

Lecture 9: October 28, 2015

Lecturer: Madhur Tulsiani

1 The conjugate gradient method

In the last lecture we saw the steepest descent or gradient descent method for finding a solution to the linear system $Ax = b$ for $A \succ 0$. The method guarantees $\|x_t - x^*\| \leq \varepsilon \cdot \|x_0 - x^*\|$ after $t = O(\kappa \log(1/\varepsilon))$ iterations, where κ is the condition number of the matrix A . We will see that the conjugate gradient can obtain a similar guarantee in $O(\sqrt{\kappa} \log(1/\varepsilon))$ iterations.

For the steepest descent method, if we start from $x_0 = 0$, we get

$$x_t - x^* = (I - \eta A)(-x^*),$$

which gives $x_t = p(A)b$ for some polynomial p of degree at most t . The conjugate gradient method just takes this idea of finding an x of the form $p(A)b$ and runs with it. The method finds an $x_t = p_t(A)b$ where p_t is the *best* polynomial of degree at most t i.e., the polynomial which minimizes the function $\frac{1}{2} \langle Ax, x \rangle - \langle b, x \rangle + c$. However, the method does not explicitly work with polynomials. Instead we use the simple observation that any vector of the form $p_t(A)b$ lies in the subspace $\text{Span}(\{b, Ab, \dots, A^t b\})$ and the method finds the best vector in the subspace at every time t .

Definition 1.1 Let $\varphi : V \rightarrow V$ be a linear operator on a vector space V and let $v \in V$ be a vector. The Krylov subspace of order t defined by φ and v is defined as

$$\mathcal{K}_t(\varphi, v) := \text{Span}(\{v, \varphi(v), \dots, \varphi^t(v)\}).$$

Thus, at step t of the conjugate gradient method, we find the best vector in the space $\mathcal{K}_t(A, b)$ (we will just write the subspace as \mathcal{K}_t since A and b are fixed for the entire argument). The trick of course is to be able to do this in an iterative fashion so that we can quickly update the minimizer in the space \mathcal{K}_{t-1} to the minimizer in the space \mathcal{K}_t . This can be done by expressing the minimizer in \mathcal{K}_{t-1} in terms of a convenient orthonormal basis $\{u_0, \dots, u_{t-1}\}$ for \mathcal{K}_{t-1} . It turns out that if we work with a basis which is orthonormal with respect to the inner product $\langle \cdot, \cdot \rangle_A$, at step t we only need to update the component of the minimizer along the new vector u_t we get to obtain a basis for \mathcal{K}_t .

1.1 The algorithm

Recall that we defined the inner product $\langle x, y \rangle_A := \langle Ax, y \rangle$ where $\langle \cdot, \cdot \rangle$ denotes the standard inner product on \mathbb{R}^n . We saw that the Gram-Schmidt process can be used to compute an orthonormal basis for any inner product space, and thus it can also be used to find an orthonormal basis for \mathcal{K}_t under the inner product $\langle \cdot, \cdot \rangle_A$. It turns out that the usual complexity of Gram-Schmidt is still

too large for this application and in the homework you will see a way of speeding it up for this application, but we ignore this issue for now.

Let $\langle u_0, \dots, u_{t-1} \rangle$ be an orthonormal basis for \mathcal{K}_t under the inner product $\langle \cdot, \cdot \rangle_A$. Let $x_{t-1} = \sum_{i=0}^{t-1} \alpha_i \cdot u_i$. Then,

$$\begin{aligned} f(x_{t-1}) &= \frac{1}{2} \cdot \langle Ax_{t-1}, x_{t-1} \rangle - \langle b, x_{t-1} \rangle + c \\ &= \frac{1}{2} \cdot \langle x_{t-1}, x_{t-1} \rangle_A - \langle b, x_{t-1} \rangle + c \\ &= \frac{1}{2} \cdot \left(\sum_{i=0}^{t-1} \alpha_i^2 \right) - \sum_{i=0}^{t-1} \alpha_i \cdot \langle b, u_i \rangle + c \\ &= \frac{1}{2} \cdot \sum_{i=0}^{t-1} (\alpha_i - \langle b, u_i \rangle)^2 + \frac{1}{2} \sum_{i=0}^{t-1} \langle b, u_i \rangle^2 + c. \end{aligned}$$

Check that the second term does not depend on the choice of the orthonormal basis.

