Chapter 3

Interpolation methods

Problem:

Some measured values (dependent variable) are dependent on independent variables in one -, two -, three- or four-dimensional space measurement, and generally they are represented by the three space coordinates (depending upon coordinate system e.g. x_n, y_n, z_n or r_n, α_n, z_n or $r_n, \alpha_n, \vartheta_n$ (see 2 vector analysis)) and the time t_n. We have a discontinuous value tables in this case. For the one dimensional case e.g.:

independent variable	**dependend variable**
x_0	$y_0 = f(x_0)$
x_1	$y_1 = f(x_1)$
$\vdots$	$\vdots$
x_n	$y_n = f(x_n)$

The points $x_0, x_1, ..., x_n$ are called **supporting points**, and the points $y_0, y_1, ..., y_n$ are the **basic** or **supporting values**.
If we are looking for function values, whose arguments lie within the range (x_0, x_n), we name it **interpolation** In contrast if the function values we are looking for lie outside the range (x_0, x_n) we call this **extrapolation**. By interpolation or extrapolation we try to find a continuous function $w = p(x)$, which reflects the original function $y_n = f(x_n)$ as exactly as possible (see figure 3.1). It is always assumed that the interpolation function only matches the original function on the supporting points. The accuracy for the interval in between, e.g. the matching of both functions, depends on the number and the distribution of the supporting points. According to the **sampling theorem** the **quantisation error** increases proportionally to the rise of the function.

Note:

No interpolation algorithm can be used as replacement for an enlargement ot the measured value density. By means of an interpolation algorithm one receives in each case **approximated values**.

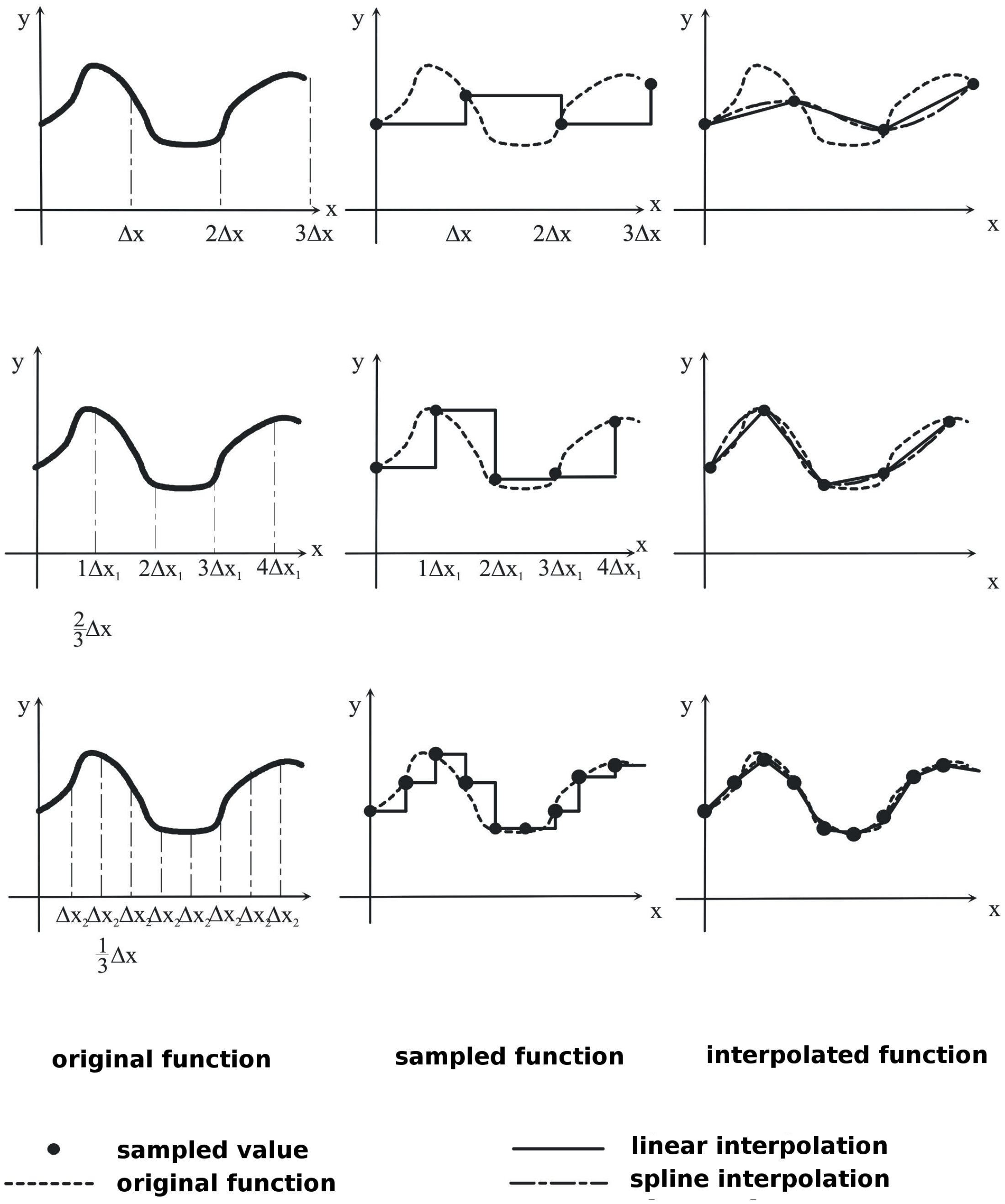

Figure 3.1: representation of the discontinuous measured data acquisition

Example for application of interpolation: The pollutant concentration $C(x)$, which runs from a refuse dump, is measured at the points x_0, x_1, x_2 (see figure 3.2). The pollutant concentration at the point x_{Fl}, which cause the danger by flowing in the river, is to be estimated by interpolation. A conclusion is to be given whether this value exceeds the limiting value.

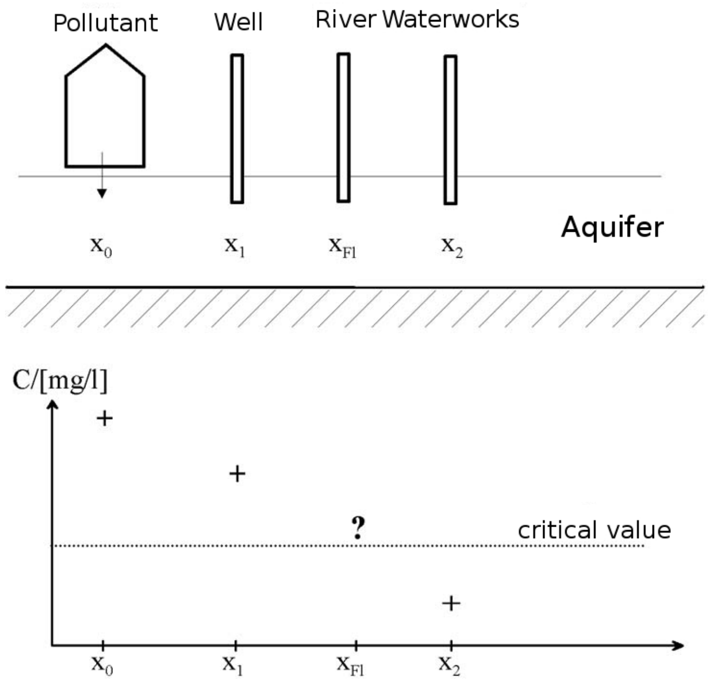

Figure 3.2: representation of an interpolation problem

x_0	$C_0 = f(x_0)$
x_1	$C_1 = f(x_1)$
x_{Fl}	?
x_2	$C_2 = f(x_2)$

For the solution of this problem an interpolation function $w = p(x)$ is to seek for as "replacement" for the function $C_n = f(x_n)$. This function should fulfil the following conditions:

$$w_i = p(x_i) = C_i \forall i = 1, \ldots, n \tag{3.1}$$

i.e.

$$\begin{aligned}
w_0 &= p(x_0) = C_0 \\
w_1 &= p(x_1) = C_1 \\
&\vdots \\
w_n &= p(x_n) = C_n
\end{aligned} \tag{3.2}$$

Then it is supposed that the intermediate values of the function $w = p(x)$ are a good approximation of the intermediate values of the function $C_n = f(x_n)$.
For the determination of the function $w = p(x)$ different interpolation methods can be used. We differentiate thereby one- and multi-dimensional procedures. The multidimensional methods play an important role in connection with the **geographical information systems (GIS)** and are also often applied in connection with **geostatistics**.

In the following some methods will be introduced in connection with water economical questions.

- Polynomial interpolation
- Spline interpolation (peace wise polynomial interpolation)
- Kriging method

3.1 Polynomial interpolation

In this method $p(x)$ has the form of an algebraic polynomial of order n:

$$w = p(x) = a_0 + a_1 x + a_2 x^2 + ... + a_n x^n \tag{3.3}$$

The advantage of this method is that the intermediate values can be computed as easily as possible.

