APPM 4600 Homework 2

14 September 2024

1. (a) Consider the function

$$f(x) = (1+x)^n - 1 - nx.$$

Observe that

$$\lim_{x\to 0}\left|\frac{f(x)}{x}\right| = \lim_{x\to 0}\left|\frac{(1+x)^n - 1 - nx}{x}\right| = \lim_{x\to 0}\left|n(1+x)^{n-1} - n\right| = 0.$$

Thus, $f(x) = o(x)$ as $x \to 0$, which implies that

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

(b) Observe that

$$\lim_{x\to 0}\left|\frac{x\sin\sqrt{x}}{x^{\frac{3}{2}}}\right| = \lim_{x\to 0}\left|\frac{\sin\sqrt{x} + \frac{1}{2}\sqrt{x}\cos\sqrt{x}}{\frac{3}{2}\sqrt{x}}\right| = \lim_{x\to 0}\left|\frac{\frac{3}{4\sqrt{x}}\cos\sqrt{x} - \frac{1}{4}\sin\sqrt{x}}{\frac{3}{4\sqrt{x}}}\right| = \lim_{x\to 0}\left|\cos\sqrt{x} - \frac{\sqrt{x}}{3}\sin\sqrt{x}\right|$$
$$= 1 \neq 0.$$

Thus, as $x \to \infty$,

$$x\sin\sqrt{x} = O\left(x^{\frac{3}{2}}\right).$$

(c) Observe that

$$\lim_{t\to\infty}\left|\frac{e^{-t}}{\frac{1}{t^2}}\right| = \lim_{t\to\infty}\left|\frac{t^2}{e^t}\right| = \lim_{t\to\infty}\left|\frac{2t}{e^t}\right| = \lim_{t\to\infty}\left|\frac{2}{e^t}\right| = 0.$$

Thus, as $t \to \infty$,

$$e^{-t} = o\left(\frac{1}{t^2}\right).$$

(d) Observe that

$$\lim_{\epsilon\to 0}\left|\frac{\int_0^\epsilon e^{-x^2}\,\mathrm{d}x}{\epsilon}\right| = \lim_{\epsilon\to 0}\left|\frac{\mathrm{d}}{\mathrm{d}x}\int_0^\epsilon e^{-x^2}\,\mathrm{d}x\right| = \lim_{\epsilon\to 0}\left|e^{-\epsilon^2}\right| = 1 \neq 0.$$

Thus, as $\epsilon \to 0$,

$$\int_0^\epsilon e^{-x^2}\,\mathrm{d}x = O(\epsilon).$$

2. (a) We have the exact problem $A\mathbf{x} = \mathbf{b}$ with solution $\mathbf{x} = A^{-1}\mathbf{b}$. The perturbed problem is $A\mathbf{x}^* = (\mathbf{b} + \mathbf{\Delta b})$ with solution

$$\mathbf{x}^* = A^{-1}(\mathbf{b} + \mathbf{\Delta b}).$$

The change in solution is

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

(b) The condition number of A is bounded as

$$\kappa(A) \leq \frac{\sigma_1}{\sigma_n},$$

where σ_1 and σ_2 are the singular values of A. We have that A has singular value decomposition

$$A = \begin{bmatrix} 10^{-10} & 1 \\ -1 & 10^{-10} \end{bmatrix} \begin{bmatrix} 5 \times_1 0^9 & 0 \\ 0 & \frac{1}{2} \end{bmatrix} \begin{bmatrix} 0 & 1 \\ 1 & 10^{-10} \end{bmatrix},$$

so we have $\sigma_1 = 5 \times 10^9$ and $\sigma_2 = \frac{1}{2}$ and the condition number is bound as

$$\kappa(A) \leq \frac{\sigma_1}{\sigma_2} = \frac{5 \times_1 0^9}{\frac{1}{2}} = 10^{10},$$

which suggests that the problem may be poorly conditioned.

