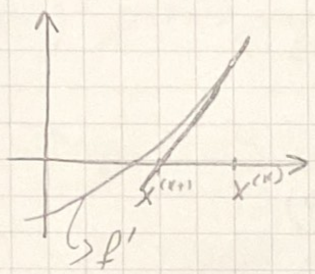
Optimization - 15 - 27/10/23

II Second order methods:

▸ **Newton method:** $x^{(k+1)} = x^{(k)} - a\mathcal{H}(x^{(k)})^{-1} \underbrace{\nabla f(x_k)}_{g_k}$

In R, Newton method is equavalent of finding roots of f':

$$x^{(k+1)} = x^{(k)} - a \frac{f'(x^{(k)})}{f''(x^{(k)})}, \quad a = 1, \quad f''(x^{(k)}) \sim \text{1D Hor...}$$



- Newton method is quadratic in convergance near local minima

▸ **Quasi-Newton method:**

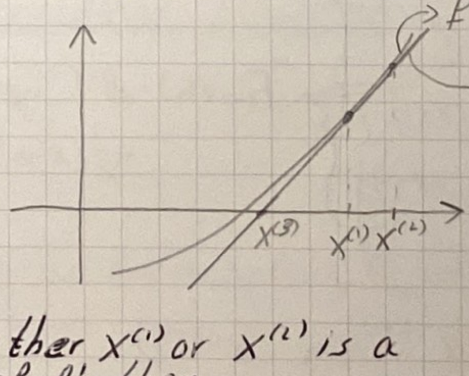
@ $\mathbb{R}$ -> second method: $f \to \min \iff f' = 0$

Suppose I don't want to compute 2nd derivative, then I do approximation of it instead.

$$f''(x^{(k)}) \simeq \frac{f'(x^{(k)}) - f'(x^{(k-1)})}{x^{(k)} - x^{(k-1)}}$$

$$\Rightarrow \boxed{x^{(k+1)} = x^{(k)} - \frac{f'(x^{(k)})(x^{(k)} - x^{(k-1)})}{f'(x^{(k)}) - f'(x^{(k-1)})}}$$

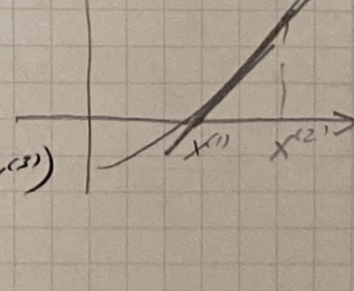
Input: $x^{(1)}, x^{(2)}$

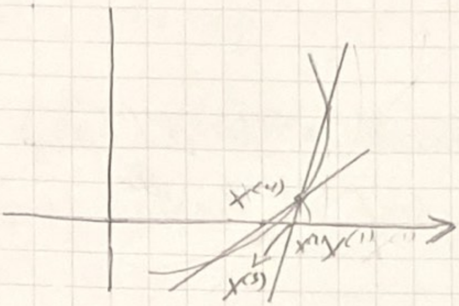


line through $(x^{(1)}, f'(x^{(1)}))$, $(x^{(2)}, f'(x^{(2)}))$ $= \ell(x^{(1)}, x^{(2)})$

If either $x^{(1)}$ or $x^{(2)}$ is a root of f', then $x^{(3)} = \ell(x^{(1)}, x^{(2)}) \cap$ Xaxis is again the root

Otherwise $(x^{(1)}, x^{(2)}) \to (x^{(2)}, x^{(3)})$ and proceed,





Then: If $x^{(1)}, x^{(2)}$ is sufficiently close to a, $f'(a) = 0$, then $x^{(k)} \xrightarrow[k\to\infty]{} a$ and this convergence is quadratic

ⓑ $\mathbb{R}^n$, $x^{(k+1)} = x^{(k)} - a\,Q^{(k)} g^{(k)}$

↳ positive definite matrix $\sim H(x^{(k)})$

- Davidson - Fletcher - Powell (DFP)
- Broyden - Fletcher - Goldfarb - Shanno (BFGS)
- Variations of BFGS method

$z^{(k)} = f(x^{(k)})$, $g^{(k)} = \nabla f(x^{(k)})$

Define: $F_k(x) = z^{(k)} + g^{(k)}(x - x^{(k)}) + \frac{1}{2}(x - x^{(k)})^T \mathcal{H}^{(k)} (x - x^{(k)})$

We want to find $\mathcal{H}^{(k)}$ as an approximation to the hessian $H(x^{(k)})$

- $\mathcal{H}^{(1)} = id$, $k \mapsto k+1$
- Do the line search for F_k this gives $a^{(k)}$:

$x^{(k+1)} = x^{(k)} - a^{(k)} (\mathcal{H}^{(k)})^T g^{(k)}$

- How to define F_{k+1}?

