Predicting Housing Prices using Machine Learning Techniques

B.Balakumar[1], P.Raviraj[2], S.Essakkiammal[3]

[1]*Assistant Professor, Centre for Information Technology and Engineering, Manonmaniam Sundaranar University, Tirunelveli, India,*
balakumarmsu@gmail.com
[2]*Professor, Department of CSE, GSSS Institute of Engineering and Technology for Women,KRS Road,Metagalli,Mysore, Karnataka-570016,*
drpraviraj@gmail.com
[3]*PG Scholar, Centre for Information Technology and Engineering, Manonmaniam Sundaranar University, Tirunelveli, India,*
essakkiammal0202@gmail.com

ABSTRACT:

This project, apply basic machine learning concepts on data collected for housing prices, to predict the selling price of a new home. First explore the data to obtain important features and descriptive statistics about the dataset. Next, properly split the data into testing and training subsets, and determine a suitable performance metric for this problem. Then analyze performance graphs for a learning algorithm with varying parameters and training set sizes. This will enable you to pick the optimal model that best generalizes for unseen data. Finally, test this optimal model on a new sample and compare the predicted selling price to our statistics.

General Terms

Machine Learning

Keywords- Machine Learning**,** DecisionTree Regressor, Performance metrics,GridSearchCV.

I. INTRODUCTION

Development of civilization is the foundation of increase of demand of houses day by day. Accurate prediction of house prices has been always a fascination for the buyers, sellers and for the bankers also. Many researchers have already worked to unravel the mysteries of the prediction of the house prices.There are many theories that have been given birth as a consequence of the research work contributed by the various researchers all over the world. Some of these theories believe that the geographical location and culture of a particular area determine how the home prices will increase or decrease whereas there are other schools of thought who emphasize the socio-economic conditions that largely play behind these house price rises. We all know that house price is a number from some defined assortment, so obviously prediction of prices of houses is a regression task. To forecast house price one person usually tries to locate similar properties at his or her neighborhood and based on collected data that person will try to predict the house price. All these indicate that house price prediction is an emerging research area of regression which requires the knowledge of machine learning. This has motivated to work in this domain.

II. RELATED WORK

There are two major challenges that researchers have to face. The biggest challenge is to identify the optimum number of features that will help to accurately predict the direction of the house prices. Kahn mentions that productivity growth in various residential construction sectors does impact the growth of the housing prices. The model that Kahn worked with shows how housing prices can have an appearently trendy appearance in which housing wealth rises faster than income for an extended period, then collapses and experiences an extended decline.

Lowrance mentions in his doctoral thesis that he found the interior living space to be the most influential factor determining the housing prices with his research work. He also cites the medium income of the census tract that holds the prices.

Pardoe utilizes features such as floor size, lot size category, number of bathrooms, and number of bedrooms, standardized age and garage size as features and utilizes linear regression techniques for predicting the house prices.

The second major challenge that is faced by the researchers is to find out the machine learning technique that will be the most effective when it comes to accurately predicting the house prices. Ng and Deisenroth constructs a cell phone based application using Gaussian processes for regression. Huet al. uses maximum information coefficient (MIC) to build accurate mathematical models for predicting house prices. Limsombunchao builds a model by using features like house size, house age, house type, number of bedrooms, number of bathrooms, number of garages, amenities around the house and geographical location. His work on the house price issue in New Zealand compared accuracy performance between Hedonic and Artificial Neural Network models and observed that neural networks perform better compared to the hedonic models when it comes to accurately predicting the prices of the houses. Bork and Moller [3] uses time series based models for predicting the prices of the houses.

The present work is unique from all these works as instead of looking at the problem from the regression perspective that tries to predict a price for the house, the work constructs the problem as a classification problem i.e. predicting whether the price of the house will increase or decrease.

Install

This project requires **Python** and the following Python libraries installed:

- NumPy
- Pandas
- matplotlib
- scikit-learn

You will also need to have software installed to run and execute a Jupyter Notebook

If you do not have Python installed yet, it is highly recommended that you install the Anaconda distribution of Python, which already has the above packages and more included.

Data

The modified housing dataset consists of 489 data points, with each datapoint having 3 features. This dataset is a modified version of the Housing dataset found on the UCI Machine Learning Repository.

Features

1. RM: average number of rooms per dwelling
2. LSTAT: percentage of population considered lower status
3. PTRATIO: pupil-teacher ratio by town

Target Variable 4. MEDV: median value of owner-occupied homes

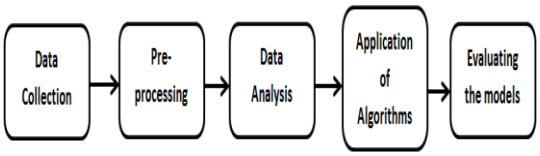
Data Exploration

In this first section of this project, you will make a cursory investigation about the housing data and provide your observations. Familiarizing yourself with the data through an explorative process is a fundamental practice to help you better understand and justify your results.

