The first task of this class is to prove the convergence theorem for successive over relaxation (SOR) introduced last week. This proof is not very intuitive and was probably discovered by playing with the equations until something worked.

**Theorem 1.** Let $A$ be an $n \times n$ Hermitian positive definite matrix and $\omega$ be in $(0, 2)$. Then the SOR method for iteratively solving $Ax = b$ converges for any starting vector and right hand side.

**Proof.** The reduction matrix for SOR is $G_{\text{SOR}} = (\omega^{-1}D+L)^{-1}((\omega^{-1}-1)D-U)$ and we set $Q = (\omega^{-1}D + L)$. We shall show that $\rho(G_{\text{SOR}}) < 1$.

We note that

$$ (I - G_{\text{SOR}}) = (\omega^{-1}D+L)^{-1}((\omega^{-1}D+L) - ((\omega^{-1}-1)D-U)) = Q^{-1}A. $$

Let $x \in \mathbb{C}^n$ be an eigenvector of $G_{\text{SOR}}$ with eigenvalue $\lambda$ and set $y = (I - G_{\text{SOR}})x = (1 - \lambda)x$. Then, by (5.1),

$$ Qy = Ax $$

and

$$ (Q - A)y = (Q - A)Q^{-1}Ax = (A - AQ^{-1}A)x = A(I - Q^{-1}A)x = AG_{\text{SOR}}x = \lambda Ax. $$

Taking the inner product of (5.2) with $y$ in the second place and (5.3) with $y$ in the first place gives

$$ (Qy, y) = (Ax, y), \text{ and } (y, (Q - A)y) = (y, \lambda Ax). $$

We note that $d_{ii} = (Ae_i, e_i)$ where $e_i$ is the $i$th standard basis vector. It follows that the diagonal matrix $D$ has real positive entries and hence is Hermitian.

The above equations can thus be rewritten

$$ \omega^{-1}(Dy, y) + (Ly, y) = (1 - \bar{\lambda})(Ax, x), \text{ and } (\omega^{-1} - 1)(Dy, y) - (y, Uy) = (1 - \lambda)\bar{\lambda}(x, Ax). $$

Now since $A$ is Hermitian, $(Ly, y) = (y, Uy)$ so adding the two equations above gives

$$ (2\omega^{-1} - 1)(Dy, y) = (1 - |\lambda|^2)(Ax, x). $$
Now, $x \neq 0$ implies $(Ax, x) > 0$ so $Ax \neq 0$. As $Q$ is nonsingular, (5.2) implies $y \neq 0$. In addition, the assumption on $\omega$ implies that $2\omega^{-1} - 1 > 0$ so that the left hand side of (5.4) is positive (from above we know that $D$ is diagonal with positive numbers on the diagonal). For the right hand side of (5.4) to be positive, it is necessary that $|\lambda| < 1$, i.e., $\rho(G_{SOR}) < 1$. □

**Remark 1.** The above proof works for other splittings of $A$, e.g. $A = D + L + U$ where $D$ is Hermitian positive definite and $L^* = U$ ($D$, $L$ and $U$ need not be diagonal, strictly lower triangular, and strictly upper triangular, respectively).

**Remark 2.** Unlike the analysis for Gauss-Seidel and Jacobi given earlier, the above proof for SOR does not lead to any explicit bound for the spectral radius. All it shows is that the spectral radius has to be less than one.

One step of an iterative method for solving $Ax = b$ can be used to provide an “approximate” inverse. Specifically, we define $B : \mathbb{R}^n \rightarrow \mathbb{R}^n$ by defining $By$ to be the result of one iterative step applied to solving $Ax = y$ with initial iterate $x_0 = 0$. All of the iterative methods that we have considered up to now have been of the form

$$Qx_{i+1} = (Q - A)x_i + b.$$  

With this notation, the reduction matrix $G$ is given by $G = I - Q^{-1}A$ and the corresponding approximate inverse is $B = Q^{-1}$.

