Mid-term Extra Problems

Chapter 1

Q 1.

Example 1.8 Sensitivity. Consider the tangent function, $f(x) = \tan(x)$. Since $f'(x) = \sec^2(x) = 1 + \tan^2(x)$, we have

$$\text{Condition number} \approx \left|\frac{x f'(x)}{f(x)}\right| = \left|\frac{x(1+\tan^2(x))}{\tan(x)}\right| = \left|x\left(\frac{1}{\tan(x)} + \tan(x)\right)\right|.$$

Compute the condition number using the finite-difference approximation formula for x = 0.1 and x = π + 0.1.

Solution:

Case 1: when $x = 0.1$

$$\text{cond} = \left|\frac{x f'(x)}{f(x)}\right| = \left|\frac{x(1+\tan^2 x)}{\tan x}\right| = \left|x\left(\frac{1}{\tan x} + \tan x\right)\right|$$

$$= \left|0.1 \cdot \left(\frac{1}{\tan 0.1} + \tan 0.1\right)\right| \approx 1.01$$

Case 2: when $x = \pi + 0.1$

$$\text{cond} = \left|\frac{x f'(x)}{f(x)}\right| = \left|x\left(\frac{1}{\tan x} + \tan x\right)\right|$$

$$= \left|(\pi+0.1)\cdot\left(\frac{1}{\tan(\pi+0.1)} + \tan(\pi+0.1)\right)\right|$$

$$= \left|(\pi+0.1)\cdot\left(\frac{1}{\tan 0.1} + \tan 0.1\right)\right|$$

$$\approx 32.63$$

Q 2.

Now let's compute the condition number using the ratio between forward and backward error with Δx = 0.01.

Solution: we have $f(x) = \tan x$

when $\Delta x = 0.01$

① $x = 0.1$

$$\left|\frac{\Delta y}{y}\right| = \left|\frac{f(x+\Delta x) - f(x)}{f(x)}\right| = \left|\frac{f(x+\Delta x)}{f(x)} - 1\right|$$

$$= \left|\frac{\tan 0.11}{\tan 0.1} - 1\right|$$

$$\left|\frac{\Delta x}{x}\right| = \left|\frac{0.01}{0.1}\right| = 0.1$$

$$\Rightarrow \text{cond} = \left|\frac{\Delta y / y}{\Delta x / x}\right| = \frac{|\Delta y / y|}{|\Delta x / x|} = \frac{\left|\frac{\tan 0.11}{\tan 0.1} - 1\right|}{0.1}$$

$$= 10\left|\frac{\tan 0.11}{\tan 0.1} - 1\right|$$

$$\approx 1.00$$

② $x = \pi + 0.1$

$$\left|\frac{\Delta y}{y}\right| = \left|\frac{f(x+\Delta x)}{f(x)} - 1\right| = \left|\frac{\tan(\pi+0.11)}{\tan(\pi+0.1)} - 1\right| = \left|\frac{\tan 0.11}{\tan 0.1} - 1\right|$$

$$\left|\frac{\Delta x}{x}\right| = \left|\frac{0.01}{\pi+0.1}\right| = \frac{0.01}{\pi+0.1}$$

$$\Rightarrow \text{cond} = \left|\frac{\Delta y / y}{\Delta x / x}\right| = \frac{|\Delta y / y|}{|\Delta x / x|} = \frac{\left|\frac{\tan 0.11}{\tan 0.1} - 1\right|}{\frac{0.01}{\pi+0.1}}$$

$$= \frac{(\pi+0.1)\left|\frac{\tan 0.11}{\tan 0.1} - 1\right|}{0.01}$$

$$= 100(\pi+0.1)\left|\frac{\tan 0.11}{\tan 0.1} - 1\right|$$

$$\approx 32.42$$

Q 3.

Which of the following two expressions is better to implement? Explain why. Here, $x - y = \varepsilon > 0$ is smaller than the machine precision, but 2ε is larger than the machine precision.

a) $fl(x-y) + fl(x-y)$
b) $fl((x-y) + (x-y))$

Solution.

It is better to implement (b). The expression (b) means the number $(x-y)+(x-y) = 2(x-y) = 2\varepsilon$ is calculated first, then calculates $fl(2\varepsilon)$.

However, in (a), since $x - y = \varepsilon < \varepsilon_{mach} \Rightarrow fl(x-y) = 0$

Chapter 2

Q 1.

