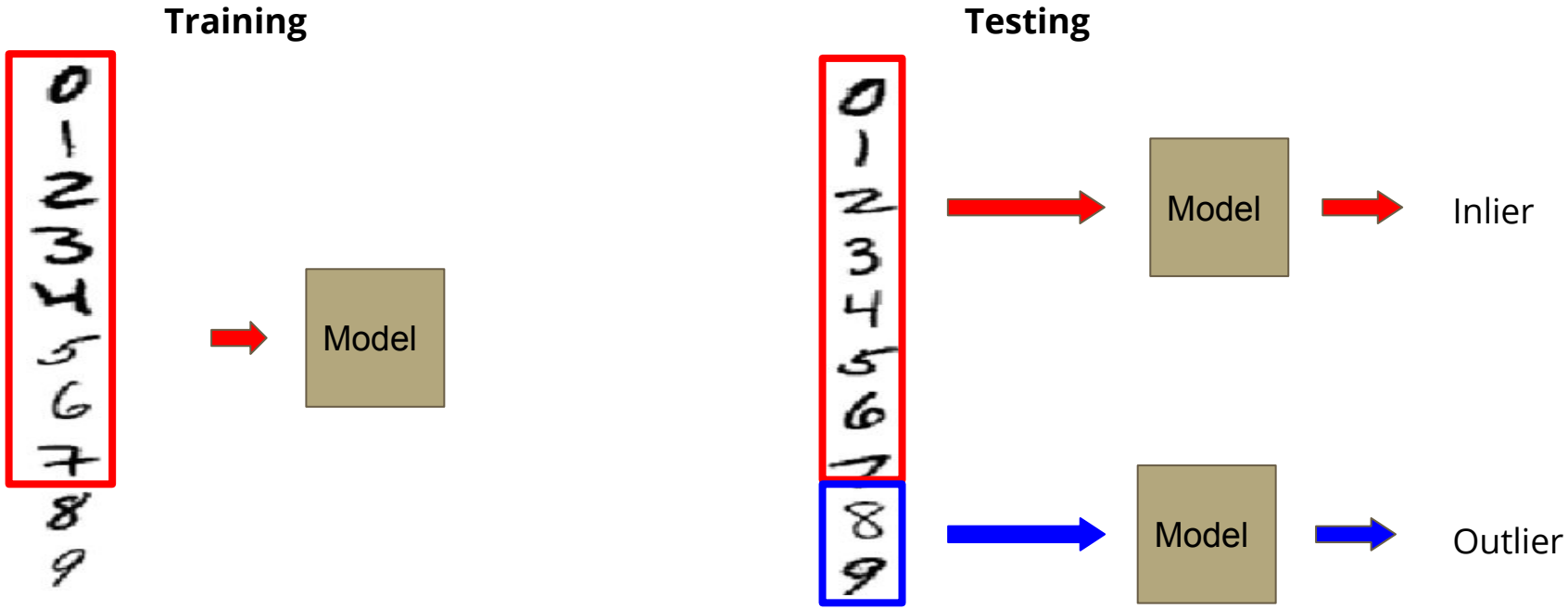
Homework 10
Anomaly Detection

ML TAs
ntu-ml-2020spring-ta@googlegroups.com

Goal

- Semi-supervised anomaly detection: 在只給定乾淨的(無anomaly)training data的情況下，分辨 testing data 中哪些 data 是來自 training 或是從未見過的類別

