TECHNICAL UNIVERSITY OF MUNICH

TUM Data Innovation Lab

Revolution of Real Estate Valuation

Authors	Laura Altschäffel, Octav Drăgoi, Lorena Méndez, Tetsuhiro Tamada, Gusnadi Wiyoga
Mentor	M.Sc. Oliver Bachmann
Co-Mentor	Dr. Maximilian Engel
Project Lead	Dr. Ricardo Acevedo Cabra (Department of Mathematics)
Supervisor	Prof. Dr. Massimo Fornasier (Department of Mathematics)

Mar 2020

Abstract

In the real estate industry, return calculation methods available for investors are oversimplified — Many real estate firms assume annual growth rates of the property values to be constant, for instance. This project, empowered by Capital Bay GmbH, aims to improve this growth calculation approach by introducing more precise statistical and machine learning models. For this purpose, we developed four fundamentally different models based on the dataset provided by 21st Real Estate GmbH, together with the macroeconomic factors we collected from various data providers.

First, we collect socio- and macroeconomic variables, aiming to use them in predicting the rent and sale prices across Germany. Then, we train an Elastic Net in order to produce a simple, yet accurate model. Next, we investigate a stochastic process in which the house price index and the interest rate are interdependently tied. Furthermore, we implement a gradient boosted trees algorithm and train it using our datasets. Lastly, we set up vector-autoregressive models which, in addition to the prediction, enable us to capture the interrelationships among macroeconomic variables. Based on our thought process and observations, we set forth a set of hypotheses that we validate or invalidate using visualizations, R^2 accuracy metrics, and more.

Contents

1 Introduction

Capital Bay GmbH is a digitally-oriented real estate company founded in 2016. Working together with the real-time real estate evaluation software provider 21st Real Estate GmbH, Capital Bay helps investors, buyers and sellers with novel service in this industry.

The objective of this project is to develop value prediction systems for German residential real estate properties. For real estate investors, the conventional way of investment return estimation widely offered by firms, including Capital Bay, is oversimplified. In specific, it is based merely on the discounted cash flow methods where cash flows are very simply projected into the future, by assuming static annual growth rates for instance. Capital Bay provided us with this project which aims for a more scientific estimation approach of cash flows by examining different statistical and machine learning prediction models. Thereby the investor's cash flows we aim to predict are the rental income which will be incorporated periodically into Capital Bay's cash flow model, and the purchase price, which enters the cash flow model at the very end when the asset is sold.

Specifically, the following are the initial project requirements given by Capital Bay and workflow we applied to fulfill those:

- *Literature review*: In the real estate sector, a large spectrum of factors influence house prices and market rents ranging from property size to inflation rates and GDP. We reviewed the relevant literature on those factors (Section 2.1) as well as prediction modelling (Section 3.1) to guide our research forward.

- *Extension of the given dataset*: Capital Bay provided us with a dataset which contained house prices and rents of German cities as well as other socio- and macroeconomic variables such as population, migration balance and unemployment rates. We extended this dataset by collecting additional data from multiple sources (Section 2.2) and merging and transforming those datasets (Section 2.3).

- *Implementation of market rent prediction models*: The target variable specified as an initial system requirement was tile-level rents of German cities — tile refers to a geographical region of 200 m × 200 m within a city in this project. We implemented different prediction models in Python and R using Jupyter Notebook, trained them using our final dataset while having their unique benefits and drawbacks in mind. For example, certain models aim for higher accuracy in the shorter term, while others can be used for flexible indefinite time horizon predictions (Section 3.2).

- *Prediction with flexible time horizons and scenario generation capabilities*: We evaluated our models based on 3 different prediction time horizons following Capital Bay's project specification. Moreover, we included variance research into our project in order to generate meaningful future scenario distributions and confidence intervals for our predictions (see Section 3.5 and Section 3.6).

While considering evaluation approaches of implemented models, however, we updated the following requirements in a way that they better fit with Capital Bay's objectives:

- *Choice of the target variable.* Additional to market rents, we included house price as the target variable which was the focus of most of literature we found during the initial research. In addition, we decided to aim at predicting the growth rates of the real estate prices rather than their absolute values — this is proven to yield more meaningful metrics and ultimately better models (see Hypothesis 4 in Section 3.3).
- *Choice of the granularity level.* As stated above, the dataset provided by Capital Bay consisted of tile-level rents and house prices: Munich contains 293 tiles in our dataset, for instance. We decided to aim for city-level growth rate prediction for real estate prices by aggregating tiles within a city by the median (see Section 2.3). This standardized the granularity level across our models, giving us a meaningful and comparable set of metrics.

We evaluated the performance of our models by developing and testing hypotheses as described in Section 3.3 and Section 4.

2 Data

2.1 Significant Factors on House Pricing

In the interest to model and forecast real estate prices, possible factors that have significant impacts on the rent and sale prices were identified in the scientific literature. These factors can be classified in different categories:

Object Intrinsic Factors

One approach reviewed is hedonic price method (see [HM10]), also known as hedonic regression. It is based on the idea that commodities are characterized by their properties, hence the value of a commodity can be calculated by adding up the estimated values of its separate properties. In real estate, object intrinsic factors including **age and size of the building** and **floor level** are considered significant. Unfortunately, these factors are not available in our dataset and hence out of our modeling scope.

Micro-Location Factors

This refers to the elements in an immediate area that affect the rent or sale prices directly. For our modeling task, 21st provided information on the tile level, where each tile is a geographical region of 200 m × 200 m within a city and micro-location factors are defined in terms of scores for elements such as **centricity**, **nature** and **connectivity** for each specific tile. However, the scores are not comparable between cities and they were not available as a time series. For these reasons they were not considered for modelling.

Macro-Location Factors

The macro-location elements describe the demographic properties of the city or region where the property is located, for example the **unemployment rate**, **household income**, **population density**, etc. These factors are significantly contributing to the attractiveness of the property location and are used most often as a prerequisite for an investment (cf. [Kur11]).

The study [Brä+06] used factors from the real estate demand as well as supply side and highlighted **household income** and **vacancy rates** as important influencing factors. Moreover it predicts real estate prices until 2020 and provide an outlook until 2030.

Economic Factors

(Macro-)Economic factors are the factors that have a direct impact on the economy, for example **interest rates**, **tax rates**, **law**, **policies**, **wages**, and **governmental activities**. These factors induce considerable influence on the real estate investment value in the future.

The paper [YS18] suggests that the **mortgage rate** takes the first place as an external financing factor to purchase a property. Therefore, an increase or decrease in mortgage rate influences the house price movements and should be taken into account when trying to predict real estate prices. Also, in real estate markets, external funding is mostly done through mortgages, which have a high association with the **interest rate** movements.

According to [RW12], **GDP** is a major determinant of real estate prices in several regions. Also, the **current account** of Germany is considered significant since it seems to have a relationship with real estate markets as mentioned in [AJ09].

Leasing Regulation

Leasing regulation like **rent index** (Mietspiegel), **rent control** (Mietpreisbrembse) or **rent cap** (Mietendecke) are also significantly affecting an asset's value, however, these regulations are already considered elsewhere in Capital Bay's investment valuation process.

2.2 Data Collection

In order to obtain information for the modeling purposes, we conducted an extensive research on the data providers available in the market. As our project is constituted of a cooperation with a commercial company, data providers often put restrictions on the use of their data or offer some price on the commercially usable data.

Providers such as Oxford Economics Ltd., Bulwiengesa AG, Nexiga GmbH, Microm GmbH and Michael Bauer Research GmbH were contacted but then filtered out due to these conditions; although for trial purposes, Oxford Economics has provided us information about six cities (two cities for each level of rental growth: low, medium and high).

All the variables used in this project come from the following sources: 21st Real Estate GmbH, Das Statistische Bundesamt, Deutsche Bundesbank, The Organisation for Economic Cooperation and Development (OECD), INKAR and Empirica. For more detailed descriptions, see Appendix A.1.1, where we document our painstaking process of contacting all data providers, receiving incomplete and/or unclean data, and sorting it out for our uses.

2.3 Data Transformation

The dataset provided by 21st for this project comprises 149 German cities (with each having $> 50,000$ inhabitants) in the form of tiles. Each tile is a region of 200 m × 200 m within a city. This dataset contains a total of 17,634 tiles which are randomly sampled by 21st by applying the following formula:

$$x_i = \max\{0.05 \times X_i, 100\} \tag{1}$$

where x_i denotes the number of tiles in City i sampled for this project and X_i is the total number of tiles assigned to City i in the database which 21st owns.

Figure 1 shows the distribution of cities across all German states in our dataset. In terms of the amount of tiles, Table 1 provides an example of 7 cities and the number of tiles assigned to each of them. Berlin has more than double the number of tiles in Munich which can be explained by its large population. Small cities such as Augusburg and Lüdenscheid have the minimum number of tiles defined by 21st in our dataset (100 tiles).

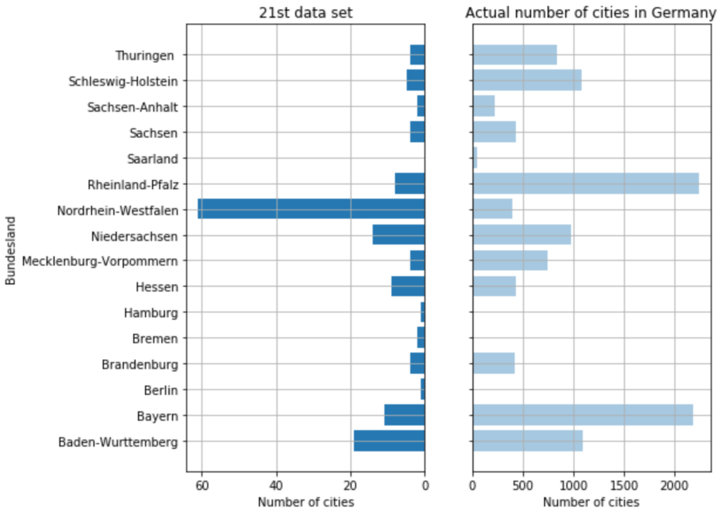


Figure 1: Number of cities in the 21st dataset vs. Actual number of cities in Germany.

City	Number of sampled tiles	Total number of tiles
Munich	293	5860
Berlin	744	14880
Hamburg	575	11500
Leipzig	189	3780
Augsburg	100	≤2000
Lüdenscheid	100	≤2000

Table 1: Number of tiles sampled for this project by 21st vs. Total number of tiles in the 21st database (see Equation 1).

Our main variables, rent and sale prices, come from 21st and are on tile level. However, since none of the other factors required for modelling were on the same spatial level, they were aggregated for each city using the median.

By combining the information of all the different data providers (as detailed in Appendix A.1.4), we obtained **58 macro-economic and macro-location factors** (see Section 2.1), rent and sale prices which are detailed in Appendix A.1.2. Since all of these variables have different granularities and time-frames, we classified them in 9 different groups as indicated in Table 2.

group	spatial level	time level	period	number of variables
0	city	quarter	2011 - 2018	2
1	city	year	2012 - 2018	1
2	city	year	2011 - 2017	16
3	city	year	2009 - 2017	19
4	city	year	2009 - 2016	4
5	country	quarter	2009 - 2018	4
6	country	quarter	2009 - 2018	2
7	country	year	2009 - 2018	6
8	country	year	2009 - 2018	2
9	state	year	2009 - 2018	4

Table 2: Classification of variables according to their granularities and periods. Groups 6 and 8 are similar to groups 5 and 7, respectively, the difference relies on the fact that variables of groups 6 and 8 represent growths as shown in Appendix A.1.2.

Variables from groups 5-9, for which their spatial level is country or state level, are treated equally for all cities in the respective time frames and region.

Even when almost 90% of the features were on a yearly basis, the information was not aggregated into this time level because the rent and sale prices are available for only 8 years. This means, an 8-point time series for each city and a set to train the models even smaller. Therefore, all the features at yearly level (groups 1-4 and 7-9) were expanded at quarterly level by taking the value of the corresponding year for all the quarters.

With the exception of groups 6 and 8, the growths of the variables were included in the final dataset. These growths were calculated based on 3 different ”lookback periods”:

1. **quarter on quarter:** rate of change in performance between one quarter and the previous quarter (included only for groups 0 and 5).
2. **year on year:** rate of change in performance between one quarter and the same quarter from the previous year.
3. **two-year on two-year:** rate of change in performance between one quarter and the same quarter from the 2 years before.

By applying these different growth calculation approaches to the collected 58 macro-economic and macro-location factors, we generated the final dataset consisting of **178 features at city-quarterly level for 149 cities in Germany**, including sale, rent prices and their growths as described in the next section.

2.4 Feature Selection

6 different targets were defined in order to predict the rent and sale prices: **quarter-on-quarter, year-on-year and 2year-on-2year growth of rent and sale prices**, as described in Section 2.3 and Section 3.3.

Due to the short time frame we were dealing with, we excluded for all the targets:

- year-on-year growth of group 1.
- 2year-on-2year growth of groups 0, 1 and 2.

Additionally, we excluded group 4 and its transformations when predicting quarter-on-quarter and year-on-year targets. Due to the data availability and empirical results of the Feature Regression, **the time windows of the targets** are: 2012Q2 - 2018Q1 (6 years) for quarter-on-quarter growth, 2013Q1 - 2018Q4 (6 years) for year-on-year growth and 2014Q1 - 2018Q4 (5 years), for 2year-on-2year growth of rent and sale prices as illustrated in Figure 2.

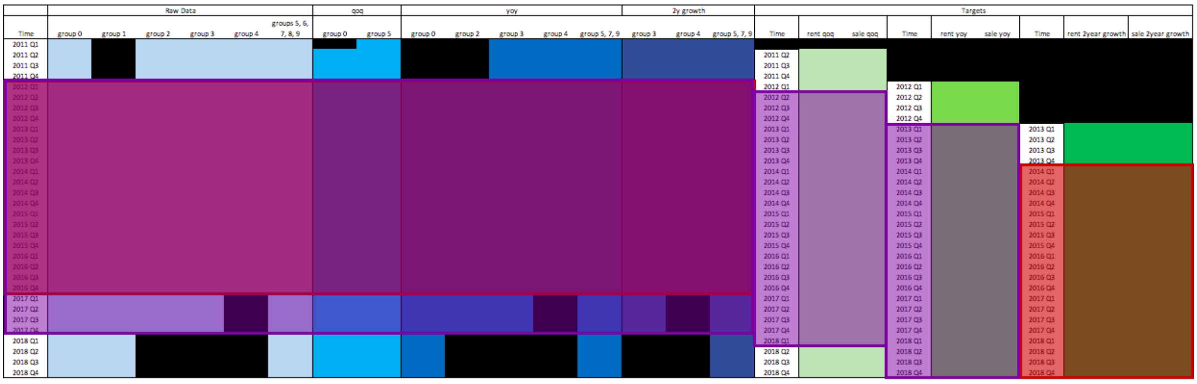


Figure 2: Time windows of the targets. The different shades of blue represent a transformation of the acquired variables, the shades of green stand for the targets and the black areas denote the missing values. As for shadows, the red shadow represents the data considered for modelling the 2year-on-2year growths and the predictions with respect to this target; the purple shadow is the information used for predicting quarter-on-quarter and year-on-year growths and the respective predictions.

Corresponding to each target, the relevant features were selected based on the Pearson's correlation coefficient (see Appendix A.2). For each pair of variables whose correlation coefficient is above 0.8 (in terms of absolute value), we identified one of the variables to be removed. The feature that was removed is the one that has less correlation (in terms of absolute value) with the target. The specified threshold of 0.8 was taken based on empirical results according to the feature regression 3.5.

The benefits of performing a feature selection are:

- **Reduction of overfitting:** Less redundant data means less opportunity to make decisions based on noise.
- **Improvement of accuracy:** Less misleading data means that modeling accuracy improves.
- **Reduction of training time:** Fewer data points reduce algorithm complexity and thus enable models to be trained faster.

As a result of this process we obtained **6 different datasets** A.1.3, each one of them corresponding to a specific target. These datasets were used to fit the feature regression

(Section 3.5) and the XGBoost model (Section 3.7). For the stochastic model (Section 3.6), the only factor used to predict the targets is the mortgage rate.

Whereas for VAR models, in order to avoid an overparametrization due to the large number of exogenous features, only a few number of features were chosen for each target. The selected variables and the selection methodology using an iterative forward feature selection algorithm are described in Section 3.8.

2.5 Trends of Real Estate Prices in Germany

In this section, we present our findings on the trends regarding real estate prices in Germany by visualizing the data provided by 21st. Overall, rents and house prices increased in all German states from 2011 to 2018 as shown in Figure 3. However, the graph also suggests the existence of a country-wide phenomenon of price and rent fall around the end of 2016 which then rises back to the previous trend at the beginning of 2017. The data provider 21st is aware of this effect and confirmed that this behavior is not caused by any error in its price calculation model. Figure 4 illustrates the distribution of tile-level house prices per square meter in Bavarian cities broken down by quarters. As shown in the graph, Munich had house prices per square meter ranging from €5000 to €9000 in the first quarter of 2018. In terms of the house price growth, Berlin had the highest growth rate between 2011 and 2018 as suggested in Figure 5: The house price in 2011 more than doubled in 2018 in contrast to Lüdenscheid where the house price remained almost constant during the time period.

