Washington University in St. Louis

James McKelvey School of Engineering

ESE 527 Final Report

Hanxin song 490260

Houdao Jiang 502831

Project Advisor: Dr. Patricio La Rosa

Title: Predictive analysis of the US housing price

Analytics Framework: Predictive Analytics

1. Exclusive summary

The objective of the project is to propose predictive analytics on the US. housing price. To approach our topic, we are planning to use Kaggle: Housing Prices Competition for Kaggle Learn Users data set that contains features affecting housing price and housing price. We will select the SalePrice as the target value and others will be independent variables in the model. The independent variables cover various features which impact the decision of customers to purchase a household at that price level.

By generating the predictive model, we will be able to impact the decisions of housing buyers and housing agencies. Hopefully, we can prevent buyers from frauds by our model. We have tested multiple algorithms for the model and the final model is built based on ANN.

1.1 Project Description

1.1.1 Motivation

The goal of the project is to help buyers determine the prediction price for housing based on the features input.

Our team noticed the increasing demand and price of housing under the pressure of Covid-19. As young adults who are about to move out of their parent's house, we are concerned about whether the money we paid for housing is reasonable. Therefore, we decide to build models with Machine Learning algorithms to perform predictive analytics on the house pricing dataset to help them make the decision.

1.1.2 Decisions to be impacted:

The number of house sales in the US. has generally grown since 2011. In 2020, 6.5 million houses were sold(Mac, 2021). The price of houses is usually determined by

estate agents, meanwhile, most people are not able to determine and understand the rationality of the price and market fluctuations.

Based on these observations, we aim to help buyers to get a better understanding of the housing price. By using our model, buyers can figure out if it's a suitable price for the houses they are concerned about.

1.1.3 Business/Societal value:

Societal Value: Housing is an essential part of people's lives, but the housing market has a lot of issues of opacity. By applying our model, people can protect their money from fraud and unreasonable prices.

Business Value: Our model can be applied to online housing platforms. The viewers of the website are more likely to convert to buyers after they ensure the rationality of the price.

2. Data Description and Preprocessing

2.1 Dataset Description

The data set we are using in this project is a Kaggle: Housing Prices Competition for Kaggle Learn Users data set. There are 80 features and 118,260 observations within the dataset. The target value we select is "SalePrice", and other variables will be independent variables that will be applied to the models.Table 1 is the quick view of the dataset.

The data set contains plenty of features of house facilities that impact people's decisions to purchase a household or not. For instance, the square foot of the entire house/amount of bedrooms/location of the household, etc.

Those factors have a significant impact on whether consumers buy a house. For instance, a family with many kids will seek a house with enough bedrooms. A single person might look for a house with a good location that may reduce daily commute time. Therefore, the dataset will help support the development of the analytics to impact your decision.

	Id	MSSubClass	MSZoning	LotFrontage	LotArea	Street	Alley	LotShape	LandContour	Utilities	...	PoolArea	PoolQC	Fence	MiscFeature	MiscVal
0	1	60	RL	65.0	8450	Pave	NaN	Reg	Lvl	AllPub	...	0	NaN	NaN	NaN	0
1	2	20	RL	80.0	9600	Pave	NaN	Reg	Lvl	AllPub	...	0	NaN	NaN	NaN	0
2	3	60	RL	68.0	11250	Pave	NaN	IR1	Lvl	AllPub	...	0	NaN	NaN	NaN	0
3	4	70	RL	60.0	9550	Pave	NaN	IR1	Lvl	AllPub	...	0	NaN	NaN	NaN	0
4	5	60	RL	84.0	14260	Pave	NaN	IR1	Lvl	AllPub	...	0	NaN	NaN	NaN	0

Table 1. A Quick View of the Dataset.

2.2 Data Preprocessing

2.2.1 Data Cleaning

The goal of our data cleaning process is to fit the origin data into our outlier detection algorithms and predictive models. To achieve our goal, we first write code to detect unknown values (such as “?” “Nan” “NA”, etc.) and determine the property of these unknown values. We found that NA in categorical data means do not have and NA in numerical data is meaningless. Then we separated numerical and categorical features of housing prices and saved them into different variables. After that, we performed different strategies of data cleaning based on the data type. For numerical data, we simply replace the confirmed meaningless value with 0. For categorical data, we assign labels for categorical data based on the properties of the data. For example, we labeled 3 for “Ex” which means excellent, 2 for “Gd” which means good. Since “NA” in categorical data is meaningful, we replace them with -2.

2.2.2 Feature Selection:

We applied two methods of feature selection: observation and simple regression. To do feature selection by observation, we have to study the properties of each feature in order to remove unrelated features. For example, ID, Fireplaces are not related to our target: sale price. Therefore, we can remove them from our feature list. This method is majorly used for categorical features. Simple regression feature selection is based on the statistical significance of least square regression. Using statistical significance, we can rate the correlation with stars, and the more stars the feature has, the more correlation it has with the target value(As shown in Table 2). Besides the simple regression test, we also draw a correlation heatmap based on the numerical data (shown in table 3). The darker the color is in the table, the more correlation it has with the corresponding value. Based on the result of the two methods and the heat map, we selected 14 numerical and 8 categorical features for our predictive models and sale price as our target value. After we test all the models, we remove two of the numerical features due to the result of the validation process.

