Pattern Recognition and Machine Learning

Indian Institute of Technology, Jodhpur

॥ त्वं ज्ञानमयो विज्ञानमयोऽसि ॥

LAB 2

Report

Harshil Kaneria

Department of Computer Science, IIT Jodhpur

Feb 13, 2023

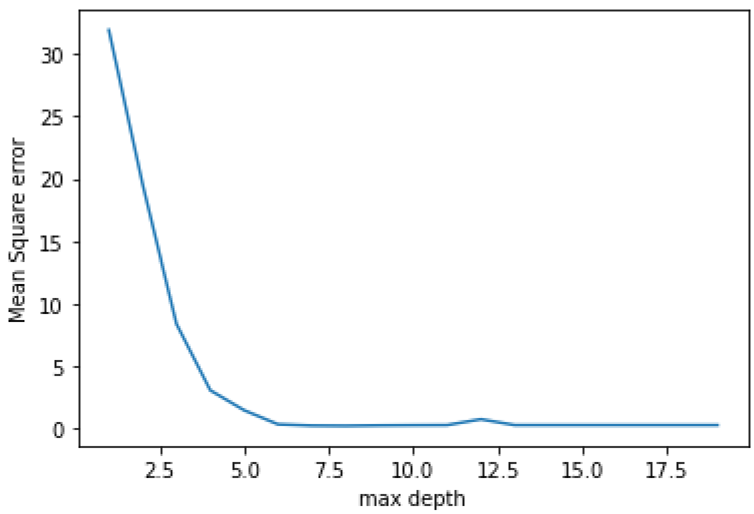
1)

1)

a) Pre-processing:

b) Splitting Into Training and Testing data: Split data into training set (70%) Validation set (10%) and testing set (20%).

2) In this question, I trained the regression decision tree on training dataset. I will vary max depth hyper parameter. In regression decision tree model, there is always a fear of overfitting if max depth of tree is not controlled. So, I varied max depth and controlled it.



The above graph shows the variation of mean squared error with max depth and we see that at max depth = 6 there is minimum mean squared error. So, we choose optimal hyper parameter as max depth = 6.

3) I performed different types of Cross validations using the hyper parameter taken out in previous question.

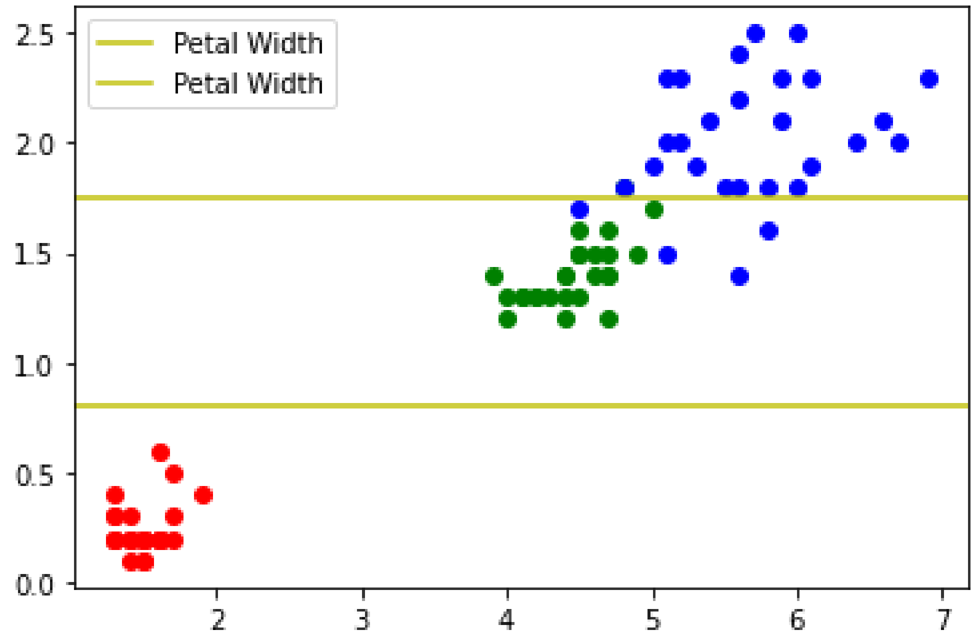
Following are Mean Squared Value of different CV types:

a) Hold out CV: 0.3521
b) 5-Fold CV: average - 0.311
c) Repeated 5-Fold CV: average – 0.309

