Optimization Algorithms: Theory and Software Implementation

Prof. Thirumulanathan D

Department of Mathematics

Institute of IIT Kanpur

Lecture: 23

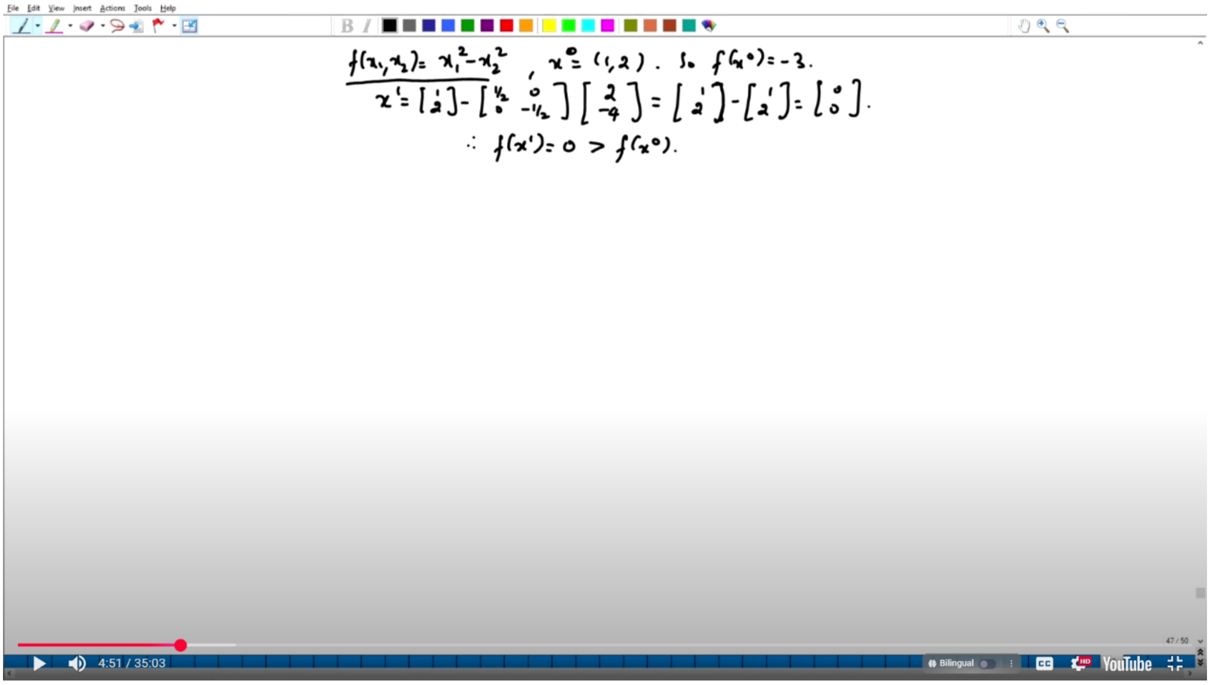
Hello everyone, this is the third lecture of week five. We are continuing our discussion on Newton's method. In the previous lectures, we covered the general form of Newton's method, looked at examples, and discussed its quadratic order of convergence. We also noted a major drawback: Newton's method is only locally convergent, meaning it requires an initial guess sufficiently close to the solution to converge. We saw an example of this with the function involving square roots.

Today, we will examine the other two issues we mentioned: the Newton direction may not be a descent direction if the Hessian is not positive definite, and the computational expense and potential singularity of the Hessian matrix. We will provide examples for each.

First, let's consider the issue where the Newton direction may not be a descent direction. Recall that the direction is given by $d_k = -H_k^{-1}g_k$. This is guaranteed to be a descent direction only if H_k is positive definite. If H_k is not positive definite, this direction might lead to an increase in the function value.

