April 17th, 2022

Final Report

Group 30, ISYE 7406 Data Mining and Statistical Learning

House Prices - Advanced Regression Techniques

Mohammed Salman Saeed	882	ssaeed@gatech.edu
Ryan Brent Keeney	077	rkeeney6@gatech.edu
Thang Sung Trieu	886	ttrieu8@gatech.edu

Abstract

Our project applies advanced Regression techniques to predict the final price of each home from the Ames Housing dataset, compiled by Dean De Cock, a high-dimensional data set with 79 predicting variables. The goal of this analysis is to predict the sales price for each house.

During this analysis, we find that there are many missing data points, 5.88% of data is missing in the training data (see Figure 1 below) and 6% of data is missing in the testing data. Comprehensive EDA helped identify outliers using several types of univariate graphical techniques such as boxplots, histograms, and density plots were used to understand the data better. Missing values were imputed using a categorical median or imputed based on highly correlated variables.

Due to the high dimensionality of the dataset, variables selection is critical. Our team used three methods to evaluate which variables should be passed on to the final model, lowering the effort and cost for future predictions. Relative to AIC stepwise regression, BIC stepwise penalizes the model more for its complexity and is preferred for scenarios where there are few significant coefficients relative to the entire predictor set. In this case, both BIC stepwise linear regression and LASSO regression selected the same set of predictors, with a few minor expectations.

Finally, our team evaluated four optimized models, with parameters selected using a cross-validated grid search. In our analysis, the baseline model (Linear Regression) performed the best. Overall, house prices can be predicted with a high degree of confidence with all the models evaluated. The XGBoost, or random forest with gradient boosting, outperformed the random forest model slightly, while the KNN model performed the worst.

Introduction

Utilizing the Ames Housing dataset, our team will focus on evaluating variable selection and modeling methodologies. The data presents unique high-dimensional challenges with 79 explanatory variables. Due to the relatively small size of the dataset - just over 1,500 records - we will also use k-folds cross-validation as a resampling procedure to build confidence in and test the effectiveness of our models [6, 7].

Our primary focus will focus on evaluating variable selection and modeling methodologies. However, during the process of analyzing the dataset and researching primary objectives, additional questions will be addressed.

Primary Research Questions

What method of variable selection and/or feature engineering results in the best prediction?

The dataset, with 79 predicting variables, has high dimensionality and potential multicollinearity [1, 9]. Relying on many factors may increase the cost for future predictions if additional data must be collected. Our team has

proposed reviewing methods such as stepwise, LASSO, and ridge regression techniques to address these two issues [2, 3, 4]. Additionally, the team will discuss methodologies to improve prediction through feature engineering [5]. With a high number of predicting variables, we would also use dimensionality reduction techniques such as PCA if appropriate [10].

What modeling methodology results in the best prediction?
Our team will compare the performance of 3 advanced regression techniques; random forest, random forest with gradient boosting, and k-nearest-neighbor against a baseline (linear regression). The results from our proposed approaches will be evaluated on Root-Mean-Squared-Error (RMSE) between the logarithm of the predicted value and the logarithm of the observed sales price.

Secondary Research Questions
Our team will also explore and discuss the related research questions.

- How can exploratory data analysis be effectively communicated in high-dimensional datasets?
- What visual methods are effective with high-dimensional datasets [8]?
- How should parameter tuning be performed for the selected regression techniques?

Problem Statement or Data Sources

The Ames Housing dataset is publicly available from the Kaggle competition page [1]. It contains housing data with 79 explanatory variables which describe various aspects of residential homes. While the competition challenges users to predict the final price of each home, our team will focus on evaluating variable selection and modeling methodologies. The results from our proposed approaches will be evaluated on Root-Mean-Squared-Error (RMSE) between the logarithm of the predicted value and the logarithm of the observed sales price.

The data was provided to us by Ned Devid Research. It consisted of four cvs files: netincome (monthly net income), eps (earning per share) , sales (monthly sales), and pricing (weekly stock prices) and one text file: description(company description and industry). The project challenges us to identify trending themes in the market before they are identified by other investors using historical pricing data and company business descriptions, our team will focus on evaluating outperformed stock and modeling methodologies. The result from our proposed approached will help us identifying emerging themes before they are well established by other investors, and the use natural language processing and wordnet combination will improve the accuracy in determining the themes.

As an approach to identify trending themes we divided this into three steps. First, we used the pricing (weekly stock prices) dataset compared that to the S&P 500 index to help us point out any individual stock that have outperformed the S&P500 over the past 5 years or any period of times. Next, we used the K-Means algorithm to

find the group of outperformed stocks that are moving together based on the changes in net income, stock prices, earning per share over the past year as well as beta and alpha values to find any correlation between the market and stock trend. Lastly, from the group of outperformed stocks we used the Natural Language Processing to find words that occur most frequently, and used WordNet(similar to thesaurus) to find similar word in company business description dataset in order to manually determined trending themes.

As part of the Exploratory Data Analysis, we first look at the training and testing dataset dimension. The training dataset has 1460 records with 81 columns and the testing dataset has 1459 records with 80 columns. During this analysis, we find that there are many missing data points, 5.88% of data is missing in the training data (see Figure 1 below) and 6% of data is missing in the testing data. Hence, as a part of the data preparation, we must come up with some way to handle missing data points.

