \ldots, \hat{p}_N) \\ = M(p_1, \ldots, p_n, p_n, \ldots, p_N; \hat{p}_1, \ldots, \hat{p}_n, \hat{p}_n, \ldots, \hat{p}_N). \end{aligned}$$

A natural application of this principle with regards to AVMs is the switch of currencies: the accuracy of the AVM prediction should not depend on which currency the prices are measured in.

Outside the AVM context, there may be situations where condition 1 is inappropriate. For example, the use of ratios presupposes that the actual and predicted values are both positive. While this is certainly the case in an AVM setting, it will not necessarily be true for temperature readings in Celsius or Fahrenheit.[6] A second concern is that near-zero values of p_n or $\hat{p}_n$ can lead to very high prediction error ratios for certain observations that could distort some of the ratio based metrics (see Hyndman and Koehler, 2006). Again, this problem is unlikely to arise in real-estate setting.

2.3 Symmetric treatment of actual values and predictions

For AVMs it is often – but not always – useful to treat actual values and predictions symmetrically in the performance metric. Suppose a house sells for 1 million dollars, but that its predicted price is 1.1 million dollars. Consider now a second house where the actual price is 1.1 million dollars and the predicted price is 1 million dollars. Symmetry between actual and predicted values implies that the predictions on these two houses should be viewed as equally accurate. It is worth noting that there are situations where one might not want to treat p and $\hat{p}$ symmetrically. For example, Shiller and Weiss (1999) argue that mortgage providers may prefer valuations to err on the side of being too low rather than too high. However, for many AVMs this symmetry condition is appropriate.

The principle of equal treatment of actual and predicted values can be expressed as follows: Suppose that for two observations m and n the prediction error ratio of m equals the reciprocal of the prediction error ratio of n: i.e., $\hat{p}_m/p_m = p_n/\hat{p}_n$. Then predictions $\hat{p}_m$ and $\hat{p}_n$ should be viewed as equally accurate.

Condition 2: *Swapping all the p_n and $\hat{p}_n$ terms has no impact on the absolute value of the*

[6] While the issue of negative or zero temperatures could be resolved by switching to Kelvin, in the climate change literature the focus is anyway predominantly on temperature changes in levels and not ratios (see for example IPCC, 2013).

performance metric.

More formally, this condition can be written as follows:

$$|M(p_1,\ldots,p_N;\hat{p}_1,\ldots,\hat{p}_N)| = |M(\hat{p}_1,\ldots,\hat{p}_N;p_1,\ldots,p_N)|\,.$$

The reason for allowing for sign changes in condition 2 is that, for location based metrics, the sign indicates the direction of the errors. However, for all our metrics, overall performance is measured by the absolute value of the score. Therefore, if swapping p and $\hat{p}$ simply changes the sign of M, this will not affect the observed ranking of ML methods.

An example of a metric that violates this condition is the often used Mean Prediction Error (MPE) – see next section – since

$$\left|\frac{1}{N}\sum_{n=1}^{N}\left[\left(\frac{p_n}{\hat{p}_n}\right)-1\right]\right| \neq \left|\frac{1}{N}\sum_{n=1}^{N}\left[\left(\frac{\hat{p}_n}{p_n}\right)-1\right]\right|.$$

3 Common ratio and difference based metrics

3.1 Ratio based metrics

Many performance metrics used for ML models center around the prediction error ratio $(p_n/\hat{p}_n)$. Here we list the most commonly used:

Mean Prediction Error (MPE):

$$MPE = \frac{1}{N}\sum_{n=1}^{N}\left[\left(\frac{p_n}{\hat{p}_n}\right)-1\right].$$

Median Prediction Error (MDPE):

$$MDPE = med\left[\left(\frac{p_n}{\hat{p}_n}\right)-1\right], \tag{1}$$

where $med(p_n/\hat{p}_n)$ is the median value of actual price divided by predicted price.

Mean Absolute Prediction Error (MAPE):

$$MAPE = \frac{1}{N}\sum_{n=1}^{N}\left|\left(\frac{p_n}{\hat{p}_n}\right)-1\right|.$$

Median Absolute Prediction Error (MDAPE):

$$MDAPE = med\left|\left(\frac{p_n}{\hat{p}_n} - 1\right)\right|,$$

Mean Squared Prediction Error (MSPE):

$$MSPE = \frac{1}{N}\sum_{n=1}^{N}\left[\left(\frac{p_n}{\hat{p}_n}\right) - 1\right]^2.$$

In the context of AVMs, these metrics (except MDAPE) are used for example by Schultz, Wersing, and Werwatz (2014). MAPE is used by D'Amato (2007). MDAPE in reciprocal form and some other metrics (discussed below) are used by Hyndman and Koehler (2006). Ceh, Kilibarda, Lisec and Bajat (2018) use MAPE and the Coefficient of Dispersion (COD) defined below.

Coefficient of Dispersion (COD):

$$COD = \frac{1}{N}\sum_{n=1}^{N}\left|\left[\frac{p_n}{\hat{p}_n}\Big/ med\left(\frac{p}{\hat{p}}\right)\right] - 1\right|.$$

The COD metric is also used by Moore (2006) and Yacim and Boshoff (2018). Moore states that COD is widely used to measure quality in the tax assessment field. An alternative definition of COD replaces the median with the arithmetic mean (see Spüler et al, 2015). This example illustrates the importance of providing a precise formula to avoid confusion (particularly given the interdisciplinary nature of the literature).

In the economics and statistics forecasting literatures, these ratio based metrics are often defined in reciprocal form. For example, in each of the formulas above, Hyndman and Koehler (2006) replace $(p_n/\hat{p}_n) - 1$ with $1 - (\hat{p}_n/p_n)$. This provides an alternative version of the MPE formula: **MPE′**:

$$MPE' = \frac{1}{N}\sum_{n=1}^{N}\left[1 - \left(\frac{\hat{p}_n}{p_n}\right)\right].$$

Note that MPE and MPE′ are not equivalent. Similarly, MAPE′ and MSPE′ defined below are not equivalent to MAPE and MSPE.

MAPE′:

$$MAPE' = \frac{1}{N}\sum_{n=1}^{N}\left|1 - \left(\frac{\hat{p}_n}{p_n}\right)\right|,$$

MSPE′:

$$MSPE' = \frac{1}{N}\sum_{n=1}^{N}\left[1 - \left(\frac{\hat{p}_n}{p_n}\right)\right]^2.$$

Hyndman and Koehler (2006) also consider the following metrics which they attribute to Makridakis (1993):

Symmetric Mean Absolute Percentage Error (sMAPE):

$$sMAPE = \frac{1}{N}\sum_{n=1}^{N}\left(\frac{|p_n - \hat{p}_n|}{p_n + \hat{p}_n}\right).$$

However, median based measures will identify the same median observation even when in reciprocal form, and therefore one metric will be the reciprocal of the other. Hence we do not consider reciprocals of median based metrics any further here.

Symmetric Median Absolute Percentage Error (sMdAPE):

$$med\left(\frac{|p_n - \hat{p}_n|}{p_n + \hat{p}_n}\right).$$

Here we use Hyndman and Koehler's terminology for these two methods (i.e. sMAPE and sMdAPE). These so-called symmetric metrics address the dilemma over which of p_n and $\hat{p}_n$ should go in the denominator by adding them together and putting both in the denominator. In this sense these methods are symmetric. However, these two methods do not really deal with the problem of symmetry (see section 3) – a point that is also noted by Hyndman and Koehler (2006). In section 3 we show how to design metrics that are properly symmetric in their treatment of p and $\hat{p}$.

COD can also be defined in reciprocal form:

COD':

$$COD' = \frac{1}{N}\sum_{n=1}^{N}\left|\left[\frac{\hat{p}_n}{p_n}\Big/ med\left(\frac{\hat{p}}{p}\right)\right] - 1\right|.$$

The final ratio based metric considered here counts the proportion of actual values lying within a certain range from their corresponding predicted values (see Matysiak, 2017). While Matysiak does not provide an explicit formula, our interpretation of this metric is as follows:

Percentage Error Range (PER):

PER is the percentage of observations n for which the following condition is satisfied:

$$PER(x) = 100 \left| \frac{p_n}{\hat{p}_n} - 1 \right| > x,$$

where x is set to a value such as 10. For example, suppose we set x equal to 10, and this gives us the answer $PER(10) = 40$. This tells us that for 40 percent of the predictions the error is larger than 10 percent.

PER can likewise be defined in reciprocal form as follows:

PER′:

The percentage of observations n for which the following condition is satisfied:

$$PER'(x) = 100 \left| \frac{\hat{p}_n}{p_n} - 1 \right| > x,$$

The difference between PER and PER′ can be illustrated with an example. Suppose $\hat{p}_n = 1$, and $p_n = 1.11$. Then, $100|(p_n/\hat{p}_n) - 1| = 11 > 10$ while in reciprocal form $100|(\hat{p}_n/p_n) - 1| = 9.9 < 10$. In other words, here observation n lies within the specified range for PER′(10) but not for PER(10).

