AI VIET NAM
@aivietnam.edu.vn

Advanced Linear Regression
Loss Functions

Quang-Vinh Dinh
Ph.D. in Computer Science

Year 2024

Review

❖ Flowchart

Linear equation

$$\hat{y} = wx + b$$

where $\hat{y}$ is a predicted value,

w and b are parameters

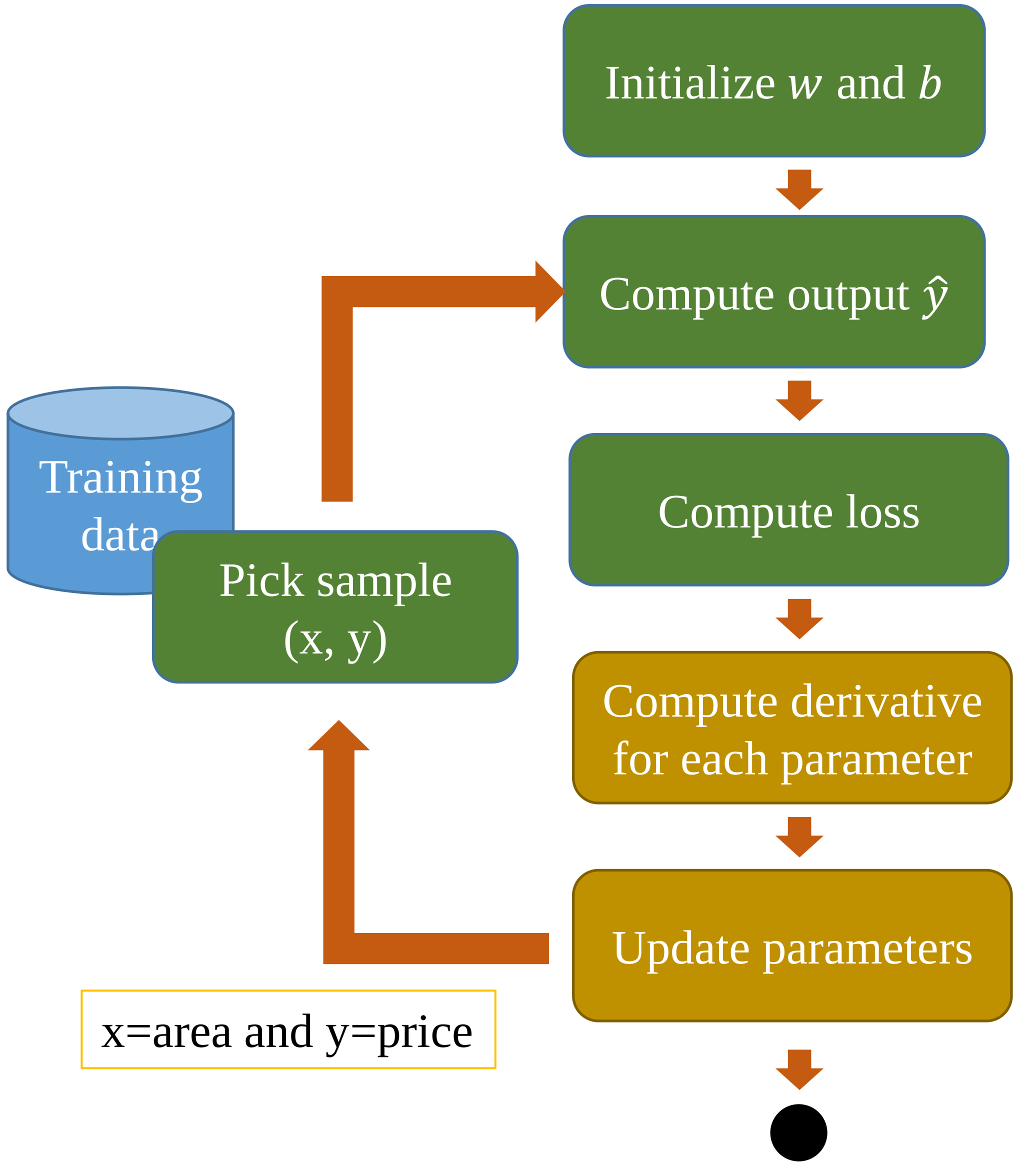
and x is an input feature

Error (loss) computation

Idea: compare predicted values $\hat{y}$ and label values y

Squared loss

$$L(\hat{y}, y) = (\hat{y} - y)^2$$



Review

❖ Formulae

Linear equation

$$\hat{y} = wx + b$$

where $\hat{y}$ is a predicted value,

w and b are parameters

and x is an input feature

Error (loss) computation

Idea: compare predicted values $\hat{y}$ and label values y

Squared loss

$$\mathrm{L}(\hat{y}, y) = (\hat{y} - y)^2$$

Find better w and b

Use gradient descent to minimize the loss function

Compute derivate for each parameter

$$\frac{\partial L}{\partial w} = \frac{\partial L}{\partial \hat{y}} \frac{\partial \hat{y}}{\partial w} = 2x(\hat{y} - y)$$

$$\frac{\partial L}{\partial b} = \frac{\partial L}{\partial \hat{y}} \frac{\partial \hat{y}}{\partial b} = 2(\hat{y} - y)$$

Update parameters

$$w = w - \eta \frac{\partial L}{\partial w} \qquad b = b - \eta \frac{\partial L}{\partial b}$$

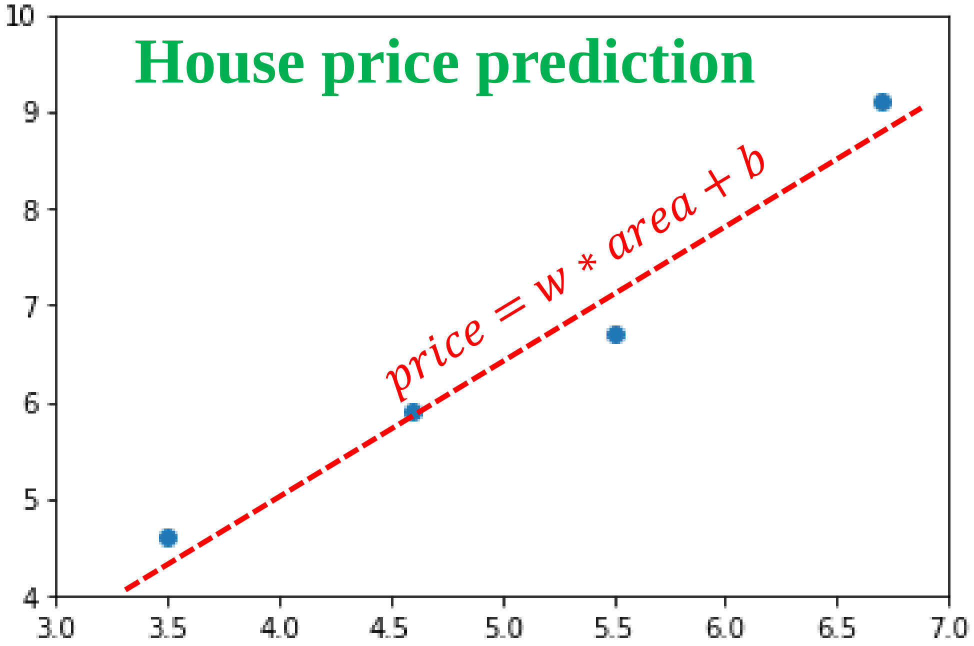
η is learning rate

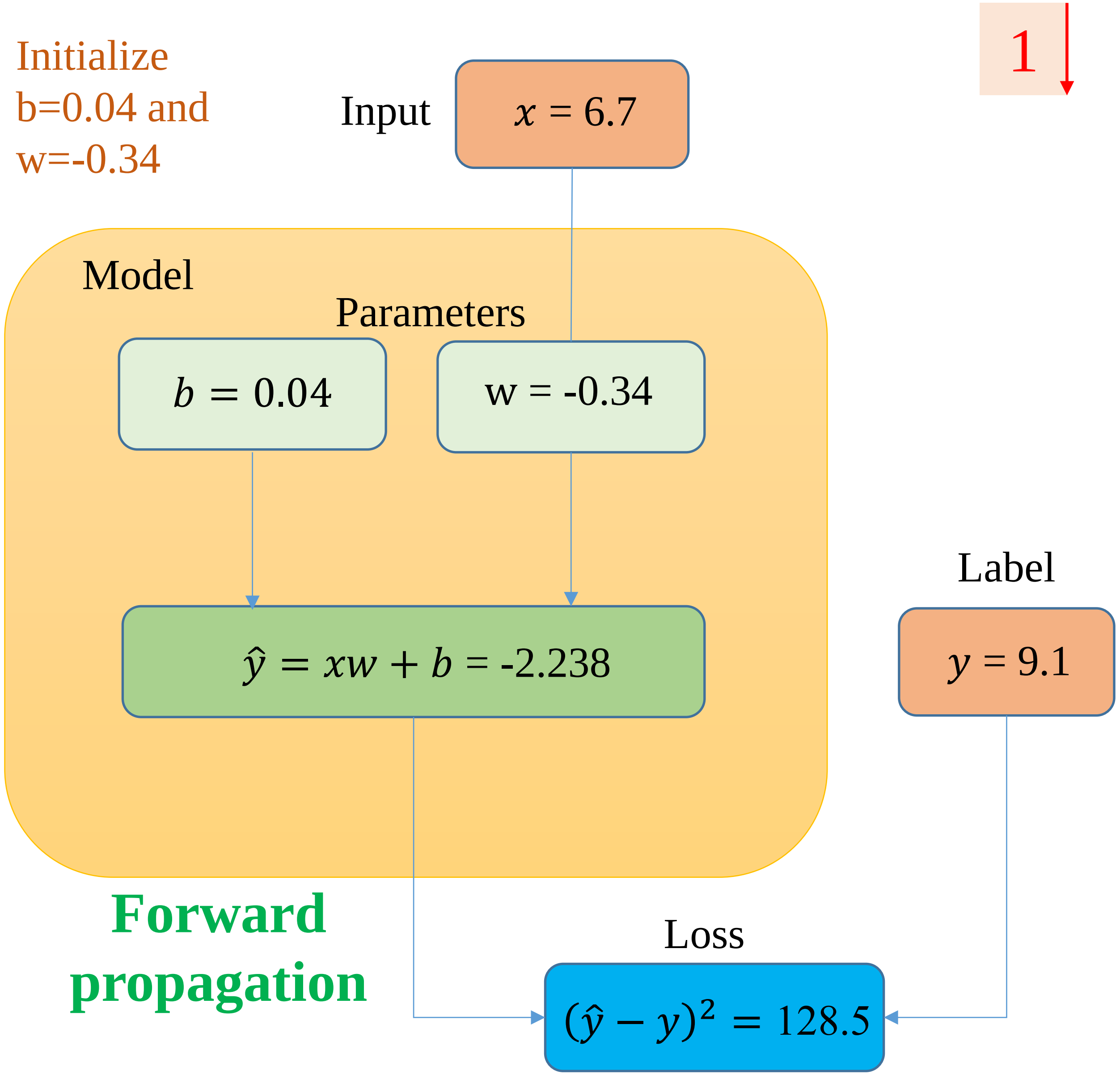
Review

❖ Example

Given sample data

	Feature	Label
	area	price
	6.7	9.1
	4.6	5.9
	3.5	4.6
	5.5	6.7





Review

2

Backpropagation

Input

$x = 6.7$

$\eta = 0.01$

Model

Parameters

$b = 0.26676$

$w = 1.17929$

$b = b - \eta \frac{\partial L}{\partial b}$

$\mathrm{w} = \mathrm{w} - \eta \frac{\partial L}{\partial w}$

$\hat{y} = xw + b = \text{-}2.238$

Label

$y = 9.1$

$\frac{\partial L}{\partial w} = 2x(\hat{y} - y) = -151.9292$

$\frac{\partial L}{\partial b} = 2(\hat{y} - y) = -22.676$

Loss

$(\hat{y} - y)^2 = 128.5$

3

Forward propagation

Input

$x = 6.7$

Model

Parameters

$b = 0.26676$

$w = 1.17929$

$b = b - \eta \frac{\partial L}{\partial b}$

$\mathrm{w} = \mathrm{w} - \eta \frac{\partial L}{\partial w}$

$\hat{y} = xw + b = 8.168$

Label

$y = 9.1$

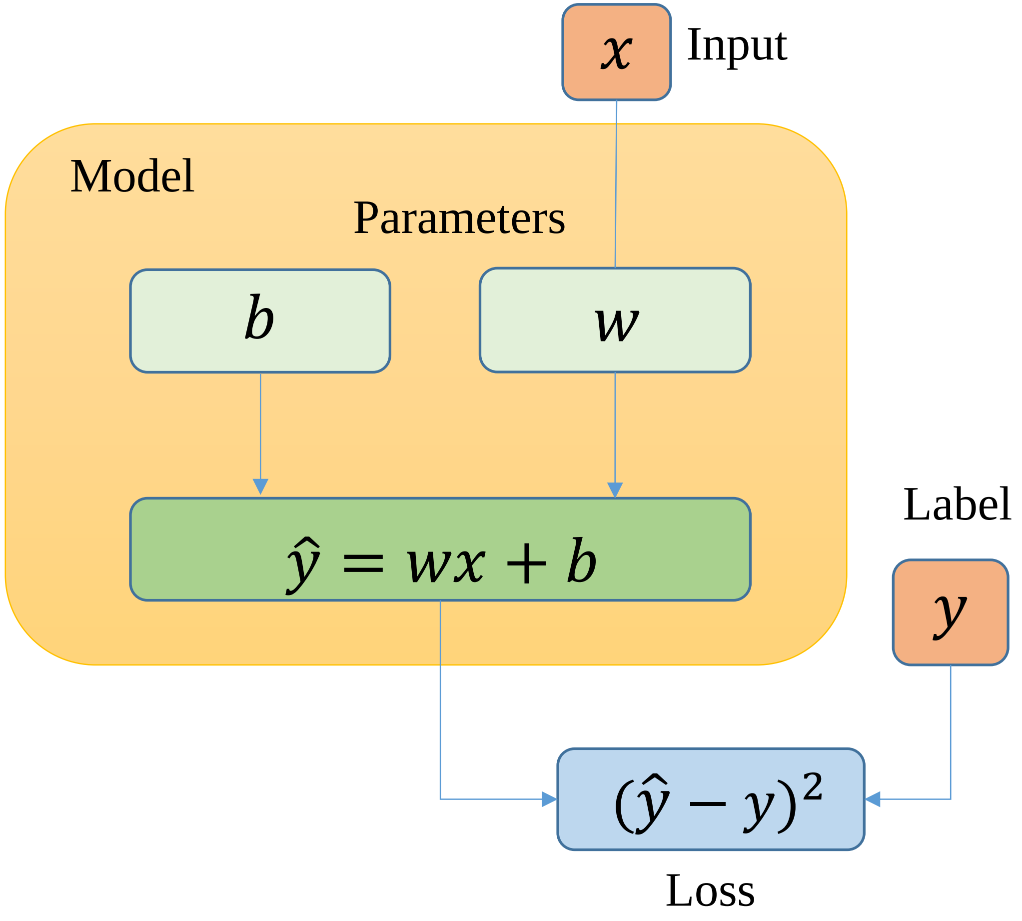
Loss

$(\hat{y} - y)^2 = 0.868$

New w and b help the loss reduce

Review

❖ Summary (simple version)



1) Pick a sample (x, y) from training data

2) Compute the output $\hat{y}$

$$\hat{y} = wx + b$$

3) Compute loss

$$L = (\hat{y} - y)^2$$

4) Compute derivative

$$\frac{\partial L}{\partial w} = 2x(\hat{y} - y) \qquad \frac{\partial L}{\partial b} = 2(\hat{y} - y)$$

