Finding Undervalued Homes is the AMES

Debbie Trinh

June 2, 2023

Introduction

Purchasing a home is one of the biggest financial decisions in a person’s life.

Searching for a new home in an unfamiliar area can be overwhelming.

Hiring consultants eases the stress and uncertainty of finding an affordable home, which would enable clients to focus on other important matters.

Today's Objective

Compare a wide range of machine learning models to identify the most effective model to predict sales prices based on key features.

Discover undervalued homes for clients so they get the best bang for their buck and can purchase property that appreciates in value.

Housing Dataset Overview

2,580 records of residential homes in Ames, Iowa

79 features

Time Range: 2006-2010

Data Preprocessing Techniques

Numerical Variables:

StandardScaler used to scale the original data.

KNNImputer employed for imputing missing values.

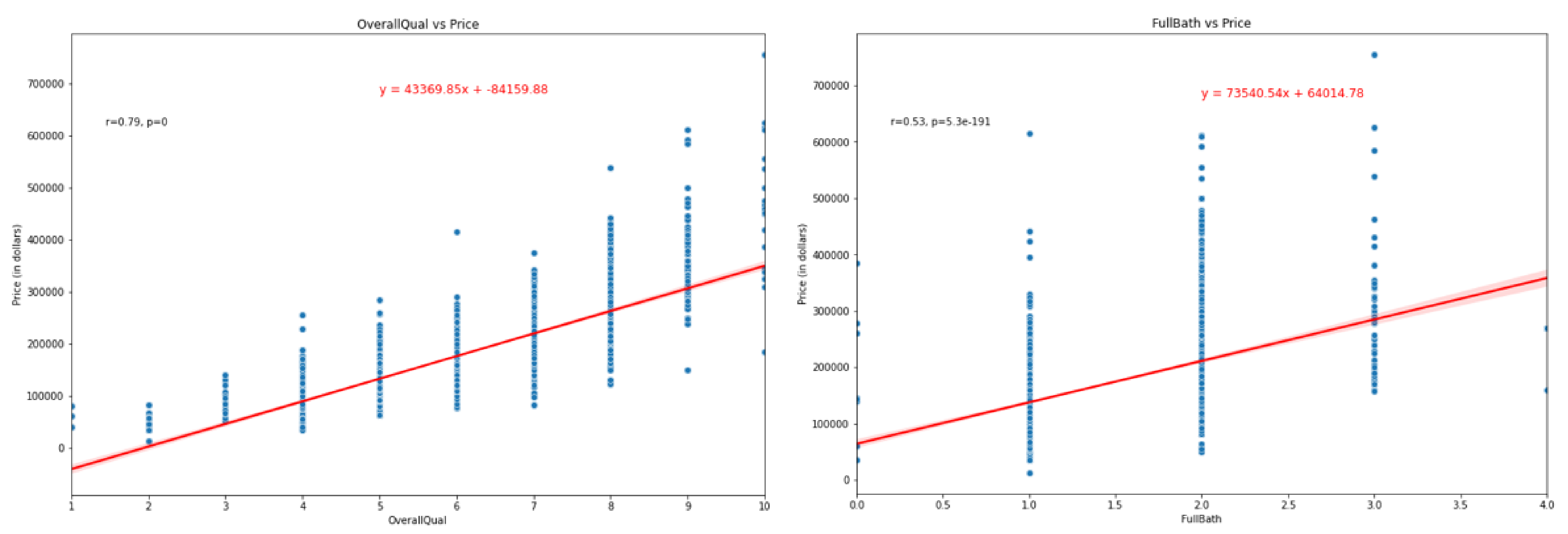
Rescaled back to the original scale using inverse_transform.

Categorical Variables:

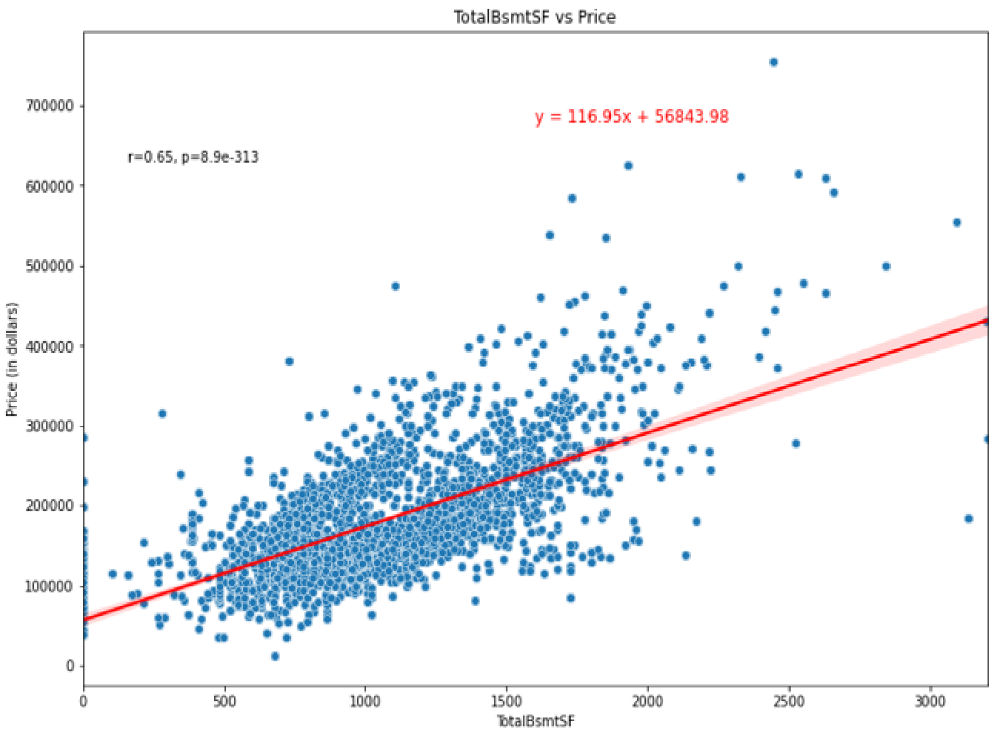
Missing values imputed with "Unknown".

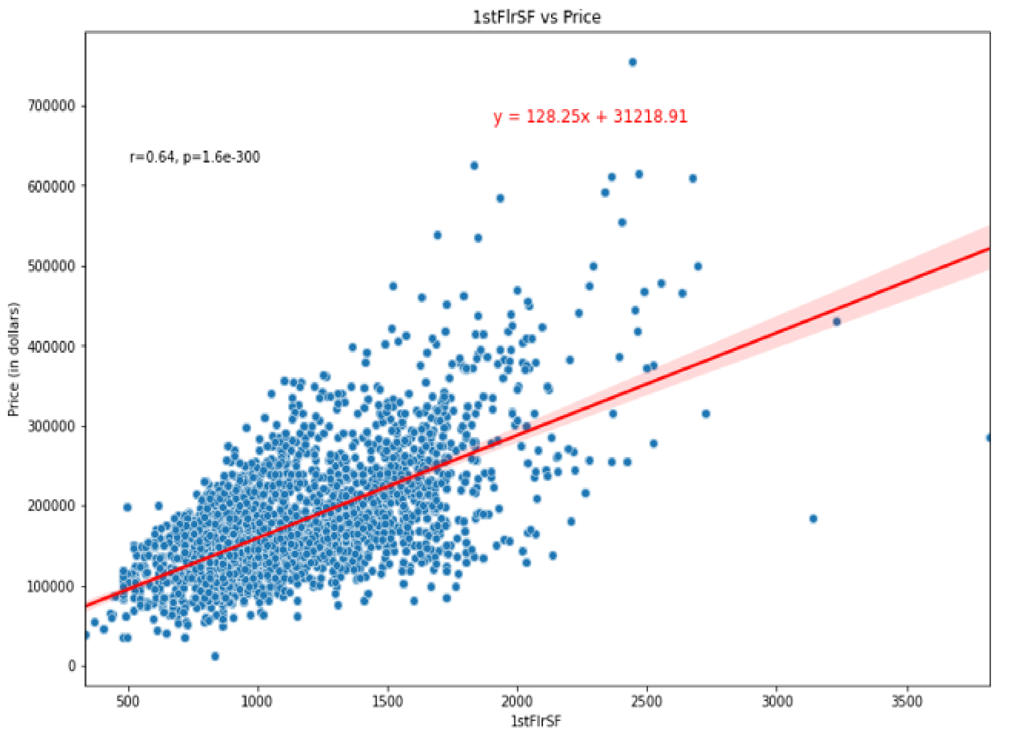
Encoded using one-hot encoding.

