Housing Prediction

Great Learning

March 2023

hector.sanchez.johnson@gmail.com

Milestone 1

Initial EDA

Intro

- We are tasked to **predict housing prices** of the provided location and identify the **most important features** to consider.

- This study is required in order to locate **market opportunities that can be leveraged** favorably for business, social and personal reasons.

- Some opportunities that this may bring are:
  - Corporate **investments**
  - REIT portfolio **growth**
  - **Government** housing
  - **NGO** projects
  - **Individual** home purchase

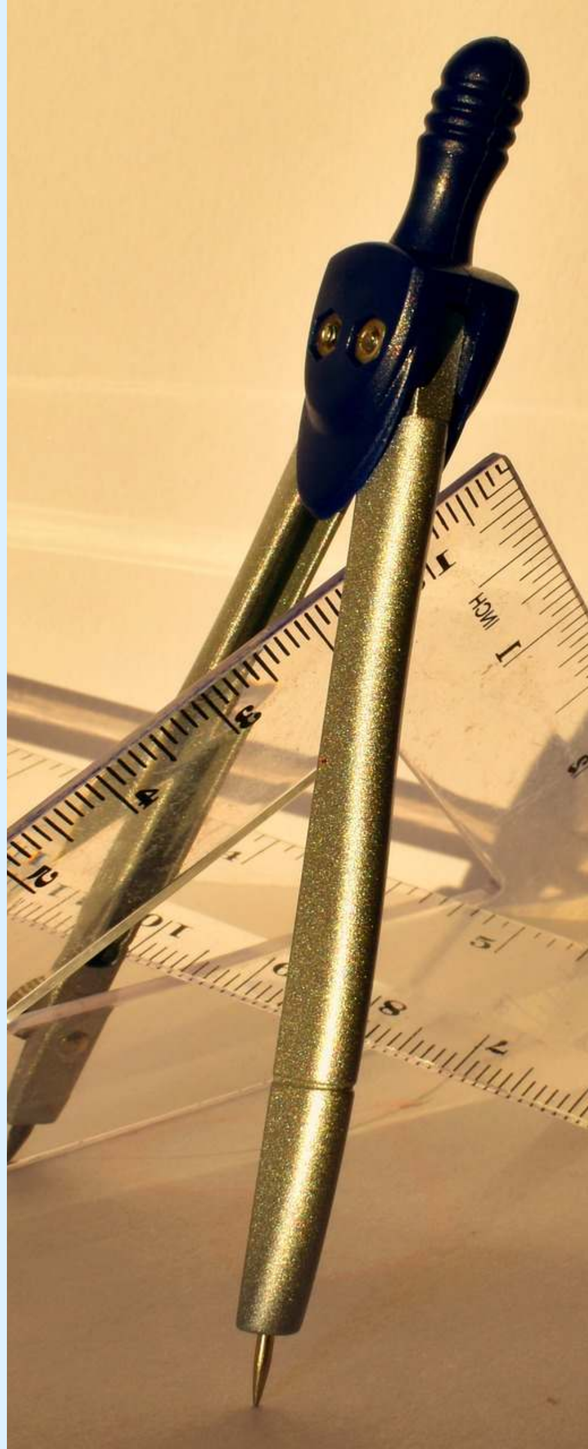
Overview

- The collected data includes information regarding **individual properties**.
- The **features consist of** coordinates, measurements for lot and building, number of rooms and baths, build and renovation dates, and other accessories.
- We can assume the data was collected through various **real estate agencies and government** documents.
- **Additional data** could be acquired through third party data providers (APIs) if needed.

Data

This dataset has 23 features:

- **cid**: a notation for a house
- **dayhours**: Date house was sold
- **price**: Price is prediction **TARGET**
- **room_bed**: Number of Bedrooms per house
- **room_bath**: Number of bathrooms per bedrooms
- **living_measure**: square footage of the home
- **lot_measure**: square footage of the lot
- **ceil**: Total floors (levels) in house
- **coast**: House with view to a waterfront (1/0)
- **sight**: Has been viewed
- **condition**: How good the condition is (0-5)
- **quality**: grade given to the housing unit, based on grading system
- **ceil_measure**: square footage of house apart from basement
- **basement_measure**: square footage of the basement
- **yr_built**: Built Year
- **yr_renovated**: Year when house was renovated
- **zipcode**: zip code
- **lat**: Latitude coordinate
- **long**: Longitude coordinate
- **living_measure15**: Living room area in 2015
- **lot_measure15**: lot size area in 2015
- **furnished**: Based on the quality of room (1/0)
- **total_area**: Measure of both living and lot



Review

- Total of **23 columns** with **21,613 entries**

- Only **‘dayhours’ (date)** is non-numeric
  - Some columns had nulls and incorrect values (“$”)

- Several are **categorical features**, but are measured numerically, such as:
  - coast, sight, condition, quality & furnished

- **Feature selection and engineering** will benefit from the following columns:
  - dayhours - separating Year, Month and Day
  - lat & long - for geo-grouping
  - zipcode

- **Dummy columns** and **one-hot encoding**:
  - coast & furnished are 1/0 for Yes/No
  - sight, condition & quality need dummies

