PROJECT REPORT ON HOUSING PRICE PREDICTION

Submitted by:

SATU VINAYAK POLE

ACKNOWLEDGMENT

I would like to express my special gratitude to my SME Mr. Shwetank Mishra as well as “Flip Robo” team for letting me work on “Housing Price Prediction” project. Also thanks to my institute ‘Data Trained’ .

Also I would like to thank websites such as StackOverflow, geeksforgeeks and Youtube who has helped me in solving issues and errors.

INTRODUCTION

Business Problem Framing

Houses is the basic need of mankind and since it the most basic need it is a very large market and there are various companies working in this domain. Data science comes as a very important tool to solve problems in the domain to help the companies increase their overall revenue, profits, improving their marketing strategies and focusing on changing trends in house sales and purchases.

House price prediction can help the person determine the selling price of a house and can help the customer to arrange the right time to purchase a house. House Price prediction, is important to drive Real Estate efficiency. The aim is to predict the efficient house pricing for real estate customers with respect to their budgets and priorities. By analysing previous market trends and price ranges, and also upcoming developments future prices are predicted with cost of property depending on number of attributes considered.

Now as a data scientist our work is to analyse the dataset and apply our skills towards predicting house price.

Conceptual Background of the Domain Problem

A US based company names 'Surprise Housing' wants to enter Australian market for selling houses and earning fortune from them. They wants to buy houses for a prices that is below actual price and and sell those houses at higher prices.Ultimately they don't know anything about the prices of Australian market so they take old data of houses that has all the features and with respect to their prices. Now they want to know which features such as garage, house floor, fireplaces, etc.predict the price of house.

As a data scientist I need to provide them with a model which will consider all the features that are or not responsible for the price of house.This model will further help them to predict the price of the house they want to buy.

Review of Literature

The factors that affect the land price have to be studied and their impact on price has also to be modelled. An analysis of the past data is to be considered. As the real estate is fast developing sector, the analysis and forecast of land prices using mathematical modelling and other scientific techniques is an immediate urgent need for decision making by all those concerned. The increase in population as well as the industrial activity is attributed to various factors, the most prominent being the recent spurt in the knowledge sector viz. Information Technology (IT) and Information technology enabled services. Demand for land started of showing an upward trend and housing and the real estate activity started booming. The need for predicting the trend in land prices was felt by all in the industry viz. the Government, the regulating bodies, lending institutions, the developers and the investors. Therefore, in this project report, we present various important features to use while predicting housing prices with good accuracy. We can use regression models, using various features to have lower

Residual Sum of Squares error. While using features in a regression model some feature engineering is required for better prediction.

The primary aim of this report is to use these Machine Learning Techniques and provide ML models which can then serve the users. The main objective of a Buyer is to search for their dream house which has all the amenities they need. Furthermore, they look for these houses/Real estates with a price in mind and there is no guarantee that they will get the product for a deserving price and not overpriced. Similarly, A seller looks for a certain number that they can put on the estate as a price tag and this cannot be just a wild guess, lots of research needs to be put to conclude a valuation of a house

Motivation for the Problem Undertaken

In this case I am provided with csv file /data using which I will need to data cleaning provide them with a best non overfitting regression model that will help them in buying and selling homes. They can accordingly manipulate the strategy of the firm and concentrate on areas that will yield high returns. Further, the model will be a good way for the management to understand the pricing dynamics of a new market. The **relationship between house prices and the economy** is an important motivating factor for predicting house prices.

Analytical Problem Framing

This particular problem has two datasets namely train and test data. After uploading the datasets using pandas library in python I got to know that train dataset has a shape of 1168 rows and 81 columns while test had the shape of 292 rows and 80 columns. It was known that train dataset had the label present i.e. 'Sale_Price'. Looking at the label it was known that it was going to be a regression problem.