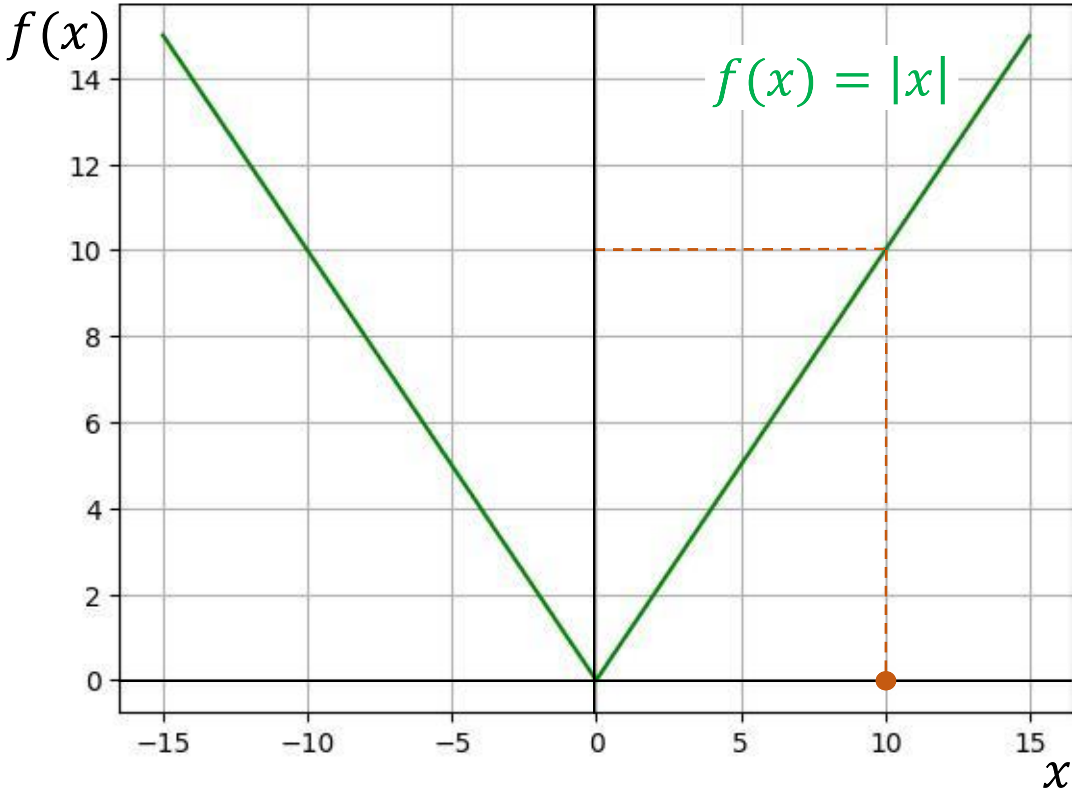
5) Update parameters

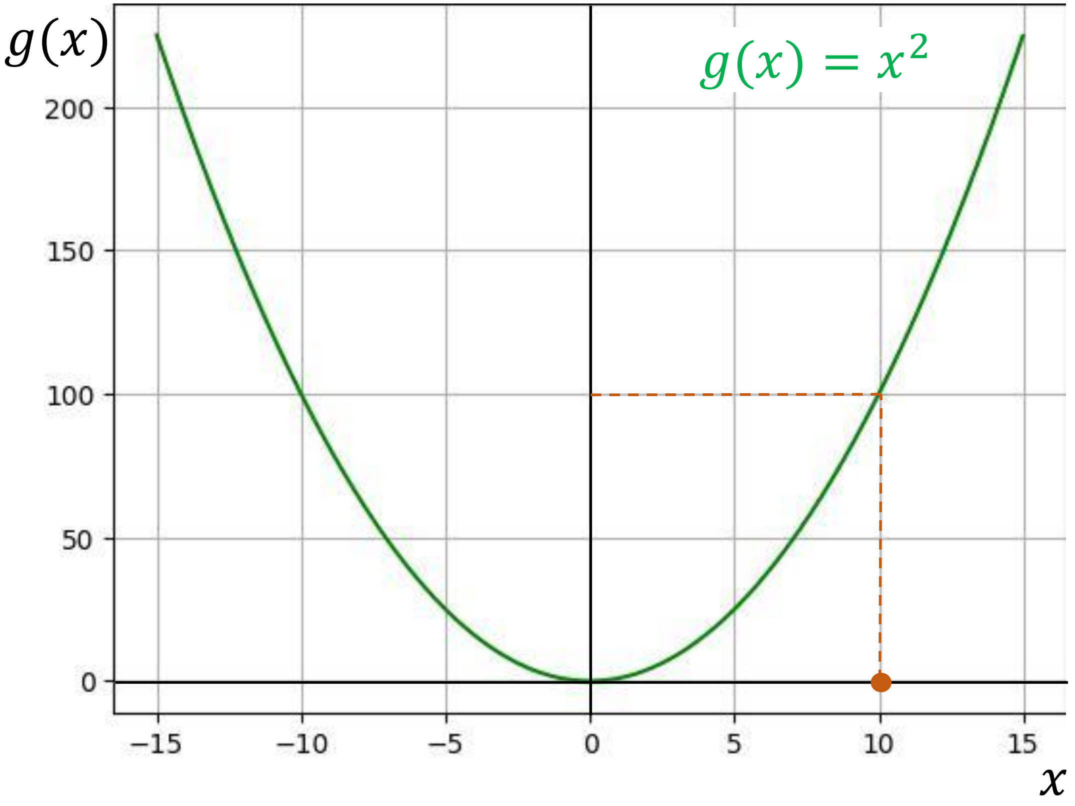
$$w = w - \eta \frac{\partial L}{\partial w} \qquad b = b - \eta \frac{\partial L}{\partial b}$$

η is learning rate

Objectives

❖ Loss functions



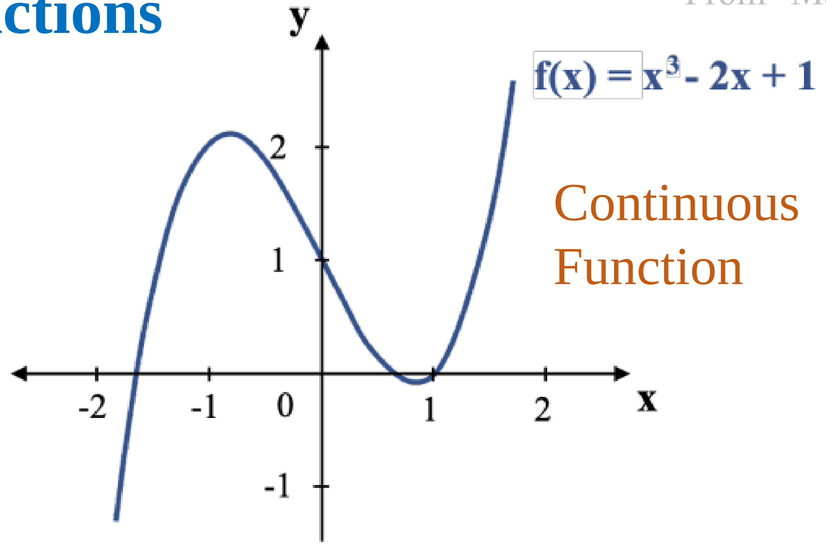


Outline

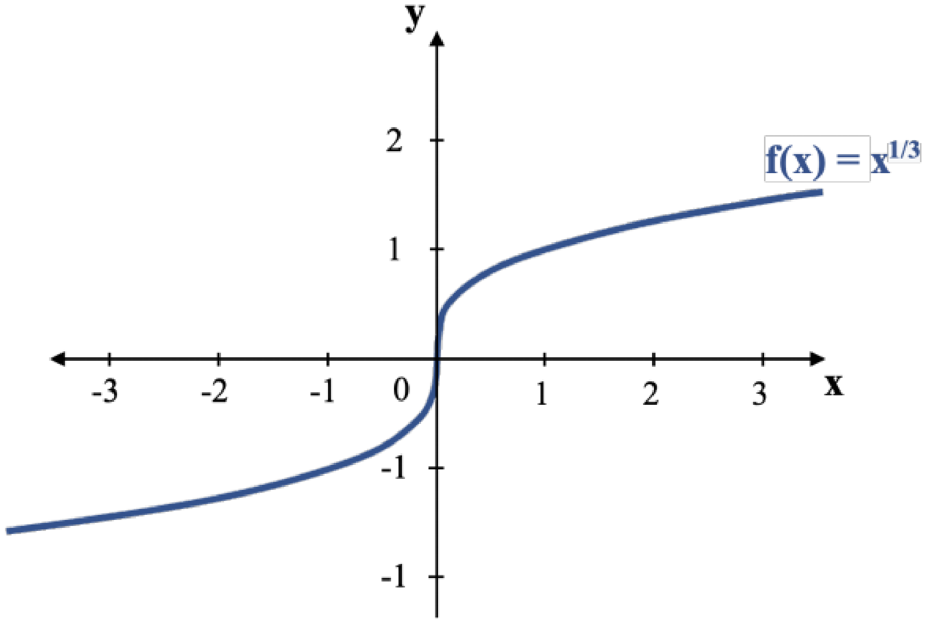
Conditions for Loss Functions

Loss functions

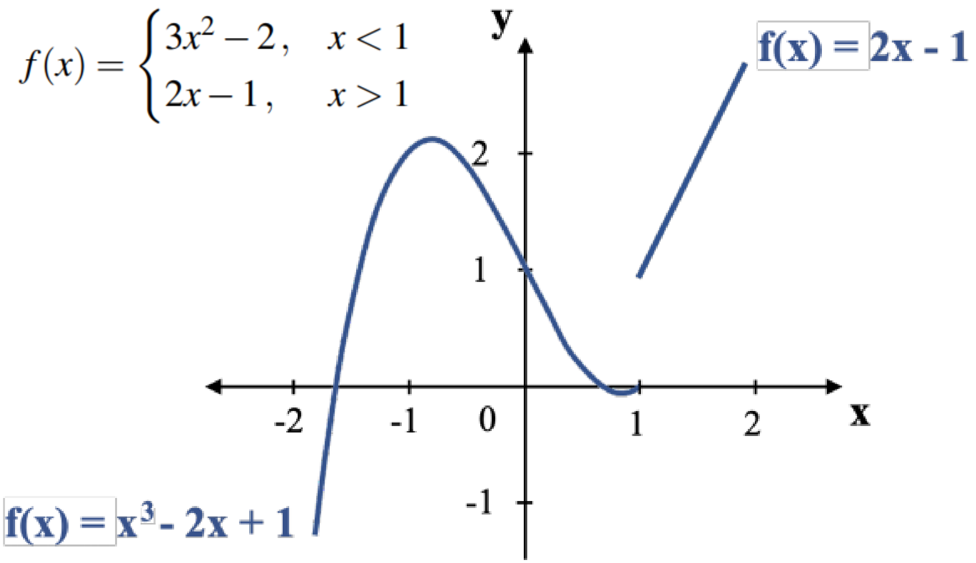
From "Machine Learning Simplified"

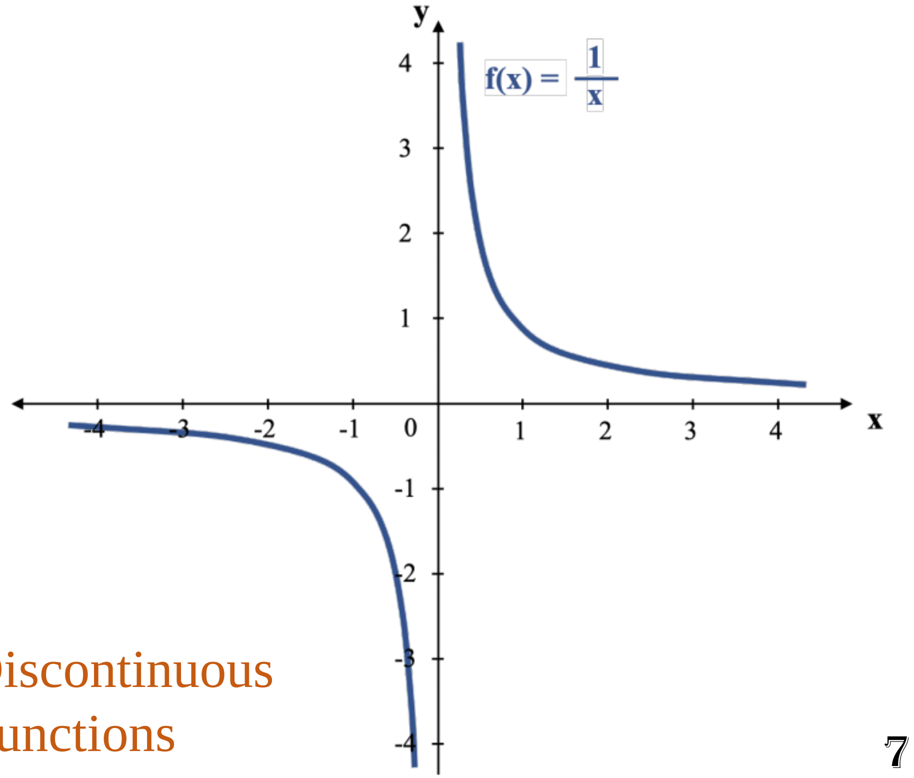


Continuous
Function



Continuous non-differentiable functions



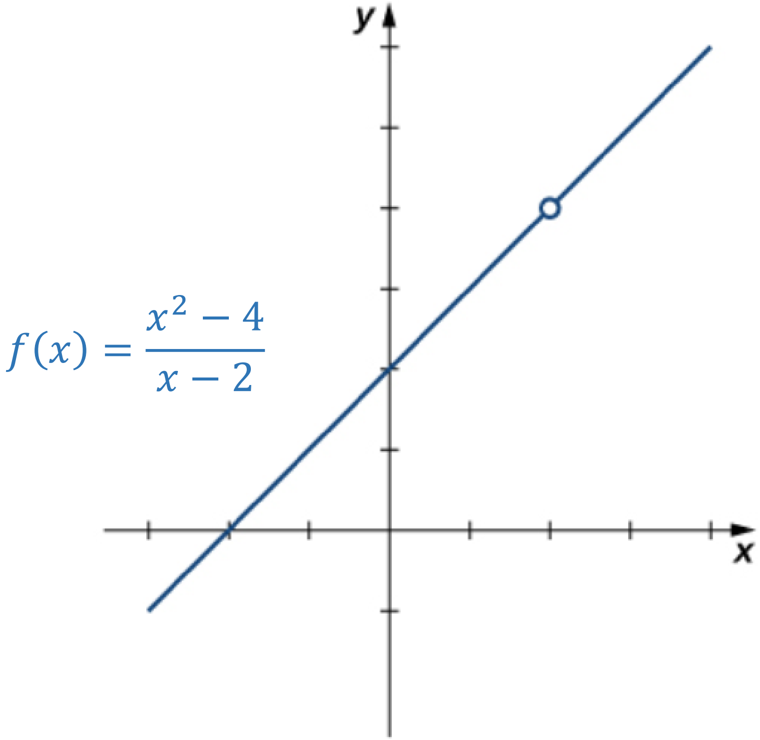


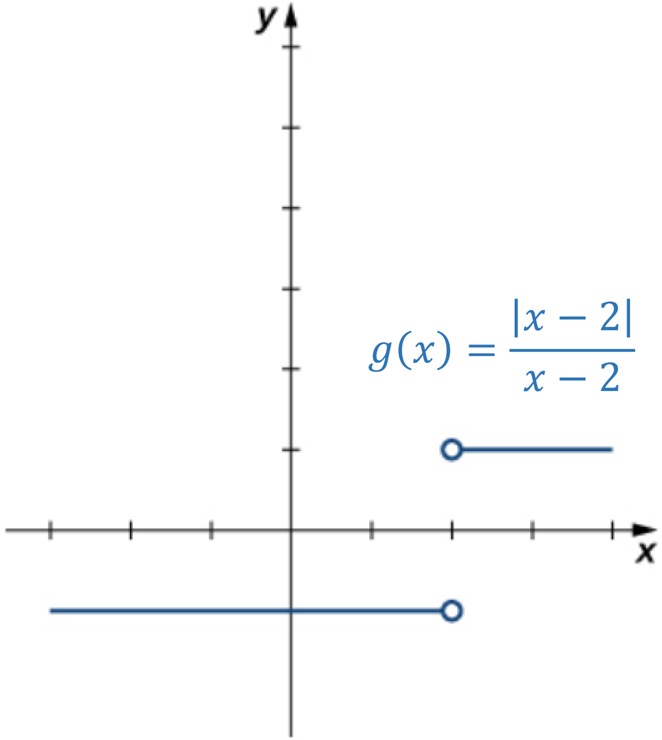
Discontinuous
Functions

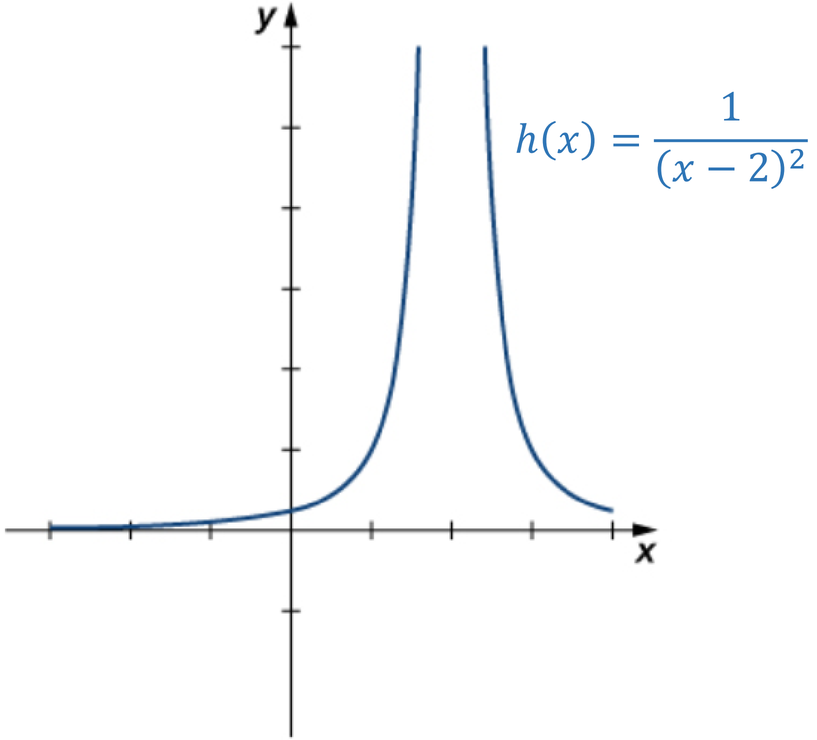
Conditions for Loss Functions

❖ Definition

from the reference book







$$\lim_{x \to 2} f(x) = ?$$

$$\lim_{x \to 2} g(x) = ?$$

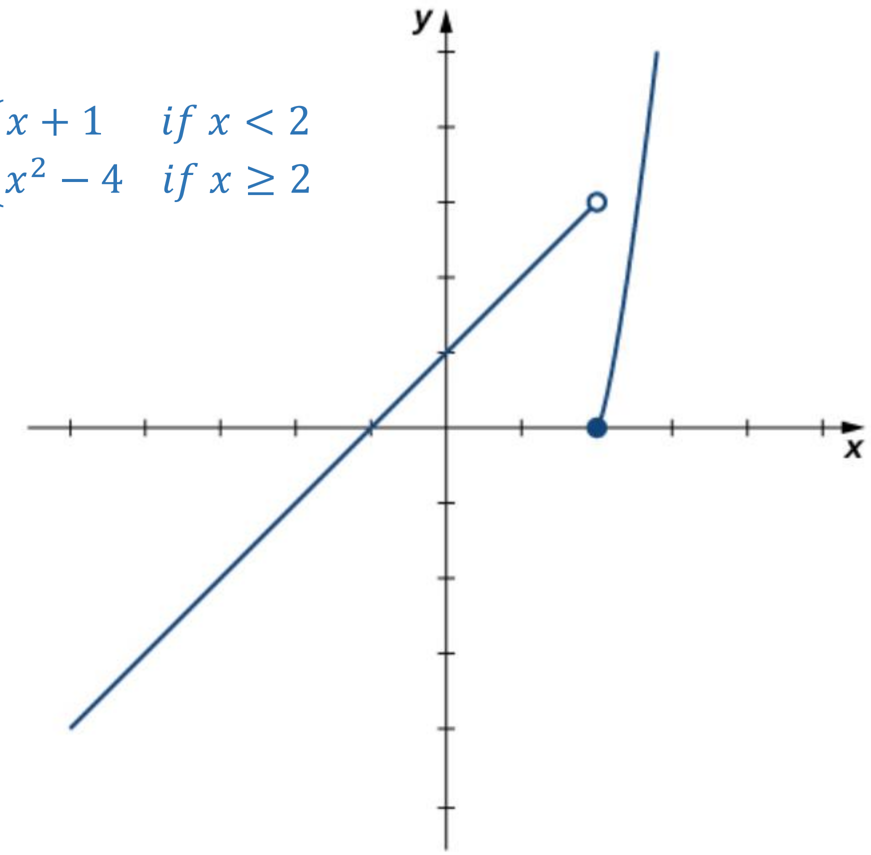
$$\lim_{x \to 2} h(x) = ?$$

Conditions for Loss Functions

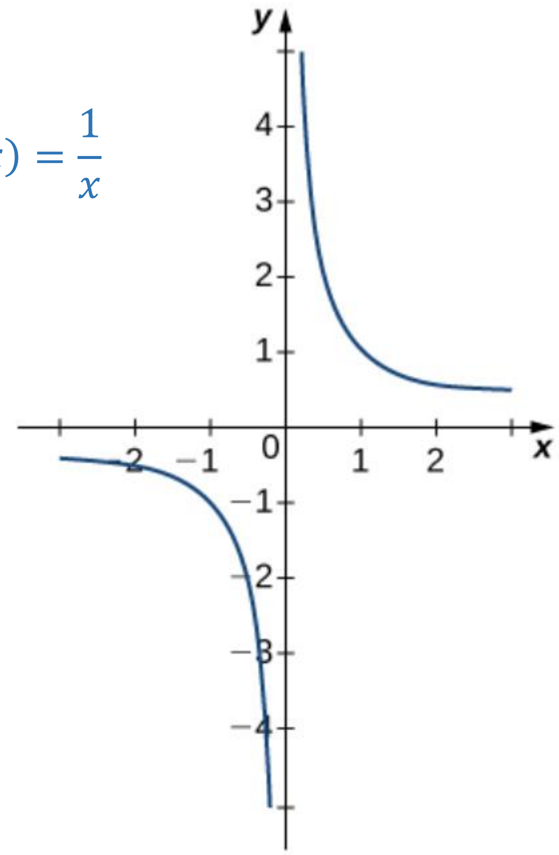
❖ Definition

from the reference book

$$f(x) = \begin{cases} x+1 & if\ x < 2 \\ x^2 - 4 & if\ x \geq 2 \end{cases}$$



$$g(x) = \frac{1}{x}$$



$$\lim_{x \to 2} f(x) = ?$$

$$\lim_{x \to 0} g(x) = ?$$

Continuity

Definition

A function $f(x)$ is **continuous at a point** a if and only if the following three conditions are satisfied:

i. $f(a)$ is defined

ii. $\lim_{x \to a} f(x)$ exists

iii. $\lim_{x \to a} f(x) = f(a)$

A function is **discontinuous at a point** a if it fails to be continuous at a.

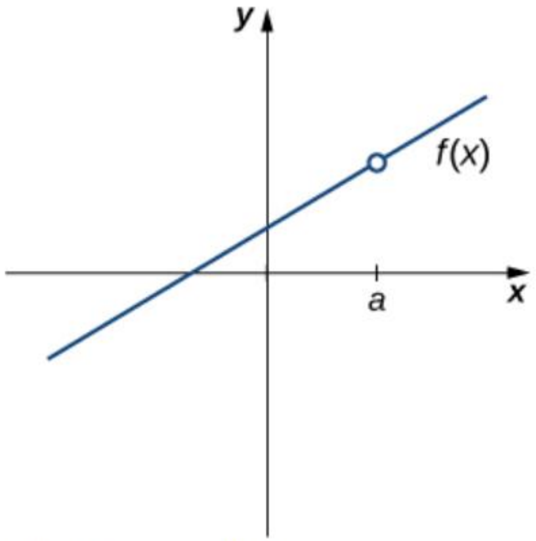


Figure 2.32 The function $f(x)$ is not continuous at a because $f(a)$ is undefined.

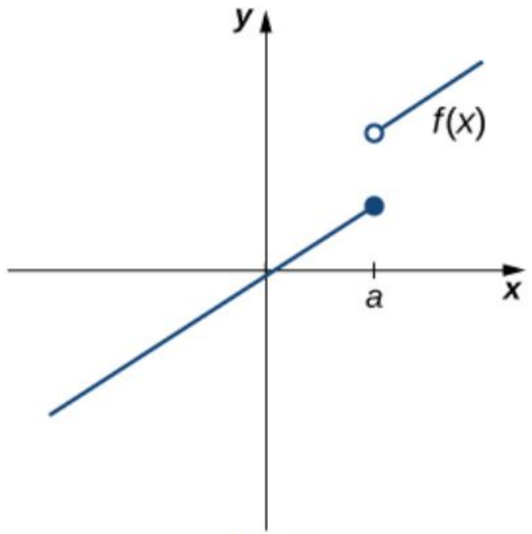


Figure 2.33 The function $f(x)$ is not continuous at a because $\lim_{x \to a} f(x)$ does not exist.

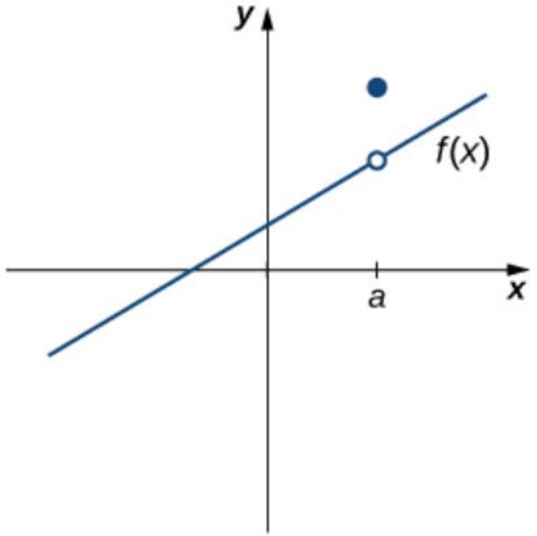


Figure 2.34 The function $f(x)$ is not continuous at a because $\lim_{x \to a} f(x) \neq f(a)$.

Differentiability

❖ Definition

Definition

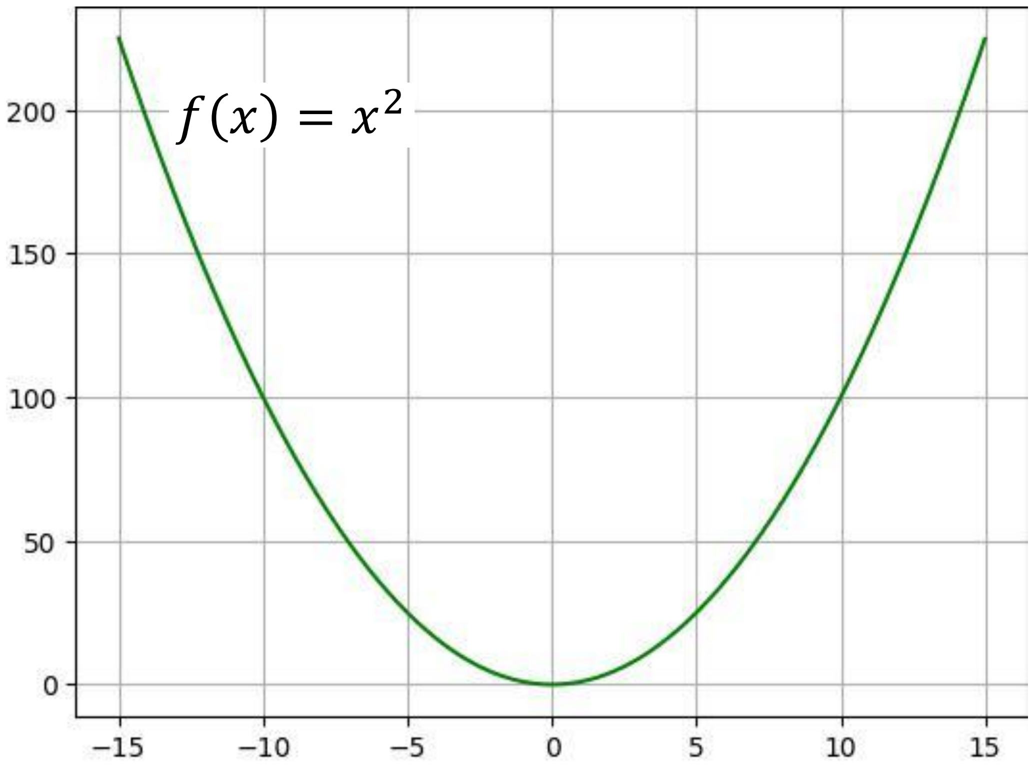
Let f be a function. The **derivative function**, denoted by f', is the function whose domain consists of those values of x such that the following limit exists:

$$f'(x) = \lim_{h \to 0} \frac{f(x+h) - f(x)}{h}. \qquad (3.9)$$

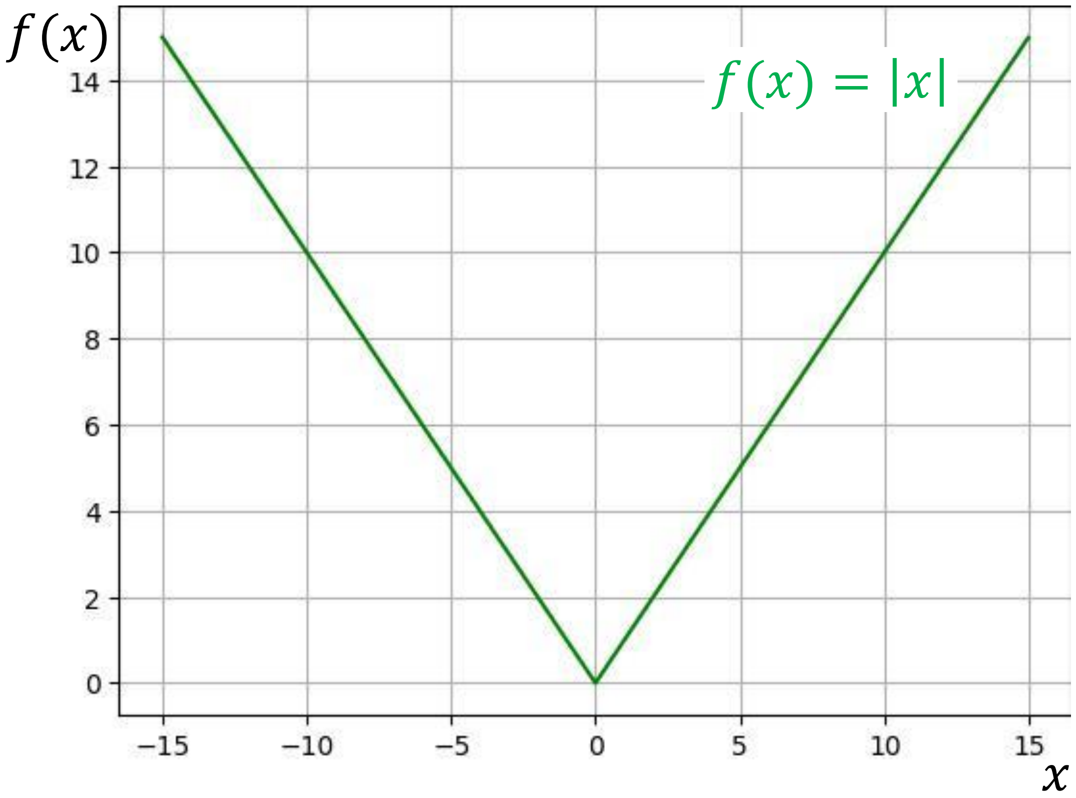
A function $f(x)$ is said to be **differentiable at** $\boldsymbol{a}$ if $f'(a)$ exists. More generally, a function is said to be **differentiable on** $\boldsymbol{S}$ if it is differentiable at every point in an open set S, and a **differentiable function** is one in which $f'(x)$ exists on its domain.

https://openstax.org/details/books/calculus-volume-1/

Check if the function is continuous and differentiable



Check if the function is continuous and differentiable



Discussion 1: Is it OK to use the following loss function?

$$L = \frac{1}{2}(\hat{y} - y)^2$$

Discussion 2: if so, construct formulas

Discussion 3: What about the following loss function?

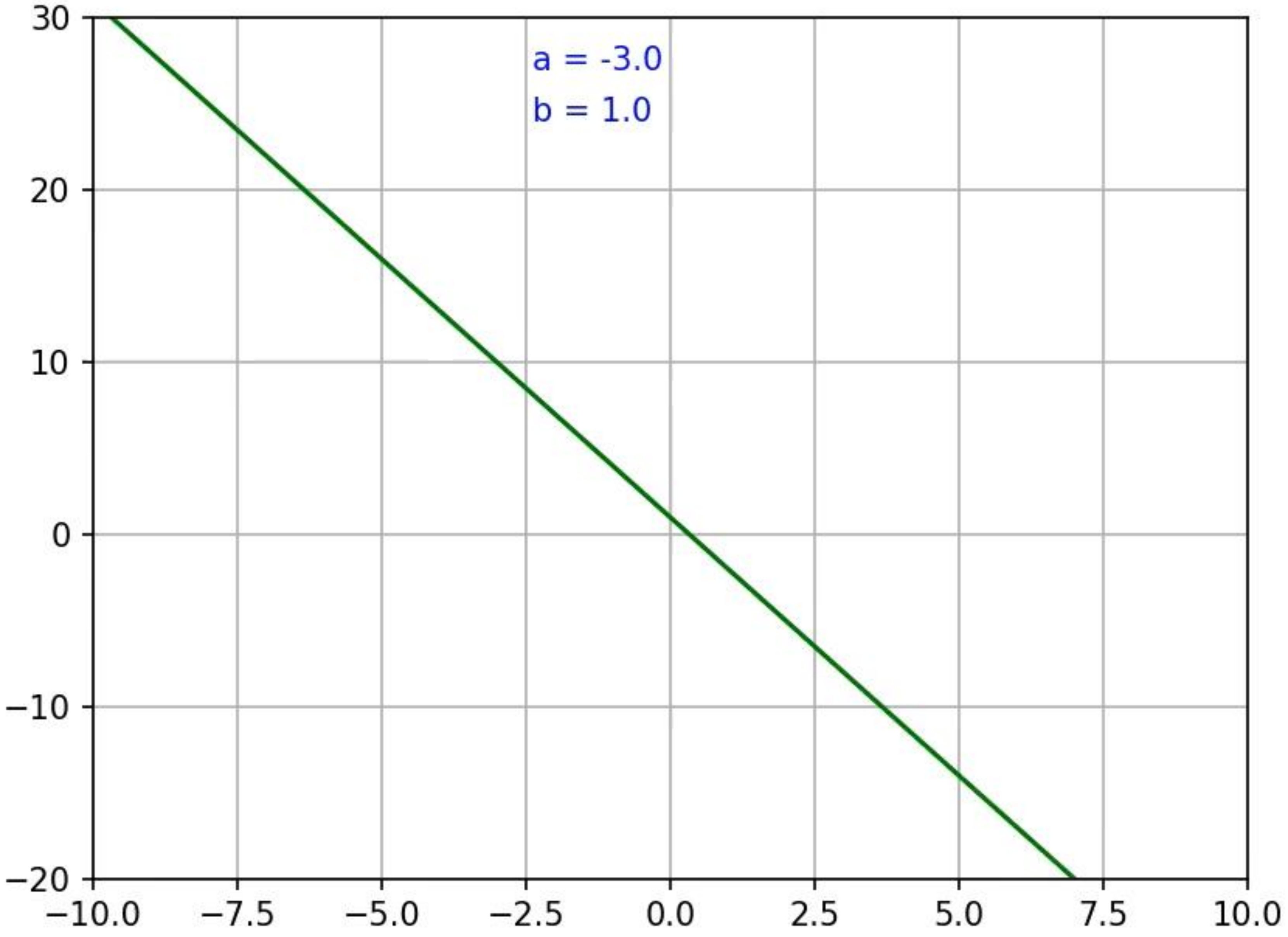
$$L = \frac{1}{2}(y - \hat{y})^2$$

Discussion 4: Construct the connection between the two following losses?

$$\mathrm{L}_1 = \frac{1}{2}(\hat{y} - y)^2 \qquad \mathrm{L}_2 = (\hat{y} - y)^2$$

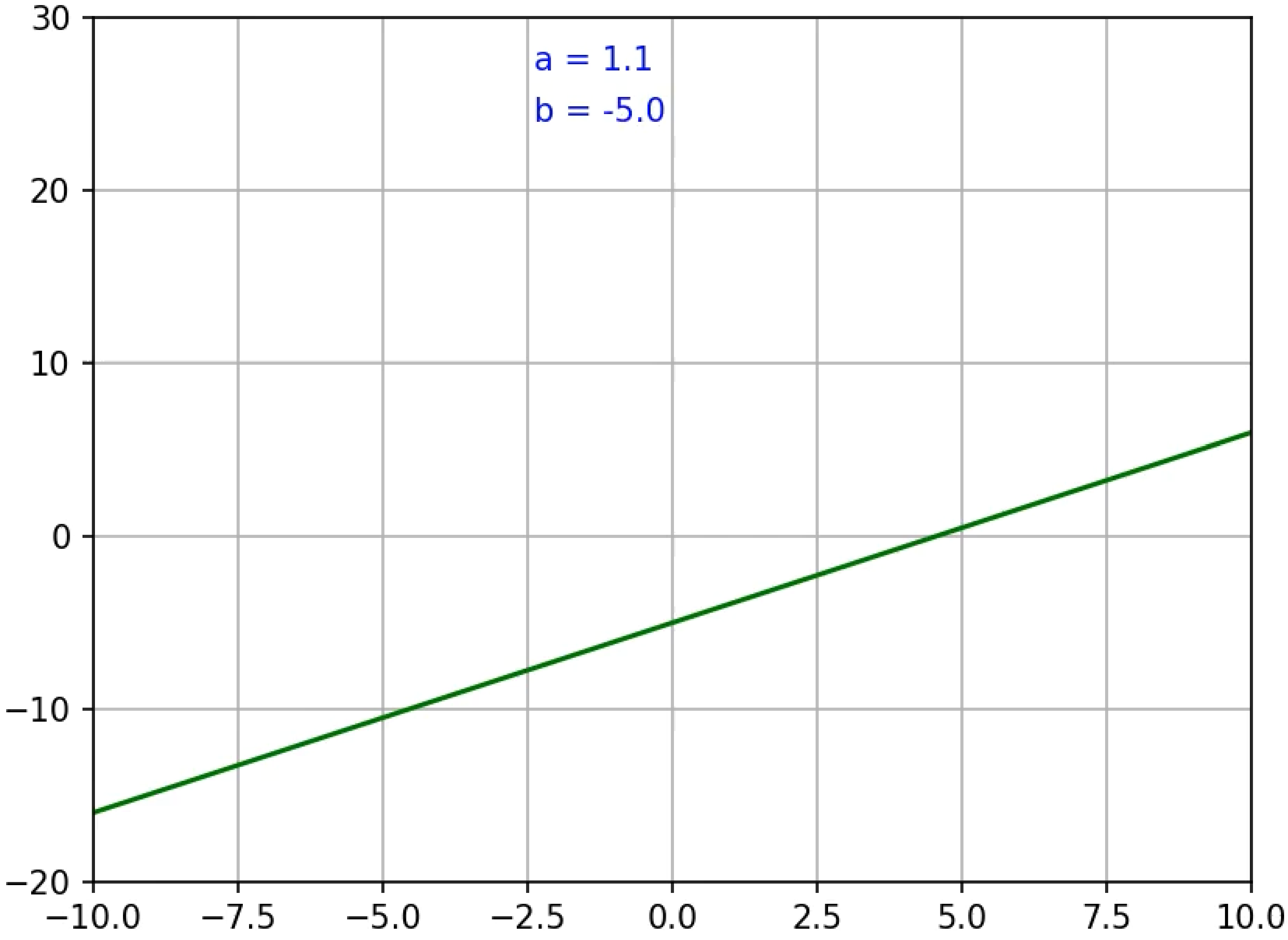
$$\hat{y} = wx + b$$

Discussion 5:
Can we remove b?



$$\hat{y} = wx + b$$

Discussion 5:
Can we remove b?



