

# Classification: Probabilistic Generative Model

# Classification



- Credit Scoring
  - Input: income, savings, profession, age, past financial history .....
  - Output: accept or refuse
- Medical Diagnosis
  - Input: current symptoms, age, gender, past medical history .....
  - Output: which kind of diseases
- Handwritten character recognition
- Face recognition
  - Input: image of a face, output: person

Input:  output:  
金

# Example Application

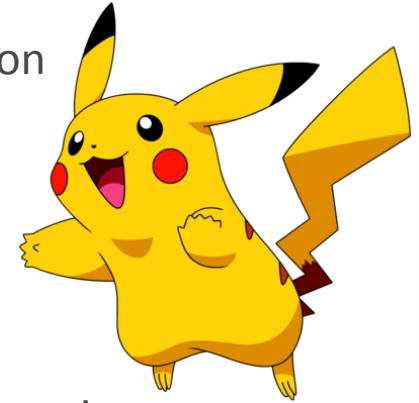


$$f(\text{Pikachu}) = \text{Electric}$$

$$f(\text{Squirtle}) = \text{Water}$$

$$f(\text{Bulbasaur}) = \text{Grass}$$

pokemon games (*NOT* pokemon cards or Pokemon Go)



# Example Application

- **Total:** sum of all stats that come after this, a general guide to how strong a pokemon is **320**
- **HP:** hit points, or health, defines how much damage a pokemon can withstand before fainting **35**
- **Attack:** the base modifier for normal attacks (eg. Scratch, Punch) **55**
- **Defense:** the base damage resistance against normal attacks **40**
- **SP Atk:** special attack, the base modifier for special attacks (e.g. fire blast, bubble beam) **50**
- **SP Def:** the base damage resistance against special attacks **50**
- **Speed:** determines which pokemon attacks first each round **90**

Can we predict the “type” of pokemon based on the information?

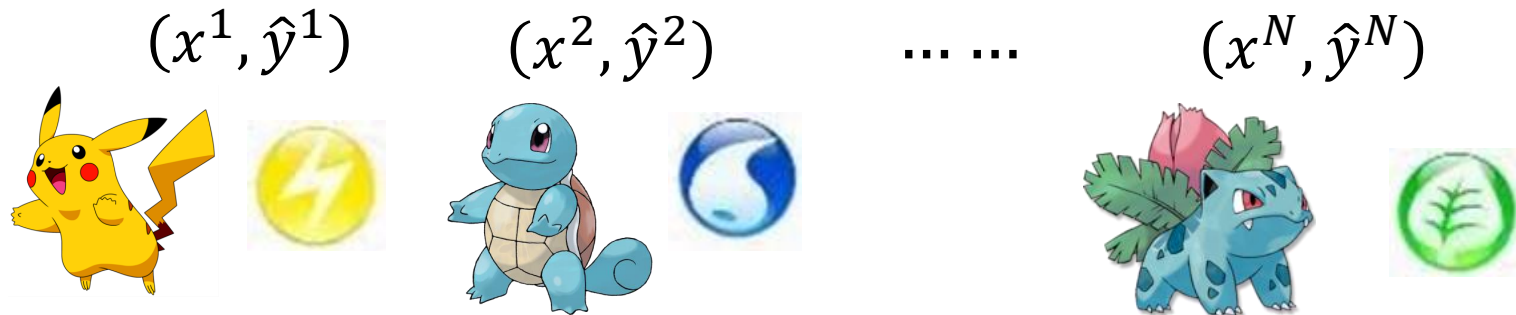
# Example Application

x	防禦方的屬性																		
	一般	格鬥	飛行	毒	地面	岩石	蟲	幽靈	鋼	火	水	草	電	超能力	冰	龍	惡	妖精	
一般	1x	1x	1x	1x	1x	1/2x	1x	0x	1/2x	1x	1x	1x	1x	1x	1x	1x	1x	1x	
格鬥	2x	1x	1/2x	1/2x	1x	2x	1/2x	0x	2x	1x	1x	1x	1x	1/2x	2x	1x	2x	1/2x	
飛行	1x	2x	1x	1x	1x	1/2x	2x	1x	1/2x	1x	1x	2x	1/2x	1x	1x	1x	1x	1x	
毒	1x	1x	1x	1/2x	1/2x	1/2x	1x	1/2x	0x	1x	1x	2x	1x	1x	1x	1x	1x	2x	
地面	1x	1x	0x	2x	1x	2x	1/2x	1x	2x	2x	1x	1/2x	2x	1x	1x	1x	1x	1x	
岩石	1x	1/2x	2x	1x	1/2x	1x	2x	1x	1/2x	2x	1x	1x	1x	1x	2x	1x	1x	1x	
蟲	1x	1/2x	1/2x	1/2x	1x	1x	1x	1/2x	1/2x	1/2x	1x	2x	1x	2x	1x	1x	2x	1/2x	
幽靈	0x	1x	1x	1x	1x	1x	1x	2x	1x	1x	1x	1x	1x	2x	1x	1x	1/2x	1x	
鋼	1x	1x	1x	1x	1x	2x	1x	1x	1/2x	1/2x	1/2x	1x	1/2x	1x	2x	1x	1x	2x	
火	1x	1x	1x	1x	1x	1/2x	2x	1x	2x	1/2x	1/2x	2x	1x	1x	2x	1/2x	1x	1x	
水	1x	1x	1x	1x	2x	2x	1x	1x	1x	2x	1/2x	1/2x	1x	1x	1x	1/2x	1x	1x	
草	1x	1x	1/2x	1/2x	2x	2x	1/2x	1x	1/2x	1/2x	2x	1/2x	1x	1x	1x	1/2x	1x	1x	
電	1x	1x	2x	1x	0x	1x	1x	1x	1x	1x	2x	1/2x	1/2x	1x	1x	1/2x	1x	1x	
超能力	1x	2x	1x	2x	1x	1x	1x	1x	1/2x	1x	1x	1x	1x	1/2x	1x	1x	0x	1x	
冰	1x	1x	2x	1x	2x	1x	1x	1x	1/2x	1/2x	1/2x	2x	1x	1x	1/2x	2x	1x	1x	
龍	1x	1x	1x	1x	1x	1x	1x	1x	1/2x	1x	1x	1x	1x	1x	1x	2x	1x	0x	
惡	1x	1/2x	1x	1x	1x	1x	1x	2x	1x	1x	1x	1x	1x	2x	1x	1x	1/2x	1/2x	
妖精	1x	2x	1x	1/2x	1x	1x	1x	1x	1/2x	1/2x	1x	1x	1x	1x	1x	2x	2x	1x	