Consider the function $f(x_1, x_2) = x_1^2 - x_2^2$. This is a quadratic function with an indefinite Hessian. Let's take the initial point $x_0 = (1, 2)$. The function value at x_0 is $f(1, 2) = 1 - 4 = -3$. The gradient is $g(x) = [2x_1, -2x_2]$, so at x_0, $g = [2, -4]$. The Hessian is $H = [[2, 0], [0, -2]]$, which is indefinite. Its inverse is $H^{-1} = [[1/2, 0], [0, -1/2]]$. The Newton step is $x_1 = x_0 - H^{-1}g = (1, 2) - (1/2 * 2, -1/2 * -4) = (1, 2) - (1, 2) = (0, 0)$. The function value at x_1 is $f(0, 0) = 0$, which is greater than $f(x_0) = -3$. Thus, the step increased the function value, demonstrating that the Newton direction was not a descent direction here. Note that this function has no minimum (it goes to $-\infty$), so this example might seem contrived.

(Refer Slide Time 4:51)



$f(x_1, x_2) = x_1^2 - x_2^2$, $x^0 = (1, 2)$. So $f(x^0) = -3$.

$$x^1 = \begin{bmatrix} 1 \\ 2 \end{bmatrix} - \begin{bmatrix} 1/2 & 0 \\ 0 & -1/2 \end{bmatrix} \begin{bmatrix} 2 \\ -4 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \end{bmatrix} - \begin{bmatrix} 1 \\ 2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}.$$

$\therefore f(x^1) = 0 > f(x^0)$.

Let's consider a function that does have a minimum. We return to our familiar function

$f(x_1, x_2) = x_1^2e^{x_2} + x_2^2e^{x_1}$, we choose the initial point $x_0 = (-\sqrt{2}, -\sqrt{2})$.

We know the global minimum is at (0, 0). Let's see what happens when we apply Newton's method from this point.

We define the function, its gradient, and its Hessian in Python.

The gradient is:

$\nabla f(x) = [2x_1e^{x_2} + x_2^2e^{x_1}, 2x_2e^{x_1} + x_1^2e^{x_2}]$

The Hessian is:

$H(x) = [[2e^{x_2} + x_2^2e^{x_1}, 2x_1e^{x_2} + 2x_2e^{x_1}],$

$[2x_1e^{x_2} + 2x_2e^{x_1}, 2e^{x^1} + x_1^2e^{x_2}]]$

We implement Newton's method and run it from $x_0 = (-\sqrt{2}, -\sqrt{2})$. We observe that the algorithm does not converge to (0, 0). Instead, it converges to the point (-2, -2). Let's check the gradient and Hessian at (-2, -2).

At (-2, -2):

$\nabla f(x) = [2*(-2)*e^{-2} + (-2)^2*e^{-2}, 2*(-2)*e^{-2} + (-2)^2*e^{-2}] = [-4e^{-2} + 4e^{-2}, -4e^{-2} + 4e^{-2}] = (0, 0)$

So, (-2, -2) is a critical point.

The Hessian at (-2, -2) is:

