University of Wisconsin-Milwaukee

Sheldon B. Lubar School of Business

Assignment for Course:	BUS MGMT 709 – 001 – Predictive Analytics for Managers
Submitted to:	Layth C. Alwan, Ph.D.
Submitted by:	Daniel Kinkel Silva Gebhardt
Date of Submission:	05.20.2022
Title of Assignment:	Project Report

CERTIFICATION OF AUTHORSHIP: I certify that I am the author of this paper and that any assistance I received in its preparation is fully acknowledge and disclosed in the paper. I have also cited any sources from which I used data, ideas of words, whether quoted directly or paraphrased. I also certify that this paper was prepared by me specifically for this course.

Student Signature (can by typed): Daniel Kinkel, Silva Gebhardt

Instructor's Grade on Assignment:

Instructor's Comments:

Contents

1. Introduction & Goals

The present paper includes a professional-grade report on a predictive analytics project that fundamentally examined and predicted house pricing. The real estate markets, all over the world present interesting opportunities for data analysts to analyze and predict where property prices are moving towards. The housing market was surging last year and might continue to do so in the foreseeable future. Experts are still seeing a post-pandemic rebound with steady mortgage rates, job recoveries, and the law of supply and demand all working together to make home prices keep rising. These latest occurrences, as well as the implosion of the housing bubble in 2009, show how dynamic the housing market can be and how difficult it can be to make a reliable prediction. Nevertheless, predicting house prices provides value to help people who plan to buy a house. Having a price estimate or a price range can be very helpful in planning their finances more accurately. In addition, house price predictions are also beneficial for property investors to know the trend of housing prices in a certain location. There are a variety of factors that impact real estate prices. Therefore, it is important to determine the most important factors to create the optimal trade-off between the validity and feasibility of the prediction model. Speaking with experts in the field and by executing further research, the following eight factors seem to be the determinants on which the majority of sources agree:

Neighborhood comps: One of the best indicators of your home’s value is the sale prices of similar homes in your neighborhood that have sold recently. These comparable homes are often referred to as “comps” and most real estate experts rely on comps to estimate your home value.

Location: There are three primary aspects that mainly determine the quality of the location. The quality of local schools, employment opportunities, and proximity to shopping, entertainment, and recreational centers. In addition, a location’s proximity to highways, utility lines, and public transit can all impact a home’s overall value. When it comes to calculating a home’s value, location can be more important than even the size and condition of the house.

Home size and usable space: When estimating a home’s market value, size is an important element to consider, since a bigger home can positively impact its valuation. Livable space is what is most important to buyers and appraisers. Bedrooms and bathrooms are also highly valued. Higher numbers of bedrooms or bathrooms are usually associated with a higher housing price.

Age and condition: Typically, newer homes appraise at a higher value. The fact that critical parts of the house, like plumbing, electrical, the roof, and appliances are newer and therefore less likely to break down, can generate savings for a buyer.

Upgrades and renovations: These can add value to a property, especially in older houses that may have outdated features. However, there is a wide range of possible improvements, and the scope of renovations can vary significantly.

The local market: If there are a lot of buyers competing for fewer homes it's a seller's market. Conversely, a market with few buyers but many homes on the market is referred to as a buyer's market. Buying in a buyer's market allows for more room to negotiate the home's price, timeline, and contingencies in the contract. Together with that, government policies and legislation, including tax incentives, deductions, and subsidies can boost or hinder demand for real estate

Economic indicators: Real estate prices often follow the cycles of the economy and are also highly influenced by the current interest rates. Lower rates bring in more buyers, reflecting the lower cost of getting a mortgage, but also expanding the demand for real estate, which can then drive up prices.

Given that some variables like the interest rate vary over time, we would technically need a panel dataset and a fixed or random-effects model. To limit the complexity in this initial analysis, we focus on housing prices at one point in time which allows for traditional regression analysis. We looked for pre-existing datasets and found one on the popular machine learning website kaggle.com that does include most of the data we are interested in:
The dataset[1] from the popular machine learning website Kaggle provides the following factors: date, price, bedrooms, bathrooms, sqft_living, sqft_lot, floors, waterfront, view, and condition. Besides, the dataset gives us information about where the object is located with street, city and zip code and country.

[1] https://www.kaggle.com/datasets/shree1992/housedata

2. Analysis

2.1 Preprocessing and Splitting

For our analysis, we start by importing the dataset and checking for apparent data quality issues and missing values. Since none are found, we neither have to impute those nor remove the observations and continue our analysis.

As another preprocessing step, we construct an age variable from the ‘year_build’ column. The new continuously scaled variable should allow for easier integration in the regression model. Additionally, we transform the renovated column into a dummy variable that expresses if a renovation has taken place. We considered building a second age-like column but since only a part of the houses was renovated this would result in a lot of missing values.

Next, we select a subset of features that will be used for the subsequent prediction process. In essence, we use most of the available information, with the exception of ‘date’, ‘country’ (all houses are in the USA), ‘city’ (less information than ‘statezip’) and ‘address’ (too granular). We recognize that ‘street’ provides useful information but in order to utilize it in a regression context, we would probably have to generalize it to a higher-level feature like ‘district’. This could be achieved by geographic clustering method but due to complexity, we restrain from that for now.

Lastly, we split our dataset into a training (80%) and a validation data set (20%). This is useful to be able to later evaluate the predictive performance of our final models on data that was not used in the training process.

2.2 Graphical exploration

In order to get a better understanding of our data, we have a look at the distributions of both the dependent and independent variables. Figure 1 shows the histograms of the continuous variables. The age variable seems to be relatively equally distributed, ranging from 0 to over 100 years. The price distribution is clinched together due to the presence of outliers. Still, it is noticeable that the distribution is right-skewed, which makes a log-transformation appealing. Similar observations can be made for the square feet data on living and lot. Consequently, we log those as well. The results presented in Figure 2, show that after doing this the distributions are much more centered and closer to normal distributions.

