Math 5630/6630 **Fall 2024**

Homework 5

Tags: Extrapolation & Differentiation *Due Date: 11/07/2024 11:59PM CST*

1 Homework Problems

This part of the homework assignment should be submitted via Canvas. You can scan the answers into PDF files, or typeset them in Word or LaTeX, then convert/compile them into PDF files.

Problem 1.1. *(General Richardson extrapolation) Assume $f(h)$ can be expanded as (similar to the Taylor series)*

$$f(h) = f(0) + C_1 h^{\alpha_1} + C_2 h^{\alpha_2} + \cdots$$

*where the powers $0 < \alpha_1 < \alpha_2 < \cdots$, but they may **not** be integers.*

1. *Use the idea of Richardson extrapolation to derive the "best" approximation to $f(0)$ with $f(h)$, $f(\frac{h}{2})$, and $f(\frac{h}{4})$. Here "best" means the error's power in h is maximized.*

2. *Does your formula depend on the constants $C_1, C_2, \cdots$?*

3. *Suppose $f_1(h)$ satisfies $f_1(0) = A$ with powers $(\alpha_1, \alpha_2, \cdots) = (1, 2, 3, \cdots)$, and $f_2(h)$ satisfies $f_2(0) = A$ with powers $(\alpha_1, \alpha_2, \cdots) = (2, 4, 6, \cdots)$. Can the Richardson extrapolation formula derived from f_2 be applied to the function f_1? Why or why not?*

4. *Can the Richardson extrapolation formula derived from f_1 be applied to the function f_2? Why or why not?*

Problem 1.2. *Derive an approximation formula for $f'(x_0)$ using 4 points $f(x_0)$, $f(x_0+h)$, $f(x_0+2h)$, $f(x_0+3h)$ with the truncation error at $\mathcal{O}(h^3)$.*

Problem 1.3. *Derive an approximation formula for $f''(x_0)$ using 5 points $f(x_0-2h)$, $f(x_0-h)$, $f(x_0)$, $f(x_0+h)$, $f(x_0+2h)$ with the truncation error at $\mathcal{O}(h^4)$.*

Problem 1.4. *Let $p_n(x)$ be the interpolation polynomial at data points $(x_0, f(x_0))$, $\cdots$, $(x_n, f(x_n))$. The error formula is already known (using Newton's form) as*

$$f(x) - p_n(x) = Q(x)(x-x_0)(x-x_1)\cdots(x-x_n),$$

where $Q(x) = f[x_0, x_1, \cdots, x_n, x]$.

i. *Differentiate the above formula and show*

$$f'(x_0) - p_n'(x_0) = Q(x_0)\prod_{k=1}^{n}(x_0 - x_k).$$

ii. *Suppose we are using* $f(x_0)$, $f(x_0+\frac{h}{2})$, $f(x_0+\frac{h}{4})$, $\cdots$, $f(x_0+\frac{h}{2^m})$ *to derive an approximation to* $f'(x_0)$, *that is,*

$$f'(x_0) \approx \frac{1}{h}\left(w_0 f(x_0) + \sum_{k=1}^{m} w_k f(x_0 + \frac{h}{2^k})\right).$$

where $w_0, w_1, \cdots, w_m$ *are coefficients. Use (i) to show the error is bounded by*

$$\left| f'(x_0) - \frac{1}{h}\left(w_0 f(x_0) + \sum_{k=1}^{m} w_k f(x_0 + \frac{h}{2^k})\right)\right| \leq \frac{M}{(m+1)!}\frac{h^m}{2^K}$$

where $M = \max_{x_0 \leq \zeta \leq x_0 + h/2} |f^{(m+1)}(\zeta)|$ *and* $K = \frac{m(m+1)}{2}$.

Problem 1.5. *Let* $f(x) \in C^4(\mathbb{R})$ *and the constants* $M_k = \max_{x\in\mathbb{R}} |f^{(k)}(x)|$, $k = 0, 1, 2, \cdots$. *Consider the finite difference approximation formula for* $f'(x_0)$ *as*

$$f'(x_0) \approx \frac{1}{h}\left(w_0 f(x_0 + h) + w_1 f(x_0 + 2h) + w_2 f(x_0 - \frac{2}{3}h)\right)$$

1. *Find the coefficients* w_0, w_1, w_2 *such that the truncation error has the highest order.*

2. *Suppose the evaluation of* $f(x)$ *suffers from a rounding error that*

$$\boxed{f(x)} = f(x)(1+\delta), \quad |\delta| \leq \eta,$$

where η is the rounding unit. What is the total rounding error of the finite difference approximation (note that all basic operations do not have rounding errors)?

