分类

Classification: Probabilistic Generative Model

概率生成模型

Classification

x → Function → Class n

- Credit Scoring
  - Input: income, savings, profession, age, past financial history ……
  - Output: accept or refuse binary classification
- Medical Diagnosis
  - Input: current symptoms, age, gender, past medical history ……
  - Output: which kind of diseases
- Handwritten character recognition

Input:  output: 金

- Face recognition
  - Input: image of a face, output: person

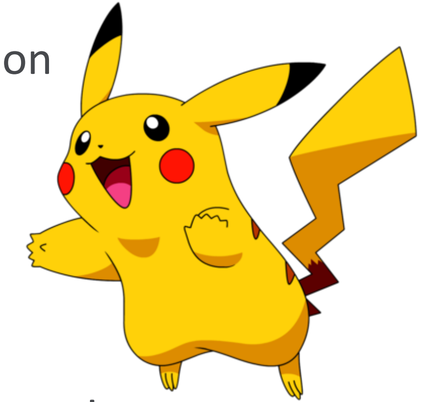
Example Application

$f($  $) =$ 

$f($  $) =$ 

$f($  $) =$ 

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



Example Application

features

- **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

one pokemon = a feature vector

Can we predict the "type" of pokemon based on the information?

属性相克表

Example Application

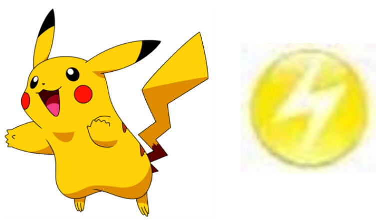
×	防禦方的屬性																	
攻擊方的屬性	一般	格鬥	飛行	毒	地面	岩石	蟲	幽靈	鋼	火	水	草	電	超能力	冰	龍	惡	妖精
一般	1×	1×	1×	1×	1×	½×	1×	0×	½×	1×	1×	1×	1×	1×	1×	1×	1×	1×
格鬥	2×	1×	½×	½×	1×	2×	½×	0×	2×	1×	1×	1×	1×	½×	2×	1×	2×	½×
飛行	1×	2×	1×	1×	1×	½×	2×	1×	½×	1×	1×	2×	½×	1×	1×	1×	1×	1×
毒	1×	1×	1×	½×	½×	½×	1×	½×	0×	1×	1×	2×	1×	1×	1×	1×	1×	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×

這些倍數適用於XY及之後的遊戲。

How to do Classification

- Training data for Classification

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



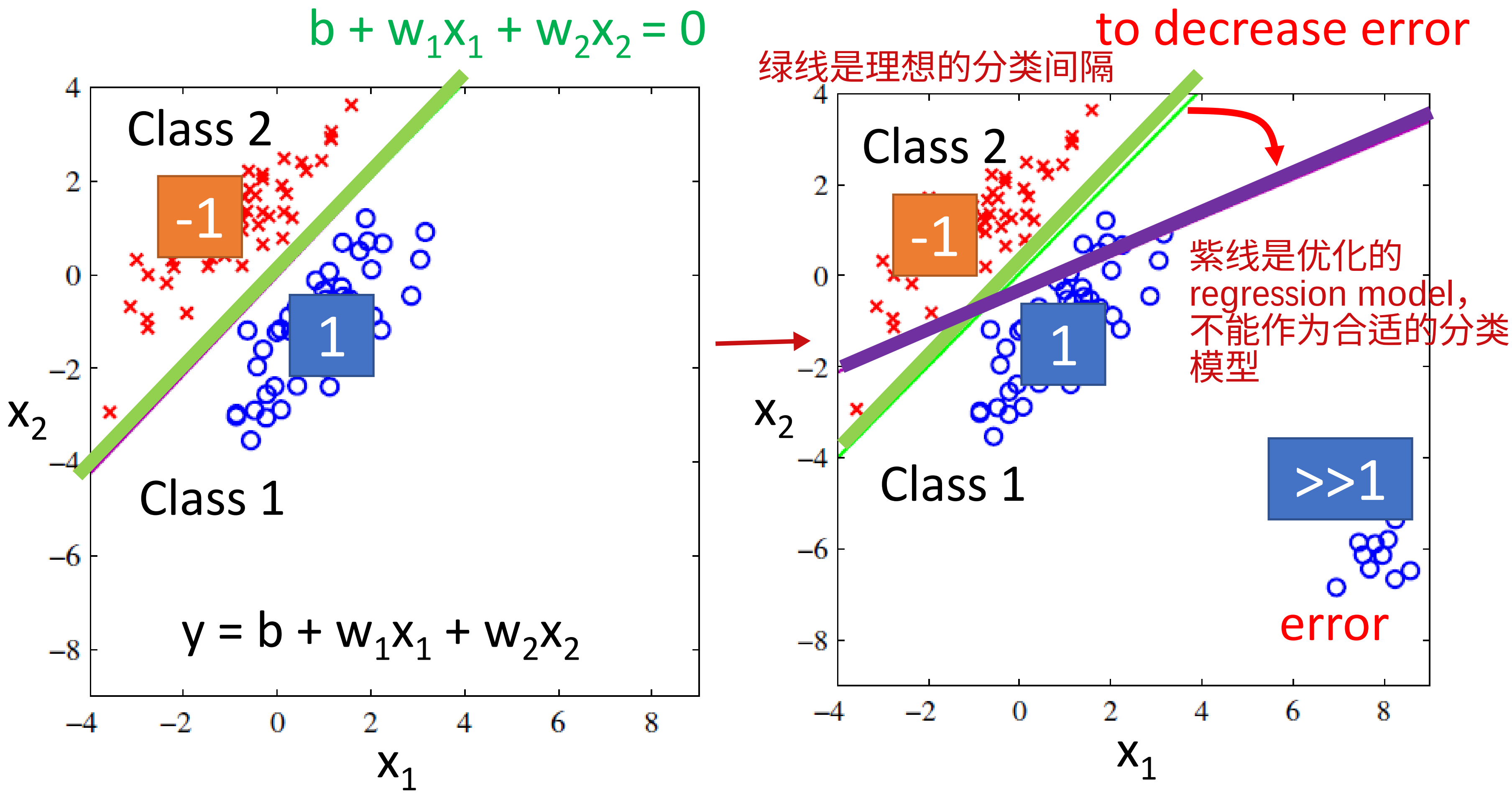
Classification as Regression?

