International Journal of Pure and Applied Mathematics
Volume 73 No. 2 2011, 219-225

INTERPOLATION OF FUNCTIONS WITH APPLICATION SOFTWARE

Lazim Kamberi[1 §], Teuta Zenku[2]
[1]Ivo Ribar Llolla, No. 103
Tetova 1200, MACEDONIA
[2]Mehmed Mehmedi, No. 31
Krçova, MACEDONIA

Abstract: In this paper is given the problem of interpolation functions at general and of the interpolation function of Lagrange in particular. With some concrete examples is made illustration of finding interpolation polynomial with Lagrange method, then was introduced an application software that is used quite for solving a lot of math problems, *WolframMathematica*, which quickly and accurately makes finding interpolation requested polynomial.

Finally with the same application software, which has the power of programming in it, I developed a program that most clearly shows the form of getting the interpolation polynomial of Lagrange, depending on number n and the points x_i and y_i for $i \leq n$.

AMS Subject Classification: functions, interpolation, polynomial interpolation, Lagrange method, *WolframMathematica*
Key Words:

1. Introduction

The most basic problem of interpolation arises as follows: In a segment $[a, b]$ are given $n+1$ points $x_0, x_1, \dots, x_n$ that are called nodes of interpolation and the respective values $y_0 = f(x_0)$, $y_1 = f(x_1), \dots, y_n = f(x_n)$ of a function f. Required a simple function, in our case a polynomial P_n of degree less or equal to n that have the same values with function f at interpolation nodes, namely:

$$P_n(x_i) = f(x_i) \quad \text{for} \quad i = 0, 1, 2, \dots, n.$$

Received: August 23, 2011

§Correspondence author

2. Lagrange Interpolation Polynomial

2.1. Interpolation polynomial. Form of Lagrange

If greater precision is required, which can not achieved with linear interpolation, than the interpolation is perform with a polynomial of higher degree. Here will give a general formula, called Lagrange interpolation formula, for finding interpolation polynomial. These problems were treated in references [1], [3], [4]. The formula was first published by Warning (1779), rediscovered by Euler in 1783, and published by Lagrange in 1795.

Let be given $n+1$ different values of the argument: $x_0, x_1, \ldots, x_n$ (does not mean to be with the same distance) and for function $y = f(x)$ let be known corresponding values:

$$y_0 = f(x_0),\, y_1 = f(x_1), \ldots,\, y_n = f(x_n)$$

Required polynomial $P(x)$ with degree not greater than n, so that:

$$P(x_0) = y_0,\, P(x_1) = y_1,\, \ldots,\, P(x_n) = y_n$$

If we get that:

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

need, i.e., arrange coefficients a_k $(k = 0, 1, \ldots, n)$. By conditions $P(x_k) = y_k$ we get system of linear equations as follows:

$$\begin{aligned} &a_0 + a_1 x_0 + \cdots + a_n x_0^n = y_0, \\ &a_0 + a_1 x_1 + \cdots + a_n x_1^n = y_1, \\ &\cdots \quad \cdots \quad \cdots \quad \cdots \quad \cdots \\ &a_0 + a_1 x_n + \cdots + a_n x_n^n = y_n. \end{aligned}$$

Determinant of this system, the so-called determinant of Wandermond, is different from zero because $x_j \neq x_k \; for \; j \neq k$.

Accordingly, the abovementioned system has unique solution (which can arrange, for example, with the Cramer rule).

Polynomial:

$$L_k(x) = \frac{(x - x_0)\cdots(x - x_{k-1})(x - x_{k+1})\cdots(x - x_n)}{(x_k - x_0)\cdots(x_k - x_{k-1})(x_k - x_{k+1})\cdots(x_k - x_n)}$$

or, short,

$$L_k(x) = \prod_{\substack{i=1 \\ i \neq k}}^{n} \frac{x - x_i}{x_k - x_i}$$

has degree n and properties:

$$L_k(x_k) = 1, \;\; L_k(x_i) = 0, \;\; i \neq k$$

(polynomials $L_k(x)$ are called coefficients of Lagrange).

If we list a:

$$L(x) = y_0 L_0(x) + y_1 L_1(x) + \cdots + y_n L_n(x) = \sum_{k=1}^{n} y_k L_k(x) \quad \cdots (1)$$

then we get that $P(x)$ is polynomial with the required properties:

$$P(x_k) = y_k \; (k = 0, \, 1, \ldots, \, n)$$

For more $P(x)$ is unique.

