Lecture 1

Interpolation and Polynomial Approximation

Interpolation

- In many applications, we may need to have an approximation of some physical quantities.
- We can measure the physical quantity at some certain locations.
- To approximate the physical quantity in a domain (interval or area), we can estimate the physical quantity in between the collected data points by using the measured data.

Weiertrass Approximation Theorem

Theorem (Weiertrass Approximation Theorem)

Suppose that f is defined and continuous on $[a, b]$. For each $\epsilon > 0$, there exists a polynomial $P(x)$, with the property that

$$|f(x) - P(x)| < \epsilon, \text{ for all } x \text{ in } [a, b].$$

Taylor polynomial

By Taylor theorem, for a smooth enough function f, we have

$$
\begin{aligned}
f(x) &= \sum_{i=0}^{n} \frac{f^{(i)}(x_0)}{i!}(x-x_0)^i + O(|x-x_0|^{n+1}) \\
&= f(x_0) + f'(x_0)(x-x_0) + \frac{f''(x_0)}{2!}(x-x_0)^2 + \cdots + O(|x-x_0|^{n+1}).
\end{aligned}
$$

For example, $f = e^x$, we can approximate f by

$$
\begin{aligned}
&P_0(x) = e^0 = 1,\ P_1(x) = e^0 + e^0 x = 1 + x \\
&P_2(x) = 1 + x + \frac{x^2}{2},\ P_3(x) = 1 + x + \frac{x^2}{2} + \frac{x^3}{6}
\end{aligned}
$$

One of the method for approximating a function is to find polynomial which fit the data at some given points. This process is called Interpolation.

Given some points $x_0, x_1, x_2, x_3, \dots, x_n \in \mathbb{R}$ and the value of f at those points which are $f(x_0), f(x_1), \dots, f(x_n)$, we would like to find a polynomial $P(x)$ such that

$$P(x_i) = f(x_i) \; \forall i = 0, 1, \dots, n.$$

Example

x	0	1
$f(x)$	2	1

  We consider the polynomial $P(x) = 2 - x$ and we have

$$P(0) = 2 = f(0),$$
$$P(1) = 2 - 1 = 1 = f(1)$$

x	0	1	2
$f(x)$	1	2	0

  We consider the polynomial $P(x) = -\frac{3}{2}x^2 + \frac{5}{2}x + 1$ and we have

$$P(0) = 1 = f(0),$$
$$P(1) = -\frac{3}{2} + \frac{5}{2} + 1 = 2 = f(1)$$
$$P(2) = -\frac{3}{2} \cdot 4 + \frac{5}{2} \cdot 2 + 1 = 0 = f(2)$$

Finding the polynomial

The question is how to find the polynomial P which fits the data, i.e. satisfies

$$P(x_i) = f(x_i) \ \forall i = 0, 1, \ldots n.$$

For any polynomial, P, of degree k, P can be written in the following form

$$P(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_k x^k.$$

If P satisfies $P(x_i) = f(x_i) \ \forall i = 1, 2, \ldots n$, we have

$$a_0 + a_1 x_i + a_2 x_i^2 + \cdots + a_k x_i^k = f(x_i) \ \forall i = 0, 1, \ldots, n$$

Cont.

Therefore, to find P, we can solve the linear system

$$\begin{pmatrix} 1 & x_0 & \cdots & x_0^k \\ 1 & x_1 & \cdots & x_1^k \\ 1 & \vdots & \ddots & \vdots \\ 1 & x_n & \cdots & x_n^k \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_k \end{pmatrix} = \begin{pmatrix} f(x_0) \\ f(x_1) \\ \vdots \\ f(x_n) \end{pmatrix}$$

When will the system be solvable?
Clearly, at least, we need $k \geq n$ to make sure the system have a solution for any given data.
In fact, we check that the system has a unique solution if $k = n$ and $x_i \neq x_j$ for $i \neq j$.

Example

x 1 6

y 1 7

x	0	1
$f(x)$	2	1

We will solve

$$\begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \end{pmatrix} = \begin{pmatrix} 2 \\ 1 \end{pmatrix}$$

and obtain $a_0 = 2$ and $a_1 = -1$ and thus, we have

$f(4) = 4.8 - 0.2$

$$P(x) = -x + 2.$$

.

$(6, 7)$

$(1, 1)$ $\frac{6}{5}$ $1.2x - 0.2 = f(x)$

Lagrange Polynomial

For a given set of points $(x_0, f(x_0)), \ldots, (x_n, f(x_n))$ with distinct x_i, the n-th Lagrange interpolating polynomial is the polynomial of lowest degree which satisfies $P(x_i) = f(x_i)\ \forall i = 0, 1, \ldots, n$.
Instead of solving the linear system to obtain the coefficient a_i. There is another way to find the Lagrange polynomial P satisfying

$$P(x_i) = f(x_i)\ \forall i = 0, 1, \ldots n.$$

We can consider the polynomial P is written as a sum of polynomials of degree n such that

$$P(x) = \sum_{i=0}^{n} f(x_i) L_i(x)$$

where L_i is a polynomial of degree n satisfying

$$L_i(x_j) = \delta_{ij} = \begin{cases} 1 & \text{if } x_i = x_j \\ 0 & \text{if } x_i \neq x_j \end{cases}$$

Cont.

If we can find such L_i, we can check that

$$
\begin{aligned}
P(x_j) &= \sum_{i=0}^{n} f(x_i)L_i(x_j) \\
&= 0 \cdot f(x_0) + \cdots + 0 \cdot f(x_{j-1}) + 1 \cdot f(x_j) + 0 \cdot f(x_{j+1}) + \cdots + 0 \cdot f(x_n) \\
&= f(x_j).
\end{aligned}
$$

Therefore, P is the interpolation polynomial for the given data.
Next question is how to find the polynomial L_i for given $x_0, \ldots, x_n$.
In fact, it is not hard to find L_i. We can check

$$L_i = \prod_{j \neq i, j=1}^{n} \frac{(x - x_j)}{(x_i - x_j)}$$

satisfies the constrain.

Example

x	0	1	2
$f(x)$	1	2	0

  We have

$$L_0 = \frac{(x-1)}{(0-1)} \cdot \frac{(x-2)}{(0-2)} = \frac{1}{2}(x^2-3x+2),$$
$$L_1 = \frac{(x-0)}{(1-0)} \cdot \frac{(x-2)}{(1-2)} = -(x^2-2x)$$
$$L_2 = \frac{(x-0)}{(2-0)} \cdot \frac{(x-1)}{(2-1)} = \frac{1}{2}(x^2-x)$$

and

$$\begin{aligned} P(x) &= 1 \cdot L_0(x) + 2 \cdot L_1(x) + 0 \cdot L_2(x) \\ &= \frac{1}{2}(x^2-3x+2) - 2(x^2-2x) \\ &= -\frac{3}{2}x^2 + \frac{5}{2}x + 1 \end{aligned}$$

Lecture 2

Lagrange Interpolation

Review

Given some data points $(x_0, f(x_0)), (x_1, f(x_1)), \ldots, (x_n, f(x_n))$ with $x_i \in [a, b]$, to estimate the function value of f in $[a, b]$, we can approximate the function f by a polynomial of degree $k + 1$, namely,

$$f(x) \approx P(x) := a_0 + a_1 x + a_2 x^2 + \cdots + a_k x^k, \ \forall x \in [a, b].$$

One of possible criteria for the polynomial P is $P(x_i) = f(x_i)$ for all $i = 0, 1, \ldots, n$.
That is, We are going to seek for a polynomial P which fit the data exactly at $x = x_0, \ldots, x_n$.

Review

We can prove that there exist a unique polynomial P of degree at most n which fits the data point $(x_0, f(x_0)), (x_1, f(x_1)), \dots, (x_n, f(x_n))$.
One of the way to find this polynomial P is to solve a linear system

$$\begin{pmatrix} 1 & x_0 & \cdots & x_0^n \\ 1 & x_1 & \cdots & x_1^n \\ 1 & \vdots & \ddots & \vdots \\ 1 & x_n & \cdots & x_n^n \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_n \end{pmatrix} = \begin{pmatrix} f(x_0) \\ f(x_1) \\ \vdots \\ f(x_n) \end{pmatrix}$$

and P is then defined by

$$P(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_n x^n.$$

Lagrange Polynomial

For a given set of points $(x_0, f(x_0)), \ldots, (x_n, f(x_n))$ with distinct x_i, the n-th Lagrange interpolating polynomial is the polynomial of lowest degree which satisfies $P(x_i) = f(x_i)\ \forall i = 0, 1, \ldots, n$.
Instead of solving the linear system to obtain the coefficient a_i.
There is another way to find the Lagrange interpolating polynomial P.
We can consider the polynomial P is written as a sum of polynomials L_i of degree n, namely,

$$P(x) = \sum_{i=0}^{n} f(x_i) L_i(x)$$

where L_i is a polynomial of degree n satisfying

$$L_i(x_j) = \delta_{ij} = \begin{cases} 1 & \text{if } x_i = x_j \\ 0 & \text{if } x_i \neq x_j \end{cases}$$

Cont.

If we can find such L_i, we can check that

$$\begin{aligned}
P(x_j) &= \sum_{i=0}^{n} f(x_i)L_i(x_j) \\
&= 0 \cdot f(x_0) + \cdots + 0 \cdot f(x_{j-1}) + 1 \cdot f(x_j) + 0 \cdot f(x_{j+1}) + \cdots + 0 \cdot f(x_n) \\
&= f(x_j).
\end{aligned}$$

Therefore, P is the interpolation polynomial for the given data.
Next question is how to find the polynomial L_i for given $x_0, \ldots, x_n$.
In fact, it is not hard to find L_i. We can check

$$L_i = \prod_{j \neq i, j=1}^{n} \frac{(x - x_j)}{(x_i - x_j)}$$

satisfies the constrains.

Example

x	0	1	2
$f(x)$	1	2	0

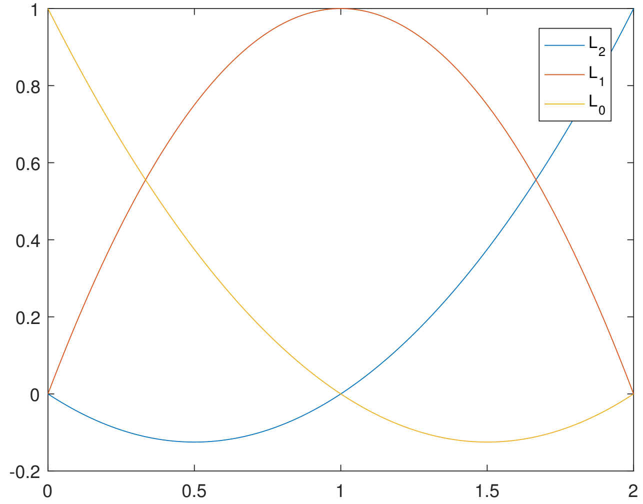
  We have

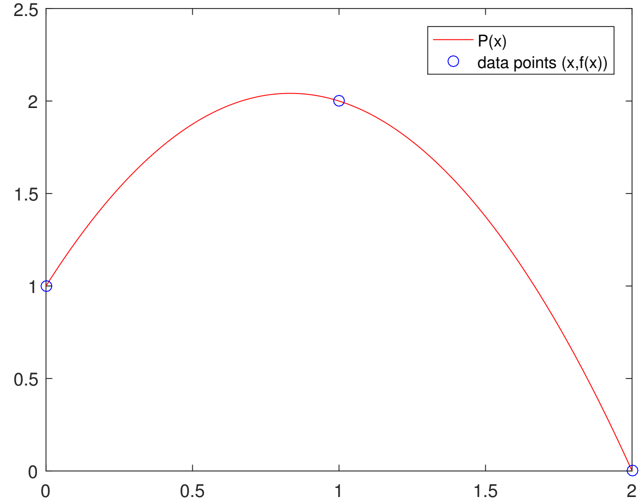
$$L_0 = \frac{(x-1)}{(0-1)} \cdot \frac{(x-2)}{(0-2)} = \frac{1}{2}(x^2 - 3x + 2),$$
$$L_1 = \frac{(x-0)}{(1-0)} \cdot \frac{(x-2)}{(1-2)} = -(x^2 - 2x)$$
$$L_2 = \frac{(x-0)}{(2-0)} \cdot \frac{(x-1)}{(2-1)} = \frac{1}{2}(x^2 - x)$$

and

$$\begin{aligned} P(x) &= 1 \cdot L_0(x) + 2 \cdot L_1(x) + 0 \cdot L_2(x) \\ &= \frac{1}{2}(x^2 - 3x + 2) - 2(x^2 - 2x) \\ &= -\frac{3}{2}x^2 + \frac{5}{2}x + 1 \end{aligned}$$

Graph of L_i and P





Theorem

Theorem

If $x_0, x_1, \ldots, x_n$ are $n+1$ distinct numbers and f is a function whose values are given at these numbers, then an unique polynomial $P(x)$ of degree at most n exists with

$$f(x_i) = P(x_i), \forall\ i = 0, 1, ..., n.$$

Proof.

We can easily check $P(x) = \sum_{i=0}^{n} f(x_i) L_i(x)$ with $L_i(x) = \prod_{j\neq i, j=1}^{n} \dfrac{(x - x_j)}{(x_i - x_j)}$ satisfies

$$f(x_i) = P(x_i), \forall\ i = 0, 1, ..., n.$$

Assuming there is another polynomial $\tilde{P}$ of degree at most n satisfies $f(x_i) = P_2(x_i), \forall\ i = 0, 1, ..., n$, we have $P - \tilde{P}$ is an another polynomial of degree at most n and

$$P(x_i) - \tilde{P}(x_i) = 0\ \forall i = 0, 1, \ldots, n$$

By Fundamental theorem of algebra, we have $P - \tilde{P} = 0$.

$\square$

Example: Linear Lagrange interpolation

For example, given two data points $(x_0, f(x_0)), (x_1, f(x_1))$, we have

$$L_0(x) = \frac{x - x_1}{x_0 - x_1}, \; L_1(x) = \frac{x - x_0}{x_1 - x_0}$$

The linear Lagrange interpolating polynomial P is defined by

$$P(x) = f(x_0)\frac{x - x_1}{x_0 - x_1} + f(x_1)\frac{x - x_0}{x_1 - x_0}$$

and the graph of P is a straight line joining $(x_0, f(x_0))$ and $(x_1, f(x_1))$.

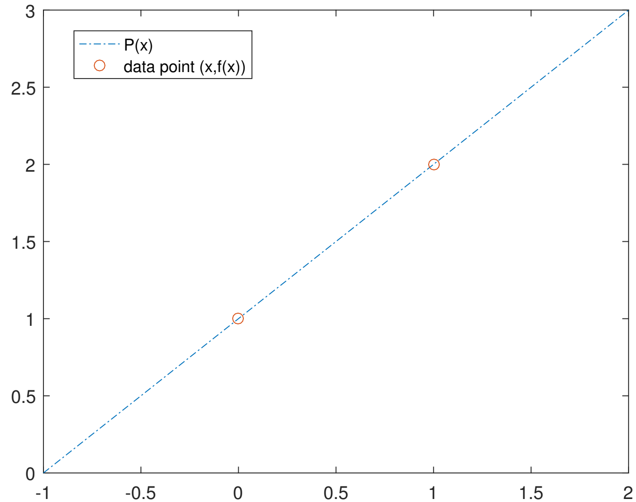
Cont.

Given data:

x	0	1
$f(x)$	1	2

The linear Lagrange interpolating polynomial P is defined by

$$\begin{aligned} P(x) &= 1 \cdot \frac{x-1}{0-1} + 2 \cdot \frac{x-0}{1-0} \\ &= x + 1 \end{aligned}$$



Error of approximation

Theorem

Suppose $x_0, x_1, \ldots, x_n$ are distinct numbers in the interval $[a, b]$ and $f \in C^{n+1}[a, b]$. Then, for each x in $[a, b]$, a number $\xi(x)$ (generally unknown) between $x_0, x_1, \ldots, x_n$ and hence in (a, b), exists with

$$f(x) = P(x) + \frac{f^{(n+1)}(\xi(x))}{(n+1)!}(x - x_0)(x - x_1)\cdots(x - x_n),$$

where $P(x)$ is the Lagrange interpolating polynomial.

We remark that if we use Taylor polynomial at point x_0 to approximate a function, the error term will be

$$\frac{f^{(n+1)}(\tilde{\xi}(x))}{(n+1)!}(x - x_0)^n$$

We obverse that they have a similar error term. By comparing two error terms, we know the Taylor polynomial has a better approximation power near the point x_0 while the Lagrange interpolating polynomial has a more uniform performance between the data points $x_0, x_1, \ldots, x_n$.

Proof

Proof.

First, if $x = x_i$ for some $i = 0, 1, \ldots, n$, we have

$$f(x) - P(x) = 0 = \frac{f^{(n+1)}(\xi(x))}{(n+1)!}(x-x_0)(x-x_1)\cdots(x-x_n) \text{ for any } \xi(x).$$

Suppose $x \neq x_i$ for $i = 0, 1, \ldots, n$. We consider g as a function of t defined by

$$\begin{aligned} g(t) &:= f(t) - P(t) - [f(x) - P(x)]\frac{(t-x_0)}{(x-x_0)}\frac{(t-x_1)}{(x-x_1)}\cdots\frac{(t-x_n)}{(x-x_n)} \\ &= f(t) - P(t) - [f(x) - P(x)]\prod_{i=0}^{n}\frac{(t-x_i)}{(x-x_i)}. \end{aligned}$$

Since $f \in C^{n+1}([a,b])$ and $P \in C^{n+1}([a,b])$, we have $g \in C^{n+1}([a,b])$. $\square$

Proof

Proof.

If $t = x_j$, we have

$$g(x_k) = f(x_j) - P(x_j) - [f(x) - P(x)] \prod_{i=0}^{n} \frac{(x_j - x_i)}{(x - x_i)}$$
$$= 0 - [f(x) - P(x)] \cdot 0 = 0$$

Moreover, if $t = x$, we have

$$g(x) = f(x) - P(x) - [f(x) - P(x)] \prod_{i=0}^{n} \frac{(x - x_i)}{(x - x_i)}$$
$$= f(x) - P(x) - [f(x) - P(x)] \cdot 1 = 0$$

Thus, g has $n+2$ roots $x, x_0, x_1, \ldots, x_n$ in $[a, b]$. By Generalized Rolle's Theorem, there exists a $\xi \in (a, b)$ such that

$$0 = g^{(n+1)}(\xi) = f^{(n+1)}(\xi) - P^{(n+1)}(\xi) - [f(x) - P(x)] \frac{d^{n+1}}{dt^{n+1}} \Big(\prod_{i=0}^{n} \frac{(t - x_i)}{(x - x_i)} \Big).$$

Proof

Proof.

Since P is a polynomial of degree at most n, we have $P^{(n+1)} \equiv 0$.
Since $\prod_{i=0}^{n}(t - x_i) = t^{n+1} + b_n t^n + \cdots + b_0$, we have

$$\frac{d^{n+1}}{dt^{n+1}}\Big(\prod_{i=0}^{n}\frac{(t-x_i)}{(x-x_i)}\Big) = \frac{1}{\prod_{i=0}^{n}(x-x_i)}\frac{d^{n+1}}{dt^{n+1}}\Big(\prod_{i=0}^{n}(t-x_i)\Big) = \frac{\frac{d^{n+1}}{dt^{n+1}}\Big(t^{n+1}\Big)}{\prod_{i=0}^{n}(x-x_i)}$$

$$= \frac{(n+1)!}{\prod_{i=0}^{n}(x-x_i)}.$$

Thus, we have

$$f(x) - P(x) = f^{(n+1)}(\xi)\frac{\prod_{i=0}^{n}(x-x_i)}{(n+1)!}$$

$\square$

Example

For $f = x^3 - \frac{9}{2}x^2 + \frac{9}{2}x + 1$ and $x = 0, 1, 2$. We have

x	0	1	2
$f(x)$	1	2	0

and

$$P(x) = 1 \cdot L_0(x) + 2 \cdot L_1(x) + 0 \cdot L_2(x) = -\frac{3}{2}x^2 + \frac{5}{2}x + 1$$

We have

$$\begin{aligned} f(x) - P(x) &= x^3 - 3x^2 + 2x = x(x-1)(x-2) \\ &= \frac{f^{(3)}(\xi)}{3!}x(x-1)(x-2) \end{aligned}$$

Since $\frac{d}{dx}(x(x-1)(x-2)) = 3x^2 - 6x + 2$, we have
$\max_{x\in[0,2]}\{|x(x-1)(x-2)|\} = \frac{\sqrt{3}}{3}(1 - \frac{\sqrt{3}}{3})(1 + \frac{\sqrt{3}}{3}) = \frac{2\sqrt{3}}{9}$.
Thus, we have

$$|f(x) - P(x)| \leq \frac{|f^{(3)}(\xi)|}{3!}\frac{2\sqrt{3}}{9} = \frac{2\sqrt{3}}{9}\ \forall x \in [0,2]$$

Example

For $f = e^x$ and $x = 0, 1, 2$. We have

x	0	1	2
$f(x)$	1	e	e^2

and

$$P(x) = 1 \cdot L_0(x) + e \cdot L_1(x) + e^2 \cdot L_2(x) \approx 1.4762x^2 + 0.2420x + 1$$

We have

$$f(x) - P(x) = \frac{f^{(3)}(\xi(x))}{3!}(x^3 - 3x^2 + 2x) = \frac{e^{\xi(x)}}{3!}x(x-1)(x-2)$$

Thus, we have

$$\begin{aligned} |f(x) - P(x)| &\leq \max_{\xi \in [0,2]} \frac{|f^{(3)}(\xi)|}{3!} \max_{x \in [0,2]} |x^3 - 3x^2 + 2x| \\ &= \frac{e^2}{3!}\frac{2\sqrt{3}}{9} \approx 0.4740 \ \forall x \in [0, 2] \end{aligned}$$

Lecture 3

Piece-wise Interpolation and Neville's Method

Piece-wise Interpolation

- Given a data set $\{(x_i, f(x_i))\}_{i=0}^{n}$, we can use Lagrange interpolation to obtain a polynomial, P, of degree at most n satisfying $P(x_i) = f(x_i)$.
- However, sometimes this approach may not give us a good approximation of the function f when even if there are many data points, since the Lagrange interpolation is "unstable" when n is large.
- Therefore, in some situations, we may want to obtain a lower degree polynomial to approximate the function f.
- We know that it is impossible to find a lower degree polynomial to fit all of the possible data points.

Cont.

One of the approaches is using a piece-wise polynomial function P to approximate f.
That is, for each sub-interval $[x_{i-1}, x_i]$, we consider P is a polynomial of degree $k < n$.
Namely, $\mathbf{P(x) = P_i(x)}$ for $\mathbf{x \in [x_{i-1}, x_i]}$ where P_i is a polynomial of degree $\mathbf{k < n}$.
Since the interpolating polynomial P is defined piecewisely, we do not need a single polynomial which fit all of the data.
For each sub-interval $[x_{i-1}, x_i]$, we only need the interpolating polynomial $P = P^{(i)}$ fit the data at two end points. That is $P^{(i)}(x_i) = f(x_i)$ and $P^{(i)}(x_{i-1}) = f(x_{i-1})$.

Piece-wise linear Interpolation

The simplest way to obtain a piece-wise polynomial interpolation is considering P is piece-wise linear function. $(k = 1)$
Since, for each sub-interval $[x_{i-1}, x_i]$, there are only two equations corresponding to two end points. We can define the linear function P_i uniquely. The piece-wise linear function P defined by $P(x) = P_i(x)$ for $x \in [x_{i-1}, x_i]$ is called a piece-wise linear interpolation.

Example

For given data

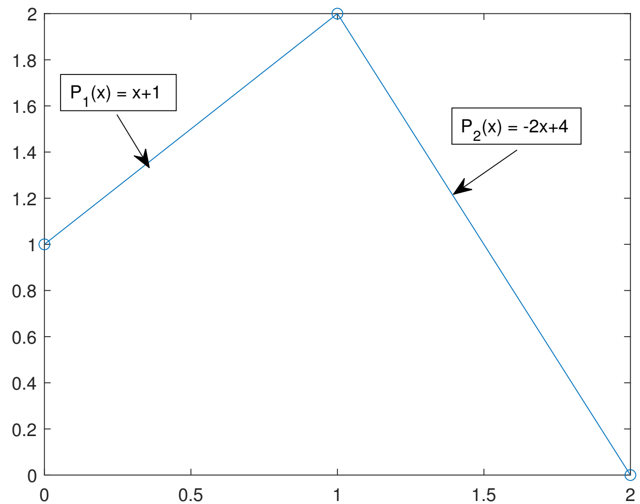
x	0	1	2
$f(x)$	1	2	0

,

we have

$$P_1 = 1 \cdot \frac{x-1}{0-1} + 2 \cdot \frac{x-0}{1-0} = x+1$$

$$P_2 = 2 \cdot \frac{x-2}{1-2} + 0 \cdot \frac{x-1}{2-1} = -2x+4$$



How to get a piece-wise polynomial P with degree larger than 1?
One of the possible way is fitting the data at more points. For example, we can use the data at x_{i-2}, x_{i-1}, x_i to define the piece-wise polynomial in $[x_{i-1}, x_i]$. Namely, we define the polynomial P_i of degree at most 2 by fitting

$$P_i(x_i) = f(x_i),\ P_i(x_{i-1}) = f(x_{i-1}),\ P_i(x_{i-2}) = f(x_{i-2}).$$

It is not hard to check the piecewise polynomial P defined using these P_i satisfies $P(x_i) = f(x_i)$ for all $x = 0, \ldots, 1$.

Cont.

It is obvious you can also use the data x_{i-1}, x_i, x_{i+1} to define the piece-wise polynomial in $[x_{i-1}, x_i]$.
We denote the polynomial of degree at $k-1$ which fits the data at point $x_{m_1}, x_{m_2}, \ldots, x_{m_k}$ as $P_{m_1,m_2,\ldots,m_k}$.
We have at least two choices of polynomial of degree at most 2 to approximate f in $[x_{i-1}, x_i]$ which are $P_{i-1,i,i+1}$ and $P_{i-2,i-1,i}$.
Then, which additional point give us a better approximate of f at x?

Interpolation of $P_{i-1,i,i+1}$ *and* $P_{i-2,i-1,i}$

Of course, the best additional point gives a polynomial of degree at most 2 approximate of f at x is x itself.
If the polynomial is defined by fitting the function f at x, x_{i-1}, x_i, we have $P(x) = f(x)$.
However, normally, the problem is that we do not have the data at x.
Then, one of the possible way is preform a linear interpolation of $P_{i-1,i,i+1}$ and $P_{i-2,i-1,i}$ to approximate the polynomial defined by fitting the data at x, x_{i-1}, x_i.

Cont.

We can consider an operator g mapping a number y to a polynomial of degree at most 2 which fits the data at y, x_{i-1}, x_i.
Then we have $g(x_{i-2}) = P_{i-2,i-1,i}$ and $g(x_{i+1}) = P_{i-1,i,i+1}$. To approximate $g(x)$, we consider the linear interpolation of g by using the data $(x_{i-2}, P_{i-2,i-1,i})$ and $(x_{i+1}, P_{i-1,i,i+1})$ which gives

$$\begin{aligned} g(x) &\approx \frac{x - x_{i+1}}{x_{i-2} - x_{i+1}} P_{i-2,i-1,i} + \frac{x - x_{i-2}}{x_{i+1} - x_{i-2}} P_{i-1,i,i+1} \\ &= \frac{x - x_{i+1}}{x_{i-2} - x_{i+1}} P_{i-2,i-1,i} - \frac{x - x_{i-2}}{x_{i-2} - x_{i+1}} P_{i-1,i,i+1} \end{aligned}$$

Neville's method

This approach will provide a polynomial, P, of degree at most 3 defined by

$$P(x) = \frac{x - x_{i+1}}{x_{i-2} - x_{i+1}} P_{i-2,i-1,i}(x) - \frac{x - x_{i-2}}{x_{i-2} - x_{i+1}} P_{i-1,i,i+1}(x)$$

which fit the data at $x_{i-2}, x_{i-1}, x_i, x_{i+1}$.

Therefore, it is a iterative way to construct the polynomial fitting more and more data.

Theorem

Theorem

Let f be defined at $x_0, x_1, \ldots, x_k$ and let x_j and x_i are two distinct numbers. Then

$$P(x) = \frac{(x - x_j)P_{0,1,\ldots,j-1,j+1,\ldots,k}(x) - (x - x_i)P_{0,1,\ldots,i-1,i+1,\ldots,k}(x)}{x_i - x_j}$$

is the $k-th$ Lagrange polynomial that interpolates f at points $x_0, x_1, \ldots, x_k$

Lecture 4

Neville's Method and Divided-difference

Neville's Method

Theorem

Let f be defined at $x_0, x_1, \ldots, x_k$ and let x_j and x_i are two distinct numbers. Then

$$P(x) = \frac{(x - x_j)P_{0,1,\ldots,j-1,j+1,\ldots,k}(x) - (x - x_i)P_{0,1,\ldots,i-1,i+1,\ldots,k}(x)}{x_i - x_j}$$

is the $k-th$ Lagrange polynomial that interpolates f at points $x_0, x_1, \ldots, x_k$

Proof

First we have $f(x) = \dfrac{[(x - x_j) - (x - x_i)]f(x)}{x_i - x_j} = \dfrac{(x - x_j)f(x) - (x - x_i)f(x)}{x_i - x_j}$.

Thus, we have

$$P(x) - f(x) = \frac{(x - x_j)[P_{0,1,\ldots,j-1,j+1,\ldots,k}(x) - f(x)] - (x - x_i)[P_{0,1,\ldots,i-1,i+1,\ldots,k}(x) - f(x)]}{x_i - x_j}.$$

For $x = x_s$, $s \neq i, j$, we have

$$P_{0,1,\ldots,j-1,j+1,\ldots,k}(x) - f(x) = 0, \text{ and } P_{0,1,\ldots,i-1,i+1,\ldots,k}(x) - f(x) = 0$$

For $x = x_i$, we have

$$P_{0,1,\ldots,j-1,j+1,\ldots,k}(x) - f(x) = 0, \text{ and } (x - x_i) = 0.$$

For $x = x_j$, we have

$$P_{0,1,\ldots,i-1,i+1,\ldots,k}(x) - f(x) = 0, \text{ and } (x - x_j) = 0.$$

Therefore, $P(x_s) - f(x_s) = 0$ if $s = 0, 1, , \ldots, k$

Neville's method

Therefore, we can generate the higher degree Lagrange interpolating polynomial iteratively.

x_0	$P_0 = f(x_0)$			
x_1	$P_1 = f(x_1)$	$P_{0,1}$		
x_2	$P_2 = f(x_2)$	$P_{1,2}$	$P_{0,1,2}$	
x_3	$P_3 = f(x_3)$	$P_{2,3}$	$P_{1,2,3}$	$P_{0,1,2,3}$

Cont.

To simplify the notation, we denote $Q_{i,j} = P_{i-j,i-j+1,\ldots,i-1,i}$. We have the following table:

x_0	$P_0 = Q_{0,0}$			
x_1	$P_1 = Q_{1,0}$	$P_{0,1} = Q_{1,1}$		
x_2	$P_2 = Q_{2,0}$	$P_{1,2} = Q_{2,1}$	$P_{0,1,2} = Q_{2,2}$	
x_3	$P_3 = Q_{3,0}$	$P_{2,3} = Q_{3,1}$	$P_{1,2,3} = Q_{3,2}$	$P_{0,1,2,3} = Q_{3,3}$

x_i	1	3	4	6
y_i	0	1	3	-2

To calculate $P(2)$, we can first copy the data to the table

1	0
3	1
4	3
6	-2

Example

x_i	1	3	4	6
y_i	0	1	3	-2

Next, we calculate the linear interpolations.

1	0	
3	1	$\dfrac{(2-1)\cdot 1-(2-3)\cdot 0}{(3-1)}=\dfrac{1}{2}$
4	3	$\dfrac{(2-3)\cdot 3-(2-4)\cdot 1}{(4-3)}=-1$
6	-2	$\dfrac{(2-4)\cdot(-2)-(2-6)\cdot 3}{(6-4)}=8$

Example

x_i	1	3	4	6
y_i	0	1	3	-2

Next, we calculate the quadratic interpolations.

$$
\begin{array}{cccc}
\hline
1 & 0 & & \\
3 & 1 & \frac{1}{2} & \\
4 & 3 & -1 & \frac{(2-1)\cdot(-1)-(2-4)\cdot(0.5)}{4-1} = 0 \\
6 & -2 & 8 & \frac{(2-3)\cdot(8)-(2-6)\cdot(-1)}{6-3} = -4 \\
\hline
\end{array}
$$

Example

x_i	1	3	4	6
y_i	0	1	3	-2

Finally, we calculate the cubic interpolations and obtain $P(2) = -0.8$.

$$
\begin{array}{ccccc}
\hline
1 & 0 & & & \\
3 & 1 & \frac{1}{2} & & \\
4 & 3 & -1 & 0 & \\
6 & -2 & 8 & -4 & \frac{(2-1)\cdot(-4)-(2-6)\cdot(0)}{6-1} = -\frac{4}{5} \\
\hline
\end{array}
$$

Input:	a number x, a set of data point $(x_i, f(x_i))_{i=0}^{n}$
Output:	the table Q with $P(x) = Q_{n,n}$
Step 1:	For $i = 0, \ldots, n$, $Q_{i,0} = f(x_i)$
Step 2:	For $i = 1, \ldots, n$, For $j = 1, 2, \ldots, i$, set $Q_{i,j} = \dfrac{(x - x_{i-j})Q_{i,j-1} - (x - x_i)Q_{i-1,j-1}}{x_i - x_{i-j}}$
Step 3:	OUTPUT Q

The algorithm can be modified to allow for the addition of new interpolating nodes.

We can also add a stopping criterion $|Q_{i,i} - Q_{i-1,i-1}| < \epsilon$ where ϵ is a prescribed error tolerance.

Divided Differences

Neville's method gives us a way to calculate the higher degree polynomial at a specific point.
Sometimes, we may need to evaluate the interpolating polynomial repeatedly.
For each evaluation, Neville's method require $O(n^2)$ operations. If we need to evaluate the interpolating polynomial at m points, it requires $O(mn^2)$ operations.
We consider the polynomial is written in a linear combination of $\{1, x, x^2, \ldots, x^n\}$, that is $P(x) = a_0 + a_1 x + \cdots + a_n x^n$.
By using Horner's Method, we can evaluate the polynomial by using $O(n)$ operations.
However, to find the coefficients a_i, we may need $O(n^3)$ operations.
Therefore, we need $O(n^3 + mn)$ to evaluate the interpolating polynomial at m points.

Divided Differences

Is there an efficient way for both obtaining the "coefficients" and evaluating the interpolating polynomial?
In fact, we use a iterative way to generate the interpolating polynomial P.
We consider the interpolating polynomial P is written as a linear combination of

$$\mathcal{N}_n = \{1, (x - x_0), (x - x_0)(x - x_1), \ldots, \prod_{i=0}^{n-1}(x - x_i)\}.$$

That is,

$$P(x) = b_0 + b_1(x - x_0) + b_2(x - x_0)(x - x_1) + \cdots + b_n \prod_{i=0}^{n-1}(x - x_i).$$

(Since $\mathcal{N}_n$ is a linear independent set and there are n polynomials of degree at most n in $\mathcal{N}_n$, for any polynomial P of degree at most n, P can be written in the above form which called Newton's form of the polynomial)

Divided Differences

For

$$Q_{n,n} = b_0 + b_1(x - x_0) + b_2(x - x_0)(x - x_1) + \cdots + b_n \prod_{i=0}^{n-1}(x - x_i),$$

we have

$$\begin{aligned} f(x_s) = Q_{n,n}(x_s) =& b_0 + b_1(x_s - x_0) + b_2(x_s - x_0)(x_s - x_1) + b_{n-1}\prod_{i=0}^{n-2}(x_s - x_i) + b_n \prod_{i=0}^{n-1}(x_s - x_i) \\ =& b_0 + b_1(x_s - x_0) + b_2(x_s - x_0)(x_s - x_1) + b_{n-1}\prod_{i=0}^{n-2}(x_s - x_i) \end{aligned}$$

for $s = 0, 1, \ldots, n-1$. Thus, we have

$$Q_{n-1,n-1}(x) = b_0 + b_1(x - x_0) + b_2(x - x_0)(x - x_1) + b_{n-1}\prod_{i=0}^{n-2}(x - x_i)$$

and $Q_{n,n}(x) = Q_{n-1,n-1}(x) + b_n \prod_{i=0}^{n-1}(x - x_i)$.

Cont.

Inductively, we have

$$Q_{k,k}(x) = b_0 + b_1(x - x_0) + b_2(x - x_0)(x - x_1) + \cdots + b_k \prod_{i=0}^{k-1}(x - x_i)$$

and $Q_{k,k}(x) = Q_{k-1,k-1}(x) + b_k \prod_{i=0}^{k-1}(x - x_i)$ for all k.
Thus, we can find the coefficient b_i inductively.
Considering the $k-$th derivative of $Q_{k,k}$, we have

$$Q_{k,k}^{(k)} \equiv (k!)b_k.$$

Since $Q_{s,k} = \dfrac{(x - x_{s-k})Q_{s,k-1}(x) - (x - x_s)Q_{s-1,k-1}}{x_s - x_{s-k}}$, we have

$$Q_{s,k}^{(k)} = k\left(\frac{Q_{s,k-1}^{(k-1)} - Q_{s-1,k-1}^{(k-1)}}{x_s - x_{s-k}}\right).$$

Cont.