$H_{11} = 2e^{-2} + 4e^{-2} = 6e^{-2}$

$H_{12} = 2*(-2)*e^{-2} + 2*(-2)*e^{-2} = -4e^{-2} - 4e^{-2} = -8e^{-2}$

$H_{21} = -8e^{-2}$

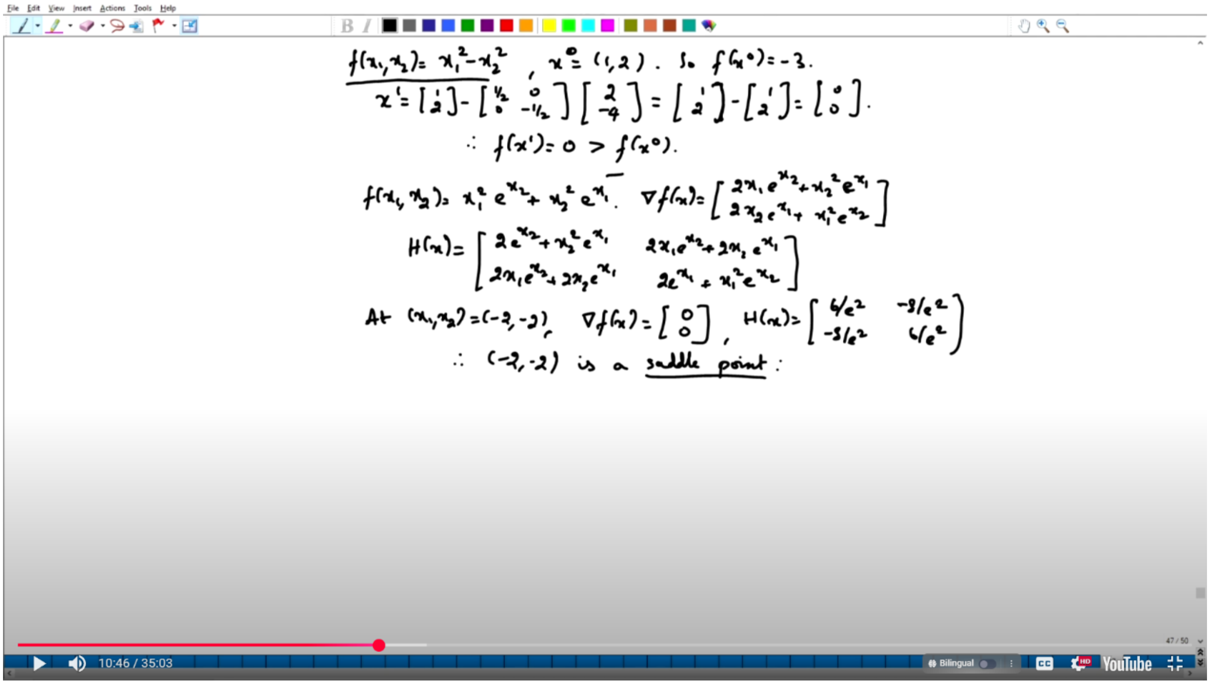
$H_{22} = 2e^{-2} + 4e^{-2} = 6e^{-2}$

So,

$H = [[6e^{-2}, -8e^{-2}], [-8e^{-2}, 6e^{-2}]]$

The determinant of H is $(6e^{-2})*(6e^{-2}) - (-8e^{-2})*(-8e^{-2}) = 36e^{-4} - 64e^{-4} = -28e^{-4} < 0$. Since the determinant is negative, the Hessian is indefinite at this point, meaning (-2, -2) is a saddle point, not a minimum.

(Refer Slide Time 10:46)



$\underline{f(x_1,x_2) = x_1^2 - x_2^2}$, $x^0 = (1,2)$. So $f(x^0) = -3$.

$$x^1 = \begin{bmatrix} 1 \\ 2 \end{bmatrix} - \begin{bmatrix} 1/2 & 0 \\ 0 & -1/2 \end{bmatrix} \begin{bmatrix} 2 \\ -4 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \end{bmatrix} - \begin{bmatrix} 1 \\ 2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}.$$

$\therefore f(x^1) = 0 > f(x^0)$.

$f(x_1, x_2) = x_1^2 e^{x_2} + x_2^2 e^{x_1}$. $\nabla f(x) = \begin{bmatrix} 2x_1 e^{x_2} + x_2^2 e^{x_1} \\ 2x_2 e^{x_1} + x_1^2 e^{x_2} \end{bmatrix}$

$$H(x) = \begin{bmatrix} 2e^{x_2} + x_2^2 e^{x_1} & 2x_1 e^{x_2} + 2x_2 e^{x_1} \\ 2x_1 e^{x_2} + 2x_2 e^{x_1} & 2e^{x_1} + x_1^2 e^{x_2} \end{bmatrix}$$

At $(x_1, x_2) = (-2, -2)$, $\nabla f(x) = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$, $H(x) = \begin{bmatrix} 6/e^2 & -8/e^2 \\ -8/e^2 & 6/e^2 \end{bmatrix}$

$\therefore (-2,-2)$ is a $\underline{\text{saddle point}}$.

