XIV. Orthogonal Transform and Multiplexing

◉ 14-A Orthogonal and Dual Orthogonal

Any $M \times N$ discrete linear transform can be expressed as the matrix form:

$$\underbrace{\begin{bmatrix} y[0] \\ y[1] \\ y[2] \\ \vdots \\ y[M-1] \end{bmatrix}}_{\mathbf{Y}=} = \underbrace{\begin{bmatrix} \phi_0^*[0] & \phi_0^*[1] & \phi_0^*[2] & \cdots & \phi_0^*[N-1] \\ \phi_1^*[0] & \phi_1^*[1] & \phi_1^*[2] & \cdots & \phi_1^*[N-1] \\ \phi_2^*[0] & \phi_2^*[1] & \phi_2^*[2] & \cdots & \phi_2^*[N-1] \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \phi_{M-1}^*[0] & \phi_{M-1}^*[1] & \phi_{M-1}^*[2] & \cdots & \phi_{M-1}^*[N-1] \end{bmatrix}}_{\mathbf{A}} \underbrace{\begin{bmatrix} x[0] \\ x[1] \\ x[2] \\ \vdots \\ x[N-1] \end{bmatrix}}_{\mathbf{X}}$$

$$y[m] = \langle x[n], \phi_m[n] \rangle = \sum_{n=0}^{N-1} x[n]\phi_m^*[n]$$

inner product

Orthogonal: $\langle \phi_k[n], \phi_h[n] \rangle = \sum_{n=0}^{N-1} \phi_k[n]\phi_h^*[n] = 0$ **when** $\boldsymbol{k \neq h}$

orthogonal transforms 的例子：

- discrete Fourier transform
- discrete cosine, sine, Hartley transforms
- Walsh Transform, Haar Transform
- discrete Legendre transform
- discrete orthogonal polynomial transforms

  Hahn, Meixner, Krawtchouk, Charlier

為什麼在信號處理上，我們經常用 orthogonal transform?

Orthogonal transform 最大的好處何在？

- If partial terms are used for reconstruction

for orthogonal case,

perfect reconstruction: $x[n]=\sum_{m=0}^{N-1} C_m^{-1} y[m]\phi_m[n]$

partial reconstruction: $x_K[n]=\sum_{m=0}^{K-1} C_m^{-1} y[m]\phi_m[n] \qquad K<N$

reconstruction error of partial reconstruction

$$\left\|x[n]-x_K[n]\right\|^2=\sum_{n=0}^{N-1}\left\|\sum_{m=K}^{N-1} C_m^{-1} y[m]\phi_m[n]\right\|^2$$

$$=\sum_{n=0}^{N-1}\sum_{m=K}^{N-1} C_m^{-1} y[m]\phi_m[n]\sum_{m_1=K}^{N-1} C_{m_1}^{-1} y^*[m_1]\phi_{m_1}^*[n]$$

$$=\sum_{m=K}^{N-1}\sum_{m_1=K}^{N-1} C_m^{-1} y[m]C_{m_1}^{-1} y^*[m_1]\sum_{n=0}^{N-1}\phi_m[n]\phi_{m_1}^*[n]$$

$$=\sum_{m=K}^{N-1}\sum_{m_1=K}^{N-1} C_m^{-1} y[m]C_{m_1}^{-1} y^*[m_1]C_m\delta[m-m_1]=\sum_{m=K}^{N-1} C_m^{-1}\left|y[m]\right|^2$$

