Ames Housing Sale Price Prediction Model

Ling Chong Gold

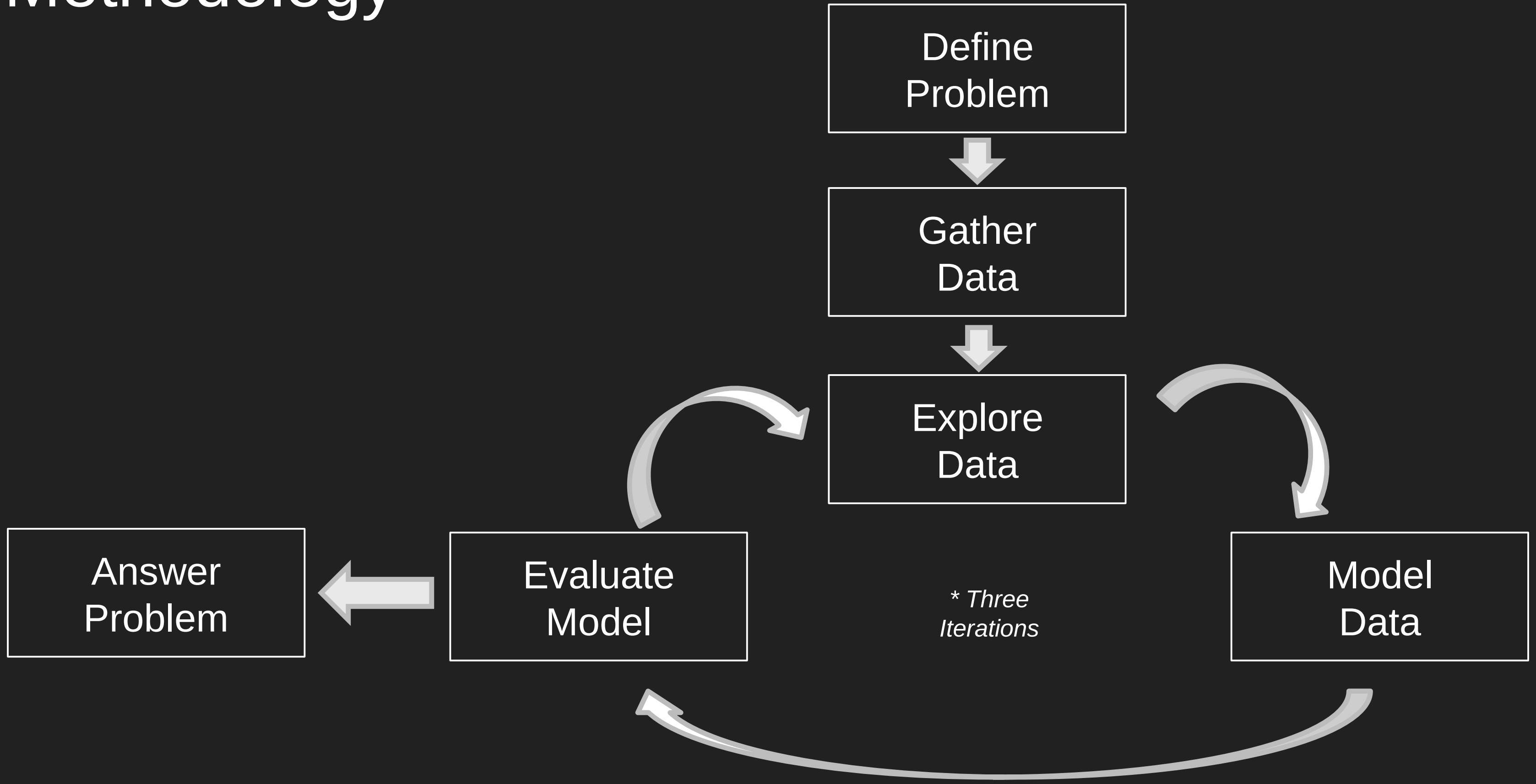
Agenda

- Background
- Methodology
- Problem Statement
- Gather and Data Cleaning
- Exploring Data
- Model Data
- Second Iteration
- Third Iteration
- Conclusion

Background

- 2 datasets of Aimes Iowa Housing Dataset was Provided
- Test dataset consists of 80 columns and 879 rows
- Create a model for price prediction
- Refine and improve the model
- Score is calculated based on the Root Mean Square Error after submission to Kaggle

