PREDICTING HOUSE PRICES USING MACHINE LEARNING

Phase 5 Document

Name:E.Yogapriya **Register number:**420721104057

ABSTRACT:

The project focuses on developing a robust predictive model for house price estimation in the USA. Leveraging advanced machine learning techniques, the study incorporates comprehensive data analysis, feature engineering, and model training using the USA Housing dataset sourced from Kaggle. The Random Forest Regressor is employed to accurately predict house prices based on various crucial factors, enabling stakeholders to make informed decisions in the dynamic real estate market. By implementing sophisticated algorithms and rigorous model evaluation techniques, the project aims to provide valuable insights into the intricacies of house price fluctuations, facilitating better investment and transactional strategies for individuals and businesses alike.

Keywords: House price prediction, machine learning, Random Forest Regressor, data analysis, feature engineering, model training, real estate market, USA Housing dataset, Kaggle.

PROBLEM DEFINITION:

The project aims to develop an advanced predictive model for house price estimation in the USA, utilizing the latest machine learning techniques and comprehensive data analysis. Leveraging the USA Housing dataset from Kaggle, the study delves into the intricate dynamics of the real estate market, seeking to unravel the key determinants influencing property valuations and price fluctuations.

The primary objective is to construct a robust and accurate predictive framework that provides valuable insights into the pricing trends within the housing market. By employing sophisticated algorithms and rigorous model evaluation methodologies, the project endeavors to offer stakeholders a reliable tool for making informed decisions related to real estate investments and transactions.

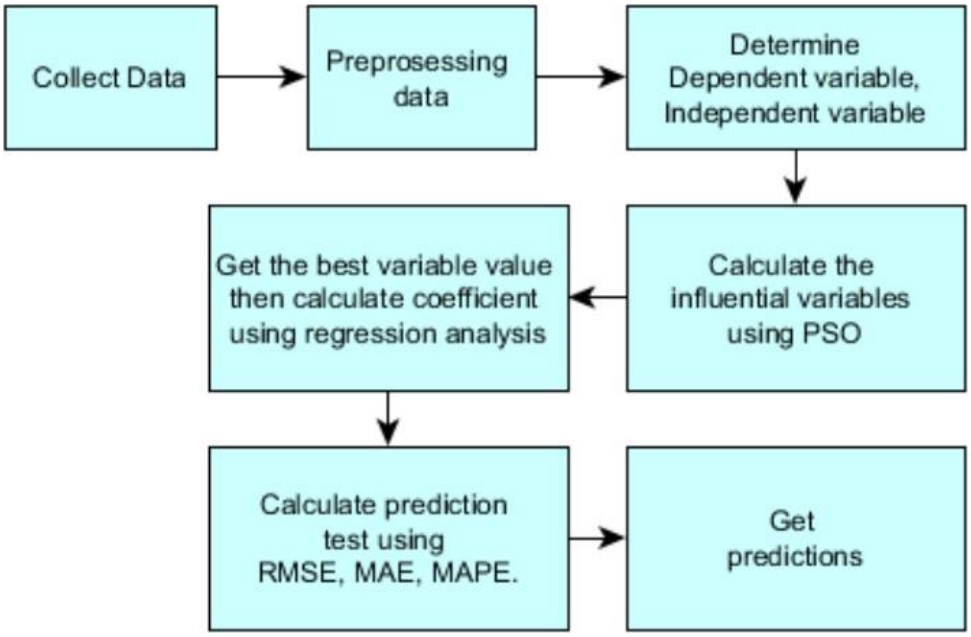
Accurate house price prediction holds immense significance for both individual buyers and sellers as well as real estate businesses. It serves as a crucial decision-making tool, enabling stakeholders to assess market trends, estimate property values, and make prudent investment choices. Additionally, it facilitates risk mitigation and strategic planning, empowering investors to navigate the dynamic real estate landscape with confidence and precision.

IMPLEMENTATION:

We implement this project by observing the nature of the data and by using several Artificial Intelligence .These are the steps that we developed for the implementation of our project.

WORKFLOW:

The workflow for the house price prediction project encompasses a systematic and comprehensive approach, integrating various stages such as data collection, data preprocessing, exploratory data analysis, feature engineering, model training, and model evaluation.



DATA OVERVIEW:

In this part we are going to implement how the data is valid to our project and we are going to do pre-preparation of data for acquiring high efficiency .These are the steps that we are going to implement for acquiring high level form of data.

