Week 14: Monday, Nov 23

Logistics

1. Please do use the `eig` command for HW 5!

2. Note: if $M = LL^T$ is symmetric and positive definite, then singularity of $A - \lambda M$ is equivalent to singularity of $L^{-1}(A - \lambda M)L^{-T} = L^{-1}AL^{-T} - \lambda I$.

3. HW 4 is graded.

Some minimization problems

Last time, we sketched the following two-step strategy for approximating the solution to linear systems via Krylov subspaces:

1. Build a sequence of Krylov subspaces, either the obvious $\mathcal{K}_m(A, b)$ or a “preconditioned” space $\mathcal{K}_m(M^{-1}A, M^{-1}b)$, possibly derived from looking at iterates of a promising stationary method.

2. Choose a criterion to extract an approximate solution from each space.

Let us now turn to the problem of choosing a good solution from a subspace $\mathcal{V}_m$ in the case when $A$ is symmetric and positive definite. The following four equivalent criteria seem rather natural, and are the basis for the conjugate gradient method:

1. Define a function
   \[ \phi(z) = \frac{1}{2} z^T A z - z^T b. \]
   
   This function is convex (the Hessian $A$ is positive definite), and has a unique global minimum at $x = A^{-1}b$:
   \[ \nabla \phi(x) = Ax - b = 0. \]

   One way to choose the solution $x_m \in \mathcal{V}_m$ is therefore to minimize $\phi$ over the subspace $\mathcal{V}_m$. 
2. If \( x_m \) minimized \( \phi \) over the subspace \( \mathcal{V}_m \), that means that the directional derivative \( v^T \nabla \phi(x_m) = v^T(Ax_m - b) \) must be zero for all \( v \in \mathcal{V}_m \). Put differently, the residual \( Ax_m - b \) must be orthogonal to \( \mathcal{V}_m \). This is a Galerkin condition.

3. Given \( x_m \in \mathcal{V}_m \), write the error \( e_m = x_m - x \). Note that \( A \) defines an inner product and an associated norm \( \| z \|_A^2 = z^T A z \). Measuring \( e_m \) in this norm gives
\[
\| e_m \|_A^2 = (x_m - x)^T A (x_m - x) = x_m^T A x_m - 2x_m^T b + x^T A x = 2\phi(x_m) + \| x \|_A^2.
\]
Therefore, minimizing \( \| e_m \|_A^2 \) is equivalent to minimizing \( \phi(x_m) \).

4. Just as \( A \) denotes an inner product, so does \( A^{-1} \); and if we write \( r_m = Ax_m - b = Ae_m \), we have
\[
\| e_m \|_A^2 = e_m^T A e_m = (A e_m)^T A^{-1} (A e_m) = r_m^T A^{-1} r_m = \| r_m \|_{A^{-1}}^2.
\]
Therefore, minimizing \( \| r_m \|_A^2 \) is equivalent to minimizing \( \phi(x_m) \).

The Lanczos connection

Suppose now that the columns of \( V_m \) form an orthonormal basis for the Krylov subspace \( \mathcal{K}_m(A, b) \) produced via the Lanczos process. If we choose \( x_m = V_m y_m \) so that the residual is orthogonal to every vector in \( \mathcal{K}_m(A, b) \) (point 2 above), then we have
\[
V_m^T (Ax_m - b) = V_m^T A V_m y_m - V_m^T b = T_m y_m - \beta e_1 = 0,
\]
where \( T_m \) is the tridiagonal matrix produced by the Lanczos process, and \( V_m^T b = \beta e_1 \) because the first column of \( V_m \) is proportional to \( b \). We can solve for \( y_m \) by writing an unpivoted LU factorization of \( T_m \), which gives us
\[
x_m = V_m U_m^{-1} L_m^{-1} y_m,
\]
and then note that if \( P_m = V_m U_m^{-1} \) and \( z_m = L_m^{-1} y_m \), we can extend to \( P_{m+1} \) by appending a column to \( P_m \) via a simple recurrence, and we can extend to \( z_{m+1} \) by appending an entry to \( z_m \). This is one way to derive the conjugate gradient iteration; see the Lanczos chapter in Golub and Van Loan or the derivation in Demmel.
Another approach to CG

An alternate approach to the conjugate gradient method does not directly invoke Lanczos, but instead relies on properties that must be satisfied at each step by the residual \( r_m = b - Ax_m \) and the update \( d_m = x_{m+1} - x_m \). We assume throughout that \( x_m \) is drawn from \( \mathcal{K}_m(A, b) \), which implies that \( r_m \in \mathcal{K}_{m+1}(A, b) \) and \( d_m \in \mathcal{K}_{m+1}(A, b) \).

