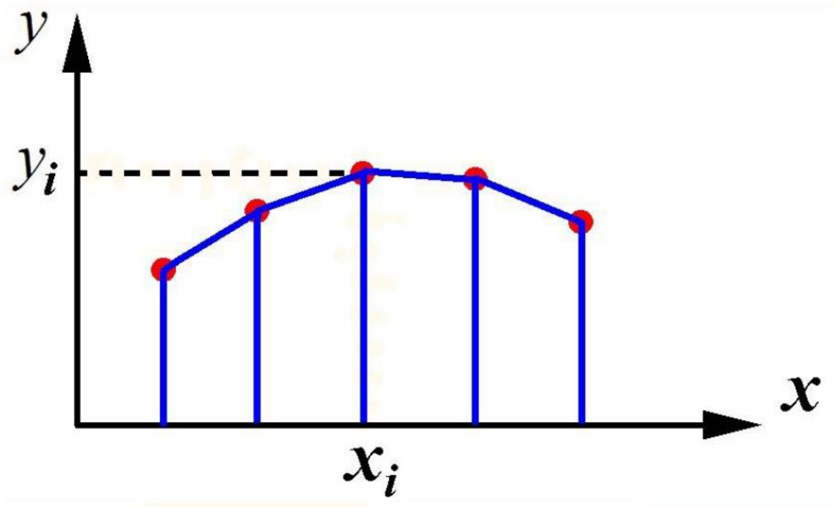
# Chapter 6. Interpolation by splines




recall approximation/interpolation
by the linear function:

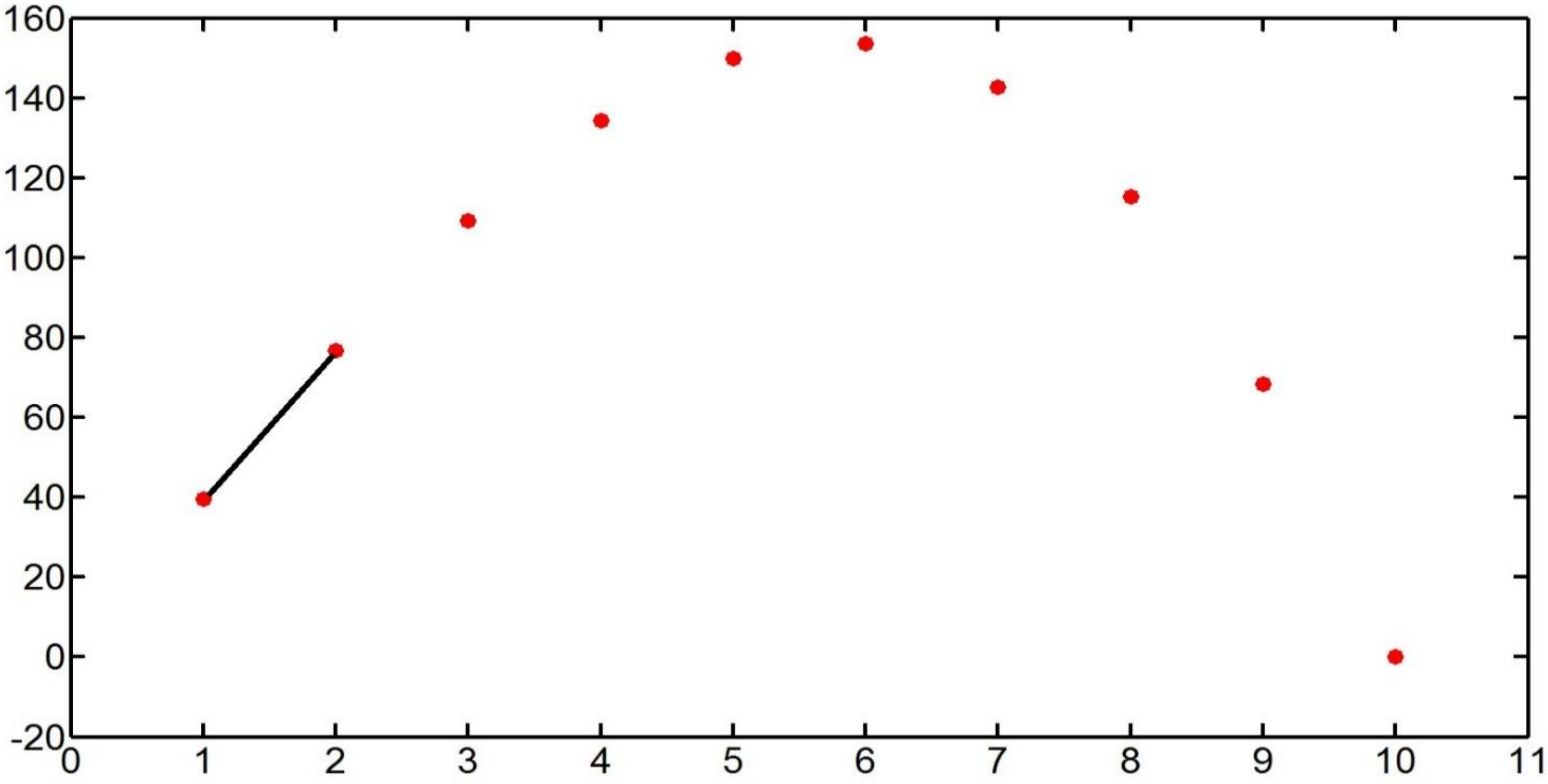
$$f(x) \approx y_i + (y_{i+1} - y_i)(x - x_i)/h$$

**No spikes, splashes in this case. However, a drawback:**
**there are jumps, discontinuities of the first-order derivative**

