Assignment 2: Data Analytics

Bosse Behrens
Student ID: 12347333
Person A
TU Wien
e12347333@student.tuwien.ac.at

Ghazal Arzanian
Student ID: 12334230
Person B
TU Wien
e12334230@student.tuwien.ac.at

Overview

In this Assignment we will follow the Cross-Industry Standard Process for Data Mining (CRISP-DM) to explore and learn about the data mining process.

1 Business Understanding

Responsible: A + B jointly

1.1 Describing data source and a scenario for business analytics task

We choose a dataset in housing prices of single-family homes in Miami in the year 2016. The data originates from kaggle (https://www.kaggle.com/datasets/deepcontractor/miami-housing-dataset) and provides no further information on its origin. Most probably these are records from local real estate platforms or city/federal records of housing data.

The scenario: We are part of the data analysis team of a real estate agency located in the area of Miami. As part of real estate work we need to set the prices of properties our agency is selling. To do so we need to analyze past prices in terms of attributes of the homes sold to predict the prices for new sales.

1.2 Clearly define and describe the Business Objectives

Our primary business objective is to earn the largest margin of revenue possible. To do so we want to calculate the actual prices of homes as precisely as possible. From that baseline the marketing/pricing team can then make decisions about up-/downpricing, but they need to know what the actual would be, so a home is not unintentionally sold for a too high or too low price. Also with the provided intelligence our real estate agents can make better decisions on buying new properties, e.g. is a home sold cheaply for its actual value? Which homes are overpriced and not worth acquiring? That can help our stakeholders, investors and homebuyers to make informed decisions. It should improve our pricing strategy to enable our real estate agents to set competitive and market-accurate prices. Also we can provide investors with insights into undervalued or overvalued properties. Furthermore we want to increase our operational efficiency by automating our pricing predictions partly instead of relying on manual analysis.

1.3 Clearly define and describe the Business Success Criteria

The main business success criteria in this task is to predict the actual values of homes as precisely as possible. If we go beyond this assignment the business success criteria would also involve the increase in our revenue/profit margin growth. For this we could set a pre-specified goal in terms of growth we want to achieve with our data mining. Since we do not have access to that kind of information in this scenario, for this assignment the business success criteria is similar to the data mining success criteria, which we will discuss in the next parts.

1.4 Clearly define and describe the Data Mining Goals

We have a data mining goal, as already discussed in short form before. The primary goal is to develop a predictive model for the housing prices. This model will be a regression model and should be as well-fitted as possible so we can predict housing prices accurately.

1.5 Clearly define and describe the Data Mining Success Criteria

Our data mining success criteria are metrics that measure the quality of our model. For the quality of the predictions we choose the Root Mean Squared Error (RMSE), which indicates how well a model's predictions are when calculating the error between the predictions and observations. To do so the data has first to be split into train and test sets, so the model is not trained on the same data it is tested on. Since we want to use it to make future predictions for home prices on new data this is very important. Efficiency should also be a goal so the model runs in a reasonable time in a real-world situation. A second value we choose is the $R^2/Adjusted\ R^2$. This value explains how much of the variance in the data is explained by the model.

1.6 Are there any AI risk aspects that may require specific consideration?

A considerable risk might be the stability of the real estate market. Since Housing real estate prices are volatile and have played a big part in past crises (world finance crisis 2007/2008 for example), the model should always be taken with a grain of salt, since it might not br 100% accurate. Therefore it also has to be robust to trends and sudden changes in the market to prevent artificial inflation of prices that do not reflect reality and can lead to a housing crisis in the worst case. Another risk might be some bias in socioeconomic data. Even if most of the features are unrelated to that, for example if certain neighborhoods that have specific demographics are treated differently in terms of pricing. Even if that might be difficult to prevent when implementing such a model, especially in the private sector, it should not be totally forgotten about. Since this model is only used internally, safety should not be an issue but transparency in how it makes predictions might still be a good thing to prevent aforementioned problems.

2 Data Understanding: Data Description Report

Responsible: A

2.1 Attribute types and their semantic

The data set contains 17 attributes of ratio as well as nominal and ordinal scales, mainly about the location relative to nearby points of interests and the about the types of area of the property. The following Table 1 depicts a detailed overview for all features.

Table 1: Miami housing dataset attributes

Attribute	Description	Scale/Data Type
LATITUDE	Latitude of the property	Continuous
LONGITUDE	Longitude of the property	Continuous
PARCELNO	ID for each property	Nominal
SALE_PRC	Sale price of the property in $	Ratio/Continuous
LND_SQFOOT	Land area in square feet	Ratio/Continuous
TOT_LVG_AREA	Floor area in suqare feet	Ratio/Continuous
SPEC_FEAT_VAL	Value of special features (e.g., pools) in $	Ratio/Continuous
RAIL_DIST	Distance to nearest rail line in feet	Ratio/Continuous
OCEAN_DIST	Distance to the ocean in feet	Ratio/Continuous
WATER_DIST	Distance to nearest body of water in feet	Ratio/Continuous
CNTR_DIST	Distance to Miami central business district in feet	Ratio/Continuous
SUBCNTR_DI	Distance to nearest subcenter in feet	Ratio/Continuous
HWY_DIST	Distance to nearest highway in feet	Ratio/Continuous
age	Age of the structure in years	Ratio/Continuous
avno60plus	Airplane noise dummy (if exceeding some value)	Nominal/Binary
structure_quality	Quality of the structure rated 1 - 5	Ordinal/Integer
month_sold	Sale month in 2016 (1 - 12)	Ordinal/Integer

2.2 Statistical properties

The dataset contain 17 features and 13932 observations. Table 2 gives an overview over some univariate statistical properties. We choose to show the value ranges of the features, as well as mean and median, as we think they already can give a good idea about some statistical properties of each feature. In the next subsection we will also inspect the actual value dístributions of some chosen important features in more detail. As we can already see in this table, there are some features that might have very skewed distributions. To this categorizations the features `LND_SQFOOT, SPEC_FEAT_VALUE, HWY_DIST` and `avno60plus` can be identified. All have means and medians that are far closer to the minimum than the maximum in terms of absolute values. The same is true for our chosen response variable we want to predict on, `SALE_PRC`. An uneven distribution is especially problematic, which is why we will have to explore and process this further.

