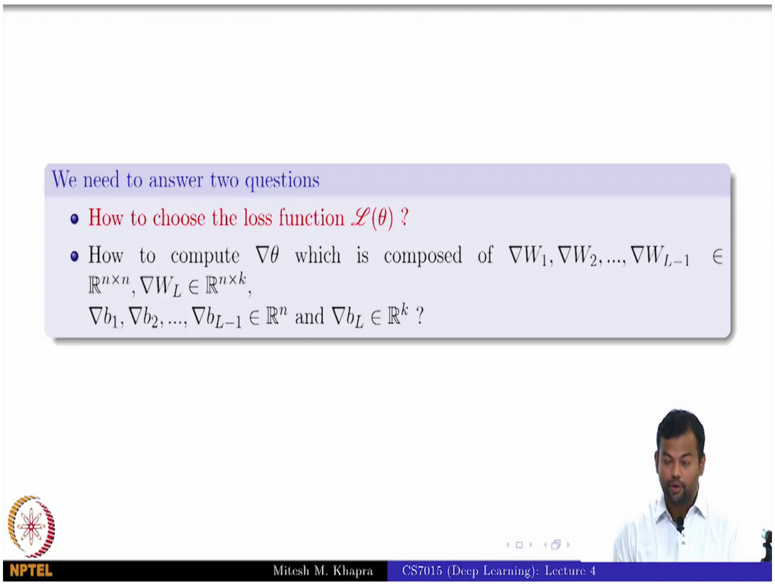
Deep Learning
Prof. Mitesh M. Khapra
Department of Computer Science and Engineering
Indian Institute of Technology, Madras

Module - 4.3
Lecture - 04
Output Functions and Loss Functions

We go on to the next module where we will be talking about Output Functions and Loss Functions, ok.

(Refer Slide Time: 00:19)

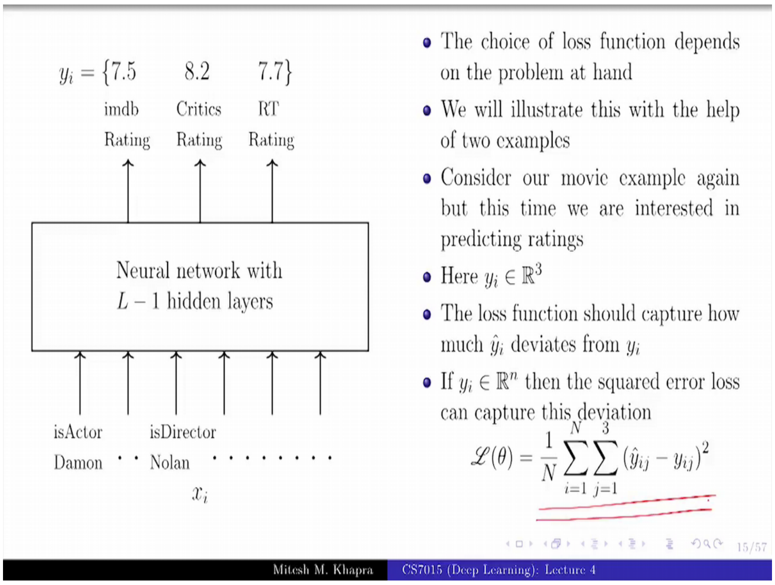


We need to answer two questions

- How to choose the loss function $\mathscr{L}(\theta)$?
- How to compute $\nabla\theta$ which is composed of $\nabla W_1, \nabla W_2, ..., \nabla W_{L-1} \in \mathbb{R}^{n \times n}, \nabla W_L \in \mathbb{R}^{n \times k}$, $\nabla b_1, \nabla b_2, ..., \nabla b_{L-1} \in \mathbb{R}^n$ and $\nabla b_L \in \mathbb{R}^k$?

The question that we are going to focus on is how to choose the loss function, but I will show you that it is tightly coupled with the choice of the output function also. Remember that we had said that we have a spatial O function as the output function, I have not told you what that O is, and now that is what we are going to define.

(Refer Slide Time: 00:33)


$y_i = \{7.5 \quad 8.2 \quad 7.7\}$
imdb Rating
Critics Rating
RT Rating
Neural network with $L-1$ hidden layers
isActor Damon
isDirector Nolan
x_i

- The choice of loss function depends on the problem at hand
- We will illustrate this with the help of two examples
- Consider our movie example again but this time we are interested in predicting ratings
- Here $y_i \in \mathbb{R}^3$
- The loss function should capture how much $\hat{y}_i$ deviates from y_i
- If $y_i \in \mathbb{R}^n$ then the squared error loss can capture this deviation

$$\mathscr{L}(\theta) = \frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{3}(\hat{y}_{ij} - y_{ij})^2$$

15/57

Mitesh M. Khapra CS7015 (Deep Learning): Lecture 4

Now, the choice will be loss function actually depends on the problem at hand. And that is exactly the question which had come up, right that in some cases it is to have sigmoid as the output function because your values are between 0 to 1, but whatever there are cases where your output is not between 0 to 1. So, it definitely depends on the choice of the on the problem that you are trying to solve. So, we will illustrate this with the help of 2 examples, and these 2 examples will cover a broad range of problems that you will encounter or if you are working in machine learning.

