ML2017 Fall HW6

TAs
ntu.mlta@gmail.com

Outline

- Unsupervised Learning & Dimension Reduction
  - Principal Components Analysis (PCA) of colored faces
  - Visualization of Chinese word embedding
  - Image clustering

PCA of colored faces - outline

- 學習用 numpy 實做 PCA 以達到 dimensionality reduction 的目的
- 跟以往不同，這次是對彩色的臉做 PCA。
- 數據集來自 Aberdeen University 的 Prof. Ian Craw，並經過助教們的挑選及對齊，總共有 415 張 600 X 600 X 3 的彩圖。
- 連結： https://drive.google.com/open?id=1_zD31Iglz6eTh55ushu-5dtciatuVyPy

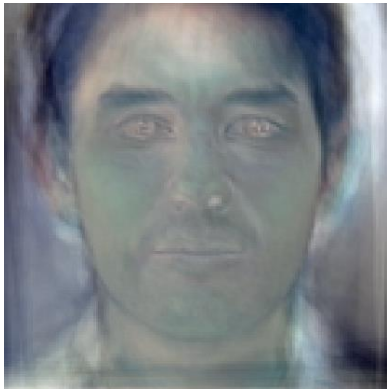
PCA of colored faces - requirements

- 只能用 numpy.linalg.svd 或 np.linalg.eig 實做PCA
- 只能用 scikit-image 讀寫圖片
- 也就是說程式會在只有 numpy 和 scikit-image 的環境下執行。
- 程式要求在三分鐘內，在與 pca.sh 相同的目錄中儲存 reconstruction.jpg 。
- $1 是所有照片的資料夾（相對路徑）
- reconstruction.jpg 是 $2 這張照片用前四個 Eigenfaces 重建的結果。
- 執行方式：**bash pca.sh $1 $2**，例如：**bash pca.sh ../imgs 414.jpg**

PCA of colored faces - report qeustions

1. 請畫出所有臉的平均。
2. 請畫出前四個 Eigenfaces，也就是對應到前四大 Eigenvalues 的 Eigenvectors。
3. 請從數據集中挑出任意四個圖片，並用前四大 Eigenfaces 進行 reconstruction，並畫出結果。
4. 請寫出前四大 Eigenfaces 各自所佔的比重，也就是 $\frac{s_i}{\sum s_j}$，請用百分比表示並四捨五入到小數點後一位。

PCA of colored faces - reminder



- 請記得先減去平均再計算 Eigenfaces, Eigenvalues
- Eigenfaces 是奇怪的顏色是正常的，如右上圖（第十個Eigenface)
- 因為 Eigenfaces 會有負值，因此在畫圖時，請用以下方式轉換：
  - `M -= np.min(M)`
  - `M /= np.max(M)`
  - `M = (M * 255).astype(np.uint8)`
- 程式只會執行最多三分鐘。
- 只能 `import` numpy 和 skimage
- 程式的結果是有標準答案的（可容許每個值相差 `±1` 以內），可以事先和同學比看看

