Analysis of Ames Housing Market

Group 5: Brandon Leff, Dehui Hou, Xiuxiao Hughes, Lulu Xue

1 Introduction

The house market is an integral aspect to our society. Analyzing the housing market in order to see price trends is important to buyers, sellers, businesses such as real estate companies, policy makers and so on. There are various studies in the literature to serve the needs of different stakeholders. For example,

1. Federal Reserve Bank Chicago [4] used linear models to study the major factors impacting house price in Cook County. They focus on understanding the impact of distressed sales to the house price in the close proximity. The result has implication on economical and financial policies.
2. More than 100 real estate technology companies are using machine learning and artificial intelligence to model the housing market. They have significant impact to the house price and therefore impact the affordability and the investment worthiness of the houses [6].

Given the importance of the house price prediction problem, it is interesting to apply what we learned in class to this problem. The goals of this project include:

1. Simulate a real-life data science problem and see if we could analyze the housing market using the methods we learned from the class. During this process, we expect to enhance our understanding of the models and practice our skills in applied data analysis.
2. Use the models to tell us what aspects of a house are important for predicting the sale price of the house. During this process, we expect to use models to help us improve our understanding of the housing market and potentially discover new knowledge about the house price.

To do so, we obtained a data set about the housing market in Ames, Iowa from Kaggle [1]. The data set's original purpose was for an advanced regression competition on Kaggle so we decided this would be a great data set to use. The data set consisted of 1460 samples and 79 features. 47 of the 79 features were categorical with varying amounts of classes. Each row of the data set represented a different house that was sold in Ames, Iowa during the years of 2006 through 2010. Each feature represented different aspects of the house varying in meaning from area, size, location, and the presence of specific amenities.

The remainder of this report is organized as follows. We first introduce the notations in Section 2, followed by an introduction to the methods in Section 3. In Section 4, we analyze the data and perform necessary transformations to prepare it for the modeling stage. The modeling results are discussed in Section 5. A conclusion is provided to summarize the main findings of this report in Section 6.

2 Notation

In this paper, we use a series of mathematical equations to explain our modeling and analysis. First regarding dimensionality of the data, we define n to be the number of samples from the data and define p to be the number of features.

When referring to penalized regression models LASSO and Ridge, λ is defined to be a constant $\in \mathbb{R}$ that defines the amount of penalty used in the regression model. If set to $= 0$, the penalized regression reverts to the Ordinary Least Squares Regression problem.

When referring to penalized regression model Elastic Net, L_1^r is defined as a constant $\in [0,1]$ representing a ratio that controls the proportion of L_1 and L_2 penalty used.

X is set to be a $n \times (p-1)$ matrix of dependent variables, Y is set to be a $n \times 1$ vector of the independent variable, and $\hat{Y}$ is set to be a $n \times 1$ vector of the predicted values from the respective model. β is defined as a $p \times 1$ vector of coefficients from the respective model.

3 Methods

3.1 Regularized Linear Models

Here we focus on three regularized linear models, including LASSO, Ridge and Elastic Net. They are all designed based on linear regression, whose fitting is a straight line. Compared with ordinary linear regression, regularized model handles the issues like multicolinearity and noise features better. Especially when the number of features are high, regularized models are less prune to overfitting issue and reduce the model variance. However, on the other hand, the penalization biases the model. The penalty strength could be used to find the best balance between bias and variance.

1. LASSO

   In this project, one of the 3 forms of penalized regression we use is LASSO. LASSO utilizes the L_1 penalty in order to allow some coefficients to shrink to exactly 0. This means that LASSO can be seen as a subset selection method and can be used for performing variable selection. The equation for LASSO is shown below.

   $$\min_{\beta \in \mathbb{R}^p} \left\{ \|Y - X\beta\|_2^2 + \lambda \|\beta\|_1 \right\}$$

   where $\|\beta\|_1 = \sum_{j=1}^{p} |\beta_j|$.

2. Ridge

   In this project, one of the 3 forms of penalized regression we use is Ridge. Ridge regression utilizes the L_2 penalty in order to shrink the coefficients close to 0. The equation for Ridge is shown below.

   $$\min_{\beta \in \mathbb{R}^p} \left\{ \|Y - X\beta\|_2^2 + \lambda \|\beta\|_2^2 \right\}$$

   where $\|\beta\|_2^2 = \sum_{j=1}^{p} \beta_j^2$

3. Elastic Net

   In this project, one of the 3 forms of penalized regression we use is the Elastic Net [2]. Elastic Net utilizes both the L_1 and L_2 penalty allowing coefficients to shrink both close to and exactly 0. This method can either be defined using two λ values λ_1 and λ_2 which measure the amount of the L_1 and L_2 penalty respectively or also with some proportion $L_1^r \in [0,1]$ where L_1^r measures the amount of L_1 penalty used and $(1 - L_1^r)$ measures the amount of L_2 penalty used. When referring to the Elastic Net method later in the paper, the method using the L_1 Ratio is used. The equation for Elastic Net is shown below.

   $$\min_{\beta \in \mathbb{R}^p} \left\{ \|Y - X\beta\|_2^2 + L_1^r \|\beta\|_1 + (1 - L_1^r) \|\beta\|_2^2 \right\}$$

   where $\|\beta\|_1 = \sum_{j=1}^{p} |\beta_j|$, and $\|\beta\|_2^2 = \sum_{j=1}^{p} \beta_j^2$

   Since in this project we wanted to compare the performance and variable importance between these 3 different penalized regression models, we opted to keep L1 ratio constant at 0.5 effectively perfectly balancing the proportion of the L_1 and L_2 penalties. This was also decided as the model was always selecting the smallest value in the grid search for L1 ratio effectively always preferring LASSO, so to draw insight on the differences, the decision to keep L1 ratio constant at 0.5 was made.