Indeed, if $Q(x)$ is polynomial with degree $\leq n$ and $Q(x_k) = y_k \, , k = 0, \, 1, \ldots, \, n$ then we get: $R(x) = P(x) - Q(x)$ polynomial with degree not greater then n, and for which:

$$R(x_k) = P(x_k) - Q(x_k) = y_k - y_k = 0 \; (k = 0, \, 1, \ldots, \, n)$$

Since a polynomial of degree n may has at least n root, while $R(x)$ has $n+1$, follows that $R(x)$ is zero polynomial, so $Q(x) = P(x)$.

Thus we prove the following theorem:

Theorem 2.1.1. *If $x_0, \, x_1, \ldots, \, x_n$ are different real numbers and $y_0, y_1, \ldots,$ y_n real arbitrary numbers, then there exist a given polynomial $P(x)$ with the property $P(x_k) = y_k, \;\; k = 0, \, 1, \ldots, \, n$ and degree $P \leq n$.*

Polynomial $P(x)$ is defined by formula (1).

Formula (1) is called the Lagrange interpolation formula.

To emphasize that for $n = 1$ (means $x_0 = a, \, x_1 = b$) we get:

$$P(x) = \frac{x-b}{a-b} y_0 + \frac{x-a}{b-a} y_1 \quad ,$$

end this is the formula for linear interpolation.

For $n = 2$ (i.e. $x_0 = a, \, x_1 = b, \, x_2 = c$), again, we get:

$$P(x) = \frac{(x-b)(x-c)}{(a-b)(a-c)} y_0 + \frac{(x-a)(x-c)}{(b-a)(b-c)} y_1 + \frac{(x-a)(x-b)}{(c-a)(c-b)} y_2,$$

which expresses formula for quadratic interpolation ([1], [3]).

2.2. Problems

Let us illustrate how it can be found the Lagrange interpolation polynomial with some examples solved by the above formulas. Such examples are given and solved in references [2], [6].

Problem 2.2.1. *Through the tree points* $(1,17)$, $(3,10)$, $(7,-5)$ *there passes a unique quadratic polynomial. Find it.*

Solution.

$$\begin{aligned}P(x) =&17\left(\frac{(x-3)(x-7)}{(1-3)(1-7)}\right)+10\left(\frac{(x-1)(x-7)}{(3-1)(3-7)}\right)+(-5)\left(\frac{(x-1)(x-3)}{(7-1)(7-3)}\right)\\ =&17\left(\frac{x^2-10x+21}{12}\right)+10\left(-\frac{x^2-8x+7}{8}\right)-5\left(\frac{x^2-4x+3}{24}\right)\\ =&-\frac{x^2}{24}-\frac{10x}{3}-\frac{163}{8}.\end{aligned}$$

Problem 2.2.2. Construct interpolation polynomial of Lagrange for given function f:

i	0	1	2	3
x_i	0	0.1	0.3	0.5
y_i	-0.5	0	0.2	1

Solution. Since we have four points, polynomial degree will be three, so:

$$L_3(x)=\sum_{i=0}^{3} l_i(x)y_i.$$

Calculate polynomials l_i :

$$\begin{aligned}l_0(x) =&\frac{(x-x_1)(x-x_2)(x-x_3)}{(x_0-x_1)(x_0-x_2)(x_0-x)}\\ =&\frac{(x-0.1)(x-0.3)(x-0.5)}{(0-0.1)(0-0.3)(0-0.5)}\\ =&-\frac{x^3-0.9x^2+0.23x-0.012}{0.015}.\end{aligned}$$

So,

$$l_0(x)=-\frac{x^3-0.9x^2+0.23x-0.012}{0.015}.$$

$l_1(x)$ not calculated because multiply by zero.

$$\begin{aligned} l_2(x) =& \frac{(x-x_0)(x-x_1)(x-x_3)}{(x_2-x_0)(x_2-x_1)(x_2-x_3)} \\ =& \frac{(x-0)(x-0.1)(x-0.5)}{(0.3-0)(0.3-0.1)(0.3-0.5)} \\ =& -\frac{x^3-0.6x^2+0.05x}{0.012}. \end{aligned}$$

