Eigensystems

Aiichiro Nakano

Collaboratory for Advanced Computing & Simulations
Department of Computer Science
Department of Physics & Astronomy
Department of Quantitative & Computational Biology
University of Southern California

Email: anakano@usc.edu

- Matrix diagonalization methods in the context of quantum mechanics
- Matrix decompositions
- Vector space: projection & rotation

Eigenvalue Problem

- **Eigenvalue problem in *N*-dimensional vector space**

real symmetric
***N*×*N* matrix**

$$A|n\rangle = \lambda_n |n\rangle$$

***n*-th eigenvector**

$$|n\rangle = x^{(n)} \in R^N$$

***n*-th eigenvalue**

or more explicitly

$$\sum_{j=1}^{N} A_{ij} x_j^{(n)} = \lambda_n x_i^{(n)}$$

***i*-th element of the *n*-th eigenvector**

Orthonormal Basis

- **The basis set** $\{|n\rangle | n = 1, \ldots, N\}$ **can be made orthonormal, *i.e.*,**
  $\langle m|n\rangle = \sum_{i=1}^{N} x_i^{(m)} x_i^{(n)} = \delta_{mn}$

- **Orthogonal matrix:** $\boldsymbol{U} = [\boldsymbol{x}^{(1)}\ \boldsymbol{x}^{(2)} \ldots \boldsymbol{x}^{(N)}]$ **or** $U_{in} \equiv x_i^{(n)}$

  $U^T U = I$ $\quad \because \sum_{i=1}^{N} x_i^{(m)} x_i^{(n)} = \sum_{i=1}^{N} \overbrace{U_{im}}^{U_{mi}^T} U_{in} = (U^T U)_{mn} = \delta_{mn}$

(Proof: orthogonality)

For Hermitian matrix:

$$(A^\dagger)_{ij} = A_{ji}^* = A_{ij}$$

- $\lambda_m \neq \lambda_n$

$$\langle m|A|n\rangle = \lambda_n \langle m|n\rangle$$

$$-)\quad \langle m|A|n\rangle = \lambda_m \langle m|n\rangle$$

$$0 = (\lambda_n - \lambda_m)\langle m|n\rangle$$

$$\langle m|A|n\rangle = \lambda_m \langle m|n\rangle \xleftarrow[\text{(real eigenvalue)}]{\text{complex conjugate}} \langle n|A|m\rangle = \lambda_m \langle n|m\rangle$$

$$\lambda_n \langle n|n\rangle = \langle n|A|n\rangle = \langle n|A^\dagger|n\rangle = \lambda_n^* \langle n|n\rangle$$

$$0 = (\lambda_n - \lambda_n^*)\langle n|n\rangle \Leftrightarrow \lambda_n = \lambda_n^*$$

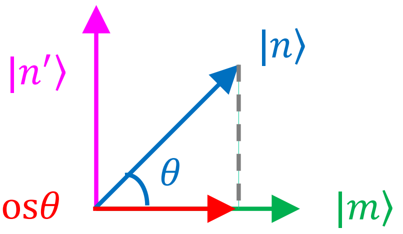
- $\lambda_m = \lambda_n$ **(degenerate): use Gram-Schmidt orthogonalization procedure**

1. **Orthogonal projection:** $|n'\rangle \leftarrow |n\rangle - |m\rangle\langle m|n\rangle = (1 - |m\rangle\langle m|)|n\rangle$

   $$\langle m|n'\rangle = \langle m|n\rangle - \overbrace{\langle m|m\rangle}^{1}\langle m|n\rangle = 0$$