	Estimate	Std. Error	t value	Pr(>|t|)	
(Intercept)	-0.012858	0.015175	-0.847	0.397090	
Id	-0.013626	0.015385	-0.886	0.376081	
MSSubClass	-0.098731	0.020267	-4.871	1.35e-06	***
LotFrontage	-0.051678	0.021394	-2.416	0.015946	*
LotArea	0.072992	0.015446	4.726	2.73e-06	***
OverallQual	0.319810	0.029058	11.006	< 2e-16	***
OverallCond	0.053740	0.019780	2.717	0.006740	**
YearBuilt	0.100216	0.038622	2.595	0.009646	**
YearRemodAdd	0.044033	0.025423	1.732	0.083672	.
MasVnrArea	0.087698	0.018057	4.857	1.45e-06	***
BsmtFinSF1	0.043235	0.037467	1.154	0.248880	
BsmtFinSF2	-0.001473	0.018500	-0.080	0.936552	
BsmtUnfSF	-0.002738	0.032483	-0.084	0.932848	
TotalBsmtSF	NA	NA	NA	NA	
X1stFlrSF	0.123166	0.038949	3.162	0.001627	**
X2ndFlrSF	0.135507	0.037832	3.582	0.000363	***
LowQualFinSF	0.023619	0.016419	1.438	0.150703	
GrLivArea	NA	NA	NA	NA	
BsmtFullBath	0.076388	0.023098	3.307	0.000987	***
BsmtHalfBath	0.016132	0.016524	0.976	0.329252	
FullBath	0.047878	0.026941	1.777	0.075943	.
HalfBath	-0.002500	0.022647	-0.110	0.912118	
BedroomAbvGr	-0.086567	0.022511	-3.846	0.000130	***
KitchenAbvGr	-0.065009	0.018843	-3.450	0.000591	***

Table 2. Statistical Significance of Features.

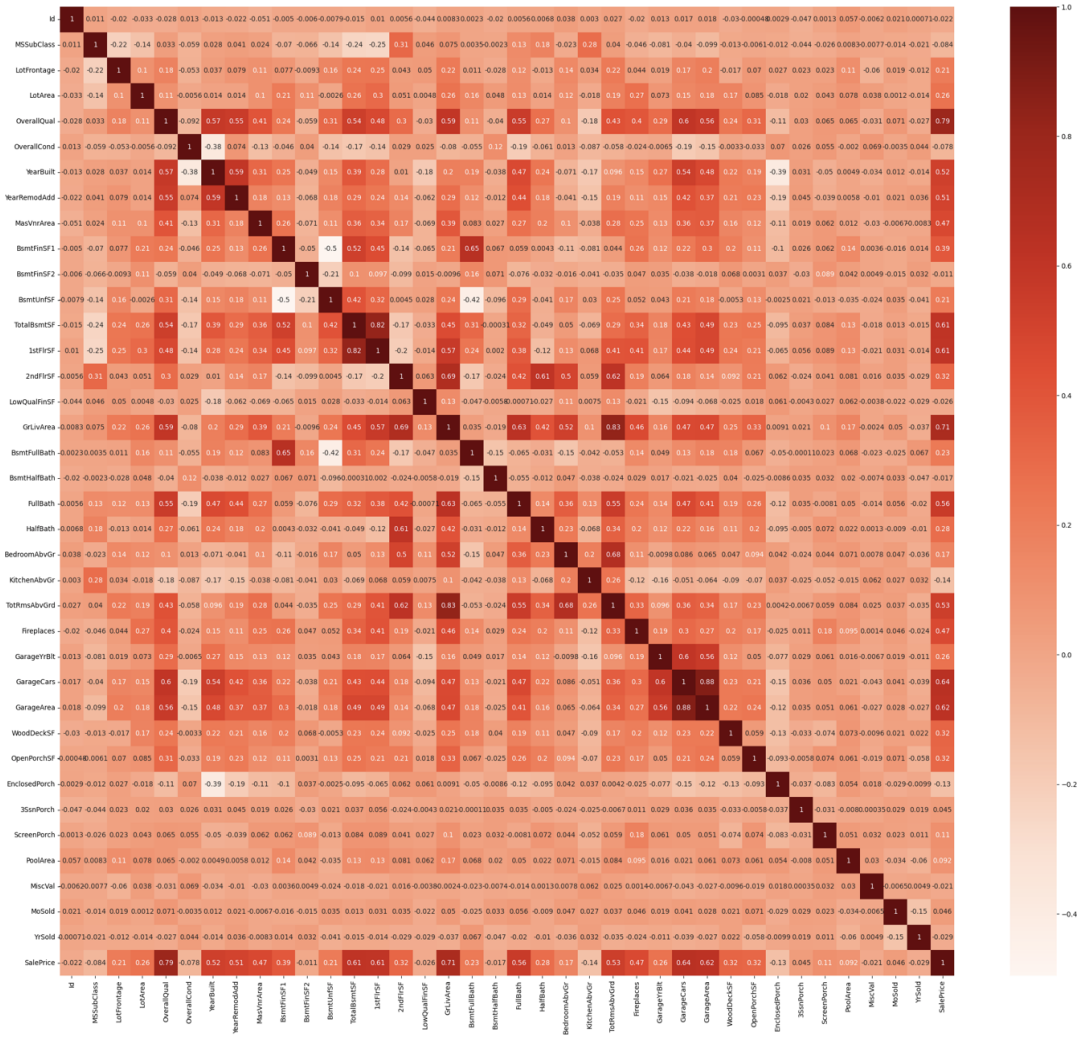


Table 3. Correlation Heatmap.

2.2.3 Standardization:

Most machine learning algorithms just see numbers — if there is a vast difference in the range, say few ranging in thousands and few ranging in the tens, and it makes the underlying assumption that higher ranging numbers have superiority of some sort. So these more significant numbers start playing a more decisive role while training the mode(Roy, 2020)l.

