Lecture 7 Matrices and Eigensystems

- As we have seen, physics is full of eigenvalue equations,

e.g. $$\overbrace{\left(-\frac{\hbar^2}{2m}\nabla^2 + V\right)}^{H}\psi_n = E_n\psi_n$$

① Special form of V,

Solns are given by special functions,

- Hermite ($V(x) = \frac{1}{2}kx^2$)
- Spherical Harmonics (spherical coordinates)
- Laguerre (Central Force)

⋮

— Solve only limited form of H.

② Numerical Soln to Diff. Eq. (HW #1)

③ Matrix Solutions

General Approach

$$H\Psi_\alpha = E_\alpha \Psi_\alpha$$

① Find a complete set of suitable basis states $\{\phi_n\}$.

— Such that — $\{\phi_\alpha\}$ is complete (any wavefn can be expanded in terms of $\{\phi_n\}$)

$$\Psi_\alpha = \sum_{i=1}^{n} C_{\alpha,i}\,\phi_i$$

— $\{\phi_\alpha\}$ is an orthonormal set

$$\langle \phi_\alpha | \phi_\beta \rangle = \delta_{\alpha,\beta}$$

— suitable : $\{\phi_n\}$ based on some knowledge of the system such as symmetry or

$H = H_0 + \Delta H$ such that $H_0 \phi_\alpha = E_\alpha \alpha$

e.g. $V(x) = \frac{1}{2} M\omega^2 x^2 + \epsilon \hbar\omega \left(\frac{M\omega}{\hbar}\right)^2 x^4$

(We have already seen 1st order perturbation theory HW #3 in getting ΔE)

This is an alternative approach

(2)

$$H|\psi_\alpha\rangle = E_\alpha|\psi_\alpha\rangle$$

(3)

$$\langle\phi_j|H|\psi_\alpha\rangle = E_\alpha\langle\phi_j|\psi_\alpha\rangle \quad (*)$$

In terms of $|\psi_\alpha\rangle = \sum_{i=1}^{n} C_{\alpha,i}|\phi_i\rangle$, (n typically is infinite)

$$(*) \Rightarrow \sum_{i=1}^{n} H_{j,i}\, C_{\alpha,i} = E_\alpha C_{\alpha,j}$$

$$H_{ji} = \langle\phi_j|H|\phi_i\rangle$$

☆ This is a matrix eigenvalue equation,

$$\begin{pmatrix} H_{11} & H_{12} & & \\ H_{21} & \cdots & & \\ & & & H_{nn} \end{pmatrix}\begin{pmatrix} C_{\alpha 1} \\ \vdots \\ C_{\alpha n} \end{pmatrix} = E_\alpha \begin{pmatrix} C_{\alpha 1} \\ \vdots \\ C_{\alpha n} \end{pmatrix}$$

Note: typically H_{ij} is symmetric

$$(\underset{\sim}{H} - \underset{\sim}{I}E)\cdot \underset{\sim}{C} = 0$$

☆ So, E_α, $|\psi_\alpha\rangle$ can be solved by this matrix equation!

☆ The problem becomes find the eigenvalues E & eigenvectors $\underset{\sim}{C}$ of the above equation.

(Note: n is infinite for most system but a finite truncation usually gives good results for low energy states.)

Eigenvalues & Eigenvectors of Matrices

(review of terms)

$$\underset{\sim}{A} \cdot \underset{\sim}{X}_R = \lambda \underset{\sim}{X}_R$$

$\underset{\sim}{A}$: General matrix

λ: eigenvalue

$\underset{\sim}{X}_R$: right eigenvectors

<u>Note</u>: λ's are solution to $\det(\underset{\sim}{A} - \underset{\sim}{I}\lambda) = 0$

$$\underset{\sim}{X}_L \cdot \underset{\sim}{A} = \lambda' \underset{\sim}{X}_L$$

$\underset{\sim}{X}_L$: left eigenvectors

☆ $\lambda = \lambda'$ (left eigenvalues = right eigenvalues)

$$\underset{\sim}{X}_L \cdot \underset{\sim}{A} \cdot \underset{\sim}{X}_R = \underset{\sim}{X}_L \lambda \underset{\sim}{X}_R \Rightarrow \lambda' \underset{\sim}{X}_L \cdot \underset{\sim}{X}_R = \lambda \underset{\sim}{X}_L \cdot \underset{\sim}{X}_R$$