由於 $C_m^{-1}\left|y[m]\right|^2$ 一定是正的，可以保證 <u>K 越大, reconstruction error 越小</u>

For non-orthogonal case,

perfect reconstruction: $x[n] = \sum_{m=0}^{N-1} B[n,m]\,y[m]$ $\quad$ $\mathbf{B} = \mathbf{A}^{-1}$

partial reconstruction: $x_K[n] = \sum_{m=0}^{K-1} B[n,m]\,y[m]$ $\quad$ $K < N$

reconstruction error of partial reconstruction

$$\left\|x[n] - x_K[n]\right\|^2 = \sum_{n=0}^{N-1}\left\|\sum_{m=K}^{N-1} B[n,m]\,y[m]\right\|^2$$

$$= \sum_{n=0}^{N-1}\sum_{m=K}^{N-1} B[n,m]\,y[m]\sum_{m_1=K}^{N-1} B^*[n,m_1]\,y^*[m_1]$$

$$= \sum_{m=K}^{N-1}\sum_{m_1=K}^{N-1} y[m]\,y^*[m_1]\sum_{n=0}^{N-1} B[n,m]\,B^*[n,m_1]$$

由於 $y[m]\,y^*[m_1]\sum_{n=0}^{N-1} B[n,m]\,B^*[n,m_1]$ 不一定是正的，

無法保證 K 越大, reconstruction error 越小

◉ 14-B Frequency and Time Division Multiplexing

傳統 Digital Modulation and Multiplexing：使用 Fourier transform

- **Frequency-Division Multiplexing (FDM)**

$$z(t) = \sum_{n=0}^{N-1} X_n \exp\left(j2\pi f_n t\right)$$

$X_n = 0$ or 1

X_n can also be set to be -1 or 1

When (1) $t \in [0, T]$ (2) $f_n = n/T$

$$z(t) = \sum_{n=0}^{N-1} X_n \exp\left(j\frac{2\pi nt}{T}\right)$$

it becomes the **orthogonal frequency-division multiplexing (OFDM).**

Furthermore, if the time-axis is also sampled

$t \in [0,T]$
sampling for t-axis

$$t = mT/N, \qquad m = 0,\ 1,\ 2,\ \ldots..,\ N-1$$

$$z\left(m\frac{T}{N}\right) = \sum_{n=0}^{N-1} X_n \exp\left(j\frac{2\pi nm}{N}\right)$$

then the OFDM is equivalent to the transform matrix of the inverse discrete Fourier transform (IDFT), which is one of the discrete orthogonal transform.

Modulation: $$Y_m = z\left(m\frac{T}{N}\right) = \sum_{m=0}^{N-1} A[m,n]X_n$$

$$\mathbf{A} = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & e^{j\frac{2\pi}{N}} & e^{j\frac{4\pi}{N}} & \cdots & e^{j\frac{2(N-1)\pi}{N}} \\ 1 & e^{j\frac{4\pi}{N}} & e^{j\frac{8\pi}{N}} & \cdots & e^{j\frac{4(N-1)\pi}{N}} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & e^{j\frac{2(N-1)\pi}{N}} & e^{j\frac{4(N-1)\pi}{N}} & \cdots & e^{j\frac{2(N-1)(N-1)\pi}{N}} \end{bmatrix}$$

Modulation: $$Y_m = \sum_{m=0}^{N-1} A[m,n] X_n$$

Demodulation: $$X_n = \frac{1}{N} \sum_{m=0}^{N-1} A^*[m,n] Y_m$$

Example: $N = 8$

$$X_n = [1, 0, 1, 1, 0, 0, 1, 1] \qquad (n = 0 \sim 7)$$

• Time-Division Multiplexing (TDM)

$$z(0) = X_0, \quad z\left(\frac{T}{N}\right) = X_1, \quad z\left(2\frac{T}{N}\right) = X_2, \quad \cdots\cdots, \quad z\left((N-1)\frac{T}{N}\right) = X_{N-1}$$

$$y(m) = z\left(m\frac{T}{N}\right) = \sum_{m=0}^{N-1} A[m,n]X_n$$

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix}$$

(also a discrete orthogonal transform)

思考：

既然 time-division multiplexing 那麼簡單

那為什麼要使用 frequency-division multiplexing
和 orthogonal frequency-division multiplexing (OFDM)?

