L4 - 16/10 - Optimizatio

Last time: $\min f(x)$, $x^{(k+1)} = x^{(k)} + a^{(k)} + d^{(k)}$

Goal: $f(x^{(1)}) > f(x^{(2)}) > \dots > f(x^{(k)}) > \dots$
↳ to reach, choose a descent direction

- $d^{(k)} = -\nabla f(x^{(k)})$ gradient, fastest descent: first order methods
- $d^{(k)} = (H(x^{(k)}))^{-1} \nabla f(x^{(k)})$ Newton's method: second order methods
  ↳ hessian of f

I First order methods: $g^{(k)} := \nabla f(x^{(k)})$ notation

What is the "actual" trajectory in the fastest descent method exact line search?

$f(x^{(k)} - a g^{(k)}) \longrightarrow \min$ w.r.t. a

$$0 = \frac{\partial}{\partial d} f(x^{(k)} - a g^{(k)}) = -(\nabla f(\underbrace{x^{(k)} - a g^{(k)}}_{x^{(k+1)}}))^T = -(g^{(k+1)})^T g^{(k)}$$

▷ Conjugate gradient method

If A is symetric positive-definite matrix, then we can define a new scalar product $\langle x, y \rangle_A = \langle Ax, y \rangle = y^T A x$

One can check that this is:

- bi-linear
- non-degenerate ($\langle x, y \rangle_A \geq 0$ and it is 0 iff $x=0$ or $y=0$)
- symmetric ($\langle x, y \rangle_A = \langle y, x \rangle_A$)
- bilinearity follows from bilinearity of the standard scalar product
- symmetry $\Leftarrow$ symedry of A
- non-dejeneracy $\Leftarrow$ positive definitness of A

Example: $A = \begin{pmatrix} 2 & -1 \\ -1 & 3 \end{pmatrix}$ p.d. matrix

$x = (x_1, x_2)$, $y = (y_1, y_2)$

$$\langle x, y \rangle_A = (x_1, x_2) \begin{pmatrix} 2 & -1 \\ -1 & 3 \end{pmatrix} \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = (x_1, x_2) \begin{pmatrix} 2y_1 - y_2 \\ -y_1 + 3y_2 \end{pmatrix} = \begin{matrix} 2x_1y_1 - x_1y_2 \\ -x_1y_2 + 3x_2y_2 \end{matrix}$$

$(\mathbb{R}^n, \langle \cdot, \cdot \rangle_A)$ n-dimensional vector space =>

=> $\exists$ orthonormal $p_1, \dots, p_n$ s.t. $\langle p_i, p_i \rangle_A = 1$, $i \neq j$ $\langle p_i, p_j \rangle_A = 0$

these directions $p_1, \dots, p_n$ are called conjugate

For the method: 'take quadratic approx. of f'

$$f(x) = \frac{1}{2} x^T \underbrace{A}_{\text{positive definite}} x + bx + c \longrightarrow \min ?$$

for quadratic function: $\left.\begin{matrix} \nabla f(x) = Ax + b \\ H(f) = A \end{matrix}\right\} (*)$

(1) Start with $d^{(1)} = g^{(1)}$ ($= \nabla f(x^{(1)})$, where $x^{(1)}$ starting point)

(2) Compute a analytically:

$$\frac{\partial f(x + ad)}{\partial a} = (\nabla f(x + ad))^T d \overset{(*)}{=} (Ax + aAd + b)^T d$$
$$= a d^T A d + d^T (Ax + b) = 0$$

(3) Since A is positive definite, the $d^T A d = \langle d, d \rangle_A > 0$

$$a = -\frac{d^T (Ax + b)}{\langle d, d \rangle_A}$$

in particular, $a^{(k)} = -\dfrac{d^{(k)T}(Ax^{(k)} + b)}{\langle d^{(k)}, d^{(k)} \rangle_A} = -\dfrac{d^{(k)T} \cdot g^{(k)}}{\langle d^{(k)}, d^{(k)} \rangle_A}$

$$\Rightarrow x^{(k+1)} = x^{(k)} + a^{(k)} d^{(k)}$$

(4) $d^{(k+1)} = -g^{(k+1)} + \beta^{(k)} d^{(k)}$ where $\beta^{(k)}$ is choosen so that $d^{(k+1)}$ is conjugate to $d^{(k)}$