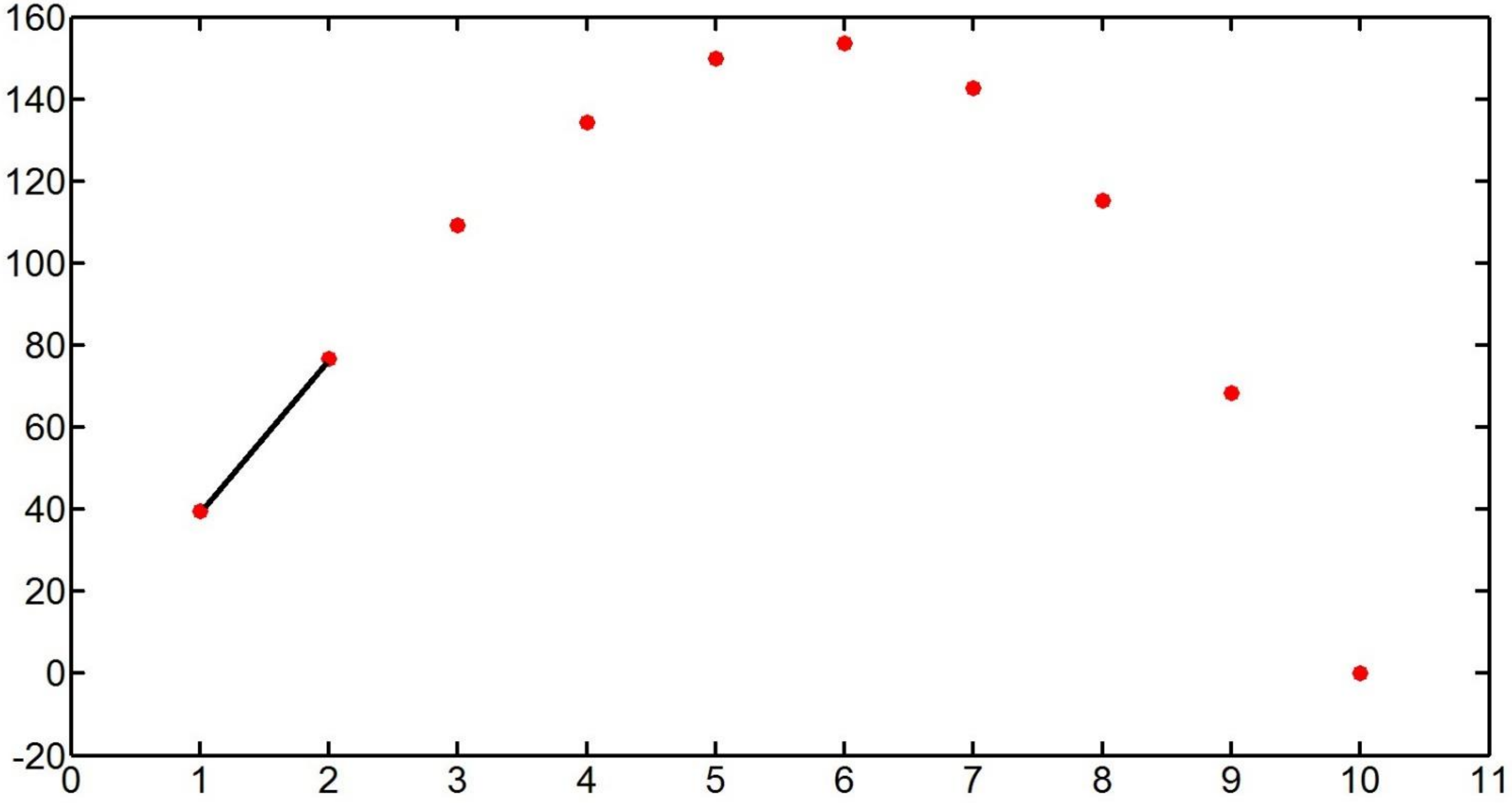
**which is** $(y_{i+1} - y_i)/h$ *when* $x_i < x < x_{i+1}$,

**and** $(y_{i+2} - y_{i+1})/h$ *when* $x_{i+1} < x < x_{i+2}$.

**That is why, it was suggested to use splines for interpolation purposes**



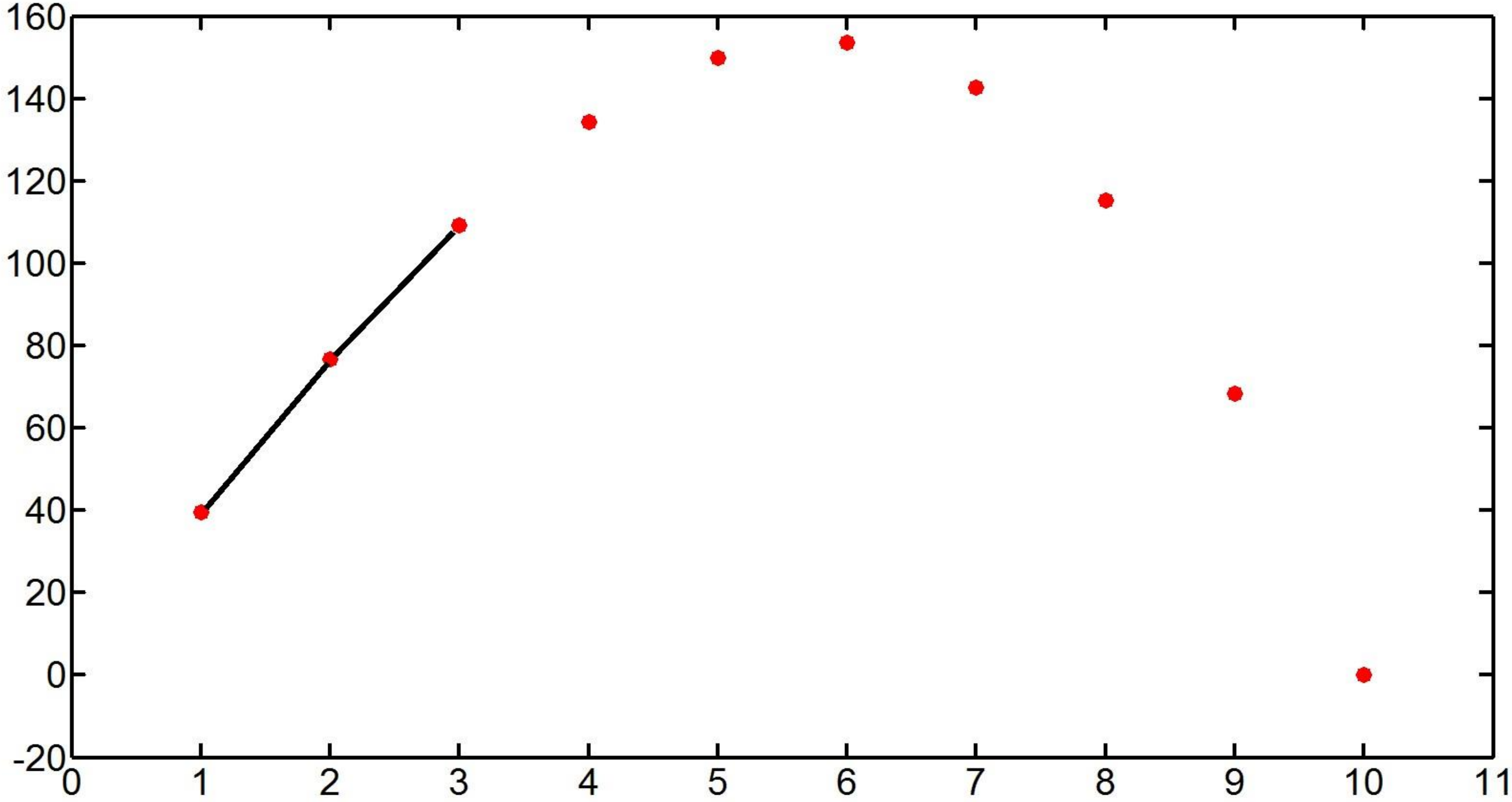

**Spline $S(x)$ is a train of linked parabolas of degree 2 or 3, whose derivative $dS/dx$ is continuous** everywhere, including points $x_i$
**(invented in 1946).**