Symmetric Matrices

Def. $\underset{\sim}{A} = \underset{\sim}{A}^T$

$$\longrightarrow \quad \underset{\sim}{X}_L = \underset{\sim}{X}_R^T$$

To show: $A X_R = \lambda X_R$

$A^T X_R = \lambda X_R$

$X_R^T A = \lambda X_R^T \Rightarrow \underset{\sim}{X_L = X_R^T}$

Orthogonality

$\overline{X}_R$ - columns of $\underset{\sim}{x}_R$

$\overline{X}_L$ - rows of $\underset{\sim}{x}_L$

$\underset{\sim}{A}\,\underset{\sim}{x}_R^i = \lambda^i \underset{\sim}{x}_R^i$

$\downarrow$

$\underset{\sim}{A}\begin{pmatrix} x_R^1 & x_R^2 \end{pmatrix} = \begin{pmatrix} \underset{\sim}{x}_R^1 & \underset{\sim}{x}_R^2 \end{pmatrix}\begin{pmatrix} \lambda^1 & \\ & \lambda_2 \end{pmatrix}$

check: $\begin{pmatrix} x_{11} & x_{12} \\ x_{21} & x_{22} \end{pmatrix}\begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix}$

$= \begin{pmatrix} \lambda_1 x_{11} & \lambda_2 x_{12} \\ \lambda_1 x_{21} & \lambda_2 x_{22} \end{pmatrix}$

$= \begin{pmatrix} \lambda_1 \underset{\sim}{x}_R^1 & \lambda_2 \underset{\sim}{x}_R^2 \end{pmatrix}$

$$A \cdot \overline{X}_R = \overline{X}_R \operatorname{diag}(\lambda_1 \lambda_2 \lambda_3 \ldots \lambda_n)$$

$$\overline{X}_L \cdot A = \operatorname{diag}(\lambda_1 \lambda_2 \ldots \lambda_n)\, \overline{X}_L$$

$\downarrow$

$$\overline{X}_L A \overline{X}_R = \overline{X}_L \overline{X}_R \operatorname{diag}(\lambda_1 \ldots \lambda_n)$$

$$\ominus \quad \overline{X}_L A \overline{X}_R = \operatorname{diag}(\lambda_1 \lambda_2 \ldots \lambda_n)\, \overline{X}_L \overline{X}_R$$

$$\Rightarrow \quad \overline{X}_L \overline{X}_R \operatorname{diag}(\lambda_1 \ldots \lambda_n) = \operatorname{diag}(\lambda_1 \ldots \lambda_n)\, \overline{X}_L \overline{X}_R$$

$\Rightarrow$ $\overline{X}_L \overline{X}_R$ is diagonal

☆ Thus, left eigenvectors are orthogonal to righteigenvectors!

Symmetric Matrices : $\overline{X}_L \overline{X}_R = \text{diag}$ orthonormal.

$\overline{X}_L \overline{X}_L^T = \underset{\sim}{I}$

Direct Method

recall eigensystem equation:

$$(\underset{\sim}{A} - \underset{\sim}{I}\lambda) \cdot \underset{\sim}{x} = 0$$

★ λ's are determined by

$$\det(\underset{\sim}{A} - \underset{\sim}{I}\lambda) = 0$$

$$\parallel$$

$$f(\lambda) = 0$$

$\Rightarrow$ Eigenvalues soln $\approx$ zeros of $f(\lambda)$!

- Newton's Method if $f'(\lambda)$ can be easily calculated
- Bisection, false position, Secant.

Evalution of Characteristic Polynomials from Tridiagonal Form

- From Householder:

$$P_{n-2} \cdots P_1 A P_1 \cdots P_{n-2} - I\lambda = (\hat{A} - I\lambda) = \begin{pmatrix} \alpha_1-\lambda & \beta_2 & & & & \\ \gamma_2 & \alpha_2-\lambda & \beta_3 & & & \\ & \gamma_3 & \alpha_3-\lambda & \beta_4 & & \\ & & \gamma_4 & \ddots & & \\ & & & & \alpha_{n-1}-\lambda & \beta_n \\ & & & & \gamma_n & \alpha_n-\lambda \end{pmatrix}$$

- characteristic equation defined by

$0 = \det(\hat{A} - I\lambda)$ will be a polynomial in λ of order n.

eigenvalues λs are the zeros of this equation!

- Because of the form of the tridiagonal form,

$P_n(\lambda) = \det(\hat{A} - I\lambda)$ can be found recursively:

$$P_j(\lambda) = (\alpha_j - \lambda) \cdot P_{j-1}(\lambda) - \beta_j \gamma_j P_{j-2}(\lambda)$$

with $P_0(\lambda) = 1 \qquad P_1(\lambda) = \alpha_1 - \lambda$.

Can easily check with a small matrix.

- Similar to other recursion relations, this is the most effectively used <u>arithmetically</u> (i.e., for a specific value of λ, the recursion generates a sequence of values correspond to $P_n(\lambda)$'s).

→ With $\det(\hat{A} - I\lambda) = P_n(\lambda) = 0$ found, then λ's can be found by bisection, Newton's method, etc.

