Backtracking line search alg.

- **Given**
  - Point $x$
  - Descent direction $\Delta x$
  - $\alpha \in (0, 0.5)$
  - $\beta \in (0, 1)$

- $t=1$

- While $f(x+t\Delta x) > f(x) + \alpha t \nabla f(x)^T \Delta x$
  - $t := \beta t$

- **Boyd & Vandenberghe**: pick
  - $\alpha$ in $[0.01, 0.3]$
  - $\beta$ in $[0.1, 0.8]$

- For every iteration, do line and start from $t=1$
Newton’s method – alg.

- Start from some $x$ in $\text{dom } f$
- Repeat
  - Determine descent direction $\Delta x_{nt}$
    - solving a linear system $\nabla^2 f(x) \Delta x = -\nabla f(x)$
  - **Line search** to choose step size $t$
  - Update: $x \leftarrow x + t \Delta x_{nt}$
- Until stopping criterion

Good stopping criterion:

$$\frac{1}{2} \nabla f(x)^T \nabla^2 f(x)^{-1} \nabla f(x) \leq \epsilon$$

Convergence analysis for Newton’s

- (Really see book for details.)
- Two phases:
  - Gradient is large
    - Damped Newton Phase
      - Step size $t < 1$
      - Linear convergence
  - Gradient is small
    - Pure Newton Phase
      - Step size $t = 1$
      - Quadratic convergence
        - $c^2(t)$
      - Only lasts 6 steps

©2008-2010 Carlos Guestrin
Proof intuition

Assumptions:
- Strong convexity:
  \( \nabla^2 f(x) \geq m I \quad \forall x \)
- Upper bound on Hessian:
  \[ M I \leq \nabla^2 f(x) \]
  Implied by strong convexity
- Hessian of \( f \) is Lipschitz continuous: (bound on third derivative)
  \[ \| \nabla^2 f(x) - \nabla^2 f(y) \|_2 \leq L \| x - y \|_2 \]
  \( L \) is Lipschitz constant

Proof shows Newton follows two phases (details in book):
- Constants: \( 0 < \alpha < \frac{1}{C} \); \( 0 < \gamma < 1 \)
- Damped Newton Phase:
  \[ f_k - f_{k+1} \geq \gamma \]
  \[ \text{step size } \tau_k \leq \gamma \]
- Pure Newton Phase:
  \[ \| \nabla f_k \|_2 \leq \gamma \]
  \[ \| \nabla f_{k+1} \|_2 \leq \left( \frac{1}{L\gamma} \| \nabla f_k \|_2 \right)^2 \]

Complexity of Damped Newton Phase

- When gradient is large:
  \[ \| \nabla f_k \|_2 \geq \gamma \]

  \[ f_k - f_{k+1} \geq \gamma \]

- Relate to suboptimality:
  \[ f_0 - p^* = \epsilon_0 \geq 0 \quad \text{initial error} \]

  \[ f_k - p^* \leq \epsilon_k - \gamma \]

  \[ \frac{\epsilon_k}{\gamma} = \frac{f_0 - p^*}{\delta} \quad \text{steps in damped Newton phase} \]

- Total number of iterations in Damped Newton Phase
  \[ \frac{f_0 - p^*}{\delta} \]
Pure Newton Phase

- Proof guarantees that
  - When gradient is small: \( \| \nabla f_k \| \leq \varepsilon \) always in Pure Newton
  - Gradient decreases quadratically:
    \[
    \frac{1}{2m} \| \nabla f_k \| \leq \left( \frac{1}{2m} \| \nabla f_k \| \right)^2
    \]
  - From convexity, can relate gradient to error:
    \[
    \frac{\text{error}}{\| f \|} \leq \frac{1}{2m^2} \| \nabla f_k \|^2 \leq \left( \frac{1}{2m^2} \| \nabla f_k \|^2 \right)^k
    \]
  - Applying bound recursively:
    \[
    \| f_k \| \leq \left( \frac{1}{2m^2} \| \nabla f_k \|^2 \right)^k \leq \left( \frac{1}{2m^2} \| \nabla f_k \|^2 \right)^k \leq \left( \frac{1}{2} \right)^k
    \]
    
    So, done after 6 iterations!

The power of quadratic convergence

- Linear convergence (e.g., at rate \( \frac{1}{2} \))
  - Bits of precision improve linearly
    \[
    f_k - p^* \leq \varepsilon \left( \frac{1}{2} \right)^k \leq \varepsilon \quad \Rightarrow \quad \log_2 \frac{\varepsilon}{\varepsilon_0} \leq k \quad \text{linearly with iteration}
    \]

- Quadratic convergence (e.g., at rate \( \frac{1}{2} \))
  - Bits of precision improve exponentially:
    \[
    f_k - p^* \leq \varepsilon \left( \frac{1}{2} \right)^k \leq \varepsilon \quad \Rightarrow \quad \log_2 \frac{\varepsilon}{\varepsilon_0} \leq 2k \quad \text{doubles with iteration!}
    \]
    
    So, done after 6 iterations!

\[ \log_2 \log_2 \frac{\varepsilon}{\varepsilon_0} \quad \text{very slow growth} \quad \text{after 6 iterations} \]

\[ \varepsilon \approx 5 \times 10^{-20} \]
Putting it all together: Complexity of Newton’s method – Simple Analysis

- **Damped Newton Phase:**

  \[ \text{Iterations: } \frac{f_0 - p^*}{\delta} \quad \text{from proof} \quad \delta = \lambda \eta \frac{m}{\eta^2} \]

- **Pure Newton Phase:**

  6 iterations, when
  \[ \eta \leq \min \{1, \frac{3(1-2\eta)}{2} \} \frac{m^2}{\epsilon} \]