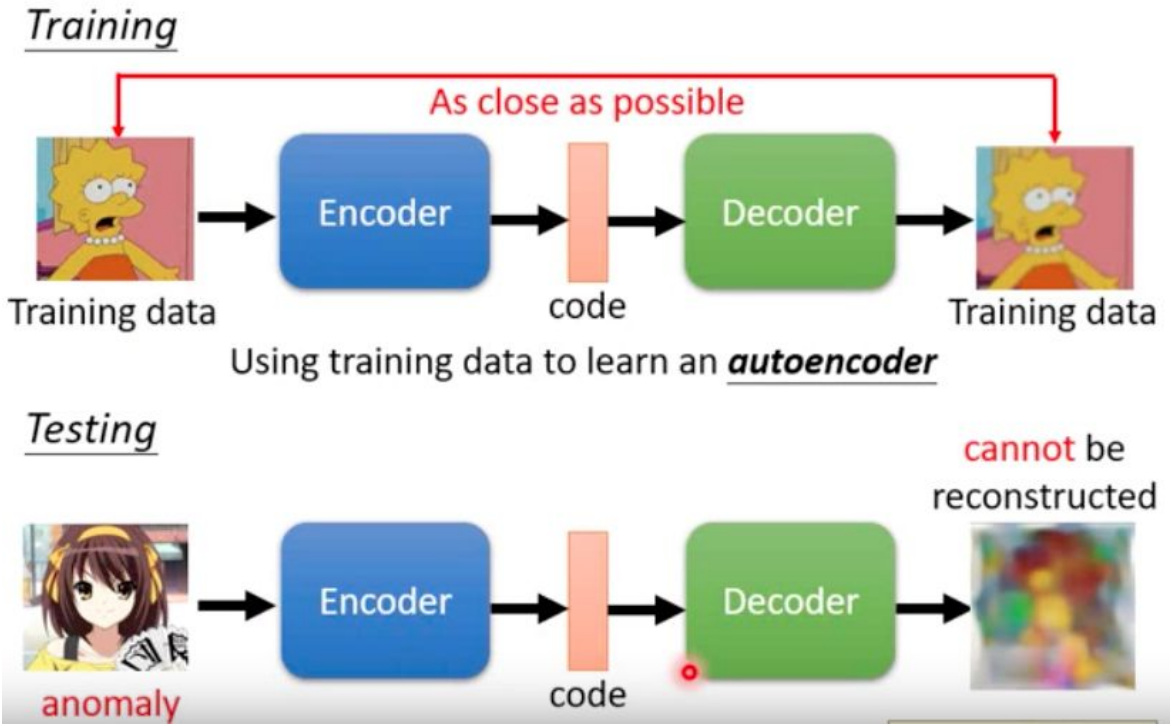


Data

- Training: 某個 image dataset 的 training data (大小32*32*3) 中的屬於某些label的data（40000 筆）
- Testing: 此 dataset 的所有 testing data（10000 筆）
- Notice: 請勿使用額外data進行training，亦不可使用pretrained model。可用額外data輔助validation。禁止搜尋或手標給定的data。

Method 1- Autoencoder

可以用MSE計算原圖跟復原的圖差多少



Method 2- KNN

- 假設 training data 的 label 種類不多
- 假設 training data 有 n 群
- 用 K-means 計算 training data 中的 n 個 centroid，再用這 n 個 centroid 對 training data 分群
- Inlier data 與所分到群的 centroid 的距離應較 outlier 的此距離來得小

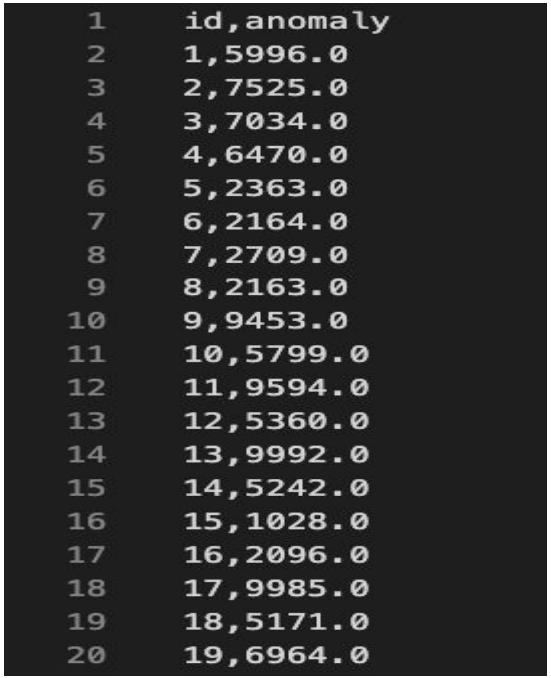
Method 3- PCA

- 計算 training data 的 principal component
- 將 testing data 投影在這些 component 上
- 再將這些投影重建回原先 space 的向量
- 對重建的圖片和原圖計算平方差，inlier data 的數值應該較 outlier 的數值為小