3.2 Metrics based on differences

There is a large number of metrics that are based on the difference between realized values and predicted values to measure accuracy, either in squared form as $(p_n - \hat{p}_n)^2$ or directly as $(p_n - \hat{p}_n)$. Note that these metrics are closely related to the L1 and L2 loss functions, in that L1 loss minimizes the error which is defined as the sum of all the absolute differences between the true value and the predicted value, while L2 loss minimizes the squared value of these differences. Measures based on squared differences put more emphasis on outliers, as due to squaring, predictions which are far away from actual values are penalized more strongly compared with predictions that lie closer.

Masias et al. (2016) use the following three metrics to evaluate AVMs:

Root Mean Squared Error (RMSE):

The RMSE is simply the root of the mean square error (often referred to as quadratic loss or L2 loss).

$$RMSE = \sqrt{\frac{\sum_{n=1}^{N} \left(p_n - \hat{p}_n\right)^2}{N}}, \tag{2}$$

Mean Absolute Error (MAE):

MAE measures the average of the sum of absolute differences between observation values and predicted values.

$$MAE = \frac{1}{N} \sum_{n=1}^{N} \left| p_n - \hat{p}_n \right|,$$

$\mathbf{R}^2$:

$$1 - R^2 = 1 - \frac{\sum_{n=1}^{N} \left(p_n - \hat{p}_n\right)^2}{\sum_{n=1}^{N} \left(p_n - \bar{p}\right)^2},$$

where $\bar{p}$ is the arithmetic mean of the observed prices.
We use $1 - R^2$ as our performance metric so that smaller values are better, which makes this measure comparable with the other metrics considered here.

To compare the performance of AVMs, Kok, Koponen and Martinez-Barbosa (2017) use MAPE, MSE and R^2 (where MSE is the square of RMSE). Yacim and Boshoff (2018) use RMSE, MAE, and R^2, Peterson and Flanagan (2009) use MAPE and RMSE, and Zurada, Levitan and Guan (2011) use RMSE, MAE, MAPE, and CC (see below). MAE is used by Smith, McClendon, and Hoogenboom (2007) to measure the accuracy of temperature predictions. Diaz-Robles et al. (2008) use RMSE, MAE, and R^2 to predict particulate matter levels in urban areas. Abdul-Wahab and Al-Alawi (2002) use R^2 to predict ozone levels. Bogin and Shui (2018) use RMSE and R^2 to compare the performance of AVMs, while Bajari, Nekipelov, Ryan and Yang (2015) use RMSE to compare methods of predicting grocery store sales. Wu, Ho and Lee (2004) use MAPE and RMSE to compare methods of predicting travel times.

Spüler, Sarasola-Sanz, Birbaumer, Rosenstiel, and Ramos-Murguialday (2015) use some additional metrics to evaluate methods of decoding neural signals in the brain. These are listed below.

Pearson Correlation Coefficient (CC):

$$1 - CC = 1 - \frac{\sum_{n=1}^{N} (p_n - \bar{p})(\hat{p}_n - \bar{\hat{p}})}{\sqrt{\sum_{n=1}^{N} (p_n - \bar{p})^2} \sqrt{\sum_{n=1}^{N} (\hat{p}_n - \bar{\hat{p}})^2}} = 1 - \frac{cov(p, \hat{p})}{sd(p) \times sd(\hat{p})},$$

where $cov(p, \hat{p})$ is the covariance between the p and $\hat{p}$ vectors, and $sd(p)$ and $sd(\hat{p})$ are the standard deviations of p and $\hat{p}$. Here again we use $1 - CC$ as our performance metric so that smaller values are better.

Normalized Root Mean Squared Error (NRMSE):

$$NRMSE = \frac{\sqrt{(1/N) \sum_{n=1}^{N} (p_n - \hat{p}_n)^2}}{(p_{max} - p_{min})},$$

where p_{max} and p_{min} are the maximum and minimum observed values of p.

In the context of predicting electricity demand, Jurado et al. (2015) use a variant on NRMSE, where they divide by a variance term and do not take the square root. Hence their metric is a normalized mean squared error (NMSE), which we do not list here as a separate measure.

Signal-Noise Ratio (SNR):

$$SNR = \frac{var(p - \hat{p})}{var(\hat{p})},$$

where $var()$ denotes the variance of the variable in question.

Voyant et al. (2017) use a variant of NRMSE, which divides by the arithmetic mean $\bar{p}$ instead of $(p_{max} - p_{min})$ for the prediction of solar radiation. In addition to NRMSE, Voyant et al. (2017) also use MAE, RMSE, MAPE, and the following metric:

Mean Bias Error (MBE):

$$MBE = \frac{1}{N} \sum_{n=1}^{N} (p_n - \hat{p}_n).$$

Note that the mean here can likewise be replaced by a median, which then becomes the Median Bias Error (MDBE).

Median Bias Error (MDBE):

$$MDBE = med_n (p_n - \hat{p}_n).$$

In a survey paper on forecasting wind power generation, Foley et al. (2012) discuss the following metrics referred to above: MBE, MAE, RMSE, NRMSE, MAPE, R^2, CC. In addition, they also consider the Standard deviation of the errors (SDE).

Standard deviation of the errors (SDE):

$$SDE = sd(p - \hat{p}),$$

where $sd(\cdot)$ denotes the standard deviation between the vectors of realized and predicted prices.

3.3 Some additional Metrics

Even though a number of additional metrics could be considered, we limit ourselves here to two extensions that we think could benefit the AVM literature. One is the group of dissimilarity metrics from the price index literature. The other is a group of metrics that are robust to outliers in the data source.

3.3.1 Dissimilarity metrics

Diewert's dissimilarity measures (see Diewert 2002, 2009) were proposed in the price index literature for measuring the dissimilarity of price vectors across time periods or countries (for an application see e.g. World Bank, 2013). We believe that these metrics are equally well suited to measure the dissimilarity between actual and predicted values in ML models. Here we present two of the three dissimilarity metrics proposed by Diewert (2002, 2009).[7]

Dissimilarity Metric 1 (DM1):

$$DM1 = \frac{1}{N} \sum_{n=1}^{N} \left[\frac{\hat{p}_n}{p_n} + \frac{p_n}{\hat{p}_n} - 2 \right].$$

Dissimilarity Metric 2 (DM2):

$$DM2 = \frac{1}{N} \sum_{n=1}^{N} \left[\left(\frac{\hat{p}_n}{p_n} - 1 \right)^2 + \left(\frac{p_n}{\hat{p}_n} - 1 \right)^2 \right].$$

Both DM1 and DM2 equal zero when the vectors of predicted and realized values are the same.

[7] It turns out that the third metric corresponds to LMSPE which will be discussed in 4.2.

3.3.2 Robustness to outliers in the error distribution

The extent to which model performance is robust with respect to extreme values is an important consideration in many ML implementations. Median based metrics (such as MDPE and MDPE') are more robust measures of central tendency in the error distribution than means (see for example Wilcox and Keselman, 2003). Similarly, the interquartile and 90-10 quantile ranges, which are discussed below, are more robust measures of dispersion than variance-based measures such as MSPE, RMSE, or mean absolute deviation measures such as MAPE. Hence quantile based metrics for measuring the dispersion of the error distribution are useful additional diagnostic tools. These metrics can be defined on the prediction errors measured as ratios or in levels, as shown below:

Inter-Quartile Range in Ratios (IQRat):

$$\text{IQRat} = \ln\left(\frac{p_n}{\hat{p}_n}\right)_{75} - \ln\left(\frac{p_n}{\hat{p}_n}\right)_{25}, \tag{3}$$

where $(p_n/\hat{p}_n)_{75}$ is the 75th percentile of the prediction error ratio distribution, and $(p_n/\hat{p}_n)_{25}$ is the corresponding 25th percentile.

90-10 Percentile Range in Ratios (9010Rat):

$$\text{9010Rat} = \ln\left(\frac{p_n}{\hat{p}_n}\right)_{90} - \ln\left(\frac{p_n}{\hat{p}_n}\right)_{10}.$$

Inter-Quartile Range in Levels (IQLev):

$$\text{IQLev} = (p_n - \hat{p}_n)_{75} - (p_n - \hat{p}_n)_{25},$$

where now $(p_n - \hat{p}_n)_{75}$ is the 75th percentile of the distribution of prediction errors in differences, and $(p_n - \hat{p}_n)_{25}$ is the corresponding 25th percentile.