Based on a value table with $n+1$ pairs of variates maximally an n-th order polynomial can be exactly determined:

$$y := p(x) = \sum_{k=0}^{n} a_k \cdot x^k \tag{3.4}$$

with the property:

$$y(x_i) \approx p(x_i) = \sum_{k=0}^{n} a_k \cdot x_i^k = w_i \tag{3.5}$$

This polynomial is the interpolation polynomial to the given system of interpolation supporting points.

Normally we look for polynomials of lower order ($n \leq 3$), which fit together the pairs of values at least by pairs:

$p(x) = a_0 + a_1 x$	linear interpolation
$p(x) = a_0 + a_1 x + a_2 x^2$	quadratic interpolation
$p(x) = a_0 + a_1 x + a_2 x^2 + a_3 x^3$	cubic interpolation

The application of polynomials with higher order makes the calculations more difficult and leads to very large fluctuations.

From the different display formats for the polynomials follow different interpolation methods for the determination of the coefficients a_i of an n-th order polynomial. All this different methods lead to the same polynomial.

This interpolation methods are:

- analytical power function
- Lagrange
- Aiken
- Newton

3.1.1 Analytical power function

This method assumes that for each supporting point the polynomial $w = p(x)$ fulfils the condition $y(x_i) = p(x_i)$. In this case we get for the $n+1$ supporting points a system of $n+1$ equations with $n+1$ unknown coefficients $a_0, \ldots, a_n$.

$$
\begin{aligned}
a_0 + a_1 x_0 + a_2 x_0^2 + ... + a_n x_0^n &= y_0 \\
a_0 + a_1 x_1 + a_2 x_1^2 + ... + a_n x_1^n &= y_1 \\
&\vdots \\
a_0 + a_1 x_n + a_2 x_n^2 + ... + a_n x_n^n &= y_n
\end{aligned}
\tag{3.6}
$$

This equation system can be written as a matrix equation:

$$\mathbf{X} \cdot \mathbf{A} = \mathbf{Y}$$

with the matrices:

$$
\mathbf{X} = \begin{pmatrix} 1 & x_0 & x_0^2 & \cdots & x_0^n \\ 1 & x_1 & x_1^2 & \cdots & x_1^n \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & x_n^2 & \cdots & x_n^n \end{pmatrix} \qquad \mathbf{A} = \begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_n \end{pmatrix} \qquad \mathbf{Y} = \begin{pmatrix} y_0 \\ y_1 \\ \vdots \\ y_n \end{pmatrix}
$$

The matrix **X** and the vector **Y** on the right side represent the known coefficients, whereby **A** is the solution vector. The linear equation system can be solved by all the known methods (see section 1.3, solution of equations systems, page 22)
The determinant of this linear equation system is:

$$D = \begin{vmatrix} 1 & x_0 & \cdots & x_0^n \\ 1 & x_1 & \cdots & x_1^n \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_n & \cdots & x_n^n \end{vmatrix} = \begin{pmatrix} (x_1 - x_0) \cdot (x_2 - x_0) \cdot (x_3 - x_0) \cdot \ldots \cdot (x_n - x_0) \cdot \\ \cdot (x_2 - x_1) \cdot (x_3 - x_1) \cdot \ldots \cdot (x_n - x_1) \cdot \\ \cdot (x_3 - x_2) \cdot \ldots \cdot (x_n - x_2) \cdot \\ \vdots \\ \cdot (x_{n-1} - x_{n-2}) \cdot (x_n - x_{n-2}) \cdot \\ \cdot (x_n - x_{n-1}) \end{pmatrix} \tag{3.7}$$

and is named the VANDERMOND determinant.

Since all supporting points are (must be) pairwise different, is $D \neq 0$ and the linear equation system is explicit solvable.

There is only one polynomial of the order n which fulfils the property $y_i = f(x_i) = p(x_i)$ with the coefficients (see section 1.2.3 determinants, page 19):

$$a_0 = \frac{D_{a_0}}{D}, a_1 = \frac{D_{a_1}}{D}, \cdots, a_n = \frac{D_{an}}{D} \tag{3.8}$$

With this coefficients the interpolation polynomial is:

$$y(x) \approx p(x) = a_0 + a_1 \cdot x + a_2 \cdot x^2 + \cdots + a_n \cdot x^n$$

The interpolation value at the place x_P is:

$$y(x_P) \approx p(x_P) = a_0 + a_1 \cdot x_P + a_2 \cdot x_P^2 + \cdots + a_n \cdot x_P^n$$

Although the beginning of this method is very simple, the final determination of the interpolation polynomial requires a relative large computation, particularly if a great number of supporting points are to be taken in account.

Example for the interpolation with the analytical power function method: Find a quadratic polynomial by using the values of the following table and calculate the value $y = f(\frac{1}{2})$ at the place $x = \frac{1}{2}$.

x	0	1	2
y	0	1	0

Since only three supporting points are given, the polynomial can only be a second order polynomial. A quadratic polynomial has the form

$$p(x) = a_0 + a_1 x + a_2 x^2$$

It must be:

$$y_i = p(x_i)$$

$$y_i = a_0 + a_1 x_i + a_2 x_i^2$$

$$\begin{aligned}
p(0) = 0 \quad &\Rightarrow \quad a_0 + a_1 \cdot 0 + a_2 \cdot 0 = 0 \quad &&\Rightarrow \quad a_0 = 0 \\
p(1) = 1 \quad &\Rightarrow \quad a_0 + a_1 \cdot 1 + a_2 \cdot 1^2 = 1 \quad &&\Rightarrow \quad a_1 + a_2 = 1 \\
p(2) = 0 \quad &\Rightarrow \quad a_0 + a_1 \cdot 2 + a_2 \cdot 2^2 = 0 \quad &&\Rightarrow \quad 2a_1 + 4a_2 = 0
\end{aligned}$$

From this three equations follows:

$$a_0 = 0$$

$$a_1 = 2$$

$$a_2 = -1$$

Thus the interpolation polynomial is:

$$p(x) = 2x - x^2$$

With this function the value at the place $x = \frac{1}{2}$ can be computed:

$$f\left(\frac{1}{2}\right) \approx p\left(\frac{1}{2}\right) = 2 \cdot \frac{1}{2} - \left(\frac{1}{2}\right)^2 = \frac{3}{4}$$

3.1.2 Lagrange interpolation formula

Lagrange wrote the interpolation in the following form:

$$y(x_P) \approx p(x_P) = L_0(x_P) \cdot y_0 + L_1(x_P) \cdot y_1 + \ldots + L_n(x_P) \cdot y_n \tag{3.9}$$

With the Lagrange interpolation no closed analytical functions are computed, but only single values $p(x_P)$ for each interpolation point x_P. Thereby the coefficients $L_i(x)$ of the interpolation values y_i (for $i = 0, 1, \ldots, n$) are n-th order polynomials of x_P. These are computed from the supporting points x_i and are called the Lagrange **polynomials**. The Lagrange polynomials of n-th order are:

$$\begin{aligned}
L_0(x) &= \frac{(x_P - x_1)(x_P - x_2)\cdots(x_P - x_n)}{(x_0 - x_1)(x_0 - x_2)\cdots(x_0 - x_n)} \\
L_1(x) &= \frac{(x_P - x_0)(x_P - x_2)\cdots(x_P - x_n)}{(x_1 - x_0)(x_1 - x_2)\cdots(x_1 - x_n)} \\
&\vdots \\
L_i(x) &= \frac{(x_P - x_0)(x_P - x_1)\cdots(x_P - x_{i-1})(x_P - x_{i+1})\cdots(x_P - x_n)}{(x_i - x_0)(x_i - x_1)\cdots(x_i - x_{i-1})(x_i - x_{i+1})\cdots(x_i - x_n)} \\
&\vdots \\
L_n(x) &= \frac{(x_P - x_0)(x_P - x_1)(x_P - x_2)\cdots(x_P - x_{n-1})}{(x_n - x_0)(x_n - x_1)\cdots(x_n - x_{n-1})}
\end{aligned} \tag{3.10}$$

Thus the Lagrange **interpolation polynomial** is:

$$\begin{aligned}
y &= f(x_P) \approx p(x_P) = L_0(x_P)y_0 + L_1(x_P)y_1 + \ldots + L_n(x_P)y_n \\
y &= \frac{(x_P - x_1)(x_P - x_2)\cdots(x_P - x_n)}{(x_0 - x_1)(x_0 - x_2)\cdots(x_0 - x_n)} y_0 \\
&+ \frac{(x_P - x_0)(x_P - x_2)\cdots(x_P - x_n)}{(x_1 - x_0)(x_1 - x_2)\cdots(x_1 - x_n)} y_1 \\
&\vdots \\
&+ \frac{(x_P - x_0)(x_P - x_1)(x_P - x_2)\cdots(x_P - x_{n-1})}{(x_n - x_0)(x_n - x_1)\cdots(x_n - x_{n-1})} y_n
\end{aligned} \tag{3.11}$$

