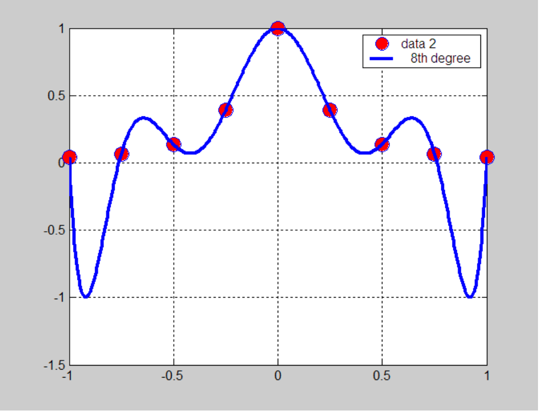
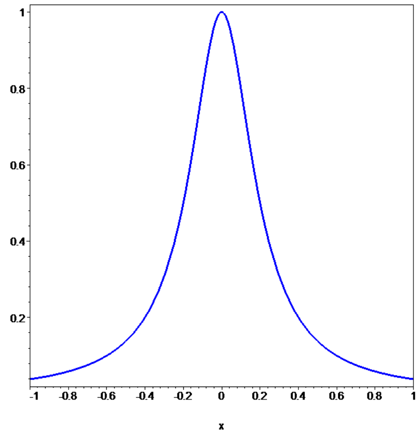
Newton Polynomials Divided Differences





There are two disadvantages to using the Lagrange Polynomial (or Neville's method) for interpolation.

- **First, they involve more arithmetic operations than does the divided difference method.**
- **Second and more importantly, if we desire to add or subtract a point from the set used to construct the polynomial, we <u>essentially to start over in the computations.</u>**
- **Both the Lagrange polynomials and Neville's method also must repeat all of the arithmetic if we must interpolate at a new x-value.**
- **The divided difference method avoids all of this computation.**

Our treatment of divided-difference tables assumes that the function, f(x), is known at several values for x:

x_0	f_0
x_1	f_1
x_2	f_2
.	.
x_n	f_n

We do not assume that the x's are evenly spaced or even that the values are arranged in any particular order (but some ordering may be advantageous).

Straight Line

Lagrange linear interpolation polynomial

$$P_1(x) = y_0 \frac{(x - x_1)}{(x_0 - x_1)} + y_1 \frac{(x - x_0)}{(x_1 - x_0)}$$
$$= y_0 L_{1,0}(x) + y_1 L_{1,1}(x)$$

$$L_{1,0}(x) = \frac{(x - x_1)}{(x_0 - x_1)}$$ **and** $$L_{1,1}(x) = \frac{(x - x_0)}{(x_1 - x_0)}.$$

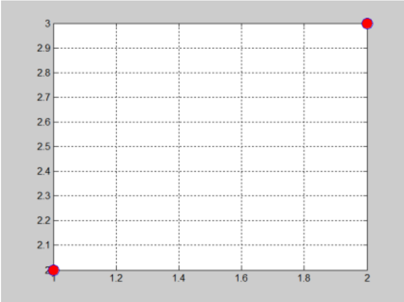
The Newton form of the equation of a straight line passing through two points (x_0, y_0) and (x_1, y_1)

$$P_1(x) = a_0 + a_1(x - x_0)$$

$$a_0 = y_0 \quad , \quad a_1 = \frac{y_1 - y_0}{x_1 - x_0}$$

Example: (Comparison)

$(x_0, y_0):(1,2)$ $(x_1, y_1):(2,3)$

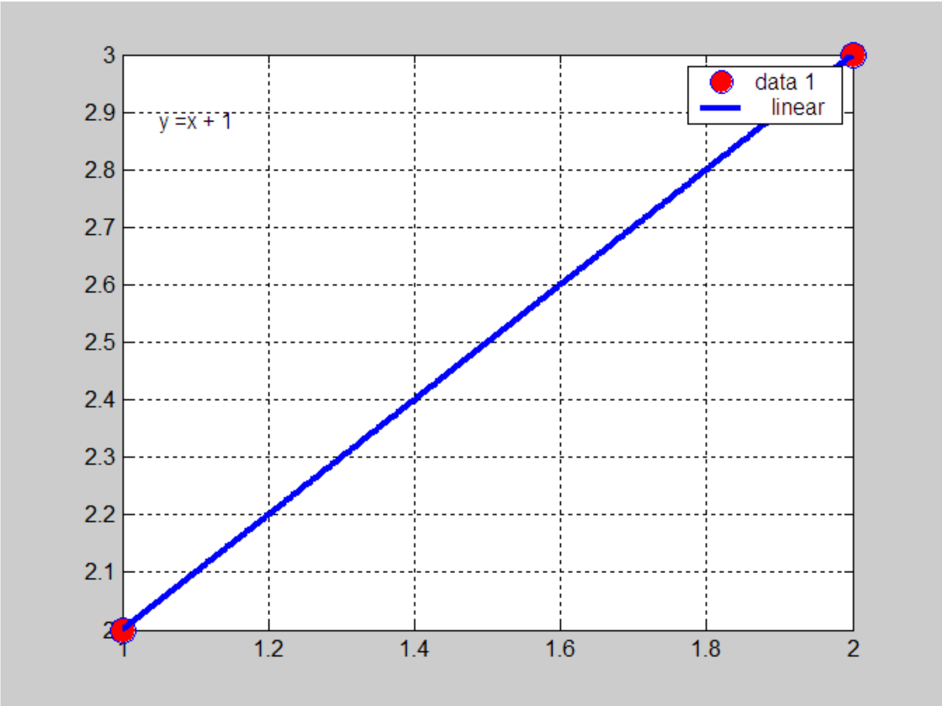


Lagrange Interpolation

$$
\begin{aligned}
P_1(x) &= y_0 \frac{(x-x_1)}{(x_0-x_1)} + y_1 \frac{(x-x_0)}{(x_1-x_0)} \\
&= 2\frac{(x-2)}{(1-2)} + 3\frac{(x-1)}{(2-1)} \\
&= -2(x-2) + 3(x-1) \\
&= -2x + 4 + 3x - 3 = x + 1
\end{aligned}
$$

Newton Interpolation

$$
\begin{aligned}
P_1(x) &= a_0 + a_1(x-x_0) \\
&= y_0 + \frac{(y_1-y_0)}{(x_1-x_0)}(x-x_0) \\
&= 2 + \frac{(3-2)}{(2-1)}(x-1) \\
&= 2 + (x-1) = x + 1
\end{aligned}
$$



3
2.9
2.8
2.7
2.6
2.5
2.4
2.3
2.2
2.1
2
1
1.2
1.4
1.6
1.8
2
y =x + 1
data 1
linear

Newton Interpolation Parabola

$$P_2(x) = a_0 + a_1(x - x_0) + a_2(x - x_0)(x - x_1)$$

$$a_0 = y_0, \quad a_1 = \frac{y_1 - y_0}{x_1 - x_0},$$

$$a_2 = \frac{\dfrac{y_2 - y_1}{x_2 - x_1} - \dfrac{y_1 - y_0}{x_1 - x_0}}{x_2 - x_0}$$

