5. Approximation of functions

Tento učební text byl podpořen z Operačního programu Praha - Adaptabilita

Hana Hladíková

OPPA
PRAHA PRAGUE PRAGA PRAG
EVROPSKÁ UNIE

Evropský sociální fond Praha & EU: Investujeme do vaší budoucnosti

Approximating a function $f(x)$ means substituting it by some function $g(x)$ with some properties. We find a system of simple basis functions $g_0(x),\ g_1(x),\ g_2(x),\ \dots,g_n(x)$. The function $g(x)$ will be then a linear combination of functions g_i, i.e.

$g(x) = c_0 g_0(x) + c_1 g_1(x) + c_2 g_2(x) + \ldots + c_n g_n(x)$. According to the type of the given problem the basis functions and coefficients of linear combination are chosen.

We will study three types of approximation:
(1) a) approximation on a neighborhood of a given point,
(2) b) interpolation,
(3) c) L2 approximation.

Taylor polynomial

Let a function $f(x)$ have continuous derivatives till the n+1st order on an interval $\langle x_0 - h, x_0 + h\rangle$. Then for each element $x \in \langle x_0 - h, x_0 + h\rangle$ we have $f(x) = T_n(x) + R_{n+1}$, where $T_n(x)$ is **Taylor polynomial of degree** n

$$T_n(x) = f(x_0) + \frac{f'(x_0)}{1!}(x-x_0) + \frac{f''(x_0)}{2!}(x-x_0)^2 + \cdots + \frac{f^{(n)}(x_0)}{n!}(x-x_0)^n$$

and R_{n+1} is **Lagrange form of the remainder**:

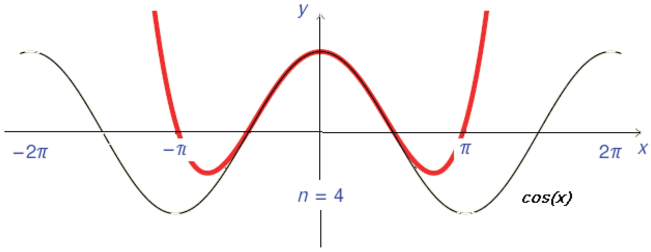
$$R_{n+1} = \frac{f^{(n+1)}(\xi)}{(n+1)!}(x-x_0)^{n+1},\ \xi \in \langle x_0 - h, x_0 + h\rangle\,.$$

Example 1: Approximate the function $f(x) = \cos(x)$ at $x_0 = 0$ by the Taylor polynomial of the 4th degree. Find all x's for which the error is smaller then 10^{-3}.

$$T_4(x) = \cos(0) + \frac{\cos'(0)}{1!}(x-0) + \frac{\cos''(0)}{2!}(x-0)^2 + \frac{\cos^{(3)}(0)}{3!}(x-0)^3 + \frac{\cos^{(4)}(0)}{4!}(x-0)^4 =$$

$$= \cos(0) - \frac{\sin(0)}{1!}x - \frac{\cos(0)}{2!}x^2 + \frac{\sin(0)}{3!}x^3 + \frac{\cos(0)}{4!}x^4 = 1 - \frac{x^2}{4} + \frac{x^4}{24}$$

$$|\sin(x)| \le 1 \Rightarrow |R_5| = \left|\frac{\cos^{(5)}(\xi)}{(5)!}(x-0)^5\right| = \left|\frac{-\sin(\xi)}{120}x^5\right| \le \frac{1}{120}\left|x^5\right| \le 10^{-3} \Rightarrow |x| \le 0,65$$



Obr. 5.1 Taylorův polynom 4. stupně pro funkci $f(x) = \cos(x)$ v bodě $x_0 = 0$

.

