Interpolation

Ágnes Baran, Csaba Noszály

Lagrangian-interpolation

The problem:

We have n observations (measurements,data)

$$f_0, f_1, \ldots, f_n$$

in the pairwise different points

$$x_0, x_1, \ldots, x_n$$

. We are searching for a minimal degree φ polynomial, for which

$$\varphi(x_i) = f_i, \quad i = 0, \ldots, n$$

Proposition

There is exactly one polynomial of degree at most n, which fulfills the conditions

$$\varphi(x_i) = f_i, \quad i = 0, \ldots, n$$

Proof:

existence by construction

Let

$$\ell_i(x) = \frac{(x-x_0)(x-x_1)\cdots(x-x_{i-1})(x-x_{i+1})\cdots(x-x_n)}{(x_i-x_0)(x_i-x_1)\cdots(x_i-x_{i-1})(x_i-x_{i+1})\cdots(x_i-x_n)}$$

that is

$$\ell_i(x) = \prod_{\substack{j=0\\ j\neq i}}^{n} \frac{x-x_j}{x_i-x_j}, \quad i = 0,\ldots,n$$

$\ell_i(x)$ a polynomial of degree n and

$$\ell_i(x_k) = \begin{cases} 0, & \text{if } k \neq i, \\ 1, & \text{if } k = i \end{cases}$$

By setting

$$\varphi(x) = \sum_{i=0}^{n} f_i \ell_i(x)$$

one can see, that φ fulfills the requirements.

uniqueness

Assume that $\varphi(x)$ and $\psi(x)$ are of degree at most n interpolating polynomials. Then for the difference

$$\Phi(x) = \varphi(x) - \psi(x)$$

we have $\Phi(x_i) = 0 \quad i = 0, \ldots, n$ and Φ is of degree at most n. But according to the main theorem of algebra, a nonzero polynomial of degree n has at most n (real) roots, so $\Phi(x) \equiv 0$

Defining the Lagrange-polynomial recursively (Newton-form)

Denote by $L_k(x)$ the Lagrange-polynomial of (x_0, f_0), (x_1, f_1),..., (x_k, f_k).

- For 1 point, (x_0, f_0):

  $L_0(x) \equiv f_0$

- For 2 points, (x_0, f_0), (x_1, f_1), we are looking for L_1 in the form:

  $L_1(x) = L_0(x) + b_1(x - x_0)$,

  which can be solved for b_1:

$$L_1(x_1) = f_1$$
$$L_1(x_1) = f_0 + b_1(x_1 - x_0) \implies$$
$$b_1 = \frac{f_1 - f_0}{x_1 - x_0}$$

- For 3 points, (x_0, f_0), (x_1, f_1), (x_2, f_2):

$$L_2(x_2) = f_2$$

$$L_2(x_2) = L_1(x_2) + b_2(x - x_0)(x - x_1) \implies$$

$$b_2 = \frac{f_2 - L_1(x_2)}{(x_2 - x_0)(x_2 - x_1)}$$

$$b_2 = \frac{1}{x_2 - x_0}\left(\frac{f_2 - f_1}{x_2 - x_1} - \frac{f_1 - f_0}{x_1 - x_0}\right)$$

- For $k+1$ points, (x_0, f_0), (x_1, f_1),..., (x_k, f_k):

$$L_k(x) = L_{k-1}(x) + b_k\omega_k(x)$$

with $\omega_k(x) = (x - x_0) \cdot \ldots \cdot (x - x_{k-1})$ Then solve for b_k:

$$b_k = (f_k - L_{k-1}(x_k))/\omega_k(x_k)$$

We need a simpler, organized way to compute the b_k-s.

Divided differences (DD)

For a given pairwise different $x_0, x_1, \dots, x_n$ points and $f_0, f_1, \dots, f_n$ values:

- $[x_i]f = f_i$ 0-th order DD.
- 1-st order DD of points x_i, x_{i+1}:

$$[x_i, x_{i+1}]f := \frac{f_{i+1} - f_i}{x_{i+1} - x_i}$$

- k-th order DD of points $x_i, \dots, x_{i+k}$

$$[x_i, \dots, x_{i+k}]f = \frac{[x_{i+1}, \dots, x_{i+k}]f - [x_i, \dots, x_{i+k-1}]f}{x_{i+k} - x_i}$$

Proposition

In the Newton-form of the Lagrange-polynomial we have:

$$b_k = [x_0, \dots, x_k]f$$

Computational scheme

$$
\begin{array}{c|ccccc}
x_0 & [x_0]f \searrow & & & & \\
 & & [x_0,x_1]f \searrow & & & \\
x_1 & [x_1]f \nearrow & & [x_0,x_1,x_2]f \searrow & & \\
 & \searrow & [x_1,x_2]f \nearrow & & \dots & \\
\vdots & & \vdots & & \searrow & \\
\vdots & & \vdots & & & [x_0,\dots,x_n]f \\
\vdots & & \vdots & & \dots \nearrow & \\
x_{n-1} & [x_{n-1}]f \searrow & & & & \\
 & & [x_{n-1},x_n]f \nearrow & & & \\
x_n & [x_n]f \nearrow & & & &
\end{array}
$$

$$
\begin{aligned}
L_n(x) = {} & [x_0]f + [x_0,x_1]f \cdot (x-x_0) + [x_0,x_1,x_2]f \cdot (x-x_0)(x-x_1) \\
& + \dots + [x_0,\dots,x_n]f \cdot (x-x_0)\cdots(x-x_{n-1})
\end{aligned}
$$

Example

Compute the Lagrange interpolation polynom for the point set:

$$(-2,-31),(-1,-7),(0,-1),(2,5)$$

-2	-31			
		24		
-1	-7		-9	
		6		2
0	-1		-1	
		3		
2	5			

$$L_3(x) = -31 + 24(x+2) - 9(x+2)(x+1) + 2(x+2)(x+1)x$$

Remark

The Lagrange polynomial does not depend on the order of the data points, therefore we can also choose the lower edge of the table:

-2	-31			
		24		
-1	-7		-9	
		6		2
0	-1		-1	
		3		
2	5			

$$L_3(x) = 5 + 3(x-2) - 1 \cdot (x-2)x + 2(x-2)x(x+1)$$

The result is:

$$L_3(x) = 2x^3 - 3x^2 + x - 1$$

in both cases.