Since the main goal of this project is to construct a working model which has the capability of predicting the value of houses, we will need to separate the dataset into **features** and the **target variable**. The **features**, 'RM', 'LSTAT', and 'PTRATIO', give us quantitative information about each data point. The **target variable**, 'MEDV', will be the variable we seek to predict. These are stored in features and prices, respectively.

III. METHODOLOGY

Methodology represents a description about the framework that is undertaken. It consists of various milestones that need to be achieved in order to fulfil the objective. We have undertaken different data mining and machine learning concepts.



Developing a Model

In this project,develop the tools and techniques necessary for a model to make a prediction. Being able to make accurate evaluations of each model's performance through the use of these tools and techniques helps to greatly reinforce the confidence in your predictions.

Implementation: Define a Performance Metric

It is difficult to measure the quality of a given model without quantifying its performance over training and testing. This is typically done using some type of performance metric, whether it is through calculating some type of error, the goodness of fit, or some other useful measurement. For this project, you will be calculating the coefficient of determination, R^2, to quantify your model's performance. The coefficient of determination for a model is a useful statistic in regression analysis, as it often describes how "good" that model is at making predictions.

The values for R^2 range from 0 to 1, which captures the percentage of squared correlation between the predicted and actual values of the **target variable**. A model with an R^2 of 0 always fails to predict the target variable, whereas a model with an R^2 of 1 perfectly predicts the target variable. Any value between 0 and 1 indicates what percentage of the target variable, using this model, can be explained by the **features**. A model can be given a negative R^2 as well, which indicates that the model is no better than one that naively predicts the mean of the target variable.

For the performance_metric function in the code cell below, you will need to implement the following:

- Use r2_score from sklearn.metrics to perform a performance calculation between y_true and y_predict.
- Assign the performance score to the score variable.

```
In [4]: # TODO: Import 'r2_score'
from sklearn.metrics import r2_score

def performance_metric(y_true, y_predict):
    """ Calculates and returns the performance score between
        true and predicted values based on the metric chosen. """

    # TODO: Calculate the performance score between 'y_true' and
    score = r2_score(y_true, y_predict)

    # Return the score
    return score
```

Goodness of Fit

Assume that a dataset contains five data points and a model made the following predictions for the target variable:

True Value	Prediction
3.0	2.5
-0.5	0.0
2.0	2.1
7.0	7.8
4.2	5.3

Run the code cell below to use the performance_metric function and calculate this model's coefficient of determination

```
In [5]: # Calculate the performance of this model
score = performance_metric([3, -0.5, 2, 7, 4.2], [2.5, 0.0, 2.1, 7.8, 5.3])
print("Model has a coefficient of determination, R^2, of {:.3f}.".format(score))
```

```
Model has a coefficient of determination, R^2, of 0.923.
```

Analyzing Model Performance

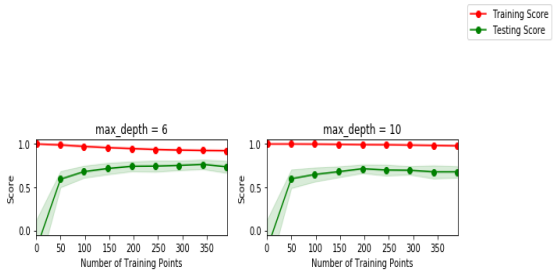
In this third section of the project, you'll take a look at several models' learning and testing performances on various subsets of training data. Additionally, you'll investigate one particular algorithm with an increasing 'max_depth' parameter on the full training set to observe how model complexity affects performance. Graphing your model's performance based on varying criteria can be beneficial in the analysis process, such as visualizing behavior that may not have been apparent from the results alone.

Learning Curves

The following code cell produces four graphs for a decision tree model with different maximum depths. Each graph visualizes the learning curves of the model for both training and testing as the size of the training set is increased. Note that the shaded reigon of a learning curve denotes the uncertainty of that curve (measured as the standard deviation). The model is scored on both the training and testing sets using R^2, the coefficient of determination.

Run the code cell below and use these graphs to answer the following question.