3.2 Model Evaluation Methods

The data with known sale price is divided into two part:

- 70% of the data is used for training and validation. We will perform 10-fold cross-validation on this data to find the best parameters and hyper-parameters.
- 30% of the data is used for testing, so we don't use it in any way in the model building and parameter tuning steps. And it is only used to evaluate and compare model performance after we have all parameters and hyper-parameters determined.

Note: in the real Kaggle competition, all the above mentioned data can be used to tune hyper-parameters. There is a separate blind test data set for the final model comparison between competing teams.

3.3 Model Performance Evaluation Metrics

When training the models, we perform log transformation on the house price as the target variable. However, for model performance evaluation, we first transform the prediction back into the original dollar amount scale and then calculate the metrics listed here. In this way, the metric is easier to understand.

Two metrics are used to compare the algorithms:

1. Root Mean Squared Error (RMSE)

   We choose RMSE over MSE mainly because RMSE is in the original unit of the target variable. That is, RMSE in measured in dollars, which is easier to understand.

   $$RMSE = \sqrt{\frac{1}{n}\sum_{i}^{n}(\hat{Y}_i - Y_i)^2}$$

   where $\hat{Y}$ are the predicted y-values of the corresponding model and Y are the ground truth values.

2. Mean Absolute Percentage Error (MAPE)

   Percentage error is more intuitive for our problem. Because for example, a 10 thousand dollar error for a one million house is perceived as smaller than a 10 thousand dollar error for a 100 thousand house. Among the various metrics proposed in the literature to address this issue, MAPE [5] is the most widely used and easiest to explain metric.

   $$MAPE = \frac{1}{n}\sum_{i}^{n}\frac{|\hat{Y}_i - Y_i|}{Y_i}$$

   where $\hat{Y}$ are the predicted y-values of the corresponding model and Y are the ground truth values.

3.4 Feature Ranking Methods

In this project, we use the magnitude of the coefficient to rank the features. Since we will normalize the features, the coefficients of different features are comparable.

To handle the model coefficient stability issue (see Section 5.3), we repeat the model building and hyper-parameter tuning for a random split of the training data 100 times and then use the average coefficient to rank the features.

4 Real Data Study

In this section, we first clean the data to ensure a high data quality. Then we go through several data analysis and feature transformation steps to prepare the data for the proposed linear models.

4.1 Data Cleaning

4.1.1 Missing Data Processing

There are several types of missing data. We process them differently. For missing values, after observing for each variable, we find that not all missing values are meaningless. Some were missing by random and some were missing for a reason. For missing by reason, an example was for the variable poolQC. An NA in category poolQC, meant no pool for the given house. There were many variables in the data set like this where NA actually represented the absence of that amenity in the home. Moreover, for missing values missing by random, we filled them in with median values of the corresponding variables.

4.1.2 Outliers

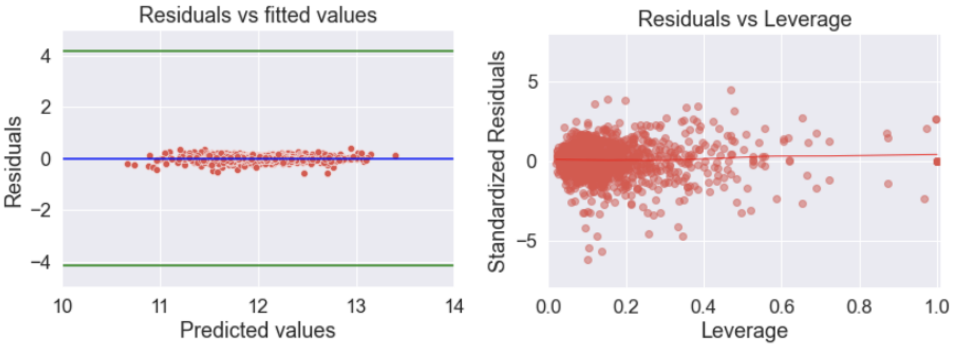


Figure 1: Identify outliers that do not follow the fitted models.

- Outlying in Y

  As shown in the left panel of Figure 1, cases outlying in Y can be identified through residuals. The two green lines in the Residuals vs. fitted values plot are the critical values of Bonferroni's procedure at the alpha = 0.05. It was calculated as $t(1-\frac{\alpha}{2n}, n-p-1) = 4.1586$. Where n is the number of observations and p is the number of the predictors. Since all points are inside the boundary of the critical values there are no outliers in Y.

- Outlying in X

  From the plot residuals vs. leverage plot in the right panel of Figure 1, we don't see any obvious point is far away from the crow. There is no point at the right top corner, so there is no obvious outlying in X.

4.2 Model Assumption Check

We observed that the data had rough linear relations so a linear model may be a good choice. To check, we verify the assumptions for linear models.

- Normally Distributed Dependent Variable

  As shown in Figure 2, the original distribution of the dependent variable is right skewed (left image). We did a log transformation of the dependent variable so that it is normally distributed (right image).

- Linearity

  The Residuals vs. predicted values plot as shown in the left panel of Figure 3 does not show a clear nonlinear pattern. It is an indication that the linearity assumption is met.

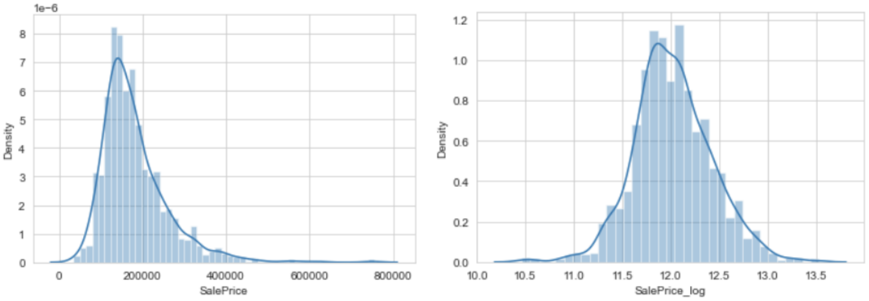


Figure 2: Identify outliers that do not follow the fitted models.

