Kaggle Competition: Predicting Housing Sales Price in Ames, Iowa

August 2019

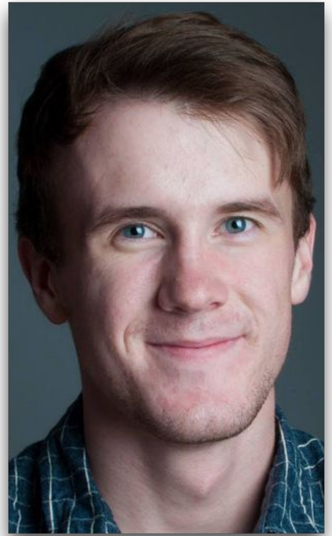
Fred(Lefan) Cheng
Paul Dingus
Wenjun Ma
Haoyun Zhang

Team Introduction

Please contact us if your company is looking for Data Science or Data Analytics talents

Fred(Lefan) Cheng

- Linkedin: https://www.linkedin.com/in/lefancheng/
- Email: fredchengnyc@gmail.com



Paul Dingus

- Linkedin: https://www.linkedin.com/in/paul-dingus/
- Email: williampdingus@gmail.com

Wenjun Ma

- Linkedin: https://www.linkedin.com/in/wenjun-ma-phd/
- Email: wenjunma0303@gmail.com

Haoyun Zhang

- Linkedin: https://www.linkedin.com/in/Haoyun-Zhang-UPenn/
- Email: haoyunz@sas.upenn.edu

Data Exploration

The Purpose is to Predict Housing Price with Least Error (RMSE)

We trained 5 models, which are proven to be effective, to predict prices based on houses information

Over 80 Input Variables: Information about houses

- Overall Quality
- Above-Grade Living Area
- Size of garage in square feet
- .
- .
- .
- Original construction date
- Neighborhood
- Lot size in square feet

Lasso Regression
Ridge Regression
Elastic Regression

+

Gradient Boosting
XGBoosting

⇓

Stack Regressor

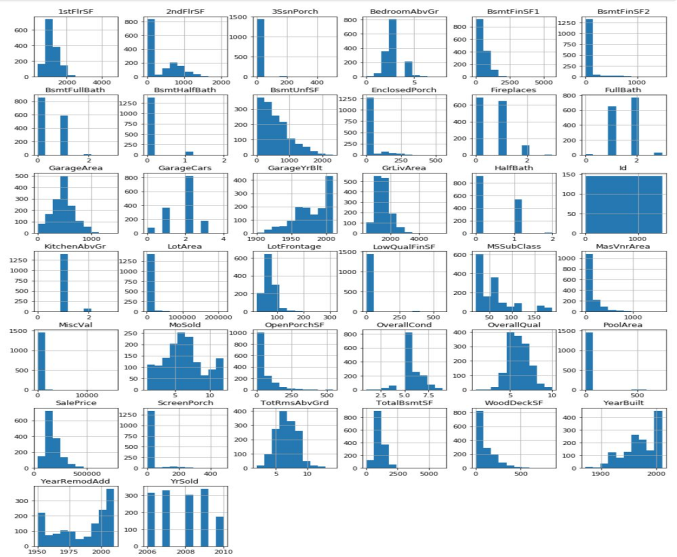
Output Variable: Housing Sales Price

RMSE: Root Mean Squared Error

A glance of dataset

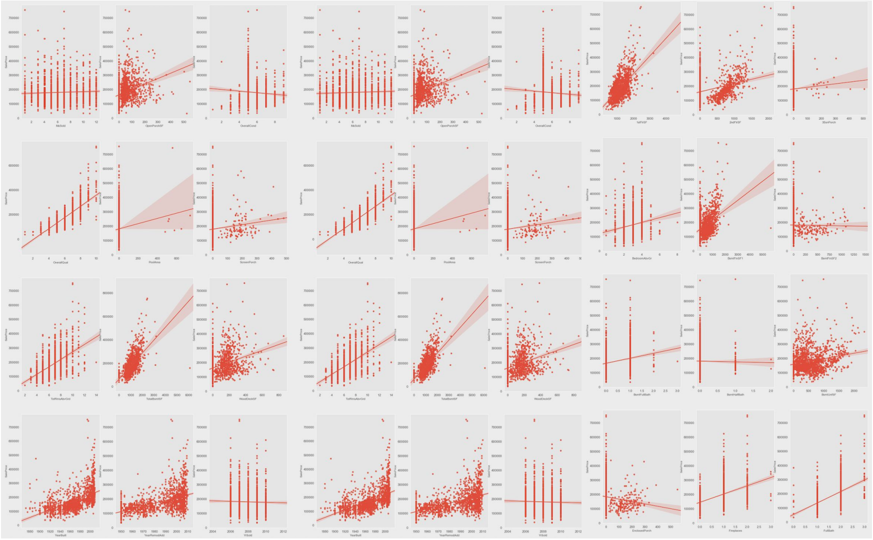
Explorating dataset is the foundation of the following pre-processing and modeling

