16-811 Homework 2

Changsheng Shen (Bobby)
changshs@andrew.cmu.edu

09/25/2019

Q1:

(a): See q1.py attached.

(b):

```
Interpolated value at x = 0.3333333333333333 is: 0.7165315560493046
Actual value at x = 0.3333333333333333 is: 0.7165313105737893
```

(c):

```
Interpolated value at x = 0.05  with n = 2 points is: 0.9976470588235294
Interpolated value at x = 0.05  with n = 4 points is: 0.9901364705882355
Interpolated value at x = 0.05  with n = 40 points is: 0.9615384615384615
Actual value at x = 0.05 is: 0.9615384615384615
```

(d):

```
Max error for n = 2 is: 0.5737522791671381
Max error for n = 4 is: 0.3853045754904466
Max error for n = 6 is: 0.48888096747915344
Max error for n = 8 is: 0.7319021780828148
Max error for n = 10 is: 1.1768940117150546
Max error for n = 12 is: 1.9694654921610955
Max error for n = 14 is: 3.381811520697178
Max error for n = 16 is: 5.913249213154253
Max error for n = 18 is: 10.479990230487914
Max error for n = 20 is: 18.768360900950597
Max error for n = 40 is: 8520.73984400868
```

We can observe that E_n increases as n increases. This makes sense because as n increases, we are using a higher order polynomial to fit the function that leads to oscillation in error near the edge of the interpolation interval, (ref: Runge's Phenomenon).

Q2: To ensure 6 decimal digit accuracy, the error should be at most $5 * 10^{-7}$.

In the case of linear interpolation:

For any $\overline{x} \in [x_i, x_{i+1}]$, $f(\overline{x})$ can be approximated with the polynomial of degree 1, given by $p_1(\overline{x})$,

Then the error $e_1(\overline{x})$ is given by:

$$e_1(\overline{x}) = \frac{f''(\xi)}{2!}(\overline{x} - x_i)(\overline{x} - x_{i+1})$$

Since $f(x) = cos(x)$, $f''(x) = -cos(x) \leq 1$

Therefore:

$$e_1(\overline{x}) = \frac{f''(\xi)}{2!}(\overline{x} - x_i)(\overline{x} - x_{i+1}) \leq \frac{1}{2!}(\overline{x} - x_i)(\overline{x} - x_{i+1})$$

Define the (uniform) interpolation interval length $x_{i+1} - x_i$ as h

Then we have:

$$(\overline{x} - x_i) \cdot (\overline{x} - x_{i+1}) \leq \frac{h}{2} \cdot \frac{h}{2} = \frac{h^2}{4}$$

Therefore:

$$e_1(\overline{x}) = \frac{f''(\xi)}{2!}(\overline{x} - x_i)(\overline{x} - x_{i+1}) \leq \frac{1}{2!} \cdot \frac{h^2}{4} = \frac{h^2}{8}$$

To ensure 6 decimal digit accuracy:

$$\frac{h^2}{8} < 5 * 10^{-7}$$

$$h < 2 * 10^{-3}$$

And since we are interpolating over the interval $[-\frac{\pi}{2}, \frac{3\pi}{2}]$, we need at least $2\pi/(2 * 10^{-3}) = 3142$ intervals (3143 points).

If we use quadratic interpolation, similarly:

$$e_2(\overline{x}) = \frac{f'''(\xi)}{3!}(\overline{x} - x_{i-1})(\overline{x} - x_i)(\overline{x} - x_{i+1})$$

Since $f(x) = cos(x)$, $f'''(x) = sin(x) \leq 1$

and consider the function $g(y) = (y - h) \cdot y \cdot (y + h)$:

$$g(-h) = g(0) = g(h) = 0$$

, therefore the function's maxima is at the point where $g'(y) = 0$. Solving the equation we get $y = \pm\frac{h}{\sqrt{3}}$

This gives

$$g(y) \leq \frac{2}{3} \cdot \frac{h^3}{\sqrt{3}}$$

Then we can get:

$$e_2(\overline{x}) = \frac{f'''(\xi)}{3!}(\overline{x} - x_{i-1})(\overline{x} - x_i)(\overline{x} - x_{i+1}) \leq \frac{1}{6} \cdot \frac{2}{3} \cdot \frac{h^3}{\sqrt{3}} = \frac{h^3}{9\sqrt{3}}$$

To ensure 6 decimal digit accuracy:

$$\frac{h^3}{9\sqrt{3}} < 5 * 10^{-7}$$

$$h < \sqrt[3]{7.7942} * 10^{-2}$$

And since we are interpolating over the interval $[-\frac{\pi}{2}, \frac{3\pi}{2}]$, we need at least $2\pi/(\sqrt[3]{7.7942} * 10^{-2}) = 317$ intervals (318 points).