Methodology



Define Problem

Gather Data

Explore Data

Model Data

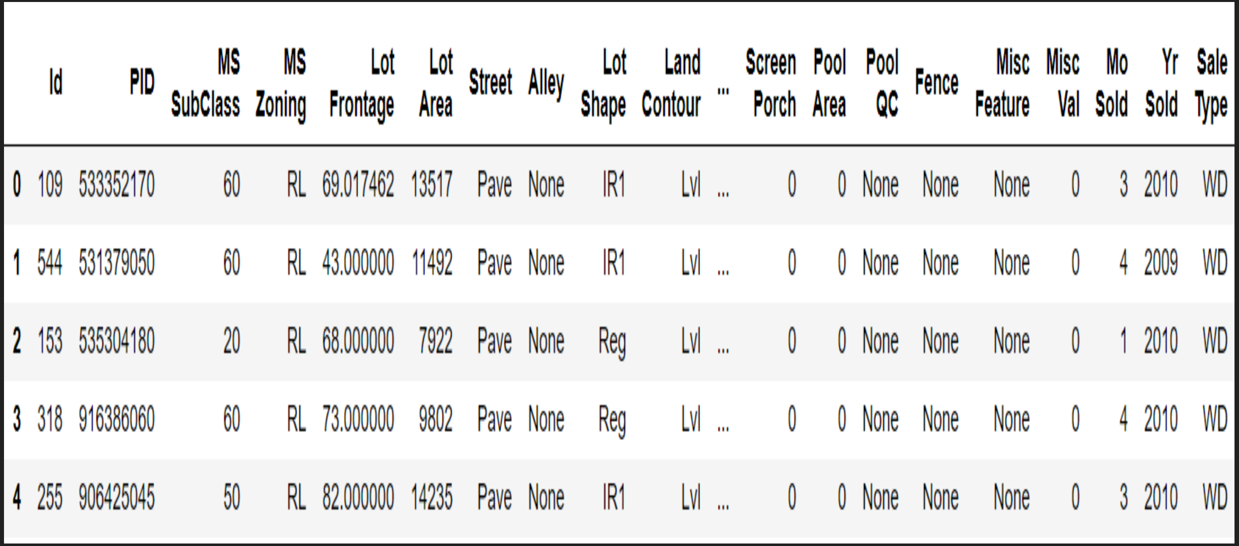
Evaluate Model

Answer Problem

** Three Iterations*

Problem Statement

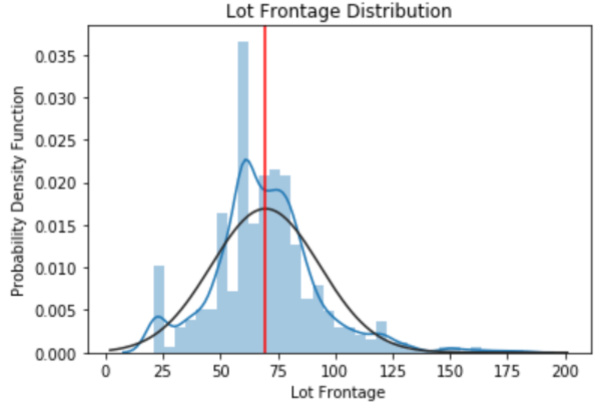
- From the Ames Housing dataset, create a model to predict the sale price and perform improvements to the model after it is created



	Id	PID	MS SubClass	MS Zoning	Lot Frontage	Lot Area	Street	Alley	Lot Shape	Land Contour	...	Screen Porch	Pool Area	Pool QC	Fence	Misc Feature	Misc Val	Mo Sold	Yr Sold	Sale Type
0	109	533352170	60	RL	69.017462	13517	Pave	None	IR1	Lvl	...	0	0	None	None	None	0	3	2010	WD
1	544	531379050	60	RL	43.000000	11492	Pave	None	IR1	Lvl	...	0	0	None	None	None	0	4	2009	WD
2	153	535304180	20	RL	68.000000	7922	Pave	None	Reg	Lvl	...	0	0	None	None	None	0	1	2010	WD
3	318	916386060	60	RL	73.000000	9802	Pave	None	Reg	Lvl	...	0	0	None	None	None	0	4	2010	WD
4	255	906425045	50	RL	82.000000	14235	Pave	None	IR1	Lvl	...	0	0	None	None	None	0	3	2010	WD

Gathering and Clean Data

- Data was provided and there were quite a number of null values
- 3 Row with Null Values Exclusive to Train dataset was dropped
- Most of the null values are due to Python recognizing NA as null
  (*They are filled with 'None' or 0 dependent on the columns data type)*
- Lot Frontage has a total of 490 null values
  (They are filled with the mean)



Exploring Data and Feature Engineering

- A column for Total Finished Basement Square Feet was created

  (Basement Finish Square Feet 1 + Basement Finish Square Feet 2)

