Boston Housing : Linear Regression and MLP

Pranay Kumar Verma

***Abstract*—In recent years, the field of Machine Learning has touched almost every aspect of human life. Machine Learning has played an important role in understanding hidden patterns in data which is used to derive valuable and helpful information. This is formally known as knowledge discovery from data (KDD). On such aspect is to understand the factors which housing prices depend on [4]. While the factors like size and location of a house are the simplest and most common to determine the price of a particular unit, there are various other factors that affect real estate prices which can't be seen or understood directly. This study is done to understand the features which affect housing prices in a city and is an attempt to make predictions after learning those features.**

I. Motivation

There has been a rise in real estate sector in recent years and so it has become important to understand the driving factors for the housing prices. This is required to help businesses to plan accordingly and mitigate factors that could result in an unusual price rise. As for consumers, this will help to determine approximate pricing that will keep a check on the prices so that they pay only for what they are getting. Also, with the rise in city dwelling population, the ability to predict prices helps governments and authorities to regulate market for the construction companies.

The prediction can be done using past data i.e. by carefully studying the impact of various factors over target variable and applying the learning over future prices. This process of learning the past feature values and use that learning to predict the outcome for unseen feature values is known as regression. The aim of this study is to predict the median prices of houses [3] using the Boston Housing dataset using techniques like Linear Regression and MLP Regression (Multi Layer Perceptron Regression).

II. Prior Work

Regression is defined as a process to find out the relation between two or more variable. Once the relationship is identified, values of one variable can be predicted based on other variables. The most common and simplest form of regression is Linear Regression where there is only one independent and one dependent variable. The two variables are highly correlated and the relationship of the variables can be represented by a straight line.

Before applying any Machine Learning method, data must be analyzed manually to understand which features are best for the task at hand. This activity of identifying features is called Feature Selection. Data must also be scanned for any missed values and if found any, those must be filled in with appropriate values (0, average of all observations, etc.). Entirely new features can also be created with the help of existing features. This is known as Feature Extraction. These activities of feature selection and feature extraction together account for Dimensionality Reduction of the dataset. Dimensionality is helpful because it reduces the size of the data while preserving the important features to start the learning.

In this study, we are going to be using Boston Housing Dataset that doesn't require any preprocessing in terms of feature normalization, feature extraction, feature selection, missing values, etc. The data comes from sklearn.dataset library and is in ready to use state. However, a check can be performed to determine if the features has any correlation with the target variable. This can be done by either plotting individual feature against target variable or by creating a correlation matrix.

III. Model/Algorithm/Method

A. Dataset

The dataset used in this project is the Boston Housing dataset that can be loaded using the sklearn.dataset library. The dataset consists of 13 features and 1 target variable. There are a total of 506 observations in the dataset which can be used to make predictions. A brief description of features is given below [1]:

1) CRIM - per capita crime rate by town

2) ZN - proportion of residential land zoned for lots over 25,000 sq.ft.
3) INDUS - proportion of non-retail business acres per town
4) CHAS - Charles River dummy variable (= 1 if tract bounds river; 0 otherwise)
5) NOX - nitric oxides concentration (parts per 10 million)
6) RM - average number of rooms per dwelling
7) AGE - proportion of owner-occupied units built prior to 1940
8) DIS - weighted distances to five Boston employment centres
9) RAD - index of accessibility to radial highways
10) TAX - full-value property-tax rate per $10,000
11) PTRATIO - pupil-teacher ratio by town.
12) B - $1000(Bk - 0.63)^2$ where Bk is the proportion of blacks by town
13) LSTAT - % lower status of the population
14) MEDV - Median value of owner-occupied homes in $1000's

The last feature MEDV is the target variable which we are trying to predict in this study.

B. Preprocessing

There is not much preprocessing required on the dataset as its a standard dataset that is provided in python sklearn.dataset library. However, we will normalize the features before we proceed with our analysis. We could use only one feature, some features or all of the features to make a prediction. We will be using all features in this study.

We will check the correlation of all features with the target variable MEDV [2] and then proceed with the prediction algorithm as mentioned in the subsequent sections.

C. Initial Analysis

To solve the problem at hand, we are going to make use of correlation matrix to determine the most relevant features in predicting the housing price and to see if there exists a correlation or not. The correlation matrix is shown in Fig. 1.

As can be seen, our target variable MEDV has strong correlation with RM (0.7) and LSTAT (-0.74). Now, we plot individual correlation graphs of MEDV with these two features in figures Fig. 2 and Fig. 3.