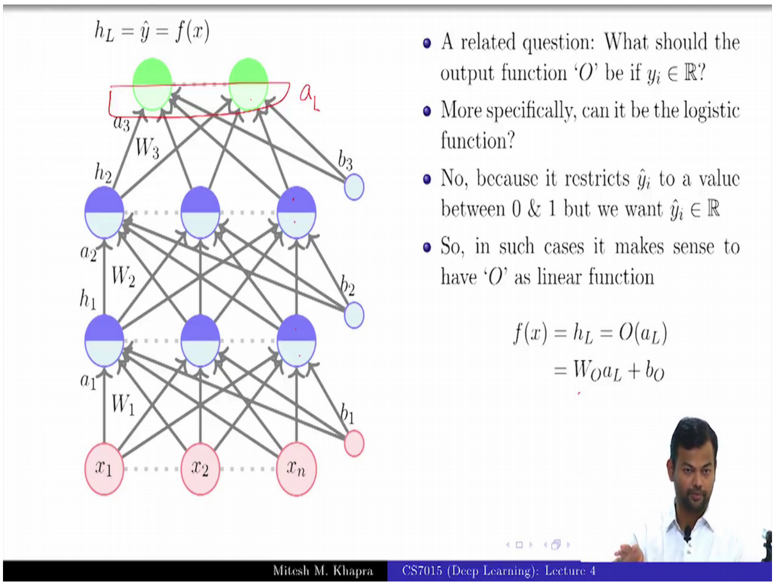
So, the first problem is again you are given the input as movie. You are using a neural network with l minus 1 hidden layers and an output layer y hat right. So, this is, sorry, this is a true one. So, you have an output layer and the output layer is going to predict the IMDB rating the critics rating and the rotten tomatoes rating.

Is that fine, ok? So, what kind of problem is this? People have done machine learning, this is a regression problem. And notice that the output values that you want to predict are not bounded it by 0 and 1, they are still bounded by 1 to 10, but in general you could imagine that there could be problems. So, there are no bounds at all right it could be a very large number, is that clear? Now here yi belongs to R 3.

So, remember in all these cases we were assuming that we just want to predict one value, but nothing stops you from predicting multiple values at the same weight. So, your output is now 3 dimensional, you are taking an n dimensional input and trying to predict

3 values from it, fine the loss function should capture how much yi had dev deviates from yi. So, this is a valid or maybe we corrected on this lane, ok. So, this is the formula which was supposed to be in there. So, you take, you have predicted 3 values, and you know the true 3 values, you just take the difference between these. Is that clear? The first element of the predicted value, minus first value of the actual value and so on for all the 3 values that you want to predict.

(Refer Slide Time: 02:42)



- A related question: What should the output function 'O' be if $y_i \in \mathbb{R}$?
- More specifically, can it be the logistic function?
- No, because it restricts $\hat{y}_i$ to a value between 0 & 1 but we want $\hat{y}_i \in \mathbb{R}$
- So, in such cases it makes sense to have 'O' as linear function

$$f(x) = h_L = O(a_L)$$
$$= W_O a_L + b_O$$

Mitesh M. Khapra CS7015 (Deep Learning): Lecture 4

Now, you have a loss function, but what should be the output function in this case? Can it be the logistic function? Yes, no, it will be bounded between 0 to 1, and you know that your output cannot be bounded between 0 to 1, ok. So, in such cases then what is a good output function to use? One option is to scale it. So, I will keep that aside, why do that? It is unnatural and you are actually clamping it and then trying to scale it, right. So, can you do something more natural in that, just use a sum which is linear function, right. So, what we could do is you could have O as a linear function.

So, what that means is, again remember that this is a of l, and I know all the computations that have happened so far; a linear transformation, non-linear, linear, non-linear and then again linear. So, I have computed a of l, from that I want to compute the final output, right. So, I could just have it as a linear function of the input which is a of L in this case.

