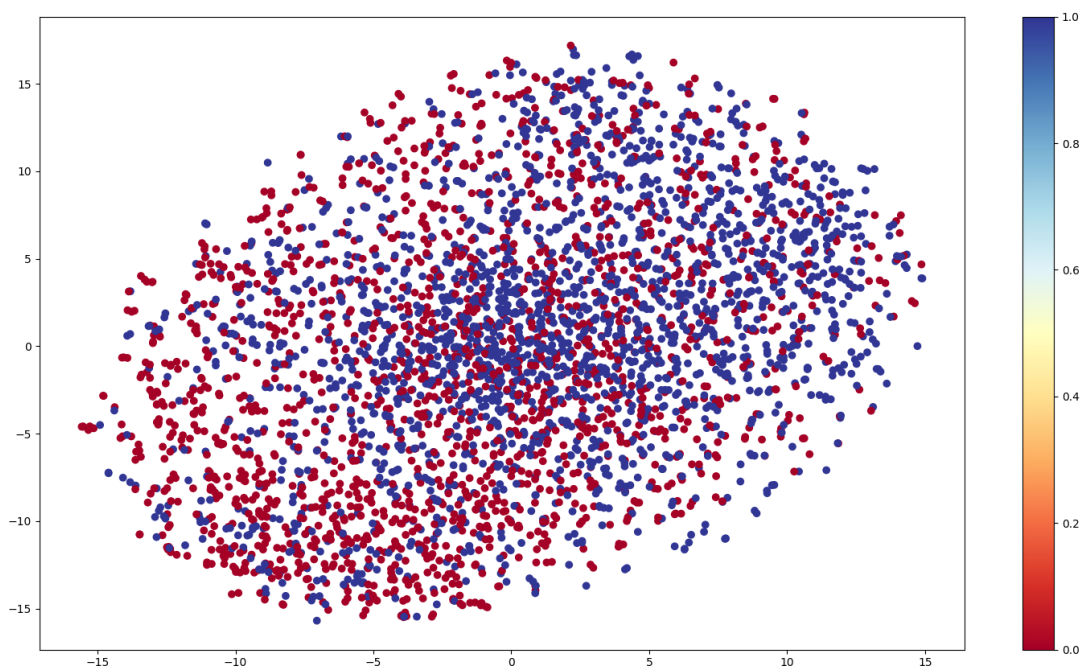


1. (1%)請比較有無 `normalize(rating)` 的差別。並說明如何 `normalize`。  
將 `training data` 標準化後拿去學習，之後預測的結果再用 `training data` 原來的平均值跟標準差還原回來。 $(\text{result} = \text{pred} * \text{training\_std} + \text{training\_mean})$   
結果差異：  
沒標準化: 0.89257(Kaggle public score)  
標準化: 0.90940(Kaggle public score)
2. (1%)比較不同的 `latent dimension` 的結果。
3. (1%)比較有無 `bias` 的結果。  
有 `bias`: 0.89257(Kaggle public score)  
沒有 `bias`: 0.88089(Kaggle public score)  
看起來 `bias` 的傾向並沒有很明顯
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。  
將 `movie embedding` 跟 `user embedding` 後 `concatenate` 在一起後再通過一個 DNN，`output` 則是將他視為 `regression` 問題，用 `relu` 選出一個數字作為預測的分數。  
結果差異：  
MF: 0.89257(Kaggle public score)  
DNN: 0.86832(Kaggle public score)
5. (1%)請試著將 `movie` 的 `embedding` 用 `tsne` 降維後，將 `movie category` 當作 `label` 來作圖。



Red = {Action, Animation, Children's, Comedy, Fantasy, Musical, Romance, War, Western}

Blue = {Adventure, Crime, Documentary, Drama, Film-Noir, Horror, Mystery, Sci-Fi, Thriller'}

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。