學號：R06723025 系級： 財金碩一 姓名：林耘寬

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

	Public	Private
With normalization	0.86144	0.86177
Without normalization	0.86179	0.86257

Normalize 的方法就是對 rating 減去 mean，並且除以標準差，使用 numpy 內建的函數，可以發現有 normalize 的結果較佳。

2. (1%)比較不同的 latent dimension 的結果。
(collaborator:)

Dim	Public	Private
32	0.89002	0.88809
64	0.86398	0.86447
128	0.86179	0.86257
256	0.86144	0.86177

可以發現在 dimension 越高的情況下，loss 通常也越低，同時它 loss 下降的速度也越快。

3. (1%)比較有無 bias 的結果。
(collaborator: r06922075 翁瑋)

	Public	Private
With bias	0.86144	0.86177
Without bias	0.86291	0.86474

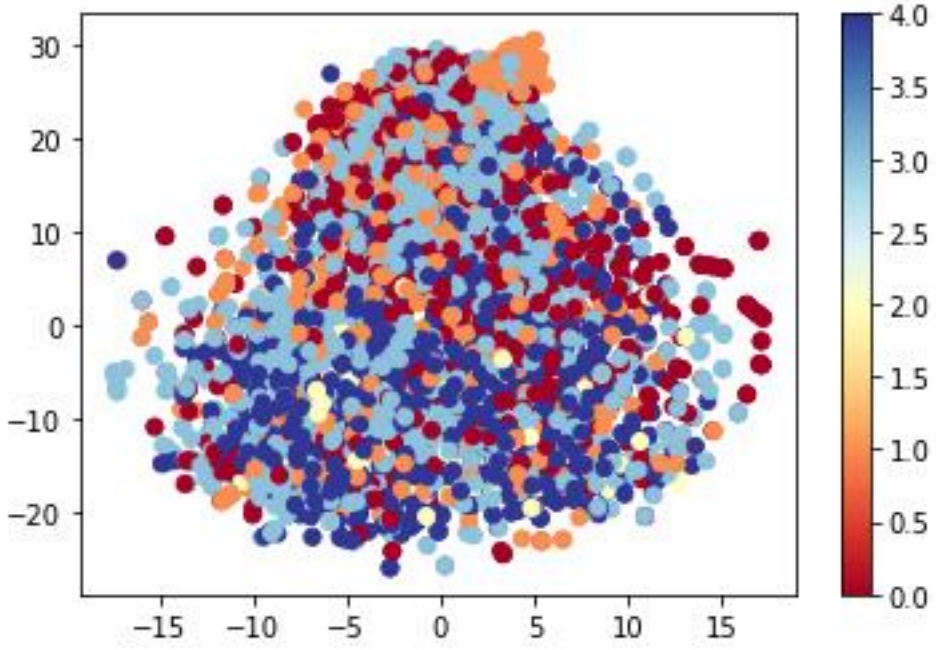
由數據很明顯可得知有加 bias 的結果好過沒加 bias 的，bias 可以用來解釋個人對於評分的偏好，或者大眾對於這部電影的偏好。

4. (1%)請試著用 DNN 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。
   (collaborator: r06922075 翁瑋)

	Public	Private
MF	0.86144	0.86177
DNN	0.87205	0.86903

DNN 我的作法是將 users 和 movies 兩個 embedding layers 從 dot 改成 concatenate，然後再接 DNN 的 layers，從實驗數據來看，DNN 的表現較差，在測試過程中發現，如果 layers 太多層或者 units 調太高好像很容易造成 overfitting，並且 DNN 訓練中收斂速度很快。

5. (1%)請試著將 movie 的 embedding 用 tsne 降維後，將 movie category 當作 label 來作圖。
   (collaborator: r06922075 翁瑋)



6. (BONUS)(1%)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。
   (collaborator:)