When we run Newton's method from (-√2, -√2), the function value increases from approximately 0.9725 at x_0 to 1.081 at x_1, confirming the step was not a descent step. The algorithm converges to this saddle point instead of the true minimum at (0, 0).

For comparison, if we run the Fletcher-Reeves conjugate gradient method or gradient descent from the same initial point (-√2, -√2), both algorithms converge correctly to (0, 0) in about 20-30 steps. This shows that while Newton's method is faster when it works, it can fail by converging to saddle points or even maxima if the Hessian is not positive definite.

Now, let's discuss the third issue: the computational expense of inverting the Hessian and the possibility of encountering a singular Hessian. Inverting an n×n matrix requires $O(n^3)$ operations, which becomes prohibitively expensive for large n (e.g., in machine learning problems with thousands of variables). Moreover, the Hessian might be singular (non-invertible).

Consider the function $f(x_1, x_2) = x_1^3 - 3x_1 + x_2^3 - 3x_2$. Its gradient is $\nabla f(x) = [3(x_1^2 - 1), 3(x_2^2 - 1)]$. The critical points are at (1,1), (1,-1), (-1,1), and (-1,-1). The Hessian is $H(x) = [[6x_1, 0], [0, 6x_2]]$.

This Hessian is positive definite only at (1,1), which is a local minimum. It is indefinite at (1,-1) and (-1,1) (saddle points) and negative definite at (-1,-1) (local maximum). Notice that at (0,0), the Hessian is H(0,0) = [[0,0],[0,0]], which is singular.

If we try to run Newton's method starting from (0,0), the algorithm will fail because it attempts to invert a singular matrix. In practice, you would get an error. The solution is to start from a different initial point or use a modified algorithm that handles singular Hessians.

We have now seen examples of all three main issues with Newton's method:

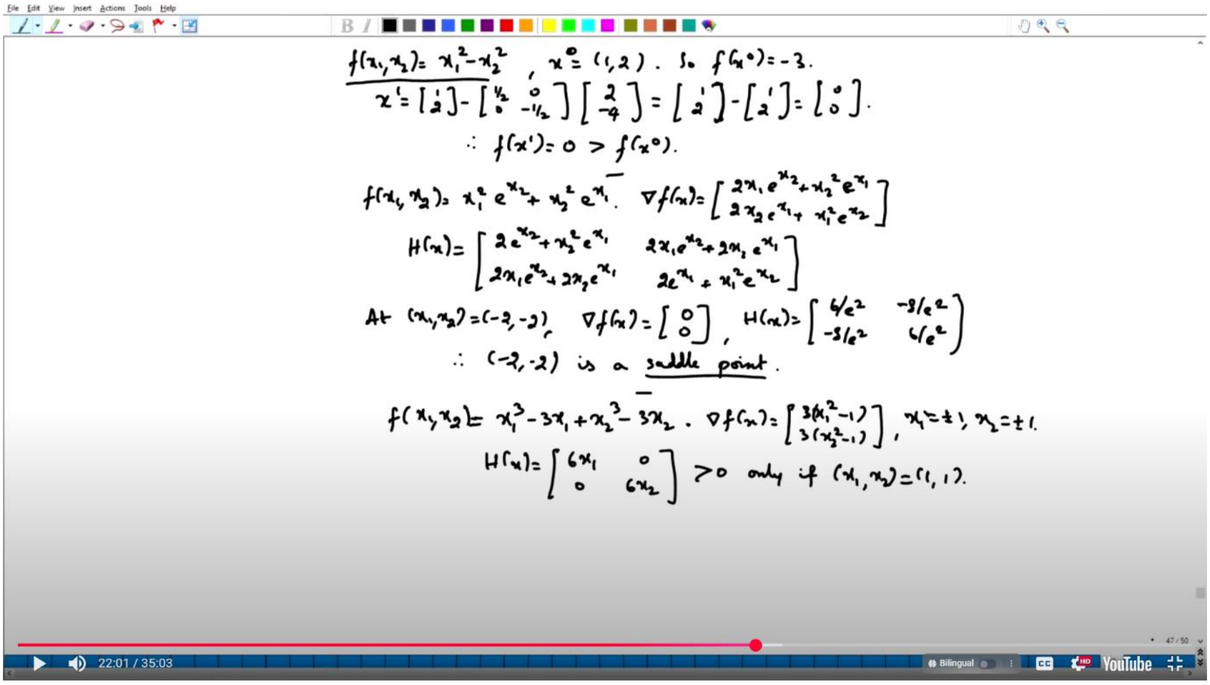
1. Lack of global convergence (only local convergence).