Example

Find the minimal degree polinomial that fits the data points:

$$(-2,-5),(-1,3),(1,-5),(2,-9)$$

$$\begin{array}{r|rrrr}
-2 & -5 & & & \\
 & & 8 & & \\
-1 & 3 & & -4 & \\
 & & -4 & & 1 \\
1 & -5 & & 0 & \\
 & & -4 & & \\
2 & -9 & & &
\end{array}$$

$$L_3(x) = -5 + 8(x+2) - 4(x+2)(x+1) + (x+2)(x+1)(x-1)$$

Example

Compute the Lagrange polynomial which fits the previous data and the new data point (0, 9) as well

-2	-5			
		8		
-1	3		-4	
		-4		1
1	-5		0	
		-4		
2	-9			

Extend the "old" table with the new data, and compute the new divided differences:

-2	-5				
		8			
-1	3		-4		
		-4		1	
1	-5		0		2
		-4		5	
2	-9		5		
		-9			
0	9				

$$L_4(x) = L_3(x) + 2(x+2)(x+1)(x-1)(x-2)$$

Exercise 1

Find the Lagrange polynomial:

(a) $(-3,-6)$, $(-2,-17)$, $(-1,-8)$, $(1,-2)$, $(2,19)$,

(b) $(-3,-31)$, $(-2,-8)$, $(1,1)$, $(2,22)$,

(c) $(-2,-13)$, $(-1,-4)$, $(1,2)$,

(d) $(-2,-5)$, $(-1,3)$, $(0,1)$, $(2,15)$,

(e) $(-1,4)$, $(1,2)$, $(2,10)$, $(3,40)$,

(f) $(-2,38)$, $(-1,5)$, $(1,-1)$, $(2,-10)$, $(3,-7)$,

(g) $(-2,-33)$, $(-1,-2)$, $(1,6)$, $(2,7)$, $(3,-18)$,

(h) $(-3,-209)$, $(-2,-43)$, $(-1,-1)$, $(1,-1)$, $(2,-19)$.

Horner's algorithm

Let

$$p(x) = a_n x^n + a_{n-1} x^{n-1} + \cdots + a_1 x + a_0$$

where $a_n \neq 0$. For a given $x^\star \in \mathbb{R}$, compute $p(x^\star)$!
We can rearrange the operations

$$p(x^\star) = (((\cdots(a_n x^\star + a_{n-1})x^\star + \cdots)x^\star + a_2)x^\star + a_1)x^\star + a_0$$

which leads us to the following algorithm:

$$\begin{aligned} c_0 &= a_n \\ c_1 &= c_0 x^\star + a_{n-1} \\ c_2 &= c_1 x^\star + a_{n-2} \\ &\vdots \\ c_n &= c_{n-1} x^\star + a_0 = p(x^\star) \end{aligned}$$

In tabular form:

	a_n	a_{n-1}	$\cdots$	a_2	a_1	a_0
$x^\star$	c_0	c_1	$\cdots$	c_{n-2}	c_{n-1}	c_n

$p(x^\star) = c_n$

Example

$$p(x) = 2x^5 + 3x^4 - 3x^2 + 5x - 1, \qquad p(-2) = ?$$

	2	3	0	-3	5	-1
-2	2	-1	2	-7	19	-39

$p(-2) = -39$

Generalized Horner scheme

$$L_n(x) = b_0 + b_1 \cdot (x - x_0) + b_2 \cdot (x - x_0)(x - x_1) + \\ + \dots + b_n \cdot (x - x_0)(x - x_1)(x - x_{n-1})$$

where $b_k = [x_0, \dots, x_k]f$. $L_n(x^\star) = ?$

$$\begin{aligned} c_0 &= b_n \\ c_1 &= c_0(x^\star - x_{n-1}) + b_{n-1} \\ c_2 &= c_1(x^\star - x_{n-2}) + b_{n-2} \\ &\vdots \\ c_n &= c_{n-1}(x^\star - x_0) + b_0 = L_n(x^\star) \end{aligned}$$

Remark

Ususally we want the values of the Lagrange polynomial not the coefficients. The generalized Horner method helps us to avoid the transformation of the polynomial to classical form.

Computing Lagrange polynomial in Octave/Matlab

The function `polyfit`:

The call `polyfit(x,f,n-1)` returns the coefficient of the minimal degree polinomial,

$$p(1)x^{n-1} + p(2)x^{n-2} + \ldots + p(n-1)x + p(n)$$

which fits the data points: (x_i, f_i), $i = 1, \ldots, n$.

