Oya İlayda Yalçın

60692

07/12/2020

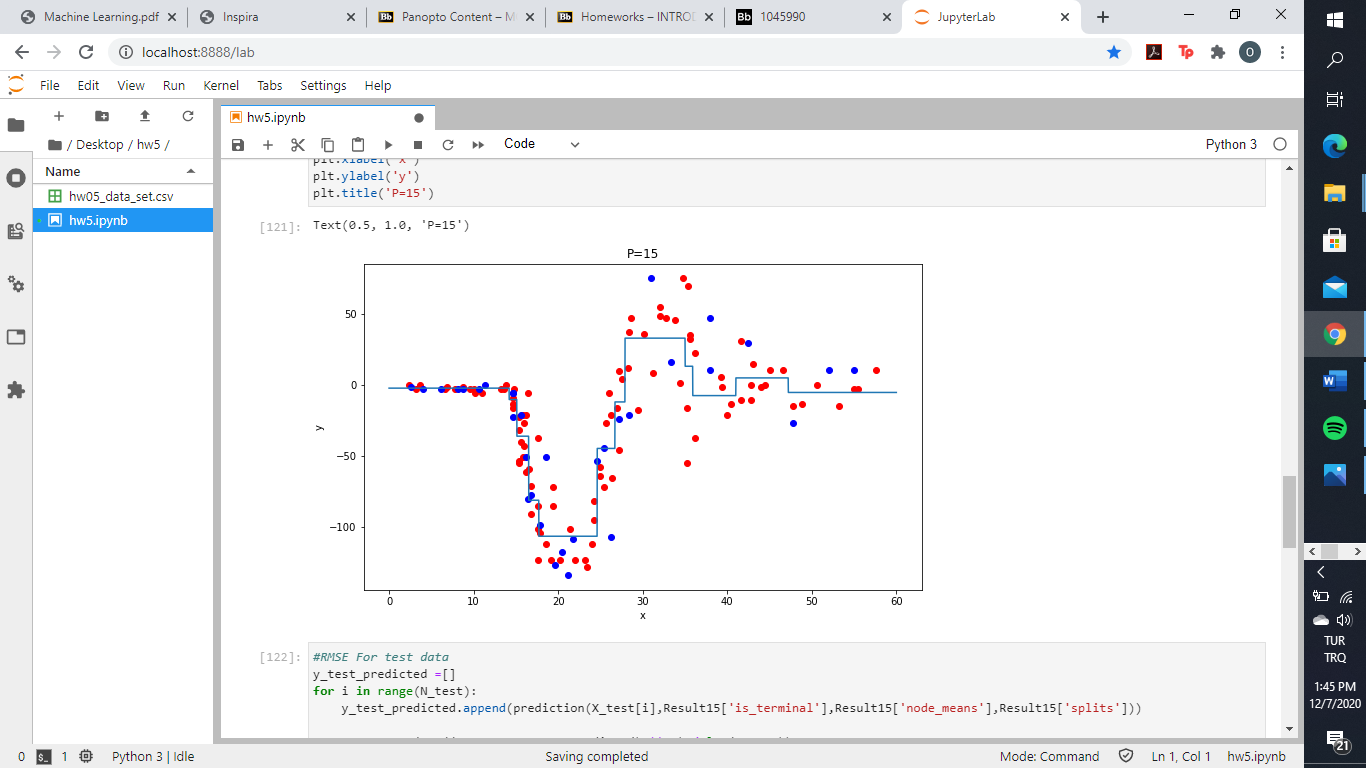
ENGR 421

**ENGR 421 HOMEWORK 4 REPORT**

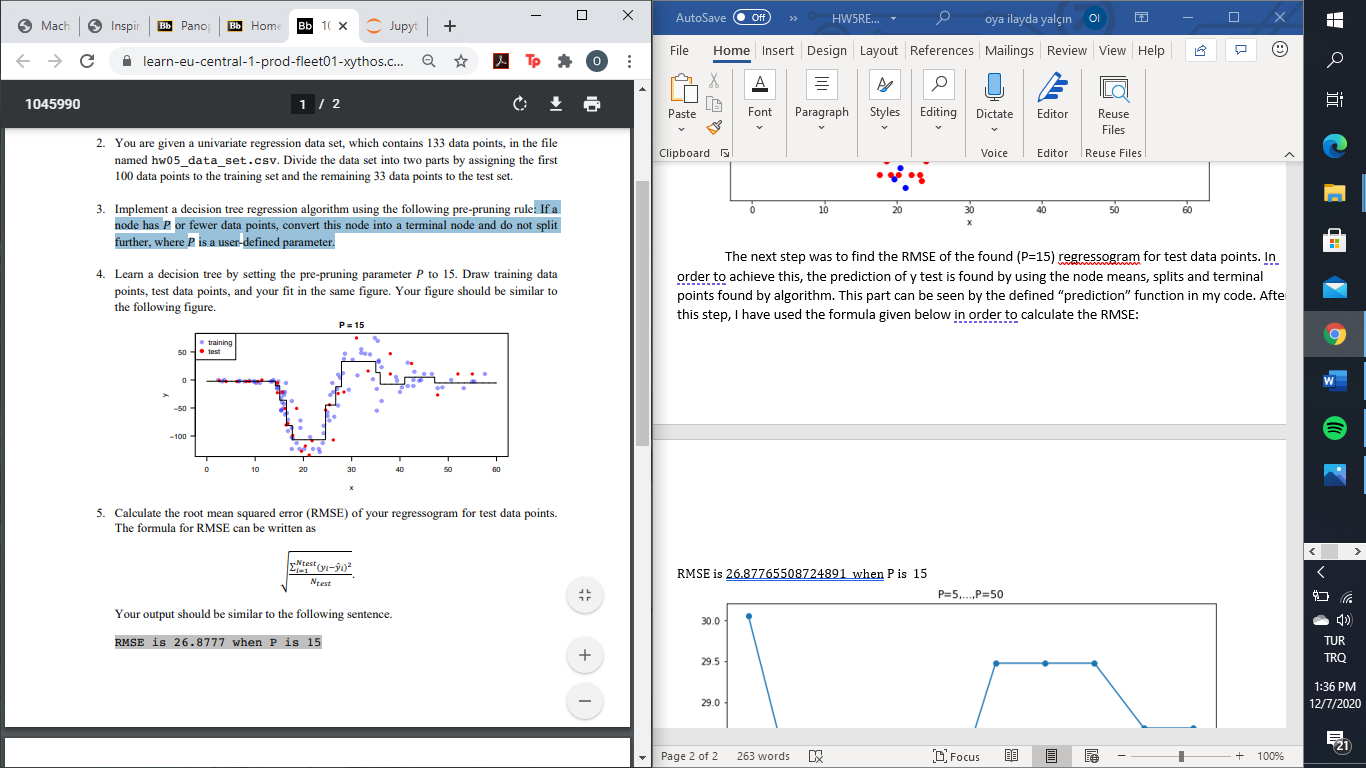
In this homework we were asked to implement a decision algorithm according to the data set given to us.

First, the univerate regresion data given to us has been divided into training set and test set. The training set contained the first 100 data and the test set contained the remaining 33 data.

After this step, a decision tree regression algoritm has been implemented with the rule of “: If a node has P or fewer data points, convert this node into a terminal node and do not split further, where P is a user-defined parameter.” From this pregiven rule, I have implemented a decision tree very similar to our lab7. The only thing that I have added was, node\_means which was to calculate means of nodes since we had to create a regression and we did not have multiple classes for the data given to us, so I modified the lab code accordingly. In addition, I implemented a part in which the algorithm decides whether to go right or left. Which was again to find a node mean. This algorithm working with a while loop, had a break commend when P become 0. For our first part P parameter was given as 15. So, for the graph below I have found approximately 33 iterations, when P was 15. The ending graph is also given below, with the training and test data point.(Training set being red and test set being blue)



