CSIE 5432/5433 — Machine Learning Foundations/Techniques

Name: 李吉昌 **Homework 4**

Student Number: r08922a27 **Due Date:** December 4 2020, 13:00

Deterministic Noise

1. Answer: [c]

由題意可得 $L_{square} = \frac{1}{2}\int_0^2 (e^x - w\cdot x)^2\,dx = \frac{1}{2}\int_0^2 (e^{2x} - 2w\cdot xe^x + w^2x^2)\,dx = \frac{8}{2}w^2 - (2e^2+2)w + \frac{1}{2}(e^2+1)$, 令 $\frac{dL_{square}}{dw} = 0$, $\frac{8}{3}w - 1 - e^2 = 0$, 得 $w = \frac{3+3e^2}{8}$。

Learning Curve

2. Answer: [b]

令 $h^* = \text{argmin}_{h\in\mathcal{H}} E_{out}(h)$, 因為 $\mathcal{A}(\mathcal{D})$ 為使 E_{in} 最小的 hypothesis, 在給定任意 $\mathcal{D}$ 的條件下, $E_{in}(\mathcal{A}(\mathcal{D})) \leq E_{in}(h^*)$ 恆成立, 則 $\mathbb{E}_{\mathcal{D}}[E_{in}(\mathcal{A}(\mathcal{D}))] \leq \mathbb{E}_{\mathcal{D}}[E_{in}(h^*)]$ 亦恆成立。

令 $\mathcal{D}$ 為由 N 筆 $i.i.d.$ 資料 $\{(\mathbf{x}_i, y_i)\}_{i=1}^N$ 構成的資料集, 則:

$$\begin{aligned}
\mathbb{E}_{\mathcal{D}}[E_{in}(h^*)] &= \int_{\mathcal{D}} P(\mathcal{D})E_{in}(h^*)d\mathcal{D} = \int_{\mathcal{D}} P(\mathcal{D})[\frac{1}{N}\sum_{i=1}^N \text{err}(h^*(\mathbf{x}_i), y_i)]d\mathcal{D}, \quad \because \{(\mathbf{x}_i, y_i)\}_{i=1}^N \text{ 為 } i.i.d. \\
&= \int_{\mathbf{x}_1}\int_{\mathbf{x}_2}\dots\int_{\mathbf{x}_N} P(\mathbf{x}_1)P(\mathbf{x}_2)\dots P(\mathbf{x}_N)[\frac{1}{N}\sum_{i=1}^N \text{err}(h^*(\mathbf{x}_i), y_i)]d\mathbf{x}_1 d\mathbf{x}_2 \dots d\mathbf{x}_N \\
&= \frac{1}{N}\sum_{i=1}^N [(\prod_{\substack{j=1\\ j\neq i}}^N \int_{\mathbf{x}_j} P(\mathbf{x}_j)d\mathbf{x}_j)\int_{\mathbf{x}_i} P(\mathbf{x}_i)\text{err}(h^*(\mathbf{x}_i), y_i)d\mathbf{x}_i], \quad \because \int_{\mathbf{x}_j} P(\mathbf{x}_j)d\mathbf{x}_j = 1, \\
&= \frac{1}{N}\sum_{i=1}^N \int_{\mathbf{x}_i} P(\mathbf{x}_i)\text{err}(h^*(\mathbf{x}_i), y_i)d\mathbf{x}_i \\
&= \frac{1}{N}\sum_{i=1}^N \underset{\mathbf{x}_i\sim\mathcal{P}}{\mathcal{E}}\text{err}(h^*(\mathbf{x}_i), y_i), \quad \because \underset{\mathbf{x}_i\sim\mathcal{P}}{\mathcal{E}}\text{err}(h^*(\mathbf{x}_i), y_i) = E_{out}(h^*), \\
&= \frac{1}{N}\sum_{i=1}^N E_{out}(h^*) = \frac{1}{N}\cdot N \cdot E_{out}(h^*) = E_{out}(h^*)
\end{aligned}$$

$E_{out}(h^*)$ 與 $\mathcal{D}$ 無關, 對隨機變數 $\mathcal{D}$ 可視為一常數, 因此 $E_{out}(h^*) = \mathbb{E}_{\mathcal{D}}[E_{out}(h^*)]$。此外, 給定任意 $\mathcal{D}$ 的條件下, $E_{out}(h^*) \leq E_{out}(\mathcal{A}(\mathcal{D}))$ 恆成立, 則 $\mathbb{E}_{\mathcal{D}}[E_{out}(h^*)] \leq \mathbb{E}_{\mathcal{D}}[E_{out}(\mathcal{A}(\mathcal{D}))]$ 恆成立。

綜上所述, $\mathbb{E}_{\mathcal{D}}[E_{in}(\mathcal{A}(\mathcal{D}))] \leq \mathbb{E}_{\mathcal{D}}[E_{in}(h^*)] = E_{out}(h^*) = \mathbb{E}_{\mathcal{D}}[E_{out}(h^*)] \leq \mathbb{E}_{\mathcal{D}}[E_{out}(\mathcal{A}(\mathcal{D}))]$, 可推得 $\mathbb{E}_{\mathcal{D}}[E_{in}(\mathcal{A}(\mathcal{D}))] \leq \mathbb{E}_{\mathcal{D}}[E_{out}(\mathcal{A}(\mathcal{D}))]$ 恆成立, $\mathbb{E}_{\mathcal{D}}[E_{in}(\mathcal{A}(\mathcal{D}))] > \mathbb{E}_{\mathcal{D}}[E_{out}(\mathcal{A}(\mathcal{D}))]$ 的情況與結果矛盾, 必 always false。

