V Eigenvalue Problems

<u>Aim</u>: Given a square matrix A, find eigenvalues / eigenvectors of A. ($n \times n$ matrix)

① Review/definitions

- Given a square matrix A, we define the eigenvectors $\underline{v}$ & associated eigenvalues λ the solutions of
$$A\underline{v} = \lambda \underline{v}$$
The set of all eigenvalues is the <u>spectrum</u> of A

- One eigenvalue may have more than 1 eigenvector associated to it. The subspace spanned by these eigenvectors is called the <u>eigenspace</u> associated to λ.

- To find λ & $\underline{v}$, we must solve
$$(A - I\lambda)\underline{v} = 0$$
which only has non-trivial solutions if $\det(A - I\lambda) = 0$

This yields an n-th order polynomial (if A is $n \times n$) equation called the characteristic equation
$$p_A(\lambda) = 0$$
Note that if A is real it may still have complex eigenvalues.

- $$p_A(\lambda) = k(\lambda - \lambda_1) \cdots (\lambda - \lambda_n)$$
where $\lambda_i \in \mathbb{C}$ and we allow for $\lambda_i = \lambda_j$

We define the <u>algebraic multiplicity</u> of a particular eigenvalue λ_k as the number of factors $(\lambda - \lambda_k)$ in $p_A(\lambda)$.

We define the <u>geometric multiplicity</u> of λ_k as the dimension of the eigenspace associated with λ_k.

The algebraic multiplicity is always greater or equal to the geometric multiplicity. If the algebraic multiplicity is strictly greater than the geometric multiplicity then the eigenvalue λ_k is called <u>defective</u>.

A matrix which has at least one defective eigenvalue is called a <u>defective matrix</u>

<u>Example</u>: The matrix $\begin{pmatrix} 1 & 1 \\ -1 & 3 \end{pmatrix}$ has an eigenvalue $\lambda = 2$ with algebraic multiplicity 2 but only one eigenvector $\underline{v} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$

- Rank & nullity: The <u>nullity</u> of A is the dimension of the eigenspace of the null eigenvalue. The rank of A is ($n -$ nullity)

② <u>Analytics vs numerics</u>

- When seeking Evalues & Evectors analytically, one usually first solves the characteristic equation to find the Evalues and then solve, for each λ_k,

$$A\underline{v}_k = \lambda_k \underline{v}_k$$

to find the Evector (or subspace) associated with λ_k

- Solving high-order polynomial equations numerically, in practise, is actually quite difficult. Moreover, once a solution is found, its algebraic multiplicity remains unknown.

$\Rightarrow$ Finding Evalues & Evectors numerically requires thinking about the problem entirely differently.

③ Similarity transformations / change of basis

A matrix actually represents an intrinsic geometric operation on vectors; as such, the properties of that geometric operation should be the same regardless of the coordinate system it is expressed in.

⇒ A change of basis (of coordinates) is usually referred to as a similarity transformation;

⇒ The properties of the transformed matrix (i.e. the matrix expressed in the new coordinate system) are invariant, and in particular

- ⊕ the determinant & the trace are invariant
- ⊕ the characteristic polynomial, the eigenvalues & eigenvectors are invariant
- ⊕ the rank & nullity are invariant
- ⊕ the rank & nullity are invariant.

Mathematically:

- A set of n vectors form a basis for $\mathbb{R}^n$ provided they are linearly independent (note that they don't need to be orthogonal).

- Suppose we have such a set $\{\underline{e}_1, \ldots, \underline{e}_n\}$ then a vector $\underline{x}$ has coordinates $(x_1, \ldots x_n)$ in the basis formed by $\{\underline{e}_i\}$ implies that

$$\underline{x} = \sum_{i=1}^{n} x_i \underline{e}_i$$

- Suppose we now have another (new) basis $\{\underline{n}_1 \ldots \underline{n}_n\}$, and the vector $\underline{x}$ has the new coordinates $\{x_1', \ldots, x_n'\}$ in the new basis

$$\Rightarrow \quad \underline{x} = \sum_{i=1}^{n} x_i \underline{e}_i = \sum_{k}^{n} x_k' \underline{n}_k$$