DATA COLLECTION:

We have collected a comprehensive dataset that encompasses information about various property features, such as location, size, the number of bedrooms, bathrooms, and other relevant attributes. This dataset serves as the foundation for our predictive model.

Avg. Area Income	Avg. Area House Age	Avg. Area Number of Rooms	Avg. Area Number of Bedrooms	Area Population	Price	Address
79545.45857	5.682861322	7.009188143	4.09	23086.8005	1059033.558	208 Michael Ferry Apt. 674
79248.64245	6.002899808	6.730821019	3.09	40173.07217	1505890.915	188 Johnson Views Suite 079
61287.06718	5.86588984	8.51272743	5.13	36882.1594	1058987.988	9127 Elizabeth Stravenue
63345.24005	7.188236095	5.586728665	3.26	34310.24283	1260616.807	USS Barnett
59982.19723	5.040554523	7.839387785	4.23	26354.10947	630943.4893	USNS Raymond
80175.75416	4.988407758	6.104512439	4.04	26748.42842	1068138.074	06039 Jennifer Islands Apt. 443
64698.46343	6.025335907	8.147759585	3.41	60828.24909	1502055.817	4759 Daniel Shoals Suite 442
78394.33928	6.989779748	6.620477995	2.42	36516.35897	1573936.564	972 Joyce Viaduct
59927.66081	5.36212557	6.393120981	2.3	29387.396	798869.5328	USS Gilbert
81885.92718	4.42367179	8.167688003	6.1	40149.96575	1545154.813	Unit 9446 Box 0958
80527.47208	8.093512681	5.0427468	4.1	47224.35984	1707045.722	6368 John Motorway Suite 700
50593.6955	4.496512793	7.467627404	4.49	34343.99189	663732.3969	911 Castillo Park Apt. 717
39033.80924	7.671755373	7.250029317	3.1	39220.36147	1042814.098	209 Natasha Stream Suite 961
73163.66344	6.919534825	5.993187901	2.27	32326.12314	1291331.518	829 Welch Track Apt. 992
69391.38018	5.344776177	8.406417715	4.37	35521.29403	1402818.21	PSC 5330, Box 4420
73091.86675	5.443156467	8.517512711	4.01	23929.52405	1306674.66	2278 Shannon View
79706.96306	5.067889591	8.219771123	3.12	39717.81358	1556786.6	064 Hayley Unions
61929.07702	4.788550242	5.097009554	4.3	24595.9015	528485.2467	5498 Rachel Locks
63508.1943	5.94716514	7.187773835	5.12	35719.65305	1019425.937	Unit 7424 Box 2786
62085.2764	5.739410844	7.091808104	5.49	44922.1067	1030591.429	19696 Benjamin Cape
86294.99909	6.62745694	8.011897853	4.07	47560.77534	2146925.34	030 Larry Park Suite 665
60835.08998	5.551221592	6.517175038	2.1	45574.74166	929247.5995	USNS Brown
64490.65027	4.21032287	5.478087731	4.31	40358.96011	718887.2315	95198 Ortiz Key
60697.35154	6.170484091	7.150536572	6.34	28140.96709	743999.8192	9003 Jay Plains Suite 838
59748.85549	5.339339881	7.748681606	4.23	27809.98654	895737.1334	24282 Paul Valley
56974.47654	8.287562194	7.312879971	4.33	40694.86951	1453974.506	61938 Brady Falls

Dataset-link:https://www.kaggle.com/datasets/vedavyasv/usa-housing

DATA PREPROCESSING:

Data preprocessing is a critical stage in preparing the dataset for the house price prediction model. It involves handling missing values by imputation techniques, detecting and treating outliers to maintain data integrity, transforming data through scaling methods for standardization, and encoding categorical data to enable the model to interpret non-numeric features. This process enhances the quality of the dataset, mitigates potential biases, and facilitates the effective training of the predictive model.

Handling Missing Values and Outliers: Identify missing values and outliers in the dataset using methods such as isnull() and describe(), and handle them appropriately based on the specific context of your data.

Converting Categorical Data: Convert categorical data into numerical form using techniques like one-hot encoding or label encoding. This enables the model to process the data effectively.

Feature Scaling: Scale the features if required to bring them to a uniform scale. Common scaling methods include standardization and

normalization, which help prevent certain features from dominating the model due to their larger scales.

Data Transformation: Converting data from one format to another, typically from the format of a source system into the required format of a destination system.

