

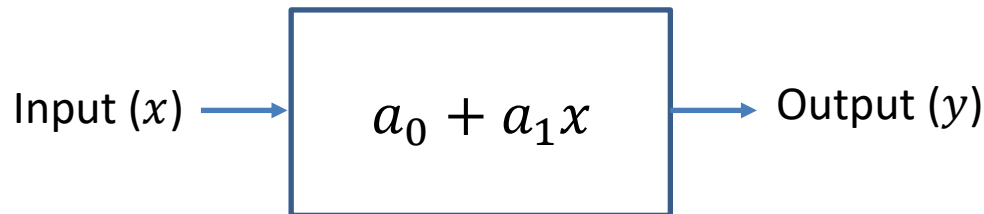
Week 8: Curve Fitting and Interpolation

Acknowledgement: Un-copyrighted photos and pictures were taken from the Internet

Multiple Linear Regression

Linear Least-Squares Regression

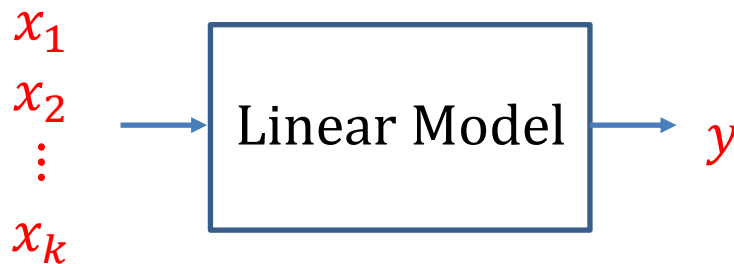
- In the **linear least-squares method**, we build a **linear model** to **predict** the **value of a quantity**.
- A **linear** model will be in the form of $a_0 + a_1x$. Here, a_0 and a_1 are the **parameters** of the **model**.
- We find a_0 and a_1 using the data points.
- The model has **one input (x)** and **one output (y)**.



x y
↓ ↓
 (x_1, y_1)
 (x_2, y_2)
⋮
 (x_n, y_n)

Multiple Linear Regression

- Sometimes, we may need to build a **linear model** with **multiple inputs**.
- In general, a **model** may have k inputs, $x_1, x_2, \dots, x_k$, and 1 output, y .
- This is called **multiple linear regression**.
- We build the model using n data points.

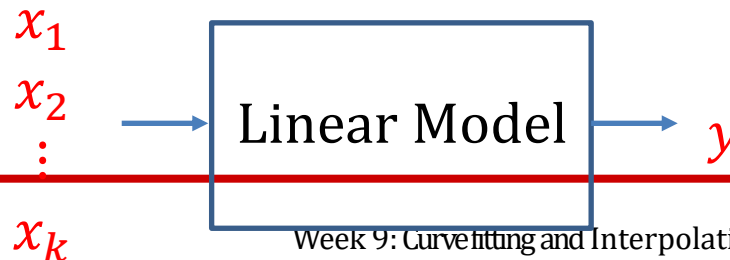
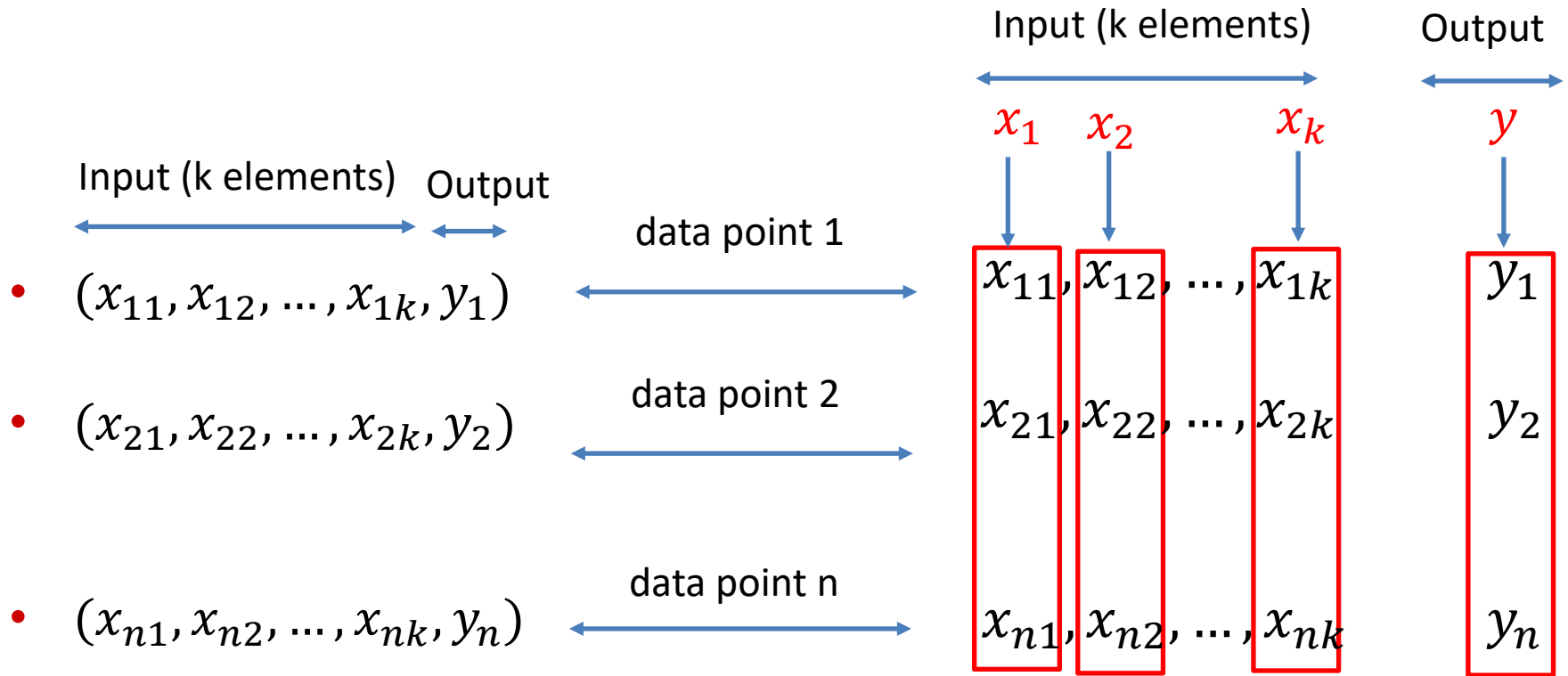


Input (k elements)	Output
$(x_{11}, x_{12}, \dots, x_{1k})$	y_1
$(x_{21}, x_{22}, \dots, x_{2k})$	y_2
$\vdots$	$\vdots$
$(x_{n1}, x_{n2}, \dots, x_{nk})$	y_n

n data points

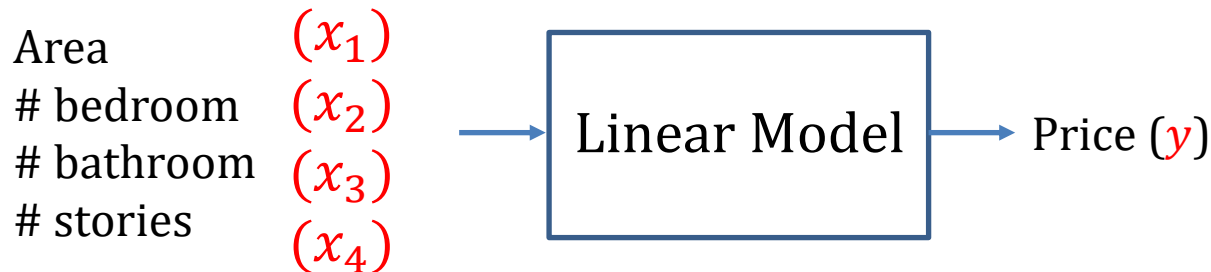
Multiple Linear Regression

- In multiple linear regression, each input data point has **multiple entries**.



Multiple Linear Regression

- For example, we may build a model to **predict the price of a house** based on **its area, the number of bathrooms, bedrooms, and stories**.
- Here, our model will have 4 inputs: area, #bedrooms, #bathrooms, #stories.
- It will have one output.



Multiple Linear Regression

- In multiple linear regression, each input becomes a **vector** with **multiple entries**.

Input					
	x_1	x_2	x_3	x_4	y
data points	area	bedrooms	bathrooms	stories	price
1	7420	4	2	3	13300000
2	8960	4	4	4	12250000
3	9960	3	2	2	12250000
4	7500	4	2	2	12215000
5	7420	4	1	2	11410000

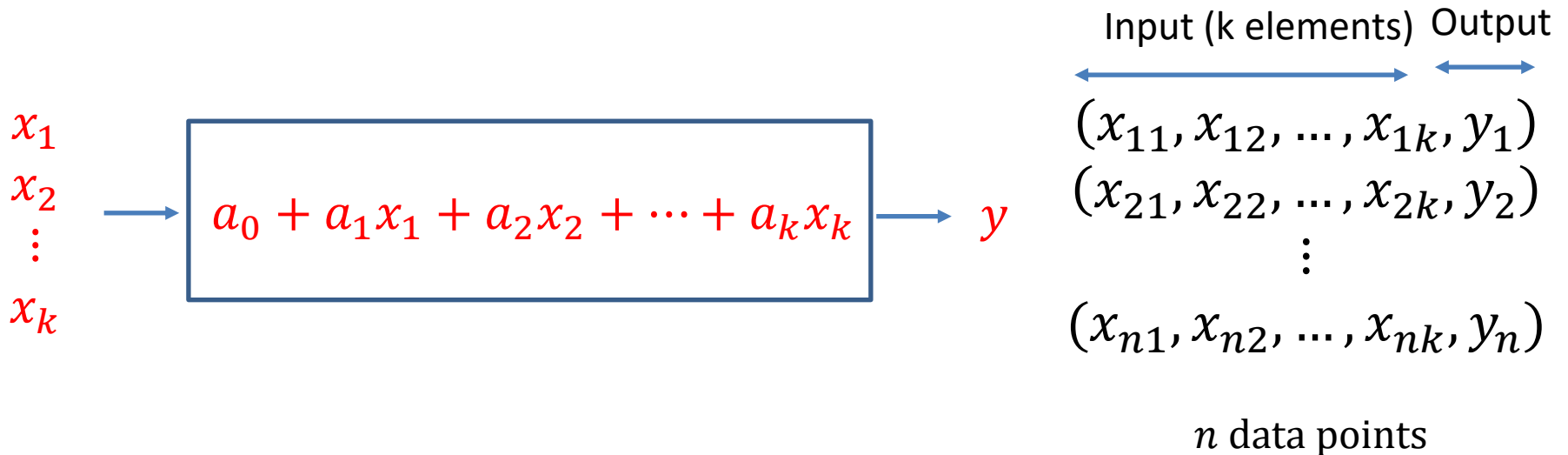


Multiple Linear Regression

- In multiple linear regression, a linear model will be in the form of

$$a_0 + a_1x_1 + a_2x_2 + \cdots + a_kx_k$$

- Here, $a_0, a_1, \dots, a_k$ are constant parameters.
- We need to find the parameters using the data points.



Multiple Linear Regression

- The **optimal** values of $a_0, a_1, a_2, \dots, a_k$ is given by

$$\begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_k \end{bmatrix} = (X^T X)^{-1} X^T Y$$

Optimal parameters

$$\begin{pmatrix} x_{11}, x_{12}, \dots, x_{1k}, y_1 \\ x_{21}, x_{22}, \dots, x_{2k}, y_2 \\ \vdots \\ x_{n1}, x_{n2}, \dots, x_{nk}, y_n \end{pmatrix}$$

n data points

$$X = \begin{bmatrix} 1 & x_{11} & x_{12} & \dots & x_{1k} \\ 1 & x_{21} & x_{22} & \dots & x_{2k} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n1} & x_{n2} & \dots & x_{nk} \end{bmatrix} \quad X^T = \begin{bmatrix} 1 & 1 & 1 & \dots & 1 \\ x_{11} & x_{21} & x_{31} & \dots & x_{n1} \\ x_{12} & x_{22} & x_{32} & \dots & x_{n2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_{1k} & x_{2k} & x_{3k} & \dots & x_{nk} \end{bmatrix} \quad Y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}$$

Multiple Linear Regression

To compute $(X^T X)^{-1} X^T Y$:

1. Compute the matrix X .
2. Compute the matrix X^T using X .
3. To compute $X^T X$, multiply X^T by X .
4. Compute the inverse of $X^T X$.
5. To compute $(X^T X)^{-1} X^T$, multiply the inverse of $X^T X$, i.e., $(X^T X)^{-1}$, by X^T .
6. Multiply Y by the matrix $(X^T X)^{-1} X^T$.

$$\begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_k \end{bmatrix} = (X^T X)^{-1} X^T Y \quad X = \begin{bmatrix} 1 & x_{11} & x_{12} & \dots & x_{1k} \\ 1 & x_{21} & x_{22} & \dots & x_{2k} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n1} & x_{n2} & \dots & x_{nk} \end{bmatrix} \quad X^T = \begin{bmatrix} 1 & 1 & 1 & \dots & 1 \\ x_{11} & x_{21} & x_{31} & \dots & x_{n1} \\ x_{12} & x_{22} & x_{32} & \dots & x_{n2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_{1k} & x_{2k} & x_{3k} & \dots & x_{nk} \end{bmatrix}$$

Optimal parameters

Example

- Build a linear model to predict the **price of a house** based on **its area, number of bathrooms, bedrooms and stories**.
- Use the data in the table below.

Data point	x_1	x_2	x_3	x_4	y
	area	bedrooms	bathrooms	stories	price
1	7420	4	2	3	13300000
2	8960	4	4	4	12250000
3	9960	3	2	2	12250000
4	7500	4	2	2	12215000
5	7420	4	1	2	11410000

Example

- We first form matrix X and vector Y .

$$X = \begin{bmatrix} 1 & 7420 & 4 & 2 & 3 \\ 1 & 8960 & 4 & 4 & 4 \\ 1 & 9960 & 3 & 2 & 2 \\ 1 & 7500 & 4 & 2 & 2 \\ 1 & 7420 & 4 & 1 & 2 \end{bmatrix} \quad Y = \begin{bmatrix} 13300000 \\ 12250000 \\ 12250000 \\ 12215000 \\ 11410000 \end{bmatrix} \quad \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_k \end{bmatrix} = (X^\top X)^{-1} X^\top Y$$

$$\begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \\ a_4 \end{bmatrix} = \begin{bmatrix} 36934664.1791126 \\ -1580.22388059678 \\ -3922350.74626969 \\ 931417.910447620 \\ 479291.044776074 \end{bmatrix}$$

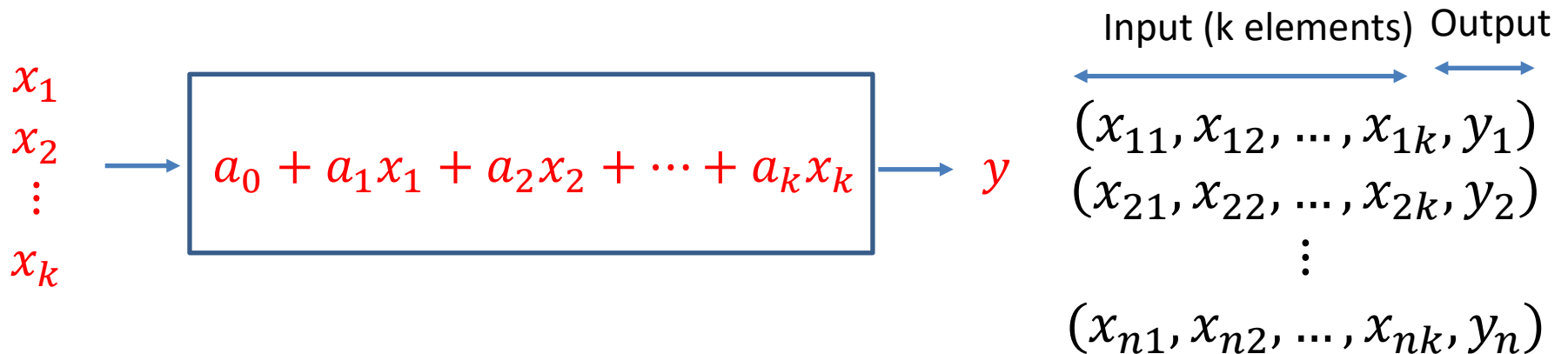
How does it work?

- The **error** for the data point i ($x_{i1}, x_{i2}, \dots, x_{ik}, y_i$) is

$$y_i - (a_0 + a_1x_{i1} + a_2x_{i2} + \dots + a_kx_{ik})$$

- We choose $a_0, a_1, a_2, \dots, a_k$ such that the sum of squares of error is minimized.