However, $\underline{n}_k = \sum_{i=1}^{n} (\underline{n}_k \cdot \underline{e}_i)\, \underline{e}_i = \sum_{i=1}^{n} (\underline{n}_k^T \underline{e}_i)\, \underline{e}_i$

effectively the j-th coordinate of $\underline{n}_k$ in the $\{\underline{e}_i\}$ basis

$$\Rightarrow \quad \underline{x} = \sum_{i=1}^{n} x_i \underline{e}_i = \sum_{k=1}^{n} x_k' \sum_{i=1}^{n} (\underline{n}_k^T \underline{e}_i)\, \underline{e}_i$$

$$\Rightarrow \quad x_i = \sum_k x_k' \, (\underline{n}_k^T \underline{e}_i)$$

let N be the matrix with coefficients $N_{ij} = \underline{n}_j^T \underline{e}_i$

$$\Rightarrow \quad \underline{x} = N \underline{x}'$$

$$\Rightarrow \quad \boxed{\underline{x}' = N^{-1} \underline{x}}$$

$\Downarrow$

The columns of N_{ij} are the coordinates of the j-th vector $\underline{n}_j$ in the old basis $\{\underline{e}_i\}$

The effect of a change of coordinates (change of basis) on a matrix (rather than a vector) does not change its geometrical intrinsic nature;

$\Rightarrow$

If $\quad A\underline{x} = \underline{y}$

then $\quad A N \underline{x}' = N \underline{y}'$

$$\Rightarrow \quad N^{-1} A N \underline{x}' = \underline{y}'$$

$$\Rightarrow \quad \underline{A}' \underline{x}' = \underline{y}'$$

is the expression of the same geometrical transformation in the new coordinate system provided

$$\boxed{A' = N^{-1} A N}$$

Note: If N is an orthogonal matrix $N^{-1} = N^T$

$$\Rightarrow \quad \boxed{A' = N^T A N}$$

A and A', share all of the properties listed earlier

$\Rightarrow$ If A & A' share the same Evectors & Evalues, then by finding an appropriate coordinate change we may be able to "reveal" the eigenvalues of A'.

A well-known but not very useful example

- Given a real, symmetric matrix A, if we knew the eigenvectors (which we don't), we could construct the matrix

$$V = \begin{pmatrix} | & | & & | \\ v_1 & v_2 & \cdots & v_n \\ | & | & & | \end{pmatrix}$$

$\leftarrow$ matrix with column vectors the Evectors of A

then

$$V^T A V = A' = \begin{pmatrix} \lambda_1 & & 0 \\ & \ddots & \\ 0 & & \lambda_n \end{pmatrix}$$

change of coordinate into the new basis formed by the Evectors

where, quite obviously, the eigenvalues of A' are the same as the eigenvalues of A.

ASIDE • Note the following theorems

① A is non-defective

$$\Longleftrightarrow \begin{pmatrix} \lambda_1 & & 0 \\ & \ddots & \\ 0 & & \lambda_n \end{pmatrix} = V^{-1} A V$$

where V is formed by the normalized E-vectors of A

(here $\underline{\lambda_i}$ and $\underline{v_i}$ can be complex)

② A is non-defective & $AA^* = A^*A \Longleftrightarrow VV^* = I$

③ A is non-defective & $AA^T = A^TA \Longleftrightarrow VV^T = I$

④ The Schur factorization.

- If A is an $n\times n$ matrix with Evalues $\{\lambda_i\}$ $(\lambda_i \in \mathbb{C})$ then there exists a unitary matrix P such that

$$A = PTP^* \qquad \text{(the Schur factorization of } A)$$

where
$$T = \begin{pmatrix} \lambda_1 & x & \cdots & x \\ & \ddots & \ddots & \vdots \\ 0 & & \ddots & x \\ & & & \lambda_n \end{pmatrix}$$
(i.e T is upper triangular with $\{\lambda_i\}$ in the diagonal).

- To prove this statement, we will construct an algorithm to create this factorization. But note that once T is known, the $\{\lambda_i\}$ are on the diagonal and can directly be read. Moreover this reveals the algebraic multiplicity of the Evalues.

