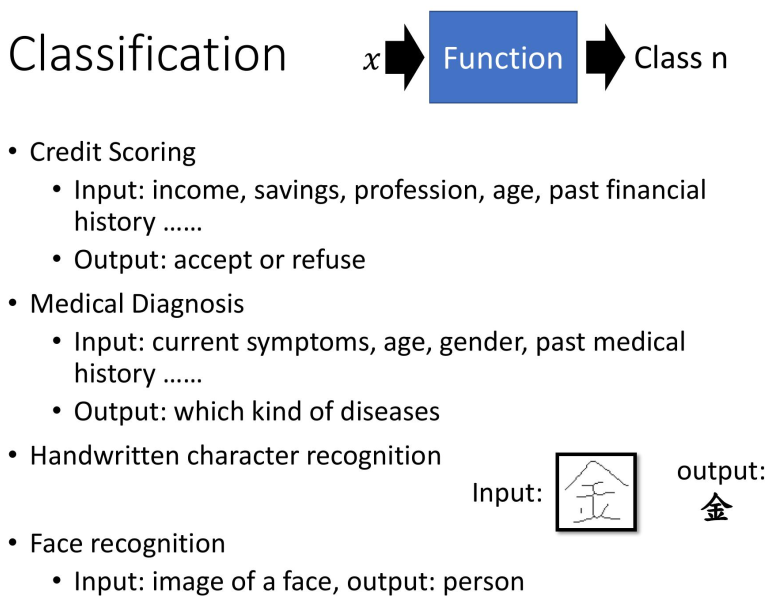
ML Lecture 4: Classification

> 臺灣大學人工智慧中心 科技部人工智慧技術暨全幅健康照護聯合研究中心 http://ai.ntu.edu.tw

Brief Introduction of "Classification"

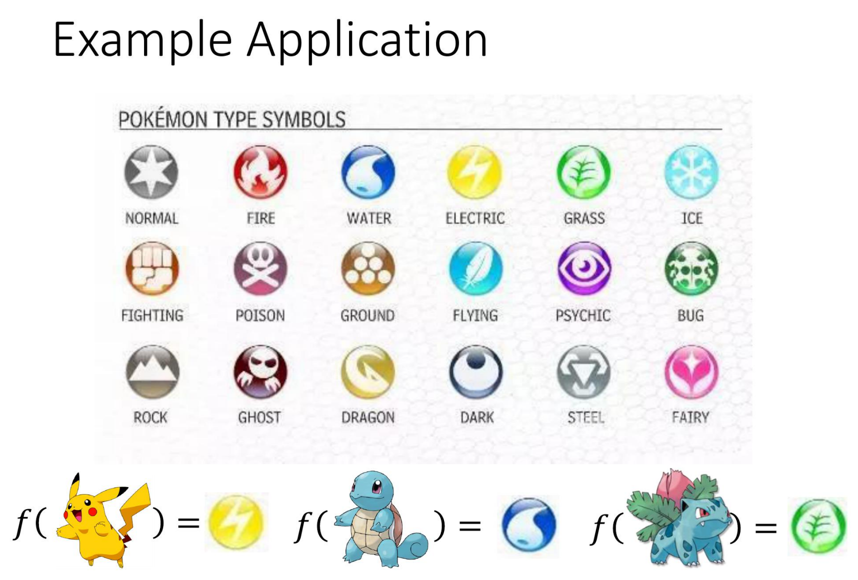
- Definition：找出一個 **function**，Input ***x***，Output ***x*** 屬於 ***n* *個 class 的哪一個***
- Application：金融業決定是否核准貸款、醫療上診斷、手寫文字辨識、人臉辨識等



Example Application

- Problem Definition：已知寶可夢有 18 種屬性，找出一個 function Input 一隻寶可夢，Output 他是屬於哪一種屬性
- Example：