Exercise 1.2 Let $\{u_0, \dots, u_{t-1}\}$ and $\{w_0, \dots, w_{t-1}\}$ be two orthonormal bases for the space $\mathcal{K}_{t-1}(A, b) = \text{Span}(\{b, Ab, \dots, A^{t-1}b\})$. Then

$$\sum_{i=0}^{t-1} \langle b, u_i \rangle^2 = \sum_{i=0}^{t-1} \langle b, w_i \rangle^2.$$

Thus, the minimizer can always be found by choosing $\alpha_i = \langle b, u_i \rangle$ for each i . If our algorithm has a basis $\{u_0, \dots, u_{t-1}\}$ for \mathcal{K}_{t-1} , the minimizer of $f(x)$ in \mathcal{K}_{t-1} must have $\alpha_i = \langle b, u_i \rangle$ for each i . At the next step, we add a new linearly independent vector $A^t b$ to the space \mathcal{K}_{t-1} to obtain \mathcal{K}_t (if this is not linearly independent then $\mathcal{K}_{t-1} = \mathcal{K}_t$). We can use this vector to update the basis to $\{u_0, \dots, u_t\}$ for u_t obtained from Gram-Schmidt. If the new minimizer in \mathcal{K}_t is $\sum_{i=0}^t \beta_i \cdot u_i$, we must still have $\beta_i = \langle b, u_i \rangle \alpha_i$ for all $i \leq t-1$. Thus, we only need to update the minimizer by adding $\langle b, u_t \rangle \cdot u_t$. The update step can then be described as

- Let $x_t = \sum_{i=0}^{t-1} \langle b, u_i \rangle \cdot u_i$ for a basis $\{u_0, \dots, u_{t-1}\}$ orthonormal under the inner product $\langle \cdot, \cdot \rangle_A$.
- Extend $\{u_0, \dots, u_{t-1}\}$ to a basis of \mathcal{K}_t by defining

$$v_t = A^t b - \sum_{i=0}^{t-1} \langle A^t b, u_i \rangle_A \cdot u_i \quad \text{and} \quad u_t = \frac{v_t}{\sqrt{\langle v_t, v_t \rangle_A}}.$$

- Update $x_{t+1} = x_t + \langle b, u_t \rangle \cdot u_t$.

Notice that the basis extension step here seems to require $O(t)$ matrix-vector multiplications in the t^{th} iteration and thus we will need $O(t^2)$ matrix-vector multiplications in total for t iterations. This would negate the quadratic advantage we are trying to gain over steepest descent. However, in the homework you will see a way of extending the basis using only $O(1)$ matrix-vector multiplications in each step.

1.2 Bounding the number of iterations

Since x_t lies in the subspace \mathcal{K}_{t-1} , we have $x_t = p(A)b$ for some polynomial p of degree at most $t-1$. Thus,

$$x_t - x^* = p(A)b - x^* = p(A)Ax^* - x^* = (I - p(A)A)(x_0 - x^*),$$

since $x_0 = 0$. We can think of $I - p(A)A$ as a polynomial $q(A)$, where $\deg(q) \leq t$ and $q(0) = 1$. Recall from last lecture that the minimizer of $f(x)$ is the same as the minimizer of $\langle x - x^*, x - x^* \rangle_A = \|x - x^*\|_A^2$. Since $p(A)b$ is the minimizer of $f(x)$ in \mathcal{K}_{t-1} , we have

$$\|x_t - x^*\|_A^2 = \min_{q \in Q_t} \|q(A)(x_0 - x^*)\|_A^2,$$

where Q_t is the set of polynomials defined as

$$Q_t := \{q \in \mathbb{R}[z] \mid \deg(q) \leq t, q(0) = 1\}.$$

Use the fact that if λ is an eigenvalue of a matrix M , then λ^t is an eigenvalue of M^t (with the same eigenvector) to prove that the following.

