HW5 Report

學號：R05945012 系級：生醫電資碩二 姓名：張凱崴

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

Method (dim=128)	No normalization	Standardization	Feature scaling
Valid loss	0.8594	0.8538	0.8690
Public score	0.86063	0.85871	0.86876
Private score	0.86147	0.85920	0.87069

我使用了兩種 normalize 的方法，standardization 指的是把 rating 減平均並除以標準差，feature scaling 則是把 rating 線性轉換成[0,1]之間。根據上表的結果，standardization 的方式最好，而 feature scaling 甚至比沒 normalize 還要差。

2. (1%)比較不同的 latent dimension 的結果。

Dimension	32	64	128	256	512
Valid loss	0.8584	0.8575	0.8538	0.8557	0.8578
Public score	0.86145	0.85961	0.85871	0.85959	0.85943
Private score	0.86170	0.85910	0.85920	0.85847	0.86042

接續上題，使用 standardization，並改變不同的 latent dimension 做實驗。根據上表的結果，dimension 並不是越大越好，設為 128 有最好的結果。

3. (1%)比較有無 bias 的結果。

Model	With bias	No bias
Valid loss	0.8538	0.8624
Public score	0.85871	0.85840
Private score	0.85920	0.86177

比較有無加上 bias 的模型，有 bias 的結果比較好，非常合理，因為評分者可能有自己評分的偏好。

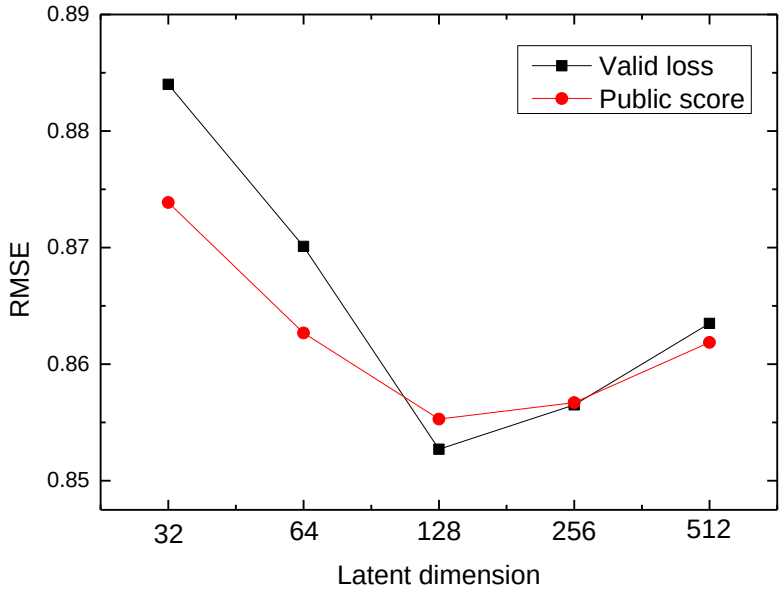
4. (1%)請試著用 DNN 來解決這個問題，並且說明實做的方法(方法不限)。並比較 MF 和 NN 的結果，討論結果的差異。

Model	MF	NN
Valid loss	0.8538	0.8508
Public score	0.85871	0.85493
Private score	0.85920	0.85577

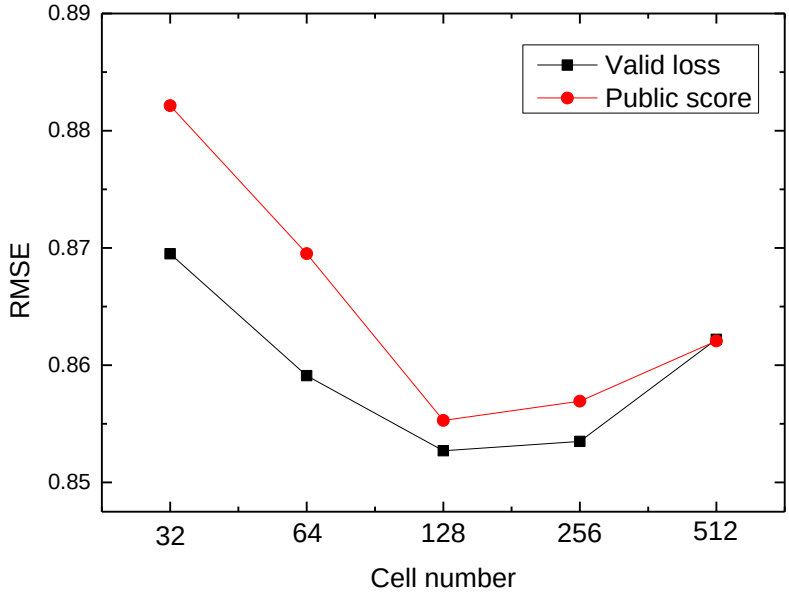
我將 movie 和 user 的 embedding layer (dim=128)做 concatenate 後，接上 dense layer 以構成 NN 的模型。NN 部分為一層 128 cell 的 hidden layer 以及一層 output，activation 分別為 LeakyReLU 和 linear。調整至最佳參數後，比 MF 還好一些。將兩個 model 做 ensemble 後，可得到更好的分數 (public score: 0.84648)。