Q3:

Denote $f(x) = x - tan(x)$, then $f'(x) = 1 - sec^2(x) = -tan^2(x) \leq 0$

Since $\frac{3\pi}{2} < 7 < \frac{5\pi}{2}$ we can try to evaluate the value of $f(x)$ around these points.

Let's define $\pi_- = 3.14 < \pi$ and $\pi_+ = 3.1416 > \pi$:

$$f(\frac{3\pi_-}{2}) = -413.8778226538852 < 0$$

$$f(\frac{3\pi_+}{2}) = 90751.98186832224 > 0$$

$$f(7) = 6.128552017275681 > 0$$

$$f(\frac{5\pi_-}{2}) = -413.8778226538852 < 0$$

$$f(\frac{5\pi_+}{2}) = 54456.21567531984 > 0$$

By looking at the shape of $tan(x)$, and the fact that $f'(x) = -tan^2(x) = 1 - \frac{1}{cos^2(x)} \leq 0$ and is continuous, we can say that there is only one root between $[\frac{3\pi_+}{2}, \frac{5\pi_-}{2}]$, and the first root less than 7 is less than $\frac{3\pi_-}{2}$.

Therefore, we start with $x_1 = \frac{3\pi_-}{2}$ and $x_2 = \frac{5\pi_-}{2}$, running our program (q3.py) gives:

$$root1 = 4.493409457909064$$

$$root2 = 7.7252518369377094$$

Since $root2 > 7$, therefore we find the solutions:

$$x_{low} = 4.493409457909064, x_{high} = 7.7252518369377094$$

Q4:

We can write $f(x)$ in the form of: $f(x) = (x-\xi)^2 \cdot g(x)$.

And denote

$$h(x) = \frac{f(x)}{f'(x)} = \frac{(x-\xi)^2 \cdot g(x)}{2(x-\xi)g(x) + (x-\xi)^2 g'(x)} = \frac{(x-\xi) \cdot g(x)}{2g(x) + (x-\xi)g'(x)}$$

Also we have:

$$\epsilon_n = \xi - x_n$$

$$\epsilon_{n+1} = \epsilon_n + h(\xi - \epsilon_n)$$

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

Then we evaluate

$$\lim_{n\to\infty} \frac{\epsilon_{n+1}}{\epsilon_n} = \lim_{n\to\infty} \frac{h(\xi-\epsilon_n)}{\epsilon_n} + 1 = \lim_{x\to\xi} \frac{h(x)}{\xi - x} + 1 = 1 - \lim_{x\to\xi} \frac{h(x)}{x-\xi} = 1 - \lim_{x\to\xi} \frac{g(x)}{2g(x)} = \frac{1}{2}$$

Therefore, the Newton's method now converges linearly.

Now for the iteration $x_{n+1} = x_n - 2\frac{f(x_n)}{f'(x_n)}$:

now

$$h(x) = \frac{2f(x)}{f'(x)} = \frac{2(x-\xi) \cdot g(x)}{2g(x) + (x-\xi)g'(x)}$$

Similarly, we evalute:

$$\lim_{n\to\infty} \frac{\epsilon_{n+1}}{\epsilon_n^2} = \lim_{n\to\infty} \frac{\epsilon_n + h(\xi - \epsilon_n)}{\epsilon_n^2} = \lim_{n\to\infty} \frac{\epsilon_n + h(x_n)}{\epsilon_n^2}$$

For the numerator:

$$\begin{aligned}\epsilon_n + h(x_n) &= \epsilon_n + \frac{2(x_n-\xi)\cdot g(x_n)}{2g(x_n) + (x_n-\xi)g'(x_n)} = \epsilon_n + \frac{-2\epsilon_n \cdot g(x_n)}{2g(x_n) + (x_n-\xi)g'(x_n)} \\ &= \frac{-2\epsilon_n \cdot g(x_n) + 2\epsilon_n g(x_n) - \epsilon^2 \cdot g'(x_n))}{2g(x_n) + (x_n-\xi)g'(x_n)} = \frac{-\epsilon^2 \cdot g'(x_n))}{2g(x_n) + (x_n-\xi)g'(x_n)} \end{aligned} \tag{1}$$

With this, let's go back to the original equation:

$$\lim_{n\to\infty} \frac{\epsilon_{n+1}}{\epsilon_n^2} = \lim_{n\to\infty} \frac{\epsilon_n + h(\xi-\epsilon_n)}{\epsilon_n^2} = \lim_{n\to\infty} \frac{\epsilon_n + h(x_n)}{\epsilon_n^2} = \lim_{x\to\xi} \frac{\frac{-\epsilon^2 \cdot g'(x))}{2g(x)+(x-\xi)g'(x)}}{\epsilon_n^2} = \frac{g'(\xi)}{2g(\xi)}$$

Therefore the iteration does converge quadratically.