  > 皮卡丘-->雷、傑尼龜-->水、妙蛙種子-->草



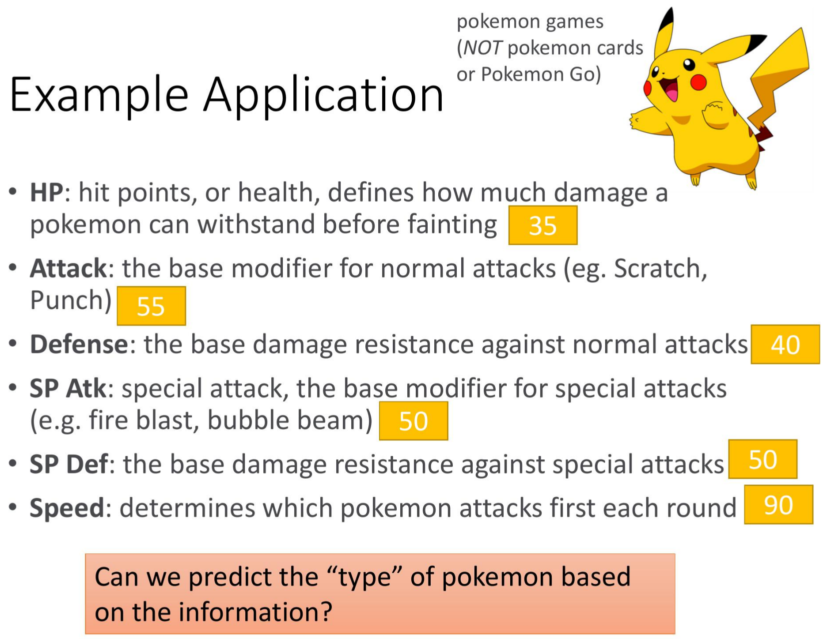
How to Express a Pokemon？

- 將寶可夢 **數值化** 作為 輸入
- 在寶可夢的電玩中，一隻寶可夢有許多特性，我們就以一組數字（一個vector）來描述他的特性

> 總強度、生命值、攻擊力、防禦力、特殊攻擊力、特殊攻擊的防禦力、速度

- Example：

> 皮卡丘 （320, 35, 55, 40, 50, 50, 90）



- 為什麼要預測寶可夢的屬性？

  在決鬥的時候，可能遇到圖鑑上沒有的寶可夢，這時可預測寶可夢的屬性，以相剋的屬性應戰。

Example Application


×	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	1×	1×	1×	1×	1×	½×	1×	0×	½×	1×	1×	1×	1×	1×	1×	1×	1×	1×
格鬥	2×	1×	½×	½×	1×	2×	½×	0×	2×	1×	1×	1×	1×	½×	2×	1×	2×	½×
飛行	1×	2×	1×	1×	1×	½×	2×	1×	½×	1×	1×	2×	½×	1×	1×	1×	1×	1×
毒	1×	1×	1×	½×	½×	½×	1×	½×	0×	1×	1×	2×	1×	1×	1×	1×	1×	2×
地面	1×	1×	0×	2×	1×	2×	½×	1×	2×	2×	1×	½×	2×	1×	1×	1×	1×	1×
岩石	1×	½×	2×	1×	½×	1×	2×	1×	½×	2×	1×	1×	1×	1×	2×	1×	1×	1×
蟲	1×	½×	½×	½×	1×	1×	1×	½×	½×	½×	1×	2×	1×	2×	1×	1×	2×	½×
幽靈	0×	1×	1×	1×	1×	1×	1×	2×	1×	1×	1×	1×	1×	2×	1×	1×	½×	1×
鋼	1×	1×	1×	1×	1×	2×	1×	1×	½×	½×	½×	1×	½×	1×	2×	1×	1×	2×
火	1×	1×	1×	1×	1×	½×	2×	1×	2×	½×	½×	2×	1×	1×	2×	½×	1×	1×
水	1×	1×	1×	1×	2×	2×	1×	1×	1×	2×	½×	½×	1×	1×	1×	½×	1×	1×
草	1×	1×	½×	½×	2×	2×	½×	1×	½×	½×	2×	½×	1×	1×	1×	½×	1×	1×
電	1×	1×	2×	1×	0×	1×	1×	1×	1×	1×	2×	½×	½×	1×	1×	½×	1×	1×
超能力	1×	2×	1×	2×	1×	1×	1×	1×	½×	1×	1×	1×	1×	½×	1×	1×	0×	1×
冰	1×	1×	2×	1×	2×	1×	1×	1×	½×	½×	½×	2×	1×	1×	½×	2×	1×	1×
龍	1×	1×	1×	1×	1×	1×	1×	1×	½×	1×	1×	1×	1×	1×	1×	2×	1×	0×
惡	1×	½×	1×	1×	1×	1×	1×	2×	1×	1×	1×	1×	1×	2×	1×	1×	½×	½×
妖精	1×	2×	1×	½×	1×	1×	1×	1×	½×	½×	1×	1×	1×	1×	1×	2×	2×	1×

（寶可夢屬性相剋表）

How to do Classification？

- 收集 data：例如將編號400以下的當作 training data，編號400以上的當作 testing data（共800隻的情況）
- data 表示法，以 pair 表示