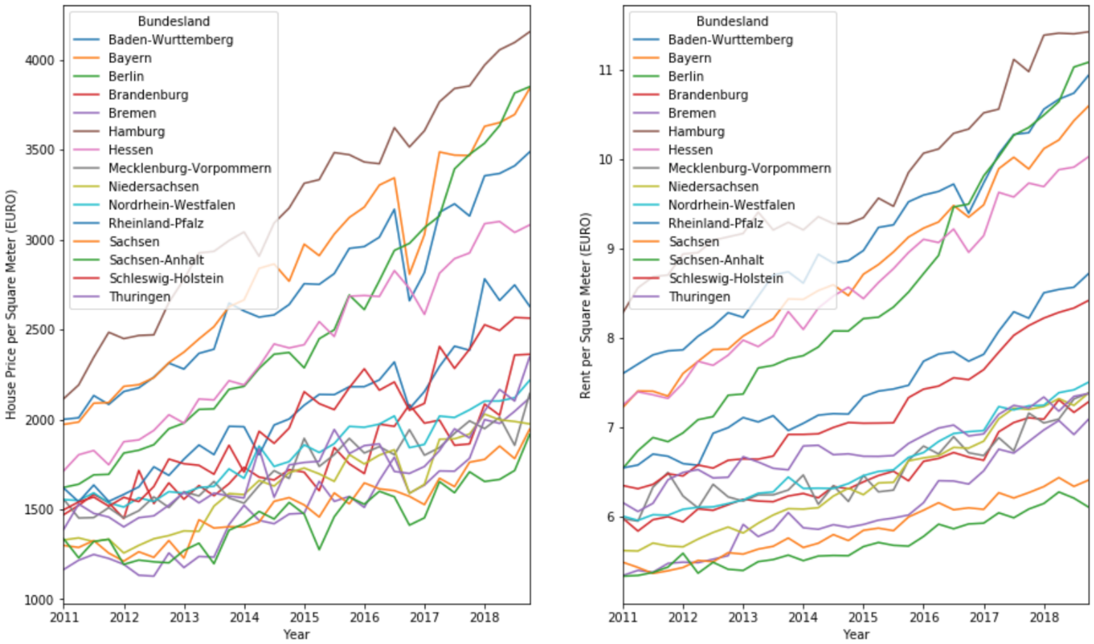


Figure 3: Development of rent and house price in Germany according to the 21st data.

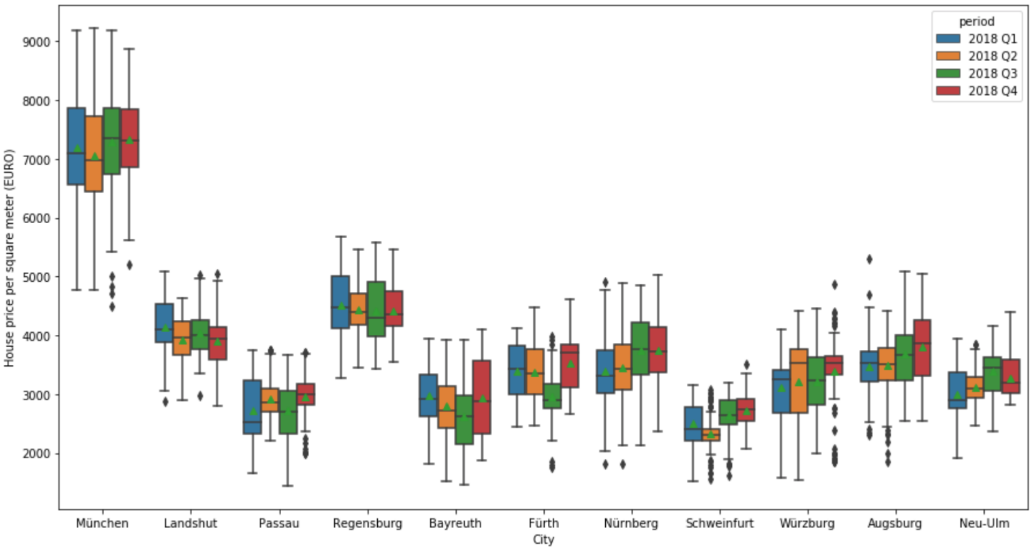


Figure 4: Distribution of house price across tiles in Bavaria in 2018.

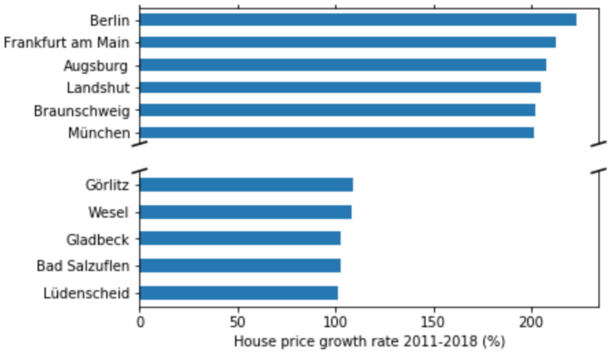


Figure 5: German cities with the highest/lowest house price growths according to 21st data. Here, growth rates are calculated by comparing the median house price of one city in 2018 to the median house price of the same city in 2011.

3 Model Implementation

3.1 Prework and Research

When modeling and forecasting real estate prices, straightforward ideas including **feature regression analysis**, **time series models** (also called autoregressive models) and **vector autoregressive models** have been thoroughly discussed in [BT10]. This book assumes no prior knowledge of econometrics but introduces a broad range of quantitative techniques that are relevant for the analysis of real estate data. For practical purposes, numerous detailed examples such as modeling Sydney office market that uses information from estimated market rents and vacancy rates and modeling Helsinki office capital values which is based on the discounted cash flow (DCF) model have been provided.

Further research on **autoregressive models** is conducted in [GKM11], which compares the forecasting performance of diferent models: VAR, factor-augmented VAR (**FAVAR**), and various Bayesian time series models (**BVAR**) utilizing a large set of macroeconomic variables and spatial priors. A deeper discussion on BVAR is found in [Lit86], which demonstrates that this inexpensive, reproducible statistical technique is as accurate, on average, as those used by the best known commercial forecasting services and considers the problem of economic forecasting, the justification for the Bayesian approach, its implementation, and the performance of one small BVAR model over the past. Another type of autoregressive model is called structural VAR (**SVAR**), which can be used to examine the dynamic relationship between house prices, income, interest rates, housing permits and share prices, as illustrated in [Yİ17].

These VAR methods have the similarity that they can be applied to multiple time series only on a single section (a single tile or city). In order to apply a calibration on the whole panel data with multiple time series across different sections (such as countries, cities, sectors, markets or combinations of these) an application of panel VAR (**PVAR**) can be referred to [CC13] and [DG14].

As 21st data is encrypted using id numbers for which exact location and coordinates cannot be traced back, any **spatial model** must be omitted. This includes the ones found in [Pac+00], [YS16] and [Cro+] such as Spatial Error Models (SEM), Spatial Durbin Models (SDM), Spatial Classification or Geographical Weighted Regression. [Cro+] introduced a novel four-stage methodology for real-estate valuation and showed that space, property, economic, neighbourhood and time features are all contributing factors in producing a house price predictor in which validation shows a 96.6% accuracy on Gaussian Process Regression beating regression-kriging, random forests and an M5P-decision-tree.

Modeling using **stochastic processes** has been proposed by [YS18], where the price change in housing markets is defined in the form of stochastic differential equations (SDEs). It proposes a general SDE system on the price structure in terms of house price index and mortgage rate and a calibration of the relevant parameters via a discretization procedure. It shows that stochastic models are flexible in terms of the choice of structure and compact with respect to the number of exogenous variables involved.

For modeling the **market rents**, a derivation from the prediction of the house prices is possible. In [KA16], four different estimation techniques are compared to test for differences in the measured relationship between rents and prices. [Gal08] uses error-correction models and long-horizon regression models to examine how well the rent–price ratio predicts future changes in real rents and prices. The results lend empirical support to the view that the rent–price ratio is an indicator of valuation in the housing market.

3.2 Model Overview

Here is an overview of the models we have implemented, why we chose them, what the main features are, and how the pros and cons stack up against each other.

1. **Constant Model**, detailed in 3.4.
   - **Pros:** Simple calculation.

- **Cons:** It serves only as a baseline which doesn't include any additional feature.

2. **Feature Regression**, detailed in 3.5.
    - **Pros:** Despite the simplicity of this model, it has many desirable characteristics for us, such as feature selection and straightforward interpretability.
    - **Cons:** The model's simplicity also proves detrimental when dealing with features that are not directly linearly correlated with our target. Other models, such as `XGBoost`, can appropriately use nonlinear features, at the cost of added complexity.

3. **XGBoost**, detailed in 3.7.
    - **Pros:** XGBoost can capture non-linear relationship in the data and performs implicit variable selection. Furthermore, the algorithm is highly efficient and flexible. It has a good model performance and a higher computational speed than other implementations of gradient boosting.
    - **Cons:** The models are hard to interpret and various parameters have to be tuned in order to fit the model.

4. **Stochastic Models**, detailed in 3.6.
    - **Pros:** Variable horizon predictions, great for forecasting very long periods of time, may be useful in cash flow analyses.
    - **Cons:** Uses only one additional exogenous variable, the mortgage interest rate; adding more features turns out to be very difficult, mathematically. Because of this, the results might be worse than other models.

5. **Vector Autoregressive (VAR) Models**, detailed in 3.8.
   VAR
    - **Pros:** VAR is an established model and the literature is numerously available (e.g. the book [Lüt05]). It can capture the interdependencies between macroeconomic variables (the features) well. Forecasts for the features and the target are scenario-consistent. Python packages are readily available as an open source tool for statistical analysis such as `tsa.vector_ar`.
    - **Cons:** VAR can only be applied for one single section (one aggregated city or one tile). Due to this fact and to the size of our data, this model doesn't provide a reliable prediction for our target variable. Furthermore, it was only possible to integrate a few influencing factors into our model.

   Bayesian VAR
    - **Pros:** In addition to VAR, it is more robust and can reduce the risk of overfitting due to its iterative nature for updating the prior distribution of the hyperparameters using the observed data.
    - **Cons:** In addition to the cons of VAR, BVAR packages are not well established in Python, but packages are available in the `R` language. Furthermore, BVAR face an increased complexity compared to VAR.

3.3 Common Methodology and Hypotheses

One of the biggest pitfalls of the data is the relatively short timespan of the targets and the variables. Therefore, when constructing lagged or growth variables, or considering targets further out in the future, this timespan can be reduced to as little as 3-4 years. Having this in mind, we try to stay away from very lagged features or long-term predictions, since we have no effective way of achieving good results with our dataset. All the transformations performed and the final time windows we considered for the final dataset are illustrated in Figure 2 and clarified in Sections 2.3 and 2.4.

Due to the time-window limitations aforementioned, for the feature regression (Section 3.5), XGBoost (Section 3.7) and VAR models (Section 3.8), we split the dataset in different periods according to the targets:

- **quarter-on-quarter and year-on-year targets:**
  - train set: 4 years (2012 - 2015), 66% of the data.
  - test set: 2 years (2016 - 2017), 33% of the data.
- **2year-on-2year targets:**
  - train set: 4 years (2012-2015), 80% of the data.
  - test set: 1 year (2016), 20% of the data.

As for the stochastic model detailed in Section 3.6, we split our dataset so that the test set spans over 11 quarters ($>$ 2016), and the train set covers the remaining previous 4 years (2012-2015), aiming to have verifiable growth predictions over at least 2 years (2016-2017).

We train the models on the train set, and report the R^2 results exclusively for the test set. For each time horizon (1, 4 and 8 quarters respectively), we take every quarter from the test set as a start date, and iterate the projection for the required number of quarters in the future. Using these numbers as the predicted variable, we can compute the R^2 against the observed variable, as well as transform it back into growth and compute R^2 for that.

Since the time horizon of our dataset is so short, we can hypothesize the following:

Hypothesis 1 *Shorter term predictions will yield much better results than long term, irrespective of the model under supervision.*

In determining our target variable, we faced a choice between the house price index `sale_cell` and house rent prices `rent_cell`. Taking the literature into consideration, most of the data we have available is more useful in predicting the prices, rather than rents, so we arrive at our second hypothesis:

Hypothesis 2 *All else equal, training the same models to predict `sale_cell` will yield better results than predicting `rent_cell`.*

We compute the R^2 of the predicted values against the real ones, which is a statistical measure that represents how much of the dependent variable's variance is explained by our independent variable's variance (see Appendix A.2). This metric is misleading, however, when applied to the raw variable `sale_cell`, because of its high autocorrelation factor of around 80%. In other words, this year's prices are very similar to last year's, so it is easy to guess the ballpark of how much an apartment costs this year by just predicting last year's price, unchanged (see Section 3.4).

As a solution, we defined 6 different targets: **quarter-on-quarter, year-on-year and 2year-on-2year growth of rent and sale prices**. We will refer to those targets as `sale_growth` or `rent_growth`, unless something particular applies to one of them. They were calculated as described in Section 2.3. These are harder, but more interesting targets to predict. Intuitively, this way we predict how a house price changes using 3 different lookback periods, rather than whether the house is expensive or cheap, in absolute terms. To sum it up in hypotheses:

Hypothesis 3 *The variable `sale_cell` is highly autocorrelated and models predicting it will have large R^2, while `sale_growth` will be a much harder target to predict.*

Hypothesis 4 *Depending on the model, predicting `sale_growth` and transforming that back in absolute prices, will yield better results than even predicting the absolute `sale_cell` directly.*

Regarding the tile vs. city-level time series prediction, the usual bias-variance trade-off applies here, which states that models with finer granularity will have a better fit (less bias) but higher deviations (more variance). Formulating it as a hypothesis for this scenario:

Hypothesis 5 *For all models, training them to predict tile-level prices will yield models with more accurate predictions, but much wider confidence intervals.*

Going into model-specific analysis, the pros and cons highlighted in Section 3.2 can be further interpreted as a set of testable hypotheses:

Hypothesis 6 *Engineering the features for the **feature regression** model so that they are better correlated linearly with the target will improve the model's performance, since the raw features are not guaranteed to have this quality.*

Hypothesis 7 *Compared to the baseline of the **constant model**, the **stochastic model** will have similar results on the shorter horizons, but will get increasingly better as time stretches out. This is because of the long-term calibration; this model is better suited to predict long-term trends, rather than accurately capture short-term movements.*

Hypothesis 8 *The **feature regression** model will perform considerably better than the **stochastic model** specifically over shorter timespans, thanks to the predictive power of the features we designed.*

Hypothesis 9 *Also, the* ***XGBoost*** *model will perform considerably better than the* ***feature regression****, since our features most likely exhibit non-linear dependencies to the target, and the machine learning model captures them better than a linear model.*

3.4 Constant Model

As a performance baseline for our experiments we include an elementary case of statistical forecasting, the constant model. The predictions for all values in the test set are merely the values from the last observed quarter. This depends on the prediction horizon; for example, for a horizon of 4 quarters, the predicted value for Munich's rent in 2017Q2 is just the value from 2016Q2.

In mathematical terms, for a target $y_{t,i}$ indexed by time t and city label i, across a horizon of Δt, our prediction $\hat{y}$ is:

$$\hat{y}_{t+\Delta t,i} = y_{t,i}.$$

Naturally, in terms of growth, this model will always forecast 0. We realize this might not be the most realistic forecast model, but it is a good enough baseline for our purposes.

3.5 Feature Regression

Based on [Boj16], we aim to predict the rent and house prices for Germany using a linear model. To produce a more accurate model we selected an Elastic Net, which is a linear regression model trained with both L_1 and L_2 -norm regularization of the coefficients. By adding this penalty, we get lower variance compared with the multiple linear regression and also some coefficients are driven to zero. This helps to prevent overfitting, as well as selecting the most important features. The objective function to minimize is:

$$\underset{w}{min}\frac{1}{2n_{samples}}||Xw-y||^2{}_2+\alpha\rho||w||_1+\frac{\alpha(1-\rho)}{2}||w||^2{}_2$$

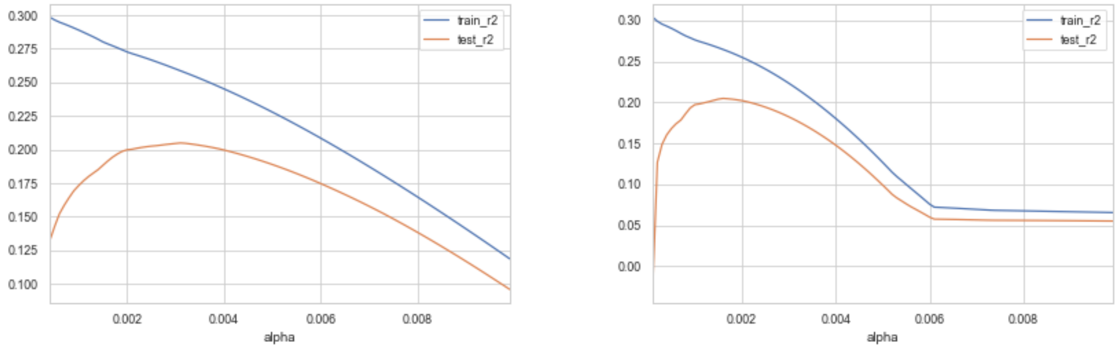
where X represents the features, w the coefficients, y the target and α and ρ are the hyperparameters.

The penalization parameter α is scanned across a relevant range, and optimized for the best R^2 out of sample. The ρ parameter, which determines the weighting between the L_1 and L_2 norms, is set either at 0.8 or 0.4, also by scanning the range from 0 to 1 and picking the one that yields the best R^2 for the optimal α.

This model uses the variables collected and their transformations (Section 2.3) as features to predict the development of house and rent prices. These features had to be selected so that the cross-correlation between them is low (see Section 2.4).

