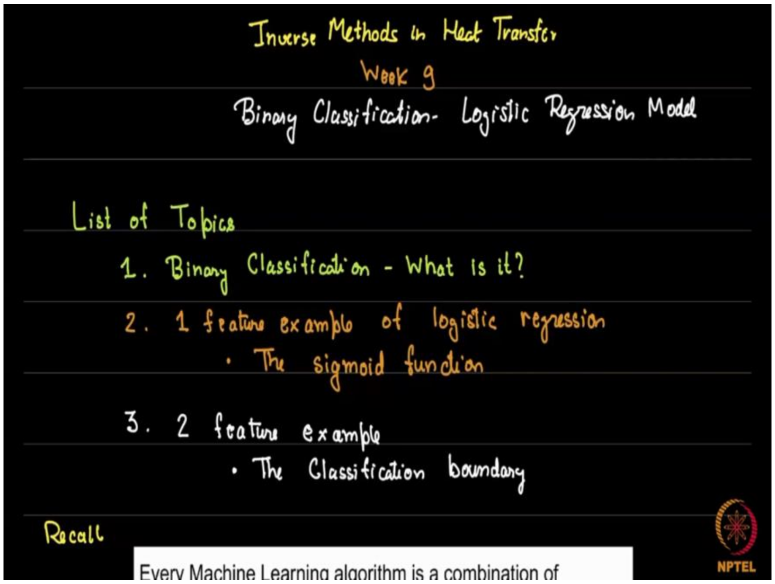
Inverse Methods in Heat Transfer
Prof. Balaji Srinivasan
Department of Mechanical Engineering
Indian Institute of Technology, Madras

Lecture - 48
Logistic Regression the Forward Model

(Refer Slide Time: 00:19)



Inverse Methods in Heat Transfer
Week 9
Binary Classification- Logistic Regression Model

List of Topics

1. Binary Classification - What is it?
2. 1 feature example of logistic regression
    - The sigmoid function
3. 2 feature example
    - The Classification boundary

Recall

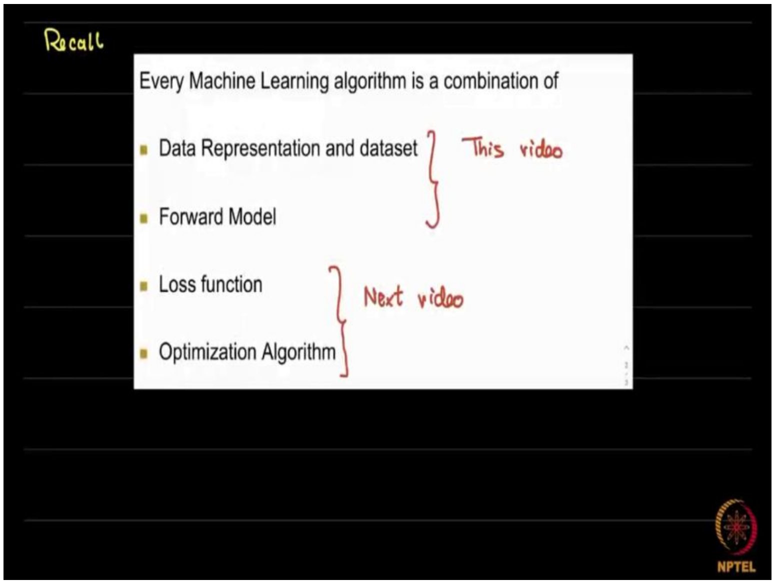
Every Machine Learning algorithm is a combination of

Welcome back, this is week nine of inverse methods and heat transfer. In this video we will be looking at our first classification algorithm this is called logistic regression model. This will be done in two parts, this video and the next video. The task that we will be trying to do is what is known as binary classification. So, the list of topics that I will be covering within this video is, what is binary classification, why is it important in practice, where could we see something on that shot and let us say heat transfer scenarios.

And then I will give you a couple of examples to set up what we will be doing in the next video. The first is a one feature example, remember when we had linear regression, we called whatever went into the input as a feature. So, for example you could have x, x^2etcetera as features of x_1, x_2 as features when we looked at linear models. Similarly, we will look at a simple just to start with what is a one feature example of logistic regression.

And there you will see the sigmoid function that I introduced you to in the last video arises naturally. Next, we will move on to a two-feature example and we will talk about something called the classification boundary, which is an important concept within a binary classification.

(Refer Slide Time: 01:39)



Recall

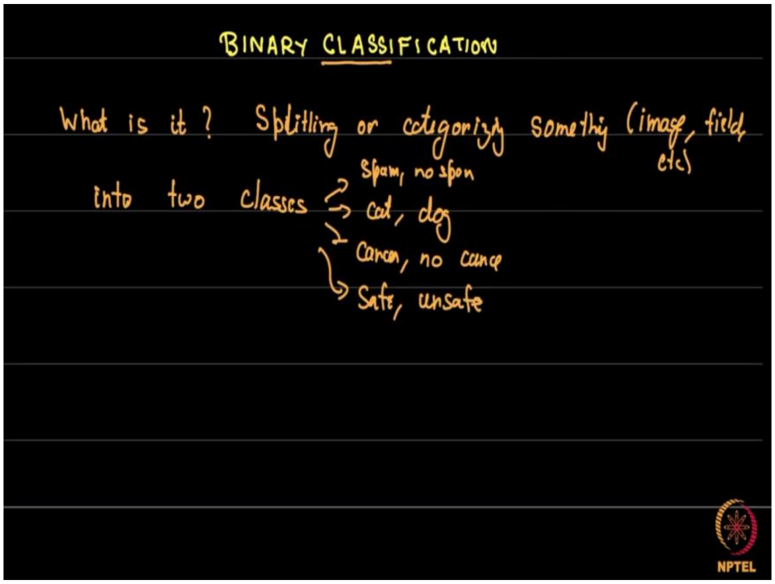
Every Machine Learning algorithm is a combination of

- Data Representation and dataset
- Forward Model
- Loss function
- Optimization Algorithm

Now recollect that every machine learning algorithm that we will be discussing is basically a combination of four things. How we represent the data, what is the forward model, what is the loss function and what is the optimization algorithm used and these two is what we will be covering in this video. So, we will simply be covering the how we represent cases data in case of classification problems and what the forward model is.

And the loss function and the optimization algorithm, we will be covering in the next video and then I will summarize the net situation. We will do the same thing for our categorical or multi-class case as well. So, let us now step into the data representation and data set portion of binary classification and talk about what it actually is.

