Nonlinear Optimization

Last class: $f: \mathbb{R} \to \mathbb{R}$, minimize $f(x)$.

↳ finding a local min

① Golden Section Search
- If f is unimodal on $[a, b]$ then we can iteratively shrink the interval in which we know x^* lies in.
- Only require evaluation of f.

② Newton's Method

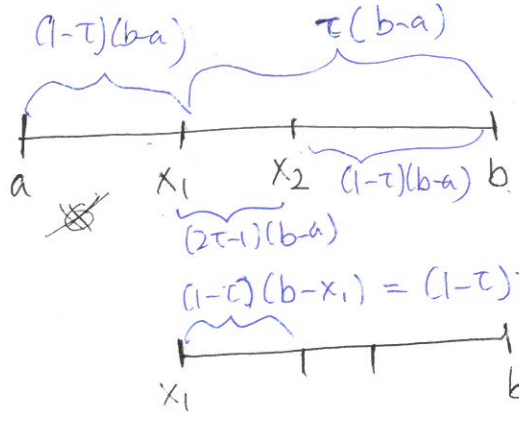
Idea: approximate $f(x)$ using a Quadratic function. And find the critical point of the approximate function

$$x_{k+1} = x_k - \frac{f'(x_k)}{f''(x_k)}$$

~ requires derivative & 2nd derivative of f.

- Does not always converge,
- Does not always converge to a minimum

Derivation of τ in Golden Section Search



$$(2\tau - 1)(b-a) = (1-\tau)\tau(b-a)$$

$$2\tau - 1 = \tau - \tau^2$$

$$\tau^2 + \tau - 1 = 0$$

Solve

$$\tau = 0.618\ldots$$

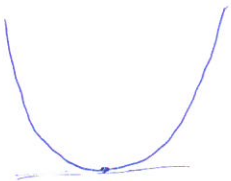
Today : - conditioning
- $f: \mathbb{R}^n \to \mathbb{R}$ - Newton's method
- Gradient Descent

Sensitivity and conditioning

Recall: The problem of finding a root of $f(x)$ has absolute condition number $\left|\frac{1}{f'(x)}\right| \approx \left|\frac{\Delta x}{\Delta y}\right|$

Minimization and root finding are related problems, but minimization problems have worse conditioning.

because $f'(x) \approx 0$ near a minimum.



Root finding: if $|f(\hat{x})| \leq \varepsilon$ then $|\hat{x} - x^*| \leq \left|\frac{\varepsilon}{f'(x^*)}\right|$

Minimization is akin to finding a multiple root:

Taylor series expansion:

$$f(\hat{x}) = f(x^*+h) = f(x^*) + \underbrace{f'(x^*)}_{0} h + \frac{1}{2} f''(x^*) h^2 + O(h^3)$$

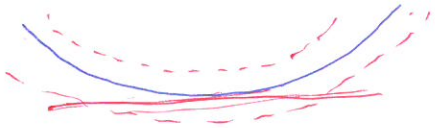
At x^* (minimum) $f'(x^*) = 0$ so

$$f(\hat{x}) \approx f(x^*) + \frac{1}{2} f''(x^*) h^2.$$

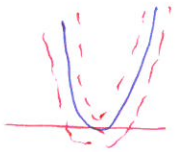
$$h^2 \approx 2\frac{f(\hat{x}) - f(x^*)}{f''(x^*)} \qquad \text{If } f''(x^*) \neq 0$$

$$\Rightarrow \text{If } |f(\hat{x}) - f(x^*)| \leq \epsilon \Rightarrow |\hat{x} - x^*| \leq \sqrt{\frac{2\epsilon}{f''(x^*)}}$$

Obs Smaller 2nd derivative $\Rightarrow$ worse cond.



worse cond.



higher 2nd derivative

better cond

Obs There is a sqrt $\sqrt{\epsilon} > \epsilon$ since ϵ is small

If $f''(x^*) \approx 1$, error will be $\sqrt{\epsilon}$

So even if ϵ is small, say ϵ_{mach},

the solution $\hat{x}$ can be computed to half as many digits of accuracy as the underlying machine precision.

If the derivative $f'(x)$ is available, the problem of ~~find~~ solving $f'(x) = 0$. has cond num: $\left|\frac{1}{f''(x)}\right|$

The sensitivity / conditioning result generalizes to $f: \mathbb{R}^n \to \mathbb{R}$.

Newton's Method for $f: \mathbb{R}^n \to \mathbb{R}$

(minimizing)

Recall: Taylor series expansion for $f: \mathbb{R} \to \mathbb{R}$.

$$f(x+h) = f(x) + f'(x)h + \frac{1}{2}f''(x)h^2 + O(h^3)$$

This result extends to $f: \mathbb{R}^n \to \mathbb{R}$.

$$f(\underline{x}+\underline{s}) = f(\underline{x}) + \nabla f(\underline{x})^T \underline{s} + \frac{1}{2}\underline{s}^T H_f(\underline{x})\,\underline{s}$$

$$\nabla f(\underline{x}) = \begin{bmatrix} \frac{\partial}{\partial x_1} f(\underline{x}) \\ \frac{\partial}{\partial x_2} f(\underline{x}) \\ \vdots \end{bmatrix}$$

Idea: Start with x_k.