Noisy Virtual Examples

3. Answer: [d]

令 $\mathbf{X} = \begin{pmatrix} -\mathbf{x}_1^T- \\ -\mathbf{x}_2^T- \\ \vdots \\ -\mathbf{x}_N^T- \end{pmatrix}$, $\xi = \begin{pmatrix} -\epsilon^T- \\ -\epsilon^T- \\ \vdots \\ -\epsilon^T- \end{pmatrix}$, $\tilde{\mathbf{X}} = \begin{pmatrix} -\mathbf{x}_1^T+\epsilon^T- \\ -\mathbf{x}_2^T+\epsilon^T- \\ \vdots \\ -\mathbf{x}_N^T+\epsilon^T- \end{pmatrix} = \mathbf{X}+\xi$

$\mathbf{X}_h = \begin{pmatrix} \mathbf{X} \\ \tilde{\mathbf{X}} \end{pmatrix}$, $\mathbf{X}_h^T = \left(\mathbf{X}^T, \tilde{\mathbf{X}}^T\right)$, 其中 $\mathbf{X}^T\mathbf{X}$ 為常數, $\mathbb{E}(\mathbf{X}^T\mathbf{X}) = \mathbf{X}^T\mathbf{X}$

$$
\begin{aligned}
\mathbb{E}(\mathbf{X}_h^T\mathbf{X}_h) &= \mathbb{E}(\mathbf{X}^T\mathbf{X}+\tilde{\mathbf{X}}^T\tilde{\mathbf{X}}) = \mathbb{E}(\mathbf{X}^T\mathbf{X}+(\mathbf{X}+\xi)^T(\mathbf{X}+\xi)) = \mathbb{E}(2\mathbf{X}^T\mathbf{X}+\xi^T\mathbf{X}+\mathbf{X}^T\xi+\xi^T\xi) \\
&= 2\mathbf{X}^T\mathbf{X}+\mathbb{E}(\xi)^T\mathbf{X}+\mathbf{X}^T\mathbb{E}(\xi)+\mathbb{E}(\xi^T\xi) \\
&= 2\mathbf{X}^T\mathbf{X}+\begin{pmatrix} \mathbb{E}(-\epsilon^T-) \\ \mathbb{E}(-\epsilon^T-) \\ \vdots \\ \mathbb{E}(-\epsilon^T-) \end{pmatrix}^T \mathbf{X}+\mathbf{X}^T\begin{pmatrix} \mathbb{E}(-\epsilon^T-) \\ \mathbb{E}(-\epsilon^T-) \\ \vdots \\ \mathbb{E}(-\epsilon^T-) \end{pmatrix}+\mathbb{E}(\sum_{n=1}^{N}\epsilon\epsilon^T), \quad \because \mathbb{E}(\epsilon)=0, \\
&= 2\mathbf{X}^T\mathbf{X}+\begin{pmatrix} -\mathbf{0}^T- \\ -\mathbf{0}^T- \\ \vdots \\ -\mathbf{0}^T- \end{pmatrix}^T \mathbf{X}+\mathbf{X}^T\begin{pmatrix} -\mathbf{0}^T- \\ -\mathbf{0}^T- \\ \vdots \\ -\mathbf{0}^T- \end{pmatrix}+\mathbb{E}(\sum_{n=1}^{N}\epsilon\epsilon^T) \\
&= 2\mathbf{X}^T\mathbf{X}+\mathbb{E}(\sum_{n=1}^{N}\epsilon\epsilon^T) = 2\mathbf{X}^T\mathbf{X}+\sum_{n=1}^{N}\mathbb{E}(\epsilon\epsilon^T) = 2\mathbf{X}^T\mathbf{X}+\sum_{n=1}^{N}\sigma^2\mathbf{I}_{d+1} = 2\mathbf{X}^T\mathbf{X}+N\sigma^2\mathbf{I}_{d+1}
\end{aligned}
$$

4. Answer: [e]

$\mathbb{E}(\mathbf{X}_h^T\mathbf{y}_h) = \mathbb{E}(\mathbf{X}^T\mathbf{y}+\tilde{\mathbf{X}}^T\mathbf{y}) = \mathbb{E}(\mathbf{X}^T\mathbf{y}+(\mathbf{X}^T+\xi^T)\mathbf{y}), \quad \because \mathbf{X}, \mathbf{y}$ 為常數, 可得下面結果,

$$
= 2\mathbf{X}^T\mathbf{y}+\mathbb{E}(\xi)^T\mathbf{y} = 2\mathbf{X}^T\mathbf{y}+\begin{pmatrix} \mathbb{E}(-\epsilon^T-) \\ \mathbb{E}(-\epsilon^T-) \\ \vdots \\ \mathbb{E}(-\epsilon^T-) \end{pmatrix}^T \mathbf{y} = 2\mathbf{X}^T\mathbf{y}+\begin{pmatrix} -\mathbf{0}^T- \\ -\mathbf{0}^T- \\ \vdots \\ -\mathbf{0}^T- \end{pmatrix}^T \mathbf{y} = 2\mathbf{X}^T\mathbf{y}
$$

