**HOUSE USE CASE STUDY PROJECT**

SUBMITTED BY: **Sindhu Shree N**

INTERNSHIP BATCH: **19**

**ACKNOWLEDGEMENT:**

It is not possible to prepare a project without the assistance and encouragement of other people. This one is certainly no exception. I would like to extend my sincere thanks to all of them.

I would like to extend the deep sense of thanks and gratitude to the research papers and scholarly articles that are helped me to complete this project successfully.

**INTRODUCTION:**

Houses are one of the necessary needs of each and every person around the globe and therefore housing and real estate market is one of the markets which is one of the major contributors in the world’s economy. It is a very large market and there are various companies working in the 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. Predictive modelling, Market mix modelling, recommendation systems are some of the machine learning techniques used for achieving the business goals for housing companies. Our problem is related to one such housing company. A US-based housing company named Surprise Housing has decided to enter the Australian market. The company uses data analytics to purchase houses at a price below their actual values and flip them at a higher price. For the same purpose, the company has collected a data set from the sale of houses in Australia. The data is provided in the CSV file below. The company is looking at prospective properties to buy houses to enter the market. You are required to build a model using Machine Learning in order to predict the actual value of the prospective properties and decide whether to invest in them or not. For this company wants to know: Which variables are important to predict the price of variable? • How do these variables describe the price of the house?

**Conceptual Background of the domain problem:**

An increase in house demand occurs each year, indirectly causing house price increases every year. The problem arises when there are numerous variables such as “Remodified year” and “OverallCondition” that may influence the house price, thus most stakeholders including buyers and developers, house builders and the real estate industry would like to know the exact attributes or the accurate factors influencing the house price to help investors make decisions and help house builders set the house price.

**Review of Literature:**

In this project the Literature review focuses on predicting house price based on the model of machine learning as well as analysing attributes primarily that affect house price.

**Motivation for the problem undertaken:**

The housing market is highly competitive, and I want to be the best real estate agent in the area. To compete with my peers, I decide to leverage a few basic machine learning concepts to assist myself with finding the best-selling price for their home. Luckily, I've come across the Housing dataset which contains aggregated data on various features for houses in Greater communities, including the median value of homes for each of those areas. My task is to build an optimal model based on a statistical analysis with the tools available. This model will then be used to estimate the best-selling price for homes.

**Analytical Problem Framing:**

This project aims to create a house price prediction model using regression to obtain optimal prediction results. Regression is used to determine the optimal coefficient in prediction. In this study, I wanted to know the performance of the developed model in time series data.

**Data Sources and their formats.**

The source data for this project is based on a US-based housing company named **Surprise Housing** which has decided to enter the Australian market. The company uses data analytics to purchase houses at a price below their actual values and flip them at a higher price. For the same purpose, the company has collected a data set from the sale of houses in Australia. The data is provided in the CSV file.

**Data Pre-processing Done:**

Before applying machine learning models to the dataset one should clean the data and this one has no exception in that sense. We perform some of the Exploratory Data Analysis (EDA) process.

There are 80 input features they are: ‘Id’, ‘MSSubClass’, 'MSZoning', 'LotFrontage',

'LotArea’, 'Street', 'Alley', 'LotShape', 'LandContour', 'Utilities', 'LotConfig',

