Machine Learning Homework 5

資工碩一 r09922055 陳柏妤

1. **(d)**

$$\phi(X) = \begin{bmatrix} 1 & -2 & 4 \\ 1 & 0 & 0 \\ 1 & 2 & 4 \end{bmatrix}，y = \begin{bmatrix} -1 \\ +1 \\ -1 \end{bmatrix}$$

$$\begin{cases} 2w_1 - 4w_2 - b \geq 1 \\ b \geq 1 \\ -2w_1 - 4w_2 - b \geq 1 \end{cases} \Rightarrow \begin{cases} b \geq 1 \\ (i) + (iii) \Rightarrow 8w_2 + 2b \leq -2 \Rightarrow \\ (i) - (iii) \Rightarrow 4w_1 \geq 0 \end{cases} \begin{cases} w_2 \leq -\frac{1}{2} \\ w_1 \geq 0 \end{cases}$$

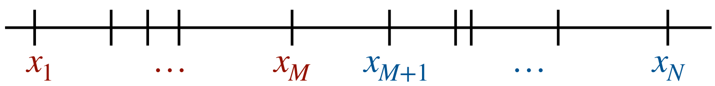
$\frac{1}{2}w^T w \geq \frac{1}{8}$，$(w_1 = 0,\ w_2 = \frac{-1}{2},\ b = 1)$ at lower bound

2. **(b)**

承上題，$\phi(x) = [1,\ x,\ x^2]^T = [z_0,\ z_1,\ z_2]$，$\frac{-1}{2}z_2 + 1 = 0$ is the fattest boundary which separate $\{(\phi(x_n),\ y_n)\}_{n=1}^{3}$, and all three examples are support vector candidates.

$$distance(z_n,\ b,\ w) = \frac{1}{\|w\|} y_n(w^T z_n + b) = 2 \cdot ((-1) \cdot (-2 + 1)) = 2$$

3. **(e)**



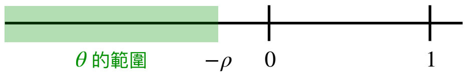
能夠分開 negative examples 和 positive examples 的最胖的 boundary 位於 x_M 和 x_{M+1} 之間，此時 $margin = \frac{1}{2}(x_{M+1} - x_M)$

4. **(a)**

a. (+1, +1) 出現的機率：

因為 x_1, x_2 介於 0 到 1 之間，$h(x) = sign(x - \theta)$ ，其中 $\theta \leq -\rho$ 使得 $margin \geq \rho$。

$\Rightarrow$ 必可產生 (+1, +1) 的 dichotomy



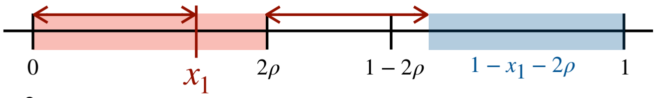
b. (-1, -1) 出現的機率：

$h(x) = sign(x - \theta)$，其中 $\theta > max(x_1, x_2)$ 時就可產生 (-1, -1) 的 dichotomy

（即使 $margin < \rho$ return $h(x) = -1$ 一樣可以產生 (-1, -1) 的 dichotomy）

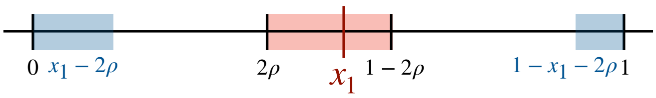
c. (+1, -1) 或 (-1, +1) 兩種 dichotomy 出現的機率：考慮以下三種 x_1 和 x_2 相距 2ρ 的情況

・$0 \leq x_1 \leq 2\rho$，此時 x_2 可能的區域位於 $1 - x_1 - 2\rho$ 的範圍



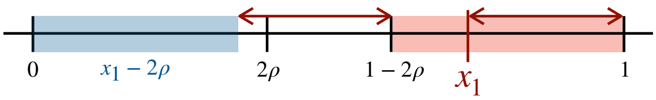
$$\int_0^{2\rho} 1 - x_1 - 2\rho \ dx_1 = x_1 - \frac{1}{2}x_1^2 - 2\rho x_1 |_0^{2\rho} = 2\rho - 2\rho^2 - 4\rho^2 = 2\rho - 6\rho^2$$