2. **Normalization:** $|n'\rangle \leftarrow |n'\rangle / \langle n'|n'\rangle^{1/2}$

   $$\langle n'|n'\rangle = 1$$



$|m\rangle\langle m|n\rangle = |m\rangle\cos\theta$

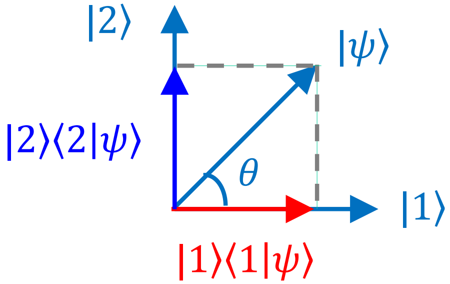
Directional cosine

Completeness

- **Arbitrary N-dimensional vector can be represented as a linear combination of (linearly independent) N vectors**

$$|\psi\rangle = \overbrace{\sum_{n=1}^{N} |n\rangle\langle n|}^{=1}\,\psi\rangle$$

2D example
(just Cartesian coordinates)



***i.e.*, $\sum_{n=1}^{N} |n\rangle\langle n| = 1$ or equivalently $\sum_{n=1}^{N} x_i^{(n)} x_j^{(n)} = \delta_{ij}$**

$$\psi_i = \sum_{n=1}^{N} x_i^{(n)} \sum_{j=1}^{N} x_j^{(n)} \psi_j = \sum_{j=1}^{N} \overbrace{\sum_{n=1}^{N} x_i^{(n)}\, x_j^{(n)}}^{\delta_{ij}} \psi_j$$

- **Orthogonal matrix**

$$U^T U = U U^T = I$$

$$\therefore U^{-1} = U^T$$

$$\delta_{ij} = \sum_{n=1}^{N} x_i^{(n)}\, x_j^{(n)} = \sum_{n=1}^{N} U_{in}\, \overbrace{U_{jn}}^{U_{nj}^T} = (UU^T)_{ij}$$

$\therefore$ Column-aligned eigenvectors, $U = [x^{(1)}\ x^{(2)}\ \ldots\ x^{(N)}]$, can be made an orthogonal matrix

Orthogonal Transformation

$\sum_{i=1}^{N} x_i^{(m)} \times \quad \left(\sum_{j=1}^{N} A_{ij} x_j^{(n)} = \lambda_n x_i^{(n)} \right)$

$$\sum_{i=1}^{N} \sum_{j=1}^{N} \overbrace{x_i^{(m)}}^{U_{mi}^T} A_{ij} \overbrace{x_j^{(n)}}^{U_{jn}} = \lambda_n \sum_{i=1}^{N} x_i^{(m)} x_i^{(n)} = \overbrace{\lambda_n \delta_{mn}}^{\equiv \Lambda_{mn}}$$

orthogonality

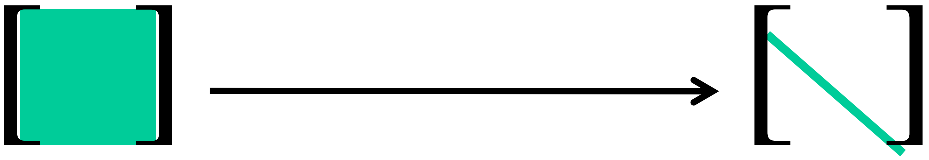
- **Matrix eigenvalue problem = find an orthogonal transformation matrix**

Spectral decomposition

$$\boxed{\begin{gathered} U^T A U = \Lambda \\ \Lambda_{mn} = \lambda_m \delta_{mn} \end{gathered}}$$

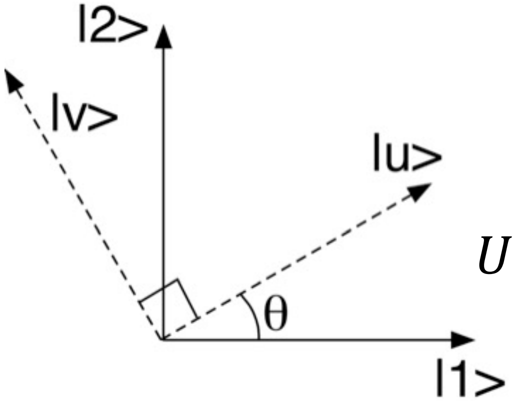
- **Grand strategy:** **Nudge the matrix *A* towards diagonal form by a sequence of orthogonal transformations (successive elimination of off-diagonal elements)**

$$A \rightarrow P_1^T A P_1 \rightarrow \overbrace{P_2^T P_1^T}^{U^T} A \overbrace{P_1 P_2}^{U} \rightarrow \cdots$$
$$U = P_1 P_2 \cdots$$