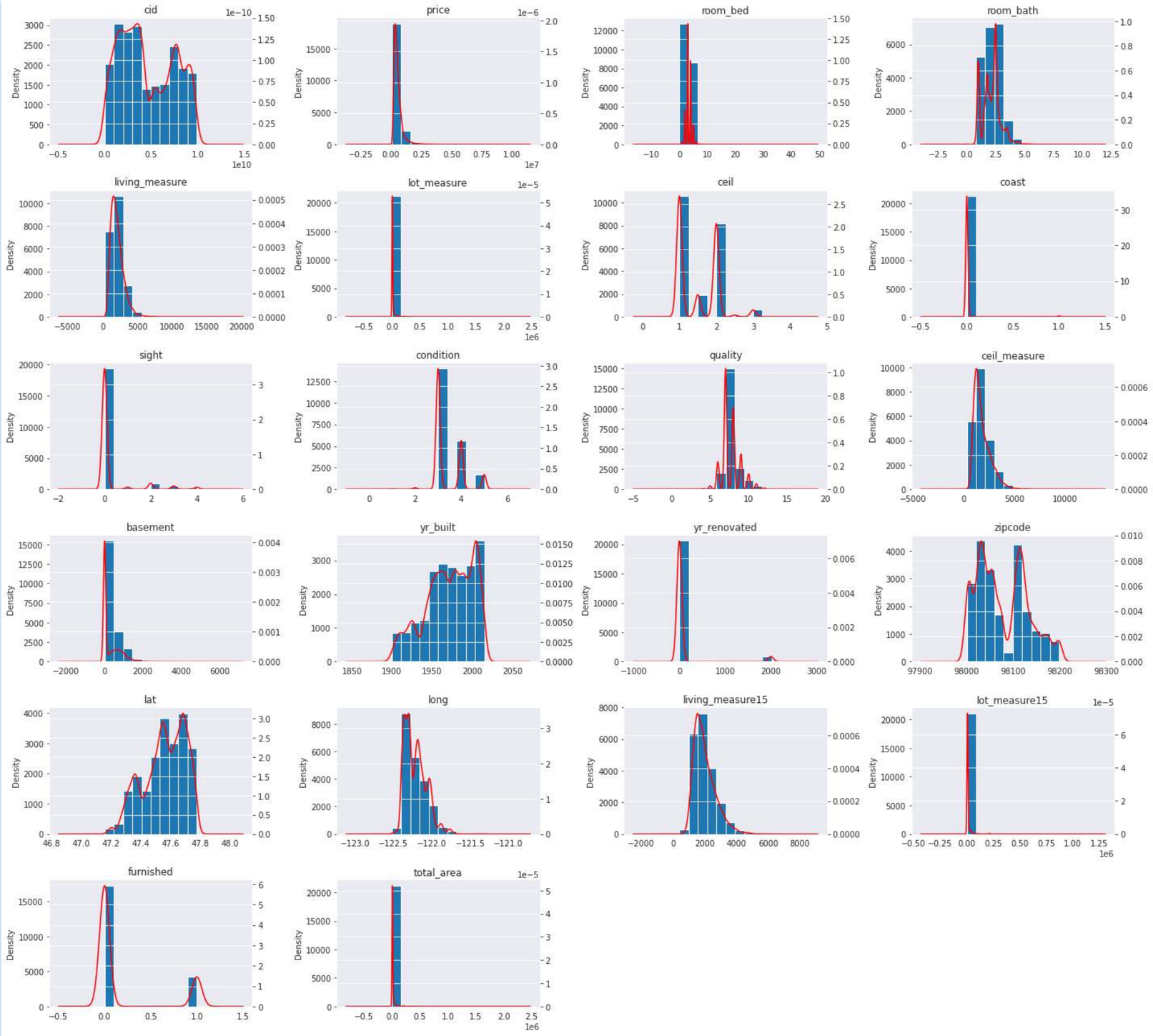
Important Stats

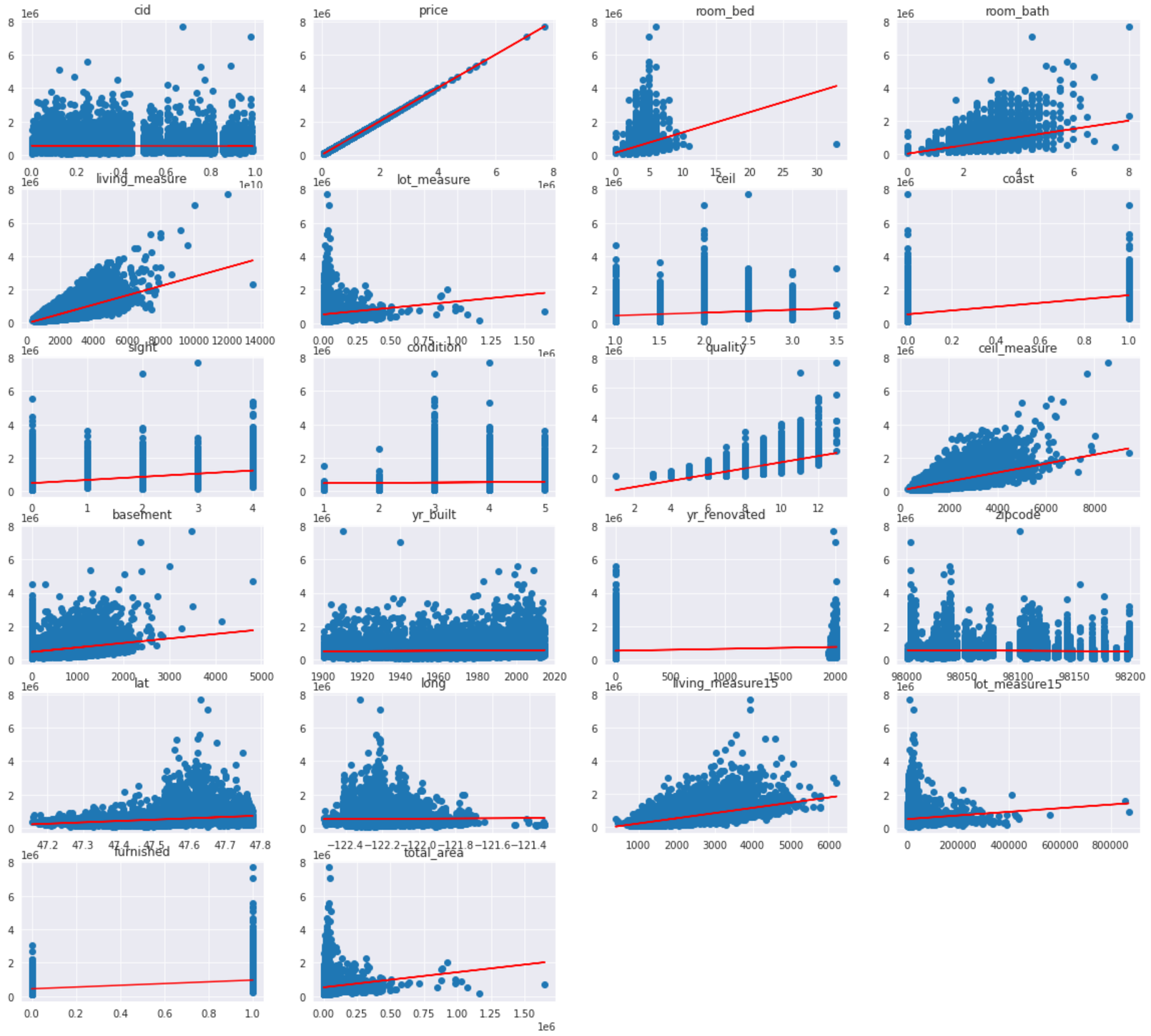
	MAX	AVG	MIN
price	$7,700,000.00	$539,936.33	$75,000.00
room_bed	33	3.3	0
room_bath	8	2.1	0
ceil	3.5	3.3	0
living_measure	13,540	2,079.9	290
lot_measure	1,651,359	15,098.8	520
sight	4	0.23	0
condition	5	3.4	1
quality	13	7.6	1
ceil_measure	9,410	1,788.7	290
basement	4,820	291.3	0
total_area	1,652,659	17,184	1,423

Univariate

Observations from **Data Describe** and **Distributions**:

- cid
  - Multiple sales for the same property
- room_bed, room_bath, living, lot & total_area
  - Show some larger buildings
- sight
  - Averaging 0.23 of 4.0 = not many views
- condition & quality
  - They're a little over 50%





Bivariate

From the **Scatter Plot** with **Linear Regression** vs the target variable, we observe the following columns have strong correlation (most to least):

- room_bed
- living_measure
- ceil_measure
- room_bath
- coast
- quality
- total_area
- basement
- furnished

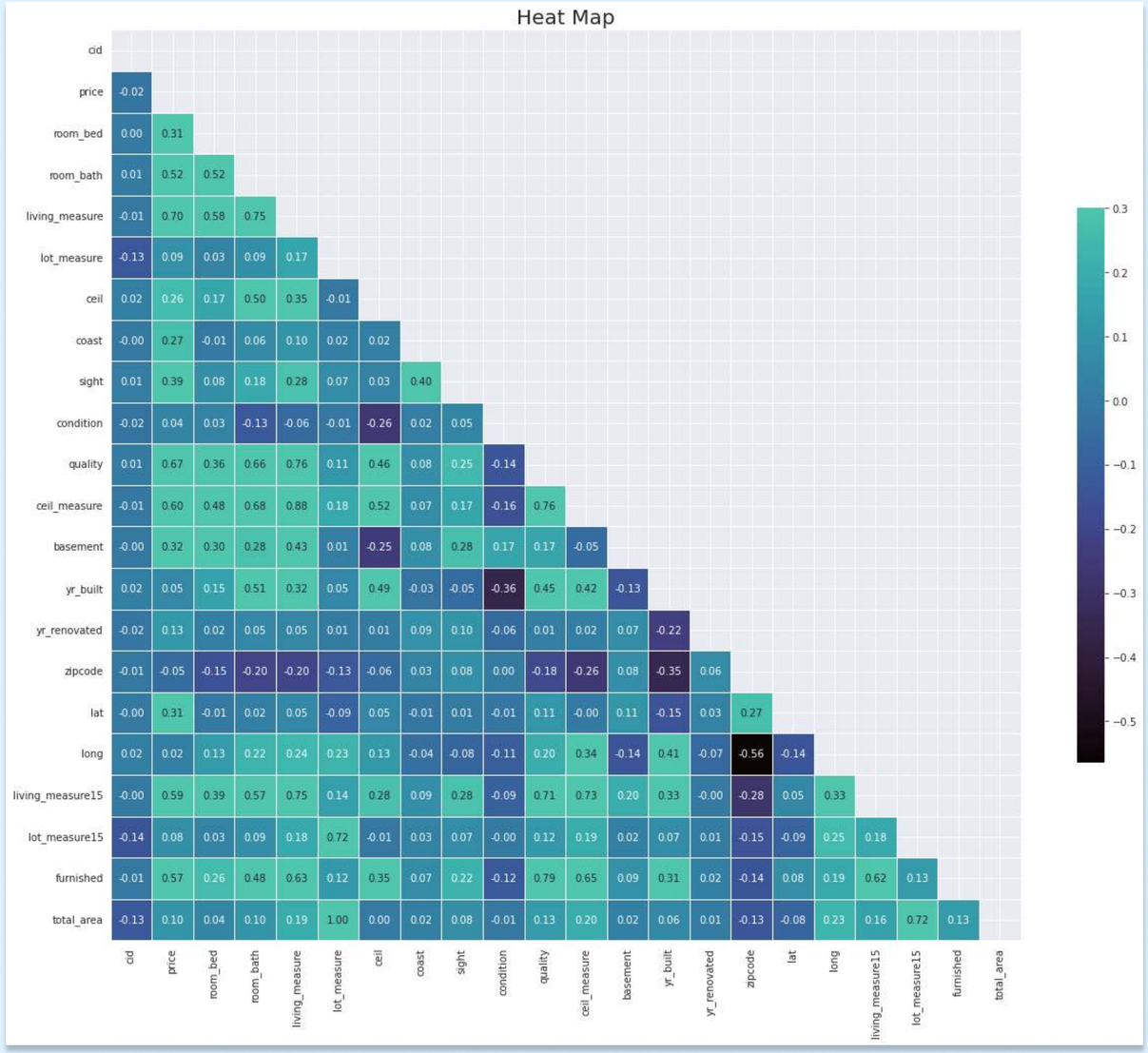
Heat Map

From the Heat Map we can observe the following **Correlations**:

- **Positive** correlations:
  - 1.00 - lot_measure / total_area
  - 0.88 - ceil_measure / living_measure
  - 0.79 - quality / furnished

- **Negative** correlations:
  - -0.56 - zipcode / long
  - -0.36 - condition / yr_built
  - -0.35 - zipcode / yr_built

