

# Machine Learning

---

Logistic Regression

Dr. Shuang LIANG

# Recall: Linear Regression

**Model**  $y = b + wx_1$

**Loss**  $e = |y - \hat{y}|$   $L$  is mean absolute error (**MAE**)  
 $e = (y - \hat{y})^2$   $L$  is mean square error (**MSE**)

**Optimization** Gradient Descent

**Regularization** L1 Regularization – Lasso  
L2 Regularization – Ridge Regression

# Recall: Gradient Descent

$$w^*, b^* = \arg \min_{w, b} L$$

➤ (Randomly) Pick initial values  $w^0, b^0$

➤ Compute

$$\frac{\partial L}{\partial w} \Big|_{w=w^0, b=b^0}$$
$$\frac{\partial L}{\partial b} \Big|_{w=w^0, b=b^0}$$

$$w^1 \leftarrow w^0 - \eta \frac{\partial L}{\partial w} \Big|_{w=w^0, b=b^0}$$

$$b^1 \leftarrow b^0 - \eta \frac{\partial L}{\partial b} \Big|_{w=w^0, b=b^0}$$

➤ Update  $w$  and  $b$  iteratively

# Today's Topics

- Type of classifiers
- Logistic Regression
- Logistic Regression vs Linear Regression
- Limitation of Logistic Regression

# Today's Topics

- *Type of classifiers*
- Logistic Regression
- Logistic Regression vs Linear Regression
- Limitation of Logistic Regression

# Types of classifiers

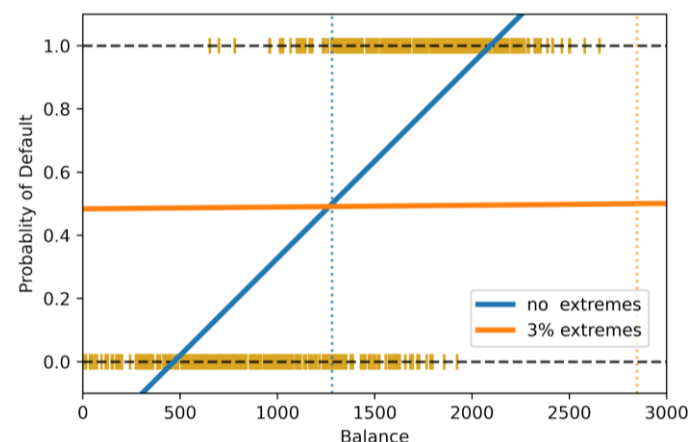
- Instance based classifiers
  - Use observation directly (no models)
  - e.g. K nearest neighbors
- Generative
  - build a generative statistical model
  - e.g., Bayesian networks
- ***Discriminative***
  - *directly estimate a decision rule/boundary*
  - *e.g., decision tree, logistic regression*

# Today's Topics

- Type of classifiers
- *Logistic Regression*
- Logistic Regression vs Linear Regression
- Limitation of Logistic Regression

# Motivation

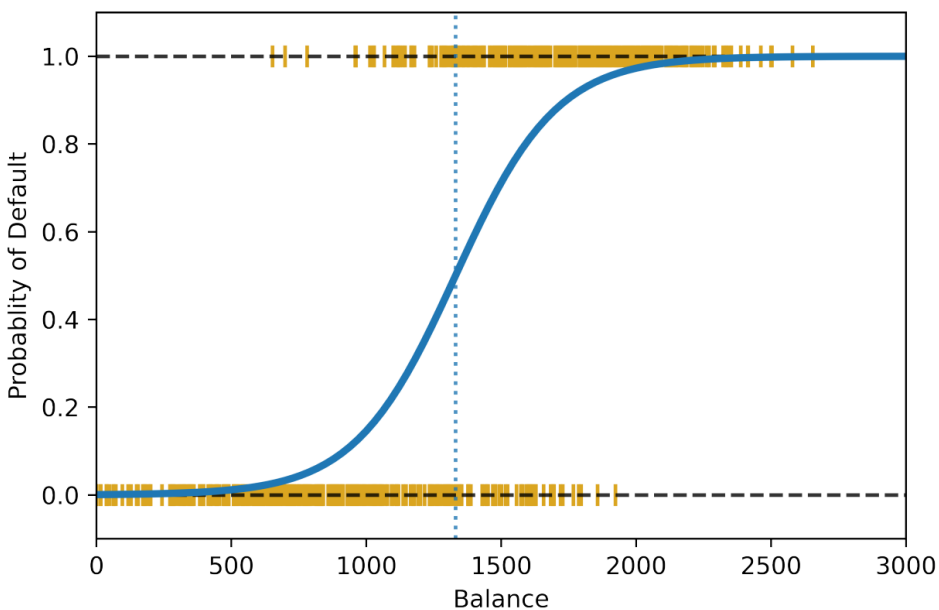
- Rather than modeling the output  $y$  directly, we can **model the probability** that  $x$  belongs to a particular category.
- In the previous lecture, we used a linear regression model but
  - The predicted value is not in  $[0,1]$
  - Very large or small values of the prediction contribute to the error even if they indicate we are very confident in the resulting classification
- **Solution:** map the prediction from  $(-\infty, +\infty)$  to  $[0,1]$



