**How does Machine Learning Model Works ?**

A machine learning model identifies price patterns for a given data that has been trending in the past, those patterns are used for new outcomes or predictions that are being looked into.

**First Machine Learning Model - Decision Tree**

It divides data into two parts, The prediction data for any given data is consideration of the historical average of those data in the same category. We use data to break data into two groups, then again determine the prediction of the data into each group. This particular step is known as **fitting or training the mode**l. The data used to fit the model is called the training data. Once a model has been fit, it can be applied to new data to predict prices of additional given data.

More factors of data can be captured using a tree that has more "splits". These are called "Deeper" trees.

**Basic Data Exploration**

* **Brief About Pandas**

Pandas is the primary tool for data scientists to use for exploring and manipulating data. The most important part of the Pandas Library is the DataFrames. A DataFrame holds the type of data you might think as Tables, Similar to sheet in Excel or a table in a SQL database

**Steps to create a machine learning model**

* **Selection Of Data**

These are many ways to select a subset of data, Right now we will focus on two approaches -

* Dot Notation, Which we use to select Prediction Target(y)
* Selecting with a column lost, which we use to select the Features(X)
* **Building Model**

The steps to build and using a model are:

* **Define**: What type of model will it be? A decision tree? Some other type of model? Some other parameters of the model type are specified too.
* **Fit**: Capture patterns from provided data. This is the heart of modelling.
* **Predict**: Just what it sounds like
* **Evaluate**: Determine how accurate the model's predictions are.

**Model Validation**

Measuring model quality is the key to iteratively improving ML models.

**What is Model Validation ?**

The relevant measure of model quality is predictive accuracy, in easy words, How accurate are the predictions made by a model.

**Loophole while measuring Predictive Accuracy**

* Making predictions with training data and comparing those predictions to the target values in the training data.

First, the model quality needs to be summarised in an understandable way. If predicted data is compared to actual data of 10,000 values there will be a mix of good and bad data predictions. Looking through a list of 10,000 predicted and actual values would be pointless. This needs to be summarised into a single metric which brings us to **MAE (Mean Absolute Error)**

| **Prediction Error :[ *Error = Actual Data - Predicted ]*** |
| --- |

With the MAE metric, we take the absolute value of each error. This converts each error to a positive number. Then an average of those absolute errors is taken, This measures our Model Quality.

**The Problem With “In-Sample” Scores**

Using a single “SAMPLE” of data for both building and evaluating the model. Since the model's pattern was derived from training data, the model will appear accurate in the training data.

Since model’s practical value comes from making prediction on new data, we measure performance on the data that was not used to build the model, the most straightforward way to do this is to exclude some data from the model-building process, and then use those to test the model’s accuracy on data it hasn’t seen before. This data is validation data.

**Underfitting and Overfitting**

The decision tree model has many options. The most important options determine the tree’s depth. It’s common for a tree to have a 10 split between the top level and a leaf, As the tree gets deeper, the dataset gets sliced up into leaves with fewer data.

If a tree only has 1 split, it divides the data into two groups. If each group is split again, we would get 4 grou[s of houses. Splitting those again will create 8 groups. If it keeps doubling the number of groups by adding more splits at each level, we will have 2^10 groups of houses by the time we get to the 10th level. That’s 1024 leaves.

When a data is divided into so many leaves, we also have less data into each leaf, this may make predictions that are quite close to those actual values, but they may make very unreliable predictions for new data because each prediction is based on only a few houses. This is a phenomenon called **Overfitting,** where a model matches the training data almost perfectly, but does poorly in validation and other new data.

On the flip side, if a tree is very shallow, it doesn’t divide up the houses into very distinct groups.

At an extreme, if a tree divides data into 2 or 4, each group still has a wide variety of houses. Resulting predictions may be far off for most data, even in training data. When a model fails to capture important patterns in the data, so it performs poorly even in training data, that is called **Underfitting.**

Since we care about accuracy on new data, which we estimate from our validation data, we want to find the sweet spot between underfitting and overfitting. Visually we want the low point of the (red) validation curve in the figure below.