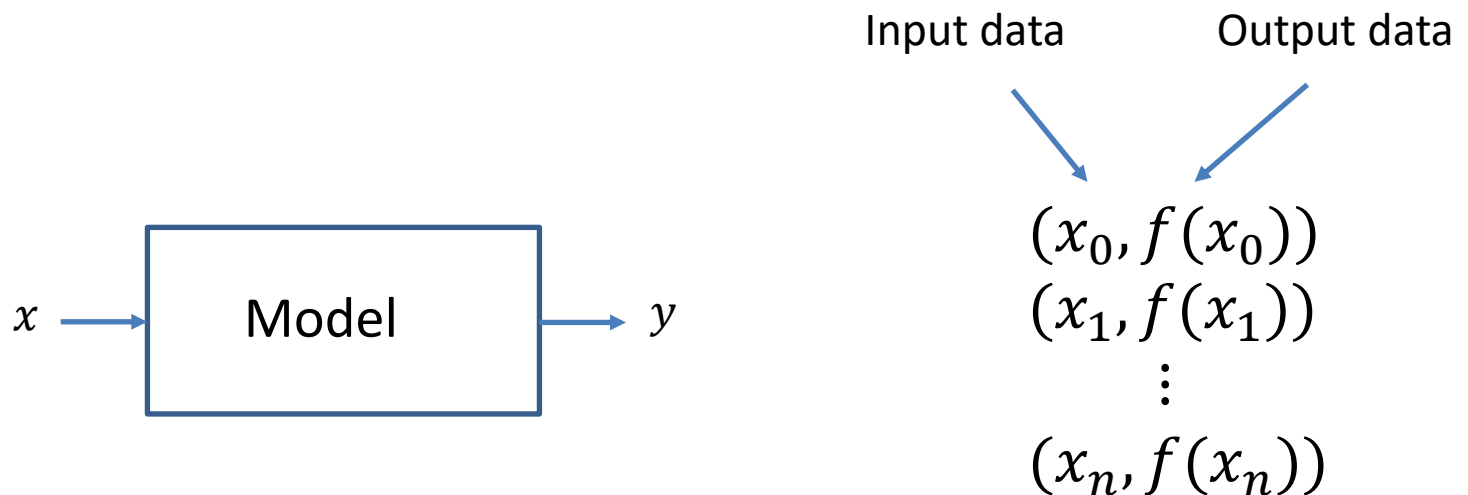
$$\sum_{i=1}^n (y_i - (a_0 + a_1x_{i1} + a_2x_{i2} + \dots + a_kx_{ik}))^2$$



Interpolation

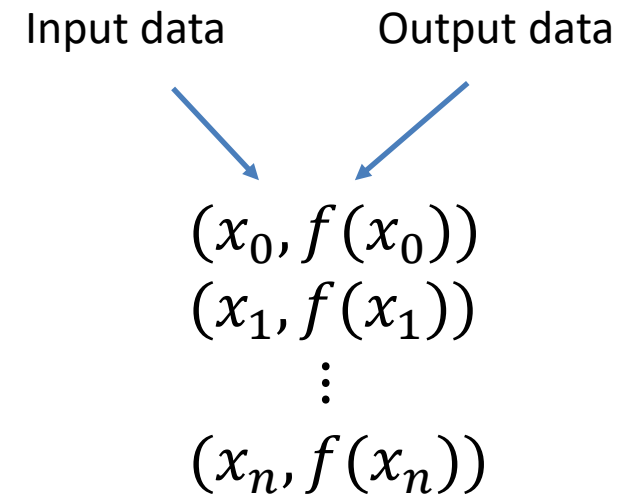
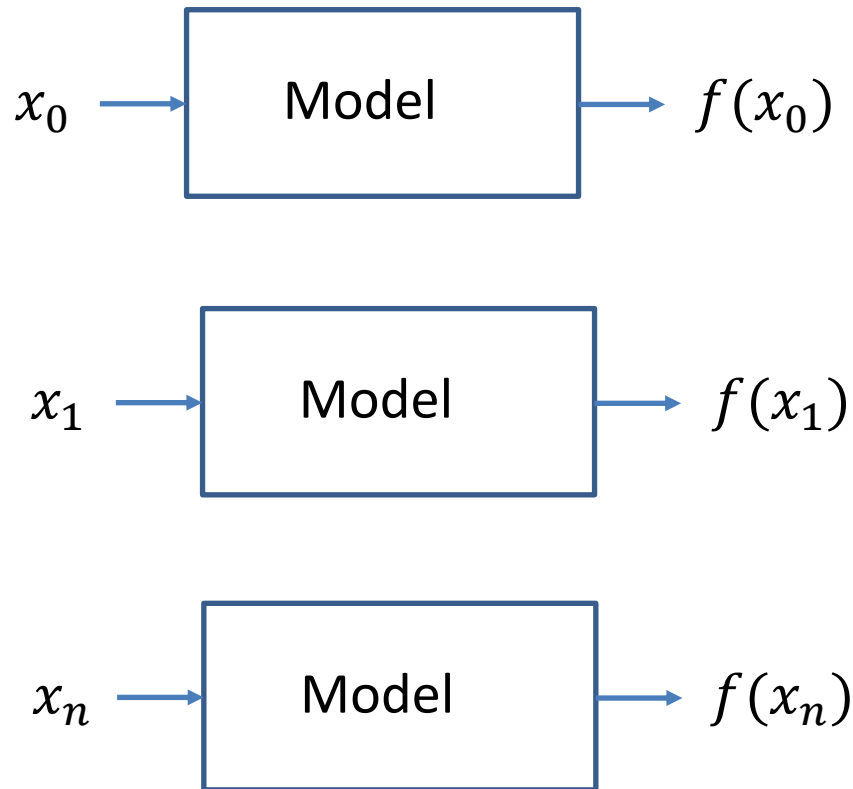
Interpolation

- Like **curve fitting**, in **interpolation** problems we would like to **fit** a **model** to a set of **data points**.
- In **interpolation**, we find the **parameters** of the model such that the model passes through the data points.
- That is, for each **input data point**, the **output of the model should be equal to the output data**.



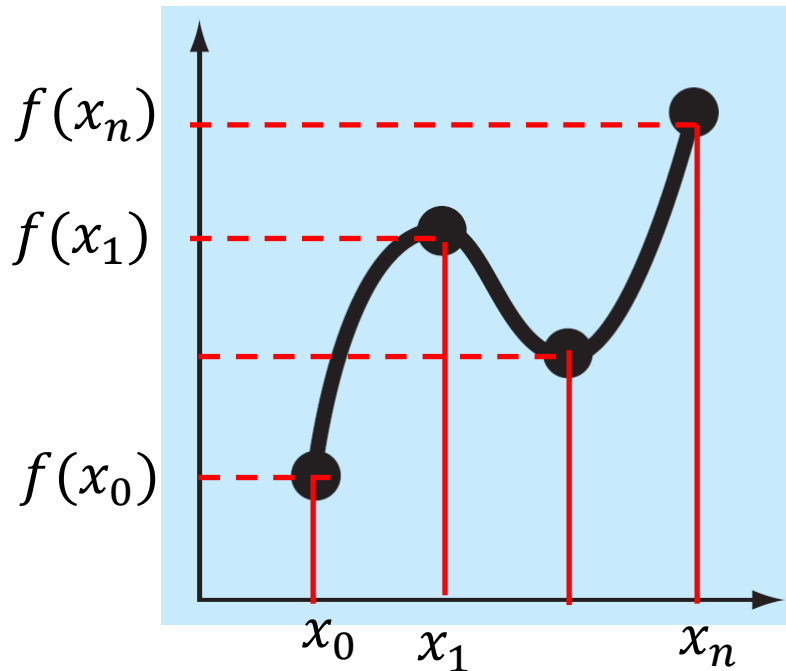
Interpolation

- In **interpolation**, we find the **parameters** of the model such that the **model passes through the data points**.

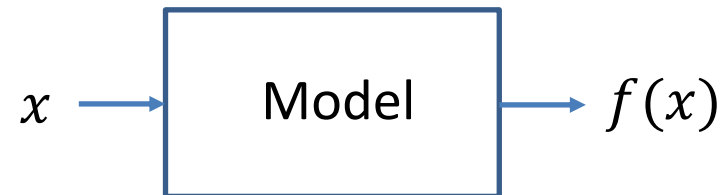


Interpolation

- In **interpolation**, we find the **parameters** of the model such that the **model passes through the data points**.

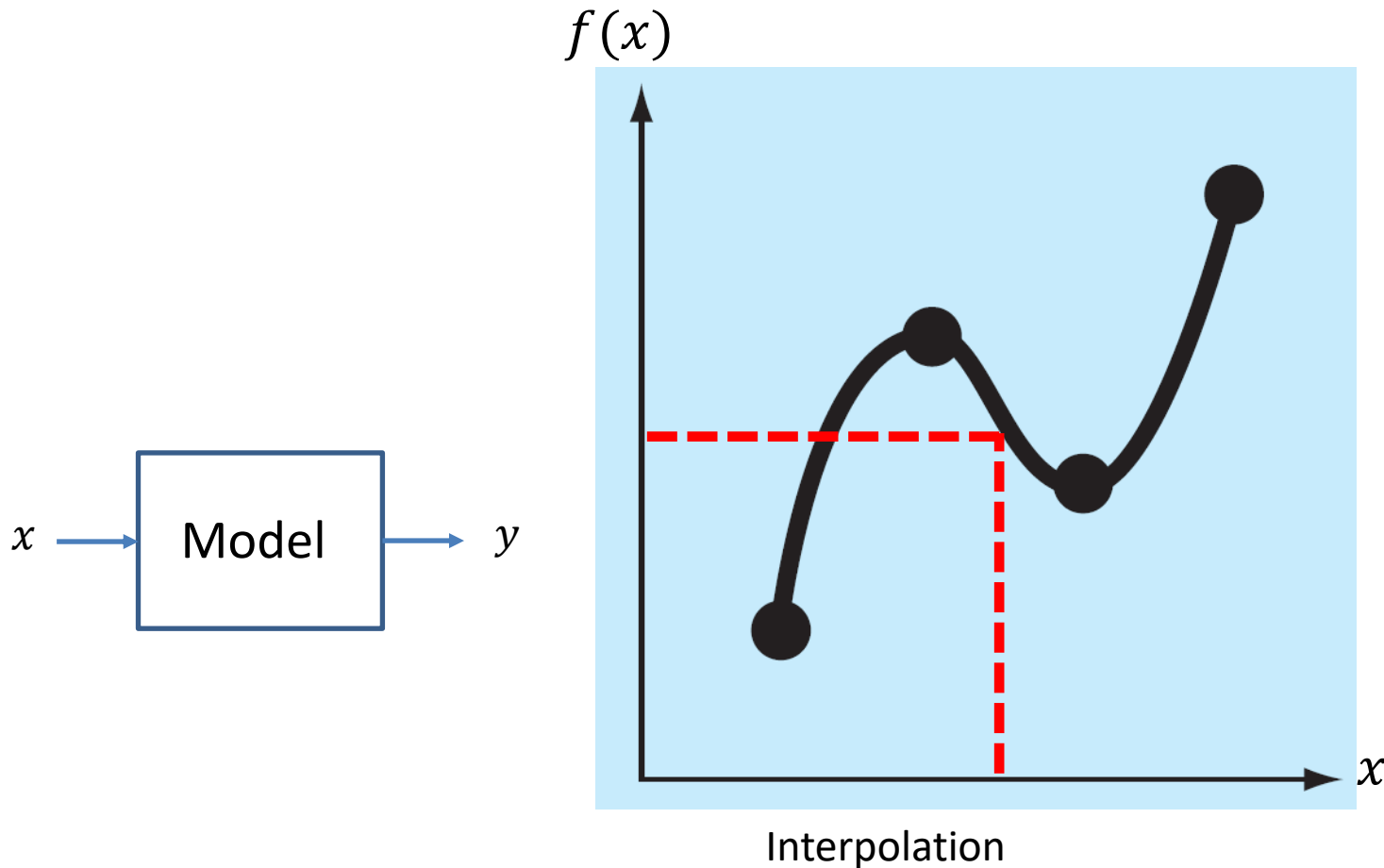


$$\begin{aligned} &(x_0, f(x_0)) \\ &(x_1, f(x_1)) \\ &\vdots \\ &(x_n, f(x_n)) \end{aligned}$$



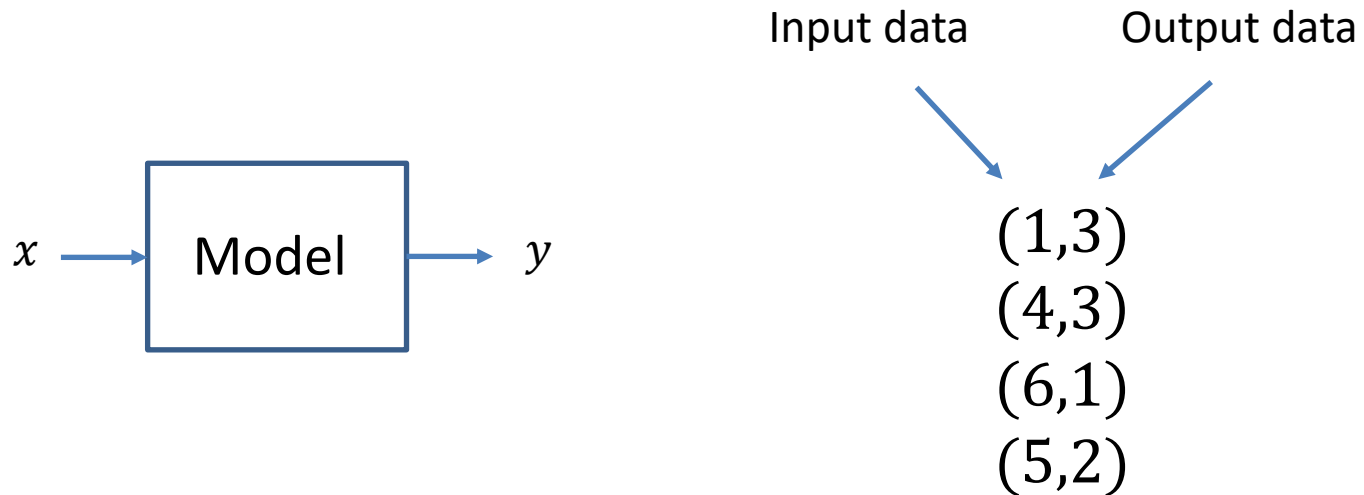
Interpolation

- The model will be used to predict the output values at other points.



Example

- Fit an interpolation model to the following data.



Example

- $x_0 = 1, x_1 = 4, x_2 = 6, x_3 = 5$

- $f(x_0) = 3$

- $f(x_1) = 3$

- $f(x_2) = 1$

- $f(x_3) = 2$

$$(1,3)$$

$$(4,3)$$

$$(6,1)$$

$$(5,2)$$

$$(x_0, f(x_0))$$

$$(x_1, f(x_1))$$

$$(x_2, f(x_2))$$

$$(x_3, f(x_3))$$

$$(x_0, f(x_0))$$

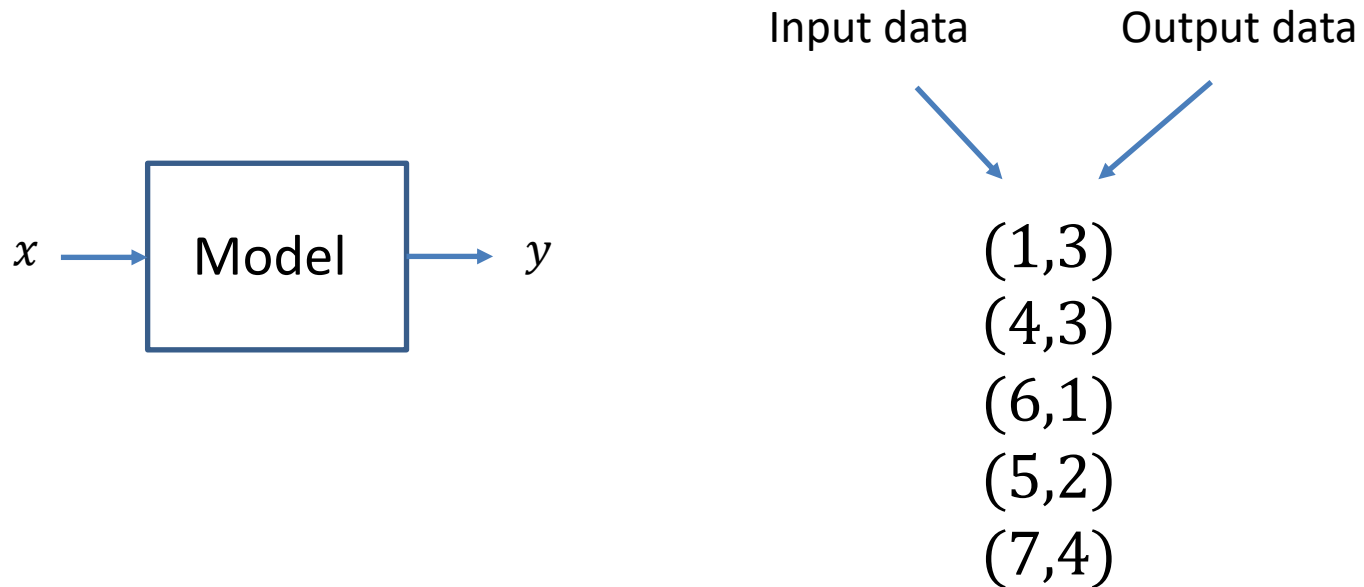
$$(x_1, f(x_1))$$

$$\vdots$$

$$(x_n, f(x_n))$$

Example

- Fit an interpolation model to the following data.



Example

- $x_0 = 1, x_1 = 4, x_2 = 6, x_3 = 5, x_4 = 7$

- $f(x_0) = 3$

- $f(x_1) = 3$

- $f(x_2) = 1$

- $f(x_3) = 2$

- $f(x_4) = 4$

$$(1,3)$$

$$(4,3)$$

$$(6,1)$$

$$(5,2)$$

$$(7,4)$$

$$(x_0, f(x_0))$$

$$(x_1, f(x_1))$$

$$(x_2, f(x_2))$$

$$(x_3, f(x_3))$$

$$(x_4, f(x_4))$$

$$(x_0, f(x_0))$$

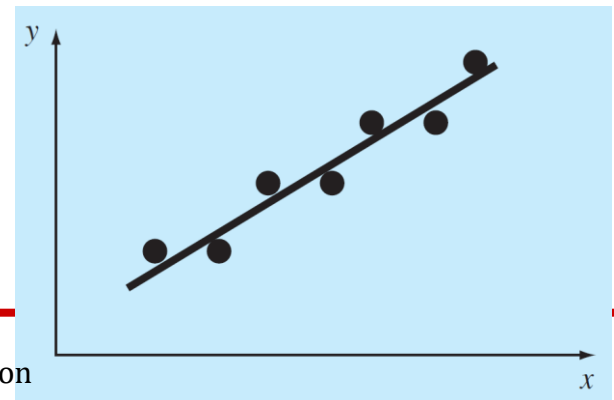
$$(x_1, f(x_1))$$

$$\vdots$$

$$(x_n, f(x_n))$$

Regression

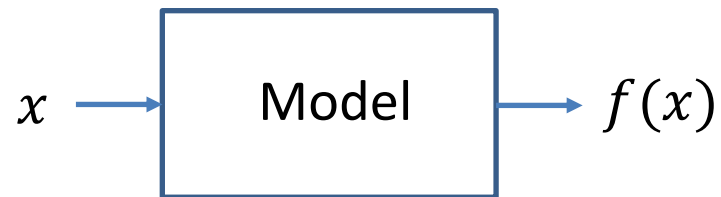
- In the regression, we **do not force** the **model** to **pass through all the data points** because of **uncertainties** in the **model** and **data**.
- In regression, we **may not** have an **accurate model** of the **data**.
- For example, we **may not** have an **accurate model** for the **population**.
- Also, the data points might be **noisy**. That is, the **output values** may not be **accurate**.
- For example, the collected **population data** may not **exact**. Some information might be **missing**.



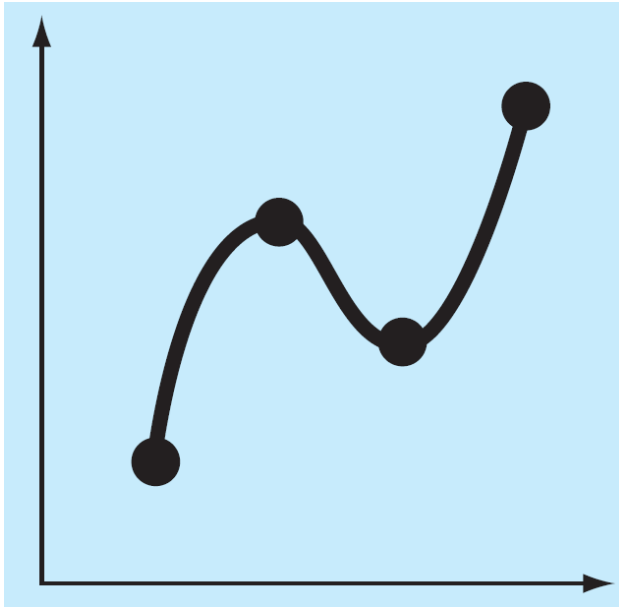
Interpolation

- In **interpolation** problems, the **data points** are **accurate** (not **noisy**).
- Also, we may have an **accurate model** for the **data**.
- But we do not know the parameters of the model.
- We find the **parameters** of the model such that the **model passes through the data points**.