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

把classification当作regression来做



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):

$$f(x)$$

$$x \Rightarrow \begin{cases} g(x) > 0 & \text{Output = class 1} \\ else & \text{Output = class 2} \end{cases}$$

- 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

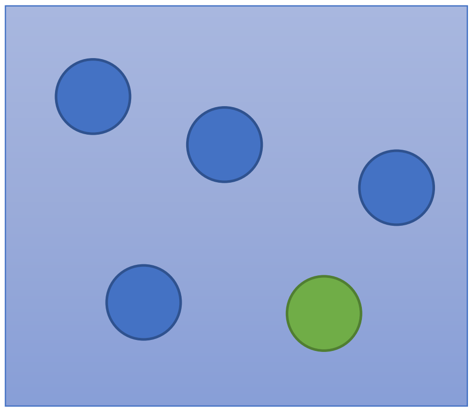
感知机 支持向量机

Not Today

Two Boxes

Box 1

$P(B_1) = 2/3$

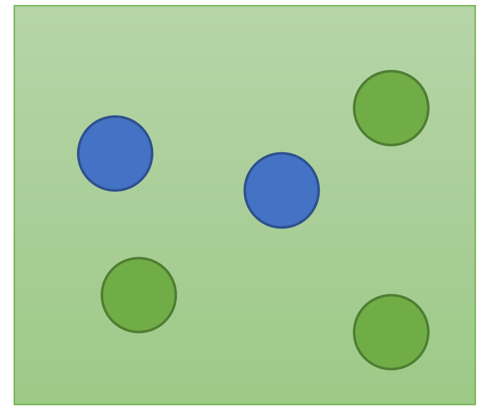


$P(\text{Blue}|B_1) = 4/5$

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

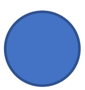
Box 2

$P(B_2) = 1/3$



$P(\text{Blue}|B_1) = 2/5$

$P(\text{Green}|B_1) = 3/5$

 from one of the boxes

Where does it come from?

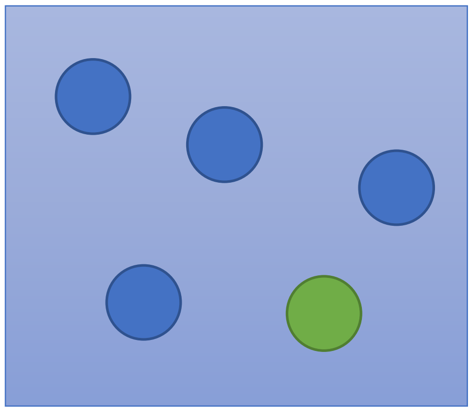
$$P(B_1 \mid \text{Blue}) = \frac{P(\text{Blue}|B_1)P(B_1)}{P(\text{Blue}|B_1)P(B_1) + P(\text{Blue}|B_2)P(B_2)}$$

Two Classes

Estimating the Probabilities
From training data

Class 1

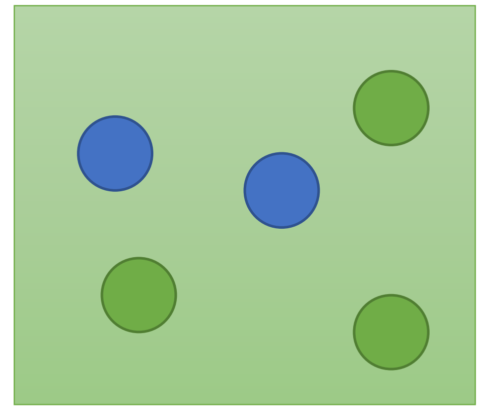
$\boxed{P(C_1)}$



$\boxed{P(x|C_1)}$

Class 2

$\boxed{P(C_2)}$



$\boxed{P(x|C_2)}$

Given an x, which class does it belong to

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

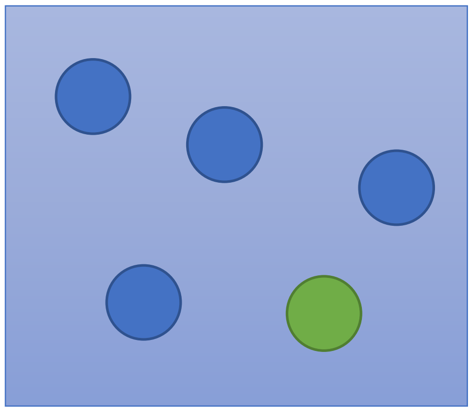
Generative Model $P(x) = P(x|C_1)P(C_1) + P(x|C_2)P(C_2)$

生成模型

Prior

Class 1

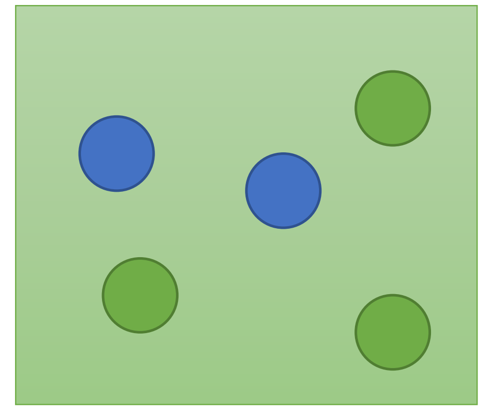
$P(C_1)$



Water

Class 2

$P(C_2)$



Normal

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

$P(x|C_1) = ?$ P( |Water) = ?

Each Pokémon is represented as a vector by its attribute. → $\boxed{\text{feature}}$