> 例如：（皮卡丘、電），（傑尼龜、水），（妙蛙種子、草）

How to do Classification

- Training data for Classification

$(x^1, \hat{y}^1)$ $(x^2, \hat{y}^2)$ …… $(x^N, \hat{y}^N)$

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

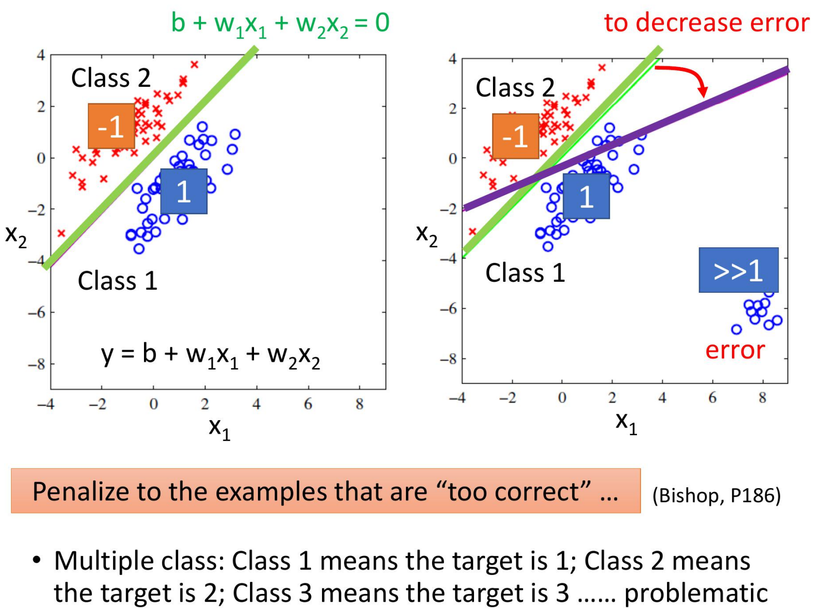
- 可將 Classification 當作 Regression 的問題硬解嗎？
    - Case 1：若圖形分佈如左圖（$y = b + w_1 * x_1 + w_2 * x_2$）

    > Regression 的 output 為綠色為等於 0 的線，恰為 Class 1 及 Class 2 的分界線 ==> Regression 可得出和 Classification 相似的結果

- Case 2：圖形的右下角有一些 output 遠大於 1 的 error 的點

  > 此時，Regression 的 output 為紫色的線，但對 Classification 而言，綠色分界線才好
  > ==> Regression 得出的結果 和 Classification 的結果相去甚遠

- 結論：<u>Regression 定義function 好壞的方式對 Classification **不適用**</u>，將無法得出好的結果



- Classification 理想解法：

  > f 為我們要找的 classification function(model) g 為 f 中內建的一個 function $L(f)$ 為 loss function，即 function f 在 training data 上 predict 錯誤的次數總和

Ideal Alternatives

- Function (Model):

  $f(x)$: $x \rightarrow$ $g(x) > 0$ Output = class 1; $else$ Output = class 2

- Loss function:

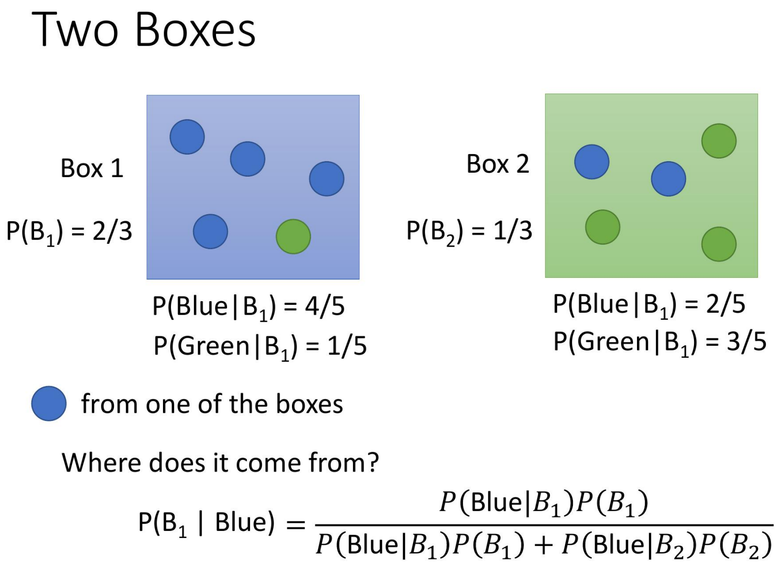
$$L(f) = \sum_n \delta(f(x^n) \neq \hat{y}^n)$$