Table 2: Summary Statistics for Miami Housing Dataset

Attribute	Min	Max	Mean	Median
LATITUDE	25.4343	25.9744	25.7288	25.7318
LONGITUDE	-80.5422	-80.1197	-80.3275	-80.3389
PARCELNO	1.02e+11	3.66e+12	2.36e+12	3.04e+12
SALE_PRC	72000	2650000	399942	310000
LND_SQFOOT	1248	57064	8620.88	7500.00
TOT_LVG_AREA	854	6287	2058.04	1877.50
SPEC_FEAT_VAL	0	175020	9562.49	2765.50
RAIL_DIST	10.50	29621.50	8348.55	7106.30
OCEAN_DIST	236.1	75744.9	31691.0	28541.8
WATER_DIST	0.00	50399.80	11960.29	6922.60
CNTR_DIST	3825.6	159976.5	68490.3	65852.4
SUBCNTR_DI	1462.8	110553.8	41115.0	41109.9
HWY_DIST	90.20	48167.30	7723.77	6159.75
age	0.00	96.00	30.67	26.00
avno60plus	0.00	1.00	0.0149	0.00
month_sold	1.00	12.00	6.66	7.00
structure_quality	1.00	5.00	3.51	4.00

To also show correlations, we decided to focus only of the meaningful values with absolute values > 0.5. Table 3 shows all these highly correlated pairs of features.

When giving it some thought, it might not be unsurprising to see most features that are related to the geographical location of the property being highly correlated. For example the Miami city center is located directly at the coast. Therefore is only makes sense that `CNTR_DIST` (distance to the city center) and `WATER_DIST` (distance to the nearest body of water) are highly correlated.

Of higher interest are the correlations between the area of total living space and the sale price as well as the value of special features. Also the correlation between the age of the home and the distance to the city center seems to be meaningful.

We therefore formulate the hypotheses, that more living area and more special features in value have a significant effect effect on the price. Also homes closer to the city center seem to be older.

Table 3: High Correlations Between Variables

Column 1	Column 2	Correlation
LONGITUDE	LATITUDE	0.7212
CNTR_DIST	LATITUDE	-0.7173
WATER_DIST	LONGITUDE	-0.7643
CNTR_DIST	LONGITUDE	-0.7920
TOT_LVG_AREA	SALE_PRC	0.6673
SPEC_FEAT_VAL	TOT_LVG_AREA	0.5061
CNTR_DIST	WATER_DIST	0.5270
SUBCNTR_DI	CNTR_DIST	0.7664
age	CNTR_DIST	-0.5483

2.3 Data Quality Aspects

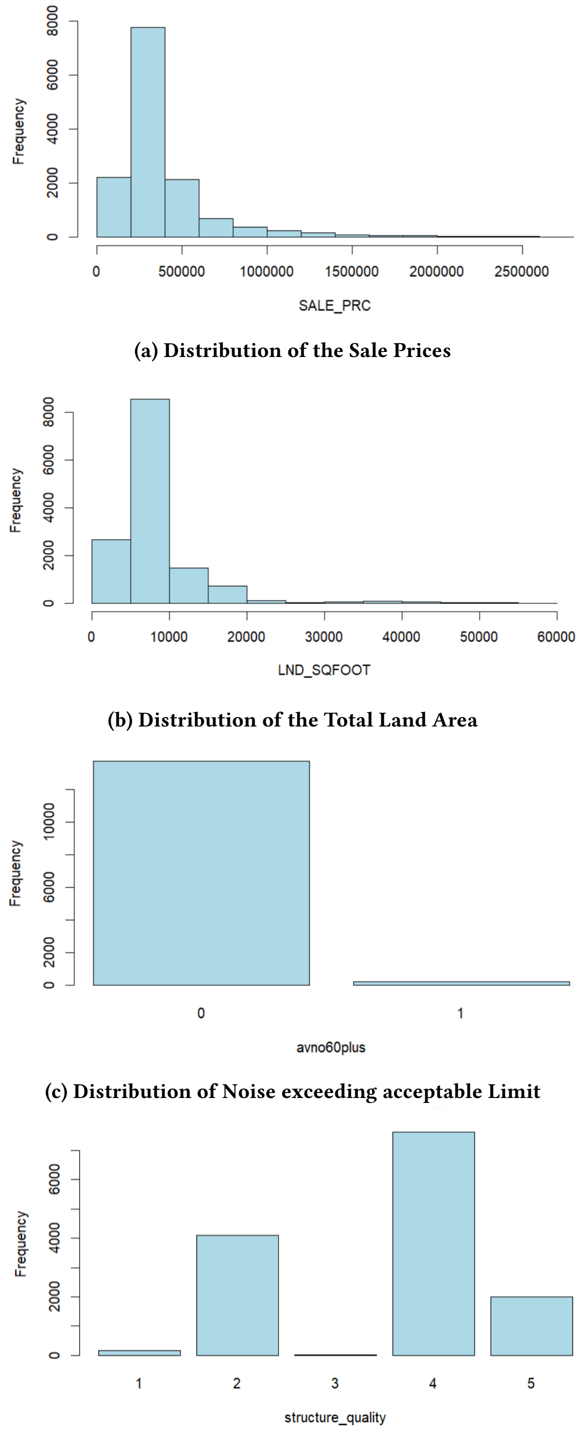
Fortunately, in our dataset there is no missing data present. Furthermore, information on data provenance and data cleansing applied before is sadly also not available and we can only assume no data has been artificially imputed or been created. After the first look on some statistical properties, especially regarding the value ranged, so far all values also seem plausible. One thing we did notice though were duplicates in the PARCELNO column. Since this should be an unique identifier for each property we investigated these observations more closely. In fact, all values in the affected rows were identical, except sale price and sometimes also month sold. We concluded from this, that some homes were simply sold multiple times in 2016 or the real estate agency changed the price for some reason, e.g. not being able to sell for a too high price, maybe noticing actual value is higher and adjusting, etc. As already discussed in 2.2, some features have very uneven distributions. Especially for the response, e.g. the sale price, this will lead to problems if not addressed properly. Additional feature we did not identify before with an uneven distribution in the frequency of the structure quality grades. There are only 16 rows with grade 3 out of all 13,932 observations. Also there seem to be much more homes where the dummy variable for the noise level is zero than where it is one.

Regarding outliers, so far we could only identify one possible outlier. In the SPEC_FEAT_VAL, which is a very uneven distributed feature, with the minimum being 0 the maximum observations has a value of $175,020$ while the second largest is only $123,590$. For all other column there do not seem to be single observations standing out with really high or low values, but bivariate analysis may point out further outliers.