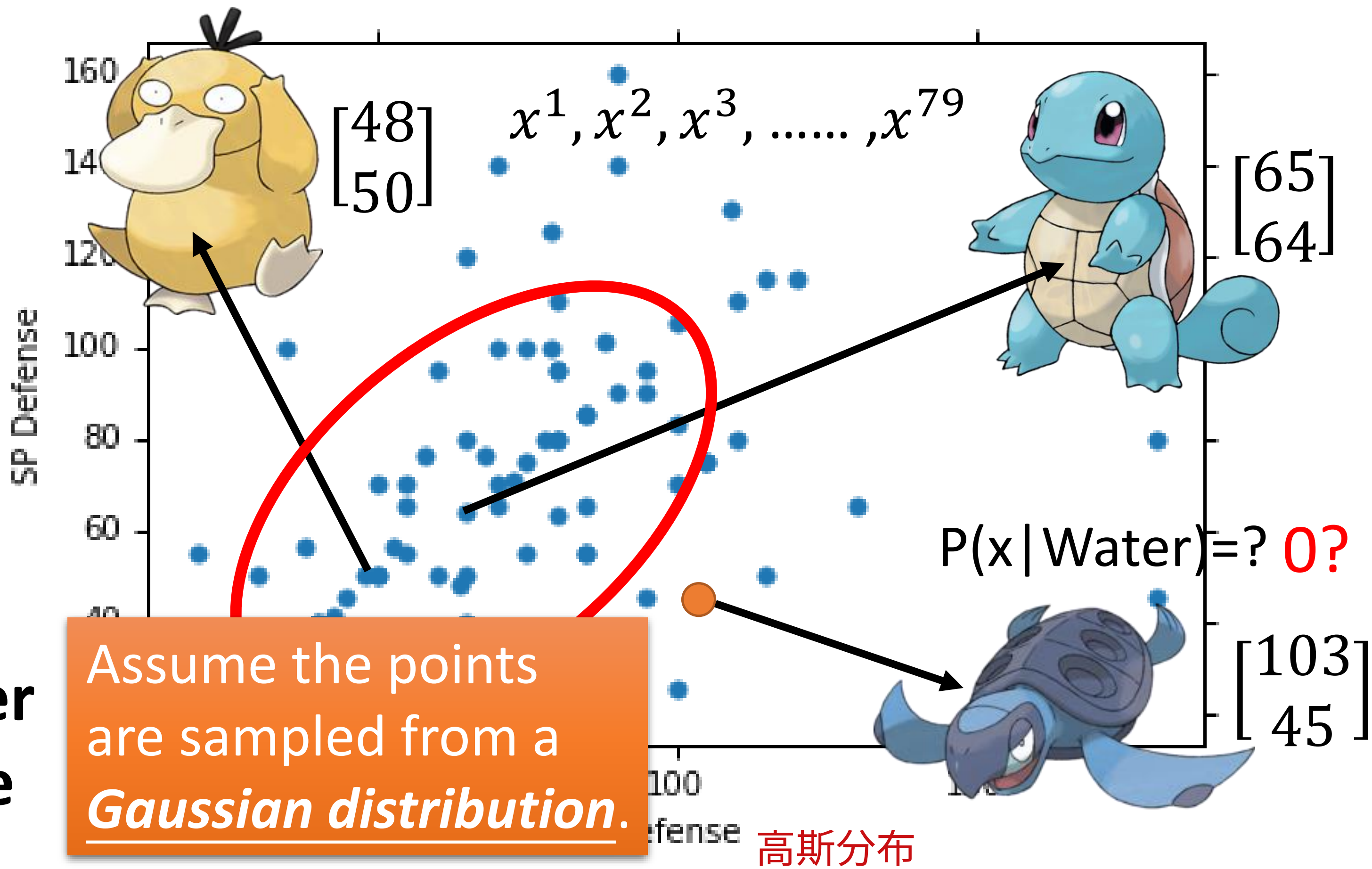
Water Type

79 in total

杰尼龟 可达鸭 蚊香蝌蚪

Probability from Class - Feature

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



Water Type

Assume the points are sampled from a ***Gaussian distribution***.

高斯分布

Gaussian Distribution

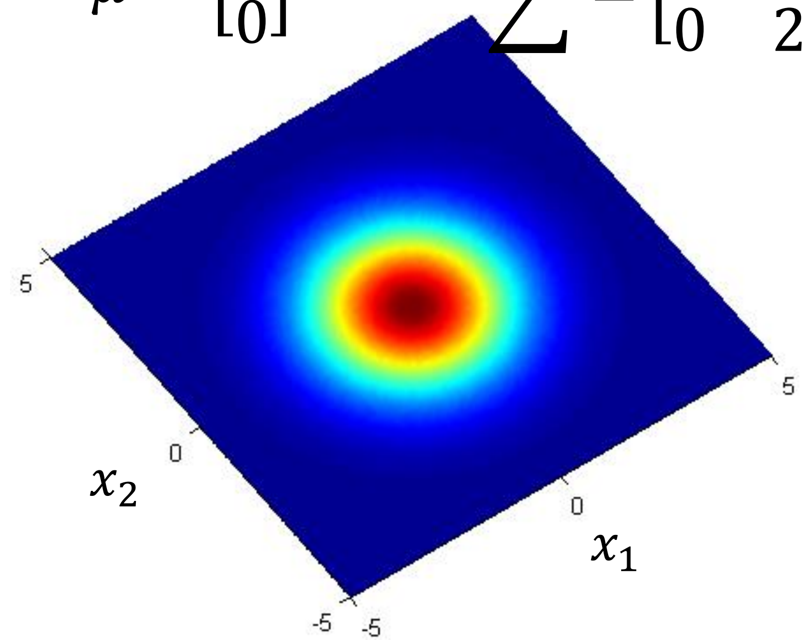
$$f_{\mu,\Sigma}(x) = \frac{1}{(2\pi)^{D/2}} \frac{1}{|\Sigma|^{1/2}} exp\left\{-\frac{1}{2}(x-\mu)^T \Sigma^{-1}(x-\mu)\right\}$$

Input: vector x, output: probability of sampling x

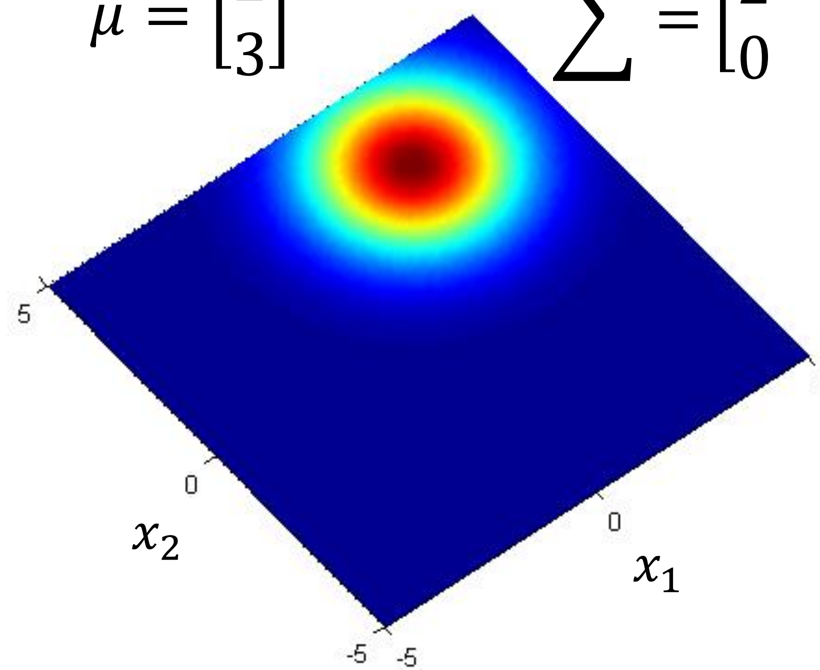
The shape of the function determines by **mean $\boldsymbol{\mu}$** and **covariance matrix $\boldsymbol{\Sigma}$**

