

(1.) (C)

$$\bar{E} = \int_0^2 (e^x - \omega x)^2 dx = \int_0^2 (e^{2x} - 2\omega x e^x + \omega^2 x^2) dx$$

$$= \left. \frac{1}{2} e^{2x} - 2\omega (x e^x - e^x) + \frac{1}{3} \omega^2 x^3 \right|_0^2$$

$$= \frac{1}{2} (e^4 - 1) - 2\omega (2e^2 - e^2 + 1) + \frac{1}{3} \omega^2 8$$

$$\frac{d\bar{E}}{d\omega} = -2(e^2 + 1) + \frac{16}{3} \omega = 0$$

$$\Rightarrow \omega = \frac{3}{8} (e^2 + 1)$$

magnitude of deterministic noise

$$= |e^x - \omega x|$$

$$= \left| e^x - \frac{3}{8} (e^2 + 1) x \right|$$

2. (b)

$$h^* = \operatorname{argmin}_{h \in H} E_{\text{out}}(h)$$

$$A(n) = \operatorname{argmin}_{h \in H} E_n(h)$$

$$E_0[E_n(A(n))] > E_0[E_{\text{out}}(A(n))] \text{ is wrong}$$

3. (d)

$$X_n = \begin{bmatrix} x_1 & \dots & x_n & x_1^{\sim} & \dots & x_n^{\sim} \\ 1 & \dots & 1 & 1 & \dots & 1 \end{bmatrix}^T \quad \sigma^2 = E[\varepsilon^2] = E[\varepsilon \varepsilon^T]$$

$$X_n^T X_n = \begin{bmatrix} x_1 & \dots & x_n & x_1^{\sim} & \dots & x_n^{\sim} \\ 1 & \dots & 1 & 1 & \dots & 1 \end{bmatrix} \begin{bmatrix} -x_1 & \dots & -x_n \\ \vdots & & \vdots \\ -x_n & \dots & -x_1 \\ \vdots & & \vdots \\ -x_1^{\sim} & \dots & -x_n^{\sim} \\ \vdots & & \vdots \\ -x_n^{\sim} & \dots & -x_1^{\sim} \end{bmatrix}$$

$$= \begin{bmatrix} \sum x_{i,0}^2 + \underbrace{\sum \tilde{x}_{i,0}^2}_{\varphi N \sigma^2} & \dots & \sum x_{i,0} x_{i,d} + \underbrace{\sum \tilde{x}_{i,0} \tilde{x}_{i,d}}_{\varphi_0} \\ \vdots & & \vdots \\ \underbrace{\sum x_{i,d} x_{i,0} + \sum \tilde{x}_{i,d} \tilde{x}_{i,0}}_{\varphi_0} & \dots & \underbrace{\sum x_{i,d}^2 + \sum \tilde{x}_{i,d}^2}_{\varphi N \sigma^2} \end{bmatrix}$$

$$= 2X^T X + N \sigma^2 I_{d+1}$$

4. (e)

$$\varepsilon \in \mathbb{R}^{d+1}$$

$$x_h^T y_h$$

$$= \begin{bmatrix} x_1^T & \equiv & x_1^T & x_1^T & \equiv & x_1^T \\ 1 & & 1 & 1 & & 1 \end{bmatrix} \begin{bmatrix} y \\ y \end{bmatrix}$$

$$= \begin{bmatrix} x^T y + \tilde{x}^T y \end{bmatrix}$$

$$= \begin{bmatrix} 2x^T y + \varepsilon \end{bmatrix}$$

$$E[x_h^T y_h] = \begin{bmatrix} 2x^T y \end{bmatrix}$$

5. (d)

$$X^T X = Q \Gamma Q^T$$

$$X_{n \times (d+1)} \quad X_{(d+1) \times 1}$$

$$\underset{p}{z} = \underset{p}{X} \underset{p}{Q} \quad \begin{matrix} (d+1) \times (d+1) \\ n \times (d+1) \end{matrix}$$