Interpolation

Let us for $n+1$ points $x_0,\ x_1,\ x_2,\dots,x_n \in D(f)$ have $n+1$ values
$y_0=f(x_0),\ y_1=f(x_1),\ \dots,y_n=f(x_n)$. For various classes of functions $\Phi\left(x,\ a_0,\ a_1,\ a_2,\dots,a_n\right)$ of one variable x that are determined by the coefficients $a_0,\ a_1,\ a_2,\dots,a_n$ we will find the function $g(x)$ such that $g\left(x_i,\ a_0,\ a_1,\ a_2,\dots,a_n\right)=y_i$. The graphs of the functions $f(x)$ and $g(x)$ go through the points $[x_i,\ y_i]$ for $i=0,\dots,n$. These points are called **poles**, and, thus the method is called **interpolation of function** - substituting the function between poles. According to the chosen classes of functions we get polynomial, spline, exponential ... interpolation.

Polynomial interpolation

Interpolating polynomial

Polynomial of degree at most n is the function $P_n(x)=a_0+a_1\,x+a_2\,x^2+\ldots+a_n\,x^n$,
$x,\quad a_i\in\mathbb{R},\ n\in\mathbb{N}$.

Let us have $n+1$ points $[x_i,\ y_i]$ and let $x_i<x_{i+1}$. Then *there exists exactly one* polynomial $P_n(x)$ of degree at most n with $P_n(x_i)=y_i,\ \ i=0\ldots n.$. This polynomial is called **interpolating polynomial**.

If we express equations $P_n(x_i)=y_i$ we get $n+1$ equations with $n+1$ unknowns:

$$\begin{pmatrix} 1 & x_0 & \ldots & (x_0)^{n-1} & (x_0)^n \\ 1 & x_1 & \ldots & (x_1)^{n-1} & (x_1)^n \\ \vdots & \vdots & \ddots & \vdots & \\ 1 & x_n & \ldots & (x_n)^{n-1} & (x_n)^n \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_n \end{pmatrix} = \begin{pmatrix} y_0 \\ y_1 \\ \vdots \\ y_n \end{pmatrix}$$

The matrix of the system has a non-zero determinant (**Vandermonde determinant**), and the system has a unique solution. This method (**method of undetermined coefficients**) is *badly conditioned* and with a greater number of poles the number or operations is too great. There are other ways of computing interpolating polynomials - for instance Newton and Lagrange interpolating polynomials. The difference is only in the way of computing, we get identical polynomials (see the foregoing statement).

Example 2: Find the interpolating polynomial for $f(x)$ (given by the chart below) by the method of undetermined coefficients:

i	0	1	2
x_i	0	1	3
y_i	1	2	0

$$\begin{pmatrix} 1 & 0 & 0 \\ 1 & 1 & 1^2 \\ 1 & 3 & 3^2 \end{pmatrix} \begin{pmatrix} a_0 \\ a_1 \\ a_2 \end{pmatrix} = \begin{pmatrix} 1 \\ 2 \\ 0 \end{pmatrix} \Rightarrow$$

$$\Rightarrow a_0=1,\ a_1=\frac{5}{3},\ a_2=-\frac{2}{3}\Rightarrow P_2(x)=1+\frac{5}{3}x-\frac{2}{3}x^2.$$

Lagrange interpolating polynomial

Lagrange interpolating polynomial of degree n, denoted by $L_n(x)$, will be of the form

$$L_n(x) = \sum_{i=0}^{n} y_i \cdot l_i(x), \quad l_i(x) = \frac{(x-x_0)(x-x_1)\dots(x-x_{i-1})(x-x_{i+1})\dots(x-x_n)}{(x_i-x_0)(x_i-x_1)\dots(x_i-x_{i-1})(x_i-x_{i+1})\dots(x_i-x_n)}.$$

It is clear that $l_j(x_j) = 1$ and $l_i(x_j) = 0$ for $i \neq j$, thus $L_n(x)$ is an interpolating polynomial of degree n with the poles $[x_i,\ y_i]$.