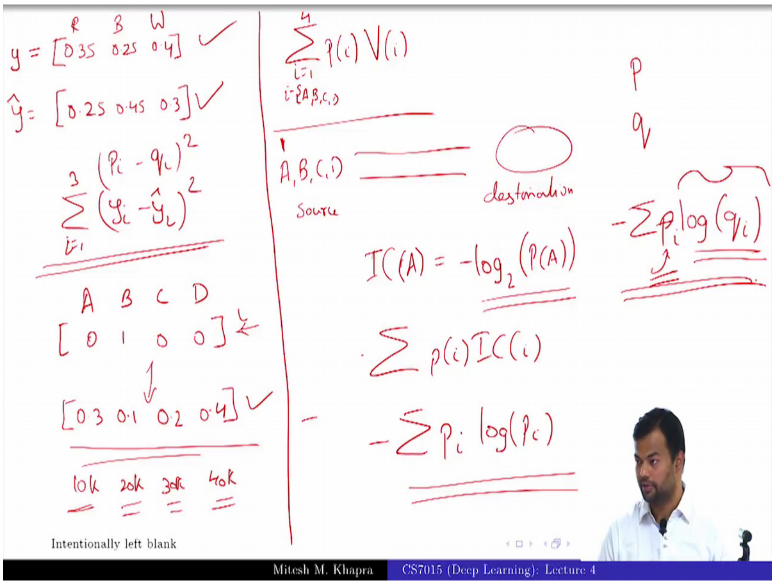
Does it make sense? How many of you feel it makes sense, why? Because now it is no longer bounded, you could this linear transformation your weights could be adjusted in any way to get a value whatever you wanted, whether you wanted between 1 to 10 or 1 to 100 or 1 to 1000, these weights could be adjusted to do that. So, at least you are not bounding it and it is free to learn, what is the range from the data it should be able to run, but how should you adjust these W's.

So, that you get the desired range, now tell me why would it not happen that you learn W's you start predicting values like 1000, 10000 and so on in this particular case where your input is bounded by 1 to 10 sorry; your output is bounded between 1 to 10. Why would it happen? I this is my argument and you prove me wrong, right I would say that if you have chosen a linear transformation which is not bounded, I the network could learn weights which start producing a rating of 10000, 20000 and so on because it is not bounded.

But you know that that is wrong because the ratings can only be between 1 to 10; so why would that not happen, because you are minimizing this loss function. So, if you start predicting values like 10000, when your actual rating was 9 then you have a 10000 minus 90 whole squared loss, that is a very high loss. So, it will start moving you away from that configuration. So, the training is always guided by the objective function. So, if your training happens well it will try to prevent this.

Now, suppose let us take a simple thing rate that; you are given a our same ball example for probability. So, you are given an earn which has balls of 3 colors; say black, white and yellow.

(Refer Slide Time: 05:38)



And you have to put the balls in that. So, you know that the true probability distribution is actually 0.35, 0.25 and 0.4, for red black and white, ok. This is the true probability distribution, you have put say thousands of balls and an on. Now what you do is, you just allow me to peep into the earn or you allow me to take some samples from there. You tell me take these 100 samples, and you ask me, tell me what this probability is, right. So, this is the true probability that you know is true right because you know it because you have estimated.

Now, you just give me a small sample from there and ask me to estimate it, and based on that I actually estimate this, ok. So, there was a true probably distribution and an estimated probably distribution. Now I want to find out how wrong I went, right afterwards you tell me the answer you tell me that this is what the true was and this is what you predicted.

Now, I want a way of computing how wrong I was, right. So, how do I do that? You already know this and these are 2 vectors, what can I do? You could just do the; this is valid anything wrong with this? In principle no, you could just treat these as any 2 vectors you have a true value you have a predicted value you just take the squared error difference between them? But you know this is a probability distribution, right? You should be able to do something better than this, you know this is a special quantity this is not just any number that you are predicting, you are trying to predict a distribution. So, you should be able to do something better than that right. So, that is what we want to see, how to do something better than this, that is what our quest is.

Now, again why we are at this, right, I also want to make a because this is something people do not immediately understand. So, I just want to make a case for something else. So, I will just do that ok. Now suppose there is this IPL, and there are 4 teams in the semifinal, let us call them A B C and D, now I was not in town after the semifinal. So, I just know the results up to semifinal, and then the finals also happen, and one of these teams wins, let us call it the B team, right the B team wins. Can you express this in terms of probability? Can you express this in terms of distribution? What do you mean my 0 and 1 B has won.

So, it is a certain event because it has 1 now. So, what is going to be the distribution? 0 1 0 0. So, this event happens with 100 percent probability. Now the same case can you ok. So now, let us do the same thing that is as I said I was not in town, and you asked me tell me which team would win that is; I know these 4 teams have qualified in the semifinals, and I know who the players are and so on.

And with my limited knowledge of cricket I will predict something right so, say I predict this. Can you again tell me how wrong I was? You know what the true label is and you know what I predicted. You can tell me how wrong I was, ok. So, the case which am trying to make is that even if the event is certain, you can still write it as a probability distribution where all the mass is allocated to the correct output. Can you relate this to a classification problem? When you see training data, you have already observed it, suppose there were 4 classes possible.

Apple, orange, mango and banana; if you have seen it is apple, and if you ask you what is the distribution, what will you tell me 0, 1, 0, 0; you will express it as this one hot vector; where all the probability mass is concentrated on the guy which is correct, right. So, even certain events which happen with certainty you can write them as a distribution rate, where all the masses are located on the true label. So, that is how all classification problems when you are dealing with multiple class classification problems it is often the case that you will write it as this.

That your true label is given to you in this format, there were 4 possible events, 4 possible classes or k cost possible classes, out of which only one is correct and then you make a prediction, and you want to now find out how different was your prediction from the true label. You are trying to get the set of how this relates to a classification problem, and this is that is why this is of interest to us ok, ok.

