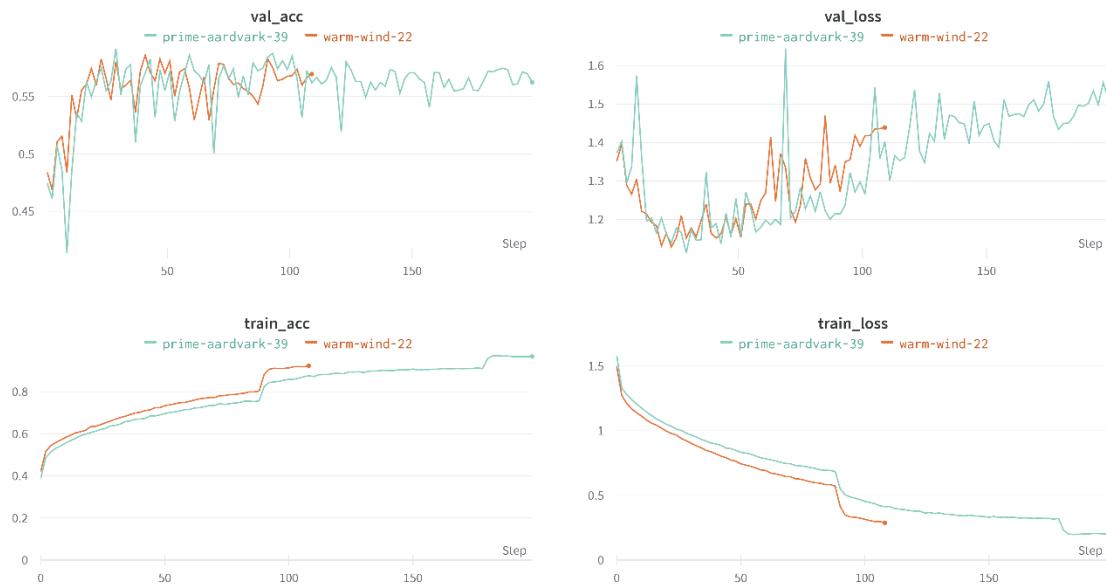


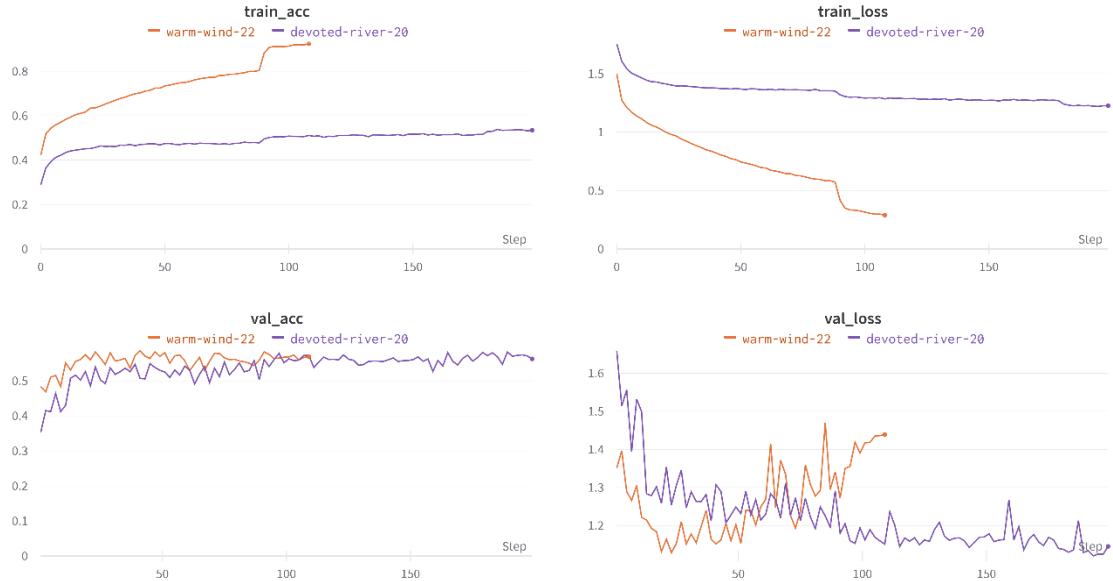
1. (1%) 實作 early-stopping，繪製 training, validation loss 的 learning curve，比較實作前後的差異，並說明 early-stopping 的運作機制

As you can see below, if I use early stopping technique, it'll break the training loop when overfitting. The orange line is what I set early stopping with threshold 5. That is, if the model loss rise up 5 times consequently, then stop training. The other one doesn't set early stopping and you can see it'll complete the training loop even overfitting occur.



2. (1%) 嘗試使用 augmentation，說明實作細節並比較有無該 trick 對結果表現的影響(validation 或是 testing 擇一即可)，且需說明為何使用這些 augmentation 的原因。(ref: <https://pytorch.org/vision/stable/transforms.html>)

As you can see below, if I use data augmentation, it can conquer overfitting. The other configurations are the same and the breakpoint of orange line is because of early-stopping. I set data augmentation technique on purple one and the others didn't.



I use RandomChoice to choose transform_set randomly including RandomHorizontalFlip, ColorJitter, RandomRatio. ColorJitter will adjust the brightness, contrast, saturation and hue of the input image randomly. So, it can increase the diversity of training dataset properly. Though, lecture TA is not very suggestive to use RandomVerticalFlip skill on training image, because it'll transform the image that no human can recognize it. So, I use RandomHorizontalFlip instead.