2.4 Visual exploration

First we can see the visualizations of the distributions of some selected features: The sale price (Figure 1a), the total land area of a property sold (Figure 1b), the dummy variable for noise exceeding the acceptable limit (Figure 1c) and the structure quality grades (Figure 1d). As we can now visually identify the distributions of the continuous variable are very right-skewed as argues before and the classes for both the noise dummy and the structure quality are very unbalanced.

Figure 1: Uneven Distributions of several Features



(a) Distribution of the Sale Prices

(b) Distribution of the Total Land Area

(c) Distribution of Noise exceeding acceptable Limit

(d) Distribution of Structure Quality Grades

Now we also want to visually explore some relationships between the variables we discussed before. In Figure 2a we can see the Total Living Area plotted against the Sale Price. We already know they have a high correlation and from this plot with the linear regression trendline we see they have some kind of influence on each other, though it is obviously not perfect colinearity.

It is also the same for Total Living Area vs Special Feature Value (Figure 2b) but here we can additionally see the outlier for the Special Feature Value we pointed out in subsection 2.3 with the unusual high value.

In Figure 2c we see the Boxplots of the distributions for the Sale Price based on the home exceeding the acceptable aircraft noise level or not. We remember that the overall frequency of the Noise exceeding was very skewed with way more homes being under the level (value 0). As we can see in this plot now (Outliers are not shown in the boxplot) the overall distribution seem very similar for both values. Even though the value frequencies are very uneven we therefore do not have to worry too much about this feature.

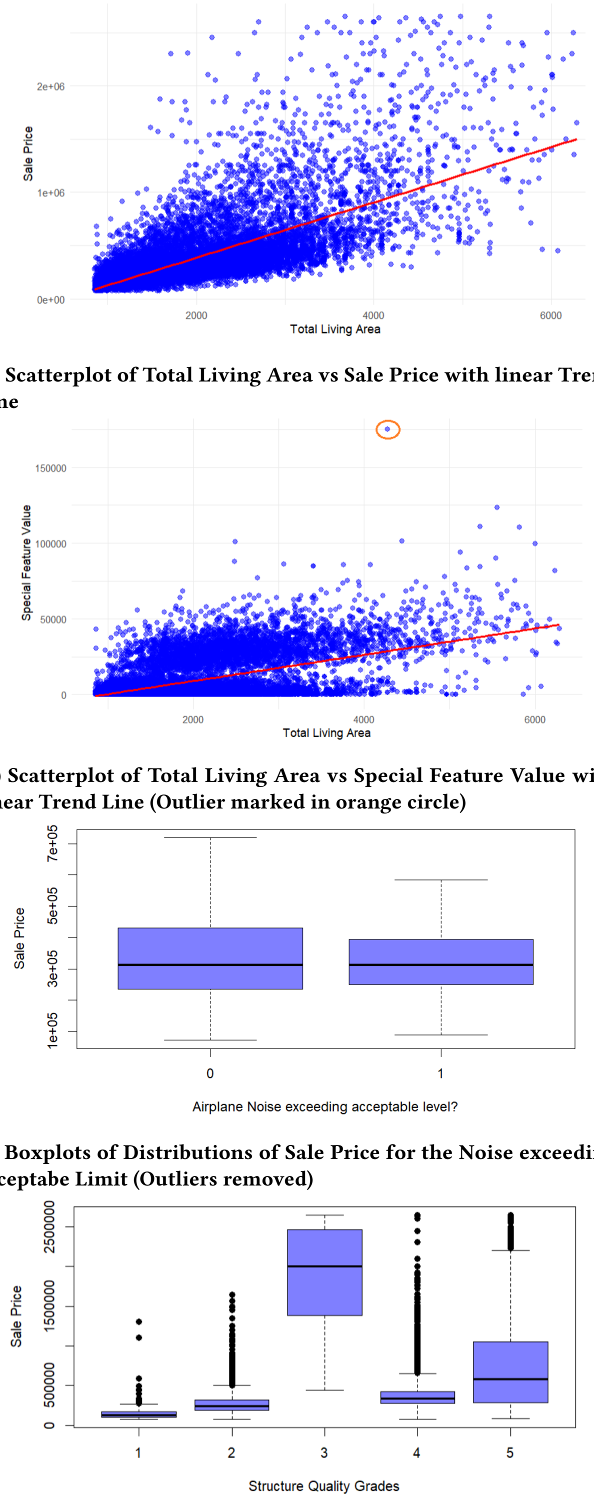
In Figure 2d we observe the distributions of Sale Price for each Structure Quality Grade form one to five. For all grades except 3 the result is somewhat to be expected in the median and interquartiles getting larger with a respective increase in the quality grade, even though there quite a few outliers everywhere but these can probably been explained due to some very large and special homes that are just more expensive. For the grade 3 though the the median and interquartiles are by far the highest and also with the biggest range of interquartiles. This is most probably due to the very small size of only 16 observations with respective grading of 3. That sample size is just way too small to result in an interpretable and useful distribution.

2.5 Evaluate for ethically sensitive and biased Data

Overall since we are not working directly with the personal data of persons the data is not overly ethically sensitive. There are still some concerns to be addressed though. We have the exact geo-spatial locations through the longitude and latitude data, as well as the parcel number ID and the sale price. Furthermore in combination with distances to water, city center, etc. this could be used to triangulate the exact property and by using the unique ID and the sale price this might be backtracked to the specific buyer/seller of the home. This would be data that should not be easily accessible for the public and falls under legal regulations. Since this is company-internal data and also buy doing all these aforementioned steps only doable by tremendous effort and by combination with additional user data we do not have to worry too much about this. Still using longitude/latitude directly in a regression is not that easy so we combine that with preventing aforementioned risks and group the geo-spatial data into more general classes like neighborhoods or Zip-codes.

For the structure quality grade, the noise level dummy and the aforementioned grouping by areas are still of mild concern since they could be used to predict socioeconomical poorer areas which might also overlap with specific demographics.

Figure 2: Relationships between some selected Features



(a) Scatterplot of Total Living Area vs Sale Price with linear Trend Line

(b) Scatterplot of Total Living Area vs Special Feature Value with linear Trend Line (Outlier marked in orange circle)

(c) Boxplots of Distributions of Sale Price for the Noise exceeding acceptabe Limit (Outliers removed)

(d) Boxplots of Distributions of Sale Price for each Structure Quality Grade

Additionally as mentioned before there are several continuous features that are distributed unevenly we will have to address.
For the two categorical features of quality grade and noise level the frequencies are also unbalanced but this tends to not be that big of a problem when doing regression on data. If this was a classification problem we would definitely have to address that problem further.