- **Neutral**:
  - lat, yr_renovated, lot_measure15, living_measure15

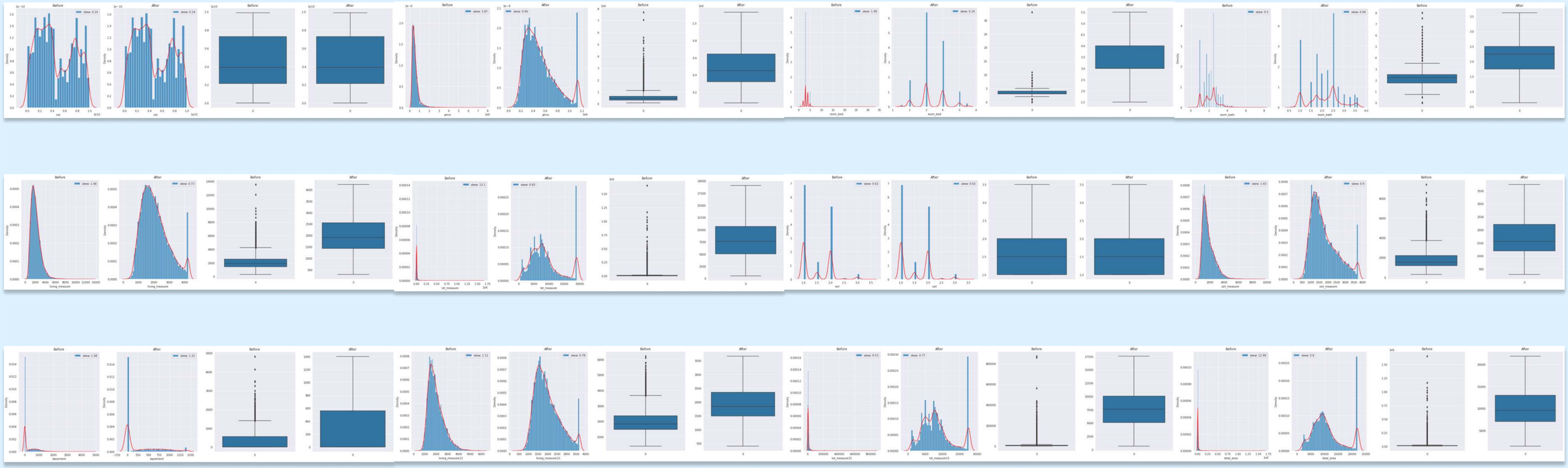


Milestone 2

Feature Engineering & EDA

Outliers

- We start by reviewing the **skewness** of the data (*Addendum: Data Skewness*)
- Since most data is very skewed we will use the **IQR** instead of the Z-Score for normal distributions



Features

- From the **dayhours** date column we split into Year, Month and Day and create the **sell_season**:
  - Spring, Summer, Autumn, Winter

- From **yr_built** we can bin properties into age categories of **property_age**:
  - Under 10 years, Between 10 and 30 years, Between 30 and 50 years, Over 50 years

- From the Real Estate industry we know:
  - Properties can be classified into **property_class** A, B, C and D based on **property_age** and **condition**
    - *https://realwealth.com/learn/classes-of-property-real-estate/*
  - Property values by **zipcode** can define **property_tiers** from 1 to 8
    - *https://thebasispoint.com/whats-the-most-popular-home-price-bracket/*
    - *https://uszipcode.readthedocs.io/index.html*
  - The IRS tracks the median **income_tier** per household by **zipcode**
    - *https://www.irs.gov/pub/irs-soi/20zpdoc.docx*

Features

property_class

- **A**: Under 10y, condition 5-3, otherwise Class B
- **B**: Between 10-30y, condition 5-3, otherwise Class C
- **C**: Between 30-50y, condition 5-2, otherwise Class D
- **D**: Over 50y, condition 5-1

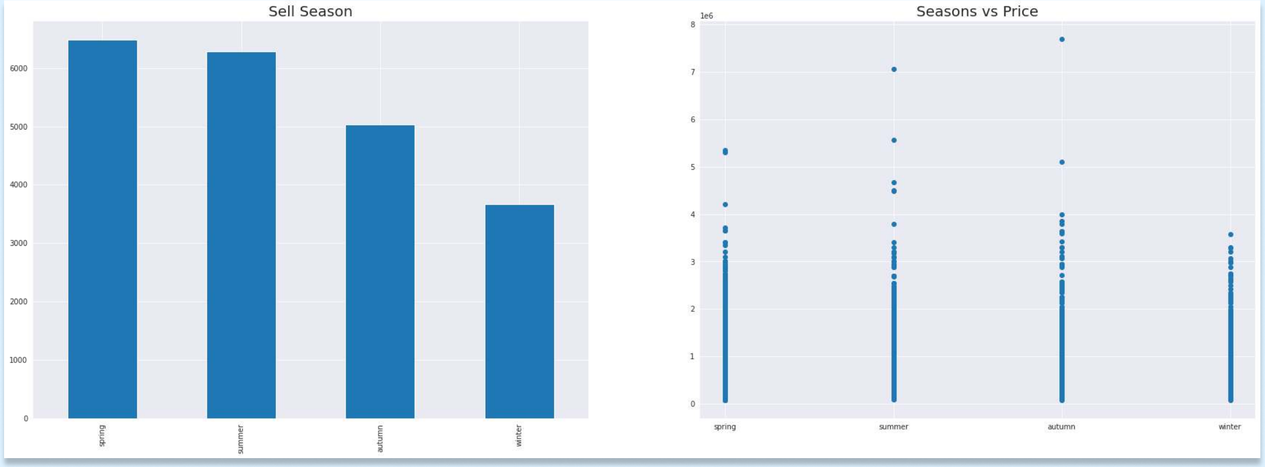
property_tier

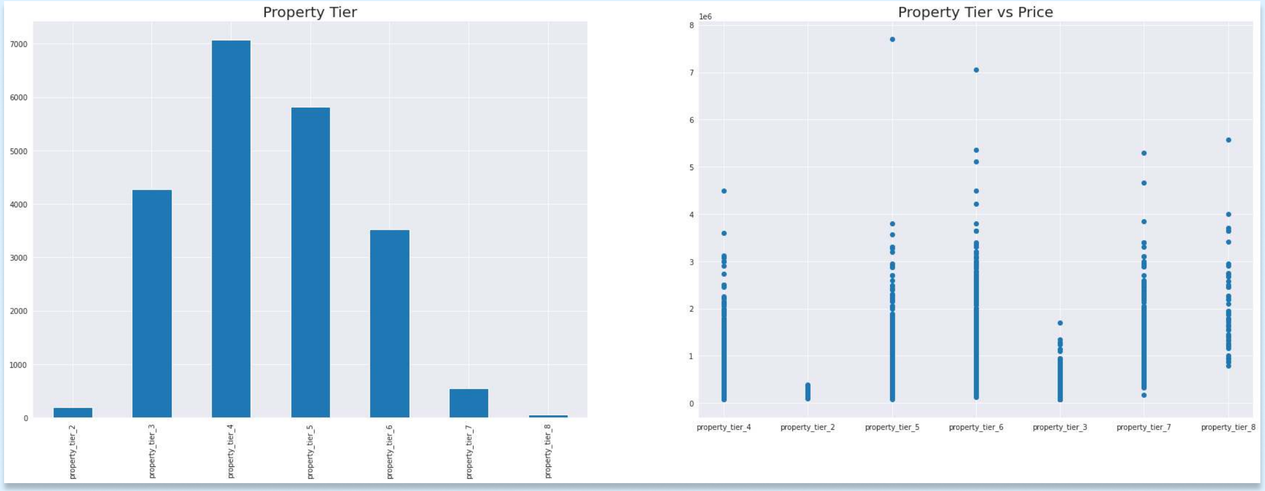
- **1**: Under $150k
- **2**: Between $150k - $199,999
- **3**: Between $200k - $299,999
- **4**: Between $300k - $399,999
- **5**: Between $400k - $499,999
- **6**: Between $500k - $749,999
- **7**: Between $750k - $999,999
- **8**: Over $1M

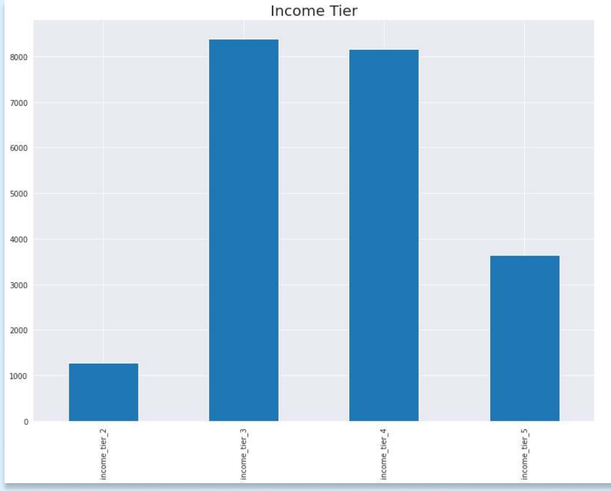
income_tier

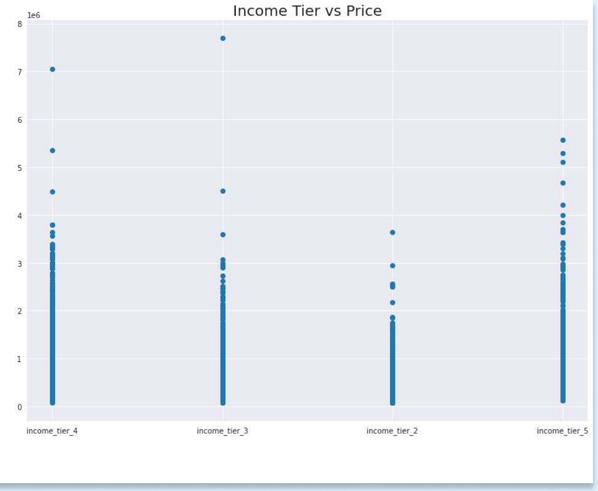
- **1**: Under $25k
- **2**: Between $25k - $50k
- **3**: Between $50k - $75k
- **4**: Between $75k - $100k
- **5**: Between $100k - $200k
- **6**: Over $200k