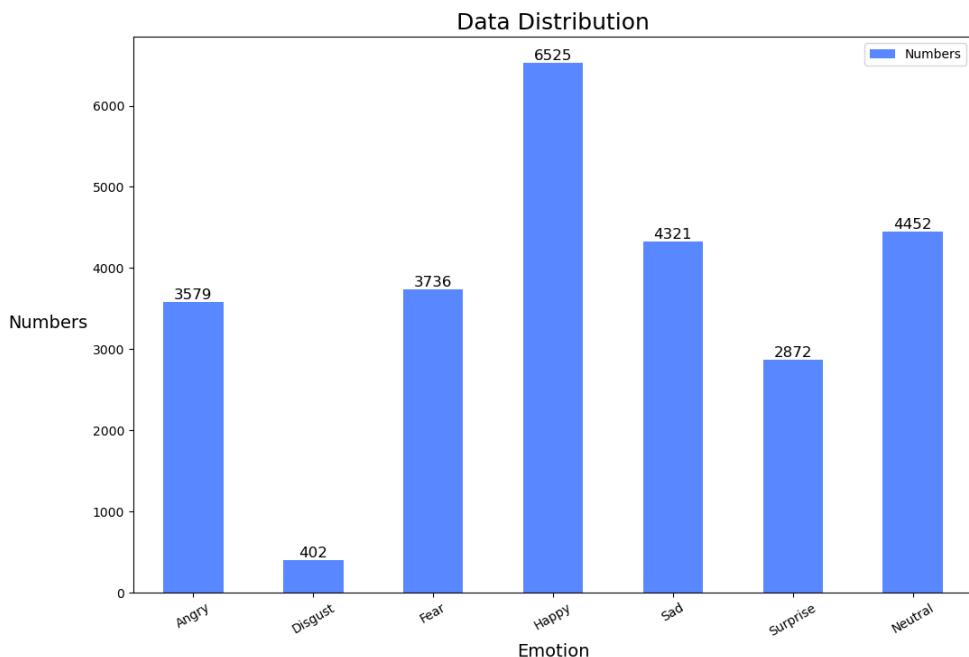
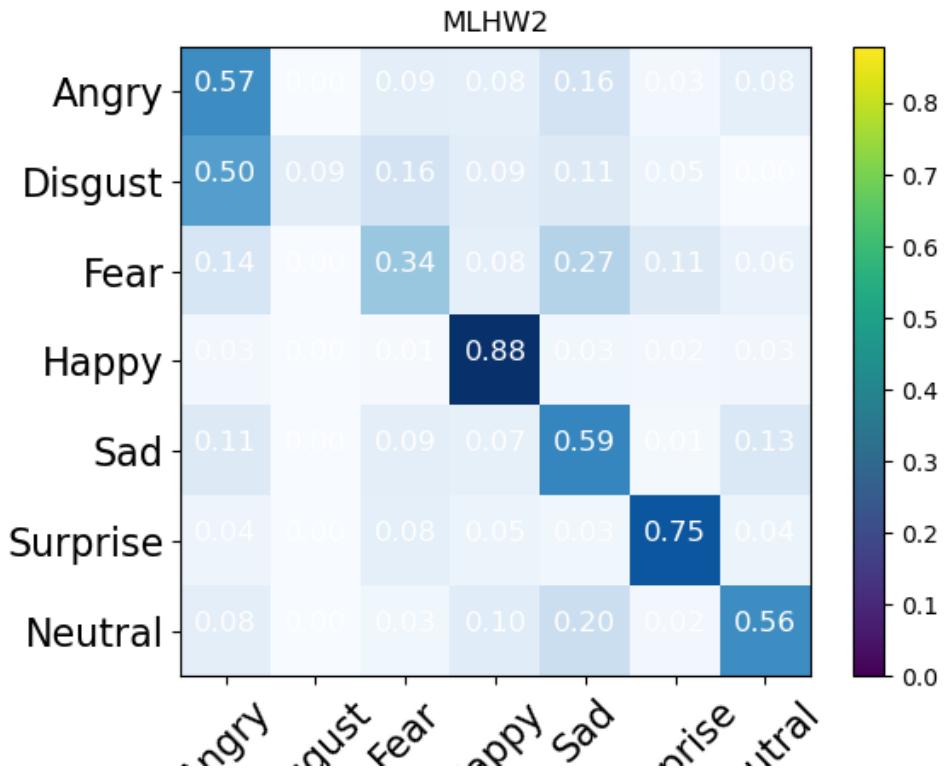
```
transform_set = [
```

```
    transforms.RandomHorizontalFlip(p=0.5),
    transforms.ColorJitter(brightness=(0, 5), contrast=(0, 5), saturation=(0, 5), hue=(-0.1, 0.1)),
    transforms.RandomRotation(30, center=(0, 0), expand=False)
]
```

3. (1%) 畫出 confusion matrix 分析哪些類別的圖片容易使 model 搞混，找出模型出錯的例子，並分析可能的原因。

(ref: https://en.wikipedia.org/wiki/Confusion_matrix)

As you can see the confusion matrix below. The second class(emotion Disgust) is the worst result of the classification and the Happy class is the best. Also, the Fear class is not good enough. I think the main reason is data imbalance that shown below of second one. The prior probability of these two classes are 0.0155 and 0.1443 respectively. Under this circumstance, the model can't learn this class by enough images properly. And the bad result of Fear class. I think it's just not learn very well with bad model structure and bad configuration.



4. (1%) 請統計訓練資料中不同類別的數量比例，並說明：對 testing 或是 validation 來說，不針對特定類別，直接選擇機率最大的類別會是最好的結果嗎？
 (ref: <https://arxiv.org/pdf/1608.06048.pdf>, or hints: imbalanced classification)

The result of data distribution is shown above. The prior probability of the highest number training class is $6525/25887=0.252$. If not targeting a specific category and just choose the Happy class, it would be worse than the result that generate by normal classification.

