Ames Housing Price Prediction

Ivy Chan
Tay Tyn Long
Ling Chong Gold

Agenda

Background

Methodology

Problem Statement

Gather and Data Cleaning

Exploring Data

Model Data

Second Iteration

Third Iteration

Conclusion

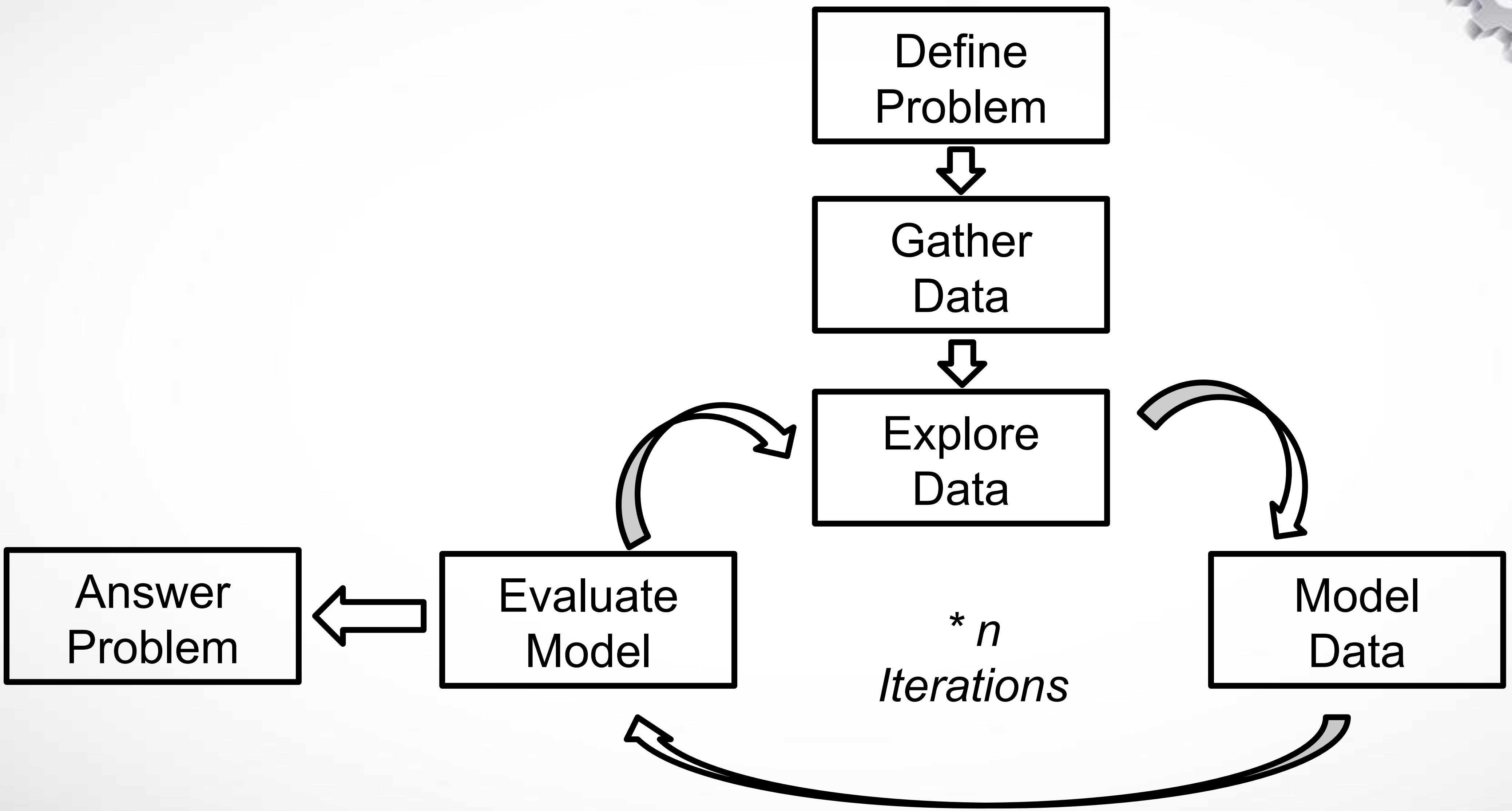
Problem Statement

- To create a Model based on the Ames Housing Dataset to predict the price of a house.
- To Identify what are the features that will influence price of a house in Ames.

