HW4 Report

學號：B05902017 系級： 資工四 姓名：王盛立

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）

	Reconstruction loss	public accuracy	private accuracy
ModelA + PCA(n_components = 32)	0.1065	0.7812	0.7764
ModelA + TSNE	0.1065	0.8088	0.8021
ModelB + PCA(n_components = 32)	0.0963	0.7923	0.7886
ModelB + TSNE	0.0963	0.8121	0.8077

```
class ModelA(nn.Module):
    def __init__(self):
        super(Autoencoder, self).__init__()
        # define: encoder
        self.encoder = nn.Sequential(
            nn.Conv2d(3,8,3,2,2),
            nn.SELU(0.3),
            nn.Conv2d(8,16,3,2,2),
            nn.SELU(0.3),
            nn.Conv2d(16,32,3,2,2),
            nn.SELU(0.3),
            nn.Conv2d(32,64,3,2,2),
            nn.SELU(0.3),
        )

        self.encoderLinear = nn.Sequential(
            nn.Linear(576, 256),
            nn.SELU(0.3),
        )
```

```
        self.decoderLinear = nn.Sequential(
            nn.Linear(256,576),
            nn.SELU(0.3),
        )

        # define: decoder
        self.decoder = nn.Sequential(
            nn.ConvTranspose2d(64,32,2,2),
            nn.SELU(0.3),
            nn.ConvTranspose2d(32,16,2,2),
            nn.SELU(0.3),
            nn.ConvTranspose2d(16,8,2,2),
            nn.SELU(0.3),
            nn.ConvTranspose2d(8,3,2,2),
            nn.SELU(0.3),
            nn.Tanh(),
        )

class ModelB(nn.Module):
    def __init__(self):
        super(ModelTwo, self).__init__()

        # define: encoder
        self.encoder = nn.Sequential(
            nn.Conv2d(3,8,3,1,1),
            nn.SELU(0.2),
            nn.MaxPool2d(2,2),
            nn.Conv2d(8,16,3,1,1),
            nn.SELU(0.2),
            nn.MaxPool2d(2,2),
            nn.Conv2d(16,32,3,1,1),
            nn.SELU(0.2),
            nn.MaxPool2d(2,2),
            nn.Conv2d(32,64,3,1,1),
            nn.SELU(0.2),
            nn.MaxPool2d(2,2),
        )

        self.encoderLi = nn.Sequential(
            nn.Linear(256, 128),
            nn.SELU(0.2),
        )

        self.decoderLi = nn.Sequential(
            nn.Linear(128,256),
            nn.SELU(0.2),
        )
```

```
        # define: decoder
        self.decoder = nn.Sequential(
            nn.ConvTranspose2d(64,32,2,2),
            nn.SELU(0.2),
            nn.ConvTranspose2d(32,16,2,2),
            nn.SELU(0.2),
            nn.ConvTranspose2d(16,8,2,2),
            nn.SELU(0.2),
            nn.ConvTranspose2d(8,3,2,2),
            nn.SELU(0.2),
            nn.Tanh(),
        )
```