'LandSlope', 'Neighborhood', 'Condition1', 'Condition2', 'BldgType',

'HouseStyle', 'OverallQual', 'OverallCond', 'YearBuilt', 'YearRemodAdd',

'RoofStyle', 'RoofMatl', 'Exterior1st', 'Exterior2nd', 'MasVnrType',

'MasVnrArea', 'ExterQual', 'ExterCond', 'Foundation', 'BsmtQual',

'BsmtCond', 'BsmtExposure', 'BsmtFinType1', 'BsmtFinSF1',

'BsmtFinType2', 'BsmtFinSF2', 'BsmtUnfSF', 'TotalBsmtSF', 'Heating',

'HeatingQC', 'CentralAir', 'Electrical', '1stFlrSF', '2ndFlrSF',

'LowQualFinSF', 'GrLivArea', 'BsmtFullBath', 'BsmtHalfBath', 'FullBath',

'HalfBath', 'BedroomAbvGr', 'KitchenAbvGr', 'KitchenQual',

'TotRmsAbvGrd', 'Functional', 'Fireplaces', 'FireplaceQu', 'GarageType',

'GarageYrBlt', 'GarageFinish', 'GarageCars', 'GarageArea', 'GarageQual',

'GarageCond', 'PavedDrive', 'WoodDeckSF', 'OpenPorchSF',

'EnclosedPorch', '3SsnPorch', 'ScreenPorch', 'PoolArea', 'PoolQC',

'Fence', 'MiscFeature', 'MiscVal', 'MoSold', 'YrSold', 'SaleType',

'SaleCondition'.

Out of these 80 features we have dropped some features like ‘Id‘, ’Alley’, ’Fence’,

’PoolQC’, ’GarageYrBlt’, ‘MiscFeature’. Since Id is unique for every individual house it does not affect the Target feature in any way and all other features are having more than 50% of null values in their respective columns so it would definitely not affect the Price of the house

So I have dropped all of them. So now I have left out with 75 features. So in this 75 features

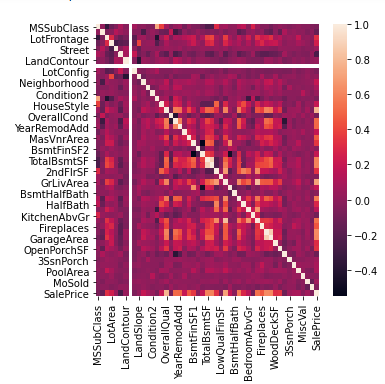
there are less percent of NaN values which are to be replaced by statistical values.

Since the dataset contains both numerical and categorical features I have replaced a numerical feature with the mean values and categorical feature with the median values and dropped some of the NaN value rows using dropna() function. And since we have training data and testing data separately I have done this cleaning process for the both.

For the model to proceed with the data efficiently, the categorical variables should be encoded. Data has to be pre-processed as a machine learning models are better at reading numbers than words. Using get\_dummies(), the categorical data can be replaced with numbers with increasing the features. But before encoding I have merged both train and test data row wise in order to avoid the mismatch in the number of columns. And scaled a data using the MinMaxScaler in order to obtain a better predictions.

**Data Inputs-Logic-Output Relationships:**

In this next step we are going to find the relations between the input feature and the target feature by using the correlation heatmap using seaborn library which shows the relation between each feature with the target feature which as shown below.



The darker shades shows highly negative correlation and the lighter shades shows highly positive correlation. Corresponding to the target feature we can see that “OverallCond” and “YearRemodAdd” features has very positive correlation with the target. Which implies the house having good facilities and if it is recently modified then that house will have more price. And remaining features are also contributing to the target that is they are also playing a major role in predicting price of the house.

**Hardware** **and software requirements and tools used:**

**Hardware requirements:**

**PROCESSOR**: Intel(R) Core(TM) i3 CPU

**MONITOR** : Any display unit

**HARD DISK** : 240GB SSD

**RAM** : 8.00GB

**Software requirements:**

**OPERATING SYSTEM**: Windows 10 Pro

**FRONT END** : Jupyter Notebook (Anaconda3)

**BACK END** : Excel 2013

**Tools Used:**

1) Scikit learn

2) Pandas Library

3) Numpy Library

4) Seaborn

5) Matplotlib

**Models Development and Evaluation:**

The target feature is a numerical (continuous) data we need to apply the regression models to predict the outputs. And since we have a training and testing data separately no need to apply train\_test\_split function to split the data. Directly we send the training set to train the model and testing set to predict the output.

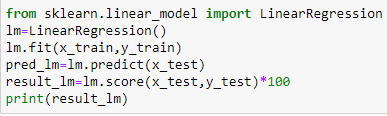
Here we are using six regression algorithms and selecting the best one out of it.