If for a square matrix **A** and a non-trivial vector **z**, we have **Az = 0**, then which of the following are true?

a) A is full rank
$\boxed{\text{b)}}$ A has linearly dependent columns
$\boxed{\text{c)}}$ A is singular.

Solution.

let $A = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{pmatrix}$ and $\vec{z} = \begin{pmatrix} z_1 \\ z_2 \\ \vdots \\ z_n \end{pmatrix}$ (at least one entry in $\vec{z}$ is not zero)

$$\Rightarrow A\vec{z} = \begin{pmatrix} a_{11}z_1 + a_{12}z_2 + \cdots + a_{1n}z_n \\ a_{21}z_1 + a_{22}z_2 + \cdots + a_{2n}z_n \\ \vdots \\ a_{n1}z_1 + a_{n2}z_2 + \cdots + a_{nn}z_n \end{pmatrix} = \vec{0} \Rightarrow (a_{11}+a_{21}+\cdots+a_{n1})z_1 + (a_{12}+a_{22}+\cdots+a_{n2})z_2 + \cdots + (a_{1n}+a_{2n}+\cdots+a_{nn})z_n = 0$$

Since there are at least entries in $\vec{z}$ are non-zero, the columns of A must be linearly dependent to satisfy the equation above $\Rightarrow$ A is rank-deficient and singular

Q 2.

In what case does a square linear system have a unique solution?

Solution.

The square linear system has a unique solution when the square matrix is nonsingular.

Q 3.

For a square linear system, what kind of transformation of the problem leaves the solution unchanged?

Solution.

The solution to the square linear system is unchanged if both sides of the system are premultiplied by a nonsingular matrix. That is to say for $A\vec{x} = \vec{b} \Rightarrow MA\vec{x} = M\vec{b}$ (M is nonsingular)

Q 4.

For a square linear system, does the solution change if the nonsingular matrix is post-multiplied by another matrix? Is the solution recoverable?

Solution.

The solution does change since the system after being post-multiplied by another matrix, denoted by M, is $AM\vec{x} = \vec{b}$, which has a different solution than $A\vec{x} = \vec{b}$

The solution is recoverable. For the system $AM\vec{x}_1 = \vec{b} \Rightarrow \vec{x}_1 = (AM)^{-1}\vec{b} = M^{-1}A^{-1}\vec{b}$

For the system $A\vec{x}_2 = \vec{b} \Rightarrow \vec{x}_2 = A^{-1}\vec{b}$

Thus, we can pre-multiply $\vec{x}_1$ by M so that $\vec{x}_2 = \vec{x}_1$

Q 5.

What is the computational complexity of back-substitution?

Solution.

Assume that the matrix A of the system $A\vec{x} = \vec{b}$ is n by n, then the computational complexity of back-substitution is $O(n^2)$

Q 6.

When is the square linear system Ax = b more ill-conditioned? When A's columns are close to becoming linearly dependent or far from becoming linearly dependent. Explain.

Solution.

The square linear system $A\vec{x} = \vec{b}$ is more ill-conditioned when the A's columns are close to becoming linear dependent since matrix A is close to be singular when it is close to linearly independent, which causes a large condition number, making $A\vec{x} = \vec{b}$ ill-conditioned.

Q 7.

Is it possible to LU factorize the following matrix? How?

$A = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$ $\quad P = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$ $\quad A = LU$

Solution.

possible. Pre-multiply A by a permutation matrix $P = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$

Q 8.

A 2-by-2 square linear system can be represented as two linear equations, each representing a straight line, where the solution x = A⁻¹b is the intersection of the two straight lines. The following two represent two different 2-by-2 square linear systems, depicted through the solid straight lines. Which one is more ill-conditioned? Why?

Solution.

The right system depicted is more ill-conditioned. Since these two lines in the right system are nearly parallel, indicating that coefficients in these two equations are nearly proportional, which means columns in matrix A are nearly linearly independent, indicating that matrix A is nearly singular with large condition number. Thus, the right system is more ill-conditioned.

Q 9.

In the above figures, what do the dashed lines and the gray parallelograms represent? What do they indicate about the solution to their corresponding square linear system?

Solution.

The dashed lines represent the perturbation or error in determining each straight line. Both left and right systems have the same amount of inaccuracy when determining straight lines, depicted by the dashed lines being away from the solid lines by the same amount. This creates the inaccuracy in determining the solution $\vec{x} = A^{-1}\vec{b}$, which is depicted by gray parallelograms.

Q 10.

Which matrix has a higher condition number?

