Conjugate gradient methods

Given matrix $A \in \mathbb{R}^{n \times n}$ symmetric and positive definite, consider linear system

$$(*) \quad Ax = b$$

Solution to $(*)$ also is the solution to the optimization problem

$$(**) \quad \min_{x \in \mathbb{R}^n} \left\{ \frac{1}{2} x^T A x - b^T \cdot x \right\} \longrightarrow \text{denote } \phi(x)$$

Conjugate gradient (CG) methods provide an iterative approach to approximate the solution x^* to $(*)$ / $(**)$ using A-conjugate directions

Definition The vectors $\{p_0, \dots, p_k\}$ are A-conjugate if $p_i \neq 0$, $i = 0:k$ and

$$p_i^T A p_j = 0, \quad \forall i \neq j$$

Property A-conjugate vectors are linearly independent.

Notice $c_0 p_0 + \dots + c_k p_k = 0 \; / \; p_i^T A \cdot \Rightarrow c_i = 0$

Assume that $p_0, \dots, p_{n-1}$ are A-conjugate vectors. (thus form a basis to $\mathbb{R}^n$).

Given an initial guess $x_0 \in \mathbb{R}^n$, define

$$x_{k+1} = x_k + \alpha_k p_k, \qquad k = 0, 1, \dots, n-1$$

where $\alpha_k \in \mathbb{R}$ (positive or negative!) is obtained as solution to

$$\min_{\alpha \in \mathbb{R}} \phi(x_k + \alpha p_k) = \frac{1}{2}(x_k + \alpha p_k)^T A (x_k + \alpha p_k) - b^T (x_k + \alpha p_k)$$

<u>Notation</u> for $x \in \mathbb{R}^n$, $\nabla \phi(x) = Ax - b = \underline{r(x)}$

$r(x)$ is called residual.

$$\varphi(\alpha) = \phi(x_k + \alpha p_k) = \frac{1}{2}(p_k^T A p_k)\alpha^2 + \alpha(p_k^T A x_k - b^T p_k) + \phi(x_k)$$

Minimizer

$$\boxed{\alpha_k = -\frac{p_k^T r_k}{p_k^T A p_k}}$$

Theorem Let $A \in \mathbb{R}^{n\times n}$ symmetric and positive definite matrix and $p_0, \dots, p_{n-1}$ A-conjugate vectors in $\mathbb{R}^n$.

Then for any $x_0 \in \mathbb{R}^n$, the sequence

$$x_{k+1} = x_k + \alpha_k p_k, \qquad \alpha_k = -\frac{p_k^T r_k}{p_k^T A p_k}$$

converges to the solution x^* of the problem $(*)/(**)$ in <u>at most</u> n steps (iterations).

Proof Since $\{p_0, \dots, p_{n-1}\}$ are linearly independent, they form a basis to $\mathbb{R}^n$. Then

$$x^* - x_0 = c_0 p_0 + \dots + c_{n-1} p_{n-1}$$

where the coefficients c_k are obtained as

$$c_k = \frac{p_k^T A(x^* - x_0)}{p_k^T A p_k} = \frac{p_k^T (b - A x_0)}{p_k^T A p_k} = -\frac{p_k^T r_0}{p_k^T A p_k}$$

Notice that $x_k = x_0 + \alpha_0 p_0 + \dots + \alpha_{k-1} p_{k-1}$ implies $p_k^T A (x_k - x_0) = 0 \Rightarrow p_k^T r_k = p_k^T r_0$

thus $\underline{c_k = \alpha_k}$

<u>Remark</u> Eigenvectors associated with distinct eigenvalues are othogonal <u>AND</u> A - orthogonal :

$$A v_i = \lambda_i v_i \qquad \text{with } \lambda_i \neq \lambda_j$$
$$A v_j = \lambda_j v_j$$

then

$$\left.\begin{aligned} v_j^T A v_i &= \lambda_i v_j^T v_i \\ v_i^T A v_j &= \lambda_j v_i^T v_j \end{aligned}\right\} \Rightarrow (\lambda_i - \lambda_j) v_i^T v_j = 0$$

Thus $v_i^T v_j = 0$ and $v_i^T A v_j = 0$.

