

1 Problem 1

Question: State and verify the Lagrange Interpolation Theorem.

1.1 What is Lagrange Interpolation Theorem?

In numerical analysis, **Lagrangian interpolation** is a polynomial interpolation method named after French 18th century mathematician Joseph Lagrange. Many practical problems use functions to represent some kind of internal connection or law, and many functions can only be understood through experiments and observations. If you observe a physical quantity in practice and get the corresponding observations in several different places, Lagrangian interpolation can find a polynomial that takes the observed value at each observation point. Such a polynomial is called a **Lagrangian (interpolation) polynomial**. Mathematically, Lagrangian interpolation gives a polynomial function that happens to traverse several known points on a two-dimensional plane. The Lagrange interpolation method was first discovered by the British mathematician Edward Waring in 1779, and soon after (1783) was discovered again by Leonhard Euler. In 1795, Lagrange published this interpolation method in his book “Basic Tutorials for Normal Schools”, and his name has since been associated with this method.

For a given $n + 1$ points $(x_0, y_0), (x_1, y_1), \dots, (x_n, y_n)$, there is only one Lagrangian polynomial L corresponding to their number of times not exceeding n . If a polynomial with a higher number of counts is counted, there is an infinite number because all polynomials that differ from L by $\lambda(x_0, y_0)(x_1, y_1) \dots (x_n, y_n)$ satisfy the condition.

1.2 Definition

For a polynomial function, the given $n + 1$ value points are known:

$$(x_0, y_0), (x_1, y_1), \dots, (x_n, y_n)$$

Where x_j corresponds to the position of the argument,
and y_j corresponds to the value of the function at this position.

Assuming that any two different x_j are different from each other, the **Lagrangian interpolation polynomial** obtained by applying the Lagrangian interpolation formula is:

$$L(x) := \sum_{j=0}^n y_j \ell_j(x) \quad (1)$$

Each of $\ell_j(x)$ is a **Lagrangian basic polynomial** (or **interpolation basis function**) whose expression is:

$$\begin{aligned} \ell_j(x) &:= \prod_{i=0, i \neq j}^n \frac{x - x_i}{x_j - x_i} \\ &= \frac{x - x_0}{x_j - x_0} \dots \frac{x - x_{j-1}}{x_j - x_{j-1}} \frac{x - x_{j+1}}{x_j - x_{j+1}} \dots \frac{x - x_n}{x_j - x_n} \end{aligned} \quad (2)$$

The **Lagrangian basic polynomial** $\ell_j(x)$ is characterized by a value of **1** on x_j and a value of **0** on other points $x_i, i \neq j$.

1.3 Proof of existence

For a given $n + 1$ point: $(x_0, y_0), \dots, (x_n, y_n)$, the idea of Lagrange interpolation is to find one at a point x_j takes a value of **1**, while at other points the polynomial $\ell_j(x)$ whose value is **0**. Thus, the polynomial $y_j \ell_j(x)$ takes the value y_j at point x_j and **0** at other points.

And the polynomial

$$L(x) := \sum_{j=0}^n y_j \ell_j(x)$$

can satisfy :

$$L(x_j) = \sum_{i=0}^n y_i \ell_i(x_j) = 0 + 0 + \dots + y_j + \dots + 0 = y_j$$

Polynomials with a value of **0** at other points are easy to find, for example:

$$(x - x_0) \cdots (x - x_{j-1})(x - x_{j+1}) \cdots (x - x_n)$$

It takes the value $(x_j - x_0) \cdots (x_j - x_{j-1})(x_j - x_{j+1}) \cdots (x_j - x_n)$ at point x_j .

Since x_i has been assumed to be different from each other, the above value is not equal to zero. Thus, by dividing the polynomial by this value, you get a polynomial that satisfies ”**1** at x_j and **0** at other points”:

$$\begin{aligned} \ell_j(x) &:= \prod_{i=0, i \neq j}^n \frac{x - x_i}{x_j - x_i} \\ &= \frac{x - x_0}{x_j - x_0} \cdots \frac{x - x_{j-1}}{x_j - x_{j-1}} \frac{x - x_{j+1}}{x_j - x_{j+1}} \cdots \frac{x - x_n}{x_j - x_n} \end{aligned}$$

This is the Lagrangian basic polynomial.

1.4 Proof of uniqueness

There is only one Lagrangian polynomial with a number not exceeding n , because the Lagrangian polynomial does not exceed n for any two times: P_1 and P_2 , their difference $P_1 - P_2$ is $\mathbf{0}$ at all $n + 1$ points, so it must be a polynomial $(x - x_0)(x - x_1) \cdots (x - x_n)$ multiples.