For the purposes of better-fitting models, we have implemented a scale function. The formula behind the scale function is (x-x_bar)/s where x is the real x value, x_bar is the sample mean and s is the sample standard deviation. After the scaling, the data is compressed in a small range which will be helpful for fitting models such as ANN.

2.2.4 Outlier detection:

Among all methods that the professor introduced to us in class, our team has tried multiple methods for our case which include: depthout(Figure 1), Mahalanobis Distance, KNN, RKOF, isolation forest, LOF and the join of them. The dataset we used for outlier detection is the preprocessed numerical dataset without categorical data which will be omitted in this step.

With the help of detection functions, we have successfully detected some outliers which may harm the performance of the model. By applying the Depthout method, we managed to find out 42 outliers; with the help of the Mahalanobis Distance method, we have successfully found 48 outliers; by applying the K nearest neighbor's distance method, we have successfully found 40 outliers; at last, with the help of Join assessment of outlier detection methods, we have successfully found out 89 outliers. We have carefully looked into the outlier groups since we wish to find commonalities among those outliers.However, after detecting outliers, we found it hard to deal with these outliers. We want to eliminate them but also want to keep the integrity of the dataset (distribution shown in Figure 2). Finally, we applied the IQR method to deal with outliers to deal with our concern.

The IQR algorithm is simple but powerful. By applying the algorithms, we need to first find out the upper and lower quantile of each numerical feature. Then, we can set a boundary based on the quantile of each numerical feature. If the data point is greater than the upper quantile, we replace it with the number of upper quartiles and if the data point is lower than the lower quantile, we replace it with the number of lower quantiles. By applying this algorithms, we keep the integrity of the model and eliminate outliers (new distribution shown in Figure 3)