- Another factorization is also very useful if we only want to work with real matrices: the real Schur factorization
$$A = Q\tilde{T}Q^T$$
where Q is a real orthogonal matrix and $\tilde{T}$ is a real upper triangular matrix except for 2×2 blocks around the diagonal containing the real and imaginary parts of the complex Evalues

$$\tilde{T} = \begin{pmatrix} \lambda_1 & x & x & x & x & \cdots \\ & \mathrm{Re}\lambda_2 & \mathrm{Im}\lambda_2 & x & & \vdots \\ & \mathrm{Im}\lambda_2 & \mathrm{Re}\lambda_2 & x & & \vdots \\ & & & \lambda_4 & \ddots & \vdots \\ & & & & & \lambda_n \end{pmatrix}$$
(eg here with λ_i real except for λ_2 and its complex conjugate λ_3)

- Note that both factorizations still work if the eigenvalues are defective

- To construct the Schur factorization we will first need to learn about the Hessenberg form.

The Hessenberg form

- It is always possible to find an orthogonal coordinate transformation

$$A = PHP^T \quad (\text{equivalently } H = P^T A P)$$

where

$$H = \begin{pmatrix} \times & \times & \cdots & \cdots & \times \\ \times & \times & \ddots & & \vdots \\ 0 & \times & \ddots & \ddots & \vdots \\ \vdots & 0 & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & \times & \times \end{pmatrix}$$

(i.e. H has $H_{ij} = 0$ if $i > j+1$).

- To prove this statement, we will construct an algorithm to put A into a Hessenberg form.

⑤ Searching for eigenvalues: a 2-step process

⇒ We will from now on construct an algorithm which reveals the (complex) eigenvalues of any matrix A by reduction onto a Schur form (of the real kind).

⇒ The standard algorithm for doing this consists in 2 steps

Step 1: Direct reduction of A to its Hessenberg form

Step 2: Iterative convergence of the initial Hessenberg form towards the real Schur form.

Why do this 2-step process?

One could think of using Householder matrices to reduce A to Schur form directly

$P^{(1)}A \rightarrow$ eliminates all subdiag elements of first column

$P^{(2)}P^{(1)}A \rightarrow$ ~ of second column

$\}$ $n-1$ times

then multiply to right by

$$P^{(n-1)} \cdots P^{(1)} A P^{(1)T} \cdots P^{(n-1)T}$$

Problem: the multiplication to the right repopulates the zeros with non-zero elements!

Example

$P^{(1)}A$ acts on the whole matrix A:

$$\begin{pmatrix} \boxed{\begin{array}{cccc} s_1 & X & \cdots & X \\ 0 & X & & X \\ \vdots & \vdots & & \vdots \\ 0 & X & \cdots & X \end{array}} \end{pmatrix}$$

influence region of $P^{(1)}$ on A

So $P^{(1)}AP^{(1)T}$ also acts on whole matrix A

(but $P^{(1)T}$ is not the correct Householder matrix for $P^{(1)}A$ so it just repopulates the zeros)

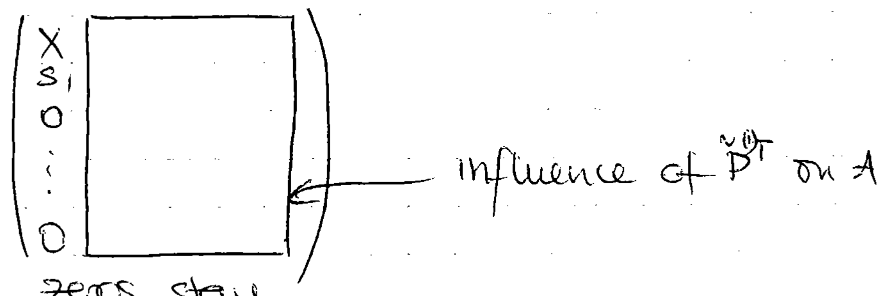
Idea: Instead construct the first Householder matrix to zero elements only below the subdiagonal

$$\tilde{P}^{(1)}A \quad \begin{pmatrix} X\,X\,X\,X\,X\,X \\ \boxed{\begin{array}{cccc} s_1 & X & \cdots & X \\ 0 & X & & \vdots \\ \vdots & \vdots & & \\ 0 & X & \cdots & X \end{array}} \end{pmatrix}$$

untouched row

influence block of $\tilde{P}^{(1)}$ on A.

so $\tilde{P}^{(1)} A \tilde{P}^{(1)T}$ leaves the first column untouched!



so the zeros stay.