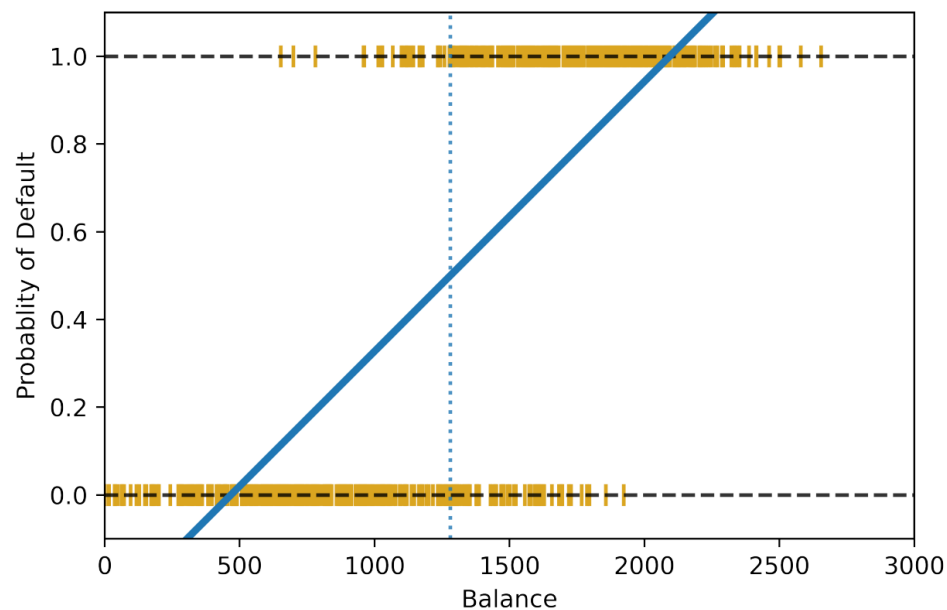


# Motivation

- **Solution:** map the prediction from  $(-\infty, +\infty)$  to  $[0,1]$



Logistic regression



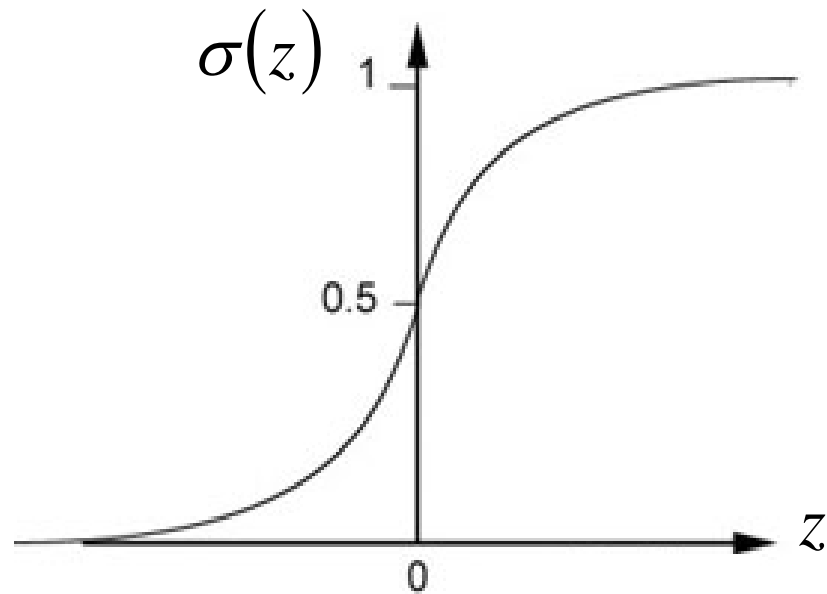
Linear regression

# The Logistic Function - Sigmoid

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

Try to calculate two formulas:

- $1 - \sigma(z)$
- $\sigma'(z)$



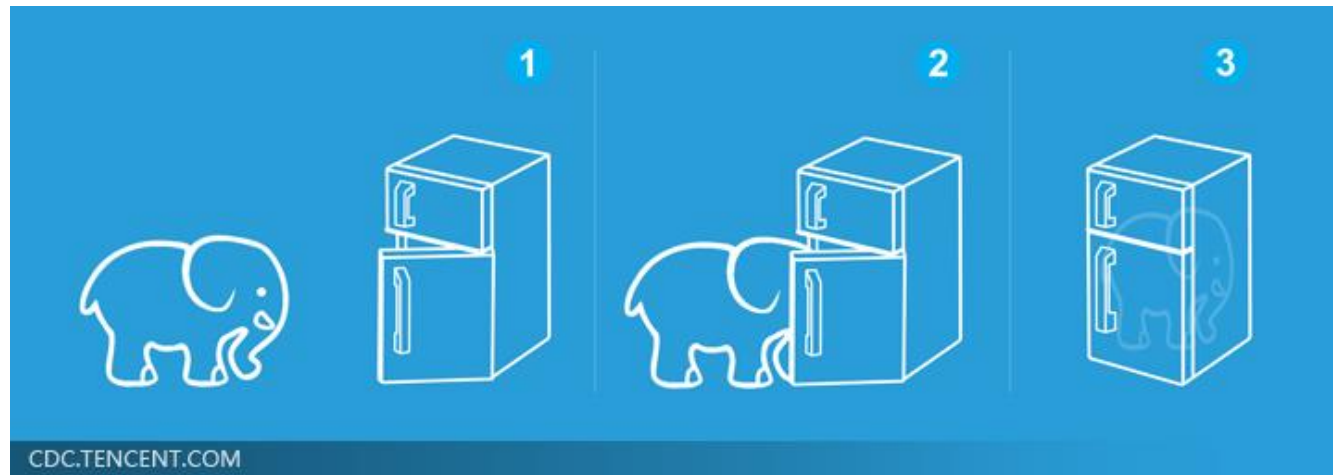
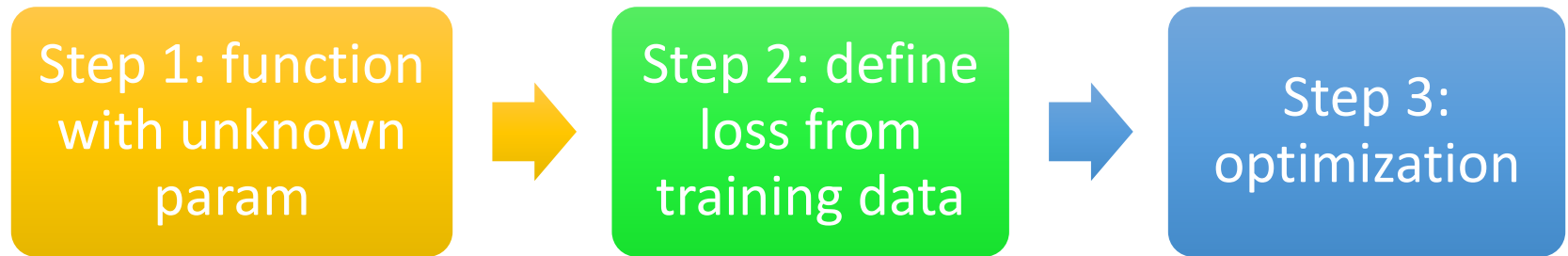
# The Logistic Function - Sigmoid

- **Properties**

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

- $\sigma'(z) = -\frac{-e^{-z}}{(1+e^{-z})^2} = \frac{1}{(1+e^{-z})} \frac{1}{(1+e^z)} = \sigma(z)(1 - \sigma(z))$

# Recall: Typical process of ML



# Step1: Function Set

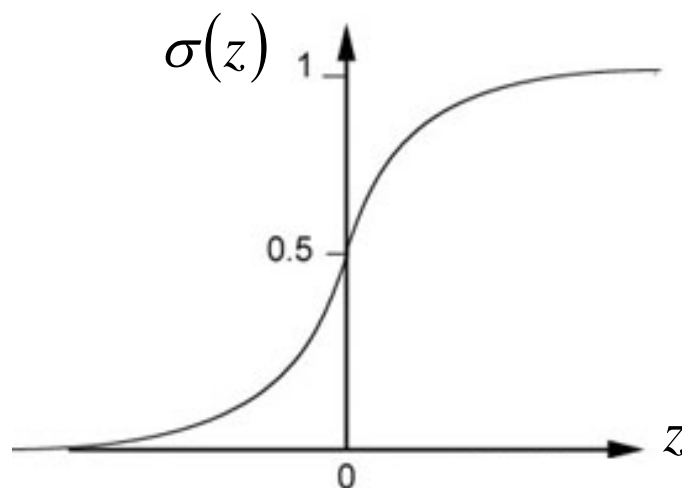
- **Label prediction:** quantize the probability
  - If  $p(1|x) \geq 1/2$ , you predict class 1
  - If  $p(1|x) < 1/2$ , you predict class 0
- Logistic regression models the probability that X belongs to a particular class using the logistic function

$$p(1|x) = P(Y = 1|X = x) = \sigma\left(\sum_i w_i x_i + b\right)$$
$$p(0|x) = P(Y = 0|X = x) = 1 - \sigma\left(\sum_i w_i x_i + b\right)$$

# Step1: Function Set

- **Interpretation**

- Very large  $|\sum_i w_i x_i + b|$  corresponds to  $p(1|x)$  very close to 0 or 1 (high confidence)
- Small  $|\sum_i w_i x_i + b|$  corresponds to  $p(1|x)$  very close to 0.5 (low confidence)



## **Logistic Regression**

Step 1:  $f_{w,b}(x) = \sigma \left( \sum_i w_i x_i + b \right)$

Output: between 0 and 1

## **Linear Regression**

$$f_{w,b}(x) = \sum_i w_i x_i + b$$

Output: any value

Step 2:

Step 3:

## Step2: Goodness of a function

Training  
Data

$x^1$	$x^2$	$x^3$	$\dots \dots$	$x^N$
$C_1$	$C_1$	$C_2$		$C_1$

Assume the data is generated based on  $f_{w,b}(x) = P_{w,b}(C_1|x)$

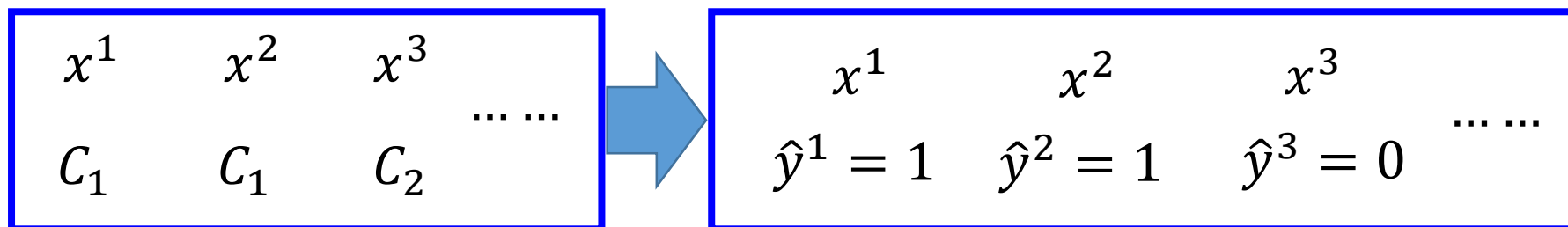
Given a set of  $w$  and  $b$ , what is its probability of generating the data?

$$L(w, b) = f_{w,b}(x^1) f_{w,b}(x^2) \left(1 - f_{w,b}(x^3)\right) \cdots f_{w,b}(x^N)$$

The most likely  $w^*$  and  $b^*$  is the one with the largest  $L(w, b)$ .

$$w^*, b^* = \arg \max_{w, b} L(w, b)$$





$\hat{y}^n$ : 1 for class 1, 0 for class 2

$$L(w, b) = f_{w,b}(x^1) f_{w,b}(x^2) (1 - f_{w,b}(x^3)) \dots$$

$$w^*, b^* = \arg \max_{w,b} L(w, b)$$

$$= w^*, b^* = \arg \min_{w,b} -\ln L(w, b)$$

$$-\ln L(w, b)$$

$$= -\ln f_{w,b}(x^1) \Rightarrow -[1 \ln f(x^1) + 0 \ln(1 - f(x^1))]$$

$$-\ln f_{w,b}(x^2) \Rightarrow -[1 \ln f(x^2) + 0 \ln(1 - f(x^2))]$$

$$-\ln(1 - f_{w,b}(x^3)) \Rightarrow -[0 \ln f(x^3) + 1 \ln(1 - f(x^3))]$$

⋮

## Step2: Goodness of a function

$$L(w, b) = f_{w,b}(x^1) f_{w,b}(x^2) (1 - f_{w,b}(x^3)) \cdots f_{w,b}(x^N)$$

$$-\ln L(w, b) = -\left[ \ln f_{w,b}(x^1) + \ln f_{w,b}(x^2) + \ln (1 - f_{w,b}(x^3)) \right] \cdots$$

$\hat{y}^n$ : 1 for class 1, 0 for class 2

$$= \sum_n \underbrace{-\left[ \hat{y}^n \ln f_{w,b}(x^n) + (1 - \hat{y}^n) \ln (1 - f_{w,b}(x^n)) \right]}_{\text{Cross entropy between two Bernoulli distribution}}$$

$$H(p, q) = - \sum_x p(x) \ln(q(x))$$

Distribution p:

$$p(x = 1) = \hat{y}^n$$

$$p(x = 0) = 1 - \hat{y}^n$$



cross  
entropy

Distribution q:

$$q(x = 1) = f(x^n)$$

$$q(x = 0) = 1 - f(x^n)$$

## Step2: Goodness of a function

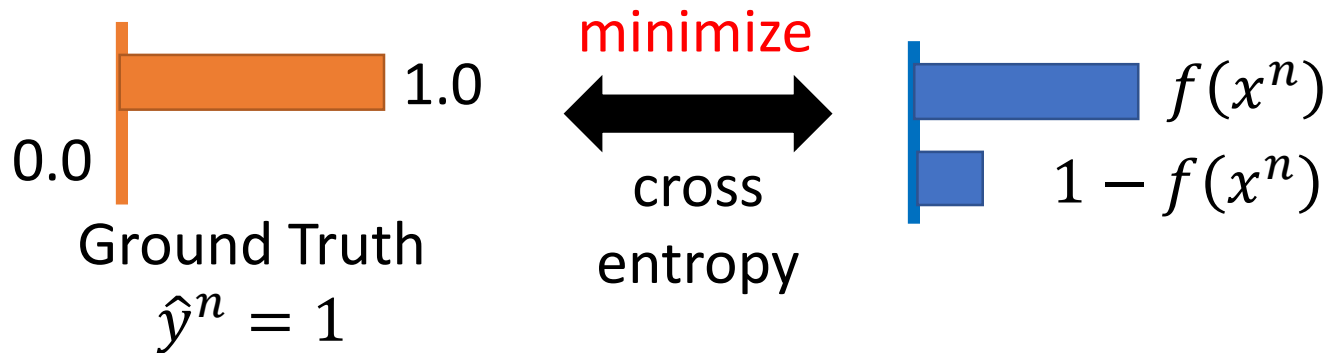
$$L(w, b) = f_{w,b}(x^1) f_{w,b}(x^2) \left(1 - f_{w,b}(x^3)\right) \cdots f_{w,b}(x^N)$$

$$-\ln L(w, b) = -\left[\ln f_{w,b}(x^1) + \ln f_{w,b}(x^2) + \ln \left(1 - f_{w,b}(x^3)\right)\right] \cdots$$

$\hat{y}^n$ : 1 for class 1, 0 for class 2

$$= \sum_n -\left[\hat{y}^n \ln f_{w,b}(x^n) + (1 - \hat{y}^n) \ln \left(1 - f_{w,b}(x^n)\right)\right]$$

Cross entropy between two Bernoulli distribution



## Logistic Regression

Step 1:  $f_{w,b}(x) = \sigma \left( \sum_i w_i x_i + b \right)$

Output: between 0 and 1

Training data:  $(x^n, \hat{y}^n)$

Step 2:  $\hat{y}^n$ : 1 for class 1, 0 for class 2

$$L(f) = \sum_n l(f(x^n), \hat{y}^n)$$

## Linear Regression

$$f_{w,b}(x) = \sum_i w_i x_i + b$$

Output: any value

Training data:  $(x^n, \hat{y}^n)$

$\hat{y}^n$ : a real number

$$L(f) = \frac{1}{2} \sum_n (f(x^n) - \hat{y}^n)^2$$

Cross entropy:

$$l(f(x^n), \hat{y}^n) = -[\hat{y}^n \ln f(x^n) + (1 - \hat{y}^n) \ln(1 - f(x^n))]$$

