NYCDSA Machine Learning Project: Ames Housing Dataset

The All-American Regex

Agenda

I. Introduction & Background

II. Data Pre-Processing

III. Linear Models

IV. Tree-Based Models

V. Stacked Models

VI. Conclusions & Next Steps

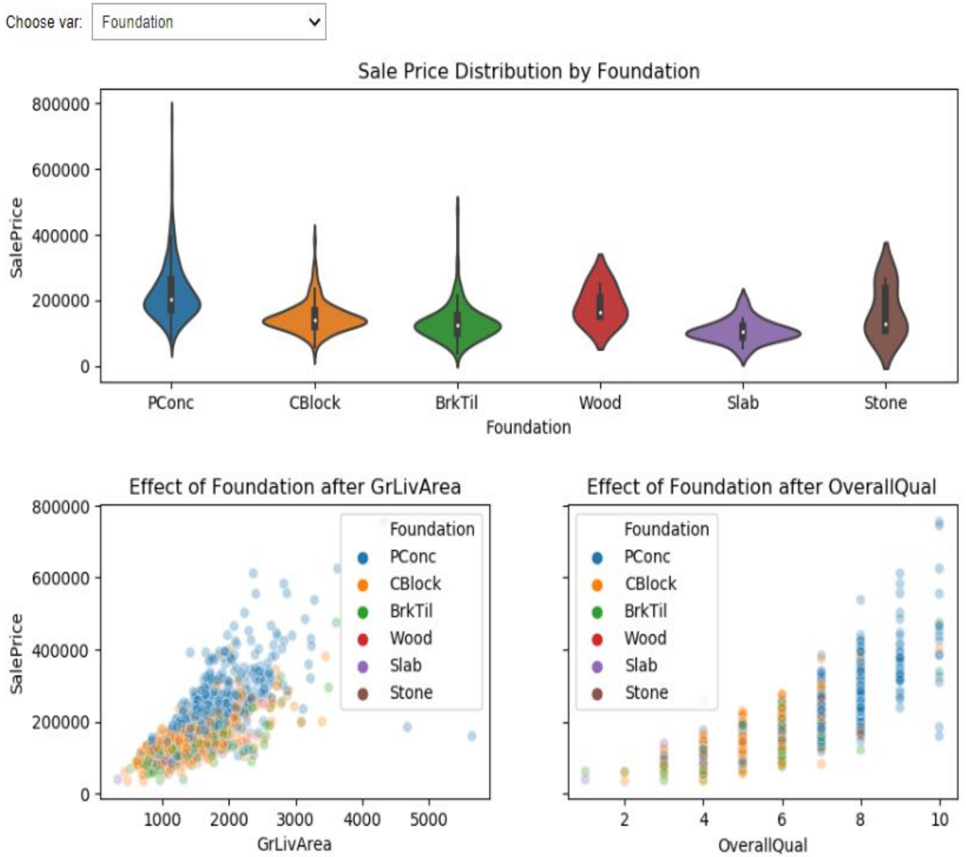
Background

- Housing sales in Ames, Iowa from 2006 to 2010
- 2,919 sales (1,460 in the training set)
- 80 features (23 nominal, 23 ordinal, 14 discrete, 20 continuous)
  - Size, quality, area, year, etc.
- Originally deployed by Dean De Cock in 2011 as an alternative to the Boston Housing Dataset (Harrison and Rubenfeld 1978)

Features & Some EDA

- ❑ Overall material and finish (OverallQual) and above ground square footage (GrLivArea) had the strongest linear relationships with the sale price

- ❑ We took a look at each feature with respect to the two above and brainstormed how we could feed them into a model



Pre-Processing: Exclusion and Imputation

- Some features appeared to be collinear with others or insignificant, so we decided to remove them from our data (eg. Utilities)
    - We ended up moving forward with 53 features

- Dealing with “NaN” values - Fill with “None”, 0, or the mode of the training set based on variable type
    - LotFrontage was filled with the neighborhood median of the training set

- Two outliers in the training set (>4,000SF, <$200k) were removed

Pre-Processing: Handling Features of Different Types

- One-hot encoding
  - Negligible minority classes were dropped

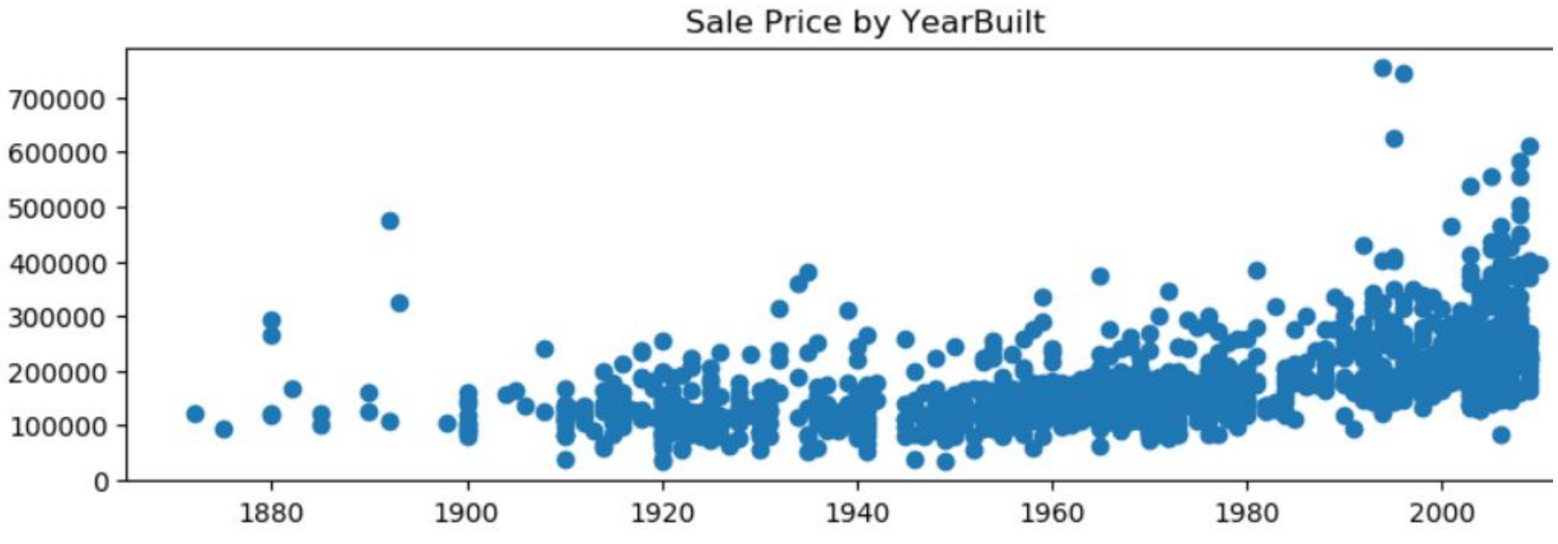
Foundation_BrkTil	Foundation_CBlock	Foundation_PConc
0	0	1
0	1	0
0	0	1
1	0	0
0	0	1
0	0	0
0	0	1
0	1	0

