Power Method for Calculating Eigenvalues and Eigenvectors

Zexuan Sun
School of Mathematics and Statistics, Wuhan University

1 Introduction

In this document, I will give detailed algorithm of this project, which mainly is divided into three sub-algorithm, the method to reduce the order of matrix, the way to calculate eigenvalues that are conjugate complex roots and the extension of Householder tranformation to complex number.

2 Reduce the Order

Suppose that we want to get the second maximal modulus eigenvalue of a matrix A, in order to use power method for iteration, we have to reduce the order of A. Assume that

$$Ax_1 = \lambda_1 x_1, \tag{1}$$

Say we have a unitary matrix satisfies that

$$Px_1 = \alpha e_1, \tag{2}$$

where $e_1 = (1, 0, \cdots, 0)^{\mathrm{T}}$. Combine equations (1) and (2), we have

$$PAP^* e_1 = \lambda_1 e_1, \tag{3}$$

namely, PAP^* has the shape as below:

$$PAP^* = \begin{bmatrix} \lambda_1 & * \\ 0 & B_1 \end{bmatrix}, \tag{4}$$

where B_1 is a square of $n-1$ order, whose eigenvalues are $\lambda_2, \cdots, \lambda_n$. Therefore, to get λ_2, we can simple implement power method on B_2. About the unitary matrix P, we can employ Household transformation to get it.

3 Calculate Conjugate Complex Roots

It is true that power method would not converge if the maximal modulus eigenvalues are two conjugate complex roots. So I set N as the maximum number of iterations. If power method does not converge after N times of iterations, I will use another function to solve the problem.

In the situation af conjugate complex roots, suppose we have iterated k times, we have approximation as follows:

$$\vec{v}_k = \vec{u}_1 + \vec{u}_2, \tag{5}$$

where $\vec{u}_1$ and $\vec{u}_2$ are two eigenvectors corresponding to the two conjugate complex roots. Multiply this vector by A twice, we have

$$\begin{aligned} \vec{v}_{k+1} &= \lambda_1 \vec{u}_1 + \lambda_2 \vec{u}_2 \\ \vec{v}_{k+2} &= \lambda_1^2 \vec{u}_1 + \lambda_2^2 \vec{u}_2 \end{aligned}, \tag{6}$$

where λ_1 and λ_2 are two conjugate complex roots. Therefore, there exists b, c satisfy

$$\begin{array}{l} \lambda_1^2 + b\lambda_1 + c = 0 \\ \lambda_2^2 + b\lambda_2 + c = 0 \end{array} \tag{7}$$

b, c are tow unkonwn coefficients, then we have

$$c\vec{v}_k + b\vec{v}_{k+1} + \vec{v}_{k+2} \approx \left(c + b\lambda_1 + \lambda_1^2\right)\vec{u}_1 + \left(c + b\lambda_2 + \lambda_2^2\right)\vec{u}_2 = 0. \tag{8}$$

Namley, vector $x = [b; c]$ (x is a column vector, syntax of is used here) satisfy:

$$[\vec{v}_{k+1}, \vec{v}_k]\,\vec{x} = \vec{v}_{k+2}. \tag{9}$$

After solving the linear system, we can get b, c and then the two conjugate complex roots are derived.

4 Extension of Householder tranformation

Suppose complex vector $x = (z_1, z_2, \cdots, z_n)^{\mathrm{T}}$, where $z_i \in C$. Set $D = \operatorname{diag}[\widetilde{z}_1, \cdots, \widetilde{z}_n]$,where $\widetilde{z}_1 = 1$, if $z_i = 0$, otherwise $\widetilde{z}_1 = \bar{z}_i / |z_i|$. Thus, we have

$$Dx = [|z_1|, \cdots, |z_n|]^{\top} \tag{10}$$

Set $y = Dx$, use Householder for real vectors on y, we have

$$Hy = [\sigma, 0, 0, \cdots, 0]^{\mathrm{T}}, \tag{11}$$

where $\sigma = \|y\|_2$.

Finally, we have Householder transformation for complex vectors

$$G = H \cdot D. \tag{12}$$

It is easy to check that G is orthogonal matrix, since we have

$$\bar{G}^{\mathrm{T}} G = (\bar{HD})^{\mathrm{T}}(HD) = \bar{D}^{\mathrm{T}}\bar{H}^{\mathrm{T}}HD = I \tag{13}$$

$$G\bar{G}^{\mathrm{T}} = (HD)(\bar{HD})^{\mathrm{T}} = HD\bar{D}^{\mathrm{T}}\bar{H}^{\mathrm{T}} = I \tag{14}$$

5 Acknowledgement

The algorithm to extend Household tranformation to complex vector bothered me quite a long time. So I turned to *Yilin Chen* for help. The idea from *Yilin Chen* is brilliant. Without the help from him, I was not able to finish the project. Thanks him a lot. Also, the algorithm from *zhihu* is great, but I am not able to find the author now. Sorry about this. Thanks again. Also, thanks all those who have helped me complete my project.