Missing Values

Training data set

index	missing_values
PoolQC	1451
MiscFeature	1402
Alley	1365
Fence	1176
FireplaceQu	690
LotFrontage	259
GarageType	81
GarageYrBlt	81
GarageFinish	81
GarageQual	81
GarageCond	81
BsmtExposure	38
BsmtFinType2	38
BsmtQual	37
BsmtCond	37
BsmtFinType1	37
MasVnrType	8
MasVnrArea	8
Electrical	1

Figure 1.

Next, we ran the boxplot method in R to see if there was a definite correlation between neighborhoods and sale prices. We also wanted to see if there is any skewness regarding outliers, an outlier is an observation point that is distant from other observations, through the data quartiles (or percentiles) and average. Based on the result (see Figure 2.), the areas such as Northridge and Northridge Heights have much higher prices (and high outliers) in terms of pricing.

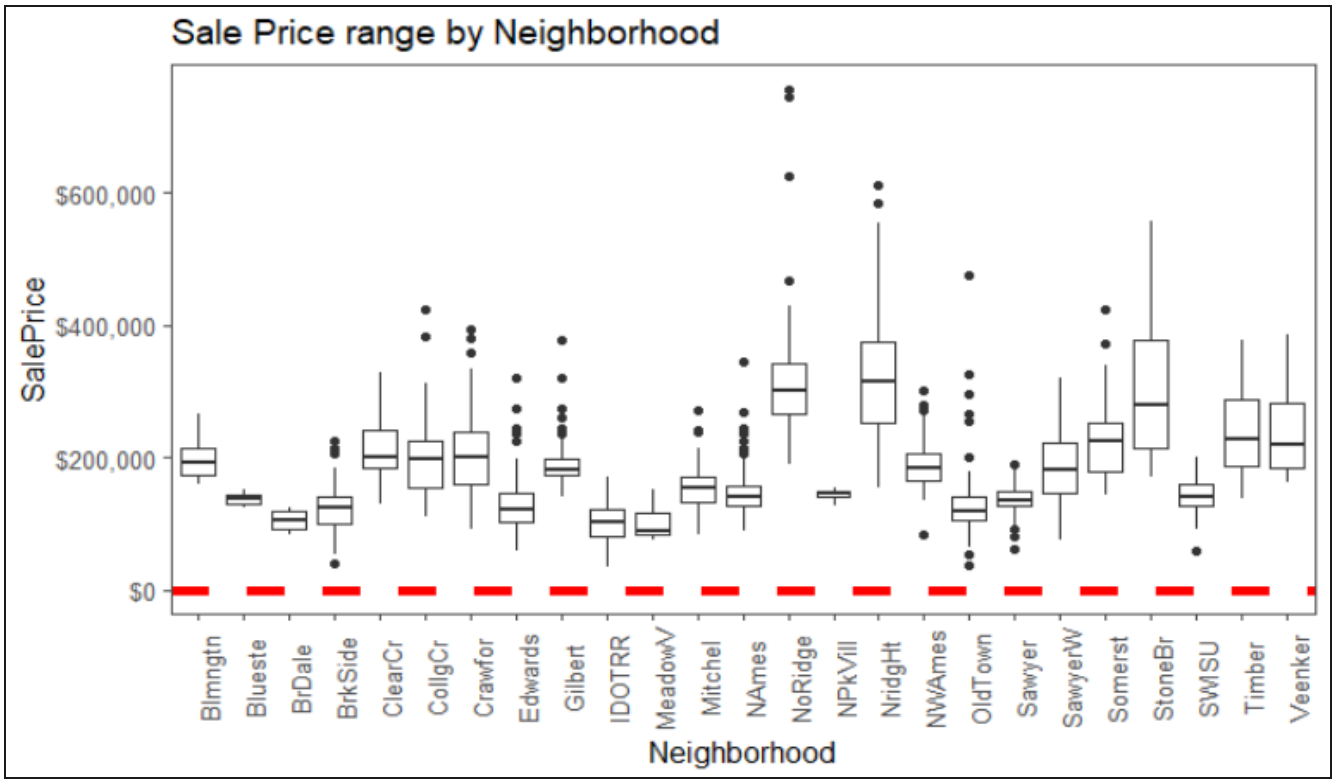


Figure 2.

Next, we ran both density plots and histograms for all numeric variables to understand the distribution and skewness of the data. There are 38 numeric variables and a sample of their density plots is shown here. These plots can also help identify multicollinearity. Multicollinearity variables generally occur when there are high correlations between two or more predictor variables. In other words, one predictor variable can be used to predict the other. (Please see Figure 3)

- YearBuilt Downturn in recent years after a high peak
- LotFrontage Highly skewed with majority <100 feet
- TotalBsmtSF Highly skewed with very few >2000 sqft
- GrLivArea Skewed with a minority >2500 sqft
- OverallQual Most houses are in 5-7 range
- OverallCond Most houses are in average (5) condition