(Refer Slide Time: 02:32)



BINARY CLASSIFICATION

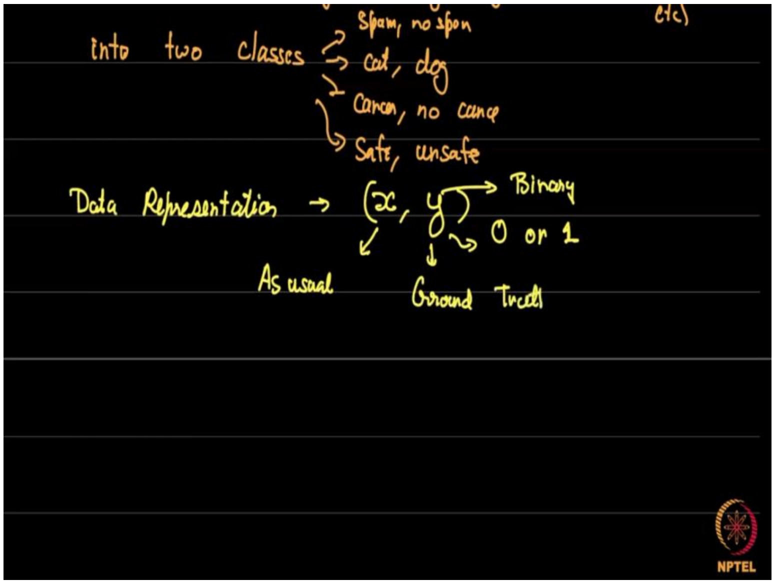
What is it? Splitting or categorizing something (image, field etc)

into two classes → Spam, no spam
→ cat, dog
→ Cancer, no cancer
→ Safe, unsafe

So, what is a binary classification? So, binary classification is simply splitting or categorizing or classifying of course something whatever it is, it could be an image, it could be a field into two possibilities or into two what we call them classes, which is why it is called classification. So, for example of course the standard example I have been giving throughout, we when you have an email you can say spam or not a span.

Or if you have an image, you could say cat or dog, you could say cancer, no cancer or thermally safe or thermally unsafe, so stuff like this. So, we have just two classes in which we wish to split it.

(Refer Slide Time: 03:45)



Spam, no spam etc)

into two classes → cat, dog
→ Cancer, no cancer
→ Safe, unsafe

Data Representation → (x, y)

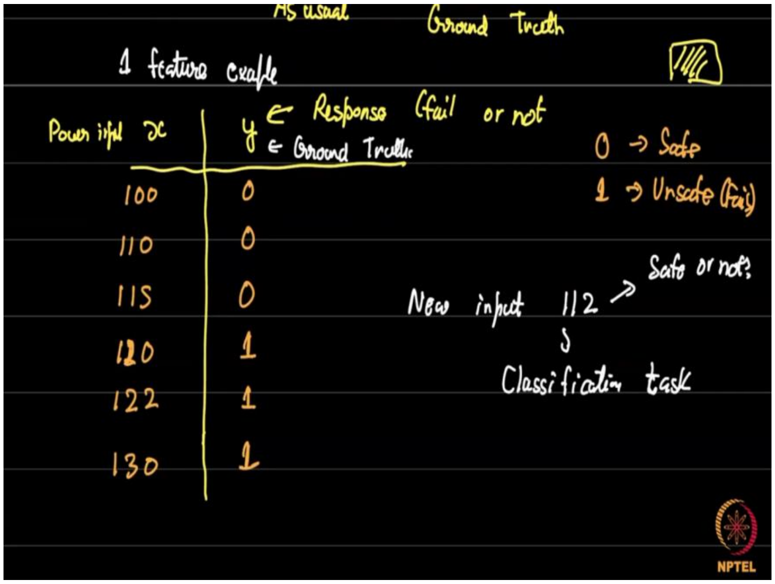
x → As usual

y → Binary
→ 0 or 1
→ Ground Truth

Now how do we represent data in such a case? Now you do nothing special for the input. So, for example you have an image just like we discussed last week, you represent it by pixels and unroll it etcetera. If you have an email of course there are many special techniques to work with text or if you have simply a flow field or a temperature field you represent it with numbers as usual. So, this for data representation is both for x and for y.

So, this is as usual whatever the input is it stays as usual. Now why on the other hand you can as it turns out you can do it in two different ways. So, we are going to take the first very simple sort of method to use this we will say 0 or 1. So, y is basically a binary remember y is the ground truth.

(Refer Slide Time: 04:49)



As usual Ground Truth

1 feature case

Power input x	y ← Response (fail or not ← Ground Truth
100	0
110	0
115	0
120	1
122	1
130	1

0 → Safe

1 → Unsafe (fail)

New input 112 → Safe or not?

Classification task

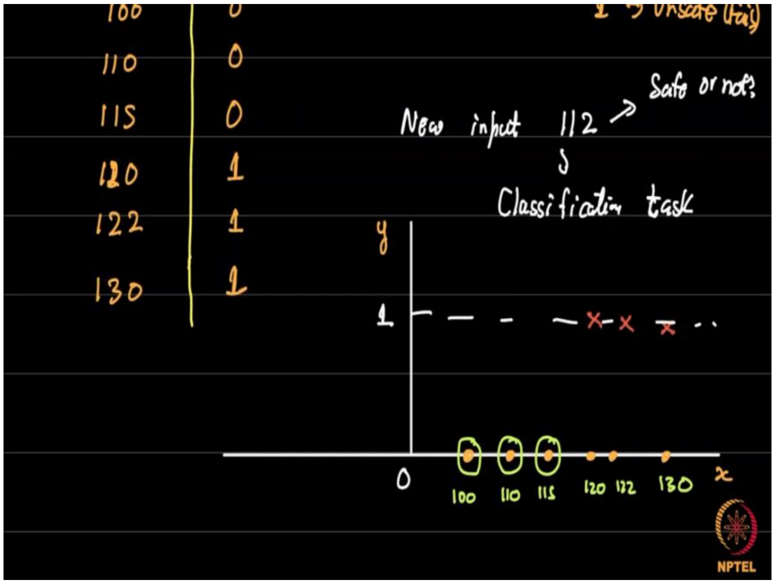
So, I could have a case like this. Let us say we are looking at the design of a thermal chip and what you have as x is let us say power input. So, let us say you have a chip and you have a hole on your mobile and a whole bunch of apps are running and as the number of apps increases, obviously the chip gets heated up, because the power input is high and here is the response, that is does this thermal chip just this chip fail or not.

So, let us say we have a lot of measurements that we make as we are going about it while doing testing. So, during testing of this chip you collect a data set, so let us say I am just making some numbers up. So, 100 Watts or something is the power which is input this is safe 110, this is safe, so let us say I have decided 0 means safe and 1 means unsafe or fails, if something happens something burns up.

Similarly, 115 it is safe and 120 it starts failing and 122 it fails again and 130 it fails. So, let us say we collected this data. Now what you want of course is when you give a new power input, it something like 112, is it safe or not. So, this would be a classification task. you can think of it similar to let us say measuring sizes of tumour, this is of course very simple and very simplified more accurately.

Obviously, no real case will be just this one non-one this is what I said as a one feature example. usually you will have multiple features, I will give you an example of this later on in this video. So, this is a one feature example. Now y is ground truth what you saw.

