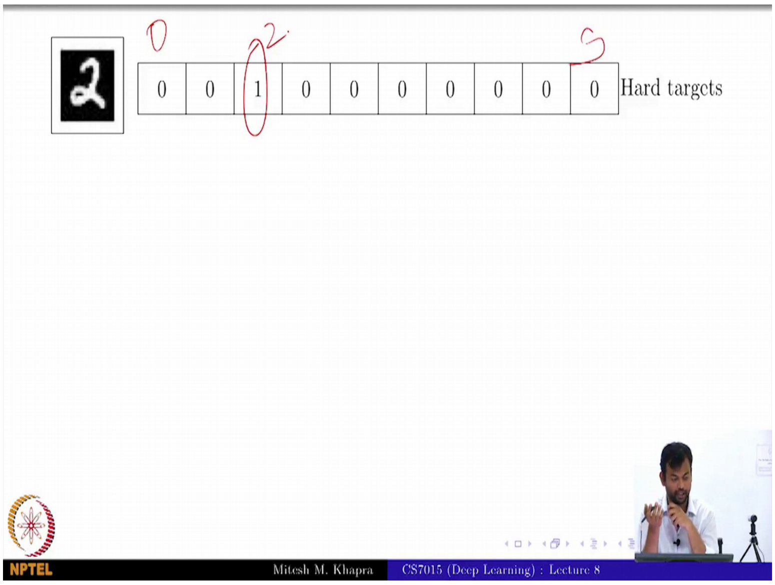
Deep Learning
Prof. Mithesh M. Khapra
Department of Computer Science and Engineering
Indian Institute of Technology, Madras

Module – 8.8
Lecture – 08
Adding Noise to the Outputs

So now going on to the next module which is Adding Noise to the Outputs.

(Refer Slide Time: 00:19)

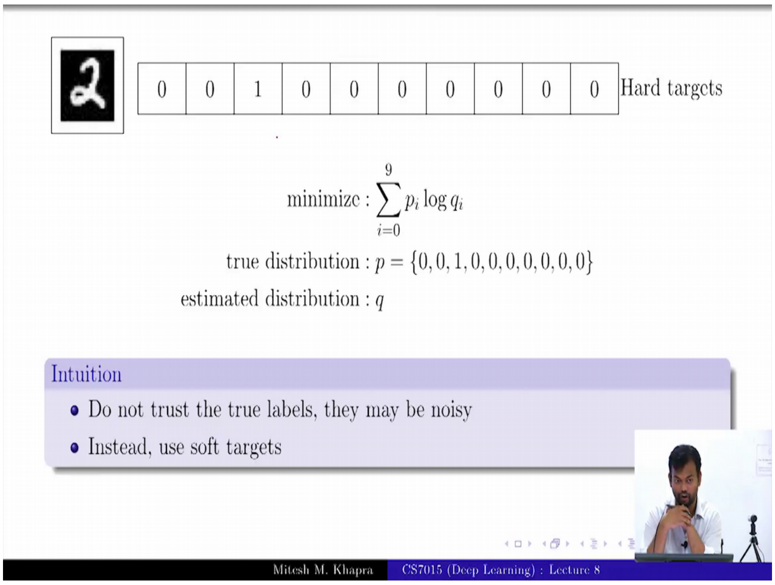


0	0	1	0	0	0	0	0	0	0

Hard targets

So, here when you are given some training data, this is the label vector that you have been given, right? Where one of these elements is 1; so, these are like 0 to 9 8, where which digit it is and in this case it happens to be digit 2. So, that element is 1 right that is the true training data given to you.

(Refer Slide Time: 00:40)