If we insert for x_P one of the points $x_0, x_1, ..., x_{n-1}, x_n$, there is always a factor in the numerator which is equal to zero. For the i-th Lagrange polynomial the denominator is also equal to zero, so the limes of the i-th Lagrange polynomial is equal to 1. Since all other Lagrange polynomials are zero, the Lagrange **interpolation polynomial** is:

$$y_i = y(x_i) = f(x_i) \approx p(x_i) = 1 \cdot y_i$$

A disadvantage of the Lagrange method is that the computation of the Lagrange interpolation polynomials must be accomplished again when an increase of the supporting point number should be taken into account, which is identical with the increase of the order of the interpolation polynomial. This is clearly to be seen in the following example.
Note:

- The weights (coefficients) $L_i(x_i)$ in the Langrange interpolation formula always have to be computed again, if the supporting points number changes.
- The sum of the weights is always equal to one (important for checking the results)

$$\sum L_i(x_i) = 1$$

Example for Lagrange interpolation:
For the function $y_n = f(x_n)$ the values at the equidistant points $x_n = x_0 + 2nh$ for $n = -1, 0, 1, 2$ (see table) are given:

n	**−1**	**0**	**1**	**2**
$\mathbf{x}_n$	$x_0 - 2h$	x_0	$x_0 + 2h$	$x_0 + 4h$
$\mathbf{f}(x_n)$	y_{-1}	y_0	y_1	y_2

Find an approximate value $w = f(x) = f(x_0 + h)$ for $x = \frac{1}{2}$. According to the rules of the polynomial interpolation in the case with four supporting points maximally a 3rd order polynomial can be developed. It is also possible to accomplish a piecewise interpolation. This has the advantage that we can reduce computation work. The accuracy is however declined. In this case we try to find an optimum between the required accuracy and the cost of computation. The supporting points used in the piecewise interpolation are this next to the interpolation point.

1. **Linear interpolation**
   The interpolation function at the point $x = \frac{1}{2}$ is written with the help of the Lagrange interpolation formula as (see equation 3.1.2):

   $$w_{\frac{1}{2}} = L_0(x_{\frac{1}{2}})y_0 + L_1(x_{\frac{1}{2}})y_1$$

   Here the supporting points $x = 0$ and $x = 1$ are used, where the point $x = \frac{1}{2}$ is in between. The weights L_0 and L_1 are (see equation 3.10):

$$L_0(x_{\frac{1}{2}}) = \frac{x_{\frac{1}{2}} - x_1}{x_0 - x_1} = \frac{x_0 + h - x_0 - 2h}{x_0 - x_0 - 2h} = \frac{1}{2}$$

$$L_1(x_{\frac{1}{2}}) = \frac{x_{\frac{1}{2}} - x_0}{x_1 - x_0} = \frac{x_0 + h - x_0}{x_0 + 2h - x_0} = \frac{1}{2}$$

Then the searched value is:

$$w_{\frac{1}{2}} = \frac{1}{2}(y_0 + y_1)$$

The result of the linear interpolation is thereby equal to the arithmetic middle.

2. **Quadratic interpolation**
   In this case the interpolation function is (see equation 3.1.2):

$$w_{\frac{1}{2}} = L_0(x_{\frac{1}{2}})y_0 + L_1(x_{\frac{1}{2}})y_1 + L_2(x_{\frac{1}{2}})y_2$$

   The corresponding coefficients are (see equation 3.10):

$$L_0(x_{\frac{1}{2}}) = \frac{(x_{\frac{1}{2}} - x_1)(x_{\frac{1}{2}} - x_2)}{(x_0 - x_1)(x_0 - x_2)} = \frac{3}{8}$$

$$L_1(x_{\frac{1}{2}}) = \frac{(x_{\frac{1}{2}} - x_0)(x_{\frac{1}{2}} - x_2)}{(x_1 - x_0)(x_1 - x_2)} = \frac{3}{4}$$

$$L_2(x_{\frac{1}{2}}) = \frac{(x_{\frac{1}{2}} - x_0)(x_{\frac{1}{2}} - x_1)}{(x_2 - x_0)(x_2 - x_1)} = -\frac{1}{8}$$

   and the result is:

$$w_{\frac{1}{2}} = \frac{3}{8}y_0 + \frac{3}{4}y_1 - \frac{1}{8}y_2$$

3. **cubic interpolation**
   In the same way follows (see equation 3.1.2):

$$w_{\frac{1}{2}} = L_{-1}(x_{\frac{1}{2}})y_{-1} + L_0(x_{\frac{1}{2}})y_0 + L_1(x_{\frac{1}{2}})y_1 + L_2(x_{\frac{1}{2}})y_2$$

   We get the following Lagrange factors (see equation 3.10):

$$L_{-1}(x_{\frac{1}{2}}) = \frac{(x_{\frac{1}{2}} - x_0)(x_{\frac{1}{2}} - x_1)(x_{\frac{1}{2}} - x_2)}{(x_{-1} - x_0)(x_{-1} - x_1)(x_{-1} - x_2)} = -\frac{1}{16}$$

$$L_0(x_{\frac{1}{2}}) = \frac{(x_{\frac{1}{2}} - x_{-1})(x_{\frac{1}{2}} - x_1)(x_{\frac{1}{2}} - x_2)}{(x_0 - x_{-1})(x_0 - x_1)(x_0 - x_2)} = \frac{9}{16}$$

$$L_1(x_{\frac{1}{2}}) = \frac{(x_{\frac{1}{2}} - x_{-1})(x_{\frac{1}{2}} - x_0)(x_{\frac{1}{2}} - x_2)}{(x_1 - x_{-1})(x_1 - x_0)(x_1 - x_2)} = \frac{9}{16}$$

$$L_2(x_{\frac{1}{2}}) = \frac{(x_{\frac{1}{2}} - x_{-1})(x_{\frac{1}{2}} - x_0)(x_{\frac{1}{2}} - x_1)}{(x_2 - x_{-1})(x_2 - x_0)(x_2 - x_1)} = -\frac{1}{16}$$

   Thus the result is:

$$w_{\frac{1}{2}} = -\frac{1}{16}y_{-1} + \frac{9}{16}y_0 + \frac{9}{16}y_1 - \frac{1}{16}y_2$$

3.1.3 Newton interpolation formula

3.1.3.1 Arbitrary supporting points

The disadvantage of the Lagrange method is that the Lagrange polynomials must be computed again and again, which can be avoided in the Newton method. With the Newton method only one auxiliary item should be added when further supporting points are taken into account.
The method begins with the following formula:

$$\begin{aligned} p(x) = & \quad b_0 + b_1(x-x_0) \\ & + \quad b_2(x-x_0)(x-x_1) \\ & + \quad b_3(x-x_0)(x-x_1)(x-x_2) \\ & \vdots \\ & + \quad b_n(x-x_0)(x-x_1)\cdots(x-x_{n-1}) \end{aligned} \tag{3.12}$$

If we want to find a certain interpolation value $p(x_P)$, x will be replaced by x_P in the polynomial.
The coefficients are determined again in such a way that the interpolation polynomial at the supporting point matches exactly with the interpolation values (x_n, y_n). If we respectively replace x_P by $x_0, \ldots, x_n$ in the Newton formula, we get an equation system with n equations and n variables. Since in each case the corresponding factors $(x_P - x_i = 0)$ are equal to zero, the polynomial items will be omitted. Then we know the basic value y_i from the polynomial value $p(x_i)$.

$$\begin{aligned} y_0 &= b_0 + b_1\underbrace{(x_0-x_0)}_{=0} + \cdots \\ y_1 &= b_0 + b_1(x_1-x_0) + b_2(x_1-x_0)\underbrace{(x_1-x_1)}_{=0} + \cdots \\ y_2 &= b_0 + b_1(x_2-x_0) + b_2(x_2-x_0)(x_2-x_1) \\ &\vdots \\ y_n &= b_0 + b_1(x_n-x_0) + b_2(x_n-x_0)(x_n-x_1) + \cdots \\ & \quad + b_n(x_n-x_0)(x_n-x_1)\cdots(x_n-x_{n-1}) \end{aligned} \tag{3.13}$$

The equation system can be solved step by step with $b_0, b_1, b_2 \ldots, b_n$. By inserting the first equation into the second we get b_1. Once again inserting this into the third equation it yields b_2. After $n+1$ steps we insert $b_0, \ldots, b_{n-1}$ in to the $n+1$-th equation and this yield b_n.