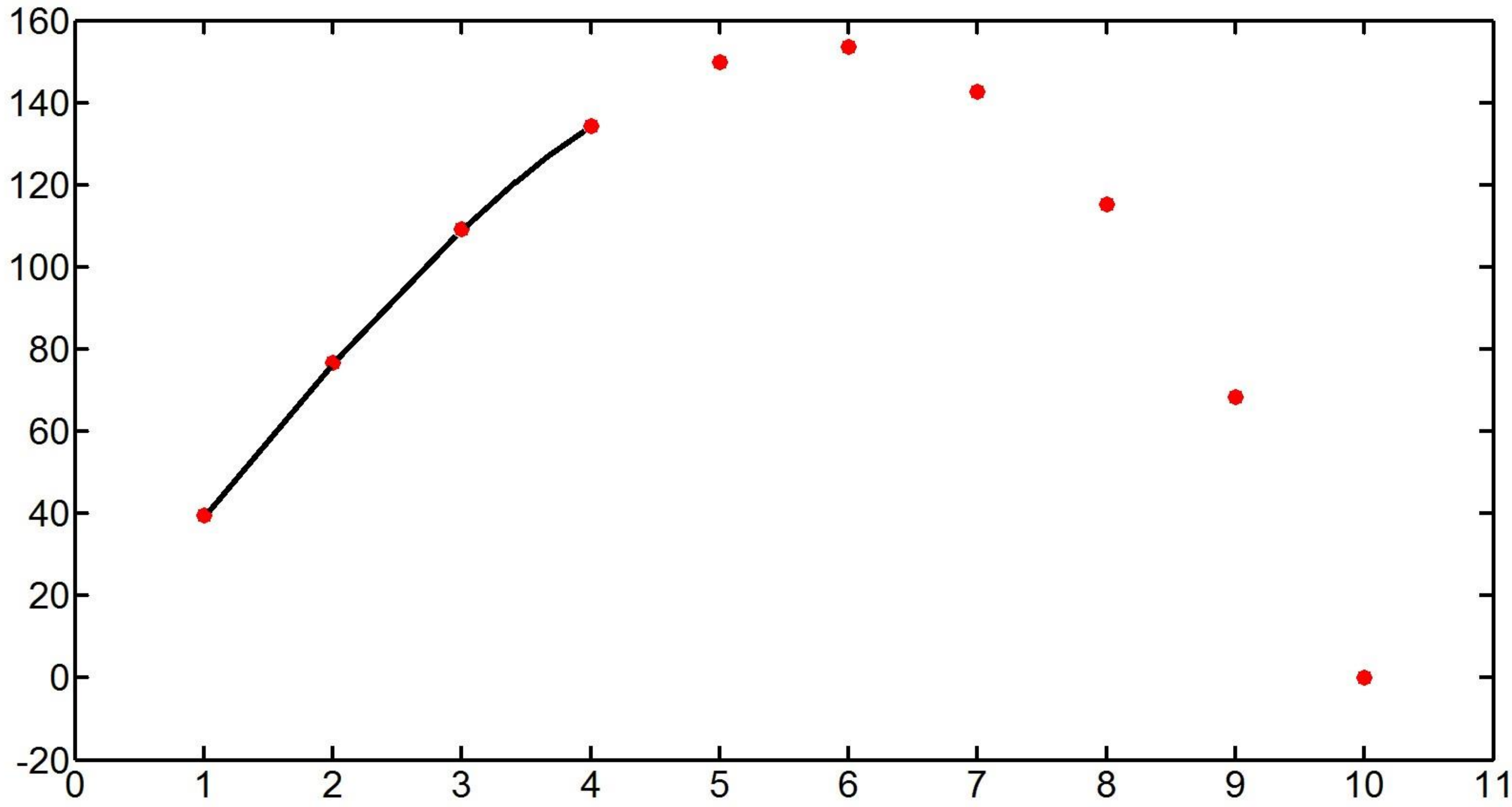


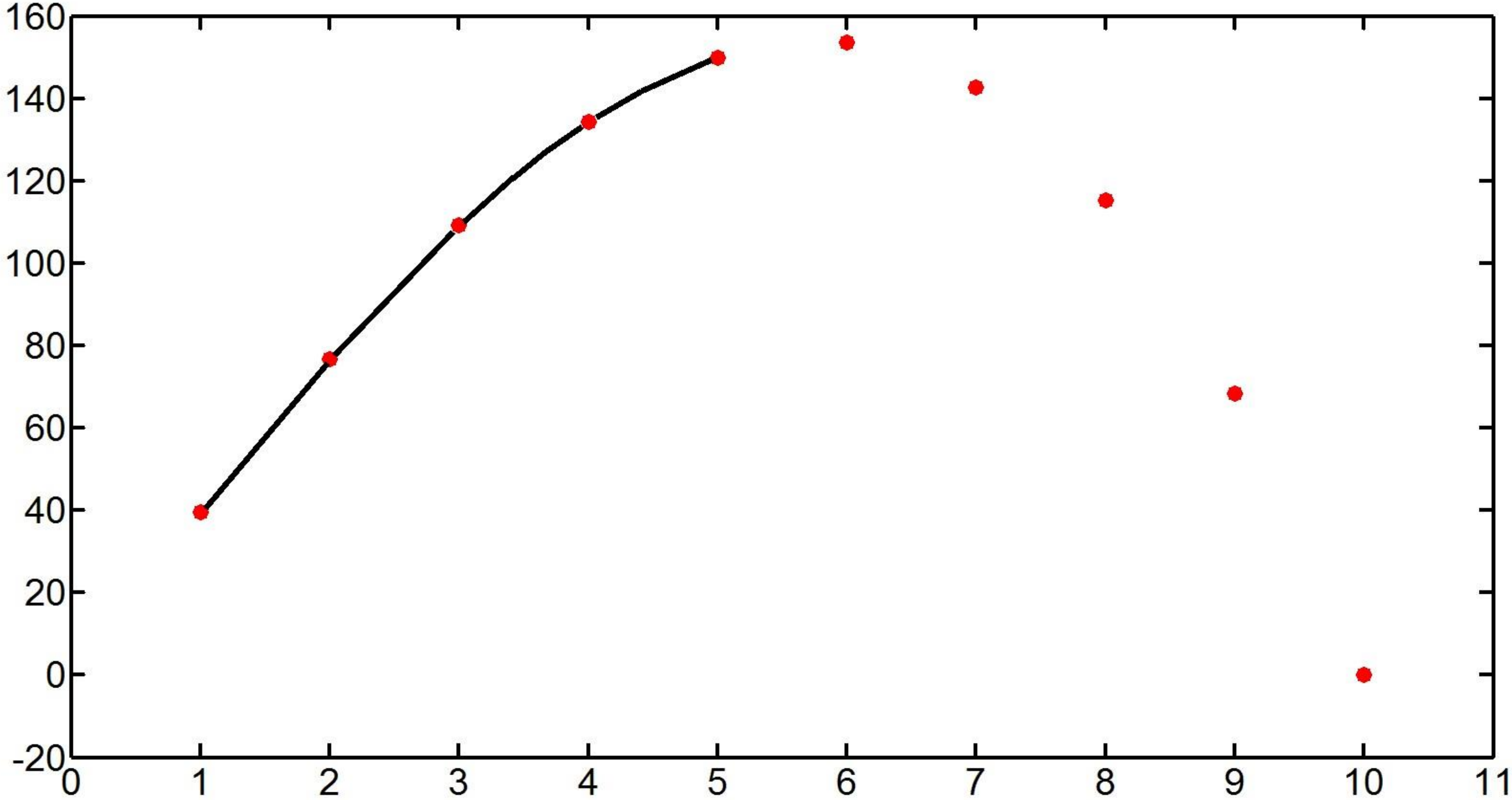

**First, we consider quadratic