Power Method

- Let consider an arbitrary vector

$$\underset{\sim}{x} = \sum a_j \underset{\sim}{u}_j$$

where $\quad \underset{\sim}{A} \cdot \underset{\sim}{u}_j = \lambda_j \underset{\sim}{u}_j$

- Also, for convenience, we order our λ's so that

$$|\lambda_1| > |\lambda_2| > \ldots > |\lambda_N|$$

Now, let $\underset{\sim}{A}$ operates on x

$$\begin{aligned}\underset{\sim}{A} \cdot \underset{\sim}{x} &= \underset{\sim}{A} \sum a_j \underset{\sim}{u}_j \\ &= \sum a_j \underset{\sim}{A} \underset{\sim}{u}_j \\ &= \sum a_j \lambda_j \underset{\sim}{u}_j\end{aligned}$$

For repeated application of $\underset{\sim}{A}$,

$$\underset{\sim}{A}^m \underset{\sim}{x} = \sum a_j \lambda_j^m \underset{\sim}{u}_j$$

$$= \lambda_1^m \left(a_1 \underset{\sim}{u}_1 + \sum_{j=2}^{N} a_j \left(\frac{\lambda_j}{\lambda_1}\right)^m \underset{\sim}{u}_j \right)$$

Since $|\lambda_1| > |\lambda_j| \quad \forall j \neq 1$,

$$\left(\frac{\lambda_j}{\lambda_1}\right)^m \to 0 \quad \text{for } m \to \infty.$$

$\Rightarrow$ for large m $\quad \underset{\sim}{A}^m \cdot \underset{\sim}{x} \sim \lambda_1^m a_1 \underset{\sim}{u}_1$

1 $$\lim_{m\to\infty} \frac{|\underset{\sim}{A}^{m+1} \cdot \underset{\sim}{x}|}{|\underset{\sim}{A}^m \cdot \underset{\sim}{x}|} = \lim_{m\to\infty} \frac{\lambda_1^{m+1} a_1 |\underset{\sim}{u}_1|}{\lambda_1^m a_1 |\underset{\sim}{u}_1|} = \lambda_1$$

or 2 $$\lim_{m\to\infty} |\underset{\sim}{A}^m \cdot \underset{\sim}{x}|^{1/m} = \lambda_1 \lim_{m\to\infty} (a_1 |\underset{\sim}{u}_1|)^{1/m}$$

$$= \lambda_1$$

(useful later for Lyapunov exponents)

Smallest Eigenvalue λ_N :

→ Do the same with $\underset{\sim}{A}^{-1} \underline{x}$!

since $1/\lambda_N$ will be the largest eigenvalue of $\underset{\sim}{A}^{-1}$!

A particular Implementation

- get both λ_1 & $\underset{\sim}{u}_1$ —

- Consider an arbitrary <u>unit</u> vector

$\underset{\sim}{e}_0$ such that $|\underset{\sim}{e}_0| = 1$.

- Then form the follow sequences.

$$\underset{\sim}{e}_1 = \underset{\sim}{A} \cdot \underset{\sim}{e}_0 / \gamma_0 \quad \text{where } \gamma_0 = |\underset{\sim}{A} \cdot \underset{\sim}{e}_0|$$

↑
normalization constant.

and $\underset{\sim}{e}_2 = \underset{\sim}{A} \cdot \underset{\sim}{e}_1 / \gamma_1 \;, \quad \gamma_1 = |\underset{\sim}{A} \cdot \underset{\sim}{e}_1|$

Similar arguement as before,

$$\Rightarrow \quad \xi_m \to \lambda_1 \quad \text{as } m \to \infty$$

$$\underset{\sim}{e}_m \to \underset{\sim}{u}_1 \quad \text{as } m \to \infty.$$

☆ – extension to $\{A_m\}$ with varying A_m's

→ basic numerical method to get Lyapunov exponents.

To get eigenvector fast if we know a good estimate of λ.

$$\underset{\approx}{A}\,\underset{\sim}{u}_j = \lambda_j \underset{\sim}{u}_j$$

- let say we have ξ being a good estimate of λ

then consider,

subtracting $\xi\underset{\sim}{u}_j$ from both sides.

$$(\underset{\approx}{A} - \underset{\approx}{I}\xi)\,\underset{\sim}{u}_j = (\lambda_j - \xi)\,\underset{\sim}{u}_j$$

- this is eigenvalue equation again:

$\underset{\sim}{u}_j$ is also an eigenvector of $\underset{\approx}{A} - \underset{\approx}{I}\xi$ with eigenvalue $(\lambda_j - \xi)$.

☆ - Now, since $\lambda_j \simeq \xi$, $\lambda_j - \xi \simeq 0$,

☆ - if we apply inverse Power Method on $\underset{\approx}{A} - \underset{\approx}{I}\xi$:

Note: If $(\underset{\sim}{A} - \underset{\sim}{I}\xi)\, u_j = (\lambda_j - \xi)\, u_j$,

multiply $\frac{1}{\lambda_j - \xi}(\underset{\sim}{A} - \underset{\sim}{I}\xi)^{-1}$ to both sides,

$$\left(\frac{1}{\lambda_j - \xi}\right) u_j = (\underset{\sim}{A} - \underset{\sim}{I}\xi)^{-1} u_j$$

---

① $\frac{1}{\lambda_j - \xi}$ will be very large.