$$\frac{u_i}{v_i} = \frac{(\Gamma + \lambda I)^{-1}}{(\Gamma)^{-1}}$$

$$\min_{w \in \mathbb{R}^{d+1}} \frac{1}{N} \|zw - y\|^2 + \frac{\lambda}{N} w^T w$$

$$\frac{2}{N} (z^T z w - z^T y) + \frac{2}{N} \lambda w = 0$$

$$w \leftarrow (z^T z + \lambda I)^{-1} z^T y$$

$$w \leftarrow (Q^T X^T X Q + \lambda I)^{-1} Q^T X^T y$$

$$w \leftarrow (Q^T Q \Gamma Q^T Q + \lambda I)^{-1} Q^T X^T y$$

$$w \leftarrow (\underbrace{\Gamma}_{(d+1) \times (d+1)} + \lambda I)^{-1} \underbrace{Q^T X^T y}_{(d+1) \times 1}$$

$$w \leftarrow \left(\frac{1}{\Gamma + \lambda I} \right)$$

6. (a)

$$\min_{\omega \in \mathbb{R}} \frac{1}{N} \sum_{n=1}^N \underbrace{(\omega \cdot x_n - y_n)^2}_{\omega^2 x_n^2 - 2\omega x_n y_n + y_n^2} + \frac{\lambda}{N} \omega^2 = \{$$

$$\frac{d}{d\omega} = \frac{1}{N} \sum_{n=1}^N (2\omega x_n^2 - 2x_n y_n) + \frac{2\lambda}{N} \omega = 0$$

$$\omega \cdot \sum_{n=1}^N 2x_n^2 - \sum_{n=1}^N 2x_n y_n + \frac{2\lambda}{N} \omega = 0$$

$$\omega \cdot \left(\sum_{n=1}^N 2x_n^2 + 2\lambda \right) = \sum_{n=1}^N 2x_n y_n$$

$$\omega = \frac{\sum_{n=1}^N x_n y_n}{\sum_{n=1}^N x_n^2 + \lambda}$$

7. (d)

$$\min_{y \in \mathbb{R}} \frac{1}{N} \sum_{n=1}^N (y - y_n)^2 + \frac{2K}{N} \Omega(y) = \xi$$

$y^2 - 2y y_n + y_n^2$

$$\frac{d\xi}{dy} = \frac{1}{N} \sum_{n=1}^N (2y - 2y_n) + \frac{2K}{N} \Omega'(y) = 0$$

$$y - \frac{1}{N} \sum_{n=1}^N y_n = -\frac{K}{N} \Omega'(y)$$

$$\begin{aligned} \Omega'(y) &= \frac{N}{K} \left(\frac{1}{N} \sum_{n=1}^N y_n - y \right) \\ &= \frac{N}{K} \left(\frac{1}{N} (N + 2K) y - \frac{K}{N} - y \right) \\ &= \frac{N}{K} \left(\frac{2Ky}{N} - \frac{K}{N} \right) \\ &= \frac{N}{K} \cdot \frac{K}{N} (2y - 1) \\ &= (2y - 1) \end{aligned}$$

$$\Omega(y) = \int (2y - 1) dy = y^2 - y + C = (y - 0.5)^2 + C$$

8. (b)

$$\underline{f}(x) = \Gamma^{-1} x$$

$$\min_{\tilde{w} \in \mathbb{R}^{d+1}} \frac{1}{N} \sum_{n=1}^N (\tilde{w}^T \underline{f}(x_n) - y_n)^2 + \frac{\lambda}{N} (\tilde{w}^T \tilde{w})$$

$$\min_{\tilde{w} \in \mathbb{R}^{d+1}} \frac{1}{N} \sum_{n=1}^N (\tilde{w}^T \Gamma^{-1} x_n - y_n)^2 + \frac{\lambda}{N} (\tilde{w}^T \tilde{w})$$

$$\tilde{w}^T \Gamma^{-1} \rightarrow w^T$$

$$\Omega((\Gamma^{-1})^T \tilde{w}) = \tilde{w}^T \tilde{w}$$

$$\begin{aligned} \Omega(w) &= w^T \cdot \Gamma \cdot \Gamma^T \cdot w \\ &= w^T \Gamma^2 w \end{aligned}$$

