Machine Learning HW12

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

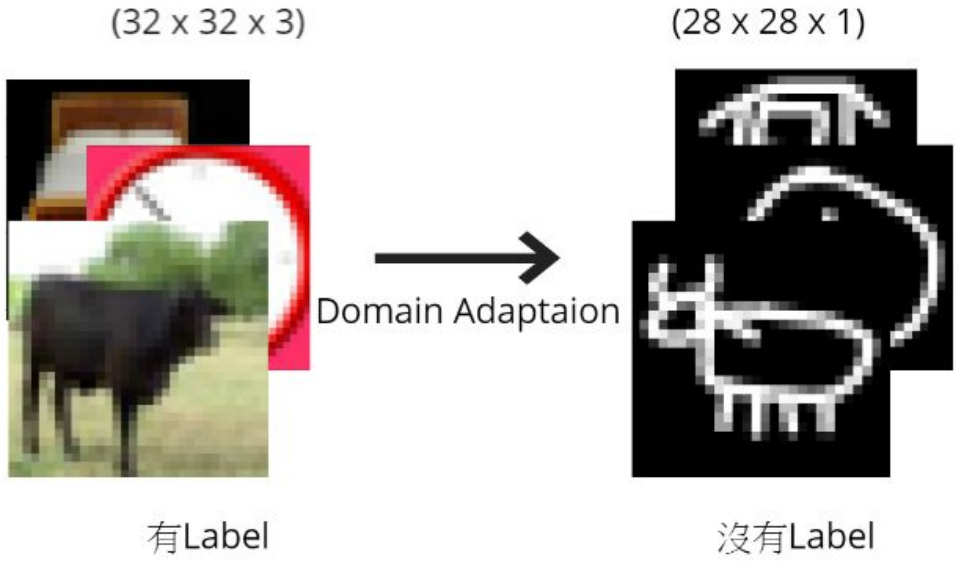
Outline

- Task Description
- Guideline
- Dataset & Kaggle
- Regulations & Grading Policy

Task Description

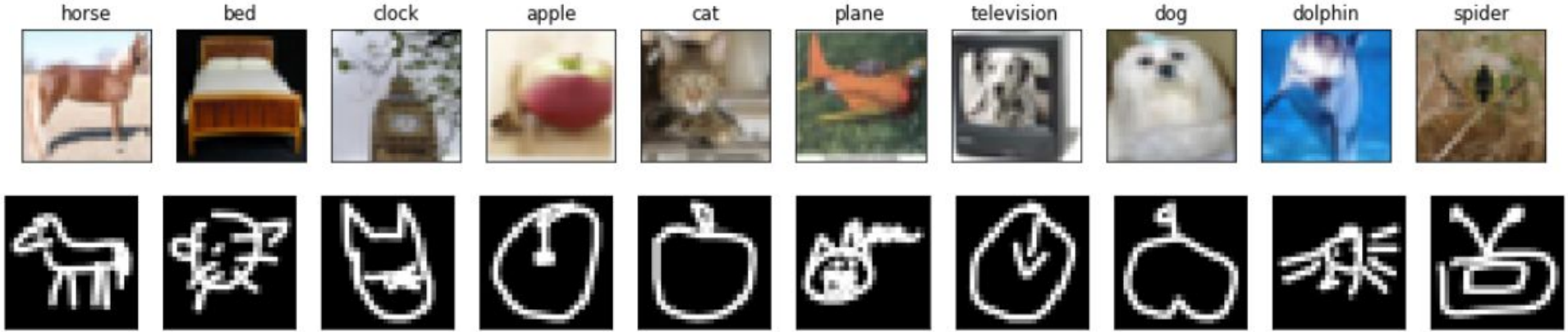
利用Domain adversarial training
讓model把domain B當作domain A
以達到成功預測

- Domain Adaptation: 讓模型可以在訓練時只需要 A dataset label，不需要 B dataset label 的情況下提高 B dataset 的準確率。(A dataset & task 接近 B dataset & task)
- 給定真實圖片 & 標籤以及大量的手繪圖片，請設計一種方法使得模型可以預測出手繪圖片的標籤為何。



Dataset

- Training : 5000 張真實圖片 + label, 32 x 32 RGB
- Testing : 100000 張手繪圖片, 28 x 28 Gray Scale
- Label: 總共需要預測 10 個 class, 如下圖所示。
    - 資料下載下來是以 0 ~ 9 作為label



Data Format

- 解壓縮後用以下幾行就能以 dataloader 使用。

```
source_dataset = ImageFolder('train_data', transform=YOUR_SOURCE_TRANCEFORM)
target_dataset = ImageFolder('test_data', transform=YOUR_TARGET_TRANCEFORM)
source_dataloader = DataLoader(source_dataset, batch_size=32)
target_dataloader = DataLoader(target_dataset, batch_size=32)
```