90-10 Percentile Range in Levels (9010Lev)

$$\text{9010Lev} = (p_n - \hat{p}_n)_{90} - (p_n - \hat{p}_n)_{10}.$$

4 Applying the symmetry conditions

4.1 Which metrics satisfy which condition?

In Table 1 we classify the metrics introduced so far according to whether they satisfy conditions 1 and 2. Metrics that violate both symmetry conditions are referred to here as Class 0 metrics. Metrics that satisfy condition 1 but violate condition 2 (equal treatment of actual and predicted values) are referred to here as Class 1 metrics. Metrics that satisfy condition 2 but violate condition 1 are referred to as Class 2 metrics. Metrics that satisfy both conditions 1 and 2 are referred to as Class 3 metrics.

Metric Class	**Metric Name**
Class 0 metrics	$1 - R^2$, NRMSE, SNR.
Class 1 metrics	MPE, MDPE, MAPE, MDAPE, MSPE, MPE′, MDPE′, MAPE′, MSPE′, COD, COD′, PER, PER′
Class 2 metrics	sMAPE, sMdAPE, RMSE, MAE, 1-CC, MBE, MDBE, SDE, IQLev, 9010Lev
Class 3 metrics	DM1, DM2, IQRat, 9010Rat

Table 1: Metrics classification according to symmetry

None of the metrics that are traditionally used to measure the performance of AVMs satisfy both symmetry conditions (in particular note that all metrics that are based on differences automatically fail condition 1). However, four of the alternative metrics that were introduced 3.3 do satisfy both conditions. These are Diewert's dissimilarity measure 1 and 2 as well as 1QRat and 9010Rat.

We believe, that in order to be useful for the evaluation of AVM performance, metrics should at least satisfy symmetry condition 1 (i.e. symmetry with respect to the prediction errors). But this condition is violated by all metrics that fall under class 0 or class 2, which includes the most frequently used metrics in the AVM literature. For example, R^2, RMSE and MAE all violate condition 1.[8]

As discussed above, it will depend on the purpose of an AVM model whether symmetry condition 2 is required. When symmetry with respect to actual and predicted values is not required, metrics that fall within class 1 can be suitable for judging AVM performance. However, if both symmetry conditions should be satisfied, there are only four metrics available in Table 1: DM1, DM2, IQRat and 9010Rat. None of these metrics has been applied in the AVM literature so far. An alternative to switching to these new metrics to judge AVM performance, is to alter the existing metrics in such a way that they will satisfy our symmetry conditions. We will show how this can be done in the next sections.

4.2 Converting Class 1 to Class 3

Here we present two ways to transform Class 1 metrics into Class 3 metrics. Both imply substituting the term $(p_n/\hat{p}_n - 1)$ with an alternative formulation.

Log Solution:

$$\text{Replace } \left(\frac{p_n}{\hat{p}_n} - 1\right) \text{ with } \ln\left(\frac{p_n}{\hat{p}_n}\right).$$

It is worth noting that these two terms are first order Taylor series approximations of each other. This log solution works for MPE, MDPE, MAPE, MDAPE, MSPE, and PER. We refer to these log variants as LMPE, LMDPE, LMAPE, LMDAPE, LMSPE, and LPER. For PER, the new metric is defined as follows:

The log Percentage Error Range (LPER) is the proportion of observations n for which the following condition is satisfied:

$$\left|\ln\left(\frac{p_n}{\hat{p}_n}\right)\right| > x,$$

where x is set to a value such as 0.1 or 0.2 [given that $\ln(1+x) \approx x$ when x is small].

[8] One implication of this is that model performance becomes sensitive to the units in which prices are expressed. For example, outcomes will be sensitive to whether prices are expressed in actual units or in units of 1000s.

The log transformation for LMAPE′ and LMSPE′ takes the following form:

$$\text{Replace} \quad \left(1 - \frac{\hat{p}_n}{p_n}\right) \quad \text{with} \quad \ln\left(\frac{p_n}{\hat{p}_n}\right).$$

Note that, with this log transformation it emerges that LMAPE′=LMAPE, and LMSPE′=LMSPE.

Max-Min Solution:

$$\text{Replace} \quad \left(\frac{p_n}{\hat{p}_n} - 1\right) \quad \text{with} \quad \left(\frac{\max(p_n, \hat{p}_n)}{\min(p_n, \hat{p}_n)} - 1\right).$$

The max-min solution can be applied to MPE, MDPE, MAPE, MSPE, and PER. We refer to these methods as mmMPE, mmMDPE, etc. Interestingly, mmMPE and mmMAPE reduce to the same formula. In the case of PER, mmPER is defined as follows:

mmPER is the percentage of observations n for which:

$$\frac{max(p_n, \hat{p}_n)}{min(p_n, \hat{p}_n)} > 1 + x. \tag{4}$$

For MPE′, MAPE′, and MSPE′:

$$\text{Replace} \quad \left(1 - \frac{\hat{p}_n}{p_n}\right) \quad \text{with} \quad \left(1 - \frac{\min(p_n, \hat{p}_n)}{\max(p_n, \hat{p}_n)}\right).$$

It follows that mmMPE = mmMPE′, mmMAPE = mmMAPE′, and mmMSPE = mmMSPE′.

The log and max-min approaches therefore generate many more metrics. As far as we are aware, none of these metrics have previously been discussed or used. By construction, these metrics satisfy both conditions 1 and 2.

4.3 Converting Class 2 to Class 3

We again present two ways to transform Class 2 metrics into Class 3 metrics. Both involve substituting an alternative formulation for the term $(p_n - \hat{p}_n)$.

Log Solution:

Let $\bar{p}$ and $\bar{p}_G$ denote arithmetic and geometric means respectively, and replace p_n with $\ln(p_n)$, $\hat{p}_n$ with $\ln(\hat{p}_n)$, and $\bar{p}$ with $\ln(\bar{p}_G)$.

This log solution can be used to turn the following class 2 metrics into class 3 metrics: RMSE, MAE, MBE, MDBE, SDE. The log transformed metric LMAE = LMAPE, while LMBE =

LMPE, and LMDBE = LMDPE.

Max-Min Solution:

$$\text{Replace} \quad (p_n - \hat{p}_n) \quad \text{with} \quad \left(\frac{\max(p_n, \hat{p}_n)}{\min(p_n, \hat{p}_n)} - 1 \right).$$

This max-min solution can be used to turn the following class 2 metrics into class 3 metrics: MAE, MBE, MDBE. It follows that mmMPE = mmMAPE = mmMAE = mmMBE, and mmMDPE = mmMDAPE = mmMDBE.

5 An Empirical Comparison of Performance Metrics

5.1 The dataset

To illustrate our analysis, we estimate an Automated Valuation Model (AVM) for apartments in Austria's second largest city, Graz. The dataset was provided by the firm ZTdatenforum (www.zt.co.at) and consists of all transactions for the city of Graz for the years 2014-2017. The following variables are available for each transaction: actual transaction price, time of sale (from which monthly, quarterly and yearly variables are constructed), internal space in square meters, balcony (yes/no), parking (yes/no), garden (yes/no), cellar (yes/no), private sale or purchase directly from builder, zoning classification determining the maximum allowed building density, area usage classifications (residential, mixed, commercial, industrial), postcode, district, longitude and latitude, as well as distances to kindergartens, schools, pharmacies, park-and-ride, and the city center. In total we have 5599 transactions. The empirical application described here is for illustrative purposes and should not be interpreted as the results of a complete AVM.

5.2 The prediction framework

As noted previously, our focus is on predicting transaction prices for individual properties given their characteristics. We train the following prediction methods: Linear Regression (hedonic) model, Random Forest model, Multivariate Adaptive Regression Splines (MARS) model, a model that uses Quantile Regression with LASSO Penalties, and a Neural Net model. We use the following notation to denote the five prediction methods.

M1: Linear Regression

M2: Random Forest

M3: Multivariate Adaptive Regression Splines (MARS)

M4: Quantile Regression with LASSO Penalty

M5: Neural Net

We chose these methods because they are widely applied in the AVM literature. A short description of each of these methods is provided in Appendix 2. We use "R" (R Core Team, 2013) to perform all computations.[9]

Step 1: Cleaning the data. This step involves:

(a) Removing outliers. We set the minimum and maximum level for floor area (at $20m^2$ and $170m^2$ respectively) and also delete the bottom 5% and top 1% of the transactions with respect to square meter prices (corresponding to a m^2 price range of about 1000 to 6000 EUR). The reason for deleting more observations at the low end of the price distribution is that we have been advised by experts in the field that many of these are not normal market transactions but rather transactions between relatives or friends, or are the result of foreclosures.

(b) Centering and scaling.

We center and scale all numeric variables to lie between 0 and 1. Scaling inputs helps to avoid situations where one or several features dominate others in magnitude and as a result the model hardly picks up the contribution of the smaller scale variables, even if they are strong. Some ML algorithms are more sensitive to this than others.[10]

While some of the models can handle missing data (i.e. Random Forest, or General Boosted Model), others, like Neural Nets cannot. For this illustration – rather than imputing the

[9] There are various packages in "R" that can be used to train ML algorithms. For many ML related processes the "caret package" (Kuhn, 2008) is a starting point as it provides many different ML techniques in one comprehensive framework.

