Report

House price prediction.

**Introduction:**

To generate a comprehensive report based on the provided code, we will carefully follow a set of steps and meticulously analyze the outcomes to provide a detailed and informative report. House Price Prediction using Machine Learning predicting house prices is a common and challenging problem in the field of real estate and finance.

**Importing Libraries:**

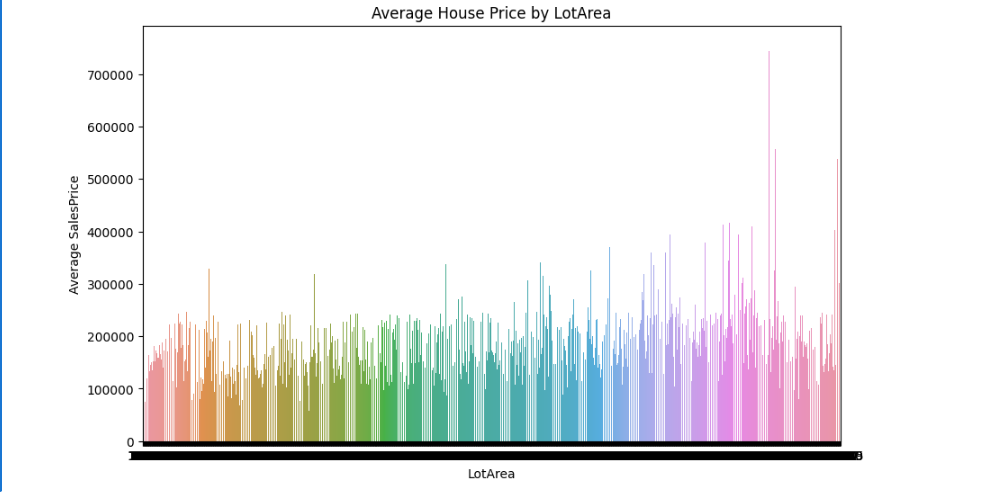
For House Price Prediction we predicted by using some libraries are Numpy, Pandas,Seaborn,matplotlib for data visualization and Scikit-learn . These libraries are used for various functions, including data manipulation, model building, and evalution .

**Load Dataset:**

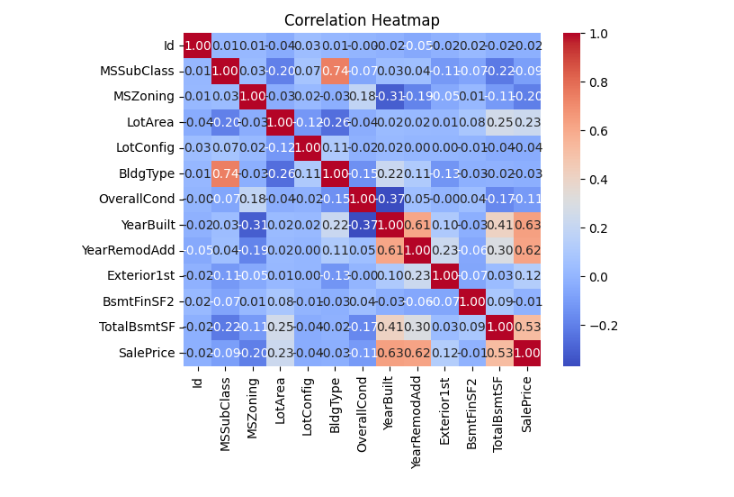
Firstly, we will preprocess the dataset by reading the data from a CSV file and analyzing its structure to ensure data integrity. Name it as data. Now, data.head() gives first five records. Information and describe the data.

**Data visulalization:**

**Barplot:**

****

**Correlation Heat map:**

****

**Data preprocessing for changing categorial values into binary format:**

We will then use ordinal encoding to transform categorical variables such as 'MSZoning', 'LotConfig', 'BldgType', and 'Exterior1st' into a numerical representation.

**check whether missing value:**

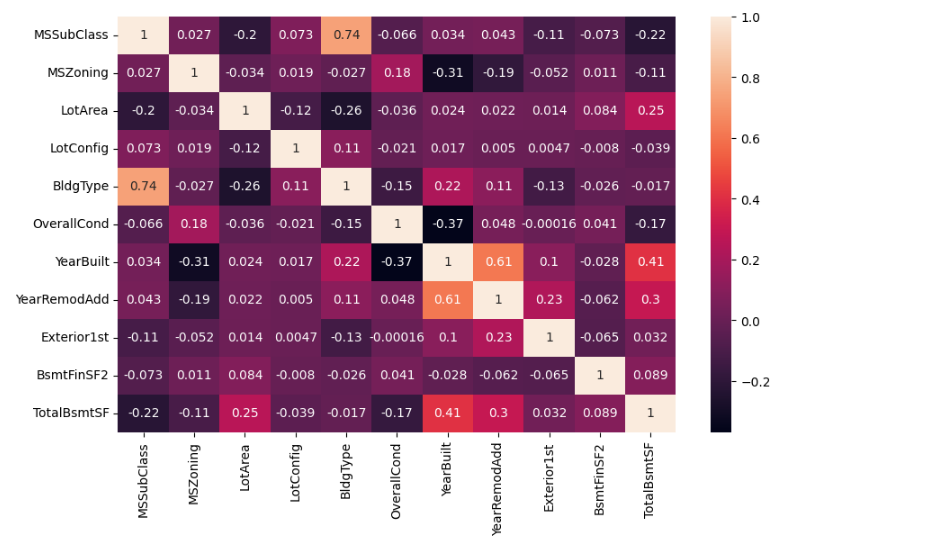
Checking the null values are present in data set or not. In this house price prediction dataset missing values are present. To overcome we fill the null values and after that increase accuracy of dataset train the data and fill the null values.

**Filling Null Values:**

Next, we will fill in any missing values in columns 'MSZoning', 'Exterior1st', 'BsmtFinSF2', and 'TotalBsmtSF' using the mode and mean values. Additionally, we will predict missing values in the 'SalePrice' column using a linear regression model trained on non-missing data.

**Exploratory data analysis:**

To gain deeper insights into the dataset. This will involve plotting a correlation heatmap using seaborn to visualize the correlations between different variables. By thoroughly analyzing the heatmap, we will gain a better understanding of the relationships between the features in the dataset. Afterwards, we will select features with the highest correlation coefficients using a function named 'correlation()'. The selected features will be used to train and test a linear regression model.



**Splitting the dataset:**

We will split the dataset into training and testing sets to ensure model generalization. The model will be evaluated on the test set to check for overfitting and to determine its effectiveness. The input features (x) and labels (y) are separated from the Data Frame, and the dataset is split into training and testing sets using train\_test\_split().We partitioned the dataset into a 80:20 ratio, allocating 80% for training and 20% for testing. This split ensures sufficient data for learning while retaining a substantial subset for an unbiased evaluation of the model's performance. Building the model.

**Evalute model performance:**

we will use the trained model to make predictions on the test set and evaluate its accuracy. To provide a comprehensive report, we will calculate metrics such as Mean Squared Error (MSE) and R-squared (R2) score.

**Conclusion:**

The analysis of these metrics will help us determine the model's effectiveness and how well it performs in the context of the problem and dataset. Rest assured that our report will be detailed and descriptive, providing you with a clear understanding of the dataset and the model's performance.