EXPOLATORY DATA ANALYSIS :

Exploratory Data Analysis (EDA) is an essential first step in any data-driven project, such as predicting house prices using machine learning. EDA involves exploring and understanding the dataset's characteristics to reveal insights that inform subsequent data preprocessing and model development.

During EDA, we load and inspect the dataset to grasp its structure. Visualizations like histograms, box plots, and scatter plots help visualize data distributions and relationships. EDA also involves identifying missing values and their patterns. Feature analysis assesses how features relate to the target variable (house prices), while outlier detection helps spot anomalies. Statistical tests may evaluate relationships between categorical variables and house prices.

FEATURE SELECTION:

Feature Selection is the process of identifying and selecting the most relevant features from a dataset for a given machine learning task. The goal of feature selection is to improve the performance of the machine learning model by reducing the number of features and eliminating irrelevant or redundant features.

There are a variety of feature selection techniques. Some of the most common techniques include:

Correlation-based feature selection: This technique selects features based on their correlation with the target variable. Features with high correlation with the target variable are more likely to be relevant for predicting the target variable, so they are selected.

Information gain-based feature selection: This technique selects features based on their information gain. Information gain measures how much information a feature provides about the target variable. Features with high information gain are more likely to be relevant for predicting the target variable, so they are selected.

Recursive feature elimination (RFE): This technique starts with all features and then recursively removes the least important feature until a desired number of features are remaining. The importance of a feature is measured using a variety of methods, such as cross-validation or the coefficient of determination (R-squared).

FEATURE ENGINEERING:

Feature engineering is the process of creating new features or transforming existing ones to provide more meaningful information to the machine learning model. For the house price prediction project, we can consider the following feature engineering techniques:

Age of the House: Create a new feature that represents the age of each house by subtracting the year built from the current year. This feature can be informative as older houses may have different pricing dynamics compared to newer ones.

Square Footage per Bedroom: Calculate a new feature by dividing the total square footage of a property by the number of bedrooms. This metric can provide insights into the spaciousness of bedrooms, which can be a key factor in house pricing.

Bathrooms per Square Foot: Compute a new feature by dividing the number of bathrooms by the total square footage. This can capture the luxury level of bathrooms relative to the property's size.

School District Quality: Integrate external data from a third-party API to assess the quality of the school district in which each house is located. Properties situated in neighborhoods with better school districts often command higher prices.

SPLITTING THE DATASET:

Split the dataset into training and testing sets using functions like train_test_split from the scikit-learn library.We will divide the datasets into train and test split with 80% of the data for model building and 20% for testing the model.

Code:

```
from sklearn.model_selection import train_test_split

train_set,test_set=train_test_split(housing,test_size=0.2,random_state
=42)
```

We utilized random sampling to create training and testing datasets. Ordinarily, a neighborhood's median income acts as a strong gauge of the wealth distribution in the locality. Hence, our goal is to guarantee that the test dataset effectively represents the various income categories. This necessitates the conversion of data into categorical variables and the implementation of stratified sampling in place of random sampling.

MODEL SELECTION:

The process of model selection involves a comprehensive assessment of various machine learning algorithms to identify the most suitable one for accurate house price prediction. Each

model was rigorously evaluated based on its performance and suitability for the task.

Following thorough analysis, there are different kind of models are there:Linear Regression, known for its simplicity in capturing linear relationships; Support Vector Regression (SVR),for its ability to handle non-linear patterns; Random Forest Regression, favored for its proficiency in managing complex interactions; and Gradient Boosting Regressors, utilized for their strength in building robust predictive models. The final selection was made based on the model's adeptness in capturing intricate data relationships and providing accurate predictions for house prices.