② u_j is an eigenvector of $(\underset{\sim}{A} - \underset{\sim}{I}\xi)$ with $\lambda_j - \xi$ and it is also an eigenvector of $(\underset{\sim}{A} - \underset{\sim}{I}\xi)^{-1}$ with $\frac{1}{\lambda_j - \xi}$

☆ ⇒ With inverse power method on ⊛

$\underset{\sim}{u}_j$ can be obtained quickly!

Recall that the iterative procedure for power method looks like,

- start with $\underset{\sim}{e}_0$ = normalized random vector
- $(\underset{\approx}{A} - \underset{\approx}{I}\delta)^{-1} \cdot e_m$ (apply $(\underset{\approx}{A} - \underset{\approx}{I}\delta)^{-1}$)
- form $k_m = |(\underset{\approx}{A} - \underset{\approx}{I}\delta)^{-1} \cdot \underset{\sim}{e}_m|$

  $\underset{\sim}{e}_{m+1} = (\underset{\approx}{A} - \underset{\approx}{I}\delta)^{-1} \cdot \underset{\sim}{e}_m / k_m$

<u>Alternative without explicitly calculating $(\underset{\approx}{A} - \underset{\approx}{I}\delta)^{-1}$</u>

<u>Note</u>: $(\underset{\approx}{A} - \underset{\approx}{I}\delta)^{-1} \underset{\sim}{e}_m = (k_m \underset{\sim}{e}_{m+1})$

$$\rightarrow \underbrace{(\underset{\approx}{A} - \underset{\approx}{I}\delta)}_{\text{given}} \underbrace{\underset{\sim}{x}}_{\text{need to solve}} = \underbrace{\underset{\sim}{e}_m}_{\text{known}}$$

- Linear Eq for $\underset{\sim}{x}$ can be solve with LU (with pivoting)
- normalized $\underset{\sim}{x} \rightarrow \underset{\sim}{e}_{m+1} = \dfrac{\underset{\sim}{x}}{|\underset{\sim}{x}|}$
- repeat LU backsub!

★ (LU decomp is already done)

★ GJ shouldn't be reused since it involve RHS at each application

General Tactics in Solving Eigenvalue Problems

- A few more definitions:

normal / defective matrices: → def. $[\underset{\sim}{A}, \underset{\sim}{A}^H] = \underset{\sim}{A}\underset{\sim}{A}^H - \underset{\sim}{A}^H\underset{\sim}{A} = 0$
H - conjugate-transpose

- A matrix $\underset{\sim}{A}$ normal, one can always find a complete set orthogonal eigenvectors that span the N-dim vector space.

- A matrix $\underline{A}$ is defective if its set eigenvectors is not complete.

Symmetric / Hermitian (they are normal):

Symmetric : $\underset{\sim}{A} = \underset{\sim}{A}^T$ (real matrices)

Hermitian : $\underset{\sim}{A} = \underset{\sim}{A}^H$ (complex matrices)
↳ Hermitian conjugate

- if $\underline{A}^H \cdot \underline{A} = \underline{A} \cdot \underline{A}^H$, $\underline{A}$ is normal.
(Commute)

☆ - the complete set of eigenvalues are real for sym / Hermitian matrices!

- nonsymmetric / nonHermitian matrices might have complex eigenvalues.

- Starting with our eigenvalue Problem definition,

$$\underline{\underline{A}} \cdot \underline{\underline{X}}_R = \underline{\underline{X}}_R \cdot \mathrm{diag}(\lambda_1, \ldots \lambda_N)$$

If $\underline{\underline{A}}$ is non defective, $\underline{\underline{X}}_R^{-1}$ exists, and

$$\underline{\underline{X}}_R^{-1} \cdot \underline{\underline{A}} \cdot \underline{\underline{X}}_R = \mathrm{diag}(\lambda_1, \ldots, \lambda_N)$$

- The left hand side is a <u>similarity transformation</u>

[def. $\underline{\underline{Z}}^{-1} \cdot \underline{\underline{A}} \cdot \underline{\underline{Z}} = \underline{\underline{A}}'$ where $\underline{\underline{Z}}$ is a nonsingular matrix.

- This similarity transformation is <u>special</u> such that the particular choice of $\underline{\underline{Z}} = \underline{\underline{X}}_R$ <u>diagonalizes</u> $\underline{\underline{A}}$!

☆ Basic idea in most transformation-based eigenvalue solver:

- to find the correct $\underline{\underline{Z}}$ such that

$$\underline{\underline{Z}}^{-1} \cdot \underline{\underline{A}} \cdot \underline{\underline{Z}} = \mathrm{diag}(\lambda_1, \ldots, \lambda_N)$$

☆ Note : In the previous discussion, $\underline{\underline{A}}$ can be symmetric or nonsymmetric.

- For nonsymmetric $\underline{\underline{A}}$,

  the correct $\underline{\underline{Z}}$ that diagonalizes $\underline{\underline{A}}$,

  i.e. $\underline{\underline{Z}}^{-1} \cdot \underline{\underline{A}} \cdot \underline{\underline{Z}} = \text{diag}(\lambda_1, \ldots, \lambda_N)$

  might be complex !