2. The Newton direction may not be a descent direction if the Hessian is not positive definite.

3. Computational expense and potential singularity of the Hessian.

Despite these issues, Newton's method has a powerful property: under certain conditions, it converges quadratically. Let's state and prove this theorem for the one-dimensional case for simplicity. The multi-dimensional case is analogous but involves more complex notation.

(Refer Slide Time 22:01)



$$\underline{f(x_1,x_2) = x_1^2 - x_2^2}, \quad x^0 = (1,2). \text{ So } f(x^0) = -3.$$

$$x^1 = \begin{bmatrix} 1 \\ 2 \end{bmatrix} - \begin{bmatrix} 1/2 & 0 \\ 0 & -1/2 \end{bmatrix} \begin{bmatrix} 2 \\ -4 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \end{bmatrix} - \begin{bmatrix} 1 \\ 2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}.$$

$$\therefore f(x^1) = 0 > f(x^0).$$

$$f(x_1,x_2) = x_1^2 e^{x_2} + x_2^2 e^{x_1}. \quad \nabla f(x) = \begin{bmatrix} 2x_1 e^{x_2} + x_2^2 e^{x_1} \\ 2x_2 e^{x_1} + x_1^2 e^{x_2} \end{bmatrix}$$

$$H(x) = \begin{bmatrix} 2e^{x_2} + x_2^2 e^{x_1} & 2x_1 e^{x_2} + 2x_2 e^{x_1} \\ 2x_1 e^{x_2} + 2x_2 e^{x_1} & 2e^{x_1} + x_1^2 e^{x_2} \end{bmatrix}$$

$$\text{At } (x_1,x_2) = (-2,-2), \quad \nabla f(x) = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \quad H(x) = \begin{bmatrix} 6/e^2 & -8/e^2 \\ -8/e^2 & 6/e^2 \end{bmatrix}$$