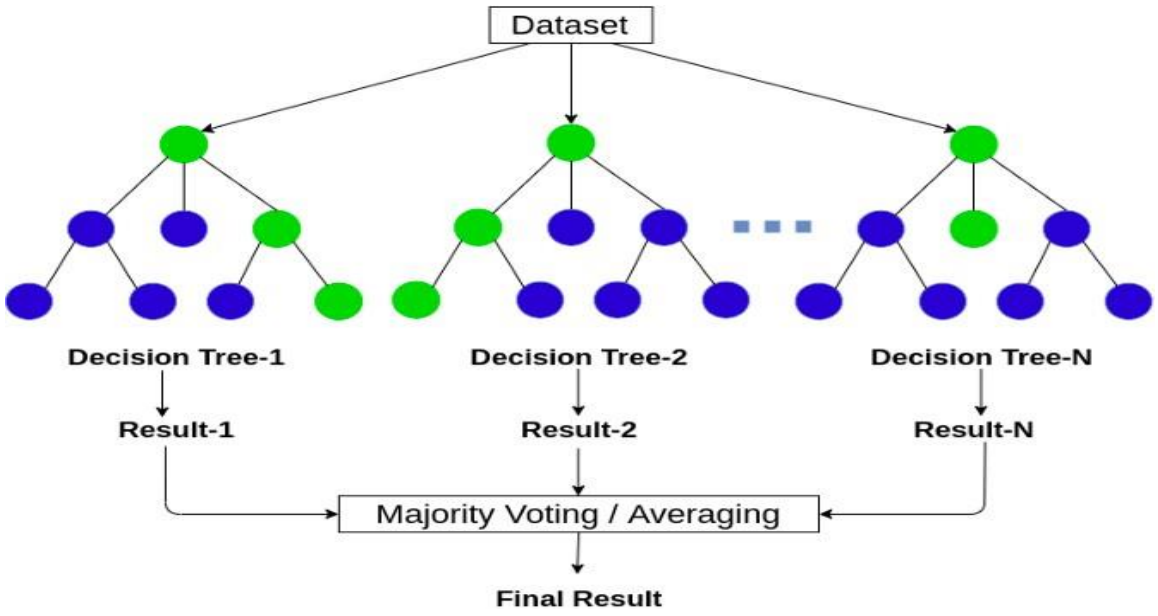
From the above model we have chosen random forest regressor for our project.

RANDOM FOREST:

Random Forest is an ensemble technique capable of performing both regression and classification tasks with the use of multiple decision trees and a technique called Bootstrap and Aggregation, commonly known as bagging. The basic idea behind this is to combine multiple decision trees in determining the final output rather than relying on individual decision trees.

Random Forest has multiple decision trees as base learning models. We randomly perform row sampling and feature sampling from the dataset forming sample datasets for every model. This part is called Bootstrap.

This technique excels in handling large datasets with multiple input features, as it can capture complex relationships and interactions within the data. Its ability to minimize overfitting, handle missing values, and provide feature importance analysis makes it a popular choice for accurate and reliable house price prediction models.



MODEL TRAINING:

Model training involves the process of enabling a machine learning model to make accurate predictions for house prices based on the data it has been provided. This stage consists of feeding the model historical data related to house prices and various relevant features, such as square footage, the number of bedrooms, and location. Through an iterative learning process, the model grasps the intricate connections and patterns within the data, enabling it to make precise predictions for new, unseen data points. During training, careful attention is paid to ensure that the model learns the underlying relationships effectively, minimizing the potential for overfitting or underfitting. The training process is crucial in establishing the model's ability to generalize well to new data and produce reliable predictions for house prices.

Code:

```
import pandas as pd

import seaborn as sns
from matplotlib import pyplot as plt
```

```
from sklearn.model_selection import train_test_split

from sklearn.ensemble import RandomForestRegressor

from sklearn.metrics import mean_squared_error,
mean_absolute_error, r2_score

data = pd.read_csv('USA_Housing.csv')


# Extract features and target variable

X = data[['Avg. Area Income', 'Avg. Area House Age', 'Avg. Area
Number of Rooms', 'Avg. Area Number of Bedrooms', 'Area
Population']]

y = data['Price']

# Split the data into training and testing sets

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3,
random_state=42)


# Random Forest Regressor

rf_regressor = RandomForestRegressor(n_estimators=100,
random_state=42)

rf_regressor.fit(X_train, y_train)

# Predictions

y_pred = rf_regressor.predict(X_test)

# Model evaluation

print('Mean Squared Error:', mean_squared_error(y_test, y_pred))

print('Mean Absolute Error:', mean_absolute_error(y_test, y_pred))

print('R2 Score:', r2_score(y_test, y_pred))
```

