Homework 1

Jaden Snell

January 26, 2024

Question 1

1. Consider the polynomial
$p(x) = (x-2)^9 = x^9 - 18x^8 + 144x^7 - 672x^6 + 2016x^5 - 4032x^4 + 5376x^3 - 4608x^2 + 2304x - 512.$

   i. Plot $p(x)$ for $x = 1.920, 1.921, 1.922, \ldots, 2.080$ (i.e. $x = [1.920 : 0.001 : 2.080];$) evaluating p via its coefficients.

   ii. Produce the same plot again, now evaluating p via the expression $(x-2)^9$.

   iii. What is the difference? What is causing the discrepancy? Which plot is correct?

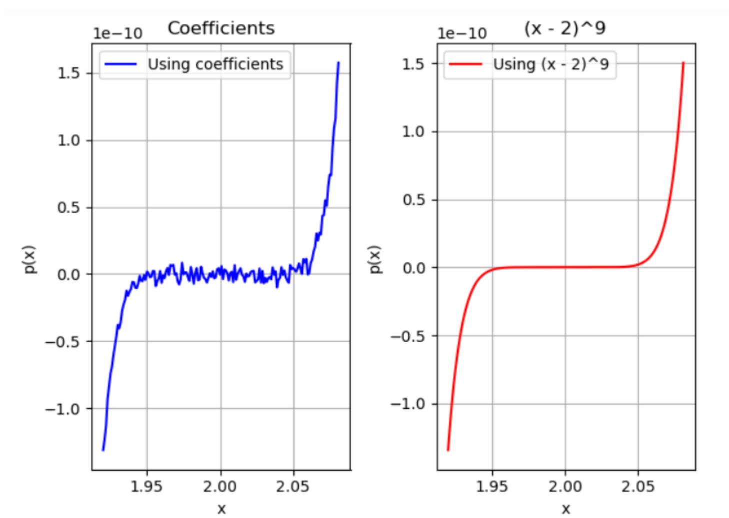

Figure 1: Floating point arithmetic errors and the factored form.

The difference is the amount of noise in the first graph compared to the second. This is likely caused due to floating-point arithmetic errors due to the sheer number of calculations required to calculating the polynomial. The second polynomial is in factored form requiring less operations giving less floating-point errors thus the less noisy graph. The second plot is the correct plot.

Question 2

2. How would you perform the following calculations to avoid cancellation? Justify your answers.

   i. Evaluate $\sqrt{x+1} - 1$ for $x \simeq 0$.

   ii. Evaluate $\sin(x) - \sin(y)$ for $x \simeq y$.

   iii. Evaluate $\frac{1-\cos(x)}{\sin(x)}$ for $x \simeq 0$.

i. To evaluate $sqrt(x+1)-1$ for $x \simeq 0$ without cancellation, I would multiply by the conjugate yielding $(sqrt(x+1)-1) \cdot \frac{sqrt(x+1)+1}{sqrt(x+1)+1} = \frac{x}{sqrt(x+1)+1}$ which does not contain subtraction thus avoids cancellation as $x \simeq 0$ does not cause underflow/overflow issues with the division.

ii. To evaluate $\sin(x) - \sin(y)$, I would use the trigonometric identity for the difference of sines yielding $2\cos(\frac{x+y}{2})\sin(\frac{x-y}{2})$. Despite the $\frac{x-y}{2}$ in the sine, this avoids cancellation because $\sin(z) \approx z$ for small z, i.e it's stable.

iii. I would utilize another trigonometric identity that $\sin^2(x) = 1 - \cos^2(x)$. To achieve this, it's necessary to multiply the original equation by the conjugate yielding $\frac{\sin^2(x)}{\sin(x)(1+\cos(x))} = \frac{\sin(x)}{1+\cos(x)}$. This avoids cancellation because the equation no longer contains dividing by a number close to 0.

Question 3

3. Find the second degree Taylor polynomial $P_2(x)$ for $f(x) = (1 + x + x^3)\cos(x)$ about $x_0 = 0$.

   (a) Use $P_2(0.5)$ to approximate $f(0.5)$. Find an upper bound for the error $|f(0.5) - P_2(0.5)|$ using the error formula and compare it to the actual error.

