



NYCDSA Machine Learning Project:

House Prices: Advanced Regression Techniques

Iris (Jo Wen) Chen, Jialan Zhu, Nikhil Taparia, Lukas Frei

The Project

“Predict sales prices and practice feature engineering, RFs, and gradient boosting”

1,460

OBSERVATIONS

Medium-sized dataset

80

FEATURES

Detailed descriptions and
circumstances of sales

Sale Price

DEPENDENT VARIABLE

Regression

4,357

TEAMS

Highly popular competition
on Kaggle

Workflow

In conducting our research, we emphasized on a clearly defined workflow to increase the degree of comprehensibility of our findings



EDA

Numerical and visual
exploratory data analysis



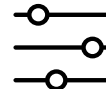
DATA PREPARATION

Analyzing features, dropping
statistically irrelevant / highly
correlated features, as well as
filling in NA's



FEATURE ENGINEERING

Creating two new features

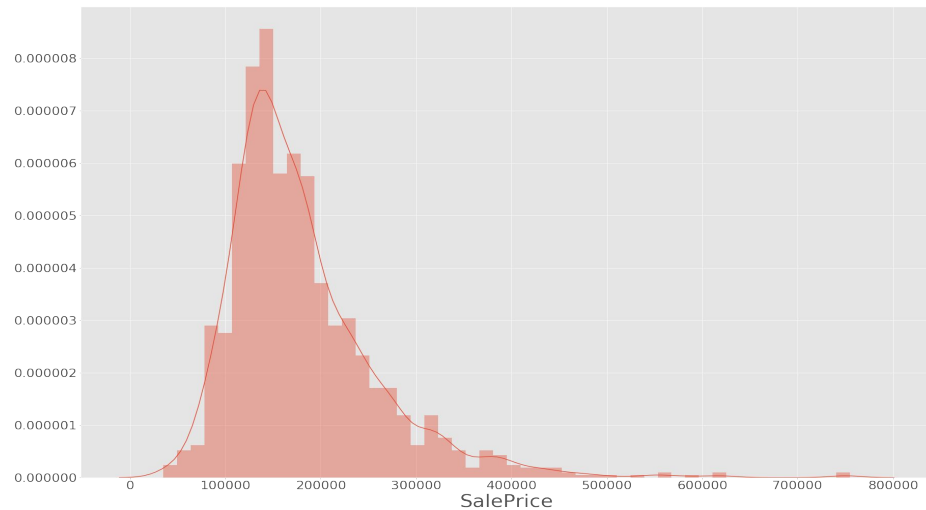


TRAINING & TUNING MODELS

Training and tuning several ML
algorithms in sklearn

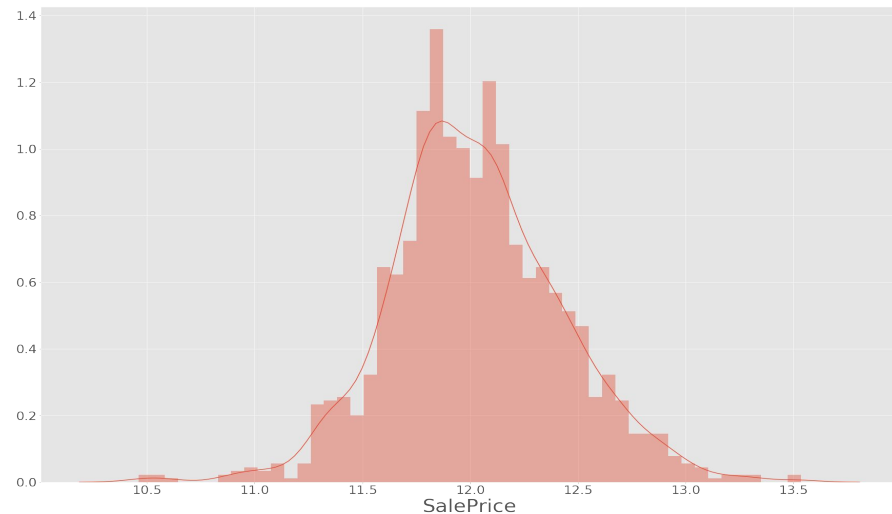
