Preconditioned conjugate gradient method

Solve $A\vec{x} = \vec{b}$, A sym. posdef.

If one can find Q sym. posdef. which is easy to invert and approximate A well, then, writing

$$Q^{-1} = S S^T$$

$$\underbrace{S^T A S}_{\hat{A}} \underbrace{S^{-1}\vec{x}}_{\hat{\vec{x}}} = \underbrace{S^T \vec{b}}_{\hat{\vec{b}}}$$

$\hat{A}$ ↳ sym. posdef.

If $Q = A$ was true, then $\hat{A} = S^T S^{-T} S^{-1} S = I$

$\Rightarrow$ If Q approximates A well, then we expect $\kappa(\hat{A}) \ll \kappa(A)$

Conjugate gradient for $\hat{A}\hat{\vec{x}} = \hat{\vec{b}}$

In each iteration

$$\langle \vec{r}^{(k)}, \vec{v}^{(k)} \rangle = \langle \vec{r}^{(k)}, \vec{r}^{(k)} \rangle$$

$$\vec{r}^{(k)} = \vec{b} - A\vec{x}^{(k)}$$

① $\hat{t}_k = \dfrac{\langle \hat{\vec{r}}^{(k)}, \hat{\vec{r}}^{(k)} \rangle}{\langle \hat{\vec{v}}^{(k)}, \hat{A}\hat{\vec{v}}^{(k)} \rangle}$

② $\hat{\vec{x}}^{(k+1)} = \hat{\vec{x}}^{(k)} + \hat{t}_k \hat{\vec{v}}^{(k)}$

③ $\hat{\vec{r}}^{(k+1)} = \hat{\vec{r}}^{(k)} - \hat{t}_k \hat{A}\hat{\vec{v}}^{(k)}$

$$\vec{r}^{(k)} = \vec{b} - A\vec{x}^{(k)}$$
$$= S^{-T}\hat{\vec{b}} - S^{-T}\hat{A}\cancel{S^{-1}S}\,\hat{\vec{x}}^{(k)}$$
$$= S^{-T}\left(\hat{\vec{b}} - \hat{A}\hat{\vec{x}}^{(k)}\right)$$

④ $\hat{s}_k = \dfrac{\langle \hat{\vec{r}}^{(k+1)}, \hat{\vec{r}}^{(k+1)} \rangle}{\langle \hat{\vec{r}}^{(k)}, \hat{\vec{r}}^{(k)} \rangle}$

⑤ $\hat{\vec{v}}^{(k+1)} = \hat{\vec{r}}^{(k+1)} + \hat{s}_k \hat{\vec{v}}^{(k)}$

Notation: $\vec{x}^{(k)} = S\hat{\vec{x}}^{(k)}$ $\quad \vec{v}^{(k)} = S\hat{\vec{v}}^{(k)}$ $\quad \vec{r}^{(k)} = S^{-T}\hat{\vec{r}}^{(k)}$

$S \cdot$ ② $\quad \vec{x}^{(k+1)} = \vec{x}^{(k)} + \hat{t}_k \vec{v}^{(k)}$

S. ⑤ $\vec{v}^{(k+1)} = S\,\hat{\vec{r}}^{(k+1)} + \hat{s}_k\,\vec{v}^{(k)}$ $\qquad \langle x, By\rangle = \langle B^T x, y\rangle$

$$= S\,S^T\,\vec{r}^{(k+1)} + \hat{s}_k\,\vec{v}^{(k)}$$

$$= Q^{-1}\,\vec{r}^{(k+1)} + \hat{s}_k\,\vec{v}^{(k)}.$$

①
$$\hat{t}_k = \frac{\langle \hat{\vec{r}}^{(k)}, \hat{\vec{r}}^{(k)}\rangle}{\langle \hat{\vec{v}}^{(k)}, \hat{A}\,\hat{\vec{v}}^{(k)}\rangle} = \frac{\langle S^T\vec{r}^{(k)}, S^T\vec{r}^{(k)}\rangle}{\langle S^{-1}\vec{v}^{(k)}, \hat{A}\,S^{-1}\vec{v}^{(k)}\rangle}$$

$$= \frac{\langle \vec{r}^{(k)}, S\,S^T\vec{r}^{(k)}\rangle}{\langle \vec{v}^{(k)}, S^{-T}\hat{A}\,S^{-1}\vec{v}^{(k)}\rangle} = \frac{\langle \vec{r}^{(k)}, Q^{-1}\,\vec{r}^{(k)}\rangle}{\langle \vec{v}^{(k)}, A\,\vec{v}^{(k)}\rangle}.$$

- Choose the preconditioner Q

  Jacobi preconditioner: Q = diagonal of A

- —

5.1 Power method for matrix eigenvalue

Let A be an $n\times n$ matrix (possibly w/ complex entries).
Then the characteristic polynomial of A is

$$\det(A - \lambda I)$$

(deg-n poly. in λ). It has n complex roots (counting multiplicities), which are the (complex) eigenvalues of A.