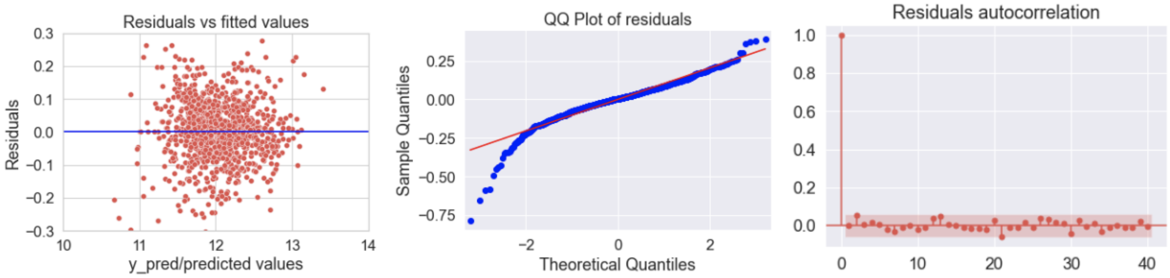


Figure 3: Identify outliers that do not follow the fitted models.

- Normality of Error Terms

  The QQ plot in the middle panel in Figure 3 shows there is a little heavy tail on the left, but most of the points are forming a straight-line pattern which indicates that the errors are likely from a normal distribution.

- Independence of Error Terms

  The autocorrelation plot on the right panel in Figure 3 above shows that the residuals are not correlated to each other. It means that the errors are independent from each other.

- Constant Variance of Error Terms

  We used the Goldfeld Quandt test for homoscedasticity. The null hypothesis is that the variance of the error terms is equal. The alternative hypothesis is the variance of the error terms is not equal. The Goldfeld Quandt test result has an F-score equal to 0.56265. The p-value is 0.99999. Since p-value is more than 0.05, we failed to reject the null hypothesis that the variance of the errors is equal. The assumption of constant variance is met.

- Multicollinearity

  VIF values greater than 10 are often taken as an indication of high multicollinearity. The VIF values from Table 1 show a few high VIF values from our data set. The VIF values equal to infinity indicate perfect correlation. We have more than 50 VIF values larger than 10. Due to this severe multicollinearity, the model estimates tend to have larger variance. We can fit regressions with penalized terms to add a degree of bias to lower the variance so that the model can achieve a good bias-variance trade-off. In this project, the regressions with penalized terms that we use are Ridge, Elastic Net and LASSO regressions.

Feature	VIF
BsmtQualNotExist	∞
BsmtFinSFRec	∞
GarageFinishNotExist	∞
YearBuilt	9.142005e+04
PoolQCNotExist	4.151891e+04
GarageYrBlt	3.547780e+04

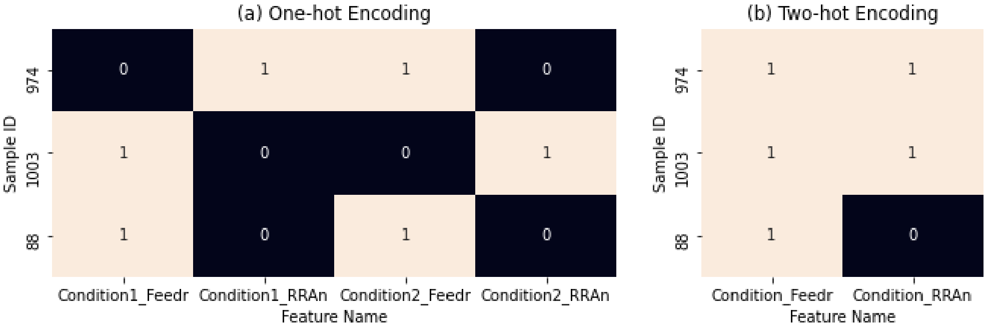
Table 1: Variance Inflation Factors for Variables

4.3 Feature Engineering

4.3.1 Categorical Variable Encoding

We could use one-hot encoding to represent most categorical variables. Since this is a standard approach, we will not describe in detail here.

However, there are a few categorical variables need to be processed using two-hot encoding (a special instance of the so-called multi-hot encoding).



Sample ID	Condition1_Feedr	Condition1_RRAn	Condition2_Feedr	Condition2_RRAn
974	0	1	1	0
1003	1	0	0	1
88	1	0	1	0

Sample ID	Condition_Feedr	Condition_RRAn
974	1	1
1003	1	1
88	1	0

Figure 4: Two-hot Encoding vs. One-hot Encoding.

Figure 4 use a real data example to show the difference between one-hot encoding and two-hot encoding. This is related to two columns named *Condition1* and *Condition2*. They represent two conditions the house has. For example, if the house is on a feeder road, one of the two conditions will be set to *Feedr*. If we use one-hot encoding, we will have two columns representing *Condition1* is *Feedr* and *Condition2* is *Feedr*. However, the meaning is the same no matter which condition column we write the *Feedr* in. So a better approach is to only use one feature *Condition_Feedr* to represent if *Feedr* is presented in either of the two condition columns. Similarly, for the condition of *RRAn*, we only need one column to encode it in two-hot encoding.

We also performed two-hot encoding for *Exterior1st* and *Exterior2nd*, because we found there is no logical difference between 1st and 2nd here.