在訓練 NN 的過程，我改變 embedding layer 的 dimension 以及 hidden layer 的 cell 個數，紀錄及作圖如下：

Dimension	32	64	128	256	512
Valid loss	0.8840	0.8701	0.8508	0.8565	0.8635
Public score	0.87387	0.86268	0.85493	0.85670	0.86186
Private score	0.87288	0.86337	0.85577	0.85841	0.86365



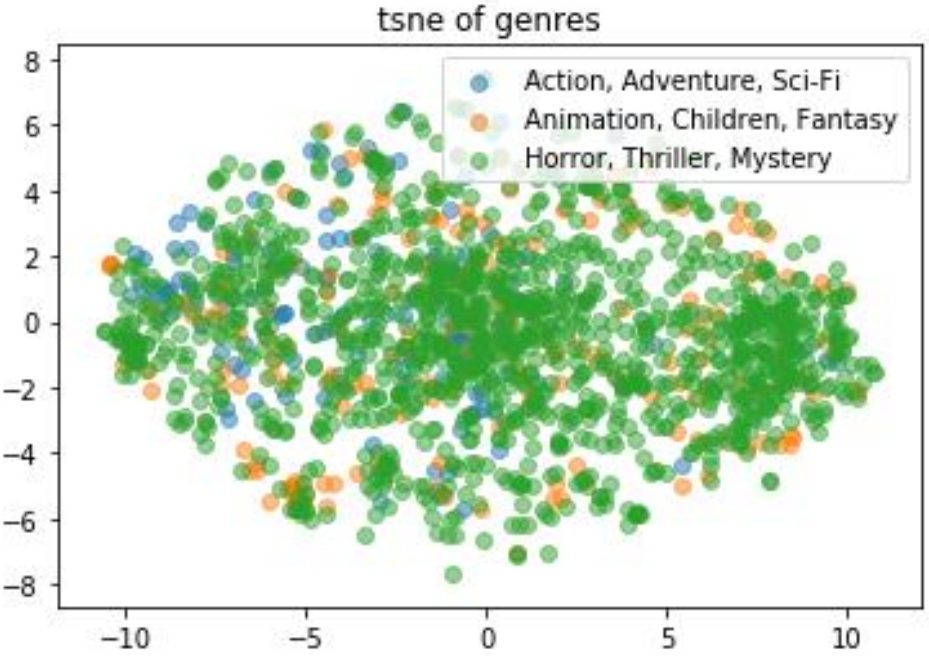
Cell	32	64	128	256	512
Valid loss	0.8695	0.8591	0.8508	0.8535	0.8622
Public score	0.88214	0.86952	0.85493	0.85693	0.86207
Private score	0.88133	0.86878	0.85577	0.85677	0.86187



因此選擇 embedding layer dimension 為 128，hidden layer cell 個數也為 128。

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

我將相似類型的電影分為同一個組別，例如 action、adventure 和 sci-fi 會被分為同一個組別。以 tsne 降維後做圖如下：



雖然沒有很明顯地分群，但可看出藍色(action, adventure, sci-fi)的點大部分落在圖的左方，而綠色(horror, thriller, mystery)在正中間(0,0)及右邊(7.5,0)最密集。

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。

我將 user 和 movie 的額外資訊預先對不同 ID 做好 embedding，並將其作為新的 embedding layers 的初始 weights，再跟原本 dim=128 的 embedding layers 做 concatenate 後，連接 NN 作為新的模型。

Embedding 的方式以 userID:796 與 movieID:1 為例，如下：
796::F::1::10::48067 ➔ [0,1,1,10] (M, F, age, occupation)
1::Toy Story (1995)::Animation|Children's|Comedy ➔ [1995, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0] (year, Action, Adventure, Animation, Children, Comedy, Crime, Documentary, Drama, Fantasy, Film-Noir, Horror, Musical, Mystery, Romance, Sci-Fi, Thriller, War, Western)。以上的 embedding 都會再對各自的項目進行標準化，以防止數值差異過大。

訓練出來的結果相較於沒有加額外資訊的模型有所進步。如下表：

Model	Only rating	With other features
Valid loss	0.8508	0.8494
Public score	0.85493	0.85017
Private score	0.85577	0.85094

最後再將較好的模型與 MF 做 ensemble，可得到最佳分數 (public score: 0.84435)