Regularization

5. Answer: [d]

$\mathbf{Z} = \mathbf{X}\mathbf{Q}$, 則 $\mathbf{Z}^T\mathbf{Z} = \mathbf{Q}^T\mathbf{X}^T\mathbf{X}\mathbf{Q} = \mathbf{Q}^T(\mathbf{Q}\Gamma\mathbf{Q}^T)\mathbf{Q} = \mathbf{I}_{d+1}\Gamma\mathbf{I}_{d+1} = \Gamma$

根據 Lecture 14 slides 第 10 頁得最佳解公式並由題意代入 $\mathbf{Z}^T\mathbf{Z} = \Gamma$:

$\mathbf{v} = (\mathbf{Z}^T\mathbf{Z})^{-1}\mathbf{Z}^T\mathbf{y} = \Gamma^{-1}\mathbf{Z}^T\mathbf{y}$, $\because \Gamma$ 為對角矩陣 , 則 Γ^{-1} 亦為對角矩陣, 得 $v_i = \dfrac{1}{\gamma_i}(\mathbf{Z}^T\mathbf{y})_i$ 。

$\mathbf{u} = (\mathbf{Z}^T\mathbf{Z}+\lambda\mathbf{I}_{d+1})^{-1}\mathbf{Z}^T\mathbf{y} = (\Gamma+\lambda\mathbf{I}_{d+1})^{-1}\mathbf{Z}^T\mathbf{y}$, $\because (\Gamma , \lambda\mathbf{I}_{d+1})$ 為對角矩陣 , 則 $(\Gamma+\lambda\mathbf{I}_{d+1})^{-1}$ 亦為對角矩陣, 得 $u_i = \frac{1}{\gamma_i+\lambda}(\mathbf{Z}^T\mathbf{y})_i$ 。

綜上所述, 得 $\frac{u_i}{v_i} = \frac{\gamma_i}{\gamma_i+\lambda}\cdot\frac{(\mathbf{Z}^T\mathbf{y})_i}{(\mathbf{Z}^T\mathbf{y})_i} = \frac{\gamma_i}{\gamma_i+\lambda}$ 。

6. Answer: [a]

令 $\mathbf{X} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{pmatrix}_{N\times 1}$, $\mathbf{y} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{pmatrix}_{N\times 1}$

根據 Lecture 14 slides 第 10 頁得最佳解公式 $w^* = (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I}_{1\times 1})^{-1}\mathbf{X}^T\mathbf{y} = \frac{\sum_{n=1}^N x_n y_n}{\sum_{n=1}^N x_n x_n + \lambda}$

由題意可知最佳解在 constraint 邊界, 得 $C = (w^*)^2 = (\frac{\sum_{n=1}^N x_n y_n}{\sum_{n=1}^N x_n x_n + \lambda})^2$。

7. Answer: [d]

令 $\Omega(y) = (y+c)^2$, 當 $\frac{d}{dy}\{\frac{1}{N}(\sum_{n=1}^N (y-y_n)^2 + 2K(y+c)^2)\} = 0$ 時, y 為最佳解,

$\frac{d}{dy}\{\frac{1}{N}(\sum_{n=1}^N (y-y_n)^2 + 2K(y+c)^2)\} = 0$, 則 $\sum_{n=1}^N (y-y_n) + 2K(y+c) = 0$

移項整理得 $y = \frac{\sum_{n=1}^N y_n - 2K\cdot c}{N+2K}$, 又由題意得 $\frac{\sum_{n=1}^N y_n - 2K\cdot c}{N+2K} = \frac{\sum_{n=1}^N y_n + K}{N+2K}$,

綜上所述, 因 $-2K\cdot c = K$, 可得 $c = -0.5$, 因此 $\Omega(y) = (y-0.5)^2$。

8. Answer: [b]

令 $\begin{cases} \tilde{L}(\tilde{\mathbf{w}}) = \frac{1}{N}\sum_{n=1}^N (\tilde{\mathbf{w}}^T\Phi(\mathbf{x}_n) - y_n)^2 + \frac{\lambda}{N}(\tilde{\mathbf{w}}^T\tilde{\mathbf{w}}) \\ L(\mathbf{w}) = \frac{1}{N}\sum_{n=1}^N (\mathbf{w}^T\mathbf{x}_n - y_n)^2 + \frac{\lambda}{N}\Omega(\mathbf{w}) \end{cases}$

$\tilde{\mathbf{w}}^T\Phi(\mathbf{x}_n) = \tilde{\mathbf{w}}^T\Gamma^{-1}\mathbf{x}_n = \tilde{\mathbf{w}}^T((\Gamma^{-1})^T)^T\mathbf{x}_n = ((\Gamma^{-1})^T\tilde{\mathbf{w}})^T\mathbf{x}_n = ((\Gamma^T)^{-1}\tilde{\mathbf{w}})^T\mathbf{x}_n$

因為對角矩陣 Γ 為對稱, $\Gamma = \Gamma^T$, 因此 $\tilde{\mathbf{w}}^T\Phi(\mathbf{x}_n) = (\Gamma^{-1}\tilde{\mathbf{w}})^T\mathbf{x}_n$。

令 $\mathbf{w} = \Gamma^{-1}\tilde{\mathbf{w}}$, 可得 $L = \frac{1}{N}\sum_{n=1}^N (\tilde{\mathbf{w}}^T\Phi(\mathbf{x}_n) - y_n)^2 + \frac{\lambda}{N}\Omega(\mathbf{w})$, 當 $L = \tilde{L}$, 由封閉性得 $\Omega(\mathbf{w}) = \tilde{\mathbf{w}}^T\tilde{\mathbf{w}}$,

