**MACHINE LEARNING**

**ASSIGNMENT-1**

Data set for House Price Prediction

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **House Id** | **House Type (BHK)** | **Size**  **(Sq. ft)** | **Location** | **Year Built** | **Parking** | **Price**  **(Rs)** |
| 1 | 2 | 7420 | Erode | 2018 | 2 | 20,00,000 |
| 2 | 1 | 8810 | Salem | 2012 | 1 | 10,00,000 |
| 3 | 2 | 6090 | Chennai | 2005 | 1 | 300,00,000 |
| 4 | 3 | 9095 | Coimbatore | 2009 | 2 | 28,00,000 |
| 5 | 1 | 3000 | Madurai | 2020 | 1 | 15,00,000 |

1. **Feature**

Features are input values or variables that are used by models to make predictions. In this dataset features include House Id, House Type, Location, Year Built and parking

1. **Label**

The output value or variable that model is trying to predict. In this example Price is the label for this dataset.

1. **Prediction**

The estimated value that is predicted by our model based on the given input variables. The model predicts the output based on the train data given by us.

1. **Outlier**

A value that is significantly different from other values which means the range of that value is far different when compared to other values. In this example the price of house id 3 (300,00,000) varies huge different when compared to other prices.

1. **Test Data**

The data which is used to evaluate the model’s performance after training. It tests how well our model predicts the house prices.

1. **Training Data**

The data that is used to train our model. Training Data should be more in order to develop our model more precise and accurate.

1. **Model**

The mathematical algorithm that makes prediction based on the input. For House Price Prediction we can use Linear Regression model which predicts house prices based on features like size and location.

1. **Validation Data**

Validation Data is used to fine tune the performance of model. It is used to check how well our model is performing and to make decisions about changes to improve the model, like adjusting hyperparameters.

1. **Hyper Parameter**

Hyper Parameter are configuration variables that need to be set before training the model. They are not learned from data like parameters, but they control how the model should be trained. Example of hyper Parameter is Learning rate which determines how quickly a model update its parameter during training of our model.

1. **Epoch**

Epochs are number of times the model should process the training data. If the model processes the data only once then it might not learn enough from data which leads to poor performance. Increasing the number of Epochs makes the model to learn more from data.

1. **Loss Function**

It computes the difference between the predicted outputs of machine learning algorithm and actual values. It is also known as Error function.

1. **Learning Rate**

When training the model, the algorithm updates the model’s parameters to reduce the difference between predicted and actual values. Learning rate determine the size of those updates made to the model’s parameters. Finding the right learning rate is crucial for efficient and effective training.

1. **Overfitting**

When a model is trained too much from training data then it starts to memorize the values so it performs well on training data but it performs poor on new data which causes to overfitting. For example, if we include irrelevant features like color of the tiles the model becomes very complex and fail to make prediction for new houses.

1. **Underfitting**

When a model is too simple that has no complexity to determine the patterns in data it leads to poor performance where underfitting occurs. For example, If we train our model only based on the number of bedrooms and leaving other features then it will not give a correct House price prediction.

1. **Regularization**

It is a technique that is used to prevent overfitting by adding constraints to the model.

1. **Cross-Validation**

Cross Validation is like testing your model with some random datasets. For example, if there are 5 data we use 4 data for training and one for validation that is testing the model.

1. **Feature Engineering**

It is the process of modifying the features in order to improve model’s performance. For example, features like age of house can be added to our dataset to increase model accuracy.

1. **Dimensionality Reduction**

It is a technique to remove the number of irrelevant features or redundant information that confuses the model.

1. **Bias**

Bias occurs when a model is too simple and fails to capture the pattern in data well. It is similar to underfitting.

1. **Variance**

Variance involves how much model’s predictions change when it’s trained on different subsets of same data. Managing variance involves finding the correct level of complexity so that it performs well in new data.

**22CSR150**

**PRASANNA S**