Example

Find the Lagrange polinomial that passes through the points:

$$(-2,-5),(-1,3),(0,1),(2,15)$$

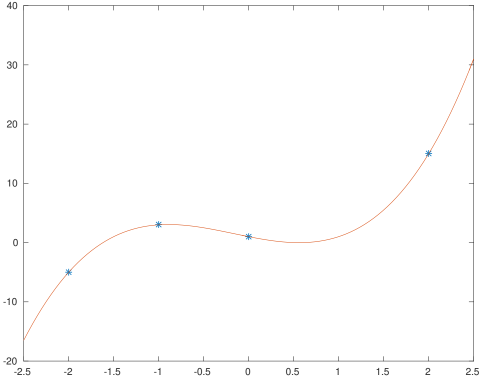
Solution

```
>>x=[-2, -1, 0, 2];
>>f=[-5, 3, 1, 15];
>>p=polyfit(x,f,3)

p=

2.0000 1.0000 -3.0000 1.0000
```

Plot the data and the polynomial!

```
x=[-2, -1, 0, 2];
f=[-5, 3, 1, 15];
p=polyfit(x,f,3);
xx=linspace(-2.5,2.5);
yy=polyval(p,xx);
figure; plot(x,f,'*',xx,yy)
```

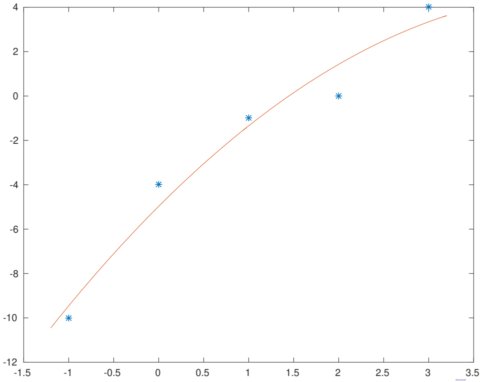
The call `yy=polyval(p,xx)` evaluates the polinomial *p* at the points given in vector *xx*.


40
30
20
10
0
-10
-20
-2.5
-2
-1.5
-1
-0.5
0
0.5
1
1.5
2
2.5

Important

The function `polyfit` can be used to find the LSA of data points, which is usually not fit tha data.

```
x=[-1 0 1 2 3]; f=[-10 -4 -1 0 4]; p=polyfit(x,f,2);
xx=linspace(-1.2,3.2); ff=polyval(p,xx);
figure; plot(x,f,'*',xx,ff)
```

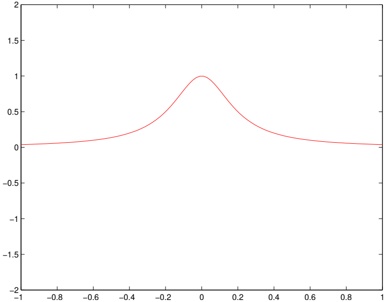


Remark

The error does not necessarily reduces by increasing the number of the points. Sometimes it might became arbitrarily large.

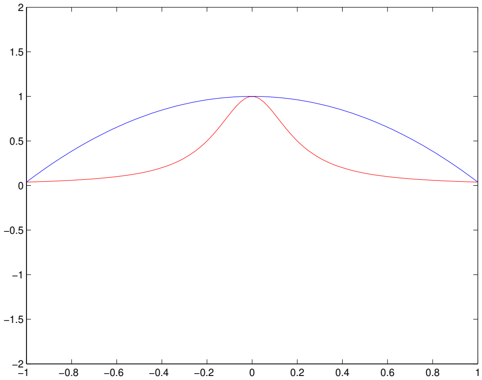
Example

Consider the function $f(x) = \frac{1}{1+25x^2}$ over $[-1, 1]$

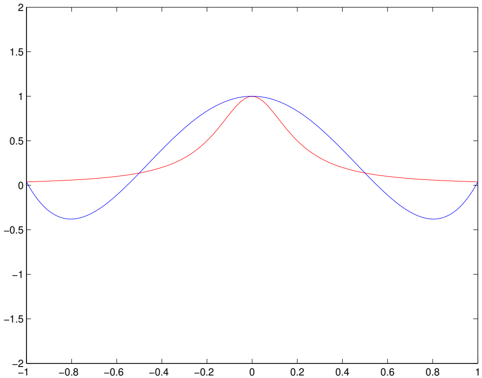


We are going to interpolate it with increasing number of equidistant base points.

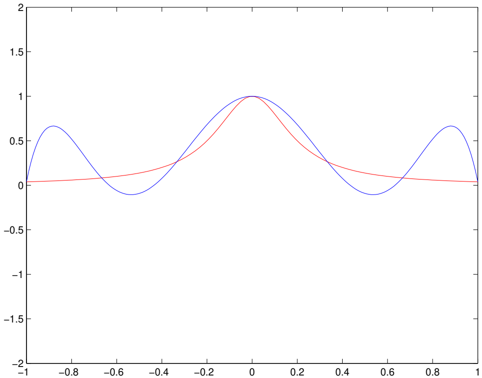
Lagrange interpolation, $n = 2$



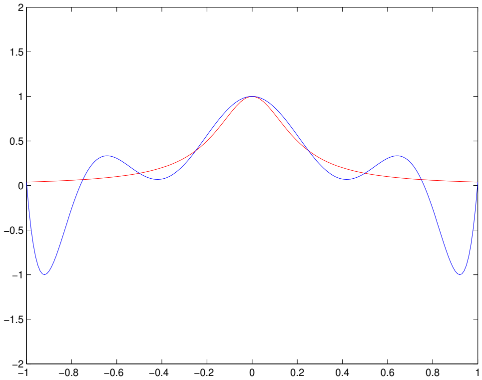
Lagrange interpolation, $n = 4$



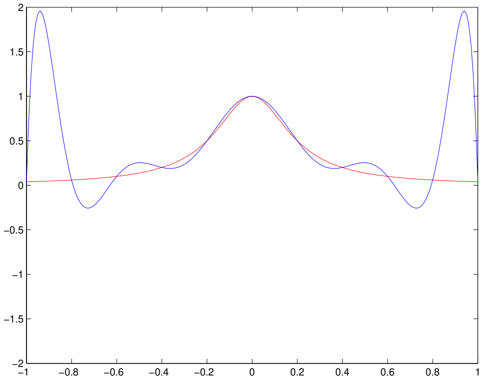
Lagrange interpolation, $n = 6$



Lagrange interpolation, $n = 8$



Lagrange interpolation, $n = 10$



Exercise 2

Plot the following 3 functions in one figure:

