APPM 4600
Homework 2

January 30, 2025

Jude Gogolewski

---

1 O notation

(a) Show that $(1+x)^n = 1 + nx + o(x)$ as $x \to 0$

to start, we'll Taylor expand $(1+x)^n$

$$(1+x)^n = 1 + n(x) + \frac{n(n-1)}{2}(x^2) + \frac{n(n-1)(n-2)}{3!}(x^3)$$

If we now get rid of $1+nx$ and look solely at the remaining terms and take the limit $x \to 0$ compared to our $0(x)$:

$$\lim_{x\to 0}\left(\frac{n(n-1)}{2}(x^2) + \frac{n(n-1)(n-2)}{3!}(x^3)\right)/x =$$

$$\lim_{x\to 0}\left(\frac{n(n-1)}{2}(x) + \frac{n(n-1)(n-2)}{3!}(x^2)\right) = 0$$

showing that

$$(1+x)^n = 1 + nx + o(x)$$

(b) Show that $xsin(\sqrt{x}) = O(x^{(3/2)})$ as $x \to 0$ Let us start by setting up a Taylor series

for the simpler expression $sin(\sqrt{x})$:

$$sin(\sqrt{x}) = \sqrt{x} - \frac{(\sqrt{x})^3}{3!} + \frac{(\sqrt{x})^5}{5!} - ... \approx x$$

The series above shows that for small values of x, $sinx \approx x$. If we now apply x to our series, we see:

$$xsin(\sqrt{x}) = x\sqrt{x} - x\frac{(\sqrt{x})^3}{3!} + x\frac{(\sqrt{x})^5}{5!} - ... \approx x^{3/2}$$

now as we can see $xsin(\sqrt{x}) \approx x^{3/2}$ or $xsin(\sqrt{x}) = O(x^{3/2})$. In this case, our approximation is big O because, as for small values of x, the remaining terms die off faster than $Cx^{3/2}$, $C \geq 0$.

(c) Show that $e^{-t} = o(\frac{1}{t^2})$ as $t \to \infty$. to prove this, I will set up the limit for this system

and view its tendency as $t \to \infty$

$$\lim_{t\to\infty} \frac{e^{-t}}{1/t^2} = \lim_{t\to\infty} \frac{t^2}{e^t} = \frac{\infty}{\infty}$$

because this equation tends to infinity over infinity, we can use L'Hospitals rule twice to get:

$$\lim_{t\to\infty} \frac{t^2}{e^t} = \lim_{t\to\infty} \frac{2}{e^t} = 0$$

because the limit tends to 0 as $t \to \infty$ we can say e^{-t} is $o(1/t^2)$

(d) Show that $\int_0^\epsilon e^{-x^2} dx = O(\epsilon)$ as $\epsilon \to 0$ to start, I want to recognize that One e^{-x^2}

is not a function we can integrate. This is the equation for a normal distribution. And Two $e^{-x^2} \leq 1$ for all x. Because of that fact, we can set up a relation:

$$\int_0^\epsilon e^{-x^2} dx \leq \int_0^\epsilon 1 dx = \epsilon$$
$$\int_0^\epsilon e^{-x^2} dx = O(\epsilon)$$

2 $\boldsymbol{A}x = \boldsymbol{b}$

$$A = \frac{1}{2}\begin{bmatrix} 1 & 1 \\ 1+10^{-10} & 1-1^{-10} \end{bmatrix}, b = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$
$$A^{-1} = \begin{bmatrix} 1-10^{10} & 10^{10} \\ 1+10^{10} & -10^{10} \end{bmatrix}, x = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

(a) To start we know; $\Delta x = A^{-1}\Delta b$
if we now sub in and simplify, we get

$$\Delta x = A^{-1}\Delta b = \begin{bmatrix} 1-10^{10} & 10^{10} \\ 1+10^{10} & -10^{10} \end{bmatrix} \begin{bmatrix} \Delta b_1 \\ \Delta b_2 \end{bmatrix} = \begin{bmatrix} (1-10^{10})\Delta b_1 + 10^{10}\Delta b_2 \\ (1+10^{10})\Delta b_1 - 10^{10}\Delta b_2 \end{bmatrix}$$