Submission Format

- 第一行為 id, label。
- 接下來 10 萬行分別代表第 k 張圖片的 label。
- Evaluate Metrics = @1 Accuracy。

```
id,label
0,0
1,8
2,1
3,1
4,0
5,0
6,6
7,7
8,9
9,9
```

Regulations

- **禁止手標 label 或上網尋找 label(有標或找 dataset 但沒用在 model 上也一樣)，一被發現或檢舉，該作業以 0 分計。**
- **禁止使用外來的 pre-trained model，即使是 torchvision 的也一樣。**

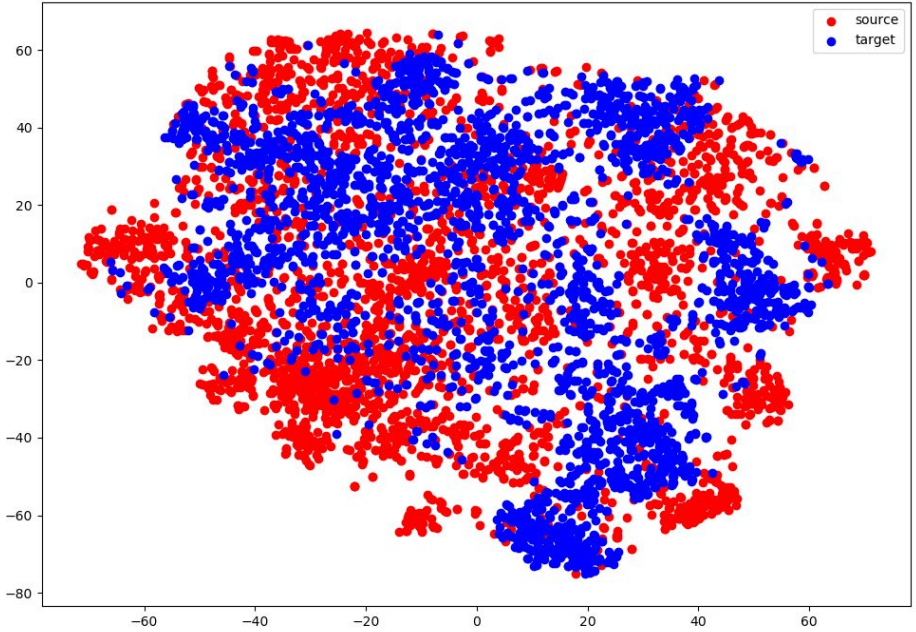
Report

1. 請描述你實作的模型架構、方法以及 accuracy 為何。其中你的方法必須為 domain adversarial training 系列(就是你的方法必須要讓輸入 training data & testing data 後的某一層輸出 distribution 要相近)。(2%)
2. 請視覺化真實圖片以及手繪圖片通過沒有使用 domain adversarial training 的 feature extractor 的 domain 分布圖。(2%)
3. 請視覺化真實圖片以及手繪圖片通過有使用 domain adversarial training 的 feature extractor 的 domain分布圖。(2%)

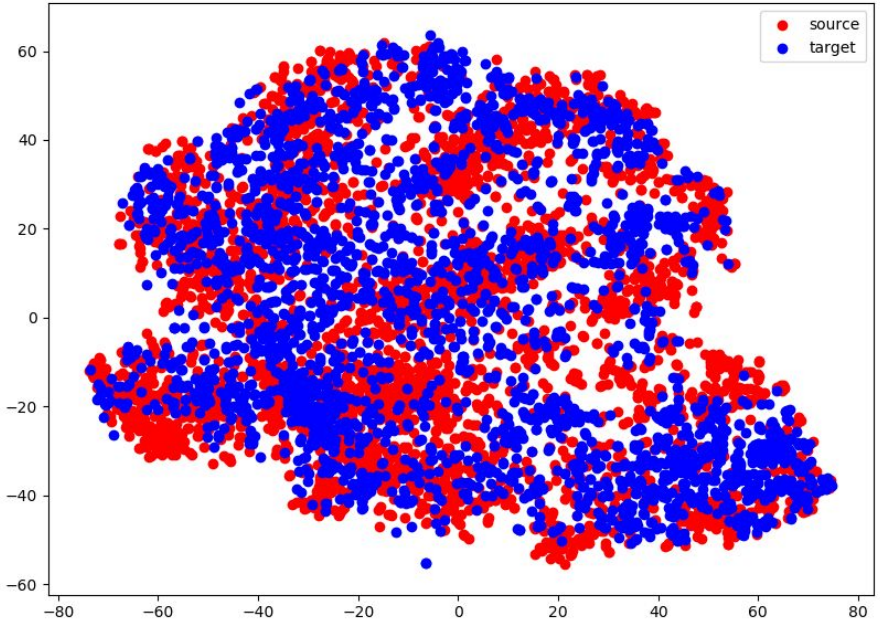
- Report Template

Report - Example

只要把feature extractor的latent降維就可以了
可以用PCA



without DaNN (第 2 小題)



