Last Time:

- Root Finding
- Newton's Method
- Minimization

Today:

- Line Search
- Constrained Minimization

---

* Line Search

- Often Δx step from Newton is too big and overshoots the minimum
- To fix this, check $f(x+\Delta x)$ and "backtrack" until we get a "good" reduction
- Many strategies exist.
- A simple + effective one is "Armijo rule":

$\alpha = 1$ ← "step length"

while $f(x+\alpha\Delta x) > f(x) + b\,\underbrace{\alpha \nabla f(x)^T \Delta x}_{\text{expected reduction from gradient}}$ ← tolerance (b)

$\alpha \leftarrow c\alpha$ ← scalar < 1 (c)

end

* Intuition:

  - Make sure step agrees with linearization within some tolerance.

* Typical Values:

$$c = 1/2 \quad , \quad b = 10^{-4} - 0.1$$

* Take-Away Messages

  - Newton with simple + cheap modifications ("globalization strategies") is extremely effective at finding local minima.

---

* Equality Constraints

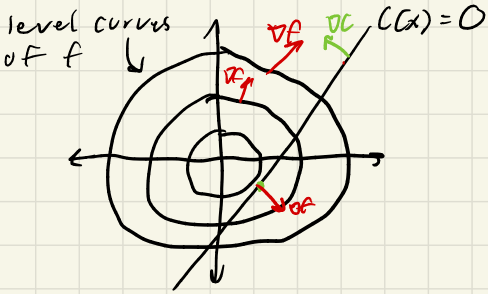
$$\min_x \; f(x) \quad \leftarrow \; f(x): \mathbb{R}^n \rightarrow \mathbb{R}$$

$$\text{s.t.} \;\; c(x) = 0 \quad \leftarrow \; c(x): \mathbb{R}^n \rightarrow \mathbb{R}^m$$

- First-Order Necessary Conditions:

  1) Need $\nabla f(x) = 0$ in <u>free</u> directions

  2) Need $c(x) = 0$



$$f(x): \mathbb{R}^2 \rightarrow \mathbb{R}$$

$$c(x): \mathbb{R}^2 \rightarrow \mathbb{R}$$

* Any non-zero component of ∇f must be normal to the constraint surface/manifold

$$\Rightarrow \quad \nabla f + \underbrace{\lambda}_{\text{"Lagrange multiplier"/"Dual variable"}} \nabla c = 0 \quad \text{for some } \lambda \in \mathbb{R}$$

- In general:

$$\frac{\partial f}{\partial x} + \lambda^T \frac{\partial c}{\partial x} = 0 \, , \quad \lambda \in \mathbb{R}^m$$

- Based on this gradient condition, we define:

$$\underbrace{L(x,\lambda)}_{\text{"Lagrangian"}} = f(x) + \lambda^T c(x)$$

- Such that:

$$\left.\begin{aligned} \nabla_x L(x,\lambda) &= \nabla f + \left(\frac{\partial c}{\partial x}\right)^T \lambda = 0 \\ \nabla_\lambda L(x,\lambda) &= c(x) = 0 \end{aligned}\right\} \text{"KKT condition"}$$

- We can solve this with Newton:

$$\nabla_x L(x+\Delta x, \lambda+\Delta\lambda) \approx \nabla_x L(x,\lambda) + \frac{\partial^2 L}{\partial x^2}\Delta x + \underbrace{\frac{\partial^2 L}{\partial x \partial \lambda}}_{\left(\frac{\partial c}{\partial x}\right)^T} \Delta\lambda = 0$$

$$\nabla_\lambda L(x+\Delta x, \lambda) \approx \underbrace{c(x) + \frac{\partial c}{\partial x}\Delta x = 0}$$

$$\Rightarrow \frac{\partial c}{\partial x}\Delta x = -c(x)$$

$$\underbrace{\begin{bmatrix} \frac{\partial^2 L}{\partial x^2} & \left(\frac{\partial c}{\partial x}\right)^T \\ \frac{\partial c}{\partial x} & 0 \end{bmatrix}}_{\text{"KKT system"}} \begin{bmatrix} \Delta x \\ \Delta \lambda \end{bmatrix} = \begin{bmatrix} -\nabla_x L(x,\lambda) \\ -c(x) \end{bmatrix}$$

* Gauss-Newton Method:

$$\frac{\partial^2 L}{\partial x^2} = \nabla^2 f + \underbrace{\frac{\partial}{\partial x}\left[\left(\frac{\partial c}{\partial x}\right)^T \lambda\right]}_{\text{This term is expensive to compute.}}$$

- We often drop the 2nd "constraint curvature" term
- Called "Gauss-Newton"
- Slightly slower convergence than full Newton (more iterations) but much cheaper per-iteration.