$A = \begin{bmatrix} 2 & 0 \\ 0 & 3 \end{bmatrix} \quad B = \begin{bmatrix} 2 & 0 \\ 0 & 0.5 \end{bmatrix}$

Solution: We can derive that

$A^{-1} = \frac{1}{\det A}\begin{pmatrix} 3 & 0 \\ 0 & 2 \end{pmatrix} = \begin{pmatrix} \frac{1}{2} & 0 \\ 0 & \frac{1}{3} \end{pmatrix}$

$B^{-1} = \frac{1}{\det B}\begin{pmatrix} 0.5 & 0 \\ 0 & 2 \end{pmatrix} = \begin{pmatrix} 0.5 & 0 \\ 0 & 2 \end{pmatrix}$

$\Rightarrow \mathrm{cond}(A) = \|A\|_\infty \|A^{-1}\|_\infty = 3 \times \frac{1}{2} = \frac{3}{2}$

$\mathrm{cond}(B) = \|A\|_\infty \|A^{-1}\|_\infty = 2 \times 2 = 4$

Thus, matrix B has a higher condition number.

Chapter 3

Q 1.

The least squares problem for an overdetermined linear system is given as follows:

$\min_x \|r\|_2^2 = \min_x \|b - Ax\|_2^2$

where r is the residual vector. Is the minimum of the squared norm of the residual vector zero? Explain your answer.

Solution.

Assume that the size of matrix A is m by n $(m > n)$ and $\vec{b} \in \mathbb{R}^m$

Since A is overdetermined, there are more equations than unknowns. When $\vec{r} = \vec{0} \Rightarrow A\vec{x} = \vec{b}$, and there are cases for this linear system.

Case 1: There are identical row entries in A but different row entries in $\vec{b}$ $\Rightarrow$ The system $A\vec{x} = \vec{b}$ cannot be solved

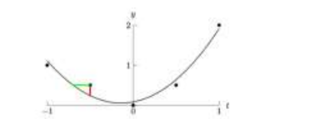
Case 2: There are both identical row entries in A and $\vec{b}$

$\Rightarrow$ The system $A\vec{x} = \vec{b}$ has general solution

Thus, the minimum of the squared norm of the residual vector may not be 0.

Q 2.

The following figure shows five data points (t_i, y_i) and a quadratic polynomial fitting these points. Which of the distances represents an element of the residual vector? The green or the red one? Explain your answer.



Solution.

The red one should represent an element of the residual vector.

The residual vector $\vec{r} = \vec{b} - A\vec{x}$ where the y_i $(1 \le i \le 5)$ consists of $\vec{b}$ and $A\vec{x}$ generates a quadratic polynomial fitting the given data, which both reflect the values in y-axis. $\Rightarrow \|\vec{b} - A\vec{x}\|_2 = \|\vec{r}\|_2$ and the red distance is an element of the residual vector.

(Chapter 3 Continued)

Q 3.

When do we have a unique solution to the least squares problem $Ax \cong b$? How to pick a solution when there are infinitely many solutions?

Solution.

We have a unique solution when matrix A is full rank.

There are infinitely many solutions if A is rank-deficient.

We are prone to chose the solution $\vec{x}$ with the minimum norm.

(1) Solutions with smaller norms are typically more numerically stable. Large-norm solutions can lead to some amplified errors.

(2) The minimum-norm solution has physical significance in many application. For example, in control systems or minimizing energy, the minimum-norm solution fits these kinds of requirements because it corresponds to a solution that uses the least amount of energy.

Q 4.

Is the system of normal equations an overdetermined or a square linear system?

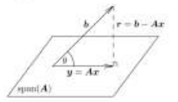
Solution.

The normal equation system is $A^TA\vec{x} = A^T\vec{b}$ where the matrix A^TA is square.

However, the square linear system is $A\vec{x} \simeq \vec{b}$ where the number of rows is greater than the number of columns, which is overdetermined because the number of equations is greater than the number of unknowns.

Q 5.

If, for a least squares problem $Ax \cong b$, we have $A^Tr = 0$, what does it mean in terms of the following figure? In what case will $A^Tr = 0$ hold?



Solution.

For a least squares problem $A\vec{x} = \vec{b}$, $A^T\vec{r} = \vec{0}$ indicates that $\vec{r}$ is orthogonal to span(A), which is shown in the figure.

The norm of the residual vector is minimum in this case.

