Today:

- Newton in nD
  - ↳ Quasi-Newton
- Nonlinear LSQ
  - ↳ Gauss-Newton
- Constrained optimization
- Interpolation

Announcements:

- CS job talk yesterday
- HW9

Newton's method (n D)

What does Newton's method look like in n dimensions?

$$x_{k+1} = x_k - H_f^{-1}(x_k)\nabla f(x_k) \leftarrow \text{Opt}$$

$$f'(x_k)/f''(x_k)$$

$$x_{k+1} = x_k - J_f^{-1}(x_k) f(x_k) \leftarrow \text{Eqn. solving}$$

$$\frac{f(x_k)}{f'(x_k)}$$

Newton's method (n D): Observations

Drawbacks?

- Hessians !?
- local convergent
- ill-conditioned Hessians $\Rightarrow$ not good

Demo: Newton's method in n dimensions

Quasi-Newton Methods

Secant/Broyden-type ideas carry over to optimization. How?

$$x_{k+1} = x_k - B_k^{-1}(x_k)\nabla f(x_k)$$

BFGS: Secant-type method, similar to Broyden:

$$\mathbf{B}_{k+1} = \mathbf{B}_k + \frac{\mathbf{y}_k\mathbf{y}_k^T}{\mathbf{y}_k^T\mathbf{s}_k} - \frac{\mathbf{B}_k\mathbf{s}_k\mathbf{s}_k^T\mathbf{B}_k}{\mathbf{s}_k^T\mathbf{B}_k\mathbf{s}_k}$$

where

- $\mathbf{s}_k = \mathbf{x}_{k+1} - \mathbf{x}_k$
- $\mathbf{y_k} = \nabla f(\mathbf{x}_{k+1}) - \nabla f(\mathbf{x}_k)$

In-Class Activity: Optimization Methods

In-class activity: Optimization Methods

$$f(\underbrace{x+h}_{\tilde{x}}) = f(x_k) + f'(x_k)\,h + f''(x_k)\cdot\frac{h^2}{2}$$

$$\hat{x} = x+h \qquad \tilde{x} - x_k = h$$

Nonlinear Least Squares: Setup

$r : \mathbb{R}^m \to \mathbb{R}^n$

What if the f to be minimized is actually a 2-norm?

$$f(\mathbf{x}) = \|\mathbf{r}(\mathbf{x})\|_2, \qquad \mathbf{r}(\mathbf{x}) = \mathbf{y} - \mathbf{f}(\mathbf{x})$$

$$\varphi(\vec{x}) = \frac{1}{2} r(\vec{x})^T r(\vec{x})$$

$$\|r\|_2^2$$

$$\frac{\partial}{\partial x_i} \varphi = \frac{1}{2} \sum_{j=1}^{n} \frac{\partial}{\partial x_i} [r_j(\vec{x})]^2 = \sum_{j=1}^{n} \left(\frac{\partial}{\partial x_i} r_j\right) r_j$$

$$\longrightarrow \nabla \varphi = J_r(x)^T \vec{r}(x)$$

Gauss-Newton

r_i are small near min

For brevity: $J := J_r(x)$.

$$H_\varphi(\vec{x}) = J^T(\vec{x}) J(\vec{x}) + \sum_i r_i H_{r_i}(\vec{x})$$

$$\rightsquigarrow \tilde{H}_\varphi(\vec{x}) = J^T(\vec{x}) J(\vec{x})$$

$$x_{k+1} = x_k - \tilde{H}_\varphi^{-1}(\vec{x}_k) \nabla\varphi(x_k)$$

$$= \vec{x}_k - \underbrace{(J^T J)^{-1} J^T \vec{r}}_{\vec{h}_k}$$

$$J^T J h = J^T \vec{r}$$

$$\Leftrightarrow \vec{J} h_k \cong \vec{r}$$

Gauss-Newton: Observations?

Demo: Gauss-Newton

Observations?

Convergence still similar to orig Newton
locally convergent
ill-conditioned Hessians are still bad

Levenberg-Marquardt

If Gauss-Newton on its own is poorly, conditioned, can try Levenberg-Marquardt:

$$\left(J^T J + \underbrace{\mu_k I}_{\text{"regularization term"}}\right) h_k = J^T r_k$$

Constrained Optimization: Problem Setup

Want $\mathbf{x}^*$ so that

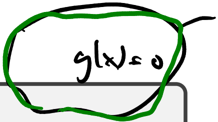
$$f(\mathbf{x}^*) = \min_{\mathbf{x}} f(\mathbf{x}) \quad \text{subject to} \quad \mathbf{g}(\mathbf{x}) = \mathbf{0}$$

No inequality constraints just yet. This is *equality-constrained optimization*. Develop a necessary condition for a minimum.

s is a feasible dir if

$x + \alpha s$ feasible for some $\alpha \in [0, r]$ $\quad g(x) = 0$

Constrained Optimization: Necessary Condition



$$\nabla f(x) \cdot s \geq 0$$

At the bdry where $g(x)=0$:

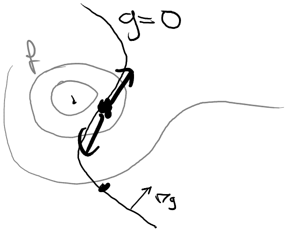
$$-\nabla f(x) \in \operatorname{rowspan} J_g$$

$$-\nabla f(x) = J_g^T \lambda$$

necessary condition

Lagrange Multipliers

unconstr. NC:
$\nabla f(x^*) = 0$



Seen: Need $-\nabla f(\mathbf{x}) = J_{\mathbf{g}}^T \lambda$ at the (constrained) optimum.

Idea: Turn constrained optimization problem for $\mathbf{x}$ into an *unconstrained* optimization problem for $(\mathbf{x}, \lambda)$. How?

$$\mathcal{L}(\vec{x}, \vec{\lambda}) = f(x) + \lambda^T g(x)$$

Lagrange Multipliers: Development

$$\mathcal{L}(\mathbf{x}, \lambda) := f(\mathbf{x}) + \lambda^T \mathbf{g}(\mathbf{x}).$$

Demo: Sequential Quadratic Programming

Inequality-Constrained Optimization

Want $\mathbf{x}^*$ so that

$$f(\mathbf{x}^*) = \min_{\mathbf{x}} f(\mathbf{x}) \quad \text{subject to} \quad \mathbf{g}(\mathbf{x}) = \mathbf{0} \quad \text{and} \quad \mathbf{h}(\mathbf{x}) \leqslant \mathbf{0}$$

This is *inequality-constrained optimization*. Develop a necessary condition for a minimum.