・$2\rho \leq x_1 \leq 1 - 2\rho$，此時 x_2 可能的區域位於 $(x_1 - 2\rho) + (1 - x_1 - 2\rho) = 1 - 4\rho$ 的範圍



$$\int_{2\rho}^{1-2\rho} 1 - 4\rho \ dx_1 = x_1 - 4\rho x_1 |_{2\rho}^{1-2\rho} = 1 - 8\rho + 16\rho^2$$

・$1 - 2\rho \leq x_1 \leq 1$，此時 x_2 可能的區域位於 $x_1 - 2\rho$ 的範圍



$$\int_{1-2\rho}^{1} x_1 - 2\rho \ dx_1 = \frac{1}{2}x_1^2 - 2\rho x_1 |_{1-2\rho}^{1} = (\frac{1}{2} - 2\rho) - (\frac{1}{2} - 4\rho + 6\rho^2) = 2\rho - 6\rho^2$$

$$\mathbb{E}(\# \ dichotomies) = 1 \cdot 1 + 1 \cdot 1 + 2 \cdot ((2\rho - 6\rho^2) + (1 - 8\rho + 16\rho^2) + (2\rho - 6\rho^2))$$
$$= 2 + 2 \cdot (1 - 4\rho + 4\rho^2) = 2 + 2 \cdot (1 - 2\rho)^2$$

5. **(c)**

by L202 p.9，Lagrange Dual 可化簡為：

$$\max_{all\ \alpha_n \ge 0,\ \sum y_n \alpha_n = 0} (\min_{b,\ w} \frac{1}{2} w^T w + \sum_{n=1}^{N} \alpha_n (1 - y_n (w^T z_n)))$$，將此式套入 uneven-margin SVM

得到：

$$\max_{all\ \alpha_n \ge 0,\ \sum y_n \alpha_n = 0} (\min_{b,\ w} \frac{1}{2} w^T w + \sum_{n=1}^{N} \alpha_n (\rho_+ - (y_n w^T z_n))[y_n = +1] + \sum_{n=1}^{N} \alpha_n (\rho_- - (y_n w^T z_n))[y_n = -1])$$

$$\Rightarrow \max_{all\ \alpha_n \ge 0,\ \sum y_n \alpha_n = 0} (\min_{b,\ w} \frac{1}{2} w^T w - \sum_{n=1}^{N} \alpha_n y_n w^T z_n + \sum_{n=1}^{N} (\alpha_n \rho_+)[y_n = +1] + \sum_{n=1}^{N} (\alpha_n \rho_-)[y_n = -1])$$

求最小值，對 w_i 做偏微分，最小值必落在 $w = \sum_{n=1}^{N} \alpha_n y_n z_n$ 時

$$\Rightarrow \max_{all\ \alpha_n \ge 0,\ \sum y_n \alpha_n = 0,\ w = \sum \alpha_n y_n z_n} (\frac{-1}{2} w^T w + \sum_{n=1}^{N} (\alpha_n \rho_+)[y_n = +1] + \sum_{n=1}^{N} (\alpha_n \rho_-)[y_n = -1])$$

$$\Rightarrow \min_{all\ \alpha_n \ge 0,\ \sum y_n \alpha_n = 0} (\frac{1}{2} w^T w - \sum_{n=1}^{N} (\alpha_n \rho_+)[y_n = +1] - \sum_{n=1}^{N} (\alpha_n \rho_-)[y_n = -1])$$