這些倍數適用於XY及之後的遊戲。

# How to do Classification

- Training data for Classification

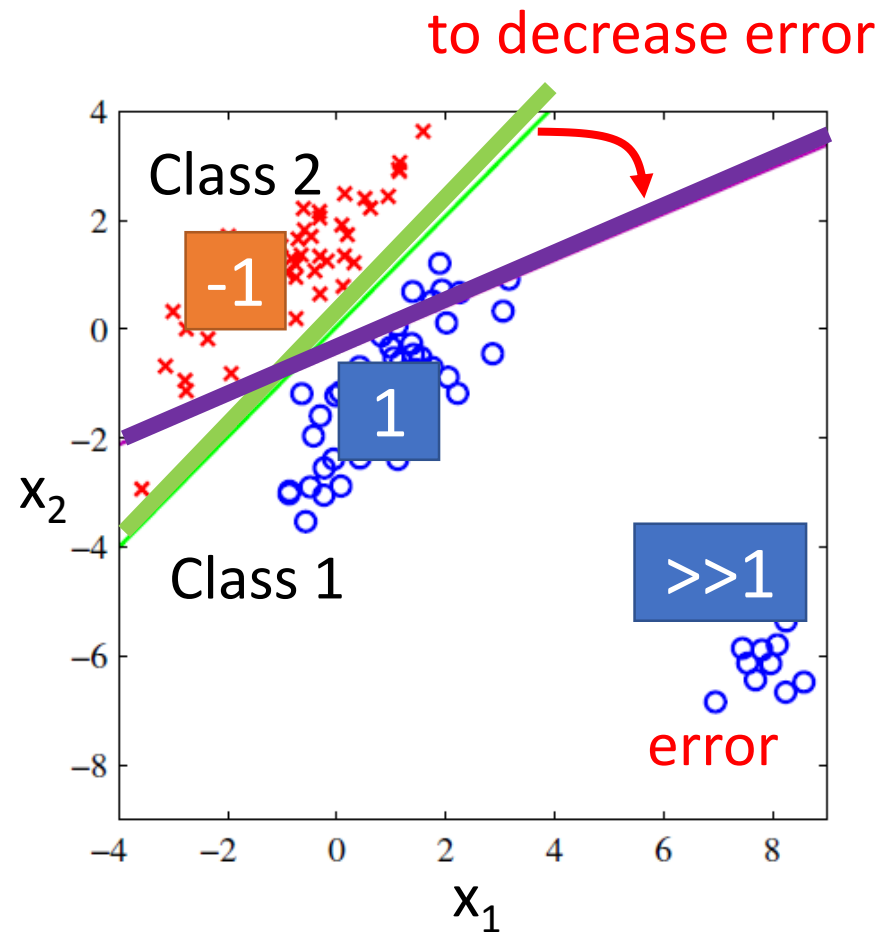
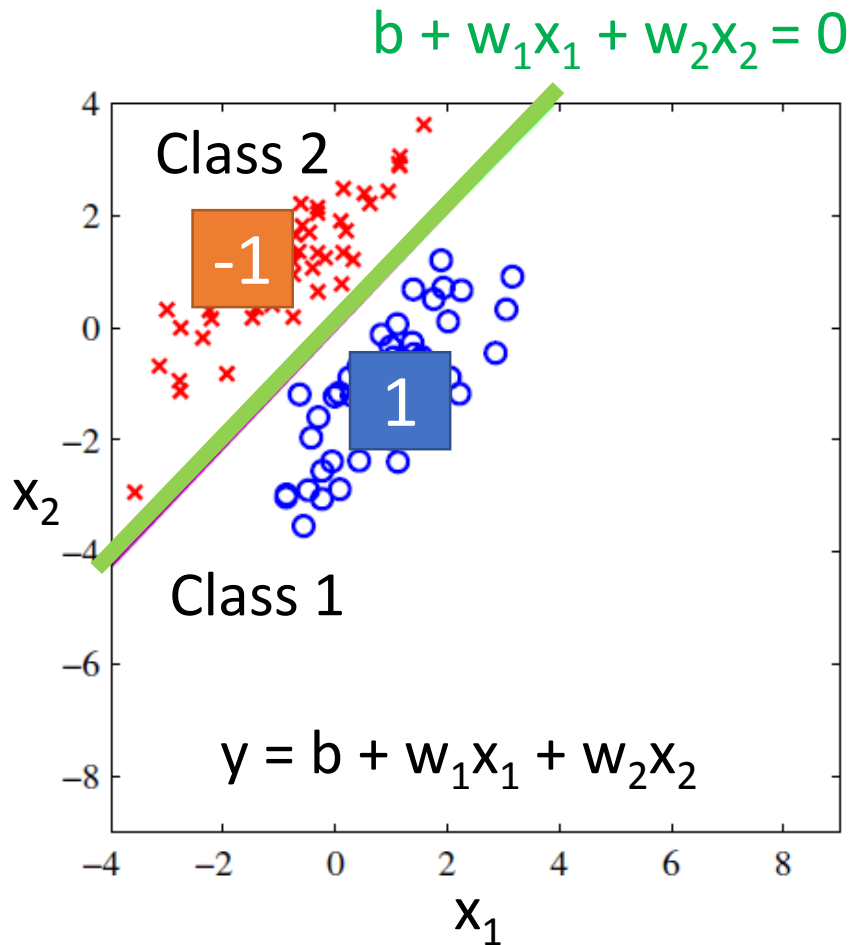


## Classification as Regression?

Binary classification as example

Training: Class 1 means the target is 1; Class 2 means the target is -1

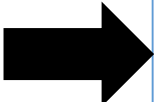
Testing: closer to 1 → class 1; closer to -1 → class 2



Penalize to the examples that are “too correct” ... (Bishop, P186)

- Multiple class: Class 1 means the target is 1; Class 2 means the target is 2; Class 3 means the target is 3 ..... problematic

# Ideal Alternatives

- Function (Model):  
 $x$    $f(x)$   

$g(x) > 0$	Output = class 1
<i>else</i>	Output = class 2

- Loss function:

$$L(f) = \sum_n \delta(f(x^n) \neq \hat{y}^n)$$

The number of times  $f$  get incorrect results on training data.

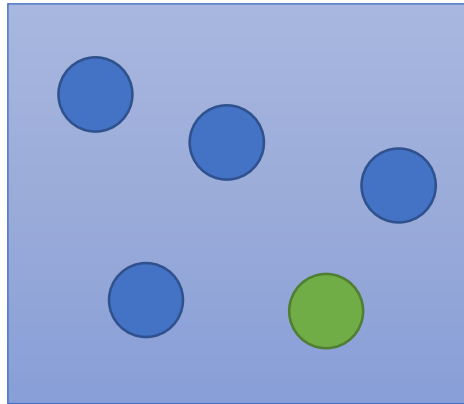
- Find the best function:
  - Example: Perceptron, SVM

Not Today



# Two Boxes

Box 1

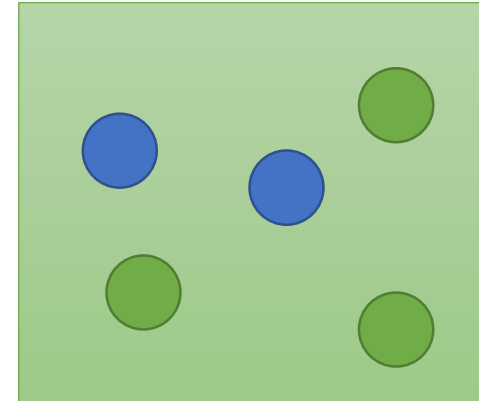


$$P(B_1) = 2/3$$

$$P(\text{Blue} | B_1) = 4/5$$

$$P(\text{Green} | B_1) = 1/5$$

Box 2



$$P(B_2) = 1/3$$

$$P(\text{Blue} | B_2) = 2/5$$

$$P(\text{Green} | B_2) = 3/5$$

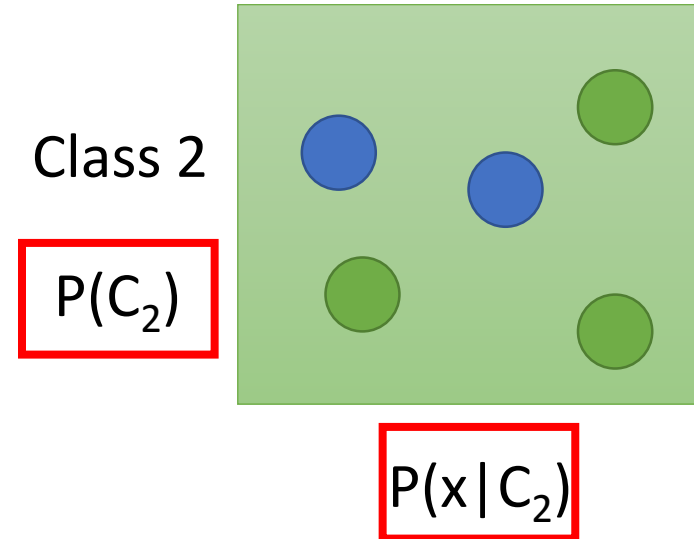
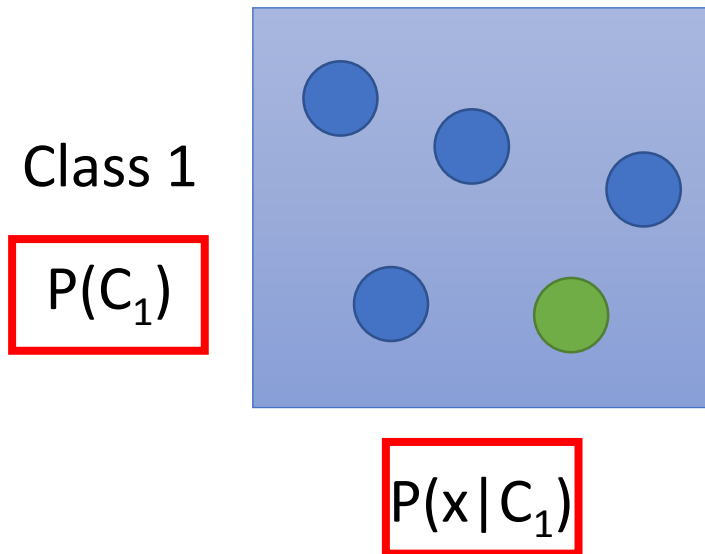
 from one of the boxes

Where does it come from?

$$P(B_1 | \text{Blue}) = \frac{P(\text{Blue} | B_1)P(B_1)}{P(\text{Blue} | B_1)P(B_1) + P(\text{Blue} | B_2)P(B_2)}$$

