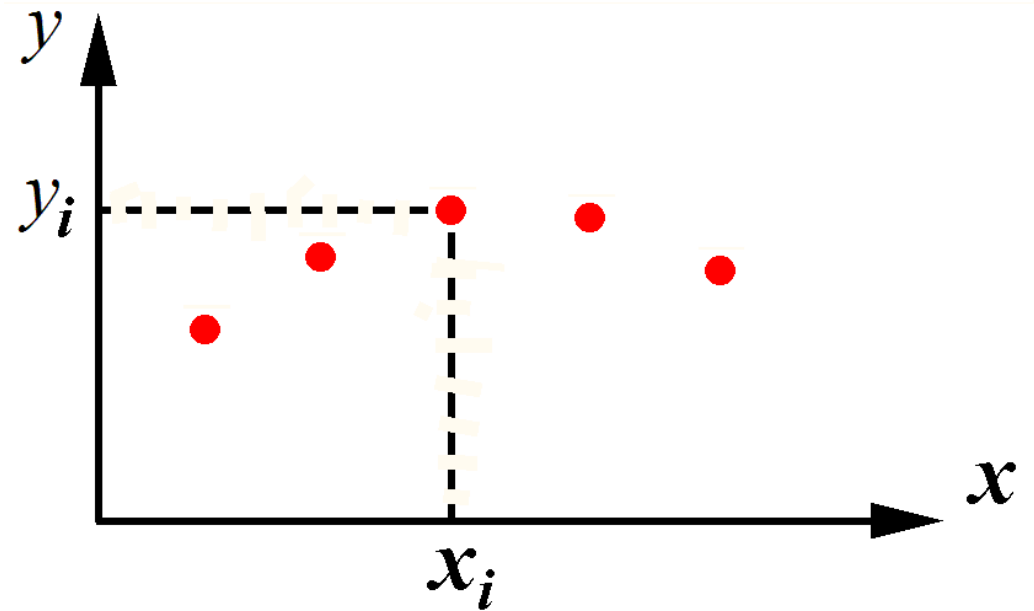


Chapter 5. Interpolation by polynomials

Suppose that values of a function $f(x)$ are given at finite number of nodes $x_0, x_1, x_2, \dots, x_n$ (the function is given by table/array).

What value can be adopted as approximate value of f at a point x lying between nodes?

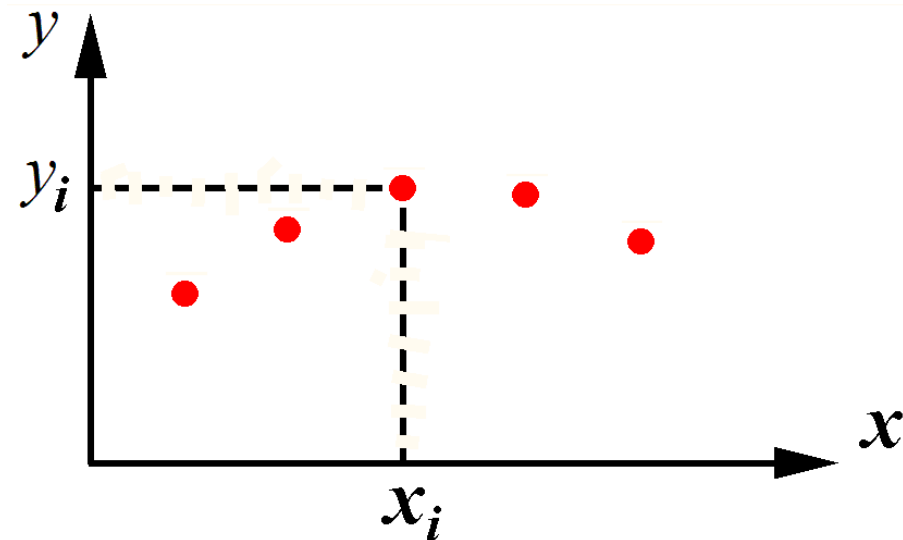


$x_0, x_1, x_2, \dots, x_n$

$y_0, y_1, y_2, \dots, y_n$

If $x_i < x < x_{i+1}$, then $f(x) = ?$ approximately

An evident approach is to plot a linear function or a polynomial $P(x)$ passing through some of the given points (x_i, y_i) and assume that $f(x) \approx P(x)$.



(We already used linear function $P_1(x)$ to obtain trapezoids formula and parabola $P_2(x)$ to obtain Simpson's formula).

Is it possible to plot a single polynomial that passes through **all** $n+1$ points (x_i, y_i) in the plane (x, y) ?

A general form of n^{th} degree polynomial is:

$$P_n(x) = a_0 + a_1x + a_2x^2 + \dots + a_nx^n$$

It involves $n+1$ coefficient a_i .