5. (4%) Refer to math problem:

https://hackmd.io/@lH2AB7kCSAS3NPw2FffsGg/r1otQp7Gi?fbclid=IwAR0cs5CajVy_zhDmHEDgze2V1_Jlxp95N45BF6hg1l6CgG-6IViYGAIGReE

MLHW

Q1: Generative classification model of K classes $C_1 \dots C_K \Rightarrow$ prior probabilities $P(C_k) = \pi_k$

\Rightarrow conditional probability $P(x_i | C_k)$

Input feature vector

Suppose given training dataset $((\vec{x}_i, \vec{t}_i))_{i=1}^N \rightarrow \vec{t}_i = (t_i^1, \dots, t_i^K) \in \{0, 1\}^K$

<sol> The probability of one data point

$$\text{Step 1: Construct } P(\vec{x}, \vec{t}) = P(\vec{x} | \vec{t}) P(\vec{t}) = \prod_{k=1}^K [P(\vec{x} | C_k) P(C_k)]^{t_k^k} = \prod_{k=1}^K [P(\vec{x} | C_k) \pi_k]^{t_k^k}$$

$L(\theta) = f(x_1, x_2, \dots, x_N; \theta) \rightarrow$ Joint Probability Distribution

$$L(\theta) = \prod_{k=1}^K \left[P(x_1 | C_k) \pi_k \right]^{t_1^k} \prod_{k=1}^K \left[P(x_2 | C_k) \pi_k \right]^{t_2^k} \cdots \prod_{k=1}^K \left[P(x_N | C_k) \pi_k \right]^{t_N^k}$$

$$= \prod_{i=1}^N \prod_{k=1}^K \left[P(x_i | C_k) \pi_k \right]^{t_i^k}$$

Step 2: Apply log

$$\log(L(\theta)) = \log \left\{ \left[P(x_1 | C_1) \pi_1 \right]^{t_1^1} \left[P(x_1 | C_2) \pi_2 \right]^{t_1^2} \cdots \left[P(x_1 | C_K) \pi_K \right]^{t_1^K} \right. \\ \left. \left[P(x_2 | C_1) \pi_1 \right]^{t_2^1} \left[P(x_2 | C_2) \pi_2 \right]^{t_2^2} \cdots \left[P(x_2 | C_K) \pi_K \right]^{t_2^K} \right. \\ \vdots \quad \vdots \quad \vdots \quad \vdots \\ = t_1^1 \log [P(x_1 | C_1) \pi_1] + t_1^2 \log [P(x_1 | C_2) \pi_2] + \cdots + t_1^K \log [P(x_1 | C_K) \pi_K] \\ + t_2^1 \log [P(x_2 | C_1) \pi_1] + t_2^2 \log [P(x_2 | C_2) \pi_2] + \cdots + t_2^K \log [P(x_2 | C_K) \pi_K] \\ \vdots \quad \vdots \quad \vdots$$

$$= \sum_{k=1}^K t_i^k \log [P(x_i | C_k) \pi_k] + \sum_{k=1}^K t_2^k \log [P(x_2 | C_k) \pi_k] + \cdots + \sum_{k=1}^K t_N^k \log [P(x_N | C_k) \pi_k]$$

$$= \sum_{i=1}^N \sum_{k=1}^K t_i^k \log [P(x_i | C_k) \pi_k] = \sum_{i=1}^N \sum_{k=1}^K t_i^k [\log P(x_i | C_k) + \log \pi_k]$$

Step 3: Derivative
and let it to be 0

If we have constraint and want to find maximum or minimum value,
we can use Lagrange multipliers.

We can think about it that if we don't add constraint in the expression before. differentiate Π_k

$$\Rightarrow \frac{\partial}{\partial \Pi_k} \left(\sum_{i=1}^N \sum_{k=1}^K t_i^k \left[\log(P(X_i|C_k)) + \log \Pi_k \right] \right)$$

$$= \frac{\partial}{\partial \Pi_k} \left(\sum_{i=1}^N \sum_{k=1}^K t_i^k \overset{\text{log}}{\log} P(X_i|C_k) \right) + \frac{\partial}{\partial \Pi_k} \left(\sum_{i=1}^N \sum_{k=1}^K t_i^k \log \Pi_k \right)$$

$$\frac{\partial}{\partial \Pi_k} \begin{pmatrix} t_1^1 \log \Pi_1 + & \boxed{t_1^2 \log \Pi_2 + \dots + t_1^K \log \Pi_K} \\ t_2^1 \log \Pi_1 + & t_2^2 \log \Pi_2 + \dots + t_2^K \log \Pi_K \\ \vdots & \vdots & \vdots & \vdots \\ t_N^1 \log \Pi_1 + & t_N^2 \log \Pi_2 + \dots + t_N^K \log \Pi_K \end{pmatrix}$$

If apply any value that $\in \{1, \dots, K\}$
assume $k=2$

↓
Reserve to differentiate

$$\Rightarrow t_1^2 \frac{1}{\Pi_2} + t_2^2 \frac{1}{\Pi_2} + t_3^2 \frac{1}{\Pi_2} + \dots + t_N^2 \frac{1}{\Pi_2} = \sum_{i=1}^N t_i^2 \cdot \frac{1}{\Pi_2}$$

$$\Rightarrow \text{so if we choose } k \in \{1, \dots, K\} \Rightarrow \sum_{i=1}^N t_i^k \frac{1}{\Pi_k} \text{ where } k \in \{1, \dots, K\}$$

$$\text{We must let } \frac{\partial L(\theta)}{\partial \Pi_k} \text{ equals to } 0 \Rightarrow \sum_{i=1}^N t_i^k \frac{1}{\Pi_k} = 0$$

* Contraction:

$$\sum_{i=1}^N t_i^k = N_k \quad (\text{the number of data points belong to } C_k) > 0 \text{ and } \Pi_k > 0$$

then $\sum_{i=1}^N t_i^k \frac{1}{\Pi_k}$ must greater than 0 but we want the result of differentiation equals to zero. So, we must add a constraint term.

MLHW Q1 cont.

(Method 1) The constraint is $\sum_{k=1}^K \pi_k = 1$ because the summation of all class of probability is 1

$$\Rightarrow \frac{\partial}{\partial \pi_k} \left(\sum_{i=1}^N \sum_{k=1}^K t_i^k \log \pi_k + \lambda \left(\sum_{k=1}^K \pi_k - 1 \right) \right)$$

$$= \sum_{i=1}^N t_i^k \frac{1}{\pi_k} + \lambda = 0 \Rightarrow N_k \cdot \frac{1}{\pi_k} + \lambda = 0 \Rightarrow \pi_k = -\frac{N_k}{\lambda}$$

$$\frac{\partial}{\partial \lambda} \left(\sum_{i=1}^N \sum_{k=1}^K t_i^k \log \pi_k + \lambda \left(\sum_{k=1}^K \pi_k - 1 \right) \right) = \sum_{k=1}^K \pi_k - 1 = 0 \Rightarrow 1 = \sum_{k=1}^K \pi_k.$$

$$1 = \sum_{k=1}^K \pi_k = \sum_{k=1}^K -\frac{N_k}{\lambda} = \frac{-N}{\lambda} \Rightarrow \lambda = -N \Rightarrow \boxed{\pi_k = +\frac{N_k}{N}}$$

Q.E.D.