Compare with the Newton Linear Interpolation

$$P_1(x) = a_0 + a_1(x - x_0)$$

$$a_0 = y_0 \quad , \quad a_1 = \frac{y_1 - y_0}{x_1 - x_0}$$

Lagrange Interpolation Parabola

$$P_2(x) = y_0 \frac{(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)} +$$
$$y_1 \frac{(x-x_0)(x-x_2)}{(x_1-x_0)(x_1-x_2)} +$$
$$y_2 \frac{(x-x_0)(x-x_1)}{(x_2-x_0)(x_2-x_1)}$$
$$= y_0 L_{2,0}(x) + y_1 L_{2,1}(x) + y_2 L_{2,2}(x)$$

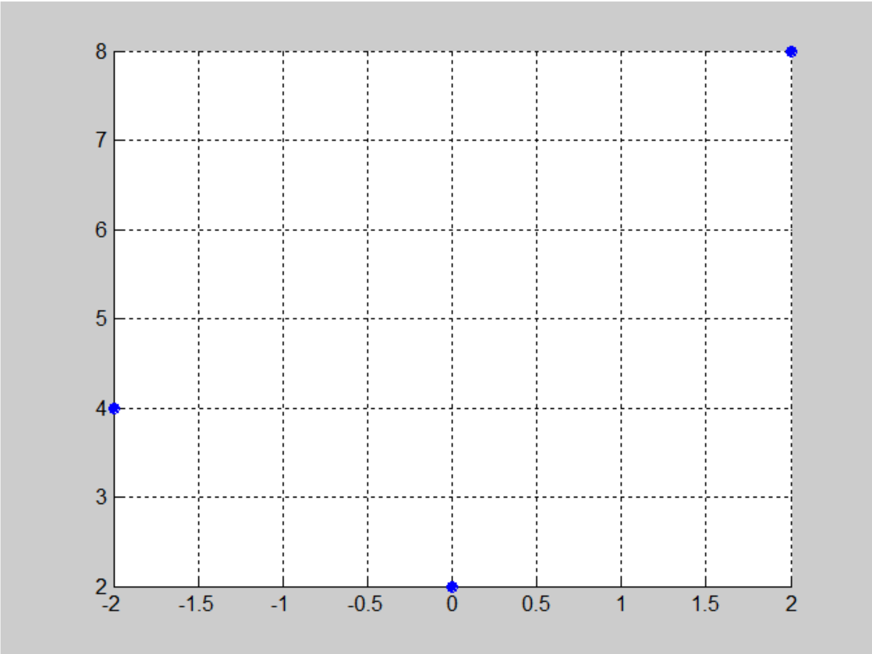
Compare with the Lagrange Linear Interpolation

$$P_1(x) = y_0 \frac{(x-x_1)}{(x_0-x_1)} + y_1 \frac{(x-x_0)}{(x_1-x_0)}$$
$$= y_0 L_{1,0}(x) + y_1 L_{1,1}(x)$$

Example: **Suppose that we have the following data pairs – x values and f(x) values- unknown function.**

x	y
-2	4
0	2
2	8

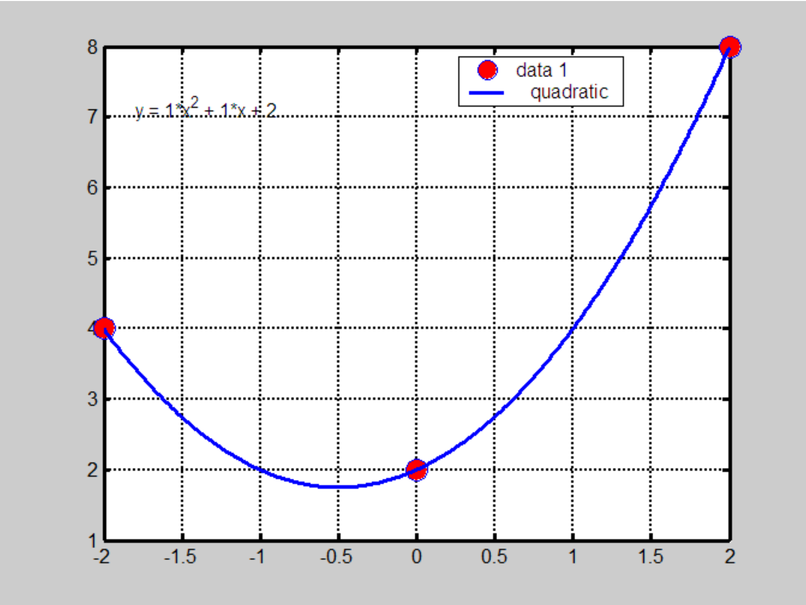
Find a quadratic polynomial using the three given points.

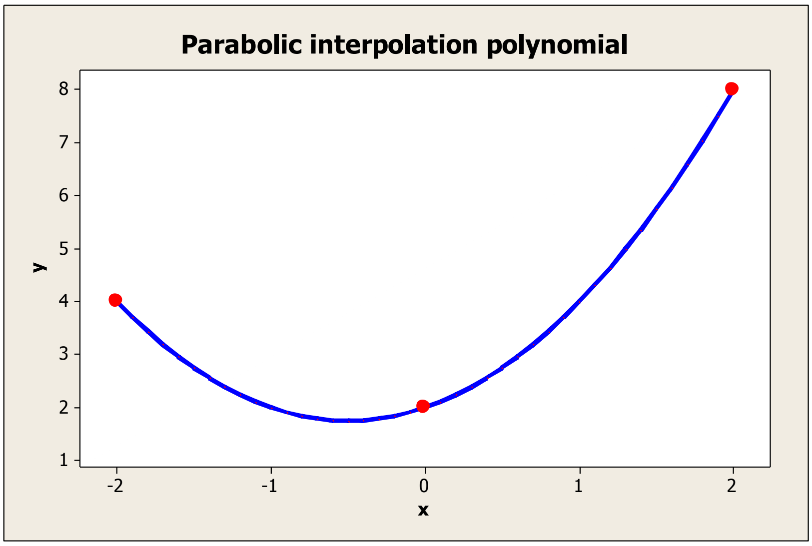


This problem solved before using Lagrange Polynomial approximation

$$p(x)=\frac{(x-0)(x-2)}{(-2-0)(-2-2)}(4)+\frac{(x-(-2))(x-2)}{(0-(-2))(0-2)}(2)+\frac{(x-(-2))(x-0)}{(2-(-2))(2-0)}(8)$$
$$=\frac{x(x-2)}{8}(4)+\frac{(x+2)(x-2)}{-4}(2)+\frac{x(x+2)}{8}(8)$$
$$=x^2+x+2$$

```
>> plot(x,y,'bo');grid on
```