$$(w^T w = \sum_{n=1}^{N} \sum_{m=1}^{N} \alpha_n \alpha_m y_n y_m x_n^T x_m)$$

6. **(e)**

w^* 為原本 SVM 的最佳解、w' 為 uneven-SVM 的最佳解

・證：$\dfrac{\|w'\|}{\|w^*\|} = \dfrac{\rho_+ + \rho_-}{2}$

by margin 定義，$margin(b, w^*) = \dfrac{1}{\|w^*\|}$ （講義公式）

when $y_n = +1$，希望 $\min\limits_{n=1,\ldots,N,\ y_n=+1} y_n(w^T x_n + b) = \rho_+ \Rightarrow margin^+(b, w') = \dfrac{\rho_+}{\|w'\|}$

when $y_n = -1$，希望 $\min\limits_{n=1,\ldots,N,\ y_n=-1} y_n(w^T x_n + b) = \rho_- \Rightarrow margin^-(b, w') = \dfrac{\rho_-}{\|w'\|}$

設轉換前的 SVs 和轉換後的 SVs 是相同的：

$$\Rightarrow 2 \cdot margin(b, w^*) = margin^+(b, w') + margin^-(b, w')$$

$$\Rightarrow \frac{2}{\|w^*\|} = \frac{\rho_+}{\|w'\|} + \frac{\rho_-}{\|w'\|} = \frac{\rho_+ + \rho_-}{\|w'\|} \Rightarrow \frac{\|w'\|}{\|w^*\|} = \frac{\rho_+ + \rho_-}{2}$$

・證：$\alpha'_n = \alpha^*_n(\dfrac{\rho_+ + \rho_-}{2})$

$$\begin{cases} w^* = \sum_{n=1}^N \alpha^*_n y_n x_n \\ w' = \sum_{n=1}^N \alpha'_n y_n x_n \end{cases} \Rightarrow w' = (\frac{\rho_+ + \rho_-}{2}) w^* = \sum_{n=1}^N (\frac{\rho_+ + \rho_-}{2}) \alpha^*_n y_n x_n = \sum_{n=1}^N \alpha'_n y_n x_n$$

$$\Rightarrow \alpha'_n = \alpha^*_n(\frac{\rho_+ + \rho_-}{2})$$

・證：$b' = (\dfrac{\rho_+ + \rho_-}{2})(b^*) + (\dfrac{\rho_+ - \rho_-}{2})$

when $y_n = +1$，$\begin{cases} y_n(w^{*T} x_n + b^*) = 1 \\ y_n(w'^T x_n + b') = \rho_+ \end{cases} \Rightarrow \begin{cases} w^{*T} x_n + b^* = 1 \\ w'^T x_n + b' = \rho_+ \end{cases}$

$$w'^T x_n + b' = (\frac{\rho_+ + \rho_-}{2}) w^{*T} x_n + b' = (\frac{\rho_+ + \rho_-}{2})(1 - b^*) + b' = \rho_+$$

when $y_n = -1$，$\begin{cases} y_n(w^{*T} x_n + b^*) = 1 \\ y_n(w'^T x_n + b') = \rho_- \end{cases} \Rightarrow \begin{cases} w^{*T} x_n + b^* = -1 \\ w'^T x_n + b' = -\rho_- \end{cases}$

$$w'^T x_n + b' = (\frac{\rho_+ + \rho_-}{2}) w^{*T} x_n + b' = (\frac{\rho_+ + \rho_-}{2})(-1 - b^*) + b' = -\rho_-$$

綜合藍色和紅色的式子 $\Rightarrow b' = (\dfrac{\rho_+ + \rho_-}{2})(b^*) + (\dfrac{\rho_+ - \rho_-}{2})$

by “complementary slackness”, when optimal：

$$\Rightarrow \alpha_n(\rho_+ - y_n(w^T x_n + b))\frac{y_n + 1}{2} + \alpha_n(\rho_- - y_n(w^T x_n + b))\frac{-y_n + 1}{2} = 0$$