轉換之對角矩陣 Γ^{-1} 可逆, 可代入 $\tilde{\mathbf{w}} = \Gamma\mathbf{w}$, $\Omega(\mathbf{w}) = (\Gamma\mathbf{w})^T(\Gamma\mathbf{w}) = \mathbf{w}^T\Gamma^T\Gamma\mathbf{w} = \mathbf{w}^T\Gamma\Gamma\mathbf{w} = \mathbf{w}^T\Gamma^2\mathbf{w}$。

因為 $\mathbf{w}$ 和 $\tilde{\mathbf{w}}$ 轉換關係為 bijective, 當 $\mathbf{w} = \Gamma^{-1}\tilde{\mathbf{w}}$, $\Omega(\mathbf{w}) = \mathbf{w}^T\Gamma^2\mathbf{w}$ 時,

$\frac{1}{N}\sum_{n=1}^N (\tilde{\mathbf{w}}^T\Phi(\mathbf{x}_n) - y_n)^2 + \frac{\lambda}{N}\tilde{\mathbf{w}}^T\tilde{\mathbf{w}} = \frac{1}{N}\sum_{n=1}^N (\mathbf{w}^T\mathbf{x}_n - y_n)^2 + \frac{\lambda}{N}\Omega(\mathbf{w})$ 恆成立,

則 $\min_{\mathbf{w}}\{\frac{1}{N}\sum_{n=1}^N (\tilde{\mathbf{w}}^T\Phi(\mathbf{x}_n) - y_n)^2 + \frac{\lambda}{N}\tilde{\mathbf{w}}^T\tilde{\mathbf{w}}\} = \min_{\tilde{\mathbf{w}}}\{\frac{1}{N}\sum_{n=1}^N (\mathbf{w}^T\mathbf{x}_n - y_n)^2 + \frac{\lambda}{N}\Omega(\mathbf{w})\}$ 必成立,

其中, 最佳解關係亦為 $\mathbf{w}^* = \Gamma^{-1}\tilde{\mathbf{w}}^*$。

9. Answer: [b]

令 $L(\mathbf{w}) = \sum_{n=1}^N (\mathbf{w}^T\mathbf{x}_n - y_n)^2$

Stage 1:

因為 $\mathbf{B}$ 為對角矩陣, $\sum_{i=0}^d \beta_i w_i^2 = \mathbf{w}^T\mathbf{B}\mathbf{w}$, 其中 $\mathbf{B}$ 具對稱性, 故 $\mathbf{B} = \mathbf{B}^T$,

$$\begin{aligned} \frac{d}{d\mathbf{w}}\{\frac{1}{N}(L(\mathbf{w}) + \lambda\mathbf{w}^T\mathbf{B}\mathbf{w})\} &= \frac{1}{N}(L'(\mathbf{w}) + \lambda\frac{d}{d\mathbf{w}}\{\mathbf{w}\}\mathbf{B}\mathbf{w} + \lambda\frac{d}{d\mathbf{w}}\{\mathbf{B}\mathbf{w}\}\mathbf{w}) \\ &= \frac{1}{N}(L'(\mathbf{w}) + \lambda(\mathbf{B} + \mathbf{B}^T)\mathbf{w}) = \frac{1}{N}(L'(\mathbf{w}) + 2\lambda\mathbf{B}\mathbf{w}) \end{aligned}$$

當 $\mathbf{w}$ 為最佳解時, $L'(\mathbf{w}) = -2\lambda\mathbf{B}\mathbf{w}$

Stage 2:

$$
\begin{aligned}
\frac{d}{d\mathbf{w}}\{\frac{1}{N+K}(L(\mathbf{w})+\sum_{k=1}^{K}(\mathbf{w}^T\tilde{\mathbf{x}}_k-\tilde{y}_k)^2)\} &= \frac{1}{N+K}(L'(\mathbf{w})+\frac{d}{d\mathbf{w}}\{\|\tilde{\mathbf{X}}\mathbf{w}-\tilde{\mathbf{y}}\|^2\}) \\
&= \frac{1}{N+K}(L'(\mathbf{w})+2(\frac{d}{d\mathbf{w}}\{\tilde{\mathbf{X}}\mathbf{w}\}(\tilde{\mathbf{X}}\mathbf{w}-\tilde{\mathbf{y}}))) \\
&= \frac{1}{N+K}(L'(\mathbf{w})+2(\tilde{\mathbf{X}}^T\tilde{\mathbf{X}}\mathbf{w}-\tilde{\mathbf{X}}^T\tilde{\mathbf{y}}))
\end{aligned}
$$

當 $\mathbf{w}$ 為最佳解時, $L'(\mathbf{w})=-2(\tilde{\mathbf{X}}^T\tilde{\mathbf{X}}\mathbf{w}-\tilde{\mathbf{X}}^T\tilde{\mathbf{y}})$

由 Stage 1 和 Stage 2 結果得:

$\lambda\mathbf{B}\mathbf{w}-\mathbf{0}=\tilde{\mathbf{X}}^T\tilde{\mathbf{X}}\mathbf{w}-\tilde{\mathbf{X}}^T\tilde{\mathbf{y}}$, 由封閉性得 $\begin{cases}\tilde{\mathbf{X}}^T\tilde{\mathbf{X}}=\lambda\mathbf{B}\\ \tilde{\mathbf{X}}^T\tilde{\mathbf{y}}=\mathbf{0}\end{cases}$, 可推至 $\begin{cases}\tilde{\mathbf{X}}=\sqrt{\lambda}\cdot\sqrt{\mathbf{B}}\\ \tilde{\mathbf{y}}=\mathbf{0}\end{cases}$ 。

Leave-one-out

10. Answer: [e]

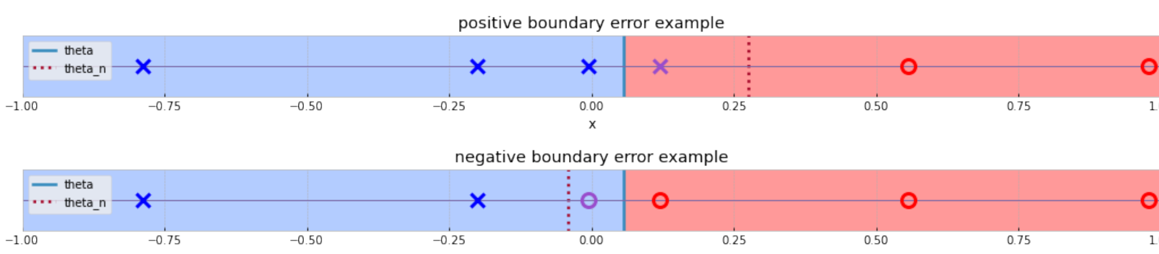
當 $\mathbf{x}_n$ 為 positive, 則 negative examples 的數量會大於 positive examples, 預測結果必為 negative, $\mathbf{e}_n=1$; 當 $\mathbf{x}_n$ 為 negative, 則 negative examples 的數量會小於 positive examples, 預測結果必為 positive, $\mathbf{e}_n=1$, 綜上所述, $\mathcal{A}_{majority}$ 在任何 leave-one-out 的情況皆必答錯, $\mathbf{e}_n=1$ 恆成立, 得 $E_{loocv}(\mathcal{A}_{majority})=\frac{1}{2N}\sum_{n=1}^{2N}1=1$。

11. Answer: [c]

令所有 N 筆訓練資料得到的 hypothesis 為 θ; 使用 $\mathbf{x}_n$ 作為 valid sample, 剩下 $N-1$ 筆訓練資料求得的 hypothesis 為 θ_n , 將所有 $\mathbf{e}_n$ 的可能分成兩種情況討論:

(1) 若 valid sample 為在 θ 的左/右的邊界 sample:

以下圖為例, 當 valid sample 在 θ 左或右邊的情況時, 有可能發生兩種 valid sample 同時都預測錯誤的狀況。



(2) 若 valid sample 不是 θ 的邊界 sample:

假使存在一非邊界 sample $\mathbf{x}_n$ 在成為 valid sample 後會使 θ_n 將其誤判, 若 $\mathbf{x}_n$ 為 positive , 代表 $\mathbf{x}_n$ 在 θ_n 左邊, 但如果 $\mathbf{x}_n$ 不是 θ 的邊界 sample, 表示必存在一 $\mathbf{x}'_n$ 比 $\mathbf{x}_n$ 更小, 但仍為 positive, 代表 θ_n 最大只能位在大於 $\mathbf{x}'_n$ 且小於 $\mathbf{x}_n$ 的區間內, 與假設矛盾; 若 $\mathbf{x}_n$ 為 negative , 代表 $\mathbf{x}_n$ 在 θ_n 右邊, 但如果 $\mathbf{x}_n$ 不是 θ 的邊界 sample, 表示必存在一 $\mathbf{x}'_n$ 比 $\mathbf{x}_n$ 更大, 但仍為 negative, 代表 θ_n 最大只能位在小於 $\mathbf{x}'_n$ 且大於 $\mathbf{x}_n$ 的區間內, 與假設矛盾, 綜上所述, valid samples 不是在 θ 的時候, $\mathbf{e}_n=0$ 恆成立。

由 (1)(2) 得, 有可能發生 $\mathbf{e}_n = 1$ 的情況只有在 $\mathbf{x}_n$ 為決定 θ 的左右兩個點的時候, 因此 $\sum_{n=1}^{N} \mathbf{e}_n \leq 2$, 得 $E_{loocv} = \frac{1}{N}\sum_{n=1}^{N} \mathbf{e}_n \leq \frac{2}{N}$, 且根據題意 $N \geq 4$, 則 $\frac{2}{N} \leq \frac{1}{2}$, 得 $\frac{2}{N}$ 為 tightest upper bound。

12. Answer: [e]

(1) constant model:

令 i, j 為訓練參數資料的 index, $L_{constant}(w_0) = (w_0 - y_i)^2 + (w_0 - y_j)^2$, $L'_{constant}(w_0) = 2(2w_0 - y_i - y_j)$, 當 $w_0 = \frac{y_i + y_j}{2}$ 時為最佳解:

$$w_0|_{\mathbf{e}_1} = \frac{2+0}{2} = 1,\ \mathbf{e}_1 = (1-0)^2 = 1$$

$$w_0|_{\mathbf{e}_2} = \frac{0+0}{2} = 0,\ \mathbf{e}_2 = (0-2)^2 = 4$$

$$w_0|_{\mathbf{e}_3} = \frac{0+2}{2} = 1,\ \mathbf{e}_3 = (1-0)^2 = 1$$

$$E_{loocv-constant} = \frac{1}{3}(1+4+1) = 2$$

(2) linear model:

令 i, j 為訓練參數資料的 index, $L_{linear}(w_0, w_1) = (w_0 + w_1 x_i - y_i)^2 + (w_0 + w_1 x_j - y_j)^2$, 訓練資料兩點構成一線, 則 $w_0 = \frac{x_i y_j - x_j y_i}{x_i - x_j}$, $w_1 = \frac{y_i - y_j}{x_i - x_j}$:

$$w_0|_{\mathbf{e}_1} = \frac{\rho \cdot 0 - (-3) \cdot 2}{\rho - (-3)} = \frac{6}{\rho + 3},\ w_1|_{\mathbf{e}_1} = \frac{2-0}{\rho - (-3)} = \frac{2}{\rho + 3},\ \mathbf{e}_1 = (\frac{6}{\rho+3} + \frac{2}{\rho+3} \cdot 3 - 0)^2 = (\frac{12}{\rho+3})^2$$

$$w_0|_{\mathbf{e}_2} = \frac{3 \cdot 0 - (-3) \cdot 0}{3 - (-3)} = 0,\ w_1|_{\mathbf{e}_2} = \frac{0-0}{3-(-3)} = 0,\ \mathbf{e}_2 = (0 + 0 \cdot \rho - 2)^2 = 4$$

$$w_0|_{\mathbf{e}_3} = \frac{3 \cdot 2 - \rho \cdot 0}{3 - \rho} = \frac{6}{3-\rho},\ w_1|_{\mathbf{e}_3} = \frac{2-0}{\rho-3} = \frac{2}{\rho-3},\ \mathbf{e}_3 = (\frac{6}{3-\rho} + \frac{2}{\rho-3} \cdot (-3) - 0)^2 = (\frac{12}{\rho-3})^2$$

$$E_{loocv-linear} = \frac{1}{3}((\frac{12}{\rho+3})^2 + 4 + (\frac{12}{\rho-3})^2)$$

根據題意, $E_{loocv-constant} = E_{loocv-linear}$, 由 (1)(2) 得 $2 = \frac{1}{3}((\frac{12}{\rho+3})^2 + 4 + (\frac{12}{\rho-3})^2)$,
整理移項得多項式 $\rho^4 - 162\rho^2 - 1215 = 0$, 因式分解為 $(\rho^2 - (81 + 36\sqrt{6}))(\rho^2 - (81 - 36\sqrt{6})) = 0$,
其中, [e] 選項滿足其中一解 $\rho = \sqrt{81 + 36\sqrt{6}}$。

13. Answer: [d]

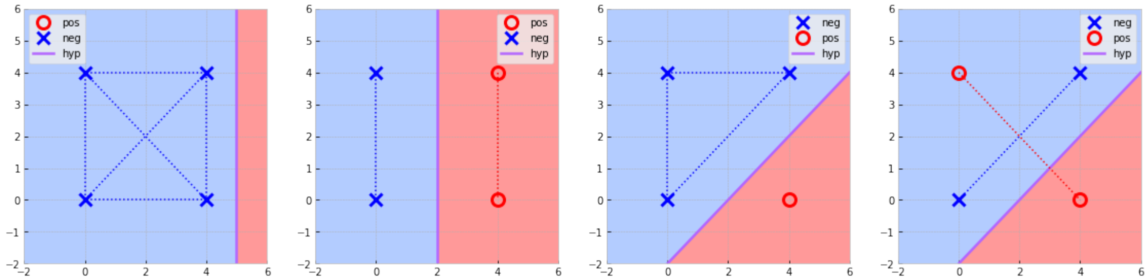
令 $(\mathbf{x}_i, y_i)$ 為第 i 個從分佈 $\mathcal{P}$ 抽到的 sample, 證明如下:

$$\begin{aligned}
\underset{\mathcal{D}_{\text{val}} \sim \mathcal{P}^k}{\text{Var}}[E_{val}(h)] &= \underset{\mathcal{D}_{\text{val}} \sim \mathcal{P}^k}{\text{Var}}[\frac{1}{K}\sum_{i=1}^{K} \text{err}(h(\mathbf{x}_i), y_i)],\ \mathbf{x}_i \text{ 為 mutually independent, Var 可拆成線性疊加,} \\
&= \sum_{i=1}^{K} \underset{(\mathbf{x}_i, y_i) \sim \mathcal{P}}{\text{Var}}[\frac{1}{K}\text{err}(h(\mathbf{x}_i), y_i)],\ (\mathbf{x}_i, y_i) \text{ 來自同一分佈 } \mathcal{P} \text{ 可全部記作 } (\mathbf{x}, y), \\
&= K \cdot \underset{(\mathbf{x}, y) \sim \mathcal{P}}{\text{Var}}[\frac{1}{K}\text{err}(h(\mathbf{x}), y)] = K \cdot \frac{1}{K^2} \cdot \underset{(\mathbf{x}, y) \sim \mathcal{P}}{\text{Var}}[\text{err}(h(\mathbf{x}), y)] \\
&= \frac{1}{K} \cdot \underset{(\mathbf{x}, y) \sim \mathcal{P}}{\text{Var}}[\text{err}(h(\mathbf{x}), y)]
\end{aligned}$$