- Garage Cars Column was dropped

  (Details can be inferred from Garage Area)

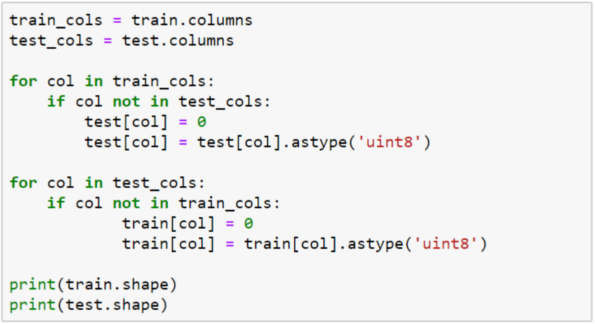
- A column for Age When Sold was created

  (Year Sold – Year Built)

- Ordinal Encoding was performed for Columns Depicting Quality

  (Central Air, Electrical, Functional)

- One Hot Encoding was performed for columns with discrete object values

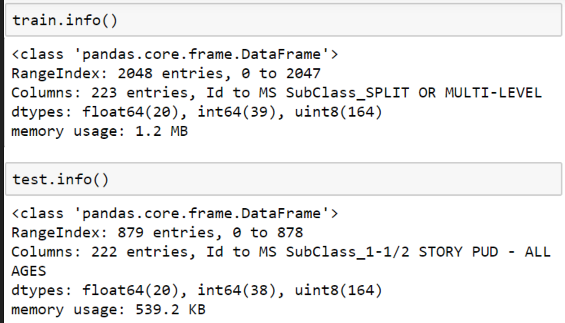


```
train_cols = train.columns
test_cols = test.columns

for col in train_cols:
    if col not in test_cols:
        test[col] = 0
        test[col] = test[col].astype('uint8')

for col in test_cols:
    if col not in train_cols:
            train[col] = 0
            train[col] = train[col].astype('uint8')

print(train.shape)
print(test.shape)
```



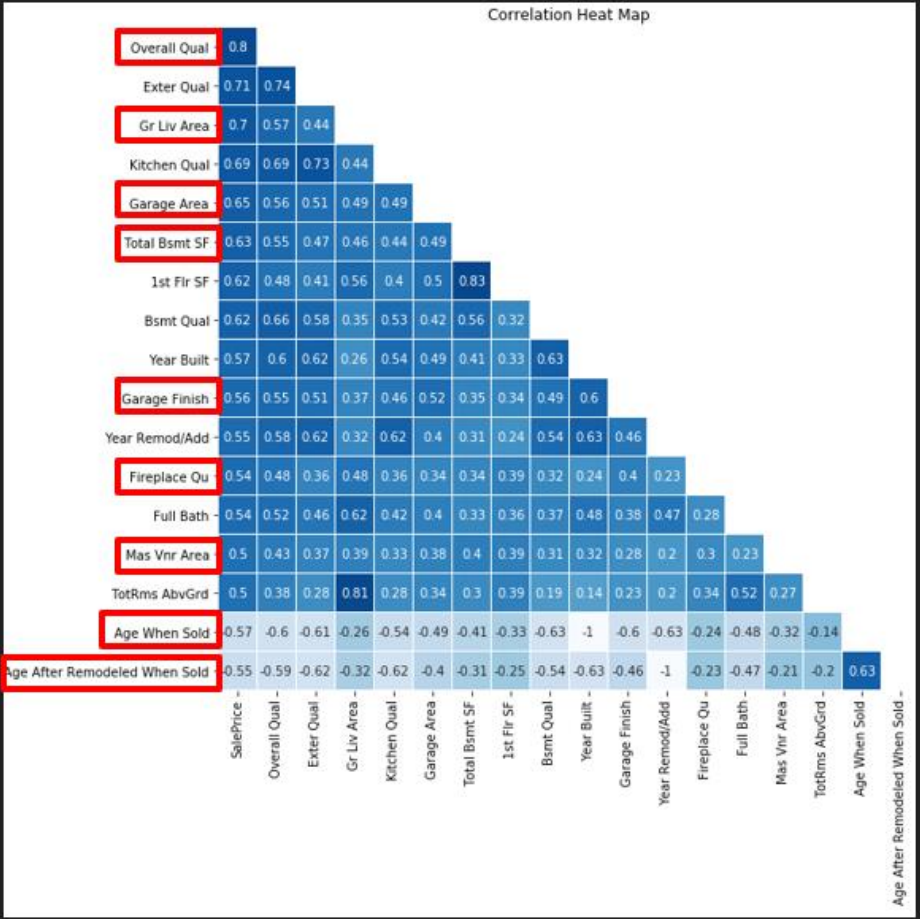
```
train.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2048 entries, 0 to 2047
Columns: 223 entries, Id to MS SubClass_SPLIT OR MULTI-LEVEL
dtypes: float64(20), int64(39), uint8(164)
memory usage: 1.2 MB
```

```
test.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 879 entries, 0 to 878
Columns: 222 entries, Id to MS SubClass_1-1/2 STORY PUD - ALL
AGES
dtypes: float64(20), int64(38), uint8(164)
memory usage: 539.2 KB
```