- approx $f(\underline{x})$ with $f(\underline{x}_k) + \nabla f(\underline{x}_k)^T \underline{s} + \frac{1}{2}\underline{s}^T H_f(\underline{x}_k)\underline{s}$ (*)
- find the minima of (*) w.r.t $\underline{s}$.

→ (*) is minimized when: (analogous to $f''(x)h = -f'(x)$)

$$\underbrace{H_f(\underline{x}_k)}_{\text{matrix}} \underbrace{\underline{s}_k}_{\text{vector}} = \underbrace{-\nabla f(\underline{x}_k)}_{\text{vector}}$$

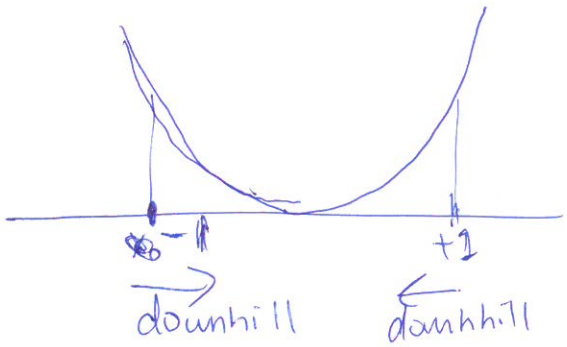
So update: $x_{k+1} = x_k + s_k$.

Steepest Descent ↔ Gradient Descent

We'll discuss a variation of what is in your textbook.

- The negative gradient of a differentiable function points "downhill", or towards $\underline{x}$ ~~points~~ with lower function values $f(\underline{x})$

Why? Demo #1 in 1D: $f(x) = x^2$



$f'(-1) = 2(-1) = -2$

$\Rightarrow -f'(-1)$ is positive

$\Rightarrow$ points to the right (downhill)

$f'(1) = 2(1) = 2$

$\Rightarrow -f'(1)$ is negative

$\Rightarrow$ points to the left (downhill)

Demo #2 Using Taylor's Thm, we can locally approximate f using

$$f(\underline{x}+\underline{s}) = f(\underline{x}) + \nabla f(\underline{x}+\alpha\underline{s})^T s \qquad \alpha \in [0,1]$$

If $\nabla f(\underline{x}) \neq \underline{0}$, then if we let $s = -\nabla f(\underline{x})$

(move $\underline{x}$ toward $-\nabla f(\underline{x})$)

Then, by the continuity of ∇f

$$f(\underline{x}+\underline{s}) = f(\underline{x}) + \underbrace{\nabla f(\underline{x}+\alpha s)}^{\text{Close to } \nabla f(\underline{x})} \left(-\nabla f(\underline{x})\right)$$

$$= f(\underline{x}) + (\text{Something negative})$$

(Note: This is also why $\nabla f(\underline{x}^*) = 0$ at a minimum x^*, otherwise we can move further "downhill".)

In fact $-\nabla f(\underline{x})$ is, locally, the direction of the steepest descent!

$$\underline{x}_{k+1} = \underline{x}_k - \alpha \nabla f(\underline{x}_k)$$

In machine learning, α is called the learning rate.

eg// Reusit midterm → A2 grade prediction

$$A = \begin{bmatrix} q_1^{(1)} & q_2^{(1)} & \cdots & q_5^{(1)} \\ q_1^{(2)} & q_2^{(2)} & \cdots & q_5^{(2)} \\ & & & \end{bmatrix} \quad \underline{b} = \begin{bmatrix} b^{(1)} \\ b^{(2)} \\ \vdots \\ b^{(M)} \end{bmatrix}$$

← each row = one student

student = M

$\underline{b}$: A2 grade

column = one question

problem: find $\underline{x} = \begin{bmatrix} x_1 \\ \vdots \\ x_5 \end{bmatrix}$

to minimize $\|A\underline{x} - \underline{b}\|_2$

We'll treat this as a nonlinear optimization problem.

$$\min_{\underline{x}} f(\underline{x}) \quad \text{where} \quad f(x) = \frac{1}{M}(A\underline{x} - \underline{b})^T(A\underline{x} - \underline{b})$$

Expanding $f(x)$

$$f(\underline{x}) = \frac{1}{M}\sum_{m=1}^{M}\underbrace{\left(q_1^{(m)}x_1 + q_2^{(m)}x_2 + \cdots + q_5^{(m)}x_5 - b^{(m)}\right)}_{A\underline{x}-\underline{b}}{}^2$$

Let's optimize $f(x)$ using gradient descent.

$$\nabla f(\underline{x}) = \begin{bmatrix} \frac{1}{M}\sum_{m=1}^{M} 2q_1^{(m)}\left(q_1^{(m)}x_1 + q_2^{(m)}x_2 + \cdots + q_5^{(m)}x_5 - b^{(m)}\right) \\ \frac{1}{M}\sum_{m=1}^{M} 2q_2^{(m)}\left(q_1^{(m)}x_1 + q_2^{(m)}x_2 + \cdots + q_5^{(m)}x_5 - b\right) \\ \vdots \end{bmatrix}$$

Idea: we start with some $\underline{x}_0$.

Then take gradient descent steps

$$\underline{x}_{k+1} = \underline{x}_k - \alpha \nabla f(\underline{x}_k)$$

Until convergence.

Why gradient descent?

① Instead of computing $\nabla f(\underline{x})$ exactly, we can estimate $\nabla f(\underline{x})$ using a subset of our data in $A, \underline{b}$

② gradient descent also works for more complicated functions like neural networks!