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

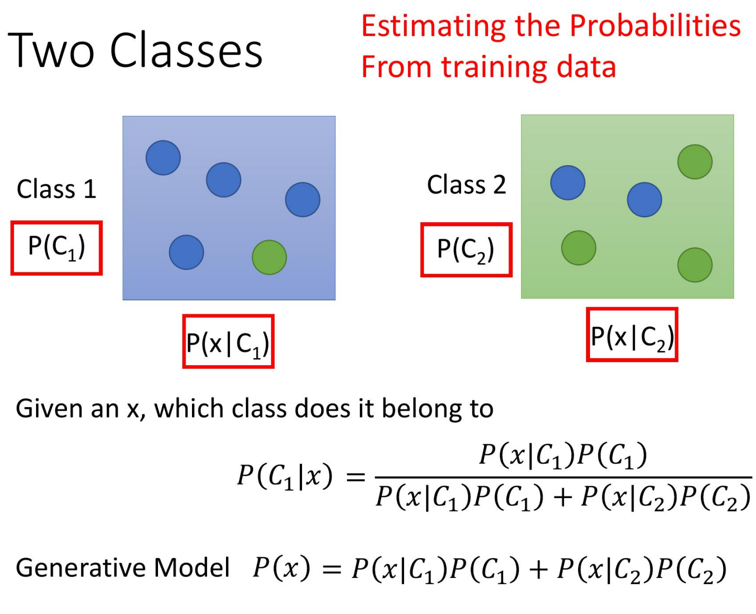
- Find the best function:
  - Example: Perceptron, SVM Not Today

Solution（Probabilistic Perspective）

- 條件機率：如下圖所示，可輕易計算

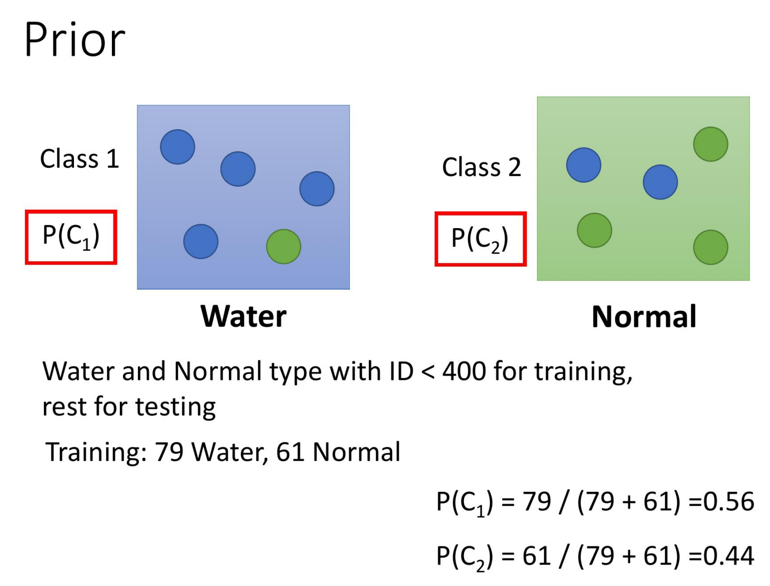


- **將 Box 1, Box 2，換成 Class 1, Class 2** Input 一隻寶可夢，看他從哪個 class 來的機率最大，機率最大的 class 即為 Output
- **Generative Model**： 從 training data 估測出 $P(C_1)$、$P(x|C_1)$、$P(C_2)$、$P(x|C_2)$，有這 4 個值，即可算出 <u>每一個 x 出現的機率</u>



- **Prior**：$P(C_1)$、$P(C_2)$ 的機率可稱作 Prior

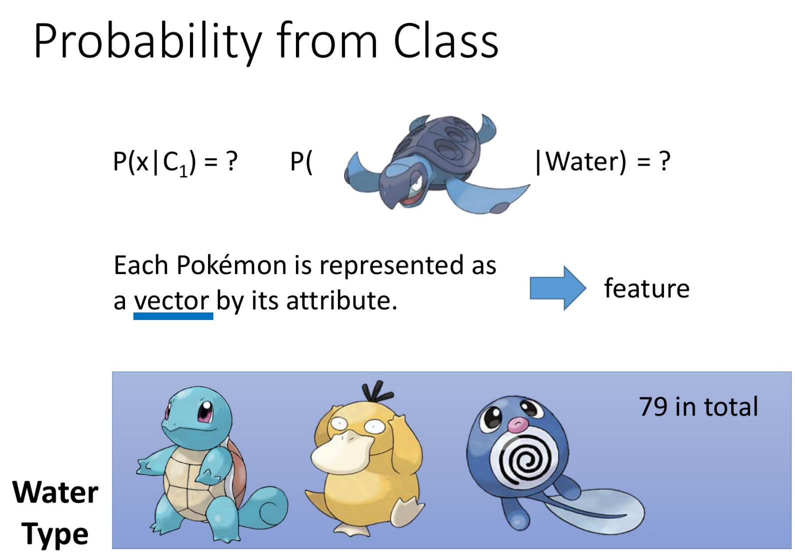
> 例如：class 1 為水系(79隻)、 class 2 為一般系(61隻) ==> $P(C_1) = 79/(79+61) = 0.56$
> ==> $P(C_2) = 61/(79+61) = 0.44$



