**Machine Learning Fall 2020 ——— Homework 5**

學號：B07902037 系級： 資工三 姓名：蔡沛勳

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。）**

我用 CNN 和 DNN實做了不同的Autoencoder，具體結構如下：

CNN:

|  |
| --- |
| class AE(nn.Module):  def \_\_init\_\_(self):  super(AE, self).\_\_init\_\_()    self.encoder = nn.Sequential(  nn.Conv2d(3, 64, 3, stride=1, padding=1),  nn.ReLU(True),  nn.MaxPool2d(2),  nn.Tanh(),  nn.Conv2d(64, 128, 3, stride=1, padding=1),  nn.ReLU(True),  nn.MaxPool2d(2),  nn.Tanh(),  nn.Conv2d(128, 256, 3, stride=1, padding=1),  nn.ReLU(True),  nn.MaxPool2d(2),  nn.Tanh(),    nn.Conv2d(256, 256, 3, stride=1, padding=1),  nn.ReLU(True),  nn.Tanh(),  #nn.MaxPool2d(2)  nn.Conv2d(256, 256, 3, stride=1, padding=1),  nn.ReLU(True),  )    self.decoder = nn.Sequential(  nn.ConvTranspose2d(256, 128, 5, stride=1),  nn.ReLU(True),  nn.Tanh(),  nn.ConvTranspose2d(128, 64, 9, stride=1),  nn.ReLU(True),  nn.Tanh(),  nn.ConvTranspose2d(64, 3, 17, stride=1),  nn.Tanh()  )  def forward(self, x):  x1 = self.encoder(x)  x = self.decoder(x1)  return x1, x |

DNN:

|  |
| --- |
| class AE2(nn.Module):  def \_\_init\_\_(self):  super(AE2, self).\_\_init\_\_()    self.encoder = nn.Sequential(  nn.Linear(3 \* 32 \* 32, 1024),  nn.LeakyReLU(0.5),  nn.Tanh(),  nn.Linear(1024, 1024),  nn.LeakyReLU(0.5),  nn.Tanh(),  nn.Linear(1024, 1024),  nn.LeakyReLU(0.5),  )    self.decoder = nn.Sequential(  nn.Linear(1024, 1024),  nn.LeakyReLU(0.5),  nn.Tanh(),  nn.Linear(1024, 1024),  nn.LeakyReLU(0.5),  nn.Tanh(),  nn.Linear(1024, 3 \* 32 \* 32),  nn.LeakyReLU(0.5),  nn.Tanh(),  )  def forward(self, x):  x = torch.reshape(x, (-1, 3 \* 32 \* 32))  x1 = self.encoder(x)  x = self.decoder(x1)  x = torch.reshape(x, (-1, 3, 32, 32))  return x1, x |

兩 Autoencoder皆以Adam在learning rate = 0.00001做100個epoch。之後CNN得到結果為4096維的資料，DNN結果為1024維的資料。兩者皆以PCA降到128維後再用t-SNE降到2維，最終以K-Means做分類。得到結果如下：

|  |  |  |
| --- | --- | --- |
|  | reconstruction loss | accuracy |
| CNN model | 7.41368 | 0.79589 |
| DNN model | 6.72101 | 0.81089 |

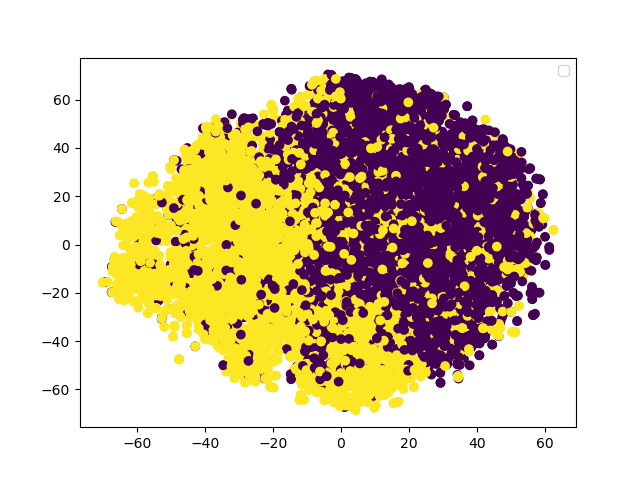
1. **(1%) 從dataset選出2張圖，並貼上原圖以及經過autoencoder後reconstruct的圖片。**

以下兩張圖片為DNN實作的Autoencoder原本即reconstruct後的結果比較，可以看出畫片中的內容再reconstruct後模糊不少。

****

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

以下為DNN實作的Autoencoder再經過PCA及t-SNE降至二維後的分布情形。

****

1. **(3%)Refer to math problem**

**1.**

t = 1:

t = 2:

t = 3:

t = 4:

t = 5:

t = 6:

t = 7:

t = 8:

**2.**

從題目得到

設

可推得

可轉換成

對微分得

對微分得