PREDICTING HOUSE PRICES USING MACHINE LEARNING

Over long ago, there is manually decide the price of any property. But problem is that in manually there are 25% percent error is occurred and such affect is loss of money. But now there is big change by changing the old technology. Today's Machine Learning is trending technology. Data is the heart of Machine Learning. Nowadays the booming of AI and Machine Learning in market. All industry are move towards automation. But without data we can't train model. Basically in Machine Learning involves building these model from previous data and by using them to predict new data. The market demand for housing is increases daily because our population is rising rapidly.in rural area there is lack of jobs due to this public is migrating for financial purpose.so result is increasing demand of housing in cities. People who don't know the actual price of that particular house and they suffer loss of money. In this project, the house price prediction of the house is done using different Machine Learning algorithms like Leaner Regression, Decision Tree Regression, K- Means Regression and Random Forest Regression. 80% of data form kwon dataset is used for training purpose and remaining 20% of data used for testing purpose. This work applies various techniques such as features, labels, reduction techniques and transformation techniques such as attribute combinations, set missing attributes as well as looking for new correlations. This all indicates that house price prediction is an emerging research area and it requires the knowledge of machine learning.

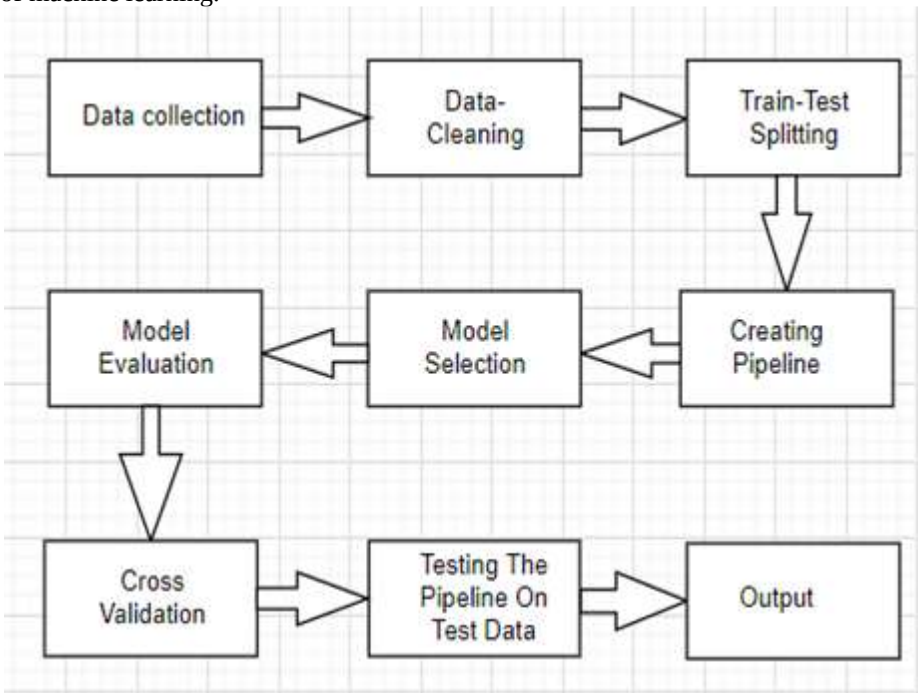

Fig 1. Research Flow Diagram

PROPOSED SYSTEM

In this proposed system, we focus on predicting house price using machine learning algorithms like Leaner Regression, Decision Tree, k-Means, and Random Forest. We proposed the system "House Price Prediction Using Machine Learning" we have predict the house price using multiple features. In this proposed system, we are able to train model…..the previous data taken and out of this 80% of data is used for training purpose and remaining 20% of data used for testing purpose. Hare, the raw data is stored in '.csv' file. We are majorly used two machine learning libraries to solve these problems. The first one was 'pandas' and another one is 'numpy'. The pandas used for to load '.csv' file into Jupiter notebook and also used to clean the data as well as manipulate the data. Another was sklearner, which was used for real analysis and it has containing various inbuilt functions which help to solve the problem.one more library was used which is nothing but numpy. For the purpose of train-test splitting numpy was used.

Linear regression for house price prediction

Linear regression is a mainly used technique for the prediction of house prices due to its simplicity and interpretability. It assumes a linear relationship between the independent variables (such as how many bedrooms, number of bathrooms, and square footage) and the dependent variable (house price). By fitting a linear regression model to historical data, we can estimate the coefficients that represent the relationship between the target variable and the features. This enables us to make predictions on new data by multiplying the feature values with their respective coefficients and summing them up. Linear regression provides insights into the impact of each feature on the house price, enabling us to understand the significance of different factors and make informed decisions in the real estate market.

House price prediction using machine learning

Machine learning involves training a computer to recognize patterns and make predictions based on data. In the case of house price prediction, we can use historical data on various features of a house, such as its location, size, and amenities, to train a machine-learning model. Once the model is trained, it can analyze new data on a given house and make a prediction of its market value.

WE HAVE USED KAGGLE USA_Housing.csv

- Import the required libraries and modules, including pandas for data manipulation, scikit-learn for machine learning algorithms, and LinearRegression for the linear regression model.
- Loading the required dataset with pd.read_csv and select the features we want to use for prediction (e.g., bedrooms, bathrooms, sqft_living, sqft_lot, floors, and zip code), as well as the target variable (price).

- ## Simple Linear Regression
- Simple linear regression uses a traditional slope-intercept form,

  where *a* and *b* are the coefficients that we try to "learn" and produce the most accurate predictions. *X* represents our input data and *Y* is our prediction.
- $$Y = bX + a$$

Multivariable Regression

A more complex, multi-variable linear equation might look like this,

where *w* represents the coefficients or weights, our model will try to learn.

$$Y(x_1, x_2, x_3) = w_1 x_1 + w_2 x_2 + w_3 x_3 + w_0$$

The variables x_1, x_2, x_3 represent the attributes or distinct pieces of information, we have about each observation.

Loss function

Given our Simple Linear Regression equation:

$$Y = bX + a$$

Data Cleaning

Data Cleaning is the way to improvise the data or remove incorrect, corrupted or irrelevant data.

- We can easily delete the column/row (if the feature or record is not much important).
- Filling the empty slots with mean/mode/0/NA/etc. (depending on the dataset requirement).

Mean Squared Error (MSE) Cost Function

The MSE is defined as:

$$MSE = J(W) = \frac{1}{m}\sum_{i=1}^{m}(y^{(i)} - h_w(x^{(i)}))^2$$

where

$$h_w(x) = g(w^T x)$$

The MSE measures how much the average model predictions vary from the correct values. The number is higher when the model is performing “bad” on our training data.

The first derivative of MSE is given by:

$$MSE' = J'(W) = \frac{2}{m}\sum_{i=1}^{m}(h_w(x^{(i)}) - y^{(i)})$$