

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

(collaborator:與 B04902021 陳弘梵同學討論)

答：

由下方實驗數據可以觀察到，在其他條件不變的情況下(latent dimension = 15, dropout = 0.2)，normalize 後，誤差值有一定幅度的下降。

Normalize 的方法為，將 rating 在訓練之前先標準化(訓練時，計算 rmse 時要乘上 std 值)，並將 mean、std 記錄下來，然後利用所存下的數值，將 predict 的結果先乘上 std，再加上 mean 來還原答案(需 clip 到 1.0~5.0 之間)。

	Training rmse	Validation rmse	Kaggle score
有 normalize	0.7850	0.8477	0.84849
沒有 normalize	0.8086	0.8491	0.84981

2. (1%)比較不同的 latent dimension 的結果。

(collaborator:無)

答：

由下表可看出，隨著 latent dimension 逐漸升高：

- 1) Training rmse 不斷下降
- 2) Validation rmse 先下降後上升 (dimension = 15, 25, 50 時最低)
- 3) Kaggle score (rmse) 先下降後上升(dimension = 15 時最低)

	Training rmse	Validation rmse	Kaggle score
Dim = 5	0.8485	0.8629	0.86435
Dim = 10	0.8248	0.8533	0.85556
Dim = 15	0.8086	0.8491	0.84981
Dim = 25	0.7786	0.8492	0.85118

Dim = 50	0.7737	0.8490	0.85135
Dim = 100	0.7465	0.8495	0.85153
Dim = 150	0.7393	0.8501	0.85179

註：Drop out = 0.2，有加上 bias

3. (1%)比較有無 bias 的結果。

(collaborator:無)

答：

由下表可看出，在加入了 bias 後，不僅是對 validation set 的誤差，在 kaggle score 上也能看見誤差值的下降，推論 bias 對於提升正確率有正向的影響。

	Training rmse	Validation rmse	Kaggle score
沒有加 bias	0.7382	0.8540	0.85398
有加 bias	0.7156	0.8507	0.85079

註：dimension = 256

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: 與 B04902089 林政豪同學討論)

答：

我的 Dnn 模型架構為，先將 user embedding、movie embedding (兩者同 Matrix Factorization 中的兩個 embedding) concatenate 起來，再丟進 4 層的 Dense (unit 分別為 256, 128, 64, 1), dropout 皆為 0.4。

MF 模型有加上 bias。

觀察結果發現 MF model 的結果比 DNN model 來的好(但依正常邏輯來說一個“好的” DNN model 所訓練出來的結果應該會比較好)，推測原因為我並沒有細調 DNN model 的參數(以及 dimension 的大小)，因此才會產生這樣的結果。

	Training rmse	Validation rmse	Kaggle score
MF 結果	0.7156	0.8507	0.85079
DNN 結果	0.8270	0.8703	0.87140

註：latent dimension = 256

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

(collaborator: 上學期修課同學 B04902097 陳家棋，本學期修課同學 B04705003 林子雋)

