House Price Prediction

THAPAR INSTITUTE
OF ENGINEERING & TECHNOLOGY
(Deemed to be University)

Computer Science and Engineering Department

Thapar Institute of Engineering and Technology

(Deemed to be University), Patiala – 147004

Machine Learning Project

Submitted By:

Anchal

102103227

Chetna Sharma

102103396

Submitted To:

Ms. Kudratdeep Aulakh

Index

Sr. No.	**Content used**	**Page No.**
1.	Introduction	3
2	Libraries used	5
3.	Algorithm used	6
4.	Code and Screenshots	7

1.Introduction

1.1HousePricePrediction

Dataset link(.xlsx)
https://docs.google.com/spreadsheets/d/1EsFosB6JxawN201VJqBmBL3Av3ruMbBe/edit#gid=1199153200

1.2 Description

This dataset concerns the housing prices in the housing city of Boston. The dataset provided has instances with 13 features.

The Description of the dataset is shown below:

1	**Id**	To count the records.
2	**MSSubClass**	Identifies the type of dwelling involved in the sale.
3	**MSZoning**	Identifies the general zoning classification of the sale.
4	**LotArea**	Lot size in square feet.
5	**LotConfig**	Configuration of the lot
6	**BldgType**	Type of dwelling
7	**OverallCond**	Rates the overall condition of the house
8	**YearBuilt**	Original construction year

9	**YearRemodAdd**	Remodel date (same as construction date if no remodeling or additions).
10	**Exterior1st**	Exterior covering on house
11	**BsmtFinSF2**	Type 2 finished square feet.
12	**TotalBsmtSF**	Total square feet of basement area
13	**SalePrice**	To be predicted

2. Libraries Used

- **Pandas** A powerful data manipulation and analysis library in Python.
- **Scikit-learn** A widely-used machine learning library for building and analyzing models.
- **XGBoost** An optimized gradient boosting library for classification and regression tasks.
- **Matplotlib** A comprehensive plotting library for creating visualizations in Python.
- **Seaborn** A data visualization library that provides an enhanced interface to create aesthetic and informative statistical graphics.

3.Algorithm(s) Used

Random Forest Regression-

Random Forest is an ensemble technique that uses multiple of decision

trees and can be used for both regression and classification tasks.

Linear Regression-

Linear Regression predicts the final output-dependent value based on the given independent features. Like, here we have to predict SalePrice depending on features like MSSubClass, YearBuilt, BldgType, Exterior1st etc. To read more about Linear Regression refer this.

Regression Algorithm-

XGBoost Regressor is used in our project. XGBoost is an efficient implementation of gradient boosting that can be used for regression predictive modeling.

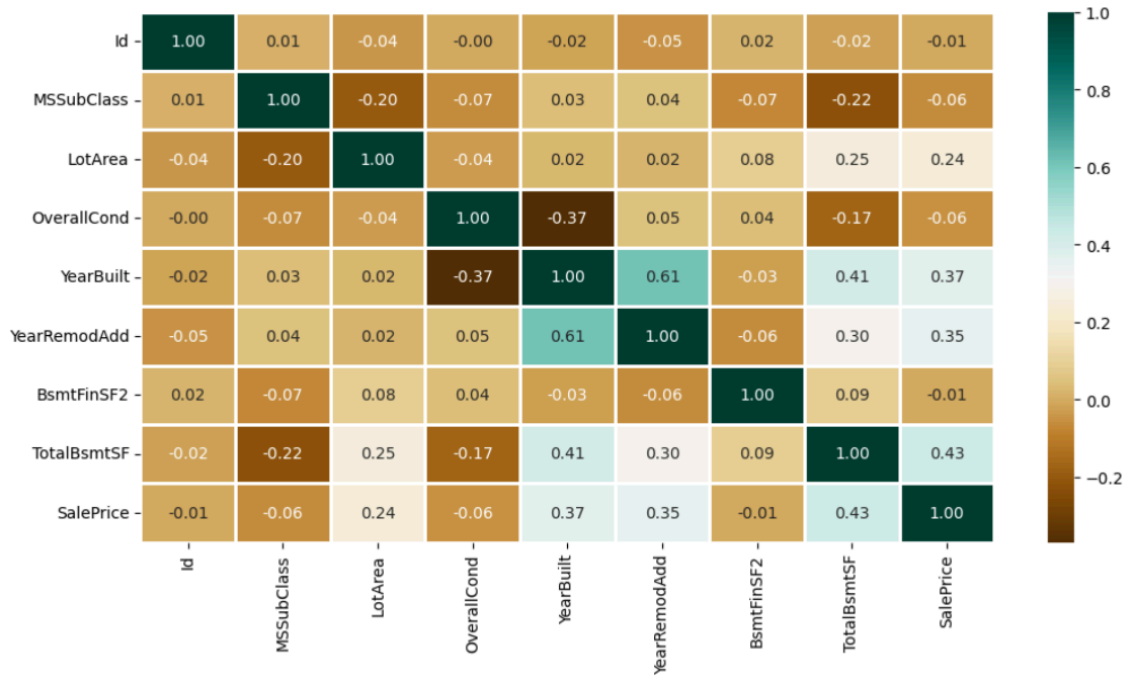
4.Code and Screenshots

1. Exploratory Data Analysis

EDA refers to the deep analysis of data so as to discover different patterns and spot anomalies. Before making inferences from data it is essential to examine all your variables.

```
plt.figure(figsize=(12, 6))
sns.heatmap(dataset.corr(),
            cmap = 'BrBG',
            fmt = '.2f',
            linewidths = 2,
            annot = True)
```

```
<Axes: >
```