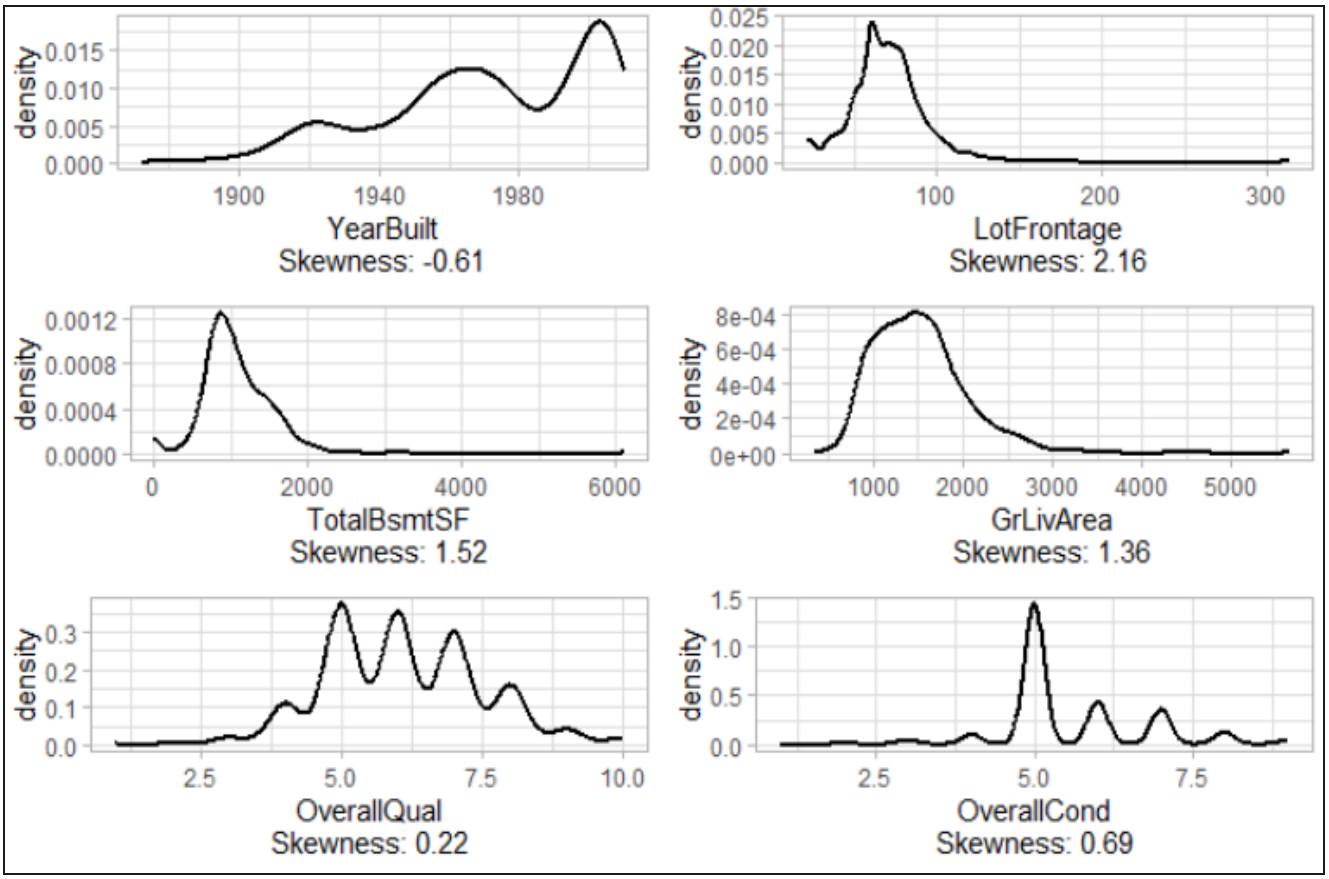


Figure 3.

Lastly, we used a correlation matrix to identify variables with a high correlation (positive or negative) with the SalePrice and also to identify multicollinearity. Some key observations are:

- SalePrice is highly correlated with Overall Quality but not with Overall condition of the house
- Houses built in later years (YearBuilt) have a higher sale price than those built in earlier years
- Above grade Living area (GrLivArea) is positively correlated with sales price as well as other variables indicating size of the house such as Total basement size (TotalBsmtSF)
- Larger garages fetch higher prices (GarageCars)
- The year the house was built (YearBuilt), garage was built (GarageYrBlt) and remodel year (YearRemodAdd) exhibit multicollinearity