EDA – Univariate Analysis – I

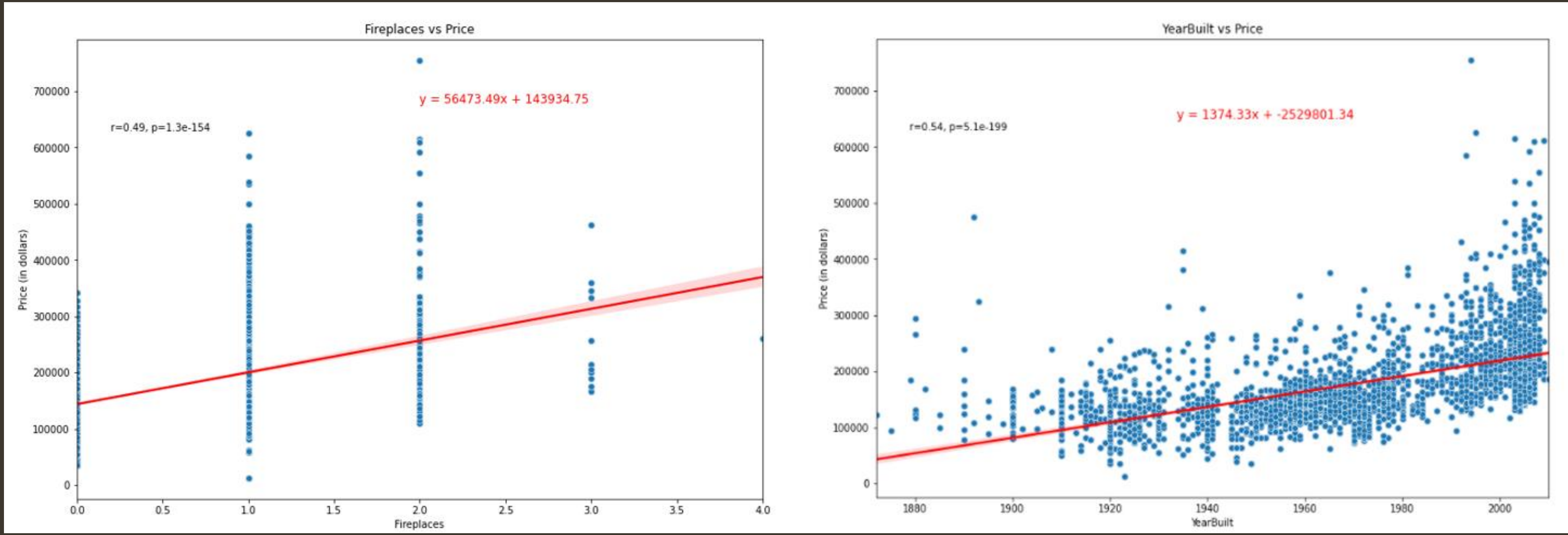


EDA – Univariate Analysis – II

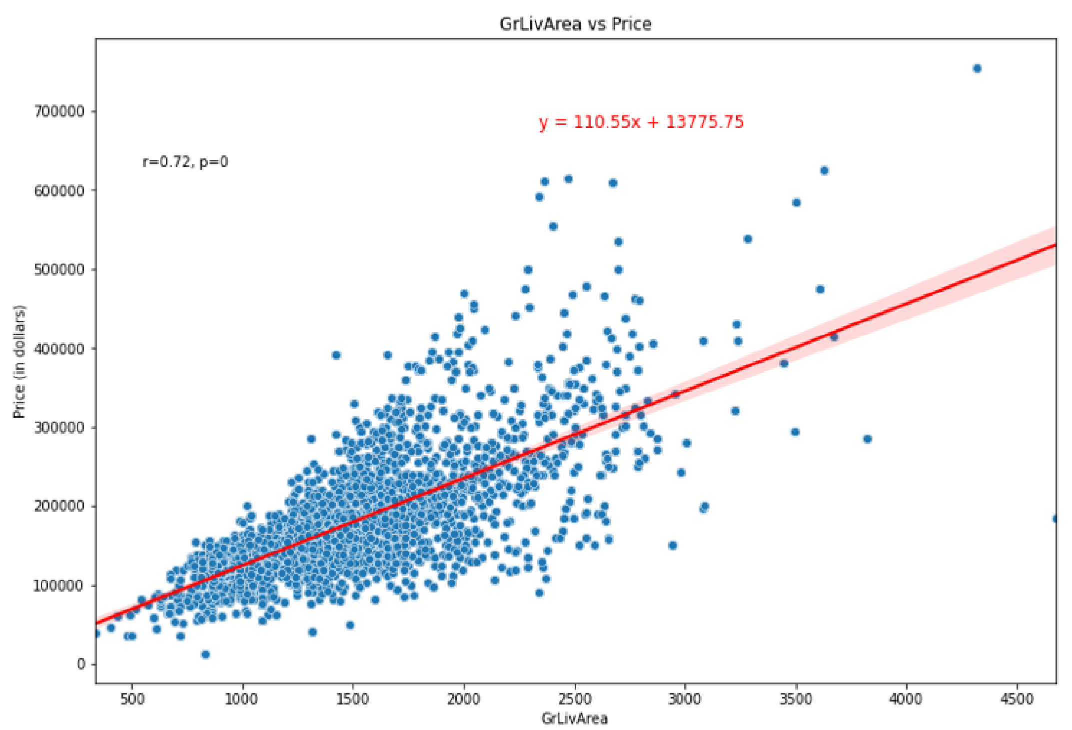


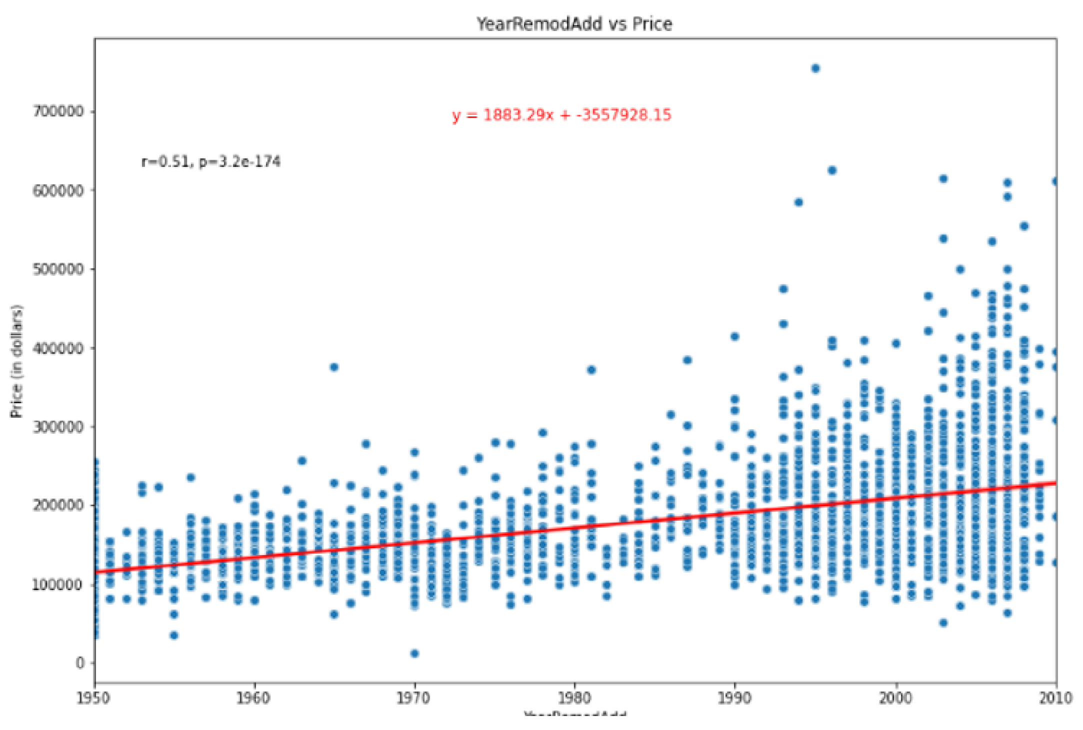


EDA – Univariate Analysis – III