(Method 2) $f(\pi_k) = \sum_{i=1}^N \sum_{k=1}^K t_i^k \log \pi_k$ and $g(\pi_k) = \sum_{k=1}^K \pi_k = 1$ ^{constraint function}

$$f'_{\pi_k} = \frac{\partial f(\pi_k)}{\partial \pi_k} = \lambda g'_{\pi_k} = \lambda \frac{\partial g(\pi_k)}{\partial \pi_k}$$

↓

$$\sum_{i=1}^N t_i^k \frac{1}{\pi_k} = \lambda \cdot 1 \Rightarrow N_k \cdot \frac{1}{\pi_k} = \lambda \Rightarrow \pi_k = \frac{N_k}{\lambda}$$

$$\text{apply to } g(\pi_k) = \sum_{k=1}^K \frac{N_k}{\lambda} = 1 = \frac{N}{\lambda} \Rightarrow \lambda = N \Rightarrow \boxed{\pi_k = \frac{N_k}{N}}$$

Q.E.D.

ML HW 2

Q2(a) Given $\vec{w} \in \mathbb{R}^m$, $\vec{A} \in \mathbb{R}^{m \times m}$ (they're independent)

$$\frac{\partial \vec{w}^T \vec{A} \vec{w}}{\partial \vec{w}} = \vec{A}^T \vec{w} + \vec{A} \vec{w}$$

$$\begin{aligned} \text{Theorem 1: } (\frac{\partial \vec{U} \vec{V}}{\partial \vec{X}})_i &= \frac{\partial \vec{U} \vec{V}}{\partial x_i} = U \frac{\partial V}{\partial x_i} + V \frac{\partial U}{\partial x_i} = U \left(\frac{\partial V}{\partial \vec{X}} \right)_i + V \left(\frac{\partial U}{\partial \vec{X}} \right)_i \\ &= \left(U \frac{\partial V}{\partial \vec{X}} + V \frac{\partial U}{\partial \vec{X}} \right)_i \end{aligned}$$

Theorem 2: Refer to theorem 1

$$\begin{aligned} \frac{\partial \vec{U}^T \vec{V}}{\partial \vec{X}} &= \frac{\partial \sum_k u_k v_k}{\partial \vec{X}} = \sum_k \frac{\partial u_k v_k}{\partial \vec{X}} = \sum_k \left(u_k \frac{\partial v_k}{\partial \vec{X}} + v_k \frac{\partial u_k}{\partial \vec{X}} \right) \\ &= \sum_k \frac{\partial v_k}{\partial \vec{X}} u_k + \sum_k \frac{\partial u_k}{\partial \vec{X}} v_k = \frac{\partial \vec{V}}{\partial \vec{X}} \vec{U} + \frac{\partial \vec{U}}{\partial \vec{X}} \vec{V} \end{aligned}$$

$$\begin{aligned} \text{Theorem 3: } (\frac{\partial \vec{A} \vec{U}}{\partial \vec{X}})_{ij} &= \frac{\partial \sum_k a_{jk} u_k}{\partial x_i} = \sum_k a_{jk} \frac{\partial u_k}{\partial x_i} \\ &= \sum_k \left(\frac{\partial \vec{U}}{\partial \vec{X}} \right)_{ik} (\vec{A}^T)_{kj} = \left(\frac{\partial \vec{U}}{\partial \vec{X}} \vec{A}^T \right)_{ij} \end{aligned}$$

Theorem 4: Refer to theorem 2 and theorem 3

$$\frac{\partial \vec{U}^T \vec{A} \vec{V}}{\partial \vec{X}} = \frac{\partial \vec{U}^T (\vec{A} \vec{X})}{\partial \vec{X}} = \frac{\partial \vec{U}}{\partial \vec{X}} \vec{A} \vec{X} + \frac{\partial (\vec{A} \vec{X})}{\partial \vec{X}} \vec{U} = \frac{\partial \vec{U}}{\partial \vec{X}} \vec{A} \vec{X} + \frac{\partial \vec{V}}{\partial \vec{X}} \vec{A}^T \vec{U}$$

$$\text{Theorem 5: } \frac{\partial \vec{X}}{\partial \vec{X}} = \vec{I}$$

Refer to theorem 4 and theorem 5

$$\frac{\partial \vec{X}^T \vec{A} \vec{X}}{\partial \vec{X}} = \frac{\partial \vec{X}}{\partial \vec{X}} \vec{A} \vec{X} + \frac{\partial \vec{X}}{\partial \vec{X}} \vec{A}^T \vec{X} = \vec{I} \vec{A} \vec{X} + \vec{I} \vec{A}^T \vec{X} = \boxed{(\vec{A} + \vec{A}^T) \vec{X}}$$

Q.E.D

Q2(b) Given $\vec{A}, \vec{B} \in \mathbb{R}^{m \times m}$. Show that $\frac{\partial \text{tr}(\vec{A}\vec{B})}{\partial a_{ij}} = b_{\bar{j}}$

(Method 1)

$$\vec{A} = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1m} \\ a_{21} & \ddots & & \\ \vdots & & \ddots & \\ a_{m1} & & a_{mm} \end{bmatrix} \quad \vec{B} = \begin{bmatrix} b_{11} & b_{12} & \dots & b_{1m} \\ b_{21} & \ddots & & \\ \vdots & & \ddots & \\ b_{m1} & & b_{mm} \end{bmatrix}$$

$$\vec{AB} = \begin{bmatrix} a_{11}b_{11} + a_{12}b_{21} + \dots + a_{1m}b_{m1} & a_{11}b_{12} + a_{12}b_{22} + \dots + a_{1m}b_{m2} & \dots & a_{11}b_{1m} + \dots + a_{1m}b_{mm} \\ a_{21}b_{11} + a_{22}b_{21} + \dots + a_{2m}b_{m1} & \ddots & & \vdots \\ \vdots & & & \\ a_{m1}b_{11} + a_{m2}b_{21} + \dots + a_{mm}b_{m1} & \ddots & & a_{m1}b_{1m} + \dots + a_{mm}b_{mm} \end{bmatrix}$$