Distributions of variables



Some variables are significantly skewed that might need to be standardized

Relationship between input and output variables

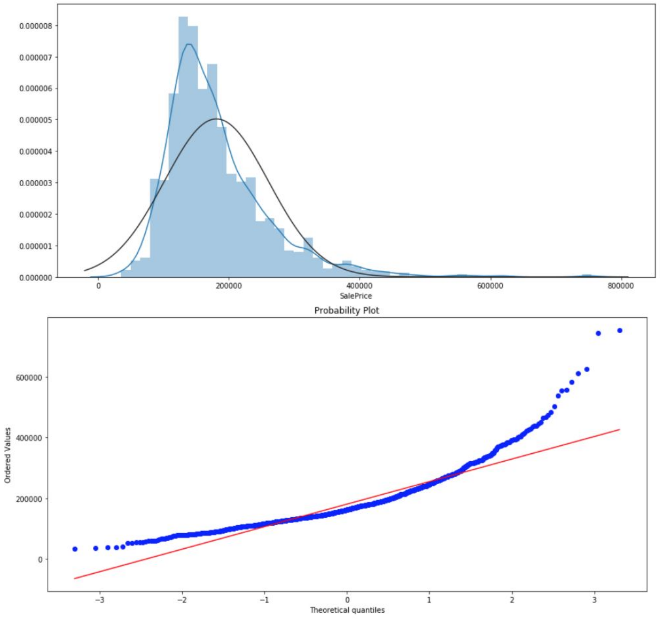


Outliers exist and some variables have strong linear relationship with price

Transform the Target Feature by Taking the Log for Normalization

A violation of normality assumption of linear regression manifests in the target feature that it’s notable right skew.

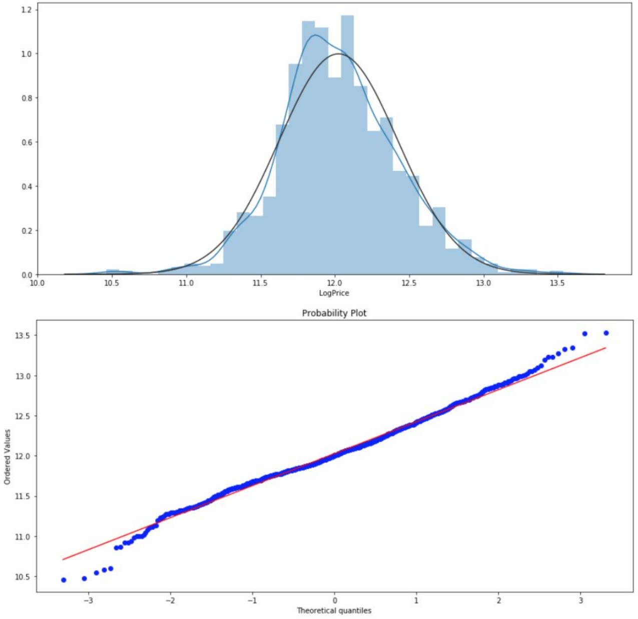
Before Transformation



Distribution of y

Normal distribution

After Transformation



Distribution of y

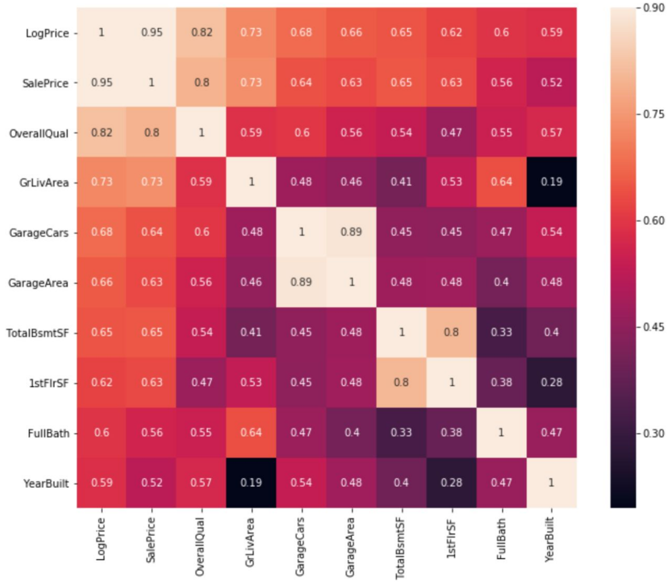
Normal distribution

It would be advantages to work with the normally distributed output variable (Sales Price)

Removing Outliers of Important - Highly Correlated - Features

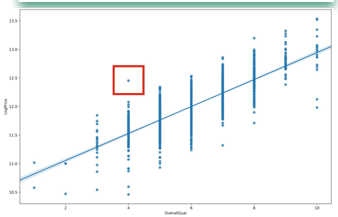
Filter out outliers in important features that have high correlation with Sales Price and remove them.