Sale Price Original

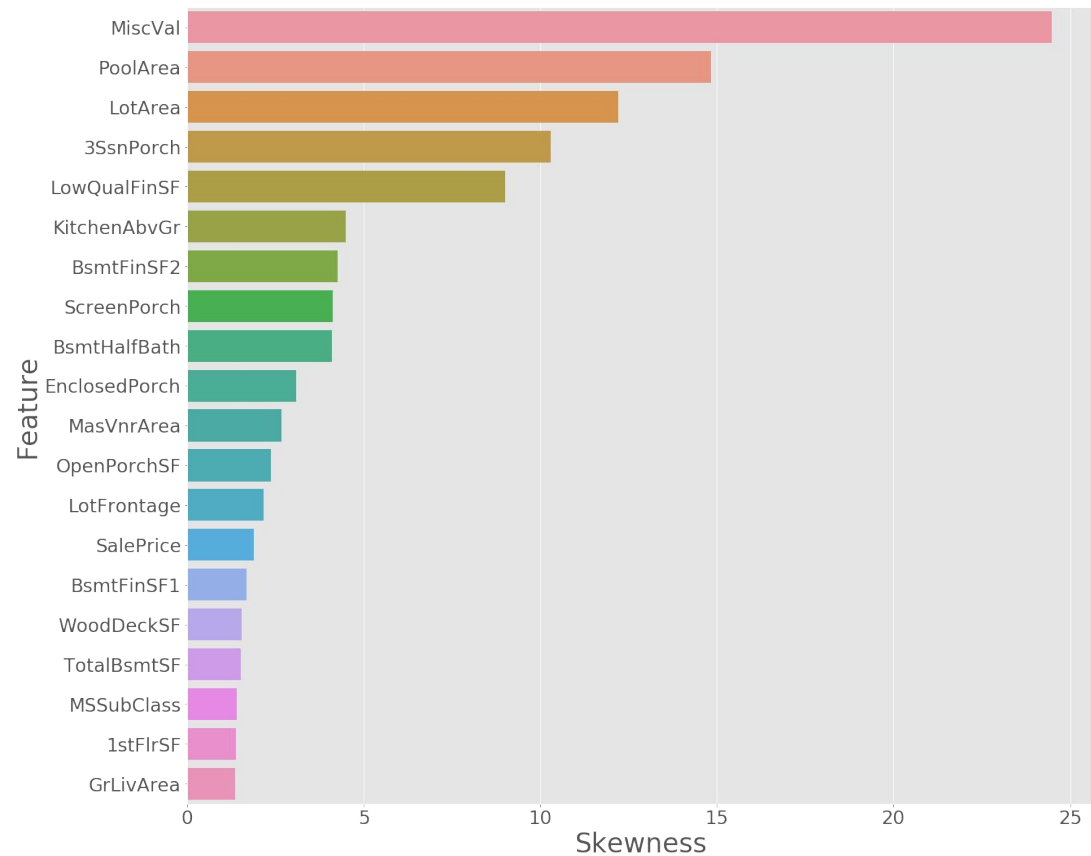
Heavily right-skewed distribution



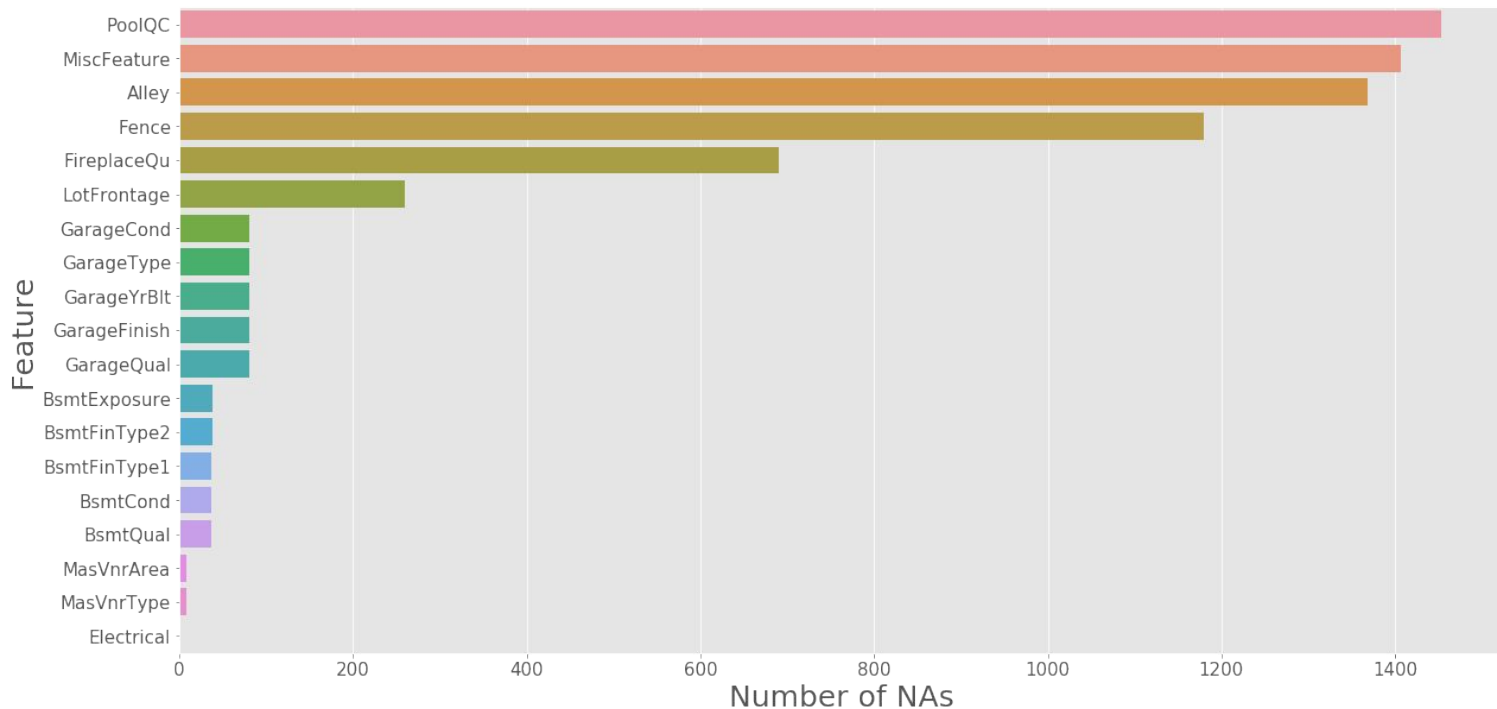
Sale Price After Log+1

Log transformation of the sale price distribution to
achieve approx. normal distribution





**FEATURES
WITH
SKEWNESS > 1**



Lots of NAs

Some features almost exclusively filled with NAs

NAs with meaning

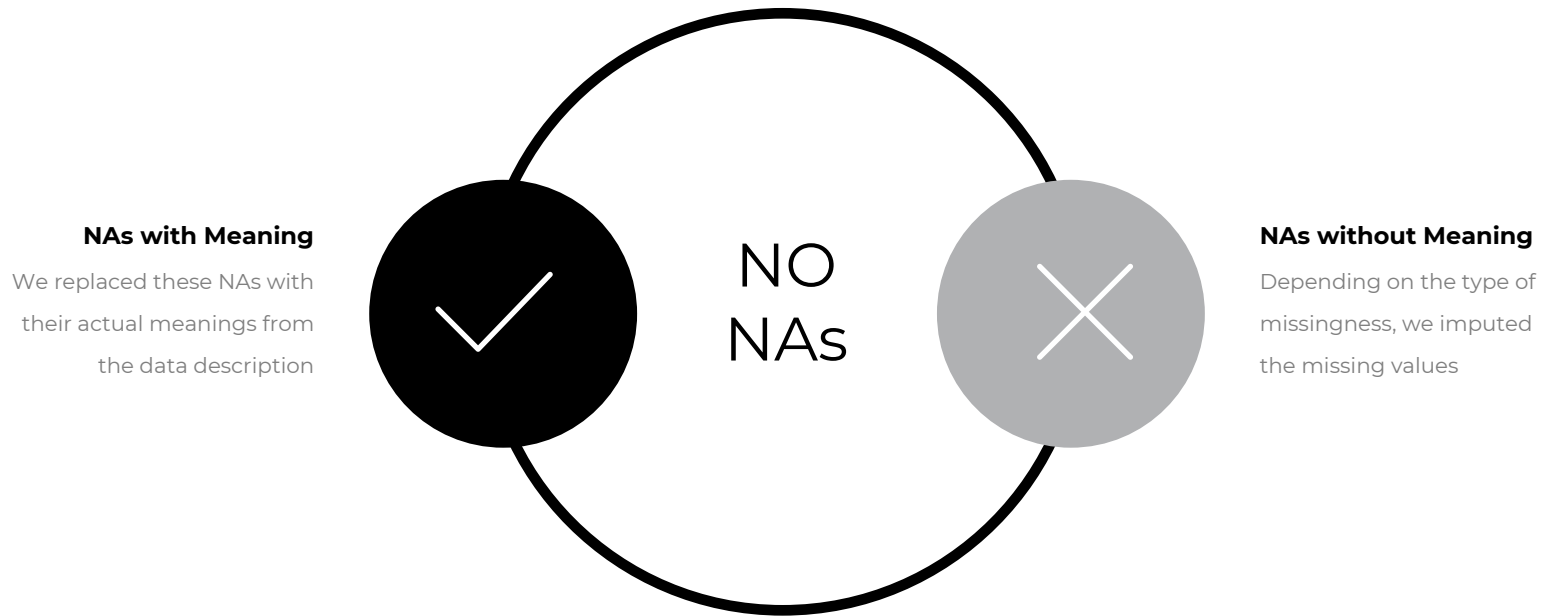
Some NAs have explicit meanings and do not represent missing observations