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

Newton Interpolation Parabola

$$P_2(x) = a_0 + a_1(x - x_0) + a_2(x - x_0)(x - x_1)$$

$$a_0 = y_0, \quad a_1 = \frac{y_1 - y_0}{x_1 - x_0},$$

$$a_2 = \frac{\dfrac{y_2 - y_1}{x_2 - x_1} - \dfrac{y_1 - y_0}{x_1 - x_0}}{x_2 - x_0}$$

$(x_0, y_0) = (-2,4)$, $(x_1, y_1) = (0,2)$, $(x_2, y_2) = (2,8)$

$$a_0 = y_0 = 4,$$

$$a_1 = \frac{y_1 - y_0}{x_1 - x_0} = \frac{2-4}{0-(-2)} = -1,$$

$$a_2 = \frac{\dfrac{y_2 - y_1}{x_2 - x_1} - \dfrac{y_1 - y_0}{x_1 - x_0}}{x_2 - x_0} = \frac{\dfrac{8-2}{2-0} - \dfrac{2-4}{0-(-2)}}{2-(-2)} = 1$$

Thus

$$p(x) = 4 - (x+2) + x(x+2) = x^2 + x + 2$$

Check this result with the result of Lagrange Polynomial.

The calculations can be performed in a systematic manner, using a "divided-difference table".

x_i	y_i	d_i	dd_i
-2	**4**		
		$\frac{y_1-y_0}{x_1-x_0}=\frac{2-4}{0-(-2)}=$ **-1**	
0	**2**		$\frac{d_2-d_1}{x_2-x_0}=\frac{3--(1)}{2-(-2)}=$ **1**
		$\frac{y_2-y_1}{x_2-x_1}=\frac{8-2}{2-0}=3,$	
2	**8**		

The coefficients of the Newton polynomial are the top entries in this table.

$$P_2(x)=a_0+a_1(x-x_0)+a_2(x-x_0)(x-x_1)$$

$$\begin{aligned} p(x) &= 4+(-1)(x+2)+(1)x(x+2) \\ &= x^2+x+2 \end{aligned}$$

Additional Data Points

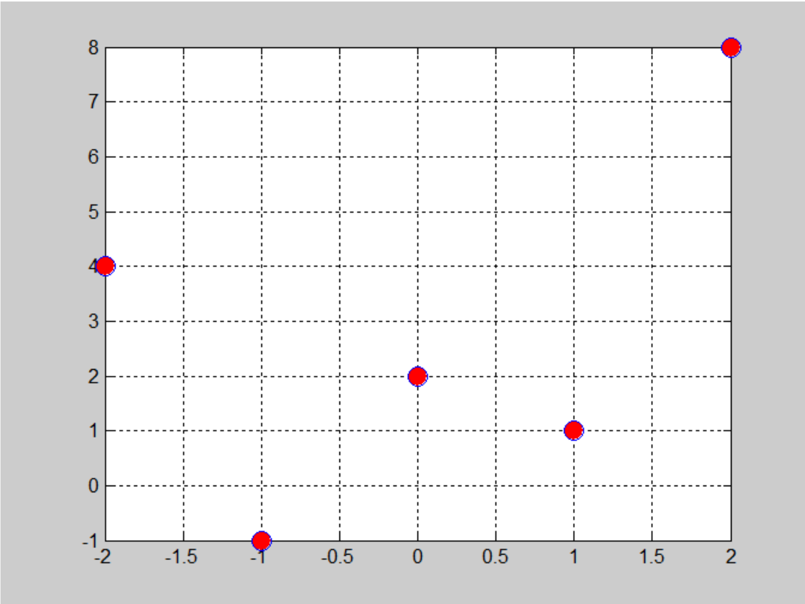
Extend the previous example adding two new points.

Old table

x	y
-2	**4**
0	**2**
2	**8**

New Table

x	y
-2	**4**
0	**2**
2	**8**
-1	**-1**
1	**1**



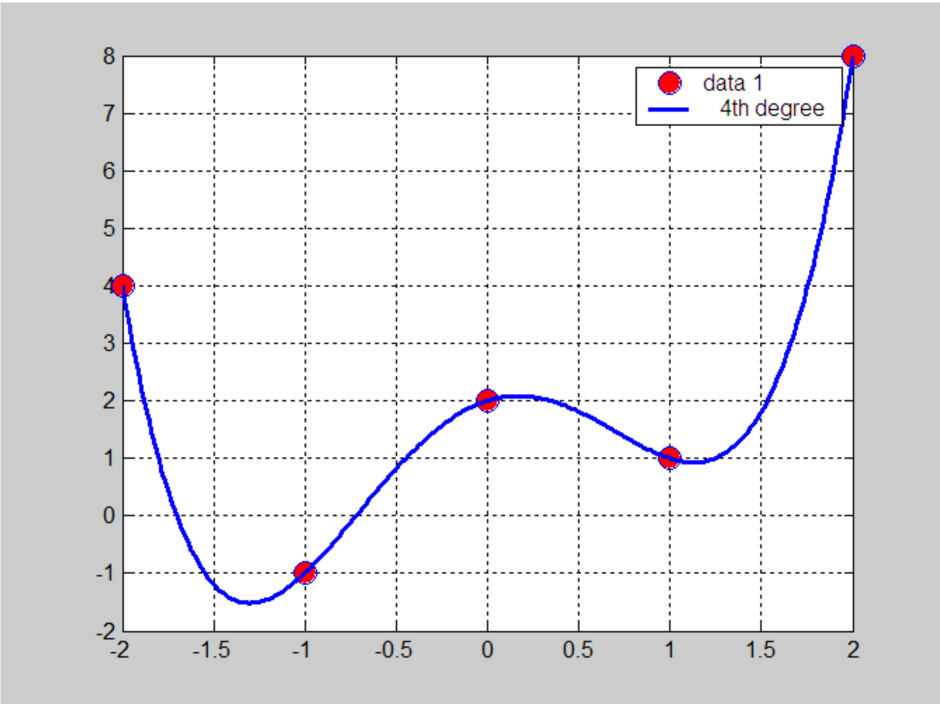
The divided-difference table becomes

x_i	y_i	d_i	dd_i	ddd_i	$dddd_i$
-2	**4**				
		$\frac{(2-4)}{0-(-2)}=$ **-1**			
0	2		$\frac{(3+1)}{(2+2)}=$ **1**		
		$\frac{(8-2)}{(2-0)}=3$		$\frac{(0-1)}{(-1+2)}=$ **-1**	
2	8		$\frac{(3-3)}{(-1-0)}=0$		$\frac{(2+1)}{(1+2)}=$ **1**
		$\frac{(-1-8)}{(-1-2)}=3$		$\frac{(2-0)}{(1-0)}=2$	
-1	**-1**		$\frac{(1-3)}{(1-2)}=2$		
		$\frac{(1+1)}{(1+1)}=1$			
1	**1**				