☆ - If $\underline{\underline{Z}}$ is restricted to real matrices (in most numerical applications),

$$\underline{\underline{Z}}^{-1} \cdot \underline{\underline{A}} \cdot \underline{\underline{Z}} \longrightarrow \text{Jordan form at best}$$

- block diagonal with 2x2 blocks → complex conjugate eigenvalues!

$$\begin{pmatrix} \lambda_1 & 0 & 0 & \\ 0 & a & b & \\ 0 & c & d & \\ & & & \lambda_2 \end{pmatrix}$$

↳ all real entries

- For symmetric $\underline{\underline{A}}$, with real $\underline{\underline{Z}}$

  $\underline{\underline{Z}}^{-1} \cdot \underline{\underline{A}} \, \underline{\underline{Z}} \rightarrow \text{diag}(\lambda_1, \ldots, \lambda_2)$

  - λ_i real
  - $\underline{\underline{Z}}$ orthonormal & real.

Important Point about Similarity Transformation:

$$\det(\underline{\underline{Z}}^{-1}\cdot\underline{\underline{A}}\cdot\underline{\underline{Z}} - \lambda\underline{\underline{1}})$$

$$= \det(\underline{\underline{Z}}^{-1}\cdot(\underline{\underline{A}} - \lambda\underline{\underline{1}})\cdot\underline{\underline{Z}})$$

$$= \det\underline{\underline{Z}}^{-1}(\det(\underline{\underline{A}} - \lambda\underline{\underline{1}}))\det\underline{\underline{Z}}$$

$$= \det(\underline{\underline{A}} - \lambda\underline{\underline{1}})$$

$\Rightarrow$ So, $\underline{\underline{A}}' = \underline{\underline{Z}}^{-1}\cdot\underline{\underline{A}}\cdot\underline{\underline{Z}}$ will have the same eigenvalues as $\underline{\underline{A}}$!

Basic Numerical Strategy

☆ Attempt to transform A into diagonal form by a sequence of real similarity transforms

$$A \to P_1^{-1} A P_1 \to P_2^{-1} P_1^{-1} A P_1 P_2 \to$$
$$P_3^{-1} P_2^{-1} P_1^{-1} A P_1 P_2 P_3 \to \ldots \underset{\sim}{\mathrm{diag}(\lambda_1, \ldots, \lambda_n)}$$

where $\underset{\sim}{X_R = P_1 P_2 P_3 \ldots}$ gives the eigenvectors

---

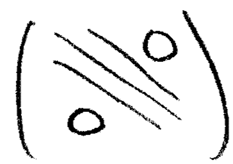
☆ If only eigenvalues are of interests, then we don't need to reduce A all the way to diagonal $\longrightarrow$ triangular (by similar transforms)

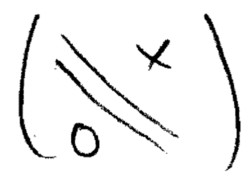
- less work
- λ will be diagonal elements

☆☆ A <u>finite</u> sequence of these transforms usually can't take A to actual diagonal or triangular form! ex. Jacobi QR/QL

Pre-process

- speed of convergence of these transformation algorithm can be significantly improved if $\underline{\underline{A}}$ is first put in special forms by a smaller # of preprocess steps.

standard shapes : tridiagonal  TD (sym)

Hessenberg  H (non-sym)

Householder Transform ($\underline{\underline{A}} \longrightarrow$ TD or H)

☆ Can reduce a $n \times n$ matrix $\underline{\underline{A}}$ to TD or H in $(n-2)$ steps!

(it annihilates the required part of the cols & [the corresponding row] in one step.)
(for symmetric)

Construction:

Householder matrix $\quad \underline{\underline{P}} = \underline{\underline{1}} - 2\,\underline{w} \cdot \underline{w}^T$, $\underline{w}$ real vector, $|\underline{w}|^2 = 1$

Note: $w \cdot w^T$ is a matrix (outer product) $\quad \begin{pmatrix} w \end{pmatrix} \begin{pmatrix} w^T \end{pmatrix} = \begin{pmatrix} & \end{pmatrix}$

13

$w^T \cdot w$ is a scalar (inner product) $\quad \begin{pmatrix} w^T \end{pmatrix} \begin{pmatrix} w \end{pmatrix} = 1$

Note:

$$P \cdot P = (1 - 2ww^T)(1 - 2ww^T)$$

$$= 1 - 4ww^T + 4w \cdot \underbrace{(w^T \cdot w)}_{= 1 \text{ by def.}} w^T$$

$$= 1$$

☆☆ Householder transform is orthogonal.

$$\Rightarrow \quad P^{-1} = P$$

And, since $P = P^T$ (by construction), $P^T = P^{-1}$ (orthogonal)

Rewriting $\underline{P}$ as:

$$P = 1 - \frac{u \cdot u^T}{H}$$

← in this case u can be any vector

where $H = \frac{1}{2} |u|^2$

Suppose $\underset{\sim}{x}$ is the first column of A,

choose $\underset{\sim}{u} = \underset{\sim}{x} \mp |x| \underset{\sim}{e}_1$

where $\underset{\sim}{e}_1 = (1\ 0\ 0\ 0)^T$ unit vector.