We introduce the divided difference notation. The zeroth divided difference of function f with respect to x_i, denoted $f[x_i]$, is simply the value of f at x_i :

$$f[x_i] = f(x_i) = Q_{i,0}^{(0)}.$$

The first divided difference of f with respect to x_i and x_{i+1} is denoted $f[x_i, x_{i+1}]$ and defined as

$$f[x_i, x_{i+1}] = \frac{f[x_{i+1}] - f[x_i]}{x_{i+1} - x_i} = \frac{Q_{i+1,0}^{(0)} - Q_{i,0}^{(0)}}{x_{i+1} - x_i} = Q_{i+1,1}^{(1)}.$$

The second divided difference of f, $f[x_i, x_{i+1}, x_{i+2}]$, is defined and defined as

$$f[x_i, x_{i+1}, x_{i+2}] = \frac{f[x_{i+1}, x_{i+2}] - f[x_i, x_{i+1}]}{x_{i+2} - x_i} = \frac{Q_{i+2,1}^{(1)} - Q_{i+1,1}^{(1)}}{x_{i+2} - x_i} = \frac{Q_{i+2,2}^{(2)}}{2!}.$$

Divided Differences

Assuming $f[x_i, x_{i+1}, \ldots, x_{i+k-1}]$ and $f[x_{i+1}, x_{i+2}, \ldots, x_{i+k}]$ are the $(k-1)$-th divided differences, we define the k-th divided differences $f[x_i, x_{i+1}, \ldots, x_{i+k}]$ as

$$
\begin{aligned}
f[x_i, x_{i+1}, \ldots, x_{i+k}] &= \frac{f[x_{i+1}, x_{i+2}, \ldots, x_{i+k}] - f[x_i, x_{i+1}, \ldots, x_{i+k-1}]}{x_{i+k} - x_i} \\
&= \frac{Q^{(k-1)}_{i+k,k-1} - Q^{(k-1)}_{i+k-1,k-1}}{(k-1)!(x_{i+k} - x_i)} = \frac{Q^{(k)}_{i+k,k}}{k!}.
\end{aligned}
$$

Thus, we have

$$
P(x) = f[x_0] + f[x_0, x_1](x - x_0) + \cdots + f[x_0, x_1, \ldots, x_n] \prod_{i=0}^{k-1} (x - x_i).
$$

Lecture 5

Divided-difference

Divided-difference

The zeroth divided differences $f[x_i]$ is defined as the function value of f at x_i. that is,

$$f[x_i] = f(x_i).$$

Assuming $f[x_i, x_{i+1}, \dots, x_{i+k-1}]$ and $f[x_{i+1}, x_{i+2}, \dots, x_{i+k}]$ are the $(k-1)$-th divided differences, we define the k-th divided differences $f[x_i, x_{i+1}, \dots, x_{i+k}]$ as

$$f[x_i, x_{i+1}, \dots, x_{i+k}] = \frac{f[x_{i+1}, x_{i+2}, \dots, x_{i+k}] - f[x_i, x_{i+1}, \dots, x_{i+k-1}]}{x_{i+k} - x_i} = \frac{Q^{(k)}_{i+k,k}}{k!}.$$

For example, the first divided differences $f[x_i, x_{i+1}]$ is defined as

$$f[x_i, x_{i+1}] = \frac{f[x_{i+1}] - f[x_i]}{x_{i+1} - x_i} = Q^{(1)}_{i+1,1}.$$

Cont.

The interpolation polynomial P can be written as

$$P(x) = f[x_0] + f[x_0, x_1](x - x_0) + \cdots + f[x_0, x_1, \ldots, x_n] \prod_{i=0}^{k-1} (x - x_i).$$

Therefore, we can generate the higher degree Lagrange interpolating polynomial iteratively.

x_0	$f[x_0] = f(x_0)$			
x_1	$f[x_1] = f(x_1)$	$f[x_0, x_1]$		
x_2	$f[x_2] = f(x_2)$	$f[x_1, x_2]$	$f[x_0, x_1, x_2]$	
x_3	$f[x_3] = f(x_3)$	$f[x_2, x_3]$	$f[x_1, x_2, x_3]$	$f[x_0, x_1, x_2, x_3]$

Example

x_i	1	3	4	6
y_i	0	1	3	-2

We first copy the data to the table and obtain $b_0 = 0$

1	$f[x_0] = 0$
3	$f[x_1] = 1$
4	$f[x_2] = 3$
6	$f[x_3] = -2$

Cont.

x_i	1	3	4	6
y_i	0	1	3	-2

Next, we will compute the first divided differences $f[x_i, x_{i+1}]$ and obtain $b_1 = \frac{1}{2}$.

x_i	$f[x_i]$	$f[x_i, x_{i+1}]$	$f[x_i, x_{i+1}, x_{i+2}]$	$f[x_i, x_{i+1}, x_{i+2}, x_{i+3}]$
1	0			
		$f[x_0, x_1] = \frac{1-0}{3-1} = \frac{1}{2}$		
3	1			
		$f[x_1, x_2] = \frac{3-1}{4-3} = 2$		
4	3			
		$f[x_2, x_3] = \frac{(-2)-3}{6-4} = -\frac{5}{2}$		
6	-2			

Example

x_i	1	3	4	6
y_i	0	1	3	-2

Next, we will compute the second divided differences $f[x_i, x_{i+1}, x_{i+2}]$ and obtain $b_2 = \dfrac{1}{2}$.

x_i	$f[x_i]$	$f[x_i, x_{i+1}]$	$f[x_i, x_{i+1}, x_{i+2}]$	$f[x_i, x_{i+1}, x_{i+2}, x_{i+3}]$
1	0			
		$\dfrac{1}{2}$		
3	1		$f[x_0, x_1, x_2] = \dfrac{2 - \frac{1}{2}}{4 - 1} = \dfrac{1}{2}$	
		2		
4	3		$f[x_1, x_2, x_3] = \dfrac{-\dfrac{5}{2} - 2}{6 - 3} = -\dfrac{3}{2}$	
		$-\dfrac{5}{2}$		
6	-2			

Example

x_i	1	3	4	6
y_i	0	1	3	-2

Next, we will compute the third divided differences $f[x_i, x_{i+1}, x_{i+2}, x_{i+3}]$ and obtain $b_3 = -\frac{2}{5}$.

x_i	$f[x_i]$	$f[x_i, x_{i+1}]$	$f[x_i, x_{i+1}, x_{i+2}]$	$f[x_i, x_{i+1}, x_{i+2}, x_{i+3}]$
1	0			
		$\frac{1}{2}$		
3	1		$\frac{1}{2}$	
		2		$f[x_0, x_1, x_2, x_3] = \frac{-\frac{3}{2} - \frac{1}{2}}{6-1} = -\frac{2}{5}$
4	3		$-\frac{3}{2}$	
		$-\frac{5}{2}$		
6	-2			

Example

x_i	0	1	2	4
$f(x_i) = x_i^2$	0	1	4	16

x_i	$f[x_i]$	$f[x_i, x_{i+1}]$	$f[x_i, x_{i+1}, x_{i+2}]$	$f[x_i, x_{i+1}, x_{i+2}, x_{i+3}]$
0	0			
		$\dfrac{1-0}{1-0} = 1$		
1	1			
		$\dfrac{4-1}{2-1} = 3$		
2	4			
		$\dfrac{16-4}{4-2} = 6$		
4	16			

Example

x_i	0	1	2	4
$f(x_i) = x_i^2$	0	1	4	16

x_i	$f[x_i]$	$f[x_i, x_{i+1}]$	$f[x_i, x_{i+1}, x_{i+2}]$	$f[x_i, x_{i+1}, x_{i+2}, x_{i+3}]$
0	0			
		1		
1	1		$\frac{3-1}{2-0} = 1$	
		3		
2	4		$\frac{6-3}{4-1} = 1$	
		6		
4	16			

Example

x_i	0	1	2	4
$f(x_i) = x_i^2$	0	1	4	16

We have

$$P = 0 + 1\cdot(x-0) + 1\cdot(x-0)(x-1) + 0\cdot(x-0)(x-1)(x-2) = x + x(x-1) = x^2.$$

x_i	$f[x_i]$	$f[x_i, x_{i+1}]$	$f[x_i, x_{i+1}, x_{i+2}]$	$f[x_i, x_{i+1}, x_{i+2}, x_{i+3}]$
0	0			
		1		
1	1		1	
		3		$\dfrac{1-1}{4-0} = 0$
2	4		1	
		6		
4	16			

Input:	a set of data point $(x_i, f(x_i))_{i=0}^{n}$
Output:	the number $F_{0,0}, F_{1,1}, \ldots, F_{n,n}$
Step 1:	For $i = 0, \ldots, n$, $F_{i,0} = f(x_i)$
Step 2:	For $i = 1, \ldots, n$, For $j = 1, 2, \ldots, i$, set $F_{i,j} = \dfrac{F_{i,j-1} - F_{i-1,j-1}}{x_i - x_{i-j}}$
Step 3:	OUTPUT $(F_{0,0}, F_{1,1}, \ldots, F_{n,n})$

The Algorithm can be modified to add additional data points.

Theorem

Theorem

Suppose that $f \in C^n([a,b])$ *and* $x_0, x_1, \ldots, x_n$ *are distinct numbers in* $[a,b]$. *Then a number* ξ *exists in* (a,b) *with*

$$f[x_0, x_1, \ldots, x_n] = \frac{f^{(n)}(\xi)}{n!}.$$

Proof

Proof.

We consider the function $g = f - P_n$ and $g \in C^n$ where P_n is the Lagrange interpolating polynomial. Since $g(x_i) = 0$ for all $i = 0, \ldots, n$, g has $n+1$ root in $[a, b]$.
By Generalized Rolle's Theorem, there exists a $\xi \in (a, b)$ such that $g^{(n)}(\xi) = 0$.
Thus,

$$\begin{aligned} f^{(n)}(\xi) = P^{(n)}(\xi) &= \frac{d^n}{dx^n}(f[x_0] + f[x_0, x_1](x - x_0) + \cdots + f[x_0, x_1, \ldots, x_n] \prod_{i=0}^{n-1}(x - x_i))[\xi] \\ &= n! f[x_0, x_1, \ldots, x_n]. \end{aligned}$$

(The leading coefficient of $\prod_{i=0}^{n-1}(x - x_i)$ is 1.) $\square$

For $x_i = x_0 + ih$

Newton's divided-difference formula can be expressed in a simplified form when the nodes are arranged consecutively with equal spacing.
We consider the case that $x_i = x_0 + ih$ where $h > 0$ is the step size.
It is clear that we have $x_{i+1} - x_i = h$ for all $i = 0, 1, \ldots, n$.
For $x = x_0 + sh$, we have $x - x_i = (s - i)h$ and therefore, the

$$\begin{aligned}
P(x) &= P(x_0 + sh) \\
&= f[x_0] + shf[x_0, x_1] + s(s-1)h^2 f[x_0, x_1, x_2] + \cdots + \Big(\prod_{i=0}^{n-1}(s-i)\Big)h^k f[x_0, \ldots, x_n] \\
&= f[x_0] + \sum_{k=1}^{n}\Big(\Big(\prod_{i=0}^{k-1}(s-i)\Big)h^k f[x_0, x_1, \ldots, x_k]\Big).
\end{aligned}$$

Cont.

Using binomial-coefficient notation, $\binom{s}{k} = \dfrac{s(s-1)\cdots(s-k+1)}{k!}$, we have

$$P(x) = P(x_0 + sh) = f[x_0] + \sum_{k=1}^{n} k!h^k f[x_1, x_2, \ldots, x_k] \binom{s}{k}$$

Forward Differences

Using the notation Δ in Aitken's Δ^2 method, we have

$$f[x_i, x_{i+1}] = \frac{f(x_{i+1}) - f(x_i)}{h} = h^{-1}\Delta f(x_i)$$

and

$$f[x_i, x_{i+1}, x_{i+2}] = \frac{1}{2h}\frac{\Delta f(x_{i+1}) - \Delta f(x_i)}{h} = \frac{\Delta^2 f(x_i)}{2!h^2}.$$

By Introduction, we have

$$f[x_i, x_{i+1}, \ldots, x_{i+k}] = \frac{\Delta^k f(x_i)}{k!h^k}.$$

Therefore, we have

$$P(x) = P(x_0 + sh) = f(x_0) + \sum_{k=1}^{n} \begin{pmatrix} s \\ k \end{pmatrix} \Delta^k f(x_0) \text{ (Newton Forward-Difference Formula)}$$

Backward Differences

By reorder the data points, we consider the data is order corresponding to $x_n, x_{n-1}, \ldots, x_0$ and we obtain the formula

$$P(x) = f[x_n] + f[x_n, x_{n-1}](x - x_n) + \cdots + f[x_n, x_{n-1}, \ldots, x_0] \prod_{i=0}^{n-1} (x - x_{n-i}).$$

If $x_i = x_0 + ih$ and $x = x_n + sh = x_i + (s + n - i)h$, we have

$$P(x) = f[x_n] + shf[x_n, x_{n-1}](x - x_n) + \cdots + h^n f[x_n, x_{n-1}, \ldots, x_0] \prod_{i=0}^{n-1} (s + i).$$

By consider
$\begin{pmatrix} -s \\ k \end{pmatrix} = \dfrac{-s(-s-1)\cdots(-s-k+1)}{k!} = (-1)^k \dfrac{\prod_{i=0}^{k-1}(s+i)}{k!}$, we have

$$P(x) = f[x_n] + \sum_{k=1}^{n} k! h^k f[x_n, x_{n-1}, \ldots, x_{n-k}] \begin{pmatrix} -s \\ k \end{pmatrix}.$$

∇ notation

Given a sequence $\{p_n\}_{n=0}^{\infty}$, we define the backward-difference ∇p_n (read nabla p_n)by

$$\nabla p_n = p_n - p_{n-1} \text{ for } n \geq 1.$$

The higher powers are defined recursively by,

$$\nabla^k p_n = \nabla(\nabla^{k-1} p_n).$$

Backward Differences

Using the ∇ notation , we have

$$f[x_i, x_{i-1}] = \frac{f(x_i) - f(x_{i-1})}{h} = h^{-1}\nabla f(x_i)$$

and

$$f[x_i, x_{i-1}, x_{i-2}] = \frac{1}{2h}\frac{\nabla f(x_i) - \nabla f(x_{i-1})}{h} = \frac{\nabla^2 f(x_i)}{2!h^2}.$$

By Introduction, we have

$$f[x_i, x_{i-1}, \ldots, x_{i-k}] = \frac{\nabla^k f(x_i)}{k!h^k}.$$

Therefore, we have

$$P(x) = P(x_0 + sh) = f(x_0) + \sum_{k=1}^{n} \begin{pmatrix} -s \\ k \end{pmatrix} \nabla^k f(x_0) \text{ (Newton Backward-Difference Formula)}$$

Centered Differences

We consider the data is ordered as $x_0, x_{-1}, x_1, x_{-2}, x_2, \ldots, x_{-m}, x_m$. The $(2m+1)$-th interpolating polynomial P can be written as (Stirling's formula)

$$\begin{aligned}P(x) =& f[x_0] + \frac{sh}{2}(f[x_0, x_1] + f[x_0, x_{-1}]) + s^2h^2 f[x_{-1}, x_0, x_1] \\ &+ \frac{s(s^2-1)h^3}{2}(f[x_{-2}, x_{-1}, x_0, x_1] + f[x_{-1}, x_0, x_1, x_2]) \\ &+ \cdots + s^2(s^2-1)\cdots(s^2-(m-1)^2)h^{2m} f[x_{-m}, \ldots, x_m] \\ &+ \frac{s(s^2-1)\cdots(s^2-m^2)}{2} h^{2m+1}(f[x_{-m-1}, \ldots, x_m] + f[x_{-m}, \ldots, x_{m+1}])\end{aligned}$$

Example

x_i	1	3	4	6	7
y_i	0	1	3	-2	-2

$(\cdot)$: Centered Differences, $\overline{\cdot}$: Forward Differences, $\underline{\cdot}$: Backward differences

x_i	$f(x_i)$	first	second	third	fourth
x_{-2}	$\overline{f(x_{-2})}$				
		$\overline{f[x_{-2}, x_{-1}]}$			
x_{-1}	$f(x_{-1})$		$\overline{f[x_{-2}, x_{-1}, x_0]}$		
		$(f[x_{-1}, x_{-0}])$		$(\overline{f[x_{-2}, x_{-1}, x_0, x_1]})$	
x_0	$(f(x_0))$		$(f[x_{-1}, x_0, x_1])$		$(\underline{\overline{f[x_{-2}, x_{-1}, x_0, x_1, x_2]}})$
		$(f[x_0, x_1])$		$(\underline{f[x_{-1}, x_0, x_1, x_2]})$	
x_1	$f(x_1)$		$\underline{f[x_0, x_1, x_2]}$		
		$\underline{f[x_1, x_2]}$			
x_2	$\underline{f(x_2)}$				

Example

x_i	1	3	4	6	7
y_i	0	1	3	-2	-2

$(\cdot)$: Centered Differences, $\overline{\cdot}$: Forward Differences, $\underline{\cdot}$: Backward differences

x_i	$f(x_i)$	first	second	third	fourth
$x_{-2} = 1$	$\overline{0}$				
		$\overline{\frac{1}{2}}$			
$x_{-1} = 3$	1		$\overline{\frac{1}{2}}$		
		(2)		$\overline{(-\frac{2}{5})}$	
$x_0 = 4$	(3)		$(-\frac{3}{2})$		$\overline{\underline{(\frac{59}{360})}}$
		$(-\frac{5}{2})$		$\underline{(\frac{7}{12})}$	
$x_1 = 6$	-2		$\underline{\frac{5}{6}}$		
		$\underline{0}$			
$x_2 = 7$	$\underline{-2}$				

Lecture 6

Hermite Interpolation

Osculating polynomials

Taylor polynomials fits the function value of f and its higher order derivatives at a one location x_0.
Lagrange polynomials fits the function value of f at $n+1$ locations.
Osculating polynomials generalize both the Taylor polynomials and the Lagrange polynomials

Definition

The osculating polynomial approximating f is the polynomial $P(x)$ of least degree such that

$$\frac{d^kP(x_i)}{dx^k} = \frac{d^kf(x_i)}{dx^k}, \text{ for } i = 0, 1, \ldots n, \text{ and } k = 0, 1, \ldots, m_i.$$

We can prove that osculating polynomial P is a polynomial of degree at most $\sum_{i=0}^{n} m_i + n$.

Hermite Polynomials

Hermite Polynomial is a special example of osculating polynomials with $m_i = 1$ for all i.
Thus, a Hermite Polynomial, H_{2n+1}, approximating f fits the function value of f and the first derivative of f at $x_0, x_1, \ldots, x_n$. That is, H_{2n+1} satisfies

$$\begin{aligned} H_{2n+1}(x_i) &= f(x_i) \;\text{ for } i = 0, 1, \ldots n, \\ H'_{2n+1}(x_i) &= f'(x_i) \;\text{ for } i = 0, 1, \ldots n. \end{aligned}$$

Theorem

Theorem

If $f \in C^1([a,b])$, and $x_0, x_1, \ldots, x_n$ are distinct, the unique polynomial of least degree agreeing with f and f' at $x_0, \ldots, x_n$ is the Hermite polynomial of degree at most $2n+1$ given by

$$H_{2n+1} = \sum_{j=0}^{n} f(x_j) H_{n,j} + \sum_{j=0}^{n} f'(x_j) \hat{H}_{n,j},$$

where $L_{n,j}$ denoting the j-th Lagrange coefficient polynomial of degree n and

$$H_{n,j}(x) = [1 - 2(x - x_j) L'_{n,j}(x_j)] L^2_{n,j}(x),$$
$$\hat{H}_{n,j}(x) = (x - x_j) L^2_{n,j}(x).$$

Moreover, if $f \in C^{2n+2}([a,b])$, we have

$$f(x) = H_{2n+1}(x) + \frac{\prod_{i=0}^{n}(x - x_i)^2}{(2n+2)!} f^{(2n+2)}(\xi(x)), \text{ where } \xi(x) \in (a,b).$$

Proof: $H_{2n+1}(x_i) = f(x_i)$

Recall that

$$L_{n,j}(x_i) = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{if } i \neq j \end{cases}.$$

Since

$$\begin{aligned} H_{n,j}(x) &= [1 - 2(x - x_j)L'_{n,j}(x_j)]L^2_{n,j}(x), \\ \hat{H}_{n,j}(x) &= (x - x_j)L^2_{n,j}(x), \end{aligned}$$

we have

$$H_{n,j}(x_i) = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{if } i \neq j \end{cases} \text{ and } \hat{H}_{n,j}(x_i) = \begin{cases} 0 & \text{if } i = j \\ 0 & \text{if } i \neq j \end{cases}.$$

Therefore,
$H_{2n+1}(x_i) = \sum_{j=0}^{n} f(x_j)H_{n,j}(x_i) + \sum_{j=0}^{n} f'(x_j)\hat{H}_{n,j}(x_i) = f(x_i) \cdot 1 + \sum_{j \neq i} f(x_j) \cdot 0 + \sum_i f'(x_j) \cdot 0.$

Proof: $H'_{2n+1}(x_i) = f'(x_i)$

We have

$$H'_{n,j}(x) = \frac{d}{dx}\Big([1-2(x-x_j)L'_{n,j}(x_j)]L^2_{n,j}(x)\Big)$$
$$= 2[1-2(x-x_j)L'_{n,j}(x_j)]L'_{n,j}(x)L_{n,j}(x) - 2L'_{n,j}(x_j)L^2_{n,j}(x),$$
$$\hat{H}'_{n,j}(x) = \frac{d}{dx}[(x-x_j)L^2_{n,j}(x)] = L^2_{n,j}(x) + 2(x-x_j){L'}_{n,j}(x)L_{n,j}(x)$$

Thus, we have

$$H'_{n,j}(x_i) = \begin{cases} 2[1-2\cdot 0\cdot L'_{n,j}(x_j)]L'_{n,i}(x_i)\cdot 1 - 2L'_{n,j}(x_j)\cdot 1 & \text{if } i=j \\ 0 & \text{if } i\neq j \end{cases} = \begin{cases} 0 & \text{if } i=j \\ 0 & \text{if } i\neq j \end{cases}$$

$$\hat{H}'_{n,j}(x_i) = \begin{cases} 1 + 2\cdot 0\cdot {L'}_{n,j}(x_i)\cdot 1 & \text{if } i=j \\ 0 & \text{if } i\neq j \end{cases} = \begin{cases} 1 & \text{if } i=j \\ 0 & \text{if } i\neq j \end{cases}$$

Therefore,
$H'_{2n+1}(x_i) = \sum_{j=0}^{n} f(x_j)H'_{n,j}(x_i) + \sum_{j=0}^{n} f'(x_j)\hat{H}'_{n,j}(x_i) = \sum_i f(x_j)\cdot 0 + f'(x_i)\cdot 1 + \sum_{j\neq i} f'(x_j)\cdot 0.$

Proof: error estimate

Let $g = f(t) - H_{2n+1}(t) - [f(x) - H_{2n+1}(x)] \dfrac{\prod_{i=0}^{n}(t - x_i)^2}{\prod_{i=0}^{n}(x - x_i)^2}$.
We can prove that $g'(x)$ has $2n+2$ roots. Using generalized Rolle's Theorem and the same trick for the proof of error estimate for Lagrange interpolation, we will obtain the result that

$$f(x) = H_{2n+1}(x) + \frac{\prod_{i=0}^{n}(x - x_i)^2}{(2n+2)!} f^{(2n+2)}(\xi(x)).$$

Proof: uniqueness

Assume $\tilde{H}_{2n+1}$ is another polynomial of degree at most $2n+1$ satisfying

$$\begin{aligned}\tilde{H}_{2n+1}(x_i) &= f(x_i) = H_{2n+1}(x_i) \ \text{ for } i = 0, 1, \ldots n,\\ \tilde{H}'_{2n+1}(x_i) &= f'(x_i) = H'_{2n+1}(x_i) \ \text{ for } i = 0, 1, \ldots n.\end{aligned}$$

Instead of considering H_{2n+1} is an interpolation of f, we can consider H_{2n+1} is an interpolation of $\tilde{H}_{2n+1}$. By using the previous error estimate, we have

$$\tilde{H}_{2n+1}(x) - H_{2n+1}(x) = \frac{\prod_{i=0}^{n}(x - x_i)^2}{(2n+2)!}\tilde{H}_{2n+1}^{(2n+2)}(\xi(x)).$$

Since $\tilde{H}_{2n+1}$ is another polynomial of degree at most $2n+1$, we have $\tilde{H}_{2n+1}^{(2n+2)} \equiv 0$ and $\tilde{H}_{2n+1} = H_{2n+1}$.

Example

x_i	0	1
$f(x_i)$	1	2
$f'(x_i)$	0	1

$$L_{1,0} = \frac{x - x_1}{x_0 - x_1} = 1 - x, \;\; L_{1,1} = \frac{x - x_0}{x_1 - x_0} = x$$

$$\begin{aligned} H_{1,0}(x) &= [1 - 2(x - x_0)L'_{1,0}(x_0)]L^2_{1,0} = [1 - 2(x - 0)(-1)](1 - x)^2 \\ &= (1 + 2x)(1 - x)^2 = 2x^3 - 3x^2 + 1 \end{aligned}$$

$$\begin{aligned} H_{1,1}(x) &= [1 - 2(x - x_1)L'_{1,1}(x_1)]L^2_{1,0} = [1 - 2(x - 1)(1)]x^2 \\ &= (3 - 2x)x^2 = -2x^3 + 3x^2 \end{aligned}$$

$$\hat{H}_{1,0}(x) = (x - x_0)L^2_{1,0}(x) = x(1 - x)^2 = x^3 - 2x^2 + x$$
$$\hat{H}_{1,1}(x) = (x - x_1)L^2_{1,1}(x) = (x - 1)x^2 = x^3 - x^2$$

Example

x_i	0	1
$f(x_i)$	1	2
$f'(x_i)$	0	1

$$\begin{aligned} H_3(x) &= 1 \cdot H_{1,0}(x) + 2 \cdot H_{1,1}(x) + 0 \cdot \hat{H}_{1,0}(x) + 1 \cdot \hat{H}_{1,1}(x) \\ &= (2x^3 - 3x^2 + 1) + 2(-2x^3 + 3x^2) + 1 \cdot (x^3 - x^2) \\ &= -x^3 + 2x^2 + 1 \end{aligned}$$

$$H_3'(x) = -3x^2 + 4x$$

Hermite Polynomials Using Divided Differences

There is an alternative method for generating Hermite approximations which use the concept of divided differences.
First, we define $z_0, z_1, \ldots, z_{2n+1}$ by

$$z_{2i} = z_{2i+1} = x_i, \text{ for } i = 0, \ldots, n.$$

Since $f[x, y] = f'(\xi)$ for some ξ between x and y, it is reasonable to define

$$f[z_{2i}, z_{2i+1}] = f'(x_i).$$

The Hermite approximation H_{2n+1} is given by

$$H_{2n+1}(x) = f[z_0] + \sum_{k=1}^{2n+1} f[z_0, z_1, \cdots, z_k] \prod_{i=0}^{k-1} (x - z_i).$$

Example

x_i	0	1
$f(x_i)$	1	2
$f'(x_i)$	0	1

z_i	$f(x_i)$	first	second	third
$z_0 = x_0$	$f(x_0)$			
		$f[z_0, z_1] = f'(x_0)$		
$z_1 = x_0$	$f(x_0)$		$f[z_0, z_1, z_2]$	
		$f[z_1, z_2] = f[x_0, x_1]$		$f[z_0, z_1, z_2, z_3]$
$z_2 = x_1$	$f(x_1)$		$f[z_1, z_2, z_3]$	
		$f[z_2, z_3] = f'(x_1)$		
$z_3 = x_1$	$f(x_1)$			

Example

x_i	0	1
$f(x_i)$	1	2
$f'(x_i)$	0	1

z_i	$f(x_i)$	first	second	third
0	1			
		0		
0	1		$\frac{1-0}{1-0}=1$	
		$\frac{2-1}{1-0}=1$		$\frac{0-1}{1-0}=-1$
1	2		$\frac{1-1}{1-0}=0$	
		1		
1	2			

We have

$H(x) = 1 + 0\cdot(x-0) + 1\cdot(x-0)^2 + (-1)\cdot(x-0)^2(x-1) = 1 + x^2 - x^2(x-1) = 1 + 2x^2 - x^3$

Lecture 7

Hermite Interpolation

Review:Hermite Polynomials

Hermite Interpolating Polynomial H_{2n+1} is a polynomial of degree at most $2n+1$ fits the function value of f and the first derivative of f at $x_0, x_1, \ldots, x_n$. That is, H_{2n+1} satisfies

$$H_{2n+1}(x_i) = f(x_i) \text{ for } i = 0, 1, \ldots n,$$
$$H'_{2n+1}(x_i) = f'(x_i) \text{ for } i = 0, 1, \ldots n.$$

Theorem

Theorem

If $f \in C^1([a,b])$, and $x_0, x_1, \ldots, x_n$ are distinct, the unique polynomial of least degree agreeing with f and f' at $x_0, \ldots, x_n$ is the Hermite polynomial of degree at most $2n+1$ given by

$$H_{2n+1} = \sum_{j=0}^{n} f(x_j) H_{n,j} + \sum_{j=0}^{n} f'(x_j) \hat{H}_{n,j},$$

where $L_{n,j}$ denoting the j-th Lagrange coefficient polynomial of degree n and

$$\begin{aligned} H_{n,j}(x) &= [1 - 2(x - x_j) L'_{n,j}(x_j)] L^2_{n,j}(x), \\ \hat{H}_{n,j}(x) &= (x - x_j) L^2_{n,j}(x). \end{aligned}$$

Moreover, if $f \in C^{2n+2}([a,b])$, we have

$$f(x) = H_{2n+1}(x) + \frac{\prod_{i=0}^{n} (x - x_i)^2}{(2n+2)!} f^{(2n+2)}(\xi(x)), \text{ where } \xi(x) \in (a,b).$$

Proof: $H_{2n+1}(x_i) = f(x_i)$

Recall that

$$L_{n,j}(x_i) = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{if } i \neq j \end{cases}.$$

Since

$$\begin{aligned} H_{n,j}(x) &= [1 - 2(x - x_j)L'_{n,j}(x_j)]L^2_{n,j}(x), \\ \hat{H}_{n,j}(x) &= (x - x_j)L^2_{n,j}(x), \end{aligned}$$

we have

$$H_{n,j}(x_i) = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{if } i \neq j \end{cases} \text{ and } \hat{H}_{n,j}(x_i) = \begin{cases} 0 & \text{if } i = j \\ 0 & \text{if } i \neq j \end{cases}.$$

Therefore,
$H_{2n+1}(x_i) = \sum_{j=0}^{n} f(x_j)H_{n,j}(x_i) + \sum_{j=0}^{n} f'(x_j)\hat{H}_{n,j}(x_i) = f(x_i) \cdot 1 + \sum_{j \neq i} f(x_j) \cdot 0 + \sum_i f'(x_j) \cdot 0.$

Proof: $H'_{2n+1}(x_i) = f'(x_i)$

We have

$$H'_{n,j}(x) = \frac{d}{dx}\Big([1-2(x-x_j)L'_{n,j}(x_j)]L^2_{n,j}(x)\Big)$$
$$= 2[1-2(x-x_j)L'_{n,j}(x_j)]L'_{n,j}(x)L_{n,j}(x) - 2L'_{n,j}(x_j)L^2_{n,j}(x),$$
$$\hat{H}'_{n,j}(x) = \frac{d}{dx}[(x-x_j)L^2_{n,j}(x)] = L^2_{n,j}(x) + 2(x-x_j)L'{}_{n,j}(x)L_{n,j}(x)$$

Thus, we have

$$H'_{n,j}(x_i) = \begin{cases} 2[1-2\cdot 0\cdot L'_{n,j}(x_j)]L'_{n,i}(x_i)\cdot 1 - 2L'_{n,j}(x_j)\cdot 1 & \text{if } i=j \\ 0 & \text{if } i\neq j \end{cases} = \begin{cases} 0 & \text{if } i=j \\ 0 & \text{if } i\neq j \end{cases}$$

$$\hat{H}'_{n,j}(x_i) = \begin{cases} 1 + 2\cdot 0\cdot L'{}_{n,j}(x_i)\cdot 1 & \text{if } i=j \\ 0 & \text{if } i\neq j \end{cases} = \begin{cases} 1 & \text{if } i=j \\ 0 & \text{if } i\neq j \end{cases}$$

Therefore,
$H'_{2n+1}(x_i) = \sum_{j=0}^{n} f(x_j)H'_{n,j}(x_i) + \sum_{j=0}^{n} f'(x_j)\hat{H}'_{n,j}(x_i) = \sum_i f(x_j)\cdot 0 + f'(x_i)\cdot 1 + \sum_{j\neq i} f'(x_j)\cdot 0.$

Proof: error estimate

We can check that, for $x = x_j$ for some $j = 0, \ldots, n$, we have

$$f(x_j) - H_{2n+1}(x_j) = 0 = \frac{\prod_{i=0}^{n}(x_j - x_i)^2}{(2n+2)!} f^{(2n+2)}(\xi(x)) \text{ for any } \xi(x) \in (a, b).$$

Given a fixed $x \neq x_j$, we then define a function g by

$$g(t) = f(t) - H_{2n+1}(t) - [f(x) - H_{2n+1}(x)] \frac{\prod_{i=0}^{n}(t - x_i)^2}{\prod_{i=0}^{n}(x - x_i)^2}. \text{ (} g \text{ is a function of } t\text{)}$$

We can check that x_j is a root of g for $i = 0, 1, \ldots, n$ since $f(x_j) = H_{2n+2}(x_j)$ and $\prod_{i=0}^{n}(x_j - x_i)^2 = 0$.

Moreover, we can check that $g(x) = f(x) - H_{2n+1}(x) - [f(x) - H_{2n+1}(x)] \cdot 1 = 0$.

Therefore, g has $n + 2$ roots in $[a, b]$.

Proof: error estimate

Assuming $x \in (x_k, x_{k+1})$, by Rolle's Theorem, for each $j \neq k$, there is a root of g', $\xi_j \in (x_j, x_{j+1})$, such that $g'(\xi_j) = 0$.
Moreover, by Rolle's Theorem, there are two roots of g', $\xi_k \in (x_k, x)$ and $\xi_{n+1} \in (x, x_{k+1})$ such that $g'(\xi_k) = g'(\xi_{n+1})$.
Since $\xi_j \in (x_j, x_{j+1})$ for all j and $\xi_{n+1} \in (x_k, x_{k+1})$, we have $\xi_j \neq x_i$ for any i, j.
Consider $\tilde{g}(t) = \prod_{i=0}^{n}(t - x_i)^2$, we have $\tilde{g}(t) = (t - x_j)^2 \prod_{i \neq j}(t - x_i)^2$ for any j.
We then have $\tilde{g}'(t) = 2(t - x_j) \prod_{i \neq j}(t - x_i)^2 + (t - x_j)^2 \dfrac{d\left(\prod_{i \neq j}(t - x_i)^2\right)}{dt}$ and thus, $g'(x_j) = 0$ for any j.
Therefore, $g'(x)$ has $2n + 2$ roots ($x_0, x_1, \dots x_n$ and $\xi_0, \xi_1, \dots, \xi_n$).

Proof: error estimate

By generalized Rolle's Theorem, we have $g^{(2n+2)}$ has a root $\xi(x)$.

$$0 = g^{(2n+2)}(\xi(x)) = f^{(2n+2)}(\xi(x)) - H_{2n+1}^{(2n+2)}(\xi(x)) - \frac{[f(x) - H_{2n+1}(x)]}{\prod_{i=0}^{n}(x - x_i)^2}\tilde{g}^{(2n+2)}(\xi(x))$$

where $\tilde{g}(t) = \prod_{i=0}^{n}(t - x_i)^2$.
We can check that the leading coefficient of $\tilde{g}$ is 1, namely

$$\tilde{g}(t) = t^{2n+2} + a_{2n+1}t^{2n+1} + \cdots + a_1 t + a_0$$

where a_j are constant depending on $\{x_i\}_{i=0}^{n}$.
Since $\dfrac{d^{2n+2}}{dt^{2n+2}}(t^j) = 0$ for $j \leq 2n+1$, we have
$\tilde{g}^{(2n+2)}(t) = \dfrac{d^{2n+2}}{dt^{2n+2}}(t^{2n+2}) = (2n+2)!$. Since $H_{2n+1}^{(2n+2)} \equiv 0$, we have

$$f^{(2n+2)}(\xi(x)) = \frac{[f(x) - H_{2n+1}(x)](2n+2)!}{\prod_{i=0}^{n}(x - x_i)^2} \implies f(x) - H_{2n+1}(x) = \frac{f^{(2n+2)}(\xi(x))\prod_{i=0}^{n}(x - x_i)^2}{(2n+2)!}$$

Proof: uniqueness

Assume $\tilde{H}_{2n+1}$ is another polynomial of degree at most $2n+1$ satisfying

$$\tilde{H}_{2n+1}(x_i) = f(x_i) = H_{2n+1}(x_i) \text{ for } i = 0, 1, \ldots n,$$
$$\tilde{H}'_{2n+1}(x_i) = f'(x_i) = H'_{2n+1}(x_i) \text{ for } i = 0, 1, \ldots n.$$

Instead of considering H_{2n+1} is an interpolation of f, we can consider H_{2n+1} is an interpolation of $\tilde{H}_{2n+1}$. By using the previous error estimate, we have

$$\tilde{H}_{2n+1}(x) - H_{2n+1}(x) = \frac{\prod_{i=0}^{n}(x - x_i)^2}{(2n+2)!}\tilde{H}_{2n+1}^{(2n+2)}(\xi(x)).$$

Since $\tilde{H}_{2n+1}$ is another polynomial of degree at most $2n+1$, we have $\tilde{H}_{2n+1}^{(2n+2)} \equiv 0$ and $\tilde{H}_{2n+1} = H_{2n+1}$.