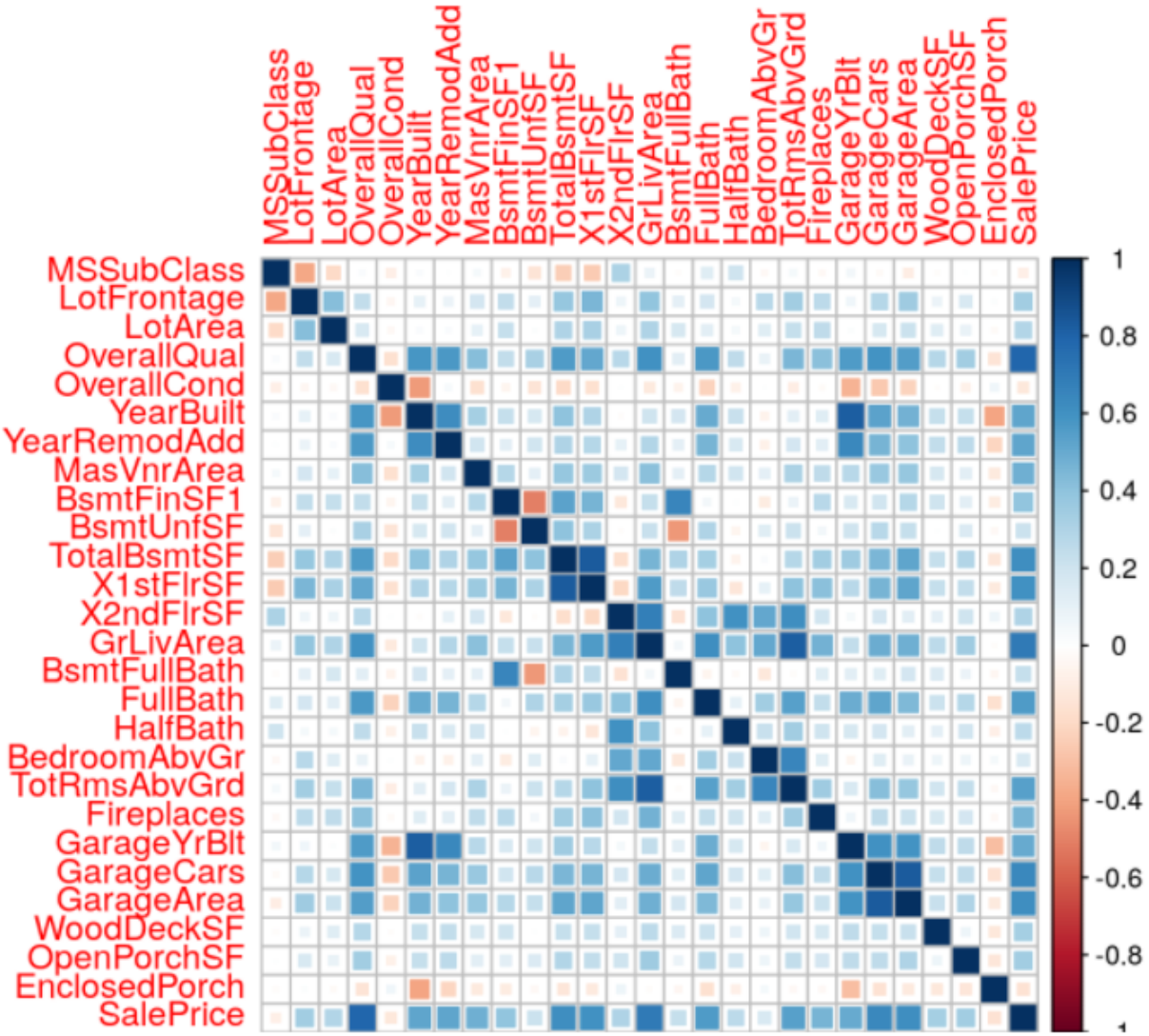


Figure 4.

At the end of our data analysis, we find many problems/challenges within our dataset that we must handle and address in order to increase the model accuracy.

1. Address missing data points and null values
2. Address multicollinearity variables
3. Run dimensionality reduction techniques to reduce the number of input variables in data set
4. Run variable selection methods to remove the insignificant and redundant variables in our model

Proposed Methodology

Our team approached the analysis with the following process:

Step	Method	Details
01	Data Preparation	The data was loaded and inspected.
02	EDA	With 79 variables, we needed to understand the correlation between the different variables and the distribution of numerical variables. This would also help us to tease out possible areas of multicollinearity and provide direction for feature engineering. Comprehensive EDA also helped to identify outliers Several types of univariate graphical techniques such as boxplots, histograms, and density plots were used to understand the data better.
03	Feature Engineering and Cleaning	Missing data points in LotFrontage were imputed with median values. The remaining NA data points were imputed as "None" or "0" for categorical and numerical predictors, respectively. Outliers (GrLivArea > 4000) were removed.
04	Variable Selection	Two methods of variable selection were evaluated and compared. 1. Stepwise linear regression 2. LASSO regression
05	Model Development	Each model was developed using the following procedure. 1. 80/20 test/train split, 5-fold cross-validation 2. Numerical predictors were normalized 3. Categorical predictors were dummy stepped where appropriate 4. Predictors with correlations $\rho = 1$ were identified and one was kept. 5. Outcomes were log-transformed 6. A primary tuning grid was evaluated using 5-fold cross-validation. Where appropriate, a secondary tuning grid was also used to focus on critical parameter spaces. The optimized set of parameters was selected based on RSME from the tuning grids 7. The model was retrained utilizing the optimized parameter set. 8. Performance metrics were collected for the

		optimized model on the withheld testing data. *Additional analyses, including goodness-of-fit and variable importance, were also performed.*
06	Model Evaluation	Each tuned model performance was evaluated on withheld test data (performance metrics: RSME and RSQ).

Feature Engineering, Cleaning, and Imputation

There are many missing data points within this data set. Missing data points in LotFrontage were imputed with median values. The remaining NA data points were imputed as “None” or “0” for categorical and numerical predictors, respectively, based on their associated numerical or categorical predictors. Outliers (GrLivArea > 4000) were removed.