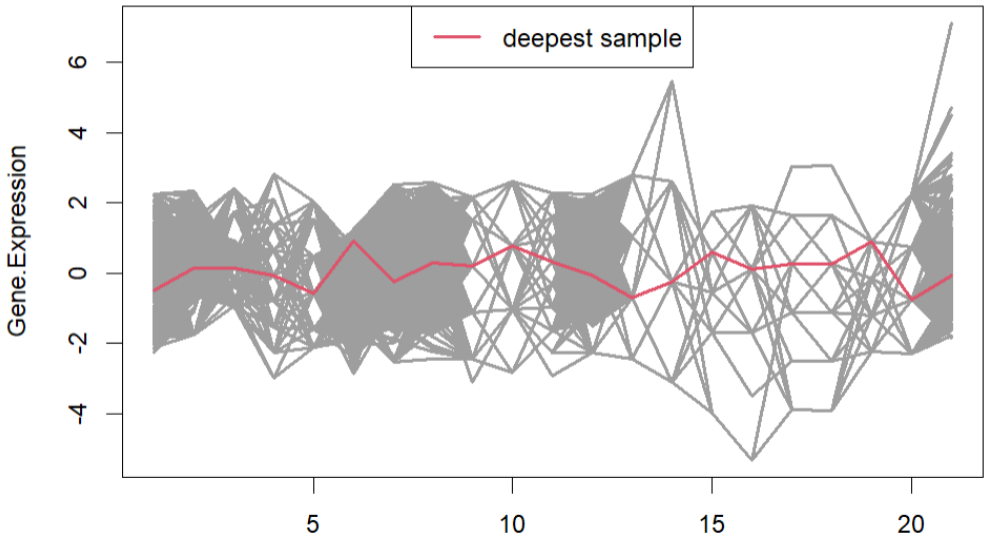


Figure 1 Depth-based Approach

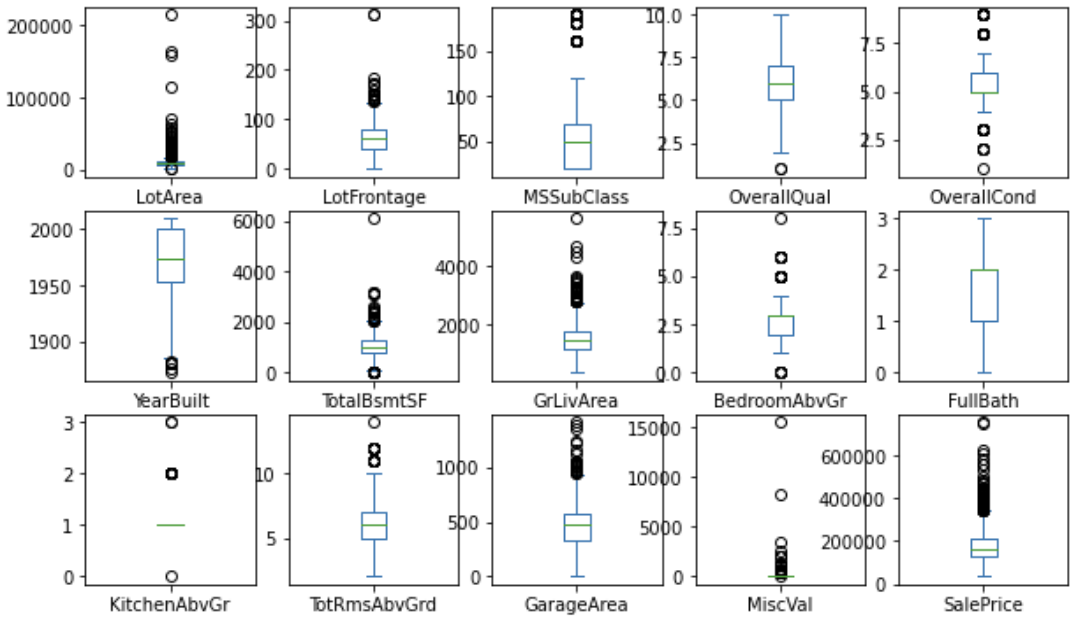


Figure 2. Distribution of Numerical Features Before IQR

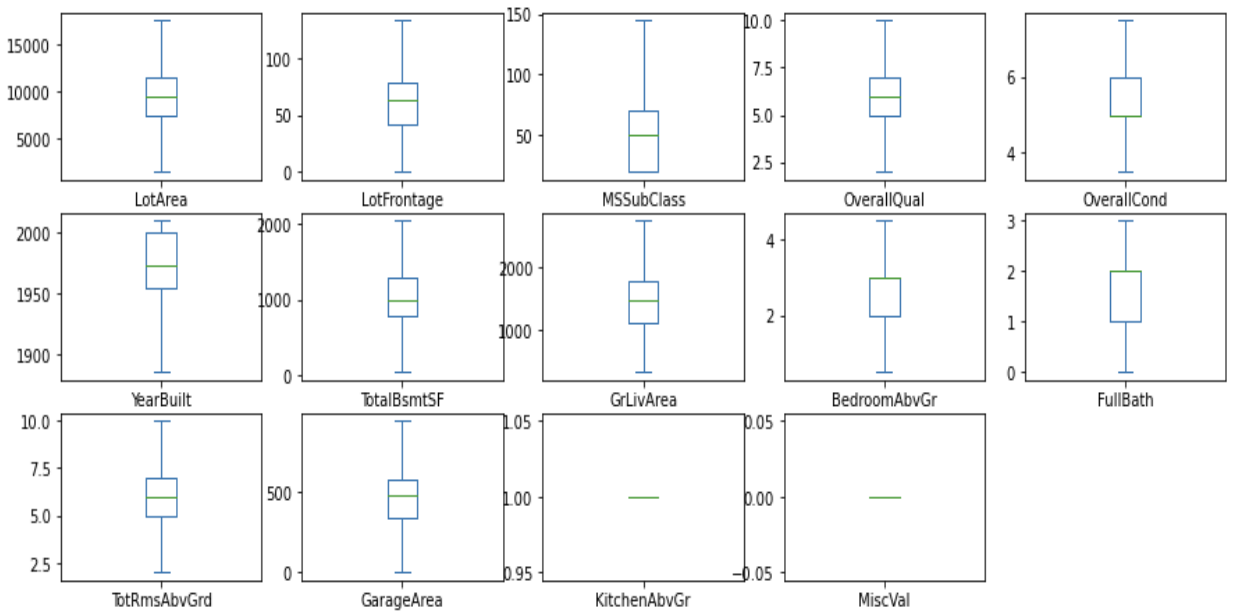


Figure 3. Distribution of Numerical Features After IQR

3. Model And Results

3.1 Model

3.1.1 Decision Tree

We decided to apply the Decision Tree method since it's a Machine Learning Algorithm that it's able to deliver information visually and clearly to customers. As a project that is closely related to real-life applications, we need to find a way to deliver information efficiently and clearly. Other complex ML Algorithms might be efficient to obtain a reasonable result, but there are natural disadvantages in delivering information for other powerful tools.

However, this method can not provide the most acceptable result in most cases, but it can help people who wish to purchase or sell a household quickly estimate the corresponding SalePrice.

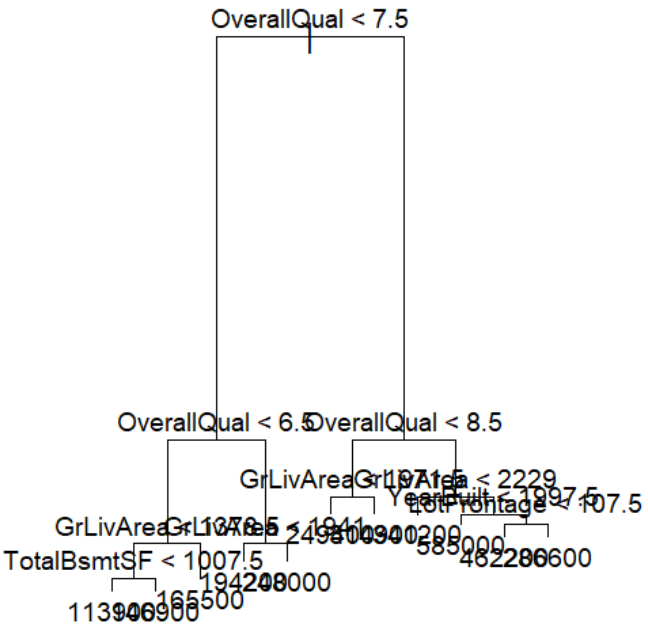


Figure 4. Tree Plot

3.1.2 Least Square Regression

The R squared value of the LM method is **0.792278**, unlike our intuitive expectation, LM method performs well on the certain data set. We may conclude that features we collected as independent variables are not complexly correlated with our target value.

Otherwise, the LM model will generate an awfully fitted model, for which is not reasonable to be chosen.

3.1.3 Least Absolute Deviation Regression

The least absolute deviations method (the LAD method) is one of the principal alternatives to the least-squares methods when one seeks to estimate regression parameters. The goal of the LAD regression is to provide a robust estimator.[5]

Given a family Φ of functions ϕ, and given the data, the function $\phi \in \Phi$ which minimizes L2(ϕ) over Φ is called the least squares regression over Φ, and the function which minimizes L1(ϕ) over Φ is called the least absolute deviations regression over Φ.(Carmona,2014)

$$\mathcal{L}_1(\varphi) = \sum_{j=1}^{n} |y_j - \varphi(x_j)|.$$

The R squared value of Least Absolute Deviation Method is **0.78291**, we may observe that the performance of LM is better than LAD method.