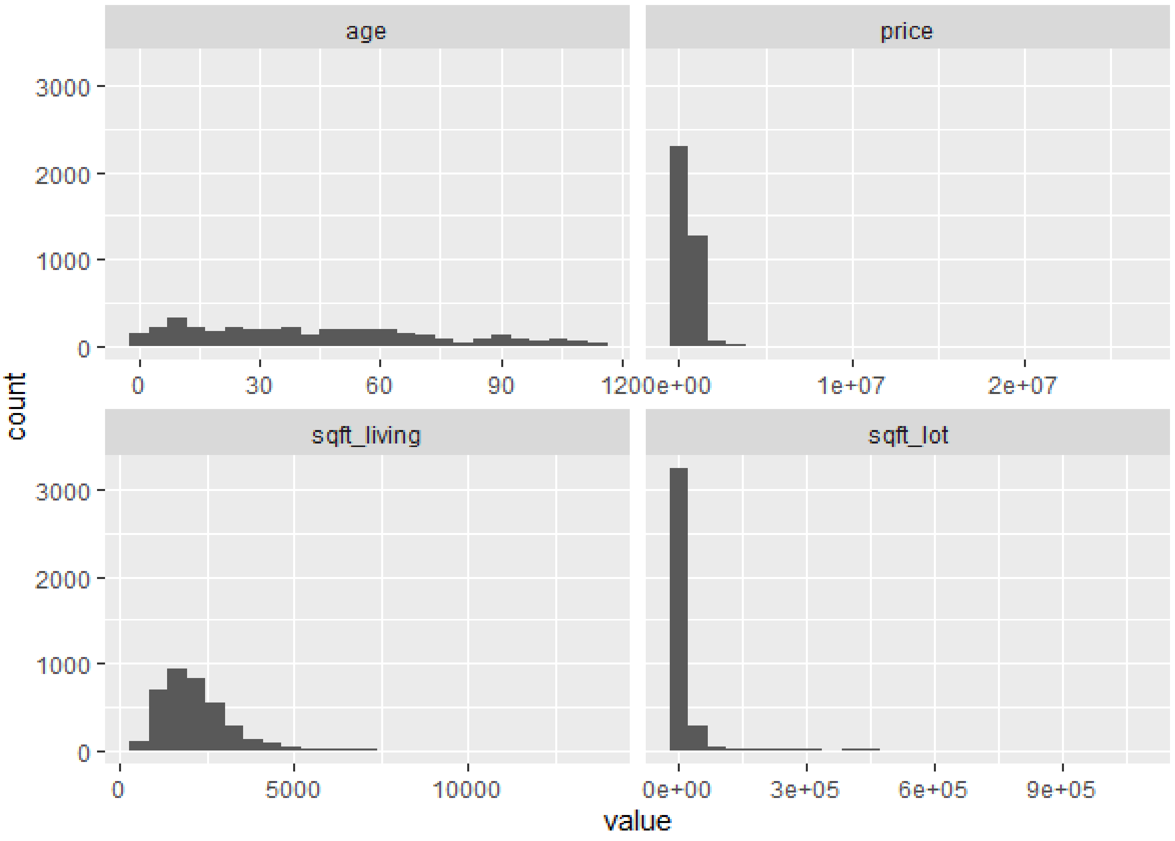


Figure 1: Histograms of the continuous variables

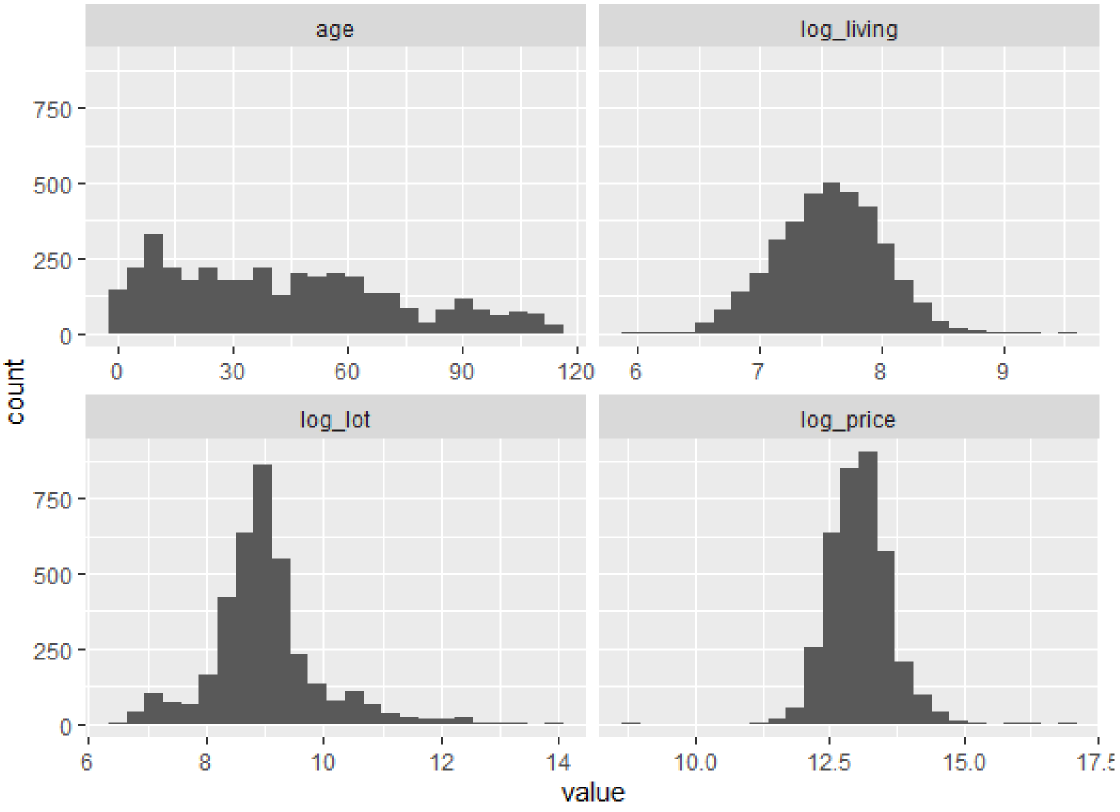


Figure 2: Histograms of the log variables

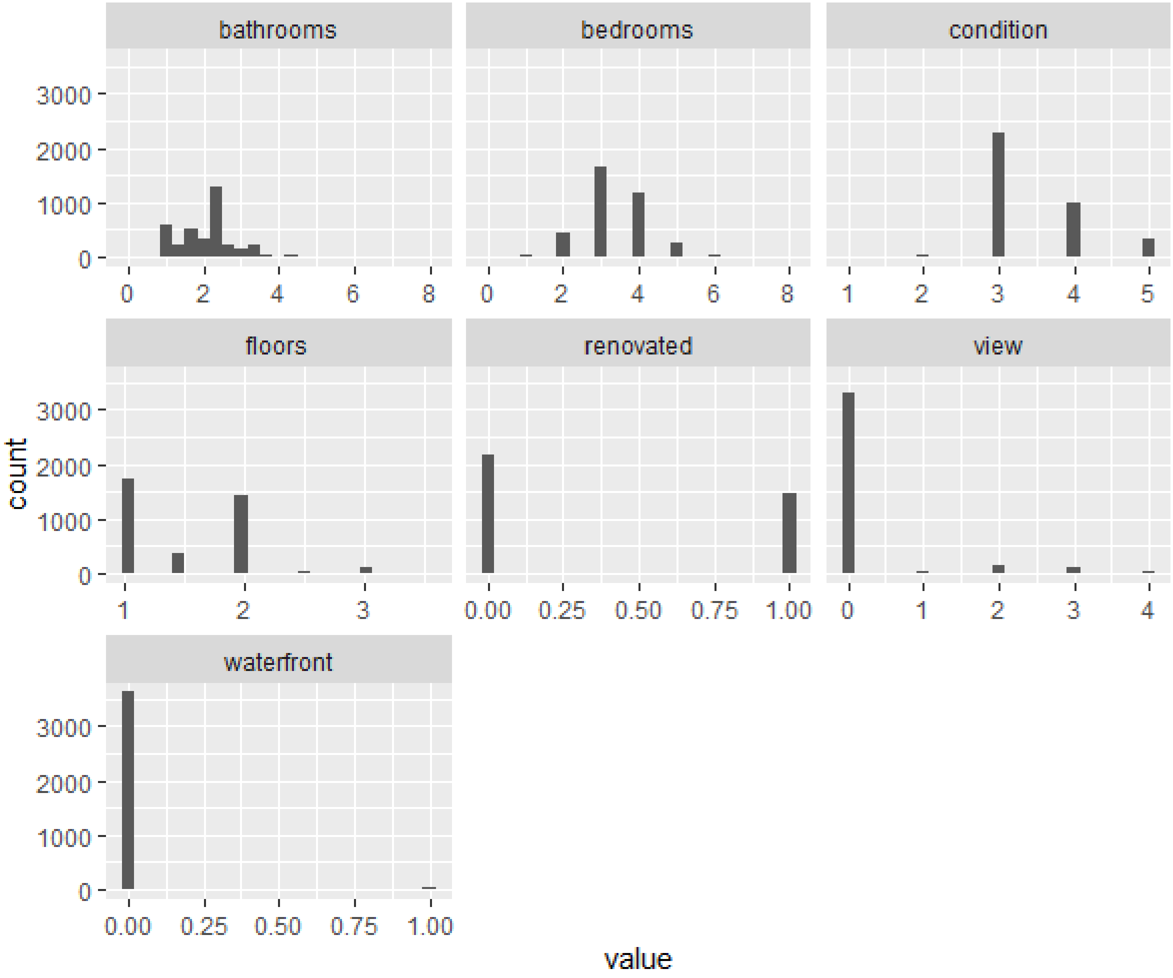
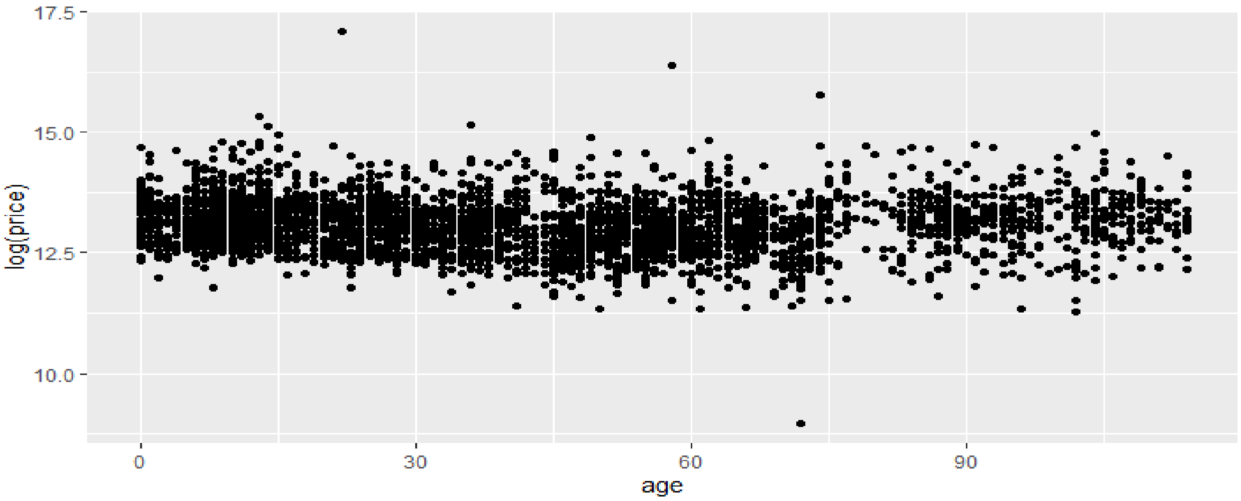


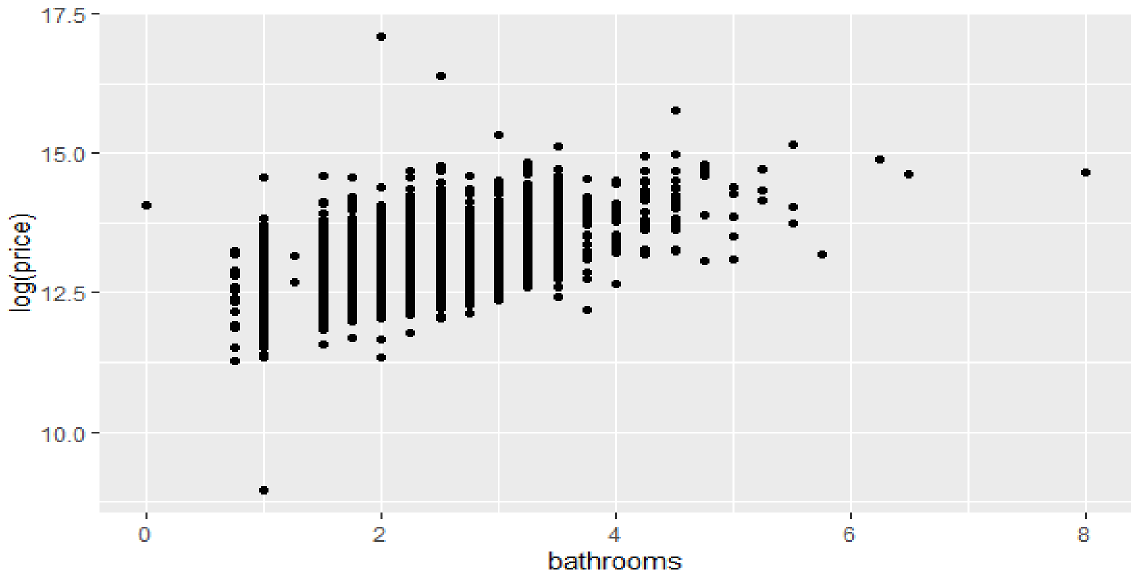
Figure 3: Histograms of discrete data

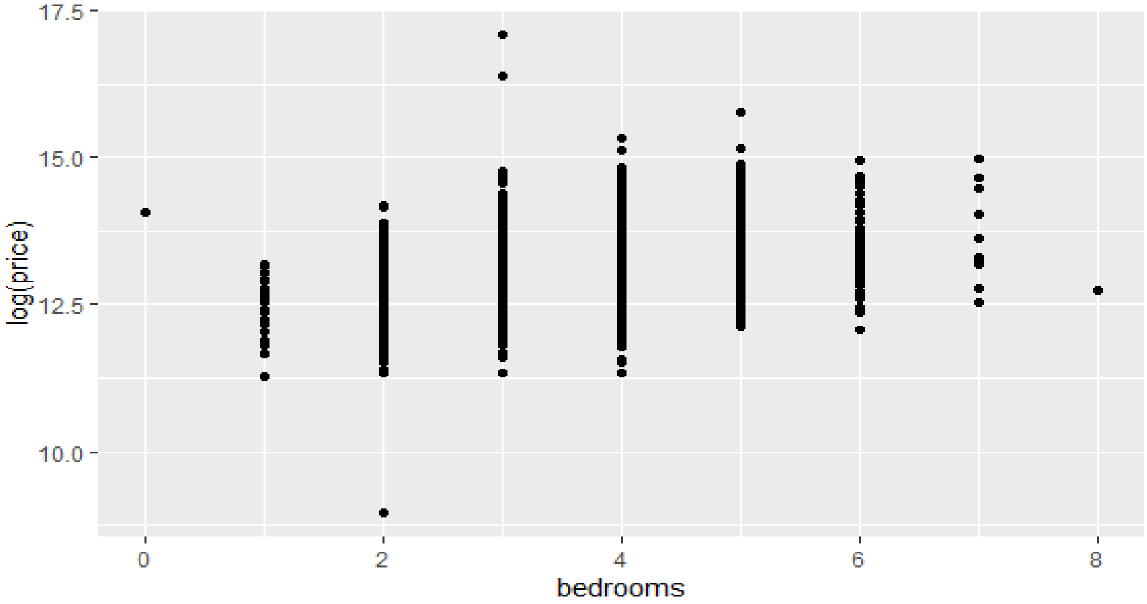
For the discrete data (Figure 3), we can observe that most houses have 1 to 3 bathrooms, but interestingly the dataset also includes half and quarter bathrooms which is not what we commonly would expect. Since this seems to be done consistently, we do not mind this circumstance. The ‘bedrooms’ variable shows the highest number of observations for 3 and 4 bedrooms which could mean that not a lot of small houses are present in our sample. Looking at the floors we can see a similar number of houses with 1 floor and 2 floors.

