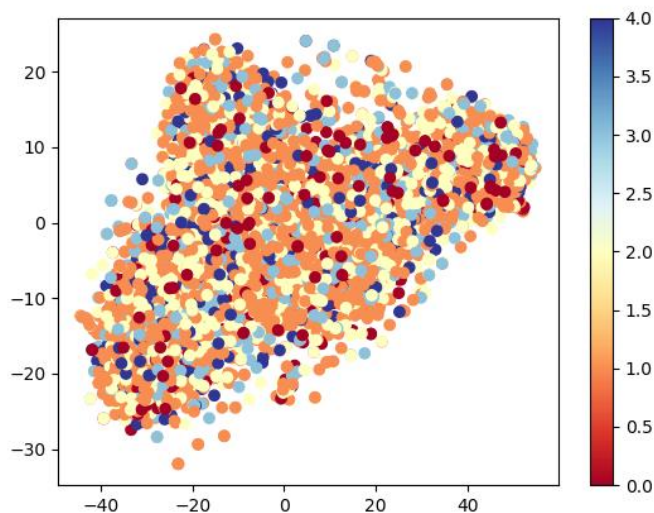


數據皆以 public 為準

1. (1%)請比較有無 normalize(rating)的差別。並說明如何 normalize.
$$y' = (y - \text{mean}) / (\text{standard deviation})$$

有 normalize: 0.86500
無 normalize: 0.88216
有 normalize 好許多
2. (1%)比較不同的 latent dimension 的結果。
皆有 normalize
128: 0.86500
256: 0.87263
128 優於 256
3. (1%)比較有無 bias 的結果。
有 bias: 0.86669
無 bias: 0.86500
兩者差異不大
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。
把經過 Embedding layer 的 users 和 movies 接成一個 vector，再通過 3 層 Dense。
DNN: 0.92730
MF:0.86500
MF 勝過簡單的 DNN 許多，應該是已經 overfit 了
5. (1%)請試著將 movie 的 embedding 用 tsne 降維後，將 movie category 當作 label 來作圖。



animation children's fantasy 一類，adventure drama action 一類，comedy romance 一類，crime thriller horror 一類，剩下的一類，看起來完全沒有分開，應該是 movie embedding 不適合用 tsne 降維，或是我自己的分類不太好導致結果混在一起。

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。

Movie 總共有 17 類，所以把 movie 變成 17 維的 vector，如果是該類就=1 反之=0，User 有 7 種年紀和 21 種工作，所以把 User 分成 28 維，如果是該類就=1 反之=0，然後接起來通過 3 層 Dense，誤差為 1.06877，感覺不太好。