Rotation

- **General real symmetric 2×2 matrix:** $H = \begin{bmatrix} \varepsilon_1 & \delta \\ \delta & \varepsilon_2 \end{bmatrix}$
- **General orthonormal matrices:** $|u\rangle = \begin{bmatrix} \cos\theta \\ \sin\theta \end{bmatrix} = \cos\theta|1\rangle + \sin\theta|2\rangle; |v\rangle = \begin{bmatrix} -\sin\theta \\ \cos\theta \end{bmatrix}$
- **Eigenvalue solution**



$$U = [u \quad v] = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$$

$$\begin{bmatrix} \lambda-\varepsilon_1 & -\delta \\ -\delta & \lambda-\varepsilon_2 \end{bmatrix} \begin{bmatrix} \cos\theta \\ \sin\theta \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad \det(\lambda I - H) = \begin{vmatrix} \lambda-\varepsilon_1 & -\delta \\ -\delta & \lambda-\varepsilon_2 \end{vmatrix} = (\lambda-\varepsilon_1)(\lambda-\varepsilon_2) - \delta^2 = 0$$

$$\lambda_\pm = \frac{\varepsilon_1 + \varepsilon_2 \pm \sqrt{(\varepsilon_1-\varepsilon_2)^2 + 4\delta^2}}{2}$$

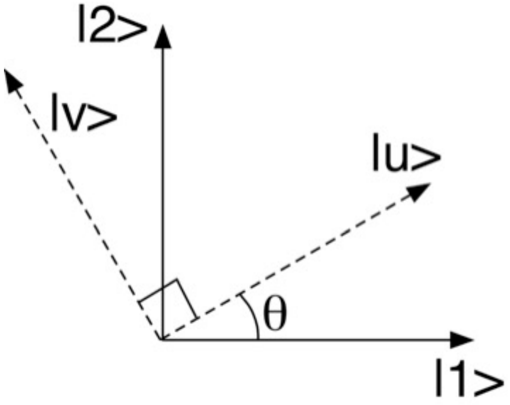
$$\theta = \tan^{-1}\left(\frac{-\varepsilon_1 + \varepsilon_2 + \sqrt{(\varepsilon_1-\varepsilon_2)^2 + 4\delta^2}}{2\delta}\right) \xrightarrow{\delta\to 0} \frac{\delta}{\varepsilon_1 - \varepsilon_2}$$

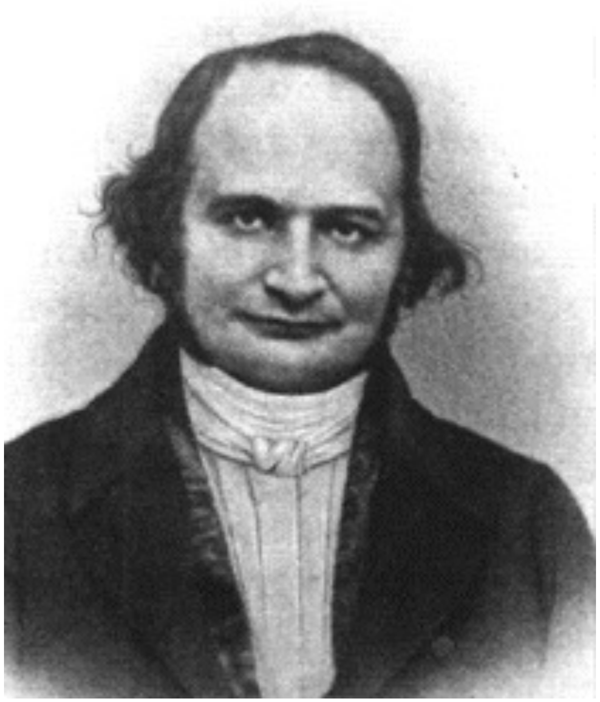
for λ_+ and $\varepsilon_1 > \varepsilon_2$