2)

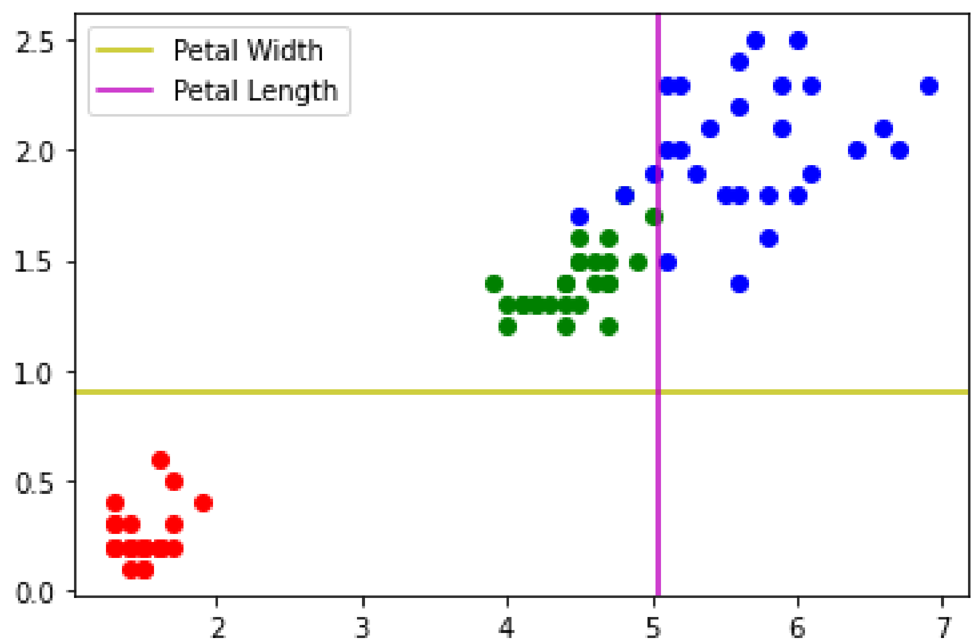
Classifier

1) Decision Boundary of Decision Tree Classifier (max depth =2):

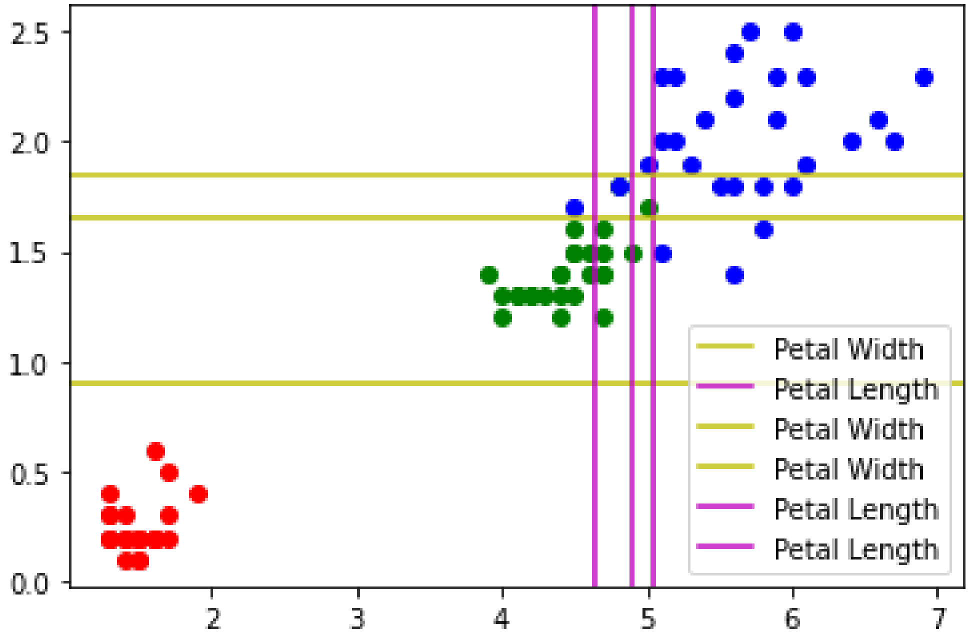


PS: Decision boundary may change a little because I have shuffled the dataset at beginning.

