Gradient Descent

So, yesterday we were at this point where we were actually discussing the weight space symmetry right, I was telling you that you can swap the neurons and the weights and you will get the same cost, which means that any of those solutions is okay for that matter. Of course, you know these are all represent, I mean these are just simplistic figures right, it just allow you to get a hang of what is going on, nobody knows how the actual landscape is, especially in a high dimensional space. So, that is why yesterday right what I showed you was something like this right, which was to show that, I mean so this cost right that you are looking at along the Y axis versus the unknowns, which you are trying to estimate theta, like the weights and the biases right. So, you can have multiple places right where you hit the same local minima. Therefore, any of those solutions is equally valid, the reason why we are saying, why we wanted to bring this up is because just to know that when you are solving for a cost function right, so all this is an optimization problem right at the end of the day. So, when you are doing that, one is not asking for a global minimum or something okay, because one does not even know where that is and how to get there.

And till now right we have not seen a systematic way of even solving for the loss right, the loss could be as I said right depending upon the task, it can be a regression problem, wherein maybe you have something like a mean square error as a cost, it can be a classification problem in which case it could be a cross entropy kind of loss. Till now we have not seen a systematic, the PLA was just so what just very simple, very simple case right. Now today right we will try to look at what is called a gradient descent, which is the accepted sort of a vehicle for doing optimization of you know in such a high dimensional space. It does not mean that right it is a best out there, just that right it is mathematically air well, also say computationally wieldy, otherwise you know you can have unwieldy, so you know unwieldy approaches, I do not know how many of you have heard about things like simulated annealing, graduated non-convexity and all, also those are again you know methods that can guarantee you know a good solution but then they would just take you know too long to try out okay and one does not even get there.

So if you see right most of these, most of anything that you read right will employ a gradient descent okay and a gradient descent is the simplest way to understand it right is to sort of you know take an example. Suppose I had right suppose I had a convex surface like this right now. So for example right so the idea is that right you want to be there right that is your kind of optimum solution and that means this is the unknown theta right for which the cost is this is the loss, you do not have a convex loss like I said but let us say if you had it then a gradient descent would be the most ideal thing to use because then the idea is that wherever you start from because your initial guess could be anywhere. For example I randomly initialize my weights and biases right and suppose I start from there okay then if you look at the look at the gradient equation right the gradient descent, let me write that down the gradient descent we will call this Gd. So the gradient descent equation will look like this I mean so n plus 1th iteration that and it is an iterative algorithm okay it does not take you directly to the global minimum because you do not have an analytical gradient and

all that.

So theta n plus 1 is equal to theta n minus I will explain it what these things are let us say dou L theta or okay let us just say dL theta by d theta theta equal to theta n and this can all be vectors okay in which case you are looking at you know looking at a kind of a vector form or simplest take a scalar form right and if you see so this alpha is called the step size okay and there are ways to which we will see later again right this is going to be a very quick sort of review on deep learning. So I would not go into the details that we normally go into right when we take deep learning as such but just to suffice to say that the step size is something very important right if you overstep right you could land in trouble if you take too small a step size right you could take forever so it is alright so the step size is an important thing it is of course a number you know alpha is actually larger than 0, greater than 0 and okay and so the idea is this right so for example if you are here okay which means that you have you know a negative gradient right I mean suppose I start here on the curve so at that point I have you know a negative gradient and that I mean right a negative gradient if you see that negative of then minus of that right then that gives you a positive value and what this means is that if I am here let us say if this is theta n then I would actually head towards theta n plus 1 which will actually take me forward like this right in this kind of direction towards the minimum. If on the other hand let us say if I am here if that is my theta n to begin with then because you have a positive gradient right at that point negative of that right will actually mean that your I mean if your theta n is here then your theta n plus 1 you will right it will then take you right other this one direction. If you come from here then it will it will kind of right take you forward towards the minimum if you are on a positive gradient negative of that it will then kind of try to so the idea is that you want to eventually get to get to this place right so you can take small small steps and get there right that is the idea so theta n plus 1 you will be here then again right you might then your theta n plus 1 okay then this theta n becomes theta n plus 1 then this n plus 1 it will then become the next n and then right and then you can imagine if you kind of right if you do this do this continuously this is right iterative so nth iteration you go on and on and on the hope is that right you will kind of hit that. Now this is okay if you had a convex surface right but as I said okay our landscape right is not this is not something right which is going to be as neat as this but gradient descent was actually typically meant for meant for right these kind of surfaces and but then what what turned out what has happened is that this is the one that that they use for deep networks also okay but those I mean they just this is happy right hitting a local minimum so what that means is that see your actual surface for example it could be let us draw something okay could be something like that we do not even know okay.