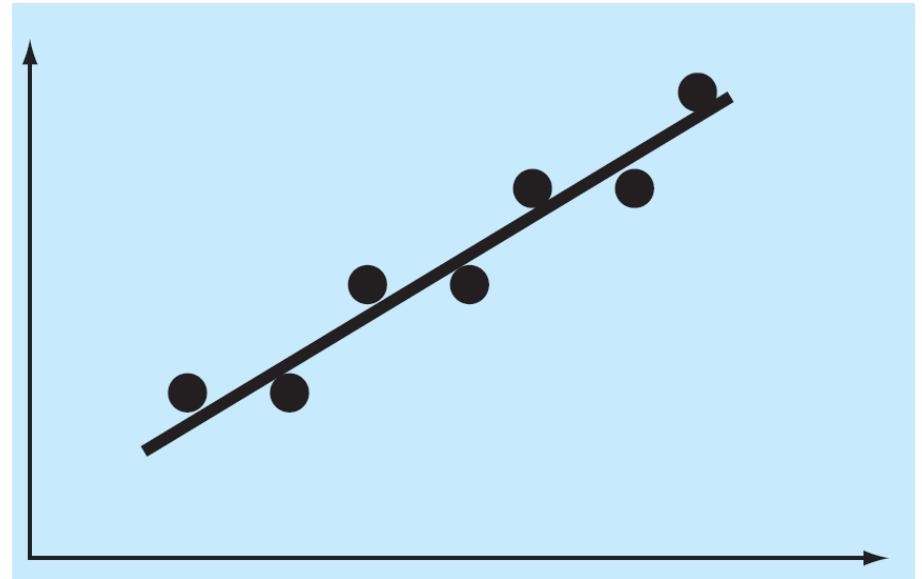
$$\begin{array}{c} (x_0, f(x_0)) \\ (x_1, f(x_1)) \\ \vdots \\ (x_n, f(x_n)) \end{array}$$



Interpolation VS Regression



Interpolation

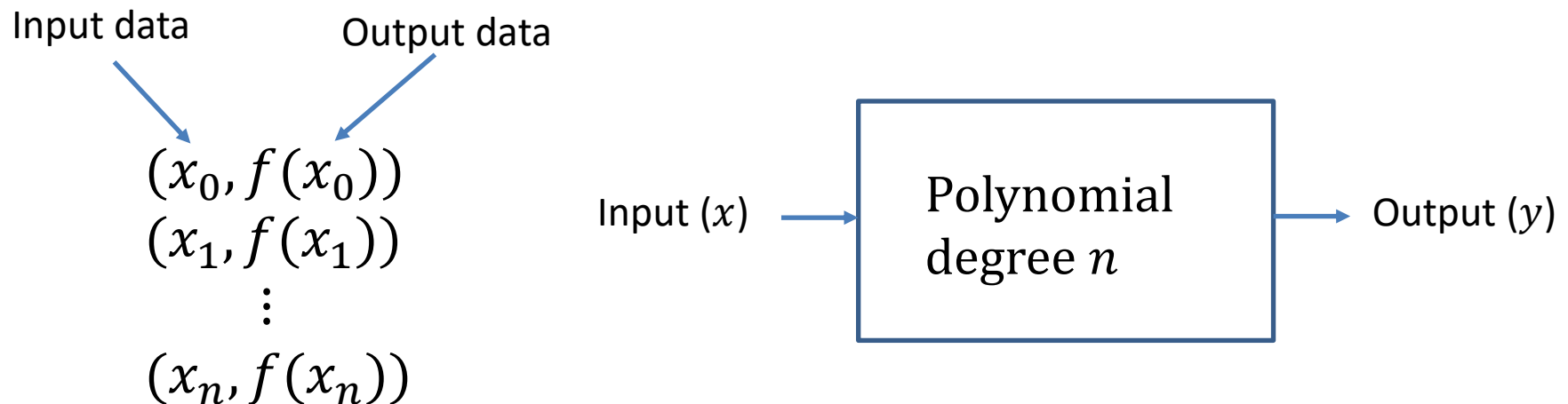


Regression

Newton's Interpolation Method

Newton's Interpolation Method

- Suppose that we have $n + 1$ data points.
- Each data point corresponds to an **input-output** pair.
- In Newton's method, we use a **polynomial** as our **interpolation model**.
- We fit a **polynomial** of **degree n** to $n + 1$ data points.



Newton's Interpolation Method

- If we have $n + 1$ data points, the interpolation model (in the Newton's method), will be following polynomial.

$$f_n(x) = b_0 + b_1(x - x_0) + b_2(x - x_0)(x - x_1) + \cdots + b_n(x - x_0)(x - x_1) \cdots (x - x_{n-1})$$

- The polynomial has $n + 1$ terms.

b_0	First term	$(x_0, f(x_0))$
$b_1(x - x_0)$	Second term	$(x_1, f(x_1))$
$b_2(x - x_0)(x - x_1)$	Third term	$\vdots$
$\vdots$		$(x_{n-1}, f(x_{n-1}))$
$b_n(x - x_0)(x - x_1) \cdots (x - x_{n-1})$	$n + 1$ -th term	$(x_n, f(x_n))$

Newton's Interpolation Method

- $b_0, b_1, \dots, b_n$ are the constant parameters of the model.
- We find them using the data points.



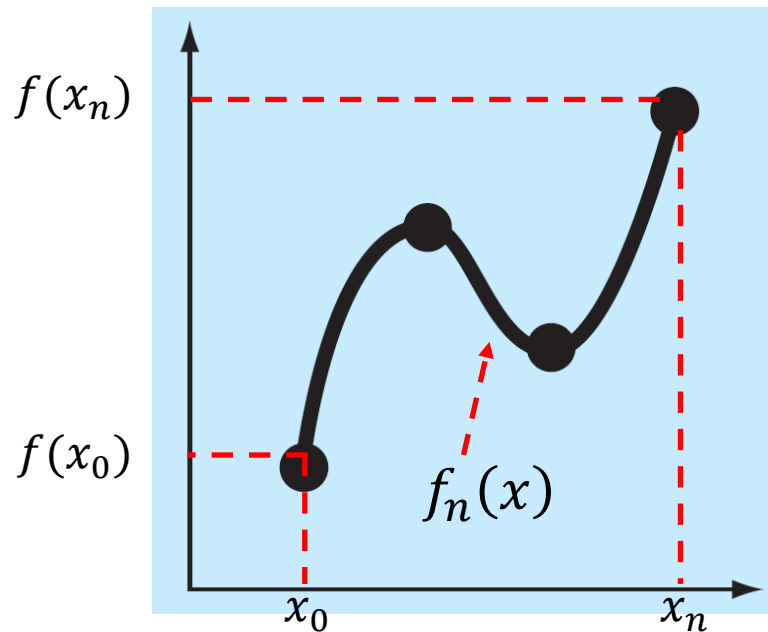
$$\begin{aligned} & (x_0, f(x_0)) \\ & (x_1, f(x_1)) \\ & \vdots \\ & (x_{n-1}, f(x_{n-1})) \\ & (x_n, f(x_n)) \end{aligned}$$

$$f_n(x) = b_0 + b_1(x - x_0) + b_2(x - x_0)(x - x_1) + \cdots + b_n(x - x_0)(x - x_1) \cdots (x - x_{n-1})$$

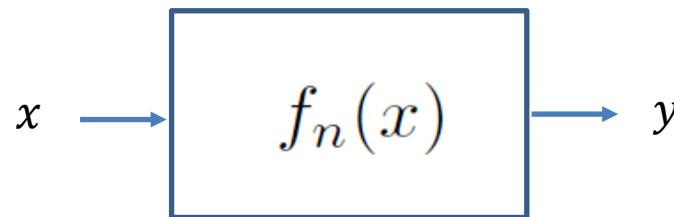
Newton's Interpolation Method

- Problem:** Find the parameters $b_0, b_1, \dots, b_n$ such that the polynomial passes through all the data points.

$$f_n(x) = b_0 + b_1(x - x_0) + b_2(x - x_0)(x - x_1) + \dots + b_n(x - x_0)(x - x_1) \dots (x - x_{n-1})$$



$$\begin{aligned} &(x_0, f(x_0)) \\ &(x_1, f(x_1)) \\ &\vdots \\ &(x_{n-1}, f(x_{n-1})) \\ &(x_n, f(x_n)) \end{aligned}$$



Finite Divided Difference

- Before we proceed with the solution of the Newton's method, we need to introduce the notion of **finite divided differences**.
- For **two data points**, $(x_i, f(x_i))$ and $(x_j, f(x_j))$, the **first finite divided difference** of x_i and x_j is computed as

$$f[x_i, x_j] = \frac{f(x_i) - f(x_j)}{x_i - x_j} \quad \begin{matrix} (x_i, f(x_i)) \\ (x_j, f(x_j)) \end{matrix}$$

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

first finite divided difference of
 $(x_1, f(x_1))$ and $(x_2, f(x_2))$

$$f[x_3, x_2] = \frac{f(x_3) - f(x_2)}{x_3 - x_2}$$

first finite divided difference of
 $(x_2, f(x_2))$ and $(x_3, f(x_3))$

Finite Divided Difference

- For **three data points**, $(x_i, f(x_i))$, $(x_j, f(x_j))$, and $(x_k, f(x_k))$, the **second finite divided difference** of x_i , x_j and x_k is computed using

$$f[\overline{x_i, x_j}, \overline{x_j, x_k}] = \frac{f[\overline{x_i, x_j}] - f[\overline{x_j, x_k}]}{x_i - x_k}$$

- Example: **Second finite divided difference** of $(x_1, f(x_1))$, $(x_2, f(x_2))$, and $(x_3, f(x_3))$.

$$f[x_3, x_2, x_1] = \frac{f[x_3, x_2] - f[x_2, x_1]}{x_3 - x_1}$$

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

$$f[x_3, x_2] = \frac{f(x_3) - f(x_2)}{x_3 - x_2}$$

Finite Divided Difference

- The **third** finite divided difference of x_i, x_j, x_l and x_k is computed using

$$f[x_i, x_j, x_l, x_k] = \frac{f[x_i, x_j, x_l] - f[x_j, x_l, x_k]}{x_i - x_k}$$

- Example: $f[x_3, x_2, x_1, x_0] = \frac{f[x_3, x_2, x_1] - f[x_2, x_1, x_0]}{x_3 - x_0}$

- $f[x_3, x_2, x_1] = \frac{f[x_3, x_2] - f[x_2, x_1]}{x_3 - x_1}$

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

Finite Divided Difference

- The n -th finite divided difference of $x_0, x_1, \dots, x_{n-1}, x_n$ is computed using

$$f[x_n, x_{n-1}, \dots, x_1, x_0] = \frac{f[x_n, x_{n-1}, \dots, x_1] - f[x_{n-1}, x_{n-2}, \dots, x_0]}{x_n - x_0}$$

- Remove the last element (x_0), and compute the divided difference of the remaining elements $(x_n, x_{n-1}, \dots, x_1)$

$$f[x_n, x_{n-1}, \dots, x_1]$$

- Remove the first element (x_n) and compute the divided difference of the remaining elements $(x_{n-1}, x_{n-2}, \dots, x_0)$.

$$f[x_{n-1}, x_{n-2}, \dots, x_0]$$

- Compute the fraction.

$$\frac{f[x_n, x_{n-1}, \dots, x_1] - f[x_{n-1}, x_{n-2}, \dots, x_0]}{x_n - x_0}$$

Newton's Method: General Form

- The solution of the Newton's method is given by the finite divided differences.

$$f_n(x) = b_0 + b_1(x - x_0) + b_2(x - x_0)(x - x_1) + \cdots + b_n(x - x_0)(x - x_1) \cdots (x - x_{n-1})$$

$$b_0 = f(x_0)$$

$$b_1 = f[x_1, x_0]$$

$$b_2 = f[x_2, x_1, x_0]$$

$$b_3 = f[x_3, x_2, x_1, x_0]$$

.

.

$$b_n = f[x_n, x_{n-1}, \dots, x_1, x_0]$$

$$(x_0, f(x_0))$$

$$(x_1, f(x_1))$$

$\vdots$

$$(x_{n-1}, f(x_{n-1}))$$

$$(x_n, f(x_n))$$

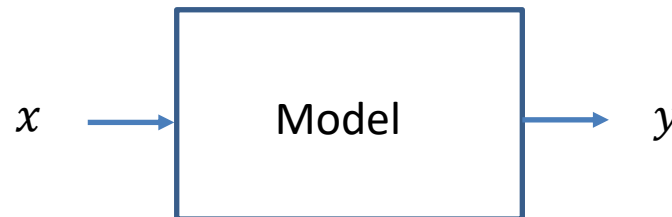
Example

- Using Newton's method, build an **interpolation model** for the function $f(x) = \ln(x)$.

a) Use the following data points to build your model.

$(1, \ln(1))$
 $(4, \ln(4))$
 $(6, \ln(6))$
 $(5, \ln(5))$

b) Use your model to estimate the value of $\ln(2)$



Example

- $x_0 = 1, x_1 = 4, x_2 = 6, x_3 = 5$

- $f(x_0) = \ln(1)$

- $f(x_1) = \ln(4)$

- $f(x_2) = \ln(6)$

- $f(x_3) = \ln(5)$

$$(1, \ln(1)) \quad (x_0, f(x_0))$$

$$(4, \ln(4)) \quad (x_1, f(x_1))$$

$$(6, \ln(6)) \quad (x_2, f(x_2))$$

$$(5, \ln(5)) \quad (x_3, f(x_3))$$

Example

- Since we have **four data points**, we need to fit the following polynomial to the data points.

$$f_3(x) = b_0 + b_1(x - x_0) + b_2(x - x_0)(x - x_1) + b_3(x - x_0)(x - x_1)(x - x_2)$$

b_0	First term	$(1, \ln(1))$
$b_1(x - x_0)$	Second term	$(4, \ln(4))$
$b_2(x - x_0)(x - x_1)$	Third term	$(6, \ln(6))$
$\vdots$		$(5, \ln(5))$
$b_n(x - x_0)(x - x_1) \cdots (x - x_{n-1})$	$n + 1$ -th term	

$$f_n(x) = b_0 + b_1(x - x_0) + b_2(x - x_0)(x - x_1) + \cdots + b_n(x - x_0)(x - x_1) \cdots (x - x_{n-1})$$

Example

- We need to compute the first, second, and third finite divided differences.

$$f_3(x) = b_0 + b_1(x - x_0) + b_2(x - x_0)(x - x_1) + b_3(x - x_0)(x - x_1)(x - x_2)$$

$$b_0 = f(x_0)$$

$$b_1 = f[x_1, x_0]$$

$$b_2 = f[x_2, x_1, x_0]$$

$$b_3 = f[x_3, x_2, x_1, x_0]$$

$$(1, \ln(1))$$

$$(4, \ln(4))$$

$$(6, \ln(6))$$

$$(5, \ln(5))$$

- $x_0 = 1, x_1 = 4, x_2 = 6, x_3 = 5$

Example

- First, start with the **third finite divided difference**.

$$f[x_3, x_2, x_1, x_0] = \frac{f[x_3, x_2, x_1] - f[x_2, x_1, x_0]}{x_3 - x_0}$$

- Then, write down the **second finite divided differences** that are needed to calculate the **third finite divided difference**.

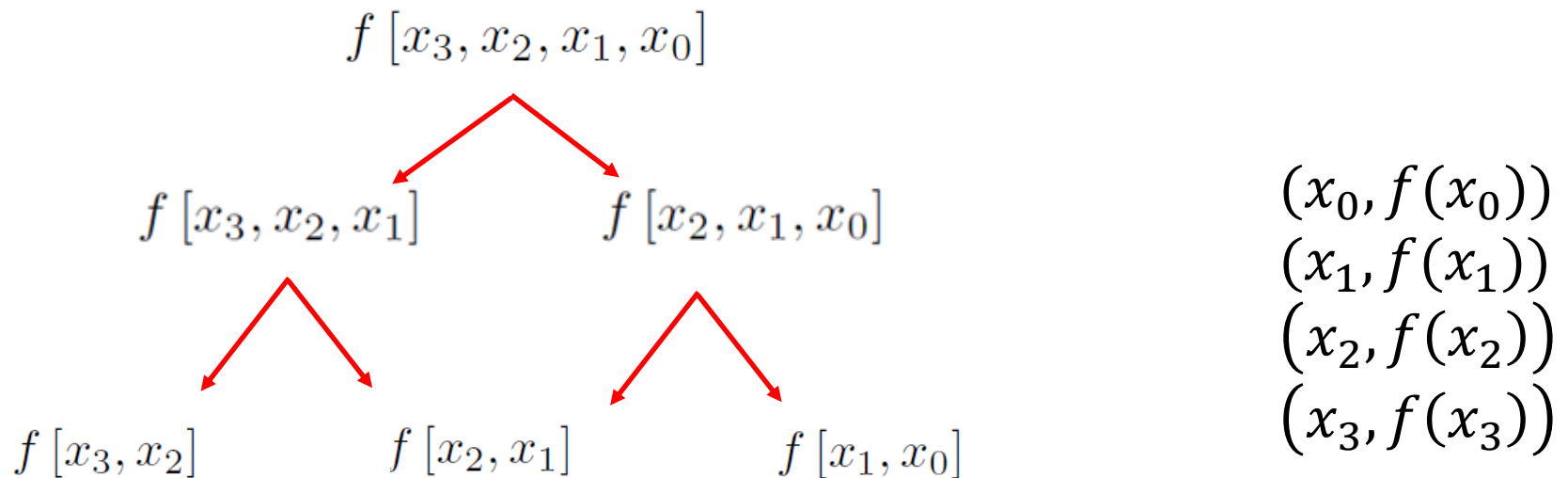
$$f[x_3, x_2, x_1] = \frac{f[x_3, x_2] - f[x_2, x_1]}{x_3 - x_1} \quad f[x_2, x_1, x_0] = \frac{f[x_2, x_1] - f[x_1, x_0]}{x_2 - x_0}$$

- Then, write down the **first finite divided differences** that are needed to calculate the **second finite divided differences**.

$$f[x_3, x_2] = \frac{f(x_3) - f(x_2)}{x_3 - x_2} \quad f[x_2, x_1] = \frac{f(x_2) - f(x_1)}{x_2 - x_1} \quad f[x_1, x_0] = \frac{f(x_1) - f(x_0)}{x_1 - x_0}$$

Example

- You can also form the following **tree**.
- Start with computing the finite divided differences at the lowest level of the tree.
- Then, compute the finite divided differences at upper levels.