[10] For example, Neural Net algorithms do not have the property of scale invariance (see e.g. Hastie, Tibshirani and Friedman, 2009).

missing values – we only include properties for which we have full coverage in our dataset (see the list in section 6.1).

Step 2: Splitting the dataset into training and testing partitions

We randomly split the dataset into a training and a testing sample. The same training and test sets are used for all models. This test set is our hold-out sample, which is used to test prediction performance of the various ML algorithms.

Step 3: Tuning the various models

Most of the models we consider need some degree of model tuning to find the optimal hyper-parameters. This involves making comparisons between different model versions. We use grid-searches on hyper-parameters and k-fold cross-validation (in our case $k = 10$) on the training set to select between model variations of one model family.

We use the LRMSE metric to measure performance in the tuning stage. This raises an important point: Performance metrics are needed at two stages in the modelling process. First, they are needed to tune the individual ML methods. Second, they are used to compare performance across different ML methods. While we focus here on the use of metrics in the second stage, similar issues arise in the first stage. Here it is practical for us to focus on a single metric during the first stage, so as to obtain unambiguous results when tuning the model. One attraction of RMSE in this regard is that it is a standard tuning metric which can often be taken "off-the-shelf" in ML estimation packages. If it is applied to the prices in log form then it corresponds to our LRMSE metric (which satisfies both conditions 1 and 2).[11] The result of this exercise is the model specification that then gets evaluated in the final stage via the test set (hold-out sample).

Step 4: Evaluating model performance

We use the metrics described above to compare the performance of five ML prediction methods. For each method, the models are trained on the tuning sample. Performance across methods is then evaluated using the holdout sample. We use all 56 metrics discussed above to make

[11] When LRMSE is used to tune the models in the first stage, then to ensure internal consistency LRMSE should also be one of the metrics used to compare performance across ML methods in the second stage.

this evaluation, and compare how sensitive the results are to the choice of metric.

6 Metric Results and Implications

6.1 How do the rankings of prediction methods depend on the performance metric?

The comparison of the predictive performance of the five AVM models using 49 different metrics can be found in Appendix 1, Table A1.

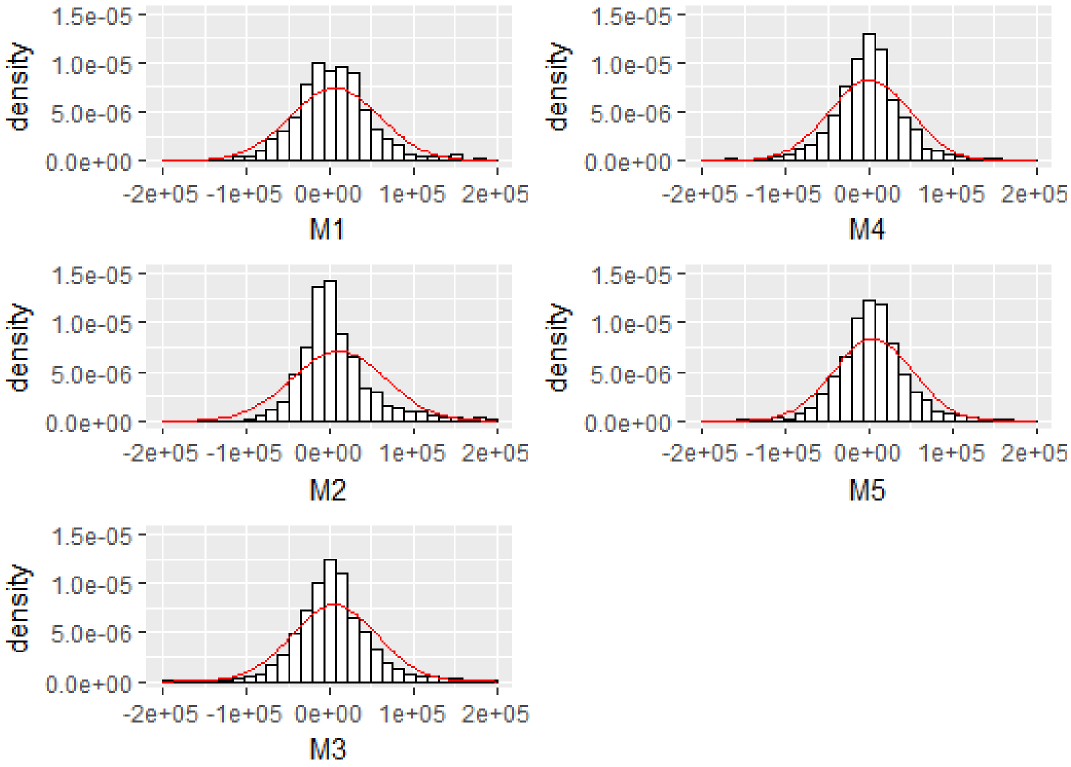
Each apartment i in the training set is characterized by a price (p_i) and a set of d characteristics $X_i = x_{i,1}, x_{i,2}, ...x_{i,d}$. The jth characteristic of the ith apartment is therefore denoted by $x_{i,j}$. With all models we attempt to predict the logarithm of price of each apartment i, $(ln(p_i))$, given its characteristics (x_i). A histogram of the errors, $(p - \hat{p})$, measured in Euros for each of these methods is presented in Figure 1.

In an AVM setting, some of the more commonly used metrics (listed in order of their appearance in section 3) are: Mean Prediction Error (MPE), Mean Absolute Prediction Error (MAPE), Coefficient of Dispersion (COD), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), and R^2. With respect to these more commonly used metrics, the best performing model is either M4 (Quantile Regression with LASSO Penalty) or M5 (Neural Net), as can be seen from Table 1.

When we expand the comparison to all 49 metrics considered above, a different picture emerges (see results in Table A1): M2 (Random Forest) performs best for 27 metrics, M4 (Quantile Regression with LASSO Penalty) is best for 11 metrics, M5 (Neural Net) is best for 14 metrics, while M1 (Linear Regression) and M3 (Multivariate Adaptive Spline Model) never perform best.

Thus the model ranking obtained from the commonly used metrics in Table 1 does not coincide with the ranking obtained from Table A1. In terms of “wins”, the methods in Table A1 are

Figure 1: Histograms of the Prediction Errors



Note: The prediction methods are labeled as follows:

M1 = Linear Regression; M2 = Random Forest (RF); M3 = Multivariate Adaptive Regression Spline (MARS); M4 = Quantile Regression with LASSO Penalty (QR); M5 = Neural Net (NN).

ranked as follows: M2, followed by M4, and M5, with M1 and M3 coming clearly last. By contrast, M2 did not rank first for any of the metrics considered in Table 1.

However, more important than the overall number of “wins” by individual models is the pattern that emerges for whole classes of metrics. M4 (Quantile Regression with LASSO Penalty) and M5 (Neural Net) dominate for class 0 metrics, M2 (Random Forest) dominates for class 1 and 3 metrics, while for class 2 metrics M4 (Quantile Regression with LASSO Penalty), M2 (Random Forest), and M5 (Neural Net) score about equally well.

From section 3 we know that class 0 metrics violate both the condition of symmetric treatment of prediction errors and the condition of symmetric treatment of actual values and predictions. Class 1 metrics satisfy the first of these conditions but violate the second. Conversely, class 2

Table 2: Model Performance Rankings Based on Metrics from the AVM Literature

	M1	M2	M3	M4	M5
MPE	5	4	3	1	2
MAPE	5	2	4	1	3
COD	5	3	4	1.5	1.5
RMSE	4.5	4.5	3	2	1
MAE	5	3	4	1	2
R^2	4	5	3	2	1

Note: The best performing metric is ranked 1 and the worst performing is ranked 5.

The prediction methods are as follows: M1 = Linear Regression; M2 = Random Forest; M3 = Multivariate Adaptive Regression Spline; M4 = Quantile Regression with Lasso Penalty; M5 = Neural Net.

metrics violate the first of these conditions while satisfying the second. Class 3 metrics satisfy both. When symmetric treatment of losses and gains is not an issue, then class 1 metrics can be used to judge AVM performance. On the other hand, if both symmetry conditions should be satisfied, class 3 metrics need to be used.

Since M2 (Random Forest) scores best with the overall ranking ("winning" for 27 of the 49 performance metrics) as well as scoring best with respect to both class 1 and class 3 metrics, we would recommend choosing this model in the current context. By contrast, if we restricted attention to the standard metrics used in the AVM literature, we would have chosen either model M4 or M5.

6.2 What set of metrics do we recommend?

It is not always practicable to calculate the performance of alternative models for almost 50 metrics. Hence there is a need to prioritize and decide which metrics are most important.

We present below four metrics that we think are particularly useful for evaluating predictive performance.