(Refer Slide Time: 07:13)



100	0
110	0
115	0
120	1
122	1
130	1

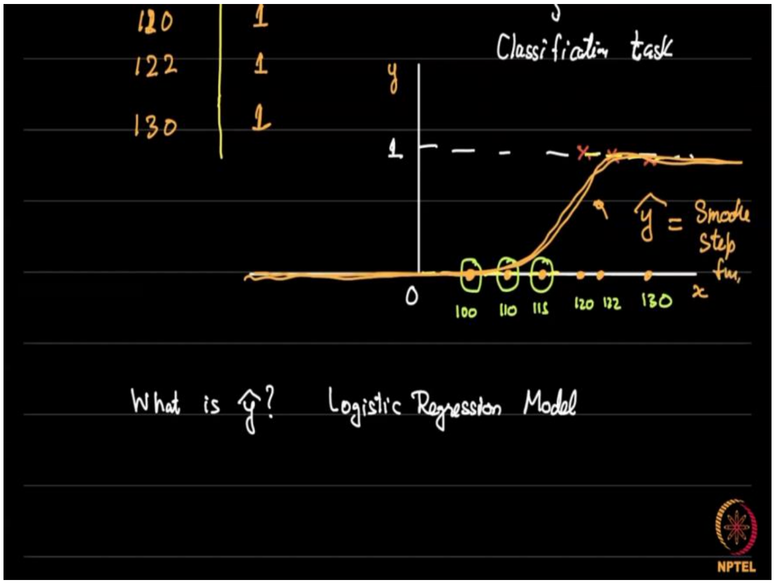
1 → unsafe (fail)

New input 112 → Safe or not?

Classification task

So, if we draw this in a graph just like we were doing the linear regression cases. So, let us say I plot these points and I have 100, 110, 115, 120, these are not really evenly based 122 and 130, something of that sort. I could say that y as a plot looks like this, that at these points it was safe, so let me draw this with a green circle when it is safe, so safe, safe so this was 100, 110, 115 and this is 120, 122, 130 and this was unsafe here so, I will draw this with a red cross, so this is 0, 0, 0 and at this point I have 1.

(Refer Slide Time: 08:21)

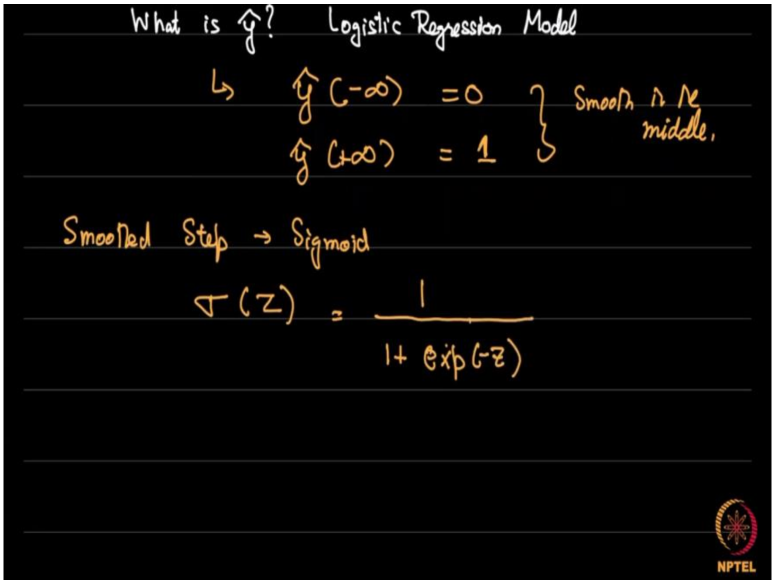


Now our modelling task is to find out $\hat{y}$ which will fit this data. In that sense this is like a regression task which is why it is called logistic regression model. You will see this in some detail just shortly. How should I fit so the question really is how should I fit some curve here which will fit these data points? So, now the simplest solution, so let me just draw a solution for example is something on this side.

So, this could be a $\hat{y}$ and indeed in the earlier days of machine learning this was the $\hat{y}$ that was chosen a step function. This turns out due to multiple reasons especially for optimization this is not ideal. So, we will actually not use this model, we will not use a step function model, because it is non-ideal for what we wish to do. Now you can think of a few other models, but I am not confusing you with those at this point, but you can think of a natural model.

A natural model could be a smooth function that goes from 0 here to 1 here. So, this could be $\hat{y}$ and this $\hat{y}$ is a smoother step function.

(Refer Slide Time: 10:06)



What is $\hat{y}$? Logistic Regression Model

↳ $\hat{y}(-\infty) = 0$ } Smooth in the middle.

$\hat{y}(+\infty) = 1$

Smoothed Step → Sigmoid

$$\sigma(z) = \frac{1}{1+ \exp(-z)}$$

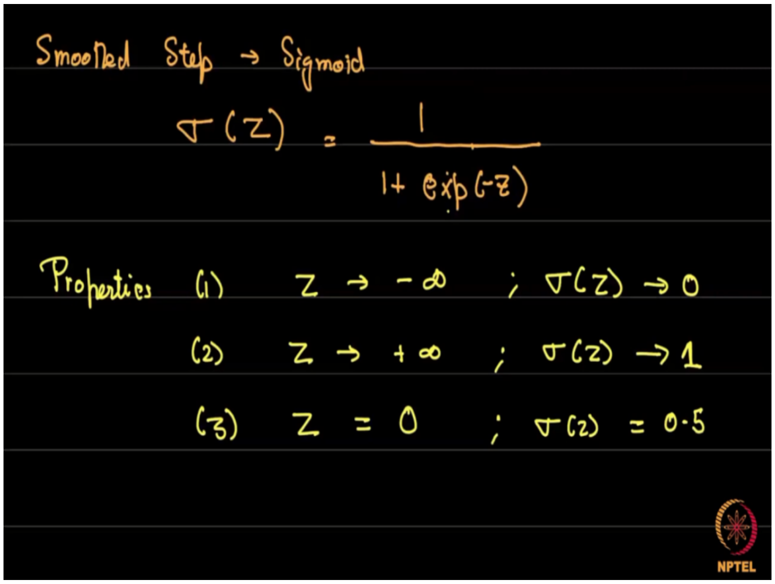
So, y hat has the following properties, $\hat{y}(-\infty) = 0$. I am obviously writing it with some mathematical abuse here, I will make it a little bit more precise of $\hat{y}(+\infty)$ will reach 1 and it is smooth in the middle. So, this is the general idea of a function, obviously we need it to switch somewhere in between 115 and 120. So, that is where we want to see this rise and I will tell you how to do this in two steps.

So, the idea is we create a new function which is a smooth step and that is called the sigmoid. And the sigmoid is written as sigmoid of some variable Z is,

$$\sigma(z) = \frac{1}{1 + \exp{(-z)}}$$

So, this is called the sigmoid function.

