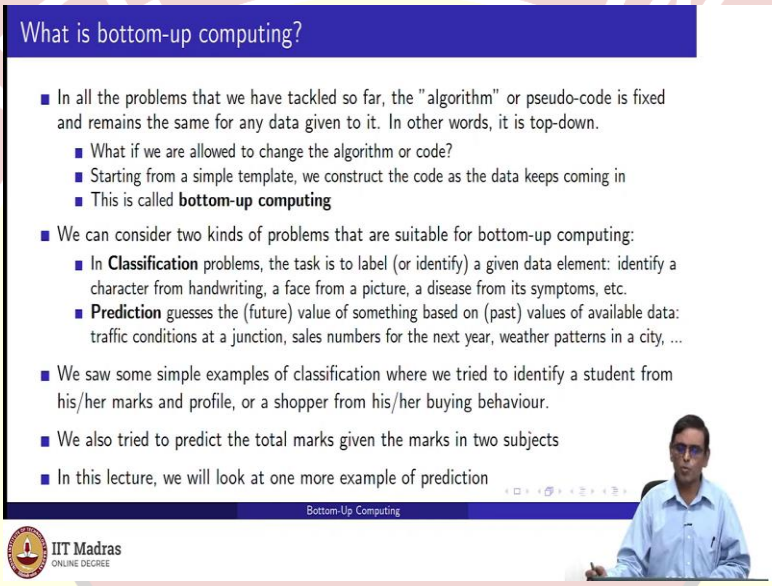
IIT Madras

ONLINE DEGREE

Computational Thinking
Professor G. Venkatesh
Indian Institute of Technology, Madras
Formalized notations and summary of bottom-up approach

Welcome back to this next big session of computational thinking, in our conservational videos that we saw, the lectures that we had so far, you must have seen that bottom-up computing method is a very different kind of method from what we have seen so far. How is it different?

(Refer Slide Time: 00:38)



It is different because typically all the algorithms of low coder we wrote, the code remains fixed. We look at all the data, we do all the analysis, we write the code, once the code is written, it is good, it is fixed and then you can keep giving it different data elements, it runs through the data elements and gives result that is how the code was prepared earlier.

Now, we are asking, that is good. That is why the normal algorithm or pseudocode method is what is called top-down method. Now, we are asking what if a code can keep changing nothing to prevent our code, keep on changing, you can keep on rewriting a code write. So, what if we can basically change the world as we see that. For first two data elements the code might be okay, but then a new data element comes you find that this code is not looking, So, we change the code.

So, if your code can become dynamic, it will keep changing, then it becomes very hard to understand what is going on, that is one of the problem. So, we need a very simple template, which is then used to produce the code, if you do not have a template, if you allow any code to be added or subtracted from the given piece of programme, then after some data comes in and goes out, your code is not going to be any more like what it was before, you will never understand what is going on. It is only bugs in it, you will never be able to find. So, as a tester programme, you know analysing it you must have seen not data.

If you keep changing then, you know, it becomes. So, we want a simple template from which the code is constructed, so that you are not getting random pieces of code coming, it is filling from kind of bucket, something is there, inside which the code is getting filled, in which the code is getting filled, so it easier to understand what is going on.

This kind of learning programing method, computing method where there is a template into which you can add anything and the end of some amount of data, the template is completely filled out. Now I have is a code and then you capitalize this code with data element which is called bottom-up computing. Now, as we saw in the previous lectures, there are two kinds of problems that are suitable for the first bottom-up computing.

The first kind of problem is what is called a classification problem. The classification problem, the idea, the task is to label data set that is coming in, the data limit that is coming in, as something. Like for example, you want to label something as a car or a tree or something like that if it is a picture or if it is handwriting, you want to basically identify a character is it, A B C is it 1 or a 2 or a 0.5 or something or if you have a picture, you might say, if it a picture of a person, then you basically from the page, you would not identify who the person is looking at the new size and all.

Or you might has the data might actually be, the symptoms the doctor has collected about the patient, can I identify the disease by using this symptom. These are examples of classifying. So, classifying the handwriting into characters or the phases into people or names or people or symptoms into diseases that is all. The second kind of problem we have is what is called prediction, prediction, what you do is you are given some data values and you are not given some data values, you want to find the not given data values from the given data.

Typically for example, there are some observable values you can see of something going on and there are some hidden things that you cannot see. So, can I guess what those hidden things are? By looking at what is observed? I cannot see like that thing, can I guess what the hidden thing is by looking at observers is one way of doing things, the other way is because I cannot see the future.

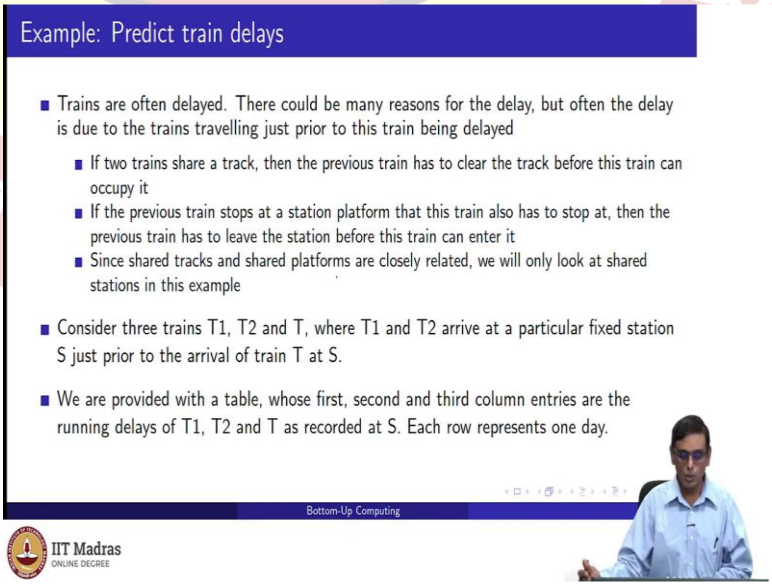
Some people say I can see in see the future I cannot believe it. So, can I predict what will happen in the future by using past values. So, one way is to use past values to find future value, another way is to find hidden values from observable value. So, these are two ways to use prediction. So examples are, for example, you can write or predict what the traffic will be at 10 o'clock tomorrow, tomorrow now a or what will be the sales of this company next year or what is going to be the weather the weather in Chennai .

So, these are all things that you can do try and predict from the data. So, I have past weather patterns for all the days of the year for the last 30 years, I can use that to complete the weather for Chennai, personally says also I have various parameters I can use it to figure it out. So, this was simple example of classification, if I identify a student from the profile or from the March or something not great or we try to find a shopper, who is the shopper? Who is this srivastan or ahmed all, we are looking at the buying behaviour.

We also try to predict the total marks given marks in two subject, this is more like a predictive thing. So, we said directly given master marks in physics, maths and chemistry, can we find physics or physics and chemistry can you find math? Something like that. So, in this lecture, we are not going to discuss classification.