3. *The total error is the sum of truncation and rounding errors. Find a good choice for h to minimize this total error.*

1.1 Extra Problems for MATH 6630

Problem 1.6. *This is a followup of Problem 1.4. Suppose we are using $f(x_0 + \frac{h}{2})$, $f(x_0 + \frac{h}{4})$, $\cdots$, $f(x_0 + \frac{h}{2^m})$, $f(x_0 + \frac{h}{2^{m+1}})$ to derive an approximation to $f'(x_0)$. Estimate the error.*

2 Programming Problems

Implement the following program tasks using your favorite programming language. The `Python` or `MATLAB` starter kit is available at GitHub Link. Follow the guidelines there for your submission.

Problem 2.1. *In this problem, you will implement the Richardson extrapolation based on the formulation in Problem 1.1.*

You will be given two inputs:

a vector of data values: $a_k = f(2^{-k}h)$, $k = 1, 2, \cdots, m$ *and*

a vector of powers $(\alpha_1, \cdots, \alpha_{m-1})$ *in ascending order of length* $(m-1)$ [1].

Implement Richardson extrapolation to find the "best" approximation of $f(0)$*. Note that "best" means the truncation error has the highest order in* h*.*

Problem 2.2. *Implement a program to evaluate the following series as accurately as possible.*

$$A(\beta) = \sum_{i=0}^{\infty} \frac{(-1)^i}{(2i+1)^\beta}, \qquad 0 < \beta \leq 1.$$

Hint: There are many ways. Using Richardson extrapolation is a choice. Let $f(h, \beta)$ *be a function defined by*

$$f(h, \beta) = \sum_{i=0}^{1/h} \frac{(-1)^i}{(2i+1)^\beta}$$

The vector of data values is:

$$a_k = f(2^{-k}, \beta), \quad k = 1, 2, \cdots, m$$

The vector of powers is $(\beta, \beta+1, \beta+2, \cdots, \beta+m-2)$*. Because of rounding errors, the length* m *should not exceed 15.*

[1] The meaning of these powers are in Problem 1.1.

Problem 2.3. *Given a vector of distinct numbers $a_0, a_1, \cdots, a_n$. Let the nodes $x_k = x + a_k h$, use* ***Lagrange interpolation*** *to compute coefficients $c_0, c_1, \cdots, c_n$ of the finite difference approximation*

$$f'(x) \approx \frac{1}{h}\left(c_0 f(x+a_0 h) + c_1 f(x+a_1 h) + \cdots + c_n f(x+a_n h)\right).$$

Caution: there is a factor $\frac{1}{h}$ in the front.

Hint: $c_k = hL_k'(x)$, L_k is the k-th Lagrange polynomial. The coefficients are irrelevant to the choice of h.

$$L_k'(x) = \sum_{m=0, m\neq k}^{n} \frac{1}{x_k - x_m} \prod_{j=0, j\neq k, j\neq m}^{n} \frac{x - x_j}{x_k - x_j}.$$

2.1 Extra Problems for MATH 6630

Problem 2.4. *This is a followup of Problem 2.3. Given a vector of distinct numbers $a_0, a_1, \cdots, a_n$. Let the nodes $x_k = x + a_k h$, compute coefficients $c_0, c_1, \cdots, c_n$ of the finite difference approximation*

$$f^{(l)}(x) \approx \frac{1}{h^l}\left(c_0 f(x+a_0 h) + c_1 f(x+a_1 h) + \cdots + c_n f(x+a_n h)\right), \quad 1 \leq l \leq n.$$

Caution: there is a factor $\frac{1}{h^l}$ in the front.

Hint: Using Lagrange polynomial leads to $c_k = h^l L_k^{(l)}(x)$, but it is not trivial to implement $L_k^{(l)}(x)$. There is a more straightforward way to solve it.