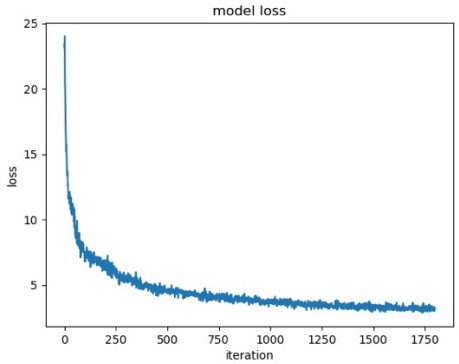
學號：R08942085 系級： 電信所碩一 姓名：陳芃彣

1. (1%) 請使用不同的 Autoencoder model, 以及不同的降維方式(降到不同維度), 討論其 reconstruction loss & public / private accuracy。 (因此模型需要兩種, 降維方法也需要兩種, 但 clustrering 不用兩種。)

第一種 AE:

```
self.encoder = nn.Sequential(
    nn.Conv2d(3, 32, kernel_size = (3, 3), stride = (2, 2), padding = (1, 1)),
    nn.Conv2d(32, 64, kernel_size = (3, 3), stride = (2, 2), padding = (1, 1)),
    nn.Conv2d(64, 128, kernel_size = (3, 3), stride = (2, 2), padding = (1, 1)),
    nn.Conv2d(128, 256, kernel_size = (3, 3), stride = (2, 2), padding = (1, 1)),
)
#define decoder
self.decoder = nn.Sequential(
    nn.ConvTranspose2d(256, 128, 2, 2),
    nn.ConvTranspose2d(128, 64, 2, 2),
    nn.ConvTranspose2d(64, 32, 2, 2),
    nn.ConvTranspose2d(32, 3, 2, 2),
    nn.Tanh(),
)
```



PCA(n_components=64)+TSNE

Private Score	Public Score
0.72857	0.72296

PCA(n_components=128)

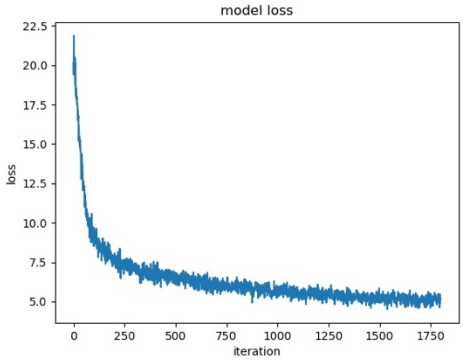
Private Score	Public Score
0.78095	0.79592

第二種 AE:

```
class Autoencoder(nn.Module):
    def __init__(self):
        super(Autoencoder, self).__init__()

        #define encoder
        self.encoder = nn.Sequential(
            nn.Conv2d(3, 8, kernel_size = (3, 3), stride = (1, 1), padding = (1, 1)),
            nn.MaxPool2d(2, 2),
            nn.Conv2d(8, 32, kernel_size = (3, 3), stride = (1, 1), padding = (1, 1)),
            nn.MaxPool2d(2, 2),
            nn.Conv2d(32, 128, kernel_size = (3, 3), stride = (1, 1), padding = (1, 1)),
            nn.MaxPool2d(2, 2),
            nn.Conv2d(128, 256, kernel_size = (3, 3), stride = (1, 1), padding = (1, 1)),
            nn.MaxPool2d(2, 2),
        )
        #define decoder
        self.decoder = nn.Sequential(
            nn.ConvTranspose2d(256, 128, 2, 2),
            nn.ConvTranspose2d(128, 32, 2, 2),
            nn.ConvTranspose2d(32, 8, 2, 2),
            nn.ConvTranspose2d(8, 3, 2, 2),
            nn.Tanh(),
        )
    def forward(self, x):

        encoded = self.encoder(x)
        decoded = self.decoder(encoded)
        #Total AE: return latent & reconstruct
        return encoded, decoded
```



PCA(n_components=64)+TSNE

Private Score	Public Score
0.80158	0.80000

PCA(n_components=128)

Private Score	Public Score
0.76333	0.76740

2. (1%) 從 dataset 選出 2 張圖，並貼上原圖以及經過 autoencoder 後 reconstruct 的圖片。

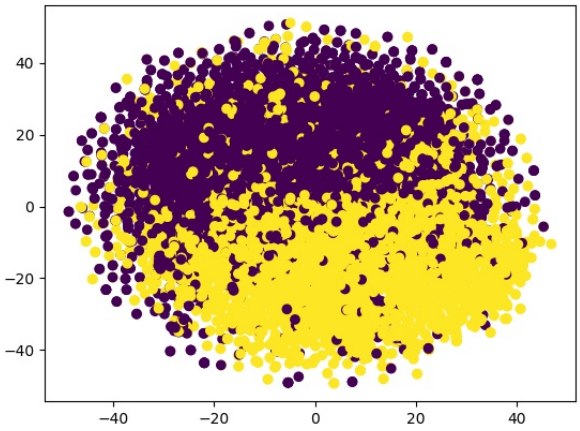
original_img1: 

reconstruct_img1: 

original_img2: 

reconstruct_img2: 