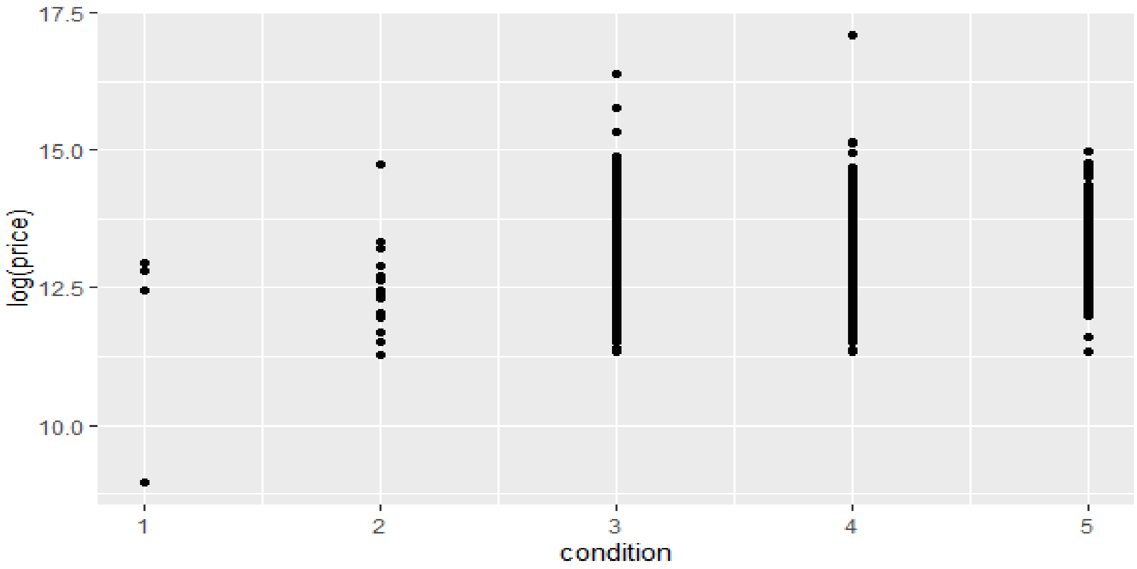
Also, we again have .5 values which allow for special cases that are in between. The ‘condition’ data consists of 5 categories of which the third one is attributed to most houses, followed by the fourth and fifth categories. The ‘view’ histogram shows values between 0 and 4 and most observations are given a score of 0. Given the documentation for the dataset, it is hard to tell if these two variables are ordinal or interval-scaled which has implications for how we implement them in our regression model. Moving forward we treat them as ordinal, which is the safer assumption. The ‘renovated’ and ‘waterfront’ variables are binary and consequently only show values of 0 or 1. We see that a significant portion of houses was renovated after building, but the majority were not. Also, it becomes apparent that only a small minority of houses are located directly at the waterfront. The 'statezip' data is too granular to be visualized in a meaningful way.

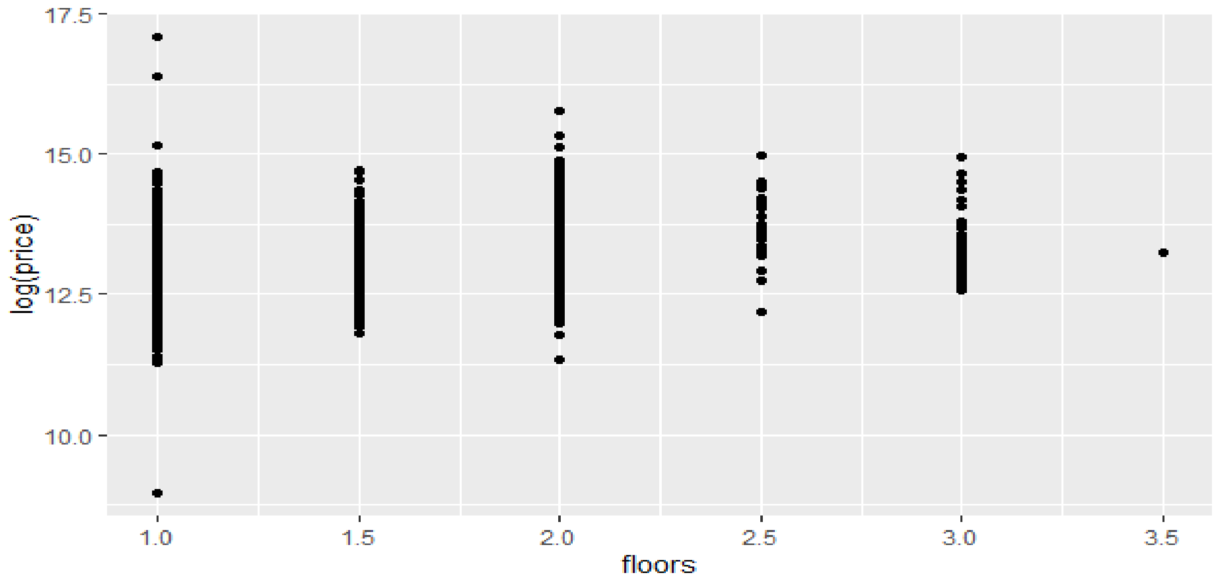
To get an impression of the relationship between the (logged) housing price and the potential predictor variables, we produce different scatter plots (following Figures). Looking at the one for age, there is hardly an observable trend in the data. Based on our data it appears that this variable does not have the explanatory and predictive power we expected. A clearer relationship can be observed for the bathroom data, which shows a noisy but clearly positive slope. For bedrooms, floors and condition the relationship seems more ambiguous again, with possibly a slightly positive trend. A clear relationship can be observed for the logged living area in square feet. It appears that larger living space coincides with larger housing prices. For the logged area of the lot and the discrete variables 'renovated' and 'view' the scatter plots neither shows a strong positive or negative relationship. While the houses located at a waterfront seem to be rare, they seem to have higher prices on average than the ones without one.