相同的Σ，不同的μ

$\mu = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$ $\sum = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}$



$\mu = \begin{bmatrix} 2 \\ 3 \end{bmatrix}$ $\sum = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}$



Gaussian Distribution

https://blog.slinuxer.com/tag/pca

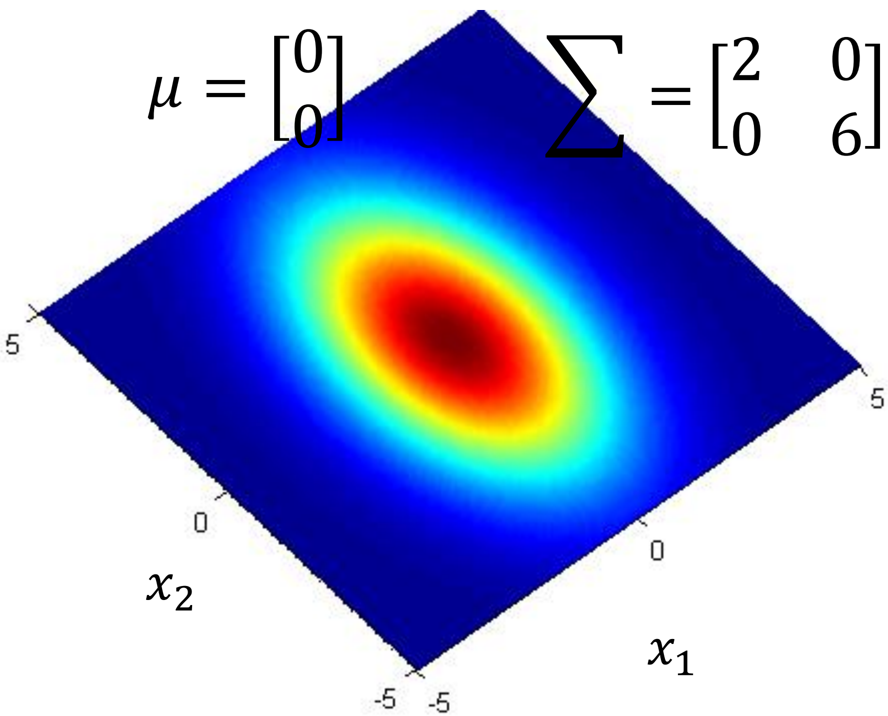
$$f_{\mu,\Sigma}(x) = \frac{1}{(2\pi)^{D/2}} \frac{1}{|\Sigma|^{1/2}} exp\left\{-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)\right\}$$

Input: vector x, output: probability of sampling x

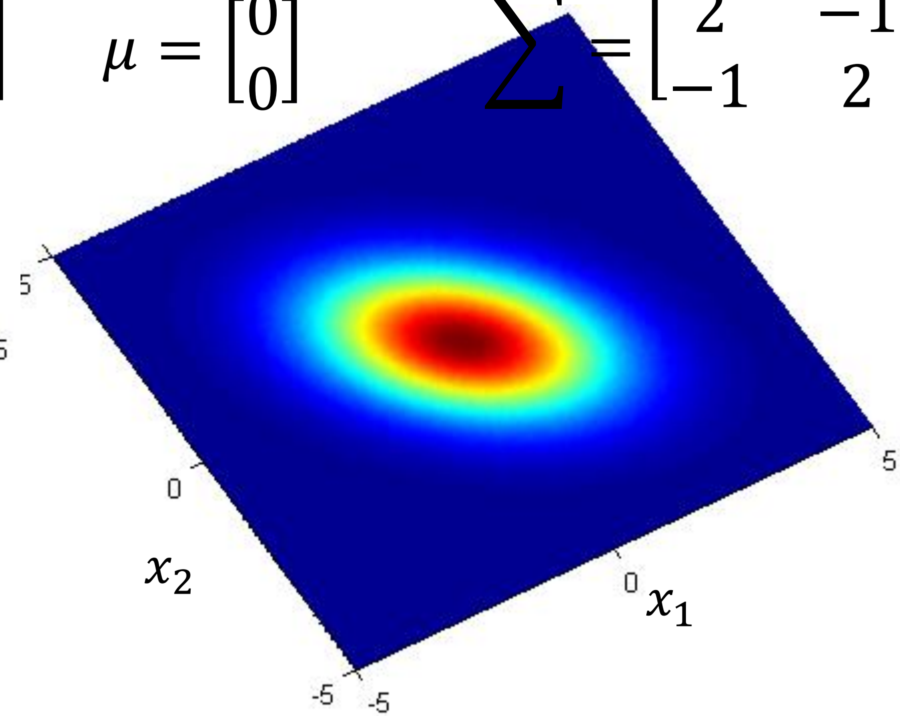
The shape of the function determines by **mean** $\boldsymbol{\mu}$ and **covariance matrix** $\boldsymbol{\Sigma}$

相同的μ，不同的Σ

$$\mu = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad \sum = \begin{bmatrix} 2 & 0 \\ 0 & 6 \end{bmatrix}$$



