Finding Eigenvalues

Jason Slemons Claus Kirchner
University of Washington
Department of Applied Mathematics
29 November 2005

Motivation

Example 1: ODE

Consider the system of ordinary differential equations

$$y' = Ay \tag{1}$$

Let $A \in \mathbb{R}^{n \times n}$ be regular.
One solution to (1) is given by

$$y(t) = e^{\lambda \cdot t} x \tag{2}$$

where λ is an eigenvalue and x is the corresponding eigenvector.

Motivation

Example 2: Factor analysis

Factor analysis is a method used in a variety of fields, e.g. in statistics and psychology.
In statistics, the correlation matrix of n random variables is considered.
Eigenvectors correspond to the so-called factors. The aim is to find the factors that influence the variance significantly.
Eigenvalues serve as a measure for the impact of a factor on the overall variance.

Example 3: Fiber Optic Design

SIMAX, Volume 28 Issue 1 pages 105 to 117, © 2006 SIAM

Overview

- Definitions
- Properties of Matrices
- Decompositions
- Algorithms
  - QR iteration
  - Divide and Conquer
  - Jacobi
  - other things I think are neato.
- dqds
  - Symmetric Positive Definite case
  - Problems with the Symmetric Indefinite and non Symmetric cases

Definitions

- eigenvalue λ eigenvector v in $Av = \lambda v$
- Symmetric Matrix $A = A^T$
- Positive Definite: $v^T A v > 0 \; \forall v \neq 0$
- Normal: $A^T A = AA^T$, $\Leftrightarrow$ A is unitarily diagonalizable.
- Unitary: U So that $U^{-1} = U^T$.

- Tridiagonal:

$$T = \begin{pmatrix} b_1 & a_1 & & & & \\ c_1 & b_2 & \ddots & & & \\ & \ddots & \ddots & a_i & & \\ & & c_i & b_{i+1} & \ddots & \\ & & & \ddots & \ddots & a_{n-1} \\ & & & & c_{n-1} & b_n \end{pmatrix} \tag{3}$$

- Upper Hessenberg:

$$A = \begin{pmatrix} b_1 & a_1 & * & * & * & * \\ c_1 & b_2 & \ddots & * & * & * \\ & \ddots & \ddots & a_i & * & * \\ & & c_i & b_{i+1} & \ddots & * \\ & & & \ddots & \ddots & a_{n-1} \\ & & & & c_{n-1} & b_n \end{pmatrix} \tag{4}$$

Properties

- Triangular matrices
- Diagonal matrices
- Symmetric matrices(λ is real)
- Symmetric Positive Definite
  - $\lambda > 0$
  - $A = LL^T$ (necc. $a_{i,i} > 0$)
  - nicest kind of matrix
- Normal matrices $\Leftrightarrow$ A is unitarily diagonalizable.

Decompositions

- Similarity $A = SBS^{-1}$(same eigenvalues)
- Something else: $A = SBS^{T}$(Sylvesters Theorm)
- Reduction to Tridiagonal form(symmetric A) $A = UTU^{T}$
- Reduction to Upper Hessenberg form(non symmetric A) $A = UHU^{T}$

Decompositions

- Jordan Decomposition $A = SJS^{-1}$. J is block diagonal,

$$J_b = \begin{pmatrix} \lambda & 1 & & \\ & \ddots & 1 & \\ & & \lambda & \end{pmatrix} \tag{5}$$

- Schur Decomposition $A = UBU^*$ where B is upper triangular(non symmetric) or diagonal(symmetric).

- Singular Value Decomposition $A = U\Sigma V^T$. U, V unitary, Σ has singular values($\sigma = |\lambda|$ for symmetric A)

Eigenvalues of a Matrix A

- State problem. Any ideas? exact solvers.

Eigenvalues of a Matrix A

- State problem. Any ideas? exact solvers.
- Power Method:

$$
\begin{aligned}
u_{k+1} &= A u_k && (6)\\
u_{k+1} &= \frac{u_{k+1}}{||u_{k+1}||} && (7)\\
\lambda_{k+1} &= u_{k+1}^T A u_{k+1} && (8)\\
k &= k+1 && (9)
\end{aligned}
$$

What if we use A^{-1} instead of A?

Eigenvalues of a Matrix A

We should probably call this the inverse Power Method. or Jasons Method. either one.