   (b) Find a bound for the error $|f(x) - P_2(x)|$ when $P_2(x)$ is used to approximate $f(x)$. This will be a function of x.

   (c) Approximate $\int_0^1 f(x)dx$ using $\int_0^1 P_2(x)dx$.

   (d) Estimate the error in the integral.

The second degree Taylor polynomial of the equation, $P_2(x)$ for $f(x) = (1 + x + x^3)cos(x)$ about $x_0 = 0$ can be found through:

$$\begin{aligned}
P_2(x) &= f(0) + f'(0)x + \frac{f"(0)x^2}{2} \\
&= (1 + 0 + 0^3)\cos(0) + ((3(0)^2 + 1)\cos(0) - (0^3 + 0 + 1)\sin(0))x + \frac{f"(0)x^2}{2} \\
&= 1 + x + \frac{(-6(0)^2 - 2) * \sin(0) + (-(0)^3 + 5(0) - 1)\cos(0)x^2}{2} \\
&= 1 + x - \frac{x^2}{2}
\end{aligned}$$

(a) $P_2(0.5) = 1+0.5-\frac{0.5^2}{2} = \frac{11}{8}$ and the upper bound calculation is $|R_2(0.5)| = |\frac{f^{(3)}(0)}{3!}0.5^3| = \frac{1}{16} = 0.0625$. Meanwhile the actual error is approximately 0.05107 which makes sense because the upper bound, $0.0625 > 0.05107$.

(b) The generalized error bound calculation is $|R_2(x)| = |\frac{f^{(3)}(0)}{3!}x^3| = |\frac{x^3}{2}|$.

(c) We can approximate $\int_0^1 f(x)\,dx$ using

$$\begin{aligned}
\int_0^1 P_2(x)\,dx &= \int_0^1 (1 + x - \frac{x^2}{2})\,dx \\
&= \frac{4}{3}
\end{aligned}$$

(d) The error in the integral can be estimated through $\int_0^1 R_2(x)\,dx = \int_0^1 \frac{x^3}{2}\,dx = \frac{1}{8}$.

Question 4

4. Consider the quadratic equation $ax^2 + bx + c = 0$ with $a = 1, b = -56, c = 1$.

   (a) Assume you can calculate the square root with 3 correct decimals (e.g. $\sqrt{(2)} \approx 1.414 \pm \frac{1}{2}10^{-3}$) and compute the relative errors for the two roots to the quadratic when computed using the standard formula

   $$r_{1,2} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}.$$

   (b) A better approximation for the "bad" root can be found by manipulating $(x - r_1)(x - r_2) = 0$ so that r_1 and r_2 can be related to a, b, c. Find such relations (there are two) and see if either can be used to compute the "bad" root more accurately.

(a) Using the quadratic formula for the equation given above yield $\frac{56 \pm \sqrt{56^2 - 4}}{2} = \frac{56 \pm \sqrt{3132}}{2} = \frac{56 \pm 55.964}{2}$. The solutions can be given as $x_1 = 55.982$, $x_2 = 0.018$. The relative error for x_1 is $|\frac{x - fl(x)}{x}| = |\frac{55.982137159266 - 55.982}{55.982137159266}| \approx 2 \times 10^{-6}$. The relative error for x_2 is $|\frac{x - fl(x)}{x}| = |\frac{0.017862840733555 - 0.018}{0.017862840733555}| \approx 8 \times 10^{-3}$.

(b) An alternative method of approximation utilizes the relationship $r_1 * r_2 = \frac{c}{a}$, meaning that in this case, $r_2 = 1/r_1 = 1/55.982 = 0.01786288$ giving a much more accurate measurement. Another method of approximation can be found through the identity $r_1 + r_2 = \frac{-b}{a}$ meaning $r_2 = \frac{-b}{a} - r_1 = 56 - r_1$. This method doesn't actually increase the accuracy when computing the bad root.