Visualization of Chinese word embedding - outline

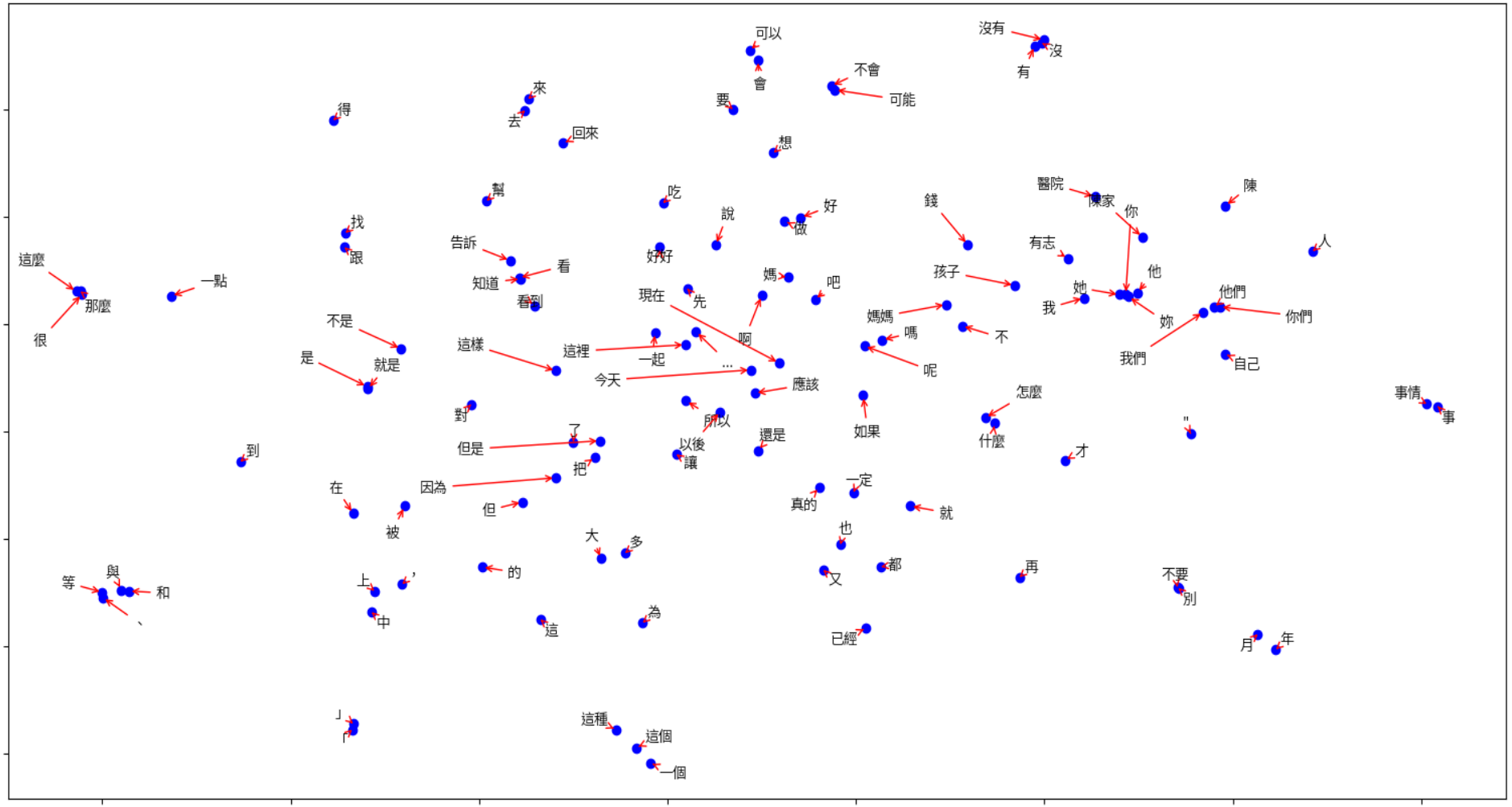
- 用任何 word2vec 的套件在中文的句子上訓練中文字的 word embedding
  - 有 python-package 的常用 word2vec 套件：gensim, word2vec, glove-python, glove
- 用任何dimension reduction的演算法在二維平面上視覺化 word embedding
  - 可以使用任何 dimension reduction 的演算法，但建議使用 TSNE
- 從視覺化的結果觀察 word embedding 訓練的成果

Visualization of Chinese word embedding - data

- 從三個 Final Project 的 training data 各取其中有中文句子的部份：
  - TV Conversation : provideData/training_data/1_train.txt ~ 5_train.txt
  - Chinese QA : train-v1.1.json
  - Listen & Translate : data/train.caption
- 用 '。' 或 '\n' 當做句子之間的分界，然後去掉長度小於6的
- 總共有 578810 句
- 連結：https://drive.google.com/open?id=1E5lElPutaWqKYPhSYLmVfw6olHjKDgdK

Visualization of Chinese word embedding - reminder

- 建議用 jieba 分詞
- 因為 jieba 預設主要是簡體字，建議使用繁體分詞更好的 dict.txt.big
  - 從連結下載詞典，然後用 jieba.set_dictionary (path_to_downloaded_dict.txt.big)
- Visualization 的時候，只針對出現次數 ≥K 的詞，建議 6000 ≥ K ≥ 3000
- 用 adjustText 避免圖表文字的重疊
- 用 matplotlib 作圖的話，要注意中文字體的設定，否則會出現亂碼
- 禁止使用 pretrained word embedding