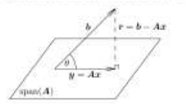
If $\vec{x}$ is not the solution to linear least squares problem, $\vec{r}$ is not of the shortest distance between $\vec{b}$ and span(A), indicating that $\vec{r}$ will not be orthogonal to span(A)

Q 6.

The bound of the condition number of the least squares solution is given as follows:

$\frac{\|\Delta x\|_2}{\|x\|_2} \le \mathrm{cond}(A) \frac{1}{\cos(\theta)} \frac{\|\Delta b\|_2}{\|b\|_2}$

Describe how this bound depends on θ in terms of the following figure.



Solution. Rearrange the inequality, we have

$\mathrm{cond}(\vec{x}) \le \mathrm{cond}(A) \cdot \frac{1}{\cos\theta}$

We can know that matrix A and vector $\vec{b}$ determine the sensitivity of $\vec{x}$ together.

In terms of A, if A is rank-deficient, then cond(A) is large (note that $\theta \in [0, 90°] \Rightarrow \frac{1}{\cos\theta} \in [1, +\infty]$), creating more sensitivity in determining $\vec{x}$.

In terms of θ, if $\vec{b}$ is orthogonal to span(A) $(\theta = 90°)$

$\frac{1}{\cos\theta} \to +\infty$, also creating more sensitivity in determining $\vec{x}$.

(Chapter 3 continued)

$A \to m\times n \ (m>n)$

$$\underset{m\times m}{Q}\ A = Q\begin{pmatrix} R \\ 0 \end{pmatrix} = (\underset{m\times n}{Q_1}\ Q_2)\begin{pmatrix} R \\ 0 \end{pmatrix} \to m\times n = \underset{m\times n}{Q_1} \overset{n\times n}{R}$$

(R: $n\times n$; $\begin{pmatrix} R \\ 0 \end{pmatrix}$: $m\times n$)

Q 7.

Why is it preferred that the least squares problem be solved through QR factorization instead of solving normal equations?

Solution.

Since when using the normal equation to solve $\vec{x}$, $\mathrm{cond}(A^TA) = \mathrm{cond}^2(A)$, which is squared.

However, using QR factorization to decompose A to obtain a upper-triangular matrix to reduce the computational complexity when solving the reduced least squares system.

Q 8.

Show that **cond**(R) = **cond**(A), where **A** = **QR**.

Proof. Assume A is a matrix with size of $m\times n \ (m>n)$

Q is $m\times n$ orthogonal matrix and R is a $n\times n$ upper triangular matrix

Since A is not square $\Rightarrow A^+ = (A^TA)^{-1}A^T$ ①

We have $A = QR$ ② and can derive that $(Q^TQ = I)$ ③

$$A^+ = (A^TA)^{-1}A^T = [(QR)^TQR]^{-1}(QR)^T$$

$$= (R^TQ^TQR)^{-1}R^TQ^T$$

$$= (R^TR)^{-1}R^TQ^T$$

$$= R^{-1}(R^T)^{-1}R^TQ^T$$

$$= R^{-1}Q^T$$

$\Rightarrow \mathrm{cond}(A) = \|A\|\|A^+\| = \|QR\|\|R^{-1}Q^T\|$

$= \|R\|\|R^{-1}Q^T\|$ ④

Since $\|R^{-1}Q^T\| \le \|R^{-1}\|\|Q^T\| = \|R^{-1}\|$ ⑤

$\Rightarrow \|R^{-1}\| = \|R^{-1}\cdot I\| = \|R^{-1}\cdot Q^TQ\| \le \|R^{-1}Q^T\|\|Q\| = \|R^{-1}Q^T\|$ ⑥

From ⑤ and ⑥ $\Rightarrow \|R^{-1}Q^T\| = \|R^{-1}\|$

$\Rightarrow \mathrm{cond}(A) = \|R\|\|R^{-1}Q^T\| = \|R\|\|R^{-1}\| = \mathrm{cond}(R)$

Q 9.

Show that the least squares problem can be reduced through QR factorization to the problem of solving a triangular linear system.

proof.

Assume A is $m\times n$ matrix $(m>n)$

vector $\vec{b} \in \mathbb{R}^m$

Through QR factorization we can obtain

$A = QR$ where $Q \in \mathbb{R}^{m\times n}$ $(Q^TQ = I)$ and $R \in \mathbb{R}^{n\times n}$

and $r_{ij} = 0 \ (i<j)$

From normal equation, we can obtain that

$A^TA\vec{x} = A^T\vec{b} \Rightarrow (QR)^TQR\vec{x} = (QR)^T\vec{b}$

$\Rightarrow R^TQ^TQR\vec{x} = R^TQ^T\vec{b}$

$\Rightarrow R\vec{x} = Q^T\vec{b}$

Since R is $n\times n$ triangular matrix, we can prove that least squares problem can be reduced through QR factorization to the problem of solving a triangular system.