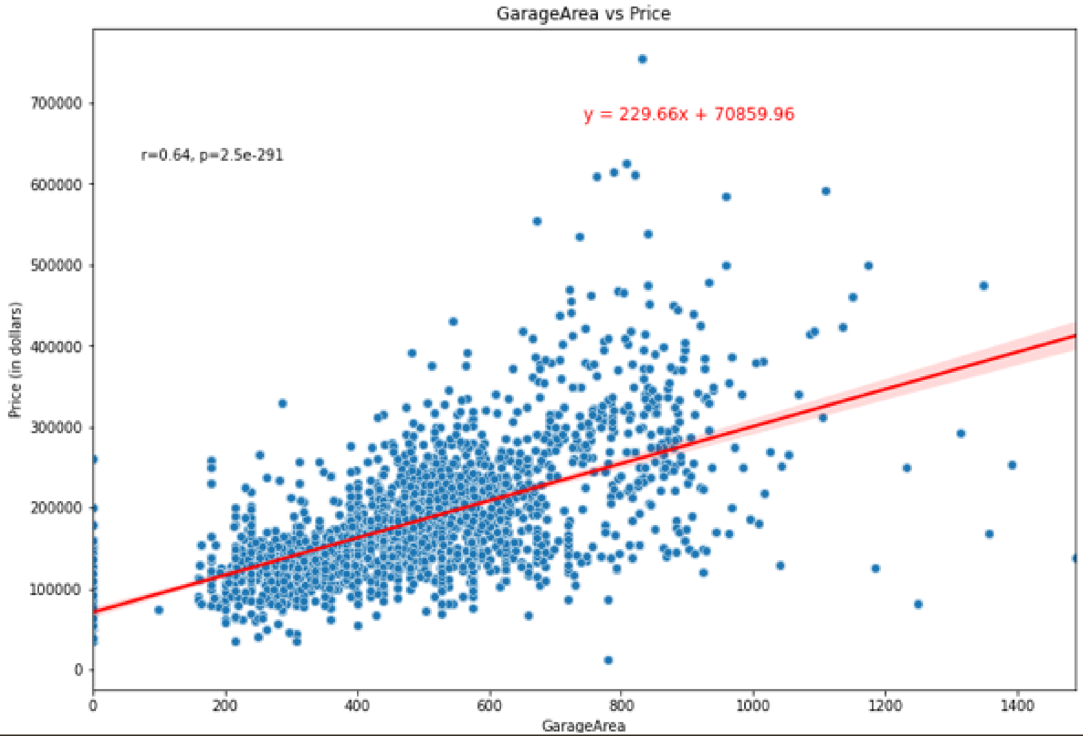


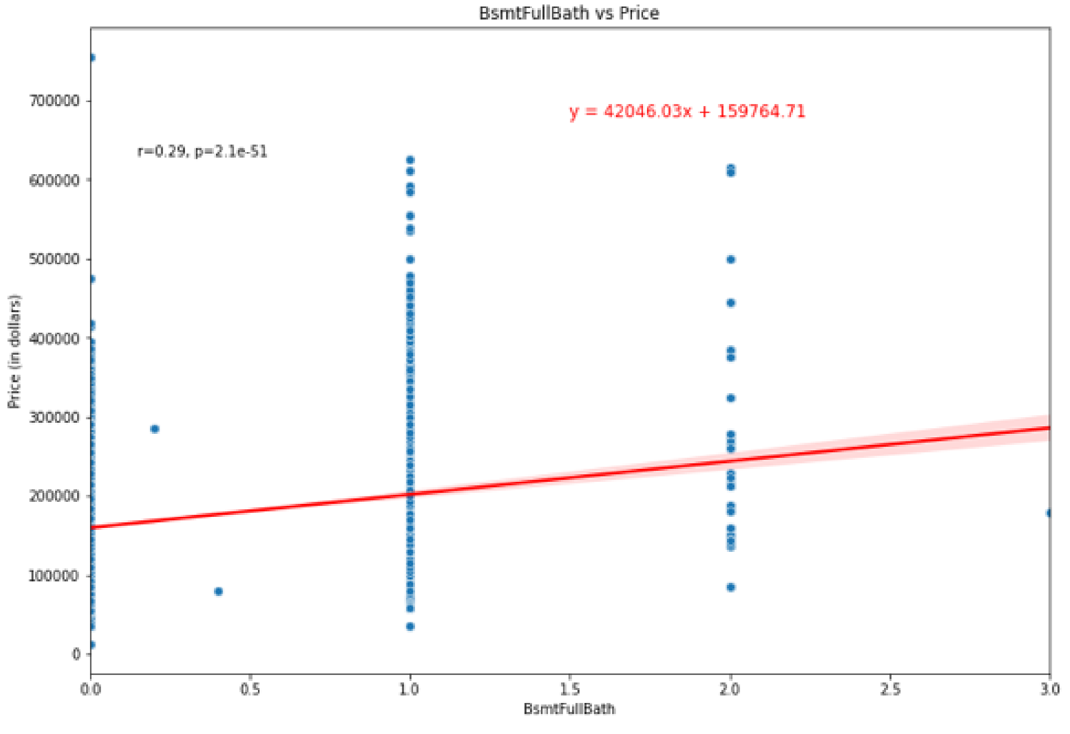
EDA – Univariate Analysis – IV



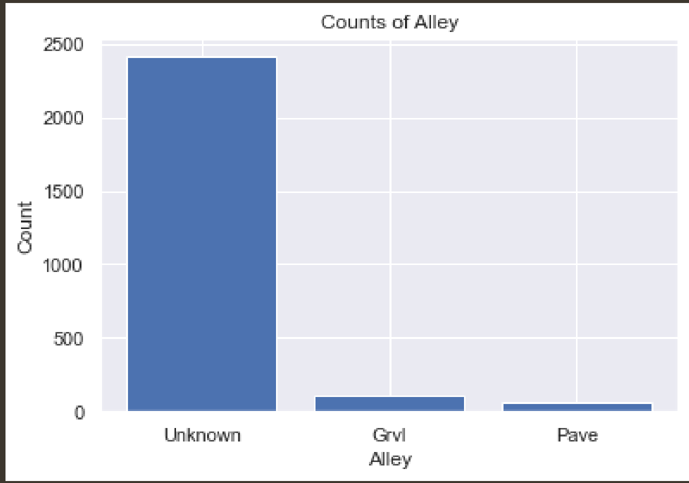


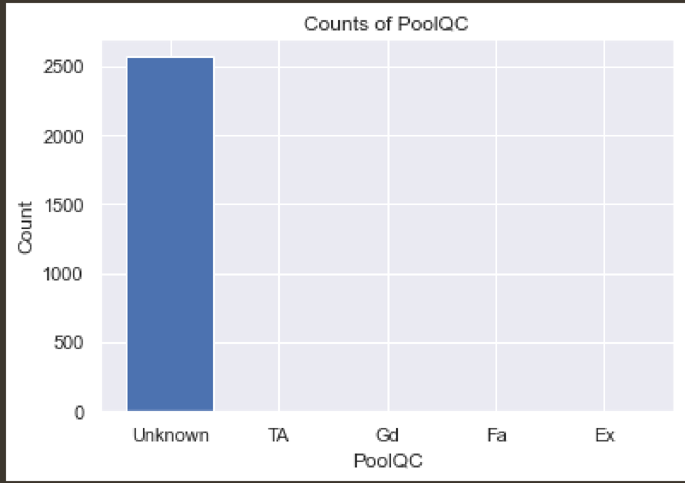