A Monte Carlo simulation was performed to generate possible scenarios for the rent and sale prices (Figure 7), by using the following procedure:

1. The Elastic Net was fitted on the corresponding dataset.

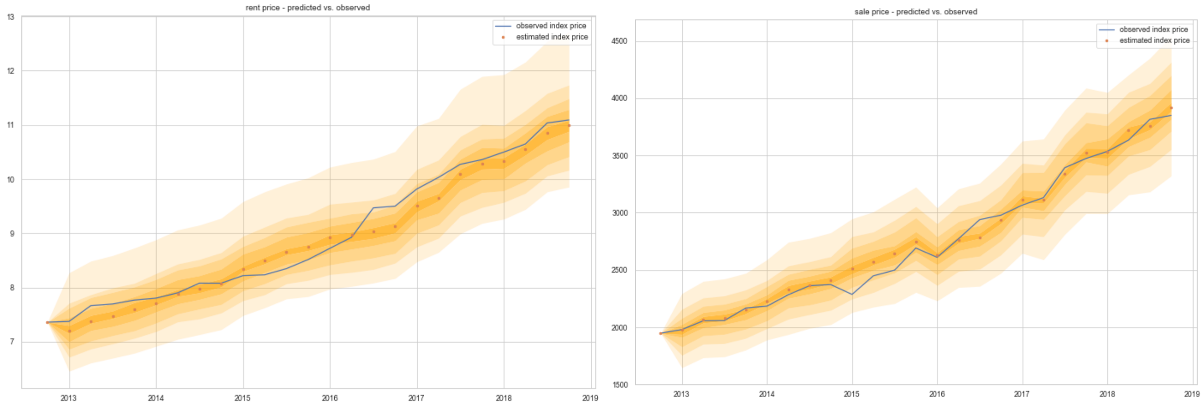


(a) R^2 of `sale_qoq` for $\rho = 0.4$.

(b) R^2 of `sale_qoq` for $\rho = 0.8$.

Figure 6: Choosing the optimal hyperparameters α and ρ for the `sale_qoq` target. $\alpha = 0.003$ and $\rho = 0.4$.

2. A multiple linear regression was performed using the variables selected by the Elastic Net (those whose coefficient is different than zero).
3. For each coefficient, a set of ”new coefficients” was obtained by sampling randomly from a normal distribution that has the coefficient as mean and the variance according to the multiple linear regression.
4. ”New predictions” were estimated using ”new coefficients” from the already generated sets.

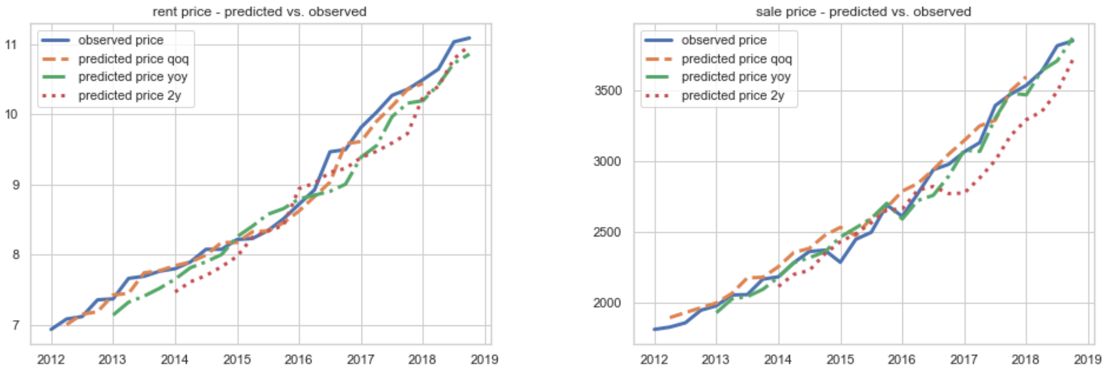


(a) Rent prices and simulations.

(b) Sale prices and simulations.

Figure 7: Predictions of the rent and sale prices of Berlin using the year-on-year growth as target. The shades of yellow represent quantiles of 100 multiple scenarios generated using a Monte Carlo simulation.

After predicting all the targets, we did the corresponding calculations to obtain the predicted rent and sale prices. These are illustrated in Figure 8. The model results are summarized in Section 4 and the coefficients of the model can be reviewed in Appendix A.3.



(a) Rent prices and predictions.

(b) Sale prices and predictions.

Figure 8: Different horizon predictions of rent and sale prices in Berlin (2012-2018), from the Feature Regression model. The blue line corresponds to the observed prices, while the dotted lines refer to the predictions for different time horizons (qoq, yoy, 2y).

3.6 Stochastic Models

3.6.1 Mathematical Background

For this modeling problem, we consider a stochastic process in which the h_t house price index's recursive equation is dependent on the interest rate r_t. Our model is based on [YS18], where the dependency between h_t and r_t is modeled as a mean-reverting recursive process. This is based on economic fundamentals which state that the interest rate dictates house prices, since a considerable part of the housing market is financed through debt and therefore the house prices naturally incorporate the cost of borrowing.

The authors assume that there is a natural long-term equilibrium level for h_t (denoted by μ_h) and r_t (denoted by μ_r); at every step, the process converges towards the mean, with an added noise term. They model this process as a system of stochastic differential equations that embed this mean-reverting process in terms of parameters $\lambda, \kappa \geq 0, \mu_h, \mu_r \in \mathbb{R}, \sigma_h, \sigma_r \geq 0$:

$$\frac{dh_t}{h_t} = \lambda(\mu_h - r_t)dt + \sigma_h dZ_t \tag{2}$$

$$dr_t = \kappa(\mu_r - r_t)dt + \sigma_r dW_t \tag{3}$$

For our purposes, we use the discretized version of this model, where the process runs in step-wise increments rather than continuously. We index the time series by $t_i, 1 \leq i \leq T$, and we allow a variable distance $\Delta t = t_{i+1} - t_i$ between time increments. For h_t and r_t, the authors consider the following discrete model:

$$h_{t_i+\Delta t} = h_{t_i} + \lambda(\mu_h - r_{t_i})h_{t_i}\Delta t + \sigma_h h_{t_i}(Z_{t_i+\Delta t} - Z_{t_i}), \tag{4}$$

$$r_{t_i+\Delta t} = r_{t_i} + \kappa(\mu_r - r_{t_i})\Delta t + \sigma_r(W_{t_i+\Delta t} - W_{t_i}) \tag{5}$$

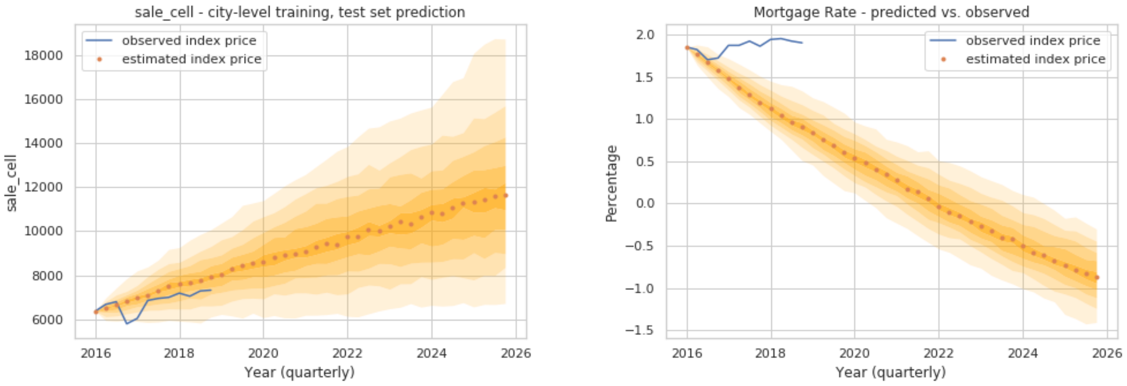
To fit the parameters, we need to minimize the noise variance and solve the optimization problem. The original paper just uses a convex solver, but in our original contribution, we found closed-formula solutions that also generalize to training on multiple data series. All of the mathematical derivations, including the formulas for the solutions, can be found in Appendix A.4.

3.6.2 Evaluation Methodology and Metrics

We train 4 sets of parameters: the convergence rate for the price series λ and for the interest rate κ, together with the optimal interest rate for each price series μ_h and for the interest rate series itself μ_r. κ and μ_r depend only on one series r_t, therefore Equations 4 and 5 can be used straight away.

To train λ and μ_h, we have to consider multiple house price time series, one for each city or tile. It makes sense for each series to have an optimal interest rate μ_h, but the convergence rate between all of them can be shared, therefore we train an individual μ_h for each series and a global λ for all of them.

One of our main goals for this model is to generate scenarios and obtain predicted distributions for the target variable. To achieve this goal, we perform $N = 300$ Monte Carlo simulations, by drawing the noise variable $Z_t \sim \mathcal{N}(0, \sqrt{\Delta t})$ and plugging it back into the equation system to get N predicted time series $\hat{h}_t$ and $\hat{r}_t$. Plots from Figure 9 are a sample of the long term prediction this model yields. They are done just like the ones in Section 3.5, by coloring equidistant quantiles for every time step with a hue that fades away proportional to how far away the quantile is from the median.

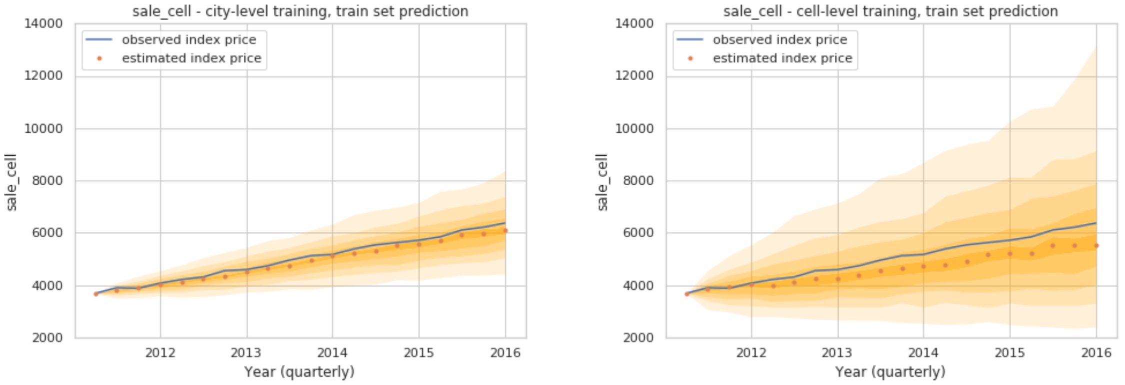


(a) House price index prediction.

(b) Interest rates prediction.

Figure 9: 10 year (2016-2026) test set prediction for Munich, with colored quantiles for the predicted distributions obtained through 200 Monte Carlo simulations.

The tile vs. city debate also applies here. This model can be trained on either tile-level or city-level, by considering h_t to be either the raw price series for each tile, or the city-level aggregated price series. In Figure 10, notice that the tile-level training yields large noise variance σ_h, and so for the actual metrics collection we decided to go forward with the higher bias, less variance model trained on city-level series.



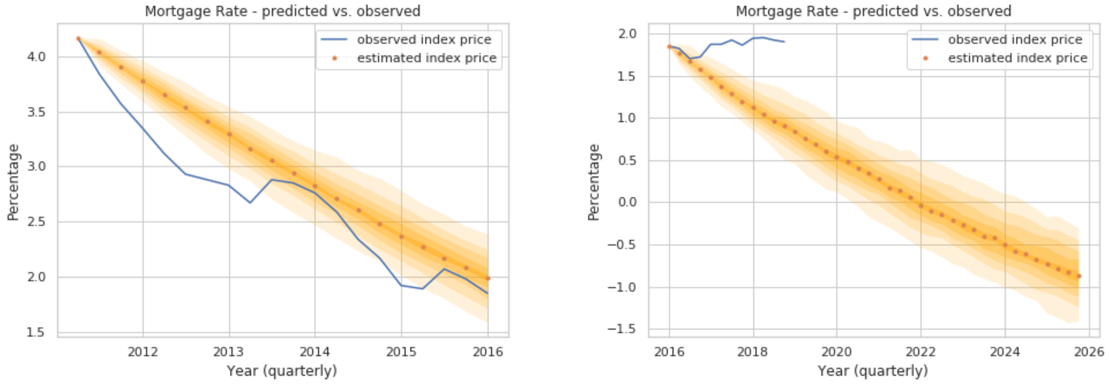
(a) Training over cities, using fewer series aggregated across tiles.

(b) Training over tiles, using all the tile-level series available.

Figure 10: Train set (2011-2016) predictions for Munich - illustration of variance difference between different calibration methods.

Given a start date and a set of historical data, this model returns a projected series of arbitrary length into the future.

Looking at the mortgage rate evolution in Figure 11, we noticed a shift in regime between the train and test sets, which possibly makes it hard for this model to accurately predict the test set targets. The reader should keep this in mind as they parse the results section. This can be mitigated with longer training datasets where different trend regimes are explored, or even with a hard bound on the convergence value for the interest rate (i.e. forcing $\mu_r \geq 0$, for example).

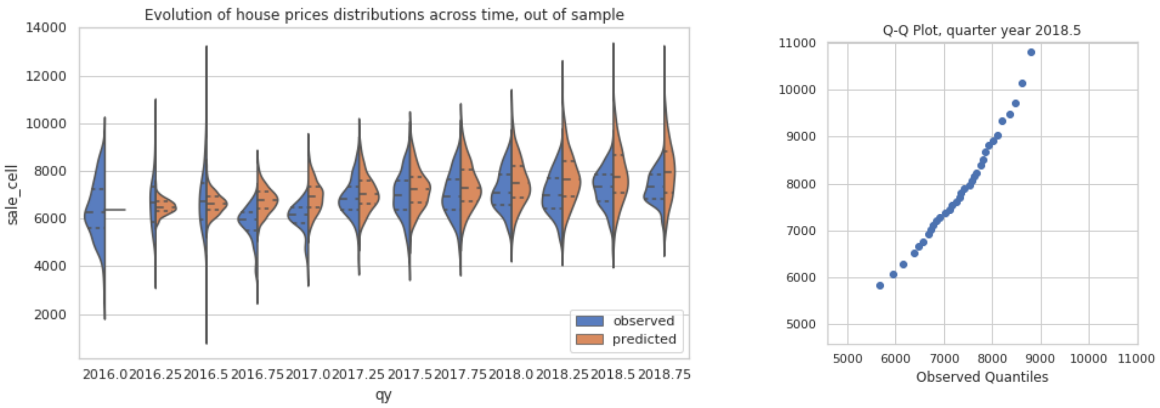


(a) Predictions over train set, showing a good fit for the trend.

(b) Predictions over test set, train set trend does not generalize.

Figure 11: Mortgage rate predictions vs. observed values, showing a poor capture of the trend shift between the train and test set.

Intrinsically, this model yields a distribution of predictions, for each city. This means that, for each city, we can compare the distribution of predicted prices with the observed distribution of prices across tiles, and see whether our model generates a similar set of predictions. The timeline of distributions, for the test set, is shown in Figure 12, together with a Q-Q plot — the means are quite similar, yet the predicted distributions tend to have much larger tails in the future.



(a) Violin plot of distributions.

(b) Q-Q plot for quarter 3 of 2018.

Figure 12: Analysis plots of predicted vs. observed distributions of `sale_cell` for Munich.

In Appendix A.5, we explore how to transform these predicted distributions for `sale_cell` to ones for `rent_cell`. This makes sense for Capital Bay, since their cash flow analysis specifically takes into consideration the rents a building would continuously generate across periods typically lasting for 10 years or more — exactly the kind of predictions this model can yield. Figure 13 shows the 10-year prediction outline for `rent_cell`, for the same scenario as in Figure 9. For graphs showing distribution comparison for `rent_cell` similar to how Figure 12 shows it for `sale_cell`, refer to the aforementioned Appendix A.5.

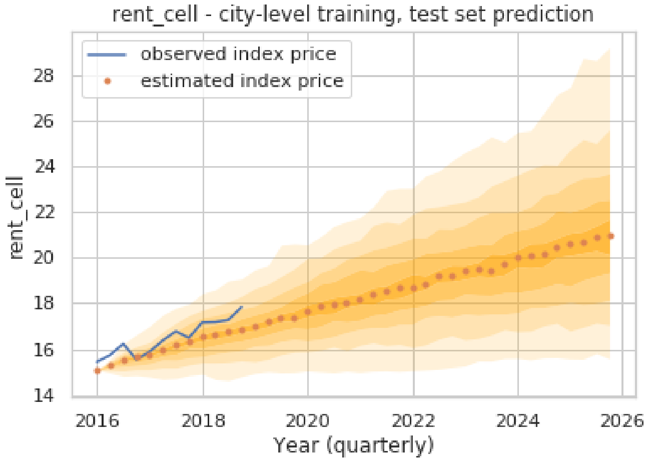


Figure 13: 10 year (2016-2026) graph showing predictions for rent prices, after taking the scenario from Figure 9 and transforming the house prices in rent prices.

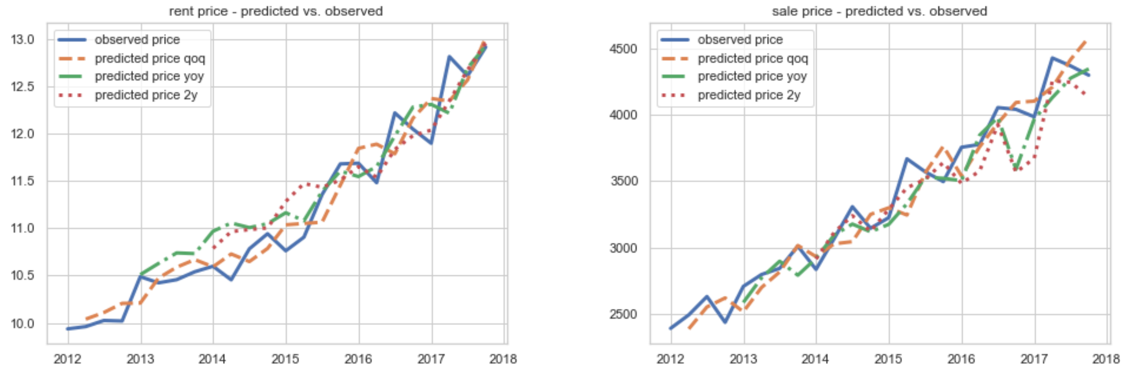
Note. As opposed to Figure 9 where the prediction seems to overshoot the actual trend, in Figure 13 the prediction actually undershoots the trend. Also, the trend prediction doesn't start from the same spot as the real value. All these artifacts can most likely be attributed to the poor fit of the linear regression that transforms `sale_cell` in `rent_cell`.