2.6 Potential Risks and additional Bias

Some additional potential risks and bias can always be underlying in the way the data was collected and what was collected. Maybe the data of homes sold are mostly from some specific areas that have a better socioeconomic power on average. This could results when training the model to be biased towards this and not deliver fair results when it comes to areas with few or no samples. This also holds true in regards to the other features, for example only collecting data from larger homes, not collecting data near the airport, etc. Also the data could have historically already been biased in its property valuation such that a model trained on it would always carry on the same bias. All this could lead to more grave risks since the estate market has historically always been volatile. To gain insight into such potential biases and risks we would need to consult with experts on the local real estate and also perhaps some official census bureaus to gain insight on historical as well as demographic data. Only then it would be possible to circumvent such risks completely.

2.7 Recommended Preprocessing

Based on the previous sections we first have to address the duplicates in `PARCELNO`. Since these observations are not complete duplicates but stem from homes being sold multiple times in the year or just having the price changed, we suggest it's best to maybe just keep one record or take the average for the observations with the same identifier.
Outliers are present but they also seem to actually carry information and not originate form faulty data, so it would be better to keep them.
geo-spatial data is more difficult to do regression on, since the relation to the response can never be linear, e.g. longitude might results in higher sale prices in specific intervals, but never in a linear way. Also there is the ethical concern that with the exact geo-coordinates some properties can be identified specifically. Therefore some derived values like the Zip-code of the home or resulting from that the general area description might be better. These classes can also be easier interpreted than longitude and latitude.
Furthermore as we have shown quite a few features have very right-skewed distributions. To handle this some transformation like log should be applied. Since many features are similar, like all features that measure the distance to something in feet or the features that all give some specific feet squared area, when transforming one of these it might be better to transform all to keep the same scales. The target variable Sale Price is skewed as well, but we will refrain from transforming it since it is our goal to predict the prices on the original scale as well as possible and a model that was fitted for a transformed target variable does not necessarily translate to a good model on the original scale.
In the end unnecessary columns need to be removed and categorical features encoded. The longitude and latitude features are not needed anymore after deriving another class feature from them and also the parcel number identifier carries no further information for the regression. For the Structure Quality Grade and the new categorical area feature some One-Hot encodings need to be applied to create dummy variables. The Airplane Noise dummy already is encoded and needs no further transformation.

3 Data Preparation report

Responsible: B

3.1 Preprocessing

3.1.1 Duplicates in `PARCELNO`*.* First we handled the duplicates in the column `PARCELNO` which is a unique identifier for each property. Since some properties are sold or have their price changed multiple times in the timespan of our data, the yare appear multiple times. We think this might be a problem since all the other values are the same for the observation and can lead to inconsistencies for the model. Our reasoning is that we want to predict the price for future sales as accurately as possible, so we therefore keep the observations with the later month sold and remove the earlier. Some homes have been sold multiple times in the same month. Since we don't know which the later one is in this case, we take the average sale price of both and aggregate the into one observation.

3.1.2 Removal of Identifier column. We remove `PARCELNO` since it carries no predictive power for a regression model and would just lead to a more complex model for no additional gain.

3.1.3 Transformation of skewed distributed Variables. As seen in subsection 2.3 many features have very right-skewed distributions which can impact models by violating the assumption of normally distributed residuals, leading to biased estimates and reduced model performance. We therefore use a Log-transformation on the skewed features. Since many features share similar semantics, e.g. measuring the distance to some Point of Interest or the area of the property, we also transform variables that do not have a very skewed distribution to give similar features the same interpretability. The affected features are `WATER_DIST, TOT_LVG_AREA, LND_SQFOOT, SPEC_FEAT_VAL,n RAIL_DIST, OCEAN_DIST, CNTR_DIST, SUBCNTR, HWY_DIST, age`. For the features with zero-values, namely age, SPEC_FEAT_VAL and WATER_DIST we used the $log(x+1)$ transformation instead of the normal $log(x)$.

3.1.4 Scaling. For regression models in general scaling is an important aspect since it ensures that all predictors contribute equally to the model's performance and convergence during optimization. We use the `scale` function in R which centers the data around its mean and scaling it for unit variance. This is accomplished by substracting the mean and dividing by the standard deviation. We do this for all numerical Ratio columns (except the respsonse), e.g. all but `avno60plus, structure_quality` and `month_sold`. Longitude and Latitude also not which we will come to in the next section where a new attribute is derived and they will b e removed. In actuality almost all of R's function, base and additional packages, apply scaling internally. We therefore demonstrate this step but continue with the unscaled data.

3.2 Further Preprocessing considered

3.2.1 Outlier Removal. After inspecting the data we determined that the identified outliers represent genuine variations rather than data entry errors or anomalies. These extreme values could identify unique property features or exceptional market conditions that are relevant to the analysis. Removing them would result in the loss of valuable information and potentially weaken the model's ability to generalize across all types and segments.

3.2.2 Feature Selection. All available features were retained in the dataset as we deemed each potentially relevant to predicting the sale price of homes. Instead of removing attributes at this stage, we decided to allow the model to determine the significance of each predictor. Feature importance metrics and regularization techniques can be employed in next stages to identify and retain the most influential variables, ensuring that valuable information is not prematurely discarded.

3.2.3 Creating Interaction Terms. Since many features share semantic similarities we considered to introduce interaction terms. While interaction terms can enhance complexity and capture underlying relationships, their creation was postponed to the exploratory modeling phase. Introducing interaction terms prematurely could lead to an overly complex model with increased risk of overfitting. Instead, our focus remained on ensuring the foundational preprocessing steps were robust, which allows for the informed introduction of interaction terms based on initial model performance and residual analyses.

3.3 Analyzing Options for derived and potential Attributes

We first wanted to transform the geo-spatial data longitude and latitude since it hard to interpret for models. We therefore decided on first converting these into the respective zip-codes of the properties. To do so we used the packages `sf, tigris` to first transform them into geo-spatial objects and then use ZIP Code Tabulation Areas (ZCTAs) to align them and get the zip-codes. After this we also removed the longitude and latitude columns because of the mentioned aspects. Now in the next step we wanted to further cluster the areas since the last step left us with over 70 unique zip-codes which would have meant a high increase of dimensionality when encoding them with One-Hot Encoding. We used maps of Miami to manually map them further into 13 general regions in Miami. This new categorical variable of the regions then also needed to be encoded by One-Hot Encoding.
Further derived values without external additional data sources were considered but we did not really come up with something we considered of a high potential.