Example

x_i	0	1
$f(x_i)$	1	2
$f'(x_i)$	0	1

$$L_{1,0} = \frac{x - x_1}{x_0 - x_1} = 1 - x, \;\; L_{1,1} = \frac{x - x_0}{x_1 - x_0} = x$$

$$\begin{aligned} H_{1,0}(x) &= [1 - 2(x - x_0)L'_{1,0}(x_0)]L^2_{1,0} = [1 - 2(x - 0)(-1)](1 - x)^2 \\ &= (1 + 2x)(1 - x)^2 = 2x^3 - 3x^2 + 1 \end{aligned}$$

$$\begin{aligned} H_{1,1}(x) &= [1 - 2(x - x_1)L'_{1,1}(x_1)]L^2_{1,0} = [1 - 2(x - 1)(1)]x^2 \\ &= (3 - 2x)x^2 = -2x^3 + 3x^2 \end{aligned}$$

$$\hat{H}_{1,0}(x) = (x - x_0)L^2_{1,0}(x) = x(1 - x)^2 = x^3 - 2x^2 + x$$

$$\hat{H}_{1,1}(x) = (x - x_1)L^2_{1,1}(x) = (x - 1)x^2 = x^3 - x^2$$

Example

x_i	0	1
$f(x_i)$	1	2
$f'(x_i)$	0	1

$$\begin{aligned} H_3(x) &= 1 \cdot H_{1,0}(x) + 2 \cdot H_{1,1}(x) + 0 \cdot \hat{H}_{1,0}(x) + 1 \cdot \hat{H}_{1,1}(x) \\ &= (2x^3 - 3x^2 + 1) + 2(-2x^3 + 3x^2) + 1 \cdot (x^3 - x^2) \\ &= -x^3 + 2x^2 + 1 \end{aligned}$$

$$H_3'(x) = -3x^2 + 4x$$

Hermite Polynomials Using Divided Differences

There is an alternative method for generating Hermite approximations which use the concept of divided differences.
First, we define $z_0, z_1, \ldots, z_{2n+1}$ by

$$z_{2i} = z_{2i+1} = x_i, \text{ for } i = 0, \ldots, n.$$

Since $f[x, y] = f'(\xi)$ for some ξ between x and y, it is reasonable to define

$$f[z_{2i}, z_{2i+1}] = f'(x_i).$$

The Hermite approximation H_{2n+1} is given by

$$H_{2n+1}(x) = f[z_0] + \sum_{k=1}^{2n+1} f[z_0, z_1, \cdots, z_k] \prod_{i=0}^{k-1} (x - z_i).$$

Example

x_i	0	1
$f(x_i)$	1	2
$f'(x_i)$	0	1

z_i	$f(x_i)$	first	second	third
$z_0 = x_0$	$f(x_0)$			
		$f[z_0, z_1] = f'(x_0)$		
$z_1 = x_0$	$f(x_0)$		$f[z_0, z_1, z_2]$	
		$f[z_1, z_2] = f[x_0, x_1]$		$f[z_0, z_1, z_2, z_3]$
$z_2 = x_1$	$f(x_1)$		$f[z_1, z_2, z_3]$	
		$f[z_2, z_3] = f'(x_1)$		
$z_3 = x_1$	$f(x_1)$			

Example

x_i	0	1
$f(x_i)$	1	2
$f'(x_i)$	0	1

z_i	$f(x_i)$	first	second	third
0	1			
		0		
0	1		$\frac{1-0}{1-0} = 1$	
		$\frac{2-1}{1-0} = 1$		$\frac{0-1}{1-0} = -1$
1	2		$\frac{1-1}{1-0} = 0$	
		1		
1	2			

We have
$H(x) = 1 + 0 \cdot (x-0) + 1 \cdot (x-0)^2 + (-1) \cdot (x-0)^2(x-1) = 1 + x^2 - x^2(x-1) = 1 + 2x^2 - x^3$

Another example

x_i	-1	0	1
$f(x_i) = x^3$	-1	0	1
$f'(x_i) = 3x^2$	3	0	3

z_i	$f(x_i)$	first	second	third	fourth
$z_0 = x_0$	$f(x_0)$				
		$f[z_0, z_1] = f'(x_0)$			
$z_1 = x_0$	$f(x_0)$		$f[z_0, z_1, z_2]$		
		$f[z_1, z_2] = f[x_0, x_1]$		$f[z_0, z_1, z_2, z_3]$	
$z_2 = x_1$	$f(x_1)$		$f[z_1, z_2, z_3]$		$f[z_0, z_1, z_2, z_3, z_4]$
		$f[z_2, z_3] = f'(x_1)$		$f[z_1, z_2, z_3, z_4]$	
$z_3 = x_1$	$f(x_1)$		$f[z_2, z_3, z_4]$		$f[z_1, z_2, z_3, z_4, z_5]$
		$f[z_3, z_4] = f[x_1, x_2]$		$f[z_2, z_3, z_4, z_5]$	
$z_4 = x_2$	$f(x_2)$		$f[z_3, z_4, z_5]$		
		$f[z_4, z_5] = f'(x_2)$			
$z_5 = x_2$	$f(x_2)$				

Cont.

z_i	$f(x_i)$	first	second	third	fourth
-1	-1				
		3			
-1	-1		$f[z_0, z_1, z_2]$		
		$f[z_1, z_2] = f[x_0, x_1]$		$f[z_0, z_1, z_2, z_3]$	
0	0		$f[z_1, z_2, z_3]$		$f[z_0, z_1, z_2, z_3, z_4]$
		0		$f[z_1, z_2, z_3, z_4]$	
0	0		$f[z_2, z_3, z_4]$		$f[z_1, z_2, z_3, z_4, z_5]$
		$f[z_3, z_4] = f[x_1, x_2]$		$f[z_2, z_3, z_4, z_5]$	
1	1		$f[z_3, z_4, z_5]$		
		3			
1	1				

Cont.

z_i	$f(x_i)$	first	second	third	fourth
-1	-1				
		3			
-1	-1		$f[z_0, z_1, z_2]$		
		$\dfrac{0-(-1)}{0-(-1)} = 1$		$f[z_0, z_1, z_2, z_3]$	
0	0		$f[z_1, z_2, z_3]$		$f[z_0, z_1, z_2, z_3, z_4]$
		0		$f[z_1, z_2, z_3, z_4]$	
0	0		$f[z_2, z_3, z_4]$		$f[z_1, z_2, z_3, z_4, z_5]$
		$\dfrac{1-0}{1-0} = 1$		$f[z_2, z_3, z_4, z_5]$	
1	1		$f[z_3, z_4, z_5]$		
		3			
1	1				

Cont.

z_i	$f(x_i)$	first	second	third	fourth
-1	-1				
		3			
-1	-1		$\frac{1-3}{0-(-1)} = -2$		
		1		$f[z_0, z_1, z_2, z_3]$	
0	0		$\frac{0-1}{0-(-1)} = -1$		$f[z_0, z_1, z_2, z_3, z_4]$
		0		$f[z_1, z_2, z_3, z_4]$	
0	0		$\frac{1-0}{1-0} = 1$		$f[z_1, z_2, z_3, z_4, z_5]$
		1		$f[z_2, z_3, z_4, z_5]$	
1	1		$\frac{3-1}{1-0} = 2$		
		3			
1	1				

Cont.

z_i	$f(x_i)$	first	second	third	fourth	fifth
-1	-1					
		3				
-1	-1		$\dfrac{1-3}{0-(-1)} = -2$			
		1		$\dfrac{-1-(-2)}{0-(-1)} = 1$		
0	0		$\dfrac{0-1}{0-(-1)} = -1$		0	
		0		$\dfrac{1-(-1)}{1-(-1)} = 1$		0
0	0		$\dfrac{1-0}{1-0} = 1$		0	
		1		$\dfrac{2-1}{1-0} = 1$		
1	1		$\dfrac{3-1}{1-0} = 2$			
		3				
1	1					

We have $H(x) =$
$-1+3(x+1)+(-2)\cdot(x+1)^2+1\cdot(x+1)^2(x-0)+0\cdot(x+1)^2(x-0)^2+0\cdot(x+1)^2(x-0)^2(x-1) = x^3$

Lecture 8

Cubic Splines

High degree polynomial

Using high degree Lagrange interpolation or Hermite interpolation is generally a unstable process. The interpolation polynomial can be very sensitive to the measurement error (error of the data).
That is, even if two data sets $\{(x_i, f(x_i))\}_{i=0}^{n}$ and $\{x_i, \tilde{f}(x_i)\}_{i=0}^{n}$ are similar, the the interpolations for those two data sets can have a huge difference.
Moreover, if the function f is not smooth, using higher degree interpolation may not provide a better result. (In some case, it can be diverging.)

Piecewise approximate

Form the error analysis, we know that the error bound is basically a product of two terms.
One is the derivative of f and the other is the production of distance between x and x_i.
For example, for Lagrange interpolation, the error term is the product of

$$\frac{f^{(n)}(\xi(x))}{n!} \text{ and } \prod_{i=0}^{n}(x - x_i).$$

If the function f is not a smooth function, we can not expect the term $\dfrac{f^{(n)}(\xi(x))}{n!}$ is small (especially when n is large).
To reduce the error, one can try to use a piecewise approximation which reduce the term $\prod_{i=0}^{n}(x - x_i)$.

Example: piecewise linear interpolation

Assume we use the piecewise linear interpolating polynomial P to approximate the function f, we can obtain the error approximate from the Theorem that if $f \in C^2$, we have

$$|f(x) - P(x)| \leq \frac{f^{(2)}(\xi(x))}{2}(x - x_i)(x - x_{i+1}), \forall x \in [x_i, x_{i+1}].$$

(assuming $a = x_0 < x_1 < \cdots < x_n = b$)
Therefore, if $|x_i - x_{i+1}| \leq h$ for all i, since $|x - x_i|$ and $|x - x_{i+1}|$ are both smaller than $|x_{i+1} - x_i|$, we have

$$|f(x) - P(x)| \leq \frac{f^{(2)}(\xi(x))}{2}h^2 \; \forall x \in [a, b]$$

(In fact, we can prove that the optimal bound is
$|f(x) - P(x)| \leq \dfrac{f^{(2)}(\xi(x))}{8}h^2 \; \forall x \in [a, b]$. You can prove it as an exercise)

Piecewise linear interpolation and piecewise Hermite interpolation

Therefore, instead of using higher degree polynomial, it is reasonable to use a piecewise interpolation. (Another advantage of piesewise interpolation is the computational speed of the approximation.)
But using piecewise linear interpolation will provide a non-differentiable approximation.
($P'(x)$ is not continuous at points $\{x_i\}_{i=1}^{n-1}$.)
To obtain a continuous differentiable interpolation, one possible way is to use piecewise Hermite interpolation.

Piecewise Hermite interpolation

We consider H is a piecewise Hermite interpolation such that

$$H(x) = H_{3,i}(x) \ \text{ for } x \in [x_i, x_{i+1}]$$

where $H_{3,i}$ are the Hermite polynomials of degree at most 3 with

$$\begin{aligned} H_i(x_j) &= f(x_j) \\ H_i'(x_j) &= f'(x_j) \end{aligned}$$

for $j = i, i+1$. It is not hard to check that H is continuous differentiable since $H'_{3,i}(x_{i+1}) = H'_{3,i+1}(x_{i+1}) = f'(x_{i+1})$ for all $i = 0, \ldots, n-1$.

However, piecewise Hermite cubic polynomial is not the piecewise interpolating polynomial with least degree which is continuous differentiable.
(One can find a piecewise quadratic polynomial which is continuous differentiable and fit the function value at x_i)
Another problem is that piecewise Hermite interpolation requires the information of the derivative of f which is generally not easy to obtain.
Therefore, we are going to introduce an interpolation, which are widely applied, called cubic spline interpolation.

Instead of fitting the first derivative of f, cubic spline interpolating polynomial S requires the derivative of interpolating polynomial is continuous. Since there are four coefficients for a piecewise cubic polynomial ($S_i(x) = a_i + b_i x + c_i x^2 + d_i x^3$ where S_i is a cubic polynomial defined in $[x_i, x_{i+1}]$), we have total $4n$ coefficients.
We can count that if we just require S is continuous differentiable and fit the function value at x_i, we only have $3n-1$ equations which are

$$S_i(x_i) = f(x_i),\ S_i(x_{i+1}) = f(x_{i+1}) \text{ (2n equations)}$$
$$S_i'(x_{i+1}) = S_{i+1}'(x_{i+1}) \text{ for } i = 0, \ldots, n-2 \text{ (n-1 equations)}$$

Therefore, we can have some extra conditions for the cubic spline interpolation which is the second derivative of S is also continuous.

Definition

Given a function f defined on $[a, b]$ and a set of nodes

$$a = x_0 < x_1 < \cdots < x_n = b,$$

a cubic spline interpolant S for f is a function that satisfies the following conditions:

1. S is a cubic polynomial, denoted $S_i(x)$, on $[x_i, x_{i+1}]$ for each $i = 0, 1, \ldots n - 1$.
2. $\begin{array}{r} S_i(x_i) = f(x_i), \\ S_i(x_{i+1}) = f(x_{i+1}), \end{array}$ for $i = 0, 1, \ldots, n - 1$ (2n equations)
3. $S_i(x_{i+1}) = S_{i+1}(x_{i+1})$ for $i = 0, \ldots, n - 2$ (by (2))
4. $S_i'(x_{i+1}) = S_{i+1}'(x_{i+1})$ for $i = 0, \ldots, n - 2$ (n-1 equations)
5. $S_i''(x_{i+1}) = S_{i+1}''(x_{i+1})$ for $i = 0, \ldots, n - 2$ (n-1 equations)
6. One of the following sets of boundary conditions is satisfied:
   1. $S''(x_0) = S''(x_n) = 0$ (natural (or free) boundary)
   2. $S'(x_0) = f'(x_0)$ and $S'(x_n) = f'(x_n)$ (clamped boundary).

Example

x_i	0	1	2
$f(x_i)$	1	2	2

The spline consists of two cubic polynomials, S_0 and S_1 where S_0 is defined on $[0, 1]$ and S_1 is defined on $[1, 2]$. We consider

$$S_0(x) = a_0 + b_0 x + c_0 x^2 + d_0 x^3$$
$$S_1(x) = a_1 + b_1(x-1) + c_1(x-1)^2 + d_1(x-1)^3$$

$S_i(x_i) = f(x_i),\ S_i(x_{i+1}) = f(x_{i+1})$

x_i	0	1	2
$f(x_i)$	1	2	2

Since S fit the data, we have

$$\begin{aligned} a_0 &= f(0) = 1 \\ a_0 + b_0 + c_0 + d_0 &= f(1) = 2 \end{aligned}$$

and

$$\begin{aligned} a_1 &= f(1) = 2 \\ a_1 + b_1 + c_1 + d_1 &= f(2) = 2. \end{aligned}$$

$S_i'(x_{i+1}) = S_{i+1}'(x_{i+1})$

x_i	0	1	2
$f(x_i)$	1	2	2

$$S_0'(x) = b_0 + 2c_0 x + 3d_0 x^2$$
$$S_1'(x) = b_1 + 2c_1(x-1) + 3d_1(x-1)^2$$

and

$$S_0'(1) = b_0 + 2c_0 + 3d_0 = b_1 = S_1'(1)$$

$S_i''(x_{i+1}) = S_{i+1}''(x_{i+1}),\ S''(x_0) = S''(x_n) = 0$

x_i	0	1	2
$f(x_i)$	1	2	2

$$
\begin{aligned}
S_0''(x) &= 2c_0 + 6d_0 x \\
S_1''(x) &= 2c_1 + 6d_1(x-1)
\end{aligned}
$$

We have

$$
\begin{aligned}
S_0''(1) &= 2c_0 + 6d_0 = 2c_1 = S_1''(1) \\
S_0''(0) &= 2c_0 = 0 \\
S_1''(2) &= 2c_1 + 6d_1 = 0
\end{aligned}
$$

Cont.

x_i	0	1	2
$f(x_i)$	1	2	2

By the solving all of the equations, we have

$$S(x) = \begin{cases} 1 + \frac{5}{4}x - \frac{1}{4}x^3 & \text{if } x \in [0,1] \\ 2 + \frac{1}{2}(x-1) - \frac{3}{4}(x-1)^2 + \frac{1}{4}(x-1)^3 & \text{if } x \in (1,2] \end{cases}$$

In general case, we can denote

$$S_i(x) = a_i + b_i(x - x_i) + c_i(x - x_i)^2 + d_i(x - x_i)^3$$

and $x_{i+1} - x_i = h_i$ for $i = 0, 1, \ldots, n-1$. Use condition (3), we have

$$S_{i+1}(x_{i+1}) = a_{i+1} = a_i + b_i h_i + c_i h_i^2 + d_i h_i^3 = S_i(x_{i+1}). \tag{1.1}$$

Similarly, since $S_i'(x_{i+1}) = S_{i+1}'(x_{i+1})$ and $S_i''(x_{i+1}) = S_{i+1}''(x_{i+1})$, we have

$$\begin{aligned} S_{i+1}'(x_{i+1}) &= b_{i+1} = b_i + 2c_i h_i + 3d_i h_i^2 = S_i'(x_{i+1}), \\ S_{i+1}''(x_{i+1}) &= 2c_{i+1} = 2c_i + 6d_i h_i = S_i''(x_{i+1}), \end{aligned} \tag{1.2}$$

Cont.

Since $2c_{i+1} = 2c_i + 6d_i h_i$, we have

$$d_i = \frac{c_{i+1} - c_i}{3h_i}, \text{ for } i = 0, 1, \ldots, n-1.$$

Substituting it to (1.1) and (1.2), we obtain

$$\begin{aligned} a_{i+1} &= a_i + b_i h_i + \frac{h_i^2}{3}(2c_i + c_{i+1}) \\ b_{i+1} &= b_i + h_i(c_i + c_{j+1}) \end{aligned} \tag{1.3}$$

Using 1.3), we have

$$b_i = \frac{(a_{i+1} - a_i)}{h_i} - \frac{h_i(2c_i + c_{i+1})}{3}$$

Cont.

Using

$$b_i = \frac{(a_{i+1} - a_i)}{h_i} - \frac{h_i(2c_i + c_{i+1})}{3}$$

$$b_{i-1} = \frac{(a_i - a_{i-1})}{h_{i-1}} - \frac{h_{i-1}(2c_{i-1} + c_i)}{3}$$

and

$$b_{i+1} = b_i + h_i(c_i + c_{j+1})$$

we have

$h_{i-1}c_{i-1} + 2(h_{i-1} + h_i)c_i + h_i c_{i+1} = \dfrac{3}{h_i}(a_{i+1} - a_i) - \dfrac{3}{h_{i-1}}(a_i - a_{i-1})$ which gives a set of equations of $\{c_i\}_{i=0}^n$ when $\{a_i\}_{i=0}^n$ are known.

Cont.

Since $S_i(x_i) = a_i = f(x_i)$, we can easily obtain the value of $\{a_i\}_{i=0}^n$.
By the natural boundary conditions, we have

$$c_0 = S_0''(x_0) = 0 \text{ and } c_n = S_n''(x_n).$$

Therefore, we can find $\{c_i\}_{i=0}^n$ by solving the linear system $Ax = b$ (A is strictly diagonally dominant) where

$$A = \begin{pmatrix} 1 & 0 & 0 & \cdots & \cdots & 0 \\ h_0 & 2(h_0 + h_1) & h_1 & \ddots & & \vdots \\ 0 & h_1 & 2(h_1 + h_2) & h_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & 0 \\ \vdots & & \ddots & h_{n-2} & 2(h_{n-2} + h_{n-1}) & h_{n-1} \\ 0 & \cdots & \cdots & 0 & 0 & 1 \end{pmatrix}$$

$$x = (c_0, c_1, \cdots, c_n)^t, \;\; b = (0, \frac{3(a_2 - a_1)}{h_1} - \frac{3(a_1 - a_0)}{h_0}, \cdots, \frac{3(a_n - a_{n-1})}{h_{n-1}} - \frac{3(a_{n-1} - a_{n-2})}{h_{n-2}}, 0)^t$$

Cont.

After solving $\{c_i\}_{i=1}^n$ and $\{a_i\}_{i=1}^n$, we obtain $\{b_i\}$ and $\{d_i\}$ by

$$\begin{aligned} d_i &= \frac{c_{i+1} - c_i}{3h_i} \\ b_i &= \frac{(a_{i+1} - a_i)}{h_i} - \frac{h_i(2c_i + c_{i+1})}{3}. \end{aligned}$$

Theorem

Theorem

If f is defined at $a = x_0 < x_1 < \cdots < x_n = b$, then f has a unique natural spline interpolant S on the nodes $x_0, x_1, \ldots, x_n$, that is, a spline interpolant that satisfies the natural boundary conditions $S''(a) = S''(b) = 0$.

Proof.

From the previous discussion, we know that we can solve the linear system to obtain $\{a_i\}_{i=0}^{n-1}$, $\{c_i\}_{i=0}^{n-1}$, $\{b_i\}_{i=0}^{n-1}$ and $\{d_i\}_{i=0}^{n-1}$. Since the system is non-singular, the cubic spline polynomial is unique. $\square$

Lecture 9

Cubic Splines

Review

Given a function f defined on $[a, b]$ and a set of nodes

$$a = x_0 < x_1 < \cdots < x_n = b,$$

a cubic spline interpolant S for f is a function that satisfies the following conditions:

1. S is a cubic polynomial, denoted $S_i(x)$, on $[x_i, x_{i+1}]$ for each $i = 0, 1, \ldots n - 1$.
2. $\begin{aligned} S_i(x_i) &= f(x_i), \\ S_i(x_{i+1}) &= f(x_{i+1}), \end{aligned}$ for $i = 0, 1, \ldots, n - 1$ (Sfits the data)
3. $S_i(x_{i+1}) = S_{i+1}(x_{i+1})$ for $i = 0, \ldots, n - 2$ (by (2))
4. $S_i'(x_{i+1}) = S_{i+1}'(x_{i+1})$ for $i = 0, \ldots, n - 2$ ($S \in C^1$)
5. $S_i''(x_{i+1}) = S_{i+1}''(x_{i+1})$ for $i = 0, \ldots, n - 2$ ($S \in C^2$)
6. One of the following sets of boundary conditions is satisfied:
    1. $S''(x_0) = S''(x_n) = 0$ (natural (or free) boundary)
    2. $S'(x_0) = f'(x_0)$ and $S'(x_n) = f'(x_n)$ (clamped boundary).

Review

Since $S_i(x_i) = a_i = f(x_i)$, we can easily obtain the value of $\{a_i\}_{i=0}^n$.
By the natural boundary conditions, we have

$$c_0 = S_0^{''}(x_0) = 0 \text{ and } c_n = S_n''(x_n).$$

Therefore, we can find $\{c_i\}_{i=0}^n$ by solving the linear system $Ax = b$ (A is strictly diagonally dominant) where

$$A = \begin{pmatrix} 1 & 0 & 0 & \cdots & \cdots & 0 \\ h_0 & 2(h_0+h_1) & h_1 & \ddots & & \vdots \\ 0 & h_1 & 2(h_1+h_2) & h_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & 0 \\ \vdots & & \ddots & h_{n-2} & 2(h_{n-2}+h_{n-1}) & h_{n-1} \\ 0 & \cdots & \cdots & 0 & 0 & 1 \end{pmatrix}$$

$$x = (c_0, c_1, \cdots, c_n)^t, \; b = \left(0, \frac{3(a_2-a_1)}{h_1} - \frac{3(a_1-a_0)}{h_0}, \cdots, \frac{3(a_n-a_{n-1})}{h_{n-1}} - \frac{3(a_{n-1}-a_{n-2})}{h_{n-2}}, 0\right)^t$$

Cont.

After solving $\{c_i\}_{i=1}^{n}$ and $\{a_i\}_{i=1}^{n}$, we obtain $\{b_i\}$ and $\{d_i\}$ by

$$d_i = \frac{c_{i+1} - c_i}{3h_i}$$

$$b_i = \frac{(a_{i+1} - a_i)}{h_i} - \frac{h_i(2c_i + c_{i+1})}{3}.$$

Algorithm (Natural Cubic Spline)

Input	$n,\ \{x_i\}_{i=0}^{n},\ \{a_i\}_{i=0}^{n} = \{f(x_i)\}_{i=0}^{n}$
Output	$\{a_i\}_{i=0}^{n-1}, \{b_i\}_{i=0}^{n-1}, \{c_i\}_{i=0}^{n-1}, \{d_i\}_{i=0}^{n-1}$
Step 1:	For $i = 0, 1, \ldots, n-1$, set $h_i = x_{i+1} - x_i$.
Step 2:	For $i = 1, 2, \ldots, n-1$, set $\alpha_i = \dfrac{3}{h_i}(a_{i+1} - a_i) - \dfrac{3}{h_{i-1}}(a_i - a_{i-1})$.
Step 3:	Set $\begin{aligned} l_0 &= 1 \\ \mu_0 &= 0; \\ z_0 &= 0, \end{aligned}$
Step 4:	For $i = 1, 2, \ldots, n-1$, set $\begin{aligned} l_i &= 2(x_{i+1} - x_{i-1}) - h_{i-1}\mu_{i-1} \\ \mu_i &= h_i / l_i; \\ z_i &= (a_i - h_{i-1} z_{i-1}) / l_i. \end{aligned}$
Step 5:	Set $\begin{aligned} l_n &= 1 \\ z_n &= 0; \\ c_n &= 0, \end{aligned}$
Step 6	For $j = n-1, n-2, \ldots, 0$ set $\begin{aligned} c_j &= z_j - \mu_j c_{j+1}; \\ b_j &= (a_{j+1} - a_j)/h_j - h_j(c_{j+1} + 2c_j)/3; \\ d_j &= (c_{j+1} - c_j)/(3h_j). \end{aligned}$
Step 7	OUTPUT $(\{a_i\}_{i=0}^{n-1}, \{b_i\}_{i=0}^{n-1}, \{c_i\}_{i=0}^{n-1}, \{d_i\}_{i=0}^{n-1})$

Example

x_i	-1	0	1	2
$f(x_i) = x_i^3 + 1$	0	1	2	9

We have $h_i = 1$

$$A = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & 4 & 1 & 0 \\ 0 & 1 & 4 & 1 \\ 0 & 0 & 0 & 1 \end{pmatrix},\ b = \begin{pmatrix} 0 \\ 3(2-1) - 3(1-0) \\ 3(9-2) - 3(2-1) \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 18 \\ 0 \end{pmatrix}$$

Example

x_i	-1	0	1	2
$f(x_i) = x_i^3 + 1$	0	1	2	9

We have $c_0 = c_3 = 0$

$$\begin{pmatrix} 4 & 1 \\ 1 & 4 \end{pmatrix} \begin{pmatrix} c_1 \\ c_2 \end{pmatrix} = \begin{pmatrix} 0 \\ 18 \end{pmatrix}$$

Thus

$$\begin{pmatrix} 4 & 1 \\ 0 & \frac{15}{4} \end{pmatrix} \begin{pmatrix} c_1 \\ c_2 \end{pmatrix} = \begin{pmatrix} 0 \\ 18 \end{pmatrix}$$

and $c_2 = \dfrac{6 \cdot 4}{5} = 4.8, \;\; c_1 = -1.2$

Example

x_i	-1	0	1	2
a_i	0	1	2	9
c_i	0	-1.2	4.8	0

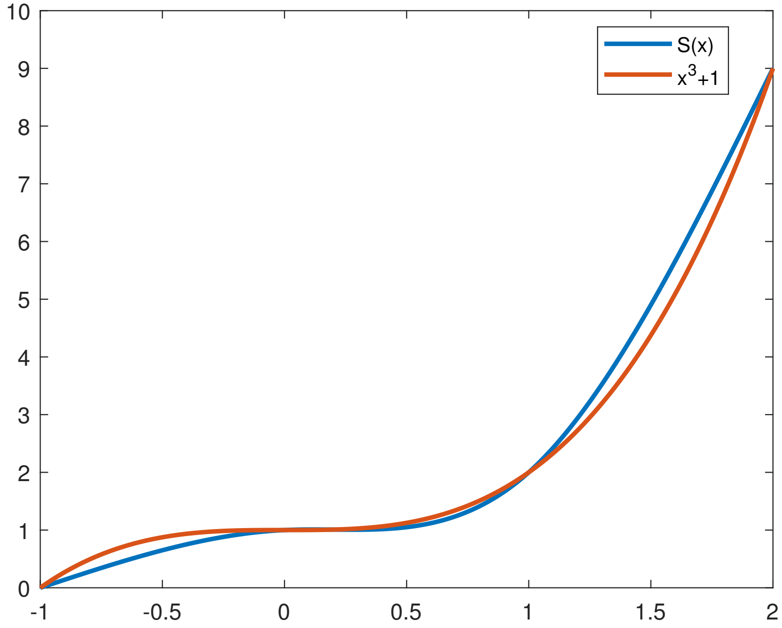
$$d_i = \frac{c_{i+1} - c_i}{3h_i},\ d_0 = -0.4,\ d_1 = 2,\ d_2 = -1.6$$

$$b_i = \frac{(a_{i+1} - a_i)}{h_i} - \frac{h_i(2c_i + c_{i+1})}{3}, b_0 = 1.4,\ b_1 = 0.2,\ b_2 = 3.8$$

$$S(x) = \begin{cases} 1.2(x+1) - 0.4(x+1)^3 & \text{if } x \in [-1, 0] \\ 1 + 0.2x - 1.2x^2 + 2x^3 & \text{if } x \in [0, 1] \\ 2 + 1.2(x-1) + 4.8(x-1)^2 - 1.6(x-1)^3 & \text{if } x \in [1, 2] \end{cases}$$

Example

x_i	-1	0	1	2
a_i	0	1	2	9



Clamped Splines

For clamped splines, instead of using the boundary condition $S''(x_0) = S''(x_n)$, we consider the cubic spline S which matches the first derivative of f at two endpoint x_0 and x_n, namely

$$S'(x_0) = f'(x_0),\ S'(x_n) = f'(x_n).$$

For example,

x_i	0	1	2
$f(x_i)$	1	2	2
$f'(x_i)$	1		0

The spline consists of two cubic polynomials, S_0 and S_1 where S_0 is defined on $[0, 1]$ and S_1 is defined on $[1, 2]$. We consider

$$S_0(x) = a_0 + b_0 x + c_0 x^2 + d_0 x^3$$
$$S_1(x) = a_1 + b_1(x-1) + c_1(x-1)^2 + d_1(x-1)^3$$

$S_i(x_i) = f(x_i),\ S_i(x_{i+1}) = f(x_{i+1})$

x_i	0	1	2
$f(x_i)$	1	2	2
$f'(x_i)$	1		0

Since S fit the data, we have

$$\begin{aligned} a_0 &= f(0) = 1 \\ a_0 + b_0 + c_0 + d_0 &= f(1) = 2 \end{aligned}$$

and

$$\begin{aligned} a_1 &= f(1) = 2 \\ a_1 + b_1 + c_1 + d_1 &= f(2) = 2. \end{aligned}$$

$S_i'(x_{i+1}) = S_{i+1}'(x_{i+1})$

x_i	0	1	2
$f(x_i)$	1	2	2
$f'(x_i)$	1		0

$$S_0'(x) = b_0 + 2c_0x + 3d_0x^2$$
$$S_1'(x) = b_1 + 2c_1(x-1) + 3d_1(x-1)^2$$

and

$$S_0'(1) = b_0 + 2c_0 + 3d_0 = b_1 = S_1'(1)$$
$$S_0'(0) = b_0 = 1,\ S_1'(2) = b_1 + 2c_1 + 3d_1 = 0$$

$S_i''(x_{i+1}) = S_{i+1}''(x_{i+1}),\ S''(x_0) = S''(x_n) = 0$

x_i	0	1	2
$f(x_i)$	1	2	2
$f'(x_i)$	1		0

$$S_0''(x) = 2c_0 + 6d_0 x$$
$$S_1''(x) = 2c_1 + 6d_1(x-1)$$

We have

$$S_0''(1) = 2c_0 + 6d_0 = 2c_1 = S_1''(1)$$

$S_i''(x_{i+1}) = S_{i+1}''(x_{i+1}),\ S''(x_0) = S''(x_n) = 0$

x_i	0	1	2
$f(x_i)$	1	2	2
$f'(x_i)$	1		0

$$
\begin{aligned}
S_0''(x) &= 2c_0 + 6d_0 x \\
S_1''(x) &= 2c_1 + 6d_1(x-1)
\end{aligned}
$$

We have

$$S_0''(1) = 2c_0 + 6d_0 = 2c_1 = S_1''(1)$$

Cont.

x_i	0	1	2
$f(x_i)$	1	2	2
$f'(x_i)$	1		0

$$
\begin{aligned}
b_0 = 1,\ a_0 &= 1 \\
c_0 = -0.25,\ d_0 &= 0.25 \\
a_1 = 2,\ b_1 &= 0.25 \\
c_1 = -0.5,\ d_1 &= 0.25
\end{aligned}
$$

$$
S(x) = \begin{cases} 1 + x - \frac{1}{4}x^2 + \frac{1}{4}x^3 & \text{if } x \in [0,1] \\ 2 + \frac{1}{4}(x-1) - \frac{1}{2}(x-1)^2 + \frac{1}{4}(x-1)^3 & \text{if } x \in (1,2] \end{cases}
$$

Theorem

Theorem

If f is defined at $a = x_0 < x_1 < \cdots < x_n = b$ and differentiable at a and b, then f has a unique clamped cubic spline interpolant S on the nodes $x_0, x_1, \ldots, x_n$, that is, a spline interpolant that satisfies the clamped boundary conditions $S'(a) = f'(a),\ S'(b) = f'(b)$.

Proof.

By using the equations discussed before, we have

$$a_i = f(x_i)\ \forall i = 0, \ldots, n$$

$$d_i = \frac{c_{i+1} - c_i}{3h_i}\ \forall i = 0, \ldots, n-1$$

$$b_i = \frac{(a_{i+1} - a_i)}{h_i} - \frac{h_i(2c_i + c_{i+1})}{3}\ \forall i = 0, \ldots, n-1.$$

and

$$h_{i-1}c_{i-1} + 2(h_{i-1} + h_i)c_i + h_i c_{i+1} = \frac{3}{h_i}(a_{i+1} - a_i) - \frac{3}{h_{i-1}}(a_i - a_{i-1})$$

□

Cont.

Proof.

By the clamped boundary condition, we have

$$S'(x_0) = b_0 = f'(a)$$
$$S'(x_n) = b_n = f'(b)$$

Since $b_0 = \dfrac{(a_1 - a_0)}{h_i} - \dfrac{h_0(2c_0 + c_1)}{3}$, we have

$$2h_0c_0 + h_0c_1 = \frac{3(a_1 - a_0)}{h_i} - 3f'(a).$$

$\square$

Cont.

Proof.

Since $b_n = b_{n-1} + h_{n-1}(c_{n-1} + c_n)$ and
$b_{n-1} = \dfrac{(a_n - a_{n-1})}{h_i} - \dfrac{h_{n-1}(2c_{n-1} + c_n)}{3}$, we have

$$f'(b) = \frac{(a_n - a_{n-1})}{h_i} + \frac{h_{n-1}c_{n-1}}{3} + \frac{2h_{n-1}c_n}{3}$$

Therefore, we have

$$2h_0c_0 + h_0c_1 = \frac{3(a_1 - a_0)}{h_i} - 3f'(a).$$

$$h_{n-1}c_{n-1} + 2h_{n-1}c_n = 3f'(b) - \frac{3(a_n - a_{n-1})}{h_i}$$

$\square$

Cont.

Proof.

Thus, $\{c_i\}_{i=0}^{n}$ satisfies $Ax = b$ where

$$A = \begin{pmatrix} 2h_0 & h_0 & 0 & \cdots & \cdots & 0 \\ h_0 & 2(h_0+h_1) & h_1 & \ddots & & \vdots \\ 0 & h_1 & 2(h_1+h_2) & h_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & 0 \\ \vdots & & \ddots & h_{n-2} & 2(h_{n-2}+h_{n-1}) & h_{n-1} \\ 0 & \cdots & \cdots & 0 & h_{n-1} & 2h_{n-1} \end{pmatrix}$$

$$x = (c_0, c_1, \cdots, c_n)^t$$

$$b = (\frac{3(a_1 - a_0)}{h_1} - 3f'(a), \frac{3(a_2 - a_1)}{h_1} - \frac{3(a_1 - a_0)}{h_0}, \cdots, \frac{3(a_n - a_{n-1})}{h_{n-1}} - \frac{3(a_{n-1} - a_{n-2})}{h_{n-2}}, \frac{3(a_{n-1} - a_{n-2})}{h_{n-2}} - f'(b))^t$$

Therefore, there exist unique coefficients $\{a_i\}, \{c_i\}, \{b_i\}, \{d_i\}$ for the clamped cubic spline. □

Example

x_i	-1	0	1	2
$f(x_i) = x_i^3 + 1$	0	1	2	9
$f'(x_i) = 3x_i^2$	3			12

We have $h_i = 1$

$$A = \begin{pmatrix} 2 & 1 & 0 & 0 \\ 1 & 4 & 1 & 0 \\ 0 & 1 & 4 & 1 \\ 0 & 0 & 1 & 2 \end{pmatrix}, \quad b = \begin{pmatrix} 3(1-0) - 3 \cdot 3 \\ 3(2-1) - 3(1-0) \\ 3(9-2) - 3(2-1) \\ 3 \cdot 12 - 3(9-2) \end{pmatrix} = \begin{pmatrix} -6 \\ 0 \\ 18 \\ 15 \end{pmatrix}$$

Example

x_i	-1	0	1	2
$f(x_i) = x_i^3 + 1$	0	1	2	9
$f'(x_i) = 3x_i^2$	3			12

$$\left(\begin{array}{cccc|c} 2 & 1 & 0 & 0 & -6 \\ 1 & 4 & 1 & 0 & 0 \\ 0 & 1 & 4 & 1 & 18 \\ 0 & 0 & 1 & 2 & 15 \end{array}\right) \to \left(\begin{array}{cccc|c} 2 & 1 & 0 & 0 & -6 \\ 0 & \frac{7}{2} & 1 & 0 & 3 \\ 0 & 1 & 4 & 1 & 18 \\ 0 & 0 & 1 & 2 & 15 \end{array}\right) \to \left(\begin{array}{cccc|c} 2 & 1 & 0 & 0 & -6 \\ 0 & \frac{7}{2} & 1 & 0 & 3 \\ 0 & 0 & \frac{26}{7} & 1 & \frac{120}{7} \\ 0 & 0 & 1 & 2 & 15 \end{array}\right)$$

$$\left(\begin{array}{cccc|c} 2 & 1 & 0 & 0 & -6 \\ 0 & \frac{7}{2} & 1 & 0 & 3 \\ 0 & 0 & \frac{26}{7} & 1 & \frac{120}{7} \\ 0 & 0 & 1 & 2 & 15 \end{array}\right) \to \left(\begin{array}{cccc|c} 2 & 1 & 0 & 0 & -6 \\ 0 & \frac{7}{2} & 1 & 0 & 3 \\ 0 & 0 & \frac{26}{7} & 1 & \frac{120}{7} \\ 0 & 0 & 0 & \frac{45}{26} & \frac{270}{26} \end{array}\right)$$

and $c_3 = 6,\ c_2 = 3,\ c_1 = 0,\ c_0 = -3.$

Example

x_i	-1	0	1	2
a_i	0	1	2	9
c_i	-3	0	3	6

$$d_i = \frac{c_{i+1} - c_i}{3h_i},\ d_0 = 1,\ d_1 = 1,\ d_2 = 1$$

$$b_i = \frac{(a_{i+1} - a_i)}{h_i} - \frac{h_i(2c_i + c_{i+1})}{3}, b_0 = 3,\ b_1 = 0,\ b_2 = 3$$

$$S(x) = \begin{cases} 3(x+1) - 3(x+1)^2 + (x+1)^3 = (x+1)(x^2 - x + 1) = x^3 + 1 \\ 1 + x^3 \\ 2 + 3(x-1) + 3(x-1)^2 + (x-1)^3 = x^3 + 1 \end{cases}$$

Theorem

Theorem

Let $f \in C^4([a,b])$ with $\max_{a \le x \le b} |f^{(4)}(x)| = M$. If S is the unique clamped cubic spline interpolant to f with respect to the nodes $a = x_0 < x_1 < \cdots < x_n = b$, then, for all x in $[a,b]$

$$|f(x) - S(x)| \le \frac{5M}{384} \max_{0 \le j \le n-1} (x_{j+1} - x_j)^4.$$

Lecture 10

Numerical Differentiation

In many applications, we may need to approximate the derivative of a function using some given data $\{(x_i, f(x_i))\}_{i=0}^{n}$.
For example, we want to approximate the velocity of an object, v .
Sometimes, it is not easy to measure the velocity directly but we may have the displacement measurement of the object, s (position of the object).
We can use the displacement data to approximate the velocity. For example, $v(x) = \dfrac{s(x_{i+1}) - s(x_i)}{x_{i+1} - x_i}$ for $x \in (x_i, x_{i+1})$.
In the first few weeks, we discussed polynomial interpolations for a given set of data $\{(x_i, f(x_i))\}_{i=0}^{n}$ which provide an polynomial approximation of f.
Instead of approximating the function value of f, interpolation polynomial also give us a easier way to estimate the derivative of f.

