Shifted QR Method (not from the book)

Shifted method...

function QR-ITERATION($A \in \mathbb{R}^{n \times n}$)
  for $k \leftarrow 1, 2, 3, \ldots$
    $Q, R \leftarrow$ QR-FACTORIZE($A - \alpha I$)
    $A \leftarrow RQ + \alpha I$
  return diag(R)

weird idea to mult QR in reverse order...

Last time we ended with the QR method for finding all the eigenvalues at one and noted that if a real matrix had complex conjugate pairs of eigenvalues that those correspond to 2x2 blocks on the diagonal of A.

After running the algorithm for $k \to \infty$

$$A = \begin{bmatrix} \lambda_1 & * & * & * & * \\ 0 & \lambda_2 & * & * & * \\ 0 & 0 & \multicolumn{2}{c}{\boxed{B_1}} & * \\ 0 & 0 & & & * \\ 0 & 0 & 0 & 0 & \lambda_5 \end{bmatrix}$$

2x2 block corresponding to conjugate eigenvalue pair of the form

$$\begin{cases} \lambda_3 = u + iv \\ \lambda_4 = u - iv \end{cases}$$

Let's try another computational example to illustrate how this problem can be solved by using shifts.

Note that in addition to breaking the symmetries that occur with complex conjugate paris, shifts can also be used to speed convergence of the QR algorithm. In this example, we don't worry much about the speed of convergence, only making the 2x2 blocks go away...

```
julia> A=rand(5,5) .- 0.5
5×5 Matrix{Float64}:
 -0.218795   -0.126796  -0.120855   -0.145837  -0.237225
 -0.0994244  -0.485595   0.153475    0.349418  -0.15137
  0.175467   -0.487663   0.0700245  -0.483441  -0.189984
 -0.298285   -0.134561   0.188673   -0.229446   0.282893
  0.361462    0.399642   0.149389    0.449996   0.26109
```

← random matrix

Check that there is at least one complex conjugate eigenvalue pair for the random matrix.

```
julia> eigvals(A)
5-element Vector{ComplexF64}:
 -0.43405217323023815 - 0.3709297587200085im
 -0.43405217323023815 + 0.3709297587200085im
 -0.12472283350143709 + 0.0im
   0.1950523665351203 - 0.34268861341003043im
   0.1950523665351203 + 0.34268861341003043im
```

one pair

another pair

We got lucky. This matrix actually has two complex conjugate eigenvalue pairs. If you matrix had all real eigenvalues, then create another until you get one with at least one complex conjugate eigenvalue pair.

Now we try the QR method, just like last Tuesday...

```
julia> AA=copy(A)
       for k=1:10000
           z=qr(AA)
           AA=z.R*z.Q
       end
```

The results are

```
julia> AA
5×5 Matrix{Float64}:
 -0.534326    0.453657   0.20071    -0.240526    0.242135
 -0.325452   -0.333778   0.417407   -0.403965    0.463689
  1.5e-323   -1.0e-323   0.0944327  -0.488672   -0.199359
 -1.0e-323    0.0        0.261034    0.295672   -0.309777
  0.0         0.0        0.0         0.0        -0.124723
```

?

Since there are two 2x2 blocks corresponding to the two complex conjugate eigenvalue pairs, it's not even clear which block corresponds to which pair

```
julia> eigvals(A)
5-element Vector{ComplexF64}:
 -0.4340521732302815 - 0.3709297587200085im
 -0.4340521732302815 + 0.3709297587200085im
 -0.12472283350143709 + 0.0im
   0.195052366535120 3 - 0.3426886134100304 3im
   0.195052366535120 3 + 0.3426886134100304 3im
```

only one eigenvalue was actually found

We now try the shifted algorithm, with a shift given by an imaginary number to break the complex conjugate symmetry in the original matrix, and hopefully lead to a result where all the eigenvalues explicitly appear on the diagonal...