Feature Insights



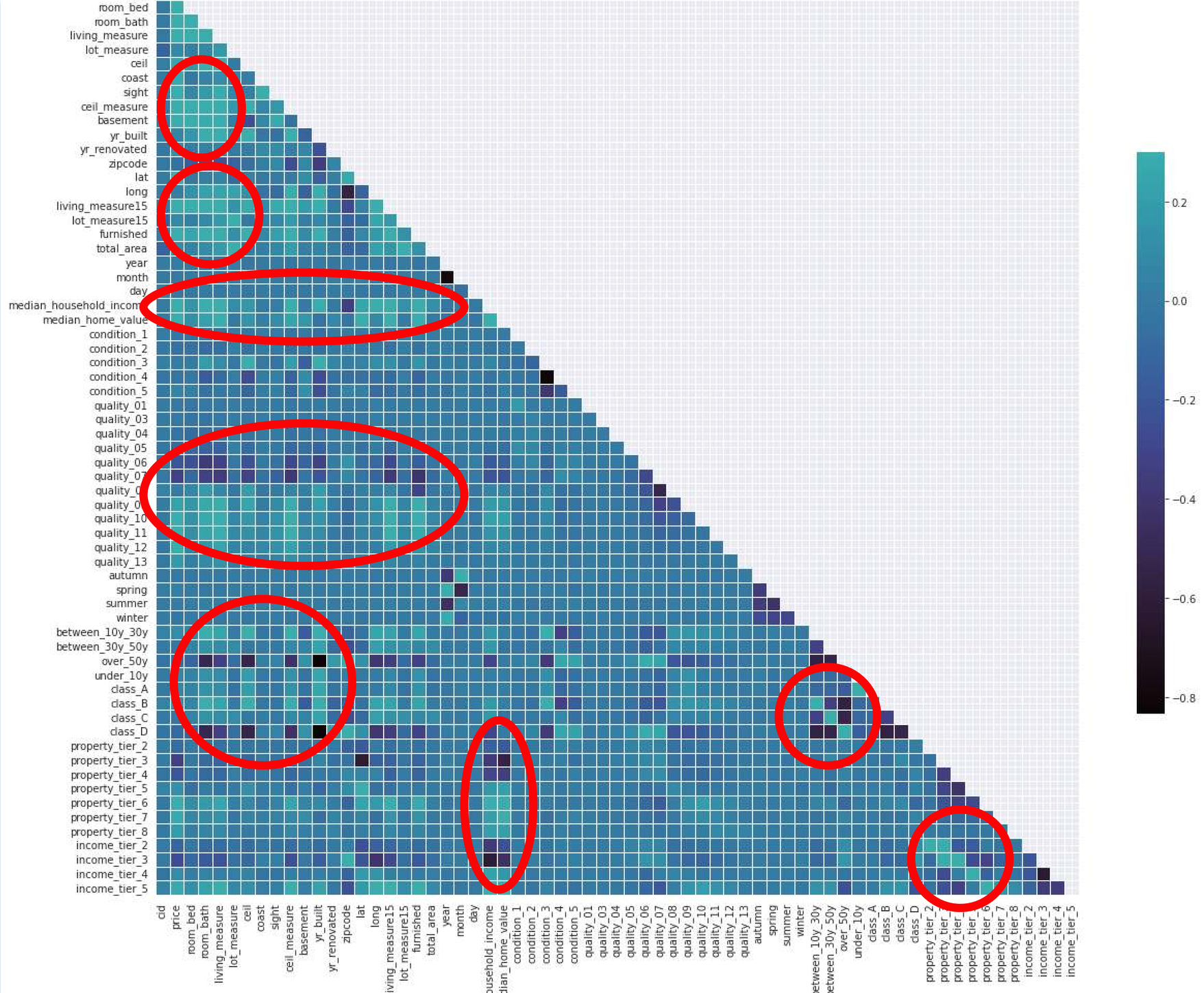




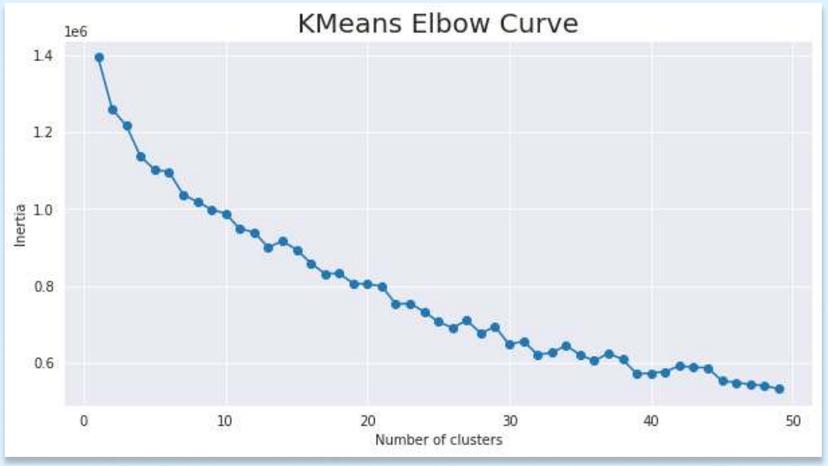


One-Hot Encoding

- yr_renovated
- condition
- quality
- sell_season
- property_age
- property_class
- property_tier
- income_tier

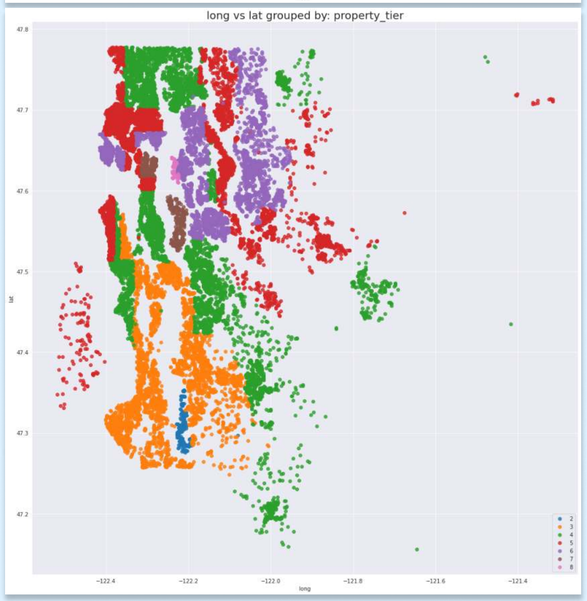


KMeans Cluster



Longitude & Latitude Clusters:

- KMeans Clusters: 4
- Property Class: A, B, C, D
- Property Tier: 2 - 8
- Income Tier: 2 - 5



Milestone 3

Prediction Models

Prediction Models

- **K-Means - 4 Clusters**
  - Get Baseline and find Multicollinearity
- **OLS Linear Regression**
  - Test other models and reduce features
- **Cross Validation with SciKit Linear Regression**
  - Compare to find best models
- **Decision Tree**
  - View Decision Trees for relevant models
- **Random Forest**
  - See if we can improve the models
- **XG Boost**
  - As the leading parallel tree gradient boosting regression model
- **ADA Boost**
  - As an alternative weights-based approach
- **Cat Boost**
  - As an alternative gradient boosting technique

