Caleb Logemann
MATH 562 Numerical Analysis II
Homework 4

1. For each of the following, show that the statement is correct, or give a counter-example. If nothing else is written, assume that $A \in \mathbb{C}^{m \times m}$.

   (a) If λ is an eigenvalue of A and $\mu \in \mathbb{C}$, then $\lambda - \mu$ is an eigenvalue of $A - \mu I$.
   Yes this is a true statement.

   Proof. Let $\mathbf{x}$ be the eigenvector for the eigenvalue λ, that is $A\mathbf{x} = \lambda\mathbf{x}$. Thus

   $$\begin{aligned}(A - \mu I)\mathbf{x} &= A\mathbf{x} - \mu I\mathbf{x} \\ &= \lambda\mathbf{x} - \mu\mathbf{x} \\ &= (\lambda - \mu)\mathbf{x}\end{aligned}$$

   Therefore $\mathbf{x}$ is an eigenvector of $A - \mu I$ and the corresponding eigenvalue is $\lambda - \mu$. □

   (b) If A is real and λ is an eigenvalue of A, then $-\lambda$ is an eigenvalue of A.
   This is false. Consider the matrix

   $$A = \begin{bmatrix} 2 & 0 \\ 0 & 3 \end{bmatrix}.$$

   The eigenvalues of this matrix are 2 and 3, neither -2 nor -3 are eigenvalues.

   (c) If A is real and λ is an eigenvalue of A, then $\bar{\lambda}$ is an eigenvalue of A.

   Proof. Let A be real and let λ be an eigenvalue of A. Thus there exists a nonzero eigenvector $\mathbf{x}$ such that $A\mathbf{x} = \lambda\mathbf{x}$. Taking the complex conjugate of both sides results in $A\bar{\mathbf{x}} = \bar{\lambda}\bar{\mathbf{x}}$, since A is real. Therefore $\bar{\lambda}$ is an eigenvalue of A. □

   (d) If λ is an eigenvalue of A and A is nonsingular, then λ^{-1} is an eigenvalue of A^{-1}.

   Proof. Let A be nonsingular and let λ be an eigenvalue of A. Since A is nonsingular $\lambda \neq 0$. There exists a nonzero eigenvector $\mathbf{x}$ such that $A\mathbf{x} = \lambda\mathbf{x}$. Left multiplying by A^{-1} results in $\mathbf{x} = \lambda A^{-1}\mathbf{x}$. Dividing by λ, shows that $A^{-1}\mathbf{x} = \lambda^{-1}\mathbf{x}$. Therefore λ^{-1} is an eigenvalue of A^{-1}. □

(e) If all the eigenvalues of A are zero, than $A = 0$.

This is false. Consider the matrix

$$A = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}.$$

Both of the eigenvalues of this matrix are zero, however $A \neq 0$.

(f) If A is Hermitian and λ is an eigenvalue of A, then λ is a singular value of A.

This is false. If

$$A = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -2 & 0 \\ 0 & 0 & -2 \end{bmatrix},$$

then A is Hermition, the eigevalues of A are $\{1, -2, -2\}$, but the singular values of A are $\{1, 2, 2\}$. If A is Hermitian than the singular values are the absolute values of the eigenvalues.

(g) If A is diagonalizable and all eigenvalues are equal, then A is diagonal.

Proof. Let A be diagonalizable with all eigenvalues equal. There exists some matrix X such that $A = X \Lambda X^{-1}$. Since all of the eigenvalues of A are the same, $\Lambda = \lambda I$. Therefore $A = X \lambda I X^{-1} = \lambda X X^{-1} = \lambda I$, so A is diagonal. $\square$

2. (a) Let $A \in \mathbb{C}^{m \times m}$ be tridiagonal and Hermitian, with all of its subdiagonal and superdiagonal entries nonzero. Prove that the eigenvalues of A are distinct.