When $A$ is symmetric positive definite (SPD) real (or Hermitian positive definite) it is advantageous to have approximate inverses which are of the same type. We shall consider the case when $A$ is SPD real. The approximate inverse corresponding to the Jacobi method is $B = D^{-1}$ and is SPD when $A$ is SPD. The approximate inverse in the case of Gauss-Seidel is $B = (D + L)^{-1}$ and is generally not symmetric. We can develop a symmetric approximate inverse from Gauss-Seidel by introducing the “transpose” iteration,

$$(D + U)x_{i+1} = -Lx_i + b.$$  

Note that the implementation of this method as a sweep is similar to the original Gauss-Seidel method except that one goes through the vector in reverse order. The pseudo-code is as follows:

```c
FUNCTION gst(X, B, A, n)
FOR j = n, n − 1, . . . , 1 DO {
    X_j = (B_j - \sum_{k\neq j, A_{j,k} \neq 0} A_{j,k} X_k)A_{j,j}^{-1}.}
RETURN
END
```
We get a symmetric approximate inverse by defining $By$ by first applying one step of Gauss-Seidel with zero initial iterate and right hand side $y$ and using the result $x_1$ as an initial iterate for one step of the transpose iteration, i.e., $By = x_2$ where
\[(D + L)x_1 = y, \quad \text{and} \quad (D + U)x_2 = -Lx_1 + y.\]

We can explicitly compute $B = Q^{-1}$ by computing the reduction matrix $G = I - Q^{-1}A$ for the two step procedure and identifying $Q^{-1}$.

Let $x$ be the solution of $Ax = y$ and, as usual, set $e_i = x - x_i$ with $x_0 = 0$. The errors $e_1$ and $e_0$ are related by $e_1 = -(D + L)^{-1}Ue_0 = (I - (D + L)^{-1}A)e_0$ (the usual Gauss-Seidel reduction matrix) while the errors $e_2$ and $e_1$ are related by $e_2 = -(D + U)^{-1}Le_1 = (I - (D + U)^{-1}A)e_1$. Thus the error reduction matrix for the two steps is given by

\[G = (I - (D + U)^{-1}A)(I - (D + L)^{-1}A)\]
\[= I - [(D + U)^{-1} + (D + L)^{-1} - (D + U)^{-1}A(D + L)^{-1}]A\]
\[= I - (D + U)^{-1}[(D + L) + (D + U) - A](D + L)^{-1}A\]
\[= I - (D + U)^{-1}D(D + L)^{-1}A.\]

Thus, the approximate inverse associated with the two steps is given by
\[(5.6) \quad B = (D + U)^{-1}D(D + L)^{-1}\]
and is obviously SPD when $A$ is SPD (why?).

**Exercise 1.** The symmetric successive over relaxation method (SSOR) is defined in an analogous fashion, i.e., by taking one step of SOR with zero initial iterate followed by one step of the transpose iteration
\[
(\omega^{-1}D + U)x_{i+1} = ((\omega^{-1} - 1)D - L)x_i + b.
\]

Compute the approximate inverse associated with SSOR. Your result should reduce to (5.6) when $\omega = 1$.

**Optimization of SOR** The whole point of introducing a parameter into the SOR iteration is so that by judicious choice, one can get a significantly faster iteration. To illustrate that it is indeed possible to do this, we consider a special class of matrices, specifically, those satisfying the so-called “Property A” condition. Its definition will be given shortly but first we shall simplify the convergence study to the case when the diagonal of $A$ coincides with that of the identity.

Recall the SOR iteration,
\[
(\omega^{-1}D + L)x_{i+1} = ((\omega^{-1} - 1)D - U)x_i + b
\]
where $A = D + L + U$. Assume that $D_{ii} > 0$ and set $D^{1/2}$ to be the diagonal matrix with entries $D_{ii}^{1/2}$. We set

$$\hat{x}_i = D^{1/2}x_i$$

then

$$\left(\omega^{-1}I + \hat{L}\right)\hat{x}_{i+1} = \left(\left(\omega^{-1} - 1\right)I - \hat{U}\right)\hat{x}_i + \hat{b}$$

where

$$\hat{L} = D^{-1/2}LD^{-1/2}, \quad \hat{U} = D^{-1/2}UD^{-1/2}$$

$$\hat{b} = D^{-1/2}b, \quad \hat{A} = D^{-1/2}AD^{-1/2} = I + \hat{L} + \hat{U}.$$ 