$\iff \langle d^{(k)}, d^{(k+1)} \rangle_A = 0$

$\iff \langle d^{(k)}, -g^{(k+1)} + \beta^{(k)} d^{(k)} \rangle = 0$

$\iff -\langle d^{(k)}, g^{(k+1)} \rangle_A + \beta^{(k)} \langle d^{(k)}, d^{(k)} \rangle_A = 0$

$$\Rightarrow \boxed{\beta^{(k)} = \frac{\langle d^{(k)}, g^{(k+1)} \rangle_A}{\langle d^{(k)}, d^{(k)} \rangle_A} \simeq 0.1}$$

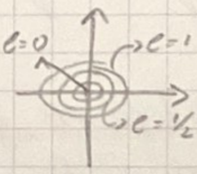
It is a theorem that the conjugate gradient descent converges near local minimum.

Example:

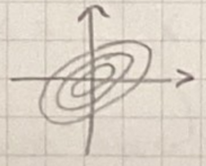
- $A = \begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix}$

$\langle x, x \rangle_A = (x_1, x_2) \begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = (x_1, x_2) \begin{pmatrix} 2x_1 \\ x_2 \end{pmatrix} = 2x_1^2 + x_2^2$

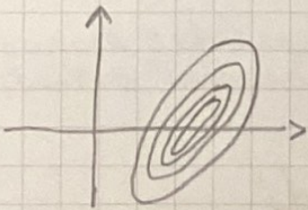
where $2x_1^2 + x_2^2 = c$, $c > 0$ are ellipses



- $A = \begin{pmatrix} 2 & 1 \\ 1 & 1 \end{pmatrix}$, $b = 0$, $c = 0$, $\langle x, x \rangle_A = c$, $c > 0$



- In general



these are level ellipses of $\frac{1}{2} x^T A x + bx + c$

A?

- $Hf(x^{(1)})$
- Update $A \longleftarrow Hf(x^{(k)})$ each step
- It can be given: (**) $Ax = b$ solve this $x = A^{-1}b$

↳ if the size of A is large, then this is inefficient

Instead solve $\frac{1}{2} x^T A x - b x \to \min$

This problem has the same solution as (**)

Use Conjugate Gradient Descent

- If A is not available, then

  Fletcher-Reeves: $\beta^{(k)} = \frac{g^{(k)T} g^{(k)}}{g^{(k-1)T} g^{(k-1)}}$

  Polak-Ribiere: $\beta^{(k)} = \frac{g^{(k)T} (g^{(k)} - g^{(k-1)})}{g^{(k-1)T} g^{(k-1)}}$

▷ Momentum method:

$x^{(k+1)} = x^{(k)} + v^{(k+1)}, \quad v^{(k+1)} = -\alpha g^{(k)} + \underbrace{\beta}_{\text{small}} v^{(k)}$

Problem: "too much" of momentum
↳ overshooting

▷ Nesterov momentum method:

$x^{(k+1)} = x^{(k)} + v^{(k+1)}, \quad v^{(k+1)} = \beta v^{(k)} - \alpha \nabla f(x^{(k)} + \beta v^{(k)})$

▷ Adagrad (adaptive gradient):

$x_i^{(k+1)} = x_i^{(k)} - \frac{\alpha}{\underbrace{\varepsilon}_{10^{-8} \text{ small}} + \sqrt{s_i^{(k)}}} \cdot g_i^{(k)}, \quad s_i^{(k)} = \sum_{j=0}^{k} (g_i^{(j)})^2$

↳ ith coordinate of $x^{(k+1)}$ (avoid zero division)

To compensate small steps (due monotonically non-decresing $s_i^{(k)}$), one can use "accelerated" methods RMS Prop, Adadelta, Adam

▷ a Hyper gradient method:

$\frac{\partial f(x^{(k)})}{\partial \alpha} = -(g^{(k)})^T g^{(k-1)}$

$\Rightarrow \alpha^{(k+1)} = \alpha^{(k)} - \mu \frac{\partial f(x^{(k)})}{\partial \alpha} = \alpha^{(k)} + \mu (g^{(k)})^T \cdot g^{(k-1)}$

↳ μ: hyper gradient learning rate

Exercise:

Do 2 steps of conjugate gradient descent

$f(x,y) = x^2 + xy + y^2 + 5$ starting point $(1,1)$

For the same function/starting point d 2 steps of hyper gradient method.