NOVA IMS
Information Management School

MAA

Mestrado em Métodos Analíticos Avançados
Master Program in Advanced Analytics

# Generalized Additive Model Implementation for Germany Real Estate Market

## Model, API, UI Development

Berk Münger

Internship Report presented as the partial requirement for obtaining a Master's degree in Data Science and Advanced Analytics

NOVA Information Management School
Instituto Superior de Estatística e Gestão de Informação
Universidade Nova de Lisboa

**NOVA Information Management School**

**Instituto Superior de Estatística e Gestão de Informação**

Universidade Nova de Lisboa

# Generalized Additive Model Implementation for Germany Real Estate Market

## Model, API, UI Development

by

Berk MÜNGER

Internship Report presented as the partial requirement for obtaining a Master's degree in Data Science and Advanced Analytics

**Advisor:** Bruno Miguel Pinto Damásio

**External Supervisor:** Stephan Zhechev

November 2020

# ABSTRACT

Hedonic pricing approach one of the most accepted methodologies for the real estate price assessment by delivering attribute-based value.  It emerges from the value changing regarding object attributes conditions. In real estate market, these changes can be property renovation, material, and construction depreciation, or even expanding the plot area.

The scope of the internship report is to be explained the development first prototype General Additive Model of predicting House square meter price basis on Hedonic pricing theory for a certain region of Germany.

In addition to the model development, bringing it into live via Rest API and User Interface is explained in this report.

*Data Science Service GMBH* is the owner of the project and specialized in real estate property appraisal that is derived from statistical learning models, currently only at Austria. The outcome of this project enables us to get into Germany Real Estate Market as well.

The necessary data has been brought by German Market Partner, *Forschung und Beratung für Wohnen, Immobilien und Umwelt GmbH* (F+B), however Data Science Service GMBH (DSS) is responsible for delivering the model product from beginning to end.

R Programming Drake package is used for parallel computation and to be generated maintainable adaptive data pipeline.  Parameter selection based on information criteria has been done for each model in every kind of real estate property.

Lastly, the statistical model is delivered by rest API to UI (Shiny Application), both are developed with R programming language.

# KEYWORDS

General Additive Model (GAM); Hedonic pricing theory; Real estate property appraisal; mgcv.

# INDEX

# LIST OF FIGURES

# LIST OF TABLES

# LIST OF ABBREVIATIONS AND ACRONYMS

| | |
|---|---|
| **DSS** | Data Science Service GMBH |
| **GAM** | General Additive Model |
| **GLM** | General Linear Model |
| **GZLM** | Generalized Linear Model |
| **GCV** | Generalized Cross-Validation |
| **AIC** | Akaike Information Criterion |
| **HPM** | Hedonic Price Methodology |
| **OOB** | Out of Bag Error |
| **EFH** | House Property |
| **ETW** | Flat Property |
| **PLZ** | Postal Code |
| **W** | Wiggleness |
| **re** | Random Effect |
| **PIRLS** | Poisson Iteratively Reweighted Least Squares |
| **cr** | Cubic Regression Splines |
| **TPRS** | Thin Plate Regression Splines |
| **REML** | Restricted maximum likelihood |

# 1. INTRODUCTION

Statistical value assessment for real estate properties is not only crucial for the governments and the authorities but also vastly important for Banks and Insurance Companies. Since the subprime mortgage crisis happened in 2007, especially financial risk department of the banks whose offer individual housing loan strictly rely on the statistical analysis outcomes and its derivative results.

Even though the monthly published official real estate price index allows to be tracked general market trend for each country, it is not comprehensive enough in assessing individual property value except being an input for more extended analysis. This analysis can be conducted with real estate property characteristics such as location, construction year, floor, area size, room number even floor material. Therefore, in the real estate market, the characteristics prices approach highly preferable because of its simplicity as well as the fact that the property most likely to be revised on time.

The hedonic price methodology (HPM), that most well know and valid characteristics prices approach at the real estate market and it is presented with joint envelope function that includes all characteristics as variable, implicitly with their price effect for every single characteristic.

Hedonic price methodology can be implemented through various algorithm however, black-box algorithms don't guarantee the interpretability of model regarding character changes in first prototype development. The functional form model that also allows nonlinearities has broadly used Wallace (1996) or Malpezzi (2003). General Additive Model (GAM) offers nonlinear relation in functional from including covariates effects. Moreover, some variables are strictly known as positive (garage) or negative (age) effect for predicted value in any circumstances, hence the predicted value should be updated in the expected direction. GAM model can be adjusted also within that scope.

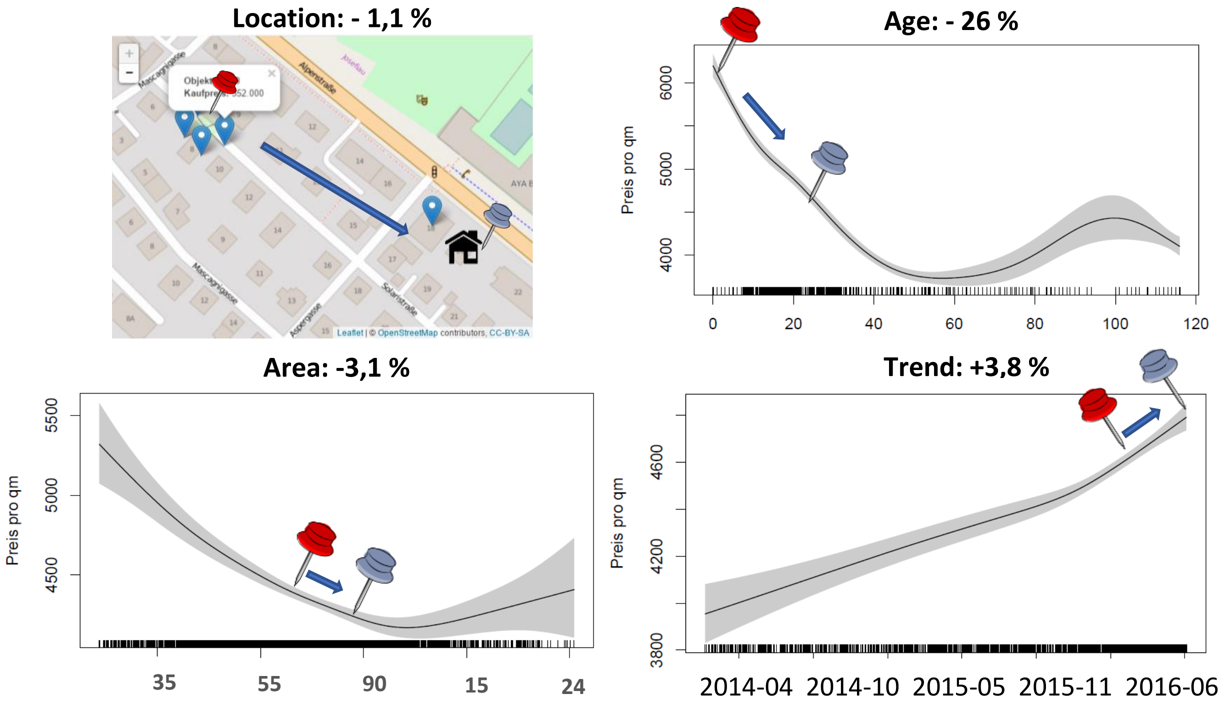


Figure 1-1: GAM model variable effect in Hedonic Pricing

Data Science Service Gmbh (DSS) since 2016 delivers Austria Real Estate Price Index with Hedonic Price Methodology and its advanced implementation through General Additive (GAM) model. Long years of studies and data investments allow DSS to obtain a comprehensive GAM model to serve advanced analysis for Austria Real Estate Market whereas German Real Estate Market Analysis has not fully conducted yet.

Austria Market experience shows that some certain of variables such as construction year, postal code, building area, plot area, various attributes such as garage, terrace, cellar, floor, etc. information allows developing simple yet accurate prediction models. Launching a heuristic model with the simplest implementation, following iterative product development strategy, allows us to get feedback to our users in the improvement of model and helps to present outcomes in the German Market.

Regarding explained concept above, this internship report includes all necessary steps for bringing the simple GAM model into production. For the sake of simplicity, the model that will be detailed through that report is limited with only one type of property and two census track level data. Germany - North Rhine-Westphalia and one individual type – House.  The remaining states model is developed with same methodology.

North Rhine-Westphalia is located in western Germany and most densely populated German state. The GAM model basis on 308.000 House and 148 variables. However, the GAM model will be conducted with heuristically specified inputs which corresponds to 24 independent variables.

| ***Observation*** | |
|---|---|
| *≈ 10.000.000* | [1 Country - 3 Different Object Type (House, Flat, Rent Flat)] |
| *≈ 2.200.000* | [1 Object Type (House) - 16 Different State] |
| *≈ 420.000* | [1 Object Type (House) - 1 State (North Rhine-Westphalia) - 53 Region] |
| *≈ 385.600* | [North Rhine-Westphalia – Houses Cleaned Data] |
| ***≈ 308.500*** | Model Training Data |

Table 1-1: Observations with Granular Levels

The ultimate goals of this report are:

1- Developing simple but accurate GAM model that also tackles data problem by implementing elaborative solutions.

2- Launch this model as ready to use product into the German Real Estate Market.

In order to reach these goals below procedure has been followed:

1- Regarding the exploratory analysis data cleaning, data transformation and data imputation steps are completed with R programming.

   a. Data preparation procedure is conducted for not only training data but also test data.

   b. Trained data, test data and Input data have been limited with certain threshold for each individual.

   c. Data preparation is functionalised also to be used in prediction.

d. R Drake Package is used to make it possible parallel computation.

2- GAM model is developed with *mgcv* R package using bam() function.

a. Three groups of explanatory covariates will be included into model:
Object characteristics (age, size, etc.)
Location characteristics
Time indicators (quarter)

b. Continuous covariate effects are modeled using penalized regression splines:
In default, penalized regression spline method in *mgcv* package are thin plate regression method. i.e. low rank smoothers base on truncated Eigen-decomposition.

c. Spatial indices on two hierarchical levels are modeled using random effects (penalized by a ridge penalty)

d. Automated selection of the smoothing parameter using the Generalized Cross Validation (GCV) criterion.

e. Insignificant covariates such as the age increased also predicted price increase is eliminated with *model update* functionality by eliminating some of splines with Stepwise Algorithm approach.

f. The residuals from the GAM model included hedonic price approach, using a boosted tree model with the *XGBoost* implementation. Parameter tuning with the caret package in R is performed.

g. From simple to complex GAM models are compared with Generalized Linear Model (GLM) through Test Dataset regarding RMSE value.

3- Best model is carried out into API and launched as Product.

a. GAM model is saved as *.Rdata* file so as to be loaded into API rest service.

b. API rest service is called over User Interface through main product called as *Immazing*.

*c.* The Austria QUICK product is adapted as German QUICK product with R programming Shiny Package.

# 2. LITERATURE REVIEW

## 2.1. HEDONIC PRICE METHOD (HPM)

In assessment of real estate property value, several approaches have been proposed in the literature and the hedonic method, the repeat-sales method and stratification method are mostly highlighted methods. Hill (2013) compares all these three models and concludes that hedonic price theory becomes more popular due to overcomes the problems in repeated sales.

The hedonic price theory was first developed by Waugh (1928) to distinguish land characteristics. However, the first time was used as methodology for developing price measures for automobiles. By A. Court at 1939 (Goodman, 1998, pp. 291-298). The method was popularized by Griliches (1961) in context of location effects on house prices.

Reviews of hedonic price theory in a real estate context that is not only location effect are provided in Follain and Jimenez (1985) and Malpezzi (2003) studies. In these studies, HPM implies that individual characteristics creates utility rather than itself. As housing characteristics are non-separable and traded in bundles, real estate is usually treated as a heterogeneous good such as structural (physical) characteristics, like floor space area, constructional condition, age etc. HPM enables the price of a housing unit is decomposed into implicit prices of the characteristics which are estimated in a regression analysis of price against characteristics. The real estate properties have several levels of spatial units such as district, city, state, country. Therefore, HPM should also be considered as multilevel or hierarchical regression problem. (Gelman & Hill, 2007).