Models

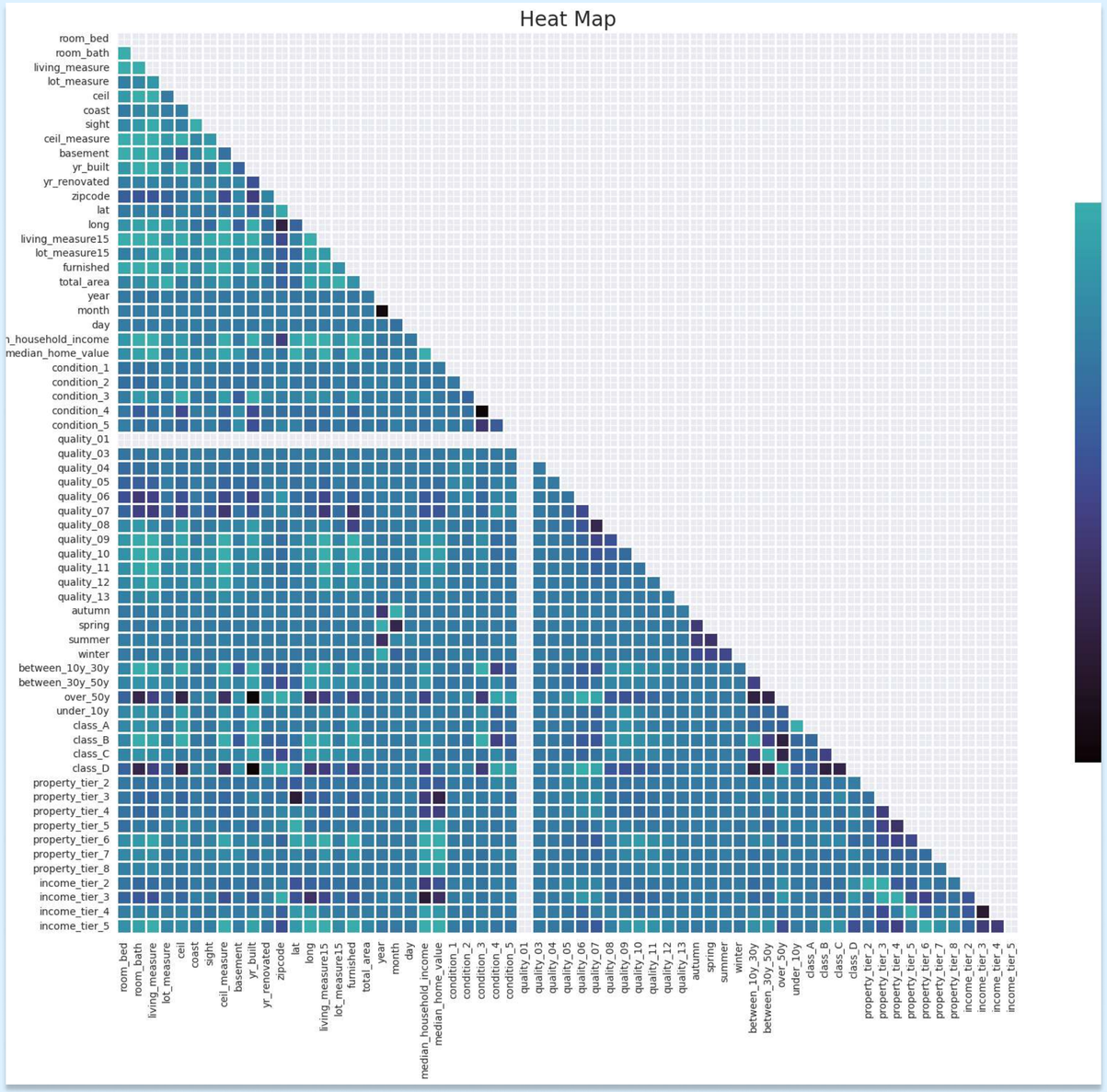
- **K-Means - 4 Clusters**
  - Model 0 (Baseline): Using all dummy data (one-hot)
  - Model 1: Auto-removing multicollinearity > 70%
- **OLS Linear Regression**
  - Model 2: Removing columns used for engineered features
  - Model 3: Removing features with p > 5%
  - Model 4: Removing Multicollinearity with VIF ~> 5
- **Cross Validation with SciKit Linear Regression**
  - Comparing the RSquare of all Models for best ones
- **Decision Tree**
  - Model 5: Based on Model 2 which uses all features
  - Model 6: Based on Model 4 which uses least features
- **Random Forest**
  - Model 7: Based on Model 2 which uses all features
  - Model 8: Based on Model 4 which uses least features
- **XG Boost**
  - Model 9: Using all dummy data
  - Model 10: Removed low Permutation Features
  - Model 11: Based on Model 4 which uses least features
- **ADA Boost**
  - Model 12: Using all dummy data
- **Cat Boost**
  - Model 13: Using all dummy data
  - Model 14: Based on Model 4 which uses least features

Models 0 & 1: K-Means

- **K-Means - 4 Clusters**

  - **Model 0 (Baseline)**:
    - Using all dummy data (one-hot)

  - **Model 1**: Auto-removing multicollinearity > 70%:
    - ceil_measure, class_A, class_B, class_C, class_D, condition_4, income_tier_5, living_measure, living_measure15, month, over_50y, quality_09, total_area



Models 2 - 4: OLS LR

- **OLS Linear Regression**
  - **Model 2**: Removing columns used for engineered features:
    - zipcode, yr_built, year, month, day, condition_1, condition_2, condition_3, condition_4, condition_5, median_household_income, median_home_value, under_10y, between_10y_30y, between_30y_50y, over_50y, lat, long
  - **Model 3**: Removing features with p > 5%
    - living_measure, lot_measure, ceil, ceil_measure, basement, living_measure15, furnished, total_area, quality_03, quality_10, income_tier_5
  - **Model 4**: Removing Multicollinearity with VIF ~> 5
    - quality_01, quality_07, quality_08, autumn, winter, spring, summer, class_A, class_B, class_C, property_tier_2, property_tier_3, property_tier_4, property_tier_5, property_tier_6, property_tier_7,property_tier_8, income_tier_3

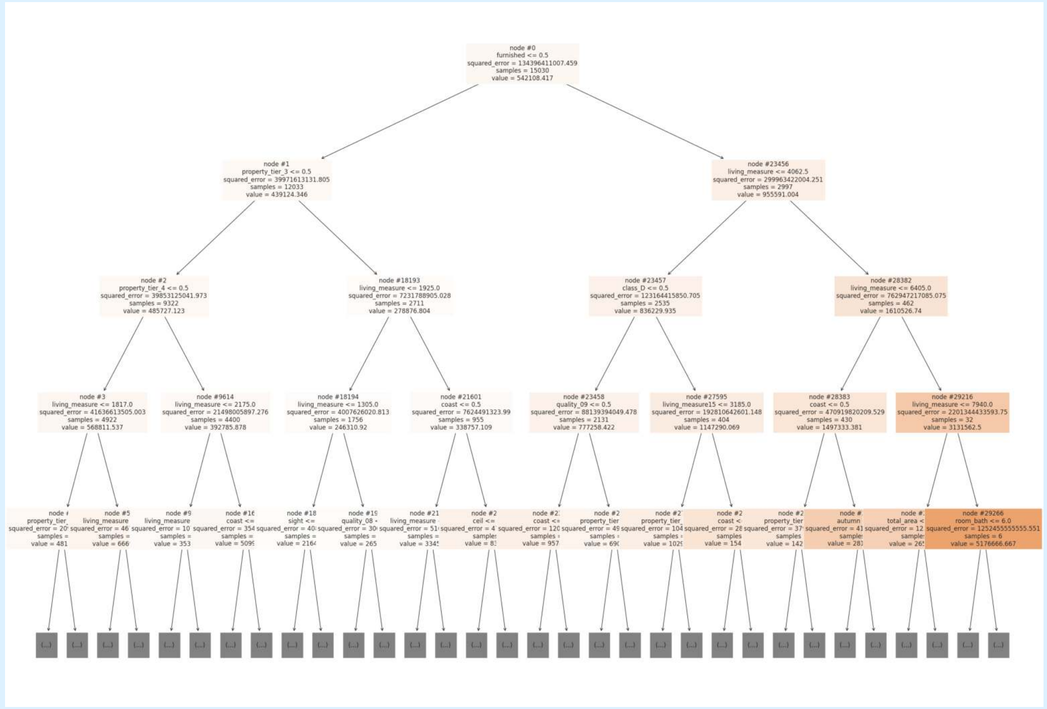
Cross-Validation SciKit LR

- **Cross Validation with SciKit Linear Regression**
  - Model 2 has the highest RSquare at 79% compared to the baseline model of complete over-fitting

Model 0: 4C KMeans (All Features)	
RSquared	0.815 (+/- 0.048)
Mean Squared Error	24948266426.227 (+/- 10295447264.802)
Model 1: 4C KMeans (Auto-Removed)	
RSquared	0.785 (+/- 0.057)
Mean Squared Error	29067952087.755 (+/- 13147224150.371)
Model 2: OLS Linear Regression (Manually Removed)	
RSquared	**0.790 (+/- 0.051)**
Mean Squared Error	28278547301.063 (+/- 11123438449.677)
Model 3: OLS Linear Regression (Removed P-Value > 5%)	
RSquared	0.757 (+/- 0.058)
Mean Squared Error	32841729282.006 (+/- 14009885730.499)
Model 4: OLS Linear Regression (Removed VIFs)	
RSquared	0.515 (+/- 0.062)
Mean Squared Error	64949908700.958 (+/- 15460609111.654)

Models 5 & 6: Decision Trees

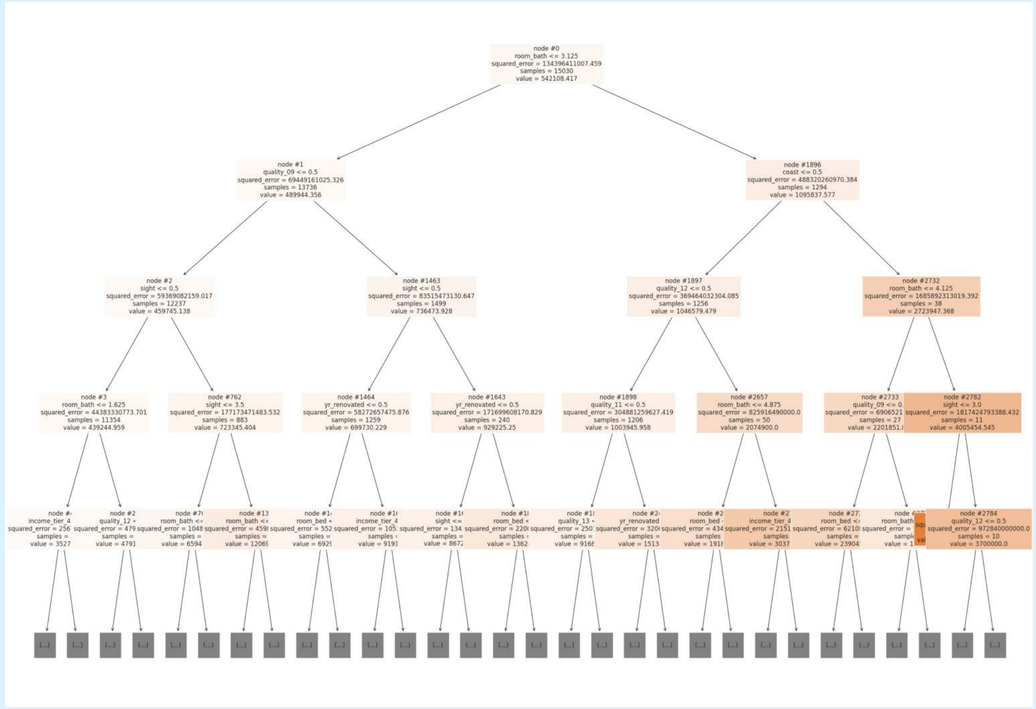
- **Decision Trees**
  - **Model 5**: Based on Model 2 which uses all features