This is going to take one more example of predictions, because the previous lectures that we had, we kind of dealt with little bit, not too much, perhaps you get no benefit by seeing one more example in a little bit more detail through slides and so on. So, we are going to actually go through one more example prediction in more detail.

(Refer Slide Time: 05:50)



So, what is the problem we are going to address, the problem is that of predicting train delay, everybody has faced this problem. You go to the station you are expecting the train to come 10:30 and you know at 11 o’clock it is still come, 11:30 it has still not come. You have meeting, you are nervous, you need to be someplace and so on, so all these issues, this train is also delayed, train is delayed.

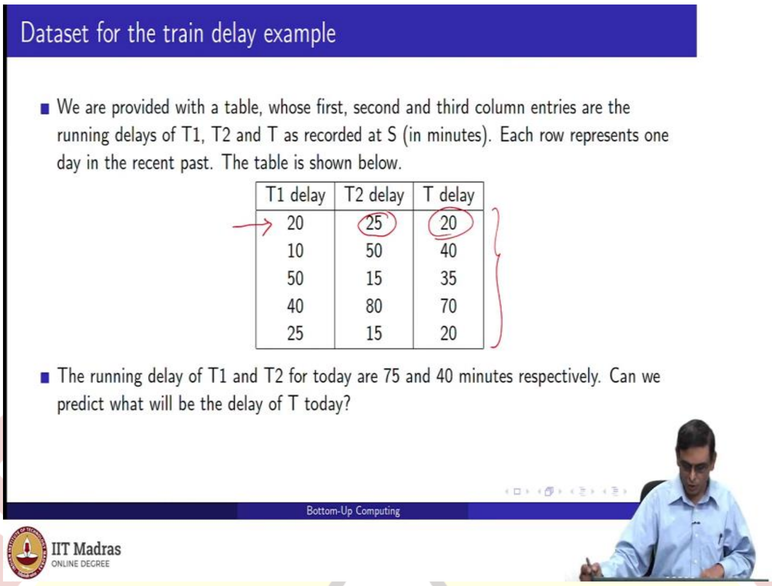
Trains have been delayed for a very specific reason, often the train is delayed because the train prior to it in some station are delayed because the trains share tracks. So, if the train prior, train is delayed, then it is still occupying the tracks because which the train cannot go, so that is why it is late or a station for example a train sitting in a platform and what does the train goes from the platform, you cannot enter the platform, so that is why you may get delayed.

So, because there are shared platforms or shared tracks, therefore the trains get delayed and tracks, track enter a platform the track enter the platform, so some platform. In some sense already models share tracks, so we should only consider shared platform in this later. I will ignore share tracks that was relate, so we consider only share track and what is the problem that we are looking at there are assuming that there are three trains T1, T2 and T.

We have information about T1, T2. We also have information about T. We will see okay. T1 and T2 but what we know is that T1 and T2 arrive at a particular station S. Before T arrives, so T1 comes, when T2 comes or T2 comes when T1 comes T. That is how they come in the station S. Prior to the arrival of T, there is T 1 and T 2, T 1 and T 2 are delayed, there is a chance that T is also delayed. That is basically what the problem.

What we have provided this is a table whose entries are the delays of T1, T2 and T, of examples of T1 T2 and T delays. As recorded in the system S, this is what we have. Each column represents either T1 or T2 or T and each row represents a day.

(Refer Slide Time: 07:55)

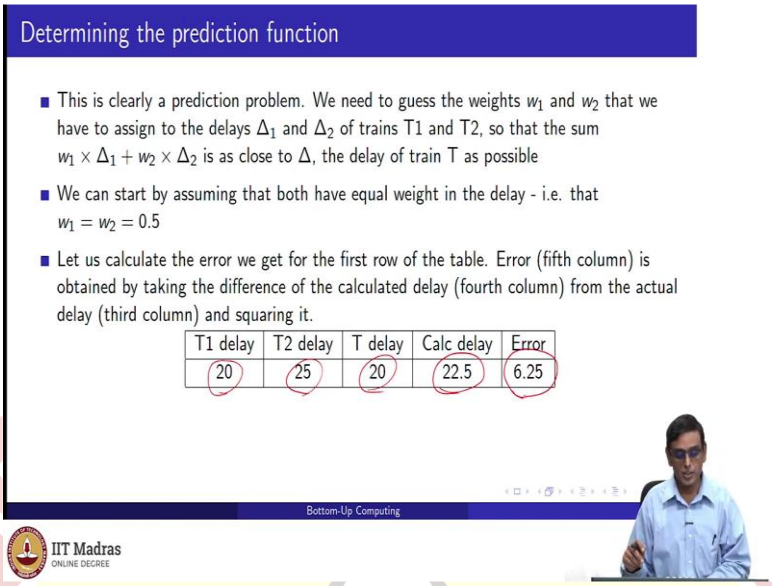


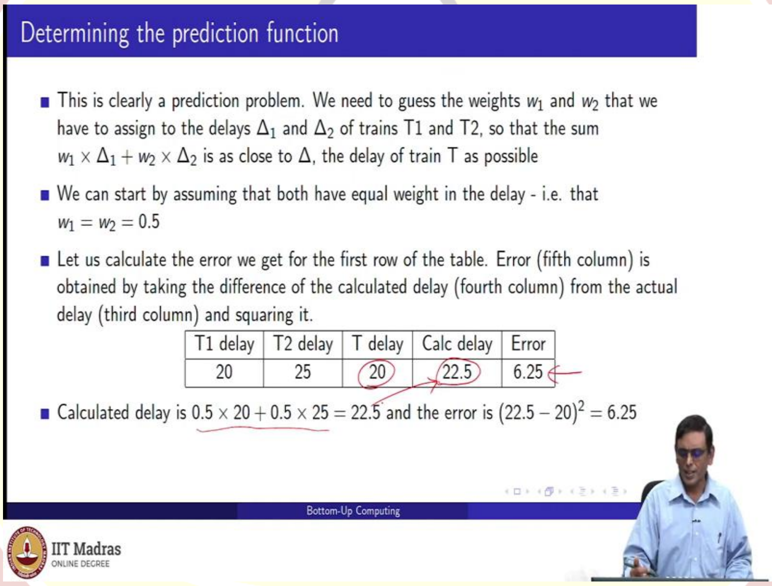
Here is an example of the table, so we have T1 delay, we have to T2 delay and we have T delay, if you see this first row here, it says basically that T1 delay is 20, T2 delay is 25 and T delay is 25. Then early on Tuesday T1 was late by 10 minutes, T2 was late by 50 minutes and T was delayed 40 like that, so those are the row.

Now, we are given from number of days for previous days and what we know for today, today we are waiting for the train. But we know that T1 and T2 already are already late. How many minutes will, we know that T1 is delayed by 75 minutes and you know T2 is delayed by 40 minutes. But T1 and T2 come prior to, before, maybe even one hour before in station S compared to the T.