3.7 XGBoost

XGBoost is a very popular and efficient gradient boosting algorithm, which can be used for regression as well as classification. It is a supervised learning method and its objective function contains a training loss part and a regularization part. The regularization term controls the complexity of the model and prevent overfitting. The model seeks a good bias-variance trade-off. A detailed description of the model can be found in [CG16].

To fully leverage the advantages of the XGBoost model over other algorithms, different parameters need to be considered and their values have to be specified for the implementation. Therefore, we tuned a set of parameters for tree boosters that has a positive impact on the performance which includes `learning_rate`, `max_depth`, `min_child_weight`, `gamma`, `subsample`, `reg_lambda`. We used the Python library `Hyperopt` (cf. [BYC13]), which optimizes the hyperparameters of machine learning algorithms in order to automatically tune the hyperparameters for our model. `Hyperopt` takes as an input a space of hyperparameters in which it will search and moves according to the result of past trials. We have optimized the model for the best R^2 out of sample.

After finishing this optimization procedure for each of the targets and getting the predictions, we did the corresponding calculations to obtain the predicted rent and sale prices. These are illustrated in Figure 14. A summary of the results can be found in Section 4.



(a) Rent prices and predictions.

(b) Sale prices and predictions.

Figure 14: Different horizon predictions of rent and sale prices in Frankfurt (2012-2018), from XGBoost. The blue line corresponds to the observed prices, while the dotted lines refer to the predictions for different time horizons (`qoq`, `yoy`, `2y`).

An implementation of scenario prediction is not straightforward for XGBoost. Section 5 raises a question for future research with regards to simulating features which may be applied to XGBoost.

3.8 Vector Autoregressive Models

3.8.1 Mathematical Background

Standard Vector Autoregressive (VAR) Model

For modeling house prices with macroeconomic dependent variables, VAR models are commonly suggested. VAR is a multiple time series model which enables us to model and forecast a number of independent equations and simultaneously to capture the interrelationships among macroeconomic variables by an impulse response analysis (see Appendix A.6.3). It is thus often used in applied macroeconomics (see [Lüt05]).

In VAR models all variables are treated as endogenous and interdependent, both in a dynamic and in a static sense, although in some relevant cases, exogenous variables could be included. An m-dimensional and p-order vector autoregressive model $VAR_m(p)$ is defined in [CC13] as follows

$$y_t = a_0 + \sum_{i=1}^{p} \Phi_i y_{t-i} + u_t, \quad t = 1, 2, ..., T, \quad u_t \sim iid(0, \sigma_u)$$

where $y_t = (y_{1t}, y_{2t}, ..., y_{mt})$ is an $m \times 1$ vector of endogenous variables, jointly determined by its own lags and the lags of other variables. a_0 is a $m \times 1$ vector for the fixed effect, Φ_i are $m \times m$ coefficient matrices, and u_t is an $m \times 1$ matrix of unobserved shocks (disturbances).

VAR models require the estimation of many free parameters and the number of parameters to estimate grow very fast with the size of the model. This often leads to overparameterization. Since we just have limited data for training and validating models, there is an increased risk of overfitting and therefore inaccurate out-of-sample forecasts.

Bayesian VAR (BVAR) Model

To overcome these problems, we consider Bayesian VAR models, which shrink the model parameters by using informative priors. The difference with standard VAR models lies in the fact that the coefficients are considered as random quantities having their own distribution, the so-called posterior distribution. The vector $\beta := (a_0, \Phi_1, .., \Phi_p)$ consists of matrices containing the model's unknown parameters. Bayesian inference derives the posterior probability of β as a consequence of prior beliefs in combination with observed data. The conditional posterior of β can be obtained by multiplying this prior by the likelihood function. This makes the models more robust and reduces the risk of overfitting as well as the estimation uncertainty.

We use the `BVAR`-package from the `R` language (see BVAR package), which implements hierarchical Bayesian estimation of VAR models in the fashion of [GLP15]. It uses a combination of the frequently used priors, Minnesota, sum-of-coefficients and dummy-initial-observation priors for the conditional Gaussian prior for β. The draws from the posterior predictive density are generated from the training data to predict one step (quarter, year or two years) ahead. Then the procedure is integrated, updating the estimation sample step by step until the end of the test set. The posterior distribution of the hyperparameters is re-estimated at each iteration.

3.8.2 Evaluation Methodology

In VAR models, we calibrate the set of parameters $\hat{\beta} := (\hat{a_0}, \hat{\Phi}_1, .., \hat{\Phi}_p)$ consisting of one vector and p coefficient matrices of size $m \times m$. Note that p describes the lag of the variables used. Due to the problematics of the short timespan of our datasets as described in Section 3.3, we try to stay away from a very lagged model and thus we choose $p = 2$. A sequential prediction by forecasting one quarter ahead and integrating the predicted value into the training set would be another possibility to extend the time series. But prediction error will carry over through the model, hence we did not apply this approach.

Further, we try to avoid the problem of overparametrization by choosing only the most relevant features using a forward selection algorithm as described in [Sic18]. It starts by regressing on each feature individually, and then observing which feature would improve the model the most using the mean squared error (MSE) value (see Appendix A.2). Then it incorporates the winning feature into the model. It iterates through the remaining features to find the next feature which improves the model the most, again using the MSE. It does this until there are K features in the model.

For our purpose, we choose to include a maximum of $K = 5$ variables. The choice of this value is based on the experiment that adding more variables did not add the resulting R^2 tremendously. Note that since we are working with matrices, linear algebra comes into play and for some targets we have to select less variables. The overview of the variables used in the model for each target can be seen in Appendix A.6.1.

Recall that VAR models are only applicable on one section, hence we construct our model only on the city-level by aggregating all tiles within a city by the median values for each time point. Then we apply VAR model to predict the next 4 or 8 values (depending on the target) for each city.

For the evaluation purposes, we append the prediction values for all cities altogether into one vector and the corresponding test values into another vector in order to calculate the R^2 for each target as shown in Table 3. For the forecast plots and some visualizations on the macroeconomic interdependencies see Appendix A.6.2 and A.6.3.

4 Results

4.1 Evaluation Approach — Metrics and Results

Table 3 offers a comparative overview for our models, with respect to different metrics. We shall interpret it in Section 4.2 where we test hypotheses and answer questions. Here is how to read the metric names:

1. The first part denotes the time horizon for the prediction, with `1q`, `1y` and `2y` referring to one quarter, one year and two year horizons, respectively.

2. The second part indicates the type of metric, with `r2` denoting R^2.

3. The third part refers to the dependent variable, where `sale_cell` is the house price, `rent_cell` is the rent costs, and so on.

Metric	Constant Model	Feature Regression	Stochastic Model	XGBoost	VAR
`1q_r2_sale_cell`	0.911	0.913	0.829	**0.920**	0.778
`1y_r2_sale_cell`	0.842	**0.916**	0.874	0.896	0.741
`2y_r2_sale_cell`	0.691	**0.920**	0.837	0.912	0.891
`1q_r2_sale_cell_growth`	-0.017	**0.205**	≤ -1.0	0.130	≤ -1.0
`1y_r2_sale_cell_growth`	-0.251	**0.254**	-0.005	0.126	≤ -1.0
`2y_r2_sale_cell_growth`	≤ -1.0	**0.347**	-0.05	0.306	≤ -1.0
`1q_r2_rent_cell`	0.975	0.907	-	**0.979**	0.901
`1y_r2_rent_cell`	0.932	**0.977**	-	0.975	0.884
`2y_r2_rent_cell`	0.794	**0.968**	-	**0.968**	0.955
`1q_r2_rent_cell_growth`	-0.075	-0.008	-	**0.216**	≤ -1.0
`1y_r2_rent_cell_growth`	≤ -1.0	0.241	-	**0.259**	≤ -1.0
`2y_r2_rent_cell_growth`	≤ -1.0	0.251	-	**0.301**	-0.248

Table 3: Out-of-sample metrics results for trained models. Bold text references best-in-class model for predicting the corresponding target.

4.2 Hypothesis Testing

In this section, we analyze our hypotheses one by one, presenting results and answering the questions we set forth. This aims to be a comprehensive analysis of our models' features and performances.

If we look at the R^2 performance of the feature regression model, for example, we notice that Hypothesis 1 actually is **invalidated**: performances seem to increase for longer prediction time spans. We can come up with two explanations:

1. `qoq` performance is measured over 7 intervals, whereas `2y` is measured over one interval. Less data makes test results for longer periods more prone to idiosyncratic behaviour, in other words, maybe this 2 year interval is just easier to predict/fits better on our train set?

2. Quarterly measurements, as opposed to yearly ones, can suffer from seasonality problems, for which we have no mitigation in place. This may explain the much worse performance for `qoq`, compared to `yoy` or `2y`.

Comparing the performance between `sale_cell_growth` and `rent_cell_growth`, Hypothesis 2 is **validated** — scores are consistently higher when predicting house price growth versus rent growth. Even though the R^2 scores for the absolute values might look larger for `rent_cell` than for `sale_cell`, this can very well be an artifact of the different distributions for different targets.

House Price Growth Model	R^2 test score for `sale_cell_growth` prediction	R^2 test score for `sale_cell_growth` prediction which is transformed back to absolute prices	R^2 test score for `sale_cell` prediction
Linear Regression	0.196	0.844	0.835
Elastic Net	0.207	0.847	0.836

Table 4: R^2 metrics for the prediction of different targets: `sale_cell_growth` vs. `sale_cell_growth` then transforming back to absolute prices vs. `sale_cell`.

For assessing the validity of Hypothesis 3 and 4, we compare the performance between the prediction of `sale_cell` and `sale_cell_growth` with the basic regression models as shown in Table 4. Note that we calculated these values using the dataset before selecting the features, which explains the deviation to the main results.

The R^2 values for the growth prediction (first column) is around 0.2 while for absolute prediction (third column) around 0.8. This observation indeed **confirms** our Hypothesis 3 about the high autocorrelation of `sale_cell` and the difficulty of `sale_cell_growth` prediction. However, transforming the growth prediction values back to the absolute prices yield a slightly better R^2 value, which further **validates** our Hypothesis 4.

Hypothesis 5 is **validated** by looking at the residual values for each method of fitting in Figure 4, as well as inspecting visual plots of the distributions for the simulated scenarios, as we saw in Figure 10. As mentioned in Section 3.6, we decided to train on city-wide levels and report R^2 metrics as such in Table 3.

Also, Hypothesis 7 is **validated** by the results Table 3. The constant model's R^2 decays quickly, giving the stochastic model an increasing edge as the horizons grow. Notice the quite good accuracy for the constant model within one quarter, though; it beats every other model!

For Hypothesis 6, we trained the model with and without the linearized features; what we noticed in figure 15 was an increase in the training R^2, but a decrease in the test R^2, therefore actually **invalidating** our hypothesis. This might be because of overfitting, or maybe collinearity issues with the features. See the full set of figures and transformations in the respective notebook, within the codebase.

Hypotheses 8 and 9 are **validated** by the R^2 results for `rent_cell_growth` summarized in 3.

The XGBoost model and the feature regression had the best performance in R^2 terms, with extra possibility for improvement by tuning more hyperparameters and adding more train data. The discrepancies can occur due to feature nonlinearities, we discovered some that predict the target better with tree-like cuts (smaller or greater than a threshold) as opposed to applying a directly proportional linear relationship (see Figure 16).

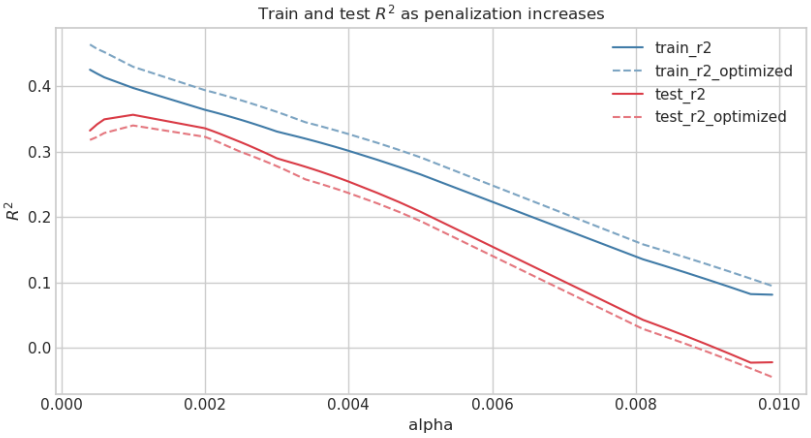
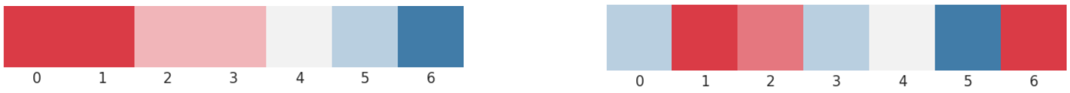


Figure 15: **Feature Regression** train and test R^2, for raw and optimized variables, as the penalty term for the elastic net increases. We used the `sklearn` elastic net package, generating this plot with $\rho = 0.4$ and target `sale_cell_2ygrowth`.



(a) Linear dependency for `sale_cell_yoy`.

(b) Nonlinear dependency for `pop_density_2ygrowth`.

Figure 16: Target (`sale_cell_2ygrowth`) heatmap across quantiles of different features. A feature like `sale_cell_yoy` that linearly correlates with the target variable would have a smooth ramp of colors across the spectrum, i.e. the target would increase/decrease proportionally with the feature. On the other hand, `pop_density_2ygrowth` has clear nonlinearities, with the target increasing, then decreasing, then increasing again, as the feature increases in value; these are the variations that models like **XGBoost** can capture.

5 Conclusion and Outlook

Through diligent construction of a clean and well documented dataset, together with extensive literature research, we managed to implement and use a number of different models that shine through their own strengths. Simple linear models like an **elastic net** have great interpretability and yield very good results, machine learning models like **XGBoost** have an advantage dealing with nonlinearities and provide accurate predictions, while trend-oriented models like our **stochastic model** generate useful long-term scenarios for predicting house prices and rents, and time series-oriented models like **VAR** models provide a scenario-consistent prediction for both the target variable (house prices or rents) and the relevant macroeconomic variables.

Throughout our work, we have had several ideas for future development that we had the chance to only partially tackle within this project. Here is what we envision to be the future developments with the highest potential:

- *City/tile clustering.* In this project, we developed city-level prediction models by simply aggregating all tiles within a city. Intuitively, clustering of tiles based on similar characteristics, however, may generate better results: a model for city center tiles, one for rich suburbs, one for tiles close to industrial areas, for instance. This has a potential of smoothing out tile idiosyncratic behaviors, ultimately generating a dataset that is better suited to train our models.

  We have briefly experimented with KNN clustering of tiles based on `sale_cell_growth` profile, yet results were inconclusive and therefore we left them out of the report. We would suggest defining a robust way of clustering tiles, visualizing and understanding it, training a model for each cluster and check the out of sample results.

- *Feature simulation.* To generate scenarios for the feature regression, we simulated the regression parameters, while for the stochastic model we simulated the noise variable. These are all relatively straightforward ways of implementing scenario prediction, as opposed to simulating the actual features used within the model; i.e. having a process to generate plausible values for household income, or GDP, rather than just changing the regression coefficient.

  This is a much more involved procedure, which requires in-depth understanding of every feature and careful calibration of the generative process. We believe that, if done right, this not only has the potential of yielding more accurate results, but also can be applied to any model that uses features, even if it is difficult to get predictions otherwise, like in the case of XGBoost.

- *Features for the stochastic model.* This was a topic to which we dedicated a significant amount of time. Emboldened by the advantage of flexible horizon predictions, we wanted to adapt the model so that it can depend on more features, not just the mortgage interest rate. Unfortunately, the optimization equations become too complex to handle, and we could not arrive at a closed-form solution to it. Alas, one might still be able to find an approximate minimum point solution to this augmented model, yet we did not explore this possibility further.

A Appendix

A.1 Data

A.1.1 Data providers description

21st Real Estate GmbH
The dataset provided by 21st comprises 149 German cities (with each having $> 50,000$ inhabitants) in the form of tiles, where each tile is a region of 200 m × 200 m within a city. From the universe of 21st database, 5% of all tiles in a specific city (but at least 100 tiles) have been randomly sampled, making a total of 20,000 tiles available to us. The exact geographical location and coordinates of each tile have been encrypted using id numbers and are therefore unknown for us. This pre-sampling procedure and encryption of the exact location are due to the data protection policy of 21st. Obtained variables include:

- Target variables: **sale and rent prices** at tile level from 2011 to 2018 on a quarterly basis.
- Micro-location factors: scores for factors for each tile including **connectivity**, **nature**, **urbanity** and **income** are part of 21st data. Note that the scores are not comparable across cities and due to the difficulty of incorporating these to our models, we dropped these factors from our modeling scope.
- Macro-location factors: 17 macro-location variables including **population, migration balance, number of births** and **unemployment rate** were provided for each city from 2011 to 2017 on a yearly basis.

In terms of micro-location factors, income scores experienced a small variance over time in the 21st dataset. This is due to the fact that raw income data are generally only available down to a ZIP code level which entails many tiles. As scores are calculated as percentiles, and a ZIP code's income percentile, relative to the rest of the city, is barely changing throughout time, these scores are rather 'sticky'.