0	0	1	0	0	0	0	0	0	0	Hard targets

$$\text{minimize} : \sum_{i=0}^{9} p_i \log q_i$$

true distribution : $p = \{0, 0, 1, 0, 0, 0, 0, 0, 0, 0\}$

estimated distribution : q

Intuition

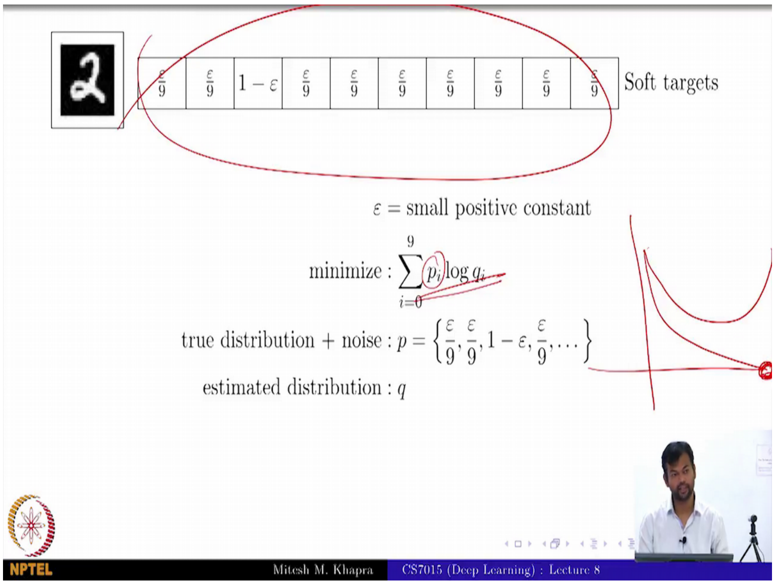
- Do not trust the true labels, they may be noisy
- Instead, use soft targets

Mitesh M. Khapra CS7015 (Deep Learning) : Lecture 8

So, what you could do is actually and actually what you try to do is, minimize this quantity p i log q i, where what is p i? p i is the vector which was given and, what is q i? The predicted probabilities ok. So now, when you try to add noise to the output, what you actually do is, you see that I do not trust the true labels, they may be noisy.

Whatever data you have given to me that is one way of looking at it. That I do not trust it, I will just say that it is noisy. The other way of looking at it is that in some way I am ensuring that; I do not try to over fit to this label, right? Because now my true whatever I am trying to optimize, let me just go to that and let us see. So, instead what we will do is we will use soft targets.

(Refer Slide Time: 01:19)



$\frac{\varepsilon}{9}$	$\frac{\varepsilon}{9}$	$1-\varepsilon$	$\frac{\varepsilon}{9}$	$\frac{\varepsilon}{9}$	$\frac{\varepsilon}{9}$	$\frac{\varepsilon}{9}$	$\frac{\varepsilon}{9}$	$\frac{\varepsilon}{9}$	$\frac{\varepsilon}{9}$

Soft targets

ε = small positive constant

minimize : $\sum_{i=0}^{9} p_i \log q_i$

true distribution + noise : $p = \left\{\frac{\varepsilon}{9}, \frac{\varepsilon}{9}, 1-\varepsilon, \frac{\varepsilon}{9}, \dots\right\}$

estimated distribution : q

NPTEL

Mitesh M. Khapra

CS7015 (Deep Learning) : Lecture 8

So, this is what i mean by soft target. Assume that there was some epsilon noise in your labels; so, instead of treating this as 1 and all 0s. Treat the true label as 1 minus epsilon and divide that among the remaining 9 entities right that probability must divided among the remaining 9 entities.

So now when you are trying to minimize this, what is p i? This soft distribution right and q i is the predicted distribution. So, you see why this acts as a regularization, why does it act as a regularization? What is the aim of regularization? Do not over fit on the training data right. To over fit on the training data, what should it have done? It should have treated only the correct label. Now if I am giving it this information then I am not allowing it to over fit on the training data right.

Because now with this distribution, this quantity will not get minimized, when q i is equal to the 1-hour distribution where all the masses on 2. Do you get that? So, in some sense we are making sure that, now if it tries to over fit on the training data; it will not get the minimized error right. So, you have this corrupted the outputs of it everyone gets this, is ok? The trainer no that is the whole point

Student: (Refer Time: 02:40).

No.

So, that is thing right; so, some of these are heuristics based. So now, we have started with this whole derivation, where we try to show the relations between trainer error tested or not, but things that we have seen some of these things right, even whatever unfortunately, I tried to prove on the previous slide the weight decay thing; even that is only for these neat networks where you do not have any hidden layer and so on right.

So, most of these are just heuristics, you are just saying that the principle is that; you will not allow the true training error as computed from the training data to go to 0. If you do that you know that you are going to over fit. So, try whatever you can to avoid that ok. That is the idea, do you agree that doing this is going in that direction?

Student: (Refer Time: 03:25).

Training data. The hope is that if you do not do that then it will not under fit on the test it right.

There is no I mean I have you are you looking for a proof, where I say that doing this we will ensure that a training error does not go to 0, but the test error comes close to the training error. There is no such proof right, just a heuristic. It is going by the principle that if I do not allow the training error to go to 0. Then hopefully I will over fit, I will not over it as much as I would have otherwise right.

So, that you can think of it as this way right; so, this is the curve that you are seeing it. This was a training curve; this was your test curve. You are preventing from entering this region where the error is 0; that means, you will end up somewhere here right? And you know that that is a more preferred point as compared to this. That is the intuition that you are going right, is that?