Before Removing Outliers

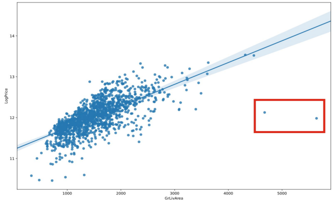


Select important features with high correlation with price

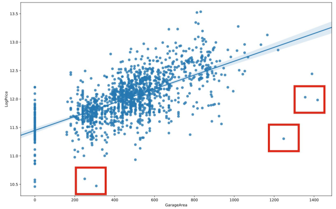
Overall Quality



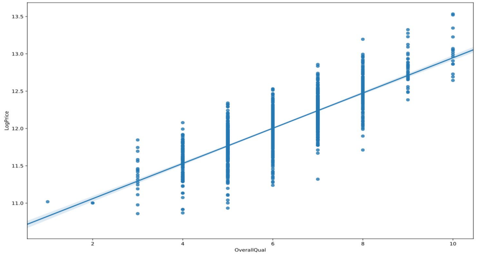
Above-Grade Living Area

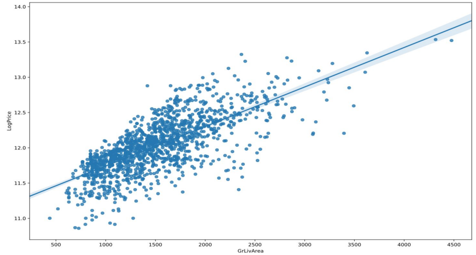


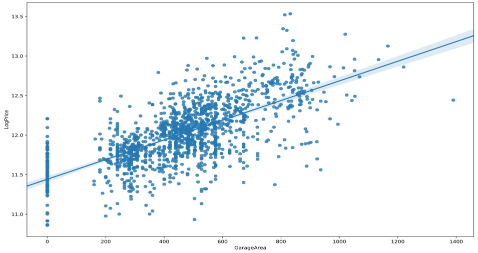
Size of garage in square feet



After Removing Outliers



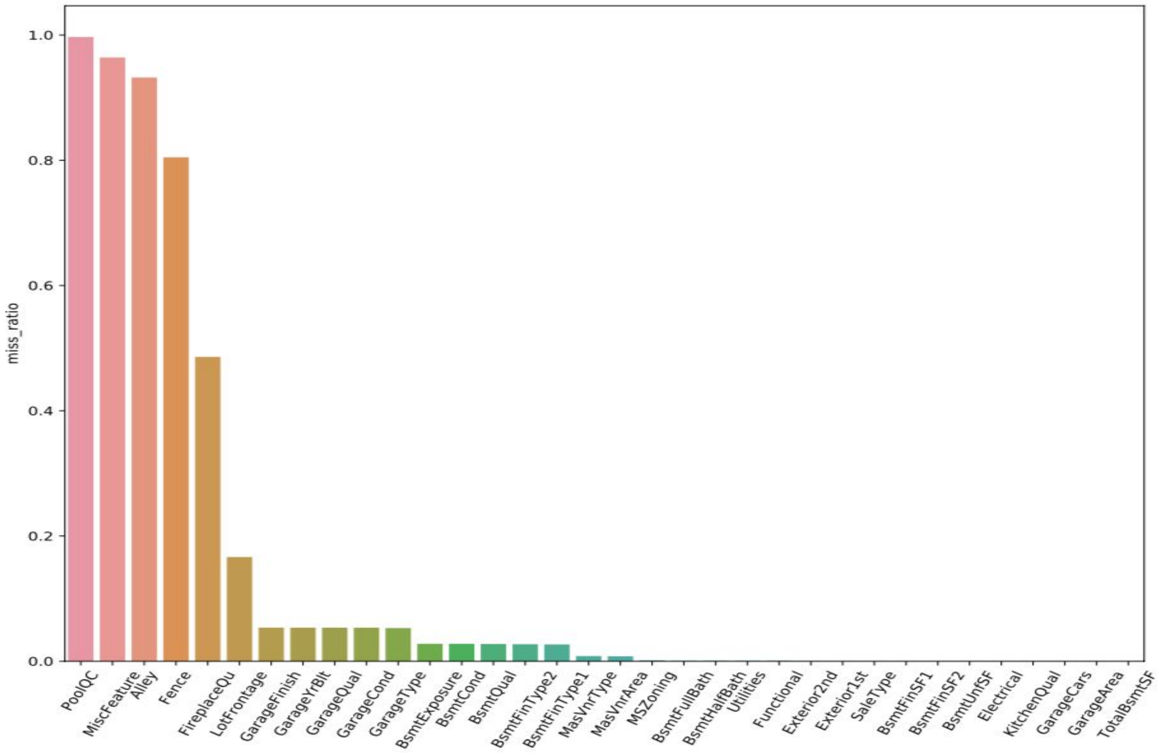




Missing Values

General view of missing values

Ratio of missing values of each feature



Number of Missing values
13965

Train Dataset	Test Dataset
6965	7000

Number of Features with missing values
34

Train Dataset	Test Dataset
19	33

Missing value imputation of train dataset

Pseudo Missing Values

```
-------------------------
Alley ------------- 1369
PoolQC ------------ 1453
MiscFeature ------- 1406
Fence ------------- 1179
FireplaceQu ------- 690
GarageType -------- 81
GarageYrBlt ------- 81
GarageFinish ------ 81
GarageQual -------- 81
GarageCond -------- 81
-------------------------
```

No Alley Access
No Pool
All Covered
No Fence
No Fireplace
No Garage

Impute with ‘No ***’

Real Missing Values

```
-------------------------
BsmtQual ---------- 37
BsmtCond ---------- 37
BsmtFinType1 ------ 37
BsmtFinType2 ------ 38
BsmtExposure ------ 38
lotFrontage ------- 259
MasVnrType -------- 8
MasVnrArea -------- 8
Electrical -------- 1
-------------------------
```