- the function

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

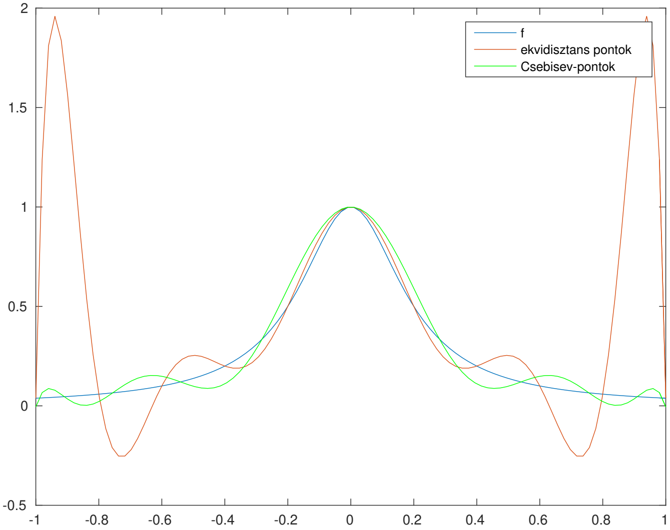
  on the interval $[-1, 1]$
- the Lagrange polinomial of f at the points

  $$-1, -0.8, -0.6, ..., 0.6, 0.8, 1$$

- the Lagrange polinomial of f at the points

  $$x_k = \cos\left(\frac{2k-1}{22}\pi\right), \quad k = 1, 2, \ldots, 11$$

  (these are the so called Csebisev-points)


f
ekvidisztans pontok
Csebisev-pontok
2
1.5
1
0.5
0
-0.5
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6
0.8
1

Remark

The Csebisev points on $[-1, 1]$

$$x_k = \cos\left(\frac{2k-1}{2n}\pi\right), \quad k = 1, 2, \ldots, n$$

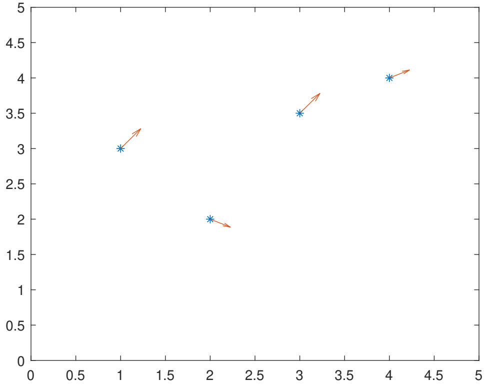
are the optimal, that is for a fixed $f : [-1, 1] \to \mathbb{R}$ and n the error

$$\max_x |f(x) - P(x)|$$

is minimal, where P is the Lagrange polinomial of $(x_k, f(x_k))$ and x_k's are the Csebisev points.

Hermite-interpolation

Sometimes we want to control the values of derivatives too. We need a polynomial that passes through the points in a prescribed direction.



Hermite-interpolation

The problem:

We have a set of points and a set of conditions at the points:

x_0	x_1	x_2	$\dots$	x_n
f_{00}	f_{10}	f_{20}	$\dots$	f_{n0}
f_{01}	f_{11}	f_{21}	$\dots$	f_{n1}
f_{02}	f_{12}	f_{22}	$\dots$	f_{n2}
$\vdots$				
f_{0,m_0-1}	f_{1,m_1-1}	f_{2,m_2-1}	$\dots$	f_{n,m_n-1}

We are looking for a polynomial $H(x)$, for which

$$H^{(j)}(x_i) = f_{ij}, \qquad i = 0, 1, \dots, n, \quad j = 0, \dots, m_i - 1.$$

Let $m = \sum_{i=0}^{n} m_i$, the number of conditions.

Proposition

The task of Hermite-interpolation can be solved uniquely in the space of at most $m-1$ degree polynomials.

Example

Find the Hermite-polynomial of the given data!

x_i	-2	-1	1
$f(x_i)$	1	6	-2
$f'(x_i)$	74	-12	-4
$f''(x_i)$		16	

Tha number of conditions $m = 7$, therefore the desired polynomial is of degree at most 6.

Tha data:

x_i	-2	-1	1
$f(x_i)$	1	6	-2
$f'(x_i)$	74	-12	-4
$f''(x_i)$		16	

-2	1		
		74	
-2	1		
-1	6		
		-12	
-1	6		8
		-12	
-1	6		
1	-2		
		-4	
1	-2		

Let us compute the missing values:

-2	1		
		74	
-2	1		
-1	6		
		-12	
-1	6		8
		-12	
-1	6		
1	-2		
		-4	
1	-2		

The first order divided differences:

-2	1		
		74	
-2	1		
		5	
-1	6		
		-12	
-1	6		8
		-12	
-1	6		
		-4	
1	-2		
		-4	
1	-2		

The second order DD's:

-2	1		
		74	
-2	1		-69
		5	
-1	6		-17
		-12	
-1	6		8
		-12	
-1	6		4
		-4	
1	-2		0
		-4	
1	-2		

The 3-rd order DD's:

-2	1			
		74		
-2	1		-69	
		5		52
-1	6		-17	
		-12		25
-1	6		8	
		-12		-2
-1	6		4	
		-4		-2
1	-2		0	
		-4		
1	-2			