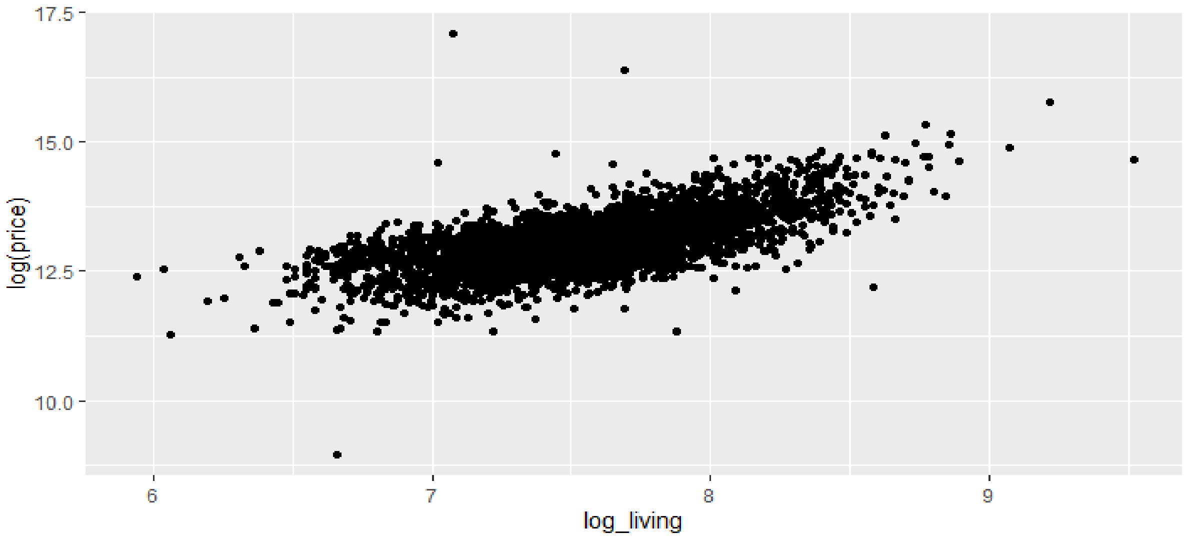


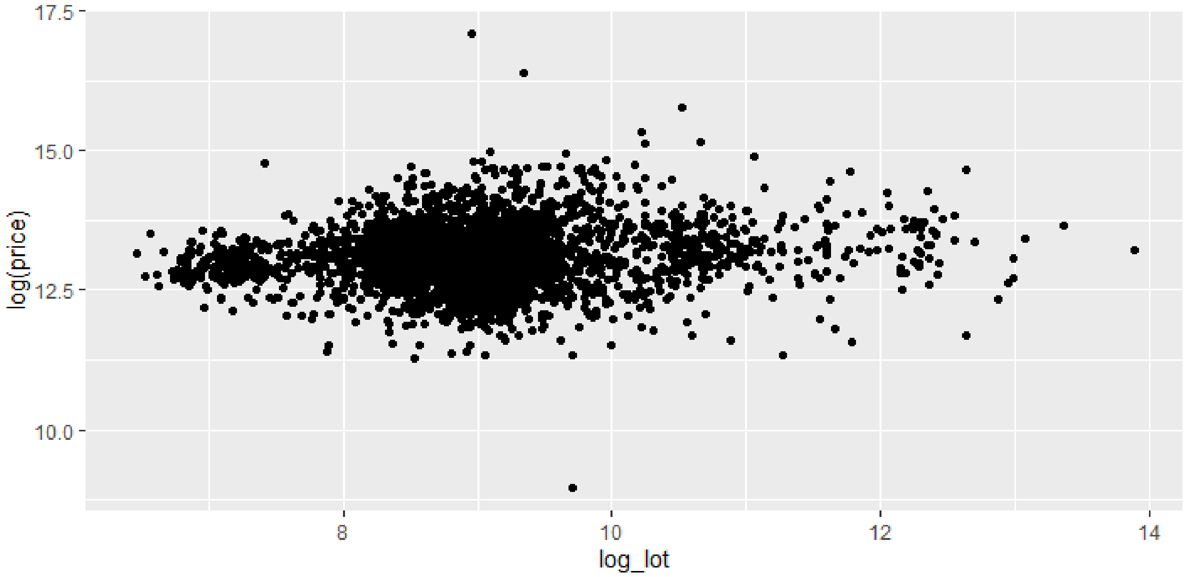

17.5
15.0
12.5
10.0
log(price)
0
2
4
6
8
bathrooms

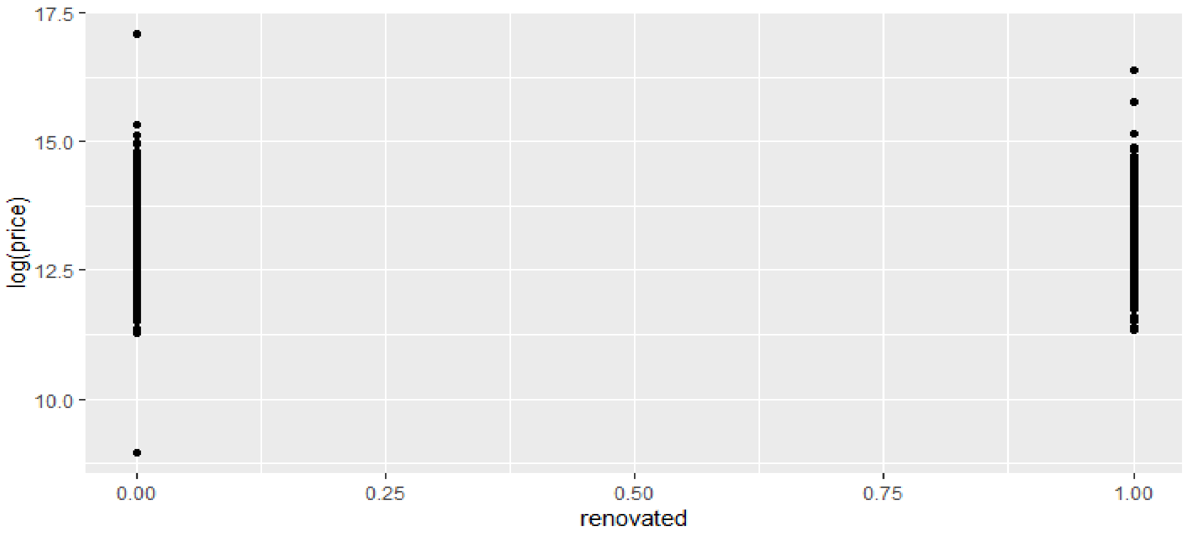

17.5
15.0
12.5
10.0
log(price)
0
2
4
6
8
bedrooms

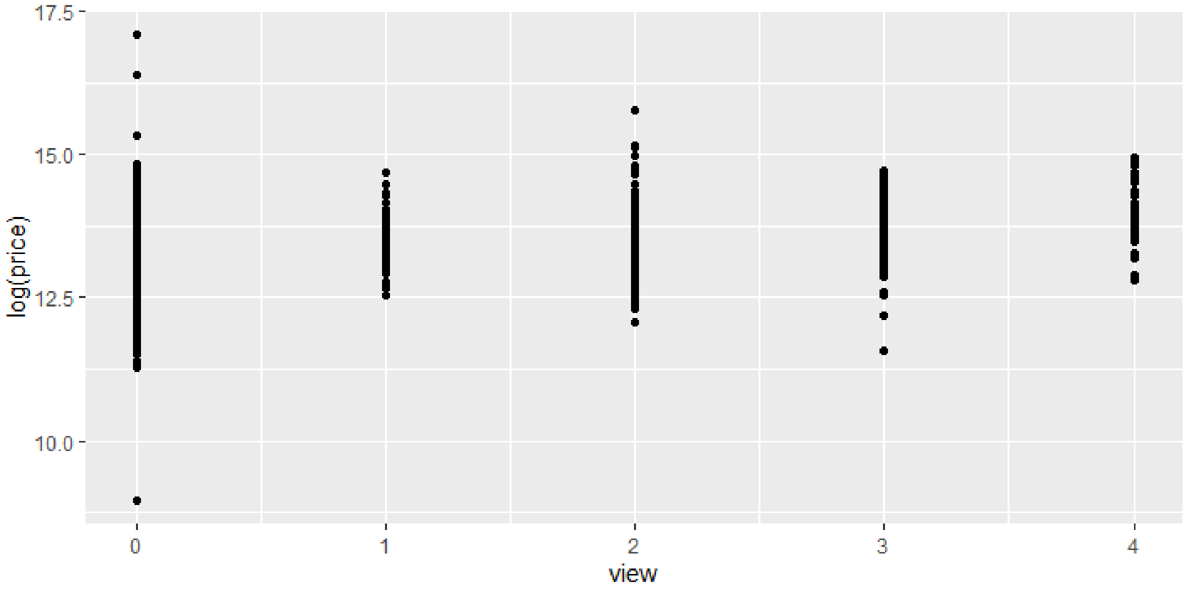

17.5
15.0
12.5
10.0
log(price)
1
2
3
4
5
condition

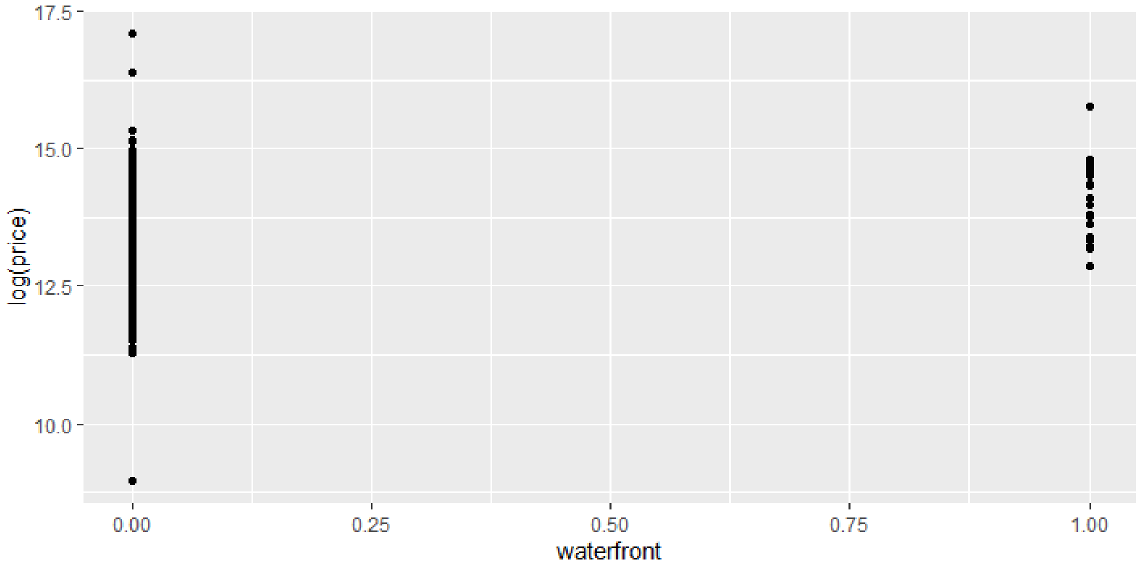

17.5
15.0
12.5
10.0
log(price)
1.0
1.5
2.0
2.5
3.0
3.5
floors


17.5
15.0
12.5
10.0
log(price)
6
7
8
9
log_living


17.5
15.0
12.5
10.0
log(price)
8
10
12
14
log_lot

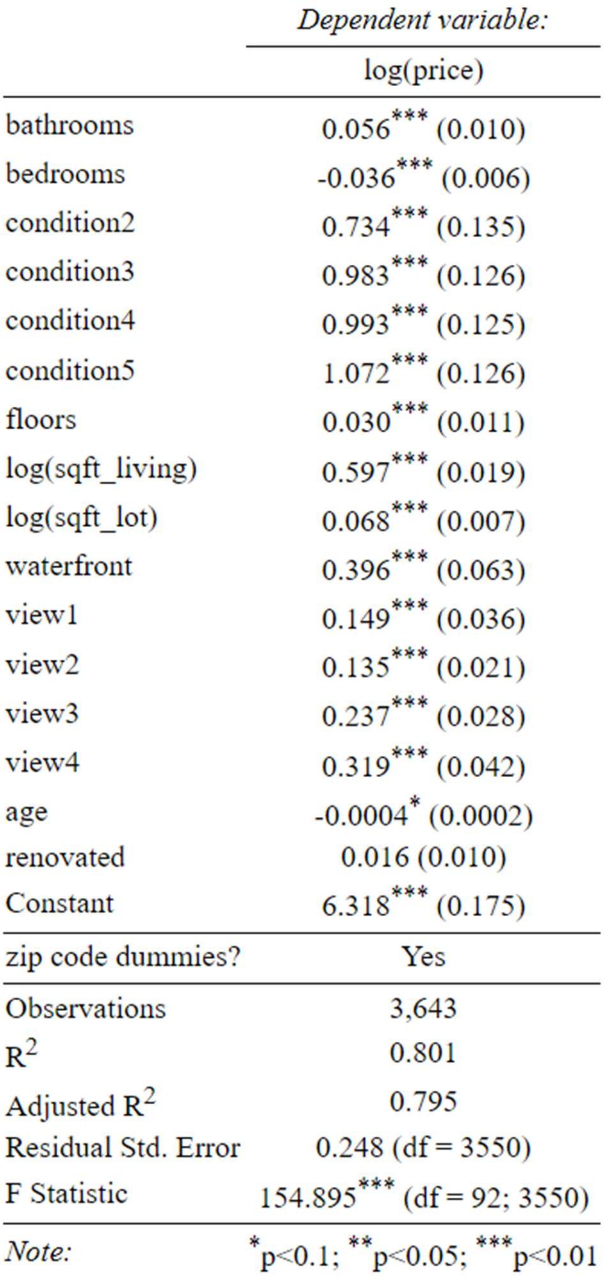

17.5
15.0
12.5
10.0
log(price)
0.00
0.25
0.50
0.75
1.00
renovated


17.5
15.0
12.5
10.0
log(price)
0
1
2
3
4
view


17.5
15.0
12.5
10.0
log(price)
0.00
0.25
0.50
0.75
1.00
waterfront

2.3 Regression Modelling

After an initial graphical exploration, we run a first regression model with all potential predictor variables. Since there are 77 different zip codes our model includes 76 dummy variables that we do not report in detail for now. A condensed regression output is shown in Figure 4. We find significant p-values for most of our regressor variables except 'age', 'renovated', and some of the before-mentioned dummies.