Model Data – 1st Iteration



Correlation Heat Map

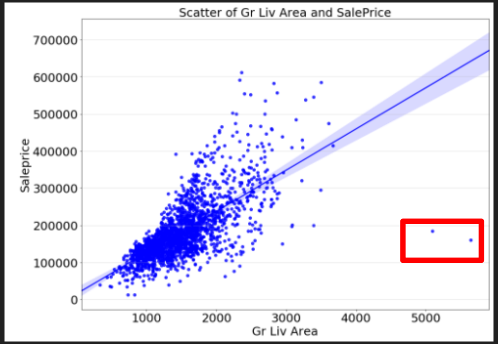
	SalePrice	Overall Qual	Exter Qual	Gr Liv Area	Kitchen Qual	Garage Area	Total Bsmt SF	1st Flr SF	Bsmt Qual	Year Built	Garage Finish	Year Remod/Add	Fireplace Qu	Full Bath	Mas Vnr Area	TotRms AbvGrd	Age When Sold	Age After Remodeled When Sold
Overall Qual	0.8																	
Exter Qual	0.71	0.74																
Gr Liv Area	0.7	0.57	0.44															
Kitchen Qual	0.69	0.69	0.73	0.44														
Garage Area	0.65	0.56	0.51	0.49	0.49													
Total Bsmt SF	0.63	0.55	0.47	0.46	0.44	0.49												
1st Flr SF	0.62	0.48	0.41	0.56	0.4	0.5	0.83											
Bsmt Qual	0.62	0.66	0.58	0.35	0.53	0.42	0.56	0.32										
Year Built	0.57	0.6	0.62	0.26	0.54	0.49	0.41	0.33	0.63									
Garage Finish	0.56	0.55	0.51	0.37	0.46	0.52	0.35	0.34	0.49	0.6								
Year Remod/Add	0.55	0.58	0.62	0.32	0.62	0.4	0.31	0.24	0.54	0.63	0.46							
Fireplace Qu	0.54	0.48	0.36	0.48	0.36	0.34	0.34	0.39	0.32	0.24	0.4	0.23						
Full Bath	0.54	0.52	0.46	0.62	0.42	0.4	0.33	0.36	0.37	0.48	0.38	0.47	0.28					
Mas Vnr Area	0.5	0.43	0.37	0.39	0.33	0.38	0.4	0.39	0.31	0.32	0.28	0.2	0.3	0.23				
TotRms AbvGrd	0.5	0.38	0.28	0.81	0.28	0.34	0.3	0.39	0.19	0.14	0.23	0.2	0.34	0.52	0.27			
Age When Sold	-0.57	-0.6	-0.61	-0.26	-0.54	-0.49	-0.41	-0.33	-0.63	-1	-0.6	-0.63	-0.24	-0.48	-0.32	-0.14		
Age After Remodeled When Sold	-0.55	-0.59	-0.62	-0.32	-0.62	-0.4	-0.31	-0.25	-0.54	-0.63	-0.46	-1	-0.23	-0.47	-0.21	-0.2	0.63	

- Features higher than 0.5 in correlation with sale price is selected and heatmap plotted

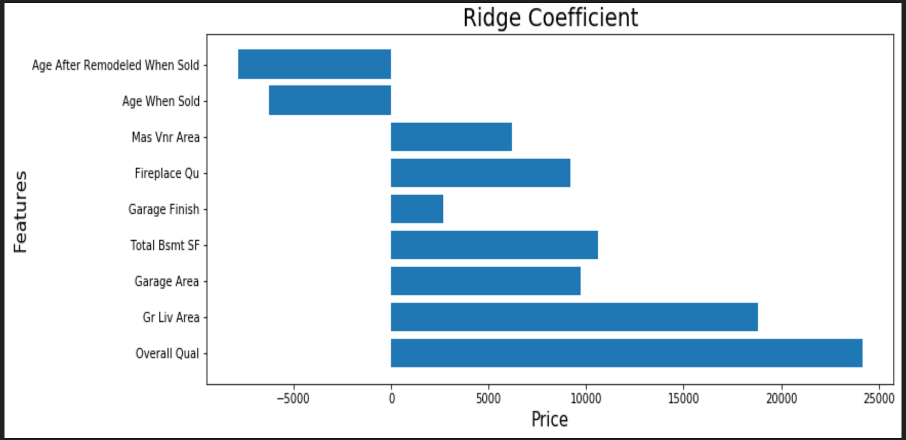
- 9 features were selected base on
  - Their correlation with sale price
  - They are not correlated with each other
  - If 2 features are correlated with each other the one with a higher correlation with sale price is selected