So,

$$l_2(x) = -\frac{x^3-0.6x^2+0.05x}{0.012}.$$

$$\begin{aligned} l_3(x) =& \frac{(x-x_0)(x-x_1)(x-x_2)}{(x_3-x_0)(x_3-x_1)(x_3-x_2)} \\ =& \frac{(x-0)(x-0.1)(x-0.3)}{(0.5-0)(0.5-0.1)(0.5-0.3)} \\ =& \frac{x^3-0.4x^2+0.03x}{0.04}. \end{aligned}$$

So,

$$l_3(x) = \frac{x^3-0.4x^2+0.03x}{0.04}.$$

By replacing at $L_3(x) = \sum_{i=0}^{3} l_i(x)y_i$ found polynomial and performing action, we get:

$$\begin{aligned} L_3(x) =& \frac{500}{12}x^3 - 30x^2 + \frac{91}{12}x - 0.5 \\ =& 41.66666667x^3 - 30x^2 + 7.583333333x - 0.5. \end{aligned}$$

3. Aplication of WolframMathematica for Interpolating Polynomial

To save time and provide less attempt we can use application software *WolframMathematica* for specific an interpolation polynomial we do this by order *InterpolatinPolynomial [data, var.]*, (such a problems are also treated in [5]). This would best be explained with the upper Problem 2.2.2, i.e.

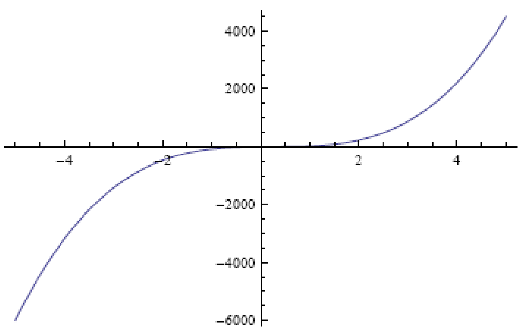
```
In[1]:= InterpolatingPolynomial[{{0, -0.5}, {0.1, 0}, {0.3, 0.2}, {0.5, 1}}, x]

Out[1]= 1 + (-0.5 + x) (3. + (3.33333 + 41.6667 (-0.3 + x)) x)
```

Graph of this polynomial is:

```
In[1]:= Plot[InterpolatingPolynomial[{{0, -0.5}, {0.1, 0}, {0.3, 0.2}, {0.5, 1}}, x], {x, -5, 5}]
```

Out[1]=



4. Our Results

To save time I have tried to construct an program at application software *WolframMathematica* for finding the interpolation polynomial at Lagrange form which will function according formula (1), and will account Lagrange interpolation polynomial, depending on n and the points x_i and y_i for $i \leq n$.

For Problem 2.2.2. the programe in *WolframMathematica* is:

```
In[8]:= n = 3;
        For [i = 0, i ≤ n, i++, x[i] = Input["x[i]"]; y[i] = Input["y[i]"]];
        l[k_] := (Product[(x1 - x[i]), {i, 0, k - 1}] Product[(x1 - x[i]), {i, k + 1, n}]) /
           (Product[(x[k] - x[i]), {i, 0, k - 1}] Product[(x[k] - x[i]), {i, k + 1, n}]);
          L3[x_] = Simplify[Sum[l[k] y[k], {k, 0, n}]];
          Print["Polinomi interpolues i Lagranzhit eshte: ", L3[x]];

Polinomi interpolues i Lagranzhit eshte: 41.6667 (-0.1 + x) (0.12 - 0.62 x + x^2)
```

5. Conclusion

In connection with interpolation polynomial is possible with this application software to program other programs for their appointment for example with the method: Aitkin, Newton, divided differences etc. and their graphic presentation rapidly and accurately in a given interval according to our wishes.

References

[1] Analizë Numerike, *Fatmir Hoxha*, Tiranë (2008), 63-111.

[2] Fatmir Hoxha, *Ushtrime nga Analiza Numerike*, Tiranë (1997), 152-178.

[3] Branko Trpenkovski, Naum Cellakoski, *Elemente të Matematikës Numerike*, Shkup (1992).

[4] Douglas N. Arnold, *A Concise Introduction to Numeric Analysis*, Minneapolis (2001), 1-48.

[5] Branden Archer, Eric W. Weisstein, Lagrange interpolating polynomial, *MathWorld – A Wolfram Web Resource*, http://mathworld.wolfram.com/LagrangeInterpolatingPolynomial.html

[6] Martha L. Abell, James P. Braselton, *Mathematica by Example*, Academic Press (2004).

226