	Dependent variable:
	log(price)
bathrooms	0.056^{***} (0.010)
bedrooms	-0.036^{***} (0.006)
condition2	0.734^{***} (0.135)
condition3	0.983^{***} (0.126)
condition4	0.993^{***} (0.125)
condition5	1.072^{***} (0.126)
floors	0.030^{***} (0.011)
log(sqft_living)	0.597^{***} (0.019)
log(sqft_lot)	0.068^{***} (0.007)
waterfront	0.396^{***} (0.063)
view1	0.149^{***} (0.036)
view2	0.135^{***} (0.021)
view3	0.237^{***} (0.028)
view4	0.319^{***} (0.042)
age	-0.0004^{*} (0.0002)
renovated	0.016 (0.010)
Constant	6.318^{***} (0.175)
zip code dummies?	Yes
Observations	3,643
R^2	0.801
Adjusted R^2	0.795
Residual Std. Error	0.248 (df = 3550)
F Statistic	154.895^{***} (df = 92; 3550)
Note:	*p<0.1; **p<0.05; ***p<0.01

Figure 4: Regression output

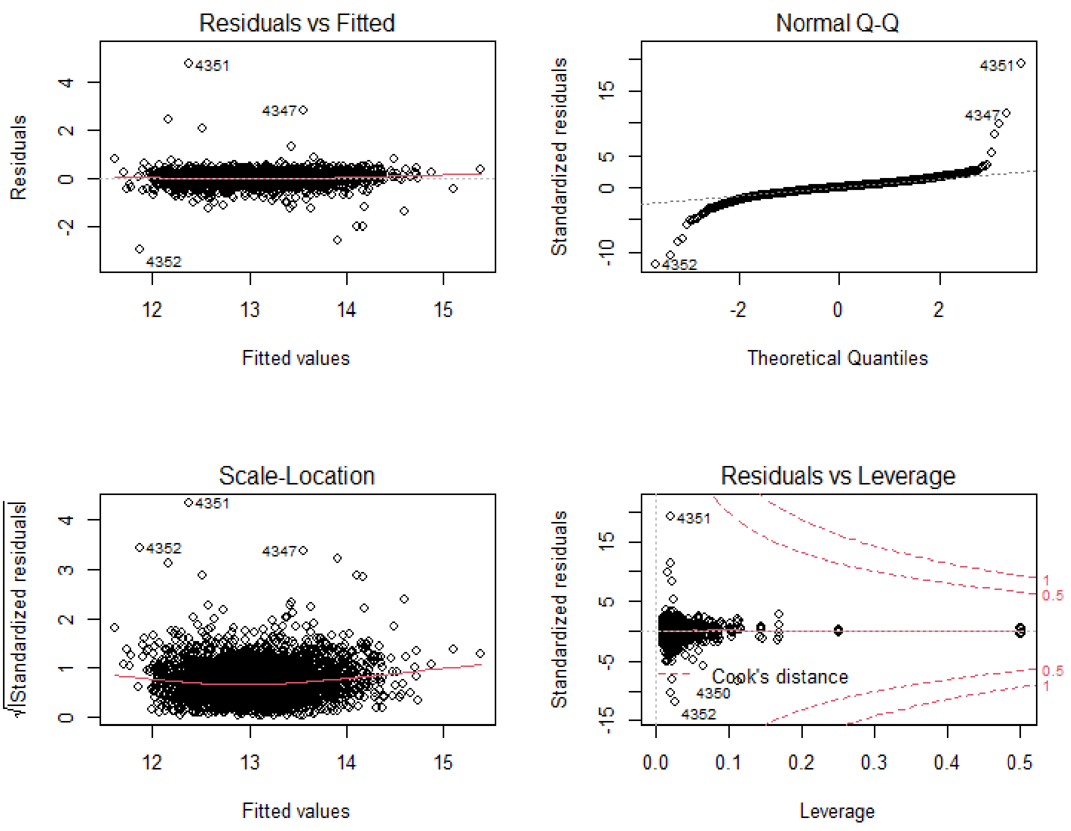
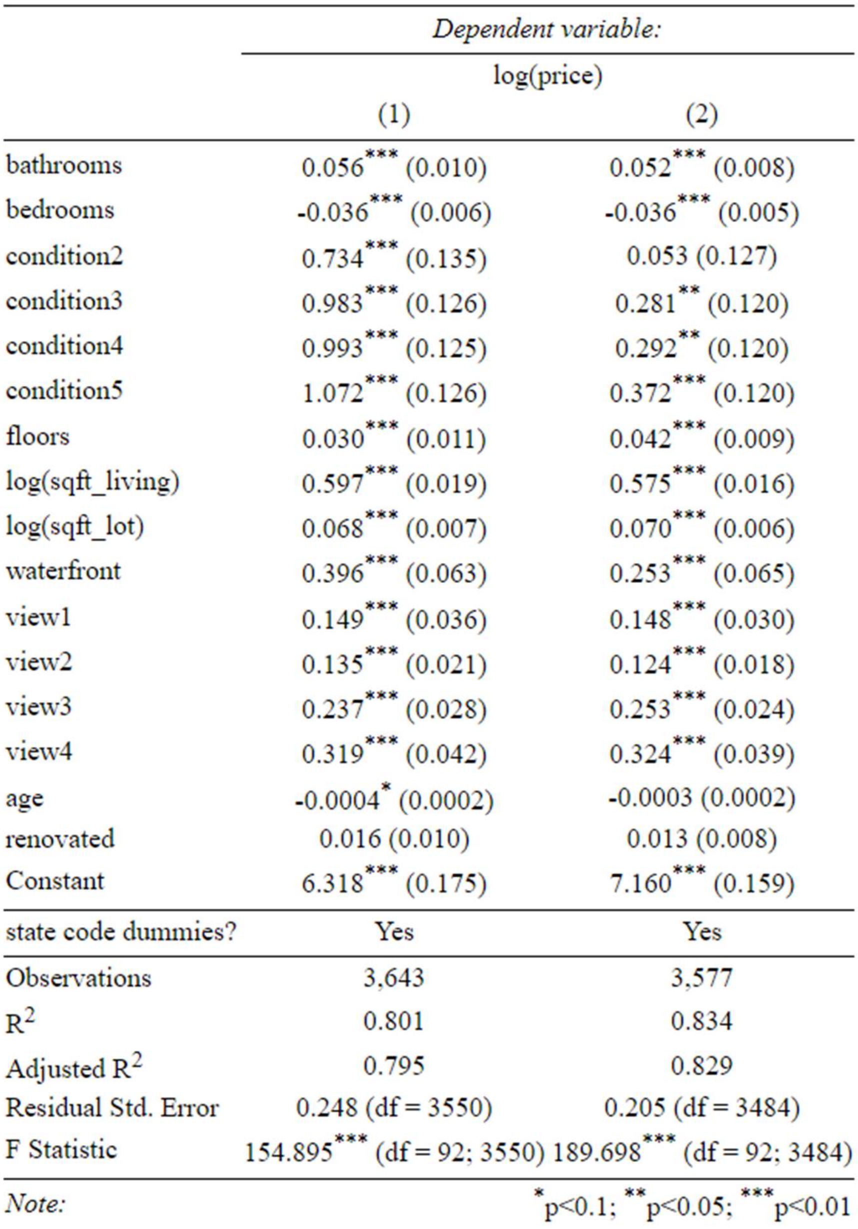


Figure 5: Residual plots

Since our aim is to predict housing prices, we are not particularly focused on the beta coefficients but rather on how our model can be improved. Therefore, we first have a look at the residual plots, presented in Figure 5. The residuals seem to have a constant variance but are far from normally distributed. The main reason for this is the fat tails that we assume to be caused by outliers.

Since outliers can bias the estimates of our model, we decide to remove observations with extremely high or low housing prices by applying the box-plot method. A common convention is to label values that deviate more than 1.5 times the interquartile range from the upper or lower quartile. Since the pricing data is skewed, we apply this method to the logged prices, to account for particularly high and low prices equally. By doing that, our training sample decreases from 3,643 to 3,577 observations.

Running the regression again results in the updated regression model and residual plots, presented in Figure 6 and Figure 7. We can see that the outlier removal has changed some of the coefficients quite significantly. Also, the residual plots show that there could still be potential outliers here, but since the majority does not have a combination of large residual and high leverage, we only remove observation ‘4350’ for now.



	Dependent variable:	
	log(price)	
	(1)	(2)
bathrooms	0.056*** (0.010)	0.052*** (0.008)
bedrooms	-0.036*** (0.006)	-0.036*** (0.005)
condition2	0.734*** (0.135)	0.053 (0.127)
condition3	0.983*** (0.126)	0.281** (0.120)
condition4	0.993*** (0.125)	0.292** (0.120)
condition5	1.072*** (0.126)	0.372*** (0.120)
floors	0.030*** (0.011)	0.042*** (0.009)
log(sqft_living)	0.597*** (0.019)	0.575*** (0.016)
log(sqft_lot)	0.068*** (0.007)	0.070*** (0.006)
waterfront	0.396*** (0.063)	0.253*** (0.065)
view1	0.149*** (0.036)	0.148*** (0.030)
view2	0.135*** (0.021)	0.124*** (0.018)
view3	0.237*** (0.028)	0.253*** (0.024)
view4	0.319*** (0.042)	0.324*** (0.039)
age	-0.0004* (0.0002)	-0.0003 (0.0002)
renovated	0.016 (0.010)	0.013 (0.008)
Constant	6.318*** (0.175)	7.160*** (0.159)
state code dummies?	Yes	Yes
Observations	3,643	3,577
R^2	0.801	0.834
Adjusted R^2	0.795	0.829
Residual Std. Error	0.248 (df = 3550)	0.205 (df = 3484)
F Statistic	154.895*** (df = 92; 3550)	189.698*** (df = 92; 3484)
Note:		*p<0.1; **p<0.05; ***p<0.01

