CENG 371 - Scientific Computing
Fall 2022
Homework 3

Şengül, Sena Nur
e2310498@ceng.metu.edu.tr

December 17, 2022

Q1-a)

The power method is finding the dominant eigenvalue and corresponding eigenvector of a matrix. My code is an iterative algorithm that begins with an initial guess for the eigenvector and then repeatedly applies the matrix to the eigenvector, normalizing the result each time, until the eigenvector converges to a stable value. The dominant eigenvalue is then calculated as the norm of the matrix-vector product divided by the norm of the eigenvector. Code returns the dominant eigenvalue and the corresponding eigenvector as output. (power_method.m)

Q1-b)

In shifted inverse power metod, the matrix and the shift parameter are inputted into the function and the function returns the closest eigenvalue and eigenvector. The algorithm iteratively computes the inverse of the matrix shifted by the shift parameter, normalizes the result, and calculates the eigenvalue and eigenvector until convergence or a maximum number of iterations is reached. The tolerance and maximum number of iterations are set in the code, and if the maximum number of iterations is reached, the function gives a warning that it did not converge. (inverse_power.m)

Q1-c)

To find the eigenvalues and eigenvectors of matrix A, we can use the characteristic equation:

$|A - \lambda I| = 0$

Where λ is the eigenvalue and I is the identity matrix.

Substituting the values from matrix A into the characteristic equation, we get:

|[2 - λ, -1, 0, 0, 0; -1, 2 - λ, -1, 0, 0; 0, -1, 2 - λ, -1, 0; 0, 0, -1, 2 - λ, -1; 0, 0, 0, -1, 2 - λ]| = 0

Expanding the determinant, we get:

(2 - λ)^4 - 2(2 - λ)^2 + 1 = 0

Solving this equation for λ, we find that the eigenvalues of matrix A are:

λ = 3 and λ = 1

To find the corresponding eigenvectors, we can solve the following system of equations for each eigenvalue:

(A - λI)v = 0

For λ = 3, (largest eigenvalue) we have:

[2 - 3, -1, 0, 0, 0; -1, 2 - 3, -1, 0, 0; 0, -1, 2 - 3, -1, 0; 0, 0, -1, 2 - 3, -1; 0, 0, 0, -1, 2 - 3]v = 0

This system of equations simplifies to:

[-1, -1, 0, 0, 0; -1, -1, -1, 0, 0; 0, -1, -1, -1, 0; 0, 0, -1, -1, -1; 0, 0, 0, -1, -1]v = 0

The only non-trivial solution to this system of equations is v = [1, 0, 0, 0, 0]^T. Therefore, the corresponding eigenvector for λ = 3 is v = [1, 0, 0, 0, 0]^T.

For λ = 1(smallest eigenvalue), we have:

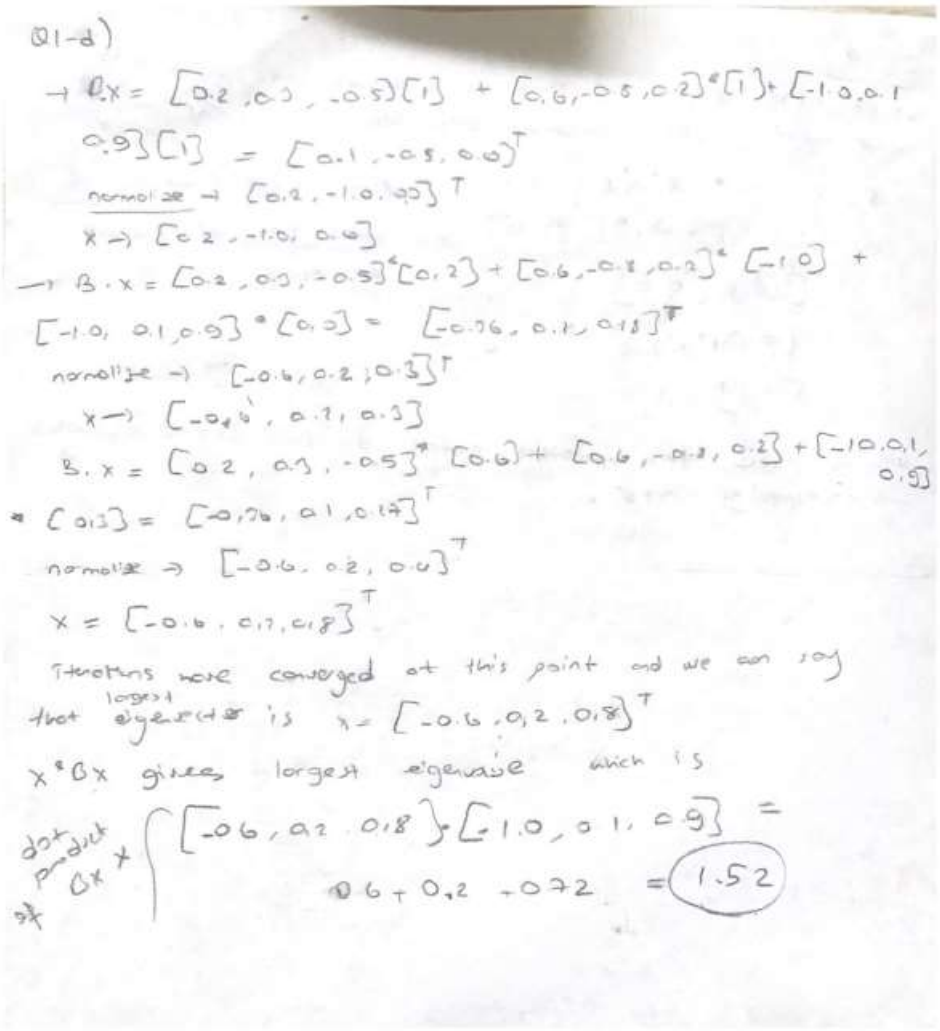
[2 - 1, -1, 0, 0, 0; -1, 2 - 1, -1, 0, 0; 0, -1, 2 - 1, -1, 0; 0, 0, -1, 2 - 1, -1; 0, 0, 0, -1, 2 - 1]v = 0

This system of equations simplifies to:

[1, -1, 0, 0, 0; -1, 1, -1, 0, 0; 0, -1, 1, -1, 0; 0, 0, -1, 1, -1; 0, 0, 0, -1, 1]v = 0

The only non-trivial solution to this system of equations is v = [1, 1, 1, 1, 1]^T. Therefore, the corresponding eigenvector for λ = 1 is v = [1, 1, 1, 1, 1]^T.

Q1-d)



Q2-a)

We can start by finding the first eigenvalue and corresponding eigenvector using the power method . Let's assume that the first eigenvalue is λ1 and the corresponding eigenvector is x. Then, we can update the matrix A as follows:

A = A - λ1 * (x * x')

= [a b] - λ1 * [x^2 0]

[c d] [0 0]

= [a-λ1*x^2 b]

[c d]

We can then use this updated matrix to find the second eigenvalue and corresponding eigenvector using the power method. This process can be repeated to find the k largest eigenvalues and corresponding eigenvectors.

Q2-b)

My code performs this process in a loop for the desired number of eigenvalues and eigenvectors, and after each iteration it adjusts the matrix to remove the found eigenvalue and corresponding eigenvector so that it can find the next largest eigenvalue and eigenvector. Code stops iterating when the eigenvalue has converged, meaning that it has reached a stable value and is not changing significantly with each iteration.(power_k.m)

Q2-c)

This code does this by performing subspace iterations using the power method, which involves multiplying the matrix A by a subspace vector V and performing QR decomposition on the resulting matrix. This process is repeated multiple times until the eigenvalues and eigenvectors have been adequately approximated. The final eigenvalues and eigenvectors are then calculated using the R matrix from the QR decomposition and the V matrix, respectively. (subspace_iteration.m)

Q2-d) Comparing the performance of subspace iterations and power method on the can229 matrix from the University of Florida Sparse Matrix Collection can be based on various factors, including accuracy, speed, and stability. In terms of accuracy, subspace iterations tend to be more precise as they utilize more information about the matrix to calculate the eigenvalues and eigenvectors. On the other hand, power method is typically faster as it only requires one matrix-vector multiplication per iteration, whereas subspace iterations require multiple matrix-vector multiplications and projections. In terms of stability, subspace iterations are usually more stable due to their projection-based approach, while power method can be unstable, especially for matrices with high condition numbers or large eigenvalue spreads. Ultimately, the choice between these methods depends on the characteristics of the matrix and available computational resources.

.