Jacobi Transformation

- **Successive 2D rotations to eliminate off-diagonal (i, j)–(j, i) pairs**

$$\begin{array}{cc} & \begin{matrix} & i & & j & \end{matrix} \\ \begin{matrix} \\ i \\ \\ j \\ \\ \end{matrix} & \begin{bmatrix} & \vdots & & \vdots & \\ \cdots & * & \cdots & 0 & \cdots \\ & \vdots & & \vdots & \\ \cdots & 0 & \cdots & * & \cdots \\ & \vdots & & \vdots & \end{bmatrix} \end{array}$$

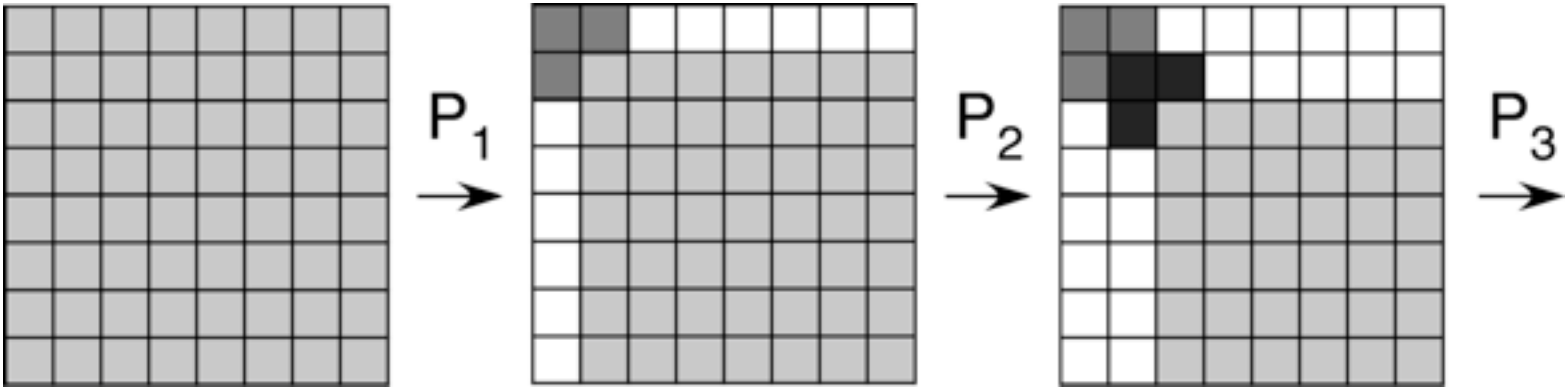




Carl Jacobi
(1804-1851)

Householder Transformation

- Eliminate an entire row (but the first 2 elements) at a time



- The outcome is a tridiagonal matrix

Alston Householder
(1904-1993)

Projection Matrix

- **Let an *N*-dimensional vector** $v\ (\in \Re^N)$ **& the projection matrix**

$$P = I - \frac{2vv^T}{v^Tv} = I - \frac{2|v\rangle\langle v|}{\langle v|v\rangle}$$

then *P* is symmetric & orthonormal, *i.e.*,

$$P^TP = PP^T = I$$

(Proof)