The 4-th order DD's:

-2	1				
		74			
-2	1		-69		
		5		52	
-1	6		-17		-27
		-12		25	
-1	6		8		-9
		-12		-2	
-1	6		4		0
		-4		-2	
1	-2		0		
		-4			
1	-2				

The 5-th order ones:

-2	1					
		74				
-2	1		-69			
		5		52		
-1	6		-17		-27	
		-12		25		6
-1	6		8		-9	
		-12		-2		3
-1	6		4		0	
		-4		-2		
1	-2		0			
		-4				
1	-2					

The 6-th order DD's:

-2	1						
		74					
-2	1		-69				
		5		52			
-1	6		-17		-27		
		-12		25		6	
-1	6		8		-9		-1
		-12		-2		3	
-1	6		4		0		
		-4		-2			
1	-2		0				
		-4					
1	-2						

-2	**1**						
		74					
-2	1		**-69**				
		5		**52**			
-1	6		-17		**-27**		
		-12		25		**6**	
-1	6		8		-9		**-1**
		-12		-2		3	
-1	6		4		0		
		-4		-2			
1	-2		0				
		-4					
1	-2						

$$
\begin{aligned}
H(x) = & \mathbf{1} + \mathbf{74}(x+2) - \mathbf{69}(x+2)^2 + \mathbf{52}(x+2)^2(x+1) \\
& -\mathbf{27}(x+2)^2(x+1)^2 + \mathbf{6}(x+2)^2(x+1)^3 \\
& -\mathbf{1}(x+2)^2(x+1)^3(x-1)
\end{aligned}
$$

Exercise 3

Find the minimal degree polynomial that fits the given data!

(a)

x_i	-1	1
$f(x_i)$	7	3
$f'(x_i)$	-8	-4

(b)

x_i	-1	1
$f(x_i)$	3	1
$f'(x_i)$	9	-7
$f''(x_i)$		-18

(c)

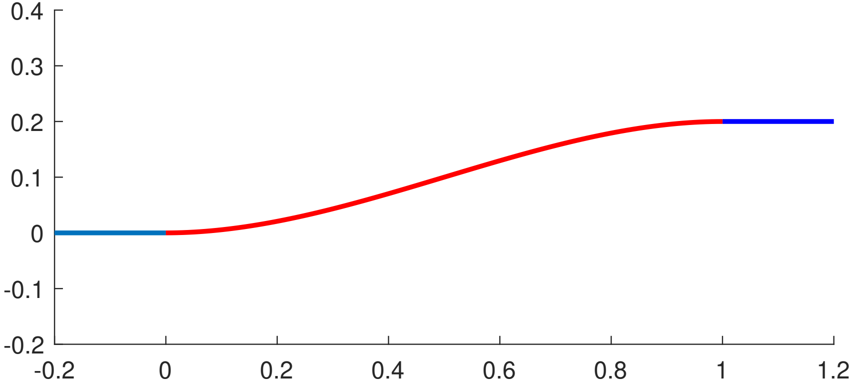
x_i	-1	1	2
$f(x_i)$	4	6	94
$f'(x_i)$	9	17	213

(d)

x_i	-2	-1	1
$f(x_i)$	13	3	7
$f'(x_i)$	-31	14	18
$f''(x_i)$		-40	

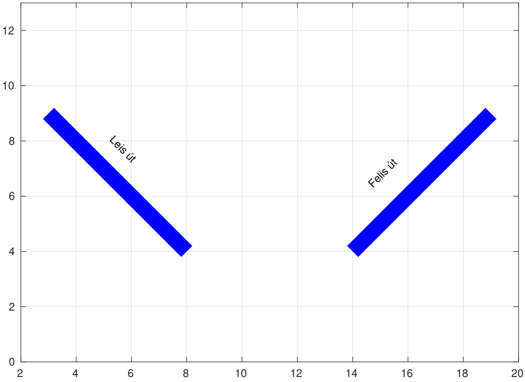
Exercise 4

We are going to build a ramp that connects the street and the port of a garage. The difference between the level of the street and the entry is 0.2m. The distance between them is 1m. Design the ramp that let us drive as smooth as possible to and from the garage.



Exercise 5

Connnect the lower end points of the segments given in the figure below with a smooth curve! (one without abrupt bends)



Applications

Example

For a given real valued, differentiable function find the equation of its tangential line at x_0!

We are looking for a Hermite polynomial of degree at most 1, for which $H(x_0) = f(x_0)$ and $H'(x_0) = f'(x_0)$

$$\begin{array}{c|cc} x_0 & f(x_0) & \\ & & f'(x_0) \\ x_0 & f(x_0) & \end{array}$$

$$H(x) = f(x_0) + f'(x_0)(x - x_0)$$

Example

Find the Hermite polynomial that fulfills the conditions:

$$x_0, f(x_0), f'(x_0), f''(x_0), \ldots, f^{(n)}(x_0)$$

$$
\begin{array}{c|cccccc}
x_0 & f(x_0) & & & & & \\
 & & f'(x_0) & & & & \\
x_0 & f(x_0) & & \frac{f''(x_0)}{2!} & & & \\
 & & f'(x_0) & & \frac{f'''(x_0)}{3!} & & \\
x_0 & f(x_0) & & \frac{f''(x_0)}{2!} & & \ddots & \\
 & & f'(x_0) & & & & \frac{f^{(n)}(x_0)}{n!} \\
x_0 & f(x_0) & & & & & \\
\vdots & \vdots & & & & & \\
x_0 & f(x_0) & & & & & \\
 & & f'(x_0) & & & & \\
x_0 & f(x_0) & & & & &
\end{array}
$$

$$
\begin{aligned}
H(x) = & f(x_0) + f'(x_0)(x - x_0) + \frac{f''(x_0)}{2!}(x - x_0)^2 + \frac{f'''(x_0)}{3!}(x - x_0)^3 \\
& + \cdots + \frac{f^{(n)}(x_0)}{n!}(x - x_0)^n
\end{aligned}
$$

It is the Taylor polynomial of f around x_0.