There are differences between the two regression methods. The Least Square fitted values changed dramatically while the LAD fitted values remained the same. This robustness of the least absolute deviations regression can be extremely useful. Indeed, there are times when one wants the estimations and predictions to change with changing data, however, with noisy data, it is generally not a good idea to use estimation and prediction procedures which are too sensitive to small changes, mostly because the latter are very likely due to the noise, and for this reason, they should not have an overly dramatic impact on the outcome(Carmona,2014)

3.1.4 Random Forest

The R squared value of the LM method is **0.8198267** with 500 trees and 12 features are selected for each tree**.** As we reduce the number of selected features for each tree, we can observe a reduction on R squared value to **0.808649** with 2 features selected for each tree.

These results are in line with our intuitive expectations and are better than the first two methods. Random Forest algorithm is a machine learning algorithm that uses ensemble learning, which enables organizations to solve regression and classification problems. It may also reduce the negative effect caused by overfitting in reality.

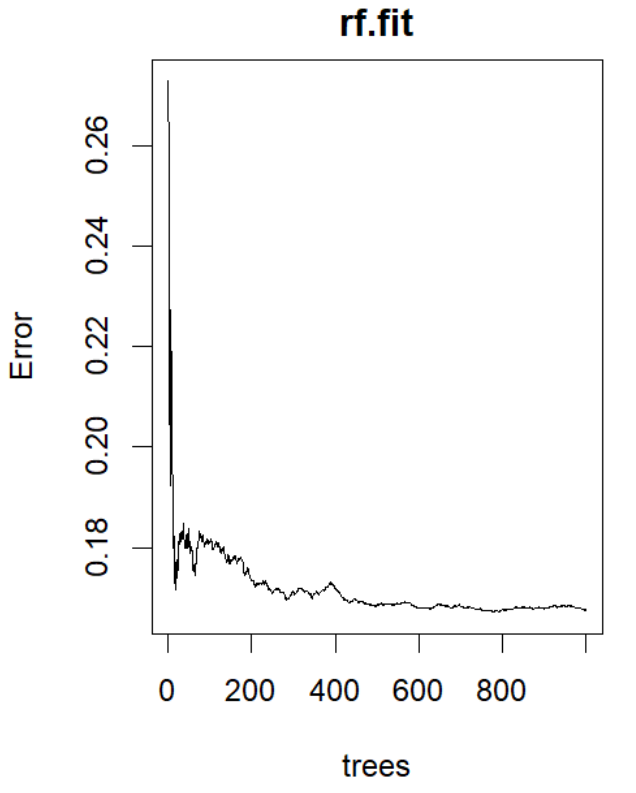


Figure 5.

The plot generated by Random Forest suggests that large numbers of trees significantly increase the accuracy of the algorithm. However, in practical applications, a large number of trees might over-spend resources and lead to a Long-cycle operation for which will increase the cost on certain projects. We have to choose applied trees wisely to avoid unnecessary cost.

3.1.5 ANN

Artificial Neural Network, abbreviation ANN, is a model simulating an animal's brain as shown in graph 4. Input layer is where we put our feature's data. Then it will be multiplied by the learned weights and calculated by activation functions in the hidden layer to generate outputs in the output layer. The more hidden layer ANN has, the more complex the model is. For the purpose of reducing errors, we need to decide the tradeoff between model complexity and overfitting. It is usually defined as variance-bias decomposition.

After validation for multiple runs, we chose Tensorflow on Python as the operating package, Relu as our activation function, RMSprop as our optimizer, 0.04 as our learning rate and error is generated by mean absolute error. Also the iteration we set for the mode is 40. Because as shown in graph 5, the error is not reducing after 20 iterations. For model accuracy comparison purposes, we use R square value to score the model performance. The R squared Value for testing is 0.8560359910172952.