14. Answer: [c]

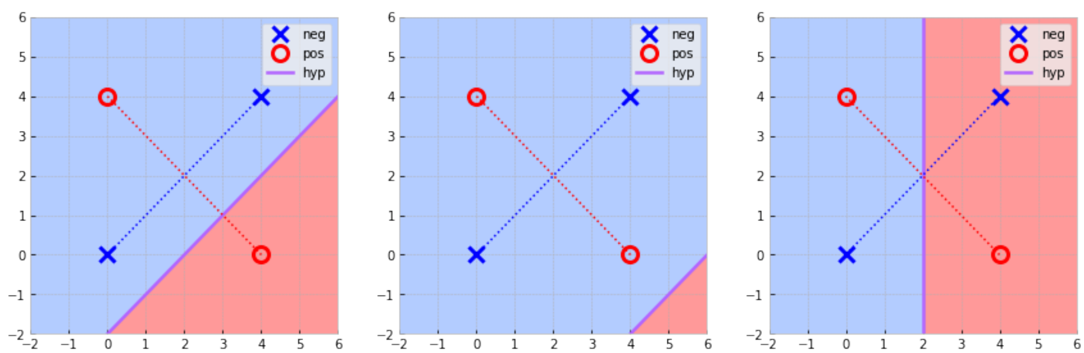
$\mathbb{E}_{y1,y2,y3,y4}(\min_{\mathbf{w}\in\mathbb{R}^{2+1}} E_{in}(\mathbf{w})) = \sum_{i=1}^{16} P(y1,y2,y3,y4) \min_{\mathbf{w}\in\mathbb{R}^{2+1}} E_{in}(\mathbf{w})|_{y1,y2,y3,y4}$

每個同 label 的頂點和其他同 label 的頂點連線形成 connected component, 兩個 label 對應到兩個 connected component, 四個頂點所有 label 可能的組合可以簡化成四種 connected component 的情況討論:



若 hyper-line 與任一 connected component 的 edge 發生交點, 代表 edge 對應到的兩頂點介於 hyper-line 兩邊, 兩頂點必為異號, 表示本該同號的兩頂點被 hyper-line 分錯, 代表該 hyper-line 無法使 $E_{in}(\mathbf{w})|_{y1,y2,y3,y4} = 0$ 成立, 因此當 $E_{in}(\mathbf{w})|_{y1,y2,y3,y4} = 0$ 時, hyper-line 與任一 connected component 的 edge 必定不會發生交點。

第一種情況只有一個 connected component, 上述交點條件不發生一定能使 $E_{in} = 0$; 第二種情況兩 connected component 分別為兩線段, 只要兩線段不產生交點, 必能找到一條 hyper-line 劃分兩線段, 使 $E_{in} = 0$; 第三種情況為一三角形與一頂點, 只要一頂點不位在三角形內, 必能找到一條 hyper-line 劃分兩條線, 使 $E_{in} = 0$; 第四種為矩形同邊異號對角同號的狀況, $E_{in} \neq 0$ 恆成立, 證明如下:



如上圖所示, 若 hyper-line 沒有介於兩 connected component 之間, 則無法劃分不同的 connected component, 所有頂點將被歸類為同號, 無法將四點正確分類; 若 hyper-line 介於兩 connected component 之間, 會與至少一對角線產生交點, 而產生交點表示對角線兩端位在 hyper-line 的兩邊, hyper-line 兩邊的頂點必為異號, 在所有該情況的可能下, 不存在一 hypothesis 能夠使 $E_{in} = 0$, 但如第一張圖所示, 可以利用一三角形與一頂點對應的 hypothesis 分對其中三點, 使得 $E_{in} = \frac{1}{4}$。

綜上所述, 第四種 connected component 同邊異號對角同號的情況共有 2 個, 其他前三種 connected component 共有 14 個, 所有頂點構成的 label 任一組合發生的機率為 $\frac{1}{16}$, 可得:

$$\begin{aligned}
\mathbb{E}_{y1,y2,y3,y4}(\min_{\mathbf{w}\in\mathbb{R}^{2+1}} E_{in}(\mathbf{w})) &= 2 \cdot P[\text{同邊異號對角同號}] \cdot \min_{\mathbf{w}\in\mathbb{R}^{2+1}} E_{in}(\mathbf{w})|_{\text{同邊異號對角同號}} \\
&\quad + 14 \cdot P[\text{其他}] \cdot \min_{\mathbf{w}\in\mathbb{R}^{2+1}} E_{in}(\mathbf{w})|_{\text{其他}} \\
&= 2 \cdot \frac{1}{16} \cdot \frac{1}{4} + 14 \cdot \frac{1}{16} \cdot 0 = \frac{2}{64}
\end{aligned}$$

15. Answer: [a]

$E_{out}(g) = p\epsilon_+ + (1-p)\epsilon_-$, 根據題意得 $E_{out}(g) = E_{out}(g_c) = 1-p$, 則 $p\epsilon_+ + (1-p)\epsilon_- = 1-p$, $p(\epsilon_+ - \epsilon_- + 1) = 1 - \epsilon_-$, 得 $p = \frac{1-\epsilon_-}{\epsilon_+ - \epsilon_- + 1}$。