Different types of NAs

Not all NAs have meaning

DEALING WITH NAs

The different types of NAs within the data set require different forms of treatment in order to avoid suffering a loss of information.



FEATURE ENGINEERING

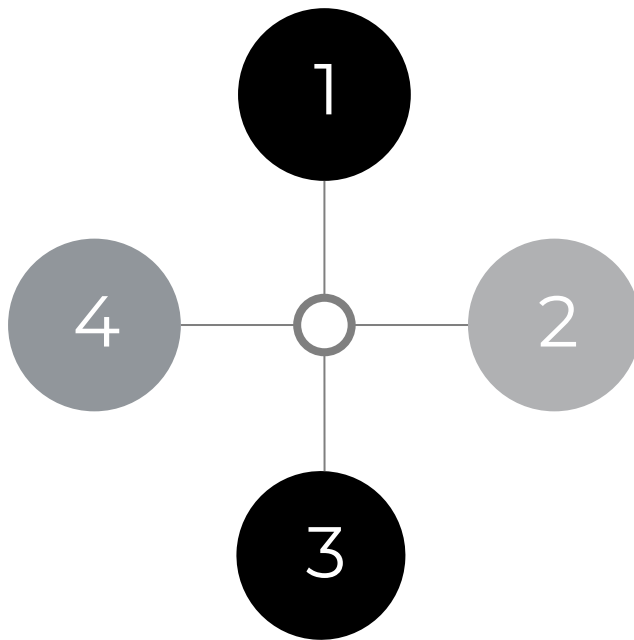
We individually analyzed features and discussed their relevance to this project

DROPPING FEATURES

We decided to drop features irrelevant to the Sales Price and features with high multicollinearity

Bathrooms

$\text{BsmtFullBath} + \text{FullBath} + .5 * (\text{BsmtHalfBath} + \text{HalfBath})$



ORDINAL ENCODING

Replacing ordinal categorical features with corresponding integers

YearSinceRemod

$\text{YrSold} - \text{YearRemodAdd}$

Regression Algorithms

We decided to apply both linear and non-linear algorithms in order to determine the best fit for our data

LINEAR

Starting Point

Multiple Linear Regression
and Penalized Regressions
(Ridge, Lasso, Elastic Net)