So, we want to know basically, what is the delay of P going to be today? Can we use this delays of the previous delays, by the information you got, by the information, can we use it to predict the delay of T? So, we know basically that T1 and T2 are delayed by 75 and 40 minutes, we do not know T's delay, but we have the past days. That is said, they are all divided for the part. So, can we do something? Now the usual way in prediction is to try and predict, use try and find numerical function that somehow computes the value of a T delay from T1 delay to T2 delay, that is how we do it.

(Refer Slide Time: 09:42)





So, the simplest way is to do a weighted sum, let us assume there are weight w1 and w2 that we can assign to the delays of train T1 and T2, let us say T1's delay is delta 1, into delay delta 2. We assign weight w1 to T1, w2 to T2. So, the delay of train T is going to be let us say, w and our prediction, our guess of the delay of train T is w1 times delta 1 plus w2 times delta 2, but we have given for five days, we have already, we are given also the value of delta.

So, objective our problem now is to find the w1 and w2 such that w1 plus delta 1 plus w2 delta 2, for each row is as close to the delta as 1 each row. You have to look at all the rows for each row we want to make delta 1, delta 2, as close to delta as possible five rows 72 variables, we know that there are only two rows that we can solve it, two grammatical equation we have solved with.

There are five rows with what you might call an old defined order. So, you have too many values, too many rows, too many numbers. So, what we can do is we can try to make it as close to data as possible for as many rows as possible. We can try that. Let us see, how to do. Now, we have to start somewhere. So, with a good way to start by saying that we do not know anything. So, we can assume basically that both of them are equal. So, we assume that both of them equally. So, they are 0.5 weight each.

So, we can assume that, we will assume that both of them have 0.5 weight each and I will start. Where do you start? Let us start with the first group. Let us start with the first group. So remember, I was given the first row I was given a delay T1 which was 20. I was given the delay of T2 is 25 and a delay of T which is 20.

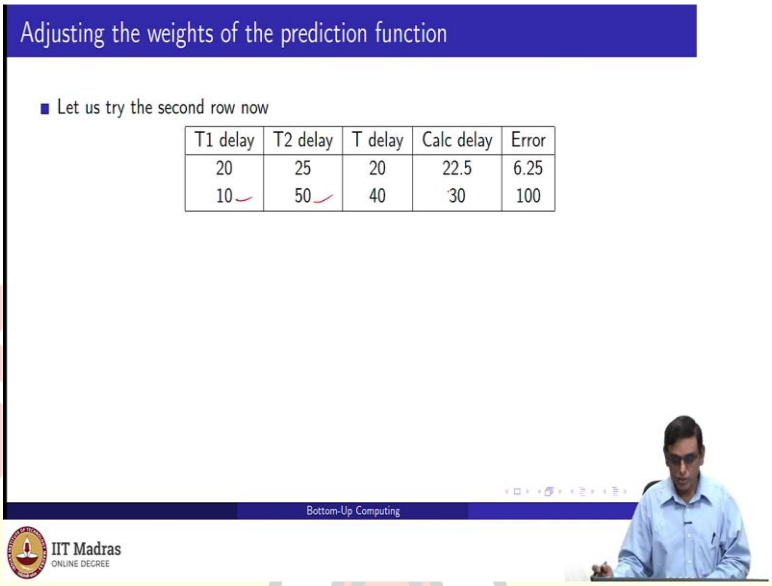
What I have done is, I have computed another delay, this delay it called the calculated delay 22.5 and then I have found out another thing which is called error, squared error function, I have not written square just to save space, but it is a squared error function and that squared error function is 6.25. Let us see how we have calculated that. So, to the calculator delays, number of weights of w1 is 0.5 and w2 is 0.5.

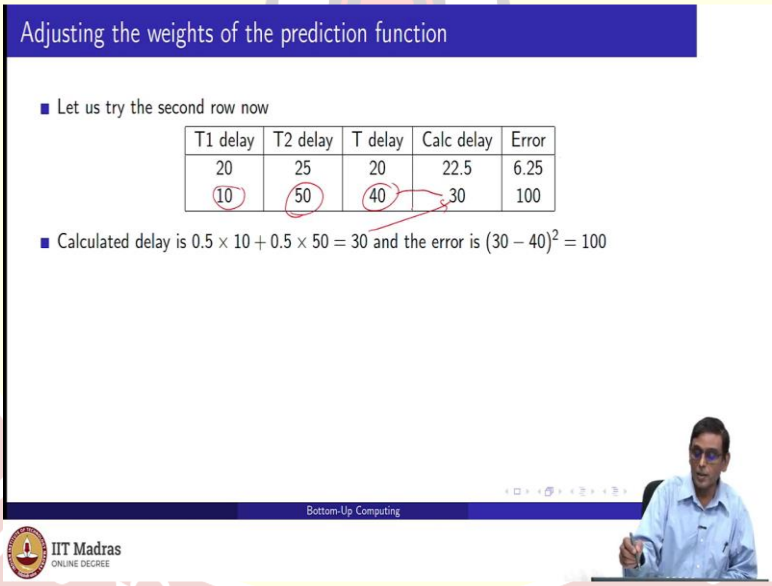
So, the calculated delays apply a weight of 0.5 on 20, apply a weight of 0.5 on 25, so you get 0.5 to 20 plus, 0.5 into 25 and so 0.5 to 20 is then 0.5 to 25 is 12.5, 10 plus 12.5 is 22.5 to the sum of those two give me 22.5 and so, I have of calculated delay that. Now this 22.5, we want to see this error is, the actual delay which you got for the day is 20, but you calculated 22.5. So, the error the difference is 22.5 minus 20 and we have called the square of that error.

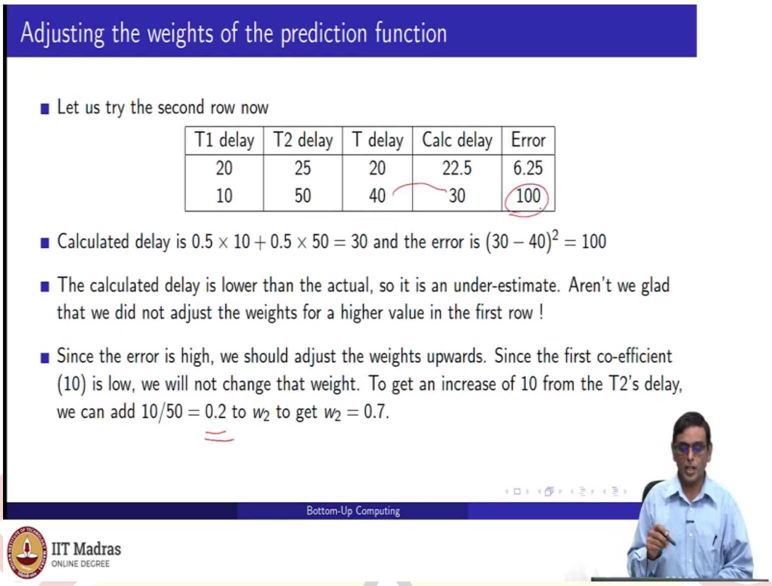
The reason we are making, squaring it is because you want to remove negatives signs and you do not want to worry about negative and positive and squaring gives you the impression of a distance. Remember that x square plus y squared is equal to r square, x square plus y square is the distance you kind of, this way you can calculate for x1 and y coordinates with x1 minus y2 minus on them and then add until you get the distance right. So, there is something like that is what we want to. So, that is why we are taking square. So, 22.5 minus 20 is 2.5, you squared 2.5, you get 6.25, which is why the error is 6.25.