So, this so, we will see this soon. Now the next thing that we need is how many of you know what is entropy? Forget about cross just entropy, that is why I have left 2 slides intentionally blank ok. So, so now, let us see where do I go with entropy, ok. How many of you know what is expectation? Please, fine. So, again the same thing now I knew that this was the distribution which I think I am into gambling, am not I am into gambling, and I try to bet on these teams.

And I bet some amount on each of these, can you tell me what is the expected? Reward that I will get. So, what am I saying wait. Suppose this is the case that if team a wins I get 10 k rupees or my net profit is 10 k rupees, if team B wins my net profit is 20 k rupees and C and D so on, right. You get the setup for every even there is an associated value with it. This is the value of event A winning, B winning, C winning. So, the net profit in each of these case. So, what is my expected net profit? Do not give me a formula [FL] how sigma overall events, how many events do I have yet for right. So, rather I should say I equal to ABCD, the probability of I multiplied by the value associated with that event. So, this is how you compute expectation, everyone gets this, ok.

So now suppose say am doing this rate there are suppose 4 symbols. I do not know what am teaching. So, and I am trying to communicate this from a source to a destination, ok. And now suppose these are the 4 symbols that I give, and if these one of these symbols is say with probability 1, and if I transmit it, what is the information that this guy gets? So, this is assumed that is that sun is going to rise today. If I tell you this when you are sleeping in the night, what will you tell me [FL]? So basically are not gaining any information? Well it is a certain event you know this is going to happen?

Now, one of these events, suppose am going to say that this there is going to be a cyclone tomorrow morning. What is the probability of a cyclone happening? In Chennai almost one, but still it is a very rare event. So, if I tell you something which is very rare that message has a very high information content. So, if event which has a very high probability has a very low information content, and an event which has a very low probability has a very high information content? So, you can measure the information content of an event.

So, so the point is that what you can have is that the information content of an event, you can write it as, how many of you this how many of you have seen this before? Ok, all of you have seen this right. So, this is the value associated with an event, ok, now can you tell me what is the expected information content? For every event now I have given you the value associated with that even. So, what is the expected information content? Summation p of I into information content of I, and this like and this is of course, log right. So, it would be so, what is this called? This is called the entropy.

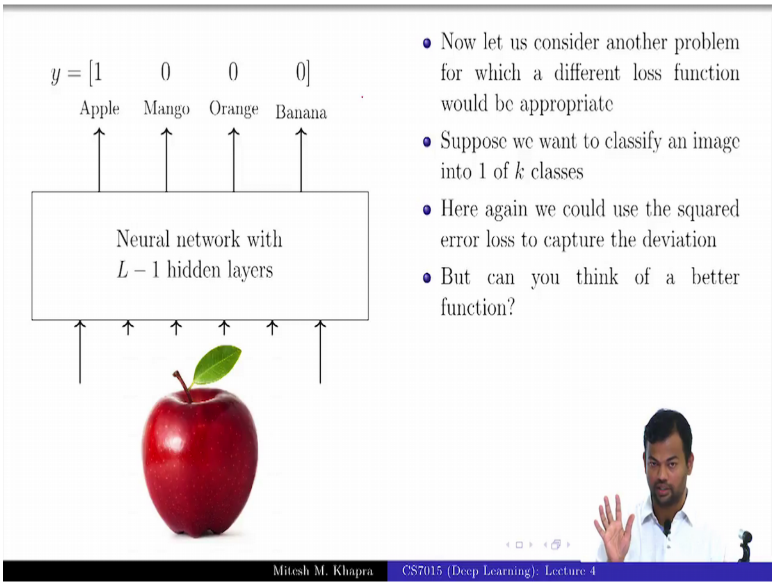
Now, what is cross entropy? How many distributions are you dealing with here? 1 which is the p distribution, which tells you how likely these messages were, and based on that you are trying to calculate the entropy of this situation. So now, what is cross entropy? You have a true distribution say you have a predicted distribution, this is what you predicted. So, that means, according to your predictions, the information content of every event is going to be log of qi, because that is what you predicted, but what are the actual properties which with these which these events are going to occur pis. So, then the expectation has to be computed over pis.

So, then what you will have is summation pi log qi. So, this is what you estimated the information content to be, but the actual events are going to happen with this probability, right. So, this is your value associated with the event, and this is the actual probability of the event. So, this quantity is known as the cross entropy, is it clear? And this is a way of measuring when would this be in when would this be minimized? When both are same that means, if your prediction is very close to your true distribution this quantity will be no minimized actually.

So, that is what we wanted actually, you wanted to predict some distributions in all of these cases, and you wanted a measure which tells you that this prediction was good, and what is the definition of good? It is as close to the correct value. So, cross entropy gives you a measure of telling how close a predicted distribution is to a true distribution, is that clear.

