作業五　學號：R06922130　系級：資工碩一 姓名：葉韋辰

1. 請比較有無 normalize(在 rating 上)的差別。並說明如何 normalize (1%)

Collaborators：黃禹程 R06944034、鄭克宣 R06921083、丁縉楷 R06922129、葉孟元 R04921094

NORMALIZATION	PRIVATE 分數	PUBLIC 分數	平均
有	**0.85077**	**0.84926**	**0.850015**
無	0.85676	0.85335	0.855055

- ***架構***：*如圖 1-1 所示*
- ***參數***：*batch=512, earlystopping, opt=adam, latent_dim=120, validation_split=0.1*
- ***做法***：*preprocess 時先用* ***np.mean*** *與* ***np.std*** *分別計算 training data y 之平均值與標準差，再透過* $y := \frac{y - mean\_y}{std\_y}$ *做 normalization*
- ***結果***：*有無 Normalization 皆通過 public 與 private 之 simple 與 strong baseline，然有 normalization 表現****略優****，不排除是 shuffle training data 造成之差異*

```
weichenyeh@weichenyeh-B9440UA:~/Dropbox/MLee/Full_hw5/y
weichenyeh@weichenyeh-B9440UA:~/Dropbox/MLee/Full_hw5/y$ python3 train.py
Using TensorFlow backend.
Layer (type)                 Output Shape              Param #   Connected to
==================================================================================================
input_1 (InputLayer)         (None, 1)                 0
input_2 (InputLayer)         (None, 1)                 0
embedding_1 (Embedding)      (None, 1, 120)            724920    input_1[0][0]
embedding_2 (Embedding)      (None, 1, 120)            474360    input_2[0][0]
flatten_1 (Flatten)          (None, 120)               0         embedding_1[0][0]
flatten_2 (Flatten)          (None, 120)               0         embedding_2[0][0]
embedding_3 (Embedding)      (None, 1, 1)              6041      input_1[0][0]
embedding_4 (Embedding)      (None, 1, 1)              3953      input_2[0][0]
dot_1 (Dot)                  (None, 1)                 0         flatten_1[0][0]
                                                                 flatten_2[0][0]
flatten_3 (Flatten)          (None, 1)                 0         embedding_3[0][0]
flatten_4 (Flatten)          (None, 1)                 0         embedding_4[0][0]
add_1 (Add)                  (None, 1)                 0         dot_1[0][0]
                                                                 flatten_3[0][0]
                                                                 flatten_4[0][0]
==================================================================================================
Total params: 1,209,274
Trainable params: 1,209,274
Non-trainable params: 0
```

圖 1：MF 之架構圖

2. 比較不同的 latent dimension 的結果(1%)

Collaborators：黃禹程 R0694403、鄭克宣 R06921083、丁縉楷 R06922129、葉孟元 R04921094

DIM	PRIVATE 分數	PUBLIC 分數	平均
60	0.85017	0.85019	0.85018
120	**0.84955**	**0.84931**	**0.84943**
240	0.85005	0.84993	0.84999

- ***架構***：*如圖 1-1 所示*
- ***參數***：*batch=512, earlystopping, opt=adam, validation_split=0.1*
- ***結果***：*dimension=****120*** *效果****略優****，直觀上會認定 dimension 愈大愈好，但從實驗實據可發現不必然成立，不過不排除是 shuffle training data 造成之差異*

3. 比較有無 bias 的結果(1%)

Collaborators：黃禹程 R06944034、丁縉楷 R06922129、葉孟元 R04921094

BIAS	PRIVATE 分數	PUBLIC 分數	平均
有	**0.84955**	**0.84931**	**0.84943**
無	0.85091	0.85041	0.85066

- ***架構：****如圖 1 所示*
- ***參數：****batch=512, earlystopping, opt=adam, latent_dim=120, validation_split=0.1*
- ***做法：****增設 user bias 與 movie bias 之 embedding layers，將參數 embeddings_initializer 設為'zero'，最後與相乘後的 user vector 與 movie vector 相加*
- ***結果：****有* ***bias*** *之表現****略優***

4. 請試著用 DNN(投影片 p.28)來解決這個問題，並且說明實做的方法(方法不限)。並比較 MF 和 NN 的結果，討論結果的差異(1%)

Collaborators：黃禹程 R06944034、丁縉楷 R06922129

模型	PRIVATE 分數	PUBLIC 分數	平均
MF	**0.85077**	**0.84926**	**0.850015**
DNN	0.88690	0.88659	0.886745