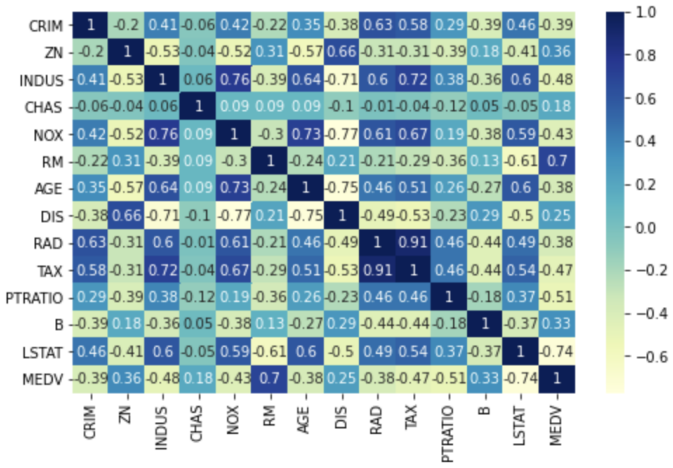

Fig. 1: Correlation Matrix

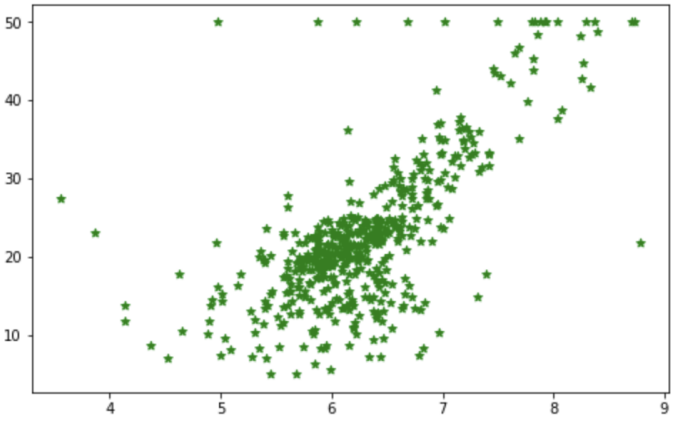

Fig. 2: Correlation of RM with MEDV

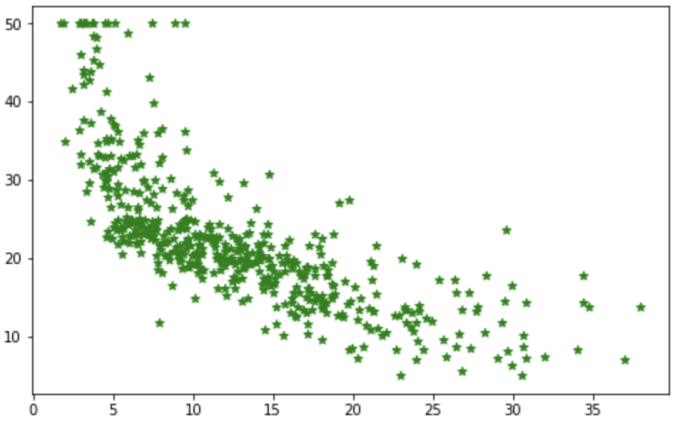

Fig. 3: Correlation of LSTAT with MEDV

D. Models Used

1) Linear Regression Model: This model works on the basis of how a variable is related to another variable. This works best with higher correlation value between the two variables. A regression model consists of two variables: 1) Independent variable

or regressor or predictor, 2) Dependent variable or target variable. The model learns the correlation between the independent variable and dependent variable and then apply the model to predict target variable for unseen values of independent variable. The Linear Regression model is defined by the equation:

$$y = \beta_0 + \beta_1 X$$

2) MLP Regression Model: This is a regression model that is based on Deep Learning technique of Multi Layer Perceptron. An MLP is created by interconnecting perceptrons in multiple layers. First layer is the input layer and last layer is the output layer. The layers in between are the hidden layers and there can be more than one hidden layers. The python library sklearn.neural_network provides for a parameter to configure the model with the required number of layers and required number of perceptrons in every layer.

E. Metric Used

The following metrics are used to evaluate the performance of the models used for regression.

- *Mean Absolute Error (MAE)* - This tells us the variations in between the expected and actual values of the predictions.

  $\text{MAE} = \frac{1}{n}\sum_{t=1}^{n}|y_i - x_i|$ Where y_i is the predicted value and x_i is the actual value for the i^{th} instance

- *Mean Squared Error (MSE)* - This measures the average of the sum of the squared errors.

  $\text{MSE} = \frac{1}{n}\sum_{t=1}^{n}(y_i - x_i)^2$ Where y_i is the predicted value and x_i is the actual value for the i^{th} instance

- *Root Mean Squared Error (RMSE)* - This is just the squared root of MSE, this frequently used as a measure of difference over MSE as the units end up being the same as original

  $\text{RMSE} = \sqrt{\frac{1}{n}\sum_{t=1}^{n}(y_i - x_i)^2}$ Where y_i is the predicted value and x_i is the actual value for the i^{th} instance

- *R2 Score (Coefficient of Determination)* - This is a statistical measure of how close the actual values are to the fitted regression line.