Q5:

(a): See q5.py attached.

(b): The roots for this polynomial are:

$$x_1 = 3.0, x_2 = 1.0 + 2j, x_3 = 1.0 - 2j$$

where $j = \sqrt{-1}$

Q6:

(a):

$$xp(x) = x^4 - 4x^3 + 6x^2 - 4x$$
$$p(x) = x^3 - 4x^2 + 6x - 4$$

$$x^2q(x) = x^4 + 2x^3 - 8x^2$$
$$xq(x) = x^3 + 2x^2 - 8x$$
$$q(x) = x^2 + 2x - 8$$

Construct the equation with the Q matrix from the coefficients:

$$Q\overline{x} = \begin{bmatrix} 1 & -4 & 6 & -4 & 0 \\ 0 & 1 & -4 & 6 & -4 \\ 1 & 2 & -8 & 0 & 0 \\ 0 & 1 & 2 & -8 & 0 \\ 0 & 0 & 1 & 2 & -8 \end{bmatrix} \begin{bmatrix} x^4 \\ x^3 \\ x^2 \\ x^1 \\ 1 \end{bmatrix}$$

If $p(x)$ and $q(x)$ share a common root, there should be some $\overline{x} \neq \mathbf{0}$ that satisfies $Q\overline{x} = 0$

Equivalently means:

$$det(Q) = 0$$

Using program (q6.py), we can get $det(Q) = 0$, therefore $p(x)$ and $q(x)$ share a common root.

(b):

To apply the ratio method, we construct Q_1 and Q_2 by removing rows and columns from Q:

$$Q_1 = \begin{bmatrix} -4 & 6 & -4 & 0 \\ 1 & -4 & 6 & -4 \\ 2 & -8 & 0 & 0 \\ 1 & 2 & -8 & 0 \end{bmatrix}$$

$$Q_2 = \begin{bmatrix} 1 & 6 & -4 & 0 \\ 0 & -4 & 6 & -4 \\ 1 & -8 & 0 & 0 \\ 0 & 2 & -8 & 0 \end{bmatrix}$$

The common root is given by $x = \frac{-det(Q_1)}{det(Q_2)} = 2$

Q7:

(a):

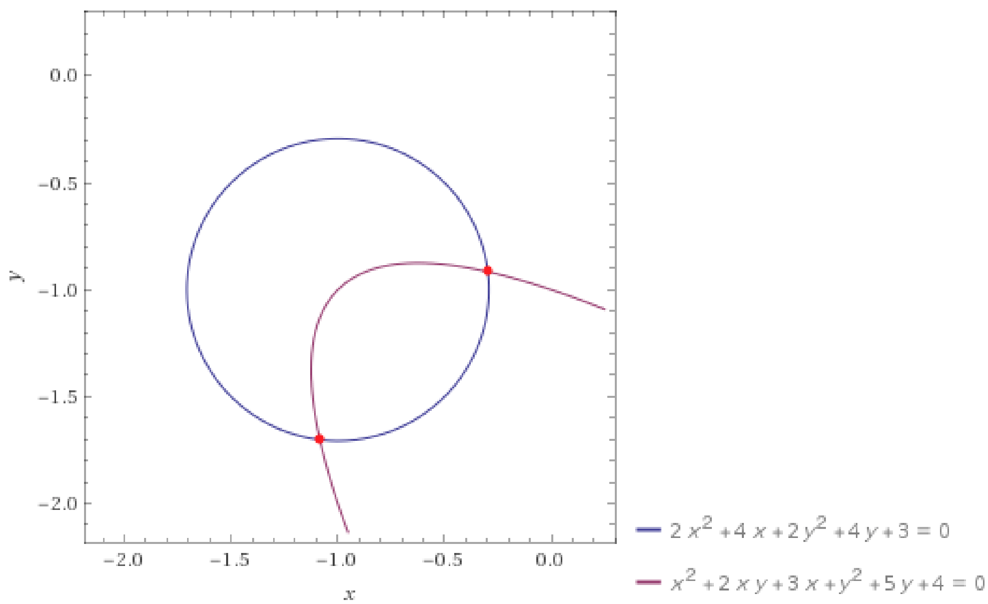

Figure 1: Contours of $p(x, y) = 0$ and $q(x, y) = 0$

(b):

Re-write the functions in terms of y:

$$p(y) = 2y^2 + 4y + (2x^2 + 4x + 3)$$
$$q(y) = y^2 + (2x + 5)y + (x^2 + 3x + 4)$$