Piecewise interpolation

Increasing the base points not neccessarily decreases the error.

Instead of fitting a high order polynomial, we divide the interval into smaller subintervals and we apply lower order interpolation on them.

Let us divide $[a, b]$ into m (equally-sized) subintervals:

$$a = x_0 < x_1 < x_2 < \cdots < x_m = b$$

Perform the Lagrange-interpolation (a linear one) on each of the pieces! It is the so called piecewise-linear interpolation.

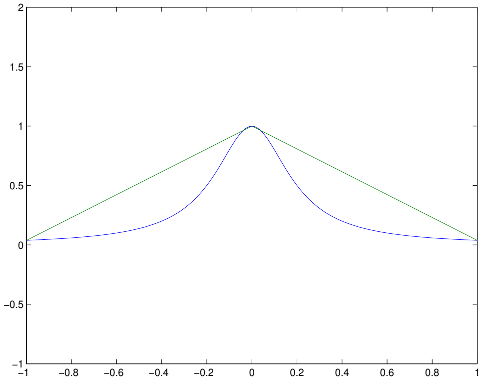
Error of the linear interpolation

If $h := x_i - x_{i-1}$, $i = 1, \ldots, m$, and $f \in C^2[a, b]$ then for the piecewise interpolation $L_{m\times 1}(x)$ of f we have:

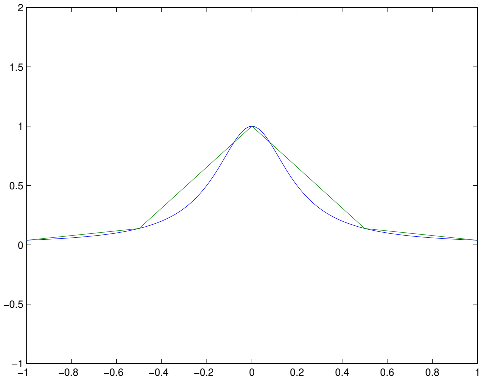
$$|f(x) - L_{m\times 1}(x)| \leq \frac{M_2}{8} h^2, \qquad x \in [a, b]$$

where $M_2 = \max_{x\in[a,b]} |f''(x)|$.
Increasing the number of subintervals results in smaller error.

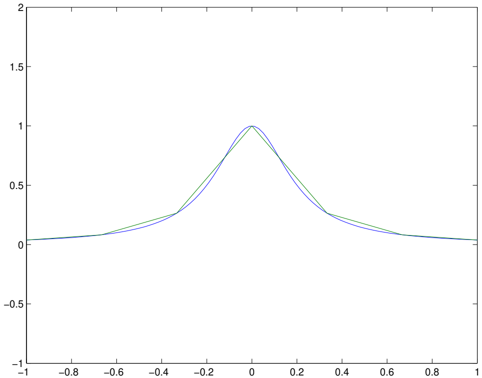
Piecewise linear interpolation, 2 subintervals:



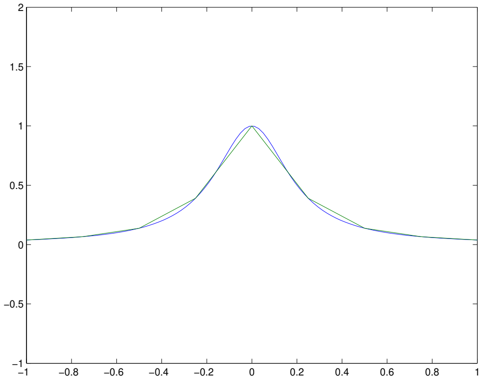
Piecewise linear interpolation, 4 subintervals:



Piecewise linear interpolation, 6 subintervals:



Piecewise linear interpolation, 8 subintervals:



Piecewise Hermite-interpolation

The piecewise linear interpolation is continous, but not differentiable at the base points.

We can get a continuously differentiable interpolation function if we prescribe the values of the derivatives at the nodes.

In this case we know the following data in the $[x_{i-1}, x_i]$ interval

x_{i-1}	x_i
$f(x_{i-1})$	$f(x_i)$
$f'(x_{i-1})$	$f'(x_i)$

There are 4 data points, therefore the degree of the polynomial cannot be higher than 3.

Example

Calculate the piecewise third order polynomial

$$H(x) = \begin{cases} H_1(x), & \text{ha} \quad x \in [-1, 1] \\ H_2(x), & \text{ha} \quad x \in (1, 3] \end{cases}$$

which is contiuously differentiable and satisfies the conditions $H(-1) = 4$, $H(1) = 6$, $H(3) = 12$, $H'(-1) = -3$, $H'(1) = 13$, $H'(3) = 9$!