Background

- 2 datasets of Aimes Iowa Housing Dataset was Provided
- Training Data
  - 2051 rows of observations with 81 columns
- Test Data
  - 879 rows of obseravtions with 80 columns
- Score is calculated based on the Root Mean Square Error after submission to Kaggle
- Aim to have a generalized model that has good predictive power AND is understandable

Methodology



Data Cleaning

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)

Columns that could not be imputed with ‘None’ or 0

- Lot Frontage- 490 null values
- Garage Yr Blt-117 null values
- Various methods were used

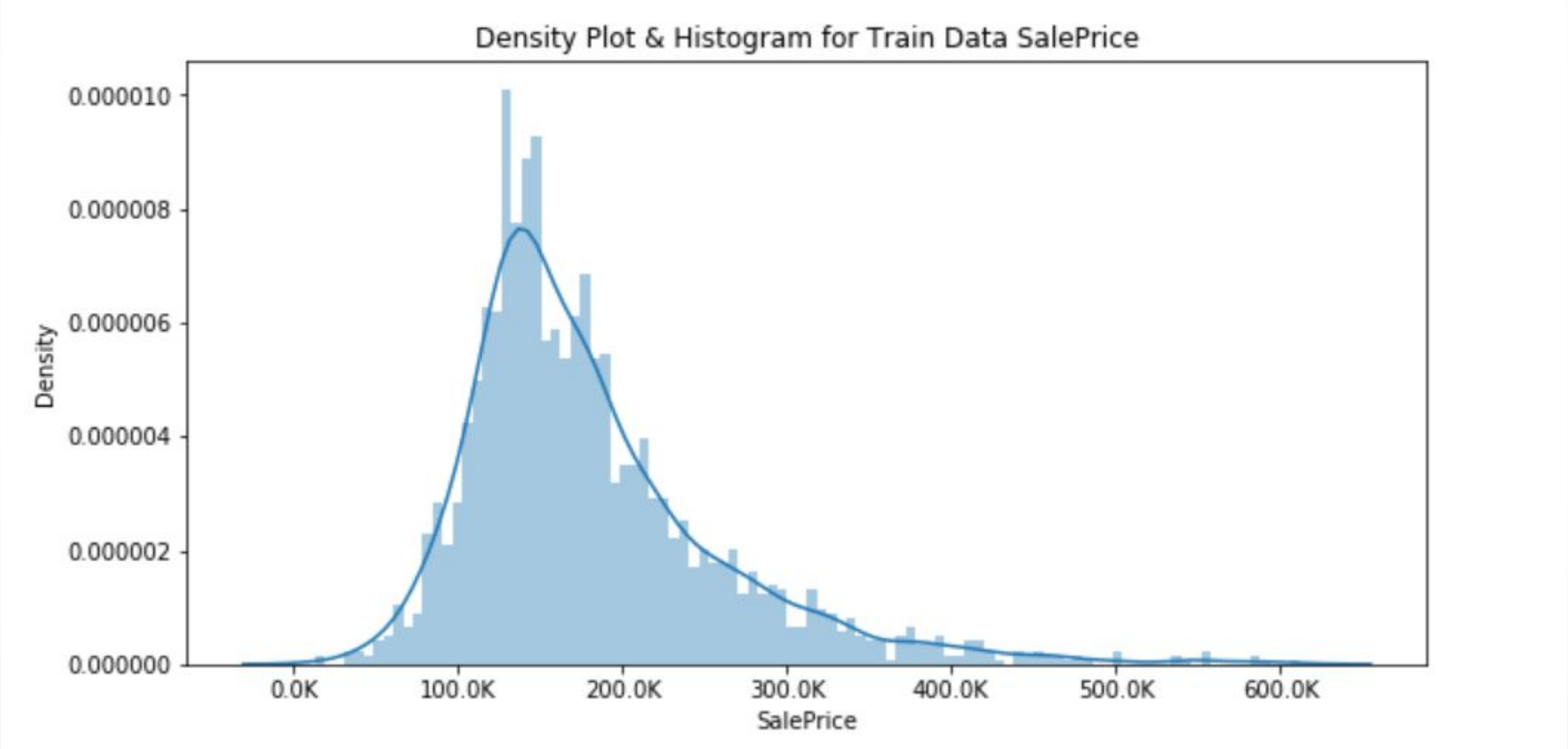
Ordinals and dummy columns

- Referred to data dictionary and converted features to numeric

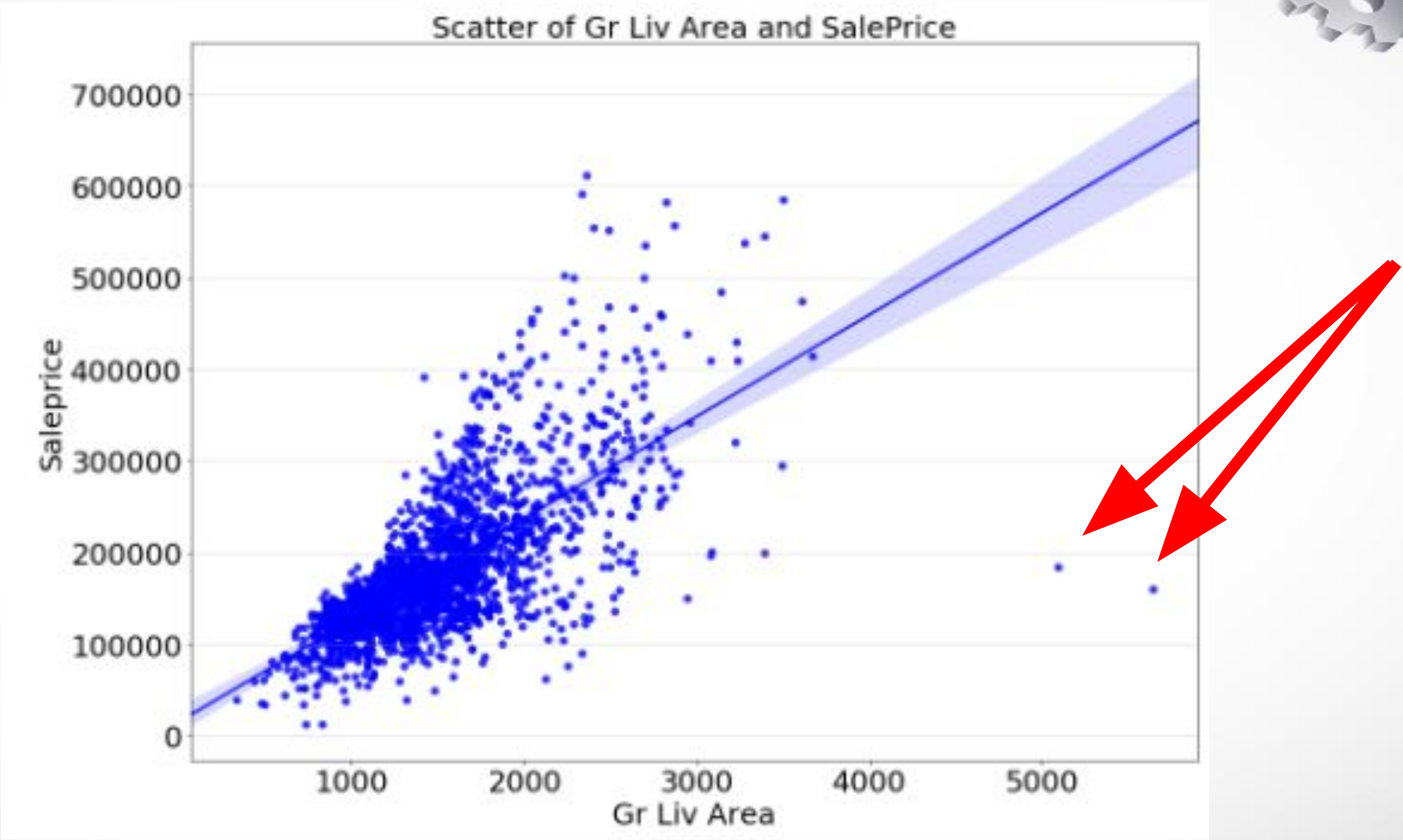
Others

- 3 Rows with Null Values Exclusive to Train dataset was dropped

Distribution of SalePrice in Training Data



Outliers in Training Data



Feature Engineering

Some features created:

- High Quality Finish Area
- Lot Size Overall Quality
- Garage Overall
- Fireplace Overall
- Sale Overall Condition
- Total Baths
- Total Basement SF
- Age When Sold

Feature Selection

Correlation-Selection

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

If 2 features are correlated with each other the one with a higher correlation with sale price is selected

Select interaction predictors based on correlation with Saleprice

Lasso-down

Apply lasso and increase alpha until x variables are left
(Alpha =~2000)

Apply Polynomials

Lasso was performed again to remove poor interaction predictors

Target Encoding

Further reduction of features is desired to simplify the model without losing information by dropping features