↳ $F_{k+1}(x^{(k+1)}) = z^{(k+1)}$

↳ $\nabla F_{k+1}(x^{(k+1)}) = g^{(k+1)}$

↳ $\nabla F_{k+1}(x^{(k)}) = g^{(k)}$

$F_{k+1}(x) = z^{(k+1)} + g^{(k+1)}(x - x^{(k+1)}) + \frac{1}{2}(x - x^{(k+1)})^T \mathcal{H}^{(k+1)} (x - x^{(k+1)})$

$\Rightarrow \nabla F_{k+1}(x) = g^{(k+1)} + \mathcal{H}^{(k+1)}(x - x^{(k+1)})$

$\Rightarrow \nabla F_{k+1}(x^{(k)}) = g^{(k+1)} + \mathcal{H}^{(k+1)}(x^{(k)} - x^{(k+1)}) = g^{(k)}$

$\Rightarrow \mathcal{H}^{(k+1)} \underbrace{(x^{(k)} - x^{(k+1)})}_{s^{(k)}} = \underbrace{g^{(k)} - g^{(k+1)}}_{g^{(k)}}$ (Notation)

→

Second Equation ~ descrete second derivative in $\mathbb{R}$

$$\boxed{H^{(k+1)} s^{(k)} = y^{(k)}} \quad \text{where, } s^{(k)} = x^{(k)} - x^{(k+1)}$$
$$y^{(k)} = g^{(k)} - g^{(k+1)}$$

We want:
- $H^{(k+1)}$ symmetric
- $H^{(k+1)}$ positive definite

$$(s^{(k)})^T H^{(k+1)} s^{(k)} = \boxed{(s^{(k)})^T y^{(k)} > 0}$$ curvature condition (either has to be checked or forced)

Let's look for $H^{(k+1)}$ in the following form:

$$H^{(k+1)} = H^{(k)} + a u u^T + b v v^T, \quad a, b > 0, \quad u, v \in \mathbb{R}^n$$

*Note: uu^T is positive definite: $w^T(uu^T)w = (u^T w)^T u^T w = \langle u, w\rangle^2 > 0$ if $w \neq 0$, $w \perp n$

Take $u = y^{(k)} = \nabla g^{(k)} - \nabla g^{(k+1)}$, $v = H^{(k)} s^{(k)} = H^{(k)}(x^{(k)} - x^{(k+1)})$

Then a, b are decided from the second equation

$$a = \frac{1}{(y^{(k)})^T s^{(k)}}, \quad b = -\frac{1}{(s^{(k)})^T (H^{(k)})^T s^{(k)}}$$

$$H^{(k+1)} = H^{(k)} + \frac{y^{(k)}(y^{(k)})^T}{(y^{(k)})^T s^{(k)}} - \frac{H^{(k)} s^{(k)} (s^{(k)})^T (H^{(k)})^T}{\underbrace{(s^{(k)})^T (H^{k})^T s^{(k)}}_{\vee\, 0 \;\; H^{(k)} > 0}}$$

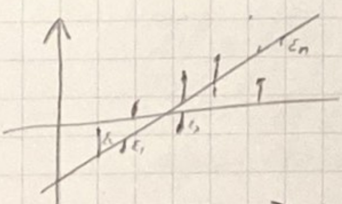
Stochastic Gradient Descent

$$\min_x \frac{1}{n} \sum_{i=1}^{n} f_i(x), \quad x \in \mathbb{R}^m, \quad f_i : \mathbb{R}^m \to \mathbb{R}$$

$x_1, \dots, x_m$

$y_1, \dots, y_m$

*Least-squares optimization: $\frac{1}{n} \|Ax - b\|_2^2 = \frac{1}{n} \sum_{i=1}^{n} (\underset{\mathbb{R}^m}{a_i} x + \underset{\mathbb{R}^m}{b_i})^2$

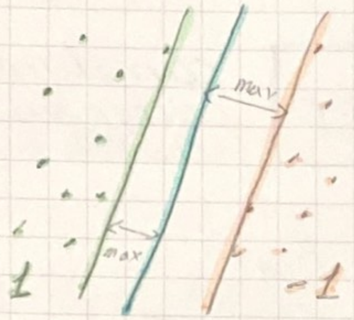


$\varepsilon_1^2 + \varepsilon_2^2 + \dots + \varepsilon_n^2 \to \min$

* $\frac{1}{n}\|Ax-b\|_2^2 + \lambda\|x\|_1 = \frac{1}{n}\sum_{i=1}^{n}(a_i x + b_i)^2 + \sum_{j=1}^{m}|x_j|$

ℓ_1 - least squares

* SVM - Support Vector Machine



$$\frac{1}{2}\|x\|_2^2 + \underbrace{\frac{1}{n}}_{\min}\sum_{i=1}^{n}\max\{0, 1 - \underbrace{y_i}_{\in[-1,1]}(x^T a_i - b)\}$$

* Deep Neural Networks (DNN)

$$\frac{1}{n}\sum_{i=1}^{n} loss(y_i, DNN(x, a_i))$$

training data

weights objective to find

$$x^{(k+1)} = x^{(k)} - \alpha^{(k)}\nabla f = x^{(k)} - \alpha^{(k)}\frac{1}{n}\sum_{i=1}^{n}\nabla f_i(x)$$

steepest descent

-> [Robbins, Monro, 1951]

$$x^{(k+1)} = x^{(k)} - \alpha^{(k)}\underbrace{\nabla f_{i(k)}(x)}_{\text{Example of stochastic gradient}},\ i(k) \text{ we choose at random}$$