We want see ~~the actor~~ the action of $\underset{\sim}{P}$ on $\underset{\sim}{x}$:

$$\underset{\sim}{P} \cdot \underset{\sim}{x} = x - \frac{u}{H} \cdot (x \mp |x| e_1)^T \cdot x$$

$$H = \frac{1}{2}|u|^2 = \frac{1}{2} u^T u$$

$$= \frac{1}{2} (\underset{\sim}{x} - |x| \underset{\sim}{e}_1)^T (\underset{\sim}{x} - |\underset{\sim}{x}| \underset{\sim}{e}_1)$$

$$= \frac{1}{2} \left(2|x|^2 - 2|x| x_1\right)$$

$$= |x|^2 - |x| x_1$$

so, $$\underset{\sim}{P} \cdot \underset{\sim}{x} = x - u \frac{\cancel{|x|^2 \mp |x| x_1}}{\cancel{|x|^2 \mp |x| x_1}}$$

$$= \pm |x| \underset{\sim}{e}_1$$

So, the effect of $\underset{\sim}{P}$ on $\underset{\sim}{x}$ is to reduce it to one with only one nonzero element.

To reduce a symmetric matrix to tridiagonal

We choose $\underset{\sim}{x}$ for the first householder matrix P_1 to be the lower (n-1) elements of the 1st column.

Thus,

$$P_1 \cdot A = \left(\begin{array}{c|c} 1 & 0 \\ \hline 0 & {}^{(n-1)}P_1 \end{array} \right) \cdot \left(\begin{array}{c|c} a_{11} & a_{12} \cdots a_{1n} \\ \hline a_{21} & \\ \vdots & \times \\ a_{n1} & \end{array} \right)$$

$$= \left(\begin{array}{c|c} a_{11} & \overbrace{a_{12} \cdots a_{1n}}^{\text{unchanged by } \underset{\approx}{P}} \\ \hline k & \\ 0 & \times \\ \vdots & \\ 0 & \end{array} \right)$$

Where $k = |(a_{21}, \ldots a_{n1})|$ $\rightarrow$ magnitude of partial column of A.

${}^{(n-1)}P_1$ - $(n-1)\times(n-1)$ householder matrix with $x = (a_{21}, \ldots, a_{n1})^T$

$$\underset{\approx}{P} = \underset{\approx}{1} - \frac{\underset{\sim}{u} \cdot \underset{\sim}{u}^T}{H}$$

$$\underset{\sim}{u} = \underset{\sim}{x} - |x| \underset{\sim}{e}_1$$

The complete orthogonal transformation is then

$$A' = P \cdot A \cdot P = \left(\begin{array}{c|cc} a_{11} & k & 0 \\ \hline k & & \\ 0 & & X \end{array} \right)$$

$\uparrow$

$(P^{-1} = P^T = P)$

Note: Householder is a similarity transform also.

Note: P^T with the same x reduces the first row because A is symmetric!

<u>2nd step:</u>

Choose

$$P_2 = \left(\begin{array}{c|c} \begin{matrix} 1 & 0 \\ 0 & 1 \end{matrix} & 0 \\ \hline 0 & {}^{(n-2)}P_2 \end{array} \right)$$

where ${}^{(n-2)}P_2$ is a $(n-2)\times(n-2)$ householder matrix with $x = (a'_{32} \dots a'_{n2})^T$ being the $(n-2)$ elements of the 2nd column of A'.

$\rightarrow$ this reduces A' to $A'' = \left(\begin{array}{cc|cc} a_{11} & k & 0 & 0 \\ k & a_{22} & k' & 0 \\ \hline 0 & k' & & \\ 0 & 0 & & X \end{array} \right)$

☆ Thus, for a total of $(n-2)$ householder transforms $P_{n-2} \cdots P_1 A P_1 \cdots P_{n-2}$ reduces A to a tridiagonal form! (17)

<u>Note</u>: Since <u>the cancelations of large numbers tends to reduce accuracy</u>, it is perferred to arrange the matrix so that the largest off diagonal elements will be taken care of last.

— (by row or column rotation)

<u>Note 2</u>: Instead of matrix multiplications:

$$P \cdot A \cdot P$$

it is more efficient to compute

$$\underset{\sim}{P} \equiv \frac{A \cdot u}{H} \quad \text{(a vector)}$$

Then $$\underset{\sim}{A} \cdot \underset{\sim}{P} = A \cdot \left(1 - \frac{u \cdot u^T}{H}\right) = \underset{\sim}{A} - \underset{\sim}{P} \cdot \underset{\sim}{u}^T$$

$$A' = P \cdot A \cdot P = A - \underset{\sim}{P} \cdot \underset{\sim}{u}^T - \underset{\sim}{u} \cdot \underset{\sim}{P}^T + 2K \underset{\sim}{u} \cdot \underset{\sim}{u}^T$$

$$\underset{\sim}{q} = \underset{\sim}{p} - K\underset{\sim}{u}$$

the $\underline{A}' = A - \underset{\sim}{q}\underset{\sim}{u}^T - \underset{\sim}{u}\underset{\sim}{q}^T$

steps:

(1) $\rightarrow$ $\underset{\sim}{u}$

(2) $\rightarrow$ $H = \frac{1}{2}|u|^2$

(3) $\rightarrow$ $\underset{\sim}{p} = \dfrac{\underline{A} \cdot \underset{\sim}{u}}{H}$

(4) $\rightarrow$ $K = \dfrac{\underset{\sim}{u}^T \cdot \underset{\sim}{p}}{2H}$

(5) $\rightarrow$ $\underset{\sim}{q} = \underset{\sim}{p} - K\underset{\sim}{u}$

(6) $\rightarrow$ $\underset{\sim}{A}' = A' - \underset{\sim}{q}u^T - \underset{\sim}{u}\underset{\sim}{q}^T$

only vector · vector & matrix · vector.

— Now, we look at two representative transformation ← with real similarity transform methods that take $\underline{\underline{A}}$ to diagonal form or triangular form. [For complex or degenerate λ's, other considerations need to be taken. Press.]

★ Note: Different from the preprocess transform, the reduction to diag or Triangular form can't be done in finite steps!

(1) Jacobi transform (for real symmetric matrices)

- individual P_i are rotations to zero two specific off-diagonal elements.

- based on the idea that

$$R^{-1}\begin{pmatrix} a & b \\ b & d \end{pmatrix} R \rightarrow \begin{pmatrix} a' & 0 \\ 0 & d' \end{pmatrix}$$

$$R = \begin{pmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{pmatrix}$$

- problem: already zeroed elements can be changed back to nonzero by subsequent P's.

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & c & s & 0 \\ 0 & -s & c & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}\begin{pmatrix} x & 0 & x & x \\ 0 & x & \boxed{a} & x \\ x & \boxed{a} & x & x \\ x & x & x & x \end{pmatrix} = \begin{pmatrix} x & \epsilon & x & x \\ \epsilon & x & 0 & x \\ x & 0 & x & x \\ x & x & x & x \end{pmatrix}$$

☆ these elements might become nonzero BUT smaller!

- It can be shown that upon repeated application of P_i's, $\varepsilon \rightarrow 0$ (all the off diagonal elements converge to 0!)

- While foolproof for real symmetric matrices, convergence takes time!

③ QR / (~~QL~~) transforms (work with non-symmetric $\underline{A}$ also)

$$A = Q_1 R \quad (*)$$

Any real matrix can be decomposed into
Q - orthogonal
R - upper triangular
(L - lower triangular)
- (we will show explicitly later with householder)

Now, consider

$$A' = RQ$$

Since Q is orthogonal, $R = Q^T A$ (from (*))
$(QQ^T = 1)$

$$\Rightarrow \underline{A' = Q^T A Q}$$

☆ (A' is a orthogonal similarity transform of A)

- with A', we can decompose it again,
$A' = Q_1 R_1$, then $A^2 = R_1 Q_1 = Q_1^T A' Q_1$
$= Q_1^T Q_0^T A Q_0 Q_1$

→ Thus, we can generate a sequence of orthogonal similarity transform or A

☆ Again, it can be shown that

1). If A has λs of different absolute values

then $A^n \to D$ (if A is symmetric)

$A^n \to T$ (if A is nonsymmetric)

see F.S. Acton "Num Methods that usually works"

(in both case eigenvalues can be read off from the diagonal!) [For complex λ's, see Press]

2) if λ's are degenerate

then $A^n \to$ D with blocks (if A is symmetric)

$A^n \to$ T with blocks (if A is non-symmetric)

☆☆ Convergent is <u>fast</u> for tridiagonal matrix $O(n)$! $\begin{pmatrix} x & x & 0 \\ x & x & x \\ 0 & x & x \end{pmatrix}$ (D with super or sub diagonals)

and $O(n^2)$ for Hessenberg matrix with <u>shifting</u>.

$\begin{pmatrix} x & x & x \\ x & x & x \\ 0 & x & x \end{pmatrix}$ (T with subdiagonal)

The QR decomposition

- Recall the householder transform can

reduce $\underset{\sim}{x} = \begin{pmatrix} a_1 \\ \vdots \\ a_n \end{pmatrix} \rightarrow \begin{pmatrix} |\underset{\sim}{x}| \\ 0 \\ \vdots \\ 0 \end{pmatrix} = \underset{\sim}{P} \cdot \underset{\sim}{x}$

$$\underset{\sim}{P} = \underset{\approx}{1} - \frac{\underset{\sim}{u}\,\underset{\sim}{u}^T}{H} \qquad \underset{\sim}{u} = \underset{\sim}{x} - |\underset{\sim}{x}|\underset{\sim}{e}_1$$

$$H = \tfrac{1}{2}|\underset{\sim}{u}|^2 = \tfrac{1}{2}\underset{\sim}{u}^T \cdot \underset{\sim}{u}$$

this is orthogonal!