9. (b)

$$\sum_{i=0}^d \beta_i w_i^2 = w^T B w$$

Regularization.

$$\min_{w \in \mathbb{R}^{d+1}} \frac{1}{N} \sum_{n=1}^N (w^T x_n - y_n)^2 + \frac{\lambda}{N} w^T B w$$

$$\Rightarrow \frac{1}{N} (Xw - y)^T (Xw - y) + \frac{\lambda}{N} w^T B w$$

$$\frac{\lambda}{N} w^T B w \Leftrightarrow \frac{1}{N} (\tilde{X}w - \tilde{y})^T (\tilde{X}w - \tilde{y})$$

$$\lambda B w^T w \Leftrightarrow (\tilde{X}w - \tilde{y})^T (\tilde{X}w - \tilde{y})$$

10. (e)

$$\begin{aligned}\bar{E}_{\text{locv}} &= \frac{1}{N} \sum_{n=1}^N e_n = \frac{1}{N} \sum_{n=1}^N \text{err}(g_n^*(x_n), y_n) \\ &= \frac{1}{2N} (N + N) = 1\end{aligned}$$

11. (c)



Assume $\frac{N}{2}$ data set < 0 , $\frac{N}{2}$ data set > 0

$$\bar{E}_{\text{loocu}} = \frac{1}{N} \sum_{n=1}^N e_n = \frac{1}{N} \sum_{n=1}^N \text{err}(g_n^-(x_n), y_n)$$

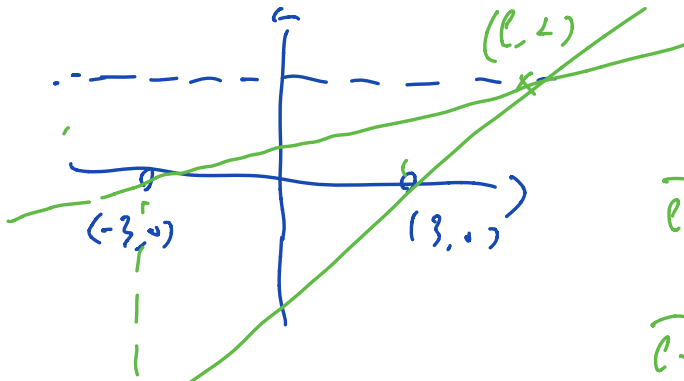
Observation:

Error only happens when the closet point to the origin is removed

\Rightarrow 2 points satisfied

$$\Rightarrow \bar{E}_{\text{loocu}} = 2/N$$

12. (e)



$$\frac{6}{l+3} \cdot 2 = h_1$$

$$\frac{6}{l-3} \cdot 2 = h_2$$

$$\bar{E}_{\text{loocu}} = \frac{1}{N} \sum_{n=1}^N e_n = \frac{1}{N} \sum_{n=1}^N \text{err}(g_n^-(x_n), y_n)$$

$$\bar{E}_{\text{loocu}, C} = \frac{1}{3} [2^2 + 1^2 + 1^2] = \frac{1}{3} \cdot 6 = 2$$

$$\begin{aligned} \bar{E}_{\text{loocu}, \text{lin}} &= \frac{1}{3} \left[4 + \left(\frac{12}{l+3} \right)^2 + \left(\frac{12}{l-3} \right)^2 \right] \\ &= \frac{1}{3} \left\{ 4 + \frac{(l+3)^2 + (l-3)^2}{(l+3)^2 (l-3)^2} \cdot 144 \right\} \end{aligned}$$

$$\Rightarrow l = 3\sqrt{9 + 4\sqrt{6}}$$

13. (d)