EDA – Univariate Analysis – V

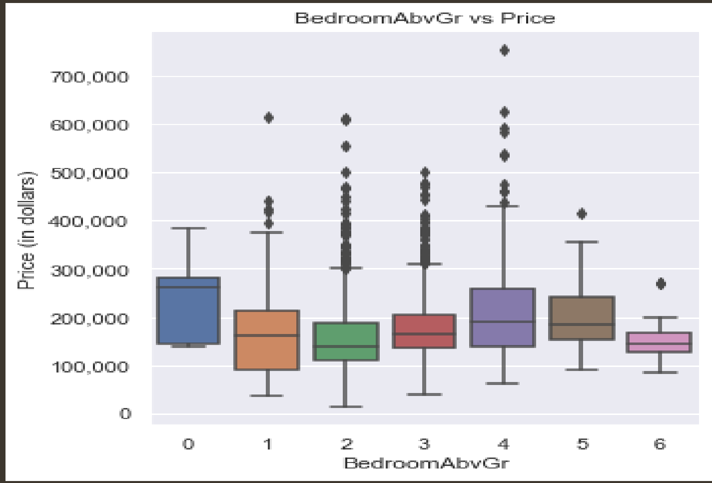


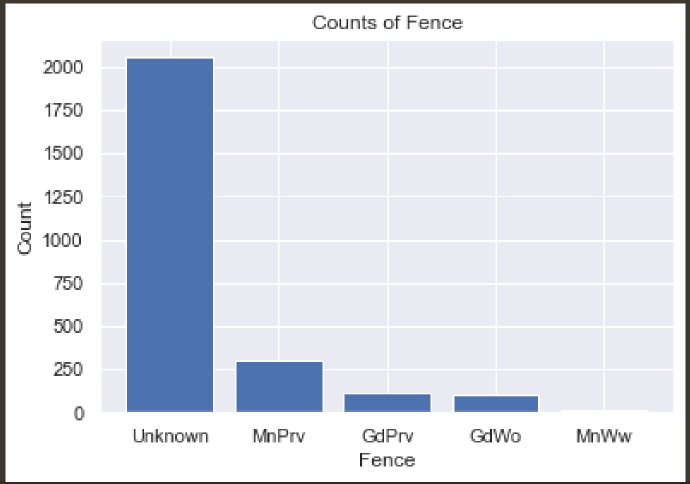


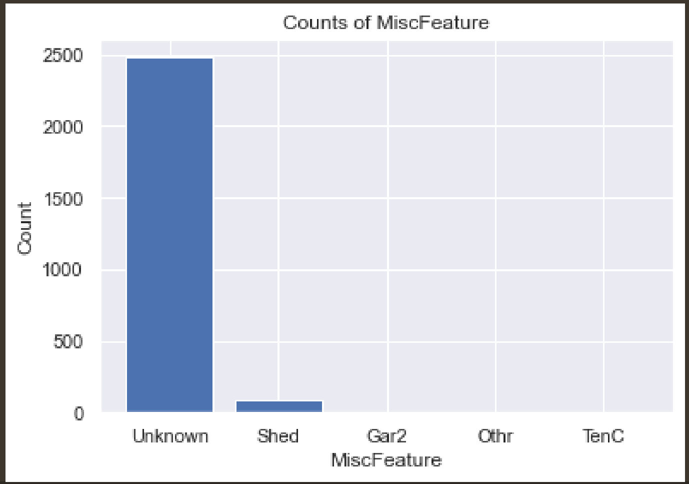
EDA – Dropped Features – Many Nulls or Indistinct Trend