Exercise 1.3 Let $\lambda_1, \dots, \lambda_n$ be the eigenvalues of A . Then for any polynomial q and any $v \in \mathbb{R}^n$,

$$\|q(A)v\|_A \leq \left(\max_i |q(\lambda_i)| \right) \cdot \|v\|_A.$$

Using the above, we get that

$$\|x_t - x^*\|_A \leq \left(\min_{q \in Q_t} \max_i |q(\lambda_i)| \right) \cdot \|x_0 - x^*\|_A.$$

Thus, the problem of bounding the norm of $x_t - x^*$ is reduced to finding a polynomial q of degree at most t such that $q(0) = 1$ and $q(\lambda_i)$ is small for all i .

Exercise 1.4 Verify that using $q(z) = \left(1 - \frac{2z}{\lambda_1 + \lambda_n}\right)^t$ recovers the guarantee of the steepest descent method.

Note that the conjugate gradient method itself does not need to know anything about the optimal polynomials in the above bound. The polynomials are only used in the analysis of the bound. The following claim, which can be proved by using slightly modified Chebyshev polynomials, suffices to obtain the desired bound on the number of iterations.

Claim 1.5 For each $t \in \mathbb{N}$, there exists a polynomial $q_t \in Q_t$ such that

$$|q_t(z)| \leq 2 \cdot \left(1 - \frac{2}{\sqrt{\kappa} + 1}\right)^t \quad \forall z \in [\lambda_1, \lambda_n].$$

We will prove the claim later using Chebyshev polynomials. However, using the claim we have that

$$\|x_t - x^*\|_A \leq \left(\min_{q \in Q_t} \max_i |q(\lambda_i)| \right) \cdot \|x_0 - x^*\|_A \leq 2 \cdot \left(1 - \frac{2}{\sqrt{\kappa} + 1}\right)^t \cdot \|x_0 - x^*\|_A.$$

Thus, $O(\sqrt{\kappa} \log(1/\varepsilon))$ iterations suffice to ensure that $\|x_t - x^*\|_A \leq \varepsilon \cdot \|x_0 - x^*\|_A$.

1.3 Chebyshev polynomials

The Chebyshev polynomial of degree t is given by the expression

$$P_t(z) = \frac{1}{2} \cdot \left[\left(z + \sqrt{z^2 - 1} \right)^t + \left(z - \sqrt{z^2 - 1} \right)^t \right].$$

Note that this is a polynomial since the odd powers of $\sqrt{z^2 - 1}$ will cancel from the two expansions. For $z \in [-1, 1]$ this can also be written as

$$P_t(z) = \cos(t \cos^{-1}(z)),$$

which shows that $P_t(z) \in [-1, 1]$ for all $z \in [-1, 1]$.

Using these polynomials, we can define the required polynomials q_t as

$$q_t(z) = \frac{P_t\left(\frac{\lambda_1 + \lambda_n - 2z}{\lambda_n - \lambda_1}\right)}{P_t\left(\frac{\lambda_1 + \lambda_n}{\lambda_n - \lambda_1}\right)}.$$

The denominator is a constant which does not depend on z and the numerator is a polynomial of degree t in z . Hence $\deg(q_t) = t$. Also, the denominator ensures that $q_t(0) = 1$. Finally, for $z \in [\lambda_1, \lambda_n]$, we have $\left| \frac{\lambda_1 + \lambda_n - 2z}{\lambda_n - \lambda_1} \right| \leq 1$. Hence, the numerator is in the range $[-1, 1]$ for all $z \in [\lambda_1, \lambda_n]$. This gives

$$|q_t(z)| \leq \frac{1}{P_t\left(\frac{\lambda_1 + \lambda_n}{\lambda_n - \lambda_1}\right)} \leq 2 \cdot \left(\frac{\sqrt{\kappa} - 1}{\sqrt{\kappa} + 1} \right)^t = 2 \cdot \left(1 - \frac{2}{\sqrt{\kappa} + 1} \right)^t.$$

The last bound above can be computed directly from the first definition of the Chebyshev polynomials.