So now instead of using the squared error which was actually pi minus qi right. So, pi was my true distribution, and qi was my predicted distribution I can use cross entropy which is given by this model. And it does the same thing it gives me a principled way of measuring how close my predicted distribution is to my true distribution, do you get this?

(Refer Slide Time: 16:46)



$y = [1 \quad 0 \quad 0 \quad 0]$
Apple
Mango
Orange
Banana
Neural network with $L-1$ hidden layers
Mitesh M. Khapra
CS7015 (Deep Learning): Lecture 4

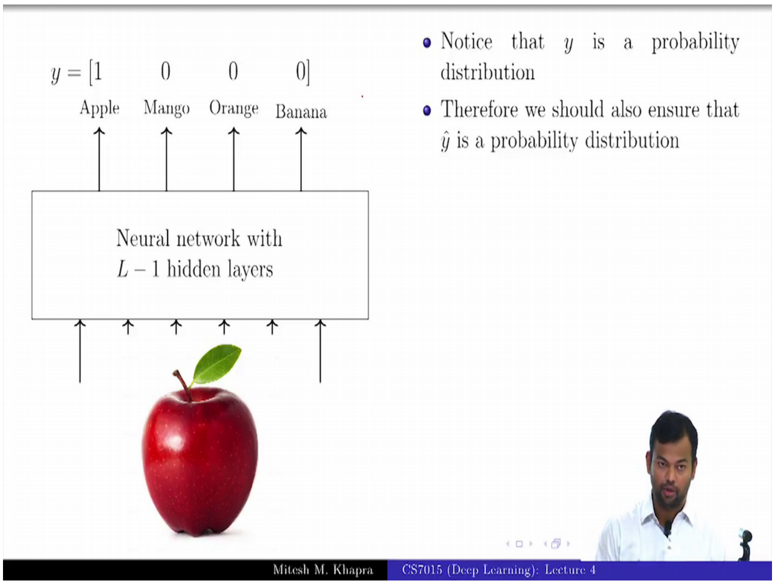
- Now let us consider another problem for which a different loss function would be appropriate
- Suppose we want to classify an image into 1 of k classes
- Here again we could use the squared error loss to capture the deviation
- But can you think of a better function?

So now so, this was for whatever we have done so far, right till this point this was for regression, right, now I wanted to enter into classification, for which I have built this set up of how to take the difference between 2 distributions. So now let us consider this problem where we have this situation and which is a classification situation, that you are given 4 possible classes out of which one is the correct class. And this is the true data given to you this is the true distribution, all the probability mass is focused on one of these classes.

Now, we want to given an image classify this into one of k classes, if you could again use a squared error loss, but since we are dealing with probability distributions here, we want to use something special. So, before we get to what the special is going to be, what do I first need to tell you? In the earlier case my output was not bounded was it also dependent, was there any condition on if the IMDB rating is something the critics rating should be something else or the IIT rotten tomatoes rating should be something else, no.

Now, in this case is there a tightly coupled behavior between the outputs, why? Because they should sum to one, we are trying to predict a probability distribution. So, the sum should one right. So, I need an output function which ensures this, you get this header?

(Refer Slide Time: 18:08)

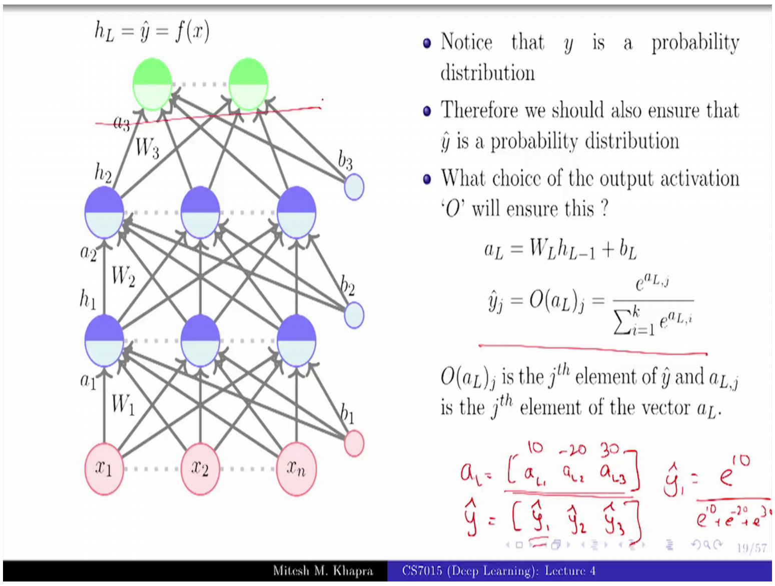


- Notice that y is a probability distribution
- Therefore we should also ensure that $\hat{y}$ is a probability distribution

Mitesh M. Khapra CS7015 (Deep Learning): Lecture 4

Now, we should ensure that y hat is also a probability distribution, whatever we are predicting is also a distribution. So now, can I use a sigmoid function, yes, it will give me values between 0 to 1 and probabilities are between 0 to 1. But the sum would not be y so, sigmoid is ruled out.