Model 5 (OLS all features) - Feature Splits:

1.a) furnished
2.a) property_tier_3
2.b) living_measure
3.a) property_tier_4
3.b) living_measure
3.c) class_D
3.d) living_measure
4.a) living_measure
4.b) living_measure
4.c) living_measure
4.d) coast
4.e) quality_09
4.f) living_measure15
4.g) coast
4.h) living_measure

- **Decision Trees**
  - **Model 6**: Based on Model 4 which uses least features



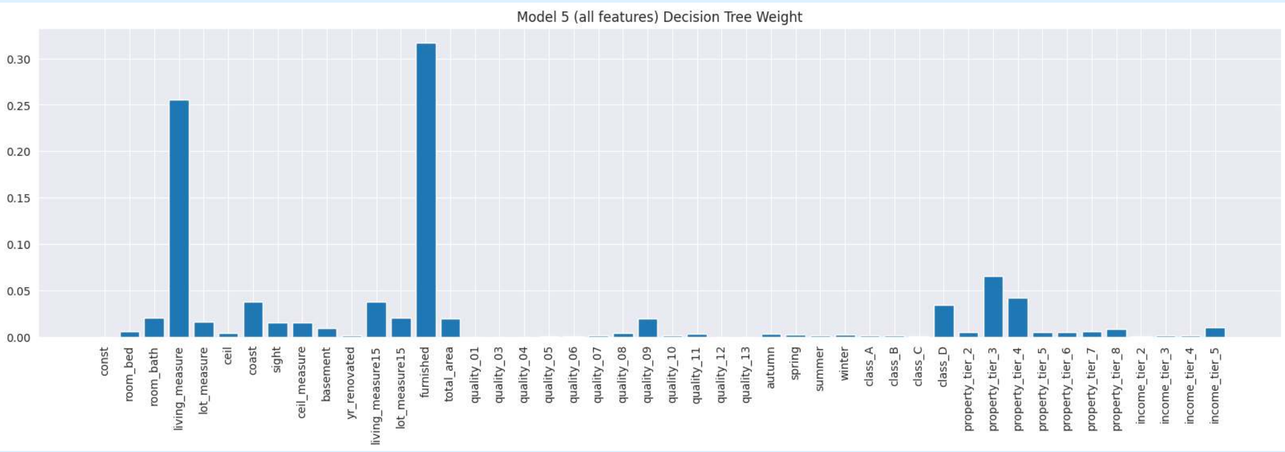
Model 6 (OLS least features) - Feature Splits:

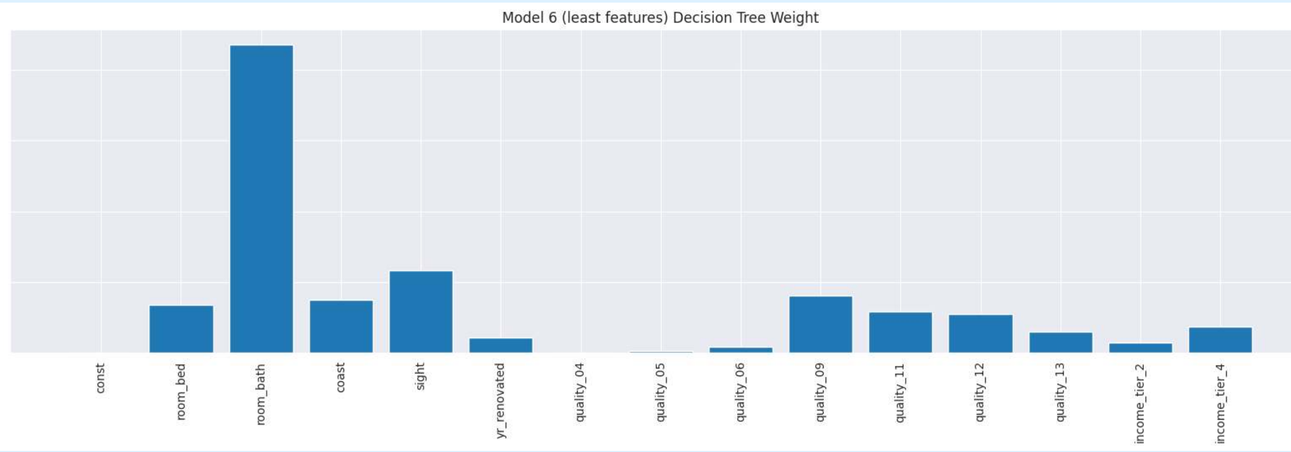
1.a) room_bath
2.a) quality_09
2.b) coast
3.a) sight
3.b) sight
3.c) quality_08
3.d) room_bath
4.a) room_bath
4.b) sight
4.c) yr_renovated
4.d) yr_renovated
4.e) quality_12
4.f) room_bath
4.g) quality_09
4.h) sight

Models 5 & 6: Decision Trees

- **Decision Trees**
  - **Model 5**: Based on Model 2 which uses all features

- **Decision Trees**
  - **Model 6**: Based on Model 4 which uses least features





From Model 5 (all features):

- Most important features are furnished, property_tier3/4 and living_measure
- Half of the tree is defined by living_measure

From Model 6 (least features):

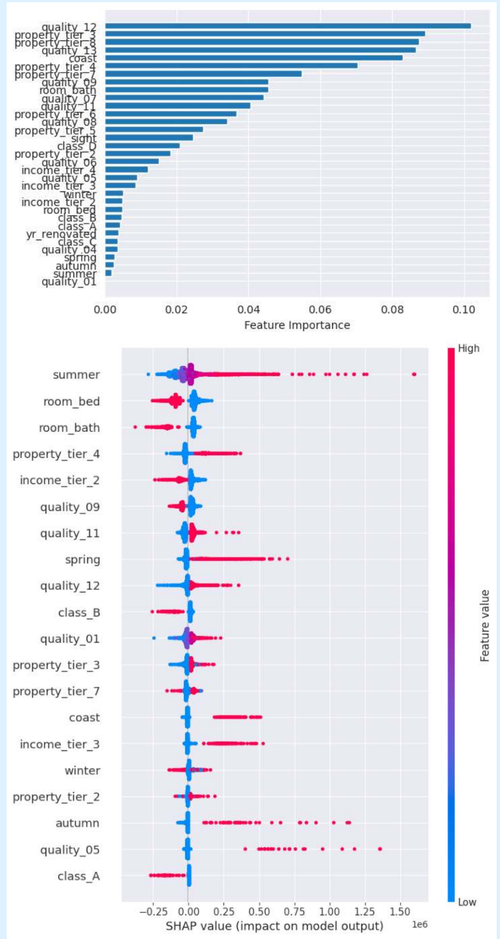
- Most important features are room_bath, quality09/08, coast and sight

Models 7 & 8: Random Forest

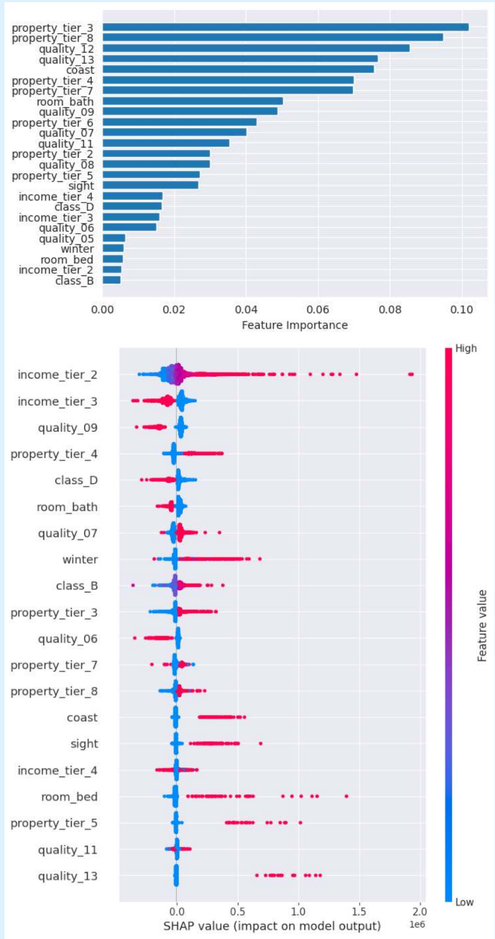
- **Random Forest**
  - **Model 7**: Based on Model 2 which uses all features
  - **Model 8**: Based on Model 4 which uses least features