- Binary variables
  - eg. Number of fireplaces → Is there a fireplace?

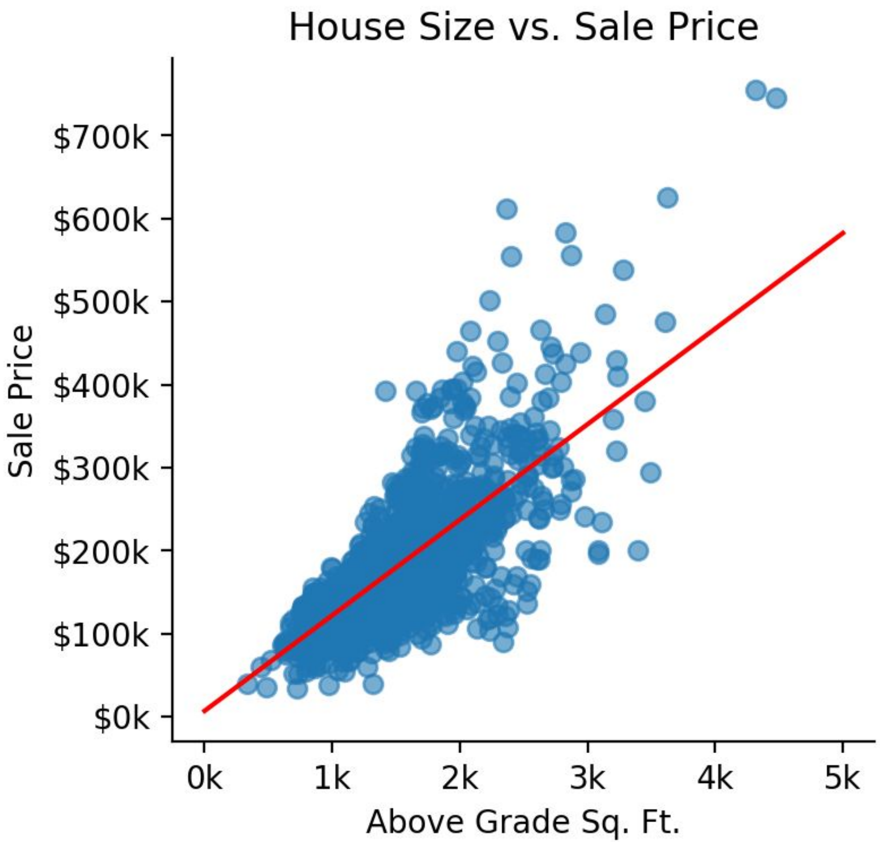
Fireplaces	PavedDrive
0	1
1	1
1	1
1	1
1	1
0	1
1	1
1	1
1	1

Pre-Processing: Special Cases

- Transformations
  - eg. LotArea → log(LotArea)
- Grouping
  - eg. YearBuilt before 1950 → 1950



