The Eigenvalue Problem

The Basic problem:
For $A \in \mathbb{R}^{n \times n}$ determine $\lambda \in \mathbb{C}$ and $x \in \mathbb{R}^n$, $x \neq 0$ such that:

$$Ax = \lambda x.$$ 

$\lambda$ is an eigenvalue and $x$ is an eigenvector of $A$.

- An eigenvalue and corresponding eigenvector, $(\lambda, x)$ is called an eigenpair.
- The spectrum of $A$ is the set of all eigenvalues of $A$.
- To make the definition of a eigenvector precise we will often normalize the vector so it has $\|x\|_2 = 1$. 

CSCD37H – Analysis of Numerical Algorithms – p.71/183
Alternative Definition

Note that the definition of eigenvalue is equivalent to finding \( \lambda \) and \( x \neq 0 \) such that,

\[
(A - \lambda I)x = 0.
\]

But the linear system \( Bx = 0 \) has a nontrivial solution iff \( B \) is singular. Therefore we have that \( \lambda \) is an eigenvalue of \( A \) iff \((A - \lambda I)\) is singular iff \(\det(A - \lambda I) = 0\).
Properties (From Lin. Alg.)

- For $A \in \mathbb{R}^{n \times n}$, $det(A - \lambda I)$ is a polynomial of degree $\leq n$ in $\lambda$, the characteristic polynomial.

- For triangular matrices, $L$ or $U$,

\[
det(L) = \prod_{i=1}^{n} l_{ii}, \quad det(U) = \prod_{i=1}^{n} u_{ii},
\]

and the eigenvalues are the diagonal entries of the matrix (since
\[
det(L - \lambda I) = \prod_{i=1}^{n} (l_{ii} - \lambda) \text{ has only the roots } l_{11}, l_{22} \cdots l_{nn}.\]

- For an upper triangular matrix with distinct eigenvalues, $U$, an eigenvector corresponding to the eigenvalue, $u_{ii}$, can be determined by solving the linear system,

\[
[U - u_{ii}I]y = 0,
\]
Eigenvectors of $U$

That is,

$$
\begin{bmatrix}
     u_{11} - u_{ii} & u_{12} & \cdots & u_{1n} \\
     0 & u_{22} - u_{ii} & \cdots & u_{2n} \\
     \vdots & \vdots & \ddots & \vdots \\
     0 & 0 & 0 & u_{nn} - u_{ii}
\end{bmatrix}
\begin{bmatrix}
     y_1 \\
     y_2 \\
     \vdots \\
     y_n
\end{bmatrix} =
\begin{bmatrix}
     0 \\
     0 \\
     \vdots \\
     0
\end{bmatrix}.
$$

This system can be solved using (modified back sub):

- set $y_n = y_{n-1} = \cdots y_{i+1} = 0$;
- set $y_i = 1$;
- for $j = (i-1), (i-2) \cdots 1$,
  $$
y_j = -\frac{\sum_{r=j+1}^{i} u_{jr} y_r}{(u_{jj} - u_{ii})};$$
- end
- normalize by setting $x = y/\|y\|_2$;
The General Case

Note that this algorithm must be modified for multiple eigenvalues (we will consider this case later). A similar procedure works for lower triangular matrices (exercise).

We have shown that the eigenvalue problem is easy, for triangular matrices, and the eigenvector problem is also easy, for triangular matrices, when the eigenvalues are distinct. We will now consider algorithms for the case of general matrices. The basic approach is to transform the general problem to an equivalent ‘easy’ problem (ie., an equivalent triangular eigenproblem).