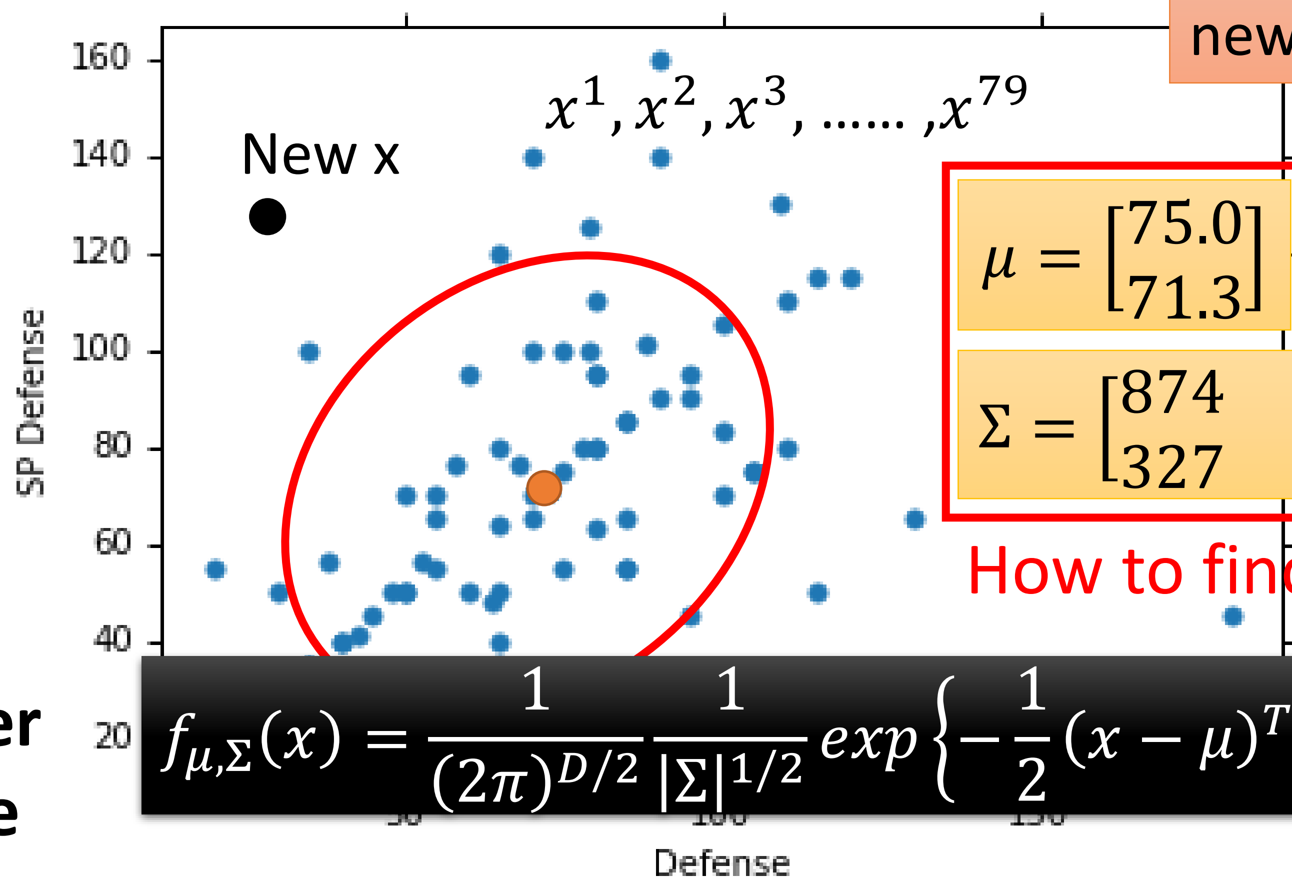
$$\mu = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad \sum = \begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix}$$



Probability from Class

Assume the points are sampled from a Gaussian distribution

Find the Gaussian distribution behind them → Probability for new points



$$\mu = \begin{bmatrix} 75.0 \\ 71.3 \end{bmatrix}$$

$$\Sigma = \begin{bmatrix} 874 & 327 \\ 327 & 929 \end{bmatrix}$$

How to find them?

Water Type

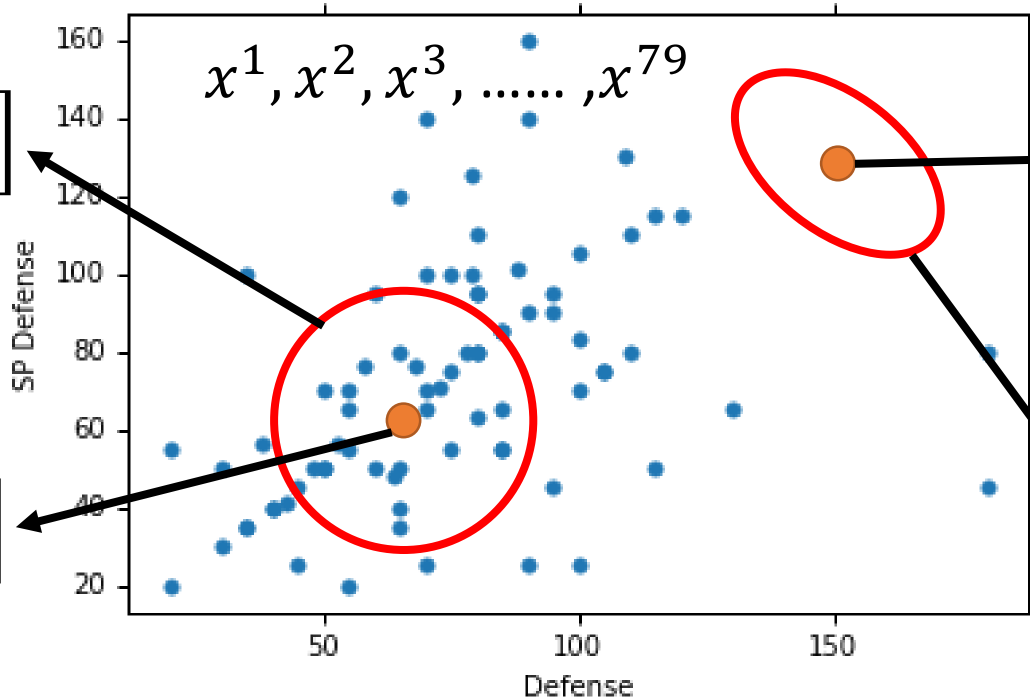
$$f_{\mu,\Sigma}(x) = \frac{1}{(2\pi)^{D/2}} \frac{1}{|\Sigma|^{1/2}} exp\left\{-\frac{1}{2}(x-\mu)^T \Sigma^{-1} (x-\mu)\right\}$$

Maximum Likelihood

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

$$\Sigma = \begin{bmatrix} 5 & 0 \\ 0 & 5 \end{bmatrix}$$

$$\mu = \begin{bmatrix} 75 \\ 75 \end{bmatrix}$$



$$\mu = \begin{bmatrix} 150 \\ 140 \end{bmatrix}$$

$$\Sigma = \begin{bmatrix} 2 & -1 \\ -1 & 2 \end{bmatrix}$$

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

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

Maximum Likelihood

We have the "Water" type Pokémons: $x^1, x^2, x^3, \ldots\ldots, x^{79}$

We assume $x^1, x^2, x^3, \ldots\ldots, x^{79}$ generate from the Gaussian (μ^*, Σ^*) with the ***maximum likelihood***

