Predicting House Prices

Abstract - We review the regression and classification models to predict the price of houses based on a set of factors. Inspired by the Linear Regression model, we aim to train and test using a housing dataset and compare the results with the KNN Classification model. Other than presenting the predicted values, we assess the average accuracy of the models by comparing the actual prices.

Keywords - Multiple Linear Regression, KNN, Lasso Regression, House Price Prediction

I. INTRODUCTION

The housing market is one of the most unpredictable markets because the demand for a particular house could depend on various factors such as the weather conditions of the area, size of the house, proximity to schools, hospitals, grocery stores, etc., age and health of the house, etc. It is nearly impossible to filter all the data to arrive at the house that is suitable to our requirements manually. However, machine learning can help in easing this process where it can crunch through enormous amounts of data and constraints to reveal correlations and trends between the various attributes and factors of the data to eventually output the best house price for the customer by inputting the various constraints of the customer.

The dataset for this project contains the house price and the date that it was listed along with other potentially independent attributes like number of bedrooms, bathrooms, floors, lot area, location information, year built, etc. Our analysis in the following report shows the regression performed and the results showing the trends and correlations between the various data points.

II. OBJECTIVE

In this project, we will be predicting the prices of houses by using three different machine-learning models. i.e. Linear Regression, KNN with different K values, and Lasso method.

A. Libraries Used

- sklearn.model_selection-train_test_split – used to divide dataset randomly for training and testing
- Sklearn LinearRegression – algorithm used for unsupervised machine learning.
- sklearn.metrics accuracy_score -this library is used to compare the predicted values to the actual values in test dataset.

B. Factors

To develop these models, we have selected six variables-sqft_living, bathrooms, bedrooms, grade, floors and view. We will train our system by inputting historical data and the results produced by this train and test method will be compared with each other based on the efficiency of the model. The comparison of the machine learning models gives us an insight into the accuracy and functioning of the model.

The end goal of this project is to predict the Housing prices as accurately as possible using the best machine learning model.

III. MACHINE LEARNING ALGORITHM

A. Multiple Linear Regression

Multiple Linear Regression is the extended version of simple Linear Regression. It is also a regression model that is used to build a relationship between dependent and independent variables. In simple Linear Regression, we need only one predictor to calculate the predicted variable. However, Multiple Linear Regression needs more than one variable to predict the response variable. The main advantage of Multiple Linear Regression is that it acts on any size of the dataset [1].

The Multiple Linear Regression equation is given below:

y = b0 + b1 *x1 + b2 * x2 +.... + bn * xn

Where,

y= outcome or predictive variable.

b0= Constant or intercept.

b1, b2, b3……. bn= Coefficients

x1, x2, x3……...xn= Predictors

Before building a Multi Linear Regression Model, we had to complete some steps. The steps are as follows.

Firstly, we identified dependent and independent variables in the dataset because these variables are the main attributes to get the correct result [2].

Secondly, we recognized the categorical and numerical or continuous variables to convert categorical variables to dummy variables. In the given dataset, we don't have categorical variables. Thus, we did not make any dummy variables.

Thirdly, we created a heatmap to know the correlation between different variables. From the correlation, we selected 6 independent and one dependent variable.

Fourthly, we installed some libraries to get different results from the model.

Finally, we divided the dataset into training and test dataset to know the performance and accuracy of the model.

After completing the aforesaid steps, we executed the Multi Linear Regression and got the following results:

Predictor	**coefficient**
sqft_living	203.161613
bathrooms	-14607.39388
bedrooms	-31084.96629
grade	92334.11787
floors	-21272.49654

Table 1. Coefficient value for the predictors

- Result from testing the model

Predicted	**Actual**	**% Error**
346470	297000	16.65682
1399923	1578000	-11.285

Table 2. Comparing predicted and actual price values using Multiple Linear Regression

- Accuracy of the model

R2 coefficient of determination is evaluated to determine the accuracy. The value of r2 in the Model Performance for test data is 0.576 which represents that 57.6% data fit the regression model. We are getting very high r2 and adjusted_r2 values for both training and testing dataset, thus this is a very good model. Since there is not much difference in the performance for Training and test datasets, this model is not overfit.

