Polynomial Interpolation

Tutorial homework question: 8b.

H By Hand; **C** Computer; **T** Theory; **E** Extra; **A** Advanced.

Recommended: H1.*; C4; H6.b; H7; H8.a; C9.a; H10.b; C13; C14.*

Taylor Series

H1. Let $f(x) = \sqrt{x}$.

- **a.** Compute $f'(x)$, $f''(x)$ and $f'''(x)$.
- **b.** Find the quadratic Taylor polynomial p_2 and cubic Taylor polynomial $p_3(x)$ around $x_0 = 4$.
- **c.** Use $p_2(x)$ and $p_3(x)$ to approximate $\sqrt{4.5}$, $\sqrt{4.25}$, $\sqrt{3.75}$ and $\sqrt{3.5}$ (without using a calculator).
- **d.** How would you expect the errors $|f(x) - p_2(x)|$ and $|f(x) - p_3(x)|$ to depend on $|x - x_0|$?
- **e.** Determine the actual error of the approximations in (b), and relate this to your predictions in (c).

E2. Use the linear and cubic Taylor polynomials of $\sin(x)$ to approximate $\sin(2°)$. Use the error term of the Taylor polynomial to bound the error of the approximation.

Hint: Use the bounds $|\sin(x)|, |\cos(x)| \leq 1$ for all x.

A3. One definition of π is as the first positive solution of $\sin(x) = 0$. Use the bisection method starting with $a = 3$ and $b = 3.5$ and the Taylor series for sin with remainder term to estimate π to an accuracy of 10^{-2}. Do *not* use the built-in `sin` function!

Hint: At all steps of the bisection method, you should evaluate $\sin(x)$ to an accuracy sufficient to determine whether the value is strictly positive or strictly negative.

Polynomial Interpolation

C4. Use the Matlab command `polyfit` to compute the polynomial interpolating the data:

x	0.0	0.6	1.5	1.7	1.9	2.1	2.3	2.6	2.8	3.0
y	0.13	0.38	1.50	1.49	1.12	0.98	1.15	1.89	2.35	2.28

.

Check your answer by using the Matlab command `polyval` to evaluate the polynomial at the interpolation points. Estimate the value of y when $x = 1.0$ and $x = 2.0$.

Plot the polynomial over the intervals $[0, 3]$ and over the interval $[1.5, 3]$, and compare with plots of the raw data. Comment on whether your estimates for y are reliable.

Hint: The command `cs=polyfit(xs,ys,n)` to computes the *coefficients* c_i of the interpolating polynomial for $n+1$ data points. You can define the polynomial *function* by `p=@(x)polyval(cs,x)`.

HC5. Write down and sketch the Lagrange basis polynomials $L_{3,k}$ for the values x below.

x	0.6	1.5	1.7	1.9
y	0.38	1.50	1.49	1.12

.

Use your answer to compute the interpolating polynomial p for the data, and estimate the value of y when $x = 1.0$.

By defining appropriate y-values, use Matlab's `polyfit` command to compute the Lagrange basis polynomials, and the interpolating polynomial, and plot these on the same axes.

H6. Write down explicitly the Lagrange basis polynomials for the following sets of interpolation points:

a. $x_0 = a$, $x_1 = b$. **b.** $x_0 = -h$, $x_1 = 0$, $x_2 = +h$. **c.** $x_0 = 0$, $x_1 = h$, $x_2 = 2h$.

Hence, or otherwise, write down the formula for the polynomial interpolating a function f at the given data points.

Note: This is an important theoretical question which will come in useful for finding differentiation formulae!

H7. For $k = 1, 2, 3$, compute the interpolating polynomial of degree k for the data $(x_i, y_i)_{i=0,\ldots,k}$ below using (i) Neville's method, (ii) nested form, directly computing the coefficients a_i, and (iii) divided differences.

i	0	1	2	3
x_i	0.6	1.5	1.7	1.9
y_i	0.38	1.50	1.49	1.12

H8. Compute nested interpolating polynomials of degree 3 for the data $y = f(x)$ below using divided differences, and approximate the missing value:

a.

x	2.1	2.3	2.6	2.7	2.4
y	6.9	7.6	8.5	8.8	

b.

x	0.25	0.5	0.75	1.0	0.8
y	-0.07	-0.03	0.34	1.10	

Suppose $f(1.1) = 2.00$ is added to the data for (b). Construct the interpolating polynomial of degree 4, and compute a new approximation for the missing value.

C9. Write a Matlab function to compute the coefficients a_i of the nested polynomial interpolating data (x_i, y_i) using divided differences.

Write another function to evaluate the polynomial at the point w given the interpolation points x_i and coefficients a_i.

Use your code to compute the interpolating polynomial for the data below, and estimate the values of y for the given x.

a.

x	0.1	0.2	0.3	1.0	0.25
y	0.620500	-0.283987	0.006601	0.248424	

b.

x	0.4	0.5	0.6	0.7	0.8	1.0	0.9
y	-0.36770	-0.19481	-0.17695	-0.01375	0.22363	0.65809	

.

H10. Consider the following functions, interpolation points x_i and unknown x-values.

a. $f(x) = x \log x$, $x_i = \{0.3, 0.5, 0.6\}$, $x = 0.4, 0.7$.

b. $f(x) = \cos(3e^x)$, $x_i = \{0.6, 0.7, 0.8, 0.9, 1.0\}$, $x = 0.55, 0.75, 0.85$.

For each case, construct the polynomial interpolants. Use your interpolants to approximate f and f' at the value x given, and calculuate the actual error. Which approximations are more accurate, and why?

A11. For the interpolation in Question 10(a), use the error formula for polynomial interpolation to give bounds for the errors of your approximations for $f(x)$, and compare with the actual errors.

E12. Approximate $\sqrt{5}$ by interpolating the function $f(x) = 5^x$ at the values $x_0 = -2$, $x_1 = -1$, $x_2 = 0$, $x_3 = 1$ and $x_4 = 2$.

C13. Use the Matlab command `sd=spline(xs,ys)` to compute the data for the spline s interpolating the data of Question 4. Examine the data structure `s`, especially the fields `sd.breaks` and `sd.coefs`. Write down the cubic polynomial s_1 defining s on the interval $[0.6, 1.5]$, and compute $s_1(1.0)$.

Check your answer by using the Matlab command `ppval(sd,x)` to evaluate the piecewise-polynomial spline s at the interpolation points, and at $x = 1.0$. Plot a graph of s and compare your results to the polynomial interpolation. Which do you think is more accurate?

C14. Recall that for the interval $[-1,+1]$, equally-spaced nodes $x_0,\ldots,x_n$ are given by $x_i = 2i/n - 1$, and Chebyshev nodes by $x_i = -\cos((2i+1)\pi/(2n+2))$.

For each of

(i) Equally-spaced nodes and polynomial interpolation.

(ii) Chebyshev nodes and polynomial interpolation.

(iii) Equally-spaced nodes and spline interpolation.

and for $n = 8, 16, 32$, compute and plot the interpolants for the following functions and intervals:

a. For $k = n/4, n/2, n$, the Lagrange basis functions interpolating $y_k = 1$, $y_i = 0$ for $k \neq i$

b. Random data y_i taking values in $[-1,+1]$.

c. The Gaussian function $f(x) = e^{-x^2}$.

d. The function $f(x) = e^{-x^2}$ perturbed by random data taking values in $[-0.01,+0.01]$

e. The function $f(x) = e^{-(5x)^2}$.

Comment on your findings.