We can impose the condition $P_n(x_i) = y_i$, $i=0,1,\dots,n$

$$a_0 + a_1x_i + a_2x_i^2 + \dots + a_nx_i^n = y_i$$

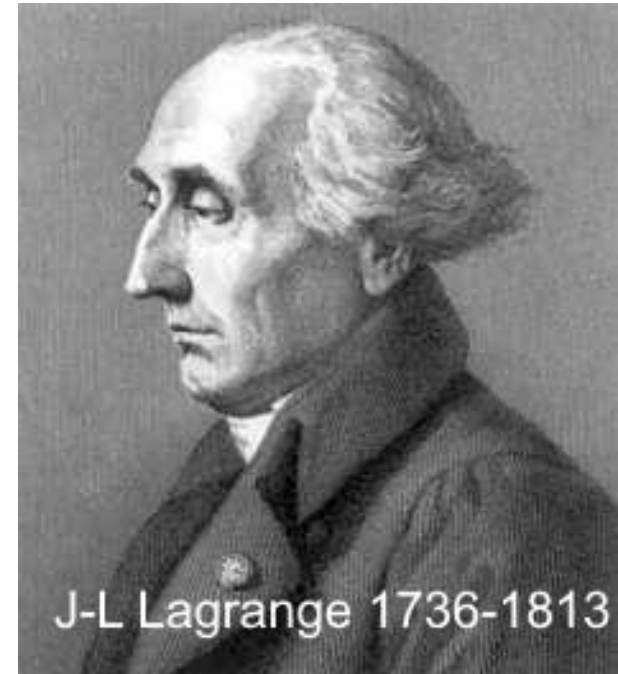
and obtain the system of $n+1$ algebraic equations with respect to a_i . The system can be solved with methods described in Chapter 2.

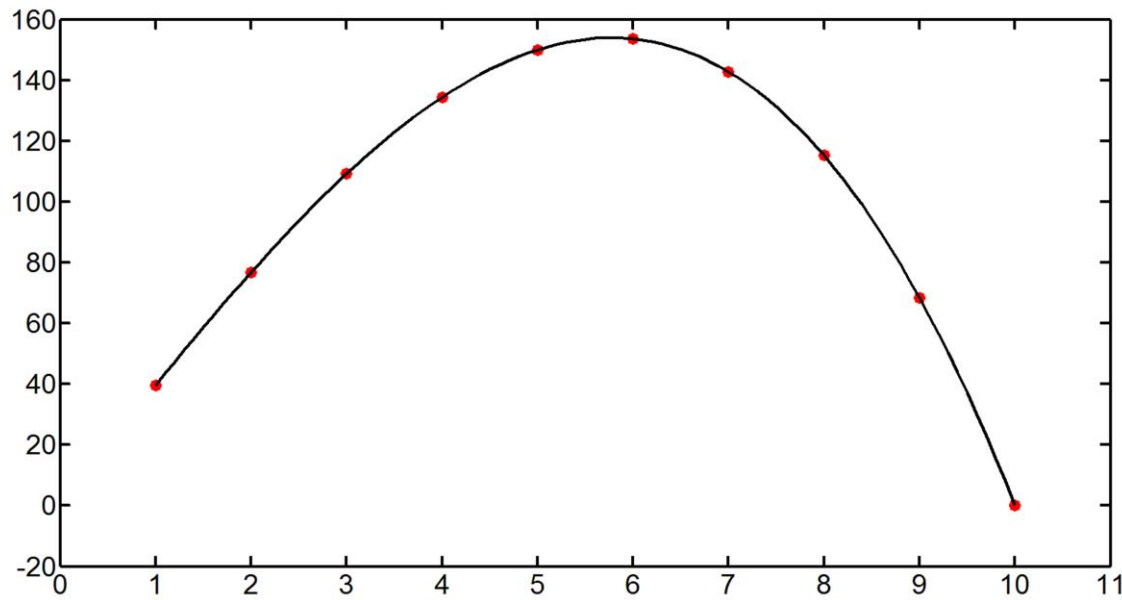
Unfortunately, this way of getting $P_n(x)$ needs a great deal of computations.

Lagrange suggested a form of the required polynomial which does not need solving any algebraic equations.

Lagrange's polynomial:

$$\begin{aligned} P(\mathbf{x}) = & y_0 \frac{(\mathbf{x}-x_1)(\mathbf{x}-x_2) \dots (\mathbf{x}-x_n)}{(x_0-x_1)(x_0-x_2) \dots (x_0-x_n)} + \\ & + y_1 \frac{(\mathbf{x}-x_0)(\mathbf{x}-x_2) \dots (\mathbf{x}-x_n)}{(x_1-x_0)(x_1-x_2) \dots (x_1-x_n)} + \\ & \dots + y_n \frac{(\mathbf{x}-x_0)(\mathbf{x}-x_2) \dots (\mathbf{x}-x_{n-1})}{(x_n-x_0)(x_n-x_2) \dots (x_n-x_{n-1})} \end{aligned}$$





Regarding the error of approximation $f(x) \approx P(x)$
at $x_i < x < x_{i+1}$:

Theorem

If there exist derivatives $f^{n+1}(x)$ of function $f(x)$ up to the order $n+1$ on segment $[x_0, x_n]$, then

$$f(x) - P(x) = f^{n+1}(c) \cdot (x-x_0)(x-x_1) \dots (x-x_n)/(n+1)!$$

where $x_0 \leq c \leq x_n$, $!$ is the factorial
(proof is omitted)

```

clear
x0= 1
x1= 2
x2= 3
x3= 5
    y0= 2
    y1= 2.9
    y2= 4.2
    y3= 6
plot(x0,y0,'o',x1,y1,'o',x2,y2,'o',x3,y3,'o')
    for i=1:101
        x= 1+ (i-1)*4*0.01
        y=y0* (x-x1)*(x-x2)*(x-x3) / ((x0-x1)*(x0-x2)*(x0-x3) ) +...
            y1* (x-x0)*(x-x2)*(x-x3) / ((x1-x0)*(x1-x2)*(x1-x3) ) + ...
            y2* (x-x0)*(x-x1)*(x-x3) / ((x2-x0)*(x2-x1)*(x2-x3) ) +...
            y3* (x-x0)*(x-x1)*(x-x2) / ((x3-x0)*(x3-x1)*(x3-x2))
        xp(i)=x
        yp(i)=y
    end
plot(xp,yp,'k','LineWidth',3)

```

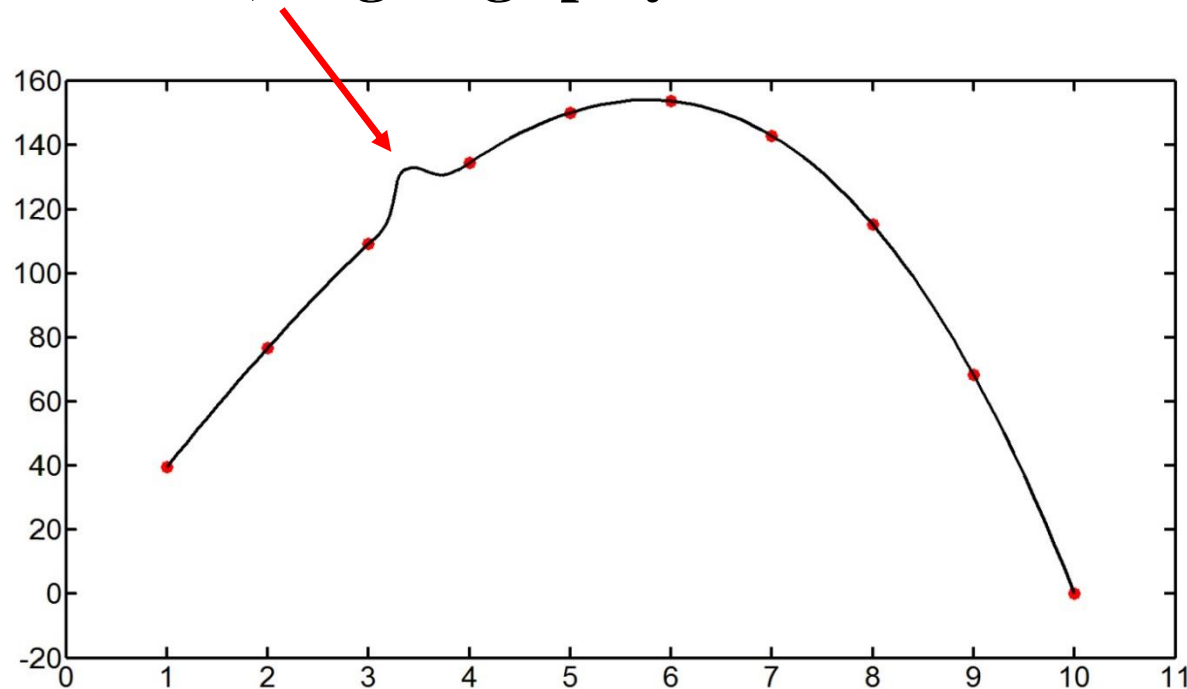
```

clear
x0= 1
x1= 2
x2= 3
x3= 5
    y0= 2
    y1= 2.9
    y2= 4.2
    y3= 6
plot(x0,y0,'o',x1,y1,'o',x2,y2,'o',x3,y3,'o')
    for i=1:101
        x= 1+ (i-1)*4*0.01
        y=y0* (x-x1)*(x-x2)*(x-x3) / ((x0-x1)*(x0-x2)*(x0-x3) ) +...
          y1* (x-x0)*(x-x2)*(x-x3) / ((x1-x0)*(x1-x2)*(x1-x3) ) + ...
          y2* (x-x0)*(x-x1)*(x-x3) / ((x2-x0)*(x2-x1)*(x2-x3) ) +...
          y3* (x-x0)*(x-x1)*(x-x2) / ((x3-x0)*(x3-x1)*(x3-x2))
        xp(i)=x
        yp(i)=y
    end
plot(xp,yp,'k','LineWidth',3)

    ypp=interpLn([x0 x1 x2 x3; y0 y1 y2 y3],xp)
    plot(xp,ypp,'r','LineWidth',3)

```


Sometimes, Lagrange polynomial exhibits an unusual behavior:



This may happen when degree n of the polynomial is large

This is a specific property of polynomials.

As a result, we get a bad approximation of function $f(x)$.