- **Probability from Class**

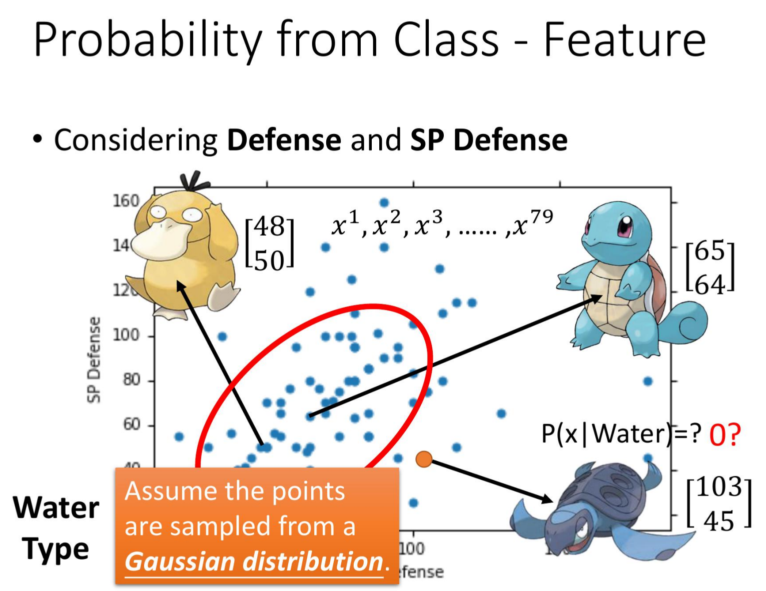
  每隻寶可夢都可以一個 **vector** 表示，這個 vector 又可稱之為 **feature(特徵值)**

  > 例如：從水系神奇寶貝 sample 出一隻海龜的機率有多大？



- 將 79 隻水系神奇寶貝依 **防禦力** 及 **特殊防禦力** 畫在二維平面上

  > **轉化問題**： 給一個新的點，代表的是沒有在 training data 裡面的寶可夢 從 Gaussian distribution 裡面 sample 出這個點的機率是多少？
  >
  > ==> **給這 79 個點， 要怎麼找到那個 Gaussian distribution？**