Experiment

程式碼實作細節如下, 可以透過 parser 的 −−tra_path/−−tst_path 設置訓練資料和測試資料的路徑

```
python code.py --tra_path hw4_train.dat --tst_path hw4_test.dat
```

```python
from liblinearutil import *
import numpy as np
import argparse

'''Define Function'''


def Q_transform(X):
    X_scnd = []
    for i in range(1, X.shape[1]):
        for j in range(i, X.shape[1]):
            X_scnd.append(X[:, i] * X[:, j])
    X_scnd = np.array(X_scnd).T
    return np.hstack((X, X_scnd))


def get_data(path, bias=1.0, transform=None):
    X = []
    for x in open(path, 'r'):
        x = x.strip().split(' ')
        x = [float(v) for v in x]
        X.append([bias] + x)

    X = np.array(X)
    X, Y = np.array(X[:, :-1]), np.array(X[:, -1])

    if transform is not None:
        X = transform(X)

    return X, Y


def main():
    '''Parsing'''
    parser = argparse.ArgumentParser(
        description='Argument Parser for MLF HW4.')

    parser.add_argument('--tra_path', default='hw4_train.dat')
```

```python
parser.add_argument('--tst_path', default='hw4_test.dat')
args = parser.parse_args()

# load data
X_tra, Y_tra = get_data(path=args.tra_path, transform=Q_transform)
X_tst, Y_tst = get_data(path=args.tst_path, transform=Q_transform)

log10_lambda_choices = [-4, -2, 0, 2, 4]
lambda_choices = [10**lmd for lmd in log10_lambda_choices]

'''Answer questions'''
print('RUNNING Q16...')
best_log_lmd = 0
max_acc = 0
for i in range(len(lambda_choices)):
    lmd = lambda_choices[i]
    model = train(
        Y_tra, X_tra, '-s 0 -c {:f} -e 0.000001 -q'.format(1 / (2*lmd)))
    _, pre_acc, _ = predict(Y_tst, X_tst, model, '-q')
    if pre_acc[0] >= max_acc:
        best_log_lmd = log10_lambda_choices[i]
        max_acc = pre_acc[0]
print('Answer of Q16 : {:2d}\n'.format(best_log_lmd))

print('RUNNING Q17...')
best_log_lmd = 0
max_acc = 0
for i in range(len(lambda_choices)):
    lmd = lambda_choices[i]
    model = train(
        Y_tra, X_tra, '-s 0 -c {:f} -e 0.000001 -q'.format(1 / (2*lmd)))
    _, pre_acc, _ = predict(Y_tra, X_tra, model, '-q')
    if pre_acc[0] >= max_acc:
        best_log_lmd = log10_lambda_choices[i]
        max_acc = pre_acc[0]
print('Answer of Q17 : {:2d}\n'.format(best_log_lmd))

print('RUNNING Q18...')
best_lmd_idx = 0
max_acc = 0
for i in range(len(lambda_choices)):
    lmd = lambda_choices[i]
    model = train(Y_tra[:120], X_tra[:120],
                  '-s 0 -c {:f} -e 0.000001 -q'.format(1 / (2*lmd)))
    _, pre_acc, _ = predict(
        Y_tra[120:], X_tra[120:], model, '-q')
    if pre_acc[0] >= max_acc:
        best_lmd_idx = i
        max_acc = pre_acc[0]
model = train(Y_tra[:120], X_tra[:120],
              '-s 0 -c {:f} -e 0.000001 -q'.format(1 / (2*lambda_choices[best_lmd_idx])))
```

```python
    _, pre_acc, _ = predict(Y_tst, X_tst, model, '-q')
    print('Answer of Q18 : {:.4f}\n'.format((100 - pre_acc[0]) * 0.01))

    print('RUNNING Q19...')
    model = train(Y_tra, X_tra,
                  '-s 0 -c {:f} -e 0.000001 -q'.format(1 / (2*lambda_choices[best_lmd_idx])))
    _, pre_acc, _ = predict(Y_tst, X_tst, model, '-q')
    print('Answer of Q19 : {:.4f}\n'.format((100 - pre_acc[0]) * 0.01))

    print('RUNNING Q20...')
    folds_num = 5
    X_folds = np.vsplit(X_tra, folds_num)
    Y_folds = np.hsplit(Y_tra, folds_num)
    best_lmd_idx = 0
    max_acc = 0
    for i in range(len(lambda_choices)):
        lmd = lambda_choices[i]
        acc_list = []
        for fold_idx in range(folds_num):
            X_tra_cv = np.vstack([X_folds[f_idx]
                                  for f_idx in range(folds_num) if fold_idx != f_idx])
            Y_tra_cv = np.hstack([Y_folds[f_idx]
                                  for f_idx in range(folds_num) if fold_idx != f_idx])
            X_val_cv = X_folds[fold_idx]
            Y_val_cv = Y_folds[fold_idx]
            model = train(Y_tra_cv, X_tra_cv,
                          '-s 0 -c {:f} -e 0.000001 -q'.format(1 / (2*lmd)))
            _, pre_acc, _ = predict(Y_val_cv, X_val_cv, model, '-q')
            acc_list.append(pre_acc[0])
        acc = np.mean(acc_list)
        if acc >= max_acc:
            best_lmd_idx = i
            max_acc = acc
    print('Answer of Q20 : {:.4f}\n'.format((100 - max_acc) * 0.01))


if __name__ == "__main__":
    main()
```

16. Answer: [b]	17. Answer: [a]	18. Answer: [e]	19. Answer: [d]	20. Answer: [c]
-2	-4	0.1433	0.13	0.12