One of the main challenges in HPM is that it doesn't suggest certain the functional form of the dependence of price on characteristics. HPM is adapted to the non-linear functionality. (Ekeland, 2004). The most used specification to address this problem is the semi-log form (Sirmans, 2005), but this only seems to mitigate the problem of possible nonlinear relationships to some extent. Therefore, Anglin and Gencay (1996) demand the use of semi or nonparametric specifications for this situation. Other examples of semi and nonparametric approaches for real estate can be found in study whose Mason and Quigley (1996).

| **Property Attributes** | **Elements** | **Authors** |
|---|---|---|
| Location | Distance to city center | Tang (1975) |
| | Accessibility to transportation | Sirpal (1994); Meen (2001); Des Rosiers, Lagana, Theriault, & Beaudoin (1996) |
| Structure | Number of bedrooms | Kain and Quigley (1970) |
| | Number of bathrooms | Rodriguez & Sirmans (1994) |

Table 2-1: Most well-known HPM Literature in Real Estate Market

## 2.2. Generalized Additive Model

### 2.2.1. General Additive Model (GAM)

The linear regression model has the form:

$$Y = X\beta + \varepsilon$$

where $Y$ is a n-dimensional vector for the response variable, $X$ is the matrix $(n \times p)$ of the independent $p$ variables $X_1, \dots, X_p$, $\beta$ is the vector of the model parameters and $\varepsilon$ is the vector of random disturbances with 0 mean and variance $\sigma^2 I$ $(\text{I denotes the identity matrix})$.

| Name | Method | Limitation |
|---|---|---|
| Linear Regression Model | Interaction Between Different Variables | Nonlinearity |
| General Linear Model | Multiple Linear Regression for single dependent variable | The distribution of single dependent variable is normal.<br>Link function is identity function. |

Table 2-2: Limitation of GLM and Linear Regression

The linear regression model presents certain limits and are inadequate when the assumption of normality of the response variable is no longer justified. Therefore, the linear model is extended to the Generalized Linear Model (GZLM). GZLM (McCullagh & Nelder, 1989, pp. 8) relates the mean of a response $(Y)$ to a linear combination of independent variables. The response is assumed to be conditionally distributed according to some exponential family distribution such as binomial, passion, gamma distributions etc.

The distribution $Y$ is related to the linear combination of the independent variables (covariates) via the link function $g(E[Y]) = X\beta$.

| ***GENERAL ADDITIVE MODEL*** | |
|---|---|
| Generalized Linear Model | Additive Model |
| The distribution of response variable can be non-normal | A framework that is positioned between parametric and nonparametric settings |
| Does not have to be continuous variable | Replacing each linear term with a general, non-linear one sum of univariate regression functions |
| Linear Combination of variables predicts the dependent variable via a link function | |
| ***GAM is to maximize the quality of prediction of a dependent variable from various distribution, by estimating non parametric functions of independent variables that are connected to dependent variable via a link function.*** | |

Table 2-3: GAM – GZLM comparision

To introduce more flexibility in the dependence structure between the response variables and covariables, GAM replace the linear dependence functions with more flexible non-linear functions

(Hastie & Tibshirani, 1990). The dependences are generally presented by non-parametric smoothing functions. These functions are called splines (De Boor, 1978).

The GAM allows a broad range of distributions for the response variable to be adopted, and link functions for measuring the effects of the predictor variables on the dependent regressors as reported by McCullagh and Nelder (1989) and Hastie and Tibshirani (1990). Popularly used distributions in GAM modeling are Normal, Gamma and Poisson distributions.

$$E[Y] = g^{-1}\left(\beta_0 + \sum_{j=1}^{J} f_j(x_j)\right) \qquad f_j(x_j) = \sum_{k=1}^{K} \beta_{j,k} b_{j,k}(x_j)$$

$f_j$ is a smooth function (spline) of the covariate $x_j$, $\beta_0$ is an intercept term, and $g^{-1}$ is the inverse link function. Each $f_j$ is represented by a sum of $K$ basis size, fixed basis functions $b_{j,k}$ multiplied by corresponding coefficients ($\beta_{j,k}$), which need to be estimated. These basis functions can be considered as extra columns in the data as similar to a transformed variable.

Weighting basis functions ($\beta_{j,k}$) enables to obtain the estimated spline. The weightings correspond to the coefficients of each basis function that is estimated from the data in such a way that by maximizing log likelihood. However, overfitting is the main challenge in the result of basis function expansion with the likelihood that makes the function more wiggly rather than smooth. To avoid overfitting, wiggleness ($W$), is controlled with smoothing parameter ($\lambda$) value in the finding of spline coefficients ($\beta_{j,k}$). The tradeoff between model fit and smoothness is controlled by the smoothing parameter. The spline that is set as closest as possible to the data with log-likelihood gains smoothness with $\lambda$ by maximizing log - likelihood. (Wood, 2012).

$$L_p = log(Likelihood) - \lambda W$$

(A) estimated using REML; (B) l set to zero (no smoothing); (C) l is set to a very large value.

### 2.2.2. Smoothers (Splines)

In GAM, the model is based on smoothers. In general, it can be classified into three types:

a. Regression splines (thin-plate spline, cubic regression spline etc.)
b. Local regression (loess) splines
c. Smoothing Splines.

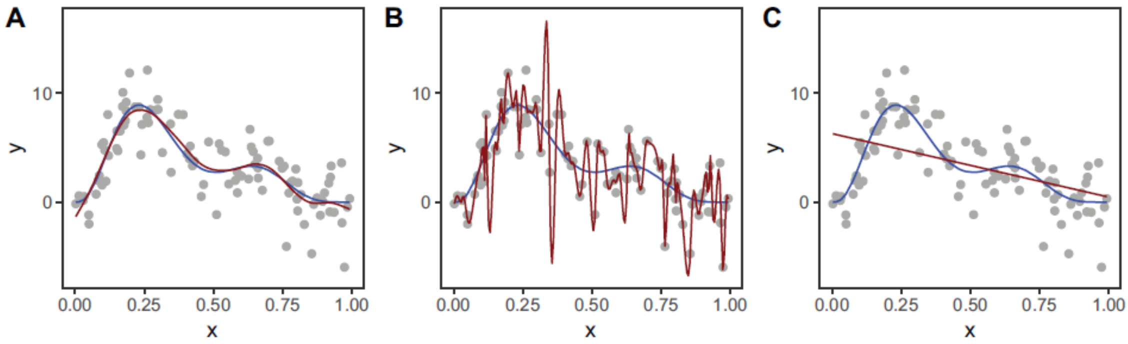


Figure 2-1 Smoothing Parameter Effect Comparison (Hastie & Tibshirani, 1990)

Regression splines are most likely to be used because of easy computation, and it can be written as a linear combination of basis function; which is well suited for estimation and prediction (De Boor, 1978) (Wood, 2000).

### 2.2.2.1. Regression Spline

a. Cubic Regression Spline

Cubic spline essentially a connection of multiple cubic polynomial regressions. We choose points of the variable at which to create sections, and these points are referred to as knots. Separate cubic polynomials are fit at each section and then joined at the knots to create a continuous curve. (Wood, 2006).

b. Thin Plate Spline

The thin-plate regression splines are based on thin-plate smoothing splines (Duchon, 1977). Compared to thin-plate smoothing splines, thin-plate regression splines produce fewer basis expansions and thus make direct fitting of generalized additive models possible.

The Thin-Plate Spline analysis is intended for scatter plot smoothing. The Thin-Plate Spline analysis uses a penalized least squares method to fit a nonparametric regression model. Generalized cross-validation (GCV) function to select the amount of smoothing. The R package that was used in this project (*mgcv*) offers GCV method in default.

### 2.2.2.2. Local Regression Splines

Local Regression Loess (loess)  belongs to the class of nearest neighborhood-based smoothers. To appreciate loess, we have to understand the most simplistic member of this family: the running mean smoother. Running mean smoothers are symmetric, moving averages. Smoothing is achieved by sliding a window based on the nearest neighbors across the data, and computing the average of Y at each step. The level of smoothness is determined by the width of the window. While appealing due to their simplicity, running mean smoothers have two major issues: they're not very smooth and they perform poorly at the boundaries of the data. This is a problem when we build predictive models, and hence we need more sophisticated choices, such as loess.

The local regression approach has been used to estimate house prices by researcher like Wallace (1991) and has been combined with a parametric model by Clapp (2004) to estimate local house price indices.

### 2.2.2.3. Smoothing Splines

Smoothing splines take a completely different approach to deriving smooth curves. Rather than using a nearest-neighbor moving window, we estimate the smooth function by minimizing the penalized sum of squares. For a certain feature, the smoothing spline is produced by finding the function $f(x)$ that minimizes the penalized residual sum of squares.  (Hastie & Tibshirani, 1986). The function $f(x)$ should have a continuous second derivative.

$$\underbrace{\sum_{i=1}^{n}\left(y_i - f(x_i)\right)^2}_{\text{Fit observed data}} + \underbrace{\lambda \int (s''(x))^2 dx}_{\text{Penalty Term}}$$

### 2.2.3. Real Estate Market Covariates

The floor area is the central property characteristic and a pronounced positive effect on the purchase price is expected. Malpezzi (2003) advises a logarithmic transformation considering multiplicative structures.

Year of sale and year of construction reflects property depreciation over time and should therefore have a decreasing effect on flat prices. Nevertheless, a vintage effect, which has the opposite consequences, is possible (Can, 1998, pp. 61-86). Goodman and Thibodeau (2003, pp. 181-201) and Brunauer (2010) report non-linear age effects. The year of the time of sale can be regarded as the remaining unexplained temporal heterogeneity. It is a measure for the quality adjusted development of prices over time and is modeled as a numeric covariate.

### 2.2.4. Spatial Heterogeneity in GAM

Fixed effects can be integrated by allowing the intercepts vary over space. Slope heterogeneity can be controlled through spatial interaction effects with explanatory covariates (Kestens, 200). heterogeneity using many fixed effects can result in insufficient observations within regions for parameter estimations which, due to the loss of degrees of freedom, decrease the prediction accuracy. Therefore, it is common practice to interact where only "one-variable-at-a-time" is considered (McMillen and Redfearn, 2010, pp. 713). Thus, a trade-off between both data fidelity and reduction of prediction accuracy is required which is partially solved by the random effects model (Goldstein, 2011).

The simplest random effects model is the one-way intercept model, whereby intercepts vary spatially. Random effects can be approximated as a weighted average of the mean of the observations in the spatial units (corresponding to a dummy specification) and the overall mean of the whole country (Gelman & Hill, 2007). The weights are determined by the amount of information in each region. Random effects models can be generalized by two-way structures (e.g., random time effects) and by letting predictors interact with the random effects, allowing different intercepts and/or slopes within each region (Goldstein, 2011).

# 3. GAM MODEL IMPLEMENTATION

The general additive model was conducted following steps:

1. Data Preparation
    - 1.1. Data Cleaning and Outlier Treatment
    - 1.2. Data Imputation with Random Forest
    - 1.3. Data Transformation
2. Model Fitting
    - 2.1. Generalized Linear Model
    - 2.2. General Additive Model
        - 2.2.1. Spline: Cubic Spline, Smoothing Parameter Selection: GCV and REML, Knots: 5
        - 2.2.2. Spline: Thin Plate Spline, Smoothing Parameter Selection: GCV and REML, Knots: Each Observation – PCA method: eigen decomposition.
    - 2.3. Ensemble Learning: Model Residual – XGBoost + General Additive Model
3. Model Validation

## 3.1. DATA PREPARATION

| **PREPARED DATA** | | | |
|---|---|---|---|
| **Binary** | | | |
| stell | balc | rh | dhh |
| wintergarten | garten | loggia | furnishing |
| gar | cell | villa | renovation |
| **Continuous** | **log transformed** | **Time** | **Location** |
| age | ln_area | year_fac | KGS05a |
| Rating | ln_plot | quarter | PLZ1 |
| efh_bp | | **offset_cond** | |

Table 3-1: Model Data Inputs Overview

Data has been cleaned from the variables that are not planned to be used in model development such as sauna, pool. Unbalanced amount of inputs most likely was reason of cleaning. (Appendix – 9.3. Deleted Variables)

General Additive Model is highly sensitive of outliers. Therefore, strict outlier treatment has been conducted with heuristic approach such as houses older than 150 year is changed with 150. Time inputs corresponds to advertised time of property. Older than 2014 advertisement deleted from raw data.

Outliers are treated into same function with data transformation. (Appendix – 9.4. Data Preparation Function)

Offset Condition is prepared for adding to the GAM model that enables the predicted value differentiation even though the model doesn't includes that variable as independent variable such as "swimming pool". (Appendix – 9.2. Offset Values for House)

Data describing the neighborhood or location is accounted for on the most granular level available, in a possibly nonlinear functional form as splines – non parametric functions

## 3.2. Data Transformation

The model pipeline is run since new data has been presented for model improvement. Pipeline steps has instructed below respectively.

### 3.2.1. Geocoding

The raw data obtained from German Partner (F+B) is not including longitude-latitude information of observations.

Therefore, the dataset is sent to *Wigeogis (GIS Data Solution Provider)* in order to obtain lon, lat from property address information. Generated lon, lat information is merged back to raw data.

*adr_qua_short* column is categorical variable that defines quality of address description into address input and it enables us to classify which observations are well geocoded.

### 3.2.2. Migration

The raw data variable names are adapted into master data schema which is designed by DSS for the purpose of variable name unification also considering Austria Data.

Therefore, the variables are renamed and transformed according to the model needs. Variable types are aligned with expected variable types.

In addition to variable renaming, variable values also refactored with set of rules such as date Format, upper lower Case, treatment of NA, NULL, “” values.

### 3.2.3. Data Transformation

Even though the dataset is refined to populate a database, the input variables have not been treated enough to construct GAM Model yet.