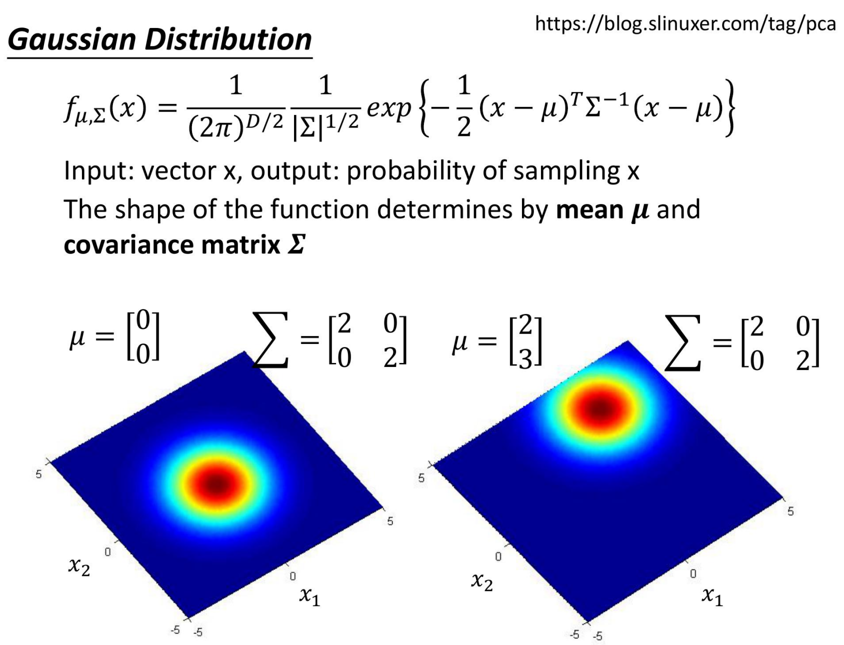
Gaussian distribution

- **定義**

> **Input** ：vector x（代表一隻寶可夢的數值） **Output**：機率（寶可夢被 s ample 出的機率）

機率的分佈由以下兩者決定

> **Mean** (μ) ：是 vector **Variance** (Σ)：是 matrix



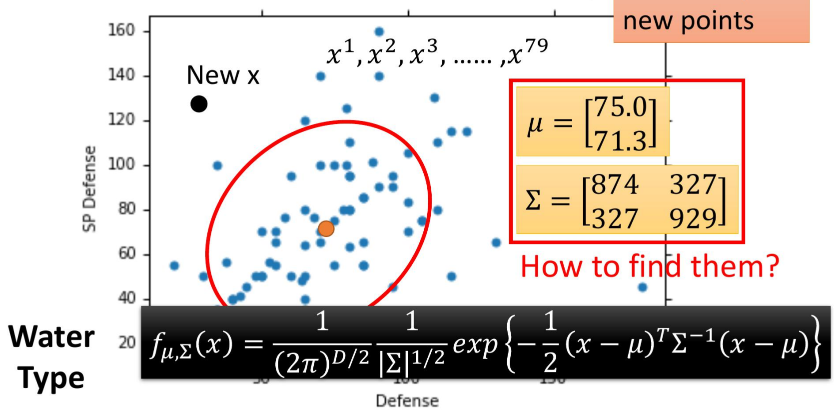
估算 Gaussian distribution

> 知道 μ 以及 x，就能將 function 估計出來 轉化問題 ==> **如何找 μ 以及 Σ 呢？**

Probability from Class

Assume the points are sampled from a Gaussian distribution

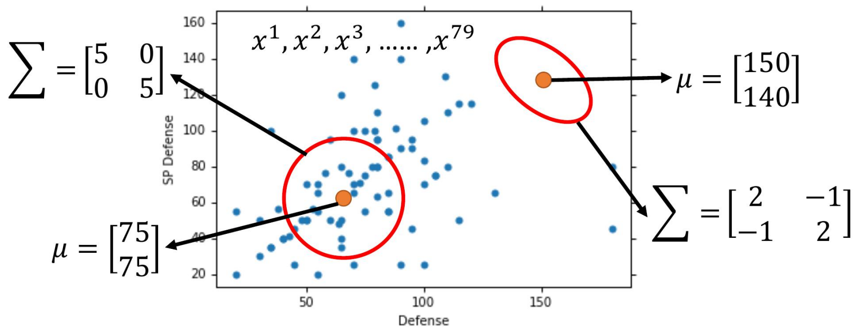
Find the Gaussian distribution behind them ➡ Probability for new points



Maximum Likelihood, $L(\mu, \Sigma)$

- 給一組 Gaussian 的 μ 及 Σ，算這個 Gaussian sample 出這 79 個點的機率 求出哪一個 Gaussian，sample 出這 79 個點的機率最大！

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 μ and covariance matrix Σ can generate these points. ➡ Different Likelihood

Likelihood of a Gaussian with mean μ and covariance matrix Σ

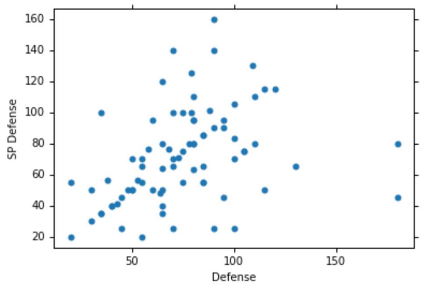
= the probability of the Gaussian samples $x^1, x^2, x^3, \ldots\ldots, x^{79}$

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

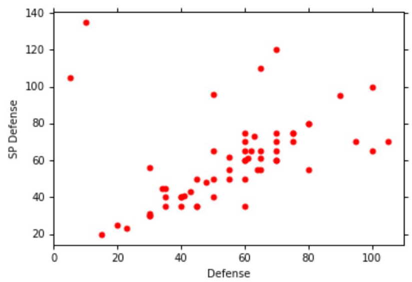
==> 估計出 μ^2 及 Σ^2 後，就可以代回貝氏定理，做分類問題了

Maximum Likelihood

Class 1: Water



Class 2: Normal



$$\mu^1 = \begin{bmatrix} 75.0 \\ 71.3 \end{bmatrix} \quad \Sigma^1 = \begin{bmatrix} 874 & 327 \\ 327 & 929 \end{bmatrix} \qquad \mu^2 = \begin{bmatrix} 55.6 \\ 59.8 \end{bmatrix} \quad \Sigma^2 = \begin{bmatrix} 847 & 422 \\ 422 & 685 \end{bmatrix}$$

