We now consider preconditioning. First, we consider a motivational example (involving $A_3$ discussed earlier) illustrating the need for preconditioning.

**Example 1.** Consider the tridiagonal matrix $n \times n$ matrix $A_3$ defined by

$$(A_3)_{ij} = \begin{cases} 2 & \text{if } i = j, \\ -1 & \text{if } |i - j| = 1, \\ 0 & \text{otherwise.} \end{cases}$$

The eigenvalues and eigenvectors of $A_3$ are

$$\lambda_i = 2 - 2 \cos \left( \frac{i\pi}{n+1} \right).$$

and

$$\phi_i = \left( \sin \left( \frac{i\pi}{n+1} \right), \sin \left( \frac{2i\pi}{n+1} \right), \ldots, \sin \left( \frac{ni\pi}{n+1} \right) \right)^t,$$

for $i = 1, \ldots, n$.

Suppose we use $\tau = 1/\lambda_n$ in Richardson’s method,

$$x_{i+1} = x_i + \tau(b - Ax_i).$$

Then $\rho(G) = 1 - 1/K$ where $K = \lambda_n/\lambda_1$ and $G$ is the corresponding reduction matrix. We shall call $K$ the spectral condition number of $A_3$. By the results of the previous class, we know that we get a reduction $\rho(G)$ per iteration (in either the $\ell^2$ norm or the $A$-norm). Suppose that we would like to iterate until we reduce the $A$–norm by a factor of $\epsilon$ ($\epsilon$ could be $10^{-6}$, for example).

The number of iterations $m$ required to do this thus satisfies

$$(1 - 1/K)^m \leq \epsilon \quad \text{or} \quad m \ln(1 - 1/K) \leq \ln(\epsilon).$$

Now, as $\ln(1 - 1/K) < 0$,

$$m \geq \frac{\ln(\epsilon)}{\ln(1 - 1/K)} \approx K \ln(1/\epsilon)$$

where we replaced $\ln(1 - 1/K)$ with the Taylor approximation near 1, i.e.

$$\ln(1 - 1/K) \approx -1/K.$$

Thus, we see that for a fixed reduction $\epsilon$, the number of iterations grows proportional to the condition number $K$. For large $n$, from (8.1) we see that $\lambda_n \approx 4$ and using a Taylor expansion of $\cos(\theta)$ around $\theta = 0$ gives

$$\lambda_1 \approx \frac{\pi^2}{(n+1)^2} \quad \text{and} \quad K \approx \frac{4(n+1)^2}{\pi^2}.$$
This means that the condition number $K$ (and hence the number of iterations) grows proportional to $n^2$. This is way too many iterations in practice. Even choosing the optimal parameter $\tau = 2/(\lambda_1 + \lambda_n)$ does not help much. In this case, the reduction rate is

$$\frac{1 - 1/K}{1 + 1/K} \approx 1 - 2/K$$

The above analysis suggests that we need half as many iterations in this case when compared to the suboptimal choice $\tau = 1/\lambda_n$. Thus, the number of iterations will still grow proportional to $n^2$.

**Remark 1.** Let $A$ be a symmetric and positive definite real $n \times n$ matrix and $\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n$ be its eigenvalues. Taking $B$ to be the identity in the corollary of the previous lecture gives

$$\lambda_n = \|A\|_2.$$  

Clearly, the eigenvalues of $A^{-1}$ are $\lambda_n^{-1} \leq \cdots \leq \lambda_2^{-1} \leq \lambda_1^{-1}$ so $\|A^{-1}\|_2 = \lambda_1^{-1}$. Thus, the spectral condition number $K$ is given by

$$K = \lambda_n/\lambda_1 = \|A\|_2 \|A^{-1}\|_2.$$

The norms appearing above are the operator norms induced by the $\| \cdot \|_2$ norm on $\mathbb{R}^n$. In general, the condition number of a matrix $A$ with respect to a norm $\| \cdot \|$ on $\mathbb{R}^n$ is defined by

$$\text{Cond}(A) = \|A\| \|A^{-1}\|$$

and depends on the choice of norm.