- Interaction terms related to selected will be created during the next iteration

Model Data – 1st Iteration

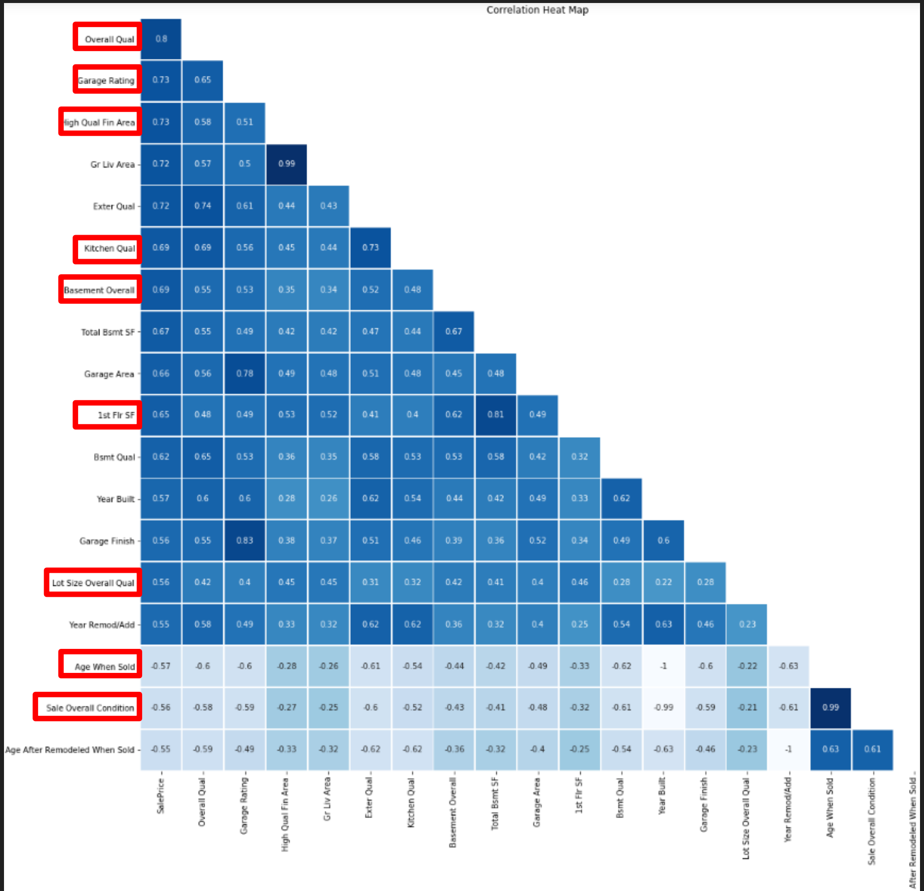


- Outliers was discovered based on scatterplot
- These outliers will be removed during the next iteration



- 1 unit increase in Age After Remodeled When Sold is equals to around 8000 decrease in Sale Price
- 1 unit increase in Age When Sold is equals to around 5000 decrease in Sale Price