The Newton interpolation polynomial

$$P_4(x) = a_0 + a_1(x - x_0) + a_2(x - x_0)(x - x_1) +$$
$$a_3(x - x_0)(x - x_1)(x - x_2) + a_4(x - x_0)(x - x_1)(x - x_2)(x - x_3)$$

$$P_4(x) = 4 + (-1)(x + 2) + (1)(x + 2)(x) + (-1)(x + 2)(x)(x - 2) +$$
$$(1)(x + 2)(x)(x - 2)(x + 1).$$



Quartic interpolation polynomial

$$P_4(x) = 4 + (-1)(x+2) + (1)(x+2)(x) + (-1)(x+2)(x)(x-2) +$$
$$(1)(x+2)(x)(x-2)(x+1).$$
$$= x^4 - 3x^2 + x + 2.$$

```
x =  -2     0     2    -1     1

y =   4     2     8    -1     1
>> P=polyfit(x,y,4)
P = 1.0000    0.0000   -3.0000    1.0000    2.0000
```

General Form

Consider the nth-degree polynomial written in a special way:

$$P_n(x) = a_0 + (x - x_0)a_1 + (x - x_0)(x - x_1)a_2 + \ldots$$
$$+ (x - x_0)(x - x_1)\ldots(x - x_{n-1})a_n.$$

If we choose the a_i so that $P_n(x) = f(x)$ at the <u>n +1</u> known points, $(x_i, f_i), i = 0 \ldots, n$ then $P_n(x)$ is an interpolating polynomial. We will show that the a_i's are readily determined by using what are called the *<u>divided difference of the tabulated values.</u>*

Standard notation for divided difference

A special standard notation for divided differences is

$$f[x_0, x_1] = \frac{f_1 - f_0}{x_1 - x_0},$$

called the *first divided difference between x_0 and x_1*.

The function

$$f[x_1, x_2] = \frac{f_2 - f_1}{x_2 - x_1},$$

is the *first divided difference between x_1 and x_2.*

(We use $f[x_0] = f_0 = f(x_0)$ **)**

In general,

$$f[x_s, x_t] = \frac{f_t - f_s}{x_t - x_s},$$

is the first divided difference between *x_s and x_r.*

Observe that

$$f[x_s, x_t] = \frac{f_t - f_s}{x_t - x_s} = \frac{f_s - f_t}{x_s - x_t} = f[x_t, x_s]$$

Second Order Differences

Second and higher order differences are defined in terms of lower-order differences.

Second order:

$$f[x_0, x_1, x_2] = \frac{f[x_1, x_2] - f[x_0, x_1]}{x_2 - x_0}$$

n^th^- order:

$$f[x_0, x_1, \ldots, x_n] = \frac{f[x_1, x_2, \ldots, x_n] - f[x_0, x_1, \ldots, x_{n-1}]}{x_n - x_0}$$

Zero order difference:

$$f[x_s] = f_s$$

$x_0 \quad f[x_0]$

$$f[x_0,x_1]=\frac{f[x_1]-f[x_0]}{x_1-x_0}$$

$x_1 \quad f[x_1]$ *

$$f[x_1,x_2]=\frac{f[x_2]-f[x_1]}{x_2-x_1}$$

$x_2 \quad f[x_2]$

* $$f[x_0,x_1,x_2]=\frac{f[x_1,x_2]-f[x_0,x_1]}{x_2-x_0}$$

With this notation the polynomial

$$P_n(x) = a_0 + (x - x_0)a_1 + (x - x_0)(x - x_1)a_2 + \ldots$$
$$+ (x - x_0)(x - x_1)\ldots(x - x_{n-1})a_n.$$

<u>can be re-expressed</u>

$$P_n(x) = f[x_0] + (x - x_0) f[x_0, x_1] + (x - x_0)(x - x_1) f[x_0, x_1, x_2] + \ldots$$
$$+ (x - x_0)(x - x_1)\ldots(x - x_{n-1}) f[x_0, x_1, \ldots, x_n].$$

$$\begin{aligned}
a_0 &= f[x_0], \\
a_1 &= f[x_0, x_1], \\
a_2 &= f[x_0, x_1, x_2], \\
&. \\
&. \\
&. \\
a_n &= f[x_0, x_1, \ldots, x_n].
\end{aligned}$$

$$P_n(x) = f[x_0] + \sum_{k=1}^{n} f[x_0, x_1, \ldots, x_k](x - x_0)\ldots(x - x_{k-1}).$$

This is known as Newton's interpolatory divided-difference Formula.

Example: **Suppose that we have the following data pairs – x values and f(x) values- where f(x) is <u>some unknown function.</u>**

i	x	f(x)
0	3.2	22.0
1	2.7	17.8
2	1.0	14.2
3	4.8	38.3
4	5.6	51.7

Divided Differences

x_i	f_i	$f[x_i,x_{i+1}]$	$f[x_i,x_{i+1},x_{i+2}]$	$f[x_i,....,x_{i+3}]$	$f[x_i,....,x_{i+4}]$
x_0	f_0	$f[x_0,x_1]$	$f[x_0,x_1,x_2]$	$f[x_0,x_1,x_2,x_3]$	$f[x_0,x_1,x_2,x_3,x_4]$
x_1	f_1	$f[x_1,x_2]$	$f[x_1,x_2,x_3]$	$f[x_1,x_2,x_3,x_4]$	
x_2	f_2	$f[x_2,x_3]$	$f[x_2,x_3,x_4]$		
x_3	f_3	$f[x_3,x_4]$			
x_4	f_4				

Divided Difference Table

x_i	f_i	$f[x_i,x_{i+1}]$	$f[x_i,x_{i+1},x_{i+2}]$	$f[x_i,....,x_{i+3}]$	$f[x_i,....,x_{i+4}]$
3.2	**22.0**	**8.400**	**2.856**	**-0.528**	**0.256**
2.7	**17.8**	**2.118**	**2.012**	**0.0865**	
1.0	**14.2**	**6.342**	**2.263**		
4.8	**38.3**	**16.750**			
5.6	**51.70**				

$$f[x_0,x_1]=\frac{f_1-f_0}{x_1-x_0}=\frac{17.8-22.0}{2.7-3.2}=\frac{-4.2}{-0.5}=8.4$$

$$f[x_0,x_1,x_2]=\frac{f[x_1,x_2]-f[x_0,x_1]}{x_2-x_0}=\frac{2.118-8.40}{1.0-3.2}=2.856$$

$$f[x_0,x_1,x_2,x_3]=\frac{f[x_1,x_2,x_3]-f[x_0,x_1,x_2]}{x_3-x_0}=\frac{2.012-2.856}{4.8-3.2}=-0.5275$$

$$f[x_0,\ldots,x_4]=\frac{f[x_1,\ldots,x_4]-f[x_0,\ldots,x_3]}{x_4-x_0}=\frac{0.0865-(-0.528)}{5.6-3.2}=0.2560$$

Interpolating polynomial of <u>degree-3</u> that fits the data at all points from x_0=3.2 to x_3=4.8.

$$\begin{aligned}P_3(x)&=22.0+8.4(x-3.2)+2.856(x-3.2)(x-2.7)\\&\quad-0.528(x-3.2)(x-2.7)(x-1.0)\end{aligned}$$

What is the fourth-degree polynomial that fits at all five points?