  $r^2 = \frac{\sum_{i=1}^{n}(\hat{y}_i - \bar{y})^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}$. Where y_i is the actual value, $\hat{y}_i$ is the predicted value and $\bar{y}$ for the i^{th} instance

IV. Results and Future Scope

Linear Regression and MLP Regression (two variations) were fitted and compared against each other using MAE, MSE, RMSE and R2 as the evaluation metric. Following table shows the SSE of three different models:

Method	MAE	MSE	RMSE	R2
Linear Regression	3.14	20.72	4.55	0.72
MLP with Hidden=(26)	2.33	12.44	3.52	0.83
MLP with Hidden=(20,26,10)	2.01	10.83	3.29	0.85

As is evident from the table above, an increase in the number of layers and perceptrons per layer decreases the error of the model.

Following graphs plot the predictions using each of the models in the above table:

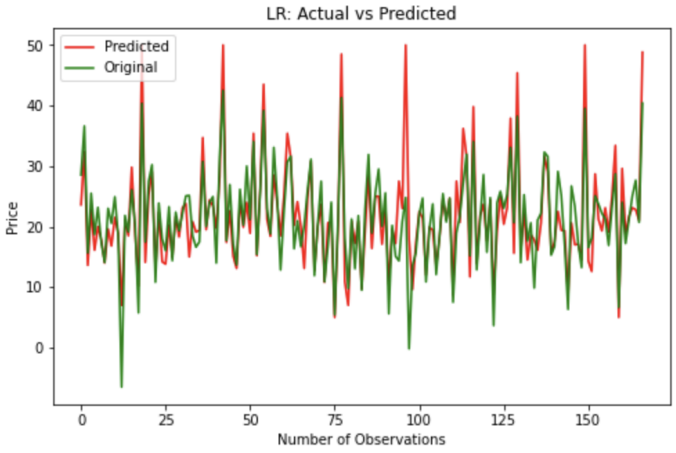

Fig. 4: Actual vs Prediction using LR

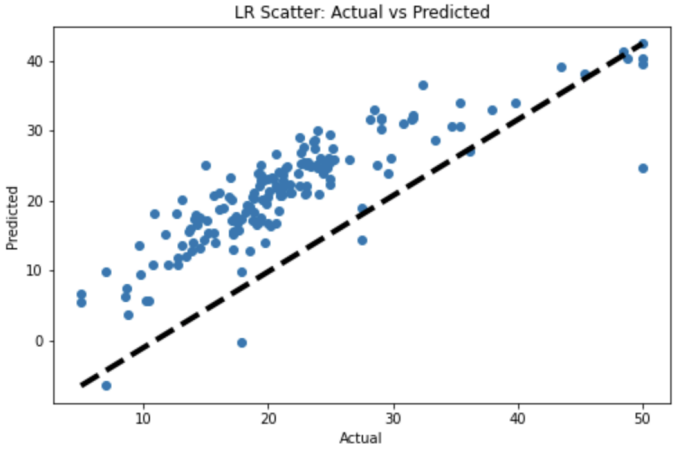

Fig. 5: Scatter: Actual vs Prediction using LR

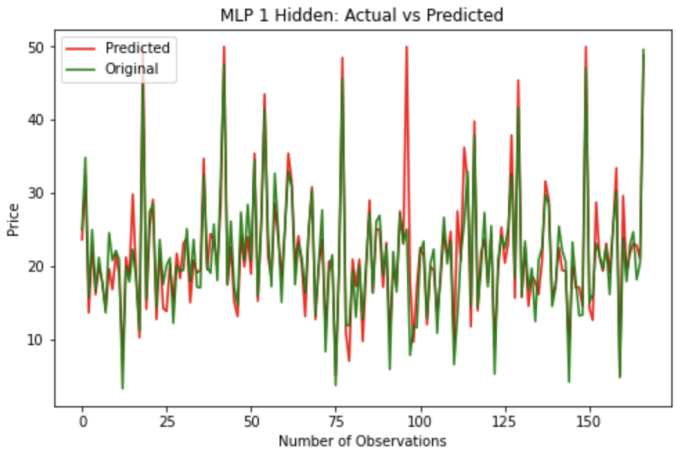

Fig. 6: Actual vs Prediction using MLP Hidden=(26)