4.3.2 Nonlinear Transformation

There are features that have a clear nonlinear relation with the target house price. As shown in Figure 5, there is a nonlinear relation between the feature *YearBuilt* and the target variable *SalePrice*. We add square of *YearBuilt* as a new feature into the data so that the linear models can represent nonlinear relation with respect to *YearBuilt*.

Similarly, we also transformed *MoSold*, which represents the month the house was sold. We considered this variable as a continuous variable and try to use a simple quadratic transformation to capture some periodical effects of this feature. In practice, we may want to try cosine / sine transform to better represent the periodical effects.

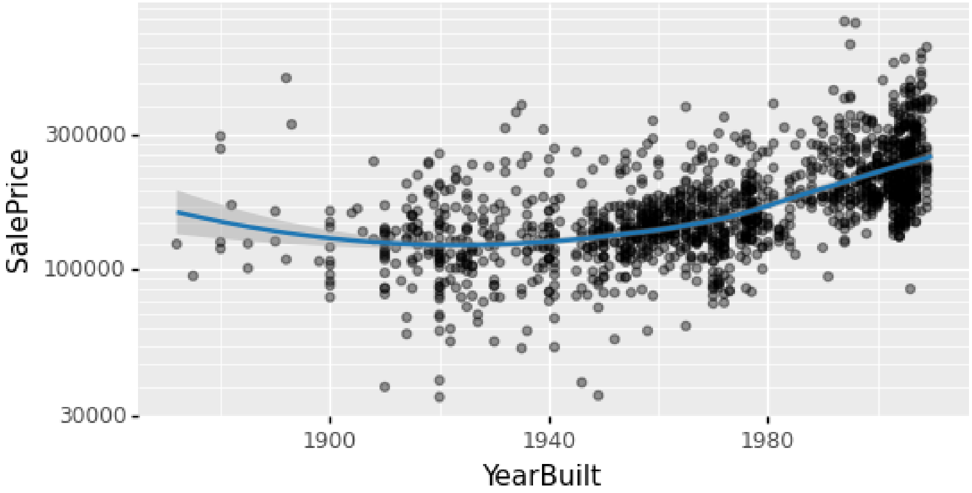


Figure 5: The nonlinear relation between the year the house was built and the sale price of the house. Sale price is shown in log scale.

4.3.3 Interactions

Several interaction terms are added based on our empirical knowledge. The interaction terms are listed in Table 2. We are not sure if these interaction terms are useful to the model, but will let the regularized models to select useful ones.

Feature 1	Feature 2	Meaning of the Interaction Term
BedroomAbvGr	GarageCars	More people in the house needs more garage space for cars or storage.
BedroomAbvGr	FullBath	Living more comfortable with more bedrooms and bathrooms.
GrLivArea	LotArea	Additional benefits of big outdoor area and big indoor living area.

Table 2: List of feature interactions considered in the models.

4.4 Normalization

We use normalization because of two reasons:

- Numerical stability and convergence speed of the regression algorithms. When the variance of the variables are very different, some of the algorithms that makes use of inverse of pseudo inverse of matrix may face numerical stability issues. And for some of the algorithm that based on gradients may have slow convergence rate issues. Normalization may help address these issues.

- Interpretability of the model coefficients. The impact of a coefficient may be magnified by the feature if the feature values are very large. Normalizing the features may help us to interpret the coefficient as the impact of one standard deviation change in the corresponding variable to the target variable.

The normalization is performed as:

$$x_n = \frac{x - \bar{x}}{std(x)} \tag{1}$$

where, x is the feature to be normalized, x_n is the feature after normalization, $\bar{x}$ is the mean of x, and $std(x)$ is the standard deviation of x.

5 Modeling Result

The final goal of this project is to understand the important features that drive the house price. There are several necessary intermediate steps to perform before achieving the final goal:

- Ensure the models are accurate so that we can trust the coefficients in the models. We achieve this in Section 5.1.
- Verify that the redundant features will have small or even zero coefficients to ensure the best features are selected to interpret. Particularly, the verification tells us which of the three models we use for the purpose of interpreting the features. This step is discussed in Section 5.2.
- Verify that the coefficients are stable so that all the selected features are reliable. This is described in Section 5.3.

After all the above steps, we will list the important features and interpret the results. There are two aspects in the interpretation:

- The top features are the ones we expected. We will discuss them in Section 5.4.1
- For unexpected results, we will analyze to see if it's due to the limitation of the algorithm or due to the limitation of our knowledge (i.e., discovered new knowledge). This is discussed in Section 5.4.2

5.1 Model Performance Analysis

Model	RMSE (thousand dollars)	MAPE
LASSO	32.68	9.73%
Ridge	31.8	10.19%
Elastic Net	30.83	9.61%

Table 3: Model performance comparison.

We only focus on the model performance in this subsection. As shown in Table 3, all three methods work reasonably well with RMSE around 31 to 32 thousand dollars and a absolute percentage error around 10%. The elastic net has the best performance in terms of both metrics. It combines the benefits of LASSO and Ridge.

Here we emphasize more on MAPE, because we believe an error of 10k dollars for a 1 million dollar house is much less severe than an error of 10k for a 100k dollar house. In this sense, we conclude that:

- LASSO is more accurate than Ridge.
- Elastic net is the best in terms of model accuracy.

5.2 Coefficient Path Analysis

A very important difference between LASSO, Elastic Net and Ridge is how they shrink the coefficients. This will impact the way we interpret the feature rankings.

First, we use Figure 6 to show how the coefficients change when the regularization strength changes. It is observed that LASSO is able to shrink the coefficients to zero while Ridge is not. Elastic Net is in-between, that is, when the regularization strength is higher compared with LASSO, the coefficients are also shrunk to zero. This is in line with our understanding of the three algorithms.