- ***架構：*** ***MF*** *如圖 1 所示，DNN 如圖 2 所示*
- ***參數：****batch=512, earlystopping, opt=adam, latent_dim=120, validation_split=0.1*
- ***做法：****參考 TA Hour 投影片，concatenate user embedding 與 movie embedding，再放入 DNN*
- ***結果：****DNN* ***略遜****，礙於時間有限並未最佳化 DNN 模型之參數與架構*

```
Layer (type)                 Output Shape              Param #     Connected to
==================================================================================================
input_1 (InputLayer)         (None, 1)                 0
__________________________________________________________________________________________________
input_2 (InputLayer)         (None, 1)                 0
__________________________________________________________________________________________________
embedding_1 (Embedding)      (None, 1, 60)             362460      input_1[0][0]
__________________________________________________________________________________________________
embedding_2 (Embedding)      (None, 1, 60)             237180      input_2[0][0]
__________________________________________________________________________________________________
flatten_1 (Flatten)          (None, 60)                0           embedding_1[0][0]
__________________________________________________________________________________________________
flatten_2 (Flatten)          (None, 60)                0           embedding_2[0][0]
__________________________________________________________________________________________________
concatenate_1 (Concatenate)  (None, 120)               0           flatten_1[0][0]
                                                                   flatten_2[0][0]
__________________________________________________________________________________________________
dense_1 (Dense)              (None, 150)               18150       concatenate_1[0][0]
__________________________________________________________________________________________________
dense_2 (Dense)              (None, 50)                7550        dense_1[0][0]
__________________________________________________________________________________________________
dense_3 (Dense)              (None, 1)                 51          dense_2[0][0]
==================================================================================================
Total params: 625,391
Trainable params: 625,391
Non-trainable params: 0
```

圖 2：DNN 之架構圖

5. 請試著將 movie 的 embedding 用 tsne 降維後，將 movie category 當作 label 來作圖(如投影片 p.29) (1%)

Collaborators：黃禹程 R06944034、鄭克宣 R06921083、丁縉楷 R06922129、蔡孟庭 R05922078

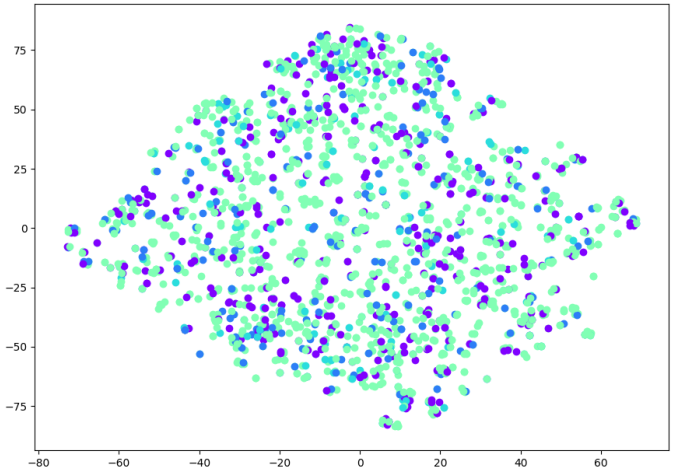

圖 3：TSNE

- ***作法：****將電影分成以下* ***4 類****並做* `tsne`
  - `'Adventure','Action','War','Western'`
  - `'Thriller','Crime','Horror'`
  - `'Documentary', 'Film-Noir' ,'Mystery' ,'Sci-Fi'`
  - `'Fantasy', 'Romance', 'Musical', 'Comedy','Drama', 'Children\'s','Animation'`
- ***結果：****如圖 3 所示，曾嘗試調整不同分類方式但並未得到更佳結果*

6. BONUS: 試著使用除了 rating 以外的 feature, 並說明你的作法和結果，結果好壞不會影響評分

Collaborators：黃禹程 R06944034、鄭克宣 R06921083、丁縉楷 R06922129、葉孟元 R04921094

額外 FEATURE	PRIVATE 分數	PUBLIC 分數	平均
有	0.86349	0.86334	0.863415
無	**0.85077**	**0.84926**	**0.850015**

- ***架構：****如圖 1 所示*
- ***參數：****`batch=512, earlystopping, opt=adam, latent_dim=120, validation_split=0.1`*
- ***做法：****將* `users.csv` *中的* `user` ***年齡****、****性別****與****職業****抽出，並架出相對應之* `embedding layers` *，至於其他資訊則直接捨棄*
- ***結果：額外*** ***`features`*** ***之****效果****略遜****，或許是肇因於並未最佳化其參數設定*