splines**

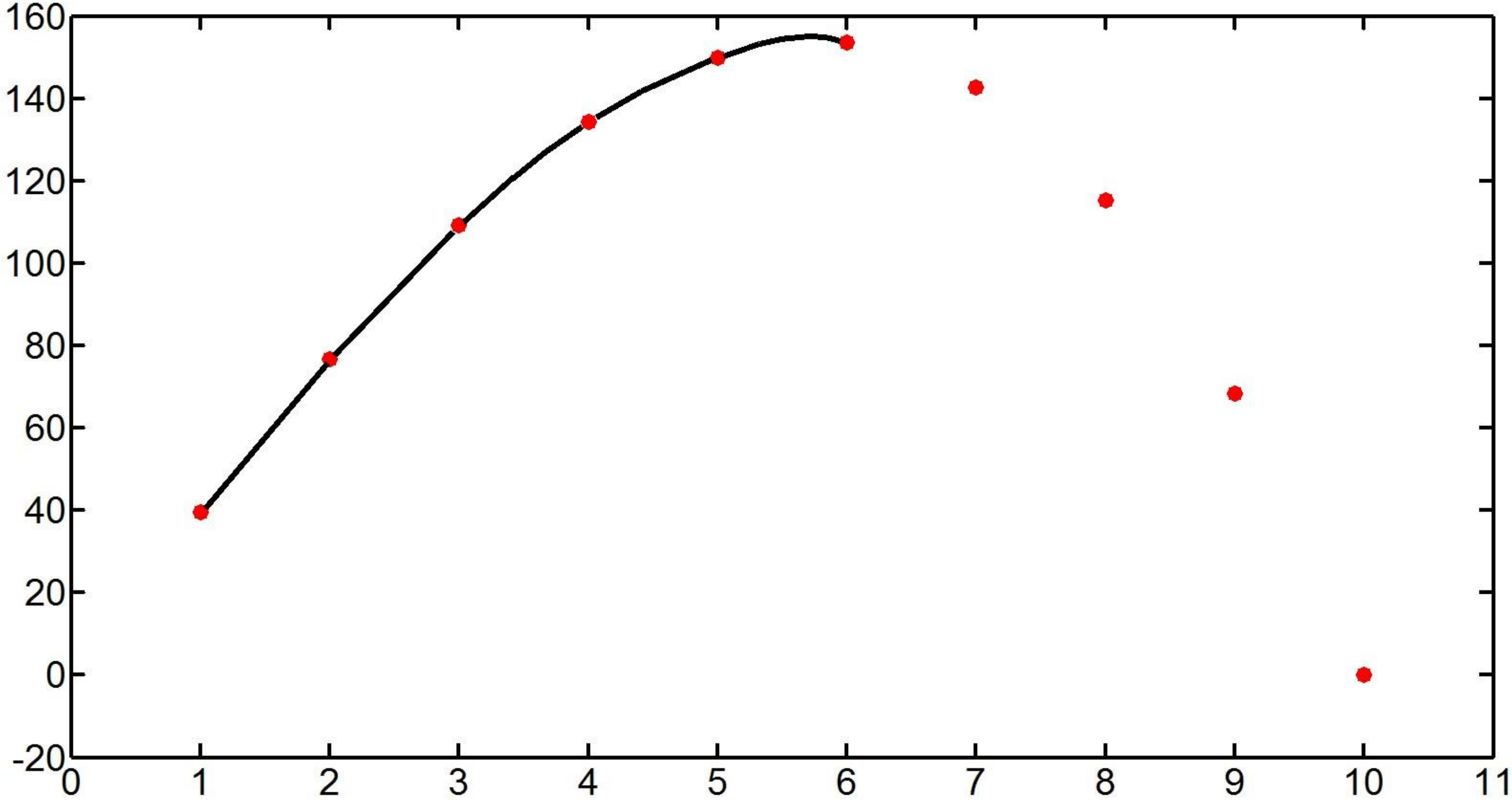
$$S(x) = a_i + b_i \cdot (x - x_i) + c_i \cdot (x - x_i)^2$$

