Fall 2018: Numerical Analysis
Assignment 4 (due Nov. 8, 2018)

1. **[Eigenvalue/vector properties, 4pts]** Prove the following statements, using the basic definition of eigenvalues and eigenvectors, or give a counterexample showing the statement is not true. Assume $A \in \mathbb{R}^{n\times n}$, $n \geq 1$.

   (a) If λ is an eigenvalue of A and $\alpha \in \mathbb{R}$, then $\lambda + \alpha$ is an eigenvalue of $A + \alpha I$, where I is the identity matrix.

   (b) If λ is an eigenvalue of A and $\alpha \in \mathbb{R}$, then $\alpha\lambda$ is an eigenvalue of αA.

   (c) If λ is an eigenvalue of A, then for any positive integer k, λ^k is an eigenvalue of A^k.

   (d) If B is "similar" to A, which means that there is a nonsingular matrix S such that $B = SAS^{-1}$, then if λ is an eigenvalue of A, it is also an eigenvalue of B. How do the eigenvectors of B relate to the eigenvectors of A?

   (e) Every matrix with $n \geq 2$ has at least two distinct eigenvalues, say λ and μ, with $\lambda \neq \mu$.

   (f) Every real matrix has a real eigenvalue.

   (g) If A is singular, then it has an eigenvalue equal to zero.

   (h) If all the eigenvalues of a matrix A are zero, then $A = 0$.

2. **[2+2+2+2pt]** Power Method and Inverse Iteration.

   (a) Implement the Power Method for an arbitrary matrix $A \in \mathbb{R}^{n\times n}$ and an initial vector $\boldsymbol{x}_0 \in \mathbb{R}^n$.

   (b) Use your code to find an eigenvector of

$$A = \begin{bmatrix} -2 & 1 & 4 \\ 1 & 1 & 1 \\ 4 & 1 & -2 \end{bmatrix},$$

   starting with $x_0 = (1, 2, -1)^T$ and $x_0 = (1, 2, 1)^T$. Report the first 5 iterates for each of the two initial vectors. Then use MATLAB's `eig(A)` to examine the eigenvalues and eigenvectors of A. Where do the sequences converge to? Why do the limits not seem to be the same?

   (c) Implement the Inverse Power Method for an arbitrary matrix $A \in \mathbb{R}^{n\times n}$, an initial vector $\boldsymbol{x}_0 \in \mathbb{R}^n$ and an initial eigenvalue guess $\theta \in \mathbb{R}$.

   (d) Use your code from (c) to calculate *all* eigenvectors of A. You may pick appropriate values for θ and the initial vector as you wish (obviously not the eigenvectors themselves). Always report the first 5 iterates and explain where the sequence converges to and why.

Please also hand in your code.

3. **[2+3+2+2pt]** Orthogonalization methods.

   (a) Given any two nonzero vectors $\boldsymbol{x}$ and $\boldsymbol{y}$ in $\mathbb{R}^n$, construct a Householder matrix H, such that $H\boldsymbol{x}$ is a scalar multiple of $\boldsymbol{y}$. Is the matrix H unique?

   (b) Use Householder matrices to transform the matrix A into tridiagonal form.

$$A = \begin{bmatrix} 2 & 1 & 2 & 2 \\ 1 & -7 & 6 & 5 \\ 2 & 6 & 2 & -5 \\ 2 & 5 & -5 & 1 \end{bmatrix}.$$

   (c) Use Householder matrices to compute the QR-factorization of the matrix from homework 3, problem 6, i.e.:

$$\begin{bmatrix} 9 & -6 \\ 12 & -8 \\ 0 & 20 \end{bmatrix}.$$

   (d) Use Givens rotations to transform the vector

$$\boldsymbol{x} = \begin{bmatrix} 4 \\ -3 \\ 1 \end{bmatrix}$$

   to a multiple of the first unit vector. Specify the Givens rotations you used.

4. **[1+2+1pt]** Let the matrix $A \in \mathbb{R}^{n\times n}$ be defined by its components

$$\begin{aligned} &a_{i,i} = 2i \quad \text{for} \quad i = 1, ...n, \\ &a_{i,i+1} = -i, \quad a_{i+1,i} = -i \quad \text{for} \quad i = 1, ...n-1, \end{aligned}$$

   the remaining components are zero.

   (a) Sketch the Gerschgorin discs of A.