Variable Selection

A combination of AIC and BIC stepwise linear regression and LASSO regression were utilized. LASSO is an ideal method in this scenario because there are few significant coefficients relative to the entire predictor set. Relative to AIC stepwise regression, BIC stepwise penalizes the model more for its complexity and is preferred for scenarios where there are few significant coefficients relative to the entire predictor set. In this case, both BIC stepwise linear regression and LASSO regression selected the same set of predictors, with a few minor expectations. In each case, the outcome variable was log-transformed. A LASSO variable trace plot is supplied in the Appendix [Additional Figures].

Ridge regression is not a variable selection technique but does address multicollinearity. We excluded ridge regression in favor of evaluating multicollinearity manually through EDA correlation matrices and then systematically removing the remaining highly correlated variables as the models were developed.

With a high number of predicting variables, we also considered dimensionality reduction techniques such as PCA [10]. Eventually, we did not use PCA as it is not recommended for data sets with both categorical and continuous predicting variables. Other methods, such as Multiple Factor Analysis are more appropriate for data sets with both categorical and continuous predicting variables but were not explored in this analysis.

Model Development

Our team compared the performance of 3 advanced regression techniques; random forest, random forest with gradient boosting, and k-nearest-neighbor against a baseline (linear regression). The results from our proposed approaches are evaluated with Root-Mean-Squared-Error (RMSE) between the logarithm of the predicted value

and the logarithm of the observed sales price. Taking logs means that errors in predicting expensive houses and cheap houses will affect the result equally.

Each model was developed using the following procedure.

1. 80/20 test/train split, 5-fold cross-validation
2. Numerical predictors were normalized
3. Categorical predictors were dummy stepped where appropriate
4. Predictor groups with correlations $\rho = 1$ were identified and one was kept.
5. Outcomes were log-transformed
6. A primary tuning grid was evaluated using 5-fold cross-validation. Where appropriate, a secondary tuning grid was also used to focus on critical parameter spaces. The optimized set of parameters was selected based on RSME from the tuning grids
7. The model was retrained utilizing the optimized parameter set.
8. Performance metrics were collected for the optimized model on the withheld testing data.

Model Tuning

For the KNN, XGBoost, and Random Forest models, a primary tuning grid was evaluated using 5-fold cross-validation. Where appropriate, a secondary tuning grid was also used to focus on critical parameter spaces. The optimized set of parameters was selected based on RSME from the tuning grids. Tuning grid figures are supplied in the Appendix [Additional Figures]. Model parameters and descriptions are listed in the Appendix [Tuning Parameters].

The random model was tuned using a grid search and evaluated with 5-fold cross-validation. The random forest was evaluated with a tuning grid for mtry and min_n. A standard tree size of 1000 was selected, but 500 was also evaluated.

The xgboost model was tuned with a 6 parameters grid using 5-fold cross-validation. The following parameters were evaluated in the tuning grid. The L1 and L2 regularization terms are also evacuated within the tuning grid.

The KNN model was evaluated with different neighbor quantities, k, using a 5-fold cross-validation.

Analysis and Results

R was used for this analysis. The packages utilized were; tidyverse, tidymodels, gt, vip, vip, ggplot2, MASS, and glmnet. Code and data are available publicly at: https://github.com/rbkeeney1/Group-30-ISYE-7406.

Feature Engineering

We hypothesize that imputing missing data points in LotFrontage using a linear regression method from LotArea would improve this analysis (rather than the median value). However, this is beyond the scope of this analysis (comparison of variable selection techniques and models) and is not evaluated.

Variable Selection

LASSO is an ideal method in this scenario because there are few significant coefficients relative to the entire predictor set. Relative to AIC stepwise regression, BIC stepwise penalizes the model more for its complexity and is preferred for scenarios where there are few significant coefficients relative to the entire predictor set. In this case, both BIC stepwise linear regression and LASSO regression selected the same set of predictors, with a few minor expectations.

Variable Selection	
Model	Adj. RSQ
AIC Stepwise Linear Regression	0.937
BIC Stepwise Linear Regression	0.931
LASSO	0.932

Since the AIC stepwise linear regression model was not significantly different from the LASSO and BIC stepwise linear regression model, the sparse predictor set based on LASSO and BIC stepwise linear regression was utilized. A LASSO variable trace plot is supplied in the Appendix [Additional Figures].

Model Evaluation

The results from our proposed approaches are evaluated with Root-Mean-Squared-Error (RMSE) between the logarithm of the predicted value and the logarithm of the observed sales price.

In our analysis, the baseline model (Linear Regression) performed the best. Overall, house prices can be predicted with a high degree of confidence with all the models evaluated. The XGBoost, or random forest with gradient boosting, outperformed the random forest model slightly, while the KNN model performed the worst. The KNN model required a large number of neighbors to provide a reasonable estimation. Tuning grid figures are supplied in the Appendix [Additional Figures]. Model parameters and descriptions are listed in the Appendix [Tuning Parameters].

Model	**Parameters**	**RSME**	**RSQ**

Linear Regression	(none)	0.104	0.930
XGBoost	mtry = 3 trees = 1000 min_n = 9 tree_depth = 14 learn_rate = 0.0256089626570059 loss_reduction = 0.0218368033705335 sample_size = 0.837010128129041	0.116	0.914
Random Forest	mtry=15 trees=1000 min_n=4	0.126	0.907
KNN	k=21	0.151	0.866

Conclusions

In our analysis, the baseline model (Linear Regression) performed the best. Overall, house prices can be predicted with a high degree of confidence with all the models evaluated. The XGBoost, or random forest with gradient boosting, outperformed the random forest model slightly, while the KNN model performed the worst.

