學號：R05922080 系級： 資工碩二 姓名：王鵬傑

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

	Public	Private
With Normalize	0.87368	0.86649
Without Normalize	0.87001	0.86431

Normalize 的方法是直接將 training dataset 的 rating 做 Normalize，在 testing 時再將 predict 出來的值乘上 std + mean。

兩者分數的結果沒有差很多，反而是做了 Normalize 的結果較差一點。

2. (1 %)比較不同的 embedding dimension 的結果。

Embedding dim	Public	Private
16	0.87001	0.86431
32	0.87644	0.87084
64	0.88243	0.87255
128	0.89386	0.90291
256	0.92505	0.92739

Embedding 的 Dimension 越大反而效果越差。

3. (1 %)比較有無 bias 的結果。

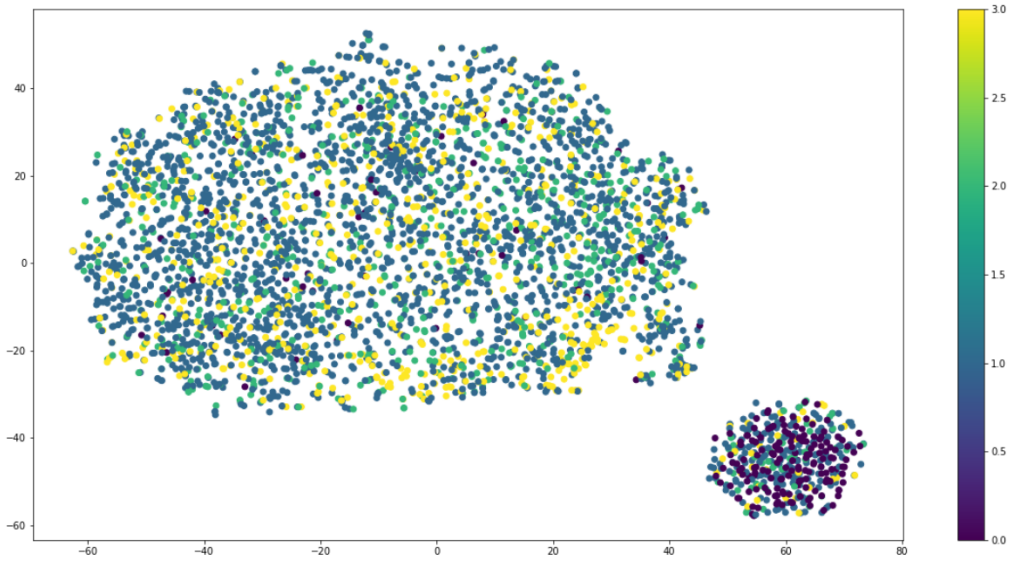
	Public	Private
With Bias	0.87238	0.86375
Without Bias	0.87001	0.86431

沒有 Bias 的時候效果比較好，有可能是因為太多的參數要 Learn 反而會造成 Overfitting 等等的問題。

4. (1 %)請試著將 movie 的 embedding 用 tsne 降維後，將 movie category 當作 label 來作圖。

將 Label 分成三類：

Category	Label
0	No label
1	Comedy, Fantasy, Romance, Drama, Musical, Sci-Fi, Animation, Children's
2	Action, Documentary, Western
3	Thriller, Horror, Crime, Mystery, War, Adventure, Film-Noir



可以看出沒有被 label 的自成一群在右下角，而其他類別分佈在左半邊。

5. (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。

將 users.csv 的性別和年齡取出來，和原本的 train.csv 做 combine，變成

```
array([[560292,   1121,   1214, ...,      1,     25,     15],
       [892886,   4822,   3196, ...,      0,     25,      0],
       [604781,   2401,   2456, ...,      1,     18,      3],
       ...,
       [210756,   2417,   1036, ...,      1,     35,     17],
       [ 56089,   2708,    574, ...,      1,     25,     17],
       [824841,   3512,    858, ...,      0,     25,     20]])
```

性別: s_embedding = Embedding(4096, 16)(s_input)

年齡: a_embedding = Embedding(8192, 16)(a_input)

在個別做 Flatten

最後將兩個做 Dot(axis = 1)(a_embedding, s_embedding)

Kaggle 分數為

	Public	Private
No other feature	0.87001	0.86431
With other feature	0.87829	0.86834

分數比起來反而較差，可能這兩個 Feature 反而是雜訊，或者是 Overfitting 的問題。