Das Statistische Bundesamt (Destatis)
Destatis is the Federal Statistical Office of Germany, which has the task to collect and to provide statistical information concerning economy, society and environment in Germany. Obtained variables:

- Macro-location factors: **unemployment rates (female, male and entire population), household income** and **population** from 2007 to 2018 on a yearly basis for each German state.
- Economic factors: **Germany's gross domestic product, inflation** (annual change of consumer price index), **gross national income** and **gross value added** for agriculture, industry and services of Germany from 2009 to 2018 on a yearly basis.

Deutsche Bundesbank

Deutsche Bundesbank is the central bank of Germany which offers information on market-relevant interest rates. Obtained factors:

- Economic factors: **mortgage rate** from 2003 to 2019 on a quarterly basis.

The Organisation for Economic Cooperation and Development (OECD)

OECD is an intergovernmental economic organisation that works on establishing international norms and finding evidence-based solutions to a range of social, economic and environmental challenges. It publishes books, reports, statistics, working papers and reference materials through OECD iLibrary. Obtained variables:

- Economic factors: **gross domestic product, unemployment rate** and **Germany's current account** from 2006 to 2019 on a quarterly basis.

INKAR (Indikatoren zur Raum- und Stadtentwicklung)

Data and maps on living conditions in Germany and Europe have been published by The Federal Institute for Building, Urban and Spatial Research (BBSR) online at INKAR online. INKAR comprises more than 600 statistical parameters on almost all socially important topics. Obtained variables:

- Macro-location factors: 18 variables such as **life expectancy, population density, living space per resident, inflow and outflow of people, average age of the population** and **number of university students per 1000 residents** from 2007 to 2018 on a yearly basis for each city.

A comparison between the income data from Oxford Economics and INKAR showed that the data only differed by a constant. The INKAR data has the advantage that, like our target variable, it is on city level, but it is on yearly level instead of quarterly level. We looked through all the statistical parameters available and choose the ones that seem to have considerable influence on the rent or sale prices according to the literature.

Empirica

Empirica AG is an independent economic and social science research and consulting institute. Empirica regio GmbH is a database specialised in the processing, analysis and provision of framework data for the real estate industry. Empirica has provided us with time series of vacancy rates. Some analyses of the vacancy data can be found in the CBRE-empirica-Leerstandsindex. Obtained variables:

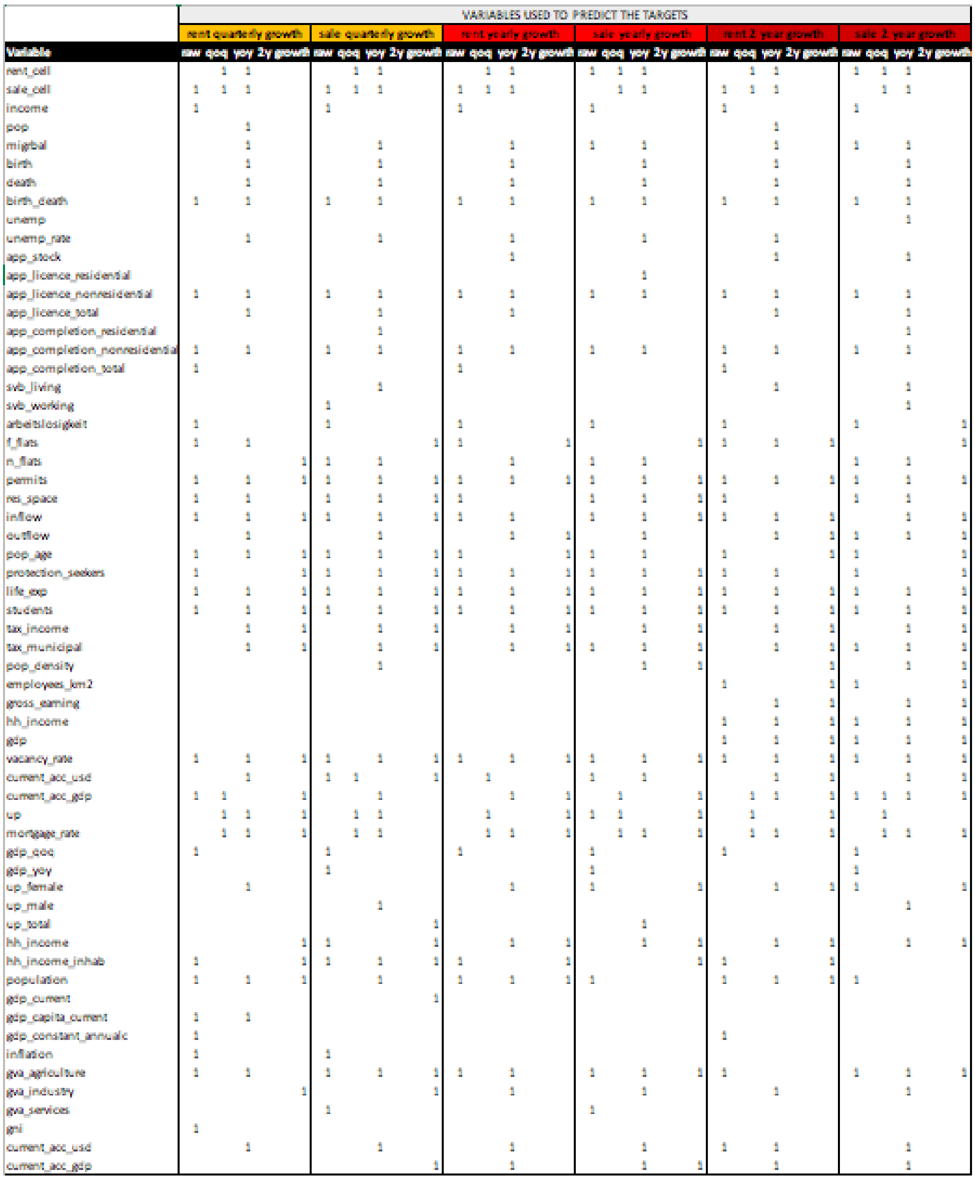
- Macro-location factors: **vacancy rate** for all German independent cities and counties from 2005 to 2017 on a yearly basis.

A.1.2 Variable description

Variable	Information from Original Sources						Transformation (Section 2.3)
Name	Description	Spacial Level	Time Frame	Time Unit	Source	Sample Value	Group
rent_cell	Rent per m^2 of comparable residential buildings (median of tile-level rents)	City(Originally tile)	2011Q1 - 2018Q3	Quarter	21st	5.818573178	0
sale_cell	House price per m^2 of comparable residential buildings (median of tile-level house prices)	City(Originally tile)	2011Q1 - 2018Q3	Quarter	21st	1368.555929	0
income	Average annual income in 1000 EURO	City	2012 - 2018	Year	21st	65.269461	1
pop	Population of the city	City	2011 - 2017	Year	21st	82801	2
migrbal	Migration balance	City	2011 - 2017	Year	21st	-5751	2
birth	Number of births	City	2011 - 2017	Year	21st	763	2
death	Number of deaths	City	2011 - 2017	Year	21st	952	2
birth_death	”birth” divided by ”death”	City	2011 - 2017	Year	21st	0,801470588	2
unemp	Number of unemployed people	City	2011 - 2017	Year	21st	5151	2
unemp_rate	Unemployment rate	City	2011 - 2017	Year	21st	11,8	2
app_stock	Housing stock Number of apartments (Wohungsbestand)	City	2011 - 2017	Year	21st	48109	2
app_licence_residential	Number of construction permits for residential buildings	City	2011 - 2017	Year	21st	320	2
app_licence_nonresidential	Number of construction permits for non-residential buildings	City	2011 - 2017	Year	21st	6	2
app_licence_total	Number of construction permits in total	City	2011 - 2017	Year	21st	326	2
app_completion_residential	Number of construction completions for residential buildings	City	2011 - 2017	Year	21st	165	2
app_completion_nonresidential	Number of construction completions for non-residential buildings	City	2011 - 2017	Year	21st	1	2
app_completion_total	Number of construction completions in total	City	2011 - 2017	Year	21st	166	2
svb_living	Number of regular employees paying social insurance (Sozialversicherungspflichtige Beschäftigte) living in the given area	City	2011 - 2017	Year	21st	27016	2
svb_working	Number of regular employees paying social insurance living outside the given area	City	2011 - 2017	Year	21st	39028	2
arbeitslosigkeit	Number of employed persons per squared kilometer	City	2009 - 2017	Year	INKAR	93.2	3
f_flats	Ready-made apartments per 1000 portfolio apartments (Fertiggestellte Wohnungen je 1.000 Wohnungen des Bestandes)	City	2009 - 2017	Year	INKAR	3.7	3
n_flats	Ready-made apartments in new buildings per 1000 residents	City	2009 - 2017	Year	INKAR	2.0	3
permits	Number of construction permits for new residential buildings per 1000 residents	City	2009 - 2017	Year	INKAR	3.9	3
res_space	Living space per resident in square meters	City	2009 - 2017	Year	INKAR	45.6	3
inlfow	Number of people moved in per 1000 residents	City	2009 - 2017	Year	INKAR	87.377.9	3
outflow	Number of people moved out per 1000 residents	City	2009 - 2017	Year	INKAR	77.9	3
pop_age	Average age of residents	City	2009 - 2017	Year	INKAR	42.6	3
protection_seekers	Percentage of foreigners seeking governmental protection (e.g. refugees, asylum seekers)	City	2009 - 2017	Year	INKAR	0.5	3
life_exp	Average life expectancy of newborns	City	2009 - 2017	Year	INKAR	79.02	3
students	Number of university students per 1000 residents	City	2009 - 2017	Year	INKAR	100.8	3
tax_income	Income tax per resident in euros	City	2009 - 2017	Year	INKAR	284.7	3
tax_municipal	Taxable capacity of the municipality per resident in euros	City	2009 - 2017	Year	INKAR	599.3	3
pop_density	Number of residents per square kilometers	City	2009 - 2017	Year	INKAR	1459.0	3
employees_km2	Number of employed persons per squared kilometer	City	2009 - 2016	Year	INKAR	988.3	4
gross_earning	Monthly gross income per employed person in euros	City	2009 - 2016	Year	INKAR	2126.0	4
hh_income	Average house hold income in euros per resident	City	2009 - 2016	Year	INKAR	1501.0	4
gdp	GDP per resident in 1000 EURO	City	2009 - 2016	Year	INKAR	38.4	4
vacancy_rate	Percentage of all available units in a rental property that are vacant or unoccupied	City	2009 - 2017	Year	Empirica	3.64	3
current_acc_usd	Germany's current account balance in US Dollar (quarterly data)	Country	2009 - 2018	Quarter	OECD	38937.57	5
current_acc_gdp	Germany's current account balance as a % of GDP (quarterly data)	Country	2009 - 2018	Quarter	OECD	5.561069	5
up	Number of unemployed people as a percentage of the labor force	Country	2009 - 2018	Quarter	OECD	6.184005	5
mortgage_rate	Effective interest rates of German banks / New business / Housing loans to households	Country	2009 - 2018	Quarter	Bundesbank	3.97	5
gdp_qoq	GDP growth in percentage compared to the previous quarter	Country	2009 - 2018	Quarter	OECD	1.924936	6
gdp_yoy	GDP growth in percentage compared to the same quarter of the previous year	Country	2009 - 2018	Quarter	OECD	5.934946	6
up_female	Unemployment as percent of civilian labour force - Female	State	2009 - 2018	Year	Bundesamt	6.9	9
up_male	Unemployment as percent of civilian labour force - Male	State	2009 - 2018	Year	Bundesamt	7.5	9
up_total	Unemployment as percent of civilian labour force	State	2009 - 2018	Year	Bundesamt	7.2	9
hh_income	Disposable income of households	State	2009 - 2017	Year	Bundesamt	214929.0	3
hh_income_inhab	Disposable income of households per inhabitant	State	2009 - 2017	Year	Bundesamt	20122.0	3
population	Population	State	2009 - 2018	Year	Bundesamt	2802266.0	9
gdp_current	Gross domestic product (GDP), current prices	Country	2009 - 2018	Year	Bundesamt	3761.1	7
gdp_capita_current	GDP per capita, current prices	Country	2009 - 2018	Year	Bundesamt	46853.0	7
gdp_constant_annual	GDP, constant prices (annual change)	Country	2009 - 2018	Year	Bundesamt	3.7	8
inflation	Inflation (annual change of CPI)	Country	2009 - 2018	Year	Bundesamt	2.5	8
gva_agriculture	Gross value added: Agriculture (share of GDP)	Country	2009 - 2018	Year	Bundesamt	0.7	7
gva_industry	Gross value added: Industry (share of GDP)	Country	2009 - 2018	Year	Bundesamt	27.5	7
gva_services	Gross value added: Services (share of GDP)	Country	2009 - 2018	Year	Bundesamt	61.6	7
gni	Gross national income per capita, Atlas method	Country	2009 - 2018	Year	Bundesamt	47360.0	7
current_acc_usd	Germany's current account balance in US Dollar (yearly data)	Country	2009 - 2017	Year	OECD	198311.9	3
current_acc_gdp	Germany's current account balance as a % of GDP (yearly data)	Country	2009 - 2017	Year	OECD	5.835584	3

Table 5: Description of the all the variables collected as described in Section 2.2.

A.1.3 Selected features



	VARIABLES USED TO PREDICT THE TARGETS																							
	rent quarterly growth				sale quarterly growth				rent yearly growth				sale yearly growth				rent 2 year growth				sale 2 year growth			
Variable	**raw**	**qoq**	**yoy**	**2y growth**	**raw**	**qoq**	**yoy**	**2y growth**	**raw**	**qoq**	**yoy**	**2y growth**	**raw**	**qoq**	**yoy**	**2y growth**	**raw**	**qoq**	**yoy**	**2y growth**	**raw**	**qoq**	**yoy**	**2y growth**
rent_cell		1	1			1	1			1	1		1	1	1			1	1		1	1	1	
sale_cell	1	1	1		1	1	1		1	1	1			1	1		1	1	1			1	1	
income	1				1				1				1				1				1			
pop			1																1					
migbal			1				1				1		1		1				1		1		1	
birth			1				1				1				1				1				1	
death			1				1				1				1				1				1	
birth_death	1		1		1		1		1		1		1		1		1		1		1		1	
unemp																							1	
unemp_rate			1				1				1				1				1					
app_stock											1								1				1	
app_licence_residential															1									
app_licence_nonresidential	1		1		1		1		1		1		1		1		1		1		1		1	
app_licence_total			1				1				1								1				1	
app_completion_residential							1																1	
app_completion_nonresidential	1		1		1		1		1		1		1		1		1		1		1		1	
app_completion_total	1								1								1							
svb_living							1												1				1	
svb_working					1																		1	
arbeitslosigkeit	1				1				1				1				1				1			1
f_flats	1		1					1	1			1				1	1		1	1				1
n_flats				1	1		1				1		1		1						1		1	
permits	1		1	1	1		1	1	1		1	1	1		1	1	1		1	1	1		1	1
res_space	1		1		1		1	1	1				1		1	1	1				1		1	
inflow	1		1	1	1		1	1	1		1		1		1	1	1		1	1			1	1
outflow			1				1				1	1			1				1	1	1		1	1
pop_age	1		1	1	1		1	1	1			1	1		1		1			1	1			1
protection_seekers	1			1	1		1	1	1		1	1	1		1	1	1		1		1			1
life_exp	1		1	1	1		1	1	1		1	1	1		1	1	1		1	1	1		1	1
students	1		1	1	1		1	1	1		1	1	1		1	1	1		1	1	1		1	1
tax_income			1	1			1	1			1	1			1	1			1	1			1	1
tax_municipal			1	1			1	1			1	1	1		1	1			1	1	1		1	1
pop_density							1								1	1				1			1	1
employees_km2																	1			1	1			1
gross_earning																			1	1			1	1
hh_income																	1		1	1	1		1	1
gdp																	1		1	1	1		1	1
vacancy_rate	1		1	1	1		1	1	1		1	1	1		1	1	1		1	1	1		1	1
current_acc_usd			1		1	1		1		1			1		1				1	1			1	1
current_acc_gdp	1	1		1			1				1	1		1		1		1	1	1	1	1	1	1
up		1	1	1		1	1			1		1	1	1		1		1		1		1		
mortgage_rate		1	1	1		1	1			1	1	1		1	1	1		1	1	1		1	1	1
gdp_qoq	1				1				1				1				1				1			
gdp_yoy					1								1								1			
up_female			1								1		1			1			1	1	1			1
up_male							1																1	
up_total								1							1									
hh_income				1	1			1			1	1			1	1			1	1			1	1
hh_income_inhab	1			1	1		1	1	1			1				1	1			1				
population	1		1	1			1		1		1	1	1				1		1	1	1			
gdp_current								1																
gdp_capita_current	1		1																					
gdp_constant_annualc	1																1							
inflation	1				1																			
gva_agriculture	1		1		1		1	1	1		1		1		1	1	1				1		1	1
gva_industry				1				1			1				1				1				1	
gva_services					1								1											
gni	1																							
current_acc_usd			1				1				1				1		1		1				1	
current_acc_gdp								1			1				1	1			1				1	

Figure 17: Features selected for each target. The 6 main columns on the right represent the targets, the number 1 indicates which transformation of the variable (on the left) was considered as feature for the corresponding target after the correlation analysis described in Section 2.4.

