Q_2 Explain the steepest (gradient) descent algorithm and NR algorithm.

For unconstrained OP!

If confused see video!!

SD iterative 1-st order opti to find local min

Eq: $x_{k+1} = x_k + \alpha_k d_k$, where: $x_0 \rightarrow$ Start point,

$\underbrace{\alpha_k \in \arg\min_{\alpha \geq 0} f(x_k + \alpha d_k)}_{\text{Line search}}$, $d_k = -\nabla f(x_k)$

← Exact line search yields orthogonal d_k

Update x_{k+1}

if $\|\alpha_k d_k\| < \varepsilon$ or $k = max\_iter$ output: $x' = x_{k+1}$, $f(x')$ and break

---

NR: only change: $d_k = -H_f(x_k)^{-1} \nabla f(x_k)$ slide 19!!

if $H_f(x_k)$ not PD make it PD → several methods

Why Hessian? Rate of Change of the $\nabla f(x_k)$, $H_f(x_k)^{-1}$ adjust the stepsize by large stepsize if small change (flat).

Inverse = bad computation

Line search: Golden section search, backtracking (inexact)

GSS: See slide or video

Backtrack:

SD works well far out where NR performs better the closer you are to minima

Condition number

$r = \frac{\min \rho(H_f(x_k))}{\max \rho(H_f(x_k))} \begin{cases} \approx 1 \text{ fast} \\ \approx 0 \text{ slow} \end{cases}$ ↓ Converge for SD

$\rho(H_f(x_k))$ = eigenvalues Hessian

OP = Optimization Problem

NR = Newton Rhapson, SD = Steepest descent