Optimization from Fundamentals

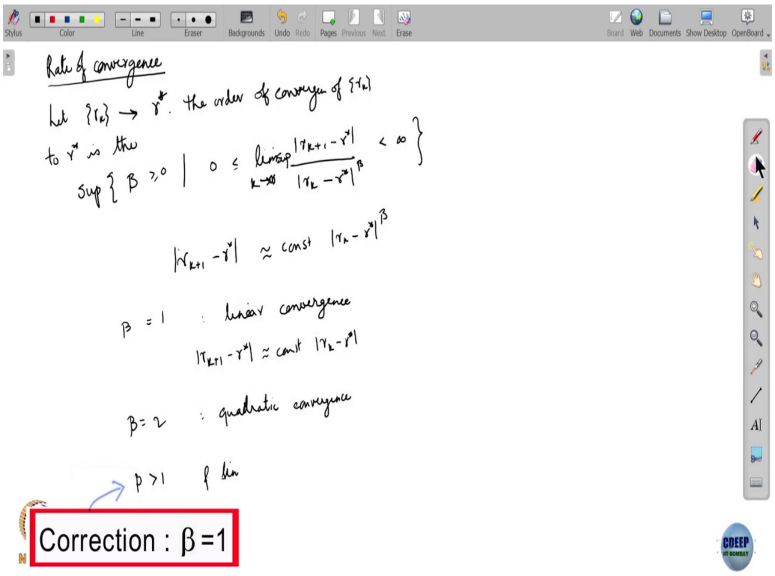
Prof. Ankur Kulkarni
Department of Systems and Control Engineering
Indian Institute of Technology, Bombay

Lecture - 20A
Steepest descent method and rate of convergence

Ok. Welcome everyone. So, now, we completed our study of this Steepest descent method or more generally line search methods in the previous lecture. So, what we will now analyze is the basic issue of how fast these methods can be ok; so, for we will now discuss what the rates of convergence or the speed of convergence of algorithms ok.

So, in order to do that we need to first have a notion or a definition of what we would call the rate of convergence right. So, let me define that for you to begin with.

(Refer Slide Time: 00:56)



Rate of convergence

Let $\{r_k\} \to r^*$. the order of convergen of $\{r_k\}$ to r^* is the

$$\sup\left\{ \beta \geq 0 \;\middle|\; 0 \leq \limsup_{k\to\infty} \frac{|r_{k+1}-r^*|}{|r_k-r^*|^\beta} < \infty \right\}$$

$$|r_{k+1}-r^*| \approx \text{const } |r_k-r^*|^\beta$$

$\beta = 1$: linear convergence

$$|r_{k+1}-r^*| \approx \text{const } |r_k-r^*|$$

$\beta = 2$: quadratic convergence

$\beta > 1$ $\int$ lin

Correction : β =1

So, the rate of convergence, so let r k be any sequence of numbers that converge to r say or r star say ok. So, we say that the order of convergence, there of convergence, the order of convergence of r k to r star is the largest value of beta you know greater than equal to 0 such that, this quantity this limit which is greater than equal to 0, this limit is k tends to infinity is finite ok.

So, what this is mean? It basically tells you how what is the largest value; so, if this quantity is finite. If this limit is finite it effectively tells you that for large enough k, r k plus 1 minus r star the absolute difference between these two is roughly equal to some constant times r k minus r star raise to the largest search value of beta right. So, for if you are so, if you are so the that is effectively what this particular thing we studying.

So, if you are, if you are going so, for large enough k your iterates behave in this kind of way right. So, if beta is if beta is to be precise here I should be taking the lim sup ok. So, if beta is taken it turns out to be 1 we say it is we say it is linear convergence, which means that your the distance between r k and r star which or.