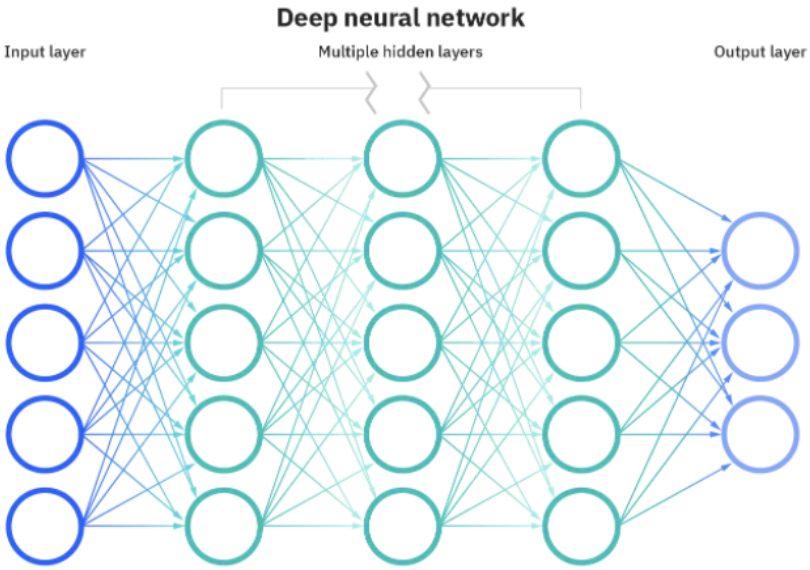


Figure 6. ANN

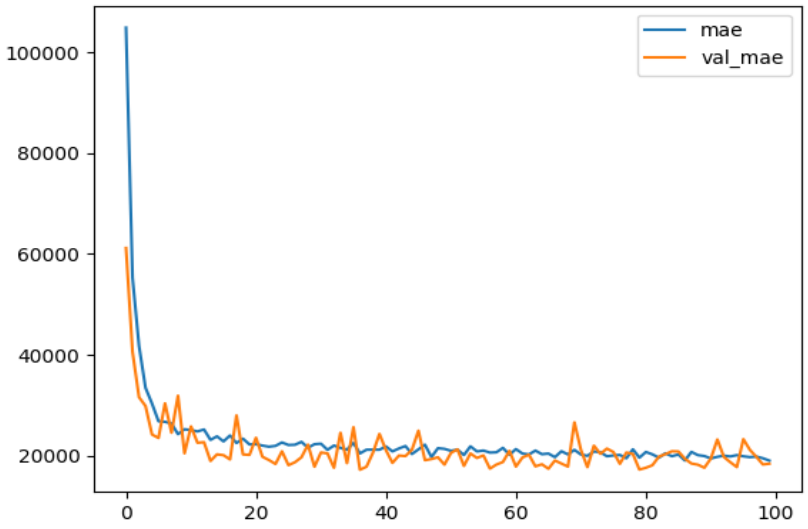


Figure 7. MAE of ANN by Iterations

3.2 Results

Based on the scores of R square value, the ANN has a slight advantage among all the algorithms. Therefore, we have decided to use ANN as our final predictive model. Following are the graphs of each algorithm and accuracy results table.

Algorithms	Scores for Testing
Decision Tree	NA
Least Square Regression	0.792
Least Absolute Deviation Regression	0.783
Random Forest	0.82
ANN	0.856

