Regression Project:
Ames Housing Pricing Predictions

January 2020

Problem Statement & Key Objectives

Context:

Kaggle competition with the objective of developing a model to predict housing prices in Ames Iowa based on historical data. Over 100+ DSI students across the nation participated. This model achieved a ranking within the top 15 based on RMSE and is valid for price predictions

Key Objectives:

- ***Create model to predict housing prices based on historical data***
- ***Evaluate model***
- ***Disseminate techniques and methods to technical audience and colleagues***

Executive Summary

- Key objectives: Create model to predict housing prices, evaluate model, disseminate techniques and methods to technical audience
- Normal Distribution is an important factor for machine learning algorithms. Addressing outliers and applying natural log can assist in normalizing and thereby improving prediction performance
- In addition, Understanding Variable Correlation and Feature Engineering can elevate model prediction performance
- Seasonality affected Ames sales volume but Average Sale Prices remained fairly consistent
- For the Ames Housing Dataset Ranking Neighborhoods reduced multicollinearity, improved SalePrice predictions and directly drove a 3000+ reduction in Kaggle RSME score
- Visualizing and identifying high mean variance within a variable can indicate degree of influence on Target (Sale Price)
- The model suggests that Gr Liv Area, Overall Quality, Age, Neighborhood Rank, Bathrooms, Total Sq Ft, Is_New Were Leading Features With Impactful Coefficients
- The value of the model is that it is accurate and can facilitate real estate agency decision making and housing price prediction
- Key recommendation: trial first, then place model into production in order to drive operational/strategic real estate agency decision making. Over time the model will continue to learn and increasingly become more accurate

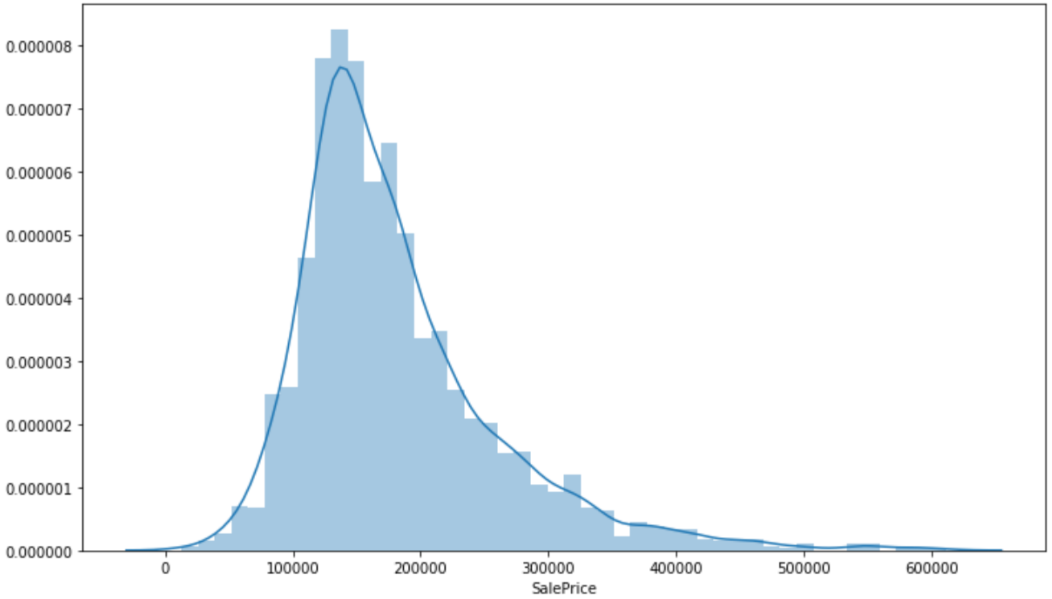
Process Overview

- Clean, Impute, Process raw data to achieve a suitable format for modeling
- Remove Outliers
- EDA
- Numerical Data to Categorical
- Feature Engineering
- Natural Log
- Modeling
- Model Evaluation & Optimization
- Interpretation