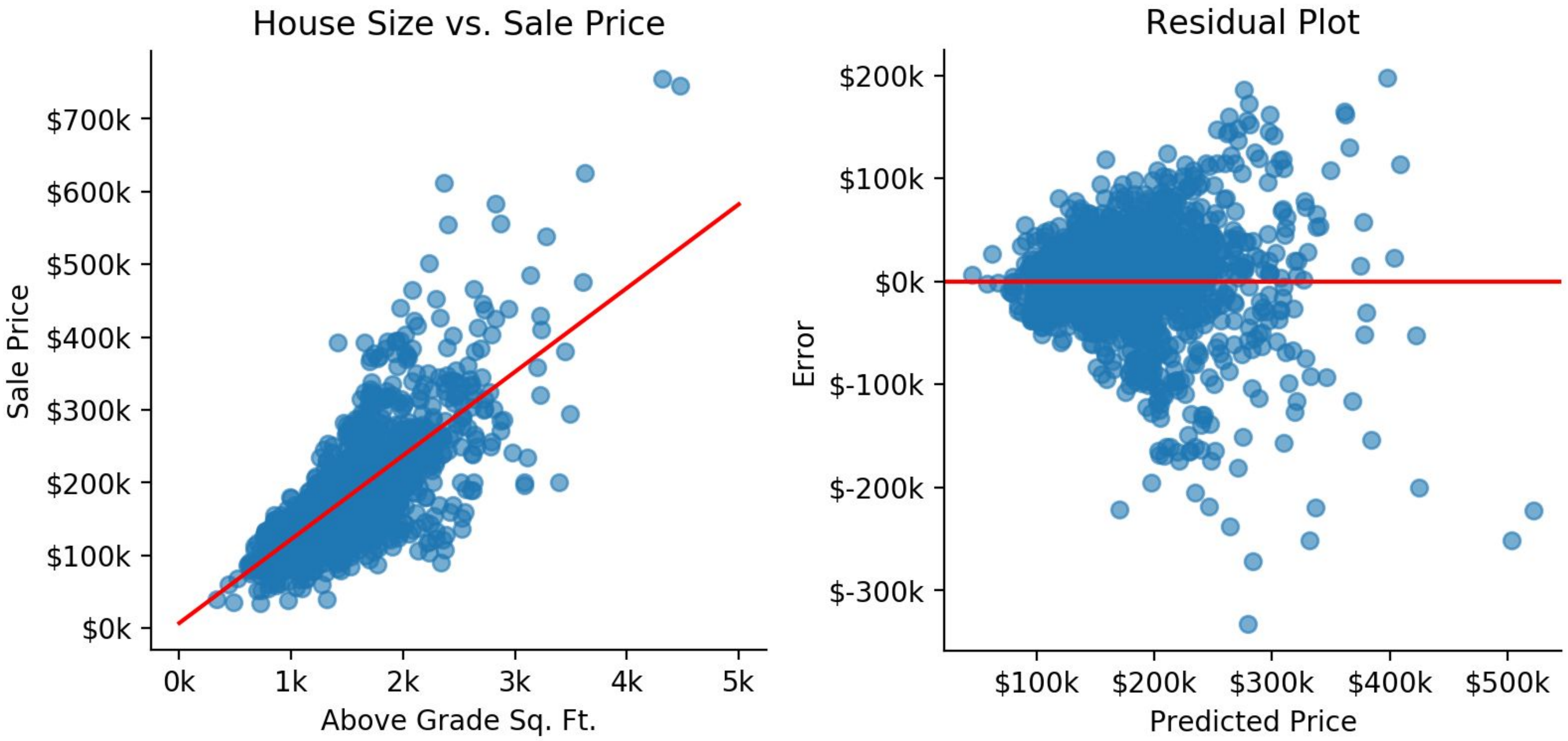
Simple Linear Regression



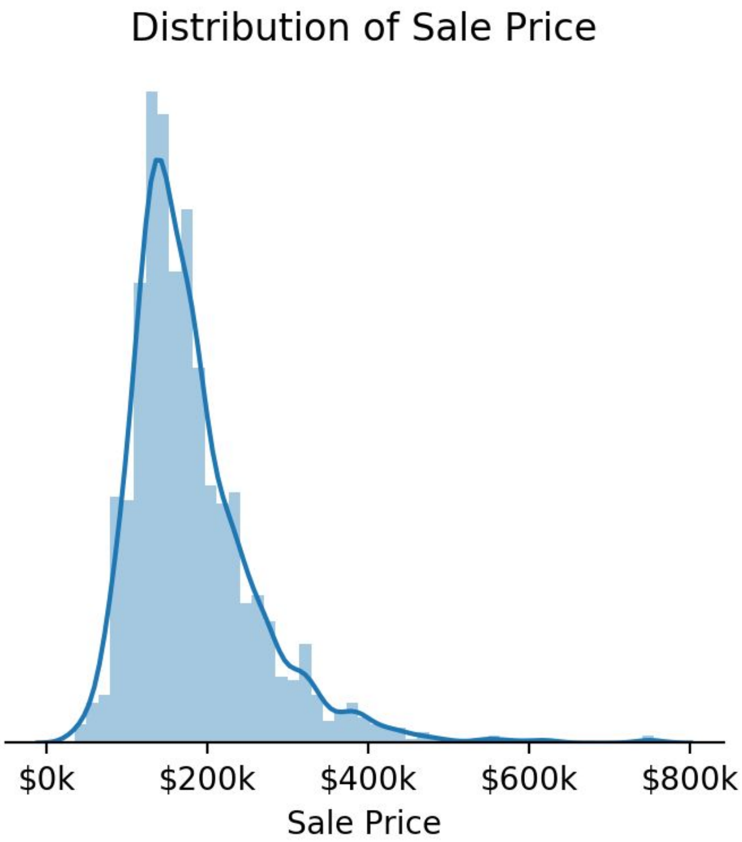
SalePrice = 7,165 + 115 * GrLivArea

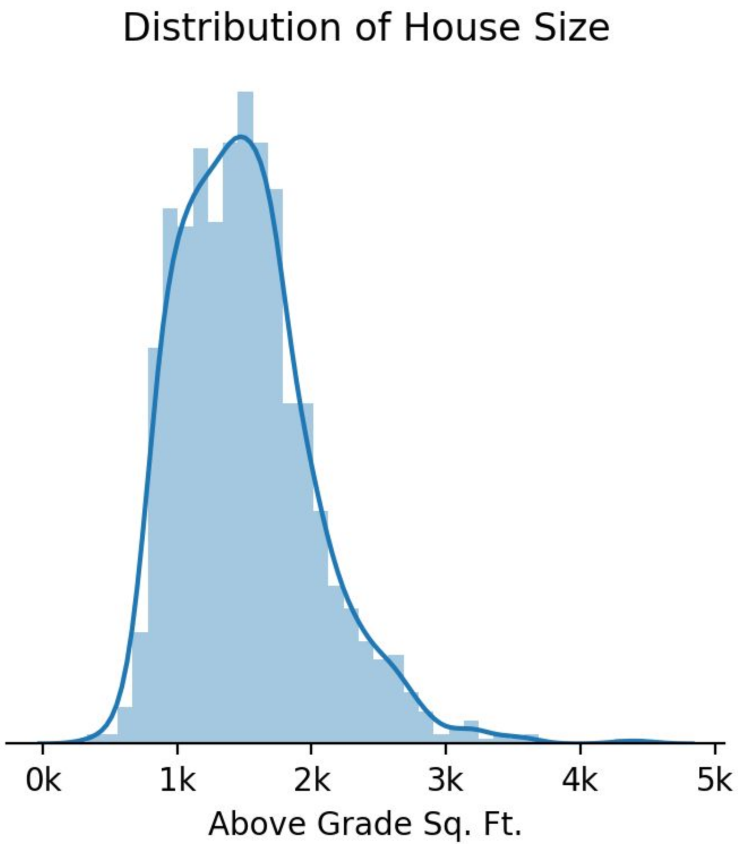
- R2 (Training) = 0.54
- RMSLE (Training) = 0.273
- RMSLE (CV) = 0.273