◉ 14-C Code Division Multiple Access (CDMA)

除了 frequency-division multiplexing 和 time-division multiplexing，是否還有其他 multiplexing 的方式？

使用其他的 orthogonal transforms
即 code division multiple access (CDMA)

CDMA is an important topic in spread spectrum communication

參考資料

[1] M. A. Abu-Rgheff, *Introduction to CDMA Wireless Communications*, Academic, London, 2007

[2] 邱國書, 陳立民譯, "CDMA 展頻通訊原理", 五南, 台北, 2002.

CDMA 最常使用的 orthogonal transform 為 Walsh transform

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ y_7 \\ y_8 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \\ 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \end{bmatrix} \begin{array}{c} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \\ x_8 \end{bmatrix} \end{array} \begin{array}{l} \leftarrow \text{channel 1} \\ \leftarrow \text{channel 2} \\ \leftarrow \text{channel 3} \\ \leftarrow \text{channel 4} \\ \leftarrow \text{channel 5} \\ \leftarrow \text{channel 6} \\ \leftarrow \text{channel 7} \\ \leftarrow \text{channel 8} \end{array}$$

channel 1

當有兩組人在同一個房間裡交談 (A 和B交談)，(C 和D交談) ，
如何才能夠彼此不互相干擾？

(1) Different Time

(2) Different Tone

(3) Different Language

CDMA 分為：

(1) Orthogonal Type　　(2) Pseudorandom Sequence Type

Orthogonal Type 的例子：兩組資料 [1, 0, 1]　　[1, 1, 0]

(1) 將 0 變為 −1　　[1, −1, 1]　　[1, 1, −1]

(2) 1, −1, 1 modulated by [1, 1, 1, 1, 1, 1, 1, 1] (channel 1)

→ [1, 1, 1, 1, 1, 1, 1, 1, -1, -1, -1, -1, -1, -1, -1, -1, 1, 1, 1, 1, 1, 1, 1, 1]

1, 1, −1 modulated by [1, 1, 1, 1, -1, -1, -1, -1] (channel 2)

→ [1, 1, 1, 1, -1, -1, -1, -1, 1, 1, 1, 1, -1, -1, -1, -1, -1, -1, -1, -1, 1, 1, 1, 1]

(3) 相合

[2, 2, 2, 2, 0, 0, 0, 0, 0, 0, 0, 0, -2, -2, -2, -2, 0, 0, 0, 0, 2, 2, 2, 2] demodulation

[2, 2, 2, 2, 0, 0, 0, 0, 0, 0, 0, 0, -2, -2, -2, -2, 0, 0, 0, 0, 2, 2, 2, 2]

[1, 1, 1, 1, 1, 1, 1, 1]   [1, 1, 1, 1, 1, 1, 1,1]

[1, 1, 1, 1, 1, 1, 1, 1]

內積＝8

注意：

(1) 使用 N-point Walsh transform 時，總共可以有 N 個 channels

(2) 除了 Walsh transform 以外，其他的 orthogonal transform 也可以使用

(3) 使用 Walsh transform 的好處

- Orthogonal Transform 共通的問題:　需要同步 synchronization

$\mathbf{R_1} = [\ 1,\ \ 1,\ \ 1,\ \ 1,\ \ 1,\ \ 1,\ \ 1,\ \ 1]$

$\mathbf{R_2} = [\ 1,\ \ 1,\ \ 1,\ \ 1, -1, -1, -1, -1]$

$\mathbf{R_5} = [\ 1, -1, -1,\ \ 1,\ \ 1, -1, -1,\ \ 1]$

$\mathbf{R_8} = [\ 1, -1,\ \ 1, -1,\ \ 1, -1,\ \ 1, -1]$

但是某些 basis, 就算不同步也近似 orthogonal

$<\mathbf{R_1}[n], \mathbf{R_1}[n]> = 8, \quad <\mathbf{R_1}[n], \mathbf{R_k}[n]> = 0$ if $k \neq 1$

$<\mathbf{R_1}[n], \mathbf{R_k}[n-1]> = 2$ or 0　 if $k \neq 1$.