Further during data cleaning it was found out that both dataset contained no duplicates but there were null values present. After looking at the amount of null values 5 columns were dropped as the null values in a column were >40%. The dropped column names were PoolQC , Fence and MiscFeature. Other remaning columns with null values were treated using pandas fillna function wherein continuous type columns were filled with mean of the same columns while incase of categorical columns null values were filled with mode of the same column.

Further it was understood that ID column had only unique ID's/numbers that are provided to each house. Also the utility column consisted of only 1 kind of data. Since such type of column data are not useful for us they were dropped from the column.

Further the data was plotted and analysed, their skewness were removed, they were checked for outliers and further they were used for feature selection methods wherein some columns were dropped as they were not useful in model building, then VIF was plotted which showed us that some columns showed multicorrelation with other features. Hence they were dropped too, and after that a model was produced.

Data Sources and their formats

This data was provided to me by Flip Robo technologies in csv (comma separated values) format.

Also, I was having two datasets one is train and other is test. I have built model using train dat aset and predicted SalePrice for test dataset. My train dataset was having 1168 rows and 81 columns including target, and my test dataset was having 292 rows and 80 columns excludi ng target. In this particular datasets I have object, float and integer types of data.

Data Preprocessing Done

- First both datasets were imported using pandas libraray.
- Then they were checked for null and duplicate vales. There were no duplicate values found in dataset, many columns had null values present in them.In 5 columns null values were >40% hence they were dropped immediately as this columns hamper the model prediction.
- It was also found that ID column and utlities column had a type of data that was not at all useful for us, hence these columns were dropped.
- Further null values were treated using mean(incase of continuous data) and mode(incase of categorical data).
- Further columns were checked if there are any outliers present in them using boxplots. By looking at the plots it was found that outliers present in dataset were >40%. Deleting these outliers was not good for our model performance and they were not dropped.
- Further both datasets were checked for skewness and distplots were plotted to confirm their skewness.
- Further skewness was removed and both the datasets were tried into feature selecction techniques wherein columns that are not related to label were dropped and columns that showed multicollinearity problem were dropped after using VIF.

Data Inputs- Logic- Output Relationships

I have used box plot for each pair of categorical features that shows the relation with the median sale price for all the sub categories in each categorical feature.
And also for continuous numerical variables I have used scatterplot to show the relationship between continuous numerical variable and target variable.

Hardware and Software Requirements and Tools Used

Hardare used to complete this model is
Model: ASUS TUF A15
Processor: AMD RYZEN 5 4600H OCTA CORE
RAM: 8GB
ROM:500 GB SSD
Software used to complete this mode is:
Software: Jupyter Notebook,Python, Anaconda Library

Libraries used:

1. Numpy
2. Pandas
3. Scikit-learn
4. Seaborn
5. matplotlib

Model/s Development and Evaluation

Identification of possible problem-solving approaches (methods)

I have used imputation technique to fill null values. To remove skewness I have used Power Transformer technique using yeo-johnson method. Also I have standardized the data using Standard Scalar method. I have used k-best feature selection method with k-classif and used Variance Inflation Factor(VIF)to check for multicollinearity within features.

Testing of Identified Approaches (Algorithms)

Since 'sale price' was continuous column I knew that it is regression problem hence I have used Regression algorithms to make model. I have algorithms such as:

- Random Forest Regressor
- AdaBoost Regressor
- Decision Tree Regressor
- Gradient Boosting Regressor.