Proof. Let $A \in \mathbb{C}^{m \times m}$ be tridiagonal and Hermitian, with all subdiagonal and superdiagonal entries nonzero. Let λ be a eigenvalue of A, that is $A\mathbf{x} = \lambda \mathbf{x}$ for some nonzero vector $\mathbf{x}$. Consider the matrix $A - \lambda I$. Since λ is an eigenvalue of A, this matrix is singular, that is $\operatorname{rank}(A - \lambda I) \leq m - 1$. Now consider any $(m-1) \times (m-1)$ principal submatrix of $A - \lambda I$. This submatrix will be diagonal, with all diagonal entries nonzero, because $A - \lambda I$ is tridiagonal with all subdiagonal and superdiagonal entries nonzero. This implies that the submatrix is nonsingular, hence has a rank of $m - 1$. This implies that $\operatorname{rank}(A - \lambda I) \geq m - 1$. Therefore $\operatorname{rank}(A - \lambda I) = m - 1$. The geometric multiplicity of the eigenvalue λ is equal to the dimension of the nullspace of $A - \lambda I$. The rank-nullity theorem now states that the dimension of the nullspace of $A - \lambda I$ is 1, therefore the geometric multiplicity of all eigenvalues is 1. For Hermitian matrices the geometric multiplicity and algebraic multiplicity of eigenvalues are equivalent, therefore the algebraic multiplicities of all the eigenvalues of A are 1. This implies that all eigenvalues are distinct. $\square$

(b) Let A be upper-Hessenberg, with all of its subdiagonal entries nonzero. Give an example that shows that the eigenvalues of A are not necessarily distinct.

One example of this is

$$A = \begin{bmatrix} 3 & 3 & 0 \\ 3 & 3 & 3 \\ 0 & -3 & 3 \end{bmatrix}$$

This matrix is upper-Hessenberg, but all of the eigenvalues of A are 3. One reason this is possible is that the geometric multiplicity of this eigenvalue is 1, not 3.

3. Suppose A is $m \times m$ and has a complete set of orthonormal eigenvectors, $\mathbf{q}_1, \ldots, \mathbf{q}_m$, and with corresponding eigenvalues $\lambda_1, \ldots, \lambda_m$. Assume that the ordering is such that $|\lambda_j| \geq |\lambda_{j+1}|$. Furthermore assume that $|\lambda_1| > |\lambda_2| > |\lambda_3|$. Consider the artificial version of the power method $\mathbf{v}^{(k)} = A\mathbf{v}^{(k-1)}/\lambda_1$ with $\mathbf{v}^{(0)} = \alpha_1\mathbf{q}_1 + \cdots + \alpha_m\mathbf{q}_m$, where α_1 and α_2 are both nonzero. Show that the sequence converges linearly to $\alpha_1\mathbf{q}_1$ with asymptotic constant $C = |\lambda_2/\lambda_1|$.

Proof. First note that

$$\begin{aligned} \mathbf{v}^{(k)} &= \frac{A\mathbf{v}^{(k-1)}}{\lambda_1} \\ &= \frac{A^k\mathbf{v}^{(0)}}{\lambda_1^k} \\ &= \frac{A^k}{\lambda_1^k}(\alpha_1\mathbf{q}_1 + \cdots + \alpha_m\mathbf{q}_m) \\ &= \frac{1}{\lambda_1^k}\Big(\alpha_1 A^k\mathbf{q}_1 + \cdots + \alpha_m A^k\mathbf{q}_m\Big) \end{aligned}$$

Since the vectors $\mathbf{q}_i$ are eigenvectors of A.

$$\begin{aligned} &= \frac{1}{\lambda_1^k}\Big(\alpha_1\lambda_1^k\mathbf{q}_1 + \cdots + \alpha_m\lambda_m^k\mathbf{q}_m\Big) \\ &= \alpha_1\mathbf{q}_1 + \alpha_2\left(\frac{\lambda_2}{\lambda_1}\right)^k\mathbf{q}_2 + \cdots + \alpha_m\left(\frac{\lambda_m}{\lambda_1}\right)^k\mathbf{q}_m \end{aligned}$$

Since $|\lambda_1| > |\lambda_i|$ for $2 \leq i \leq m$, $\left|\frac{\lambda_i}{\lambda_1}\right| < 1$. This means that as $k \to \infty$, $\left(\frac{\lambda_i}{\lambda_1}\right)^k \to 0$ and $\mathbf{v}^{(k)} \to \alpha_1\mathbf{q}_1$. To see how quickly $\mathbf{v}^{(k)}$ is converging to $\alpha_1\mathbf{q}_1$, consider $\left|\mathbf{v}^{(k)} - \alpha_1\mathbf{q}_1\right|$. From the previous work we know that

$$\left\|\mathbf{v}^{(k)} - \alpha_1\mathbf{q}_1\right\| = \left\|\alpha_2\left(\frac{\lambda_2}{\lambda_1}\right)^k\mathbf{q}_2 + \cdots + \alpha_m\left(\frac{\lambda_m}{\lambda_1}\right)^k\mathbf{q}_m\right\|$$