So if you are lucky right if you are lucky you might have started here right if you had started here just that you know somebody gave you some initials just by magic right you got parameters to start with from there then then you can hope to hit this point right through a gradient descent because you see you know it is always seeking seeking you know a direction such that the cost will will actually actually go down right it never it will

never allow for an increase in cost see for example right when we were here right it did not it did not kind of take you that way because that would have meant the cost would go up when you were here it did not it did not take you that way because that again would have meant that the that the cost will go up right so it will never so that is why it is called a greedy strategy right greedy in the sense it will always seek seek the basin or seek the lowest point. So in a way that sounds good right because then that is what we want but at the same time right it has its pros and cons I mean this is a one hand it is also a good thing but on the other hand right the bad thing is that if if I had started here unfortunately then there is no way that I can actually escape this I mean I will I will I will end up there and I have to I have to kind of accept the solution because I do not even know sitting there think of a hill think of a think of a think of a hill right well let us say you know multiple such mounds and you know sitting here I do not see right whether the other side is actually you know more whatever right is it is it more down or is it more up I cannot see unless I climb right so so if I climb here then maybe right I can see that oh right there is a lower point there but then sitting here and when the and then when you are not allowing me to increase the cost right I will never end up there. So I can never see what sorts on the other side right so and again it will no so it could also happen from this direction and whichever direction from wherever you are coming it can happen that you end up in some so right that is called a that is called a local minima because the global is actually here okay but then as far as as far as deep learning algorithms are concerned as far as deep learning network is concerned as I as I told you there is nothing like a single minima or something right because you can have as many right the lucky part is what has turned out to be lucky is that any of those looks fine which is which is arguable right in the sense that it but then we have to go with empirical evidence right nobody has been able to show that what you have got is is is the best or you know is a global minimum you cannot you cannot prove any of that. So when you have you know a million parameter space 60 million for example AlexNet I think has close to 60 60 million unknowns so where in that space right how do you even know where you are but it turns out that you know that that even a local minimum is kind of acceptable and you are not you are not really worried about it and and and the and this local minima can also can also come in various forms something like that it could also be something like this we do not even know right it could be it could be a combination of all that. Now right you might then right then kind of one wonders that okay so right so let me first write down the pros and cons right then I will again again come back to this equation.

So let me just simply list the pros and cons of of gradient descent right so what it can do is it can it can it can actually guarantee it can guarantee only a local a local minimum. So for example right if you start from some other point as initial point you may end up somewhere else right your weight per weight configuration could completely change as it follows you know a greedy strategy as it follows a greedy strategy. Then point number two is that the step size has to be carefully chosen the step size that is that alpha okay the step size I mean the alpha you know each one will call it by some name we will call it alpha has to be has to be carefully chosen else else it can it can zigzag what does this right zigzag mean is this see for example if suppose suppose I suppose right go back to this and and you

know what can even happen is something worse can also happen right one thing one thing one thing is that I mean you know you know if you have chosen chosen you know a big step size right you may actually jump from here to there then I think you are right you may just you may just be right zigzagging there because you are kind of right you are kind of I mean you have to take a small step size right otherwise you will jump over to the other side then you will come back to this side then again jump over that is what they mean by zigzag but zigzag does not just mean that see for example I mean if your surface was let us say right what to say yeah right so it was so so right if it was something like that then what can happen is right depending upon the slope it does not have to be symmetric right so what can happen is if you have a certain slope here right which is of course you know a negative slope therefore right you will go there you will take a step size but okay now yeah right so I do not have to draw it exactly like this okay aha okay so I had something like that right okay now what I could do is you know I could actually end up there okay now if the slope here is higher and because my alpha is still the same right I could I could actually I could actually end up being there and then from there I could end up being there and then from there right I might I might I might just go out completely right it can happen no so this zigzagging it does not mean that you are actually contained within that basin or something even that is not guaranteed so so the step size can be such that you keep oscillating and then beyond a point you just just go out right so all of that can happen okay so that is why the step size has to be carefully chosen okay and does not by itself differentiate yeah well okay that is fine I think you know different initial values so the other thing is different initial values can lead to a different can lead to a different local minima lead to different local minima correct yeah because as I said depending upon where you start with you might you might end up with a weight configuration that does not even match your first one but as I said it does not seem to be a problem because people are happy with the final the empirical evidence shows that it is not at all dependent upon how you start you can have any random initialization and go then as another point all functions involved should be yeah right this one is important all terms are all functions involved should be should be actually differentiable all functions involved should be differentiable because you have that you have that dl by by d theta right differentiable correct so so in a way right there are kind of see pros and cons like I said and but it has been found to work okay now one can ask right so so one can ask a question that you know why did we choose that to be I mean intuitively right when I showed it it seemed to intuitively make sense that you should had you know you should take a negative of you know the gradient and move so it is though that is why you have that minus alpha dl by you know d theta but there is a more formal way to to actually understand that why you take a step like that and why that step size is like that so ideally right you can you can think of it like this right let us say that I have theta n plus 1 I mean theta n right that is my that is the set of parameters I had in the previous iteration then I can say I want to do plus alpha delta theta right so I want to move by a delta theta amount now which way should I go right I mean okay right that is what you want to you want to ask right but you change my change my say parameters but I want to change it in a way that it is actually you know it should be a meaningful way to change change my delta theta right because I cannot just arbitrarily add something and move on right so I have to have a principled way to actually do it okay