可以
沒有
沒
有
會
不會
可能
要
來
去
得
回來
想
幫
吃
說
好
做
錢
醫院
陳家
陳
你
人
有志
找
跟
告訴
看
知道
看到
好好
媽
吧
孩子
他
她
我
妳
他們
你們
我們
自己
這麼
那麼
很
一點
現在
先
啊
媽媽
嗎
不
呢
不是
是
就是
這樣
這裡
一起
...
今天
應該
對
所以
還是
如果
怎麼
什麼
"
才
事情
事
到
了
但是
以後
讓
把
在
因為
但
被
真的
一定
就
也
大
多
都
又
再
不要
別
等
與
和
、
上
'
中
的
這
為
已經
月
年
」
「
這種
這個
一個

Visualization of Chinese word embedding - report questions

1. 請說明你用哪一個 word2vec 套件，並針對你有調整的參數說明那個參數的意義
2. 在 Report.pdf 上放上你visualization的結果
3. 請討論你從 visualization 的結果觀察到什麼？

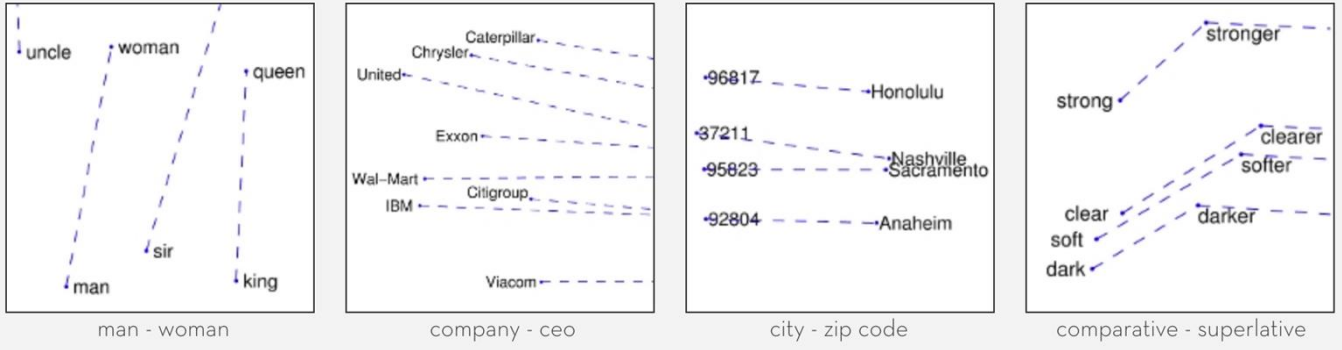


Image clustering - outline

- 目標：分辨給定的兩張 images 是否來自同一個 dataset
  - 所有的 image 都來自兩個不同的 dataset
  - 除了 image 本身之外，沒有任何 label
  - 只能用我們給的 data，不能使用額外的 dataset (包括用額外資料 train 的 model)
  - 在 kaggle deadline 之後會公布一個小型的 dataset，包含 10000 張 images。這個 dataset 前 5000 張 images 跟後 5000 張 images 是分別從兩個 dataset 得到的。到時候請大家對這個 dataset 做 visualization

Image clustering - evaluation

- F1-Score
- $$F1 = 2\frac{p \cdot r}{p + r} \quad \text{where} \quad p = \frac{tp}{tp + fp}, \quad r = \frac{tp}{tp + fn}$$

	prediction positive	prediction negative
ground true positive	true positive (tp)	false negative (fn)
ground true negative	false positive (fp)	true negative (tn)

Image clustering - evaluation (cont.)

	prediction positive	prediction negative
ground true positive	true positive (tp)	false negative (fn)
ground true negative	false positive (fp)	true negative (tn)

- simple example
predict	1	1	1	1	0	0	0	0	0	0
ground true	1	0	0	0	0	0	0	0	0	1
result	tp	fp	fp	fp	tn	tn	tn	tn	tn	fn