Try shifted QR...

```
julia> AA=copy(A)
       for k=1:10000
           z=qr(AA-0.5im*I)
           AA=z.R*z.Q+0.5im*I
       end
```

shifting by $\frac{i}{2}$

other values would also work, and like with the power method, some choices will result in faster convergence than others.

upper triangular matrix. The 2x2 blocks are gone.

```
julia> AA
5×5 Matrix{ComplexF64}:
 -0.434052-0.37093im    -0.369957+0.128429im   0.212175-0.268528im    -0.0821971+0.317428im  -0.255514+0.104312im
 -7.4e-323-2.5e-323im    0.195052-0.342689im  -0.265947-0.250651im     0.190701-0.158337im    0.204166-0.0150239im
       0.0+1.0e-323im   -5.0e-324+5.0e-324im  -0.124723+8.88178e-16im  -0.397006+0.133362im  -0.346921-0.0576199im
       0.0+0.0im              0.0+0.0im             0.0-5.0e-323im    -0.434052+0.37093im    0.0504858-0.0342337im
       0.0+0.0im              0.0+0.0im        5.0e-324+5.0e-324im          0.0+0.0im         0.195052+0.342689im
```

eigenvalues on the diagonal...

```
julia> eigvals(A)
5-element Vector{ComplexF64}:
 -0.43405217323023815 - 0.3709297587200085im
 -0.43405217323023815 + 0.3709297587200085im
 -0.1247228335014709 + 0.0im
  0.1950523665351203 - 0.34268861341003043im
  0.1950523665351203 + 0.34268861341003043im
```

The same eigenvalues as found by Julia...

These marks check that all the expected eigenvalues from the built-in Julia subroutine agree with the diagonal terms.

The lecture on Tuesday ended with the start of an algorithm for computing the 2 matrix norm...

Continue finding... $\|A\|_2$

Recall on Tuesday that we had

Find $\|A\|_2 = \max\{\|Ax\|_2 : \|x\|_2 = 1\}$

$$= \sqrt{\max\{\|Ax\|_2^2 : \|x\|_2 = 1\}}$$

$$\|Ax\|_2^2 = Ax \cdot Ax = (Ax)^T Ax = x^T \underbrace{A^T A}_{B} x$$

$B = A^T A$

Note B is symmetric. } hypothesis for the spectral theorem...

B is positive (semi) definite...

From two weeks ago note that the eigenvector-eigenvalue problem $A^T A x = \lambda x$

is related to the critical points of the constrained optimization

$$\mathcal{L} = \|Ax\|^2 - \lambda(\underbrace{\|x\|^2 - 1}_{\text{constraint...}})$$ ← Lagrange mult formulation

critical points given

$$\nabla_x \mathcal{L} = 2A^T Ax - 2\lambda x = 0$$

gives $A^T A x = \lambda x$ eigenvalue problem

This is how to maximize $\|Ax\|^2$ subject to $\|x\|^2 = 1$.

For this reason, it is not surprising that finding the maximum in

$$\max\{\|Ax\|_2^2 : \|x\|_2 = 1\}$$

is related to the eigenvalue-eigenvector problem.

I find our book very nice for making the connection between Lagrange multipliers, optimization and eigenvalues.

Again from last time...

$$\|Ax\|_2^2 = Ax \cdot Ax = (Ax)^T Ax = x^T \underbrace{A^T A}_{B} x$$

Let's solve $Bx = \lambda x$ to find eigenvalues an eigenvectors...

B is symmetric positive semidefinite...

Spectral theorem: There exists an orthonormal basis made out of eigenvectors of B.

The spectral theorem is stated in our book without explanation.

Let's now spend a little more time understanding where this theorem comes from and how to understand it.

We already know that eigenvectors of a symmetric or Hermitian matrix which correspond to different eigenvalues are orthogonal.

What about linearly independent eigenvectors that correspond to the exact same eigenvalue?

They might not be orthogonal...but if they are not, then it is possible to use Gram-Schmidt orthogonalization to construct two different eigenvectors which span the same space and are othogonal.

Example: Suppose x_1 and x_2 are independent with

$$Bx_1 = 4x_1 \quad \text{and} \quad Bx_2 = 4x_2$$

but $x_1 \cdot x_2 \neq 0$

$$A = \begin{bmatrix} x_1 & x_2 \end{bmatrix} \qquad A = QR = \begin{bmatrix} q_1 & q_2 \end{bmatrix} \begin{bmatrix} \alpha & \beta \\ 0 & \gamma \end{bmatrix}$$

$$A = \begin{bmatrix} \alpha q_1 & \beta q_1 + \gamma q_2 \end{bmatrix}$$

Thus...

$$x_1 = \alpha q_1 \qquad x_2 = \beta q_1 + \gamma q_2$$

$$Q = AR^{-1} = \begin{bmatrix} x_1 & x_2 \end{bmatrix} \begin{bmatrix} a & b \\ 0 & c \end{bmatrix}$$

Thus

$$q_1 = a x_1 \qquad q_2 = b x_1 + c x_2$$

$$Aq_1 = A a x_1 = a A x_1 = a 4 x_1 = 4 a x_1 = 4 q_1$$

$$Aq_1 = 4q_1$$

$$Aq_2 = A(bx_1 + cx_2) = bAx_1 + cAx_2 =$$

$$= b4x_1 + c4x_2 = 4(bx_1 + cx_2) = 4q_2$$

$$Aq_2 = 4q_2$$

In summary...

① If B is symmetric or Hermitian in the complex case then the eigenvalues are real.

② If B is positive semidefinite then the eigenvalues are non-negative. (that is $\lambda \geq 0$).

③ If B is symmetric or Hermitian in the complex case and if x_1 and x_2 are eigenvectors corresponding to different eigenvalues λ_1 and λ_2 such that $\lambda_1 \neq \lambda_2$ Then $x_1 \cdot x_2 = 0$.

④ If x_1 and x_2 correspond to the same eigenvalue but are only linearly independent they can be made orthogonal using Gram-Schmidt.

⑤ If B is symmetric there are n independent eigenvector (deflation argument – skip this for now).

proof

The above 5 facts (of which we verified 4) form a sketch of the proof for the spectral theorem...

Spectral theorem: There exists an orthonormal basis made out of eigenvectors of B.

We now use the spectral basis given by the spectral theorem to show how to compute the matrix 2 norm...

Let x_k be a basis of $\underbrace{\text{eigenvectors}}^{\text{orthonormal}}$ of B

then any $x \in \mathbb{R}^n$ can be written as

$$x = c_1 x_1 + c_2 x_2 + \cdots + c_n x_n$$

$$\|x\|^2 = x \cdot x =$$

$$= (c_1 x_1 + c_2 x_2 + \cdots + c_n x_n) \cdot (c_1 x_1 + c_2 x_2 + \cdots + c_n x_n)$$

$$= c_1^2 + c_2^2 + \cdots + c_n^2$$

$$Bx = B c_1 x_1 + B c_2 x_2 + \cdots + B c_n x_n$$

There was an error in the notes here which are now fixed in pink

$$= c_1 \lambda_1 x_1 + c_2 \lambda_2 x_2 + \cdots + c_n \lambda_n x_n$$

$$\|Ax\|^2 = Ax \cdot Ax = x^T A^T A x = x^T B x$$

$$= (c_1 x_1 + c_2 x_2 + \cdots + c_n x_n)^T (c_1 \lambda_1 x_1 + c_2 \lambda_2 x_2 + \cdots + c_2 \lambda_n x_n)$$

cross terms vanish because orthonormal...

$$c_1^2 \lambda_1 + c_2^2 \lambda_2 + \cdots + c_n^2 \lambda_n$$

Therefore, by definition...

$$\|A\|_2^2 = \max\left\{ \|Ax\|_2^2 : \|x\|_2^2 = 1 \right\}$$

$$= \max\left\{ \underbrace{\sum_{k=1}^{n} c_k^2 \lambda_k}_{\text{weighted avereage of the } \lambda_k.} : \sum_{k=1}^{n} c_k^2 = 1 \right\}$$

$$= \max\left\{ \lambda_k : k = 1, \dots, n \right\}$$

Remember that the λ_k here are the eigenvalues of the matrix B.

Thus...

note $\sqrt{\lambda_k}$ makes sense since $\lambda_k \geq 0$

$$\|A\|_2 = \sqrt{\max\left\{ \lambda_k : k = 1, \dots, n \right\}} = \max\left\{ \sqrt{\lambda_k} : k = 1, \dots, n \right\}$$

where λ_k are the eigenvalues of B.

$$A^T A x_k = \lambda_k x_k$$

In other words...

To find $\|A\|_2$ take the square root of the largest eigenvalue of $B = A^T A$.