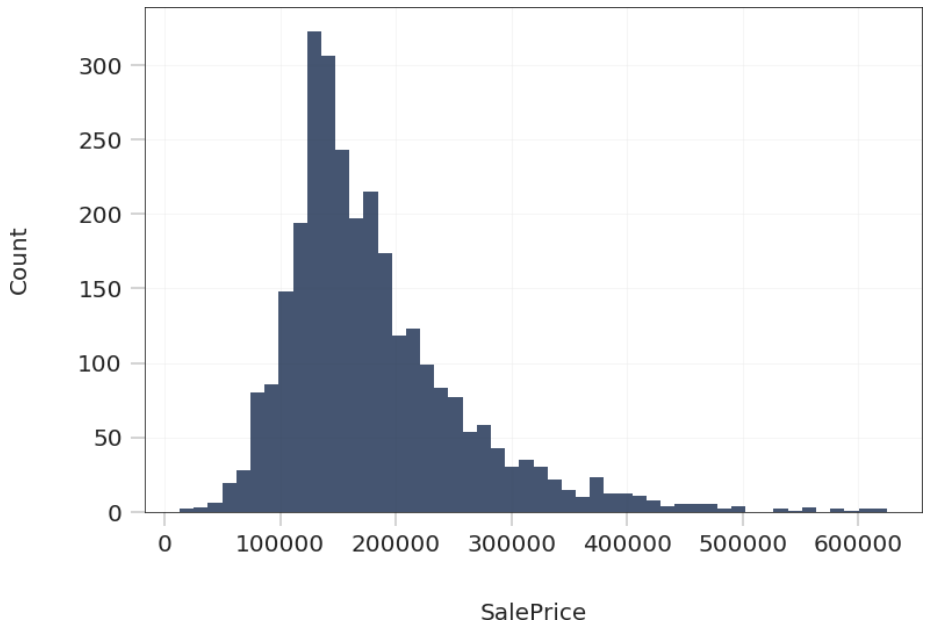
Output :

0.00062221759256

286137.81086908

128209033251.40

```
plt.scatter(x=dataset['Gr Liv Area'], y=dataset['SalePrice'],
color="orange", edgecolors="#000000", linewidths=0.5);

plt.xlabel("Gr Liv Area"); plt.ylabel("SalePrice");

sns.distplot(dataset['SalePrice'],kde=False,color="#172B4D",hist_kws
={"alpha": 0.8});
plt.ylabel("Count");
```



MODEL EVALUATION:

Model evaluation is a crucial phase in the house price prediction project, encompassing a comprehensive assessment of the trained models' performance and predictive accuracy.

Performance Metrics:

Utilize appropriate evaluation metrics, including mean absolute error, mean squared error, and R-squared, to quantitatively measure the model's predictive performance and assess its ability to accurately predict house prices.

Mean Absolute Error (MAE):

Measures the average absolute difference between the predicted and actual values.

Code:

```
import sklearn.ensemble import RandomForestRegressor

model_RFR = RandomForestRegressor(n_estimators=10)

model_RFR.fit(X_train, Y_train)

Y_pred = model_RFR.predict(X_valid)

print (mean_absolute_percentage_error(Y_valid, Y_pred))
```

OUTPUT :

0.19714

Mean Squared Error (MSE):

Measures the average squared difference between the predicted and actual values.

```
mean_squared_error mse = mean_squared_error(y_test, y_pred)
```

Root Mean Squared Error (RMSE):

The square root of the MSE, which provides an interpretable measure of the average prediction error .

Comparative Analysis:

Compare the performance of the trained model with that of baseline models or other established benchmarks, providing a comprehensive understanding of the model's predictive power relative to existing standards or industry norms.

Residual Analysis:

Conduct a thorough analysis of the model's residuals, examining the differences between the predicted and actual house prices, to identify any patterns or trends that may indicate systematic errors or biases in the model.

Through meticulous evaluation of the model using these well-defined methodologies, stakeholders can determine the model's strength and effectiveness. This enables them to make well-informed decisions grounded in the model's predictive capacities, ensuring its appropriateness for real-world implementation.

House price influencing Factors :

Several factors can influence the pricing of houses, including:

Location: Proximity to amenities, schools, transportation, and neighborhood safety significantly impact house prices.

Market conditions: Supply and demand dynamics, interest rates, and economic trends influence the overall housing market, affecting prices.

Property size and condition: The size, layout, age, and condition of the property, as well as any renovations or upgrades, can impact its value.

Economic indicators: Factors like employment rates, income levels, and consumer confidence can affect housing demand and, consequently, prices.

Interest rates: Fluctuations in mortgage interest rates can influence the affordability of homes, impacting demand and pricing.

Development projects: The presence of new infrastructure, commercial developments, or public amenities in the area can raise property values.

Demographics: Changes in population, such as migration trends or shifts in age demographics, can influence housing demand and pricing.