$$P_{ij} = \delta_{ij} - \frac{2v_iv_j}{\sum_{k=1}^{N} v_k^2}$$

← symmetric w.r.t. *i* ↔ *j*

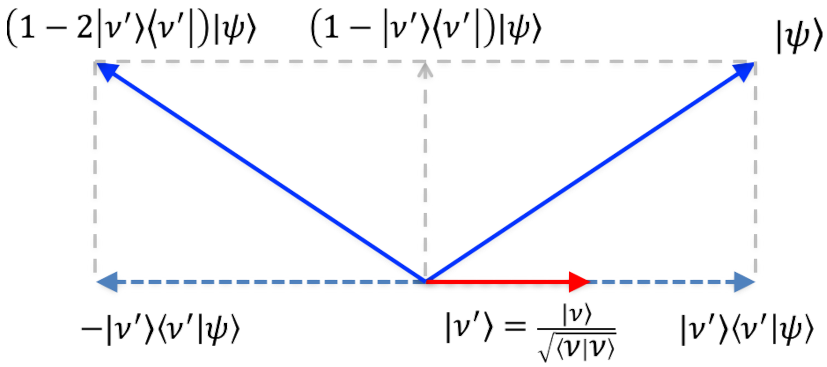
$$PP = \left(I - \frac{2vv^T}{v^Tv}\right)\left(I - \frac{2vv^T}{v^Tv}\right)$$

$$= I - \frac{4vv^T}{v^Tv} + \frac{4vv^Tvv^T}{v^Tvv^Tv}$$

$$= I - \frac{4vv^T}{v^Tv} + \frac{4vv^T}{v^Tv}$$

$$= I$$

Mirror image: reflect twice = do nothing



Householder Matrix

- **For *x* (∈ R^N), let $v = x \mp \|x\|_2 e_1$ where**

  $e_1 = \begin{bmatrix} 1 \\ 0 \\ \vdots \end{bmatrix}$ **& the vector 2-norm is** $\|x\|_2 = \sqrt{x^T x} = \sqrt{\sum_{i=1}^{N} x_i^2}$

  then the Householder matrix below, when multiplied, eliminates all the elements of *x* but one:

$$Px = \left(I - \frac{2vv^T}{v^T v}\right)x = \mp\|x\|_2 e_1$$

(Proof)

$$v^T v = (x^T \pm \|x\|_2 e_1^T)\,(x \pm \|x\|_2 e_1) = \|x\|_2^2 \pm 2\|x\|_2 x_1 + \|x\|_2^2 = 2\|x\|_2(\|x\|_2 \pm x_1)$$

$$\begin{aligned} Px &= x - \frac{2vv^T}{2\|x\|_2(\|x\|_2 \pm x_1)}x \\ &= x - \frac{(x \pm \|x\|_2 e_1)(x^T \pm \|x\|_2 e_1^T)x}{\|x\|_2(\|x\|_2 \pm x_1)} \\ &= x - \frac{(x \pm \|x\|_2 e_1)\|x\|_2(\|x\|_2 \pm x_1)}{\|x\|_2(\|x\|_2 \pm x_1)} \\ &= x - x \mp \|x\|_2 e_1 = \mp\|x\|_2 e_1 \end{aligned}$$

Tridiagonalization

- **Householder matrix can be used for tridiagonalization: Let**

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1N} \\ a_{21} & & & \\ \vdots & & & \\ a_{N1} & & & \end{bmatrix} = \begin{bmatrix} a_{11} & A_{12} = A_{21}^T \\ & \\ A_{21} & A_{22} \end{bmatrix}$$

$$v \quad \left(\in R^{N-1}\right) = A_{21} + \text{sign}\left(a_{21}\right)\left\|A_{21}\right\|_2 e_1$$

then

$$\bar{P}A_{21} \equiv \left(I_{N-1} - \frac{2vv^T}{v^Tv}\right)A_{21} = \text{-sign}\left(a_{21}\right)\left\|A_{21}\right\|_2 e_1 \equiv ke_1$$

$$PAP \equiv \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & & & \\ \vdots & & \bar{P} & \\ 0 & & & \end{bmatrix} \begin{bmatrix} a_{11} & A_{21}^T \\ & \\ A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & & & \\ \vdots & & \bar{P} & \\ 0 & & & \end{bmatrix}$$