# Two Classes

Estimating the Probabilities  
From training data

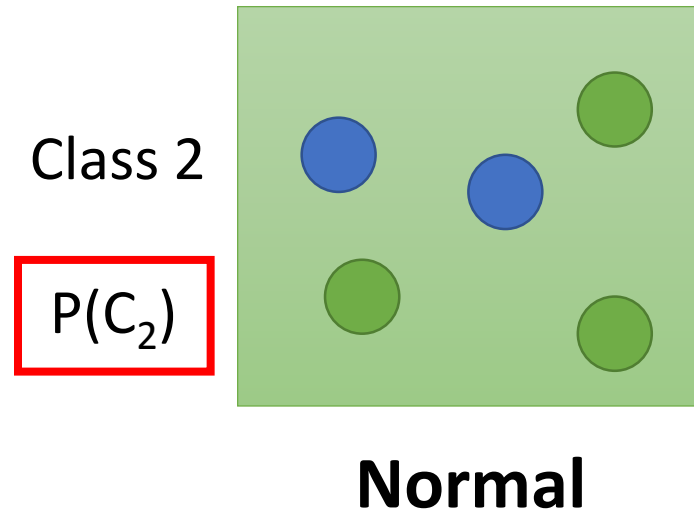
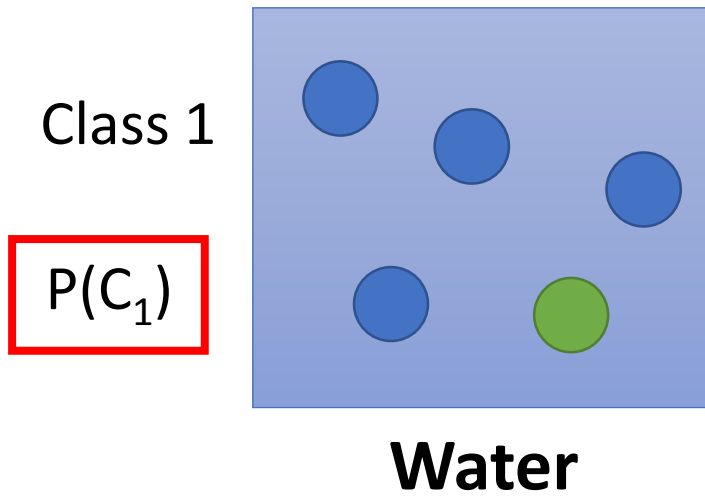


Given an  $x$ , which class does it belong to

$$P(C_1|x) = \frac{P(x|C_1)P(C_1)}{P(x|C_1)P(C_1) + P(x|C_2)P(C_2)}$$

Generative Model  $P(x) = P(x|C_1)P(C_1) + P(x|C_2)P(C_2)$

# Prior



Water and Normal type with ID < 400 for training,  
rest for testing

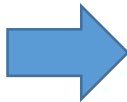
Training: 79 Water, 61 Normal

$$P(C_1) = 79 / (79 + 61) = 0.56$$

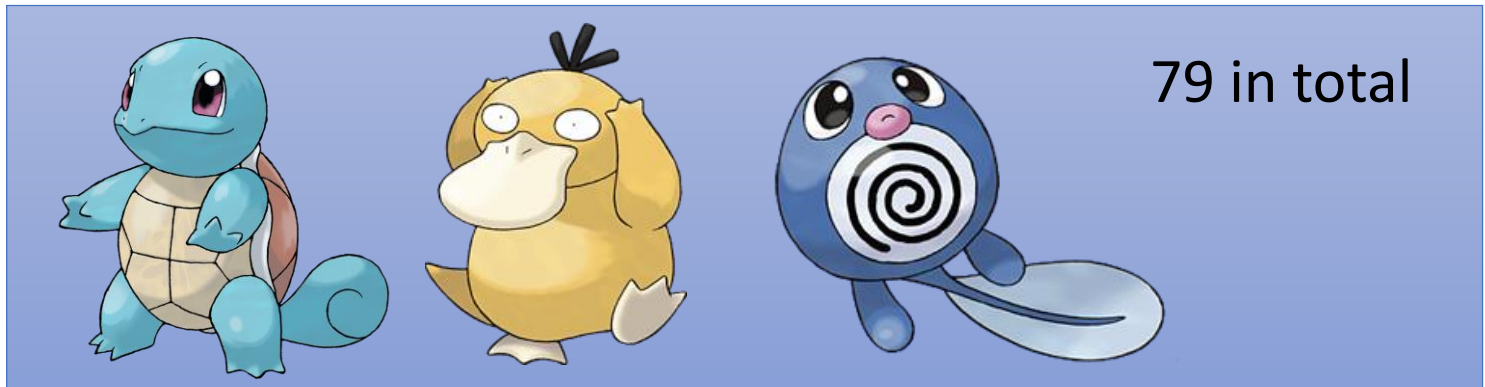
$$P(C_2) = 61 / (79 + 61) = 0.44$$

# Probability from Class

$$P(x|C_1) = ? \quad P(\text{  | \text{Water}) = ?$$

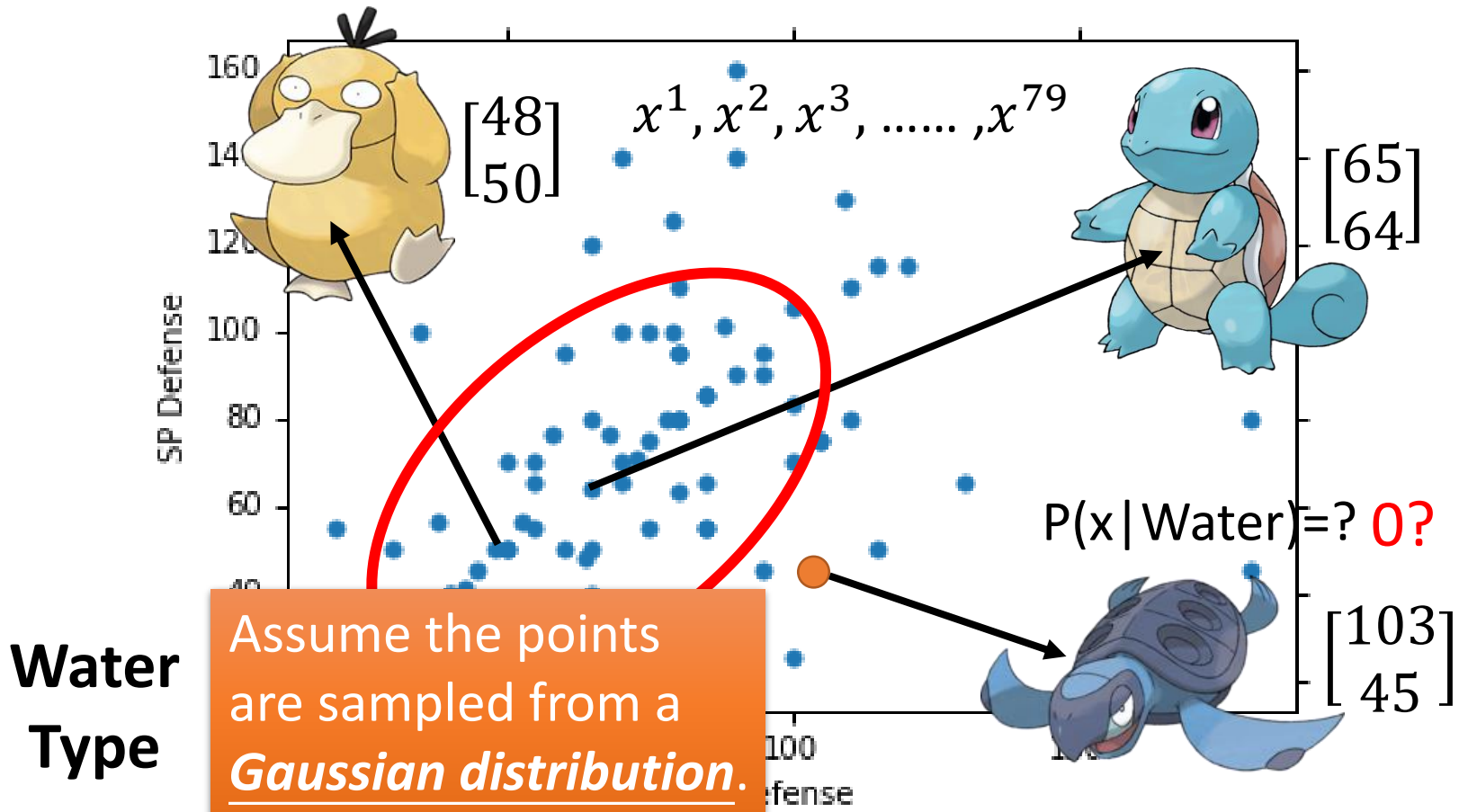
Each Pokémon is represented as a vector by its attribute.  feature