Note that $\hat{L}$ and $\hat{U}$ are also lower and upper triangular. Now $\hat{x} = D^{1/2}x$ is the solution of $\hat{A}\hat{x} = \hat{b}$ and (5.7) is the corresponding SOR method. Set $e_i = x - x_i$ and $\hat{e}_i = \hat{x} - \hat{x}_i = D^{1/2}e_i$. Then $e_i$ converges to zero if and only if $\hat{e}_i$ converges to zero and their norms are related in the obvious way. In this way, we can reduce the study of SOR to the case when $\hat{A} = I + \hat{L} + \hat{U}$. This is an example of rescaling $A$ to obtain some desired property. Thus, at least in the case when $D_{ii} > 0$, it suffices to analyze the simpler case when $D = I$.

**Definition 1.** (Property A) Suppose that $A$ is an $n \times n$ matrix with 1’s on the diagonal. Writing $A = I + L + U$ with $L$ and $U$ strictly lower and upper triangular, we say that $A$ satisfies Property A if all of the eigenvalues of

$$J_z = \frac{1}{z}L + zU$$

are independent of $z$ ($z \neq 0$). Note that we allow $z$ to be complex.

**Example 1.** Any tridiagonal matrix satisfies Property A. Let $A$ be a tridiagonal matrix of dimension $n \times n$. To check Property A, we show that $J_z$, for any nonzero $z$, is similar to $J_1$. This suffices since similar matrices share the same spectrum. To do this, we introduce the diagonal similarity transformation matrix $M_z$ which has $(M_z)_{ii} = z^i$, $i = 1, \ldots, n$, as diagonal entries. A simple computation gives

$$(M_z^{-1}J_1M_z)_{ij} = z^{j-i}(J_1)_{ij}$$

from which it immediately follows that for tridiagonal $A$,

$$M_z^{-1}J_1M_z = J_z$$

i.e., $J_z$ is similar to $J_1$.

**Example 2.** In this example, we consider a finite difference matrix obtained from lexicographical ordering of the unknowns in a finite difference approximation to a boundary value problem associated with a partial differential equation. Let $\Omega$ be an open connected bounded subset of $\mathbb{R}^2$ (we refer to
as our domain). We want to approximate the function \( U \) defined on \( \Omega \) satisfying

\[
-\Delta U(x) = f(x) \quad \text{for all } x = (x_1, x_2) \in \Omega,
\]

\[
U(x) = 0 \quad \text{for } x \in \partial \Omega.
\]

Here \( \partial \Omega \) denotes the boundary of \( \Omega \) and

\[
\Delta u \equiv \frac{\partial^2 u}{\partial x_1^2} + \frac{\partial^2 u}{\partial x_2^2}
\]

is the Laplacian. We think of covering \( \mathbb{R}^2 \) with a uniform grid consisting of lines parallel to the \( x \)-axis (at \( y = jh, \ j \) an integer) and lines parallel to the \( y \)-axis (at \( x = ih, \ i \) an integer). Here \( h \) is a (small) positive number and determines the accuracy of the approximation. The nodes of the grid are the points \( x_{i,j} = (ih, jh) \) and are where the lines intersect. We seek a nodal function \( u_{i,j} \) where \( u_{i,j} \) approximates \( U(x_{i,j}) \). We get equations for \( \{u_{i,j}\} \) by replacing the derivatives on the left hand side of (5.8) by finite differences. Specifically,

\[
-\frac{\partial^2 u(x_{i,j})}{\partial x_1^2} \approx \frac{2u_{i,j} - u_{i-1,j} - u_{i+1,j}}{h^2}
\]

and

\[
-\frac{\partial^2 u(x_{i,j})}{\partial x_2^2} \approx \frac{2u_{i,j} - u_{i,j-1} - u_{i,j+1}}{h^2}
\]

which gives

\[
4u_{i,j} - u_{i-1,j} - u_{i+1,j} - u_{i,j-1} - u_{i,j+1} = h^2 f(x_{i,j}).
\]

We require that (5.9) holds for each \( x_{i,j} \in \Omega \) and we set \( u_{i,j} = 0 \) when \( x_{i,j} \) is outside of \( \Omega \) (from the boundary condition). The unknowns are the values of \( u_{i,j} \) for \( x_{i,j} \) in \( \Omega \). Thus, there are \( n \) unknowns where \( n \) is the number of grid nodes inside \( \Omega \). There are also \( n \) equations.

