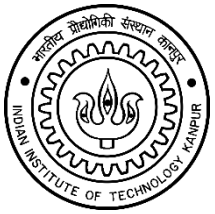

Indian Institute of Technology Kanpur

Mini Project -2

MTH 308A – Principles of Numerical Analysis

Submitted by – **Ashu Prakash**

Roll No.- **14146**

DESIGNING OF AN APPROXIMATE FUNCTION BY INTERPOLATION.

We have been given $a, b \in \mathbb{R}$ $(a < b)$ and a positive integer n. We have to choose a discrete computational grid $X_{grid} = \{x_j\}_{\{j=1\}}^{n}$ with $x_j \epsilon [a, b]$, $j \epsilon \mathbb{N}$, j is bounded by $'n'$ ofcourse. Also $x_i \neq x_j$ for $i \neq j$.

For the interpolation, I am using Chebyshev Interpolation of a function *esp*. Chebyshev – Lobatto Grid.

Our goal is to compute a Chebyshev series, on the interval $x \epsilon [a, b]$.

Step 1: Creation of interpolation points:

$$x_k = \frac{b-a}{2} \cos\left(\frac{\pi k}{N}\right) + \left(\frac{b+a}{2}\right), \quad k = 0,1,2, \ \ldots, N$$

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

$$f_k = f(x_k), \quad k = 0,1, \ \ldots, N$$

The above two steps are incorporated in the code as

```
[xGrid, fGrid] = discreteData(nGrid, a, b, f);
```

Step 3: This step is for the computation of the interpolation matrix. We define $p_j = 2$ if $j = 0\ or\ j = N$ and $p_j = 1, j\epsilon[1, N-1]$. The elements of the interpolation matrix are defined as

$$I_{jk} = \frac{2}{p_j p_k N} \cos\left(\frac{j\pi k}{N}\right)$$

Step 4: Now we compute the coefficients through a vector – matrix multiplication:

$$a_j = \sum_{k=0}^{N} \mathrm{I_{jk}} f_k, \quad j = 0,1,2, \ \ldots, N$$

Step 5: Finally, we define the output of Chebyshev approximation

$$f \approx \sum_{j=0}^{N} \mathrm{a_j} \cos\ \left(\ j \arccos \frac{2x - (b+a)}{b-a}\ \right)$$

The steps 3, 4 and 5 are scripted in the function

```
approx = approxFunction(xEval, xGrid, fGrid);
```

The main.m file is returning the `maxError/maxExact` that is nothing but the max norm of f divided by the max norm of the error.

So, Finally my approximated function is based on the discrete data (X_{grid}, F_{grid}).

Example input values:

```
a = 1;
b = 100;
f = @(x) ((x-a).*(x-b)).^11;
nGrid = 100;
nEval = 60;
```

Example outputs:

```
Elapsed time is 0.003142 seconds.
maxError/maxExact = 7.1688e-15
```

```
Elapsed time is 0.003332 seconds.
maxError/maxExact = 1.0014e-14
```

```
Elapsed time is 0.008379 seconds.
maxError/maxExact = 4.8195e-15
```

Reference:

[1] Boyd, John P. *Finding the Zeros of a Univariate Equation: Proxy Rootfinders, Chebyshev Interpolation, and the Companion Matrix*. Society for Industrial and Applied Mathematics, 55(2):375-396, 2013

[2] Class notes of *"Principles of Numerical Analysis" MTH308A,* Spring 2016 – 17, Indian Institute of Technology Kanpur.