Denote $P = [p_0, \ldots, p_{n-1}]$ where $p_0, \ldots, p_{n-1}$ are A - conjugate. Then

$P^T A P$ is a diagonal matrix.

Eigenvectors of the matrix A may be used as A-conjugate directions :
however, finding the eigenvectors is very expensive. The CG method provides an efficient approach

Theorem (subspace minimization)

Let $x_0 \in \mathbb{R}^n$, $x_{k+1} = x_k + \alpha_k p_k$, $\alpha_k = -\dfrac{p_k^T r_k}{p_k^T A p_k}$

where $p_0, \dots, p_{n-1}$ are A-conjugate.

Denote $S_k = \text{span}\{p_0, \dots, p_{k-1}\}$

Then i) $r_k^T \cdot p_i = 0$, $i = 0:k-1$

(thus $r_k \perp S_k$)

ii) x_k is solution to the minimization problem

$$\min_{x \in \{x_0 + S_k\}} \phi(x)$$

Proof : i) $x_1 = x_0 + \alpha_0 p_0$

$r_1 = Ax_1 - b = r_0 + \alpha_0 A p_0$ (since $r_0 = Ax_0 - b$)

$p_0^T \cdot r_1 = p_0^T \cdot r_0 + \alpha_0 p_0^T A p_0 = p_0^T \cdot r_0 - p_0^T \cdot r_0 = 0$

Remark $x_{k+1} = x_k + \alpha_k p_k \Rightarrow \boxed{r_{k+1} = r_k + \alpha_k A p_k}$

By induction, assume that

$r_{k-1}^T \cdot p_i = 0$, $i = 0:k-2$

and <u>show that</u> $r_k^T \cdot p_i = 0, \quad i = 0:k-1$

indeed, since $r_k = r_{k-1} + \alpha_{k-1} A p_{k-1}$

then $r_k^T \cdot p_i = r_{k-1}^T \cdot p_i + \alpha_{k-1} p_{k-1}^T A p_i = 0$

for $\underline{i = 0:k-2}$

For $i = k-1$, $r_k^T \cdot p_{k-1} = r_{k-1}^T \cdot p_{k-1} + \alpha_{k-1} p_{k-1}^T A p_{k-1} = 0$

since $\alpha_{k-1} = - \dfrac{p_{k-1}^T \cdot r_{k-1}}{p_{k-1}^T A p_{k-1}}$

<u>Proof of ii)</u> Let $x_k^* = \min\limits_{\{x_0 + S_k\}} \phi(x)$

Then $x_k^* = x_0 + c_0 p_0 + \cdots + c_{k-1} p_{k-1}$

$\varphi(c_0, \ldots, c_{k-1}) = \phi(x_0 + c_0 p_0 + \cdots + c_{k-1} p_{k-1})$

$$\frac{\partial \varphi}{\partial c_i} = \nabla \phi \cdot p_i = (x_0 + c_0 p_0 + \cdots + c_{k-1} p_{k-1})^T \cdot A p_i - b^T \cdot p_i = \cancel{\ldots + c_i p_i^T A p_i}$$

$$\Rightarrow \frac{\partial \varphi}{\partial c_i} = 0 \Longrightarrow + p_i^T r_0 + c_i p_i^T A p_i = 0$$

$$\Rightarrow c_i = - \frac{p_i^T r_0}{p_i^T A p_i} = - \frac{p_i^T \cdot r_i}{p_i^T A p_i}$$

$$\Rightarrow \boxed{c_i = \alpha_i}$$

Conjugate gradient iteration

→ generates a new vector p_k in terms of only p_{k-1} and such that p_k is A-orthogonal on all $p_0, \dots, p_{k-1}$.

Define $\boxed{p_0 = -r_0}$ → the steepest descent direction.