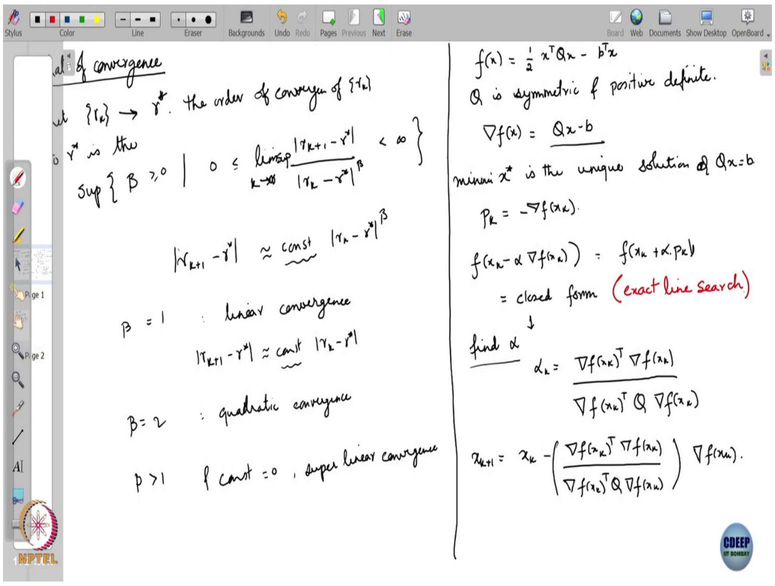
So, which means the updated distance between where you want to be and where you currently are is roughly equal to a constant times, your earlier distance r k minus r star. Of course, this you because this is convergence eventually the r k minus r star will decrease to 0 as well r k plus 1 minus r star.

But, what we are asking is how much progress are we making relative to where we were and how much progress have we made in the new iteration relative to where we were in the previous iteration right. And, that is what is captured capturing being captured by the rate of convergence.

Now, if beta equals 2 we says it is quadratic convergence. And, this is usually very very good, because if you can get, if your iterates can converge quadratically; that means, you are making the progress you are making is much more than what you at each iteration you have you are progressing much more than you had in the previous iteration right.

And, if it turns out that beta is equal to 0 sorry if it turns out that beta is greater than sorry not 0 here not equal to 0, if it turns out that this is if it is greater than 0, greater than 1. And, if it turns out that beta is greater than 1 and this constant here.

(Refer Slide Time: 05:54)



al of convergence

et $\{r_k\} \rightarrow r^*$. the order of convergen of $\{r_k\}$

o r^* is the

$$\sup \left\{ \beta \geq 0 \;\middle|\; 0 \leq \limsup_{k \to \infty} \frac{|r_{k+1} - r^*|}{|r_k - r^*|^\beta} < \infty \right\}$$

$$|r_{k+1} - r^*| \approx \underbrace{\text{const}} |r_k - r^*|^\beta$$

$\beta = 1$: linear convergence

$$|r_{k+1} - r^*| \approx \underbrace{\text{const}} |r_k - r^*|$$

$\beta = 2$: quadratic convergence

$\beta > 1$ & const $= 0$, super linear convergence

$$f(x) = \frac{1}{2} x^T Q x - b^T x$$

Q is symmetric & positive definite.

$$\nabla f(x) = \underline{Qx - b}$$

minimi x^* is the unique solution of $Qx = b$

$$p_k = -\nabla f(x_k).$$

$$f(x_k - \alpha \nabla f(x_k)) = f(x_k + \alpha . p_k)$$

$=$ closed form (exact line search)

$\downarrow$

find α

$$\alpha_k = \frac{\nabla f(x_k)^T \nabla f(x_k)}{\nabla f(x_k)^T Q \nabla f(x_k)}$$

$$x_{k+1} = x_k - \left(\frac{\nabla f(x_k)^T \nabla f(x_k)}{\nabla f(x_k)^T Q \nabla f(x_k)} \right) \nabla f(x_k).$$

And, this constant is actually equal to 0 ok. In that the constant here sorry the constant here is equal to 0, then we say it is super linear convergence ok. So, now, to let us to we what we will do is we look at the rates of by looking to study the rates of convergence, we can be I we will look at first the simplest form of a descent method, which is the steepest descent method.

And, that to we will study it only on only first on quadratic functions. Because, once you it turns out that a lot of what we want to say can what we want to learn about the rates of convergence can be learn from just this ok.