2) After removing the widest Iris-versicolor from Training dataset, the decision boundary plotted is as follows:



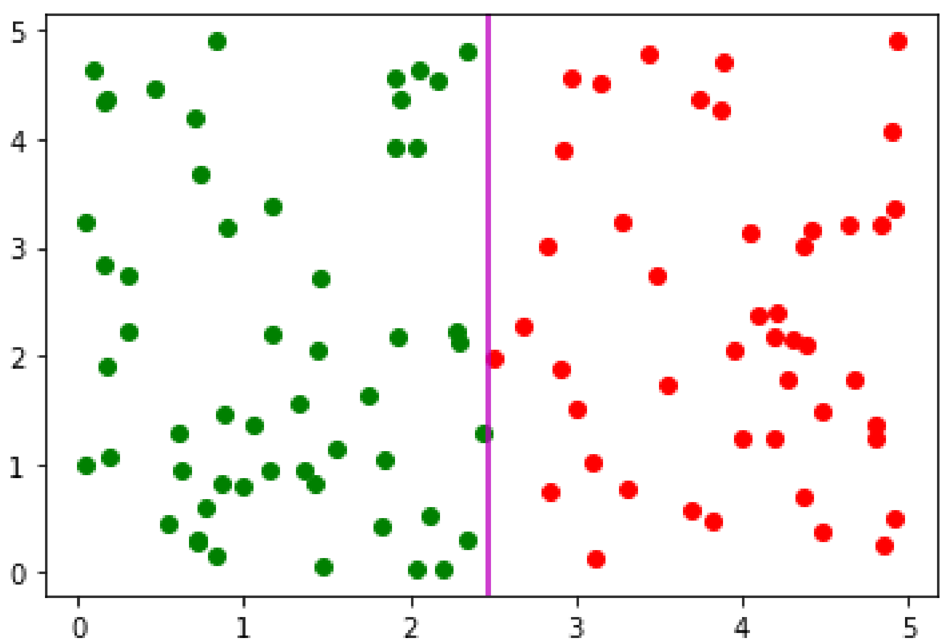
3) Decision Boundary for Decision Tree Classifier (max depth = None):



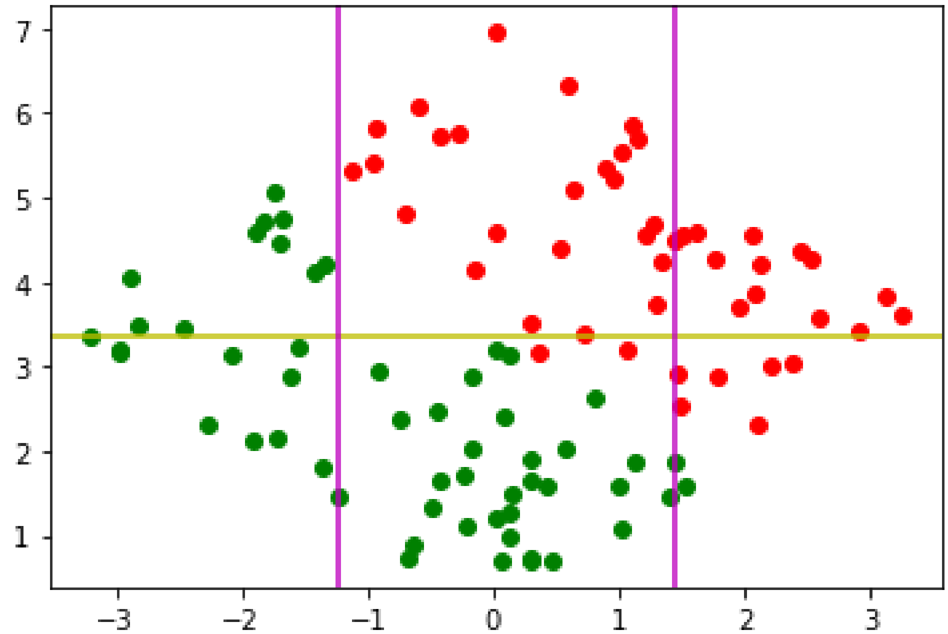
In part 1, the Decision Tree trained had max depth = 2. This hyper parameter limits the further splitting after level 2. In part 3, making max depth = None, we remove that restriction and this tree would classify or split until each leaf node has only one class present in it. Therefore, we see that the decision boundary of max depth = 2 decision tree is simpler and lesser than that of max depth = None decision tree.

