Department of Mathematics, University of Utah

Analysis of Numerical Methods I

MATH 6610 – Section 001 – Fall 2020

Homework 4

Approximation techniques

Due Thursday, December 3, 2020 by 11:59pm MT

Submission instructions:

Create a private repository on `github.com` named `math6610-homework-4`. Add your LaTeX source files and your Matlab/Python code and push to Github. To submit: grant me (username `akilnarayan`) write access to your repository.

You may grant me write access before you complete the assignment. I will not look at your submission until the due date+time specified above. If you choose this route, I will only grade the assignment associated with the last commit before the due date.

All commits timestamped after the due date+time will be ignored

All work in commits before the final valid timestamped commit will be ignored.

Problem assignment:

P1. Let $w(x)$ be a strictly positive, bounded weight function on an interval I on the real line. (I may be unbounded if w decays at infinity sufficiently quickly.) Given $x_1, \ldots, x_N \in I$, let I_N be the associated degree-$(N-1)$ polynomial interpolation operator, i.e., if f is continuous, then $I_N f$ is degree-$(N-1)$ polynomial that interpolates f at the x_j. Define

$$C_w(I) = \left\{ f : I \to \mathbb{R} \;\middle|\; \|f\|_{w,\infty} < \infty \right\}, \qquad \|f\|_{w,\infty} := \sup_{x \in I} w(x)|f(x)|.$$

Prove the following weighted version of Lebesgue's Lemma,

$$\|f - I_N f\|_{w,\infty} \leq [1 + \Lambda_w] \inf_{p \in P_{N-1}} \|f - p\|_{w,\infty},$$

where P_{N-1} is the space of polynomials of degree at most $N-1$, and

$$\Lambda_w = \sup_{x \in I} w(x) \sum_{j=1}^{N} \frac{|\ell_j(x)|}{w(x_j)},$$

where $\ell_j \in P_{N-1}$ is the cardinal Lagrange interpolant, $\ell_j(x_i) = \delta_{i,j}$.

P2. Define the standard L^2 Sobolev spaces of periodic functions on $[0, 2\pi]$: Given a non-negative integer s,

$$H_p^s([0,2\pi]) = \left\{ f : [0,2\pi] \to \mathbb{C} \;\middle|\; f^{(r)}(0) = f^{(r)}(2\pi) \text{ for } r = 0, \ldots s-1, \text{ and } \|f\|_{H_p^s} < \infty \right\},$$

where $f^{(r)}$ denotes the rth derivative of f (with $f^{(0)} \equiv f$), and

$$\|f\|_{H_p^s}^2 = \sum_{j=0}^{s} \left\| f^{(j)} \right\|_{L^2}^2 = \sum_{j=0}^{s} \int_0^{2\pi} \left| f^{(j)}(x) \right|^2 \mathrm{d}x.$$

Let f_n denote the frequency-n Fourier Series approximation to f on $[0, 2\pi]$, i.e.,

$$f_n(x) = \sum_{|j| \le n} \widehat{f}_j(x) \frac{1}{\sqrt{2\pi}} e^{ijx}.$$

Prove that,

$$\|f - f_n\|_{H_p^j} \le n^{j-s} \|f\|_{H_p^s}, \qquad 0 \le j \le s \tag{1}$$

P3. (Discrete Fourier Transforms) Given $f \in L^2([0, 2\pi])$, consider the problem of approximating f by a frequency-n Fourier series:

$$f(\theta) \approx p(\theta) = \sum_{|j| \le n} c_j \phi_j(\theta), \qquad \phi_j(\theta) = \frac{1}{\sqrt{2n+1}} e^{ij\theta}.$$

In this problem, we'll consider interpolative approximations with the equidistant nodal set on $[0, 2\pi]$,

$$\theta_k = \frac{2\pi k}{2n+1}, \qquad k = 0, \ldots, 2n.$$

The interpolation conditions on these $2n + 1$ points furnish constraints for the $2n + 1$ coefficients c_j defined by the linear system:

$$Vc = f, \qquad (f)_j = f(\theta_j), \qquad (c)_j = c_j.$$

The map defined by matrix V is called the Discrete Fourier Transform (DFT).

(a) Show that V is unitary, hence the DFT is a unitary operator.

(b) Assume that $\|f\|_\infty := \sup_{\theta \in [0,2\pi]} |f(\theta)|$ is finite. Define the interpolation operator I_n as the map $f \mapsto p$, with f and p as given above, where both are treated as elements of $L^2([0, 2\pi])$. Derive a bound for the norm of this operator:

$$\|I_n\|_{L^2([0,2\pi]) \mapsto L^2([0,2\pi])},$$

which may be in terms of n and $\|f\|_\infty$, and $\|f\|_{L^2}$.