37 No Basement
Id 949 misses Exposure
Id 333 misses FinType2
259 miss lot Frontage
8 miss MasVnrType
8 miss MaxVnrArea
1 Electrical

Features	Group	Imputation
BsmtExposure BsmtFinType2	Neighborhood YearBuilt	Mode
LotFrontage	Neighborhood	Median
MasVnrType MasVnrArea	Neighborhood YearBuilt	Mode
Electrical	YearBuilt	SBrkr

Missing value imputation of test dataset

```
BsmtCond ---------- 45
BsmtQual ---------- 44
BsmtExposure ------ 44
BsmtFinType1 ------ 42
BsmtFinType2 ------ 42
BsmtFinSF1 -------- 1
BsmtFinSF2 -------- 1
BsmtUnfSF --------- 1
TotalBsmtSF ------- 1
BsmtFullBath ------ 2
BsmtHalfBath ------ 2
```

Id 2218
Id 2219
Id 2349
Wrong inputs

No Basement

```
GarageQual -------- 78
GarageCond -------- 78
GarageYrBlt ------- 78
GarageFinish ------ 78
GarageType -------- 76
GarageCars -------- 1
GarageArea -------- 1
```

Id 2127
Id 2577
Wrong inputs

No Garage

```
lotFrontage ------- 227
MasVnrType -------- 16
MasVnrArea -------- 15
MSZoning ---------- 4
Utilities --------- 2
Exterior1st ------- 1
Exterior2nd---- --- 1
KithchenQual ------ 1
Functional -------- 2
SaleType ---------- 1
```

LotFrontage	Median
MasVnrType	None+BrkFace
MasVnrArea	0
MSZoning	RM+RL
Utilities	AllPub
Exterior	VinylSd
KitchenQual	TA
Functional	Mod
SaleType	WD

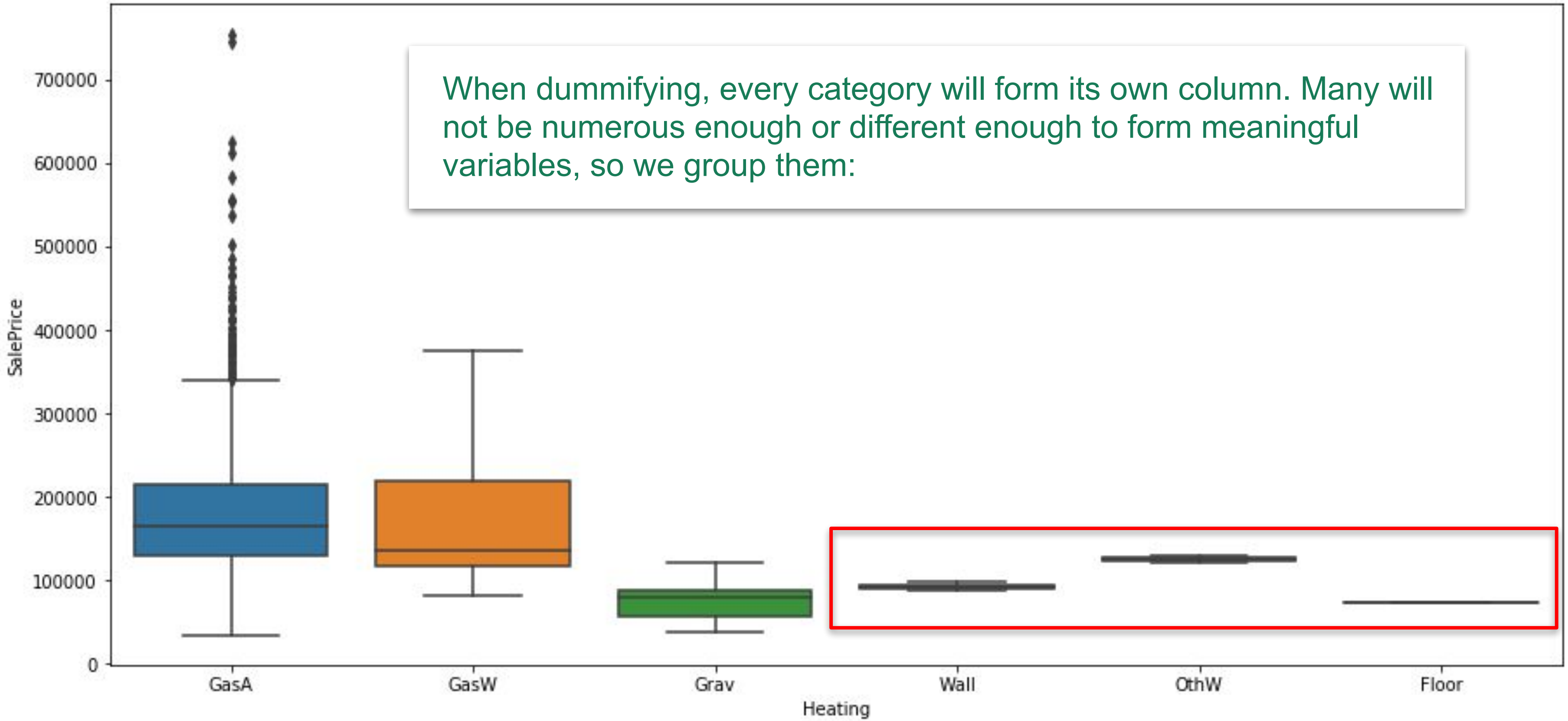
Feature Engineering

Dealing with different types of data

Grouping data into different categories can help to sort through it and organize it effectively