Therefore, if the difference $P_1 - P_2$ is not equal to $\mathbf{0}$, the number of times must not be less than $n + 1$. However, $P_1 - P_2$ is the difference between two polynomials whose number does not exceed n , and the number of times does not exceed n . So $P_1 - P_2 = 0$, that is to say $P_1 = P_2$. This proves the uniqueness.

1.5 Geometric property

Lagrangian basic polynomial $\ell_0, \ell_1, \dots, \ell_n$ used in Lagrangian interpolation (by a group $x_0 < x_1 < \dots < x_n$ determined) can be thought of as a linear space consisting of polynomials of not more than n a set of bases of $\mathbb{K}_n[X]$.

First, if there is a set of coefficients: $\lambda_0, \lambda_1, \dots, \lambda_n$

let $P = \lambda_0 \ell_0 + \lambda_1 \ell_1 + \dots + \lambda_n \ell_n = \mathbf{0}$

Then, on the one hand, the polynomial P is the Lagrangian interpolation polynomial satisfies:

$$P(x_0) = \lambda_0, P(x_1) = \lambda_1, \dots, P(x_n) = \lambda_n$$

On the other hand, P is a zero polynomial, so the value is always $\mathbf{0}$.

So:

$$\lambda_0 = \lambda_1 = \dots = \lambda_n = \mathbf{0}$$

This proves that $\ell_0, \ell_1, \dots, \ell_n$ are linearly independent. At the same time it contains a total of $n + 1$ polynomials, which is exactly equal to the dimension of $\mathbb{K}_n[X]$. So $\ell_0, \ell_1, \dots, \ell_n$ form a set of bases for $\mathbb{K}_n[X]$.

The advantage of the Lagrangian basic polynomial as a base is that all polynomials are homogeneous (both n th polynomials).

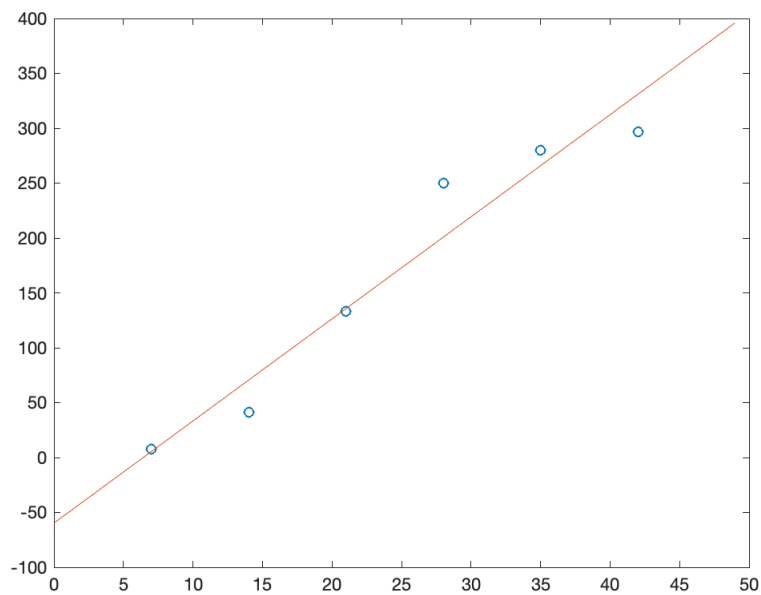
2 Problem 2

Question: Give some data

$t(\text{days})$	7	14	21	28	35	42
$P(\text{number of observed flies})$	8	41	133	250	280	297

- (a) Use the one-term model to fit the data via the least-squares method.
- (b) Use fifth-order polynomial to interpolate the data.
- (c) Use low-order polynomial to minimize the L_1 norm, L_2 norm and L_∞ norm.
Try second, third order polynomials, respectively.