**Water  
Type**



# Probability from Class - Feature

- Considering **Defense** and **SP Defense**

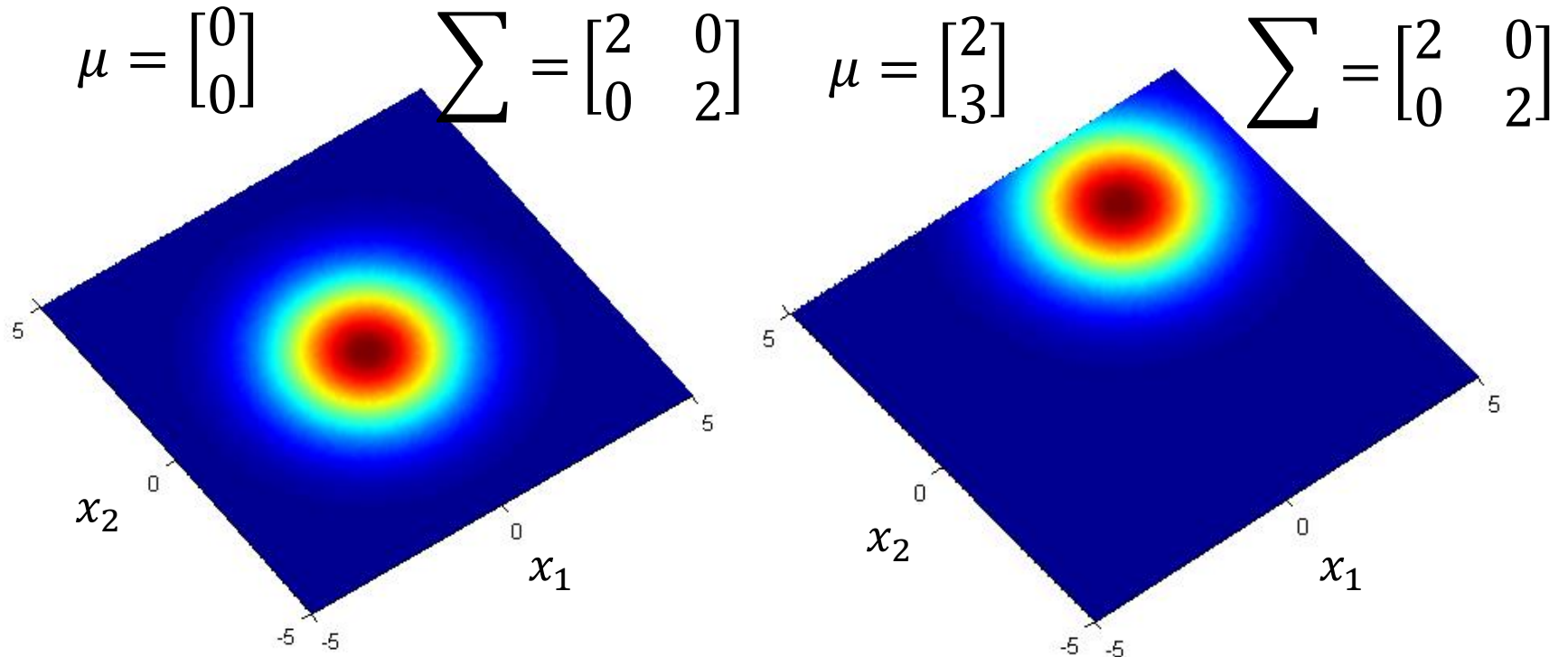


# Gaussian Distribution

$$f_{\mu, \Sigma}(x) = \frac{1}{(2\pi)^{D/2}} \frac{1}{|\Sigma|^{1/2}} \exp \left\{ -\frac{1}{2} (x - \mu)^T \Sigma^{-1} (x - \mu) \right\}$$

Input: vector  $x$ , output: probability of sampling  $x$

The shape of the function determines by **mean  $\mu$**  and **covariance matrix  $\Sigma$**

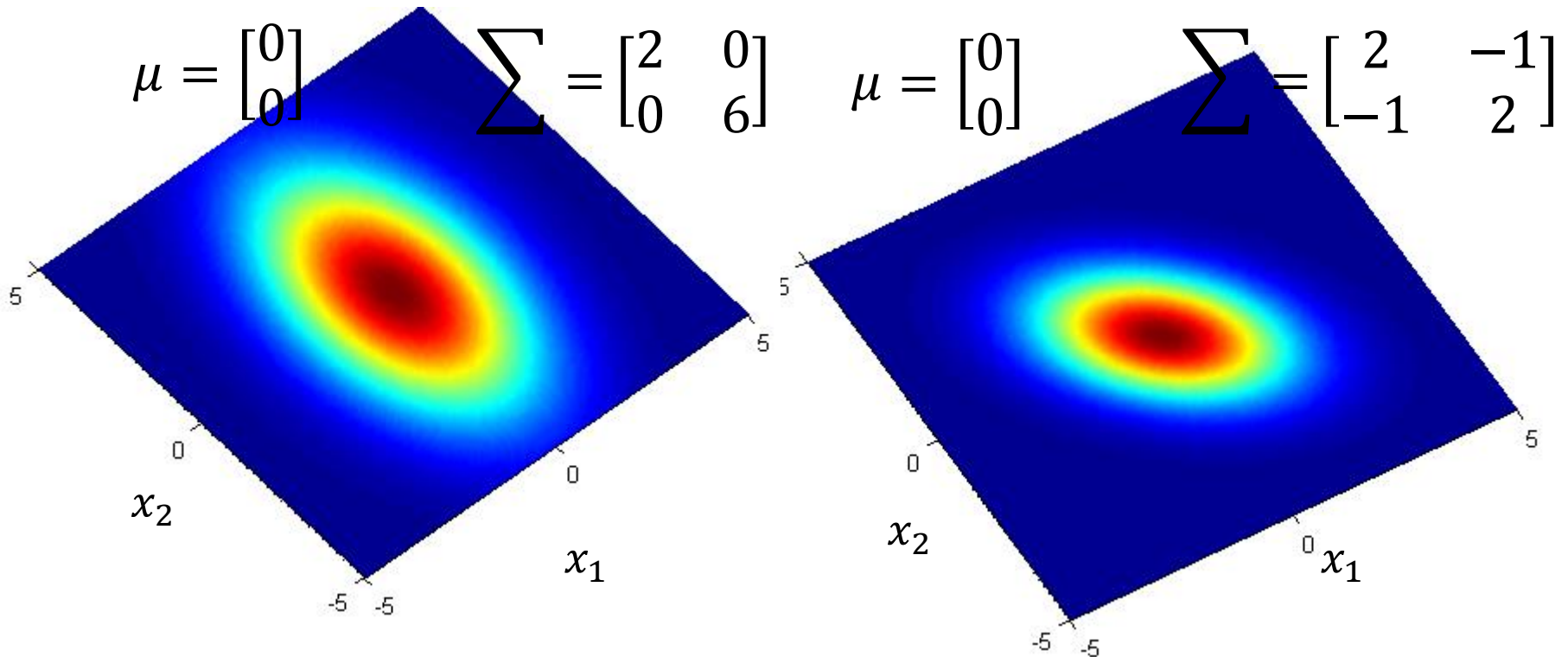


# Gaussian Distribution

$$f_{\mu,\Sigma}(x) = \frac{1}{(2\pi)^{D/2}} \frac{1}{|\Sigma|^{1/2}} \exp\left\{-\frac{1}{2}(x - \mu)^T \Sigma^{-1}(x - \mu)\right\}$$

Input: vector  $x$ , output: probability of sampling  $x$

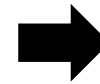
The shape of the function determines by **mean  $\mu$**  and **covariance matrix  $\Sigma$**