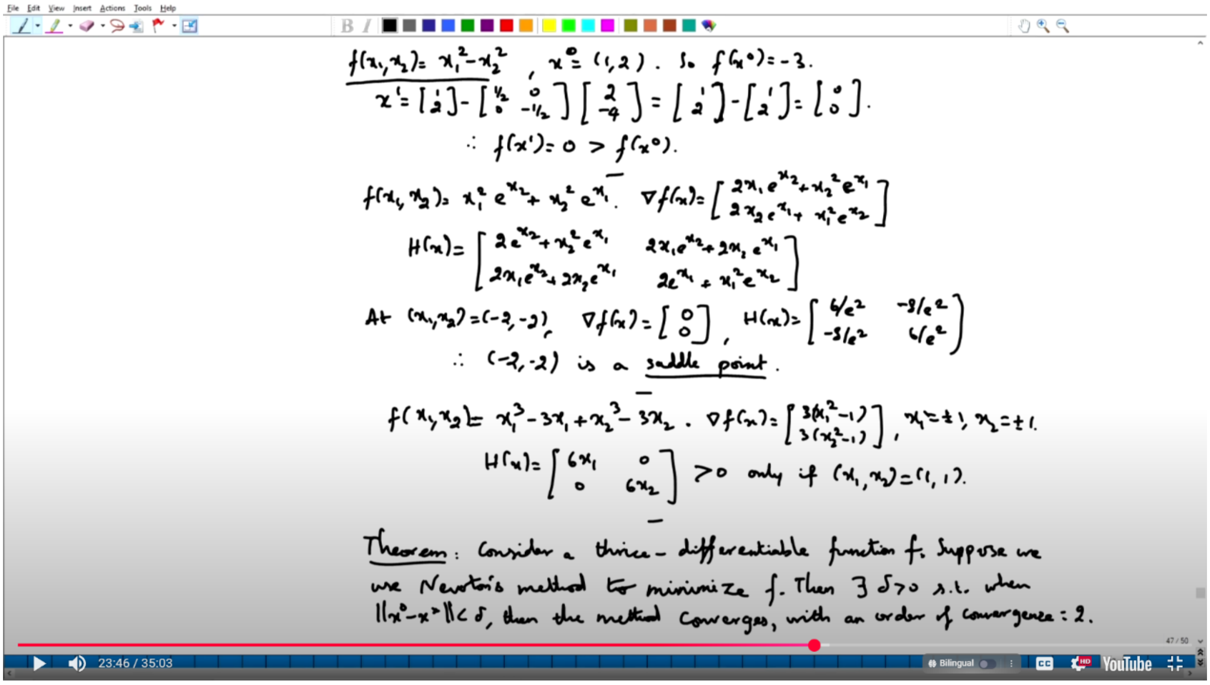
$$\therefore (-2,-2) \text{ is a } \underline{\text{saddle point}}.$$

$$f(x_1,x_2) = x_1^3 - 3x_1 + x_2^3 - 3x_2. \quad \nabla f(x) = \begin{bmatrix} 3(x_1^2-1) \\ 3(x_2^2-1) \end{bmatrix}, \quad x_1 = \pm 1, x_2 = \pm 1.$$

$$H(x) = \begin{bmatrix} 6x_1 & 0 \\ 0 & 6x_2 \end{bmatrix} > 0 \text{ only if } (x_1,x_2) = (1,1).$$

Theorem: Let f: R → R be a three-times continuously differentiable function. Let x* be a minimizer such that f'(x*) = 0 and f''(x*) > 0. Then, there exists a $\delta > 0$ such that if $|x_0 - x^*| < \delta$, Newton's method converges to x* with an order of convergence of 2.

(Refer Slide Time 23:46)



$f(x_1,x_2) = x_1^2 - x_2^2$, $x^0 = (1,2)$. So $f(x^0) = -3$.

$$x^1 = \begin{bmatrix} 1 \\ 2 \end{bmatrix} - \begin{bmatrix} 1/2 & 0 \\ 0 & -1/2 \end{bmatrix} \begin{bmatrix} 2 \\ -4 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \end{bmatrix} - \begin{bmatrix} 1 \\ 2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}.$$

$\therefore f(x^1) = 0 > f(x^0)$.

$f(x_1,x_2) = x_1^2 e^{x_2} + x_2^2 e^{x_1}$. $\nabla f(x) = \begin{bmatrix} 2x_1 e^{x_2} + x_2^2 e^{x_1} \\ 2x_2 e^{x_1} + x_1^2 e^{x_2} \end{bmatrix}$

$$H(x) = \begin{bmatrix} 2e^{x_2} + x_2^2 e^{x_1} & 2x_1 e^{x_2} + 2x_2 e^{x_1} \\ 2x_1 e^{x_2} + 2x_2 e^{x_1} & 2e^{x_1} + x_1^2 e^{x_2} \end{bmatrix}$$

At $(x_1,x_2) = (-2,-2)$, $\nabla f(x) = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$, $H(x) = \begin{bmatrix} 6/e^2 & -8/e^2 \\ -3/e^2 & 6/e^2 \end{bmatrix}$

$\therefore (-2,-2)$ is a saddle point.