Since $|\lambda_2| > |\lambda_i|$ for $3 \le i \le m$

$$\left\|\mathbf{v}^{(k)} - \alpha_1 \mathbf{q}_1\right\| \le \left|\frac{\lambda_2}{\lambda_1}\right|^k \|\alpha_2 \mathbf{q}_2 + \cdots + \alpha_m \mathbf{q}_m\|$$

This shows that $\mathbf{v}^{(k)}$ is convergin to $\alpha_1 \mathbf{q}_1$ linearly with coefficient $\left|\frac{\lambda_2}{\lambda_1}\right|$. $\square$

4. Consider the matrix

$$A = \begin{bmatrix} -1 & 0 & 1 \\ 1 & -1 & 0 \\ 0 & 1 & -1 \\ 1 & 0 & 1 \end{bmatrix}$$

(a) Calculate the eigenvalues and eigenvectors of $A^T A$

First we must compute the matrix, $A^T A$.

$$A^T A = \begin{bmatrix} 3 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 3 \end{bmatrix}$$

The eigenvalues can be found by using the characteristic polynomial, that is $p(z) = \det(zI - A^T A)$.

$$\begin{aligned} \det(zI - A^T A) &= \begin{vmatrix} z-3 & 1 & 0 \\ 1 & z-2 & 1 \\ 0 & 1 & z-3 \end{vmatrix} \\ &= (z-3)^2(z-2) - (z-3) - (z-3) \\ &= (z-3)((z-3)(z-2) - 2) \\ &= (z-3)\left(z^2 - 5z + 4\right) \\ &= (z-3)(z-4)(z-1) \end{aligned}$$

The eigenvalues are the zeros of the characteristic polynomial, therefore $\operatorname{spec}(()A) = \{1, 3, 4\}$.

The eigenvectors of $A^T A$ can be found by solving the following systems

$$\begin{aligned} (I - A^T A)\mathbf{x} &= \mathbf{0} \\ (3I - A^T A)\mathbf{x} &= \mathbf{0} \\ (4I - A^T A)\mathbf{x} &= \mathbf{0} \end{aligned}$$

First I will solve $(I - A^T A)\mathbf{x} = \mathbf{0}$ using the augmented system.

$$\begin{bmatrix} -2 & 1 & 0 & 0 \\ 1 & -1 & 1 & 0 \\ 0 & 1 & -2 & 0 \end{bmatrix}$$
$$\begin{bmatrix} 1 & -1/2 & 0 & 0 \\ 1 & -1 & 1 & 0 \\ 0 & 1 & -2 & 0 \end{bmatrix}$$
$$\begin{bmatrix} 1 & -1/2 & 0 & 0 \\ 0 & -1/2 & 1 & 0 \\ 0 & 1 & -2 & 0 \end{bmatrix}$$
$$\begin{bmatrix} 1 & -1/2 & 0 & 0 \\ 0 & 1 & -2 & 0 \\ 0 & 1 & -2 & 0 \end{bmatrix}$$
$$\begin{bmatrix} 1 & -1/2 & 0 & 0 \\ 0 & 1 & -2 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$
$$\begin{bmatrix} 1 & 0 & -1 & 0 \\ 0 & 1 & -2 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

Thus the solutions to this system are of the form

$$\begin{bmatrix} x \\ 2x \\ x \end{bmatrix}$$

The eigenvector with 2-norm equal to one for eigenvalue 1 is

$$\begin{bmatrix} 1/\sqrt{6} \\ 2/\sqrt{6} \\ 1/\sqrt{6} \end{bmatrix}$$

The eigenvector vector for eigenvalue 3 can be found as

$$\begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

Thus the solutions to this system are of the form

$$\begin{bmatrix} x \\ 0 \\ -x \end{bmatrix}$$

The eigenvector with 2-norm equal to one for eigenvalue 3 is

$$\begin{bmatrix} 1/\sqrt{2} \\ 0 \\ -1/\sqrt{2} \end{bmatrix}$$

Lastly the eigenvector for eigenvalue 4 is needed.