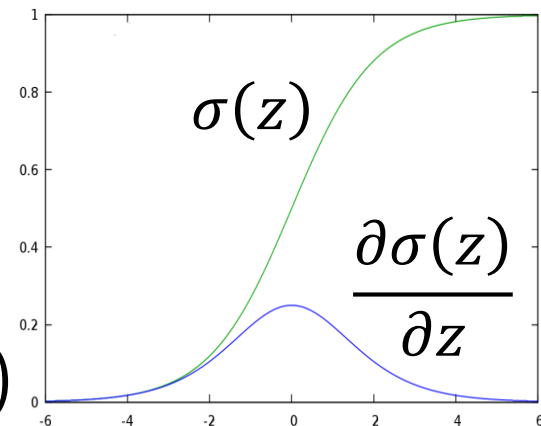
# Step3: Find the best function

- Loss: Cross-Entropy  $(1 - f_{w,b}(x^n)) x_i^n$

$$\frac{-\ln L(w, b)}{\partial w_i} = \sum_n - \left[ \hat{y}^n \frac{\ln f_{w,b}(x^n)}{\partial w_i} + (1 - \hat{y}^n) \frac{\ln (1 - f_{w,b}(x^n))}{\partial w_i} \right]$$

$$\frac{\partial \ln f_{w,b}(x)}{\partial w_i} = \frac{\partial \ln f_{w,b}(x)}{\partial z} \frac{\partial z}{\partial w_i} \quad \frac{\partial z}{\partial w_i} = x_i$$

$$\frac{\partial \ln \sigma(z)}{\partial z} = \frac{1}{\sigma(z)} \frac{\partial \sigma(z)}{\partial z} = \frac{1}{\cancel{\sigma(z)}} \cancel{\sigma(z)} (1 - \sigma(z))$$



$$\begin{aligned} f_{w,b}(x) &= \sigma(z) \\ &= 1 / (1 + \exp(-z)) \end{aligned}$$

$$z = w \cdot x + b = \sum_i w_i x_i + b$$

# Step3: Find the best function

- Loss: Cross-Entropy  $(1 - f_{w,b}(x^n)) x_i^n - f_{w,b}(x^n) x_i^n$

$$\frac{-\ln L(w, b)}{\partial w_i} = \sum_n - \left[ \hat{y}^n \frac{\ln f_{w,b}(x^n)}{\partial w_i} + (1 - \hat{y}^n) \frac{\ln (1 - f_{w,b}(x^n))}{\partial w_i} \right]$$

$$\frac{\partial \ln (1 - f_{w,b}(x))}{\partial w_i} = \frac{\partial \ln (1 - f_{w,b}(x))}{\partial z} \frac{\partial z}{\partial w_i} \quad \frac{\partial z}{\partial w_i} = x_i$$

$$\frac{\partial \ln (1 - \sigma(z))}{\partial z} = - \frac{1}{1 - \sigma(z)} \frac{\partial \sigma(z)}{\partial z} = - \frac{1}{1 - \sigma(z)} \sigma(z) (1 - \sigma(z))$$

$$\begin{aligned} f_{w,b}(x) &= \sigma(z) \\ &= 1 / (1 + \exp(-z)) \end{aligned}$$

$$z = w \cdot x + b = \sum_i w_i x_i + b$$

# Step3: Find the best function

- Loss: Cross-Entropy  $(1 - f_{w,b}(x^n)) x_i^n - f_{w,b}(x^n) x_i^n$

$$\frac{-\ln L(w, b)}{\partial w_i} = \sum_n - \left[ \hat{y}^n \frac{\ln f_{w,b}(x^n)}{\partial w_i} + (1 - \hat{y}^n) \frac{\ln (1 - f_{w,b}(x^n))}{\partial w_i} \right]$$

$$= \sum_n - \left[ \hat{y}^n \frac{(1 - f_{w,b}(x^n)) x_i^n}{\partial w_i} - (1 - \hat{y}^n) \frac{f_{w,b}(x^n) x_i^n}{\partial w_i} \right]$$

$$= \sum_n - \left[ \hat{y}^n - \cancel{\hat{y}^n f_{w,b}(x^n)} - f_{w,b}(x^n) + \cancel{\hat{y}^n f_{w,b}(x^n)} \right] x_i^n$$

$$= \sum_n - \left( \hat{y}^n - f_{w,b}(x^n) \right) x_i^n$$

Larger difference, larger update

$$w_i \leftarrow w_i - \eta \sum_n - \left( \hat{y}^n - f_{w,b}(x^n) \right) x_i^n$$

## Logistic Regression

Step 1:  $f_{w,b}(x) = \sigma \left( \sum_i w_i x_i + b \right)$

Output: between 0 and 1

Training data:  $(x^n, \hat{y}^n)$

Step 2:  $\hat{y}^n$ : 1 for class 1, 0 for class 2

$$L(f) = \sum_n l(f(x^n), \hat{y}^n)$$

## Linear Regression

$$f_{w,b}(x) = \sum_i w_i x_i + b$$

Output: any value

Training data:  $(x^n, \hat{y}^n)$

$\hat{y}^n$ : a real number

$$L(f) = \frac{1}{2} \sum_n (f(x^n) - \hat{y}^n)^2$$

Logistic regression:  $w_i \leftarrow w_i - \eta \sum_n - \left( \hat{y}^n - f_{w,b}(x^n) \right) x_i^n$

Step 3:

Linear regression:  $w_i \leftarrow w_i - \eta \sum_n - \left( \hat{y}^n - f_{w,b}(x^n) \right) x_i^n$



# Step3: Find the best function

• Loss: Square Error      Step 1:  $f_{w,b}(x) = \sigma \left( \sum_i w_i x_i + b \right)$

Step 2: Training data:  $(x^n, \hat{y}^n)$ ,  $\hat{y}^n$ : 1 for class 1, 0 for class 2

$$L(f) = \frac{1}{2} \sum_n (f_{w,b}(x^n) - \hat{y}^n)^2$$

$$\begin{aligned} \text{Step 3:} \quad \frac{\partial (f_{w,b}(x) - \hat{y})^2}{\partial w_i} &= 2(f_{w,b}(x) - \hat{y}) \frac{\partial f_{w,b}(x)}{\partial z} \frac{\partial z}{\partial w_i} \\ &= 2(f_{w,b}(x) - \hat{y}) f_{w,b}(x) (1 - f_{w,b}(x)) x_i \end{aligned}$$

$\hat{y}^n = 1$       If  $f_{w,b}(x^n) = 1$  (close to target)   $\partial L / \partial w_i = 0$

                 If  $f_{w,b}(x^n) = 0$  (far from target)   $\partial L / \partial w_i = 0$

# Step3: Find the best function


• Loss: Square Error      Step 1:  $f_{w,b}(x) = \sigma \left( \sum_i w_i x_i + b \right)$

Step 2: Training data:  $(x^n, \hat{y}^n)$ ,  $\hat{y}^n$ : 1 for class 1, 0 for class 2

$$L(f) = \frac{1}{2} \sum_n (f_{w,b}(x^n) - \hat{y}^n)^2$$

$$\begin{aligned} \text{Step 3:} \quad \frac{\partial (f_{w,b}(x) - \hat{y})^2}{\partial w_i} &= 2(f_{w,b}(x) - \hat{y}) \frac{\partial f_{w,b}(x)}{\partial z} \frac{\partial z}{\partial w_i} \\ &= 2(f_{w,b}(x) - \hat{y}) f_{w,b}(x) (1 - f_{w,b}(x)) x_i \end{aligned}$$

$\hat{y}^n = 1$       If  $f_{w,b}(x^n) = 1$  (far from target)   $\partial L / \partial w_i = 0$

                 If  $f_{w,b}(x^n) = 0$  (close to target)   $\partial L / \partial w_i = 0$

# Step3: Find the best function

- Based on Gradient Descent Method

**Loss: Cross-Entropy**

Larger difference, larger update

$$w_i \leftarrow w_i - \eta \sum_n - \left( \hat{y}^n - f_{w,b}(x^n) \right) x_i^n$$