A.1.4 Merging map

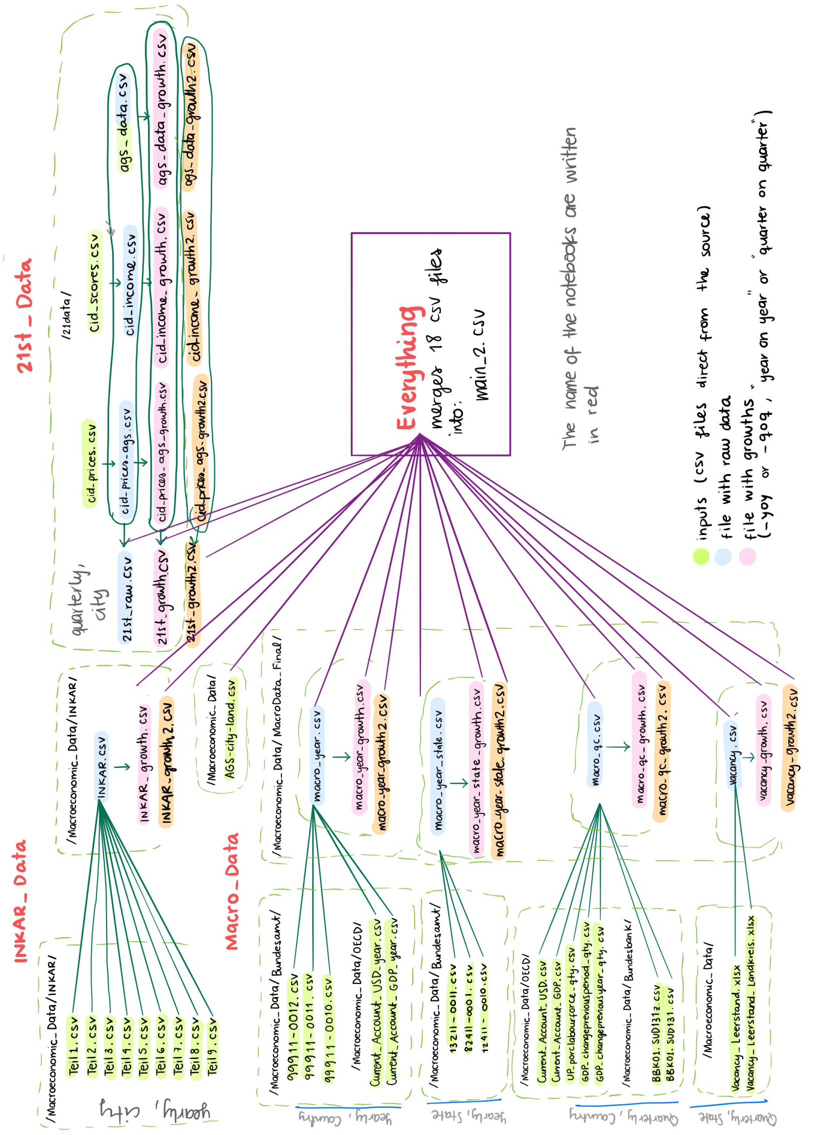


Figure 18: Merging map. This map represents all the files and notebooks used to create the final data set and their dependencies. The names of the notebooks are on red, the files as we obtained them from their source are highlighted in green, the files in blue are either mergings or reformats of the 'green' files, while the red and orange files represent qoq/yoy growths and 2year growths, respectively. The lines are a sign of dependencies between files and notebooks.

A.2 Statistical measures

R^2

In statistics, the coefficient of determination, denoted R^2, is the proportion of the variance in the dependent variable that is predictable from the independent variable(s). An R^2 of 1 indicates that the regression predictions perfectly fit the data.

Values of R^2 outside the range 0 to 1 can occur when the model fits the data worse than a horizontal hyperplane. This would occur when the wrong model was chosen, or nonsensical constraints were applied by mistake.

Its main purpose is either the prediction of future outcomes or the testing of hypotheses, on the basis of other related information. It provides a measure of how well observed outcomes are replicated by the model, based on the proportion of total variation of outcomes explained by the model.

In [DS98], R^2 is defined as follows:

Definition 1 *Let the data set have n values marked $y_1, ..., y_n$ (collectively known as y_i, $i = 1, ..., n$), each associated with a fitted (or modeled, or predicted) value $\hat{y}_1, ..., \hat{y}_n$. And define the residuals as $e_i = y_i - \hat{y}_i$.*

If $\bar{y}$ is the mean of the observed data: $\bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i$, then the variability of the data set can be measured using three sums of squares formulas:

- *The total sum of squares (proportional to the variance of the data):*

$$SS_{tot} = \sum_{i=1}^{n}(y_i - \bar{y})^2,$$

- *The regression sum of squares, also called the explained sum of squares:*

$$SS_{reg} = \sum_{i=1}^{n}(\hat{y}_i - \bar{y})^2,$$

- *The sum of squares of residuals, also called the residual sum of squares:*

$$SS_{res} = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2 = \sum_i e_i^2.$$

And the coefficient of determination is defined as

$$R^2 = 1 - \frac{SS_{res}}{SS_{tot}}.$$

MSE

The mean square error (MSE) is a frequently used measure of the differences between values (sample or population values) predicted by a model or an estimator and the values observed. MSE represents the second sample moment of the differences between predicted values and observed values or the quadratic mean of these differences. These deviations are called residuals when the calculations are performed over the data sample that was used for estimation and are called errors (or prediction errors) when computed out-of-sample.

Definition 2 *The MSE of an estimator $\hat{\theta}$ with respect to an estimated parameter θ is defined in e.g. [DS98] as*

$$MSE(\hat{\theta}) = \mathbb{E}((\hat{\theta} - \theta)^2).$$

For the data set $y_1, ..., y_n$, the MSE of predicted values $\hat{y}_n$ of a regression's dependent variable y_n, is computed as follows:

$$MSE = \frac{\sum_{i=1}^{n} (\hat{y}_n - y_n)^2}{n}.$$

Pearson's correlation coefficient

In statistics, the Pearson's correlation coefficient $\rho_{X,Y}$, is a measure of the linear correlation between two variables X and Y. According to the Cauchy-Schwarz inequality it has a value between +1 and -1, where 1 is total positive linear correlation, 0 is no linear correlation, and -1 is total negative linear correlation.

Given a pair of random variables X and Y, $\rho_{X,Y}$ is defined as

$$\rho_{X,Y} = \frac{cov(X, Y)}{\sigma_X \sigma_Y},$$

where $cov(X, Y)$ is the covariance of X and Y, and σ_X, σ_Y are the standard deviations of X, Y, respectively.

When applied to a sample, Pearson's correlation coefficient is commonly represented by r_{xy} and may be referred to as the sample correlation coefficient or the sample Pearson's correlation coefficient. We can obtain a formula for r_{xy} by substituting estimates of the covariances and variances based on a sample into the formula above. Given paired data $\{(x_1, y_1), \ldots, (x_n, y_n)\}$ consisting of n pairs, r_{xy} is defined as:

$$r_{xy} = \frac{\sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n} (x_i - \bar{x})^2} \sqrt{\sum_{i=1}^{n} (y_i - \bar{y})^2}},$$

where n is sample size x_i, y_i are the individual sample points indexed with i: $\bar{x} = \frac{1}{n} \sum_{i=1}^{n} x_i$ (the sample mean); and analogously for $\bar{y}$.

Rearranging gives us the following formula for r_{xy}:

$$r_{xy} = \frac{n \sum x_i y_i - \sum x_i \sum y_i}{\sqrt{n \sum x_i^2 - (\sum x_i)^2} \sqrt{n \sum y_i^2 - (\sum y_i)^2}},$$

where n, x_i, y_i are defined as above.

A.3 Feature regression

A.3.1 Coefficients of targets associated with rent prices

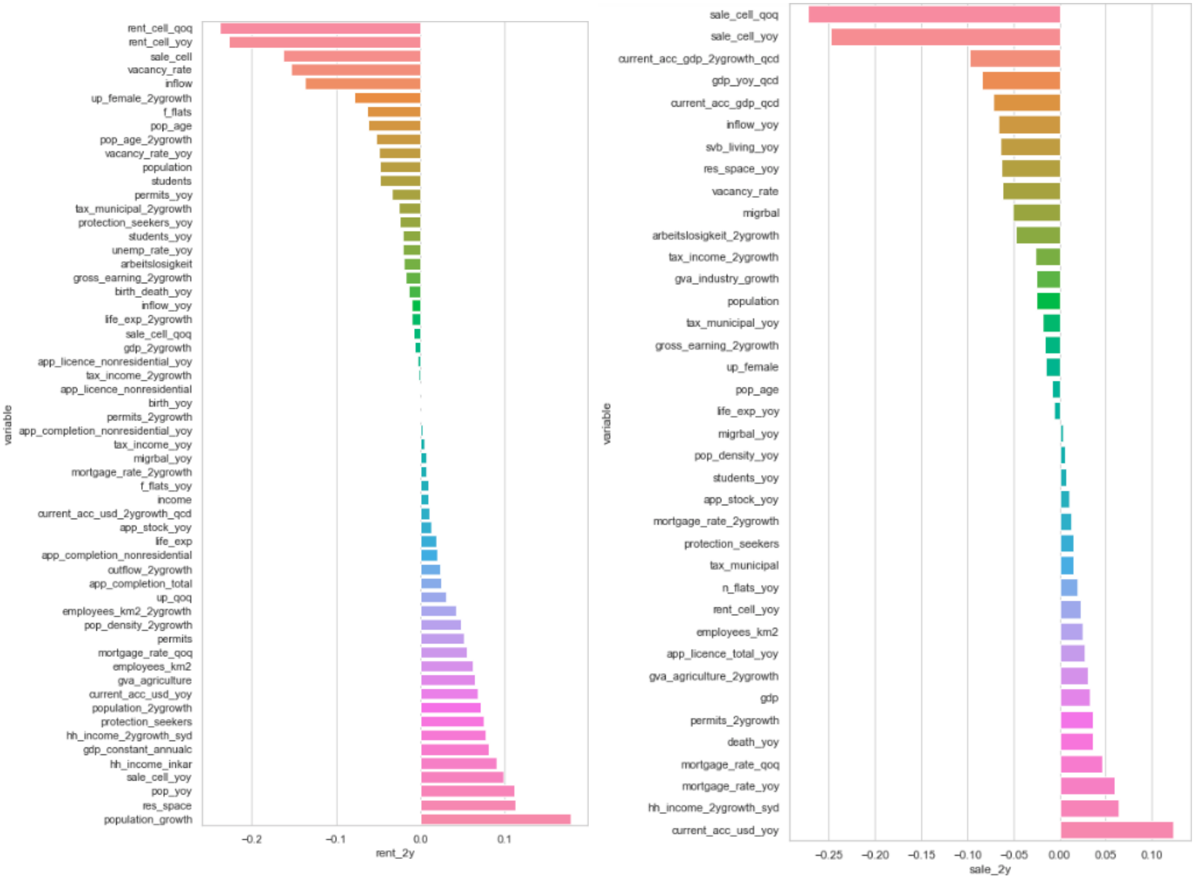
VARIABLE	rent_qoq	rent_yoy	rent_2y
intercept	-2.94E-02	-2.72E-01	-5.68E-01
app_completion_nonresidential	1.75E-05	1.16E-04	1.17E-04
app_completion_nonresidential_yoy	-8.91E-07	1.02E-05	-1.50E-05
app_completion_total	2.48E-07	1.90E-06	3.02E-06
app_licence_nonresidential	1.37E-05	-1.08E-05	-5.25E-06
app_licence_nonresidential_yoy	-5.13E-05	-5.74E-05	-1.74E-04
app_licence_total_yoy		-4.86E-05	-1.84E-04
arbeitslosigkeit	-4.09E-05	-8.57E-05	-4.18E-05
birth_death		6.17E-04	
current_acc_gdp_2ygrowth_qcd		1.55E-03	
current_acc_gdp_cyd		9.70E-03	
current_acc_usd_cyd			1.08E-06
current_acc_usd_qoq		-3.42E-04	
employees_km2			7.50E-06
f_flats		-7.17E-04	-1.78E-03
f_flats_2ygrowth		3.43E-04	
f_flats_yoy			1.17E-03
gdp_capita_current	-1.13E-06		
gdp_constant_annualc			-8.76E-03
gdp_qoq		-1.55E-03	4.76E-03
gni	1.28E-06		
hh_income_inhab	1.75E-06	4.43E-06	6.22E-06
hh_income_inkar			1.89E-05
income	1.31E-05	2.27E-05	9.98E-05
inflow	-1.24E-05	-8.08E-05	-1.62E-04
life_exp		2.84E-03	4.50E-03
mortgage_rate_qoq		8.19E-03	
mortgage_rate_yoy		-2.88E-02	9.04E-03
n_flats_yoy		-1.01E-03	
permits		1.24E-03	3.98E-03
permits_2ygrowth		-1.49E-03	
permits_yoy		-9.00E-04	-4.70E-03
pop_age		-1.74E-03	-3.93E-03
population	-2.83E-10	-7.92E-10	-1.23E-09
protection_seekers		2.38E-03	5.46E-03
rent_cell_qoq		-2.74E-01	-2.33E-01
rent_cell_yoy		-2.73E-01	-1.80E-01
res_space		5.75E-04	1.55E-03
sale_cell	-1.29E-06	-7.97E-06	-2.05E-05
sale_cell_yoy		3.42E-02	3.74E-02
students	4.00E-09	-2.53E-05	-5.02E-05
students_2ygrowth		2.02E-03	
vacancy_rate	-4.77E-05	-2.08E-03	-3.50E-03
vacancy_rate_2ygrowth		-1.49E-02	-2.07E-02

A.3.2 Coefficients of targets associated with sale prices

variable	sale_qoq	sale_yoy	sale_2y
intercept	-7.69E-01	3.71E-01	5.17E-01
app_completion_nonresidential	1.74E-05	7.71E-05	3.16E-05
app_completion_nonresidential_yoy	4.30E-05	4.46E-04	-7.56E-05
app_completion_residential_yoy	-4.08E-04		-2.43E-04
app_licence_nonresidential	-4.21E-05	-1.98E-04	-2.42E-04
app_licence_nonresidential_yoy		-1.19E-04	-1.85E-04
app_licence_residential_yoy		2.78E-03	
app_licence_total_yoy			8.02E-03
arbeitslosigkeit	-1.30E-04	-2.52E-05	3.19E-04
arbeitslosigkeit_2ygrowth			-4.45E-02
current_acc_gdp_2ygrowth_qcd			-3.85E-03
current_acc_gdp_qcd			-2.48E-02
current_acc_usd_qcd	1.67E-07	-1.98E-06	
current_acc_usd_yoy			1.35E-01
employees_km2			1.62E-05
gdp			4.90E-04
gdp_qoq	1.64E-02	1.86E-02	
gdp_yoy_qcd	7.79E-04	4.79E-03	-2.10E-02
gva_agriculture_2ygrowth		3.08E-02	
gva_agriculture_growth		2.09E-03	
hh_income_inhab	8.00E-06		
hh_income_inkar			1.43E-05
hh_income_syd	-3.59E-08		
income	1.11E-04	1.67E-05	-1.18E-04
inflation	-3.14E-04		
inflow	9.95E-05	-2.16E-05	
life_exp	9.34E-03		4.06E-04
migrbal		-6.25E-07	-6.70E-07
mortgage_rate_2ygrowth			2.12E-02
mortgage_rate_qoq			4.13E-03
mortgage_rate_yoy		-2.66E-02	9.81E-02
n_flats	1.15E-03	-5.03E-03	-7.70E-03
n_flats_yoy		1.28E-03	9.33E-03
outflow			-2.46E-04
permits	3.13E-03	4.69E-03	4.96E-03
permits_2ygrowth		2.07E-03	7.96E-03
permits_yoy			-4.13E-03
pop_age		-3.90E-03	-6.03E-03
population		-1.16E-09	-1.82E-09
protection_seekers			8.24E-03
rent_cell			1.77E-03
res_space	-1.04E-03	3.72E-04	1.76E-03
sale_cell	-4.91E-05		
sale_cell_qoq	-2.09E-01	-2.89E-01	-3.25E-01
sale_cell_yoy	-1.42E-01	-2.14E-01	-2.75E-01
students	-1.43E-05	-2.19E-05	-4.21E-05
svb_working	5.75E-08		
tax_municipal		1.64E-05	-3.32E-06
up_female		-2.63E-03	-4.42E-03
vacancy_rate	-2.25E-03	-2.32E-03	-6.70E-03

A.3.3 Coefficient plots

While comparing the coefficients of the rent and sale prices targets, we can notice that some of the main features of the models are transformations of rent and sale prices themselves. Even when the time window may be reduced by including the transformations, it was good to include them. Also, the fact that these factors have a bigger impact that most of the others explain the simmetry in the Montecarlo simulations shown in Figure 7.



(a) Coefficients of `2y_rent_cell`.

(b) Coefficients of `2y_sale_cell`.

Figure 19: Coefficients of 2year-on-2year rent and sale prices from the Feature Regression model. To have a better interpretation of the importance of the features, we normalized them before fitting the model.