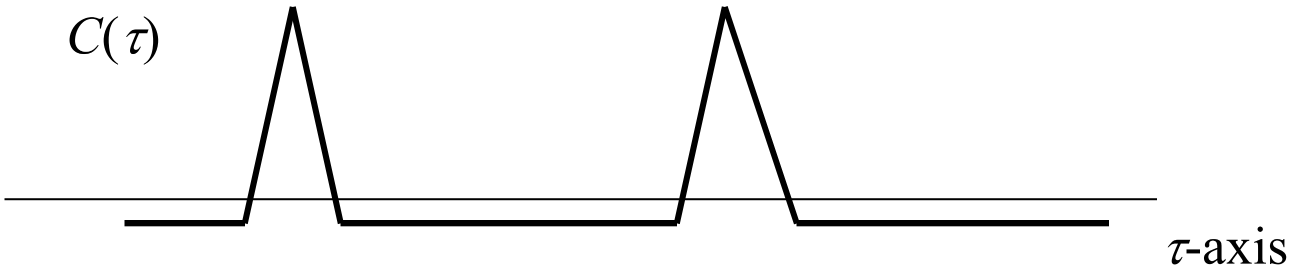
這裡的shift為circular shift

Pseudorandom Sequence Type

不為 orthogonal，capacity 較少

但是不需要同步 (asynchronous)

Pseudorandom Sequence 之間的 correlation



$b_1 p(t+\tau_1) + b_2 p(t+\tau_2)$

recovered: $\int \left(b_1 p(t+\tau_1) + b_2 p(t+\tau_2)\right) p(t+\tau_1)\, dt = b_1 C(0) + b_2 C(\tau_2 - \tau_1) \approx b_1$

(若 $C(0) = 1$, $C(\tau_2 - \tau_1) \approx 0$)

τ_1, τ_2 不必一致

CDMA 的優點：

(1) 運算量相對於 frequency division multiplexing 減少很多

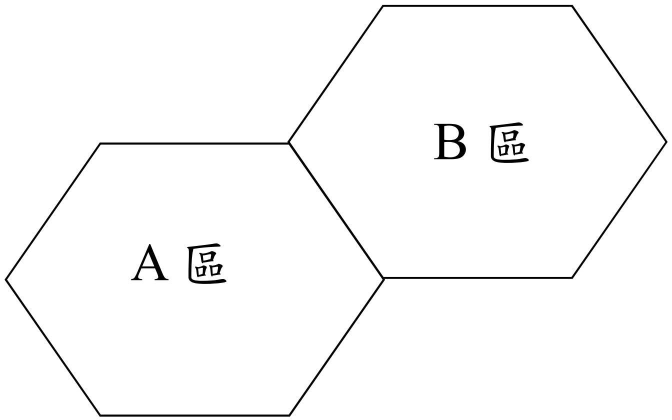
(2) 可以減少 noise 及 interference的影響

(3) 可以應用在保密和安全傳輸上

(4) 就算只接收部分的信號，也有可能把原來的信號 recover 回來

(5) 相鄰的區域的干擾問題可以減少

相鄰的區域，使用差距最大的「語言」，則干擾最少



假設 A 區使用的 orthogonal basis 為 $\phi_k[n]$, $k = 0, 1, 2, \ldots, N-1$

B 區使用的 orthogonal basis 為 $\mu_h[n]$, $h = 0, 1, 2, \ldots, N-1$

設法使 $\max\left(\left|\dfrac{\langle \phi_k[n], \mu_h[n] \rangle}{\langle \phi_k[n], \phi_h[n] \rangle}\right|\right)$ 為最小

$k = 0, 1, 2, \ldots, N-1$, $h = 0, 1, 2, \ldots, N-1$

附錄十五 Conventional Machine Learning Methods

雖然近年來有許多準確度很高的 advanced machine learning 的方法，但一些較傳統的方法，由於架構簡單，不需大量且價格昂貴的 GPU，且 training time 較少，仍被產業界廣泛的使用

Conventional machine learning methods for classification:

(i) Nearest Neighbor, *k* Nearest Neighbors (*kNN*)

(ii) Support Vector Machine (SVM)

(iii) Neural Network

Deep learning and the convolutional neural network (CNN) can be viewed as the improved versions of the neural network.