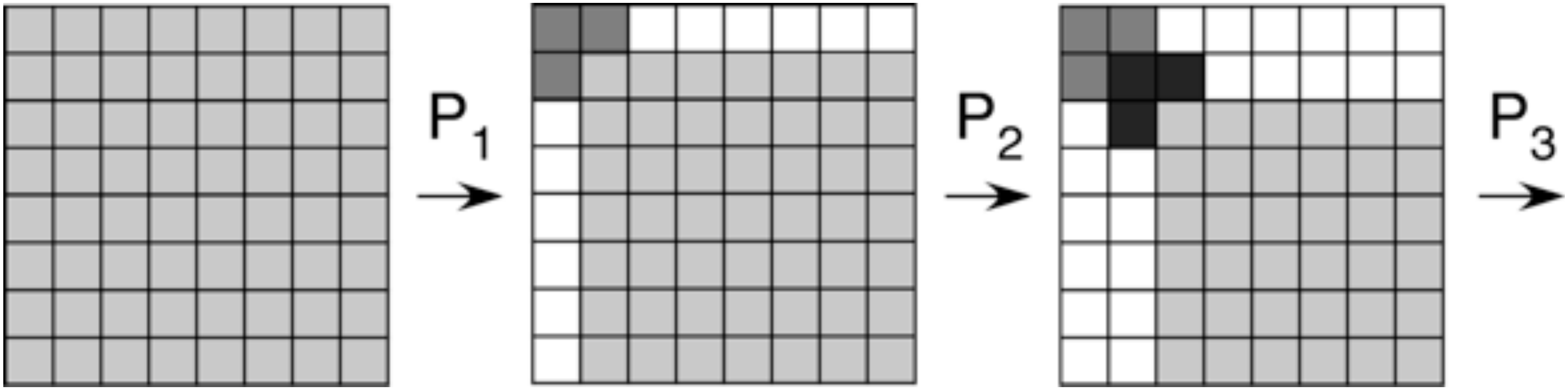
$$\overbrace{\bar{P}}^{N-1\times N-1} \; \overbrace{A_{21}}^{N-1\times 1} = \begin{bmatrix} k \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

$$= \begin{bmatrix} a_{11} & A_{21}^T \\ k & \\ 0 & \bar{P}A_{22} \\ \vdots & \\ 0 & \end{bmatrix} \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & & & \\ \vdots & & \bar{P} & \\ 0 & & & \end{bmatrix} = \begin{bmatrix} a_{11} & k & 0 & \cdots & 0 \\ k & & & & \\ 0 & & & \bar{P}A_{22}\bar{P} & \\ \vdots & & & & \\ 0 & & & & \end{bmatrix}$$

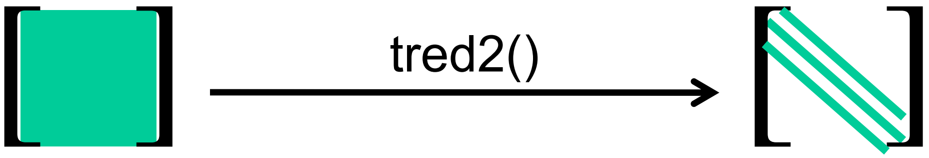
Householder Transformation

- After (N–2) such transformations, all the off-diagonal elements but the diagonal & upper/lower sub-diagonal elements are eliminated

$$PAP \equiv \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & & & \\ \vdots & & \overline{P} & \\ 0 & & & \end{bmatrix}\begin{bmatrix} a_{11} & A_{21}^T \\ & \\ A_{21} & A_{22} \end{bmatrix}\begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & & & \\ \vdots & & \overline{P} & \\ 0 & & & \end{bmatrix} = \begin{bmatrix} a_{11} & k & 0 & \cdots & 0 \\ k & & & & \\ 0 & & \overline{P}A_{22}\overline{P} & & \\ \vdots & & & & \\ 0 & & & & \end{bmatrix}$$



- The outcome is a tridiagonal matrix (done in tred2() in *Numerical Recipes*)