A.4 Mathematical Derivation of the Stochastic Model Calibration Steps

Recap the model: we first discretize the time into a finite set of intervals $t_1, ..., t_N$, where $t_i < t_{i+1}$ for all $i \in [0, N]$, with $t_0 = 0$ and $t_N = T$. Using a sufficiently large N and an evenly spaced time-lattice $t_i = \frac{iT}{N}$, we approximate the HPI returns and mortgage rate. Starting from initial values observed from the real data, the HPI returns and mortgage rate are determined as follows:

$$h_{t+\Delta t} = h_t + \lambda(\mu_h - r_t)h_t\Delta t + \sigma_h h_t(Z_{t+\Delta t} - Z_t), \tag{6}$$

$$r_{t+\Delta t} = r_t + \kappa(\mu_r - r_t)\Delta t + \sigma_r(W_{t+\Delta t} - W_t) \tag{7}$$

where $\Delta t = t_{i+1} - t_i$. To calibrate the parameters, we minimize the variance of the noise variables via the least squares method:

$$(\hat{\lambda}, \hat{\mu}_h) = argmin \sum_{i=1}^{N-1} \left(\left(\frac{h_{i+1} - h_i}{h_i} \right) - \lambda(\mu_h - r_i)\Delta t \right)^2, \tag{8}$$

$$(\hat{\kappa}, \hat{\mu}_r) = argmin \sum_{i=1}^{N-1} (r_{i+1} - r_i - \kappa(\mu_r - r_i)\Delta t)^2 \tag{9}$$

In the original paper, there is no explanation on how we reach these optimal points, so we derive a method ourselves. Since these equations are quadratic, we can obtain analytic formulations of the solution points. To do this, we take the partial derivatives with respect to λ and μ_h, and set them to 0. Because the functions are quadratic and therefore convex, this will yield the global minimum point.

$$\frac{\partial}{\partial \lambda} = 2 \sum_{i=1}^{N-1} \left(\lambda(\mu_h - r_i)^2 \Delta t^2 - (\mu_h - r_i)\frac{h_{i+1} - h_i}{h_i}\Delta t \right) = 0 \tag{10}$$

$$\frac{\partial}{\partial \mu_h} = 2 \sum_{i=1}^{N-1} \left(\lambda^2 \mu_h \Delta t^2 - \lambda^2 r_i - \lambda \frac{h_{i+1} - h_i}{h_i}\Delta t \right) = 0 \tag{11}$$

Through algebraic manipulations, the first equation is equivalent to:

$$\lambda = \frac{\sum_{i=1}^{N-1}(\mu_h - r_i)\frac{h_{i+1}-h_i}{h_i}}{\Delta t \sum_{i=1}^{N-1}(\mu_h - r_i)^2}$$

and the second one is equivalent to:

$$\lambda = \frac{\sum_{i=1}^{N-1} \frac{h_{i+1}-h_i}{h_i}}{\Delta t \sum_{i=1}^{N-1}(\mu_h - r_i)}$$

Both of the fractions are equal to λ. Therefore, now set the fractions to be equal, do the cross-product, and expand further to get a relation for μ_h:

$$\Delta t \sum_{i=1}^{N-1}(\mu_h - r_i)^2 \sum_{i=1}^{N-1} \frac{h_{i+1} - h_i}{h_i} = \Delta t \sum_{i=1}^{N-1}(\mu_h - r_i) \sum_{i=1}^{N-1}(\mu_h - r_i)\frac{h_{i+1} - h_i}{h_i}$$

$$
\begin{multlined}
\mu_h^2 \left((N-1) \sum_{i=1}^{N-1} \frac{h_{i+1}-h_i}{h_i} \right) - 2\mu_h \left(\sum_{i=1}^{N-1} r_i \sum_{i=1}^{N-1} \frac{h_{i+1}-h_i}{h_i} \right) + \\
\left(\sum_{i=1}^{N-1} r_i^2 \sum_{i=1}^{N-1} \frac{h_{i+1}-h_i}{h_i} \right) = \mu_h^2 \left((N-1) \sum_{i=1}^{N-1} \frac{h_{i+1}-h_i}{h_i} \right) - \\
\mu_h \left(n \sum_{i=1}^{N-1} r_i \frac{h_{i+1}-h_i}{h_i} + \sum_{i=1}^{N-1} r_i \sum_{i=1}^{N-1} \frac{h_{i+1}-h_i}{h_i} \right) + \sum_{i=1}^{N-1} r_i \sum_{i=1}^{N-1} r_i \frac{h_{i+1}-h_i}{h_i}
\end{multlined}
$$

$$
\begin{multlined}
\sum_{i=1}^{N-1} r_i^2 \sum_{i=1}^{N-1} \frac{h_{i+1}-h_i}{h_i} - \sum_{i=1}^{N-1} r_i \sum_{i=1}^{N-1} r_i \frac{h_{i+1}-h_i}{h_i} = \\
\mu_h \left(-(N-1) \sum_{i=1}^{N-1} r_i \frac{h_{i+1}-h_i}{h_i} + \sum_{i=1}^{N-1} r_i \sum_{i=1}^{N-1} \frac{h_{i+1}-h_i}{h_i} \right)
\end{multlined}
$$

$$
\hat{\mu}_h = \frac{\sum_{i=1}^{N-1} r_i^2 \sum_{i=1}^{N-1} \frac{h_{i+1}-h_i}{h_i} - \sum_{i=1}^{N-1} r_i \sum_{i=1}^{N-1} r_i \frac{h_{i+1}-h_i}{h_i}}{\sum_{i=1}^{N-1} r_i \sum_{i=1}^{N-1} \frac{h_{i+1}-h_i}{h_i} - (N-1) \sum_{i=1}^{N-1} r_i \frac{h_{i+1}-h_i}{h_i}}
$$

We have a solution for μ_h! Plug this back in the relationship for λ, and notice how the whole formula simplifies reasonably nice:

$$
\lambda = \frac{\sum_{i=1}^{N-1} \frac{h_{i+1}-h_i}{h_i}}{\Delta t \sum_{i=1}^{N-1} (\mu_h - r_i)} = \frac{1}{\Delta t} \frac{\sum_{i=1}^{N-1} \frac{h_{i+1}-h_i}{h_i}}{(N-1)\mu_h - \sum_{i=1}^{N-1} r_i}
$$

$$
= \frac{\frac{1}{\Delta t} \sum_{i=1}^{N-1} \frac{h_{i+1}-h_i}{h_i}}{\frac{(N-1) \sum_{i=1}^{N-1} r_i^2 \sum_{i=1}^{N-1} \frac{h_{i+1}-h_i}{h_i} - (N-1) \sum_{i=1}^{N-1} r_i \sum_{i=1}^{N-1} r_i \frac{h_{i+1}-h_i}{h_i} + \left(\sum_{i=1}^{N-1} r_i\right)^2 \sum_{i=1}^{N-1} \frac{h_{i+1}-h_i}{h_i} - (N-1) \sum_{i=1}^{N-1} r_i \sum_{i=1}^{N-1} r_i \frac{h_{i+1}-h_i}{h_i}}{\sum_{i=1}^{N-1} r_i \sum_{i=1}^{N-1} \frac{h_{i+1}-h_i}{h_i} - (N-1) \sum_{i=1}^{N-1} r_i \frac{h_{i+1}-h_i}{h_i}}}
$$

$$
= \frac{1}{\Delta t} \frac{\sum_{i=1}^{N-1} r_i \sum_{i=1}^{N-1} \frac{h_{i+1}-h_i}{h_i} - (N-1) \sum_{i=1}^{N-1} r_i \frac{h_{i+1}-h_i}{h_i}}{(N-1) \sum_{i=1}^{N-1} r_i^2 - \left(\sum_{i=1}^{N-1} r_i \right)^2}
$$

The final solutions will therefore be, where $n = N-1$:

$$
\hat{\lambda} = \frac{1}{\Delta t} \frac{\sum_i r_{t_i} \sum_i \frac{h_{t_{i+1}} - h_{t_i}}{h_{t_i}} - n \sum_i r_{t_i} \frac{h_{t_{i+1}} - h_{t_i}}{h_{t_i}}}{n \sum_i r_{t_i}^2 - \left(\sum_i r_{t_i} \right)^2} \tag{12}
$$

$$
\hat{\mu}_h = \frac{\sum_i r_{t_i}^2 \sum_i \frac{h_{t_{i+1}} - h_{t_i}}{h_{t_i}} - \sum_i r_t \sum_i r_t \frac{h_{t+1} - h_t}{h_t}}{\sum_i r_{t_i} \sum_i \frac{h_{t_{i+1}} - h_{t_i}}{h_{t_i}} - n \sum_i r_{t_i} \frac{t_{i+1} - h_{t_i}}{h_{t_i}}} \tag{13}
$$

All the sums range along the time increments for $i = \{1, 2, \dots, T\}$. Similarly, for the other equation, by symmetry we can deduce the solutions:

$$\hat{\kappa} = \frac{1}{\Delta t} \frac{\sum_i r_{t_i} \sum_i (r_{t_{i+1}} - r_{t_i}) - n \sum_i r_{t_i}(r_{t_{i+1}} - r_{t_i})}{n \sum_i r_{t_i}^2 - (\sum_i t_{it})^2} \tag{14}$$

$$\hat{\mu}_r = \frac{\sum_i r_{t_i}^2 \sum_i (r_{t_{i+1}} - r_{t_i}) - \sum_i r_{t_i} \sum_i r_{t_i}(r_{t_{i+1}} - r_{t_i})}{\sum_i r_{t_i} \sum_i (r_{t_{i+1}} - r_{t_i}) - n \sum_i r_{t_i}(r_{t_{i+1}} - r_{t_i})} \tag{15}$$

Note. This particular interpretation of the model refers to fitting for a single price series h_t (indexed by $t_i, 1 \leq i \leq T$). One can easily adapt it to fit for N multiple price series $h_{jt_i}, 1 \leq j \leq N, 1 \leq i \leq T$, by summing up over both j (the series index) and t_i (the time index) within the solution equations for λ and μ_h. (indexation is similar to the VAR models, as seen in Section 3.8).

A.5 Inferring Market Rents from House Prices and its Application to the Stochastic Model

One more desire was to attempt to extrapolate the rent prices from the house prices, and use this model to generate a continuous prediction for the rent prices. Looking at a lagged cross-correlation analysis between `sale_cell` and `rent_cell` described in Figure 20, we generate a simple regressive model with `rent_cell` as the dependent variable and 8 previous lags of `sale_cell` as independent variables (see Figure 21). Then, we apply this model to the predicted series for `sale_cell` we plotted in Figure 12 to generate a predicted series for `rent_cell`. We visualize the distributions comparison between the one we obtain for `rent_cell` with the observed one across tiles of Munich in Figure 22.

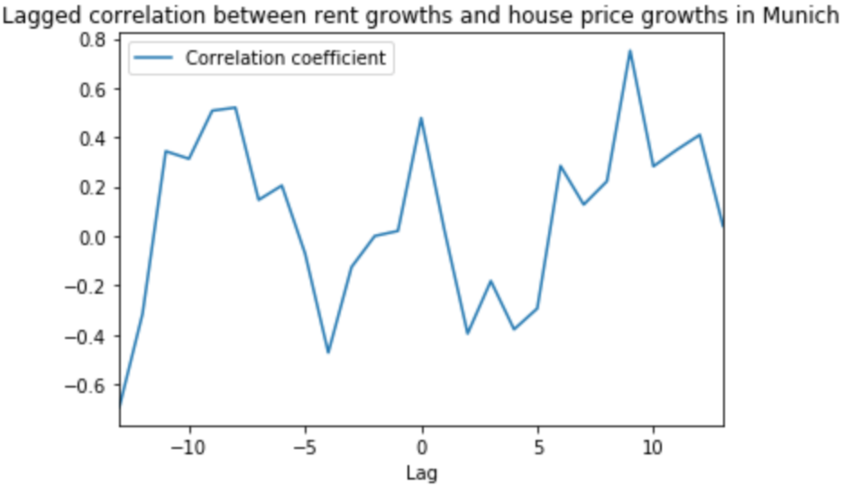
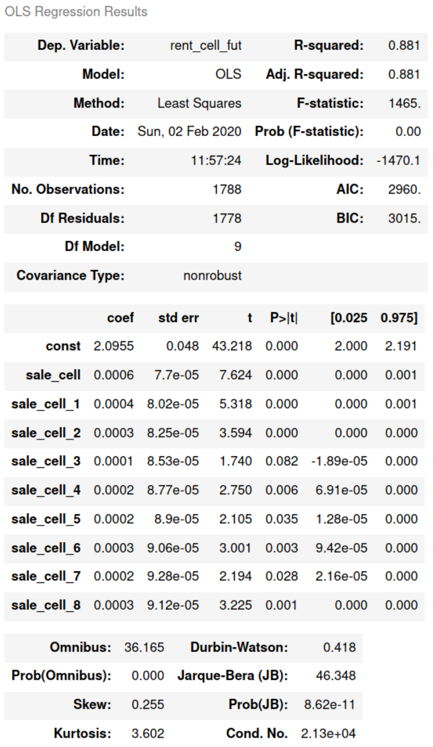


Figure 20: Lagged correlation between rent growth and price growth in Munich. Positive lag means that rent growth data is in the later time point than price growth data, negative lag means vice versa.

The means are even more skewed than what we saw for `sale_cell` in figure 12 back in Section 3.6 — this is most likely due to the unsatisfying fit of the cross-correlation model, unable to provide a good prediction for `rent_cell`. A different approach would be necessary to yield good results.

As a side note, one can see the large tails for quarters 1 and 2 of 2016; this is the direct data coming from 21st. It is the same data anomaly we noticed in Section 2.5, for Bayern and Baden-Württemberg. Munich is part of Bayern, after all.

Figure 22 shows the fit across all Germany, whereas in Figure 23, the fit is done only for Munich. In spite of much fewer data points available for fitting, the predicted distributions look much closer to the observed ones, indicating that this might be a better way to pursue this question. However, we did not run this for all cities, merely for Munich, and therefore it remains something to be further investigated.



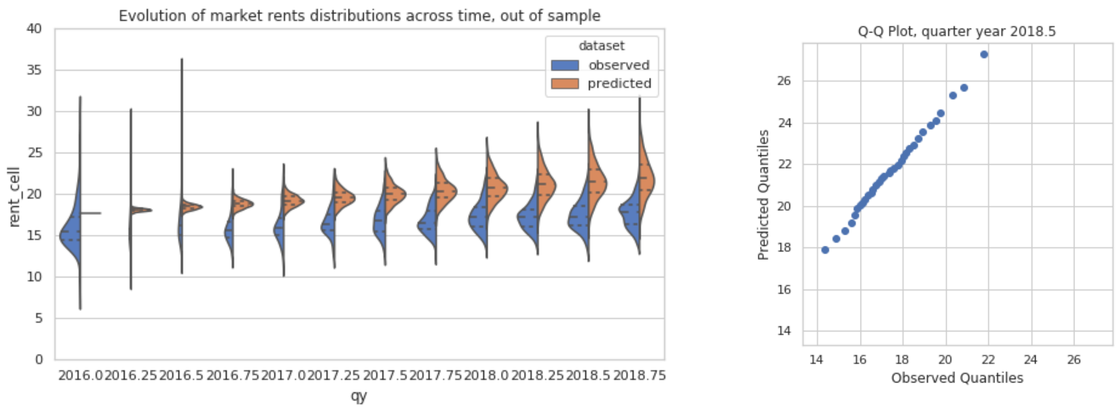
OLS Regression Results

Dep. Variable:	rent_cell_fut	**R-squared:**	0.881
Model:	OLS	**Adj. R-squared:**	0.881
Method:	Least Squares	**F-statistic:**	1465.
Date:	Sun, 02 Feb 2020	**Prob (F-statistic):**	0.00
Time:	11:57:24	**Log-Likelihood:**	-1470.1
No. Observations:	1788	**AIC:**	2960.
Df Residuals:	1778	**BIC:**	3015.
Df Model:	9		
Covariance Type:	nonrobust		

	coef	**std err**	**t**	**P>|t|**	**[0.025**	**0.975]**
const	2.0955	0.048	43.218	0.000	2.000	2.191
sale_cell	0.0006	7.7e-05	7.624	0.000	0.000	0.001
sale_cell_1	0.0004	8.02e-05	5.318	0.000	0.000	0.001
sale_cell_2	0.0003	8.25e-05	3.594	0.000	0.000	0.000
sale_cell_3	0.0001	8.53e-05	1.740	0.082	-1.89e-05	0.000
sale_cell_4	0.0002	8.77e-05	2.750	0.006	6.91e-05	0.000
sale_cell_5	0.0002	8.9e-05	2.105	0.035	1.28e-05	0.000
sale_cell_6	0.0003	9.06e-05	3.001	0.003	9.42e-05	0.000
sale_cell_7	0.0002	9.28e-05	2.194	0.028	2.16e-05	0.000
sale_cell_8	0.0003	9.12e-05	3.225	0.001	0.000	0.000

Omnibus:	36.165	**Durbin-Watson:**	0.418
Prob(Omnibus):	0.000	**Jarque-Bera (JB):**	46.348
Skew:	0.255	**Prob(JB):**	8.62e-11
Kurtosis:	3.602	**Cond. No.**	2.13e+04

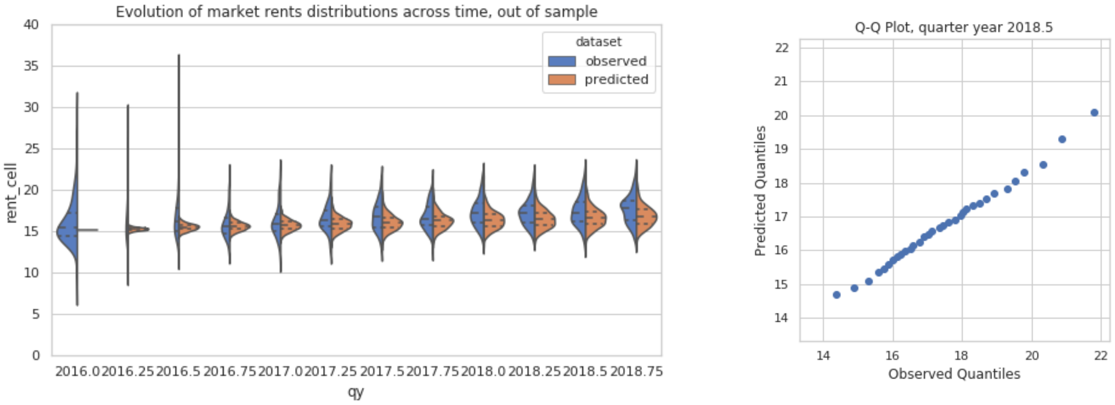
Figure 21: Regression summary of `rent_cell` against 8 lags of `sale_cell`. Train R^2 (< 2016) is 0.881, test R^2 (≥ 2016) is 0.900.