$f(x_1,x_2) = x_1^3 - 3x_1 + x_2^3 - 3x_2$. $\nabla f(x) = \begin{bmatrix} 3(x_1^2-1) \\ 3(x_2^2-1) \end{bmatrix}$, $x_1 = \pm 1$, $x_2 = \pm 1$.

$H(x) = \begin{bmatrix} 6x_1 & 0 \\ 0 & 6x_2 \end{bmatrix} > 0$ only if $(x_1,x_2) = (1,1)$.

Theorem: Consider a thrice-differentiable function f. Suppose we use Newton's method to minimize f. Then $\exists\, \delta > 0$ s.t. when $\|x^0 - x^*\| < \delta$, then the method converges, with an order of convergence : 2.

Proof (for n=1):

The Newton iteration is $x_{k+1} = x_k - f'(x_k)/f''(x_k)$.

Define the error $e_k = x_k - x^*$.

We have $e_{k+1} = x_{k+1} - x^* = x_k - x^* - f'(x_k)/f''(x_k) = e_k - f'(x_k)/f''(x_k)$.

Now, consider the Taylor expansion of $f'(x^*)$ around x_k:

$0 = f'(x^*) = f'(x_k) + f''(x_k)(x^* - x_k) + (1/2)f'''(\xi_k)(x^* - x_k)^2$ for some ξ_k between x_k and x^*.

Thus, $f'(x_k) = f''(x_k)e_k - (1/2)f'''(\xi_k)e_k^2$.

Substitute this into the expression for e_{k+1}:

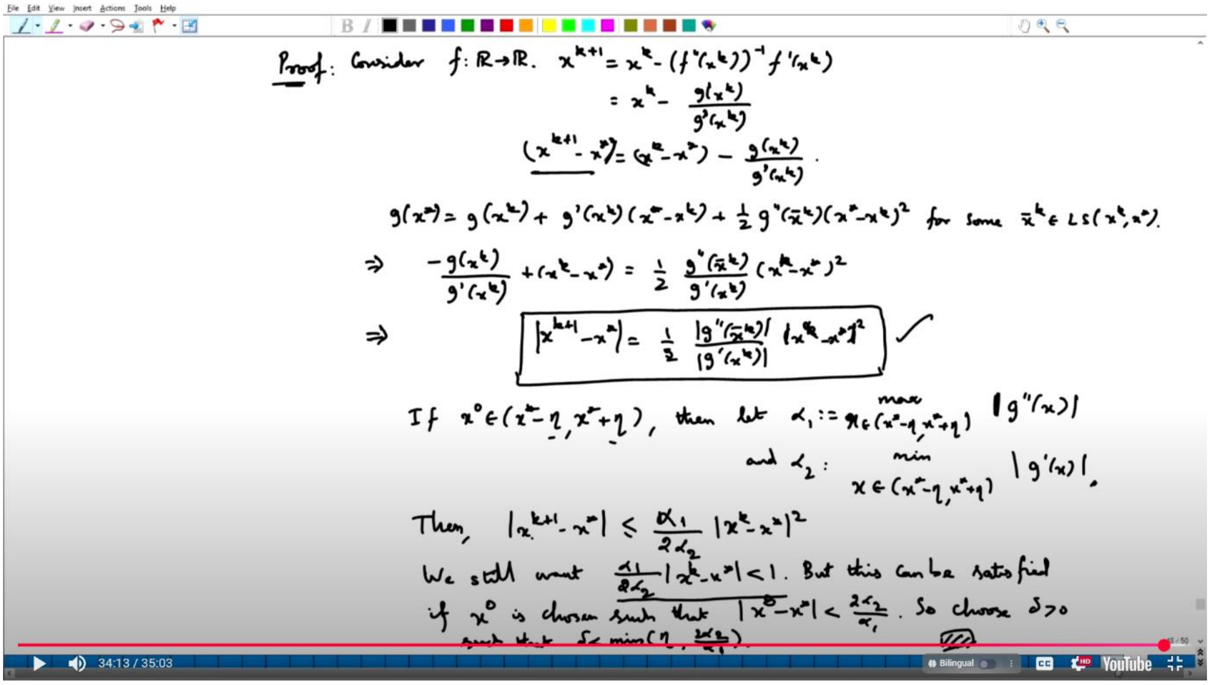
$e_{k+1} = e_k - [f''(x_k)e_k - (1/2)f'''(\xi_k)e_k^2] / f''(x_k) = e_k - e_k + (1/2)(f'''(\xi_k)/f''(x_k))e_k^2 = (1/2)(f'''(\xi_k)/f''(x_k))e_k^2$.