$$= \begin{bmatrix} \sum_i a_{1k}b_{k1} & \sum_i a_{1k}b_{k2} & \dots & \sum_i a_{1k}b_{km} \\ \sum_i a_{2k}b_{k1} & \ddots & & \vdots \\ \vdots & & & \\ \sum_i a_{mk}b_{k1} & \ddots & & \sum_i a_{mk}b_{km} \end{bmatrix}$$

$$\text{Trace}(\vec{AB}) = \sum_k a_{1k}b_{k1} + \sum_k a_{2k}b_{k2} + \sum_k a_{3k}b_{k3} + \dots + \sum_k a_{mk}b_{km}$$

$$= \sum_k \sum_l a_{kl}b_{lk}$$

$$\frac{\partial \text{tr}(\vec{AB})}{\partial a_{ij}} = \frac{\partial \sum_k \sum_l a_{kl}b_{lk}}{\partial a_{ij}} = \sum_k \sum_l b_{lk} \frac{\partial a_{kl}}{\partial a_{ij}} = \sum_k \sum_l b_{lk} \delta_{ik} \delta_{jl} = b_{\bar{j}i}$$

Q.E.D

(Method 2) Refer to Q2(a)

$$\frac{\partial \text{tr}(\vec{AB})}{\partial a_{ij}} = \text{tr}\left(\vec{B} \frac{\partial \vec{A}}{\partial a_{ij}}\right) = \text{tr}\left(\vec{B} \vec{e}_i \vec{e}_{\bar{j}}^T\right) = \text{tr}\left(\vec{e}_{\bar{j}}^T \vec{B} \vec{e}_i\right) = b_{\bar{j}i}$$

Q.E.D

\vec{e}_i is represent i -th element is 1, others element is 0

$\vec{e}_i \vec{e}_{\bar{j}}^T$ is represent (i, \bar{j}) element has value 1, the others is 0

Q2(c) Show that $\frac{\partial \log(\det(\Sigma))}{\partial \sigma_{ij}} = e_j^T \Sigma^{-1} e_i$ where $\Sigma = \begin{bmatrix} \sigma_{11} & \sigma_{12} & \dots & \sigma_{1m} \\ \sigma_{21} & \sigma_{22} & \dots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{m1} & \sigma_{m2} & \dots & \sigma_{mm} \end{bmatrix} \in \mathbb{R}^{m \times m}$

a non-singular covariance matrix and \vec{e}_j is the unit vector along the j -th axis (e.g. $\vec{e}_3 = [0, 0, 1, 0, \dots, 0]^T$)

$$\text{Given } |\vec{A}| = \sum_{j=1}^n (-1)^{i+j} a_{ij} |\vec{A}_{ij}| = \sum_{i=1}^m (-1)^{i+j} a_{ij} |\vec{A}_{ij}|$$

Cramer Rule $\vec{A}\vec{x} = \vec{e}_i = [0 \dots \underset{i\text{-th element}}{1} \dots 0]^T \Rightarrow \vec{x} = \vec{A}^{-1} \vec{e}_i$.

$$x^{(i)} = e_j^T \vec{A}^{-1} e_i = \frac{(-1)^{i+j} |\vec{A}_{ij}|}{|\vec{A}|} = \frac{\partial \log |\vec{A}|}{\partial a_{ij}}$$

$$\text{Proof: } \frac{\partial \log(|\Sigma|)}{\partial \sigma_{ij}} = \frac{1}{|\Sigma|} \times \frac{\partial |\vec{\Sigma}|}{\partial \sigma_{ij}}$$

$$\text{Refer to the given theorem} \rightarrow \frac{\partial |\vec{\Sigma}|}{\partial \sigma_{ij}} = (-1)^{i+j} |\vec{\Sigma}_{ij}|$$

$$\Rightarrow \frac{\partial \log(|\Sigma|)}{\partial \sigma_{ij}} = \frac{(-1)^{i+j} |\vec{\Sigma}_{ij}|}{|\vec{\Sigma}|} = \boxed{e_j^T \vec{\Sigma}^{-1} e_i} \quad \text{Q.E.D.}$$

Q3: Given conditional probability density function are given by Gaussian Distribution with a shared covariance matrix, namely $p(\vec{x}|C_k) = \mathcal{N}(\vec{x}|\mu_k, \Sigma)$

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

Step 1: $P(\vec{x}, \vec{t}_k) = P(\vec{x}|\vec{C}) P(\vec{C}_k) = \prod_{k=1}^K \left[P(\vec{x}|C_k) P(C_k) \right]^{t_k^k} = \prod_{k=1}^K \left[P(\vec{x}|C_k) \pi_k \right]^{t_k^k}$

Construct $L(\theta)$

$$\begin{aligned} &= \prod_{i=1}^N \prod_{k=1}^K \left[P(x_i|C_k) \pi_k \right]^{t_i^k} = \prod_{i=1}^N \prod_{k=1}^K \left[\mathcal{N}(\vec{x}_i|\vec{\mu}_k, \Sigma) \pi_k \right]^{t_i^k} \\ &= \prod_{i=1}^N \prod_{k=1}^K \left[\frac{1}{(2\pi)^{\frac{D}{2}}} \frac{1}{|\Sigma|^{\frac{1}{2}}} \exp\left\{-\frac{1}{2}(\vec{x}_i - \vec{\mu}_k)^T \Sigma^{-1} (\vec{x}_i - \vec{\mu}_k)\right\} \right]^{t_i^k} \end{aligned}$$