So, let us look at this function f which is half x transpose Q x minus b transpose x. And, I am going to take Q to be is symmetric and positive definite ok. The gradient of f can you can evaluate is equal to Q x minus b x star is then the unique, let x star be the unique solution, x star which is the minimizer of f is the unique solution. In the minimizer x star is the unique solution of Q x equals b alright.

Now, the good thing about quadratics is that you can calculate a lot of these the you can calculate a lot of these things in closed form. So, in fact, when we are doing, this since we are doing now, if you are doing steepest descent with line search, the actual step that minimizes the function along a certain search direction. So, the search direction for us is p k as which is equal to minus gradient of x k.

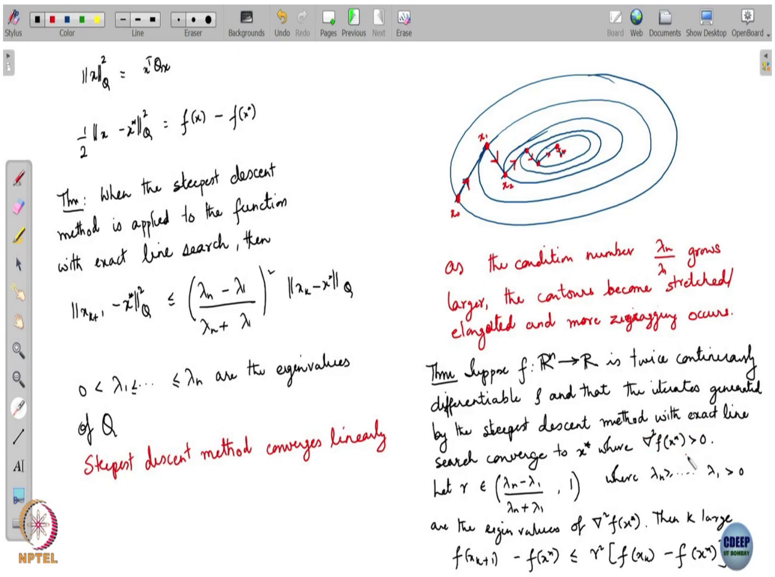
So, the it is we can actually calculate from x k minus alpha grad f of x k. So, this is equal to say remember f of x k plus alpha p k. So, we can actually calculate this in closed form; closed form, we can minimize find the alpha that minimizes this it turns out alpha so, find alpha. So, it turns and set that as alpha k.

So, it turns out alpha k is equal to gradient of f at x k transpose gradient of f at x k divided by k transpose Q times the same gradient of f at x k ok. So, the iteration then becomes x k plus 1 equals x k minus this term, gradient of f at x k transpose, gradient of f at x k divided by gradient of f at x k transpose Q times, gradient of f at x k, the whole thing times grad in f at x k.

So, this actually gives us you can now substitute this in all of these expressions in you know in terms of this expression Q that where the gradient is equal to Q x minus b; and you can actually get, you can actually get, the this in a much more explicit form where x k plus 1 can be written in terms of x k.

Now, it turns out that you can also therefore, using this then we can also calculate compute from here how this x the distance between x k plus 1 and x star. And the distance between or the difference distance between f of x minus f of x star.

(Refer Slide Time: 11:16)



$\|x\|_Q^2 = x^T Q x$

$\frac{1}{2}\|x - x^*\|_Q^2 = f(x) - f(x^*)$

Thm: When the steepest descent method is applied to the function with exact line search, then

$\|x_{k+1} - x^*\|_Q^2 \leq \left(\frac{\lambda_n - \lambda_1}{\lambda_n + \lambda_1}\right)^2 \|x_k - x^*\|_Q$

$0 < \lambda_1 \leq \cdots \leq \lambda_n$ are the eigenvalues of Q

Steepest descent method converges linearly

as the condition number $\frac{\lambda_n}{\lambda_1}$ grows larger, the contours become stretched/ elongated and more zigzagging occurs.

Thm Suppose $f: \mathbb{R}^n \to \mathbb{R}$ is twice continuously differentiable f and that the iterates generated by the steepest descent method with exact line search converge to x^* where $\nabla^2 f(x^*) > 0$.

