Midterm Exam

Due Apr 17, 2024

- The Chebyshev polynomial of the first kind of order n is defined as follows,

$$T_n(x) = \cos[n \cdot \arccos(x)], \qquad x \in [-1,1], \qquad n = 0,1,2,\dots$$

- Explicit expressions for the first six Chebyshev polynomials are

$$\begin{aligned} T_0(x) &= 1, & T_1(x) &= x \\ T_2(x) &= 2x^2 - 1, & T_3(x) &= 4x^3 - 3x \\ T_4(x) &= 8x^4 - 8x^2 + 1, & T_5(x) &= 16x^5 - 20x^3 + 5x \end{aligned}$$

- The Chebyshev polynomials $T_n(x)$ satisfy the following three-term recurrence relation:

$$\boldsymbol{T_{n+1}(x) = 2xT_n(x) - T_{n-1}(x), \qquad n = 1, 2, 3, \dots}$$

with starting values $T_0(x) = 1, T_1(x) = x$.

Problem 1.1

Now define the following sum of Chebyshev polynomials,

$$P_\sigma^K(\varepsilon) = \frac{1}{D}\sum_{n=0}^{K} c_n^\sigma \, T_n(\varepsilon)$$

where

$$c_n^\sigma = \sqrt{4 - 3\delta_{0,n}}\cos\big(n \cdot \arccos(\sigma)\big)$$

D is the normalization constant assuring that $P_\sigma^K(\sigma) = 1$.

Make a user-defined function and then take $K = 22$ and $\sigma = 0$, try to **calculate and plot** $P_\sigma^K(\varepsilon)$ as a function of ε in the range of $[-1,1]$. Show your result in a plot using `linspace` with 200 points of ε in the range of $[-1,1]$.

Problem 1.2

Next, it would be nice to avoid the explicit computation of the Chebyshev polynomials appearing in $P_\sigma^K(\varepsilon)$. Instead, we use the following algorithm to perform the sum:

- Input: $x; c_0, c_1, \dots, c_N$
- Output: $S_N(x) = \sum_{k=0}^{N} c_k T_k(x)$

Starting from $\boldsymbol{b_{N+1} = 0}$; $\boldsymbol{b_N = c_N}$

Do $\boldsymbol{r = N-1, N-2, \dots, 1}$

$$\boldsymbol{b_r = 2x b_{r+1} - b_{r+2} + c_r}$$

Show analytically that

$$S_N(x) = \sum_{k=0}^{N} c_k T_k(x) = x b_1 - b_2 + c_0$$

Hint: using the matrix notation and the recurrence relation. Also notice the similarity between the relation of b_r and the recurrence relation.

Problem 1.3

Using the algorithm introduced in the previous slide to define a **new** user-defined function and again take $K = 22$ and $\sigma = 0$, **recalculate and replot** $P_\sigma^K(\varepsilon)$ as a function of ε in the range of $[-1,1]$. Show your result again in a plot using `linspace` with 200 points of ε in the range of $[-1,1]$.

Next, make a direct comparison between these two different methods. Namely plot the results from these two methods in a single figure to see whether they are overlapped with each other. Do you get the same result?

Problem 1.4

Now we define a 2000×2000 tridiagonal matrix. The 2000 central diagonal matrix elements have been in the file “central_diag.txt” and the 1999 up or down diagonals are in “up_diag.txt”. Up diagonals equals down diagonals. Load these data to construct the tridiagonal matrix $\mathbf{H}$.

First, use the `numpy` eigensolver to find all the eigenvalues of $\mathbf{H}$.

Next, try to use the user-defined function of QR decomposition (similar to what we have done in the homework 3) to find all the eigenvalues of $\mathbf{H}$ and order them in an ascending way. Note that we just need to know the eigenvalues. The eigenvectors can be discarded.

Do you get the same results?

For preparation of the next subproblem, please normalize the matrix $\mathbf{H}$ to $\mathbf{H}'$

$$\mathbf{H}' = \frac{\mathbf{2H} - (E_{min} + E_{max})\mathbb{I}}{E_{max} - E_{min}}$$

Where E_{max} and E_{min} are the maximum and minimun eigenvalues you just obtained, $\mathbb{I}$ is the identity matrix.

Problem 1.5

We also define a 2000×1 column vector $|\psi\rangle$ whose elements are random numbers. Please also load this vector from the file "random_vector.txt".

Calculate the normalization constant D_ψ for $|\psi\rangle$ which equals $\langle\psi|\psi\rangle$. The row vector $\langle\psi|$ is the transpose of the column vector $|\psi\rangle$.

Redefine $|\tilde{\psi}\rangle = \frac{1}{\sqrt{D_\psi}}|\psi\rangle$, now assume $K = 22$ and $\sigma = 0$ try to numerically calculate the following quantity,

$$\langle\tilde{\psi}|P_\sigma^K(\mathbf{H}')|\tilde{\psi}\rangle$$

where

$$P_\sigma^K(\mathbf{H}') = \frac{1}{D}\sum_{n=0}^{K} c_n^\sigma\, T_n(\mathbf{H}')$$

To save space, you should transform the tridiagonal matrix $\mathbf{H}'$ into a 3×2000 $\widetilde{\mathbf{H}}'$ matrix when performing the calculation.