$$\alpha^*(1 - y_n(w^T x_n + b)) = 0$$

因為 KKT condition 是最佳解的必要且充要條件 $\Rightarrow$ 要證明 w'、α'、b' 符合 KKT condition

- 證 dual feasible：$\alpha'_n \geq 0$

  已知 $\alpha^*_n \geq 0$，$\alpha'_n = \alpha^*_n(\dfrac{\rho_+ + \rho_-}{2})$

  又 $\dfrac{\rho_+ + \rho_-}{2} \geq 0 \Rightarrow \alpha'_n \geq 0$

- 證 dual inner optimal：$\sum y_n \alpha'_n = 0$；$w' = \sum \alpha'_n y_n x_n$

  1. $\sum y_n \alpha'_n = 0$

     已知 $\sum y_n \alpha^*_n = 0 \Rightarrow \sum y_n \alpha'_n = \sum y_n (\dfrac{\rho_+ + \rho_-}{2})\alpha^*_n = 0$

  2. $w' = \sum \alpha'_n y_n x_n$

     $w' = (\dfrac{\rho_+ + \rho_-}{2})w^* = (\dfrac{\rho_+ + \rho_-}{2})\sum \alpha^*_n y_n x_n = \sum \alpha'_n y_n x_n$

- 證 primal feasible：$\begin{cases} y_n(w'^T x_n + b') \geq \rho_+, \ when \ \ y_n = +1 \\ y_n(w'^T x_n + b') \geq \rho_-, \ when \ \ y_n = -1 \end{cases}$

  1. $y_n(w'^T x_n + b') \geq \rho_+, \ when \ \ y_n = +1 \Rightarrow$ 證：$w'^T x_n + b' \geq \rho_+$

     $w'^T x_n + b' = (\dfrac{\rho_+ + \rho_-}{2})w^{*T} x_n + \rho_+ - (\dfrac{\rho_+ + \rho_-}{2})(1 - b^*)$

     $= (\dfrac{\rho_+ + \rho_-}{2})(1 - b^*) + \rho_+ - (\dfrac{\rho_+ + \rho_-}{2})(1 - b^*) = \rho_+ \geq \rho_+$

  2. $y_n(w'^T x_n + b') \geq \rho_-, \ when \ \ y_n = -1 \Rightarrow$ 證：$w'^T x_n + b' \leq -\rho_-$

     $w'^T x_n + b' = (\dfrac{\rho_+ + \rho_-}{2})w^{*T} x_n + (-\rho_- - (\dfrac{\rho_+ + \rho_-}{2})(-1 - b^*))$

     $(\dfrac{\rho_+ + \rho_-}{2})(-1 - b^*) - \rho_- - (\dfrac{\rho_+ + \rho_-}{2})(-1 - b^*) = -\rho_- \leq -\rho_-$

- 證 primal-inner optimal：

  $\alpha'_n(\rho_+ - y_n(w'^T x_n + b'))\dfrac{y_n + 1}{2} + \alpha'_n(\rho_- - y_n(w'^T x_n + b'))\dfrac{-y_n + 1}{2} = 0$

  By 前一點，

  1. 當 $y_n = +1 \Rightarrow y_n(w'^T x_n + b') = \rho_+$、$\dfrac{-y_n + 1}{2} = 0$

     $\Rightarrow \alpha'_n(\rho_+ - y_n(w'^T x_n + b'))\dfrac{y_n + 1}{2} + \alpha'_n(\rho_- - y_n(w'^T x_n + b'))\dfrac{-y_n + 1}{2} = 0$