From this observation, it seems both LASSO and Elastic Net are suitable for our purpose, i.e., understanding which features are more important to determine the house price.

However, there is another aspect we need to compare. How do the algorithms select features when two or more features are correlated.

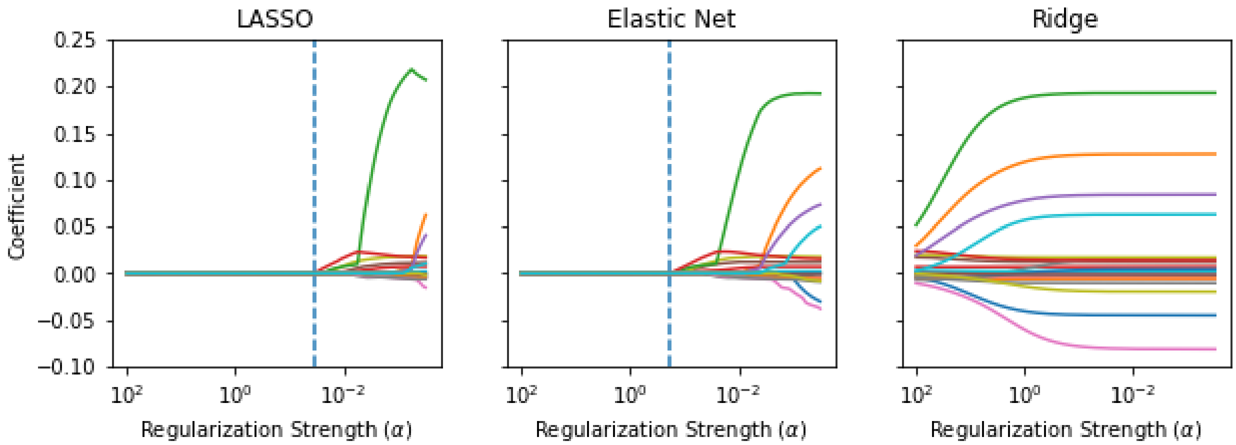


Figure 6: Comparison of Coefficient Path for Randomly Chosen 20 Features

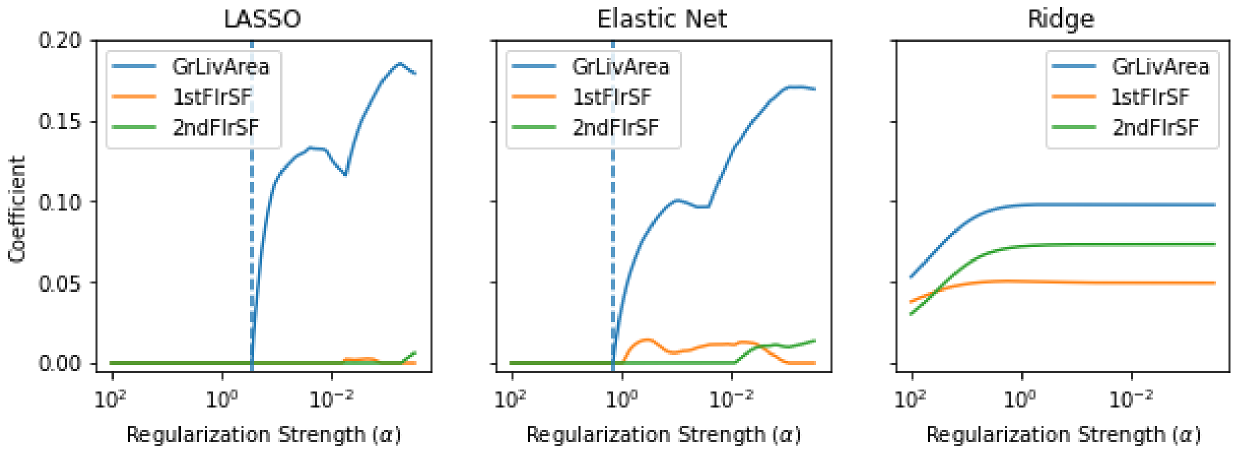


Figure 7: Comparison of Coefficient Path for Randomly Chosen 20 Features

	1st Floor Area	2nd Floor Area
Living Area	0.57	0.69

Table 4: The correlation between living area and the area of two floors.

We use a set of correlated features, including GrLivArea (the living area above ground), 1stFlrSF (the square feet of the 1st floor), and 2ndFlrSF (the square feet of the 2nd floor). The correlations are shown in Table 4. The coefficients paths for these features are shown in Figure 7.

It is observed from Figure 7 that:

- LASSO selects the total living area (GrLivArea) as the most important feature among the three area-related features and suppresses the coefficients of the other two features to zero (for most of the regularization strengths). We consider this as a desired property of LASSO, because the area for each floor does not provide too much additional information when we already know the total living area and therefore should not be selected.
- Elastic Net also selects the total living area as the most important feature among the three. However, it does not suppress the other two correlated features as hard as LASSO. Therefore the model is slightly less interpretable than LASSO.

- Ridge selects all three features, which is aligned with the observations for other features in Figure 6.

Based on the above observations, we decide to **use LASSO as the main method to generate feature ranking**. At the same time, we also confirm the feature ranking with the one generated by Elastic Net.

5.3 Feature Coefficient Stability

While it may seems sufficient to just run the LASSO once to generate a feature ranking, our experience in this project suggests the feature coefficients are not stable. Following this lead, we finally identified a reference that theoretically discussed the relation between sparsity and stability.

In the following, we use our experience in this project to empirically show that LASSO is not stable in terms of feature ranking. And show that aggregating the results for multiple runs may alleviate the stability issue.