(c) Assume that $f \in H_p^s([0, 2\pi])$ for some $s > 0$. Prove a variant of Lebesgue's Lemma using a combination of (i) the interpolation operator norm above and (ii) the bound for L^2-optimal Fourier series approximation (1). I.e., provide a bound for

$$\|f - I_n f\|_{L^2},$$

where the bound depends only on norms of f, and on n, and s.

P4. (Non-polynomial interpolation) Let $x_1, \ldots, x_{n+1}$ be distinct real-valued nodes on a compact interval $[a, b]$, and let f be a given continuous function. Prove that the interpolation problem on these nodes that seeks a function p from the space

$$p \in \operatorname{span}\{1, e^x, e^{2x}, \ldots, e^{nx}\},$$

is unisolvent. (Hint: this problem can be reduced to polynomial interpolation.)

P5. This problem concerns interpolation, quadrature formulas, and differentiation formulas. All these problems should be done without a computer.

(a) Let $h(x) = x^3 - 1$. Compute the degree-3 polynomial that interpolates $h(x)$ at $x = -1, 0, 1, 2$.

(b) Let $g(x) = x^4 - 1$. Compute the degree-3 polynomial that interpolates $g(x)$ at $x = -1, 0, 1, 2$.

(c) Compute weights for the closed 4-point Newton-Cotes quadrature rule on $[-1, 1]$. (I.e., the equidistant rule with nodes at the boundaries.)

(d) Consider weights w_j and w'_j for a quadrature rule of the form

$$\int_0^1 f(x)\,\mathrm{d}x \approx w_0 f(0) + w_1 f(1) + w'_0 f'(0) + w'_1 f'(1),$$

where f' is the derivative of f. Compute these weights for a quadrature rule that is exact for all polynomials up to degree 3.

(e) Given $h > 0$, compute weights for the following one-sided differentation formula,

$$f'(x) = w_0 f(x) + w_1 f(x+h) + w_2 f(x+2h)$$

so that the formula has as high a degree of polynomial accuracy as possible. What is the truncation error for this formula?

(f) Given $h > 0$, compute the weights for the following central differentiation formula:

$$f''(x) = w_{-1} f(x-h) + w_0 f(x) + w_1 f(x+h)$$

so that the formula has as high a degree of polynomial accuracy as possible. What is the truncation error for this formula?

Computing assignment:

C1. (Polynomial interpolation) This problem concerns univariate polynomial interpolation.

a. Let $h(x) = 1/(1+25x^2)$. Let $h_N(x)$ denote the degree-$(N-1)$ polynomial interpolant of $h(x)$ at N equispaced points on the interval $[-1,1]$. Plot h and the interpolant h_N for $N = 5, 20, 50$.

b. Plot the Lebesgue function for equispaced points on this interval for $N = 5, 20, 50$. Use this to explain your findings in the previous part.

c. Let $j_N(x)$ denote the degree-$(N-1)$ polynomial interpolant of $h(x)$ at N Chebyshev points on $[-1,1]$. Plot h and the interpolant j_N for $N = 5, 20, 50$.

d. Plot the Lebesgue function for Chebyshev points on this interval for $N = 5, 20, 50$. Use this to explain your findings in the previous part.

C2. (Fourier series error) In this problem you will approximate the following functions on the interval $[0, 2\pi]$:

$$f_j(\theta) := \exp g_j(\theta),$$

where $g_j(\theta)$ are defined by the iterative relations for $j \geq 1$:

$$g_j(\theta) := \int_0^\theta g_{j-1}(\tau)\,\mathrm{d}\tau - c_j, \qquad\qquad g_0(\theta) := \begin{cases} 1, & 0 \leq \theta < \frac{\pi}{2} \\ -1, & \frac{\pi}{2} \leq \theta < \frac{3\pi}{2} \\ 1 & \frac{3\pi}{2} \leq \theta \leq 2\pi \end{cases}$$

The coefficients c_j are chosen such that $\int_0^{2\pi} g_j(\theta)\,\mathrm{d}\theta = 0$.

With p_n the discrete Fourier Transform approximation described in P3, plot $\|f_0 - p_n\|_{L^2}$ as a function of n and use this to determine a rate of convergence. Note that you cannot really compute the errors exactly, so you will need to use a discrete (say equidistant) grid to do so; you will need to use a sufficiently refined grid so that errors measure mainly the error in Fourier interpolation and are not dominated by the quadrature (discretization) error.

Repeat this experiment for $j = 1, 2, 3$, plotting the L^2 error as a function of n and also determining the rate of convergence.
(Think a bit about the best way to present these results visually. From your expectation of how the error should behave, what is a revealing way to visualize the errors?)

C3. (AAA rational approximation) Implement the AAA algorithm. Apply this algorithm for approximation of the function

$$f(z) = \tan(\beta z),$$

for complex numbers z and a given positive real parameter β. Use 1000 equispaced points on the unit circle in the complex plane as the training grid, and plot the maximum error (on this training grid) as a function of the number of iterations of the algorithm. Plot such curves for $\beta = 4, 16, 64, 256$.