NPTEL

NPTEL ONINE CERTIFICATION COURSE

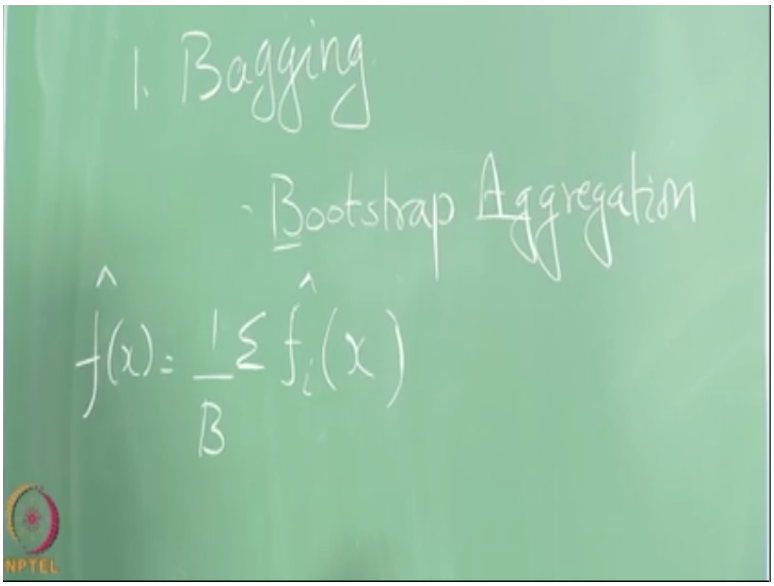
Introduction to Machine Learning

Lecture-59
Ensemble Methods- Bagging,Committee Machines and Stacking

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

So we will move on to another topic which is essentially ensemble methods so what do people do in ensemble methods is that instead of using a single classifier or repressor you tend to use a set of them right in order to make the same prediction typically these end up improving some performance or the other of the of the classifier right statistically speaking more often than not they end up in reducing the variance of your classifier right but that also ends up giving you better empirical performance at the end of the day right.

(Refer Slide Time: 01:17)



So we are going to talk about several approaches for an ensemble methods I will start off with the one that is familiar too familiar to all of us called bagging right so, so what is bagging why did I say similar at all of us so bagging stands for boots top bootstrap aggregation okay do not

ask me how they dread bagging out of boots bootstrap aggregation right but the idea is very simple all of you know what bootstrap sampling right select mean about bootstrap sampling.

So what I am essentially going to do is I am going to create you give me one training set of size n I am going to create multiple training sets of size n by sampling by replacement and then I am going to train a classifier on each of those sets I am going to train a classifier on each of those sets and how will I combine the outputs of the classifier sorry I can do a majority vote or average what sorry cannot end up blowing majority vote right.

So average what average if you can if your classifier is going to produce probabilities for the class labels right I could do some kind of a weighted average of the probabilities the classifier is just going to give me one or zero right I end up essentially doing majority vote okay does it make sense so the idea is very simple backing idea is very simple I am going to produce lot of classifiers right so I am going to call them fi right so it could it could be it could be regression as well.

So it does not have to be classification right the situation I just take an average of the outputs of all the classify so each fi is trained on one bag which I have produced from the from the original sample like this is another back derivation from the word left bagging it so it is bootstrap aggregation but then each of those bootstrap sample you produce is sometimes called a bag right so if I produce be bags then I eventually average by I mean be to get me the prediction.

And if I am doing it for classification I can produce majority vote on average the probabilities okay the few things to note so backing reduces variance right so in, in effect it ends up giving you better classifiers normally then what you would get by training on a single sample of the data right or producing a single classifier it is particularly useful when you are dealing with unstable classifiers right it can take an unstable classifier and produce something that is more stable right that is just a fallout of reducing variance right it can take an unstable classifier.

And produce something more stable so one thing that you have to be careful about when you are bagging is that if you bag bad classifiers the performance can become arbitrarily worse something that has a classification accuracy less than .5 less than or equal to 0.5 two classes sorry each when you change the data on which you train the classifier right you are going to end

up with a different classifier in sets of data yes you could as well if you want to but it is a good point if you initialize two different variables different values for the parameters.