(c) The relative error in the solution is $||\mathbf{\Delta x}||/||\mathbf{x}||$. The condition number provides an upper bound for the relative error in the solution relative to the relative perturbation, that is,

$$\kappa = \lim_{\delta \to 0} \sup_{||\mathbf{\Delta b}|| < \delta} \frac{\frac{||\mathbf{\Delta x}||}{||\mathbf{x}||}}{\frac{||\mathbf{\Delta b}||}{\mathbf{b}}}.$$

From the absolute error (1), we known that the error is dominated by the $\Delta b_1 - \Delta b_2$ term. The error will be larger if the perturbations in each dimension are not matched. The relative error for various perturbations is given below.

Perturbation	Relative Error
$\Delta b_1 = 10^{-5}, \Delta b_2 = 10^{-5}$	1.002×10^{-5}
$\Delta b_1 = 2 \times 10^{-5}, \Delta b_2 = 0.7 \times 10^{-5}$	1.3×10^5

Notice that the case with unmatched perturbations has large error. In general, we cannot expect the perturbation terms to be equal, so we should expect the system to be very poorly conditioned.

3. (a) The problem is $f(x) = e^x - 1$. We consider a perturbation δx and examine the perturbed solution δf. It has relative condition number

$$\kappa = \lim_{\delta \to 0} \frac{\frac{||\delta f||}{||f||}}{\frac{||\delta x||}{||x||}} = \lim_{\delta \to 0} \frac{||\delta f||}{||\delta x||} \frac{||x||}{||f||} = e^x \left(\frac{x}{e^x - 1} \right) = \frac{xe^x}{e^x - 1}.$$

Thus, this problem is ill conditioned for large values of x, where κ will grow large.

(b) This algorithm is not stable for small values of x, where cancellation will occur between the e^x and 1 terms and result in a loss of precision.

(c) Computation with the algorithm above gives $1.000000082740371 \times 10^{-9}$, which is correct only for 8 digits. This is expected, since the algorithm involves a large cancellation.

(d) We Taylor expand e^x and find

$$f(x) \approx x + \frac{x^2}{2!} + \frac{x^3}{3!} + \dots.$$

To compute f(x) accurately to 16 digits, we only need the first two terms,

$$f(x) = x + \frac{x^2}{2}.$$

(e) Observe that

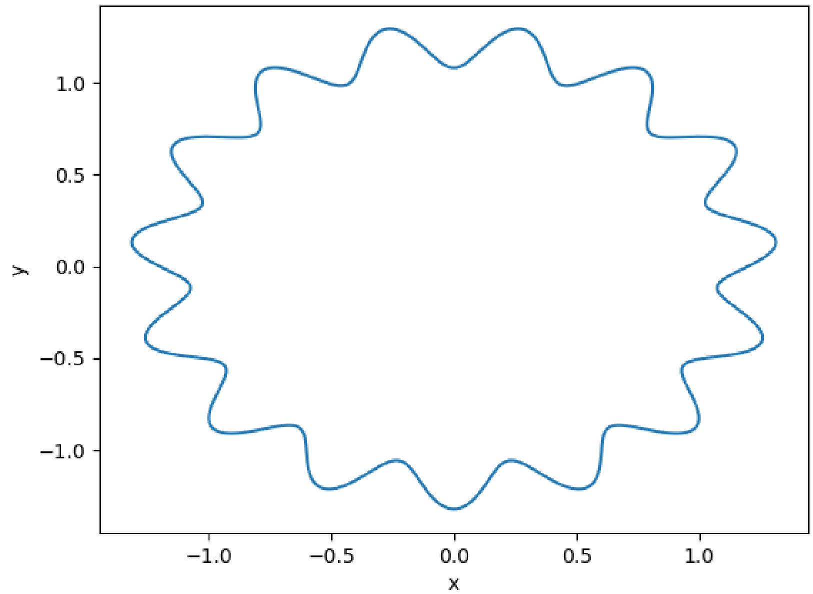
$$f(9.999999995000000 \times 10^{-10}) \approx 1.00000000000000006228159146 \times 10^{-09},$$

which is accurate to 16 digits.