- power method using $(A - \lambda I)^{-1}$ for different λ.
- $(A - kI)^{-1}$ has evals $\frac{1}{\lambda_1 - k}, \frac{1}{\lambda_2 - k}, \frac{1}{\lambda_3 - k}, \ldots, \frac{1}{\lambda_n - k}$. This gives the evalue closest to k.
- Works well if you have good guesses already from another solver. Finds evectors given evalues.

YAM- yet another method.

- QR iteration. Its okay.

$$
\begin{aligned}
A_i &= Q_i R_i && (10)\\
A_{i+1} &= R_i Q_i && (11)\\
i &= i+1 && (12)
\end{aligned}
$$

- notice that $A_{i+1} = R_i Q_i = Q_i^T (Q_i R_i) Q_i = Q_i^T (A_i) Q_i$. This converges. Can add shifts to help convergence.

- $O(n^3)$ flops for one iteration? even if we just had one iteration per eigenvalue this is $O(n^4)$. reduce to Upper Hessenberg ($\frac{10}{3}n^3 + O(n^2)$) and then one iteration of QR is only $6n^2 + O(n)$ work.

The Symmetric Eigenproblem

Definition of Problem

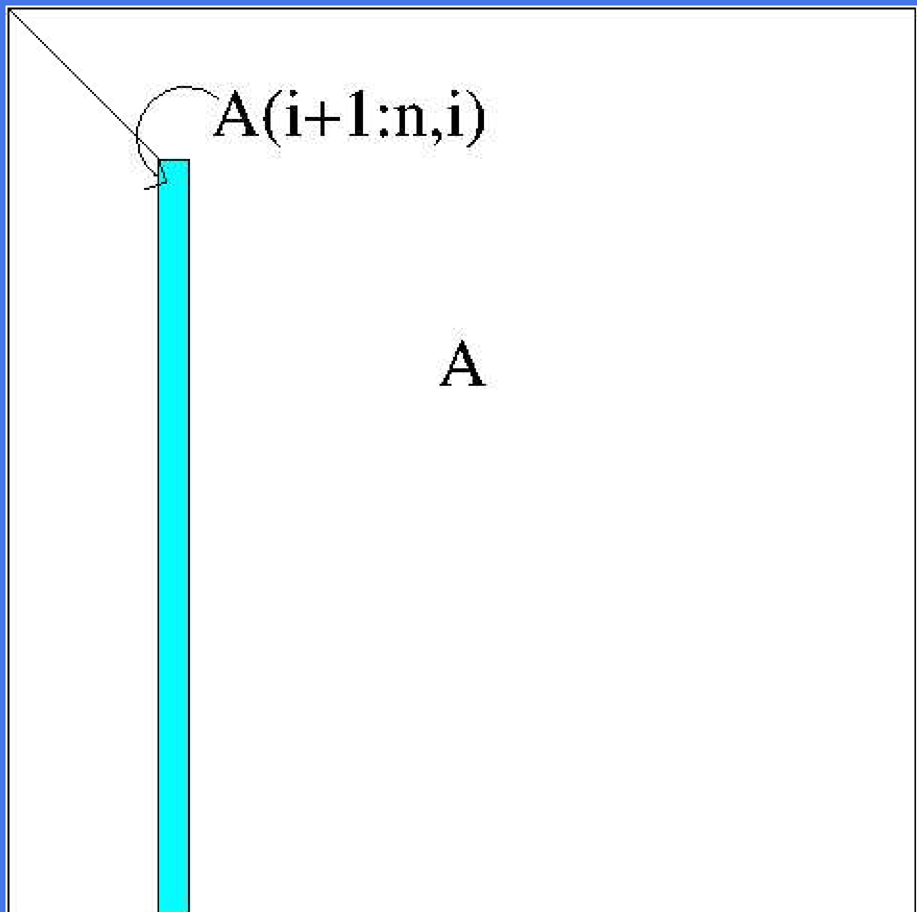
- Different kinds(QR iteration, bisection, divide and conquer, dqds, holy grail, etc.)
- What they do
  - reduce to tridiagonal system (upper hessenberg form if not symmetric)
  - find e-vectors and values of the tridiagonal problem(upper hessenberg for non symmetric)
  - back transform e-vectors

Serial Tridiagonalization

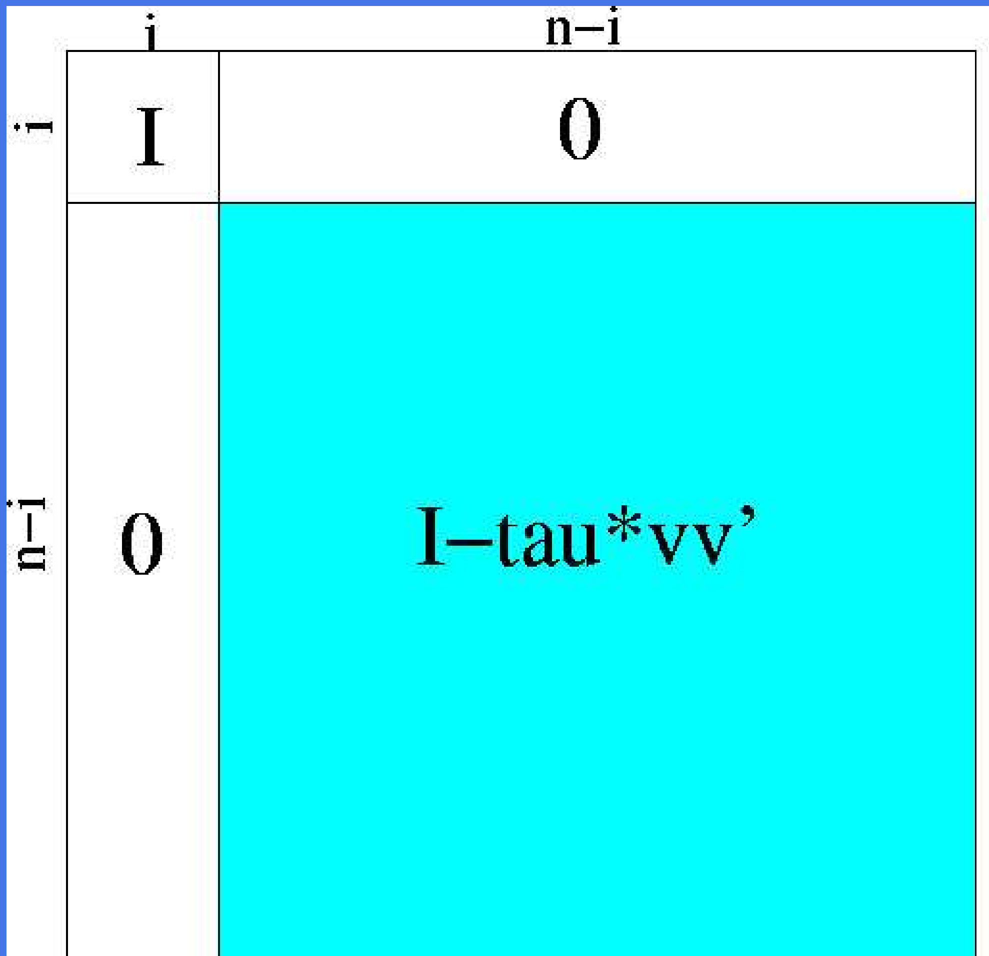
- Given a symmetric matrix A
- A Householder matrix is $H_1 = I - \tau v v'$ for a vector v and scalar τ chosen so that H is orthonormal and so that $H^t = H^{-1}$. $\tau = \frac{2}{vv'}$
- choosing v cleverly yeilds $H_1 A H_1^T$, a matrix with a first row and column thats zero except for an entry on the diagonal and super/sub-diagonal.
- Unsymmetric A?