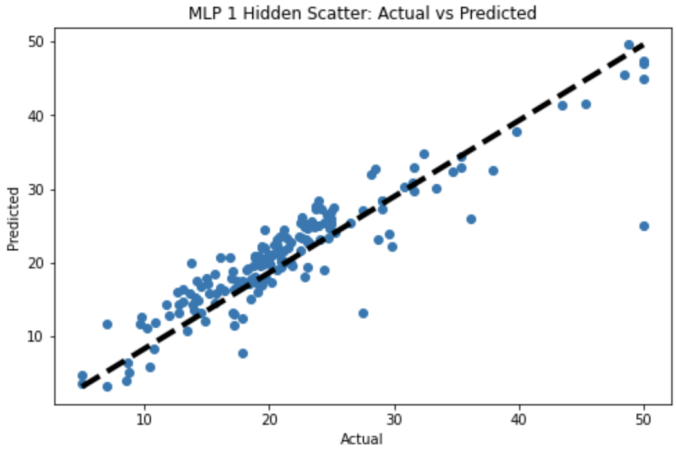

Fig. 7: Scatter: Actual vs Prediction using MLP Hidden=(26)

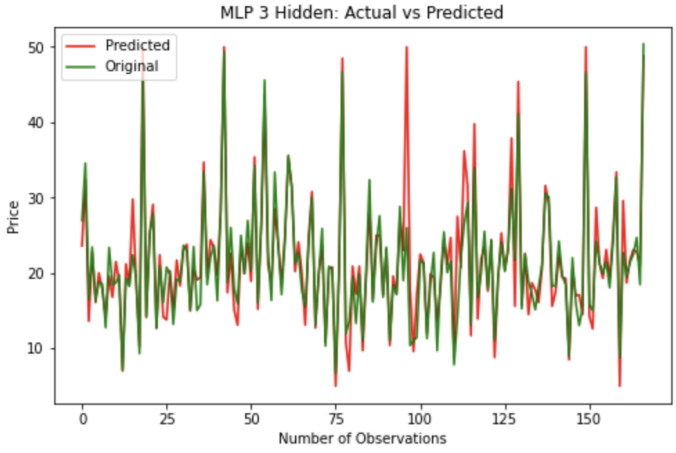

Fig. 8: Actual vs Prediction using MLP Hidden=(20,26,10)

Conclusion: As we can see from the results and the graphs above, Deep Learning techniques are far more accurate than simple Machine Learning algorithms such as Linear Regression. During our experiments, it was also noted that the error changes as we change the configuration (number of layers and number of perceptrons per layer) of the MLP Regression algorithm.

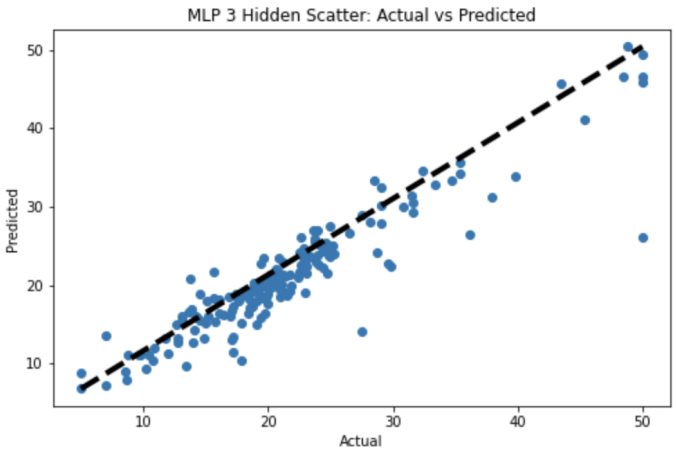

Fig. 9: Scatter: Actual vs Prediction using MLP Hidden=(20,26,10)

Future Scope: Simple linear regression and basic Deep Learning methodologies are used in this study to predict the price of the housing. More work can be done in order to understand how to arrive at the most optimum configuration for the current methodologies used in this study. Apart from this, more advanced techniques such as Recursive Neural Network (RNN) and Convolutional Neural Network (CNN) can be employed to produce more accurate results.

REFERENCES

[1] Index of /ml/machine-learning-databases/housing. https://archive.ics.uci.edu/ml/machine-learning-databases/housing/.

[2] Machine Learning Project: Predicting Boston House Prices With Regression. https://towardsdatascience.com/machine-learning-project-predicting-boston-house-prices-with-regression-b4e47493633d.

[3] E. L. Lydia, G. H. Bindu, A. Sirisham, and P. P. Kiran. Electronic governance of housing price using boston dataset implementing through deep learning mechanism. *International Journal of Recent Technology and Engineering (IJRTE) ISSN*, pages 2277–3878.

[4] J. Mu, F. Wu, and A. Zhang. Housing value forecasting based on machine learning methods. In *Abstract and Applied Analysis*, volume 2014. Hindawi, 2014.