First, note that \( r_m \perp \mathcal{K}_m(A, b) \) and \( d_m \perp_A \mathcal{K}_m(A, b) \). The former statement comes from the Galerkin criterion in the previous section. The latter statement comes from recognizing that \( r_{m+1} = Ad_m + r_m \perp \mathcal{K}_m(A, b) \); with Galerkin condition \( r_m \perp \mathcal{K}_m(A, b) \), this means \( Ad_m \perp \mathcal{K}_m(A, b) \). Together, these statements give us \( r_m \) and \( d_m \) to within a scalar factor, since there is only one direction in \( \mathcal{K}_{m+1}(A, b) \) that is orthogonal to all of \( \mathcal{K}_m(A, b) \), and similarly there is only one direction that is \( A \)-orthogonal. This suggests the following idea to generate the sequence of approximate solutions \( x_k \):

1. Find a direction \( p_{k-1} \in \mathcal{K}_k(A, b) \) that is \( A \)-orthogonal to \( \mathcal{K}_{k-1}(A, b) \).
2. Compute \( x_k = x_{k-1} + \alpha_k p_{k-1} \) so that
   \[
   r_k = r_{k-1} - \alpha_k A p_{k-1} \perp r_{k-1},
   \]
   i.e. set \( \alpha_k = \frac{r_{k-1}^T r_{k-1}}{p_{k-1}^T A p_{k-1}} \). Orthogonality to the rest of \( \mathcal{K}_k(A, b) \) follows automatically from the construction.
3. Take \( r_k \in \mathcal{K}_{k+1}(A, b) \) and \( A \)-orthogonalize against everything in \( \mathcal{K}_k(A, b) \) to generate the new direction \( p_k \). As with the Lanczos procedure, the real magic in this idea is that we have to do very little work to generate \( p_k \) from \( r_k \). Note that for any \( j < k - 1 \), we have \( p_j^T A r_k = (A p_j)^T r_k = 0 \), because \( A p_j \in \mathcal{K}_j+2(A, b) \subset \mathcal{K}_k(A, b) \) is automatically orthogonal to \( r_k \). Therefore, we really only need to choose
\[
\beta_k = \frac{r_k^T A p_k}{p_{k-1}^T A p_{k-1}} = \frac{(r_k^T r_k)/\alpha_k}{r_{k-1}^T r_{k-1}/\alpha_k} = \frac{r_k^T r_k}{r_{k-1}^T r_{k-1}}.
\]

\(^1\) \( u \perp_A v \) means \( u \) and \( v \) are orthogonal in the \( A \)-induced inner product, i.e. \( u^T A v = 0 \).
Putting everything together, we have the following coupled recurrences for the solutions \( x_k \), residuals \( r_k \), and search directions \( p_k \):

\[
\begin{align*}
\alpha_k &= \frac{(r_{k-1}^T r_k)}{(p_{k-1}^T A p_{k-1})} \\
x_k &= x_{k-1} + \alpha_k p_{k-1} \\
r_k &= r_{k-1} - \alpha_k A p_{k-1} \\
\beta_k &= \frac{(r_k^T r_k)}{(r_{k-1}^T r_{k-1})} \\
p_k &= r_k + \beta_k p_{k-1}.
\end{align*}
\]

The sequences \( r_k \) and \( p_k \) respectively form orthogonal and \( A \)-orthogonal bases for the nested Krylov subspaces generated by \( A \) and \( b \).

### The many approaches to CG

The description I have given in these notes highlights (I hope) how orthogonality of the residuals and \( A \)-orthogonality of search directions follows naturally from the Galerkin condition, and how the rest of the CG iteration can be teased out of these orthogonality relations. However, this is far from the only way to “derive” the method of conjugate gradients. The discussion given by Demmel and by Saad (in *Iterative Methods for Sparse Linear Systems*) highlights the Lanczos connection, and uses this connection to show the existence of \( A \)-orthogonal search directions. Golub and Van Loan show the Lanczos connection, but also show how conjugate gradients can be derived as a general-purpose minimization scheme applied to the quadratic function \( \phi(x) \). Trefethen and Bau give the iteration without derivation first, and then gradually explain some of its properties. If you find these discussions confusing, or simply wish to read something amusing, I recommend Shewchuk’s “Introduction to the Conjugate Gradient Method Without the Agonizing Pain.”