In general, the condition number provides an indication of the behavior of iterative methods. To achieve a fixed reduction, the direct application of a simple iterative method to a matrix with a large condition number requires a number of steps proportional to the condition number. The idea of “preconditioning” is to transform the problem with large condition number to one which has a significantly smaller one.

**Preconditioning:** Suppose that we want to iteratively solve the system

(8.2) \[ Ax = b \]

involving an $n \times n$ nonsingular matrix $A$ which is poorly conditioned (i.e., has large condition number). A preconditioner $B$ is another nonsingular $n \times n$ matrix. Multiplying (8.2) by $B$ does not change the solution (why?) and we consider the iterative solution of

(8.3) \[ BAx = Bb. \]

A good preconditioner will satisfy the following properties:
(1) The application of $B$ to a vector $v \in \mathbb{R}^n$ should be relatively cheap, i.e., it should not take much more computer effort than the cost of applying $A$ to a vector. So, for example, if $A$ is sparse and has $O(n)$ non-zeroes, then the application of an ideal $B$ (to a vector in $\mathbb{R}^n$) should only involve $O(n)$ operations.

(2) The condition number of $BA$ should be significantly smaller than that of $A$.

We first consider two trivial examples for $B$. Clearly, $B = I$ does nothing at all. Alternatively, we could consider $B = A^{-1}$ in which case the system becomes $x = A^{-1}b$ and we have solved the problem. Of course, the whole point of iterative methods is to get an accurate approximation to the solution $x$ in much less time (computational effort) than it would have taken to solve the problem by direct methods. Neither of the extremes, $B = I$ or $B = A^{-1}$ are useful choices for preconditioned iterative methods.

The construction and analysis of preconditioners has been the subject of intensive research in the last forty years. As we proceed with this course, we shall examine some basic ideas for the construction of preconditioners. As for now, we shall assume that $B$ has been given and see what properties are required for effective preconditioning.

We consider the case when $A$ and $B$ are positive definite (real) matrices. The Richardson method applied to (8.3) becomes

$$x_{i+1} = x_i + \tau B(b - Ax_i)$$

and the error $e_i = x - x_i$ satisfies

$$e_{i+1} = Ge_i$$

where $G = I - \tau BA$. In general, $BA$ is no longer symmetric but, as we have already observed, $BA$ is self adjoint with respect to either the $A$-inner product or the $B^{-1}$-inner product. This means that $G$ is also self adjoint with respect to either inner product. The corollary of the last class implies

$$\|G\|_{B^{-1}} = \|G\|_A = \rho(G) = \max_{i=1}^n |1 - \tau \lambda_i|.$$ 

Here $\lambda_1 \leq \lambda_2 \leq \cdots \lambda_n$ are the (real) eigenvalues of $BA$. Now if $\phi_i$ is the eigenvector (in the $A$-orthonormal basis for $\mathbb{R}^n$ from the general spectral theorem) corresponding to $\lambda_i$, we have

$$\lambda_i = \lambda_i(\phi_i, \phi_i)_A = (BA\phi_i, \phi_i)_A = (ABA\phi_i, \phi_i) = (BA\phi_i, A\phi_i) > 0$$

since $B$ is positive definite and $A\phi_i \neq 0$. The analysis of the previous class applies here as well and we find that we can get a fix reduction in a number of iterations proportional to $K = \lambda_n/\lambda_1$. The difference is that the eigenvalues appearing are now those for $BA$ and not for $A$. The following proposition
is a useful characterization of the eigenvalues of $BA$ when $B$ and $A$ are symmetric and positive definite.

