Review For Final Examination — Math 3607

Tae Eun Kim — Spring 2020

Below is the list of important concepts and topics to review from materials after Midterm 2.

Chapter 12: polynomial interpolation and approximation

- the idea, key types, and implementation of piecewise polynomial interpolation; B-splines will not be on the exam.
- least square approximation and normal equations;
- orthogonality – orthogonal vectors and orthogonal matrices;
- two versions of QR factorization, Moore-Penrose inverse of a matrix, a.k.a. the pseudo-inverse, and the connection to least square problems;
- triangularization of matrices – Gram–Schmidt (review this from linear algebra) and, more importantly, Householder.

Chapter 13: nonlinear rootfinding

- rootfinding tools built in MatLab;
- conditioning of rootfinding problem;
- various iteration methods – their ideas, implementation, and rates of convergence;

Chapter 14: numerical differentiation and integration

- different viewpoints on finite difference formulas (FD, BD, CD), *e.g.*, geometric, interpolation-based, extrapolated, . . . ;
- accuracy of finite difference formulas;
- determination of optimal step size for finite differences;
- Newton-Cotes family of quadrature formulas and their derivations from various viewpoints, *e.g.*, geometric, interpolation-based, extrapolated, . . . ;
- error analysis for quadrature methods – local and global;
- implementation of composite methods;

Chapter 15: ordinary differential equations

- significance of 1st-order ODEs (why is 1st-order sufficient?);
- Runge-Kutta methods (multi-step algorithms) – derivation and accuracy
- application/implementation of the algorithms;
- usage of `ode45`;

Problem 1. (Newton's Method)

In this problem, we calculate the value of $\sqrt[4]{7}$ numerically using Newton's method.

(a) Define function $f(x)$ whose root is $\sqrt[4]{7}$ and derive the corresponding Newton's iterative formula.

(b) Now write a MatLab program to implement Newton's method with the following details:

- Set $x_0 = 1.5$.
- You must include a statement defining $f(x)$.
- Utilize the previous result in the iterative steps; do NOT define $f'(x)$.
- Stop the iteration if either $|f(x_{n+1})| \leq 10^{-10}$ or $|x_{n+1} - x_n| \leq 10^{-8}$.

Problem 2. (Trapezoidal Method)

Consider approximating the integral $I\{f\} = \int_a^b f(x)\,\mathrm{d}x = \int_0^\pi \sin x\,\mathrm{d}x$ using composite trapezoidal method

$$I_h^{[\mathrm{t}]}\{f\} = h\left(\frac{f(x_1)}{2} + \sum_{i=2}^{n-1} f(x_i) + \frac{f(x_n)}{2}\right)$$

where $x_i = (i-1)h$ and $h = \pi/(n-1)$ for $i \in \mathbb{N}[1,n]$. Write a MatLab program which calculates $I_h^{[\mathrm{t}]}\{f\}$ for $n = 5, 9, 17, 33, 65, \ldots, 2049$. Tabulate the values of h, $I_h^{[\mathrm{t}]}\{f\}$, and $I_h^{[\mathrm{t}]}\{f\} - I\{f\}$ using `disp` or `fprintf` function.

Problem 3. (Richardson Extrapolation and Improved Difference Formulas)

(a) Use the first-order forward difference formula

$$D_h^{[1f]}\{f\}(x) = \frac{f(x+h) - f(x)}{h} = f'(x) + \frac{f''(x)}{2}h + \frac{f'''(x)}{6}h^2 + \mathcal{O}(h^3)$$

and Richardson extrapolation to obtain a second-order forward difference formula $D_h^{[2f]}\{f\}(x)$ for first derivatives. Write down explicitly the leading error term.

(b) Let $a < b$ and $x_j = a + (b-a)j/n$ for $j = 0, 1, 2, \ldots, n$ and suppose that the values $y_j = f(x_j)$, $0 \le j \le n$ are known and saved in MatLab as `ydp`; a, b, and n are also saved as `a`, `b`, `n`, respectively. Nothing else is known about f.

Your mission, should you choose to accept it, is to calculate numerically the derivatives $f'(x_j)$ for $0 \le j \le n$ using second-order methods using MatLab.

Problem 4. (Mechanical Vibration and the Modified Euler Method)

Suppose that the motion of a certain spring-mass system satisfies the differential equation

$$u'' + u' + \frac{1}{5}u^3 = 3\cos\omega t$$

and the initial conditions

$$u(0) = 2, u'(0) = 0.$$

Write a MatLab program to plot the trajectory $u(t)$ for $0 \le t \le 100$ using the *modified Euler method* explained below.

The **modified Euler formula** for the initial value problem $\mathbf{y}' = \mathbf{f}(t, \mathbf{y})$, $\mathbf{y}(t_0) = \mathbf{y}_0$ is given by

$$\mathbf{y}_{n+1} = \mathbf{y}_n + h\mathbf{f}\left(t_n + \frac{h}{2}, y_n + \frac{h}{2}\mathbf{f}(t_n, \mathbf{y}_n)\right).$$

Problem 5. (Lorenz Model and `ode45`)

The Lorenz equations are the nonlinear autonomous three-dimensional system

$$\begin{aligned}\dot{x} &= \sigma(y - x)\\ \dot{y} &= x(\rho - z) - y,\\ \dot{z} &= xy - \beta z\end{aligned}$$

where the dot notation indicates the time-derivative $\frac{d}{dt}$. Using

$$\sigma = 10, \quad \rho = 28, \quad \beta = 8/3,$$

plot the three-dimensional trajectory of the particle initially located at $(x, y, z) = (-8, 8, 27)$ for $0 \leq t \leq 10$ using `ode45`.