A REPORT ON

HOUSE PRICE PREDICTION USING MACHINE LEARNING

SUBMITTED BY:  
Name: Kishan Kumar Singh

Company Name: Coincent.ai

Date: Jun 20,2023

Bndlkfl mdvlkmvnkclkfnb

**Abstract**

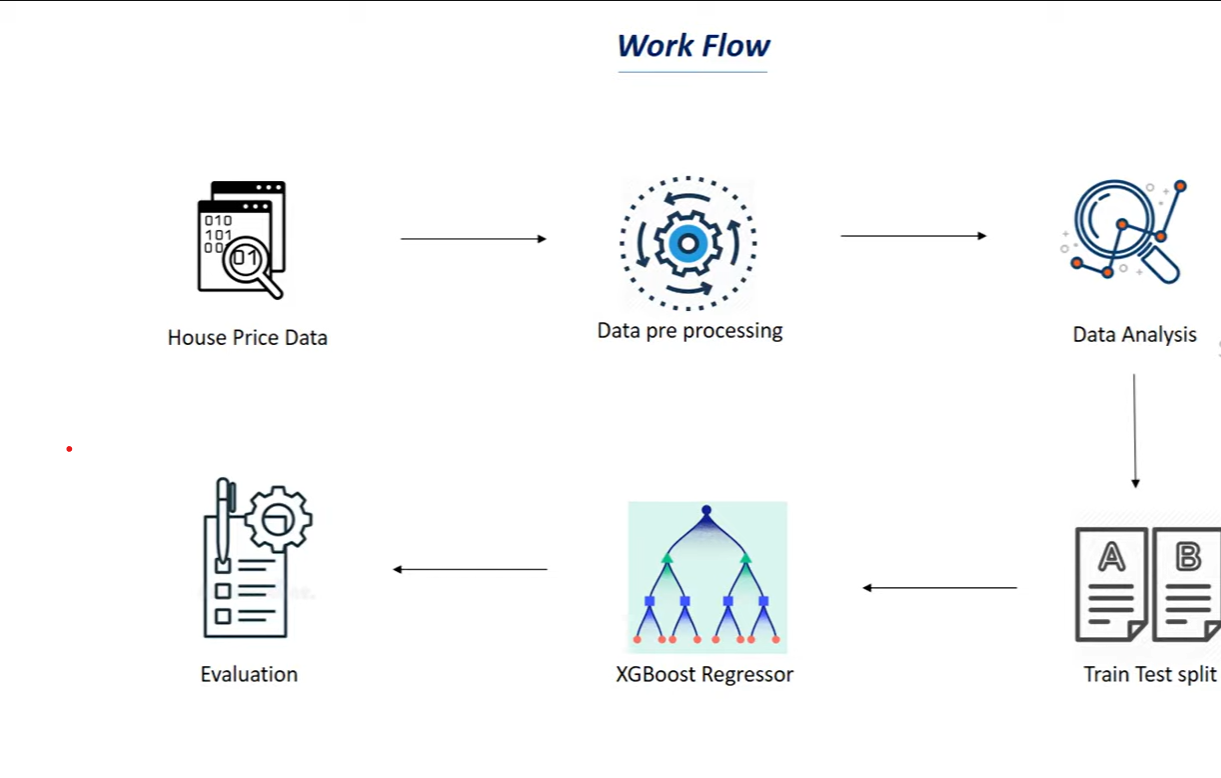
House price prediction is an important topic of real-estate. Since housing price is correlated to other factors such as area, population and apart from that it also related to size of the different sides of the house like basement, lot, bedroom, bathroom etc. Here, machine learning techniques are applied to analyze the price of different houses in different places for buyers and sellers. Here, we have used different machine learning algorithms and dependencies for predicting the price of different houses.

**INTRODUCTION:**

**Aim:**

* Create an effective price prediction model
* Validate the models prediction accuracy
* Identify the important home price attributes which feed the models prediction power.

Model Workflow:



**Dataset:**

Here, the dataset contains 7 columns and 699 rows having column name as predicted price, bedrooms, bathrooms, sqft\_living, sqft\_lot, sqft\_above, sqft\_basement. Here(sqft = square feet)

**Data Exploration:**

Data exploration is the first step in data analysis and typically involves summarizing the main characteristics of a data set, including its size accuracy initial patterns in the data and other attributes. It is commonly conducted by data analysts using visual analytics tool and more advanced statistical tool software, Python. Before it will conduct data is to be collected from multiple sources, warehouses and organizations such as Kaggle and many more. An initial exploration of the dataset can help with more familiarize with the data.