Data anomalies are detected and set to fixed values. Outlier treatments are made with simple calculations using empirical thresholds.

| Descriptions | Value | Transformation |
|---|---|---|
| model_obj_type | String | Observations are classified among Flat (etw), Rent Flat (etw_miete), House (efh) and Rent House (efh_miete). This variable enables to determine constructing model type. |
| rh | Binary ( 0 / 1 ) | House Type: Townhouse. One Type of House and determined considering "haustyp" string input. Looking inputs: "REIHENECK","REIHENEND","REIHENHAUS","REIHENMITTEL" |
| dhh | Binary ( 0 / 1 ) | House Type: Semi Detach House. One Type of House and determined considering "haustyp" string input. Inputs: "DOPPELHAUSHAELFTE", "MEHRFAMILIENHAUS", "ZWEIFAMILIENHAUS", "APARTMENTHAUS", |
| villa | Binary ( 0 / 1 ) | House Type: Villa. Inputs: "DOPPELHAUSHAELFTE", "MEHRFAMILIENHAUS", "ZWEIFAMILIENHAUS", "APARTMENTHAUS" |

| | | |
|---|---|---|
| letztemodernisierung (renavation year) | numeric | letztemodernisierung < 1990 **OR** letztemodernisierung < baujahr (construction year) + 5 set **NA** or **RENAVATION YEAR** |
| Factor of Year (factor) | Factor | real estate conveyed date will be used as categorical variable into GAM Model |
| cond3_gut ( Furnishing Condition) | Factor (0,1) | Binary Categorical Input for Model |
| cond1_schlecht ( Furnshing Condition ) | Factor (0,1) | Binary Categorical Input for Model |
| saniert ( renovated ) | Factor (0,1) | Binary Categorical Input for Model as depends on hauszustand (house condition) input |
| Wohnflache | House Area | ln_area |
| Age | Current Year – Construction Year | ln_age |
| Grundstuckflache | Plot area | ln_plot |

Table 3-2: Data Transformations

Below Figures show the log transformations:

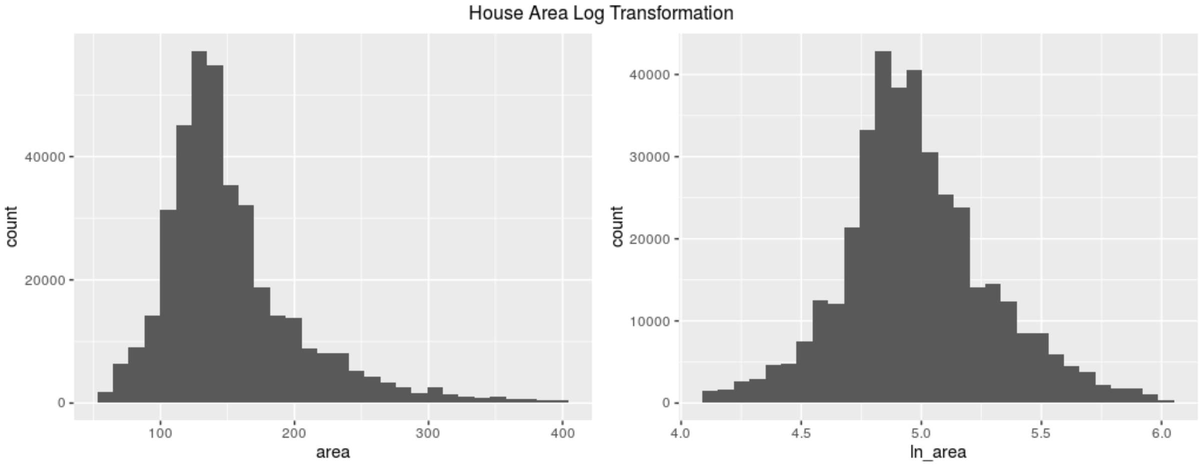


Figure 3-1: Log Transformation for House Area

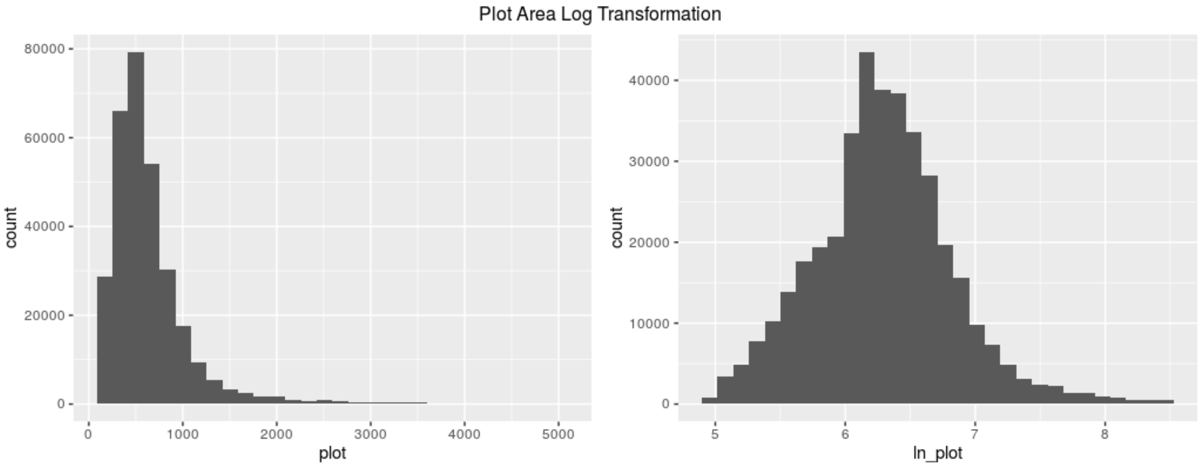


Figure 3-2: Log Transformation for Plot Area

### 3.2.4. Granular Spatial Data Merging

Each observation at current dataset only includes coordinates (longitude and latitude) and PLZ (postal code) information. In this step, this dataset will be enriched with Germany State codes, name, real estate property index information basis on postal code. (*efh_bp*) For observations that doesn't have coordinate information however valid postal code data, Postal Code Centroids have assigned as coordinates. Binary value for variable *exact_geocode* is set to "0". Implicit cleaning for observations that do not have valid postal code is made.

Duplicated data is deleted from dataset regarding square meter, year, coordinates, sales price, advertised year. Added variables are:

| Variable | Type | Description |
|---|---|---|
| exact_geocode | Binary value | Indicates the property is whether coordinated exactly or not |
| Rating | Discrete value | property scoring point generated data by German Business Partner |
| KGS02a | Spatial Factor | A certain Region belongs to a state consists of some Postal Code |
| efh_bp | Continuous value | Price index for per m² |
| PLZ | Spatial Factor | Postal Code |
| Lat_plz / lon_plz | Spatial Factor | Coordinates of Centre of PLZ |

Table 3-3: Spatial Data Variables

## 3.3. DATA IMPUTATION

Data Imputation for the variables "age" and "ln_plot" highly important before fitting into the model. Therefore, these two variables have taken care of with random forest method due to having highly computation power through *ranger R packages*.

Out-of-bag error (OOB) Error estimate has been preferred in tuning of parameters. The study of error estimates in ), gives empirical evidence to show that the out-of-bag estimate is as accurate as using a test set of the same size as the training set. Therefore, using the out-of-bag error estimate removes the need for a set aside test set.

The imputation is done for every subregion of all data. Below example only belongs to 1200 real estate property that is grouped of certain postal codes for age variable.

### 3.3.1. Out-of-bag (OOB) score

Each individual tree of a random forest uses a sample of the rows of the dataset which means that some of the rows did not get used for training. We can take advantage of this fact and pass those unused rows through the first tree and treat it as a validation set.

For the second tree, we could pass through the rows that were not used for the second tree, and so on. Similarly, we can perform the same procedure for all the trees of the Random Forest. Effectively, we now have a different validation set for each tree. To calculate our prediction, we would average all the trees where that row is not used for training.

### 3.3.2. Tuning Parameters for random forest

$m_{try}$ : The hyperparameter that controls the split-variable randomization feature of random forests. It helps to balance low tree correlation with reasonable predictive strength. In default it is preferred to be selected as equals to $\frac{feature\ quantity}{3}$ that indicates 7.

$Number\ of\ trees$ : the number of trees needs to be sufficiently large to stabilize the error rate. A good rule of thumb is to start with 10 times the number of features; however, as you adjust other hyperparameters such as $m_{try}$ and node size, more or fewer trees may be required. 220 number of trees is set in default.

Grid search across several hyperparameters is done for finding optimum number of trees and randomized feature between 1-20 and 50-200 respectively. Feature importance basis on the average total reduction of the loss function for a given feature across all trees for imputing age.

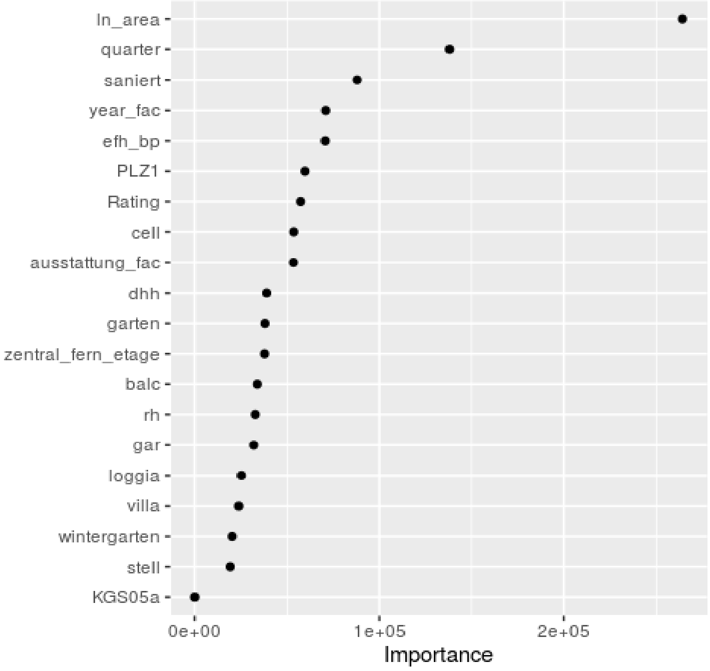


Figure 3-3: Variable Importance imputing of age

| mtry | num.trees | OOB_RMSE | perc_gain |
|---|---|---|---|
| 11 | 450 | 2.390.263 | 1.475.819 |
| 7 | 250 | 2.390.441 | 1.468.479 |
| 12 | 350 | 2.390.629 | 1.460.753 |
| 12 | 400 | 2.393.701 | 1.334.134 |
| 9 | 300 | 2.394.126 | 1.316.590 |
| 8 | 250 | 2.394.456 | 1.303.014 |
| 9 | 450 | 2.394.805 | 1.288.611 |
| 10 | 500 | 2.395.058 | 1.278.182 |
| 12 | 250 | 2.395.525 | 1.258.952 |
| 9 | 200 | 2.397.287 | 1.186.299 |

Table 3-4: Grid Research Result Top 10 Lowest Error

## 3.4. MODEL

The data that will be fit into model is to be explored at Appendix 9.4

The multiple model is constructed regarding two different spline selection method and two different smoothing parameter selection methodology in addition to one simple GLM model. Following procedure is followed in fitting GAM Model:

1) The variables and corresponding signs are introduced. For instance, adding a balcony to an apartment should increase the price rather than decreasing it. The variables that cannot be known also specified. (Appendix 9.5)

2) Three types of smoothers are used in selecting of splines weighting: thin plate regression splines (TPRS), cubic regression splines (CRS), and random effects.

$$\begin{gathered} lnp\_qm\sim \\ s(KGS05a\,, bs =' re') + s(PLZ1, bs = 're', by = by\_plz) + \\ s(ln_{area}) + s\big(ln_{plot}\big) + \quad s(age) + \\ (quarter, bs = 're') \end{gathered}$$

   a. $s(ln_{area}) + s\big(ln_{plot}\big) + \quad s(age)$

      These are splined with two different method. CRS and TPRS spline types are used for these three smoothing functions.

   b. $s(KGS05a\,, bs =' re')$

      Random effect is to most simple way to obtain group-level splines. This is useful if different groups differ substantially in how wiggly they are. Therefore, random effect spline methodology is fixed for census track level data. Besides each real estate advertised time is considered with random effect.

3) Two different smoothing parameter selection methodology is conducted.

   a. Minimized generalized cross-validation (GCV) score: This method is selected as a argument of GAM function sourced by *mgcv* R package.

      The below procedure is followed at fitting data into model:

      1. Leave out observation i
      2. Estimate a smoothing curve using the n-1 remaining observations
      3. Predict the value at omitted point $X_i$
      4. Compare predicted value at $X_i$ with real value $Y_i$
      5. The difference between the original and predicted value at $X_i$ is given by

         $$Y_i - \hat{f}_\lambda^{-i}(X_i)$$

         This process is repeated for each observation
      6. Adding up all squared residuals gives the cross-validation error

         $$CV[\lambda] = \frac{1}{n}\sum_{i=1}^{n}\big(Y_i - \hat{f}_\lambda^{-i}(X_i)\big)^2$$

      7. The cross-validation error is calculated for various values of the smoothing parameter. The value of parameter that gives lowest value for CV is considered the closest the optimal amount of smoothing.

b. Restricted Maximum Likelihood (REML)

Since GAM has a Bayesian interpretation it can be treated like a standard mixed model by separating out the fixed effects and estimating the smoothing parameters as variance parameters. The variance of the coefficients depend on *p*, which in turn depends on $\lambda = \lambda 1, \ldots, \lambda p$.)