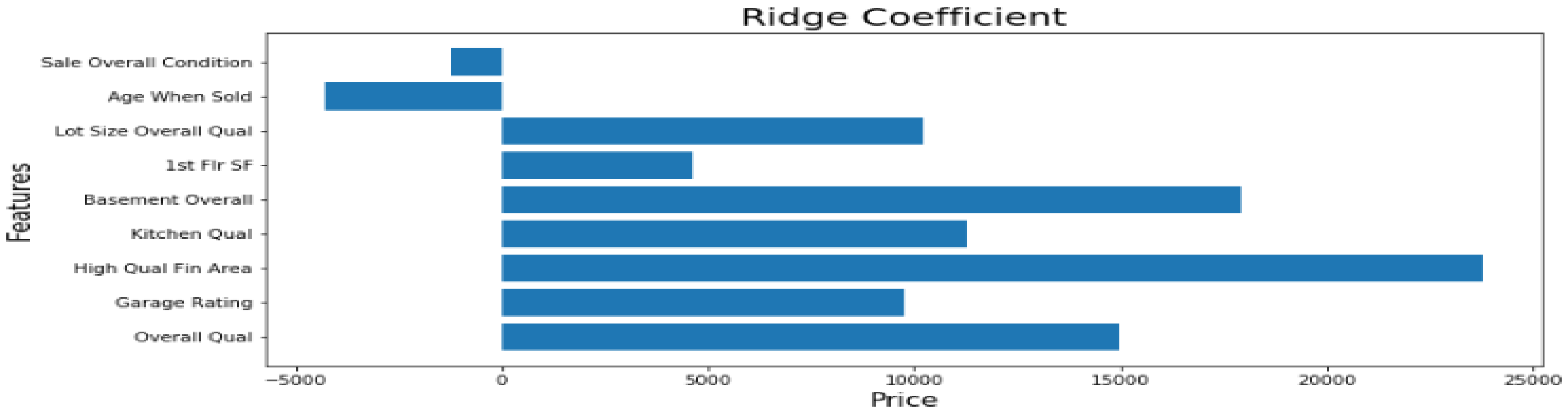
Model Data – 2nd Iteration



- High Quality Finish Area was created
- Lot Size Overall Quality was created
- Garage Overall was created
- Fireplace Overall was created
- Sale Overall Condition created

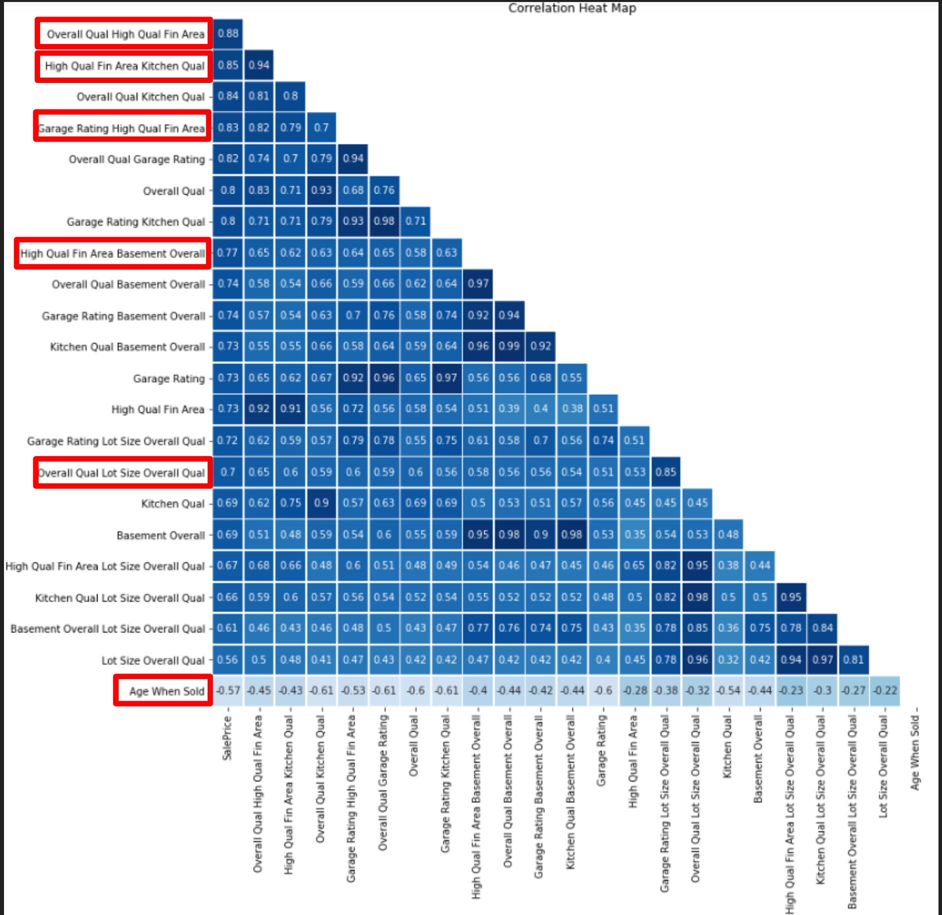
- Heatmap was plotted again
- Top 9 features was selected using the same methodology during the first iteration

Model Data – 2nd Iteration



- 1 unit increase in Sale Overall Condition is equals to around 1000 decrease in Sale Price
- 1 unit increase in Age When Sold is equals to around 4000 decrease in Sale Price