x_i	-1	1	3
$H(x)$	4	6	12
$H'(x)$	-3	13	9

-1	4			
		-3		
-1	4		2	
		1		2
1	6		6	
		13		
1	6			

x_i	-1	1	3
$H(x)$	4	6	12
$H'(x)$	-3	13	9

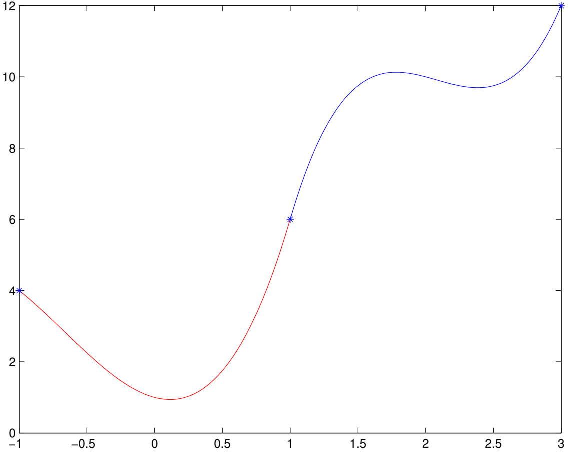
$$H_1(x) = 6 + 13(x-1) + 6(x-1)^2 + 2(x-1)^2(x+1)$$

1	6			
		13		
1	6		-5	
		3		4
3	12		3	
		9		
3	12			

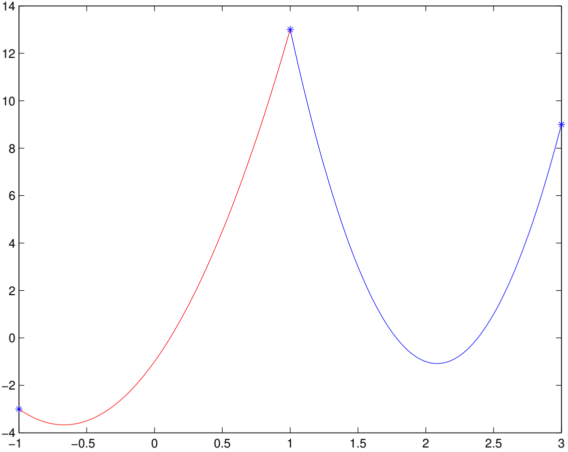
x_i	-1	1	3
$H(x)$	4	6	12
$H'(x)$	-3	13	9

$$H_2(x) = 6 + 13(x-1) - 5(x-1)^2 + 4(x-1)^2(x-3)$$

The shapes of the polynomials H_1 and H_2 are:



The plot of derivative of H_1 and H_2:



Cubic spline interpolation

If we demand the continuity of the first derivatives of the interpolating function at the common points of the subintervals, but do not fix their values, then we shall have one free parameter:

x_i	-1	1	3
$H(x)$	4	6	12
$H'(x)$	-3	α	9

$$
\begin{array}{r|lllll}
-1 & 4 & & & \\
 & & -3 & & \\
-1 & 4 & & 2 & \\
 & & 1 & & \frac{\alpha-5}{4} \\
1 & 6 & & \frac{\alpha-1}{2} & \\
 & & \alpha & & \\
1 & 6 & & &
\end{array}
\qquad
\begin{array}{r|lllll}
1 & 6 & & & \\
 & & \alpha & & \\
1 & 6 & & \frac{3-\alpha}{2} & \\
 & & 3 & & \frac{3+\alpha}{4} \\
3 & 12 & & 3 & \\
 & & 9 & & \\
3 & 12 & & &
\end{array}
$$

$$H_1(x) = 6 + \alpha(x-1) + \frac{\alpha - 1}{2}(x-1)^2 + \frac{\alpha - 5}{4}(x-1)^2(x+1)$$
$$H_2(x) = 6 + \alpha(x-1) + \frac{3 - \alpha}{2}(x-1)^2 + \frac{3 + \alpha}{4}(x-1)^2(x-3)$$

The extra free parameter allows us to put another condition, namely we can demand the continuity of the second derivative at the matching points.

$$H_1''(1) = 2\alpha - 6$$
$$H_2''(1) = -2\alpha$$

The $H_1''(1) = H_2''(1)$ implies

$$2\alpha - 6 = -2\alpha \quad \Rightarrow \quad \alpha = \frac{3}{2}$$

The idea of cubic spline-interpolation

We have the values:

x_0	x_1	x_2	$\dots$	x_{n-1}	x_n
$f(x_0)$	$f(x_1)$	$f(x_2)$	$\dots$	$f(x_{n-1})$	$f(x_n)$
$f'(x_0)$					$f'(x_n)$

We search for a function $S(x)$ for which

- $S(x_i) = f(x_i)$
- $S'(x_0) = f'(x_0)$ and $S'(x_n) = f'(x_n)$
- $S|_{[x_{i-1},x_i]} = S_i$ cubic polynomial, $i = 1, \dots, n$
- S is twice continuously differentiable

1. We introduce the unknowns $\alpha_1, \ldots, \alpha_{n-1}$:

x_0	x_1	x_2	$\ldots$	x_{n-1}	x_n
$f(x_0)$	$f(x_1)$	$f(x_2)$	$\ldots$	$f(x_{n-1})$	$f(x_n)$
$f'(x_0)$	α_1	α_2	$\ldots$	α_{n-1}	$f'(x_n)$

2. We calculate the third order $S_i(x)$ Hermite polynomial for $[x_{i-1}, x_i]$.

3. We write the conditions $S_i''(x_i) = S_{i+1}''(x_i)$ $(i = 1, \ldots, n-1)$. These represent a system of linear equations for the unknowns α_i.