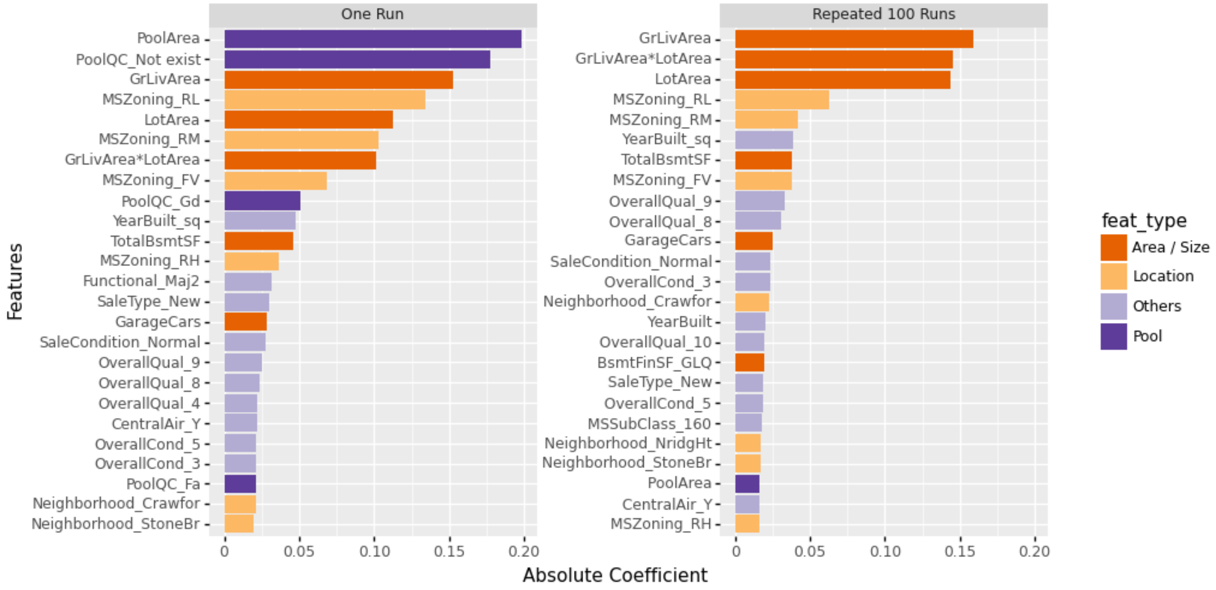


Figure 8: Feature Ranking Generated by Running LASSO Once and Running LASSO for 100 Times.

The left panel in Figure 8 shows the result we obtained when only run the LASSO once. The ranking is determined based on the absolute value of the coefficients. As observed, the features *PoolArea* and *PoolQC_Not exist* ranked as the top 2. These two features are not expected to be ranked so high because only a few houses (about 1%) have pools in Ames, IA. And the increase in price due to pool usually only takes small portion of the house price. So the ranking is not what we expected.

The right panel in Figure 8 shows the result we obtained by repeatedly running the algorithm 100 times, each time with different random split for training and testing data. The feature coefficients varies (See Figure 11 in the appendix for boxplots of coefficients), but the mean absolute coefficients show a more stable pattern of the relative ranking between features. The ranking looks more reasonable. We will try to interpret these features in Section 5.4. Here we mainly focus on the pool-related features that lead us to the investigation of feature ranking stability. More specifically, most pool-related features are not ranked in top 20. The highest ranking one is *PoolArea*, which ranked at 23rd place.

These observations empirically verifies that LASSO does not generate stable feature ranking result. A literature search suggests that Ref. [7] have investigated this topic and have provided a theoretical proof to show that models that generate sparse coefficients are less stable than models that generate dense coefficients. There is a trade-off between sparsity and stability.

Based on both the theoretical results in the literature and the empirical results in this project, we will **use the repeated 100 runs of LASSO results to rank and analyze the top features in the following sections**.

5.4 Interpreting the Feature Importance

In this Subsection, we interpret the top features that are ranked higher as expected. And we also try to explain why an expected top feature does not ranked to the top by the algorithm.

5.4.1 Top Features that Align Well with Our Understanding of the Housing Market

The interpretation is based on the right panel of Figure 8. We summarized the top 20 features into 3 categories as shown in Table 5. Roughly speaking, the features that represent house size and usable area are mostly ranked in the top. The features related to the location of the house is ranked slightly lower than area / size features but also in the top. Other top features are more or less related to the quality of the house, including the age of the house, quality description and etc.

Feature Type	Feature List
Area / Size	GrLivArea, LotArea, Basement Size (TotalBsmtSF, BsmtFinSF), Garage size (GarageCars)
Location	MSZoning, Neighborhood
Others	YearBuilt, OverallQual, SaleCondition, SaleType, MSSubClass

Table 5: The top features categorized into 3 types.

These top features look reasonable based on our experience. We confirmed the feature ranking in two ways. First, we compare with the ones generated by Elastic Net. Second, we confirm the top features with a few studies in the literature.

For Elastic Net, we run the model over 50 random states, we used the metric of mean absolute coefficient to measure the importance of variables on each model. The results were that the top 5 important variables were the interaction term between living square footage with lot area, living square footage, lot area, and zoning classification levels of Residential Medium Density and Residential Low Density. All of these factors align with our intuition that the most important variables of house price are size of the house, size of the lot, as well as the type of neighborhood. The top features aligned well with the LASSO result, and there are only minor difference in the ranks of a few features.

For comparison with the literature, we identified two sources that listed the top features. They are introduced in the following.