- Log Median Prediction Error (LMDPE): see (1) with $(p_n/\hat{p}_n - 1)$ replaced by $\ln(p_n/\hat{p}_n)$;
- Log Root Mean Square Error (LRMSE): see (2) with $(p_n - \hat{p}_n)$ replaced by $\ln(p_n/\hat{p}_n)$;
- Max-Min Percentage Error Range (mmPER): see (4);
- Inter-Quartile Range in Ratios (IQRAT): see (3).

The results for these metrics are shown in Table 2. These metrics all belong to class 3. Also, the metrics LMDPE, mmPER, and IQRAT are new to the literature.

LMDPE is selected as a measure of central tendency since, in addition to satisfying conditions 1 and 2, it is median based and hence also robust to outliers. LRMSE, mmPER and IQRAT all measure dispersion. mmPER and IQRAT are quantile based and hence robust to outliers. LRMSE is not robust to outliers. However, RMSE is widely available in ML modelling packages. Hence LRMSE can easily be used as loss function for ML models (by applying RMSE to the log prices). When LRMSE is used as loss function, the ML models it should also be used as one of the metrics to evaluate performance across models.

An alternative to LRMSE would be Max-Min Mean Absolute Prediction Error (mmMAPE), although compared to LRMSE it is less intuitively appealing. mmMAPE is a symmetrified version of a number of different widely used metrics, of which MAPE is just one example (see section 4.2).

$$mmMAPE = \frac{1}{N}\sum_{n=1}^{N}\left[\frac{\max(p_n, \hat{p}_n)}{\min(p_n, \hat{p}_n)} - 1\right]$$

M2 (Random Forest) performs best according to mmPER(10) and IQRAT, M4 (Quantile Regression with LASSO Penalty) performs best according to LMDPE, while M5 (Neural Net) performs best according to LRMSE. For LMDPE, the difference in the results between M2 and M4 is very small. Both medians are closely centred on the target value of zero. Hence not much weight should be given to LMDPE as a reason for preferring M4 to M2. Similarly, the difference in performance for LRMSE between M5 (0.259) and M2 (0.260) is minimal. Hence based on this short list, M2 again comes out best.

Table 3: Predictive Performance of Methods M1-M5 (Short List)

	M1	M2	M3	M4	M5
LMDPE	0.020	-0.004	0.017	**-0.001**	0.014
LRMSE	0.283	0.260	0.265	0.270	**0.259**
mmPER(10)	0.733	**0.647**	0.683	0.654	0.692
IQRAT	0.377	**0.287**	0.322	0.308	0.315

Note: The prediction methods are as follows: M1 = Linear Regression; M2 = Random Forest; M3 = Multivariate Adaptive Regression Spline; M4 = Quantile Regression with Lasso Penalty; M5 = Neural Net.

7 Conclusion

According to Kuhn (2016, p.6) the following three ingredients are necessary to build effective ML prediction models: 1) *intuition and deep knowledge* of the problem context, 2) *relevant* data, and 3) a *versatile computational toolbox* of algorithms.
We recommend adding a fourth ingredient to this list: the *appropriate choice of performance metrics* for model selection via cross-validation.

The choice of metric for evaluating the predictive performance of ML methods is a potential source of confusion in the AVM literature. A number of metrics have been proposed, but there has been little attempt to undertake a systematic analysis of their properties. We have shown here that the existing metrics in the literature do not satisfy two symmetry conditions. However, a number of new metrics are presented that do satisfy these conditions. More generally, we have imposed some structure on the list of metrics by sorting them into classes based on their properties.

Our empirical application illustrates the need to think carefully about which set of metrics should actually be used to choose between models. We evaluated the predictive performance of five ML methods using 49 different metrics. These metrics were not unanimous in their ranking of ML methods. Furthermore, we showed that focusing specifically on metrics that are frequently used in the AVM literature yields misleading results. Taking a larger set of metrics, our ranking of ML methods was quite different.

It is also important to consider how ML methods are ranked for different classes of metrics. The ranking obtained from class 3 metrics is particularly relevant, as these metrics satisfy both symmetry conditions (symmetry with respect to the treatment of prediction error ratios as well as symmetry with respect to the treatment of actual values and predictions). In our empirical application, the Random Forest model (M2) performed best in this regard.

Finally, we propose a list of four key metrics – all of which belong to class 3 and three of which are new to the literature – that we think are particularly useful for evaluating model performance.

Acknowledgements

We acknowledge financial support for this project from the Austrian Research Promotion Agency (FFG), grant #10991131. We also thank ZTdatenforum (www.zt.co.at) for providing us with the dataset.

References

Abdul-Wahab, S. A. and S. M. Al-Alawi (2002), "Assessment and prediction of tropospheric ozone concentration levels using artificial neural networks," *Environmental Modelling and Software* 17(3), 219-228.

Ahn, J.J., H.W. Byun, K.J. Oh, and T.Y. Kim (2012), "Using ridge regression with genetic algorithm to enhance real estate appraisal forecasting", *Expert Systems and Applications* 39, 8369-8379.

Akaike, H. (1973), "Information theory and an extension of the maximum likelihood principle", in Petrov, B. N.; Csáki, F., 2nd International Symposium on Information Theory, Tsahkadsor, Armenia, USSR, September 2-8, 1971, Budapest: Akadémiai Kiadó, pp.

267–281. Republished in Kotz, S.; Johnson, N. L., eds. (1992), Breakthroughs in Statistics, I, Springer-Verlag, pp. 610–624.

Antipov E. and E.B. Pokryshevskaya (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), 1772-1778.

Bajari, P., D. Nekipelov, S. P. Ryan and M. Yang (2015), “Machine Learning Methods for Demand Estimation,” *American Economic Review* 105(5), May, 481-485.

Bogin, A. N. and J. Shui (2018), “Appraisal Accuracy, Automated Valuation Models, And Credit Modeling in Rural Areas,” *FHFA Staff Working Papers* 18-03, Federal Housing Finance Agency.

Breiman, L., J. Friedman, C.J. Stone, R.A. Olshenand (1984), “Classification and Regression Trees,” Taylor & Francis.

Breiman, L. (2001), “Random Forests,” *Machine Learning* 45(1), 5-32.

Breiman, L. and P. Spector (1992), “Submodel Selection and Evaluation in Regression. The X-Random Case” *International Statistical Review / Revue Internationale de Statistique* 60(3), 291-319.

Ceh, M., M. Kilibarda, A. Lisec and B. Bajat (2018), “Estimating the Performance of Random Forest versus Multiple Regression for Predicting Prices of the Apartments,” *ISPRS International Journal of Geo-Information* 7(5), 1-16.

D’Amato, M. (2007), “Comparing Rough Set Theory with Multiple Regression Analysis as Automated Valuation Methodologies,” *International Real Estate Review* 10(2), 42-65.

Deo, R., O. Kisi, P. Singh (2017), “Drought forecasting in eastern Australia using multivariate adaptive regression spline, least square support vector machine and M5Tree model”, *Atmospheric Research* 184, 149-175.

Diaz-Robles, L. A., J. C. Ortega, J. S.Fu, G. D.Reed, J. C. Chow, J. G. Watson, and J. A. Moncada-Herrera (2008), “A hybrid ARIMA and artificial neural networks model to forecast particulate matter in urban areas: The case of Temuco, Chile,” *Atmospheric*

Environment 42(35), 8331-8340.

Diewert, W. E. (2002), "Similarity and Dissimilarity Indexes: An Axiomatic Approach," Discussion Paper 02-10, Department of Economics, University of British Columbia, Vancouver, Canada.

Diewert, W. E. (2003), "Hedonic regressions: A Consumer Theory Approach," in *Scanner Data and Price Indexes, Conference on Research in Income and Wealth*, Volume 64, Robert C. Feenstra and Matthew D. Shapiro (eds.), National Bureau of Economic Research, The University of Chicago Press, 317-348.

Diewert, W. E. (2009), "Similarity Indexes and Criteria for Spatial Linking," in Purchasing Power Parities of Currencies: Recent Advances in Methods and Applications, D. S. P. Rao (ed.). Edward Elgar: Cheltenham, UK, Chapter 8, 183-216.

Foley, A. M., P. G. Leahy, A. Marvuglia, and E. J. McKeogh (2012), "Current Methods and Advances in Forecasting of Wind Power Generation," *Renewable Energy* 37(1), 1-8.

Friedman, J. H. (1991), "Multivariate Adaptive Regression Splines," *The Annals of Statistics* 19(1), 1-67.

Goodfellow, I., J. Bengio, and A. Courville (2016),"Deep Learning", *MIT Press*, note=`http://www.deeplearningbook.org`.

Hastie, T., R. Tibshirani, and J. Friedman (2009), *Statistical Learning: Data Mining, Inference, and Prediction.* Springer Series in Statistics, Second Edition.

He, Q., L. Kong, Y. Wang, S. Wang, T. A. Chan, and E. Holland (2016), "Regularized Quantile Regression under Heterogeneous Sparsity with Application to Quantitative Genetic Traits," *Computational Statistics amd Data Analysis* 95(4), 222-239.

