# Best Supervised Learning Algorithm To Predict House Prices

Gustavo M. Ramos
*Applied Mathematics*
*San Diego State University*
San Diego, USA
gramos3383@sdsu.edu

***Abstract*—House buying can be a very timely and costly process. Determining the value of a house on the other hand can be an even longer process. How do we determine the value a house will be and what exactly is the best way to identify house sale price value? As an example, we will use house price training data from a Kaggle competition to determine best way to find sale prices for homes. In order to do so, we will first be loading the data and looking for any missing values so that we can fill in the data for those null as best we can. After, we will do some exploratory data analysis and try normalizing any skewed features in our data. Then, we will implement supervised learning algorithms such as: Linear Regression, Random Forest, XGB, LightGBM to determine which algorithm will be most effective.**

***Index Terms*—**

## INTRODUCTION

Many people want to buy houses but now a days it has become very expensive and pricey to purchase. Let alone, some houses will be very over priced for what they have to offer to many sometimes. It becomes very difficult to determine whats a reasonable price based on many other house price ranges. many people have never bought houses before and might not know what factors constitute as important factors to a houses sale price. so how exactly is that one might be able to predict what a houses sale price be if they are only given certain information about a house.

In order to do so, i have taken given information from a kaggle repository for house price data predictions competition. From there, we load our test and training data that we plan to use for the purpose of predicting sale prices. It is crucial that we use anomaly detecting and exploratory data analysis in order to understand what features might play important factors into our predictions. following along on those notes we will need to use feature engineering techniques in order to make our data much more normal and enhancing the modeling accuracy of our predictions.

After having handled the data, we begin to implement our approaches. We begin are supervised learning algorithm with Linear Regression while applying root mean squared error as a our numeric value to compare among the other supervised learning algorithms. We follow along after linear regression with Random Forest Regressor, XGBoost; And lastly LightGBM. they will each be applied under the assumptions that the data is to be :

- Will not have empty values and features will be filled based on whether they are numeric values that might need Mean, median applications or if they are categorical and might need mode applications to better fill data to enhance model accuracy.
- Data is to be normalized and transformed before applying supervised learning algorithms.
- Will use cross validation with N-folds to better enhance model accuracy with out over fitting.

## APPROACH

In order to test the data we will need to do the following approaches. we will begin by handling data, then detecting for any anomalies in data, then we will go ahead the needs for supervised learning to train test data.the following approach is:

- 1 Data Importing and Pre-processing
- 2 Data Analysis and Visualization
- 3 Data Analytics

### *A. Data Importing and Pre-processing*

Train and Test data are loaded onto the notebook and then we begin to use pandas package to check our dimensions for our data frame. after we examine the data types of our features in order to better understand which features are most likely going to be numerical and categorical. after doing so we then determine the percentage of empty values per feature and sort them based on most to least missing. Following up, we then determine the best way to fill in the empty values depending on their data type (numerical or categorical).

### *B. Data Analysis and Visualization*

Folowing up on Data pre-processing, we continue by making a numerical and categorical list of all our features in the table. after doing so we show a statistical summary of the training data set after making sure their are no empty values. We will begin to do Anomaly detection by creating plots of graphs such as our SalePrice vs GrLivArea to get rid of any outliers that might be affecting our predictions. adding on to plots, we also make distributions of our Sale Price feature

along with other numerical and categorical data that might be skewed and affecting our training set. we will handle skewed data by normalizing the data such as our Sale Price feature. Following along with more visuals, we also create a heat map to find any features that might be heavily correlated that might affect our accuracy results. after having transformed all skewed features in our training data set we can begin to do label and one hot encoding before training our data.

### C. Data Analytics

To do Supervised Learning algorithms we will begin by downloading our needed packages from SKlearn such as Linear Regression, Decision Tree, XGBoost, and LightGBM. Other packages imported are Lasso, Kfold, and cross val score. After having transformed our training data set, we then train our data. In order to do cross validation, we add a method defined as 'cross validation rmse' where the arguments passed in our our model, and the number of folds we intend to use for the cross validation. for the purpose of this method our cross validation folds will be a constant number of 5 folds. our method will return our root mean squared error (L2 Loss).

after creating our method, we label variable that will be assigned to our supervised learning models. We assign linear regression model its own variable and the same applies for Decision Tree, XGBoost, LightGBM. for the purpose of our tree base model we will use a max depth of 10 for our tree and our learning rate will equal 0.1. We then implement our models into our cross validation method where they will go under 5-fold cross validation and return the root mean squared error. we will then Output the L2 Loss and their standard Deviations.