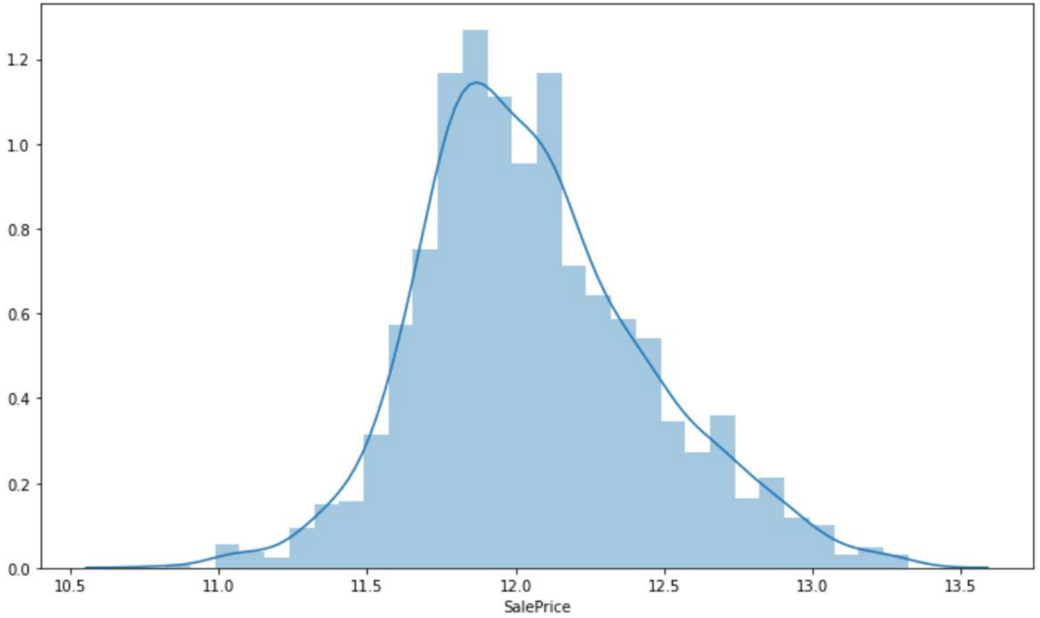
Training Sales Data Is Not Originally Normally Distributed. Natural Log and Addressing Outliers Resolves This

Training Data Sales Price by Modeling Stage

Original Dataset (Right-skewed)



Outliers Resolved & Natural Log Applied (Closer to Normal Distribution)