$$at \quad x_i \le x \le x_{i+1} \;, \qquad i = 0, 1, \ldots, n-1$$

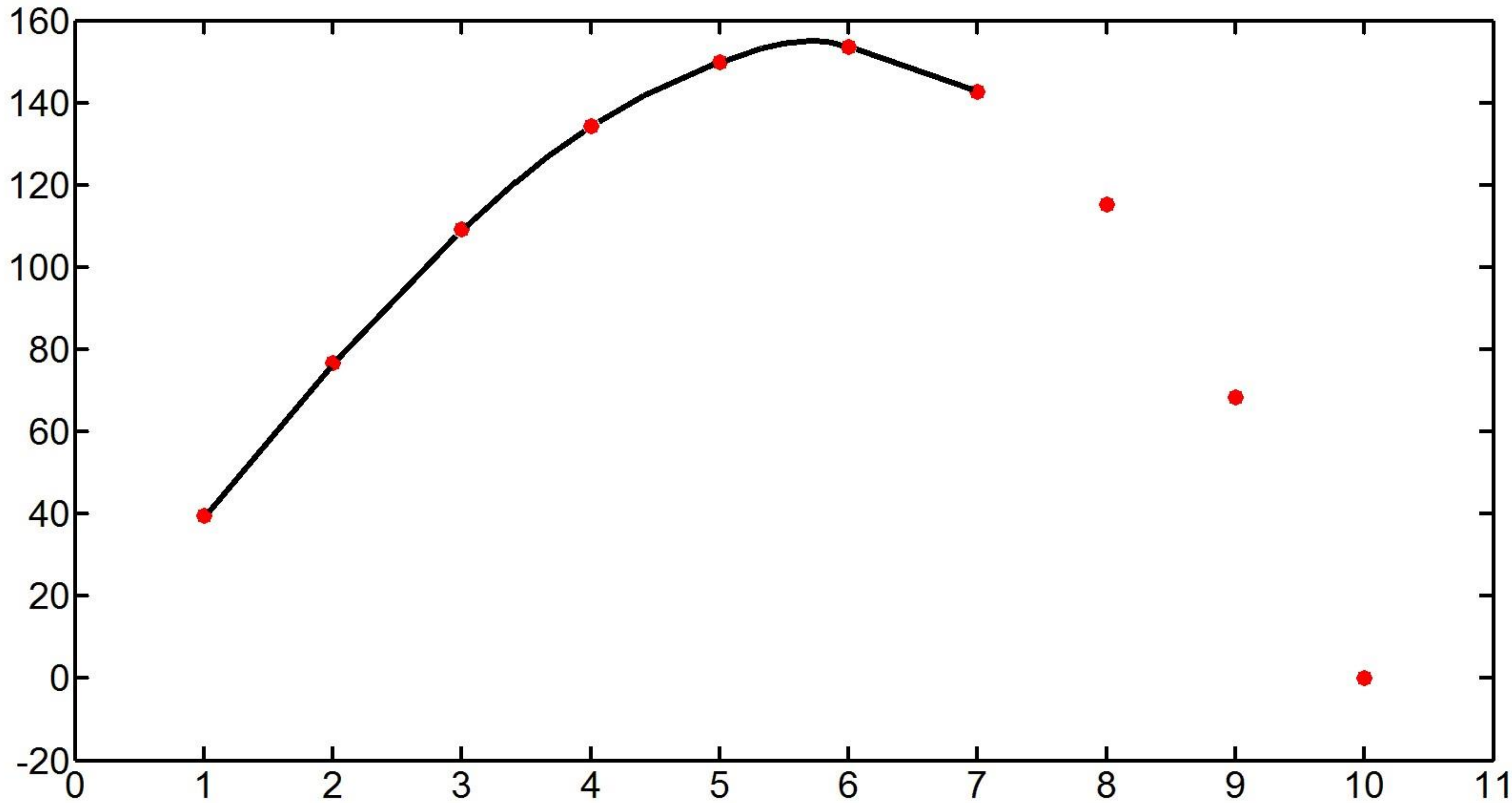

160
140
120
100
80
60
40
20
0
-20
0
1
2
3
4
5
6
7
8
9
10
11

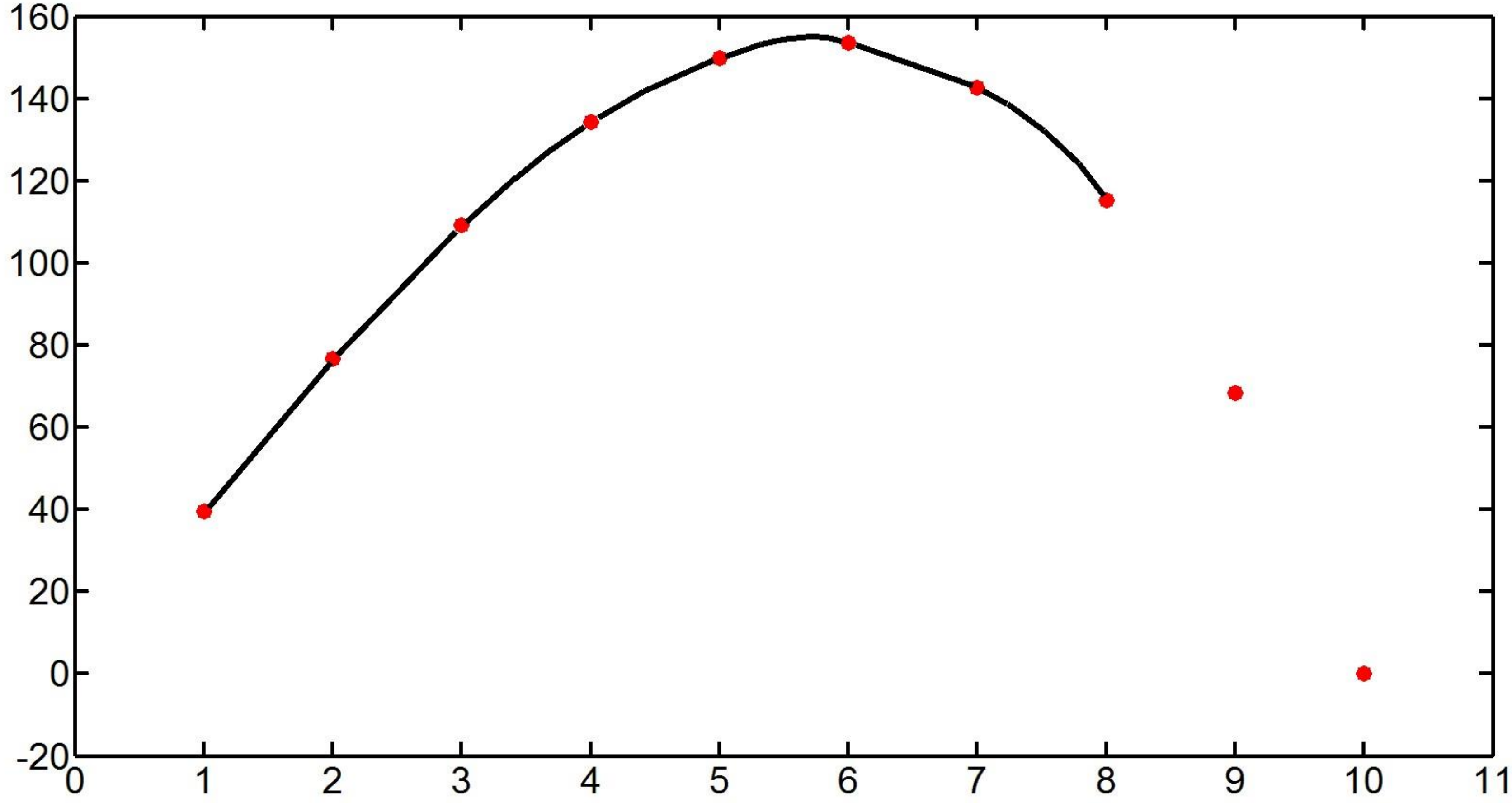

160
140
120
100
80
60
40
20
0
-20
0
1
2
3
4
5
6
7
8
9
10
11

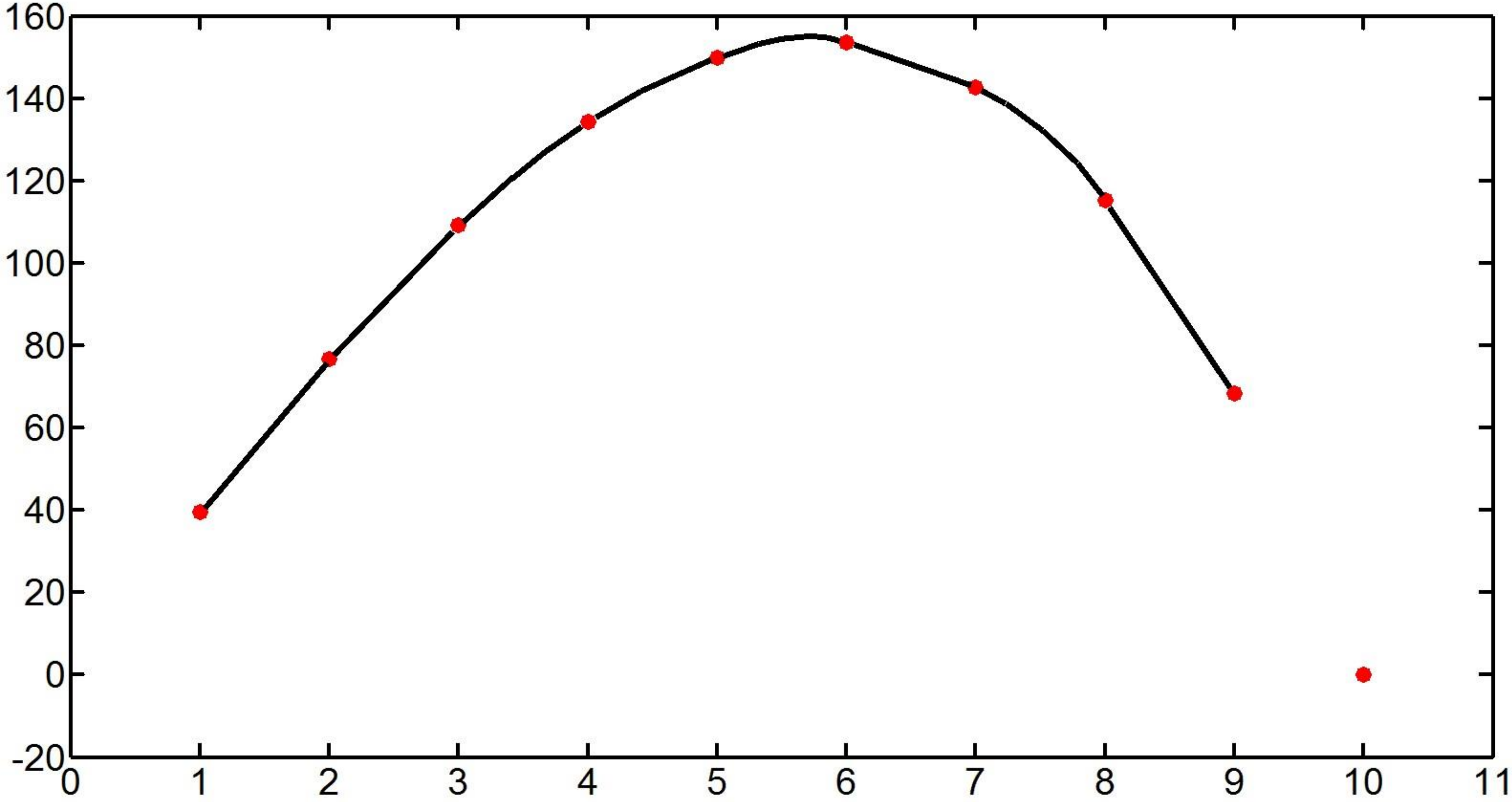

160
140
120
100
80
60
40
20
0
-20
0
1
2
3
4
5
6
7
8
9
10
11

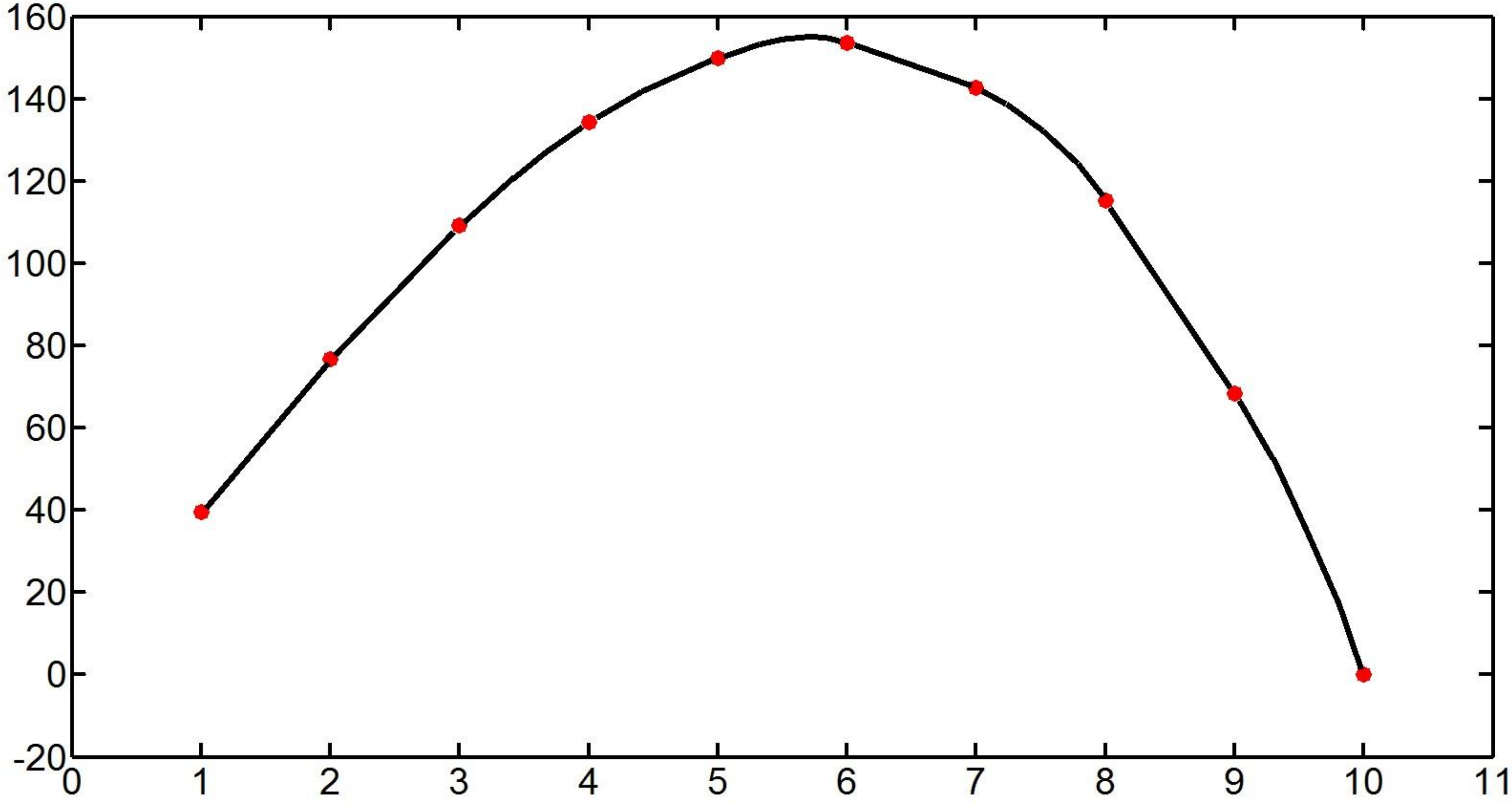


160
140
120
100
80
60
40
20
0
-20
0
1
2
3
4
5
6
7
8
9
10
11


160
140
120
100
80
60
40
20
0
-20
0
1
2
3
4
5
6
7
8
9
10
11


160
140
120
100
80
60
40
20
0
-20
0
1
2
3
4
5
6
7
8
9
10
11


160
140
120
100
80
60
40
20
0
-20
0
1
2
3
4
5
6
7
8
9
10
11




$$S(x) = a_i + b_i \cdot (x - x_i) + c_i \cdot (x - x_i)^2$$

$$\text{at} \quad x_i \le x \le x_{i+1} \;, \qquad i = 0, 1, \ldots, n-1$$

$$S(x) = a_i + b_i \cdot (x - x_i) + c_i \cdot (x - x_i)^2$$

$$\text{at } x_i \le x \le x_{i+1}, \qquad i = 0, 1, \ldots, n-1$$

**Let us find coefficients** $a_i$, $b_i$, $c_i$ **using 3 conditions:**

**1) The condition that** $S(x)$ **equals to given** $y_i$ **at the left endpoint** $x_i$ **:**

$$S(x_i) = y_i \quad \Rightarrow \quad a_i = y_i \qquad i = 0, 1, \ldots, n-1$$