Example

$$f[x_1, x_0] = \frac{1.386294 - 0}{4 - 1} = 0.4620981$$

$$f[x_2, x_1] = \frac{1.791759 - 1.386294}{6 - 4} = 0.2027326$$

$$f[x_3, x_2] = \frac{1.609438 - 1.791759}{5 - 6} = 0.1823216$$

- $x_0 = 1, x_1 = 4$
- $x_2 = 6, x_3 = 5$
- $f(x_0) = \ln(1)$
- $f(x_1) = \ln(4)$
- $f(x_2) = \ln(6)$
- $f(x_3) = \ln(5)$

$$f[x_2, x_1, x_0] = \frac{0.2027326 - 0.4620981}{6 - 1} = -0.05187311$$

$$f[x_3, x_2, x_1] = \frac{0.1823216 - 0.2027326}{5 - 4} = -0.02041100$$

$$f[x_3, x_2, x_1, x_0] = \frac{-0.02041100 - (-0.05187311)}{5 - 1} = 0.007865529$$

Example

- The polynomial model is

$$f_3(x) = 0 + 0.4620981(x - 1) - 0.05187311(x - 1)(x - 4) + 0.007865529(x - 1)(x - 4)(x - 6)$$

- For $x = 2$, we have $f_3(x) = 0.6287686$
- $\epsilon_t = 9.3 \%$

Newton's Method: General Form

- The solution of the Newton's method is given by the finite divided differences.

$$f_n(x) = b_0 + b_1(x - x_0) + b_2(x - x_0)(x - x_1) + \cdots + b_n(x - x_0)(x - x_1) \cdots (x - x_{n-1})$$

$$b_0 = f(x_0)$$

$$b_1 = f[x_1, x_0]$$

$$b_2 = f[x_2, x_1, x_0]$$

$$b_3 = f[x_3, x_2, x_1, x_0]$$

.

.

$$b_n = f[x_n, x_{n-1}, \dots, x_1, x_0]$$

$$(x_0, f(x_0))$$

$$(x_1, f(x_1))$$

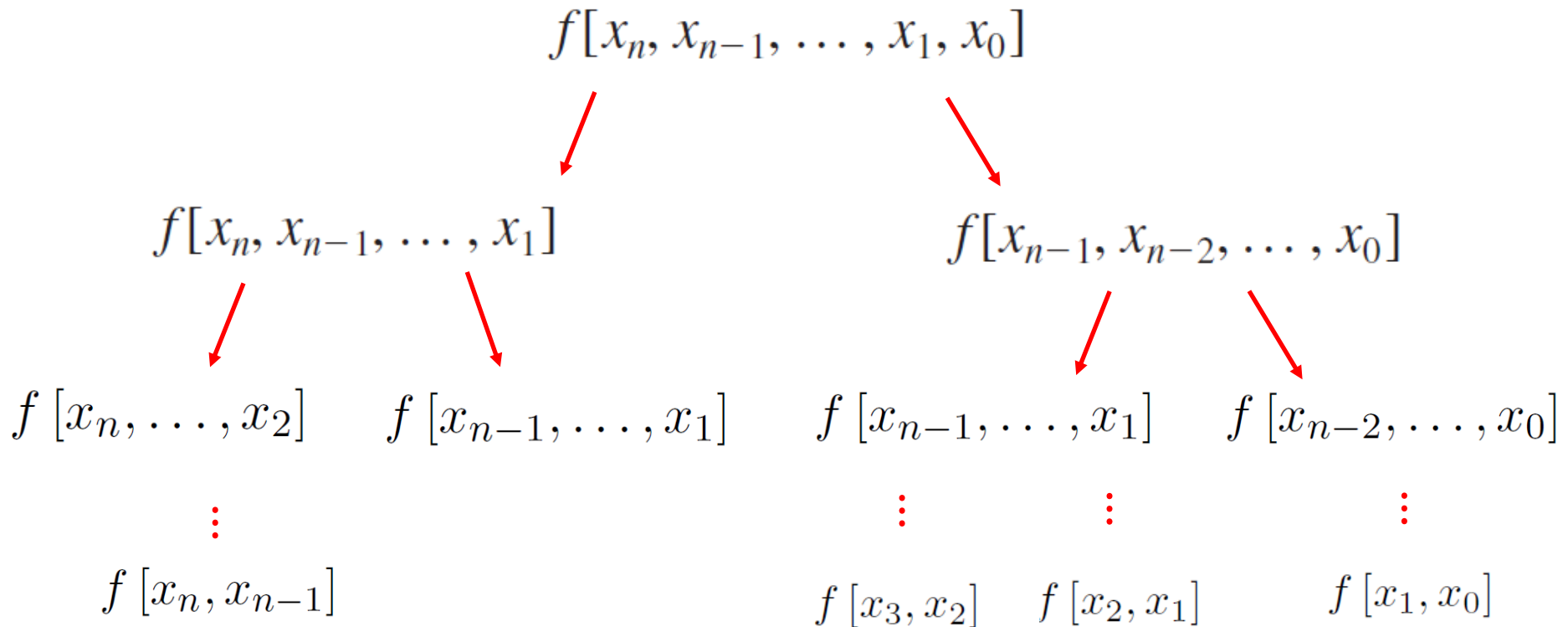
$\vdots$

$$(x_{n-1}, f(x_{n-1}))$$

$$(x_n, f(x_n))$$

Finite Divided Difference

- Start with b_n and form the following tree. Expand the tree until you reach the first finite divided differences.
- Compute the divided differences.



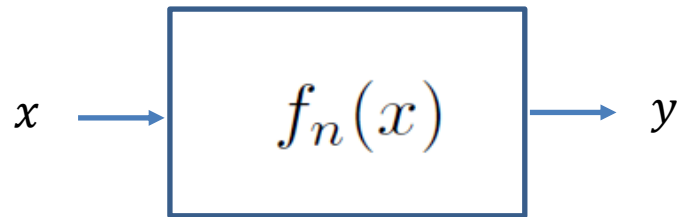
Lagrange Interpolation Method

Lagrange's Method

- The Lagrange's method fits a polynomial of order n to $n + 1$ data points.
- Lagrange's polynomial is in the following form

$$f_n(x) = L_0(x)f(x_0) + L_1(x)f(x_1) + \cdots + L_i(x)f(x_i) + \cdots + L_n(x)f(x_n)$$

- We need to compute the $L_0(x), L_1(x), \dots, L_n(x)$ using data points.



$$\begin{array}{c} (x_0, f(x_0)) \\ (x_1, f(x_1)) \\ \vdots \\ (x_{n-1}, f(x_{n-1})) \\ (x_n, f(x_n)) \end{array}$$

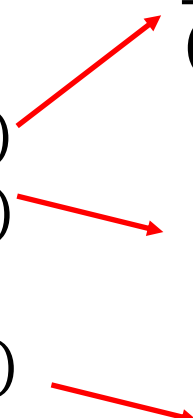
Lagrange's Method

To compute $L_0(x)$:

1. Remove the $(x_0, f(x_0))$ from the data (**first data point**).
2. Form the fractions $\frac{(x-x_j)}{(x_0-x_j)}$ for each **remaining** data point x_j .
3. Then, **multiply** all the fractions.

$$L_0(x) = \frac{(x - x_1)}{(x_0 - x_1)} \times \frac{(x - x_2)}{(x_0 - x_2)} \times \cdots \times \frac{(x - x_n)}{(x_0 - x_n)}$$

$(x_1, f(x_1))$
 $(x_2, f(x_2))$
 $\vdots$
 $(x_n, f(x_n))$



$\frac{(x - x_1)}{(x_0 - x_1)}$
 $\frac{(x - x_2)}{(x_0 - x_2)}$
 $\frac{(x - x_n)}{(x_0 - x_n)}$

Lagrange's Method

To compute $L_1(x)$:

1. Remove the $(x_1, f(x_1))$ from the data (**second data point**).
2. Form the fractions $\frac{(x-x_j)}{(x_1-x_j)}$ for each **remaining** data point x_j .
3. Then, **multiply** all the fractions.

$$L_1(x) = \frac{(x-x_0)}{(x_1-x_0)} \times \frac{(x-x_2)}{(x_1-x_2)} \times \cdots \times \frac{(x-x_n)}{(x_1-x_n)}$$

$(x_0, f(x_0))$ $\rightarrow$ $\frac{(x-x_0)}{(x_1-x_0)}$
 $(x_2, f(x_2))$ $\rightarrow$ $\frac{(x-x_2)}{(x_1-x_2)}$
 $\vdots$
 $(x_n, f(x_n))$ $\rightarrow$ $\frac{(x-x_n)}{(x_1-x_n)}$

Lagrange's Method

To compute $L_2(x)$:

1. Remove the $(x_2, f(x_2))$ from the data (**third data point**).
2. Form the fractions $\frac{(x-x_j)}{(x_2-x_j)}$ for each **remaining** data point x_j .
3. Then, **multiply** all the fractions.

$$L_2(x) = \frac{(x-x_0)}{(x_2-x_0)} \times \frac{(x-x_1)}{(x_2-x_1)} \times \cdots \times \frac{(x-x_n)}{(x_2-x_n)}$$

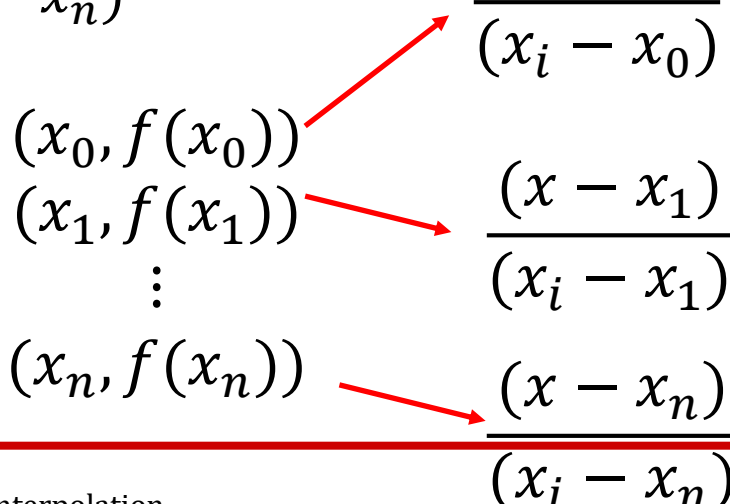
$(x_0, f(x_0))$
 $(x_1, f(x_1))$
 $\vdots$
 $(x_n, f(x_n))$

$\frac{(x-x_0)}{(x_2-x_0)}$
 $\frac{(x-x_1)}{(x_2-x_1)}$
 $\frac{(x-x_n)}{(x_2-x_n)}$

Lagrange's Method

To compute $L_i(x)$:

1. Remove the $(x_i, f(x_i))$ from the data.
2. Form the fractions $\frac{(x-x_j)}{(x_i-x_j)}$ for each remaining data point x_j .
3. Then, multiply all the fractions.

$$L_i(x) = \frac{(x-x_0)}{(x_i-x_0)} \times \frac{(x-x_1)}{(x_i-x_1)} \times \cdots \times \frac{(x-x_n)}{(x_i-x_n)}$$


$(x_0, f(x_0))$
 $(x_1, f(x_1))$
 $\vdots$
 $(x_n, f(x_n))$

$\frac{(x-x_0)}{(x_i-x_0)}$
 $\frac{(x-x_1)}{(x_i-x_1)}$
 $\frac{(x-x_n)}{(x_i-x_n)}$

Lagrange's Method

To compute $L_n(x)$:

1. Remove the $(x_n, f(x_n))$ from the data.
2. Form the fractions $\frac{(x-x_j)}{(x_n-x_j)}$ for each remaining data point x_j .
3. Then, multiply all the fractions.

$$L_n(x) = \frac{(x-x_0)}{(x_n-x_0)} \times \frac{(x-x_1)}{(x_n-x_1)} \times \cdots \times \frac{(x-x_{n-1})}{(x_n-x_{n-1})}$$

$(x_0, f(x_0))$
 $(x_1, f(x_1))$
 $\vdots$
 $(x_{n-1}, f(x_{n-1}))$

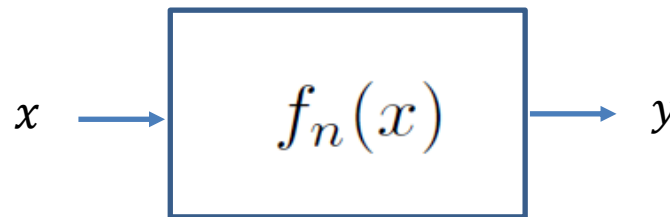
$\frac{(x-x_0)}{(x_n-x_0)}$
 $\frac{(x-x_1)}{(x_n-x_1)}$
 $\frac{(x-x_{n-1})}{(x_n-x_{n-1})}$

52

Example

- Find the Lagrange's interpolation model for the data below.

(1,2)
(2,4)
(3,5)



Example

- $x_0 = 1, x_1 = 2, x_2 = 3$
- $f(x_0) = 2, f(x_1) = 4, f(x_2) = 5$

$(1,2)$	$(x_0, f(x_0))$
$(2,4)$	$(x_1, f(x_1))$
$(3,5)$	$(x_2, f(x_2))$

Example

- $x_0 = 1, x_1 = 2, x_2 = 3, f(x_0) = 2, f(x_1) = 4, f(x_2) = 5$

$$f_2(x) = L_0(x)f(x_0) + L_1(x)f(x_1) + L_2(x)f(x_2)$$

$$L_0(x) = \frac{(x-x_1)}{(x_0-x_1)} \times \frac{(x-x_2)}{(x_0-x_2)} = \frac{(x-2)}{(1-2)} \times \frac{(x-3)}{(1-3)} \quad \begin{matrix} (1,2) \\ (2,4) \\ (3,5) \end{matrix}$$

$$L_1(x) = \frac{(x-x_0)}{(x_1-x_0)} \times \frac{(x-x_2)}{(x_1-x_2)} = \frac{(x-1)}{(2-1)} \times \frac{(x-3)}{(2-3)}$$

$$L_2(x) = \frac{(x-x_0)}{(x_2-x_0)} \times \frac{(x-x_1)}{(x_2-x_1)} = \frac{(x-1)}{(3-1)} \times \frac{(x-2)}{(3-2)} \quad \begin{matrix} (x_0, f(x_0)) \\ (x_1, f(x_1)) \\ (x_2, f(x_2)) \end{matrix}$$

Example

$$f_2(x) = 2 \frac{(x-2)}{(1-2)} \times \frac{(x-3)}{(1-3)} + 4 \frac{(x-1)}{(2-1)} \times \frac{(x-3)}{(2-3)} + 5 \frac{(x-1)}{(3-1)} \times \frac{(x-2)}{(3-2)}$$

(1,2)

(2,4)

(3,5)

General Polynomial Interpolation

General Polynomial Interpolation

- The Polynomials in **Lagrange's and Newton's methods** have special structures.
- Sometimes, we may need to fit a polynomial of the following form to our data points.

$$f_n(x) = a_0 + a_1x + a_2x^2 + \cdots + a_nx^n$$

- This is not in the form of **Lagrange's and Newton's polynomials**.

$$f_n(x) = L_0(x)f(x_0) + L_1(x)f(x_1) + \cdots + L_i(x)f(x_i) + \cdots + L_n(x)f(x_n)$$

$$f_n(x) = b_0 + b_1(x - x_0) + b_2(x - x_0)(x - x_1) + \cdots + b_n(x - x_0)(x - x_1) \cdots (x - x_{n-1})$$

General Polynomial Interpolation

- Suppose that we have $n+1$ data points.

$$\begin{aligned}(x_0, f(x_0)) \\ (x_1, f(x_1)) \\ \vdots \\ (x_n, f(x_n))\end{aligned}$$

- To fit the following polynomial to our data points, we need to find $a_0, a_1, \dots, a_n$

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

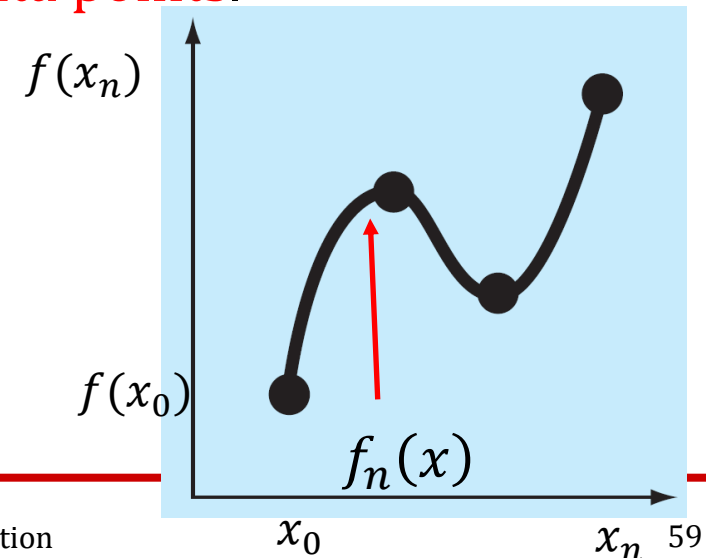
- The **polynomial should pass through all data points.**

$$f_n(x_0) = f(x_0)$$

$$f_n(x_1) = f(x_1)$$

$$\vdots$$

$$f_n(x_n) = f(x_n)$$



General Polynomial Interpolation

- We obtain a system of linear equations with $n + 1$ unknowns $a_0, a_1, \dots, a_n$.
- $a_0, a_1, \dots, a_n$ are obtained by solving this system of linear equations.

$$f_n(x_0) = a_0 + a_1x_0 + a_2x_0^2 + \dots + a_nx_0^n = f(x_0)$$

$$f_n(x_1) = a_0 + a_1x_1 + a_2x_1^2 + \dots + a_nx_1^n = f(x_1)$$

$$\vdots$$

$$f_n(x_n) = a_0 + a_1x_n + a_2x_n^2 + \dots + a_nx_n^n = f(x_n)$$

$$(x_0, f(x_0))$$

$$(x_1, f(x_1))$$

$$\vdots$$

$$(x_n, f(x_n))$$

Example

- Fit the polynomial $a_0 + a_1x + a_2x^2$ to the following data points.