Example 3: Find the Lagrange interpolating polynomial for $f(x)$ from the previous example.

$$L_2(x) = \sum_{i=0}^{2} y_i \cdot l_i(x) = 1\frac{(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)} + 2\frac{(x-x_0)(x-x_2)}{(x_1-x_0)(x_1-x_2)} + 0\frac{(x-x_0)(x-x_1)}{(x_2-x_1)(x_2-x_1)} =$$

$$= 1\frac{(x-1)(x-3)}{(0-1)(0-3)} + 2\frac{(x-0)(x-3)}{(1-0)(1-3)} + 0\frac{(x-0)(x-1)}{(3-0)(3-1)} \Rightarrow L_2(x) = 1 + \frac{5}{3}x - \frac{2}{3}x^2$$

Newton interpolation polynomial

First define **relative difference** of degree k for a function $f(x)$ with poles $[x_i,\ f(x_i)]$:
$f[x_i] = y_i,\ k = 0,$
$f[x_{i+1}, x_i] = \frac{y_{i+1}-y_i}{x_{i+1}-x_i},\ k = 1,$
$f[x_{i+k}, x_i] = \frac{f[x_{i+k},x_{i+1}]-f[x_{i+k-1},x_i]}{x_{i+k}-x_i},\ 1 < k \le n.$

Newton interpolating polynomial of degree n, denoted by $N_n(x)$, is of the form:

$$N_n(x) = f[x_0] + f[x_1, x_0](x-x_0) + f[x_2, x_0](x-x_0)(x-x_1) + \ldots + f[x_n, x_0](x-x_0)(x-x_1)\ldots(x-x_n)$$

Example 4: Find the Newton interpolating polynomial for $f(x)$ from the previous examples.

i	x_i	$f(x_i) = f[x_i]$	$f[x_i, x_{i-1}]$	$f[x_i, x_{i-2}]$
0	0	1		
1	1	2	$\frac{2-1}{1-0} = 1$	
2	3	0	$\frac{0-2}{3-1} = -1$	$\frac{-1-1}{3-0} = -\frac{2}{3}$

$$N_2(x) = f[x_0] + f[x_1, x_0](x-0) + f[x_2, x_0](x-0)(x-1) =$$
$$= 1 + 1(x-0) - \frac{2}{3}(x-0)(x-1) = 1 + \frac{5}{3}x - \frac{2}{3}x^2$$

Approximation error for the interpolation

Let the function $f(x)$ have continuous derivatives up to order $n+1$ in the interval $\langle x_0, x_n\rangle$, $x_i \in \langle x_0, x_n\rangle$, $x_i < x_j,\ i < j$ and $P_n(x)$ be an interpolating polynomial. Then for each $x \in \langle x_0, x_n\rangle$ there exists $\xi \in (x_0, x_n)$ such that:

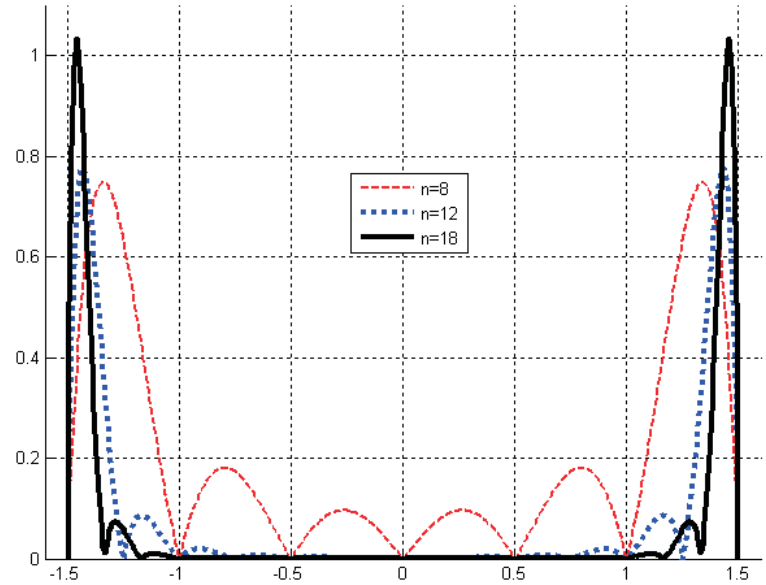
$$f(x) - P_n(x) = \frac{f^{(n+1)}(\xi)}{(n+1)!}\omega_n(x), \text{ where } \omega_n(x) = (x-x_0)(x-x_1)\ldots(x-x_n).$$