Model Data – 3rd Iteration

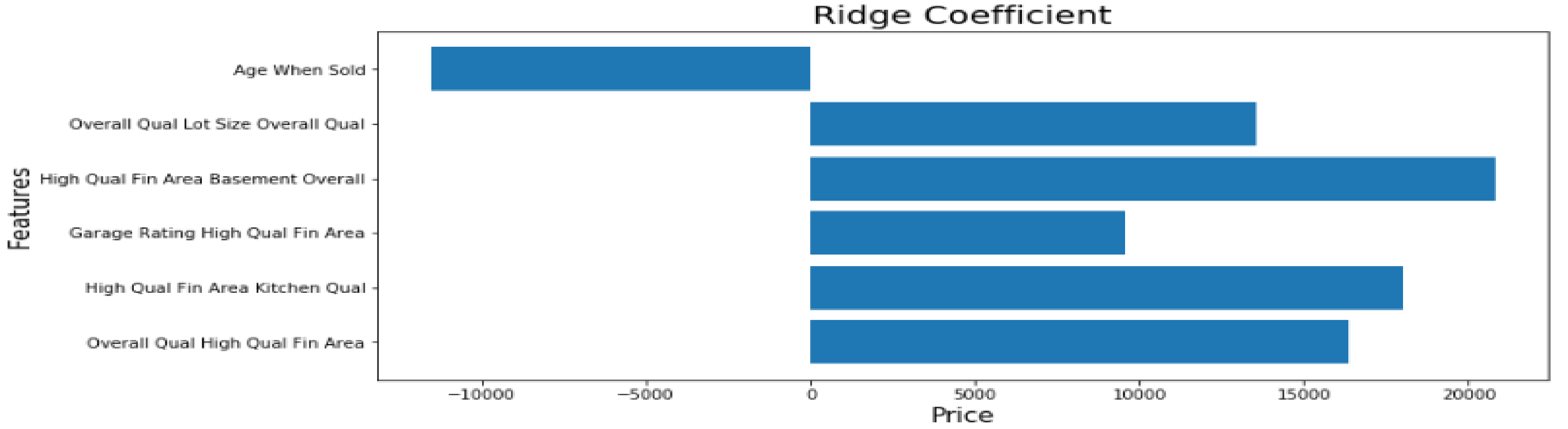


- Top 9 features was selected and Polynomial Feature was performed on them

- Another heatmap was plotted for visualization and select the best predictors based on the same methodology as per previous iterations

- 6 features was selected

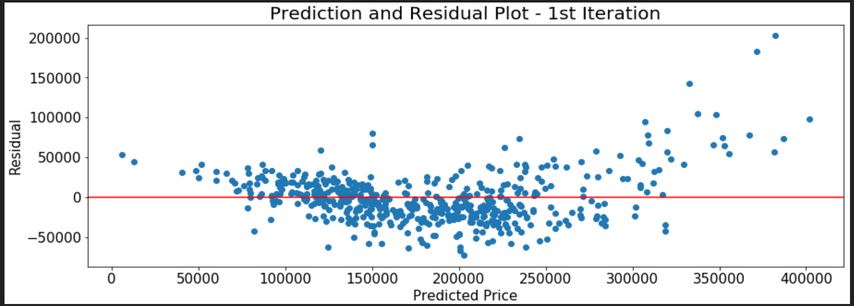
Model Data – 3rd Iteration

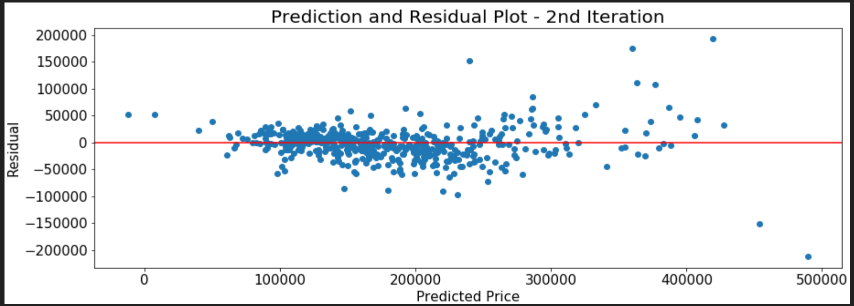


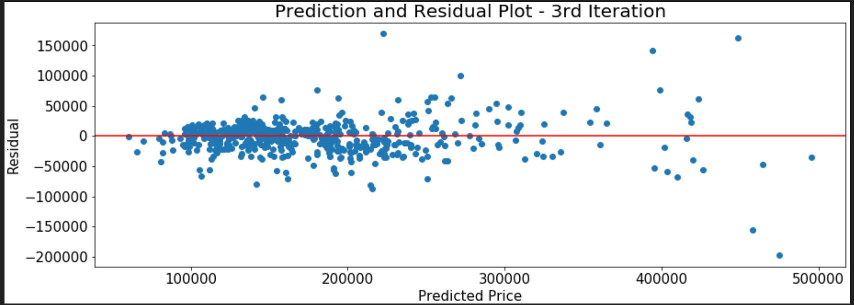
- 1 unit increase in Age When Sold is equals to around 10000 decrease in Sale Price