NON-LINEAR

Basic Algorithms

KNN

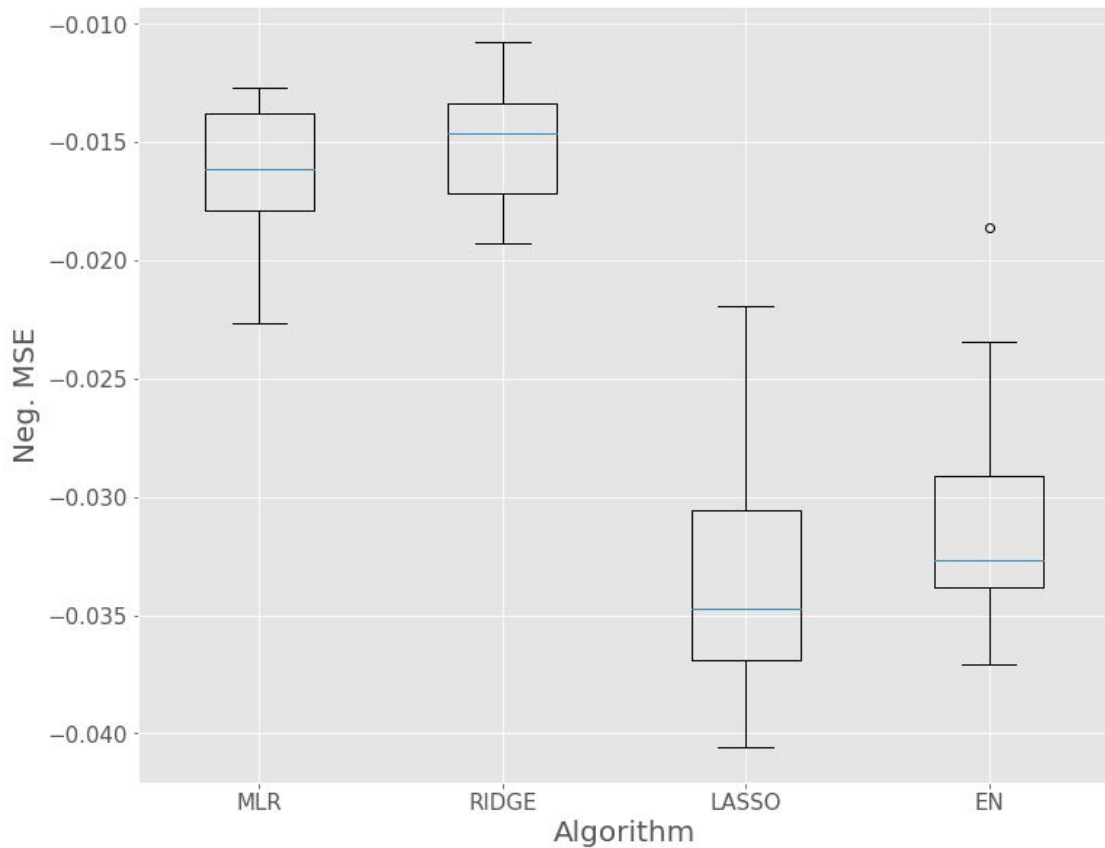
ENSEMBLES

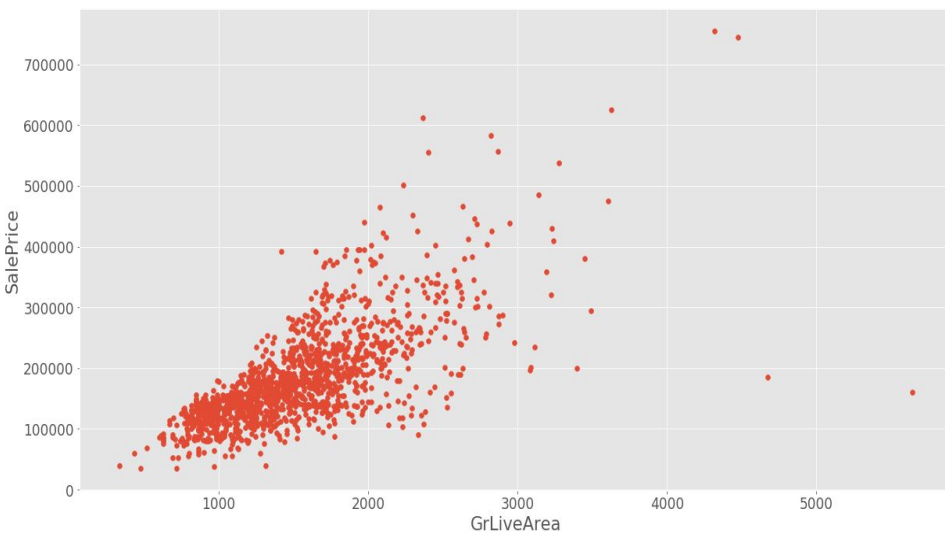
Exploring Powerful Algorithms

Random Forests, XGBoost

Baseline Linear Models

To get a first impression, we ran the linear regression algorithms on our data without making any adjustments





