

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

(collaborator:無)

這裡實作的模型是 Dimension 為 250，並且將 output 做一層 8 個單元的 Neural Network，使用“`elu`”函數。並切出 10% 的 training data 作為 validation。Dropout Rate 為 0.1。

以下比較三種處理 Rating 的方式：

1.不處理。2.標準化。3.減去 1 再除以 4 (使資料介於 0~1 之間)。

由於經過處理後，RMSE 值也會改變，因此採用 Kaggle 分數作為標準。

方式	不處理	標準化	減去 1 再除以 4
Kaggle RMSE	0.84640	0.91132	0.90989

可以發現經過標準化或重新使 Rating 縮放到[0,1]，都會使 Kaggle 誤差值變高。

2. (1%)比較不同的 latent dimension 的結果。

(collaborator:無)

在第一題的模型中，不使用 `normalize`，調整 dimension 如下：

Dimension	10	30	50	100	150	200
Val. RMSE	0.8477	0.8573	0.8453	0.8464	0.8419	0.8462
Dimension	250	300	350	400	500	1000
Val. RMSE	0.8464	0.8474	0.8442	0.8455	0.8527	0.8518

可以看出在 Dimension 為 50~250 時，RMSE 較小，400 以上則 RMSE 偏大。但基本上因為幅度變動不大，且每次進行訓練時 RMSE 都會浮動，因此參考價值不大，或是要多做幾次做平均。

3. (1%)比較有無 bias 的結果。

(collaborator:無)

沿用第一題的模型，僅在有無 bias 上的差別。

方法	有 bias	無 bias
Validation RMSE	0.8464	0.8549

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:無)

僅使用 DNN，也就是將 UserID 和 MovieID 輸入，得到 Rating。實作方式為 input dimension 為 2，output dimension 為 5(五種 Rating)，視為一種 Classification。但最

後結果的 Validation Set 準確率只有 33% 左右，不盡理想。因為 ID 實質上並沒有特殊意義，因此準確率會非常低。

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

(collaborator: 無)

分成五類如下

Class 1: Animation, Children's, Comedy, Fantasy

Class 2: Romance, Drama, Musical, Mystery

Class 3: Adventure, Action, Sci-Fi, Film-Noir, Western

Class 4: Crime, Thriller, Horror

Class 5: War, Documentary