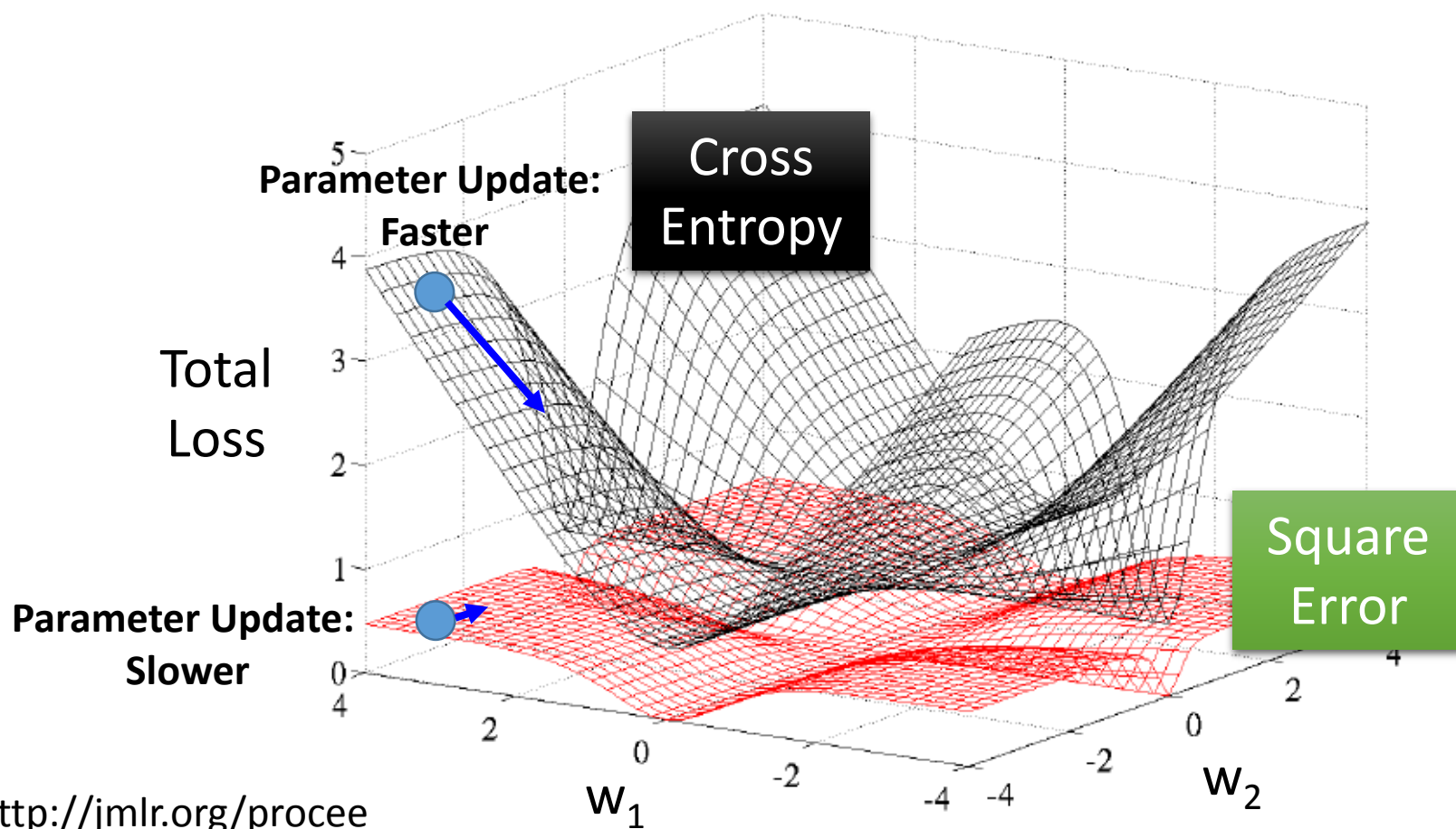
**Loss: Square Error**

$$L(f) = \frac{1}{2} \sum_n (f_{w,b}(x^n) - \hat{y}^n)^2$$

$$\frac{\partial (f_{w,b}(x) - \hat{y})^2}{\partial w_i}$$

$$= 2(f_{w,b}(x) - \hat{y})f_{w,b}(x)(1 - f_{w,b}(x))x_i$$

# Cross Entropy v.s. Square Error



<http://jmlr.org/proceedings/papers/v9/glorot10a/glorot10a.pdf>

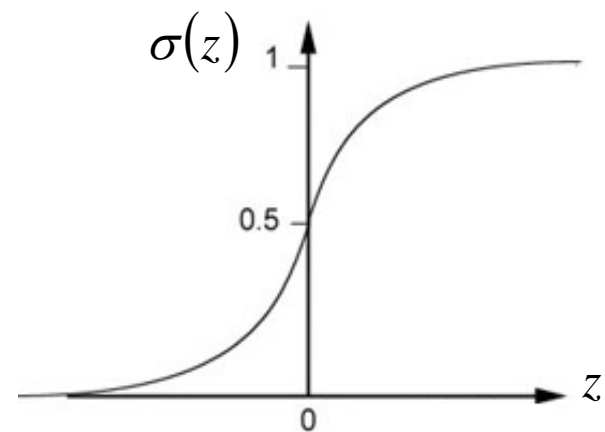
# Logistic Regression

- **Summary**

- Function set

$$f_{w,b}(x) = \sigma \left( \sum_i w_i x_i + b \right)$$

Output: between 0 and 1



- Loss: Cross Entropy

$$= \sum_n - \left[ \hat{y}^n \ln f_{w,b}(x^n) + (1 - \hat{y}^n) \ln (1 - f_{w,b}(x^n)) \right]$$

- Optimization: Gradient Descent

$$w_i \leftarrow w_i - \eta \sum_n \underline{\left( \hat{y}^n - f_{w,b}(x^n) \right) x_i^n}$$

# Today's Topics

- Type of classifiers
- Logistic Regression
- *Logistic Regression vs Linear Regression*
- Limitation of Logistic Regression

## Logistic Regression

Step 1:  $f_{w,b}(x) = \sigma \left( \sum_i w_i x_i + b \right)$

Output: between 0 and 1

Training data:  $(x^n, \hat{y}^n)$

Step 2:  $\hat{y}^n$ : 1 for class 1, 0 for class 2

$$L(f) = \sum_n l(f(x^n), \hat{y}^n)$$

Step 3: Logistic regression:  $w_i \leftarrow w_i - \eta \sum_n - \left( \hat{y}^n - f_{w,b}(x^n) \right) x_i^n$

Linear regression:  $w_i \leftarrow w_i - \eta \sum_n - \left( \hat{y}^n - f_{w,b}(x^n) \right) x_i^n$