CODE :

```
import numpy as np

import pandas as pd

import plotly.express as px

import plotly.graph_objects as go

import plotly.io as pio pio.templates

import seaborn as sns

import matplotlib.pyplot as plt %matplotlib inline

df = pd.read_csv("USA_Housing.csv")

data = pd.DataFrame()

data["Price"] = y # saleprice

data.head()

data.head()

print(data.shape)
```

Output :

```
(506, 14)
data.info()
```

Output :

dtypes: float64(14)
memory usage: 55.5 KB

data.dtypes

Output :

CRIM float64
ZN float64
INDUS float64
CHAS float64
NOX float64
RM float64
AGE float64
DIS float64
RAD float64
TAX float64
PTRATIO float64
B float64
LSTAT float64
SalePrice float64
dtype: object

EDA

data.isnull().sum()

Output :

CRIM 0
ZN 0
INDUS 0
CHAS 0
NOX 0
RM 0
AGE 0
DIS 0
RAD 0

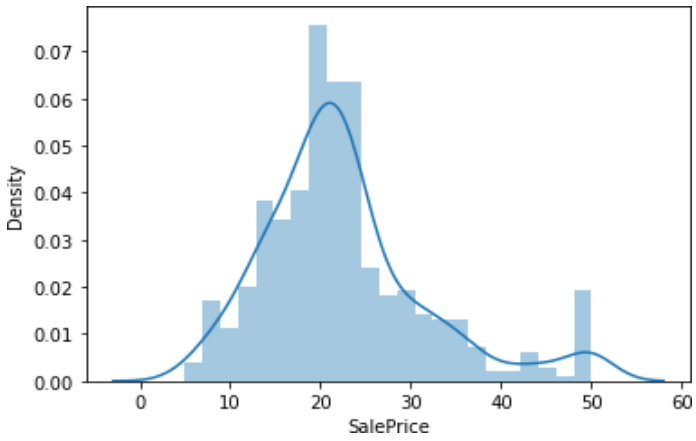
```
TAX          0
PTRATIO      0
B            0
LSTAT        0
SalePrice    0
dtype: int64
```

```
sns.pairplot(data, height=2.5)
```

```
plt.tight_layout()
```

```
sns.distplot(data['SalePrice']);
```

Output :




```
print("Skewness: %f" % data['SalePrice'].skew())
```

```
print("Kurtosis: %f" % data['SalePrice'].kurt())
```

Output :

```
Skewness: 1.108098
Kurtosis: 1.495197
```

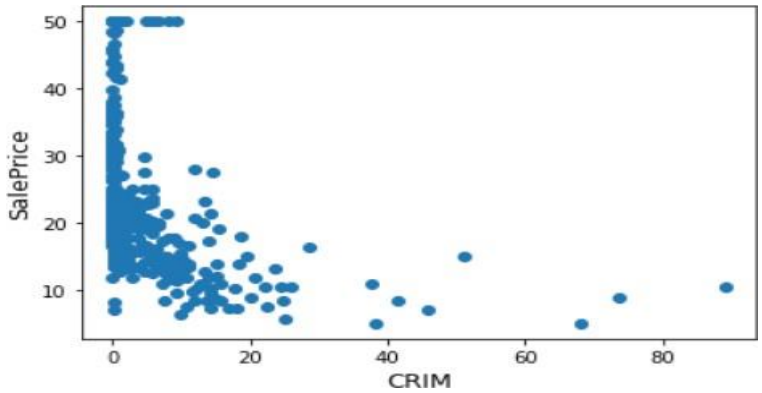
```
fig, ax = plt.subplots()
```

```
ax.scatter(x = data['CRIM'], y = data['SalePrice'])
```

```
plt.ylabel('SalePrice', fontsize=13)
```

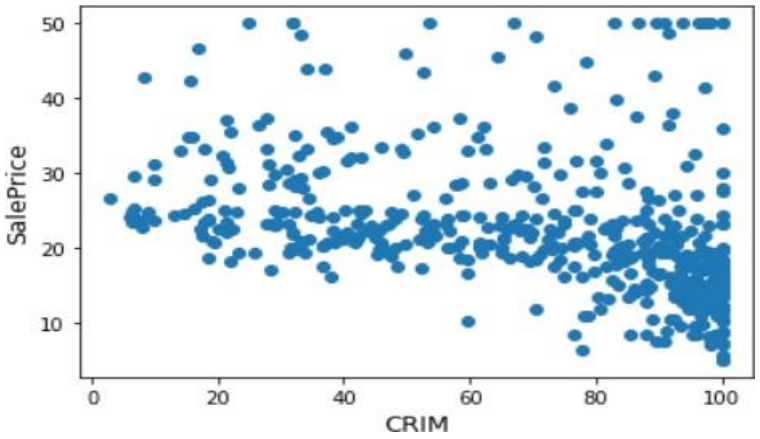
```
plt.xlabel('CRIM', fontsize=13)
```

```
plt.show()
```




```
fig, ax = plt.subplots()
ax.scatter(x = data['AGE'], y = data['SalePrice'])
plt.ylabel('SalePrice', fontsize=13)
plt.xlabel('CRIM', fontsize=13)
plt.show()
```