Models 9, 10 & 11: XG Boost

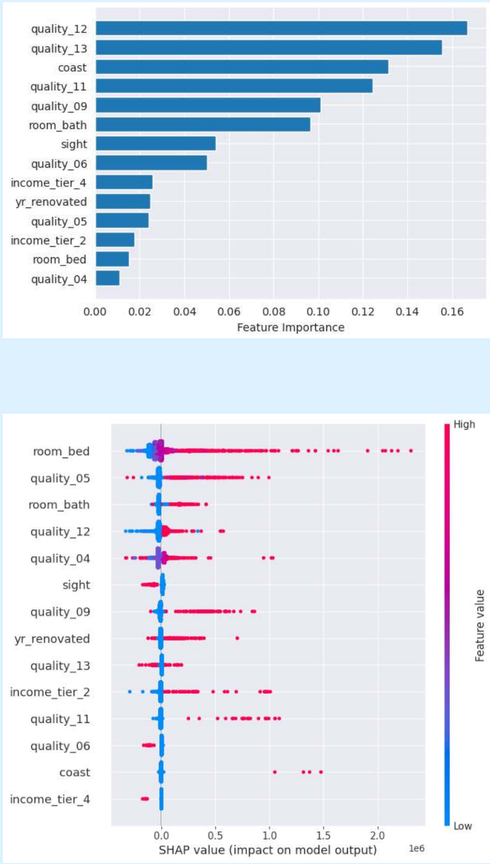
- **Model 9**: Using all dummy data
- **Top 10**: quality_12, property_tier_3, property_tier_8, quality, 13, coast, property_tier_4, property_tier_7, quality_09, room_bath & quality_07



- **Model 10**: w/o Low Permutation
- **Top 10**: property_tier_3, property_tier_8, quality_12, quality_13, coast, property_tier_4, property_tier_7, room_bath, quality_9 & property_tier_6



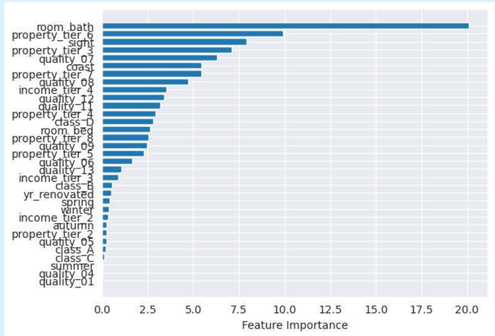
- **Model 11**: Model 4 with least features
- **Top 10**: quality_12, quality_13, coast, quality_11, quality_09, room_bath, sight, quality_06, income_tier_4 & yr_renovated

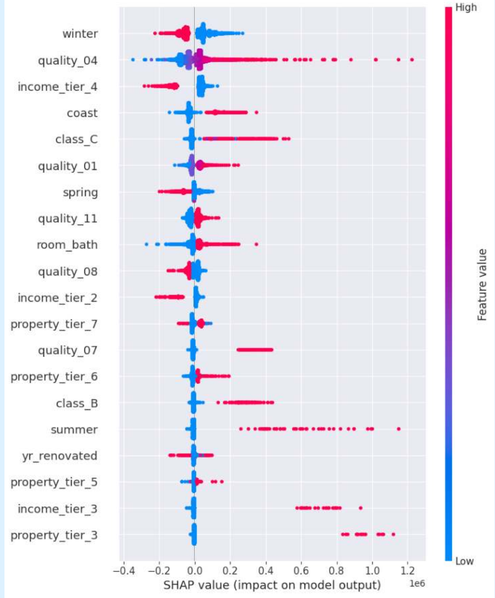


Models 12, 13 & 14: ADA & Cat

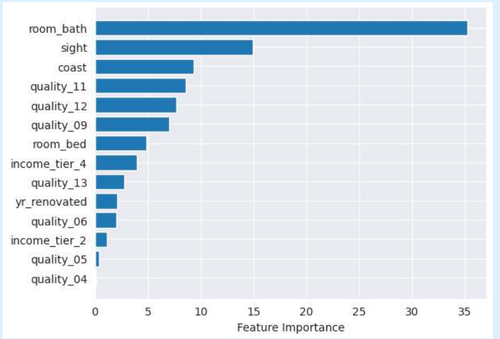
- **Model 12**: ADA using all dummy data
- Poor Performance

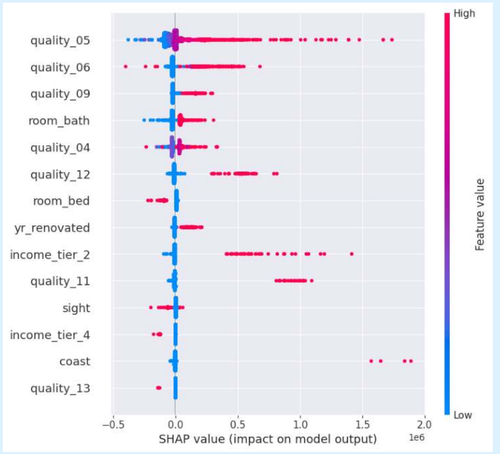
- **Model 13**: CAT using all dummy dat
- **Top 10**: room_bath, property_tier_6, sight, property_tier_3, quality_07, coast, property_tier_7, quality_08, income_tier_2 & quality_12





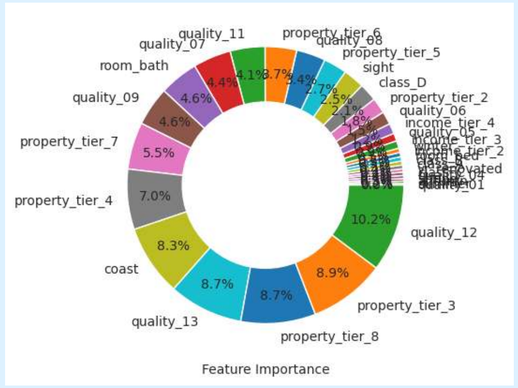
- **Model 14**: CAT Model 4 with least features
- **Top 10**: room_bath, sight, coast, quality_11, quality_10, quality_09, room_bed, income_tier_4, quality_13 & yr_renovated