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

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

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

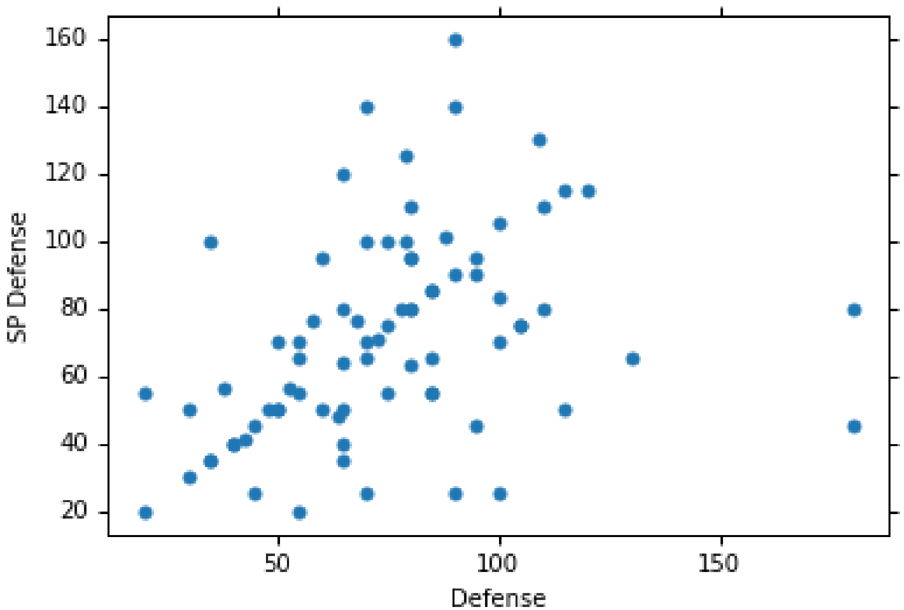
$$\mu^* = \frac{1}{79} \sum_{n=1}^{79} x^n$$

average

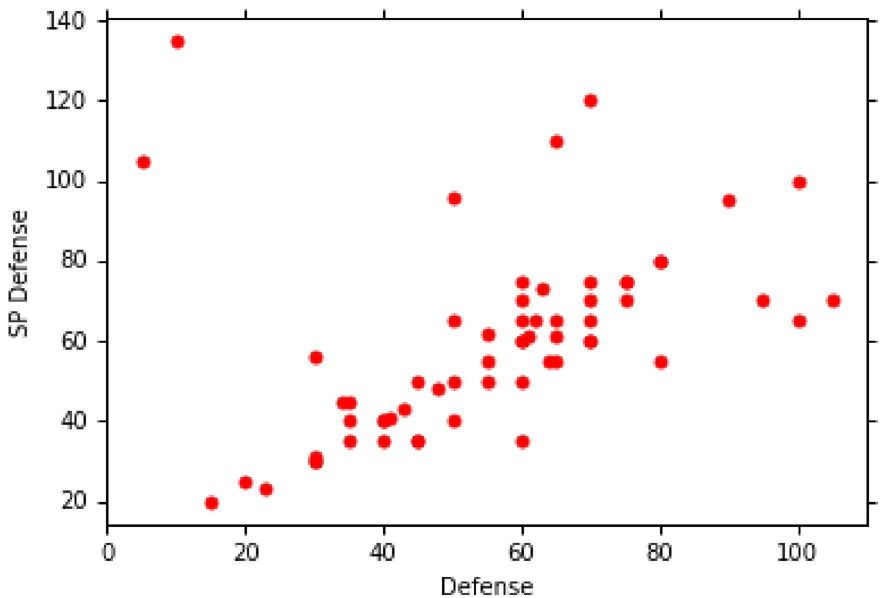
$$\Sigma^* = \frac{1}{79} \sum_{n=1}^{79} (x^n - \mu^*)(x^n - \mu^*)^T$$

Maximum Likelihood

Class 1: Water



Class 2: Normal



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

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

Now we can do classification ☺

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

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

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

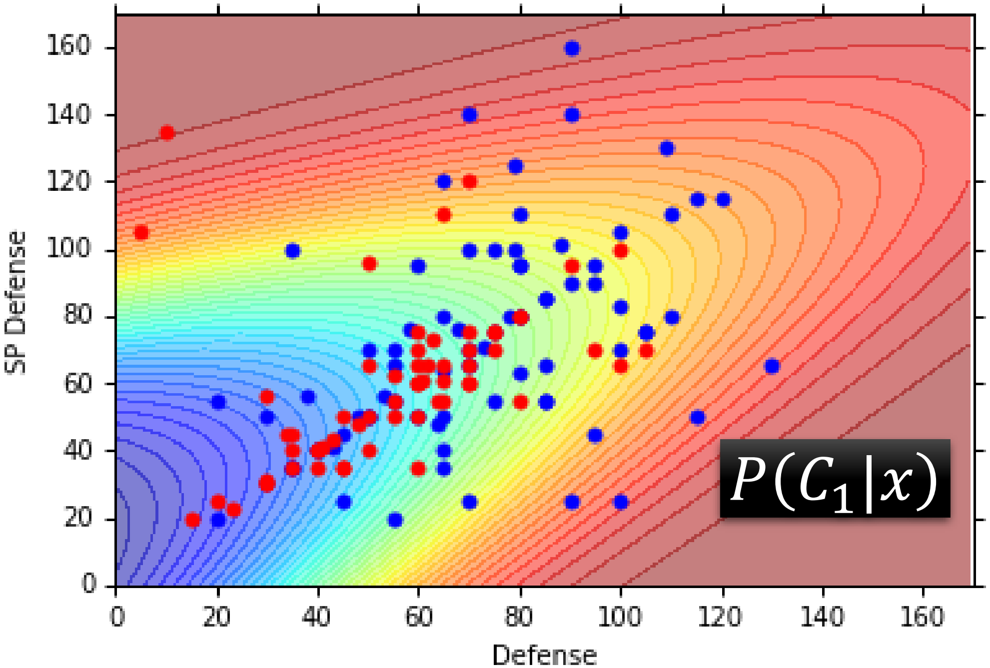
$$P(C_1|x) = \frac{P(x|C_1)P(C_1)}{P(x|C_1)P(C_1) + P(x|C_2)P(C_2)}$$