## Linear Regression

$$f_{w,b}(x) = \sum_i w_i x_i + b$$

Output: any value

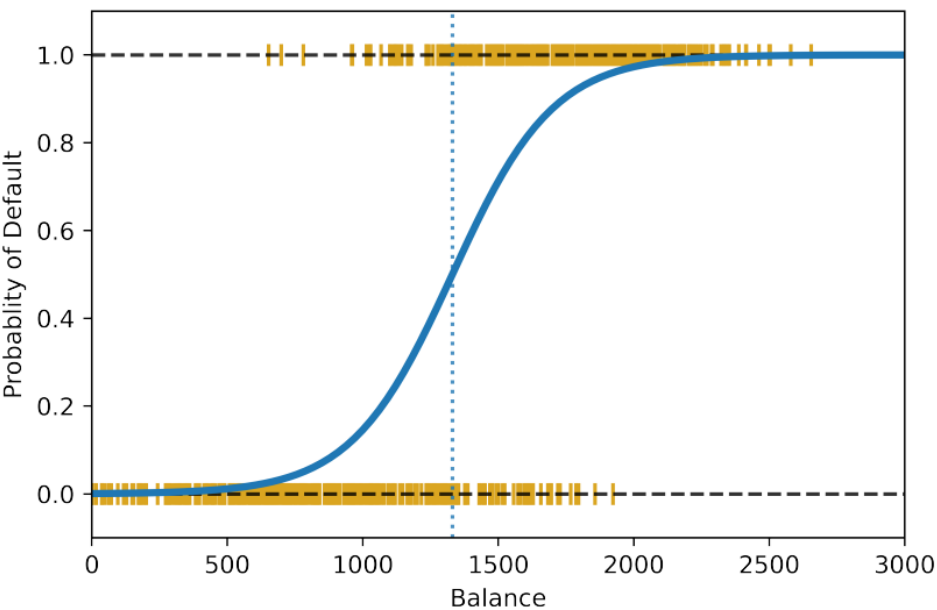
Training data:  $(x^n, \hat{y}^n)$

$\hat{y}^n$ : a real number

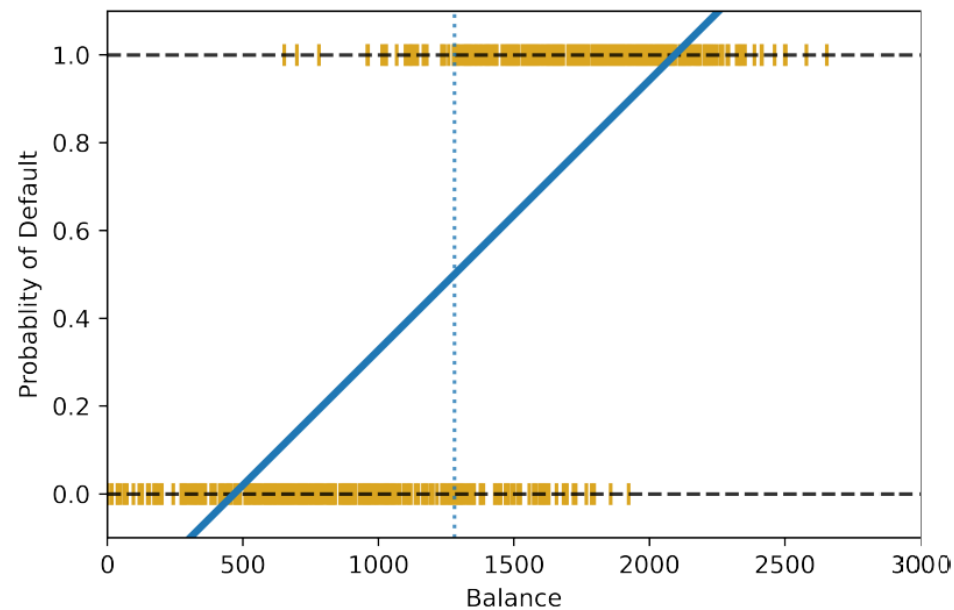
$$L(f) = \frac{1}{2} \sum_n (f(x^n) - \hat{y}^n)^2$$

# Logistic Regression v.s. Linear Regression

- **From Data**
- Comparison of logistic and linear regression for **balanced** data



**Logistic regression**

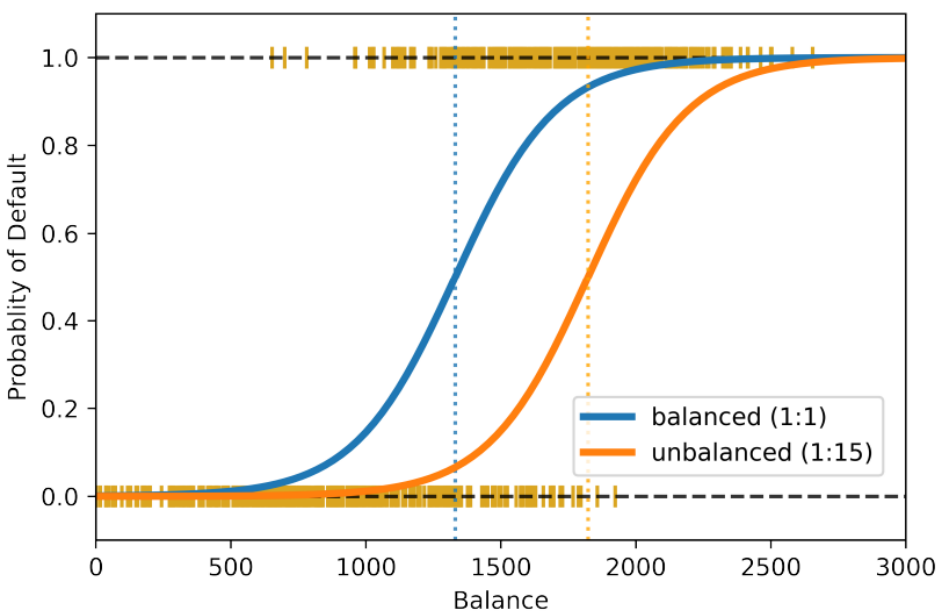


**Linear regression**

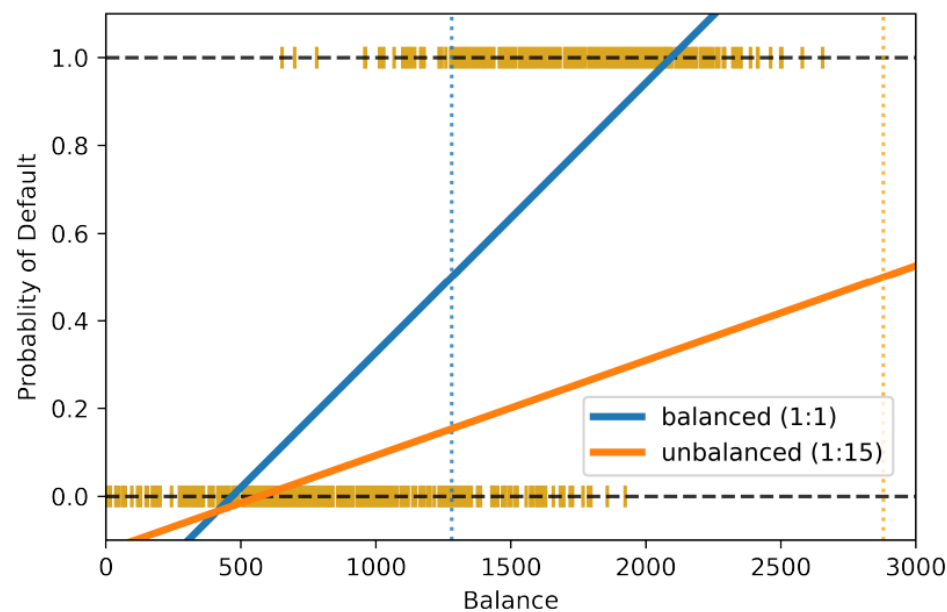


# Logistic Regression v.s. Linear Regression

- **From Data**
- Comparison of logistic and linear regression for **unbalanced** data