Simple Linear Regression: Residuals

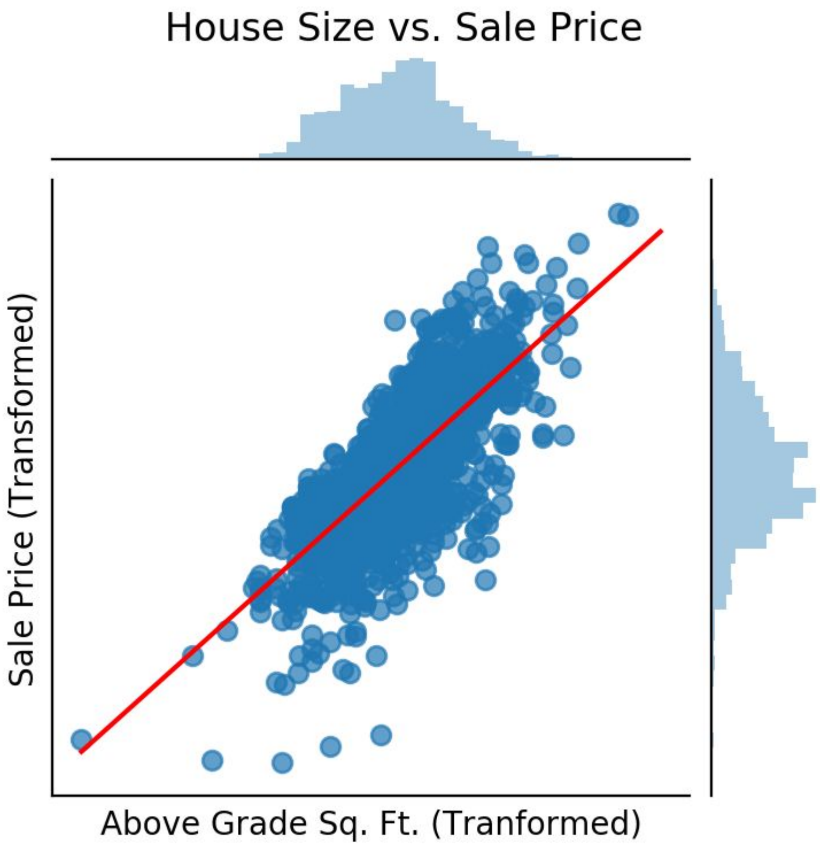


Simple Linear Regression: Variable Distributions





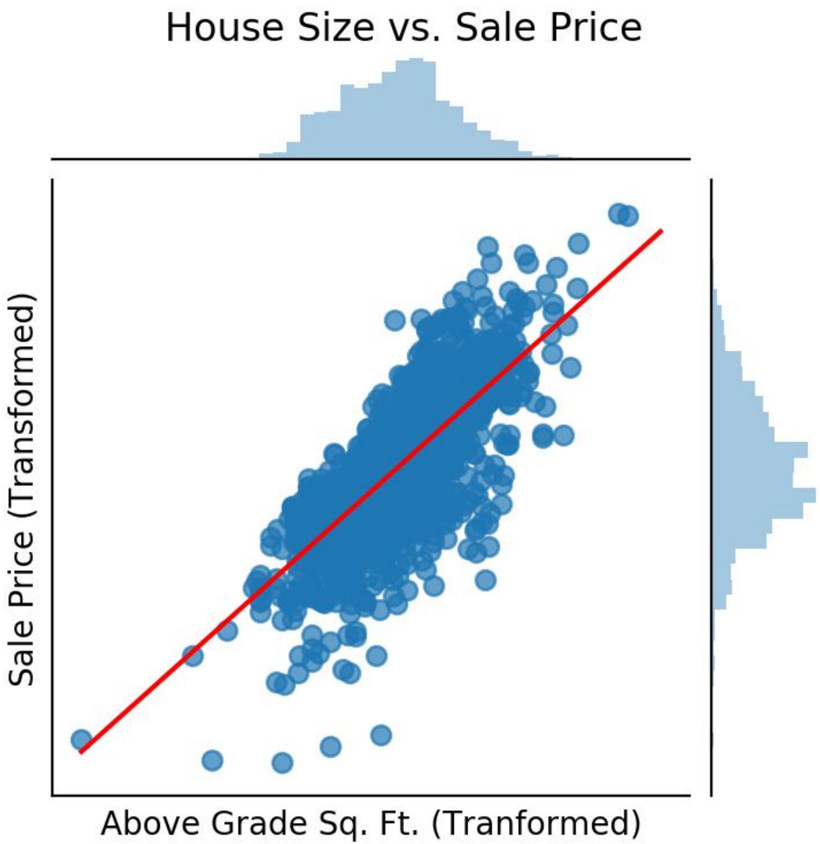
Simple Linear Regression: Box-Cox

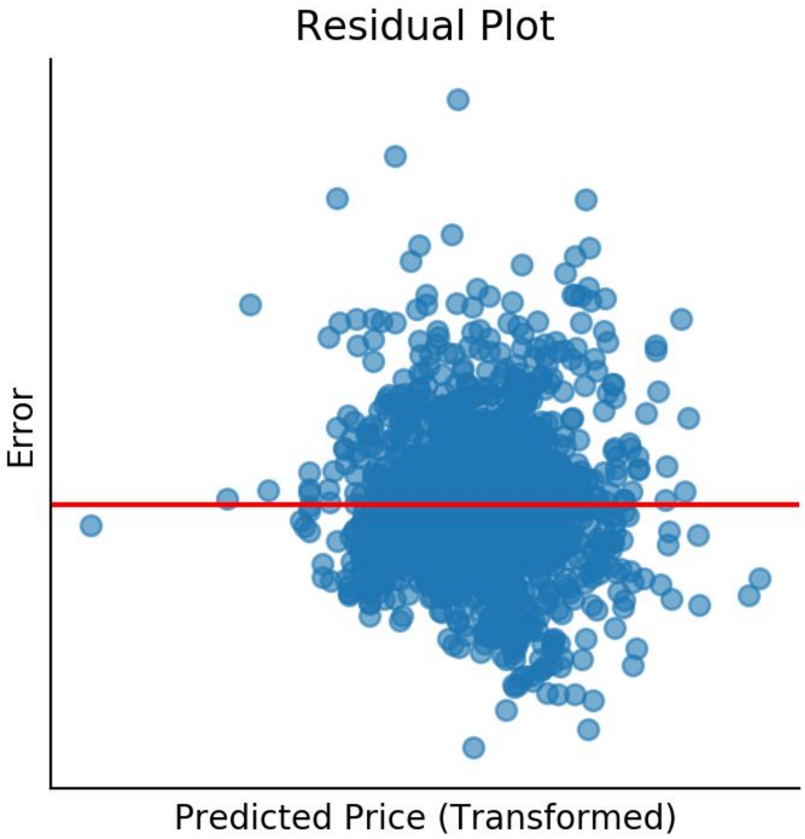


Transformed Model

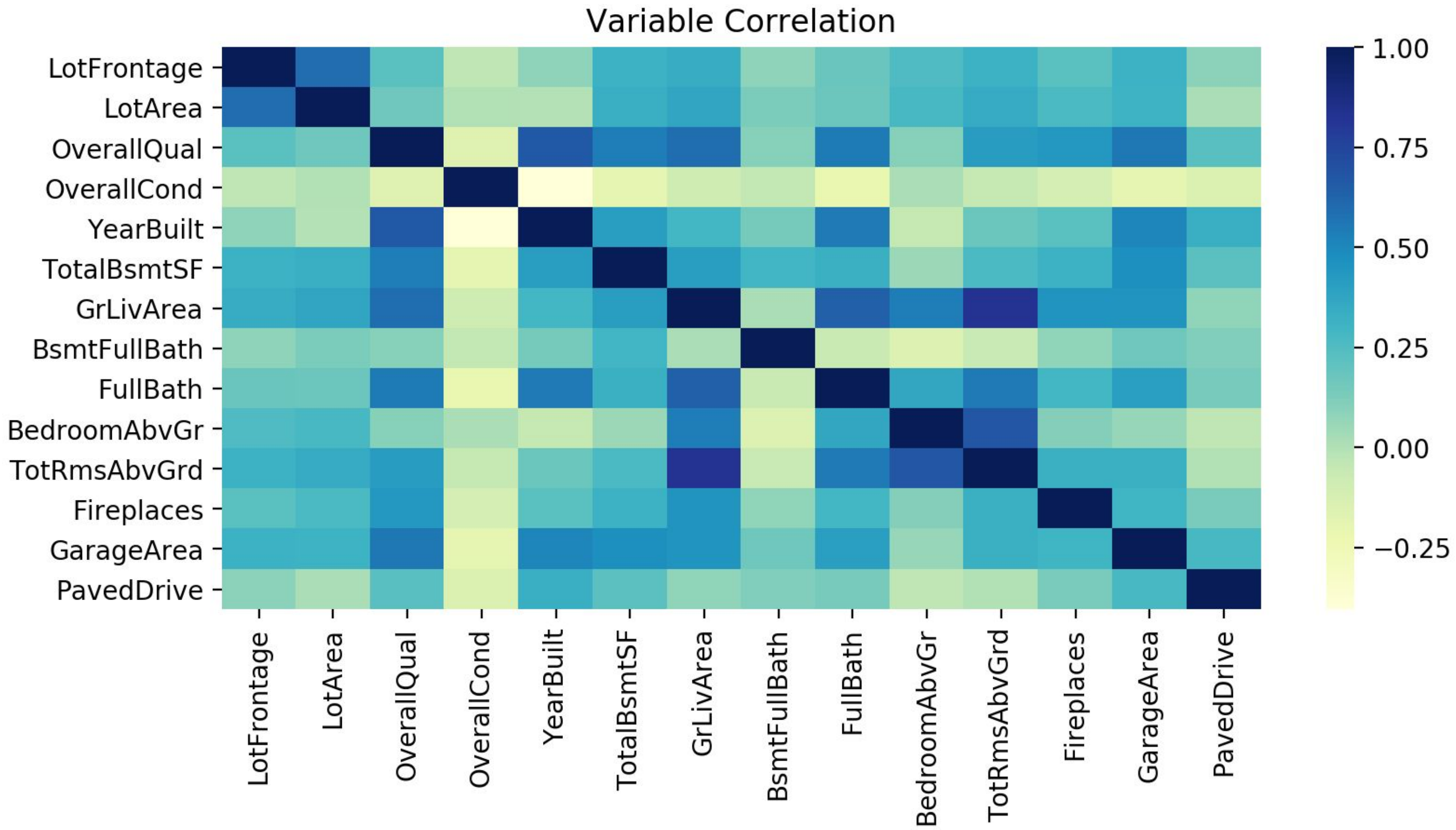
- R^2 (Training) = 0.54
- RMSLE (Training) = 0.269
- RMSLE (CV) = 0.270

Simple Linear Regression: Box-Cox





Multiple Linear Regression



Multiple Linear Regression

L2 Regularization

L1 Regularization

No Regularization

- R^2 (Training) = 0.61
- RMSLE (Training) = 0.250
- RMSLE (CV) = 0.251

L2 Regularization (Ridge)