- **Total:**

  \[ \frac{\lambda^2 \epsilon}{\eta \min \{1, \frac{3(1-2\eta)}{2} \} (f_0 - p^*)} + 6 \text{ iterations} \]

  - Bound depends on unknown quantities! 😊

---

Self-Concordance:

**A smoothness assumption**

- **Self-concordant function in 1D:**

  \[ |f''(x)| \leq \frac{3}{2} (f''(x))^3 \quad \forall x \quad \text{where } f(x) \text{ is self-concorded in } + \text{ for all } x, v \]

  - A bound on the third derivative in terms of the second, rather than unknown constants

- **Example:** \( f(x) = -\log x \)

  \[ f''(x) = \frac{1}{x^2}, \quad f'''(x) = -\frac{6}{x^3} \]

  \[ \frac{3}{2} \left( \frac{f''(x)}{2} \right)^2 \leq \frac{3}{2} \left( \frac{6}{2x^3} \right)^2 = 1 \quad \text{for } f(x) = -\log x \text{ is perfectly self-concordant} \]
When function is Self-Concordant, Analysis of Newton’s method is very clean!

For self-concordant functions, no constants in bound:

\[ \text{# iterations} \leq \frac{f_o - p^*}{\epsilon} + \log \log \frac{1}{\epsilon} \]

For example:

\[ \alpha = 0.1, \quad \rho = 0.8 \]

\[ \text{# iterations} \leq 3.78 \left( f_o - p^* \right) + 6 \]

In Practice, bounds are loose, but general behavior captured

Bound on \# iterations from previous slide:

\[ 3.78 \left( f_o - p^* \right) + 6 \]

Example from book:
- Self concordant function
- Empirical bound:
Solving problems with equality constraints

- Equality constraints:
  \[ \begin{align*}
  \forall x & \in \mathbb{R}^n \quad f(x) = 0 \\
  \eta & \in \mathbb{R}^m
  \end{align*} \]

- Seems very hard

Null space

- Equality constraints: \( \forall \mathbf{x} \in \mathbb{R}^n, \mathbf{A} \mathbf{x} = \mathbf{b} \)

- Given one solution: \( \mathbf{x}_1 : \mathbf{A} \mathbf{x}_1 = \mathbf{b} \)

- Find other solutions: \( \mathbf{x}_2 = \mathbf{x}_1 + \mathbf{g} \)
  \[ \begin{align*}
  \mathbf{A} \mathbf{x}_2 &= \mathbf{A} \mathbf{x}_1 + \mathbf{A} \mathbf{g} \\
  &= \mathbf{b} + \mathbf{b} \\
  &= \mathbf{b}
  \end{align*} \]

- Since Null Space is a linear subspace:
  \[ \mathbf{N}(\mathbf{A}) = \left\{ \mathbf{x} \in \mathbb{R}^n \mid \mathbf{A} \mathbf{x} = \mathbf{0} \right\} \]
  \[ \mathbf{N}(\mathbf{A}) = \left\{ \mathbf{x} + \mathbf{F} \mathbf{z} \mid \mathbf{z} \in \mathbb{R}^k \right\} \]

\[ \mathbf{F} \] is a basis for \( \mathbf{N}(\mathbf{A}) \)
Eliminating linear equalities

- Equivalent optimization problems:
  \[
  \min_x f(x) \quad \text{subject to} \quad Ax = b, \quad x \in \mathbb{R}^n
  \]

- Find basis for null space of A (linear algebra)
  - Solve unconstrained problem
    \[
    \begin{bmatrix}
    A \\
    \end{bmatrix} \begin{bmatrix}
    x \\
    \end{bmatrix} = \begin{bmatrix}
    b \\
    \end{bmatrix}
    \]

- A concern…
  - e.g., A structure, e.g., sparsity
  - F could lose sparsity

Solving quadratic problems with equality constraints

- Quadratic problem with equality constraints:
  \[
  \min_x \frac{1}{2} x^T P x + \epsilon x
  \]

- KKT condition \( x^* \) solution iff
  \[
  \begin{array}{c}
  \text{feasibility:} \\
  \text{grad} \text{ minimizing} \\
  A x = b, \quad \epsilon (x^t) + A^t v^* = 0 \\
  P x^* + \epsilon x^* + A^t v^* = 0 \quad \text{c} \\
  \end{array}
  \]

- Rewriting:
  \[
  \begin{bmatrix}
  P & A^t \\
  A & 0 \\
  \end{bmatrix}
  \begin{bmatrix}
  x^t \\
  v^* \\
  \end{bmatrix}
  = \begin{bmatrix}
  \epsilon \\
  b \\
  \end{bmatrix}
  \]

- Solve linear system:
  - Any solution is OPT
  - If no solution, unbounded
Newton’s method with equality constraints

- Quadratic approximation:
  \[
  f(x + \Delta x) = f(x) + \nabla f(x)^T \Delta x + \frac{1}{2} \Delta x^T \nabla^2 f(x) \Delta x
  \]
- Start feasible, stay feasible:
  \[
  x^{(n)} = x^{(n)} + t \Delta x \Rightarrow A x^{(n)} + t \Delta x \leq b
  \]
- KKT:
  \[
  \begin{bmatrix}
  \nabla^2 f(x) & A^T \\
  A & 0
  \end{bmatrix}
  \begin{bmatrix}
  \Delta x^f \\
  w
  \end{bmatrix}
  =
  \begin{bmatrix}
  -\nabla f(x) \\
  0
  \end{bmatrix}
  \]
- Solve linear system:
- Move accordingly:
  \[
  x^{(n+1)} = x^{(n)} + t \Delta x^f
  \]
  (always remain feasible)