## Evaluation

In our key findings to determine which supervised learning algorithm will be most accurate; We determine the following olist from most to least accurate based upon our Squared mean error (L2 Loss). They are:

- Linear Regression: L2 Loss = 0.0145, STD = 0.000887
- XGBoost: L2 Loss = 0.0146, STD = 0.000724
- LightGBM: L2 Loss = 0.0150, STD = 0.000525
- Decision Tree: L2 Loss = 0.0232, STD = 0.00136

based on these errors, our best supervised learning algorithm to implement to predict house prices would be our Linear Regression Model. XGBoost was approximately the same mean squared error with a smaller standard deviation. However, since XGBoost is tree based; the implementation process would be much greater in time it would be costly and very inefficient in terms of engineering time to implement this learning algorithm when Linear Regression takes less time to implement. Our LightGBM mean Squared error was slightly larger in error with a difference of less than 0.2 percent. However similar to Our XGBoost, it would also not be very cost and time efficient to implement if we wanted too. The most inefficient was our decision tree model which resulted in a approximate error difference 1 percent.

Due to these squared errors in our tree based models can be contributed to the fact they most likely have some over fitting happening and can affect our accuracy.

Following up on our accuracy results we then determine the feature that played most important weight roles in predicting our sale price. We were able to do so by by using sklearn packages and implementing a tree based model to do. Only can a tree based model help in determining our role importance. Out top ten important features are listed below from greatest to least important.

- 11stFlrSF
- 2BsmtUnfSF
- 3LotArea
- 4GarageArea
- 5BsmtFinSF1
- 6GrLivArea
- 7TotalSF
- 8LotFrontage
- 9OpenPorchSF
- 10MoSold

The tree based model used to find feature importances was Our XGBoost model that we fitted using sklearn's fit method.

## Related Work

further explaining our approaches, we discuss our methods. Starting with Linear regression. For the purpose of this project, it was the most accurate for our predictions. It also turned out that Linear Regression is the simplest of our Supervised Learning algorithms since its is easiest to train, implement and interpret our data on.However, since we were prone to outliers; we had to do some anomaly detection and get rid of outliers since Linear regression is prone to outlying data.

For Our tree based model such as Decision trees, XGBoost, and LightGBM, we find that they are relatively much easier to handle since they need less data pre processing and preparation time. it does not require normalization and is easy to interpret. they are very great at handling large scale data. however they can be very prone to over fitting. and can be affect to noisy data depending on max depth of these trees.

## Conclusions

In order to determine which supervised learning algorithm we compared Linear Regression, Decision Trees, XGBoost Regressor, LightGBM Regressor.But first, we needed to do some data pre processing and cleaning. This was done by evaluating where we had missing data values and determined how to fill in that data after categorizing whether they were numerical or categorical. We then did some exploratory data analysis to try to find any skewed features and also anomaly detection to better train our data on.It is then we were able to implement our models onto our transformed data and evaluate which algorithm was most effective. As a result, we concluded that Linear Regression was by far most effective since it had the lowest mean squared error and it was the most effective in terms of time and complexity. In contrast, our other models were too expensive in terms of time and complexity. These

models were also very prone to over fitting. In the end, We were able to determine the importance of Features that influence the predictions.

## REFERENCES


[1] "House Prices - Advanced Regression Techniques." Kaggle, https://www.kaggle.com/competitions/house-prices-advanced-regression-techniques/data.

[2] "House Prices Beginners." Kaggle.com, www.kaggle.com/code/skashperova/house-prices-beginners. Accessed 11 Dec. 2022.

[3] "House Prices Prediction." Kaggle.com, www.kaggle.com/code/vshantam/house-prices-prediction#Feature-Engineering. Accessed 11 Dec. 2022.

[4] "House Price Prediction." Kaggle.com, www.kaggle.com/code/emrearslan123/house-price-prediction. Accessed 11 Dec. 2022.