so so what is a principled way so what is a principled way to choose principled way to choose this guy to choose delta theta okay a gradient descent is actually a principled strategy okay and that I thought we will just show okay. So you can so let us let us kind of read look at let let write U be a U be a unit vector right which tells unit vector which which tells me the direction to move because the magnitude is anyway 1 direction to move that means right what it means is I am at a certain point on that surface right I want to know which way to go so so in a way right what you can do is you can examine right examine right U transpose U transpose the gradient of theta now the gradient that I am going to write this now as this symbol okay this I think you are all familiar with right the top triangle is delta the inverted triangle is gradient right gradient with respect to theta of right L theta so in a sense right you are asking if my if I have a gradient then which way should my U be right and this can be instead right post like rate minimize with respect to U this quantity U transpose gradient theta with respect to L theta right and I am putting this this gradient right so so this is actually a vector now right. So for example you can think of it as some function of let us say if it is f of X1 comma X2 then you are looking at dou f by dou X1 and dou f by dou X2 right I mean it is like it is like whatever it is probably very huge right but I am just saying okay so so in a sense right you have a number right and you want you and you want this one that right number to be a to be a minimum now this quantity right we know right I said it is all right so this quantity is basically nothing but so I can write this as minimization U with respect to norm U this we this we have already seen it norm gradient of of L and then cos let us say whatever it let us what do you say let us say cos gamma right where we will say right gamma is the angle that U makes with respect to a gradient of L right with respect to the parameter theta of course.

Now we know that this is 1 because we have chosen this one unit vector and this we know is actually a positive quantity right it is a norm and therefore right if you if you kind of want this to be a minimum right then then all of that seem to be centered around around this third term which is cos of gamma where gamma is the angle between U and so this is the angle between between U and U and the gradient U and delta theta not delta theta the gradient gradient vector right and when we and we know that right the lowest value right I mean because because we are doing this one a minimization so lowest value that let us say cos gamma can take is minus 1 right which then means that means that gamma should then be 180 80 degrees right which is which is which is why which is why which is why if you see if you if you if you saw that equation right it had minus alpha dL by right d theta so this so this negative of the gradient right is is basically you know it is coming from from there right because because because it is supposed to go exactly in the opposite direction right with respect to the gradient because gamma is the angle that U makes with respect to the gradient. Then the other thing right is not clear is what should be the step size right now it looks like I mean so it looks like which way I should go I know now but then the other thing is why let us say what should be that step size right I mean I know which orientation which way I should go but then how much should I should I go right so this delta theta we still want to see right what should that what should that be and here of course you know here here we saw that it is actually right dL by d theta but that also right one can kind of show as