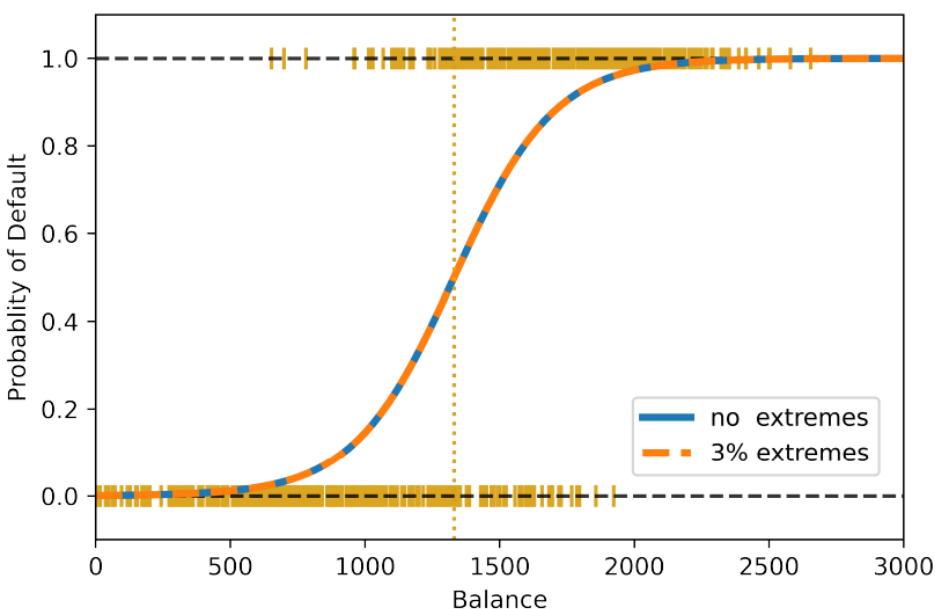
Logistic regression



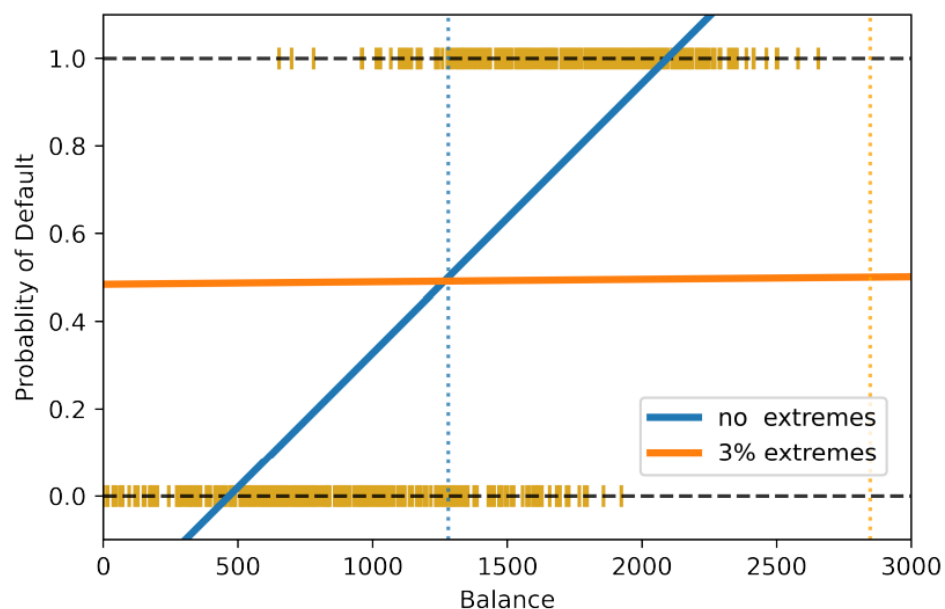
Linear regression

# Logistic Regression v.s. Linear Regression

- **From Data**
- Comparison of logistic and linear regression for data with **extreme values**



**Logistic regression**



**Linear regression**

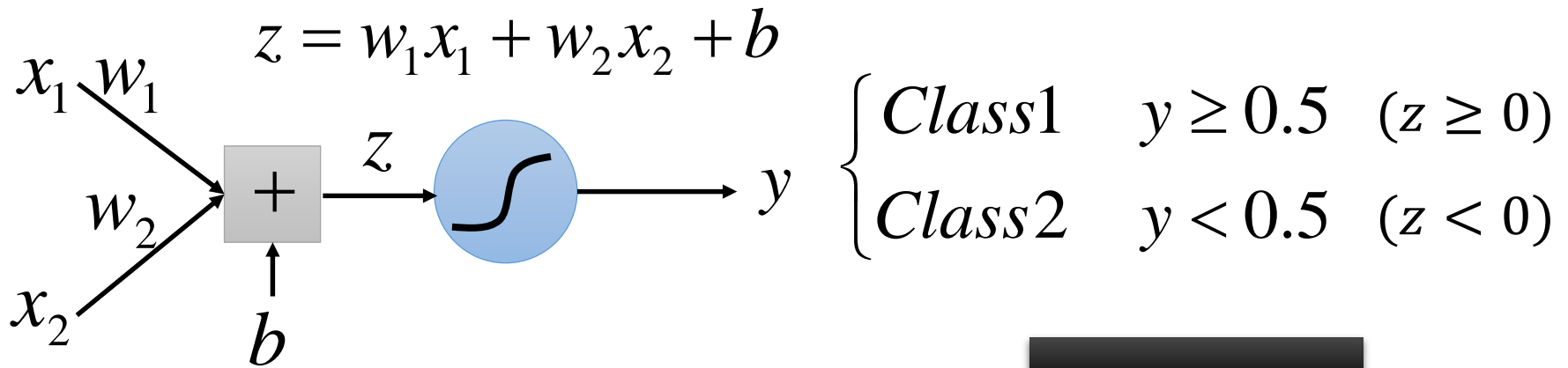
# Today's Topics

- Type of classifiers
- Logistic Regression
- Logistic Regression vs Linear Regression
- *Limitation of Logistic Regression*

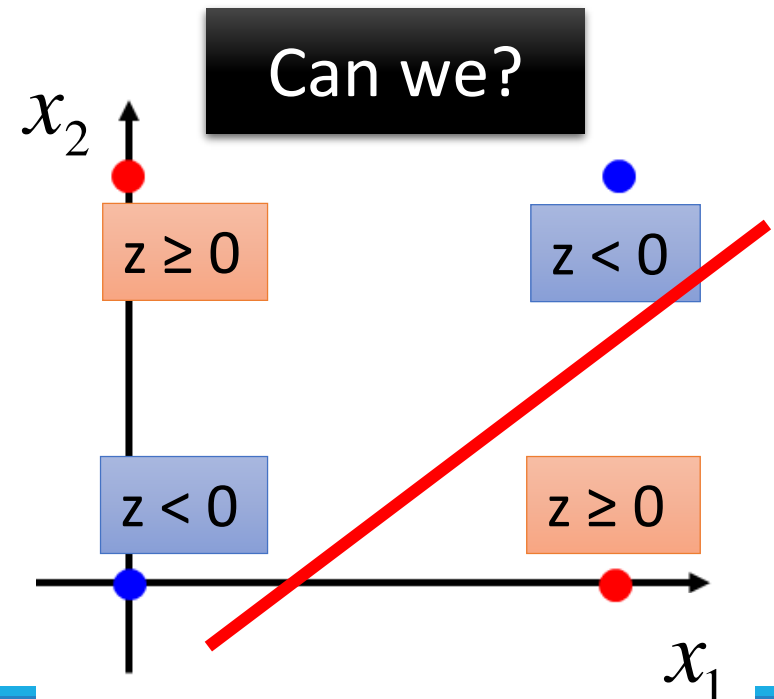
# Limitation of Logistic Regression

- Logistic Regression has some advantages
  - No prior assumptions about data distribution
  - Useful for tasks that require probabilities to make decision
  - Sigmoid is a derivable convex function of any order, and it is easy to find the optimal solution
- But there are situations where logistic regression is powerless

# Limitation of Logistic Regression



Input Feature		Label
$x_1$	$x_2$	
0	0	Class 2
0	1	Class 1
1	0	Class 1
1	1	Class 2