$$
\begin{aligned}
b_0 &= y_0 \\
b_1 &= \frac{(y_1 - y_0)}{(x_1 - x_0)} = [x_1 x_0] \\
b_2 &= \frac{(y_2 - y_0) - \frac{(y_1 - y_0)}{(x_1 - x_0)}(x_2 - x_0)}{(x_2 - x_0)(x_2 - x_1)} \\
&= \frac{(y_2 - y_1) + (y_1 - y_0) - [x_1 x_0](x_2 - x_0)}{(x_2 - x_0)(x_2 - x_1)} \\
&= \frac{(y_2 - y_1)}{(x_2 - x_0)(x_2 - x_1)} + \frac{(y_1 - y_0)}{(x_2 - x_0)(x_2 - x_1)} - \frac{[x_1 x_0](x_2 - x_0)}{(x_2 - x_0)(x_2 - x_1)} \\
&= \frac{[x_2 x_1]}{(x_2 - x_0)} + \frac{[x_1 x_0](x_1 - x_0)}{(x_2 - x_0)(x_2 - x_1)} - \frac{[x_1 x_0](x_2 - x_0)}{(x_2 - x_0)(x_2 - x_1)} \\
&= \frac{[x_2 x_1]}{(x_2 - x_0)} + \frac{[x_1 x_0](x_1 - x_0) - [x_1 x_0](x_2 - x_0)}{(x_2 - x_0)(x_2 - x_1)} \\
&= \frac{[x_2 x_1]}{(x_2 - x_0)} + \frac{-[x_1 x_0](x_2 - x_1)}{(x_2 - x_0)(x_2 - x_1)} \\
b_2 &= \frac{[x_2 x_1] - [x_1 x_0]}{(x_2 - x_0)} = [x_2 x_1 x_0]
\end{aligned}
\tag{3.14}
$$

Hereby we use the abbreviated notation, which is called **divided differences** of first and higher order:

$$
\begin{aligned}
[x_k x_i] &:= \frac{y_k - y_i}{x_k - x_i} \\
[x_l x_k x_i] &:= \frac{[x_l x_k] - [x_k x_i]}{(x_l - x_i)} \\
[x_m x_l x_k x_i] &:= \frac{[x_m x_l x_k] - [x_l x_k x_i]}{(x_m - x_i)} \\
&\vdots \\
[x_n x_{n-1} \cdots x_1 x_0] &:= \frac{[x_n x_{n-1} \cdots x_1] - [x_{n-1} \cdots x_1 x_0]}{(x_n - x_0)}
\end{aligned}
\tag{3.15}
$$

Thus the coefficients results to:

$$
\begin{aligned}
b_0 &= y_0 \\
b_1 &= \frac{(y_1 - y_0)}{(x_1 - x_0)} = [x_1 x_0] \\
b_2 &= \frac{[x_2 x_1] - [x_1 x_0]}{(x_2 - x_0)} = [x_2 x_1 x_0] \\
b_3 &= \frac{[x_3 x_2 x_1] - [x_2 x_1 x_0]}{(x_3 - x_0)} = [x_3 x_2 x_1 x_0] \\
&\vdots \\
b_n &= \frac{[x_n x_{n-1} \cdots x_1] - [x_{n-1} \cdots x_1 x_0]}{(x_n - x_0)} = [x_n x_{n-1} \cdots x_1 x_0]
\end{aligned}
\tag{3.16}
$$

Particularly the coefficients can be determined conveniently according to the following **computation scheme** (example for 5 supporting points):

x_0	$y_0 = \mathbf{b}_0$	$\frac{(y_1-y_0)}{(x_1-x_0)} = [x_1 x_0] = \mathbf{b}_1$	$[x_2 x_1 x_0] = \mathbf{b}_2$	$[x_3 x_2 x_1 x_0] = \mathbf{b}_3$	$[x_4 x_3 x_2 x_1 x_0] = \mathbf{b}_4$
x_1	y_1	$\frac{(y_2-y_1)}{(x_2-x_1)} = [x_2 x_1]$	$[x_3 x_2 x_1]$	$[x_4 x_3 x_2 x_1]$	
x_2	y_2	$\frac{(y_3-y_2)}{(x_3-x_2)} = [x_3 x_2]$	$[x_4 x_3 x_2]$		
x_3	y_3	$\frac{(y_4-y_3)}{(x_4-x_3)} = [x_4 x_3]$			
x_4	y_4				

(3.17)

According to equation 3.12 the value y at the place x be interpolated:

$$
\begin{aligned}
y(x) \approx \quad p(x) = b_0 \quad &+ \quad b_1(x - x_0) \\
&+ \quad b_2(x - x_0)(x - x_1) \\
&+ \quad b_3(x - x_0)(x - x_1)(x - x_2) \\
&+ \quad b_4(x - x_0)(x - x_1)(x - x_2)(x - x_3)
\end{aligned}
\tag{3.18}
$$

This equation also can be used, in order to compute the interpolation function $w = p(x)$ distribution and possibly to plot the function.

3.1.3.2 Equidistant supporting point distribution

The **equidistant supporting point distribution** $x_0, x_1 = x_0 + h, \ldots, x_n = x_0 + nh$ (h is the increment) is given, then the interpolation by the NEWTON method is:

$$p(x) = y_0 + \frac{\Delta y_0}{h}(x-x_0) + \frac{\Delta^2 y_0}{2! \cdot h^2}(x-x_0)(x-x_1) + \ldots + \frac{\Delta^n y_0}{n! \cdot h^n}(x-x_0)\ldots(x-x_{n-1}) \quad (3.19)$$

The elements $\Delta y_0, \Delta^2 y_0, \ldots, \Delta^n y_0$ are called **finite differences**. The exponent does not represent exponentiation, but step by step differences. By comparing the equations 3.19 and 3.12 on page 81 we get:

$$\begin{aligned} b_0 & \quad \widehat{=}\; y_0 \\ b_1 = \frac{y_1 - y_0}{x_1 - x_0} & \quad \widehat{=}\; \frac{\Delta y_0}{h} \\ b_2 = \frac{[x_2 x_1] - [x_1 x_0]}{(x_2 - x_0)} & \quad \widehat{=}\; \frac{\Delta^2 y_0}{2! \cdot h^2} \end{aligned} \quad (3.20)$$

These differences are computed according to the following scheme:

x_0	y_0	$\Delta y_0 = y_1 - y_0$	$\Delta^2 y_0 = \Delta y_1 - \Delta y_0$	$\cdots$	$\Delta^n y_0 = \Delta^{n-1} y_1 - \Delta^{n-1} y_0$
x_1	y_1	$\Delta y_1 = y_2 - y_1$	$\Delta^2 y_1 = \Delta y_2 - \Delta y_1$		
x_2	y_2	$\ldots$	$\ldots$		
$\ldots$	$\ldots$	$\Delta y_{n-2} = y_{n-1} - y_{n-2}$	$\Delta^2 y_{n-2} = \Delta y_{n-1} - \Delta y_{n-2}$		
x_{n-1}	y_{n-1}	$\Delta y_{n-1} = y_n - y_{n-1}$			
x_n	y_n				

The scheme for an example with $n = 4$:

x_0	y_0	$\Delta y_0 = y_1 - y_0$	$\Delta^2 y_0 = \Delta y_1 - \Delta y_0$	$\Delta^3 y_0 = \Delta^2 y_1 - \Delta^2 y_0$	$\Delta^4 y_0 = \Delta^3 y_1 - \Delta^3 y_0$
x_1	y_1	$\Delta y_1 = y_2 - y_1$	$\Delta^2 y_1 = \Delta y_2 - \Delta y_1$	$\Delta^3 y_1 = \Delta^2 y_2 - \Delta^2 y_1$	
x_2	y_2	$\Delta y_2 = y_3 - y_2$	$\Delta^2 y_2 = \Delta y_3 - \Delta y_2$		
x_3	y_3	$\Delta y_3 = y_4 - y_3$			
x_4	y_4				

By back substitution we see, that each finite difference is a combination of the y-values of the first column, e.g.:

$$\Delta^3 y_0 = y_3 - 3y_2 + 3y_1 - y_0 \tag{3.21}$$

3.1.3.3 Examples for the NEWTON method

1. For the function $y_n = f(x_n)$ the values at equidistant points $x_n = x_0 + 2nh$ for $n = -1, 0, 1, 2$ are given (see table):

n	$\mathbf{-1}$	**0**	**1**	**2**
$\mathbf{x}_n$	$x_0 - 2h$	x_0	$x_0 + 2h$	$x_0 + 4h$
$\mathbf{f}(x_n)$	y_{-1}	y_0	y_1	y_2

Find an approximate value $x = \frac{1}{2}$ for $y_{\frac{1}{2}} = f(x_0 + h)$.
Solve this example with the NEWTON method and compare the results with those from the LAGRANGE interpolation formula.

a) linear interpolation:

$$p(x_{\frac{1}{2}}) = y_0 + \frac{\Delta y_0}{\Delta x}(x_{\frac{1}{2}} - x_0)$$