to why why you why you choose that. So in order to do that right imagine that imagine that you have the cost cost function right which is which is L right and L of L of theta right this this is our cost. Now imagine that that I take a small step again a small step that means I take my alpha to be really small alpha alpha delta theta right I move forward by small amount. So if you do a Taylor series approximation of this right what will that be what will that be L theta plus alpha delta theta is something wrong or is it okay something wrong here something is wrong what is wrong I of course I have to write other terms but what is already wrong here this should be transpose right it is a vector no the whole thing should be a scalar this is a scalar valued quantity right delta theta is a vector right by delta theta I mean I do not mean a scalar okay it is like you know theta 1 theta 2 in whatever right so many weights so many biases so everything right so it is a kind of a vector I am not putting it as a vector explicitly right then what will you get after that the third term you are doing a Taylor's okay gradient square how do you write that so alpha square alpha is a scalar there is no point no no problem the alpha square by let us say two factor then what happens then what do you write exactly very good so what do you do so you have sorry delta not not not the gradient okay so delta theta transpose then I will just write this as dou square l okay no I think we are already using gradient square right so gradient square theta l theta and then delta theta right okay this is a Hessian this is the Hessian of right l theta so matrix containing the second order second order partial derivatives of l theta with respect to theta right so Hessian is the matrix of second anyway but then right this is not important right now second order partial derivatives so again right if you go back to that example f of x 1 comma x 2 it will be like dou square f by dou x 1 square then dou square f by dou x 1 dou x 2 dou square f by dou x 2 dou x 1 then dou square f by dou x 2 square right 2 cross 2 if you think about so matrix of second order partial derivatives okay but right now what we will do is we will actually ignore and then of course then you will have higher order terms of all that so right from the second order term right we will just ignore it because our alpha is very small right I mean all this Taylor series approximation is good if you write around around around a neighborhood right small neighborhood so this alpha right being small so we will so we will ignore all these terms okay ignore and simply retain right up to a up to a up to a linear term okay we will not take quadratic and all now if you look at l of theta plus alpha delta theta minus l theta that will be a change in the cost right because l theta is where I am already I have a certain value for l at theta and I am trying to move forward right which is my alpha alpha delta theta and this is nothing but alpha delta theta transpose gradient of l theta now what do you want this to be right and you want to move in a manner such that the cost should go down no right that means that means my l theta plus alpha delta theta should be less than l theta right that's what I want no I want my cost to go down right I want to move in a manner such that my cost goes down that means right that means you want this to this to be this to be a negative sort of a quantity right and maybe as small as possible right how much can you go but you want it to be a negative because then you are kind of going down whichever way whether you start from there or you start from here you want to move in a direction such that the cost will go down of course if you are doing gradient ascent then it will be right you can you have gradient descent you also have gradient ascent okay gradient ascent then you want the cost to go up okay but in the okay

let's forget about gradient ascent and all we are talking about a descent algorithm right a minimization right and therefore right what this means is that okay then this quantity right is what is what then then we need to appropriately pick out of which this is already fixed right this is simply a gradient and so which means I read the whole the whole control rise with what should be right delta theta transpose okay but the range of this quantity right but the range of delta theta transpose gradient l theta is such that is such that suppose suppose we said that the angle between them is let us say beta right such that cos beta is equal to again like delta theta transpose gradient l theta by norm delta theta norm gradient l theta and this is a number that is greater than or equal to minus 1 and less than or equal to 1 and so this so this quantity itself is such that if you take its ratio right with respect to its magnitude of the you know individual or norm of the individual vectors okay you get you know that right that's how that's how it is and therefore right if you choose okay choose delta theta to be equal to minus gradient of l theta okay if you do that right then then when what you find is at the right a numerator you will have you will have norm of delta l theta squared right in the bottom because of the fact that right norm of delta theta now delta theta is itself delta l theta therefore you have bottom also norm square the 2 will cancel off and then and then right you will be you will be left with minus 1 which is the which is the smallest value right that that that the cos beta can can take out there.

So if you pick right any other any other angle delta theta right then then this inequality right you won't be able to I mean you will be you will you will end up with a number where cos beta is actually right greater than minus 1. So gradient descent is actually a principled way right in which right given any arbitrary surface right you can take as long as as long as you are able to compute the gradient which is why there was a differentiable differentiable sort of condition right that came right in between. Of course there are there are there are other algorithms that don't even insist on that but I think gradient descent given that it is it is simple and and you know easy to implement. So all that it all that it you know requires is you should be able to have a differentiable function right and and you see and always right remember when you are talking about L theta if you are talking about you know deep network the L theta is coming at the end right that is at the output. Now this L theta right is going to be a function of so many things right in between I mean you have got your you know you have got the input layer then you have got the hidden layer 1 hidden layer 2 and maybe you have got 50 hidden layers and after that right you arrive at the output layer.

Now when you express L theta in terms of all these weights and biases right you you should not you should not you know you should not what you call you should not end up with something which is actually a non sort of a differentiable you know entity right somewhere somewhere right along the way. So which is why which is why that that is something right that that we have to kind of say take care of.