Hill, R. J. (2013), "Hedonic Price Indexes for Housing: A Survey, Evaluation and Taxonomy," *Journal of Economic Surveys* 27(5), December, 879-914.

Hyndman, R. J. and A, B. Koehler (2006), "Another Look at Measures of Forecast Accuracy," *International Journal of Forecasting* 22, 679-688.

IPCC (2013), "Summary for Policymakers," in: *Climate Change 2013: The Physical Science*

Basis. Contribution of Working Group I to the Fifth Assessment Report of the Intergovernmental Panel on Climate Change, Stocker, T.F., D. Qin, G.-K. Plattner, M. Tignor, S.K. Allen, J. Boschung, A. Nauels, Y. Xia, V. Bex and P.M. Midgley (eds.). Cambridge University Press, Cambridge, United Kingdom and New York, NY, USA.

Jurado, S., À. Nebot, F. Mugica, and N. Avellana (2015), "Hybrid methodologies for electricity load forecasting: Entropy-based feature selection with machine learning and soft computing techniques," *Energy* 86(15), June, 276-291.

Kok, N., E.-L. Koponen and C. A. Martinez-Barbosa (2017), "Big Data in Real Estate? From Manual Appraisal to Automated Valuation," *Journal of Portfolio Management* 43(6), 202-211.

Koenker, R. and G. Bassett (1978), "Regression Quantiles," *Econometrica* 46(1), 33-50.

Koenker, R. (2004), "Quantile regression for longitudinal data", *Journal of Multivariate Analysis*, Volume 91(1), 74-89.

Kuhn, M. (2008), "Building Predictive Models in R Using the caret Package," *Journal of Statistical Software* 28(5), 1-26.

Kuhn M, (2016), *Applied Predictive Modeling,* Springer (corrected 5th printing).

Li, Y., Y. He, Y. Su, and L. Shu (2016), "Forecasting the daily power output of a grid-connected photovoltaic system based on multivariate adaptive regression splines," *Applied Energy* 180, 392-401.

Makridakis, S. (1993), "Accuracy Measures: Theoretical and Practical Concerns," *International Journal of Forecasting* 9, 527-529.

Malpezzi, S. (2008), "Hedonic pricing models: a selective and applied review," in T. O'Sullivan and K. Gibb (eds.), *Housing Economics and Public Policy*, 67-89. Blackwell Science Ltd: Oxford, UK.

Masias, V. H., M. A. Valle, F. Crespo, R. Crespo, A. Vargas and A. Laengle (2016), "Property Valuation using Machine Learning Algorithms: A Study in a Metropolitan-Area of Chile," Paper Presented at the AMSE Conference: Santiago/Chile.

Matysiak, G. A. (2017), “Automated Valuation Models (AVMs): a brave new world?” Cracow University of Economics, Mimeo.

Milborrow, S. (2011), “earth: Multivariate Adaptive Regression Splines” R package, Derived from mda:mars by T. Hastie and R. Tibshirani.

Moore, J. W. (2006), “Performance comparison of automated valuation models,” *Journal of Property Tax Assessment and Administration* 3, 43–59.

Peterson, S. and A. B. Flanagan (2009), “Neural Network Hedonic Piricing Models in Mass Real Estate Appraisal,” *Journal of Real Estate Research* 31(2), 147-164.

R Core Team (2013). R: “A language and environment for statistical computing,” *R Foundation for Statistical Computing*. Vienna, Austria. ISBN 3-900051-07-0, URL http://www.R-project.org/.

Ripley, B. (2016), “Package ’nnet’: Feed-Forward Neural Networks and Multinomial Log-Linear Models”, https://cran.r-project.org/web/packages/nnet/nnet.pdf

Schulz, R., M. Wersing and A. Werwatz (2014), “Automated valuation modelling: a specification exercise,” *Journal of Property Research* 31(2), 131-153.

Selim, H. (2009), “Determinants of house prices in Turkey: Hedonic regression versus artificial neural network,” *Expert Systems with Applications* 36(2-2), 2843-2852.

Sherwood, B. (2017), ”rqPen: Penalized Quantile Regression”, https://cran.r-project.org/web/-packages/rqPen/index.html

Shiller, R. J. and A. N. Weiss (1999), “Evaluating real estate valuation systems,” *Journal of Real Estate Finance and Economics* 18, 147–161.

Smith, B. A., R. W. McClendon, and G. Hoogenboom (2007), “Improving Air Temperature Prediction with Artificial Neural Networks,” *International Journal of Computational Intelligence* 3(3), 179-186.

Spüler, M., A. Sarasola-Sanz, N. Birbaumer, W. Rosenstiel, and A. Ramos-Murguialday (2015), “Comparing Metrics to Evaluate Performance of Regression Methods for Decoding of Neural Signals,” *Conf Proc IEEE Eng Med Biol Soc*, August, 1083-1086.

Tibshirani, R. T. (1996), "Regression Shrinkage and Selection via the LASSO," *Journal of the Royal Statistical Society, Series B* 58(1), 267-288.

Varian, H. R. (2014), "Big Data: New Tricks for Econometrics," *Journal of Economic Perspectives*, 28 (2): 3-28.

Voyant, C., G. Notton, S. Kalogirou, M.-L. Niveta, C. Paoli, F. Motte, and A. Fouilloy (2017), "Machine learning methods for solar radiation forecasting: A review," *Renewable Energy* 105, May, 569-582.

Wilcox, R. R. and H. J. Keselman (2003), "Modern Robust Data Analysis Methods: Measures of Central Tendency," *Psychological Methods* 8(3), 254-274.

World Bank (2013), "Measuring the Real Size of the World Economy: The Framework, Methodology, and Results of the International Comparison Program (ICP)," *World Bank Publications*

Wu, Y., and Liu, Y. (2009), "Variable Selection in Quantile Regression", *Statistica Sinica*, 19, 801-817.

Wu, C.-H., J.-M. Ho, and D. T. Lee (2004), "Travel-Time Prediction With Support Vector Regression,“ *IEEE Transactions on Intelligent Transportation Systems* 5(4), December, 276-281.

Yacim, J. A. and D. G. B. Boshoff (2018), "Impact of Artificial Neural Networks Training Algorithms on Accurate Prediction of Property Values," *Journal of Real Estate Research* 40(3), 375-418.

Yang, Y. (2007), "Consistency of Cross Validation for Comparing Regression Procedures", *The Annals of Statistics* Vol. 35, No. 6, 2450–2473.

Zurada, J., A. S. Levitan, and J. Guan (2011), "A Comparison of Regression and Artificial Intelligence Methods in a Mass Appraisal Context," *Journal of Real Estate Research* 33(3), 349-387.

Zou, H. (2006), "The Adaptive LASSO and Its Oracle Properties," *Journal of the American Statistical Association* 101(476), December, 1418-1429.

Appendix 1: Results

Table A1: Predictive Performance of Methods M1-M5 (Full List)

	M1	M2	M3	M4	M5
Class 0					
1-R2	0.318	0.353	0.285	0.256	**0.255**
NRMSE	0.063	0.066	0.059	**0.056**	**0.056**
SNR	1.078	1.488	1.101	**1.018**	1.053
Class 1					
MPE	0.035	0.031	0.030	**0.011**	0.028
MDPE	0.020	-0.004	0.017	**-0.001**	0.014
MAPE	0.226	0.199	0.205	**0.198**	0.202
MDAPE	0.190	**0.146**	0.161	0.153	0.159
MSPE	0.085	0.080	0.075	0.075	**0.072**
COD	0.221	0.200	0.201	**0.199**	**0.199**
MPE′	-0.047	**-0.039**	-0.041	-0.063	-0.040
MAPE′	0.235	**0.201**	0.213	0.216	0.209
MSPE′	0.099	**0.079**	0.088	0.101	0.083
COD′	0.240	**0.200**	0.216	0.216	0.212
PER(10)	0.719	**0.629**	0.672	0.635	0.676
PER(20)	0.469	**0.380**	0.412	0.393	0.407
PER(30)	0.281	**0.219**	0.248	0.227	0.230
PER(10)′	0.718	**0.631**	0.663	0.638	0.672
PER(20)′	0.460	**0.381**	0.400	0.385	0.392
PER(30)′	0.269	**0.214**	0.230	0.227	0.222
Class 2					
sMAPE	0.111	**0.096**	0.101	0.099	0.099
sMDAPE	0.093	**0.072**	0.080	0.076	0.080
RMSE/10000	5.433	5.718	5.137	4.872	**4.865**
MAE	3.715	3.391	3.418	**3.279**	3.307
1-CC	0.170	0.140	0.149	0.137	**0.134**
MBE/1000	6.233	10.995	5.708	**1.100**	4.909
MDBE/1000	2.262	-0.605	2.753	**-0.181**	1.775
SDE/10000	5.398	5.612	5.107	4.872	**4.841**
IQLEV/10000	5.232	**4.265**	4.717	4.378	4.615
9010LEV/100000	1.124	1.022	1.028	**0.986**	1.024
Class 3					
100 × LMPE	-0.504	**-0.405**	-0.439	-2.407	-0.487
LMDPE	0.020	-0.004	0.017	**-0.001**	0.014
LMAPE	0.225	**0.194**	0.204	0.201	0.200
LMDAPE	0.174	**0.136**	0.149	0.142	0.148
LMSPE	0.085	0.080	0.075	0.075	**0.072**
LPER(10)	0.719	**0.632**	0.666	0.639	0.674
LPER(20)	0.468	**0.383**	0.412	0.391	0.400
LPER(30)	0.287	**0.221**	0.240	0.233	0.237
mmMDPE	0.206	**0.154**	0.174	0.164	0.174
mmMAPE	0.272	**0.234**	0.245	0.245	0.240
mmMSPE	0.132	0.115	0.117	0.129	**0.110**
mmPER(10)	0.733	**0.647**	0.683	0.654	0.692
mmPER(20)	0.511	**0.419**	0.447	0.427	0.441
mmPER(30)	0.347	**0.275**	0.298	0.286	0.288
LRMSE	0.283	0.260	0.265	0.270	**0.259**
LSDE	0.283	0.260	0.265	0.269	**0.259**
DM1	0.082	0.069	0.072	0.075	**0.068**
DM2	0.184	0.159	0.163	0.176	**0.155**
IQRAT	0.377	**0.287**	0.322	0.308	0.315
9010RAT	0.731	**0.639**	0.680	0.664	0.654