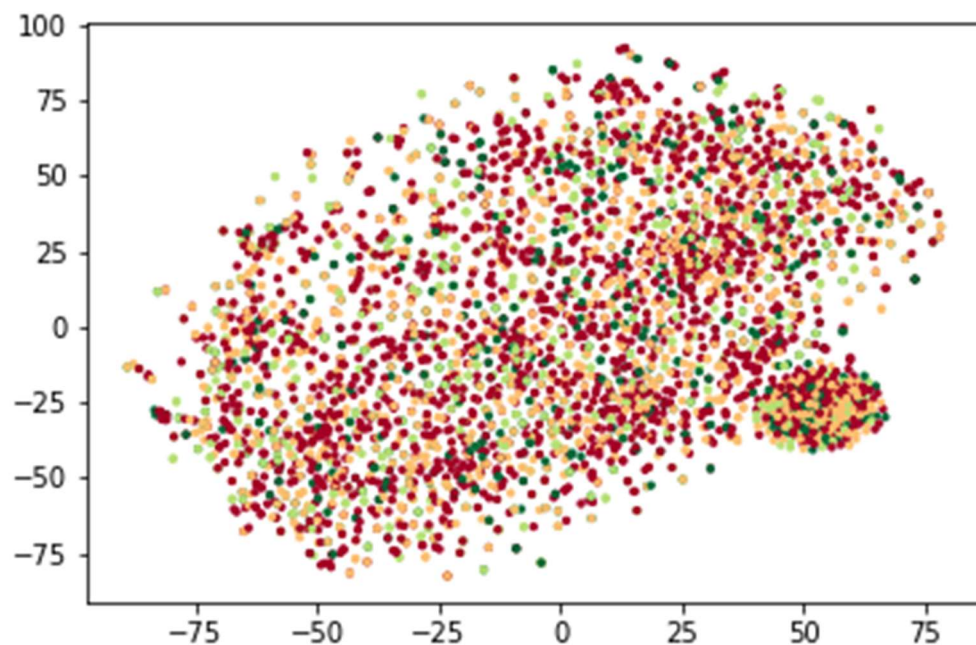
答：

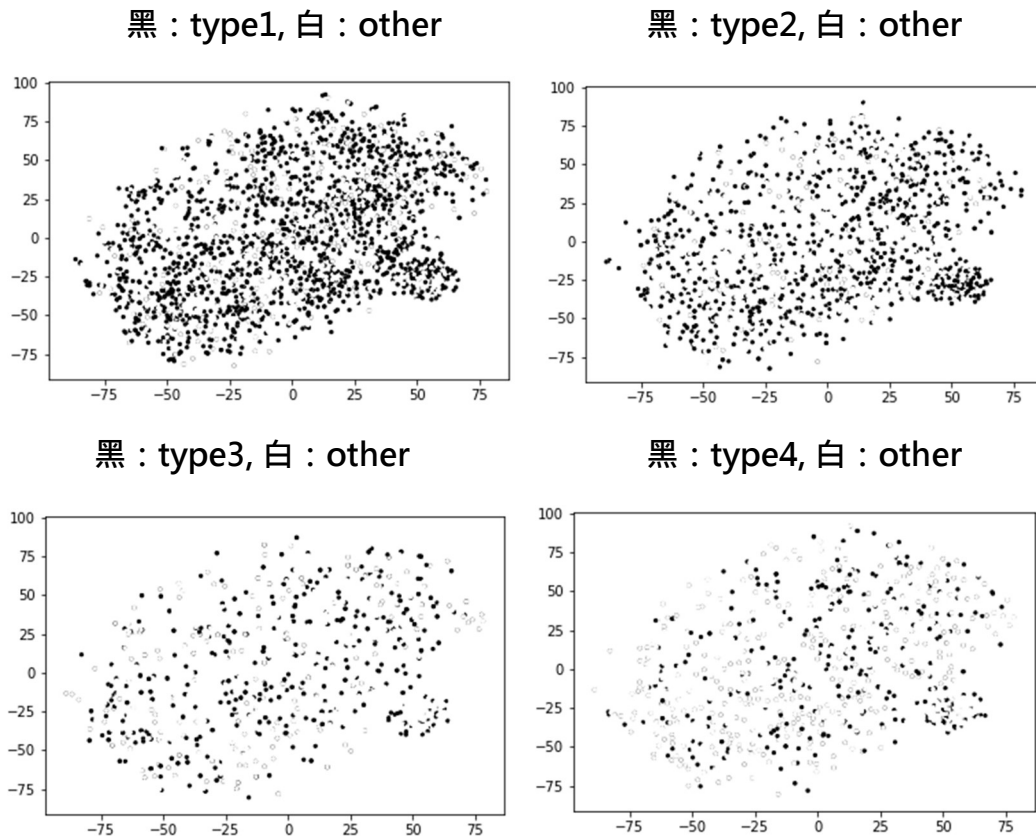
Type1 (紅色)：Drama, Musical

Type2 (橘色)：Thriller, Horror, Crime

Type3 (淺綠)：Adventure, Animation, Children's

Type4 (深綠)：Action, War





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: 資工三 B04902089 林政豪)

我的作法為, 從 users.csv, movies.csv 讀入所需的資料(從 users.csv 中, 取出 Gender(以 1 代表"M", 0 代表"F"), Age, Occupation; 從 movie.csv 中, 取出 Genres(encoding 成 0~17)), 再依照 training data(train.csv)中的 UserID, MovieID, 將相對應的資訊接到 training data 後訓練, 將做好的 2 個矩陣 concatenate 後, 放入 Dense 中(units 分別為 256, 128, 64, 1, activation function 為 elu, Dropout = 0.4)。結果如下方數據:

	Training rmse	Validation rmse	Kaggle score
Bonus model	0.8230	0.8603	0.85912