No Outlier Removal

Original Data

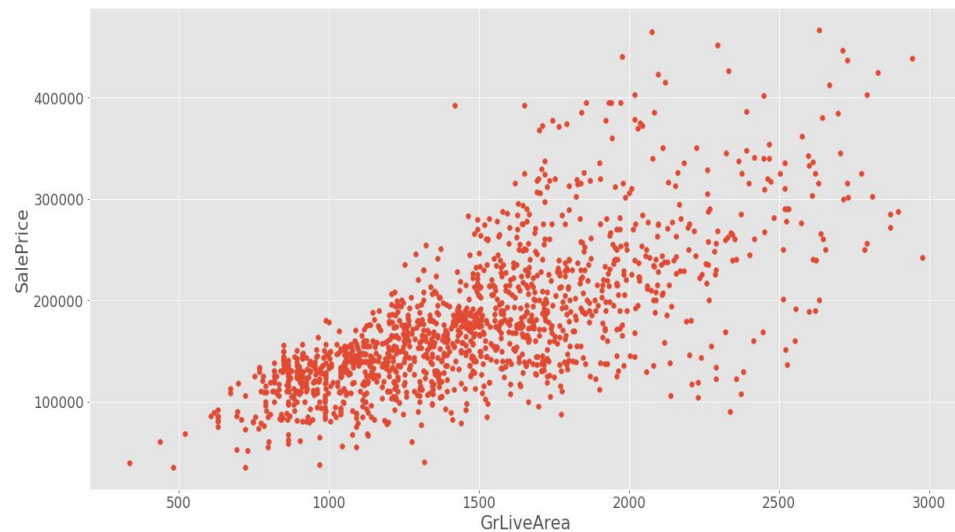
Some outliers that might skew our results

After Outlier Removal

Data After Outlier Removal

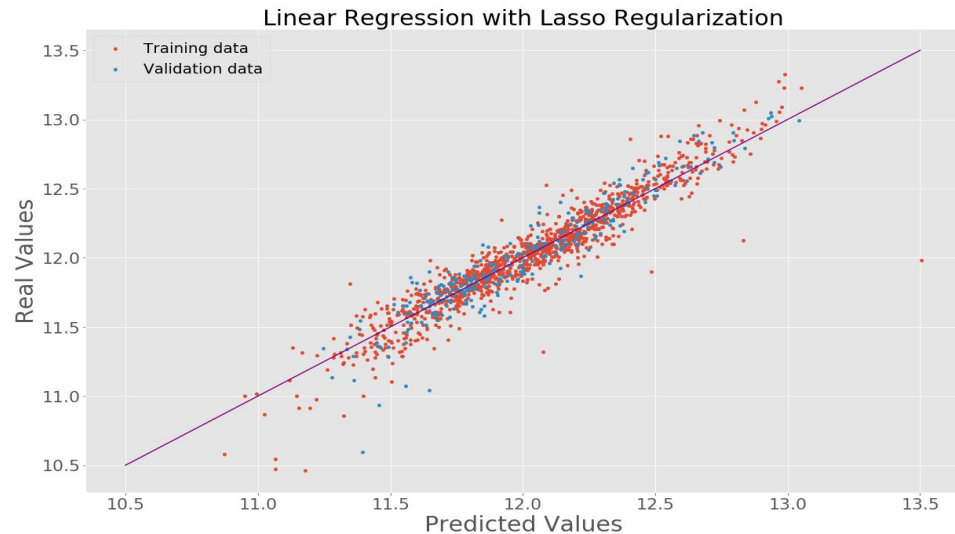
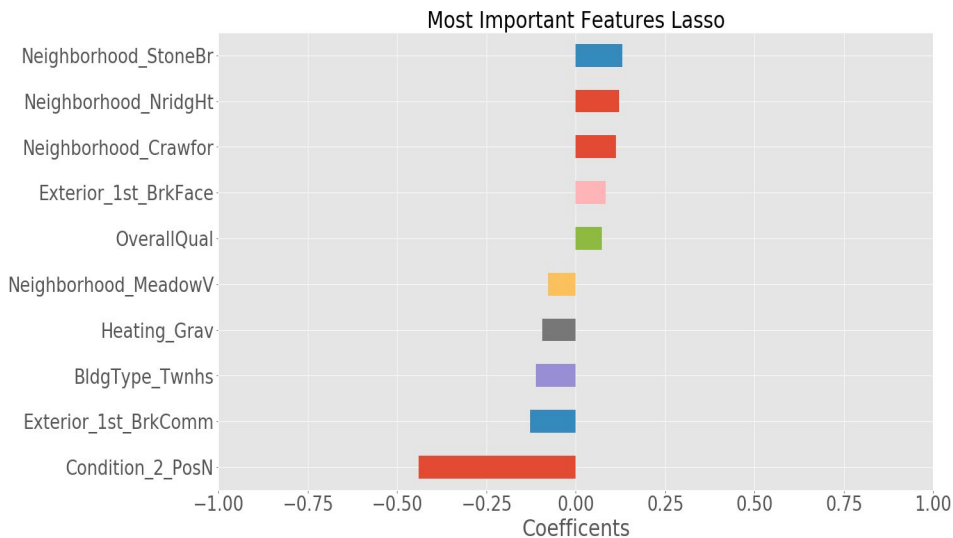
Removing all observations with GrLiveArea > 3k and