Interpolating polynomial of <u>degree-4</u> that fits the data at all points from x_0=3.2 to x_4=5.6.

$$\begin{aligned}P_4(x)&=22.0+8.4(x-3.2)+2.856(x-3.2)(x-2.7)\\&\quad-0.528(x-3.2)(x-2.7)(x-1.0)\\&\quad+0.256(x-3.2)(x-2.7)(x-1.0)(x-4.8)\end{aligned}$$

We only have to add one more term to P_3(x)

$$P_4(x)=P_3(x)+0.256(x-3.2)(x-2.7)(x-1.0)(x-4.8)$$

Find the interpolated value for x=3.0 using P_3

$$P_3(x) = 22.0 + 8.4(x-3.2) + 2.856(x-3.2)(x-2.7) \\ -0.528(x-3.2)(x-2.7)(x-1.0)$$

$P_3(3) = 20.2120$

Find the interpolated value for x=3.0 using P_4

$$P_4(x) = P_3(x) + 0.256(x-3.2)(x-2.7)(x-1.0)(x-4.8)$$

$P_4(3) = 20.2120+0.256(-0.2)(0.3)(2.0)(-1.8)$
$= 20.267296$

Compare Lagrange Interpolation Polynomial

```
>> x=[3.2 2.7 1.0 4.8 5.6]
x =    3.2000    2.7000    1.0000    4.8000    5.6000

>> y=[22.0 17.8 14.2 38.3 51.70]
y =   22.0000   17.8000   14.2000   38.3000   51.7000

>> P=polyfit(x,y,4)
P =    0.2558   -3.5208   18.6885  -36.1836   34.9600

>> xval=polyval(P,3.0)
xval =

   20.2672
```

Now Newton Polynomial solution was

$$
\begin{aligned}
P_4(x) = 22.0 + 8.4(x-3.2) + 2.856(x-3.2)(x-2.7) \\
-0.528(x-3.2)(x-2.7)(x-1.0) \\
+0.256(x-3.2)(x-2.7)(x-1.0)(x-4.8)
\end{aligned}
$$

Simplify this equation in Maple

```
> simplify(22+8.4*(x-3.2)+2.856*(x-3.2)*(x-2.7)-0.528*(x-3.2)*(x-2.7)*(x-1.0)+0.256*(x-3.2)*(x-2.7)*(x-1.0)*(x-4.8));
```

$$34.97459200 - 36.20611200\, x + 18.70016000\, x^2 - 3.523200000\, x^3 + 0.2560000000\, x^4$$

An Algorithm for Interpolation from a Divided Difference Table (Newton's Divided Difference Algorithm)

Given a set of n+1 distinct numbers $x_0,x_1,\ldots,x_n$ at the number x for the function f and a value x=u at which he interpolating polynomial is to be evaluated:

We fist find the coefficients of the interpolating polynomial. These are stored in vector dd.

```
        For i=0 to n Step 1 Do
           Set dd[i]=f[i]
        End for i.
              for j=1.. to n Step 1 Do
              set temp1=dd[j-1].
                  For k=j to n Step 1 DO
                   Set temp2=dd[k]
                   Set dd[k]=(dd[k]-temp1)/x[k]-x[k-j]).
                   Temp1=temp2
                  End for k
            End for j
```

Now compute the value of the polynomial at u. We do this by nested multiplication from the highest term.

```
             Set sum = 0
    For i=n Down to  1 Step 1 Do
            Se t sum=(sum+dd[j])(u-(x[I-1])
           Set   sum=sum +dd[0]
   End for I
  ddvalue=sum
```

ddvalue is the value of the polynomial at u, $P_n(u)$.

MATLAB M-File (Divided Differences Newton Interpolation Polynomial)

```
function [C,D]=newpoly(X,Y)
%Input     - X is a vector that contains a list of abscissas
%          - Y is a vector that contains a list of ordinates
%Output  - C is a vector that contains the coefficients
%            of the Newton intepolatory polynomial
%          - D is the divided difference table
n=length(X);
D=zeros(n,n);
D(:,1)=Y';
%Use formula (20) to form the divided difference table
for j=2:n
  for k=j:n
    D(k,j)=(D(k,j-1)-D(k-1,j-1))/(X(k)-X(k-j+1));
  end
end
%Determine the coefficients of the Newton interpolatory
polynomial
C=D(n,n);
for k=(n-1):-1:1
  C=conv(C,poly(X(k)));
  m=length(C);
  C(m)=C(m)+D(k,k);
End
```

Example:

Suppose that we have the following data pairs – x values and f(x) values- where f(x) is some unknown function.

i	x	f(x)
0	3.2	22.0
1	2.7	17.8
2	1.0	14.2
3	4.8	38.3
4	5.6	51.7

```
>> X=[3.2 2.7 1.0 4.8 5.6]
X =    3.2000    2.7000    1.0000    4.8000    5.6000
>> Y=[22.0 17.8 14.2 38.3 51.7]
Y =   22.0000   17.8000   14.2000   38.3000   51.7000
>> [C,D]=newpoly(X,Y)
C =
    0.2558   -3.5208   18.6885  -36.1836   34.9600
D =
   22.0000         0         0         0         0
   17.8000    8.4000         0         0         0
   14.2000    2.1176    2.8556         0         0
   38.3000    6.3421    2.0116   -0.5275         0
   51.7000   16.7500    2.2626    0.0865    0.2558
>>
```

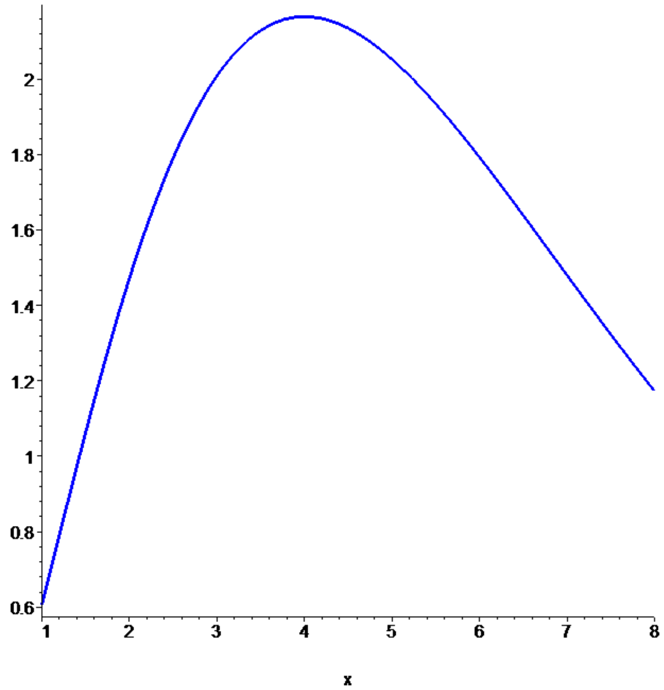
x_i	f_i	$f[x_i, x_{i+1}]$	$f[x_i, x_{i+1}, x_{i+2}]$	$f[x_i, \ldots, x_{i+3}]$	$f[x_i, \ldots, x_{i+4}]$
3.2	**22.0**	**8.400**	**2.856**	**-0.528**	**0.256**
2.7	**17.8**	**2.118**	**2.012**	**0.0865**	
1.0	**14.2**	**6.342**	**2.263**		
4.8	**38.3**	**16.750**			
5.6	**51.70**				