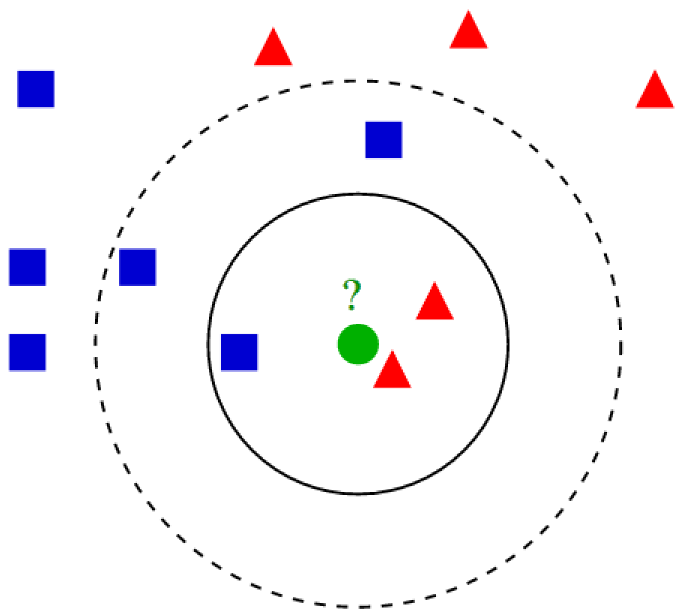
Andrew Ng, *Machine Learning*, Stanford Cousera, available from https:// www.coursera.org/course/ml

A. *k* Nearest Neighbors (*kNN*)

k Nearest Neighbors (*kNN*) (最簡單的方法)

選擇 *k* 個在特徵平面上最近的點

再投票



例子：左圖當中，input (綠色) 要被分類為紅色還是藍色？

- 3 Nearest Neighbors (solid lines)

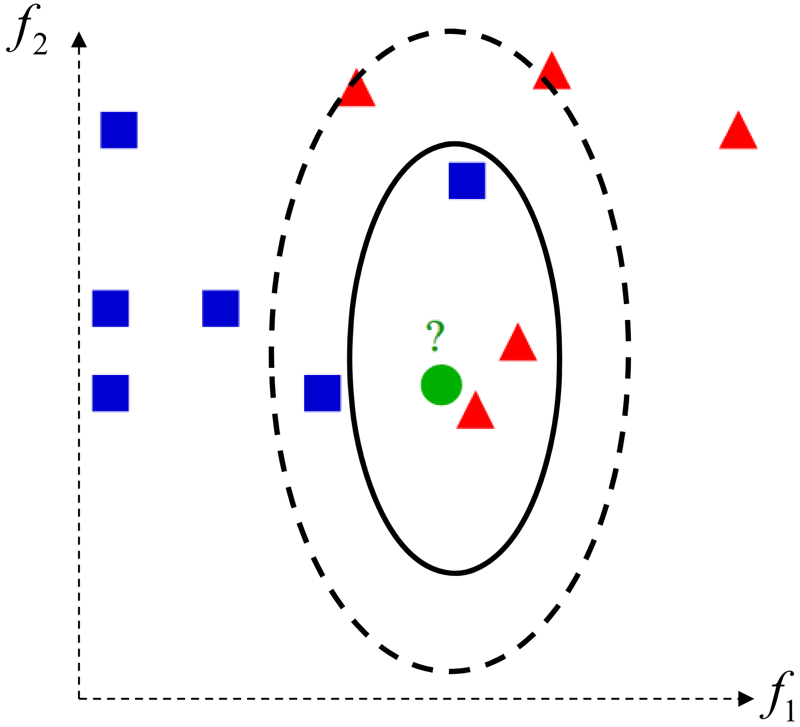
藍色1票，紅色2票，判斷結果為紅色

- 5 Nearest Neighbors (dash lines)

藍色3票，紅色2票，判斷結果為藍色

Each axis corresponds to a feature.