Set $\boxed{p_k = -r_k + \beta_k p_{k-1}}$

where β_k is defined such that

$$p_{k-1}^T A p_k = 0 \implies \boxed{\beta_k = \frac{r_k^T A p_{k-1}}{p_{k-1}^T A p_{k-1}}}$$

Theorem Assume that $x_k \neq x^*$. Then

i) $r_k^T \cdot r_i = 0$, $i = 0, 1, \dots, k-1$

ii) $\mathrm{Span}\{r_0, \dots, r_k\} = \mathrm{Span}\{r_0, Ar_0, \dots, A^k r_0\} \overset{def}{=} \mathcal{K}(r_0, k)$ Krylov space

iii) $\mathrm{Span}\{p_0, \dots, p_k\} = \mathrm{Span}\{r_0, Ar_0, \dots, A^k r_0\}$

iv) $\boxed{p_k^T A p_i = 0}$, $i = 0, 1, \dots, k-1$

Therefore, the iteration $\{x_k\}$ converges to x^* in at most n steps.

→ proof by induction (Th. 5.3 in textbook by Nocedal & Wright: Numerical Optimiz.)

Conjugate Gradient Algorithm (initial version)

Given x_0 initial guess

$r_0 = A x_0 - b$

$p_0 = -r_0$

$k = 0$

while $r_k \neq 0$

$$\alpha_k = -\frac{r_k^T \cdot p_k}{p_k^T A p_k}$$

$$x_{k+1} = x_k + \alpha_k p_k$$

$$r_{k+1} = A x_{k+1} - b$$

$$\beta_{k+1} = \frac{r_{k+1}^T A p_k}{p_k^T A p_k}$$

$$p_{k+1} = -r_{k+1} + \beta_{k+1} p_k$$

$k = k+1$

end

→ computational cost per iteration

$A p_k = O(n^2)$, $A x_{k+1} = O(n^2)$.

may be refined with some useful remarks.

Remarks

For $\alpha_k = -\dfrac{r_k^T p_k}{p_k^T A p_k}$

$$\left.\begin{array}{ll}\text{Since} & p_k = -r_k + \beta_k p_{k-1} \\ \text{and} & r_k^T \cdot p_i = 0, \quad i = 0:k-1\end{array}\right\} \Rightarrow r_k^T p_k = -r_k^T \cdot r_k$$

Thus, $\boxed{\alpha_k = \dfrac{r_k^T \cdot r_k}{p_k^T A p_k}}$

For $r_{k+1} = A x_{k+1} - b$

use $r_{k+1} = A(x_k + \alpha_k p_k) - b = r_k + \alpha_k A p_k$

$$\boxed{r_{k+1} = r_k + \alpha_k A p_k}$$

For $\beta_{k+1} = \dfrac{r_{k+1}^T A p_k}{p_k^T A p_k}$

use $A p_k = \dfrac{1}{\alpha_k}(r_{k+1} - r_k) = \dfrac{p_k^T A p_k}{r_k^T \cdot r_k}(r_{k+1} - r_k)$

Then $\boxed{\beta_{k+1} = \dfrac{r_{k+1}^T \cdot r_{k+1}}{r_k^T \cdot r_k}}$

Conjugate Gradient Algorithm (practical version)

x_0 initial guess

$r_0 = A x_0 - b$

$p_0 = - r_0$

$k = 0$

while $r_k \neq 0$

next iterate

$$\alpha_k = \frac{r_k^T \cdot r_k}{p_k^T \cdot A p_k}$$

$$x_{k+1} = x_k + \alpha_k p_k$$

$$r_{k+1} = r_k + \alpha_k A p_k$$

next direction

$$\beta_{k+1} = \frac{r_{k+1}^T r_{k+1}}{r_k^T r_k}$$

$$p_{k+1} = - r_{k+1} + \beta_{k+1} p_k$$

$k = k+1$

end

For implementation work with 3 pairs of vectors

(x, x_{new})

(r, r_{new})

(p, p_{new})

Remarks: CG algorithm generates at once with x_{k+1} the next direction p_{k+1}

Cost per iteration: $A p_k \to O(n^2)$, $p_k^T \cdot (A p_k) \to n$

$r_{k+1}^T \cdot r_k \to n$

→ No need for explicit matrix A, just the ability to evaluate matrix·vector product Ap