SalePrice > 500k



TRAIN VS. TEST PREDICTIONS LASSO

Except for a few outliers very accurate

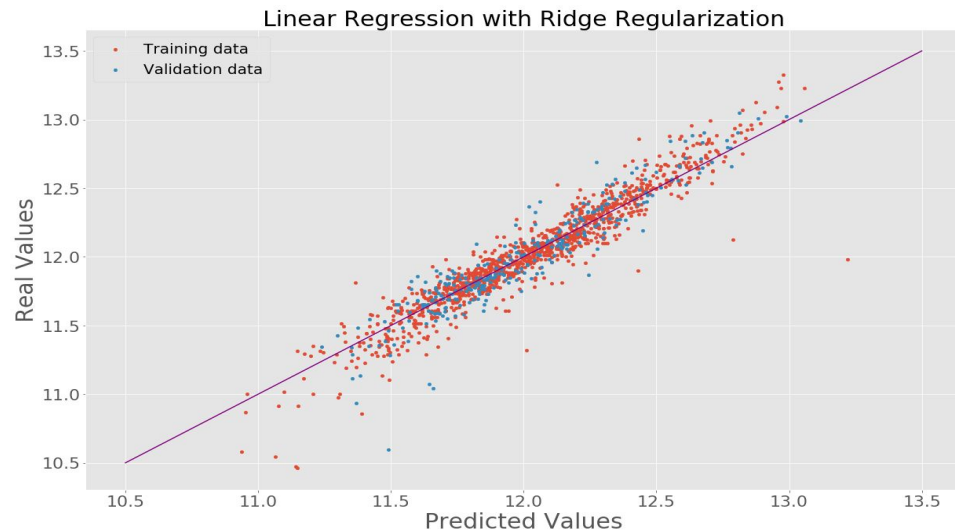
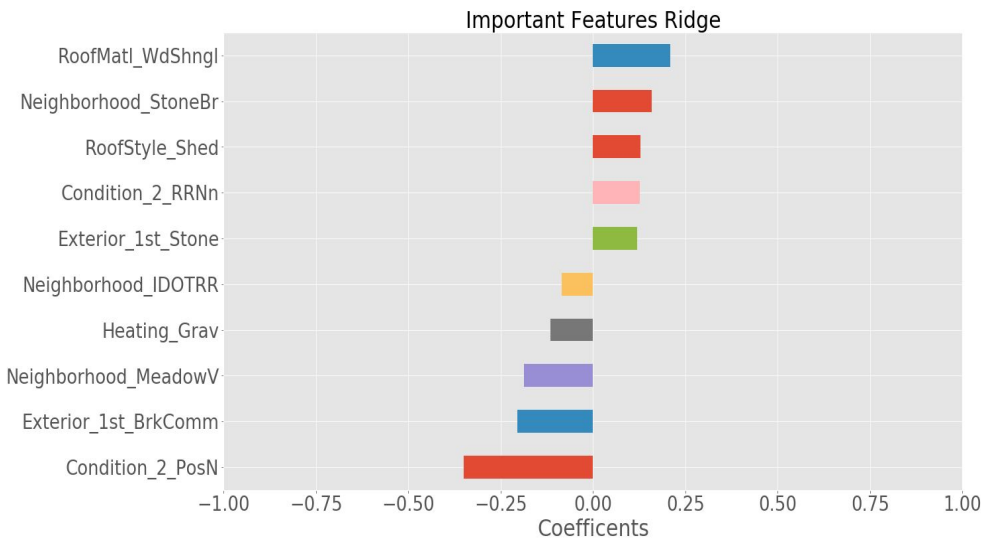