LASSO is an ideal method in this scenario because there are few significant coefficients relative to the entire predictor set. Since the AIC stepwise linear regression model was not significantly different from the LASSO and BIC stepwise linear regression model, the sparse predictor set based on LASSO and BIC stepwise linear regression was utilized.

The utilization of cross-validation and parameter tuning was critical to selecting the optimum parameter set for each model. There were many versions of the models that performed poorly, and a thorough grid search, with cross-validation, was able to significantly improve the performances of the models from their default parameters. While we completed our stated objectives, if given more time we would have liked to use a Neural Network model to predict the housing prices. This model has not been reviewed yet and we would like to explore and evaluate this model against our baseline model to see which model performs better. Even though the baseline model (Linear Regression) performed the best, this model can only learn the linear relationship between features and target thus not best for complex non-linear relationship systems. A neural network can help us to extract meaningful information and detect hidden patterns from the hidden layer in the dataset and can be used to estimate predictive models with minimal structure and assumption.

Lessons Learned

Ryan Brent Keeney

Throughout the course and project, I appreciated the focus on cross-validation for evaluating model performance. The reinforcement of this issue was helpful for me, as was the course information regarding Monte Carlo cross-validation. The other area of the course that I appreciated was the unit on local smoothers, such as LOESS. The HW for this unit was particularly well constructed. Overall, I found the coding in this course relatively easy, although I have a strong background in R and have utilized many of the models and methods this course teaches. I strongly recommend adding additional components to the course to support less experienced R users, such as leveraging the amazing framework available in the tidymodels package - this step alone would empower students to focus on model workflows and analysis rather be forced to learn the individual nuances of each package (ranger, glmnet, xgboost, random forest, etc.). Finally, The course content is technical in nature and a broader discussion and generalization of how the models or parameters work should be included. I found myself often disconnected from the derivations and needing to reference outside material just to grasp the basic methodology behind some of the modeling techniques we discussed.

Thang Sung Trieu

From this project, I learned that most of the dataset in the real world is not perfect. It requires a lot more data exploration for us to uncover any initial patterns, characteristics, and points of interest from the dataset. Data preparation it's not easy to handle or deal with if your dataset is not perfect, this process is time-consuming and requires a lot of testing in order for us to find what we're looking for. Model development and evaluation are not easy for me since I'm still learning R and have trouble with coding and testing the model, but luckily I have two wonderful teammates helping me through the issues. Overall, this course needs some improvements. The course lectures were not easy to follow and needed to have more hands-on experiences or real-world examples for students to be more engaged. In my opinion, adding a little more guidelines or structure to the homework and project will be helpful and ease any confusion.

Mohammed Salman Saeed

I appreciated the attempt to make this course more "real-world" by asking us to explain our work through presentations and interpretations rather than just code and do the math. This is an under-appreciated skill but critical to implementing any model. The recommended structure of the homework forced me to go beyond just answering the question. It made me think from the perspective of the reader and what questions they may have on the analysis. This thought process was invaluable because an analyst should not just be able to develop and design great models, but be able to defend them and sell them to a skeptical audience. Having said that, I would recommend the lecture material be enhanced to include more real-world examples. Although the majority of the work is done in R for this course and the starter code is very helpful, I'd also recommend to provide starter code in Python to enable students to more freely choose which language they will complete the assignments on.

Appendix

Data Source

The Ames Housing dataset was compiled by Dean De Cock for use in data science education, it was accessed at Kaggle.com [1].

[1] House Prices - Advanced Regression Techniques. (n.d.). Kaggle.com. Retrieved March 17, 2022, from https://www.kaggle.com/c/house-prices-advanced-regression-techniques/overview

[1A] Data Field Descriptions - Advanced Regression Techniques https://www.kaggle.com/competitions/house-prices-advanced-regression-techniques/data?select=data_description.txt

Literature Review

Variable Selection

[2] sauravkaushik8. (2015, December 1). Introduction to Feature Selection methods with an example (or how to select the right variables?). Analytics Vidhya. Retrieved March 16, 2022, from https://www.analyticsvidhya.com/blog/2016/12/introduction-to-feature-selection-methods-with-an-example-or-how-to-select-the-right-variables/

[3] Frost, J. (n.d.). Identifying the Most Important Independent Variables in Regression Models. Statistics By Jim. Retrieved March 16, 2022, from https://statisticsbyjim.com/regression/identifying-important-independent-variables/

[4] Schmarzo, W., & Vantara, H. (2017, February 2). Identifying Variables That Might Be Better Predictors. KDnuggets. Retrieved March 16, 2022, from https://www.kdnuggets.com/2017/02/schmarzo-variables-better-predictors.html

Feature Engineering

[5] Ialenti, A. (2021, February 17). How data science can help you find your ideal house at an affordable price. thenextweb. Retrieved March 16, 2022, from https://thenextweb.com/news/how-data-science-help-you-find-house-at-affordable-price-syndication

Cross-validation methods

[6] Cross-validation: evaluating estimator performance. scikit-learn. (n.d.). Retrieved March 16, 2022, from https://scikit-learn.org/stable/modules/cross_validation.html