  - tp = 1, fp = 3, fn = 1, tn = 5
  - p = 1 / (1+3) = 0.25, r = 1 / (1+1) = 0.5
  - F1 = 2*0.25*0.5 / (0.25+0.5) = 0.333

$$F1 = 2\frac{p \cdot r}{p + r} \text{ where } p = \frac{tp}{tp + fp}, \quad r = \frac{tp}{tp + fn}$$

Image clustering - data

- 總共有 140000 張 image，都是黑白圖片
- image.npy.zip
  - 輸入指令 unzip image.npy.zip，會得到一個檔案叫做 image.npy
  - 使用 np.load() 讀取 image.npy，會得到一個 140000x784 的 ndarray
  - 每一個 row 都代表一張 28x28 image
- visualization.npy (kaggle deadline 之後公布在 kaggle 上)
  - 使用 np.load() 讀取 visualization.npy，會得到一個 10000x784 的 ndarray
  - 前 5000 張 images 來自 dataset A，後 5000 張 images 來自 dataset B

Image clustering - data (cont.)

- test_case.csv
  - 每一行都有 ID, image1_index, image2_index，總共有 1,980,000 筆測資
  - ID: test case index
  - image1_index: 對應到 image.npy 裡的 row index
  - image2_index: 對應到 image.npy 裡的 row index
- sample_submission.csv
  - 第一行是 “ID,Ans”
  - 之後每一行都會有 test case ID，以及對這個 test case 的 prediction
  - 如果 test case 的兩張 image 預測後是來自同一 dataset，Ans 的地方就是 1，反之是 0

Image clustering - methods

- 如果直接在原本的 image 上做 cluster，結果會很差 (有很多冗餘資訊)

  => 需要更好的方式來表示原本的 image

- 為了找出這個更好的方式，可以先將原始 image 做 dimension reduction，用比較少的維度來描述一張 image
  - 可以試試 PCA, SVD, t-SNE, auto-encoder, or anything to represent an image in lower dimension

Image clustering - methods (cont.)

- 接著對降維過後過後的數據做 cluster
  - cluster：可以試試 K-means
- 或者你可以衡量兩個降維過後的 images，他們之間的相似度 (similarity)。如果相似度大於一個設定好的 threshold，就把這兩個 images 當成同一類別
  - 算 similarity 的方法：euclidean distance, cosine similarity.....

Image clustering - methods (cont.)

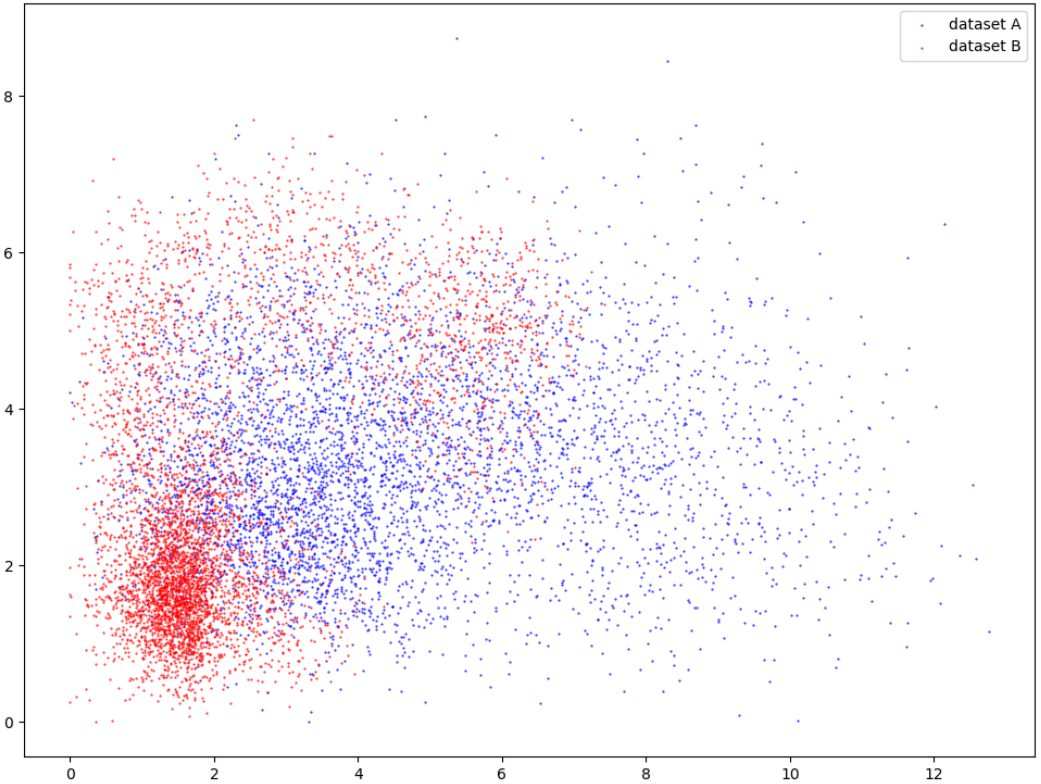
- 其他可能有幫助的事：
  - 必須找個方法來衡量方法的好壞，一個直覺的方法是利用降維過後的 feature 去 reconstruct 成原本的 image。如果 reconstruct 的結果越接近原本的 image，可以一定程度的代表你抽出來的 feature 越好
  - 對原始 image 做 data augmentation
  - try different number of cluster
  - 看看老師 unsupervised learning 上課內容