Serial Tridiagonalization

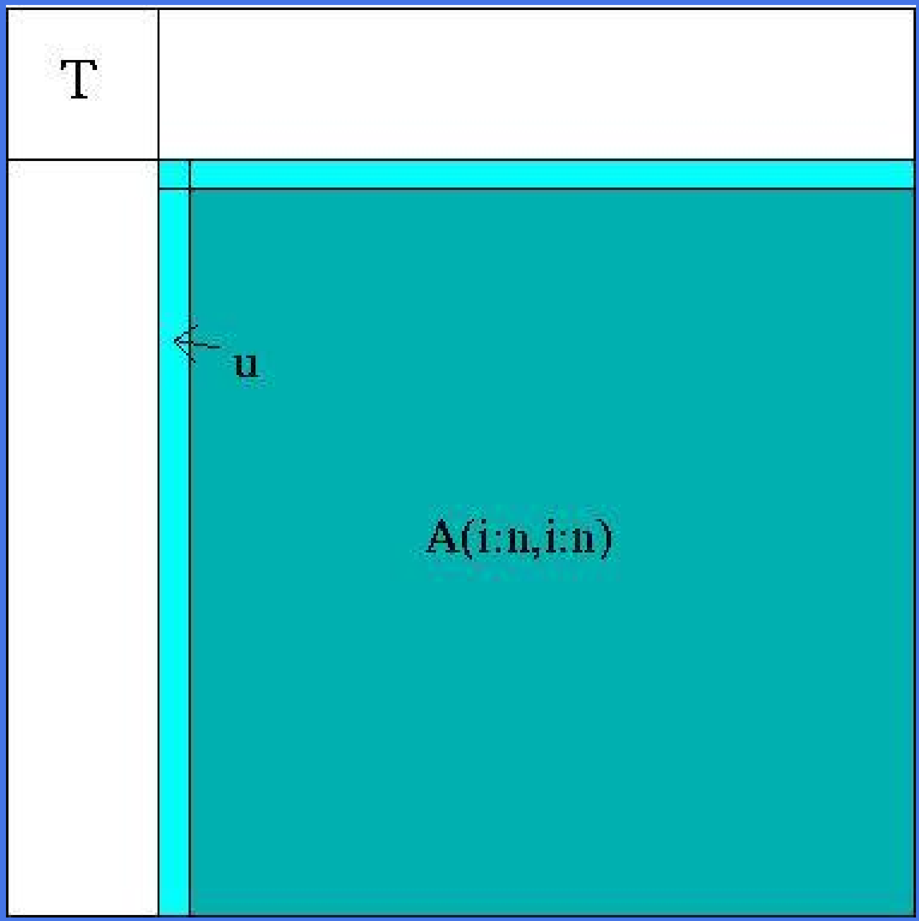
- Tridiagonalize lower $n-1$ by $n-1$ block of H_1AH_1.
- Eliminate the next row and column by another householder matrix; H_2
- Repeat: across and down the matrix. Left with $H_{n-2}...H_1A(H_{n-2}...H_1)^T = T$



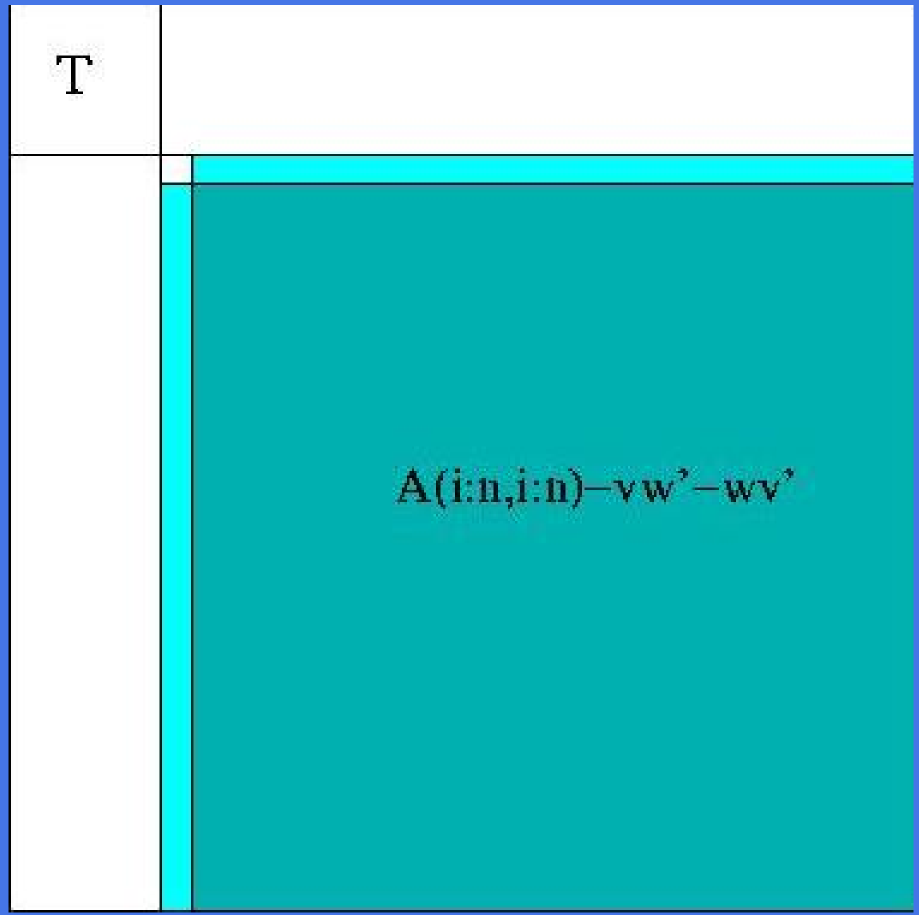
Set $u = A_{i+1:n,i}$ then $y = u \pm ||u||e_1$ and $v = y/y(1)$. $\tau = \frac{2}{vv'}$.