$$\begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 2 & 1 & 0 \\ 0 & 1 & 1 & 0 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 0 & -1 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

Thus the solutions to this system are of the form

$$\begin{bmatrix} x \\ -x \\ x \end{bmatrix}$$

The eigenvector with 2-norm equal to one for eigenvalue 3 is

$$\begin{bmatrix} 1/\sqrt{3} \\ -1/\sqrt{3} \\ 1/\sqrt{3} \end{bmatrix}$$

Thus the eigenvalue decomposition of $A^T A$ is

$$A^T A = X \Lambda X'$$

$$X = \begin{bmatrix} 1/\sqrt{6} & 1/\sqrt{2} & 1/\sqrt{3} \\ 2/\sqrt{6} & 0 & -1/\sqrt{3} \\ 1/\sqrt{6} & -1/\sqrt{2} & 1/\sqrt{3} \end{bmatrix}$$

$$\Lambda = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 3 & 0 \\ 0 & 0 & 4 \end{bmatrix}$$

(b) Use your results in (a) to compute (by hand) the SVD of A.

The singular values of A are the nonnegative square roots of the eigenvalues of $A^T A$. Thus if $A = U\Sigma V^T$ is a singular value decomposition of A, then

$$\Sigma = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \sqrt{3} & 0 \\ 0 & 0 & 2 \end{bmatrix}$$

$$V = \begin{bmatrix} 1/\sqrt{6} & 1/\sqrt{2} & 1/\sqrt{3} \\ 2/\sqrt{6} & 0 & -1/\sqrt{3} \\ 1/\sqrt{6} & -1/\sqrt{2} & 1/\sqrt{3} \end{bmatrix}$$

The unitary matrix U can be found by solving the system $U\Sigma = AV$, for U unitary. Since Σ is invertible, $U = AV\Sigma^{-1}$

$$U = \begin{bmatrix} 0 & -2/\sqrt{6} & 0 \\ -1/\sqrt{6} & 1/\sqrt{6} & 1/\sqrt{3} \\ 1/\sqrt{6} & 1/\sqrt{6} & -1/\sqrt{3} \\ 2/\sqrt{6} & 0 & 1/\sqrt{3} \end{bmatrix}$$

(c) Find the 1-, 2-, ∞-, and Frobenius norms of A.

The 1-norm of a matrix is the maximum absolute column sum. Therefore $\|A\|_1 = 3$.

The 2-norm of a matrix is the maximum singular value. Therefore $\|A\|_2 = 2$.

The ∞-norm of a matrix is the maximum absolute row sum. Therefore $\|A\|_\infty = 2$.

The Frobenius norm of a matrix is the squareroot of the sum of the squares of the entries. Therefore $\|A\|_F = \sqrt{8} = 2\sqrt{2}$.

5. Write a MATLAB function $[v, lam, k] = Pwr(A, v0)$ that uses the method of power iteration to compute the largest eigenvalue, "lam", and a corresponding eigenvector v that has length one in the 2-norm. The third argument returned, k, should be the number of iterations used in the computation. The input data is a square matrix A and a starting vector $v0$.

```
function [v, lam, k] = Pwr1(A, v0)

    % normalize v0
    v = v0/norm(v0,2);
    % replace v0 so that while loop condition works
    v0 = zeros(size(v));
    % initiate v so while loop doesn't quite at first check
    k = 0;
```

```
    lam = 0;
    while k < 500 && norm(v - v0, 2) > 1e-8
        % update next step v0 acts like v^(k-1)
        k = k + 1;
        v0 = v;

        w = A*v0;
        % normalize
        v = w/norm(w,2);
        % Rayleigh quotient
        lam = v'*A*v;
    end
end
```

```
function [v, lam, k] = Pwr2(A, v0)

    % normalize v0
    v = v0/norm(v0,2);
    % initiate lam variables so while loop condition works for first ...
       iteration
    lam0 = 0;
    lam = 1;
    k = 0;
    while k < 500 && abs(lam - lam0) > 1e-8
        % update next step v0 acts like v^(k-1)
        k = k + 1;
        v0 = v;
        lam0 = lam;

        w = A*v0;
        % normalize
        v = w/norm(w,2);
        % Rayleigh quotient
        lam = v'*A*v;
    end
end
```

```
%% Problem 5
% part (a)
A = diag([-4, 2, 1, 1, 1]) + triu(rand(5,5),1);
v0 = ones(5, 1);
[v1, lam1, k1] = Pwr1(A, v0)
[v2, lam2, k2] = Pwr2(A, v0)

% part (b)
A = diag([9, 2, 1, 5, -8]) + triu(rand(5,5),1);
v0 = ones(5, 1);
```

```
[v1, lam1, k1] = Pwr1(A, v0)
[v2, lam2, k2] = Pwr2(A, v0)
```