Image clustering - report qeustions

1. 請實作兩種不同的方法，並比較其結果。(不同的降維方法或不同的 cluster 方法都可以算是不同的方法)
2. 預測 visualization.npy 中的 label，在二維平面上視覺化 label 的分佈。(用 PCA, t-SNE 等工具把你抽出來的 feature 投影到二維，或簡單的取 feature 的前兩維)
3. visualization.npy 中前 5000 個 images 來自 dataset A，後 5000 個 images 來自 dataset B。請根據這個資訊，在二維平面上視覺化 label 的分佈，接著比較和自己預測的 label 之間有何不同。(visualization.npy 將在 Kaggle deadline 之後公布在 Kaggle 上)

*2 & 3 題請用 image.npy train 好的模型去預測 visualization.npy

- 取降維過後的 feature 前兩個維度作圖



- 把降維過後的 feature 再用 t-SNE 投影到二維

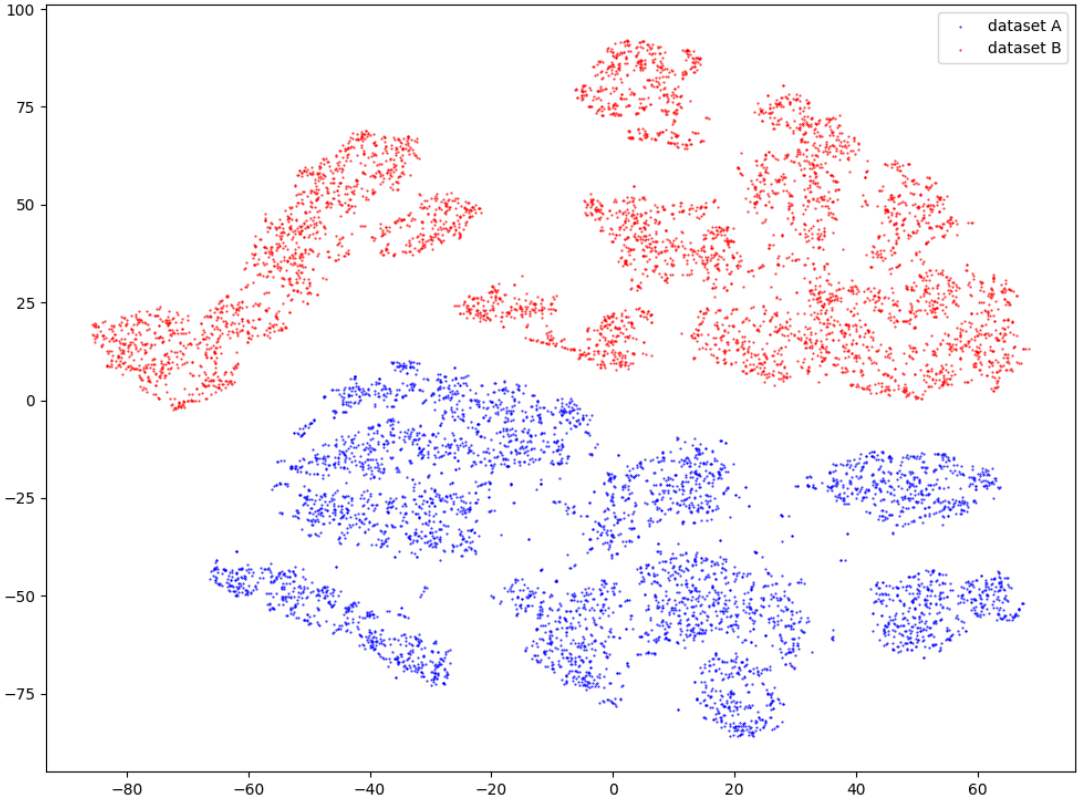


Image clustering - kaggle

- kaggle_url：
- 請至 kaggle 創帳號登入，需綁定 NTU 信箱。
- 個人進行，不需組隊。
- 隊名：學號_任意名稱 (ex. b02902000_日本一級棒)，旁聽同學請避免學號開頭。
- 每日上傳上限 5 次。
- test set 的資料將被分為兩份，一半為 public，另一半為 private。
- 最後的計分排名將以 2 筆自行選擇的結果，測試在 private set 上的準確率為準。
- kaggle 名稱錯誤者將不會得到任何 kaggle 上分數。

Deadline

1. Kaggle: 1/11 23:59 (GMT+8)
2. Report and source code: 1/12 23:59 (GMT+8)

助教會在 deadline 一到就 clone 所有程式，並且**不再重新 clone 任何檔案**

Policy I - repository

- github 上 ML2018SPRING/hw4/ 裡面請至少包含：
  - Report.pdf
  - pca.sh
  - hw4.sh (for image clustering 那題，這次只需上傳結果最好的方法)
  - your python files
  - your model files (can be loaded by your python file)

- **請不要上傳 dataset**
- 如果你的 model 超過 github 的最大容量，可以考慮把 model 放在其他地方 (http://slides.com/sunprinces/deck-16#/2%EF%BC%89)。
- model 可以是多個檔案，例如 keras model，或者是 image id mapping file。如果你的 code 需要極長的執行時間，可以把 image cluster 後的結果寫進一個 file，並在執行時讀取它。

Policy II – source code