Now, at least we could say that look I calculated 22.5, it at 20, so I must reduce 22.5 to 20, so I must reduce a weight little bit, but the error can be made, I can do that. But since I find the error was is small reduced 6.5 is considered small. So, I say that is my product let us go ahead.

(Refer Slide Time: 13:28)







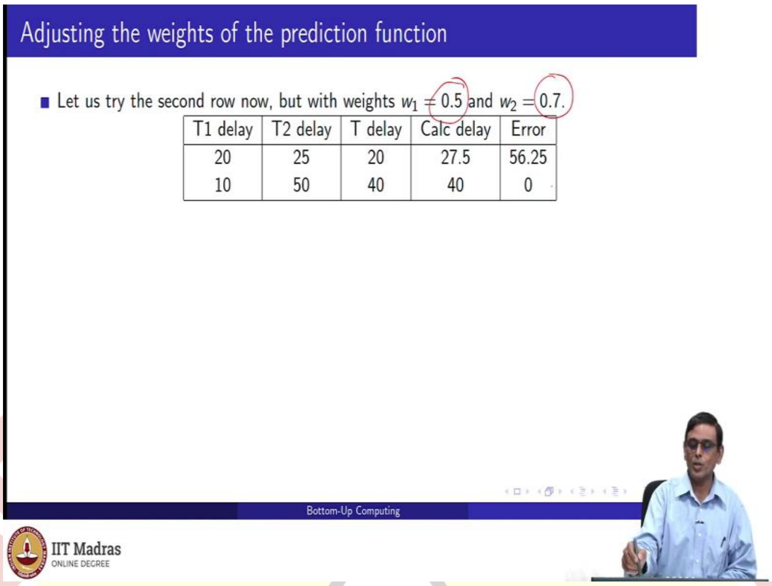
Let us write second row, the second row basically had values and again I must do the following, I must calculate the delay, how do I calculate the delay. The weights were 0.5 and 0.5, so I you can apply 0.5 to 10 I get 5, 0.5 to 50 and I get 25. 25 plus 5 is 30, so that is how you got the calculus delay. But the actual delay is 40. So, the difference is 30 minus 40 is minus 10 and I squared that I get 100. That is a high error under desire.

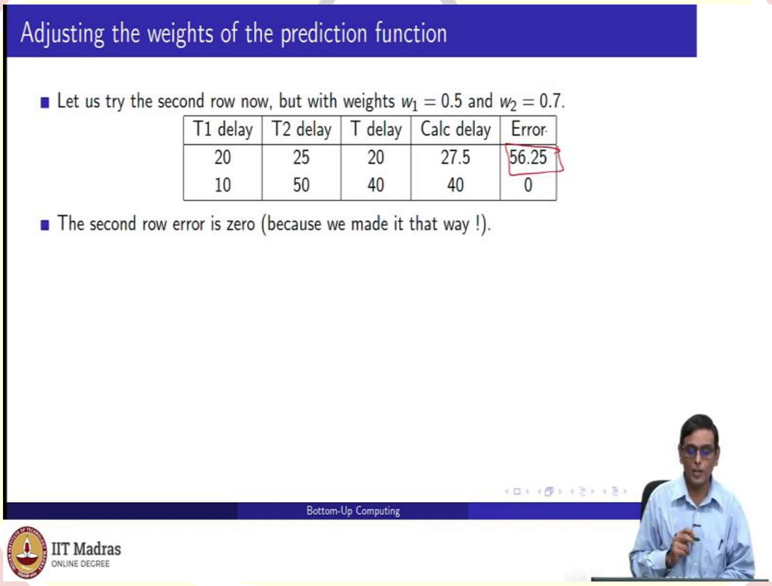
So, I say I cannot tolerate an error. So, I want to reduce where do I reduce, I could try to reduce the weight of T1, but this is number is very small, so it does not make sense. So, what I will do is I will only reduce T2, because that one has a higher right. So, I try and reduce to say T2, so let us say reduce it, how much should I reduce it T2 by, so see the difference between this and this is 10, 40 minus 30 is 10, 40 minus 30 is minus 10.

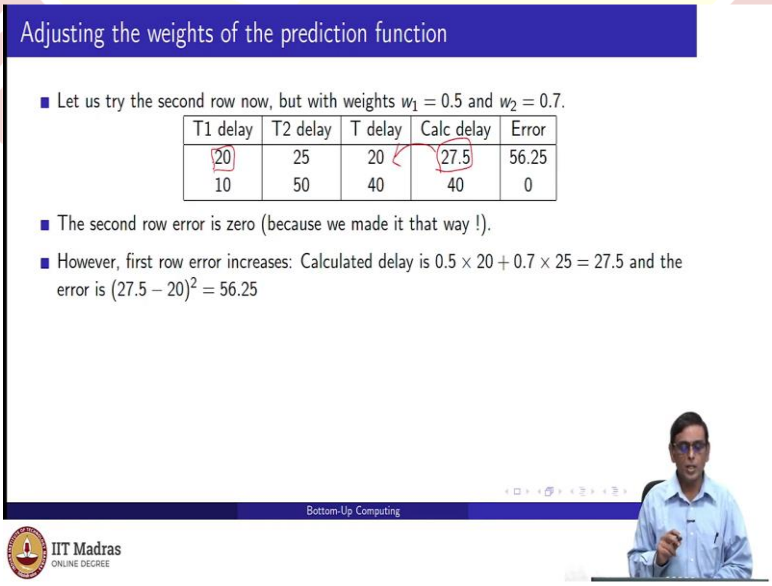
So, I was somehow you know I have basically calculated 30 that will delay 40. So, I have under estimated, so I have under estimated which are under estimated I must somehow increase the calculated delay by a number which is 10. Now I must put a weight on the T2 such that the increases is 10, how much weight should I put nearly the weight I must put?