Example: Linear Lagrange polynomial

Assume $f \in C^2$. By the error estimate for the Lagrange polynomial, given x_0 and x_1, we have

$$\begin{aligned} f(x) &= P_{0,1}(x) + \frac{(x-x_0)(x-x_1)}{2} f^{(2)}(\xi(x)) \\ &= \frac{(x-x_1)f(x_0)}{x_0-x_1} + \frac{(x-x_0)f(x_1)}{x_1-x_0} + \frac{(x-x_0)(x-x_1)}{2} f^{(2)}(\xi(x)) \end{aligned}$$

where $\xi(x)$ is between x_1 and x_0.
Assuming $f^{(2)}(\xi(x))$ is differentiable. We have

$$f'(x) = \frac{f(x_0)}{x_0-x_1} + \frac{f(x_1)}{x_1-x_0} + \frac{2x-x_1-x_0}{2} f^{(2)}(\xi(x)) + \frac{(x-x_0)(x-x_1)}{2} \frac{d}{dx} f^{(2)}(\xi(x)).$$

For $x \in [x_0, x_1]$, we have $\left| \frac{2x-x_1-x_0}{2} f^{(2)}(\xi(x)) \right| \leq M_2 |x_1 - x_0|$ and $\left| \frac{(x-x_0)(x-x_1)}{2} \frac{d}{dx} f^{(2)}(\xi(x)) \right| = O(|x_1-x_0|^2)$. Thus we have

$$f'(x) \approx \frac{f(x_0)}{x_0-x_1} + \frac{f(x_1)}{x_1-x_0} = \frac{f(x_1)-f(x_0)}{x_1-x_0}$$

Cont.

The term $\dfrac{(x-x_0)(x-x_1)}{2}\dfrac{d}{dx}f^{(2)}(\xi(x))$ is not easy to estimate. However, when $x = x_0$ or $x = x_1$, we obtain that this term equal to zero. Therefore, we have

$$\left|f'(x_0) - \frac{f(x_1)-f(x_0)}{x_1-x_0}\right| = \frac{|x_1-x_0|}{2}\cdot\left|f^{(2)}(\xi(x_0))\right|$$

$$\left|f'(x_1) - \frac{f(x_1)-f(x_0)}{x_1-x_0}\right| = \frac{|x_1-x_0|}{2}\cdot\left|f^{(2)}(\xi(x_1))\right|$$

For $x_1 - x_0 = h > 0$, we have

$$f'(x_0) \approx \frac{f(x_0+h)-f(x_0)}{h} \text{ forward-difference formula}$$

$$f'(x_1) \approx \frac{f(x_1)-f(x_1-h)}{h} \text{ backward-difference formula}$$

and both of the formulas have a error bound $\frac{h}{2}\max_{x\in[x_0,x_1]}\{|f^{(2)}(x)|\}$.

Another way to obtain the formula

In fact, we can use Taylar's Theorem to obtain the two point formula. Assuming $f \in C^2$, we have

$$f(x) = f(x_0) + f'(x_0)(x - x_0) + \frac{(x - x_0)^2}{2} f^{(2)}(\xi).$$

Therefore, we have

$$f'(x_0) = \frac{f(x) - f(x_0)}{x - x_0} - \frac{(x - x_0)}{2} f^{(2)}(\xi).$$

For $x = x_1 = x_0 + h$, we have

$$f'(x_0) = \frac{f(x_0 + h) - f(x_0)}{h} - \frac{h}{2} f^{(2)}(\xi).$$

Example

Consider $f = e^x$, $x_0 = 0$ and $x_1 = 0.1, 0.05, 0.01$. We can approximate $\dfrac{d}{dx}e^x$ at $x_0 = 0$ by

$$\frac{de^x}{dx}(0) \approx \frac{e^{0.1} - e^0}{0.1} \approx 1.0517$$

$$\frac{de^x}{dx}(0) \approx \frac{e^{0.05} - e^0}{0.05} \approx 1.0254$$

and

$$\frac{de^x}{dx}(0) \approx \frac{e^{0.01} - e^0}{0.01} \approx 1.0050$$

We have the error bound is $\dfrac{h}{2}\max_{x\in[x_0,x_1]}\{|f^{(2)}(x)|\} \leq \dfrac{h}{2}\dfrac{d^2e^x}{dx^2}(h) = \dfrac{h}{2}e^h = 0.0553, 0.0263, 0.0051$ for $x_1 = 0.1, 0.05, 0.01$

Example

Consider $f = e^x$, $x_0 = -0.1, -0.05, -0.01$ and $x_1 = 0$. We can approximate $\dfrac{d}{dx}e^x$ at 0 by

$$\frac{de^x}{dx}(0) \approx \frac{e^0 - e^{-0.1}}{0.1} \approx 0.9516$$

$$\frac{de^x}{dx}(0) \approx \frac{e^0 - e^{-0.05}}{0.05} \approx 0.9754$$

and

$$\frac{de^x}{dx}(0) \approx \frac{e^0 - e^{-0.01}}{0.01} \approx 0.9950$$

We have the error bound is $\dfrac{h}{2}\max_{x\in[x_0,x_1]}\{|f^{(2)}(x)|\} \approx \dfrac{h}{2}\dfrac{d^2e^x}{dx^2}(0) = \dfrac{h}{2} = 0.05, 0.025, 0.005$ for $x_0 = -0.1, -0.05, -0.01$.

$(n+1)$ -point formula

Next, we will use the n-th Lagrange interpolating polynomial to approximate the first derivative of f. Given $x_0, x_1, \ldots, x_n$, if $f \in C^{(n+2)}$, we have

$$f(x) = \sum_{k=0}^{n} f(x_k)L_k(x) + \frac{(x-x_0)\cdots(x-x_n)}{(n+1)!}f^{(n+1)}(\xi(x))$$

We assume $f^{(n+1)}(\xi(x))$ is differentiable and obtain

$$f'(x) = \sum_{k=0}^{n} f(x_k)L'_k(x) + \frac{d}{dx}\left(\frac{(x-x_0)\cdots(x-x_n)}{(n+1)!}\right)f^{(n+1)}(\xi(x))$$
$$+ \frac{(x-x_0)\cdots(x-x_n)}{(n+1)!}\Big)\frac{d}{dx}f^{(n+1)}(\xi(x))$$

Sicne $\frac{d}{dx}((x-x_0)\cdots(x-x_n)) = (x-x_j)\frac{d}{dx}\prod_{i\neq j}(x-x_i) + \prod_{i\neq j}(x-x_i)$, we have

$$f'(x_j) = \sum_{k=0}^{n} f(x_k)L'_k(x_j) + \frac{\prod_{i\neq j}(x_j-x_i)}{(n+1)!}f^{(n+1)}(\xi(x_j))$$

which called an $(n+1)$ -point formula to approximate $f'(x_j)$.

Three points formula

We then consider the three points formula to approximate $f'(x_0)$. We have

$$L_0(x) = \frac{(x - x_1)(x - x_2)}{(x_0 - x_1)(x_0 - x_2)}$$
$$L_1(x) = \frac{(x - x_0)(x - x_2)}{(x_1 - x_0)(x_1 - x_2)}$$
$$L_2(x) = \frac{(x - x_0)(x - x_1)}{(x_2 - x_0)(x_2 - x_1)}$$

and $L_0'(x) = \dfrac{2x - x_1 - x_2}{(x_0 - x_1)(x_0 - x_2)}$, $L_1'(x) = \dfrac{2x - x_0 - x_2}{(x_1 - x_0)(x_1 - x_2)}$, $L_2'(x) = \dfrac{2x - x_0 - x_1}{(x_2 - x_0)(x_2 - x_1)}$.

Cont.

$$f'(x_j) = f(x_0)\frac{2x_j - x_1 - x_2}{(x_0 - x_1)(x_0 - x_2)} + f(x_1)\frac{2x_j - x_0 - x_2}{(x_1 - x_0)(x_1 - x_2)} + f(x_2)\frac{2x_j - x_0 - x_1}{(x_2 - x_0)(x_2 - x_1)}$$
$$+ \frac{f^{(3)}(\xi_j)}{6}\prod_{j \neq k}(x_j - x_k)$$

For considering $j = 0, 1, 2$, we have three different three points formulas.

Three points formula ($j = 0$)

Consider $x_1 = x_0 + h$ and $x_2 = x_0 + 2h$. We have

$$
\begin{aligned}
f'(x_0) =& f(x_0)\frac{2x_0 - x_1 - x_2}{(x_0 - x_1)(x_0 - x_2)} + f(x_1)\frac{2x_0 - x_0 - x_2}{(x_1 - x_0)(x_1 - x_2)} + f(x_2)\frac{2x_0 - x_0 - x_1}{(x_2 - x_0)(x_2 - x_1)} \\
&+ \frac{f^{(3)}(\xi_0)}{6}(x_0 - x_1)(x_0 - x_2) \\
=& f(x_0)\frac{-3h}{(-h)(-2h)} + f(x_0 + h)\frac{-2h}{(h)(-h)} + f(x_0 + 2h)\frac{-h}{(2h)(h)} + \frac{f^{(3)}(\xi_0)}{6}(-h)(-2h) \\
=& \frac{1}{h}\Big(-\frac{3}{2}f(x_0) + 2f(x_0 + h) - \frac{1}{2}f(x_0 + 2h)\Big) + \frac{f^{(3)}(\xi_0)}{3}h^2
\end{aligned}
$$

where $\xi_0 \in (x_0, x_0 + 2h)$

Three points formula ($j = 2$)

Consider $x_0 = x_2 - 2h$ and $x_1 = x_2 - h$. We have

$$\begin{aligned}
f'(x_2) =& f(x_0)\frac{2x_2 - x_1 - x_2}{(x_0 - x_1)(x_0 - x_2)} + f(x_1)\frac{2x_2 - x_0 - x_2}{(x_1 - x_0)(x_1 - x_2)} + f(x_2)\frac{2x_2 - x_0 - x_1}{(x_2 - x_0)(x_2 - x_1)} \\
&+ \frac{f^{(3)}(\xi_2)}{6}(x_2 - x_0)(x_2 - x_1) \\
=& f(x_2 - 2h)\frac{h}{(-h)(-2h)} + f(x_2 - h)\frac{2h}{(h)(-h)} + f(x_2)\frac{3h}{(2h)(h)} + \frac{f^{(3)}(\xi_2)}{6}(h)(2h) \\
=& \frac{1}{h}\Big(\frac{3}{2}f(x_2) - 2f(x_2 - h) + \frac{1}{2}f(x_2 - 2h)\Big) + \frac{f^{(3)}(\xi_2)}{3}h^2
\end{aligned}$$

where $\xi_2 \in (x_2 - 2h, x_2)$

These two formulas are called three-point left endpoint formula and three-point right endpoint formula.

Three points formula ($j = 1$)

Consider $x_0 = x_1 - h$ and $x_2 = x_1 + h$. We have

$$
\begin{aligned}
f'(x_1) =& f(x_0)\frac{2x_1 - x_1 - x_2}{(x_0 - x_1)(x_0 - x_2)} + f(x_1)\frac{2x_1 - x_0 - x_2}{(x_1 - x_0)(x_1 - x_2)} + f(x_2)\frac{2x_1 - x_0 - x_1}{(x_2 - x_0)(x_2 - x_1)} \\
&+ \frac{f^{(3)}(\xi_1)}{6}(x_1 - x_0)(x_1 - x_2) \\
=& f(x_1 - h)\frac{-h}{(-h)(-2h)} + f(x_1)\frac{0}{(h)(-h)} + f(x_1 + h)\frac{h}{(2h)(h)} + \frac{f^{(3)}(\xi_1)}{6}(h)(-h) \\
=& \frac{1}{h}\Big(\frac{1}{2}f(x_1 + h) - \frac{1}{2}f(x_1 - h)\Big) - \frac{f^{(3)}(\xi_1)}{6}h^2
\end{aligned}
$$

where $\xi_1 \in (x_1 - h, x_1 + h)$.
This formula is called three-point midpoint formula.

Example (three-point midpoint formula)

Consider $f = e^x$, $x_0 = -h$ and $x_1 = 0$ and $x_2 = h$ with $h = 0.1, 0, 05, 0.01$.

We can approximate $\dfrac{d}{dx}e^x$ at 0 by

$$\frac{de^x}{dx}(0) \approx \frac{e^{0.1} - e^{-0.1}}{0.2} \approx 1.0017$$

$$\frac{de^x}{dx}(0) \approx \frac{e^{0.05} - e^{-0.05}}{0.1} \approx 1.0004$$

and

$$\frac{de^x}{dx}(0) \approx \frac{e^{0.01} - e^{-0.01}}{0.02} \approx 1.0000$$

We have the error bound is

$\dfrac{h^2}{6}\max_{x\in[x_0,x_1]}\{|f^{(3)}(x)|\} \approx \dfrac{h^2}{6}\dfrac{d^3e^x}{dx^3}(0) = \dfrac{h^2}{6} \approx 0.0017, 4.167\times 10^{-4}, 1.667\times 10^{-5}$.

Five-Point Formula

By using the same argument, we can obtain the five-point Formula

$$f'(x_0) = \frac{1}{12h}[f(x_0 - 2h) - 8f(x_0 - h) + 8f(x_0 + h) - f(x_0 + 2h)] + \frac{h^4}{30}f^{(5)}(\xi)$$

which is the Five-Point Midpoint Formula and

$$\begin{aligned} f'(x_0) = &\frac{1}{12h}[-25f(x_0) + 48f(x_0 + h) - 36f(x_0 + 2h) + 16f(x_0 + 3h) - 3f(x_0 + 4h)] \\ &+ \frac{h^4}{5}f^{(5)}(\xi) \end{aligned}$$

which is the Five-Point Endpoint Formula.

Lecture 11

Numerical Differentiation

Review: Numerical Differentiation

Given $\{x_i\}_{i=0}^n \subset [a, b]$, we would like to approximate the derivative of a function at x_i, $f'(x_i)$ by using the function value of function at the neighboring points of x_i.
E.g. The forward difference formula gives an approximation of $f'(x_i)$ as

$$f'(x_i) \approx \frac{f(x_{i+1}) - f(x_i)}{x_{i+1} - x_i}$$

and the three points midpoint formula gives an approximation of $f'(x_i)$ as

$$f'(x_i) \approx \frac{f(x_i + h) - f(x_i - h)}{2h}.$$

Example discussed before

Assume the function value of f at $x = x_0, x_0 \pm h, x_0 \pm 2h, \ldots$ are given.
Two points formula:

$$f'(x_0) \approx \frac{f(x_0+h) - f(x_0)}{h} \text{ forward-difference formula}$$

$$f'(x_0) \approx \frac{f(x_0) - f(x_1-h)}{h} \text{ backward-difference formula}$$

Three points formula:

$$f'(x_0) \approx \frac{1}{h}\Big(-\frac{3}{2}f(x_0) + 2f(x_0+h) - \frac{1}{2}f(x_0+2h)\Big)$$

$$f'(x_0) \approx \frac{1}{h}\Big(\frac{3}{2}f(x_0) - 2f(x_0-h) + \frac{1}{2}f(x_0-2h)\Big)$$

$$f'(x_0) = \frac{1}{h}\Big(\frac{1}{2}f(x_0+h) - \frac{1}{2}f(x_0-h)\Big)$$

Five points formula:

$$f'(x_0) \approx \frac{1}{12h}[f(x_0-2h) - 8f(x_0-h) + 8f(x_0+h) - f(x_0+2h)]$$

$$f'(x_0) \approx \frac{1}{12h}[-25f(x_0) + 48f(x_0+h) - 36f(x_0+2h) + 16f(x_0+3h) - 3f(x_0+4h)]$$

Derive Numerical Differentiation formula from Taylor series

In general, if f is smooth enough, Taylor theorem can help us to obtain the finite difference formula to approximate $f^{(k)}(x_i)$ for any $k > 1$ with any order of accuracy.
Assume $f \in C^{k+n}$ and $x_i = x_0 + ih$ for $i = \pm 1, \pm 2, \pm 3, \ldots$. By Taylar's Theorem, we have

$$\begin{aligned} f(x_j) =& f(x_0) + (x_j - x_0)f'(x_0) + \frac{(x_j - x_0)^2}{2} f^{(2)}(x_0) + \cdots + \frac{(x_j - x_0)^{k+n-1}}{(k+n-1)!} f^{(k+n-1)}(x_0) \\ &+ \frac{(x_j - x_0)^{k+n}}{(k+n)!} f^{(k+n)}(\xi_j) \\ =& f(x_0) + \sum_{s=1}^{k+n-1} \frac{(jh)^s}{s!} f^{(s)}(x_0) + \frac{(jh)^{k+n}}{(k+n)!} f^{(k+n)}(\xi_j) \end{aligned}$$

We would like to find $\{\alpha_i\}_{i=m}^{m+K}$ such that

$$f^{(k)}(x_0) = \sum_{j=m}^{m+K} \frac{\alpha_j}{h^k} f(x_j) + O(h^n)$$

Here we would like to find a $K+1$ points formula with n order accuracy to approximate $f^{(k)}(x_0)$.

Cont.

By Taylar's Theorem, we have

$$\sum_{j=m}^{m+K} \alpha_j f(x_j) = (\sum_{j=m}^{m+K} \alpha_j) f(x_0) + \sum_{s=1}^{k+n-1} \Big(\sum_{j=m}^{m+K} \alpha_j \frac{(jh)^s}{s!} \Big) f^{(s)}(x_0) + \frac{\sum_{j=m}^{m+K} \alpha_j (jh)^{k+n} f^{(k+n)}(\xi_j)}{(k+n)!}$$

If we can find a set of coefficients such that

$$\sum_{j=m}^{m+K} \alpha_j \frac{(j)^s}{s!} = \delta_{sk} = \begin{cases} 1 & \text{if } s = k \\ 0 & \text{if } s \neq k \end{cases} \text{ for } s = 0, \ldots, k+n-1,$$

we have

$$\begin{aligned} \sum_{j=m}^{m+K} \frac{\alpha_j}{h^k} f(x_j) &= \frac{1}{h^k} \Big((\sum_{j=m}^{m+K} \alpha_j) f(x_0) + \sum_{s=1}^{k+n-1} \Big(\sum_{j=m}^{m+K} \alpha_j \frac{(jh)^s}{s!} \Big) f^{(s)}(x_0) + \frac{\sum_{j=m}^{m+K} \alpha_j (jh)^{k+n} f^{(k+n)}(\xi_j)}{(k+n)!} \Big) \\ &= \frac{1}{h^k} \Big(\sum_{s=0}^{k+n-1} \delta_{sk} h^s f^{(s)}(x_0) + \frac{\sum_{j=m}^{m+K} (jh)^{k+n} f^{(k+n)}(\xi_j)}{(k+n)!} \Big) \\ &= f^{(k)}(x_0) + \frac{1}{h^k} \frac{\sum_{j=m}^{m+K} \alpha_j (jh)^{k+n} f^{(k+n)}(\xi_j)}{(k+n)!} \end{aligned}$$

Cont.

Moreover, we have

$$\frac{\sum_{j=m}^{m+K} \alpha_j (jh)^{k+n} f^{(k+n)}(\xi_j)}{(k+n)!} = h^{(k+n)} \frac{\sum_{j=m}^{m+K} \alpha_j j^{k+n} f^{(k+n)}(\xi_j)}{(k+n)!} = O(h^{k+n})$$

Therefore, we have

$$\sum_{j=m}^{m+K} \frac{\alpha_j}{h^k} f(x_j) = f^{(k)}(x_0) + O(h^n).$$

Example

Consider $K = 2$, $m = -1$, $k = 1$ and $n = 2$. We would like to find $\alpha_{-1}, \alpha_0, \alpha_1$ such that

$$\begin{aligned} \alpha_{-1} + \alpha_0 + \alpha_1 &= 0, \\ -1 \cdot \alpha_{-1} + 0 \cdot \alpha_0 + 1 \cdot \alpha_1 &= 1, \\ \frac{1}{2} \cdot \alpha_{-1} + 0 \cdot \alpha_0 + \frac{1}{2} \cdot \alpha_1 &= 0. \end{aligned}$$

Therefore we will solve the following linear system

$$\begin{pmatrix} 1 & 1 & 1 \\ -1 & 0 & 1 \\ \frac{1}{2} & 0 & \frac{1}{2} \end{pmatrix} \begin{pmatrix} \alpha_{-1} \\ \alpha_0 \\ \alpha_1 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}$$

and obtain $\alpha_{-1} = -\frac{1}{2}$, $\alpha_0 = 0$, and $\alpha_1 = \frac{1}{2}$ which gives

$$f'(x_0) \approx \frac{1}{h}\Big(\frac{1}{2}f(x_0 + h) - \frac{1}{2}f(x_0 - h)\Big).$$

Example

Consider $K = 2$, $m = 0$, $k = 1$ and $n = 2$. We would like to find $\alpha_0, \alpha_1, \alpha_2$ such that

$$\begin{aligned} \alpha_0 + \alpha_1 + \alpha_2 &= 0, \\ 0 \cdot \alpha_0 + 1 \cdot \alpha_1 + 2 \cdot \alpha_2 &= 1, \\ 0 \cdot \alpha_0 + \frac{1}{2} \cdot \alpha_1 + 2 \cdot \alpha_2 &= 0. \end{aligned}$$

Therefore we will solve the following linear system

$$\begin{pmatrix} 1 & 1 & 1 \\ 0 & 1 & 2 \\ 0 & \frac{1}{2} & 2 \end{pmatrix} \begin{pmatrix} \alpha_0 \\ \alpha_1 \\ \alpha_2 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}$$

and obtain $\alpha_2 = -\frac{1}{2}$, $\alpha_1 = 2$, and $\alpha_0 = -\frac{3}{2}$ which gives

$$f'(x_0) \approx \frac{1}{h}\Big(-\frac{3}{2} f(x_0) + 2f(x_0 + h) - \frac{1}{2} f(x_0 + 2h)\Big).$$

Number of points required

Then the question is when will the system of equations

$$\sum_{j=m}^{m+K} \alpha_j \frac{(j)^s}{s!} = \delta_{sk} = \begin{cases} 1 & \text{if } s = k \\ 0 & \text{if } s \neq k \end{cases} \text{ for } s = 0, \ldots, k+n-1$$

be solvable.
We can written the problem in matrix form which is solving α_i such that

$$\begin{pmatrix} 1 & 1 & 1 & \cdots & 1 \\ (x_m - x_0) & (x_{m+1} - x_0) & (x_{m+2} - x_0) & \cdots & (x_{m+K} - x_0) \\ (x_m - x_0)^2 & (x_{m+1} - x_0)^2 & (x_{m+2} - x_0)^2 & \cdots & (x_{m+K} - x_0)^2 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ (x_m - x_0)^{n+k-1} & (x_{m+1} - x_0)^{n+k-1} & (x_{m+2} - x_0)^{n+k-1} & \cdots & (x_{m+K} - x_0)^{n+k-1} \end{pmatrix} \begin{pmatrix} \alpha_0 \\ \alpha_m \\ \alpha_{m+1} \\ \vdots \\ \alpha_{m+K} \end{pmatrix} = k!\hat{e}_k$$

where $\hat{e}_k$ is the $k-$th elementary unit vector.

Cont.

We can see that in fact, the matrix
$$\begin{pmatrix} 1 & 1 & 1 & \cdots & 1 \\ (x_m - x_0) & (x_{m+1} - x_0) & (x_{m+2} - x_0) & \cdots & (x_{m+K} - x_0) \\ (x_m - x_0)^2 & (x_{m+1} - x_0)^2 & (x_{m+2} - x_0)^2 & \cdots & (x_{m+K} - x_0)^2 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ (x_m - x_0)^{n+k-1} & (x_{m+1} - x_0)^{n+k-1} & (x_{m+2} - x_0)^{n+k-1} & \cdots & (x_{m+K} - x_0)^{n+k-1} \end{pmatrix}$$ is the transpose of the matrix used to obtain the Lagrange interpolating polynomial.

Therefore, we have the system has a unique solution if $K = n + k - 1$.

We remark that, sometimes, the numerical differentiation formula using $K + 1 = n + k$ points to approximate $f^{(k)}(x_i)$ can have an order of accuracy higher than n.

Example

For using $f(x_{-1})$ and $f(x_1)$ to approximate $f(x_0)$, we have

$$\begin{pmatrix} 1 & 1 \\ -1 & 1 \end{pmatrix} \begin{pmatrix} \alpha_{-1} \\ \alpha_1 \end{pmatrix} = \begin{pmatrix} 0 \\ 2 \end{pmatrix}$$

and thus, $\alpha_{-1} = -\dfrac{1}{2}$ and $\alpha_1 = \dfrac{1}{2}$. Moreover, $(\alpha_{-1}, \alpha_1)^t$ also satisfies

$$\begin{pmatrix} (-1)^0 & 1 \\ (-1)^0 & 1 \\ (-1)^2 & 1 \end{pmatrix} \begin{pmatrix} \alpha_{-1} \\ \alpha_1 \end{pmatrix} = \begin{pmatrix} 0 \\ 2 \\ 0 \end{pmatrix}$$

and thus this formula has second order accuracy ($K = 1$ and $k = 1$ but $n = 2$).

Approximation of 2nd derivative

Using the system you introduce before, it is not hard to obtain a numerical differentiation formula to approximate $f^{(2)}(x_i)$.
We consider $x_{-1} = x_0 - h$, $x_1 = x_0 + h$ and we would like to find α_{-1}, α_0, α_1 such that

$$\begin{pmatrix} 1 & 1 & 1 \\ -1 & 0 & 1 \\ 1 & 0 & 1 \end{pmatrix} \begin{pmatrix} \alpha_{-1} \\ \alpha_0 \\ \alpha_1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 2 \end{pmatrix}.$$

We have $\alpha_0 = -2,\ \alpha_1 = 1.\ \alpha_{-1} = 1.$ Thus, we have

$$f^{(2)}(x_i) \approx \frac{1}{h^2}(f(x_0 + h) - 2f(x_0) + f(x_0 - h)).$$

In fact, we check that this set of coefficients also satisfies

$$\begin{pmatrix} 1 & 1 & 1 \\ -1 & 0 & 1 \\ 1 & 0 & 1 \\ -1 & 0 & 1 \end{pmatrix} \begin{pmatrix} \alpha_{-1} \\ \alpha_0 \\ \alpha_1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 2 \\ 0 \end{pmatrix}$$

Thus, this numerical differentiation formula has 2nd order accuracy and in fact, we have the Second Derivative Midpoint Formula

$$f^{(2)}(x_i) = \frac{1}{h^2}(f(x_0+h) - 2f(x_0) + f(x_0-h)) - \frac{h^2}{12}f^{(4)}(\xi)$$

for some $\xi \in (x_0 - h, x_0 + h)$

Example (Second Derivative Midpoint Formula)

Consider $f = xe^x$, $x_0 = 1$ and $x_{-1} = -h$ and $x_1 = h$ with $h = 0.1, 0, 05, 0.01$. We can approximate $\frac{d^2}{dx^2}(xe^x) = (x+2)e^x = 3e^3 \approx 8.1548$ at 1 by

$$\frac{de^x}{dx}(0) \approx \frac{1.1e^{1.1} - 2(1e^1) - 0.9e^{0.9}}{(0.1)^2} \approx 8.1662$$

$$\frac{de^x}{dx}(0) \approx \frac{1.05e^{1.05} - 2(1e^1) - 0.95e^{0.95}}{(0.05)^2} \approx 8.1577$$

and

$$\frac{de^x}{dx}(0) \approx \frac{1.01e^{1.01} - 2(1e^1) - 0.99e^{0.99}}{(0.01)^2} \approx 8.1550$$

We have the error bound is $\frac{h^2}{12} \max_{x \in [x_{-1}, x_1]}\{|f^{(4)}(x)|\} \approx \frac{h^2}{12}\frac{d^4(xe^x)}{dx^4}(1) = \frac{h^2 5e^1}{12} \approx 0.0113, 0.0028, 1.1326 \times 10^{-4}$.

Round-Off Error Instability

Next, we will discuss how Round-off error affect the stability of the approximation accuracy.
We note that for using a numerical differentiation to approximate $f^{(k)}$, we normally consider $f^{(k)} \approx \frac{1}{h^k}\sum_m \alpha_m f(x_m)$.
Since we have round off error for evaluating $f(x_m)$, we assume our computations actually use the values $\tilde{f}(x_m)$ and

$$f(x_m) = \tilde{f}(x_m) + \epsilon_m.$$

We have

$$\begin{aligned}\left|f^{(k)} - \frac{1}{h^k}\sum_m \alpha_m \tilde{f}(x_m)\right| &\leq \left|f^{(k)} - \frac{1}{h^k}\sum_m \alpha_m f(x_m)\right| + \frac{1}{h^k}\left|\sum_m \alpha_m (f(x_m) - \tilde{f}(x_m))\right| \\ &\leq O(h^n) + \frac{\epsilon \sum_m |\alpha_m|}{h^k}\end{aligned}$$

where $\epsilon = \max_m\{|\epsilon_m|\}$.

Example

For $f = e^x$,we approximate the 2nd derivative of f at $x_0 = 0$ by

$$f''(0) \approx \frac{f(h) - 2f(0) + f(-h)}{h^2}.$$

The following table shows the error for $h = 10^{-i}$

$i =$	1	2	3	4	5	6	7
error	8.33×10^{-4}	8.33×10^{-6}	8.36×10^{-6}	6.08×10^{-9}	8.27×10^{-8}	8.89×10^{-5}	0.0230
$h^2/12$	8.33×10^{-4}	8.33×10^{-6}	8.33×10^{-8}	8.33×10^{-10}	8.33×10^{-10}	8.33×10^{-12}	8.33×10^{-1}
$\frac{\epsilon}{h^2}$	2.22×10^{-14}	2.22×10^{-12}	2.22×10^{-10}	2.22×10^{-8}	2.22×10^{-6}	2.22×10^{-4}	0.0222

Lecture 12

Richardson's extrapolation

Richardson's extrapolation

Richardson's extrapolation is used to generate high-accuracy results while using low order formulas.
Extrapolation can be applied whenever it is known that an approximation technique has an error term with a predictable form, one that depends on a parameter, usually the step size h.
Suppose that for each number $h \neq 0$, we have a formula $N_1(h)$ that approximates an unknown constant M, and that the truncation error involved with the approximation has the form

$$M - N_1(h) = K_1 h + K_2 h^2 + K_3 h^3 + \cdots ,$$

for some collection of unknown constants $K_1, K_2, K_3, \ldots$.

Cont.

If $K_1 \neq 0$, the truncation error is $O(h)$ and we have

$$M - N_1(h) \approx K_1 h.$$

The object of Richardson's extrapolation is to combine these rather inaccurate $O(h)$ approximations in an appropriate way to produce formulas with a higher-order truncation error.
For example, we want to find an approximation formula $N_2(h)$, such that

$$M - N_2(h) = \hat{K}_2 h^2 + \hat{K}_3 h^3 + \cdots$$

where $\hat{K}_2, \hat{K}_3, \ldots$ are some unknown constants. If h is small, since $M - N_2(h) = O(h^2)$, we have

$$|M - N_2(h)| \approx |\hat{K}_2| h^2 \ll |K_1| h \approx |M - N_1(h)|$$

Obtain 2nd order formula from a 1st order formula

Using a first order formula N_1 with two different step size, we have

$$M = N_1(h) + K_1 h + K_2 h^2 + K_3 h^3 + \cdots$$

$$M = N_1(\frac{h}{2}) + K_1 \frac{h}{2} + K_2 \frac{h^2}{4} + K_3 \frac{h^3}{8} + \cdots .$$

Multiple the second equation by 2 and subtract it by the first equation, we will obtain

$$2M - M = 2N_1(\frac{h}{2}) - N_1(h) + K_1(h - h) + K_2(\frac{h^2}{2} - h^2) + K_3(\frac{h^3}{4} - h^3) + \cdots$$

$$\implies M = 2N_1(\frac{h}{2}) - N_1(h) - K_2 \frac{h^2}{2} - \frac{3}{4} K_3 h^3 + \cdots .$$

We consider $N_2(h) = 2N_1(\frac{h}{2}) - N_1(h)$. We then have $M - N_2(h) = O(h^2)$ and

$$M - N_2(h) = -K_2 \frac{h^2}{2} - \frac{3}{4} K_3 h^3 + \cdots .$$

Example

For using forward difference to approximate $\dfrac{de^x}{dx}(0)$, we have

$$\frac{de^x}{dx}(0) - \frac{e^h - e^0}{h} = O(h)$$

For $h = 0.1$, we have

$$\frac{de^x}{dx}(0) \approx \frac{e^{0.1} - e^0}{0.1} \approx 1.0517$$

and for $h = 0.05$, we have

$$\frac{de^x}{dx}(0) \approx \frac{e^{0.05} - e^0}{0.05} \approx 1.0254.$$

To obtain a higher order estimate, we can approximate $\dfrac{de^x}{dx}(0)$ by

$$\frac{de^x}{dx}(0) \approx 2\frac{e^{0.05} - e^0}{0.05} - \frac{e^{0.1} - e^0}{0.1} \approx 0.9991$$

Richardson's extrapolation (even terms only)

Next, we consider the truncation error has the form

$$\sum_{j=1}^{m-1} K_j h^{\alpha_j} + O(h^{\alpha_m})$$

for a collection of constants K_j and when $\alpha_1 < \alpha_2 < \cdots < \alpha_{m-1} < \alpha_m$.
In many cases, we have a formula have truncation errors that contain only even powers of h, that is

$$M = N_1(h) + K_1 h^2 + K_2 h^4 + K_3 h^6 + \cdots .$$

By replacing h by $\frac{h}{2}$, we have

$$M = N_1(\frac{h}{2}) + K_1 \frac{h^2}{4} + K_2 \frac{h^4}{16} + K_3 \frac{h^6}{64} + \cdots .$$

Obtain 4th order formula from a 2nd order formula

Multiplying the second equation by 4 and subtracting it by the first equation, we have

$$4M - M = 4N_1(\frac{h}{2}) - N_1(h) + K_1(h^2 - h^2) + K_2(\frac{h^4}{4} - h^4) + K_3(\frac{h^6}{16} - h^6) + \cdots .$$

$$\Longrightarrow \; 3M = 4N_1(\frac{h}{2}) - N_1(h) - K_2\frac{3h^4}{4} - K_3(\frac{15h^6}{16}) + \cdots$$

$$\Longrightarrow \; M = \frac{4}{3}N_1(\frac{h}{2}) - \frac{1}{3}N_1(h) - K_2\frac{h^4}{4} - K_3\frac{5h^6}{16} + \cdots$$

Defining

$$N_2(h) = \frac{4}{3}N_1(\frac{h}{2}) - \frac{1}{3}N_1(h) = N_1(\frac{h}{2}) + \frac{1}{3}(N_1(\frac{h}{2}) - N_1(h)),$$

we have

$$M = N_2(h) - K_2\frac{h^4}{4} - K_3\frac{5h^6}{16} + \cdots$$

Obtain 6th order formula from a 4nd order formula

We can repeat the argument. Since

$$M = N_2(h) - K_2\frac{h^4}{4} - K_3\frac{5h^6}{16} + \cdots$$
$$M = N_2(\frac{h}{2}) - K_2\frac{h^4}{64} - K_3\frac{5h^6}{1024} + \cdots .$$

Multiplying the second equation by 16 and subtracting it by the first equation, we have

$$16M - M = 16N_2(\frac{h}{2}) - N_2(h) - K_2(\frac{h^4}{4} - \frac{h^4}{4}) - \frac{5}{16}K_3(\frac{h^6}{4} - h^6) + \cdots$$
$$\implies 15M = 16N_2(\frac{h}{2}) - N_2(h) + \frac{15}{64}K_3h^6 + \cdots$$
$$\implies M = \frac{16}{15}N_2(\frac{h}{2}) - \frac{1}{15}N_2(\frac{h}{2}) + \frac{K_3}{64}h^6 + \cdots$$

Defining

$$N_3(h) = \frac{16}{15}N_2(\frac{h}{2}) - \frac{1}{15}N_2(\frac{h}{2}) = N_2(\frac{h}{2}) + \frac{1}{15}(N_2(\frac{h}{2}) - N_2(h)),$$

we have

$$M = N_3(h) + \frac{K_3}{64}h^6 + \cdots .$$

Thus, we have

$$|M - N_3(h)|| = O(h^6).$$

General even order case

For $N_i(h)$ satisfying

$$M - N_i(h) = K_i^{(i)} h^{2i} + K_{i+1}^{(i)} h^{2(i+1)} + \cdots ,$$

we have

$$M = N_i(h) + K_i^{(i)} h^{2i} + K_{i+1}^{(i)} h^{2(i+1)} + \cdots$$
$$M = N_i(\frac{h}{2}) + K_i^{(i)} \frac{h^{2i}}{2^{2i}} + K_{i+1}^{(i)} \frac{h^{2(i+1)}}{2^{2(i+1)}} + \cdots .$$

Thus

$$2^{2i} M - M = 2^{2i} N_i(\frac{h}{2}) - N_i(h) + 0 - \frac{3}{4} K_{i+1}^{(i)} h^{2(i+1)} + \cdots$$
$$\Longrightarrow \; M = \frac{2^{2i}}{2^{2i}-1} N_i(\frac{h}{2}) - \frac{1}{2^{2i}-1} N_i(h) - \frac{3K_{i+1}^{(i)} h^{2(i+1)}}{4(2^{2i}-1)} + \cdots$$

and $N_{i+1}(h) = N_i(\frac{h}{2}) + \frac{1}{2^{2i}-1}(N_i(\frac{h}{2}) - N_i(h))$.

Example

We consider $M = \dfrac{de^x}{dx}(0)$ and $N_1(h) = \dfrac{e^h - e^{-h}}{2h}$. Since

$$f(h) = f(0) + hf'(0) + \frac{h^2}{2}f^{(2)}(0) + \frac{h^3}{6}f^{(3)}(0) + \frac{h^4}{4!}f^{(4)}(0) + \cdots$$
$$f(-h) = f(0) - hf'(0) + \frac{h^2}{2}f^{(2)}(0) - \frac{h^3}{6}f^{(3)}(0) + \frac{h^4}{4!}f^{(4)}(0) + \cdots ,$$

we have

$$\frac{f(h) - f(-h)}{2h} = f'(0) + \frac{h^2}{6}f^{(3)}(0) + \frac{h^4}{5!}f^{(5)}(0) + \cdots .$$

Cont.

We can consider

$$N_2(h) = N_1(\frac{h}{2}) + \frac{1}{3}(N_1(\frac{h}{2}) - N_1(h)) = \frac{e^{\frac{h}{2}} - e^{-\frac{h}{2}}}{h} + \frac{1}{3}\Big(\frac{e^{\frac{h}{2}} - e^{-\frac{h}{2}}}{h} - \frac{e^{h} - e^{-h}}{h}\Big)$$

$$N_3(h) = N_2(\frac{h}{2}) + \frac{1}{15}(N_2(\frac{h}{2}) - N_2(h))$$

$$N_4(h) = N_3(\frac{h}{2}) + \frac{1}{63}(N_3(\frac{h}{2}) - N_3(h))$$

$N_1(h)$			
$N_1(\frac{h}{2})$	$N_2(h)$		
$N_1(\frac{h}{4})$	$N_2(\frac{h}{2})$	$N_3(h)$	
$N_1(\frac{h}{8})$	$N_2(\frac{h}{4})$	$N_3(\frac{h}{2})$	$N_4(h)$

Cont.

1.0269				
1.0067	$1 - 5.36 \times 10^{-5}$			
1.0017	$1 - 3.34 \times 10^{-6}$	$1 + 1.27 \times 10^{-8}$		
1.0004	$1 - 2.08 \times 10^{-7}$	$1 + 1.99 \times 10^{-10}$	$1 - 4.41 \times 10^{-13}$	

Lecture 13

Richardson's extrapolation

Review: Richardson's extrapolation

We consider $N_1(h)$ is a formula that approximates an unknown constant M with

$$M = N_1(h) + \sum_{j=1}^{m-1} K_j h^{\alpha_j} + O(h^{\alpha_m})$$

for a collection of constants $K_j \neq 0$, and
$1 \leq \alpha_1 < \alpha_2 < \cdots < \alpha_{m-1} < \alpha_m$.
Richardson's extrapolation is used to generate order α_2 formula while using order α_1 formula $N_1(h)$.