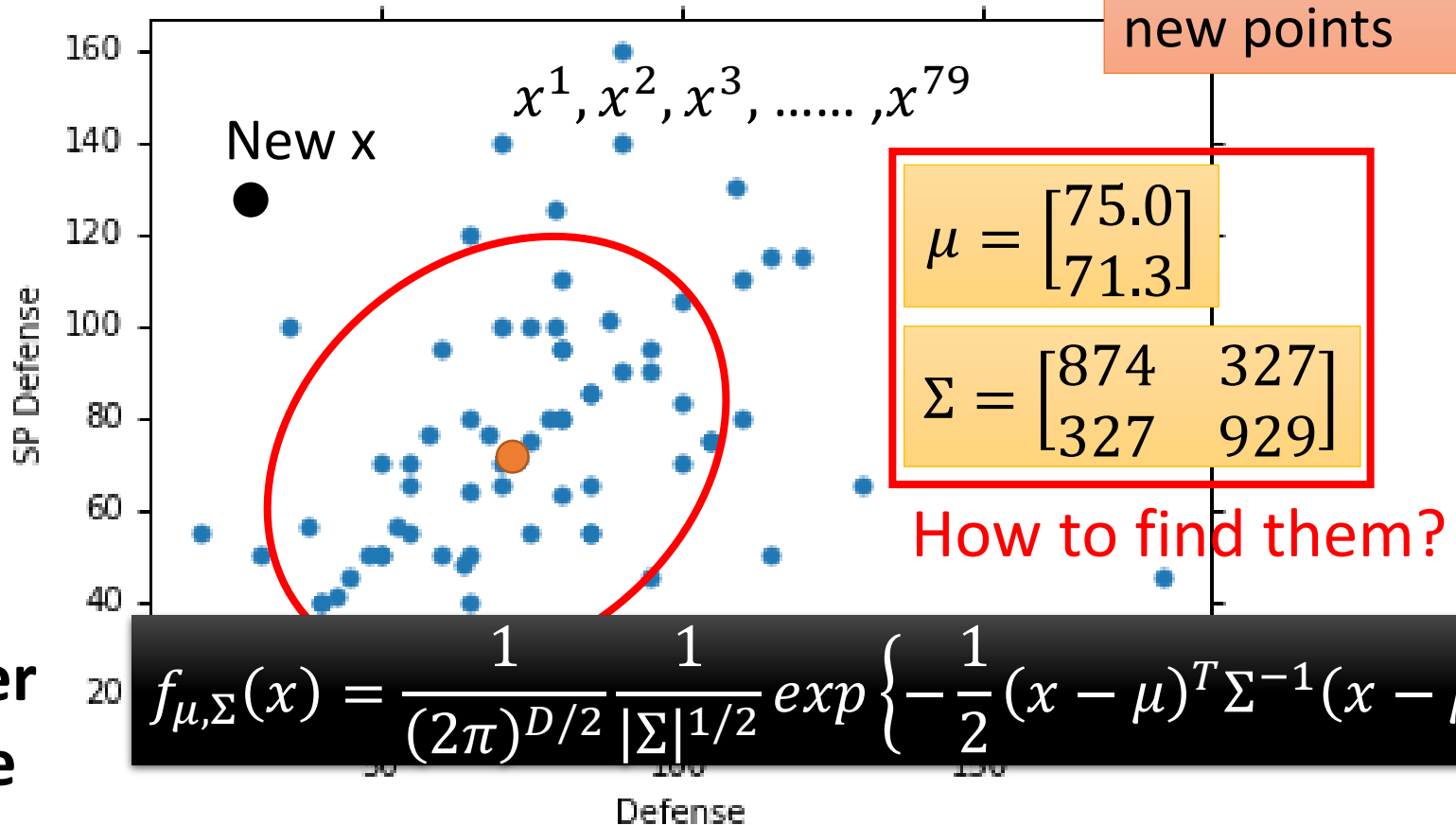
# Probability from Class

Assume the points are sampled from a Gaussian distribution

Find the Gaussian distribution behind them

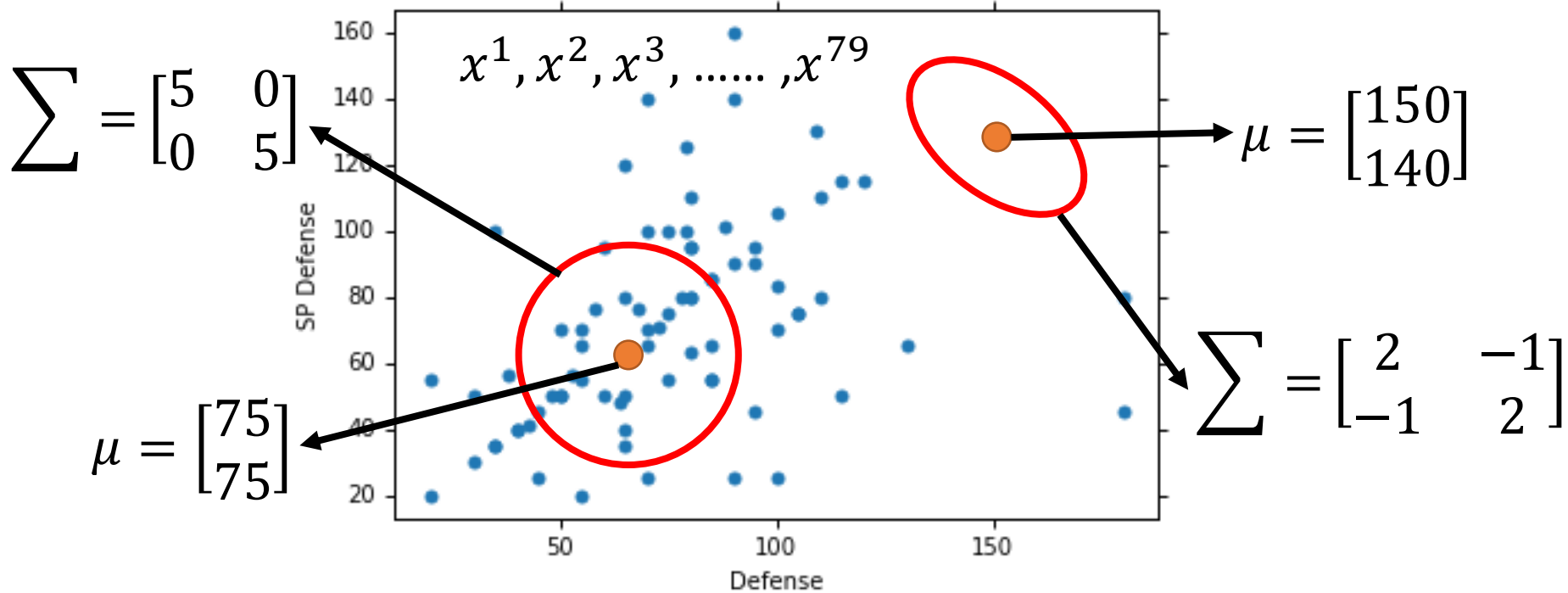


Probability for new points





**Maximum Likelihood**  $f_{\mu, \Sigma}(x) = \frac{1}{(2\pi)^{D/2}} \frac{1}{|\Sigma|^{1/2}} \exp \left\{ -\frac{1}{2} (x - \mu)^T \Sigma^{-1} (x - \mu) \right\}$



The Gaussian with any mean  $\mu$  and covariance matrix  $\Sigma$  can generate these points. ➡ Different Likelihood

Likelihood of a Gaussian with mean  $\mu$  and covariance matrix  $\Sigma$   
 = the probability of the Gaussian samples  $x^1, x^2, x^3, \dots, x^{79}$

$$L(\mu, \Sigma) = f_{\mu, \Sigma}(x^1) f_{\mu, \Sigma}(x^2) f_{\mu, \Sigma}(x^3) \dots f_{\mu, \Sigma}(x^{79})$$

# Maximum Likelihood

We have the “Water” type Pokémons:  $x^1, x^2, x^3, \dots, x^{79}$

We assume  $x^1, x^2, x^3, \dots, x^{79}$  generate from the Gaussian  $(\mu^*, \Sigma^*)$  with the **maximum likelihood**

$$L(\mu, \Sigma) = f_{\mu, \Sigma}(x^1) f_{\mu, \Sigma}(x^2) f_{\mu, \Sigma}(x^3) \dots f_{\mu, \Sigma}(x^{79})$$

$$f_{\mu, \Sigma}(x) = \frac{1}{(2\pi)^{D/2}} \frac{1}{|\Sigma|^{1/2}} \exp\left\{-\frac{1}{2}(x - \mu)^T \Sigma^{-1}(x - \mu)\right\}$$

$$\mu^*, \Sigma^* = \arg \max_{\mu, \Sigma} L(\mu, \Sigma)$$

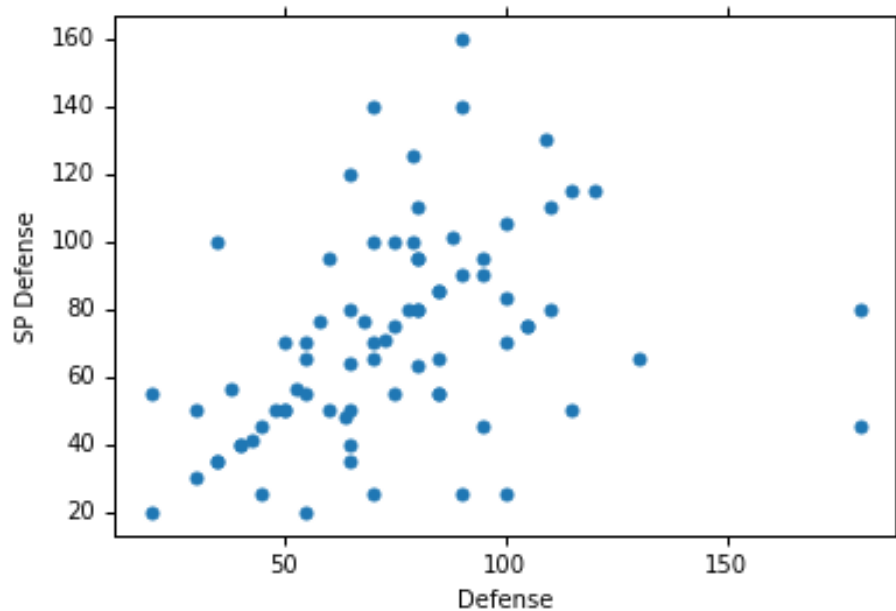
$$\mu^* = \frac{1}{79} \sum_{n=1}^{79} x^n$$

average

$$\Sigma^* = \frac{1}{79} \sum_{n=1}^{79} (x^n - \mu^*) (x^n - \mu^*)^T$$

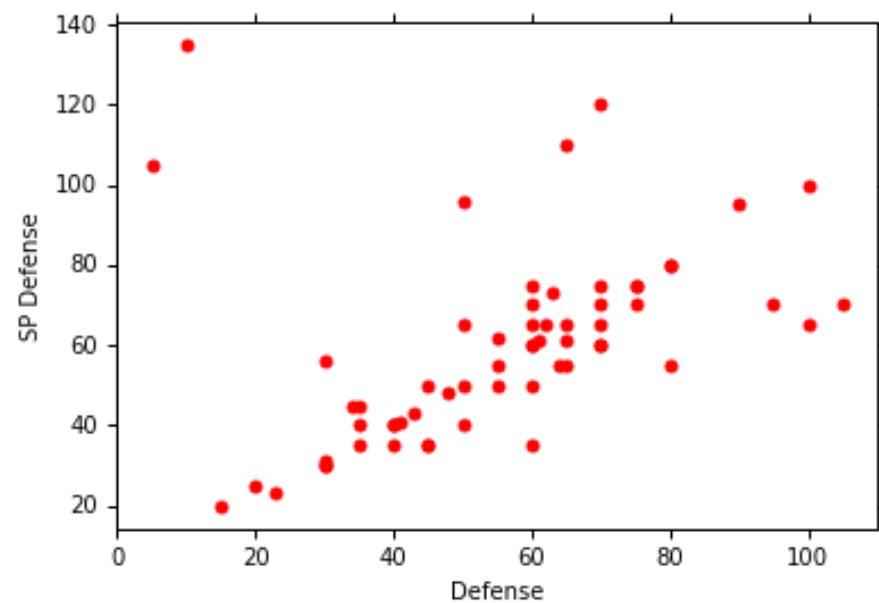
# Maximum Likelihood

## Class 1: Water



$$\mu^1 = \begin{bmatrix} 75.0 \\ 71.3 \end{bmatrix} \quad \Sigma^1 = \begin{bmatrix} 874 & 327 \\ 327 & 929 \end{bmatrix}$$