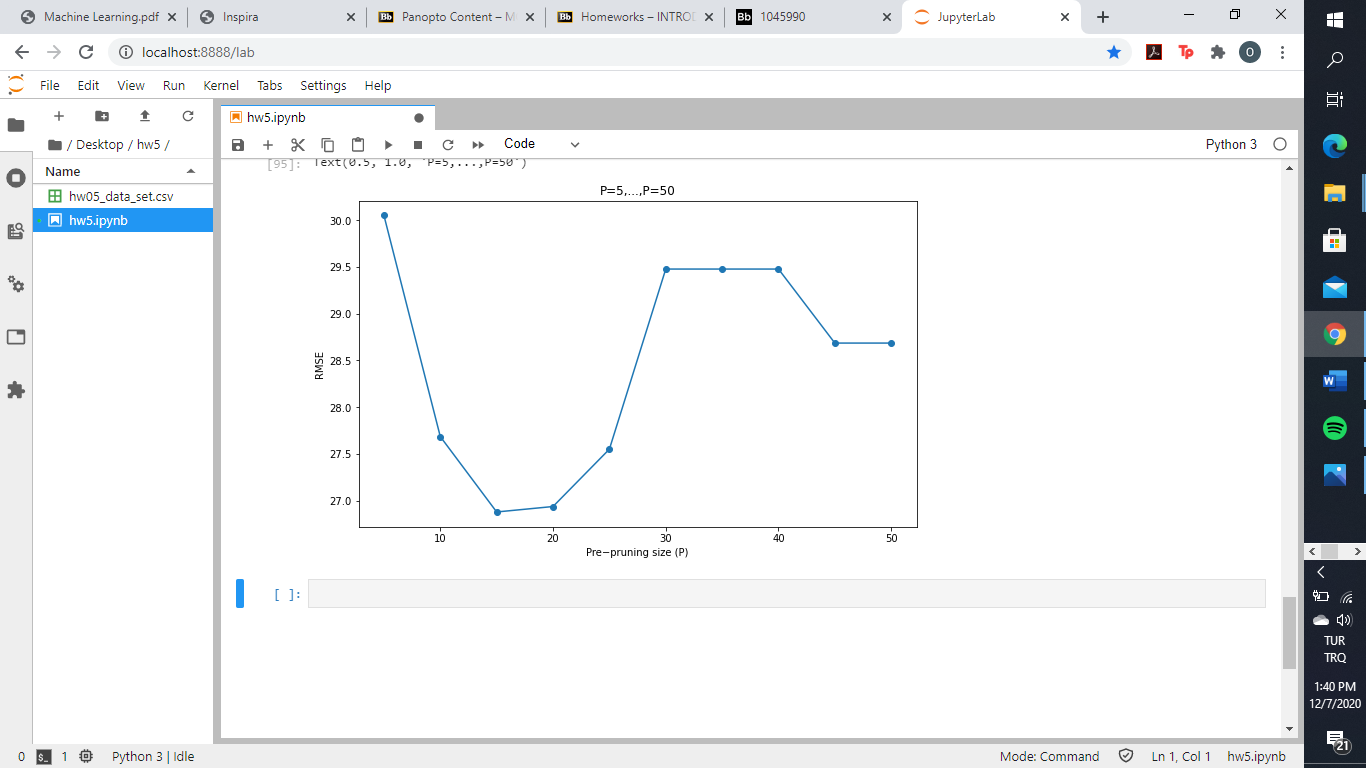
The next step was to find the RMSE of the found (P=15) regressogram for test data points. In order to achieve this, the prediction of y test is found by using the node means, splits and terminal points found by algorithm. This part can be seen by the defined “prediction” function in my code. After this step, I have used the formula given below in order to calculate the RMSE:



In the end, the resulting RMSE when P=15 is given below:

**RMSE is 26.87765508724891 when P is 15**

In the last part of this homework, we were asked to set the P 5,10,15,…50. In order to achieve this, I have created a loop, which changed P accordingly and takes the needed values in order to predict y tests, via the algorithm function defined in the first part. In the end, appending these RMSE’s found for different P parameters, plot visualizing the change of RMSE’s is given below:



Looking at the graph above, it could be seen that lowest RMSE is found at P=15, which was what we have implemented in the first place. This shows, why P=15 is also giving a good fit in the first graph. So comparing with the P’s we have calculated, 15 is the optimal point where, there is no overgeneralization and underestimation.