FEATURE IMPORTANCES

Neighborhood very important

TRAIN VS. TEST PREDICTIONS RIDGE

Also very accurate except for a few predictions



FEATURE IMPORTANCES

Neighborhoods still important, but not as important as
in Lasso

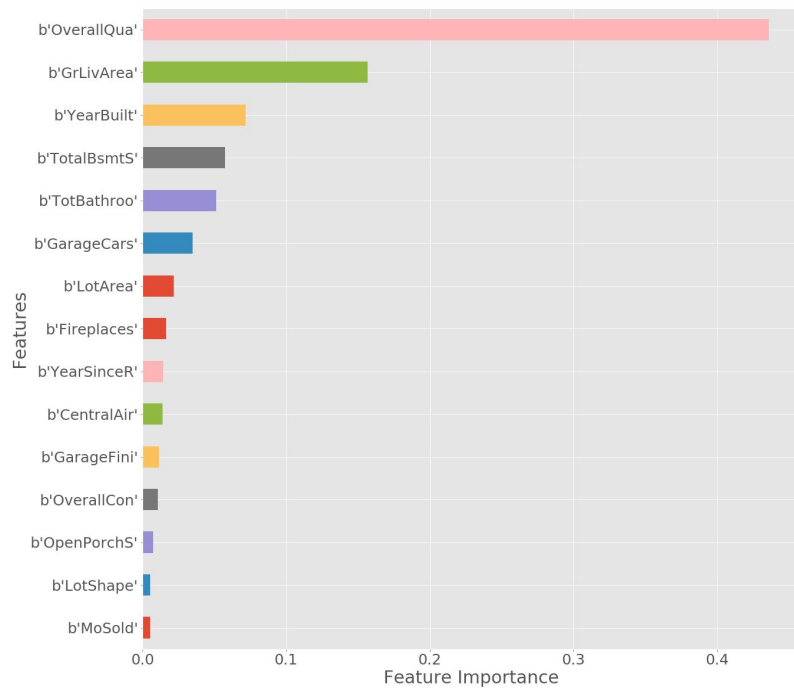
Results Linear Models

All of the results represent the RMSE (Root Mean Squared Error) on the test data set

MLR	LASSO	RIDGE	ELASTIC NET
0.180875	0.128068	0.138817	0.131943

RANDOM FOREST

One of the most powerful non-linear algorithms



TOP 3:

➤ OverallQuality

➤ GrLiveArea

➤ YearBuilt

Random Forest

Comparing the errors as we increased the number of trees

➤ **Errors do not improve significantly**

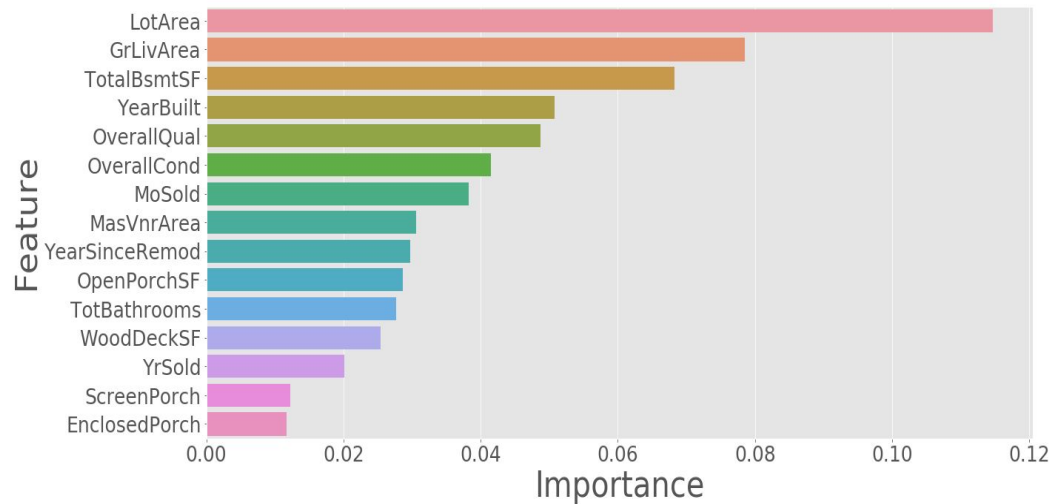
➤ **Worse performance than linear models**

➤ **RMSE: 0.137845**



XGBoost

Extreme Gradient Boosting



> **Different results from RF**

> **LotArea way more important**

> **GrLiveArea very important as well**

BEST RESULTS & KAGGLE SCORES

After tuning our models we submitted the predictions to Kaggle

	LASSO	RIDGE	RF	XGBOOST
Our RMSE	0.128068	0.138817	0.137845	0.125280
Kaggle Score	0.12406	0.13061	0.14962	0.13256

THANK YOU!

QUESTIONS?