(b) The Condition number of A is equivalent to the Euclidean norm of A times the Euclidean norm of A^{-1}:
$\kappa(A) = ||A||_2||A^{-1}||_2 = |\frac{1}{2}| \left\| \begin{matrix} 1 & 1 \\ 1+10^{-10} & 1-1^{-10} \end{matrix} \right\| \left\| \begin{matrix} 1-10^{10} & 10^{10} \\ 1+10^{10} & -10^{10} \end{matrix} \right\| = 20000000000.000004 \approx 2 * 10^{10}$

(c) $|\Delta b_1| = |\Delta b_2| \leq 10^{-5}$
using what we got in part a, let's find the relative error in the solution:
$||\Delta x|| = \sqrt{x_1^2 + x_2^2} = \sqrt{((1-10^{10})\Delta b_1 + 10^{10}\Delta b_2)^2 + ((1+10^{10})\Delta b_1 - 10^{10}\Delta b_2)^2}$
$\approx 10^{10}\sqrt{\Delta b_1^2 + \Delta b_2^2}$
$||x|| = \sqrt{2}$

$$\frac{||\Delta x||}{||x||} \approx \frac{10^{10}\sqrt{\Delta b_1^2 + \Delta b_2^2}}{\sqrt{2}}$$

if we now sub in for our values of $|\Delta b_1|$ & $|\Delta b_2|$ to get an upper bound we get:

$$\frac{||\Delta x||}{||x||} \leq \frac{10^{10}\sqrt{(10^{-5})^2 + (10^{-5})^2}}{\sqrt{2}} = 10^5$$

The relative error in the solutions is related to the perturbations in the input times

and the condition number for the operation.

$$e_{rell} = \frac{||\Delta x||}{||x||} = \kappa(A)\frac{||\Delta b||}{||b||} = (2 * 10^{10})(10^{-5}) = 2 * 10^{10}$$

if $\Delta b_1 = \Delta b_2$ there is a possibility that the error on Δx is minimized but that is not guaranteed. However, the operation will compound the error if $\Delta b_1 \neq \Delta b_2$. The latter is far more probable, as is the example of an experiment; if the former is true, we are most likely experiencing faulty equipment. As the latter is true, there is a wider berth of possible causes.

3 Let $f(x) = e^x - 1$

(a) The Relative condition number of $f(x)$:
$\kappa(f(x)) = \frac{xf'(x)}{f(x)} = \frac{xe^x}{e^x-1} = \frac{x}{1-e^{-x}}$

For this new function, we see the function is poorly conditioned at $x = 0$ as we end up with an undefined value. However, if we apply L'Hopitals rule, we see that: $\lim_{x\to 0} = 1$, which is ideal for a condition number.

(b) for the algorithm:

f(x):
$y = math.e^x$
return y - 1

This algorithm is unstable because it ignores the case where $x \approx 0$ in which we will have cancellation of terms and lose accuracy because we sacrifice precision by subtracting two numbers of relative magnitude.

(c) testing values of x:

let $x = 9.99999999500000010^{-10}$
True value: $f(x) = 10^{-9}$
computed value: $f(x) = 1.000000082740371e - 09 \approx 10^{-9}$

This value is not entirely unexpected since $y = f(x) \approx 1; x \approx 0$ ergo, we will have a subtraction of values with similar size, resulting in some cancelation of terms. This appears in the excess values of $f(x)$.

(d) Find a polynomial that will gain us 16 digits of accuracy for $f(x)$:
To start lets Taylor expand $f(x)$

$$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \frac{x^4}{4!}... \tag{1}$$
$$e^x - 1 = x + \frac{x^2}{2!} + \frac{x^3}{3!} + \frac{x^4}{4!}... \tag{2}$$

From this, we see that if we use our second line, we can continually add terms until we reach 16 digits of accuracy, as our new polynomial has no subtraction and, in turn, no cancellation of terms.

(e) Check answer to part (c):
I plugged our new polynomial into Python, tested our value, and received 16 digits of accuracy. Please check git hub.

(f) Simpler Taylor series:
After using two different Taylor series, one as a 10th-degree polynomial and the other as a 4th-degree polynomial, and received 16 digits of accuracy both times.

4 Practicing Python

(a) To accomplish part (a) and find the value S, we can accomplish this two ways:
1: using an integrated package from numpy [sum]
2: using a for loop
I used both and received the same value to different levels of accuracy:
Method 1: -23.915381134014112
Method 2: -23.91538113401411

(b) Plotting Wavy Curves:
Below, I have added to the plots. To view all the code used in this assignment, refer to Git Hub: