

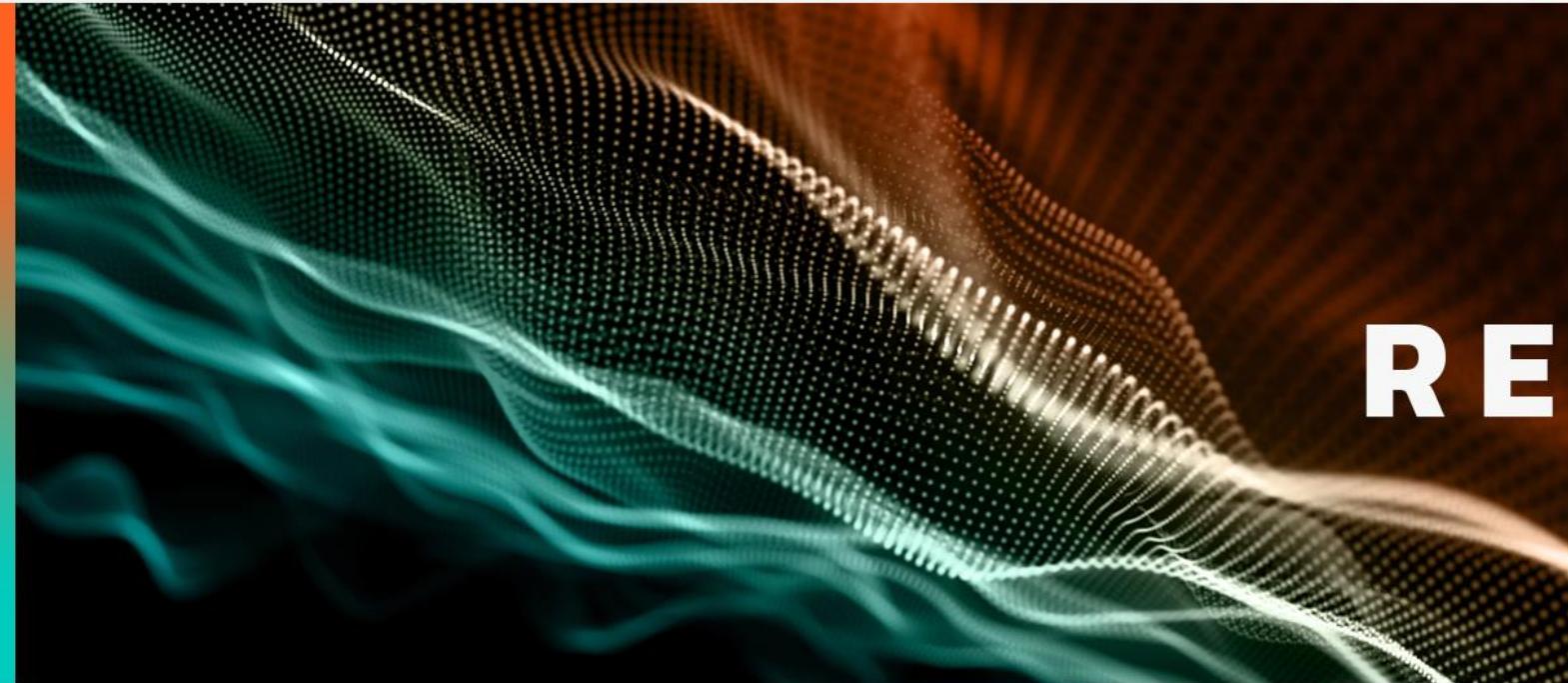
TAUTOLOGY  
INNOVATION  
SCHOOL



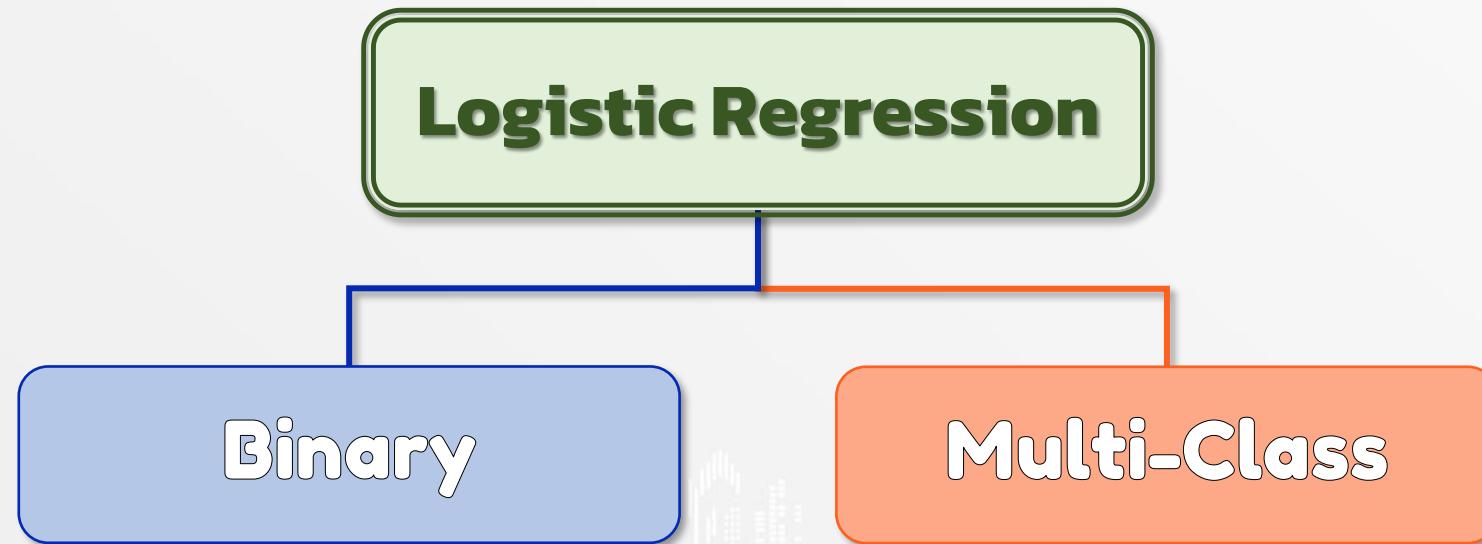
# LOGISTIC REGRESSION

BY TAUTOLOGY

LOGISTIC REGRESSION



# What is Logistic Regression?

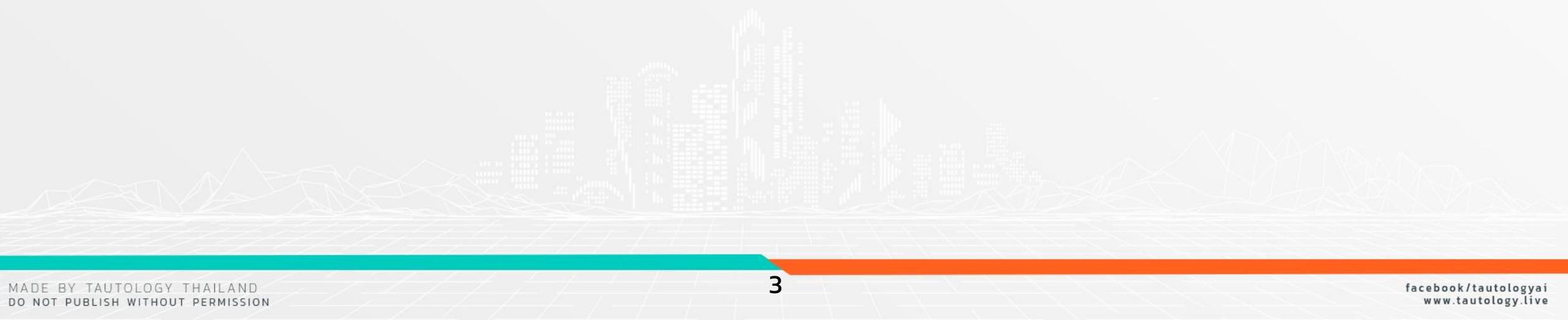


# Logistic Regression

Logistic Regression  
(Binary)

Logistic Regression  
(Multi-Class)

Workshop



# Logistic Regression (Binary)



# Introduction

# Introduction

What is Logistic  
Regression?

Data for Logistic  
Regression  
(Binary)

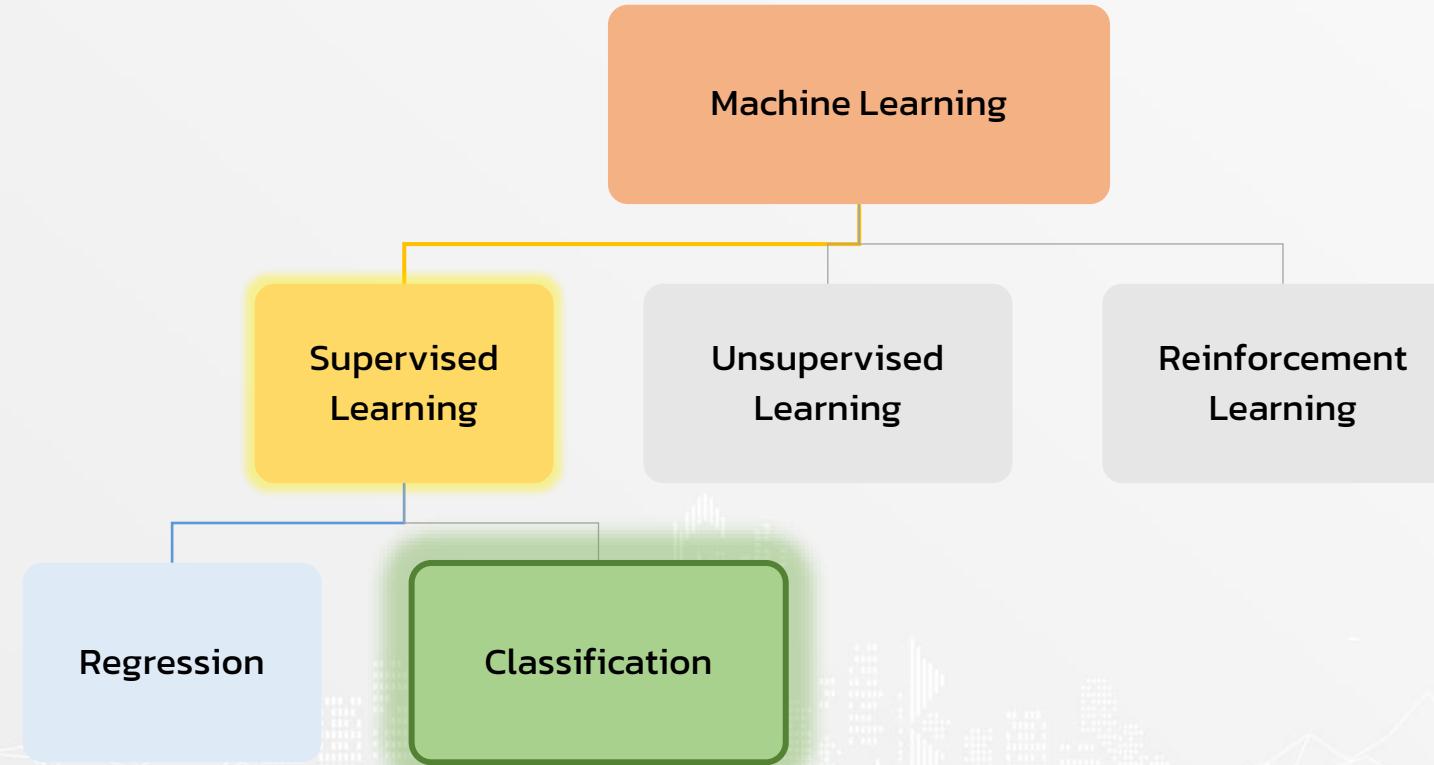
Pros & Cons

Real World  
Application

# What is Logistic Regression?

**Logistic Regression** เป็นหนึ่งใน algorithm ประเภท supervised learning ที่ใช้สำหรับแก้ปัญหา **classification** โดยมีหลักการทำงานคือ จำแนกประเภทของข้อมูล ด้วยการสร้างสมการเชิงเส้น

# What is Logistic Regression?



# What is Logistic Regression?

สมการคณิตศาสตร์ของ **Logistic Regression**

$$z = w_0 + w_1x_1 + w_2x_2 + \cdots + w_px_p$$

$$\hat{y} = \sigma(z) \quad (\text{sigmoid})$$

- โดยที่
- ◆  $\hat{y}$  คือ ตัว預測 (predicted target)
  - ◆  $z$  คือ สมการเชิงเส้น (linear function)
  - ◆  $x_1, x_2, x_3, \dots, x_p$  คือ ตัวแปรตัว (feature)
  - ◆  $w_0, w_1, w_2, \dots, w_p$  คือ สัมประสิทธิ์ (coefficient)

# What is Logistic Regression?

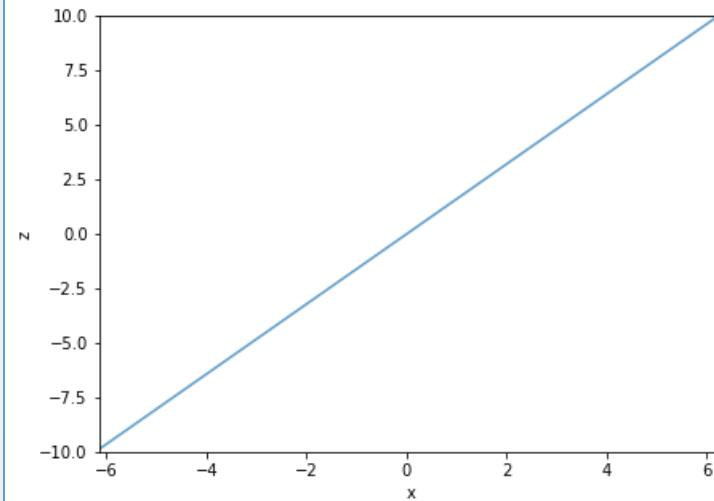
สมการคณิตศาสตร์ของ **Sigmoid Function**  $\sigma : (-\infty, \infty) \rightarrow (0, 1)$

$$\hat{y} = \sigma(z) = \frac{1}{1 + e^{-z}} = \frac{e^z}{e^z + 1}$$

- โดยที่
- ◆  $\hat{y}$  คือ ตัว預測 (predicted target)
  - ◆  $z$  คือ สมการเชิงเส้น (linear function)
  - ◆  $e$  คือ ค่าคงที่ และมีค่าเท่ากับ 2.718281828...

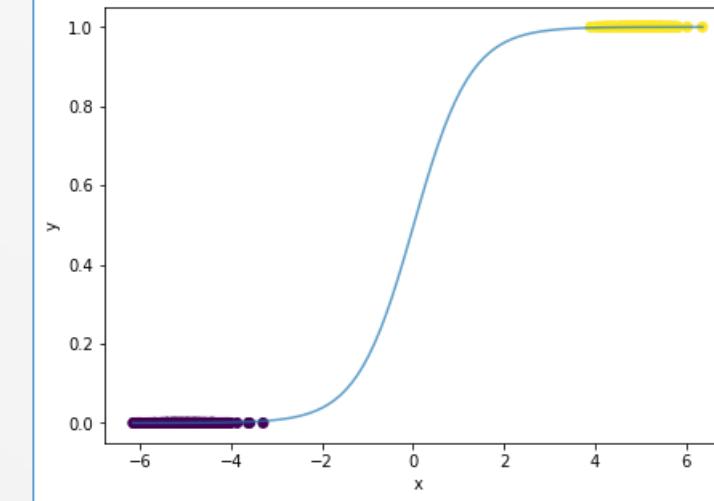
# What is Logistic Regression?

Linear Function



$$z = w_0 + w_1 x_1$$

Logistic Regression

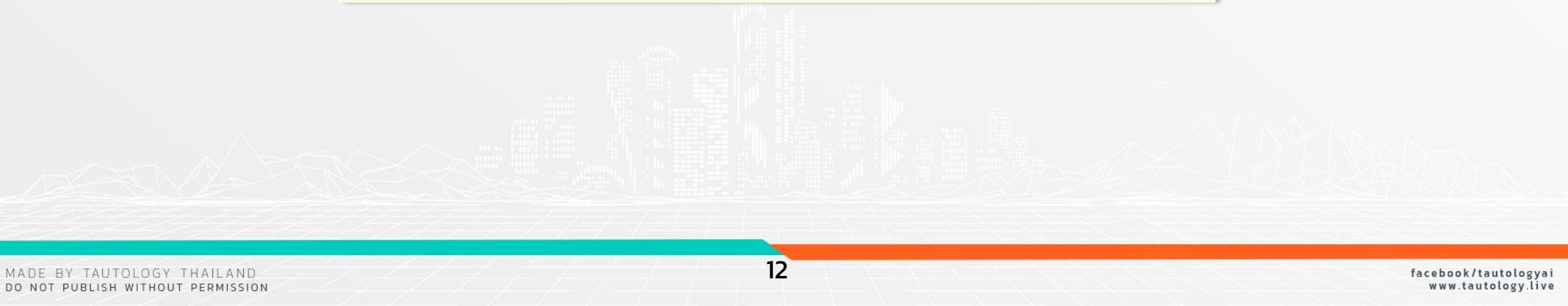


$$\hat{y} = \sigma(z)$$

# What is Logistic Regression?

## Binary

**Binary** คือการ map ค่าไปยัง **(0, 1)** เพื่อจำแนก  
ข้อมูล 2 class ด้วย function **sigmoid**



# What is Logistic Regression?

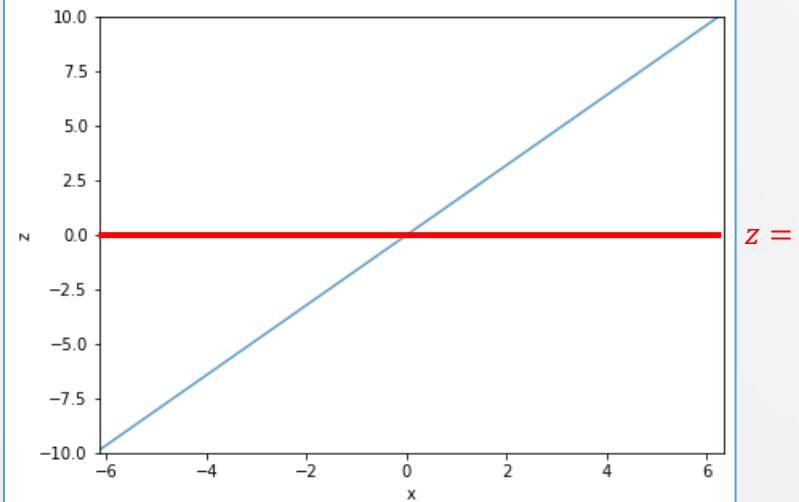
Why 0, 1?

# What is Logistic Regression?

0, 1 เป็นตัวแทนของ class ก็ง 2 กี่เราต้องการ classification เช่น เป็นโรค/ไม่เป็นโรค,  
ปลอยกู้/ไม่ปลอยกู้, Fake/True เป็นต้น

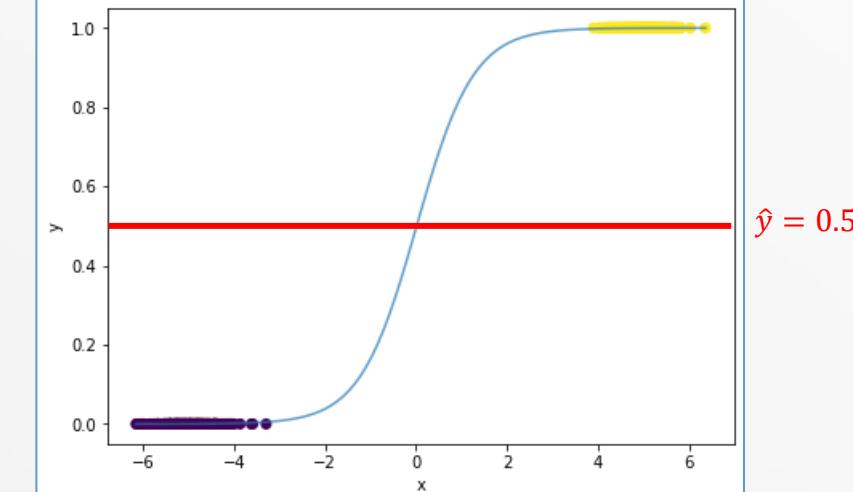
# What is Logistic Regression?

Linear Function



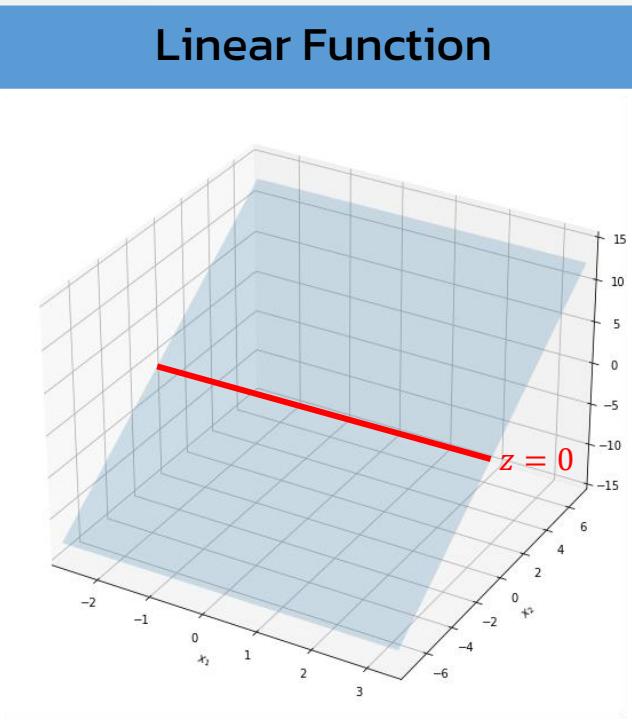
$$z = w_0 + w_1 x_1$$

Logistic Regression

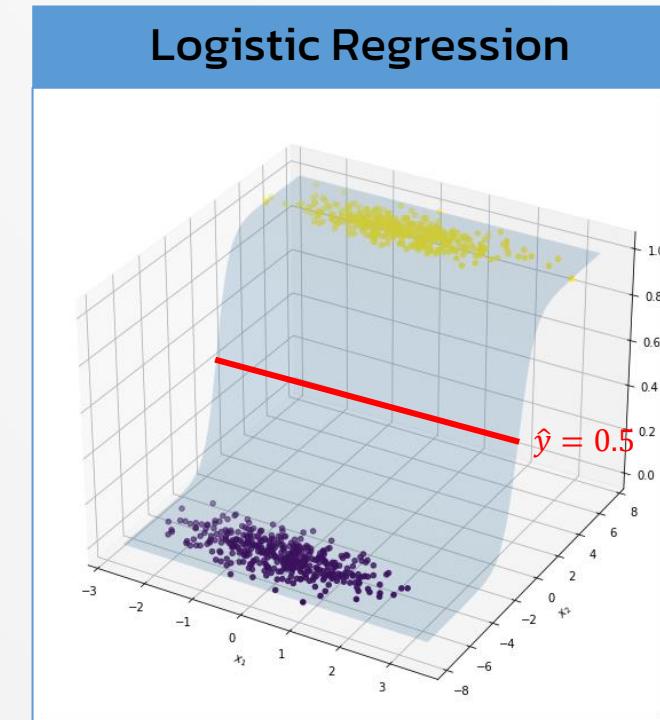


$$\hat{y} = \sigma(z)$$

# What is Logistic Regression?



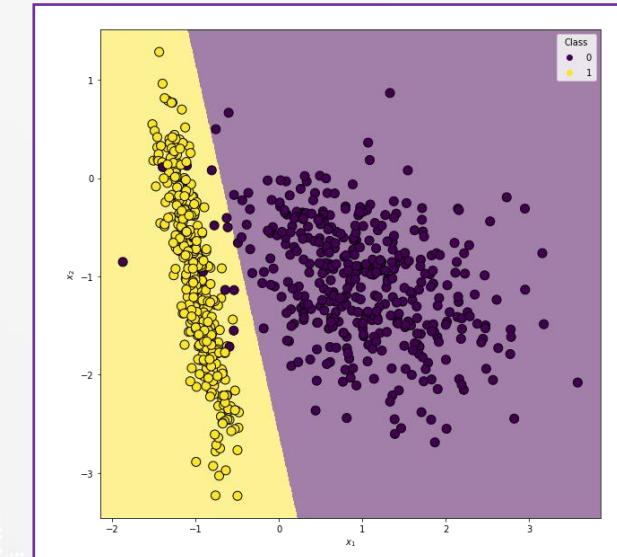
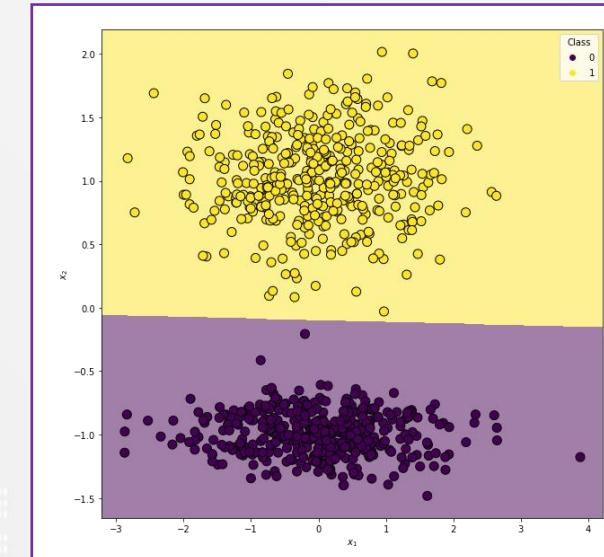
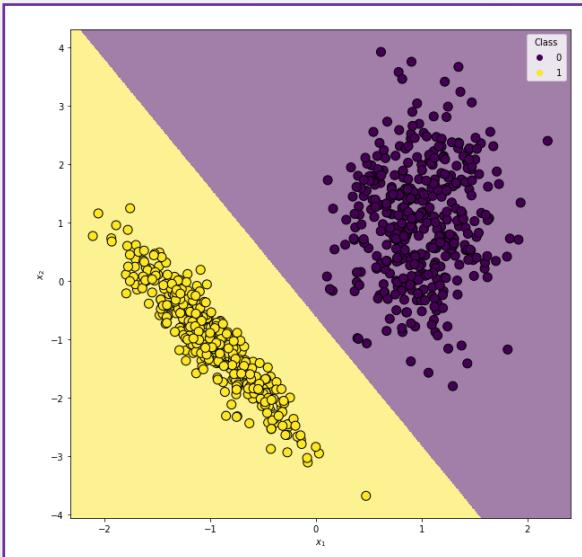
$$z = w_0 + w_1x_1 + w_2x_2$$



$$\hat{y} = \sigma(z)$$

# What is Logistic Regression?

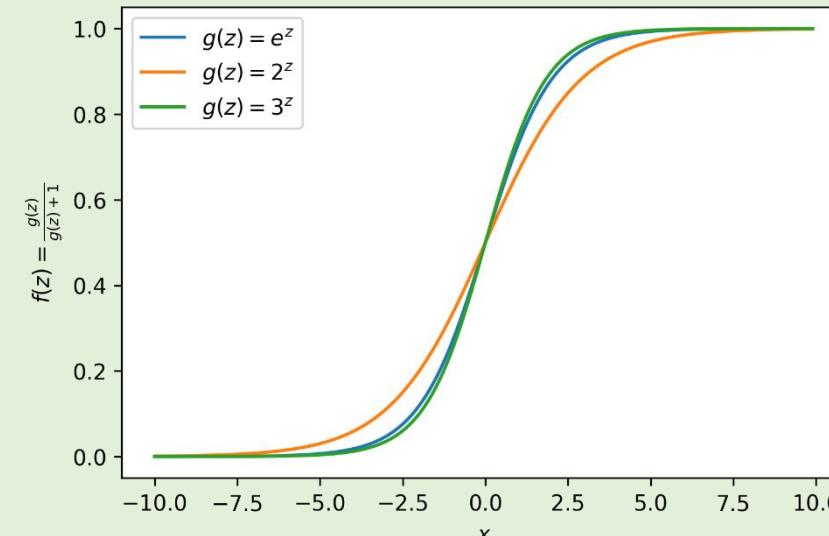
**Decision Boundary** คือ สมการที่ใช้แบ่ง class ( $z = w_0 + w_1x_1 + \cdots + w_px_p = 0$ )



# What is Logistic Regression?

Why  $e$  ?

# What is Logistic Regression?



- ◆  $\frac{\partial}{\partial z} \left( \frac{e^z}{e^z + 1} \right) = \frac{e^z}{(e^z + 1)^2}$
- ◆  $\frac{\partial}{\partial z} \left( \frac{2^z}{2^z + 1} \right) = \frac{2^z \log(2)}{(2^z + 1)^2}$
- ◆  $\frac{\partial}{\partial z} \left( \frac{3^z}{3^z + 1} \right) = \frac{3^z \log(3)}{(3^z + 1)^2}$

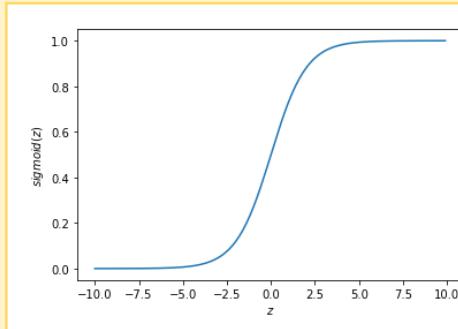
# What is Logistic Regression?

Why not step function ?

# What is Logistic Regression?

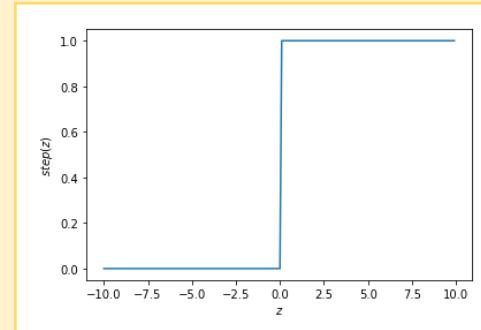
## Sigmoid Function

$$\sigma(z) = \frac{1}{1 + e^{-z}}$$



## Step Function

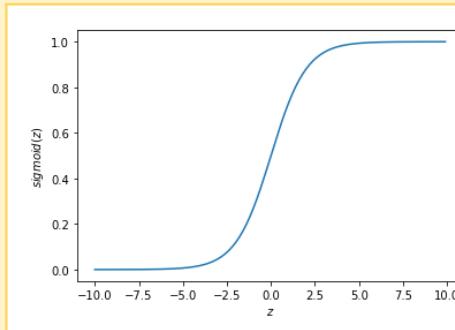
$$f(z) = \begin{cases} 1 & ; z \geq 0 \\ 0 & ; z < 0 \end{cases}$$



# What is Logistic Regression?

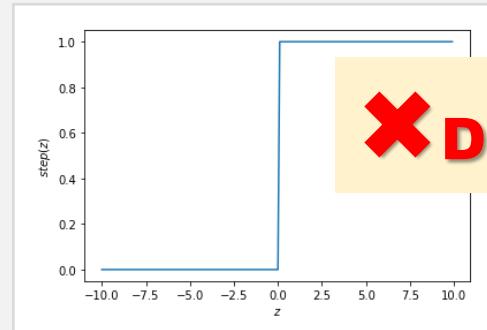
## Sigmoid Function

$$\sigma(z) = \frac{1}{1 + e^{-z}}$$



## Step Function

$$f(z) = \begin{cases} 1 & ; z \geq 0 \\ 0 & ; z < 0 \end{cases}$$



✗ Diff ไม่ได้

# Introduction

**What is Logistic  
Regression?**



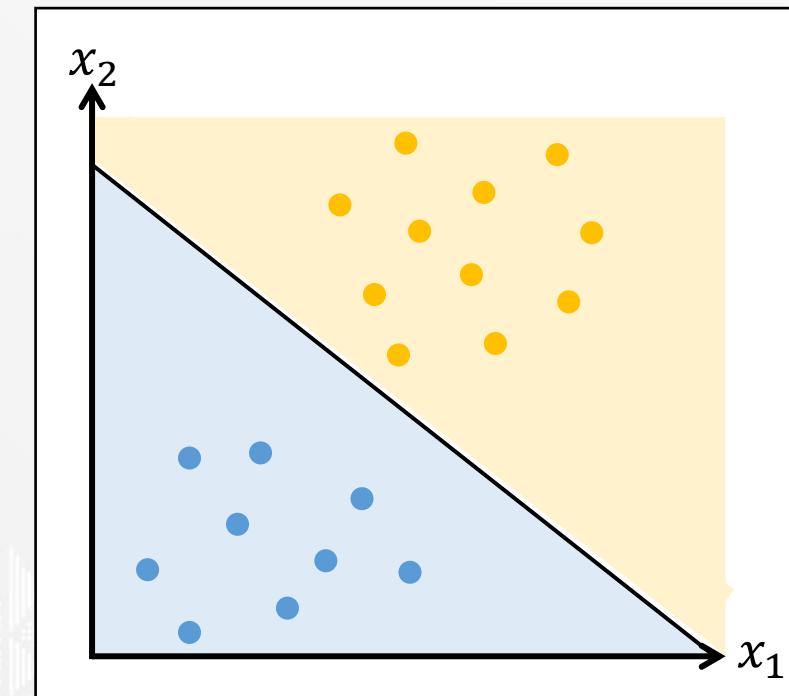
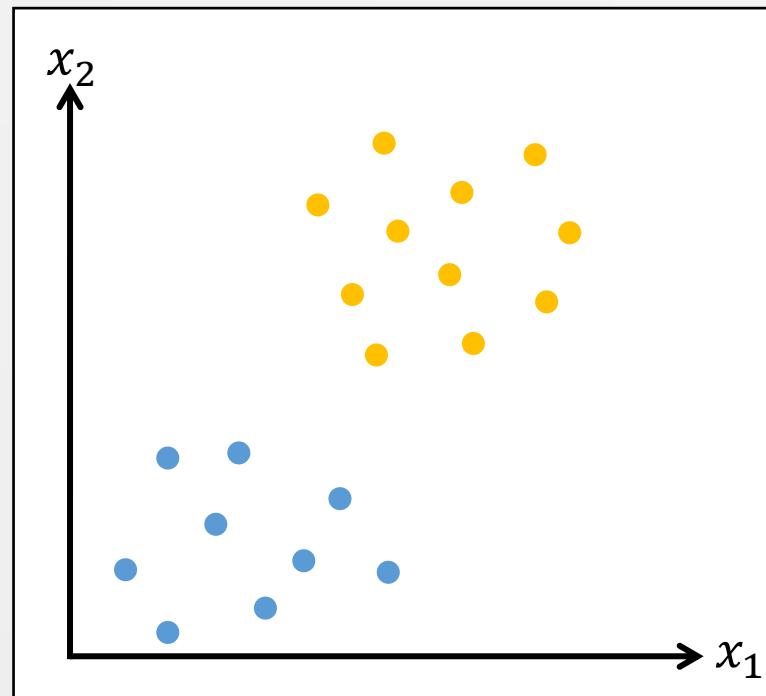
Data for Logistic  
Regression  
(Binary)

Pros & Cons

Real World  
Application

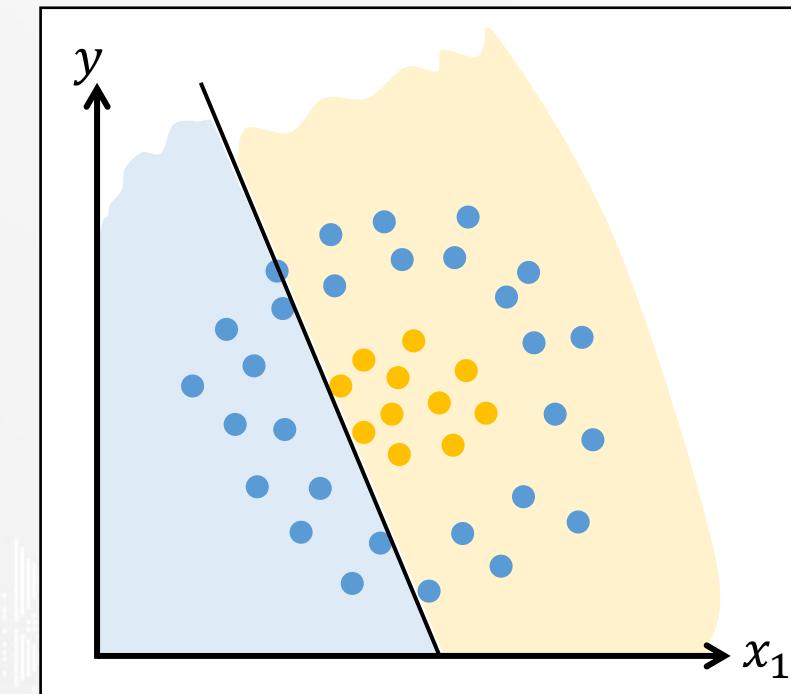
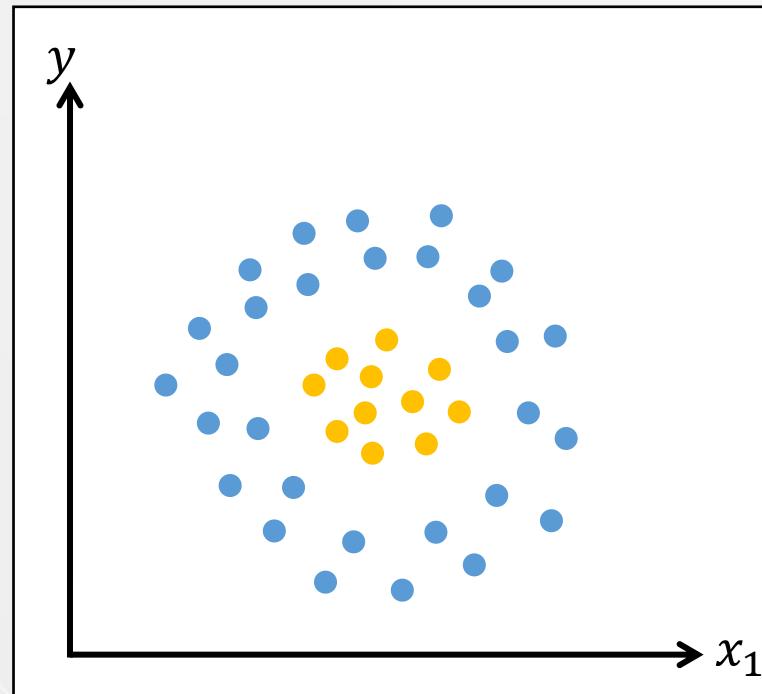
# Data for Logistic Regression (Binary)

ตัวอย่างของข้อมูลที่เหมาะสมกับ Logistic Regression



# Data for Logistic Regression (Binary)

ตัวอย่างของข้อมูลที่ไม่เหมาะสมกับ Logistic Regression



# Introduction

**What is Logistic  
Regression?**



**Data for Logistic  
Regression  
(Binary)**



**Pros & Cons**



**Real World  
Application**



# Pros & Cons

## ข้อดี

- เป็น algorithm ที่เรียบง่าย
- เป็น algorithm ที่เหมาะสมที่สุดเมื่อข้อมูลสามารถแบ่งจากกันได้อย่างสมบูรณ์แบบด้วยสมการเชิงเส้น

## ข้อเสีย

- sensitive กับ outliers

## ข้อจำกัด

- ใช้งานได้กับข้อมูลที่สามารถแบ่งจากกันได้อย่างสมบูรณ์แบบด้วยสมการเชิงเส้นเท่านั้น

# Introduction

**What is Logistic  
Regression?**



**Data for Logistic  
Regression  
(Binary)**



**Pros & Cons**



**Real World  
Application**



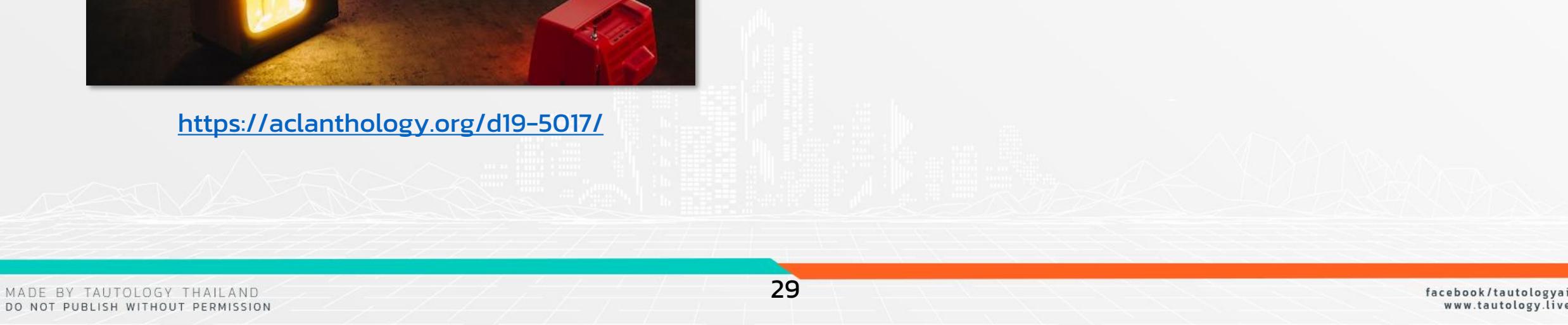
# Real World Application



## การตรวจสอบไขข้อความเชื่อ

โดยพิจารณาจาก จำนวนของคำต่าง ๆ ที่  
ปรากฏในข้อความ, ความยาวของข้อความ,  
คะแนนความอ่านง่ายของโนนุชนา เป็นต้น

<https://aclanthology.org/d19-5017/>



# Real World Application



## การจำแนกรุกิจ SME ที่มีโอกาส ล้มละลาย

โดยพิจารณาจาก ผลประกอบการ, ปริมาณ  
ดอกเบี้ยของหนี้สินบริษัท, ช่วงเวลาที่ธุรกิจค้าง  
ชำระค่าสินค้าหรือค่าบริการ เป็นต้น

<https://www.mdpi.com/2227-9091/8/4/107>

# Real World Application



## การคัดแยกผู้ป่วยที่เป็นโรคอัลไซเมอร์ ขั้นต้น

โดยพิจารณาจาก ภาพที่ได้จากการแสกนสมอง  
ด้วยวิธี MRI

<https://www.sciencedirect.com/science/article/abs/pii/s1746809420304705>

# Introduction

**What is Logistic  
Regression?**



**Data for Logistic  
Regression  
(Binary)**



**Pros & Cons**



**Real World  
Application**

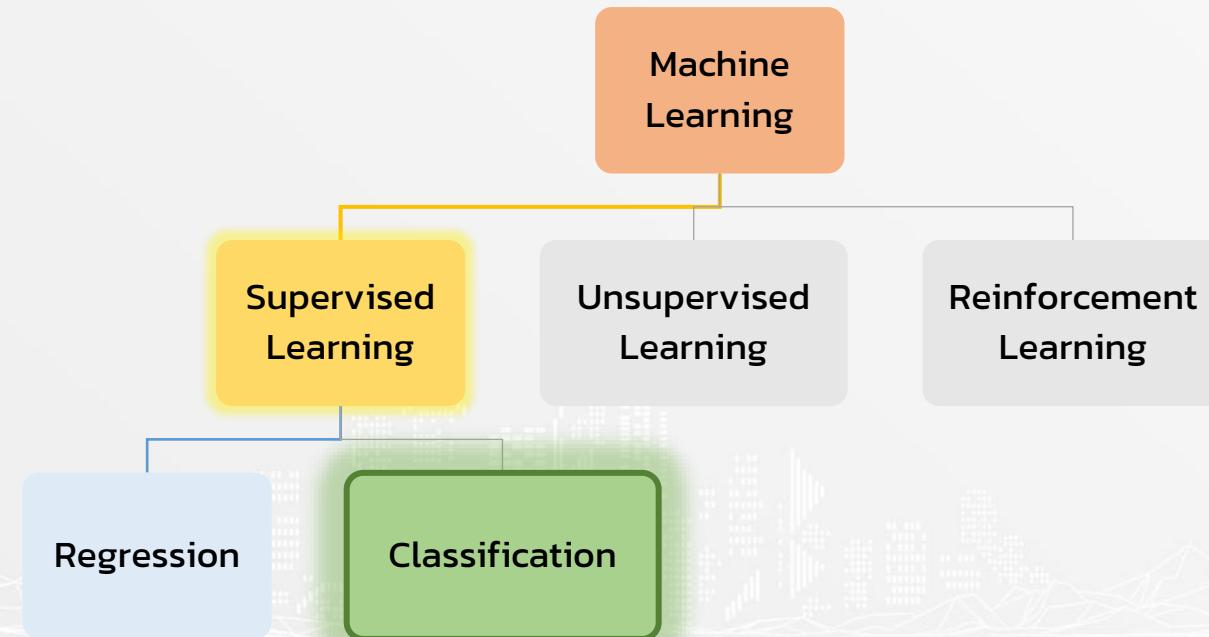


# Logistic Regression (Binary)



# Logistic Regression

**Logistic Regression** ລົງທະບຽນໃນ algorithm ປະເກດ supervised learning



# Concept of Supervised Learning

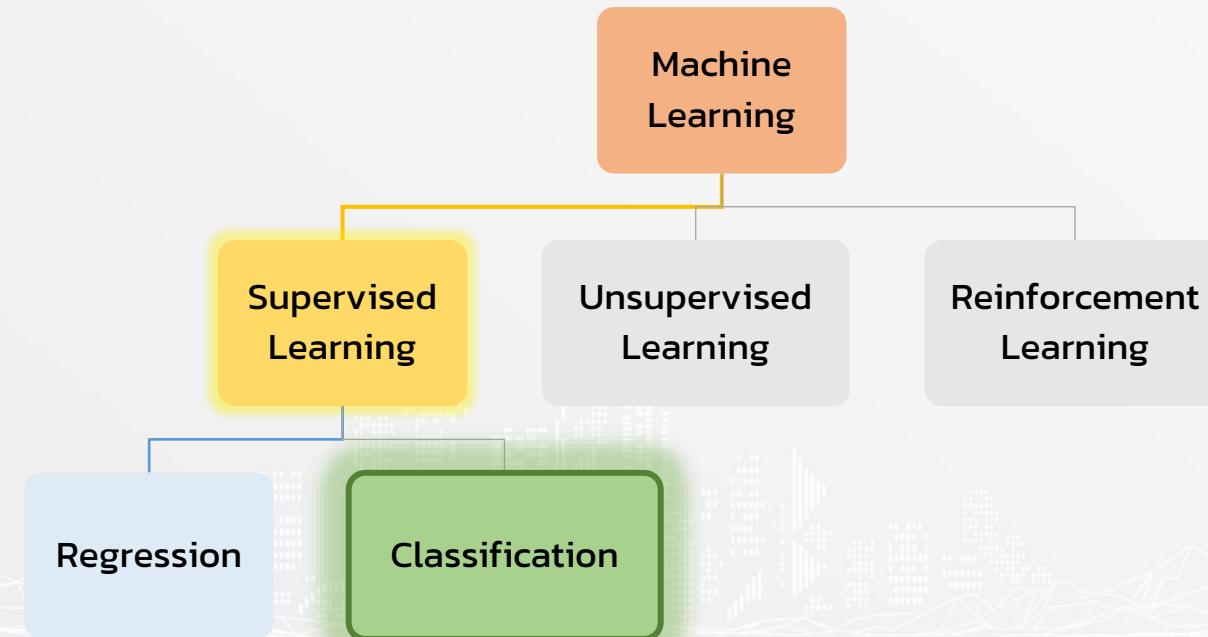
**Data   ⇒  Model   ⇒  Prediction**

# Logistic Regression (Binary)



# Logistic Regression

**Logistic Regression** ລົງທະບຽນໃນ algorithm ປະເກດ supervised learning



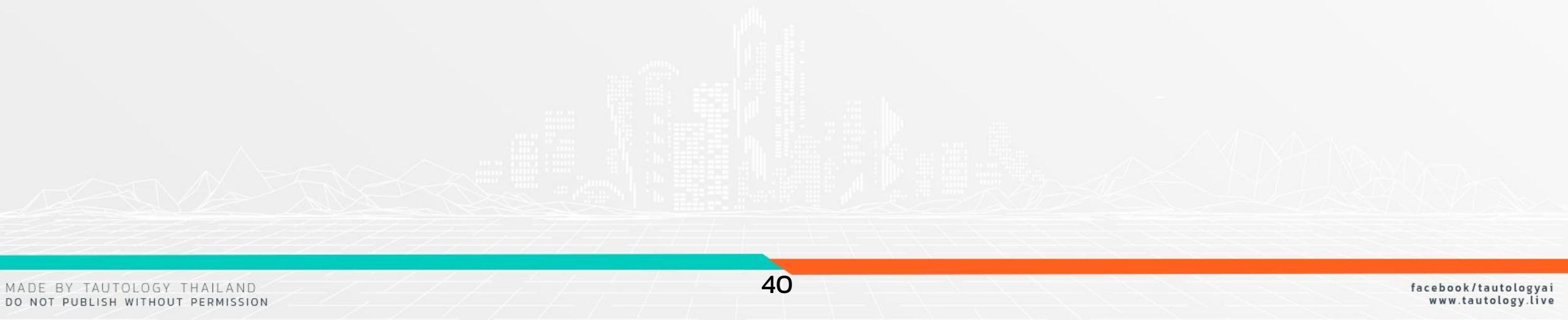
# Concept of Supervised Learning

**Data   ⇒  Model   ⇒  Prediction**

# Model Creation

# Assumption

1. Linear Relationship to Log odds
2. No Missing Features
3. No Multicollinearity



# Model Creation

**Assumption**



Real Face of the Model

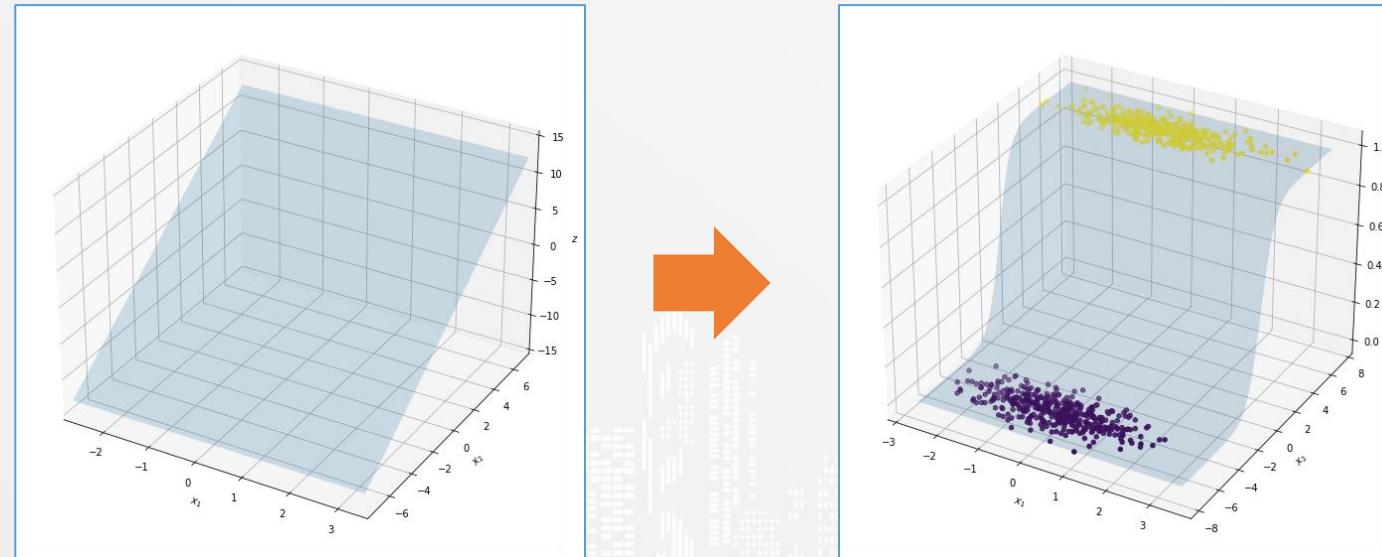
Cost Function and Cost Landscape

How to Create Model (Math)

How to Create Model (Code)

# Real Face of the Model

**Logistic regression** คือ โมเดลคอนิตศาสตร์ที่จำแนกประเภทของข้อมูล ด้วยการสร้างสมการเชิงเส้น (linear function)



# Real Face of the Model

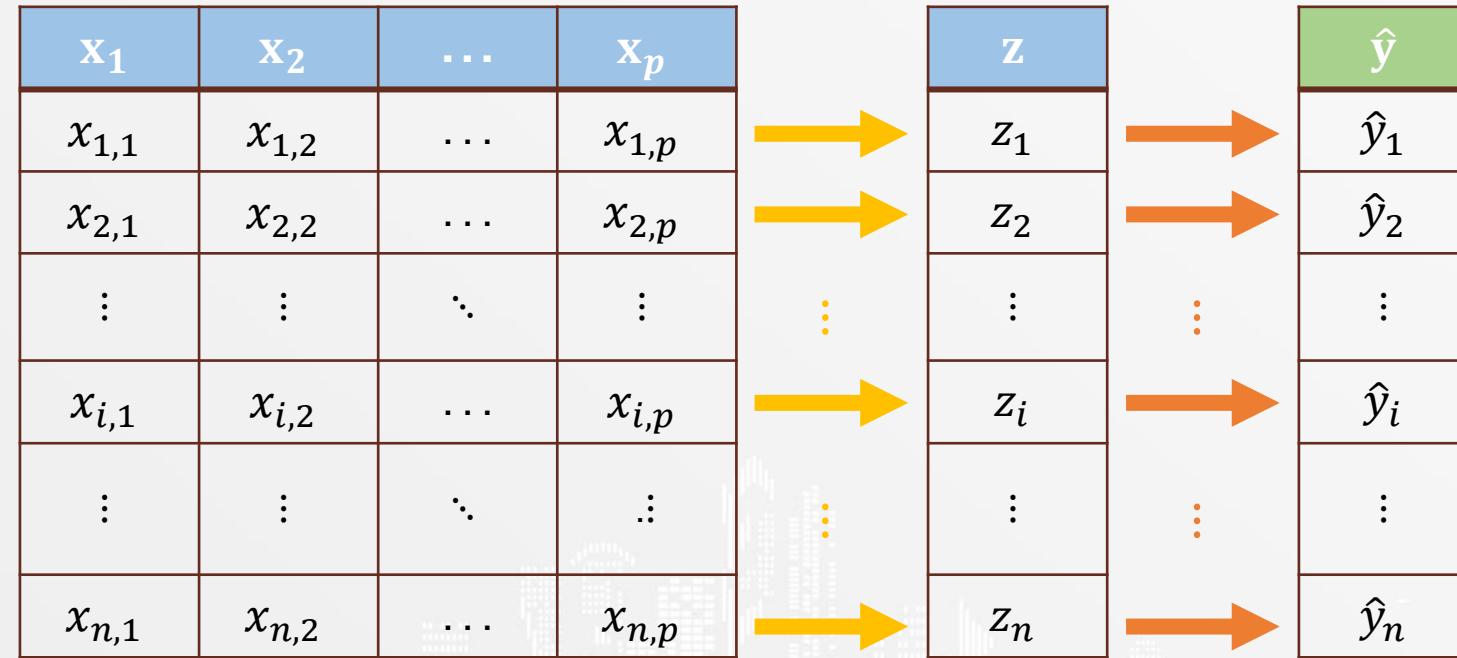
**Logistic regression** คือ โมเดลคอนิตศาสตร์ที่จำแนกประเภทของข้อมูล ด้วยการสร้างสมการเชิงเส้น (linear function)

$$z = w_0 + w_1 x_1 + w_2 x_2 + \cdots + w_p x_p$$

$$\hat{y} = \sigma(z) = \frac{1}{1+e^{-z}}$$

- โดยที่
- ◆  $\hat{y}$  คือ ตัวแปรตาม (predicted target)
  - ◆  $z$  คือ สมการเชิงเส้น (linear function)
  - ◆  $x_1, x_2, x_3, \dots, x_p$  คือ ตัวแปรตัวน (feature)
  - ◆  $w_0, w_1, w_2, \dots, w_p$  คือ สัมประสิทธิ์ (coefficient)

# Real Face of the Model



# Real Face of the Model

$$z = w_0 + w_1 x_1 + w_2 x_2 + \cdots + w_p x_p$$



$$\hat{y} = \frac{1}{1 + e^{-z}}$$



$$z_1 = w_0 + w_1 x_{1,1} + w_2 x_{1,2} + \cdots + w_p x_{1,p}$$



$$\hat{y}_1 = \frac{1}{1 + e^{-z_1}}$$

$$z_2 = w_0 + w_1 x_{2,1} + w_2 x_{2,2} + \cdots + w_p x_{2,p}$$



$$\hat{y}_2 = \frac{1}{1 + e^{-z_2}}$$

:

$$z_i = w_0 + w_1 x_{i,1} + w_2 x_{i,2} + \cdots + w_p x_{i,p}$$



$$\hat{y}_i = \frac{1}{1 + e^{-z_i}}$$

:

$$z_n = w_0 + w_1 x_{n,1} + w_2 x_{n,2} + \cdots + w_p x_{n,p}$$



$$\hat{y}_n = \frac{1}{1 + e^{-z_n}}$$

# Real Face of the Model



“ เราต้องการหา  $w_0, w_1, w_2, \dots, w_p$  ที่ทำให้ cost function ต่ำที่สุด ”

# Model Creation

**Assumption**



**Real Face of the Model**



**Cost Function and Cost Landscape**



**How to Create Model (Math)**



**How to Create Model (Code)**



# Cost Function and Cost Landscape

**Cost function** กี่เราจะใช้ในการสร้าง model คือ

$$-\frac{1}{n} \sum_{i=1}^n (y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i))$$

โดยสูตรข้างต้นมีชื่อว่า **Cross Entropy**

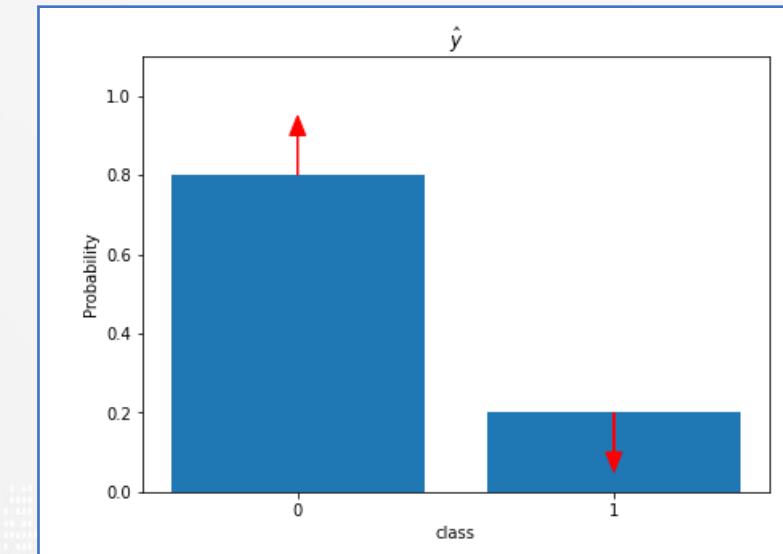
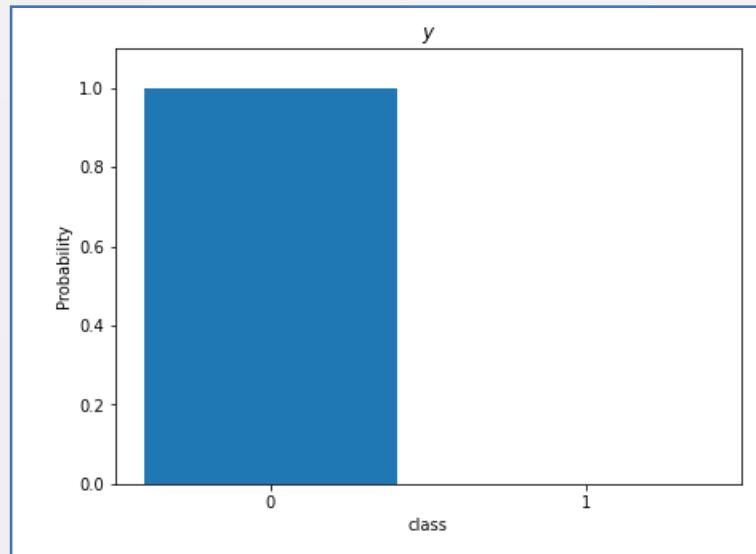
# Cost Function and Cost Landscape

## เหตุผลที่เลือกใช้ Cross Entropy

1. สามารถเปรียบเทียบ distribution ของ  $y$  กับ  $\hat{y}$  ได้
2. Cost function ที่นิยามแบบ cross entropy สามารถ diff ได้ และมีความต่อเนื่องทุกจุด
3. Cost function ที่นิยามแบบ cross entropy เป็น convex function และมีจุดต่ำสุดเพียงจุดเดียว

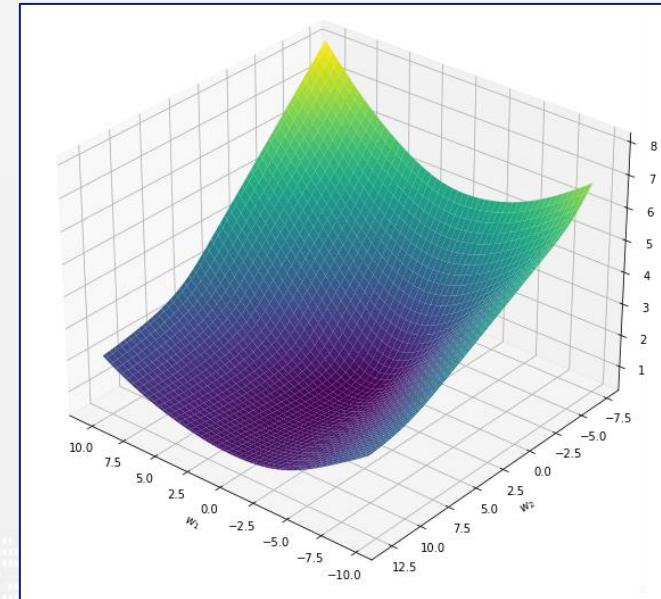
# Cost Function and Cost Landscape

1. สามารถเปรียบเทียบ distribution ของ  $y$  กับ  $\hat{y}$  ได้



# Cost Function and Cost Landscape

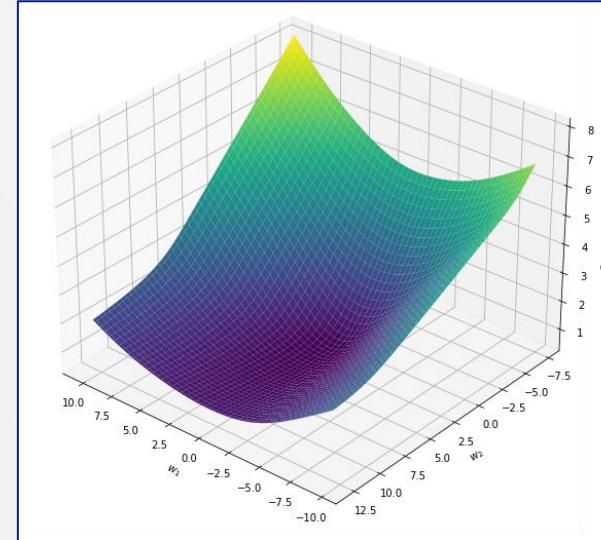
2. Cost function กี่นิยามแบบ cross entropy สามารถ diff ได้ และมีความต่อเนื่องทุกจุด



กราฟแสดง cost landscape ของ logistic regression  
โดยที่ cost function เป็น cross entropy

# Cost Function and Cost Landscape

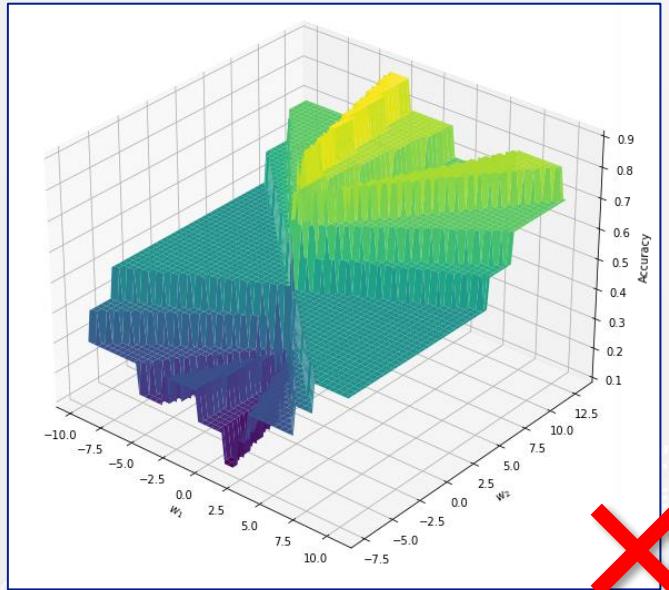
3. Cost function ที่นิยามแบบ cross entropy เป็น convex function และมีจุดต่ำสุดเพียงค่าเดียว



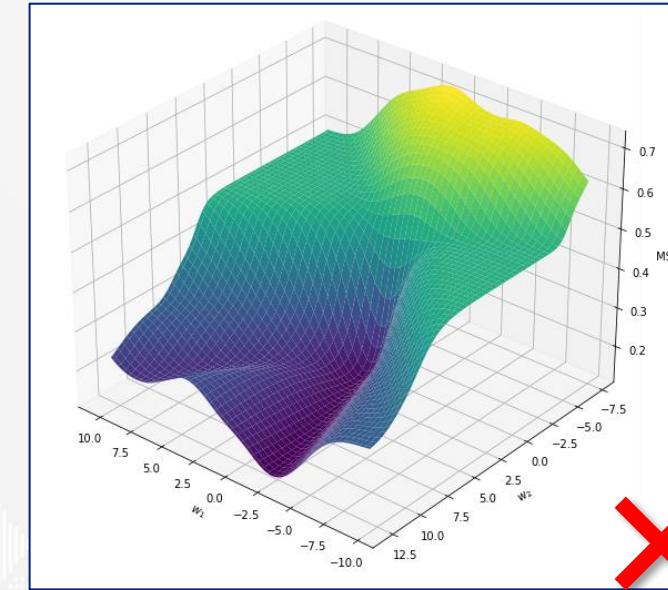
กราฟแสดง cost landscape ของ logistic regression  
โดยที่ feature เป็น no multicollinearity และ cost  
function เป็น cross entropy

# Cost Function and Cost Landscape

3. Cost function ที่นิยามแบบ cross entropy เป็น convex function และมีจุดต่ำสุดเพียงค่าเดียว



กราฟแสดง cost landscape ของ logistic regression  
โดยที่ cost function เป็น accuracy

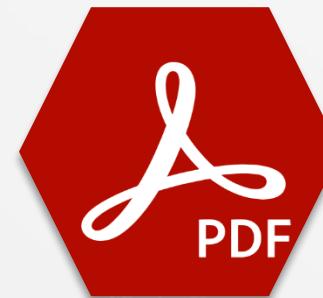


กราฟแสดง cost landscape ของ logistic regression  
โดยที่ cost function เป็น mean squared error

# Cost Function and Cost Landscape



Convexity of Cross Entropy  
for Logistic Regression (Binary)

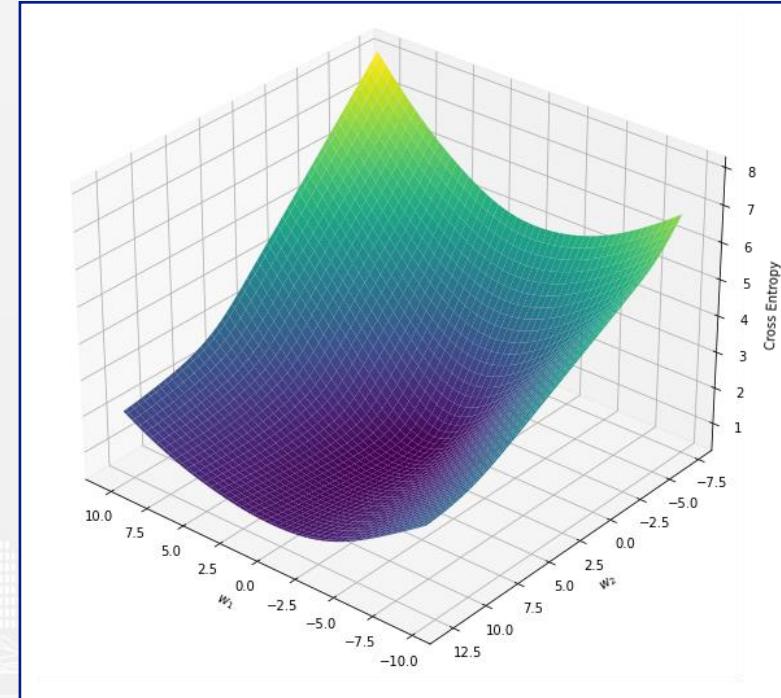


Open File

**Convexity\_CE\_LoR\_(Binary).pdf**

# Cost Function and Cost Landscape

$$Cost = -\frac{1}{n} \sum_{i=1}^n (y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i))$$



# Model Creation

**Assumption**



**Real Face of the Model**



**Cost Function and Cost Landscape**



**How to Create Model (Math)**



**How to Create Model (Code)**



# How to Create Model (Math)

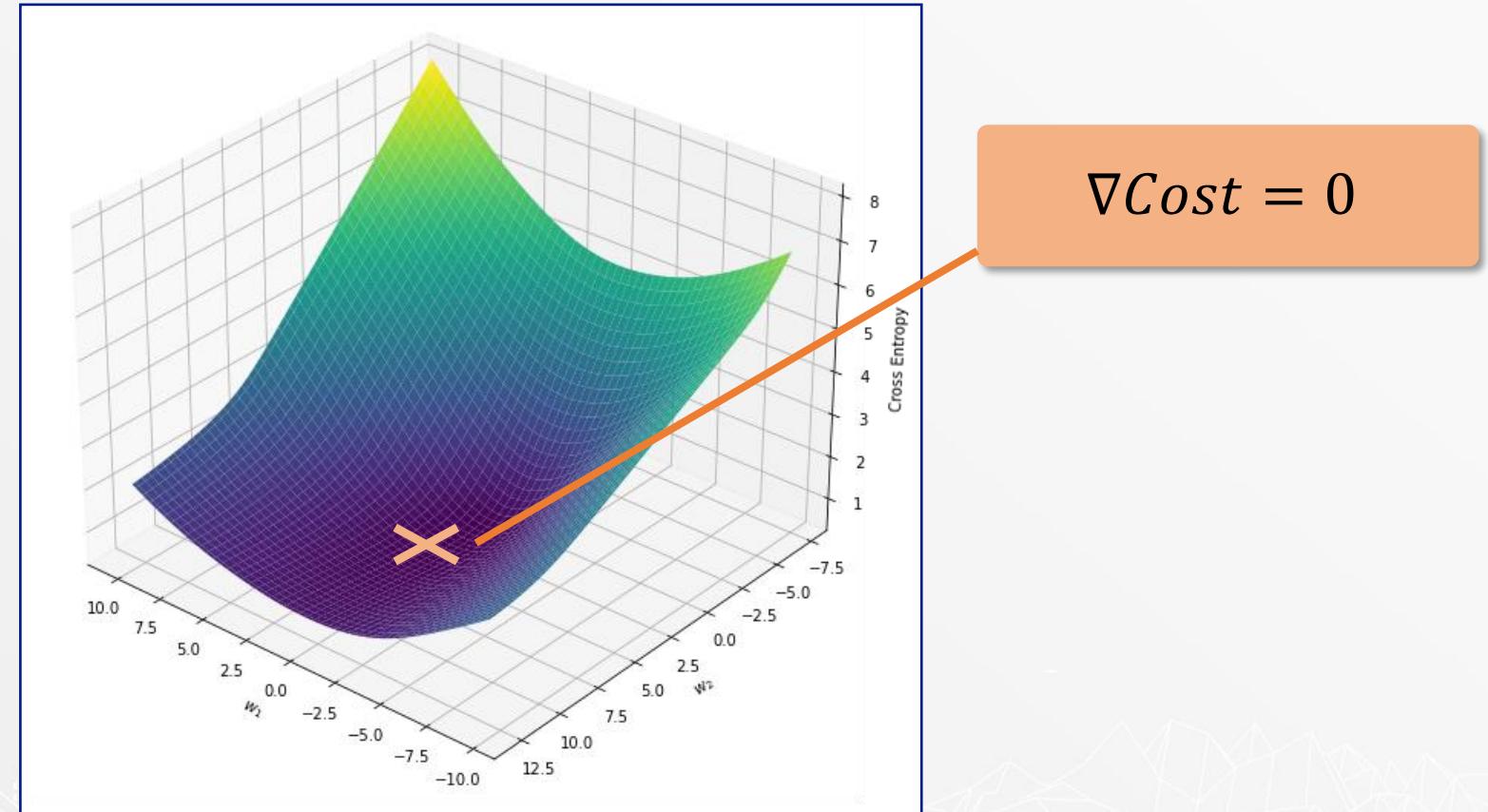
- Problem with Least Squares Method
- Gradient Descent (Logistic Regression)
- Calculation Example

# Problem with Least Squares Method

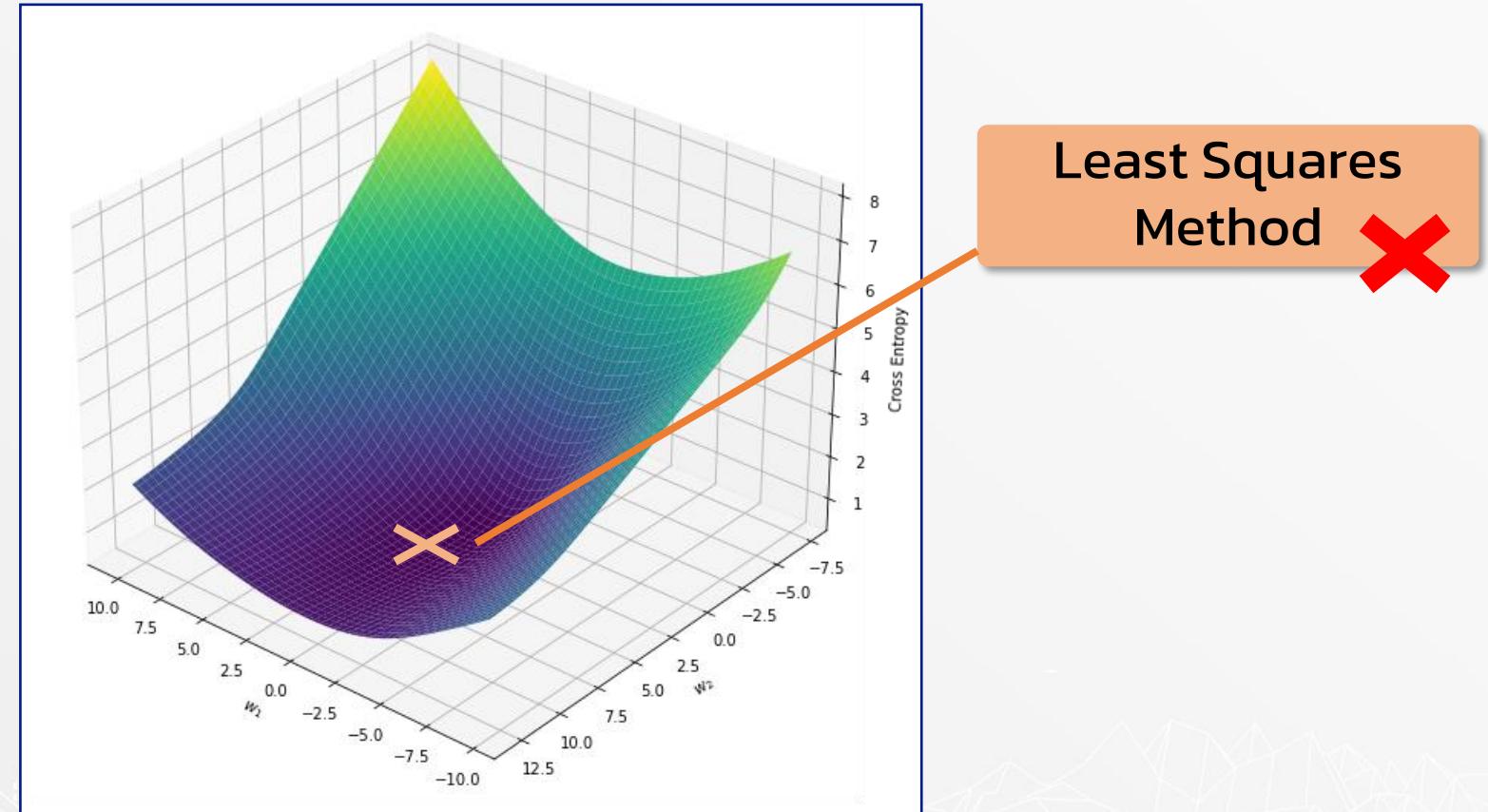


“ เราต้องการหา  $w_0, w_1, w_2, \dots, w_p$  ที่ทำให้ cost function ต่ำที่สุด ”

# Problem with Least Squares Method



# Problem with Least Squares Method



# Problem with Least Squares Method

การที่ logistic regression ไม่มี closed form solution เป็นผลเนื่องมาจากการที่ติดอยู่กับ  $z$  อยู่ในรูปของเลขซึ่งกำลังไม่สามารถถึงลงมาเพื่อแก้สมการได้

$$X_b^T \hat{\mathbf{y}} = X_b^T \mathbf{y}$$

$$\hat{\mathbf{y}} = \begin{bmatrix} \frac{1}{1+e^{-z_1}} \\ \vdots \\ \frac{1}{1+e^{-z_n}} \end{bmatrix} = \begin{bmatrix} \frac{1}{1+e^{-(w_0+w_1x_{1,1}+w_2x_{1,2}+\dots+w_px_{1,p})}} \\ \vdots \\ \frac{1}{1+e^{-(w_0+w_1x_{n,1}+w_2x_{n,2}+\dots+w_px_{n,p})}} \end{bmatrix}$$



# Problem with Least Squares Method

Note : Linear Regression

$$X_b^T \hat{\mathbf{y}} = X_b^T \hat{\mathbf{y}}$$

$$X_b^T X_b \mathbf{w} = X_b^T \hat{\mathbf{y}} \quad (\because \mathbf{z} = X_b \mathbf{w})$$

$$\mathbf{w} = (X_b^T X_b)^{-1} X_b^T \hat{\mathbf{y}}$$



# Problem with Least Squares Method

**Q : แล้วมีวิธีอื่นในการหาค่าตอบใหม่ ?**

# Problem with Least Squares Method

A : มี และวิธีการนั้นมีชื่อว่า  
**Gradient Descent**

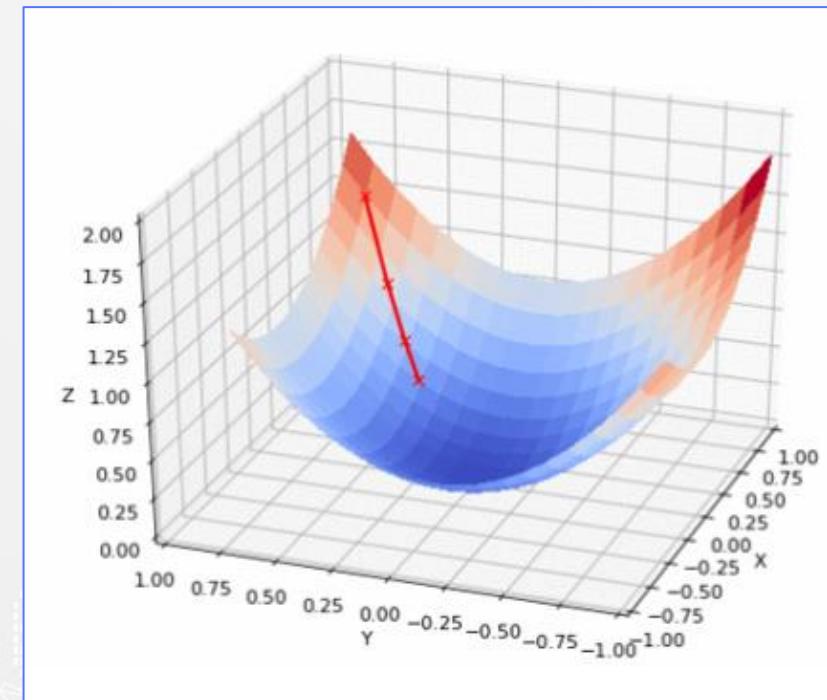
# How to Create Model (Math)

## Problem with Least Squares Method

- Gradient Descent (Logistic Regression)
- Calculation Example

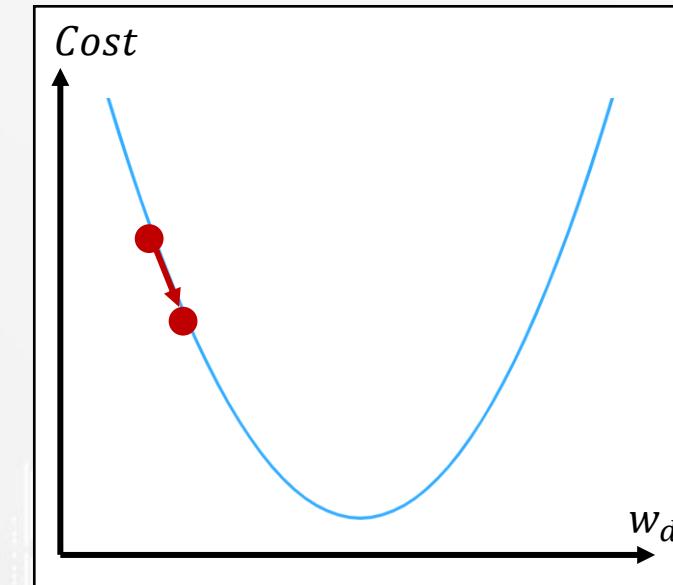
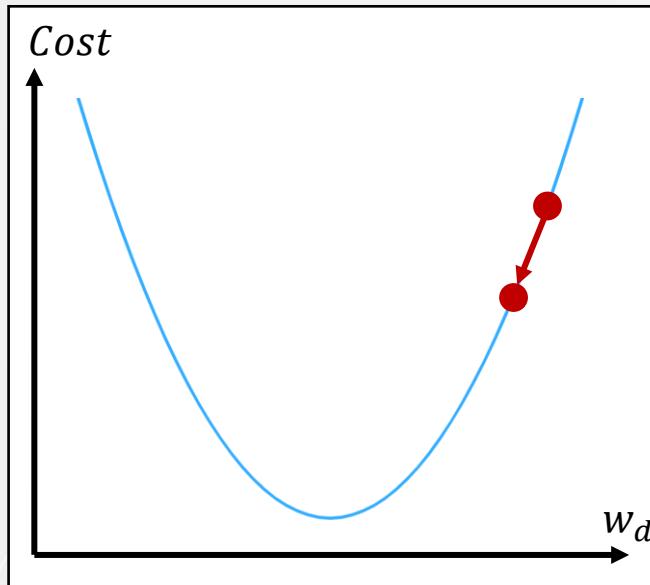
# Gradient Descent (Logistic Regression)

“ Gradient Descent ”



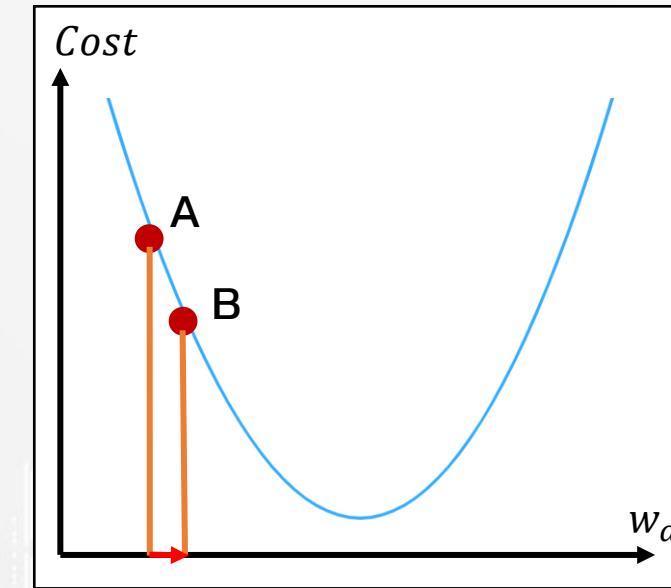
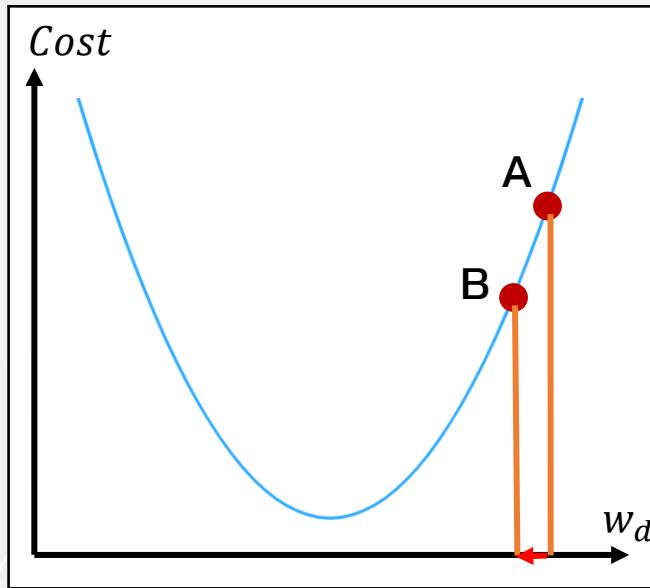
# Gradient Descent (Logistic Regression)

“ Gradient Descent ”



# Gradient Descent (Logistic Regression)

“ Gradient Descent ”



# Gradient Descent (Logistic Regression)

## Logic of Gradient Descent

1. ถ้าความซันเป็น  $+$   $\rightarrow$  ลดค่า  $w_d$
2. ถ้าความซันเป็น  $-$   $\rightarrow$  เพิ่มค่า  $w_d$

# Gradient Descent (Logistic Regression)

## Equation of Gradient Descent

$$w_d = w_d - \alpha \frac{\partial Cost}{\partial w_d}$$

โดย ◆  $\alpha$  คือ ค่าที่ใช้ควบคุม step size ของ  $w_d (> 0)$

# Gradient Descent (Logistic Regression)

เมื่อพิจารณาในทุกมิติพร้อมกัน จะได้ว่า

$$w_0 = w_0 - \alpha \frac{\partial Cost}{\partial w_0}$$

$$w_1 = w_1 - \alpha \frac{\partial Cost}{\partial w_1}$$

⋮

$$w_p = w_p - \alpha \frac{\partial Cost}{\partial w_p}$$

# Gradient Descent (Logistic Regression)

เมื่อพิจารณาในทุกมิติพร้อมกัน จะได้ว่า

$$\begin{bmatrix} w_0 \\ w_1 \\ \vdots \\ w_p \end{bmatrix} = \begin{bmatrix} w_0 \\ w_1 \\ \vdots \\ w_p \end{bmatrix} - \alpha \begin{bmatrix} \frac{\partial Cost}{\partial w_0} \\ \frac{\partial Cost}{\partial w_1} \\ \vdots \\ \frac{\partial Cost}{\partial w_p} \end{bmatrix}$$

# Gradient Descent (Logistic Regression)

## Equation of Gradient Descent

$$\mathbf{w} = \mathbf{w} - \alpha \nabla Cost$$

โดย ◆  $\alpha$  คือ ค่าที่ใช้ควบคุม step size ของ  $\mathbf{w}$

# Gradient Descent (Logistic Regression)

## Equation of Gradient Descent for Logistic Regression

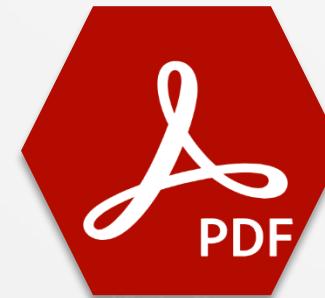
$$\mathbf{w} = \mathbf{w} + \frac{\alpha}{n} X_b^T (\mathbf{y} - \hat{\mathbf{y}})$$

โดย ◆  $\alpha$  คือ ค่าที่ใช้ควบคุม step size ของ  $\mathbf{w}$

# Gradient Descent (Logistic Regression)



Derivation of Gradient Descent  
for Logistic Regression



Open File  
**Derive\_GD\_LoR.pdf**

# Gradient Descent (Logistic Regression)

## Note

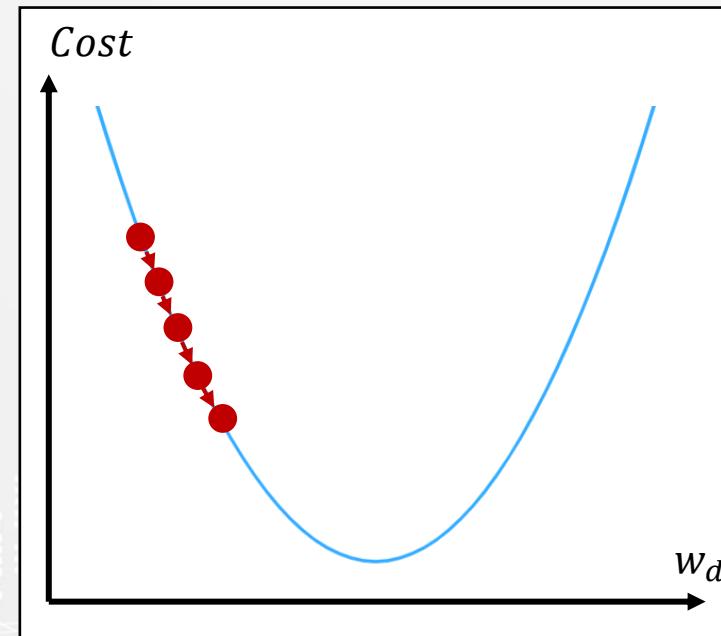
$$\mathbf{w} = \mathbf{w} + \frac{\alpha}{n} X_b^T (\mathbf{y} - \hat{\mathbf{y}})$$

โดย ◆  $\alpha$  คือ ค่าที่ใช้ควบคุม step size ของ  $\mathbf{w}$

# Gradient Descent (Logistic Regression)

## Note

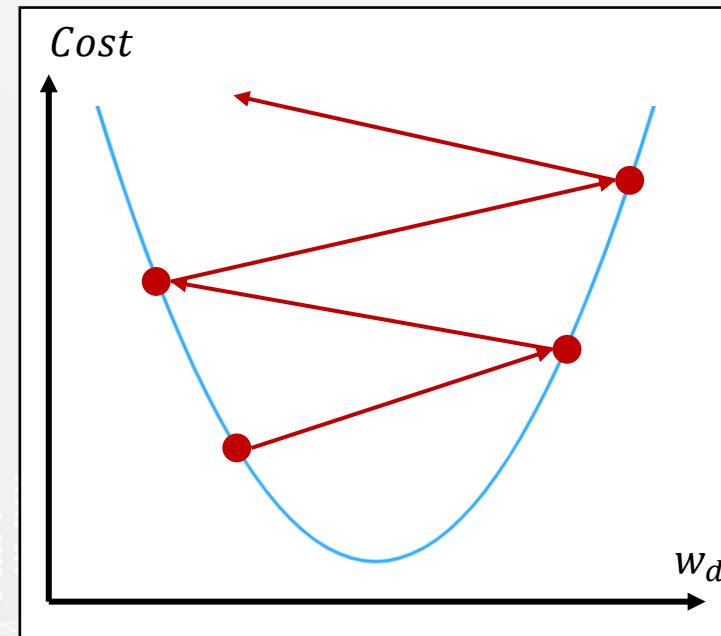
ถ้า  $\alpha$  มีค่าน้อยเกินไป : GD จะลุ่เข้าสู่คำตอบซ้ำ



# Gradient Descent (Logistic Regression)

## Note

ถ้า  $\alpha$  มีค่ามากเกินไป : GD จะลุ่วออกจากคำตوب (overflow)



# How to Create Model (Math)

- Problem with Least Squares Method**
- Gradient Descent (Logistic Regression)**
- Calculation Example

# Calculation Example

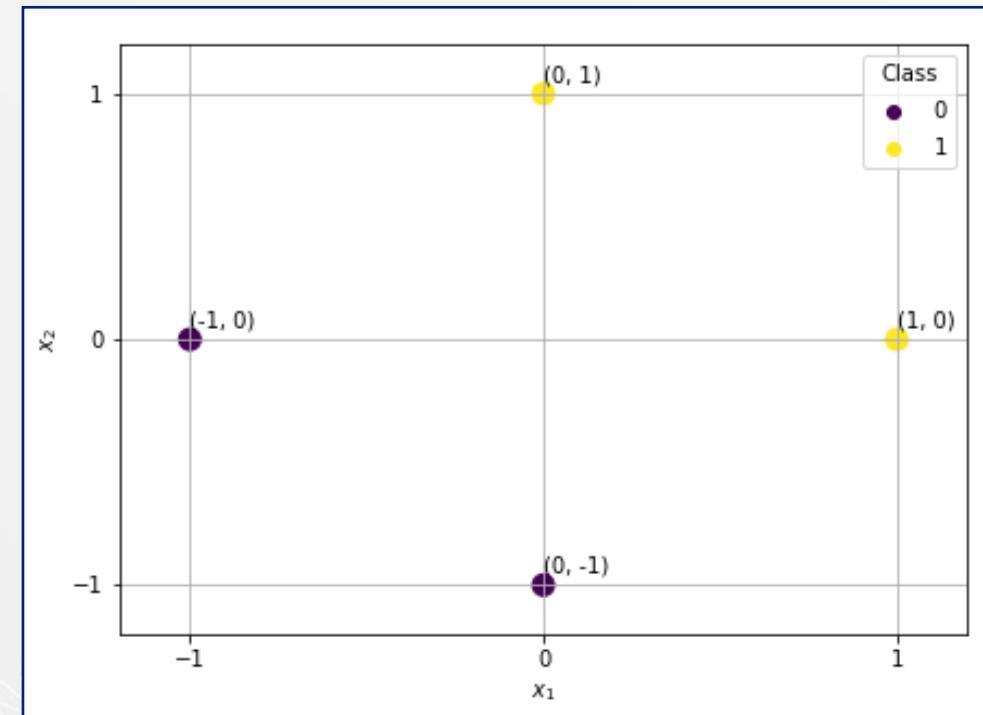
**ตัวอย่างการคำนวณ w สำหรับ logistic regression**  
**ด้วย gradient descent**

$x_1$	$x_2$	y
0	1	1
1	0	1
-1	0	0
0	-1	0

ตารางแสดง dataset

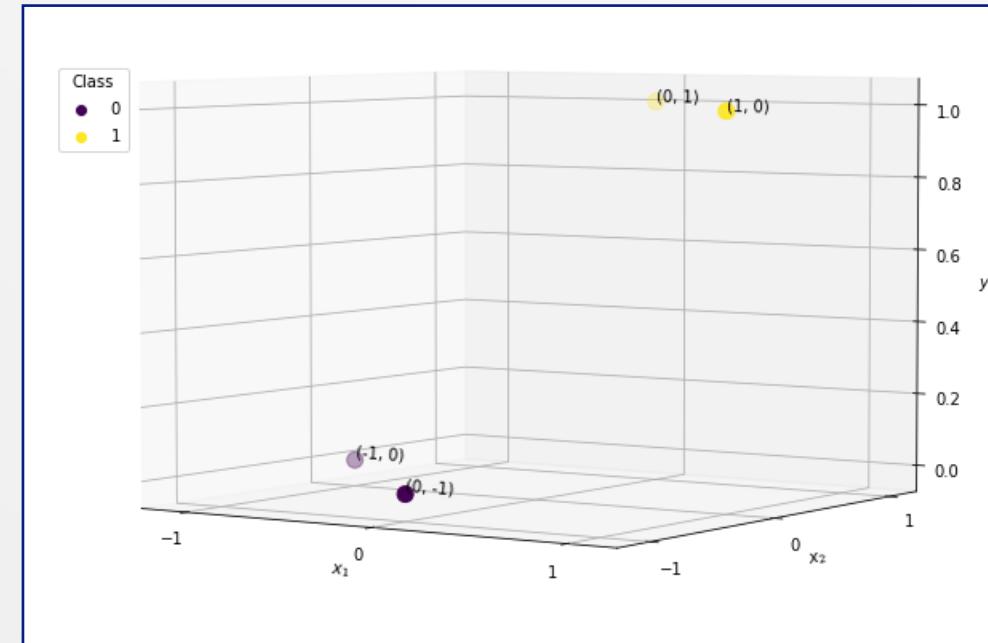
# Calculation Example

- จากข้อมูลใน dataset เราสามารถหาดالةกราฟ 2 มิติ ได้ดังนี้



# Calculation Example

- จากข้อมูลใน dataset เราสามารถหาดالةกราฟ 3 มิติ ได้ดังนี้



# Calculation Example

## ขั้นตอนในการคำนวณ gradient descent

1. สุ่ม  $w$  เริ่มต้น
2. กำหนดจำนวนรอบที่จะ update  $w$  (epoch)
3. กำหนดค่า  $\alpha$
4. For loop เพื่อ update  $w$ 
  - a) คำนวณ  $z$
  - b) คำนวณ  $\hat{y}$
  - c) คำนวณ  $w$

# Calculation Example

1. สุ่ม  $w$  เริ่มต้น

สมมติให้สุ่ม  $w = \begin{bmatrix} w_0 \\ w_1 \\ w_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$

# Calculation Example

2. กำหนดจำนวนรอบที่จะ update w (epoch)

กำหนดให้ epoch = 3

# Calculation Example

3. กำหนดค่า  $\alpha$

กำหนดให้  $\alpha = 5$

# Calculation Example

## 4. Update w

$$\mathbf{w} = \mathbf{w} + \frac{\alpha}{n} X_b^T (\mathbf{y} - \hat{\mathbf{y}})$$

โดย ◆  $\alpha$  คือ ค่าที่ใช้ควบคุม step size ของ  $\mathbf{w}$

# Calculation Example

จากข้อมูลใน dataset เราสามารถเขียน  $X, y$  และ  $X_b$  ได้ดังต่อไปนี้

$$X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \\ -1 & 0 \\ 0 & -1 \end{bmatrix}, \quad y = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \end{bmatrix} \text{ และ } X_b = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \\ 1 & -1 & 0 \\ 1 & 0 & -1 \end{bmatrix}$$

# Calculation Example

## 4. Update w : คำนวณ z

epoch = 1

$$\mathbf{z} = \mathbf{X}_b \mathbf{w} = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \\ 1 & -1 & 0 \\ 1 & 0 & -1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ -1 \end{bmatrix}$$

# Calculation Example

## 4. Update w : คำนวณ $\hat{y}$

epoch = 1

$$\hat{y} = \begin{bmatrix} \frac{1}{1+e^{-z_1}} \\ \frac{1}{1+e^{-z_2}} \\ \frac{1}{1+e^{-z_3}} \\ \frac{1}{1+e^{-z_4}} \end{bmatrix} = \begin{bmatrix} \frac{1}{1+e^{-1}} \\ \frac{1}{1+e^0} \\ \frac{1}{1+e^0} \\ \frac{1}{1+e^1} \end{bmatrix} = \begin{bmatrix} 0.731 \\ 0.5 \\ 0.5 \\ 0.269 \end{bmatrix}$$

# Calculation Example

## 4. Update w : คำนวณ w

epoch = 1

$$\begin{aligned} \mathbf{w} &= \mathbf{w} + \frac{\alpha}{n} X_b^T (\mathbf{y} - \hat{\mathbf{y}}) \\ &= \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} + \frac{5}{4} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & 1 & -1 & 0 \\ 1 & 0 & 0 & -1 \end{bmatrix} \left( \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \end{bmatrix} - \begin{bmatrix} 0.731 \\ 0.5 \\ 0.5 \\ 0.269 \end{bmatrix} \right) = \begin{bmatrix} 0 \\ 1.25 \\ 1.672 \end{bmatrix} \end{aligned}$$

# Calculation Example

## 4. Update w : คำนวณ z

epoch = 2

$$\mathbf{z} = \mathbf{X}_b \mathbf{w} = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \\ 1 & -1 & 0 \\ 1 & 0 & -1 \end{bmatrix} \begin{bmatrix} 0 \\ 1.25 \\ 1.672 \end{bmatrix} = \begin{bmatrix} 1.672 \\ 1.25 \\ -1.25 \\ -1.672 \end{bmatrix}$$

# Calculation Example

## 4. Update w : คำนวณ $\hat{y}$

epoch = 2

$$\hat{y} = \begin{bmatrix} \frac{1}{1+e^{-z_1}} \\ \frac{1}{1+e^{-z_2}} \\ \frac{1}{1+e^{-z_3}} \\ \frac{1}{1+e^{-z_4}} \end{bmatrix} = \begin{bmatrix} \frac{1}{1+e^{-1.672}} \\ \frac{1}{1+e^{-1.25}} \\ \frac{1}{1+e^{1.25}} \\ \frac{1}{1+e^{1.672}} \end{bmatrix} = \begin{bmatrix} 0.842 \\ 0.777 \\ 0.222 \\ 0.158 \end{bmatrix}$$

# Calculation Example

## 4. Update w : คำนวณ w

epoch = 2

$$\begin{aligned} \mathbf{w} &= \mathbf{w} + \frac{\alpha}{n} X_b^T (\mathbf{y} - \hat{\mathbf{y}}) \\ &= \begin{bmatrix} 0 \\ 1.25 \\ 1.672 \end{bmatrix} + \frac{5}{4} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & 1 & -1 & 0 \\ 1 & 0 & 0 & -1 \end{bmatrix} \left( \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \end{bmatrix} - \begin{bmatrix} 0.842 \\ 0.777 \\ 0.222 \\ 0.158 \end{bmatrix} \right) = \begin{bmatrix} 0 \\ 1.807 \\ 2.068 \end{bmatrix} \end{aligned}$$

# Calculation Example

## 4. Update w : คำนวณ z

epoch = 3

$$\mathbf{z} = \mathbf{X}_b \mathbf{w} = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \\ 1 & -1 & 0 \\ 1 & 0 & -1 \end{bmatrix} \begin{bmatrix} 0 \\ 1.807 \\ 2.068 \end{bmatrix} = \begin{bmatrix} 2.068 \\ 1.807 \\ -1.807 \\ -2.068 \end{bmatrix}$$

# Calculation Example

## 4. Update w : คำนวณ $\hat{y}$

epoch = 3

$$\hat{y} = \begin{bmatrix} \frac{1}{1+e^{-z_1}} \\ \frac{1}{1+e^{-z_2}} \\ \frac{1}{1+e^{-z_3}} \\ \frac{1}{1+e^{-z_4}} \end{bmatrix} = \begin{bmatrix} \frac{1}{1+e^{-2.068}} \\ \frac{1}{1+e^{-1.807}} \\ \frac{1}{1+e^{1.807}} \\ \frac{1}{1+e^{2.068}} \end{bmatrix} = \begin{bmatrix} 0.888 \\ 0.859 \\ 0.141 \\ 0.112 \end{bmatrix}$$

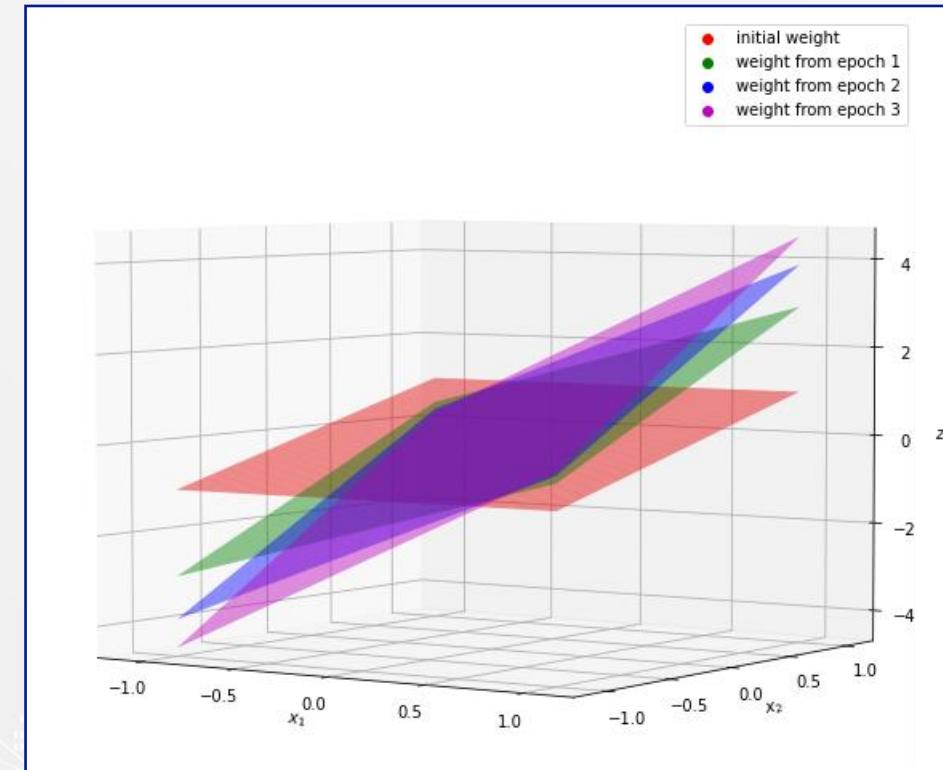
# Calculation Example

## 4. Update w : คำนวณ w

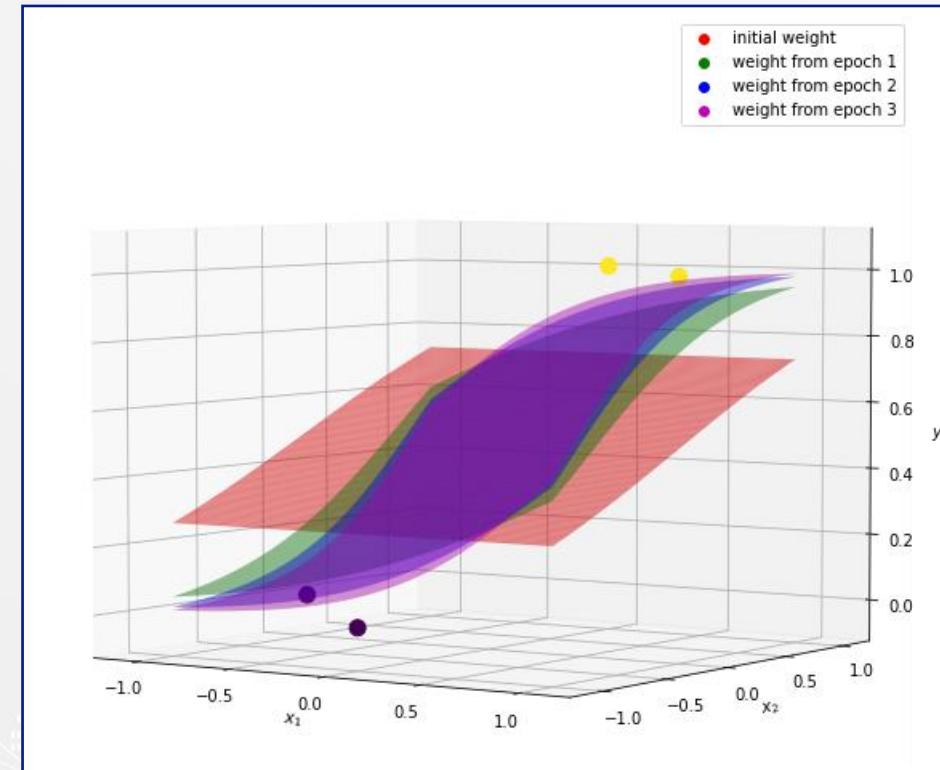
epoch = 3

$$\begin{aligned} \mathbf{w} &= \mathbf{w} + \frac{\alpha}{n} X_b^T (\mathbf{y} - \hat{\mathbf{y}}) \\ &= \begin{bmatrix} 0 \\ 1.807 \\ 2.068 \end{bmatrix} + \frac{5}{4} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & 1 & -1 & 0 \\ 1 & 0 & 0 & -1 \end{bmatrix} \left( \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \end{bmatrix} - \begin{bmatrix} 0.888 \\ 0.859 \\ 0.141 \\ 0.112 \end{bmatrix} \right) = \begin{bmatrix} 0 \\ 2.159 \\ 2.348 \end{bmatrix} \end{aligned}$$

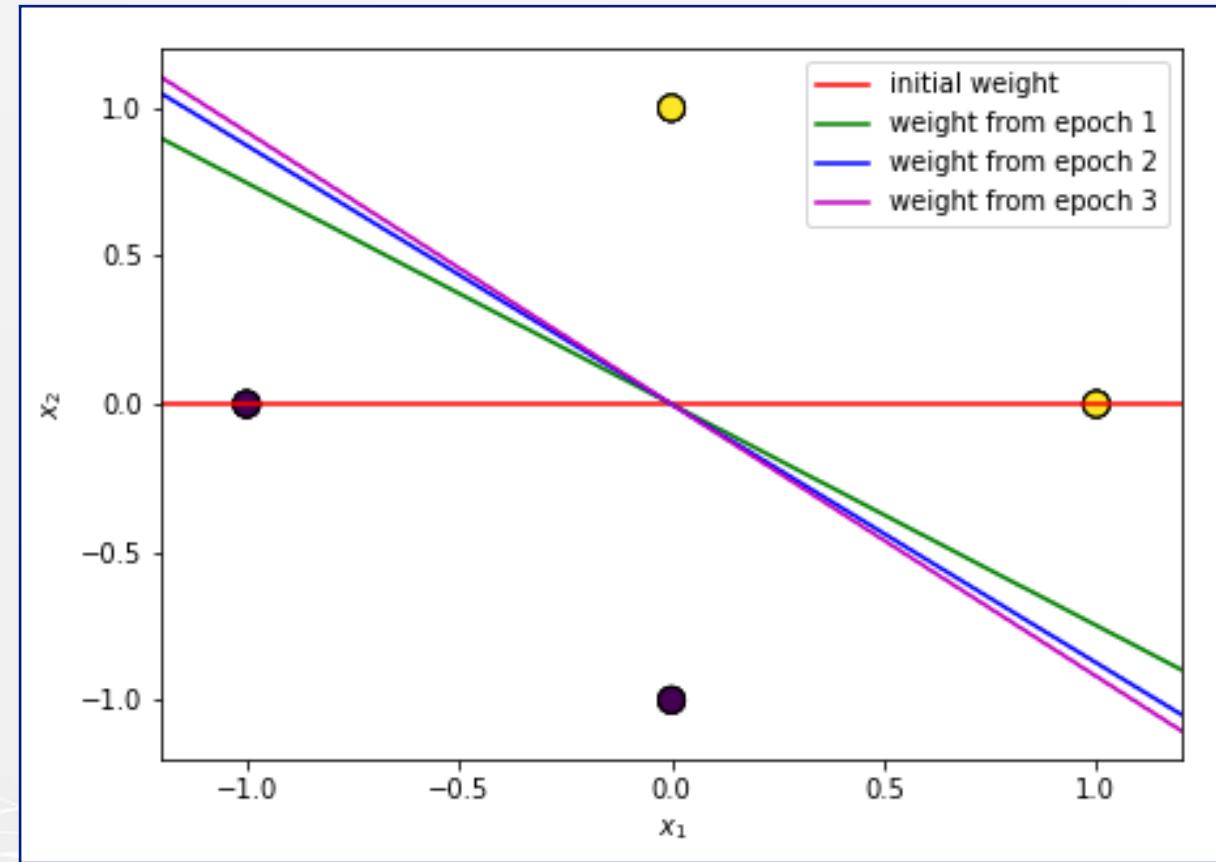
# Calculation Example



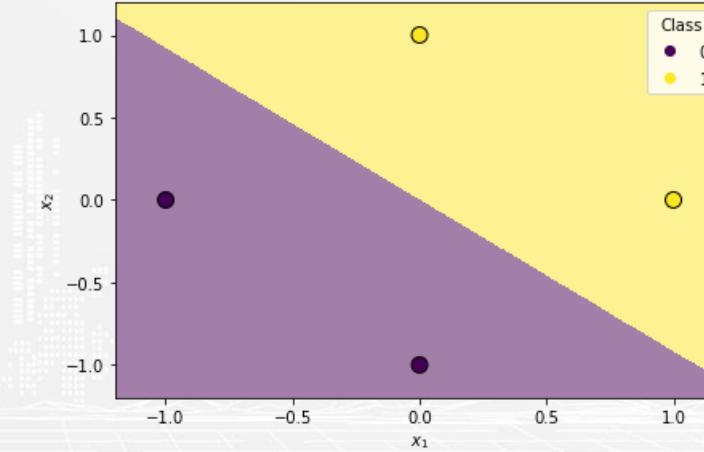
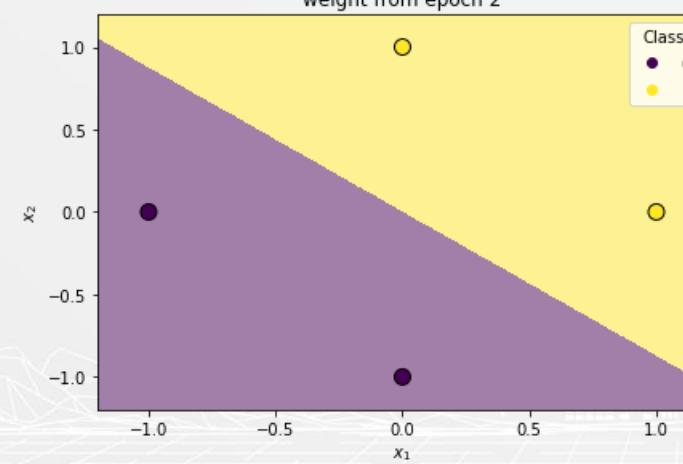
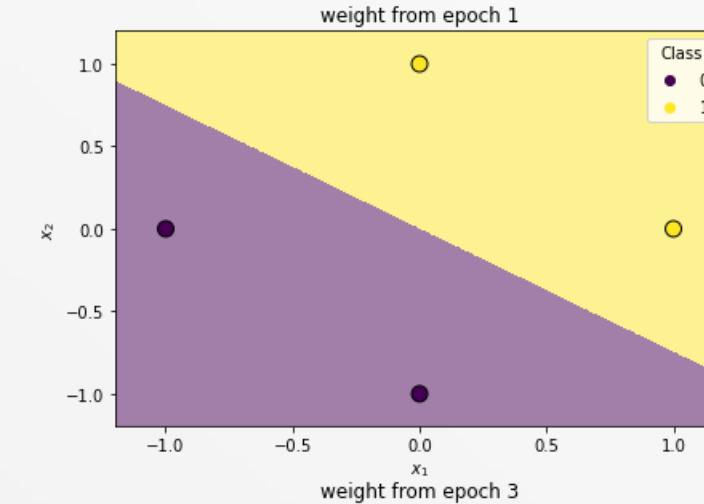
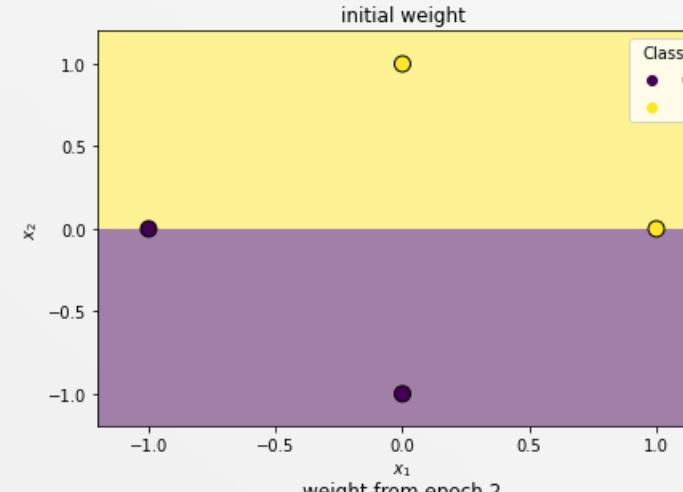
# Calculation Example



# Calculation Example



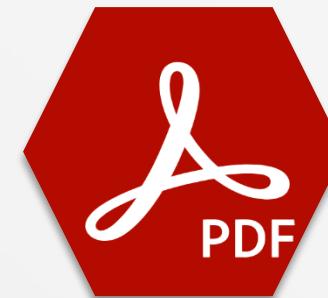
# Calculation Example



# Calculation Example



Exercise of Logistic Regression (Binary)



Open File  
**Exercise\_LoR\_(Binary).pdf**

# How to Create Model (Math)

- Problem with Least Squares Method**
- Gradient Descent (Logistic Regression)**
- Calculation Example**

# Model Creation

**Assumption**



**Real Face of the Model**



**Cost Function and Cost Landscape**



**How to Create Model (Math)**



**How to Create Model (Code)**



# How to Create Model (Code)

ตัวอย่าง code สำหรับคำนวณ w

$x_1$	$x_2$	y
0	1	1
1	0	1
-1	0	0
0	-1	0

ตารางแสดง dataset

# How to Create Model (Code)

- Code สำหรับสร้าง model จากข้อมูลของเราโดยที่

$$X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \\ -1 & 0 \\ 0 & -1 \end{bmatrix} \text{ และ } y = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \end{bmatrix}$$

```
1 clf = LogisticRegression(penalty='none')
2 clf.fit(X, y)
```

LogisticRegression(penalty='none')

# How to Create Model (Code)

- ค่า  $w_0$  จะเก็บไว้ใน attribute ชื่อ intercept\_

```
1 clf.intercept_
array([0.])
```

# How to Create Model (Code)

- ค่า  $w_1, \dots, w_p$  จะเก็บไว้ใน attribute ชื่อ `coef_`

```
1 clf.coef_
```

```
array([[10.52183855, 10.52183855]])
```

# How to Create Model (Code)

ดังนั้น เราจะได้  $w_0 = 0, w_1 = 10.5218, w_2 = 10.5218$  ซึ่งสามารถเขียนเป็น model ของ logistic regression สำหรับข้อมูลชุดนี้ได้ดังนี้

$$z = 0 + 10.5218x_1 + 10.5218x_2$$

$$\hat{y} = \sigma(z)$$

# How to Create Model (Code)

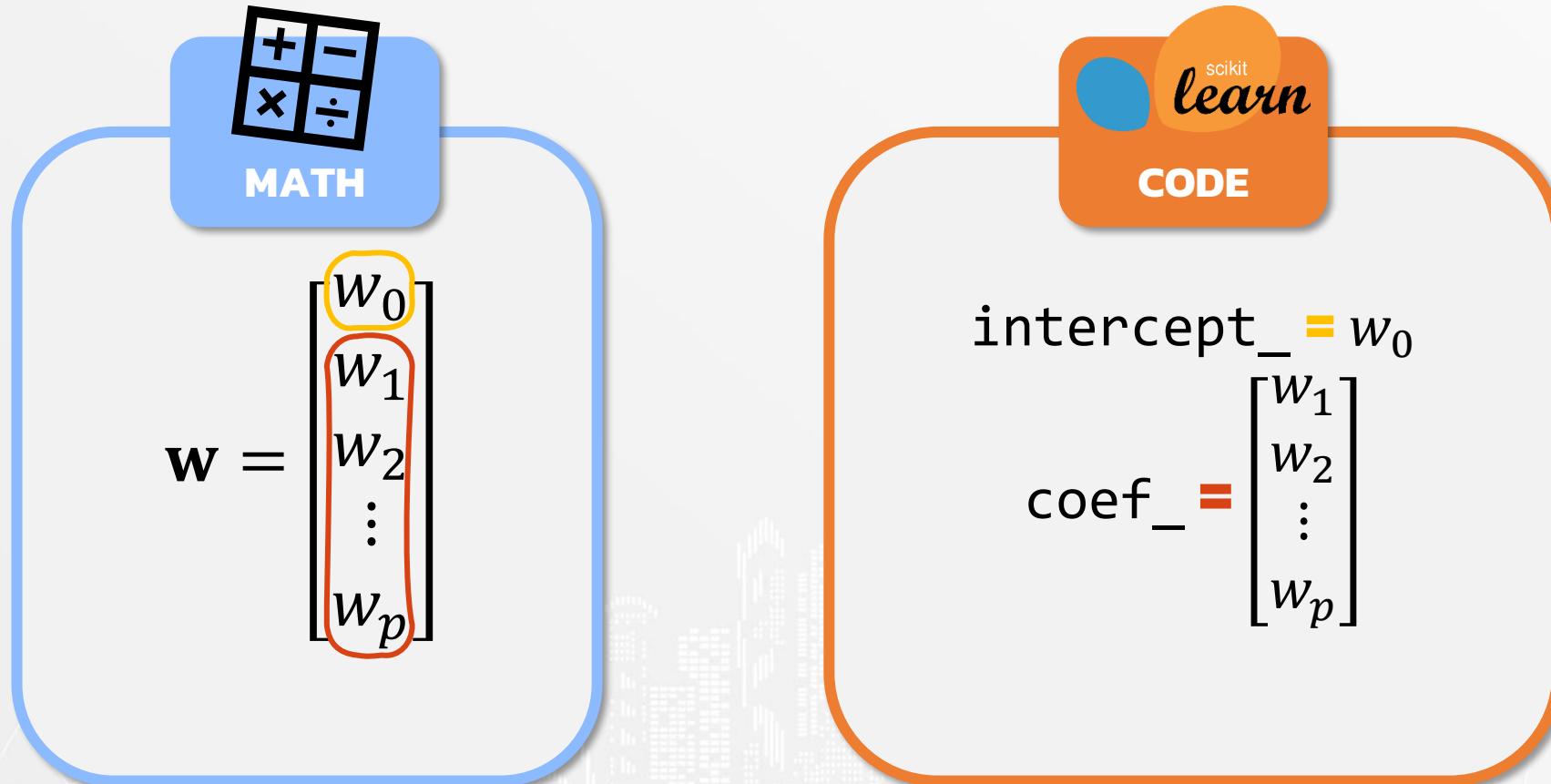


Code for this section

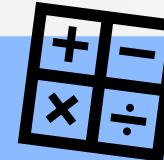


Open File  
**Model Creation (Binary).ipynb**

# How to Create Model (Code)



# How to Create Model (Code)



MATH

1. สูม  $w$  เริ่มต้น
2. กำหนดจำนวนรอบที่จะ update  $w$  (epoch)
3. กำหนดค่า  $\alpha$
4. `for i in range(epoch):  
 update w`



CODE

```
1 clf = LogisticRegression(penalty='none')  
2 clf.fit(X, y)
```

`LogisticRegression(penalty='none')`

```
1 clf.intercept_
```

`array([0.])`

```
1 clf.coef_
```

`array([[10.52183855, 10.52183855]])`

# Model Creation

**Assumption**



**Real Face of the Model**



**Cost Function and Cost Landscape**



**How to Create Model (Math)**



**How to Create Model (Code)**



# Logistic Regression (Binary)



# Prediction

# Prediction

**Logistic regression** คือ โมเดลคอมพิวเตอร์ที่จำแนกประเภทของข้อมูล ด้วยการสร้างสมการเชิงเส้น (linear function)

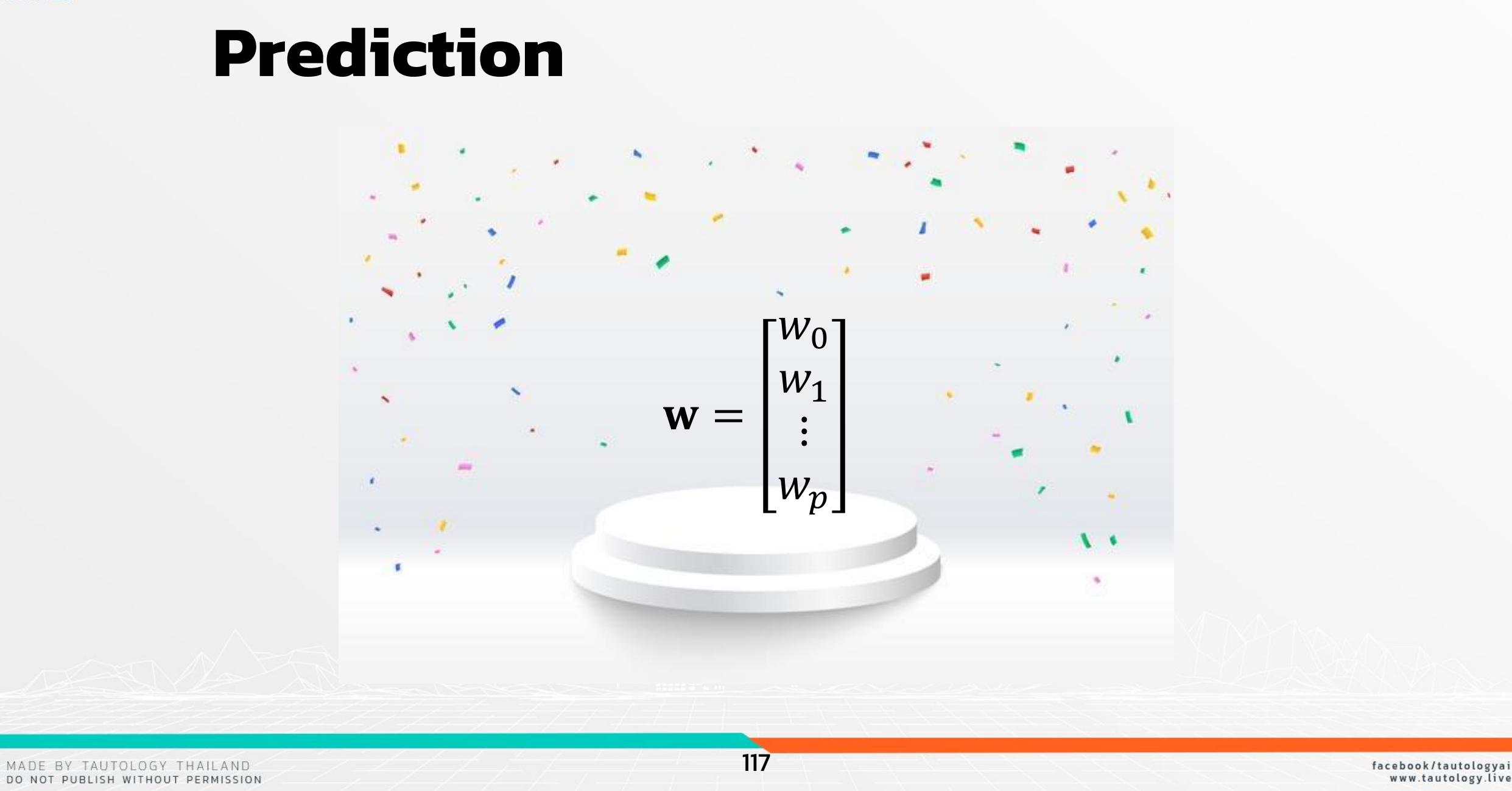
$$z = w_0 + w_1 x_1 + w_2 x_2 + \cdots + w_p x_p$$

$$\hat{y} = \sigma(z) = \frac{1}{1+e^{-z}}$$

- โดยที่
- ◆  $\hat{y}$  คือ ตัว預測 (predicted target)
  - ◆  $z$  คือ สมการเชิงเส้น (linear function)
  - ◆  $x_1, x_2, x_3, \dots, x_p$  คือ ตัว特征 (feature)
  - ◆  $w_0, w_1, w_2, \dots, w_p$  คือ สัมประสิทธิ์ (coefficient)

# Prediction

$$\mathbf{w} = \begin{bmatrix} w_0 \\ w_1 \\ \vdots \\ w_p \end{bmatrix}$$

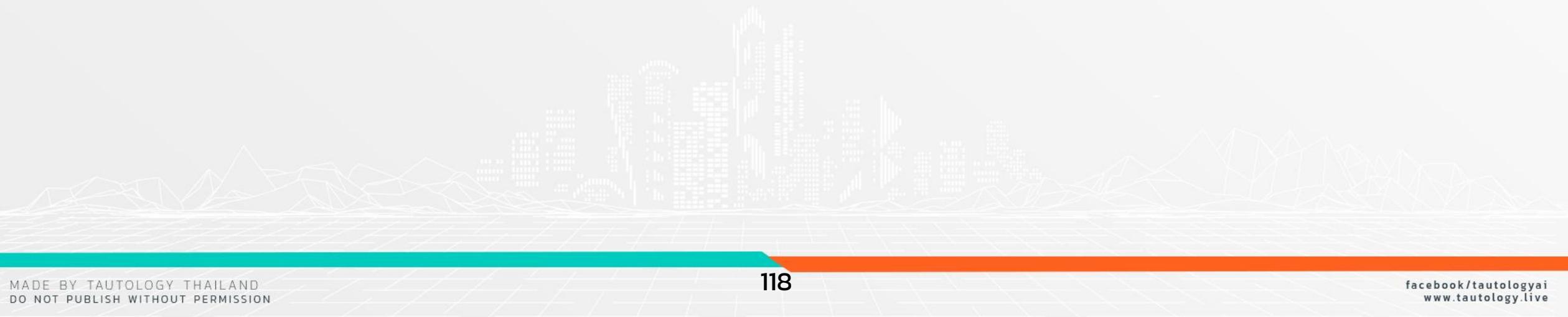


# Prediction

1-Sample

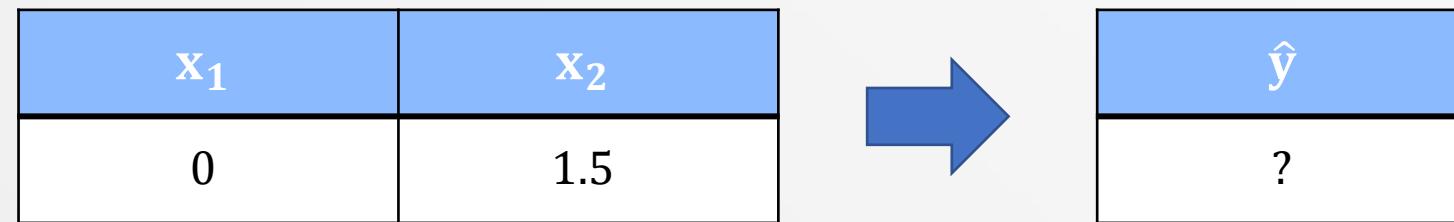
Multi-Sample

Code



# 1-Sample

ตัวอย่างการคำนวณ  $\hat{y}$



# 1-Sample

สมมติว่า พ ของปัญหานี้ที่เราหามาได้คือ

$$\mathbf{w} = \begin{bmatrix} 0 \\ 10.5218 \\ 10.5218 \end{bmatrix}$$

# 1-Sample

ซึ่งทำให้สามารถเขียนสมการ  $\hat{y}$  ได้ดังต่อไปนี้

$$\begin{aligned}\star \quad z &= 0 + 10.5218x_1 + 10.5218x_2 \\ &= 0 + 10.5218(0) + 10.5218(1.5) \\ &= 15.7827\end{aligned}$$

$$\star \quad \hat{y} = \frac{1}{1+e^z} = \frac{1}{1+e^{15.7827}} = 0.99999986015 \approx 1$$

# 1-Sample

ดังนั้น เราจะได้  $\hat{y}$  ของข้อมูลชุดนี้คือ



# Prediction

1-Sample



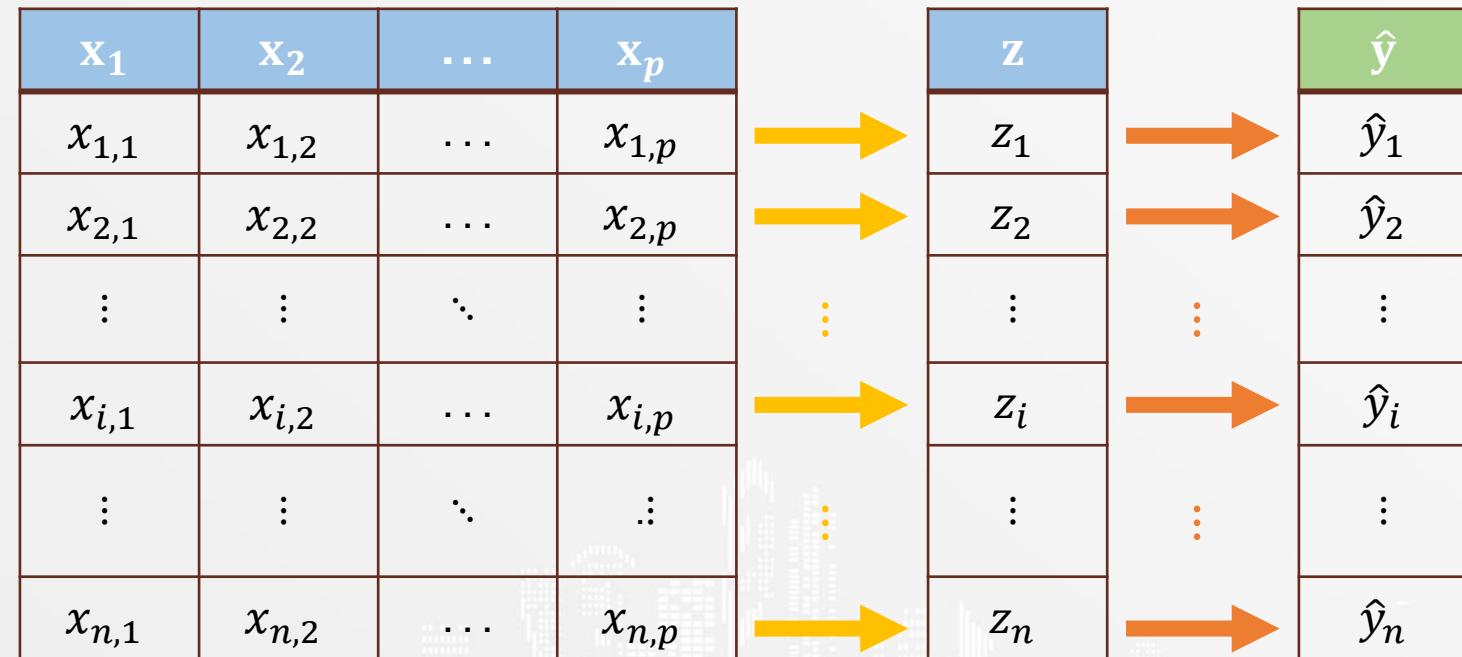
Multi-Sample



Code



# Multi-Sample



# Multi-Sample

$$z = w_0 + w_1 x_1 + w_2 x_2 + \cdots + w_p x_p$$



$$\hat{y} = \frac{1}{1 + e^{-z}}$$



$$z_1 = w_0 + w_1 x_{1,1} + w_2 x_{1,2} + \cdots + w_p x_{1,p}$$



$$\hat{y}_1 = \frac{1}{1 + e^{-z_1}}$$

$$z_2 = w_0 + w_1 x_{2,1} + w_2 x_{2,2} + \cdots + w_p x_{2,p}$$



$$\hat{y}_2 = \frac{1}{1 + e^{-z_2}}$$

:

$$z_i = w_0 + w_1 x_{i,1} + w_2 x_{i,2} + \cdots + w_p x_{i,p}$$



$$\hat{y}_i = \frac{1}{1 + e^{-z_i}}$$

:

$$z_n = w_0 + w_1 x_{n,1} + w_2 x_{n,2} + \cdots + w_p x_{n,p}$$



$$\hat{y}_n = \frac{1}{1 + e^{-z_n}}$$

# Multi-Sample

$$\begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_n \end{bmatrix} = \begin{bmatrix} 1 & x_{1,1} & x_{1,2} & \cdots & x_{1,p} \\ 1 & x_{2,1} & x_{2,2} & \cdots & x_{2,p} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n,1} & x_{n,2} & \cdots & x_{n,p} \end{bmatrix} \begin{bmatrix} w_0 \\ w_1 \\ \vdots \\ w_p \end{bmatrix} \rightarrow \begin{bmatrix} \hat{y}_1 \\ \hat{y}_2 \\ \vdots \\ \hat{y}_n \end{bmatrix} = \begin{bmatrix} \frac{1}{1+e^{-z_1}} \\ \frac{1}{1+e^{-z_2}} \\ \vdots \\ \frac{1}{1+e^{-z_n}} \end{bmatrix}$$

# Multi-Sample

$$\mathbf{z} = X_b \mathbf{w} \rightarrow \hat{\mathbf{y}} = \begin{bmatrix} \frac{1}{1+e^{-z_1}} \\ \frac{1}{1+e^{-z_2}} \\ \vdots \\ \frac{1}{1+e^{-z_n}} \end{bmatrix}$$

โดยที่  $\mathbf{z} = \begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_n \end{bmatrix}$ ,  $X_b = \begin{bmatrix} 1 & x_{1,1} & x_{1,2} & \cdots & x_{1,p} \\ 1 & x_{2,1} & x_{2,2} & \cdots & x_{2,p} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n,1} & x_{n,2} & \cdots & x_{n,p} \end{bmatrix}$ ,  $\mathbf{w} = \begin{bmatrix} w_0 \\ w_1 \\ \vdots \\ w_p \end{bmatrix}$ ,  $\hat{\mathbf{y}} = \begin{bmatrix} \hat{y}_1 \\ \hat{y}_2 \\ \vdots \\ \hat{y}_n \end{bmatrix}$

# Multi-Sample

ตัวอย่างการคำนวณ  $\hat{y}$

$x_1$	$x_2$
0	1.5
1.5	0
-1.5	0
0	-1.5



$\hat{y}$
?
?
?
?

# Multi-Sample

- สมมติว่า  $w$  ของปัญหานี้ที่เราหมายได้คือ

$$w = \begin{bmatrix} 0 \\ 10.5218 \\ 10.5218 \end{bmatrix}$$

- และจากข้อมูลใน dataset เราสามารถเขียน  $X_b$  ได้ดังต่อไปนี้

$$X_b = \begin{bmatrix} 1 & 0 & 1.5 \\ 1 & 1.5 & 0 \\ 1 & -1.5 & 0 \\ 1 & 0 & -1.5 \end{bmatrix}$$

# Multi-Sample

เราคำนวณค่า  $z$  ได้จาก  $z = X_b w$

$$\begin{aligned} z &= \begin{bmatrix} 1 & 0 & 1.5 \\ 1 & 1.5 & 0 \\ 1 & -1.5 & 0 \\ 1 & 0 & -1.5 \end{bmatrix} \begin{bmatrix} 0 \\ 10.5218 \\ 10.5218 \end{bmatrix} \\ &= \begin{bmatrix} 15.7872 \\ 15.7872 \\ -15.7872 \\ -15.7872 \end{bmatrix} \end{aligned}$$

# Multi-Sample

จาก  $\hat{y} = \frac{1}{1+e^{-z}}$  จะได้ว่า

$$\hat{\mathbf{y}} = \begin{bmatrix} \frac{1}{1+e^{-15.7872}} \\ \frac{1}{1+e^{-15.7872}} \\ \frac{1}{1+e^{15.7872}} \\ \frac{1}{1+e^{15.7872}} \end{bmatrix} = \begin{bmatrix} 0.99 \\ 0.99 \\ 0 \\ 0 \end{bmatrix}$$

# Multi-Sample

ดังนั้น เราจะได้  $\hat{y}$  สำหรับข้อมูลชุดนี้คือ

$x_1$	$x_2$
0	1.5
1.5	0
-1.5	0
0	-1.5



$\hat{y}$
0.99
0.99
0
0

# Prediction

1-Sample



Multi-Sample



Code



# Code

ตัวอย่าง code สำหรับหา  $\hat{y}$

$x_1$	$x_2$
0	1.5
1.5	0
-1.5	0
0	-1.5



$\hat{y}$
?
?
?
?

# Code

- Code สำหรับหา ŷ จากข้อมูลของเราโดยที่  $X = \begin{bmatrix} 0 & 1.5 \\ 1.5 & 0 \\ -1.5 & 0 \\ 0 & -1.5 \end{bmatrix}$

```
1 clf.predict(X)
```

```
array([1, 1, 0, 0])
```

# Code

- Code สำหรับหาค่า probability ของ sample กี่จะถูกพยากรณ์ในแต่ละ class

```
1 clf.predict_proba(X)
```

```
array([[1.39841254e-07, 9.99999860e-01],  
       [1.39841254e-07, 9.99999860e-01],  
       [9.99999860e-01, 1.39841254e-07],  
       [9.99999860e-01, 1.39841254e-07]])
```

# Code

ดังนั้น เราจะได้  $\hat{y}$  สำหรับข้อมูลชุดนี้คือ

$x_1$	$x_2$
0	1.5
1.5	0
-1.5	0
0	-1.5



$\hat{y}$
1
1
0
0

# Code

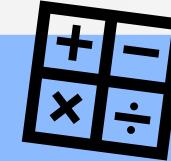


Code for this section



Open File  
**Model Creation (Binary).ipynb**

# Code



MATH

$$\mathbf{z} = \mathbf{X}_b \mathbf{w}$$

$$y = \frac{1}{1 + e^{-z}}$$



CODE

```
1 clf.predict(X)
```

```
array([1, 1, 0, 0])
```

# Prediction

1-Sample



Multi-Sample



Code



# Logistic Regression (Binary)



# Logistic Regression

**Logistic Regression  
(Binary)**



**Logistic Regression  
(Multi-Class)**

**Workshop**

# Logistic Regression (Multi-Class)



# Introduction

# Introduction

Logistic Regression  
with Multiclass

Why Softmax ?

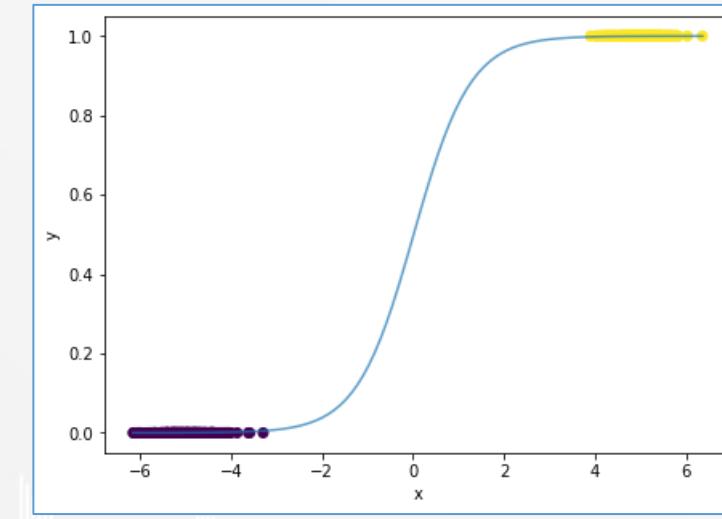
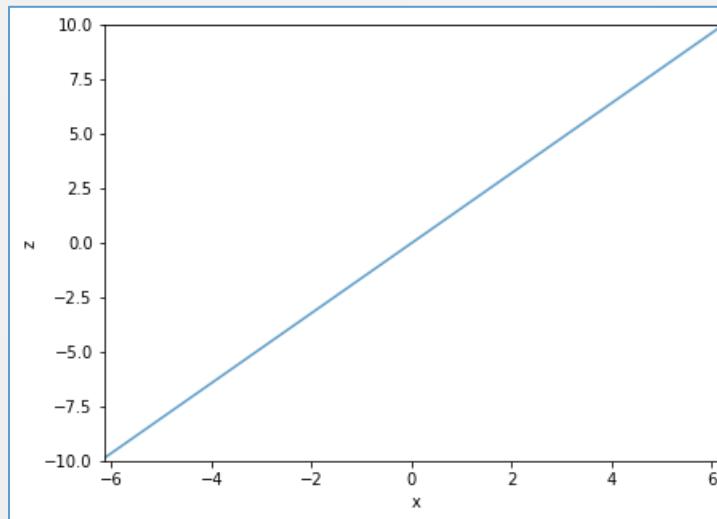
Data for Logistic  
Regression  
(Multi-class)

Pros & Cons

Real World  
Application

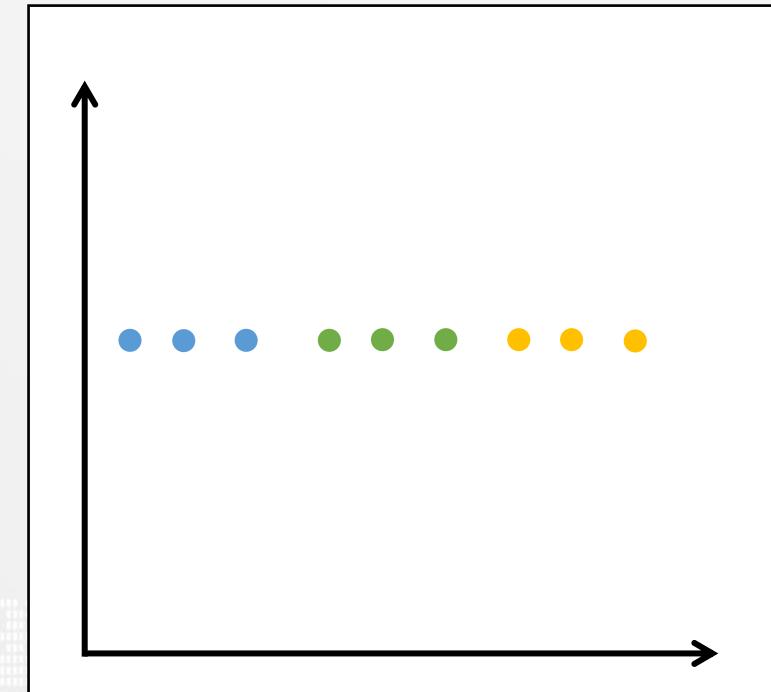
# Logistic Regression with Multiclass

Binary Classification



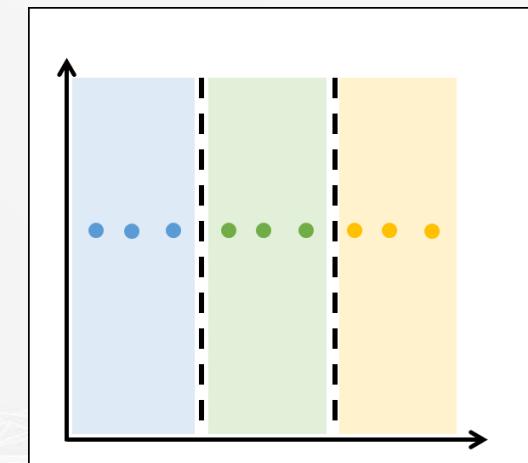
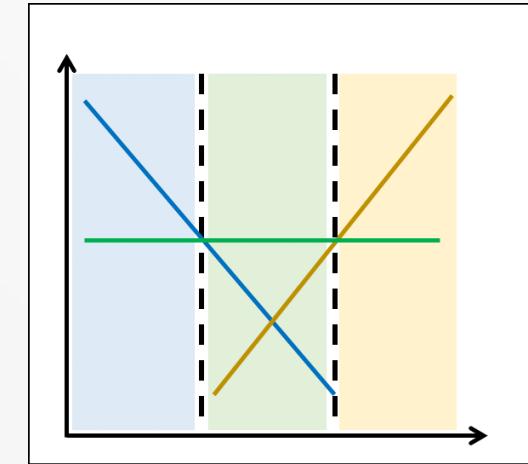
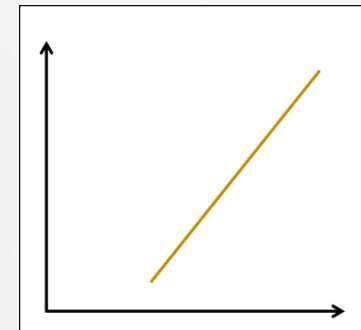
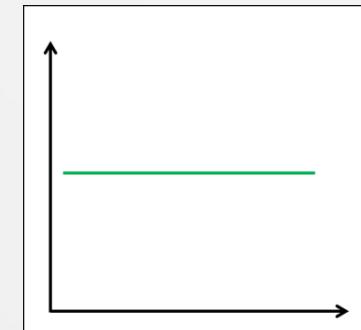
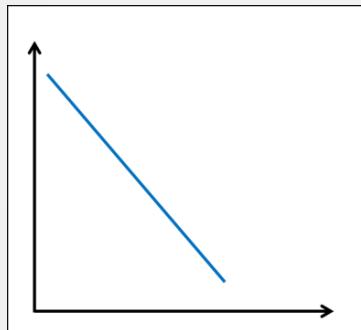
# Logistic Regression with Multiclass

Muti-Class Classification



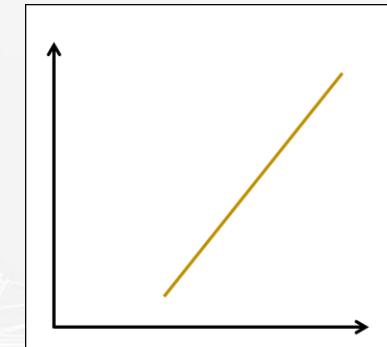
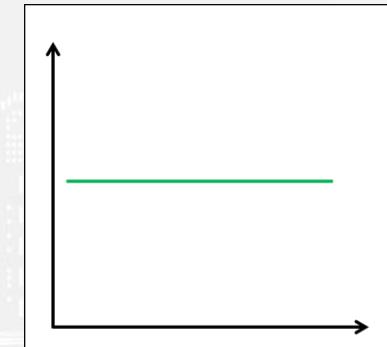
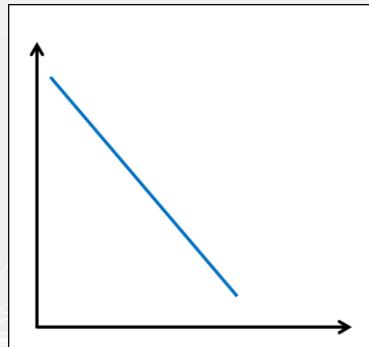
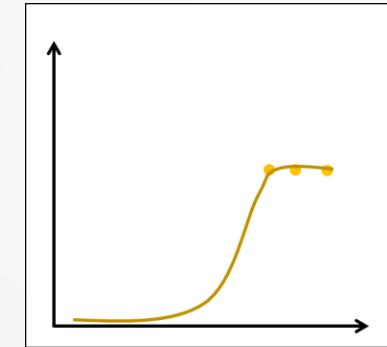
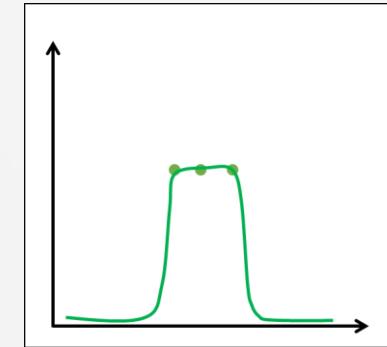
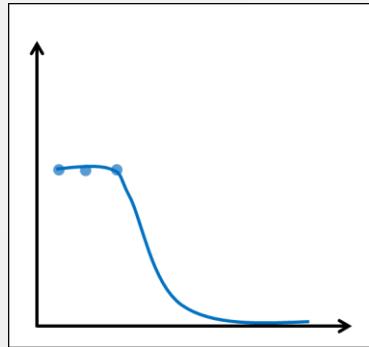
# Logistic Regression with Multiclass

Muti-Class Classification



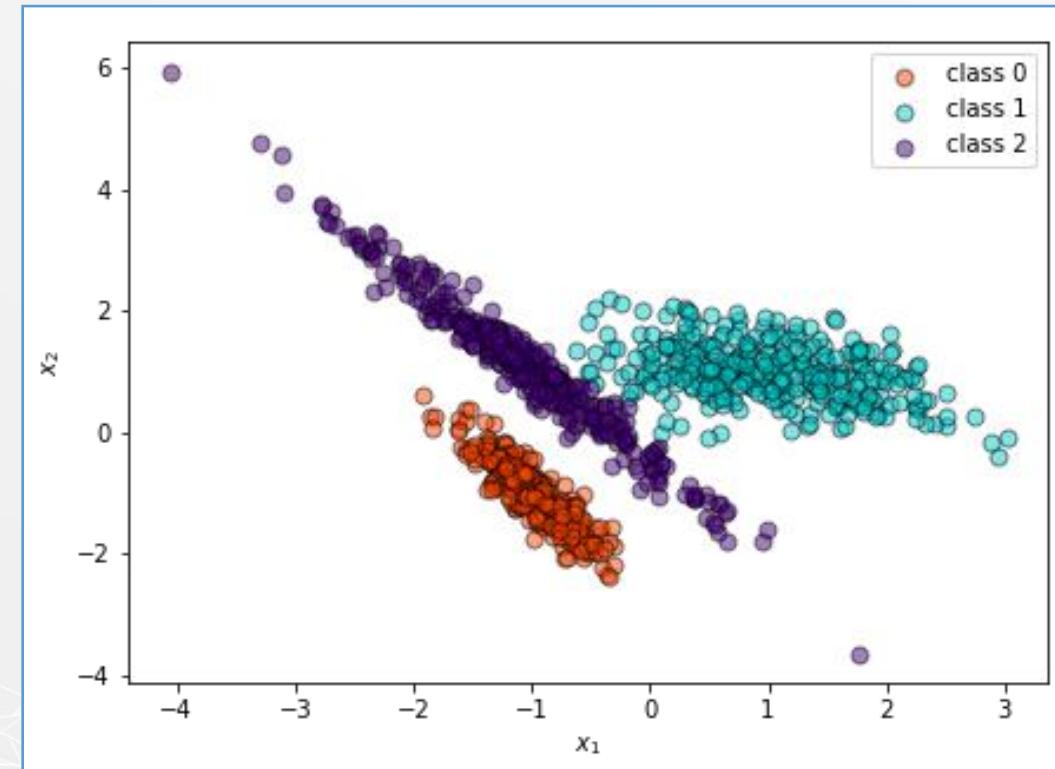
# Logistic Regression with Multiclass

## Muti-Class Classification



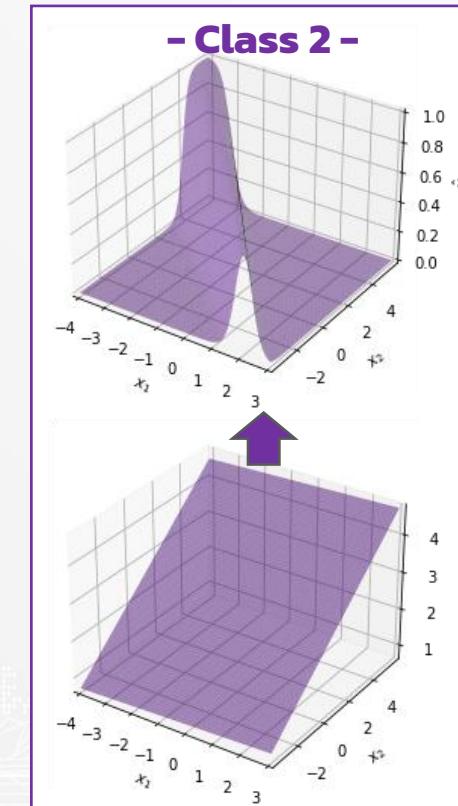
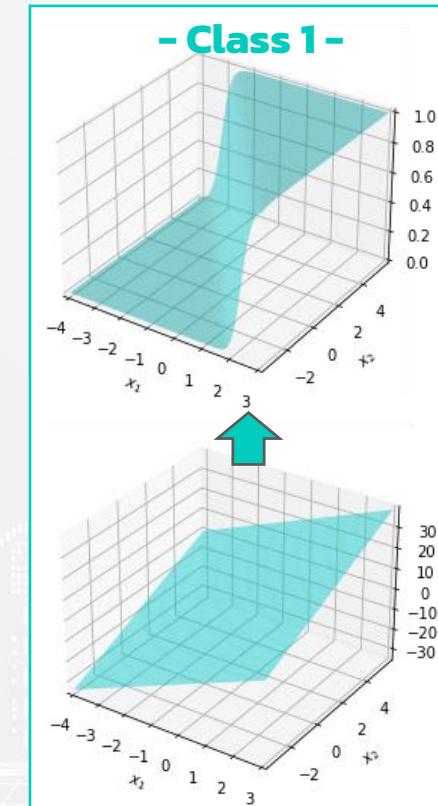
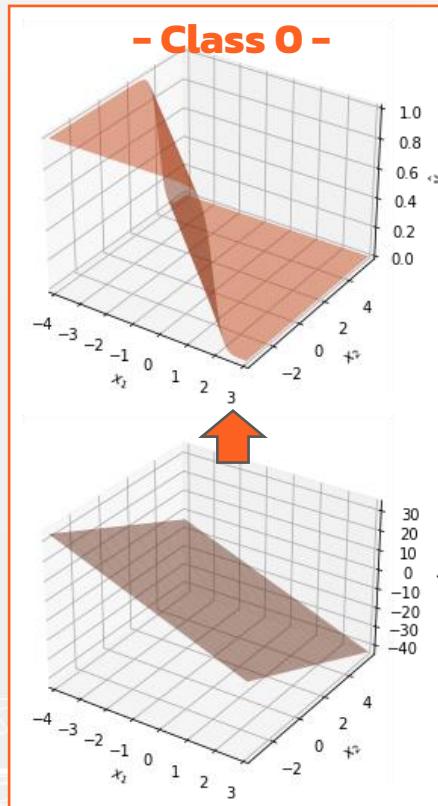
# Logistic Regression with Multiclass

Muti-Class Classification



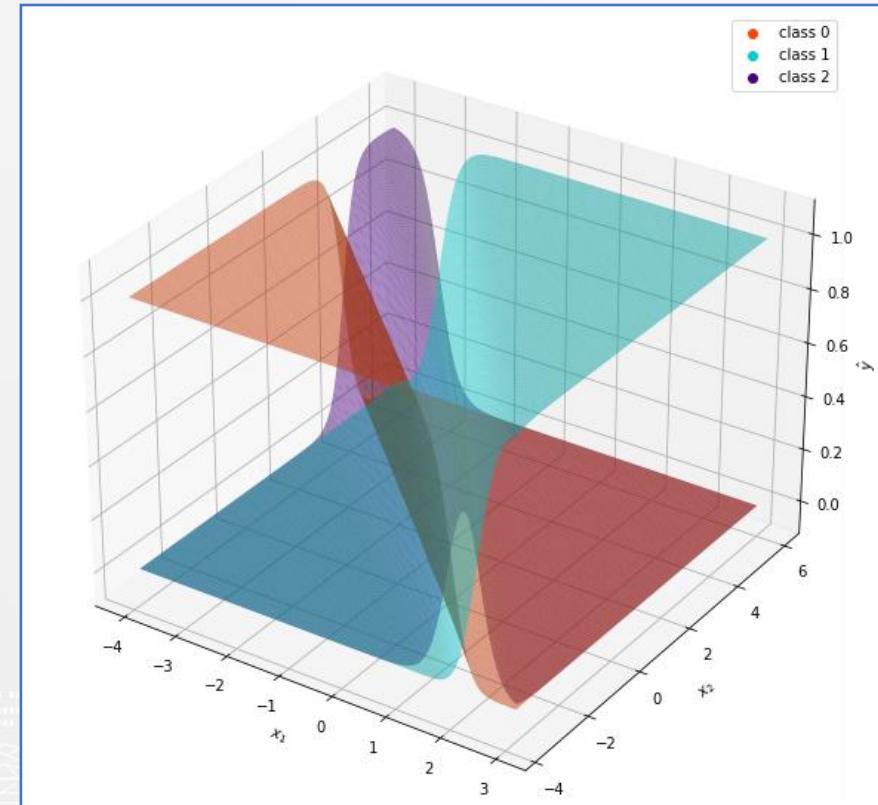
# Logistic Regression with Multiclass

## Muti-Class Classification



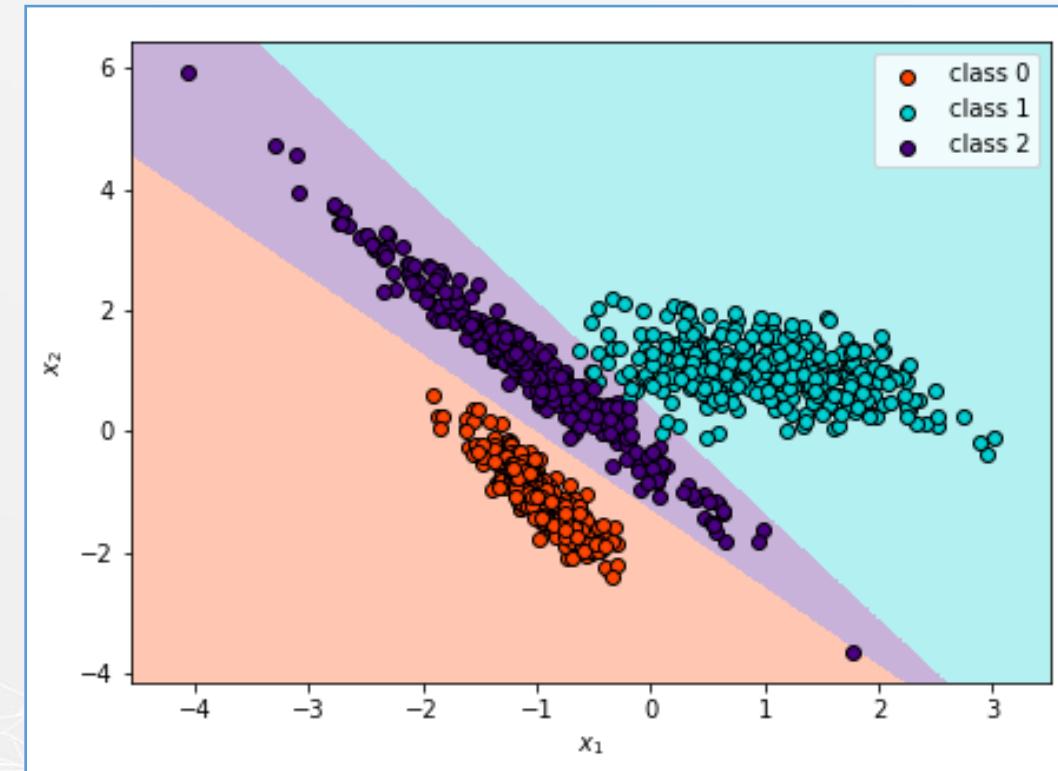
# Logistic Regression with Multiclass

Muti-Class Classification



# Logistic Regression with Multiclass

Muti-Class Classification



# Logistic Regression with Multiclass

สมการคณิตศาสตร์ของ Logistic Regression แบบ **Multiclass**

$$z_m = w_{0,m} + w_{1,m}x_1 + w_{2,m}x_2 + \cdots + w_{p,m}x_p$$

$$\hat{y}_m = \frac{e^{z_m}}{\sum_{c=0}^{k-1} e^{z_c}}$$

โดยที่

- ◆  $\hat{y}_m$  คือ ความน่าจะเป็นที่จะเป็น class ที่  $m$  (predicted target)
- ◆  $z_m$  คือ สมการเชิงเส้น (linear function) ของ class ที่  $m$
- ◆  $x_1, x_2, x_3, \dots, x_p$  คือ ตัวแปรตัว (feature)
- ◆  $w_{0,m}, w_{1,m}, \dots, w_{p,m}$  คือ สัมประสิทธิ์ที่ส่งจาก feature ไป class ที่  $m$  (coefficient)
- ◆  $m \in 0, 1, 2, \dots, k - 1$

# Introduction

**Logistic Regression  
with Multiclass**



Why Softmax ?

Data for Logistic  
Regression  
(Multi-class)

Pros & Cons

Real World  
Application

# Why Softmax ?

**Why**  $\hat{y}_m = \frac{e^{z_m}}{\sum_{c=0}^{k-1} e^{z_c}}$  ?

# Why Softmax ?

**Ans :** เราต้องการให้  $\hat{y}_m$  อยู่ในรูปของความน่าจะเป็น ดังนั้น สมการของ  $\hat{y}_m$  ต้องอยู่ในรูปของ  $\frac{f(z_m)}{\sum_{c=0}^{k-1} f(z_c)}$

# Why Softmax ?

Why  $e^z$  ?

# Why Softmax ?

Logic ของ  $f(z)$  ที่ทำให้  $\hat{y}_m$  มีสมบัติของความน่าจะเป็น

1.  $f(z)$  มีค่ามากกว่าหรือเท่ากับ 0 และเป็น monotone function
2. เมื่อ  $z$  ต่างกัน เรายากให้  $f(z)$  ต่างกันอย่างชัดเจน
3. เราซوب  $f(z)$  กี diff ได้

# Why Softmax ?

- ◆  $f(z) = z$
- ◆  $f(z) = |z|$
- ◆  $f(z) = z^2$
- ◆  $f(z) = z^n$
- ◆  $f(z) = \log z$
- ◆  $f(z) = a^z$
- ◆  $f(z) = \sigma(z)$
- ◆  $f(z) = relu(z)$
- ◆  $f(z) = \sin(z)$
- ◆  $f(z) = \cos(z)$

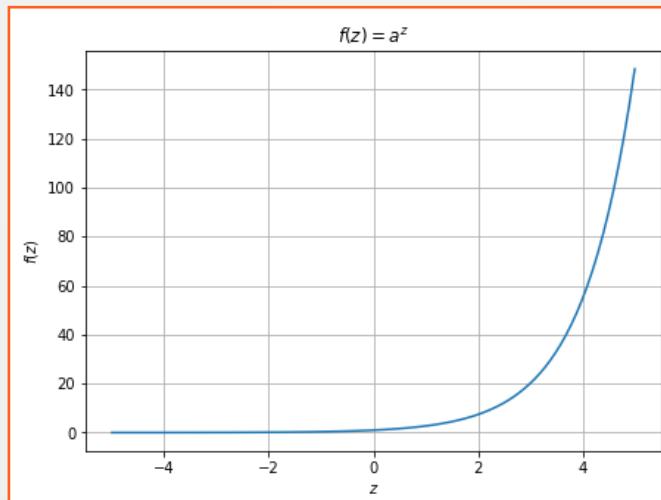
# Why Softmax ?

1.  $f(z)$  มีค่ามากกว่าหรือเท่ากับ 0 และเป็น monotone function

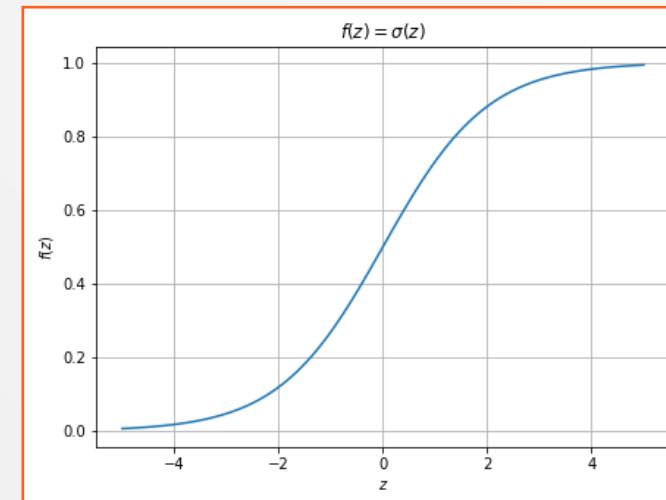
- ◆  ~~$f(z) = z$~~
- ◆  ~~$f(z) = |z|$~~
- ◆  ~~$f(z) = z^2$~~
- ◆  ~~$f(z) = z^n$~~
- ◆  ~~$f(z) = \log z$~~
- ◆  $f(z) = a^z$
- ◆  $f(z) = \sigma(z)$
- ◆  $f(z) = \text{relu}(z)$
- ◆  ~~$f(z) = \sin(z)$~~
- ◆  ~~$f(z) = \cos(z)$~~

# Why Softmax ?

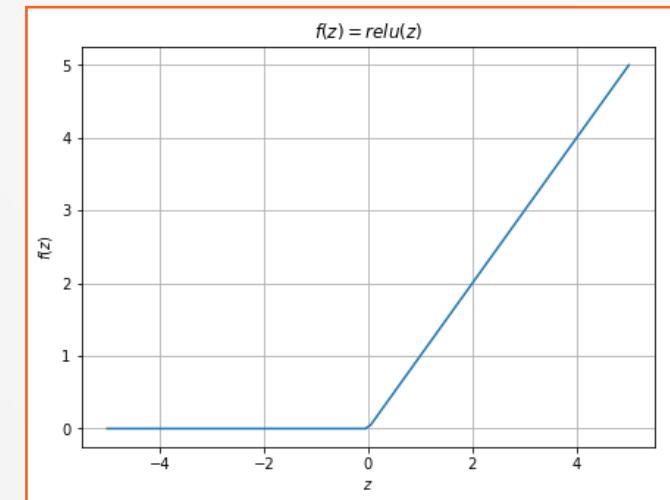
1.  $f(z)$  มีค่ามากกว่าหรือเท่ากับ 0 และเป็น monotone function



$$f(z) = a^z$$



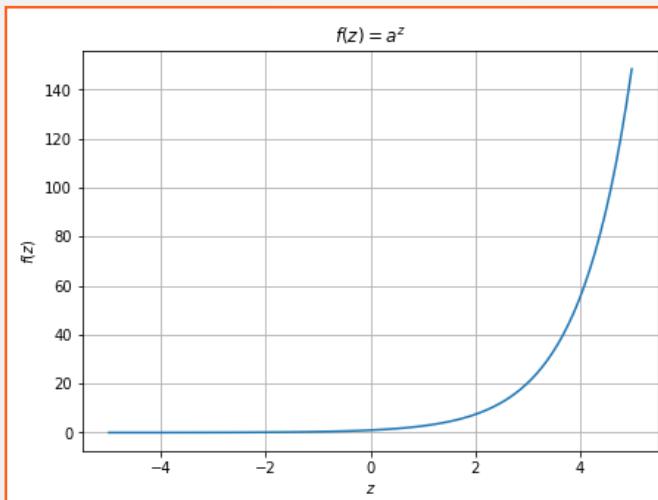
$$f(z) = \sigma(z)$$



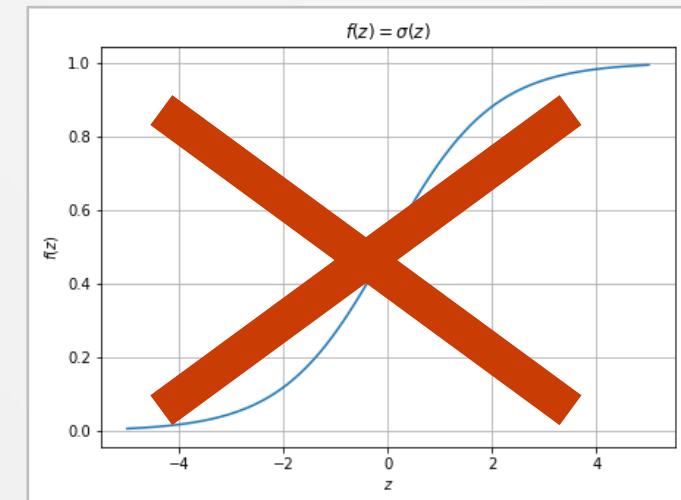
$$f(z) = \text{relu}(z)$$

# Why Softmax ?

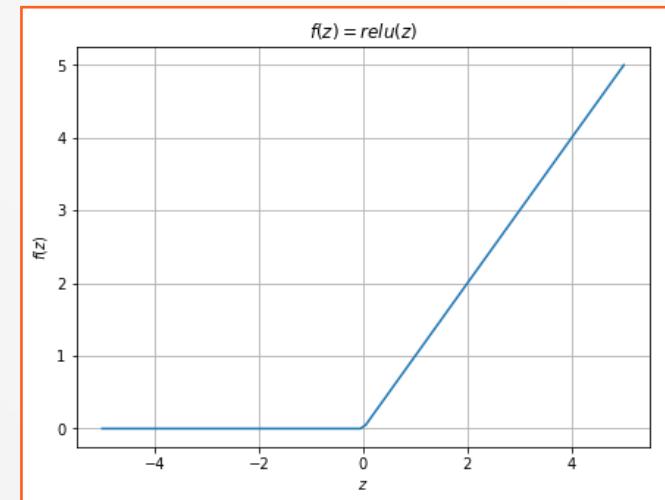
2. เมื่อ  $z$  ต่างกัน เราอยากให้  $f(z)$  ต่างกันอย่างชัดเจน



$$f(z) = a^z$$



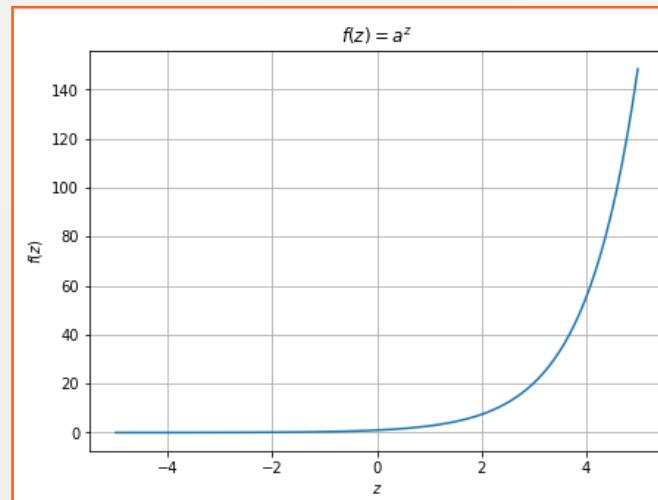
$$f(z) = \sigma(z)$$



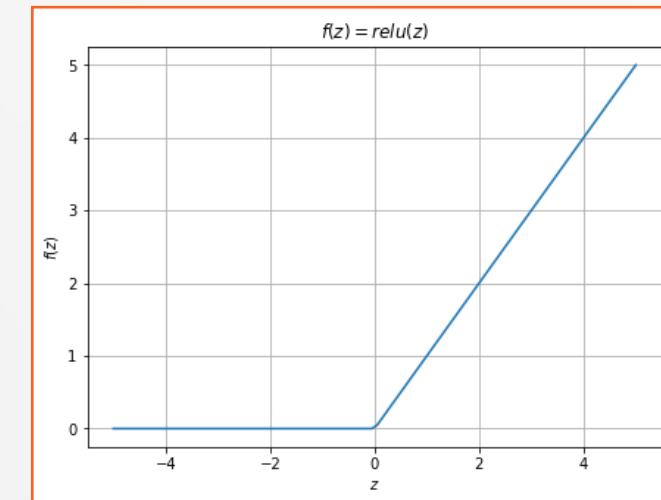
$$f(z) = \text{relu}(z)$$

# Why Softmax ?

- เมื่อ  $z$  ต่างกัน เราอยากให้  $f(z)$  ต่างกันอย่างชัดเจน



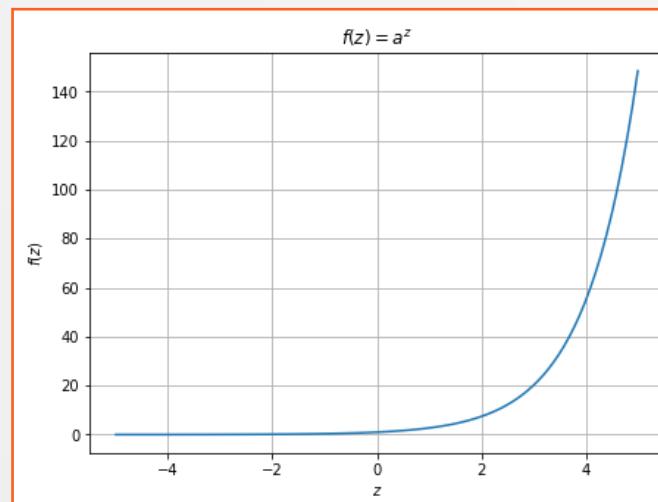
$$f(z) = a^z$$



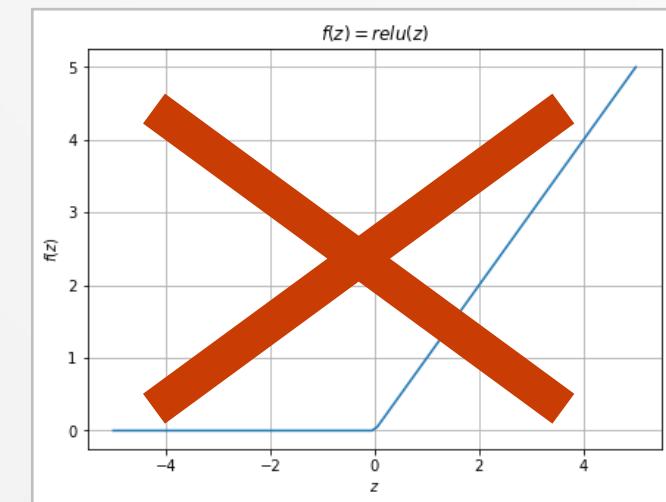
$$f(z) = \text{relu}(z)$$

# Why Softmax ?

3. เราชอบ  $f(z)$  ที่ diff ได้



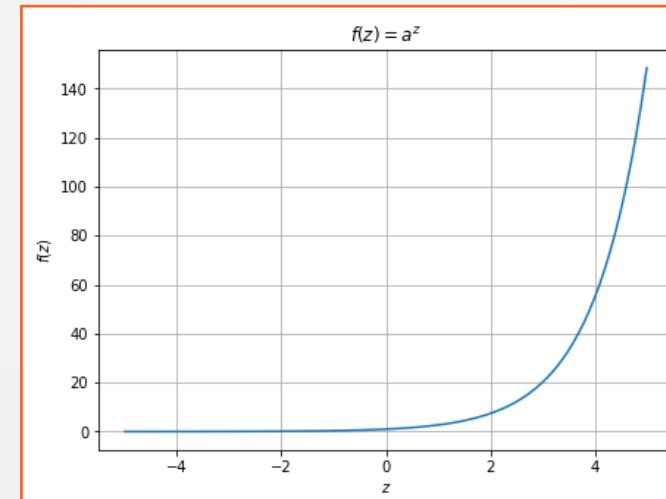
$$f(z) = a^z$$



$$f(z) = \text{relu}(z)$$

# Why Softmax ?

3. เราชอบ  $f(z)$  ที่ diff ได้



$$f(z) = a^z$$

# Why Softmax ?

Why  $a = e$  ?

\*\*  $e = 2.718281828459045$

# Why Softmax ?

$$\text{Ans: } \frac{\partial a^z}{\partial z} = a^z \log a$$

$$\frac{\partial e^z}{\partial z} = e^z$$

# Why Softmax ?

เราต้องการให้  $\hat{y}_m$  อยู่ในรูปของความน่าจะเป็น  
ดังนั้น สมการของ  $\hat{y}_m$  ต้องอยู่ในรูปของ  $\frac{f(z_m)}{\sum_{c=0}^{k-1} f(z_c)}$

# Why Softmax ?

$$\hat{y}_m = \frac{e^{z_m}}{\sum_{c=0}^{k-1} e^{z_c}}$$

# Introduction

**Logistic Regression  
with Multiclass**



**Why Softmax ?**



**Data for Logistic  
Regression  
(Multi-class)**



**Pros & Cons**

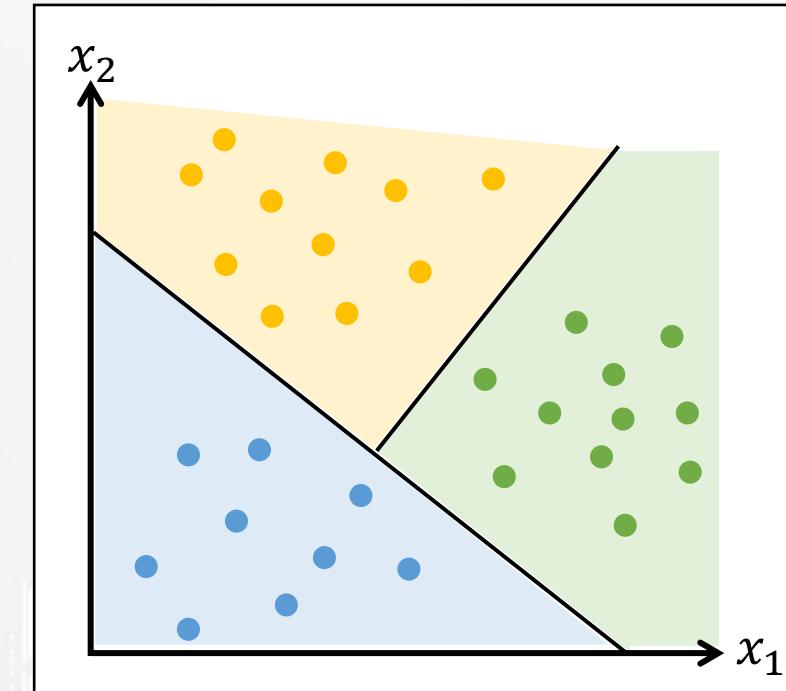
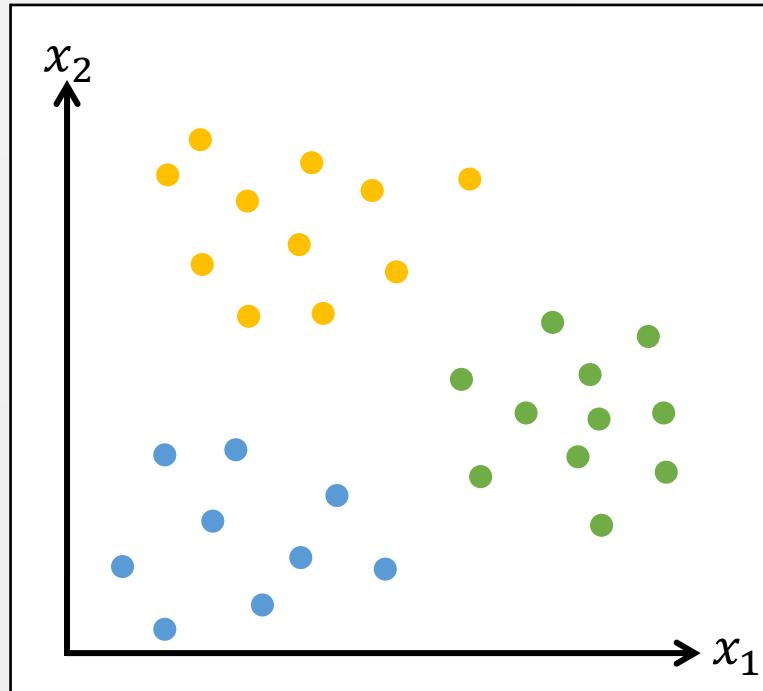


**Real World  
Application**



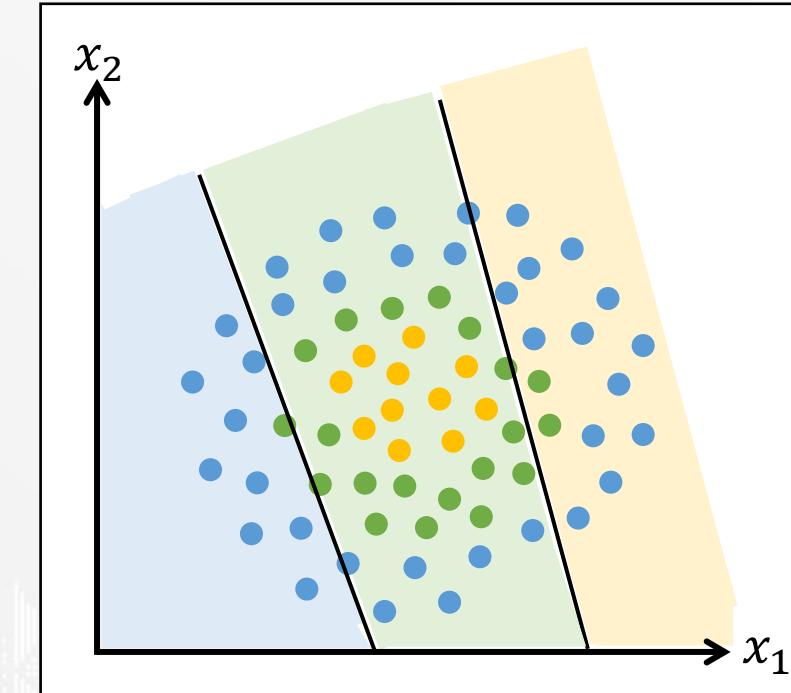
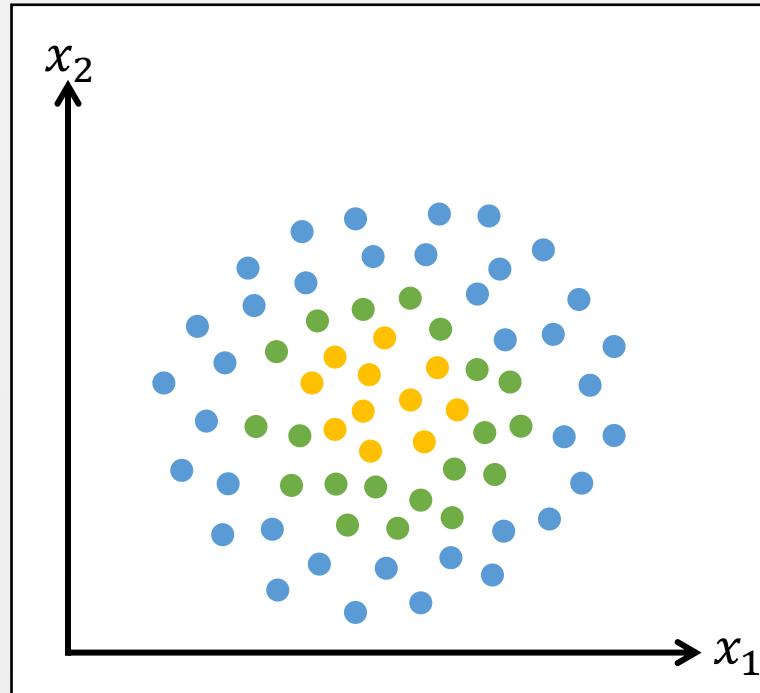
# Data for Logistic Regression (Multi-class)

ตัวอย่างของข้อมูลที่เหมาะสมกับ Logistic Regression



# Data for Logistic Regression (Multi-class)

ตัวอย่างของข้อมูลที่ไม่เหมาะสมกับ Logistic Regression



# Introduction

**Logistic Regression  
with Multiclass**



**Why Softmax ?**



**Data for Logistic  
Regression  
(Multi-class)**



**Pros & Cons**



**Real World  
Application**



# Pros & Cons

## ข้อดี

- เป็น algorithm ที่เรียบง่าย
- เป็น algorithm ที่เหมาะสมที่สุดเมื่อข้อมูลสามารถแบ่งจากกันได้อย่างสมบูรณ์แบบด้วยสมการเชิงเส้น

## ข้อเสีย

- sensitive กับ outliers

## ข้อจำกัด

- ใช้งานได้กับข้อมูลที่สามารถแบ่งจากกันได้อย่างสมบูรณ์แบบด้วยสมการเชิงเส้นเท่านั้น

# Introduction

**Logistic Regression  
with Multiclass**



**Why Softmax ?**



**Data for Logistic  
Regression  
(Multi-class)**



**Pros & Cons**



**Real World  
Application**



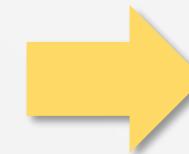
# Real World Application



**การจำแนกผู้ป่วยที่เป็นโรคเบาหวาน  
(ไม่เป็นเบาหวาน, มีโอกาสเสี่ยงสูง, เป็น  
เบาหวาน)**  
โดยจะพิจารณาจาก เพศ, อายุ, ระดับการ  
เคลื่อนไหวในชีวิตประจำวัน เป็นต้น

<https://pubmed.ncbi.nlm.nih.gov/26714019/>

# Real World Application



## การวิเคราะห์ความรุนแรงของอุบัติเหตุ บนท้องถนน

(เสียหายเฉพาะทรัพย์สิน, บาดเจ็บเล็กน้อย,  
พิการ, เสียชีวิต)

โดยพิจารณาจาก อายุของคนขับ, ประเภทของ  
ยานพาหนะ, แสงไฟบนท้องถนน, เป็นถนนที่สร้าง  
เสร็จแล้วหรือกำลังก่อสร้าง เป็นต้น

<https://www.scirp.org/journal/paperinformation.aspx?paperid=77395>

# Real World Application



**การวิเคราะห์ performance ของหุ้นใน  
อนาคต  
(ดี, อยู่ในค่าเฉลี่ยของตลาด, ไม่ดี)**  
โดยพิจารณาจาก กำไรสุทธิ, กำไรต่อหุ้น, มูลค่า  
ทางบัญชี, สัดส่วนกำไรเกียบปีก่อนหน้า เป็นต้น

<https://www.proquest.com/openview/51d5560e3bea3125896d211c7932d7a9/1?pq-origsite=gscholar&cbl=1056382>

# Introduction

**Logistic Regression  
with Multiclass**



**Why Softmax ?**



**Data for Logistic  
Regression  
(Multi-class)**



**Pros & Cons**



**Real World  
Application**

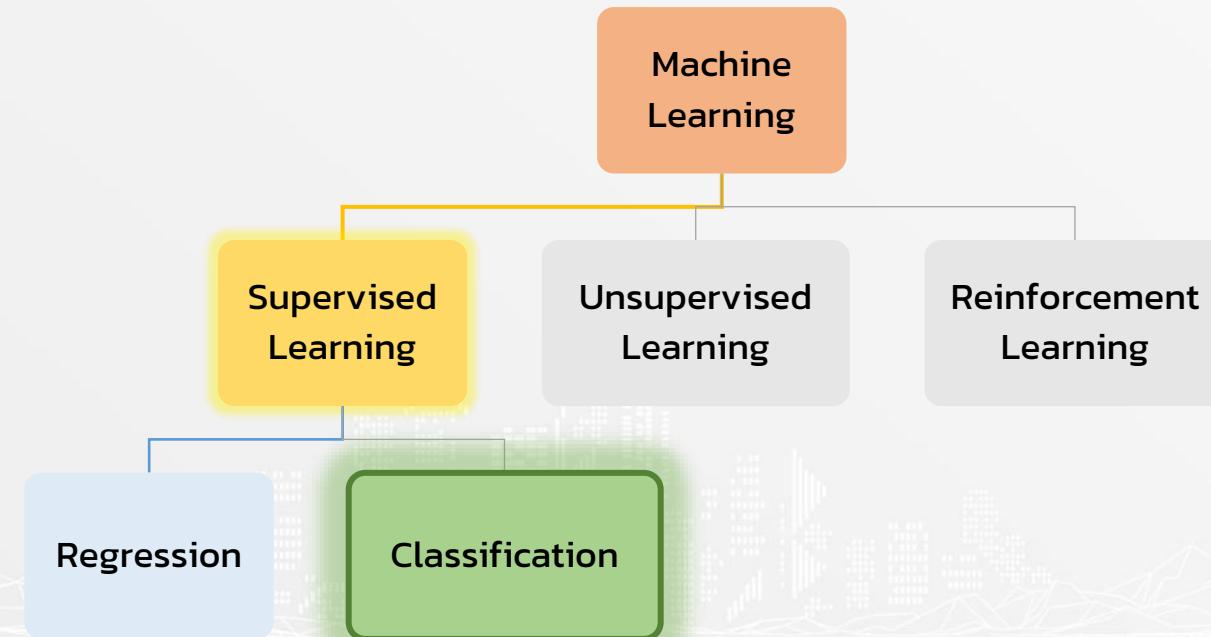


# Logistic Regression (Multi-Class)



# Logistic Regression

**Logistic Regression** ລົງທະບຽນໃນ algorithm ປະເກດ supervised learning



# Concept of Supervised Learning

Data   ⇒  Model   ⇒  Prediction

# Model Creation

# Model Creation

Assumption

Real Face of the Model

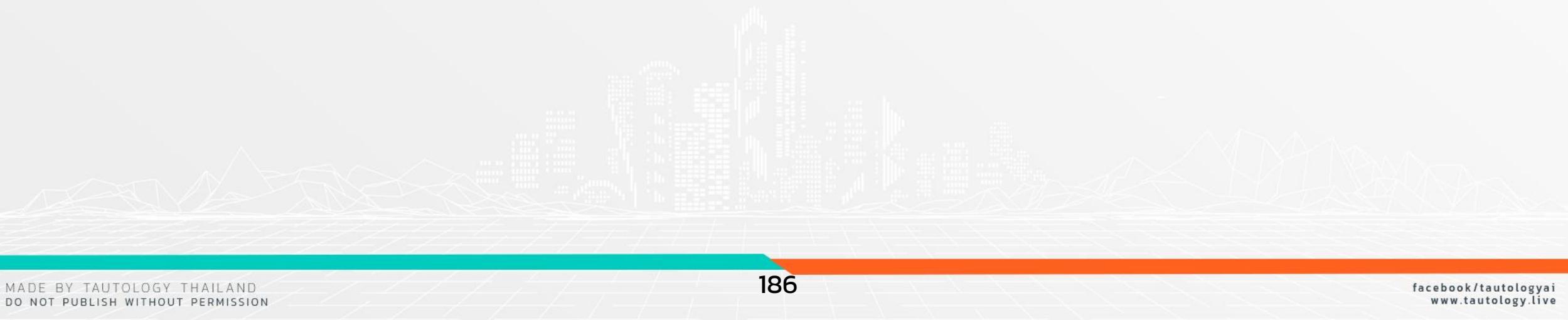
Cost Function and Cost Landscape

How to Create Model (Math)

How to Create Model (Code)

# Assumption

1. Linear Relationship to Log odds
2. No Missing Features
3. No Multicollinearity



# Model

**Assumption**



Real Face of the Model

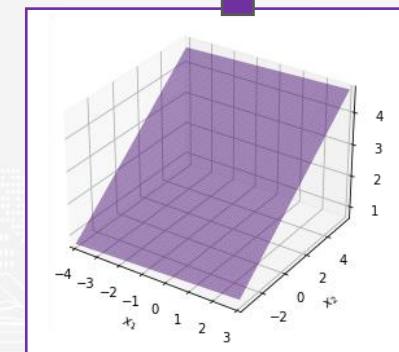
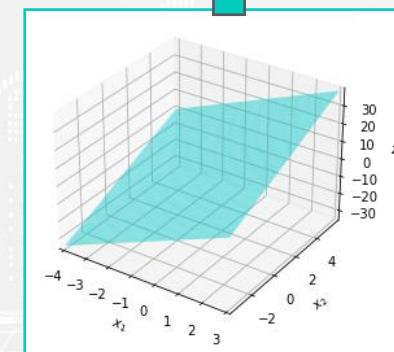
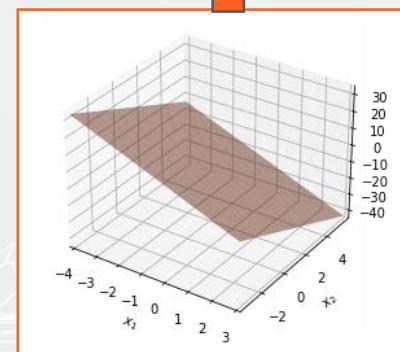
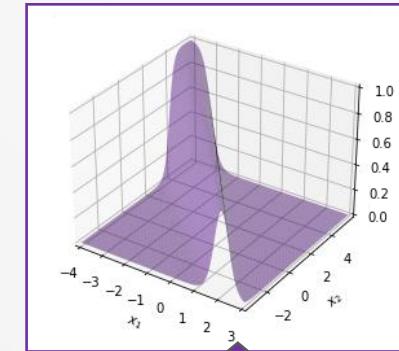
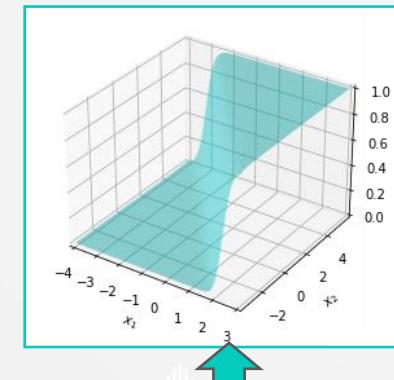
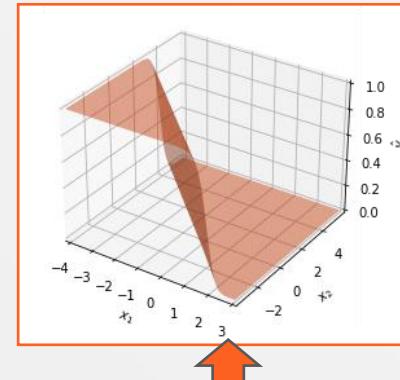
Cost Function and Cost Landscape

How to Create Model (Math)

How to Create Model (Code)

# Real Face of the Model

**Logistic regression** คือ โมเดลคณิตศาสตร์ที่ส่งค่าสมการเชิงเส้น (linear function) ไปในช่วง  $(0,1)$  เพื่อจำแนกประเภทข้อมูล



# Real Face of the Model

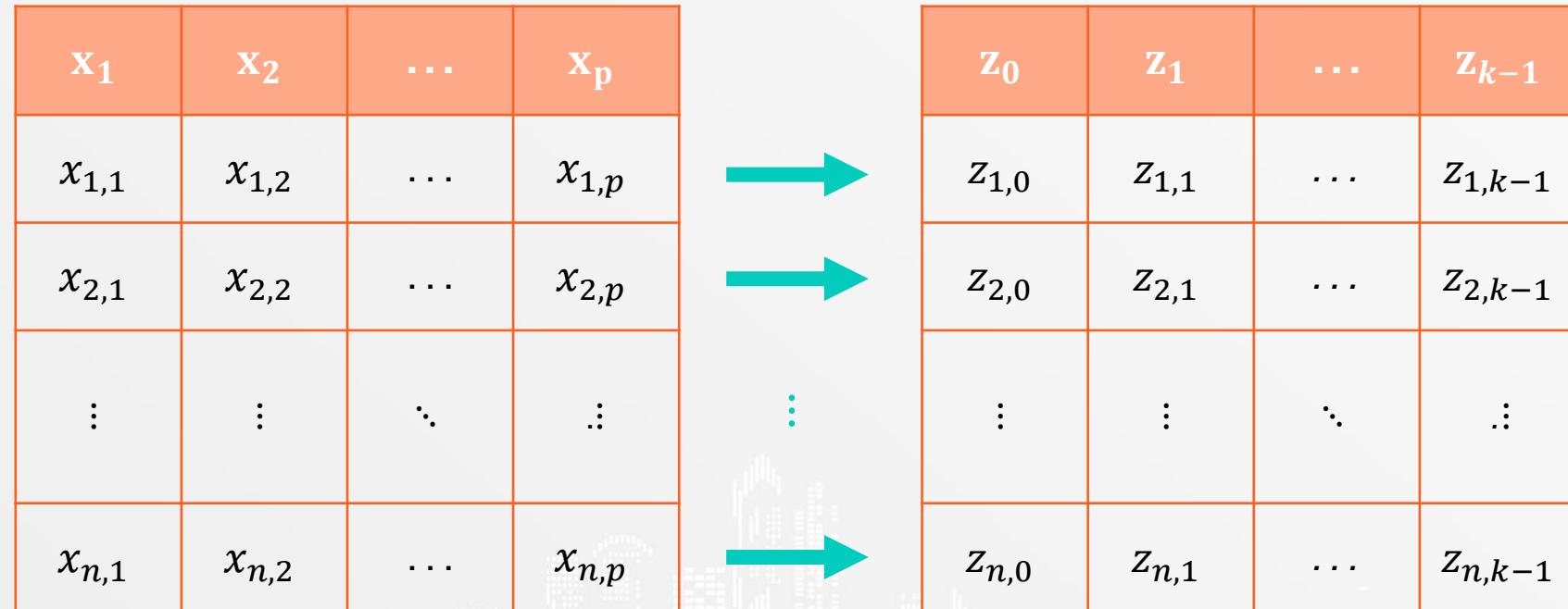
**Logistic regression** คือ โมเดลคณิตศาสตร์ที่จำแนกประเภทของข้อมูล ด้วยการสร้างสมการเชิงเส้น (linear function)

$$z_m = w_{0,m} + w_{1,m}x_1 + w_{2,m}x_2 + \cdots + w_{p,m}x_p$$

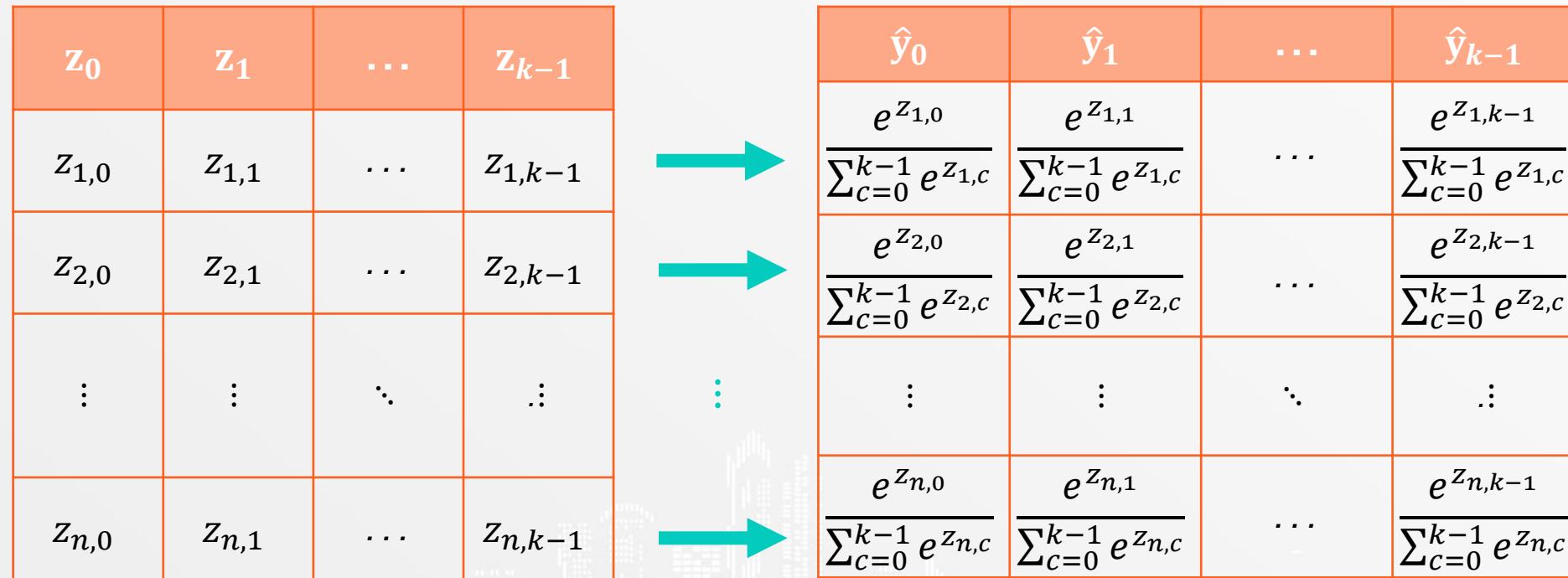
$$\hat{y}_m = \frac{e^{z_m}}{\sum_{c=0}^{k-1} e^{z_c}}$$

- โดยที่
- ◆  $\hat{y}_m$  คือ ความน่าจะเป็นที่จะเป็น class ที่  $m$  (predicted target)
  - ◆  $z_m$  คือ สมการเชิงเส้น (linear function) ของ class ที่  $m$
  - ◆  $x_1, x_2, x_3, \dots, x_p$  คือ ตัวแปรต้น (feature)
  - ◆  $w_{0,m}, w_{1,m}, w_{2,m}, \dots, w_{p,m}$  คือ สัมประสิทธิ์ที่ส่งจาก feature ไป class ที่  $m$  (coefficient)
  - ◆  $m \in 0, 1, 2, \dots, k - 1$

# Real Face of the Model



# Real Face of the Model



# Real Face of the Model

$$z_0 = w_{0,0} + w_{1,0}x_1 + w_{2,0}x_2 + \cdots + w_{p,0}x_p$$



$$z_{1,0} = w_{0,0} + w_{1,0}x_{1,1} + w_{2,0}x_{1,2} + \cdots + w_{p,0}x_{1,p}$$

$$z_{2,0} = w_{0,0} + w_{1,0}x_{2,1} + w_{2,0}x_{2,2} + \cdots + w_{p,0}x_{2,p}$$

⋮

$$z_{n,0} = w_{0,0} + w_{1,0}x_{n,1} + w_{2,0}x_{n,2} + \cdots + w_{p,0}x_{n,p}$$

# Real Face of the Model

$$z_1 = w_{0,1} + w_{1,1}x_1 + w_{2,1}x_2 + \cdots + w_{p,1}x_p$$



$$z_{1,1} = w_{0,1} + w_{1,1}x_{1,1} + w_{2,1}x_{1,2} + \cdots + w_{p,1}x_{1,p}$$

$$z_{2,1} = w_{0,1} + w_{1,1}x_{2,1} + w_{2,1}x_{2,2} + \cdots + w_{p,1}x_{2,p}$$

⋮

$$z_{n,1} = w_{0,1} + w_{1,1}x_{n,1} + w_{2,1}x_{n,2} + \cdots + w_{p,1}x_{n,p}$$

# Real Face of the Model

$$z_{k-1} = w_{0,k-1} + w_{1,k-1}x_1 + w_{2,k-1}x_2 + \cdots + w_{p,k-1}x_p$$



$$z_{1,k-1} = w_{0,k-1} + w_{1,k-1}x_{1,1} + w_{2,k-1}x_{1,2} + \cdots + w_{p,k-1}x_{1,p}$$

$$z_{2,k-1} = w_{0,k-1} + w_{1,k-1}x_{2,1} + w_{2,k-1}x_{2,2} + \cdots + w_{p,k-1}x_{2,p}$$

⋮

$$z_{n,k-1} = w_{0,k-1} + w_{1,k-1}x_{n,1} + w_{2,k-1}x_{n,2} + \cdots + w_{p,k-1}x_{n,p}$$

# Real Face of the Model



“ เราก็ต้องการหา

$$w_{0,0}, w_{1,0}, w_{2,0}, \dots, w_{p,0}$$

$$w_{0,1}, w_{1,1}, w_{2,1}, \dots, w_{p,1}$$

⋮

$$w_{0,k-1}, w_{1,k-1}, w_{2,k-1}, \dots, w_{p,k-1}$$

ที่ทำให้ cost function ต่ำที่สุด ”

# Model Creation

**Assumption**



**Real Face of the Model**



**Cost Function and Cost Landscape**

**How to Create Model (Math)**



**How to Create Model (Code)**



# Cost Function and Cost Landscape

**Cost function** กี่เราจะใช้ในการสร้าง model คือ

$$-\frac{1}{n} \sum_{i=1}^n \sum_{c=0}^{k-1} y_{i,c} \log(\hat{y}_{i,c})$$

โดยสูตรข้างต้นมีชื่อว่า **Cross Entropy**

# Model Creation

**Assumption**



**Real Face of the Model**



**Cost Function and Cost Landscape**



**How to Create Model (Math)**

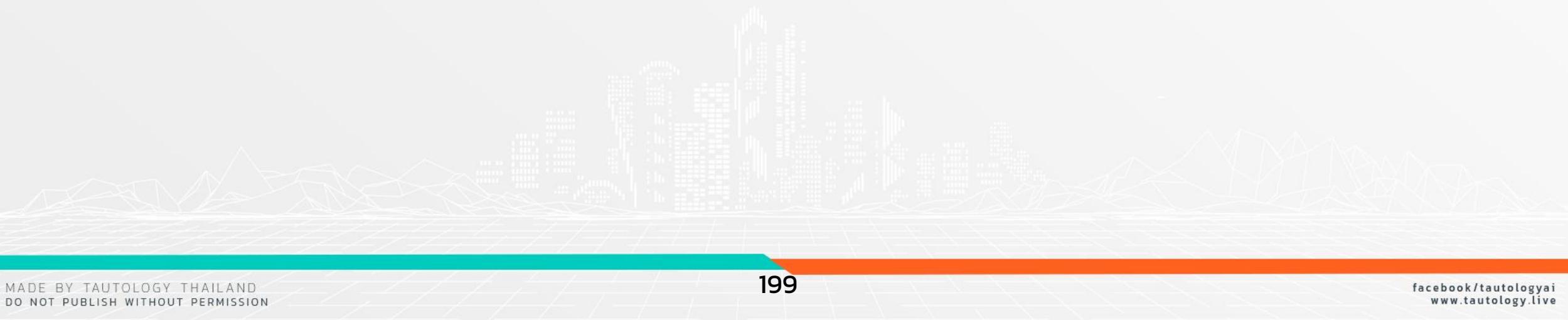


**How to Create Model (Code)**



# How to Create Model (Math)

- Gradient Descent (Logistic Regression)
- Calculation Example



# Gradient Descent (Logistic Regression)

## Equation of Gradient Descent

$$w_d = w_d - \alpha \frac{\partial Cost}{\partial w_d}$$

โดย ◆  $\alpha$  คือ ค่าที่ใช้ควบคุม step size ของ  $w_d (> 0)$

# Gradient Descent (Logistic Regression)

เมื่อพิจารณา  $w$  ที่ส่งไปยัง  $\mathbf{z}_0$  จะได้ว่า

$$w_{0,0} = w_{0,0} - \alpha \frac{\partial Cost}{\partial w_{0,0}}$$

$$w_{1,0} = w_{1,0} - \alpha \frac{\partial Cost}{\partial w_{1,0}}$$

⋮

$$w_{p,0} = w_{p,0} - \alpha \frac{\partial Cost}{\partial w_{p,0}}$$

# Gradient Descent (Logistic Regression)

เมื่อพิจารณา  $w$  ที่ส่งไปยัง  $\mathbf{z}_1$  จะได้ว่า

$$w_{0,1} = w_{0,1} - \alpha \frac{\partial Cost}{\partial w_{0,1}}$$

$$w_{1,1} = w_{1,1} - \alpha \frac{\partial Cost}{\partial w_{1,1}}$$

⋮

$$w_{p,1} = w_{p,1} - \alpha \frac{\partial Cost}{\partial w_{p,1}}$$

# Gradient Descent (Logistic Regression)

เมื่อพิจารณา  $w$  ที่ส่งไปยัง  $\mathbf{z}_{k-1}$  จะได้ว่า

$$w_{0,k-1} = w_{0,k-1} - \alpha \frac{\partial Cost}{\partial w_{0,k-1}}$$

$$w_{1,k-1} = w_{1,k-1} - \alpha \frac{\partial Cost}{\partial w_{1,k-1}}$$

⋮

$$w_{p,k-1} = w_{p,k-1} - \alpha \frac{\partial Cost}{\partial w_{p,k-1}}$$

# Gradient Descent (Logistic Regression)

เมื่อพิจารณาในทุกมิติพร้อมกัน จะได้ว่า

$$\begin{bmatrix} w_{0,0} & w_{0,1} & \cdots & w_{0,k-1} \\ w_{1,0} & w_{1,1} & \cdots & w_{1,k-1} \\ \vdots & \vdots & \ddots & \vdots \\ w_{p,0} & w_{p,1} & \cdots & w_{p,k-1} \end{bmatrix} = \begin{bmatrix} w_{0,0} & w_{0,1} & \cdots & w_{0,k-1} \\ w_{1,0} & w_{1,1} & \cdots & w_{1,k-1} \\ \vdots & \vdots & \ddots & \vdots \\ w_{p,0} & w_{p,1} & \cdots & w_{p,k-1} \end{bmatrix} - \alpha \begin{bmatrix} \frac{\partial Cost}{\partial w_{0,0}} & \frac{\partial Cost}{\partial w_{0,1}} & \cdots & \frac{\partial Cost}{\partial w_{0,k-1}} \\ \frac{\partial Cost}{\partial w_{1,0}} & \frac{\partial Cost}{\partial w_{1,1}} & \cdots & \frac{\partial Cost}{\partial w_{1,k-1}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial Cost}{\partial w_{p,0}} & \frac{\partial Cost}{\partial w_{p,1}} & \cdots & \frac{\partial Cost}{\partial w_{p,k-1}} \end{bmatrix}$$

# Gradient Descent (Logistic Regression)

## Equation of Gradient Descent

$$W = W - \alpha \nabla Cost$$

โดย ◆  $\alpha$  คือ ค่าที่ใช้ควบคุม step size ของ  $W$

# Gradient Descent (Logistic Regression)

## Equation of Gradient Descent for Logistic Regression

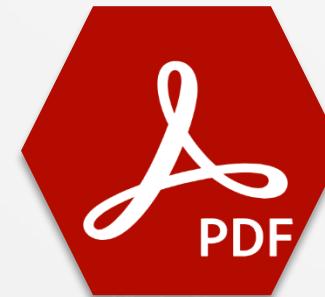
$$W = W + \frac{\alpha}{n} X_b^T (Y - \hat{Y})$$

โดย ◆  $\alpha$  คือ ค่าที่ใช้ควบคุม step size ของ  $W$

# Gradient Descent (Logistic Regression)



Derivation of Gradient Descent  
for Logistic Regression



Open File  
**Derive\_GD\_LoR.pdf**

# How to Create Model (Math)

- Gradient Descent (Logistic Regression)**
- Calculation Example



# Calculation Example

ตัวอย่างการคำนวณ  $W$  สำหรับ logistic regression  
ด้วย gradient descent

$x_1$	$x_2$	y
0	1	1
1	0	2
-1	0	3
0	-1	4

ตารางแสดง dataset

# Calculation Example

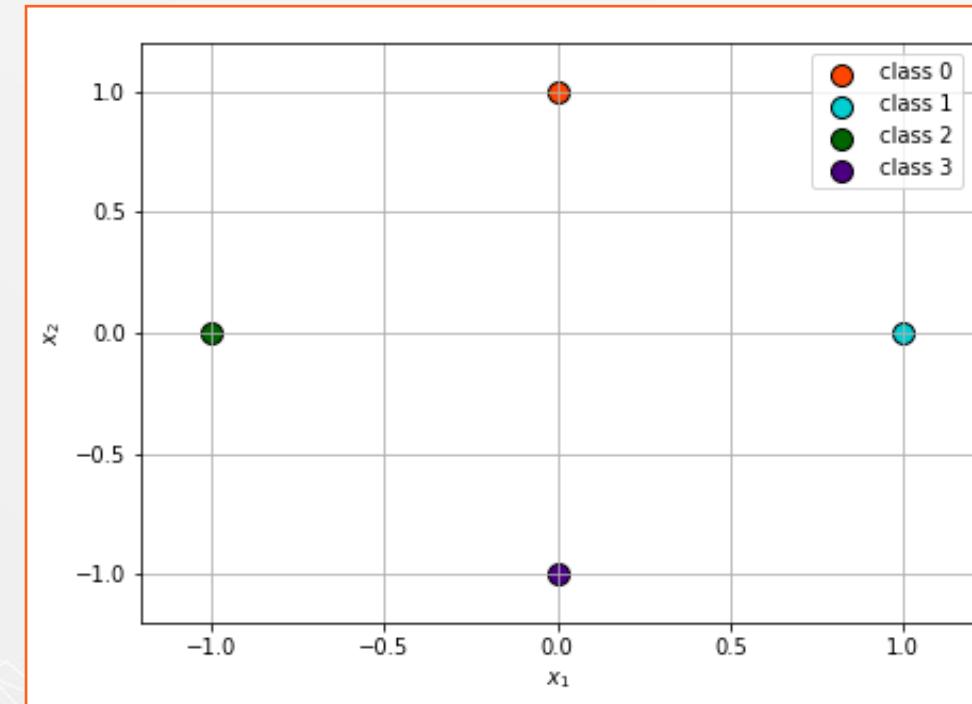
ตัวอย่างการคำนวณ  $W$  สำหรับ logistic regression  
ด้วย gradient descent

$x_1$	$x_2$	1	2	3	4
0	1	1	0	0	0
1	0	0	1	0	0
-1	0	0	0	1	0
0	-1	0	0	0	1

ตารางแสดง dataset

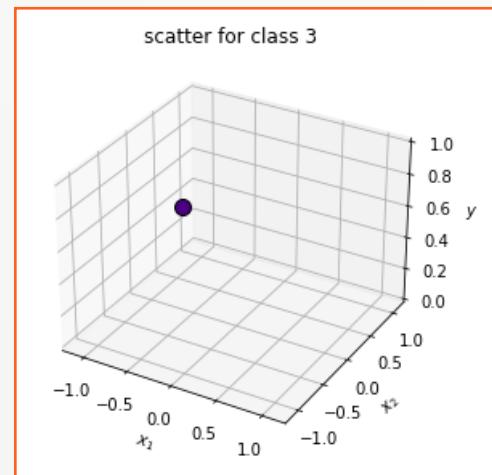
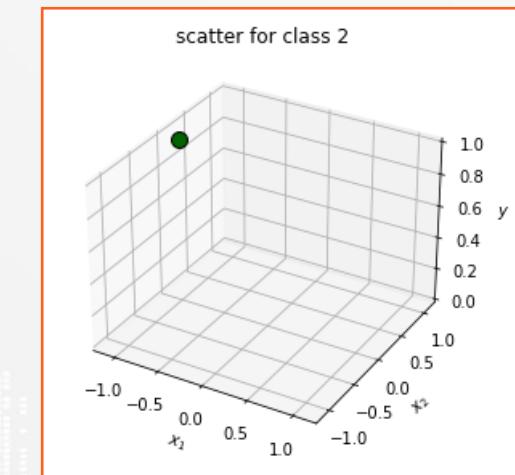
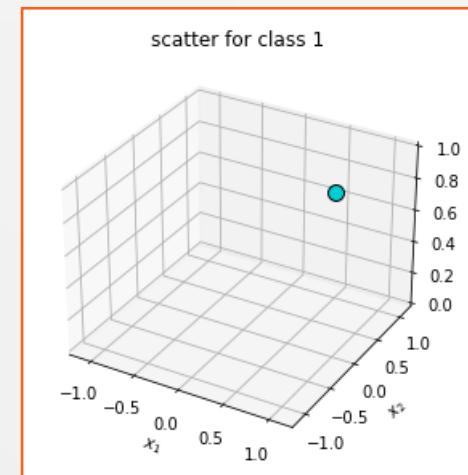
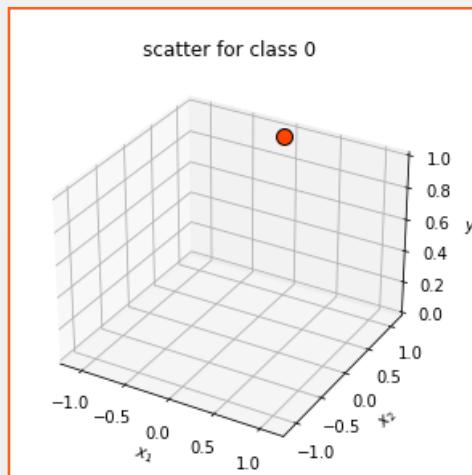
# Calculation Example

- จากข้อมูลใน dataset เราสามารถหาดีกราฟ 2 มิติ ได้ดังนี้



# Calculation Example

- จากข้อมูลใน dataset เราสามารถหาดالةฟิวชัน 3 มิติ ได้ดังนี้



# Calculation Example

## ขั้นตอนในการคำนวณ gradient descent

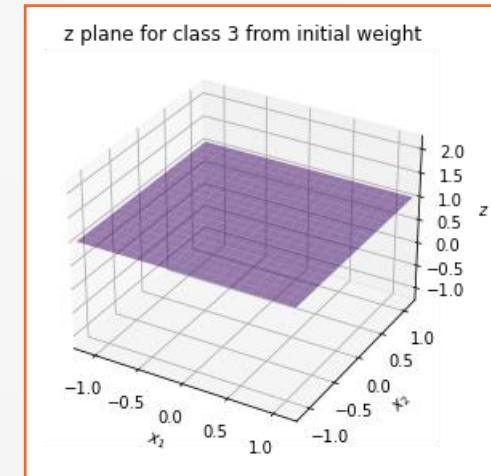
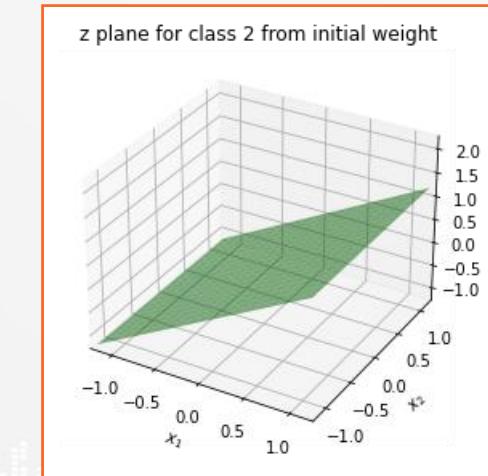
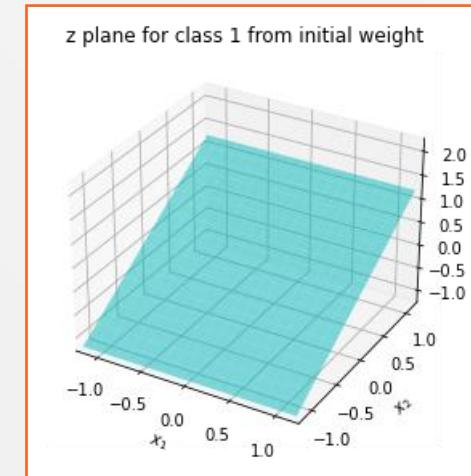
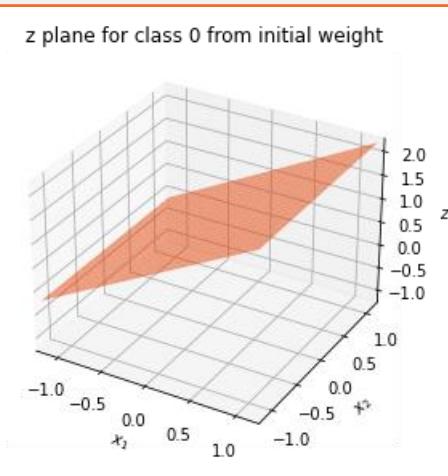
1. สุ่ม  $W$  เริ่มต้น
2. กำหนดจำนวนรอบที่จะ update  $W$  (epoch)
3. กำหนดค่า  $\alpha$
4. For loop เพื่อ update  $W$ 
  - a) คำนวณ  $Z$
  - b) คำนวณ  $\hat{Y}$
  - c) คำนวณ  $W$

# Calculation Example

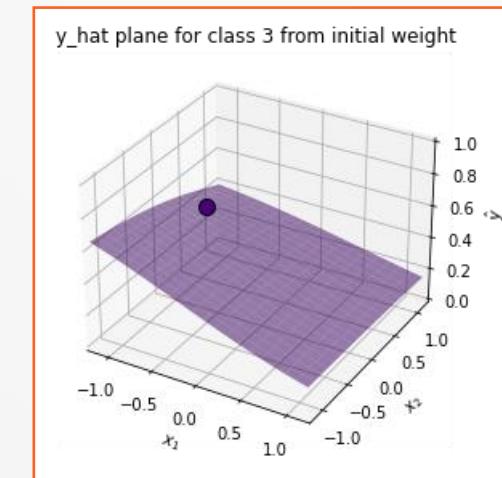
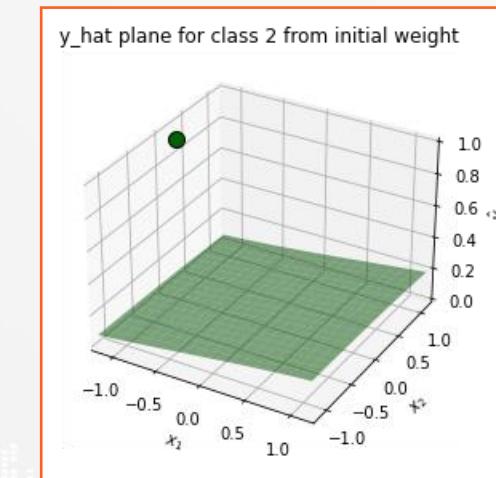
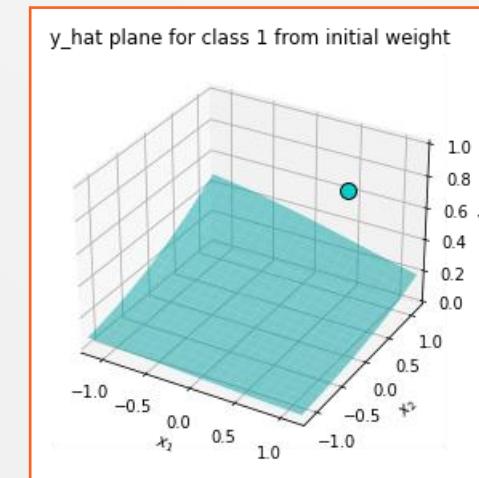
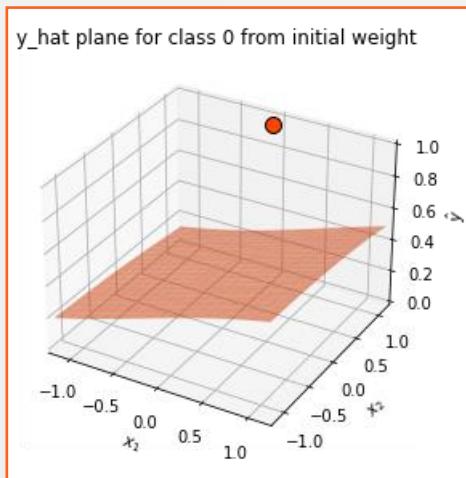
## 1. สุ่ม $W$ เริ่มต้น

$$\text{สมมติให้สุ่ม } W = \begin{bmatrix} w_{0,0} & w_{0,1} & w_{0,2} & w_{0,3} \\ w_{1,0} & w_{1,1} & w_{1,2} & w_{1,3} \\ w_{2,0} & w_{2,1} & w_{2,2} & w_{2,3} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

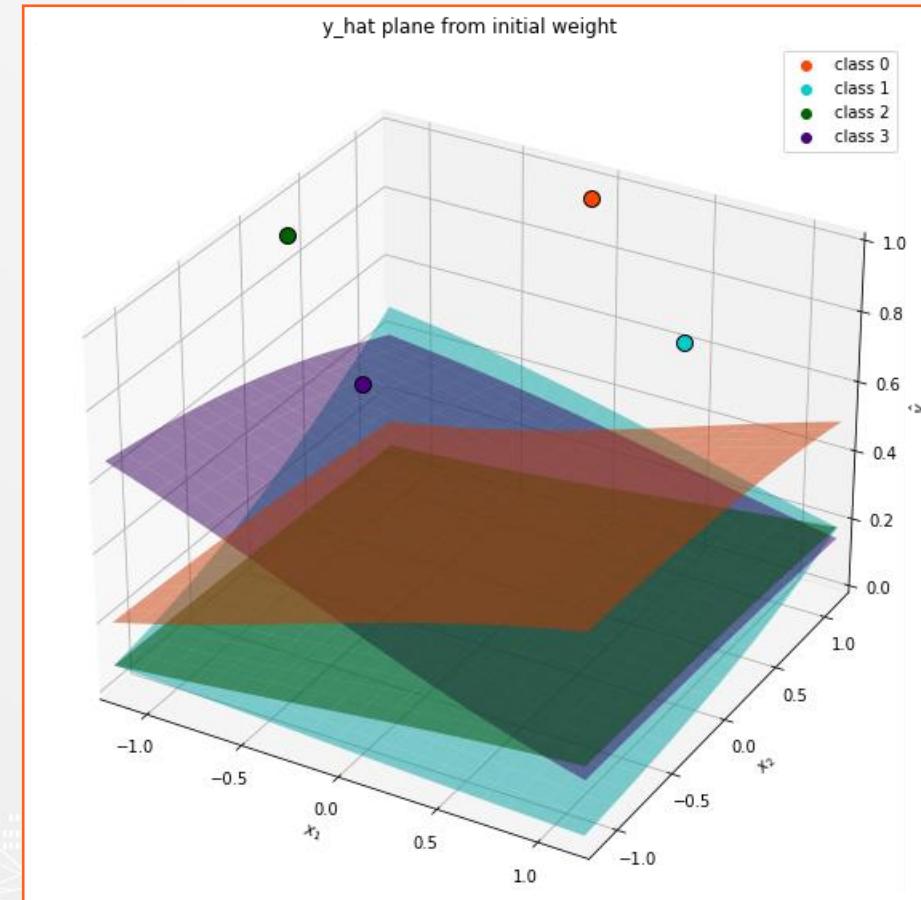
# Calculation Example



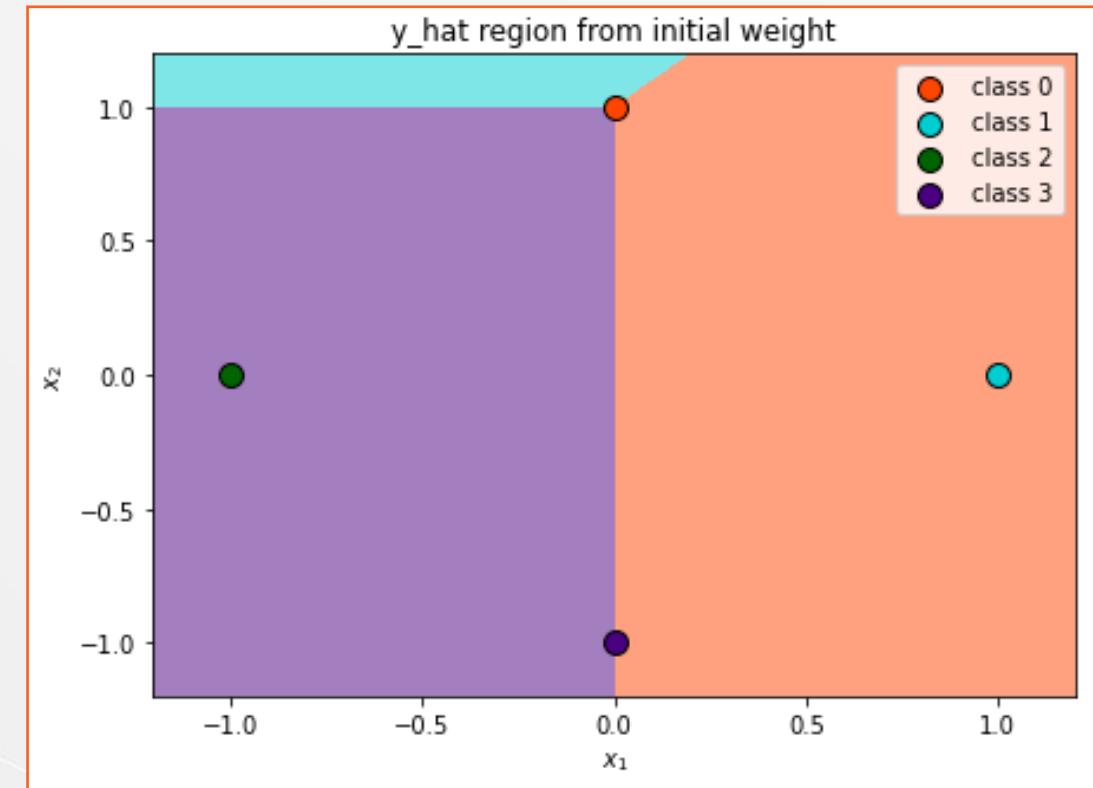
# Calculation Example



# Calculation Example



# Calculation Example



# Calculation Example

2. กำหนดจำนวนรอบที่จะ update  $W$  (epoch)

กำหนดให้ epoch = 3

# Calculation Example

3. กำหนดค่า  $\alpha$

กำหนดให้  $\alpha = 2.5$

# Calculation Example

## 4. Update $W$

$$W = W + \frac{\alpha}{n} X_b^T (Y - \hat{Y})$$

โดย ◆  $\alpha$  คือ ค่าที่ใช้ควบคุม step size ของ  $W$

# Calculation Example

จากข้อมูลใน dataset เราสามารถเขียน  $X, Y$  และ  $X_b$  ได้ดังต่อไปนี้

$$X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \\ -1 & 0 \\ 0 & -1 \end{bmatrix}, \quad Y = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \text{ และ } X_b = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \\ 1 & -1 & 0 \\ 1 & 0 & -1 \end{bmatrix}$$

# Calculation Example

## 4. Update $W$ : คำนวณ $Z$

epoch = 1

$$Z = X_b W = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \\ 1 & -1 & 0 \\ 1 & 0 & -1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 & 1 \\ 2 & 0 & 1 & 1 \\ 0 & 0 & -1 & 1 \\ 1 & -1 & 0 & 1 \end{bmatrix}$$

# Calculation Example

## 4. Update $W$ : คำนวณ $\hat{Y}$

epoch = 1

$$\hat{Y} = \begin{bmatrix} \frac{e^{z_{1,0}}}{\sum_{c=0}^3 e^{z_{1,c}}} & \frac{e^{z_{1,1}}}{\sum_{c=0}^3 e^{z_{1,c}}} & \frac{e^{z_{1,2}}}{\sum_{c=0}^3 e^{z_{1,c}}} & \frac{e^{z_{1,3}}}{\sum_{c=0}^3 e^{z_{1,c}}} \\ \frac{e^{z_{2,0}}}{\sum_{c=0}^3 e^{z_{2,c}}} & \frac{e^{z_{2,1}}}{\sum_{c=0}^3 e^{z_{2,c}}} & \frac{e^{z_{2,2}}}{\sum_{c=0}^3 e^{z_{2,c}}} & \frac{e^{z_{2,3}}}{\sum_{c=0}^3 e^{z_{2,c}}} \\ \frac{e^{z_{3,0}}}{\sum_{c=0}^3 e^{z_{3,c}}} & \frac{e^{z_{3,1}}}{\sum_{c=0}^3 e^{z_{3,c}}} & \frac{e^{z_{3,2}}}{\sum_{c=0}^3 e^{z_{3,c}}} & \frac{e^{z_{3,3}}}{\sum_{c=0}^3 e^{z_{3,c}}} \\ \frac{e^{z_{4,0}}}{\sum_{c=0}^3 e^{z_{4,c}}} & \frac{e^{z_{4,1}}}{\sum_{c=0}^3 e^{z_{4,c}}} & \frac{e^{z_{4,2}}}{\sum_{c=0}^3 e^{z_{4,c}}} & \frac{e^{z_{4,3}}}{\sum_{c=0}^3 e^{z_{4,c}}} \end{bmatrix}$$

# Calculation Example

## 4. Update $W$ : คำนวณ $\hat{Y}$

epoch = 1

$$\hat{Y} = \begin{bmatrix} 0.3 & 0.3 & 0.11 & 0.3 \\ 0.53 & 0.07 & 0.2 & 0.2 \\ 0.2 & 0.2 & 0.07 & 0.53 \\ 0.4 & 0.05 & 0.15 & 0.4 \end{bmatrix}$$

# Calculation Example

## 4. Update $W$ : คำนวณ $W$

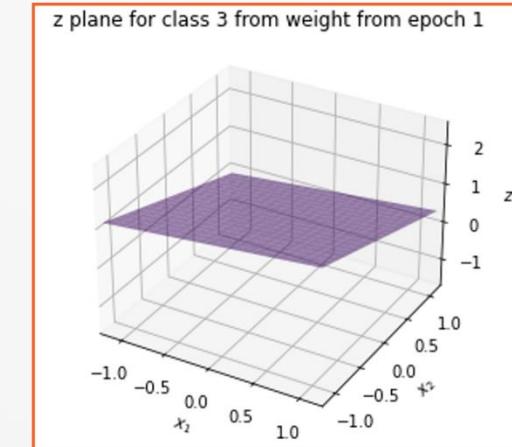
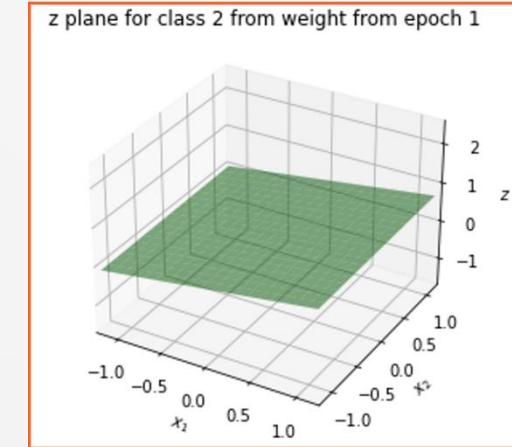
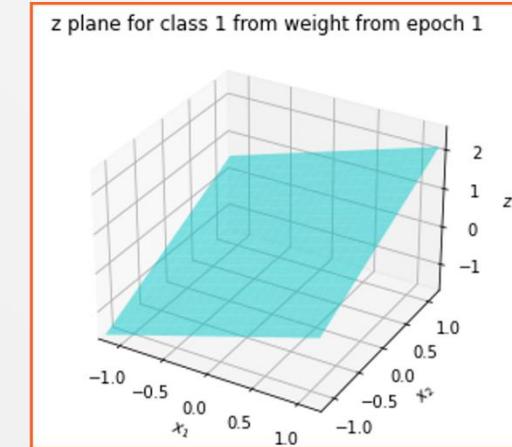
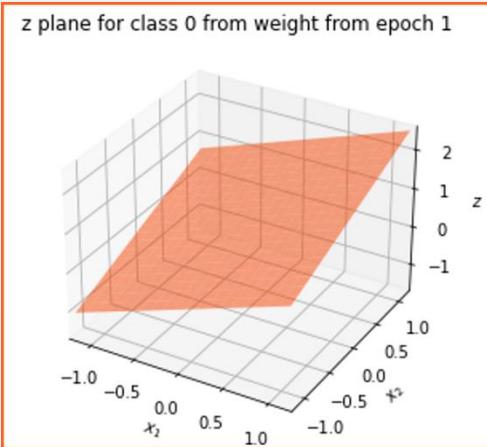
epoch = 1

$$W = W + \frac{\alpha}{n} X_b^T (Y - \hat{Y})$$

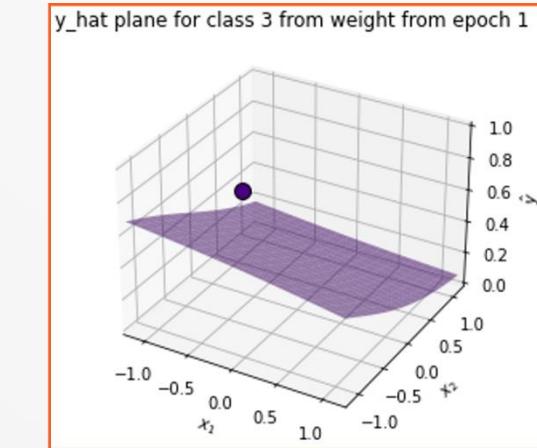
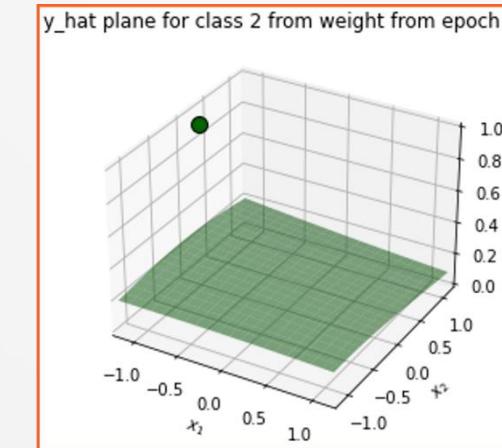
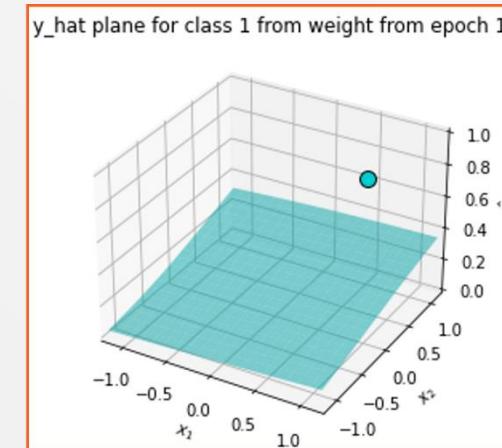
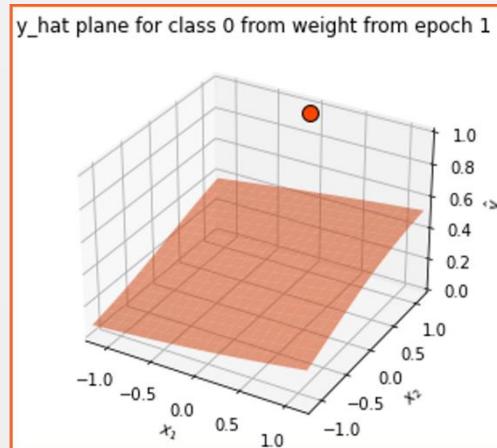
$$= \begin{bmatrix} 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} + \frac{2.5}{4} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & 1 & -1 & 0 \\ 1 & 0 & 0 & -1 \end{bmatrix} \left( \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} - \begin{bmatrix} 0.3 & 0.3 & 0.11 & 0.3 \\ 0.53 & 0.07 & 0.2 & 0.2 \\ 0.2 & 0.2 & 0.07 & 0.53 \\ 0.4 & 0.05 & 0.15 & 0.4 \end{bmatrix} \right)$$

$$= \begin{bmatrix} 0.73 & 0.24 & 0.3 & 0.73 \\ 0.79 & 0.7 & 0.3 & 0.21 \\ 0.69 & 0.85 & 0.23 & -0.56 \end{bmatrix}$$

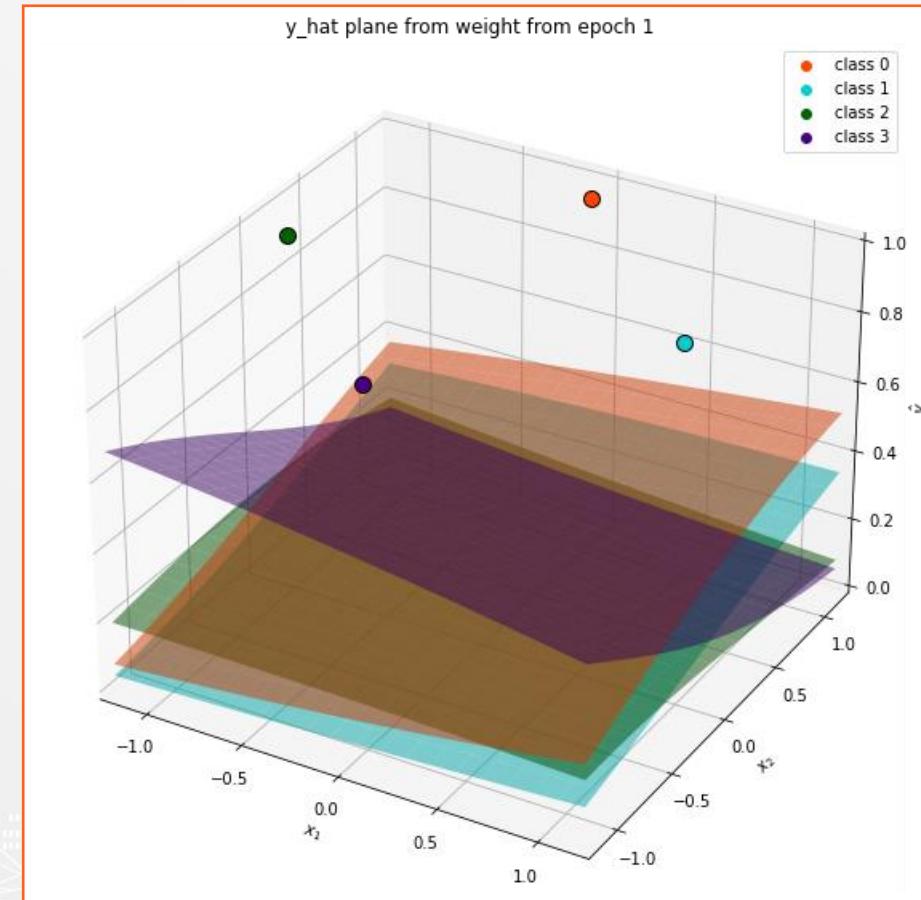
# Calculation Example



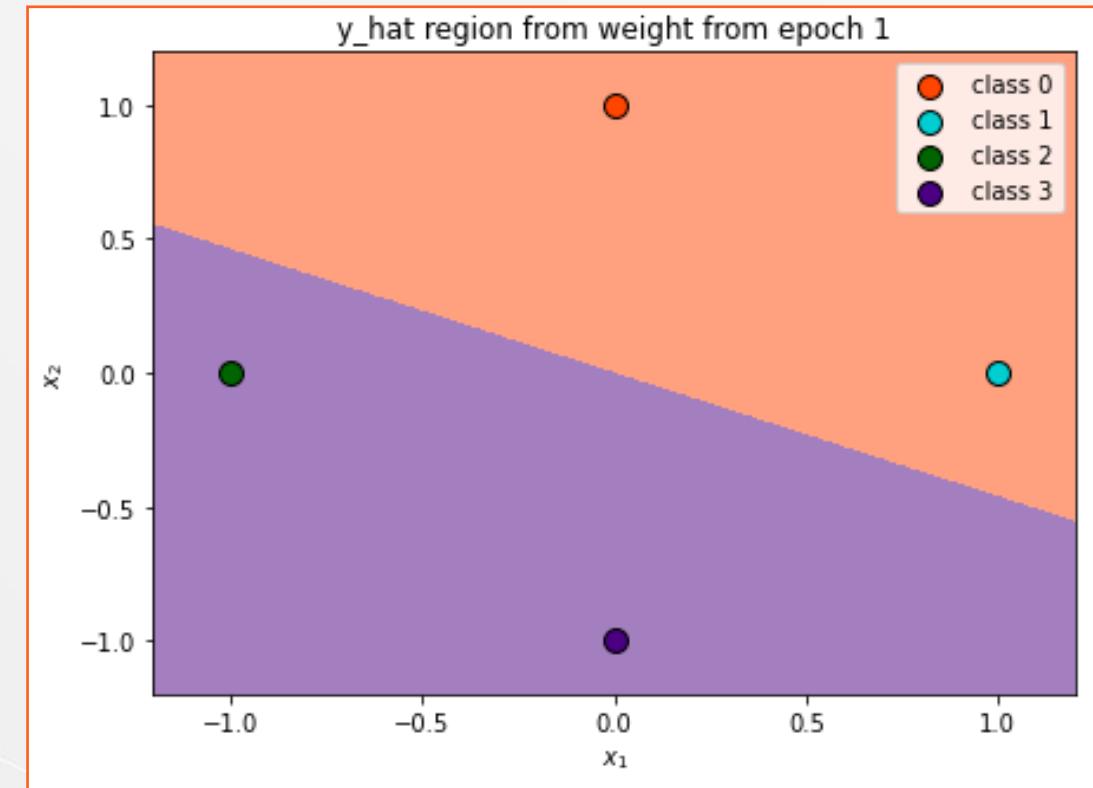
# Calculation Example



# Calculation Example



# Calculation Example



# Calculation Example

## 4. Update $W$ : คำนวณ $Z$

epoch = 2

$$\begin{aligned} Z = X_b W &= \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \\ 1 & -1 & 0 \\ 1 & 0 & -1 \end{bmatrix} \begin{bmatrix} 0.73 & 0.24 & 0.3 & 0.73 \\ 0.79 & 0.7 & 0.3 & 0.21 \\ 0.69 & 0.85 & 0.23 & -0.56 \end{bmatrix} \\ &= \begin{bmatrix} 1.42 & 1.09 & 0.32 & 0.17 \\ 1.52 & 0.94 & 0.59 & 0.94 \\ -0.056 & -0.47 & -5.33 \times 10^{-4} & 0.52 \\ 0.044 & -0.61 & 0.27 & 1.29 \end{bmatrix} \end{aligned}$$

# Calculation Example

## 4. Update $W$ : คำนวณ $\hat{Y}$

epoch = 2

$$\hat{Y} = \begin{bmatrix} \frac{e^{z_{1,0}}}{\sum_{c=0}^3 e^{z_{1,c}}} & \frac{e^{z_{1,1}}}{\sum_{c=0}^3 e^{z_{1,c}}} & \frac{e^{z_{1,2}}}{\sum_{c=0}^3 e^{z_{1,c}}} & \frac{e^{z_{1,3}}}{\sum_{c=0}^3 e^{z_{1,c}}} \\ \frac{e^{z_{2,0}}}{\sum_{c=0}^3 e^{z_{2,c}}} & \frac{e^{z_{2,1}}}{\sum_{c=0}^3 e^{z_{2,c}}} & \frac{e^{z_{2,2}}}{\sum_{c=0}^3 e^{z_{2,c}}} & \frac{e^{z_{2,3}}}{\sum_{c=0}^3 e^{z_{2,c}}} \\ \frac{e^{z_{3,0}}}{\sum_{c=0}^3 e^{z_{3,c}}} & \frac{e^{z_{3,1}}}{\sum_{c=0}^3 e^{z_{3,c}}} & \frac{e^{z_{3,2}}}{\sum_{c=0}^3 e^{z_{3,c}}} & \frac{e^{z_{3,3}}}{\sum_{c=0}^3 e^{z_{3,c}}} \\ \frac{e^{z_{4,0}}}{\sum_{c=0}^3 e^{z_{4,c}}} & \frac{e^{z_{4,1}}}{\sum_{c=0}^3 e^{z_{4,c}}} & \frac{e^{z_{4,2}}}{\sum_{c=0}^3 e^{z_{4,c}}} & \frac{e^{z_{4,3}}}{\sum_{c=0}^3 e^{z_{4,c}}} \end{bmatrix}$$

# Calculation Example

## 4. Update $W$ : คำนวณ $\hat{Y}$

epoch = 2

$$\hat{Y} = \begin{bmatrix} 0.43 & 0.31 & 0.14 & 0.12 \\ 0.4 & 0.22 & 0.16 & 0.22 \\ 0.22 & 0.15 & 0.23 & 0.4 \\ 0.16 & 0.08 & 0.2 & 0.56 \end{bmatrix}$$

# Calculation Example

## 4. Update $W$ : คำนวณ $W$

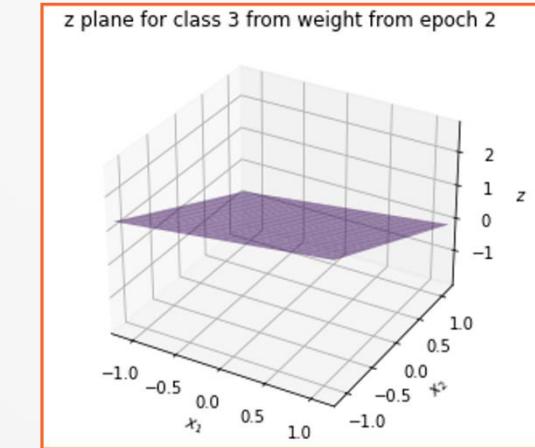
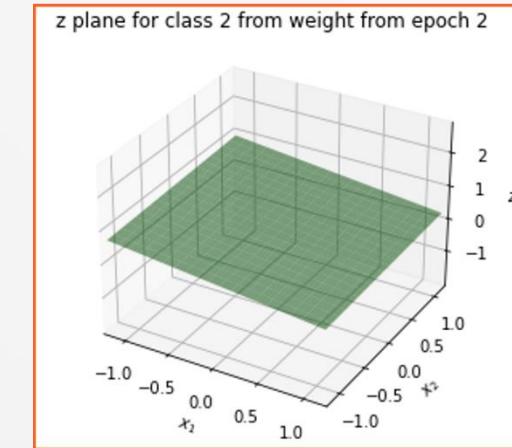
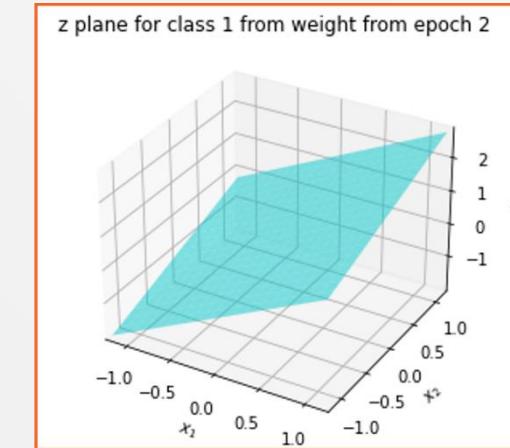
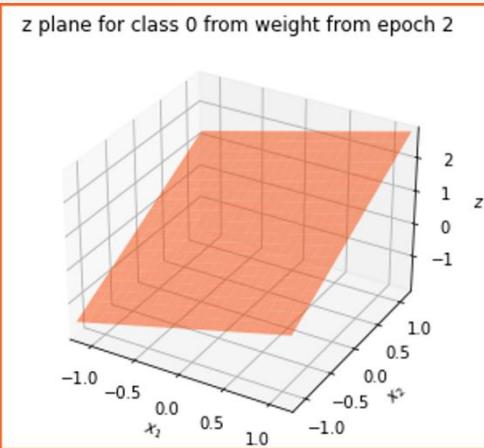
epoch = 2

$$W = W + \frac{\alpha}{n} X_b^T (Y - \hat{Y})$$

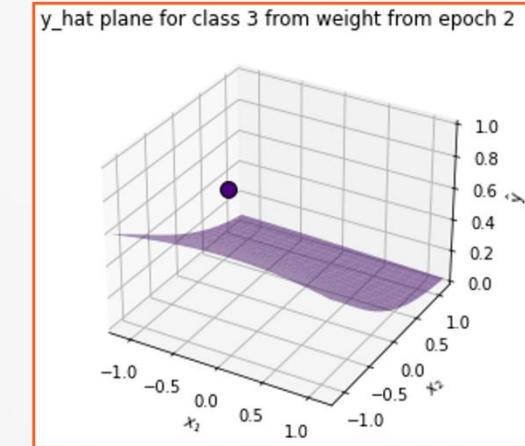
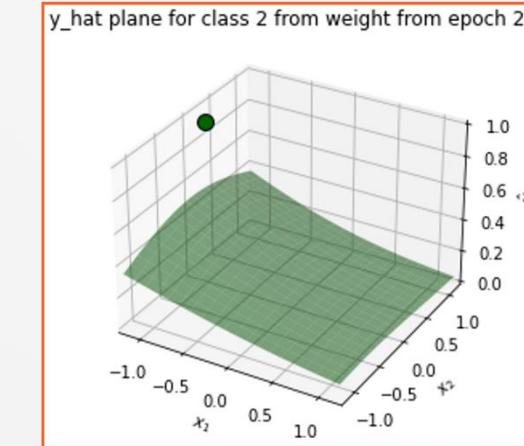
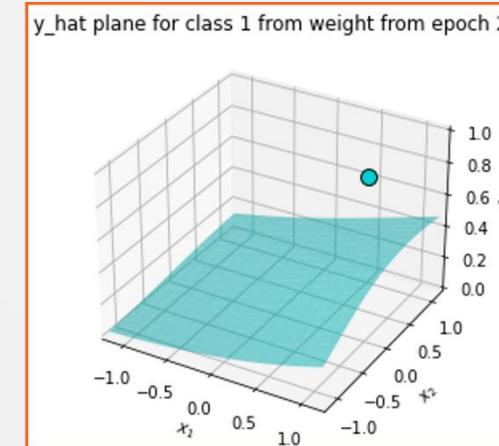
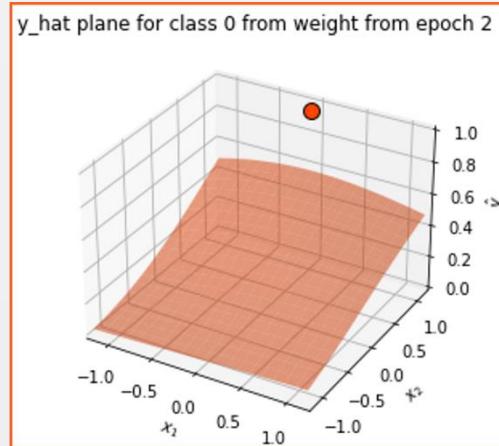
$$= \begin{bmatrix} 0.73 & 0.24 & 0.3 & 0.73 \\ 0.79 & 0.7 & 0.3 & 0.21 \\ 0.69 & 0.85 & 0.23 & -0.56 \end{bmatrix} + \frac{2.5}{4} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & 1 & -1 & 0 \\ 1 & 0 & 0 & -1 \end{bmatrix} \left( \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} - \begin{bmatrix} 0.43 & 0.31 & 0.14 & 0.12 \\ 0.4 & 0.22 & 0.16 & 0.22 \\ 0.22 & 0.15 & 0.23 & 0.4 \\ 0.16 & 0.08 & 0.2 & 0.56 \end{bmatrix} \right)$$

$$= \begin{bmatrix} 0.6 & 0.39 & 0.46 & 0.55 \\ 0.68 & 1.28 & -0.28 & 0.32 \\ 1.15 & 0.71 & 0.06 & -0.91 \end{bmatrix}$$

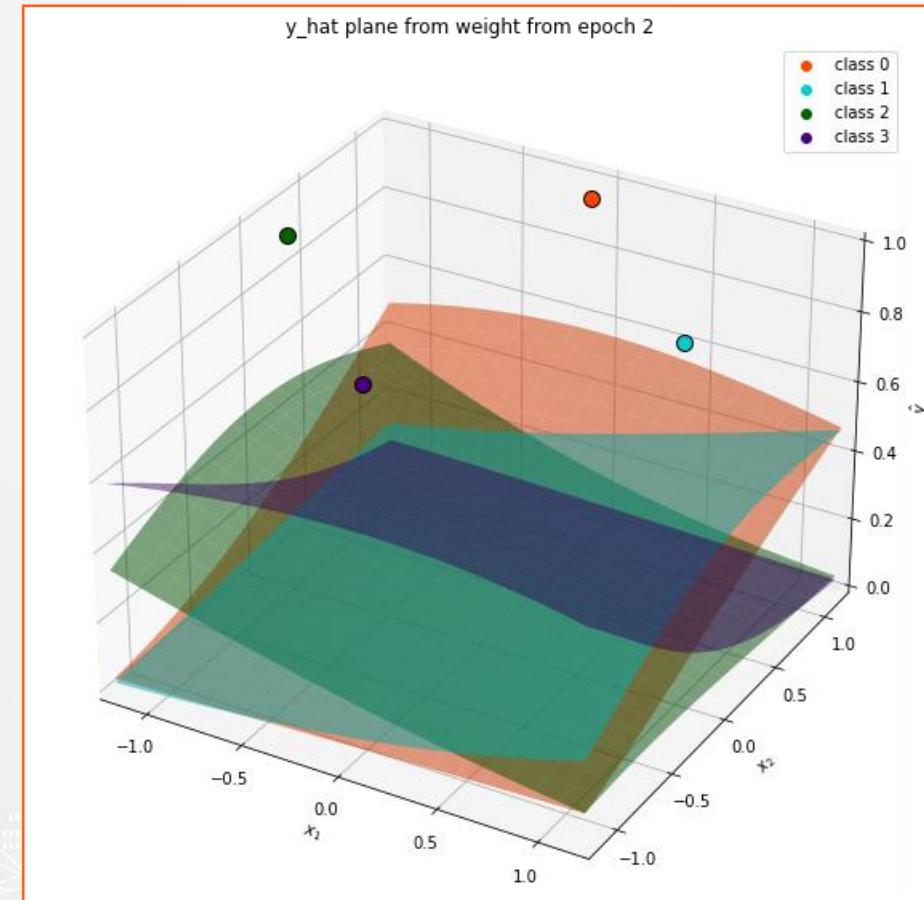
# Calculation Example



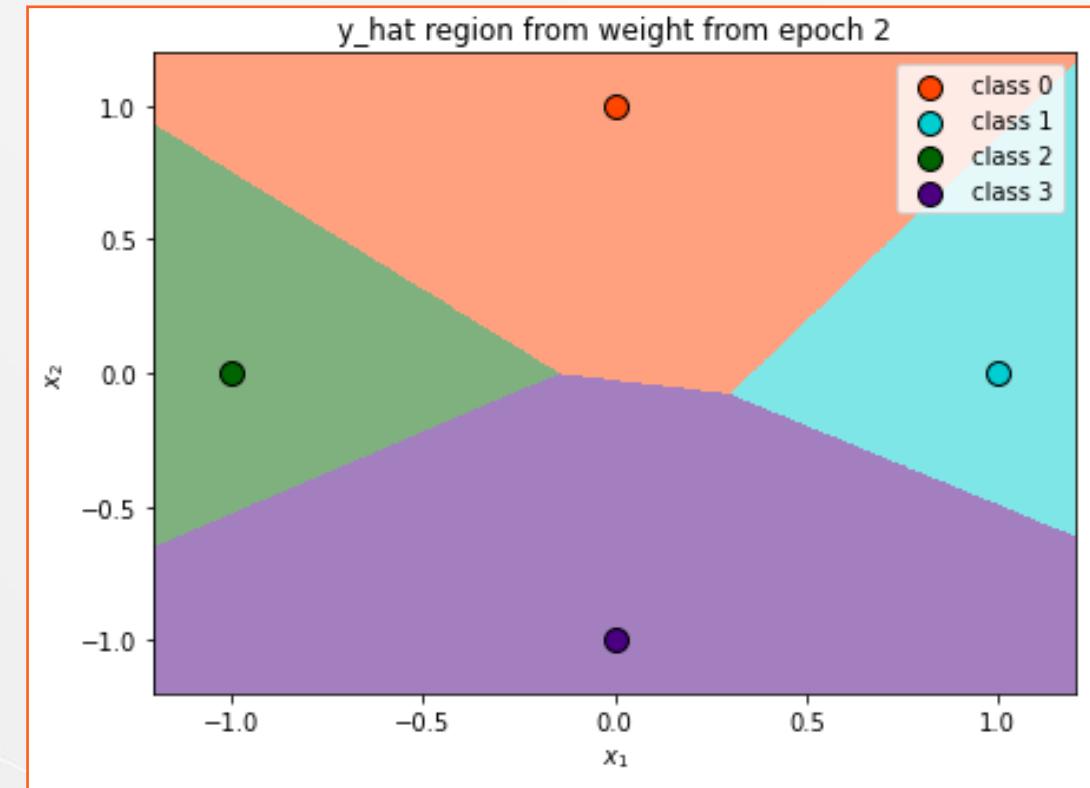
# Calculation Example



# Calculation Example



# Calculation Example



# Calculation Example

## 4. Update $W$ : คำนวณ $Z$

epoch = 3

$$Z = X_b W = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \\ 1 & -1 & 0 \\ 1 & 0 & -1 \end{bmatrix} \begin{bmatrix} 0.6 & 0.39 & 0.46 & 0.55 \\ 0.68 & 1.28 & -0.28 & 0.32 \\ 1.15 & 0.71 & 0.06 & -0.91 \end{bmatrix} = \begin{bmatrix} 1.75 & 1.1 & 0.52 & -0.37 \\ 1.28 & 1.67 & 0.18 & 0.87 \\ -0.08 & -0.89 & 0.74 & 0.23 \\ -0.54 & -0.32 & 0.4 & 1.46 \end{bmatrix}$$

# Calculation Example

## 4. Update $W$ : คำนวณ $\hat{Y}$

epoch = 3

$$\hat{Y} = \begin{bmatrix} \frac{e^{z_{1,0}}}{\sum_{c=0}^3 e^{z_{1,c}}} & \frac{e^{z_{1,1}}}{\sum_{c=0}^3 e^{z_{1,c}}} & \frac{e^{z_{1,2}}}{\sum_{c=0}^3 e^{z_{1,c}}} & \frac{e^{z_{1,3}}}{\sum_{c=0}^3 e^{z_{1,c}}} \\ \frac{e^{z_{2,0}}}{\sum_{c=0}^3 e^{z_{2,c}}} & \frac{e^{z_{2,1}}}{\sum_{c=0}^3 e^{z_{2,c}}} & \frac{e^{z_{2,2}}}{\sum_{c=0}^3 e^{z_{2,c}}} & \frac{e^{z_{2,3}}}{\sum_{c=0}^3 e^{z_{2,c}}} \\ \frac{e^{z_{3,0}}}{\sum_{c=0}^3 e^{z_{3,c}}} & \frac{e^{z_{3,1}}}{\sum_{c=0}^3 e^{z_{3,c}}} & \frac{e^{z_{3,2}}}{\sum_{c=0}^3 e^{z_{3,c}}} & \frac{e^{z_{3,3}}}{\sum_{c=0}^3 e^{z_{3,c}}} \\ \frac{e^{z_{4,0}}}{\sum_{c=0}^3 e^{z_{4,c}}} & \frac{e^{z_{4,1}}}{\sum_{c=0}^3 e^{z_{4,c}}} & \frac{e^{z_{4,2}}}{\sum_{c=0}^3 e^{z_{4,c}}} & \frac{e^{z_{4,3}}}{\sum_{c=0}^3 e^{z_{4,c}}} \end{bmatrix}$$

# Calculation Example

## 4. Update $W$ : คำนวณ $\hat{Y}$

epoch = 3

$$\hat{Y} = \begin{bmatrix} 0.52 & 0.27 & 0.15 & 0.06 \\ 0.29 & 0.42 & 0.1 & 0.19 \\ 0.2 & 0.09 & 0.45 & 0.27 \\ 0.08 & 0.1 & 0.21 & 0.61 \end{bmatrix}$$

# Calculation Example

## 4. Update $W$ : คำนวณ $W$

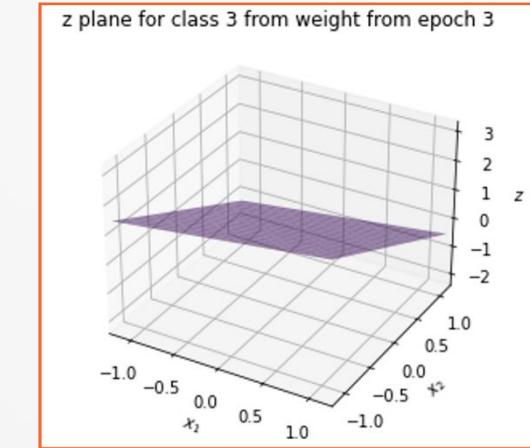
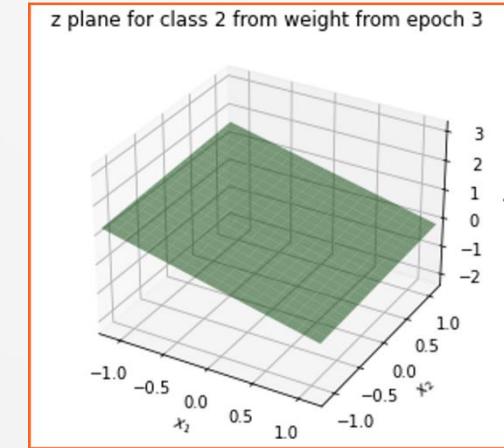
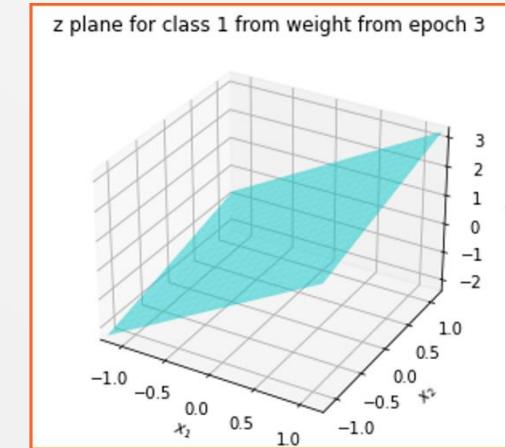
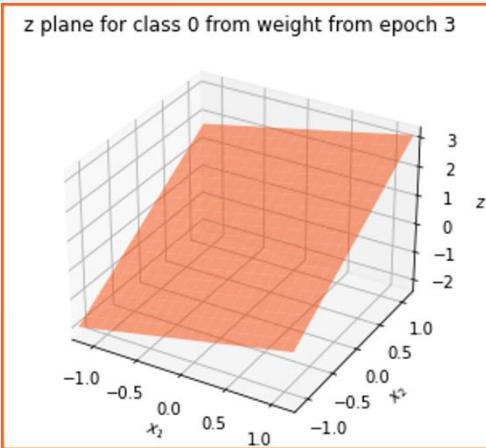
epoch = 3

$$W = W + \frac{\alpha}{n} X_b^T (Y - \hat{Y})$$

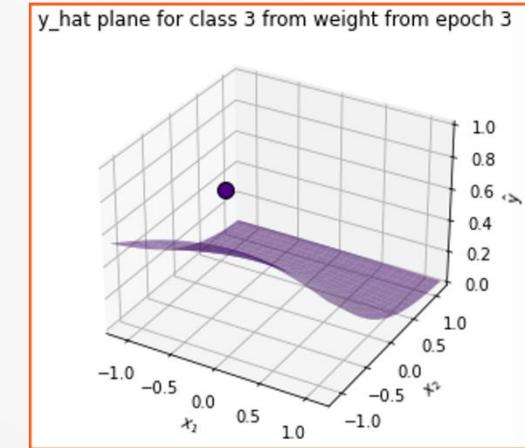
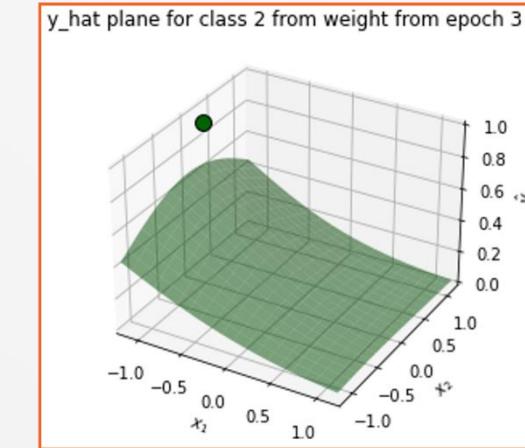
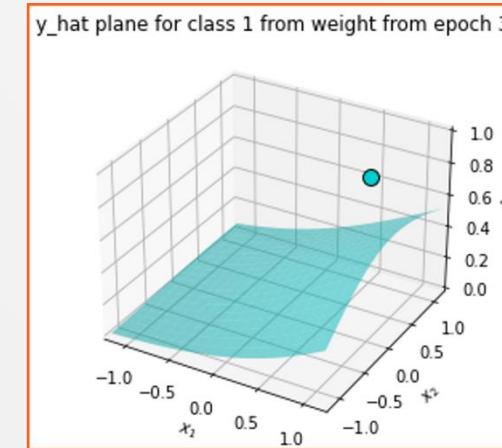
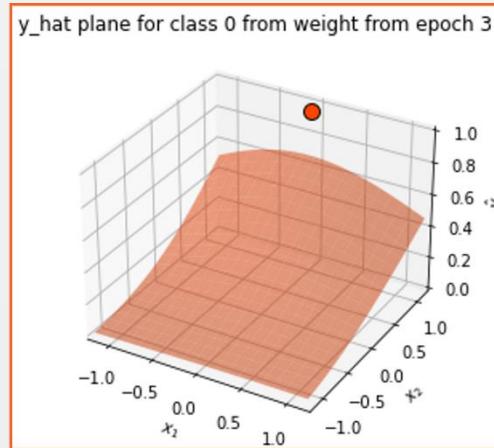
$$= \begin{bmatrix} 0.6 & 0.39 & 0.46 & 0.55 \\ 0.68 & 1.28 & -0.28 & 0.32 \\ 1.15 & 0.71 & 0.06 & -0.91 \end{bmatrix} + \frac{2.5}{4} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & 1 & -1 & 0 \\ 1 & 0 & 0 & -1 \end{bmatrix} \left( \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} - \begin{bmatrix} 0.52 & 0.27 & 0.15 & 0.06 \\ 0.29 & 0.42 & 0.1 & 0.19 \\ 0.2 & 0.09 & 0.45 & 0.27 \\ 0.08 & 0.1 & 0.21 & 0.61 \end{bmatrix} \right)$$

$$= \begin{bmatrix} 0.55 & 0.46 & 0.52 & 0.47 \\ 0.62 & 1.69 & -0.68 & 0.37 \\ 1.5 & 0.6 & 0.09 & -1.2 \end{bmatrix}$$

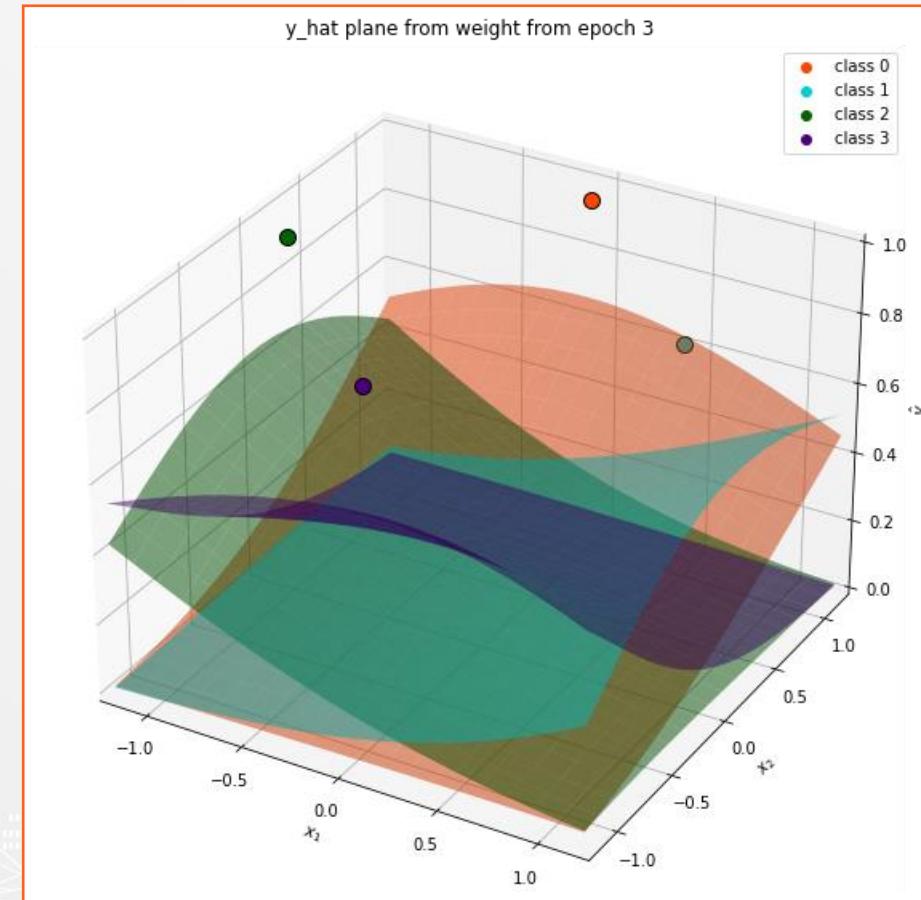
# Calculation Example



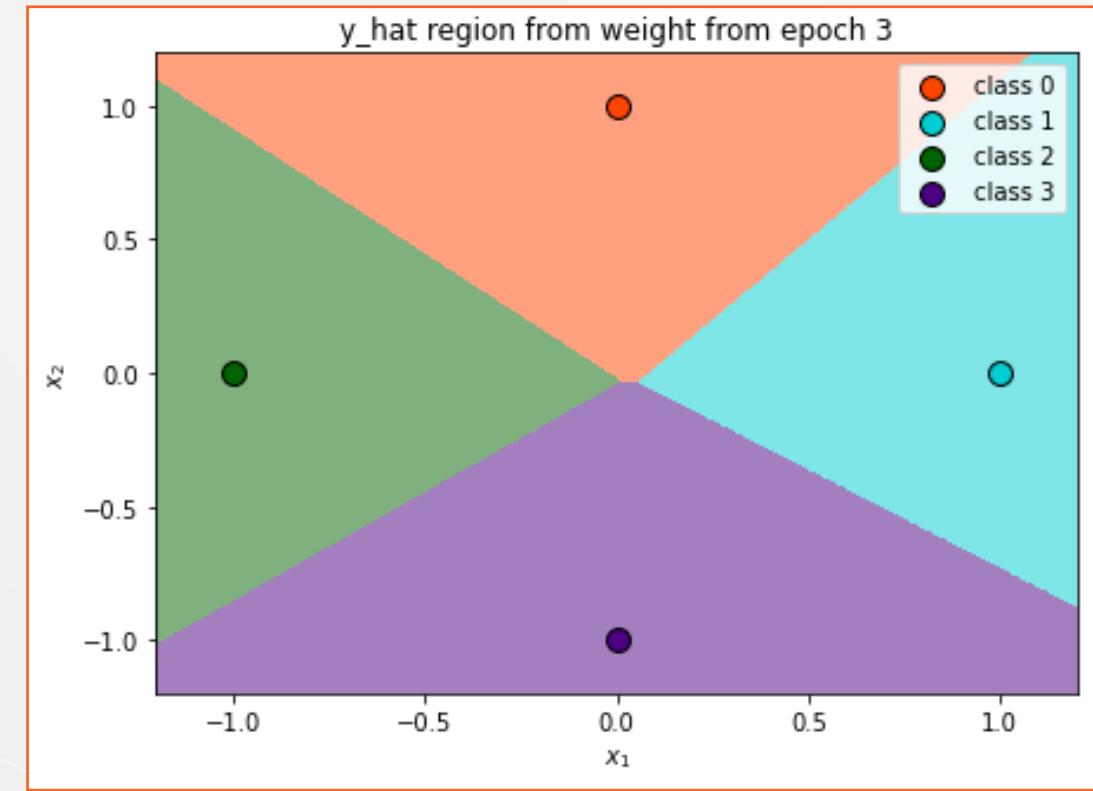
# Calculation Example



# Calculation Example

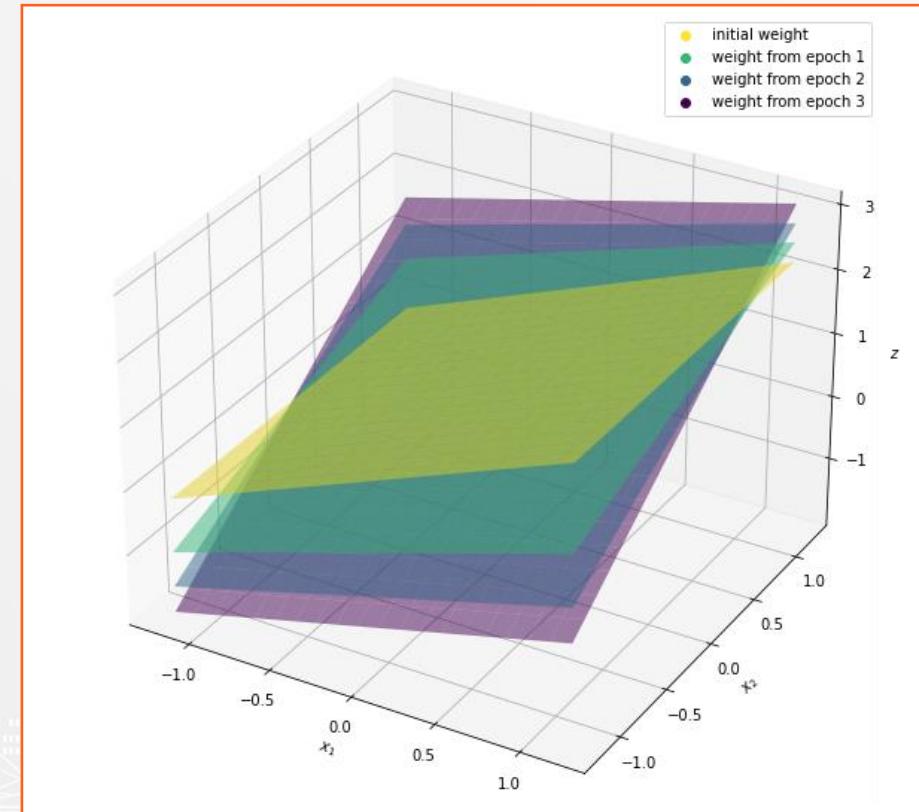


# Calculation Example



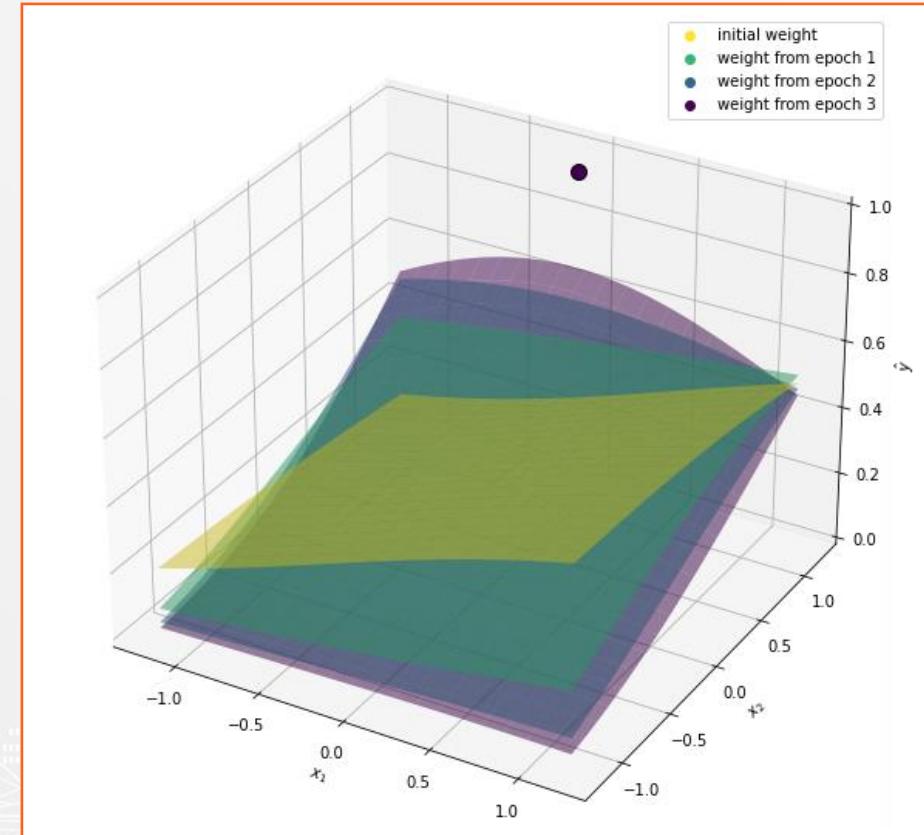
# Calculation Example

z plane ឧបនគរណៈ epoch (class 0)

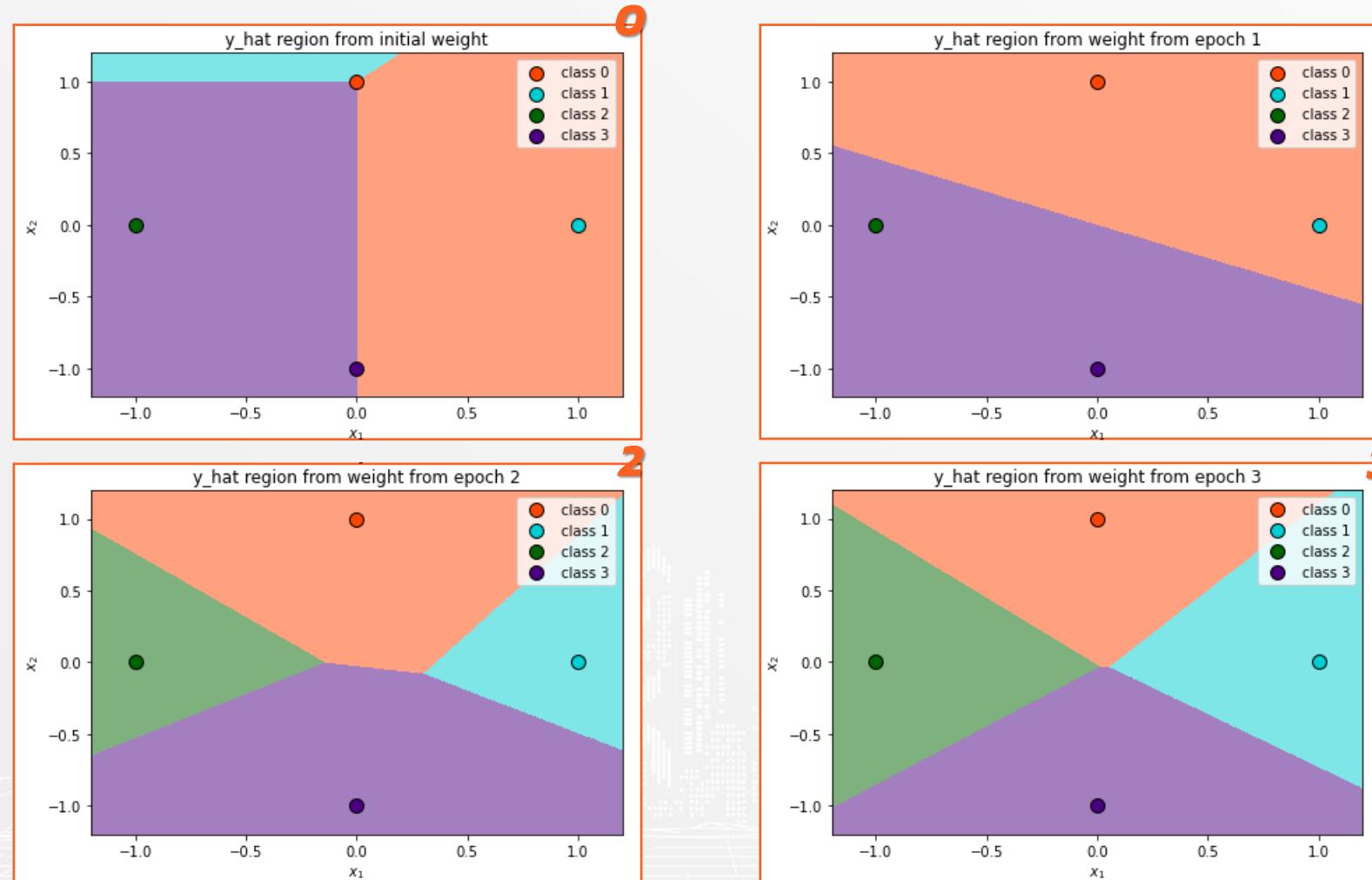


# Calculation Example

$\hat{y}$  plane នៃវត្ថុទាំង ៣ epoch (class 0)



# Calculation Example



# Calculation Example



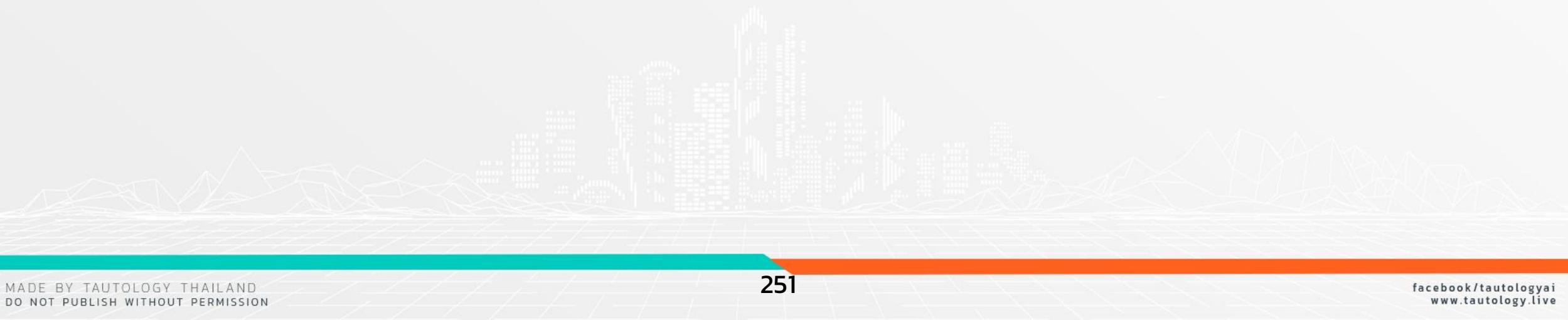
Exercise of Logistic Regression (multi-class)



Open File  
**Exercise\_LoR\_(multi-class).pdf**

# How to Create Model (Math)

- Gradient Descent (Logistic Regression)**
- Calculation Example**



# Model Creation

**Assumption**



**Real Face of the Model**



**Cost Function and Cost Landscape**



**How to Create Model (Math)**



**How to Create Model (Code)**



# How to Create Model (Code)

ตัวอย่าง code สำหรับคำนวณ  $W$

$x_1$	$x_2$	$y$
0	1	1
1	0	2
-1	0	3
0	-1	4

ตารางแสดง dataset

# How to Create Model (Code)

- Code สำหรับสร้าง model จากข้อมูลของเราโดยที่

$$X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \\ -1 & 0 \\ 0 & -1 \end{bmatrix} \text{ และ } y = \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \end{bmatrix}$$

```
1 clf = LogisticRegression(penalty='none')
2 clf.fit(x, y)
```

```
LogisticRegression(penalty='none')
```

# How to Create Model (Code)

- ค่า  $w_0$  ถูกเก็บไว้ใน attribute ชื่อ intercept\_

```
1 clf.intercept_
```

```
array([-2.62306367e-08,  2.95335134e-08,  2.54560370e-08, -2.87561552e-08])
```

# How to Create Model (Code)

- ค่า  $w_1, \dots, w_p$  จะเก็บไว้ใน attribute ชื่อ `coef_`

```
1 clf.coef_
```

```
array([[-1.44123650e-09,  1.04728380e+01],  
      [ 1.04728380e+01, -8.95380278e-10],  
      [-1.04728380e+01, -8.95382812e-10],  
      [-1.44123698e-09, -1.04728380e+01]])
```

# How to Create Model (Code)

ดังนั้น เราจะสามารถเขียนเป็น model ของ logistic regression สำหรับข้อมูลชุดนี้ได้ดังนี้

$$z_0 = 0 + 0x_1 + 10.47x_2$$

$$z_1 = 0 + 10.47x_1 + 0x_2$$

$$z_2 = 0 - 10.47x_1 + 0x_2$$

$$z_3 = 0 + 0x_1 - 10.47x_2$$

$$\hat{y} = \frac{e^{z_m}}{\sum_{c=0}^3 e^{z_c}}$$

# How to Create Model (Code)



Code for this section



Open File  
**Model Creation (multi-class).ipynb**

# How to Create Model (Code)



MATH

$$W = \begin{bmatrix} w_{0,0} & w_{0,1} & \cdots & w_{0,k-1} \\ w_{1,0} & w_{1,1} & \cdots & w_{1,k-1} \\ \vdots & \vdots & \ddots & \vdots \\ w_{p,0} & w_{p,1} & \cdots & w_{p,k-1} \end{bmatrix}$$

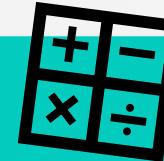


CODE

intercept\_ =  $[w_{0,0} \quad w_{0,1} \quad \cdots \quad w_{0,k-1}]$

coef\_ =  $\begin{bmatrix} w_{1,0} & w_{2,0} & \cdots & w_{p,0} \\ w_{1,1} & w_{2,1} & \cdots & w_{p,1} \\ \vdots & \vdots & \ddots & \vdots \\ w_{1,k-1} & w_{2,k-1} & \cdots & w_{p,k-1} \end{bmatrix}$

# How to Create Model (Code)



MATH

1. สูม  $W$  เริ่มต้น
2. กำหนดจำนวนรอบที่จะ update  $W$  (epoch)
3. กำหนดค่า  $\alpha$
4. for i in range(epoch):  
    update  $W$



CODE

```
1 clf = LogisticRegression(penalty='none')
2 clf.fit(X, y)

LogisticRegression(penalty='none')

1 clf.intercept_
array([-2.62306367e-08,  2.95335134e-08,  2.54560370e-08, -2.87561552e-08])

1 clf.coef_
array([[-1.44123650e-09,  1.04728380e+01],
       [ 1.04728380e+01, -8.95380278e-10],
       [-1.04728380e+01, -8.95382812e-10],
       [-1.44123698e-09, -1.04728380e+01]])
```

# Model Creation

**Assumption**



**Real Face of the Model**



**Cost Function and Cost Landscape**



**How to Create Model (Math)**



**How to Create Model (Code)**



# Logistic Regression (Multi-Class)



# Prediction

# Prediction

**Logistic regression** คือ โมเดลคณิตศาสตร์ที่จำแนกประเภทของข้อมูล ด้วยการสร้างสมการเชิงเส้น (linear function)

$$z_m = w_{0,m} + w_{1,m}x_1 + w_{2,m}x_2 + \cdots + w_{p,m}x_p$$

$$\hat{y}_m = \frac{e^{z_m}}{\sum_{c=0}^{k-1} e^{z_c}}$$

- โดยที่
- ◆  $\hat{y}_m$  คือ ความน่าจะเป็นที่จะเป็น class ที่  $m$  (predicted target)
  - ◆  $z_m$  คือ สมการเชิงเส้น (linear function) ของ class ที่  $m$
  - ◆  $x_1, x_2, x_3, \dots, x_p$  คือ ตัวแปรต้น (feature)
  - ◆  $w_{0,m}, w_{1,m}, w_{2,m}, \dots, w_{p,m}$  คือ สัมประสิทธิ์ที่ส่งจาก feature ไป class ที่  $m$  (coefficient)
  - ◆  $m \in 0, 1, 2, \dots, k - 1$

# Prediction

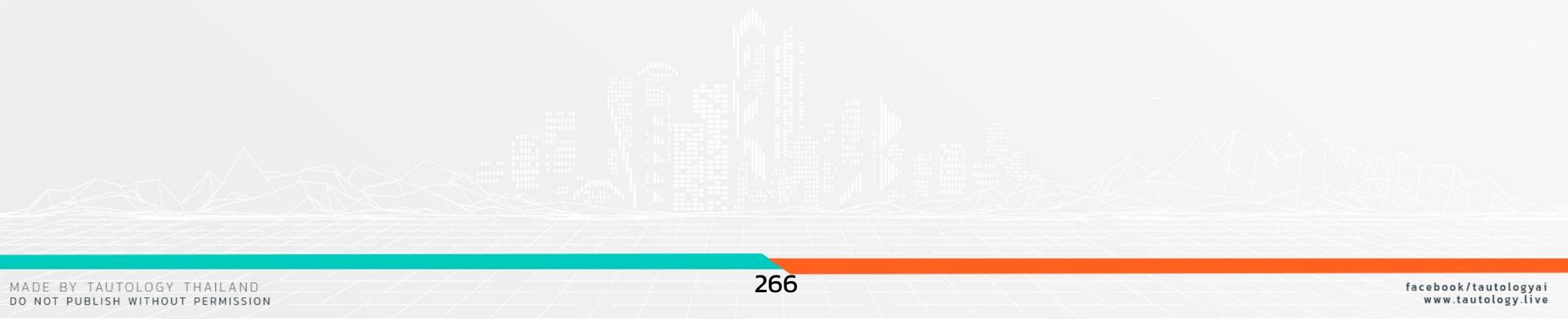
$$W = \begin{bmatrix} w_{0,0} & w_{0,1} & \cdots & w_{0,k-1} \\ w_{1,0} & w_{1,1} & \cdots & w_{1,k-1} \\ \vdots & \vdots & \ddots & \vdots \\ w_{p,0} & w_{p,1} & \cdots & w_{p,k-1} \end{bmatrix}$$

# Prediction

1-Sample

Multi-Sample

Code



# 1-Sample

ตัวอย่างการคำนวณ  $\hat{y}$

$x_1$	$x_2$
0	1.5



$\hat{y}$
?

# 1-Sample

สมมติว่า  $W$  ของปัญหานี้ที่เราหามาได้คือ

$$W = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 10.47 & -10.47 & 0 \\ 10.47 & 0 & 0 & -10.47 \end{bmatrix}$$

# 1-Sample

ซึ่งทำให้สามารถเขียนสมการ  $\hat{y}$  ได้ดังต่อไปนี้

- ♦  $z_0 = 0 + 0 \cdot x_1 + 10.47x_2 = 10.47 \times 1.5 = 15.705$
- $z_1 = 0 + 10.47x_1 + 0 \cdot x_2 = 10.47 \times 0 = 0$
- $z_2 = 0 - 10.47x_1 + 0 \cdot x_2 = 10.47 \times 0 = 0$
- $z_3 = 0 + 0 \cdot x_1 - 10.47x_2 = 10.47 \times 1.5 = -15.705$

# 1-Sample

ซึ่งทำให้สามารถเขียนสมการ  $\hat{y}$  ได้ดังต่อไปนี้

$$\star \quad \hat{y}_0 = \frac{e^{z_1}}{e^{z_1} + e^{z_2} + e^{z_3} + e^{z_4}} = \frac{e^{15.705}}{e^{15.705} + e^0 + e^0 + e^{-15.705}} \approx 1 \quad \rightarrow \quad 0$$

$$\hat{y}_1 = \frac{e^{z_2}}{e^{z_1} + e^{z_2} + e^{z_3} + e^{z_4}} = \frac{e^0}{e^{15.705} + e^0 + e^0 + e^{-15.705}} \approx 0$$

$$\hat{y}_2 = \frac{e^{z_3}}{e^{z_1} + e^{z_2} + e^{z_3} + e^{z_4}} = \frac{e^0}{e^{15.705} + e^0 + e^0 + e^{-15.705}} \approx 0$$

$$\hat{y}_3 = \frac{e^{z_4}}{e^{z_1} + e^{z_2} + e^{z_3} + e^{z_4}} = \frac{e^{-15.705}}{e^{15.705} + e^0 + e^0 + e^{-15.705}} \approx 0$$

# 1-Sample

ดังนั้น เราจะได้  $\hat{y}$  ของข้อมูลชุดนี้คือ



# Prediction

**1-Sample**



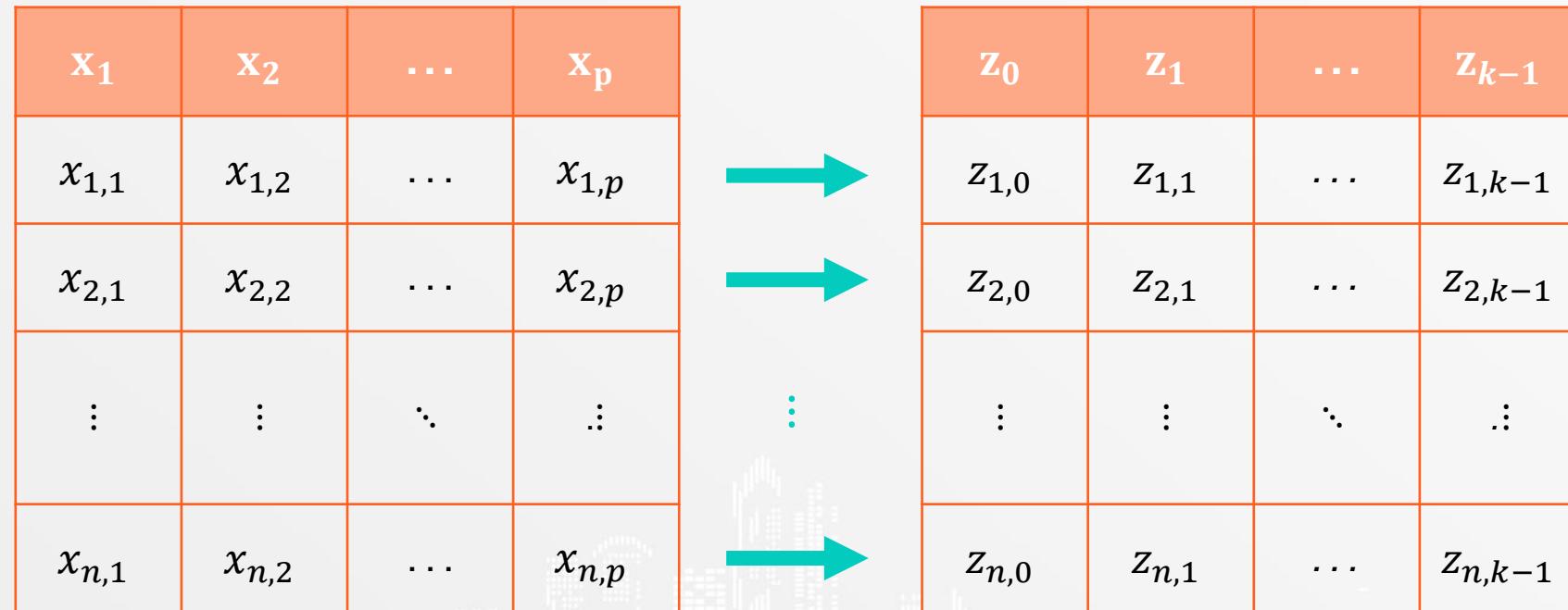
Multi-Sample



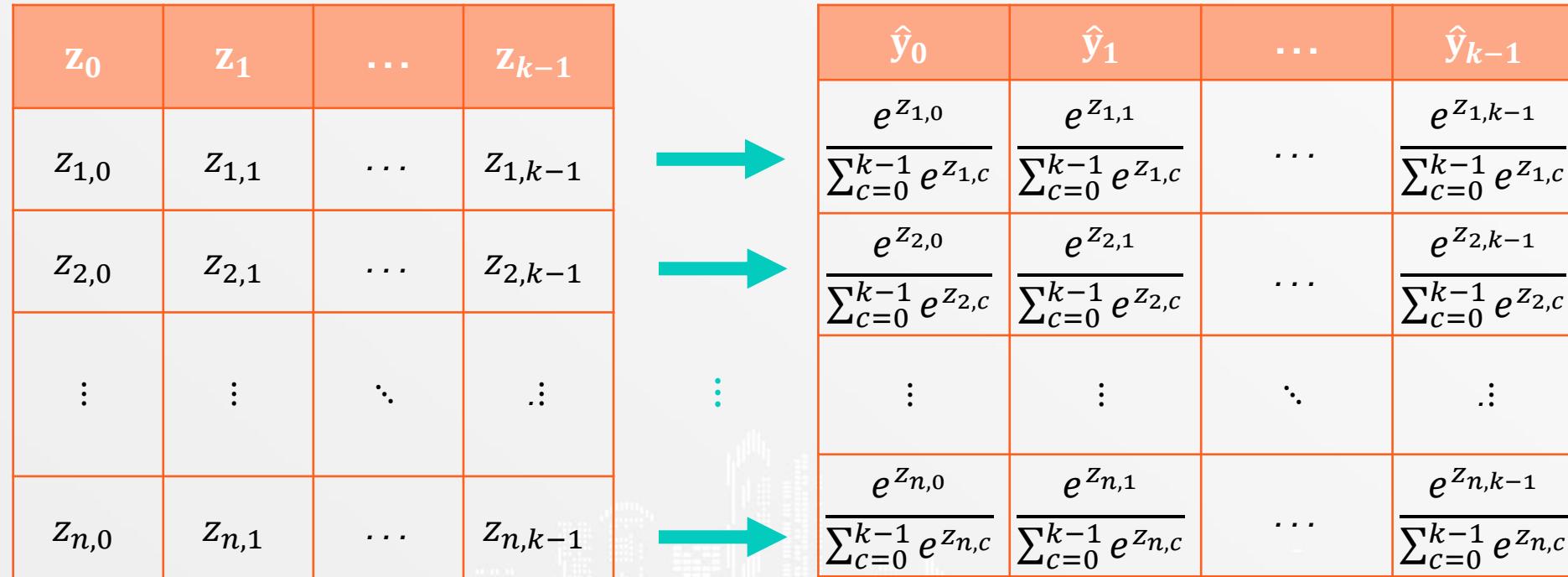
Code



# Multi-Sample



# Multi-Sample



# Multi-Sample

ตัวอย่างการคำนวณ  $\hat{y}$

$x_1$	$x_2$
0	1.5
1.5	0
-1.5	0
0	-1.5



$\hat{y}$
?
?
?
?

# Multi-Sample

- สมมติว่า  $W$  ของปัญหานี้ที่เราหามาได้คือ

$$W = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 10.47 & -10.47 & 0 \\ 10.47 & 0 & 0 & -10.47 \end{bmatrix}$$

- และจากข้อมูลใน dataset เราสามารถเขียน  $X_b$  ได้ดังต่อไปนี้

$$X_b = \begin{bmatrix} 1 & 0 & 1.5 \\ 1 & 1.5 & 0 \\ 1 & -1.5 & 0 \\ 1 & 0 & -1.5 \end{bmatrix}$$

# Multi-Sample

เราคำนวณค่า  $Z$  ได้จาก  $Z = X_b W$

$$\begin{aligned} Z &= \begin{bmatrix} 1 & 0 & 1.5 \\ 1 & 1.5 & 0 \\ 1 & -1.5 & 0 \\ 1 & 0 & -1.5 \end{bmatrix} \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 10.47 & -10.47 & 0 \\ 10.47 & 0 & 0 & -10.47 \end{bmatrix} \\ &= \begin{bmatrix} 15.705 & 0 & 0 & -15.705 \\ 0 & 15.705 & -15.705 & 0 \\ 0 & -15.705 & 15.705 & 0 \\ -15.705 & 0 & 0 & 15.705 \end{bmatrix} \end{aligned}$$

# Multi-Sample

จาก  $\hat{y}_m = \frac{e^{z_m}}{\sum_{c=1}^k e^{z_c}}$  จะได้ว่า

$$\hat{Y} = \begin{bmatrix} \frac{e^{15.705}}{e^{15.705} + e^0 + e^0 + e^{-15.705}} & \frac{e^{z_{1,1}}}{\sum_{c=0}^3 e^{z_{1,c}}} & \frac{e^{z_{1,2}}}{\sum_{c=0}^3 e^{z_{1,c}}} & \frac{e^{z_{1,3}}}{\sum_{c=0}^3 e^{z_{1,c}}} \\ \frac{e^{z_{2,0}}}{\sum_{c=0}^3 e^{z_{2,c}}} & \frac{e^{z_{2,1}}}{\sum_{c=0}^3 e^{z_{2,c}}} & \frac{e^{z_{2,2}}}{\sum_{c=0}^3 e^{z_{2,c}}} & \frac{e^{z_{2,3}}}{\sum_{c=0}^3 e^{z_{2,c}}} \\ \frac{e^{z_{3,0}}}{\sum_{c=0}^3 e^{z_{3,c}}} & \frac{e^{z_{3,1}}}{\sum_{c=0}^3 e^{z_{3,c}}} & \frac{e^{z_{3,2}}}{\sum_{c=0}^3 e^{z_{3,c}}} & \frac{e^{z_{3,3}}}{\sum_{c=0}^3 e^{z_{3,c}}} \\ \frac{e^{z_{4,0}}}{\sum_{c=0}^3 e^{z_{4,c}}} & \frac{e^{z_{4,1}}}{\sum_{c=0}^3 e^{z_{4,c}}} & \frac{e^{z_{4,2}}}{\sum_{c=0}^3 e^{z_{4,c}}} & \frac{e^{z_{4,3}}}{\sum_{c=0}^3 e^{z_{4,c}}} \end{bmatrix}$$

# Multi-Sample

จาก  $\hat{y}_m = \frac{e^{z_m}}{\sum_{c=0}^{k-1} e^{z_c}}$  จะได้ว่า

$$\hat{Y} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

# Multi-Sample

จาก  $\hat{y}_m = \frac{e^{z_m}}{\sum_{c=0}^{k-1} e^{z_c}}$  จะได้ว่า

$$\hat{Y} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \Rightarrow \hat{\mathbf{y}} = \begin{bmatrix} 0 \\ 1 \\ 2 \\ 3 \end{bmatrix}$$

# Multi-Sample

ดังนั้น เราจะได้  $\hat{y}$  สำหรับข้อมูลชุดนี้คือ

$x_1$	$x_2$
0	1.5
1.5	0
-1.5	0
0	-1.5



$\hat{y}$
0
1
2
3

# Prediction

**1-Sample**



**Multi-Sample**



**Code**



# Code

ตัวอย่าง code สำหรับหา  $\hat{y}$

$x_1$	$x_2$
0	1
1	0
-1	0
0	-1



$\hat{y}$
?
?
?
?