Figure 6: Updated regression model

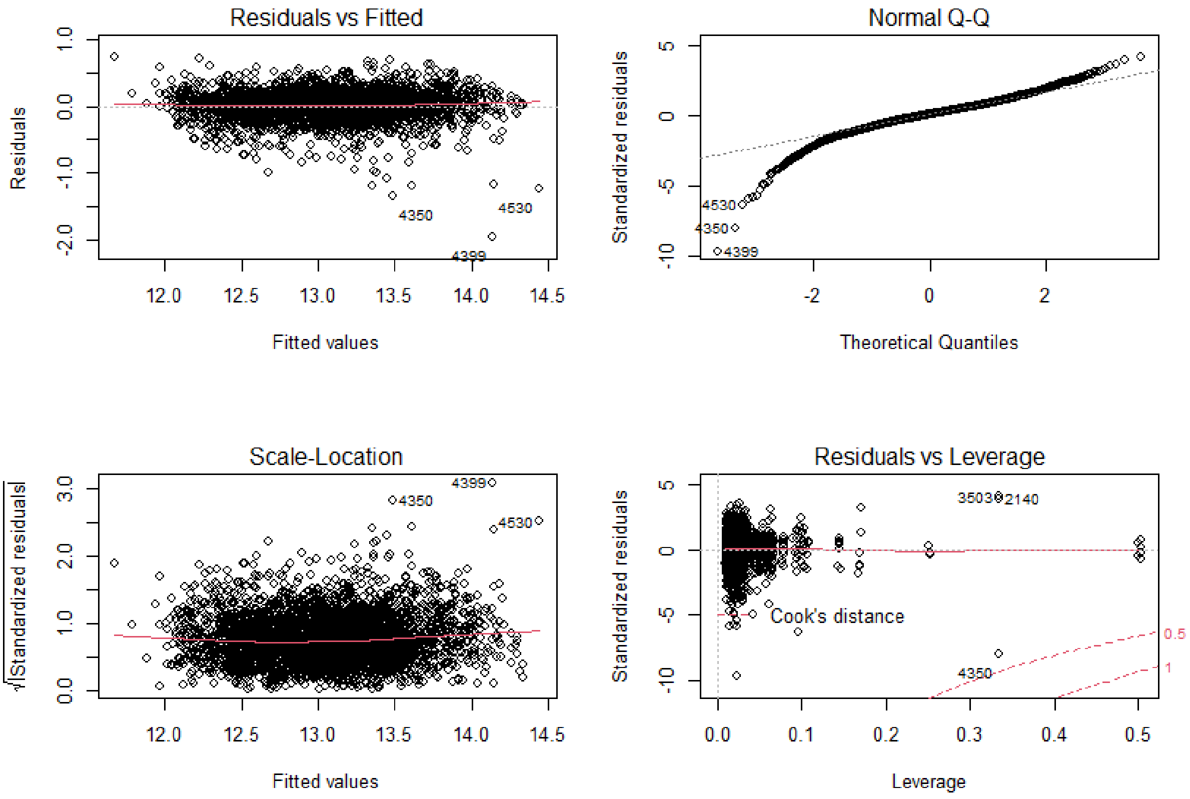
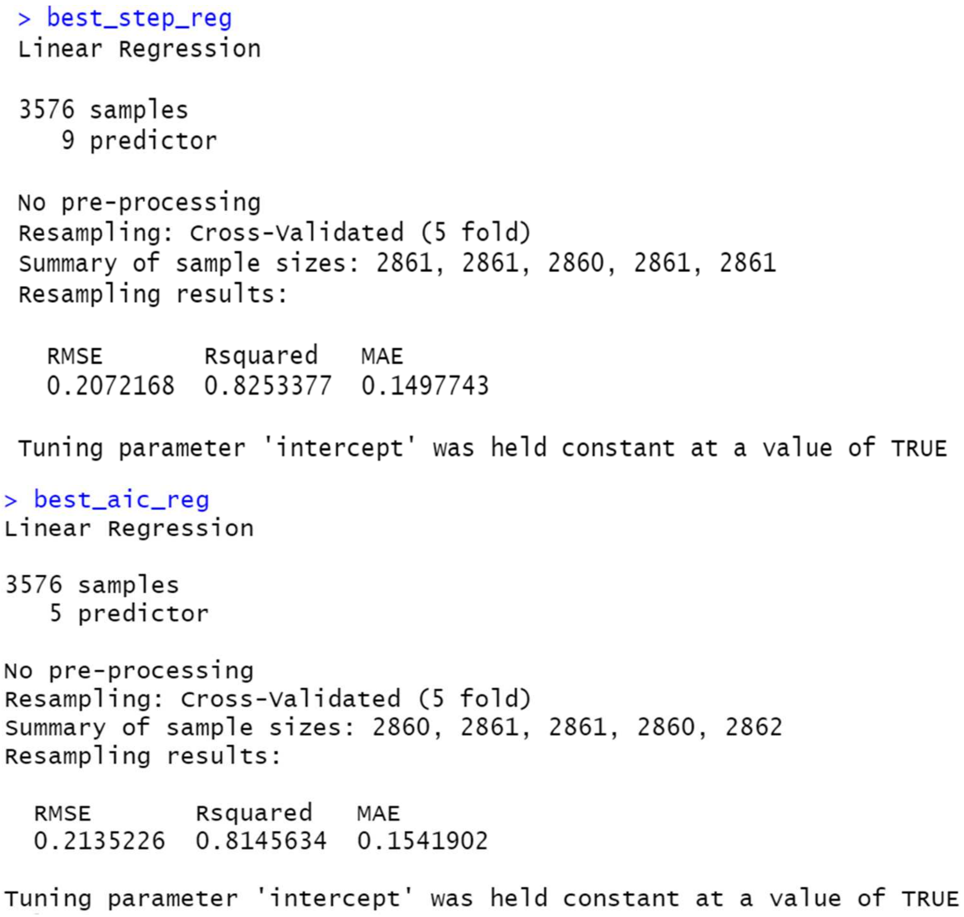


Figure 7: Updated residual plots

For now, we have just run a full model with all available predictor variables. In the next step, we use two different approaches to find models that might yield a better predictive performance. First, we apply a stepwise approach in which predictors are removed based on the AIC information criterion until no improvement can be made. The results imply that the ‘age’ and the ‘renovated’ variable should be dropped in that order. Second, we run an exhaustive search for each variable combination. The model with the lowest AIC value relies on the predictor’s ‘bathroom’, ‘log(sqft_living)’, ‘log(sqft_lot)’, ‘view' and ‘statezip’.

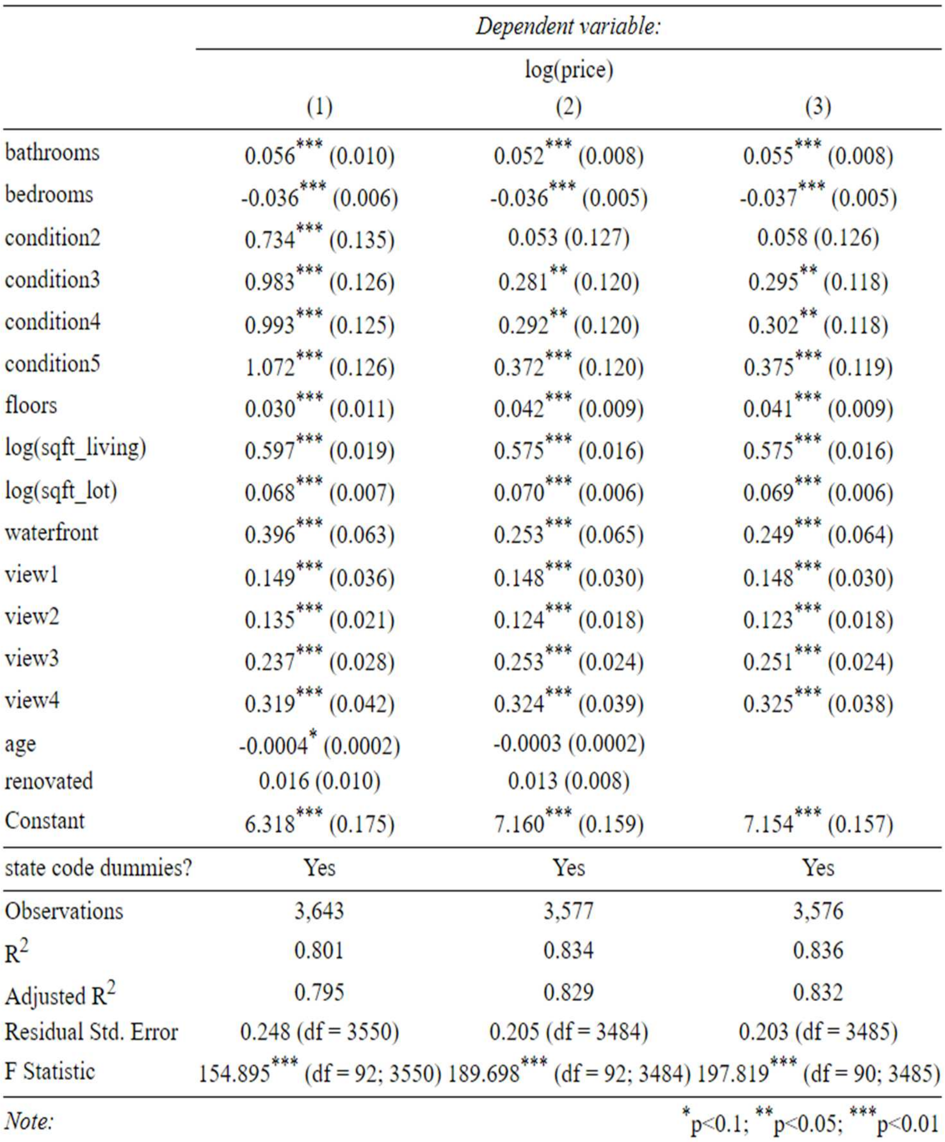


```
> best_step_reg
Linear Regression

3576 samples
   9 predictor

No pre-processing
Resampling: Cross-Validated (5 fold)
Summary of sample sizes: 2861, 2861, 2860, 2861, 2861
Resampling results:

  RMSE       Rsquared   MAE
  0.2072168  0.8253377  0.1497743

Tuning parameter 'intercept' was held constant at a value of TRUE
```

```
> best_aic_reg
Linear Regression

3576 samples
   5 predictor

No pre-processing
Resampling: Cross-Validated (5 fold)
Summary of sample sizes: 2860, 2861, 2861, 2860, 2862
Resampling results:

  RMSE       Rsquared   MAE
  0.2135226  0.8145634  0.1541902

Tuning parameter 'intercept' was held constant at a value of TRUE
```

Figure 8: Results 5-fold cross validation on the training data set

In order to decide on a final model, we test the predictive performance of both models with a 5-fold cross-validation on the training data set. The results are displayed in Figure 8. Comparing both the root mean squared error (RMSE) and the mean absolute error we find better performance for the model selected by the stepwise approach. The final regression model is displayed in column 3 of Figure 9.

