Go back to this, so a regression task for example, right in 1D I mean let us just go back to that same example, right which we actually mentioned, right. So just as you have a movie let us say, right and then you want to be able to say, right you want to be able to give sort of rate rating to that movie. Rating could be you know could be, we have so many organizations right that do a rating I mean I do not know, I have some names here, some like IMDB I think that is the you know internet movie sort of a database and then you have some rotten tomatoes and all that I mean this is I do not know I am not an expert and all that but yeah right I mean you can take examples like that. And what you have is the input right like I said could be who is the actor right, who is the director, maybe who is the who is the guy who composed the music whatever right I mean you can put all of that right. And then what you want is a block right I mean you know which takes as an input and this by itself could be an MLP right it has several layers and then the output right I mean you know you want right you want to ask for example, right I mean in terms of the IMDB rating what would this mean for a certain rating IMDB and then what would it mean with respect to another agency how would it rate it and so on. Now in all of this right you need you need something to be able to train and for example right this could be any number between let us say 0 and 10 or something that somebody just rates it as 4.

5, somebody gives 4.6, somebody says 9.8, somebody gives 9.95 right any number you can give it does not have to be a probability or anything right it is simply a rating.

So, somebody rates the movie now what you can do is you know you can actually you in all of this right in order for example, tomorrow if some movie right you get access to and you know the actor, director and all of that. So, if you were to if you have to put that as an input if this MLP had been trained right then what you could do is you could just take the new information that you have and then ask right what would be what would be the IMDB rating and all and maybe if these people actually rate it right you expect that what your see network tells hopefully is will come close to what let us say these agencies will eventually rate it. But for training you need all that information right. So, you need to be able to go back dig up you know movies for which ratings are already available and for each one you look at who is he who was the actor who was the director who was the music composer who was the producer whatever it does not matter right. So, all that you put in and then you say what was the IMDB rating what was the ROT RT what was it rating and then what was some other right I mean a critic rating maybe.

So, you can have all those rating values and then what do you do. So, for example, if you think of this as your Y then over examples and examples right I mean you cannot train a network with just one example right I mean you have to give several movies for which you should have this information available right. So, that is why that is where this training thing comes in right you have to train the network that means you have to tell this network that hey look right we have this information with us already that you know if these are the people if this is the kind of thing that you get then the output rating rate turns out to be this and similarly for the other movie and so on right. So, what you really want is a prediction of these values right why that means this is rating which could be a value let us say anything like between 0 and 5 or 0 and 10 or whatever it is and you want the cost to be such that let us say right if I have L examples right of these okay that means I have L movies and let us say Y okay what is the dimension let us say Y is some let us say right you know m dimensional let us say critics and whatever right it is some say m dimensional output then what you really want is this number right or this kind of a value right the scalar value which is like Y hat Y hat of let us say i comma j minus Y of i comma j squared right and summed over let us say i goes from 1 to L and then j goes from let us say 1 to m right and you have got let us say right 1 by L or 1 by L into m or whatever it when you can at least get it right whichever way you want. Now what you want is for every i comma j right you want this you want this number to be as small as possible so that so that a prediction right happens well and if you can if you can train a network such that right so something like this called a loss function right.

So the loss function whenever we say loss right typically you want we want loss to be a small quantity right nobody likes a loss to be big right so therefore what we will do is we will say that right this needs to be minimized right minimized with respect to what right minimized means what minimize with respect to something right and what is that something that is something will have to be the weights and the biases sitting in the MLP right. You want to sort of you want to what you say right you want to be able to arrive at these weights and these biases such that such that right at the end of the day this cost right which are which are looking at tends to be as small as possible and then if you try to do that then let us hope that right you have finished training and you are actually happy okay with how you have trained it does not always mean that you have done a good job just because this error has come down there are issues like over fitting and all that but let us kind of stay away from that for the time being. Let us just assume that right you have done the training and then and then the idea is that right once this is over then give me a give me a new movie for which you have some information I will put that in and then my then instead of you telling me I will have this network tell me what kind of a what kind of a rating right it might have okay that is kind of a regression problem. Again typically for images and all there are so many of them right which come under which come under such a such a task. The other thing is actually so right this comes under a regression problem.