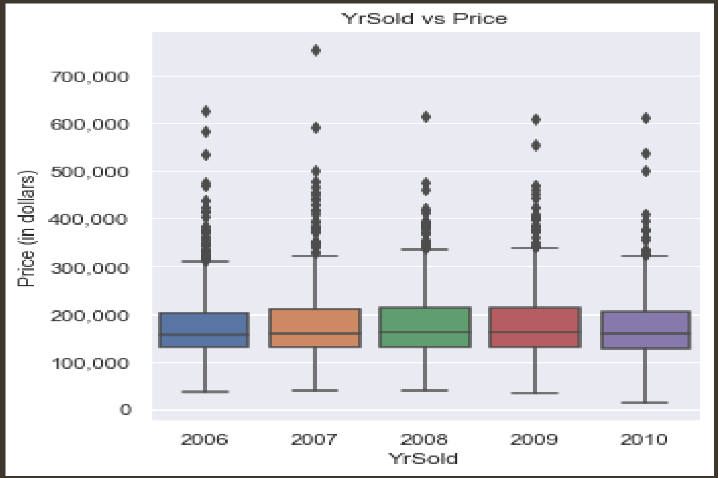




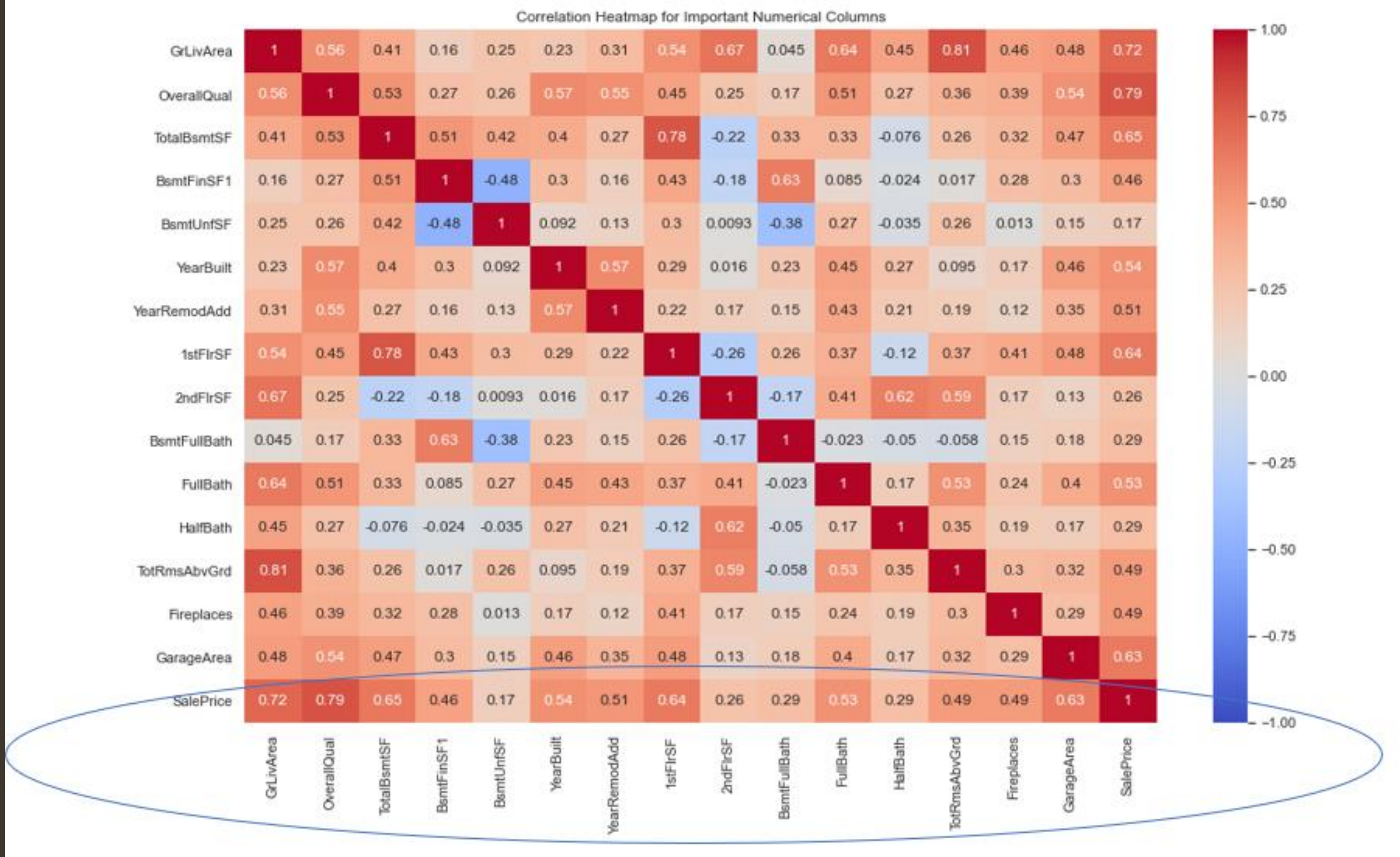








Heatmap – Correlation between Features and Target



Feature Selection Techniques

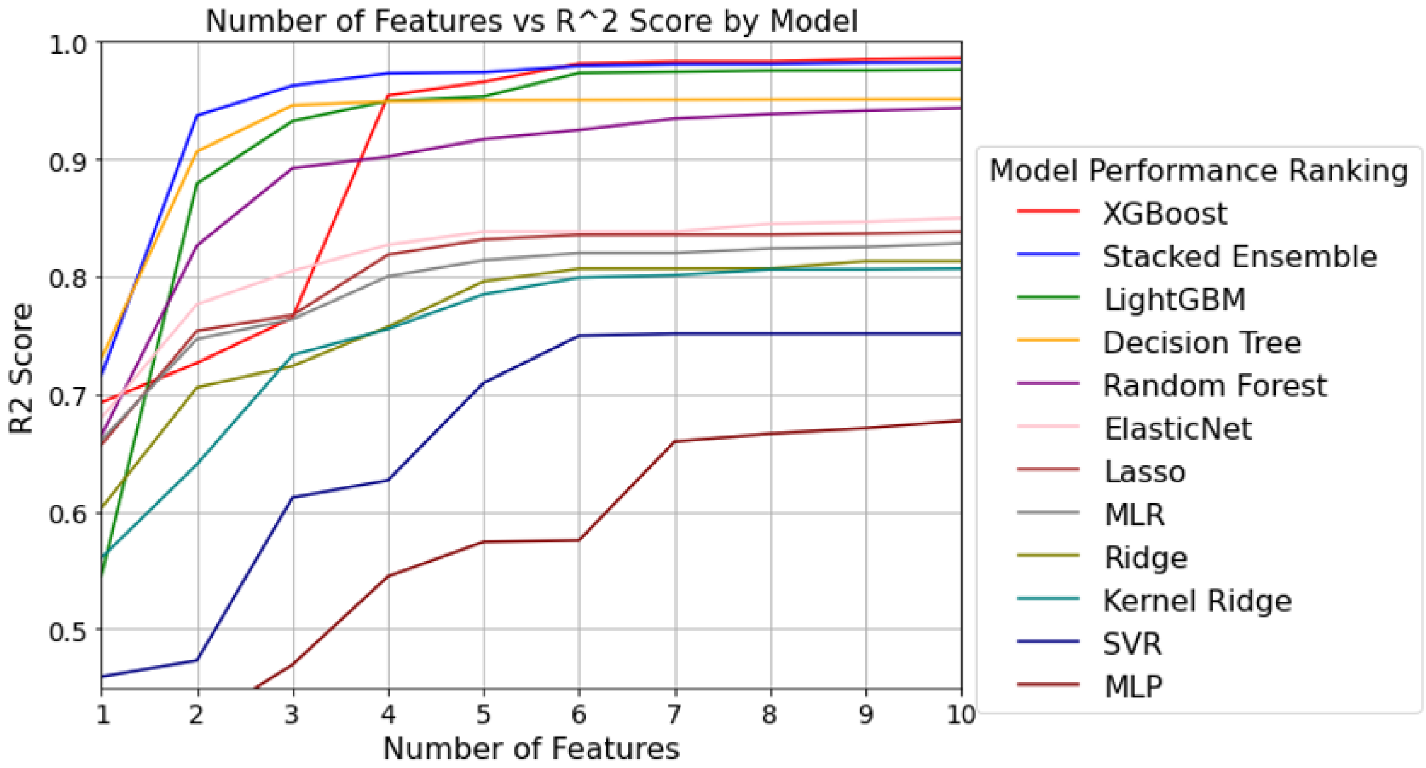
Original Technique: RFE (Recursive Feature Elimination)

- Optimal Number of Features: 10
- Selected Features:
  - 'GrLivArea', 'LotArea', 'OverallQual', 'YearBuilt', 'BsmtFinSF1', 'TotalBsmtSF', '1stFlrSF', 'Fireplaces', 'GarageArea’

Chosen Technique: Forward Feature Across All Models

- Less computationally expensive
- Improved overall efficiency

Number of Features vs R^2 Performance by Model



This is a line graph illustrating the performance of each model with their respective unique set of top 10 selected features.

Model Performance Evaluation

<table>
<tr><th>Model Performance Ranking
(Highest to Lowest)</th><th>Test R^2 of 10 Best Features (KFold CV=5, Shuffle=True)</th><th>Preprocessing and EDA</th><th>Feature Selection</th><th>Hyperparameter Tuning (Used GridsearchCV or RandomsearchCV)</th></tr>
<tr><td>XGBoost</td><td>0.986</td><td rowspan="12">1. Used StandardScalar to impute Numerical Nulls with KNN imputer, and rescaled back to original scale.