We use the formulas $N_1(h)$ and $N_1(\frac{h}{2})$ and eliminate the error term of order α_1. Namely, we have

$$2^{\alpha_1} M - M = 2^{\alpha_1} N_1(\frac{h}{2}) - N_1(h) + \sum_{j=2}^{m-1} K_j h^{\alpha_j} (\frac{2^{\alpha_1}}{2^{\alpha_j}} - 1) + O(h^{\alpha_m})$$

and $$M = N_1(\frac{h}{2}) + \frac{1}{2^{\alpha_1} - 1}\Big(N_1(\frac{h}{2}) - N_1(h)\Big) + \sum_{j=2}^{m-1} K_j h^{\alpha_j} (\frac{2^{\alpha_1 - \alpha_j} - 1}{2^{\alpha_1} - 1}) + O(h^{\alpha_m})$$

Example

We consider $M = \dfrac{de^x}{dx}(0)$ and $N_1(h) = \dfrac{e^h - e^{-h}}{2h}$. Since

$$\begin{aligned} f(h) &= f(0) + hf'(0) + \frac{h^2}{2}f^{(2)}(0) + \frac{h^3}{6}f^{(3)}(0) + \frac{h^4}{4!}f^{(4)}(0) + \cdots \\ f(-h) &= f(0) - hf'(0) + \frac{h^2}{2}f^{(2)}(0) - \frac{h^3}{6}f^{(3)}(0) + \frac{h^4}{4!}f^{(4)}(0) + \cdots, \end{aligned}$$

we have

$$\frac{f(h) - f(-h)}{2h} = f'(0) + \frac{h^2}{6}f^{(3)}(0) + \frac{h^4}{5!}f^{(5)}(0) + \cdots.$$

We can consider

$$N_2(h) = N_1(\frac{h}{2}) + \frac{1}{3}(N_1(\frac{h}{2}) - N_1(h)) = \frac{e^{\frac{h}{2}} - e^{-\frac{h}{2}}}{h} + \frac{1}{3}\Big(\frac{e^{\frac{h}{2}} - e^{-\frac{h}{2}}}{h} - \frac{e^{h} - e^{-h}}{h}\Big)$$

$$N_3(h) = N_2(\frac{h}{2}) + \frac{1}{15}(N_2(\frac{h}{2}) - N_2(h))$$

$$N_4(h) = N_3(\frac{h}{2}) + \frac{1}{63}(N_3(\frac{h}{2}) - N_3(h))$$

We have $N_4(h)$ is a formula approximate $f'(0)$ using linear combination of $f(\pm\frac{h}{8}), f(\pm\frac{h}{4}), f(\pm\frac{h}{2}), f(\pm h)$.

$N_1(h)$			
$N_1(\frac{h}{2})$	$N_2(h)$		
$N_1(\frac{h}{4})$	$N_2(\frac{h}{2})$	$N_3(h)$	
$N_1(\frac{h}{8})$	$N_2(\frac{h}{4})$	$N_3(\frac{h}{2})$	$N_4(h)$

Cont.

1.0269			
1.0067	$1 - 5.36 \times 10^{-5}$		
1.0017	$1 - 3.34 \times 10^{-6}$	$1 + 1.27 \times 10^{-8}$	
1.0004	$1 - 2.08 \times 10^{-7}$	$1 + 1.99 \times 10^{-10}$	$1 - 4.41 \times 10^{-13}$

Obtain three points endpoint formula from forward difference formula

We recall that we can approximate $f'(x_0)$ using forward difference formula

$$f'(x_0) = \frac{f(x_0 + h) - f(x_0)}{h} - \frac{h}{2} f''(\xi)$$

for $f \in C^2$.
To use Richardson's extrapolation, we need to assume $f \in C^3$ and by Taylor's theorem, we have

$$f(x_0 + h) = f(x_0) + hf'(x_0) + \frac{h^2}{2} f''(x_0) + \frac{h^3}{6} f^{(3)}(\xi)$$

and therefore

$$f'(x_0) = \frac{f(x_0 + h) - f(x_0)}{h} - \frac{h}{2} f''(x_0) - \frac{h^2}{6} f^{(3)}(\xi).$$

Obtain three points endpoint formula from forward difference formula

Using Richardson's extrapolation, we consider

$$
\begin{aligned}
f'(x_0) &= 2\frac{f(x_0+\frac{h}{2})-f(x_0)}{\frac{h}{2}} - \frac{f(x_0+h)-f(x_0)}{h} + O(h^2)\\
&= \frac{4f(x_0+\frac{h}{2})-4f(x_0)}{h} - \frac{f(x_0+h)-f(x_0)}{h} + O(h^2)\\
&= \frac{-f(x_0+h)+4f(x_0+\frac{h}{2})-3f(x_0)}{h} + O(h^2)
\end{aligned}
$$

By replacing h by $2h$, we obtain the three point endpoint formula

$$
f'(x_0) = \frac{-f(x_0+2h)+4f(x_0+h)-3f(x_0)}{2h} + O(h^2)
$$

Obtain five points midpoint formula from three points midpoint formula

We recall that we can approximate $f''(x_0)$ using three points midpoint formula

$$f''(x_0) = \frac{f(x_0+h) - f(x_0-h)}{2h} - \frac{h^2}{6} f^{(3)}(x_0) + O(h^4).$$

Using Richardson's extrapolation, we consider

$$\begin{aligned}
f'(x_0) &= \frac{f(x_0+\frac{h}{2}) - f(x_0-\frac{h}{2})}{h} + \frac{1}{3}\Big(\frac{f(x_0+\frac{h}{2}) - f(x_0-\frac{h}{2})}{h} - \frac{f(x_0+h) - f(x_0-h)}{2h}\Big) + O(h^4) \\
&= \frac{4f(x_0+\frac{h}{2}) - 4f(x_0-\frac{h}{2})}{3h} - \frac{f(x_0+h) - f(x_0-h)}{6h} + O(h^4) \\
&= \frac{-f(x_0+h) + 8f(x_0+\frac{h}{2}) - 8f(x_0-\frac{h}{2}) + f(x_0-h)}{6h} + O(h^4)
\end{aligned}$$

By replacing h by $2h$, we obtain the five points midpoint formula

$$f'(x_0) = \frac{-f(x_0+2h) + 8f(x_0+h) - 8f(x_0-h) + f(x_0-2h)}{12h} + O(h^4)$$

Obtain five points midpoint formula from second derivative formula

We recall that we can approximate $f''(x_0)$ using three points midpoint formula

$$f''(x_0) = \frac{f(x_0+h) - 2f(x_0) + f(x_0-h)}{h^2} - \frac{h^2}{12} f^{(4)}(x_0) + O(h^4).$$

Using Richardson's extrapolation, we consider

$$\begin{aligned}
f''(x_0) &= \frac{4}{3}\frac{f(x_0+\frac{h}{2}) - 2f(x_0) + f(x_0-\frac{h}{2})}{(\frac{h}{2})^2} - \frac{f(x_0+h) - 2f(x_0) + f(x_0-h)}{3h^2} + O(h^4) \\
&= \frac{16}{3}\frac{f(x_0+\frac{h}{2}) - 2f(x_0) + f(x_0-\frac{h}{2})}{h^2} - \frac{f(x_0+h) - 2f(x_0) + f(x_0-h)}{3h^2} + O(h^4) \\
&= \frac{-f(x_0+h) + 16f(x_0+\frac{h}{2}) - 30f(x_0) + 16f(x_0-\frac{h}{2}) - f(x_0-h)}{3h^2} + O(h^4)
\end{aligned}$$

By replacing h by $2h$, we obtain the five points midpoint formula

$$f'(x_0) = \frac{-f(x_0+2h) + 16f(x_0+h) - 30f(x_0) + 16f(x_0-h) - f(x_0-2h)}{12h^2} + O(h^4)$$

Remark

We remark that we cannot use the Richardson's extrapolation to obtain the higher order formula directly with the same step size h.
Assume that we use the function value of f at x_0, $x_0 \pm h$ in $N_1(h)$. We will require the function value of f at $x_0, x_0 \pm \frac{h}{2}, x_0 \pm h$ (or $x_0, x_0 \pm h, x_0 \pm 2h$) in $N_2(h)$.
However, for the next formula, we will require the function value of f at $x_0, x_0 \pm \frac{h}{4}, x_0 \pm \frac{h}{2}, x_0 \pm h$ in $N_3(h)$ which is not equally spacing.
We can check that, in fact, using Richardson's extrapolation n times, we will obtain a difference formula $N_{n+1}(h)$ which use the function value of f at

$$x_0, x_0 \pm \frac{h}{2^n}, x_0 \pm \frac{h}{2^{n-1}}, \ldots, x_0 \pm \frac{h}{2}, x_0 \pm h.$$

Lecture 15

Element of Numerical Integration

Numerical Integration

In this lecture, we are going to discuss the numerical approximation of the definite integral of a function.

A common method to approximate the integral, $\int_a^b f(x)dx$ is called numerical quadrature which uses a sum $\sum_{i=0}^n a_i f(x_i)$ to approximate $\int_a^b f(x)dx$.

One of the ways to obtain a numerical quadrature is to select a set of distinct points $\{x_0, \ldots, x_n\}$ from the interval $[a, b]$.

We then obtain the Lagrange interpolating polynomial

$$P(x) = \sum_{i=0}^{n} f(x_i)L_i(x)$$

and we can approximate the the integral, $\int_a^b f(x)dx$ by

$$\int_a^b f(x)dx \approx \int_a^b P(x) = \sum_{i=0}^{n} a_i f(x_i)$$

where $a_i = \int_a^b L_i(x)dx$.

Error bound

We recall that if $f \in C^{n+1}([a,b])$, we have

$$f(x) = P(x) + \frac{\prod_{i=0}^{n}(x - x_i) f^{(n+1)}(\xi(x))}{(n+1)!}.$$

Thus, we have

$$\int_a^b f dx = \int_a^b P(x) + \frac{1}{(n+1)!} \int_a^b \prod_{i=0}^{n} (x - x_i) f^{(n+1)}(\xi(x)) dx.$$

That is, we have

$$\int_a^b f dx \approx \sum_{i=0}^{n} a_i f(x_i)$$

with error $E(f) = \dfrac{1}{(n+1)!} \int_a^b \prod_{i=0}^{n}(x - x_i) f^{(n+1)}(\xi(x)) dx.$

Trapezoidal rule

Next, we introduce one of the most commonly used quadrature rule called trapezoidal rule.
To derive the trapezoidal rule, we consider $x_0 = a$ and $x_1 = b$. We then have

$$\begin{aligned} P(x) &= f(x_0)L_0(x) + f(x_1)L_1(x) \\ &= f(x_0)\frac{x - x_1}{x_0 - x_1} + f(x_1)\frac{x - x_0}{x_1 - x_0}. \end{aligned}$$

and

$$\begin{aligned} \int_a^b L_0(x)dx &= \frac{1}{a-b}\int_a^b (x-b)dx \\ &= \frac{1}{a-b}\frac{(x-b)^2}{2}\Big|_a^b = -\frac{(a-b)^2}{2(a-b)} = \frac{b-a}{2} \end{aligned}$$

and

$$\begin{aligned} \int_a^b L_1(x)dx &= \frac{1}{b-a}\int_a^b (x-a)dx \\ &= \frac{1}{b-a}\frac{(x-a)^2}{2}\Big|_a^b = \frac{b-a}{2} \end{aligned}$$

Error (Trapezoidal rule)

Therefore, we have

$$\int_a^b f dx \approx \frac{b-a}{2}(f(x_0) + f(x_1))$$

and the error term $E(f)$ is given as

$$E(f) = \frac{1}{2}\int_a^b (x-a)(x-b)f''(\xi(x))dx.$$

By the Weighted Mean Value Theorem for Integrals ($(x-a)(x-b) < 0$), we have

$$\int_a^b (x-a)(x-b)f''(\xi(x))dx = f''(\xi)\int_a^b (x-a)(x-b)dx$$

for some $\xi \in (a, b)$.

Thus,

$$E(f) = \frac{1}{2}f''(\xi)\int_a^b (x-a)(x-b)dx = \frac{1}{2}f''(\xi)\int_{\frac{a-b}{2}}^{\frac{b-a}{2}} (x+\frac{b-a}{2})(x-\frac{b-a}{2})dx$$

$$= \frac{1}{2}f''(\xi)\left(\frac{x^3}{3}\Big|_{\frac{a-b}{2}}^{\frac{b-a}{2}} - (\frac{b-a}{2})^2 x\Big|_{\frac{a-b}{2}}^{\frac{b-a}{2}}\right) = \frac{f''(\xi)}{16}(b-a)^3(\frac{2}{3}-2)$$

$$= -\frac{f''(\xi)(b-a)^3}{12}.$$

That is, we have

$$\int_a^b f dx = \frac{b-a}{2}(f(x_0)+f(x_1)) - \frac{f''(\xi)(b-a)^3}{12}$$

for some $\xi \in (a,b)$.

The next quadrature rule we are going to discuss is called Simpson's Rule. We consider the equally-spaced points $x_0 = a$, $x_1 = a + h$, $x_2 = b$ where $h = \dfrac{b-a}{2}$. We have

$$\begin{aligned}
\int_a^b L_0(x)dx &= \int_a^b \frac{(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)}dx \\
&= \frac{1}{2h^2}\int_{\frac{a-b}{2}}^{\frac{b-a}{2}} x(x-\frac{b-a}{2})dx \\
&= \frac{1}{2h^2}[\frac{x^3}{3} - \frac{(b-a)x^2}{4}]_{\frac{a-b}{2}}^{\frac{b-a}{2}} = \frac{1}{2h^2}\Big(\frac{(b-a)^3}{12}\Big) \\
&= \frac{b-a}{6}
\end{aligned}$$

Cont.

$$\int_a^b L_1(x)dx = \int_a^b \frac{(x-x_0)(x-x_2)}{(x_1-x_0)(x_1-x_2)}dx$$
$$= -\frac{1}{h^2}\int_{\frac{a-b}{2}}^{\frac{b-a}{2}} (x+\frac{b-a}{2})(x-\frac{b-a}{2})dx$$
$$= \frac{1}{h^2}\frac{(b-a)^3}{6}$$
$$= \frac{4(b-a)}{6}$$

Cont.

$$\int_a^b L_2(x)dx = \int_a^b \frac{(x-x_0)(x-x_1)}{(x_2-x_0)(x_2-x_1)}dx$$
$$= \frac{1}{2h^2}\int_{\frac{a-b}{2}}^{\frac{b-a}{2}} (x+\frac{b-a}{2})xdx$$
$$= \frac{1}{2h^2}\Big(\frac{(b-a)^3}{12}\Big)$$
$$= \frac{(b-a)}{6}$$

Thus, we have

$$\int_a^b fd \approx \frac{b-a}{6}(f(x_0)+4f(x_1)+f(x_2))$$

Error (Simpson's Rule)

We have

$$E(f) = \frac{1}{6}\int_a^b (x-a)(x-\frac{a+b}{2})(x-b)f^{(3)}(\xi(x))dx.$$

If we use this estimate, it seems $E(f) = O(h^4)$. In fact, we can estimate the error in another way.

By taylor theorem, we have

$$f(x) = f(x_1) + (x-x_1)f'(x_1) + \frac{(x-x_1)^2}{2}f''(x_1) + \frac{(x-x_1)^3}{6}f^{(3)}(x_1) + \frac{(x-x_1)^4}{24}f^{(4)}(\xi(x$$

and

$$\int_a^b f(x_1) = (b-a)f(x_1)$$

$$\int_a^b (x-x_1)f'(x_1)dx = 0$$

$$\int_a^b \frac{(x-x_1)^3}{6}f^{(3)}(x_1)dx = 0$$

Cont.

$$\int_a^b \frac{(x-x_1)^2}{2} f''(x_1)dx = f''(x_1)\int_a^b \frac{(x-x_1)^2}{2}dx$$
$$= \frac{f''(x_1)}{2}\frac{(b-a)^3}{12} = \frac{(b-a)^3 f''(x_1)}{24}$$
$$= \frac{(b-a)^3}{24}\left(\frac{(f(x_0)-2f(x_0)+f(x_2))}{h^2} - \frac{h^2}{12}f^{(4)}(\xi_1)\right)$$

By the Weighted Mean Value Theorem for Integrals ($(x-x_1)^4 > 0$)

$$\int_a^b \frac{(x-x_1)^4}{24} f^{(4)}(\xi(x))dx = f^{(4)}(\xi_2)\int_a^b \frac{(x-x_1)^4}{24}dx$$
$$= f^{(4)}(\xi_2)\frac{1}{120}(x-x_1)^5\Big|_a^b$$
$$= f^{(4)}(\xi_2)\frac{h^5}{60}$$

Cont.

$$
\begin{aligned}
\int_a^b f(x)dx =&(b-a)f(x_1) + \frac{(b-a)}{6}\Big(\big(f(x_0) - 2f(x_0) + f(x_2)\big)\Big) \\
&+ \left(\frac{h^5}{60}f^{(4)}(\xi_2) - \frac{h^5}{36}f^{(4)}(\xi_1)\right) \\
=&\frac{(b-a)}{6}(f(x_0) + 4f(x_1) + f(x_2)) + \frac{h^5}{12}\left(\frac{1}{5}f^{(4)}(\xi_2) - \frac{1}{3}f^{(4)}(\xi_1)\right).
\end{aligned}
$$

In fact, we can prove that the value ξ_1 and ξ_2 can be replaced by the same value $\xi \in (a, b)$. Thus, we have

$$
\int_a^b f(x)dx = \frac{(b-a)}{6}(f(x_0) + 4f(x_1) + f(x_2)) - \frac{h^5}{90}f^{(4)}(\xi).
$$

Example

We consider $a = 0, b = 1, f(x) = x^2$. We have $\int_0^1 x^2 = \dfrac{1}{3}$

$$\frac{1}{2}\Big(f(0) + f(1)\Big) = \frac{1}{2}$$

$$\frac{1}{6}\Big(f(0) + 4f(\frac{1}{2}) + f(1)\Big) = \frac{1}{6}(0 + 1 + 1) = \frac{1}{3}$$

Example

We consider $a = 0, b = 1,\ f(x) = \sin(\pi x)$. We have $\int_0^1 \sin(\pi x)dx = \dfrac{2}{\pi} \approx 0.6366$

$$\frac{1}{2}\Big(f(0) + f(1)\Big) = 0$$

$$\begin{aligned} \frac{1}{6}\Big(f(0) + 4f(\frac{1}{2}) + f(1)\Big) &= \frac{1}{6}(4\sin(\frac{\pi}{2})) = \frac{2}{3} \\ &\approx 0.6667 \end{aligned}$$

Definition

The degree of accuracy, or precision, of a quadrature formula is the largest positive integer n such that the formula is exact for x^k , for each $k = 0, 1, \ldots, n$.

For example, the trapezoidal rules have degrees of precision one and the Simpson's rules have degrees of precision three.

Lemma

The degree of precision of a quadrature formula is n if and only if the error is zero for all polynomials of degree $k = 0, 1, ..., n$, but is not zero for some polynomial of degree $n + 1$.

Closed Newton- Cotes formulas

The trapezoidal and Simpson's rules are examples of a class of methods known as Newton- Cotes formulas.
There are two types of Newton-Cotes formulas, open and closed.
The $(n+1)$-point closed Newton-Cotes formula uses nodes $x_i = x_0 + ih$, for $i = 0, 1, ..., n$, where $x_0 = a$, $x_n = b$ and $h = \dfrac{b-a}{n}$.
It is called closed because the endpoints of the closed interval [a, b] are included as nodes.

The formula has the form

$$\int_a^b f(x)dx \approx \sum_{i=0}^{n} a_i f(x_i),$$

where

$$a_i = \int_a^b L_i(x)dx = \int_{x_0}^{x_n} \prod_{j\neq i} \frac{x - x_j}{x_i - x_j}.$$

Theorem

Suppose that $\sum_{i=0}^{n} a_i f(x_i)$ *denotes the* $(n+1)$*-point closed Newton-Cotes formula with* $x_0 = a$, $x_n = b$, *and* $h = \dfrac{b-a}{n}$. *There exists* $\xi \in (a, b)$ *for which*

$$\int_a^b f(x)dx = \sum_{i=0}^{n} a_i f(x_i) + \frac{h^{n+3} f^{(n+2)}(\xi)}{(n+2)!} \int_0^n t^2(t-1)\cdots(t-n)dt,$$

if n *is even and* $f \in C^{n+2}([a, b])$, *and*

$$\int_a^b f(x)dx = \sum_{i=0}^{n} a_i f(x_i) + \frac{h^{n+2} f^{(n+1)}(\xi)}{(n+1)!} \int_0^n t(t-1)\cdots(t-n)dt,$$

if n *is odd and* $f \in C^{n+1}([a, b])$.

Example

$n = 1$: Trapezoidal rule

$$\int_a^b f(x)dx = \frac{b-a}{2}[f(x_0) + f(x_1)] - \frac{h^3}{12}f''(\xi)$$

$n = 2$: Simpson's rule

$$\int_a^b f(x)dx = \frac{b-a}{6}[f(x_0) + 4f(x_1) + f(x_2)] - \frac{h^5}{90}f^{(4)}(\xi)$$

$n = 3$: Simpson's Three-Eighths rule

$$\int_a^b f(x)dx = \frac{b-a}{8}[f(x_0) + 3f(x_1) + 3f(x_2) + f(x_3)] - \frac{3h^5}{80}f^{(4)}(\xi)$$

$n = 4$

$$\int_a^b f(x)dx = \frac{b-a}{90}[7f(x_0)+32f(x_1)+12f(x_2)+32f(x_3)+7f(x_4)] - \frac{8h^7}{945}f^{(6)}(\xi)$$

Open Newton- Cotes formulas

The open Newton-Cotes formulas do not include the endpoints of $[a, b]$. The $(n+1)$-point closed Newton-Cotes formula uses nodes $x_i = x_0 + ih$, for $i = 0, 1, ..., n$, where $x_0 = a + h$, $x_n = b - h$ and $h = \dfrac{b-a}{n+2}$.
We can label the endpoints by setting $x_{-1} = a$ and $x_{n+1} = b$.

$$\int_a^b f(x)dx = \int_{x_{-1}}^{x_{n+1}} f(x)dx \approx \sum_{i=0}^{n} a_i f(x_i),$$

where

$$a_i = \int_a^b L_i(x)dx.$$

Theorem

Suppose that $\sum_{i=0}^{n} a_i f(x_i)$ denotes the $(n+1)$-point closed Newton-Cotes formula with $x_{-1} = a$, $x_{n+1} = b$, and $h = \dfrac{b-a}{n+2}$. There exists $\xi \in (a, b)$ for which

$$\int_a^b f(x)dx = \sum_{i=0}^{n} a_i f(x_i) + \frac{h^{n+3} f^{(n+2)}(\xi)}{(n+2)!} \int_{-1}^{n+1} t^2(t-1)\cdots(t-n)dt,$$

if n is even and $f \in C^{n+2}([a, b])$, and

$$\int_a^b f(x)dx = \sum_{i=0}^{n} a_i f(x_i) + \frac{h^{n+2} f^{(n+1)}(\xi)}{(n+1)!} \int_{-1}^{n+1} t(t-1)\cdots(t-n)dt,$$

if n is odd and $f \in C^{n+1}([a, b])$.

Example

$n = 0$: Midpoint rule $x_0 = \frac{a+b}{2}$.

$$\int_a^b f(x)dx = (b-a)f(x_0) + \frac{h^3}{3}f''(\xi)$$

$n = 1$:

$$\int_a^b f(x)dx = \frac{b-a}{2}[f(x_0) + f(x_1)] - \frac{3h^3}{4}f^{(4)}(\xi)$$

$n = 2$:

$$\int_a^b f(x)dx = \frac{b-a}{3}[2f(x_0) - f(x_1) + 2f(x_2)] + \frac{14h^5}{45}f^{(4)}(\xi)$$

$n = 3$

$$\int_a^b f(x)dx = \frac{b-a}{24}[11f(x_0) + f(x_1) + f(x_2) + 11f(x_3)] - \frac{95h^5}{144}f^{(4)}(\xi)$$

Lecture 16

Composite of Numerical Integration

Composite of Numerical Integration

Similar to the numerical interpolation, using numerical integration with higher order approximation may not give us a more accurate result due to the oscillatory nature of high-degree polynomials.
Therefore, we often use a piece-wise numerical integration with low-order quadrature rule.

For example, for $f(x) = \sin(\pi x)$, we can first approximate $\int_0^{\frac{1}{2}} f(x)dx$ and $\int_{\frac{1}{2}}^1 f(x)dx$ separately and $\int_0^1 f(x)dx = \dfrac{2}{\pi} \approx 0.6366$ can be approximate by the sum of those two approximating integrals.

$$\int_0^{\frac{1}{2}} f(x)dx \approx \frac{1}{2} \cdot \frac{1}{2}\Big(\sin(\pi 0) + \sin(\frac{\pi}{2})\Big) = \frac{1}{4}$$

$$\int_{\frac{1}{2}}^1 f(x)dx \approx \frac{1}{2} \cdot \frac{1}{2}\Big(\sin(\frac{\pi}{2}) + \sin(\pi)\Big) = \frac{1}{4}$$

and

$$\int_0^1 f(x)dx = \int_0^{\frac{1}{2}} f(x)dx + \int_{\frac{1}{2}}^1 f(x)dx \approx \frac{1}{4} + \frac{1}{4} = \frac{1}{2}$$

Example

Or we can approximate $\int_0^{\frac{1}{2}} f(x)dx$ and $\int_{\frac{1}{2}}^1 f(x)dx$ by Simpson's rule.

$$\int_0^{\frac{1}{2}} f(x)dx \approx \frac{1}{2} \cdot \frac{1}{6}\Big(\sin(\pi 0) + 4\sin(\frac{\pi}{4}) + \sin(\frac{\pi}{2})\Big) = \frac{1}{12}(\frac{4}{\sqrt{2}} + 1) \approx 0.3190$$

$$\int_{\frac{1}{2}}^1 f(x)dx \approx \frac{1}{2} \cdot \frac{1}{6}\Big(\sin(\frac{\pi}{2}) + 4\sin(\frac{3\pi}{4}) + \sin(\pi)\Big) = \frac{1}{12}(\frac{4}{\sqrt{2}} + 1) \approx 0.3190$$

and

$$\int_0^1 f(x)dx = \int_0^{\frac{1}{2}} f(x)dx + \int_{\frac{1}{2}}^1 f(x)dx \approx 2 \cdot 0.3190 \approx 0.6381.$$

Example

We can further divide the interval $[0,1]$ into $[0,\frac{1}{4}]$, $[\frac{1}{4},\frac{1}{2}]$, $[\frac{1}{2},\frac{3}{4}]$ and $[\frac{3}{4},1]$ and obtain

$$\int_0^{\frac{1}{4}} f(x)dx \approx \frac{1}{4}\cdot\frac{1}{6}\Big(\sin(\pi 0) + 4\sin(\frac{\pi}{8}) + \sin(\frac{\pi}{4})\Big) \approx 0.0932$$

$$\int_{\frac{1}{4}}^{\frac{1}{2}} f(x)dx \approx \frac{1}{4}\cdot\frac{1}{6}\Big(\sin(\frac{\pi}{4}) + 4\sin(\frac{3\pi}{8}) + \sin(\frac{\pi}{2})\Big) \approx 0.2251$$

and

$$\int_0^1 f(x)dx = \int_0^{\frac{1}{4}} f(x)dx + \int_{\frac{1}{4}}^{\frac{1}{2}} f(x)dx + \int_{\frac{1}{2}}^{\frac{3}{4}} f(x)dx + \int_{\frac{3}{4}}^{1} f(x)dx \approx 0.6367.$$

Error bound

Assume we have a quadrature rule such that for any $[x, x+h] \subset [a, b]$ and $f \in C^m([a, b])$, we have

$$\int_x^{x+h} f(s)ds = h \sum_{j=0}^{J} a_j f(x + \beta_j h) + C h^{m+1} f^{(m)}(\xi(x)).$$

We consider that the interval $[a, b]$ is subdivided into n subintervals, $[x_i, x_{i+1}]$ where $x_i = a + ih$ and $h = \dfrac{b-a}{n}$.

Since $\int_{x_i}^{x_{i+1}} f(x)dx \approx h \sum_{j=0}^{J} a_j f(x_i + \beta_j h)$, we have

$$\int_a^b f(x)dx = h \sum_{i=0}^{n-1} \sum_{j=0}^{J} a_j f(x_i + \beta_j h) + \sum_{i=0}^{n-1} C h^{m+1} f^{(m)}(\xi(x_i))$$

Cont.

The error term is given by

$$E(f) = \sum_{i=0}^{n-1} C_n h^{m+1} f^{(m)}(\xi(x_i)) = Ch^{m+1} \sum_{i=0}^{n-1} f^{(m)}(\xi(x_i).$$

Since $\min_{x\in[a,b]}\{f^{(m)}(x)\} \leq f^{(m)}(\xi(x_i)) \leq \max_{x\in[a,b]}\{f^{(m)}(x)\}$, we have

$$n \min_{x\in[a,b]}\{f^{(m)}(x)\} \leq \sum_{i=0}^{n-1} f^{(m)}(\xi(x_i)) \leq n \max_{x\in[a,b]}\{f^{(m)}(x)\}$$

and

$$\sum_{i=0}^{n-1} f^{(m)}(\xi(x_i)) = nf^{(m)}(\mu)$$

for some $\mu \in (a, b)$. Since $nh = b - a$, we have

$$E(f) = C_n h^{m+1} n f^{(m)}(\mu) = C(b-a)h^m f^{(m)}(\mu).$$

Thus, we have

$$\int_a^b f(x)dx - h\sum_{i=0}^{n-1}\sum_{j=0}^{J} a_j f(x_i + \beta_j h) = C(b-a)h^m f^{(m)}(\mu)$$

Error bound for Composite Simpson's Rule

Since we have

$$\int_x^{x+h} f(x)dx = h\frac{1}{6}\Big(f(x) + 4f(x + \frac{h}{2}) + f(x+h)\Big) - \frac{1}{90}\Big(\frac{h}{2}\Big)^5 f^{(4)}(\xi),$$

we then obtain

$$\int_a^b f(x)dx - \frac{h}{6}\sum_{i=0}^{n-1}\Big(f(x_i) + 4f(x_{i+\frac{1}{2}}) + f(x_{i+1})\Big) = \frac{-1}{2^5 90}(b-a)h^4 f^{(4)}(\mu).$$

We have

$$\begin{aligned}&\frac{h}{6}\sum_{i=0}^{n-1}\Big(f(x_i) + 4f(x_{i+\frac{1}{2}}) + f(x_{i+1})\Big)\\ =&\frac{h}{6}\Big(f(a) + 4\sum_{i=0}^{n-1} f(x_{i+\frac{1}{2}}) + 2\sum_{i=1}^{n-1} f(x_i) + f(b)\Big).\end{aligned}$$

Cont.

Therefore, we have

$$\int_a^b f(x)dx - \frac{h}{6}\Big(f(a) + 4\sum_{i=0}^{n-1} f(x_{i+\frac{1}{2}}) + 2\sum_{i=1}^{n-1} f(x_i) + f(b)\Big) = -\frac{1}{2^5 90}(b-a)h^4 f^{(4)}(\mu).$$

To avoid using $x_{i+\frac{1}{2}}$, we replace h by $2h$ and obtain

$$\int_a^b f(x)dx - \frac{h}{3}\Big(f(a) + 4\sum_{i=0}^{n-1} f(x_{2i+1}) + 2\sum_{i=1}^{n-1} f(x_{2i}) + f(b)\Big) = -\frac{1}{180}(b-a)h^4 f^{(4)}(\mu)$$

where $x_i = a + ih$ for $i = 0, 1, 2, \ldots, 2n$ and $h = \dfrac{b-a}{2n}$.

Theorem

Theorem

Let $f \in C^4([a,b])$, $h = \dfrac{(b-a)}{2n}$, *and* $x_i = a + ih$ *for* $i = 0, 1, \ldots, 2n$. *There exist a* $\mu \in (a,b)$ *such that Composite Simpson's rule for* $2n$ *subintervals can be written with its error term as*

$$\int_a^b f(x)dx = \frac{h}{3}\Big(f(a) + 4\sum_{i=0}^{n-1} f(x_{2i+1}) + 2\sum_{i=1}^{n-1} f(x_{2i}) + f(b)\Big) - \frac{b-a}{180}h^4 f^{(4)}(\mu)$$

Similarly, we can obtain the following theorem for Composite Trapezoidal rule.

Theorem

Let $f \in C^2([a,b])$, $h = \dfrac{(b-a)}{n}$, *and* $x_i = a + ih$ *for* $i = 0, 1, \ldots, n$. *There exist a* $\mu \in (a,b)$ *such that Composite Trapezoidal rule for* n *subintervals can be written with its error term as*

$$\int_a^b f(x)dx = \frac{h}{2}\Big(f(a) + 2\sum_{i=1}^{n-1} f(x_i) + f(b)\Big) - \frac{b-a}{12}h^2 f^{(2)}(\mu)$$

Cont.

Also, we have the following theorem for Composite midpoint rule.

Theorem

Let $f \in C^2([a,b])$, $h = \dfrac{(b-a)}{n}$, *and* $x_i = a + ih$ *for* $i = 0, \frac{1}{2}, 1 \ldots, n - \frac{1}{2}, n$. *There exist a* $\mu \in (a,b)$ *such that Composite midpoint rule for n subintervals can be written with its error term as*

$$\int_a^b f(x)dx = h\sum_{i=0}^{n-1} f(x_{i+\frac{1}{2}}) + \frac{(b-a)}{24}h^2 f''(\mu)$$

Lecture 17

Romberg Integration

Romberg extrapolation

We discussed before on using Richardson extrapolation to obtain a higher order accuracy deference formula.
In fact, we can use the same idea to obtain a higher order accuracy quadrature rule to estimate an definite integral.
We consider that $N_1(h)$ is a formula which approximate a unknown constant M with

$$M = N_1(h) + \sum_{j=1}^{m-1} K_j h^{\alpha_j} + O(h^{\alpha_m})$$

for a collection of constants $K_j \neq 0$, and $1 \leq \alpha_1 < \alpha_2 < \cdots < \alpha_{m-1} < \alpha_m$.
We recall that we can use richardson extrapolation to obtain $N_2(h)$ with

$$M = N_2(h) + \sum_{j=2}^{m-1} \tilde{K}_j h^{\alpha_j} + O(h^{\alpha_m})$$

Cont.

For example, we showed that the composite trapezoidal rule has a truncation error of order $O(h^2)$ where $h = \dfrac{b-a}{n}$. That is, we have

$$\int_a^b f(x)dx = \frac{h}{2}[f(a) + 2\sum_{j=1}^{n-1} f(x_j) + f(b)] - \frac{(b-a)f''(\mu)}{12}h^2.$$

In fact, for $f \in C^{2m+2}([a,b])$, we can show that

$$\int_a^b f(x)dx = \frac{h}{2}[f(a)+2\sum_{j=1}^{n-1} f(x_j)+f(b)]+\sum_{j=1}^{m} K_j h^{2j}+\tilde{K}_{m+1} f^{(2m+2)}(\mu)h^{2m+2}$$

where K_j is constants that depends only on $f^{(2j-1)}(a)$ and $f^{(2j-1)}(b)$.

We have

$$N_{j+1}(h) = N_j(\frac{h}{2}) + \frac{1}{2^{\alpha_j} - 1}(N_j(\frac{h}{2}) - N_j(h)).$$

For $N_1(h) = (b-a)\dfrac{f(a)+f(a+h)}{2}$, $N_1(\frac{h}{2}) = \dfrac{b-a}{2}\dfrac{f(a)+2f(a+\frac{h}{2})+f(b)}{2}$,we have

$$\begin{aligned}
N_2(h) &= (b-a)\frac{f(a)+2f(a+\frac{h}{2})+f(b)}{4} + \frac{b-a}{3}(\frac{f(a)+2f(a+\frac{h}{2})+f(b)}{4} - \frac{f(a)+f(b)}{2}) \\
&= \frac{b-a}{6}\Big(\frac{3f(a)+6f(a+\frac{h}{2})+3f(b)}{2} + \frac{-f(a)+2f(a+\frac{h}{2})-f(b)}{2}\Big) \\
&= \frac{b-a}{6}\Big(f(a)+4f(a+\frac{h}{2})+f(b)\Big).
\end{aligned}$$

where $\int_a^b f(x)dx = N_2(h) + O(h^4)$.

Higher order approximation

We denote $R_{i,1} = N_1(\frac{h}{2^{i-1}})$. We then define $R_{i,j}$ recursively as

$$R_{i,j+1} = R_{i,j} + \frac{1}{2^{2j}-1}(R_{i,j} - R_{i-1,j}).$$

Thus, we have $\int_a^b f(x)dx = R_{i,i} + O(h^{2j})$.