  2. 當 $y_n = -1 \Rightarrow y_n(w'^T x_n + b') = \rho_-$、$\dfrac{y_n + 1}{2} = 0$

     $\Rightarrow \alpha'_n(\rho_+ - y_n(w'^T x_n + b'))\dfrac{y_n + 1}{2} + \alpha'_n(\rho_- - y_n(w'^T x_n + b'))\dfrac{-y_n + 1}{2} = 0$

w'、α'、b' 確實是 uneven-SVM 的最佳解

7. **(d)**

$k_{ij} = K(x_i, x_j)$ *the matrix K*

$$= \begin{bmatrix} K(x_1,x_1) & K(x_1,x_2) & \dots & K(x_1,x_N) \\ K(x_2,x_1) & K(x_2,x_2) & \dots & K(x_2,x_N) \\ \vdots & \vdots & \vdots & \vdots \\ K(x_N,x_1) & K(x_N,x_2) & \dots & K(x_N,x_N) \end{bmatrix} = \begin{bmatrix} \phi(x_1)^T\phi(x_1) & \phi(x_1)^T\phi(x_2) & \dots & \phi(x_1)^T\phi(x_N) \\ \phi(x_2)^T\phi(x_1) & \phi(x_2)^T\phi(x_2) & \dots & \phi(x_2)^T\phi(x_N) \\ \vdots & \vdots & \vdots & \vdots \\ \phi(x_N)^T\phi(x_1) & \phi(x_N)^T\phi(x_2) & \dots & \phi(x_N)^T\phi(x_N) \end{bmatrix}$$

$K' = log_2 K(x, x')$ *the matrix K'*

$$= \begin{bmatrix} log_2K(x_1,x_1) & log_2K(x_1,x_2) & \dots & log_2K(x_1,x_N) \\ log_2K(x_2,x_1) & log_2K(x_2,x_2) & \dots & log_2K(x_2,x_N) \\ \vdots & \vdots & \vdots & \vdots \\ log_2K(x_N,x_1) & log_2K(x_N,x_2) & \dots & log_2K(x_N,x_N) \end{bmatrix}$$

・半正定矩陣定義：$for\ any\ x \in \mathbb{R}^N,\ x^T A x \geq 0$

・又 $0 \leq K(x, x') < 2 \Rightarrow -\infty \leq log_2 K(x, x') < 1$

$\Rightarrow matrix\ K'$ 的每個 element 都介於 $-\infty$ 到 1 之間

設有一 $matrix\ K'$ 的總和為負，即：$\sum_{n=1}^{N}\sum_{m=1}^{N} log_2 K(x_n, x_m) < 0$（以下稱此 matrix 為 K'^{-}）

則當 $x = 1^N$ 時，$x^T K'^{-} x = \mathbf{1}^T K'^{-} \mathbf{1} < 0$（非半正定矩陣 $\Rightarrow$ invalid）

8. **(c)**

$\|\phi(x) - \phi(x')\|^2$

$= (\phi(x) - \phi(x'))^T(\phi(x) - \phi(x')) = \phi(x)^2 - 2\phi(x)^T\phi(x') + \phi(x')^2$

藍色部分：by gaussian kernel 定義，$\phi(x)^2 = K(x,\ x) = exp(-\gamma\|x - x\|^2) = exp(0) = 1$

紅色部分：by gaussian kernel 定義，$-2\phi(x)^T\phi(x') = -2exp(-\gamma\|x - x'\|^2) \leq 0$

因為 $-\gamma\|x - x'\|^2$ 的範圍可能是 $-\infty \sim 0$

$\Rightarrow exp(-\gamma\|x - x'\|^2)$ 的最小值是 0 $\Rightarrow -2exp(-\gamma\|x - x'\|^2)$ 的最大值是 0

$\phi(x)^2 - 2\phi(x)^T\phi(x') + \phi(x')^2 \leq 1 + 0 + 1 = 2$

9. **(d)**

$E_{in}=0$ 代表 $\forall m,\ y_m \cdot h_{1,0}(x_m) > 0$（所有點都預測正確）

$$\Rightarrow y_m(\sum_{n=1}^{N} y_n K(x_n, x_m)) > 0$$

$$\Rightarrow y_m(\sum_{n=1}^{N} y_n\ exp(-\gamma\|x_n - x_m\|^2)) > 0$$

$$\Rightarrow y_m(y_m\ exp(-\gamma\|x_m - x_m\|^2) + \sum_{n=1,n\neq m}^{N} y_n\ exp(-\gamma\|x_n - x_m\|^2)) > 0$$

$$\Rightarrow 1 + \sum_{n=1,n\neq m}^{N} y_m y_n\ exp(-\gamma\|x_n - x_m\|^2) > 0$$