(a) Violin plot of distributions.

(b) Q-Q plot for quarter 3 of 2018.

Figure 22: Analysis plots of predicted vs. observed distributions of `rent_cell` for Munich. Extrapolated from a linear regression from lagged `sale_cell`, fitted on all cities.



(a) Violin plot of distributions.

(b) Q-Q plot for quarter 3 of 2018.

Figure 23: Same as 22, but the model is fitted only on Munich data. The fit looks much better, with more realistic means and shorter tails. This might not generalize to other cities, though!

A.6 VAR models

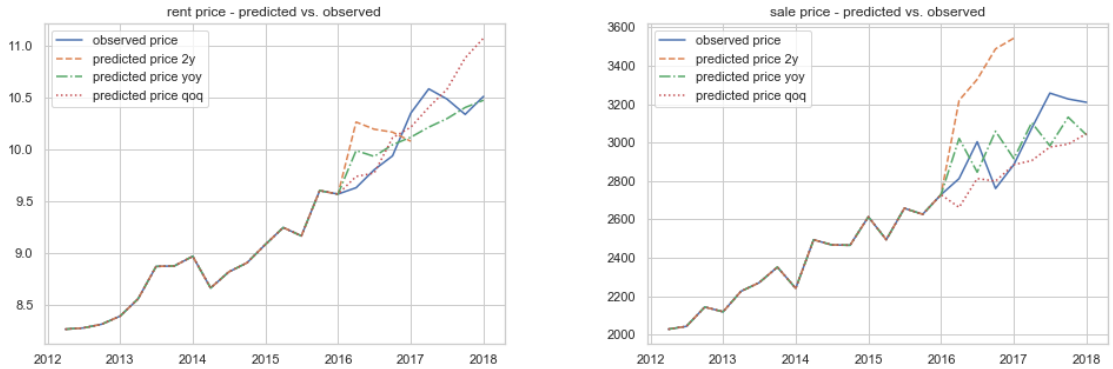
A.6.1 Overview of the selected features

Target	Number of selected variables	Selected variables
qoq_sale_cell	5	’birth_death’, ’hh_income_inhab’, ’n_flats’ ’svb_working’, ’life_exp’
yoy_sale_cell	1	’birth_death’
2y_sale_cell	5	’birth_death’,’tax_municipal’,’vacancy_rate’ ’up_female’,’migrbal’
qoq_sale_cell_growth	4	’gdp_qoq’, ’mortgage_rate_qoq’ ’current_acc_usd_growth’, ’gdp_yoy_qcd’
yoy_sale_cell_growth	1	’up’
2y_sale_cell_growth	4	’gva_agriculture_growth’,’hh_income_2ygrowth_syd’ ’employees_km2’,’mortgage_rate_2ygrowth’
qoq_rent_cell	5	’birth_death’,’hh_income_inhab’,’vacancy_rate’, ’birth_death_yoy’,’pop_age’
yoy_rent_cell	5	’birth_death’, ’hh_income_inhab’, ’vacancy_rate’ ’pop_age’, ’gdp’
2y_rent_cell	5	’birth_death’,’hh_income_inhab’,’vacancy_rate’ ’pop_age’,’gdp’
qoq_rent_cell_growth	3	’population_growth’, ’current_acc_usd_yoy’, ’up_yoy’
yoy_rent_cell_growth	3	’population_growth’, ’mortgage_rate_yoy’, ’birth_death’
2y_rent_cell_growth	5	’population_growth’,’birth_death’,’gdp_constant_annualc_x’ ’employees_km2_2ygrowth’,’vacancy_rate_yoy’

Table 6: Selected features for each of the targets based on the forward selection algorithm described in Section 3.8.

A.6.2 Forecast plots

The following figures visualize the prediction results of standard VAR models.



(a) Rent prices and predictions.

(b) Sale prices and predictions.

Figure 24: Different horizon predictions of rent and sale prices in Cologne (2012-2018), from the standard VAR model. The blue line corresponds to the observed prices, while the dotted lines refer to the predictions for different time horizons (`qoq`, `yoy`, `2y`).

While VAR models are not the best model for predicting each of the targets itself as we have seen in Table 3, the main advantage of this model is that we can achieve scenario-consistent predictions of the macroeconomic variables (see Figure 25).

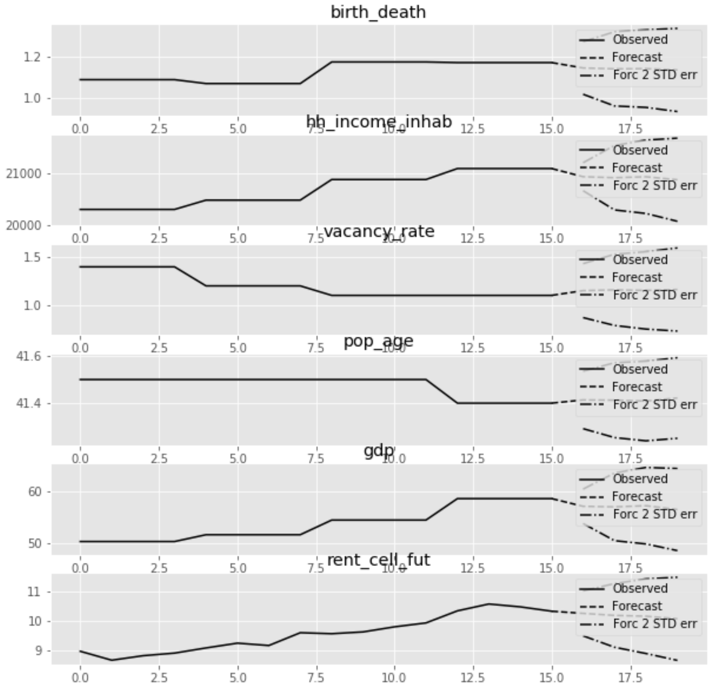


Figure 25: Illustration of standard VAR predictions of the target `2y_rent_cell` (bottom-most time series) and the corresponding macroeconomic variables as shown in Table 6 for Cologne.

A.6.3 Impulse response analysis

The impulse response analysis quantifies the reaction of every single variable in the model on an exogenous shock to the model. Two special cases of shocks can be identified: The single equation shock (Figure 26) and the joint equation shock where the shock mirrors the residual covariance structure (Figure 27). In the first case we investigate forecast error impulse responses, in the latter cumulative impulse responses. The reaction is measured for every variable a certain time after shocking the system. The impulse response analysis is therefore a tool for inspecting the inter-relation of the model variables.

They are computed in practice using the $MA(\infty)$ representation (see e.g. [Lüt05]) of the $VAR_m(p)$ process:

$$y_t = \mu + \sum_{i=0}^{\infty} \Phi_i u_{t-i},$$

where μ is the mean of y_t. Asymptotic standard errors are plotted by default at the 95% significance level.

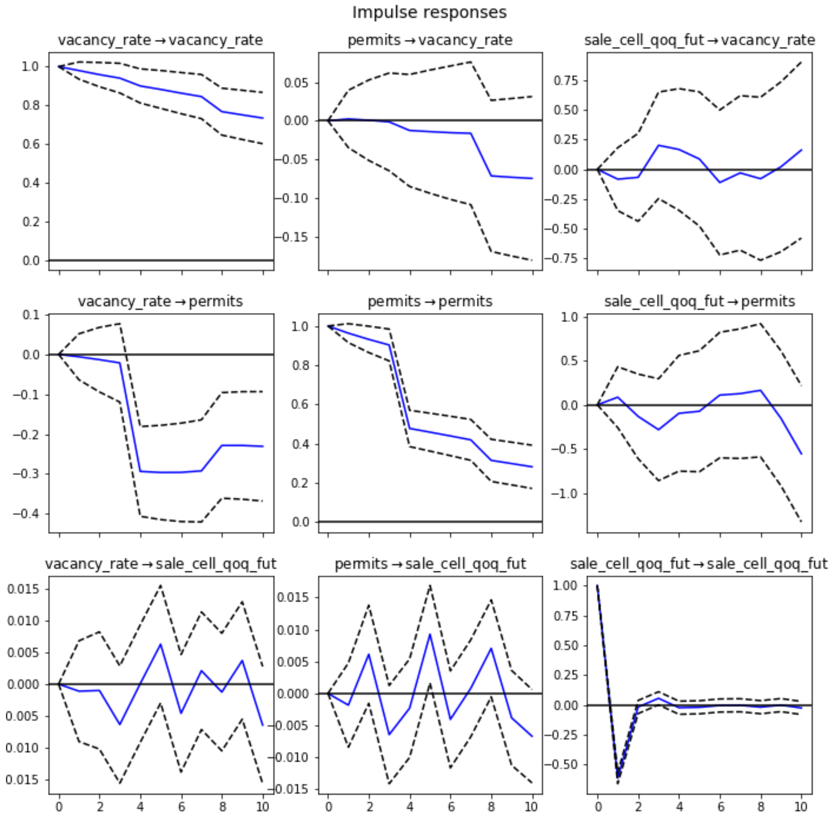


Figure 26: Illustration of the impulse response analysis of VAR models for visualizing the inter-dependencies between macroeconomic variables.

The cumulative effects $\Psi_n = \sum_{i=0}^{n} \Phi_i$ (see [Per+19]) can be plotted with the long run effects as illustrated in Figure 27.

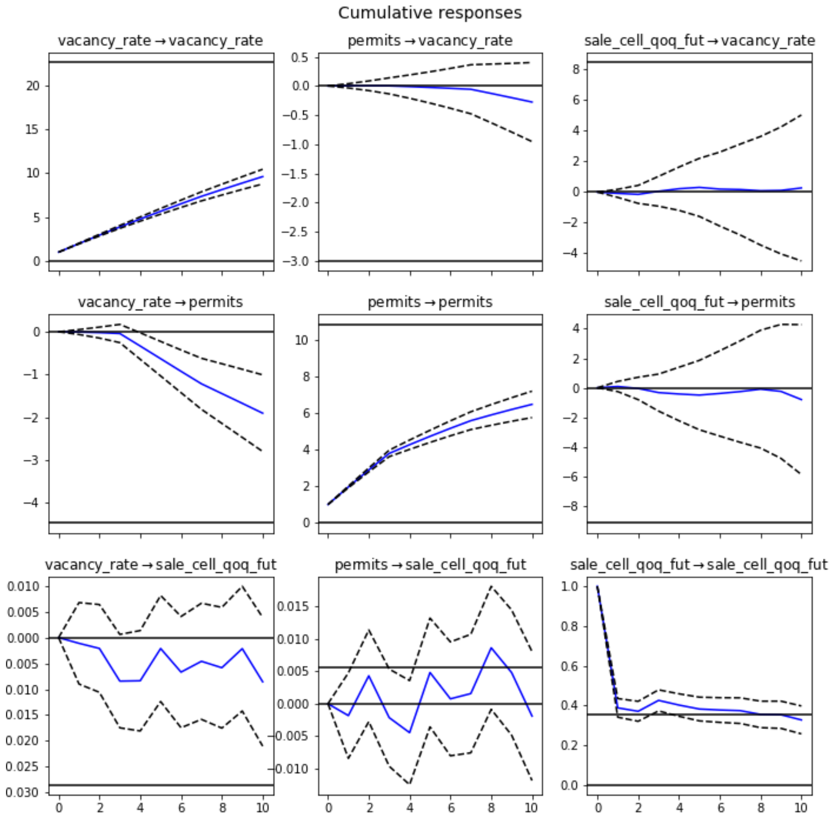


Figure 27: Illustration of the cumulative effects between macroeconomic variables.

List of Figures

List of Tables

Bibliography

[Lit86] Robert B Litterman. "Forecasting with Bayesian vector autoregressions—five years of experience". In: *Journal of Business & Economic Statistics* 4.1 (1986), pp. 25–38. URL: https://www.jstor.org/stable/pdf/1391384.pdf.

[DS98] Norman R Draper and Harry Smith. *Applied regression analysis*. Vol. 326. John Wiley & Sons, 1998.

[Pac+00] R Kelley Pace et al. "A method for spatial–temporal forecasting with an application to real estate prices". In: *International Journal of Forecasting* 16.2 (2000), pp. 229–246. URL: http://www.spatial-statistics.com/pace_manuscripts/IJOF/paper98-23.pdf.

[Lüt05] Helmut Lütkepohl. *New introduction to multiple time series analysis*. Springer Science & Business Media, 2005.

[Brä+06] Michael Bräuninger et al. *Immobilien. Teil I: Zukünftige Entwicklungen auf den Wohnungsmärkten in Deutschland. Teil II: Das Mehrfamilienhaus als Kapitalanlage–mit besonderer Berücksichtigung des Standortes Hamburg*. Tech. rep. Strategie 2030-Vermögen und Leben in der nächsten Generation, 2006. URL: https://www.econstor.eu/bitstream/10419/102538/1/729051951.pdf.

[Gal08] Joshua Gallin. "The long-run relationship between house prices and rents". In: *Real Estate Economics* 36.4 (2008), pp. 635–658. URL: https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1540-6229.2008.00225.x.

[AJ09] Joshua Aizenman and Yothin Jinjarak. "Current account patterns and national real estate markets". In: *Journal of Urban Economics* 66.2 (2009), pp. 75–89. URL: https://doi.org/10.1016/j.jue.2009.05.002.

[BT10] Chris Brooks and Sotiris Tsolacos. *Real estate modelling and forecasting*. Cambridge University Press, 2010.

[HM10] Shanaka Herath and Gunther Maier. "The hedonic price method in real estate and housing market research: A review of the literature". In: *SRE-Discussion Papers* (2010).

[GKM11] Rangan Gupta, Alain Kabundi, and Stephen Miller. "Using large data sets to forecast house prices: a case study of twenty US states". In: *Journal of housing research* 20.2 (2011), pp. 161–190. URL: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.464.2380&rep=rep1&type=pdf.

[Kur11] Björn-Martin Kurzrock. "Immobilienanalyse". In: *Immobilienwirtschaftslehre Band I*. Köln, 2011, p. 731.

[RW12] Małgorzata Renigier-Biłozor and Radosław Wiśniewski. "The impact of macroeconomic factors on residential property price indices in Europe". In: *Folia Oeconomica Stetinensia* 12.2 (2012), pp. 103–125.

[BYC13] James Bergstra, Dan Yamins, and David D Cox. "Hyperopt: A python library for optimizing the hyperparameters of machine learning algorithms". In: Citeseer. 2013. URL: https://conference.scipy.org/proceedings/scipy2013/pdfs/bergstra_hyperopt.pdf.

[CC13] Fabio Canova and Matteo Ciccarelli. “Panel Vector Autoregressive Models: A Survey. The views expressed in this article are those of the authors and do not necessarily reflect those of the ECB or the Eurosystem.” In: *VAR Models in Macroeconomics–New Developments and Applications: Essays in Honor of Christopher A. Sims*. Emerald Group Publishing Limited, 2013, pp. 205–246. URL: https://www.econstor.eu/bitstream/10419/153940/1/ecbwp1507.pdf.

[DG14] Stephane Dees and Jochen Guntner. “Analysing and forecasting price dynamics across euro area countries and sectors: A panel VAR approach”. In: (2014). URL: https://www.ecb.europa.eu/pub/pdf/scpwps/ecbwp1724.pdf.

[GLP15] Domenico Giannone, Michele Lenza, and Giorgio E Primiceri. “Prior selection for vector autoregressions”. In: *Review of Economics and Statistics* 97.2 (2015), pp. 436–451. URL: https://www.ecb.europa.eu/pub/pdf/scpwps/ecbwp1494.pdf.

[Boj16] Darja Kobe Govekarb Bojan Grum. “Influence of Macroeconomic Factors on Prices of Real Estate in Various Cultural Environments: Case of Slovenia, Greece, France, Poland and Norway”. In: (2016).

[CG16] Tianqi Chen and Carlos Guestrin. “Xgboost: A scalable tree boosting system”. In: *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*. 2016, pp. 785–794. URL: https://arxiv.org/pdf/1603.02754.pdf.

[KA16] A Krause and G Aschwanden. “Deriving a Rent-to-Price Ratio in Residential Markets: A comparison of methods”. In: (2016). URL: https://minerva-access.unimelb.edu.au/bitstream/handle/11343/91766/prmc.pdf?sequence=1.

[YS16] Jing Yao and A Stewart Fotheringham. “Local spatiotemporal modeling of house prices: A mixed model approach”. In: *The Professional Geographer* 68.2 (2016), pp. 189–201. URL: https://doi.org/10.1080/00330124.2015.1033671.

[Yİ17] Mustafa Ozan Yıldırım and Mehmet İvrendi. “House Prices and the Macroeconomic Environment in Turkey: The Examination of a dynamic relatonship.” In: *Ekonomski Anali/Economic Annals* 62.215 (2017). URL: https://doi.org/10.2298/EKA1715081Y.

[Sic18] Xavier Bourret Sicotte. *Choosing the optimal model: Subset selection*. 2018. URL: https://xavierbourretsicotte.github.io/subset_selection.html (visited on 01/2020).

[YS18] Bilgi Yilmaz and A Sevtap Selcuk-Kestel. “A stochastic approach to model housing markets: The US housing market case”. In: *Numerical Algebra, Control & Optimization* 8.4 (2018), p. 481. URL: https://www.aimsciences.org/article/doi/10.3934/naco.2018030.

[Per+19] Josef Perktold et al. *Vector Autoregressions tsa.vector_ar*. 2019. URL: https://www.statsmodels.org/dev/vector_ar.html (visited on 01/2020).

[Cro+] Henry Crosby et al. “A spatio-temporal, Gaussian process regression, real-estate price predictor.” In: URL: https://warwick.ac.uk/fac/sci/dcs/people/research/u1462772/shortpapercrosby.pdf.