$$S(x) = a_i + b_i \cdot (x - x_i) + c_i \cdot (x - x_i)^2$$

$$\text{at } x_i \le x \le x_{i+1} \;, \qquad i = 0, 1, \ldots, n-1$$

**Let us find coefficients** $a_i$ , $b_i$ , $c_i$ **using 3 conditions:**

**1) The condition that** $S(x)$ **equals to given** $y_i$ **at the left endpoint** $x_i$ **:**

$$S(x_i) = y_i \quad \Rightarrow \quad a_i = y_i \qquad i = 0, 1, \ldots, n-1$$

**2)The condition that** $S(x)$ **equals to given value** $y_{i+1}$ **at** $x_{i+1}$ **:**

$$S(x_{i+1}) = y_{i+1} \quad \Rightarrow$$

$$\Rightarrow \quad a_i + b_i \cdot (x_{i+1} - x_i) + c_i \cdot (x_{i+1} - x_i)^2 = y_{i+1}$$

$$S(x) = a_i + b_i \cdot (x - x_i) + c_i \cdot (x - x_i)^2$$

$$\text{at } x_i \le x \le x_{i+1}, \qquad i = 0, 1, \ldots, n-1$$

**Let us find coefficients $a_i$, $b_i$, $c_i$ using 3 conditions:**

**1) The condition that $S(x)$ equals to given $y_i$ at the left endpoint $x_i$ :**

$$S(x_i) = y_i \quad \Rightarrow \quad a_i = y_i \qquad i = 0, 1, \ldots, n-1$$

**2)The condition that $S(x)$ equals to given value $y_{i+1}$ at $x_{i+1}$ :**

$$S(x_{i+1}) = y_{i+1} \quad \Rightarrow$$

$$\Rightarrow \quad a_i + b_i \cdot (x_{i+1} - x_i) + c_i \cdot (x_{i+1} - x_i)^2 = y_{i+1}$$

