# Assignment 3

**5 June 2020**

**MAM2046W 2NA KDSMIL001**

1. **Deriving a second derivative approximation formula**
   We are looking for an equation that approximates the second derivative and is of the form

$$f''(x) = \lambda_1 f(x) + \lambda_2 f(x+h) + \lambda_3 f(x-h) + \lambda_4 f(x+2h) + \lambda_5 f(x-2h) \tag{1}$$

We can start by finding the Taylor expansions of each function

$$\begin{aligned}
f(x) &= f(x) \\
f(x+h) &= f(x) + hf'(x) + \frac{h^2}{2!}f''(x) + \frac{h^3}{3!}f'''(x) + \frac{h^4}{4!}f^{(4)}(x) + \mathcal{O}(h^5) \\
f(x-h) &= f(x) - hf'(x) + \frac{h^2}{2!}f''(x) - \frac{h^3}{3!}f'''(x) + \frac{h^4}{4!}f^{(4)}(x) - \mathcal{O}(h^5) \\
f(x+2h) &= f(x) + 2hf'(x) + 2hf''(x) + \frac{8h^3}{3!}f'''(x) + \frac{16h^4}{4!}f^{(4)}(x) + \mathcal{O}(h^5) \\
f(x-2h) &= f(x) - 2hf'(x) + 2hf''(x) - \frac{8h^3}{3!}f'''(x) + \frac{16h^4}{4!}f^{(4)}(x) - \mathcal{O}(h^5)
\end{aligned}$$

and then, after collecting coefficients of derivatives of f, Equation 1 looks like this, ignoring the error term

$$\begin{aligned}
f''(x) = (\lambda_1 + \lambda_2 + \lambda_3 + \lambda_4 + \lambda_5)f(x) + (\lambda_2 - \lambda_3 + 2\lambda_4 - 2\lambda_5)hf'(x) \\
+(\lambda_2 + \lambda_3 + 4\lambda_4 + 4\lambda_5)h^2 f''(x) + (\lambda_2 - \lambda_3 + 8\lambda_4 - 8\lambda_5)h^3 f'''(x) \\
+(\lambda_2 + \lambda_3 + 16\lambda_4 + 16\lambda_5)h^4 f^{(4)}(x)
\end{aligned} \tag{2}$$

Now we can equate coefficients of derivatives on either side of the equation, which we can turn into a matrix equation:

$$\begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & -1 & 2 & -2 \\ 0 & 1 & 1 & 4 & 4 \\ 0 & 1 & -1 & 8 & -8 \\ 0 & 1 & 1 & 16 & 16 \end{pmatrix} \begin{pmatrix} \lambda_1 \\ \lambda_2 \\ \lambda_3 \\ \lambda_4 \\ \lambda_5 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \frac{2}{h^2} \\ 0 \\ 0 \end{pmatrix}$$

Solving this equation (finding the inverse of the $5 \times 5$ matrix) gives us values for our $\lambda$'s, which are:

$$\lambda_1 = \frac{-30}{12h^2};\ \ \lambda_2 = \frac{16}{12h^2};\ \ \lambda_3 = \frac{16}{12h^2};\ \ \lambda_4 = \frac{-1}{12h^2};\ \ \lambda_5 = \frac{-1}{12h^2}$$

Finally, we have Equation 1 becoming

$$\begin{aligned} f''(x) &= \frac{-30f(x) + 16f(x+h) + 16f(x-h) - f(x+2h) - f(x-2h)}{12h^2} \\ &= \frac{-f(x-2h) + 16f(x-h) - 30f(x) + 16f(x+h) - f(x+2h)}{12h^2} \end{aligned} \tag{3}$$

2. **Evaluating an integral with the Composite Trapezoid Method**
   We aim to evaluate the following integral using the composite trapezoid method, using n equal subintervals

$$I = \int_a^b e^x dx \tag{4}$$

Firstly, we can look at the general form of the trapezoid method for approximating an integral:

$$\int_a^b f(x)dx = \underbrace{\frac{h}{2}\sum_{i=0}^{n-1}[f(x_i) + f(x_{i+1})]}_{\text{approximation}} - \underbrace{\frac{h^2}{12}(b-a)f''(c)}_{\text{error}} \tag{5}$$

where $n$ is the number of equal subintervals between $a$ and $b$, $h = (a-b)/n$, and $c \in [a, b]$. $c$ is arbitrary but we choose it to be the point which maximises the function on the given interval in order to look at the "worst case scenario", if you will. Applying Equation 5 to Equation 4, and ignoring the error term for now, we get

$$\begin{aligned} I &\approx \frac{h}{2}\sum_{i=0}^{n-1}[e^{x_i} + e^{x_{i+1}}] \\ &= \frac{h}{2}\left[\sum_{i=0}^{n-1} e^{x_i} + \sum_{i=0}^{n-1} e^{x_{i+1}}\right] \end{aligned}$$

These sums can be converted into 2 geometric series as each has a common ratio of $e^h$, therefore we get

$$I \approx \frac{h(1-e^{hn})}{2(1-e^h)}[e^a + e^{a+h}] \tag{6}$$

Of course, this is only an approximation as we haven't included the error term yet. In this error term though, there is $f''(c)$, where $c$ is the the point which maximises $f$. In this case, $c = b$ as $e^x$ is strictly increasing everywhere. So finally we have

$$I = \frac{h(1 - e^{hn})}{2(1 - e^h)}[e^a + e^{a+h}] - \frac{h^2}{12}(b - a)f''(b) \tag{7}$$

3. **Implementing the Composite Midpoint Method**
   The composite midpoint method has the general form of

$$\int_a^b f(x)dx = \underbrace{h\sum_{i=0}^{n-1} f(w_i)}_{\text{approximation}} + \underbrace{\frac{h^2(b-a)}{24}f''(c)}_{\text{error}} \tag{8}$$

where $h = (b - a)/n$, $w_i = a + (i + \frac{1}{2})h$, and $c \in [a, b]$. Again, we choose c to be the value which maximises $f$, to get a worst case scenario for the error. We were to use this method to estimate

$$\int_0^1 \frac{4}{1 + x^2}dx = \pi \tag{9}$$

with $n = 1000$. Below is our implementation of Equation 8 applied to Equation 9 in Python.

```python
import numpy as np
# Setting constants to allow for modification if need be
a = 0
b = 1
n = 1000
h = (b-a)/n
xs = np.linspace(a, b, n)
I = 0
# Looping through the summation, setting a new w at each iteration
for i in xs:
    w = a+(i+(1/2))*h
    I += 4/(1+(i**2))
# Multiplying by h as per the formula
I *= h
print(I)
```

Composite Midpoint Method

When we compute the value of the error term, using $c = 0$ as that maximises the function, we get a value of $\pi = 3.141450894 \pm 3.33 \times 10^{-7}$. This doesn't agree with the known value of $\pi$ but it's quite close.