Welcome to Ma221! (Apr26)

Chebyshev Polynomials

Convergence of CG

Accelerating Splitting Methods

Recall Krylov Subspace Methods:

seek "best" x_k approximating $A^{-1}b$

in $\mathcal{K}_k(A,b) = \text{span}\{b, Ab, A^2b, \ldots A^{k-1}b\}$

$= \{p_{k-1}(A)b,\ p_{k-1} \text{ degree} \le k-1\}$

CG: "best" x_k minimizes

$$\|r_k\|_{A^{-1}}^2 = \|Ax_k - b\|_{A^{-1}}^2$$

$$= (Ax_k - b)^T A^{-1}(Ax_k - b)$$

$$= (Ap_{k-1}(A)b - b)^T A^{-1}(Ap_{k-1}(A)b - b)$$

$$= (q_k(A)b)^T A^{-1}(q_k(A)b)$$

$q_k(A) = I - A \cdot p_{k-1}(A)$

$=$ polynomial of degree $\le k$ with $q_k(0) = 1$

$$= b^T q_k(A) A^{-1} q_k(A) b$$

$$= b^T (q_k(A))^2 A^{-1} b$$

A spd $\Rightarrow$ $A = Q \Lambda Q^T$ $\quad y = Q^T b$

$$= y^T q_k^2(\Lambda) \Lambda^{-1} y$$

$$= \sum_{i=1}^{n} q_k^2(\lambda_i) \lambda_i^{-1} y_i^2$$

So "best" x_k corresponds to best $q_k(x)$ i.e. minimizes weighted sum of squares $q_k^2(\lambda_i)$ over evals of A

$$\leq \max_i q_k^2(\lambda_i) \cdot \sum_{i=1}^{n} \lambda_i^{-1} y_i^2$$

$$= \max_i q_k^2(\lambda_i) \cdot y^T \Lambda^{-1} y$$

$$= \max_i q_k^2(\lambda_i) \cdot \|b\|_{A^{-1}}^2$$

$$\frac{\|Ax_k - b\|_{A^{-1}}}{\|b\|_{A^{-1}}} \leq \max_i q_k(\lambda_i)$$

So seek q_k where $q_k(0) = 1$ and $|q_k(\lambda_i)|$ small for all evals λ_i of A

Use Chebyshev polynomials
to find good $p_k(x)$

Also use Chebyshev polynomials
to accelerate splitting Methods

$x_k = R x_{k-1} + c$ where exact

$x = R x + c$

and $x_k \to x$ when $\|R\| < 1$

$$x_k - x = R(x_{k-1} - x)$$
$$= R^k (x_0 - x)$$

Seek $y_k = \sum_{i=0}^{k} c_{ki} x_i$

that converges to x faster
than x_k

If $x_0 = x \Rightarrow x_i = x \Rightarrow$

$$x = \sum_{i=0}^{k} c_{ki}\, x \Rightarrow 1 = \sum_{i=0}^{k} c_{ki}$$

$$y_k - x = \sum_{i=0}^{k} c_{ki} x_i - x$$
$$= \sum_{i=0}^{k} c_{ki} (x_i - x)$$

$$= \sum_{i=0}^{k} c_{ki} R^i (x_0 - x)$$

(***) $y_k - x = p_k(R)(x_0 - x)$

Seek polynomial $p_k(R)$
that is small at evals of R
and $p_k(1) = 1$

again use (scaled) Chebyshev
polynomials

don't need to keep $x_0, x_1, \ldots, x_k$,
just $y_{k-2}, y_{k-1}, y_k \Rightarrow$ cheap

Def: The m^{th} Chebyshev polynomial
is defined by 3-term recurrence

$T_0(z) = 1$, $T_1(z) = z$

$T_m(z) = 2 \cdot z \cdot T_{m-1}(z) - T_{m-2}(z)$

eg $T_2(z) = 2z^2 - 1$ etc

Lemma (Lemma 6.7 in text)