   (b) Use Gerschgorin's Theorem and facts about symmetric matrices to show that all eigenvalues of A are real numbers, larger or equal 1.

   (c) Using Gerschgorin discs, find an upper bound for the eigenvalues of A.

5. **[2+2pt]** An efficient way to find individual roots of a polynomial is to use Newton's method. However, as we have seen, Newton's method requires an initialization close to the root one wants to find, and it can be difficult to find *all* roots of a polynomial. Luckily, one can use the relation between eigenvalues and polynomial roots to find all roots of a given polynomial. Let us consider a polynomial of degree n with leading coefficient 1:

$$p(x) = a_0 + a_1 x + \ldots + a_{n-1}x^{n-1} + x^n \quad \text{with } a_i \in \mathbb{R}.$$

(a) Show that $p(x)$ is the characteristic polynomial of the matrix (sometimes called a companion matrix for p)

$$A_p := \begin{bmatrix} 0 & & & -a_0 \\ 1 & & & -a_1 \\ & \ddots & & \vdots \\ & & 1 & -a_{n-1} \end{bmatrix} \in \mathbb{R}^{n\times n}.$$

Thus, the roots of $p(x)$ can be computed as the eigenvalues of A_p using the QR algorithm (as implemented, e.g., in MATLAB's `eig` function).

(b) Let us consider Wilkinson's polynomial $p_w(x)$ of order 15, i.e., a polynomial with the roots $1, 2, \ldots, 15$:

$$p_w(x) = (x-1)\cdot(x-2)\cdot\ldots\cdot(x-15).$$

The corresponding coefficients can be found using the `poly()` function. Use these coefficients in the matrix A_p to find the original roots again, and compute their error. Compare with the build-in method (called `roots()`) for finding the roots of a polynomial.[1]

6. **[1+1+2pt, all extra credit]** As mentioned in class, the Google page rank algorithm has a lot to do with the eigenvector corresponding to the largest eigenvalue of a so-called stochastic matrix, which models the links between websites.[2] Stochastic matrices have non-negative entries and each column sums to 1, and one can show (under a few technical assumptions) that it has the eigenvalues $\lambda_1 = 1 > |\lambda_2| \geq \ldots \geq |\lambda_n|$. Thus, we can use the power method[3] to find the eigenvector $\boldsymbol{v}$ corresponding to λ_1, which can be shown to have either all negative or all positive entries. These entries can be interpreted as the importance of individual websites.

Let us construct a large stochastic matrices (pick a size $n \geq 100$, the size of our "toy internet") in MATLAB as follows:

```
I = eye(n);
A = 0.5*I(randperm(n),:)  + (max(2,randn(n,n))-2);
A = A - diag(diag(A));
L = A*diag(1./(max(1e-10,sum(A,1))));
```

(a) Plot the sparsity structure of L (i.e., the nonzero entries in the matrix) using the command `spy`. Each non-zero entry corresponds to a link between two websites.

(b) Plot the (complex) eigenvalues of L by plotting the real part of the eigenvalues on the x-axis, and the imaginary part on the y-axis.[4] Additionally, plot the unit circle and check that all eigenvalues are inside the unit circle, but $\lambda_1 = 1$.

[1]Note that for MATLAB functions that do not use external libraries, you can see how they are implemented by typing `edit name_of_function`. Doing that for the `roots` function will show you that MATLAB implements root finding exactly using a companion matrix as described above.

[2]See the interesting 2006 SIREV paper *The 25,000,000,000 eigenvector. The linear algebra behind Google* by Kurt Bryan and Tanya Leise. It's easy to find—just google it!

[3]We have mostly discussed the power method for symmetric matrices, but it also works for non-symmetric matrices.

[4]Please make sure that the plotted eigenvalues are not connected.

(c) The matrix L contains many zeros. One of the technical assumptions for proving theorems is that all entries in L are positive. As a remedy, one considers the matrices $S = \kappa L + (1-\kappa)E$, where E is a matrix with entries $1/n$ in every component[5]. Study the influence of κ numerically by visualizing the eigenvalues of S for different values of κ. Why will $\kappa < 1$ improve the speed of convergence of the power method?

[5] In the original Brin/Page Google paper, the authors use $\kappa = 0.85$. The introduction of the matrix E makes each matrix entry positive and also helps dealing with web pages without outgoing links, which lead to zero columns in L.