4)

a) Decision Boundary for Decision Tree Classifier (max depth = 2):



b) Decision Boundary for rotated dataset (by 45 degree):



We can clearly see that the data points have been rotated about origin by 45 degrees. Also, there is no longer only 1 orthogonal boundary. There are multiple orthogonal lines that separate the points because it can only make orthogonal lines to make the boundary. Also we can see for part a the accuracy is 100% but its not for dataset when rotated by 45 degree.

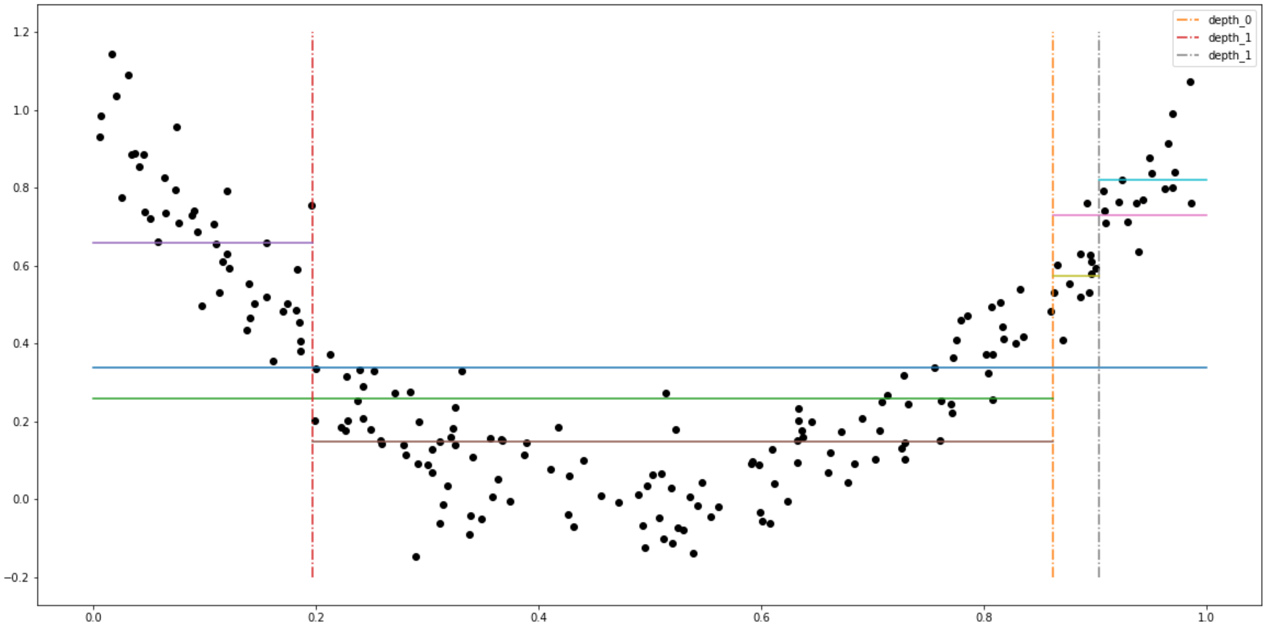
5) The decision tree in part 2, the max depth=2 is fixed. This restricts our decision tree to make more splits. The decision tree can be called restrictive in terms of max depth.

The decision tree in part 3, the max depth=None is not fixed. This decision tree makes splits until only one class present in the leaf node or only one sample is left in that node. So, it also has more orthogonal lines as decision boundary.