Therefore, $|e_{k+1}| = (1/2)|f'''(\xi_k)/f''(x_k)|\,|e_k|^2$.

Since f is three-times continuously differentiable and $f''(x^*) > 0$, there exists an interval I around x^* where $f''(x)$ is bounded away from zero and $f'''(x)$ is bounded. Let $\alpha_1 = \max\_\{x \in I\} |f'''(x)|$ and $\alpha_2 = \min\_\{x \text{ in } I\} |f''(x)|$. Then, $|e_{k+1}| \leq (\alpha_1/(2\alpha_2))\,|e_k|^2$.

If we choose δ such that $\delta < \min\{$ radius of I, $2\alpha_2/\alpha_1 \}$, then for $|e_0| < \delta$, we have $|e_1| \leq (\alpha_1/(2\alpha_2))\,\delta^2 < \delta$, and similarly for subsequent steps. Moreover, the error decreases quadratically, as $|e_{k+1}|$ is proportional to $|e_k|^2$. This proves local convergence with quadratic order.

(Refer Slide Time 34:13)



<u>Proof</u>: Consider $f: \mathbb{R} \to \mathbb{R}$. $x^{k+1} = x^k - (f''(x^k))^{-1} f'(x^k)$

$$= x^k - \frac{g(x^k)}{g'(x^k)}$$

$$\underline{(x^{k+1} - x^*)} = (x^k - x^*) - \frac{g(x^k)}{g'(x^k)}.$$

$g(x^*) = g(x^k) + g'(x^k)(x^* - x^k) + \frac{1}{2} g''(\bar{x}^k)(x^* - x^k)^2$ for some $\bar{x}^k \in LS(x^k, x^*)$.

$$\Rightarrow \quad \frac{-g(x^k)}{g'(x^k)} + (x^k - x^*) = \frac{1}{2} \frac{g''(\bar{x}^k)}{g'(x^k)} (x^k - x^*)^2$$

$$\Rightarrow \quad \boxed{|x^{k+1} - x^*| = \frac{1}{2} \frac{|g''(\bar{x}^k)|}{|g'(x^k)|} |x^k - x^*|^2} \checkmark$$

If $x^0 \in (x^* - \eta, x^* + \eta)$, then let $\alpha_1 := \max_{x \in (x^* - \eta, x^* + \eta)} |g''(x)|$

and $\alpha_2 : \min_{x \in (x^* - \eta, x^* + \eta)} |g'(x)|$.

Then, $|x^{k+1} - x^*| \leq \frac{\alpha_1}{2\alpha_2} |x^k - x^*|^2$

We still want $\frac{\alpha_1}{2\alpha_2} |x^k - x^*| < 1$. But this can be satisfied if x^0 is chosen such that $|x^0 - x^*| < \frac{2\alpha_2}{\alpha_1}$. So choose $\delta > 0$ such that $\delta < \min(\eta, \frac{2\alpha_2}{\alpha_1})$.

In the next lecture, we will discuss variants of Newton's method, such as damped Newton's method and trust region methods, which are designed to overcome these issues.

Thank you.