NEWTON’S FORWARD INTERPOLATION:

Introduction:

Interpolation is the technique of estimating the value of a function for any intermediate value of the independent variable, while the process of computing the value of the function outside the given range is called extrapolation.

Interpolation refers to the process of creating new data points given within the given set of data. The above MATLAB code computes the desired data point within the given range of discrete data sets using the formula given by Gauss.

The Gaussian interpolation comes under the Central Difference Interpolation Formulae which differs from Newton's Forward interpolation formula.

Interpolation, which is the process of computing intermediate values of a function from the

set of given values of the function {Hummel (1947), Erdos & Turan (1938) et al}, plays

significant role in numerical research almost in all branches of science, humanities,

commerce and in technical branches. A number of interpolation formulas namely Newton’s

Forward Interpolation formula, Newton’s Backward Interpolation formula, Lagrange’s

Interpolation formula, Newton’s Divided Difference Interpolation formula, Newton’s Central

Difference Interpolation formula, Stirlings formula, Bessel's formula and some others are

available in the literature of numerical analysis {Bathe & Wilson (1976)

[1]

, Jan (1930),

Hummel (1947) et al}.

In case of the interpolation by the existing formulae, the value of the dependent variable

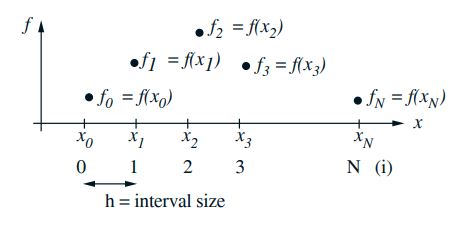
corresponding to each value of the independent variable is to be computed afresh from the

used formula putting the value

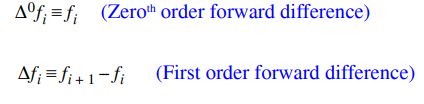
What is forward interpolation?

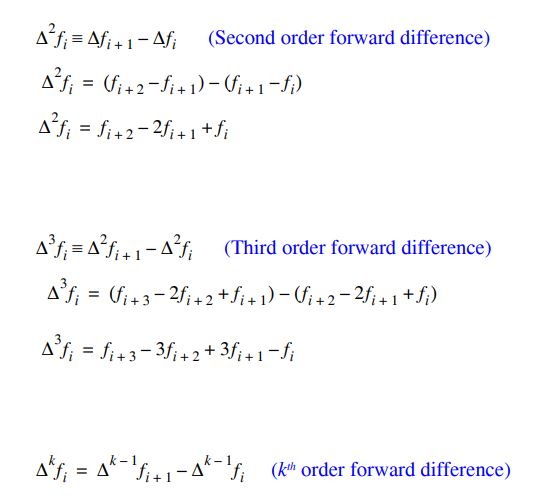
Newton's forward difference formula is a finite difference identity giving an interpolated value between tabulated points in terms of the first value and the powers of the forward difference.

Method derivation:

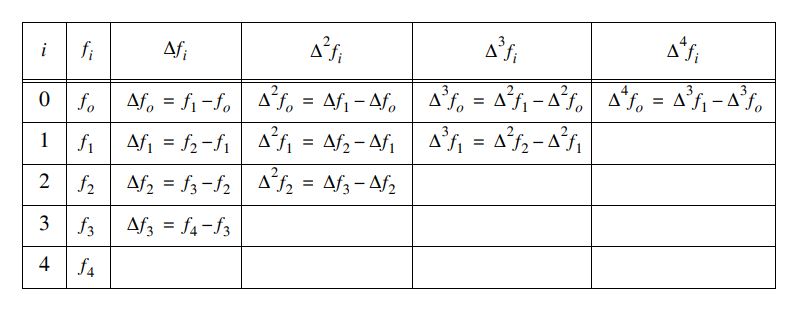


Forward differences are now defined as follows:





Newton’s Forward Difference Table:



• Formula of Newton’s Forward Interpolation:



Problem: Find the Number of Approximate Children born Rate in Bangladesh in the year of x=2013.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| X ( year) | 2010 | 2012 | 2014 | 2016 |
| Y (born rate) | 23.43 | 22.53 | 21.61 | 19.00 |

* Link: <https://www.indexmundi.com/g/g.aspx?c=bg&v=25>
* Newton’s Forward Difference Table:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| X | Y | Dely0 | Delsqry0 | Delcubey0 |
| 2010 | 23.43 | -0.90 | -0.02 | -1.67 |
| 2012 | 22.53 | -0.92 | -1.69 |  |
| 2014 | 21.61 | -2.61 |  |  |
| 2016 | 19.00 |  |  |  |

U= (x-x1)/h u=(x-x0)/h

= (2013-2012)/2 =(2013-2010)/2

=0.5 =1.5

Calculation:

F (2013) = y (1) + u\*Dely1+ ((u\*(u-1)/2)\*Delsqry1)

=22.53 +0.5\*(-0.92) + ((0.5\*(0.5-1))/2)\*(-1.69)

= 22.53 + (-0.46) + 0.21125

­­­ =22.28125

F(2013)= y(0)+u\*Dely0+((u\*(u-1)/2)\*Delsqry0)+((u\*(u-1)\*

(u-2)/6)\*Delcubey0)

=23.43+1.5(-0.90)+ ((1.5\*(1.5-1))/2)\*(-0.02)+((1.5\*(1.5-1)\*

(1.5-2)/6)\*(-1.67);