Q 10.

The reduced QR factorization is **A** = **Q₁R**. Then **R** = **Q₁⁻¹A**. Why the following derivation of R is wrong?

Q₁R = **A**, hence, **R** = **Q₁⁻¹A** = **Q₁ᵀA**.

Solution.

Since Q_1 is $m\times n$ matrix so that there is no inverse for Q_1. Thus $R = Q_1^{-1}A$ is an invalid expression

We can derive this formula as following

$A = Q_1R$

Since $Q_1^TQ_1 = I \Rightarrow Q_1^TA = Q_1^TQ_1R = R$

Chapter 7

Q 1.

If I want to interpolate five data points, what polynomial interpolant do I need? Why?

Solution.

Given five data points, to interpolate them, we need to ensure that the outputs of these points are equal to the values of the interpolant

$\Rightarrow$ We need 5 equations $\Rightarrow$ 5 unknowns are needed

Thus, we need to use a quartic polynomial interpolant.

Q 2.

If five data points need a quartic polynomial, could I also use a higher-order polynomial? What is the issue with that?

Solution.

We can use a higher-order polynomial, however, which may cause a underdetermined system with infinitely many solutions.

Q 3.

What are the main computationally involved steps in polynomial interpolation?

Solution.

① Solve the square linear system

② Evaluate the interpolant

Q 4.

For monomial basis functions, what is the computational complexity of the main steps?

Solution.

For monomial basis functions, ① solve the square linear system $O(n^3)$ where the square matrix n by n.

② evaluate the interplant requires $O(n)$

Q 4.

For Lagrange basis functions, what is the computational complexity of the main steps?

Solution.

For Lagrange basis functions, ① solve the square linear system $O(n)$ where the square matrix n by n.

② evaluate the interplant requires $O(n^2)$

proof for ① and ② (non-rigorous)

①: Since Lagrange interpolating polynomial $p(t)$ is expressed

$$p(t) = \sum_{i=0}^{n} y_i l_i(t) = y_0 l_0(t) + y_1 l_1(t) + \cdots + y_n l_n(t)$$

$$= c_0 l_0(t) + c_1 l_1(t) + \cdots + c_n l_n(t)$$

$\Rightarrow c_i = y_i \Rightarrow$ The linear system is

$$\begin{pmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{pmatrix}\begin{pmatrix} c_0 \\ c_1 \\ \vdots \\ c_n \end{pmatrix} = \begin{pmatrix} y_0 \\ y_1 \\ \vdots \\ y_n \end{pmatrix} \quad \text{let } A = \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{pmatrix}$$

Since A is diagonal

$\Rightarrow$ The computational complexity of solving the linear system is $O(n)$

② We have $p(t) = \sum_{i=0}^{n} y_i l_i(t)$

where $l_i(t) = \prod_{0\le j\le n,\ j\ne i} \frac{t - t_j}{t_i - t_j}$

Specifically, when evaluating the interpolant

$$l_i(t_{eval}) = \prod_{0\le j\le n,\ j\ne i} \frac{t_{eval} - t_j}{t_i - t_j}$$

Thus, each term $\frac{t_{eval} - t_j}{t_i - t_j}$ involves:

1 subtraction $t_{eval} - t_j$

1 division by constant $t_i - t_j$

Thus, evaluating $l_i(t_{eval})$ involves

n subtractions, n divisions and $(n-1)$ multiplications to compute the product

$\Rightarrow$ Total operations in computing $= 3n - 1 = O(n)$

$\Rightarrow$ Total cost of computing $p(t)$

$= (n - 0 + 1)(3n - 1)$

$= (n+1)(3n-1) = 3n^2 + 2n - 1 = O(n^2)$

(Chapter 7 continued)

Q 4.

For Newton basis functions, what is the computational complexity of the main steps?

Solution.

For Newton basis functions ① solve the square linear system $O(n^2)$ where the square matrix n by n.