- 1 unit increase in the interaction between Overall Qual and Lot Size Overall Qual is equals to around 14000 increase in Sale Price

Prediction and Residual Scatter – (Train Test Split)





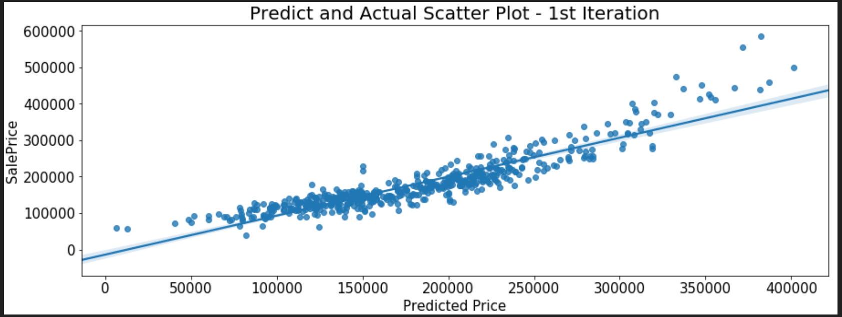


- You would want to see that the points are heteroscedastic signifying that your error rate is consistent

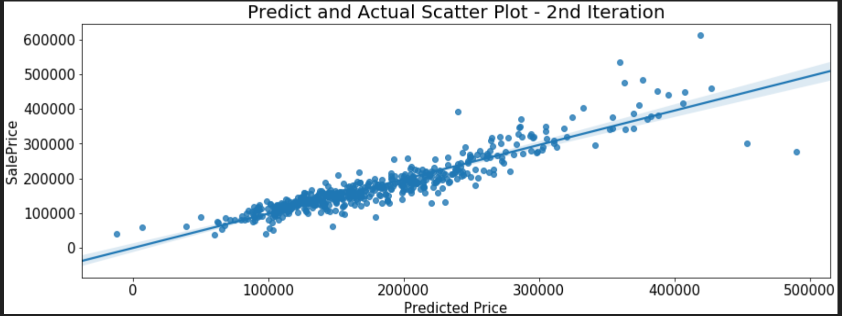
- The second iteration performed better than the first, however the error increased as predicted price increase

- The third iteration is the best when predicted price is low, but declined when predicted price is high

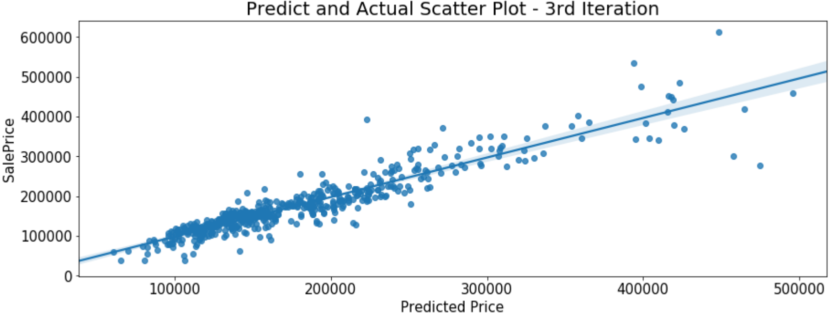
Prediction and Actual Scatter – (Train Test Split)



Predict and Actual Scatter Plot - 1st Iteration



Predict and Actual Scatter Plot - 2nd Iteration



Predict and Actual Scatter Plot - 3rd Iteration

- You would want to see that the points are highly correlated

- The second iteration performed better than the first, as the predicted price increase, the points get scattered

- The third iteration is the best when predicted price is low, but the points are scattered more when the predicted price is high

Conclusion

- The first model submitted to Kaggle for scoring:

  - Public Score : 30, 596

  - Private Score : 34, 528

- The second model submitted to Kaggle for scoring

  - Public Score: 26, 639

  - Private Score: 39, 360

- This means that the second model is not generalized enough
- The first model is more generalized despite the higher error rate
- I will consider the first model as a better model
- More improvements can be made iterating from the first model