(300,0.616)

(400,0.525)

(500,0.457)

Example

- Polynomial $a_0 + a_1x + a_2x^2$

$(300, 0.616)$	$(x_0, f(x_0))$
$(400, 0.525)$	$(x_1, f(x_1))$
$(500, 0.457)$	$(x_2, f(x_2))$

$$x_0 = 300, f(x_0) = 0.616$$

$$x_1 = 400, f(x_1) = 0.525$$

$$x_2 = 500, f(x_2) = 0.457$$

Example

- We form the following system of linear equations.

$$0.616 = a_0 + a_1(300) + a_2(300)^2$$

$$0.525 = a_0 + a_1(400) + a_2(400)^2$$

$$0.457 = a_0 + a_1(500) + a_2(500)^2$$

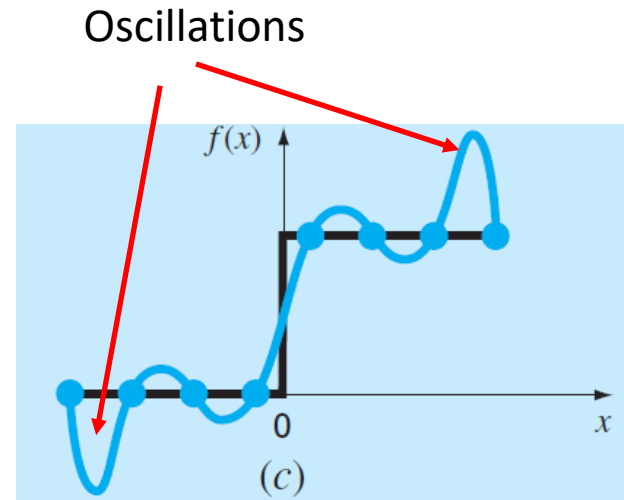
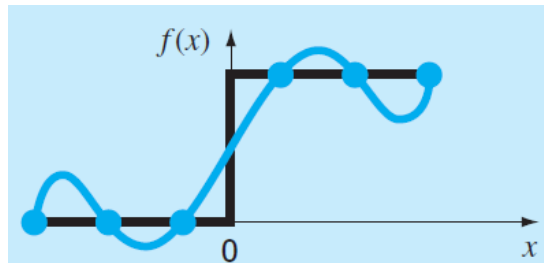
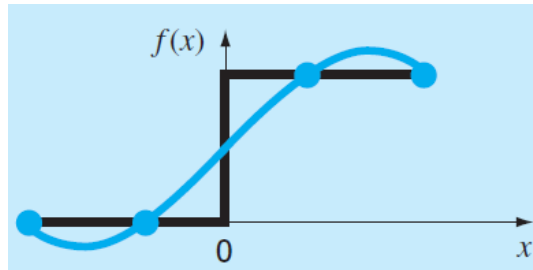
- a_0, a_1, a_2 are obtained by solving this system of linear equations.

$$\begin{bmatrix} 1 & 300 & 90,000 \\ 1 & 400 & 160,000 \\ 1 & 500 & 250,000 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} 0.616 \\ 0.525 \\ 0.457 \end{bmatrix} \quad \begin{aligned} a_0 &= 1.027 \\ a_1 &= -0.001715 \\ a_2 &= 0.00000115 \end{aligned}$$

Spline Interpolation

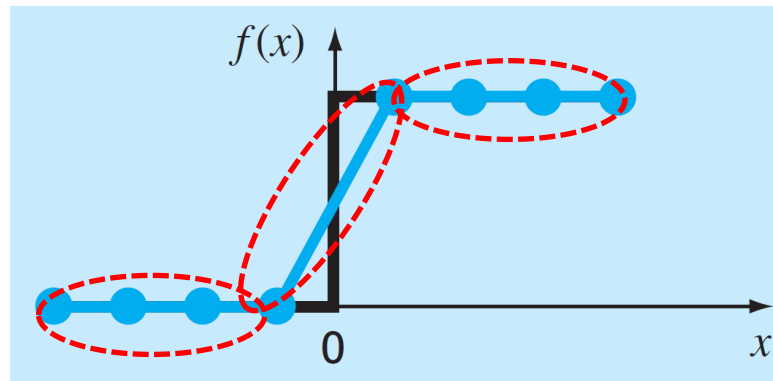
Spline Interpolation

- In **Newton's** and **Lagrange's** methods, **one polynomial** is fitted to all data points.
- In certain cases, the resulting polynomial might oscillate and introduce large errors.
- For example, when the data points have abrupt changes.



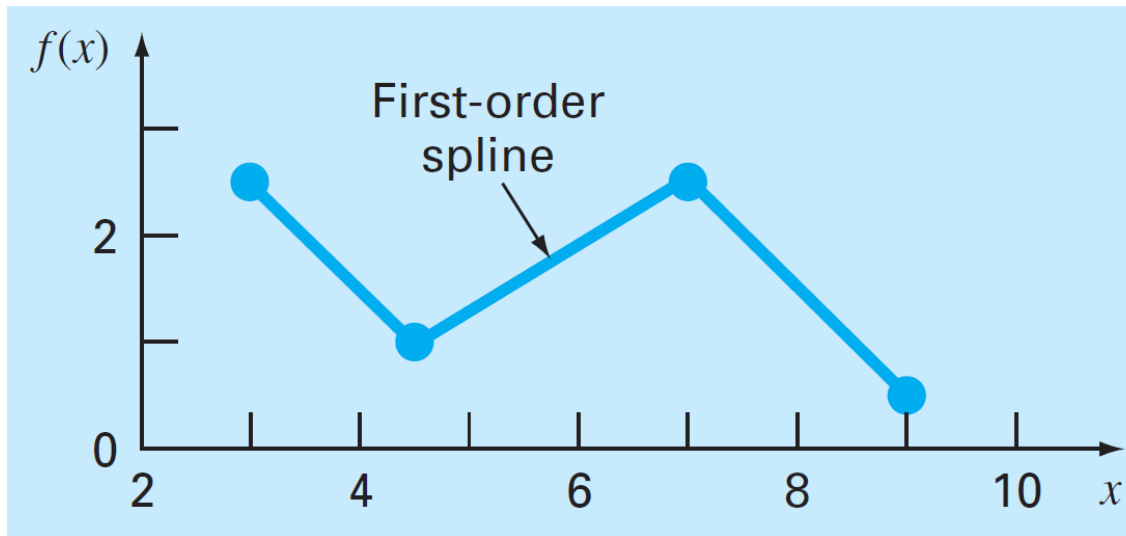
Spline Interpolation

- In such cases, the spline interpolation method can be used.
- In this method, the data points are divided into several groups.
- A **polynomial** is fitted to **each group** of the data points.



Linear Splines

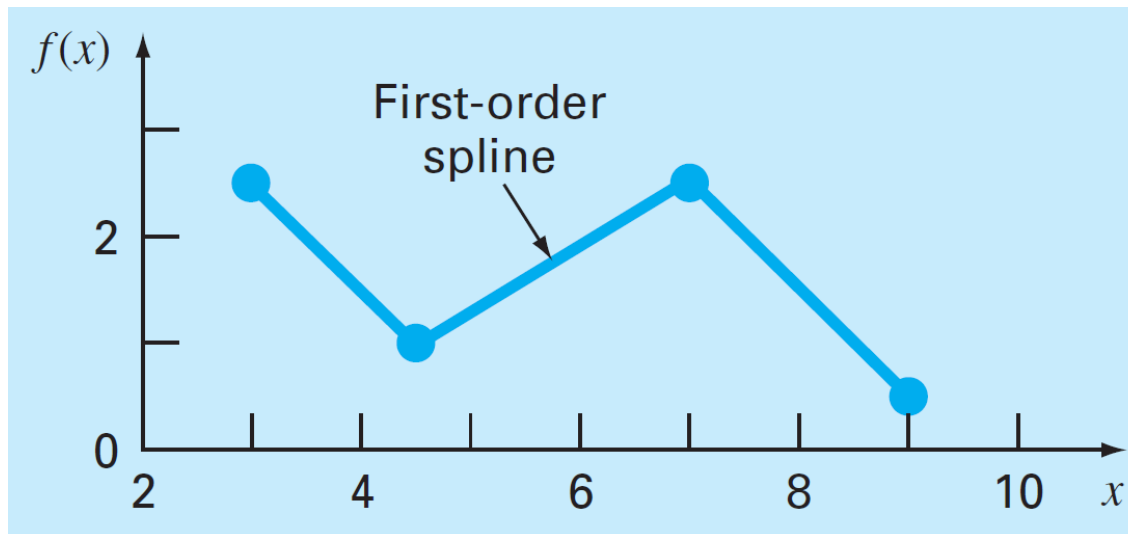
- In the linear spline method, the data points are connected by lines.
- We need to order our data points according to the increasing order.
- That is, $x_0 < x_1 < x_2 < \dots < x_n$.



$$\begin{aligned} &(x_0, f(x_0)) \\ &(x_1, f(x_1)) \\ &\vdots \\ &(x_n, f(x_n)) \end{aligned}$$

Linear Splines

- First, arrange the data points in the increasing order, i.e., $x_0 < x_1 < x_2 < \dots < x_n$.
- For every **two consecutive** data points, find the line that connects them.



$$\begin{aligned} & (x_0, f(x_0)) \\ & (x_1, f(x_1)) \\ & (x_2, f(x_2)) \\ & \vdots \\ & (x_{n-1}, f(x_{n-1})) \\ & (x_n, f(x_n)) \end{aligned}$$

Linear Splines

- The line that connects $(x_0, f(x_0))$ to $(x_1, f(x_1))$ is

$$f(x) = f(x_0) + m_0(x - x_0) \quad x_0 \leq x \leq x_1 \quad m_0 = \frac{f(x_1) - f(x_0)}{x_1 - x_0}$$

- The line that connects $(x_1, f(x_1))$ to $(x_2, f(x_2))$ is

$$f(x) = f(x_1) + m_1(x - x_1) \quad x_1 \leq x \leq x_2 \quad m_1 = \frac{f(x_2) - f(x_1)}{x_2 - x_1}$$

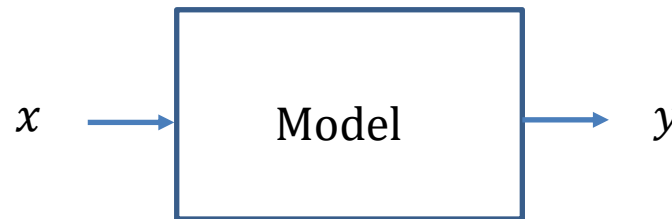
- The line that connects $(x_{n-1}, f(x_{n-1}))$ to $(x_n, f(x_n))$ is

$$f(x) = f(x_{n-1}) + m_{n-1}(x - x_{n-1}) \quad x_{n-1} \leq x \leq x_n \quad m_{n-1} = \frac{f(x_n) - f(x_{n-1})}{x_n - x_{n-1}}$$

Example

- Fit a first-order spline to the data in the table below. Evaluate your model at $x = 5$.

x	$f(x)$
3.0	2.5
4.5	1.0
7.0	2.5
9.0	0.5



Example

- $x_0 = 3, x_1 = 4.5, x_2 = 7, x_3 = 9.$
- $f(x_0) = 2.5, f(x_1) = 1, f(x_2) = 2.5, f(x_3) = 0.5.$

x	$f(x)$	
3.0	2.5	$(x_0, f(x_0))$
4.5	1.0	$(x_1, f(x_1))$
7.0	2.5	$(x_2, f(x_2))$
9.0	0.5	$(x_3, f(x_3))$

Example

- $x_0 = 3, x_1 = 4.5, x_2 = 7, x_3 = 9.$

$$f(x) = 2.5 - (x - 3), \quad 3 \leq x \leq 4.5,$$

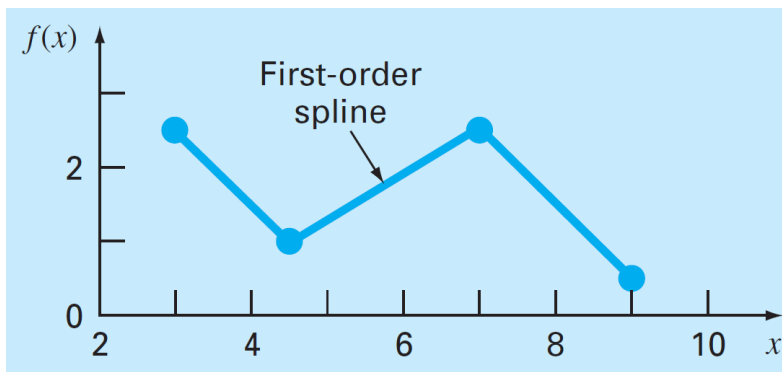
$$f(x) = 1 + 0.6(x - 4.5) \quad 4.5 \leq x \leq 7,$$

$$f(x) = 2.5 - (x - 7), \quad 7 \leq x \leq 9,$$

$$m_0 = -1$$

$$m_1 = 0.6$$

$$m_2 = -1$$

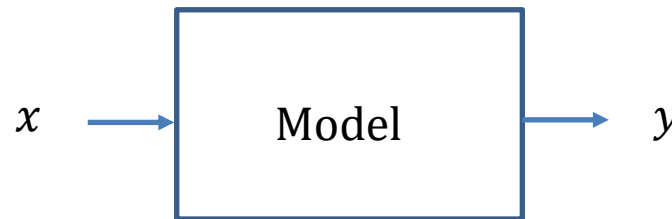


x	$f(x)$
3.0	2.5
4.5	1.0
7.0	2.5
9.0	0.5

Example

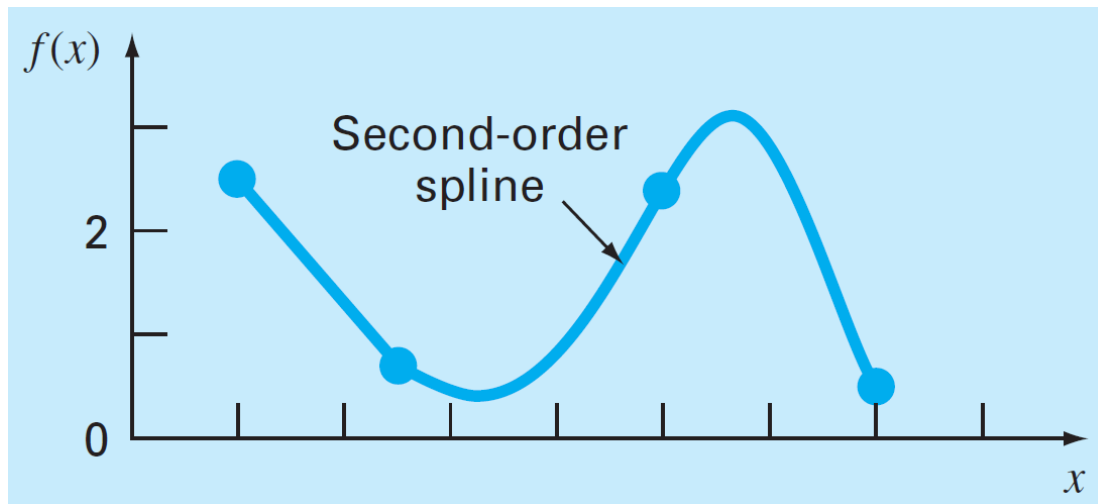
- $x = 5$ is located between 4.5 and 7. So, the output of the model at $x = 5$ is 1.3.

$$f(x) = 1 + 0.6(x - 4.5) \quad 4.5 \leq x \leq 7, \quad m_1 = 0.6$$



Quadratic Splines

- In the quadratic spline method, data points are connected by quadratic functions.
- First, arrange the data points according to the increasing order, $x_0 < x_1 < x_2 < \dots < x_n$.



$$\begin{aligned} & (x_0, f(x_0)) \\ & (x_1, f(x_1)) \\ & (x_2, f(x_2)) \\ & \vdots \\ & (x_{n-1}, f(x_{n-1})) \\ & (x_n, f(x_n)) \end{aligned}$$

Quadratic Splines

- The first polynomial connects $f(x_0)$ and $f(x_1)$.
- The second polynomial connects $f(x_2)$ and $f(x_3)$.
- The last polynomial connects $f(x_{n-1})$ and $f(x_n)$.

$$f(x) = a_1x^2 + b_1x + c_1, \quad x_0 \leq x \leq x_1$$

$$f(x) = a_2x^2 + b_2x + c_2, \quad x_1 \leq x \leq x_2$$

$$\vdots$$

$$f(x) = a_nx^2 + b_nx + c_n, \quad x_{n-1} \leq x \leq x_n$$

$$(x_0, f(x_0))$$

$$(x_1, f(x_1))$$

$$(x_2, f(x_2))$$

$$\vdots$$

$$(x_{n-1}, f(x_{n-1}))$$

$$(x_n, f(x_n))$$

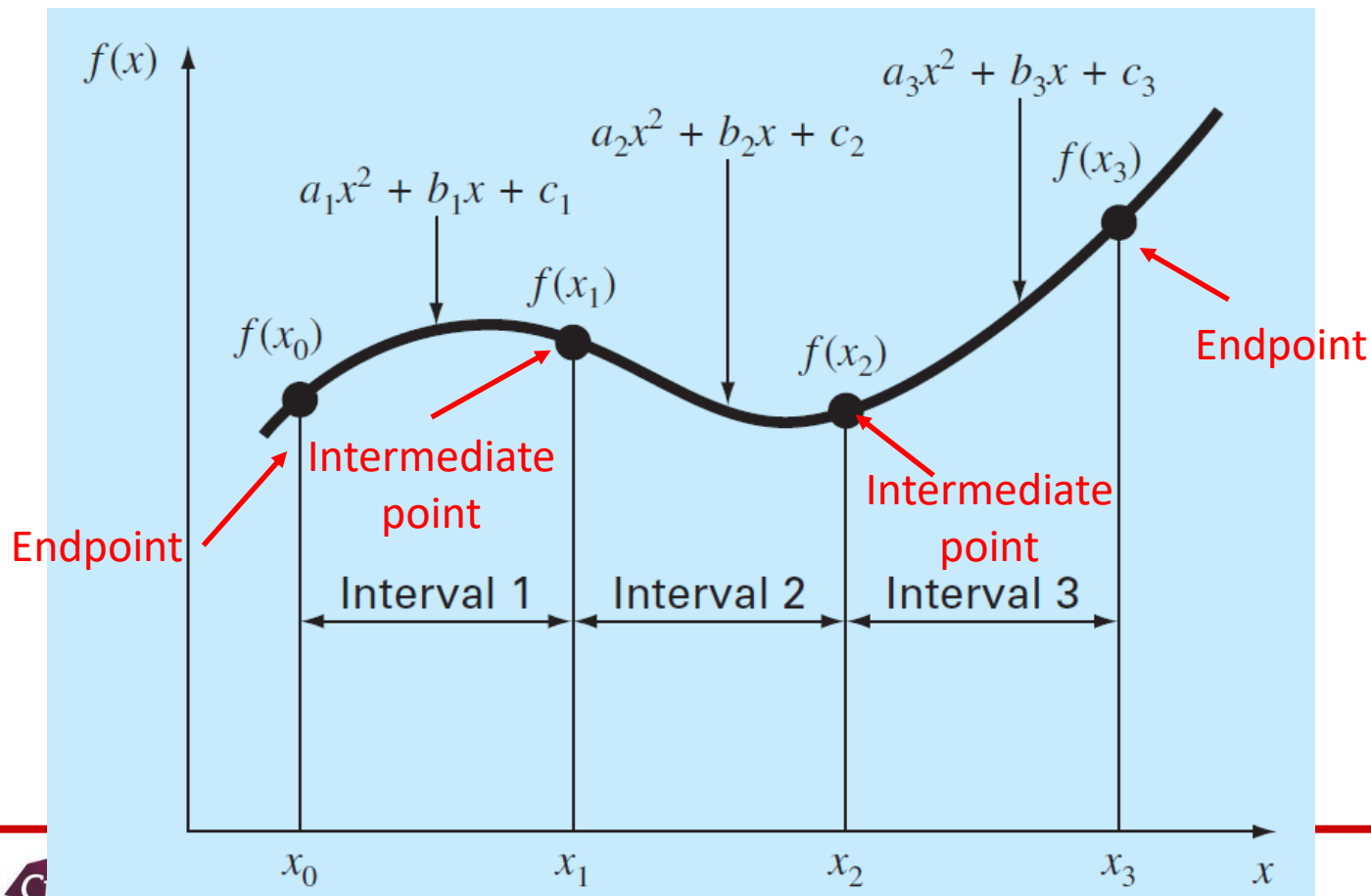
Quadratic Splines

- For $n + 1$ data points, we need to find the parameters of n quadratic functions.
- We will form a system of linear equations. The parameters will be obtained by solving the system of linear equations.