$\Rightarrow$ Often wins in wall clock time.

* Example

- Start at $[-1,-1]$, $\underbrace{[-3, 2]}_{\text{Newton gets stuck, Gauss-Newton doesn't}}$

* Take Away Message:
  - May still need to regularize $\frac{\partial^2 L}{\partial x^2}$ in Newton, even if $\nabla^2 f > 0$
  - Gauss-Newton is often used in practice.

---

* Inequality Constraints

$$\min_x \quad f(x)$$

$$\text{s.t.} \quad c(x) \geq 0$$

- We'll just look at inequalities for now
- In general, these methods are combined with the previous ones for mixed equality/inequality constraints

* First-Order Necessary Conditions:

  1) $\nabla f = 0$ in the free directions

  2) $c(x) \geq 0$

$$
\text{KKT conditions}
\begin{cases}
\nabla f - \left(\frac{\partial c}{\partial x}\right)^T \lambda = 0 & \leftarrow \text{“stationarity”} \\
c(x) \geq 0 & \leftarrow \text{“primal feasibility”} \\
\lambda \geq 0 & \leftarrow \text{“dual feasibility”} \\
\lambda^T c(x) = 0 & \leftarrow \text{“complementarity”}
\end{cases}
$$

* Intuition:

- If constraint is "active" $\underbrace{(c(x)=0) \Rightarrow \lambda > 0}_{\text{same as equality case}}$

- If constraint is "inactive" $\underbrace{(c(x)>0) \Rightarrow \lambda = 0}_{\text{same as unconstrained case}}$

- Complementarity encodes "on/off" switching

* Algorithms:

- Much harder than equality case
- Can't directly apply Newton to KKT
- Many options w/ trade offs

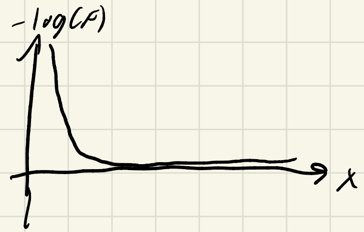
* Active-Set:

  - Have some way of guessing active/inactive constraints
  - Just solve an equality-constrained problem
  - Very fast if you can guess well
  - Very bad otherwise

* Barrier/Interior-Point

  - Replace inequalities with "barrier function" in objective that goes to infinity at constraint boundary

$$\left.\begin{array}{l} \min f(x) \\ \text{s.t. } c(x) \geq 0 \end{array}\right\} \rightarrow \min_x f(x) - \sum_{i=1}^{m} \frac{1}{\rho} \log(c_i(x))$$



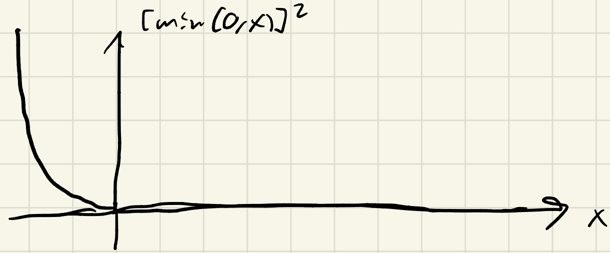
  - Gold standard for small~medium <u>convex</u> problems.

  - Requires lots of tricks/hacks for non-convex problems

* Penalty

- Replace inequality with objective term to penalize violations:

$$\left.\begin{array}{l}\min_x f(x) \\ \text{s.t. } c(x) \geq 0\end{array}\right\} \rightarrow \min_x f(x) + \frac{\rho}{2}\left[\min(0, c(x))\right]^2$$



- Easy to implement
- Has issues with numerical ill-conditioning
- Difficult to achieve high accuracy.