- **Python Only**，請使用 Python 3.5+
- **PCA of colored faces** 的部份只能使用 numpy 和 scikit-image
- **Chinese word embedding** 的部分不限定套件
- **Image clustering** 的部份可以使用 Keras 2.0.8, Tensorflow1.3.0, pytorch 0.2.0, h5py2.7.0 , Numpy, scipy, Pandas 0.20+, matplotlib, scikit-image, pillow, scikit-learn, Python Standard Lib.
- **只可使用限定的 package，以及 python 內建的 package，並且限定使用 Tensorflow 作為 Keras 的backend。需要其它套件，請來信詢問。** 若 import 其他東西，或是使用不同版本，造成批改錯誤，將不接受修正。
- 不能使用額外 data 來 training (包括 pre-training)
- 不能 call 其他線上 API
- 請附上訓練好的 model (及其參數)

Policy III – bash script

- 與之前作業相同，請在script中寫清楚使用python版本
- 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
  - PCA of colored faces: (詳見第4頁) 時限三分鐘

    bash pca.sh <images path> <target image>
  - Image clustering:

    bash hw6.sh <**image.npy** path> <test_case.csv path> <prediction file path>

Policy IV - programs scores

- PCA of colored faces: (1%) 正確性
- Kaggle Rank
  - (0.8%) kaggle 上和 reproduce 都超過 public leaderboard 的 simple baseline 分數
  - (0.8%) kaggle 上和 reproduce 都超過 public leaderboard 的 strong baseline 分數
  - (0.8%) kaggle 上和 reproduce 都超過 private leaderboard 的 simple baseline 分數
  - (0.8%) kaggle 上和 reproduce 都超過 private leaderboard 的 strong baseline 分數
  - (0.8%) 2018/1/4 23:59 (GMT+8) 前 kaggle 上超過 public simple baseline 分數
  - (BONUS) kaggle 排名前五名 (且願意上台跟大家分享的同學)
- 前五名排名以 private 平均為準，屆時助教會公布名單
- **hw4.sh 的結果必須超過 public simple baseline 否則程式部分將不會有任何分數。**