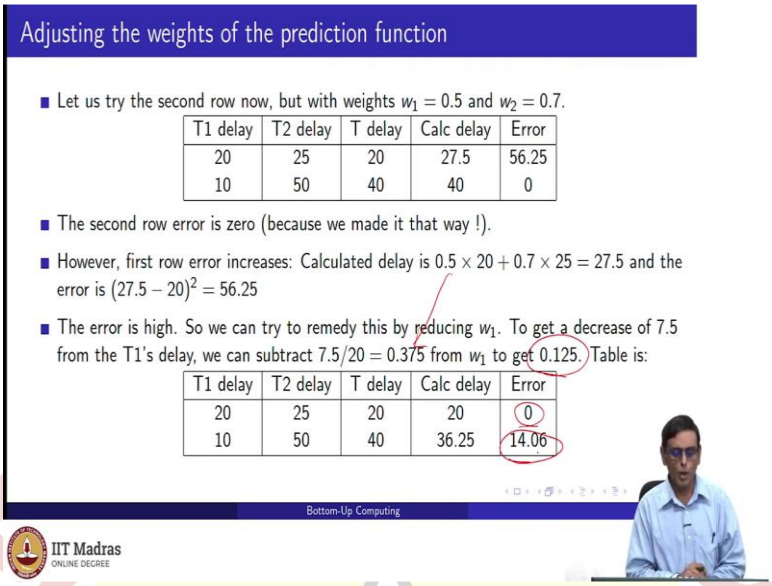
Nearly the weight which should I put nearly the weight I must put is such that 10 by 50. So, I want to get a increase of 10, 40 minus 30 is 10, I must get an increase of 10, the applying the weight on 50, so 10 by 50 is a weight I must increase. So, add 0.2 to the weight w2. That is to make w2 equals 0.7. Then basically these 100 errors vanish. Let us see whether that works.

(Refer Slide Time: 15:10)







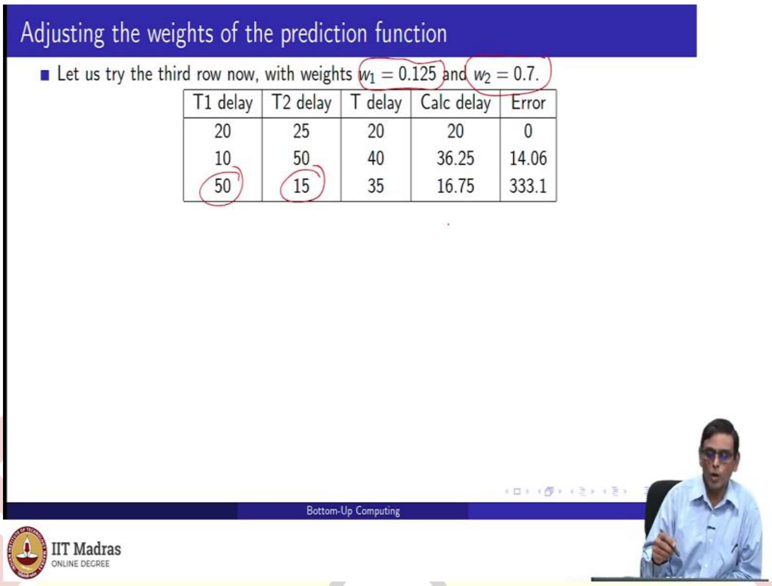


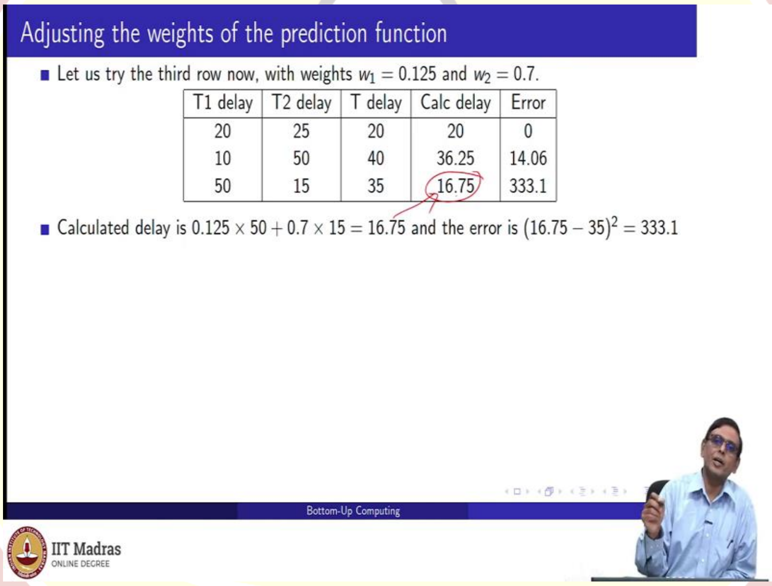
So, what I did basically is I take my weight to 0.5 now and w2 are increased by 0.2 to 0.7 and then if I calculate the second-row error, the error vanishes it is going to 0. Exactly how I wanted to be. But the problem is that the error of the first row now has become 56.2, error of the first row is 56.25. Why did it become 56.25, because the rate of w2 is now 0.7, so when I do the calculation I do 0.7 to 25.5 to 20 is 37.5 and the error is 27.5 minus 56.25. So, the error of the first row has gone up. Now, I could say this error is and just move ahead, I am not happy with 56.25, I want to control this 56.25 little bit.

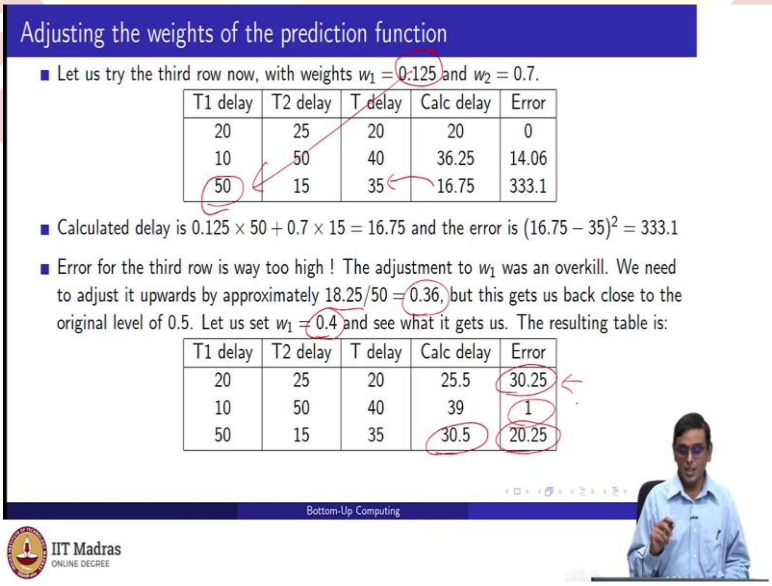
So, this I just increased weight of w2. Let me see if I can tinker with w1 to get the error down. So, let us let us reduce, so I want to reduce, how much should I reduce w1 by to make the error go away? Because we know the calculated delay is higher. Calculated delay is higher by 27.5. So, I want to reduce the calculate delay by 7.5 and I have a number of 20. So, how do I get 7.5 from 20.