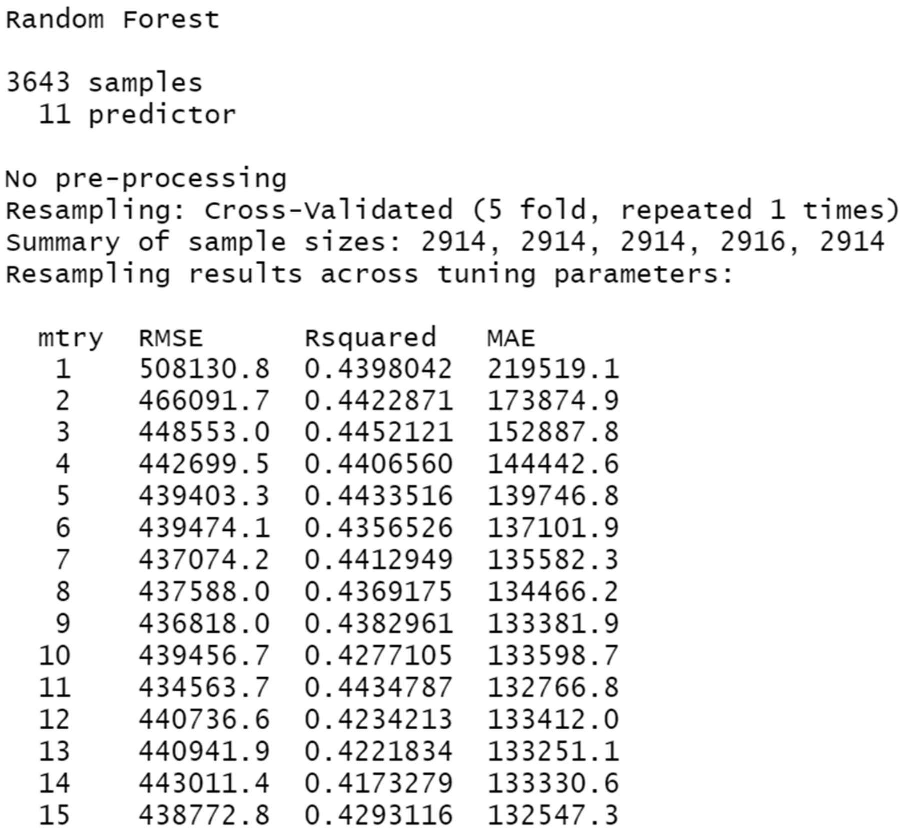


	Dependent variable:		
	log(price)		
	(1)	(2)	(3)
bathrooms	0.056*** (0.010)	0.052*** (0.008)	0.055*** (0.008)
bedrooms	-0.036*** (0.006)	-0.036*** (0.005)	-0.037*** (0.005)
condition2	0.734*** (0.135)	0.053 (0.127)	0.058 (0.126)
condition3	0.983*** (0.126)	0.281** (0.120)	0.295** (0.118)
condition4	0.993*** (0.125)	0.292** (0.120)	0.302** (0.118)
condition5	1.072*** (0.126)	0.372*** (0.120)	0.375*** (0.119)
floors	0.030*** (0.011)	0.042*** (0.009)	0.041*** (0.009)
log(sqft_living)	0.597*** (0.019)	0.575*** (0.016)	0.575*** (0.016)
log(sqft_lot)	0.068*** (0.007)	0.070*** (0.006)	0.069*** (0.006)
waterfront	0.396*** (0.063)	0.253*** (0.065)	0.249*** (0.064)
view1	0.149*** (0.036)	0.148*** (0.030)	0.148*** (0.030)
view2	0.135*** (0.021)	0.124*** (0.018)	0.123*** (0.018)
view3	0.237*** (0.028)	0.253*** (0.024)	0.251*** (0.024)
view4	0.319*** (0.042)	0.324*** (0.039)	0.325*** (0.038)
age	-0.0004* (0.0002)	-0.0003 (0.0002)	
renovated	0.016 (0.010)	0.013 (0.008)	
Constant	6.318*** (0.175)	7.160*** (0.159)	7.154*** (0.157)
state code dummies?	Yes	Yes	Yes
Observations	3,643	3,577	3,576
R^2	0.801	0.834	0.836
Adjusted R^2	0.795	0.829	0.832
Residual Std. Error	0.248 (df = 3550)	0.205 (df = 3484)	0.203 (df = 3485)
F Statistic	154.895*** (df = 92; 3550)	189.698*** (df = 92; 3484)	197.819*** (df = 90; 3485)
Note:			*p<0.1; **p<0.05; ***p<0.01

Figure 9: Final regression model (training set)

2.4 Evaluating predictive performance

Without comparison, however, it is hard to evaluate how good the predictive performance of our regression model really is. Therefore, we consider two popular machine learning algorithms: random forests and xgboost. Both ensemble methods are trained on the same training data set and tuned regarding their different hyperparameters. For the random forest algorithm, we focus on the number of variables randomly sampled as candidates at each split. For 5-fold cross-validation we find the lowest RMSE for the hyperparameter being set to 11 (Figures 10 and 11). For xgboost there are a variety of parameters to tune, consequently we set up a grid search and select the best performing set of hyperparameters based on RMSE again (Figure 12).



```
Random Forest

3643 samples
  11 predictor

No pre-processing
Resampling: Cross-Validated (5 fold, repeated 1 times)
Summary of sample sizes: 2914, 2914, 2914, 2916, 2914
Resampling results across tuning parameters:

  mtry  RMSE      Rsquared   MAE
   1    508130.8  0.4398042  219519.1
   2    466091.7  0.4422871  173874.9
   3    448553.0  0.4452121  152887.8
   4    442699.5  0.4406560  144442.6
   5    439403.3  0.4433516  139746.8
   6    439474.1  0.4356526  137101.9
   7    437074.2  0.4412949  135582.3
   8    437588.0  0.4369175  134466.2
   9    436818.0  0.4382961  133381.9
  10    439456.7  0.4277105  133598.7
  11    434563.7  0.4434787  132766.8
  12    440736.6  0.4234213  133412.0
  13    440941.9  0.4221834  133251.1
  14    443011.4  0.4173279  133330.6
  15    438772.8  0.4293116  132547.3
```

Figure 10: Random forest hyperparameter tuning

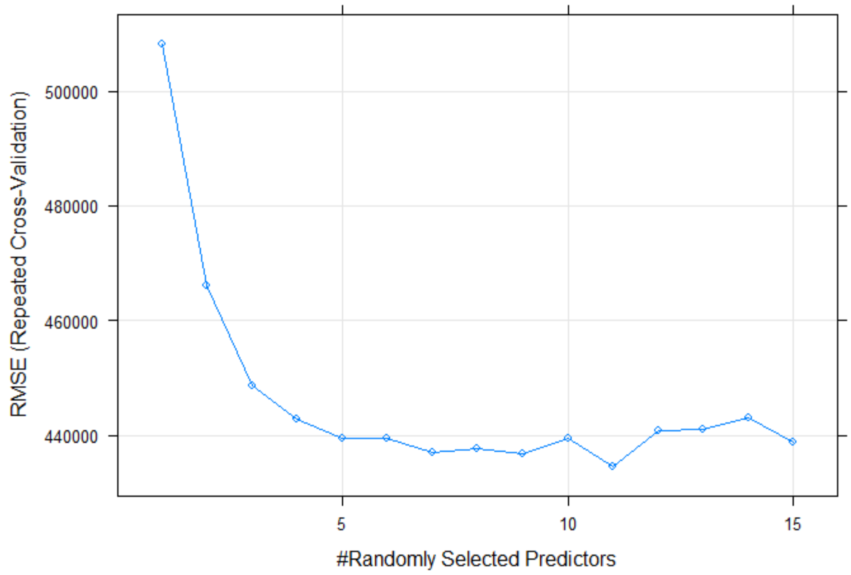
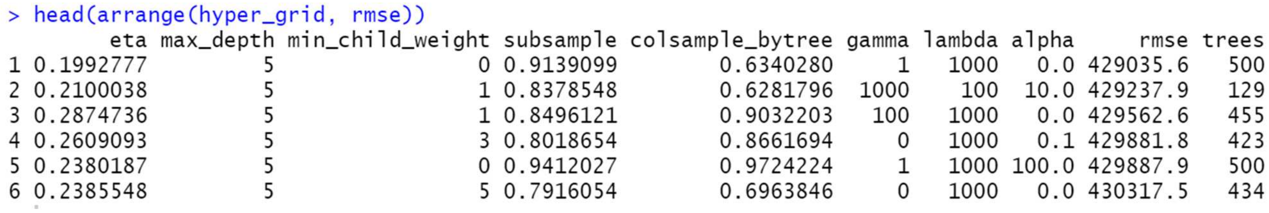


Figure 11: Visualisation random forest hyperparameter tuning



```
> head(arrange(hyper_grid, rmse))
        eta max_depth min_child_weight subsample colsample_bytree gamma lambda alpha     rmse trees
1 0.1992777         5                0 0.9139099        0.6340280     1   1000   0.0 429035.6   500
2 0.2100038         5                1 0.8378548        0.6281796  1000    100  10.0 429237.9   129
3 0.2874736         5                1 0.8496121        0.9032203   100   1000   0.0 429562.6   455
4 0.2609093         5                3 0.8018654        0.8661694     0   1000   0.1 429881.8   423
5 0.2380187         5                0 0.9412027        0.9724224     1   1000 100.0 429887.9   500
6 0.2385548         5                5 0.7916054        0.6963846     0   1000   0.0 430317.5   434
```

Figure 12: xgboost hyperparameter tuning

The final tree-based models as well as the final regression model are applied to the validation set in order to simulate the prediction process of housing prices. Based on the prediction residuals we calculate the (out-of-sample) RMSE and MAE for each of the three models. The results are presented in Figures 13 and 14. Comparing the metrics we notice that the regression model does perform significantly better in the prediction task. The boosting algorithm does produce better results than random forest but still produces higher errors on average than the regression does. There are multiple possible explanations for these results. For example, there was no outlier detection performed before training the tree-based models which might have resulted in the models falsely learning from this noise (overfitting). Also, it is possible that the preprocessing was not sufficient for the tree-based models to work effectively. Considering the high dimensionality of the data, removing highly correlated variables or constructing a smaller set of principal variables might improve the performance as well. Alternatively, it is also possible that the tuning process was not sufficient.

Another way to look at these results is to note that the regression did produce more accurate predictions than two of the most popular and successful machine learning algorithms. It shows that OLS can at the very least compete with newer modeling techniques and should always be considered also when looking at prediction tasks.

However, it should be noted that predictive performance still is far from satisfactory. Predictions with an average error of 146,386 or 87,730 (based on metric) are too imprecise to basic financial decisions on. Also this performance is not sound enough to use the predictions for investment purposes either.