(Refer Slide Time: 11:20)



Smoothed Step → Sigmoid

$$\sigma(Z) = \frac{1}{1+\exp(-z)}$$

Properties (1) $Z \to -\infty$; $\sigma(Z) \to 0$

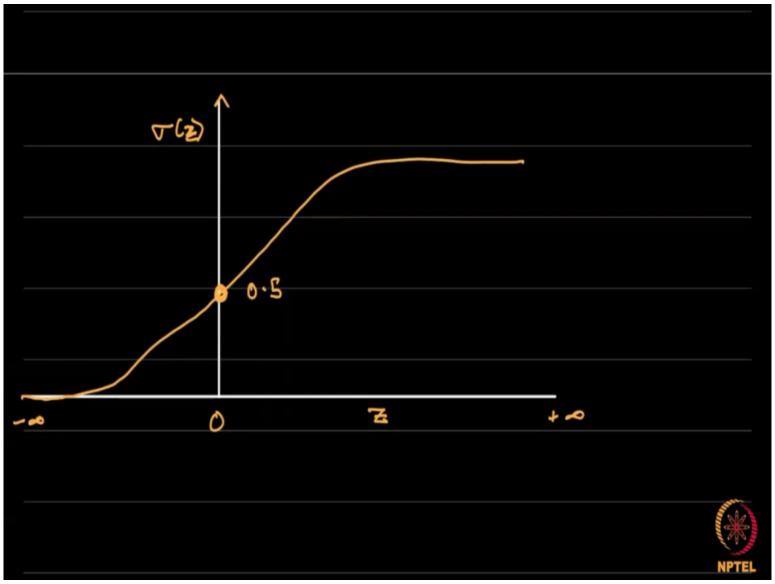
(2) $Z \to +\infty$; $\sigma(Z) \to 1$

(3) $Z = 0$; $\sigma(z) = 0.5$

It has the following properties first property is as Z tends to $-\infty$, $\sigma(z)$ you can see is $e^{+\infty}$, so which is effectively tends to ∞, so $\frac{1}{\infty}$ this tends to 0. Second property as Z tends to $+\infty$, $\sigma(z)$ becomes exponential minus infinity which is approximately extending to 0, so $\frac{1}{1}$ this tends to 1. So, instead of writing equal to tends to 1.

Third is when Z = 0, Z is exactly equal to 0 this basically becomes you can check e^0 is 1, 1 by 1 + 1 so this becomes $\sigma(z)$ is 0.5.

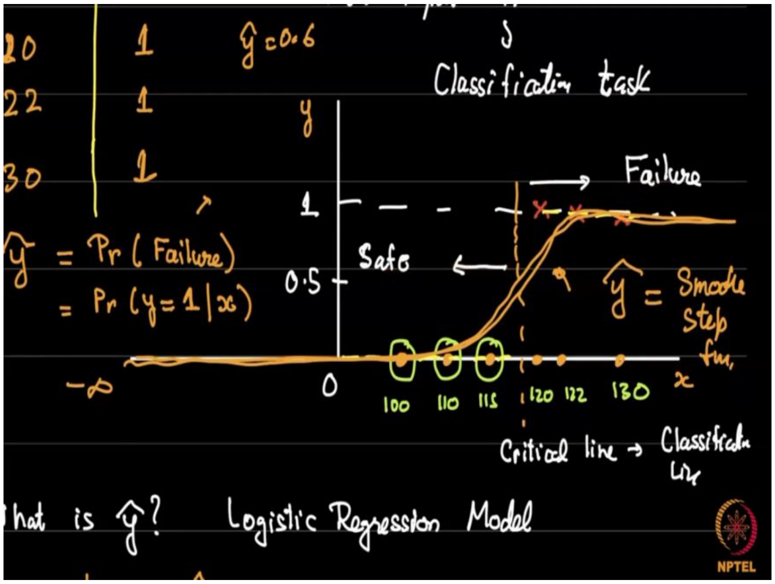
(Refer Slide Time: 12:23)



So, let us draw that now. So, sigmoid of z looks something of this sort as z tends to - infinity, it tends to 0, so 10 into $-\infty$, 10 into $+\infty$. Let us say this is z and this is $\sigma(z)$ somewhere in the middle it will actually hit at 0 this will hit 0.5. Now before I go back to our original problem and explain this a little bit further, we have z here whereas x there you know how do we translate between x and z.

You can immediately notice a few things that the sigmoid becomes flat as z tends to plus infinity and sigmoid tends to flat when it becomes minus infinity. The slope here we will calculate that it is not as bad as what I have made it to look, but it has infinite number of derivatives so we can easily take derivatives.

(Refer Slide Time: 13:41)



Now how do we use this here for our model, so I am going to mix two things, one is how we are going to actually interpret this and second thing is how we are going to actually implement this. So, how we are going to interpret this is as follows. we will say that my y hat need not be exactly 0 or 1 but it could take values in the middle. Now notice this, our original data set was y because I am now measuring what actually happened in reality.

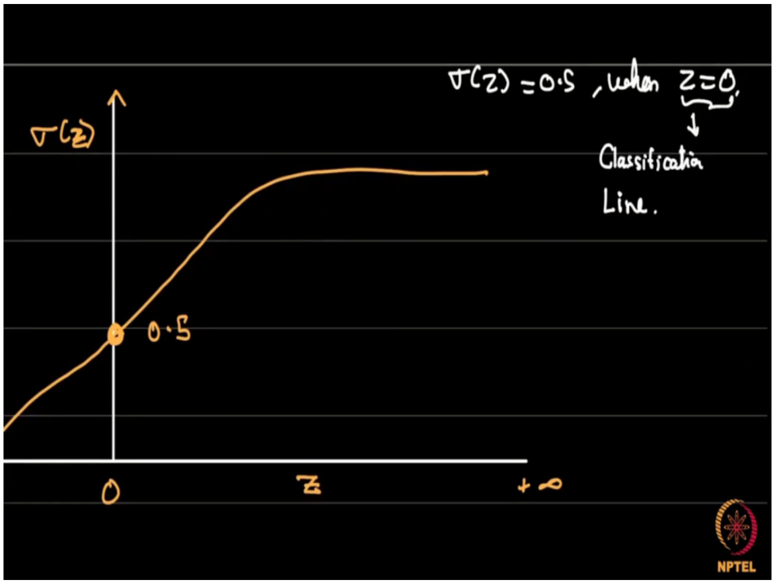
What actually happened in reality was this chip either did not say either did not fail or it failed. Whereas when I am making a guess for what will happen for a new chip that is given to me, I can only assign probabilities. What I can say is if the power is really small, obviously my probability of failure becomes smaller and smaller it gets closer and closer to 0. Similarly, if the power is really high, you are dumping a lot of power on it then the probability of failure becomes higher and higher.

So, the first thing we are going to do as far as interpretation is concerned is instead of saying y hat represents failure or success, y hat represents probability of failure. So, when probability of failure is really high it means y hat gets closer to one, when probability of failure is really low y hat gets close to 0, another way of saying it is $\hat{y}$ is probability is that y = 1 for a given check power input.

So, do not worry too much about the notation here, I will not dump you with too much complex notation, but at the very least I hope you can agree the probability of failure is the same as saying probability that y = 1. Now the next thing is this a key point that happens where the situation say flips, so let us say the probability of failure y hat turns out to be 0.6, you say well probability of failure is 0.6, then people will assume that it is more than 50 chance, so it has failed.

So, after the step of giving a probability if you still have to make a choice about whether it is 0 or 1, right of this line, where $\hat{y}$ is exactly 0.5, you will assume that it is more probable to fail and left of this line you will assume that this is safe. So, this line here this critical line is called the classification line. one side of the line you have one outcome, other side of the line you have other outcome, at least in terms of how you predict it. So, how do we find this line?