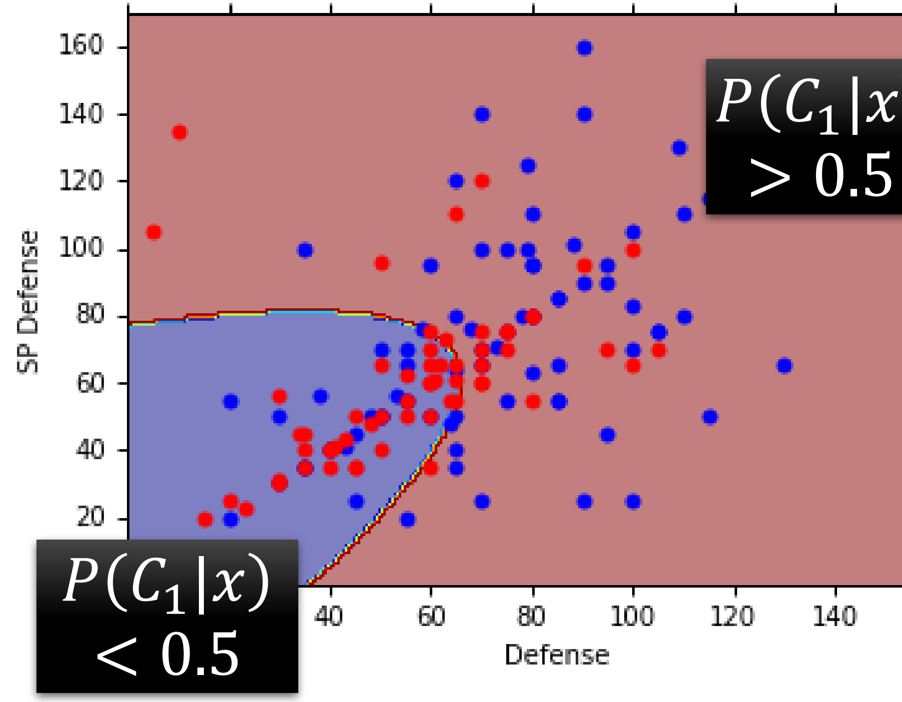
$$f_{\mu^2,\Sigma^2}(x) = \frac{1}{(2\pi)^{D/2}} \frac{1}{|\Sigma^2|^{1/2}} exp\left\{-\frac{1}{2}(x-\mu^2)^T(\Sigma^2)^{-1}(x-\mu^2)\right\}$$

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)





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

How's the results?