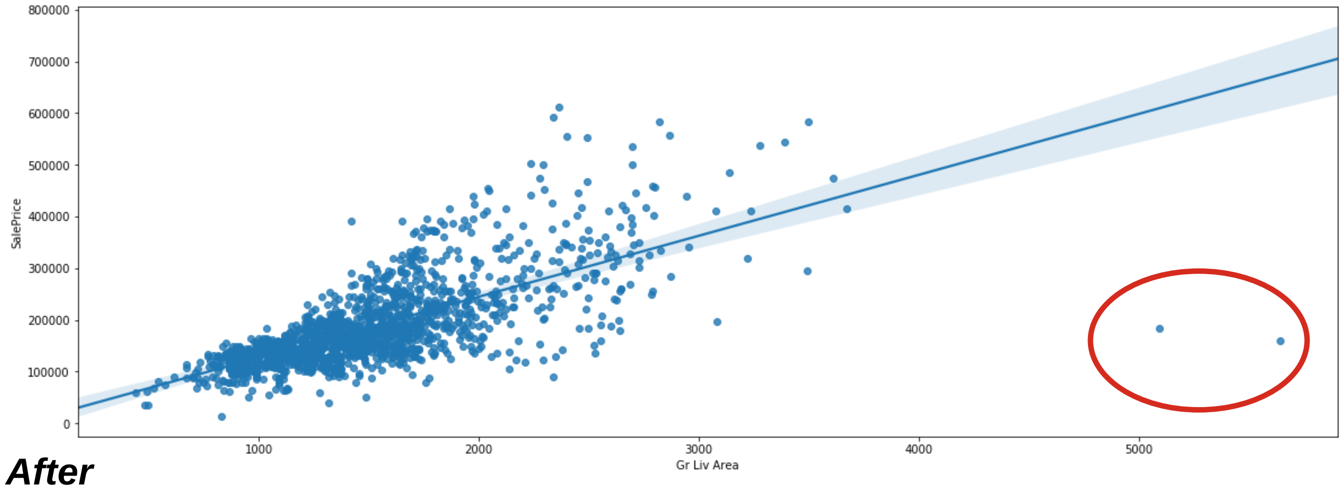


Source: Ames Housing Data

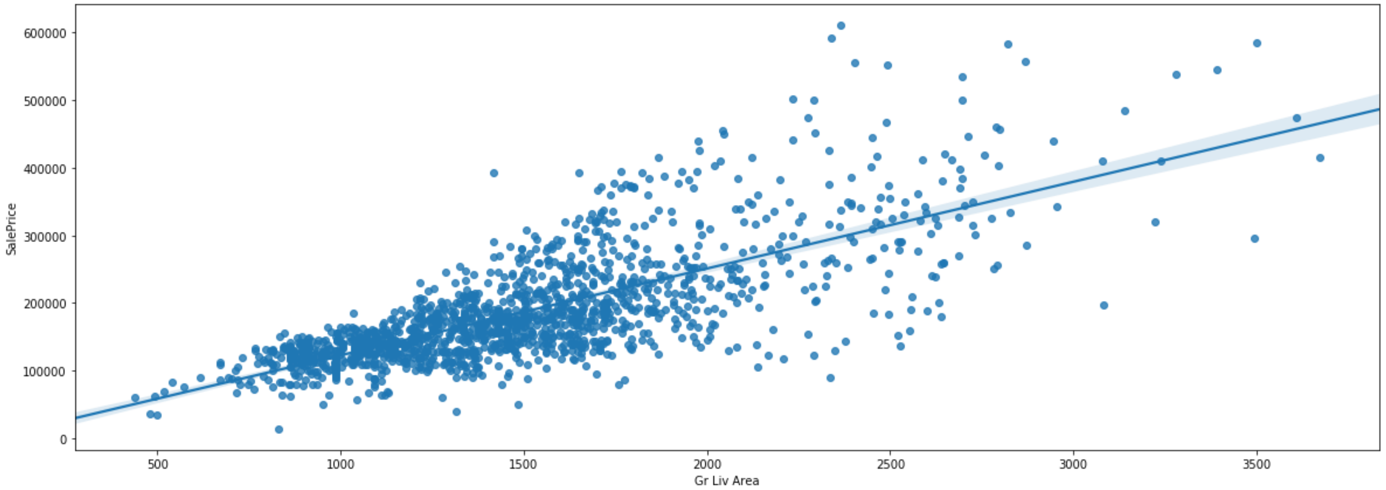
Outliers And Missing Data Were Identified And Addressed

Before

SalePrice by Gr Liv Area Sq Ft



After



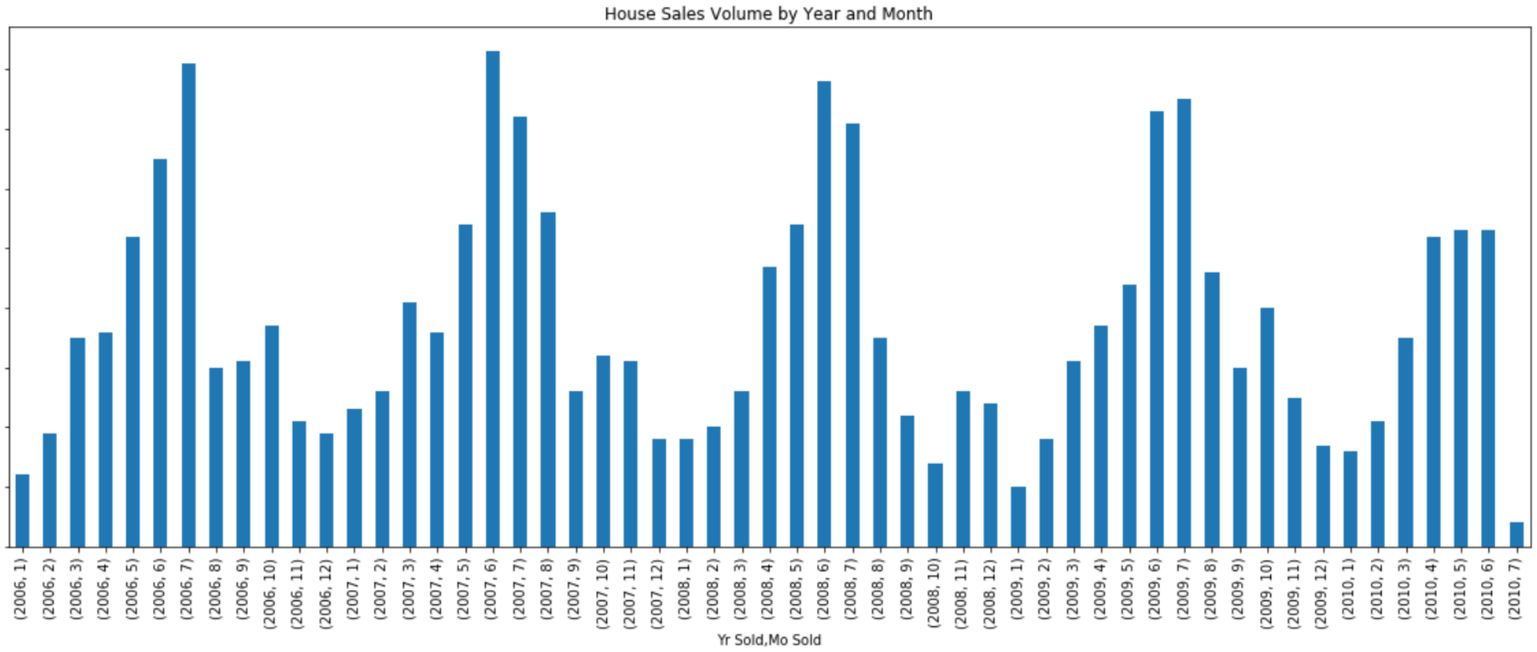
Outliers that contributed to score improvements included:

- Exorbitant Gr Liv Area Sq Ft and low SalePrice
- Data fields of Pool QC, Misc Feature, Alley, Fence, Fireplace Qu had a very high percentage of missing data and therefore were dropped from the model
- Imputing such a high volume of missing data would only mislead model

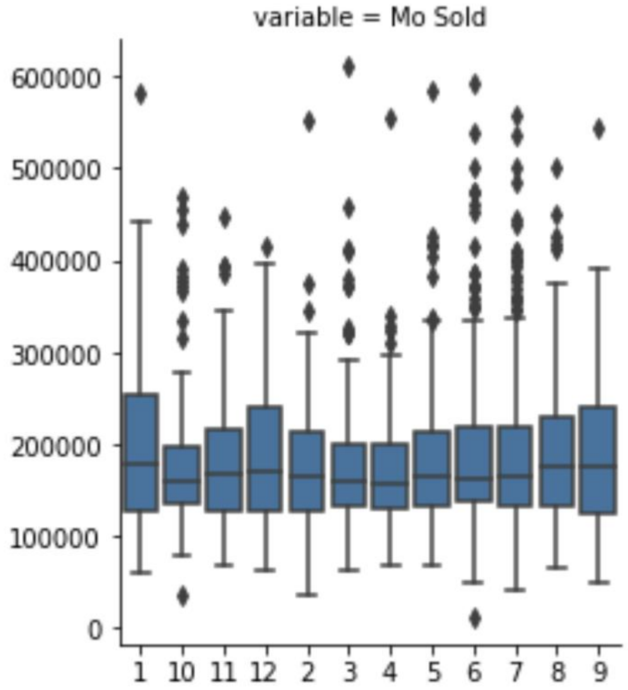
Source: Ames Housing Data

Seasonality Affected Sales Volume But Average SalesPrice Remained Fairly Consistent

Monthly Housing Sales Volumes



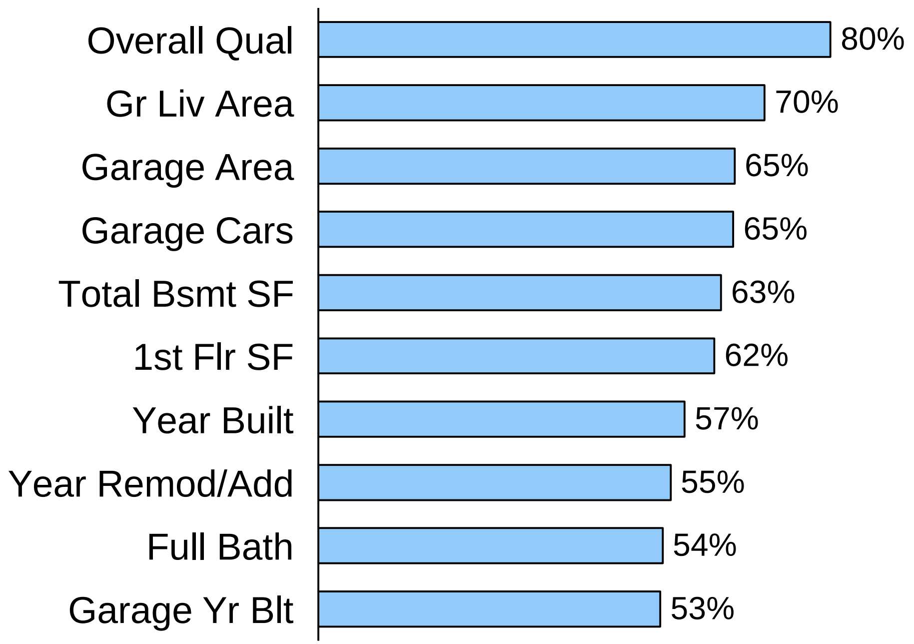
Sales Price by Month Sold



Source: Ames Housing Data

Understanding Variable Correlation And Feature Engineering Elevated Model Prediction Performance

Top 10 Feature Correlation to SalePrice



Feature Engineering

- **total_sq_ft** – was engineered, however it is important to note that ANSI and Fannie Mae dot not allow basement square footage to be included in total sq ft calculations therefore I excluded it
- **total_bathrooms** – variable was also engineered summing all full and half bathroom fields