$$\begin{array}{lll} f(x) = a_1x^2 + b_1x + c_1, & x_0 \leq x \leq x_1 & (x_0, f(x_0)) \\ f(x) = a_2x^2 + b_2x + c_2, & x_1 \leq x \leq x_2 & (x_1, f(x_1)) \\ \vdots & & \\ f(x) = a_nx^2 + b_nx + c_n, & x_{n-1} \leq x \leq x_n & (x_n, f(x_n)) \end{array}$$

Quadratic Splines

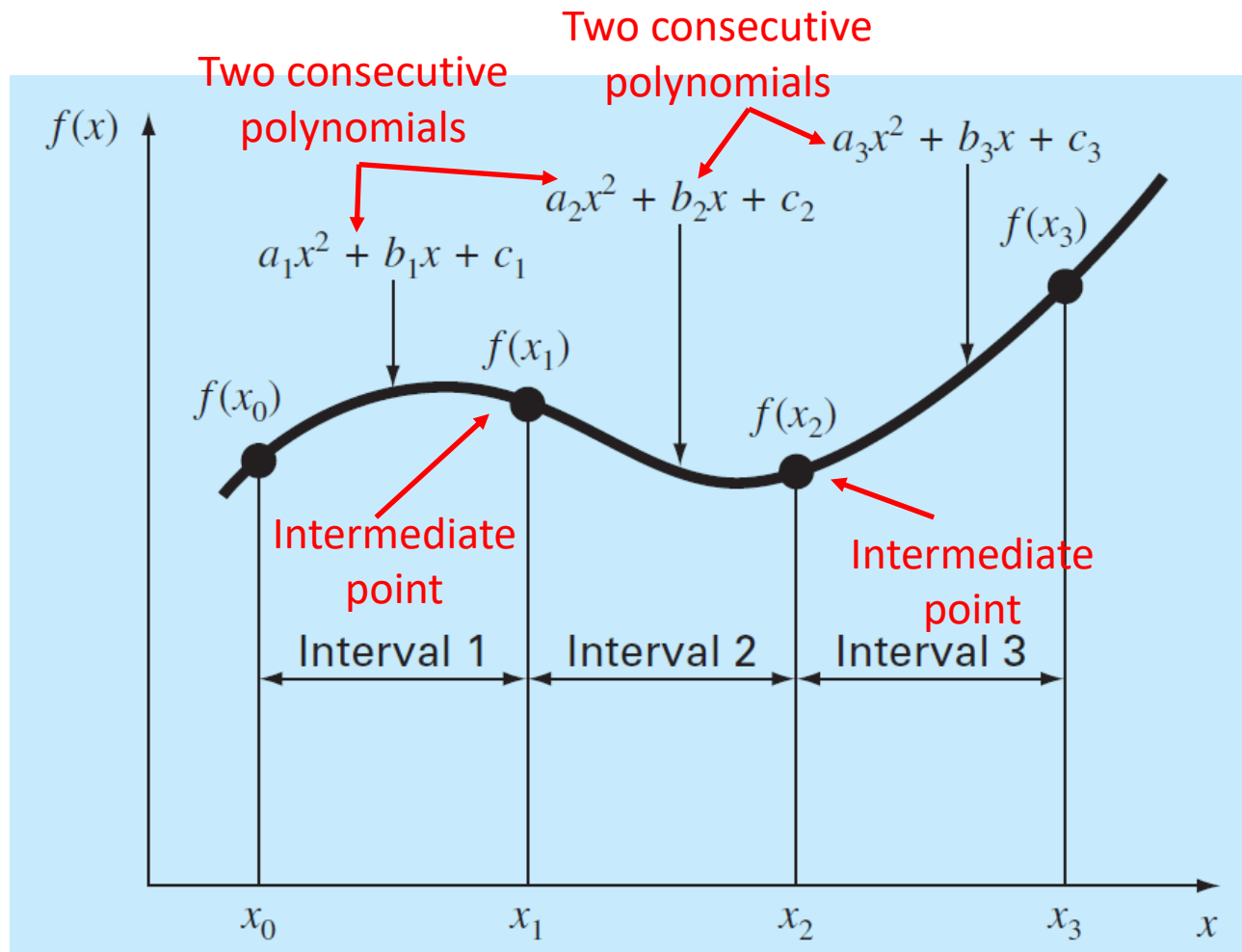
- We have **two endpoints**. x_0 and x_n are endpoints.
- $x_1, \dots, x_{n-1}$ are **intermediate points**. We have $n - 1$ intermediate points.



$$\begin{aligned} & (x_0, f(x_0)) \\ & (x_1, f(x_1)) \\ & (x_2, f(x_2)) \\ & \vdots \\ & (x_{n-1}, f(x_{n-1})) \\ & (x_n, f(x_n)) \end{aligned}$$

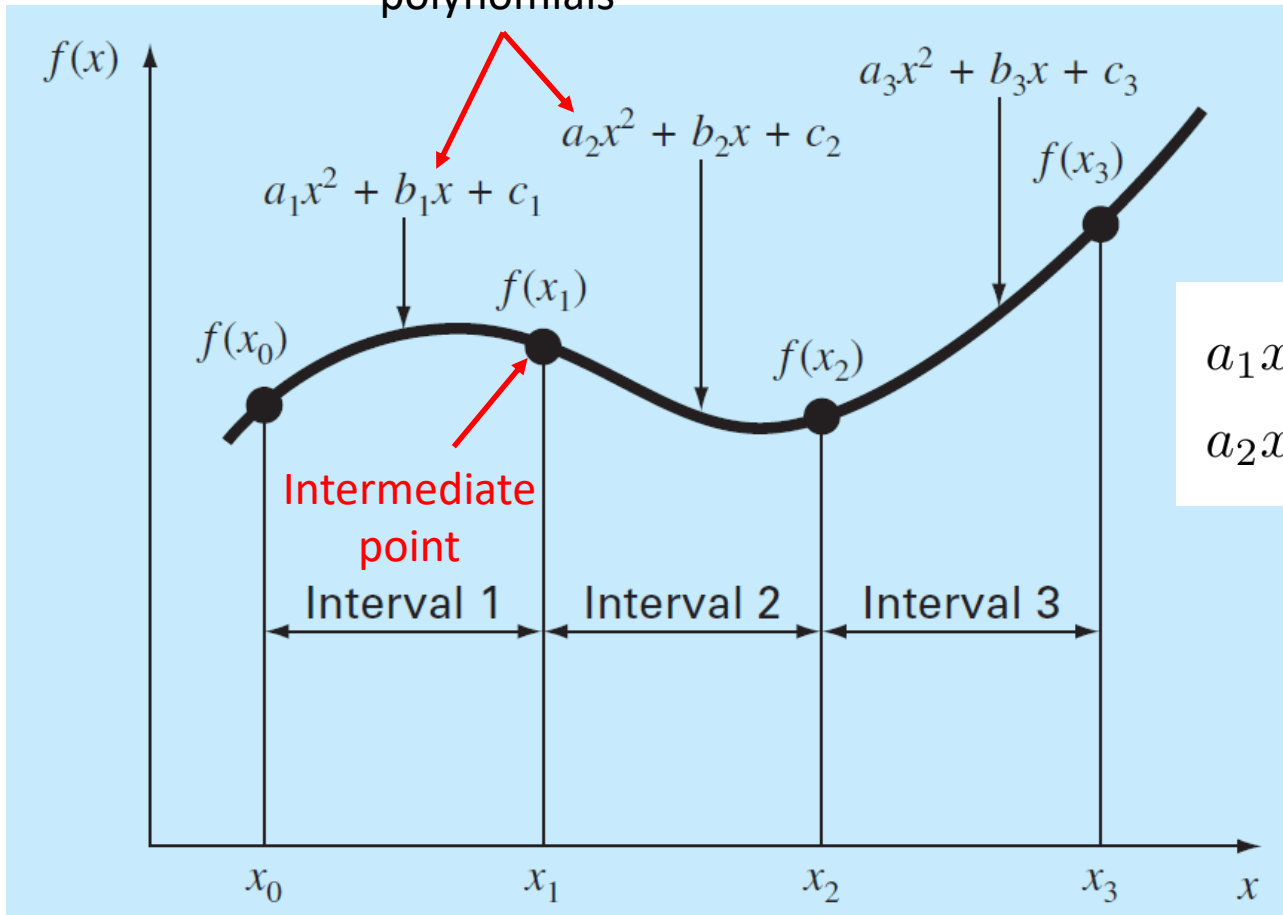
Quadratic Splines

- Two consecutive polynomials should pass through their intermediate point.



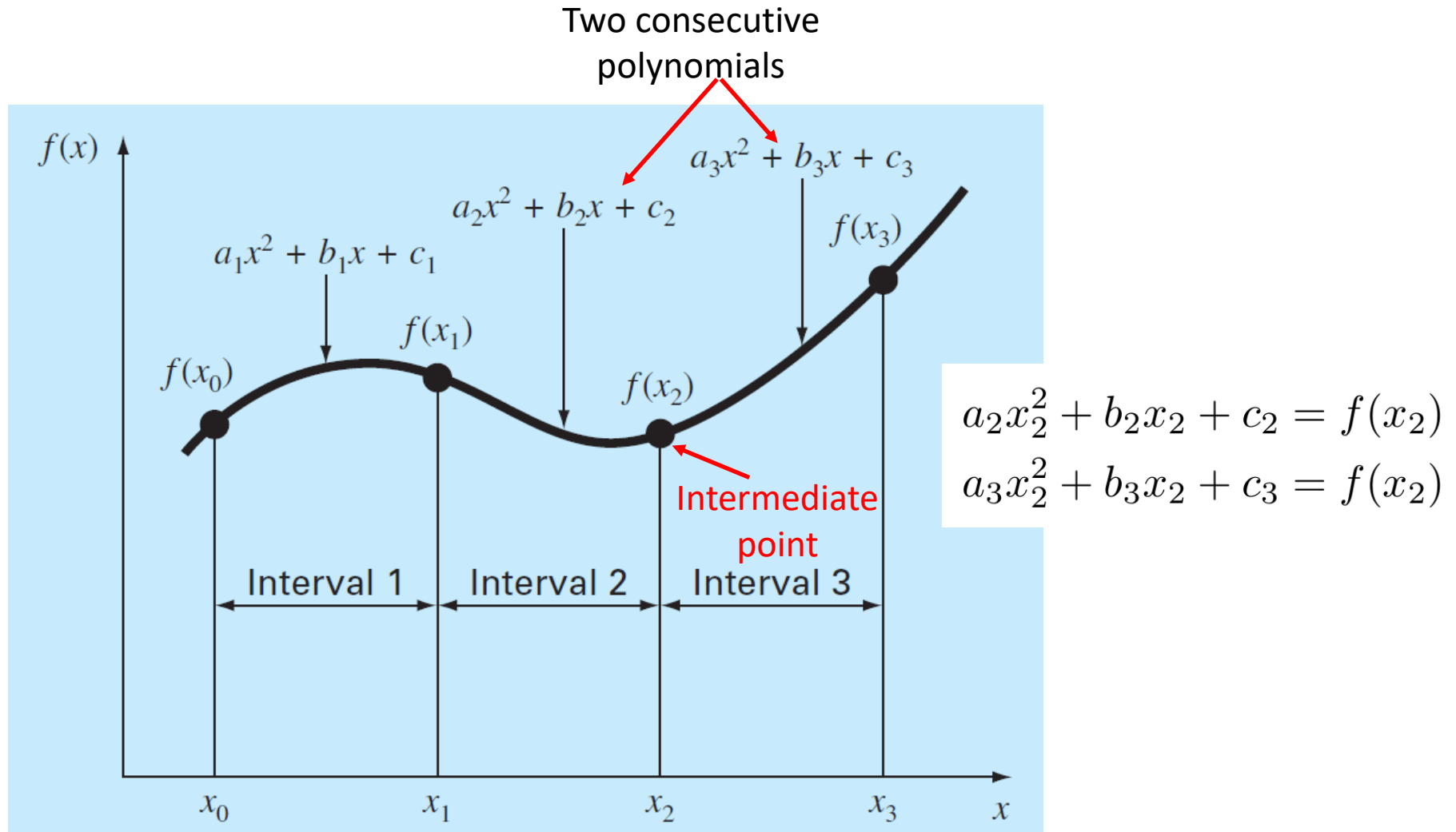
Quadratic Splines

Two consecutive
polynomials



$$a_1x_1^2 + b_1x_1 + c_1 = f(x_1)$$
$$a_2x_1^2 + b_2x_1 + c_2 = f(x_1)$$

Quadratic Splines



Quadratic Splines

- Two consecutive polynomials should pass through their intermediate point.
- We obtain $2n - 2$ linear equations.

$$f(x_1) = a_1x_1^2 + b_1x_1 + c_1$$

$$f(x_1) = a_2x_1^2 + b_2x_1 + c_2$$

$$f(x_2) = a_2x_2^2 + b_2x_2 + c_2$$

$$f(x_2) = a_3x_2^2 + b_3x_2 + c_3$$

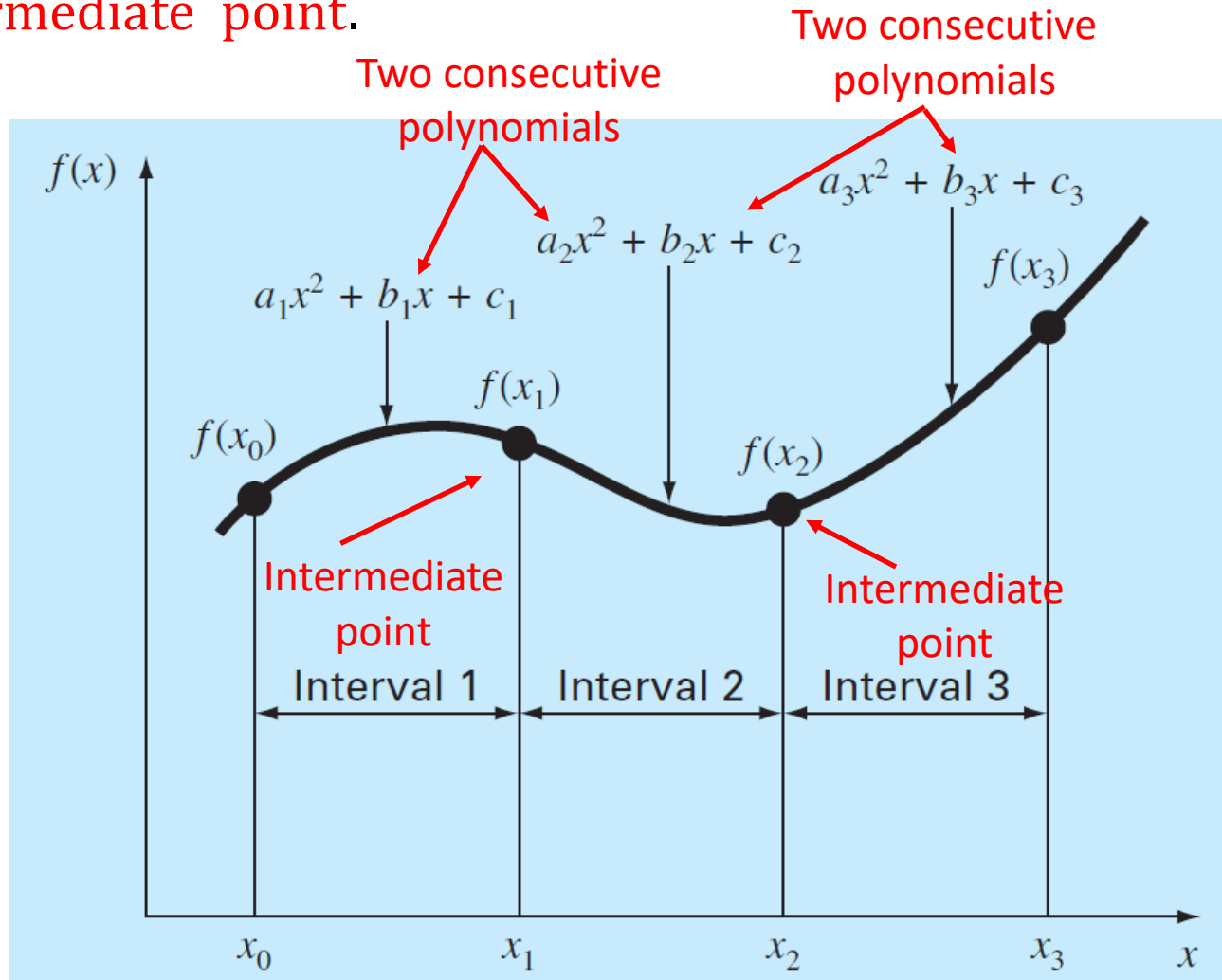
$\vdots$

$$f(x_{n-1}) = a_{n-1}x_{n-1}^2 + b_{n-1}x_{n-1} + c_{n-1}$$

$$f(x_{n-1}) = a_nx_n^2 + b_nx_{n-1} + c_n$$

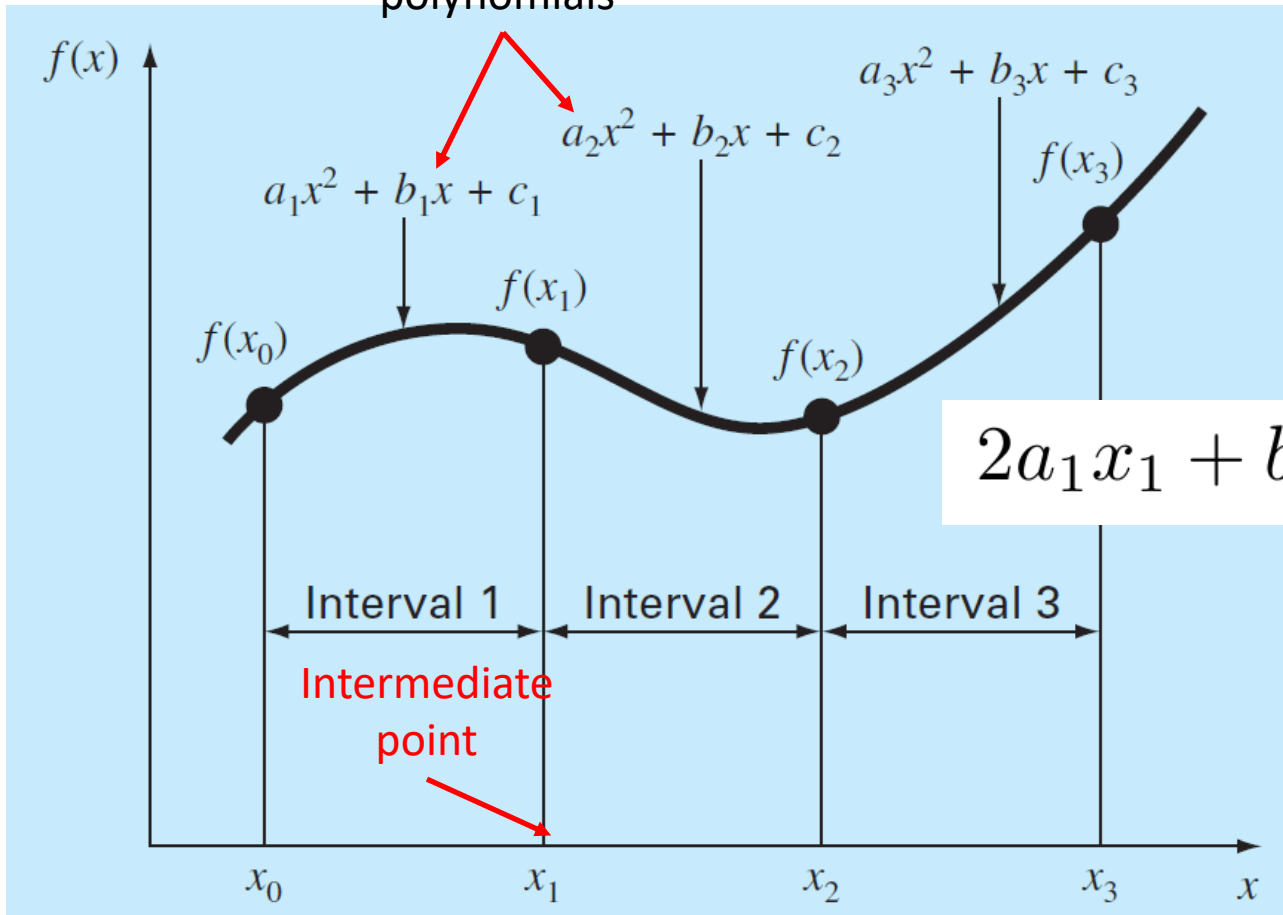
Quadratic Splines

- The **first derivative** of **consecutive polynomials** should be the same at their **intermediate point**.