② evaluate the interpolant requires $O(n)$

proof for ②

Since Newton Interpolant polynomial $P_{n-1}(t)$ can be expressed

$$P_{n-1}(t) = x_1 + x_2(t-t_1) + x_3(t-t_1)(t-t_2) + \cdots + x_n(t-t_1)(t-t_2)\cdots(t-t_{n-1})$$

Specifically, when evaluating the interpolant

$$\begin{aligned} P(t_{eval}) &= x_1 + x_2(t_{eval}-t_1) + x_3(t_{eval}-t_1)(t_{eval}-t_2) + \cdots + x_n(t_{eval}-t_1)(t_{eval}-t_2)\cdots(t_{eval}-t_{n-1}) \\ &= x_1 + (t_{eval}-t_1)[x_2 + x_3(t_{eval}-t_2) + \cdots + x_n(t_{eval}-t_2)\cdots(t_{eval}-t_{n-1})] \\ &= x_1 + (t_{eval}-t_1)[x_2 + (t_{eval}-t_2)[x_3 + (t_{eval}-t_3)(\cdots(x_{n-1} + x_n(t_{eval}-t_{n-1}))\cdots)]] \end{aligned}$$

In this formula, we can derive the number of operations

n additions ; n subtractions ; n multiplications

$\Rightarrow$ Total cost $= 3n = O(n)$

Q 5. (hard)

Newton basis functions can be added to an existing interpolant when a new data point arrives without developing the interpolant for the whole data set from scratch. Why it is not possible for monomial or Lagrange basis functions?

Solution.

Assume that there exists $(n-1)$ data points with an existing interpolant already.

We have the nth data point that arrives and use Newton basis function.

$$P(t) = x_1 + x_2(t-t_1) + x_3(t-t_1)(t-t_2) + \cdots + x_n(t-t_1)(t-t_2)\cdots(t-t_{n-1})$$

For the previous $(n-1)$ data points, the last term of $P(t)$ $x_n(t-t_1)(t-t_2)\cdots(t-t_{n-1})$ can always be zero for each point among them, indicating that the first $(n-1)$ basis functions do not change.

Thus, the new interpolant still goes through all the existing points without readjusting the first $(n-1)$ knowns $x_1, x_2, \ldots, x_{n-2}, x_{n-1}$

If the interpolant is based on monomial basis function, we have

$$P(t) = x_1 + x_2 t + x_3 t^2 + \cdots + x_n t^{n-1}$$

Hence, nth monomial basis does not become zero for first $(n-1)$ data points.

So we can only readjust the previous unknowns to interpolate again

If the interpolant is based on Lagrange basis function, we have

$$P(t) = y_1 \frac{(t-t_2)(t-t_3)\cdots(t-t_n)}{(t_1-t_2)(t_1-t_3)\cdots(t_1-t_n)} + y_2 \frac{(t-t_1)(t-t_3)\cdots(t-t_n)}{(t_2-t_1)(t_2-t_3)\cdots(t_2-t_{n-1})} + \cdots + y_{n-1} \frac{(t-t_1)(t-t_2)\cdots(t-t_{n-2})(t-t_n)}{(t_{n-1}-t_1)(t_{n-1}-t_2)\cdots(t_{n-1}-t_{n-2})(t_{n-1}-t_n)} + y_n \frac{(t-t_1)(t-t_2)\cdots(t-t_{n-1})}{(t_n-t_1)(t_n-t_2)\cdots(t_n-t_{n-1})}$$

Hence, nth Lagrange basis become zero for previous $(n-1)$ data points

However, the first $(n-1)$ Lagrange basis functions are being changed.

Q 6.

For five data points, how many equations do we have with piecewise polynomial interpolation?

Solution.

We must have four "pieces" to be divided for 5 data points and each piece must go through the left and right points. Hence, we need 8 equations.

Q 7.

For five data points, how many equations does Hermite cubic interpolation produce? How many unknowns does it have?

Solution.

We have four polynomials to be divided for 5 data points.

Each polynomial will have two equations for two points $\Rightarrow$ 8 equations are needed

Additionally we need to ensure the first derivative for each polynomial is continuous, indicating another 3 equations to make sure the derivative is equal are needed.

Hence, the total equations needed are 11.

Each polynomial has four unknowns, hence the number of total unknown is 16.

Q 8.

For five data points, how many equations does cubic spline interpolation produce? How many unknowns does it have?

Solution.

For cubic spline interpolation, the extra constraints compared with Hermite cubic interpolation are second derivatives must be equal, hence, the total equations are $11+3 = 14$

Likewise, 11 unknowns are needed in total.