無論 $y_m y_n$ 和 $\|x_n - x_m\|^2$ 是多少，γ 都要讓 $1 + \sum_{n=1,n\neq m}^{N} y_m y_n\ exp(-\gamma\|x_n - x_m\|^2)$ 大於零

最壞的情況：當 $y_m y_n = -1$、$\|x_n - x_m\|^2 = \epsilon^2$ 時

（會使得 $\sum_{n=1,n\neq m}^{N} y_m y_n\ exp(-\gamma\|x_n - x_m\|^2)$最負）

want：$1 + \sum_{n=1,n\neq m}^{N} - exp(-\gamma\epsilon^2) > 0$

$$\Rightarrow 1 - (N-1)exp(-\gamma\epsilon^2) > 0 \Rightarrow \frac{1}{N-1} > exp(-\gamma\epsilon^2) \Rightarrow ln(\frac{1}{N-1}) > -\gamma\epsilon^2$$

$$\Rightarrow \gamma\epsilon^2 > ln(N-1) \Rightarrow \gamma > \frac{ln(N-1)}{\epsilon^2}$$

10. **(c)**

$w_{t+1} \leftarrow w_t + y_{n(t)}\phi(x_{n(t)})$

$$\Rightarrow \sum_{n=1}^{N} \alpha_{t+1,n}\phi(x_n) \leftarrow \sum_{n=1}^{N} \alpha_{t,n}\phi(x_n) + y_{n(t)}\phi(x_{n(t)}) = (\sum_{n=1}^{N} \alpha_{t,n} + y_{n(t)})(\phi(x_n))$$

$$\Rightarrow \alpha_{t+1,n(t)} \leftarrow \alpha_{t,n(t)} + y_{n(t)}$$

11. **(a)**

已知 $w_t = \sum_{n=1}^{N} \alpha_{t,n}\phi(x_n)$

$$w_t^T\phi(x) = \sum_{n=1}^{N} \alpha_{t,n}\phi(x_n)^T\phi(x) = \sum_{n=1}^{N} \alpha_{t,n}K(x_n,\ x)$$

12. **(b)**

by complementary slackness，$\begin{cases} \alpha_n(1-\xi_n-y_n(w^Tx_n+b))=0 \\ (C-\alpha_n)\xi_n=0 \end{cases}$

又已知 $\xi_n \geq 0$

$\alpha_n^* = C \Rightarrow \xi_n = 1 - y_n(w^Tx_n+b) \geq 0$

$\Rightarrow 1 \geq y_n(w^Tx_n+b) = y_n((\sum_{m=1}^{N}\alpha_m y_m x_m)^T x_n + b) = y_n(\sum_{m=1}^{N}\alpha_m y_m K(x_n, x_m) + b)$

・如果 $y_n = 1 \Rightarrow 1 - \sum_{m=1}^{N}\alpha_m y_m K(x_n, x_m) \geq b$

・如果 $y_n = -1 \Rightarrow 1 + \sum_{m=1}^{N}\alpha_m y_m K(x_n, x_m) \geq -b \Rightarrow b \geq -1 - \sum_{m=1}^{N}\alpha_m y_m K(x_n, x_m)$

$y_n = 1$ 時，$1 - \sum_{m=1}^{N}\alpha_m y_m K(x_n, x_m) \geq b$；且 $\forall x_n$，b 都要小於 $1 - \sum_{m=1}^{N}\alpha_m y_m K(x_n, x_m)$

$\Rightarrow largest\ b^* = min_{n:y_n>0}(1 - \sum_{m=1}^{N}\alpha_m y_m K(x_n, x_m))$