For the first data set, the second criterion worked much better. In fact the first criterion didn't even converge fully. The first criterion stopped at 500 iterations.

```
v1 =

    1.0000
    0.0000
    0.0000
    0.0000
    0.0000


lam1 =

    -4


k1 =

   500


v2 =

    1.0000
    0.0000
    0.0000
    0.0000
    0.0000


lam2 =

   -4.0000


k2 =
```

```
    28
```

For the second data set the second criterion worked only slightly better. This may be because the eigenvalues are more spread out in the second data set.

```
v1 =

    1.0000
    0.0000
    0.0000
    0.0000
   -0.0000


lam1 =

    9.0000


k1 =

   161


v2 =

    1.0000
   -0.0000
   -0.0000
   -0.0000
    0.0000


lam2 =

    9.0000


k2 =
```

158

6. The function for the Inverse iteration is shown below

```
function [v, lam, k] = Inv(A, v0, mu)
    % set up
    v = v0/norm(v0,2);
    % make sure while condition works initially
    lam = mu;
    lam0 = mu - 1;
    k = 0;
    m = size(A);
    while k < 500 && abs(lam - lam0) > 1e-8
        % update next step v0 acts like v^(k-1)
        k = k + 1;
        v0 = v;
        lam0 = lam;

        % inverse iteration
        w = (A - mu*eye(m))\v0;
        % normalize
        v = w/norm(w,2);
        % Rayleigh quotient
        lam = v'*A*v;
    end
end
```

```
%% Problem 6
A = diag([9, 2, 1, 5, -8]) + triu(rand(5,5),1);
v0 = ones(5, 1);
[v, lam, k] = Inv(A, v0, 8.8)
```

This method converged extremely fast, much faster than the power iteration. I think that each iteration may be more work, than an iteration of the Power iteration, because the Inverse iteration requires solving a linear system. That being said the additional knowledge of approximately the eigenvalue allows for faster convergence.

```
v =

    1.0000
    0.0000
    0.0000
    0.0000
```

```
    0.0000


lam =

    9.0000


k =

     8
```

7. The function for the Rayleigh quotient iteration is shown below

```
function [v, lam, k] = Ray(A, v0)
    % set up
    v = v0/norm(v0,2);;
    lam = v'*A*v;
    % make sure while condition works initially
    lam0 = 0;
    k = 0;
    m = size(A);
    while k < 500 && abs(lam - lam0) > 1e-8
        % update next step v0 acts like v^(k-1)
        k = k + 1;
        v0 = v;
        lam0 = lam;

        % inverse iteration
        w = (A - lam0*eye(m))\v0;
        % normalize
        v = w/norm(w,2);
        % Rayleigh quotient
        lam = v'*A*v;
    end
end
```

```
%% Problem 7
A = diag([9, 2, 1, 5, -8]) + triu(rand(5,5),1);
v0 = ones(5, 1);
[v, lam, k] = Ray(A, v0)
```

The Rayleigh quotient iteration is the most efficient algorithm so far and it is able to find any eigenvalue eigenvector pair. This algorithm will locate whatever eigenvalue whose eigenvector is closest to the initial input vector. Trying all sorts of different input vectors results in different eigenvalue/eigenvector pairs being found. Each time the algorithm converges extremely quickly. The number of iterations is always in the single digits. Below is a sample result.

```
v =

   -0.0482
    0.9988
    0.0000
    0.0000
    0.0000


lam =

    2.0000


k =

     6
```