22.0000	**0**	**0**	**0**	**0**
17.8000	**8.4000**	**0**	**0**	**0**
14.2000	**2.1176**	**2.8556**	**0**	**0**
38.3000	**6.3421**	**2.0116**	**-0.5275**	**0**
51.7000	**16.7500**	**2.2626**	**0.0865**	**0.2558**

Example:

Suppose that we have the following data pairs – x values and f(x) values- where f(x) is $f(x) = x^2 e^{-x/2}$.

i	x	f(x)
0	**1.1**	**0.6981**
1	**2.0**	**1.4715**
2	**3.5**	**2.1287**
3	**5.0**	**2.0521**
4	**7.10**	**1.4480**

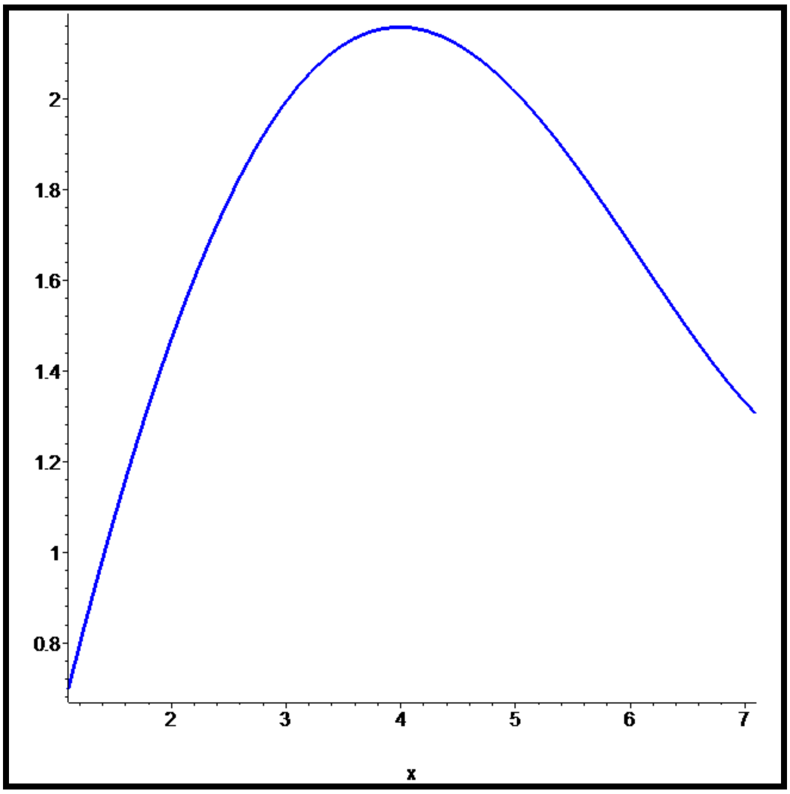


$f(x) = x^2 e^{-x/2}$.

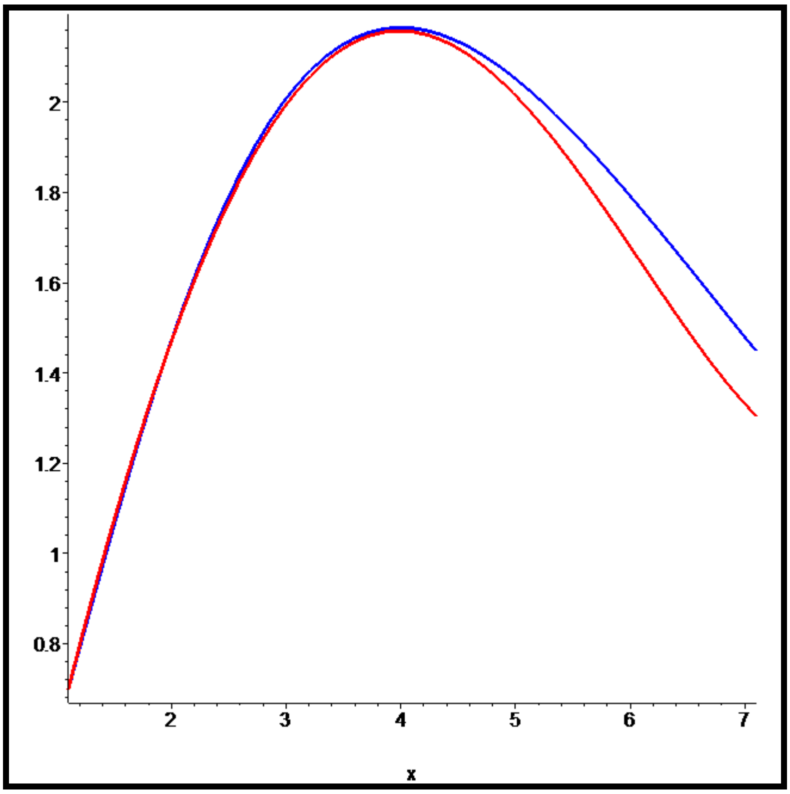
x_i	f_i	$f[x_i,x_{i+1}]$	$f[x_i,x_{i+1},x_{i+2}]$	$f[x_i,....,x_{i+3}]$	$f[x_i,....,x_{i+4}]$
1.1	**0.6981**	**0.8593**	**-0.1755**	**0.0032**	**0.0027**
2.0	**1.4715**	**0.4381**	**-0.1631**	**0.0191**	
3.5	**2.1287**	**-0.0511**	**-0.0657**		
5.0	**2.0521**	**-0.2877**			
7.1	**1.4480**				

```
>> X=[1.10 2.0 3.5 5.0 7.1]
X =    1.1000    2.0000    3.5000    5.0000    7.1000
>> Y=[0.6981 1.4715 2.1287 2.0521 1.4480]
Y =    0.6981    1.4715    2.1287    2.0521    1.4480
>> [C,D]=newpoly(X,Y)
C =    0.0026   -0.0276   -0.0745    1.2517   -0.5558
D =
    0.6981         0         0         0         0
    1.4715    0.8593         0         0         0
    2.1287    0.4381   -0.1755         0         0
    2.0521   -0.0511   -0.1631    0.0032         0
    1.4480   -0.2877   -0.0657    0.0191    0.0026
>>
```

```
> plot([0.0026*x^4-0.0276*x^3-0.0745*x^2+1.2517*x-
0.5558],x=1.1..7.1,color=[blue]);
```




```
> plot([0.0026*x^4-0.0276*x^3-0.0745*x^2+1.2517*x-
0.5558,(x^2)*exp(-x/2)],x=1.1..7.1,color=[red ,blue]);
```



Find the error of interpolates for f(1.75) using polynomial of degrees 1,2, and 3.

