

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

(collaborator:)

MF model, 30Epoch, Optimizer:adagrad, latent dimension:222

有無 Normalized	Kaggle Score(Public+Private)/2
(Rating-平均)/標準差	$(0.93340+0.93113)/2=0.932265$
無	$(0.86398+0.86366)/2=0.86382$

無 Normalized 結果較好，推測是由於 Normalized 的方式是計算所有使用者評分的標準差及平均(組間差)，但若是想消除變異，應該是考慮組內差(單一使用者的標準差及平均)。

2. (1%)比較不同的 latent dimension 的結果。

(collaborator:)

MF model, 30Epoch, Optimizer:adagrad

latent dimension	Kaggle Score(Public+Private)/2
222	$(0.86398+0.86366)/2=0.86382$
666	$(0.87353+0.87239)/2=0.87296$

latent dimension 設定較大，結果不會較理想。

3. (1%)比較有無 bias 的結果。

(collaborator:)

MF model, 30Epoch, Optimizer:adagrad, latent dimension:222

有無 bias	Kaggle Score(Public+Private)/2
有	$(0.86398+0.86366)/2=0.86382$
無	$(0.86519+0.86287)/2=0.86403$

去除 Bias 後，表現略差。

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:)

DNN model, 15Epoch, Optimizer:adam; 較 MF 在較少 Epoch 有較好的表現。

Model	Kaggle Score(Public+Private)/2
DNN	$(0.85677+0.85679)/2=0.85678$
MF(30Epoch)	$(0.86398+0.86366)/2=0.86382$

```

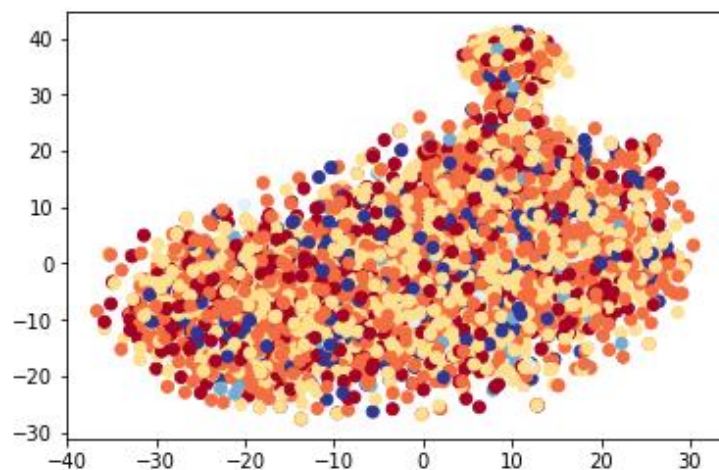
---Training DNN Model---
6040

```

Layer (type)	Output Shape	Param #	Connected to
input_1 (InputLayer)	(None, 5)	0	
input_2 (InputLayer)	(None, 1)	0	
embedding_1 (Embedding)	(None, 5, 444)	2682204	input_1[0][0]
embedding_2 (Embedding)	(None, 1, 444)	1724496	input_2[0][0]
flatten_1 (Flatten)	(None, 2220)	0	embedding_1[0][0]
flatten_2 (Flatten)	(None, 444)	0	embedding_2[0][0]
concatenate_1 (Concatenate)	(None, 2664)	0	flatten_1[0][0] flatten_2[0][0]
dense_1 (Dense)	(None, 512)	1364480	concatenate_1[0][0]
dropout_1 (Dropout)	(None, 512)	0	dense_1[0][0]
dense_2 (Dense)	(None, 256)	131328	dropout_1[0][0]
dropout_2 (Dropout)	(None, 256)	0	dense_2[0][0]
dense_3 (Dense)	(None, 256)	65792	dropout_2[0][0]
dropout_3 (Dropout)	(None, 256)	0	dense_3[0][0]
dense_4 (Dense)	(None, 256)	65792	dropout_3[0][0]
dense_5 (Dense)	(None, 1)	257	dense_4[0][0]
Total params: 6,034,349			
Trainable params: 6,034,349			
Non-trainable params: 0			
Train on 809885 samples, validate on 89988 samples			

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

(collaborator:) 以兩種 type 為一組分類



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:)

DNN model, 15Epoch, Optimizer:adam, 考慮 User 其他資訊效果較好。

Feature	Kaggle Score(Public+Private)/2
DNN(with Features: UserID、MovieID、Gender、Age、Occupation、MovieType)	(0.85677+0.85679)/2=0.85678
DNN(without other features)	(0.88696+0.88777)/2=0.887365