You introduce an additional source of variance there but the you could you could there is nothing stopping you from doing that just that you have to be careful about how we do the analysis if at all you are doing a variance analysis now do be careful about how we do the variance analysis right yeah so by that he brings have a good point so in some of the ensemble methods that we talked about right the ensemble is will typically be of the same kind of classifier okay.

The only way we are distinguishing one classifier from the other is by training it on a different data right except for one approach which I will tell you later where typically if we use end up using different kinds of classifiers okay so data would be the same but the classifiers would be different but this is one of our aggression but anyway soon I am not using different variables it is like suppose he is using a neural network right so you need to have an initial starting point for the weights right.

So if I use a different random starting point what so that was this question should I use the same random starting point or should he is a different starting point right and even then going back to your question right so you think about it this way right so right we are talking about f of X instead of that think of it as right so this hi will give me whatever features I want from x even if I want to run each classifier on a different subset of the features it will just be that that will get n rolled up into the classifier.

I can still do the averaging if I want right that is not an issue but that is not the question us asking anything else on this okay so if you throw a bad classified into the mix right your performance can become arbitrarily bad so that is something that you have to guard against okay so bagging is a very, very intuitive very simple thing and a couple of more practical things about bagging is that it is you know what they call embarrassingly parallel you know you can run how many other instances of training on bagging you want at some of the other ensemble methods.

We talked about are going to be inherently cereal in nature right so allowed to run one after the other right suppose you are looking to run this on many large data sets right so doing bagging is kind of easier because you can run it because one, one bag like or classify trained on one bag

does not depend on a classified trained on the other bag in any way right so they can be trained independently and trained in parallel.

So that is the first thing okay next thing you talk about something called committee machines okay this is nothing big okay all it says is it read a lot of different classifiers let us say we have some glass or something no and all with all the individual classifiers performed well on test eight or do they just have to learn the training do they have to generalize well or so each classifier you typically train it using whatever is your normal training procedure right.

So if normally you would expect it to generalize well on the test data right so you would want to produce classifiers that generalize well on the test data right but that is a call to be made I mean if you do not want to test each in every classify sometimes people just tested the what they call the bag the classifier right the combined prediction alone is something that they test they just train each classifier on the data that is given right.

And then they test the combined classifier there are multiple reasons for wanting to do that so one is that typically the classifiers that you use in bagging or not very powerful classifier right so the chances of the mover fitting or low so you do not really try to do a validation on the test set to make sure that it is not over fit and things like it because the classifier itself is not very powerful classifier and then you just go ahead and test it on the tester combined classifier on the data right.

So why would want to test the combined classifier on the data you will want to know whether you should produce more bags and think like that right so the nice thing about the bagging is that because you are using at any point of time you are only using a weak classifier to fit the data right and not if we classified but not necessarily a you know very strong classifier to fit the data the chances of you over fitting is very small even if you increase the number of classifiers in, in the bag even if increased number of bags and I can do this for 10,000 samples 10,000 such bags right.

And I still want to overfill the danger of over fitting is no more than training it once right so that is a nice thing about bagging I can keep making the classifier I can reduce the variance in my estimate more and more but I am not getting into any danger of over fitting right so that is a nice thick man so the other thing is committing machines did have to really think about anything you

do not even have to think about how oh my god how do I paralyzed this thing that is this typically.

I mean it is a term that people use in architecture and they are trying to think of parallel computing so things like though this is embarrassingly parallel so I can do dude that whatever levels of parallelism I want and things like that maybe I am misusing the term but, but yeah but it is really easy to paralyze right you can just want it on different sample separately yeah so what is embarrassing about it I mean why do you even have this whole parallel computing field to study something that can be paralyzed.

So easily so I am really embarrassed to be working in parallel computing and just making it up okay, okay committing machines can I want the committee which this any other questions okay so computing machines is very simple idea so I am going to train I have given a data set and I am going to train a lot of different classifiers on, on the given data right and then I am going to combine their outputs right based on some kind of waiting mechanism okay so what could be the waiting mechanism.

I will try the neural network whatever it I trained many, many different classifiers right and then I have this set of classifies that have already been trained right and I have to combine their output how do I go about doing this in there are many ways in which you can combine their output right so I am just taking this classification from the textbook elements of statistical learning and not that I completely agree with it so in committee machine.