[7] Bose, A. (2019, January 30). Cross Validation — Why & How. towardsdatascience. Retrieved March 16, 2022, from https://towardsdatascience.com/cross-validation-430d9a5fee22

Visualization

[8] Liu, Z. (2016, October 25). Insights on Housing Data: Multiple Factors behind House Price. nycdatascience. Retrieved March 16, 2022, from https://nycdatascience.com/blog/student-works/key-insights-ames-iowa-housing-data-multiple-factors-behind-house-price/

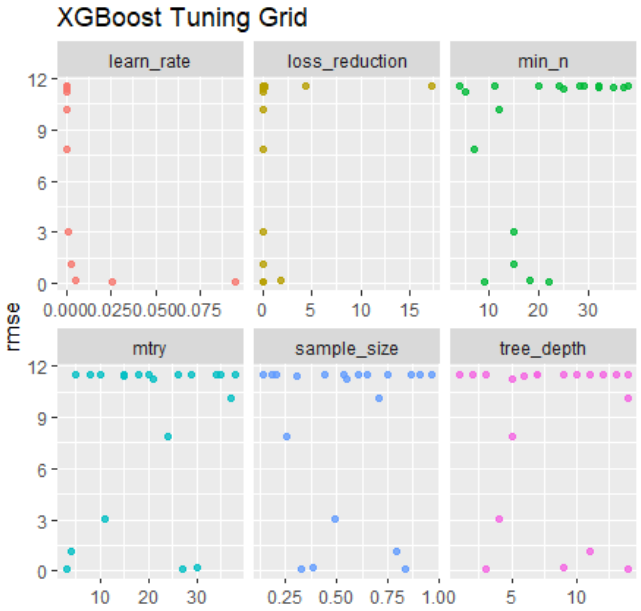
Multicollinearity

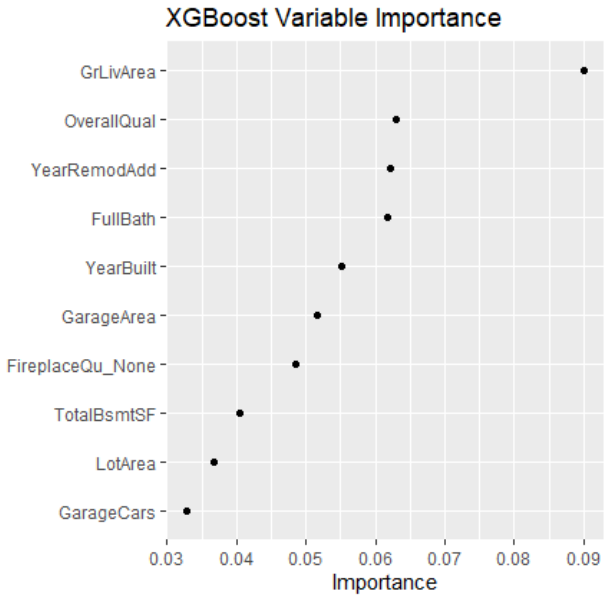
[9] GOYAL, C. H. I. R. A. G. (2021, March 19). Multicollinearity in Data Science. analyticsvidhya. Retrieved March 16, 2022, from https://www.analyticsvidhya.com/blog/2021/03/multicollinearity-in-data-science/