Weighted k Nearest Neighbors (W-kNN)



Some features are more important than others.

Assign larger weights for the features of more importance.

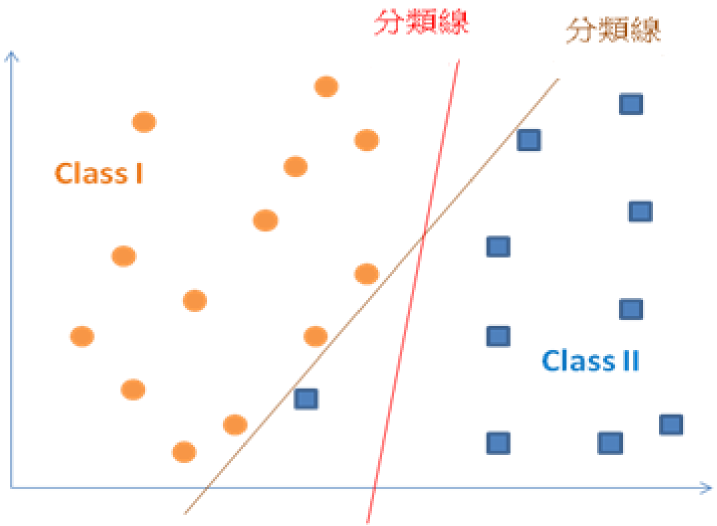
$$distance = \sqrt{w_1(f_1(a)-f_1(b))^2 + w_2(f_2(a)-f_2(b))^2 + \cdots + w_m(f_m(a)-f_m(b))^2}$$

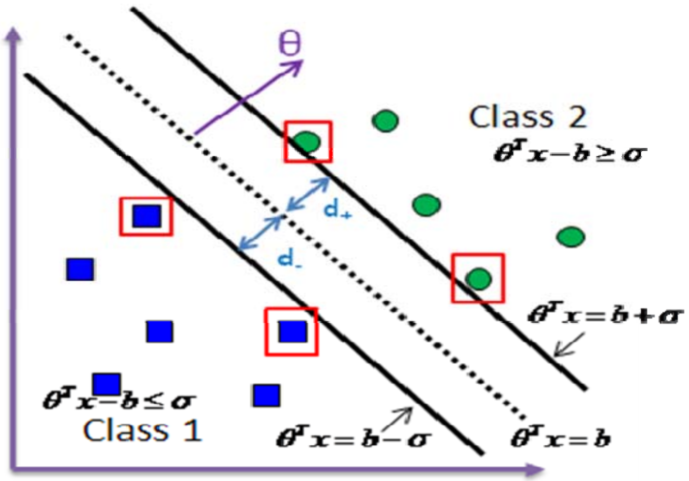
$w_1, w_2, \cdots, w_m$: weights

B. SVM (support vector machine)

SVM (support vector machine，支持向量機):

找出適當的分界線，將資料分為 Class 1, Class 2 兩群





Nonlinear SVM:

$\mathbf{x}$: inputs, $\mathbf{L_1}, \mathbf{L_2}, \mathbf{L_3}, \ldots., \mathbf{L_m}$: 標竿