## Class 2: Normal



$$\mu^2 = \begin{bmatrix} 55.6 \\ 59.8 \end{bmatrix} \quad \Sigma^2 = \begin{bmatrix} 847 & 422 \\ 422 & 685 \end{bmatrix}$$

Now we can do classification 😊

$$f_{\mu^1, \Sigma^1}(x) = \frac{1}{(2\pi)^{D/2}} \frac{1}{|\Sigma^1|^{1/2}} \exp\left\{-\frac{1}{2}(x - \mu^1)^T (\Sigma^1)^{-1} (x - \mu^1)\right\}$$

$P(C_1)$   
 $= 79 / (79 + 61) = 0.56$

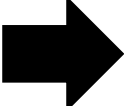
$$\mu^1 = \begin{bmatrix} 75.0 \\ 71.3 \end{bmatrix} \quad \Sigma^1 = \begin{bmatrix} 874 & 327 \\ 327 & 929 \end{bmatrix}$$

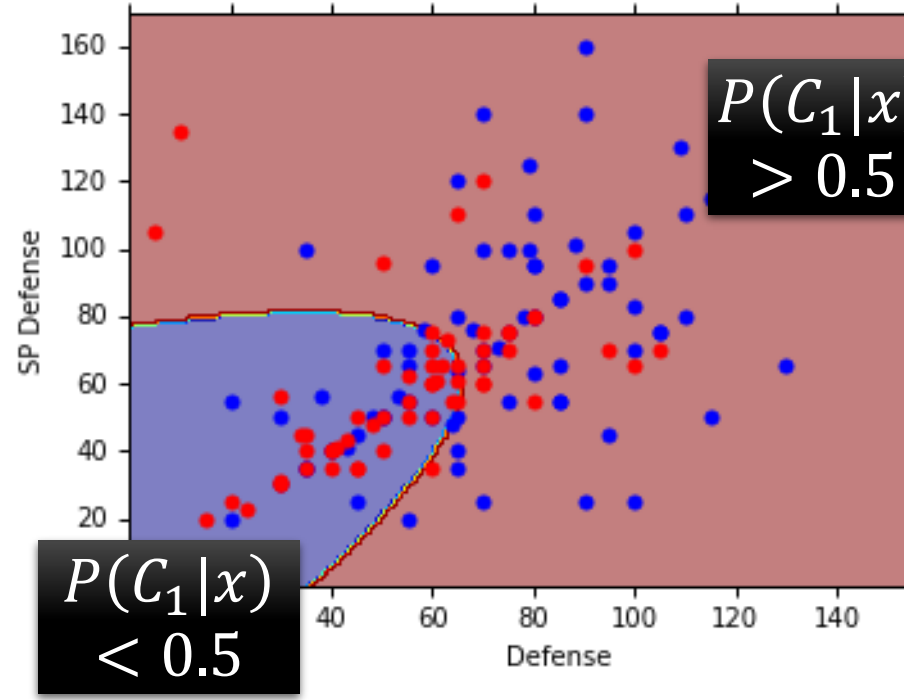
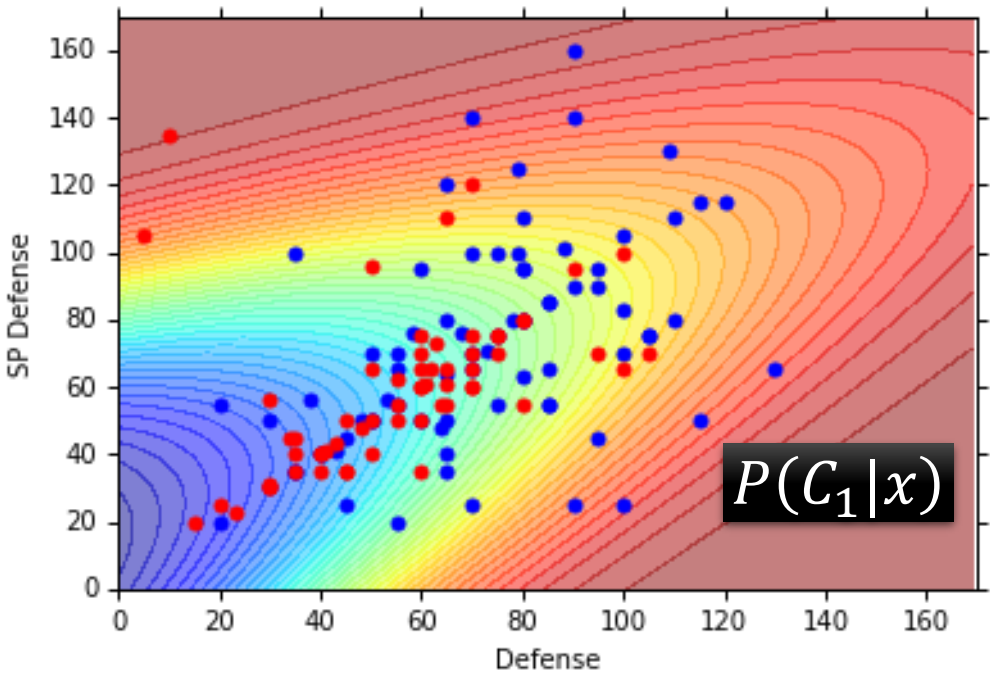
$$P(C_1|x) = \frac{P(x|C_1)P(C_1)}{P(x|C_1)P(C_1) + P(x|C_2)P(C_2)}$$

$$f_{\mu^2, \Sigma^2}(x) = \frac{1}{(2\pi)^{D/2}} \frac{1}{|\Sigma^2|^{1/2}} \exp\left\{-\frac{1}{2}(x - \mu^2)^T (\Sigma^2)^{-1} (x - \mu^2)\right\}$$

$P(C_2)$   
 $= 61 / (79 + 61)$   
 $= 0.44$

$$\mu^2 = \begin{bmatrix} 55.6 \\ 59.8 \end{bmatrix} \quad \Sigma^2 = \begin{bmatrix} 847 & 422 \\ 422 & 685 \end{bmatrix}$$

If  $P(C_1|x) > 0.5$   x belongs to class 1 (Water)



Blue points:  $C_1$  (Water), Red points:  $C_2$  (Normal)

How's the results?

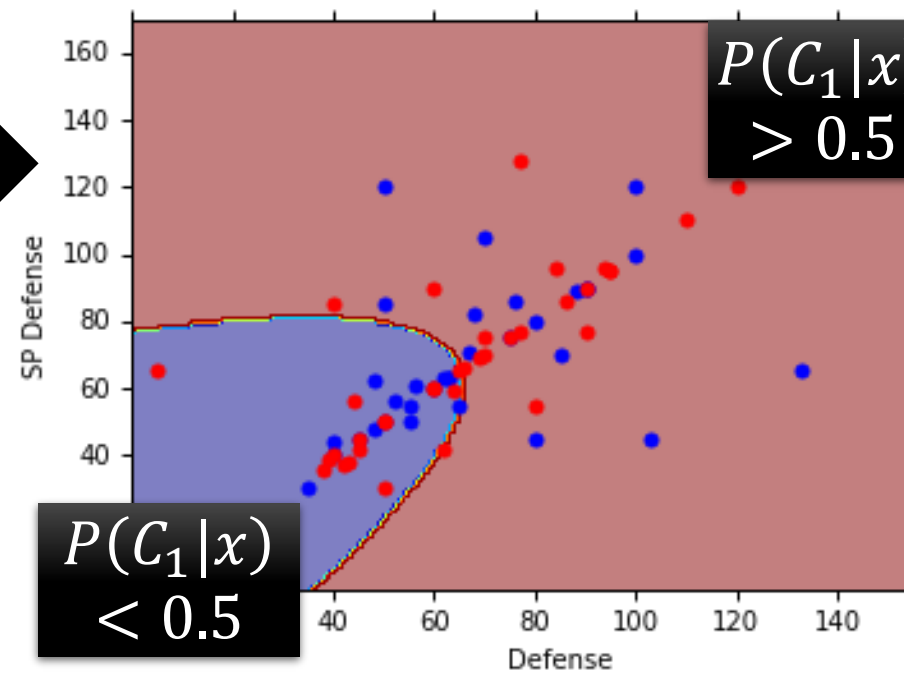
Testing data: 47% accuracy  $\rightarrow$