Run and Evaluate selected models

```
▾ RandomForestRegressor
RandomForestRegressor()
```

```
# since thge model is already trained, below code will help to predict based on train and test data

y_pred=rf.predict(x_train)

pred=rf.predict(x_test)

#printing r2 score for testing and training models.
#r2 score give value of how good the model has studied and learnt the data

print(f'training R2 score:{r2_score(y_train,y_pred)*100:.2f}%')
print(f'testing R2 score:{r2_score(y_test,pred)*100:.2f}%')
```

```
training R2 score:96.94%
testing R2 score:87.99%
```

```
Cross Validation Score for Random Forest regressor model :-  82.08811776951728
```

```
]:
#finding mean absolute error() for above model(MAE)
print('mean absolute error',mean_absolute_error(y_test,pred))

#finding root mean_squared_error(RMSE)
print('root mean squared error',np.sqrt(mean_squared_error(y_test,pred)))
```

```
mean absolute error 18423.804383561648
root mean squared error 25523.311373665332
```

The above snapshot provided is Random Forest model where training and testing accuracy alongwith cross validation scores and RMSE scores are mentioned.

2)AdaBoost

```
AdaBoostRegressor
AdaBoostRegressor()
```

```
# since thge model is already trained, below code will help to predict based on train and test data

y_pred=ab.predict(x_train)

pred=ab.predict(x_test)

#printing r2 score for testing and training models.
#r2 score give value of how good the model has studied and learnt the data

print(f'training R2 score:{r2_score(y_train,y_pred)*100:.2f}%')
print(f'testing R2 score:{r2_score(y_test,pred)*100:.2f}%')
```

```
training R2 score:84.08%
testing R2 score:80.31%
```

```
#cross validation score
print('Cross Validation Score for AdaBoost regressor model :- ',((cross_val_score(ab,x_scaled,Y,cv=3).mean())*100))
```

```
Cross Validation Score for AdaBoost regressor model :-  75.28698601155097
```

```
#finding mean absolute error() for above model(MAE)
print('mean absolute error',mean_absolute_error(y_test,pred))

#finding root mean_squared_error(RMSE)
print('root mean squared error',np.sqrt(mean_squared_error(y_test,pred)))
```

```
mean absolute error 25717.884181392215
root mean squared error 32685.338511642072
```

The above snapshot provided is AdaBoost model where training and testing accuracy alongwith cross validation scores and RMSE scores are mentioned.

3)

```
dt.fit(x_train,y_train)
```

```
DecisionTreeRegressor
DecisionTreeRegressor()
```

```
# since thge model is already trained, below code will help to predict based on train and test data

y_pred=dt.predict(x_train)

pred=dt.predict(x_test)

#printing r2 score for testing and training models.
#r2 score give value of how good the model has studied and learnt the data

print(f'training R2 score:{r2_score(y_train,y_pred)*100:.2f}%')
print(f'testing R2 score:{r2_score(y_test,pred)*100:.2f}%')
```

```
training R2 score:100.00%
testing R2 score:68.14%
```

```
#cross validation score
print('Cross Validation Score for Decision Tree regressor model :- ',((cross_val_score(dt,x_scaled,Y,cv=3).mean())*100))
```

```
Cross Validation Score for Decision Tree regressor model :-  57.56391766450037
```

```
#finding mean absolute error() for above model(MAE)
print('mean absolute error',mean_absolute_error(y_test,pred))

#finding root mean_squared_error(RMSE)
print('root mean squared error',np.sqrt(mean_squared_error(y_test,pred)))
```

```
mean absolute error 28036.883561643837
root mean squared error 41571.530201833055
```

The above snapshot provided is Decision Tree model where training and testing accuracy alongwith cross validation scores and RMSE scores are mentioned.

4

```
gbdt.fit(x_train,y_train)
```

```
GradientBoostingRegressor
GradientBoostingRegressor()
```

```
# since thge model is already trained, below code will help to predict based on train and test data

y_pred=gbdt.predict(x_train)

pred=gbdt.predict(x_test)

#printing r2 score for testing and training models.
#r2 score give value of how good the model has studied and learnt the data

print(f'training R2 score:{r2_score(y_train,y_pred)*100:.2f}%')
print(f'testing R2 score:{r2_score(y_test,pred)*100:.2f}%')
```

```
training R2 score:95.02%
testing R2 score:88.83%
```

```
#cross validation score
print('Cross Validation Score for Gradient Boosting regressor model :- ',((cross_val_score(gbdt,x_scaled,Y,cv=3).mean())*100)
```

```
Cross Validation Score for Gradient Boosting regressor model :-  82.16519848777
```

```
#finding mean absolute error() for above model(MAE)
print('mean absolute error',mean_absolute_error(y_test,pred))

#finding root mean_squared_error(RMSE)
print('root mean squared error',np.sqrt(mean_squared_error(y_test,pred)))
```

```
mean absolute error 18032.54872671405
root mean squared error 24612.329176618874
```