3. (1%) 在之後我們會給你 dataset 的 label。請在二維平面上視覺化 label 的分佈。



4. (3%)Refer to math problem

陳芃文 R08942085

1. (a) mean = [5.4, 8, 4.87]

$$cov = \begin{bmatrix} 120.4 & 5 & 32.8 \\ 5 & 122. & 29. \\ 32.8 & 29. & 81.6 \end{bmatrix}$$

singular value [152.97, 116.31, 54.72]

eigen vector [-0.617, -0.589, -0.523]
[0.678, -0.734, 0.027]
[-0.399, -0.337, 0.852]

(b) [1.186, 1.373, 2.251]
[0.758, -0.943, 0.730]
[-3.070, -4.450, 3.188]
[2.608, -2.978, 1.929]
[-1.822, -4.754, -4.251]
[3.354, 3.918, -2.527]
[-4.414, 2.556, 2.139]
[3.465, -1.731, -2.218]
[-2.313, 6.033, -0.203]
[-5.752, 0.976, -0.917]

(c) avg loss = 54.7203

2. (a) $(ab)^T = b^T a^T$

$(A^TA)^T = A^T(A^T)^T = A^TA$
$(AA^T)^T = (A^T)^TA^T = AA^T$ $\Rightarrow \therefore A^TA$ & AA^T are symmetric

$x^TAA^Tx = \underbrace{(A^Tx)^TA^Tx}_{\text{2 norm of } A^Tx} \geq 0 \Rightarrow \therefore AA^T$ is positive definite

$x^TA^TAx = \underbrace{(Ax)^TAx}_{\text{2 norm of } A^Tx} \geq 0 \Rightarrow \therefore A^TA$ is positive definite

$\because A^TA$ and AA^T are symmetric $\therefore$ they have the same non-zero eigenvalues.

(b) $\Lambda = \begin{bmatrix} \lambda_1 & & 0 \\ & \lambda_2 & \\ 0 & & \ddots \\ & & & \lambda_n \end{bmatrix}$

$\Lambda^{\frac{1}{2}}\Lambda^{\frac{1}{2}} = \Lambda$, $\Lambda^{\frac{1}{2}} = \begin{bmatrix} \sqrt{\lambda_1} & & 0 \\ & \sqrt{\lambda_2} & \\ 0 & & \ddots \\ & & & \sqrt{\lambda_n} \end{bmatrix}$

$\because \begin{cases} \Lambda^{\frac{1}{2}} = \Lambda^{\frac{T}{2}} \\ \Lambda = \Lambda^{\frac{1}{2}}\Lambda^{\frac{1}{2}} \end{cases}$

$\because \Sigma$ is symmetric, $U = [u_1 \cdots u_M]$ is orthogonal matrix of eigenvectors of Σ
Λ = diagonal $(\lambda_1 \cdots \lambda_m)$ Σ is semi positive definite,
$\therefore \lambda_1 \cdots \lambda_m \geq 0$

$\Sigma = U\Lambda U^T$
$= U\Lambda^{\frac{1}{2}}\Lambda^{\frac{1}{2}}U^T$
$= U\Lambda^{\frac{1}{2}}\Lambda^{\frac{1}{2}}U^T$
$= U\Lambda^{\frac{1}{2}}(U\Lambda^{\frac{1}{2}})^T$
$= \frac{1}{n}\sum_{i=1}^{n}(x_i - \mu)(x_i - \mu)^T$ #

2.(c) $\min \ \text{Trace}(\Phi^T \Sigma \Phi)$

s.t. $\Phi^T \Phi = I_k$

variables $\Phi \in \mathbb{R}^{m \times k}$

$\text{trace}(\Phi^T \Sigma \Phi)$ 見 by (b)

$= \frac{1}{N} \text{Trace}(\Phi^T X X^T \Phi)$

$= \frac{1}{N} \| \Phi^T X \|_F^2$

$= \frac{1}{N} \sum_{i=1}^{N} \| \Phi^T x_i \|^2$

$= \frac{1}{N} \sum_{i=1}^{N} \| \hat{x}_i^{(s)} \|^2$

↓

投影在 ϕ_1, ϕ_2 上

minimize

找到最小 k 個 eigenvalue 的 eigenvector

3. let $y_k = \begin{cases} 1 & \text{if } \hat{y}_i = k \\ \frac{-1}{K-1} & \text{if } \hat{y}_i \neq k \end{cases}$ $y_{\hat{y}_i} = [y_1 \cdots y_k]$, $g_t = [g_t^1 \cdots g_t^K] \Rightarrow L(g_t) = \sum_{i=1}^{n} \exp(-y_{\hat{y}_i}^T g_t)$

1. Initialize $g_0(x) = 0$
2. For $t=1$ to T:

(a) Minimize $\sum_{i=1}^{n} \exp\left(y_{\hat{y}_i}^T (g_t(x_i) + \alpha f(x_i))\right)$