In this example is $\Delta x = 2h$

$$\begin{aligned} p(x_{\frac{1}{2}}) &= y_0 + \frac{y_1 - y_0}{2h}(x_0 + h - x_0) \\ &= \frac{1}{2}(y_0 + y_1) \end{aligned}$$

b) Quadratic interpolation:

$$p(x_{\frac{1}{2}}) = y_0 + \frac{\Delta y_0}{\Delta x}(x_{\frac{1}{2}} - x_0) + \frac{\Delta^2 y_0}{2!\Delta x^2}(x_{\frac{1}{2}} - x_0)(x_{\frac{1}{2}} - x_1)$$

In this example is $\Delta x = 2h$

$$\begin{aligned} p(x_{\frac{1}{2}}) &= \frac{1}{2}(y_0 + y_1) + \frac{y_2 - 2y_1 + y_0}{(2h)^2 2}(x_0 + h - x_0)(x_0 + h - x_0 - 2h) \\ &= \frac{3}{8}y_0 + \frac{3}{4}y_1 - \frac{1}{8}y_2 \end{aligned}$$

It applies:

$$\Delta^2 y_0 = \Delta y_1 - \Delta y_0 = y_2 - y_1 - (y_1 - y_0) = y_2 - 2y_1 + y_0$$

Note:

The advantage of the **Newton** method is that the polynomial $L_i(x)$ does not change if the number of supporting points is changed, i.e. each time we only need to calculate the additional part of the interpolation function.

2. Given the following value table:

x	0	1	2	3	4	5
y	1	2	4	8	15	26

Determine the value $y = f(2,5)$. Choose a polynomial with a suitable order. How large is the deviation if the order of the polynomial is changed? Since the given supporting points are equidistant ($h = 1$), the NEWTON method is applicable to calculate the polynomials with different orders.
First the finite differences are computed:

$x_0 = 0$	$y_0 = 1$					
		$\Delta y_0 = 1$				
$x_1 = 1$	$y_1 = 2$		$\Delta^2 y_0 = 1$			
		$\Delta y_1 = 2$		$\Delta^3 y_0 = 1$		
$x_2 = 2$	$y_2 = 4$		$\Delta^2 y_1 = 2$		$\Delta^4 y_0 = 0$	
		$\Delta y_2 = 4$		$\Delta^3 y_1 = 1$		$\Delta^5 y_0 = 0$
$x_3 = 3$	$y_3 = 8$		$\Delta^2 y_2 = 3$		$\Delta^4 y_1 = 0$	
		$\Delta y_3 = 7$		$\Delta^3 y_2 = 1$		
$x_4 = 4$	$y_4 = 15$		$\Delta^2 y_3 = 4$			
		$\Delta y_4 = 11$				
$x_5 = 5$	$y_5 = 26$					

It is evident that the maximal interpolation polynomial order is third.

a) Linear interpolation
The searched value $x = 2,5$ lies between $x_2 = 2$ and $x_3 = 3$. Therefore the linear interpolation is accomplished only between this tow values

$$\begin{aligned} p(x) &= y_2 + \frac{\Delta y_2}{h}(x - x_2) \\ &= 4 + \frac{4}{1}(2,5 - 2) \\ \underline{p(2,5)} &\underline{= 6} \end{aligned}$$

b) Quadratic interpolation
Since the searched value is $x = 2,5$ the quadratic interpolation can be stretched among x_1, x_2 and x_3.

$$\begin{aligned} p(x) &= y_2 + \frac{\Delta y_2}{h}(x - x_2) + \frac{\Delta^2 y_2}{2!h^2}(x - x_2)(x - x_3) \\ &= 4 + \frac{4}{1}(2,5 - 2) + \frac{3}{2 \cdot 1}(2,5 - 2)(2,5 - 3) \\ &= 4 + 2 - \frac{0,75}{2} \\ \underline{p(2,5)} &\underline{= 5,625} \end{aligned}$$

c) cubic interpolation
The cubic interpolation formula requires three supporting places. In this case both of the triple x_1, x_2 and x_3 or the triple x_2, x_3 and x_4 can be used. For the first case:

$$\begin{aligned} p(x) \quad &= y_1 \quad + \frac{\Delta y_1}{h}(x - x_1) + \frac{\Delta^2 y_1}{2!h^2}(x - x_1)(x - x_2) \\ &\qquad + \frac{\Delta^3 y_1}{3!h^3}(x - x_1)(x - x_2)(x - x_3) \\ &= 2 \quad + \frac{2}{1}(2,5 - 1) + \frac{2}{2 \cdot 1}(2,5 - 1)(2,5 - 2) \\ &\qquad + \frac{1}{6 \cdot 1}(2,5 - 1)(2,5 - 2)(2,5 - 3) \\ &= 2 \quad + 3 + 0,75 - 0,0625 \end{aligned}$$

$$\underline{p(2,5) = 5,6875}$$

For the second triple:

$$\begin{aligned} p(x) \quad &= y_2 \quad + \frac{\Delta y_2}{h}(x - x_2) + \frac{\Delta^2 y_2}{2!h^2}(x - x_2)(x - x_3) \\ &\qquad + \frac{\Delta^3 y_2}{3!h^3}(x - x_2)(x - x_3)(x - x_4) \\ &= 4 \quad + \frac{4}{1}(2,5 - 2) + \frac{3}{2 \cdot 1}(2,5 - 2)(2,5 - 3) \\ &\qquad + \frac{1}{6 \cdot 1}(2,5 - 2)(2,5 - 3)(2,5 - 4) \\ &= 4 \quad + 2 - \frac{0,75}{2} + 0,0625 \end{aligned}$$

$$\underline{p(2,5) = 5,6875}$$

The deviation between the linear and the quadratic result is:

$$\left|\frac{5,625-6}{5,625}\right| = 6,7\%,$$

While the deviation between the square and the cubic result is only:

$$\left|\frac{5,6875-5,625}{5,6875}\right| = 1,1\%$$

In order to estimate the results, the given points can be plotted (see figure 3.3). The diagram shows that: Actually the value should lie between 5

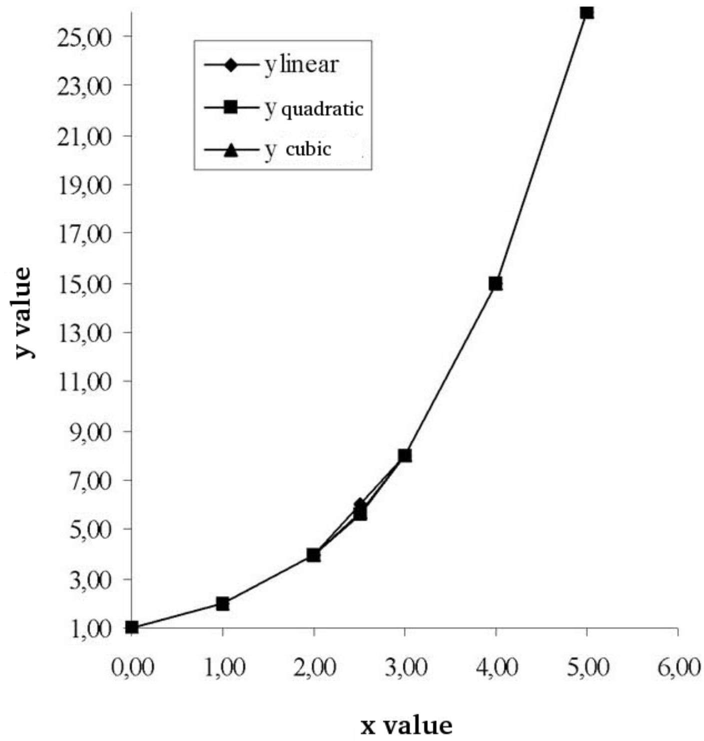

Figure 3.3: Representation of the measured interpolated values

and 6. Obviously the linear interpolation can not yield good results in this case. For this reason it is meaningful to plot given points and estimate the searched value. In a practical work it is important to have enough points in order to get a good approximation of the function. This can be ascertained that, the form of the function substantially does not change when additional points are taken into account.

3.2 Spline interpolation

To describe a given function in a certain interval we can link sections that consist of several lower degree polynomials instead of only one polynomial with high degree. The classical examples are line segments in subintervals (see figure 3.4). It is assumed that the function between two supporting points is nearly linear. This can be applied, if the supporting points are close enough to each other.

