

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

做了 normalize 後 predict 的結果會好一點點，結果如下

non-normalized: private 0.84864 public 0.84782

normalized: private 0.84562 public 0.84559

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

我分別嘗試了3種 latent dimension 為64，128，256

64: val 0.84094 private 0.84660 public 0.84516

128: val 0.841648 private 0.84562 public 0.84559

256: val 0.842244 private 0.84743 public 0.84688

從以上結果來看 latent dimension 為128的時候 accuracy 最好

3. (1%)比較有無 bias 的結果。
(collaborator:)

有 bias 後 predict 的結果會好一點點，結果如下

without bias: private 0.84818 public 0.84749

with bias: private 0.84562 public 0.84559

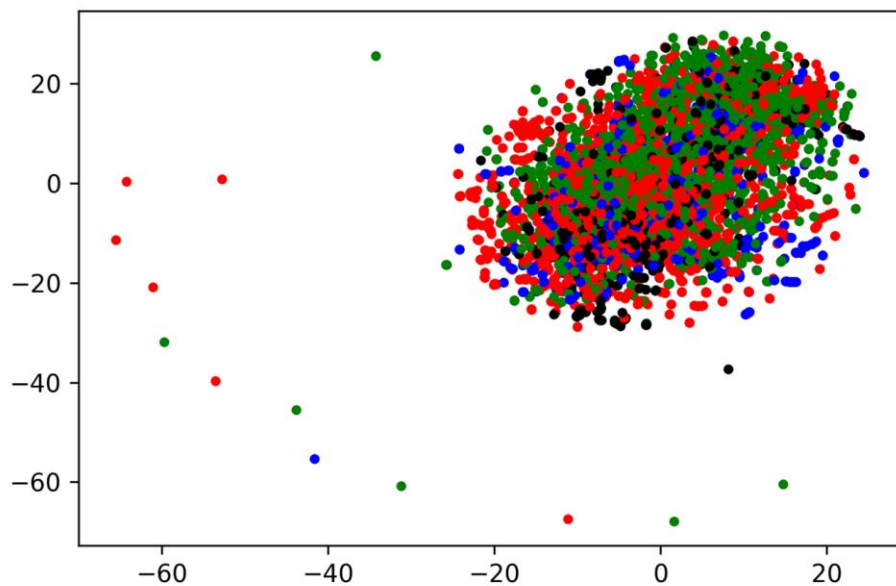
4. (1%)請試著用 DNN 來解決這個問題，並且說明實做的方法(方法不限)。
並比較 MF 和 NN 的結果，討論結果的差異。
(collaborator:)

用 DNN 做出來的結果會比較差，結果如下

DNN: private 0.85678 public 0.85533

MF: private 0.84562 public 0.84559

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



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:)

我加了電影類型，使用者性別，年齡，職業當作 feature 下去 train，結果比只有用 rating 的結果 好很多。

With additional feature: private 0.84562 public 0.84559