3.4 Analyzing Options for additional external Data Sources

We came up with quite a few additional indicators that could be implemented by using external data sources. Since we already implemented the mapping to specific regions (subsection 3.1) it would now be easy to also include for these socioeconomic indicators, such as median income, employment rates, school quality ratings in the area and local crime rates. We ultimately decided for now to not include these since they would pose serious ethical issues by discriminating against poorer areas and maybe also areas with specific demographics. In the model exploration phase we might include for some test some of these indicators to compare the model quality, but ultimately the final model that would go into production should be discussed with legal affairs of our scenario-company first.

4 Modeling

Responsible: A

4.1 Data Mining Algorithms

The first suitable data mining algorithm that comes to mind is some form of a linear regression model. This model class is always a good first approach because the models are simple and interpretable. Models should not be made more complex than they need to be and often a linear model is more than enough to gain create a good regression model. On the other hand it underlies a linear relationship between the features and the target variable and is sensitive to outliers and skewed distributions. It might also not handle data that has more complex underlying relationships. Even though we are wanting to transform the target variable, which we moved until after the stratified data partition, that handles the skewed distribution, it might still not be enough. Also after the first data exploration we feel that the underlying structures in the data are more complex than a linear model could handle.
Another consideration were models from the gradient boosted tree class. These can easily handle outliers, skewed target distributions and non-linear relationships. On the other hand they can more easily overfit the models and are also not very interpretable.
In the end we settled on **Random Forest Regression**. This Ensemble method that is a forest made up of decision regression trees can also easily handle outliers and skewed distributions, which were both considerable problems for us. Also scaling is not needed. They are a good choice for generalization and a good baseline in regression. A disadvantage is that they can be computationally expensive, but our chosen dataset is not large enough that this would prove a problem. They also reduce overfitting more than single decision trees would while still being able to handle more complex relationships in the data. Due to these points we chose this model.

4.2 Hyper-Parameters

We want to now select a hyper-parameter for tuning. We use the `ranger()` function in `R`, which has multiple parameters that are eligible for tuning. These are:

(1) `num.tress`: The number of trees to grow in the forest.
(2) `mtry`: The number of features randomly selected at each split in a tree.
(3) `min.node.size`: The minimum number of observations in terminal nodes (leaf size).
(4) `max.depth`: The maximum depth of the trees.
(5) `splitrule`: The splitting criterion.
(6) `sample.fraction`: The fraction of observations used for training each tree.

We chose to tune the hyper-parameter `max.depth`. Our housing data has a heavy right tail due to some very expensive, luxury homes. this and also our data exploration indicated that there is a mix of common patterns for the lower to mid-range priced homes and rare more complex cases that are the luxury homes. These high priced homes have way more variance in most features but only make up less than 5% of all the sales recorded. Tuning the maximum depth of the trees directly controls the complexity. To limit it can prevent overfitting on the small subset of luxury homes while still allowing the model to capture significant patterns in the broader dataset where the bulk of the observations are lower to mid-range priced homes. Also this parameters offer more direct and interpretable way to control the complexity of our model, which is important in cases like our data where outliers and skewed distributions need to be handled, because some problems still remain even after transforming the features. Therefore we think the maximum depth is a good choice for tuning a hyper-parameter.

4.3 Train/Validation/Test Partition

Next we needed to handle the data partition into train/validation/test sets. We used the `caret` package that provides stratification in the splits, meaning the splits should keep up a similar distribution in the target column of the sale price for every part. This prevents generating very different sets in terms of the distribution which could lead to poor and unusable models/results. Furthermore we set a fixed random seed for reproducibility in the splits. The split sizes we settled on were a 80%/20% for $(train + valid)/test$ and further again 80%/20% for $train/valid$. These are usual values for the splits and will also be employed in our models.

4.4 Model Training

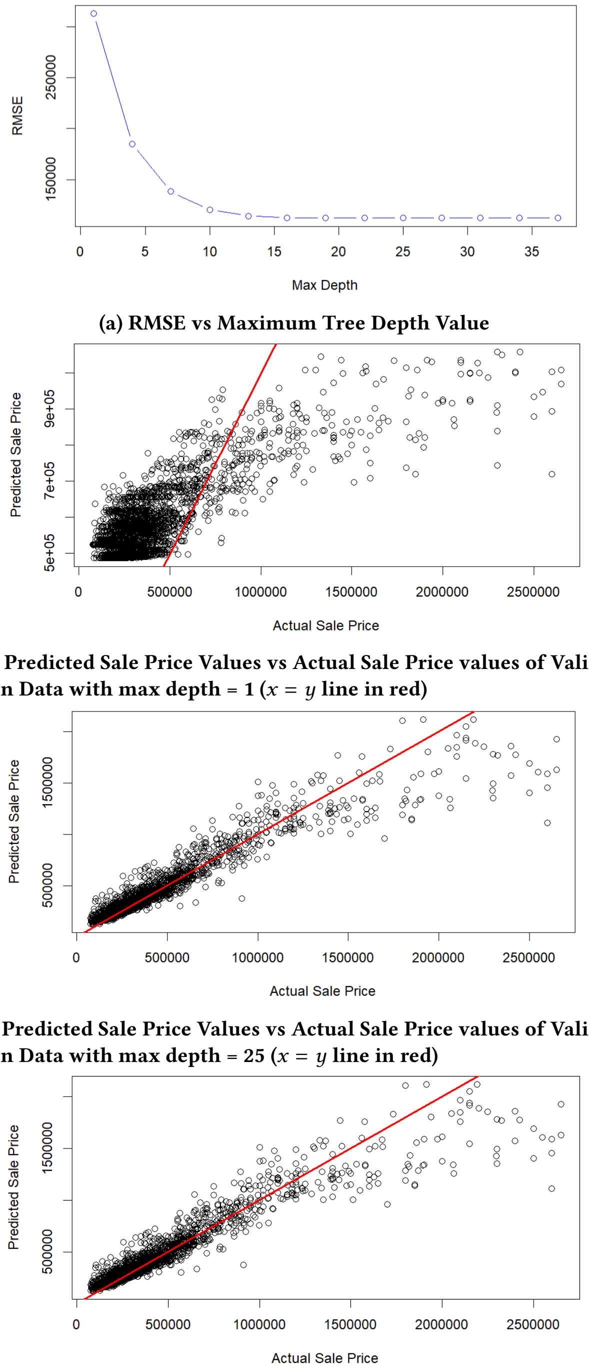
In the model training we then needed to identify the optimal value for our chosen hyper-parameter by hypertuning it. We set up a sequence of values for the maximum tree depth from 1 to 37, which covers a wide range from small values for models that will have very simple underlying logic to large values that can produce complex models. Additionally in a first step we introduced weights in the form of $\frac{price\ of\ home}{max\ priced\ home}$ to give the complex and rare cases of expensive homes more weight. This immediately greatly improved the performance of first tests without any other parameter optimizing. We therefore will keep these weights fixed from now on. The other parameters were fixed to their default values. These are:

(1) `num.tress`: 500
(2) `mtry`: $\lfloor\sqrt{number\ of\ predictors}\rfloor$
(3) `min.node.size`: 5
(4) `splitrule`: variance (splitted regarding to lowest variance)
(5) `sample.fraction`: 1 (when sampling with replacement, which is also done by default)

4.5 Report of Performance Metrics

In Table 4 we can see the RMSE, R^2 ad adjusted R^2 values we obtained in the tuning process for different values of the maximum depths, while all other parameters stayed fixed. Furthermore, in Figure 5b we see the reisudal plot for the validation set predictions for max depth = 1, in Figure 3c for max depth = 25 and in Figure 3d for max depth = 37. We can see that in Figure 5b there seems to be some big underlying structures the model does not catch. This is probably due to overfitting since max depth = 1 results in logically very simple models. On the other hand for max depth = 25 and = 37 the residual plots look almost the same.

Figure 3: Hyper-Parameter Tuning Process Figures



(a) RMSE vs Maximum Tree Depth Value

(b) Predicted Sale Price Values vs Actual Sale Price values of Validation Data with max depth = 1 ($x = y$ line in red)

(c) Predicted Sale Price Values vs Actual Sale Price values of Validation Data with max depth = 25 ($x = y$ line in red)

(d) Predicted Sale Price Values vs Actual Sale Price values of Validation Data with max depth = 37 ($x = y$ line in red)

Table 4: Summary Statistics for Miami Housing Dataset

Maximum Depth	RMSE	R^2	$adjusted\ R^2$
1	313057.5	0.1341	0.1267
7	138237.8	0.8312	0.8298
13	114392.7	0.8844	0.8834
22	112186.3	0.8888	0.8879
25	112173.8	0.8888	0.8879
37	112172.4	0.8888	0.8879

Also in Table 4 the evaluation metrics have almost the same values. This means that for max depth = 25 the model is already catching all the complexity it possibly can by tuning this parameter, while increasing the parameter further only results in more computation time and no significant gain. Therefore we choose 25 as the optimal parameter value as everything smaller is too simple and everything larger does not give any significant increase in model performance.

5 Evaluation

Responsible: B

5.1 Apply the final model

We will consider the model with the hyperparameter of maximum depth of 25 as the final model to perfume test data set on it .In Table 5 we can see the RMSE, R^2 ad adjusted R^2 values we obtained with the test dataset predictions and the actual values of them.

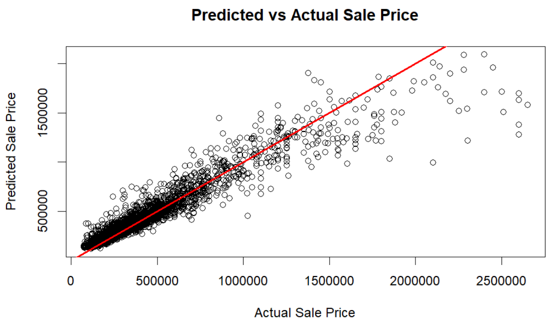
Table 5: Test Dataset Results

Metric	Test	Validation
RMSE	99525.25	112173.8
R-squared	0.8966403	0.8888
Adjusted R-squared	0.8960362	0.8879

As we can see the test dataset results are slightly better than the validation result indicating good generalization.

In the next plot we can see the result with the prediction and the real values of the test dataset.

Figure 4: Test Dataset Result



5.2 Re-training on Combined Training and Validation Data

In this part we will combine the train dataset and the validation dataset and train the model again with the hyperparameters we gained in the tuning part. We also have to update the weights with the combined dataset.

Table 6: Test Results: Separate vs Combined Training Data

Metric	Separate Train/Validation	Combined Train/Validation
RMSE	99525.25	90082.29
R-squared	0.8966	0.9153234
Adj. R-squared	0.8960	0.9148284

As we can see the test results with the combined dataset of train and validation show better results.By combining the training and validation datasets, the model has more data to learn from and also larger datasets typically reduce overfitting and improve the model's generalization to unseen data, which leads to better test performance here.

5.3 State-of-the-art performance and Base-Line Performance

State-of-the-art performance

We found this project ***Kaggle link*** as a reference for the state-of-the-art performance. It uses the same dataset and almost identical preprocessing steps, with a slight difference in scaling. Specifically, it applies Min-Max scaling for preprocessing the features. For training the model, it employs a neural network architecture. It also does not use any validation for training and only splits the data into train and test.

Baseline Performance Baseline performance refers to the evaluation of trivial models that make predictions without learning patterns from the data. These models provide a reference point to assess whether the trained model performs meaningfully better. In this task, we used the following baseline models:

Mean Predictor The mean predictor is a simple model that predicts the mean of the target variable (*SALE_PRC*) from the training set for all instances. This model represents a trivial approach to regression tasks and serves as a lower bound for performance.

Median Predictor The median predictor predicts the median value of the target variable (*SALE_PRC*) from the training set. This baseline is less sensitive to outliers compared to the mean predictor and provides another reference for evaluating the trained model. The results of these baseline models and their comparison with the trained model are discussed in the next section.

5.4 The performance Achieved With The Benchmark and Baseline

The performance Achieved With The Benchmark

Table 7: Final Model vs The Benchmark

Metric	Benchmark	Final Model
RMSE	10184.458	90082.29
R-squared	0.998927	0.9153234
Adj. R-squared	0.99892	0.9148284

The benchmark model, a neural network with multiple dense layers and 21,801 trainable parameters, significantly outperforms the implemented model in terms of RMSE (10,184 vs. 90,082) and R-squared (0.9989 vs. 0.9153). This is due to the neural network's ability to capture complex, non-linear relationships in the data, making it highly effective but computationally intensive. In contrast, the implemented model, while faster and simpler, struggles to achieve the same level of precision, highlighting the trade-off between complexity and efficiency.

Baseline Models

Table 8: Final Model vs Baseline Models

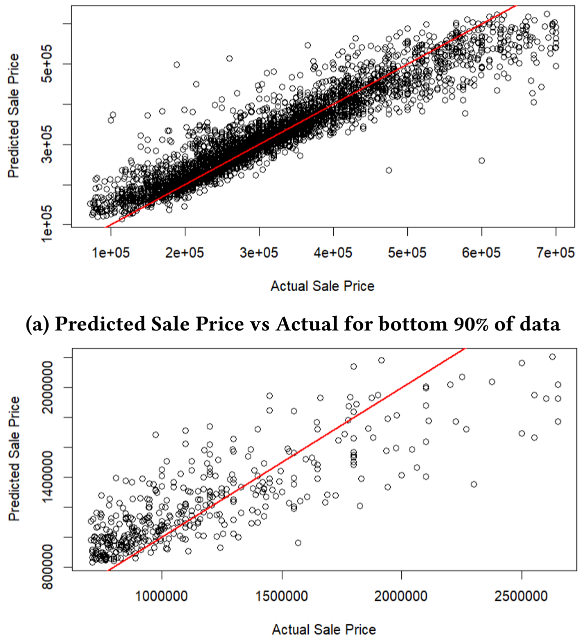
Metric	Final Model	Mean Baseline	Median Baseline
RMSE	90082.29	309587.1	321473.7
R-squared	0.9153234	-0.0001158021	-0.07838916
Adj. R-squared	0.9148284	-0.0001158021	-0.07838916

As we can see our final model outperform significantly the baseline models in all three metrics. As we noticed there are significant differences in performance in the dataspace, namely for low and high values of *SALE_PRICE*. We therefore also compare the performance of our model to the baseline and benchmark for two subspaces. To do so we divide the data at the 90% quantile which is around 700, 000$ for the Sale Price and respectively predict on each group. As we can see in Table 9 and Figure 5 the performance differs very significantly for these two groups.

Table 9: Final Model vs Baseline Models

Metric	bottom 90	Top 10	Mean B.	Benchmark
RMSE	44864.81	309587.1	254874.4	10184
R^2	0.8684	0.6451	-0.0784	0.9989
$Adj.R^2$	0.8675	0.6249	-0.0784	0.9989

Figure 5: Residual Plots for Data at the 90% Quantile



(a) Predicted Sale Price vs Actual for bottom 90% of data

(b) Predicted Sale Price vs Actual for top 10% of data

5.5 The performance obtained with the success criteria defined in the Business Understanding phase

In the business understanding phase, we described that the main goal is to predict the actual values of homes as precisely as possible. From the results we see in our best model and the metrics that we gained (RMSE: 90,082, R-Squared: 0.91, Adj R-Squared: 0.91), we can realize that the error represents approximately 18% of the typical price, which is reasonable for real estate predictions. The model explains 91.53% of the variance in home prices, meaning it captures the majority of factors influencing price.

As we can see in Figure 4, the model may still have larger prediction errors for luxury apartments due to their rarity and complexity.

5.6 Bias Evaluation for the Protected Attribute *avno60plus*

To evaluate whether the model exhibits bias, we selected the attribute *avno60plus* as the protected attribute. This attribute indicates whether airplane noise exceeds acceptable limits (*1*) or not (*0*). The data distribution for this attribute in the training and test datasets is as follows:

- **Training Dataset:**
  - Class 0: 9,517 samples
  - Class 1: 128 samples
- **Test Dataset:**
  - Class 0: 4,053 samples
  - Class 1: 78 samples

As shown, the data is highly imbalanced, with a significantly lower number of samples in Class *1*.

We evaluated the model's performance separately for each group identified by the *avno60plus* attribute. The results are shown in the table below:

Table 10: Model Performance for *avno60plus* Groups

Region (*avno60plus*)	RMSE	R-Squared
0	90,094.79	0.9165
1	89,430.55	0.5668

- For Region *0* (*avno60plus* = 0), the RMSE is 90,094.79, and the R-squared value is 0.9165. This indicates high accuracy and good variance explanation.
- For Region *1* (*avno60plus* = 1), the RMSE is 89,430.55, but the R-squared value drops significantly to 0.5668. This suggests that the model struggles to capture the variance in this subgroup, despite having a similar RMSE.

6 Deployment

Responsible: A + B jointly

6.1 Addressing Business Objectives and Recommendations for subsequent Analysis

Comparison of Performance with Business Objectives: To quickly summarize our in subsection 1.2 specified Business Objectives again we mainly wanted to help our agents to make better decisions based on our analysis, as well as identifying over- and undervalued properties. Furthermore the whole process of the pricing analysis was supposed to be more efficient by (partly) automating it by using our model.

As we could observe in the evaluation in section 4 the RMSE for test predictions using the tuned model trained on the whole train+valid was ≈ 90000$. We remember form Table 2 that the median of the Sale Price was about 400, 000$. The RMSE is therefore relative to the Sale Price not small enough that a fully automated pricing strategy using our model is justified. The errors made would be too significant to do so. As analyzed before though this RMSE value can be further explained by analyzing the data: While the maximum price in the data is 2, 650, 000$, the 90% quantile is at 705, 000$, which again shows the skewed distribution. Looking at the residual plot Figure 5 we can observe that the variance of the residuals seems to be increasing with the sale price. While for up to about 800, 000$ the residuals seem normally distributed and showing a worrying increase in variance, for the large price values this does not hold. As we remember due to this we again fitted the model to the two subspaces of the data that are the bottom 90% and the top 10% in terms of Sale Price. Here we can observe that for the subspace using the lower priced homes only, the RMSE is substantially lower and for the higher priced homes substantially larger compared to the whole data. We have the strong presumption that this is due to "normal" homes that are more similar in features and therefore also price, show more easily general patterns and can therefore be identified and regressed on better. For the more expensive homes though it is more complex, because for once there is just way less data to train on and also the more luxury and expensive a home, the more unique it is. We could also observe in the data that unexpectedly the mroe expensive homes way more often had higher values in terms of the special features (pools, special rooms, sauna, etc.). These cases therefore become way more complex to predict and combined with the fewer data available it makes sense that a very complex model is needed for these.

Recommendations for Deployment: Based on this analysis we recommend a hybrid solution. For the properties up to 800, 000$, our models predictions are very solid. This can help as a good baseline for our pricing department to start from and maybe make some small further adjustments, based on their experience and personal assessment. For the high priced homes, our models' prediction can still be used as a very first baseline estimate, but there should be put way more weight on manual analysis and experience of our agents. Since the expensive homes are way more unique they simply require more analysis and experience in the field to handle. We still think this is a good solution since by deploying our model for the bulk of the sales we can improve efficiency significantly by not needing manual analysis for every property, while our agents can focus more of their time on the Sales that are more profitable.

This should be combined with model monitoring and a feedback loop that track performance (RMSE/R^2) over time and also retrain the model with new data to adapt to changing market trends.

Recommendations for Subsequent Analysis: The feature set can always be expanded by moire indicators. The big problem simply lies in identifying which ones are significant and which ones are not. As we already discussed in subsection 3.1 there are many further feature that can be taken into account. Additional ones we did not discuss before might be macroeconomical indicators such as interests rates or inflation. Also just as in section 5 further subgroup analysis might help identifying patterns and improve performance. Also the data could be restructured to incorporate more time trends that help in forecasting new prices. Finally since we identified the problem of luxury homes being too unique to be accurately identified by not overly complex models, further external insights, e.g. real estate agents, economists or customer surveys might help getting new perspectives on the data.

6.2 Ethical Aspects, Impact assessment and Risks Identified in Deployment

Ethical Aspects: The model may unintentionally reflect biases present in the training data which could include undervaluing homes in marginalized neighborhoods. Furthermore if the model seems too intransparent and not well interpretable, it might be bad for agents, buyers and stakeholders to trust or understand it. To prevent this the mdoel should be regularly analyzed to prevent these biases to be included. Also a good and userfriendly UI and explanations have to be included for users.

Impact Assessment and Risks: In hte deployment stage an over-reliance on fully or smei-automated prediciotns could lead to uniform pricing strategies. This can possibly distort the market and also the market competition. also such a tool can be misused to infalte prices in specific regions. Furthermore as mentioned before our model struggles with niche and luxury homes. Relying on it too much for these properties could also lead to bad deciions for our own agency. Furthermore if our agents become overly reliant on it, they might neglect to gain experience themselves and some qualitative factors that can only be identified by manual analysis of experts suffer.

All these can be prevented and handled by enough human oversight and policies that handle the hybrid solution, to prevent an overreliance and misuse of our model.

6.3 Aspects to be Monitored During Deployment

The model should be trained regularly as mentioned in subsection 6.1 with new sales data. This will also result in updating model performance metrics, such as RMSE and R^2. These should be monitored for changes, especially when they get worse. Model drift, meaning the distributions of the features of input data change significantly also ahs to be monitored. Furthermore some analysis of the new data needs to be made to be sure all subgroups in the data are covered to not introduce new bias. End-user feedback should be taken into account and be monitored as well as satisfaction levels. The general market impact is also important to oversee, so changes in behavior as specified in subsection 6.2 that are due to the model can be identified and seen to properly. At last technical aspects, such as efficiency, stability and latency mof the deployed system need to be monitored to ensure user-friendly access and easy handling.

Triggers for intervention are therefore significantly increasing RMSE values, significant model drifts and also consistent underperformance in specific subgroups of the data that have not been yet identified in this process. To handle these scenarios the model would need to be trained again, with possible additional tuning and also new analysis of the feature space. Additional triggers can be low adoption rate for our agents/other users, negative feedback and also noticeable evidence of market distortion that could be due to the deployment of our model. In the data aspect an incrrease in inconsistent data, meaning more missing values and faulty data especially in key features should be recognized.

6.4 Reproducibility Aspects

We have covered all possible reproducibility aspects in our report. There is detailed information about the data preprocessing steps in subsection 3.1, we provide the exact dataset we are using in section 1 and the model in section 4. This mdoel includes the library and function used (R's `ranger()`) with additional hyperparameter settings. Furthermore since our model is a Random Forest, we specify in R's environment a seed (`set.seed(12345)`).

The persisting risks are first a change in the dataset from kaggle, even though we also provide the csv file in this submission. The biggest risk might be an updating of the existing libraries we used, as some routines and functioned could be changed/optimized and might produce different results after with the same settings.

Besides these risks we provide evrything from model trianing, evaluaiton metrics to data that ensure perfect reproducibility.

7 Summarized findings

Responsible: A + B jointly In this project, we used a random forest regression model to predict housing prices. The model worked well for lower to mid-priced homes, giving accurate predictions with low errors in these price ranges. However, it did not perform as well for luxury homes, where predictions were less reliable and had higher errors. This showed that the model struggles with very high-priced properties, and it may need additional features or a different approach to handle this segment better. Our main goal was to create a tool that could provide useful pricing insights and automate price predictions for the housing market. For most housing segments, we achieved this goal, meeting the success criteria we set. However, the model's limitations for high-priced homes suggest there is room for improvement to make it more reliable across all price ranges. The dataset we used had a good variety of features that were easy to understand, which made it easier to prepare and work with the data. During data preparation, we handled missing values, scaled the data (before knowing we did not need it), and selected features carefully to improve the model's performance. While these steps were helpful, we noticed that some attributes, were unevenly distributed, making it harder for the model to predict outliers accurately. Ethical concerns were an important part of our analysis. Although the model did not show clear bias toward sensitive attributes, its weaker performance for luxury homes could unintentionally disadvantage people or businesses dealing with high-priced properties. Transparency in the model's predictions is also an important challenge, especially for users who rely on these predictions for important decisions. For deployment, we recommend a hybrid approach. The model can be used for predicting prices of lower to mid-priced homes, while luxury properties should be reviewed by experts to ensure accuracy and trust. We also recommend regular monitoring of the model to address potential changes in data patterns or biases that could develop over time. We compared our model's performance with another study using the same dataset and found that it performed slightly worse. On the other hand the other study's model used neural networks and was therefore way more complex and hard to train. The advantage for our model is therefore its efficiency and mostly simple application. However, these comparisons also showed that other methods might handle high-priced homes better, suggesting that combining our approach with other algorithms (like gradient boosting) could improve results. In comparison to baseline trivial models like mean or median predictions our model performed very significantly better. Through this project, we learned that understanding the data is key to building a good model. Identifying and handling issues like missing values, outliers, and uneven distributions early on made a big difference in our results. Hyperparameter tuning and careful validation helped us get the best performance out of the model. For luxury homes, we realized that additional data, such as economic indicators or specific property features, could help improve predictions. Overall, we successfully demonstrated how random forest regression can be used for housing price prediction, meeting most of our goals. While the model works well for most homes, it needs improvement for luxury properties. Combining domain knowledge, additional data, and advanced algorithms could make it even more reliable and useful in real-world scenarios.