Suppose I have M classifiers the weight is assigned to each classifier is 1/M, 1/M so I treated classifies as being equal right so that is called a committee machine and I have many different classifiers as the outputs of all the classifiers I am going to give each one of them an equal weight edge or equal vote right so I call that a committee right then we go on to something more for interesting called stacking no badge me says the same classifier same algorithm but trained on different samples of the data right in committee machine.

It is the same data but trained on different algorithms right so I have a three and I could have never let works I could have anything right or it could be says it could be neural networks with different number of neurons I mean I am not saying that it has to be a completely different algorithm it is the different classifier it could for different settings of the parameters and so on so

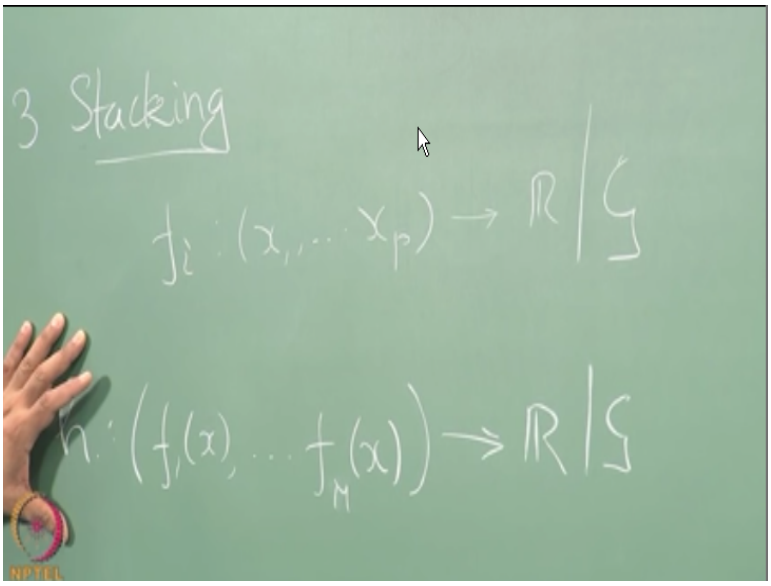
forth right and so starting the stacking is like committing machine so I have many, many different classifiers right but what I am going to do is instead of arbitrarily assigning a weight to each of the classifier what will I do what can I do.

I could do that but with stacking what do I do I learn the weights right so that is a natural thing to try and do right so I have the prediction that is made by each of the classifiers right I go ahead and I learn the weights so another way of thinking about stacking is the following so I use each of these classifiers okay to generate a feature for me so this way it is called stacking so I have a set of classifiers right they all output some it could be a probability vector or it could be just a class label or whatever at the classifier one comes and tells me okay.

I think this data point is class 1plus if I to comes and tell me I think this data point is class to the classified three comes in tell me I think the data point is class 1 and now what will happen is i will my input to when next machine learning stage will be class 1 class to class one right and now again it is a machine learning algorithm now I can run whatever machine learning thing I want it could be linear regression because I am interested in finding weights right so doing some kind of regression seems to make sense right but then you know problems with regression classification all of you know.

That so you might want to use it for classification or you might want to use some other method for classic you might want to use logistic regression for classification whatever it is right but then the inputs to this stage or they are these outputs of the first stage of classifiers and they try target is the same target as the first stage the same class level right so one way of thinking about it is like stacking these classifiers one upon the other so I first have some set of classifiers they produce features for me right the features are essentially what the class what they think are the class labels right.