Note: The prediction methods are as follows: M1 = Linear Regression; M2 = Random Forest; M3 = Multivariate Adaptive Regression Spline; M4 = Quantile Regression with LASSO Penalty; M5 = Neural Net. The best performing ML method for each metric is marked in bold.

Appendix 2: Description of the applied models

Model M1: Linear Regression
The Linear Regression model (hedonic model) serves as a benchmark for the ML models. Hedonic models have originally been used in economics to model the prices for products subject to rapid technological change, such as cars and computers (see for example Grilliches, 1961). They have since become the standard model for house price regressions, especially when it comes to house price indices.[12] The hedonic model regresses the price of a product on a vector of characteristics, whose prices are not independently observed, thereby generating shadow prices for these characteristics. If the logarithm of the price is used on the left hand side, the interpretation changes slightly: instead of shadow prices on characteristics, the estimated parameters then estimate the percentage influence of these characteristics.[13] Here we regress the logarithm of price on a linear function of explanatory characteristics. All categorical variables are included as dummy variables.

Thus, the model parameters (the βs) are chosen to minimize the Sum of Squared Errors (SSE):

$$\min_{\beta} \sum_{i}^{N} (y_i - (\beta_0 + \beta x_i))^2, \tag{5}$$

where β_0 indicates the intersect term.

Model M2: Random Forest
The random forest technique was first proposed by Breiman (2001) and has since become one of the most popular ML methods. For house price predictions they have first been used by Antipov and Pokryshevskaya (2012).

Random Forests are tree-based non-parametric ensemble methods with uncorrelated decision trees as base learners. Each tree is a simple model that is built independently using a random sample of the available variables. Averaging over many independently built trees reduces variance, increases robustness, and makes the method less prone to over-fitting. Different techniques exist to construct base learners (the individual trees). The most common one is the "Classification And Regression Tree" (CART) method, also known as the recursive partitioning procedure which was proposed by Breiman et al. (1984).

Building a CART tree begins by splitting the dataset S into two groups (S_1 and S_2) so that the overall sum of squared errors (SSE) are minimized. To find the predictor and split value that minimizes the SSE, it tries out every distinct value (split point s) of every predictor (Kuhn 2016). Thus, for each variable j and each split point s, we minimize the following:

$$\min_{j,s} \left(\sum_{i \epsilon S_1} (y_i - \bar{y_1})^2 + \sum_{i \epsilon S_2} (y_i - \bar{y_2})^2 \right), \tag{6}$$

where ($\bar{y_1}$) and ($\bar{y_2}$) denote the averages of the target values of S_1 and S_2 respectively. This process is then repeated within each subgroup, continuously splitting the data into smaller

[12] For a survey of this literature see Hill (2013).

[13] See Diewert (2003) and Malpezzi (2008) for a discussion of some of the advantages of the semilog functional form in a hedonic context.

subsets.

A random forest builds an ensemble out of many such trees. Correlation between predictors is reduced by providing the algorithm with a randomly chosen number of predictors at each split (rather than the full set of available predictors). The number of predictors presented to the algorithm at each step is generally referred to as “mtry” and is the main tuning parameter of the random forest model. The other tuning parameter is the number of individual trees that are grown and averaged. We use grid searches over these tuning parameters and CV using the LRMSE metric to find the best performing version of the random forest model at this stage.

Random Forests are robust to outliers and a good method when data are noisy. A Random Forest model can consist of mixed variables (numerical and categorical), and/or contain missing values. These features – plus the fact that they are easy to implement – have made Random Forest models very popular.

Model M3: Multivariate Adaptive Regression Splines (MARS)

An example of a non-parametric extension of linear regression is the multivariate adaptive regression spline (MARS), which was introduced by Friedman (1991). The basic idea is as follows: “A piecewise polynomial function $f(x)$ is obtained by dividing the domain of X into continuous intervals and representing f by a separate polynomial in each interval.” (Hastie, Tibshirani, and Friedman, 2009). Continuity constraints are introduced to make the resulting polynomial function $f(x)$ continuous at the threshold points (knots). The “division” of the domain X is done via hinge functions – linear basis functions that identify the threshold points (knots), where a linear regression model is shifted into a different regression line. The first step in the MARS algorithm is thus the formation of these hinge functions.[14]

Following the terminology in Hastie, Tibshirani, and Friedman (2009), we can write the regression problem as follows:

$$f(X) = \beta_0 + \sum_{m=1}^{M} \beta_m h_m(X), \tag{7}$$

where each $h_m(X)$ represents a hinge function or a product of hinge functions.

The model building process of MARS is then like a stepwise linear regression using the basis functions - and their transformations - as inputs (Hastie, Tibshirani and Friedman, 2009).

The model coming out of this regression will be over-fitted and therefore needs to be trimmed back. This can be done via a stepwise term deletion procedure. One by one the terms, that are least helpful in reducing the overall error, are removed until only the intercept term remains. Each of these deletions leaves us with a possible model. We again use CV with LRMSE as the metric to select the one that fits our data best.

Regression splines provide a highly versatile regression technique that is relatively easy to implement and has many benefits: it automatically performs variable selection, variable transformation, and interaction detection. It also generally produces lower errors compared to linear regression techniques. Like linear regression, MARS is relatively easy to interpret, but is –

[14] We follow Hastie, Tibshirani, and Friedman (2009) and first create linear basis functions with a “knot” at each observed input value $x_{i,j}$, such that for each $x_{i,j}$ we have a function-pair of the form: $max(0, X_j - x_{i,j})$ and $max(0, x_{i,j} - X_j)$ (Hastie, Tibshirani and Friedman, 2009). These piece-wise linear basis functions can then be transformed by multiplying them together (which can form non-linear functions).

because of the non-parametric parts – more flexible. Due to these benefits, applications of MARS are diverse and range from forecasting grid power output (Li et al., 2016) to droughts in Australia (Deo, Kisi, and Singh, 2017).

Model M4: Quantile Regression with LASSO Penalty

First introduced by Koenker and Bassett (1978), Quantile Regression (QR) explicitly addresses a weakness of standard regression techniques: the focus on predicting in the vicinity of the mean of the distribution, and hence often poor performance away from the mean (Zou, 2006). LASSO penalties were introduced by Tibshirani (1996) as a method to perform parameter shrinkage and parameter selection in linear regression models, but can also be applied to other statistical models. LASSO stands for "Least Absolute Shrinkage and Selection Operator" and refers to a penalty in l_1 norm of the coefficient vector. It is particularly well suited to problems with sparse data (Hastie, Tibshirani and Friedman (2009)).[15]

The combination of Quantile Regression with l_1 norm shrinkage was first applied by Koenker (2004). Wu and Liu (2009) show that the inclusion of a LASSO penalty parameter can improve interpretability without losing accuracy in fit. He et al. (2016) apply a Quantile regression model with LASSO penalties to identify genetic features that influence quantitative traits. We are not aware of any applications in the housing market of this method thus far.