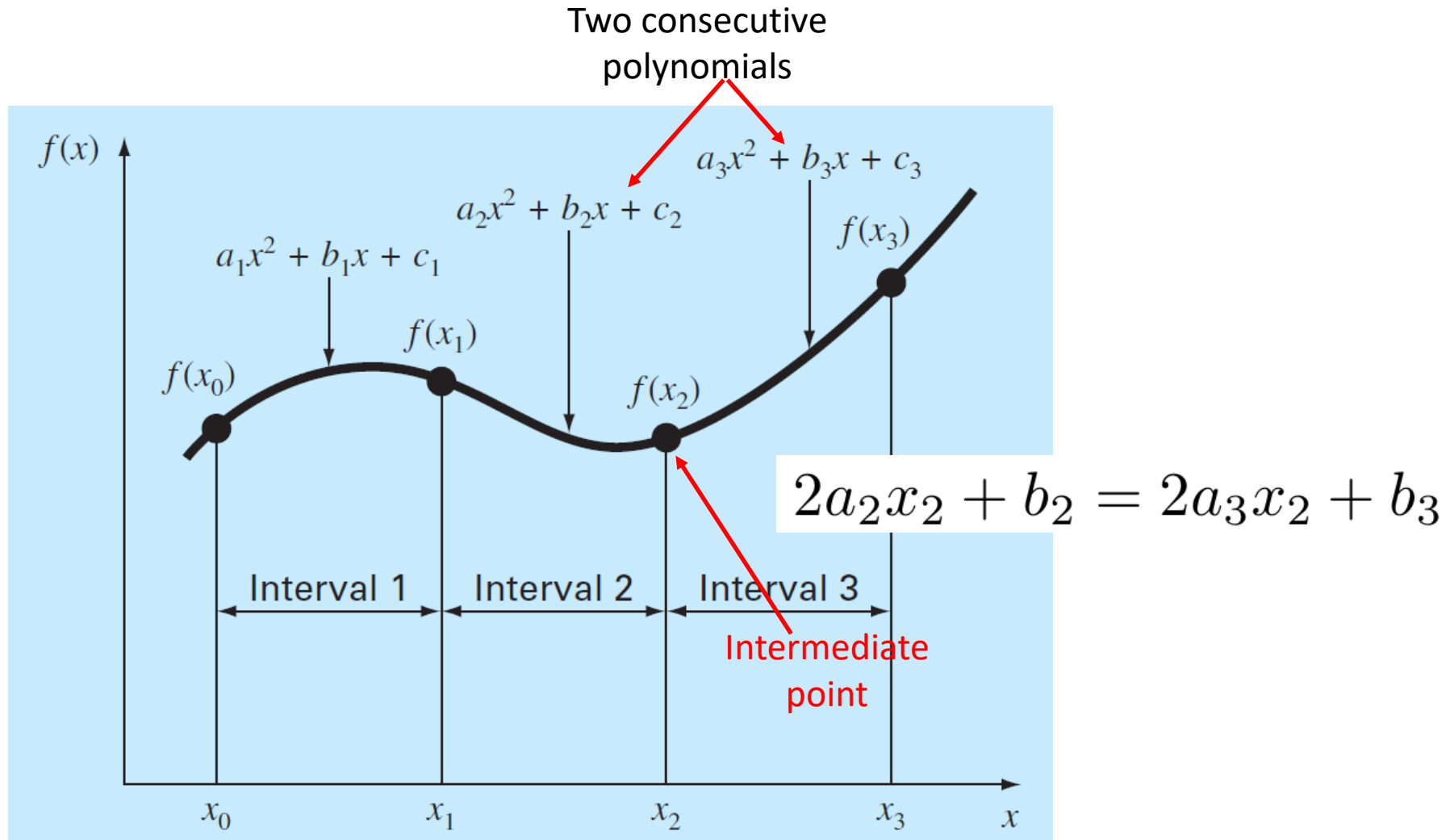
Quadratic Splines

Two consecutive
polynomials



$$2a_1x_1 + b_1 = 2a_2x_1 + b_2$$

Quadratic Splines



Quadratic Splines

- The first derivative of consecutive polynomials should be the same at their intermediate point.
- We obtain $n - 1$ linear equations.

$$2a_1x_1 + b_1 = 2a_2x_1 + b_2$$

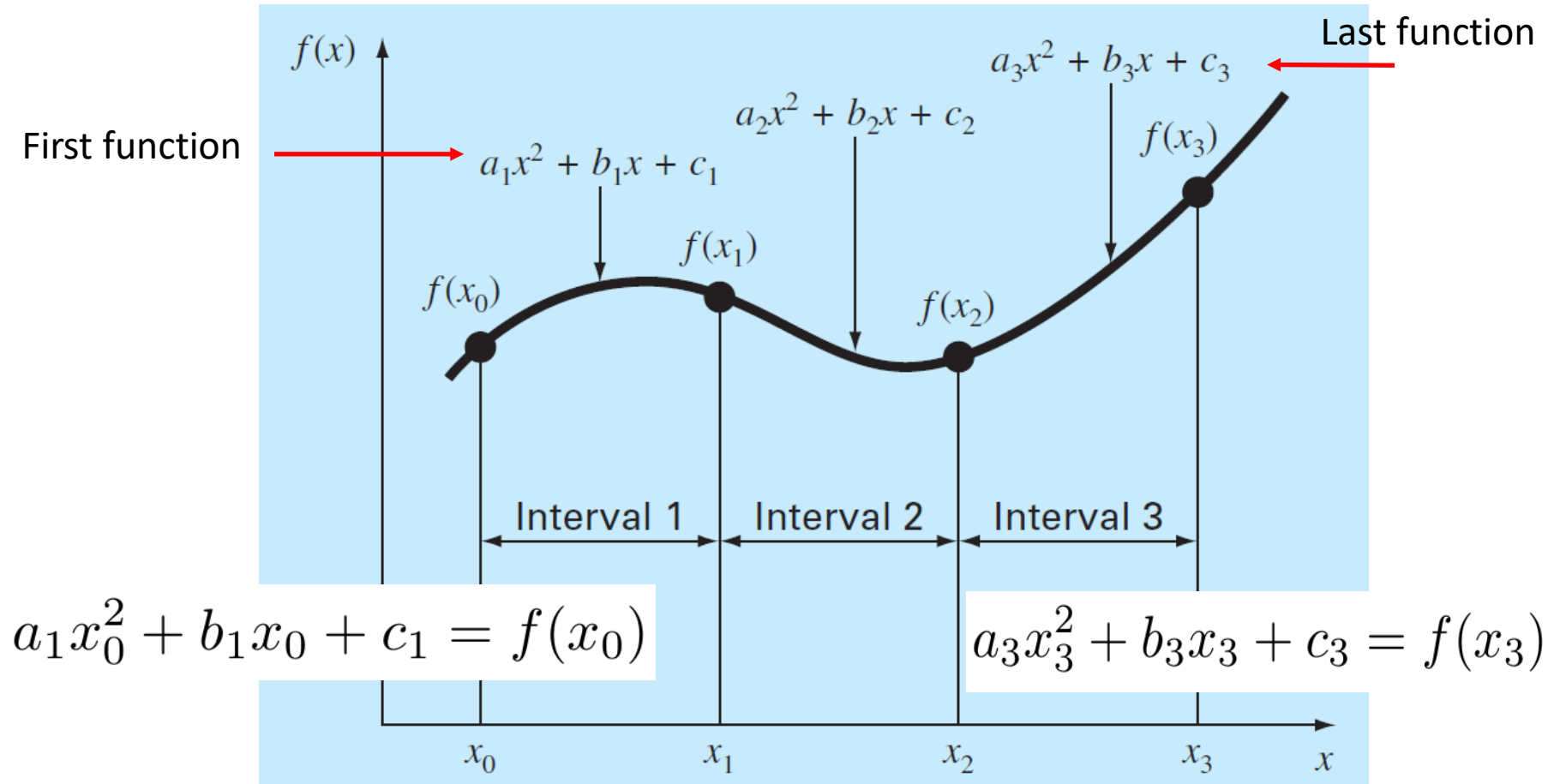
$$2a_2x_2 + b_2 = 2a_3x_2 + b_3$$

$$\vdots$$

$$2a_{n-1}x_{n-1} + b_{n-1} = 2a_nx_{n-1} + b_n$$

Quadratic Splines

- The first and last polynomial must pass through the end points.



Quadratic Splines

- The first and last polynomial must pass through the endpoints.
- We obtain 2 linear equations.

$$a_1x_0^2 + b_1x_0 + c_1 = f(x_0)$$
$$a_nx_n^2 + b_nx_n + c_n = f(x_n)$$

Quadratic Splines

- The second derivative at the first point should be zero.
- We obtain $a_1 = 0$.
- We need to find $a_1, \dots, a_n, b_1, \dots, b_n, c_1, \dots, c_n$.

$$\begin{aligned} f(x) &= a_1 x^2 + b_1 x + c_1, & x_0 \leq x \leq x_1 \\ f(x) &= a_2 x^2 + b_2 x + c_2, & x_1 \leq x \leq x_2 \\ &\vdots \\ f(x) &= a_n x^2 + b_n x + c_n, & x_{n-1} \leq x \leq x_n \end{aligned}$$

Quadratic Splines

- The parameters $a_2, \dots, a_n, b_1, \dots, b_n, c_1, \dots, c_n$ are the solutions of the following system of linear equations.
- Remember that $a_1 = 0$.

$$f(x_1) = a_1x_1^2 + b_1x_1 + c_1$$

$$f(x_1) = a_2x_1^2 + b_2x_1 + c_2$$

$$f(x_2) = a_2x_2^2 + b_2x_2 + c_2$$

$$f(x_2) = a_3x_2^2 + b_3x_2 + c_3$$

$$\vdots$$

$$f(x_{n-1}) = a_{n-1}x_{n-1}^2 + b_{n-1}x_{n-1} + c_{n-1}$$

$$f(x_{n-1}) = a_nx_n^2 + b_nx_{n-1} + c_n$$

$$f(x_0) = a_1x_0^2 + b_1x_0 + c_1$$

$$f(x_n) = a_nx_n^2 + b_nx_n + c_n$$

$$2a_1x_1 + b_1 = 2a_2x_1 + b_2$$

$$2a_2x_2 + b_2 = 2a_3x_2 + b_3$$

$$\vdots$$

$$2a_{n-1}x_{n-1} + b_{n-1} = 2a_nx_{n-1} + b_n$$

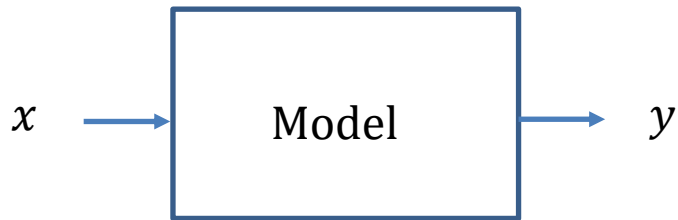
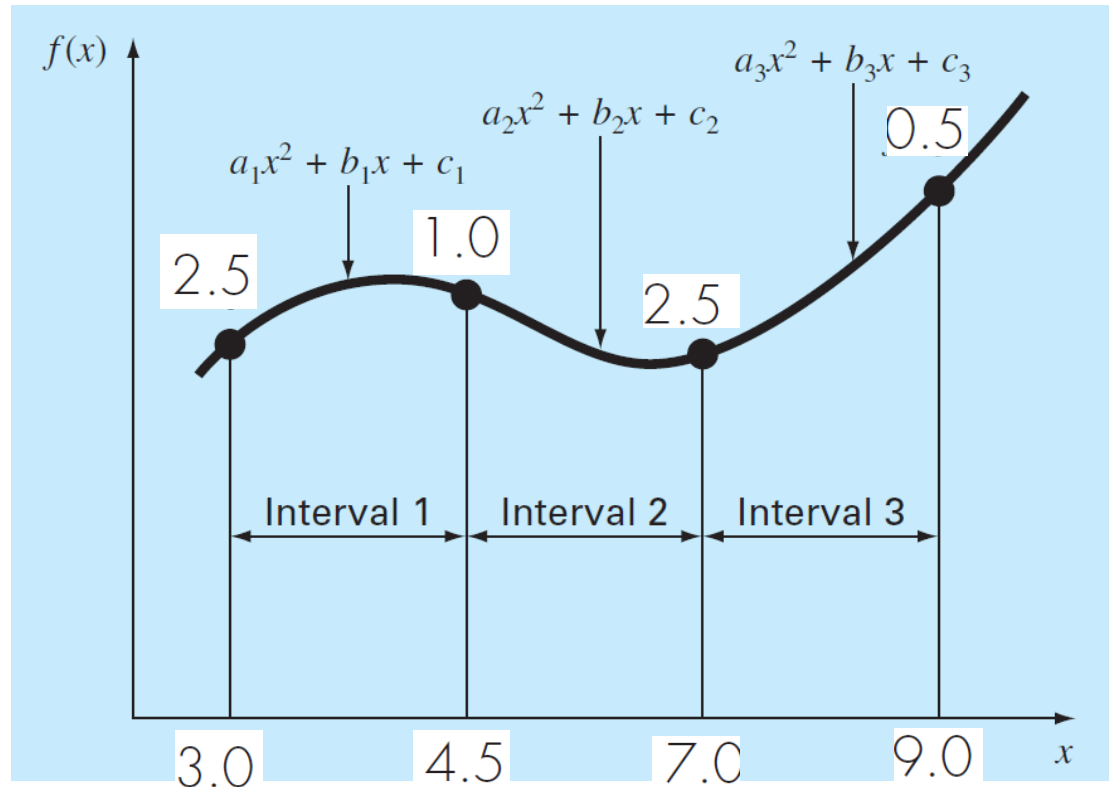
Quadratic Splines

- We form a system of linear equations with $3n$ unknowns and $3n$ equations.
 - Parameters are obtained by solving the system of linear equations.
1. The consecutive polynomials must pass through their intermediate points ($2n - 2$ conditions).
 2. The first and last polynomial must pass through the endpoints (2 conditions).
 3. The first derivatives of consecutive polynomials at their intermediate points must be equal ($n - 1$ conditions).
 4. $a_1 = 0$.