2. Imputed categorical nulls with "Unknown".

3. One hot encoded categorical variables, using drop_first=True to reduce multicollinearity.

4. Used VIF to detect and remove some multicollinear features.

5. In univariate analysis, used scatterplots and boxplots to identify features that had a strong correlation with the target variable (SalePrice) and dropped some features that did not have a clear relationship with SalePrice.

6. Used same preprocessed dataset across models.</td><td rowspan="12">Forward Feature Selection of 10 best features to improve R^2 using KFold CV=5, Shuffle=True.

For tree based models, used feature importance.

For SVR & MLP, used SelectFromModel.</td><td>n_estimators': 400, 'max_depth': 10, 'learning_rate': 0.1, 'min_child_weight': 10, 'colsample_bytree': 0.5, subsample': 0.75, 'gamma': 0</td></tr>
<tr><td>Stacked Ensemble</td><td>0.982</td><td>XGBoost, Random Forest, Decision Tree</td></tr>
<tr><td>LightGBM</td><td>0.976</td><td>n_estimators': 271, 'learning_rate': 0.1, 'max_depth': 6, 'num_leaves': 30, 'min_child_samples': 8, 'reg_alpha': 0.1, 'reg_lambda': 0.5</td></tr>
<tr><td>Decision Tree</td><td>0.951</td><td>max_depth': 15, 'min_samples_split': 4, 'min_samples_leaf': 1, 'max_features': None</td></tr>
<tr><td>Random Forest</td><td>0.943</td><td>n_estimators': 500, 'max_depth': 10, 'min_samples_split': 10, 'min_samples_leaf': 2</td></tr>
<tr><td>ElasticNet</td><td>0.85</td><td rowspan="7">Default</td></tr>
<tr><td>Lasso</td><td>0.838</td></tr>
<tr><td>Multiple Linear Regression</td><td>0.828</td></tr>
<tr><td>Ridge</td><td>0.813</td></tr>
<tr><td>Kernel Ridge</td><td>0.806</td></tr>
<tr><td>Support Vector Machine</td><td>0.751</td></tr>
<tr><td>Multi-Layer Perceptron</td><td>0.678</td></tr>
</table>

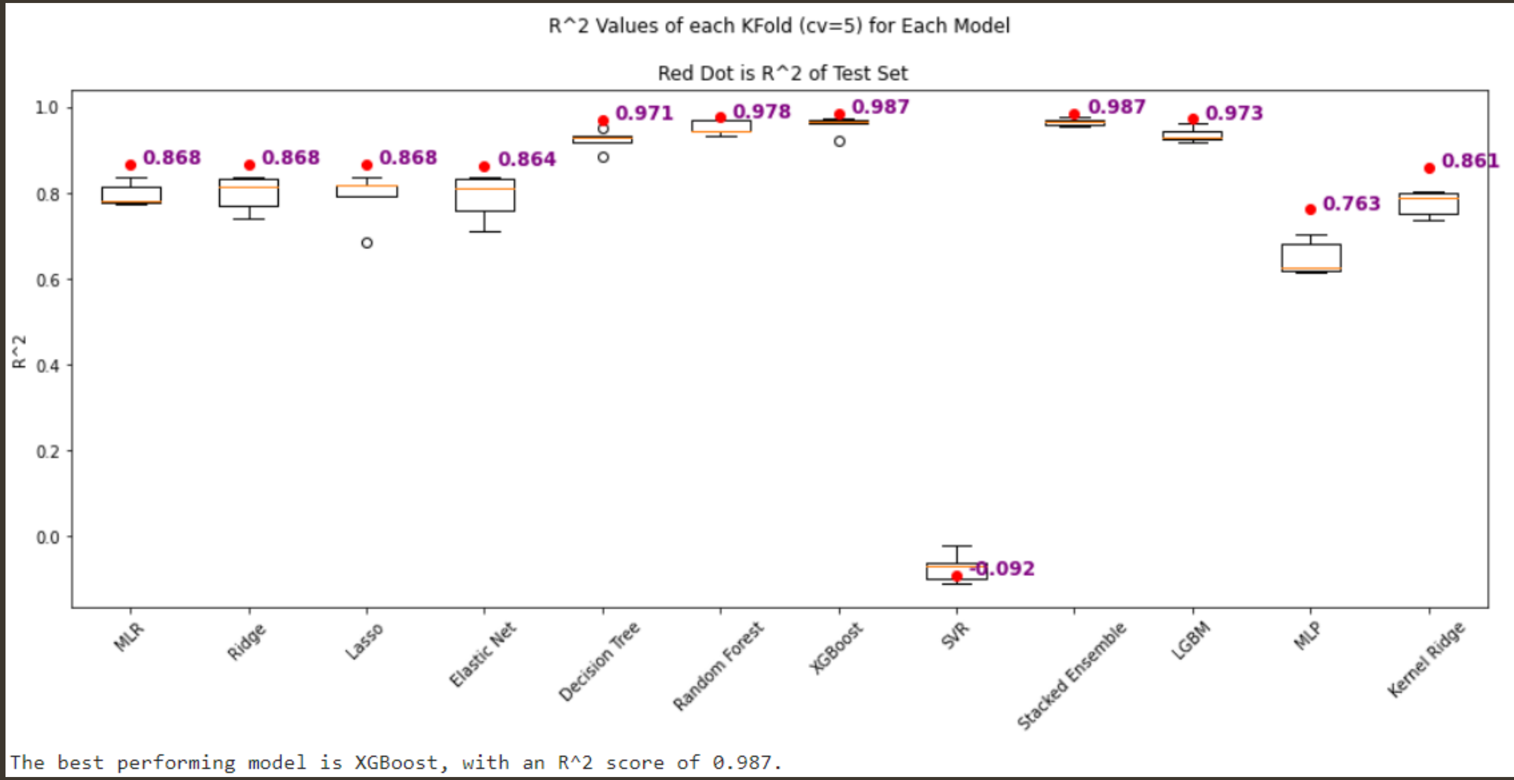
Top 10 Frequently Selected Features across Top 5 models

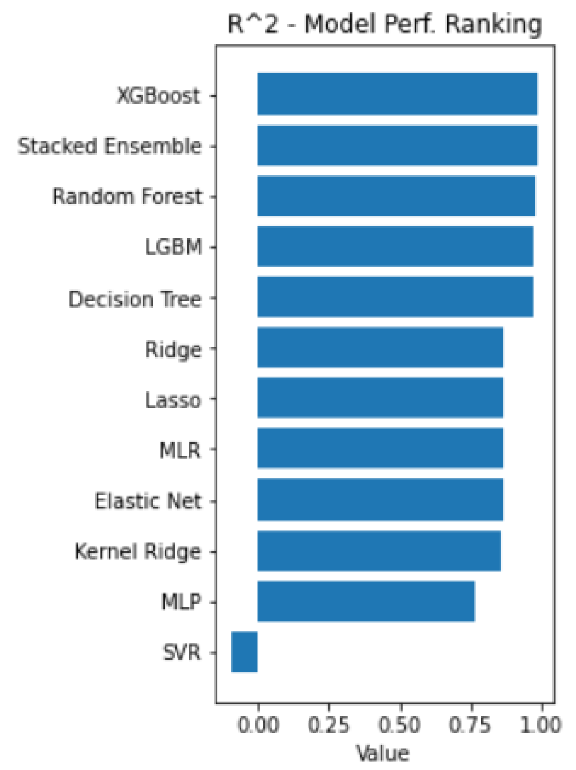
Using tree-based models (including stacked ensemble) in forward feature selection, this is the tally of how often certain features were selected across the Top 5 models: Decision Tree, Random Forest, LightGBM, Stacked Ensemble, and XGBoost.