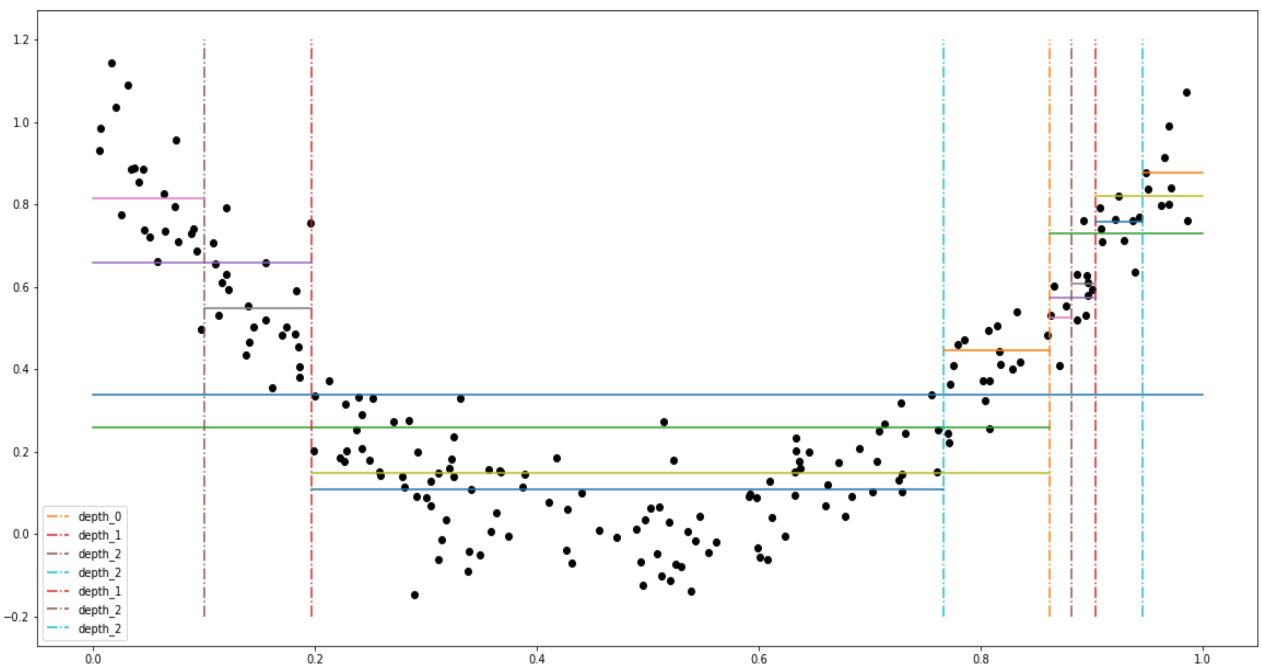
The part 4 has two decision trees trained. One with single threshold and other with 3 thresholds on which 3 orthogonal boundaries are made. The rotation causes the data set to no longer be separable with single orthogonal line. The decision tree trained has less accuracy on rotated dataset as compared to the original dataset as we can infer from decision boundary. Some of the points lies in wrong part.

Regressor

1) Regression predictions of a Decision Tree Regressor (max depth =2) using a line plot:

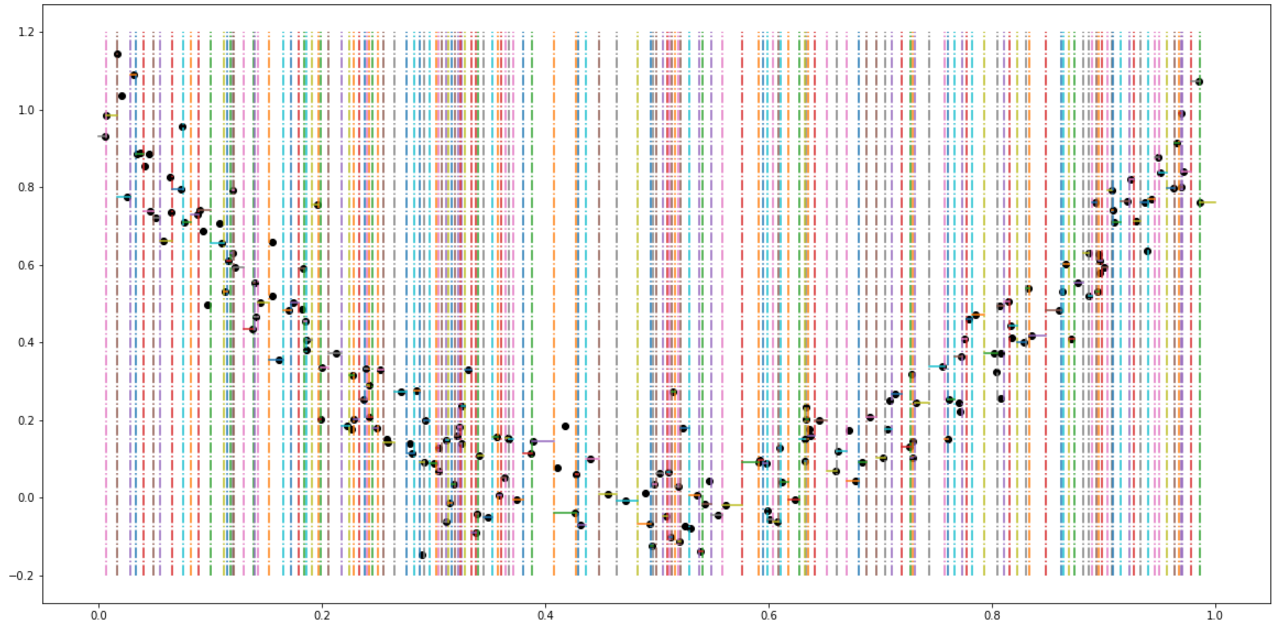


Regression predictions of a Decision Tree Regressor (max depth =3) using a line plot:

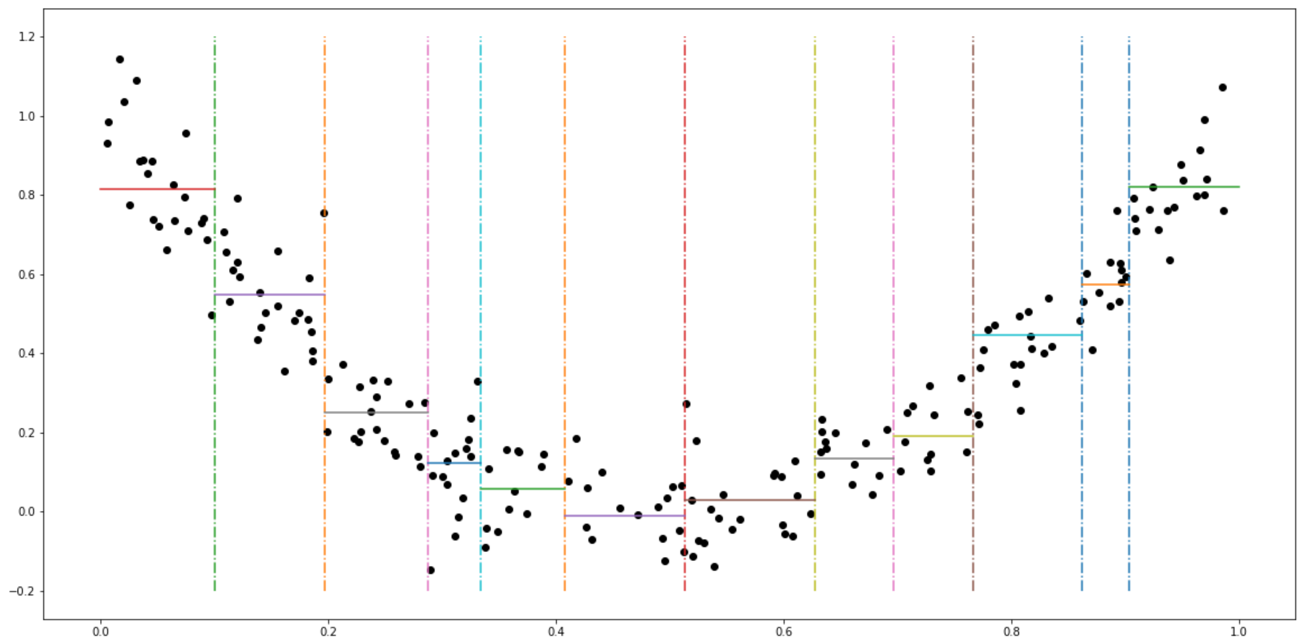


These line plots show the value that decision tree predicts for the values lying in certain intervals. From the graph, we can infer that the line between any two dotted line is the predicted value of the points lying between those two lines. We can clearly see that for **max depth = 2**, there are a smaller number of vertical lines than in **max depth = 3** tree. This means there are less decision intervals in former graph than in second graph. Also, there is depth wise coloured vertical dotted lines and horizontal lines.

2) Regression predictions of a Decision Tree Regressor (min sample leaf = 1) using a line plot:



Regression predictions of a Decision Tree Regressor (min sample leaf = 10) using a line plot:



Minimum sample in a leaf defines the minimum number of samples that should be present in a node so that it can become a leaf node. When we say **min_sample_leaf = 10,** then there should be minimum 10 samples in a leaf node.

Here in the first graph, we have defined **min sample leaf = 1**, the tree will split that number of times till only one sample is remained in the node. So, there will be many splits and many depths will form. On the other hand, when min sample leaf = 10 there would be relatively smaller number of splits. The one with min sample leaf = 1 has more chance to overfit over the training dataset.