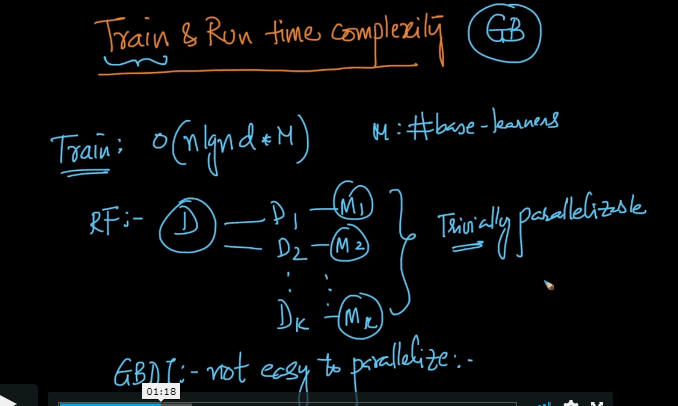
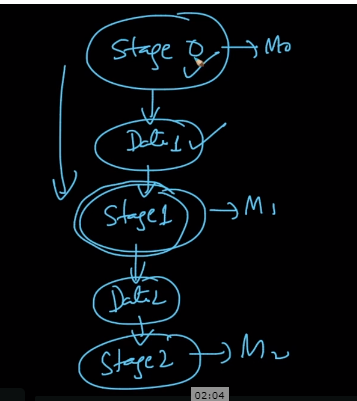
We saw that train time complexity for decision tree was o(nlgn\*d) and since we have M base learners in GBDT so order of time complexity wil look like o(nlgn \*d \*M)



But as we saw in random forest we it is trivially parallelizable i.e. multiple models can be trained simultaneously but in GBDT it is not that easy.

Because how it goes in GBDT is like its all sequential so without completing stage 0 we cant jump to stage 1 because the residual error we get from stage 0 is used as input for stage 1.



May the time complexity for RF and GBDT looks same but GBDT takes much more time to train than RF.

**Run time and Space Complexity:**

So run time Complexity for GBDT is much similar to RF’s .

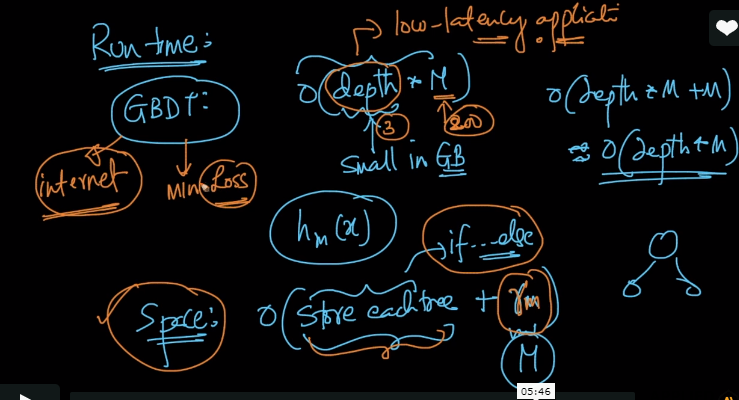
It is o(depth \* M +M) or nearly o(depth \*M)

Now since depth for GBDT’s low i.e. ranging from 2 to 5 or 7 so we just need to store the decision trees and that is nothing but just if else statement and we also need to store Gamma values for each M.

So space complexity looks like

O(store each tree + (Gamma)m).

SO GBDT can be used for low latency applications because of its low run time and space complexity.



**Comments:**