The other one right would be what is called what is called a classification problem classification where you could have you could have a distribution right which are which are trying to match. So it is like saying that right I am trying to build a network such that let us say let us say that I mean I have I have a true sort of a distribution which is let us just take a take a right simple case true sort of a distribution which looks like let us say let me say that right that my y is some 0.3 for some class for some label 0.2 for some other label and then 0.5 for some other label.

Let us say that these are probably right and what do you call I mean so this is an urn with some balls let us say right. So I have got red balls with this probability let us say blue balls with this probability y with this probability right. Now what I want is I want is a network right which can actually take examples right of the kind that I have and be able to actually get to a y hat or q or whatever it I mean you want to call this y hat y hat it is such that such that y hat should be actually as close as possible to y that means I am learning this one a distribution right. So I want to know like know right I mean you know how many of those of those red balls are there how many of those white balls are there whatever right blue balls yellow balls and so on. And let us say that let us say that right I mean again right now all of this loss function is very very important.

It is not always true that right I mean every time you can have just a loss function like this and walk away okay. I am just giving the simplest of all right which is like an MSE or something okay for the earlier example regression. But then sometimes right depending upon what kind of a network you have you have other losses of the kind sometimes they are called adversarial losses and so on right. So it does not always mean that I mean so this could be this will typically be one component of a losses there could be others okay. You can have a sum of different losses and then you have to weight them okay because you cannot equally weight them then I mean if you have multiple losses coming in which is also possible.

But then what you will have to do is you will have to carefully weight each one of them and then those are called hyper parameters. Because right you do not learn them typically right you do not learn them typically you just play around and find out right I mean on which of which one of those parameters work best for your task. Now in this case let us say that let us say that I have a network which gives me an output I mean right while I am training right I get a value like I just have an example here. So 0.35 so let us say 0.

65 this is what I get. Now I want to compare it how well have I done right this particular this one a network that I have which seems to have done the prediction of Y and it is given me a Y hat. Now I have some numbers there right which of course sum up to one as they should. Now I could then ask right now is it okay to do the same thing which I did earlier right can I just do some Y minus Y hat right for every element wise can I take the squared error and sum them up right. You could do that right nobody stops you but then the thing is right because of the fact that you know there is a difference between that and this because this being a probability sort of distribution right you want to do something you know which is more which is more you see aligned with that right.

So you want to kind of make use of the fact that you know there is kind of information content and so on right and how you do that is as follows. Okay now let us say that let us say that it for an event A with a probability let us say P of A for an event A with probability A with probability sorry P of A okay. If you if you want to if you want to kind of look at what kind of information content right it has then the information content right you can write the information content suppose you indicate this as information content as let us say would have I of A right okay. So this is the information content in A right this you can typically give as minus log of P of A right. What this actually means is that I mean if it is a certain event right then there is no information like what is already let us say P of A is 1 it is a certain event right then in that case it really there is no information that is the right known that you can get out of that.