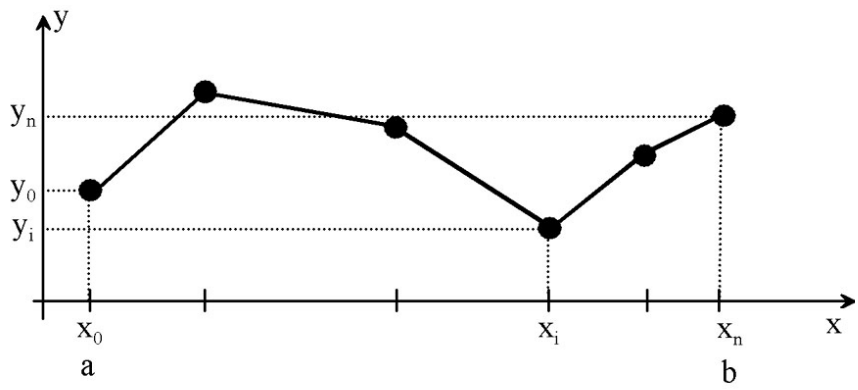

Figure 3.4: Representation of linear spline curves

Such approximations are continuous, however the first derivative is discontinuous, and i.e. vertices appear at the transition part from one interval to another. In the following the **spline interpolation method** will be described, in which cubic polynomials are built up such that the vertices are smooth, so that the first and second derivatives of the approximation are continuous. Polynomials with higher degree are not used since they oscillate strongly.
A given interval $I = (a, b)$ is divided in n **subintervals** by the x-values $x_0 = a < x_1 <$

$x_2 < \ldots < x_n = b$. The cubic polynomial pieces will be fitted in the subintervals such that the y-values y_i at the points x_i match exactly. The first and second derivatives of the functions for the left and for the right subinterval at the data points must be equal (see figure 3.5). The data points (x_i, y_i) are called the nodes of the spline (originally the term spline comes from an flexible spline devices).

A cubic polynomial has four coefficients. Generally it can be written:

$$p_i(x) = c_{0i} + c_{1i}x + c_{2i}x^2 + c_{3i}x^3 \tag{3.22}$$

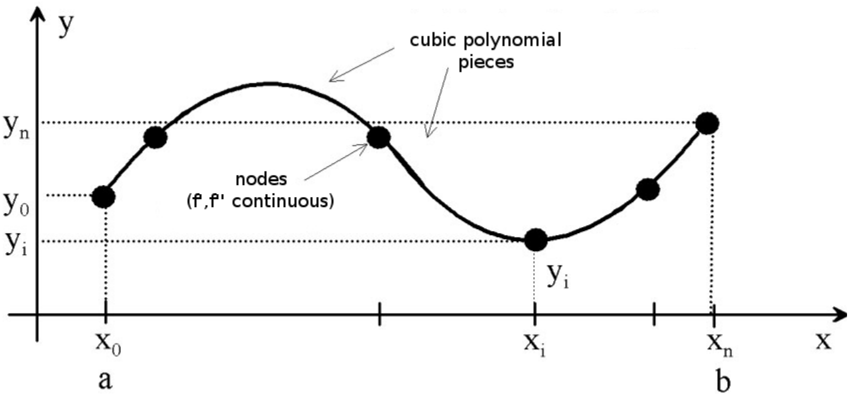

Figure 3.5: Representation of a spline curve with cubic polynomials

The Spline function is defined as follows:

1. The function $S(x)$ is two times continuous differentiable on the interval $[a, b]$.
2. $S(x)$ is given by a cubic polynomial for each interval $[x_i, x_{i+1}]$. I.e.:

$$\begin{aligned} S(x) &:= \sum p_i(x) \\ p_i(x) &:= a_i + b_i(x - x_i) + c_i(x - x_i)^2 + d_i(x - x_i)^3 \end{aligned} \tag{3.23}$$

3. $S(x)$ fulfils the interpolation constraints $S(x_i) = y_i$ for all $i = 1, \ldots, n$ in the interval $[a, b]$.
4. Depending upon the form of connection constraints we get different kinds of spline functions. The following table shows special **cubic spline functions**:

Connecting conditions	**Description**	**Comments**
$S(x_0) = S''(x_0) = 0$ $S(x_n) = S''(x_n) = 0$	**natural**	$S(x_0)$ and $S(x_n)$ is the tangent function at the graph of $S(x)$
$S''(x_0) = \alpha S''(x_n) =$	**generally**	
$S'(x_0) = \alpha S'(x_n) = \beta$	**given**	first derivative at the boundary
$S'''(x_0) = \alpha S'''(x_n) = \beta$	**given**	third derivative at the boundary
$S(x_0) \quad = S(x_n)$ $S'(x_0) \quad = S'(x_n)$ $S''(x_0) \quad = S''(x_n)$	**periodic**	
$p_0(x) \quad = p_l(x)$ $p_{n-2}(x) \quad = p_{n-1}(x)$	**not-a-knot**	$S'''(x_l)$ and $S'''(x_n)$ are continuous

In the case of n intervals it yields to $4n$ coefficients and then $4n$ constraints are expected to compute the spline. At each node (x_i, y_i) $(i = 1, \ldots, n)$ we have 4 constraints (the y value and the correlation of the derivatives). This yields $4n - 4$ constraints. At the boundary points a and b the y values are given and so we have $4n-2$ constraints. The spline is not completely defined. Two degrees of freedom remain.

$$
\begin{aligned}
&p_i(x_i) = y_i && i = 0, \ldots, n && \text{interpolation constraint} \\
&\left.\begin{aligned} p_i(x_i) &= p_{i-1}(x_i) \\ p_i'(x_i) &= p_{i-1}'(x_i) \\ p_i''(x_i) &= p_{i-1}''(x_i) \end{aligned}\right\} && i = 1, \ldots, n-1 && \begin{aligned}&\text{connecting constraints of} \\ &\text{the polynomials } p_i \text{ and } p_{i-1}\end{aligned}
\end{aligned}
\tag{3.24}
$$

We can set the second derivative at the boundary points equal to zero and get a **natural spline**.

$$
\begin{aligned}
p_n(x_n) &= a_n && S(x_0) \text{ and } S(x_n) \text{ are tangents} \\
p_n''(x_n) &= 2c_n && \text{on the graph } S(x)
\end{aligned}
\tag{3.25}
$$

Alternatively the first derivative at the boundary points can be given, in order to approximate a function.
Thus it yields an equation system with $4n$ equations for $4n + 2$ unknown quantities. The two missing equations are covered by the boundary conditions.

$$\begin{aligned} p_0''(x_0) &= 0 \\ p_n''(x_n) &= 0 \end{aligned} \qquad \text{boundary conditions} \tag{3.26}$$

This equation system can be solved according to known methods. Usually the solution of this equation system is complex, so not only one-step but also iterative methods (see section 1.3 solution methods of equation system) must be used. As is shown below, a tridiagonal equation system can be generated by a certain algorithm, so then it can be solved with a little operating expense.

Algorithm

Given $n+1$ data points x_i $(i = 0, \dots, n)$ with an increment $h_i = x_{i+1} - x_i$ and $n+1$ data values y_i (e.g. as a list of measurement), so the following algorithm (see equations 3.24 to 3.26)can be used to compute the **cubic spline function**.

$$\begin{aligned} S(x) &:= \sum_{i=0}^{n-1} p_i(x) \\ p_i(x) &= a_i + b_i(x - x_i) + c_i(x - x_i)^2 + d_i(x - x_i)^3 \end{aligned} \tag{3.27}$$

The cubic spline consist of n functions for the intervals $x_i \le x \le x_{i+1}$.

Step	Calculation	Validity	Comment
1	$a_i = y_i$	$i = 0, \dots, n$	from equation 3.24 interpolation constraint $p_i(x_i) = y_i$ for all data points
2	$c_0 = c_n = 0$		for natural splines: $S''(x_0) = S''(x_n) = 0$ $p''(x) = 2c_i + 6d_i(x - x_i)$ $p''(x_0) = 2c_0 = 0$ $p''(x_n) = 2c_n = 0$
3	$h_{i-1}c_{i-1} + 2c_i\left(h_{i-1} + h_i\right) + h_i c_{i+1} = \frac{3}{h_i}\left(a_{i+1} - a_i\right) - \frac{3}{h_{i-1}}(a_i - a_{i-1})$	$i = 1; \cdots n-1$	from equation 3.24 connection constraints $p_{i+1}(x_{i+1}) = p_i(x_{i+1})$
4	$b_i = \frac{1}{h_i}(a_{i+1} - a_i) - \frac{h_i}{3}(c_{i+1} - 2c_i)$	$i = 0, \dots, n-1$	$p'_{i+1}(x_{i+1}) = p'_i(x_{i+1})$
5	$d_i = \frac{1}{3h_i}(c_{i+1} - c_i)$	$i = 0, \dots, n-1$	$p''_{i+1}(x_{i+1}) = p''_i(x_{i+1})$