The restricted likelihood function, given the vector of smoothing parameters, λ, is obtained by integrating out beta from the joint density of the data and the coefficients. (Wood, 2004).

$$l_r(\hat{\beta}, \lambda) = \int f(y \backslash \beta) f(\beta) d\beta$$

The restricted likelihood function depends on λ and the estimates β (through the penalty), but not the random parameters β. Thus, it can be used this function to derive trial vectors for λ for a nested PIRLS iteration:

1. Given a trial vector λ, estimate β using PIRLS.
2. Update λ by maximizing the restricted log likelihood.
3. Repeat steps 1 and 2 until convergence.

4) Ensemble Learning - XGBoosting

In boosting, the trees are built sequentially such that each subsequent tree aims to reduce the errors of the previous tree. Each tree learns from its predecessors and updates the residual errors. Hence, the tree that grows next in the sequence will learn from an updated version of the residuals. (Appendix 9.6)

1. An initial model *F0* is defined to predict the target variable y. This model will be associated with a residual (y – *F0*)
2. A new model h1 is fit to the residuals from the previous step
3. Now, *F0* and h1 are combined to give *F1*, the boosted version of *F0*. The mean squared error from F1 will be lower than that from *F0*

### 3.4.1. Model Validation

| Splines | Effective Degree of Freedom (edf) | | | |
|---|---|---|---|---|
| | GCV / TP | REML / TP | GCV / CR | REML / CR |
| s(KGS05a) | 39.883 | 44.979 | 36.897 | 44.976 |
| s(PLZ1):by_plz | 806.171 | 750.318 | 813.616 | 750.302 |
| s(ln_area) | 7.661 | 7.841 | 8.743 | 8.404 |
| s(ln_plot) | 8.716 | 8.508 | 7.246 | 7.857 |
| s(age) | 7.877 | 7.879 | 8.541 | 8.720 |
| s(quarter) | 17.704 | 17.703 | 17.690 | 17.703 |

Table 3-5: Mode fitting result

**edf** : corresponds to smoothing parameter ($\lambda$), in other words estimated wiggleness. $Edf > 8$ means highly non-linear curve, edf = 1 means straight line.

The below plots shows the change of the spline functions with regards to variable. Generally speaking, the more age, the function *s(age)* decrease, The decrease is less pronounced when it increases from 100 to 130. Shade area represents the confidence interval.

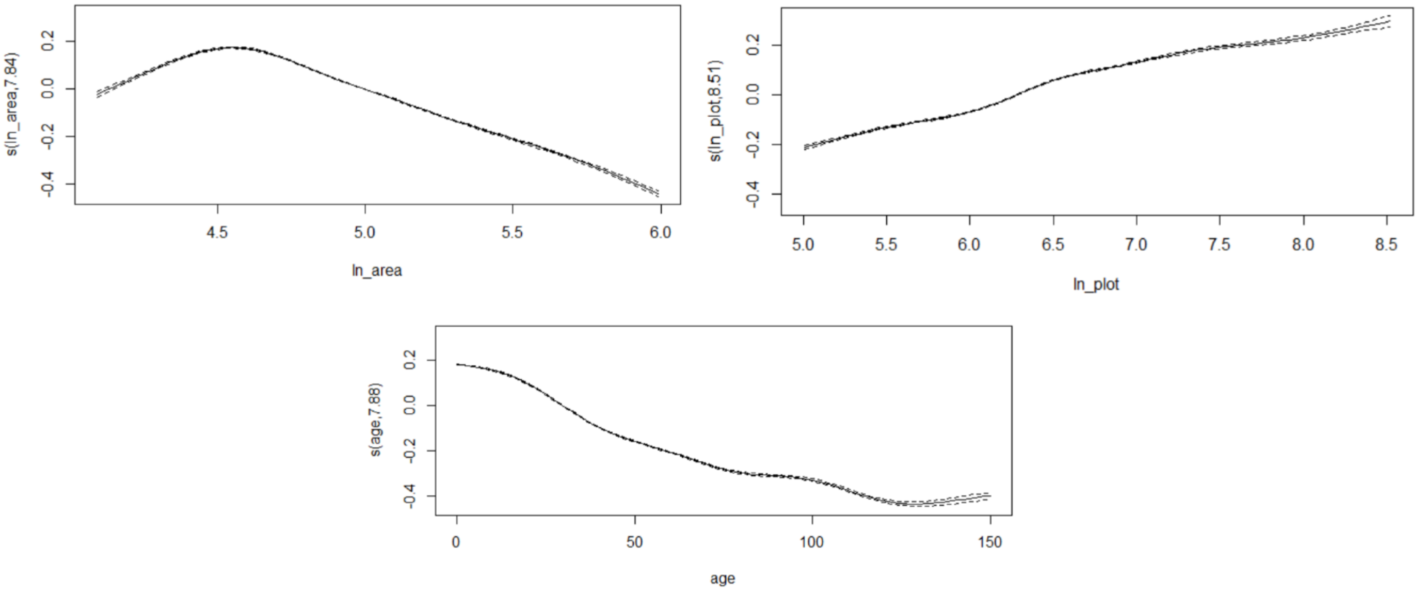


Figure 3-4: REML / TP Partial effect of variables

*Gam.check function in mgcv package* enables to make validation easily for GAM models with regards to graphical steps. Estimated value and Response value comparison is good indicator when assessing the mode performance in general aspect.

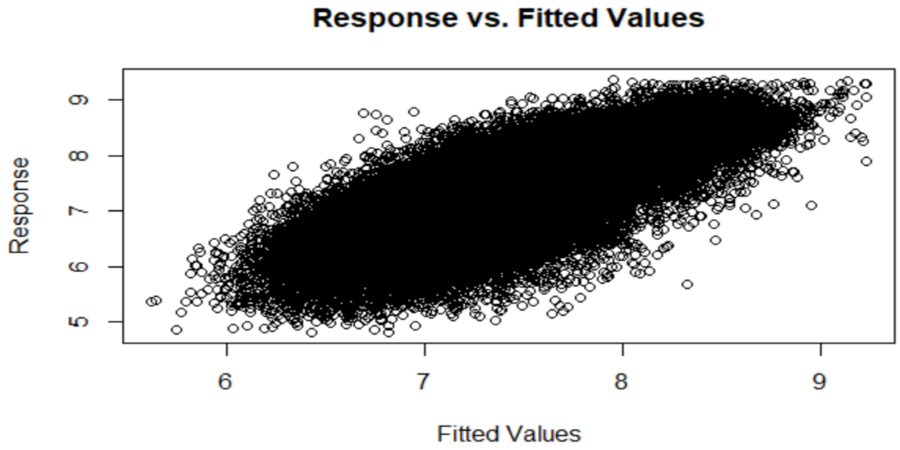


Figure 3-5: Response vs Fitted Values for GAM

#### 3.4.1.1. Normality Check

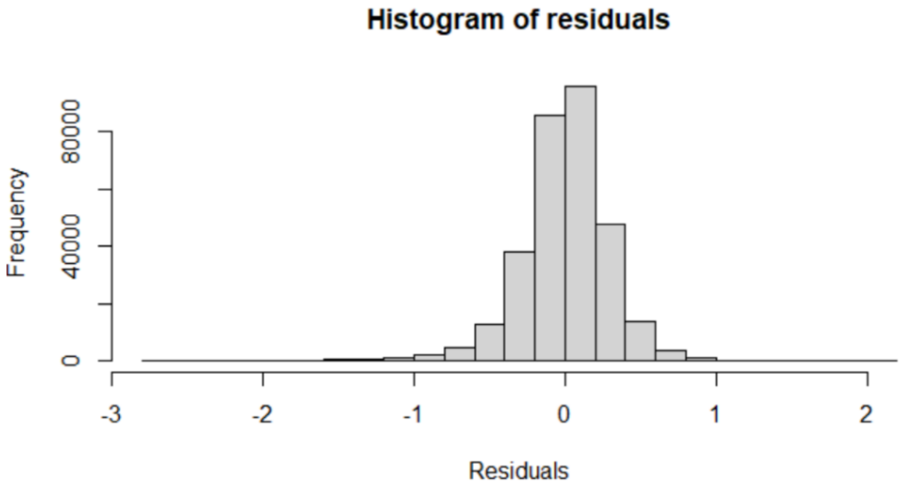


Figure 3-6: Residual Distribution

.

Jarque-Bera test is conducted in order to check the distribution of residual is whether normal or not. It is a goodness-of-fit test that simply aims to match the skewness and kurtosis of data with normal distribution.

$$H_0 : The\ dataset\ is\ from\ a\ normal\ distribution$$

$$H_1 : Otherwise$$

Ontained highly low value indicates that null hypothesis is rejected. The residual is not normally distributed.

#### 3.4.1.2. Homogeneity Check

Standardized Residual should spread randomly everywhere at Residual and Fitted Value Graph.

Bartlett's test is used to test this assumption that variances are equal across data through groups data input.

The data is separated to 4 groups through K means clustering and residual is set as input for Bartlett's test.

$$H_0 : Variance\ (\ \sigma 2\ )\ is\ equal\ across\ all\ groups$$

$$H_1 : Variance\ is\ not\ equal\ across\ all\ groups$$

The obtained P-Value extremely lower than significance level (0.05), so the null hypothesis is rejected. Even though below graph indicates that there is no clear correlation between predicted price and residual, residual distribution depends on object characteristics such as age, furnishing, ln_plot etc.

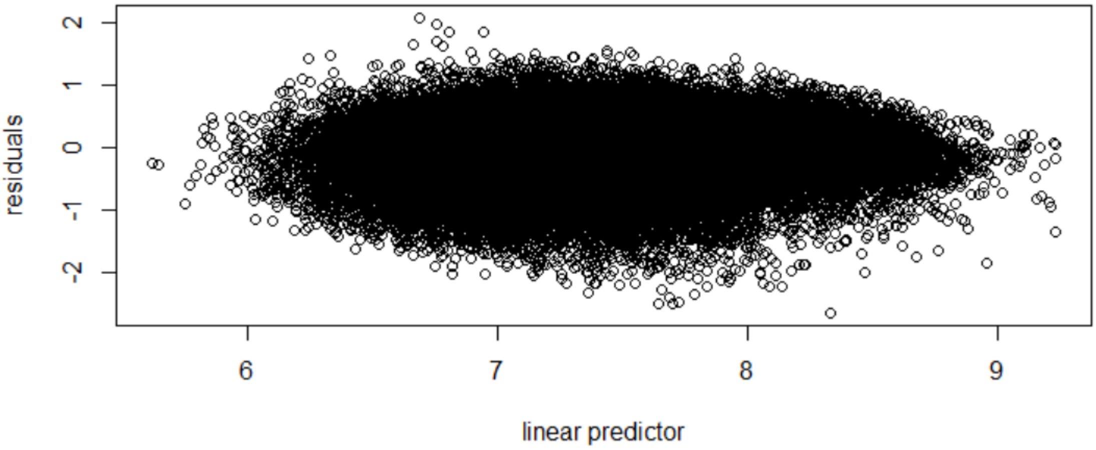


Figure 3-7: Residual Fitted Value Comprasion

### 3.4.1.3. Spatial Autocorrelation for Residual

Spatial Autocorrelation is an inferential statistical method that delivers the interpration of possible correlation for geographically near objects with any selected feature value.

In this report, lon / lat is selected as spatial data input . Fitted model residual value is considered as feature value.

The most well known method for spatial autocorrelation is the Moran's I Test. It was developed by Patrick Alfred and Pierce Moran. For the Moran's I statistic, the null hypothesis states that the attribute being analysed is randomly distributed among the features.

The residual values are randomly assigned to the lon/lat, and the Moran's I is computed. This is repeated several times to establish a distribution of expected values. The observed value of Moran's I is then compared with the simulated distribution to see how likely it is that the observed values could be considered a random draw.

Appendix 9.9 Result part shows that

- The Moran's I value is 0.14, which is relatively small. The Index is bounded by -1 and 1. It indicates that even the result is statistically significant, correlation is not strong.

- Significant result:

$$H_0 = Residuals\ are\ randomly\ assigned\ for\ each\ unique\ spatial\ input\ (lon, lat)$$

$$H_1 = Residuals\ are\ correlated\ with\ each\ unique\ spatial\ input\ (lon, lat)$$

Obtained P value , < 0.01, is lower than our alpha level of 0.05. In this case, it is concluded that there is a significant global spatial autocorrelation even though correlcation is not strong.

### 3.4.2. Price Trend for Model

GAM Model has been constructed with random effect of quarter and data was limited with advertised year at 2014. Analyzing coefficient of quarter spline enables to find out general price trend in overall from 2014 to first quarter 2020.

Extracted spline coefficients aggregated and log effect is removed.