- To find eigenvalues, one can calculate coefficients of $\det(A-\lambda I)$ and apply a polynomial root finding algorithm. However, this doesn't work for large n because polynomial roots can be very sensitive to polynomial coefficients.

Power method

Suppose the eigenvalues $\lambda_1, \cdots, \lambda_n$ satisfy

$$|\lambda_1| > |\lambda_2| \geq |\lambda_3| \geq \cdots \geq |\lambda_n|$$

and A is diagonalizable over $\mathbb{C}$, i.e., $\exists$ basis $\vec{u}^{(1)}, \cdots, \vec{u}^{(n)} \in \mathbb{C}^n$ s.t.

$$A \vec{u}^{(i)} = \lambda_i \vec{u}^{(i)} \qquad i = 1, \cdots, n$$

Then we take an initial vector $\vec{x}^{(0)} \in \mathbb{C}^n$, write

$$\vec{x}^{(0)} = a_1 \vec{u}^{(1)} + \cdots + a_n \vec{u}^{(n)}$$ assume $a_1 \neq 0$

Do iteration

$$\vec{x}^{(k+1)} = A \vec{x}^{(k)}$$

Then

$$\begin{aligned}\vec{x}^{(k)} = A^k \vec{x}^{(0)} &= a_1 A^k \vec{u}^{(1)} + \cdots + a_n A^k \vec{u}^{(n)} \\ &= a_1 \lambda_1^k \vec{u}^{(1)} + a_2 \lambda_2^k \vec{u}^{(2)} + \cdots + a_n \lambda_n^k \vec{u}^{(n)} \\ &= a_1 \lambda_1^k \left(\vec{u}^{(1)} + \frac{a_2}{a_1} \left(\frac{\lambda_2}{\lambda_1}\right)^k \vec{u}^{(2)} + \cdots + \frac{a_n}{a_1} \left(\frac{\lambda_n}{\lambda_1}\right)^k \vec{u}^{(n)} \right)\end{aligned}$$

$|\cdot| < 1$

As $k \to \infty$, $\vec{x}^{(k)}$ "converges" to the direction $\vec{u}^{(1)}$.

To get a numerical approximation of λ_1

$$\vec{x}^{(k)} \approx a_1 \lambda_1^k \vec{u}^{(1)} \qquad \vec{x}^{(k+1)} \approx a_1 \lambda_1^{k+1} \vec{u}^{(1)}$$

By taking a linear function ϕ on $\mathbb{C}^n$ $\qquad \phi(\vec{x}) = c_1 x_1 + \cdots + c_n x_n$

(such that $\phi(\vec{u}^{(1)}) \neq 0$)

$$\phi(\vec{x}^{(k+1)}) \approx \phi(\lambda_1 \vec{x}^{(k)}) = \lambda_1 \phi(\vec{x}^{(k)})$$

$$\Rightarrow \lambda_1 \approx \frac{\phi(\vec{x}^{(k+1)})}{\phi(\vec{x}^{(k)})}$$

- Power method can only give the largest eigenvalue λ_1.

Inverse power method

Suppose $|\lambda_1| \geq |\lambda_2| \geq \cdots \geq |\lambda_{n-1}| > |\lambda_n| > 0$

$A\vec{u} = \lambda \vec{u}$

$\frac{1}{\lambda}\vec{u} = A^{-1}\vec{u}$

Then A^{-1} has eigenvalues

$\lambda_n^{-1}, \cdots, \lambda_1^{-1}$ which satisfy

$$|\lambda_n^{-1}| > |\lambda_{n-1}^{-1}| \geq \cdots \geq |\lambda_1^{-1}|$$

Applying power method to A^{-1}, i.e.,

$\vec{x}^{(k+1)} = A^{-1}\vec{x}^{(k)}$ (that is, solving $\vec{x}^{(k+1)}$ from $A\vec{x}^{(k+1)} = \vec{x}^{(k)}$)

we can find λ_n^{-1} which gives λ_n (the smallest eigenvalue of A).

Shifted inverse power method

Take $\mu \in \mathbb{C}$. The matrix $(A - \mu I)^{-1}$

has eigenvalues

$(\lambda_1 - \mu)^{-1}, \cdots, (\lambda_n - \mu)^{-1}$

$A\vec{u} = \lambda\vec{u}$

$(A - \mu I)\vec{u} = (\lambda - \mu)\vec{u}$

Applying power method to $(A - \mu I)^{-1}$, we get the $(\lambda_i - \mu)^{-1}$

w/ the largest absolute value among $(\lambda_1 - \mu)^{-1}, \cdots, (\lambda_n - \mu)^{-1}$

i.e., we get the eigenvalue λ_i of A which is the closest to μ