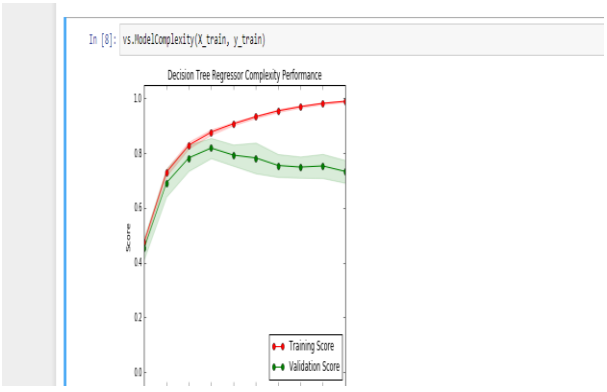
```
In [7]: # Produce learning curves for varying training set sizes and maximum depths
vs.ModelLearning(features, prices)
```



Complexity Curves

The following code cell produces a graph for a decision tree model that has been trained and validated on the training data using different maximum depths. The graph produces two complexity curves — one for training and one for validation. Similar to the **learning curves**, the shaded regions of both the complexity curves denote the uncertainty in those curves, and the model is scored on both the training and validation sets using the performance_metricfunction.

Run the code cell below and use this graph to answer the following two questions.



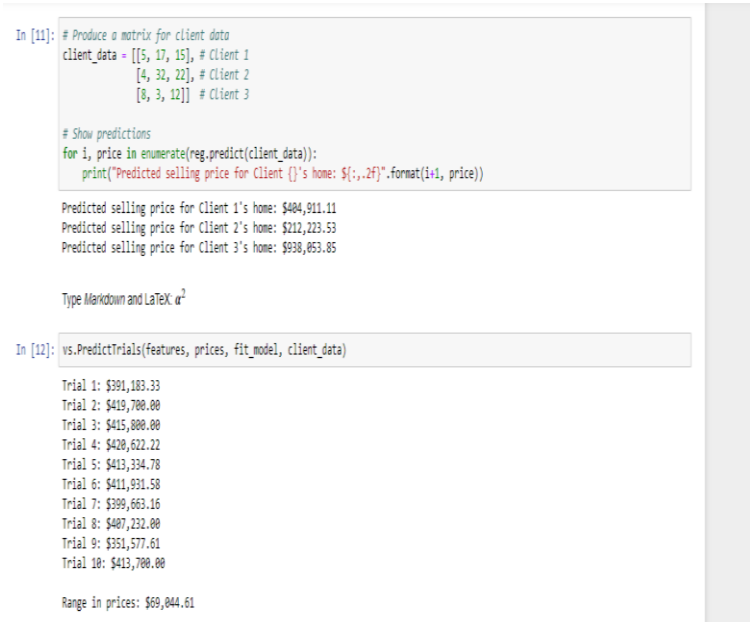
RESULT

Predicting Selling Prices

Imagine that you were a real estate agent in the Boston area looking to use this model to help price homes owned by your clients that they wish to sell. You have collected the following information from three of your clients:

Feature	Client 1	Client 2	Client 3
Total number of rooms in home	5 rooms	4 rooms	8 rooms
Household net worth (income)	Top 34th percent	Bottom 45th percent	Top 7th percent
Student-teacher ratio of nearby schools	15-to-1	22-to-1	12-to-1

Run the code block below to have your optimized model make predictions for each client's home.



CONCLUSION

In this paper, we have used machine learning algorithms to predict the house prices. We have mentioned the step by step procedure to analyze the dataset. These

feature set were then given as an input to four algorithms and a csv file was generated consisting of predicted house prices. Hence we calculated the performance of each model using different performance metrics and compared them based on these metrics. We found that Decision Tree overfits our dataset and gives the highest accuracy of 84.64. Thus we conclude that we implemented classifiers to the problem of regression to check how well can classifier fit to regression problem.

FUTURE WORK

We recommend that working on large dataset would yield a better and real picture about the model. We have undertaken only few Machine Learning algorithms that are actually classifiers but we need to train many other classifiers and understand their predicting behavior for continuous values too. By improving the error values this research work can be useful for development of applications for various respective cities.

REFERENCES

[1] https://github.com/udacity/machine-learning/tree/master/projects/boston_housing

[2] Lowrance, E.R.: Predicting the market value of single-family residential real estate. 1st edn. PhD diss., New York University, (2015).

[3] Bork, M., Moller, V.S.: House price forecast ability: a factor analysis. Real Estate Economics. Heidelberg (2016).

[4] Ng, A., Deisenroth, M.: Machine learning for a London housing price prediction mobile application. Imperial College London, (2015).

[5] Limsombunchai, Visit. "House price prediction: hedonic price model vs. artificial neural network."New Zealand Agricultural and Resource Economics Society Conference.

[6] Limsombunchao, V.: House price prediction: hedonic price model vs. artificial neural network. Lincoln University, NZ.

[7]https://www.kaggle.com/ohmets/feature-selection-forregression/Data

[8] Pow, Nissan, Emil Janulewicz, and L. Liu. "Applied Machine Learning Project 4 Prediction of real estate property prices in Montréal".