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(C1)
= 79 / (79 + 61) =0.56

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

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

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

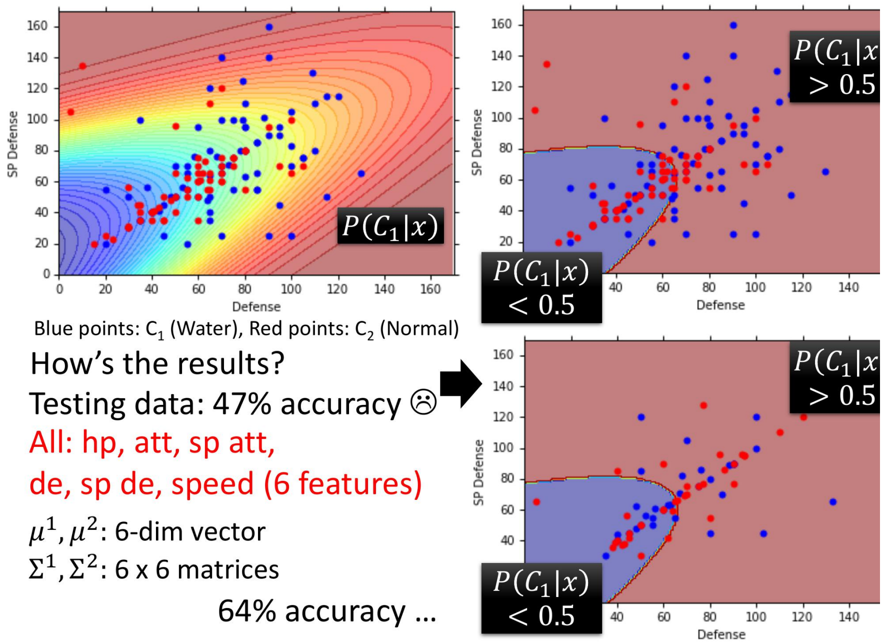
P(C2)
= 61 / (79 + 61)
=0.44

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

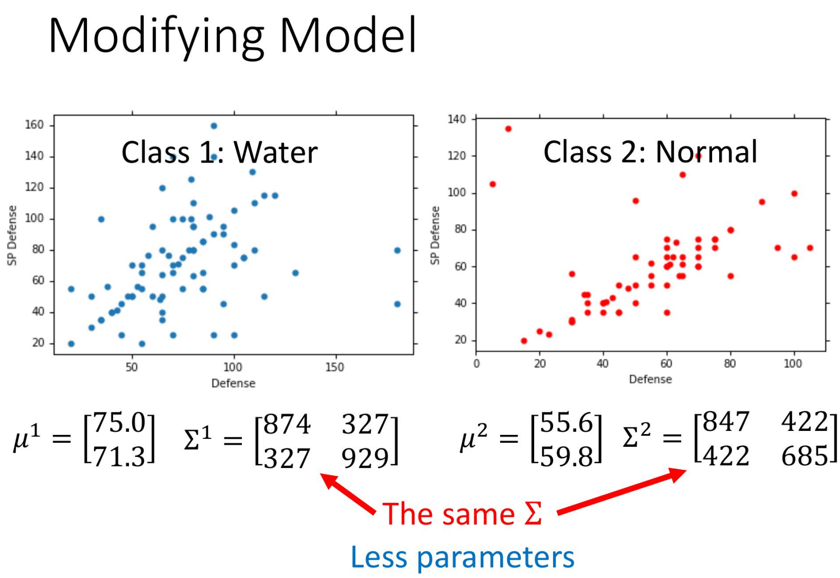
If $P(C_1|x) > 0.5$ ➡ x belongs to class 1 (Water)

Modifying Model

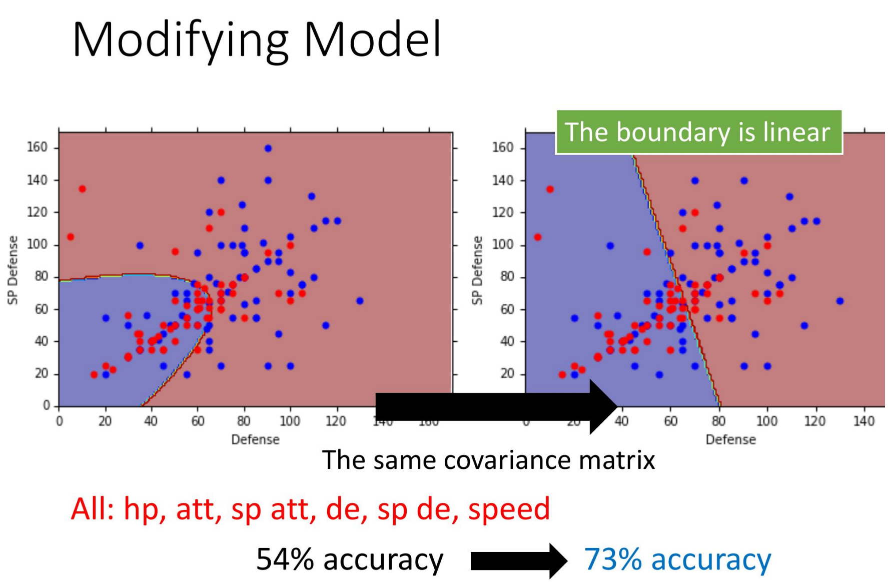
- 遇到問題： 在 **二維空間** 的結果不甚好（47% 的正確率） 就提升到 **高維空間**，如六維（使用6個特徵值）可得到 64% 的正確率，<u>*但仍不夠好*</u>



