NPTEL

NPTEL ONLINE CERTIFICATION COURSE

Introduction to Machine Learning

Lecture 39

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

Decision Trees – Introduction

So we are going to shift gears and we are going to look at a very, very popular supervised learning algorithm and also provide learning model I should say because there are many different algorithms for estimating this model or based on decision trees right , okay people at the back hear me fine okay. So decision trees have a very I mean very special place in all this machine learning stuff in that they are very widely used right and very poorly understood know in terms of I mean we talked about all this bias-variance tradeoff classification area we can show convergence we can show approximations and whole bunch of other things for all the linear classifiers linear regresses let me know lot about all the linear stuff right.

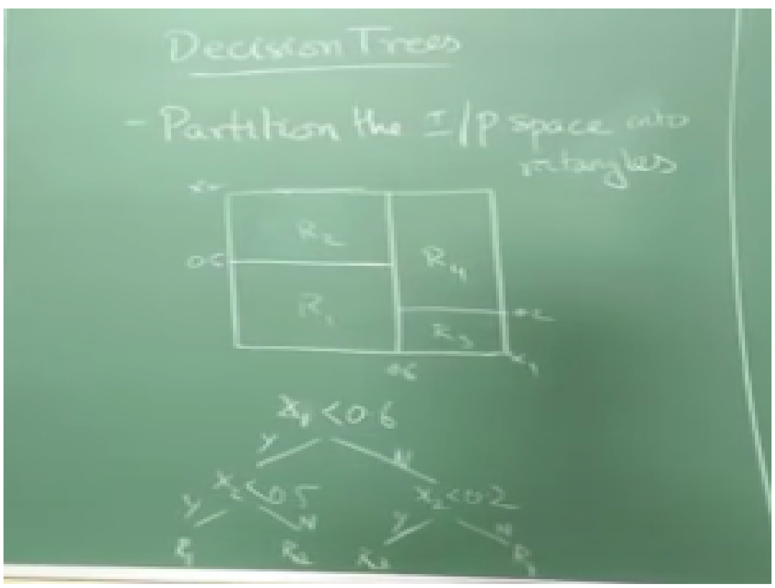
And some of the non linear stuff like SVM's and so on so forth again we have very strong theory we know about convergence and things like that right with decision trees I can tell you what the problem is and I can tell you what is the best known heuristic for solving the problem but I cannot even tell you how good the heuristic is right because there are isolated results under very special conditions people have some results on how good the heuristic is you know that is this best possible tree that you can learn right and how close will it get you to the best possible tree right.

So there are some very isolated results right but there is nothing that really something out there that we understand well right and it is incredible because it is such a simple idea it is a very simple classifier right. So it is more or less along the lines of how humans do their decision-making right, so when you are trying to decide whether something belongs to some category X or some category Y right how do you think you go about doing it right you do not do not create a hyper plane in your head right.

So typically what you end up doing is okay this is red? okay it is not red okay Is it round? oh yeah it looks round and round and okay it is blue maybe it is that right essentially what you are doing is querying some properties of the object right or some properties of the entity at hand right do I want the when do I think he is a studious boy or not right then I can ask all kinds of queries okay let us see show up for all the classes as you sit in the first row all the time is he smiling after quiz one.

So I can ask him in again build this thing so I can ask the series of queries right and then I can essentially am building some kind of a characterization of the object and then I say okay, so this is this person is class one okay this person is class 2 right. So on so forth that is the whole idea behind decision trees right you are essentially trying to if you think about what you are doing.

(Refer Slide Time: 03:23)



You are trying to partition the input space into certain regions okay right the feature one has value X feature two has value Y feature three has value Z and that gives me some region in stage space so what do I mean by that let us take a right. So the first question is a two dimensional data set now let us forget about all the all the relevance to real they real-life and things like this I have two variables X1 and X 2 now I am going to ask the question first question there is a new data point is the x1 of this data point greater than 0.6 or lesser than 0.6 okay the first question I asked.

So what am I doing in some sense I am right so I am splitting this into two parts its greater than 0.6 it will be here which is lesser than 0.6 it will be here right next question I can ask is okay suppose X 1 is greater than 0.6 okay then is x2 greater than 0.2 or not right then what do I do right, so this region is now x1lesser than 0.6 X 2 greater than point I mean x1 greater than 0.6 X 2 less the greater than 0.2 likewise this is x2lesser than 0.2 and likewise here I can ask the question is what kind of question can I ask you come on let us just say something random and do not think too much about it you can make any conditions on X 1 and then you did with X you can do anything right.