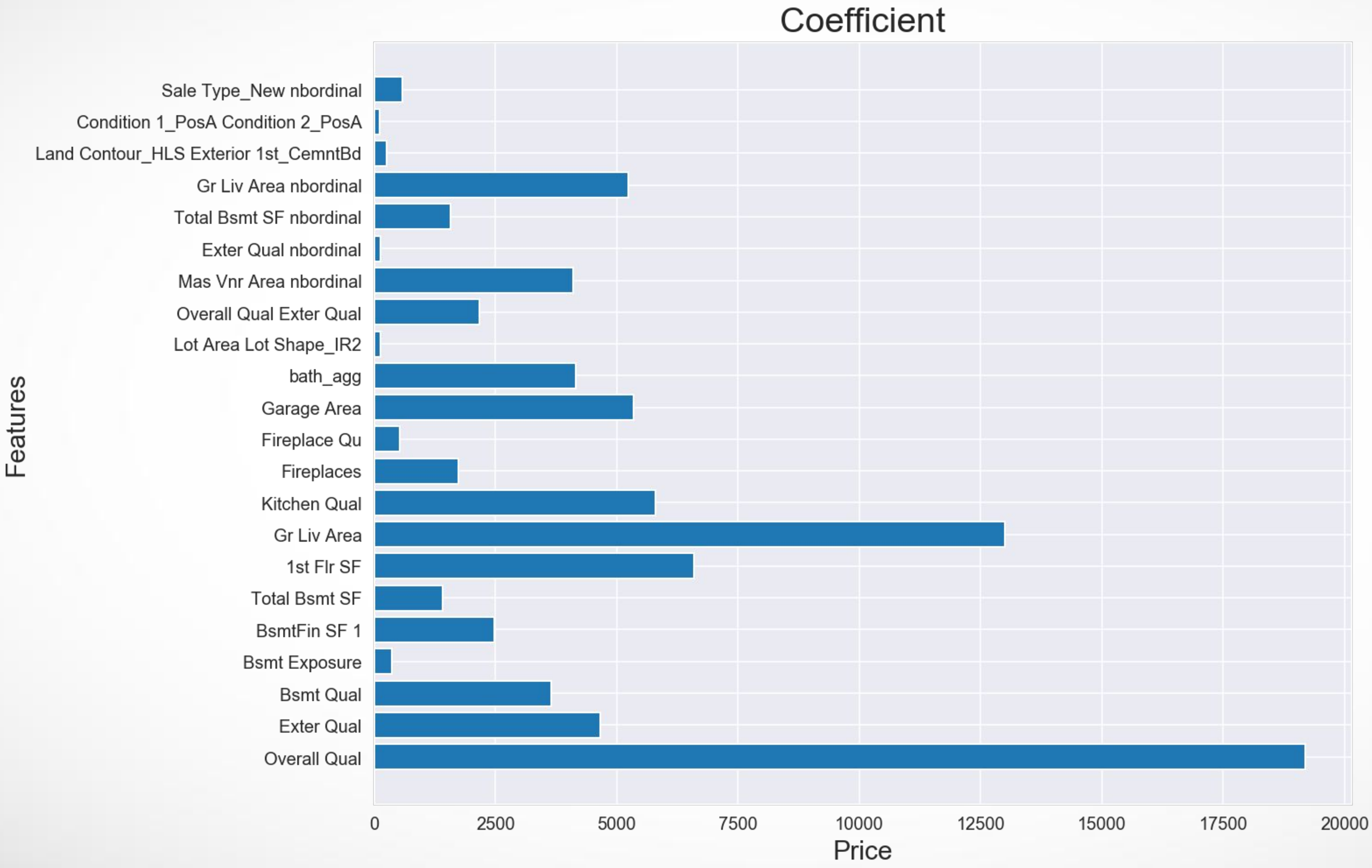
```
def ordinalize(name,listy,df1, df2):
    aggname=name+'ordinal'
    df1[aggname]=np.zeros(shape=df1.iloc[:,0].shape)
    df2[aggname]=np.zeros(shape=df2.iloc[:,0].shape)

    for item in listy:
        itemmean=df1[df1[item]>0]['SalePrice'].mean()
        print(itemmean)
        df1[aggname]=df1[aggname]+df1[item]*itemmean
        df2[aggname]=df2[aggname]+df2[item]*itemmean

    nnbavg=df1[df1[aggname]==0]['SalePrice'].mean()
    df1[aggname]=df1[aggname].replace(0,nnbavg)
    df2[aggname]=df2[aggname].replace(0,nnbavg)

    df1=df1.drop(columns=listy,inplace=True)
    df2=df2.drop(columns=listy,inplace=True)
    return
```

Results



Coefficient
Sale Type_New nbordinal
Condition 1_PosA Condition 2_PosA
Land Contour_HLS Exterior 1st_CemntBd
Gr Liv Area nbordinal
Total Bsmt SF nbordinal
Exter Qual nbordinal
Mas Vnr Area nbordinal
Overall Qual Exter Qual
Lot Area Lot Shape_IR2
bath_agg
Garage Area
Fireplace Qu
Fireplaces
Kitchen Qual
Gr Liv Area
1st Flr SF
Total Bsmt SF
BsmtFin SF 1
Bsmt Exposure
Bsmt Qual
Exter Qual
Overall Qual
Features
0
2500
5000
7500
10000
12500
15000
17500
20000
Price

Conclusions

Bias vs Variance

- Low Public vs High Private Scores

Simple vs Complex Models

- Tradeoff between comprehensibility vs. predictive power

Which model to choose? Depends on the application