- **共用 Covariance matrix** 原因：Input 的 **feature size** 跟 **covariance matrix 的平方**成正比 當 feature size 越大，variance 就越大，**容易 overfitting** ==> 所以強迫他們共用 covariance matrix，有效減少參數



- **Boundary 的改變** 共用之前，boundary 是 **<u>曲線</u>**（non-linear model） 共用之後，boundary 變成 **<u>直線</u>**（linear model) 正確率：54% ----> 73%



Three Steps of Probabilistic Model

1. **Function Set（Model）**

   Input x，將不同的 probability distribution 積分起來，就得到一個 model

   > If $P(C_1|x) > 0.5$, output **class 1** Else, output **class 2**

   1. **Goodness of a function** 找一組可以「最大化產生資料集 likelihood」的 (μ, Σ)
2. **Find the best function**

Three Steps

- Function Set (Model):

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

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

Otherwise, output: class 2

- Goodness of a function:
  - The mean μ and covariance Σ that maximizing the likelihood (the probability of generating data)
- Find the best function: easy

Porbability Distribution

- 可以選擇任何你喜歡的機率模型，不一定要用 Gaussian distribution，可用 data set 決定

- x 是個**一維向量**，假設**每一個 dimension** 從 model 中被 generate 出來的機率是 **Independent** 的 這時，稱作使用 **Naive Bayes Classifier** 而
  $P(x|C_1) = P(x_1|C_1) * P(x_2|C_1) * ...... * P(x_k|C_1)$

Posterior Probability

- $P(C_1|x) = \frac{1}{1+exp(-z)} = \sigma(z)$，$\sigma(z)$ 為 $sigmoid function$

> z 趨近 "正無窮大"，$\sigma(z)$ 趨近 1 z 趨近 "負無窮大"，$\sigma(z)$趨近 0

- **z 的推導過程**

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

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

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

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

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

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

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

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

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

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

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

$$\boxed{\begin{aligned} z = ln\frac{|\Sigma^2|^{1/2}}{|\Sigma^1|^{1/2}} & -\frac{1}{2}x^T(\Sigma^1)^{-1}x + (\mu^1)^T(\Sigma^1)^{-1}x - \frac{1}{2}(\mu^1)^T(\Sigma^1)^{-1}\mu^1 \\ & +\frac{1}{2}x^T(\Sigma^2)^{-1}x - (\mu^2)^T(\Sigma^2)^{-1}x + \frac{1}{2}(\mu^2)^T(\Sigma^2)^{-1}\mu^2 + ln\frac{N_1}{N_2}\end{aligned}}$$

- **結論**

  $P(C_1|x) = \sigma(w^T * b)$，**直接求出 w 跟 b 即可求解**

  不需要捨近求遠算出 $N_1, N_2, \mu^1, \mu^2, \Sigma$ 再回來求 w 跟 b

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

$$\boxed{\begin{aligned} z = {} & \cancel{ln\frac{|\Sigma^2|^{1/2}}{|\Sigma^1|^{1/2}}} \cancel{-\frac{1}{2}x^T(\Sigma^1)^{-1}x} + (\mu^1)^T(\Sigma^1)^{-1}x - \frac{1}{2}(\mu^1)^T(\Sigma^1)^{-1}\mu^1 \\ & \cancel{+\frac{1}{2}x^T(\Sigma^2)^{-1}x} - (\mu^2)^T(\Sigma^2)^{-1}x + \frac{1}{2}(\mu^2)^T(\Sigma^2)^{-1}\mu^2 + ln\frac{N_1}{N_2} \end{aligned}}$$

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

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

$P(C_1|x) = \sigma(w \cdot x + b)$ How about directly find ***w*** and b?

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

Then we have ***w*** and b