For N numbers, variance $\rightarrow N\sigma^2$

Since the mean is $1/N$ times the sum

the variance of the sampling distribution of the mean

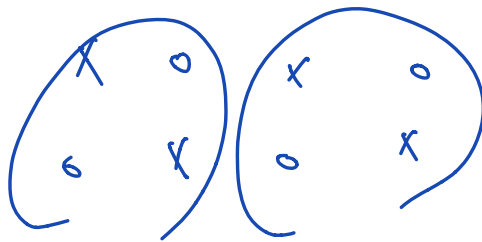
would be $1/N$ times the variance of the sum $\Rightarrow \sigma^2/N$

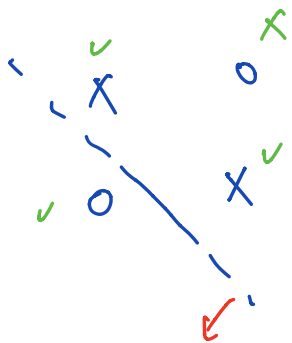
$$E_{\text{val}}(h) = \frac{1}{K} \sum_{i=1}^K \text{err}(h(x_i) - y)$$

$$\begin{aligned} V[E_{\text{val}}(h)] &= V\left[\frac{1}{K} \sum_{i=1}^K \text{err}(h(x_i) - y)\right] \\ &= \frac{1}{K^2} V\left[\sum_{i=1}^K \text{err}(h(x_i) - y)\right] \\ &= \frac{1}{K^2} \sum_{i=1}^K V[\text{err}(h(x_i) - y)] \\ &= \frac{1}{K} V[\text{err}(h(x) - y)] \end{aligned}$$

~~ff~~

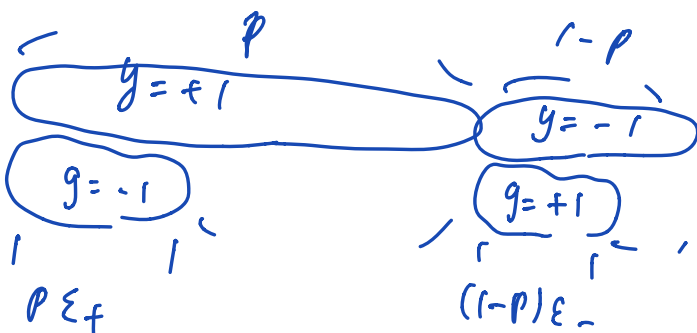
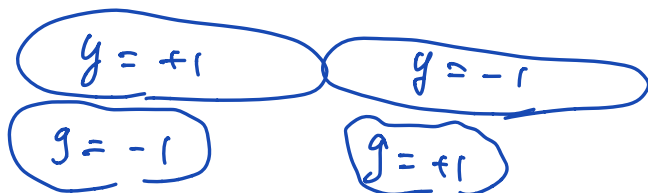
14. (c)


 \rightarrow 2 cases among 16
 can't be fit properly $\rightarrow 2/16$


 $\Rightarrow \min_{u \in \mathbb{R}^{2 \times 1}} E_{\text{in}}(u) = 1/4$

$$\Rightarrow 2/16 \cdot 1/4 = 2/64 \quad \text{XX}$$

15. (a)