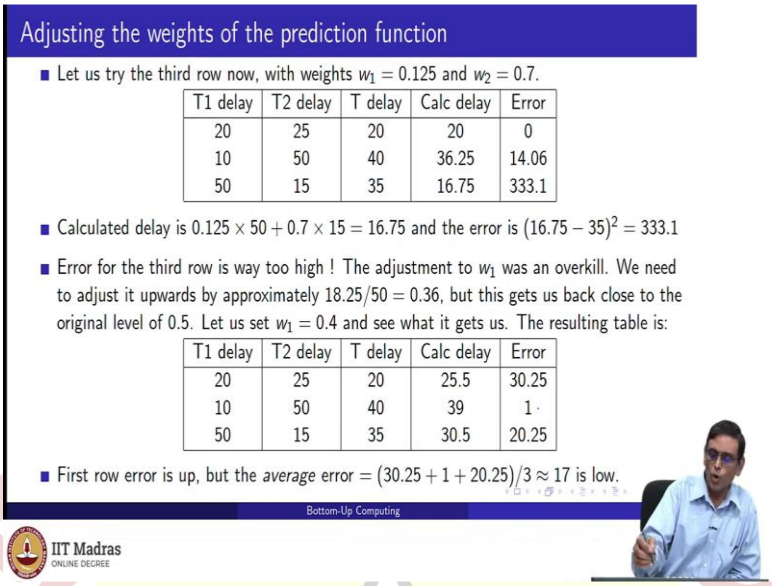
So, 7.5 divided by 20 is quite reasonable, is that we subtract from w1 0.375 then it should go away. So if I subtract from earlier value of w1 or 1.5, I am reducing 0.375 from it I will get 0.125. So, I can now compute the nearly because that is what I did. Error of first row has become 0. But now the second raw has become, wait now second row is become 14.06. 14.06 is a small error, I think it is a small error, so we ignore it and we go ahead, let us look at row 3 now.

(Refer Slide Time: 16:56)









What are the weights now, weights are w1 has been adjusted to 0.125 and w2 has been adjusted to 0.5 these are the new weights. Using these weights? I take row three. Row three, the delay is given us up 50 and 50. I can calculate how do I calculate? 50 into 0.125 plus 0.7 into 15. That gives me 16.75.

That gives me 16.75, there are 35 minus 16.75 or correctly if you do 16.75 minus 35 and then square it, you will get 333. Huge, huge, again huge. So, we do not like, you do not like this? Clearly, when we did the increase, we increase the rate, we did some adjustment from w1, you know whatever we did, basically, we reduced it, we reduce the weight of w1. Now this guy has got a huge w1 50 and we made the weight of w1, 0.125 it is small weight.

So, that is why this thing has happened. So in some sense, we want to go back we are saying no, no, this was an over case. We did too much adjustment. Let us see whether I can afford adjusting, how much should I afford adjusted. Now the difference between 16.7 and 35 is 18.25 and if I divide it 50, I get 0.2. So, I should have I just upward by 0.36.

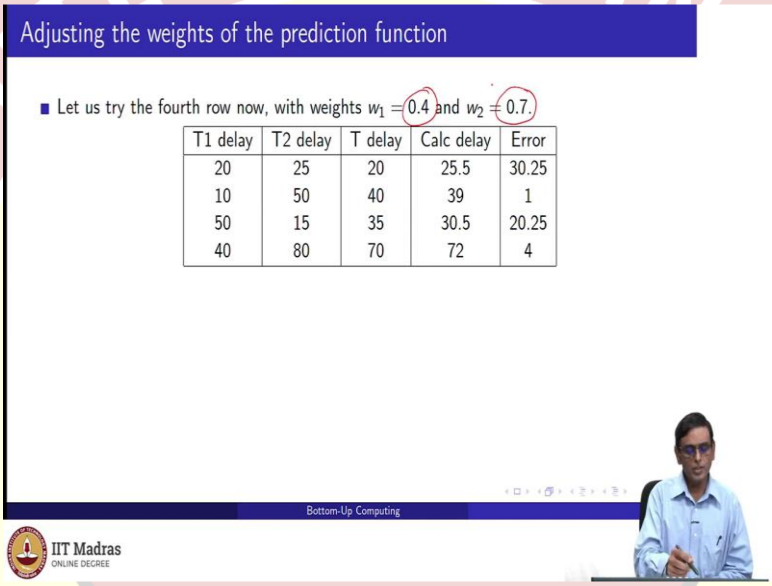
But if I add 0.36 to 0.125 I get 0.485 which is close to 0.5, where I started, pointless to go back and I will just go back to point four and I will start again. I do not want to go back and start again. So, I say I will not take it all the way back 2.5. I started with 0.5, I came down to 0.125, I do not want to go back to 0.5 again, let me go somewhere in the middle.

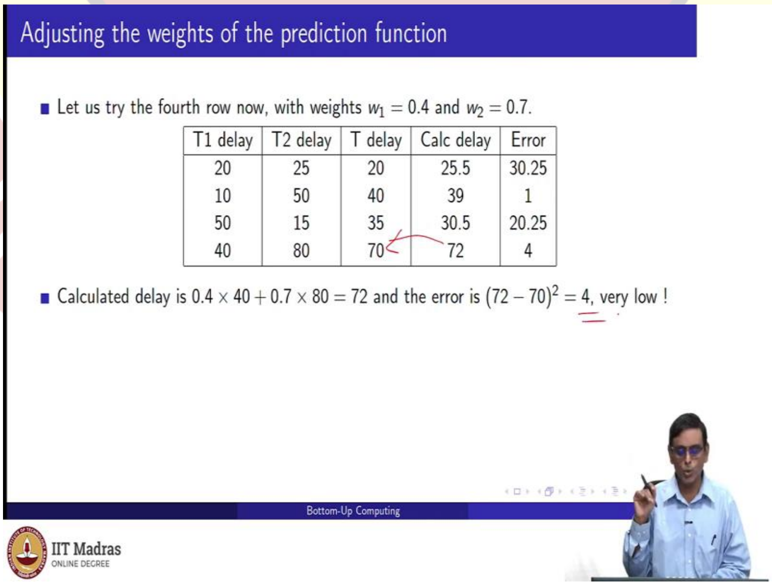
How in the middle. We must also reduce the delay. So, let us see a good 0.4. This is a scientific way of getting 0.4. But anyways, I am getting as close to 0.5 as possible, but not yet 0.5. That is the idea. So, let us say I put it at 0.4 and see what I get. So, with the delay 0.4 and this thing we know find basically the delay, calculate 30.5 and the error becomes the 30.5.