4. We solve the system of linear equations (With a tridiagonal matrix.)

Spline interpolation in Octave/Matlab

Example

Find the cubic spline for the data below!

x_i	-2	-1	0	1	2	3
S	4	1	7	4	12	9
S'	15					8

Solution

Use the function `spline`! The call `p=spline(x,y)` returns the coefficients of the cubic spline. The parameter x is the base vector and y is the value vector.

```
>>x=-2:3; y=[15 4 1 7 4 12 9 8]; p=spline(x,y)
p =
      form: 'pp'
    breaks: [-2 -1 0 1 2 3]
     coefs: [5x4 double]
    pieces: 5
     order: 4
       dim: 1

>> p.coefs
ans =
   19.0000 -37.0000 15.0000 4.0000
  -12.0000 20.0000 -2.0000 1.0000
   11.0000 -16.0000 2.0000 7.0000
  -12.0000 17.0000 3.0000 4.0000
   15.0000 -19.0000 1.0000 12.0000
```

The 5 polynomials:

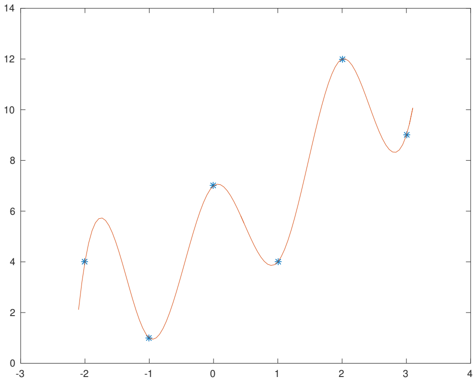
$$
\begin{aligned}
p_1(x) &= 19(x+2)^3 - 37(x+2)^2 + 15(x+2) + 4 \\
p_2(x) &= -12(x+1)^3 + 20(x+1)^2 - 2(x+1) + 1 \\
p_3(x) &= 11x^3 - 16x^2 + 2x + 7 \\
p_4(x) &= -12(x-1)^3 + 17(x-1)^2 + 3(x-1) + 4 \\
p_5(x) &= 15(x-2)^3 - 19(x-2)^2 + (x-2) + 12
\end{aligned}
$$

Verify the conditions!

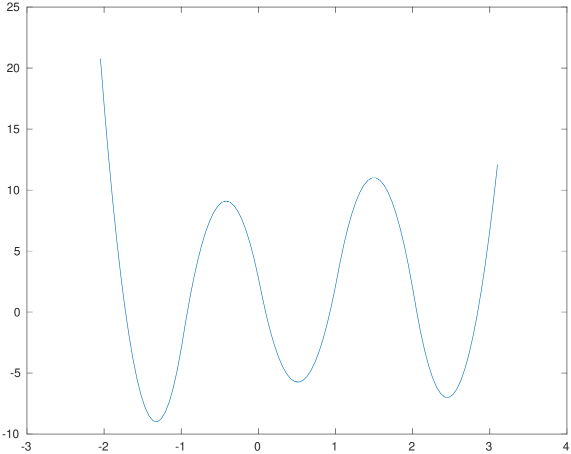
Usually we do not need the coefficients of the spline, we only need to know the values of the spline at some points:

```
>> x=-2:3;
>> y=[15 4 1 7 4 12 9 8];
>> xx=linspace(-2.1,3.1);
>> yy=spline(x,y,xx);
>> plot(x,y(2:end-1),'*',xx,yy)
```

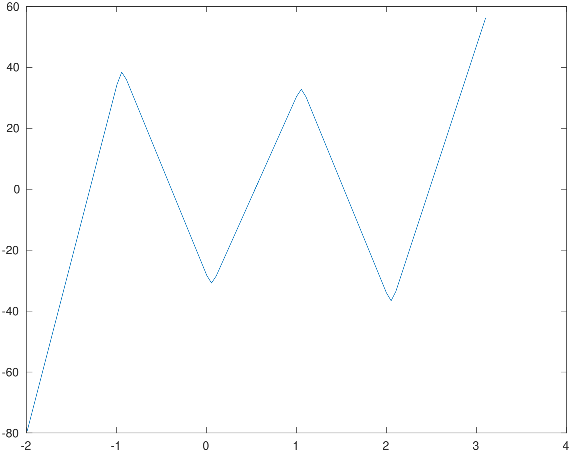
```
x=-2:3;
y=[15 4 1 7 4 12 9 8];
xx=linspace(-2.1,3.1);
yy=spline(x,y,xx);
plot(x,y(2:end-1),'*',xx,yy)
```



The 1-st derivative of previous spline:



And the second one:



It is continous, but not smooth.

Exercise 6

Plot the functions below in one figure!

- The function

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

  on the interval $[-1, 1]$

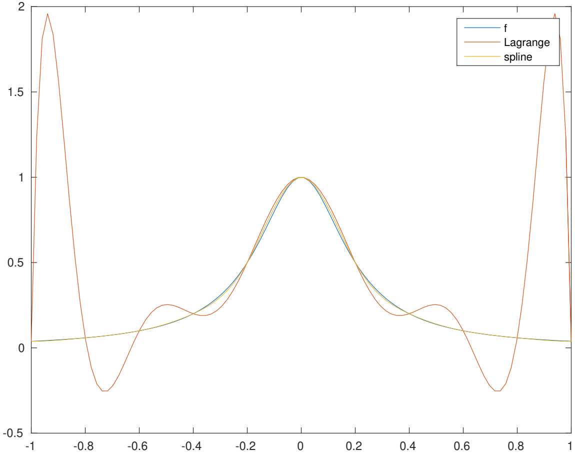
- The Lagrange polynomial of f at the nodes:

$$-1, -0.8, -0.6, ..., 0.6, 0.8, 1$$

- The az cubic spline of at the nodes:

$$-1, -0.8, -0.6, ..., 0.6, 0.8, 1$$

  At the end points set the derivative to zero.



f
Lagrange
spline
2
1.5
1
0.5
0
-0.5
-1
-0.8
-0.6
-0.4
-0.2
0
0.2
0.4
0.6
0.8
1

Exercise 7

We want to build a slide modeled in the figure. Because of physical limitations it will be built from two pieces. Design the shape of the slide! The shape must be smooth at the end points (A,C) and also at (B)!