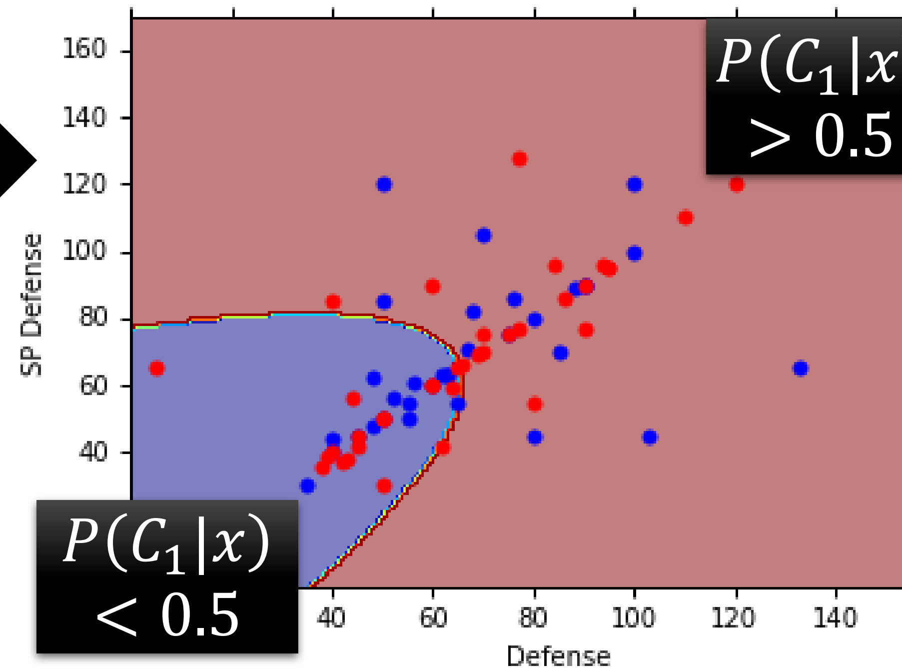
Testing data: 47% accuracy

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

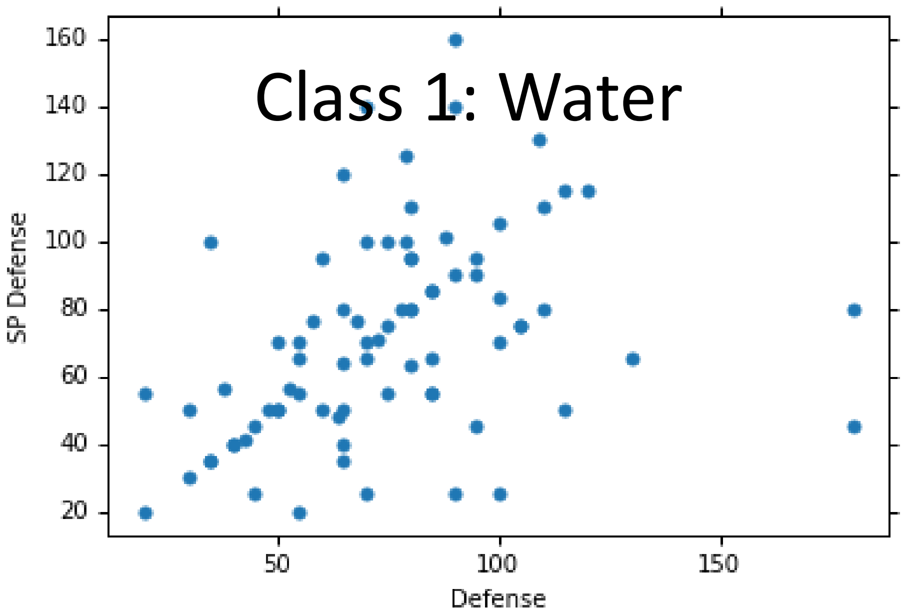
μ^1, μ^2: 7-dim vector

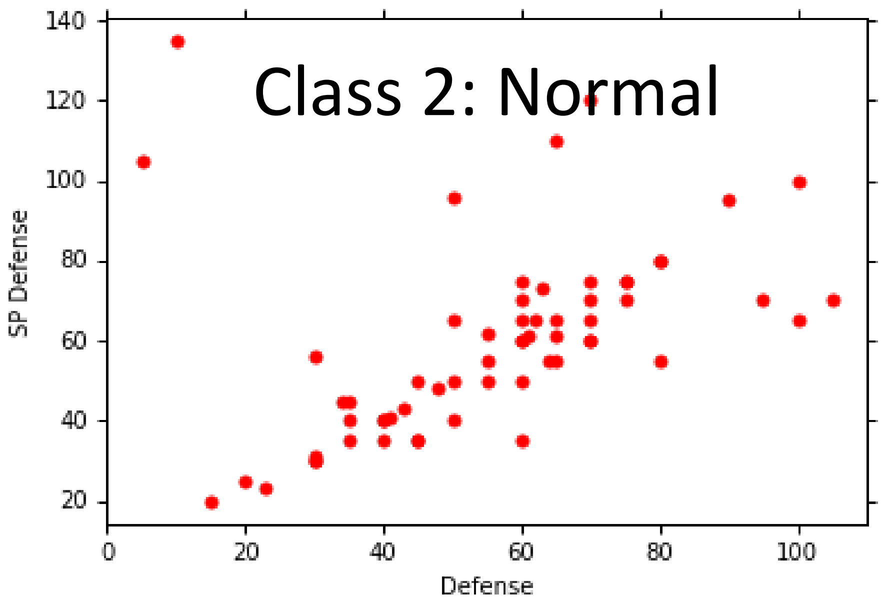
Σ^1, Σ^2: 7 x 7 matrices

54% accuracy … ☹



Modifying Model





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

The same Σ

Less parameters

Modifying Model

Ref: Bishop, chapter 4.2.2

- Maximum likelihood

"Water" type Pokémons: $x^1, x^2, x^3, \ldots\ldots, x^{79}$

"Normal" type Pokémons: $x^{80}, x^{81}, x^{82}, \ldots\ldots, x^{140}$

μ^1 $\qquad$ Σ $\qquad$ μ^2

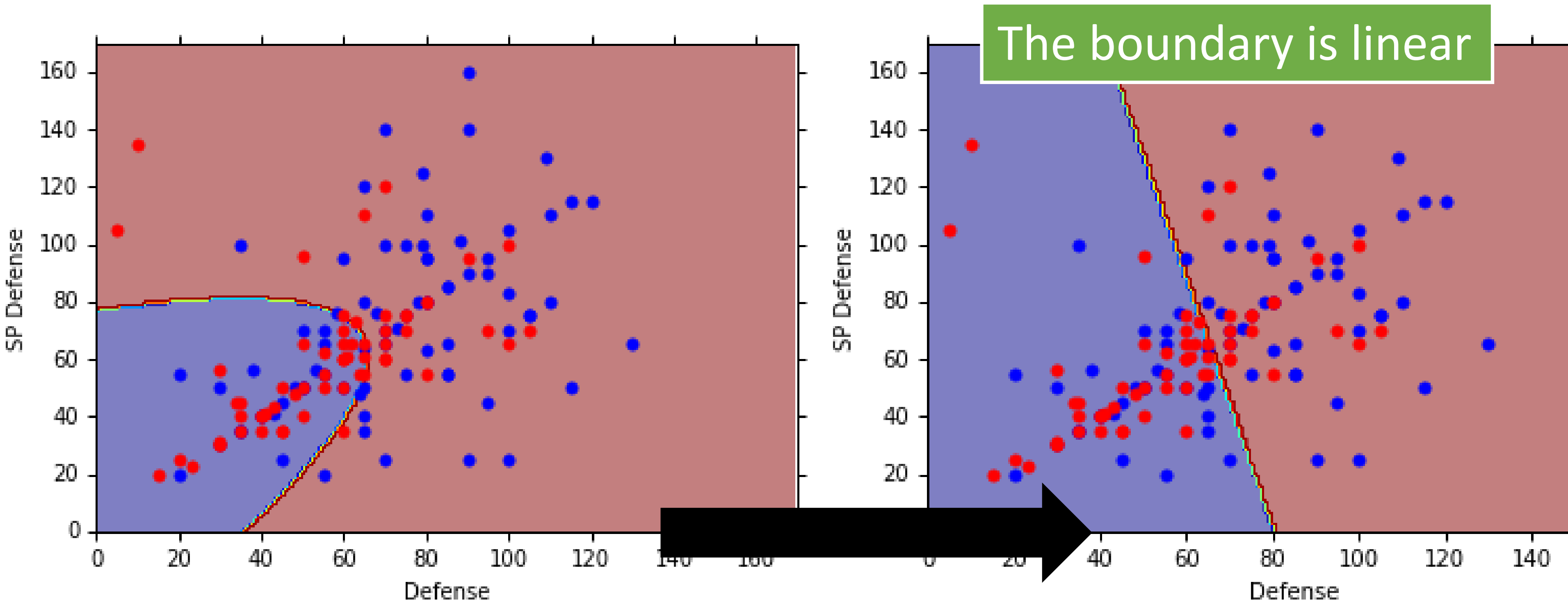
Find μ^1, μ^2, Σ maximizing the likelihood $L(\mu^1, \mu^2, \Sigma)$

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

μ^1 and μ^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 → 73% accuracy

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

Probability Distribution

- You can always use the distribution you like ☺

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

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

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*.

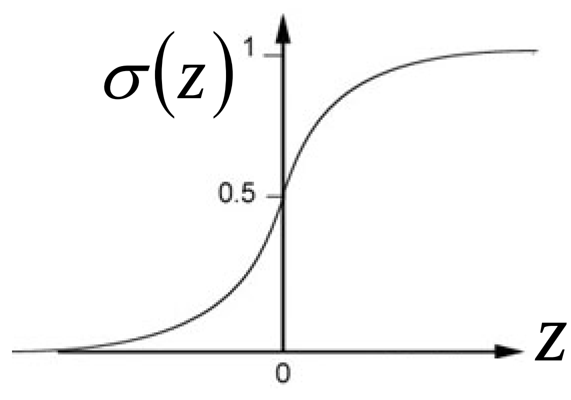
Posterior Probability

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

$$= \frac{1}{1 + \frac{P(x|C_2)P(C_2)}{P(x|C_1)P(C_1)}} = \frac{1}{1 + exp(-z)} = \sigma(z)$$

Sigmoid function

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



Warning of Math

Posterior Probability

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

End of Warning

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

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

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

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

$$P(C_1|x) = \sigma(w \cdot x + b)$$

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

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