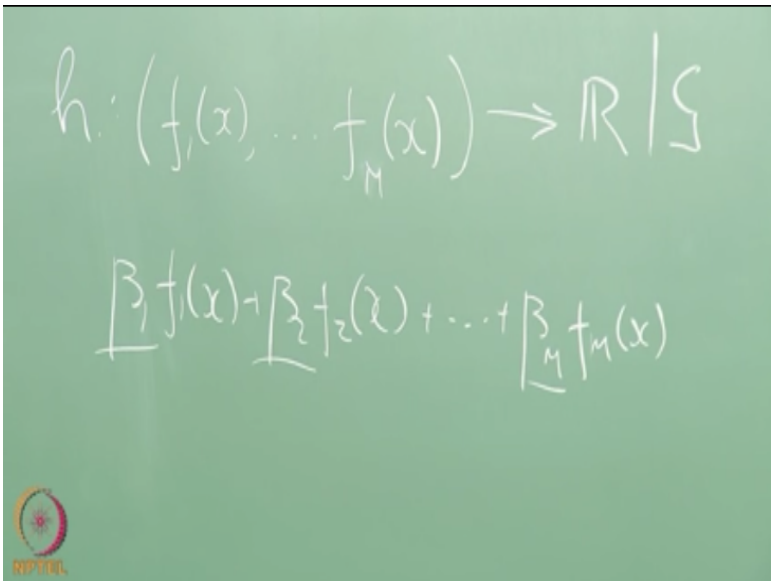
(Refer Slide Time: 18:13)



And then I learnt to combine the features I learn a predictor based on these sets of features so that is another way of thinking about it makes sense then make sense okay right let us take a classifier FM either I there are some people are actually saying they did make sense I am trying to make it easy more explicit let us take a classifier some fi right so it basically it operates on at X_1 to X_P right I just going to give me something right it is going to give me real number or, or some, some class label okay.

So that is basically the, the function you see there are closet does classification or regression or whatever rate so now what I am saying is I am going to Train another H right that is going to take as input right it is going to take this f_1 to F_M as input so F_I is the first level classifier I have m of them right and then h is going to take h of x is going to take f_1x 2 f_MX as input and it will produce whatever is the thing I am looking for real number or write so if you go and look at the structure of h right.

(Refer Slide Time: 19:34)



To make it explicit let us say I want head h to be a linear function right so that will essentially mean that H will look something like right it just going to look something like this so this is essentially saying that okay I am taking the outputs of all this classifies I am combining them in some kind of a weighted fashion the same way I tried any of their face yeah the same training data yeah the same training data that we had gives for f is in use the same training data for h may be, may be not depends on the kind of classifier that you are using right.

I mean H is a completely different training algorithm right oh I can see your confusion okay so my initial training data is going to look like this I do not know okay initiate training data is going to look like this, this is my X and that is my +1 so corresponding to this that will be a training data for H which will be f1 of this guy so now I have only two elements here but there is f1 of this and this is f2 of this and the same plus one comes in here right so I can I can do this so the dimensions do not have to be the same wait so this is stacking.

So stacking is very powerful method and in fact you can do all kinds of weird things with stacking in fact this these weights that I am learning right I can make them functions of X as well what does it mean what does it mean if my weights are functions of X type depending on where in the input space the data is coming from when I might want to trust one, one output more than the other way suppose it should say the top left quadrant of my input space then I trust f1 and f2 maybe a little bit but then if it is in the top right quadrant.

Then I trust both f2 more than f3 less or something like that I can actually do that also so with stacking this function can be arbitrarily complex that is why I did not want to write the linear thing first because it will bias you into thinking about simple linear weighted functions but this hatch can be arbitrarily complex so if you think about it in fact we are doing something like this in the in neural networks right so the first layer it gives you features are complex feature some, some, some hyper plane is being learnt in the first layer itself and it produces a complex feature and the second layer takes all these complex features.

I have produced and it learns to produce a final output right the only difference is the first layer is not trained in this way right the first layer is not trained directly using the training data it is trained using the back propagation error or whatever is the training algorithm you use it is not directly trained using this data so that is the difference right but we already looked at things like this right these are all some kind of general additive models they are called additive models fine so any questions on this can we take less pay directly as it affects so or are you meaning that training like this just simplifies you are the way you are doing it basically all, all your Plus page can be linear.

But a combination but any combination of linear classifiers you will be able to explain much more complex cream using stacking the basic idea is, is that when my classifieds need not necessarily be linear classifiers see the thing is so any of the classification algorithm that we are looking at right comes with two own biases in terms of what are the class of functions it can fit and so on so forth right and it could very well be that across the entire input space the function is so complex the final function.

I want to learn it is so complex that no individual classifier can actually fit it or if I try to fit it with a single classifier and it end up with something that has too many parameters so when you do this kind of this layer wise training so the I can get by with whatever I know a simple classifiers in the first leg first stage right I could use decision trees it need not necessarily be linear so addition trees are simple enough I do not have to grow it all the way out that I can stop at some point I can use decision trees.