Table 4. Result among various Methods on Test Data set

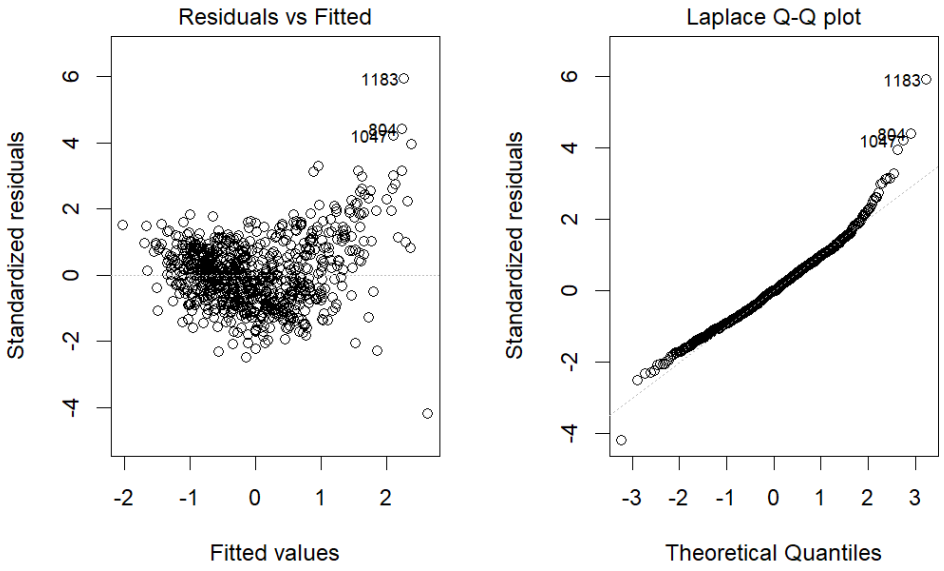


Figure 8. Residual plots of LAD Regression

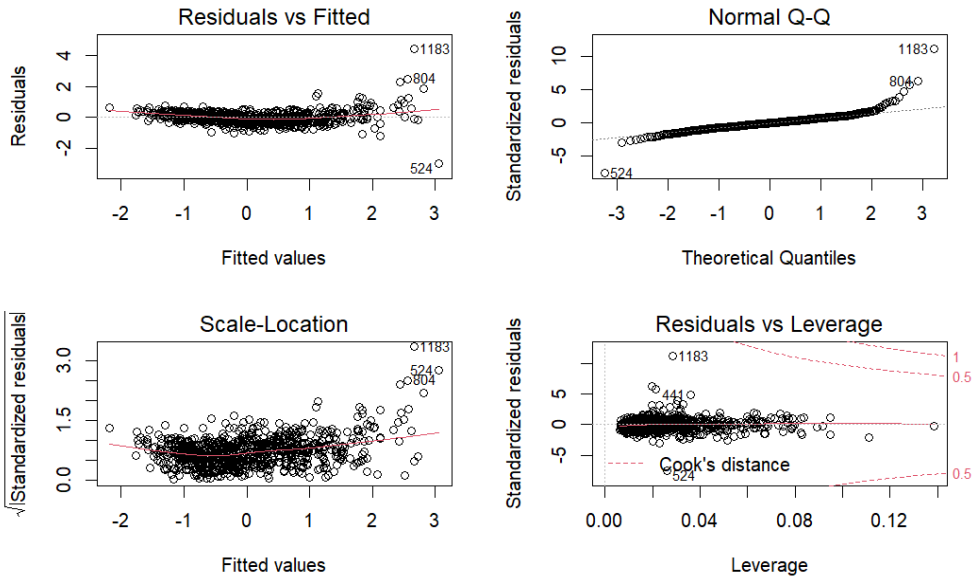
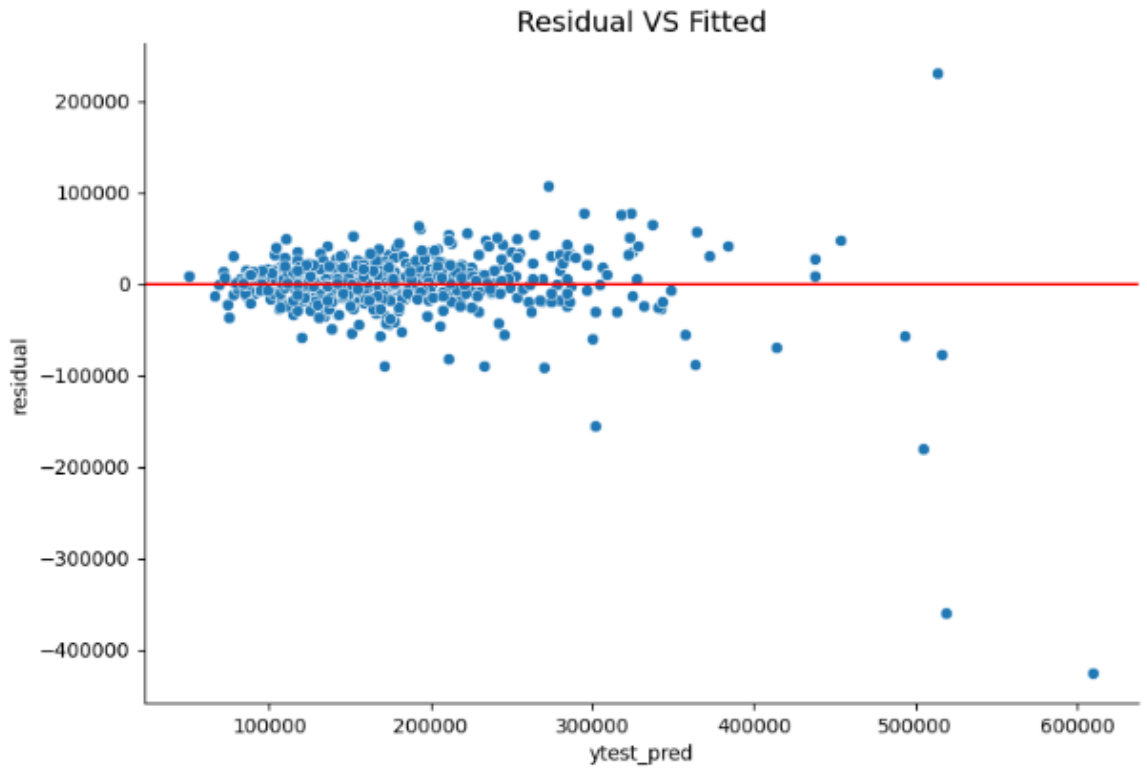
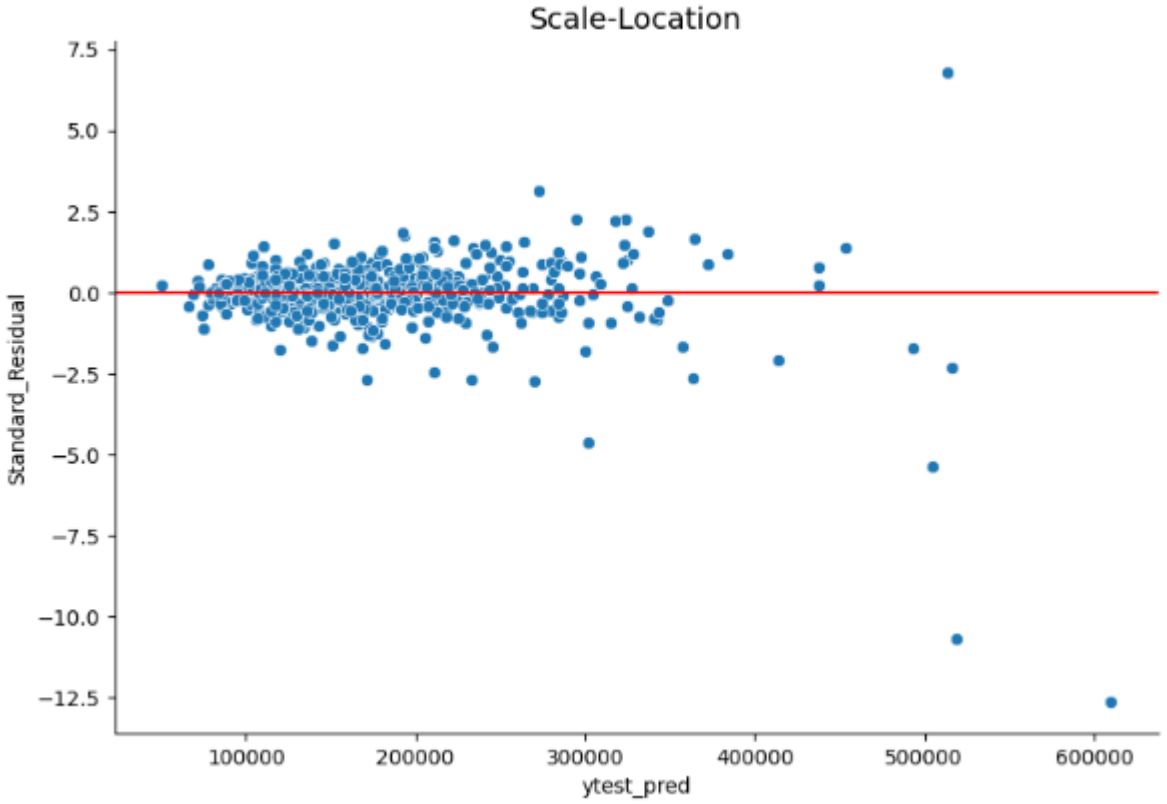
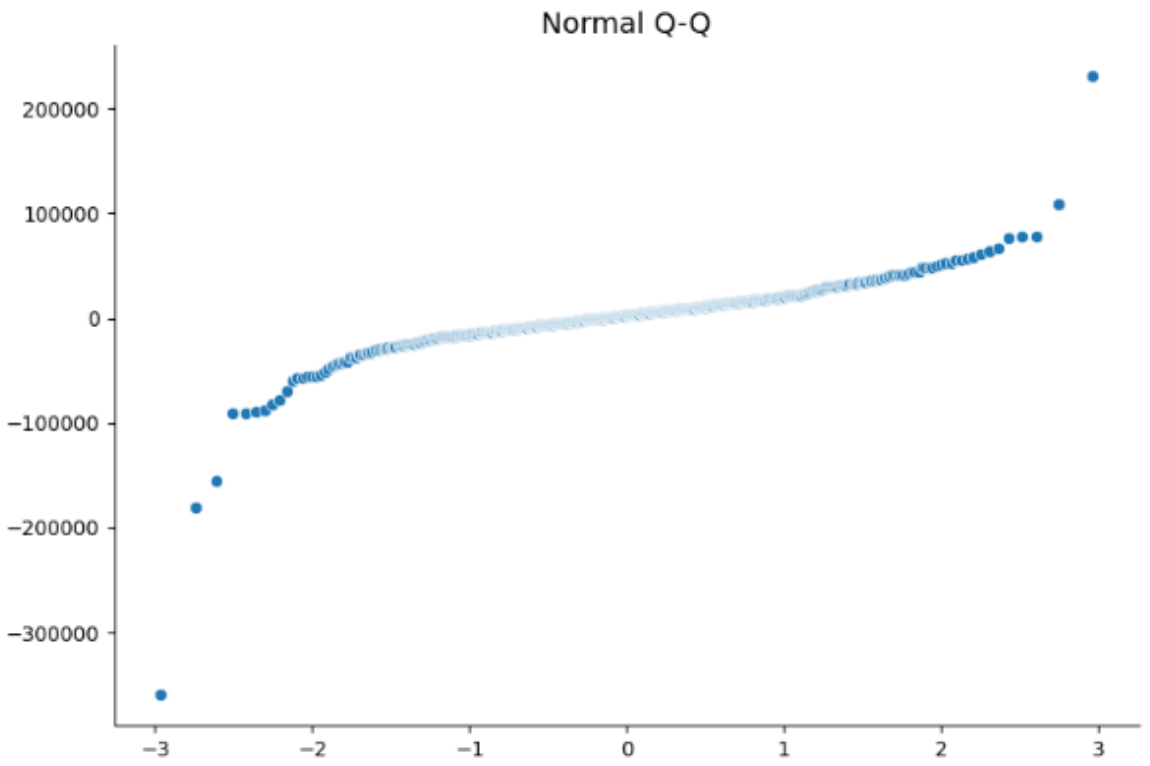