The equation in the third step of the algorithm represents a linear equation system of $n-1$ equations for the variables $c_1, \dots, c_{n-1}$. It can be written in the matrix form:

$$\mathbf{A} \cdot \mathbf{C} = \mathbf{R} \tag{3.28}$$

$$\mathbf{A} = \begin{pmatrix} 2(h_0+h_1) & h_1 & & & \\ h_1 & 2(h_1+h_2) & h_2 & & \\ & \ddots & \ddots & \ddots & \\ & & h_{n-3} & 2(h_{n-3}+h_{n-2}) & h_{n-2} \\ & & & h_{n-2} & 2(h_{n-2}+h_{n-1}) \end{pmatrix} \tag{3.29}$$

$$\mathbf{R} = \begin{pmatrix} \frac{3}{h_1}(a_2-a_1) - \frac{3}{h_0}(a_1-a_0) \\ \frac{3}{h_2}(a_3-a_2) - \frac{3}{h_1}(a_2-a_1) \\ \vdots \\ \frac{3}{h_{n-2}}(a_{n-1}-a_{n-2}) - \frac{3}{h_{n-3}}(a_{n-2}-a_{n-3}) \\ \frac{3}{h_{n-1}}(a_n-a_{n-1}) - \frac{3}{h_{n-2}}(a_{n-1}-a_{n-2}) \end{pmatrix} \tag{3.30}$$

$$\mathbf{C} = \begin{pmatrix} c_1 \\ c_2 \\ \vdots \\ c_{n-2} \\ c_{n-1} \end{pmatrix} \tag{3.31}$$

The matrix $\mathbf{A}$ is tridiagonally, symmetrically, diagonally dominant, positively definite and all entries are positive. Thus this matrix is always invertible and definitely solvable. As solution method the GAUSS algorithm for tridiagonal matrices can be used (see section 1.3.1 solutions of equation system, GAUSS algorithm).

Example spline interpolation:

Given the following measured data points and values:

i	0	1	2	3	4
x_i	-1	$-0,5$	0	$0,5$	1
y_i	$0,5$	$0,8$	1	$0,8$	$0,5$

For these 5 pairs find a natural cubic spline.
According to the definition of the spline function, 4 functions

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

$(i = 1, \ldots, 4)$ defined respectively on the intervals $x_i \leq x \leq x_{i+1}$ are searched.

Using the algorithm the following steps are implemented:

Step	**Computation**	**Result**
1	$a_i = y_i$	$a_0 = 0,5$ $a_1 = 0,8$ $a_2 = 1,0$ $a_3 = 0,8$ $a_4 = 0,5$
2	$c_0 = c_4 = 0$	$c_0 = 0$ $c_4 = 0$

Step	**Computation**	**Result**
3	$$\dfrac{\begin{bmatrix} 2(h_0+h_1) & h_1 & 0 \\ h_1 & 2(h_1+h_2) & h_2 \\ 0 & h_2 & 2(h_2+h_3) \end{bmatrix} \cdot \begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix} = \begin{bmatrix} \frac{3}{h_1}(a_2-a_1) - \frac{3}{h_0}(a_1-a_0) \\ \frac{3}{h_2}(a_3-a_2) - \frac{3}{h_1}(a_2-a_1) \\ \frac{3}{h_3}(a_4-a_3) - \frac{3}{h_2}(a_3-a_2) \end{bmatrix}}{\begin{bmatrix} 2(0,5+0,5) & 0,5 & 0 \\ 0,5 & 2(0,5+0,5) & 0,5 \\ 0 & 0,5 & 2(0,5+0,5) \end{bmatrix} \cdot \begin{bmatrix} c_1 \\ c_2 \\ c_3 \end{bmatrix} = \begin{bmatrix} \frac{3}{0,5}(1,0-0,8) - \frac{3}{0,5}(0,8-0,5) \\ \frac{3}{0,5}(0,8-1,0) - \frac{3}{0,5}(1,0-0,8) \\ \frac{3}{0,5}(0,5-0,8) - \frac{3}{0,5}(0,8-1,0) \end{bmatrix}}$$	$c_1 = 0$ $c_2 = -1,2$ $c_3 = 0$
4	$b_i = \dfrac{1}{h_i}(a_{i+1} - a_i) - \dfrac{h_i}{3}(c_{i+1} - 2c_i)$	$b_0 = 0,6$ $b_1 = 0,6$ $b_2 = 0$ $b_3 = -0,6$
5	$d_i = \dfrac{1}{3h_i}(c_{i+1} - c_i)$	$d_0 = 0$ $d_1 = -0,8$ $d_2 = 0,8$ $d_3 = 0$

This yields to the following functions according to equation 3.23:

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

Function	**Interval**
$p_0(x) = 0,5 + 0,6(x + 1)$	$-1 \leq x \leq -0,5$
$p_1(x) = 0,8 + 0,6(x + 0,5) - 0,8(x + 0,5)^3$	$-0,5 \leq x \leq 0$
$p_2(x) = 1,0 - 1,2x^2 + 0,8x^3$	$0 \leq x \leq 0,5$
$p_3(x) = 0,8 - 0,6(x - 0,5)$	$0,5 \leq x \leq 1$

The diagram of the spline is shown in figure 3.6:

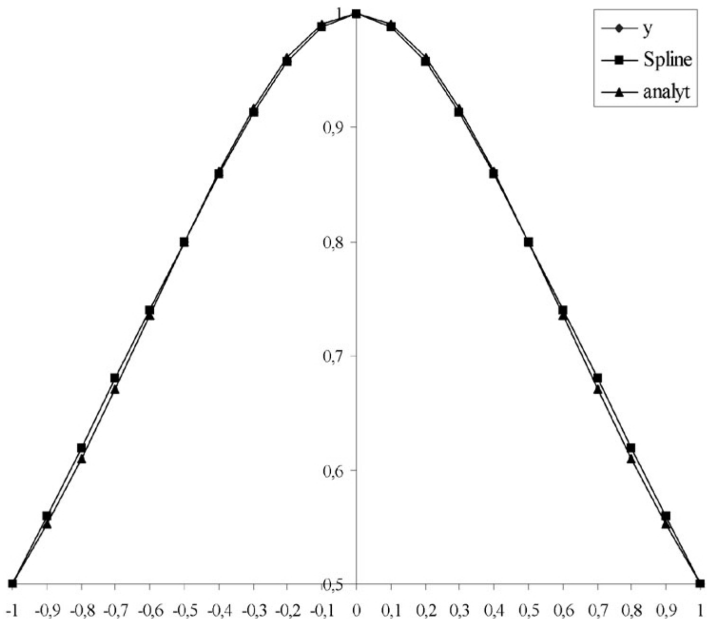

Figure 3.6: Spline interpolation function

We recognize that the spline simulates the original analytic function

$$y = \frac{1}{x^2 + 1}$$

very well. The maximum deviation of analytic solution amounts to 0.010244; which corresponds to 1.68%.

3.3 Kriging method

A family of special interpolation methods is called **Kriging**, which tries to handle with the following problem:
The sampling at a place gives informations for certain spatial points. However it is unknown which values are available for the data points and values between these points. Kriging is a method, which makes possible, to compute the value of an intermediate point or the average over an entire block. The different special methods are all based on the creation of weighted mean values of the spatial variables. Block estimations are predominantly necessary in the mining industry, while point estimations are used for map representations, which is described know.

The individual Kriging methods differ either in the kind of the goal sizes which can be estimated or in their methodical extension for the inclusion of additional information.

Additional information about the spatial behaviour of a location dependent variable exists in the cognition of other measurements, which relates to the observed variables. In hydrogeological practice for instance correlated dissolved matter or temporal repetition measurements of ground- water pressure head are common.

In a word Kriging methods have the following advantages compared to other interpolation procedures:

- Kriging yields the "best" estimated value.
- Kriging involves the information of the spatial structure of the variable and the variogram into the estimation.
- The individual spatial arrangement of the measuring point net is considered with reference to the interpolation grid.
- The reliability of the results is indicated in form of Kriging error for each estimated point.

Note:
Also in the Kriging method it must be paid attention that no information gain can be achieved by the mathematical procedures. Only the information content of the measured values (basic values) is processed. Interpolation results might contradict physical laws (e.g. ground water contour line in receiving streams). If we want to get physically correct interpolations, a fine quantized simulation by means of physical models (e.g. ground-water flow models) is necessary and meaningful. Therefore such simulation programs offer internal diagram routines for creation of isoline.