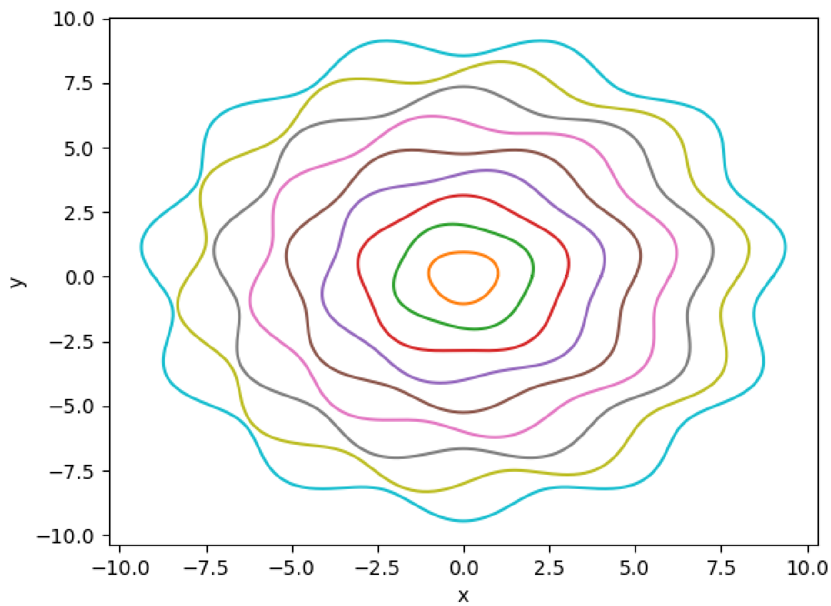
4. (a) The code for this question is listed in the appendix. For t over 0 to π inclusive with steps $\frac{\pi}{30}$, we get

$$S \approx -20.686852364346837$$

(b) The plot of the curve for $R = 1.2, \delta r = 0.1, f = 15, p = 0$ is shown below.



The plot of the curve with variable parameters is shown below.



A. Codes

Question 2. The code used to compute the results in the table in 2(c) is listed below.

```python
#! /usr/bin/env python3
import numpy as np

# pertubations
b1 = 2e-5
b2 = 0.7e-5

# b1 = 1e-5
# b2 = 1e-5

# setup system to solve
Ainv = np.array([[1-10**10, 10**10], [1+10**10, -10**10]])
b = np.array([[1], [1]])
bpert = np.array([[b1], [b2]])
# compute exact solution
x = np.matmul(Ainv, b)
# compute perturbed solution
xpert = np.matmul(Ainv, b+bpert)
# absolute error
xdif = x - xpert
# relative error
rel_error = np.linalg.norm(xdif) / np.linalg.norm(x)

print(rel_error)
```

Question 3. The code used to compute the results in problem (3) is listed below.

```python
#! /usr/bin/env python3
import numpy as np
import math

x = 9.999999995000000e-10

# naive computation
y = math.e ** x - 1
print("naive y: ", y)

y_taylor = x + x**2/2
print("taylor y: ", f'{y_taylor:.27}')
```

Question 4. The code used to compute the results in problem (4) is listed below.

```python
#! /usr/bin/env python3
import numpy as np
import math
import matplotlib.pyplot as plt
import random

# part (a)
t = np.arange(0, math.pi+1e-10, math.pi/30)
y = np.cos(t)

# compute the sum of elementwise multiplication of terms
S = np.sum(np.multiply(t, y))
print("the sum is:", S)

# part (b)
def plot_b_parametric(theta, R, dr, f, p):
    x = R*(1 + dr*np.sin(f*theta))*np.cos(theta)
    y = R*(1 + dr*np.sin(f*theta))*np.sin(theta)
    plt.plot(x, y)
    plt.xlabel("x")
```

```
    plt.ylabel("y")

theta = np.arange(0, 2*math.pi, 1e-2)
plot_b_parametric(theta, 1.2, 0.1, 15, 0)
plt.savefig("hw2_3_b1.png")
plt.close()

for i in range(10):
    R = i
    dr = 0.05
    f = 2 + i
    p = random.uniform(0, 2)
    plot_b_parametric(theta, R, dr, f, p)

plt.savefig("hw2_3_b2.png")
```