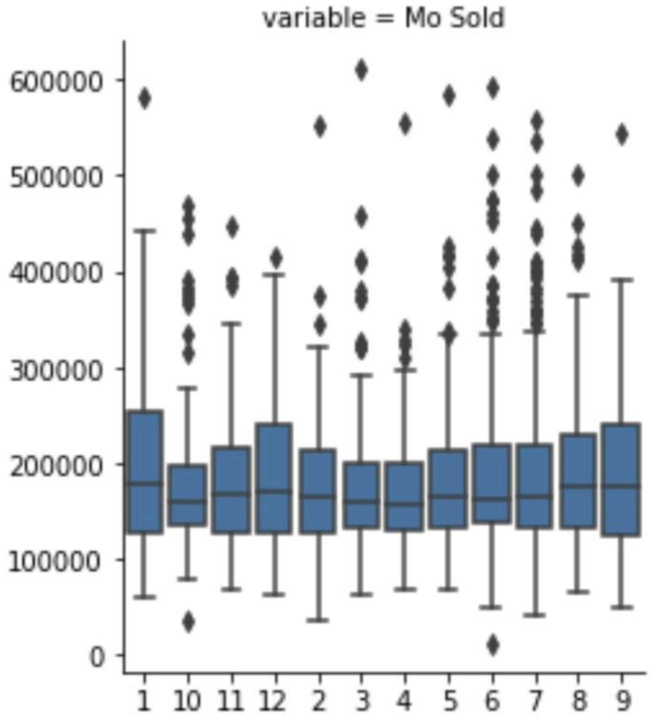
```python
# Let's create a few addl columns for features

train2['has_basement'] = train2['Total Bsmt SF'].apply(lambda x: 1 if x > 0 else 0)
train2['has_garage'] = train2['Garage Area'].apply(lambda x: 1 if x > 0 else 0)
train2['has_pool'] = train2['Pool Area'].apply(lambda x: 1 if x > 0 else 0)
train2['was_remodeled'] = (train2['Year Remod/Add'] != train2['Year Built']).astype(np.int64)
train2['is_new'] = (train2['Year Built']>=1996).astype(np.int64)
```

Source: Ames Housing Data

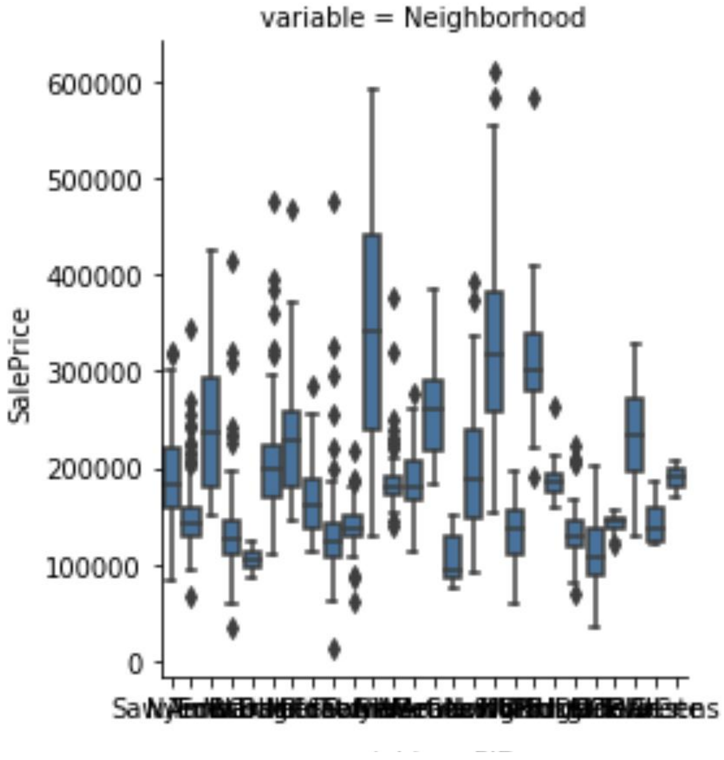
High Mean Variance Within Variable Indicate High Influence On Sales Price