Step 2: Apply log to $L(\theta)$

$$\begin{aligned} \log(L(\theta)) &= \sum_{i=1}^N \sum_{k=1}^K t_i^k \left[\log(\mathcal{N}(\vec{x}_i|\vec{\mu}_k, \Sigma)) + \log \pi_k \right] \\ &= \sum_{i=1}^N \sum_{k=1}^K t_i^k \left[\log\left(\frac{1}{\sqrt{2\pi} \sqrt{|\Sigma|}}\right) + -\frac{1}{2}(\vec{x}_i - \vec{\mu}_k)^T \Sigma^{-1} (\vec{x}_i - \vec{\mu}_k) + \log \pi_k \right] \\ &= \sum_{i=1}^N \sum_{k=1}^K t_i^k \log\left(\frac{1}{\sqrt{2\pi} \sqrt{|\Sigma|}}\right) - \sum_{i=1}^N \sum_{k=1}^K \left[\frac{1}{2}(\vec{x}_i - \vec{\mu}_k)^T \Sigma^{-1} (\vec{x}_i - \vec{\mu}_k) \right] t_i^k + \sum_{i=1}^N \sum_{k=1}^K t_i^k \log \pi_k \end{aligned}$$

Step 3: Differentiate $\frac{\partial \log(L(\theta))}{\partial \mu_k} = \frac{\partial}{\partial \mu_k} \left[- \sum_{i=1}^N \sum_{k=1}^K \left[\frac{1}{2}(\vec{x}_i - \vec{\mu}_k)^T \Sigma^{-1} (\vec{x}_i - \vec{\mu}_k) \right] t_i^k \right]$

$$\begin{aligned} &\Rightarrow \frac{\partial}{\partial \mu_k} \left[\sum_{i=1}^N -\frac{t_i^k}{2} \sum_{k=1}^K (\vec{x}_i^T - \vec{\mu}_k^T) \Sigma^{-1} (\vec{x}_i - \vec{\mu}_k) \right] = \frac{\partial}{\partial \mu_k} \left[\sum_{i=1}^N -\frac{t_i^k}{2} \sum_{k=1}^K \left(\underbrace{\vec{x}_i^T \Sigma^{-1} \vec{x}_i}_{0} - \underbrace{\vec{\mu}_k^T \Sigma^{-1} \vec{x}_i}_{\vec{\mu}_k^T \Sigma^{-1} \vec{\mu}_k} - \underbrace{\vec{x}_i^T \Sigma^{-1} \vec{\mu}_k}_{\vec{\mu}_k^T \Sigma^{-1} \vec{\mu}_k} + \underbrace{\vec{\mu}_k^T \Sigma^{-1} \vec{\mu}_k}_{\vec{\mu}_k^T \Sigma^{-1} \vec{\mu}_k} \right) \right] \\ &= \sum_{i=1}^N -\frac{t_i^k}{2} \left(\frac{\partial}{\partial \mu_k} (\vec{x}_i^T \Sigma^{-1} \vec{x}_i - \vec{\mu}_k^T \Sigma^{-1} \vec{x}_i - \vec{x}_i^T \Sigma^{-1} \vec{\mu}_k + \vec{\mu}_k^T \Sigma^{-1} \vec{\mu}_k) \right) \\ &= \sum_{i=1}^N -\frac{t_i^k}{2} \left(0 - \Sigma^{-1} \vec{x}_i - (\vec{x}_i^T \Sigma^{-1})^T + (\Sigma^{-1} + (\Sigma^{-1})^T) \vec{\mu}_k \right) = \sum_{i=1}^N -\frac{t_i^k}{2} \left[-\Sigma^{-1} \vec{x}_i - (\Sigma^{-1})^T \vec{x}_i + 2 \Sigma^{-1} \vec{\mu}_k \right] \\ &= \sum_{i=1}^N -\frac{t_i^k}{2} \left[-2 \Sigma^{-1} \vec{x}_i + 2 \Sigma^{-1} \vec{\mu}_k \right] = \sum_{i=1}^N t_i^k \left[\Sigma^{-1} \vec{x}_i - \Sigma^{-1} \vec{\mu}_k \right] = \sum_{i=1}^N t_i^k \Sigma^{-1} (\vec{x}_i - \vec{\mu}_k) = 0 \\ \sum_{i=1}^N t_i^k \Sigma^{-1} (\vec{x}_i - \vec{\mu}_k) = 0 &\stackrel{* \sum \text{ to eliminate } \Sigma^{-1}}{=} \sum_{i=1}^N t_i^k \vec{x}_i - N_k \vec{\mu}_k \Rightarrow \boxed{\vec{\mu}_k = \frac{1}{N_k} \sum_{i=1}^N t_i^k \vec{x}_i} \quad * \text{ Q.E.D.} \end{aligned}$$

MLHW Q3 Cont.

$$Q3(b) \quad \log(L(\theta)) = \sum_{i=1}^N \sum_{k=1}^K t_i^k \log \left(\frac{1}{\sqrt{2\pi} \sqrt{\det(\Sigma)}} \right) - \sum_{i=1}^N \sum_{k=1}^K \left[\frac{1}{2} (\mathbf{x}_i - \boldsymbol{\mu}_k)^T \Sigma^{-1} (\mathbf{x}_i - \boldsymbol{\mu}_k) \right] t_i^k + \sum_{i=1}^N \sum_{k=1}^K t_i^k \log \pi_k$$

$$\textcircled{1} \quad \frac{\partial}{\partial \Sigma} \left[\sum_{i=1}^N \sum_{k=1}^K t_i^k \left(-\frac{1}{2} \log 2\pi + -\frac{1}{2} \log(\det(\Sigma)) \right) \right] - \frac{\partial}{\partial \Sigma} \left[\sum_{i=1}^N \sum_{k=1}^K \left[\frac{1}{2} (\mathbf{x}_i - \boldsymbol{\mu}_k)^T \Sigma^{-1} (\mathbf{x}_i - \boldsymbol{\mu}_k) \right] t_i^k \right]$$

$$\textcircled{1} \quad -\frac{1}{2} \frac{\partial}{\partial \Sigma} \left[\sum_{i=1}^N \sum_{k=1}^K t_i^k \log(\det(\Sigma)) \right] = -\frac{1}{2} (\Sigma^{-1})^T \cdot \sum_{i=1}^N \sum_{k=1}^K t_i^k$$