*exp(qu$eff_year + qu$eff_quarter)*

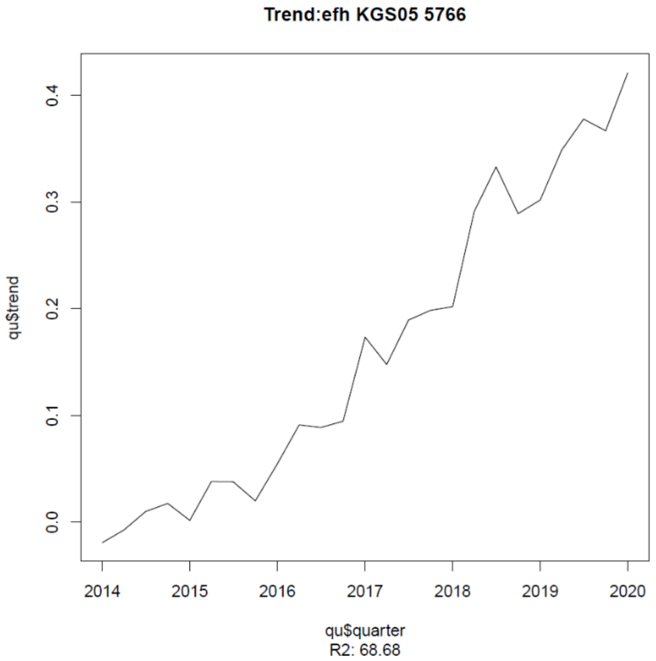


Figure 3-8: Market price development

### 3.4.3. Model Preparation for API

The GAM model for each region is fitted even though this report only covers the detail of only one for only one object type and one census track level. More than 2000 Model has been generated basis on GAM model with postal code effect and reflecting property characters in prediction.

Having high amount of model cause the problem that excessive memory usage on server side. Therefore, model should be cleaned from all the details that doesn't need to be loaded for prediction services.

One cleaning function designed for it and run through all models before loading into API services.

%60 percent of memory gained back by removing below objects from model result. Now the model that is run through API cannot be used for analysis but it only serves for prediction model.

*Residuals, fitted.values, family, linear.predictors, weights, prior.weights, formula, pred.formula, offset, var.summary, attr(op$terms,".Environment")*

# 4. MODEL DEPLOYMENT

Obtained raw data from German Partner is enriched with geocoded information. Even if geocoding data is not extensively used yet, it is highly important for comparison value assessment for real estate. In this report comparison assessment is mentioned on future works section.

Enriched Data is cleaned, transformed and imputed and splitted for each region in way of still having various postal code information and GAM model is empowered with random effect. The fitted models are stripped so that these are ready for deployment. However, there still need to be done improvements in delivering the model to the user. Necessary improvements and challenges are listed below,

1. Data Pipeline, maintainable and scalable development environment for newly coming data.
2. Backend Data Storing, the data that has been obtained as .sav file needs to be save database after preparation steps are completed.
3. Scalable REST API Services, continuously running API services collecting user input from UI, allow them run through right model.
4. Front End UI development, collection of all user inputs and make it possible to be called rest api services and respond with model result.

## 4.1. Data Pipeline

As the data cumulatively increasing every month regularly, exhaustive effort should be presented by model team. High computation power need, repetitive code checking prevents us from doing model improvement and auto deployment. Therefore, the code is structured using *R Drake Package*. This package basically isolates every step of pipeline and allow to increase computational power.

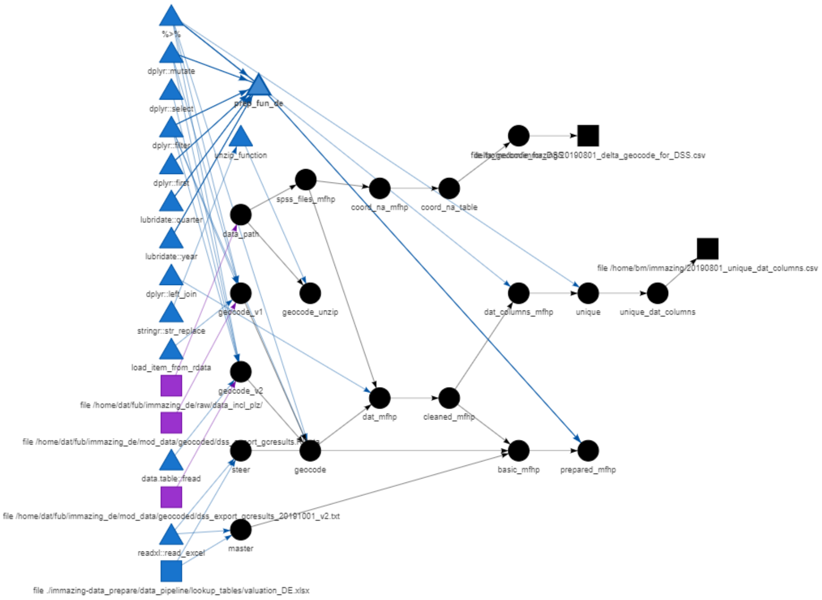
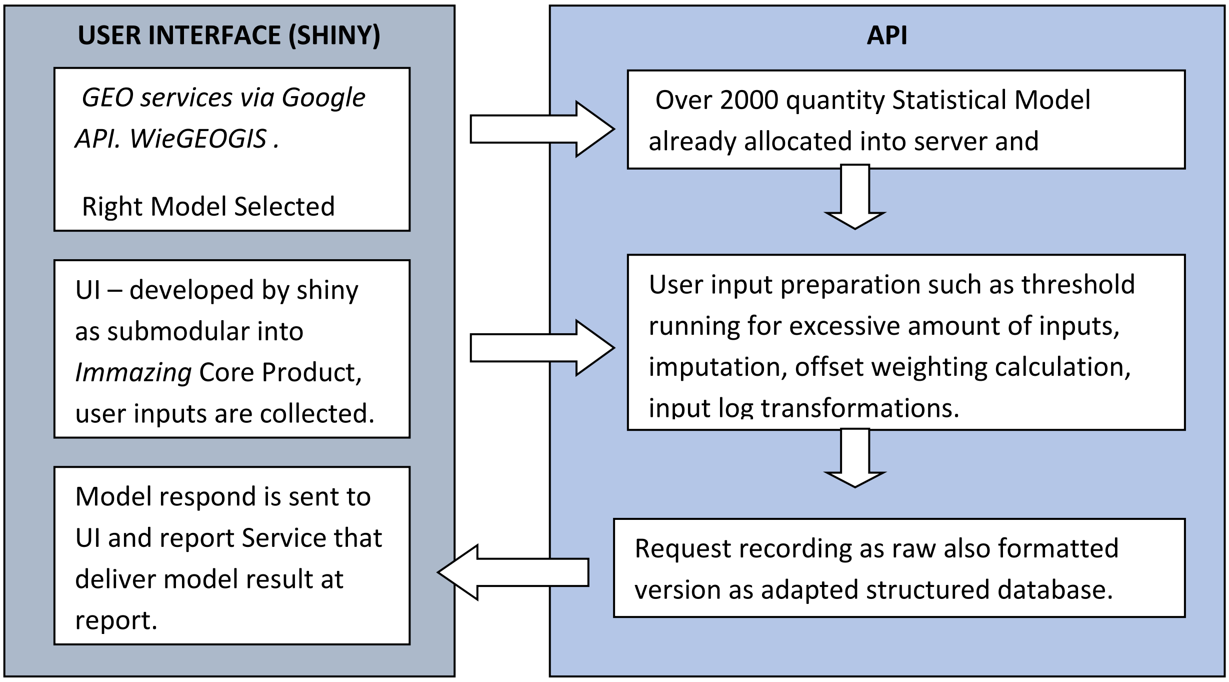


Figure 4-1: Drake Data Pipeline Diagram

## 4.2. Rest Api

Rest Services are positioned between model file as saved as *.RDS* file and UI that is developed on www.ds-s.at/immazing. Developed with R programming Running on Container that is already orchestrated through Kubernetes.



USER INTERFACE (SHINY)

*GEO services via Google API. WieGEOGIS .*

Right Model Selected

UI – developed by shiny as submodular into *Immazing* Core Product, user inputs are collected.

Model respond is sent to UI and report Service that deliver model result at report.

API

Over 2000 quantity Statistical Model already allocated into server and

User input preparation such as threshold running for excessive amount of inputs, imputation, offset weighting calculation, input log transformations.

Request recording as raw also formatted version as adapted structured database.

Figure 4-2: API UI Interaction

## 4.3. User Interface

*Immazing is* the product that has been developed with mainly R programming using *Shiny* Package. Latest shiny and effective R packages are constantly implemented. The Germany real estate assessment product is delivered as part of QUICK product.

QUICK product, as different from PRO product only returns hedonic price assessment. Even though hedonic price assessment is not directly preferred estimation method in real estate market, creates basis for develop widely acceptable methods such as comparison method.

Real Estate Properties, Location and Type inputs are enough to run nonparametric hedonic price model developed with GAM. The user directly sees the predicted price changes as the update the input variable through user interface.

# 5. RESULTS AND DISCUSSION

As the objective of study, different type of GAM models are performed regarding spline construction methodology and smoothing parameter selection method. Due to lack of ability of selection the knot amount by cubic spline, knot number is specified for Cubic Regression Spline.

In model selection MSE value is considered.

| MODEL | Spline Method | Smoothing Parameters Selection Method | Knots Amount Selection | Training Dataset<br>MSE | Test Dataset<br>MSE |
|---|---|---|---|---|---|
| GZLM | - | - | | 0.3264689 | 0.356508 |
| GAM | Thin Plate | REML | Auto | 0.2927631 | 0.292505 |
| GAM | Thin Plate | GCV | Auto | 0.2926804 | 0.292525 |
| GAM + XGBOOST | Thin Plate | REML | Auto | 0.1619583 | 0.2501415 |
| GAM + XGBOOST | Thin Plate | GCV | Auto | 0.1618962 | 0.2501837 |
| GAM | Cubic Regression | REML | k = 10 | 0.2927837 | 0.2925077 |
| GAM | Cubic Regression | GCV | k = 10 | 0.2926975 | 0.2925312 |
| GAM + XGBOOST | Cubic Regression | REML | k = 10 | 0.1614372 | 0.2500327 |
| GAM + XGBOOST | Cubic Regression | GCV | k = 10 | 0.1614928 | 0.2504123 |

Table 5-1: GAM Model Results

The models were trained with 300.000 observations also be tested with 70.000 observation. As shown above, different selection of parameter and spline fitting method is not impactful inputs for reducing MSE value for GAM model selection in our data set. Therefore Any model that is ensembled with Xgboost can be selected. For validation GAM + XGBoost, thin plate model is selected. REML or GCV, TP model.

GAM + XGBOOST - Thin Plate – REML model test dataset absolute error graph shows that %10 percent of data has 0.45 – 2.65 residual range. %90 percent of data has normal distributed residual.

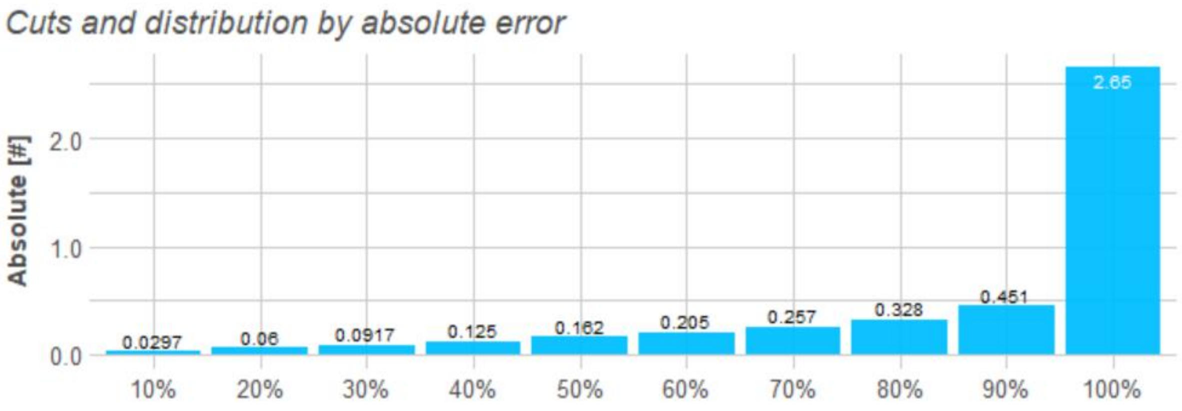


Figure 5-1: Test Dataset error distribution

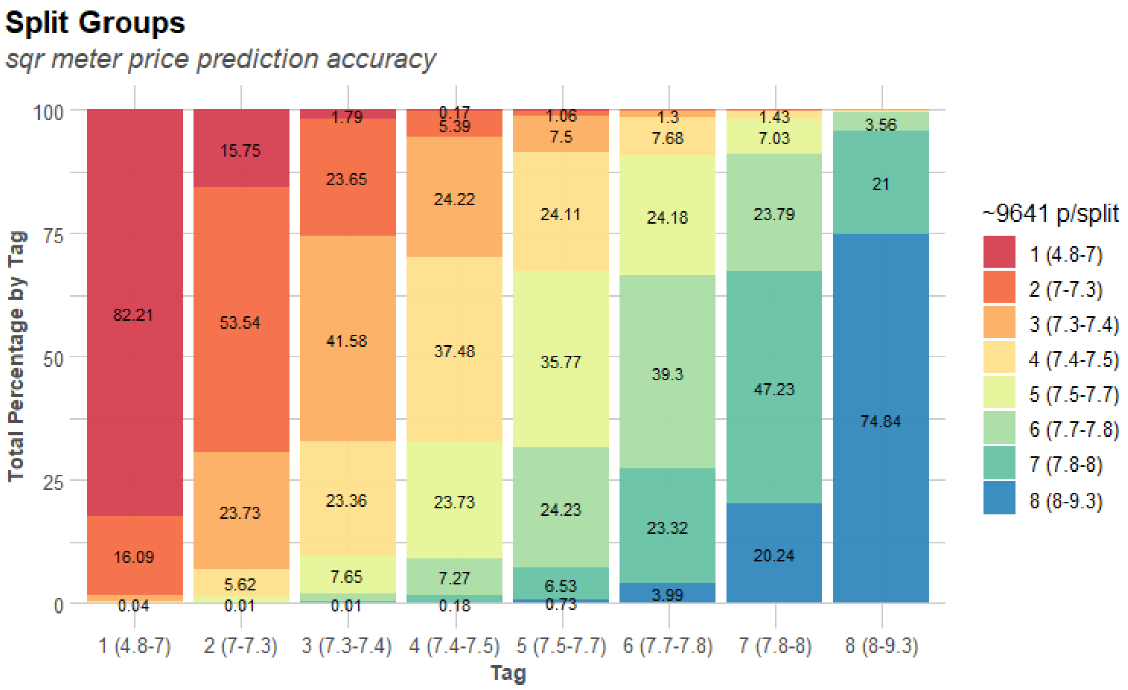


Figure 5-2: Test Dataset Price Accuracy Percentage

Figure 3-7 presents that high and low per m$^2$ price most likely estimated correctly. On the other hand, average m$^2$ prediction success got low until 35.77%.

