Sarah Mahl

FINAL EXAM

1. $I = \int_a^b f(x)\,dx$

(a)(i) If it takes one unit of time to approximate the value of I using the right-end point rule using N subintervals, it will also take one unit of time to approximate the value of I using the left-end point rule and N subintervals. This is because they both have linear convergence, and will thus converge at the same rate.

(ii) The trapezoid rule has quadratic convergence. If we say that one unit of time is defined as t, it will take $t = \frac{N}{10^2}$ units of time to approximate the value of I using the trapezoid rule and N subintervals.

(iii) Simpson's rule will approximate I even faster than the trapezoid rule, at $t = \frac{N}{10^4}$ units of time. This is due to the rate of convergence for Simpson's rule.

(b) $a = 0,\ b = 2\pi,\ f(x) = \cos(x),\ N = 100{,}000$

$$\cancel{I = \int_0^{2\pi} \cos(x)\,dx = -\sin(x)\Big|_0^{2\pi} = -\sin(2\pi) + \sin(0) = 0}$$

$$= \frac{\Delta x}{3}\left[f(x_0) + 4f(x_1) + 2f(x_2) + 4f(x_3) + 2f(x_4) + \ldots + 4f(x_{N-1}) + f(x_N)\right]$$

$$\Delta x = \frac{b-a}{N} = \frac{2\pi - 0}{100000} = \frac{2\pi}{100000}$$

$$x_0 = a = 0,\quad x_N = x_{100{,}000} = b = 2\pi$$

$$I = \frac{1}{3}\left(\frac{2\pi}{100000}\right)\left[f(0) + 4f(x_1) + 2f(x_2) + \ldots + 4f(x_{N-1}) + f(2\pi)\right]$$

With N = 100,000 subintervals, Simpson's rule would give us an estimate of I = 0. This is because f(x) = cos(x) is periodic on the interval [0, 2π], so the values for $f(x_0)$ to $f(x_N)$ in the formula would end up cancelling out.

(c) $a=-1,\ b=1,\ f(x)=e^{-2x+1}$, error $< 1\times10^{-8}$

$$\text{error}_{\text{trap}} \le \frac{(b-a)^3}{12N^2}\max_{c^*\in[a,b]}|f''(c^*)|$$

$$f'(x)=e^{-2x+1}\cdot -2 = -2e^{-2x+1}$$
$$f''(x)=-2e^{-2x+1}\cdot -2 = 4e^{-2x+1}$$

$$1\times10^{-8} < \frac{(1-(-1))^3}{12N^2}\max_{c^*\in[-1,1]}\left|4e^{-2c^*+1}\right|$$

$$\Rightarrow \max_{c^*\in[-1,1]}\left|4e^{-2c^*+1}\right| = 4e^{-2(-1)+1} = 4e^3$$

$$(12\times10^{-8})N^2 < 2^3\cdot 4e^3$$
$$(12\times10^{-8})N^2 < 8\cdot 4e^3$$
$$(12\times10^{-8})N^2 < 32e^3$$
$$N^2 < \frac{32e^3}{12\times10^{-8}} \Rightarrow N \ge \sqrt{\frac{32e^3}{12\times10^{-8}}}$$

To return an error bound less than $1\times10^{-8}$ for the trapezoid rule, $N=\sqrt{\frac{32e^3}{12\times10^{-8}}}$ subintervals are needed.

2. $\begin{cases} x^2+y^2=8 \\ x-y=0 \end{cases} \Rightarrow \begin{cases} x^2+y^2-8=0 \\ x-y=0 \end{cases}$

(a) $x_0=1,\ y_0=0$

$$\mathbf{X}_{n+1}=\mathbf{X}_n-[J(\mathbf{X}_n)]^{-1}F(\mathbf{X}_n) \quad \text{for } n=0,1,2,\dots$$
$$\mathbf{X}_1=\mathbf{X}_0-[J(\mathbf{X}_0)]^{-1}F(\mathbf{X}_0)$$

$$\hookrightarrow \mathbf{X}_0=\begin{pmatrix}x_0\\y_0\end{pmatrix}=\begin{pmatrix}1\\0\end{pmatrix}$$

$$F(\mathbf{X}_0)=\begin{pmatrix}x_0^2+y_0^2-8\\x_0-y_0\end{pmatrix}=\begin{pmatrix}(1)^2+(0)^2-8\\1-0\end{pmatrix}=\begin{pmatrix}-7\\1\end{pmatrix}$$