B. KNN Classification

The K-NN algorithm is one of the methods used for classification analysis, but the last few decades the KNN method has also been used for prediction K-Nearest Neighbor algorithm is a method to classify objects based on learning data closest to the object. Nearest Neighbor is an approach to finding cases by calculating the proximity between the new case and the old case that is based on matching the weights of a number of existing features. The working principle of K-Nearest Neighbor is to find the closest distance between the data to be evaluated with the nearest neighbor in the training data. Training data is projected into many-dimensional spaces, of which each dimension explains the features of the data. This space is divided into sections based on training data classification. A point in this space is denoted by class c, if class c is the most commonly encountered classification of the nearest neighbor of the point [3].

We have tested our model based on two values of K i.e., K =10 and K=5. The accuracy of the model with K=10 is 72% whereas the accuracy of the model with k=5 is 75%. The mean squared error which is the average of difference between the given values and predicted values for k = 10 is 324485.29 and for K = 5 is 299588.75. Using the K = 5 model we predicted the prices of two houses and compared them with the actual price. The error rate for the predictions were -6% to -8% approximately.

Predicted	**Actual**	**% Error**
277500	297000	-6.565657
1450000	1578000	-8.111534

Table 3. Comparing predicted and actual prices using KNN(K=5) Model

C. Lasso Regression

Lasso is a linear regression model that can be used when we are working on a subset of variables in a dataset. The accuracy score for predicted value is comparatively high using this regression model. It uses absolute values of weights and sum it up to reduce the absolute values. It is helpful in compressing the data as it makes predictions based on few non-zero coefficient values [4]. To minimize the dependent variables, the objective function used is

$$\min_{w} \frac{1}{2n_{\text{samples}}} ||Xw - y||_2^2 + \alpha ||w||_1$$

Here, alpha is the coefficient introduced to penalize the weights.

In the given Housing Dataset, square feet area ranges to 13000, while number of floors ranges from 1 to 3.5. In this case, weight to predict price will be lower for square feet than for the weight for floor. Hence, we need to scale the data by penalizing the weight using alpha so that big weights can be derived down in comparison to small weights.

To implement the model, dataset is divided into features and target. This data is further divided for training and testing purposes in an 80-20 ratio. The penalizing value, alpha is used as 0.1 to create the regression model. To verify the accuracy of the model, predicted price values are compared against the actual value and the model accuracy is 57.67%

Predicted	**Actual**	**% Error**
346470	297000	16.65682
1399923	1578000	-11.285

Table. 4 Comparing predicted and actual prices using Lasso Regression Model

IV. CONCLUSION

By running the correlation heatmap of all the variables with price, we were able to identify the 6 house specifications - sqft_living, bathrooms, bedrooms, grade, floors and view - that are most related to the house price.

We used multiple linear regression to predict the prices of two randomly selected houses having the aforementioned house specifications, the results of which, showed prediction errors of 16.65% and -11.28% respectively. The overall multiple linear regression model was observed to have an accuracy score of 57.67%.

We built the KNN classification models using K=5 and K=10 and determined that the KNN model with K=5 has a higher accuracy score of 75% as compared to that of K=10 which was 72%. Hence, we predicted the prices of the previously selected two houses, using the KNN model with K=5, the results of which are below which had a prediction error of -6.57% and -8.11% respectively.

We also used the Lasso regression model, but we determined that this model was as accurate as the Multiple linear regression model since the accuracy score was almost the same at 57.67%. However, lasso regression has one advantage over linear regression which is that it overcomes the overfitting problem since this model is less sensitive to the training data than linear regression.

We compared the prediction performance of multiple linear regression, lasso regression and KNN classifier models and conclude that the KNN classifier with K=5 had the highest accuracy score of 75% and would be the most suitable to predict the house prices using the given training dataset.

REFERENCES

[1] Uyanık, G. K., & Güler, N. (2013). A study on multiple linear regression analysis. Procedia-Social and Behavioral Sciences, 106, 234-240.

[2] Amral, N., Ozveren, C. S., & King, D. (2007, September). Short term load forecasting using multiple linear regression. In 2007 42nd International universities power engineering conference (pp. 1192-1198). IEEE.

[3] Zhang, S., Li, X., Zong, M., Zhu, X., & Cheng, D. (2017). Learning k for knn classification. ACM Transactions on Intelligent Systems and Technology (TIST), 8(3), 1-19.

[4] Ranstam, J., & Cook, J. A. (2018). LASSO regression. Journal of British Surgery, 105(10), 1348-1348.