Type of variable	Transformation	Examples
Continuous	Just ensure that the variable is numeric: `column.astype('float64')`	LotFrontage, LotArea, MassVnrArea, BsmtFinSF1
Ordinal Categorical	Manually encode the variables: `[‘Po’, ‘Fa’, ‘Av’, ‘Gd’, ’Ex’]` → `[2, 4, 6, 8, 10]`	OverallQual, OverallCond, ExterQual, BsmtCond
Nominal Categorical	Dummify the variables: `pd.get_dummies(column)`	MSSubClass, MSZoning, LotConfig

Dummifying Data



When dummifying, every category will form its own column. Many will not be numerous enough or different enough to form meaningful variables, so we group them:

Dummifying Data

RoofMat1	['Membran', 'ClyTile', 'Metal', 'Roll', 'WdShngl', 'WdShake']	→ 'Others'
PoolQC	['Ex', 'Gd', 'Fa' 'Have_Pool']	→ 'Have_Pool'
Condition2	['RRAn', 'RRAe']	→ 'Norm'
	[RRNn', 'Artery', and 'Feedr']	→ 'Other'
	['PosA', 'PosN']	→ 'Pos'
Heating	['Wall', 'OthW', 'Floor']	→ 'Other'
MiscFeatures	['TenC', 'Othr']	→ 'Other'

Dealing with Skewness

Reducing the skew in our continuous or ordered features will help our modeling. We applied the box-cox transformation to particularly skewed data.

If skew > threshold → Calculate skewness after log transform

If log transform reduces skew → Apply log transform, repeat.

For some variables, we found it was better to manually apply power transformations to reduce extreme skew. This was done for BsmtCond, BsmtQual, GarageCond, and GarageQual.

Dealing with Skewness

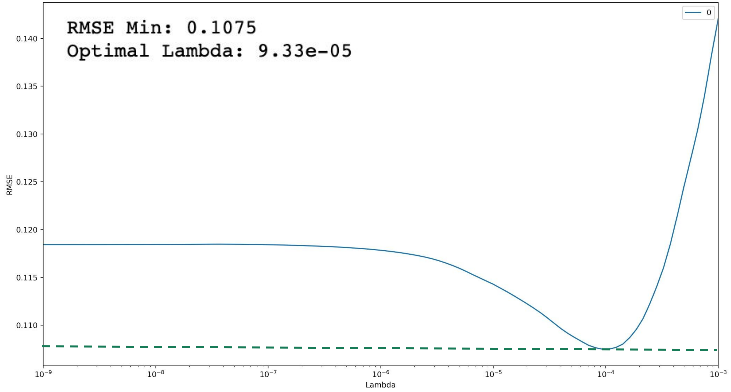
Skewed Data	Before Transformation	After Transformation
BsmtCond	-3.605053	-0.729901
GarageCond	-3.409930	0.510577
GarageQual	-3.284772	0.692163
BsmtQual	-1.267708	0.358122
TotRmsAbvGrd	0.757017	0.051948
ExterQual	0.789348	0.463858
2ndFlrSF	0.859141	0.304594
BsmtUnfSF	0.918694	0.918694
BsmtFinSF1	0.981443	-0.618988
GrLivArea	1.077108	0.014593
LotFrontage	1.102316	-1.072827
BsmtExposure	1.121619	0.200598
1stFlrSF	1.266426	0.048269
ExterCond	1.336540	-0.065391
WoodDeckSF	1.846248	0.157069
OpenPorchSF	2.477720	-0.043021
MasVnrArea	2.617879	0.535434
BsmtFinType2	3.147874	-1.085226
ScreenPorch	3.938725	2.915270
EnclosedPorch	4.010200	1.904747
BsmtFinSF2	4.137937	2.375989
3SsnPorch	11.356127	8.726202
LowQualFinSF	12.067635	8.395317
LotArea	13.310014	-0.553923
MiscVal	21.919304	5.084193