In order to understand the Kriging procedures, the following terms from the geostatistics must be known:

Mean value	$m =$	$\frac{1}{n}\sum_{a=1}^{n} Z_a$
Expected value	$E[Z] =$	$\int z \cdot p(z)dz = m$ with $p(z)$ the density function
Variance	$\mathrm{var}(Z) =$	$\sigma^2 = E\left[(Z - E[Z])^2\right] = E\left[(Z-m)^2\right]$
Covariance of two random variables $\mathbf{Z}_i, \mathbf{Z}_j$	$\mathrm{cov}(Z_i, Z_j) =$	$E\left[(Z_i - m_i)(Z_j - m_j)\right] = \sigma_{ij}$
correlations coefficient	$\rho_{ij} =$	$\frac{\sigma_{ij}}{\sqrt{\sigma_i^2 \sigma_j^2}}$
Variogramm	$\gamma(\vec{h}) =$	$\frac{1}{2}E\left[\left(Z(\vec{x}+\vec{h}) - Z(\vec{x})\right)^2\right]$

Z is a place dependent random variable with n measured values Z_a. The density function $p(z)$ the probability that Z becomes the value z_i. By computation of the inequality of two values, the variogram shows the variability of a random function, which correspond to points with distance to the vector $\vec{h}$.
Then the Kriging problem can be represented according to figure 3.7:

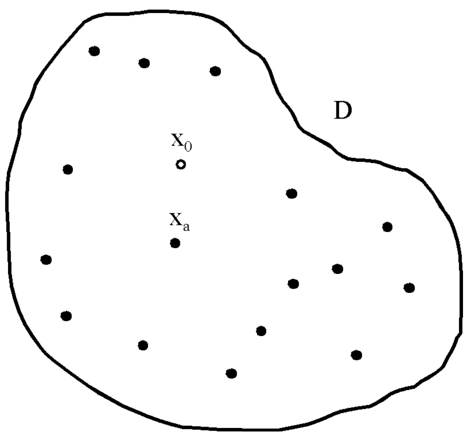

Figure 3.7: Illustration of a Kriging problem

We have a number of measured values $Z(\vec{x}_a)$, whereby Z is a random variable and $\vec{x}_a$ is a measuring point of the range D.
We assume then that $Z(\vec{x}_a)$ is a subset of the random function $Z(\vec{x})$, which has the following characteristics:
It is a second order stationary function, i.e.:

1. The expected value is constant over the range D: $E[Z(\vec{x}+\vec{h})] = E[Z(\vec{x})]$
2. The covariance between two points depends only on the vector $\vec{h}$: $\operatorname{cov}[Z(\vec{x}+\vec{h}), Z(\vec{x})] = C(\vec{h})$

Due to these assumptions we want to compute a weighted mean, in order to get an estimated value for the place $\vec{x}_0$.

The Kriging estimator $Z^*(\vec{x}_0)$ represents a linear combination of weighted sample values Z_i and n of neighbouring points:

$$Z^*(\vec{x}_0) = \sum_{i=1}^{n} \lambda_i Z(\vec{x}_i) \tag{3.32}$$

The weights λ_i are determined in such a way that the estimated value $Z^*(\vec{x}_0)$ of the unknown true value fulfils the following conditions:

1. $Z^*[\vec{x}_0)$ is unbiased, i.e.: $E^*[Z^*(\vec{x}_0) - Z(\vec{x}_0)] = 0$
2. The mean square value $E[Z^*(\vec{x}_0) - Z(\vec{x}_0)]^2$ is minimal.

Assume the stationarity is the expected value $E[Z(\vec{x_i})] = m$ and $Z(\vec{x_0}) = m$. Then the condition 1. yields:

$$E\left[\sum_{i=1}^{n} \lambda_i Z(\vec{x_i}) - Z(\vec{x_0})\right] = \sum_{i=1}^{n} \lambda_i m - m = m(\sum_{i=1}^{n} \lambda_i - 1) = 0 \tag{3.33}$$

From this follows that the sum of the weights must be one.

With the help of the variogram the expected value of the square error can be expressed:

$$\begin{aligned} E\left[Z^*(\vec{x}_0) - Z(\vec{x}_0)\right]^2 &= \operatorname{var}\left(Z^*(\vec{x}_0) - Z(\vec{x}_0)\right) \\ &= 2\sum_{i=1}^{n} \lambda_i \gamma(\vec{x_i} - \vec{x_0}) - \sum_{i=1}^{n}\sum_{j=1}^{n} \lambda_i \lambda_j (\vec{x_i} - \vec{x_j}) - \gamma(\vec{x_0} - \vec{x_0}) \end{aligned} \tag{3.34}$$

In order to minimize the error variance of the side condition 1 ($\sum_{i=1}^{n} \lambda_i = 1$), the LAGRANGE multiplier μ will be introduced. Then the following function is minimized:

$$\varphi = \text{var}(Z^*(\vec{x_0}) - Z(\vec{x_0})) - 2\mu \left(\sum_{i=1}^{n} \lambda_i - 1 \right)$$

We get the minimum by setting of the partial derivative $\frac{\partial \phi}{\partial \lambda_i}$, $(i = 1, \ldots, n)$ and $\frac{\partial \phi}{\partial \mu}$ zero.
These yield a linear Kriging equation system (KES) with $n + 1$ equations:

$$\sum_{j=1}^{n} \lambda_j \gamma(\vec{x_i} - \vec{x_j}) + \mu = \gamma(\vec{x_i} - \vec{x_0}) \qquad \text{for } i = 1, \ldots, n$$

$$\sum_{j=1}^{n} \lambda_j = 1$$

In matrix form the KES is written as follows:

$$\begin{pmatrix} \gamma(\vec{x_1} - \vec{x_1}) & \gamma(\vec{x_1} - \vec{x_2}) & \ldots & \gamma(\vec{x_1} - \vec{x_n}) & 1 \\ \gamma(\vec{x_2} - \vec{x_1}) & \gamma(\vec{x_2} - \vec{x_2}) & \ldots & \gamma(\vec{x_2} - \vec{x_n}) & 1 \\ \vdots & \vdots & \ddots & & \vdots \\ \gamma(\vec{xn} - \vec{x_1}) & \gamma(\vec{x_n} - \vec{x_2}) & \ldots & \gamma(\vec{x_n} - \vec{x_n}) & 1 \\ 1 & 1 & \ldots & 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} \lambda_1 \\ \lambda_2 \\ \vdots \\ \lambda_n \\ \mu \end{pmatrix} = \begin{pmatrix} \gamma(\vec{x_1} - \vec{x_0}) \\ \gamma(\vec{x_2} - \vec{x_0}) \\ \vdots \\ \gamma(\vec{x_n} - \vec{x_0}) \\ 1 \end{pmatrix}$$

In the case of point estimation $\gamma(\vec{x_i} - \vec{x_i}) = \gamma(0) = 0$, i.e. the diagonal entries are zero. Since in the steady case the relationship of $\gamma(\vec{h}) = C(0) - C(\vec{h})$, $\gamma(\vec{h})$ in the KES can be substituted by the covariance $C(\vec{h})$. Thus the diagonal of the matrix emerges large elements. For numerical aspects this is preferable and therefore implemented in most programs.

The Kriging estimate variance σ_K^2 for point estimation results from above equations is:

$$\sigma_K^2 = \text{var}(Z^*(\vec{x_0}) - Z(\vec{x_0})) = \mu + \sum_{i=1}^{n} \lambda_i \gamma(\vec{x_i} - \vec{x_0}) \tag{3.35}$$

In a special case, in which no spatial dependence of the data exists, we get the weights $\lambda_i = \frac{1}{n}$. Then the Kriging estimator is the simple arithmetic means of the neighbouring samples. The following characteristics distinguish the Kriging estimator:

- The KES is only solvable if the determinant of the matrix $(\gamma_{ij}) \neq 0$. Practically this means that a sample can not appear twice (i.e. with identical coordinates).
- Kriging yields an accurate interpolator.
- The KES depends only on $\gamma(\vec{h})$ or $C(\vec{h})$, however not on the values of the variable Z in the points of sample x_i. With identical data configuration the KES only need to be solved once.
- Confidential limits of the estimation can be indicated under the help of the estimation error σ_K

In practice a series of Kriging procedures were developed and applied, which regard more complex situations, e.g. intermittent variable, space time dependence etc.

3.4 Exercises

Exercises to 3:
Interpolate by means of

- analytical power function
- LAGRANGE interpolation formula
- NEWTON interpolation formula
- spline interpolation

1. Given some values of the normal distribution function $y(x) = \frac{e^{-x^2/2}}{\sqrt{2\pi}}$:

x	$1,00$	$1,20$	$1,40$	$1,60$	$1,80$	$2,00$
$y(x) = \frac{e^{-x^2/2}}{\sqrt{2\pi}}$	$0,2420$	$0,1942$	$0,1497$	$0,1109$	$0,0790$	$0,0540$

   Find the value of $y(x)$ for $x = 1,5$.

2. Interpolate the function y=$\sqrt{x}$ for the values $x = 1,03$ and $x = 1,26$ with help of the following basic values:

x	$1,00$	$1,05$	$1,10$	$1,15$	$1,20$	$1,25$	$1,30$
$\mathbf{y} = \sqrt{x}$	$1,00000$	$1,02470$	$1,04881$	$1,07238$	$1,09544$	$1,11803$	$1,14017$

3. Find a rational function with the degree as low as possible through the supporting points $(1,-2)$, $(2,3)$ and $(3,1)$.
   How does this interpolation function change, if another supporting point $(4,4)$ is taken into account?