Homework 6

Objective: Lagrange interpolation, Chebyshev and Legendre polynomials. Manipulating orthogonal polynomials.

1. a. Write down the analytic expression for the n Chebyshev nodes in the interval [1, 2].
   b. Use the error bound for the Lagrange interpolation to determine the integer which is the minimum number of Chebyshev nodes required for interpolating the Natural Logarithm \( \ln(x) \) in the interval [2, 4] to precision \( 10^{-12} \).

2. Write a Matlab script of no more than 10 lines to calculate the Newton-Cotes quadrature weights \( w_j \), \( j = 1, 2, \ldots, n, n+1 \); go to the end of this handout for more details about \( w_j \). For uniformity, we require that the script starts with the four lines:
   
   \[
   n=2; h=1/n; nodes=(0:h:1)';
   \]
   
   \[
   f=eye(n+1);
   \]
   
   where the \( j \)-th column of the matrix \( f \) are needed for the Lagrange interpolation in order to find the Lagrange Basis functions \( L^n_{n,j} \); see bottom for more details.
   
   a. Provide the script.
   b. Print off \( w \) (to 16 digits) for \( n = 1 \) (trapezoidal rule), \( n = 2 \) (Simpson’s rule), \( n = 8 \) (call it border-line rule), \( n = 20 \) (call it miserable rule) and \( n = 20 \) (call it the Devil’s rule).
   c. For each rule, print out the Lebesgue constant = norm(h*w,1).

   Hint: Make use of the three Matlab functions polyval, polyint, and polyfit in the order:
   \[
   \text{polyval(polyint(polyfit( · · · ))) .}
   \]

3. Let \( \{x_j, j = 1, 2, \ldots, n\} \) be the \( n \) roots in \((-1, 1)\) of the Legendre polynomial \( p_n(x) \) of degree \( n \). The Chebyshev node \( t_j = \cos[(n-j+1/2)\pi/n] \) is known to be close to \( x_j \). Apply Newton’s iteration to find \( x_j \) to double precision with \( t_j \) as the starting value, for \( j = 1, 2, \ldots, \lfloor n/2 \rfloor \).

   Remark: Use the recursion \( p_0(x) = 1, p_1(x) = x, \) and \( p_{k+1}(x) = [(2k+1) \cdot x \cdot p_k(x) - k \cdot p_{k-1}(x)]/(k+1) \) to evaluate \( p_n(x) \); derive a similar recursion to evaluate \( p'_k(x) \).
   
   a. For each of \( n = 6, 11 \) cases, make a plot on the interval \([-1, 1]\) to mark the locations of the points \( x_j \) and \( t_j \) for \( j = 1, 2, \ldots, \lfloor n/2 \rfloor \).
   b. Do the iterations converge quadratically. Show a numerical evidence.
   c. Describe in plain English your stopping criterion.
   d. Show the number of iterations required to obtain double precision solution for the cases \( n = 11, j = 1, 2, 3, 4, 5 \).

4. Given nodes \( \{x_i, 1 \leq i \leq m\} \) and basis functions \( \{\beta_j(x), 1 \leq j \leq n\} \), compute the 2-norm condition number of the \( m \)-by-\( n \) matrix \( B \) = \{\( b_{ij} \), \( b_{ij} = \beta_j(x_i) \)\} for the following setting.

   \[
   (A1) \quad m = n = 8, \text{ equispaced nodes } \{x_i\} \text{ in } [-1, 1] \text{ with } x_1 = -1, x_m = 1, \text{ and monomials basis } \beta_j(x) = x^{j-1}.
   \]
   
   \[
   (A2) \quad \text{Same as (A1) with } m = n = 20
   \]
   
   \[
   (A3) \quad \text{Same as (A1) with } n = 20, m = n^2
   \]
   
   \[
   (B1) \quad \text{Repeat (A1) and (A2) with Chebyshev basis } \beta_j(x) = T_{j-1}(x).
   \]
(B2) $n = 20$, Chebyshev basis $\beta_j(x) = T_{j-1}(x)$, equispaced nodes $\{x_i\}$ in $[-1, 1]$ with $x_1 = -1 + 1/m$, $x_m = 1 - 1/m$. Four experiments with $m = n, n^2/4, n^2/2, n^2$.

(B3) Repeat (B2) with the new inner product $u \cdot v =: u^T D v$ with $D$ a diagonal matrix: $D_{ii} = (1 - x_i^2)^{-1/2}$. This new inner product introduces a new 2-norm $\|u\|_2 = (u \cdot u)^{1/2}$.

There are two ways to compute the condition number: using MATLAB function `cond` or `norm`; each time you need to scale the matrix $B$ with something related to $D$.

(C1) Chebyshev basis $\beta_j(x) = T_{j-1}(x)$, Chebyshev nodes $\{x_i\}$ in $[-1, 1]$ which are the roots of $T_n(x)$. Four experiments with $m = n = 20, 40, 80, 160$.

(C2) Legendre basis $\beta_j(x) = p_{j-1}(x)$, and Legendre nodes $\{x_i\}$ in $[-1, 1]$ which are the roots of the degree $n$ Legendre polynomial $p_n(x)$. Four experiments with $m = n = 20, 40, 80, 160$.

(C3) Same as (C2), with the new inner product $u \cdot v =: u^T D v$ with $D$ a diagonal matrix:

$$D_{ii} = w_i,$$ where $w_i$ is the Gaussian quadrature weights given by the formula

$$w_i = \frac{2}{(1 - x_i^2)[p'_n(x_i)]^2},$$ see Abramowitz and Stegun, page 887, section 25.4.29.

**Lagrange Interpolation.** The $j$-th Lagrange Basis function associated with the $n+1$ distinct nodes $\{x_k, k = 0, 1, \ldots, n\}$ is defined by the formula

$$L_{n,j}(x) = \prod_{k \neq j}^{n} \frac{x - x_k}{x_j - x_k} \quad (1)$$

for $j = 0, 1, \ldots, n$. The interpolating polynomial $p_n(x)$ to a function $f(x)$ at the nodes $\{x_j\}$ is given by the formula

$$P_n(x) = \sum_{j=0}^{n} f(x_j)L_{n,j}(x) \quad (2)$$

In the associated quadrature with nodes $\{x_j\}$

$$\int_{a}^{b} f(x)dx \sim \int_{a}^{b} P_n(x)dx = \sum_{j=0}^{n} w_j f(x_j), \quad (3)$$

the weights are obviously given by

$$w_j = \int_{a}^{b} L_{n,j}(x)dx \quad (4)$$

In the special case of Newton-Cotes, $a = x_0, b = x_n$, and $\{x_j\}$ are equispaced. Moreover, the weights $w_j$ are traditionally defined as

$$w_j = \frac{1}{h} \int_{a}^{b} L_{n,j}(x)dx \quad (5)$$

and consequently, the quadrature assumes the form

$$\int_{a}^{b} f(x)dx \sim h \sum_{j=0}^{n} w_j f(x_j) \quad (6)$$