Let $r \in \left(\frac{\lambda_n - \lambda_1}{\lambda_n + \lambda_1}, 1\right)$ where $\lambda_n \geq \cdots \lambda_1 > 0$ are the eigen values of $\nabla^2 f(x^*)$. Then k large

$f(x_{k+1}) - f(x^*) \leq r^2 [f(x_k) - f(x^*)]$

Let us write this as so, first let us define the following, let us define norm Q norm of x as x transpose Q x ok. So, this is effectively as a kind of as a skewed or a tilted norm with where I am taking the waiting matrix as Q. Now, we can it is very easy to show that from here that x minus x star square Q sorry, this squared, this is the squared Q now, this squared is actually nothing but f of x minus f of x star alright. And, now using this we can derive the following theorem, theorem says the following.

Now, with when the steepest, when the steepest descent method is applied to the function f with exact line search, what do I mean by exact line search? By exact line search I mean that is where we are finding alpha in closed form right. So, this is let us, let me write this here this is what is called exact line search.

What we did in the previous lecture, where we were looking for alphas that satisfied the wolf condition. Those are what are called inexact line search, because they are not we are not actually finding the exact minimum there. We are just putting conditions that are terminating alpha must be satisfied Ok yeah.

So, with the exact line search, it then we what we get then we find that x k plus 1 minus x star squared in Q. The Q norm is less than equal to lambda n minus lambda 1 divided by lambda n plus lambda 1, the whole squared times x k minus x star Q naught. Where what are these lambdas?

These lambdas this is lambda greater than 0 greater than dot dot dot greater than lambda n are the eigen values of Q. So, these lambda 1 to lambda n they are the eigen values of Q, now lambda 1 is the smallest eigen value, lambda n is the largest eigen value. And, so, now, what is this relation saying? This relation this Q norm squared is nothing but the difference between the function value and the optimal value of the function right.

So, this Q now so, the difference between so, if effectively this here this term here or this term here is initially is capturing something like an error, the error that you have or the departure that you have from your optimal solution. So, the error it says is less than at iteration k plus 1 is less than equal to this constant times the error that you had at iteration k.

You can see, what is this constant? Well this constant is this. So, this constant is something that depends on the eigen values of Q. So, usually what happens in a when you run this sort of steepest descent algorithm on a quadratic function like this. So, remember this was a convex quadratic function, because I assumed that that my Q is symmetric and positive semi definite.

So, if I look at the if I look at the contours of Q. So, let me draw this thing here for you, so, sorry contours of f. So, this would be let us say is the my outermost contour this is another contour inside, this is a another contour inside, this is another contour etcetera right.

So, if you do this here is what we get. So, you start from here go here ok. And, then you do this and then one goes say maybe I will draw one more contour in somewhere here. In some intermediate contour that takes us here, then another one takes us here, and another one takes us here, another one takes us here etcetera.

So, you can see what the way from what I have draw this here these are the iterates that your algorithm is produced. It is starting from this particular contour this is your x this is the point x 0 from here it goes in the direction of it goes in what is the steepest direction of steepest descent. Keeps going till it encounter till it can minimize the function along that direction, that minimum gets attained here at this at the second point here.

So, this is point x 0, this is point x 1, then it moves then it again goes in the direction where the this is minimized where it goes in the direction of the negative gradient, minimizes the function along that particular direction gets reaches point x 2. Then again goes in the direction where along the gradient now negative gradient at x 2 etcetera, etcetera right.

So, this is what happens in when you apply steepest descent to this sort of quadratic problem. Now, here this was a very simple problem because we actually knew where the solution also, where the solution was and you could have found a solution by simply inverting the matrix Q. But, you could, but we are trying to see what, how the behavior of steepest descent method actually work.

Now, what you can observe here is the main thing you can observe here is that the steepest descent method does tends to do this kind of zigzag. It is it goes, it keeps going zig zagging this way all the way towards the solution. And, the reason for that is this kind of zig zag is because of the skew that is introduced by the matrix by your Hessian matrix Q.