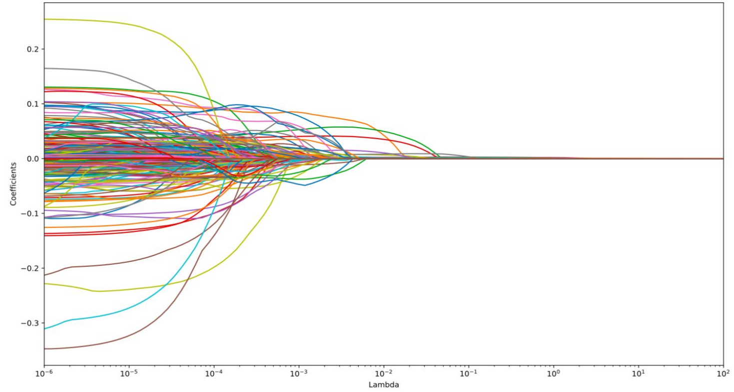
Model Fitting

Feature Selection via Lasso Regression

Analyze the lasso regression plot to decide which features need to be dropped

rMSE ~ λ | coeff ~ λ





Feature Dropped

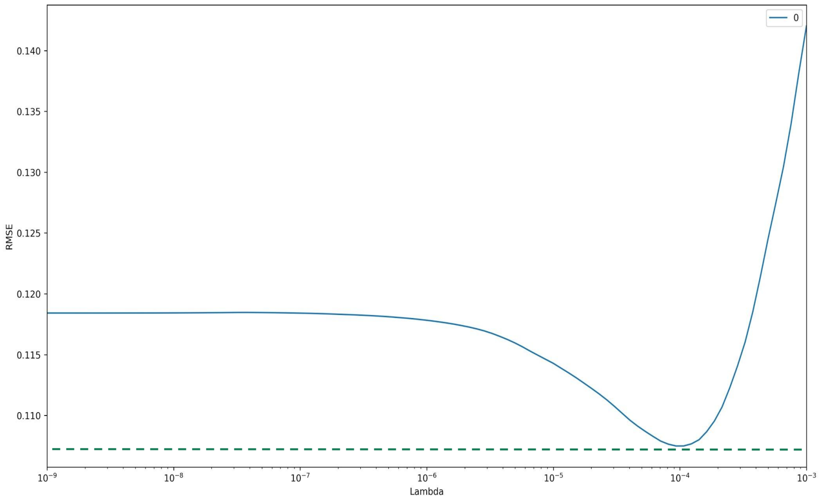
```
Numbers of Features Dropped: 119
```

	Features	Lambda
0	LandContour_Lvl	1.072267e-08
1	BsmtFullBath_1.0	5.722368e-08
4	LotConfig_Inside	2.009233e-07
5	Heating_GasA	2.656088e-07
6	LandSlope_Gtl	3.511192e-07
9	MSZoning_RL	1.629751e-06
10	PoolQC_Fa	1.873817e-06
11	Foundation_CBlock	2.154435e-06
12	Alley_None	4.328761e-06
13	Exterior2nd_MetalSd	4.977024e-06

