

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

	public	private	notes
no norm	0.85704	0.85823	
norm	3.74791	3.75211	

(標籤 - 平均) / 標準差

輸出 × 標準差 + 平均

明顯退步, 失去了每個user/movie的獨特性

如果mean/std是對user/movie可能會比較好

2. (1%)比較不同的latent dimension的結果。

dimension	public	private	notes
4	0.88810	0.88739	latent dimension太小容易遇到exploding gradient
16	0.85704	0.85823	
64	0.91032	0.91027	試了比較多模型latent開很大沒什麼用

latent dimension 16是我找到最好的中間值

3. (1%)比較有無bias的結果。

	public	private	notes
Bias	0.85704	0.85823	
No Bias	0.88317	0.88200	

沒有bias稍微退步了一些

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。

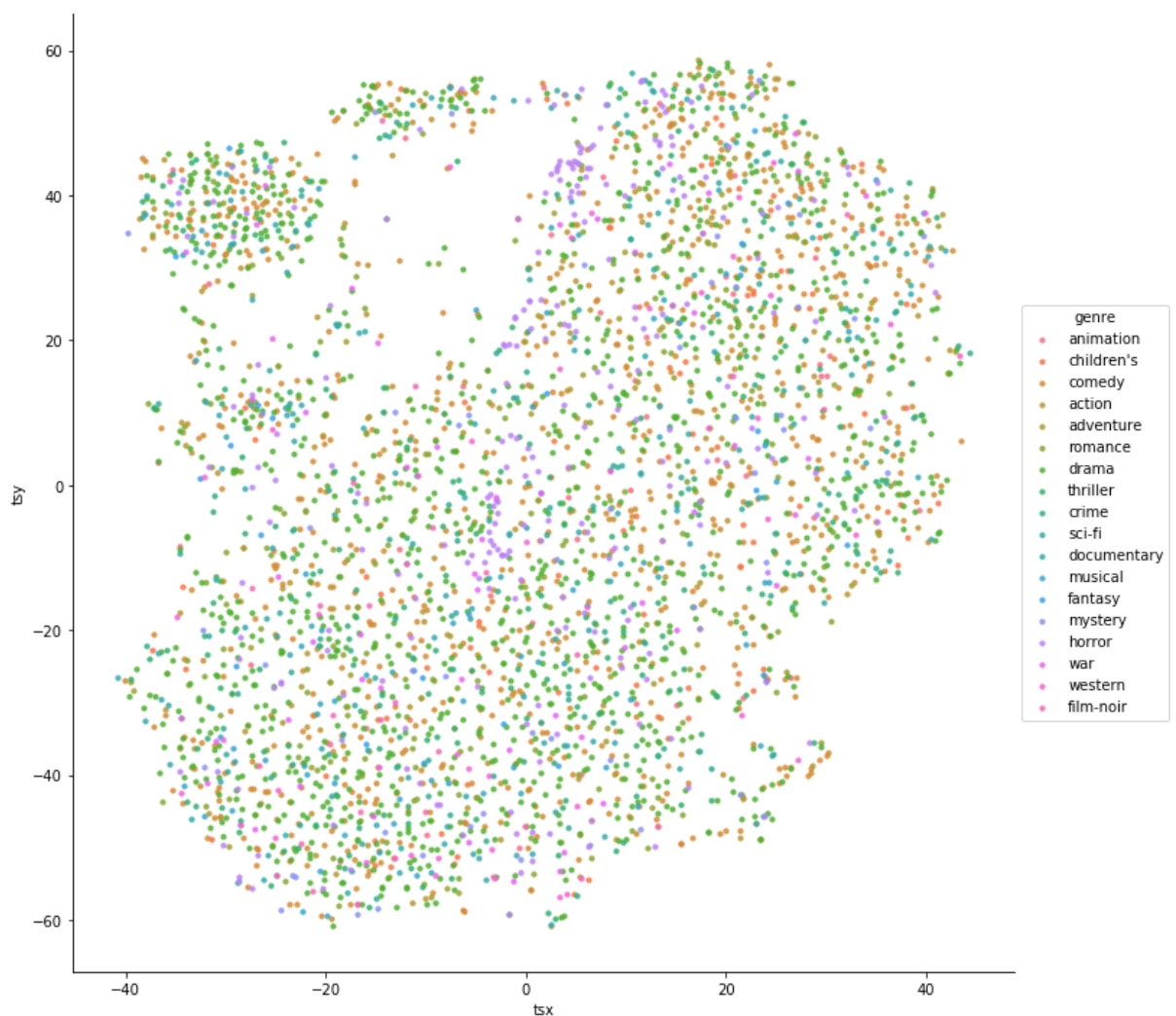
	public	private	notes
MF	0.85704	0.85823	
DNN	0.85417	0.85452	

輸入跟 mf 一樣：movie -> embedding(16), user-> movie(16), age, occupation, category都是one hot

dnn 用了1024, 32, 1 的dense layer, relu +0.5 dropout

結果比mf好一點點

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



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。