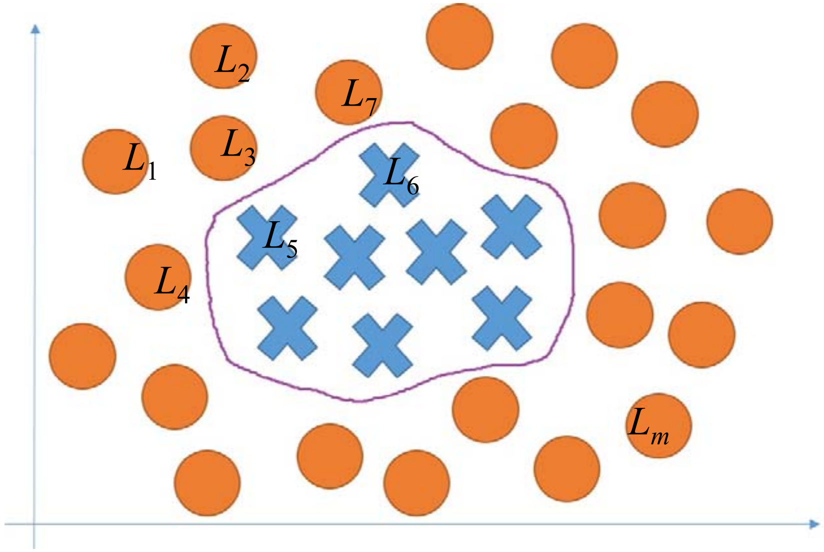
f_1 = similarity($\mathbf{x}$, $\mathbf{L_1}$)

f_2 = similarity($\mathbf{x}$, $\mathbf{L_2}$)

f_3 = similarity($\mathbf{x}$, $\mathbf{L_3}$)

$\vdots$

f_m = similarity($\mathbf{x}$, $\mathbf{L_m}$)



以 $f_1, f_2, f_3, \ldots., f_m$ 為 features

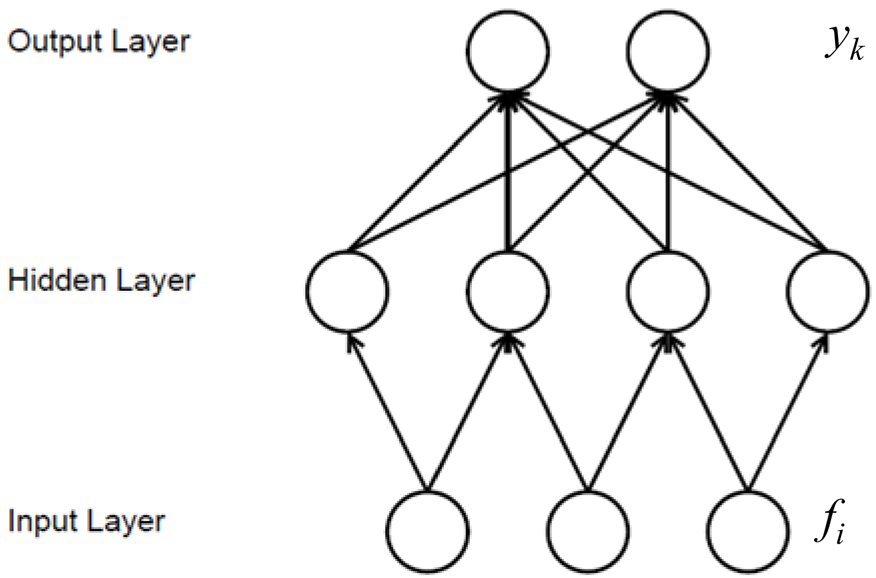
SVM 程式下載

C. C. Chang and C. J. Lin, "LIBSVM -- A library for SVM," available from http:// www.csie.ntu.edu.tw/~cjlin/libsvm/

C. Neural Network

Neural Network:

2-Layer Neural Network



判斷結果

特徵 (features)

$$x_j = g\left(w_{0,j} + \sum_i w_{i,j} f_i \right)$$

$$y_k = g\left(v_{0,k} + \sum_j v_{j,k} x_j \right)$$

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

兩層以上的 Neural Network: Deep Learning

期末的勉勵

- 人生難免會有挫折，最重要的是，**我們面對挫折的態度是什麼**

- 長遠的願景要美麗，短期的目標要務實

祝各位同學暑假愉快！

各位同學在研究上或工作上，有任何和 digital signal processing 或 time frequency analysis 方面的問題，歡迎找我來一起討論。