Homework 10 Report

Chenning Xu

1 Results

Table 1: Fit on the training data

Model	N	R^2	Adj R^2	RMSE	AIC	BIC	K-fold 1	K-fold 2
1	8292	0.7914	0.7893	0.2181	-1639.053	-1063.163	0.2198	0.2195
2	3962	0.8256	0.8218	0.2052	-1224.056	-689.8726	0.2095	0.2088
3	8292	0.8158	0.8113	0.2064	-2441.884	-1058.344	0.2099	0.2093
4	8292	0.8294	0.8248	0.1989	-3031.395	-1486.324	0.2028	0.2024
5	8292	0.8201	0.8157	0.2040	-2631.496	-1226.887	0.2073	0.2072
6	8292	0.8335	0.8289	0.1966	-3228.010	-1661.871	0.2004	0.2001
7	8292	0.8351	0.8304	0.1957	-3294.334	-1679.034	0.1996	0.1992
8	8300	0.7946	0.7896	0.2179	-1541.301	-157.5708	0.2210	0.2208

Notes: *K-fold 1* is calculated by cv_kfold, $k(5)$ $reps(5)$. *K-fold 2* is calculated by cv_kfold, $k(10)$ $reps(5)$.

I would submit **model 6** as my best attempt. All the models are est sto'ed by model#.

2 Feature Engineering

All eight models use dummy variables such as *suburb*, *council_code*, *region_code*, and *seller_code*. I removed dummy variables with fewer than 20 positive instances, as well as those whose coefficients have a p-value above 0.1. For detailed dummies like *suburb* and *seller*, this results in the removal of many variables, whereas for *region_code*, none were removed.

Table 2: Summary Statistics of Continuous Variables

	count	min	max	mean	sd
Log Price	8300	10.18867	13.36452	11.51832	.4751715
Number of rooms	8300	1	7	3.151807	.8235539
Number of Bathrooms	8300	0	6	1.573253	.6751098
Number of carspots	8300	0	3	1.665663	.7744266
Land Size in Metres	8300	50	8140	539.6192	377.3132
Building Size in Metres	4149	0	6791	157.5097	147.9551
Year the house was built	4684	1850	2018	1963.113	36.06426

	count	min	max	mean	sd
Lattitude	8292	-38.18255	-37.39946	-37.80271	.0949225
Longtitude	8292	144.4238	145.4827	144.9983	.1229395
Number of properties that exist in the suburb.	8300	121	21650	7459.17	4488.203
Date of Sell	8300	20481	21260	20936.76	200.9497

For continuous variables, transformations were applied based on data characteristics:
- **Quadratic Transformation**: Applied to variables like *distance*, *rooms*, *bathroom*, *latitude*, and *longitude* when the scatter plot against the target variable showed curvature without extreme outliers.
- **Log Transformation**: Applied to variables with extreme values and strong right skewness, such as *landsize* and *buildingarea*.
- **No Transformation**: Variables like *date*, *yearbuilt*, *car*, and *prop_count* were left untransformed as they showed no obvious curvature or skew.

3 Model Specification

3.1 Models 1 to 4

Two variables, *yearbuilt* and *buildingarea*, have a significant amount of missing data. Including these two variables reduces the training sample by half. Therefore, I tested models both with and without these variables.

Model 1 includes dummy variables for *council_code*, *housing_type*, and *seller* based on the following rationale:
1. Housing prices vary across regions.
2. Realtors may specialize in high- or low-value properties, influencing the sale price.
3. Housing type impacts value, as townhouses tend to be less expensive than single-family houses.

Model 1 also includes the following continuous variables: *distance*, *distance*2, *rooms*, *rooms*2, *bathroom*, *bathroom*2, *car*, *log*(*landsize*), *latitude*, *latitude*2, *longitude*, *longitude*2, *prop_count*, and *date*. *Distance* and $numberofrooms$ are obviously key determinants of housing value. Parking availability (*car*) and land size tend to increase housing value. *Latitude* and *longitude* are included as proxies for city center proximity, which often correlates with higher value. Both variables are correlated with distance, but including them increases model performance nonetheless.

Additionally, interaction terms included in Model 1 are:
1. $rooms * bathroom$
2. $distance * log(landsize)$
3. $distance * rooms$
4. $distance * car$
5. $distance * type\_h$

These interactions capture relationships that affect housing value, such as the combined effect of $rooms * bathroom$ on the overall value. Distance interactions with *landsize*, *rooms*, *car*, and *housing_type* allow for

different impacts near the city versus further out. For instance, in suburban areas, homes with more land, rooms, and parking spots are common, while in the city, these features can make a house significantly more expensive. Although townhouses are usually less expensive, they can be costlier than single-family homes in rural areas like Red Hook.

Model 2 is the same as Model 1 but also includes *log*(*buildingarea*) and *yearbuilt*. Both variables are highly relevant to housing price but contain approximately 50% missing values, which reduces the available training data. This model has better fit measures than **model 1** except BIC, which shows that *log*(*buildingarea*) and *yearbuilt* are still necessary independent variables. This makes sense since a larger building area should add value to houses, and older houses tend to sell for less. In later models from **model 5** to **model 8**, missing values are replaced by averages.

Model 3 is the same as Model 1 but replaces *council_code* with *suburb* as the dummy variable. *Suburb* offers more localized information, but the trade-off is the removal of more dummies with low positive values. Due to fewer observations for each dummy, including variables with high missing values (e.g., *buildingarea* and *yearbuilt*) is less suitable. This model shows improvement over **model 1** except having a slightly lower BIC, indicating that using more detailed *suburb* may be better than using *council*.

Model 4 includes both *council* and *suburb* as dummy variables, which introduces potential multicollinearity. However, this model performs the best among the first four models, prioritizing predictive accuracy over interpretability. Since the goal is prediction rather than explanation, the model is preferred despite the multicollinearity risk. This model significantly outperforms **model 1** in all measures, suggesting that the improved accuracy outweighs the risk of multicollinearity.

3.2 Models 5 to 8

In these models, I included *log*(*buildingarea*) and *yearbuilt*, but filled their missing values using mean values to allow for a larger training sample.

Model 5 is the same as **model 3**. It outperforms **model 3** in all measures, indicating that including the two variables with missing values increases model performance.

Model 6 is the same as **model 4** except that it now includes the two variables with missing values. It outperforms **model 4** as expected.

Model 7 is the same as **model 6** except that it includes dummy variable $region_code$. This is a risky move since $region_code$, $council_code$, and *suburb* have multicollinearity, but surprisingly this model still outperforms **model 6** in all measures. However, the improvement is marginal, so it may not be worth the risk to include $region_code$.

Model 8 is a streamlined model. It is based on **model 5** but removes $prop_count$, *latitude*, and *longitude*. *Prop_count* does not have a highly significant coefficient in previous models, and both *latitude* and *longitude* are correlated with distance. This model does not perform well.

4 Model Selection

I believe **model 6** is the best candidate. Excluding **model 7**, it has the highest R^2, lowest $RMSE$, lowerest AIC and BIC and lowest MSE from K-fold cross-validations. **Model 7** sligtly outperform **model 6** but I think it is not worthwhile to risk more multi-colinearity.