(a) $T_m(1) = 1$

(b) $T_m(z) = 2^m z^m + O(z^{m-1})$

(c) $T_m(\cos y) = \cos(m \cdot y)$

applies when $|z| \le 1$, $\Rightarrow$

if $|z| \le 1$ then $|T_m(z)| \le 1$

(d) $T_m(\cosh y) = \cosh(m \cdot y)$

applies to $T_m(z)$, $z \ge 1$

(e) If m is even (resp odd) then

$T_m(z)$ is even (resp odd) polynomial

$T_m(-z) = (-1)^m T_m(z)$

(f) If $|z| \ge 1$,

$$T_m(z) = \frac{1}{2}\left[(z + \sqrt{z^2-1})^m + \frac{1}{(z+\sqrt{z^2-1})^m} \right]$$

(g) $T_m(1+\varepsilon) \ge .5\,(1 + m\sqrt{2\varepsilon})$ if $\varepsilon > 0$

see Fig 6.6 in text

Apply to convergence analysis of CG

using

$$(*) \quad \frac{\|Ax_k - b\|_{A^{-1}}}{\|b\|_{A^{-1}}} \le \max_i |q_k(\lambda_i)|$$

$q_k(0) = 1$

$0 < \lambda_{min} \le \lambda_{max}$ be eval range of A

$\lambda_{min} \le z \le \lambda_{max} \Rightarrow$

$$-1 \le \frac{\lambda_{max} + \lambda_{min} - 2z}{\lambda_{max} - \lambda_{min}} \le 1$$

$$\Rightarrow \text{let } q_k(z) = \frac{T_k\left(\frac{\lambda_{max}+\lambda_{min}-2z}{\lambda_{max}-\lambda_{min}}\right)}{T_k\left(\frac{\lambda_{max}+\lambda_{min}}{\lambda_{max}-\lambda_{min}}\right)}$$

$q_k(0) = 1$ and for $\lambda_{min} \le z \le \lambda_{max}$

$$|q_k(z)| \le \frac{1}{T_k\left(\frac{\lambda_{max}+\lambda_{min}}{\lambda_{max}-\lambda_{min}}\right)} \quad \text{by prop c in Lemma}$$

$$= \frac{1}{T_k\left(\frac{\kappa+1}{\kappa-1}\right)} \qquad \kappa = \frac{\lambda_{max}}{\lambda_{min}}$$

$$= \frac{1}{T_k\left(1 + \frac{2}{\kappa-1}\right)}$$

$$\le \frac{2}{1 + \frac{2k}{\sqrt{\kappa-1}}} \quad \text{by prop g in Lemma}$$

$\Rightarrow$ need $k = O(\sqrt{\kappa})$ steps to converge

Use polynomials to accelerate splitting methods like SOR

Need a polynomial $p_k(z)$:

$p_k(z)$ small for z = eval of R

and $p_k(1) = 1$

If Splitting method converges

need all $|\lambda_i| < 1$ i.e. $\rho(A) < 1$

spectral radius

Chebyshev polynomial properties

assume z real

$\Rightarrow$ assume R has real evals

let ρ satisfy

$$-1 < -\rho \leq \lambda_{min}(R) \leq \lambda_{max}(R) \leq \rho < 1$$

Need $\lambda_{min}(R)$ and $\lambda_{max}(R)$, or ρ

to derive algorithm

eg. Model problem

limits applicability, but sometimes works

$$\rho_k(z) = \frac{T_k(z/\rho)}{T_k(1/\rho)} \quad = 1 \text{ at } z=1$$

if $|z| \le \rho \Rightarrow |\rho_k(z)| \le \dfrac{1}{|T_k(1/\rho)|}$

How much faster can this converge than $x_{k+1} = R x_k + c$?

$\rho = 1-\varepsilon$, use prop g of Lemma

$$\frac{1}{|T_k(1/\rho)|} = \frac{1}{|T_k(1/(1-\varepsilon))|}$$

$$= \frac{1}{|T_k(1+\frac{\varepsilon}{1-\varepsilon})|}$$

$$\le \frac{2}{1+k\sqrt{2\varepsilon/(1-\varepsilon)}}$$

$$\sim 2(1-k\sqrt{2\varepsilon})$$

contrast with original splitting method $\rho^k = (1-\varepsilon)^k \cong 1-k\varepsilon$