I can use neural networks whatever I want as my vice right and then later on try and combine the outputs you could you could given how much success people have had a deep learning I would suspect that if you if you work on this carefully yes you could have multiple levels of stacking

people do that I mean it is not that it is not that the reason it is called stacking is because people actually did multiple levels so it could do multiple levels of this but then the question comes how do you group these things and so on so forth right do I do I run it on every give all my first level classifies as input.

To all my second level classifiers or should I group them somehow to a form a hierarchy so those kinds of issues will arise as long as you can address them sensibly then you can go ahead and do multi-level stacking is one thing which I wanted to mention sometimes when people want to run you know competitions and so on so forth they do something very clever they do not expose the actual data to the competitors they give you the outputs of the first level classifiers okay.

And then the actual class level they do not tell you what the features were that they measured they give you the outputs of the first level classifiers and then they give you the class labels and then now all you need to do is train your second level classifier take the first level classifies output as inputs and train the second level affair and see what we can do with it in fact it ran for a couple of ways.

I do not know I am not sure they are still running there is an ensemble learning competition which essentially does then this also allows you to have some amount of data privacy rights I do not have to release X but I am releasing some amount of simple functions computed on X okay and then you build whatever classifier you wanton so it is hard for me to reverse engineer because I do not tell you what F is even a very tell you the output of F I do not tell you what F is I do not tell you what X is so it is very hard for you to recover.

Because you cannot essentially compute F inverse right so, so that is another nice thing that you can so the reason that there are so many approaches for doing this is because there is no one clear winner under all circumstances way so yeah so it depends so in fact like I said so stacking is something that you can use under a variety of circumstances you can even use it under cases where you cannot do baggie how so I mean I can use stacking when I do not want to have you want to give you access to data right so that is one case in other cases the data set is small enough.

That bagging does not make much of a sense on that right so it is not really truly representative of the underlying sample and I am not really sure we want to do that in which case I can use the

different biases given by my multiple classifiers as my that is my variation right to my ensemble so there are different ways in which you can do this next thing when I want to talk about which is more interesting thing yeah why yours and I run like I train them on a single data set and then I get the percentage of points misclassified and then I normalize them through all the classifiers like say 3% 5%.

I normalize them to come for each of the classifier you know what is a percentage of data points I got wrong okay learning this when β's are not a function of X oh that is one way of finding the β here nothing wrong with it like when betas are not the function of X so how else would be you distribute the β's that is one way of estimating β I mean what you mean missing how else would you read this remove it I can think of many ways of doing that right take the classifier that has a smallest error and give β_1 to that and make everything else.

But why would I even want to give weights to classify switch which give me higher error than the lowest error okay give me an answer why the possibility of them you having a better chance no it could be making errors in different things so I might have a one percent error another guy might have a 3% error but he might actually be capturing that 1% correctly the one that I make the error on he met the other classifier might get it correctly so I do not want to completely neglect other classifies also so that is the reason why you have to go about trying to be more clever about the weight assignments.

I can do this proportional to the errors in fact there is another, another more Bayesian approach for doing this I can look at the likelihood of the classifier given the data and I can assign the weights proportional to the likelihood the higher the likelihood the higher the weight and I can do some kind of normalization on that so that my β sum to 1 and I could do that as well instead of just looking at the error the error would be a frequentist way of looking at it a Bayesian way of looking at it.

We look at the likelihood of the data and do this there are many ways in which you can derive the data and so stacking this takes it to the extreme say okay fine I am not going to worry about a specific algorithm I mean I am just going to let it learn from the data directly someone right so yeah there are pros and cons and whole bunch of things so yeah empirically if you want to validate it finally do cross faction and then test against all of these ways of generating the joint classifier right.

But then analytically also you can try to analyze some of the variance of the combined classifier and you will end up hitting the wall at some point that says that it depends on the characteristics of the variance of the data so that is basically what that I mean we are entering territory where you can come up with whole bunch of things like ensemble methods like a thousand papers or their research papers out there which I proposed lots and lots of variations right so think of something and before you try to publish it do a literature survey you will probably find it there ok so that is, that is, that is how crowded the spaces.

IIT Madras Production

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

www.nptel.ac.in