(Refer Slide Time: 16:50)

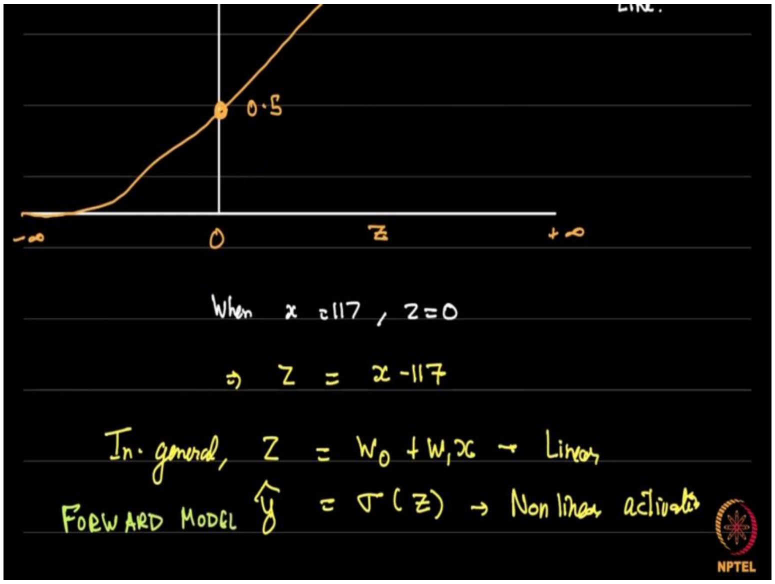


$\sigma(z) = 0.5$, when $\underbrace{z = 0}$ → Classification Line.

Well, we look back here where do we want that line, we want that line at when sigmoid hits exactly 0.5, but remember we can do that reverse calculation sigmoid of z is 0.5 when z = 0. So, when z = 0 sigma = 0.5, so z = 0 is the classification boundary or classification line. This is true both in one feature, two features or in multiple features as we will see shortly. So, z = 0 is the classification line.

But what I have, here on the x axis is x, it is not z, so how do I set that now notice let us say arbitrary I say that this boundary should happen at 117.

(Refer Slide Time: 17:47)



When $x = 117$, $z = 0$

$\Rightarrow z = x - 117$

In. general, $z = w_0 + w_1 x$ → Linear

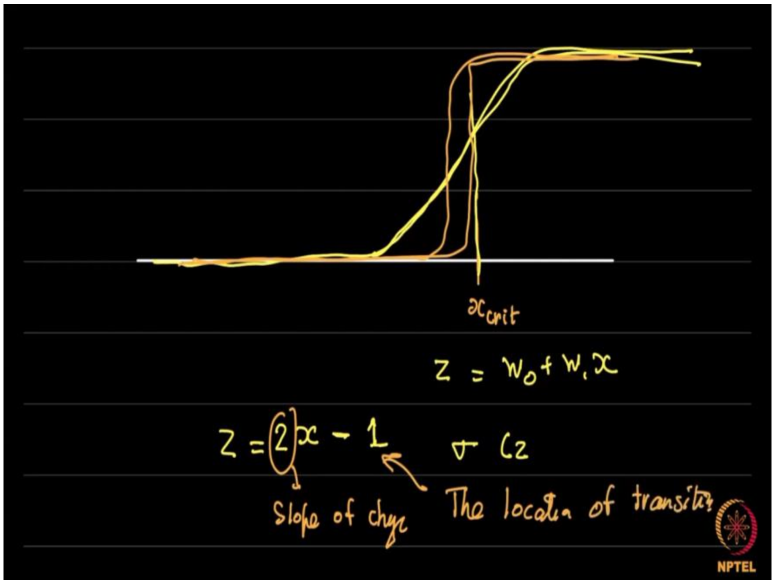
FORWARD MODEL $\hat{y} = \sigma(z)$ → Non linear activatio

So, I will say something like when x = 117, z = 0, now you might be wondering how exactly am I going to build this model, I do I have to do this by hand the answer is no we will see further details in the next video on how we actually end up building this model. But when x = 117, z = 0, so you can kind of set z is x - 117. So, this satisfies all our properties. Now notice when I do z is x - 117, when I come here the value becomes z = - 7, here the value is z = 0, here the value equal to z is 13.

Now what happens at z = - 7? $\frac{1}{1+e^{+7}}$, e^7 is a reasonably sized number will be approximately so 2 cubed is 10, so this will be around 100, so 1 by 101 so the probability that this is a failure becomes approximately 1 percent. Similarly, when you come here the probability that this is a success might become something like 99 percent. So, which tells you that in case you define z as x - 117 this would work.

Now in general we do a very clever thing. we say z is for 1 variable $w_0 + w_1 x$. Now this should immediately start looking a lot like our linear model and we say $\hat{y}$ is $\sigma(z)$. so, notice this you first do a linear transformation on x, next you do a non-linear activation. This twofold stepping first taking linear step and then taking a nonlinear step I will show this pictorially also shortly, is what makes the forward model. but why are we defining the forward model this way.

(Refer Slide Time: 20:17)



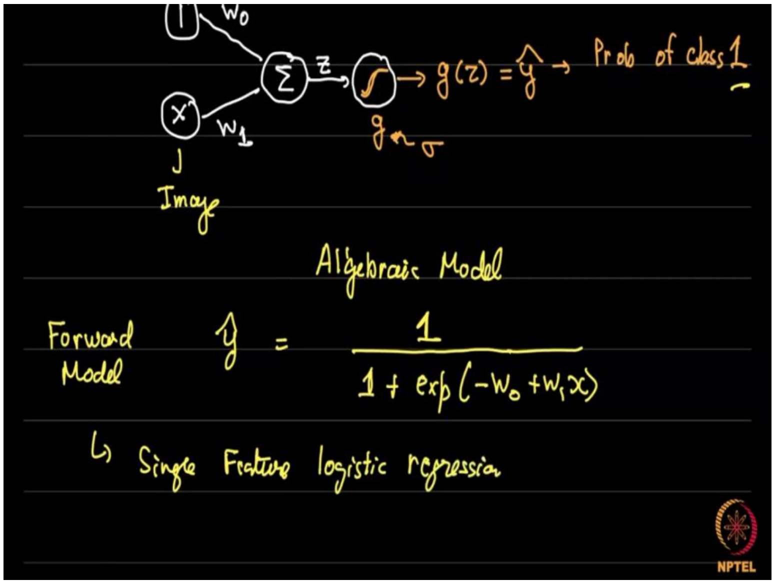
Again, let me explain it using the same example, see in general when you have only one variable, you will always have the situation. you have something like this and you have some critical point, let us call this x critical, where you want to make this jump between a failure and a success. So, you could have something like this failure, failure, failure, success, now z is like I said $w_0 + w_1 x$. Now in this case in our example here you can see w_1 is 1 and w_0 is - 117.

So, when you set w_1 is 1 and w_0 = - 17 you immediately get z = 0 means x = 117, this is x critical. but two three things could happen, one I have drawn it in a way where of course for the chip example it means when power is increased, probability of failure increases. but you could have definitely a case where higher numbers are better and you could have an opposite direction thing. So, which is why you might actually require sine flow.