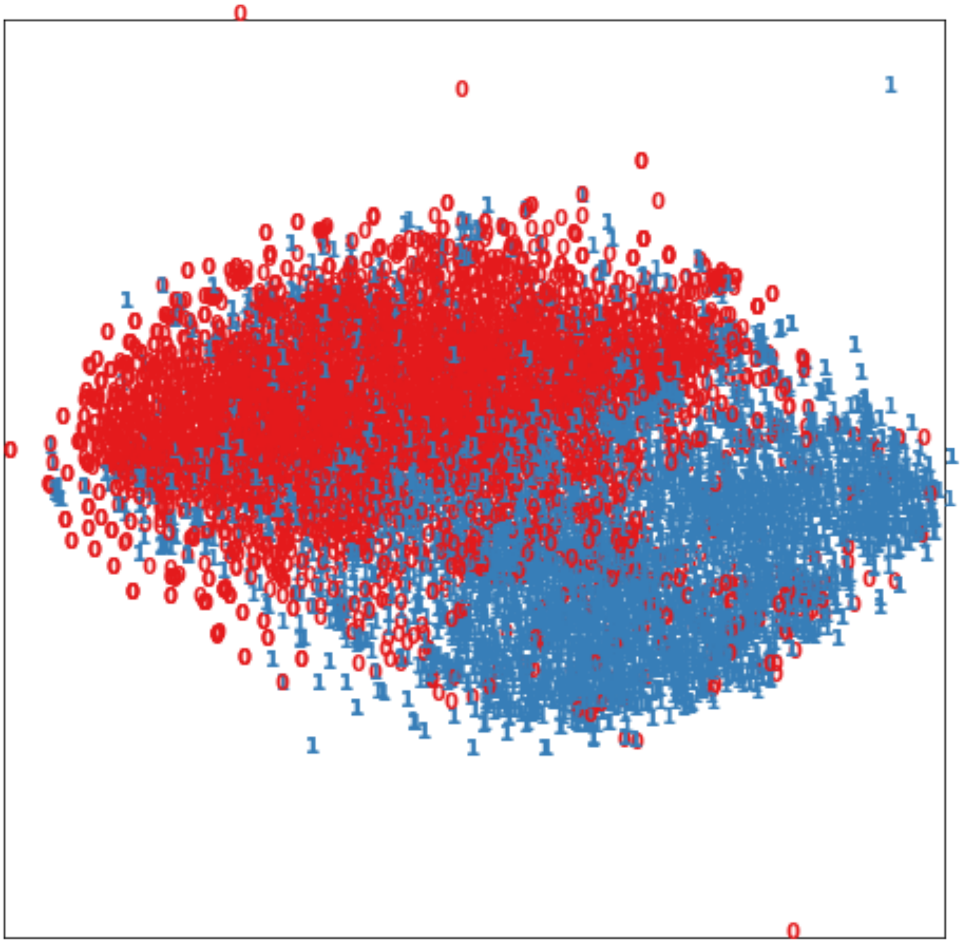
我兩個model疊的方式其實差不多，只是一個有用maxpooling強制他變小，另一個則是在convolution的過程中讓它自然變小。理論上用maxpooling會不太好，因為就強制讓他減少資訊，但是結果出來兩者其實差不多甚至maxpooling還要好一些些，可能是我兩個的結構沒有差太多的緣故。另外就是tsne比起pca要好一些，pca要設一些參數不然結果會更糟，不過tsne要做很久。

2 從dataset選出2張圖，並貼上原圖以及經過autoencoder後reconstruct的圖片

以下的四張照片上面都是原圖，下面則是經過autoencoder後reconstruct的圖片。

3 二維平面上視覺化label的分佈

這是我做tsne後，看到真正的label的分佈。從我最後的正確率可以看到有些地方是重疊的，也就是錯誤的部分。



4 Refer to math problem

1 Principle Component Analysis

(a) What are the principal axes

$$\mu = \frac{1}{10}\Sigma_{n=1}^{10} x_n = \begin{bmatrix} 5.4 \\ 8 \\ 4.8 \end{bmatrix}$$

$$\Sigma = \frac{1}{10}\Sigma_{n=1}^{10}(x_i - \mu)(x_i - \mu)^T = \begin{bmatrix} 12.04 & 0.5 & 3.28 \\ 0.5 & 12.2 & 2.9 \\ 3.28 & 2.9 & 8.16 \end{bmatrix}$$

$$\Sigma = Q\Lambda Q^T$$

$$Q = \begin{bmatrix} 0.616 & 0.678 & -0.399 \\ 0.588 & -0.734 & -0.337 \\ 0.522 & 0.027 & 0.852 \end{bmatrix}$$

$$\Lambda = \begin{bmatrix} 15.29 & 0 & 0 \\ 0 & 11.63 & 0 \\ 0 & 0 & 5.47 \end{bmatrix}$$

Σ的eigenvector為 $\begin{bmatrix} 0.616 \\ 0.588 \\ 0.522 \end{bmatrix}$ 、 $\begin{bmatrix} 0.678 \\ -0.734 \\ 0.027 \end{bmatrix}$ 、 $\begin{bmatrix} -0.399 \\ -0.337 \\ 0.852 \end{bmatrix}$ ，就是principal axis

(b) **compute the principal components for each sample**

讓w為 $\begin{bmatrix} 0.616 & 0.588 & 0.522 \\ 0.678 & -0.734 & 0.0272 \\ -0.399 & -0.337 & 0.852 \end{bmatrix}$, $x_1, x_2, \cdots, x_{10}$的$principal\ component$為