**Testing of Identified Approaches (Algorithms):**

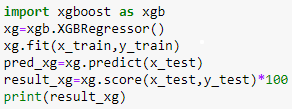
1. Linear Regression
2. XGBoost Regressor
3. Ridge Regressor
4. Decision Tree Regressor
5. Gradient Boosting Regressor
6. Random Forest Regressor

**Run and Evaluating selected models:**

1. **Linear Regression**: It is a basic and commonly used type of predictive analysis. Linear Regression fits a straight line or surface that minimizes the deprecancies between predicted and actual output values. Here it is giving an accuracy of 84.58%.



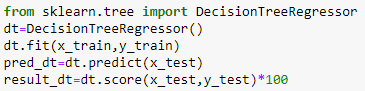
1. **XGBoost Regressor:** Extreme Gradient Boosting regressor provides an efficient and effective implementation of gradient boosting algorithm which is used for regression predictive modeling. This model is giving an accuracy of 83.13%.



### Ridge Regressor: It is a type of linear model which is similar to the linear regression used for predictive models which is giving an accuracy 84.29% for our present dataset.

### 

1. **Decision Tree Regressor**: This algorithm splits a data sample into two or more homogeneous sets based on the input variables. A part of a tree is generated with each split. As a result, a tree with decision nodes and leaf nodes is developed. A tree starts from a root node which is a best predictor. This model is giving an accuracy of 74.82%.



### Gradient Boosting Regressor: Gradient boosting is a machine learning technique for regression, which produces a prediction model in the form of an ensemble of weak prediction models, typically decision trees. Here it is giving an accuracy of 89.11%.

### 

### Random Forest Regressor: It is a type of ensemble learning method that uses numerous decision trees to achieve higher prediction accuracy and model stability. Every tree classifies a data instance based on attributes, and the forest chooses the classification that received most instances. For the present dataset this model is giving an accuracy of 88.18%.

### 

### Key metrics for success in solving problem under consideration.

### Evaluating a machine learning algorithm is an essential part of any project. The model may give the satisfying results when evaluated using metric but may give poor results when evaluated against other metrics. So here we have done two evaluation metrics they are:

### Mean Absolute Error:

### Mean Absolute Error is the average of the difference between the original values and the predicted values. It gives us the measure of how far the prediction were from the actual output. However, they don’ give us any idea of the direction of the error i.e. whether we are under predicting the data or over predicting the data. Mathematically, it is represented as:

### Mean Absolute Error =

### Mean Squared Error:

### Mean Squared Error (MSE) is quite similar to Mean Absolute Error, the only difference being that MSE takes the average of the square of the difference between the original values and the predicted values.

### Mean Squared Error = 2

### When I compared the key metrics with each other I found that Gradient Boosting Algorithm is giving a less number of these errors. Thus I picked that model as the best when compare to other models.

### Visualizations:

### 

### 

### 

### 

### Interpretation of the Results:

### Thus when we compared all the accuracy scores and metrics we got to know that the Gradient Boosting Regressor is giving a good accuracy along with the linear line when we plotted (scatter). Thus I have finalized that this model and tuned using hyper parameters in which its accuracy was improved.

### CONCLUSION:

### Key findings: Highest accuracy is seen in the Gradient Boosting Regressor which is similar to the k-fold cross validation score (for k=5) and also less MAE and MSE and improved accuracy after fine tuning of the same model.

Things I've learned by completing this project:

* How to use NumPy to investigate the latent features of a dataset.
* How to analyse various learning performance plots for variance and bias.
* How to determine the best-guess model for predictions from unseen data.
* How to evaluate a model's performance on unseen data using previous data.

**Limitations of this Work and Scope for Future work:**

Though we obtained a good model the predictions may not that accurate to the actual rate but yes it will definitely helps the real estate agents to know which features a playing a major role in estimating house price and they can implement those strategies to improve it in the future.