DaNN (第 3 小題)

GitHub Submission

- GitHub 上 hw12-<account> 請至少包含：
  - report.pdf
  - hw12_test.sh（用以 reproduce 你的 private score，誤差容忍 0.5%，限時 **20** 分鐘。）
  - 各種 training 時需要的 Python 檔案

- 請不要上傳 dataset，**違者 0 分。**

Script Usage

1. 以下的路徑，助教在跑的時候會另外指定，請保留可更改的彈性，不要寫死。
2. Script usage:
   bash hw12_test.sh <data directory> <prediction file>
   data directory: 此資料夾中會包含 test_data 和 train_data 兩個資料夾
   prediction file: 輸出結果的 csv 檔路徑

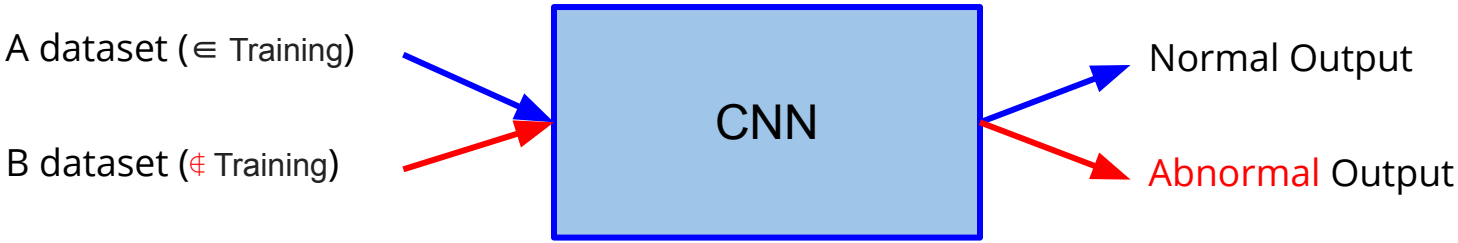
   請保留 training 的 Python 檔案，以防有任何狀況。

Links

- Kaggle
- Data

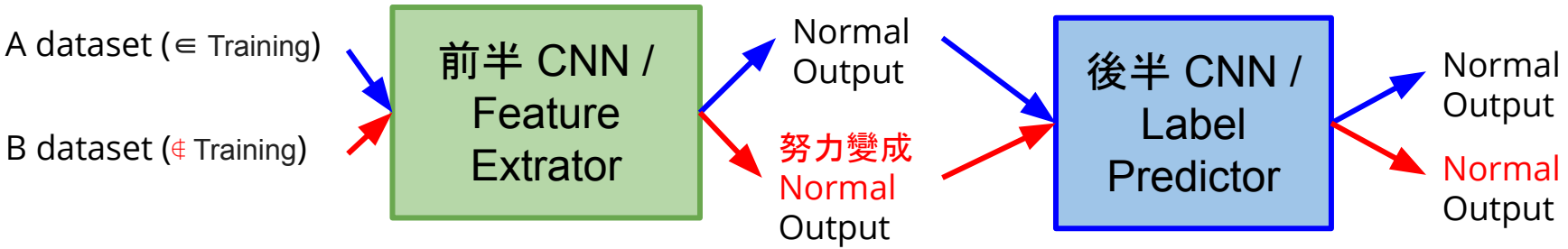
Guideline - DaNN (1/3)

- 這裡我們介紹最基礎的 DaNN (Domain-Adversarial Training of NNs)。
- 如果一個模型在測試時吃到不是與訓練集同個 distribution 的輸入，那麼輸出往往會爆走，如下圖。
- 而為什麼不能讓圖中的 CNN 在輸入 B dataset 輸出正常的 output？因為你並沒有 B dataset 的 label 使模型學習。



Guideline - DaNN (2/3)

- 為了因應這樣的情況，DaNN就將 CNN 先拆成兩個部分，並且想辦法讓前半的 CNN 在吃入兩個 A dataset & B dataset 後得到的 distribution 是相近的，那麼後半就會因為輸入是正常的 output，而發揮正常的功用。



Guideline - DaNN (3/3)

應該就是當backpropagation的時候
discriminator回傳的gradient會是:讓domain預測結果更準確
所以feature extractor就故意做反向的gradient 讓最後discriminator完全分不出來

- 而如何讓前半段的模型輸入兩種不同分布的資料，輸出卻是同個分布呢？最簡單的方法就是像 GAN 一樣導入一個 discriminator 來分辨輸入是哪個 dataset，並讓 feature extractor 來騙過 discriminator 即可。
- colab tutorial