Parameter Optimization of Lasso Regression

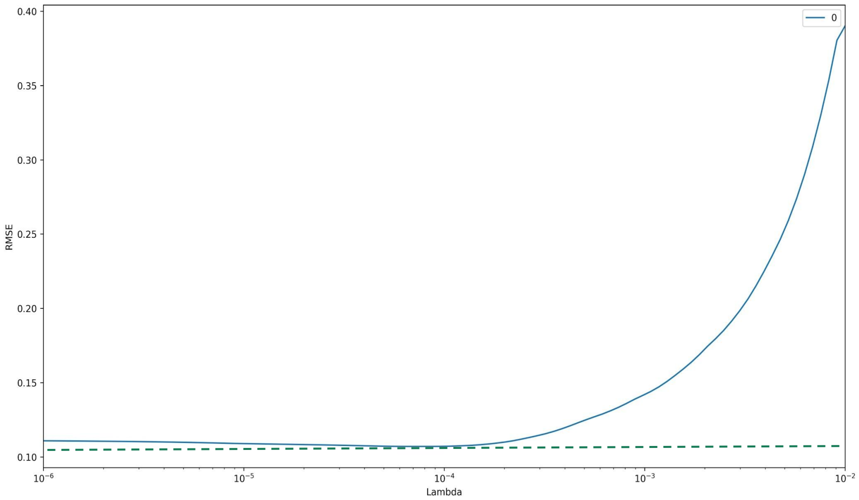
Comparison of feature selection

Before Feature Selection



RMSE Min: 0.1075
Optimal Lambda: 9.33e-05

After Feature Selection

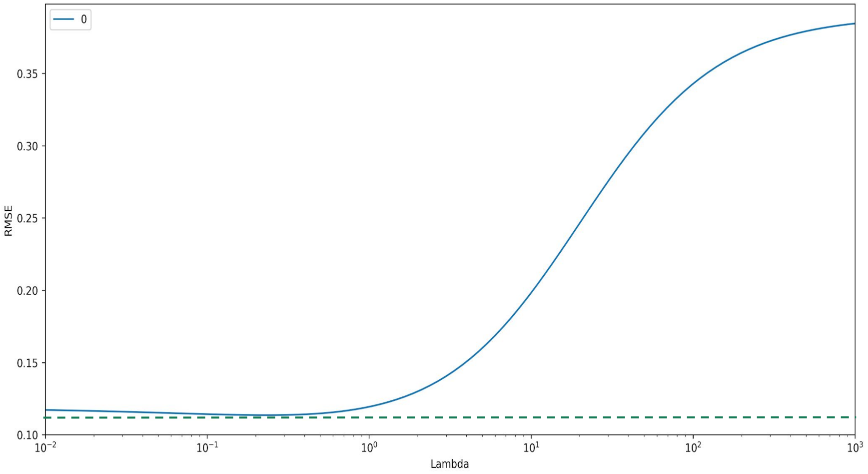


RMSE Min: 0.1071
Optimized lambda: 7.22e-05

Parameter Optimization of Ridge Regression

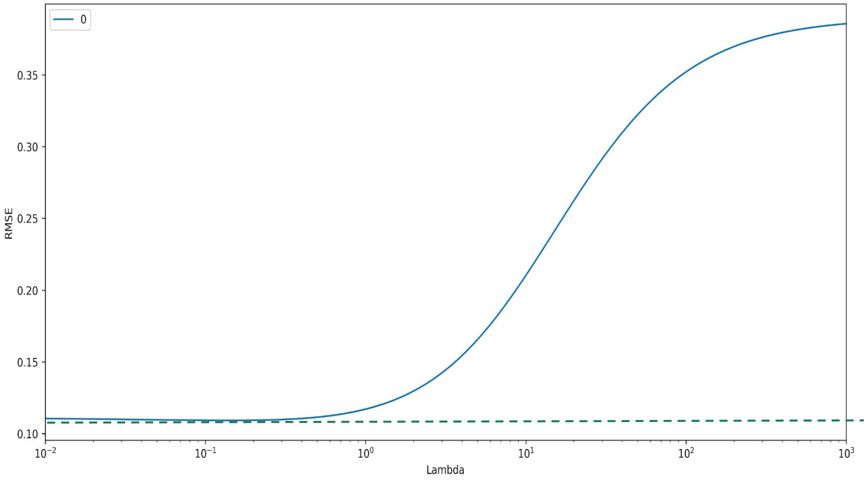
Comparison after feature selection

Before Feature Selection



RMSE Min: 0.1135
optimized lambda: 0.231

After Feature Selection



RMSE Min: 0.1091
optimized lambda: 0.145

Price Prediction by Elastic Net Regression

Model comparison and price prediction

Lasso Regression

```
R2 before Dropping is: 0.9233 / RMSE before Dropping: 0.10749
R2 std before Dropping is: 0.0077 / RMSE Std before Dropping: 0.00847
---------------------------------------------------------------------
R2 after Dropping is: 0.9243 / RMSE after Dropping: 0.1071
R2 std after Dropping is: 0.0079 / RMSE Std after Dropping: 0.00833
```

Ridge Regression

```
R2 before Dropping is: 0.9148 / RMSE before Dropping: 0.1135
R2 std before Dropping is: 0.0068 / RMSE Std before Dropping: 0.0094
---------------------------------------------------------------------
R2 after Dropping is: 0.9216 / RMSE after Dropping: 0.1091
R2 std after Dropping is: 0.0058 / RMSE Std after Dropping: 0.0077
```

Elastic Net Regression

```
R2 before Dropping is: 0.9231 / RMSE before Dropping: 0.10799
R2 std before Dropping is: 0.0079 / RMSE Std before Dropping: 0.008935
---------------------------------------------------------------------
R2 after Dropping is: 0.9241 / RMSE after Dropping: 0.107218
R2 std after Dropping is: 0.0084 / RMSE Std after Dropping: 0.008672
```

Lasso

Id	SalePrice
1461	121722.57280509300
1462	162883.24122821200
1463	183499.5354773000
1464	195278.28425508900
1465	201451.36481446300
1466	170683.33384617700
1467	176268.4884023650
1468	161506.94628051700
1469	198094.4361075500

1039 wenjun0303 0.11935 7 ~10s

Your Best Entry ↑

You advanced 249 places on the leaderboard!

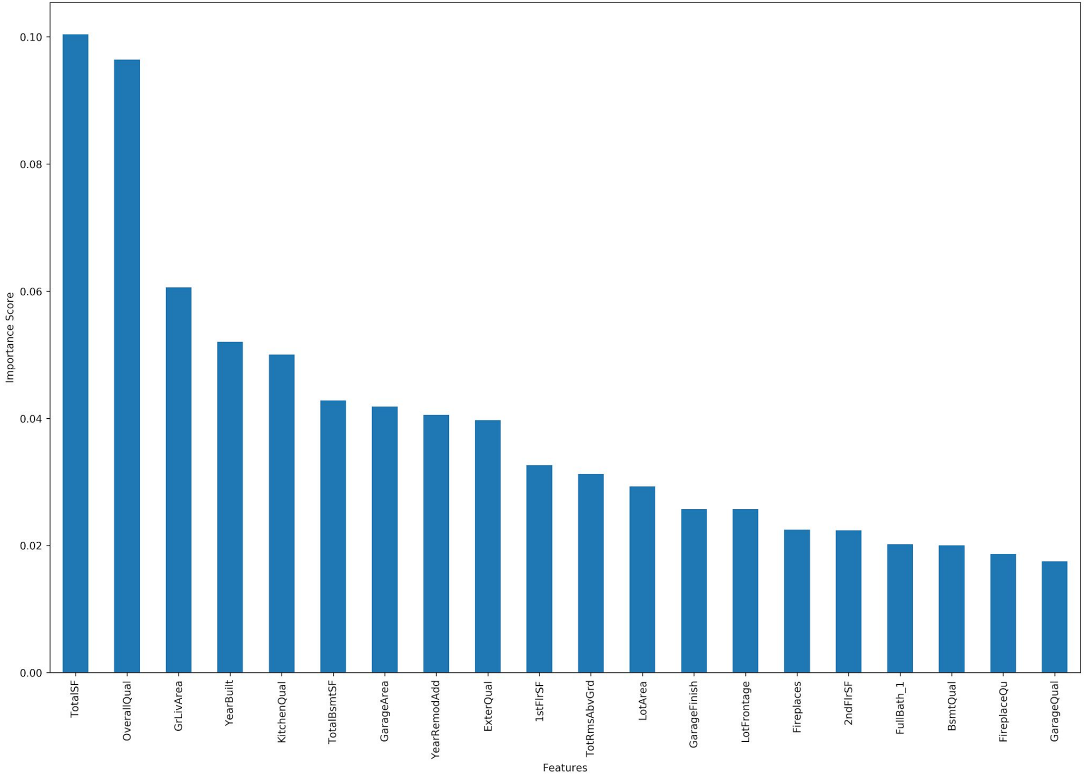
Your submission scored 0.11935, which is an improvement of your previous score of 0.12166. Great job!