$$E_{\text{out}}(g_c) = 1 - p$$

$$E_{\text{out}}(g_b) = p \epsilon_+ + (1 - p) \epsilon_-$$

$$(1 - p) = p \epsilon_+ + (1 - p) \epsilon_-$$

$$1 - p = p \epsilon_+ + \epsilon_- - p \epsilon_-$$

$$p(1 + \epsilon_+ - \epsilon_-) = 1 - \epsilon_-$$

$$p = \frac{1 - \epsilon_-}{1 + \epsilon_+ - \epsilon_-}$$

```

import scipy
import numpy as np
from liblinearutil import *

data = np.genfromtxt('hw4_train.dat')
data_test = np.genfromtxt('hw4_test.dat')

X = data[:, :-1]
X_test = data_test[:, :-1]

phi = X
phi_test = X_test

d = X.shape[1]
for i in range(d):
    for j in range(i, d):
        # print(i, j)
        phi = np.c_[phi, np.multiply(np.expand_dims(X[:, i], axis=1), np.expand_dims(X[:, j],
axis=1))]
        phi_test = np.c_[phi_test, np.multiply(np.expand_dims(X_test[:, i], axis=1),
np.expand_dims(X_test[:, j], axis=1))]
phi = np.c_[np.ones(data.shape[0]), phi]
phi_test = np.c_[np.ones(data_test.shape[0]), phi_test]
phi = np.c_[phi, np.expand_dims(data[:, -1], axis=1)]
phi_test = np.c_[phi_test, np.expand_dims(data_test[:, -1], axis=1)]

print("Q16: --- E_out ---")
prob = problem(phi[:, -1], phi[:, 0:-1])
for lmd_power in [-4, -2, 0, 2, 4]:
    lmd = 10**lmd_power
    C = 1/(2*lmd)
    param = parameter(f'-s 0 -c {C} -e 0.0000001 -q')
    model = train(prob, param, 'q')
    p_labs, p_acc, p_vals = predict(phi_test[:, -1], phi_test[:, 0:-1], model)

print("Q17: --- E_in ---")
prob = problem(phi[:, -1], phi[:, 0:-1])
for lmd_power in [-4, -2, 0, 2, 4]:
    lmd = 10**lmd_power
    C = 1/(2*lmd)
    param = parameter(f'-s 0 -c {C} -e 0.0000001 -q')
    model = train(prob, param, 'q')
    p_labs, p_acc, p_vals = predict(phi[:, -1], phi[:, 0:-1], model)

print("Q18: ---E_out---")
prob = problem(phi[:120, -1], phi[:120, 0:-1])
for lmd_power in [-4, -2, 0, 2, 4]:
    lmd = 10**lmd_power
    C = 1/(2*lmd)
    param = parameter(f'-s 0 -c {C} -e 0.0000001 -q')
    model = train(prob, param, 'q')
    p_labs, p_acc, p_vals = predict(phi[120:, -1], phi[120:, 0:-1], model)

print("Q19: ---E_out---")
prob = problem(phi[:, -1], phi[:, 0:-1])
for lmd_power in [-4, -2, 0, 2, 4]:
    lmd = 10**lmd_power
    C = 1/(2*lmd)
    param = parameter(f'-s 0 -c {C} -e 0.0000001 -q')
    model = train(prob, param, 'q')
    p_labs, p_acc, p_vals = predict(phi_test[:, -1], phi_test[:, 0:-1], model)

print("Q19: ---E_out---")
prob = problem(phi[:, -1], phi[:, 0:-1])
for lmd_power in [-4, -2, 0, 2, 4]:
    lmd = 10**lmd_power
    C = 1/(2*lmd)

```



```

param = parameter(f'-s 0 -c {C} -e 0.0000001 -q')
model = train(prob, param, 'q')
p_labs, p_acc, p_vals = predict(phi_test[:, -1], phi_test[:, 0:-1], model)

print("Q20: ---E_out---")
prob = problem(phi[:, -1], phi[:, 0:-1])
for lmd_power in [-4, -2, 0, 2, 4]:
    acc = 0
    for k in range(5):
        prob = problem(np.delete(phi[:, -1], slice(k*40, (k+1)*40), axis=0), np.delete(phi[:,
0:-1], slice(k*40, (k+1)*40), axis=0))
        lmd = 10**lmd_power
        C = 1/(2*lmd)
        param = parameter(f'-s 0 -c {C} -e 0.0000001 -q')
        model = train(prob, param, '-q')
        p_labs, p_acc, p_vals = predict(phi[k*40:(k+1)*40, -1], phi[k*40:(k+1)*40, 0:-1], model,
'-q')
        acc += p_acc[0]
    acc /= 5
    print('acc= ', acc)

# reference for multiple choices :
http://www.kaspercpa.com/statisticalreview.htm#:~:text=The%20variance%20of%20the%20sum,the%20random%20variables%20are%20independent.&text=The%20covariance%20of%20a%20random,does%20not%20change%20their%20covariances.

```