i.e. $\sqrt{\cdot}$ as many steps to converge,

an asymptotic improvement

Implement cheaply using

3-term Recurrence

$$T_{k+1}(z) = 2z\,T_k(z) - T_{k-1}(z)$$

scale to get recurrence for $\rho_k(z)$

$$\nu_k = \frac{1}{T_k(1/\rho)}$$

$$\rho_{k+1}(z) = \nu_{k+1}\cdot T_{k+1}(z/\rho)$$

$$= \nu_{k+1}\left(2\left(\frac{z}{\rho}\right)T_k\left(\frac{z}{\rho}\right) - T_{k-1}\left(\frac{z}{\rho}\right)\right)$$

$$= \nu_{k+1}\left(2\left(\frac{z}{\rho}\right)\frac{\rho_k(z)}{\nu_k} - \frac{\rho_{k-1}(z)}{\nu_{k-1}}\right)$$

$$= \left(\frac{2\nu_{k+1}}{\rho\cdot\nu_k}\right)z\cdot\rho_k(z) - \left(\frac{\nu_{k+1}}{\nu_{k-1}}\right)\rho_{k-1}(z)$$

$$= \alpha_k\cdot z\cdot\rho_k(z) + \beta_k\rho_{k-1}(z)$$

Note that $\frac{1}{\nu_k} = T_k(1/\rho)$ can be computed cheaply by 3-term recurrence, so α_k, β_k cheap

Apply 3-term recurrence to

(fact) $y_{k+1} - x = p_{k+1}(R)\cdot(x_0 - x)$

$$= \alpha_k R p_k(R)(x_0 - x) + \beta_k p_{k-1}(R)(x_0 - x)$$

$$= \alpha_k R (y_k - x) + \beta_k (y_{k-1} - x)$$

$$= \alpha_k R y_k + \beta_k y_{k-1} - \alpha_k R x - \beta_k x$$

$$y_{k+1} = \alpha_k R y_k + \beta_k y_{k-1} \underbrace{- \alpha_k R x - \beta x + x}$$

simplify last 3 terms using

$x = Rx + c$

$$-\alpha_k R x - \beta x + x = -\alpha_k (x - c) - \beta_k x + x$$

$$= \underbrace{(-\alpha_k - \beta_k + 1)}\cdot x + \alpha_k \cdot c$$

$= 0$: plug $z = 1 \to p_{k+1}(z) = \alpha_k z p_k(z) + \beta_k p_{k-1}(z)$

$1 = \alpha_k + \beta_k$

$$= \alpha_k \cdot c$$

Final algorithm!

$$y_{k+1} = \alpha_k R y_k + \beta_k y_{k-1} + \alpha_k \cdot c$$

cost ~ same as original splitting method

All this assumed R has real eval bounded by $\rho < 1$

Try on Jacobi for Model problem:

R_J was symmetric

$$\rho(R_J) = \cos\left(\frac{\pi}{N+1}\right) \approx 1 - \frac{\pi^2}{2(N+1)^2}$$
$$= 1 - O\left(\frac{1}{N^2}\right)$$

$\Rightarrow$ Jacobi take $O(N^2)$ iterations to converge, vs. $O(N)$ for Chebyshev, big improvement

But we know $SOR(\omega_{opt})$ does $O(N)$ iterations

$\Rightarrow$ Try to apply Chebyshev to $SOR(\omega_{opt})$

However, $R_{SOR(\omega_{opt})}$ has complex evals

Clever fix: Run $SOR(\omega)$ twice, in forwards (Red then Black) direction and backwards (Black then Red) directions

$\Rightarrow$ makes $R_{SSOR(\omega)}$ symmetric

SSOR(ω) = symmetric SOR(ω)

How to pick ω? Still have good estimate $\rho(R_{SSOR(\omega_{OPT})}) = 1 - O(\frac{1}{N})$

$\Rightarrow$ can Apply Chebyshev to accelerate SSOR(ω_{OPT}) $\Rightarrow$ only need $O(\sqrt{N})$ iterations to converge instead of $O(N)$

For 2D Poisson on $N \times N$ mesh, $n = N^2$

need $O(N^2 \sqrt{N})$ flops $= O(n^{5/4})$ flops

versus $O(n^{3/2})$ for SOR(ω_{OPT})

For 3D Poisson on $N \times N \times N$ mesh

$n = N^3$, need $O(N^3 \sqrt{N}) = O(n^{7/6})$ flops, closer to lower bound of $O(n)$

Finished explaining table of all algorithms for Poisson in Lecture 21