Example

We consider $f = \sin(\pi x)$ in $[0, 1]$. We have

$$R_{1,1} = \frac{1}{2}(\sin(0) + \sin(\pi)) = 0,\ R_{2,1} = \frac{1}{4}(\sin(0) + 2\sin(\frac{\pi}{2}) + \sin(\pi)) = \frac{1}{2},$$

$$R_{3,1} = \frac{1}{8}(\sin(0) + 2(\sin(\frac{\pi}{4}) + \sin(\frac{\pi}{2}) + \sin(\frac{3\pi}{4})) + \sin(\pi)) \approx 0.6036$$

$$R_{4,1} = \frac{1}{16}(\sin(0) + 2(\sin(\frac{\pi}{8}) + \sin(\frac{\pi}{4}) + \cdots + \sin(\frac{7\pi}{8})) + \sin(\pi)) \approx 0.6284$$

$$R_{2,2} = R_{2,1} + \frac{1}{3}(R_{2,1} - R_{1,1}) \approx 0.6667$$

$$R_{3,2} = R_{3,1} + \frac{1}{3}(R_{3,1} - R_{2,1}) \approx 0.6381$$

$$R_{4,2} = R_{4,1} + \frac{1}{3}(R_{4,1} - R_{3,1}) \approx 0.6367$$

Cont.

$$R_{3,3} = R_{3,2} + \frac{1}{15}(R_{3,2} - R_{2,2}) \approx 0.6362$$
$$R_{4,3} = R_{4,2} + \frac{1}{15}(R_{4,2} - R_{3,2}) \approx 0.6366$$

and

$$R_{4,4} = R_{4,3} + \frac{1}{63}(R_{4,3} - R_{3,3}) \approx 0.6366$$

$$R_{1,1} - \frac{2}{\pi} = \frac{2}{\pi},\; R_{2,2} - \frac{2}{\pi} \approx 0.030$$
$$R_{3,3} - \frac{2}{\pi} \approx -4.55 \times 10^{-4},\; R_{4,4} - \frac{2}{\pi} \approx 1.77 \times 10^{-6}$$

Example

We consider $f = e^x$ in $[-1, 1]$. We have

	$O(h^2)$	$O(h^4)$	$O(h^6)$	$O(h^8)$
1	$R_{1,1}$			
2	$R_{2,1}$	$R_{2,2}$		
3	$R_{3,1}$	$R_{3,2}$	$R_{3,3}$	
4	$R_{4,1}$	$R_{4,2}$	$R_{4,3}$	$R_{4,4}$

Example

We consider $f = e^x$ in $[-1, 1]$. We have

	$O(h^2)$	$O(h^4)$	$O(h^6)$	$O(h^8)$
1	3.0862			
2	2.5431	2.3621		
3	2.3992	2.3512	2.3505	
4	2.3626	2.3505	2.3504	2.3504

Algorithm

Input:	function f, two endpoints a, b, and number iteration n
Output:	the number $R_{1,1}, R_{2,2}, \ldots, R_{n,n}$
Step 1:	Set $h = b - a$, $R_{1,1} = \frac{h}{2}(f(a) + f(b))$, $tR_{1,1} = R_{1,1}$
Step 2:	For $i = 2, \ldots, n$, $tR_{2,1} = \frac{h}{2^i}(f(a) + 2\sum_{k=1}^{2^{i-1}-1} f(a + k\frac{h}{2^{i-1}}) + f(b))$ For $j = 2, \ldots, i$, Set $tR_{2,j} = tR_{2,j-1} + \frac{tR_{2,j-1} - tR_{1,j-1}}{4^{j-1} - 1}$. Set $R_{i,i} = tR_{2,i}$ Set $tR_{1,k} = tR_{2,k}$ for $k = 1, \ldots, i$
Step 3:	OUTPUT $(R_{1,1}, R_{2,2}, \ldots, R_{n,n})$

Lecture 18

Adaptive Quadrature Methods

Adaptive Quadrature Methods

The composite quadrature rule can help us to obtain an effective approximation of the integral $\int_a^b f(x)dx$.
We consider a quadrature rule such that for any $[x, x+h] \subset [a, b]$ and $f \in C^m([a, b])$

$$\int_x^{x+h} f(s)ds = h \sum_{j=0}^{J} a_j f(x + \beta_j h) + Ch^{m+1} f^{(m)}(\xi(x)).$$

In the previous discussion, we always assume $\{x_i\}_{i=0}^n$ is equality spaced. In fact, if we consider $\{x_i\}_{i=0}^n$ is not equality spaced and $h_i = x_{i+1} - x_i$, we can obtain

$$\int_a^b f(x)dx = \sum_{i=0}^{n-1} h_i \sum_{j=0}^{J} a_j f(x + \beta_j h) + C \sum_{i=0}^{n-1} h_i^{m+1} f^{(m)}(\xi(x_i)).$$

Cont.

We can easily check that the error term $E(f)$ is the sum of the error in each sub-interval $[x_i, x_{i+1}]$ that is

$$E(f) = C \sum_{i=0}^{n-1} h_i^{m+1} f^{(m)}(\xi(x_i)).$$

Therefore, if the magnitude of $f^{(m)}(x)$ has a huge difference in different sub-interval $[x_i, x_{i+1}]$, it is reasonable to use a smaller h_i in the region with larger magnitude of $f^{(m)}(x)$.

Example

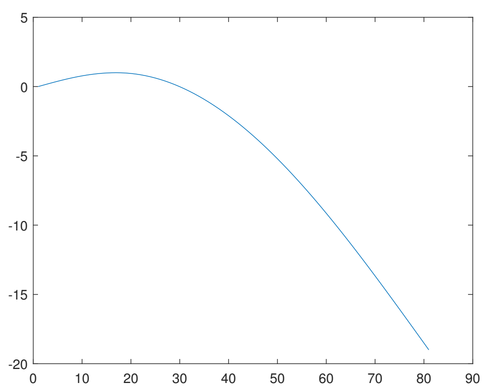
For

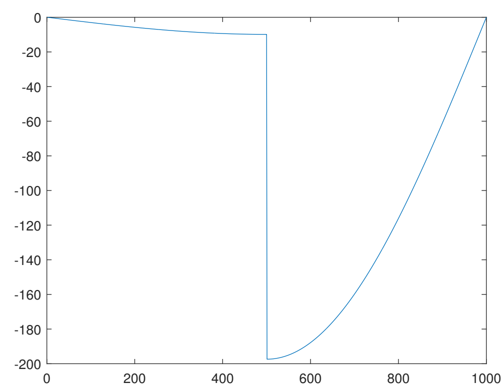
$$f = \begin{cases} \sin(\pi x) & x < 0.5 \\ 20\sin(\pi x) - 19 & x \geq 0.5 \end{cases}$$

in [0,1], we have

$$f'' = \begin{cases} -\pi^2 \sin(\pi x) & x < 0.5 \\ -20\pi^2 \sin(\pi x) & x \geq 0.5 \end{cases}.$$

The following are the plot for f and f''





Cont.

If we consider $\{x_i\}_{i=0}^{n}$ is equality spaced, we have

$$\int_a^b f dx = \frac{21}{\pi} - \frac{19}{2} \approx -2.8155$$

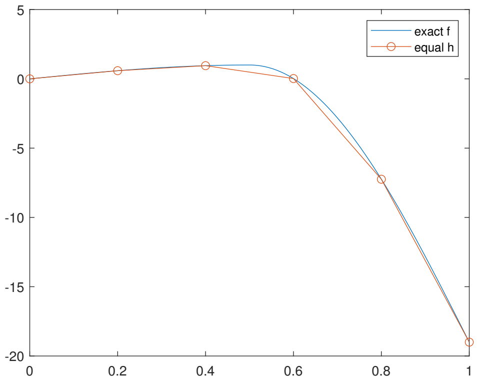
n	5	10	20	40	80
error	0.2214	0.0551	0.0138	0.0034	0.0009

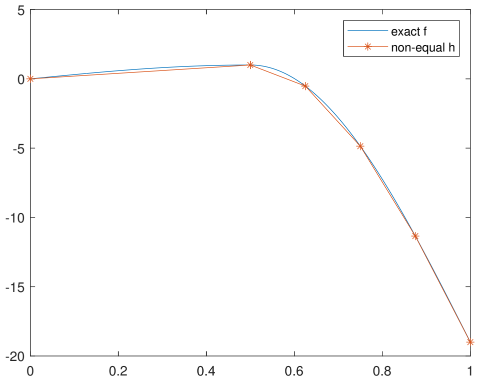
However, if we consider with

$$h_i = \begin{cases} \dfrac{5}{2n} & \text{if } i < n/5 \\ \dfrac{5}{8n} & \text{if } i \geq n/5 \end{cases}.$$

n	5	10	20	40	80
error	0.1503	0.0370	0.0092	0.0023	0.0006

For $n = 5$,


exact f
equal h
5
0
-5
-10
-15
-20
0
0.2
0.4
0.6
0.8
1


exact f
non-equal h
5
0
-5
-10
-15
-20
0
0.2
0.4
0.6
0.8
1

Estimate $f^{(m)}$

The problem is that in general, we do not have any information of $f^{(m)}$.
Can we estimate $f^{(m)}$ without using the explicit formula $f^{(m)}$?
The answer is Yes.
We consider $S(a,b) = \dfrac{b-a}{6}(f(a) + 4f(a+h) + f(b))$ where $h = \dfrac{b-a}{2}$.
We have

$$\int_a^b f(x) = S(a,b) - \frac{h^5}{90} f^{(4)}(\xi).$$

Cont.

We can use Composite Simpson's rule to approximate $\int_a^b f dx$ where

$$\int_a^b f dx = S(a, \frac{a+b}{2}) + S(\frac{a+b}{2}, b) - \frac{1}{16}\frac{h^5}{90} f^{(4)}(\tilde{\xi})$$

Thus, we have

$$S(a, \frac{a+b}{2}) + S(\frac{a+b}{2}, b) - \frac{1}{16}\frac{h^5}{90} f^{(4)}(\tilde{\xi}) = \int_a^b f dx \approx S(a,b) - \frac{h^5}{90} f^{(4)}(\tilde{\xi})$$

and

$$\frac{h^5}{90} f^{(4)}(\tilde{\xi}) \approx \frac{16}{15}\Big(S(a,b) - S(a, \frac{a+b}{2}) - S(\frac{a+b}{2}, b)\Big)$$

Cont.

Therefore, we have

$$\Big|\int_a^b f dx - S(a, \frac{a+b}{2}) - S(\frac{a+b}{2}, b)\Big| = \frac{1}{16}\frac{h^5}{90}\Big|f^{(4)}(\tilde{\xi})\Big|$$
$$\approx \frac{1}{15}\Big|S(a,b) - S(a, \frac{a+b}{2}) - S(\frac{a+b}{2}, b)\Big|.$$

If we have

$$\Big|S(a,b) - S(a, \frac{a+b}{2}) - S(\frac{a+b}{2}, b)\Big| < 15\epsilon$$

where ϵ is a specified tolerance, we have

$$\Big|\int_a^b f dx - S(a, \frac{a+b}{2}) - S(\frac{a+b}{2}, b)\Big| \lesssim \epsilon.$$

Cont.

We can continue the process iteratively.

If $\left|S(a,b) - S(a,\frac{a+b}{2}) - S(\frac{a+b}{2},b)\right| \geq \epsilon$, we can sub-divide the interval $[a,\frac{a+b}{2}]$ and $[\frac{a+b}{2},b]$. Namely, we consider approximate $\int_a^b f dx$ by

$$S(a,a+\frac{b-a}{4})+S(a+\frac{b-a}{4},a+\frac{b-a}{2})+S(a+\frac{b-a}{2},a+\frac{3(b-a)}{4})+S(\frac{3(b-a)}{4},b).$$

If

$$\left|S(a,\frac{b+a}{2}) - S(a,a+\frac{b-a}{4}) - S(a+\frac{b-a}{4},a+\frac{b-a}{2})\right| < \frac{\epsilon}{2}$$

and

$$\left|S(\frac{a+b}{2},b) - S(a+\frac{b-a}{2},a+\frac{3(b-a)}{4}) - S(\frac{3(b-a)}{4},b)\right| < \frac{\epsilon}{2}.$$

We will have

$$\left|\int_a^b f dx - \left(S(a,a+\frac{b-a}{4})+S(a+\frac{b-a}{4},a+\frac{b-a}{2})+S(a+\frac{b-a}{2},a+\frac{3(b-a)}{4})+S(\frac{3(b-a)}{4},b)\right)\right| \lesssim \epsilon.$$

Lecture 19

Gaussian Quadrature

Quadrature rule with high order accuracy

In the previous lecture, we discussed on how to improve the accuracy by using adaptive method to choose a non-uniform step size h which gives non-equally-spaced quadrature points in the entire interval $[a, b]$.
Thus, we can use less quadrature points (degree of freedoms) to give a good approximation of $\int_a^b f dx$.
However, in each of the sub-interval $[x_i, x_{i+1}]$, we are still using equally-spaced quadrature points. (Newton-Cotes quadrature, or Romberg Integration)
In fact, these equally-spaced quadrature rules generally do not provide the optimum accuracy for a fixed number of quadrature points.

Review: Newton-Cotes quadrature error

We recall that the error term $E(f) = \int_a^b f dx - \sum_{i=1}^n a_i f(x_i)$ for open (closed) Newton-Cotes quadrature with n quadrature points is bounded by

$$|E(f)| \leq \begin{cases} (b-a)^{n+2} C_n \max_{x\in[a,b]} |f^{(n+1)}(x)| & \text{if } n \text{ is odd} \\ (b-a)^{n+1} C_n \max_{x\in[a,b]} |f^{(n)}(x)| & \text{if } n \text{ is even} \end{cases}$$

if f is sufficiently smooth and C_n is a constant define on n.

Example

For example, we would like to find two quadrature points $\{x_1, x_2\}$ such that

$$\int_{-1}^{1} f dx = \omega_1 f(x_1) + \omega_2 f(x_2)$$

for all f which is a polynomial of degree at most 3.
It is easy to see that if

$$\int_{-1}^{1} x^k = \omega_1 x_1^k + \omega_2 x_2^k, \quad \text{for } k = 0, 1, 2, 3,$$

then we have

$$\int_{-1}^{1} f = \omega_1 f(x_1) + \omega_2 f(x_2)$$

for all f which is a polynomial of degree at most 3. (since $I(f) := \int_a^b f dx$ is a linear operator)

Cont.

To find $x_1, x_2, \omega_1, \omega_2$, we need to solve the following four nonlinear equations

$$\omega_1 + \omega_2 = \int_{-1}^{1} 1 = 2$$
$$\omega_1 x_1 + \omega_2 x_2 = \int_{-1}^{1} x = 0$$
$$\omega_1 x_1^2 + \omega_2 x_2^2 = \int_{-1}^{1} x^2 = \frac{2}{3}$$
$$\omega_1 x_1^3 + \omega_2 x_2^3 = \int_{-1}^{1} x^3 = 0$$

By multiplying the second equation by x_1^2 and subtracting it by the four equation, we have

$$\omega_2 x_2 (x_1^2 - x_2^2) = 0$$

and thus we have $x_1 = -x_2$ ($\omega_2 x_2 = 0$ and $x_1 = x_2$ are rejected).

Cont.

Thus, we have

$$\omega_1 + \omega_2 = \int_{-1}^{1} 1 = 2$$

$$(\omega_1 - \omega_2)x_1 = \int_{-1}^{1} x = 0$$

$$(\omega_1 + \omega_2)x_1^2 = \int_{-1}^{1} x^2 = \frac{2}{3}$$

Thus, we have

$$x_1^2 = \frac{1}{3} \implies x_1 = \frac{-1}{\sqrt{3}} \text{ and } x_2 = \frac{1}{\sqrt{3}}.$$

and

$$\omega_1 = \omega_2 = 1$$

Cont.

Thus, we have

$$\int_{-1}^{1} f dx = f(\frac{-1}{\sqrt{3}}) + f(\frac{1}{\sqrt{3}})$$

for all polynomial f of degree at most 3. (This quadrature rule has degree of precision 3)
Therefore, we have

$$\int_{a}^{b} f dx = (\frac{b-a}{2})\Big(f(\frac{b+a}{2} - \frac{b-a}{2\sqrt{3}}) + f(\frac{b+a}{2} + \frac{b-a}{2\sqrt{3}})\Big)$$

for all polynomial f of degree at most 3.

More points

In general, for a quadrature rule with n quadrature points, we have

$$\int_a^{a+h} f dx \approx \sum_{i=1}^{n} a_i f(a + \beta_i h).$$

We have $2n$ unknowns in $\sum_{i=1}^{n} a_i f(a + \beta_i h)$ which are a_i and β_i. Thus, it is natural to have a set of parameters $\{a_i, \beta_i\}_{i=1}^{n}$ such that

$$\int_a^{a+h} f dx = \sum_{i=1}^{n} a_i f(a + \beta_i h)$$

for all polynomial f of degree at most $2n - 1$. ($2n$ equations and $2n$ unknowns)
The question is how to find those quadrature points and quadrature weights.

Before discussing how to find the quadrature points and quadrature weights, we will first introduce a special type of polynomial called "Legendre Polynomial". The Legendre Polynomials is a set polynomials $\{P_0, P_1, \ldots, P_n, \ldots\}$ satisfying the following two properties:
1.) For each n, P_n is a monic polynomial of degree n (that is $P_n(x) = x^n + b_{n-1}x^{n-1} + \cdots + b_1 x + b_0$)
2.) $\int_{-1}^{1} P(x)P_n(x)dx = 0$ for all P is a polynomial of degree at most $n-1$.

Example

The first few Legendre polynomials are

$$P_0(x) = 1,\ P_1(x) = x,\ P_2(x) = x^2 - \frac{1}{2}$$

$$P_3(x) = x^3 - \frac{3}{5}x,\ P_4(x) = x^4 - \frac{6}{7}x^2 + \frac{3}{35}.$$

The Legendre polynomial P_n is a odd function if n is odd and P_n is a even function if n is even, that is,

$$P_n(x) = \begin{cases} -P_n(-x) & \text{if } n \text{ is odd} \\ P_n(-x) & \text{if } n \text{ is even.} \end{cases}$$

Properties for Legendre Polynomials

It is easy to check that, for any polynomial P of degree at most n, P can be written as a linear combination of the Legendre Polynomials $\{P_0, P_1, \ldots, P_n\}$. Namely,

$$P(x) = \sum_{i=1}^{n} a_i P_i(x).$$

We also have $\{P_0, P_1, \ldots, P_n, \ldots\}$ is a set of orthogonal polynomials in $L^2([-1,1])$ inner product. Namely,

$$\int_{-1}^{1} P_i P_j = 0 \ \forall i \neq j.$$

Moreover, we have the roots of a Legendre polynomial P_n are **distinct**, **lie in the interval** $(-1,1)$, are **symmetry** with respect to the origin.

Guassian Quadrature

The roots of a Legendre polynomial P_n are the quadrature points which provide an integral approximation formula that gives exact results for any polynomial of degree less than $2n$.

Theorem

Suppose that $x_1, x_2, \cdots x_n$ are the n-th Legendre polynomial P_n and that for $i = 1, 2, \ldots, n$ the number c_i are defined by

$$c_i = \int_{-1}^{1} \prod_{j=1, i\neq j}^{n} \frac{x - x_j}{x_i - x_j} dx.$$

If $P(x)$ is any polynomial of degree at most $2n-1$, then

$$\int_{-1}^{1} P(x)dx = \sum_{i=1}^{n} c_i P(x_i)$$

Proof

We first consider P is a polynomial of degree at most $n-1$. By the uniqueness of Lagrange polynomial, we have

$$P(x) = \sum_{j=1}^{n} P(x_i) L_i(x)$$

where $L_i(x) = \prod_{j=1, i \neq j}^{n} \dfrac{x - x_j}{x_i - x_j}$. Thus, we have

$$\begin{aligned} \int_{-1}^{1} P(x) dx &= \int_{-1}^{1} \sum_{j=1}^{n} P(x_i) L_i(x) dx \\ &= \sum_{i=1}^{n} c_i P(x_i). \end{aligned}$$

Proof

We consider P is a polynomial of degree at least n. We divide P by the n-th Legendre polynomial P_n to obtain

4|7-1|

$$\overset{2n-3}{P(x)} = \overset{n-3}{Q(x)}\overset{n}{P_n}(x) + R(x)$$

where Q is the quotient and R is the reminder polynomial of degree at most $n-1$.

For P is a polynomial of degree at most $2n-1$, we have Q is a polynomial of degree at most $n-1$.

Since x_i is the roots of P_n, we have $P_n(x_i) = 0$ and thus,

$$\boxed{P(x_i) = Q(x_i)P_n(x_i) + R(x_i) = R(x_i).}$$

Proof

Since Q is a polynomial of degree at most $n-1$, we have

$$\int_{-1}^{1} Q(x)P_n(x)dx = 0.$$

Therefore, we have

$$\begin{aligned}\int_{-1}^{1} P(x)dx = \int_{-1}^{1} \Big(Q(x)P_n(x) + R(x)\Big)dx = \int_{-1}^{1} R(x)dx \\ = \sum_{i=1}^{n} c_i R(x_i) = \sum_{i=1}^{n} c_i P(x_i).\end{aligned}$$

Lecture 20

Gaussian Quadrature

Review

The Legendre Polynomials is a set polynomials $\{P_0, P_1, \ldots, P_n, \ldots\}$ satisfying the following two properties:
1.) For each n, P_n is a monic polynomial of degree n (that is $P_n(x) = x^n + b_{n-1}x^{n-1} + \cdots + b_1 x + b_0$)
2.) $\int_{-1}^{1} P(x)P_n(x)dx = 0$ for all P is a polynomial of degree at most $n-1$.

Theorem

Suppose that $x_1, x_2, \cdots x_n$ are the n-th Legendre polynomial P_n and that for $i = 1, 2, \ldots, n$ the number c_i are defined by

$$c_i = \int_{-1}^{1} \prod_{j=1, i\neq j}^{n} \frac{x - x_j}{x_i - x_j} dx.$$

If $P(x)$ is any polynomial of degree at most $2n-1$, then

$$\int_{-1}^{1} P(x)dx = \sum_{i=1}^{n} c_i P(x_i)$$

Gaussian Quadrature points and weights

n	$\{x_i\}_{i=1}^n$	$\{c_i\}_{i=1}^n$
1	$\{0\}$	$\{2\}$
2	$\{-\frac{1}{\sqrt{3}}, \frac{1}{\sqrt{3}}\}$	$\{1, 1\}$
3	$\{-\sqrt{\frac{3}{5}}, 0, \sqrt{\frac{3}{5}}\}$	$\{\frac{5}{9}, \frac{8}{9}\frac{5}{9}\}$
4	$\{-\sqrt{\frac{3}{7}+\frac{2}{7}\sqrt{\frac{6}{5}}}, -\sqrt{\frac{3}{7}-\frac{2}{7}\sqrt{\frac{6}{5}}}, \sqrt{\frac{3}{7}-\frac{2}{7}\sqrt{\frac{6}{5}}}, \sqrt{\frac{3}{7}+\frac{2}{7}\sqrt{\frac{6}{5}}}\}$	$\{\frac{18-\sqrt{30}}{36}, \frac{18+\sqrt{30}}{36}, \frac{18+\sqrt{30}}{36}, \frac{18-\sqrt{30}}{36}\}$
5	$\{-\frac{1}{3}\sqrt{5+2\sqrt{\frac{10}{7}}}, -\frac{1}{3}\sqrt{5-2\sqrt{\frac{10}{7}}}, 0, \frac{1}{3}\sqrt{5-2\sqrt{\frac{10}{7}}}, \frac{1}{3}\sqrt{5+2\sqrt{\frac{10}{7}}}\}$	$\{\frac{322-13\sqrt{70}}{900}, \frac{322+13\sqrt{70}}{900}, \frac{128}{225}, \frac{322+13\sqrt{70}}{900}, \frac{322-13\sqrt{70}}{900}\}$

Example

We consider $f(x) = \sin(\pi x)$ and have $\int_0^1 \sin(\pi x) = \dfrac{2}{\pi} \approx 0.6366$. For $n = 2$, using Gaussian Quadrature rule, we have

$$\begin{aligned}\int_0^1 \sin(\pi x) &\approx \frac{1}{2}f(\frac{1}{2} - \frac{1}{2}\sqrt{\frac{1}{3}}) + \frac{1}{2}f(\frac{1}{2} + \frac{1}{2}\sqrt{\frac{1}{3}}) \\ &\approx 0.6162\end{aligned}$$

Using Trapezoidal rule, we obtain $\int_0^1 \sin(\pi x) \approx \dfrac{1}{2}\sin(0 \cdot \pi) + \dfrac{1}{2}\sin(\pi) = 0$

Cont.

For $n = 3$, using Gaussian Quadrature rule, we have

$$\int_0^1 \sin(\pi x) \approx \frac{5}{18} f\left(\frac{1}{2} - \frac{1}{2}\sqrt{\frac{3}{5}}\right) + \frac{4}{9} f\left(\frac{1}{2}\right) + \frac{5}{18} f\left(\frac{1}{2} + \frac{1}{2}\sqrt{\frac{3}{5}}\right)$$
$$\approx 0.6371$$

Using Simpson's Rule, we obtain

$$\int_0^1 \sin(\pi x) \approx \frac{1}{6}\sin(0 \cdot \pi) + \frac{4}{6}\sin\left(\frac{\pi}{2}\right) + \frac{1}{6}\sin(\pi) \approx 0.6667$$

Other quadrature rule

There are many other quadrature rules. For example, Gauss-Laguerre quadrature, Gauss-Hermite quadrature, Gauss–Kronrod quadrature, Gauss-Lobatto quadrature, Gauss-Radau Quadrature.
For Gauss-Laguerre quadrature rules, we consider the approximation of an integral in a semi-infinite domain in form of

$$\int_0^\infty e^{-x} f(x) dx.$$

The quadrature points for Gauss-Laguerre quadrature rules are the roots of the Leguerre polynomials which are orthogonal with respect to the inner product

$$(f, g) = \int_0^\infty e^{-x} f(x) g(x) dx.$$

Cont.

For Gauss-Hermite quadrature rules, we consider the approximation of an integral in a infinite domain in form of

$$\int_{-\infty}^{\infty} e^{-x^2} f(x) dx.$$

The quadrature points for Gauss-Hermite quadrature rules are the roots of the polynomials which are orthogonal with respect to the inner product

$$(f, g) = \int_{-\infty}^{\infty} e^{-x^2} f(x) g(x) dx.$$

Gauss-Lobatto quadrature

For composite gauss quadrature rule, in fact, we are approximating the function f by a discontinuous polynomial (piecewise polynomial but probably discontinuous at the endpoint of the sub-intervals.)
In some applications, we may want to have a continuous approximation of f and also a good approximation of the integral of f at the same time.
Therefore, there is a quadrature rule called Gauss-Lobatto quadrature rule which uses two endpoints of the interval and some additional quadrature points to obtain an approximation of $\int_{-1}^{1} f(x)$.
The additional quadrature points are the roots of the Jocobi polynomials which are orthogonal with respect to the inner product

$$(f, g) = \int_{-1}^{1} (x-1)(x+1) f(x) g(x) dx.$$

Cont.

We can show that for using Gauss-Lobatto quadrature rule, we will obtain an approximation with $2n-3$ degree of precision and

n	$\{x_i\}_{i=1}^n$	$\{c_i\}_{i=1}^n$
2	$\{-1, 1\}$	$\{1, 1\}$
3	$\{-1, 0, 1\}$	$\{\frac{1}{3}, \frac{4}{3}, \frac{1}{3}\}$
4	$\{-1, -\sqrt{\frac{1}{5}}, \sqrt{\frac{1}{5}}, 1\}$	$\{\frac{1}{6}, \frac{5}{6}\frac{5}{6}, \frac{1}{6}\}$
5	$\{-1, -\sqrt{\frac{3}{7}}, 0, \sqrt{\frac{3}{7}}, 1\}$	$\{\frac{1}{10}, \frac{49}{90}, \frac{32}{45}, \frac{49}{90}, \frac{1}{10}\}$

Lecture 21

Discrete Least Squares Approximation

Interpolation vs regression

In real application, we often handle the data with errors.
We would like to approximate the function f. We consider $\{x_i\}$is a set of points and $\tilde{y}_i = f(x_i)$.
During the measuring process, we may not obtain the exact measurement $\tilde{y}_i$. We consider the measurement data $\{y_i\}$ is written in form of

$$y_i = \tilde{y}_i + \epsilon_i.$$

When we are interpolating the function with large amount of data, the approximating error will magnified by the measurement error.
The magnifying factor is highly depend on the degree of the interpolating polynomial. That is, assuming P is the interpolating polynomial, we have

$$\max_{x\in[a,b]} |f(x) - P_n(x)| \leq C(n) \max_i |\epsilon_i|$$

where n is the degree of the polynomial P_n and $C(n)$ is a constant depend on n.

Cont.

How about using lower degree piece-wise polynomial?
Using piece-wise polynomial can improve the stability of the interpolation with respect to the measurement error.
However, using piece-wise approximation will lose the global expression power of the approximation model.
It also try to capture the wrong feature of the data which provides a locally incorrect model near the "outlier".
Therefore, sometimes, we may want to obtain a globally smooth approximation for the function f.
For example, we would like to approximate the function f by a linear function

$$f(x) \approx a_0 + a_1 x, \ \forall x \in [a, b].$$

Minimizing the mismatch

Then the question is how to determine the coefficient in our approximation function.
For using interpolation, we are fitting all of data point.
However, for using regression, we normal have the number of data points more than the number of the unknown coefficient.
For example, we consider that

$$\{x_i\}_{i=1}^{n} = \{1, 2, 3, 4, 5, 6, 7, 8, 9\}$$

$$\{y_i\}_{i=1}^{n} = \{1.2, 2.1, 2.9, 4.0, 5.1, 6.0, 6.8, 7.8, 9.0\}$$

If we want to approximate the function by linear function $a_1 x + a_0$, we have 9 data points but only 2 unknowns a_0 and a_1.

Cont.

One of the most common way to determine the coefficient is to find $\{a_1, a_0\}$ which minimize a "loss function" E.
That is, a_1 and a_0 are some constants such that

$$E(a_1, a_0) = \min_{c,d\in\mathbb{R}} \{E(c, d)\}.$$

There are many possible choice for E, for example

$$E_\infty(a_1, a_0) = \max_{i=1,\dots,n} |y_i - a_1 x_i - a_0|$$

$$E_1(a_1, a_0) = \sum_{i=1}^{n} |y_i - a_1 x_i - a_0|$$

$$E_2(a_1, a_0) = \sum_{i=1}^{n} |y_i - a_1 x_i - a_0|^2$$

Cont.

For using $E_\infty(a_1, a_0) = \max_{i=1,\dots,n} |y_i - a_1 x_i - a_0|$, we are going to find a linear function which fits the data with minimum $l_\infty-$norm.
That is,we want the maximum mismatch of the data is as small as possible.
However, it is generally difficult to find the $\{a_1, a_0\}$ which

$$E_\infty(a_1, a_0) = \min_{c,d\in\mathbb{R}} \{E_\infty(c, d)\}.$$

Another possible choice of E is $E_1(a_1, a_0) = \sum_{i=1}^n |y_i - a_1 x_i - a_0|$. In this case, we are going to minimum l_1 mismatch of the data.
To solve this problem, we can try to find $\{a_0, a_1\}$ such that

$$\frac{\partial}{\partial a_0} \sum_{i=1}^n |y_i - a_1 x_i - a_0| = 0 = \frac{\partial}{\partial a_1} \sum_{i=1}^n |y_i - a_1 x_i - a_0|.$$

However, $|\cdot|$ is not differentiable at zero. Therefore, sometimes, it is not easy to find a solution of these two equations.

Discrete Least Squares Approximation

The most commonly used "loss function" is the summation of the the l_2-error that is

$$E_2(a_1, a_0) = \sum_{i=1}^{n} |y_i - a_1 x_i - a_0|^2.$$

We easily see that E_2 can be written in form of $E_2(a_1, a_0) = (\vec{y} - X\vec{a}, \vec{y} - X\vec{a})_{l_2}$ where $\vec{y} = (y_1, \ldots, y_n)^t$, $\vec{a} = (a_1, a_0)^t$,

$$X = \begin{pmatrix} x_1 & x_2 & \cdots & x_n \\ 1 & 1 & \cdots & 1 \end{pmatrix}^t$$

and $(\cdot, \cdot)_{l_2}$ is the discrete l_2 inner product, that is

$$(\vec{x}, \vec{y}) = \sum_i x_i y_i$$

Cont.

To find a_1, a_0 such that $E_2(a_1, a_0) = \max_{c,d\in\mathbb{R}}\{(\vec{y} - X\vec{a}, \vec{y} - X\vec{a})_{l_2}\}$, we are going to find (a_1, a) such that

$$\frac{\partial}{\partial a_0}(\vec{y} - X\vec{a}, \vec{y} - X\vec{a})_{l_2} = 0$$
$$\frac{\partial}{\partial a_1}(\vec{y} - X\vec{a}, \vec{y} - X\vec{a})_{l_2} = 0.$$

We have

$$\frac{\partial}{\partial a_i}(\vec{y} - X\vec{a}, \vec{y} - X\vec{a})_{l_2}$$
$$= \lim_{h\to 0} \frac{(\vec{y} - X(\vec{a} + h\hat{e}_i), \vec{y} - X(\vec{a} + h\hat{e}_i))_{l_2} - (\vec{y} - X\vec{a}, \vec{y} - X\vec{a})_{l_2}}{h}$$

Since
$(\vec{y} - X(\vec{a} + h\hat{e}_i), \vec{y} - X(\vec{a} + h\hat{e}_i))_{l_2} = (\vec{y} - X\vec{a}, \vec{y} - X\vec{a})_{l_2} - 2h(\vec{y} - X\vec{a}, X\hat{e}_i)_{l_2} + h^2(X\hat{e}_i, X\hat{e}_i)_{l_2}$,

$$\partial_i(\vec{y} - X\vec{a}, \vec{y} - X\vec{a})_{l_2} = \lim_{h\to 0}\Big(-2(\vec{y} - X\vec{a}, X\hat{e}_i)_{l_2} + h(X\hat{e}_i, X\hat{e}_i)_{l_2}\Big)$$
$$= -2(\vec{y} - X\vec{a}, X\hat{e}_i)_{l_2} = 2(X^t\vec{y} - X^tX\vec{a}, \hat{e}_i)_{l_2}$$

Cont.

Therefore, we have

$$2(X^t\vec{y} - X^tX\vec{a}, \hat{e}_i)_{l_2} = \partial_i(\vec{y} - X\vec{a}, \vec{y} - X\vec{a})_{l_2} = 0$$

and thus,

$$X^tX\vec{a} = X^t\vec{y}.$$

where

$$X^tX = \begin{pmatrix} \sum_i x_i^2 & \sum_i x_i \\ \sum_i x_i & n \end{pmatrix}, \; X^t\vec{y} = \begin{pmatrix} \sum_i x_i y_i \\ \sum_i y_i \end{pmatrix}$$

Example

We consider

$$\{x_i\}_{i=1}^n = \{1, 2, 3, 4, 5, 6, 7, 8, 9\}$$

$$\{y_i\}_{i=1}^n = \{1.2, 2.1, 2.9, 4.0, 5.1, 6.0, 6.8, 7.8, 9.0\}$$

$$\{x_i^2\}_{i=1}^n = \{1, 4, 9, 16, 25, 36, 49, 64, 81\}$$

$$\{x_i y_i\}_{i=1}^n = \{1.2, 4.2, 8.7, 16, 25.5, 36, 47.6, 62.4, 81\}$$

$$\sum_{i=1}^n x_i = 45, \ \sum_{i=1}^n x_i^2 = 285$$

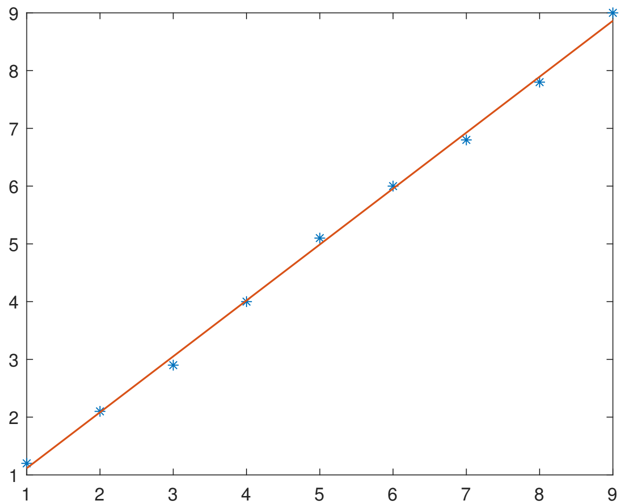
$$\sum_{i=1}^n y_i = 44.9, \ \sum_{i=1}^n x_i y_i = 282.6$$

Cont.

$$\begin{pmatrix} 285 & 45 \\ 45 & 9 \end{pmatrix} \begin{pmatrix} a_1 \\ a_0 \end{pmatrix} = \begin{pmatrix} 282.6 \\ 44.9 \end{pmatrix}$$

we have

$$\begin{pmatrix} a_1 \\ a_0 \end{pmatrix} \approx \begin{pmatrix} 0.9683 \\ 0.1472 \end{pmatrix}.$$



Least Squares Approximation for higher degree poly.

We then consider the approximate with higher degree polynomial.
For example, we consider $P(x) = a_2x^2 + a_1x + a_0$ with
$E_2(a_2, a_1, a_0) = \sum_{i=1}^{n} |y_i - a_2x_i^2 - a_1x_i - a_0|^2$.
By using similar argument, we have $E_2(a_2, a_1, a_0) = (\vec{y} - X\vec{a}, \vec{y} - X\vec{a})_{l_2}$
where $\vec{y} = (y_1, \ldots, y_n)^t$, $\vec{a} = (a_2, a_1, a_0)^t$,

$$X = \begin{pmatrix} x_1^2 & x_2^2 & \cdots & x_n^2 \\ x_1 & x_2 & \cdots & x_n \\ 1 & 1 & \cdots & 1 \end{pmatrix}^t$$

To find (a_2, a_1, a_0), we need to solve the system of equation

$$2(X^t\vec{y} - X^tX\vec{a}, \hat{e}_i)_{l_2} = \partial_i(\vec{y} - X\vec{a}, \vec{y} - X\vec{a})_{l_2} = 0 \ \text{ for } i = 0, 1, 2$$

Cont.

Thus, we need to solve

$$X^t X\vec{a} = X^t\vec{y}$$

where

$$X^t X = \begin{pmatrix} \sum_{i=1}^n x_i^4 & \sum_{i=1}^n x_i^3 & \sum_{i=1}^n x_i^2 \\ \sum_{i=1}^n x_i^3 & \sum_{i=1}^n x_i^2 & \sum_{i=1}^n x_i \\ \sum_{i=1}^n x_i^2 & \sum_{i=1}^n x_i & n \end{pmatrix}, \quad X^t\vec{y} = \begin{pmatrix} \sum_{i=1}^n x_i^2 y_i \\ \sum_{i=1}^n x_i y_i \\ \sum_{i=1}^n y_i \end{pmatrix}$$

Example

We consider

$$\{x_i\}_{i=1}^n = \{1, 2, 3, 4, 5, 6, 7, 8, 9\}$$

$$\{y_i\}_{i=1}^n = \{1.2, 2.1, 2.9, 4.0, 5.1, 6.0, 6.8, 7.8, 9.0\}$$

We have

$$\sum_i x_i^4 = 15333, \ \sum_i x_i^3 = 2025, \ \sum_i x_i^2 = 285$$

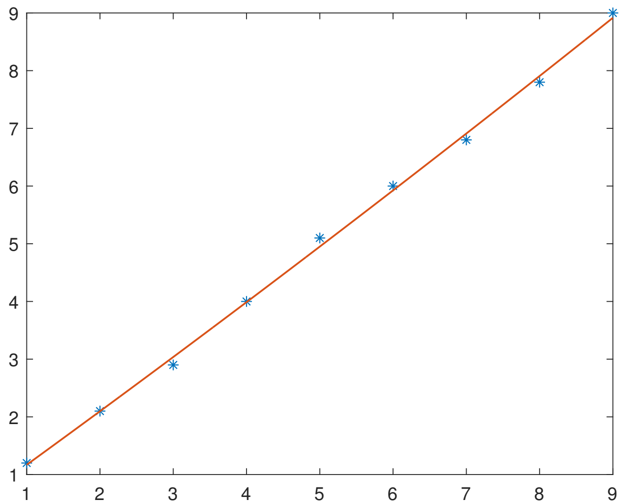
$$\sum_i x_i = 45, \ \sum_i x_i^2 y_i = 2004.6, \ \sum_i x_i y_i = 282.6, \ \sum_i y_i = 44.9$$

Cont.

We would like to find (a_2, a_1, a_0)

$$\begin{pmatrix} 15333 & 2025 & 285 \\ 2025 & 285 & 45 \\ 285 & 45 & 9 \end{pmatrix} \begin{pmatrix} a_2 \\ a_1 \\ a_0 \end{pmatrix} = \begin{pmatrix} 2004.6 \\ 282.6 \\ 44.9 \end{pmatrix}$$

$$a_2 = 0.0057, \ a_1 = 0.9110, \ a_0 = 0.2524$$



Least Squares Approximation for higher degree poly.