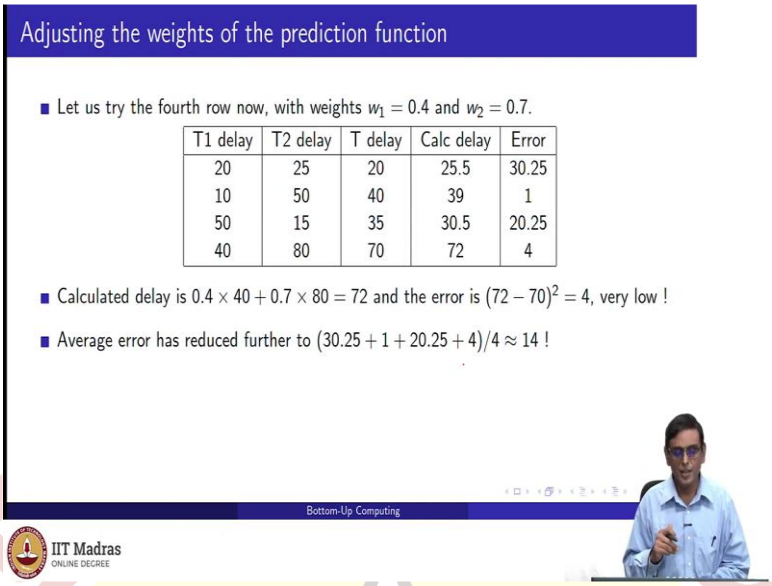
But now these things also have an error, we can calculate there are these two lines. Fortunately, the is not high, this guy is 30, it is high but it is not that high.

In fact, actually, rather than the error now I want to compute the something else. I want to compute the average error, what is average error? At hear 30.25 1 and 20.25 you can add up all three and divide it by 3. There is average. 30 plus 20. 50 plus 1 is 51 divided by 370, average is something 70 that is pretty okay. So, I am done with row 3 now. Now I want to go to the row 4.

(Refer Slide Time: 19:35)

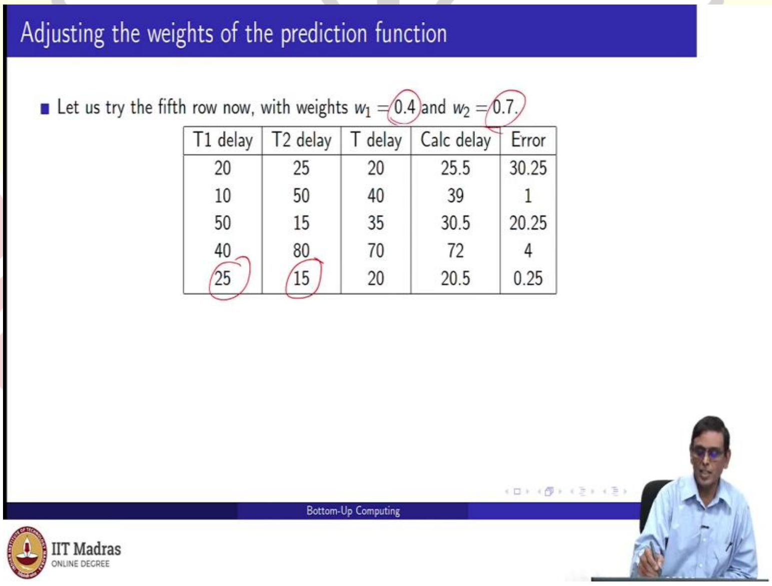






So, row 4 weights now what are the weights? Everyone is 0.4, w2 is 0.7 and I am now basically computing row four. So, let us do row 4 0.4 into 40 plus 0.7 into 80 is 72. 72 is damn close to 70. So, the error is just 2 square is 4, very, very low, no problem. So, I am just going to and in fact, actually not only has it done that, the average error has down even more. Because if I take the average of these four rows now, 30 plus 1 plus 20 by 4 is just 14. So, it is become 14, average error has become 14 I am very happy I now go to, row 5.

(Refer Slide Time: 20:13)



Adjusting the weights of the prediction function

- Let us try the fifth row now, with weights $w_1 = 0.4$ and $w_2 = 0.7$.

T1 delay	T2 delay	T delay	Calc delay	Error
20	25	20	25.5	30.25
10	50	40	39	1
50	15	35	30.5	20.25
40	80	70	72	4
25	15	20	20.5	0.25

- Calculated delay is $0.4 \times 25 + 0.7 \times 15 = 20.5$ and the error is $(20.5 - 20)^2 = 0.25$, very low !

Adjusting the weights of the prediction function

- Let us try the fifth row now, with weights $w_1 = 0.4$ and $w_2 = 0.7$.

T1 delay	T2 delay	T delay	Calc delay	Error
20	25	20	25.5	30.25
10	50	40	39	1
50	15	35	30.5	20.25
40	80	70	72	4
25	15	20	20.5	0.25

- Calculated delay is $0.4 \times 25 + 0.7 \times 15 = 20.5$ and the error is $(20.5 - 20)^2 = 0.25$, very low !

- Average error has reduced further to $(30.25 + 1 + 20.25 + 4 + 0.25)/5 \approx 11$!

Adjusting the weights of the prediction function

- Let us try the fifth row now, with weights $w_1 = 0.4$ and $w_2 = 0.7$..

T1 delay	T2 delay	T delay	Calc delay	Error
20	25	20	25.5	30.25
10	50	40	39	1
50	15	35	30.5	20.25
40	80	70	72	4
25	15	20	20.5	0.25

- Calculated delay is $0.4 \times 25 + 0.7 \times 15 = 20.5$ and the error is $(20.5 - 20)^2 = 0.25$, very low !

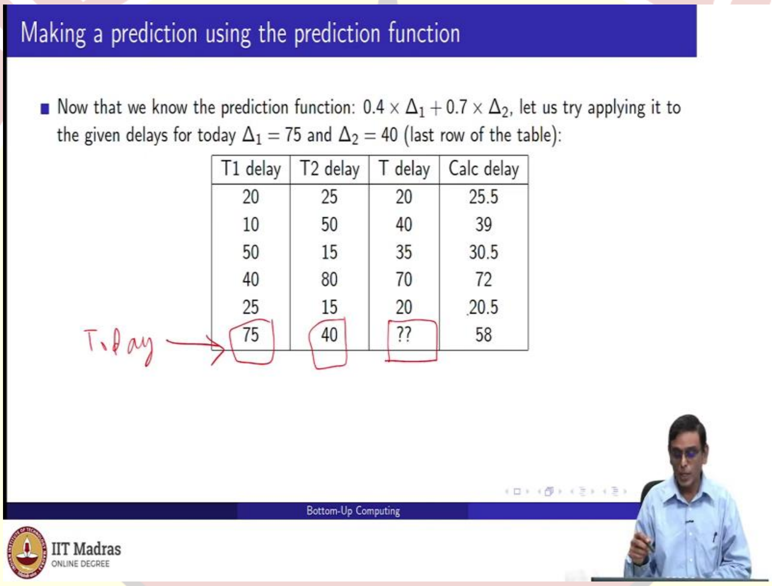
- Average error has reduced further to $(30.25 + 1 + 20.25 + 4 + 0.25)/5 \approx 11$!