Unlike least-squares regression, where the coefficients are estimated by solving the least squares minimization problem, the coefficients in a linear quantile regression are chosen by minimizing the sum of asymmetrically weighted absolute errors:

$$\min_{\beta_\tau} \sum_{i=1}^{N} \rho_\tau(y_i - x_i^T \beta_\tau), \tag{8}$$

where τ refers to the individual quantile being modelled, and the weigths $\rho_\tau(u)$ are given by: $\rho_\tau(u) = \tau u$ if $u > 0$, and $-(1-\tau)u$ otherwise.[16]

After adding the LASSO penalty term, (8) becomes:

$$\min_{\beta_\tau} \sum_{i=1}^{N} \rho_\tau(y_i - x_i^T \beta_\tau) + \alpha \sum_{j=1}^{d} |\beta_{\tau,j}|. \tag{9}$$

Tuning of the model is done by choosing the number of quantiles τ and the regularization parameter α, which controls the strength of the shrinkage process (and thus also variable selection). Too much shrinking leaves a sub-optimal model, while too little shrinking tends to lead to poor interpretability (and over-fitting). Again, for model selection in the tuning phase, we use the LRMSE metric and cross-validation to choose the best-performing regularization parameter.[17]

Model M5: Neural Nets

[15] By adding the l_1 penalty term to the Error term, the LASSO exploits the bias-variance trade-off to produce models that increase the bias in the model in order to greatly reduce the model variance and thereby combat the problem of collinearity (Kuhn, 2016).

[16] Note that for each quantile τ the solution to the minimization problem yields a separate set of regression coefficients.

[17] Sherwood (2017) provides an *R*-package called "rqPen" which implements (9).

Neural Net models have a wide variety of applications, most notably in speech recognition and machine translation, computer vision (object and activity recognition), and robotics (e.g. self-driving cars). They are particularly useful when automated feature selection is needed and when the dataset is large. Our dataset, consisting of just under 6000 transactions and a limited number of variables, is rather small for a Neural Net application. Applications to the estimation of house prices include Selim (2009), Peterson and Flanagan (2009), Zurada, Levitan and Guan (2011), Ahn et al. (2012), and Yacim and Boshoff (2018).

Hastie, Tibshirani, and Friedman (2009) describe the basic idea behind Neural Nets as extracting "linear combinations of the inputs as derived features, and then modelling the target as a nonlinear function of these features". The basic structure of Neural Nets consists of an input layer, one or more hidden layers, and an output layer. Functions of increasing complexity can be modelled by adding more layers and more units within a layer (Goodfellow, Bengio, and Courville, 2016). The strength of individual connections is indicated by weights. Hidden layers find features within the data and allow the following layers to operate directly on those features rather than the entire dataset. By repeatedly adjusting the weights – the strength of individual connections between units – the error rate is minimized.[18] Thus, a Neural Net aims to minimize the errors, where the errors are considered to be a function of the weights of the network (generally the sum of squared errors or cross-entropy). However, as the global minimum of the error function would likely lead to an overfitted solution, some regularization – either stopping early or a penalty term – is needed. The penalty term is generally implemented via "weight decay", which is a penalty in $l2$ norm (Hastie, Tibshirani, and Friedman 2009).

Here, we implement a simple feed-forwards Neural Net model via the "nnet-package" in R (Ripley, 2016). This package fits a single hidden-layer Neural Network with two tuning parameters: the number of units in the hidden layer and weight decay (to avoid over-fitting). We calibrate the model via repeated grid searches on combinations of the tuning parameters.

[18] The error rate is defined as the difference between the Neural Net prediction and the observed transaction price.

Graz Economics Papers

For full list see:
http://ideas.repec.org/s/grz/wpaper.html
Address: Department of Economics, University of Graz,
Universitätsstraße 15/F4, A-8010 Graz

01–2020 **Thomas Aronsson, Sugata Ghosh and Ronald Wendner**: Positional Preferences and Efficiency in a Dynamic Economy

14–2019 **Nicole Palan, Nadia Simoes, and Nuno Crespo**: Measuring Fifty Years of Trade Globalization

13–2019 **Alejandro Caparrós and Michael Finus**: Public Good Agreements under the Weakest-link Technology

12–2019 **Michael Finus, Raoul Schneider and Pedro Pintassilgo**: The Role of Social and Technical Excludability for the Success of Impure Public Good and Common Pool Agreements: The Case of International Fisheries

11–2019 **Thomas Aronsson, Olof Johansson-Stenman and Ronald Wendner**: Charity as Income Redistribution: A Model with Optimal Taxation, Status, and Social Stigma

10–2019 **Yuval Heller and Christoph Kuzmics**: Renegotiation and Coordination with Private Values

09–2019 **Philipp Külpmann and Christoph Kuzmics**: On the Predictive Power of Theories of One-Shot Play

08–2019 **Enno Mammen, Jens Perch Nielsen, Michael Scholz and Stefan Sperlich**: Conditional variance forecasts for long-term stock returns

07–2019 **Christoph Kuzmics, Brian W. Rogers and Xiannong Zhang**: Is Ellsberg behavior evidence of ambiguity aversion?

06–2019 **Ioannis Kyriakou, Parastoo Mousavi, Jens Perch Nielsen and Michael Scholz**: Machine Learning for Forecasting Excess Stock Returns The Five-Year-View

05–2019 **Robert J. Hill, Miriam Steurer and Sofie R. Waltl**: Owner-Occupied Housing, Inflation, and Monetary Policy

04–2019 **Thomas Aronsson, Olof Johansson-Stenman and Ronald Wendner**: Charity, Status, and Optimal Taxation: Welfarist and Paternalist Approaches

03–2019 **Michael Greinecker and Christoph Kuzmics**: Limit Orders under Knightian Uncertainty

02–2019 **Miriam Steurer and Robert J. Hill**: Metrics for Measuring the Performance of Machine Learning Prediction Models: An Application to the Housing Market

01–2019 **Katja Kalkschmied and Jörn Kleinert**: (Mis)Matches of Institutions: The EU and Varieties of Capitalism

21–2018 **Nathalie Mathieu-Bolh and Ronald Wendner**: We Are What We Eat: Obesity, Income, and Social Comparisons

20–2018 **Nina Knittel, Martin W. Jury, Birgit Bednar-Friedl, Gabriel Bachner and Andrea Steiner**: The implications of climate change on Germanys foreign trade: A global analysis of heat-related labour productivity losses

19–2018 **Yadira Mori-Clement, Stefan Nabernegg and Birgit Bednar-Friedl**: Can preferential trade agreements enhance renewable electricity generation in emerging economies? A model-based policy analysis for Brazil and the European Union

18–2018 **Stefan Borsky and Katja Kalkschmied**: Corruption in Space: A closer look at the world's subnations

17–2018 **Gabriel Bachner, Birgit Bednar-Friedl and Nina Knittel**: How public adaptation to climate change affects the government budget: A model-based analysis for Austria in 2050

16–2018 **Michael Günther, Christoph Kuzmics and Antoine Salomon**: A Note on Renegotiation in Repeated Games [Games Econ. Behav. 1 (1989) 327360]

15–2018 **Meng-Wei Chen, Yu Chen, Zhen-Hua Wu and Ningru Zhao**: Government Intervention, Innovation, and Entrepreneurship

14–2018 **Yu Chen, Shaobin Shi and Yugang Tang**: Valuing the Urban Hukou in China: Evidence from a Regression Discontinuity Design in Housing Price

13–2018 **Stefan Borsky and Christian Unterberger**: Bad Weather and Flight Delays: The Impact of Sudden and Slow Onset Weather Events

12–2018 **David Rietzke and Yu Chen**: Push or Pull? Performance-Pay, Incentives, and Information

11–2018 **Xi Chen, Yu Chen and Xuhu Wan**: Delegated Project Search

10–2018 **Stefan Nabernegg, Birgit Bednar-Friedl, Pablo Muñoz, Michaela Titz and Johanna Vogel**: National policies for global emission reductions: Effectiveness of carbon emission reductions in international supply chains

09–2018 **Jonas Dovern and Hans Manner**: Order Invariant Tests for Proper Calibration of Multivariate Density Forecasts

08–2018 **Ioannis Kyriakou, Parastoo Mousavi, Jens Perch Nielsen and Michael Scholz**: Choice of Benchmark When Forecasting Long-term Stock Returns

07–2018 **Joern Kleinert**: Globalization Effects on the Distribution of Income

06–2018 **Nian Yang, Jun Yang and Yu Chen**: Contracting in a Continuous-Time Model with Three-Sided Moral Hazard and Cost Synergies

05–2018 **Christoph Kuzmics and Daniel Rodenburger**: A case of evolutionary stable attainable equilibrium in the lab

04–2018 **Robert J. Hill, Alicia N. Rambaldi, and Michael Scholz**: Higher Frequency Hedonic Property Price Indices: A State Space Approach

03–2018 **Reza Hajargasht, Robert J. Hill, D. S. Prasada Rao, and Sriram Shankar**: Spatial Chaining in International Comparisons of Prices and Real Incomes

02–2018 **Christoph Zwick**: On the origin of current account deficits in the Euro area periphery: A DSGE perspective

01–2018 **Michael Greinecker and Christopher Kah**: Pairwise stable matching in large economies