**3) The condition that $dS/dx$ is continuous at node $x_{i+1}$ :**

$$dS/dx = b_i + 2c_i \cdot (x - x_i) \qquad at \;\; x_i \le x \le x_{i+1}$$

$$b_i + 2c_i \cdot (x_{i+1} - x_i) = b_{i+1} + 2c_{i+1} \cdot (x_{i+1} - x_{i+1})$$

$$\begin{cases} b_i + 2c_i \cdot (x_{i+1} - x_i) = b_{i+1} & (1) \qquad i=0,\ 1,\ \ldots,\ n-2 \\ b_i \cdot (x_{i+1} - x_i) + c_i \cdot (x_{i+1} - x_i)^2 = y_{i+1} - y_i & (2) \end{cases}$$

see condition 2) above

$i$=0, 1, …, n-1

We get 2n-1 equations for 2n coefficients $b_i$, $c_i$

+ additional condition $c_0 = 0$

( $S(x) = a_0 + b_0 \cdot (x - x_0)$ at $x_0 \le x \le x_1$ )

**In fact, coefficients $b_i$, $c_i$ can be calculated <u>sequentially</u> for $i$=1,2, . . . , $n$ :**

$b_1$ , $c_1$ ,

*then* $b_2$ *from (1),*

*then* $c_2$ *from (2),*

*then* $b_3$ *from (1), . . .*

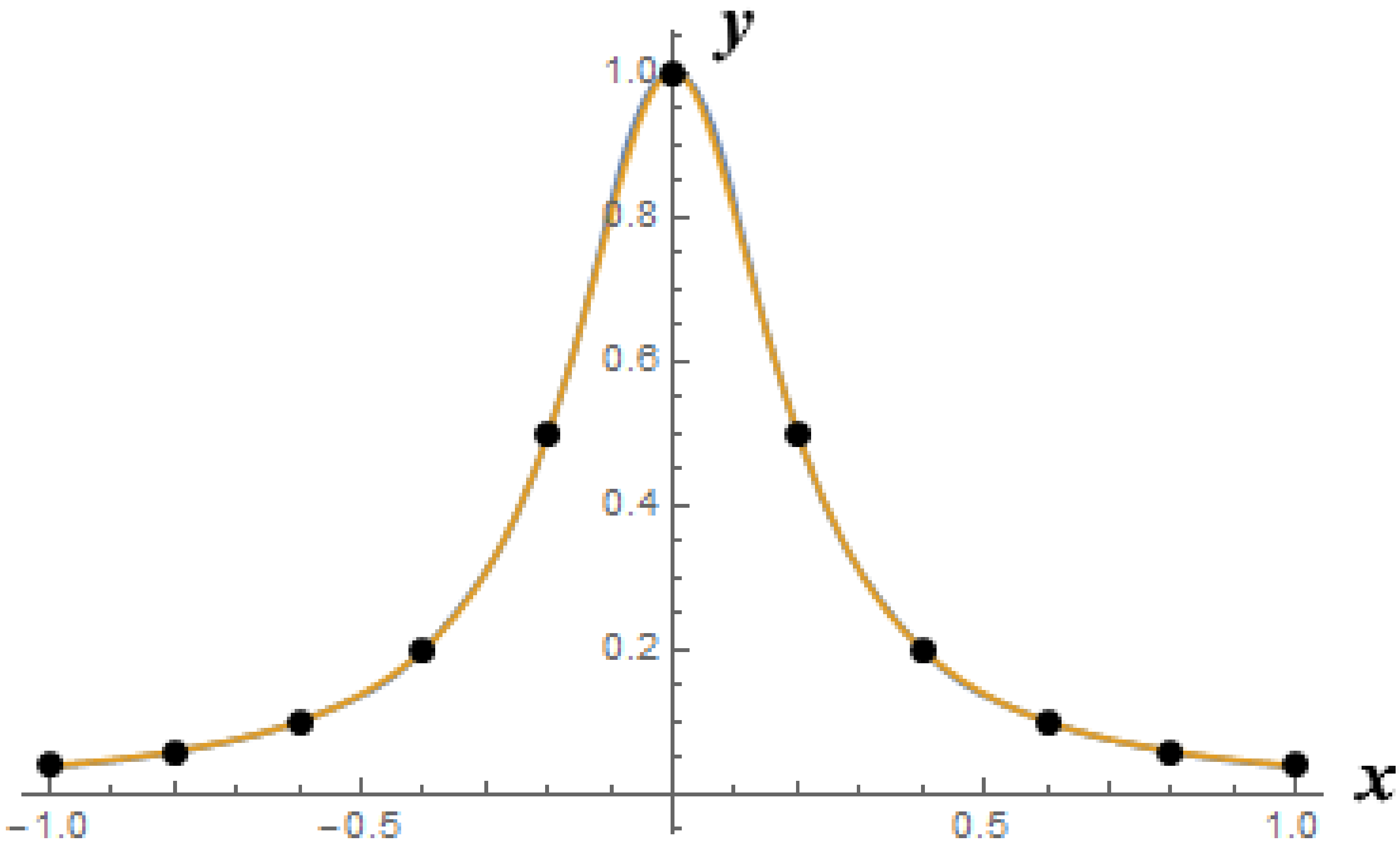
**<u>Theorem</u> on the error of interpolation:**

**If $f'''(x)$ is continuous on $[x_0, x_n]$, then**

$$|f(x)-S(x)| \leq \max|f'''(x)|\,h^3/12,$$

**where $h=\max|x_{i+1}-x_i|$ .**

Proof see in:
"Optimal Error Bounds for Quadratic Spline Interpolation"
Francois Dubeau
JOURNAL OF MATHEMATICAL ANALYSIS AND APPLICATIONS,
Vol. 198, pages 49-63, 1996




**Example: Interpolation of the function** $f(x) = 1/(1+25x^2)$
**by a** **quadratic spline** **looks perfect (the curves** ***S(x)*** **and** ***f(x)*** **coincide)**
**https://engcourses-uofa.ca/books/numericalanalysis**
**in contrast to interpolation by a Lagrangian's polynomial.**

We notice that the <u>second-order</u> derivative of

$$S(x) = a_i + b_i \cdot (x - x_i) + c_i \cdot (x - x_i)^2$$

is $d^2 S/dx^2 = 2\,c_i$

Therefore, it is constant at each interval

$$x_i \le x \le x_{i+1}$$

and usually jumps when $x$ moves from $x_i \le x \le x_{i+1}$ to next segment, since $c_i$ are usually different in different segments.