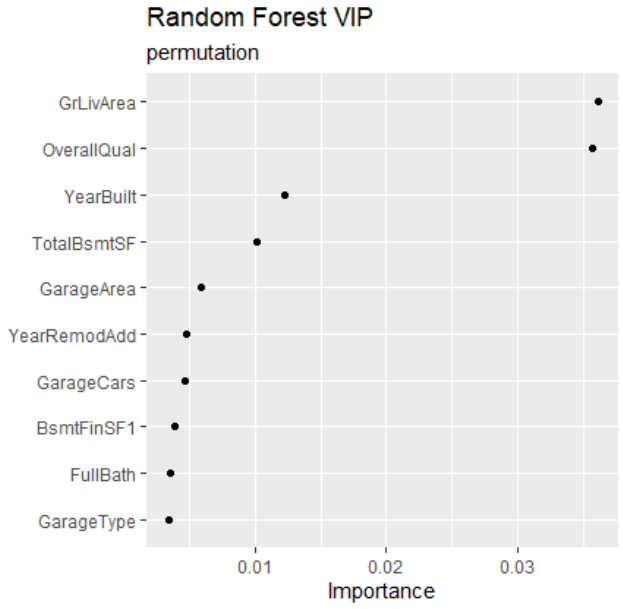
Principal Component Analysis

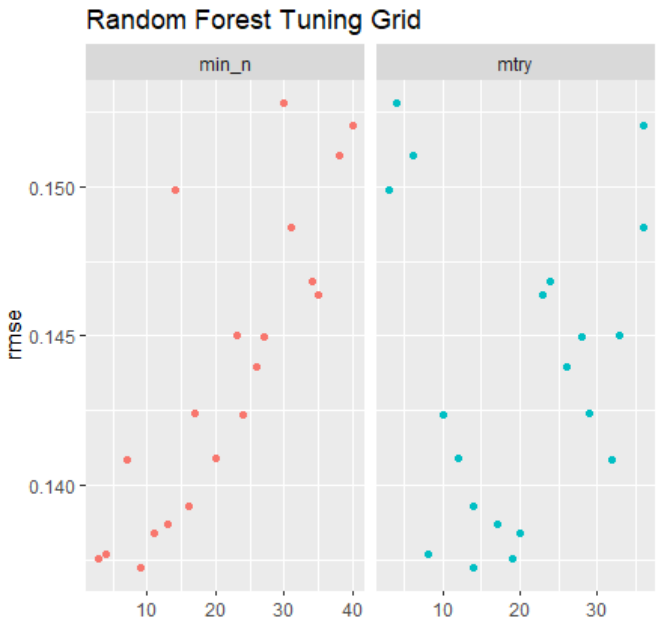
[10] Li, Lorraine (2019, May 25). Principal Component Analysis for Dimensionality Reduction. Retrieved March 18, 2022 from https://towardsdatascience.com/principal-component-analysis-for-dimensionality-reduction-115a3d157bad

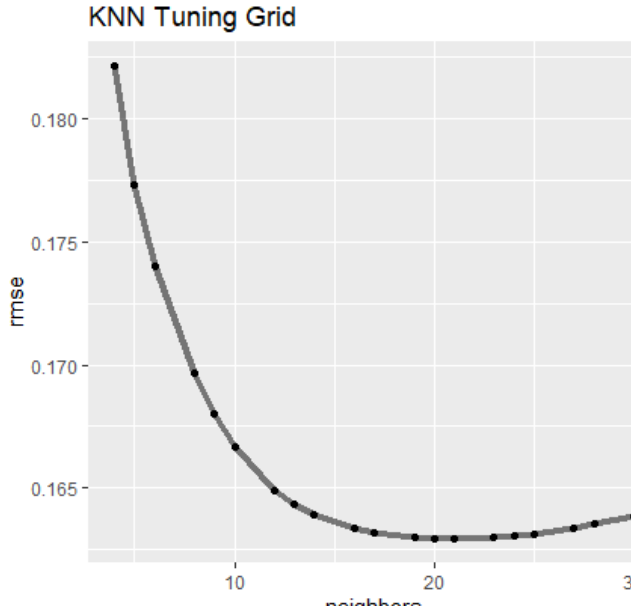
Additional Figures

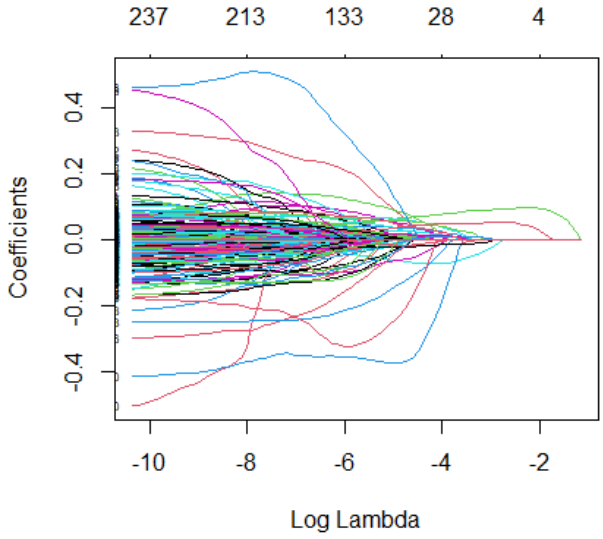



XGBoost Variable Importance
GrLivArea
OverallQual
YearRemodAdd
FullBath
YearBuilt
GarageArea
FireplaceQu_None
TotalBsmtSF
LotArea
GarageCars
0.03
0.04
0.05
0.06
0.07
0.08
0.09
Importance


Random Forest VIP
permutation
GrLivArea
OverallQual
YearBuilt
TotalBsmtSF
GarageArea
YearRemodAdd
GarageCars
BsmtFinSF1
FullBath
GarageType
0.01
0.02
0.03
Importance


Random Forest Tuning Grid
min_n
mtry
rmse
0.150
0.145
0.140
10
20
30
40
10
20
30


KNN Tuning Grid
rmse
0.180
0.175
0.170
0.165
10
20



Tuning Parameters

Random model

The random model was tuned with a grid including 7 parameters using 5-fold cross-validation. The random forest was evaluated with a tuning grid for mtry and min_n. A standard tree size of 1000 was selected, but 500 was also evaluated.

Mtry	An integer for the number of predictors that will be randomly sampled at each split when creating the tree models.
Trees	An integer for the number of trees contained in the ensemble.
min_n	An integer for the minimum number of data points in a node that is required for the node to be split further.

Boosted Tree

The xgboost model was tuned with a 6 parameters grid using 5-fold cross-validation. The following parameters were evaluated in the tuning grid. The L1 and L2 regularization terms are also evacuated within the tuning grid.

Mtry	An integer for the number of predictors that will be randomly sampled at each split when creating the tree models.
Trees	An integer for the number of trees contained in the ensemble.
min_n	An integer for the minimum number of data points in a node that are required for the node to be split further.
Tree depth	Maximum depth of nodes within each tree
Loss reduction	Boosting loss reduction
Learning rate	Boosting learning rate
Sample Size	Minimum number of samples to split a node*

KNN

The KNN model was evaluated with different neighbor-quantities, k, using a 5-fold cross-validation.

Code

Available at: https://github.com/rbkeeney1/Group-30-ISYE-7406