In general, we consider $P(x) = a_m x^m + a_{m-1}x^{m-1} + \cdots + a_1 x + a_0$ with $E_2(\vec{a}) = \sum_{i=1}^{n} |y_i - (\sum_{j=0}^{m} a_j x_i^j)|^2$.
By using similar argument, we have $E_2(\vec{a}) = (\vec{y} - X\vec{a}, \vec{y} - X\vec{a})_{l_2}$ where $\vec{y} = (y_1, \ldots, y_n)^t$, $\vec{a} = (a_m, \ldots, a_0)^t$,

$$X = \begin{pmatrix} x_1^m & x_2^m & \cdots & x_n^m \\ x_1^{m-1} & x_2^{m-1} & \cdots & x_n^{m-1} \\ \vdots & \vdots & & \vdots \\ 1 & 1 & \cdots & 1 \end{pmatrix}^t$$

Cont.

To find $(a_m, \ldots, a_0)$, we need to solve the system of equation

$$X^t X\vec{a} = X^t y.$$

We have

$$X^t X = \begin{pmatrix} \sum_i x_i^{2m} & \sum_i x_i^{2m-1} & \cdots & \sum_i x_i^m \\ \sum_i x_i^{2m-1} & \sum_i x_i^{2m-2} & \cdots & \sum_i x_i^{m-1} \\ \vdots & \vdots & & \vdots \\ \sum_i x_i^m & \sum_i x_i^{m-1} & \cdots & n \end{pmatrix}, \quad X^t y = \begin{pmatrix} \sum_i x_i^m y_i \\ \sum_i x_i^{m-1} y_i \\ \vdots \\ \sum_i y_i \end{pmatrix}$$

Least Squares Approximation for non-polynomial

We can also approximate the function f by non-polynomial function g. For example, $g(x) = \tilde{a}_0 e^{a_1 x}$. Instead of minimizing the l_2-error, we can minimize the l_2 mismatch in log-scale.
That is

$$
\begin{aligned}
E_2(a_1, a_0) &= \sum_{i=1}^{n} |ln(y_i) - ln(\tilde{a}_0 e^{a_1 x_i})|^2 \\
&= \sum_{i=1}^{n} |ln(y_i) - ln(\tilde{a}_0) - a_1 x_i|^2
\end{aligned}
$$

We denote $a_0 = ln(\tilde{a})$. To find (a_1, a_0), we need to solve the system of equation

$$2(X^t ln(\vec{y}) - X^t X \vec{a}, \hat{e}_i)_{l_2} = \partial_i (ln(\vec{y}) - X\vec{a}, ln(\vec{y}) - X\vec{a})_{l_2} = 0 \quad \text{for } i = 0, 1, 2$$

where $ln\vec{y} = (lny_1, \ldots, lny_n)^t$, $\vec{a} = (a_1, a_0)^t$,

$$X = \begin{pmatrix} x_1 & x_2 & \cdots & x_n \\ 1 & 1 & \cdots & 1 \end{pmatrix}^t.$$

Cont.

Therefore, we need to solve

$$X^t X\vec{a} = X^t ln\vec{y}.$$

We have

$$X^t X = \begin{pmatrix} \sum_i x_i^2 & \sum_i x_i \\ \sum_i x_i & n \end{pmatrix}, \ X^t\vec{y} = \begin{pmatrix} \sum_i x_i ln(y_i) \\ \sum_i ln(y_i) \end{pmatrix}$$

and $f(x) \approx e^{a_0 + a_1 x}$

Lecture 22

Orthogonal Polynomials and Least Squares Approximations

Least squares approximation

In the previous lecture, we considered the problem of discrete least squares approximation to fit a collection of data.
The other approximation problem concerns the approximation of functions. That is, given a function $f \in C[a, b]$, we would like find a polynomial P_n of degree at most n which minimize the error

$$\int_a^b |f(x) - P_n(x)|^2 dx.$$

Cont.

We consider the polynomial P_n is written in form of

$$P_n(x) = a_n x^n + a_{n-1} x^{n-1} + \cdots + a_1 x + a_0 = \sum_{k=0}^{n} a_k x^k.$$

We would like to find $(a_0, a_1, \ldots, a_n)$ minimize the "cost function" defined as

$$E_2(a_0, a_1, \ldots, a_n) = \int_a^b \left(f(x) - \sum_{k=0}^{n} a_k x^k \right)^2 dx.$$

To find $(a_0, a_1, \ldots, a_n)$, w can solve

$$\frac{\partial E_2}{\partial a_i} = 0 \text{ for each } i = 0, \cdots, n$$

Cont.

Since

$$E_2(a_0, a_1, \ldots, a_n) = \int_a^b f^2(x)dx - 2\sum_{k=0}^{n} a_k \int_a^b f(x)x^k dx + \int_a^b \left(\sum_{k=0}^{n} a_k x^k\right)^2,$$

we have

$$\frac{\partial E_2}{\partial a_i} = -2\int_a^b f(x)x^i dx + 2\sum_{k=0}^{n} a_k \int_a^b x^{i+k} dx \text{ for } i = 0, \ldots, n.$$

Therefore, we can find $(a_0, a_1, \ldots, a_n)$ by solving the equations

$$\sum_{k=0}^{n} a_k \int_a^b x^{i+k} dx = \int_a^b f(x)x^i dx \text{ for } i = 0, \ldots, n$$

Example

We consider the function $f(x) = \sin(\pi x)$, on the interval $[0, 1]$. We would like to find a polynomial of degree 2, P_2 to approximate f.
For $P_2(x) = a_2x^2 + a_1x + a_0$, we will solve

$$
\begin{aligned}
a_2 \int_0^1 x^4 dx + a_1 \int_0^1 x^3 dx + a_0 \int_0^1 x^2 dx &= \int_0^1 x^2 \sin(\pi x) dx \\
a_2 \int_0^1 x^3 dx + a_1 \int_0^1 x^2 dx + a_0 \int_0^1 x dx &= \int_0^1 x \sin(\pi x) dx \\
a_2 \int_0^1 x^2 dx + a_1 \int_0^1 x dx + a_0 \int_0^1 1 dx &= \int_0^1 \sin(\pi x) dx
\end{aligned}
$$

We write the equations in a matrix form

$$\begin{pmatrix} \int_0^1 x^4 dx & \int_0^1 x^3 dx & \int_0^1 x^2 dx \\ \int_0^1 x^3 dx & \int_0^1 x^2 dx & \int_0^1 x dx \\ \int_0^1 x^2 dx & \int_0^1 x dx & \int_0^1 1 dx \end{pmatrix} \begin{pmatrix} a_2 \\ a_1 \\ a_0 \end{pmatrix} = \begin{pmatrix} \int_0^1 x^2 \sin(\pi x) dx \\ \int_0^1 x \sin(\pi x) dx \\ \int_0^1 \sin(\pi x) dx \end{pmatrix}.$$

Since $\int_0^1 x^i dx = \frac{1}{i+1} x^{i+1}|_0^1 = \frac{1}{i+1}$, $\int_0^1 \sin(\pi x) dx = \frac{2}{\pi}$, $\int_0^1 x \sin(\pi x) dx = \frac{1}{\pi}$ and $\int_0^1 x^2 \sin(\pi x) = \frac{(\pi^2 - 4)}{\pi^3}$, we have

$$\begin{pmatrix} \frac{1}{5} & \frac{1}{4} & \frac{1}{3} \\ \frac{1}{4} & \frac{1}{3} & \frac{1}{2} \\ \frac{1}{3} & \frac{1}{2} & 1 \end{pmatrix} \begin{pmatrix} a_2 \\ a_1 \\ a_0 \end{pmatrix} = \begin{pmatrix} \frac{\pi^2 - 4}{\pi^3} \\ \frac{1}{\pi} \\ \frac{2}{\pi} \end{pmatrix}.$$

Example

$$\begin{pmatrix} 1 & \frac{5}{4} & \frac{5}{3} \\ 1 & \frac{4}{3} & 2 \\ 1 & \frac{3}{2} & 3 \end{pmatrix} \begin{pmatrix} a_2 \\ a_1 \\ a_0 \end{pmatrix} = \begin{pmatrix} \frac{5(\pi^2-4)}{\pi^3} \\ \frac{4}{\pi} \\ \frac{6}{\pi} \end{pmatrix}.$$

and

$$\begin{pmatrix} 1 & \frac{5}{4} & \frac{5}{3} \\ 0 & \frac{1}{12} & \frac{1}{3} \\ 0 & \frac{1}{4} & \frac{4}{3} \end{pmatrix} \begin{pmatrix} a_2 \\ a_1 \\ a_0 \end{pmatrix} = \begin{pmatrix} \frac{5(\pi^2-4)}{\pi^3} \\ \frac{4\pi^2-5(\pi^2-4)}{\pi^3} \\ \frac{6\pi^2-5(\pi^2-4)}{\pi^3} \end{pmatrix}.$$

$$\begin{pmatrix} 1 & \frac{5}{4} & \frac{5}{3} \\ 0 & \frac{1}{4} & 1 \\ 0 & \frac{1}{4} & \frac{4}{3} \end{pmatrix} \begin{pmatrix} a_2 \\ a_1 \\ a_0 \end{pmatrix} = \begin{pmatrix} \frac{5(\pi^2-4)}{\pi^3} \\ \frac{60-3\pi^2}{\pi^3} \\ \frac{\pi^2+20}{\pi^3} \end{pmatrix}.$$

$$\begin{pmatrix} 1 & \frac{5}{4} & \frac{5}{3} \\ 0 & \frac{1}{4} & 1 \\ 0 & 0 & \frac{1}{3} \end{pmatrix} \begin{pmatrix} a_2 \\ a_1 \\ a_0 \end{pmatrix} = \begin{pmatrix} \frac{5(\pi^2-4)}{\pi^3} \\ \frac{60-3\pi^2}{\pi^3} \\ \frac{-5(\pi^2-4)}{\pi^3} - \frac{60-3\pi^2}{\pi^3} \end{pmatrix}.$$

We have $a_0 = 3\frac{4\pi^2 - 40}{\pi^3} = \frac{12\pi^2 - 120}{\pi^3} \approx -0.050$,
$a_1 = \frac{720 - 60\pi^2}{\pi^3} \approx 4.122$, $a_2 = -\frac{720 - 60\pi^2}{\pi^3} \approx -4.122$.
Thus, we have

$$P_2(x) \approx -4.122x^2 + 4.122x - 0.050.$$

Linearly Independent Functions

Next, we are going to introduce an easier way to find the polynomial P_n. Before introducing the method, we first recall the definition of a set of linear independent functions $\{\phi_n\}$
The set of functions $\{\phi_0, \ldots, \phi_n\}$ is said to be linearly independent on $[a, b]$ if, whenever

$$c_0\phi_0(x) + c_1\phi_1(x) + \cdots + c_n\phi_n(x) = 0 \; \forall x \in [a, b],$$

we have $c_0 = c_1 = \cdots = c_n = 0$. Otherwise the set of functions is said to be linearly dependent.

Theorem

Theorem

Suppose that,for each $j = 0, 1, \ldots, n$, f_j is polynomial of degree j. Then $\{f_0, \ldots, f_n\}$ is linearly independent on any interval $[a, b]$.

Proof.

Suppose

$$g(x) := c_0 f_0(x) + c_1 f_1(x) + \cdots + c_n f_n(x) = 0 \ \forall x \in [a, b].$$

Since g is a polynomial, we have $g(x) := b_n x^n + b_{n-1} x^{n-1} + \cdots + b_1 x + b_0$. Since $g(x) \equiv 0$, we have $b_i = 0 \ \forall i$. Since f_j is a polynomial of degree j, we have

$$f_j = \sum_{i=0}^{j} a_{j,i} x^i$$

with $a_{j,j} \neq 0$. □

Cont.

Thus, we have

$$g(x) = \sum_{j=0}^{n} \sum_{i=0}^{j} c_j a_{j,i} x^i = \sum_{i=0}^{n} \sum_{j=i}^{n} (c_j a_{j,i}) x^i.$$

We consider i_0 the largest integer such that $c_{i_0} \neq 0$. We have

$$0 = b_{i_0} = \sum_{j=i_0}^{n} (c_j a_{j,i}) = c_{i_0} a_{i_0,i_0}, \text{ (contradiction)}.$$

We obtain that there not exist a i with $c_i \neq 0$ and thus $c_i = 0$ for all $i = 0, \cdots, n$.

Theorem

Let $\mathcal{P}_n$ be the set of all polynomial of degree at most n. Namely,

$$\mathcal{P}_n = \{\sum_{i=0}^{n} a_i x^i |\ a_i \in \mathbb{R},\ \text{for } i = 0, \ldots, n\}.$$

Theorem

Suppose $\{f_1, \cdots, f_n\}$ is a set of linearly independent polynomials in $\mathcal{P}_n$. Then any $f \in \mathcal{P}_n$ can be written uniquely as a linear combination of f_0, f_1, $\cdots$, f_n

Orthogonal functions

To discuss general function approximation requires the introduction of the notions of weight functions and orthogonality.

Definition

An integrable function w is called a weight function on the interval I if $w(x) \geq 0$, for all x in I, but $w \not\equiv 0$ in any subinterval of I.

The purpose of a weight function is to assign varying degrees of importance to approximations on certain portions of the interval. For example, the weight function

$$w(x) = \frac{1}{\sqrt{1 - x^2}}$$

places less emphasis near the center of the interval $(-1, 1)$ and more emphasis when $|x|$ is near 1

Cont.

Suppose $\{\phi_0, \phi_1, \cdots, \phi_n\}$ is a set of linearly independent functions on $[a, b]$ and ω is a weight function for $[a, b]$. Given $f \in C([a, b])$, we would like to find a linear combination

$$P(x) = \sum_{k=0}^{n} a_k \phi_k(x)$$

to minimize the error

$$E_w(a_0, \ldots, a_n) = \int_a^n w(x)\Big(f(x) - \sum_{k=0}^{n} a_k \phi_k(x)\Big)^2 dx.$$

For $w \equiv 1$ and $\phi_k(x) = x^k$, the problem will reduce the standard least square problem.

Cont.

To solve the problem, we need to solve

$$0 = \frac{\partial E}{\partial a_i} = 2\int_a^b w(x)\phi_i(x)\Big(f(x) - \sum_{k=0}^{n} a_k\phi_k(x)\Big)dx$$

and

$$\sum_{k=0}^{n} a_k \int_a^b w(x)\phi_i(x)\phi_k(x)dx = \int_a^b w(x)\phi_i(x)f(x)dx$$

We can write the system in form of $M\vec{a} = \vec{b}$.

$$M = (m_{ij})\,,\ m_{ij} = \int_a^b w(x)\phi_i(x)\phi_j(x)$$

$$\vec{b} = (b_i),\ b_i = \int_a^b w(x)\phi_i(x)f(x)dx$$

Orthogonality

We can also prove that M is always a symmetric semi-positive definite matrix and
M is SPD if and only if $\{\phi_i\}_{i=0}^n$ is linearly independent.
We consider

$$\int_a^b w(x)\phi_i(x)\phi_j(x) = \begin{cases} 0 & \text{if } j \neq i \\ \alpha_j > 0 & \text{if } j = i \end{cases}$$

and thus, we find a_j easily by considering

$$a_j = \frac{1}{\alpha_j}\int_a^b w(x)\phi_j(x)f(x)dx.$$

(since M is diagonal matrix and $(M^{-1})_{ij} = 0$ if $i \neq j$ and $(M^{-1})_{ij} = \dfrac{1}{\alpha_i}$ if $i = j$)

Cont.

We recall the definition of orthogonal set of function.

Definition

$\{\phi_i\}_{i=0}^n$ is said to be orthogonal with respect to the weighted L_2 inner product $(f, g)_w = \int_a^b w(x)f(x)g(x)dx$ if

$$(\phi_i, \phi_j)_w = \begin{cases} 0 & \text{if } j \neq i \\ \alpha_j > 0 & \text{if } j = i \end{cases}$$

If, in addition, $\alpha_j = 1$ for each $j = 0, 1, \cdots, n$, the set is said to be orthonormal.

Theorem

Theorem

If $\{\phi_0, \ldots, \phi_n\}$ *is an orthogonal set of functions with respect to the weighted* L_2 *inner product* $(f, g)_w = \int_a^b w(x)f(x)g(x)dx$*, then the least squares approximation to* f *on* $[a, b]$ *with respect to* w *is*

$$P(x) = \sum_{j=0}^{n} a_j\phi_j(x),$$

where, for each $j = 0, 1, \cdots, n$*,*

$$a_j = \frac{(\phi_j, f)_w}{(\phi_j, \phi_j)_w} = \frac{1}{\alpha_j}(\phi_j(x), f(x))_w.$$

Theorem

Theorem

The set of polynomial functions $\{\phi_0, \ldots, \phi_n\}$ *defined in the following way is orthogonal with respect to the weighted* L_2 *inner product* $(f, g)_w = \int_a^b w(x)f(x)g(x)dx.$

$$\phi_0 \equiv 1, \ \phi_1 = x - B_1 \ \forall x \in [a, b]$$

where

$$B_1 = \frac{(x\phi_0, \phi_0)_w}{(\phi_0, \phi_0)_w},$$

and when $k \geq 2$,

$$\phi_k(x) = (x - B_k)\phi_{k-1} - C_k\phi_{k-2}(x), \ \forall x \in [a, b]$$

where

$$B_k = \frac{(x\phi_{k-1}, \phi_{k-1})_w}{(\phi_{k-1}, \phi_{k-1})_w}, \ C_k = \frac{(x\phi_{k-1}, \phi_{k-2})_w}{(\phi_{k-2}, \phi_{k-2})_w}$$

Corollary

Corollary

For any $n > 0$, the set of polynomial functions $\{\phi_0, \cdots, \phi_n\}$ given in the previous theorem is linearly independent and

$$(\phi_n, Q_k)_w = 0,$$

for any polynomial Q_k of degree $k < n$.

Thus, we can use the recursive formula to obtain the Legendre polynomials, $P_n(x)$ which is orthogonal with respect to standard L_2 inner product on $[-1, 1]$.

$$P_0 \equiv 1,\ P_1 = x - \frac{(x1, 1)}{(1, 1)} = x\ \forall x \in [a, b]$$

and

$$P_2 = (x - B_2)P_1 - C_2 P_0(x) = (x - \frac{(x^2, x)}{(x, x)})x - \frac{(x^2, 1)}{(1, 1)}1 = x^2 - \frac{1}{3},$$

$$P_3 = (x - B_3)P_2 - C_3 P_1(x) = \cdots = x^3 - \frac{3}{5}x.$$

Lecture 23

Chebyshev Polynomial

Orthogonal Polynomial with $w(x) = \dfrac{1}{(1-x^2)^{\frac{1}{2}}}$ *in* $(-1, 1)$

The Chebyshev Polynomial $\{T_n(x)\}$ are orthogonal polynomials on $(-1, 1)$ with respect to the weight function $w = \dfrac{1}{(1-x^2)^{\frac{1}{2}}}$.
We claim that the Chebyshev Polynomial can be written in form of

$$T_n(x) = \cos(n \arccos(x)).$$

Using the formula in the previous lecture, we get

$$\begin{aligned} T_0(x) &= 1, \\ T_1(x) &= x - \frac{\int_{-}^{1} x(1-x^2)^{-\frac{1}{2}}\,dx}{\int_{-1}^{1}(1-x^2)^{-\frac{1}{2}}\,dx} = x = \cos(\arccos(x)). \end{aligned}$$

Cont.

Using the recursive formula, we have

$$\tilde{T}_2 = (x - B_2)T_1 - C_2 T_0$$

where

$$B_2 = \frac{(xT_1, T_1)_w}{(T_1, T_1)_w}, \ C_2 = \frac{(xT_1, T_0)}{(T_1, T_1)_w}.$$

First, we have

$$\begin{aligned}(T_1, T_1)_w &= \int_{-1}^{1} \frac{\cos^2(\arccos(x))}{\sqrt{1-x^2}}dx \\ &= 2\int_{0}^{1} \frac{\cos^2(\arccos(x))}{\sqrt{1-x^2}}dx\end{aligned}$$

Cont.

We consider $x = \cos\theta$ and obtain $dx = -\sin\theta d\theta$.
Thus, we have $T_n(x) = \cos(n\arccos(x)) = \cos(n\theta)$ and

$$
\begin{aligned}
(T_0, T_0)_w &= \int_\pi^0 \frac{1}{\sqrt{1-x^2}}dx \\
&= \int_\pi^0 \frac{-\sin\theta}{\sqrt{1-\cos^2\theta}}d\theta \\
&= \int_0^\pi d\theta = \pi
\end{aligned}
$$

Next, we have

$$(xT_1, T_1)_w = \int_{-1}^1 \frac{x\cos^2(\arccos(x))}{\sqrt{1-x^2}}dx = 0$$

since $\dfrac{x\cos^2(n\arccos(x))}{\sqrt{1-x^2}}$ is a odd function.

Cont.

Using the same argument, we have

$$\begin{aligned}(xT_n, T_{n-1})_w &= \int_{-1}^{1} \frac{x \cdot x}{\sqrt{1-x^2}} d\theta. \\ &= \int_{\pi}^{0} \frac{-\sin\theta\cos\theta\cos\theta}{\sqrt{1-\cos^2\theta}} d\theta \\ &= \int_{0}^{\pi} \cos^2\theta d\theta\end{aligned}$$

Since $\cos^2\theta = \dfrac{1+\cos(2\theta)}{2}$, we have

$$\begin{aligned}(xT_n, T_{n-1})_w &= \int_{0}^{\pi} \frac{1+\cos(2\theta)}{2} d\theta \\ &= \frac{\pi}{2}\end{aligned}$$

Thus, we have

$$\tilde{T}_2 = x^2 - \frac{1}{2} = \frac{1}{2}\cos(2\theta) = \frac{1}{2}T_2(x)$$

Orthogonality

For $n \geq 0$, we would like to show $T_n(x) = \cos(n \arccos(x))$ are orthogonal polynomials on $(-1, 1)$ with respect to the weight function $w = \dfrac{1}{(1-x^2)^{\frac{1}{2}}}$.

$$
\begin{aligned}
(T_n, T_m)_w &= \int_{-1}^{1} \frac{\cos(n \arccos(x)) \cos(m \arccos(x))}{\sqrt{1-x^2}} dx \\
&= -\int_{\pi}^{0} \frac{\sin\theta \cos(n\theta)\cos(m\theta)}{\sin\theta} d\theta \\
&= \int_0^{\pi} \cos(n\theta)\cos(m\theta) d\theta
\end{aligned}
$$

Using $\cos(n\theta)\cos(m\theta) = \dfrac{1}{2}\Big(\cos((n+m)\theta) + \cos((n-m)\theta)\Big)$, we have

$$
\begin{aligned}
(T_n, T_m)_w &= \int_0^{\pi} \frac{1}{2}\Big(\cos((n+m)\theta) + \cos((n-m)\theta)\Big) d\theta \\
&= \begin{cases} \dfrac{1}{2}\Big(\dfrac{\sin((n+m)\theta)|_0^{\pi}}{n+m} + \dfrac{\sin((n-m)\theta)|_0^{\pi}}{n-m}\Big) = 0 & \text{if } n \neq m \\ \dfrac{1}{2}\Big(\dfrac{\sin((n+m)\theta)|_0^{\pi}}{n+m} + \pi\Big) = \dfrac{\pi}{2} & \text{if } n = m \end{cases}
\end{aligned}
$$

Recurrence relation

Using trigonometric formulas, we have

$$T_{n+1}(x) = \cos((n+1)\theta) = \cos\theta\cos(n\theta) - \sin\theta\sin(n\theta)$$

$$T_{n-1}(x) = \cos((n-1)\theta) = \cos\theta\cos(n\theta) + \sin\theta\sin(n\theta)$$

Thus, we have

$$\begin{aligned} T_{n+1}(x) &= 2\cos\theta\cos(n\theta) - T_{n-1}(x) \\ &= 2xT_n(x) - T_{n-1}(x) \end{aligned}$$

Cont.

Since $T_0 \equiv 1$ and $T_1(x) \equiv x$, by the recurrence formula

$$T_{n+1}(x) = 2xT_n(x) - T_{n-1}(x),$$

we have T_n is a polynomial of degree n with leading coefficient 2^{n-1} for all $n \geq 1$.

Theorem

The Chebyshev polynomial $T_n(x)$ of degree $n \geq 1$ has n simple zeros in $[-1, 1]$ at

$$\tilde{x}_k = \cos(\frac{2k-1}{2n}\pi), \text{ for each } k = 1, \ldots, n.$$

Moreover, $T_n(x)$ assumes its absolute extrema at

$$x'_k = \cos(\frac{k}{n}\pi), \text{ with } T_n(x'_k) = (-1)^k, \text{ for each } k = 0, 1, \ldots, n.$$

Proof

Proof.

By the definition of $T_n(x) = \cos(n \arccos(x))$, we have

$$\begin{aligned}
T(\tilde{x}_k) &= \cos(n \arccos(\cos(\frac{2k-1}{2n}\pi))) \\
&= \cos(\frac{2k-1}{2}\pi) = \cos(k\pi - \frac{\pi}{2}) \\
&= 0
\end{aligned}$$

Moreover, $T'(x) = \dfrac{n \sin(n \arccos(x))}{\sqrt{1-x^2}}$ and we have

$T'(x'_k) = \dfrac{n \sin(k\pi))}{\sqrt{1-x^2}} = 0$ for $k = 1, 2, \ldots, n-1$. Hence x'_k for $k = 1, 2, \ldots, n-1$ plus two end-points $x'_0 = 1$ and $x'_n = -1$ are all of the possible extrema of f. $\square$

Proof.

Next, we can check

$$T(x'_k) = \cos(n \arccos(\cos(\frac{k}{n}\pi))) = \cos(k\pi)) = (-1)^k$$

for $k = 0, 1, \cdots, n$.

Since all of the possible extrema are at x'_k for $k = 0, 1, \ldots, n$, we have

$$\max_{x \in [a,b]} \{|f(x)|\} = \max_{k=0,1,\ldots n} \{|f(x'_k)|\} = 1.$$

Since $|T(x'_k)| = 1$ for all $k = 0, 1, \ldots, n$, we have that all of the extrema of are at x'_k for $k = 0, 1, \ldots, n$. $\square$

Definition of $\tilde{T}_n$ and $\tilde{\Pi}_n$

Next, we define $\tilde{T}_n(x) = \dfrac{T_n}{2^{n-1}}$ which are the monic orthogonal polynomials with respect to $w(x) = (1 - x^2)^{-\frac{1}{2}}$.
We then have

$$\tilde{T}_n(\tilde{x}_k) = 0,\ \tilde{T}_n(x'_k) = \frac{(-1)^k}{2^{n-1}}$$

We denote $\tilde{\Pi}_n$ as the set of all monic polynomials of degree n, that is

$$\tilde{\Pi}_n = \{f | f(x) = x^n + \sum_{k=0}^{n-1} a_k x^k\}$$

.

Theorem

Theorem

The polynomial $\tilde{T}_n(x)$ *have the property that*

$$\frac{1}{2^{n-1}} = \max_{x\in[-1,1]} |\tilde{T}_n(x)| \leq \max_{x\in[-1,1]} |P_n(x)|, \ \forall P_n \in \tilde{\Pi}_n$$

Moreover, equality occurs only if $P_n \equiv \tilde{T}_n$.

Proof

We assume $P_n \in \tilde{\Pi}_n$ satisfying

$$|P_n(x)| \leq \frac{1}{2^{n-1}} \, \forall x \in [-1, 1].$$

We consider $g = P_n - \tilde{T}_n$ and have g is a polynomial is a polynomial of degree at most $n-1$. We also have

$$g(x'_k) = P_n(x'_k) - \tilde{T}_n(x'_k) = P_n(x'_k) - \frac{(-1)^k}{2^{n-1}}.$$

Since $|P_n(x'_k)| \leq \dfrac{1}{2^{n-1}}$, we have $g(x'_k) \geq 0$ if k is odd and $g(x'_k) \leq 0$ if k is even. By Intermediate Value Theorem, there exist a root of g between x'_{k-1} and x'_k for each $n \geq k \geq 1$.
Therefore, g is polynomial of degree at most with n roots implying $g \equiv 0$

Minimizing Lagrange Interpolation Error

We consider $f \in C^{n+1}[-1,1]$ and P_n is the n-th Langrange Interpolating polynomial.
By the error estimate, we have

$$f(x) - P_n(x) = \frac{f^{(n+1)}(\xi(x))}{(n+1)!}(x-x_0)(x-x_1)\cdots(x-x_n)$$

where x_i are the interpolating points.
It is generally different to control the term $f^{(n+1)}(\xi(x))$.
To reduce the error, we can choose a special set of points $\{x_i\}$ to minimize $\max_{x\in[-1,1]}\{|(x-x_0)(x-x_1)\cdots(x-x_n)|\}$.
Since $(x-x_0)(x-x_1)\cdots(x-x_n)$ is a degree $n+1$ monic polynomial, we have the best choice to minimize $\max_{x\in[-1,1]}\{|(x-x_0)(x-x_1)\cdots(x-x_n)|\}$ is

$$x_k = \tilde{x}_k = \cos(\frac{2k-1}{2(n+1)}\pi).$$

Cont.

That is, we can choose the interpolating points to be the roots of the $(n+1)$-th Chebyshev polynomial and obtain the following error estimate.

Corollary

Suppose that $P(x)$ is the interpolating polynomial of degree at most n with points at the zeros of T_{n+1}. Then

$$\max_{x\in[-1,1]}\{|f(x)-P(x)|\} \leq \frac{\max_{x\in[-1,1]}\{|f^{(n+1)}(x)|\}}{2^n(n+1)!} \quad \textit{for all } f \in C^{n+1}([-1,1]).$$

Lecture 24

Trigonometric Polynomial

Instead of using polynomial $P(x) = \sum_{k=0}^{n} a_k x^k$ to approximate the function f, we can use the trigonometric polynomial $\sum_{k=0}^{n} a_k \cos(kx) + \sum_{k=1}^{n} b_k \sin(kx)$ to approximate the function.
We claim the set
$\{\frac{1}{2}, \cos(x), \sin(x), \cos(2x), \sin(2x), \ldots, \cos(kx), \sin(kx), \ldots\}$ is a set of orthogonal function in $[-\pi, \pi]$ with respect to $w(x) \equiv 1$.
Therefore, it is easy to preform the least square approximation to determine the coefficient b_k and c_k

Orthogonality of Trigonometric Polynomial

We would like to show

$$
\begin{aligned}
(\sin(nx), \sin(mx)) &= \int_{-\pi}^{\pi} \sin(nx)\sin(mx)dx \\
&= \begin{cases} \pi & \text{if } n = m \\ 0 & \text{if } n \neq m \end{cases},
\end{aligned}
$$

$$
(\cos(nx), \cos(mx)) = \begin{cases} \pi & \text{if } n = m \\ 0 & \text{if } n \neq m \end{cases},
$$

and

$$
(\cos(nx), \sin(mx)) = 0
$$

for $n, m \geq 1$.

Cont.

By $\sin(nx)\sin(mx) = \dfrac{\cos((n-m)x) - \cos((n+m)x)}{2}$, we have

$$\begin{aligned}
\int_{-\pi}^{\pi} \sin(nx)\sin(mx)dx &= \frac{1}{2}\int_{-\pi}^{\pi} \cos((n-m)x) - \cos((n+m)x)dx \\
&= \begin{cases} \pi - \dfrac{\sin((n+m)x)|_{-\pi}^{\pi}}{2(n+m)} & \text{for } n = m \\ \dfrac{1}{2}\Big(\dfrac{\sin((n-m)x)|_{-\pi}^{\pi}}{n-m} - \dfrac{\sin((n+m)x)|_{-\pi}^{\pi}}{n+m}\Big) & \text{for } n \neq m \end{cases} \\
&= \begin{cases} \pi & \text{if } n = m \\ 0 & \text{if } n \neq m \end{cases}
\end{aligned}$$

for $n \geq 1$.

Cont.

By $\cos(nx)\cos(mx) = \dfrac{\cos((n-m)x) + \cos((n+m)x)}{2}$,

$$\int_{-\pi}^{\pi} \cos(nx)\cos(mx)dx = \frac{1}{2}\int_{-\pi}^{\pi} \cos((n-m)x) + \cos((n+m)x)dx$$

$$= \begin{cases} \pi + \dfrac{\sin((n+m)x)|_{-\pi}^{\pi}}{2(n+m)} & \text{for } n = m \\ \dfrac{1}{2}\Big(\dfrac{\sin((n-m)x)|_{-\pi}^{\pi}}{n-m} + \dfrac{\sin((n+m)x)|_{-\pi}^{\pi}}{n+m}\Big) & \text{for } n \neq m \end{cases}$$

$$= \begin{cases} \pi & \text{if } n = m \\ 0 & \text{if } n \neq m \end{cases}$$

By $\sin(nx)\cos(mx) = \dfrac{\sin((n+m)x) + \sin((n-m)x)}{2}$,

$$
\begin{aligned}
\int_{-\pi}^{\pi} \sin(nx)\cos(mx)dx &= \frac{1}{2}\int_{-\pi}^{\pi} \sin((n+m)x) + \sin((n-m)x)dx \\
&= \begin{cases} -\dfrac{\cos((n+m)x)|_{-\pi}^{\pi}}{2(n+m)} + 0 & \text{for } n = m \\ -\dfrac{1}{2}\Big(\dfrac{\cos((n+m)x)|_{-\pi}^{\pi}}{(n+m)} + \dfrac{\cos((n-m)x)|_{-\pi}^{\pi}}{(n-m)}\Big) & \text{for } n \neq m \end{cases} \\
&= 0
\end{aligned}
$$

Cont.

Next, we need to proof $(\frac{1}{2}, \cos(nx)) = (\frac{1}{2}, \sin(nx)) = 0$ for $n \geq 1$ and

$$(\frac{1}{2}, \frac{1}{2}) = \frac{\pi}{2}.$$

By the result in the previous lecture, the coefficient $\{a_0, b_1, a_1, \ldots b_n, a_n\}$ for least square approximation of f which minimize

$$E = \int_{-\pi}^{\pi} |f - \frac{a_0}{2} - \sum_{k=1}^{n} a_k \cos(kx) - \sum_{k=1}^{n} b_k \sin(kx)|^2$$

satisfy

$$a_k = \frac{\int_{-\pi}^{\pi} f(x) \cos(kx) dx}{\pi} \text{ for } k = 0, 1, \ldots, n$$
$$b_k = \frac{\int_{-\pi}^{\pi} f(x) \sin(kx) dx}{\pi} \text{ for } k = 1, \ldots, n$$

Convergent result

Theorem

If $f \in C([-\pi, \pi])$, f is piecewise continuous differentiable and f is 2π periodic, we then have

$$S_n(x) = \frac{a_0}{2} + \sum_{k=1}^{n} a_k \cos(kx) + \sum_{k=1}^{n} b_k \sin(kx)$$

convergent to $f(x)$ for every $x \in [-\pi, \pi]$ where

$$a_k = \frac{\int_{-\pi}^{\pi} f(x)\cos(kx)dx}{\pi} \text{ for } k = 0, 1, \ldots, n,$$
$$b_k = \frac{\int_{-\pi}^{\pi} f(x)\sin(kx)dx}{\pi} \text{ for } k = 1, \ldots, n.$$

Example

For example, $f(x) = |x|$, we have

$$S_0(x) = \frac{\int_{-\pi}^{\pi} |x| dx}{2\pi} = \frac{\pi}{2}.$$

$$S_1(x) = \frac{\pi}{2} + \frac{\int_{-\pi}^{\pi} |x| \cos(x) dx}{\pi} \cos(x) + \frac{\int_{-\pi}^{\pi} |x| \sin(x) dx}{\pi} \sin(x)$$

$$= \frac{\pi}{2} + \frac{2}{\pi} \int_0^{\pi} x \cos(x) dx \cos(x) = \frac{\pi}{2} - \frac{4}{\pi} \cos(x)$$

$$\vdots \vdots$$

$$S_n(x) = \frac{\pi}{2} + \frac{2}{\pi} \sum_{k}^{n} \frac{(-1)^k - 1}{k^2} \cos(kx).$$

Discrete Trigonometric Approximation

Next, we will discuss the discrete trigonometric approximation.
We consider a set of points $\{x_j\}_{j=0}^{2m-1}$ is equally spaced such that

$$x_j = -\pi + \frac{j}{m}\pi \text{ for } j = -0, 1, \ldots 2m - 1.$$

Then we are going to approximate the function f by trigonometric polynomial which minimize the cost function

$$E(S_n) = \sum_{j=0}^{2m-1} |f(x_j) - S_n(x_j)|^2$$

where $S_n = \frac{a_0}{2} + \sum_{k=1}^{n} a_k \cos(kx) + \sum_{k=1}^{n} b_k \sin(kx)$.

We denote ϕ_i as $\phi_0(x) \equiv \dfrac{1}{2}$,

$$\begin{aligned}\phi_j &= \cos(jx) \text{ for } 1 \le j \le n,\\ \phi_{n+j} &= \sin(jx) \text{ for } 1 \le j \le n\end{aligned}$$

Lemma

We have

$$\sum_{k=0}^{2m-1} \phi_i(x_k)\phi_j(x_k) = \begin{cases} m & \text{if } i = j \\ 0 & \text{if } i \neq j \end{cases}$$

Proof

By $\sin(nx)\sin(mx) = \dfrac{\cos((n-m)x) - \cos((n+m)x)}{2}$, we have

$$\sum_{k=0}^{2m-1} \sin(nx_k)\sin(mx_k) = \sum_{k=0}^{2m-1} \frac{\cos((n-m)x_k) - \cos((n+m)x_k)}{2}$$

Since $e^{ix} = \cos(x) + i\sin(x)$, we have $\cos(x) = Re(e^{ix})$ and

$$\sum_{k=0}^{2m-1} \cos(rx_k) = Re(\sum_{k=0}^{2m-1} e^{ir(-\pi+\frac{k}{m}\pi)}) = Re(e^{-ir\pi}\sum_{k=0}^{2m-1} e^{ir\frac{k}{m}\pi})$$

$$= \begin{cases} Re(e^{-r\pi}\dfrac{e^{ir2\pi}-1}{e^{ir\frac{\pi}{m}}-1}) = 0 & \text{if } r \neq 0 \\ 2m & \text{if } r = 0 \end{cases}$$

Thus,

$$\sum_{k=0}^{2m-1} \sin(ix_k)\sin(jx_k) = \begin{cases} m & \text{if } i = j \\ 0 & \text{if } i \neq j \end{cases}$$

Proof

By $\cos(nx)\cos(mx) = \dfrac{\cos((n-m)x) + \cos((n+m)x)}{2}$, we have

$$\sum_{k=0}^{2m-1} \cos(ix_k)\cos(jx_k) = \sum_{k=0}^{2m-1} \frac{\cos((n-m)x_k) + \cos((n+m)x_k)}{2}$$

Thus,

$$\sum_{k=0}^{2m-1} \cos(ix_k)\cos(jx_k) = \begin{cases} m & \text{if } i = j \\ 0 & \text{if } i \neq j \end{cases}$$

By $\sin(nx)\cos(mx) = \dfrac{\sin((n+m)x) + \sin((n-m)x)}{2}$,

Proof

By $\sin(nx)\cos(mx) = \dfrac{\sin((n+m)x)+\sin((n-m)x)}{2}$, we have

$$\sum_{k=0}^{2m-1} \sin(nx)\cos(mx) = \sum_{k=0}^{2m-1} \frac{\sin((n+m)x)+\sin((n-m)x)}{2}$$

and

$$\begin{aligned}\sum_{k=0}^{2m-1} \sin(rx_k) &= \frac{1}{i} Im\Big(\sum_{k=0}^{2m-1} e^{ir(-\pi+\frac{k}{m}\pi)}\Big)\\ &= \frac{1}{i} Im\Big(e^{-r\pi}\frac{e^{ir2\pi}-1}{e^{ir\frac{\pi}{m}}-1}\Big) = 0.\end{aligned}$$

Thus,

$$\sum_{k=0}^{2m-1} \sin(nx)\cos(mx) = 0.$$

Theorem

We consider

$$S_n = \frac{a_0}{2} + \sum_{k=1}^{n} a_k \cos(kx) + \sum_{k=1}^{n} b_k \sin(kx)$$

is the trigonometric polynomial which minimize the cost function

$$E(S_n) = \sum_{j=0}^{2m-1} |f(x_j) - S_n(x_j)|^2.$$

We then have

$$a_k = \frac{1}{m} \sum_{j=0}^{2m-1} y_j \cos(kx_j) \text{ for } k = 0, 1, \ldots n$$

$$b_k = \frac{1}{m} \sum_{j=0}^{2m-1} y_j \sin(kx_j) \text{ for } k = 1, 2, \ldots n.$$

Lecture 25

Rational Function approximation

Rational Function

Instead of using polynomial $P(x) = \sum_{k=0}^{n} a_k x^k$ or trigonometric polynomial $P(x) = \frac{a_0}{2} + \sum_{k=1}^{n} a_k \cos(kx) + \sum_{k=1}^{n} b_k \sin(kx)$ to approximate the function f, we can use a rational function to approximate a function.