So typically they alternate but you can also ask a question of case X 1 given that X 1 is less than 0.6 is it less than 0.3 or not or you could ask is X 2greater than not give me some root 2 or something but 0.5 okay good right. So as soon as I write that thing that it becomes 0.5 so the regions that we can act arise each do class are in the rectangular but have some but lately on something i Function and lately a slant of the saying I can I come to that little later right.

So I am just trying to mimic the way we try to think of things right so normally what this is all of you are agreeing with me when I said okay I w ill think about attributes one at a time right so is the price okay is the TV screen of this as the right size then I am going to buy or not right so this is like that so I am just mimicking that process here okay then we will come to other things a little later right.

So I mean it will be really truly amazing if you are true class labels are going to lie like this right what is the problem what do you think is a likelihood that the class labels will actually be this kind of rectangular regions it could be high I mean depends on what process was used to generate the class label since the problem labels are generated by doing this kind of region splitting obviously they will you have to find the right regions but we are making some kind of an assumption right earlier when we made assumptions about linear right we are making some kind of an assumption about what the boundaries would be right.

Likewise we are making assumptions here that there will be rectangles right so if you do not want to make the assumption that there will be rectangles then it is not going to be little harder, so not only are these rectangles right there is something more special about these rectangles I mean they are all recursively generated right it is not like I can I cannot just take some arbitrary set of rectangles and tile the space right they are recursively generated by first splitting it into two and then splitting each section into two and so on and so forth it turns out right.

So this kind of recursive splitting is what is most tractable to handle right and for most of our decision tree discussion we will stick with this right I will come back and address that issue little later right about having more complex boundaries but almost all decision tree algorithms try all the approximations that we do for decision trees use this kind of recursive splitting of regions right like one inside the other like that how will you describe this region it becomes harder right, see now each of these regions I can describe very easily right.

But if I start doing nested rectangles right it becomes a little tricky to describe the outer region but I can do something like this provided I am willing to accept that right now it is no longer a nested really no longer nested rectangle because I had to actually fragment the outer region that will give you the inner rectangle but the outer rectangle you have to actually exclude it right I have to I have to specify the outer rectangle and then say okay to remove the inner rectangle.

So it becomes a little harder to specific okay, yeah it becomes harder right I wanted to be easy I wanted to if I want to represent this as a tree but then the way you are going it make it harder and harder so the biggest advantage of decision trees is the interpretability, so for example this tree that I have this region segmentation that I have drawn I can represent it as I mean talking to you about trees right where it said tree right the segmentation I have made I can represent it as a tree.

So what I will do is I will first ask the question is x1 less than 0.6 rights then if we say yes I will go left then last equation is right. So I can very compactly represent this rectangle segmentation as a tree right you can see what is here I am asking the question is x1 less than 0.6if it is true I go to the right and then again I ask the question is x2less than 0.5 it is true I go to the right and I say that this r1.

Right so I am essentially here right otherwise I am here then go into the other branch which is x1 is greater than or equal to 0.6 on this side and if it is less than 0.2 mean r3. r is greater than 0.2 I mean r4right so I can very compactly describe this segmentation as a tree right so what is nice about the streets is easily understandable that you show somebody okay so you are building your this you go out to become a data scientist or a data analyst or whatever okay.

Some manager who makes like 10 times what you do but who has ever heard of a hyper plane in his life right comes and asksyou for a to build a classifier here is the data build a classifier and then you tell him okay, I built this classifier and this new customer you should label him as a buyer then he will ask you why right so at that point you just talk to him about optimal separating hyper planes and show him something okay well then the next day probably the manager is going to be running infinitely more than you right.

So what you should okay so what you should be doing is showing him a decision tree right because that people can understand right people even with an MBA can understand right so you can see what is my recommendation to you guys have to finish your B Tech or anything, do not do it MBA anyway so that is easily interpretable right you say oh that is oh yeah you should you should classify him as a buyer because well this is on this parameter he is so much on that parameter is less blah, blah and then there you go right the biggest advantage of decision trees is the interpretability always easy to explain the decision tree to people.