The Federal Reserve Bank in Chicago performed a study [4] using the house price in Chicago and showed that:

- The square feet of the living area and the lot size are the two most important features among the structure characteristics of the houses. Our result also showed the area / size features are the most important ones.
- It also highlighted the adjacency to city center, waterfront and Lake Michigan are important impacting factors. This is similar to what we discovered in our data that the neighborhood and zoning (which reflect the location of the house in Ames) are top features.

OpenDoor is a company in the housing market. Its study [3] suggested that the top three important factors include:

- The comparable homes in the same neighborhood.
- The location.
- The home size and usable space.

Since our model is trying to use various features to predict the house price, it inherently considered the comparable homes in the same neighborhood. The other two factors reported by OpenDoor are the same as we discovered from our feature ranking.

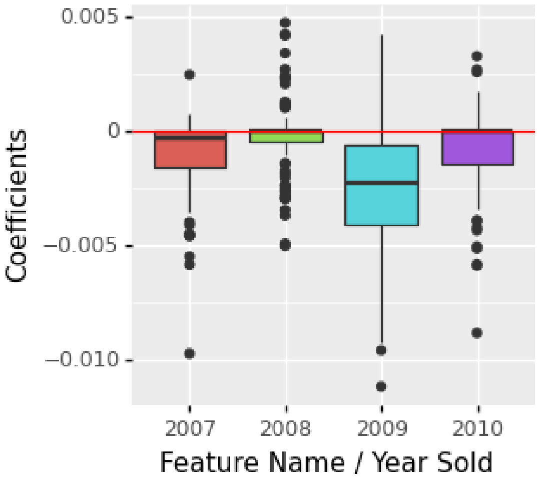


Figure 9: The Coefficients for Year Sold Features.

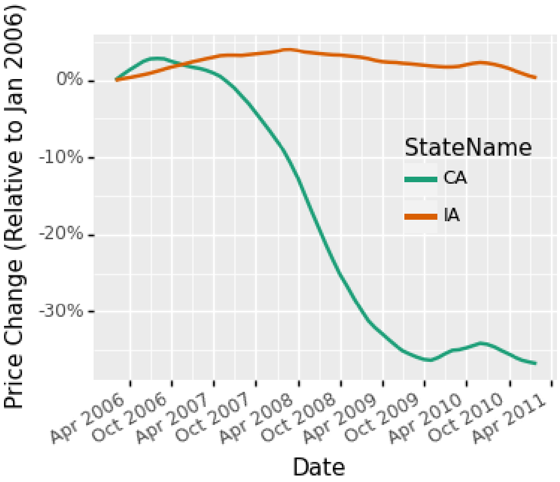


Figure 10: Zillow Price Index for CA and IA.

5.4.2 Important Features that Are Not Ranked In the Top

While most features are ranked as expected, we have one concern about the feature *YearSold*. Based on our understanding, this feature should be very important, because the house price usually change a lot from one year to another. Particularly, the data covers the year 2008. We expect a significant drop in house price in 2008. However, the feature *YearSold* is not ranked very high.

As shown in Figure 9, the coefficients are mostly around 0 and are small compared with the coefficients for other features like living area, zoning, neighborhood, and etc. This means, one of the following could be the cause:

- The algorithm is not working as expected and therefore we should not use the feature ranking, or
- We have the wrong knowledge regarding the importance of the feature *YearSold*.

Here, we validated with another data source called Zillow Price Index available at zillow.com [8] to confirm that our knowledge about the importance of *YearSold* is not accurate. So the result enhances our believe that the algorithm is correct in ranking the feature *YearSold* low.

As shown in Figure 10, the Zillow house price index for California dropped a lot during 2008. This is in line with our general impression that the house price dropped a lot during the financial crisis in 2008. However, the price in Iowa did not change too much. This is potentially because the economy in Iowa, which is mainly related to agriculture, was not impacted too much by the financial crisis.

From this observation, we believe our algorithm correctly ranked the feature *YearSold*. It may be an important feature in general scenario, but it is not an important feature for this location during these years.

6 Conclusions

This project analyzes a Kaggle competition data for house price prediction. The data contains all kinds of issues we would see in a real-world data set and therefore enabled us to practice our data science skills extensively.

We applied three regularized linear models, including LASSO, Ridge and Elastic Net, to predict the log sale price. The conclusion from the model comparison include:

- Model performance comparison suggests that Elastic Net combines the benefits of both LASSO and Ridge and achieves the best model accuracy among the three. The mean absolute percentage error is 9.61%.

- Coefficient path comparison confirms our understanding that LASSO generates sparse models while Ridge does not. Elastic Net again provides a balance between the two models.
- The good model performance ensured us that we can safely interpret the feature ranking generated from these models.

Feature rankings are used to enhance our understanding of the domain knowledge. We generated stable feature rankings by repeated runs of the model training algorithm using different data splits. The feature ranking result suggests that:

- The living area, lot area, zoning and neighborhood are the top features that impact the house price. These features are as we expected.
- Unexpectedly, the year sold is not ranked as top features. We verified from another data source and confirmed our ranking algorithm is working correctly to rank the year sold feature very low.

Overall, we successfully answered the question regarding the most important features for house price prediction. And our careful analysis ensured the correctness of the results.

7 Reference

[1] House prices - advanced regression techniques. (https://www.kaggle.com/competitions/house-prices-advanced-regression-techniques/).

[2] J Brownlee. How to develop elastic net regression models in python. machine learning mastery (https://machinelearningmastery.com/elastic-net-regression-in-python/). 2020, June 11.

[3] Joe Gomez. 8 critical factors that influence a home's value. *OpenDoor.com*, 2019.

[4] Jin Man Lee Maude Toussaint-Comeau. Determinants of housing values and variations in home prices across neighborhoods in cook county. *Profit Wise*, 2018(1):1–24, 2018.

