Institute for Information Technology, Uppsala University

Numerical Linear Algebra
VT 2021

Assignment 3
Numerical methods for eigenproblems

1. Implement the Implicitly Restarted Arnoldi method

The task is to write a MatLab function that implements a simplified version of the Implicitly Restarted Arnoldi (IRA) method (see, e.g. Jan Brandts, Lecture Notes).

Part 1

Recall that an *Arnoldi factorization* is an orthonormal basis of the Krylov subspace $\mathcal{K}^{k+1}(A, v)$ represented as the columns of a matrix V_{k+1} together with the upper Hessenberg representation $H_{k+1,k}$ of AV_k on this basis. They are related as follows,

$$AV_k = V_{k+1}H_{k+1,k}. \tag{1}$$

To compute such a factorization, $A, \boldsymbol{v}$, and k should be given. Then firstly, $\boldsymbol{v}$ needs to be scaled to norm one. The resulting vector will be the first column $\boldsymbol{v}_1$ of V_{k+1}. Then apply A to find $\boldsymbol{w} = Av_1$. Orthogonalize $\boldsymbol{w}$ to $\boldsymbol{v}_1$ and normalize the result. This gives the second column $\boldsymbol{v}_2$ of V_{k+1}. By construction, $A\boldsymbol{v}_1$ is a linear combination of $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$, and the coefficients that determine which linear combination it is, are the coefficients h_{11} and h_{21} of the matrix $H_{k+1,k} = (h_{ij})$. The next step is to compute $A\boldsymbol{v}_2$, and to orthogonalize it to $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$, to normalize the result, and so on.

Part 2

Write a MatLab function

```
[V,H] = ArnoldiInit(A,v)
```

that computes the Arnoldi factorization for given `A` and `v` with $k = 1$. Thus, `V` has two columns and `H` is two by one only.

Part 3

Next, write a function that extends a given factorization:

```
[V,H] = ArnoldiPlus(A,V,H,m)
```

In this function, you may assume that the inputs `V` and `H` represent a valid Arnoldi factorization for some $k \geq 1$. This assures that `V` has at least two columns, and `H` at least one. The function should extend the factorization by computing `m` more basis vectors to be added to `V`. The returned matrix `V` should therefore have $n \times (k+m)$ mutually orthonormal columns. The matrix `H` must be $(k+m) \times k+m-1$.

Part 4

Test your routine on an example. Compute the input factorization of ArnoldiPlus using ArnoldiInit. In your test, use a small matrix, and check whether

- `V` and `H` are indeed orthogonal and upper Hessenberg, and
- relation (1) should hold.

Part 5

Test the routine on a symmetric matrix A having the distinct and single eigenvalues $1, 2, \ldots, 20$ and compute the eigenvalue approximations for A in each expansion by one more dimension. This way you should be able to reproduce a triangle of eigenvalue approximations like in (5-7) in Jan Brandts, Lecture Notes 7-8.

Part 6

Given any value μ in the complex plane, and an Arnoldi factorization of length $k \geq 2$, we have seen that it is possible to compute the Arnoldi factorization of length $k-1$ for the start vector $\hat{\boldsymbol{v}}_1 = \tilde{\boldsymbol{v}}_1 / \|\tilde{\boldsymbol{v}}_1\|$ with $\tilde{\boldsymbol{v}}_1 = (A - \mu I)\boldsymbol{v}$ without using any additional matrix vector multiplication with the matrix A again. Write a function for this:

```
[V,H] = ArnoldiMinus(V,H,mu)
```

Again, test this routine on a simple example.

Part 7

Test your program on the matrices A and B and a starting vector $\boldsymbol{v}$, provided by the function `arnoldi_test_matrices.m`.

Part 8

Finally, write a program that combines the routines above and that implements the Implicitly Restarted Arnoldi Method as follows:

- At each iteration, list the eigenvalue approximations

- Prompt for the choice between expanding further, or removing an eigenvalue
- If you like GUIs: One could implement this idea graphically: show the approximate eigenvalues in the complex plane. A mouseclick on an eigenvalue should remove it. A mouseclick on an *expand* button should expand the space further.

2. Theoretical exercises

Answer the following questions and prove your answers.

Let n be a positive integer, A is a real matrix of size n, $\boldsymbol{v} \in \mathbb{C}^n$ with $\|\boldsymbol{v}\| = 1$ and $\mu \in \mathbb{C}$. Let $\boldsymbol{r} = A\boldsymbol{v} - \mu\boldsymbol{v}$ and assume $\boldsymbol{r} \neq \mathbf{0}$.

1. Is it true that if $\boldsymbol{v}^* A\boldsymbol{v} = \mu$ then $(\mu, \boldsymbol{v})$ is an eigenpair of A?
2. Is $(\mu, \boldsymbol{v})$ an eigenpair of $A - \boldsymbol{v}\boldsymbol{r}^*$?
3. Is $A - \mu I$ nonsingular?
4. Is $(\mu, \boldsymbol{v})$ an eigenpair of $\mu\boldsymbol{v}\boldsymbol{v}^*$?
5. If $\|A - B\| = \|\boldsymbol{r}\|$, is then $(\mu, \boldsymbol{v})$ an eigenpair of B?
6. Consider an upper-triangular matrix T. Suppose we want to switch the first and the kth diagonal elements. Thus, we are interested in a unitary matrix Q, such that $\widetilde{T} = QTQ^*$ and $\widetilde{T}(1,1) = T(k,k), \widetilde{T}(k,k) = T(1,k)$. Show that this can be achieved with Givens rotations. How many Givens rotations are required?

Success!

Deadline: as on the course webpage.

Any comments on the assignment will be highly appreciated and will be considered for further improvements. Thank you!