Question 5

5. **Cancellation of terms.** Consider computing $y = x_1 - x_2$ with $\tilde{x}_1 = x_1 + \Delta x_1$ and $\tilde{x}_2 = x_2 + \Delta x_2$ being approximations to the exact values. If the operation $x_1 - x_2$ is carried out exactly we have $\tilde{y} = y + \underbrace{(\Delta x_1 - \Delta x_2)}_{\Delta y}$.

Play with different values of x. One really small value (< 1) and one large value $> 10^5$.

(a) Find upper bounds on the absolute error $|\Delta y|$ and the relative error $|\Delta y|/|y|$, when is the relative error large?

(b) First manipulate $\cos(x+\delta) - \cos(x)$ into an expression without subtraction. Pick two values of x; say $x = \pi$ and $x = 10^6$. Then for each x, tabulate or plot the difference between your expression and $\cos(x+\delta) - \cos(x)$ for $\delta = 10^{-16}, 10^{-15}, \cdots, 10^{-2}, 10^{-1}, 10^0$ (note that you can use your logx command to uniformly distribute δ on the x-axis).

(c) Taylor expansion yields $f(x+\delta) - f(x) = \delta f'(x) + \frac{\delta^2}{2!} f''(\xi)$, $\xi \in [x, x+\delta]$. Use this expression to create your own algorithm for approximating $\cos(x+\delta) - \cos(x)$. Explain why you chose the algorithm. Then compare the approximation from your algorithm with the techniques in part (b). Use the same values for x and δ.

(a) The absolute error can be derived through $|\Delta y| = |\Delta x_1 - \Delta x_2|$ and the relative error uses the absolute error and divides by the absolute value of y: $|\frac{(\Delta x_1 - \Delta x_2)}{x_1 - x_2}|$. The relative error will become large when x_1 and x_2 are very close together as this causes the denominator to be very small.

(b) To manipulate cosine to avoid subtraction, trigonometric sum and difference identities can be used: $\cos(x+\delta) - \cos(x) = -2\sin(x + \frac{\delta}{2})\sin(\frac{\delta}{2})$

(c) Using Taylor expansion, $\cos(x+\delta) - \cos(x) \approx -\delta\sin(x) - \delta^2\frac{\cos(\xi)}{2}$, $\xi \in [x, x+\delta]$, this algorithm will be slightly more accurate than b) because it is influenced by the change in the function's slope which b does not. The code for both a, b, and c is provided below.