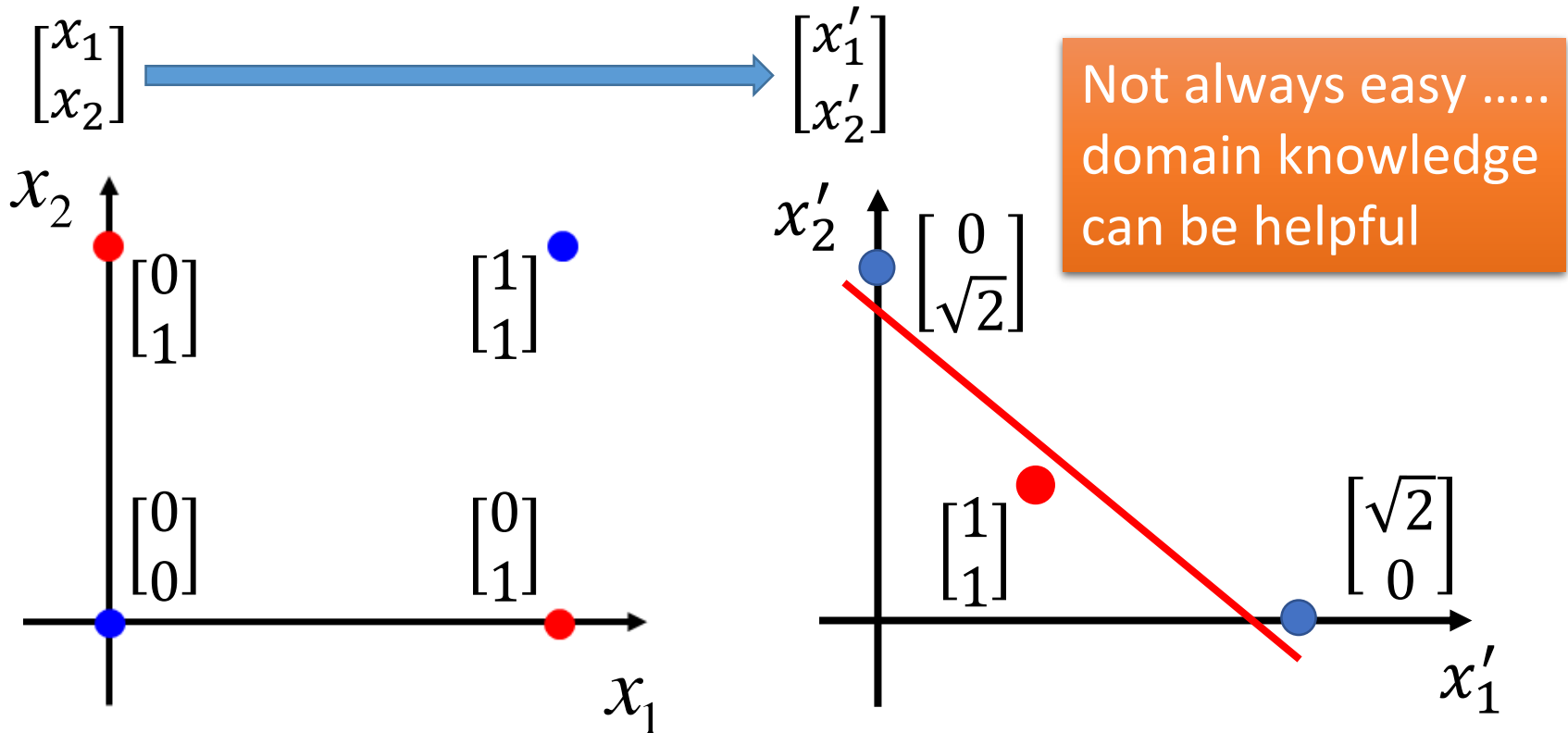


# Limitation of Logistic Regression

- Feature transformation

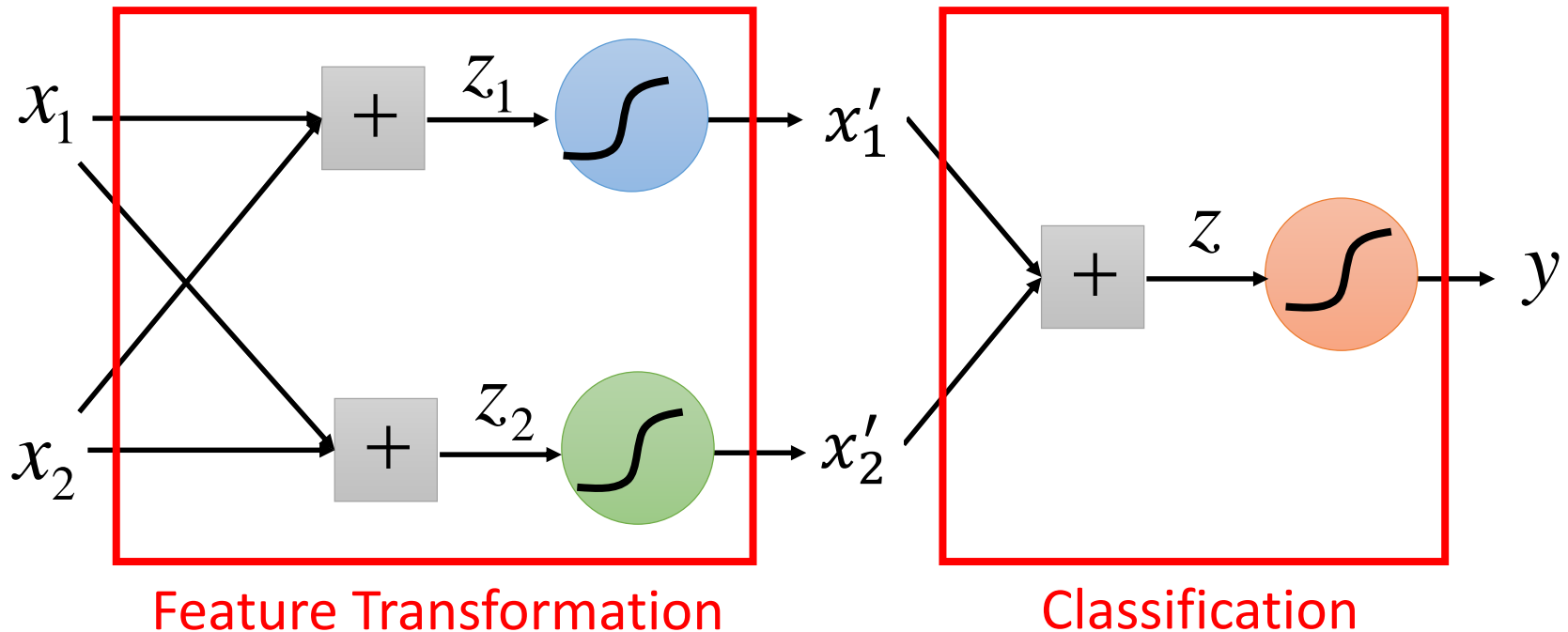
$x'_1$ : distance to  $\begin{bmatrix} 0 \\ 0 \end{bmatrix}$

$x'_2$ : distance to  $\begin{bmatrix} 1 \\ 1 \end{bmatrix}$



# Limitation of Logistic Regression

- Cascading logistic regression models



All the parameters of the logistic regressions are jointly learned.

(ignore bias in this figure)

# Summary

- **Logistic Regression**
  - Motivation
  - Sigmoid
  - model, loss, optimization
  - Difference with Linear Regression
  - Limitation



# Some questions...

- Usually we call logistic regression “逻辑回归”. Is this a reasonable name?
- Can you learn more about the structure?