[5] Chris Tofallis. A better measure of relative prediction accuracy for model selection and model estimation. *Journal of the Operational Research Society*, 66(8):1352–1362, 2015.

[6] Jennifer Conway Viriato. Ai and machine learning in real estate investment. *The Journal of Portfolio Management*, 45(7):43–54, 2019.

[7] Huan Xu, Constantine Caramanis, and Shie Mannor. Sparse algorithms are not stable: A no-free-lunch theorem. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(1):187–193, 2012.

[8] Zillow. https://www.zillow.com/research/data/. 2022.

8 Appendix

8.1 Supplemental Material

For this project, we used python through Jupyter Notebook. The links to the Github Repository that has our code files is below:

Github Repository

8.2 Feature Name and Meaning

Here we list the features that we mentioned in this report and explain their meaning. Please refer to Kaggle for a full list of feature name and meaning.

Feature Name	Meaning
GrLivArea	Above grade (ground) living area square feet
LotArea	Lot size in square feet
OverallQual	Overall material and finish quality
YearBuilt	Original construction date
Neighborhood	Physical locations within Ames city limits
MSZoning	The general zoning classification
GarageCars	Size of garage in car capacity
TotalBsmtSF	Total square feet of basement area
PoolArea	Pool area in square feet
PoolQC	Pool quality
CentralAir	Central air conditioning
OverallCond	Overall condition rating
SaleType	Type of sale
SaleCondition	Condition of sale
OverallCond	Overall condition rating
BsmtFinSF1	Type 1 finished square feet
MSSubClass	The building class
TotalBsmtSF	Total square feet of basement area
Functional	Home functionality rating

Table 6: Feature name and meaning

8.3 Feature Coefficients Boxplot

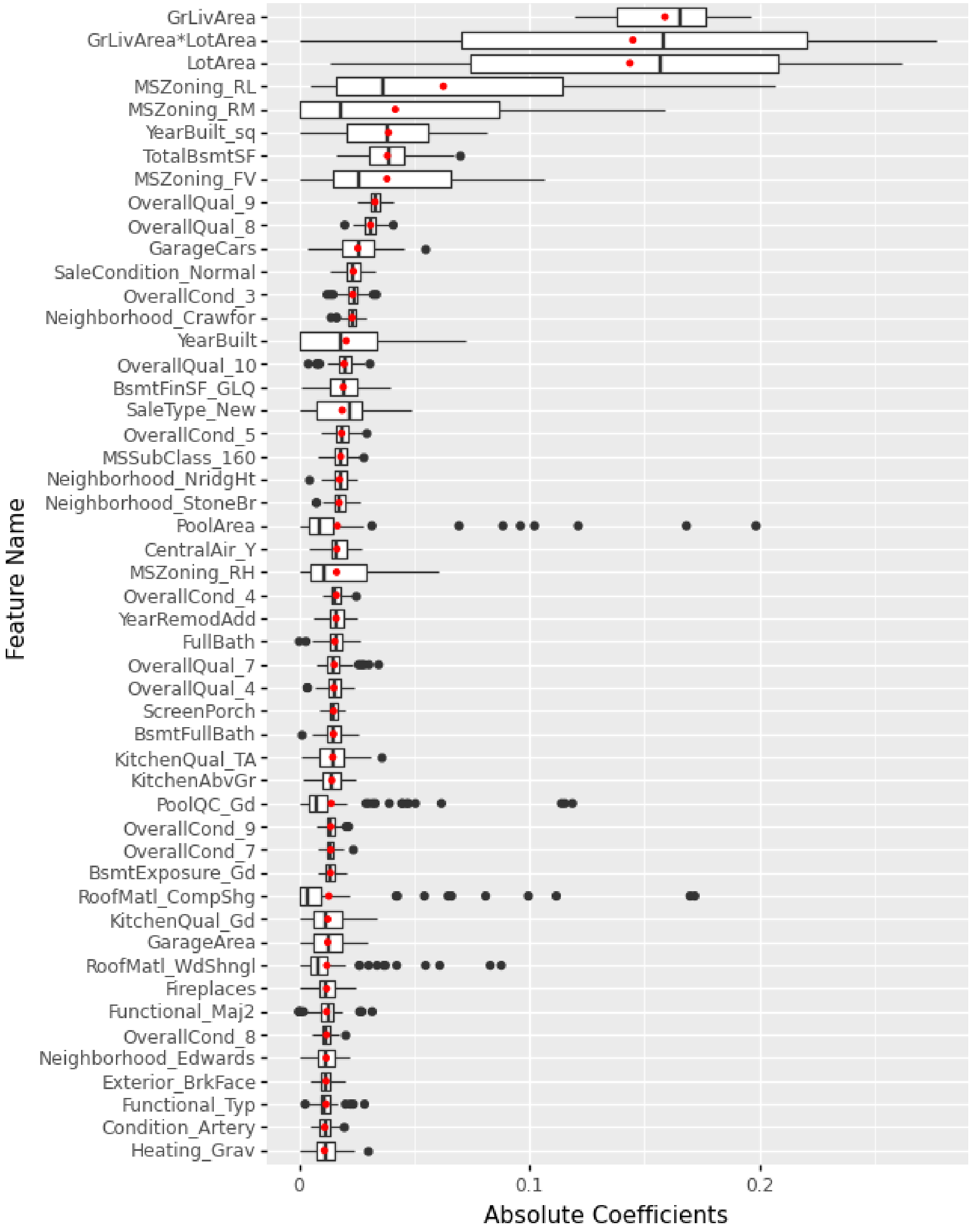


Figure 11: Feature Ranking Generated by Running LASSO 100 times. Boxplot shows the distribution of absolute coefficients. Red points show the mean of the absolute coefficients.