=23.43 - 1.35 - 0.0075+0.104

=22.1769

**Error in the Interpolation:**

**En(x) = (x - x0)(x - x1) . . .(x - xn)  f(n+1)() / (n+1)!                 x0 <  < xn**

So for the Newton's method where the nodel points  **xi,  i = 0, 1, . . . n**  are equally spaced, the error is   **En(x) = (x - x0)(x - x0- h) . . .(x - x0- nh)  f(n+1)() / (n+1)!**

|  |  |  |
| --- | --- | --- |
| = | **r(r-1). . .(r-n)** | **h(n+1)f(n+1)()** |
| **(n+1)!** |

**Advantages:**

* 1. Stirling's formula decrease much more rapidly than other difference formulae hence considering first few number of terms itself will give better accuracy.
  2. Forward or backward difference formulae use the oneside information of the function where as Stirling's formula uses the function values on both sides of f(x).

Conclusion:

The approach of interpolation, described here, can be suitably applied in inverse interpolation also.

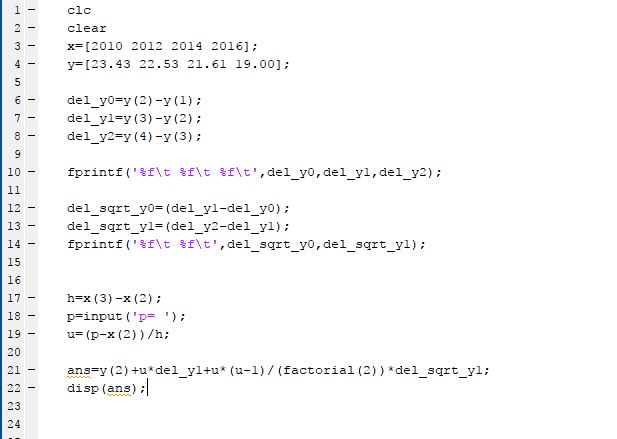
Newton’s forward interpolation formula is valid for estimating the value of the dependent variable under the following two conditions:

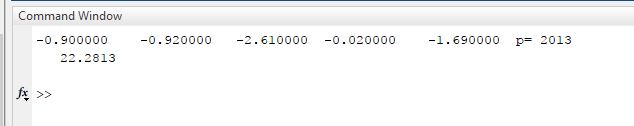
1. The given values of the independent variable are at equal interval.

2. The value of the independent variable corresponding to which the value of the dependent variable is to be estimated lies in the first half of the series of the given values of the independent variable.

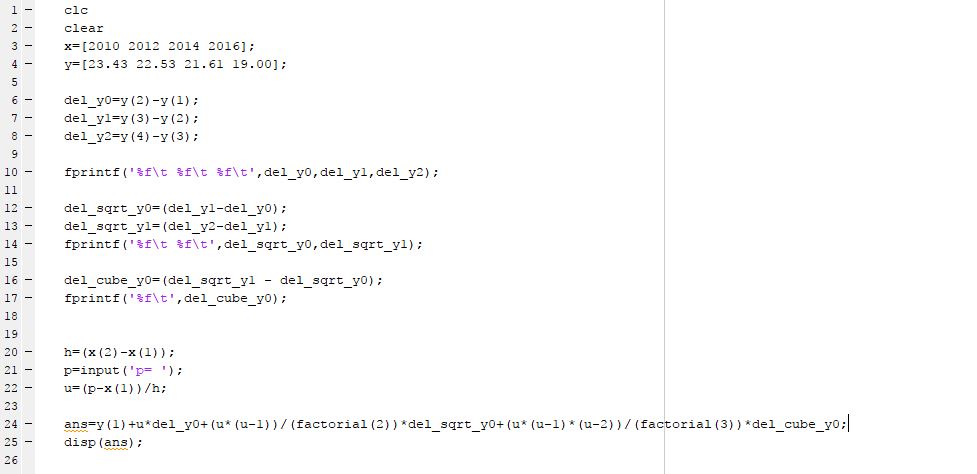
Therefore, the formula derived here is valid for representing a set of numerical data on a pair of variables by a polynomial under these two conditions only. Consequently, there is necessity of searching for some formula for representing a set of numerical data on a pair of variables by a polynomial if the value of the independent variable corresponding to which the value of the dependent variable is to be estimated lies in the last half of the series of the given values, which are at equal interval, of the independent variable. Moreover, there is also necessity of searching for some formula for representing a set of numerical data on a pair of variables by a polynomial if the given values of the independent variable are not at equal interval.

Code:



Result:

Code:



Result:



clc

clear

x=[2010 2012 2014 2016];

y=[23.43 22.53 21.61 19.00];

del\_y0=y(2)-y(1);

del\_y1=y(3)-y(2);

del\_y2=y(4)-y(3);

fprintf('%f\t %f\t %f\t',del\_y0,del\_y1,del\_y2);

del\_sqrt\_y0=(del\_y1-del\_y0);

del\_sqrt\_y1=(del\_y2-del\_y1);

fprintf('%f\t %f\t',del\_sqrt\_y0,del\_sqrt\_y1);

del\_cube\_y0=(del\_sqrt\_y1 - del\_sqrt\_y0);

fprintf('%f\t',del\_cube\_y0);

h=(x(2)-x(1));

p=input('p= ');

u=(p-x(1))/h;

ans=y(1)+u\*del\_y0+(u\*(u-1))/(factorial(2))\*del\_sqrt\_y0+(u\*(u-1)\*(u-2))/(factorial(3))\*del\_cube\_y0;

disp(ans);