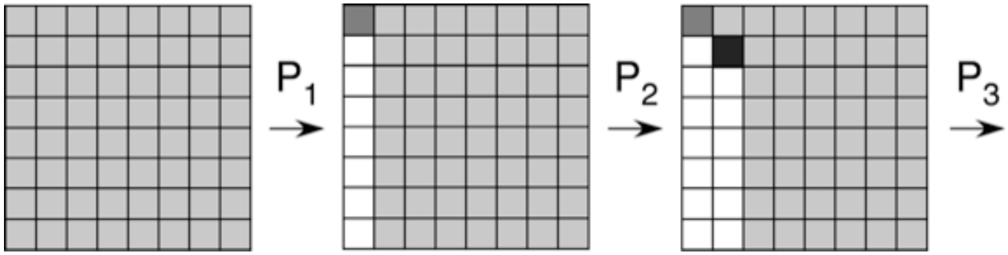
QR Decomposition

- Used for the diagonalization of a tridiagonal matrix
- Let $A = QR$, where Q is orthogonal & R is upper-triangular, $R_{ij} = 0$ for $i > j$
- QR decomposition by Householder transformation

$$A = \begin{bmatrix} a_{11} & \\ \vdots & \\ a_{N1} & \end{bmatrix} = \begin{bmatrix} A_1 & A_2 \end{bmatrix} \qquad v \quad \left(\in R^N\right) = A_1 + \mathrm{sign}(a_{11}) \|A_1\|_2 e_1$$

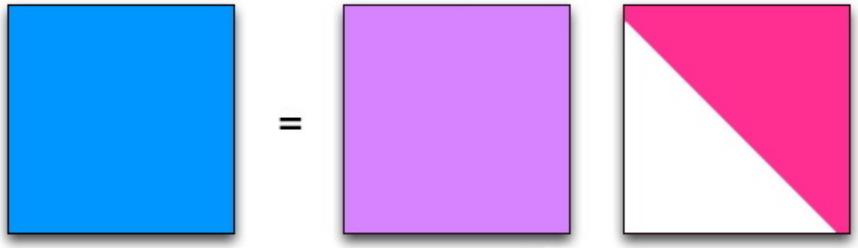
$$PA_1 \equiv \left(I_N - \frac{2vv^T}{v^T v} \right) A_1 = \text{-}\mathrm{sign}(a_{11}) \|A_1\|_2 e_1 \equiv ke_1$$

$$PA = \begin{bmatrix} PA_1 & PA_2 \end{bmatrix} = \begin{bmatrix} k & \\ 0 & \\ \vdots & PA_2 \\ 0 & \end{bmatrix}$$



- After (N–1) transformations, the matrix is upper-triangular

$$P_{N-1} \cdots P_2 P_1 A = R$$
$$A = P_1^{-1} P_2^{-1} \cdots P_{N-1}^{-1} R \equiv QR$$



Orthogonal Transformation by QR

$$A = QR \qquad A' = RQ$$

$$\Big\Downarrow \; R = Q^{-1}A = Q^{T}A$$

$$A \to A' = Q^{T}AQ$$

(QR algorithm)

$$\begin{cases} 1.\ Q_sR_s \leftarrow A_s \\ 2.\ A_{s+1} \leftarrow R_sQ_s \end{cases} \qquad s = 1,2,\dots$$

Top 10 algorithms in history
IEEE CiSE, Jan/Feb ('00)

- Metropolis Algorithm for Monte Carlo
- Simplex Method for Linear Programming
- Krylov Subspace Iteration Methods
- ◉ The Decompositional Approach to Matrix Computations
- The Fortran Optimizing Compiler
- ◉ QR Algorithm for Computing Eigenvalues
- Quicksort Algorithm for Sorting
- Fast Fourier Transform
- Integer Relation Detection
- Fast Multipole Method

(Theorem)

1. **$\lim_{s\to\infty} A_s$ is upper-triangular**
2. **The eigenvalues appear on its diagonal**

- **tqli() in *Numerical Recipes* uses QL algorithm instead to obtain lower-triangular matrix**
- **Fast — $O(N)$ operations per iteration — for a tridiagonal matrix**
- **tqli() diagonalizes a tridiagonal matrix by a sequence of rotations to eliminate subdiagonal elements, in addition to eigenvalue-shift to accelerate the convergence**