So, you might require $w_0 - x$, so in that case whenever z crosses 0.5 or whenever z crosses 0, it goes in the opposite map. The second thing that w1 controls is imagine a case where I write $z = x - 1$ but I make it $2x - 1$. So, when I do that, I will get sigmoid of z such that this slope is a whole lot sharper, so when we keep a sigmoid it is very smooth and it transitions slowly from failure to success.

As you increase the slope as you increase w_1 you are looking at sharper and sharper transitions. So, this controls the slope of change and this controls usually the location of transition.

(Refer Slide Time: 22:43)


1
w_0
Σ
z
$g(z) = \hat{y}$ → Prob of class 1
X
w_1
Image
g ~ σ
Algebraic Model
Forward Model
$\hat{y} = \frac{1}{1 + \exp(-w_0 + w_1 x)}$
↳ Single Feature logistic regression
NPTEL

Put together our model is like this. Our model is let me draw this in a figure and you will see in some sense it is easier to interpret in a figure. We do the same thing that we did with the linear model, 1 multiplied by w_0, x multiplied by w_1, you sum the two and let us call this output z. You do not stop here you put our sigmoid function on top of it and what comes out here is $g(z) = \hat{y}$. So, let us say this is a function g which in our case was sigma.

So, look at the pipeline the pipeline is X comes in, let us say the power, you multiply this by some weight multiply 1 by some weight add them to get a value run it through the sigmoid function and here you get probability of class 1, whatever class 1. In case what goes in here as X is an image, then maybe class 1 is a cat, then what goes in is an image comes here put a sigmoid, it will give you is this image a cat or a dog.

So, that is the simple example of how you build a forward model with just a single variable. Now this of course is the pictorial or the graphical representation of the model. Here is the algebraic model I have already written, but let me write it out explicitly. Our algebraic model is y-hat = sigmoid of which is,

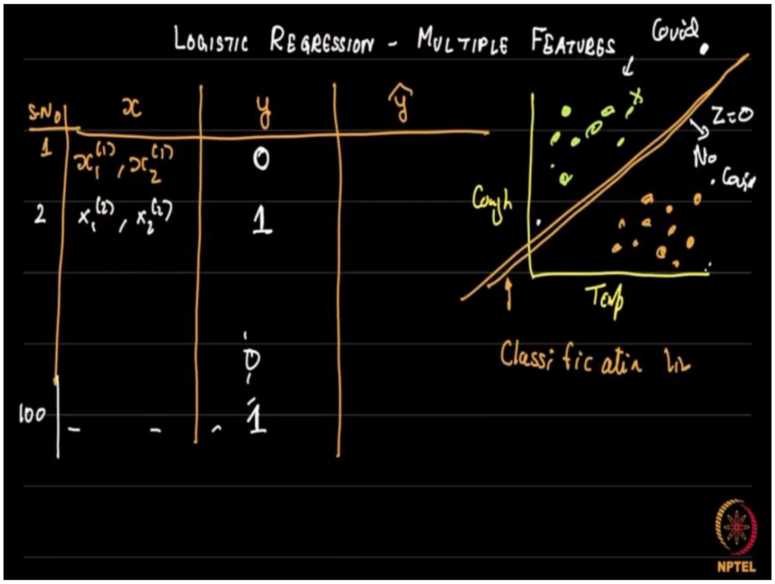
$$\hat{y} = \frac{1}{1 + \exp{(-w_0 + w_1 x)}}$$

So, this then is our forward model for single feature logistic regression this is called the logistic regression model.

Now how do we learn w_0 and w_1 can I go now of course once I give you w_0 and w_1 you can draw this picture and you can do a classification. But can I find out w_0 and w_1 given this data that is given some engineer is measuring some thermal data in a chip and he wants to find out what is the critical temperature or what is the critical power at which this chip starts failing. Now you can immediately see this is given the effect you are trying to find out the cause.

So, this is an inverse problem and we will see how to solve this inverse problem in the next video when we talk about the loss function etcetera. Now this all this is of course for a single feature. Now what we will move on to next is multiple features, so let us see that.

(Refer Slide Time: 26:04)



LOGISTIC REGRESSION - MULTIPLE FEATURES

S.No	x	y	$\hat{y}$
1	$x_1^{(1)}, x_2^{(1)}$	0	
2	$x_1^{(2)}, x_2^{(2)}$	1	
		0	
100	- -	1	

Covid
Z=0
No Case
Cough
Temp
Classification line
NPTEL

The same ideas that we used for just doing the single feature case, we can now think about multiple features. So, what do I mean by multiple features? When we give the data set, we have these cases where the input, whatever it is, whether it is an image. If it is an image, it would require a lot of input. so, taking back my favourite 60 cross 60 image this would require 3600 image or pixel information in order to give one input.

Let us take a simpler case where we have only two inputs, so let us call this the first data point has just two inputs or two inputs or the input is a two cross one vector. The ground truth again always remains either 0 or 1 and then you have $\hat{y}$. So, what could be an example of this? So, let me take a non-thermal example just for some simplicity here. So, let us say we are collecting these patients or we are finding out data for these patients who are coming to a doctor.

And these patients come with a temperature and with some cough and we measure this. So, patient one, let us say comes there with some reasonably high temperature, but the cough frequency the frequency with which this person coughs is kind of low. Patient two comes and this person again has just two features you are just measuring these two, this person has slightly lower temperature, but this person's cough frequency is high.

So, let us say we keep on collecting things like this, lots of people here and a lot of people here something like this one, I am just making something up and a new person comes. So, let us say after doing all this, we track these patients and find out whether they have covid or not and this is basically the data set, obviously this is not a realistic data set. In realistic covid, we would collect a whole lot more information rather than this we would have X-rays and whatnot.

But let us say again I am taking a simple case. So, something of the sort and you just label these people as saying person one came this guy had some temperature some body temperature and some cough frequency and you know what this person did not have covid. Person two came at some other temperature some other cough frequency but this person did have covid and similarly you could create let us say this entire database.

Let us say we have again 100 patients 0, 1, 1, 0, 0 etcetera, so this is our database and we collect that. Now the classification question is if a new patient comes can you tell me whether this person has covid or not. Now please understand I mean we should not get lost fully in machine learning; we are still doing an inverse methods course please understand how this is an inverse problem.