Before we consider this approach we will consider a special technique that is particularly appropriate if only the largest (or smallest) magnitude eigenvalue is desired.
The Power Method

Assume $A \in \mathbb{R}^{n \times n}$ has eigenvalues $\lambda_1, \lambda_2 \cdots \lambda_n$, satisfying $|\lambda_1| \geq |\lambda_2| \cdots \geq |\lambda_n|$ and that $A$ has a complete set of normalized eigenvectors, $(v_1, v_2 \cdots v_n)$, (ie., $A$ is non-defective). These eigenvectors are linearly independent and any $x \in \mathbb{R}^n$ can be expressed as,

$$x = \sum_{j=1}^{n} \alpha_j v_j.$$ 

Therefore

$$Ax = \sum_{j=1}^{n} \alpha_j Av_j = \sum_{j=1}^{n} (\alpha_j \lambda_j) v_j$$

$$A^k x = \sum_{j=1}^{n} \alpha_j (\lambda_j)^k v_j$$

For any $x_0 \in \mathbb{R}^n$ we define the normalized sequence $x_j, j = 1, 2, \cdots$ by,

$$y_j = Ax_{j-1}, \quad x_j = \frac{y_j}{\|y_j\|}.$$
Power Method (cont.)

- When $|\lambda_1| > |\lambda_2|$, we can show,

\[ x_j \to v_1, \]

and the rate of convergence is $O(\rho^j)$ where $\rho = \frac{|\lambda_2|}{|\lambda_1|}$.

- Furthermore, since $\|x_j\| = 1$ and $y_j \to \lambda_1 x_j$, we have,

\[ \|y_j\| \to |\lambda_1|. \]

- We then have that $\lambda_1$ can be determined from the observation that $\lambda_1 \in \mathbb{R}$ (since $|\lambda_1| > |\lambda_2|$ and non-real eigenvalues must appear as conjugate pairs). This implies,

\[ \lambda_1 = \pm \lim_{j \to \infty} \|y_j\|, \]

where the correct sign can be determined by comparing the first non-zero components of $x_j$ and $y_j$. 
Power Method – Observations

The choice of norm used in the definition of \( x_j \) and \( y_j \) leads to different sequences but the term **Power Method** is used to refer to any method based on such a sequence. The text uses the \( l_\infty \) norm which is efficient but makes the discussion more difficult to follow. In many cases the \( l_2 \) norm is used for discussion but is slightly more expensive to implement since it requires more work to determine \( \| y_j \| \).

**Exercise:**
For the three norms, \( l_1, l_2 \) and \( l_\infty \) implement the power method in MATLAB and verify that for various choices of \( A \) and \( x_0 \) satisfying our assumptions, the resulting sequences are different but all three converge with the same rate of convergence.
Transformational Methods

Recall that, for Linear Equations, triangular systems $Rx = b$ are easy and the $LU$ and $QR$ algorithms are based on transforming a given general problem, $Ax = b$, onto an equivalent triangular system,

$$Ux = \tilde{b}.$$ 

A similar approach will be developed for the eigenproblem.

For the general eigenvalue problem, we are given an $n \times n$ matrix, $A$, and we introduce a sequence of transformations that transform the eigenproblem for $A$ onto equivalent eigenproblems for matrices $A_r$, where $A_r \rightarrow U \ (U \text{ upper triangular})$ as $r \rightarrow \infty$.

This is an Iterative method. We will focus on justifying and developing an iterative $QR$ method, where $(n - 1)$ Householder reflections are used to define the transformation on each iteration (defining $A_r$ from $A_{r-1}$).
Similarity Transformations

The Key Result from linear algebra that justifies this approach is the
Theorem that similarity transformations preserve eigenvalues and
allow us to recover eigenvectors.

That is, given any nonsingular matrix, \( M \), the eigenproblem,

\[
Ax = \lambda x,
\]

has a solution \((\lambda, x)\) iff the eigenproblem,

\[
MAM^{-1}y = \lambda y,
\]

has a solution \((\lambda, y)\) where \( y = Mx \).
Proof

Let \((\lambda, x)\) be a solution of \(Ax = \lambda x\) and \(B = MAM^{-1}, \ y = Mx,\)

\[
By = (MAM^{-1})(Mx),
= MAx,
= M\lambda x,
= \lambda y.
\]