All: total, hp, att, sp att, de, sp de, speed (7 features)

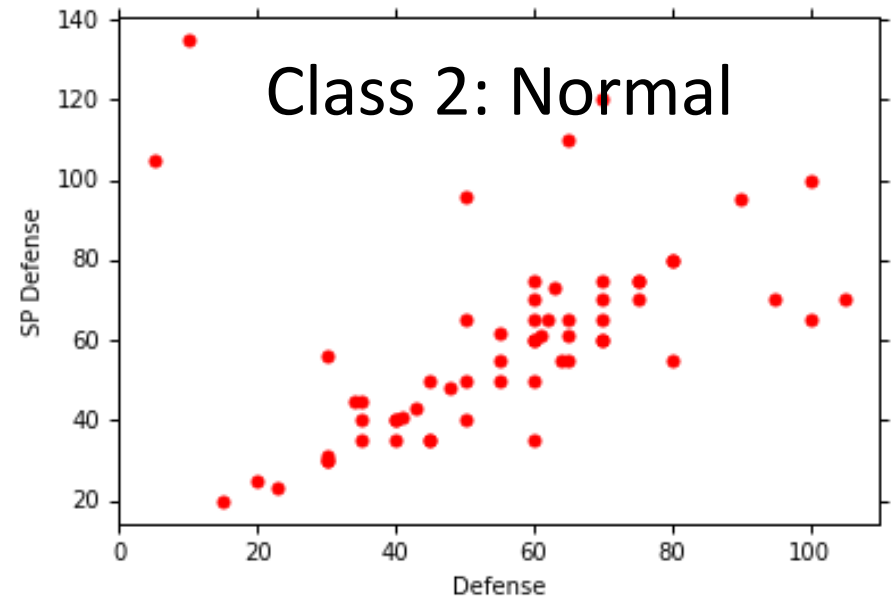
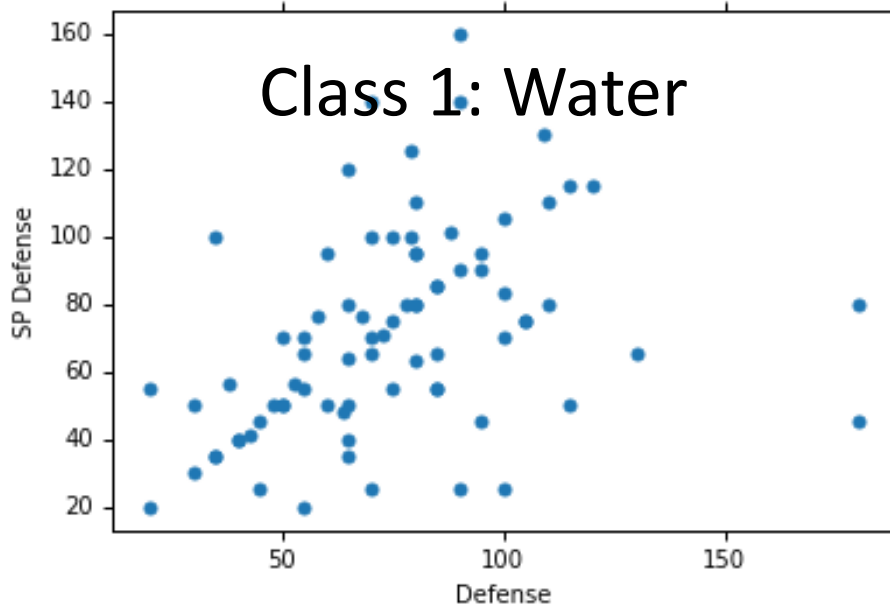
$\mu^1, \mu^2$ : 7-dim vector

$\Sigma^1, \Sigma^2$ : 7 x 7 matrices

54% accuracy ... ☹️



# Modifying Model



$$\mu^1 = \begin{bmatrix} 75.0 \\ 71.3 \end{bmatrix} \quad \Sigma^1 = \begin{bmatrix} 874 & 327 \\ 327 & 929 \end{bmatrix}$$

$$\mu^2 = \begin{bmatrix} 55.6 \\ 59.8 \end{bmatrix} \quad \Sigma^2 = \begin{bmatrix} 847 & 422 \\ 422 & 685 \end{bmatrix}$$

The same  $\Sigma$

Less parameters

# Modifying Model

Ref: Bishop,  
chapter 4.2.2

- Maximum likelihood

“Water” type Pokémons:

$x^1, x^2, x^3, \dots, x^{79}$

$\mu^1$

“Normal” type Pokémons:

$x^{80}, x^{81}, x^{82}, \dots, x^{140}$

$\mu^2$

$\Sigma$

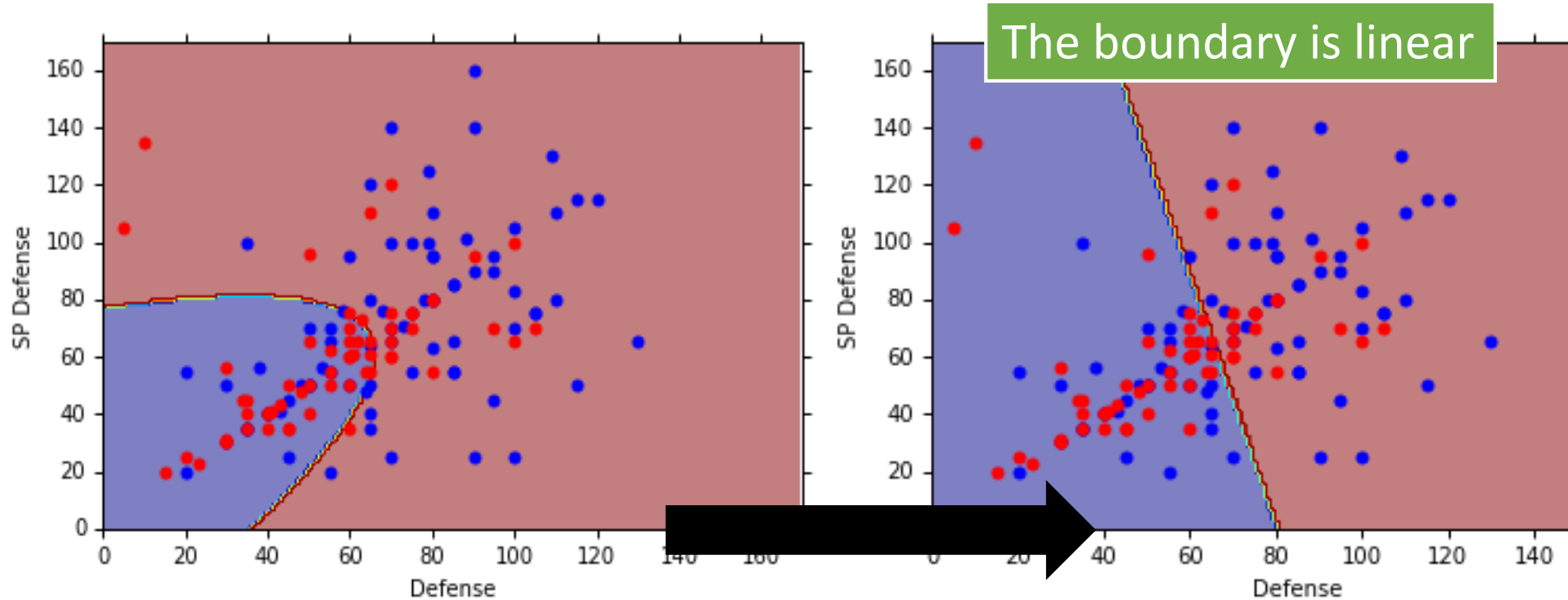
Find  $\mu^1, \mu^2, \Sigma$  maximizing the likelihood  $L(\mu^1, \mu^2, \Sigma)$

$$L(\mu^1, \mu^2, \Sigma) = f_{\mu^1, \Sigma}(x^1) f_{\mu^1, \Sigma}(x^2) \cdots f_{\mu^1, \Sigma}(x^{79}) \\ \times f_{\mu^2, \Sigma}(x^{80}) f_{\mu^2, \Sigma}(x^{81}) \cdots f_{\mu^2, \Sigma}(x^{140})$$

$\mu^1$  and  $\mu^2$  is the same

$$\Sigma = \frac{79}{140} \Sigma^1 + \frac{61}{140} \Sigma^2$$

# Modifying Model



The same covariance matrix

All: total, hp, att, sp att, de, sp de, speed

54% accuracy  $\longrightarrow$  73% accuracy



# Three Steps

- Function Set (Model):

$x$  

$$P(C_1|x) = \frac{P(x|C_1)P(C_1)}{P(x|C_1)P(C_1) + P(x|C_2)P(C_2)}$$

If  $P(C_1|x) > 0.5$ , output: class 1  
Otherwise, output: class 2

- Goodness of a function:
  - The mean  $\mu$  and covariance  $\Sigma$  that maximizing the likelihood (the probability of generating data)
- Find the best function: easy

# Probability Distribution

- You can always use the distribution you like 😊

$$P(x|C_1) = P(x_1|C_1) P(x_2|C_1) \cdots P(x_k|C_1) \cdots$$

$$\begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_k \\ \vdots \\ x_K \end{bmatrix}$$

1-D Gaussian

For binary features, you may assume they are from Bernoulli distributions.

