

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

(collaborator : None)

有 normalize (loss) : 0.873

無 normalize (loss) : 0.881

方法：取全部 train.csv 的 rating 得到 mean 和 std，並且把所有 train data 都減去 mean 後再除以 std 作為 target。predict test data 的時候將結果乘上 std 後再加上 mean。看起來有 normalize 表現上有變好。

2. (1%)比較不同的 latent dimension 的結果。

(collaborator : None)

Latent 1000 (loss) : 0.882

Latent 100 (loss) : 0.873

Latent 50 (loss) : 0.878

試了三種 Latent (1000,100,50)，最後選擇了表現最好的 100 作為最後的 model。

3. (1%)比較有無 bias 的結果。

(collaborator : None)

有 bias (loss) : 0.873

無 bias (loss) : 0.881

有加 bias 表現有變好很多，表示 rating 與每個 users 或 movies 各自的 bias 有關。

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 : None)

DNN (loss) : 0.865

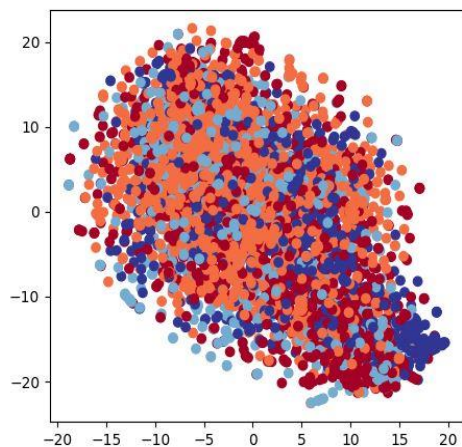
MF (loss) : 0.873

DNN 方法：Latent 一樣取 100，我把 Users 和 Movie 的 Embedding Concatenate 後，過一個兩層 Hidden Layers 的 DNN(分別 50 和 25 個 Neurons)，最後接到 Output (1 維)，loss 一樣選用 MSE。

討論：DNN 的表現比較佳，代表多加的 Hidden Layers 發揮了作用，也許是原本的 feature 特徵比較不明確，經過一些 feature transform 後變得比較好 predict 了。

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

(collaborator : None)



我把 movies 的 Genres 分成 4 大類

Class 1 : Animation、Children's、
Comedy、Fantasy、Sci-Fi

Class 2 : Romance、Drama、Musical

Class 3 : Adventure、Action、
Documentary、War、Western

Class 4 : Crime、Mystery、Film-Noir、
Documentary、War、Western

TSNE 後結果似乎分不太開，有點不理想……

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 : None)

Users：我使用了 Gender 和 Age 兩種 feature

Gender：分 2 種，男(M)、女(F)

Age：分 5 種，小孩(0~15)、青年(16~25)、青壯年(26~45)、中年(46~60)、老人(61+)

Movies：我使用了 Genres，分 18 類，類與類之間互相獨立

方法：我把 User 的 Embedding 與 Gender 和 Age Concatenate 後，過一層 Dense；

Movie 的 Embedding 與 Genres Concatenate 後，一樣也過一層 Dense。再將這兩個 Dense Concatenate 在一起再過兩層 Dense，最後接到 Output。

DNN + feature : 0.865

DNN only : 0.872

加了 feature 後的表現有明顯變好，表示 rating 跟 Age、Gender、Genres 都有關係。