Visual Examples of Low & High Influence Variables On SalePrice

<u>Low Mean Variance</u> suggests Low Influence On SalePrice

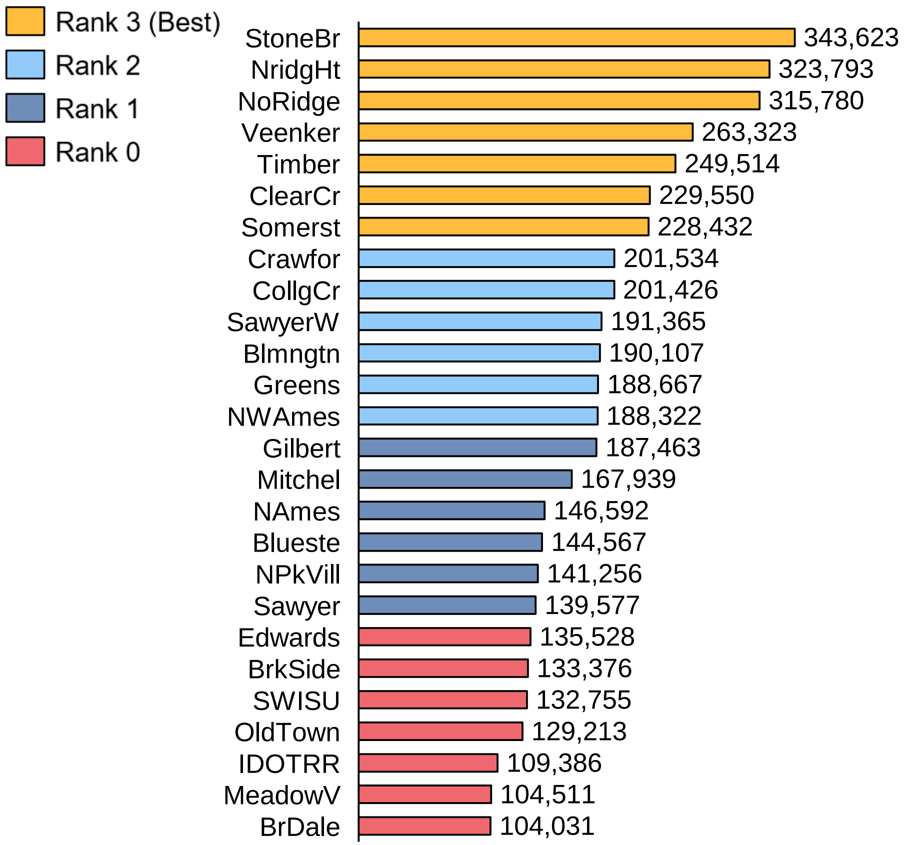


<u>High Mean Variance</u> suggests High Influence On SalePrice



Neighborhood Feature Engineering Reduced Multicollinearity And Improved Prediction Performance

Ames Neighborhoods by Average Sale Price



Neighborhood Ranking Dictionary Was Mapped To A New Column Then Dummified

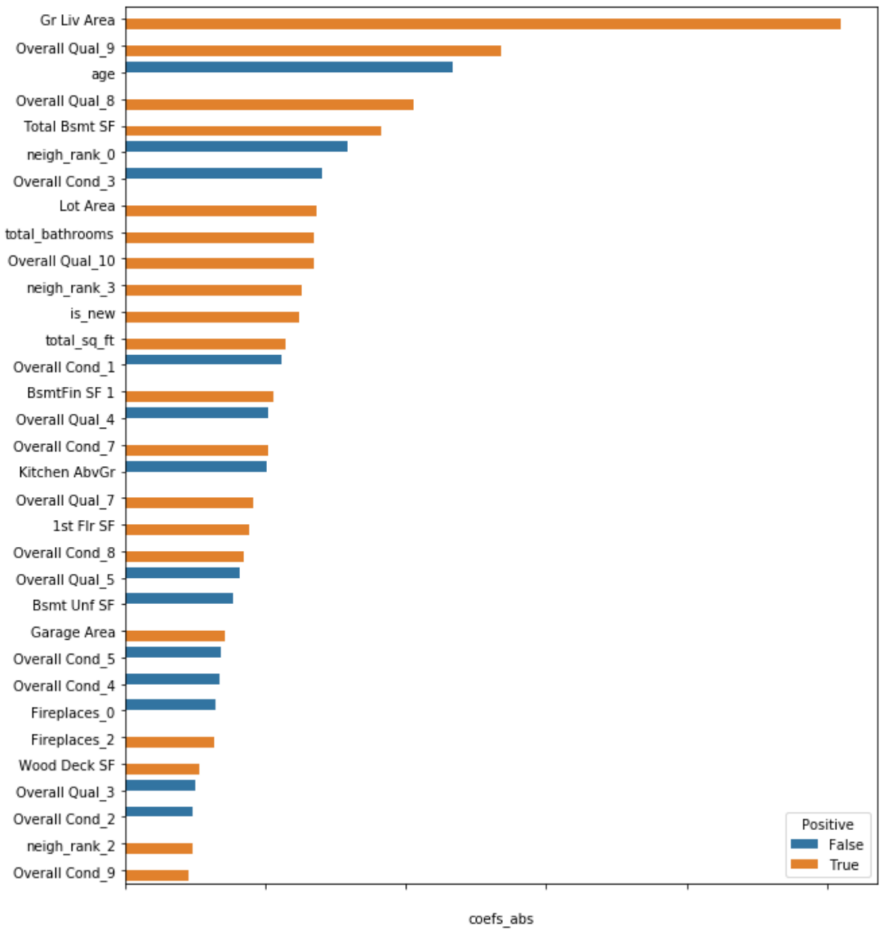
```
neigh_dict = {'StoneBr':'3',
              'NridgHt':'3',
              'NoRidge':'3',
              'Veenker':'3',
              'Timber':'3',
              'ClearCr':'3',
              'Somerst':'3',

             'Crawfor':'2',
              'CollgCr':'2',
              'SawyerW':'2',
              'Blmngtn':'2',
              'Greens':'2',
              'NWAmes':'2',

              'Gilbert':'1',
              'Mitchel':'1',
              'NAmes':'1',
              'Blueste':'1',
              'NPkVill':'1',
              'Sawyer':'1',

              'Edwards':'0',
              'BrkSide':'0',
              'SWISU':'0',
              'OldTown':'0',
              'IDOTRR':'0',
              'MeadowV':'0',
              'BrDale':'0'}
```

