Department of Mathematics, University of Utah

Orthogonal polynomials/Spectral methods for PDEs
MATH 5750/6880 – Section 002 – Fall 2018
Homework 3
Approximations with Fourier Series

Due November 1, 2018

This assignment requires simulations using Fourier series approximations. There are no tools in `pyopoly` for accomplishing this. Therefore, you will need write your own code for discrete transforms, quadrature, basis functions, etc. You may use any programming language that you are comfortable with.

1. Consider the Fourier Series approximation

$$f(x) \approx f_n(x) \coloneqq \sum_{|k| \leq n} c_k \phi_k(x), \qquad \phi_k(x) = \frac{1}{\sqrt{2\pi}} \exp(ikx),$$

for $x \in [0, 2\pi)$. A *filter* function is a function $g : \mathbb{R} \to \mathbb{R}$ satisfying

- g is an even function.
- $g(0) = 1$
- $g(\xi) = 0$, $|\xi| \geq 1$

The filtered version of f_n is the new function

$$f_n^g(x) \coloneqq \sum_{|k| \leq n} g\left(\frac{k}{n}\right) c_k \phi_k(x).$$

Find an explicit expression in terms of ϕ_k and g for a complex-valued bivariate function $G(x, y)$ such that

$$f_n^g(x) = \int_0^{2\pi} f_n(y) G(x, y) \, \mathrm{d}y.$$

(If you don't know where to get started: what should G be if $g \equiv 1$?)

2. Consider the following filter functions for $\xi \in [0, 1]$:

$$g_C(\xi) = 1 - \xi, \qquad \text{(Cesáro filter)}$$

$$g_R(\xi) = \frac{\sin(\pi\xi)}{\pi\xi}, \qquad \text{(Lanczos filter)}$$

$$g_E(\xi) = \begin{cases} 1, & \xi < \xi_c \\ \exp\left[-\alpha \left(\frac{\xi-\xi_c}{1-\xi_c}\right)^p\right], & \xi_c \le \xi \le 1 \end{cases} \qquad \text{(Exponential filter)}$$

where for the exponential filter we take $\alpha = -\log(\epsilon_{\text{mach}})$ with ϵ_{mach} machine precision. The filter order $p > 0$ and the cutoff $\xi_c \in [0, 1)$ can be freely chosen. For this exercise, choose $p = 6$ and $\xi_c = 0.25$. Consider the test function on $[0, 2\pi)$:

$$f(x) = \begin{cases} x, & x < \pi, \\ x - 2\pi, & x \ge \pi. \end{cases}$$

For odd values of N, say $N = 17, 33, 65, 129$, plot logarthmic pointwise errors over $[0, 2\pi)$ between N-point Fourier interpolant approximations f_n (where $n = (N-1)/2$) and f. Do the same for f^g with $g = g_C$, g_R, g_E. What do you observe about the pointwise error for each of these approximations as a function of x and N?

3. Numerically solve the boundary value problem

$$-u_{xx} + u = f(x), \qquad u(0) = u(2\pi), \qquad u'(0) = u'(2\pi),$$

using N degrees of freedom in a Fourier series approximation. (Again, take N odd.) Choose f such that the solution u is a smooth function, e.g.,

$$u(x) = \exp(\cos(kx)), \qquad k \in \mathbb{N},$$

for some choice of k. Solve this equation two ways, using the Galerkin method (where you should compute the expansion coefficients for f using an over-resolved discrete Fourier transform) and a collocation method.

- What is the rate of convergence in N you observe for these methods?
- Are the convergence trends affected by the value of k?
- Does the accuracy substantially differ between Galerkin and collocation approaches for this problem?
- How do convergence rates behave if you instead choose the right-hand side f as in problem 2?