Homework 1

Problem 1: Basic Matrix Operations

Define the following matrices and vectors in Matlab:

$$A = \begin{pmatrix} 1 & 2 \\ -3 & 1 \end{pmatrix}, \; B = \begin{pmatrix} 5 & 3 \\ -1 & 3 \end{pmatrix}, \; C = \begin{pmatrix} 1 & 2 & 3 \\ -9 & 2 & 4 \end{pmatrix}, \; D = \begin{pmatrix} 1 & -2 \\ -1 & 3 \\ 2 & 3 \end{pmatrix},$$

$$\mathbf{x} = \begin{pmatrix} -2 \\ 0 \end{pmatrix}, \; \mathbf{y} = \begin{pmatrix} 0 \\ 3 \end{pmatrix} \text{ and } \mathbf{z} = \begin{pmatrix} 2 \\ -3 \\ 1 \end{pmatrix}.$$

Calculate the following, and save them in nine separate `.dat` files named `A1.dat` - `A9.dat`, respectively.

(a) $B - A$,

(b) $-2\mathbf{x} + 4\mathbf{y}$,

(c) $B\mathbf{x}$,

(d) $A(\mathbf{x} - \mathbf{y})$,

(e) $C\mathbf{z}$,

(f) AB,

(g) BA,

(h) AC,

(i) CD.

Are the answers in (f) and (g) the same? Why or why not? What happens if you try to calculate CA, DC or $\mathbf{x} + \mathbf{z}$?

Next, access the following elements and save them in `A10.dat` - `A12.dat`, respectively.

(j) Both columns of the first two rows of D,

(k) The second column of C,

(l) The last two columns of the first row of C.

Problem 2: Truncation Errors

The following four expressions are exactly equal to zero:

$$x_1 = \left|80,000 - \sum_{k=1}^{800,000} 0.1\right|, x_2 = \left|80,000 - \sum_{k=1}^{640,000} 0.125\right|,$$
$$x_3 = \left|80,000 - \sum_{k=1}^{400,000} 0.2\right|, x_4 = \left|80,000 - \sum_{k=1}^{320,000} 0.25\right|.$$

However, Matlab stores numbers with a binary representation and only finitely many digits (typically 52 or 53). This means that most values with a fractional part have a small truncation error. For instance, the number 0.1 is not stored exactly in Matlab (or almost any other programming language). This error is generally so small that you do not notice, but if we add together many copies of 0.1, the error will accumulate and can potentially become quite large. As a result, x_1, x_2, x_3 and x_4 may or may not be *numerically* equal to zero. To verify this effect, use Matlab to compute each x value and save them in `A13.dat` - `A16.dat`, respectively. (Hint: Use a `for` loop to calculate the sums.)

Can you explain the differences in these values? Why are some exactly zero and some not?

Problem 3: The Logistic Map

The logistic map is a function that is often used to model population growth. It is defined by

$$P(t+1) = rP(t)\left(1 - \frac{P(t)}{K}\right).$$

Here, $P(t)$ represents the density of a population at year t, the parameter r is a growth rate and the parameter K is the maximum possible population density (known as the carrying capacity). This equation says that if we know the density at one year, we can substitute it into the right-hand side to find the population at the next year. For instance, if we knew the population density at year 1 (given by $P(1)$), we could then calculate

$$P(2) = rP(1)\left(1 - \frac{P(1)}{K}\right).$$

Once we had the population density at year 2, we could then find the density at year 3 using

$$P(3) = rP(2)\left(1 - \frac{P(2)}{K}\right).$$

We can continue in this manner for as many steps as we want.

Suppose that $r = 3$, $K = 20$ and $P(1) = 10$. Find $P(100)$ and save it in `A17.dat`.

You don't need to do anything else for the homework, but I recommend that you explore the logistic equation in Matlab on your own. Can you save $P(1), \ldots, P(100)$ in a vector? Can you plot the population density over time? What happens if you change r or K or $P(0)$? What sorts of behavior can you get? Is this behavior sensitive to small changes in the parameters? (Note: The logistic map is usually only defined for $0 \leq r \leq 4$, which ensures that $P(t)$ will never be negative.)