$$J(\mathbf{X}_0)=\begin{pmatrix}\frac{\partial F_1}{\partial x_0} & \frac{\partial F_1}{\partial y_0}\\ \frac{\partial F_2}{\partial x_0} & \frac{\partial F_2}{y_0}\end{pmatrix}=\begin{pmatrix}2x_0 & 2y_0\\1 & -1\end{pmatrix}=\begin{pmatrix}2(1) & 2(0)\\1 & -1\end{pmatrix}$$
$$=\begin{pmatrix}2 & 0\\1 & -1\end{pmatrix}$$

$$\bar{X}_1 = \bar{X}_0 - [J(\bar{X}_0)]^{-1} F(\bar{X}_0)$$

$$= \begin{pmatrix} 1 \\ 0 \end{pmatrix} - \begin{pmatrix} 2 & 0 \\ 1 & -1 \end{pmatrix}^{-1} \begin{pmatrix} -7 \\ 1 \end{pmatrix}$$

$$\hookrightarrow \begin{pmatrix} 2 & 0 \\ 1 & -1 \end{pmatrix}^{-1} = \frac{1}{2(-1)-0(1)} \begin{pmatrix} -1 & 0 \\ -1 & 2 \end{pmatrix} = \frac{-1}{2} \begin{pmatrix} -1 & 0 \\ -1 & 2 \end{pmatrix}$$

$$= \begin{pmatrix} \frac{1}{2} & 0 \\ \frac{1}{2} & -1 \end{pmatrix}$$

$$\bar{X}_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix} - \begin{pmatrix} \frac{1}{2} & 0 \\ \frac{1}{2} & -1 \end{pmatrix} \begin{pmatrix} -7 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} - \begin{pmatrix} \frac{1}{2}(-7) + 0(1) \\ \frac{1}{2}(-1) + (-1)(1) \end{pmatrix}$$

$$= \begin{pmatrix} 1 \\ 0 \end{pmatrix} - \begin{pmatrix} -\frac{7}{2} \\ -\frac{1}{2} - 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} - \begin{pmatrix} -\frac{7}{2} \\ -\frac{3}{2} \end{pmatrix} = \begin{pmatrix} 1 + \frac{7}{2} \\ \frac{3}{2} \end{pmatrix}$$

$$\boxed{\bar{X}_1 = \begin{pmatrix} \frac{9}{2} \\ \frac{3}{2} \end{pmatrix}}$$

(b) $|r - x_{n+1}| \approx \lambda |r - x_n|$

$\hookrightarrow$ Newton's method converges to $r$, $\lambda = \frac{m-1}{m}$ where $m$ = multiplicity greater than 1

$$\lambda = \frac{2-1}{2} = \frac{1}{2}$$

$N = 50$, $x_0 = \frac{19}{10}$, $y_0 = 2$

$$\begin{cases} x^2 + y^2 = 8 \\ x - y = 0 \end{cases} \qquad \begin{aligned} x^2 + x^2 &= 8 \\ 2x^2 &= 8 \\ x = \pm 2 &\Rightarrow y = \pm 2 \end{aligned}$$

$\hookrightarrow x = y$

The functions intersect at (2,2) and (-2,-2)

After $N=50$ iterations, Newton's method would converge to $\bar{X}_{51} = \begin{pmatrix} 2 \\ 2 \end{pmatrix}$ because our initial guess of $x_0 = \frac{19}{10}$ and $y_0 = 2$ is already so close to the intersection and we are assuming that Newton's method converges.

(c) $\begin{cases} x^2+y^2=8 \\ x-y=0 \end{cases} \Rightarrow \left(\begin{array}{cc|c} 1 & 1 & 8 \\ 1 & -1 & 0 \end{array}\right)$

$A = \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$

$\det \begin{bmatrix} 1-\lambda & 1 \\ 1 & -1-\lambda \end{bmatrix} = (1-\lambda)(-1-\lambda) - 1(1) = -1-\lambda+\lambda+\lambda^2-1$
$= \lambda^2 - 2 = 0$
$\lambda = \pm\sqrt{2}$

False. A is not diagonally dominant, nor is it a symmetric matrix so we cannot use positive definance. Thus, Gauss-Seidel cannot be used to find an approximate solution of this system.