Degree	Interpolated Value	Actual Value	Actual Error
1.	**1.25668**	**1.276639995**	**0.01966**
2.	**1.28520**	**1.276639995**	**-0.00856**
3.	**1.28611**	**1.276639995**	**-0.00947**

Divided Differences Evenly Spaced Data (Equal Spacing)

Newton's interpolatory divided-difference Formula

$$P_n(x) = f[x_0] + (x - x_0) f[x_0, x_1] + (x - x_0)(x - x_1) f[x_0, x_1, x_2] + \ldots$$
$$+ (x - x_0)(x - x_1)\ldots(x - x_{n-1}) f[x_0, x_1, \ldots, x_n].$$

$$P_n(x) = f[x_0] + \sum_{k=1}^{n} f[x_0, x_1, \ldots, x_k](x - x_0)\ldots(x - x_{k-1})$$

Introducing the notation

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

And

$$x = x_0 + sh \quad \text{or} \quad s = (x - x_0)/h$$

The difference $x - x_i$ **can be written as**

$$x - x_i = (s - i)h;$$

So the polynomial equation

$$P_n(x) = f[x_0] + (x - x_0) f[x_0, x_1] + (x - x_0)(x - x_1) f[x_0, x_1, x_2] + \ldots \\ + (x - x_0)(x - x_1)\ldots(x - x_{n-1}) f[x_0, x_1, \ldots, x_n].$$

becomes

$$P_n(x) = P_n(x_0 + sh) = f[x_0] + shf[x_0, x_1] + s(s-1)h^2 f[x_0, x_1, x_2] \\ + s(s-1)\ldots(s-n+1)h^n f[x_0, x_1, \ldots, x_n].$$

In general,

$$P_n(x) = \sum_{k=0}^{n} s(s-1)(s-k+1)h^k f[x_0, x_1, \ldots, x_k]$$

This formula is called the Newton forward divided-difference formula.

Using binomial-coefficient notation

$$\binom{s}{k} = \frac{s(s-1)\ldots(s-k+1)}{k!}$$

We can express $P_n(x)$ **compactly as**

$$P_n(x) = P_n(x_0 + sh) = \sum_{k=0}^{n} \binom{s}{k} k! h^k f[x_0, x_1, \ldots, x_k].$$

Example:

The following Table lists values of a function (the Bessel function of the first kind) at various points.

x	f(x)
1.0	0.7651977
1.3	0.6200860
1.6	0.4554022
1.9	0.2818186
2.2	0.1103623

We first obtain divided difference table.

x	f(x)	First divided differences	Second divided differences	Third divided differences	Fourth divided differences
1.0	**0.7651977**				
		-0.4837057			
1.3	0.6200860		**-0.1087339**		
		-0.5489460		**0.0658784**	
1.6	0.4554022		-0.0494433		**0.0018251**
		-0.5786120		0.0680685	
1.9	0.2818186		0.0118183		
		-0.5715210			
2.2	0.1103623				

The coefficients of the Newton forward divided-difference form of the interpolation polynomial are along the diagonal in the table.

Non-Equal Spacing Approach:

$$P_n(x) = f[x_0] + (x-x_0) f[x_0,x_1] + (x-x_0)(x-x_1) f[x_0,x_1,x_2] + \ldots$$
$$+ (x-x_0)(x-x_1)\ldots(x-x_{n-1}) f[x_0,x_1,\ldots,x_n].$$

$P_4(x)$= 0.7651977 - 0.4837057(x-1.0)-0.1087339(x-1.0)(x-1.3)

+ 0.0658784(x-1.0)(x-1.3)(x-1.6)

+ 0.0018251(x-1.0)(x-1.3)(x-1.6)(x-1.9)

P_4 (1.1) =?

P_4(1.1)= 0.7651977 - 0.4837057(1.1-1.0)

- 0.1087339(1.1-1.0) (1.1-1.3)

+ 0.0658784(1.1-1.0) (1.1-1.3) (1.1-1.6)

+ 0.0018251(1.1-1.0) (1.1-1.3) (1.1-1.6) (1.1-1.9)

= 0.7196480

```
X =    1.0000    1.3000    1.6000    1.9000    2.2000
Y =    0.7652    0.6201    0.4554    0.2818    0.1104
>> [C,D]=newpoly(X,Y)
C =
    0.0018    0.0553   -0.3430    0.0734    0.9777
D =
    0.7652         0         0         0         0
    0.6201   -0.4837         0         0         0
    0.4554   -0.5489   -0.1087         0         0
    0.2818   -0.5786   -0.0494    0.0659         0
    0.1104   -0.5715    0.0118    0.0681    0.0018
```

OR

```
>> P=polyfit(X,Y,4)
P =
    0.0018    0.0553   -0.3430    0.0734    0.9777
>> xval=polyval(P,1.1)
xval =
    0.7196
```

But we know the data is equally spacing.

So we can use

$$P_n(x) = P_n(x_0 + sh) = f[x_0] + shf[x_0, x_1] + s(s-1)h^2 f[x_0, x_1, x_2] \\ + s(s-1)\ldots(s-n+1)h^n[x_0, x_1, \ldots, x_n].$$

If an approximation to f(1.1) is required, the reasonable choice for $x_0, x_1, \ldots, x_n$ would be

$x_0 = 1.0$, $x_1 = 1.3$, $x_2 = 1.6$, $x_3 = 1.9$, $x_4 = 2.2$,

$$x = x_0 + sh \quad \text{or} \quad s = (x - x_0)/h$$

and

h=0.3,

s = (1.1-1.0)/0.3=0.1/0.3=1/3

$P_4(1.1)$ =?