If, the the extent of zig zagging or really depends on how much you need to keep changing your directions of descent. And, that itself depends on what depends on how the how your eigen values are how different your eigen values are.

So, if in the simplest case, if all the you as in the simplest case, what does this say? Well in the simplest case if all the eigen values are equal what would happen? If all the eigen values are equal where in that case it would be Q would be or would essentially be a would essentially be an identity matrix; so all the eigen values are equal to 1.

In that case, what would happen then lambda n would be equal to sorry I should have put a weak in equality here my mistake. In that case lambda n would be equal to lambda 1 and in and then this term would be 0, the term on the term in this inequality would be 0 and x k plus 1 would be exactly equal to would be exactly equal to x star.

So, in the case when the eigen values are all equal the first step itself would not point in this sort of direction, but rather take you point straight towards the minimum, the actual global minimum of the function. It would point you to this direction directly. But, because there is a skew in the function the shape of the function takes you in this sort of path.

It first the first you go in this direction, then you go in this direction, then you go in this direction, then this way, that this way, then this way, etcetera ok. And, the extent to which you will be end up you would end up zig zagging, really depends on in general depends on the condition number or the ratio of the largest to the smallest eigen value right.

So, as the condition number grows larger, as the condition number grows larger the contours of the quadratic become moral elongated, they tend to sort of get more stretched out the contours become stretched out, stretched or elongated and more zig zagging occurs.

Nonetheless one thing that this result actually tells us directly is that the steepest descent method convergence linearly in general right. So, the convergence of the steepest method is linear. So, this sort of there is all this zig zagging, but it does converge, but it converges only linearly.

If your condition number is mild means close to 1, then the contours themselves would look more circular and the first try itself will get you close to the solution ok. So, that the actual number of iterations would be the actual number of iterations then would probably be a little lesser ok.

So, but the so, here is the main thing then that the rate of convergence depends on the extent of curvature or the or how would of how different the eigen values of the lowest and the largest eigen values are right ok, alright.

So, this is what we showing. So, the let me make a note here. So, the steepest descent converges linearly on this sort of quadratic problem. Now, it turns out that for problems that are more non-linear, were not necessarily quadratic some pretty much similar sort of result hold. So, I will just state again the theorem for you. Theorem: suppose f from R n to R is twice continuously differentiable.

And, that iterates generated by the steepest, descent method converge to x star with exact ok, let us say were the exact line are line search converge to x star. At where the if I locate the Hessian of f at x star this Hessian is positive definite. Now, let r be let r be any scalar that satisfies this.

Lambda n is a is a scalar that lies in this interval, lambda n minus lambda 1 divided by lambda n plus lambda 1 to 1, where lamda n greater than equal to dot dot dot lambda 1, greater than 0 are the eigen values of the Hessian at x star.

Then, for k large we have f of x k plus 1 minus f of x star is less than equal to 0, less than equal to r square times f of x k minus f of x star. Now, you can see how this why this is the case you the f of x k minus f of x star is essentially the same as this error term and this is what appeared here.

So, what is happening here is that you know as you for if you are in if you are considering a twice differentiable function and you can look at iterates that come from the steepest descent method with exact line search. And, suppose they converge to an x star which, where the

Hessian is positive definite. Then, the function value or the error term is going down to 0 linearly with is having showing linear convergence.

This you can see and the rate the constant outside again depends on the, depends on the condition number of the Hessian ok. The condition number of the Hessian at the point where the function converges. So, what is the lesson here the lesson is one of course, that the rate of convergence of a steepest descent method is linear in the best case.

And, second is that the quadratics the study of the steepest descent method on the on a quadratic function, gives you a good sense of what would be happening you know for a more general non-linear function. And, the reason for that the reason for that is basically the you know the rate of convergence at the end of the day depends essentially on how the function, how the iterates behave close to the solution, because it is eventually a limiting quantity.

And, since the rate of convergence depends on and when you are close to the solution a quadratic approximation of the function is a fair approximation. So, quadratics tend to give you a how the performance of an algorithm on an quadratic, tends to give you also how the algorithm would perform on more non-linear type of functions, right.