Kaggle

Metric: ROC_AUC score

Sample output:



```
id,anomaly
1,5996.0
2,7525.0
3,7034.0
4,6470.0
5,2363.0
6,2164.0
7,2709.0
8,2163.0
9,9453.0
10,5799.0
11,9594.0
12,5360.0
13,9992.0
14,5242.0
15,1028.0
16,2096.0
17,9985.0
18,5171.0
19,6964.0
```

Report

1. (2%) 任取一個baseline model (sample code裡定義的 fcn, cnn, vae) 與你在kaggle leaderboard上表現最好的model（如果表現最好的model就是sample code裡定義的model的話就任選兩個，e.g. fcn & cnn），對各自重建的testing data的image中選出與原圖mse最大的兩張加上最小的兩張並畫出來。（假設有五張圖，每張圖經由 autoencoder A重建的圖片與原圖的 MSE分別為 [25.4, 33.6, 15, 39, 54.8]，則MSE最大的兩張是圖4、5而最小的是圖1、3）。須同時附上原圖與經 autoencoder重建的圖片。（圖片總數：(原圖+重建)*(兩顆model)*(mse最大兩張+mse最小兩張) = 16張）

2. (1%) 嘗試把 sample code中的KNN 與 PCA 做在 autoencoder 的 encoder output 上，並回報兩者的 auc score。

Report

四張圖 1.PCA投影 沒encoder
2.PCA 投影 encoder1降維
3.PCA投影 沒encoder降維
4.PCA 投影 encoder2降維

3. (1%) 如hw9，使用PCA或T-sne將testing data投影在2維平面上，並將testing data經第1題的兩顆model的encoder降維後的output投影在2維平面上，觀察經encoder降維後是否分成兩群的情況更明顯。(因未給定testing label，所以點不須著色)(總共 **4張圖**)

4. (2%) 說明為何使用auc score來衡量而非binary classification常用的f1 score。如果使用f1 score會有什麼不便之處？

Submission Format

GitHub 上的 hw10-<account> 裡面必須有以下檔案:

- report.pdf
- *.py（所有train/test model會用到的.py檔）
- Training時只能使用autoencoder model，testing時允許使用如report第2題提到的方法來提昇performance
- 作答report第一題所需的兩顆model所需的.pth檔
  - models/best.pth (kaggle learderboard上表現最好的model)
  - models/baseline.pth (另一顆baseline model)

Submission Format

- hw10_test.sh
  - 用同學繳交上來的 兩顆model執行testing,
  - **用法: bash hw10_test.sh <test.npy> <model> <prediction.csv>**
    - **<test.npy>**: 助教這邊存放 **test.npy** 的路徑，請同學不要寫死
    - **<model>**: 同學上繳的autoencoder model的路徑，請同學不要寫死。model的名稱一律照上頁的方式命名（best.pth, baseline.pth）。同學的程式須從名稱判斷是哪顆 model。
    - **<prediction.csv>:** model預測的output，請同學不要寫死
- hw10_train.sh
  - 說明：訓練同學上繳的兩顆 model
  - **用法: bash hw10_train.sh <train.npy> <model>**
    - **<train.npy>**: 助教這邊存放 **train.npy** 的路徑，請同學不要寫死
    - **<model>**: 訓練完 autoencoder model 之後要存檔的路徑，請同學不要寫死。用法同 testing
  - 範例:
    - Training: bash hw10_train.sh ~/data/train.npy ~/models/baseline.pth
    - Evaluation: bash hw10_test.sh ~/data/test.npy ~/models/baseline.pth ~/outputs/prediction.csv

Reproduction

- Report 及 reproduction 中所指的 score 為 public leaderboard 上的 score。
- 原則上助教只會執行 testing 的 script。請確保上傳的 model 的 testing 結果與 Kaggle 上的結果誤差在 **±3%** 之間，若超過以上範圍，才會執行 training 的 script，若一樣誤差超過 **±3%**，Kaggle 將不予計分。
- Testing 執行時間上限為 **10** 分鐘。
- Training 執行時間上限為 **30** 分鐘。

FAQ

- 若有其他問題，請在 FB 社團貼文裡或寄信至助教信箱，**請勿直接私訊助教**。
- 助教信箱：ntu-ml-2020spring-ta@googlegroups.com

Links

- Kaggle: https://www.kaggle.com/c/ml2020spring-hw10
- Colab: https://reurl.cc/8GlnEy
- Report template: https://reurl.cc/204g16
- 遲交表單: https://bit.ly/39d2x2m