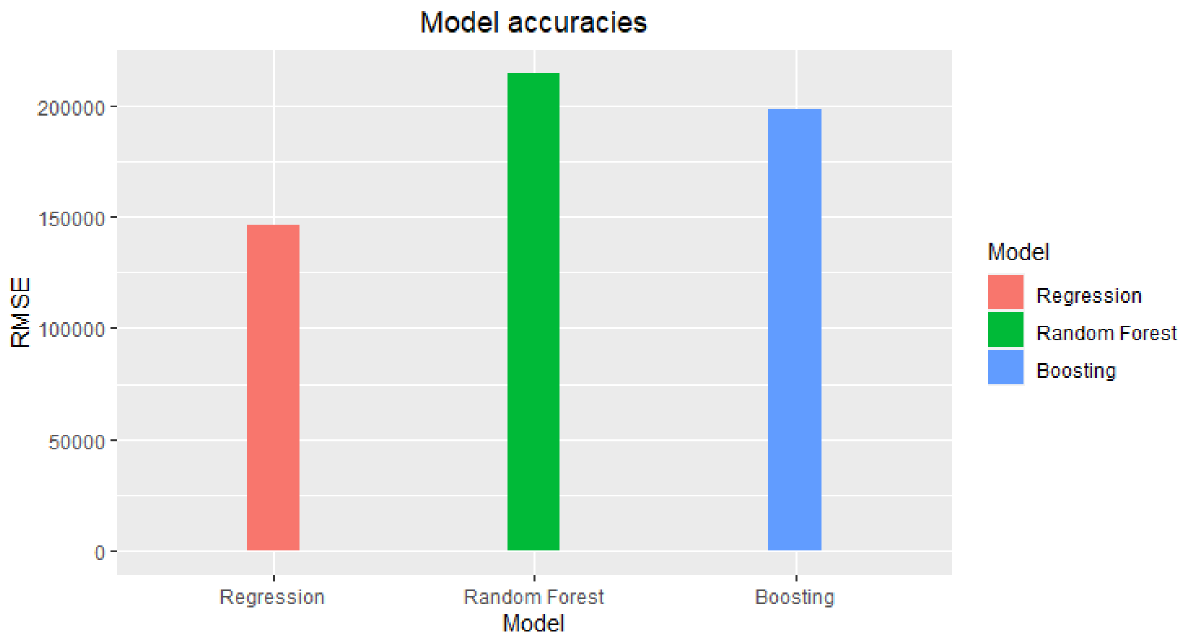


Figure 13: RMSE values for the different models

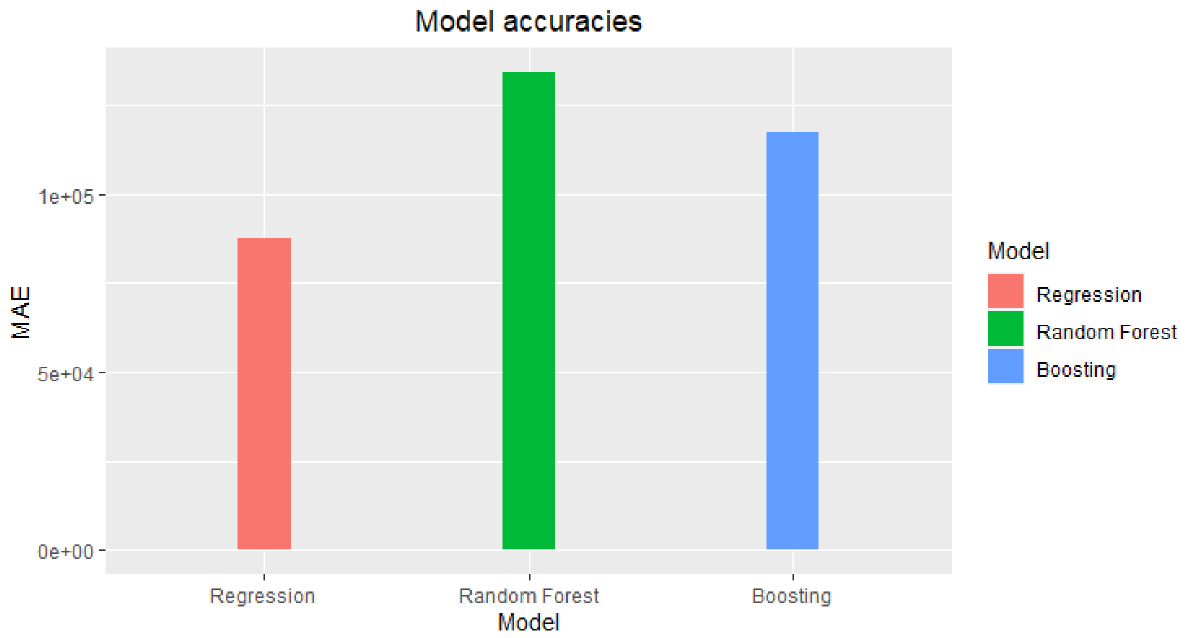


Figure 14: MAE values for the different models

2.5 Final model & main contributors

Lastly, we fit the final model on all the available data. Figure 13 shows the corresponding residuals which look very similar to the ones of the final model on the training set. The regression model itself is presented in Figure 13, including all the dummy variables for the zip codes. Reporting the t-statistics instead of the p-values allows us to better analyze which variables contribute the most to the predictions of the model. The most important factor seems to be ‘log(sqft_living)’. This aligns with the expert sentiment that living space is one of the most important aspects for homeowners. Additionally, some of the zip code dummies show t-statistics over 20, which is also very high. This also is in line with the notion that location represents a crucial driving factor in housing prices.

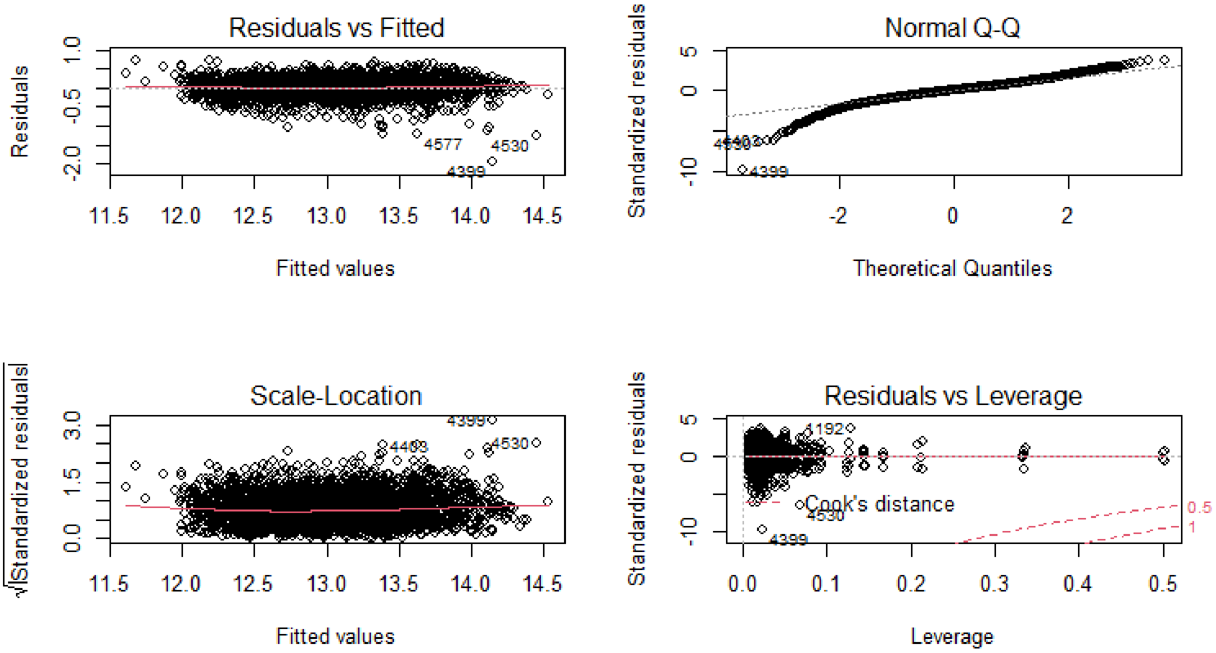


Figure 15: Residual plots of final model (full data set)

Figure 15: Final regression model (full data set)

	Dependent variable:
	log(price)
bathrooms	0.053
	t = 7.718***
bedrooms	-0.035
	t = -7.653***
condition2	-0.047
	t = -0.470
condition3	0.203
	t = 2.211**
condition4	0.211
	t = 2.303**
condition5	0.284
	t = 3.079***
floors	0.040
	t = 5.355***
waterfront	0.293
	t = 5.641***
log(sqft_living)	0.576
	t = 41.002***
log(sqft_lot)	0.072
	t = 14.478***
view1	0.144
	t = 5.482***
view2	0.116
	t = 7.406***
view3	0.230
	t = 10.891***
view4	0.293
	t = 8.959***
statezipWA 98002	0.010
	t = 0.237
statezipWA 98003	0.072
	t = 1.851*
statezipWA 98004	1.187
	t = 32.162***
statezipWA 98005	0.831
	t = 18.052***
statezipWA 98006	0.765
	t = 23.758***
statezipWA 98007	0.791
	t = 19.022***
statezipWA 98008	0.683
	t = 17.475***
statezipWA 98010	0.383
	t = 5.032***
statezipWA 98011	0.493
	t = 11.208***
statezipWA 98014	0.348
	t = 6.777***
statezipWA 98019	0.277
	t = 6.867***
statezipWA 98022	0.083
	t = 1.748*
statezipWA 98023	0.041
	t = 1.251
statezipWA 98024	0.482
	t = 7.228***
statezipWA 98027	0.532
	t = 16.159***
statezipWA 98028	0.466
	t = 13.103***
statezipWA 98029	0.722
	t = 21.394***
statezipWA 98030	0.049
	t = 1.205
statezipWA 98031	0.099
	t = 2.770***
statezipWA 98032	0.020
	t = 0.377
statezipWA 98033	0.861
	t = 25.767***
statezipWA 98034	0.590
	t = 18.064***
statezipWA 98038	0.193

	t = 5.858***	statezipWA 98058	0.215		t = 27.570***
statezipWA 98039	1.325		t = 6.580***	statezipWA 98106	0.368
	t = 11.031***	statezipWA 98059	0.425		t = 10.255***
statezipWA 98040	0.970		t = 13.223***	statezipWA 98107	0.928
	t = 27.756***	statezipWA 98065	0.508		t = 24.794***
statezipWA 98042	0.097		t = 14.324***	statezipWA 98108	0.419
	t = 2.957***	statezipWA 98068	0.566		t = 10.486***
statezipWA 98045	0.334		t = 2.767***	statezipWA 98109	1.174
	t = 8.712***	statezipWA 98070	0.286		t = 25.528***
statezipWA 98047	0.014		t = 6.025***	statezipWA 98112	1.167
	t = 0.163	statezipWA 98072	0.555		t = 32.094***
statezipWA 98050	0.460		t = 15.391***	statezipWA 98115	0.857
	t = 3.146***	statezipWA 98074	0.667		t = 27.399***
statezipWA 98051	0.298		t = 20.218***	statezipWA 98116	0.839
	t = 3.664***	statezipWA 98075	0.700		t = 23.128***
statezipWA 98052	0.765		t = 20.450***	statezipWA 98117	0.862
	t = 24.835***	statezipWA 98077	0.548		t = 27.631***
statezipWA 98053	0.679		t = 14.243***	statezipWA 98118	0.520
	t = 20.621***	statezipWA 98092	0.103		t = 15.179***
statezipWA 98055	0.199		t = 3.060***	statezipWA 98119	1.053
	t = 4.329***	statezipWA 98102	1.069		t = 26.330***
statezipWA 98056	0.384		t = 21.045***	statezipWA 98122	0.900
	t = 11.307***	statezipWA 98103	0.878		t = 24.993***
statezipWA 98057	0.067		t = 28.297***	statezipWA 98125	0.554
	t = 1.085	statezipWA 98105	1.089		t = 16.619***

statezipWA 98126	0.608
	t = 17.637***
statezipWA 98133	0.495
	t = 15.002***
statezipWA 98136	0.744
	t = 19.902***
statezipWA 98144	0.763
	t = 22.001***
statezipWA 98146	0.357
	t = 9.437***
statezipWA 98148	0.222
	t = 3.918***
statezipWA 98155	0.466
	t = 13.938***
statezipWA 98166	0.375
	t = 10.047***
statezipWA 98168	0.102
	t = 2.828***
statezipWA 98177	0.679
	t = 17.625***
statezipWA 98178	0.091
	t = 2.329**
statezipWA 98188	0.046
	t = 0.907
statezipWA 98198	0.054
	t = 1.452
statezipWA 98199	0.896
	t = 24.766***
statezipWA 98288	-0.002
	t = -0.015
statezipWA 98354	0.333
	t = 2.285**
Constant	7.209
	t = 55.912***
Observations	4,472
R^2	0.835
Adjusted R^2	0.831
Residual Std. Error	0.203 (df = 4381)
F Statistic	245.846*** (df = 90; 4381)
Note:	*p**p***p<0.01

3. Conclusion

Our report on the prediction of housing prices in the state of Washington has produced some interesting results. The predictive performance of our regression model did exceed those of both a random forest and an xgboost model. While it is likely that the performance of the alternative models could be improved by different preprocessing and tuning, this is still a noteworthy finding, given the recent hype around tree-based algorithms. Yet we have to note that the performance of the regression model currently is not high enough to allow for a serious application of the predictions. Consequently, the aim would be to improve the accuracy of the model in a future project. One starting point could be the integration of the address information to allow for a more nuanced differentiation of the location aspect which was one of the main contributors to our regression model. Another important step would be to find more recent data, as the housing market is highly volatile. This is also the reason why applying using a panel dataset would be ideal, as it would allow to incorporate of the time aspect into the regression analysis.