# Code

- Code สำหรับหา  $\hat{y}$  จากข้อมูลของเราโดยที่  $X = \begin{bmatrix} 0 & 1.5 \\ 1.5 & 0 \\ -1.5 & 0 \\ 0 & -1.5 \end{bmatrix}$

```
1 clf.predict(X)  
  
array([0, 1, 2, 3], dtype=int64)
```

# Code

- Code สำหรับหาค่า probability ของ sample กี่จะถูกพยากรณ์ในแต่ละ class

```
1 clf.predict_proba(X)
```

```
array([[9.9999699e-01, 1.50506847e-07, 1.50506847e-07, 2.26523154e-14],  
       [1.50506824e-07, 9.9999699e-01, 2.26523135e-14, 1.50506824e-07],  
       [1.50506826e-07, 2.26523137e-14, 9.9999699e-01, 1.50506825e-07],  
       [2.26523155e-14, 1.50506848e-07, 1.50506848e-07, 9.9999699e-01]])
```

# Code

ดังนั้น เราจะได้  $\hat{y}$  สำหรับข้อมูลชุดนี้คือ

$x_1$	$x_2$
0	1
1	0
-1	0
0	-1



$\hat{y}$
1
2
3
4

# Code

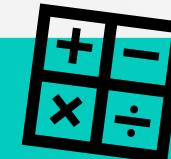


Code for this section



Open File  
**Model Creation (multi-class).ipynb**

# Code



MATH

$$Z = X_b W$$

$$\hat{y}_m = \frac{e^{z_m}}{\sum_{c=0}^{k-1} z_c}$$



CODE

```
1 clf.predict(X)
```

```
array([0, 1, 2, 3], dtype=int64)
```

# Prediction

**1-Sample**



**Multi-Sample**



**Code**



# Logistic Regression (Multi-Class)



# Logistic Regression

**Logistic Regression  
(Binary)**



**Logistic Regression  
(Multi-Class)**



**Workshop**



# Workshop

# Workshop

AI in Healthcare

Fake News

AI in Fruit Industry

# AI in Insurance

- Abstract
- Why this project important?
- Who this project for?
- Heart Disease Dataset
- What we learn from this project?

# Abstract

สร้าง model เพื่อวินิจฉัยผู้ป่วยโรคหัวใจ โดย feature ที่นำมาใช้ คือ ข้อมูลสภาวะร่างกาย เช่น

- ลักษณะการเจ็บหน้าอก
- ค่าความเข้มข้นน้ำตาลในเลือด
- ระดับคอเลสเตอรอลในเลือด



# Why this project important?



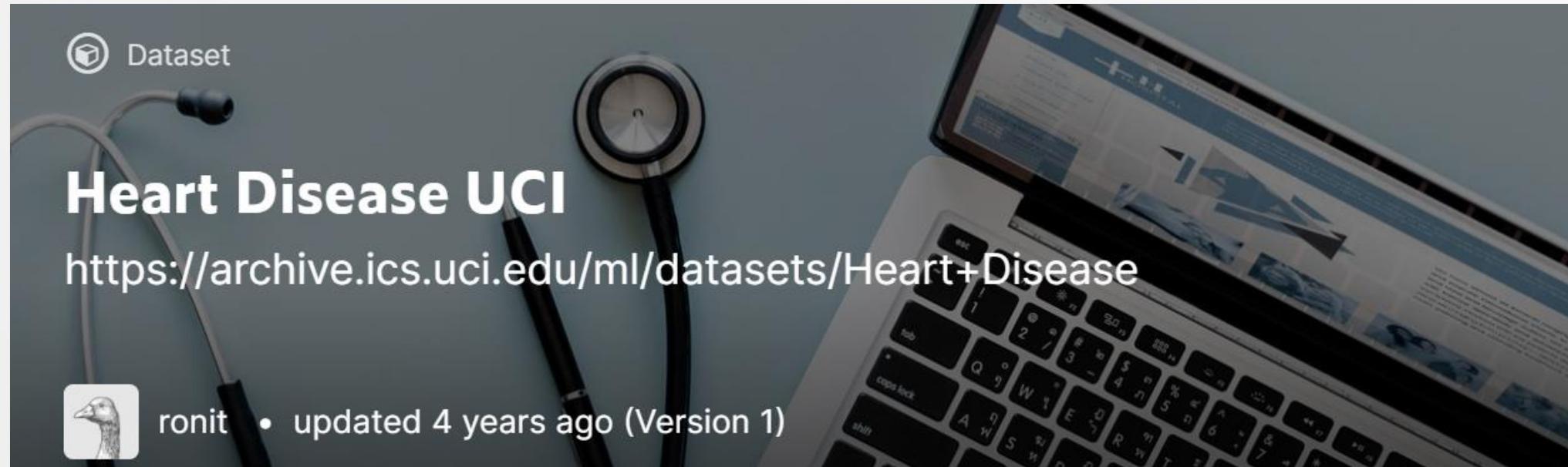
- ◆ สามารถสร้างระบบสำหรับตรวจโรคหัวใจที่ทำงานได้ตลอด 24 ชั่วโมง
- ◆ สามารถนำไปต่อยอดกับการวินิจฉัยโรคอื่น ๆ
- ◆ สามารถใช้เป็นพื้นฐานสำหรับการแพทย์ทางไกล

# Who this project is for?

- ◆ ผู้บริหารโรงพยาบาล
- ◆ บุคลากรทางการแพทย์
- ◆ นักวิเคราะห์ข้อมูล



# Heart Disease Dataset



<https://www.kaggle.com/ronitf/heart-disease-uci>

# Heart Disease Dataset

## Feature

- age : อายุ
- sex : เพศ (1 = ชาย, 0 = หญิง)
- cp : ลักษณะการเจ็บหน้าอก (0, 1, 2, 3)
- tressph : ความดันโลหิตขณะพักร
- chol : ระดับคอเลสเตอโรลในเลือด
- fbs : ค่าความเข้มข้นน้ำตาลในเลือด  $> 120 \text{ mg/dl}$  (1 = จริง, 0 = เท็จ)
- restecg : ผลคลื่นไฟฟ้าหัวใจขณะพัก

# Heart Disease Dataset

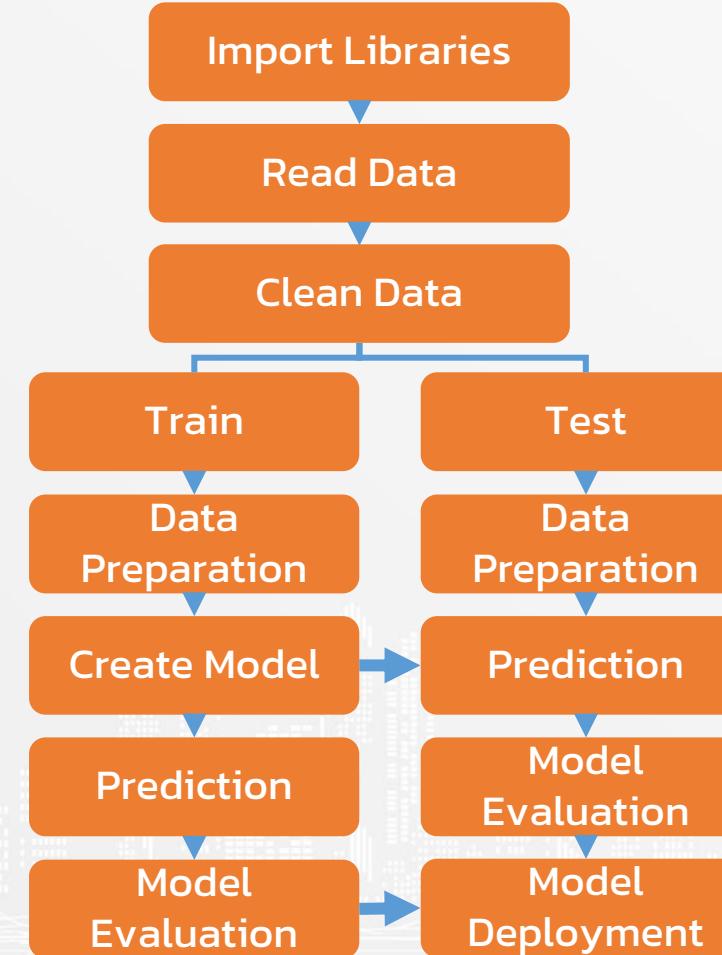
## Feature

- thalach : อัตราการเต้นหัวใจสูงสุด
- exang : อาการเจ็บหน้าอกระหว่างออกกำลังกาย (1 = เจ็บ, 0 = ไม่เจ็บ)
- oldpeak : การเกิดกราฟ ST depression ในผลการตรวจคลื่นไฟฟ้าของหัวใจ
- slop : ลักษณะความชันของกราฟ ST segment (0 = ชันขึ้น, 1 = ราบ, 2 = ชันลง)
- ca : จำนวนเส้นเลือดตีบ
- thal : ลักษณะความเครียดของหัวใจ (0, 1, 2, 3)

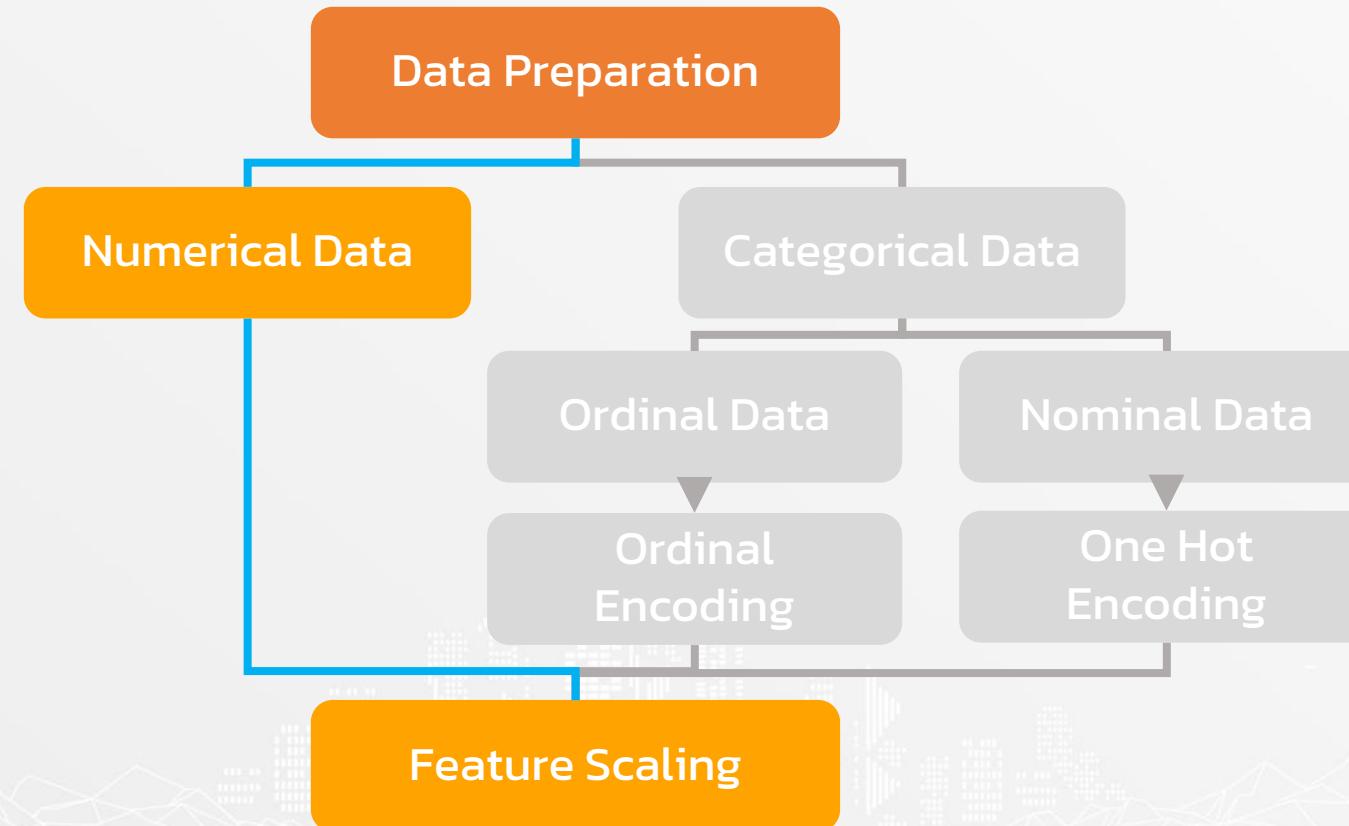
## Target

- target : การเป็นโรคหัวใจ (1 = เป็น, 0 = ไม่เป็น)

# What we learn from this project?



# Data Preparation





## 01. HEART DISEASE



# Workshop

**AI in Healthcare**

**Fake News**

**AI in Fruit Industry**

# Fake News

- Abstract
- Why this project important?
- Who this project for?
- Fake News Dataset
- What we learn from this project?

# Abstract

สร้าง model เพื่อตรวจสอบข่าวปลอมโดยพิจารณาจากหัวข้อข่าว, เนื้อหาข่าว และหมวดหมู่ข่าว



# Why this project important?



- ◆ สามารถสร้างระบบตรวจสอบข่าวปลอมที่ทำงานได้ตลอด 24 ชั่วโมง
- ◆ สามารถนำไปต่อยอดเพื่อจัดอันดับความน่าเชื่อถือของสื่อมวลชน
- ◆ สามารถนำไปประยุกต์ใช้กับงานที่มีลักษณะใกล้เคียงได้ เช่น sentimental analysis

# Who this project is for?

- ◆ บุคลากรด้านสื่อเมืองชน
- ◆ นักลงทุน
- ◆ นักวิเคราะห์ข้อมูล



# Fake News Dataset



<https://www.kaggle.com/c/fake-news/data>

# Fake News Dataset

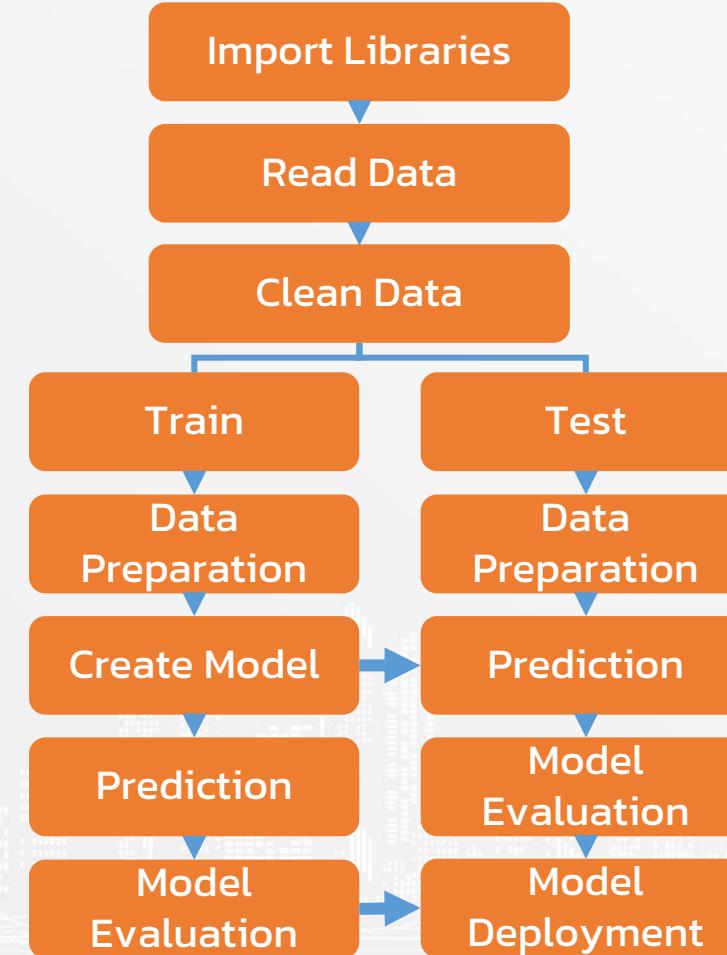
## Feature

- title : หัวข้อข่าว
- text : เนื้อหาข่าว
- subject : หมวดหมู่ข่าว

## Target

- class : ค่าความจริงของข่าว (fake, true)

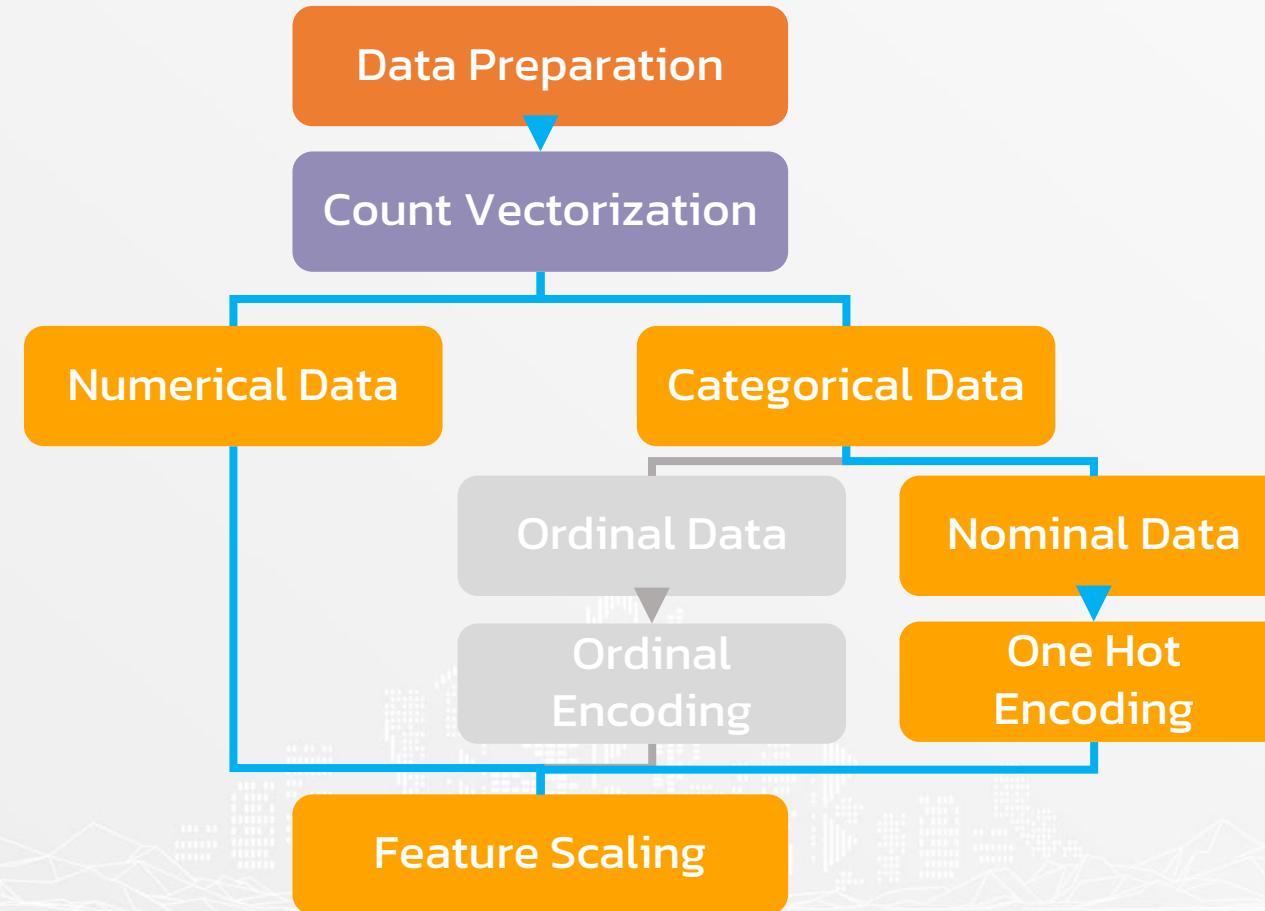
# What we learn from this project?



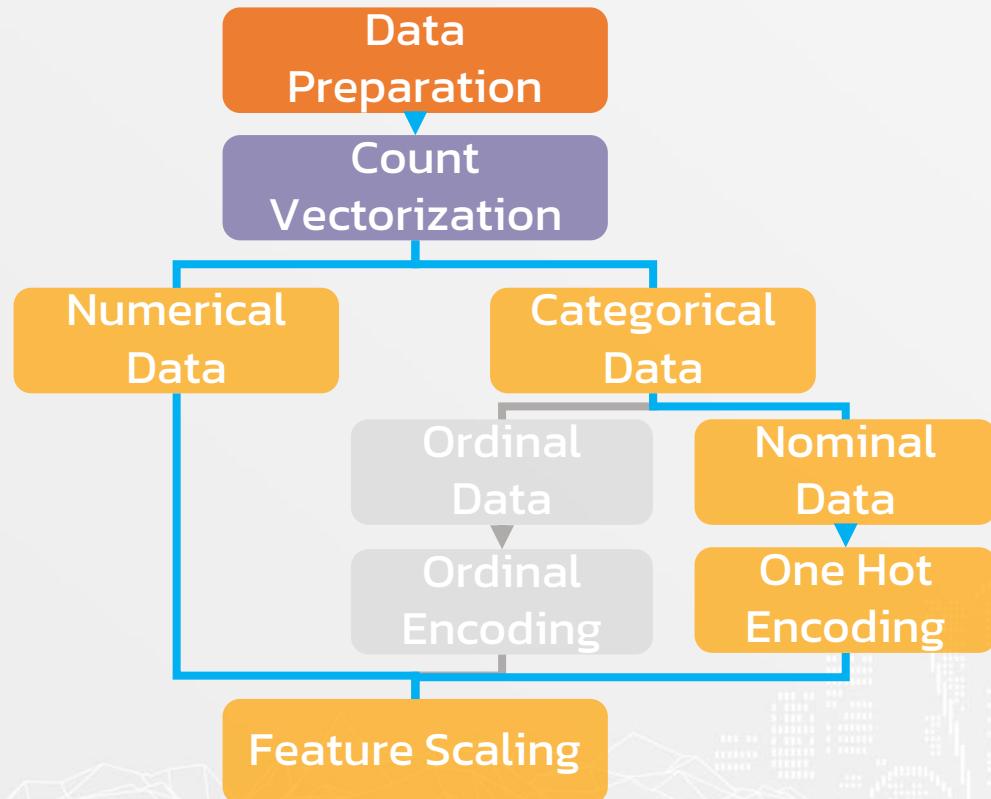
# Import Libraries

```
1 import numpy as np
2 import pandas as pd
3 import matplotlib.pyplot as plt
4
5 from sklearn.model_selection import train_test_split
6 from sklearn.feature_extraction.text import CountVectorizer
7 from sklearn.preprocessing import OrdinalEncoder, OneHotEncoder, StandardScaler, MinMaxScaler
8 from sklearn.linear_model import LogisticRegression, LogisticRegressionCV
9 from sklearn.metrics import plot_confusion_matrix, classification_report
10
11 import warnings
12 warnings.filterwarnings('ignore')
13
14 np.random.seed(12345)
```

# Data Preparation



# Data Preparation



## Count vectorization

สร้าง feature ใหม่ โดยการหา unique word จากข้อความทั้งหมดใน dataset จากนั้นให้พิจารณาว่าแต่ละข้อความประกอบด้วย unique word อะไรบ้าง และจำนวนกี่ครั้ง

	'apple'	'green'	'is'	'kiwi'	'orange'	'red'
'Apple is red'	1	0	1	0	0	1
'Kiwi is green'	0	1	1	1	0	0
'Orange is orange'	0	0	1	0	2	0

# Count Vectorization

	cnt_title _000	cnt_title _10	cnt_title _100	...	cnt_title _year	cnt_title _years	...
BAGHDAD (Reuters) – A Russian Islamic State fi...	0	0	0	...	0	0	...
WASHINGTON (Reuters) – President Donald Trump ...	0	0	0	...	0	0	...
WASHINGTON (Reuters) – Russian President Vlad... ...	0	0	0	...	0	0	...
...	...	...	...	...	...	...	...

# Code

- Count vectorization for **training set**

```
1 corpus_train = X_train["title"].tolist()
2 title_vectorizer = CountVectorizer(max_features=1000)
3 title_vectorizer.fit(corpus_train)
4 title_cnt_vec_train = title_vectorizer.transform(corpus_train).toarray()
```

```
1 title_cnt_vec_feature_name = [
2     "cnt_title_" + feature for feature in title_vectorizer.get_feature_names()
3 ]
```

```
1 X_train[title_cnt_vec_feature_name] = title_cnt_vec_train
2 X_train.drop("title", axis=1, inplace=True)
```

# Code

- Count vectorization for **test set**

```
1 corpus_test = X_test['title'].tolist()  
2 title_cnt_vec_test = title_vectorizer.transform(corpus_test).toarray()
```

```
1 X_test[title_cnt_vec_feature_name] = title_cnt_vec_test  
2 X_test.drop('title', axis=1, inplace=True)
```



## 03. FAKE NEWS



# Workshop

**AI in Healthcare**

**Fake News**

**AI in Fruit Industry**

# AI in Fruit Industry

- Abstract
- Why this project important?
- Who this project for?
- Fruit Dataset
- What we learn from this project?

# Abstract

สร้าง model เพื่อจำแนกผลไม้สด และผลไม้เสีย สำหรับ apple, banana และ orange โดยพิจารณาจากรูปผลไม้



# Why this project important?



- ◆ สามารถสร้างระบบคัดแยกผลผลิตที่ทำงานได้อย่างมีประสิทธิภาพ
- ◆ สามารถนำความรู้ไปต่อยอดเพื่อสร้าง smart farm
- ◆ สามารถนำไปต่อยอดเพื่อจำแนกผลไม้ หรือ สินค้าชนิดอื่น

# Who this project is for?

- ◆ เกษตรกรที่สนใจ AI กับการเกษตร
- ◆ ผู้ควบคุมสายการผลิต
- ◆ นักวิเคราะห์ข้อมูล



# Fruit Dataset



Dataset

## Fruits fresh and rotten for classification

Apples Oranges Bananas

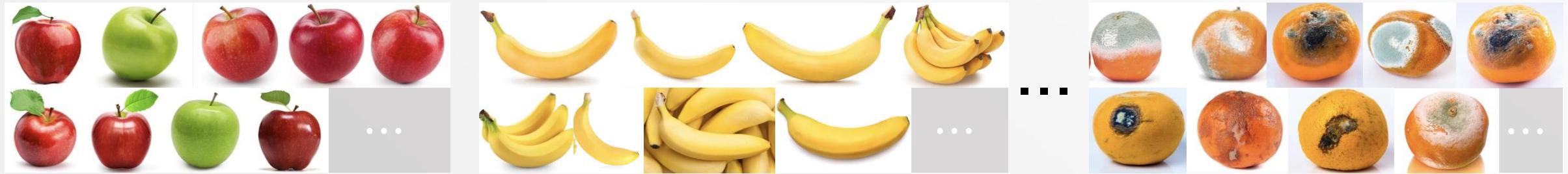


Sriram Reddy Kalluri • updated 3 years ago (Version 1)

<https://www.kaggle.com/sriramr/fruits-fresh-and-rotten-for-classification>

# Fruit Dataset

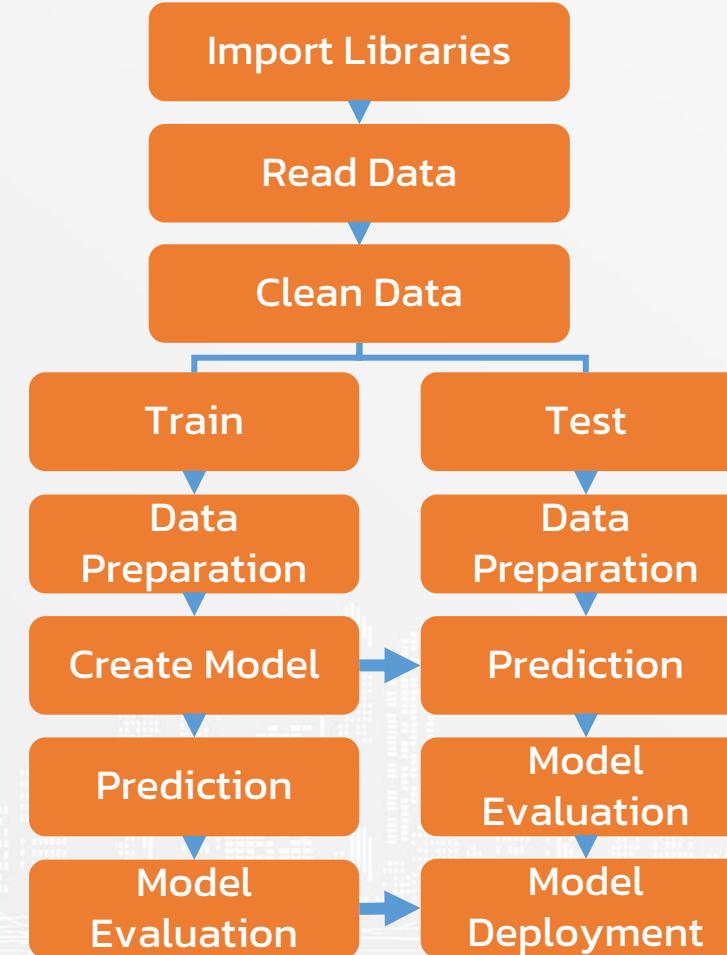
## Feature



## Target

- target : **freshapples**, **freshbanana**, **freshoranges**, **rottenapples**, **rottenbanana**, **rottenoranges**

# What we learn from this project?



# Import Libraries

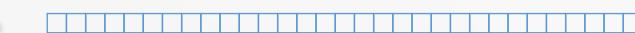
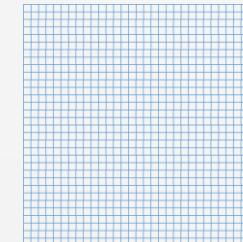
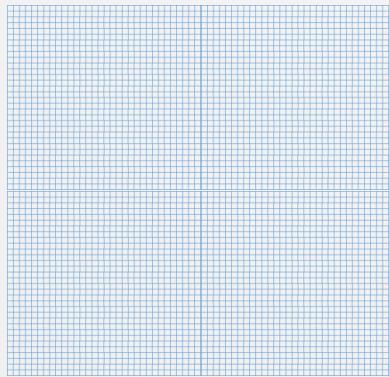
```
1 import numpy as np
2 import pandas as pd
3 import matplotlib.pyplot as plt
4
5 from glob import glob
6 from PIL import Image
7 import cv2
8 from tqdm.auto import tqdm
9 from sklearn.model_selection import train_test_split
10 from sklearn.preprocessing import OrdinalEncoder, OneHotEncoder, StandardScaler, MinMaxScaler
11 from sklearn.linear_model import LogisticRegression, LogisticRegressionCV
12 from sklearn.metrics import (
13     plot_confusion_matrix,
14     classification_report
15 )
16
17 import warnings
18 warnings.filterwarnings('ignore')
19
20 np.random.seed(12345)
```

# Read Data

```
1 classes = ['freshapples', 'freshbanana', 'freshoranges',
2             'rottenapples', 'rottenbanana', 'rottenoranges']
```

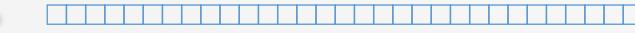
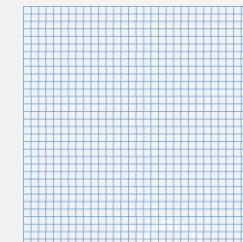
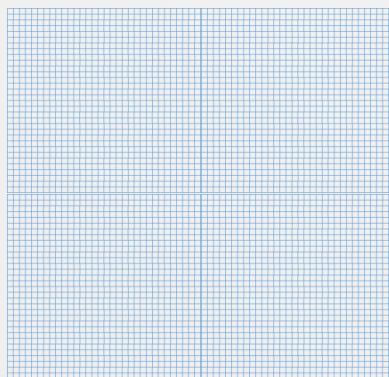
```
1 X = np.empty([0, 32*32*3])
2 y = np.empty([0, 1])
3
4 for _class in tqdm(classes):
5     img_path = glob('dataset/' + _class + '*')
6     for path in tqdm(img_path):
7         img = Image.open(path)
8         img = img.resize((32, 32))
9         img = np.array(img)
10        if img.shape[2] == 4:
11            img = cv2.cvtColor(img, cv2.COLOR_BGRA2BGR)
12        img = img.reshape(1, -1)
13        X = np.vstack([X, img])
14        y = np.vstack([y, _class])
```

# Read Data



**freshapples**

**32x32**



**freshbanana**

**32x32**

# Read Data

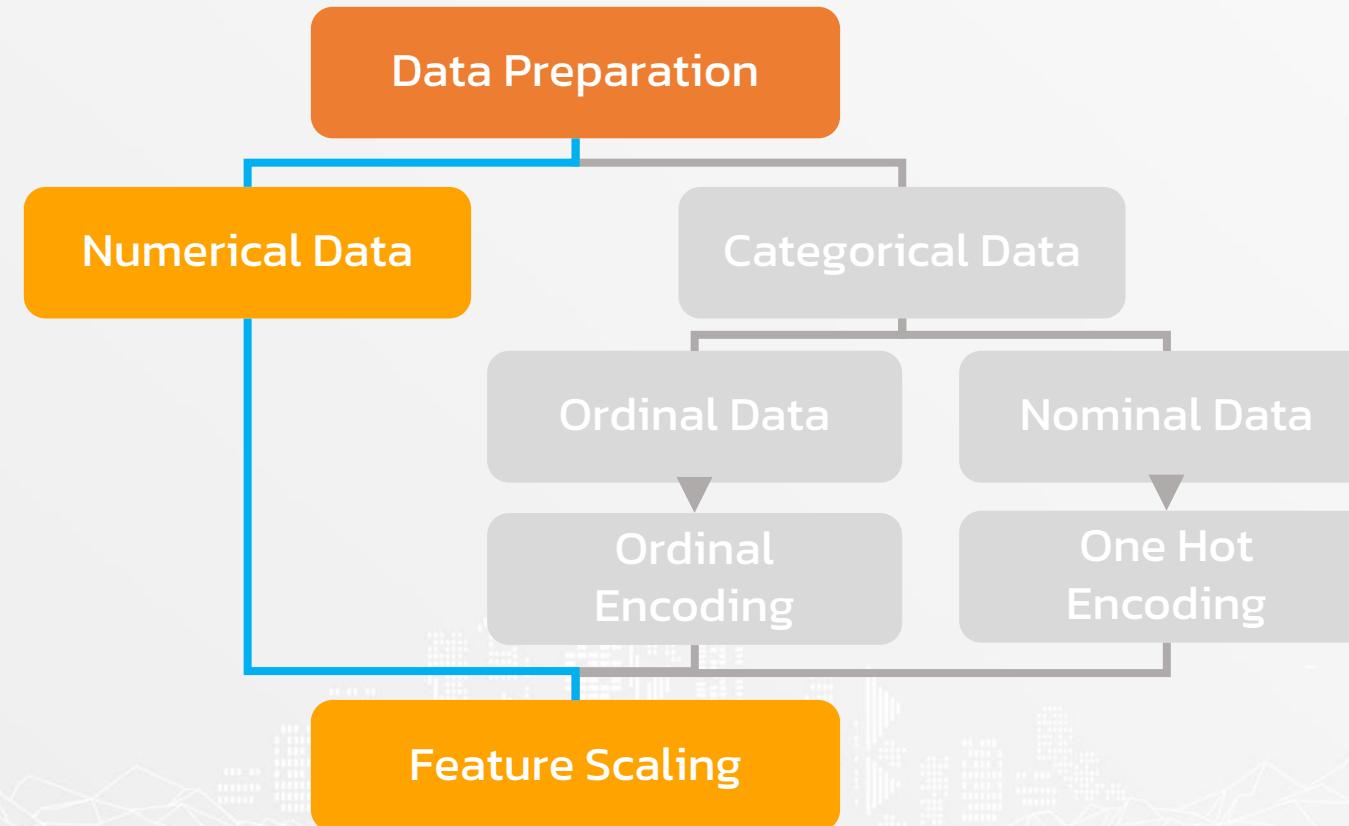
x <sub>1</sub>	x <sub>2</sub>	x <sub>3</sub>	...	x <sub>3072</sub>
0.0	0.0	0.0	...	0.0
0.0	0.0	0.0	...	0.0
0.0	0.0	0.0	...	0.0
:	:	:	:	:
255.0	255.0	255.0	...	255.0

X

y
freshapples
freshapples
freshapples
:
rottenoranges

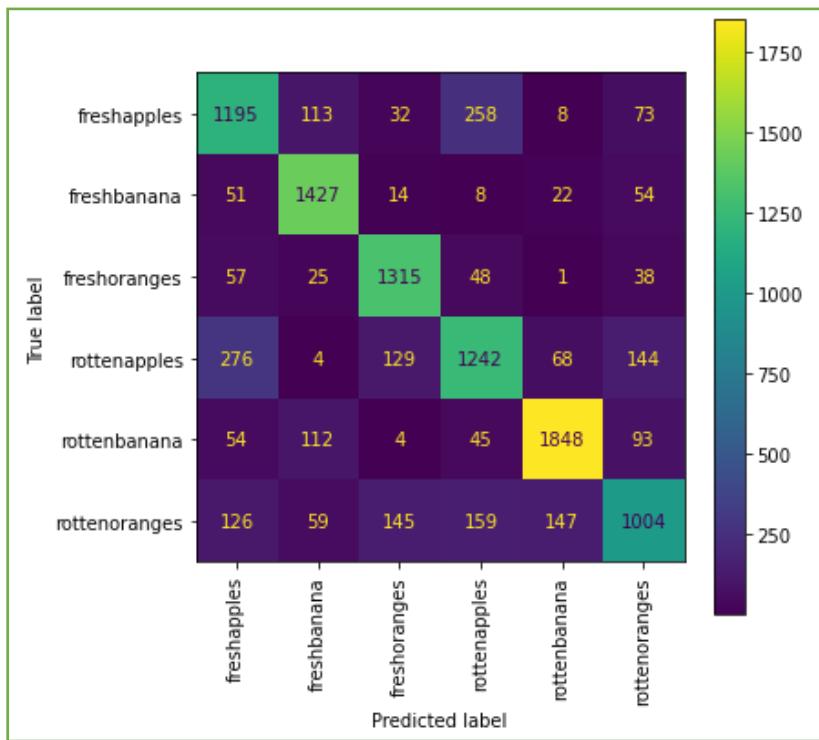
y

# Data Preparation

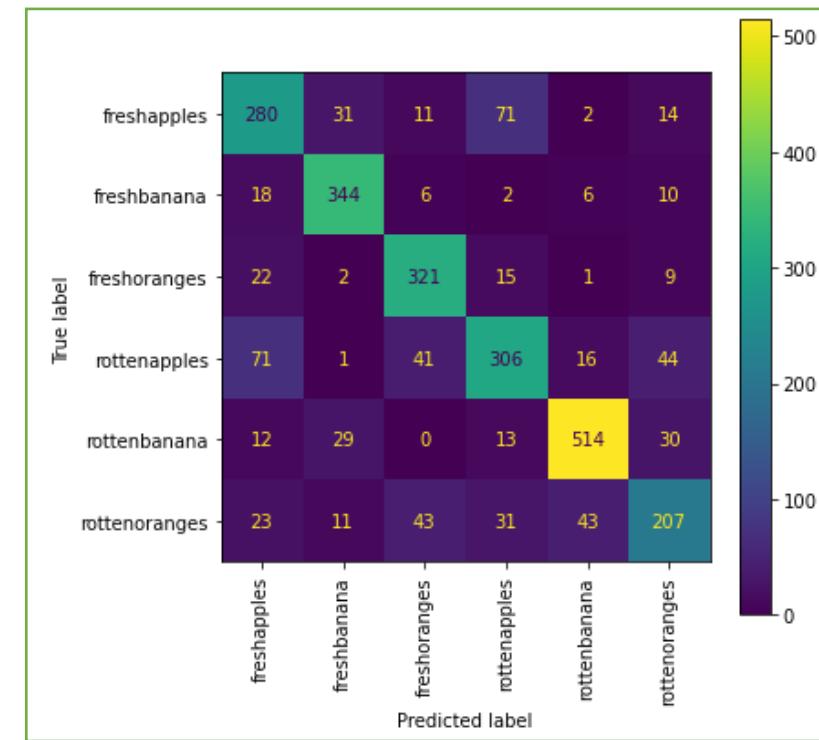


# Model Evaluation

- Confusion Matrix for **training set**



- Confusion Matrix for **test set**



# Code

- Confusion Matrix for **training set**

```
1 fig, ax = plt.subplots(figsize=(6, 6))
2 plot_confusion_matrix(clf, X_train_scaled, y_train, ax=ax)
3 plt.xticks(rotation=90)
4 plt.show()
```

- Confusion Matrix for **test set**

```
1 fig, ax = plt.subplots(figsize=(6, 6))
2 plot_confusion_matrix(clf, X_test_scaled, y_test, ax=ax)
3 plt.xticks(rotation=90)
4 plt.show()
```

# Model Evaluation

- Scoring for **training set**

	precision	recall	f1-score	support
<b>freshapples</b>	0.679363	0.711733	0.695172	1679.00000
<b>freshbanana</b>	0.820115	0.905457	0.860676	1576.00000
<b>freshoranges</b>	0.802318	0.886119	0.842139	1484.00000
<b>rottenapples</b>	0.705682	0.666667	0.685620	1863.00000
<b>rottenbanana</b>	0.882521	0.857143	0.869647	2156.00000
<b>rottenoranges</b>	0.714083	0.612195	0.659225	1640.00000
<b>accuracy</b>	0.772360	0.772360	0.772360	0.77236
<b>macro avg</b>	0.767347	0.773219	0.768746	10398.00000
<b>weighted avg</b>	0.770561	0.772360	0.770028	10398.00000

# Code

- Scoring for **training set**

```
1 report = classification_report(y_train, y_pred_train, output_dict=True)
```

```
1 print('accuracy =', report['accuracy'])
```

```
1 pd.DataFrame.from_dict(report).T
```

# Model Evaluation

- Scoring for **test set**

	precision	recall	f1-score	support
<b>freshapples</b>	0.657277	0.684597	0.670659	409.000000
<b>freshbanana</b>	0.822967	0.891192	0.855721	386.000000
<b>freshoranges</b>	0.760664	0.867568	0.810606	370.000000
<b>rottenapples</b>	0.698630	0.638831	0.667394	479.000000
<b>rottenbanana</b>	0.883162	0.859532	0.871186	598.000000
<b>rottenoranges</b>	0.659236	0.578212	0.616071	358.000000
<b>accuracy</b>	0.758462	0.758462	0.758462	0.758462
<b>macro avg</b>	0.746989	0.753322	0.748606	2600.000000
<b>weighted avg</b>	0.756430	0.758462	0.756053	2600.000000

# Code

- Scoring for **test set**

```
1 report = classification_report(y_test, y_pred_test, output_dict=True)
```

```
1 print('accuracy =', report['accuracy'])
```

```
1 pd.DataFrame.from_dict(report).T
```



## 05. FRUIT INDUSTRY



# Workshop

**AI in Healthcare**

**Fake News**

**AI in Fruit Industry**

# Logistic Regression

**Logistic Regression  
(Binary)**



**Logistic Regression  
(Multi-Class)**



**Workshop**