Outline

SECTION 1

Variants of MSE

SECTION 2

Mean Absolute Error

SECTION 3

Huber Loss

SECTION 4

Data Normalization

Discussion 6: What about the following loss function?

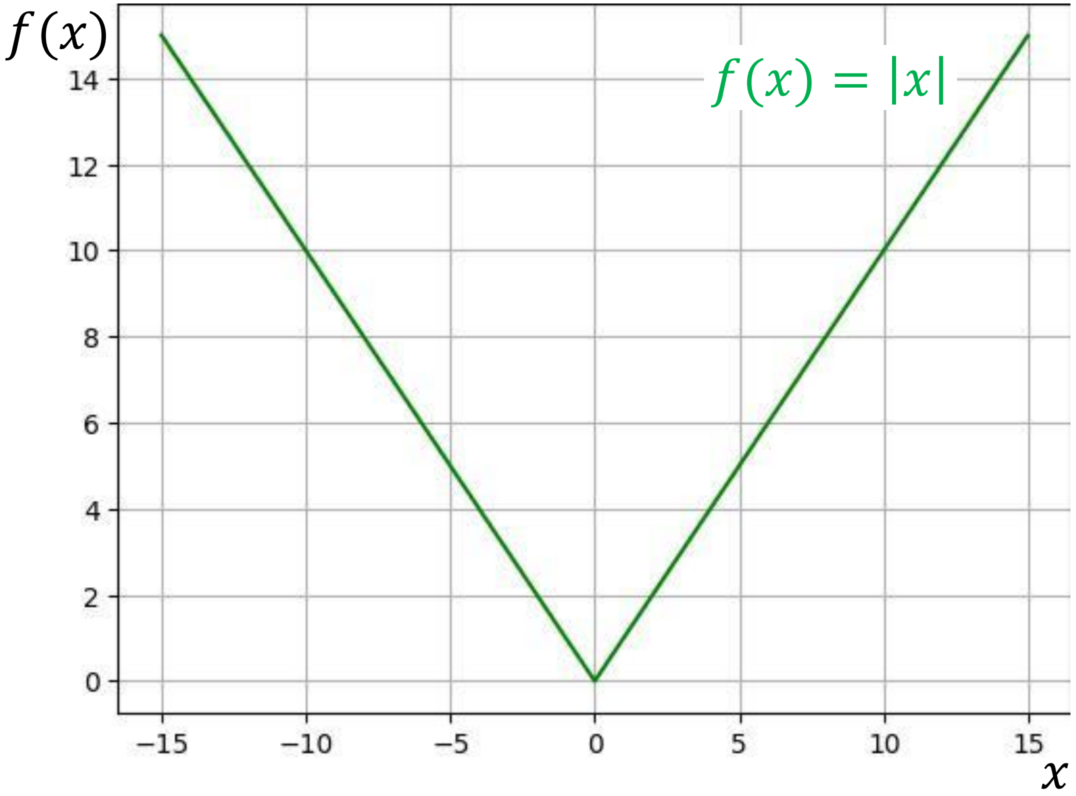
$$L = |\hat{y} - y|$$

AI VIET NAM
@aivietnam.edu.vn

Loss Functions

Feature	Label
area	price
6.7	9.1
4.6	5.9
3.5	4.6
5.5	6.7

❖ Mean Absolute Error (MAE)



$$f'(x) = \frac{x}{|x|} \quad \text{for } x \neq 0$$

w = 1.185

b = 0.340

One sample

$$L(\hat{y}, y) = |\hat{y} - y|$$

I area	y price	$\hat{y}$ prediction	L error
6.7	9.1	5.5	3.6
4.6	5.9	3.8	2.1
3.5	4.6	3.1	1.5
5.5	6.7	4.6	2.1

Loss Functions

Feature	Label
area	price
6.7	9.1
4.6	5.9
3.5	4.6
5.5	6.7

❖ Mean Absolute Error (MAE)

1) Pick a sample (x, y) from training data

2) Compute the output $\hat{y}$

$$\hat{y} = wx + b$$

3) Compute loss

$$L = |\hat{y} - y|$$

4) Compute derivative

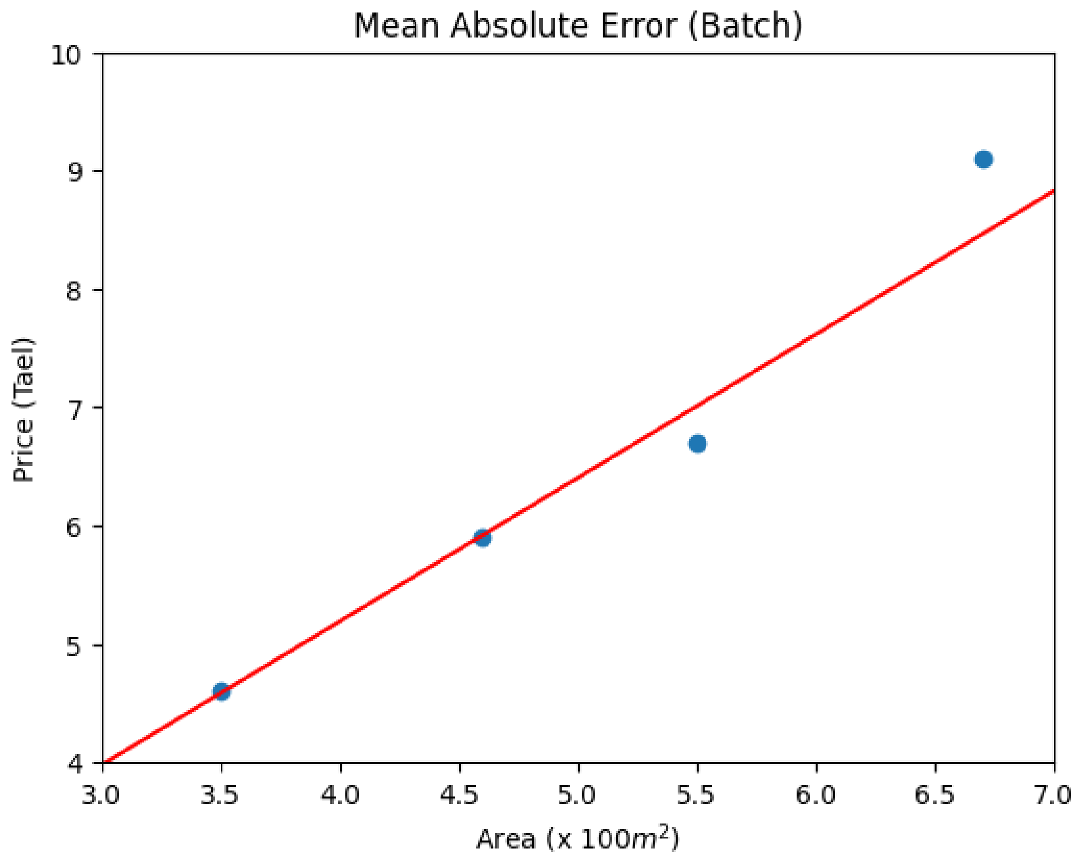
$$\frac{\partial L}{\partial w} = x\frac{(\hat{y} - y)}{|\hat{y} - y|} \qquad \frac{\partial L}{\partial b} = \frac{(\hat{y} - y)}{|\hat{y} - y|}$$

5) Update parameters

$$w = w - \eta\frac{\partial L}{\partial w} \qquad b = b - \eta\frac{\partial L}{\partial b}$$

$w = 1.185$

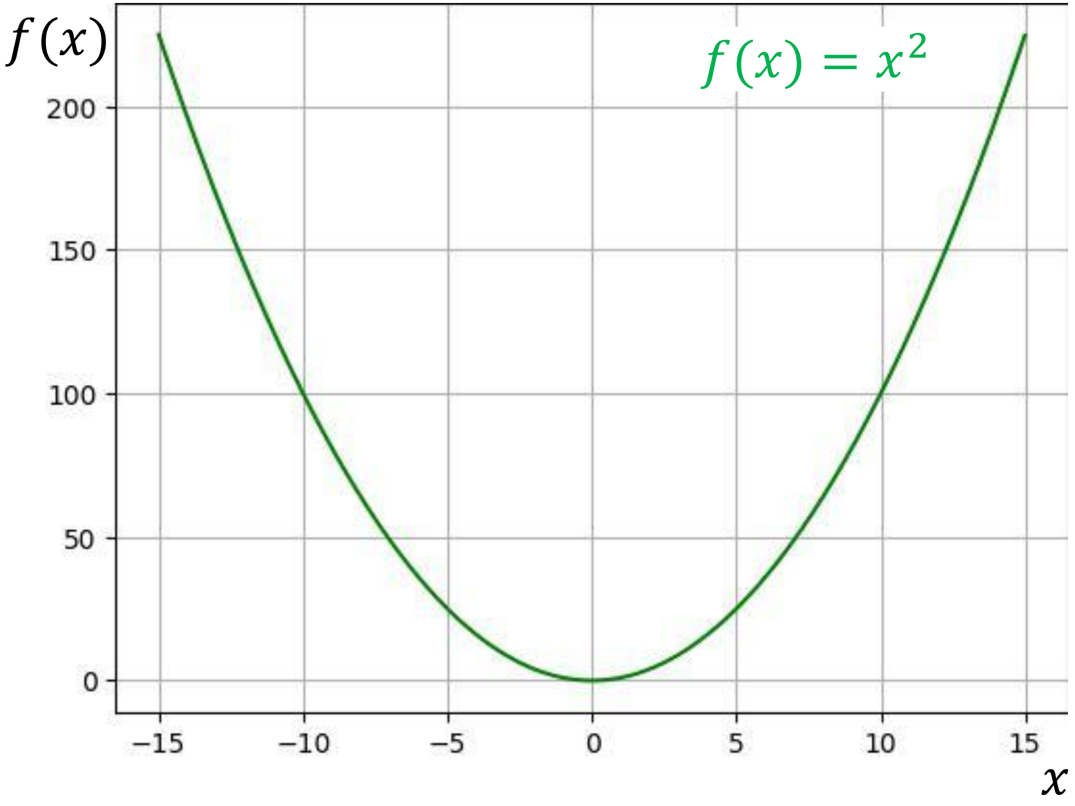
$b = 0.340$



Outline

Loss Functions

❖ **Mean Squared Error (MSE)**



$$f'(x) = 2x$$

One sample

$$L(\hat{y}, y) = (\hat{y} - y)^2$$

N samples

$$L(\hat{\boldsymbol{y}}, \boldsymbol{y}) = \frac{1}{N}\sum_{i=1}^{N}(\hat{y}_i - y_i)^2$$

I	y	$\hat{y}$	L
area	**price**	**prediction**	**error**
6.7	9.1	5.5	12.9
4.6	5.9	3.9	4.41
3.5	4.6	3.1	2.25
5.5	6.7	4.6	4.41

Loss Functions

❖ Mean Squared Error (MSE)

1) Pick a sample (x, y) from training data

2) Compute the output $\hat{y}$

$$\hat{y} = wx + b$$

3) Compute loss

$$L = (\hat{y} - y)^2$$

4) Compute derivative

$$\frac{\partial L}{\partial w} = 2x(\hat{y} - y) \qquad \frac{\partial L}{\partial b} = 2(\hat{y} - y)$$

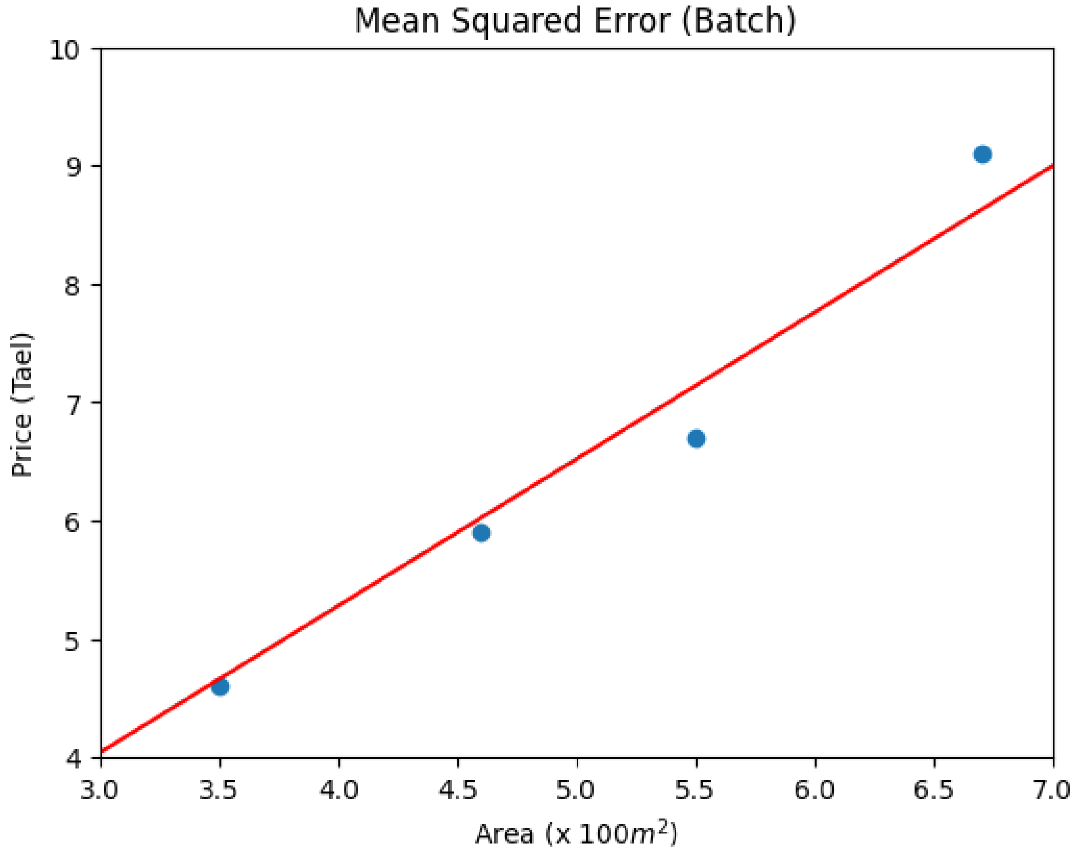
5) Update parameters

$$w = w - \eta \frac{\partial L}{\partial w} \qquad b = b - \eta \frac{\partial L}{\partial b}$$

$w = 1.207$

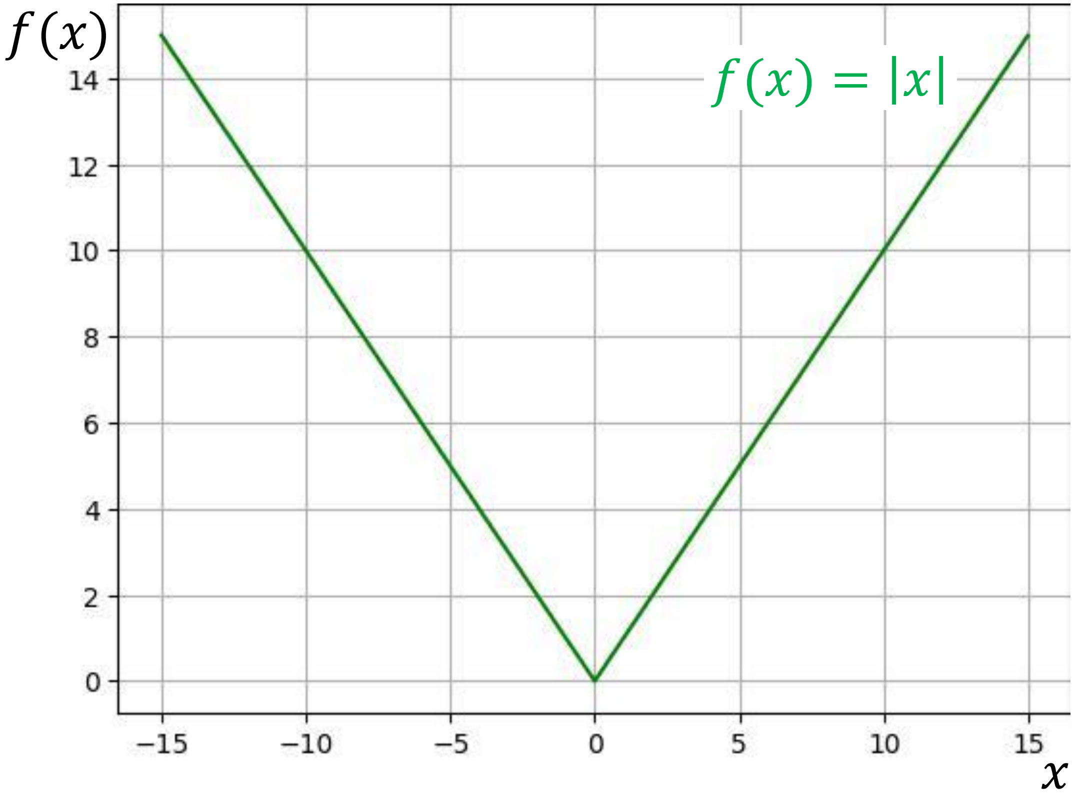
$b = 0.251$

Feature	Label
area	price
6.7	9.1
4.6	5.9
3.5	4.6
5.5	6.7



Loss Functions

❖ **Mean Absolute Error (MAE)**



$$f'(x) = \frac{x}{|x|} \quad \text{for } x \neq 0$$

One sample

$$L(\hat{y}, y) = |\hat{y} - y|$$

N samples

$$L(\hat{\boldsymbol{y}}, \boldsymbol{y}) = \frac{1}{N}\sum_{i=1}^{N} |\hat{y}_i - y_i|$$