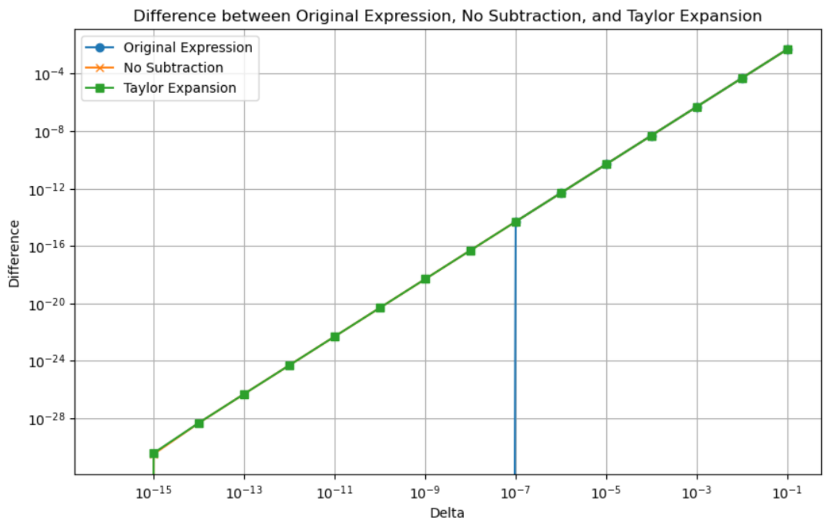

Figure 2: $x = \pi$

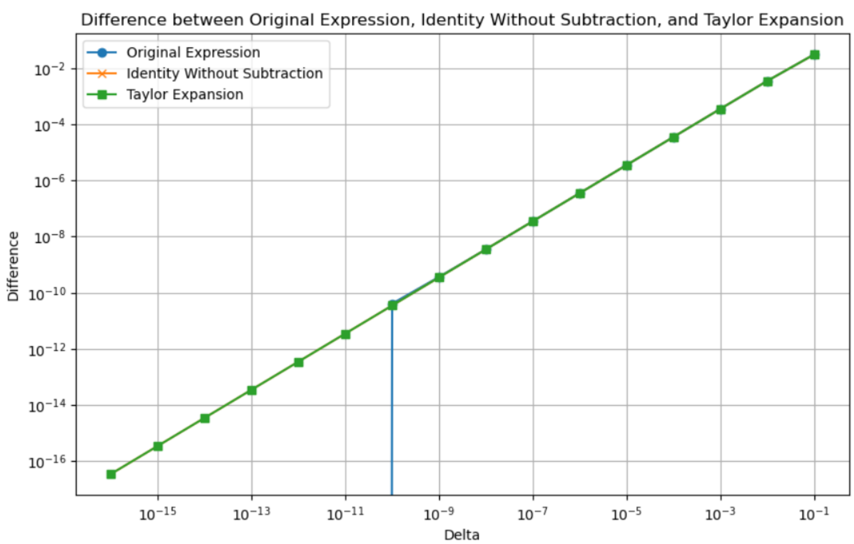

Figure 3: $x = 10^6$

Appendix: Python Code

Code for Question 1

```
import numpy as np
import matplotlib.pyplot as plt

# Problem 1
poly_coeff =
    [1,-18,144,-672,2016,-4032,5376,-4608,2304,-512]
p = np.poly1d(poly_coeff)
x = np.arange(1.920, 2.081, 0.001)

# i.
p1 = p(x)

# ii.
p2 = (x-2)**9

# Plot using coefficients
plt.subplot(1, 2, 1)
plt.plot(x, p1, label='Using coefficients', color='blue')
plt.title('Coefficients')
plt.xlabel('x')
plt.ylabel('p(x)')
plt.grid(True)
plt.legend()

# Plot using the expression (x - 2)^9
plt.subplot(1, 2, 2)
plt.plot(x, p2, label='Using (x - 2)^9', color='red')
plt.title('(x - 2)^9')
plt.xlabel('x')
plt.ylabel('p(x)')
plt.grid(True)
plt.legend()

# Show the plots
plt.tight_layout()
plt.show()
```

Listing 1: Python code for Question 1

Code for Question 5 (Small x)

```python
import numpy as np
import matplotlib.pyplot as plt
import random

def original_expression(x, delta):
    return np.cos(x + delta) - np.cos(x)

def identity_without_subtraction(x, delta):
    return -2 * np.sin(x + delta / 2) * np.sin(delta / 2)

def taylor_expansion(x, delta):
    xrand = random.uniform(x, x + delta)
    return (-delta * np.sin(x) - delta**2 * np.cos(xrand) /
        2)

x_value = np.pi
deltas = np.logspace(-16, -1, num=16, base=10)

original_values = [original_expression(x_value, d) for d in
    deltas]
identity_values = [identity_without_subtraction(x_value, d)
    for d in deltas]
taylor_values = [taylor_expansion(x_value, d) for d in
    deltas]

plt.figure(figsize=(10, 6))
plt.plot(deltas, original_values, label='Original
    Expression', marker='o')
plt.plot(deltas, identity_values, label='Identity Without
    Subtraction', marker='x')
plt.plot(deltas, taylor_values, label='Taylor Expansion',
    marker='s')
plt.xscale('log')
plt.yscale('log')
plt.xlabel('Delta')
plt.ylabel('Difference')
plt.title('Difference between Original Expression, Identity
    Without Subtraction, and Taylor Expansion')
plt.legend()
plt.grid(True)
plt.show()
```

Listing 2: Python code for Question 5 with a small x

Code for Question 5 (Big x)

```
# The code for "big x" is the same as for "small x, but x =
    1000000"
```

Listing 3: Python code for Question 5 with a big x