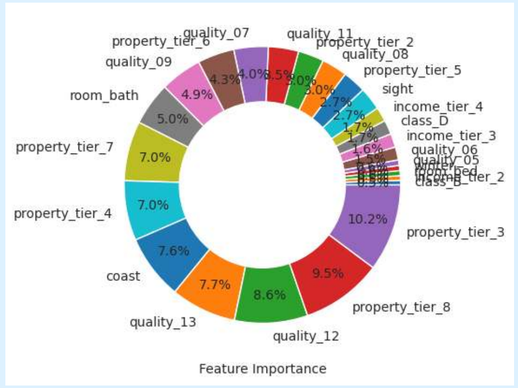


Feature Importance

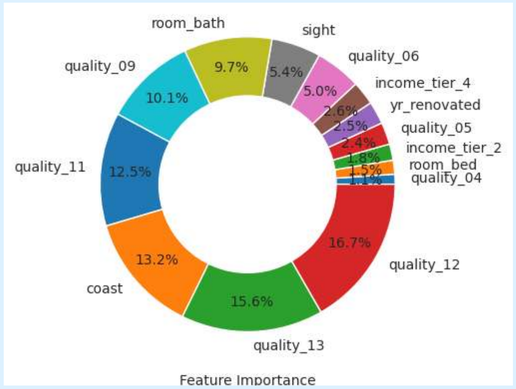
- **Model 9**: Using all dummy data



- **Model 10**: w/o Low Permutation

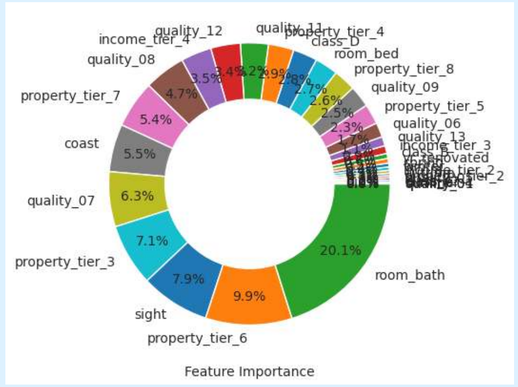


- **Model 11**: Model 4 with least features

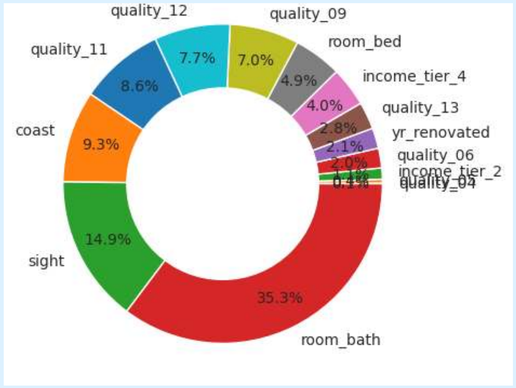


- **Model 12**: ADA using all dummy data

- **Model 13**: CAT using all dummy dat



- **Model 14**: CAT Model 4 with least features



MAE, MAPE, RMSE, R^2

MODELS	MAE	MAPE	RMSE	RSquare
Model 0 Train: 4C KMeans (All Features)	542.11K	100.00%	428.28B	-2.187
Model 0 Test: 4C KMeans (All Features)	534.87K	100.00%	416.67B	-2.191
Model 1 Train: 4C KMeans (Auto-Removed)	542.11K	100.00%	428.28B	-2.187
Model 1 Test: 4C KMeans (Auto-Removed)	534.87K	100.00%	416.67B	-2.191
Model 2 Train: OLS Linear Regression (Manually Removed)	101.99K	19.95%	27.75B	79.35%
Model 2 Test: OLS Linear Regression (Manually Removed)	101.28K	20.12%	27.5B	78.94%
Model 3 Train: OLS Linear Regression (Removed P-Value > 5%)	108.93K	21.01%	32.35B	0.759
Model 3 Test: OLS Linear Regression (Removed P-Value > 5%)	107.57K	21.53%	30.65B	0.765
Model 4 Train: OLS Linear Regression (Removed VIFs)	170.03K	35.17%	64.53B	0.520
Model 4 Test: OLS Linear Regression (Removed VIFs)	166.36K	35.70%	59.87B	0.542
Model 5 Train: Decision Tree (from Model 2)	60.61	0.02%	11.87M	1.000
Model 5 Test: Decision Tree (from Model 2)	120.22K	22.41%	49.22B	0.623
Model 6 Train: Decision Tree (from Model 4)	134.43K	29.82%	37.83B	0.718
Model 6 Test: Decision Tree (from Model 4)	166.53K	34.48%	74.15B	0.432
Model 7 Train: Random Forest (from Model 2)	118.91K	23.47%	38.57B	71.30%
Model 7 Test: Random Forest (from Model 2)	120.49K	24.14%	40.89B	68.69%
Model 8 Train: Random Forest (from Model 4)	174.76K	38.01%	66.56B	50.47%
Model 8 Test: Random Forest (from Model 4)	172.07K	38.38%	66.13B	49.36%
Model 9 Train: XGBoost with all features	78.55K	16.26%	13.16B	90.21%
Model 9 Test: XGBoost with all features	97.52K	19.22%	29.2B	77.64%
Model 10 Train: XGBoost without low Permutation cols	79.94K	16.59%	13.96B	89.61%
Model 10 Test: XGBoost without low Permutation cols	99.23K	19.26%	31.94B	75.54%
Model 11 Train: XGBoost with data from Model04	143.56K	31.10%	41.53B	69.10%
Model 11 Test: XGBoost with data from Model04	159.77K	33.71%	62.86B	51.86%
Model 12 Train: ADABoost with all features	159.35K	32.09%	68.45B	49.07%
Model 12 Test: ADABoost with all features	157.56K	32.14%	70.66B	45.89%
Model 13 Train: CatBoost with all features	88.14K	17.78%	17.43B	**87.03%**
Model 13 Test: CatBoost with all features	96.89K	19.24%	27.52B	**78.93%**
Model 14 Train: CatBoost with data from Model04	157.06K	33.09%	51.66B	61.57%
Model 14 Test: CatBoost with data from Model04	157.86K	33.81%	56.24B	56.93%

Model Comparison

- **The selected model will be the Model 13 CatBoost with all features**, because it had the highest RSquared with the smallest difference between Train and Test, and the resulting features are more explainable.

- We believe we could further improve the model if we remove extreme price level properties.

- Model 5 was the Decision Tree from Model 2 (OLS LR with all data) with 99.99% RSquare was found to be very overfitting.

- The Random Forests did not improve as expected and were actually less than the OLS Linear Regressions, and had a larger difference between Train and Test.

- XGBoost provided good results but was a bit overfitting since difference between Train and Test is large, and resulting features were mostly degrees of property tier and quality.

Actions

Corporate **investments** and REIT portfolio **growth**

- Long term investment via buy and rent newer & larger properties near the coast
- Volume purchase of Class D properties, Tier 4-5 in areas of Income Tier 3

- **Government** housing & **NGO** projects
  - Highest volume in property Class C, Tier 3 in areas of Income Tier 4-5

- **Individual** home purchase
  - On avg. best during winter months in areas of Income Tier 3-4 for properties Class B-C, Tier 5-8
  - For flipping go down a bit in Class and Tiers

Insights

Taking into account the results from the **Decision Trees**, **XGBoost** & **CatBoost** the most important features are:

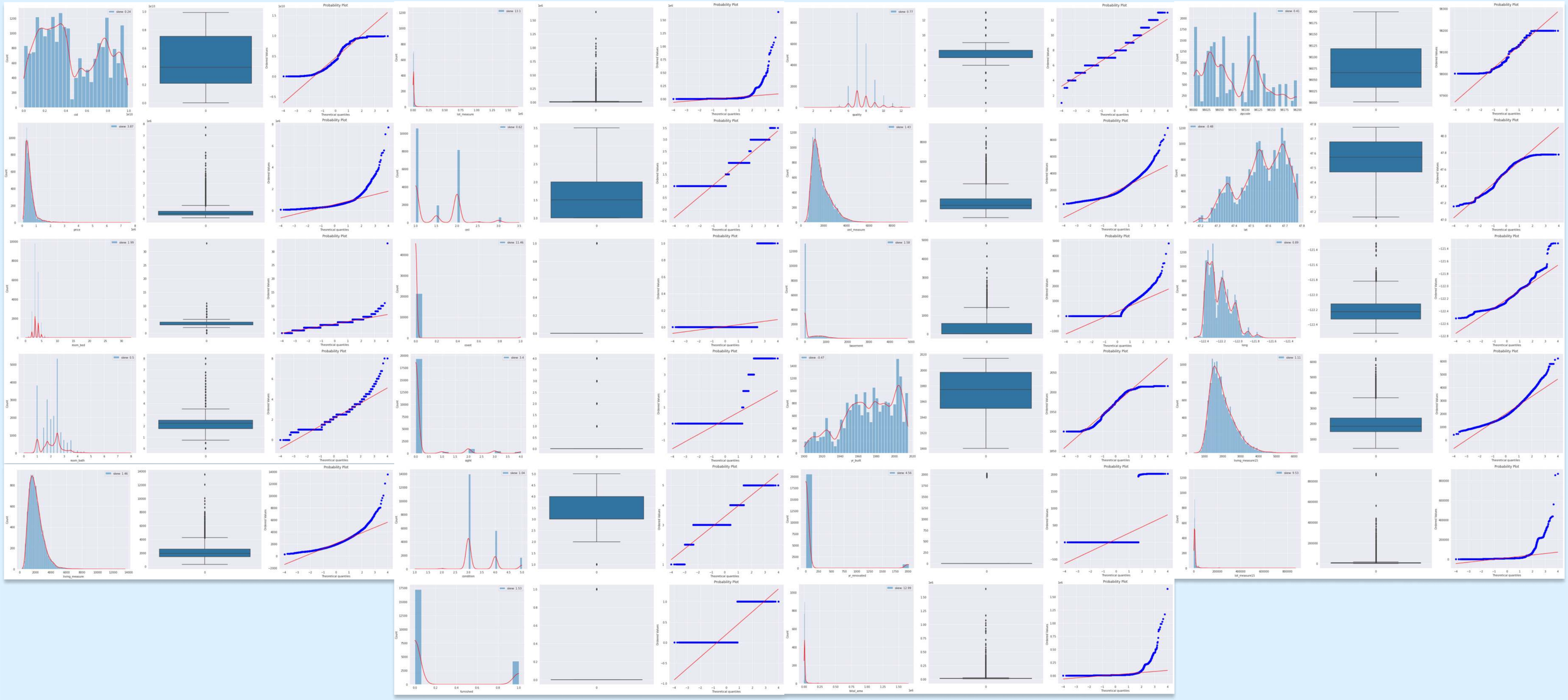
- room_bath, property tier, quality, coast & sight.

This tells us that the price is most closely related to property number of restrooms, overall quality, near a coast and demand.

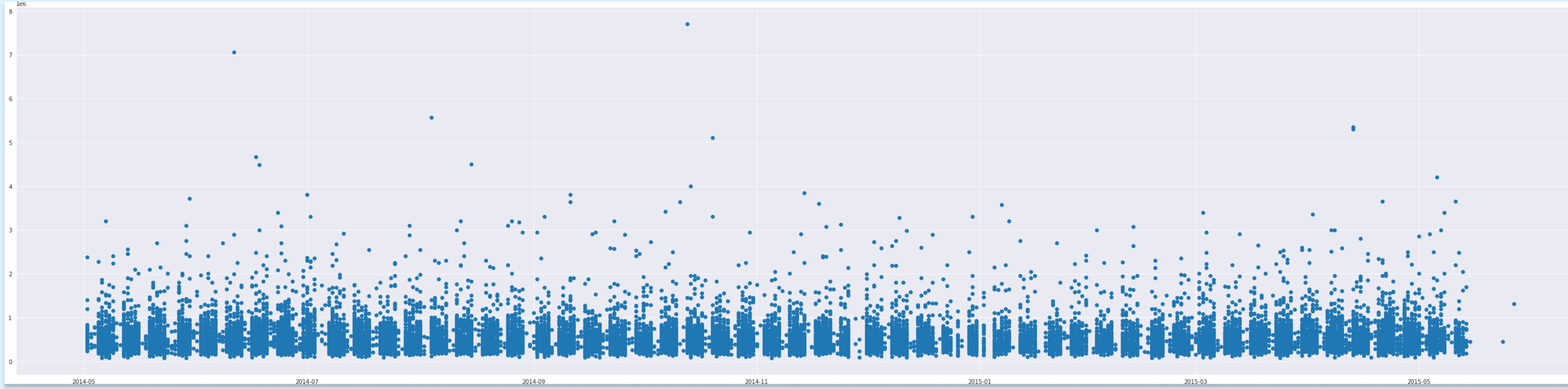
The highest prediction percentage was of **87.03%** with the **Model 13 CatBoost with all features,** which could be improved by further capping the outliers to have a more normally distributed dataset.

Addendum

Data Skewness



Prices by Date



Thank You!

by Hector Sanchez Johnson