Numerical Mathematics: Homework 3

Due on T.B.D at 24:00pm

Problem 1

Write a function called:

```
[ y_int ] = lagrangeinterp( x_int, X, Y )
```

that evaluates a Lagrange interpolation formula for a given set of points. Here, `X` and `Y` are column vectors whose entries contain the coordinates x_1, x_2,...x_n and values $f(x_1)$, $f(x_2)$,...$f(x_n)$, respectively. The output `y_int` are the values of the interpolating polynomial at the interpolation points `x_int`, which are again both column vectors.

To code things efficiently, we are going to re-write the Lagrange interpolation formula. Let $l(x) := (x - x_1)(x - x_2)\cdots(x - x_n)$, so that:

$$L_{n-1,k(x)} = l(x)\frac{w_k}{x - x_k}, \quad w_k := \frac{1}{\prod\limits_{j=1,j\neq k}^{n}(x_k - x_j)}.$$

By doing so, the interpolating polynomial can be re-expressed as:

$$P_{n-1}(x) = \sum_{k=1}^{n} f(x_k)L_{n-1,k}(x) = l(x)\sum_{k=1}^{n} f(x_k)\frac{w_k}{x - x_k}$$

Explain in your own words why the formula above is computationally more efficient than classical one derived in lecture.

Note: Your matlab routine must take special care of the case when one of the interpolating points coincide with x_1, x_2,...x_n to avoid division by zero!

Problem 2

Consider the function:

$$f(x) = \frac{1}{1 + 25x^2}$$

We are going to approximate this function with an interpolating polynomial. The interpolation nodes are to our discretion. For instance, we may choose equidistant points or the Chebyshev nodes as discussed in class. Write a function called: `[ ] = cheb_vs_equispaced( n , type )` which generates a figure with the follow items all in one plot:

1. A graph of the original function,
2. Markers at the interpolation points
3. A graph of interpolating polynomial.

The argument `n` describes the degree of the interpolating polynomial (hence `n`+1 nodes). The argument `type` specificies whether one desires an equidistant or Chebhyshev interpolation. Entering `type='equispaced'` should yield an equispaced interpolation, entering `type='Chebyshev'` should yield a Chebyshev interpolation. Describe what you observe from the plots. Which type of interpolation does a better job? Does this make sense?

Problem 3

Let us generalize the concept of polynomial interpolation a bit. Suppose that we are given a continuously differentiable function on an interval $[a, b]$. Assume that, in addition to the function values $f(x_1)$, $f(x_2)$, ... ,$f(x_n)$ at x_1, x_2, ... x_n, the derivative values $f'(x_1)$, $f'(x_2)$, ... ,$f'(x_n)$ are also provided. Let:

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

denote j'th Lagrange interpolation. Show that the $2n - 1$ degree polynomial:

$$H_{2n-1}(x) = \sum_{k=1}^{n} f(x_k) H_{n-1,k}(x) + f'(x_k) \hat{H}_{n-1,k}(x)$$

$$H_{n-1,k}(x) = [1 - 2(x - x_k) L'_{n-1,k}(x_k)]\, L^2_{n-1,k}(x), \qquad \hat{H}_{n-1,k}(x) = (x - x_k)\, L^2_{n-1,k}(x)$$

uniquely interpolates the function values and its derivatives at the respective . To show uniqueness, you may want to make use of the fundamental theorem of algebra.