As result of obtained MSE value, any ensemble model with XGboost can be used at API. Below formulas belong respectively, GAM + XGBOOST - Thin Plate – REML and GAM + XGBOOST – Cubic Regression – REML.

*dhh* and *loggia* categorical variables are eliminated from below model due to being insignificant variables. Loggia input has negative impact at price although expected effect should be positive.

Offset_condition input makes it possible to change the estimated price even though the input is not considered at formula such as swimming pool.

Thin plate Regression Formula:

$$\begin{gathered} lnp\_qm \sim s(KGS05a, bs = "re") + s(PLZ1, bs = "re", by = by\_plz) + \\ s(ln\_area) + s(ln\_plot) + s(age) + year\_fac + s(quarter, \\ bs = "re") + offset(offset\_cond) + gar + stell + balc + garten + \\ cell + wintergarten + zentral\_fern\_etage + Rating + \\ saniert + villa + efh\_bp + ausstattung\_fac + rh \end{gathered}$$

Cubic Spline Regression Model:

$$\begin{gathered} lnp\_qm \sim s(KGS05a, bs = "re") + s(PLZ1, bs = "re", \\ by = by\_plz) + s(ln\_area, bs = "cr", k = 10) + s(ln\_plot, \\ bs = "cr", k = 10) + s(age, bs = "cr", k = 10) + \\ year\_fac + s(quarter, bs = "re") + offset(offset\_cond) + \\ gar + stell + balc + garten + cell + wintergarten + \\ zentral\_fern\_etage + Rating + saniert + villa + efh\_bp + \\ ausstattung\_fac + rh \end{gathered}$$

(Appendix 9.7)

# 6. CONCLUSIONS

First prototype of non-parametric statistical model that references the hedonic price theory for Data Science Service GMBH has been developed and bring it into production and became ready to use through API for German Market. The preferred statistical model was the GAM Model that DSS is specialized at Austria Real estate market.

Even though GAM has advantage of capturing the shape of relationship without prejudging the issue by choosing a parametric form, however, one of the main disadvantages is very sensitive with gaps in the data and outlier. Therefore, strict outlier treatment was done in model preparation step such as real estate advertised year limited with 2014.

Hierarchal spatial data such as postal code, state was treated with 2 level of census track random effect. Time input also is also categorized as quarter and included GAM model with random effect.

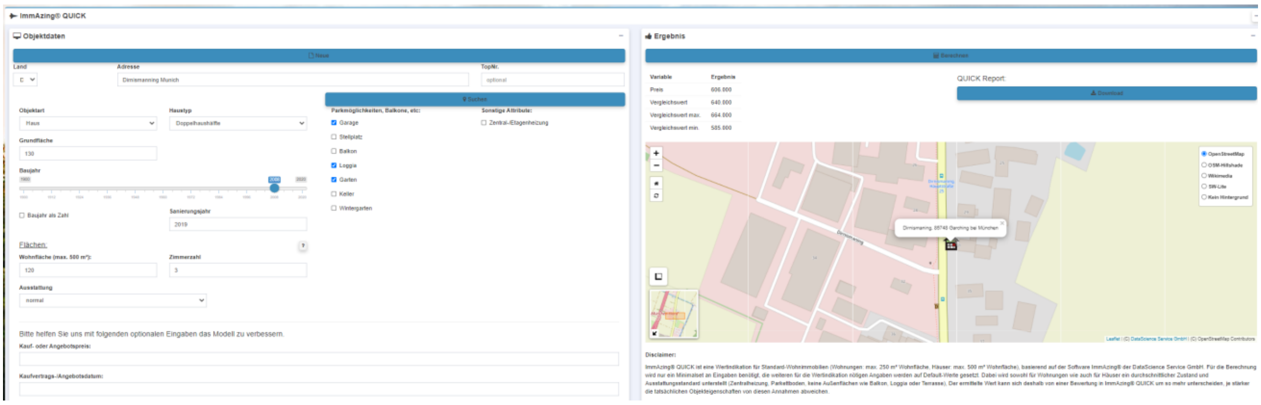
Highly correlated non-linear continuous variables are regressed with smoothing functions. It makes possible to increase degree of freedom without losing interpretability of model. (Appendix 9.4.1.)

GAM model was gained sensitivity with offset values. These empirical values made the model sensitive also different independent variables.

Data was trained with 4 different GAM model and GLM model. GAM models were diversified regarding smoothing function (spline) selection methodology and smoothing parameter selection methodology. The most well-known spline selection methods, thin plate and cubic regression spline selection methods; commonly used smoothing parameter selection methods, generalized cross validation (GCV) and maximum likelihood (REML) are preferred. The results that was obtained from 4 different model proved that even if spline complexity increased, the penalized maximum likelihood didn't allow overfitting whereas complexity decreased low smoothing parameter didn't allow underfitting. Another conclusion was the different method selection was not impactful to increase fitting.

GAM Model is ensembled with gradient boosting framework. Train dataset residuals are fitted with XGboost and MSE decreased 15% for test dataset.

The first version model was run through API services and is brought into live!



Land: D
Adresse: Dirnismanning Munich
TopNr.: optional

Objektart: Haus
Haustyp: Doppelhaushälfte
Grundfläche: 130
Baujahr: 1900 – 2008 – 2020
☐ Baujahr als Zahl
Sanierungsjahr: 2019

Parkmöglichkeiten, Balkone, etc:
☑ Garage
☐ Stellplatz
☐ Balkon
☑ Loggia
☑ Garten
☐ Keller
☐ Wintergarten

Sonstige Attribute:
☐ Zentral-/Etagenheizung

Flächen:
Wohnfläche (max. 500 m²): 120
Zimmerzahl: 3
Ausstattung: normal

Bitte helfen Sie uns mit folgenden optionalen Eingaben das Modell zu verbessern.
Kauf- oder Angebotspreis:
Kaufvertrags-/Angebotsdatum:

| Variable | Ergebnis |
|---|---|
| Preis | 606.000 |
| Vergleichswert | 640.000 |
| Vergleichswert max. | 664.000 |
| Vergleichswert min. | 585.000 |

QUICK Report:

Dirnismaning, 85748 Garching bei München

Disclaimer:
ImmAzing® QUICK ist eine Wertindikation für Standard-Wohnimmobilien (Wohnungen: max. 250 m² Wohnfläche, Häuser: max. 500 m² Wohnfläche), basierend auf der Software ImmAzing® der DataScience Service GmbH. Für die Berechnung wird nur ein Minimalset an Eingaben benötigt, die weiteren für die Wertindikation nötigen Angaben werden auf Default-Werte gesetzt. Dabei wird sowohl für Wohnungen wie auch für Häuser ein durchschnittlicher Zustand und Ausstattungsstandard unterstellt (Zentralheizung, Parkettboden, keine Außenflächen wie Balkon, Loggia oder Terrasse). Der ermittelte Wert kann sich deshalb von einer Bewertung in ImmAzing® QUICK um so mehr unterscheiden, je stärker die tatsächlichen Objekteigenschaften von diesen Annahmen abweichen.

Figure 6-1: Screenshot of Immazing Germany Quick Product

# 7. LIMITATIONS AND RECOMMENDATIONS FOR FUTURE WORKS

The project aim was to make it live a GAM model for Germany Market. Empirical data preparation that basis on concrete market experience in Austria enables to raise a GAM model for Germany.

However, Model is groomed intensively for outliers. It is known that it increases model fitting performance, nonetheless it causes to return unsatisfied predicted result for certain real estate such as the house located next to lake. Even the statistical model is currently accessible for user, still not ready for promoting to our prospect customers.

Model Validation results shows that the raw data should be extensively analyzed and cleansed. Obtained residuals with thin GAM model (thin plate regression spline, generalized cross validation parameter selection method) doesn't meet any GZLM and GAM assumption.

The amount of model, more than 2000, in case of only using one census track level, allocates 60 GB memory on server. It is not only made impossible to allow to user run one model in parallel way also limits the model improvement process in terms of usage of development server at the same time. In this study 2 census track level is used *(KG05 and PLZ1)* to prevent model from data losing.

Therefore, reducing amount of model also by randomizing effect of region will make the model more usable and easier to improve. Moreover, model studies will be more impactful for all Germany.

Macro Statistical / Environmental Variables are not considered into model yet such as distance to metro station, air quality, to name a few. These variables show great impact on Austria General Additive Model.

Data losing due to not well geocoded will cause to not to be selected most convenient property for comparison-based method. Comparison Based method, finding price multiple for the properties based on some feature of the property which derives the property's value. It involves dividing the property value by say its covered area, number of apartments. Therefore, synthetic geocoded should be imputed by considering similar objects with regards to their price at same postal code.

Even though GAM allows us to deliver crystal box model, black box algorithms should be run at least in performance assessment. Beyond that, satellite image decomposition for learning or interior picture can be used for model learning.

## 8. BIBLIOGRAPHY

Austria Real Estate Price Index. Retrieved from https://www.oenb.at/isaweb/report.do?lang=EN&report=6.6

Anglin, P., M., & Gencay R. (1996). Semiparametric Estimation of a Hedonic Price Function, Journal of Applied Econometrics, Vol. 11, No. 6, 633-648. Retrieved from https://onlinelibrary.wiley.com/doi/epdf/10.1002/%28SICI%291099-1255%28199611%2911%3A6%3C633%3A%3AAID-JAE414%3E3.0.CO%3B2-T

Can, A. (1998). GIS and spatial analysis of housing and mortgage markets. Journal of Housing Research, 61-86.

Clapp, J., M. (2004). A Semiparametric Method for Estimating Local House Price Indices, Real Estate Economics 32(1), 127-160.

Ekeland, I. (2004). Identification and Estimation of Hedonic Models, 2004, Journal of Political Economy 112(S1), 60. Retrieved from https://discovery.ucl.ac.uk/id/eprint/12718/

Follain, R. & Jimenez, E. (1985). Estimating the demand for housing characteristics, Volume 15, Issue 1, February, 77-107.

De Boor (1978). A Practical Guide to Spline January Mathematics of Computation Volume 27, 149.

Des Rosiers, F., Lagana, A., Theriault, M. & Beaudoin, M. (1996). Shopping centres and house values: An empirical investigation, Journal of Property Valuation & Investment, vol. 14, 41-62.

Duchon (1977). Thin Plate Splines. Retrieved from http://web.mit.edu/~r/current/arch/i386_linux26/lib/R/library/mgcv/html/smooth.construct.ds.smooth.spec.html

Gelman, A. & Hill, J. (2007). Data analysis using regression and multilevel/hierarchical models. Cambridge: Cambridge University Press.

Goodman, A. & Thibodeau, T. (2003). Housing market segmentation and hedonic prediction accuracy, Journal of Housing Economics, 12, 181-201.

Goodman, A. (1998). Andrew Court and the Invention of Hedonic Price Analysis, 291-298.

Goldstein, H. (2011). Multilevel statistical models. Chichester, Wiley.

Griliches Z. (1961). Hedonic Price Indexes for Automobiles: An Econometric of Quality Change. Retrieved from https://www.nber.org/system/files/chapters/c6492/c6492.pdf

Hastie, T.J. & Tibshirani, R. (1990). Generalized Additive Models.

Hastie,T.J. & Tibshirani, R. (1986). Generalized additive models (with discussion). Statist. Sci., 297–318. Retrieved from https://web.stanford.edu/~hastie/Papers/gam.pdf

Hill, J. (2013). Hedonic Price Indexes for Residential Housing: A Survey, Evaluation and Taxonomy. Retrieved from https://www.researchgate.net/publication/263259181_Hedonic_Price_Indexes_for_Residential_Housing_A_Survey_Evaluation_and_Taxonomy

Kestens Y., Thériault M., Rosiers, F. (2006). Heterogeneity in hedonic modelling of house prices: Looking at buyers' household profiles, February 2006 Journal of Geographical Systems , 61-96.