**Using cubic splines**

$$S_{cub}(x) = a_i + b_i \cdot (x-x_i) + c_i \cdot (x-x_i)^2 + d_i \cdot (x-x_i)^3$$

$$x_i \leq x \leq x_{i+1}\ , \qquad i=0,\ 1,\ \ldots,\ n-1,$$

***with coefficient $d_i$, we can provide the continuity of $d^2S_{cub}/dx^2$ at the nodes $x_i$.***

**Indeed, the derivative**

$$d^2S_{cub}/dx^2 = 2c_i + 6d_i \cdot (x-x_i)$$

**changes in** $x_i \leq x \leq x_{i+1}$

**and therefore its values at the ends of interval can be adjusted in a proper way.**

## We set 4 conditions:

1) $S_{cub}(x)$ is equal to $y_i$ at left end of segment

$$x_i \le x \le x_{i+1} \quad : \qquad y_i = a_i$$

2) $S_{cub}(x)$ is equal to $y_{i+1}$ at right end of the segment

$$y_{i+1} = a_i + b_i \cdot (x_{i+1} - x_i) + c_i \cdot (x_{i+1} - x_i)^2 + d_i \cdot (x_{i+1} - x_i)^3$$

3) **First-order derivative:**

$$dS_{cub}/dx = b_i + 2 c_i \cdot (x - x_i) + 3 d_i \cdot (x - x_i)^2$$

**the condition of equal derivatives at the right endpoint:**

$$b_i + 2 c_i \cdot (x_{i+1} - x_i) + 3 d_i \cdot (x_{i+1} - x_i)^2 = b_{i+1}$$

$$i = 1, 2, \ldots, n-1$$

**4) Second-order derivative :**

$$d^2S_{cub}/dx^2 = 2c_i + 6d_i \cdot (x-x_i)$$

**the condition of equal second derivatives at the right endpoint:**

$$2c_i + 6d_i \cdot (x_{i+1}-x_i)=2c_{i+1} \qquad i=1,2,\ldots,n-1$$

**We have got 2n+2(n-1) equations with respect to 4n coefficients of spline $S_{cub}$ .**

**We add two conditions $d^2S_{cub}/dx^2 = 0$ at endpoints $x_0, x_n$ :**

$$c_0=0, \qquad 2c_{n-1} + 6d_{n-1} \cdot (x_n-x_{n-1})=0$$

**Finally, we have 4n equations with respect to 4n coefficients.**

```
x0= 1
x1= 2
x2= 3
x3= 5
      y0= 2
      y1= 2.9
      y2= 4.2
      y3= 6
plot(x0,y0,'o',x1,y1,'o',x2,y2,'o',x3,y3,'o')

d=splin( [x0 x1 x2 x3], [y0 y1 y2 y3] )
x= 1: 0.02 : 5
y=interp(x, [x0 x1 x2 x3], [y0 y1 y2 y3], d)
plot(x,y,'b','LineWidth',3)
```

# Interpolation of functions of two independent variables

## [DOUBLE INTERPOLATION]

In the preceding sections we have derived interpolation formulae to approximate a function of a single variable. For a function of two or more variables, the formulae become complicated but a simpler procedure is to interpolate with respect to the first variable keeping the others constant, then interpolate with respect to the second variable, and so on. The method is illustrated below for a function of two variables.

The following table gives the values of $z$ for different values of $x$ and $y$. Find $z$ when $x = 2.5$ and $y = 1.5$.

| $y$ \ $x$ | 0 | 1 | 2 | 3 | 4 |
|---|---|---|---|---|---|
| 0 | 0 | 1 | 4 | 9 | 16 |
| 1 | 2 | 3 | 6 | 11 | 18 |
| 2 | 6 | 7 | 10 | 15 | 22 |
| 3 | 12 | 13 | 16 | 21 | 28 |
| 4 | 18 | 19 | 22 | 27 | 34 |

We first interpolate with respect to $x$ keeping $y$ constant. For $x = 2.5$, we obtain the following table using *linear interpolation*.

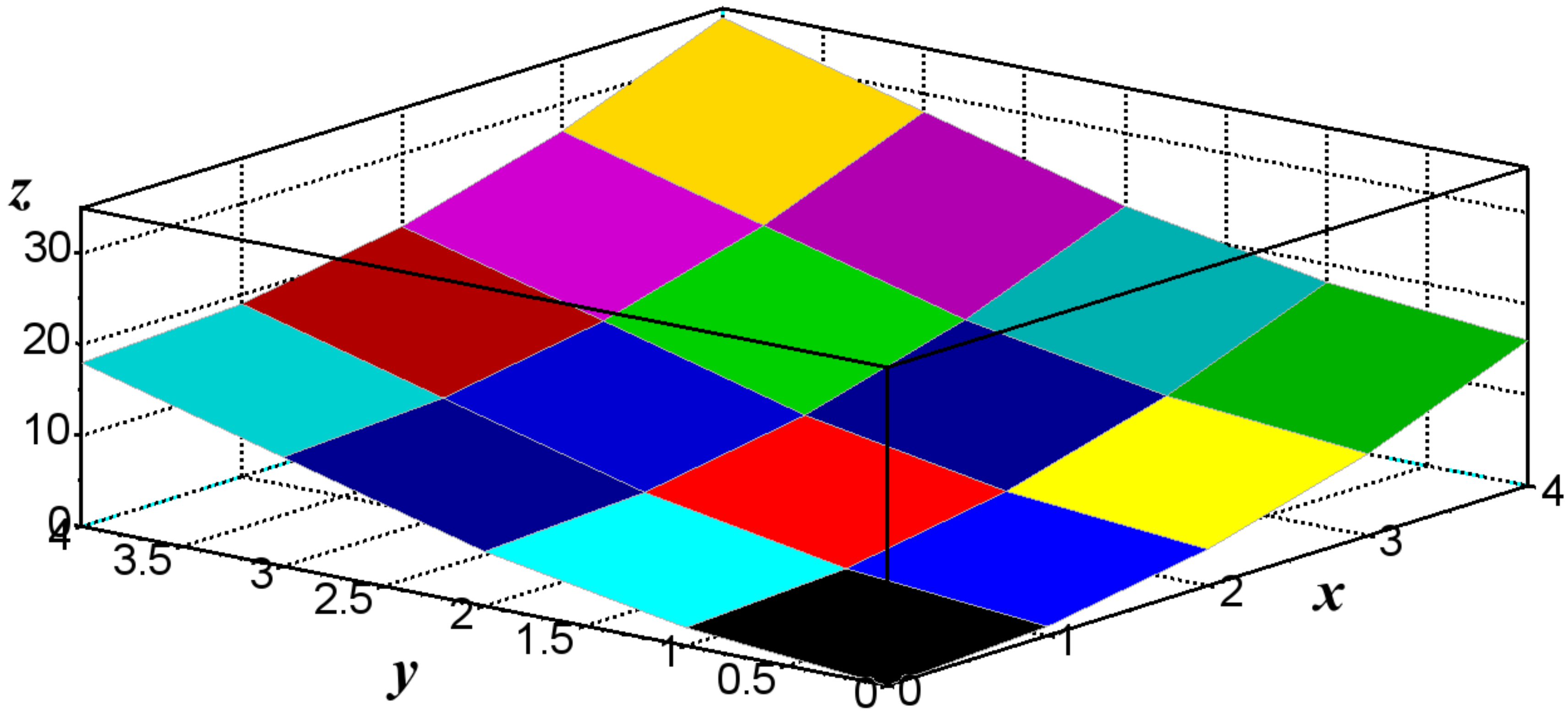
| y | z |
|---|---|
| 0 | 6.5 |
| 1 | 8.5 |
| 2 | 12.5 |
| 3 | 18.5 |
| 4 | 24.5 |

| $y$ | $z$ |
|---|---|
| 0 | 6.5 |
| 1 | 8.5 |
| 2 | 12.5 |
| 3 | 18.5 |
| 4 | 24.5 |

Now, we interpolate with respect to $y$ using linear interpolation once again. For $y = 1/5$, we obtain

$$z = \frac{8.5 + 12.5}{2} = 10.5$$

so that $z(2.5, 1.5) = 10.5$. Actually, the tabulated function is $z = x^2 + y^2 + y$ and hence $z(2.5, 1.5) = 10.0$, so that the computed value has an error of 5%.




**Scilab:**

splin2d( )

interp2d( )