$$\begin{aligned} \textcircled{2} \quad & -\frac{1}{2} \frac{\partial}{\partial \Sigma} \left[\sum_{i=1}^N \sum_{k=1}^K \underbrace{(\mathbf{x}_i^T - \boldsymbol{\mu}_k^T) \Sigma^{-1} (\mathbf{x}_i - \boldsymbol{\mu}_k)}_{\text{scalar}} t_i^k \right] = -\frac{1}{2} \frac{\partial}{\partial \Sigma} \left[\sum_{i=1}^N \sum_{k=1}^K (\mathbf{x}_i^T \Sigma^{-1} \mathbf{x}_i - \boldsymbol{\mu}_k^T \Sigma^{-1} \mathbf{x}_i - \mathbf{x}_i^T \Sigma^{-1} \boldsymbol{\mu}_k + \boldsymbol{\mu}_k^T \Sigma^{-1} \boldsymbol{\mu}_k) t_i^k \right] \\ & = -\frac{1}{2} \frac{\partial}{\partial \Sigma} \left[\sum_{i=1}^N \sum_{k=1}^K \text{trace}((\mathbf{x}_i^T - \boldsymbol{\mu}_k^T) \Sigma^{-1} (\mathbf{x}_i - \boldsymbol{\mu}_k)) t_i^k \right] \\ & = +\frac{1}{2} \left[\sum_{i=1}^N \sum_{k=1}^K t_i^k (\Sigma^{-1} (\mathbf{x}_i - \boldsymbol{\mu}_k) (\mathbf{x}_i - \boldsymbol{\mu}_k)^T \Sigma^{-1})^T \right] \\ & = +\frac{1}{2} \left[\sum_{i=1}^N \sum_{k=1}^K t_i^k \Sigma^{-1} (\mathbf{x}_i - \boldsymbol{\mu}_k) (\mathbf{x}_i - \boldsymbol{\mu}_k)^T \Sigma^{-1} \right] \end{aligned}$$

$$\textcircled{1} + \textcircled{2} = 0$$

$$-\frac{1}{2} \sum_{i=1}^N \sum_{k=1}^K t_i^k + \frac{1}{2} \left[\sum_{i=1}^N \sum_{k=1}^K t_i^k \sum_{i=1}^N \sum_{k=1}^K t_i^k (\mathbf{x}_i - \boldsymbol{\mu}_k) (\mathbf{x}_i - \boldsymbol{\mu}_k)^T \Sigma^{-1} \right] = 0$$

\Downarrow multiply Σ in 2 times

$$-\frac{1}{2} \sum_{i=1}^N \sum_{k=1}^K t_i^k + \frac{1}{2} \sum_{i=1}^N \sum_{k=1}^K t_i^k (\mathbf{x}_i - \boldsymbol{\mu}_k) (\mathbf{x}_i - \boldsymbol{\mu}_k)^T = 0$$

$$\Rightarrow \frac{1}{2} \sum_{i=1}^N \sum_{k=1}^K t_i^k (\mathbf{x}_i - \boldsymbol{\mu}_k) (\mathbf{x}_i - \boldsymbol{\mu}_k)^T = \frac{1}{2} \sum_{i=1}^N \sum_{k=1}^K t_i^k = \frac{1}{N} \sum_{i=1}^N \sum_{k=1}^K t_i^k (\mathbf{x}_i - \boldsymbol{\mu}_k) (\mathbf{x}_i - \boldsymbol{\mu}_k)^T$$

$$= \sum_{k=1}^K \frac{N_k}{N} \cdot \frac{1}{N_k} \sum_{i=1}^N t_i^k (\mathbf{x}_i - \boldsymbol{\mu}_k) (\mathbf{x}_i - \boldsymbol{\mu}_k)^T$$

$$= \sum_{k=1}^K \frac{N_k}{N} \sum_{i=1}^N t_i^k (\mathbf{x}_i - \boldsymbol{\mu}_k) (\mathbf{x}_i - \boldsymbol{\mu}_k)^T \quad \text{where}$$

$$\sum_k = \frac{1}{N_k} \sum_{i=1}^N t_i^k (\mathbf{x}_i - \boldsymbol{\mu}_k) (\mathbf{x}_i - \boldsymbol{\mu}_k)^T$$

Q.E.D.

MLHW

Q4. Given n points $x_i \in \mathbb{R}^d$, $i \in \{1, \dots, n\}$ / k cluster center $\mu = (\mu_1, \dots, \mu_k)$

$$\text{Objective } \arg \min L(C, \mu) = \sum_{i=1}^n \underbrace{\|x_i - \mu_{C(i)}\|_2^2} = \sum_{q=1}^k \sum_{i:C(i)=q} \|x_i - \mu_q\|_2^2$$

Get the distance between i -th data and the center
that i -th data belongs to the cluster q

$$(a) \quad \sum_{i=1}^m \|z_i - \bar{z}\|_2^2 \leq \sum_{i=1}^m \|z_i - z\|_2^2 \text{ where } \bar{z} = \frac{1}{m} \sum_{i=1}^m z_i, \text{ points } z_1, \dots, z_m, z \text{ is an arbitrary point in same space}$$

(Solved) Expand the expression

$$\begin{aligned} \sum_{i=1}^m \|z_i - \bar{z}\|_2^2 &= (z_1 - \bar{z})^T(z_1 - \bar{z}) + (z_2 - \bar{z})^T(z_2 - \bar{z}) + \dots + (z_m - \bar{z})^T(z_m - \bar{z}) \\ &= (z_1^T z_1 - 2\bar{z}^T z_1 + \bar{z}^T \bar{z}) + (z_2^T z_2 - 2\bar{z}^T z_2 + \bar{z}^T \bar{z}) + \dots + (z_m^T z_m - 2\bar{z}^T z_m + \bar{z}^T \bar{z}) \\ &= \sum_{i=1}^m \|z_i\|_2^2 - 2\bar{z}^T \sum_{i=1}^m z_i + m\|\bar{z}\|_2^2 \end{aligned}$$

$$\begin{aligned} \sum_{i=1}^m \|z_i - z\|_2^2 &= (z_1 - z)^T(z_1 - z) + (z_2 - z)^T(z_2 - z) + \dots + (z_m - z)^T(z_m - z) \\ &= \sum_{i=1}^m \|z_i\|_2^2 - 2z^T \sum_{i=1}^m z_i + m\|z\|_2^2 \end{aligned}$$

