NPTEL

NPTEL ONLINE CERTIFICATION COURSE

Introduction to Machine Learning

Lecture-62
Random Forests I

Prof. Balaraman Ravindran
Computer Science and Engineering
Indian Institute of Technology Madras

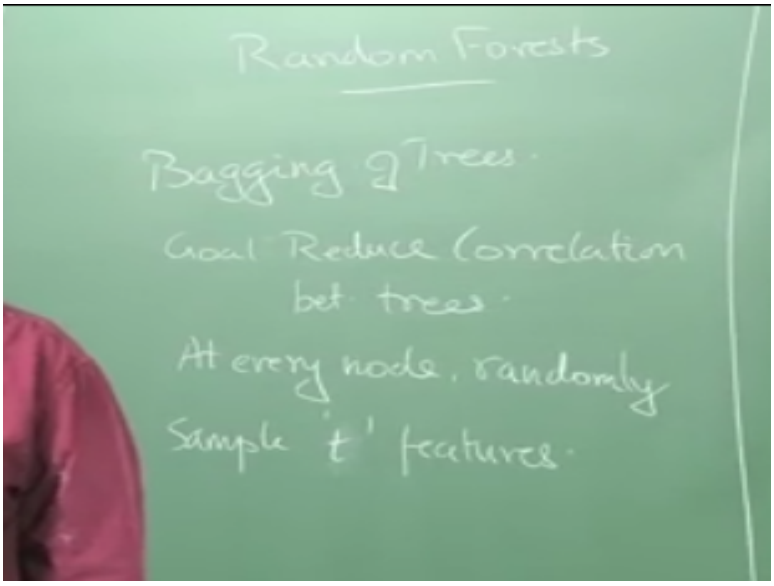
So now we know that trees are great candidates for boosting as well if you are using gradient boosting right. But trees are great candidates for bagging as well right. So what is the important property in bagging that we talked about, what does bagging help us to, reduce variance right. So bagging allows us to reduce variance, and in fact you can show that the reduction in variance is highest if the classifiers that you are building okay, or not correlated right.

So I am building many, many classifiers and the classifiers are predicting the same output right. So if the classifier parameter set I am estimating or somehow if we can make them uncorrelated right, then the reduction in variance is maximum it kind of intuitive right. If the classifiers are very correlated then there is no point, they are not really different classifiers right, they are going to give me the same output, so the variance will be high.

So if you can somehow make the classifiers uncorrelated, then the reduction in variance is high right. So there is actually a very specific relation between the amount of correlation between the classifiers and how much you pay in terms of the reduction variance right. So I am not going to derive that, I just pointing it out to you, so if you want you can look it up it is there in ESF right. And if you think about what we are doing with bagging right, we are taking one data set right and you are sampling with replacement from that right.

So the probability of the trees that you are generating being correlated express rather high, the probability of the trees that you are generating being correlated is rather high. So it can be come up with some way in which we can reduce the correlation between the trees that you are constructing right.

(Refer Slide Time: 02:17)



So I am going to be doing bagging right, but the goal is to reduce correlation between trees right. So the people who came up with the random forest had a very, very simple idea for doing this right, you start doing bagging as you would normally do okay. So you have your data set, then you create a bag by sampling with replacement from that data. And now when you start building the tree on this data set, so what you do is at every node right, sample some P features from your feature set, we use P for the regular feature description right.

So let me use a different, so we have total of P features right your data points come from some RP space, you use randomly sample some T features from that P features okay. Find out which is the best split point, which is a best split variable among these P features alone, split the data go down to each of the subsets repeat the same process, sample and other T variables right not necessarily this joint, and just sample again okay, sample again T more variables okay.

And then try to find out which is the best split point among these T variables and keep doing this. So what does this get us see if you had actually worked with the same data set right, which I mean if I just done bagging at the root level it is highly likely that each one of the bagged trees would have picked the same attribute right, just because you have sampled it again right. So it does not mean that the very predictive attributes will get discarded.

So at the higher levels of the trees it will look very similar right. But now you are getting rid of that I said okay, I have chosen randomly I have chosen T variables and only from them I am

choosing the best variable therefore I am reducing the chances that the trees will look similar. And in fact we can show that this leads to significant reduction in the variance, in the bagged estimate and random forests end up performing very well.

In fact random forests are competitive with gradient boosted trees in some applications and vice versa right. So boosted trees are better than random forests in some applications and random forests are better than boosted trees in some applications. And because some time, till sometime back there are very efficient random forest libraries and people use random forests a lot right. But now there are also very nice libraries available for radiant booster decision trees. And therefore, try everything and see which works right.

IIT Madras Production

Funded by
Department of Higher Education
Ministry of Human Resource Development
Government of India

www.nptel.ac.in