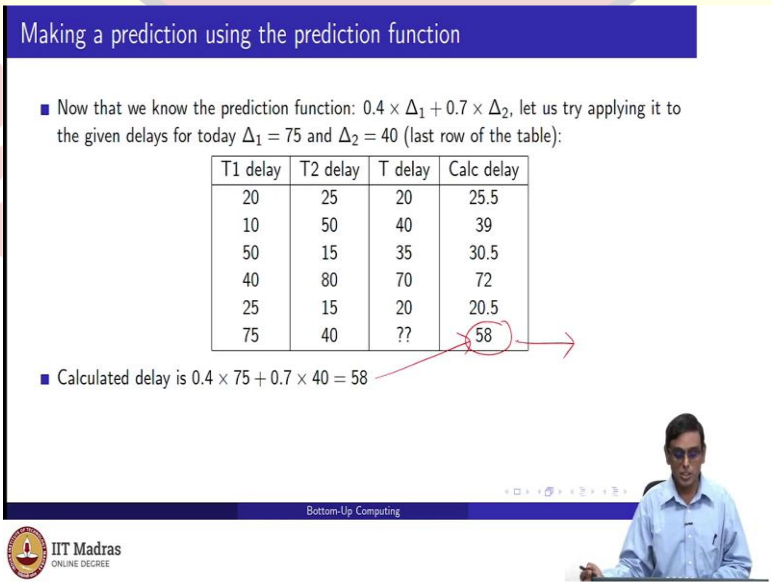
- We are done with all the data values now, and the prediction function $0.4 \times \Delta_1 + 0.7 \times \Delta_2$ seems to be a reasonably good fit to the given data.

And I go to row 5 I basically again, I have given these delays 25 and 15 and now I want to calculate it with the same base, 0.4 and 0.7, I find that the calculated delay is 20.5. Lucky, I got lucky, right 20.5 minus 20 is just 0.5, square it you get 0.25 and not only is this value low, it actually brought the average even further down. It is now at 11.

I am very, very happy with these weights, what are the weights, weights are 0.4 and 0.1. So, now I can basically that I have a reasonably good confidence that the weights w 0.4 and 0.7 work, because it giving me low error, low average error of it.So, I say basically that I am going to predict that the function prediction function is 0.4 into delta 1 plus 0.72 into delta 2.

(Refer Slide Time: 21:04)

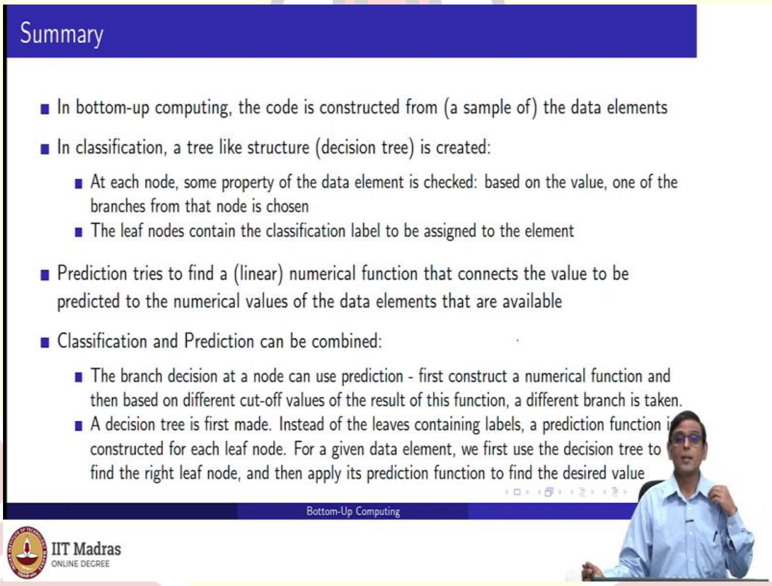




Now, what do I need to do? I have given this new data set, which is 75 minutes of delay T1 and 40 minutes of data T2, which is the data for today, it is today's data. You got 75 minutes of delay 41 and 40 minutes of delay 42 and we are asked to find the delay 40. We do not know that delay for people. Because it is not going to happen, is yet to come. But I can calculate the delay using this prediction function. So, I do that, and what do I get 0.4 and 75 plus 0.7 into is equal to 58. So, what I am going to do is, I am going to predict, I am going to say delay of train T today.

You can say you it does not, if T1 delayed 75 minutes, how can it be less than that, can be because even in the past, even a T2 came to the station S, before T could even be one or two hours before T. So, that is in the past that is why the data available to you, you got those delays, and then from those delays, you are trying to compute the delay and it could be also station long before the station which are waiting, some station S, not the station which are waiting maybe. What are this, you know, basically the calculated delay of the train, T when at least you know, can be 58 minutes by due to this method.

(Refer Slide Time: 22:30)



So, bottom-up computing as a summary. What is it? In a classification, what we try to do basically, is that what we are seeing, we did not see it in this lecture. But we saw in some of the previous lectures and as we, we try to make a tree like structure, the decision tree which no other decision tree, basically, we try to figure out based on the data set, data element that we have branch to be.

So, one of the branches chosen based on node and when you go down the bottom of the tree you will be finally at the leaf sitting a label which is assigned to the element. You say, you go down this branch and come to the tree you pick the letter A, if come down like this we pick letter P. You can do character recognition like that.

Or if you come like this, we have that person from the person's name here, another person's similar to recognize the person by face or whatever. The leaf node contains a classification. Prediction on the other hand what we are trying to do is basically to make a numerical function that takes certain values and returns a value, so it basically is a linear numerical function may not be linear but usually, most of the things we do not non linear, this non-linear stuff is very hard to do.

The linear make up is a connects the value of the predicted value to this numerical value to the known values. Come to the known values, we are trying to find a prediction value. Obviously, one can combine classification prediction. One way of combining classification prediction is that the branch in the classification tree can be based on a prediction.

It means you can first compute a unknown value from a known value and use the cut off on the unknown value to take branch, you can first compute the known value and then say unknown value between 0 and 1, 0 and 10, gold first rank between 10 and 20 take a second branch, 20 and 30 take the third branch you can do that...

So, prediction can be used as one step in the classification, it is one way it can be combined. The other ways, you can first make a decision tree and in the deep instead of using a label grab a prediction function to do that, you can have a tree, decision tree you go down the decision tree using the data and I believe you use a prediction function to actually calculate.

So, that way you can combine both prediction and classification and you get much more interesting kind of. So, with that, we come to the end of all the content of this course. The next lecture, what you will see basically is a summary of the concepts that we introduced in this course, that will be the last classification that you will have to watch. Thank you.