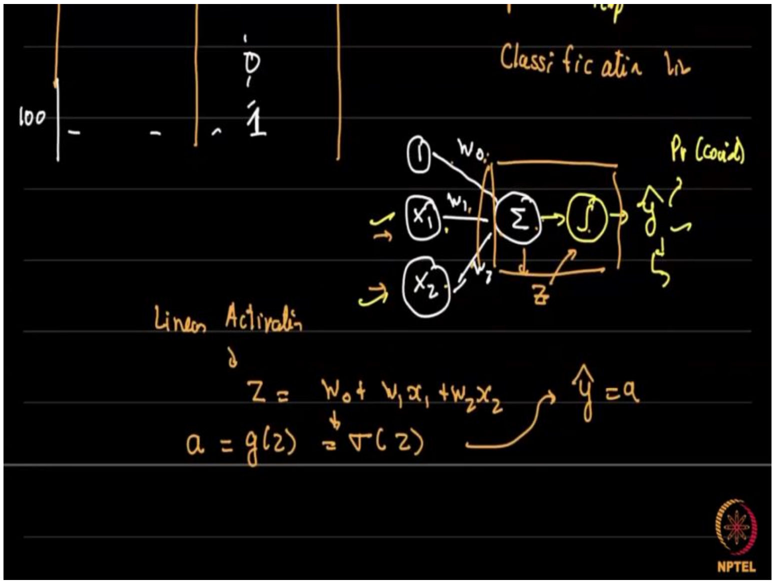
The inverse problem is like this. If you tell me, patient one has covid patient two has covid can you does not have covid etcetera, can you find out the parameters in your model that is the inverse problem. What parameters or what underlying costs would have led to this effect? Now we are mathematizing that and making it to a data rather than a physics model here I mean we really do not know the mechanism.

In fact, it is not causative, it is a correlation that we are looking at. It is not as if puff and temperature caused covid but it is just that these parameters are implicit data-based causes that we are looking for. Now coming back to this; notice that in this case we can think of a line here which I am going to call the classification line. This is very much like the classification line I drew here right at the beginning.

On one side of it we had all the failure cases on another side of it we had all the safe cases. So, similarly coming back here on one side of it, we have all let us say the no covid cases and on other side of it we had all covid cases, high temperature high cough covid, low temperature low cough no covid but high temperature but low cough no covid something of that sort. So, we have a line obviously the real in case we made a covid case of this sort.

It would not sequester or classify, so neatly it will look a little bit more complex which I will talk about shortly. Assuming this is somewhat of a realistic example let us proceed further. Now this somehow is basically going to turn out to be the equivalent of our z = 0 line but how are we going to achieve it.

(Refer Slide Time: 31:13)



Simply extending whatever we did till now. So, I am going to first start pictorially unlike last time because it is actually the more natural way to go. So, I have one I have x_1, I have x_2 as input, I am going to do exactly the same thing blindly assuming this is somehow going to magically work and it does, I will just take w_0, w_1, w_2. Interestingly enough the earliest models of neural networks were exactly like this in fact they were physical models used exactly for classifications.

We do this and we know we want an output which is again either 0 or 1 and again we will run this through a sigmoid and what comes out as output, basically will give us the probability of covid or not. You will say okay, how is this going to magically work out. Nothing it we do not really care that it works out somehow exactly there is no physics here all we are doing is I was given three numbers or two numbers x_1 and x_2.

I need to give an output that is going to be between 0 and 1, then how will it figure out covid or not. That is where the magic lies in our optimization routine and the loss function that we will give which we will give the next video. But all we are doing right now is simply setting a forward model. Please understand this fact it is a key fact. All we are doing is blindly setting a forward model which will somehow satisfy our data representation.

What is our data representation? That I should input the temperature I should input the cost frequency my output data representation said that it should look like a number which is 0 or 1. We modified it a little bit and said I will give us number between 0 and 1 and that is it, our modelling is done. This will look ridiculous but it tends to work. So, I will talk more about this sort of disturbing situation when we come to neural networks.

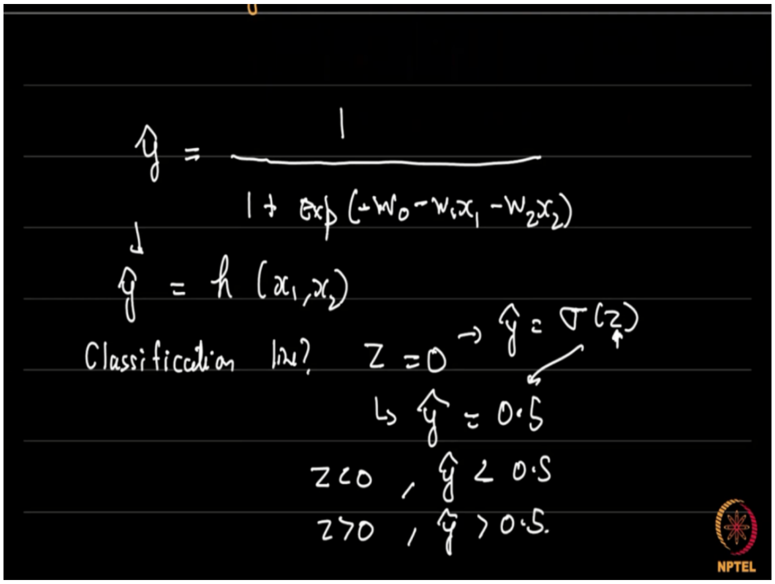
That we almost give nothing and still we get fairly good outputs. So, we have simply satisfied mathematical constraints do having nothing to do with covid saying I am only looking at a probability so it should be a number between 0 and 1. And how does this look? This looks like this, it looks like this that the output here, let us say z and z is what goes in here, so $z = w_0 + w_1x_1 + w_2x_2$, this z I am going to call as useful as the linear activation or a linear combination or a linear portion of the neuron.

Now usually when you see a neuron in neural network as you will see both these are put together and it is assumed that both the linear combination and the non-linearity happen one after the other.

Then I will say a just to say activation is g of z which is basically sigmoid of z and z is a scalar so you do sigmoid of z and then you say $\hat{y}$ is the same as a. So, this is the sequence of steps we follow for the forward model.

Give me a temperature, give me a curve frequency give me w_0, w_1, w_2, combine these then just run them through a sigmoid and I will get a probability. How do I find out that ensure that it satisfies this data set that is where the inverse problem kicks in and you look at this data set. And we have to set an objective function which we look at in the next video and you adjust it so that $\hat{y}$ works for all the cases that we have given the data set from.

(Refer Slide Time: 34:55)



$$\hat{y} = \frac{1}{1 + \exp(-W_0 - W_1x_1 - W_2x_2)}$$

$\downarrow$

$$\hat{y} = h(x_1, x_2)$$

Classification line? $Z = 0 \rightarrow \hat{y} = \sigma(z)$

$\hookrightarrow \hat{y} = 0.5$

$z < 0, \quad \hat{y} < 0.S$

$z > 0, \quad \hat{y} > 0.5.$

If I write it explicitly this is what it will look like, notice $\hat{y}$ is simply a scalar because all we are demanding out of this network is not multiple numbers but a single number which tells you how probable is covered or not. So, what it is $1 + \exp\,(-w_0 - w_1x_1 - w_2x_2)$, notice $\hat{y}$ is simply some function of $x_1,\ x_2$. Notice this, in this figure it is not so obvious that it is a function.