To get a matrix problem, we have to order the unknowns. The grid node ordering is not useful for this as there are two parameters in the grid ordering and we have only one parameter (index) for a vector. An ordering of the unknowns is a one to one (and onto) mapping \( k \) from the set of \((i, j)\) values corresponding to points \( x_{i,j} \in \Omega \) to the indices \( 1, 2, \ldots, n \). In this way, the \((i, j)\) grid unknown becomes the \( k(i,j) \)'th vector unknown. We order the equations the same way so that the equation (5.9) corresponding to \((i, j)\) becomes the \( k(i,j) \)'th equation (row of the matrix which we shall denote by \( A_5 \)).

We examine the structure of \( A_5 \) in more detail. It is clear that the \( k(i,j) \) row of the matrix has at most 5 non-zeros (note that, for example, when \( x_{i-1,j} \) is outside of \( \Omega \), \( u_{i-1,j} = 0 \) and so there is no corresponding entry in
the matrix). As \(u_{i,j}\) is the \(k(i,j)\)'th unknown and the equation corresponding to \((i,j)\) is the \(k(i,j)\)'th equation, the term \(4u_{i,j}\) produces at 4 on the \(k(i,j)\)'th diagonal. Similarly, the term \(-u_{i-1,j}\) produces the entry \(-1\) in \((A_5)_{k(i,j),k(i-1,j)}\) (when \(x_{i-1,j} \in \Omega\)). The remaining terms of (5.9) produce analogous entries in \(A_5\).

Lexicographical ordering involves defining \(k\) following the text reading direction, i.e., left to right on the first line followed by left to right on the second line, etc. For example a simple triangular grid of nodes would be ordered

\[
\begin{array}{ccccccc}
1 & 2 & 3 & 4 & 5 & 6 & \\
7 & 8 & 9 & 10 & & & \\
11 & 12 & 13 & 14 & 15 & & \\
16 & 17 & 18 & 19 & 20 & 21 & \\
\end{array}
\]

(5.10)

Similar to the previous example, we define a diagonal similarity matrix by

\[
(M_z)_{k,k} = z^{i-j} \quad \text{where} \quad k = k(i,j).
\]

There is no ambiguity with this definition as there is a unique index pair \((i,j)\) corresponding to each index in \(\{1, \ldots, n\}\) since \(k(\cdot, \cdot)\) is invertible.

We claim that \((M_z)^{-1}J_1M_z = J_z\). Indeed,

- If \(k_1\) corresponds to a grid point directly above \(k = k(i,j)\) then \(k_1 = k(i,j + 1)\) and
  \[
  ((M_z)^{-1}J_1M_z)_{k,k_1} = z^{-i+j}(J_1)_{k,k_1}z^{j-(j+1)} = z^{-1}L_{k,k_1}.
  \]
- If \(k_1\) corresponds to a grid point directly below \(k = k(i,j)\) then \(k_1 = k(i,j - 1)\) and
  \[
  ((M_z)^{-1}J_1M_z)_{k,k_1} = z^{-i+j}(J_1)_{k,k_1}z^{-j-(j-1)} = zU_{k,k_1}.
  \]
- If \(k_1\) corresponds to a grid point one to the right of \(k = k(i,j)\) then \(k_1 = k(i+1,j)\) and
  \[
  ((M_z)^{-1}J_1M_z)_{k,k_1} = z^{-i+j}(J_1)_{k,k_1}z^{i+1-j} = zU_{k,k_1}.
  \]
- If \(k_1\) corresponds to a grid point one to the left of \(k = k(i,j)\) then \(k_1 = k(i-1,j)\) and
  \[
  ((M_z)^{-1}J_1M_z)_{k,k_1} = z^{-i+j}(J_1)_{k,k_1}z^{i-1-j} = z^{-1}L_{k,k_1}.
  \]

As these are the only nonzero entries of \(L\) and \(U\), \((M_z)^{-1}J_1M_z = J_z\) and so the finite difference matrix satisfies Property A.