13. **(e)**

設 $\phi(x_n) \in \mathbb{R}^k$

・目標：將 P_2 轉為 hard-margin 的形式：

want：$\min\limits_{w,b,\xi}(\frac{1}{2}w^Tw + C\sum_{n=1}^{N}\xi_n^2) = \min\limits_{\hat{w},b}(\frac{1}{2}\hat{w}^T\hat{w})$

$$\Rightarrow w^Tw + 2C\sum_{n=1}^{N}\xi_n^2 = \hat{w}^T\hat{w} \quad \Rightarrow \hat{w} = \begin{bmatrix} w \\ \sqrt{2C}\xi_1 \\ \sqrt{2C}\xi_2 \\ \vdots \\ \sqrt{2C}\xi_N \end{bmatrix}_{(k+N)\times 1}$$

・重新定義 $y_n(w^T\phi(x_n) + b) \geq 1 - \xi_n,\ for\ \ n = 1, 2, \ldots, N$

by hard-margin，希望限制式為 $y_n(\hat{w}^T\phi(\hat{x}_n) + b) \geq 1$的形式

$$y_n(w^T\phi(x_n) + b) \geq 1 - \xi_n \Rightarrow y_n(w^T\phi(x_n) + \frac{\xi_n}{y_n} + b) \geq 1$$

want：$\hat{w}^T\phi(\hat{x}_n) = w^T\phi(x_n) + \frac{\xi_n}{y_n} \Rightarrow \phi(\hat{x}_n) = \begin{bmatrix} \phi(x_n) \\ 0 \\ \vdots \\ \frac{1}{\sqrt{2C}y_n} \\ \vdots \\ 0 \end{bmatrix}_{(k+N)\times 1}$ → 第 $k+n$ 項

$$\Diamond = \phi(\hat{x}_n)^T\phi(\hat{x}_m) = \phi(x_n)^T\phi(x_m) + \frac{[n=m]}{2Cy_n^2} = K(x_n, x_m) + \frac{[n=m]}{2C}$$

14. **(e)**

$$\hat{w} = \sum_{n=1}^{N}\alpha_n^* y_n \phi(\hat{x}_n)$$

$\Rightarrow \sqrt{2C}\xi_n = \alpha_n^* y_n \frac{1}{\sqrt{2C}y_n}$ （承上題，從 $\hat{w}$ 的第 $k+1$ 到第 $k+N$ 項可得此關係式）

$$\Rightarrow \xi_n = \alpha_n^*\frac{1}{2C},\ for\ \ n = 1, 2, \ldots, N$$

15. **(d)** 8.45
16. **(b)** “2” versus “not 2”
17. **(c)** 712
18. **(d)** 10
19. **(b)** 1
20. **(b)** 1

In [134]:
```
import numpy as np
from svmutil import *
```

In [135]:
```
def readfile(filename):
    X = []
    Y = []
    for lines in open(filename).readlines():
        temp = lines.strip().split()
        Y.append(int(temp[0]))
        x = np.zeros(36)
        for i in range(1, len(temp)):
            index_value = temp[i].split(":")
            x[int(index_value[0])-1] = float(index_value[1])
        X.append(x.tolist())
    return X, Y
def label_OneOfTheClass(Y, Class):
    label = np.zeros(len(Y))
    for i in range(len(Y)):
        if Y[i]==Class:
            label[i] = 1
        else:
            label[i] = -1
    return label
TrainFile = "../../satimage.scale"
TestFile = "../../satimage.scale.t"
X, Y = readfile(TrainFile)
Xt, Yt = readfile(TestFile)
```

In [136]:
```
## 15
label = label_OneOfTheClass(Y, 3)
m = svm_train(label, X, '-s 0 -t 0 -c 10')
SV = m.get_SV()
alpha = m.get_sv_coef()
SVnum = len(SV)
w = np.zeros(36)
for i in range(SVnum):
    for j in range(36):
        if SV[i].get(j+1) != None:
            w[j] += alpha[i][0]*SV[i][j+1]
print("15: ", np.sqrt(np.dot(w, w)))
```

```
15:  8.457084298367683
```