Id
MSSubClass
LotArea
OverallCond
YearBuilt
YearRemodAdd
BsmtFinSF2
TotalBsmtSF
SalePrice
1.00 0.01 -0.04 -0.00 -0.02 -0.05 0.02 -0.02 -0.01
0.01 1.00 -0.20 -0.07 0.03 0.04 -0.07 -0.22 -0.06
-0.04 -0.20 1.00 -0.04 0.02 0.02 0.08 0.25 0.24
-0.00 -0.07 -0.04 1.00 -0.37 0.05 0.04 -0.17 -0.06
-0.02 0.03 0.02 -0.37 1.00 0.61 -0.03 0.41 0.37
-0.05 0.04 0.02 0.05 0.61 1.00 -0.06 0.30 0.35
0.02 -0.07 0.08 0.04 -0.03 -0.06 1.00 0.09 -0.01
-0.02 -0.22 0.25 -0.17 0.41 0.30 0.09 1.00 0.43
-0.01 -0.06 0.24 -0.06 0.37 0.35 -0.01 0.43 1.00
1.0
0.8
0.6
0.4
0.2
0.0
−0.2

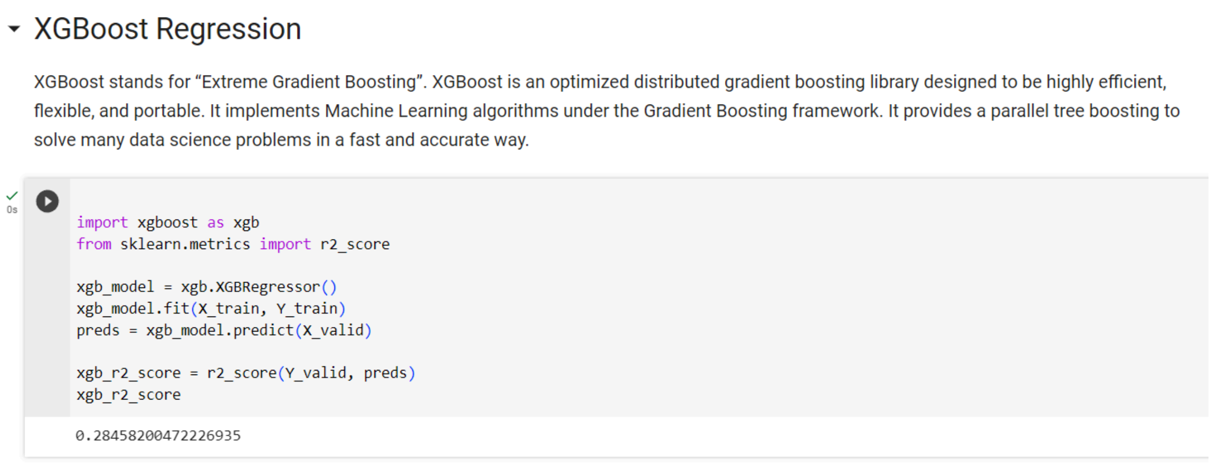
2.XGBoost Regression

XGBoost stands for “Extreme Gradient Boosting”. XGBoost is an optimized distributed gradient boosting library designed to be highly efficient, flexible, and portable. It implements Machine Learning algorithms under the Gradient Boosting framework. It provides a parallel tree boosting to solve many data science problems in a fast and accurate way.

```
import xgboost as xgb
from sklearn.metrics import r2_score

xgb_model = xgb.XGBRegressor()
xgb_model.fit(X_train, Y_train)
preds = xgb_model.predict(X_valid)

xgb_r2_score = r2_score(Y_valid, preds)
xgb_r2_score
```



XGBoost Regression

XGBoost stands for “Extreme Gradient Boosting”. XGBoost is an optimized distributed gradient boosting library designed to be highly efficient, flexible, and portable. It implements Machine Learning algorithms under the Gradient Boosting framework. It provides a parallel tree boosting to solve many data science problems in a fast and accurate way.

```
import xgboost as xgb
from sklearn.metrics import r2_score

xgb_model = xgb.XGBRegressor()
xgb_model.fit(X_train, Y_train)
preds = xgb_model.predict(X_valid)

xgb_r2_score = r2_score(Y_valid, preds)
xgb_r2_score
```

```
0.28458200472226935
```