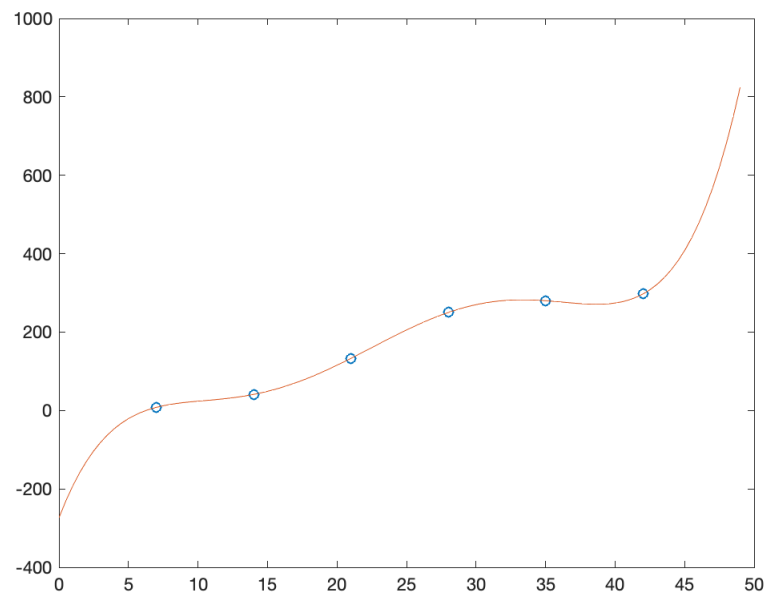
2.1 (a)



$$y = (2279 * x)/245 - 896/15$$

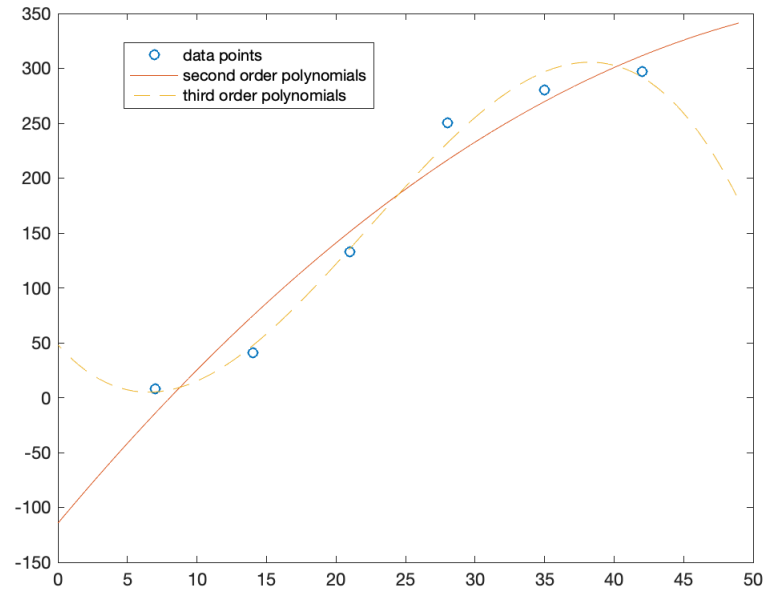
2.2 (b)

The Lagrangian interpolation is used to interpolate the data:



$$y = (11 * x^5)/84035 - (145 * x^4)/9604 + (1283 * x^3)/2058 \\ - (2181 * x^2)/196 + (9712 * x)/105 - 274$$

2.3 (c)



Second order polynomials:

$$y = (3714 * x)/245 - (41 * x^2)/343 - 572/5$$

Third order polynomials:

$$y = -(58 * x^3)/3087 + (1298 * x^2)/1029 - (6185 * x)/441 + 48$$

2.4 Matlab code

```
% HW4_2_a.m
x = [7,14,21,28,35,42];
y = [8,41,133,250,280,297];
p = polyfit(x,y,1);
poly2sym(p)
x1 = linspace(0,49);
y1 = polyval(p,x1);
figure
plot(x,y,'o')
hold on
plot(x1,y1)
hold off
```

```
% Hw4_2_b.m
x = [7 14 21 28 35 42];
y = [8 41 133 250 280 297];
x1 = linspace(0,49);
y1 = la(x,y,x1);
plot(x,y,'o')
hold on
plot(x1,y1)
hold off
```

```

% la.m

function yy=la(x1,y1,xx)

syms x

n=length(x1);

    for i=1:n

        t=x1;t(i)=[];

        L(i)=prod((x-t)./(x1(i)-t));% L向量用来存放插值基函数

    end

u=sum(L.*y1);

p=simplify(u) % p是简化后的Lagrange插值函数（字符串）

yy=subs(p,x,xx);    % p是以x为自变量的函数，并求xx处的函数值

end

```

```

% Hw4_2_c.m

x = [7 14 21 28 35 42];

y = [8 41 133 250 280 297];

p2 = polyfit(x,y,2);

p3 = polyfit(x,y,3);

poly2sym(p2)

poly2sym(p3)

x1 = linspace(0,49);

y2 = polyval(p2,x1);

y3 = polyval(p3,x1);

plot(x,y,'o')

hold on

plot(x1,y2,'-')

plot(x1,y3,'—')

```