So if A is a very rare event right in which case in which case P of A is a very small number right then you sort of think that the information probably very high it is like saying once in a while once in a while right a tsunami comes right probably very low but then that the fact is that is like high information right and therefore this information content and this minus right as you know right is there source to take care of the fact that P of A is between 0 and 1 right therefore you want the information content to be high right for values less than 1. Now what can happen is right so if you have let us say multiple events right now if there are any events okay so if you want to look at the expected information right if there are any events let us call them A1, A2 all the way up to let us say An with let us say a probabilities let us say P1, P2 all the way up to Pn then the right expected information content or what is called the right expected values you all know expected information content will then be what according to you what would you write expected information content that means the right expected value of I of A right over all the events will be summation Pi. Minus log p. Yeah so in simple terms I of Ai right i going from 1 to n which is summation which is minus summation Pi and then log Pi right I mean P of Ai is Pi okay so log Pi so i going from 1 to n right and this term right is called actually entropy right for those of you who come from information theory background all right I mean you know that right this term is called the right entropy and what you are trying to do right in a sense is right I mean so Pi is actually a true probability right and if you actually if you actually write you know if you think of a term like this where you have got minus some minus summation Pi log Qi right where you are trying to approximate or in that example right we had like you know Yi and Yi hat right similar to that right if you had Qi and you are kind of trying to push Qi as close as possible to Pi because Pi is actually the right true this one probability and you want Q to Q to look as close as possible to P then this one right again of course i going from 1 to n see for example if you had let us say 2 if you had just 2 classes then then then right I mean you know then you know in that case you will have like you know P into let us say I mean if you had if you had you know let us say write P1 as P then you are see P2 would then be say 1 minus P and similarly write Q1 would be Q and then write Q2 would be like you know 1 minus Q right but then typically you will have like n number of classes and such a thing is called actually a cross entropy this is called a cross entropy cross entropy and you can show that this cross entropy right is always so let me write this so cross entropy is always greater than or equal to the this entropy the proof for this is fairly involved okay but you know a shortcut right which if you are willing to accept something then a shortcut is this there is something called a KL at the divergence okay KL divergence okay we have we will see this at a much later time but this is called kulbach-leibler divergence okay kulbach-leibler divergence and what this just like just like write I mean you know you try to I mean you try to analyze it through a mean square error you try to find out whether let us say 2 functions are similar or not similarly this this gives you a notion of distance between 2 distributions okay distance between 2 distributions. So you I mean if this number is smaller it means that right they are really close distance between 2 distributions so for example in this case I can write kulbach-leibler between P and Q and the way okay this is just a symbol okay with a with a 2 what you call 2 lines okay in the middle okay and this is typically well it will be you can have something like summation P of X log P of X by Q of X and this log can be anything okay to the base that is up to you whatever you want to choose typically it will be a natural log and okay which then means that right this is like P of X okay summation of course log P of X minus okay and this also right the summation Pi log 1 by Q I can write it in whatever form you want so here it is like minus PX log Q of X right.

And this kl is always right is a quantity such that right it is again a number by the way right so it tells you how close the 2 are and this is actually if you accept this that the scale I mean it is always going to say right a greater than or equal to 0 then write clearly PX log PX is actually greater than or equal to PX log QX right and therefore if you take the minus right on either side right then you have minus PX okay of course on a summation and all is there okay PX log PX then be less than or equal to minus summation PX log QX okay and this is your entropy and this is your this is a cross entropy right. So this cross entropy is always a number greater than or equal to the entropy okay and the idea is that when you try to when you when you try to write minimize okay so for example the loss there was something like a mean square error when you had a regression task right. So here typically in a classification problem you will try to minimize minimize minimize the cross entropy loss the cross entropy loss okay that is why the loss functions and all right will change okay when you go from one task to another I mean if you have a regression task and the cost is typically a different kind of a different kind then minimize the cross entropy loss okay. So in a classification problem right this is what you would have and as you can see right the first term really has no effect I mean this is because P of X log P of X right that is really a true probability right. So a network does not really right it does not matter okay what that number is because it is already a given number whereas what really Q depends upon so the second term right is what really matters right in this if you look at it it is a second term that matters because your network has to actually do a prediction of Q right.

So therefore the second term is actually cross entropy and that is the one that is the one which you wanted to be as low as possible because the lowest rate which you can take is the value like you know P of X log P of X right at which time your KL divergence becomes 0 that means your you know Q and P are almost identical right identical in the in the right pullback Leibler sense okay in that sense okay. And therefore if you if you see people keep on talking about you know cross entropy loss cross entropy loss right and that is when they talk about a classification problem right. Now let us say that I have these loss functions right whatever it could be a regression problem it could be a classification problem right.