Example

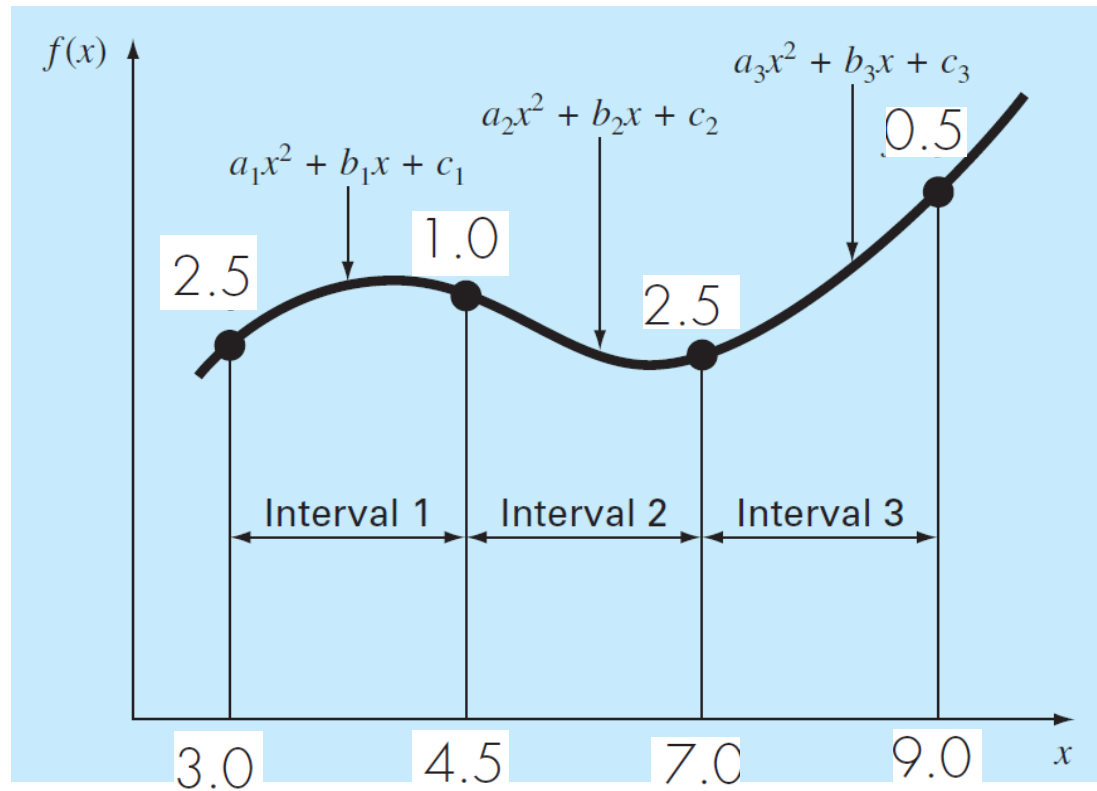
- Fit a quadratic spline to the data below. Evaluate your model at $x = 5$.

x	$f(x)$
3.0	2.5
4.5	1.0
7.0	2.5
9.0	0.5



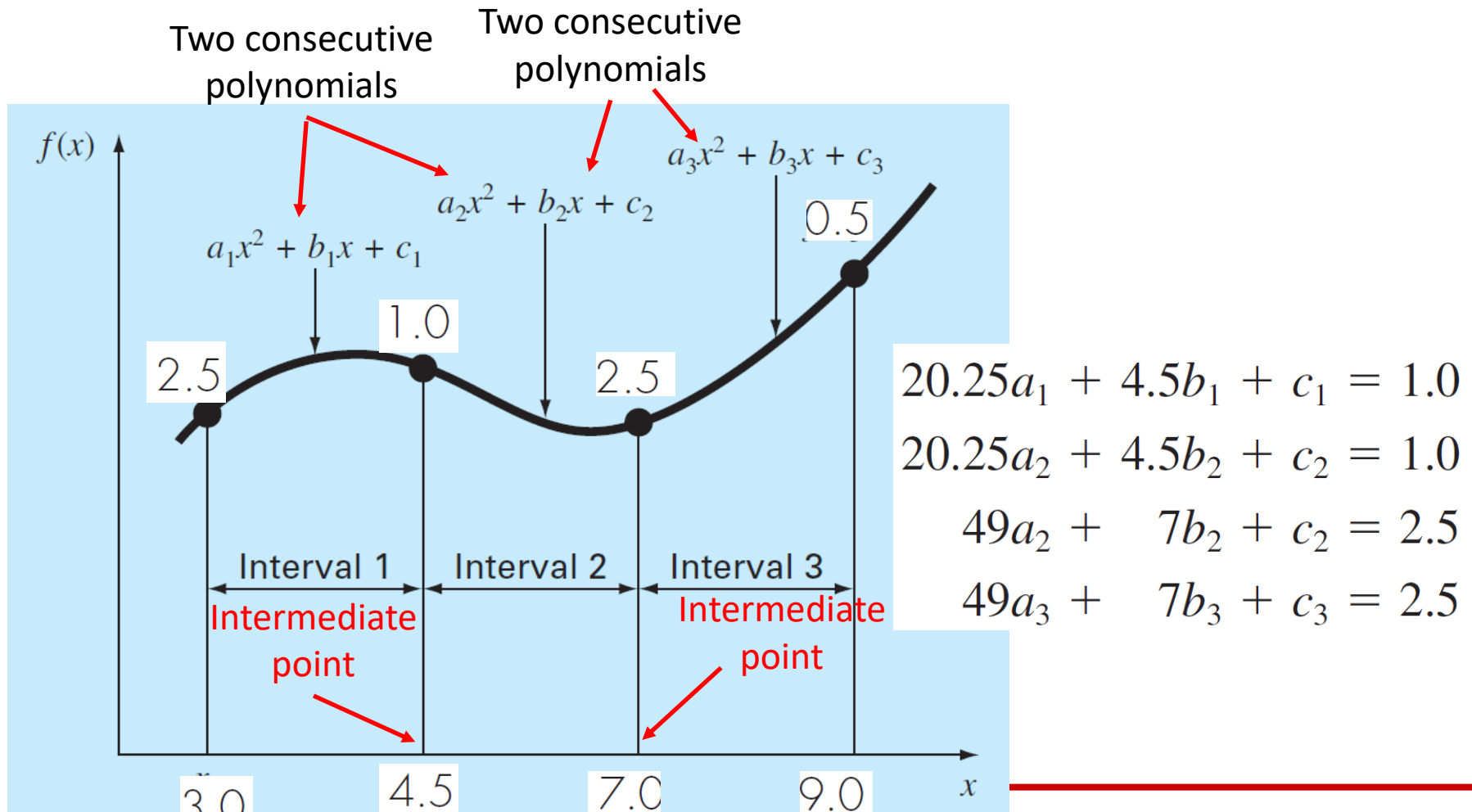
Example

- We have 4 data points. We need 3 quadratic polynomials to connect them.
- 3 quadratic polynomials have 9 parameters.



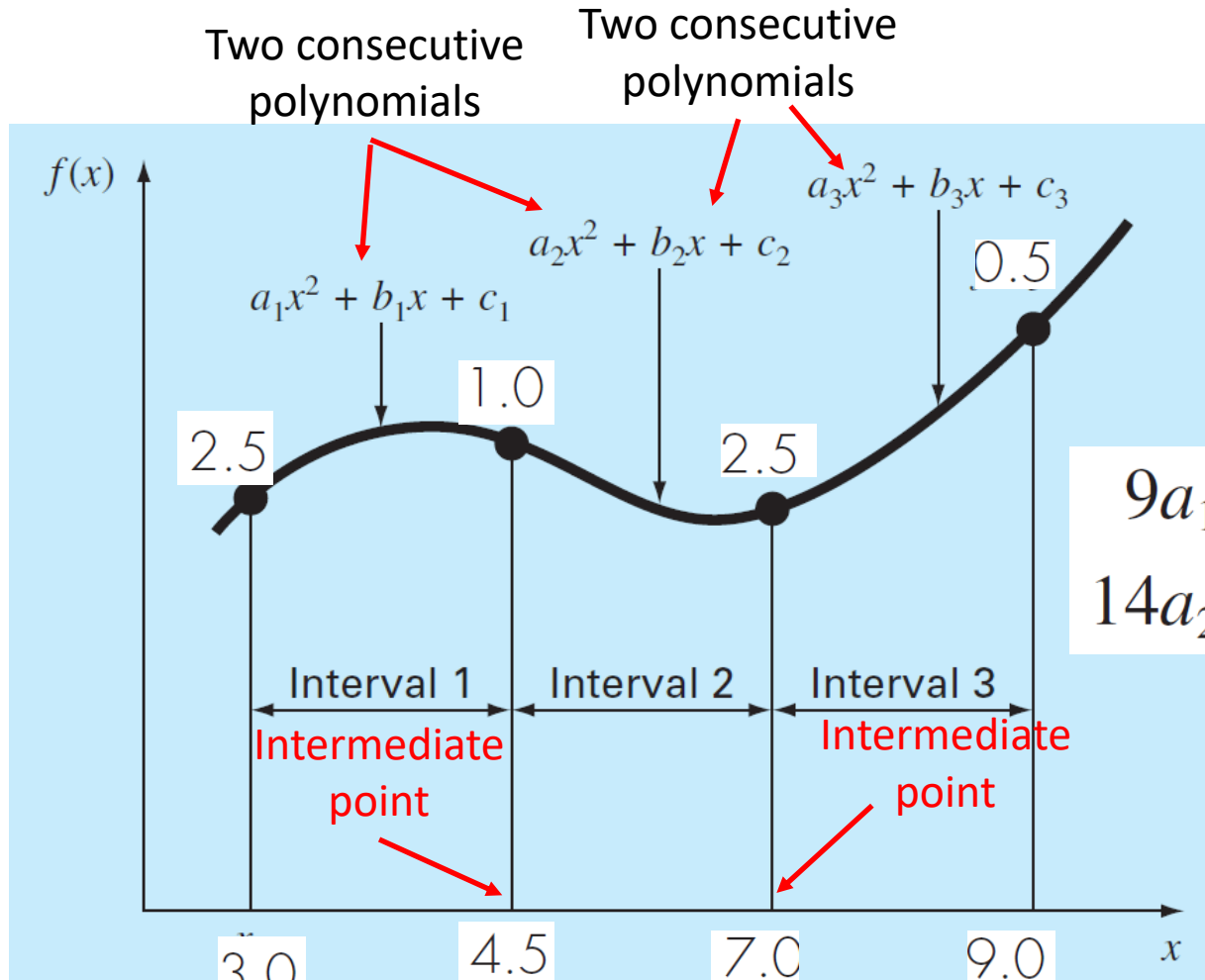
Example

- The consecutive polynomials should pass through their intermediate point.



Example

- The first derivative of consecutive polynomials should be the same at the **intermediate** points.

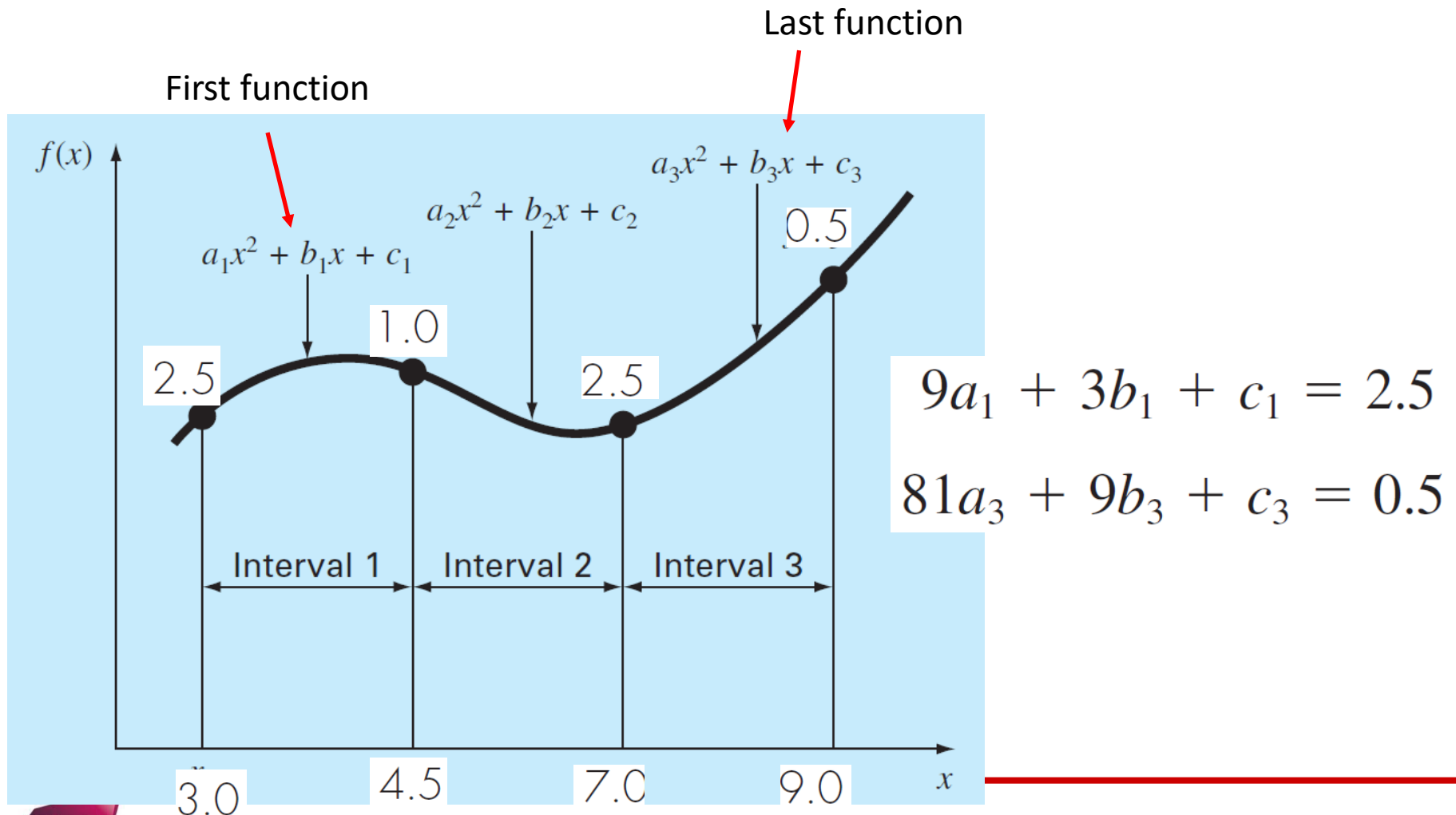


$$9a_1 + b_1 = 9a_2 + b_2$$

$$14a_2 + b_2 = 14a_3 + b_3$$

Example

- The first and last polynomial must pass through the endpoints.



Example

- Also, $a_1 = 0$.

- We have 8 equations.

$$20.25a_1 + 4.5b_1 + c_1 = 1.0$$

$$20.25a_2 + 4.5b_2 + c_2 = 1.0$$

$$49a_2 + 7b_2 + c_2 = 2.5$$

$$49a_3 + 7b_3 + c_3 = 2.5$$

$$9a_1 + b_1 = 9a_2 + b_2$$

$$14a_2 + b_2 = 14a_3 + b_3$$

$$9a_1 + 3b_1 + c_1 = 2.5$$

$$81a_3 + 9b_3 + c_3 = 0.5$$

Example

- Also, $a_1 = 0$.
- We have 8 equations.

$$\begin{bmatrix} 4.5 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 20.25 & 4.5 & 1 & 0 & 0 & 0 \\ 0 & 0 & 49 & 7 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 49 & 7 & 1 \\ 3 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 81 & 9 & 1 \\ 1 & 0 & -9 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 14 & 1 & 0 & -14 & -1 & 0 \end{bmatrix} \begin{bmatrix} b_1 \\ c_1 \\ a_2 \\ b_2 \\ c_2 \\ a_3 \\ b_3 \\ c_3 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 2.5 \\ 2.5 \\ 2.5 \\ 0.5 \\ 0 \\ 0 \end{bmatrix}$$

Example

- The values of unknowns are

$$\begin{array}{lll} a_1 = 0 & b_1 = -1 & c_1 = 5.5 \\ a_2 = 0.64 & b_2 = -6.76 & c_2 = 18.46 \\ a_3 = -1.6 & b_3 = 24.6 & c_3 = -91.3 \end{array}$$

- The quadratic polynomials are

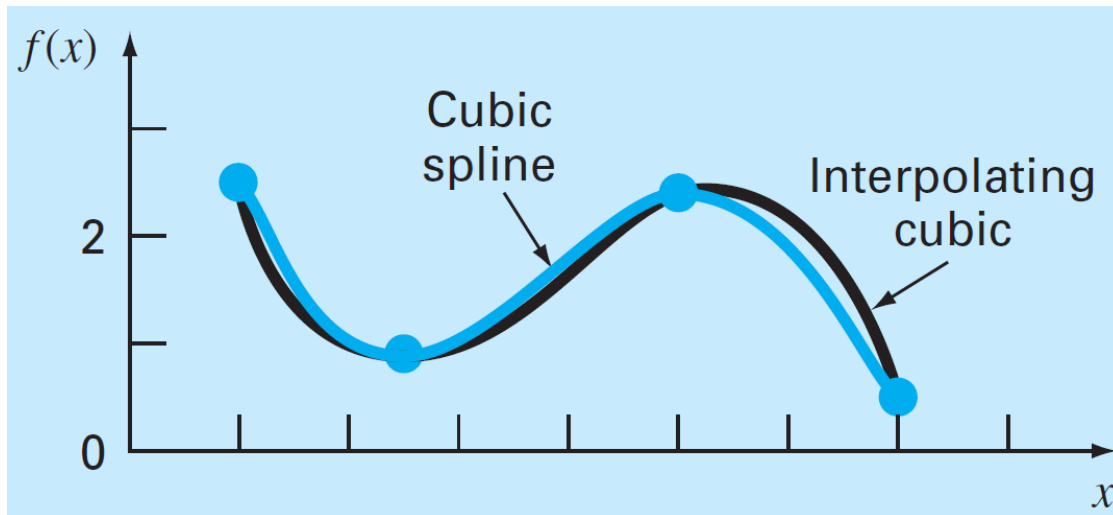
$$\begin{array}{ll} f_1(x) = -x + 5.5 & 3.0 \leq x \leq 4.5 \\ f_2(x) = 0.64x^2 - 6.76x + 18.46 & 4.5 \leq x \leq 7.0 \\ f_3(x) = -1.6x^2 + 24.6x - 91.3 & 7.0 \leq x \leq 9.0 \end{array}$$

- The output at $x = 5$ is

$$f_2(5) = 0.64(5)^2 - 6.76(5) + 18.46 = 0.66$$

Cubic Splines

- In the cubic spline method, data points are connected by cubic functions.
- First, arrange the data points according to the increasing order, $x_0 < x_1 < x_2 < \dots < x_n$.



$$\begin{aligned} &(x_0, f(x_0)) \\ &(x_1, f(x_1)) \\ &\vdots \\ &(x_n, f(x_n)) \end{aligned}$$

Cubic Splines

- The first cubic polynomial connects $f(x_0)$ and $f(x_1)$.
- The second cubic polynomial connects $f(x_2)$ and $f(x_3)$.
- The last cubic polynomial connects $f(x_{n-1})$ and $f(x_n)$.

$$\begin{array}{llll} f(x) = a_1x^3 + b_1x^2 + c_1x + d_1, & x_0 \leq x \leq x_1 & (x_0, f(x_0)) \\ f(x) = a_2x^3 + b_2x^2 + c_2x + d_2, & x_1 \leq x \leq x_2 & (x_1, f(x_1)) \\ \vdots & & (x_2, f(x_2)) \\ \vdots & & \vdots \\ f(x) = a_nx^3 + b_nx^2 + c_nx + d_n, & x_{n-1} \leq x \leq x_n & (x_{n-1}, f(x_{n-1})) \\ & & (x_n, f(x_n)) \end{array}$$

Cubic Splines

- For $n + 1$ data points, we need to find the parameters of n cubic functions.
- We will find a system of linear equations. The parameters will be obtained by solving the system of linear equations.

$$\begin{aligned} f(x) &= a_1x^3 + b_1x^2 + c_1x + d_1, & x_0 \leq x \leq x_1 \\ f(x) &= a_2x^3 + b_2x^2 + c_2x + d_2, & x_1 \leq x \leq x_2 \\ &\vdots \\ f(x) &= a_nx^3 + b_nx^2 + c_nx + d_n, & x_{n-1} \leq x \leq x_n \end{aligned}$$

Cubic Splines

We form a system of linear equations with $4n$ unknowns and $4n$ equations.

1. The **consecutive polynomials** must pass through the intermediate points ($2n - 2$ conditions).
2. The **first and last polynomials** must pass through the end points (2 conditions).
3. The first derivatives **consecutive polynomials** at their intermediate points must be equal ($n - 1$ conditions).
4. The second derivatives **consecutive polynomials** at their intermediate points must be equal ($n - 1$ conditions).
5. The **second derivatives** of **the first and last polynomials** at the endpoints should be zero (2 conditions).

Cubic Splines

We form a system of linear equations with $4n$ unknowns and $4n$ equations.

1. The **consecutive polynomials** must pass through the intermediate points ($2n - 2$ conditions).
2. The **first and last polynomials** must pass through the end points (2 conditions).
3. The first derivatives **consecutive polynomials** at their intermediate points must be equal ($n - 1$ conditions).
4. The second derivatives **consecutive polynomials** at their intermediate points must be equal ($n - 1$ conditions).
5. The **second derivatives** of **the first and last polynomials** at the endpoints should be zero (2 conditions).

END

See you next week 😊!