Kain, J. & Quigley, J. (1970). Measuring the Value of Housing Quality. Journal of the American Statistical Association, 65, 532-548.

Mason, C., & Quigley, J., M. (1996). Non-parametric hedonic housing prices. Retrieved from https://escholarship.org/uc/item/2qz3t9n2

Malpezzi, S. (2003). Hedonic pricing models: A selective and applied review. In T. O'Sullivan & K. Gibb (Eds.), Housing economics and public policy, 67–89.

McMillen, D., P. & Redfearn, C. L. (2010). Estimation and Hypothesis Testing for Nonparametric Hedonic House Price Functions, Journal of Regional Science, Vol. 50, No. 3, 712-733.

McCullagh, P. and Nelder, J.A. (1989). Generalized Linear Models. 2nd Edition, Chapman and Hall, London, 8.

Meen, G., (2001). Modelling Spatial Housing Markets.

Rodriguez, M. & Sirmans, F. (1994). Quantifying the Value of a View in Single Family Housing Markets, The Appraisal Journal, 600–03.

Sirmans, G., Zietz, E., David A. (2006). The Composition of Hedonic Pricing Models.

Sirpal, R., (1994). Empirical Modeling of the Relative Impacts of Various Sizes of Shopping Centers on the Values of Surrounding Residential Properties, Journal of Real Estate Research, American Real Estate Society, vol. 9(4), 487-506.

Wallace, N. (1991). The Construction of Residential Housing Price Indices: A Comparison of Repeat-Sales, Hedonic-Regression, and Hybrid Approaches. Retrieved from https://ideas.repec.org/a/kap/jrefec/v14y1997i1-2p51-73.html

Wallace, N. (1996). Hedonic-based price indexes for housing: theory, estimation, and index construction. Retrieved from https://www.frbsf.org/economic-research/files/wallace.pdf

Waugh, F.V. (1928). Quality factors influencing vegetable prices, Journal of Farm Economics,10. 185–196.

Wolfgang B. (2010). Spatial Heterogeneity in Hedonic House Price Models: The Case of Austria.

Wood, S. N. (2004). Stable and efficient multiple smoothing parameter estimation for generalized additive models. Journal of the American Statistical Association 99, 673–686.

Wood, S. N., (2006). An introduction to generalized additive models with R CRC Press.

Wood, S.N. (2000). Modelling and smoothing parameter estimation with multiple quadratic penalties Journal of the Royal Statistical Society: Series B, 62, 413-428.

Wood, S.N. (2012). mgcv: Mixed GAM Computation Vehicle with GCV/AIC/REML smoothness estimation.

# 9. APPENDIX

## 9.1. MGCV Package Usability

**Splines:** Does not support loess or smoothing splines, but supports a wide array of regression splines (P-splines, B-splines, thin plate splines, tensors) + tensors
**Parametric terms**: Supported, and you can penalize or treat as random effects, PIRLS, Finds smoothing parame.
**Variable selection:** Shrinkage
**Optimization:** PIRLS
**Selecting smoothing:** Finds smoothing parameters by default. Supports, both REML and GCV
**Large datasets:** Special bam function for large datasets.
**Missing values:** No special treatment. Omits observations missing values
**Multidimensional:** Supported with tensors and thin plate splines
**Model diagnostics:** Standard GAM diagnostics + the concurvity measure which is a generalization of collinearity

## 9.2. Offset Inputs

| | name | wert | efh |
|---|---|---|---|
| 1 | cond3_haus_gut | 0.03 | 1 |
| 2 | cond1_haus_schlecht | -0.05 | 1 |
| 3 | cond3_gut | 0.02 | 0 |
| 4 | cond1_schlecht | -0.04 | 0 |
| 5 | sauna | 0.02 | 1 |
| 6 | barrierefrei | 0.02 | 1 |
| 7 | alarmanlage | 0.01 | 1 |
| 8 | swimmingpool | 0.02 | 1 |
| 9 | zentral | 0.01 | 1 |
| 10 | abstellraum | 0.01 | 1 |
| 11 | wintergarten | 0.01 | 1 |
| 12 | zentral_fern_etage | 0.01 | 1 |
| 13 | parkett | 0.01 | 1 |
| 14 | garage | 0.01 | 1 |
| 15 | stell | 0 | 1 |
| 16 | villa | 0.02 | 1 |
| 17 | cell | 0.01 | 1 |
| 18 | garten | 0.01 | 1 |

Table 9-1: Model Offset Values

## 9.3. DATA IMPUTATION R CODE

```
  hyper_grid_2 <- expand.grid(
    mtry       = seq(1, 20, by = 1),
    num.trees  = seq(50, 500, by = 50),
    OOB_RMSE  = 0)
  default_model <-  ranger::ranger(formula, training , splitrule="variance", mtry=7,
                    importance = "impurity", num.threads = 6 ,num.trees = 200, write.forest = T)
  default <- sqrt(default_model$prediction.error)
  for(i in 1:nrow(hyper_grid_2))  {
    impute_model <-  ranger::ranger(formula, training , splitrule="variance",
mtry=hyper_grid_2$mtry[i], importance = "impurity", num.threads = 6 ,num.trees =
hyper_grid_2$num.trees[i], write.forest = T, respect.unordered.factors = "order")
hyper_grid_2$OOB_RMSE[i] <- sqrt(impute_model$prediction.error)
}
  hyper_grid_2 %>%
    arrange(OOB_RMSE) %>%
    mutate(perc_gain = (default - OOB_RMSE) / default * 100) %>%
    head(10)
  hyper_grid_2 %>%
    dplyr::arrange(OOB_RMSE) %>%
    head(10)
  hist(hyper_grid_2$OOB_RMSE, breaks = 20)
  impute_model$variable.importance %>%
    # tidy() %>%
    dplyr::arrange(desc(x)) %>%
    dplyr::top_n(25) %>%
    ggplot(aes(reorder(names, x), x)) +
    geom_col() +
    coord_flip() +
    ggtitle("Top 25 important variables")
  pred <- predict(impute_model$forest, data[is.na(data[[var]]),] )
  data[, paste0(var,"_imputed")] <- data[,var]
```