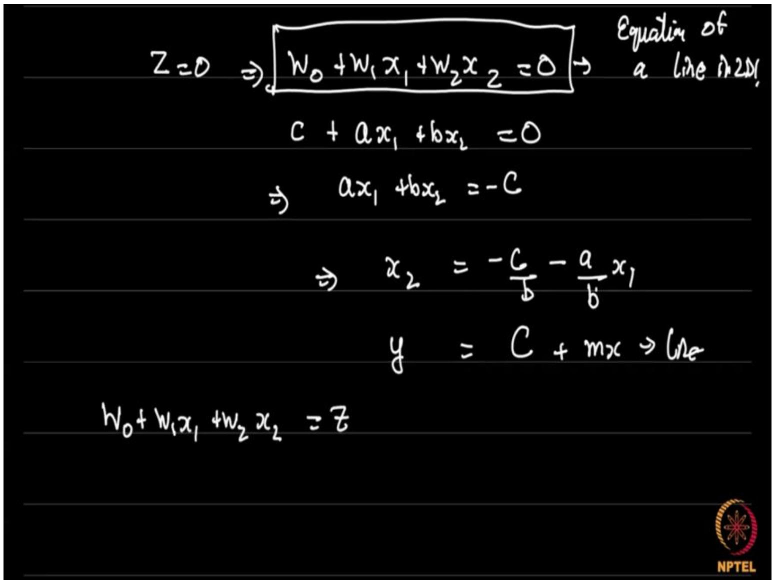
But regardless of how many things I put in the middle $\hat{y}$ is always a function of the input with some intermediate calculations going on. What about the classification line? So, the classification line is the line z = 0. Why is it the line z = 0? Because at z = 0, $\hat{y}$ will be exactly 0.5. How come? $\hat{y}$ was sigmoid of z and when z = 0, y-hat obviously is sigmoid of 0 which is exactly 0.5.

So, one side of the line you will have one class because when z is less than 0 this means $\hat{y}$ will be less than 0.5 and when z is greater than 0, $\hat{y}$ is greater than 0.5. Let us look at what this model is doing what is it actually doing. I will tell you a graphical interpretation also but all it is doing is trying to find out these parameters which will somehow turn our input into a line the line z = 0.

If you happen to fall on one side of the line it will say no covid, if you happen to fall out on the other side of the line you would say covid. Now you might say how is it doing that? Think about how we are doing it, we are doing this visually all the algorithm is doing it is doing it algebraically. Why? Because as you will see shortly, z actually means perpendicular distance from this line.

If it happens to be negative it falls on one side, if it happens to be positive it falls on another side. The genius really of the algorithm which we will see in the next video is to find out this line but this is no different from finding out the regression line which balances the data points. So, all it is doing is finding out a balance between the data points and the further you go here, the more the probability that you have covid, the further you go down here the more the probability that you do not have covid.

(Refer Slide Time: 37:43)



$$Z = 0 \Rightarrow \boxed{W_0 + W_1 x_1 + W_2 x_2 = 0} \rightarrow \text{Equation of a line in 2D}$$

$$c + a x_1 + b x_2 = 0$$

$$\Rightarrow a x_1 + b x_2 = -c$$

$$\Rightarrow x_2 = -\frac{c}{b} - \frac{a}{b} x_1$$

$$y = C + mx \Rightarrow \text{line}$$

$$W_0 + W_1 x_1 + W_2 x_2 = z$$

So, let me see say show you why this is, so z = 0 means $w_0 + w_1 x_1 + w_2 x_2 = 0$. This of course is the equation of a line in 2D, why so, it might become clearer if I label these somewhat differently. Suppose I call this,

$$c + ax_1 + bx_2 = 0$$

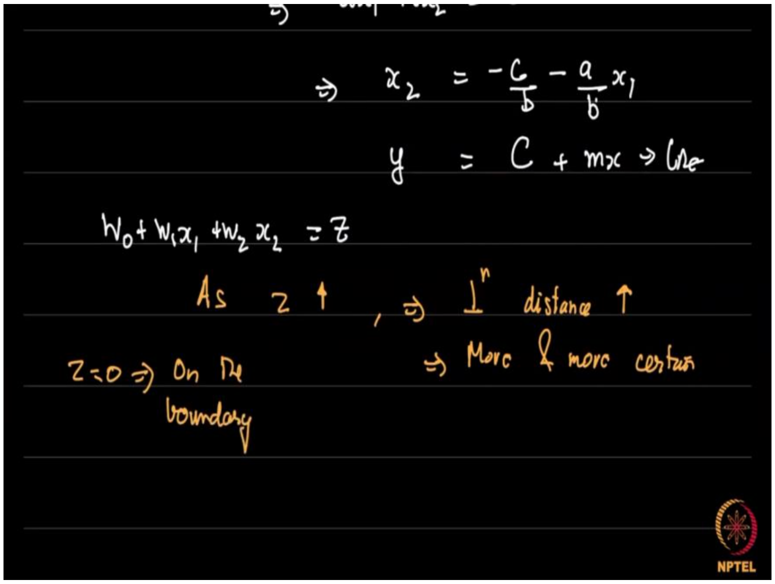
$$ax_1 + bx_2 = -c$$

$$x_2 = \frac{-c}{b} - \frac{a}{b}x_1$$

which is the same as saying $y = c + mx$, so this is a line.

Similarly, z in general is $w_0 + w_1x_1 + w_2x_2$, now you might have to brush up your 2D analytical geometry to remember this, that this essentially is the perpendicular distance. So, suppose I look at this point x_1 and x_2 and I want this distance that will exactly be z, $w_0 + w_1x_1 + w_2x_2$will give this. So, if it is positive it lies on one side, if it is negative it lies on another side notice what happens as z increases.

(Refer Slide Time: 39:23)



$$\Rightarrow x_2 = -\frac{c}{b} - \frac{a}{b}x_1$$

$$y = C + mx \Rightarrow \text{line}$$

$$W_0 + W_1x_1 + W_2x_2 = z$$

As $z \uparrow$, $\Rightarrow \perp^r$ distance $\uparrow$

$\Rightarrow$ More & more certain

$z = 0 \Rightarrow$ On the boundary

As z increases on one side this means that perpendicular distance increases. If perpendicular distance increases, it means more and more certain. In case this is not obvious let us look at the 1D case which is why I started with the 1D case, when you are right here you are only about 0.5 sure that whether this will fail or not. As you go further and further here sigmoid of z becomes closer and closer to 1 and that makes sense.

Because the further you move away from this boundary then more and more the power is increasing and that is telling you that I am more and more likely to fail. Similarly, further and further we move into this boundary the more and more the power decreases and the more and more certain you are, that this is just completely safe. So, these two put together basically tell you the interpretation of z.

So, z basically tells you which side of the classification line you are lying on and also how far you are. So, if you are exactly on the boundary then you are at z = 0 and this is basically how binary classification works. By somehow magically figuring out what this optimal line is, the classification optimal line. Notice once again the line is optimal in some sense just like it was in regression.

So, both in regression as well as classification we have somehow magically landed up on this line which needs to be optimal in some sense. But optimal in what sense, what is the lose function that I can give, that will tell me that I am well balanced. So, that goes initial question that I was erasing what is the loss function and what is the optimization algorithm that will ensure that I come somewhere in the middle, so that we will see in the next video.

In this video we saw that the data representation basically should be represented as 0 or 1 as far as y is concerned it should be between 0 and 1 as far as $\hat{y}$ is concerned and that the forward model simply is sigmoid of z where z is some $w_1x_1 + w_2x_2$ etcetera. But in the next video we will see how we will land up at w_0 and w_1 and solve the inverse problem. So, I will see in the next video, thank you.