If $\left|f^{n+1}(x)\right| \leq M, x \in \langle x_0, x_n \rangle$, then

$$|f(x) - P_n(x)| \leq \frac{\left|f^{(n+1)}(\xi)\right|}{(n+1)!} |\omega_n(x)| \leq \frac{M}{(n+1)!} |\omega_n(x)| .$$

R e m a r k:

(1) The error depends on $\omega_n(x)$, and it does not decrease with increasing number of poles.



Obr. 5.2 $|\omega_n(x)|$.

In the figure 5.2 there is the function $|\omega_n(x)|$ for various n from the interval $\langle -1,5; 1,5 \rangle$ and with 7 equidistant poles (the distance between poles is the same).

(2) Functions with oblique asymptotes should be better interpolated by spline functions.

Example 5: *Runge's phenomenon*

Let us consider the interval $\langle -5; 5 \rangle$ with 11 equidistant points: $x_i = -5 + i, i = 0, \ldots, 10$. The function $f(x) = \frac{1}{1+x^2}$ (Runge function) has the interpolating polynomial $P_{10}(x)$.

Estimate the error of the interpolation. From the graph of $P_{10}(x)$ we can see that there is a strong osciation in the neighborhoods of the endpoints. In Figure 5.3 graphs of $f(x)$ and $P_{10}(x)$ are compared.

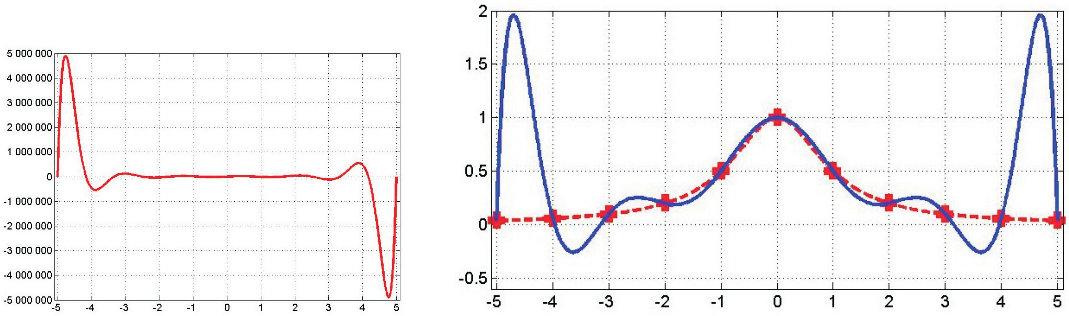


Fig. 5.3 Runge's phenomenon $|\omega_{11}(x)|$ (left fig.), Runge function a $P_{10}(x)$.

MATLAB – Polynomial interpolation

In MATLAB if we want to interpolate some given date by a polynomial of degree n we can use the function **polyfit**.

```
>> P=polyfit(X,Y,n),
```

X is a vector with $n+1$ values of independent (input) variable, Y is a vector with $n+1$ values of dependent (output) variable, n is the degree of a polynomial interpolating points $[x_i, y_i]$, and then P is the vector of coefficients of the interpolating polynomial $P(x)$, $P(x) = p(1)x^n + p(2)x^{n-1} + \ldots + p(n)x + p(n+1)$.