**Proposition 1.** Let $B$ and $A$ be symmetric and positive definite (real) $n \times n$ matrices and $\lambda_1$ and $\lambda_n$ be, respectively, the smallest and largest eigenvalues for $BA$. Then $\lambda_1$ is the maximum value for $c_0$ and $\lambda_n$ is the minimal value of $c_1$ satisfying any of the following inequalities:

\begin{align}
\text{(8.4)} & \quad c_0(Ax, x) \leq (ABAx, x) \leq c_1(Ax, x), \quad \text{for all } x \in \mathbb{R}^n, \\
\text{(8.5)} & \quad c_0(B^{-1}x, x) \leq (Ax, x) \leq c_1(B^{-1}x, x), \quad \text{for all } x \in \mathbb{R}^n, \\
\text{(8.6)} & \quad c_0(A^{-1}x, x) \leq (Bx, x) \leq c_1(A^{-1}x, x), \quad \text{for all } x \in \mathbb{R}^n.
\end{align}

Before proving the above proposition, we note that if $B$ is symmetric and positive definite and $A$ is self adjoint with respect to $B$ then the largest and smallest eigenvalues of $A$ are respectively the maximum and minimum of the Rayleigh quotient, i.e.,

\begin{align}
\text{(8.7)} & \quad \lambda_n = \max_{x \in \mathbb{R}^n, x \neq 0} \frac{(BAx, x)_A}{(x, x)_A} \quad \text{and} \quad \lambda_1 = \min_{x \in \mathbb{R}^n, x \neq 0} \frac{(BAx, x)_A}{(x, x)_A}.
\end{align}

(The quantity $\lambda(x) = \frac{(BAx, x)_A}{(x, x)_A}$ is often referred to as the Rayleigh quotient.)

**Exercise 1.** Prove (8.7). Hint: Use the general version of the spectral theorem given in the previous class.

**(Proof of the proposition).** As already observed, $BA$ is self adjoint and positive definite with respect to the $A$-inner product. By (8.7), $\lambda_n$ is the minimal number satisfying

\[ \frac{(BAx, x)_A}{(x, x)_A} \leq \lambda_n \quad \text{for all } x \in \mathbb{R}^n, \ x \neq 0. \]

This is the right hand inequality in (8.4), the left follows analogously. The two inequalities in (8.5) follow from similar arguments and the fact that $BA$ is self adjoint in the $B^{-1}$-inner product. Finally, (8.6) follows from substituting $x = A^{-1}y$ in (8.4). □

**Remark 2.** The above proposition enables us to get bounds on the spectral condition number corresponding to the preconditioning system by deriving inequalities appearing (8.4)-(8.6). Thus, if any one of the left hand inequalities hold with $c_0$ and any one of the right hand inequalities hold with $c_1$,
then the spectral condition number $K$ satisfies

$$K \equiv K(BA) \leq \frac{c_1}{c_0}.$$

**Example 2.** The matrix $A_5$ comes from a finite difference approximation to the two dimensional boundary value problem

\begin{equation}
-\Delta u(x) = f, \quad \text{for } x = (x_1, x_2) \text{ in } (0, 1)^2
\end{equation}

$$u(x) = 0, \quad \text{for } x_1 = 0 \text{ or } 1 \text{ or } x_2 = 0 \text{ or } 1.$$

Here $\Delta$ denotes the Laplacian and is defined by

$$\Delta u = \frac{\partial^2 u}{\partial x_1^2} + \frac{\partial^2 u}{\partial x_2^2}.$$

Let $h = 1/(n+1)$ (for $n$ a large integer) and $x_{i,j} = (ih, jh)$, $i, j = 0, \ldots, n+1$. Note that if $i$ or $j$ is 0 or $n+1$, then $x_{i,j}$ is on the boundary of the square $(0, 1)^2$. The finite difference approximation to (8.9) is a mesh vector $\{u^h_{i,j}\}$ satisfying

\begin{equation}
4u^h_{i,j} - u^h_{i-1,j} - u^h_{i+1,j} - u^h_{i,j+1} - u^h_{i,j-1} = h^2 f(x_{i,j}),
\end{equation}

for $i, j = 1, 2, \ldots, n$ with

\begin{equation}
u^h_{i,j} = 0 \text{ when } i \text{ or } j = 0 \text{ or } n+1.
\end{equation}

We have seen that finite difference problems can be written as linear systems when we choose an ordering of the unknowns. The above finite difference equations lead to a linear system with $n^2$ unknowns, corresponding to $i, j = 1, \ldots, n$. Homework 4 used CSR files for the five point operator on the square.