## 9.4. Fitted Model Data Exploration

### 9.4.1. Correlation Table

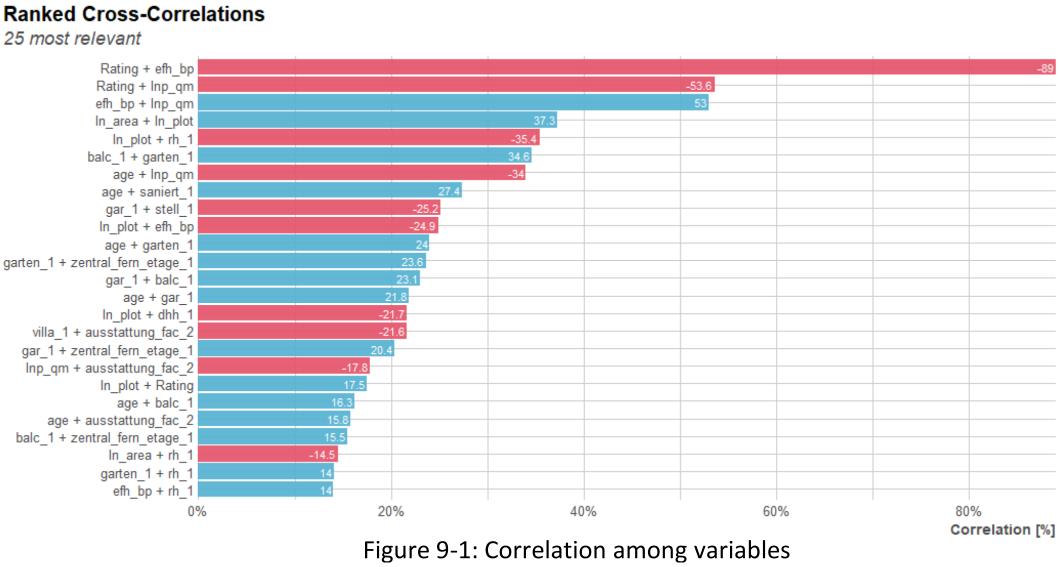


Figure 9-1: Correlation among variables

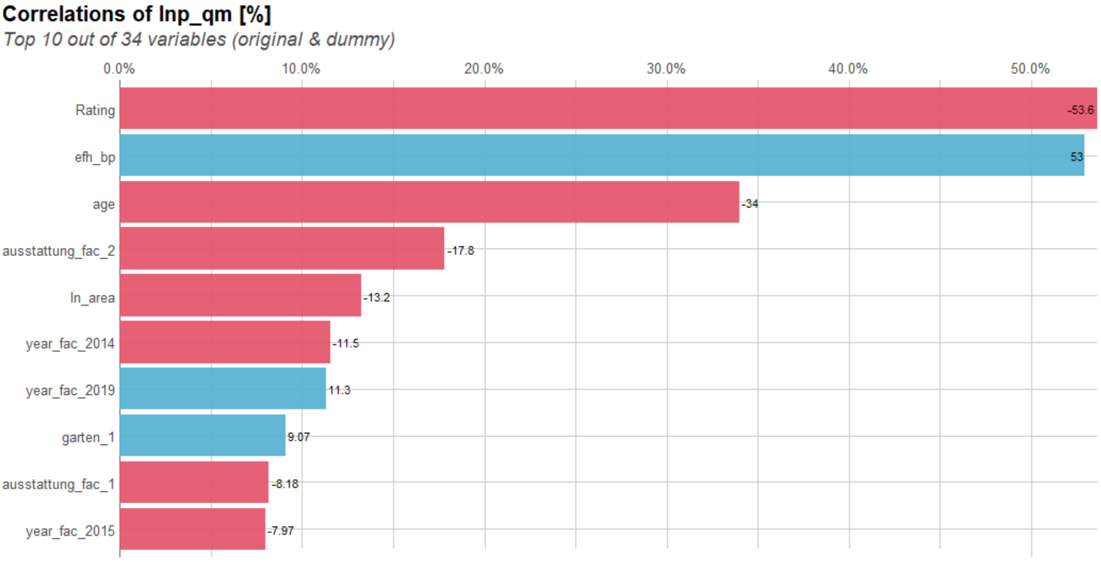


Figure 9-2: Correlation between variables log m² price

### 9.4.2. Density Functions

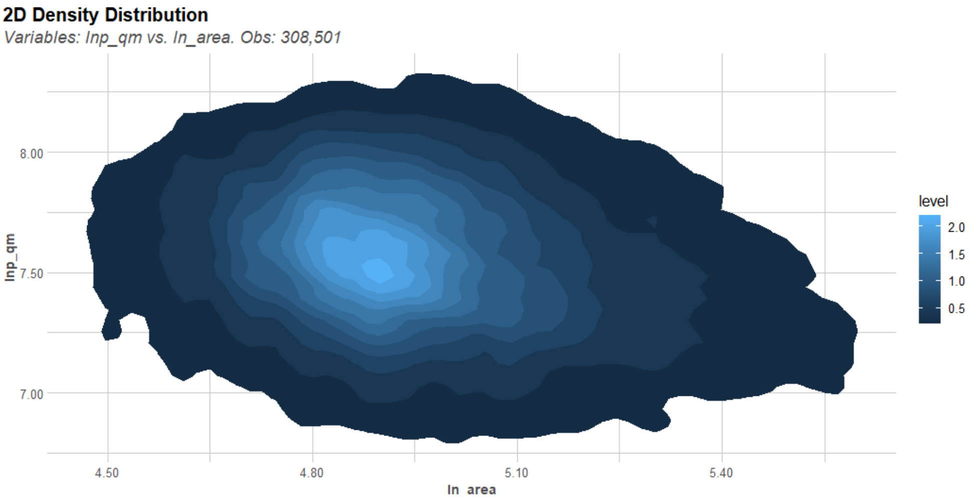


Figure 9-3: log m² price – log area density distribution

### 9.4.3. Distribution for Furnshing

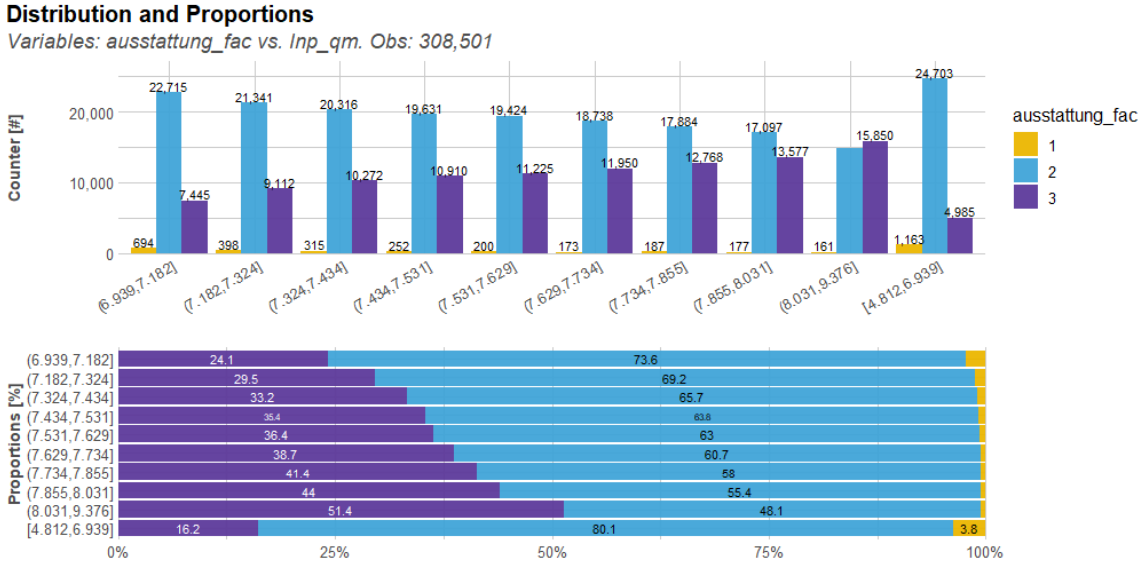


Figure 9-4: – furnishing effect on log m² price

## 9.5. Base Formel Preparation

```
nam_vz <- list(); vz <- list()
nam_vz_base <- c(
   "gar", "stell",
   "balc", "garten", "cell", "loggia", "wintergarten",
   "zentral_fern_etage", "Rating”,"saniert")
sign_base <- c(1,1, 1,1,1,1,1, 1,-1, 1)
# Specific EFH
nam_vz[["efh"]] <- c(nam_vz_base,
                "villa",
                "efh_bp")
vz[["efh"]] <- c(sign_base, 1, 1)
# Collect in list
sign_vz <- list()
for (nam in names(vz)) {
   print(nam)
   sign_vz[[nam]] <- assign_signs(nam = nam_vz[[nam]], vz = vz[[nam]])}
nam_ovz_base <- c("ausstattung_fac")
nam_ovz <- list()
nam_ovz[["efh"]] <- c(nam_ovz_base,
                 "rh", "dhh")
vars_select_all <- list()
for (nam in names(vz)) {
   print(nam)
   vars_select_all[[nam]] <- c(nam_vz[[nam]], nam_ovz[[nam]])
}
```

## 9.6. XGBOOST – RESIDUAL MODEL

```
model_residuals_efh <- function(data){
   if (is.null(data$residual)){
      stop("Add the residuals first!!!") }
   cols <- c("ln_plot", "age",  "ln_area", "quarter", "efh_bp", "PLZ1", "Rating",
          "zentral_fern_etage", "ausstattung_fac", "gar", "cell", "garten", "saniert", "balc")
   covariates <- data[ , cols]
   data <- data[!is.na(data$Rating) , ]
   model <- xgboost::xgboost(data = data.matrix(covariates), label = data$residual , max_depth = 8,
gamma=0, colsample_bytree = 1, min_child_weight = 1, subsample = 1 , eta = 0.1, nthread = 8,
nrounds = 2000,  objective = "reg:squarederror" , verbose = 0)
   return(model)
}
For (i in 1:length(d1$lnp_qm)) {
   if(d1$lnp_qm[i] - m1$model[i,"lnp_qm"] != 0) {
      print(i)
      d1 <- d1[-c(i),]
      break;
   }
}
d1$residual <- m1$residual
cols <- c("ln_plot", "age",  "ln_area", "quarter", "efh_bp", "PLZ1", "Rating",
       "zentral_fern_etage", "ausstattung_fac", "gar", "cell", "garten", "saniert", "balc")
covariates <-  data.matrix(d1[ !is.na(d1$Rating), cols])
pred_train_xg <- predict(xg_error_model,covariates)
cols <- c("ln_plot", "age",  "ln_area", "quarter", "efh_bp", "PLZ1", "Rating",
       "zentral_fern_etage", "ausstattung_fac", "gar", "cell", "garten", "saniert", "balc")
covariates <-  data.matrix(test_df[ !is.na(test_df$Rating), cols])
pred_test_xg <- predict(xg_error_model , covariates)
```

## 9.7. SELECTED MODEL OUTPUT

### 9.7.1. GAM Model (GCV / TP)

```
Family: gaussian
Link function: identity

Formula:
lnp_qm ~ s(KGS05a, bs = "re") + s(PLZ1, bs = "re", by = by_plz) +
    s(ln_area) + s(ln_plot) + s(age) + year_fac + s(quarter,
    bs = "re") + offset(offset_cond) + gar + stell + balc + garten +
    cell + loggia + wintergarten + zentral_fern_etage + Rating +
    saniert + villa + efh_bp + ausstattung_fac + rh + dhh

Parametric coefficients:
```

```
                     Estimate Std. Error t value Pr(>|t|)    
(Intercept)         6.972e+00  1.280e-01  54.451  < 2e-16 ***
year_fac2015        5.767e-02  1.533e-02   3.762 0.000169 ***
year_fac2016        1.092e-01  1.534e-02   7.122 1.07e-12 ***
year_fac2017        1.655e-01  1.537e-02  10.771  < 2e-16 ***
year_fac2018        2.366e-01  1.537e-02  15.398  < 2e-16 ***
year_fac2019        3.054e-01  1.537e-02  19.862  < 2e-16 ***
year_fac2020        3.769e-01  2.428e-02  15.528  < 2e-16 ***
gar1                5.127e-02  1.294e-03  39.624  < 2e-16 ***
stell1              1.425e-02  2.246e-03   6.344 2.25e-10 ***
balc1               3.263e-02  1.228e-03  26.566  < 2e-16 ***
garten1             6.626e-02  1.239e-03  53.483  < 2e-16 ***
cell1               1.326e-02  1.132e-03  11.713  < 2e-16 ***
wintergarten1       2.488e-02  2.197e-03  11.327  < 2e-16 ***
zentral_fern_etage1  1.424e-02  1.171e-03  12.161  < 2e-16 ***
Rating             -1.800e-01  1.311e-02 -13.731  < 2e-16 ***
saniert1            1.063e-01  2.328e-03  45.658  < 2e-16 ***
villa1              2.151e-02  3.456e-03   6.224 4.86e-10 ***
efh_bp              1.950e-01  1.593e-02  12.240  < 2e-16 ***
ausstattung_fac2     1.134e-01  4.943e-03  22.933  < 2e-16 ***
ausstattung_fac3     1.670e-01  5.018e-03  33.272  < 2e-16 ***
rh1                -1.948e-02  2.212e-03  -8.804  < 2e-16 ***

---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Approximate significance of smooth terms:
                  edf Ref.df       F p-value    
s(KGS05a)       39.883     52 14309.8   0.598    
s(PLZ1):by_plz 806.171    862   493.3   0.525    
s(ln_area)       7.661      9 32954.4  <2e-16 ***
s(ln_plot)       8.716      9 34728.2  <2e-16 ***
s(age)           7.877      9 88528.6  <2e-16 ***
s(quarter)      17.704     18   128.3  <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

R-sq.(adj) =  0.596   Deviance explained = 59.7%
GCV = 0.08617  Scale est. = 0.085916  n = 308501
```

### 9.7.2. XGBoost Model ( GCV / TP )

```
##### xgb.Booster
Handle is invalid! Suggest using xgb.Booster.complete
raw: 25.8 Mb 
call:
  xgb.train(params = params, data = dtrain, nrounds = nrounds, 
    watchlist = watchlist, verbose = verbose, print_every_n = print_every_n, 
    early_stopping_rounds = early_stopping_rounds, maximize = maximize, 
    save_period = save_period, save_name = save_name, xgb_model = xgb_model, 
    callbacks = callbacks, max_depth = 8, gamma = 0, colsample_bytree = 1, 
    min_child_weight = 1, subsample = 1, eta = 0.1, nthread = 8, 
    objective = "reg:squarederror")
params (as set within xgb.train):
  max_depth = "8", gamma = "0", colsample_bytree = "1", min_child_weight = "1"
, subsample = "1", eta = "0.1", nthread = "8", objective = "reg:squarederror",
validate_parameters = "TRUE"
callbacks:
  cb.evaluation.log()
# of features: 14
niter: 2000
nfeatures : 14
evaluation_log:
    iter train_rmse
       1   0.536201
       2   0.498589
---
    1999   0.161922
```

## 9.8. Descriptive Statistics

| variable | minimum | q1 | median | mean | q3 | maximum |
|---|---|---|---|---|---|---|
| ln_area | 4.094 | 4.787.492 | 4.942 | 4.978.128 | 5.147 | 5.991 |
| Rating | 1.700 | 4.100.000 | 4.600 | 4.556.504 | 5.100 | 7.400 |
| ln_plot | 5.011 | 5.971.262 | 6.282 | 6.292.947 | 6.601 | 8.517 |
| efh_bp | 3.045 | 4.645.640 | 5.066 | 5.008.620 | 5.350 | 9.034 |
| lnp_qm | 4.805 | 7.259.366 | 7.531 | 7.501.498 | 7.791 | 9.376 |
| age | 0 | 0 | 21 | 30.160 | 49 | 170.000 |

Table 9-2: Descriptive Statistics Continuous Variables

| Categorical Variables | | | | | |
|---|---|---|---|---|---|
| gar | 0 | 181567 | saniert | 0 | 361872 |
| gar | 1 | 204060 | saniert | 1 | 23755 |
| stell | 0 | 359503 | villa | 0 | 375583 |
| stell | 1 | 26124 | villa | 1 | 10044 |
| balc | 0 | 247714 | ausstattung_fac | 1 | 4683 |
| balc | 1 | 137913 | ausstattung_fac | 2 | 246027 |
| garten | 0 | 212183 | ausstattung_fac | 3 | 134917 |
| garten | 1 | 173444 | rh | 0 | 350483 |
| loggia | 0 | 377021 | rh | 1 | 35144 |
| loggia | 1 | 8606 | dhh | 0 | 331868 |
| wintergarten | 0 | 360809 | dhh | 1 | 53759 |
| wintergarten | 1 | 24818 | | | |
| zentral_fern_etage | 0 | 238410 | | | |
| zentral_fern_etage | 1 | 147217 | | | |

Table 9-3: Descriptive Statistics Categorical Variables

## 9.9. Model Validation Tests

### 9.9.1. Moran's I Result

```
      Monte-Carlo simulation of Moran I

data:  as.numeric(unique_points$moran_values)
weights: nblist
number of simulations + 1: 100
statistic = 0.14894, observed rank = 100, p-value = 0.01
alternative hypothesis: greater

Neighbour list object:
Number of regions: 29450
Number of nonzero links: 29450
Percentage nonzero weights: 0.003395586
Average number of links: 1
Non-symmetric neighbours list
```

### 9.9.2. Moran's Functions

```
moran_test_fun <- function(data , values){
  if(is.null(data$lon) || is.null(data$lat)){
    stop("Longitude and latitude are required.") }
  data$moran_values <- values
  unique_points <- dplyr::distinct(data,lon,lat, .keep_all=T)
  # Compute neighborhoods based on knn.
  knn <- spdep::knearneigh(sp::coordinates(unique_points[,c("lon","lat")]), longlat = TRUE)
  neib_knn <- spdep::knn2nb(knn)
  # Transform the neighborhoods into an nblist.
  nblist <- spdep::nb2listw(neib_knn)
  set.seed(1234)
  spdep::moran.mc(as.numeric(unique_points$moran_values), nblist, nsim=99)
  # spdep::moran.test(as.numeric(unique_points$moran_values) , nblist)
}
indices <- 1:dim(d1)[1]
d_moran <- d1[indices , c("lon", "lat")]
data <- d_moran[!is.na(d_moran$lon), ]
values <- d1$residual[indices]
```

### 9.9.3. Bartlett Test

```
indices <- sample(1:dim(d1)[1],50000)
d_Bartlett <- d1[indices ,c("lnp_qm", "KGS05a", "PLZ1", "by_plz", "ln_area",
"ln_plot", "age", "ausstattung_fac", "year_fac", "quarter", "offset_cond",
                            "gar", "stell", "balc", "garten", "cell",
"loggia", "wintergarten",
"zentral_fern_etage" , "Rating" , "saniert" , "villa" , "efh_bp" , "rh" ,
"dhh", "residual")]

km.res <- kmeans(d_Bartlett[,c("ln_area", "ln_plot", "age", "ausstattung_fac",
"gar", "stell", "balc", "garten", "cell", "loggia",
"wintergarten","zentral_fern_etage" , "Rating" , "saniert" , "villa" ,
"efh_bp" , "rh" , "dhh")], centers = 4)

d_Bartlett_with_cluster <- cbind(d_Bartlett, cluster = km.res$cluster)

table(d_Bartlett_with_cluster$cluster)
d_Bartlett_with_cluster$cluster <- as.factor(d_Bartlett_with_cluster$cluster)
result = bartlett.test(residual~cluster, d_Bartlett_with_cluster)
print(result)   1     2     3     4
10606 13151  5191 21052
result = bartlett.test(residual~rand_int, d1_with_group)

Result:
Bartlett test of homogeneity of variances
data:  residual by cluster
Bartlett's K-squared = 2232.7, df = 3, p-value < 1.1e-03
```

### 9.9.4. Jarque Bera Test

```
jarque.bera.test(d1$residual)
```

```
        Jarque Bera Test

data:  d1$residual
X-squared = 111290, df = 2, p-value < 2.1e-13
```