The values of $P(x)$in all components of the vector X (e.g. for making a plot) can be found by the function **polyval**:

```
>> YAP=polyval(P,X)
```

P is the vector of coefficients of the approximating polynomial, X is the vector of values of independent variable and YAP is the vector of values of the approximating polynomial.

Spline interpolation

Divide the given interval into subintervals and on them find approximating polynomials, we get **piecewise polynomial function**.

From now assume that we have $n+1$ poles $[x_i,\ y_i]$ for $i = 0, \ldots, n$ with $x_i \in \langle x_0, x_n \rangle,\ x_i < x_j,\ i < j$. Denote $h_i = x_{i+1} - x_i,\ i = 0, \ldots, n-1$.

Spline function of degree m is a function satisfying the following conditions:

(1) piecewise polynomial of degree m,
(2) *continuous up to degree* $m-1$ (derivatives up to degree $m-1$ are continuous).

Linear spline function

Linear spline function is a function $S_1(x)$ that is continuous on $\langle x_0, x_n \rangle$ and *polynomial of the first degree* on each interval $\langle x_i, x_{i+1} \rangle,\ i = 0, \ldots, n$. The graph of the function $S_1(x)$ is a polyline.

For a linear spline function $S_1(x)$ we have $S_1(x_i) = y_i,\ i = 0, \ldots, n$. Then

$$S_1(x) = \frac{y_{i+1}(x - x_i)}{h_i} + y_i \left(1 - \frac{(x - x_i)}{h_i}\right),\ x \in \langle x_i, x_{i+1} \rangle,\ i = 0, \ldots, n.$$

Quadratic spline function

Quadratic spline function is a function $S_2(x)$ with the following properties:

(1) 1) $S_2(x)$ is quadratic polynomial on each interval $\langle x_i, x_{i+1}\rangle$,
(if we denote it by $P_i(x)$ then $P_i(x) = S_2(x),\ x \in \langle x_i, x_{i+1}\rangle$

(2) 2) $S_2(x)$ is continuous and has continuous first derivative for $x \in \langle x_0, x_n\rangle$.

The most used splines are cubic spline functions:

Cubic spline functions

Cubic spline function is a function $S_3(x)$ with the following properties:

(1) 1) $S_3(x)$ is a cubic polynomial on each interval $\langle x_i, x_{i+1}\rangle$,
(if we denote it by $P_i(x)$ then $P_i(x) = S_3(x),\ x \in \langle x_i, x_{i+1}\rangle$

(2) 2)$S_3(x)$ is continuous and has continuous the first and second derivative for $x \in \langle x_0, x_n\rangle$.

Then:

(1) $P_i(x_i) = y_i,\ i = 0, \dots, n$ ($n+1$ conditions for equalities of function values),

(2) $P_i(x_{i+1}) = P_{i+1}(x_{i+1}),\ i = 0, \dots, n-2$ ($n-1$ conditions for continuity),

(3) $P_i^{'}(x_{i+1}) = P_{i+1}^{'}(x_{i+1}),\ i = 0, \dots, n-2$ ($n-1$ conditions for equalities of first derivatives),

(4) $P_i^{''}(x_{i+1}) = P_{i+1}^{''}(x_{i+1}),\ i = 0, \dots, n-2$ ($n-1$ conditions for equalities of second derivatives).

On each of n intervals we have a cubic polynomial with 4 unknown parameters. Then we need $4n$ parameters together. We have only $3(n-1) + (n+1) = 4n-2$ conditions. We need two more conditions so that the spline function was uniquely determined.

According to these conditions there exist many types of cubic splines, we will study the two types that are used mostly:

Cubic spline with - $S^{''}(x_0) = S^{''}(x_n) = 0$ ("natural conditions") is called **natural cubic spline function**.

Cubic spline with - $S^{'}(x_0) = f^{'}(x_0)$, $S^{'}(x_n) = f^{'}(x_n)$ ("tangent conditions"), where $f(x)$ is the interpolated function (then $f(x_i) = y_i$) is called **complete cubic spline function**.