In [137]:
```
## 16, 17
max_acc = -1
max_SVnum = 0
for i in range(1, 6):
    Class = i
    label = label_OneOfTheClass(Y, i)
    m = svm_train(label, X, '-s 0 -t 1 -c 10 -d 2 -g 1 -r 1')
    p_label, p_acc, p_val = svm_predict(label, X, m)
    if p_acc[0] > max_acc:
        max_acc = p_acc[0]
        best = i
    SV = m.get_SV()
    SVnum = len(SV)
    if SVnum > max_SVnum:
        max_SVnum = SVnum
print("16: ", best)
print("17: ", max_SVnum)
```

```
Accuracy = 99.9324% (4432/4435) (classification)
Accuracy = 100% (4435/4435) (classification)
Accuracy = 97.7678% (4336/4435) (classification)
Accuracy = 95.9865% (4257/4435) (classification)
Accuracy = 99.3236% (4405/4435) (classification)
16:  2
17:  712
```

In [138]:
```
## 18
C = [0.01, 0.1, 1, 10, 100]
label = label_OneOfTheClass(Y, 6)
label_t = label_OneOfTheClass(Yt, 6)
min_Eout = 100
for i in range(5):
    param = '-s 0 -t 2 -g 10 -c ' + str(C[i])
    m = svm_train(label, X, param)
    p_label, p_acc, p_val = svm_predict(label_t, Xt, m)
    if (100-p_acc[0]) < min_Eout:
        min_Eout = 100-p_acc[0]
        best = C[i]
print("18: ", best)
```

```
Accuracy = 76.5% (1530/2000) (classification)
Accuracy = 83.65% (1673/2000) (classification)
Accuracy = 89.35% (1787/2000) (classification)
Accuracy = 90.3% (1806/2000) (classification)
Accuracy = 90.3% (1806/2000) (classification)
18:  10
```

In [139]:
```
## 19
gamma = [0.1, 1, 10, 100, 1000]
min_Eout = 100
for i in range(5):
    param = '-s 0 -t 2 -c 0.1 -g ' + str(gamma[i])
    m = svm_train(label, X, param)
    p_label, p_acc, p_val = svm_predict(label_t, Xt, m)
    if (100-p_acc[0]) < min_Eout:
        min_Eout = 100-p_acc[0]
        best = gamma[i]
print("19: ", best)
```

```
Accuracy = 90.15% (1803/2000) (classification)
Accuracy = 93% (1860/2000) (classification)
Accuracy = 83.65% (1673/2000) (classification)
Accuracy = 76.5% (1530/2000) (classification)
Accuracy = 76.5% (1530/2000) (classification)
19:  1
```

In [147]:
```
## 20
gamma = [0.1, 1, 10, 100, 1000]
data_num = len(X)
choose_time = np.zeros([5])
for i in range(1000):
    X_val = []
    Y_val = []
    X_train = []
    Y_train = []
    val = np.random.randint(0, data_num, 200)
    is_val = np.zeros(data_num)
    for j in range(200):
        is_val[val[j]] = 1
    for j in range(data_num):
        if is_val[j] == 1:
            X_val.append(X[j])
            Y_val.append(Y[j])
        else:
            X_train.append(X[j])
            Y_train.append(Y[j])
    label_val = label_OneOfTheClass(Y_val, 6)
    label_train = label_OneOfTheClass(Y_train, 6)
    max_acc = 0
    for g in range(5):
        param = '-s 0 -t 2 -c 0.1 -g ' + str(gamma[g])
        m = svm_train(label_train, X_train, param)
        p_label, p_acc, p_val = svm_predict(label_val, X_val, m)
        if p_acc[0] > max_acc:
            max_acc = p_acc[0]
            best = g
    print(i, ":", gamma[best], end = "\r")
    choose_time[best] += 1
print("20: ", choose_time)
```