- Just like the reduction to tridiagonal form, a sequence of $(n-1)$ $\underset{\sim}{P}$'s can reduce $\underset{\sim}{A}$ to upper triangular form,

such that $P_{n-1} \cdots P_1 A = R$

Note : $\underset{\approx}{P}_1$: $\underset{\sim}{x}$ = 1st column of $\underset{\approx}{A}$

then $\underset{\approx}{P}_1 \underset{\approx}{A} = \begin{pmatrix} k & x & \\ 0 & x & \ddots \\ 0 & x & \\ 0 & x & \end{pmatrix}$

$\underset{\approx}{P}_2 = \left(\begin{array}{c|ccc} 1 & 0 & 0 & \cdots \\ \hline 0 & & & \\ 0 & & {}^{(n-1)}P & \\ \vdots & & & \end{array}\right) \rightarrow {}^{(n-1)}P$ – (n-1) dim householder with $\underset{\sim}{x}$ = lower (n-1) elements of the 2nd column of $\underset{\sim}{P}_1 \underset{\approx}{A}$.

then $\underset{\approx}{P}_2 \underset{\approx}{P}_1 \underset{\approx}{A} = \begin{pmatrix} k & x & x & \\ 0 & k' & x & \\ \vdots & 0 & x & \cdots \\ \vdots & \vdots & & \end{pmatrix}$

Now, compare $P_{n-1} \cdots P_1 A = R$ with

$$A = QR$$

$$\Rightarrow \quad Q = (P_{n-1} \cdots P_1)^{-1}$$

Q is orthogonal since P's are orthogonal. $\longrightarrow$ $\underline{Q = P_1 \cdots P_{n-1}}$

Note : $P = P^T = P^{-1}$ for householder

So, we have the decomposition.

☆ A maximum # of $\frac{n(n-1)}{2}$ should eliminate them all

---

Finally, we have $\quad r = \frac{n(n-1)}{2}$

$$P_r \cdots P_1 A = R$$

Since P_i are orthogonal matrices,

$$P_i^{-1} = P_i^T$$

$$\boxed{\begin{array}{l} \text{So,} \quad A = QR \\ \text{where} \quad Q = P_1^T P_2^T \cdots P_r^T \end{array}}$$

☆ It can be shown that

$$A^n \to D = \operatorname{diag}\{\lambda_1 \ldots \lambda_n\}$$ if A is sym.

AND, $\lambda_1 < \lambda_2 < \cdots < \lambda_n$

— eigenvalues arranges in increasing order of magnitude!

Shifting & rate of Convergence

It can be shown that for a ~~super~~ off-diagonal diagonal element a_{ij}

$$\text{rate}\{a_{ij}^n \to 0\} \sim \left(\frac{\lambda_i}{\lambda_j}\right)^n \quad i<j$$

☆ while we know that $\lambda_j > \lambda_i$.

If $\lambda_j \simeq \lambda_i$, then the convergence is slow!

☆ To prevent this, we can shift all diagonal values by k_n:

$$\underset{\sim}{A}' = \underset{\sim}{A} - k_n \underset{\sim}{1}$$

① $\det(A') = \det(\underset{\sim}{A} - k_n 1) = \lambda - k_n$

eigenvalues shifted

② The resulting recursive similarity transformations for $\underline{\underline{A}}$ are in the same form:

$$A^{n+1} = Q_n^T A^n Q_n \qquad \left[\begin{array}{l}\text{the same } Q_n \text{ transform} \\ A^n \to A^{n+1} \\ A'^n \to A'^{n+1}\end{array}\right.$$

so that the convergent characteristic is the same but with a different rate:

$$\text{rate}\{a_{ij}'^{n} \to 0\} \sim \left|\frac{\lambda_i - k_n}{\lambda_j - k_n}\right|^n$$

— so that we can adjust the ratio at each step n such that

$$\lambda_i - k_n \sim 0$$

to maximize the rate of convergent!

To see $A^{n+1} = Q_n^T A^n Q_n$

$$\underset{\sim}{A}' = \underset{\sim}{A} - k_n \underset{\sim}{1}$$

$$A_n' = Q_n R_n$$

$$A_{n+1}' = R_n Q_n = Q_n^+ A_n' Q_n$$

$$A_{n+1} - k_n 1 = Q_n^+ A_n Q_n - k_n Q_n^+ 1 Q_n$$

$$= Q_n^+ A_n Q_n - k_n 1$$

$$\Rightarrow \quad A_{n+1} = Q_n^T A_n Q_n$$