Policy V - report questions and scores

- PCA of colored faces
  - (.5%) 請畫出所有臉的平均。
  - (.5%) 請畫出前四個 Eigenfaces，也就是對應到前四大 Eigenvalues 的 Eigenvectors。
  - (.5%) 請從數據集中挑出任意四個圖片，並用前四大 Eigenfaces 進行 reconstruction，並畫出結果。
  - (.5%) 請寫出前四大 Eigenfaces 各自所佔的比重 (explained variance ratio)，請四捨五入到小數點後一位。
- Visualization of Chinese word embedding
  - (.5%) 請說明你用哪一個 word2vec 套件，並針對你有調整的參數說明那個參數的意義。
  - (.5%) 請在 Report 上放上你 visualization 的結果。
  - (.5%) 請討論你從 visualization 的結果觀察到什麼。
- Image clustering *2 & 3 題請用 image.npy train 好的模型去預測 visualization.npy
  - (.5%) 請比較至少兩種不同的 feature extraction 及其結果。(不同的降維方法或不同的 cluster 方法都可以算是不同的方法)
  - (.5%) 預測 visualization.npy 中的 label，在二維平面上視覺化 label 的分佈。
  - (.5%) visualization.npy 中前 5000 個 images 來自 dataset A，後 5000 個 images 來自 dataset B。請根據這個資訊，在二維平面上視覺化 label 的分佈，接著比較和自己預測的 label 之間有何不同。

Policy VI - reminders

- Report 強烈建議使用中文作答。
- 請根據 Report Template 寫Report，如果想要用其他排版模式也請註明題號以及題目內容（請勿擅自更改題號）。
- 請交 pdf 檔，檔名為 Report.pdf
- Collaborators 請附上學號與姓名
- 若有問題，請寄信詢問。並在標題打上 [HW4]

小老師制度（手把手教學）

1. 在 1/4 以前超過 simple baseline 並願意在 1/5 在上課時間教導同學撰寫作業六程式，請填寫一下表單：https://goo.gl/forms/xSn2IjAaXMbouE733
2. 1/4 將公布小老師名單在作業網頁，人數太多將以符合以下標準的同學為主:
   1. 沒有當過小老師
   2. Kaggle Public Leaderboard 成績排名較高 (但請不要因此想overfit public set)
3. 小老師當次成績 +1%

Other Policy

- script 錯誤直接 0 分。若是格式錯誤，請在公告時間內找助教修好，修完kaggle分數*0.7
- Kaggle 超過 deadline 直接 shut down，可以繼續上傳但不計入成績
- Github 遲交一天(*0.7)，不足一天以一天計算，不得遲交超過兩天，有特殊原因請找助教
- Github 遲交表單：
  - code: https://goo.gl/forms/U739TuuKJE3QDdWb2 (遲交才需填寫)
  - report: https://goo.gl/forms/ulB0FqGngd8cmvjf2 (遲交才需填寫)
- 遲交請「先上傳程式」Github 再填表單，助教會根據表單填寫時間當作繳交時間
- 請勿使用任何其他非助教提供的 data，否則以 0 分計算
- 上傳的 model 總和大小建議在 500 MB以內

FAQ

1. 作業網址：Link
2. 若有其他問題，請po在FB社團裡或寄信至助教信箱，**請勿直接私訊助教**。
3. 助教信箱：ntumlta2018@gmail.com

Link

1. 雲端使用方法：http://slides.com/sunprinces/deck-16#/2 ）
2. Kaggle：https://www.kaggle.com/t/a9a500b3bcc447c68b5a2285bf55b822
3. 作業網址：https://ntumlta.github.io/2017fall-ml-hw6/
4. Report template: https://docs.google.com/document/d/13c6RqKvcYdSBMxq4yFljUUDnj5fXmv9cGa01yfFwcFU/edit?usp=sharing
5. Github 遲交表單: 未開放
6. 小老師報名表單：未開放