R e m a r k: Cubic spline function $S_3(x)$ is continuous, has continuous the first and second derivatives on $\langle x_0, x_n\rangle$. It has the same values in poles, but the values of the first and second derivatives in poles can differ.

Existence, uniqueness and error of approximation for natural cubic spline functions (resp. complete spline functions)

For a function $f(x)$ that is defined on $\langle x_0, x_n\rangle$ there exists a unique natural cubic spline function.

For a function $f(x)$ that is defined on $\langle x_0, x_n\rangle$ there exists a unique complete cubic spline function.

Let a function $f(x)$ has continuous derivatives up to the 4th degree on $\langle x_0, x_n\rangle$, let $S_{3_p}(x)$ be its natural spline and $S_{3_u}(x)$ complete spline with poles $[x_i, f(x_i)]$ and partition $x_i \in \langle x_0, x_n\rangle$, $x_i < x_j,\ i < j$, and let $h_i = x_{i+1} - x_i$, where $C = \frac{max\{h_i\}}{min\{h_i\}},\ i = 0, \dots, n$. Then there is such a constant K that:

(1) For the natural spline S_{3_p}

$$\left|f(x) - S_{3_p}(x)\right| \le C \cdot K \cdot (max\,\{h_i\})^2 .$$

(2) For complete spline S_{3_u}

$$\left|f^{(k)}(x) - S_{3_u}^{(k)}(x)\right| \leq C \cdot K \cdot (max\,\{h_i\})^{4-k}\,, \quad k = 0, \ldots, 3.$$

Example 6: Runge's phenomenon

Consider $\langle -5; 5\rangle$ with an equidistant partition with 11 poles: $x_i = -5 + i, i = 0 \ldots 10$. Make the plot of the natural spline $S_{3_p}(x)$ for the function $f(x) = \frac{1}{1+x^2}$ (Runge funkction). It is obvious that this interpolation oscillates less then the interpolating polynomial. In Figure 5.4 we compare the graph of $f(x)$ (left) and $S_{3_p}(x)$ (right), they are almost the same:

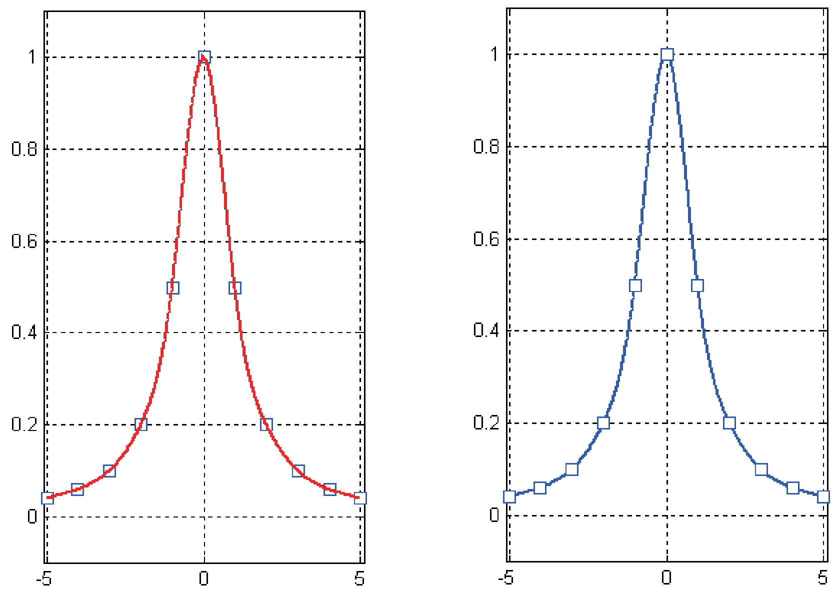


Fig. 5.4 Runge function (left), $S_{3_p}(x)$ (right)