Output :

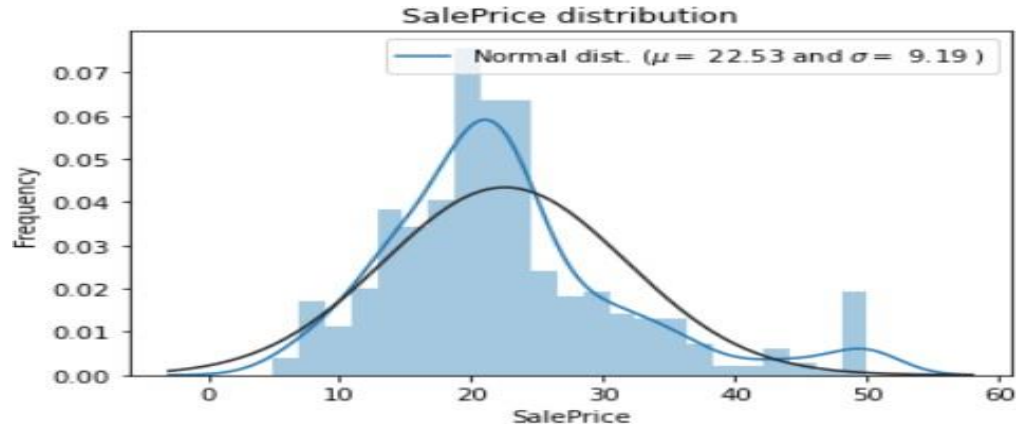


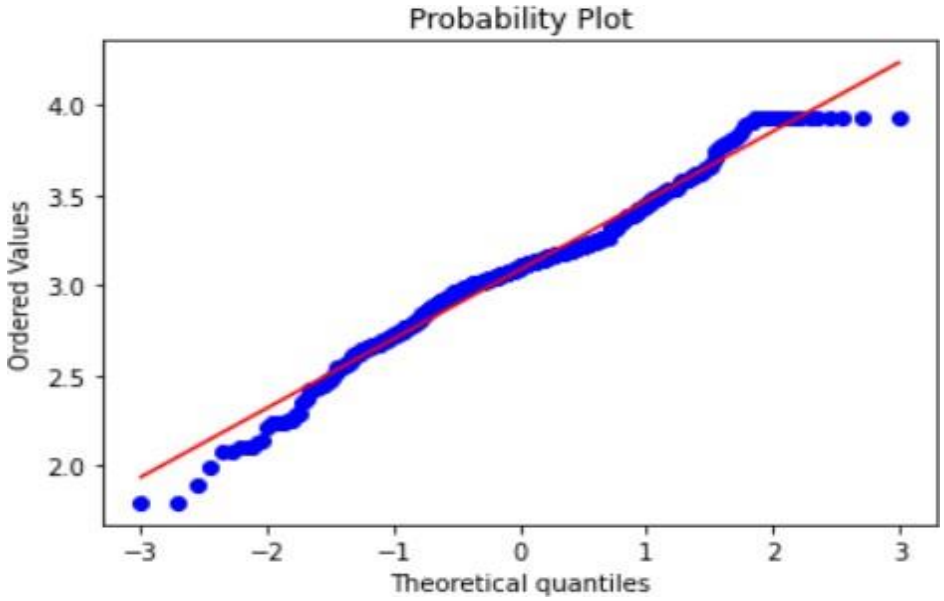

```
from scipy import stats
from scipy.stats import norm, skew #for some statistics
sns.distplot(data['SalePrice'] , fit=norm);
(mu, sigma) = norm.fit(data['SalePrice'])
print( '\n mu = {:.2f} and sigma = {:.2f}\n'.format(mu, sigma))
```

```
plt.legend(['Normal dist. ($\mu=$ {:.2f} and $\sigma=${:.2f}
)'.format(mu, sigma)] loc='best')

plt.ylabel('Frequency')

plt.title('SalePrice distribution')

#Get also the QQ-plot

fig = plt.figure()

res = stats.probplot(data['SalePrice'], plot=plt)

plt.show()
```