Figure 9. Residual plots of LM Regression

Those plots suggest a rational fitness of LAD Regression and LM Regression on the certain data set with only few potential outliers.


Residual VS Fitted
200000
100000
0
−100000
−200000
−300000
−400000
residual
100000
200000
300000
400000
500000
600000
ytest_pred


Scale-Location
7.5
5.0
2.5
0.0
−2.5
−5.0
−7.5
−10.0
−12.5
Standard_Residual
100000
200000
300000
400000
500000
600000
ytest_pred



4. Conclusion

By completing the project, both of us have learned a lot including the analytics framework, influence of outliers, data preprocessing skills and so on. And most importantly, we learned how to implement machine learning algorithms into real life cases. The time spent is long but the gain is also huge. We are both thankful for the opportunity given.

We used ANN as our final model but a problem also occurs. As mentioned, we have a small dataset which will affect the result of the model. If the dataset is smaller, ANN may not have a better result than other algorithms because the model complexity of ANN will lead to overfitting on a smaller dataset. That is one thing to be concerned about when applying the result of our project.

In addition to the model performance, we also want to talk about the feature selection. Due to the time of the project and the limitations of the dataset, we cannot take some important features into consideration such as location, demands for buying and so on. However, our model reflects the true value of a house regardless of its additional value such as demand for buying, developer etcs. These additional values are usually added to the house for better price, more profits. As a result, our project may be more helpful for buyers who want to pay for what they really get and not so friendly for real estate speculation.

5. References

1. Mac Statista Research Department, Freddie. “U.S. Home Sales 2021.” *Statista*, Freddie Mac, 4 May 2021, https://www.statista.com/statistics/275156/total-home-sales-in-the-united-states-from-2009/.
2. Holbrook, Ryan. “Fe Course Data.” Kaggle, 8 Jan. 2021, https://www.kaggle.com/ryanholbrook/fe-course-data.·
3. Cawi, Eric. “Machine Learning Morphisms: Eric.” Cawi, https://www.cawi.science/machine-learning-morphisms.
4. https://towardsdatascience.com/all-about-feature-scaling-bcc0ad75cb35
5. (2008) Least Absolute Deviation Regression. In: The Concise Encyclopedia of Statistics. Springer, New York, NY. https://doi.org/10.1007/978-0-387-32833-1_225
6. IBM Cloud Education. “What Are Neural Networks?” *IBM*, https://www.ibm.com/cloud/learn/neural-networks.