To see the converse, let \((\lambda, y)\) be an eigenpair for \(B = MAM^{-1}\), with \(x\) the solution to \(Mx = y\). With \(w = Ax = AM^{-1}y,\)

\[
Mw = MAx,
= MAM^{-1}y,
= \lambda y,
= \lambda Mx,
\]

or, after multiplying both sides by \(M^{-1},\)

\[Ax = \lambda x,\]
The ‘trick’ then is to choose the sequence of nonsingular matrices, $M_1, M_2 \cdots M_r$ such that,

\[
\begin{align*}
A_0 &= A, \\
A_1 &= M_1 A_0 M_1^{-1}, \\
\vdots & \quad \vdots \\
A_r &= M_r A_{r-1} M_r^{-1}, \\
\end{align*}
\]

for $r = 1, 2 \cdots$, and $A_r \to$ a triangular matrix. One such choice leads to the $QR$ Algorithm for eigenproblems.
**QR Based Method**

This is a stable and efficient technique first introduced and analyzed by Rutishauser and Francis in the late 1950’s. The basic idea is,

- Factor $A_r = Q_r R_r$, where $Q_r$ is orthogonal and $R_r$ is upper triangular. Recall that $Q_r \equiv Q_1 Q_2 \cdots Q_{n-1}$ the cost of this decomposition is $2/3n^3$ flops.

- Set $A_{r+1} = R_r Q_r$. This can be accomplished, after factoring $A_r = Q_r R_r$, by forming $Q_r^T R_r^T$ as a sequence of $n - 1$ Householder reflections applied to $R_r^T$ and then taking the transpose to recover $R_r Q_r$ at a cost of $1/6n^3$ flops. That is,

$$A_{r+1}^T = Q_r^T R_r^T = [Q_{n-1} Q_{n-2} \cdots Q_1] R_r^T$$
Why Does it Work?

- $A_{r+1}$ is similar to $A_r$ since,

$$Q_r^{-1} A_r Q_r = Q_r^T (Q_r R_r Q_r) = (Q_r^T Q_r) R_r Q_r = A_{r+1}.$$  

To recover the eigenvector we must ‘remember’ each $Q_r$ and note that each is a product of $n-1$ Householder reflections.

- Let $Q_r = Q_1 Q_2 \cdots Q_r$ and $R_r = R_r R_{r-1} \cdots R_1$ then we have,

$$A_{r+1} = (Q_1 Q_2 \cdots Q_r)^T A Q_1 Q_2 \cdots Q_r,$$

$$= Q_r^T A Q_r.$$  

This result follows from the first observation and induction. (Note that we will never need to save $R_r$, and will only need to save $Q_r$ if the eigenvectors are required.)

Rutishauser proved that with this iteration the $A_r$ converge to an upper triangular matrix.
Why Does $A_r$ Converge?

For insight into why this is true consider,

$$Q_r R_r = Q_{r-1} (Q_r R_r) R_{r-1} = Q_{r-1} (A_r) R_{r-1}.$$  

and From the 2nd observation above,

$$Q_{r-1}^T A Q_{r-1} = A_r \quad \text{or} \quad Q_{r-1} A_r = A Q_{r-1}.$$

We then have, from these 2 equations,

$$Q_r R_r = Q_{r-1} A_r R_{r-1} = A Q_{r-1} R_{r-1},$$

which by induction implies the key observation,

$$Q_r R_r = A^r.$$  

That is we have the $QR$ decomposition of the $r^{th}$ power of $A$. There is then a close relationship then between the sequence $A_r$ and the power method. As the power method is known to converge, under some mild assumptions, it can be shown that this $QR$ iteration will also converge.
Rate of Convergence

The rate of convergence depends on ratios \((\lambda_j / \lambda_i)^r\) for \(j \neq i\), where \(r\) is the iteration number and \(\lambda_j\) and \(\lambda_i\) are the \(j^{th}\) and \(i^{th}\) eigenvalues of \(A\). Thus we will observe slow convergence for complex eigenvalues since such eigenvalues appear as complex conjugate pairs and have equal magnitudes.

If the magnitudes of the largest eigenvalues are not well separated one can apply a ‘shifted QR’ to accelerate convergence. The Shifted QR:

\[
(A_r - k_r I) = Q_r R_r,
\]

where,

\[
A_{r+1} = R_r Q_r + k_r I.
\]