- R^2 (Training) = 0.92
- RMSLE (Training) = 0.108
- RMSLE (CV) = 0.119

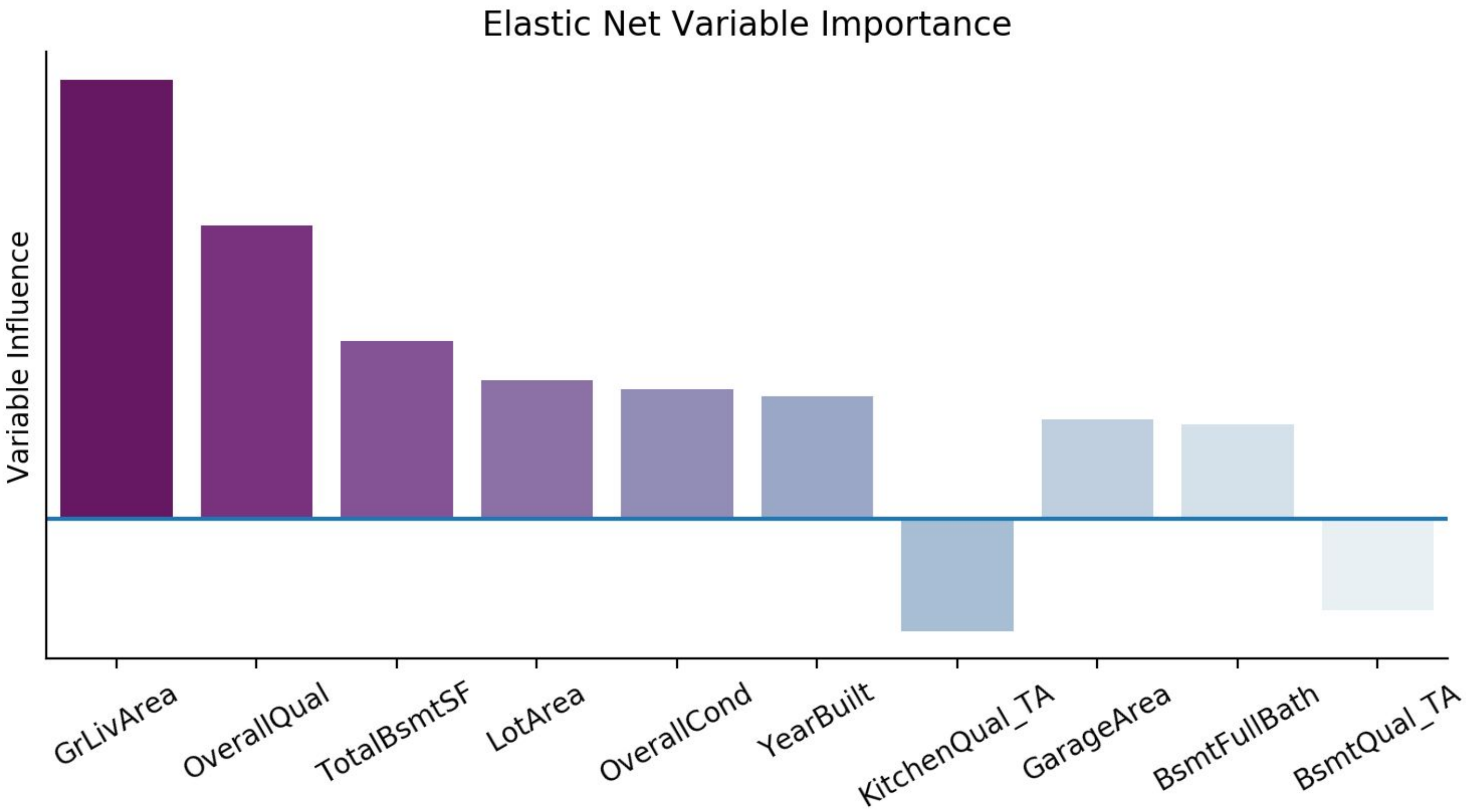
L1 Regularization (Lasso)

- R^2 (Training) = 0.92
- RMSLE (Training) = 0.109
- RMSLE (CV) = 0.119

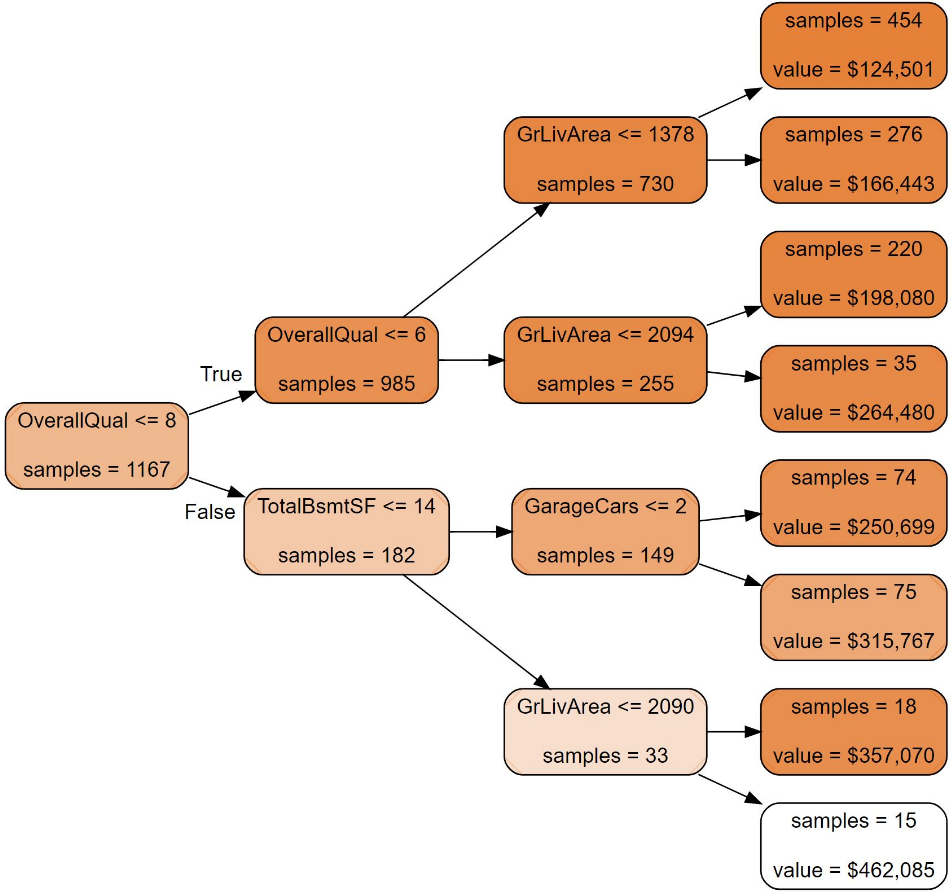
Combined (Elastic Net)