Tweet this!

Gradient Boosting

Feature Importance via Gradient Boosting

Feature Importance Score

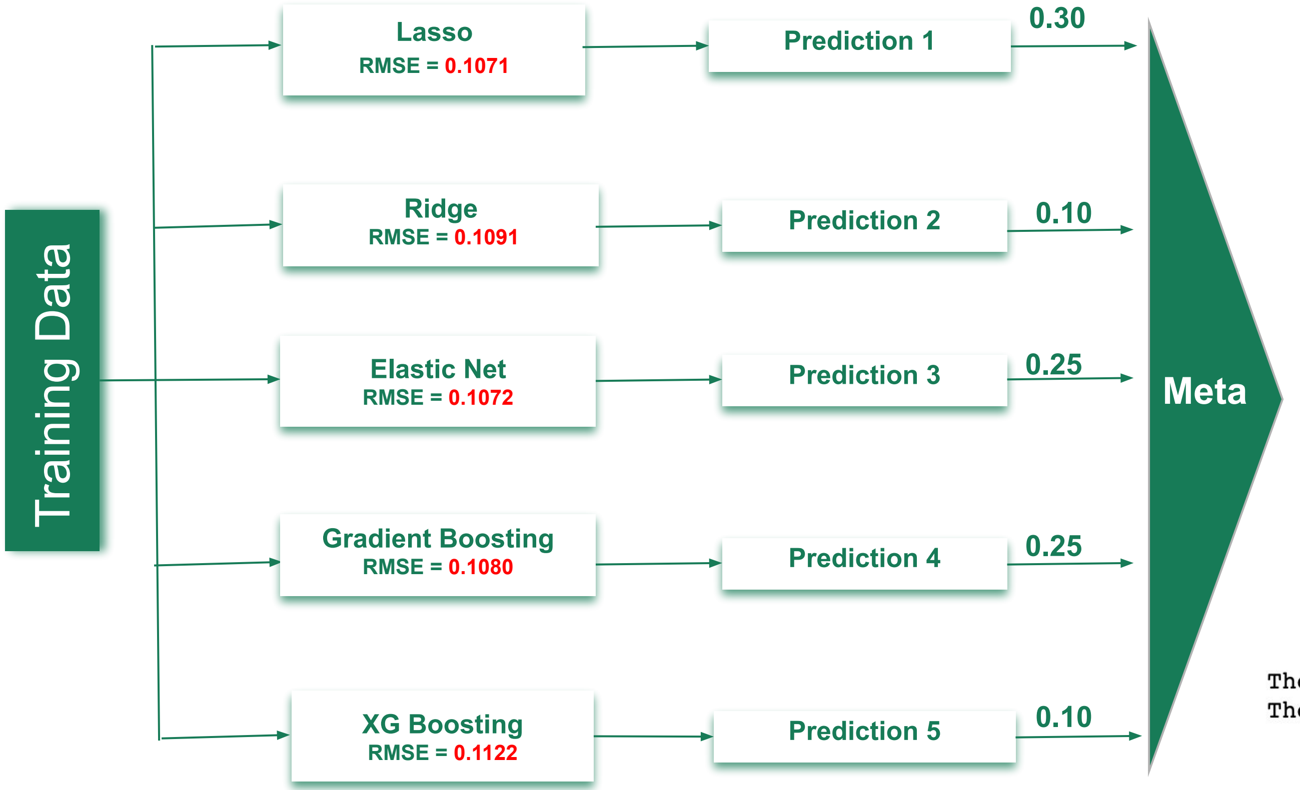


Feature Dropped by Lasso

	feature_name	importance_score
3	2ndFlrSF	0.022378
92	FullBath_1	0.020209
95	HalfBath_0	0.012514
5	GarageYrBlt	0.011994
93	FullBath_2	0.007853
6	GarageCond	0.006208
99	GarageType_Attchd	0.005253
74	Exterior2nd_VinylSd	0.004791
37	Neighborhood_NAmes	0.004009
79	Foundation_CBlock	0.003179
87	BsmtFullBath_1.0	0.003002
38	Neighborhood_OldTown	0.002543
65	Exterior1st_VinylSd	0.002511

Stack Regressor

Stack All Models(ridge, lasso, elastic, xg boosting, g boosting) and Price Prediction



Id	SalePrice
1461	121722.57280509300
1462	162883.24122821200
1463	183499.5354773000
1464	195278.28425508900
1465	201451.36481446300
1466	170683.33384617700
1467	176268.4884023650
1468	161506.94628051700
1469	198094.44361075500

```
The Final Stack RSME: 0.1049
The Final Stack RSME Std: 0.00673
```

763	**Haoyun**	0.11683

Thank you!