3. Using 2,048 points of the form $(x,y)$, we could use the singular value decomposition (SVD) to find the best-fit curve of the form $y(x) = c_1 + c_2 x + c_3 e^x$ by the linear least squares problem. With our functions of the form $y_1(x) = 1$, $y_2(x) = x$, and $y_3(x) = e^x$, we can create the system $Ax = b$ where

$$A = \begin{pmatrix} 1 & x_1 & e^{x_1} \\ 1 & x_2 & e^{x_2} \\ \vdots & \vdots & \vdots \\ 1 & x_n & e^{x_n} \end{pmatrix}, \quad b = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}, \quad x = \begin{pmatrix} c_1 \\ c_2 \\ c_3 \end{pmatrix}$$

with $n \in [1, 2048]$. The pseudoinverse would then be used to solve the system to find the vector $x$ that would minimize $\|Ax-b\|_2^2$, where we would be taking the pseudoinverse of the SVD of A. This would find the curve of best fit of the form $y(x) = c_1 + c_2 x + c_3 e^x$.

4. $t \in [0, 2\pi]$

$\hat{f}(0) = 20,\ \hat{f}(10) = 3+i,\ \hat{f}(15) = 7,\ \hat{f}(-15) = 7,\ \hat{f}(-10) = 3-i$

$$\hat{f}(k) = \begin{cases} 7 & k = 15 \\ 3+i & k = 10 \\ 20 & k = 0 \\ 3-i & k = -10 \\ 7 & k = -15 \\ 0 & \text{all other } k \end{cases} \quad \left.\begin{matrix} \\ \end{matrix}\right\} \text{complex conjugates}$$

$$f(x) \approx \sum_{k=-N}^{N} \hat{f}(k) e^{\frac{2\pi i k x}{L}},$$

$$\hat{f}(k) = \frac{1}{L} \int_0^L f(x) e^{-\frac{2\pi i k}{L}} dx$$

$L = 2\pi$

$\hat{f}(0) \equiv$ average of $f(x)$ on $x \in [0, 2\pi] = 20$

$$g(t) = 20 + 7e^{15it} + (3+i)e^{10it} + 7e^{-15it} + (3-i)e^{-10it}$$

5. (a) (i) To determine if any animal was in one of the photographs, the SVD would be taken of each photo. Because the SVD for each photo would be slightly different based on wildlife movement, we could select a singular value based on the SVD's of "empty" photographs (no animals) and the SVD of an image with an animal in it (since the SVD's of those would vary more widely). We could then filter for this singular value to find images with animals in them.

(ii) To determine if that animal was an ibex, we could use the nullspace of the SVD of the image such that the singular values of images without an ibex would be zero. Images with nonzero singular values would be the images with an ibex in them.

(b) To reduce the size of Sally's data file without losing any important information, Sally could take a Fourier transform of her data. This would identify the zero or near-zero Fourier coefficients, and thus the data in Sally's data file that would not be used in analysis. This could then be filtered out, leaving only the "important" data, and transformed back into the physical space, leaving Sally with a much smaller and easier to work with data file.

6. (a)

$$A = \begin{pmatrix} 1 & 2 \\ 2 & 1 \end{pmatrix}, \quad x = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}, \quad b = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

$$\lambda = 3, -1 \qquad \hookrightarrow x_0 = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

A is not diagonally dominant, nor is it positive definite. Therefore, Jacobi will not converge for the system $Ax=b$.

(b) The BFGS method is useful because it avoids knowing the exact Jacobian, which is difficult to compute for large matrices, and it avoids the computation of the inverse of a matrix, which is computationally complex for large matrices and takes awhile to complete (since it's an $O(N^3)$ operation). Because the formula for the BFGS method is so complex, however, it is difficult to compute by hand (and increases the likelihood of algebra mistakes).

(c) $p(x) = 35 + 986\sin(2x) + 49\cos(7x)$, $x \in [0, 2\pi]$

The Fourier coefficients of $p(x)$ are all 0.

(d) The most valuable thing I learned in this class was the applications of the FFT, since it's used in so many different things, as well as all of the coding experience from our various homework assignments.

Bonus: The SVD and FFT are related by their ability to filter out zero/near-zero values (singular values in $\Sigma$ for the SVD and Fourier coefficients for the FFT). The compact SVD is similar in theory to a denoised Fourier transform of some data. They also both transform an original set of data into another format ($A = U\Sigma V$ for the SVD of a matrix A, and the conversion into Fourier space for the FFT).

I, Sarah Mahl, completed this exam by myself with my notes, codes, and course materials without using any other intonet or human resources, signed

Sarah Mahl