Using lexicographical ordering, the finite difference problem is converted to a matrix problem

$$A_5 U = F$$

on $\mathbb{R}^{n^2}$ as discussed in an earlier class. For example, we set $k(i, j) = i + (j - 1) * n$ so that $U_{k(i,j)} = u^h_{i,j}$ and $F_{k(i,j)} = h^2 f(x_{i,j})$. The matrix $A_5$ is given by

$$\begin{cases}
4 & \text{if } i_1 = i_2 \text{ and } j_1 = j_2 \\
-1 & \text{if } i_1 = i_2 \text{ and } j_1 = j_2 + 1 \text{ and } j_2 \neq n \\
-1 & \text{if } i_1 = i_2 \text{ and } j_1 = j_2 - 1 \text{ and } j_2 \neq 1 \\
-1 & \text{if } i_1 = i_2 + 1 \text{ and } j_1 = j_2 \text{ and } i_2 \neq n \\
-1 & \text{if } i_1 = i_2 - 1 \text{ and } j_1 = j_2 \text{ and } i_2 \neq 1 \\
0 & \text{otherwise.}
\end{cases}$$
As we vary $i_1, i_2, j_1, j_2$ in the set $\{1, 2, \ldots, n\}$, we get all possible pairs of indices, $i = k(i_1, j_1), j = k(i_2, j_2) \in \{1, \ldots, n^2\}$. It is not hard to see that $A_5$ is symmetric.

Examining the matrix $A_5$ and its relation to the finite difference equation we have that

$$ (A_5V, V) = \sum_{i,j=1}^{n} (4v_{i,j}^h - v_{i-1,j}^h - v_{i+1,j}^h - v_{i,j+1}^h - v_{i,j-1}^h)v_{i,j}^h $$

where $V(k(i, j)) = v_{i,j}^h$. Now

$$ \sum_{i=1}^{n} (2v_{i,j}^h - v_{i-1,j}^h - v_{i+1,j}^h)v_{i,j}^h = \sum_{i=1}^{n} (v_{i,j}^h - v_{i-1,j}^h)v_{i,j}^h - \sum_{l=2}^{n+1} (v_{i,j}^h - v_{l-1,j}^h)v_{l-1,j}^h $$

$$ = (v_{1,j}^h)^2 + \sum_{i=2}^{n} (v_{i,j}^h - v_{i-1,j}^h)^2 + (v_{n,j}^h)^2. $$

For the first equality above, we split each term on the left into two and changed the index $l = i + 1$ for the second. Using this and the analogous identity

$$ \sum_{j=1}^{n} (2v_{i,j}^h - v_{i,j-1}^h - v_{i,j+1}^h)v_{i,j}^h = (v_{i,1}^h)^2 + \sum_{j=2}^{n} (v_{i,j}^h - v_{i,j-1}^h)^2 + (v_{i,n}^h)^2 $$

gives

$$ (A_5V, V) = \sum_{j=1}^{n} \left( (v_{1,j}^h)^2 + \sum_{i=2}^{n} (v_{i,j}^h - v_{i-1,j}^h)^2 + (v_{n,j}^h)^2 \right) $$

$$ + \sum_{i=1}^{n} \left( (v_{i,1}^h)^2 + \sum_{j=2}^{n} (v_{i,j}^h - v_{i,j-1}^h)^2 + (v_{i,n}^h)^2 \right). $$

Note that it immediately follows from the above identity that $A_5$ is positive definite (why?).