$$\begin{aligned} \sum_{i=1}^m \|z_i - z\|_2^2 - \sum_{i=1}^m \|z_i - \bar{z}\|_2^2 &= m\|z\|_2^2 - 2z^T \sum_{i=1}^m z_i - m\|\bar{z}\|_2^2 + 2\bar{z}^T \sum_{i=1}^m z_i \\ &= m\|z\|_2^2 - m\|\bar{z}\|_2^2 - \boxed{2(z^T - \bar{z}^T) \sum_{i=1}^m z_i} \rightarrow 2(z^T - \bar{z}^T) m \cdot \frac{1}{m} \sum_{i=1}^m z_i \\ &= 2(z^T - \bar{z}^T) m \cdot \bar{z} \\ &= m\|z\|_2^2 - m\|\bar{z}\|_2^2 - 2m z^T \bar{z} + 2m\|\bar{z}\|_2^2 \\ &= m\|z\|_2^2 + m\|\bar{z}\|_2^2 - 2m z^T \bar{z} = m(z^T z - 2z^T \bar{z} + \bar{z}^T \bar{z}) = m\|z - \bar{z}\|_2^2 \end{aligned}$$

$$\therefore m > 0 \text{ and } \|z - \bar{z}\|_2^2 \geq 0$$

$$\therefore \sum_{i=1}^m \|z_i - z\|_2^2 - \sum_{i=1}^m \|z_i - \bar{z}\|_2^2 \geq 0 \Rightarrow \boxed{\sum_{i=1}^m \|z_i - z\|_2^2 \geq \sum_{i=1}^m \|z_i - \bar{z}\|_2^2}$$

Q.E.D.

(b) Show that $L(C^{t+1}, \mu^t) < L(C^t, \mu^t)$, that is fix every class center and use the data belongs to new cluster to compute the distance with the center

Given the original loss function $L(C^t, \mu^t) = \sum_{i=1}^n \|x_i - \mu_{C(i)}\|_2^2$

(proof) Let C^{t+1} be the new assignment and C^t is the assignment from the previous iteration $\Rightarrow C^{t+1} \in \arg \min_{j=\{1 \dots k\}} \|x_i - \mu_j\|_2^2$
means choose the closest cluster center as x_i 's cluster

$$\begin{aligned} L(C^{t+1}, \mu^t) - L(C^t, \mu^t) &= \sum_{i=1}^n \|x_i - \mu_{C(i)}^{t+1}\|_2^2 - \sum_{i=1}^n \|x_i - \mu_{C(i)}^t\|_2^2 \\ &= \sum_{i=1}^n (\|x_i - \mu_{C(i)}^{t+1}\|_2^2 - \|x_i - \mu_{C(i)}^t\|_2^2) \end{aligned}$$

because $\mu_{C(i)}^{t+1}$ is the closest to x_i , so the distance

$\|x_i - \mu_{C(i)}^{t+1}\|_2^2$ is smaller than $\|x_i - \mu_{C(i)}^t\|_2^2$

$$\Rightarrow L(C^{t+1}, \mu^t) - L(C^t, \mu^t) < 0 \quad \text{Q.E.D.}$$

(c) Use the result from (a). Fix the cluster assignment C^{t+1} and compute the new cluster center μ^{t+1} from all the data in cluster.

$$\text{Given: } L(C^t, \mu^t) = \sum_{q=1}^k \sum_{i:C(i)=q} \|x_i - \mu_q\|_2^2$$

(proof) Consider j -th cluster, μ_j^t is previous cluster center and μ_j^{t+1} is new one.

$$\mu_j^{t+1} = \frac{1}{|\{i: C(i)=j\}|} \sum_{i:C(i)=j} x_i$$

↑
the # of data in cluster j

$$L(C^{t+1}, \mu^{t+1}) - L(C^{t+1}, \mu^t) = \sum_{q=1}^k \sum_{i:C(i)=q} \|x_i - \mu_q^{t+1}\|_2^2 - \sum_{q=1}^k \sum_{i:C(i)=q} \|x_i - \mu_q^t\|_2^2$$

$$\text{use the result from (a)} \sum_{i=1}^m \|z_i - z\|_2^2 - \sum_{i=1}^m \|z_i - \bar{z}\|_2^2 \geq 0$$

$$\text{Then } L(C^{t+1}, \mu^{t+1}) - L(C^{t+1}, \mu^t) < 0$$

Q.E.D.

M LHW Q4 cont.

(d). Use the result that $L(C^{t+1}, \mu^t) \leq L(C^t, \mu^t)$ and $L(C^{t+1}, \mu^{t+1}) \leq L(C^{t+1}, \mu^t)$
so $L(C^{t+1}, \mu^{t+1}) \leq L(C^t, \mu^t)$

(i) use completeness axiom

$$\therefore L(C^t, \mu^t) \geq L(C^{t+1}, \mu^{t+1}) \text{ and } L(C^t, \mu^t) = \sum_{i=1}^n \|x_i - \mu_{C(i)}\|_2^2 \geq 0$$

$$\therefore \exists k \text{ so that } L(C^t, \mu^t) \geq k \quad \forall t$$

implies $\lim_{t \rightarrow \infty} L(C^t, \mu^t)$ exist (in real number)

(ii) use approximation theorem

Given $L(C^t, \mu^t)$ has a finite infimum, If $\varepsilon > 0$ then there exist $a \in L(C^t, \mu^t)$
such that $\inf L \leq a < \inf L + \varepsilon$

(iii) use monotone convergence theorem

because of the theorem and axiom mentioned above,

if $L(C^t, \mu^t)$ is decreasing and bounded below then $\{L\}$ converges to a finite limit

Q.E.D.

(e) The algorithm will stop when no change in loss function occurs during the assignment step. That is, the result of loss function reach the lower bound and won't update the cluster center and loss is a fixable value.

$L(C^t, \mu^t) - L(C^{t+1}, \mu^{t+1}) = \varepsilon_t > 0 \quad \forall t$ and the lower bound is 0 and the initial loss is finite value. that greater than 0

\Rightarrow Repeat: $L(C^t, \mu^t) - \varepsilon_t$ for n times \rightarrow implies n is a finite step.

Q.E.D.