Then we get:

$$yp(y) = 2y^3 + 4y^2 + (2x^2 + 4x + 3)y$$
$$yq(y) = y^3 + (2x + 5)y^2 + (x^2 + 3x + 4)y$$

Similar to (a), we construct the matrix Q:

$$Q = \begin{bmatrix} 2 & 4 & 2x^2 + 4x + 3 & 0 \\ 0 & 2 & 4 & 2x^2 + 4x + 3 \\ 1 & 2x + 5 & x^2 + 3x + 4 & 0 \\ 0 & 1 & 2x + 5 & x^2 + 3x + 4 \end{bmatrix}$$

Using WolframAlpha to solve $det(Q)$:

$$det(Q) = 16x^4 + 80x^3 + 144x^2 + 100x + 19$$

Solving $det(Q) = 0$ we get two real roots and two imaginary roots, only consider the real roots:

$$x_1 = -1.08406$$

$$x_2 = -0.297907$$

So these are the x-values of the two intersection points of the two contours, putting them back into $p(x, y)$ and $q(x, y)$:

$$y_1 = -1.70209$$

$$y_2 = -0.915941$$

Therefore the two intersection points of the two contours are:

$$(x_1, y_1) = (-1.08406, -1.70209)$$

$$(x_2, y_2) = (-0.297907, -0.915941)$$

(c):

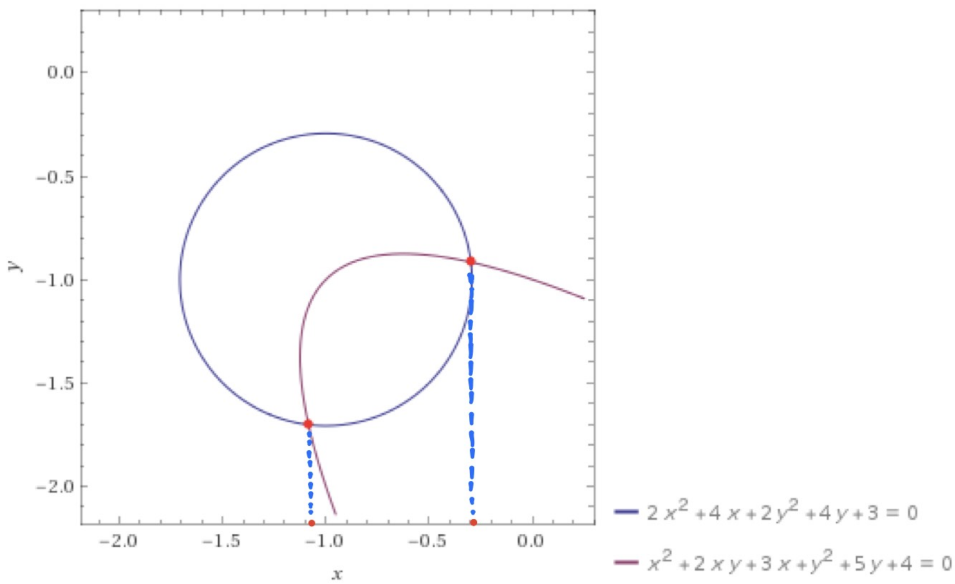

Figure 2: Contours of $p(x, y) = 0$ and $q(x, y) = 0$

Q8:

(a): We define two vectors $\vec{ij} = (x^{(j)} - x^{(i)}, y^{(j)} - y^{(i)})$ and $\vec{ik} = (x^{(k)} - x^{(i)}, y^{(k)} - y^{(i)})$, in fact $\vec{ij}$ and $\vec{ik}$ can be visualized as the two edges of the triangle formed by the three points.

Then for any point $p = (x, y)$ falls within the triangle, the vector $\vec{ip}$ can be expressed a linear combination of $\vec{ij}$ and $\vec{ik}$, with constraints on the coefficients:

$$\vec{ip} = (x - x^{(i)}, y - y^{(i)}) = \alpha * \vec{ij} + \beta * \vec{ik}$$

where

$$0 \leq \alpha \leq 1$$
$$0 \leq \beta \leq 1$$
$$\alpha + \beta \leq 1$$

Therefore, we can write the system of linear equations as:

$$\begin{bmatrix} x^{(j)} - x^{(i)} & x^{(k)} - x^{(i)} \\ y^{(j)} - y^{(i)} & y^{(k)} - y^{(i)} \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} x - x^{(i)} \\ y - y^{(i)} \end{bmatrix}$$

with constraints:

$$0 \leq \alpha \leq 1$$
$$0 \leq \beta \leq 1$$
$$\alpha + \beta \leq 1$$

(b):

I have not solved this problem yet...