The above snapshot provided is Gradient boosting model where training and testing accuracy alongwith cross validation scores and RMSE scores are mentioned.

After all the 4 models were trained, the best model chosen for hyperparameter tuning was Gradient Boosting as it had better testing accuracy then random forests and its RMSE value was low then Random Forest model.

5)

```
gbdt1.fit(x_train,y_train)
```

```
GradientBoostingRegressor(learning_rate=0.3, loss='huber', max_features='auto')
```

```
# since thge model is already trained, below code will help to predict based on train and test data

y_pred=gbdt1.predict(x_train)

pred=gbdt1.predict(x_test)

#printing r2 score for testing and training models.
#r2 score give value of how good the model has studied and learnt the data

print(f'training R2 score:{r2_score(y_train,y_pred)*100:.2f}%')
print(f'testing R2 score:{r2_score(y_test,pred)*100:.2f}%')
```

```
training R2 score:97.55%
testing R2 score:86.36%
```

```
#cross validation score
print('Cross Validation Score for Random forest tuned model :- ',((cross_val_score(gbdt1,x_scaled,Y,cv=3).mean())*100))
```

```
Cross Validation Score for Random forest tuned model :-  83.65650440496631
```

```
#finding mean absolute error() for above model(MAE)
print('mean absolute error',mean_absolute_error(y_test,pred))

#finding root mean_squared_error(RMSE)
print('root mean squared error',np.sqrt(mean_squared_error(y_test,pred)))
```

```
mean absolute error 19001.339418773947
root mean squared error 27197.375035979392
```

The above snapshot provided is of tuned Gradient boosting model where training and testing accuracy alongwith cross validation scores and RMSE scores are mentioned.
After tuning it was known that original model performed better than tuned model, hence original model with name ‘gbdt’ was saved.

Key Metrics for success in solving problem under consideration

I have used the following metrics for evaluation:

- I have used cross validation scores to predict if the model is overfitted or not.
- I have used root mean square deviation is one of the most commonly used measures for evaluating the quality of predictions.
- I have used r2 score which tells us how accurate our model is.

CONCLUSION

Key Findings and Conclusions of the Study

In this project report, we have used machine learning algorithms to predict the house prices. We have mentioned the step by step procedure to analyze the dataset and finding the correlation between the features. Thus we can select the features which are not correlated to each other and are independent in nature. These feature set were then given as an input to five 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. Then we have also saved the dataframe of predicted prices of test dataset.

Learning Outcomes of the Study in respect of Data Science

I found that the dataset was quite interesting to handle as it contains all types of data in it. Improvement in computing technology has made it possible to examine social information that cannot previously be captured, processed and analysed. New analytical techniques of machine learning can be used in property research. The power of visualization has helped us in understanding the data by graphical representation it has made me to understand what data is trying to say. Data cleaning is one of the most important steps to remove missing value and to replace null value and zero values with there respective mean, median or mode. This study is an exploratory attempt to use five machine learning algorithms in estimating housing prices, and then compare their results.

To conclude, the application of machine learning in property research is still at an early stage. We hope this study has moved a small step ahead in providing some methodological and empirical contributions to property appraisal, and presenting an alternative approach to the valuation of housing prices. Future direction of research may consider incorporating additional property transaction data from a larger geographical location with more features, or analysing other property types beyond housing development.