From $I - \tau v v'$ in the lower corner and I^i in the upper corner, form H.



A is above. By choosing H we know HAH's ith column(labeled u) will have only zeros after the sub-diagonal. What will the rest of the lower $n-i$ corner of A, $A_{i:n,i:n}$, be after conjugation? We need to find $(HAH')_{i:n,i:n}$'.



Form $z = \tau A_{i:n,i:n} v$ and then $w = z - \frac{\tau z' v}{2} v$. The updated $n-i$ block is of course, the mostly eliminated first column and the new $A_{i:n,i:n}$ which is related by $A_{i:n,i:n} = A_{i:n,i:n} - vw' - wv'$.

Serial Tridiagonalization

for $i = 1 : n - 1$

1) Choose $u = A(i + 1 : n, i)$

2) $y = u \pm ||u||e_1$

3) $v = y/y(1)$

4) Calculate $\tau = \frac{2}{vv'}$

IMPLICIT) Then form H from $I - \tau vv'$by adding an identity in the upper left corner of size i. find HAH'

5) $z = \tau * A_{i:n,i:n}v$ and $w = z - \tau/2(z'v)v$. $A_{i:n,i:n}$ is the lower right block of A thats left.

6) Update the trailing matrix by $A_{i:n,i:n} - vw' - wv'$.

End when you have eliminated everything below the sub-diagonal. Whats left is a tridiagonal matrix.

The Symmetric Tridiagonal Eigenproblem

- QR iteration with shifts takes $O(n^2)$ w/o eigenvectors. $O(n^3)$ with them. Fastest out there for $n \leq 25$

- Divide and Conquer. Theoretically $O(n^3)$ but in practice $O(n^{2.3})$ with eigenvectors. best for $n \geq 25$

- Holy Grail. $O(n^2)$ for evalues and e vectors. Pretty awesome.

The Symmetric Tridiagonal Eigenproblem

- Jacobi its accurate, but usually slow. Also old.
- Bisection. start with an interval. Takes $O(kn)$ where k is the number of eigenvalues. no eigenvectors.
- Inverse Iteration. Works with Bisection. Eigenvectors arent great if the values are clustered. with shift of a_{nn} and $u_0 = [0...0, 1]^T$ this is the same as QR iteration(with the Raleigh shift).
- dqds. Only for positive definite symmetric matrices. Its Awesome. differential quotient difference algorithm with shifts. lame name.

Givens Rotations

Given a symmetric matrix A, hit it on the left and right by Givens rotations:

$$J_i = \begin{pmatrix} 1 & & & & & & \\ & \ddots & & & & & \\ & & cos(\theta) & & -sin(\theta) & & \\ & & & \ddots & & & \\ & & sin(\theta) & & cos(\theta) & & \\ & & & & & \ddots & \\ & & & & & & 1 \end{pmatrix} \tag{13}$$

These are more accurate than Householder matrices but they are often slower to converge.

Jacobi's Method

These matrices zero out the off diagonal entry, they are unitary and they produce a series of orthogonally similar matrices A_i

$$A_i = J_{i-1}^T A_{i-1} J_{i-1} \tag{14}$$

$$A_i = J_{i-1}^T J_{i-2}^T A_{i-2} J_{i-2} J_{i-1} \tag{15}$$

$$A_i = J_{i-1}^T \ldots J_0^T A J_0 \ldots J_{i-1} \tag{16}$$

$$A_i = J^T A J \tag{17}$$

- sweep strategies abound(which i,j to minimize) and influence convergence.
- each A_i is closer to diagonal form than the last.