3000 pt Reduction In RMSE

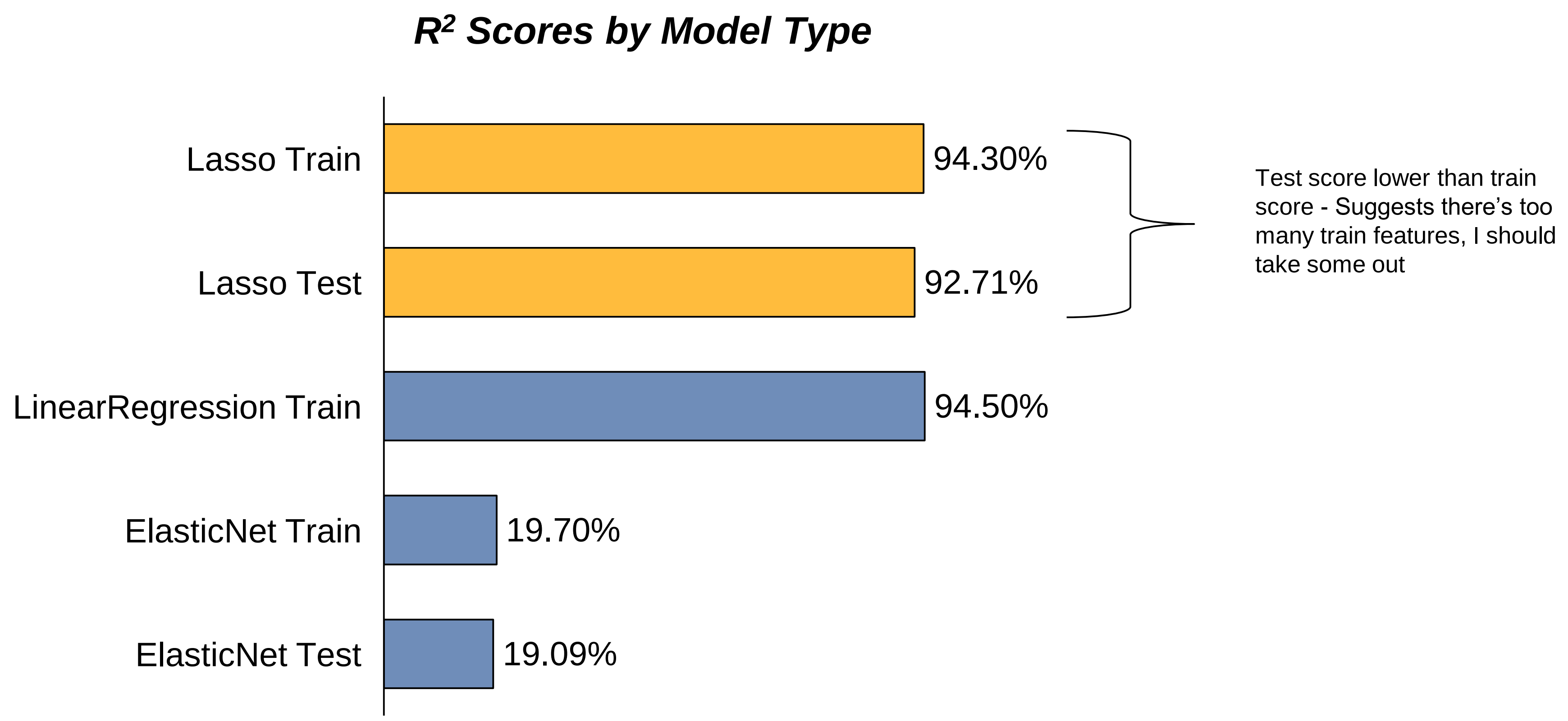
Source: Ames Housing Data

Gr Liv Area, Overall Quality, Age, Neighborhood Rank, Bathrooms, Total Sq Ft, Is_New Were Leading Features With Impactful Coefficients

Leading Features by Coefficient Lasso Ranking

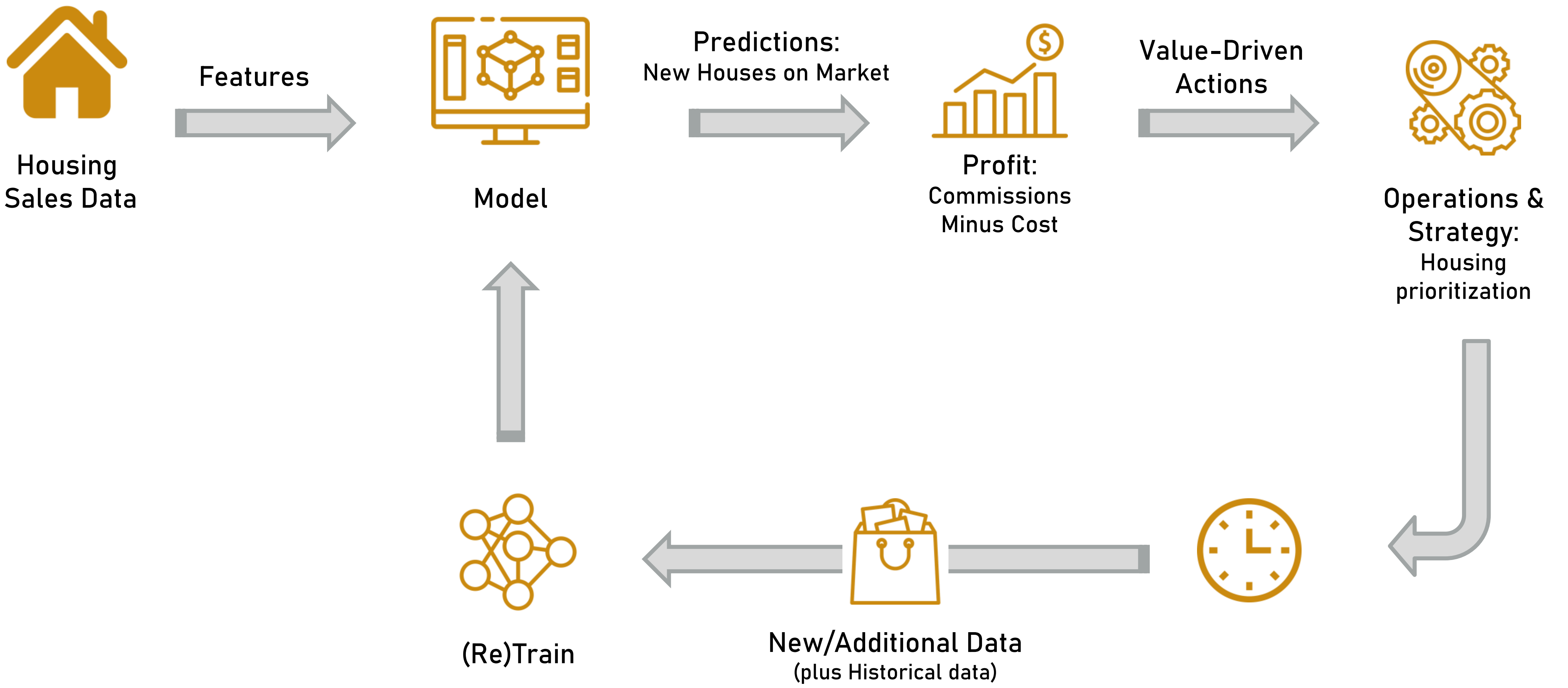


Model Will Enable Accurate Pricing Estimates, And Could Facilitate Decision Making For Real Estate Agencies



Source: Ames Housing Data

Key Recommendation: Trial First, Then Place Into Production In Order To Drive Operational/Strategic Real Estate Decision Making



Source: flaticon.com