$\Rightarrow$ This construction straightforwardly leads to the reduction of A through to a Hessenberg matrix which will then need further work to get to Schur form.

⑥ Algorithm(s) for Hessenberg reduction

There are Two possibilities for doing this, in fact:

- Procedure analogous to Gaussian elimination (see Numerical Recipes)
  Faster, but not necessarily always stable
- Householder reduction (as above).
  Always stable.

The algorithm for reduction into Hessenberg form using the Householder matrices is quite similar to the one used in QR decomposition. The main difference are

- A must be successively multiplied by $\tilde{p}^{(k)}$ on the left and by $\tilde{P}^{(k)T}$ on the right
- The $\tilde{P}^{(k)}$ are slightly different from the $p^{(k)}$ of the QR decomposition.
- Since the actual $\tilde{P}^{(k)}$ are not needed later (only H is), we return the full Hessenberg form into A.

Algorithm (for an $n\times n$ matrix A)

do $k = 1, n-2$

- consider $x^{(k)} = (0, 0, \dots 0, a_{k+1,k}, \dots a_{N,k})$
- $s_k = \pm \| x^{(k)} \|$
- $c_k = s_k (s_k - a_{k+1,k})$
- $A = \tilde{P}^{(k)} A \qquad (*)$
- $A = A \tilde{P}^{(k)T} \qquad (**)$

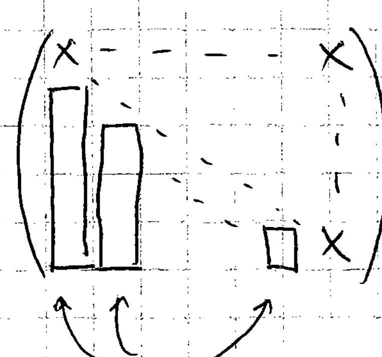
enddo.

The step $(*)$ depends on whether A is on exit entirely replaced by H (in which case the info on the $\tilde{w}$ vectors is lost) or if, as in the QR factorization, the $\tilde{u}$ vectors are stored <u>below</u> the subdiagonal of H whereas the subdiagonal itself is returned as separate vector.

<u>option 1</u>: on exit, A is H

$$\begin{pmatrix} X & X & X & \cdots & X \\ s_1 & X & & & \vdots \\ 0 & s_2 & \ddots & & \vdots \\ \vdots & 0 & \ddots & \ddots & X \\ 0 & 0 & \cdots 0 & s_{n-1} & X \end{pmatrix}$$

<u>Option 2</u> on exit A is



$\tilde{u}$ vectors

and the subdiagonal of H $(s_1 \dots s_{n-1})$ is returned separately.

$(*)$: Option 1

$a_{k+1,k} = s_k$

$a_{k+2:n,k} = 0$

do $j = k+1, n$

  do $i = k+1, n$

$$a_{ij} = a_{ij} - \sum_{m=k+1}^{n} \frac{a_{ik} a_{mk} a_{mj}}{c_k}$$

Note that neither s_k nor c_k needs to be returned.

* option 2

$$
\begin{cases}
a_{k+1,k} = a_{k+1,k} - s_k \\
\text{do } j = k+1, n \\
\quad \text{do } i = k+1, n \\
\quad\quad a_{ij} = a_{ij} - \displaystyle\sum_{m=k+1}^{n} \frac{a_{ik}\, a_{mk}\, a_{mj}}{c_k}
\end{cases}
$$

(Note that s_k must be stored & returned as separate vector, similarly for c_k)

<u>Step</u> (**)

<u>for both cases</u>.

$$
\begin{aligned}
&\text{do } i = 1, n \\
&\quad \text{do } j = k+1, n \\
&\quad\quad a_{ij} = a_{ij} - \sum_{m=k+1}^{n} \frac{a_{mk}\, a_{jk}\, a_{im}}{c_k}
\end{aligned}
$$

<u>Note</u>: All of the above applied to general matrices

⇒ If A is a real symmetric matrix then the Hessenberg form is tridiagonal and the Schur form is the diagonal matrix of the eigenvalues. In that case too, P is the matrix of the orthogonal eigenvectors