A rational function r of degree N which is a fraction of polynomials such that

$$r = \frac{p}{q}$$

where p and q are polynomials whose degrees sum to N.

For example , $r(x) = \dfrac{x+3}{x^2+2x+2}$. We have r is a rational function of degree 3.

Padé Approximation

Suppose r is a rational function of degree $N = n + m$ of the form

$$r(x) = \frac{p(x)}{q(x)} = \frac{p_0 + p_1 x + p_2 x^2 + \cdots + p_n x^n}{q_0 + q_1 x + q_2 x^2 + \cdots + q_m x^m}$$

that is used to approximate a function f on a closed interval I containing zero.

For r to be defined at zero requires that $q_0 \neq 0$. In fact, we can assume that $q_0 = 1$.

Thus, there are $N + 1$ parameters $q_1, q_2, \ldots, q_m, p_0, p_1, \ldots, p_n$ to be determined.

Cont.

The Padé approximation technique, is the extension of Taylor polynomial approximation to rational functions.
It choose the $N+1$ parameters such that

$$f^{(k)}(0) = r^{(k)}(0)$$

for $k = 0, 1, \ldots, N$.
Consider the difference

$$\begin{aligned} f(x) - r(x) = f(x) - \frac{p(x)}{q(x)} &= \frac{f(x)q(x) - p(x)}{q(x)}. \\ &= \frac{f(x)\sum_{i=0}^{m} q_i x^i - \sum_{i=0}^{n} p_i x^i}{q(x)}. \end{aligned}$$

If f can be written as $f(x) = \sum_{i=0}^{\infty} a_i x^i$, then we have

$$\frac{\left(\sum_{i=0}^{\infty} a_i x^i\right)\sum_{i=0}^{m} q_i x^i - \sum_{i=0}^{n} p_i x^i}{q(x)}$$

We can check that

$$f^{(k)}(0) = r^{(k)}(0) \ \forall 0 \leq k \leq N,$$

if and only if

$$\Big(\sum_{i=0}^{\infty} a_i x^i\Big)\Big(\sum_{i=0}^{m} q_i x^i\Big) - \sum_{i=0}^{n} p_i x^i = \sum_{i=N+1}^{\infty} b_i x^i.$$

That is, we require the first N coefficients of the expression are zeros. Namely

$$(\sum_{i=0}^{k} a_i q_{k-i}) - p_k = 0 \text{ for } k = 0, 1, \ldots, N.$$

Using the same idea, we consider the rational function r is written in form of

$$r(x) = \frac{\sum_{k=0}^{n} p_k T_k(x)}{\sum_{k=0}^{m} q_k T_k(x)}.$$

To find the coefficients, we consider

$$\Big(\sum_{i=0}^{\infty} a_i T_i(x)\Big)\Big(\sum_{i=0}^{m} q_i T_i(x)\Big) - \sum_{i=0}^{n} p_i T_i(x) = \sum_{i=N+1}^{\infty} b_i T_i(x)$$

Using $T_i(x)T_j(x) = \frac{1}{2}\Big(T_{i+j}(x) + T_{|i-j|}(x)\Big)$ with $a_i = \frac{(f, T_i)_w}{(T_i, T_i)_w}$, we can solve the system of equations.

Lecture 26

Fast Fourier transform

In the lecture 24, we introduced how to use discrete least square method to find a trigonometric polynomial approximating a function f.

Consider $\{x_i\}_{i=0}^{2m-1}$ is equally spaced and $x_i = -\pi + \frac{i\pi}{m}$.
We can claim if we consider the number of data points is the same as the number of trigonometric polynomials, then the approximating trigonometric polynomial exactly match all of the data points.
That is, given $(x_i, y_i)_{i=0}^{2m-1}$, if we consider

$P(x) = \frac{a_0}{2} + \sum_{k=1}^{m} a_k \cos(kx) + \sum_{k=1}^{m-1} b_k \sin(kx)$, then we have

$$P(x_i) = f(x_i) \text{ for all } x_i$$

Cont.

Instead of considering $P(x) = \frac{a_0}{2} + \sum_{k=1}^{m} a_k \cos(kx) + \sum_{k=1}^{m-1} b_k \sin(kx)$, we can write $P(x)$ in form of $P(x) = \sum_{k=-m}^{m-1} c_k e^{i\pi kx}$ where $i^2 = -1$ and c_k are complex number.
It is not hard to find the relation between c_k and a_k, b_k.
Since

$$c_k e^{ikx} = c_k(\cos(kx) + i\sin(kx))$$
$$c_{-k} e^{-ikx} = c_{-k}(\cos(kx) - i\sin(kx)),$$

we have

$$\sum_{k=-m}^{m} c_k e^{ikx} = c_0 + \sum_{i=1}^{m}(c_k + c_{-k})\cos(kx) + \sum_{i=1}^{m} i(c_k - c_{-k})\sin(kx)$$

Cont.

If $P(x)$ is real, then we have for $k = 1, \cdots, m$

$$\begin{aligned} Re(c_k) + Re(c_{-k}) &= a_k \\ Im(c_k) + Im(c_{-k}) &= 0 \\ Re(c_k) - Re(c_{-k}) &= 0 \\ Im(c_k) - Im(c_{-k}) &= -b_k. \end{aligned}$$

Thus,

$$c_k = \frac{a_k}{2} - i\frac{b_k}{2}, c_{-k} = c_k^* = \frac{a_k}{2} + i\frac{b_k}{2}$$

and

$$c_0 = \frac{a_0}{2}$$

Thus, we can consider for $P(x)$ is written as

$$P(x) = \sum_{k=-m}^{m-1} c_{k+m} e^{ikx}$$

Next, we define a mapping $F : \mathbb{C}^{2m} \to \mathbb{C}^{2m}$ by

$$(F(\vec{y}))_k := \frac{1}{2m} \sum_{j=0}^{2m-1} y_j e^{-i(k-m)x_j}$$

We consider $P(x) = \sum_{k=-m}^{m-1} F(\vec{y})_{k+m} e^{ikx}$. We then have

$$\begin{aligned}
P(x_l) &= \frac{1}{2m} \sum_{k=0}^{2m-1} F(\vec{y})_k e^{i(k-m)x_l} \\
&= \frac{1}{2m} \sum_{k=0}^{2m-1} \sum_{j=0}^{2m-1} y_j e^{i(k-m)(x_l - x_j)} \\
&= \frac{1}{2m} \sum_{j=0}^{2m-1} y_j \sum_{k=-m}^{m-1} e^{ik\pi(\frac{l-j}{m})}
\end{aligned}$$

Cont.

Since

$$\sum_{k=-m}^{m-1} e^{ik\pi\left(\frac{l-j}{m}\right)} = \begin{cases} 2m & \text{if } l-j=0 \\ 0 & \text{if } l-j\neq 0 \end{cases},$$

we have

$$P(x_l) = y_l \ \forall l = 0, 1, \ldots 2m-1$$

Cont.

We next define a mapping $iF : \mathbb{C}^{2m} \to \mathbb{C}^{2m}$ by

$$(iF(\vec{y}))_k := \sum_{j=0}^{2m-1} y_j e^{i(k-m)x_j}$$

We can prove that

$$iF \circ F(\vec{y}) = \vec{y} \ \forall y \in \mathbb{C}^{2m}.$$

$$\begin{aligned}
iF \circ F(\vec{y})_l &= \sum_{j=0}^{2m-1} F(\vec{y})_j e^{i(l-m)x_j} \\
&= \frac{1}{2m} \sum_{j=0}^{2m-1} \sum_{k=0}^{2m-1} y_k e^{-i(j-m)x_k} e^{i(l-m)x_j} \\
&= \frac{1}{2m} \sum_{k=0}^{2m-1} e^{imx_k} y_k \sum_{j=0}^{2m-1} e^{-i(j)x_k} e^{i(l-m)(-\pi+\frac{j\pi}{m})} \\
&= \frac{1}{2m} \sum_{k=0}^{2m-1} e^{imx_k} e^{-i\pi(l-m)} y_k \sum_{j=0}^{2m-1} e^{-i(j)(\frac{(k-m)\pi}{m})} e^{i(l-m)(\frac{j\pi}{m})}
\end{aligned}$$

Cont.

$$
\begin{aligned}
iF \circ F(\vec{y})_l &= \frac{1}{2m} \sum_{k=0}^{2m-1} e^{imx_k} e^{-i\pi(l-m)} y_k \sum_{j=0}^{2m-1} e^{-i(j)(\frac{(k-m)\pi}{m})} e^{i(l-m)(\frac{j\pi}{m})} \\
&= \frac{1}{2m} \sum_{k=0}^{2m-1} e^{imx_k} e^{-i\pi(l-m)} y_k \sum_{j=0}^{2m-1} e^{i(j)(\frac{(l-k)\pi}{m})} \\
&= \sum_{k=0}^{2m-1} e^{imx_k} e^{-i\pi(l-m)} y_k \delta_{lk} = e^{imx_l} e^{-i\pi(l-m)} y_l \\
&= e^{i\pi m(\frac{l}{m}-1)} e^{-i\pi(l-m)} y_l = y_l.
\end{aligned}
$$

Then the question is how the compute the coefficient $F(\vec{y})_i$.
If we compute the coefficient one by one, for each i, we need to compute

$$\frac{1}{2m}\sum_{j=0}^{2m-1} y_j e^{-i(k-m)x_j}$$

which require $O(4m)$ operations.
Therefore, we require $O(8m^2)$ operations totally (sum up for $i = 0, 1, \ldots 2m-1$)

Fast Fourier transform

In fact, there is a fast algorithm to compute $F(\vec{y})$.
We can write the linear map F in a matrix form such that

$$F(\vec{y}) = M\vec{y}$$

where

$$\begin{aligned} M_{kj} &= e^{-i\frac{\pi}{m}(k-m)(j-m)} = .e^{-i\frac{\pi}{m}(m^2-m(k+j)+kj)} \\ &= (-1)^{m-(k+j)}e^{-i\frac{\pi kj}{m}} \end{aligned}$$

Fast Fourier transform

$$M = (-1)^m D \begin{pmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & \omega & \omega^2 & \cdots & \omega^{(2m-1)} \\ 1 & \omega^2 & \omega^4 & \cdots & \omega^{2(2m-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \omega^{(2m-1)} & \omega^{2(2m-1)} & \cdots & \omega^{(2m-1)^2} \end{pmatrix} D$$

$\omega = e^{-i\frac{\pi}{m}}, \omega^{2m} = 1,\ D_{kj} = (-1)^{kj}\delta_{kj}$

Fast Fourier transform

Consider $\tilde{y} = Dy$,

$$\begin{aligned}
DF(\vec{y})_j &= \sum_{i=0} \omega^{2ji} \tilde{y}_{2i} + \omega^j \sum_{i=0} \omega^{2ji} \tilde{y}_{2i+1} \\
&= \sum_{i=0} \omega^{2ij} \tilde{y}_{2i} + \omega^j \sum_{i=0} \omega^{2ji} \tilde{y}_{2i+1} \\
DF(\vec{y})_{j+m} &= \sum_{i=0} \omega^{2ij+2mi} \tilde{y}_{2i} + \omega^{j+m} \sum_{i=0} \omega^{2ij} \tilde{y}_{2i+1} \\
&= \sum_{i=0} \omega^{2ij} \tilde{y}_{2i} + \omega^j (-1)^m \sum_{i=0} \omega^{2ij} \tilde{y}_{2i+1}
\end{aligned}$$

We define two matrix $D^{(1)}$ and $M^{(1)}$ by

$$\begin{aligned}
M^{(1)}_{ij} &= \omega^{2ji} \\
D^{(1)}_{ij} &= \delta_{ij} \omega^j
\end{aligned}$$

Then, we have

$$F(\vec{y}) = (-1)^m \begin{pmatrix} M^{(1)} \vec{y}_{even} - D^{(1)} M^{(1)} \vec{y}_{odd} \\ M^{(1)} \vec{y}_{even} + D^{(1)} M^{(1)} \vec{y}_{odd} \end{pmatrix}$$

Fast Fourier transform

It requires $O(2m^2)$ operations to compute $M^{(1)}\vec{y}_{even}$. Similarly, it requires $O(2m^2)$ operations to compute $M^{(1)}\vec{y}_{odd}$.
To compute $D^{(1)}\vec{z}$, we need $O(m)$ operations.
Thus, we only need $4m^2 + m$ operations to obtain $F(\vec{y})$.
Since the $M^{(1)}$has a similar structure as M, we can use the same method to reduce the operations for computing $M^{(1)}\vec{y}$.
Repeating the argument, we will have a method to compute $F(\vec{y})$ with $O(mlog(m))$ operation.

Final Exam

8 questions, 100 points

1. (10 points) Answer either (a) or (b) below to derive an $O(h^4)$ formula to approximate $f'(x_0)$. If you answer both, only (a) will be graded.
   (a) Consider the expression $Af(x_0 - 3h) + Bf(x_0 - h) + Cf(x_0 + h) + Df(x_0 + 3h)$. Expand in fifth Taylor polynomials, and choose A, B, C and D appropriately.
   (b) Using the idea Richardson's Extrapolation to obtain the formula of $f'(x_0) \approx Af(x_0 - 3h) + Bf(x_0 - h) + Cf(x_0 + h) + Df(x_0 + 3h)$.

(a) Note that $f'(x_0) = \frac{1}{2h}\left(f(x_0+h) - f(x_0-h)\right) + C_1 h^2 + O(h^4)$

$$f'(x_0) = \frac{1}{2h}\left(f(x_0+h) - f(x_0-h)\right) + C_1h^2 + O(h^4) \quad ①$$

$$f'(x_0) = \frac{1}{6h}\left(f(x_0+3h) - f(x_0-3h)\right) + C_1 9h^2 + O(h^4) \quad ②$$

$9 \times ① - ②$

$$8f'(x_0) = \frac{1}{h}\left(\frac{9}{2}f(x_0+h) + \frac{1}{6}f(x_0+3h) - \frac{9}{2}f(x_0-h) - \frac{1}{6}f(x_0-3h)\right) + O(h^4)$$

$$f'(x_0) = \frac{1}{h}\left(\frac{9}{16}f(x_0+h) + \frac{1}{48}f(x_0+3h) - \frac{9}{16}f(x_0-h) - \frac{1}{48}(x_0-3h)\right) + O(h^4)$$

Thus $A = -\frac{1}{48h}$ $B = -\frac{9}{16h}$ $C = \frac{9}{16h}$ $D = \frac{1}{48h}$

Composite T rule.

$$\int_a^b f(x)\,dx = \frac{h}{2}\left[f(a) + 2\sum_{j=1}^{n-1} f(x_j) + f(b)\right] - \frac{b-a}{12}h^2 f^{(2)}(\xi)$$

Composite Simpson's rule.

$$\int_a^b f(x)\,dx = \frac{h}{3}\left[f(a) + 4\sum_{\substack{j=1 \\ j \text{ odd}}}^{n-1} f(x_j) + 2\sum_{\substack{j=2 \\ j \text{ even}}}^{n-2} f(x_j) + f(b)\right] - \frac{b-a}{180} h^4 f^{(4)}_{\xi}$$

2. (10 points) Use the Composite Simpson's Rule with $n = 2$ to approximate $\int_1^3 \frac{2}{x^2+1}dx$. That is

$$\int_1^3 \frac{2}{x^2+1}dx \approx \sum_{i=0}^{4} \alpha_i f(x_i)$$

where $x_i = 1 + ih$ with $h = \frac{1}{2}$.

$a = 1 \qquad b = 3 \qquad h = \frac{1}{2} \qquad f(x) = \frac{2}{x^2+1}$

$$\int_1^3 f(x)\,dx = \frac{h}{3}\left[f(1) + 4f\left(\tfrac{3}{2}\right) + 2f(2) + 4f\left(\tfrac{5}{2}\right) + f(3)\right]$$

$$= \frac{1}{6}\left[\frac{2}{1+1} + 4\cdot\frac{2}{\frac{9}{4}+1} + 2\cdot\frac{2}{4+1} + 4\cdot\frac{2}{\frac{25}{4}+1} + \frac{2}{10}\right]$$

$$= \frac{1}{6}\left[1 + \frac{32}{13} + \frac{4}{5} + \frac{32}{29} + \frac{1}{5}\right]$$

$$\approx 0.9275$$

3. (10 points) Given the following data, find the least square approximation to $y = a_0 + a_1 x$.

x_i	0	1	2	3
y_i	-0.1	1.2	2.1	2.9

$$\sum_{i=1}^{4} y_i = a_1 \sum_{i=1}^{4} x_i + 4a_0 \quad ①$$

$$\sum_{i=1}^{4} x_i y_i = a_1 \sum_{i=1}^{4} x_i^2 + a_0 \sum_{i=1}^{4} x_i \quad ②$$

$$\Rightarrow \begin{bmatrix} 4 & 6 \\ 6 & 14 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \end{bmatrix} = \begin{bmatrix} 6.1 \\ 14.1 \end{bmatrix}$$

$a_0 = \frac{1}{25}$ $\qquad a_1 = 0.99.$

$\frac{x^3}{3} - \frac{x^2}{2} - \frac{1}{6}x \int_0^1 \left((x-\frac{1}{2})^2 - \frac{1}{12}\right)\cdot(x+1)$

$x^2 - x - \frac{1}{6}$ $\quad x^3 - x^2 - \frac{1}{6}x$, $\quad \frac{x^4}{4} - \frac{x^3}{3} - \frac{x^2}{12}$

4. (20 points) We define P_n be a polynomial of degree n such that P_n is a monic polynomial and

$$\int_0^1 P_n(x)Q(x)dx = 0 \text{ for any polynomial } Q \text{ of degree at most } n-1.$$

We have $P_0 = 1$ and $P_1 = x - \int_0^1 x dx = x - \frac{1}{2}$.

(a) (10 points) Prove that

$$P_n = (x - B_n)P_{n-1} - C_n P_{n-2}$$

where $B_n = \frac{\int_0^1 xP_{n-1}P_{n-1}dx}{\int_0^1 P_{n-1}P_{n-1}dx}$ and $C_n = \frac{\int_0^1 xP_{n-1}P_{n-2}dx}{\int_0^1 P_{n-2}P_{n-2}dx}$.

(b) (10 points) Using part (a) to find P_2.

(a) Since P_n is a monic polynomial, then for any n

$P_{n-1} = x^{n-1} + \sum_{i=0}^{n-2} c_i x^i \qquad P_{n-2} = x^{n-2} + \sum_{i=0}^{n-3} c_i x^i$

$(x - B_n)P_{n-1} - C_n P_{n-2} = x^n - B_n x^{n-1} + B_n \sum_{i=0}^{n-2} c_i x^i - C_n x^{n-2} - C_n \sum_{i=0}^{n-3} c_i x^i$

Thus, P_n is a monic polynomial for any n.

Next prove $\int_0^1 P_n(x)Q(x)dx = 0$ for any polynomial Q degree $\le n-1$

then any polynomial Q, which degree $\le n$ can be writen as $Q_n = \sum_{i=0}^{i=n} a_i P_i$, thus the problem reduced to prove $\int_0^1 P_n P_i dx = 0$ for $i \le n-1$

Prove by induction

First check initial condition $P_1 = x - \int_0^1 x dx = x - \frac{1}{2}$

Next assume P_n hold for $n \le k$

Want to show that $P_{k+1} = (x - B_{k+1})P_k - C_{k+1}P_{k-1}$

so we need to show $\int_0^1 P_{k+1} Q_n dx = 0$

which is equanlent to show that

$\langle (x - B_{k+1}) P_k - C_{k+1} P_{k-1}, P_i \rangle = 0$ for all $i \le k$

when $i = k$.

$\langle (x - B_{k+1}) P_k - C_{k+1} P_{k-1}, P_k \rangle$

$= \int_0^1 x P_k P_k \, dx - \frac{\int_0^1 x P_k P_k \, dx}{\cancel{\int_0^1 P_k P_k \, dx}} \cdot \cancel{\int_0^1 P_k P_k \, dx}$

$- C_{k+1} \int_0^1 P_k \cdot P_{k-1} \, dx$

Since. $\int_0^1 P_k Q_n \, dx = 0$, the $\int_0^1 P_k \cdot P_{k-1} \, dx = 0$

Thus $\langle (x - B_{k+1}) P_k - C_{k+1} P_{k-1}, P_k \rangle$

$= \int_0^1 x P_k P_k \, dx - \int_0^1 x P_k P_k \, dx = 0$.

For. $i = k-1$.

$B_{k+1} \int_0^1 P_k P_{k-1} \, dx = 0$.

$C_{k+1} \int_0^1 P_{k-1} P_{k-1} \, dx = \frac{\int_0^1 x P_k P_{k-1} \, dx}{\int_0^1 P_{k-1} P_{k-1} \, dx} \cdot \int_0^1 P_{k-1} P_{k-1} \, dx$

Thus $\langle (x - B_{k+1}) P_k - C_{k+1} P_{k-1}, P_k \rangle$

$= \int_0^1 x P_k P_{k-1} \, dx - \int_0^1 x P_k P_{k-1} \, dx$

$= 0$.

For $i \le k-2$.

$B_{k+1} \int_0^1 P_k P_i \, dx = 0 \quad C_{k+1} \int_0^1 P_{k-1} P_i \, dx = 0$.

$\int_0^1 x P_k P_i \, dx = \int_0^1 P_k (x P_i) \, dx$

Notice that $x P_i = Q_{i+1}$ is a polynomial degree at most $k-1$

then $\int_0^1 P_k Q_{i+1} \, dx = 0$

Thus $\langle (x - B_{k+1}) P_k - C_{k+1} P_{k-1}, P_i \rangle = 0$

Hold for $i \le n$.

$$P_n = (x - B_n) P_{n-1} - C_n P_{n-2}$$

We have $P_0 = 1$ and $P_1 = x - \int_0^1 x dx = x - \frac{1}{2}$.

(a) (10 points) Prove that

$$P_n = (x - B_n)P_{n-1} - C_n P_{n-2}$$

where $B_n = \frac{\int_0^1 x P_{n-1} P_{n-1} dx}{\int_0^1 P_{n-1} P_{n-1} dx}$ and $C_n = \frac{\int_0^1 x P_{n-1} P_{n-2} dx}{\int_0^1 P_{n-2} P_{n-2} dx}$.

(b) (10 points) Using part (a) to find P_2.

$$P_2 = (x - B_2) P_1 - C_2 P_0$$

$$B_2 = \frac{\int_0^1 x P_1 P_1 \, dx}{\int_0^1 P_1 P_1 \, dx} = \frac{\int_0^1 (x - \frac{1}{2})^2 x \, dx}{\int_0^1 (x - \frac{1}{2})^2 \, dx}$$

$$= \frac{\int_0^1 x^3 - x^2 + \frac{1}{4} x \, dx}{\int_0^1 x^2 - x + \frac{1}{4} \, dx} = \frac{\frac{x^4}{4} - \frac{x^3}{3} + \frac{x}{8}}{\frac{x^3}{3} - \frac{x^2}{2} + \frac{1}{4}x} \Bigg|_0^1 = \frac{\frac{1}{4} - \frac{1}{3} + \frac{1}{8}}{\frac{1}{3} - \frac{1}{2} + \frac{1}{4}} = \frac{1}{2}$$

$$C_2 = \frac{\int_0^1 x (x - \frac{1}{2}) \cdot 1 \, dx}{\int_0^1 1 \, dx} = \frac{x^3}{3} - \frac{x^2}{4} \Bigg|_0^1 = \frac{1}{12}$$

Thus $P_2 = (x - \frac{1}{2}) \cdot (x - \frac{1}{2}) - \frac{1}{12}$

$$= x^2 - x + \frac{1}{3}$$

$$\int_0^1 P_n \cdot P_{n-1}\,dx = 0.$$

for $n-1$

$$LHS = \langle (x - B_n) P_{n-1} - C_n P_{n-2},\ P_{n-1} \rangle$$

$$= \int_0^1 x P_{n-1} P_{n-1}\,dx - \frac{\int_0^1 x P_{n-1} P_{n-1}\,dx}{\int_0^1 P_{n-1} P_{n-1}\,dx} \cdot \int_0^1 P_{n-1} P_{n-1}\,dx - \frac{\int_0^1 x P_{n-1} P_{n-2}\,dx}{\int_0^1 P_{n-2} P_{n-2}\,dx} \cdot \int_0^1 P_{n-2} P_{n-1}\,dx$$

$$= \int_0^1 x P_{n-1} P_{n-1}\,dx - \int_0^1 x P_{n-1} P_{n-1}\,dx - \frac{\int_0^1 x P_{n-1} P_{n-2}\,dx}{\int_0^1 P_{n-2} P_{n-2}\,dx} \cdot (0)$$

$$= 0$$

for $\int_0^1 P_n \cdot P_k\,dx = 0$ $\quad k = n-2.$

$RHS = 0$

$$LHS = \int_0^1 x P_{n-1} \cdot P_k\,dx - \frac{\int_0^1 x P_{n-1} P_{n-1}\,dx}{\int_0^1 P_{n-1} P_{n-1}\,dx} \cdot \int_0^1 P_{n-1} \cdot P_k\,dx$$

$$- \frac{\int_0^1 x P_{n-1} P_{n-2}\,dx}{\int_0^1 P_{n-2} P_{n-2}\,dx} \cdot \int_0^1 P_{n-2} \cdot P_k\,dx.$$

since $k = n-2$ then $\int_0^1 P_{n-1} \cdot P_k\,dx = 0.$

$$LHS = \int_0^1 x P_{n-1} P_{n-2}\,dx - \int_0^1 x P_{n-1} P_{n-2}\,dx = 0.$$

for $k \le n-3$ $\quad \int_0^1 P_{n-2} \cdot P_k\,dx = 0$

$\int_0^1 P_{n-1} \cdot P_k\,dx = 0.$ $\quad$ Let $x P_k = Q_{k+1}$ which is Poly

$\int_0^1 P_{n-1} \cdot (x P_k)\,dx = \int_0^1 P_{n-1} Q_{k+1}\,dx = 0.$ degree at most $n-2$

Thus $LHS = 0.$

5. (10 points) We consider T_3 be the Chebyshev polynomial of degree 3.

(a) (5 points) Use the zeros of T_3 to construct a Lagrange interpolating polynomial of degree 2 to approximate $f(x) = 2^x$ on the interval $[-1, 1]$.

(b) (5 points) Find the maximum possible error.

$$w(x) = \frac{1}{\sqrt{1-x^2}}$$

$$T_3 = 0 \qquad n\cos^{-1}(x) = \frac{\pi}{2} + k\pi$$

$$\Rightarrow x_k = \cos\left(\frac{2k+1}{6}\pi\right)$$

$$x_0 = \frac{\sqrt{3}}{2} \qquad x_1 = 0 \qquad x_2 = -\frac{\sqrt{3}}{2}$$

$\frac{\sqrt{3}}{2}$	$2^{\frac{\sqrt{3}}{2}}$	$\dfrac{1-2^{\frac{\sqrt{3}}{2}}}{-\frac{\sqrt{3}}{2}}$	$\dfrac{2^{-\frac{\sqrt{3}}{2}}+2^{\frac{\sqrt{3}}{2}}-2}{2}$
0	1	$\dfrac{2^{-\frac{\sqrt{3}}{2}}-1}{-\frac{\sqrt{3}}{2}}$	
$-\frac{\sqrt{3}}{2}$	$2^{-\frac{\sqrt{3}}{2}}$		

$$P(x) = 2^{\frac{\sqrt{3}}{2}} - \frac{1-2^{\frac{\sqrt{3}}{2}}}{\frac{\sqrt{3}}{2}}\left(x-\frac{\sqrt{3}}{2}\right) - \frac{2^{-\frac{\sqrt{3}}{2}}+2^{\frac{\sqrt{3}}{2}}-2}{2}\left(x-\frac{\sqrt{3}}{2}\right)\cdot x$$

5. (10 points) We consider T_3 be the Chebyshev polynomial of degree 3.

(a) (5 points) Use the zeros of T_3 to construct a Lagrange interpolating polynomial of degree 2 to approximate $f(x) = 2^x$ on the interval $[-1, 1]$.

(b) (5 points) Find the maximum possible error.

$$T_n(x) = \cos(n \cos^{-1}(x)) \quad n \geq 0$$

$$T_3 = \cos(3\cos^{-1}(x)) = 0$$

$$3\cos^{-1}(x) = \frac{\pi}{2} + k\pi$$

$$x_k = \cos\left(\frac{2k-1}{6}\pi\right) \quad k = 1, 2, \ldots n$$

Thus three zero are $x_0 = \frac{\sqrt{3}}{2}$ $x_1 = 0$ $x_2 = -\frac{\sqrt{3}}{2}$.

$$P(x) = 2^{\frac{\sqrt{3}}{2}} \frac{(x-0)(x+\frac{\sqrt{3}}{2})}{\frac{3}{2}} - \frac{(x-\frac{\sqrt{3}}{2})(x+\frac{\sqrt{3}}{2})}{\frac{3}{4}}$$

$$+ 2^{-\frac{\sqrt{3}}{2}} \frac{(x-0)(x-\frac{\sqrt{3}}{2})}{\frac{3}{2}}$$

(b) $$\text{Error} = \frac{f^{(3)}(x)}{3!} \left| x\left(x^2 - \frac{3}{4}\right) \right|$$

$$= \frac{(\ln 2)^3}{3} \cdot \frac{1}{4}$$

$$= \frac{(\ln 2)^3}{12}$$

6. (10 points) For $f = |\cos(x)|$ on the interval $[-\pi, \pi]$, find a least square approximation of f in form of $P(x) = \dfrac{a_0}{2} + a_1\cos(x) + b_1\sin(x)$.

Hint: $\cos(\alpha)\cos(\beta) = \frac{1}{2}(\cos(\alpha-\beta) + \cos(\alpha+\beta))$ and $\sin(\alpha)\cos(\beta) = \frac{1}{2}(\sin(\alpha+\beta) + \sin(\alpha-\beta))$.

$$\langle f(x) - p(x), \tfrac{1}{2} \rangle = 0.$$

$$\int_{-\pi}^{\pi} f(x)\cdot\tfrac{1}{2}\,dx - \int_{-\pi}^{\pi} p(x)\,\tfrac{1}{2}\,dx = 0$$

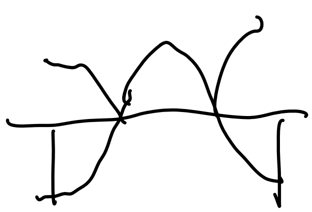
$$\int_{-\pi}^{\pi} f(x)\,dx = \int_{-\pi}^{\pi} \frac{a_0}{2}\,dx + \int_{-\pi}^{\pi} a_1\cos(x)\,dx$$

$$+ \int_{-\pi}^{\pi} b_1\sin(x)$$

$$4 = \frac{a_0}{2}\cdot 2\pi \qquad a_0 = \frac{4}{\pi}$$

$$\langle f(x) - p(x), \cos(x) \rangle = 0$$

$$\int_{-\pi}^{\pi} |\cos(x)|\cos(x)\,dx = \int_{-\pi}^{\pi} \frac{a_0}{2}\cos(x)\,dx$$



$$+ a_1\int_{-\pi}^{\pi} \frac{1+\cos(2x)}{2}\,dx + b_1\int_{-\pi}^{\pi} \sin x\cos x\,dx$$

$$2\cdot\int_{\frac{\pi}{2}}^{\pi} -\cos^2 x\,dx + \int_{-\frac{\pi}{2}}^{\frac{\pi}{2}} \cos^2 x\,dx \quad = a_1\left(\frac{1}{2}x + \frac{\sin 2x}{4}\right)\Bigg|_{-\pi}^{\pi} + b_1\,\frac{1}{4}\cos 2x\,\Bigg|_{-\pi}^{\pi}$$

$0 = a_1 \cdot \pi \Rightarrow a_1 = 0.$

$\langle f(x) - p(x), \sin(x) \rangle$

$$\int_{-\pi}^{\pi} |\cos(x)| \sin(x)\, dx = \int_{-\pi}^{\pi} \frac{a_0}{2} \sin(x)\, dx$$

$$+ a_1 \int_{-\pi}^{\pi} \cos(x) \sin(x)\, dx + b_1 \int_{-\pi}^{\pi} \sin^2(x)\, dx$$

$\cos^2 - \cos 2x$

$\cos 2x = 1 - \sin^2 - \sin^2 \qquad \frac{1 - \cos 2x}{2}$

$$\int_{-\pi}^{\pi} |\cos(x)| \sin(x)\, dx = 0 + 0 + b_1 \int_{-\pi}^{\pi} \sin^2(x)\, dx$$

$$\int_{-\frac{\pi}{2}}^{\frac{\pi}{2}} \frac{1}{2} \sin 2x\, dx + \int_{\frac{\pi}{2}}^{\pi} -\frac{1}{2} \sin 2x\, dx + \int_{-\pi}^{-\frac{\pi}{2}} -\frac{1}{2} \sin 2x\, dx$$

$$= \frac{1}{4} - \cos 2x\, dx \Big|_{-\frac{\pi}{2}}^{\frac{\pi}{2}} + 2 + (-2)$$

$= 0.$ Thus $b_1 = 0$

7. (15 points) Suppose P_{n-2} is the polynomial of degree $n-2$ such that

$$\int_{-1}^{1}(x-1)(x+1)Q(x)P_{n-2}(x)dx = 0$$

for all Q which is a polynomial of degree at most $n-3$. We denote the roots of P_{n-2} as $x_2, x_3, \cdots, x_{n-1}$. We define $x_1 = -1$, $x_n = 1$ and

$$c_i = \int_{-1}^{1} \prod_{j=1, i\neq j}^{n} \frac{x - x_j}{x_i - x_j} dx.$$

Prove that if $P(x)$ is any polynomial of degree at most $2n-3$, then

$$\int_{-1}^{1} P(x)dx = \sum_{i=1}^{n} c_i P(x_i).$$

P_n is a polynomial of degree n.

such that $\int_{-1}^{1} (x+1)(x-1) P_n Q \, dx = 0$.

P is a poly of degree at most $n-1$.

$P = \sum_{i=1}^{n} P_i L_i(x) \Rightarrow \int_{-1}^{1} P dx = \sum P_i(x_i) \int_{-1}^{1} L_i(x) dx$

next conside $2n-3$ We hae.

$P = (x+1)(x-1) P_{n-2} Q + R$.

where R is a poly of deg $\leq n-1$.

Q is a poly of deg $\leq n-3$.

$\int_{-1}^{1} P dx = \int_{-1}^{1} (x+1)(x-1) P_{n-2} Q \, dx + \int_{-1}^{1} R dx$

$$= \int_{-1}^{1} R\,dx$$

$$= \sum_{i=1}^{n} P_i(x_i) \int_{-1}^{1} L_i(x)$$

$$= \sum_{i=1}^{n} c_i\, P_i(x_i)$$

8. (15 points) Given a function $f \in C^2([-1,1])$ and $\{x_i\}_{i=0}^{m-1}$ where $x_i = -1 + (i + \frac{1}{2})h$ with $h = \dfrac{2}{m}$ for $m > 1$, we consider $P(x) = a_0 + a_1 x$ and $\tilde{P}_m(x) = \tilde{a}_0 + \tilde{a}_1 x$ are the least square polynomial minimizing the continuous square error and the discrete square error respectively. Namely,

$$\int_a^b |f(x) - P(x)|^2 dx = \min_{\hat{P}(x) = \hat{a}_0 + \hat{a}_1 x} \{ \int_{-1}^1 |f(x) - \hat{P}(x)|^2 dx \}$$

and

$$\sum_{i=0}^{m-1} |f(x_i) - \tilde{P}_m(x_i)|^2 = \min_{\hat{P}(x) = \hat{a}_0 + \hat{a}_1 x} \{ \sum_{i=0}^{m-1} |f(x_i) - \hat{P}(x_i)|^2 dx \}.$$

(a) (5 points) Prove that

$$\frac{2}{m} \sum_{i=0}^{m-1} x_i^k x_i^j = \begin{cases} 0 & \text{for } k \neq j \\ \tilde{\alpha}_j > 0 & \text{for } k = j \end{cases}$$

for $j = 0, 1$ and $k = 0, 1$.

(b) (10 points) Prove that

$$\lim_{m \to \infty} \int_{-1}^1 |P(x) - \tilde{P}_m(x)| dx = 0.$$

① $f(x_0+h) = f(x_0) + f'(x_0)h + \frac{h^2}{2}f''(x_0) + \frac{h^3}{6}f^{(3)}(x_0) + \frac{h^4}{4!}f(x_0)$

② $f(x_0-h) = f(x_0) - f'(x_0)h + \frac{h^2}{2}f''(x_0) - \frac{h^3}{6}f^{(3)}(x_0) + \frac{h^4}{4!}f(x_0)$

$+ \frac{h^5}{5!}f(x_0)$

① − ②

$f(x_0+h) - f(x_0-h) = 2f'(x_0)h + \frac{h^3}{3}f^{(3)}(x_0) + \frac{h^5}{5!}f(x_0)$

$f'(x_0) = \frac{f(x_0+h) - f(x_0-h)}{2h} - \frac{h^2}{6}f^{(3)}(x_0) - \frac{h^4}{5!\cdot 2}f^{(5)}(x)$