MACM 316 – Computing Assignment 3

- Read the *Guidelines for Assignments* first.

- Submit a one-page PDF report to Canvas and upload you Matlab scripts (as m-files). Do not use any other file formats.

- Keep in mind that Canvas discussions are open forums.

- You must acknowledge any collaborations/assistance from colleagues, TAs, instructors etc.

Computing the exponential of a matrix

Recall from calculus that the exponential of a real number x has the Taylor series expansion

$$\exp(x) = 1 + x + \frac{1}{2!}x^2 + \frac{1}{3!}x^3 + \frac{1}{4!}x^4 + \ldots.$$

Let A be an $n \times n$ matrix. The exponential of A is defined as

$$\exp(A) = I + A + \frac{1}{2!}A^2 + \frac{1}{3!}A^3 + \frac{1}{4!}A^4 + \ldots, \tag{1}$$

where I is the $n \times n$ identity matrix. Matrix exponentials are useful in a number of problems in computational mathematics, including the solution of systems of differential equations.

Warning! The exponential of a matrix is not equal to the exponential of its entries. For example:

$$\exp\left(\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}\right) = \begin{bmatrix} 1.5431 & 1.1752 \\ 1.1752 & 1.5431 \end{bmatrix} \neq \begin{bmatrix} \exp(0) & \exp(1) \\ \exp(1) & \exp(0) \end{bmatrix}.$$

A simple algorithm for computing (an approximation to) the matrix exponential involves summing the first k terms of the infinite series (1). **Your goal in this assignment to investigate the accuracy, efficiency and robustness of this algorithm**. You will be testing on a particular matrix A which is contained in the file *CA3matrix.mat*. Download this file from Canvas and load it into Matlab using the following command:

```
1  load('CA3matrix.mat');
```

This loads the matrix as a 500×500 array `A`.

(a) Let $k = 50$ and $k = 150$. Run your algorithm to get the output, which you should call `expAk`. Plot the result using the following commands:

```
1  imagesc(real(expAk));
2  colormap gray
```

(b) Run your algorithm for $k = 5, 10, 15, \ldots, 150$ and use the `tic` and `toc` commands (recall the in-class demo *TicToc.m*) to plot the computational time versus k. How does the computational time appear to depend on k? Does this agree with what you would expect from counting the number of flops in your algorithm? Explain.

(c) Matlab has a built-in function for computing the matrix exponential $\exp(A)$:

```
1  expA = expm(A);
```

Using this, compute the error of your algorithm:

```
1  err = norm(expA - expAk);
```

Note: if you don't know what a norm is, don't worry. It's just a way of measuring the error.

Now plot the error against k for the same range as in part (b) (you might want to use a log scale for the y-axis). How does the error behave? What can you say about the accuracy of your algorithm? What can you say about the robustness of your algorithm?

Hint! When writing your code, you may find the following useful:

$$\frac{1}{k!}A^k = \frac{1}{k}A \times \left(\frac{1}{(k-1)!}A^{k-1}\right).$$