(Refer Slide Time: 18:32)



- Notice that y is a probability distribution
- Therefore we should also ensure that $\hat{y}$ is a probability distribution
- What choice of the output activation 'O' will ensure this ?

$$a_L = W_L h_{L-1} + b_L$$

$$\hat{y}_j = O(a_L)_j = \frac{e^{a_{L,j}}}{\sum_{i=1}^{k} e^{a_{L,i}}}$$

$O(a_L)_j$ is the j^{th} element of $\hat{y}$ and $a_{L,j}$ is the j^{th} element of the vector a_L.

$$a_L = [\overset{10}{a_{L1}} \quad \overset{-20}{a_{L2}} \quad \overset{30}{a_{L3}}] \qquad \hat{y}_1 = \frac{e^{10}}{e^{10} + e^{-20} + e^{30}}$$

$$\hat{y} = [\hat{y}_1 \quad \hat{y}_2 \quad \hat{y}_3]$$

19/57

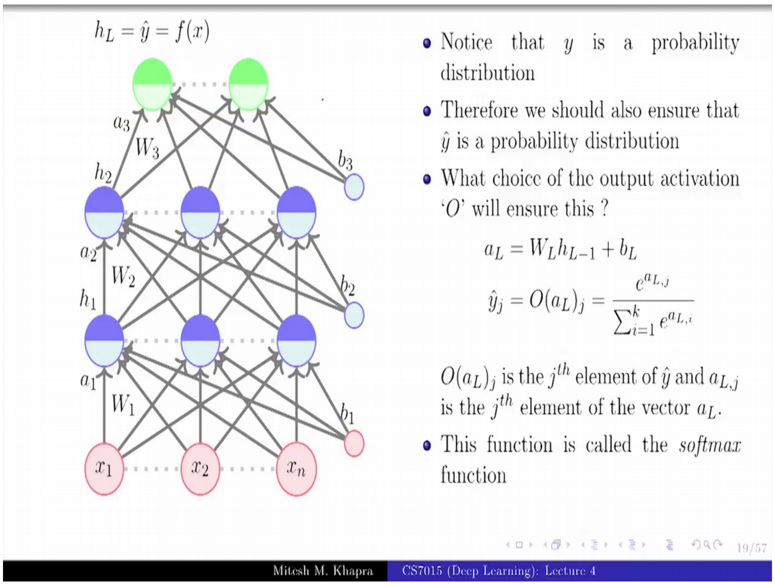
Mitesh M. Khapra CS7015 (Deep Learning): Lecture 4

So, what we use is something known as the softmax function. How many if you have seen this before? Please everyone raise your hands, otherwise you will get 0 on the assignment, fine. So, what does this? What does this function actually do let us look at this function right? So, here you had a L which was say a L 1, a L 2, a L 3, right suppose we had 3 classes, ok. So, from here I actually want to go to hL or rather I going to want to go to y hat, right; which is going to consider y hat 1, y hat 2, y hat 3, right it is going to give me probability of each of the 3 classes.

Let us assume there are only 3 classes right. So now, what this function does is, how is it going to predict y 1 hat, suppose these values were 10 minus 20 and 30. So, what is going to be y 1 hat is going to be e raised to 10 divided by e raised to 10 plus e raised to minus 20 plus e raised to 30. So now, you see how the output is comp computed from each of these values. So, why did we do this e raised to stuff why could not I have just taken 10 plus minus 20 plus 30 divided by the sum, because we have negative values.

So, once we take the exponent even the negative values become positive, right. So, that is why we need the softmax function, I hope all of you wrote this in your assignment, they did ok. So, you get this we have a different output function now, and this output function does it make sense? It gives us a probability distribution now the summation would be 1, and each of these values would be; between 0 to 1, that is exactly what we wanted.

(Refer Slide Time: 20:24)



- Notice that y is a probability distribution
- Therefore we should also ensure that $\hat{y}$ is a probability distribution
- What choice of the output activation 'O' will ensure this ?

$$a_L = W_L h_{L-1} + b_L$$

$$\hat{y}_j = O(a_L)_j = \frac{e^{a_{L,j}}}{\sum_{i=1}^{k} e^{a_{L,i}}}$$

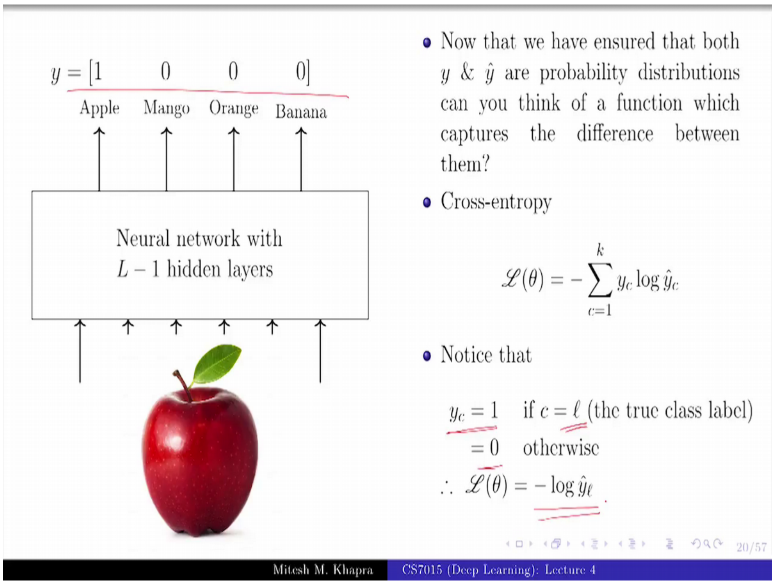
$O(a_L)_j$ is the j^{th} element of $\hat{y}$ and $a_{L,j}$ is the j^{th} element of the vector a_L.

