PREDICTING HOUSE PRICE USING MACHINE LEARNING

BATCH MEMBER

DILLIBABU V (511321106009)
DILLIBABU V (511321106008)
DINESH K (511321106011)
SANTHOSH KUMAR K (511321106021)

PROJECT TITLE: HOUSE PRICE PREDICTION

PHASE 4

TOPIC: Continue building the house price prediction model by feature engineering, model training, and evaluation.

House Price Prediction

Introduction:

The process of building a house price prediction model is a critical endeavor in the realm of real estate, finance, and property valuation. Accurately estimating the price of a house is essential for buyers, sellers, and investors to make informed decisions. In this comprehensive guide, we will continue to delve deeper into the construction of a robust house price prediction model by focusing on three fundamental components: feature selection, model training, and evaluation.

Model training is the process of feeding the selected features to a machine learning algorithm and allowing it to learn the relationship between the features and the target variable (i.e., house price). Once the model is trained, it can be used to predict the house prices of new houses, given their features.

Model evaluation is the process of assessing the performance of a trained machine learning model on a held-out test set. This is important to ensure that the model is generalizing well and that it is not overfitting the training data.

PROCEDURE:

Feature selection:

1.Identify the target variable. This is the variable that you want to predict, such as house price.

2.Explore the data. This will help you to understand the relationships between the different features and the target variable. You can use data visualization and correlation analysis to identify features that are highly correlated with the target variable.

3.Remove redundant features. If two features are highly correlated with each other, then you can remove one of the features, as they are likely to contain redundant information.

4.Remove irrelevant features. If a feature is not correlated with the target variable, then you can remove it, as it is unlikely to be useful for prediction.

Feature Selection:

We are selecting numerical features which have more than 0.50 or less than -0.50 correlation rate based on Pearson Correlation Method—which is the default value of parameter "method" in corr() function. As for selecting categorical features, I selected the categorical values which I believe have significant effect on the target variable such as Heating and MSZoning.

In [1]:

```
important_num_cols =
list(df.corr()["SalePrice"][(df.corr()["SalePrice"]>0.5
0) | (df.corr()["SalePrice"]<-0.50)].index)
cat_cols = ["MSZoning",
"Utilities","BldgType","Heating","KitchenQual","
SaleCondition","LandSlope"]
important_cols = important_num_cols + cat_cols
df = df[important_cols]
```

Checking for the missing values

In [2]:

```
print("Missing Values by Column")
print("-"*30)
print(df.isna().sum())
print("-"*30)
print("TOTAL MISSING VALUES:",df.isna().sum().sum())
```

```
Missing Values by Column
------------------------------ OverallQual 0
YearBuilt 0
YearRemodAdd 0
TotalBsmtSF 0
1stFlrSF 0
GrLivArea 0
FullBath 0
```

TotRmsAbvGrd 0
GarageCars 0
GarageArea 0
SalePrice 0
MSZoning 0
Utilities 0
BldgType 0
Heating 0
KitchenQual 0
SaleCondition 0
LandSlope 0
dtype: int64

TOTAL MISSING VALUES: 0

Model training:

Model training is the process of teaching a machine learning model
to predict house prices. It involves feeding the model historical data on house prices and features, such as square footage, number of bedrooms, and location. The model then learns the relationships between these features and house prices.

1. Prepare the data. This involves cleaning the data, removing any errors or inconsistencies, and transforming the data into a format that is compatible with the machine learning algorithm that you will be using.

2. Split the data into training and test sets. The training set will be used to train the model, and the test set will be used to evaluate the performance of the model on unseen data

3. Choose a machine learning algorithm. There are a number of different machine learning algorithms that can be used for house price prediction, such as linear regression, ridge regression, lasso regression, decision trees, and random forests.

4. Tune the hyper parameters of the algorithm. The
hyper parameters of a machine learning algorithm are parameters that control the learning process. It is important to tune the hyper parameters of the algorithm to optimize its performance.

5. Train the model on the training set. This involves feeding the training data to the model and allowing it to learn the relationships between the features and house prices.

6. Evaluate the model on the test set. This involves feeding the test data to the model and measuring how well it predicts the house prices.

If the model performs well on the test set, then you can be confident that it will generalize well to new data.

Dividing Dataset in to features and target variable:

In [12]:

```
X = dataset[['Avg. Area Income', 'Avg. Area House Age', 'Avg. Area Number of
Rooms', 'Avg. Area Number of Bedrooms', 'Area Population']]
Y = dataset['Price']
```

2. Split the data into training and test sets. The training set will be used to train the model, and the test set will be used to evaluate the performance of the model.

In [13]:

```
X_train, X_test, Y_train, Y_test = train_test_split(X, Y,
test_size=0.2, random_st
ate=101)
```

In [14]:

```
Y_train.head()
```

Out[14]:

```
3413 1.305210e+06
1610 1.400961e+06
3459 1.048640e+06
4293 1.231157e+06
1039 1.391233e+06
Name: Price, dtype: float64
```

In [15]:

```
Y_train.shape
```

Out[15]:

```
(4000,)
```

In [16]:
Y_test.head()
Out[16]:
1718 1.251689e+06
2511 8.730483e+05
345 1.696978e+06
2521 1.063964e+06
54 9.487883e+05
Name: Price, dtype: float64
In [17]:
Y_test.shape
Out[17]: (1000)

3. Train the model on the training set. This involves feeding the training data to the model and allowing it to learn the relationships between the features and the target variable.

4. Evaluate the model on the test set. This involves feeding the test data to the model and measuring how well it predicts the target variable.

Conclusion:

In the quest to build an accurate and reliable house price prediction model, we have embarked on a journey that encompasses critical phases, from feature selection to model training and evaluation. Each of these stages plays an indispensable role in crafting a model that can provide meaningful insights and estimates for one of the most
significant financial decisions individuals and businesses make—real estate transactions.

Model training is where the model's predictive power is forged. We have explored a variety of regression techniques, fine-tuning their parameters to learn from historical data patterns. This step allows the model to capture the intricate relationships between features and house prices, giving it the ability to generalize beyond the training dataset.

Finally, model evaluation is the litmus test for our predictive prowess. Using metrics like Mean Squared Error, Root Mean Squared Error, Mean Absolute Error, and R-squared, we've quantified the model's performance. This phase provides us with the confidence to trust the model's predictions and assess its ability to adapt to unseen data.