We now consider solving a variable coefficient problem on the same domain, i.e.,

$$ -\frac{\partial}{\partial x_1}a_1(x)\frac{\partial u(x)}{\partial x_1} - \frac{\partial}{\partial x_2}a_2(x)\frac{\partial u(x)}{\partial x_2} = f, \text{ for } x = (x_1, x_2) \text{ in } (0,1)^2 $$

$$ u(x) = 0, \text{ for } x_1 = 0 \text{ or } 1 \text{ or } x_2 = 0 \text{ or } 1. $$
Its finite difference approximation is of the form
\[(8.12)\quad \begin{align*}
& a_1(x_{i+1/2,j})(u_{i,j}^h - u_{i+1,j}^h) + a_1(x_{i-1/2,j})(u_{i,j}^h - u_{i-1,j}^h) \\
& + a_2(x_{i,j+1/2})(u_{i,j}^h - u_{i,j+1}^h) + a_2(x_{i,j-1/2})(u_{i,j}^h - u_{i,j-1}^h)
\end{align*}
\]
for \(i, j = 1, 2, \ldots, n\) with \((8.10)\). It leads to the matrix \(\tilde{A}_5\) (using the same ordering)

\[(\tilde{A}_5)_{k(i_1,j_1),k(i_2,j_2)} = \begin{cases}
(a_1(x_{i_1+1/2,j_1}) + a_1(x_{i_1-1/2,j_1}) + a_2(x_{i_1,j_1+1/2}) + a_2(x_{i_1,j_1-1/2}) & \text{if } i_1 = i_2 \text{ and } j_1 = j_2 \\
-a_2(x_{i_1,j_1+1/2}) & \text{if } i_1 = i_2 \text{ and } j_1 = j_2 + 1 \text{ and } j_2 \neq n \\
-a_2(x_{i_1,j_1-1/2}) & \text{if } i_1 = i_2 \text{ and } j_1 = j_2 - 1 \text{ and } j_2 \neq 1 \\
-a_1(x_{i_1+1/2,j_1}) & \text{if } i_1 = i_2 + 1 \text{ and } j_1 = j_2 \text{ and } i_2 \neq n \\
-a_1(x_{i_1-1/2,j_1}) & \text{if } i_1 = i_2 - 1 \text{ and } j_1 = j_2 \text{ and } i_2 \neq 1 \\
0 & \text{otherwise.}
\end{cases}
\]

This matrix is also symmetric. Moreover, a computation similar to that leading to \((8.11)\) gives

\[(\tilde{A}_5V, V) = \sum_{j=1}^{n} \left(a_1(x_{1/2,j})(v_{1,j}^h)^2 + \sum_{i=2}^{n} a_1(x_{i-1/2,j})(v_{i,j}^h - v_{i-1,j}^h)^2 \right. \\
\left. + a_1(x_{n+1/2,j})(v_{n,j}^h)^2 \right) + \sum_{i=1}^{n} \left(a_2(x_{i,1/2})(v_{i,1}^h)^2 + \sum_{j=2}^{n} a_2(x_{i,j-1/2})(v_{i,j}^h - v_{i,j-1}^h)^2 + a_2(x_{i,n+1/2})(v_{i,n}^h)^2 \right). \tag{8.13} \]

Suppose that the coefficients \(a_1\) and \(a_2\) are positive, bounded from above and bounded away from zero below. Specifically, suppose that there are constants \(0 < \mu_0 \leq \mu_1\) satisfying

\[\mu_0 \leq a_1(x) \leq \mu_1 \text{ and } \mu_0 \leq a_2(x) \leq \mu_1 \text{ for all } x \in (0,1)^2.\]

It follows that \(\tilde{A}_5\) is also positive definite. It can be efficiently solved by preconditioned iteration, using \(B = A_5^{-1}\) as a preconditioner. Examining the identities \((8.11)\) and \((8.13)\), we find that

\[\mu_1^{-1}(\tilde{A}_5V, V) \leq (A_5V, V) \leq \mu_0^{-1}(\tilde{A}_5V, V) \text{ for all } V \in \mathbb{R}^{n^2}.\]

This can be rewritten (with \(B^{-1} = A_5\))

\[\mu_0(B^{-1}V, V) \leq (\tilde{A}_5V, V) \leq \mu_1(B^{-1}V, V) \text{ for all } V \in \mathbb{R}^{n^2}.\]
and so \( K(B\tilde{A}_5) \leq \mu_1/\mu_0 \) follows by applying the proposition. The preconditioned iteration converges very fast when \( \mu_1/\mu_0 \) is not large and the convergence rate is independent of \( h \) and the number of unknowns. This will be illustrated in the next programming exercise.