- This function is called the *softmax* function

19/57

Mitesh M. Khapra CS7015 (Deep Learning): Lecture 4

(Refer Slide Time: 20:25)



$y = [1 \quad 0 \quad 0 \quad 0]$

Apple Mango Orange Banana

Neural network with $L-1$ hidden layers

- Now that we have ensured that both y & $\hat{y}$ are probability distributions can you think of a function which captures the difference between them?
- Cross-entropy

$$\mathscr{L}(\theta) = -\sum_{c=1}^{k} y_c \log \hat{y}_c$$

- Notice that

$$\begin{aligned} y_c &= 1 \quad \text{if } c = \ell \text{ (the true class label)} \\ &= 0 \quad \text{otherwise} \\ \therefore \ \mathscr{L}(\theta) &= -\log \hat{y}_\ell \end{aligned}$$

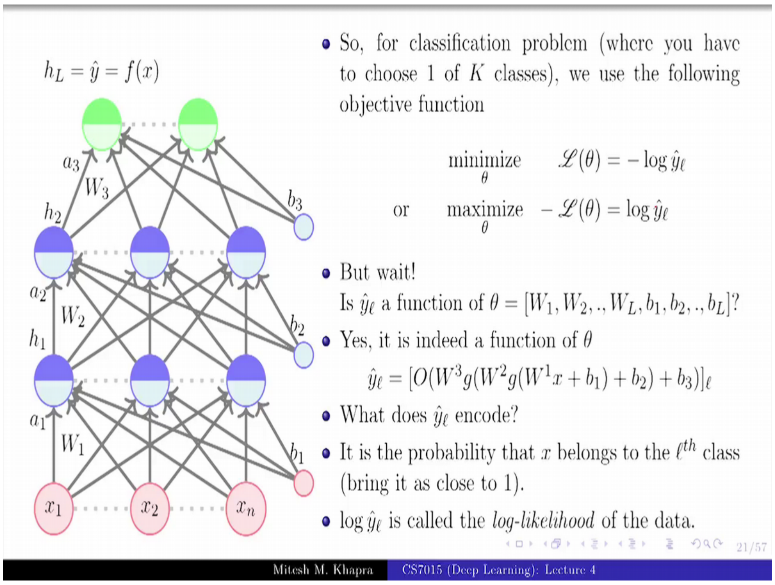
20/57

Mitesh M. Khapra CS7015 (Deep Learning): Lecture 4

And now that we have ensured that y and y hat both our distributions what is the objective function that we are going to use? Cross entropy, how many of you convinced it is cross entropy? We have 2 distributions now, we saw that a principled way of computing the difference between 2 distributions is the cross entropy so, we will use the cross entropy.

Now can you tell me something about this sum, there is something special about this sum. What are these? 3 true values and these are the predicted values, ok. What is so special about this sum? How many terms are there in this summation; k as many as the number of classes? In this case 4, how many of those terms will go to 0? All but one, right except for the correct class everything else will go to 0. So, this just boils down to the following loss function; that if l is the true class, right for that class yc is going to be one it is going to be 0 for everything else, that is exactly what this vector tells you only that term will remain. So, were actually trying to minimize this quantity.

(Refer Slide Time: 21:35)



- So, for classification problem (where you have to choose 1 of K classes), we use the following objective function

$$\underset{\theta}{\text{minimize}} \quad \mathscr{L}(\theta) = -\log \hat{y}_\ell$$

$$\text{or} \quad \underset{\theta}{\text{maximize}} \quad -\mathscr{L}(\theta) = \log \hat{y}_\ell$$

- But wait!
  Is $\hat{y}_\ell$ a function of $\theta = [W_1, W_2, .., W_L, b_1, b_2, .., b_L]$?
- Yes, it is indeed a function of θ

$$\hat{y}_\ell = [O(W^3 g(W^2 g(W^1 x + b_1) + b_2) + b_3)]_\ell$$

- What does $\hat{y}_\ell$ encode?
- It is the probability that x belongs to the ℓ^{th} class (bring it as close to 1).
- $\log \hat{y}_\ell$ is called the *log-likelihood* of the data.