Features	Frequency of Selection
OverallQual	3 times
GrLivArea	3 times
TotalBsmtSF	3 times
1stFlrSF	3 times
GarageArea	2 times
YearBuilt	2 times
YearRemodAdd	2 times
BsmtFinSF1	2 times
BsmtUnfSF	2 times
Fireplaces	2 times

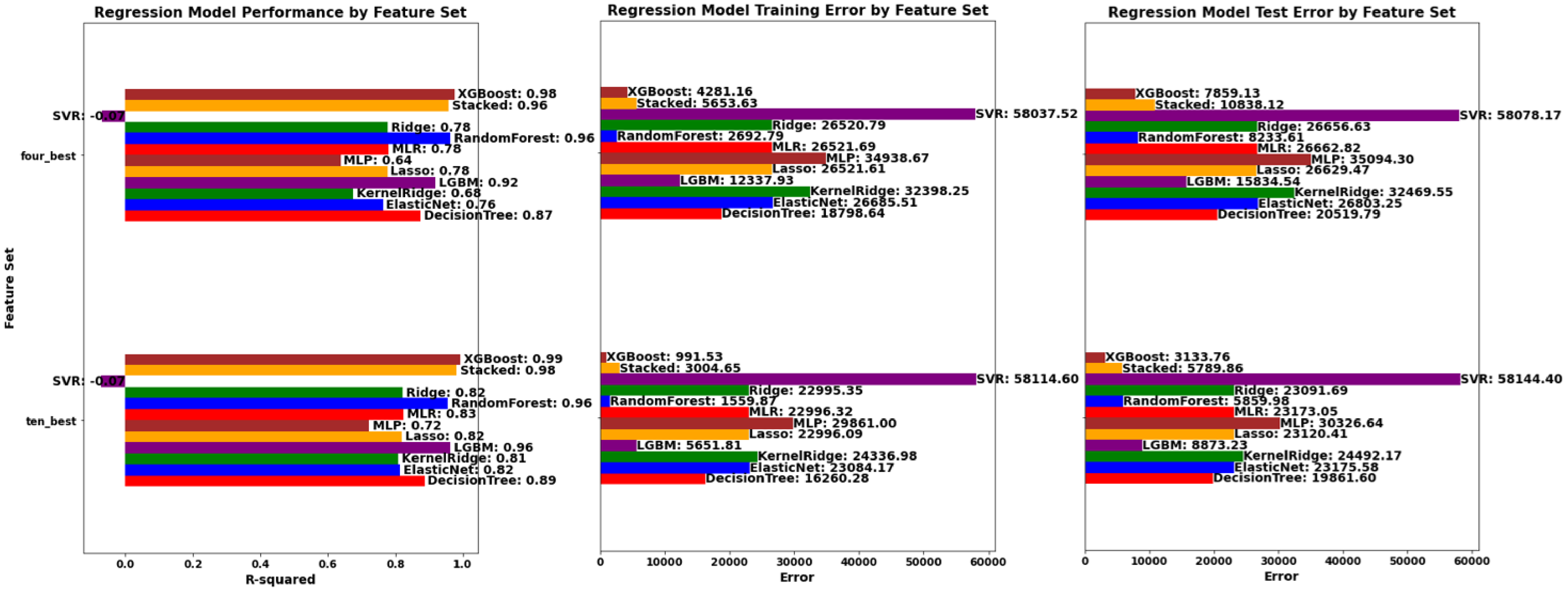
I used these specific features to automate model selection of 4 best and 10 best features with the goal of optimizing R^2.

Model Comparisons using Top 10 Frequently Selected Features + Neighborhoods





Model Comparisons using Top 4 & Top 10 Frequently Selected Features



Highest performing model based on maximizing R^2 for four_best: XGBoost (.98)
Highest performing model based on maximizing R^2 for ten_best: XGBoost (.99)

Best Model that Minimizes Training Error (MAE):
four_best: RandomForest (2692.79)
ten_best: XGBoost (991.53)

Best Model that Minimizes Test Error (MAE):
four_best: XGBoost (7859.13)
ten_best: XGBoost (3133.76)

Hyperparameter Tuning Winner Model: XGBoost

Used Top 10 Features

Best Hyperparameters:

Estimators: 300,

Learning Rate: 0.1,

Max Depth: 5

Test R^2 (cv=5, shuffle=True): .97

Top 10 Undervalued Properties			
Neighborhood	Actual Sale Price	Predicted Sale Price	Residual
Veenker	$ 150,000.00	$ 380,131.75	$ 230,131.75
NAmes	$ 84,900.00	$ 222,556.94	$ 137,656.94
NAmes	$ 167,000.00	$ 271,021.34	$ 104,021.34
MeadowV	$ 151,400.00	$ 247,910.06	$ 96,510.06
OldTown	$ 122,000.00	$ 216,369.02	$ 94,369.02
NWAmes	$ 278,000.00	$ 362,180.97	$ 84,180.97
NridgHt	$ 386,250.00	$ 470,245.03	$ 83,995.03
CollgCr	$ 239,000.00	$ 312,992.75	$ 73,992.75
CollgCr	$ 185,000.00	$ 255,453.08	$ 70,453.08
SWISU	$ 197,000.00	$ 265,626.69	$ 68,626.69

Looked for undervalued properties using residuals

where the Actual Sale Price < Predicted Sale Price.

Summary & Actionable Insights

Top features for predicting the target variable: OverallQual, GrLivArea, TotalBsmtSF, 1stFlrSF, GarageArea, YearBuilt, YearRemodAdd, BsmtFinSF1, BsmtUnfSF, Fireplaces.

To increase the value of undervalued property in the long term, prioritize investments that expand living space and feature high-quality materials, superior craftsmanship, and an elegant appearance.

Visit undervalued properties and consult with real estate brokers and appraisers to gain insights into factors not captured by the model.

Explore the impact of proximity to amenities like schools, parks, Starbucks, and shopping centers on house prices for potential price appreciation.

Further Approaches to Enhance Project

During feature selection, elected for least computationally intensive methods. With more time, explore other methods like backward stepwise selection that may select features that may perform well across models.

Perform feature engineering such as cut YearBuilt into bins that would have improved the performance of linear models.

Consider how to handle outliers that may affect housing pricing prediction.

To validate the predictions of an updated model for predicting house prices in Ames based on recent data and key features, compare the model's outputs with the actual prices listed on live platforms like Redfin.

Appendix

Models Compared & Descriptions

Multiple Linear Regression (MLR) is a statistical technique that finds a linear equation to analyze the relationship between a dependent variable and multiple independent variables. It quantifies how changes in the independent variables impact the dependent variable.

Lasso Regression boosts prediction accuracy by selectively focusing on the most significant factors. Through the use of an L1 penalty, less influential coefficients are set to zero, enabling feature selection and enhancing the accuracy of predictions by considering only the most important factors.

Ridge Regression enhances prediction accuracy by incorporating an L2 penalty, which promotes smaller coefficient values and mitigates the impact of multicollinearity. This leads to a more stable and precise model, ensuring reliable predictions.

ElasticNet Regression combines the benefits of Ridge and Lasso Regression by applying both L1 and L2 penalties, allowing for feature selection and handling correlated variables flexibly.