Divide and Conquer

We split the matrix into T_1 and T_2 the upper and lower blocks before and after *:

$$T = \begin{pmatrix} b_1 & c_1 & & & & & \\ c_1 & b_2 & \ddots & & & & \\ & \ddots & \ddots & * & & & \\ & & * & b_{i+1} & \ddots & & \\ & & & \ddots & \ddots & c_{n-1} \\ & & & & c_{n-1} & b_n \end{pmatrix} \tag{18}$$

- Where $* = c_i$ so Rewriting we have $T = T_1 \oplus T_2 + c_i v v^T$ for $v = e_i + e_{i-1}$.
- Eigenvalues of T_1 and T_2 are found recursively so $T_1 \oplus T_2 = QDQ^T$.
- using $det(I + xy^T) = 1 + y^T x$ and $det(D + c_i uu^T - \lambda I)$ we can calculate the eigenvales of T.

dqds

- get symmetric positive definite tridiagonal via the above
- bidiagonal matrix B and find $BB^T = T$:

$i = 0$

repeat

  Choose shift τ_i smaller than the smallest eigenvalue of T_i.

  Computer Cholesky factorization of $T_i - \tau_i I = B_i^T B_i$

  $T_{i+1} = B_i B_i^T + \tau_i I$

  $i = i + 1$

until convergence

dqds with Tridiagonal

- Start with Tridiagonal T, get the new $\hat{T}$, by rewriting the above as:

$T - \tau I = LU$
$UL - \tau I = \hat{L}\hat{U} = \hat{T}$
We can assume without loss that U_i has a diagonal of $u_1, ...u_n$ and a superdiagonal of 1's. L_i has subdiagonal $l_1, ...l_n$ and a diagonal of 1's.

$$L_i = \begin{pmatrix} 1 & & & & & \\ l_1 & \ddots & & & & \\ & \ddots & \ddots & & & \\ & & l_i & 1 & & \\ & & & \ddots & \ddots & \\ & & & & l_{n-1} & 1 \end{pmatrix} \tag{19}$$

dqds with Tridiagonal

$$U_i = \begin{pmatrix} u_1 & 1 & & & & \\ & u_2 & \ddots & & & \\ & & \ddots & 1 & & \\ & & & u_i & \ddots & \\ & & & & \ddots & 1 \\ & & & & & u_n \end{pmatrix} \tag{20}$$

$$L_iU_i = \begin{pmatrix} u_1 & 1 & & & & \\ l_1u_1 & u_2 + l_1 & \ddots & & & \\ & \ddots & \ddots & 1 & & \\ & & l_iu_i & u_i + l_{i-1} & \ddots & \\ & & & \ddots & \ddots & 1 \\ & & & & l_{n-1}u_{n-1} & u_n + l_{n-1} \end{pmatrix} \tag{21}$$

dqds

$UL - \tau I = \hat{L}\hat{U}$ Written out element by element this is:

$$\hat{u_1} = u_1 + l_1 - \tau \tag{22}$$

$$\textbf{for } i = 1, n-1 \tag{23}$$

$$\hat{l_i} = \frac{l_i u_{i+1}}{\hat{u_i}} \tag{24}$$

$$\hat{u_{i+1}} = l_{i+1} + u_{i+1} - \tau - \hat{l_i} \tag{25}$$

$$\textbf{endfor} \tag{26}$$

dqds

rewiting $d_i = u_{i+1} - \hat{l}_i - \tau_i$ we get dqds:

$$
\begin{aligned}
d_1 &= u_1 - \tau_1 && (27)\\
\textbf{for } i &= 1, n-1 && (28)\\
\hat{u}_i &= l_i + d_i && (29)\\
\hat{l}_i &= \frac{l_i u_{i+1}}{\hat{u}_i} && (30)\\
d_i &= d_i\left(\frac{u_{i+1}}{\hat{u}_i}\right) - \tau_i && (31)\\
\textbf{endfor} & && (32)
\end{aligned}
$$

dqds

- d gets rid of subtractions.
- with correct shift and positive T all the d's are quantities are positive.
- amazing relative accuracy because basic operations are accurate.($6n * \epsilon$ normally its just O($K(T) * \epsilon$)
- can find evalues down to 10^{-309} when QR iteration or D & C just gives 10^{-16}.

Future Work

- find out which methods give accurate evals.

- find out which matrices give accurate evals(indefinte case leads to cancellation, jordan form is bad too).

- find out which form to put our matrix in(LL^T vs just T, or Neville elimination)

NEXT TALK: Accuracy and Stability- a place in your mind.