$$wx_1 = \begin{bmatrix} 3.36 \\ -0.708 \\ 1.481 \end{bmatrix}$$

$$wx_2 = \begin{bmatrix} 9.784 \\ -3.025 \\ -0.039 \end{bmatrix}$$

$$wx_3 = \begin{bmatrix} 13.61 \\ -6.53 \\ 2.418 \end{bmatrix}$$

$$wx_4 = \begin{bmatrix} 7.934 \\ -5.06 \\ 1.16 \end{bmatrix}$$

$$wx_5 = \begin{bmatrix} 12.363 \\ -6.835 \\ -5.021 \end{bmatrix}$$

$$wx_6 = \begin{bmatrix} 7.191 \\ 1.836 \\ -3.297 \end{bmatrix}$$

$$wx_7 = \begin{bmatrix} 14.957 \\ -0.474 \\ 1.369 \end{bmatrix}$$

$$wx_8 = \begin{bmatrix} 7.077 \\ -3.813 \\ -3.048 \end{bmatrix}$$

$$wx_9 = \begin{bmatrix} 12.858 \\ 3.951 \\ -0.973 \end{bmatrix}$$

$$wx_{10} = \begin{bmatrix} 16.293 \\ -1.105 \\ -1.747 \end{bmatrix}$$

(c) **average reconstruction error if reduce dimension to 2D**

$$w = \begin{bmatrix} 0.616 & 0.588 & 0.522 \\ 0.678 & -0.734 & 0.0272 \end{bmatrix}$$

average reconstruction error = $\frac{1}{10}\Sigma_{n=1}^{10}||x_n - w^T(wx_n)||^2 = 6.068$

2

(a)

$(AA^T)^T = (A^T)^T A^T = AA^T$

$(A^T A)^T = A^T (A^T)^T = A^T A$

$(A^T A)$和(AA^T)皆為symmetric

$x^T(AA^T)x = (x^T A)(A^T x) = (A^T x)^T (A^T x) = ||A^T x||^2 \geq 0$

$y^T(A^T A)y = (y^T A^T)(Ay) = (Ay)^T (Ay) = ||Ay||^2 \geq 0$

$(A^T A)$和(AA^T)皆為positive semi-definite

讓$\lambda > 0$為AA^T的一個eigenvalue，取v為一個eigenvector

則可得$(AA^T)v = \lambda v$，因此可得$(A^T A)(A^T v) = A^T((AA^T)v) = A^T(\lambda v) = \lambda(A^T v)$

所以得知$\lambda > 0$也是$A^T A$的一個eigenvalue，$A^T v$為一個eigenvector

同理讓$\lambda > 0$為$A^T A$的一個eigenvalue，取v為一個eigenvector

則可得$(A^TA)v=\lambda v$，因此可得$(AA^T)(Av)=A((A^TA)v)=A(\lambda v)=\lambda(Av)$

所以得知$\lambda>0$也是AA^T的一個eigenvalue，A^Tv為一個eigenvector

由上述可得AA^T和A^TA有相同的non-zero eigenvalue

(b)

讓$z_1=\begin{bmatrix}\sqrt{m}\\0\\\vdots\\0\end{bmatrix}$

$z_2=\begin{bmatrix}-\sqrt{m}\\0\\\vdots\\0\end{bmatrix}$

$z_3=\begin{bmatrix}0\\\sqrt{m}\\\vdots\\0\end{bmatrix}$

$z_4=\begin{bmatrix}0\\-\sqrt{m}\\\vdots\\0\end{bmatrix}$

$z_{2m-1}=\begin{bmatrix}0\\0\\\vdots\\\sqrt{m}\end{bmatrix}$

$z_{2m}=\begin{bmatrix}0\\0\\\vdots\\-\sqrt{m}\end{bmatrix}$

可得$z_1z_2\cdots z_{2m}$的mean為$\frac{1}{2m}\sum_{k=1}^{2m}z_k=0$

Covariance matrix為$\frac{1}{2m}\sum_{k=1}^{2m}(z_k-0)(z_k-0)^T=I_m$

因為Σ為positive semi-definite

所以$\exists A$使$\Sigma=AA^T$

取$x_k=Az_k+\mu$，可得$x_1x_2\cdots x_{2m}$的mean為
$\frac{1}{2m}\sum_{k=1}^{2m}x_k=\frac{1}{2m}\sum_{k=1}^{2m}Az_k+\mu=A(\frac{1}{2m}\sum_{k=1}^{2m}z_k)+\mu=A*0+\mu=\mu$

Covariance matrix為
$\frac{1}{2m}\sum_{k=1}^{2m}(x_k-\mu)(x_k-\mu)^T = \frac{1}{2m}\sum_{k=1}^{2m}((Az_k+\mu)-\mu)((Az_k+\mu)-\mu)^T$
$= \frac{1}{2m}\sum_{k=1}^{2m}(Az_k)(Az_k)^T = \frac{1}{2m}\sum_{k=1}^{2m}(Az_kz_k^TA^T) = A(\frac{1}{2m}\sum_{k=1}^{2m}z_kz_k^T)A^T = AI_mA^T = AA^T = \Sigma$

(c)

3