$= \sum_{i=1}^{n} w_i \exp(\alpha y_{\hat{y}_i}^T f(x_i))$, where $w_i = \exp(y_{\hat{y}_i}^T g_t(x_i))$

Notice that there is a one to one corresponding $T(x)$ for $f(x)$ in the following way: $T(x)=k$, if $f_k(x)=1$

$T_t(x) = \text{argmin} \sum_{i=1}^{n} w_i \mathbb{I}(\hat{y}_i \neq T(x_i))$

<pf>

$\sum_{i=1}^{n} w_i \exp(\alpha y_{\hat{y}_i}^T f(x_i))$

$= \sum_{\hat{y}_i = T(x_i)} w_i \exp\left(\frac{-\alpha}{K-1}\right) + \sum_{\hat{y}_i \neq T(x_i)} w_i \exp\left(\frac{\alpha}{(K-1)^2}\right)$

$= \exp\left(\frac{-\alpha}{K-1}\right) \sum_i w_i + \left(\exp\left(\frac{\alpha}{(K-1)^2}\right) - \exp\left(\frac{-\alpha}{K-1}\right)\right) \sum_i w_i \mathbb{I}(\hat{y}_i \neq T(x_i))$

Since only the last sum depends on $T(x)$,

$T_t(x) = \text{argmin} \sum_{i=1}^{n} w_i \mathbb{I}(\hat{y}_i \neq T(x_i))$

plug the result into the formula

$\alpha_t = (K-1)^2 \left(\log \frac{1-err_t}{err_t} + \log(K-1)\right)$

where $err_t = \sum_{i=1}^{n} w_i \mathbb{I}(\hat{y}_i \neq T(x_i))$

(b) Update $g_t(x) = g_{t-1}(x) + \alpha_t f_t(x)$

下一頁是推失敗的

3. $L(g_T^1, \cdots, g_T^K) = \sum_{i=1}^{n} \exp\left[\frac{1}{K-1}\sum_{k \neq \hat{y}_i} g_T^k(x_i) - g_T^{\hat{y}_i}(x_i)\right]$

class 1 / not class 1 ; class K / not class K

Assume b data belong to class k

$$= \sum_{i=1}^{a} \exp\left[\frac{1}{K-1}\sum_{k \neq \hat{y}_i} g_{T-1}^k(x_i) + \frac{1}{K-1}\cdot \alpha_T^k f_T(x) - g_{T-1}^{\hat{y}_i}(x_i)\right] + \sum_{i=1}^{b} \exp\left[\frac{1}{K-1}\sum_{k \neq \hat{y}_i} g_{T-1}^k(x_i) - g_{T-1}^{\hat{y}_i}(x_i) - \alpha_T^k f_T(x_i)\right], \quad a+b=n$$

Assume a data not belongs to class k

$$L(g_T^k) - L(g_{T-1}^k) = \sum_{i=1}^{a} \exp\left[\frac{1}{K-1}\sum_{k \neq \hat{y}_i} g_{T-1}^k(x_i) - g_{T-1}^{\hat{y}_i}(x_i)\right] \exp\left[\frac{\alpha_T^k f_T(x)}{K-1} - 1\right] + \sum_{i=1}^{b} \exp\left[\frac{1}{K-1}\sum_{k \neq \hat{y}_i} g_T^k(x_i) - g_T^{\hat{y}_i}(x_i)\right] \exp\left[-\alpha_T^k f_T(x_i) - 1\right]$$

If α_T^k is very small

we can use Talyr expansion for approximation

$$L(g_T^k) - L(g_{T-1}^k) \approx \left[\frac{f_T(x)}{K-1}\sum_{i=1}^{a} \exp\left[\frac{1}{K-1}\sum_{k \neq \hat{y}_i} g_{T-1}^k(x_i) - g_{T-1}^{\hat{y}_i}(x_i)\right] - f_T(x)\sum_{i=1}^{b} \exp\left[\frac{1}{K-1}\sum_{k \neq \hat{y}_i} g_T^k(x_i) - g_T^{\hat{y}_i}(x_i)\right]\right]\alpha_T^k$$

$$\frac{\partial}{\partial \alpha_t^k} = \frac{f_T(x)}{K-1}\sum_{i=1}^{a} \exp\left[\frac{1}{K-1}\sum_{k \neq \hat{y}_i} g_{T-1}^k(x_i) - g_{T-1}^{\hat{y}_i}(x_i)\right]\exp\left[\frac{\alpha_T^k f_T(x)}{K-1} - 1\right] - f_T(x)\sum_{i=1}^{b} \exp\left[\frac{1}{K-1}\sum_{k \neq \hat{y}_i} g_T^k(x_i) - g_T^{\hat{y}_i}(x_i)\right]\exp\left[-\alpha_T^k f_T(x_i) - 1\right] = 0$$ 可以求出 α_T^k