I	y	$\hat{y}$	L
area	**price**	**prediction**	**error**
6.7	9.1	5.5	3.6
4.6	5.9	3.8	2.1
3.5	4.6	3.1	1.5
5.5	6.7	4.6	2.1

AI VIET NAM
@aivietnam.edu.vn

Loss Functions

Feature	Label
area	price
6.7	9.1
4.6	5.9
3.5	4.6
5.5	6.7

❖ Mean Absolute Error (MAE)

1) Pick a sample (x, y) from training data

2) Compute the output $\hat{y}$

$$\hat{y} = wx + b$$

3) Compute loss

$$L = |\hat{y} - y|$$

4) Compute derivative

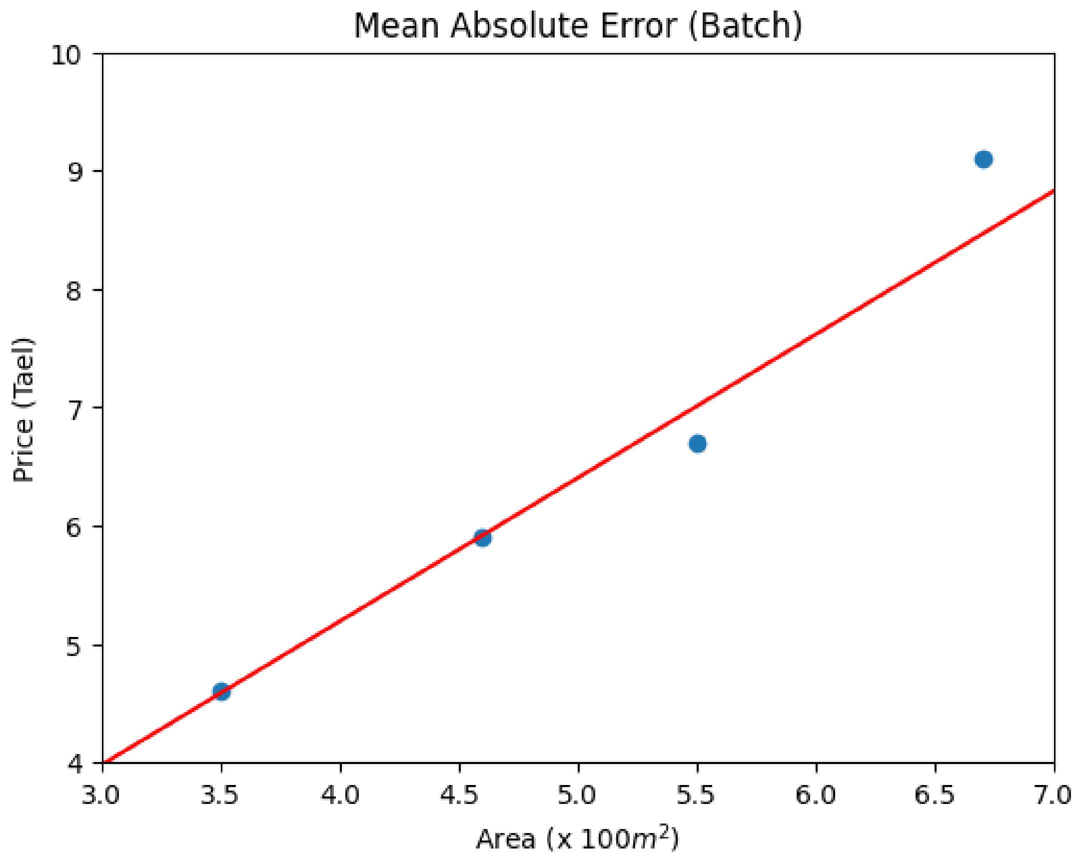
$$\frac{\partial L}{\partial w} = x\frac{(\hat{y} - y)}{|\hat{y} - y|} \qquad \frac{\partial L}{\partial b} = \frac{(\hat{y} - y)}{|\hat{y} - y|}$$

5) Update parameters

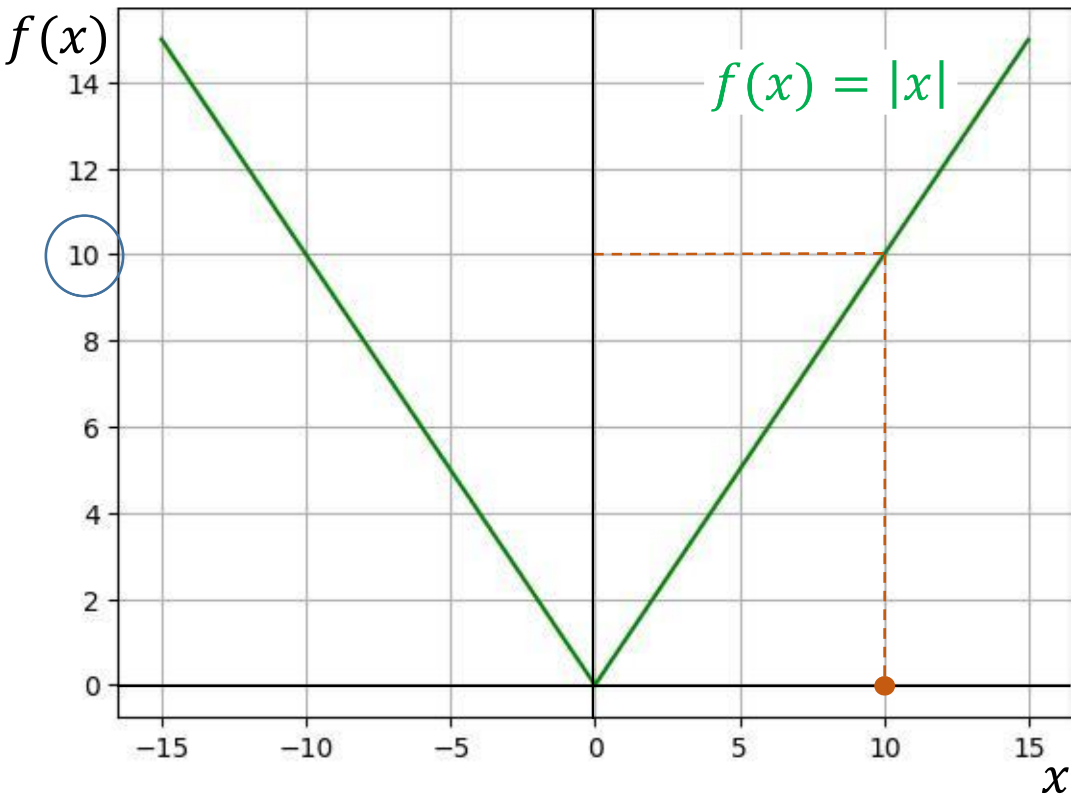
$$w = w - \eta\frac{\partial L}{\partial w} \qquad b = b - \eta\frac{\partial L}{\partial b}$$

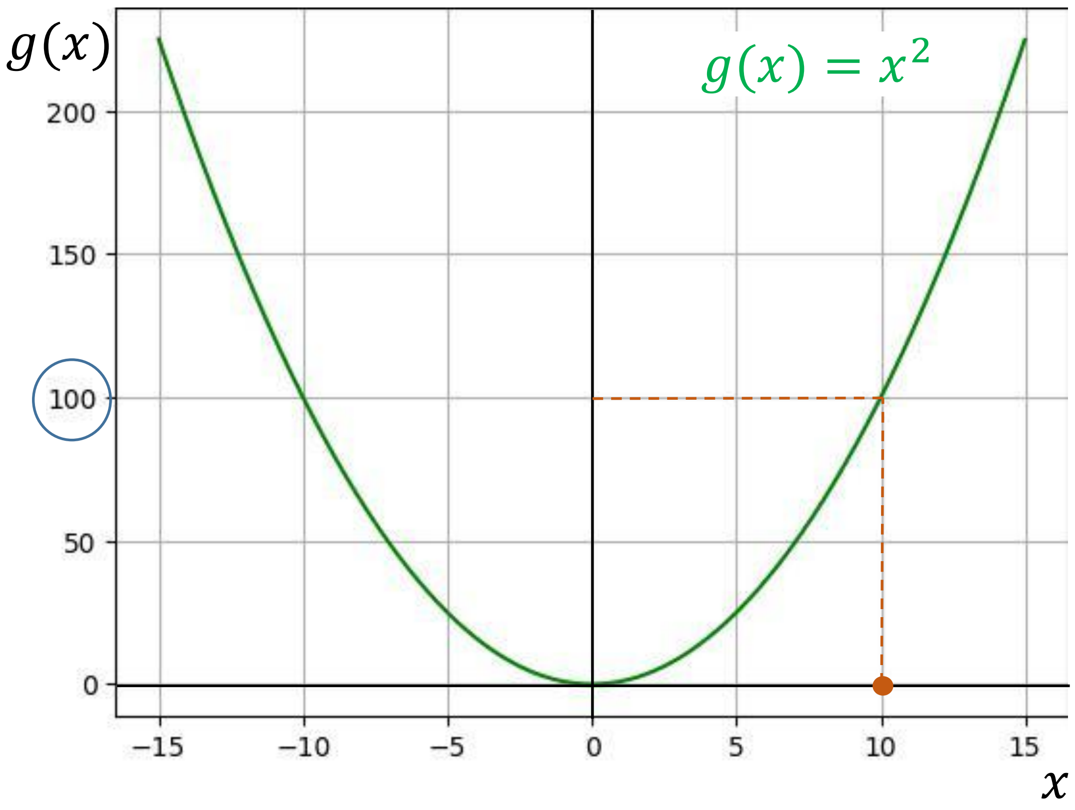
w = 1.185

b = 0.340



Quiz 4: The pros and cons of MSE and MAE when data contain outliers?





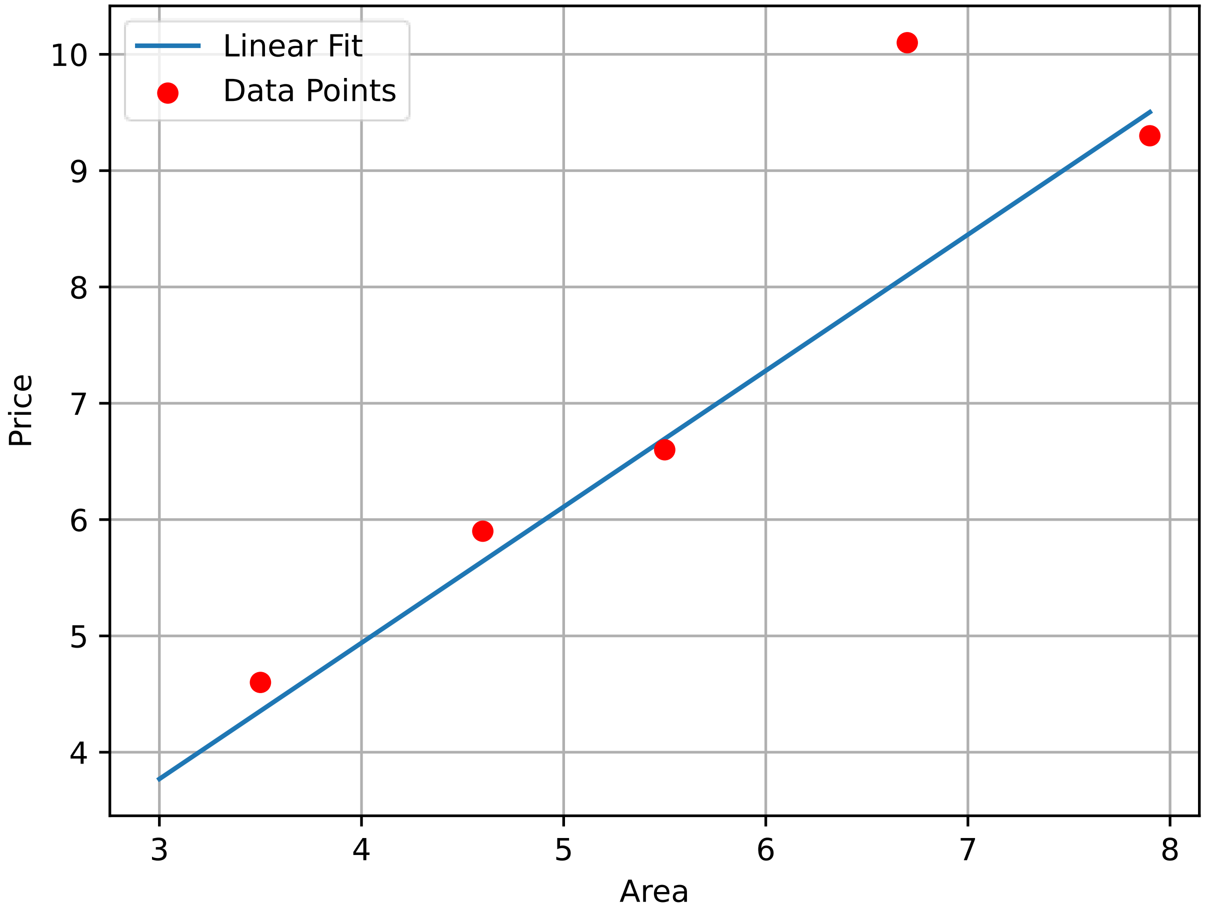
$g(10) \gg f(10)$

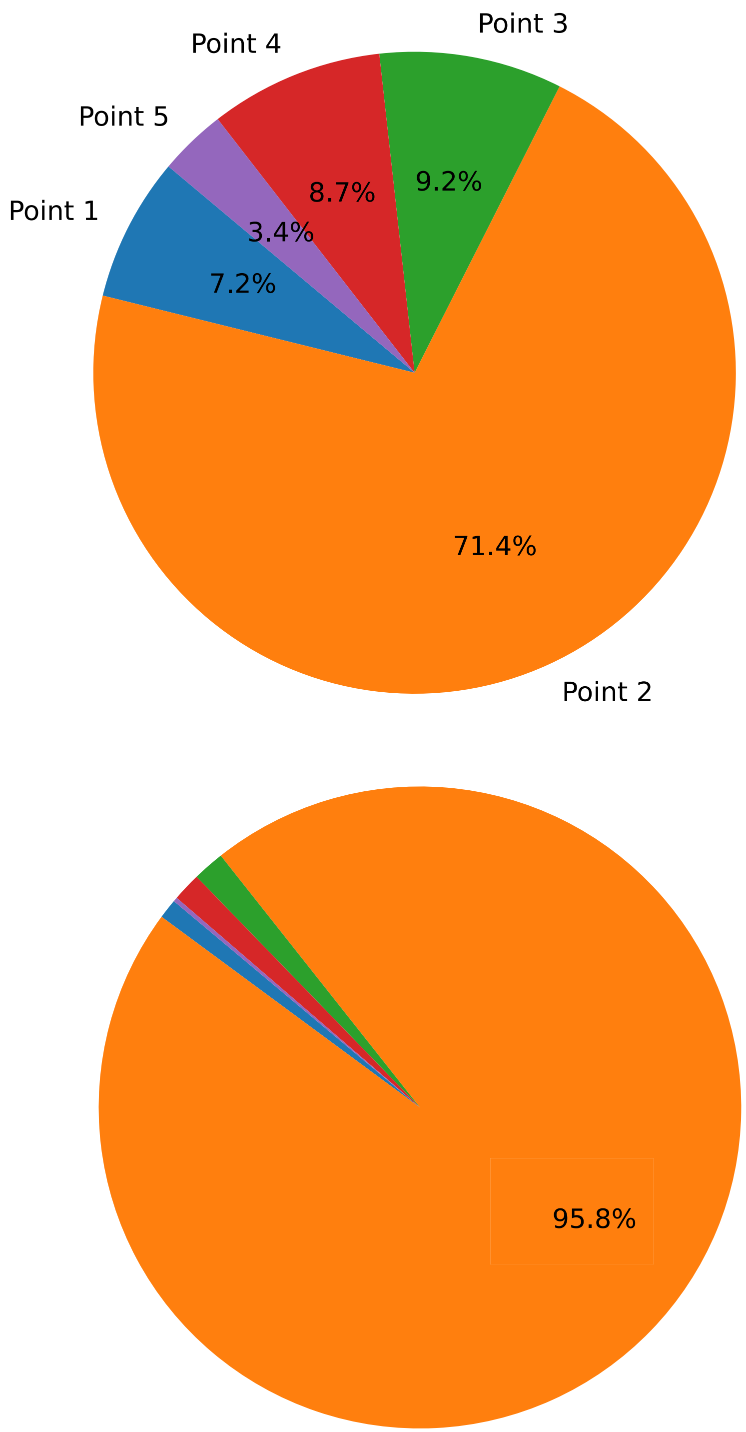
If $x = 10$ is an outlier, $g(10)$ has more negative effect

⇒ MAE is better to tolerate outliers

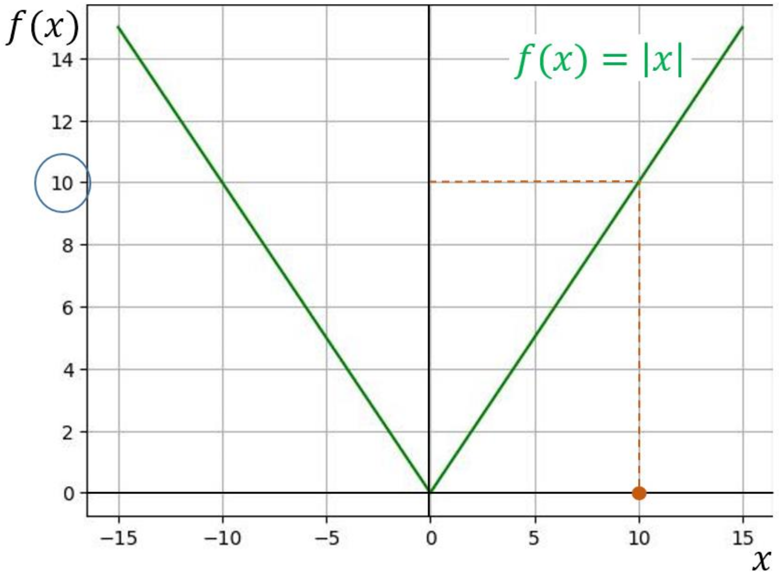
area	real price	pridicted price	absolute error	squared error
7.9	9.3	9.5	0.203	0.041
6.7	10.1	8.1	2.001	4004
4.6	5.9	5.6	0.258	0.066
3.5	4.6	4.4	0.245	0.060
5.5	6.6	6.7	0.095	0.009

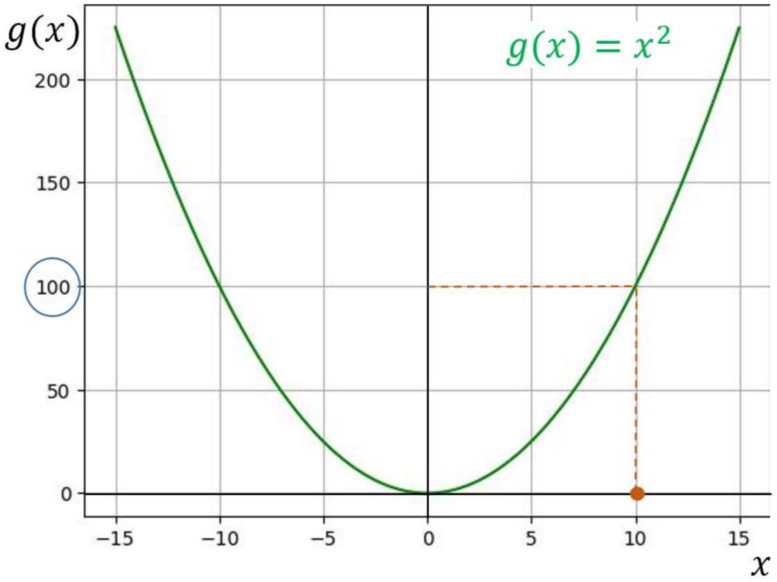
noise





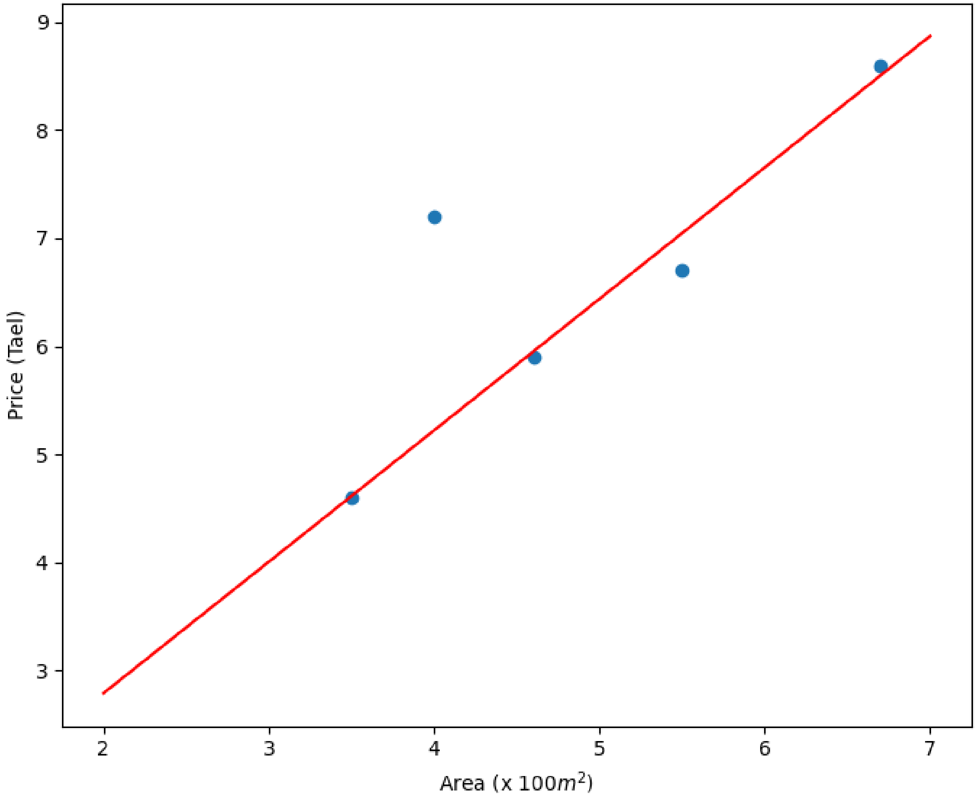
MAE



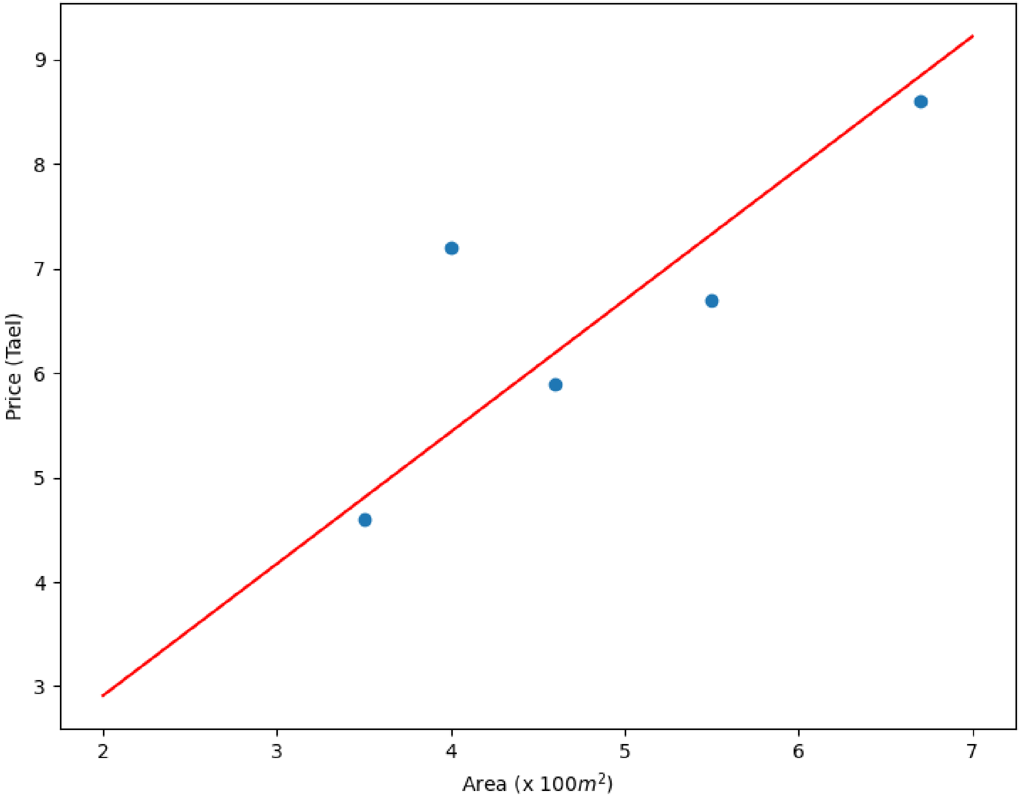


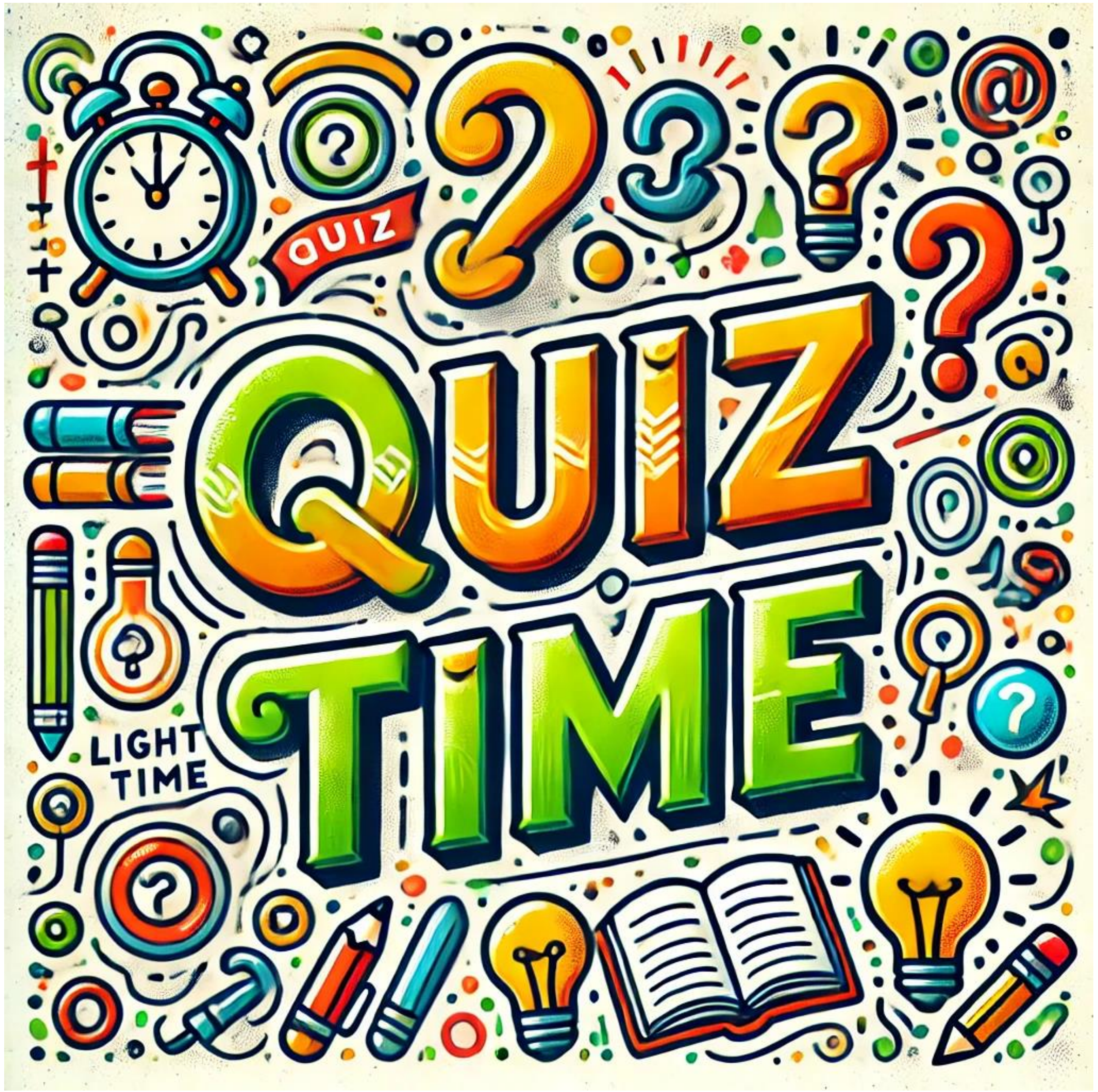
MSE

Mean Absolute Error

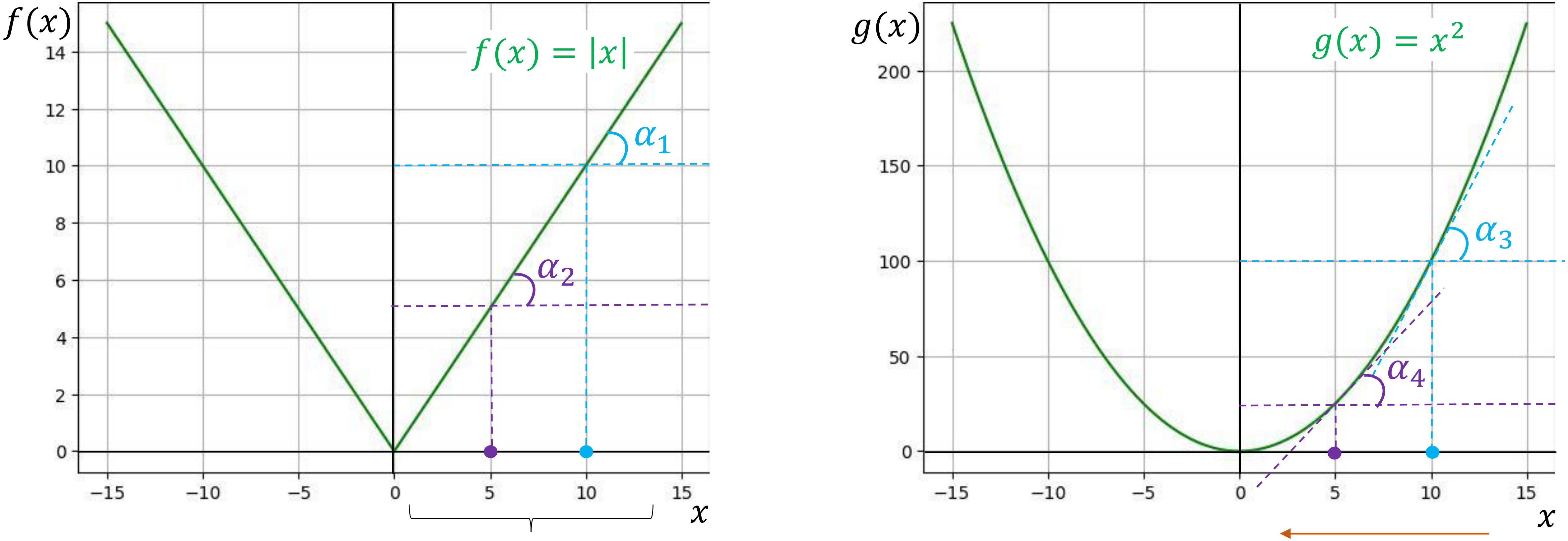


Mean Squared Error




QUIZ
QUIZ
TIME
LIGHT
TIME

Quiz 5: The pros and cons of MSE and MAE with a fixed learning rate η?



$\alpha_1 = \alpha_2$

$\alpha_3 > \alpha_4$

$\eta f'(x)$ values are constants

$\eta f'(x)$ values reduce

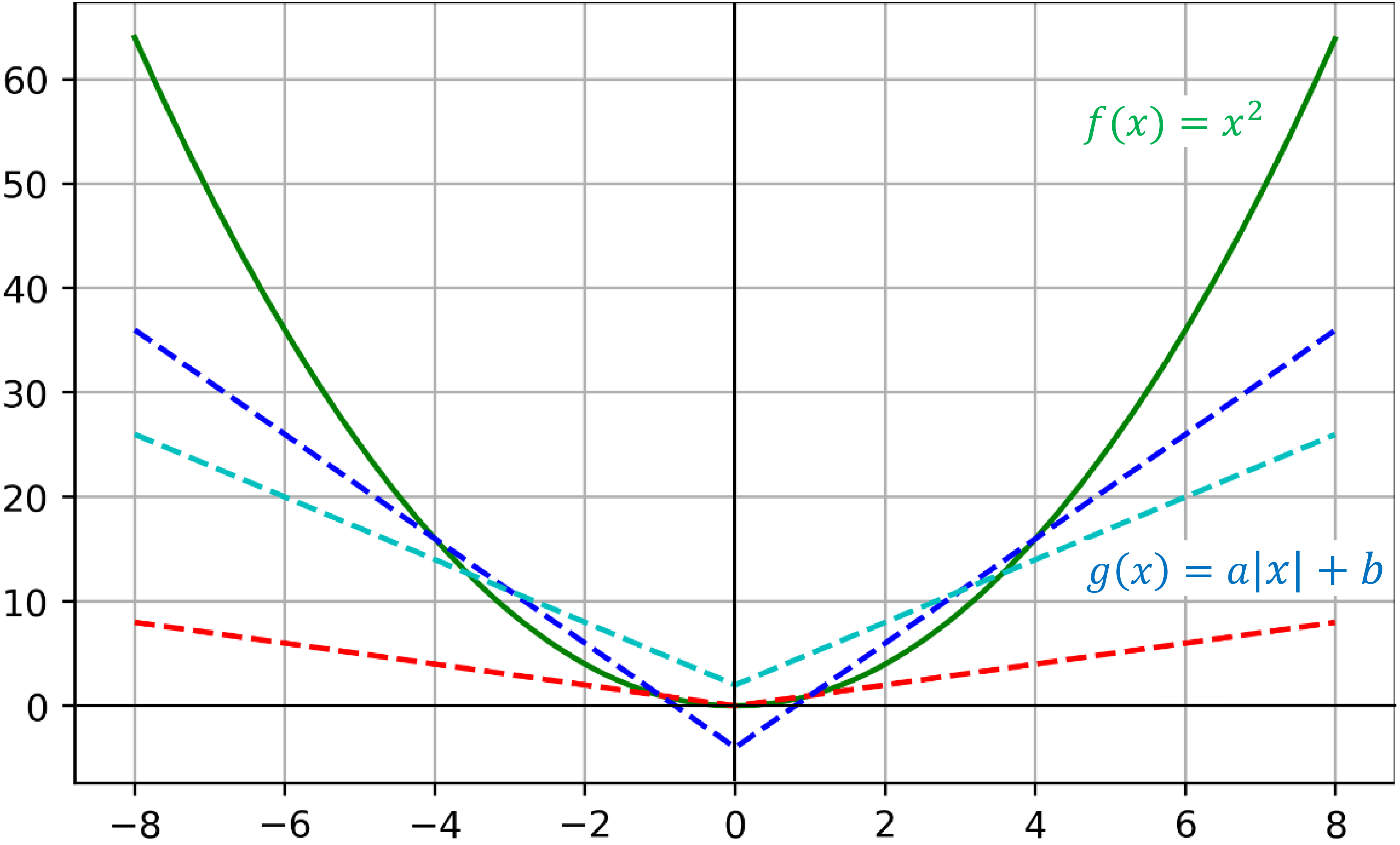
⇒ MSE is better when working with a fixed learning rate

Loss Functions

❖Requirements

Construct a function

Obtain the advantages of $f(x)$ and $g(x)$



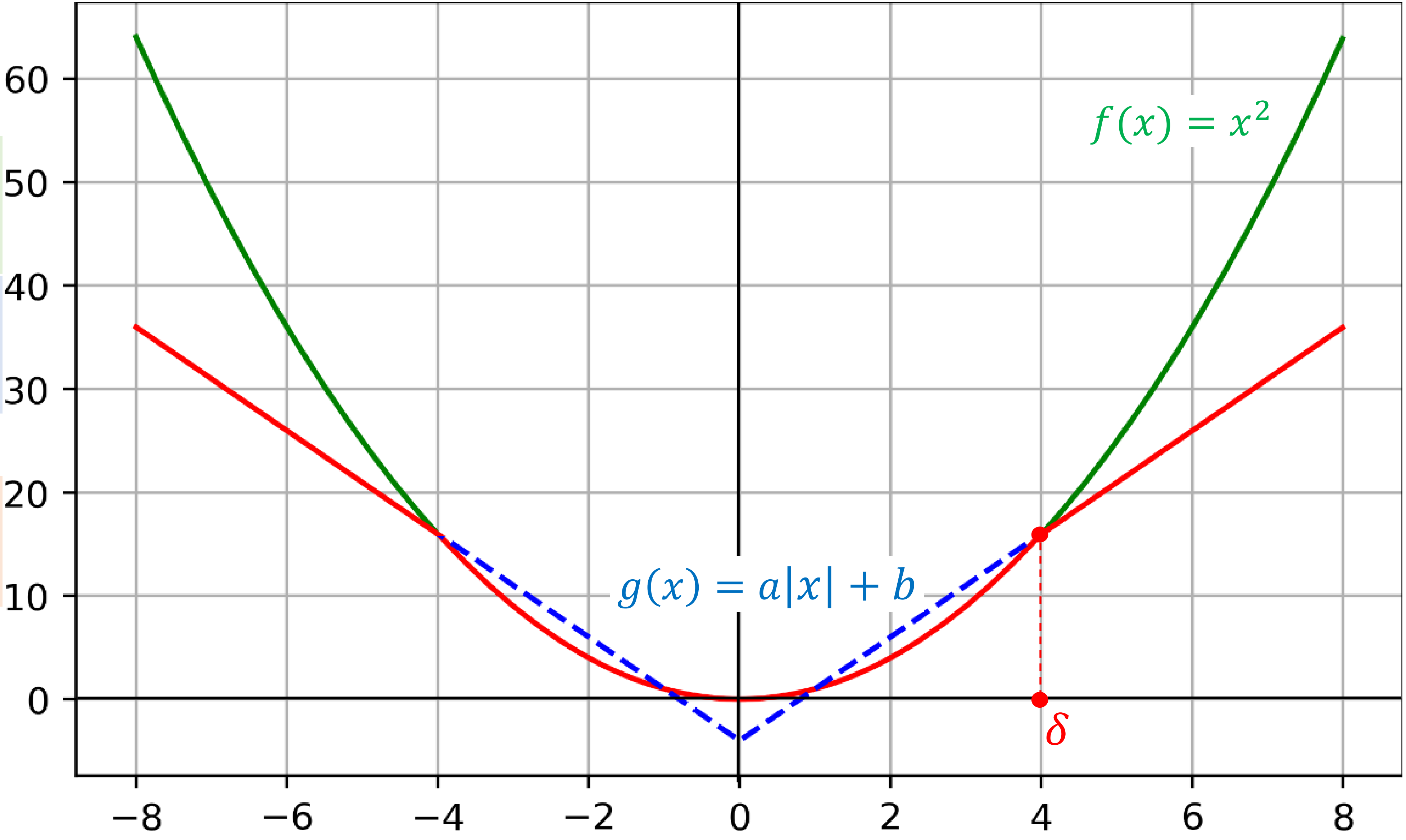
Loss Functions

❖ Requirements

Construct a function (the red line)

that is differentiable in the function domain

$k(x) = ?$



Loss Functions

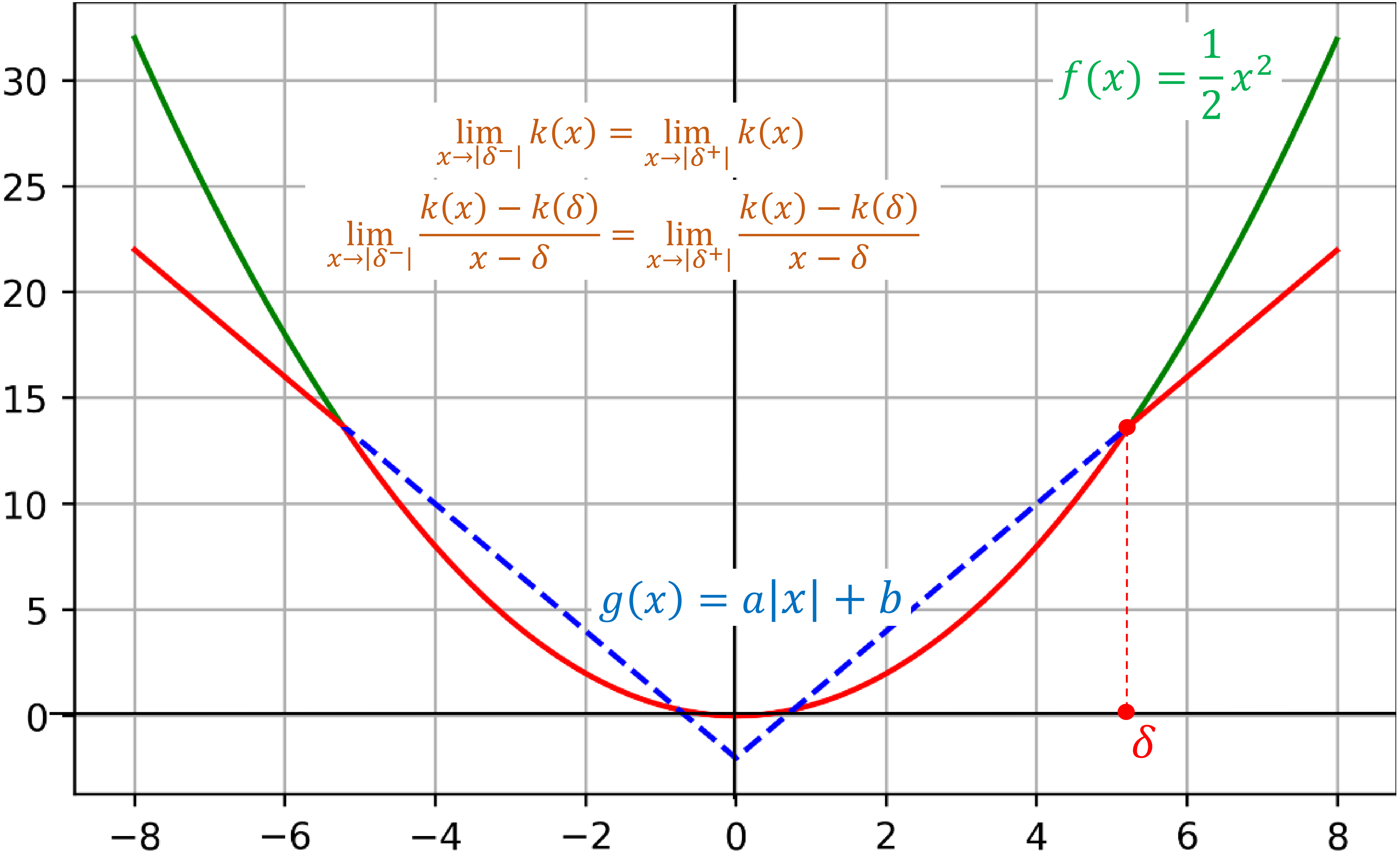
❖ Requirements

Construct a function
(the red line)

that is differentiable in the function domain

$k(x) = ?$

Change a little bit



Construct the function

$$k(x) = \begin{cases} \frac{1}{2}x^2 & \text{for } |x| \le \delta \\ a|x| + b & \text{otherwise} \end{cases}$$

To the function $k(x)$ be continuous at $x = \delta$

$$\lim_{x \to |\delta^-|} k(x) = \lim_{x \to |\delta^+|} k(x)$$

$$\Rightarrow \lim_{x \to |\delta^-|} \frac{1}{2}x^2 = \lim_{x \to |\delta^+|} a|x| + b$$

$$\Rightarrow \lim_{x \to |\delta^-|} \frac{1}{2}x^2 = \lim_{x \to |\delta^+|} a|x| + b$$

$$\Rightarrow \frac{1}{2}\delta^2 = a\delta + b \quad (1)$$

To the function $k(x)$ be differentiable at $x = \delta$

$$\lim_{x \to |\delta^-|} \frac{k(x) - k(\delta)}{x - \delta} = \lim_{x \to |\delta^+|} \frac{k(x) - k(\delta)}{x - \delta}$$

$$\Rightarrow \lim_{x \to |\delta^-|} \frac{\frac{1}{2}x^2 - \frac{1}{2}\delta^2}{x - \delta} = \lim_{x \to |\delta^+|} \frac{(a|x| + b) - (a\delta + b)}{x - \delta}$$

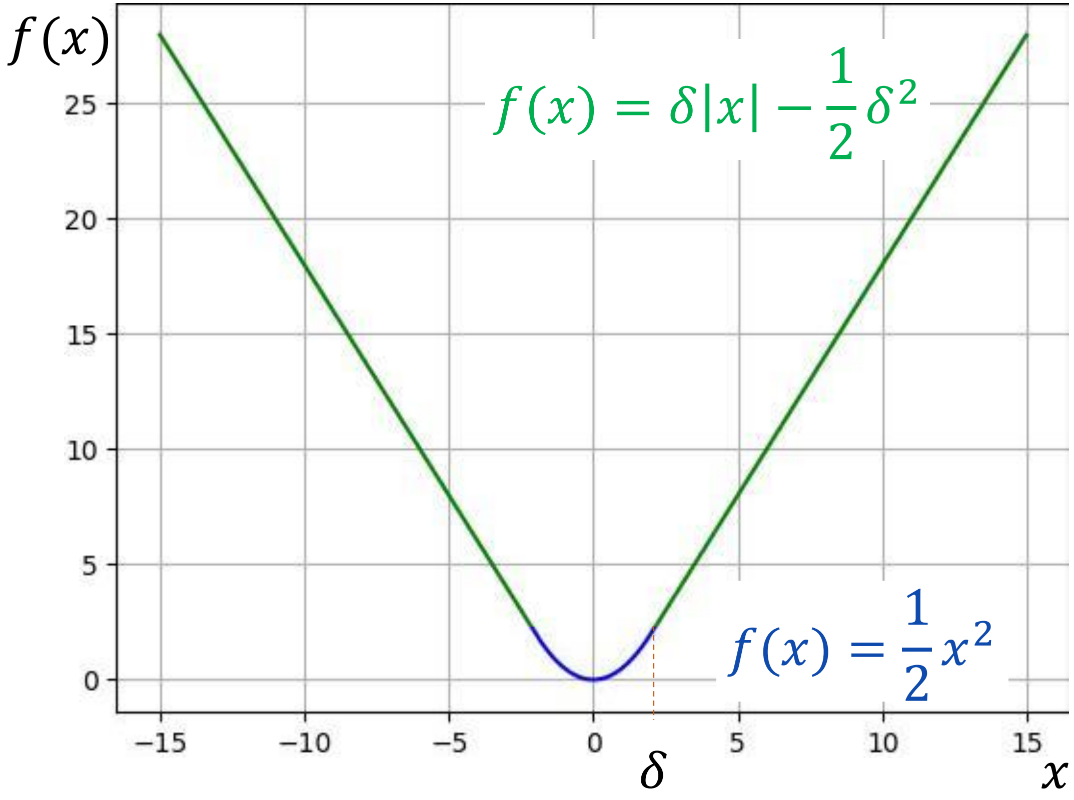
$$\Rightarrow \lim_{x \to |\delta^-|} \frac{1}{2}(x + \delta) = \lim_{x \to |\delta^+|} \frac{a(|x| - \delta)}{x - \delta}$$

$$\Rightarrow \delta = a$$

$$(1) \Rightarrow \frac{1}{2}\delta^2 = \delta\delta + b \quad \Rightarrow \quad b = -\frac{1}{2}\delta^2$$

Loss Functions

❖ Huber loss



One sample

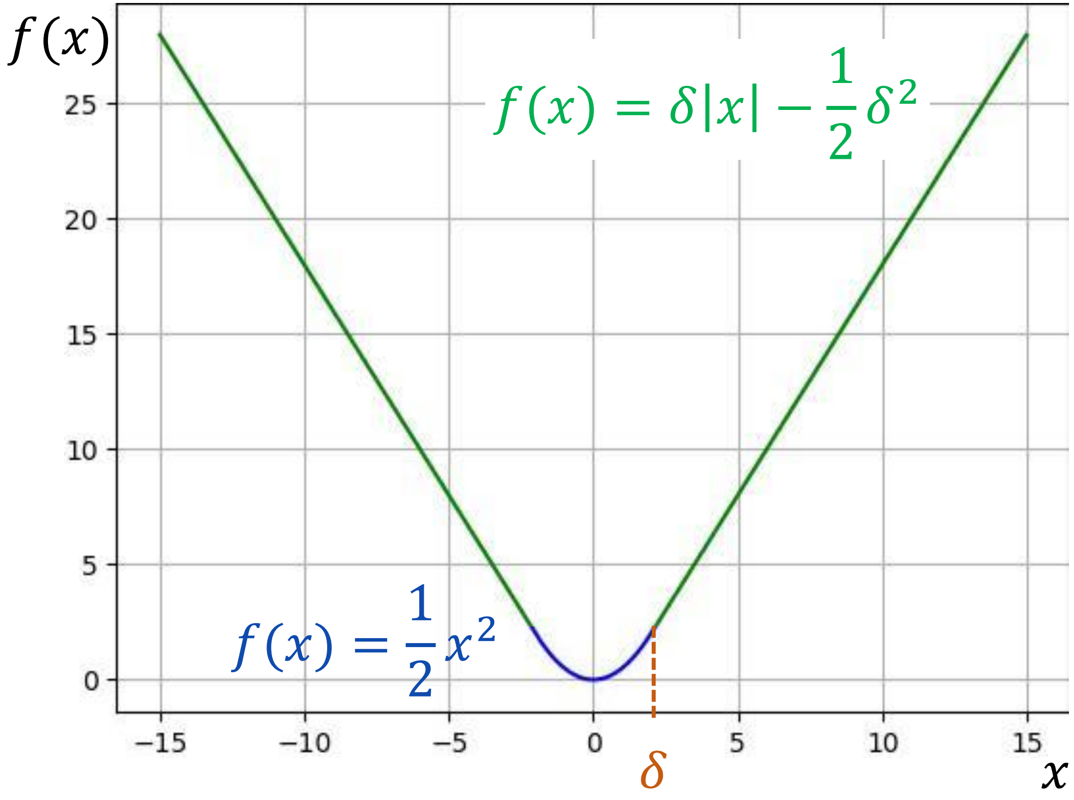
$$L(\hat{y}, y) = \begin{cases} \frac{1}{2}(\hat{y} - y)^2 & \text{for } |\hat{y} - y| \leq \delta \\ \delta|\hat{y} - y| - \frac{1}{2}\delta^2 & \text{otherwise} \end{cases}$$

I area	y price	$\hat{y}$ prediction	L error
6.7	9.1	5.5	5.2
4.6	5.9	3.8	2.2
3.5	4.6	3.1	1.1
5.5	6.7	4.6	2.2

$\delta = 2$

Loss Functions

❖ **Huber loss**



One sample

$$L(\hat{y}, y) = \begin{cases} \frac{1}{2}(\hat{y} - y)^2 & \text{for } |\hat{y} - y| \le \delta \\ \delta|\hat{y} - y| - \frac{1}{2}\delta^2 & \text{otherwise} \end{cases}$$

Compute derivative

$$L'_w = \begin{cases} x(\hat{y} - y) & \text{for } |\hat{y} - y| \le \delta \\ \delta x \frac{(\hat{y} - y)}{|\hat{y} - y|} & \text{otherwise} \end{cases}$$

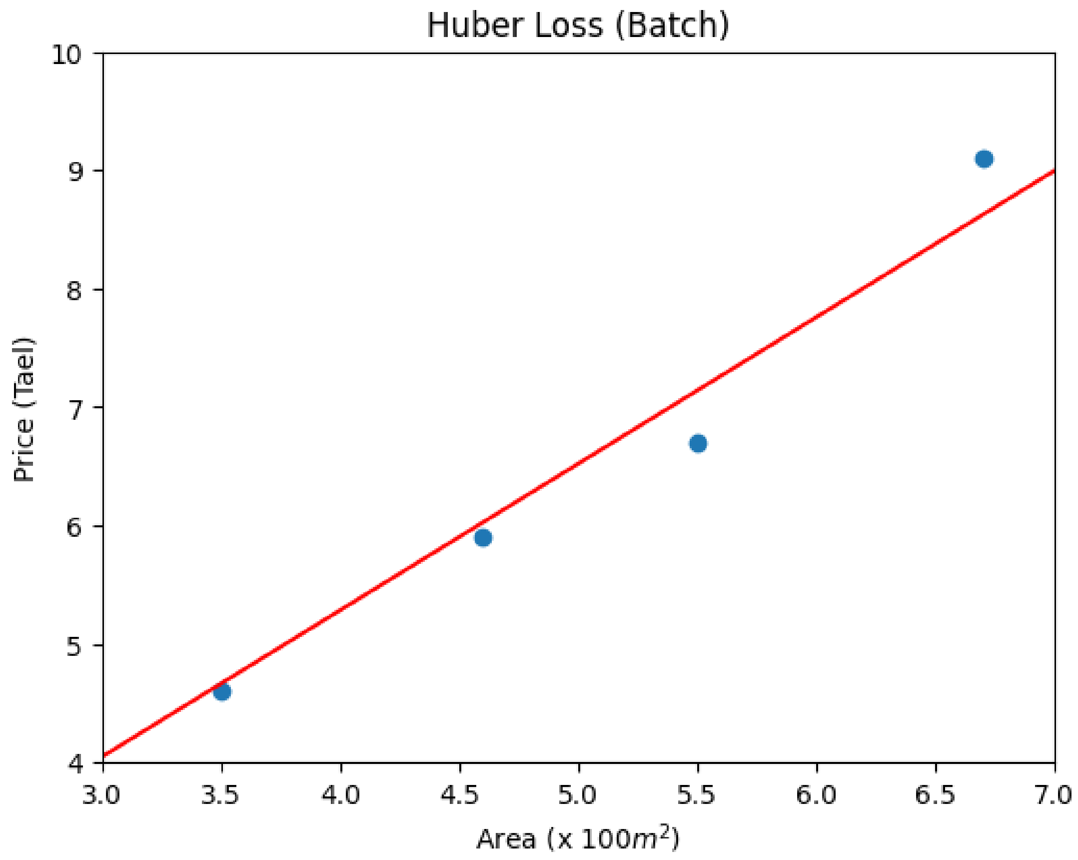
$$L'_b = \begin{cases} (\hat{y} - y) & \text{for } |\hat{y} - y| \le \delta \\ \delta \frac{(\hat{y} - y)}{|\hat{y} - y|} & \text{otherwise} \end{cases}$$

Loss Functions

❖ Huber loss

w = 2.201

b = 0.284



1) Pick a sample (x, y) from training data

2) Compute the output $\hat{y}$

$$\hat{y} = wx + b$$

3) Compute loss

$$L(\hat{y}, y) = \begin{cases} \frac{1}{2}(\hat{y} - y)^2 & \text{for } |\hat{y} - y| \leq \delta \\ \delta|\hat{y} - y| - \frac{1}{2}\delta^2 & \text{otherwise} \end{cases}$$

4) Compute derivative

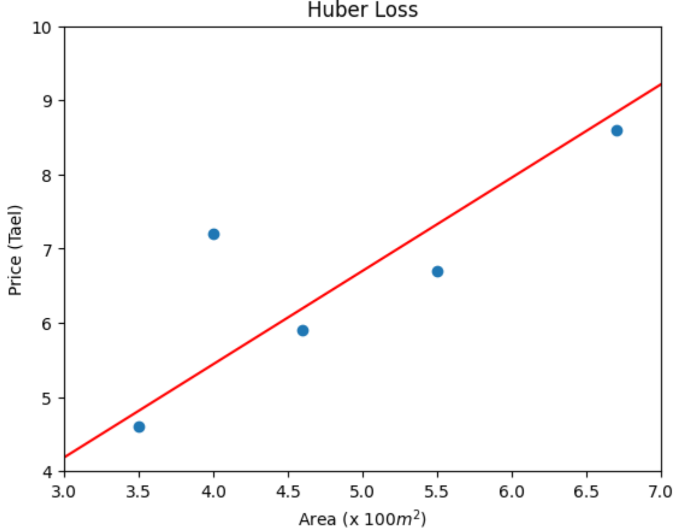
$$L'_w = \begin{cases} x(\hat{y} - y) & \text{for } |\hat{y} - y| \leq \delta \\ \delta x \frac{(\hat{y} - y)}{|\hat{y} - y|} & \text{otherwise} \end{cases}$$

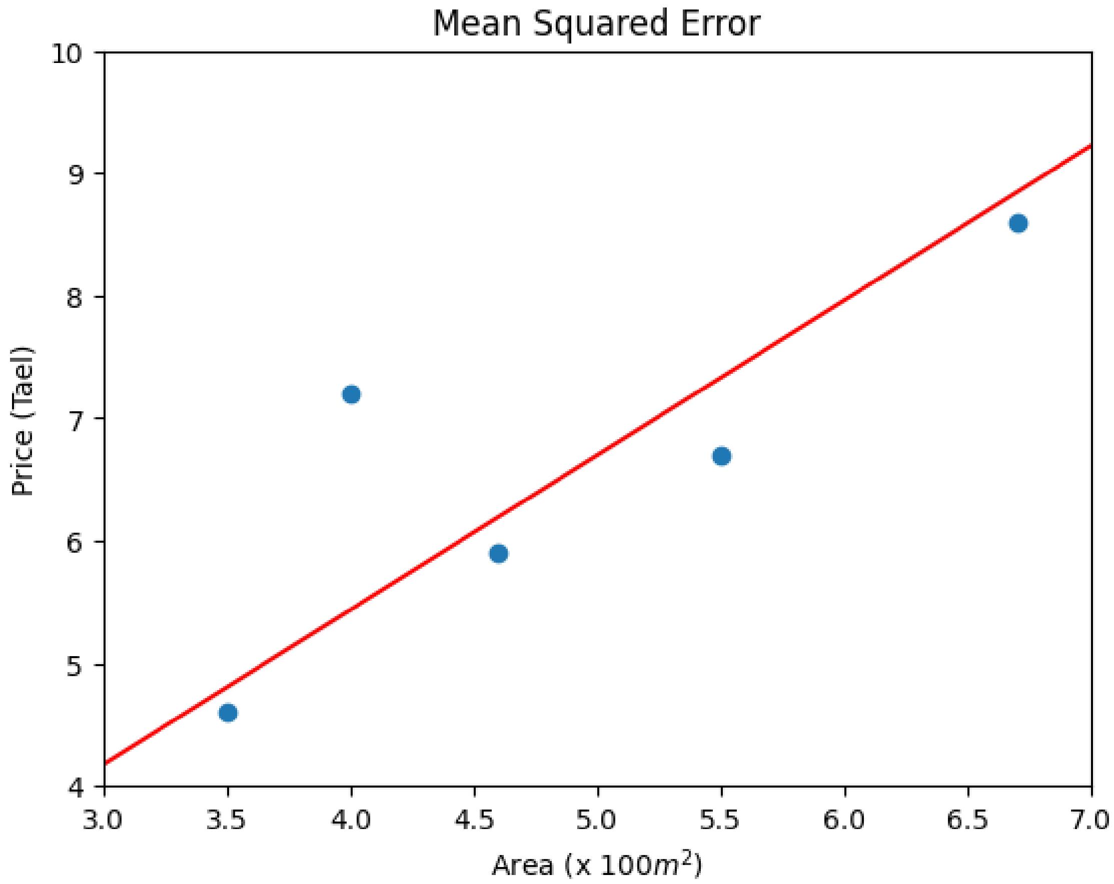
$$L'_b = \begin{cases} (\hat{y} - y) & \text{for } |\hat{y} - y| \leq \delta \\ \delta \frac{(\hat{y} - y)}{|\hat{y} - y|} & \text{otherwise} \end{cases}$$

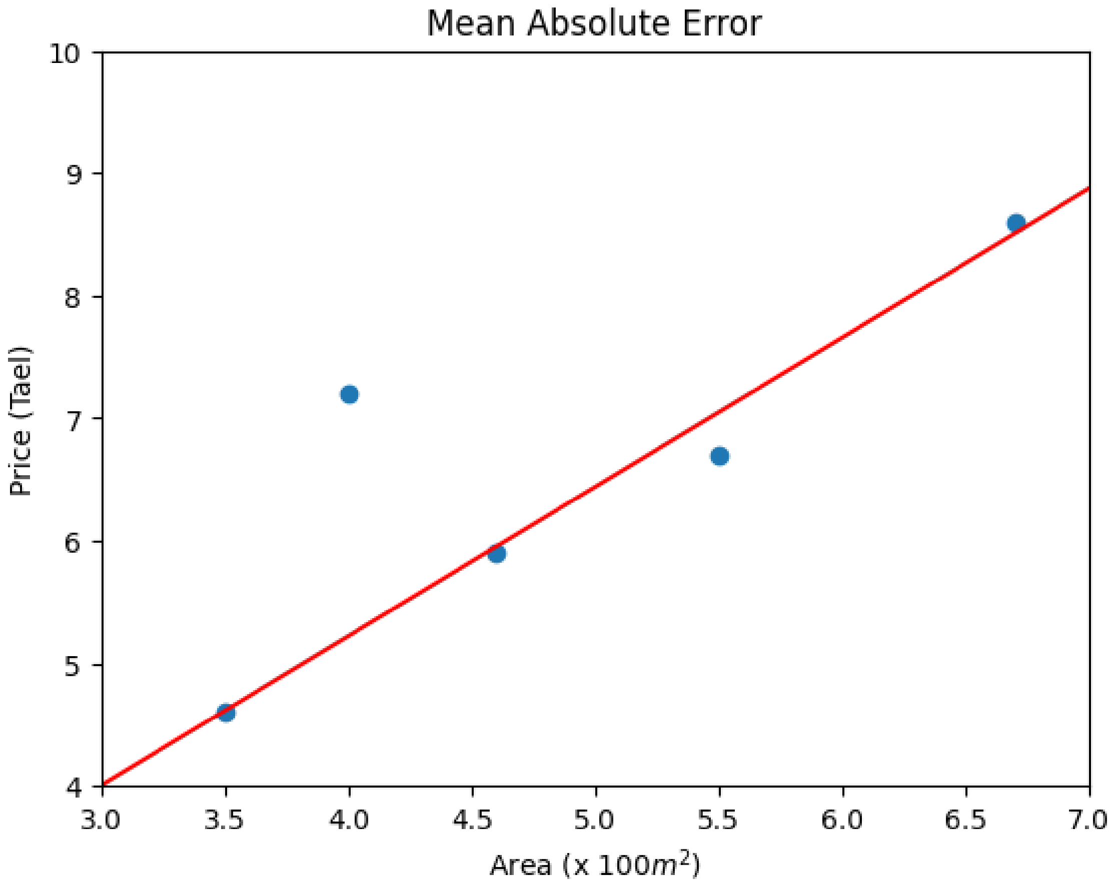
Loss Functions

- ❖ **Comparison (outliers)**
  - ❖ **Batch training**

area	price
6.7	8.6
4.6	5.9
3.5	4.6
5.5	6.7
4	7.2







Outline

SECTION 1

Variants of MSE

SECTION 2

Mean Absolute Error

SECTION 3

Huber Loss

SECTION 4

Data Normalization

Linear Regression

❖ General formula

Feature	Label
area	price
6.7	9.1
4.6	5.9
3.5	4.6
5.5	6.7

House price data

Features			Label
TV	Radio	Newspaper	Sales
230.1	37.8	69.2	22.1
44.5	39.3	45.1	10.4
17.2	45.9	69.3	12
151.5	41.3	58.5	16.5
180.8	10.8	58.4	17.9

Advertising data

Model: $\hat{y} = w_1 x_1 + b$

price = a $* \, area + b$

Model: $\hat{y} = w_1 x_1 + w_2 x_2 + w_3 x_3 + b$

Sale = $w_1 * TV + w_2 * Radio + w_3 * Newspaper + b$

Linear Regression

1) Pick a sample (x_1, x_2, x_3, y) from training data

2) Compute the output $\hat{y}$

$$\hat{y} = w_1 * TV + w_2 * R + w_3 * N + b$$

$$\hat{y} = w_1 * x_1 + w_2 * x_2 + w_3 * x_3 + b$$

3) Compute loss

$$L = (\hat{y} - y)^2$$

4) Compute derivative

$$\frac{\partial L}{\partial w_1} = 2x_1(\hat{y} - y) \qquad \frac{\partial L}{\partial w_3} = 2x_3(\hat{y} - y)$$

$$\frac{\partial L}{\partial w_2} = 2x_2(\hat{y} - y) \qquad \frac{\partial L}{\partial b} = 2(\hat{y} - y)$$

5) Update parameters

$$w_1 = w_1 - \eta \frac{\partial L}{\partial w_1} \qquad w_3 = w_3 - \eta \frac{\partial L}{\partial w_3}$$

$$w_2 = w_2 - \eta \frac{\partial L}{\partial w_2} \qquad b = b - \eta \frac{\partial L}{\partial b}$$

Features			Label
TV	Radio	Newspaper	Sales
230.1	37.8	69.2	22.1
44.5	39.3	45.1	10.4
17.2	45.9	69.3	12
151.5	41.3	58.5	16.5
180.8	10.8	58.4	17.9

Advertising data

Model

$$\text{Sale} = w_1 * TV + w_2 * Radio + w_3 * Newspaper + b$$

$$\hat{y} = w_1 x_1 + w_2 x_2 + w_3 x_3 + b$$

1) Pick a sample (x_1, x_2, x_3, y) from training data

2) Compute the output $\hat{y}$

$$\hat{y} = w_1 * TV + w_2 * R + w_3 * N + b$$

$$\hat{y} = w_1 * x_1 + w_2 * x_2 + w_3 * x_3 + b$$

3) Compute loss

$$L = (\hat{y} - y)^2$$

4) Compute derivative

$$\frac{\partial L}{\partial w_1} = 2x_1(\hat{y} - y) \qquad \frac{\partial L}{\partial w_3} = 2x_3(\hat{y} - y)$$

$$\frac{\partial L}{\partial w_2} = 2x_2(\hat{y} - y) \qquad \frac{\partial L}{\partial b} = 2(\hat{y} - y)$$

5) Update parameters

$$w_1 = w_1 - \eta \frac{\partial L}{\partial w_1} \qquad w_3 = w_3 - \eta \frac{\partial L}{\partial w_3}$$

$$w_2 = w_2 - \eta \frac{\partial L}{\partial w_2} \qquad b = b - \eta \frac{\partial L}{\partial b}$$

```python
# compute output and loss
def predict(x1, x2, x3, w1, w2, w3, b):
    return w1*x1 + w2*x2 + w3*x3 + b
def compute_loss(y_hat, y):
    return (y_hat - y)**2

# compute gradient
def compute_gradient_wi(xi, y, y_hat):
    dl_dwi = 2*xi*(y_hat-y)
    return dl_dwi
def compute_gradient_b(y, y_hat):
    dl_db = 2*(y_hat-y)
    return dl_db

# update weights
def update_weight_wi(wi, dl_dwi, lr):
    wi = wi - lr*dl_dwi
    return wi
def update_weight_b(b, dl_db, lr):
    b  = b - lr*dl_db
    return b
```

Features			Label
TV	Radio	Newspaper	Sales
230.1	37.8	69.2	22.1
44.5	39.3	45.1	10.4
17.2	45.9	69.3	12
151.5	41.3	58.5	16.5
180.8	10.8	58.4	17.9

```python
def initialize_params():
    w1 = random.gauss(mu=0.0, sigma=0.01)
    w2 = random.gauss(mu=0.0, sigma=0.01)
    w3 = random.gauss(mu=0.0, sigma=0.01)
    b  = 0

    return w1, w2, w3, b

# initialize model's parameters
w1, w2, w3, b = initialize_params()
print(w1, w2, w3, b)
```

```
0.01609506469549467 0.00607778501208891 0.0023344573891806507 0
```

```python
import numpy as np
import random

def get_column(data, index):
    result = [row[index] for row in data]
    return result

data = np.genfromtxt('advertising.csv',
                     delimiter=',',
                     skip_header=1).tolist()

# get tv (index=0)
tv_data = get_column(data, 0)

# get radio (index=1)
radio_data = get_column(data, 1)

# get newspaper (index=2)
newspaper_data = get_column(data, 2)

# get sales (index=0)
sales_data = get_column(data, 3)
```

Unnormalized data $\eta = 10^{-5}$

Features			Label
TV	Radio	Newspaper	Sales
230.1	37.8	69.2	22.1
44.5	39.3	45.1	10.4
17.2	45.9	69.3	12
151.5	41.3	58.5	16.5
180.8	10.8	58.4	17.9

```
0.01609506469549467 0.00607778501208891 0.0023344573891806507 0
```

```
x1: 230.1
x2: 37.8
x1: 69.2
y: 22.1
y_hat: 4.0947591112215855

dl_dw1: -8286.011857015827
dl_dw2: -1361.1962111916482
dl_dw3: -2491.925339006933
dl_db: -36.01048177755683

w1: 0.09895518326565295
w2: 0.019689747124005393
w3: 0.027253710779249984
b: 0.0003601048177755684
```

```python
for epoch in range(epoch_max):
    for i in range(N):
        # get a sample
        x1 = tv_data[i]
        x2 = radio_data[i]
        x3 = newspaper_data[i]
        y  = sales_data[i]

        # compute output
        y_hat = predict(x1, x2, x3, w1, w2, w3, b)

        # compute gradient w1, w2, w3, b
        dl_dw1 = compute_gradient_wi(x1, y, y_hat)
        dl_dw2 = compute_gradient_wi(x2, y, y_hat)
        dl_dw3 = compute_gradient_wi(x3, y, y_hat)
        dl_db  = compute_gradient_b(y, y_hat)

        # update parameters
        w1 = update_weight_wi(w1, dl_dw1, lr)
        w2 = update_weight_wi(w2, dl_dw2, lr)
        w3 = update_weight_wi(w3, dl_dw3, lr)
        b  = update_weight_b(b, dl_db, lr)
```

Normalized data $\eta = 10^{-2}$

Features **Label**

TV	Radio	Newspaper	Sales
230.1	37.8	69.2	22.1
44.5	39.3	45.1	10.4
17.2	45.9	69.3	12
151.5	41.3	58.5	16.5
180.8	10.8	58.4	17.9

```
0.01609506469549467 0.00607778501208891 0.0023344573891806507 0
```

```
x1: 0.5504267881241568
x2: -0.09835863697705782
x1: 0.007579284750337614
y: 22.1
y_hat: 0.008279045632653116

dl_dw1: -24.319750018095103
dl_dw2: 4.345823123098159
dl_dw3: -0.33487888747630074
dl_db: -44.1834419087347

w1: 0.2592925648764457
w2: -0.037380446218892686
w3: 0.005683246263943658
b: 0.441834419087347
```

$$x = \frac{x - x_{mean}}{x_{max} - x_{min}}$$

```
for epoch in range(epoch_max):
    for i in range(N):
        # get a sample
        x1 = tv_data[i]
        x2 = radio_data[i]
        x3 = newspaper_data[i]
        y  = sales_data[i]

        # compute output
        y_hat = predict(x1, x2, x3, w1, w2, w3, b)

        # compute gradient w1, w2, w3, b
        dl_dw1 = compute_gradient_wi(x1, y, y_hat)
        dl_dw2 = compute_gradient_wi(x2, y, y_hat)
        dl_dw3 = compute_gradient_wi(x3, y, y_hat)
        dl_db  = compute_gradient_b(y, y_hat)

        # update parameters
        w1 = update_weight_wi(w1, dl_dw1, lr)
        w2 = update_weight_wi(w2, dl_dw2, lr)
        w3 = update_weight_wi(w3, dl_dw3, lr)
        b  = update_weight_b(b, dl_db, lr)
```

Summary