If you assume all the dimensions are independent, then you are using *Naive Bayes Classifier*.

# Posterior Probability

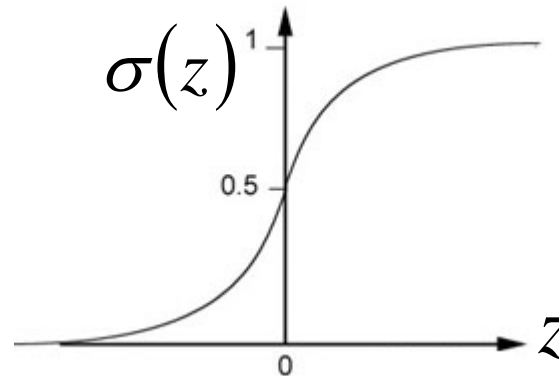
$$P(C_1|x) = \frac{P(x|C_1)P(C_1)}{P(x|C_1)P(C_1) + P(x|C_2)P(C_2)}$$

$$= \frac{1}{1 + \frac{P(x|C_2)P(C_2)}{P(x|C_1)P(C_1)}}$$

$$= \frac{1}{1 + \exp(-z)}$$

$= \sigma(z)$   
Sigmoid  
function

$$z = \ln \frac{P(x|C_1)P(C_1)}{P(x|C_2)P(C_2)}$$



Warning of Math

# Posterior Probability

$$P(C_1|x) = \sigma(z) \quad \text{sigmoid} \quad z = \ln \frac{P(x|C_1)P(C_1)}{P(x|C_2)P(C_2)}$$

$$z = \ln \frac{P(x|C_1)}{P(x|C_2)} + \ln \frac{P(C_1)}{P(C_2)} \rightarrow \frac{\frac{N_1}{N_1 + N_2}}{\frac{N_2}{N_1 + N_2}} = \frac{N_1}{N_2}$$

$$P(x|C_1) = \frac{1}{(2\pi)^{D/2}} \frac{1}{|\Sigma^1|^{1/2}} \exp \left\{ -\frac{1}{2} (x - \mu^1)^T (\Sigma^1)^{-1} (x - \mu^1) \right\}$$

$$P(x|C_2) = \frac{1}{(2\pi)^{D/2}} \frac{1}{|\Sigma^2|^{1/2}} \exp \left\{ -\frac{1}{2} (x - \mu^2)^T (\Sigma^2)^{-1} (x - \mu^2) \right\}$$

$$z = \ln \frac{P(x|C_1)}{P(x|C_2)} + \ln \frac{P(C_1)}{P(C_2)} = \frac{N_1}{N_2}$$

$$P(x|C_1) = \frac{1}{(2\pi)^{D/2}} \frac{1}{|\Sigma^1|^{1/2}} \exp \left\{ -\frac{1}{2} (x - \mu^1)^T (\Sigma^1)^{-1} (x - \mu^1) \right\}$$

$$P(x|C_2) = \frac{1}{(2\pi)^{D/2}} \frac{1}{|\Sigma^2|^{1/2}} \exp \left\{ -\frac{1}{2} (x - \mu^2)^T (\Sigma^2)^{-1} (x - \mu^2) \right\}$$

$$\ln \frac{\frac{1}{(2\pi)^{D/2}} \frac{1}{|\Sigma^1|^{1/2}} \exp \left\{ -\frac{1}{2} (x - \mu^1)^T (\Sigma^1)^{-1} (x - \mu^1) \right\}}{\frac{1}{(2\pi)^{D/2}} \frac{1}{|\Sigma^2|^{1/2}} \exp \left\{ -\frac{1}{2} (x - \mu^2)^T (\Sigma^2)^{-1} (x - \mu^2) \right\}}$$

$$= \ln \frac{|\Sigma^2|^{1/2}}{|\Sigma^1|^{1/2}} \exp \left\{ -\frac{1}{2} [(x - \mu^1)^T (\Sigma^1)^{-1} (x - \mu^1) - (x - \mu^2)^T (\Sigma^2)^{-1} (x - \mu^2)] \right\}$$

$$= \ln \frac{|\Sigma^2|^{1/2}}{|\Sigma^1|^{1/2}} - \frac{1}{2} [(x - \mu^1)^T (\Sigma^1)^{-1} (x - \mu^1) - (x - \mu^2)^T (\Sigma^2)^{-1} (x - \mu^2)]$$

$$z = \ln \frac{P(x|C_1)}{P(x|C_2)} + \ln \frac{P(C_1)}{P(C_2)} = \frac{N_1}{N_2}$$

$$= \ln \frac{|\Sigma^2|^{1/2}}{|\Sigma^1|^{1/2}} - \frac{1}{2} [(x - \mu^1)^T (\Sigma^1)^{-1} (x - \mu^1) - (x - \mu^2)^T (\Sigma^2)^{-1} (x - \mu^2)]$$

$$(x - \mu^1)^T (\Sigma^1)^{-1} (x - \mu^1)$$

$$= x^T (\Sigma^1)^{-1} x - x^T (\Sigma^1)^{-1} \mu^1 - (\mu^1)^T (\Sigma^1)^{-1} x + (\mu^1)^T (\Sigma^1)^{-1} \mu^1$$

$$= x^T (\Sigma^1)^{-1} x - 2(\mu^1)^T (\Sigma^1)^{-1} x + (\mu^1)^T (\Sigma^1)^{-1} \mu^1$$

$$(x - \mu^2)^T (\Sigma^2)^{-1} (x - \mu^2)$$

$$= x^T (\Sigma^2)^{-1} x - 2(\mu^2)^T (\Sigma^2)^{-1} x + (\mu^2)^T (\Sigma^2)^{-1} \mu^2$$

$$z = \ln \frac{|\Sigma^2|^{1/2}}{|\Sigma^1|^{1/2}} - \frac{1}{2} x^T (\Sigma^1)^{-1} x + (\mu^1)^T (\Sigma^1)^{-1} x - \frac{1}{2} (\mu^1)^T (\Sigma^1)^{-1} \mu^1$$

$$+ \frac{1}{2} x^T (\Sigma^2)^{-1} x - (\mu^2)^T (\Sigma^2)^{-1} x + \frac{1}{2} (\mu^2)^T (\Sigma^2)^{-1} \mu^2 + \ln \frac{N_1}{N_2}$$

End of Warning



$$P(C_1|x) = \sigma(z)$$

$$z = \ln \frac{|\Sigma^2|^{1/2}}{|\Sigma^1|^{1/2}} - \frac{1}{2} x^T (\Sigma^1)^{-1} x + (\mu^1)^T (\Sigma^1)^{-1} x - \frac{1}{2} (\mu^1)^T (\Sigma^1)^{-1} \mu^1$$

$$+ \frac{1}{2} x^T (\Sigma^2)^{-1} x - (\mu^2)^T (\Sigma^2)^{-1} x + \frac{1}{2} (\mu^2)^T (\Sigma^2)^{-1} \mu^2 + \ln \frac{N_1}{N_2}$$

$$\Sigma_1 = \Sigma_2 = \Sigma$$

$$z = \underbrace{(\mu^1 - \mu^2)^T \Sigma^{-1} x}_{\mathbf{w}^T} - \frac{1}{2} (\mu^1)^T \Sigma^{-1} \mu^1 + \frac{1}{2} (\mu^2)^T \Sigma^{-1} \mu^2 + \ln \frac{N_1}{N_2}$$

$$\mathbf{b}$$

$$P(C_1|x) = \sigma(\mathbf{w} \cdot x + b) \quad \text{How about directly find } \mathbf{w} \text{ and } b?$$

In generative model, we estimate  $N_1, N_2, \mu^1, \mu^2, \Sigma$

Then we have  $\mathbf{w}$  and  $b$

# Reference

- Bishop: Chapter 4.1 – 4.2
- Data: <https://www.kaggle.com/abcsds/pokemon>
- Useful posts:
  - <https://www.kaggle.com/nishantbhadauria/d/abcsds/pokemon/pokemon-speed-attack-hp-defense-analysis-by-type>
  - <https://www.kaggle.com/nikos90/d/abcsds/pokemon/mastering-pokebars/discussion>
  - <https://www.kaggle.com/ndrewgele/d/abcsds/pokemon/visualizing-pok-mon-stats-with-seaborn/discussion>

# Acknowledgment

- 感謝 江貫榮 同學發現課程網頁上的日期錯誤
- 感謝 范廷瀚 同學提供寶可夢的 domain knowledge
- 感謝 Victor Chen 發現投影片上的打字錯誤