A Decision Tree predicts a target variable by dividing data into subsets using a tree-like model, where each internal node represents a test on a specific feature & each leaf node represents a predicted value for the target.

Random Forest is a collection of decision trees that work together to make predictions, utilizing different subsets of data and features, and aggregating the individual tree predictions for improved accuracy and generalization.

Models Compared & Descriptions

XGBoost is known for constructing shallow trees and focusing on minimizing the errors of those trees. It is particularly effective in handling tabular data and can automatically capture non-linear relationships and interactions between features and the target variable. By employing boosting techniques, XGBoost sequentially adds trees to correct the errors of previous trees, leading to improved prediction accuracy.

Support Vector Regression (SVR) predicts outcomes by finding the optimal boundary between different possibilities, aiming to minimize errors and maximize the margin between predicted values and the boundary.

A stacked ensemble is a technique that combines multiple prediction models, leveraging their individual strengths, to generate more accurate and robust predictions.

Light Gradient Boosting Machine (LGBM) constructs prediction models by combining numerous basic decision trees. It is known for its efficient histogram-based approach, which speeds up training and prediction, as well as its leaf-wise tree growth strategy, which enhances predictive accuracy.

Multilayer Perceptron (MLP) is a powerful algorithm inspired by the human brain, capable of identifying intricate patterns and relationships among variables, making it well-suited for complex tasks.

Kernel Ridge is a machine learning algorithm that maps input data into a high-dimensional feature space using a kernel function, then performs ridge regression to predict outcomes. It is commonly used for regression tasks and can handle nonlinear relationships between input variables and the target variable.

Models Strengths & Weaknesses

1. MLR Regression

- Strengths:
  - Easy to understand and interpret the coefficients of the model
  - Can handle both categorical and continuous predictor variables
- Weaknesses:
  - Assumes a linear relationship between the predictors and the response, which may not always hold true
  - May not work well with high-dimensional data or correlated predictors

2. Ridge Regression

- Strengths:
  - Reduces the effect of multicollinearity in the data
  - Can handle a large number of predictors even when the sample size is small
- Weaknesses:
  - Requires tuning of the regularization parameter, which can be challenging
  - Can introduce bias in the estimates of the coefficients

Models Strengths & Weaknesses

3. Lasso Regression

- Strengths:
  - Performs feature selection by shrinking the coefficients of less important predictors to zero
  - Can work well with high-dimensional data and correlated predictors
- Weaknesses:
  - May not work well with a small sample size
  - Can be unstable in the presence of highly correlated predictors

4. Elastic Net

- Strengths:
  - Combines the strengths of Ridge and Lasso Regression by balancing between the two methods
  - Works well with high-dimensional data and correlated predictors
- Weaknesses:
  - Requires tuning of the regularization parameter, which can be challenging
  - Can be computationally expensive for large datasets

Models Strengths & Weaknesses

5. Decision Tree

- Strengths:
    - Can capture non-linear relationships between the predictors and the response
    - Easy to interpret and visualize
- Weaknesses:
    - Prone to overfitting, especially when the tree is deep
    - Can be sensitive to the choice of hyperparameters

6. Random Forest

- Strengths:
    - Reduces the overfitting of a decision tree by aggregating multiple trees
    - Can handle high-dimensional data and correlated predictors
- Weaknesses:
    - Can be computationally expensive, especially for large datasets
    - May produce biased predictions for imbalanced data

Models Strengths & Weaknesses

7. XGBoost

- Strengths
  - Known for its predictive accuracy, often outperforming other algorithms by capturing complex patterns & interactions in data.
  - Employs parallel processing and optimization techniques to expedite both training and prediction.
- Weaknesses:
  - Finding the optimal set of hyperparameters for XGBoost models can be challenging and time-consuming.
  - Can be less interpretable compared to simpler models such as linear regression or decision trees. Understanding the underlying reasoning behind the predictions may be more challenging.

8. SVR

- Strengths:
  - Can capture non-linear relationships between the predictors and the response
  - Works well with small sample sizes
- Weaknesses:
  - Requires tuning of the regularization parameter and kernel function, which can be challenging
  - Can be sensitive to outliers in the data

Models Strengths & Weaknesses

9. Stacked Ensemble

- Strengths:
    - Can combine the strengths of multiple models to improve the overall predictive performance
    - Can handle different types of predictors and non-linear relationships
- Weaknesses:
    - Can be computationally expensive, especially for large datasets
    - Requires tuning of many hyperparameters, which can be challenging

10. LGBM:

- Strengths: Fast, handles large datasets, good for many features.
- Weaknesses: Prone to overfitting, difficult to interpret.

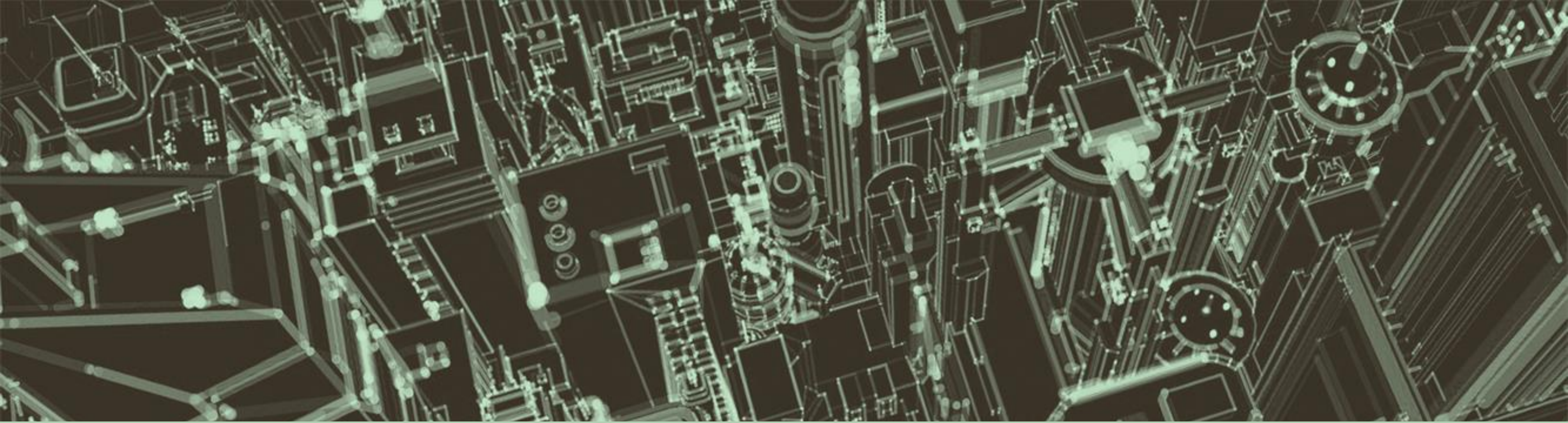
Models Strengths & Weaknesses

11. MLP:

- Strengths: Learns complex patterns, handles different data types, can be fine-tuned.
- Weaknesses: Prone to overfitting, computationally expensive, difficult to interpret.

12. Kernel Ridge:

- Strengths:
  - Kernel ridge is able to handle non-linear relationships between variables.
  - It provides a solution to overfitting by balancing the weights of the regression coefficients.
- Weaknesses:
  - The choice of kernel function can significantly affect the performance of the algorithm.
  - It is computationally intensive and can be slow for large datasets.



Thank You

Debbie Trinh
New York City Data Science Academy
January 2023 Cohort