In fact so much so that at one point when neural networks were at their peak you know you know what is the biggest problem with neural networks the opposite of decision trees incomprehensibility right this is interpretable and neural networks are incomprehensible essentially you say okay it is a black box I do not even know what hyper plane it is learning right if I get if you think optimal separating hyper planes are hard I cannot even visualize what the neural network is learning right.

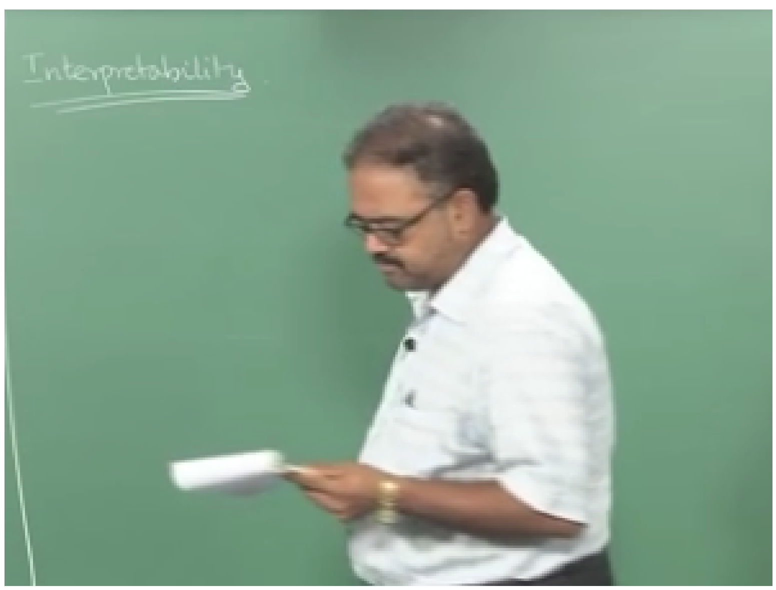
So you see here is a black box you throw in all your data at one end something will come out at the other end you just take it on faith right I so welcome to the Church of neural networks right, so that is essentially how neural networks are working so when the neural networks are at their peak there was this whole line of research where people took a neural network like that was trained on your training data etcetera, etcetera.

And then try to construct a decision tree that will give the same decisions as the neural network will give the same class labels as in your network so that you can actually understand what has what is happening it sounds weird right but remember that now I am no longer using whatever other heuristic I had for constructing the decision tree I amusing a decision tree which makes the neural network so if the neural network learned some complex function of the data right I am trying to build a decision tree that mimics the complex function right.

So it is a different decision tree that I would come up with than the one I would have constructed if I had used any of my decision tree learning heuristics on the data from the beginning okay so that is value to doing this right, so people do see that right that is value doing this a because your networks do something I cannot understand right but they seem to work now they give me a wonderful answer and I do not know what the answer means so can I use something for which I know what the meaning is and try to understand it in terms of that right.

So that is how useful decision trees are right even if you have a more complex learning mechanism at hand okay sometimes for interpretability sake so you can use decision trees right.

(Refer Slide Time: 15:28)



How expressive are decision trees. If you remember we have the discussion about neural networks I said if you have two layers of weights there is three layers of neurons and you can basically represent any Boolean function right as the branching factor might be very high but then you can represent any Boolean function so neural networks are universal approximators as in that sense is what about decision trees?

So you can it can be an Universal approximated as well right I can just keep dividing and subdividing this space okay just that my tree might become very, very large right as long as there is some kind of guarantee on the function yeah. Now I can define some variables here it is fine yeah in the eye I could do that I mean in fact he was pointing out in the beginning I could have as well draw another line here yeah okay.

Everywhere that the discussion was when we were discussing about this line versus that line yeah sure I can keep doing this if that is your question right now this becomes a much more complex tree right, so now I have splitting here once more right I will have another branch here will have another branch here and another branch somewhere there right so keeps becoming more and more complex I can but the point is conceptually you can represent anything right.

So it is powerful in the sense that it is a universal approximator right and it is just that the number of parameters can grow unbounded okay and that is another thing nice thing about it is nonparametric ever we talked about what nonparametric means right, so decision trees are actually nonparametric it can just keep growing right we can keep adding parameters as you go along.

IIT Madras production

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

www.nptel.ac.in