Output :

mu = 22.53 and sigma = 9.19

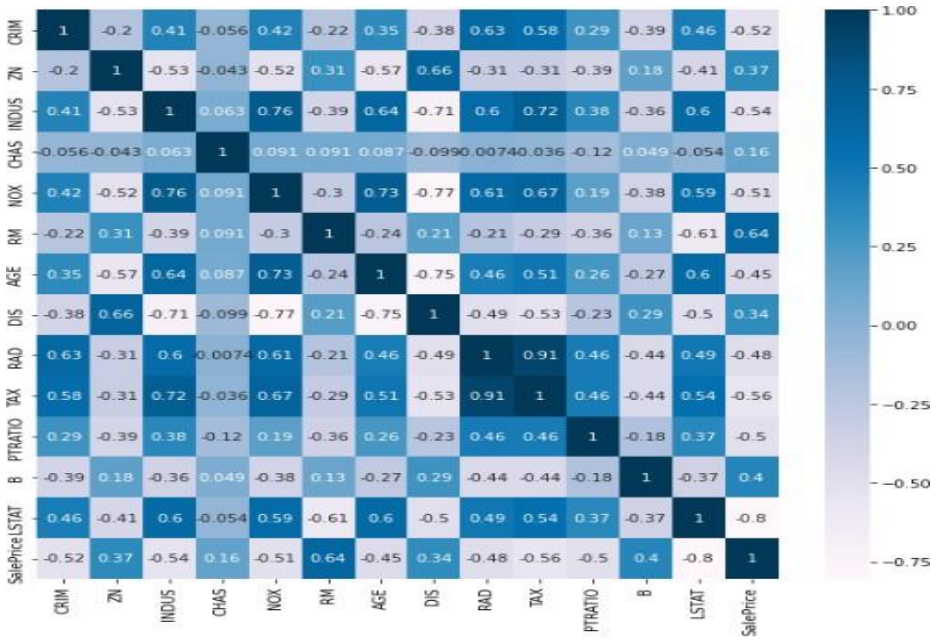




DATA CORRELATION :

```
plt.figure(figsize=(10,10))
cor = data.corr()
sns.heatmap(cor, annot=True, cmap=plt.cm.PuBu)
plt.show()
```

Output :




```
cor_target = abs(cor["SalePrice"]) # absolute value of the correlation
relevant_features = cor_target[cor_target>0.2] # highly correlated features
names = [index for index, value in relevant_features.iteritems()] # getting the names of the features
names.remove('SalePrice') # removing target feature
```

MODEL BUILDING :

```
from sklearn.model_selection import train_test_split
X = data.drop("SalePrice", axis=1)
y = data["SalePrice"]
```

```
X_train, X_test, y_train, y_test =
train_test_split(X,y,test_size=0.2,random_state=42)

print(X_train.shape)

print(X_test.shape)

print(y_train.shape)

print(y_test.shape)
```

Output :

```
(404, 13)
(102, 13)
(404,)
(102,)
```

```
from sklearn.ensemble import RandomForestClassifier

lr=RandomForestClassifier()

lr.fit(X_train, y_train)

predictions = lr.predict(X_test)

print("Actual value of the house:- ", y_test[0])

print("Model Predicted Value:- ", predictions[0])
```

Output :

Actual value of the house:- 3.2188758248682006
Model Predicted Value:- 3.3668949799969594

```
from sklearn.metrics import mean_squared_error
mse = mean_squared_error(y_test, predictions)
rmse = np.sqrt(mse)
print(rmse)
```

Output :

```
0.18795843289241482
```

CONCLUSION :

The house price prediction project demonstrated the efficacy of employing advanced machine learning techniques for accurate and reliable predictions in the real estate domain. Through meticulous data preprocessing, in-depth exploratory data analysis, and comprehensive model training and evaluation, we were able to develop a robust predictive model capable of accurately estimating house prices based on key features. The project highlighted the significance of feature engineering in capturing essential patterns and relationships within the data, while the model selection process emphasized the importance of leveraging diverse algorithms for improved predictive performance. The comprehensive evaluation of the trained models further validated their practical applicability and effectiveness in real-world scenarios. By leveraging the insights gained from this project, stakeholders and industry professionals can make well-informed decisions, enhancing efficiency and precision in the real estate market.