21/57

Mitesh M. Khapra CS7015 (Deep Learning): Lecture 4

Let us see so, for classification problems this is your objective function. You either minimize the negative log of y hat l or you can say you are maximizing this thing, ok. Now what is this quantity y hat l? No, it is a predicted probability of the correct event. So, this is a probably no wait this is an important question. So, you have y hat l here, and this is a function of I mean this optimization problem is with respect to theta. Is this a well formed objective function? Does y hat l actually depend on theta, yes, it does.

So, theta because why I tell is a function of all these things, everything here, and then a log on top of that, right. So, it is actually a function of all your parameters. So, this is a properly set objective function, we are trying to minimize or maximize with respect to theta, and you told me that y hat l is actually the probability of the predicted probability of the correct class. Hence, this quantity is also known as the ml class pattern recognition class log dash of the data.

Student: All (Refer Time: 22:53).

All good and fill in the blanks.

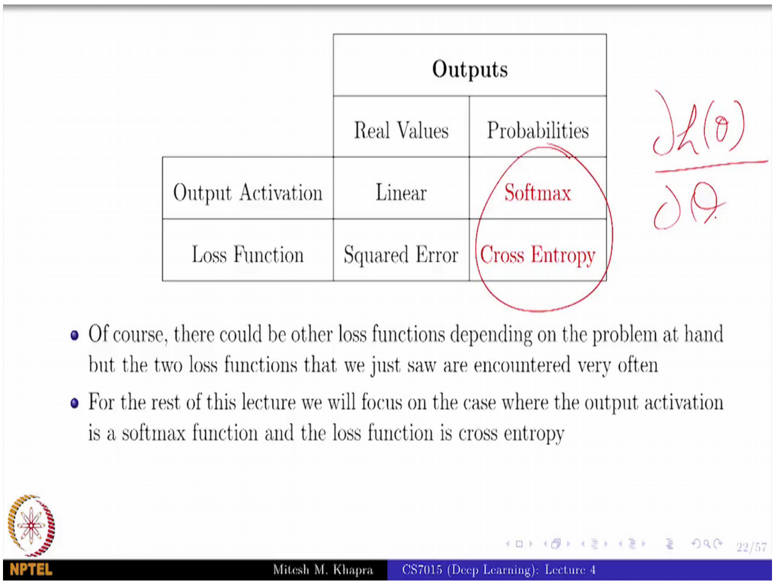
So, it is a priority of the x belonging to the l th class and then hence y hat l because it is the probability, it is the likelihood of it is called as the log likelihood of the data log likelihood, is that clear?

So, what have we done so far? We started with a feed forward neural network, we defined the hidden layers and the input layers and the weights and the biases. We kept a provision for the output layer to be something special, right? Then we went to 2 classic problems, one is regression and the other is classification. In regression we wanted to predict values of all sorts of ranges.

So, we decided to use a linear layer there. So, that there is no bound on the values that you can predict, and your objective function should take care of where the bound lies. It should not allow values which are way off from the true values, right? And that is why we use the squared error function. There the other problem that we looked at was classification; where we saw that the label actually can be treated as a distribution, where all the mass is focused on the true label and 0 everywhere.

And our job is then again to predict our distribution. So, we are given the true distribution, and we predict another distribution. So, the output again we want something special in this case which is a distribution. So, to ensure that use a spatial function which is called the who said sigmoid softmax function, fine? And then we got a prediction which is a probability distribution, and then how did we find? What was the objective function what is the difference between the true and the predicted the cross entropy, right? So, we use cross entropy as the objective function, and then with some simplification we realize that it boils down to maximize the log of the probability of the true class, or other log of the predicted probability of the true class.

(Refer Slide Time: 00:53)



	Outputs	
	Real Values	Probabilities
Output Activation	Linear	Softmax
Loss Function	Squared Error	Cross Entropy

$$\frac{\partial \mathcal{L}(\theta)}{\partial \theta}$$

- Of course, there could be other loss functions depending on the problem at hand but the two loss functions that we just saw are encountered very often
- For the rest of this lecture we will focus on the case where the output activation is a softmax function and the loss function is cross entropy

22/57

NPTEL Mitesh M. Khapra CS7015 (Deep Learning): Lecture 4

So now let us look at the summary. So, if your outputs are real values, what is your output activation going to be? Linear, what is the loss function going to be? Squared error if your output is a distribution what is the output function going to be? Softmax, what is this loss function? Squared error cross entropy. Right now this grid light actually takes care of a wide range of problems that you will see right think of any examples that have been giving you so far; movie prediction or sentiment prediction or image classification or anything, all of that you can fit into this frame of it.

And so, if you know these 2 loss functions how to deal with them, then you can deal with a large class of problems that you are going to deal. And for the rest of this lecture which will happen tomorrow we are going to focus on this. At this particular output function and this particular loss function, how do we compute? I have a loss function, what I am going to compute now the gradient with respect to all the parameters.

So, this is what we are going to focus on. So, we have seen the loss function in detail, we have seen that the loss function is tightly coupled with the output function. Now we are all set, but given this loss function how do we start computing gradients on this classification.