We divided the 8:2 for training and testing purpose respectively. **Data visualization:**

Data visualization is the graphical representation of information and data. By using visual elements like charts, maps and graphs of the data visualization tools provide an accessible way to see and understand trends and patterns in data. In the world of big data different visualization tools and technologies are essential to analyse for making data-driven decisions.

**Data Selection:**

Data selection is defined as the process of determining the appropriate data type and source, as well as suitable instruments to collect data. Data selection preceeds the actual practice of data collection. This definition distinguishes data selection from selectively excluding data that is not supportive of a research hypothesis and interactive data selection. The process of selecting suitable data for a research project can impact data integrity.

The primary objective of data selection is determination of appropriate datatype, source and instruments that allow analysts to answer the questions. This determination is often discipline specific and is primarily driven by the nature of the investigation, and accessibility to necessary data sources.

**Data transformation:**

The data transformation can be used to make highly skewed distributions less skewed. This can be valuable both for making patterns in the more interpretable and for helping to meet the assumptions of inferential statistics.

It is hard to discern a pattern in the upper panel whereas the strong relationship is shown clearly in the lower panel. The comparison of the means of data-transformed data is actually a comparison of geometric means.

**Python:**

Python is a user friendly andcontains a rich no. of libraries for different purpose. Python is also widely used for scientific and numerical computing.It has no. of libraries that can be used to train machine learning models such as:

1.**Numpy**: It is high level python library used to perform wide variety of mathematical operations on arrays.

2.**Pandas**:It is also a python library used to work with datsets and has functions for analyzing, exploring & manipulating data.

3. **Matplotlib**:It is a comprehensive library for creating static, animated, and interactive visualizations in python.

4. **seaborn**: It provides data visualizations that are typically more aesthetic and statistically sophisticated.

5. **scikit learn**: It is a open source data analysis library and gold standard for ML in the python ecosystem. It is also used for models like classification,regression, preprocessing, clustering, dimensionality reduction and etc.

6. **XG Boost**: It stands for Extreme Gradient Boosting which is used for predictive modelling. It implements gradient boosting decision tree algorithm. Boosting is an ensemble learning technique where new models are added to correct the errors made by existing models.

7.**scipy:** It is a collection of packages for mathematics, sciences and engineering.

Correlation is the statistical summary of the relationship between two sets of variables. It is used to think how our features corresponds to output when the two sets of variables are linearly related. The main 2 types of correlation is : 1.Positive Correlation

2. Negative Correlation

**CODE**:

import numpy as np

import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sb

import sklearn.datasets

from sklearn.model\_selection import train\_test\_split

from xgboost import XGBRegressor

from sklearn import metrics

data = pd.read\_csv("Data.csv")

data

Correlation of data:

Correlation = data.corr()

# COnstructing a heatmap to understand the correlation

plt.figure(figsize=(10,10))

sb.heatmap(Correlation,cbar=True,square=True,fmt='.1f',annot=True,annot\_kws={'size':8},cmap='Blues')

Here, we have cut the actual price from the dataset in(X) and added with(Y) for checking and comparing them with predicted prices after training and testing the models data.

X=data.drop(['price'],axis=1)

Y=data['price']

print(X)

print(Y)

For training and testing we have implemented models from sklearn

X\_train, X\_test, Y\_train, Y\_test = train\_test\_split(X, Y, test\_size=0.2, random\_state= 2)

Training the model with XGB regressor

model = XGBRegressor()

#training model with X\_train

model.fit(X\_train, Y\_train)

#accuracy for prediction on training data

training\_data\_pred = model.predict(X\_train)

print(training\_data\_pred)

#Error for training data

# R squared error

score\_1 = metrics.r2\_score(Y\_train, training\_data\_pred)

print("R squared error:", score\_1)

output: R squared error: 0.997074993251354

#prediction on testing data

test\_data\_pred = model.predict(X\_test)

#error value for test data

# R squared error

score\_2 = metrics.r2\_score(Y\_test, test\_data\_pred)

print("R squared error:", score\_2)

output: R squared error: 0.3689342910608846

Visualization of the result:

plt.scatter(Y\_train,training\_data\_pred,c=Y\_train,linewidth=0.001)

plt.xlabel("Actual prices")

plt.ylabel("Predicted prices")

plt.title("Actual and predicted prices")

plt.show()

**Conclusion:**

So, our aim is achieved as we have successfully ticked all out parameters as mentioned in out aim column(Introduction). It is seen that circle rate is the most effective attribute in predicting the house price and the regression is the most effective model for our dataset with R Squared Error of 0.997074993251354 for training data and 0.3689342910608846 for testing data.