MTH-308B Mini Project 2

Aniket Pandey

Roll No: 160113

Problem Statement

Given a discrete computational grid $X_{grid} = \{x_j\}_{j=1}^{n}$ along with its evaluation data $F_{grid} = \{f(x_j)\}_{j=1}^{n}$, where $x_j \in [a, b]$, $a < b$. The function $f \in C^{\infty}[a, b]$ and satisfies the property:

$$f^{(k)}(a) = f^{(k)}(b) = 0 \qquad \forall k \in [1, 10] \tag{1}$$

$$f^{(11)}(a) \neq f^{(11)}(b) \tag{2}$$

Design and implement an efficient polynomial interpolation method to approximate $f_n[a, b] \to \mathbb{R}$ on evaluation points based on the given data (X_{grid}, F_{grid}).

Proposed Algorithm

The design implements the method described in the paper, *Finding the Zeros of a Univariate Equation: Proxy Rootfinders, Chebyshev Interpolation, and the Companion Matrix* by *John P. Boyd*.

Implementation Abstract

When a function $f(x)$ is holomorphic on an interval $x \in [a, b]$, its roots on the interval can be computed by the following three-step procedure.

1. Approximate $f(x)$ on $[a, b]$ by a polynomial $f_N(x)$ using adaptive Chebyshev interpolation.
2. Form the Chebyshev Frobenius companion matrix whose elements are trivial functions of the Chebyshev coefficients of the interpolant $f_N(x)$.
3. Compute all the eigenvalues of the companion matrix.

The eigenvalues λ which lie on the real interval $\lambda \in [a, b]$ are very accurate approximations to the zeros of $f(x)$ on the target interval.

Implementation Details

Here, we construct Chebyshev interpolation of $f(x)$ to compute a Chebyshev series, including terms up to and including T_N, on the interval $x \in [a, b]$.

1. Create the set of interpolation points: X_{Grid}

$$x_k \equiv \Big(\frac{a+b}{2}\Big) + \frac{b-a}{2} cos\Big(\pi \frac{k}{N}\Big), \qquad k = 0, 1, 2, ..., N \tag{3}$$

2. Compute the grid point values of $f(x)$, the function to be approximated:

$$f_k \equiv f(x_k), \qquad k = 0, 1, 2, ..., N \tag{4}$$

   Note: Such type of points avoid the periodicity factor that may have caused the error since the points are unequally placed and heavily dependent on X_{grid}.

3. As mentioned in one of the Lectures (and Lab11), interpolating a polynomial based on the points generated by Chebyshev series results in a polynominal which explonentially converges to the actual function, as n is increased.

   The technique used to generate the interpolation matrix (*nGrid* X *nGrid*) is:

$$\Gamma_{jk} = \frac{2}{p_k p_j N} cos\Big(j\pi \frac{k}{N}\Big) \tag{5}$$

4. Compute the coefficients through a vector matrix-multiplication

$$a_j = \sum_{k=0}^{N} \Gamma_{jk} f_k, \qquad j = 0, 1, 2, ..., N \tag{6}$$

5. The function $f(k)$ can then be approximated by inverse interpolation method.

$$f \approx \sum_{j=0}^{N} a_j cos\Bigg\{ j cos^{-1}\Bigg(\frac{2x-(b+a)}{b-a}\Bigg)\Bigg\} \tag{7}$$

The vector matrix-multiplication procedure could have been accelerated by *Fast-Fourier Transform*, but since this vector-matrix multiplication costs $O(2N^2)$ while the eigensolving cost is $O(10N^3)$, the FFT is not worth the effort and computation time.

References

1. Polynomial Interpolation Wiki: *https://en.wikipedia.org/wiki/Polynomial_interpolation*

2. John P. Boyd, *Finding the Zeros of a Univariate Equation: Proxy Rootfinders, Chebyshev Interpolation, and the Companion Matrix*