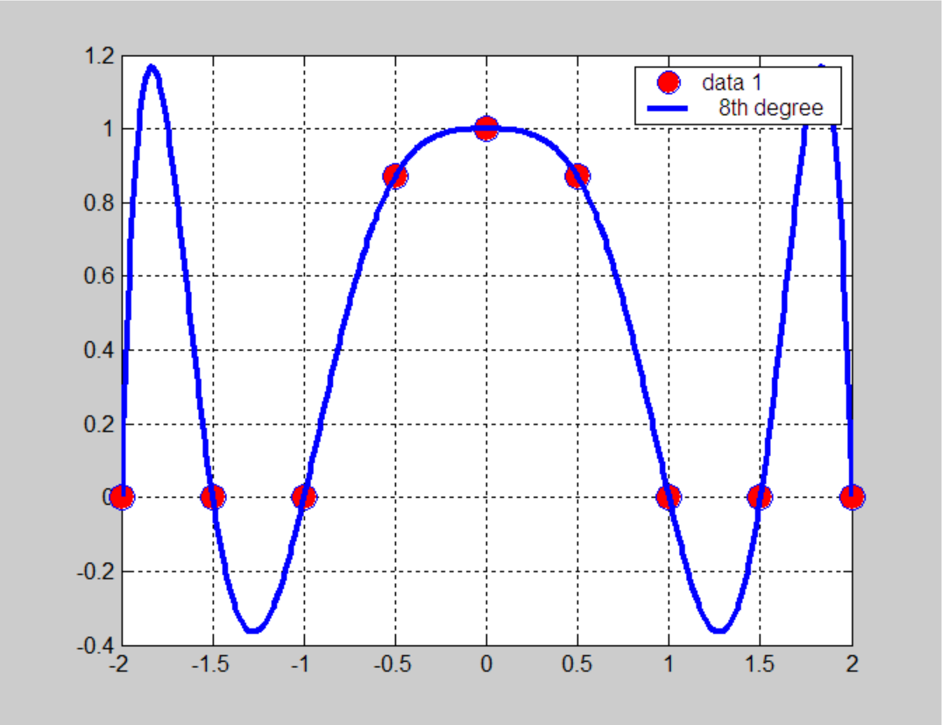
$$\begin{aligned} P_4(1.1) = P_4(1.0 + 1/3(0.3)) &= 0.7651977 \\ &+ (1/3)(0.3)(-0.4837057) \\ &+ (1/3)(-2/3)(0.3)^2(-0.1087339) \\ &+ (1/3)(-2/3)(-5/3)(0.3)^3(0.0658784) \\ &+ (1/3)(-2/3)(-5/3)(-8/3)(0.3)^4(0.0018251) \\ &= 0.7196480 \end{aligned}$$

Difficulties with Polynomial Interpolation

There are many types of problems in which polynomial interpolation through a <u>moderate number of data points works very poorly</u>.

Example:

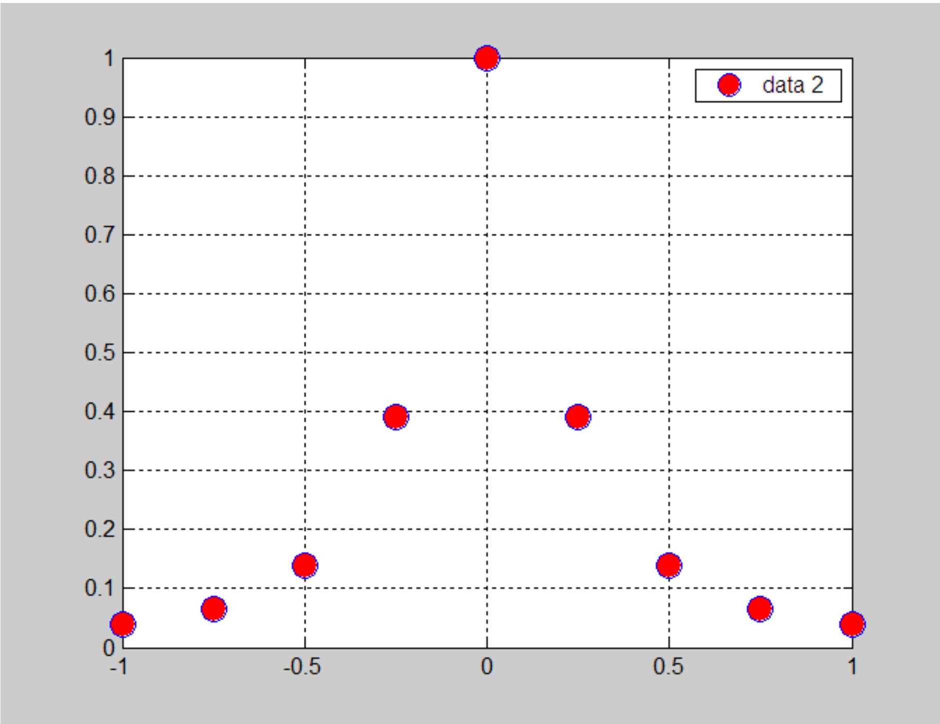
i	x	f(x)
0	-2	0
1	-1.5	0
2	-1	0
3	-0.5	0.87
4	0	1
5	0.5	0.87
6	1	0
7	1.5	0
8	2	0

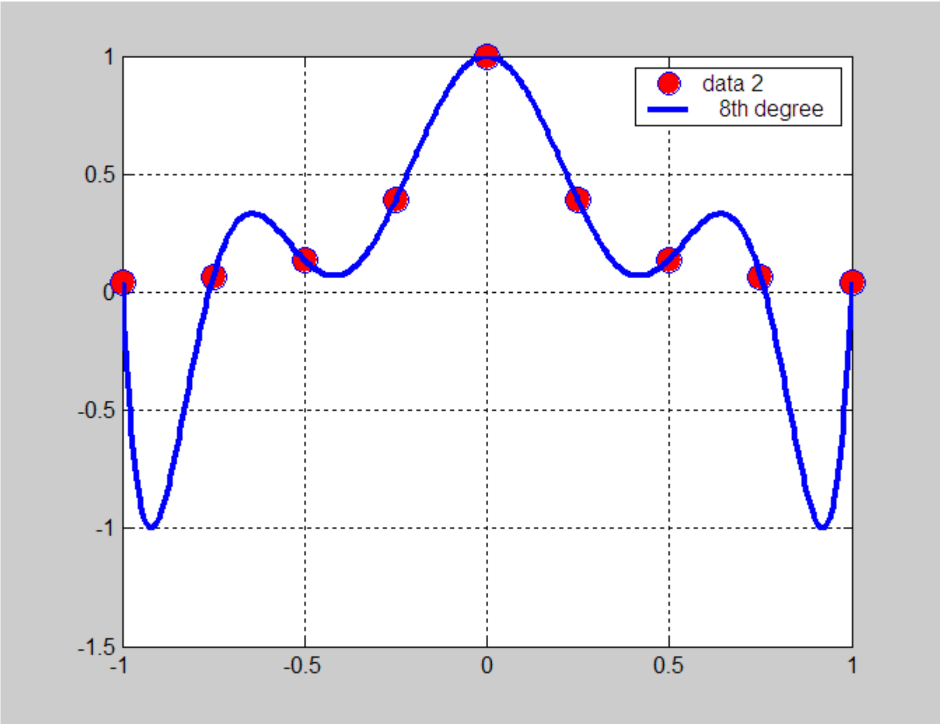


?

Example:

i	x	f(x)
0	-1	0.0385
1	-0.75	0.0664
2	-0.5	0.138
3	-0.25	0.3902
4	0.00	1.00
5	0.25	0.3902
6	0.50	0.138
7	0.75	0.0664
8	1.00	0.0385





The true Function is *Runge Function*

$$f(x) = \frac{1}{1+25x^2}$$