- R^2 (Training) = 0.92
- RMSLE (Training) = 0.111
- RMSLE (CV) = 0.118

Multiple Linear Regression



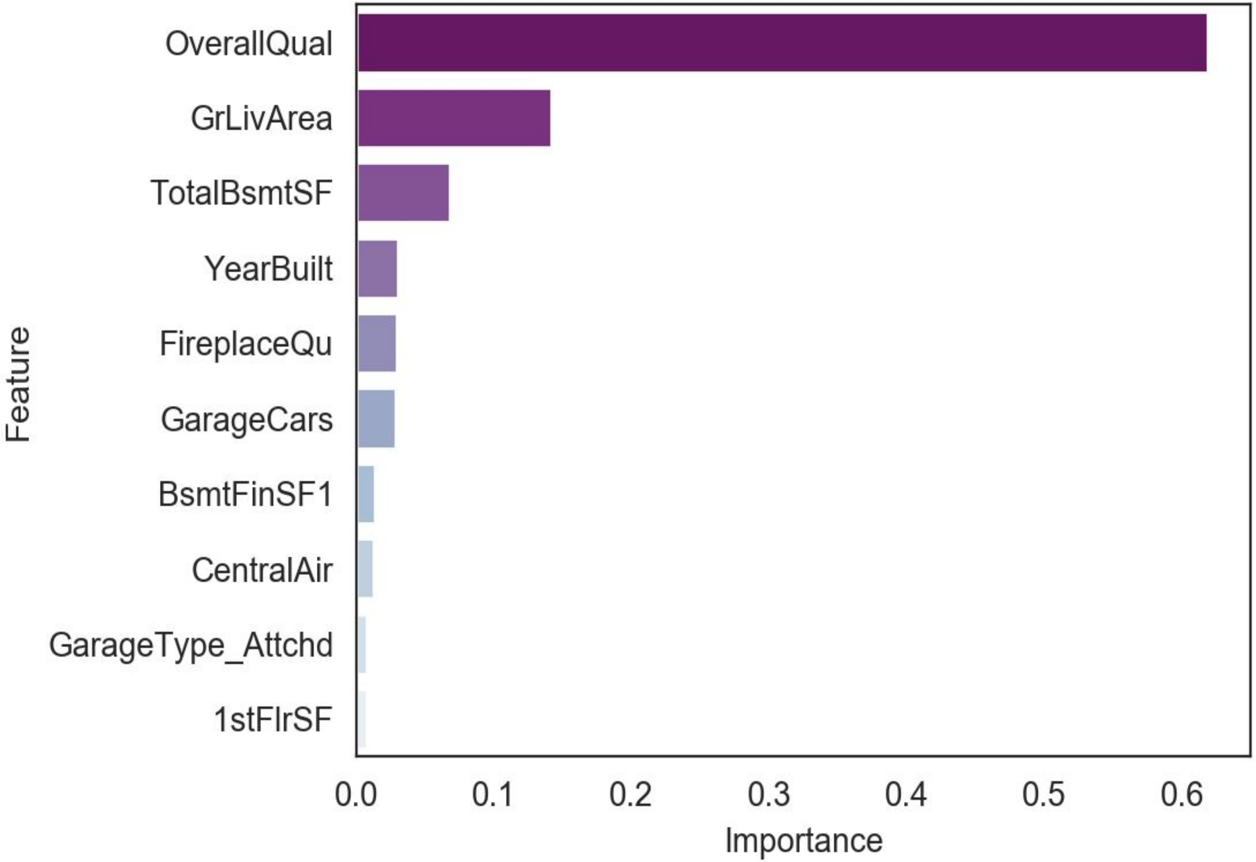
Tree-Based Models



Important Hyperparameters

- N_estimators
- Max_features
- Max_depth
- Min_samples_split
- Min_samples_leaf

Decision Tree

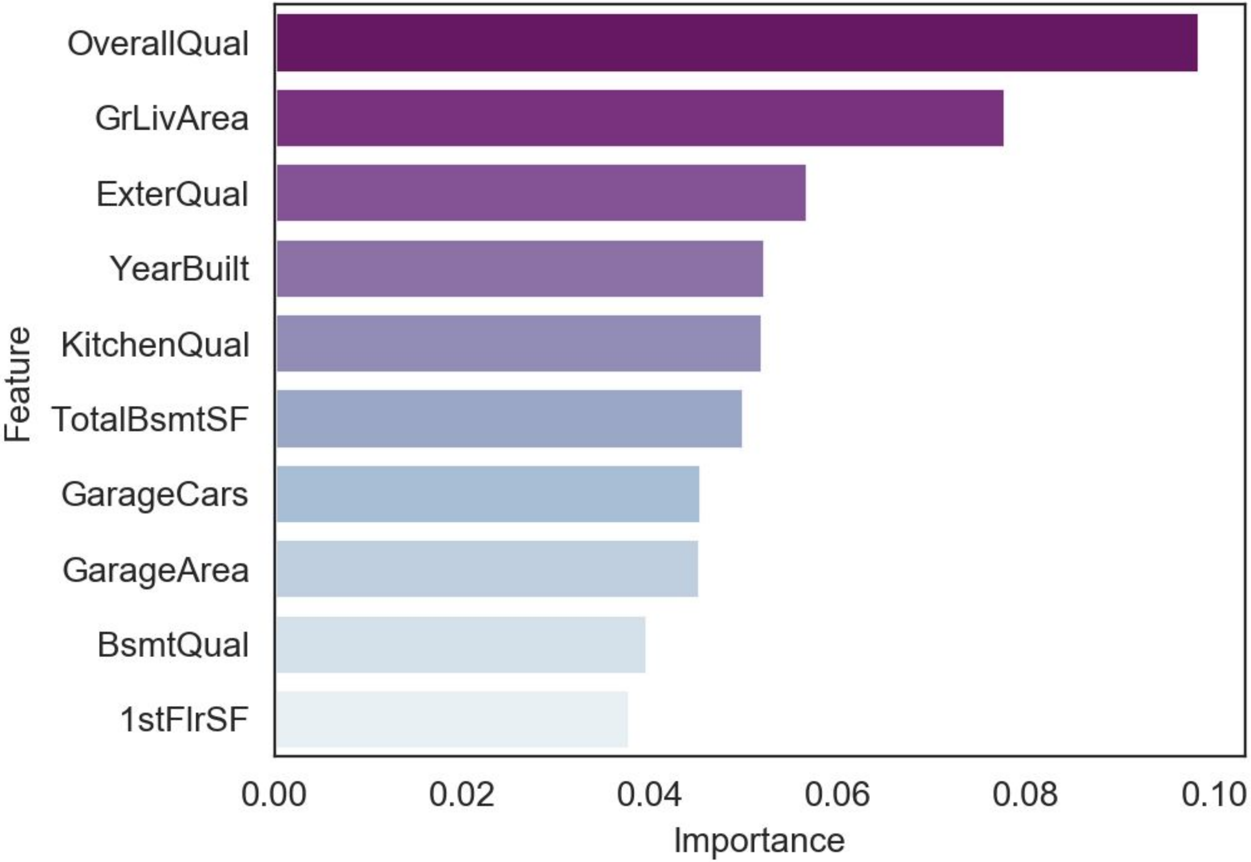


R2 (Training) = 0.87

RMSLE (Training) = 0.141

RMSLE (CV) = 0.184

Random Forest



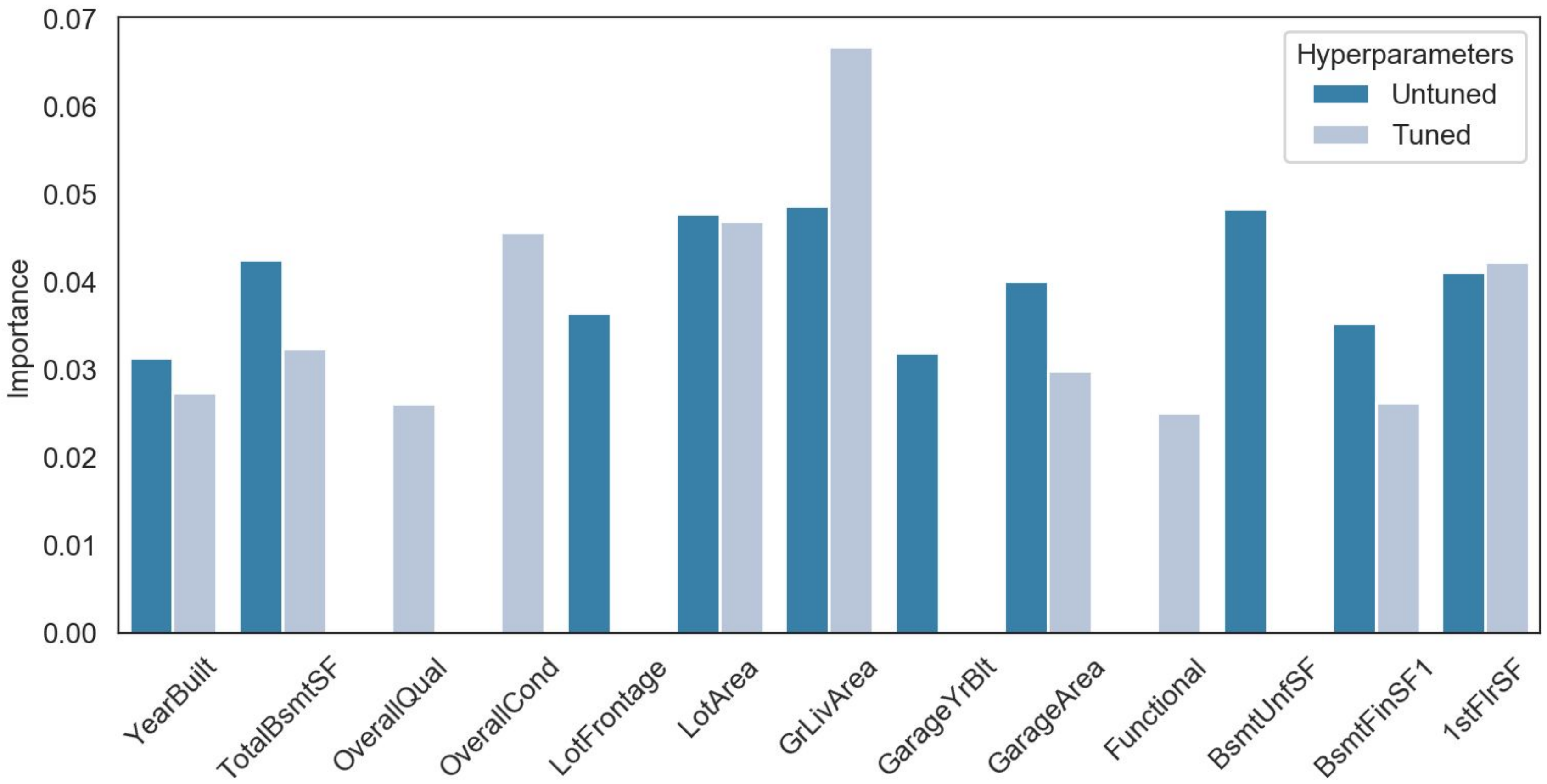
R2 (Training) = 0.85

RMSLE (Training) = 0.151

RMSLE (CV) = 0.165

Gradient Boosting

Gradient Boosting



Exploring Ensembling

- Intuition
  - Elastic Net (PL) 0.121
  - Tuned Gradient Boost (PL) 0.122
  - Tuned Random Forest (PL) 0.145
- Averaging
  - Equal weight to top 3 performers (PL) 0.123
  - Drop weakest link (PL) **0.118**

Stacking

- Stacking
  - Elastic Net, Gradient Boost, Random Forest base learners
    - Linear regression meta-model (PL) **0.117**
  - Elastic Net, Gradient Boost
    - Linear regression meta-model (PL) **0.117**

Conclusions / Next Steps

- Importance of Feature Engineering
  - EDA → Feature Engineering

- Public vs. Private Leaderboard

- Future Work
  - Explore applicability and effectiveness of PCA and MCA
  - Ensembling
    - Stacking architecture (size of stack, strategic tuning parameters)
    - Different high-level learners