學號：R05921077 系級： 電機碩一 姓名：陳立杰

1. (1%)請比較有無normalize(rating)的差別。並說明如何normalize.  
    我 是先做rate[i]=(rate[i]-mean)/std後拿來做training，然後predict出一個值後再用rate[i]=rate[i]\*std+mean來還原他的值，我發現normalize後training的速度會變快，原本大概要train 2X個epoch，normalize後只需1X個epoch，但是準確度會稍微下降， 原本是0.85716，normalize後為0.86223。
2. (1%)比較不同的latent dimension的結果。  
   假設k為latent dimension，k=2, 4, 8, 110, 120, 130時在public set上的分數分別為0.89915, 0.86864, 0.86361, 0.85819, 0.85716, 0.85729，從這些data可以看到比較大的k值的準確率會比較高，但是太大的時候準確度還是會稍微下降。
3. (1%)比較有無bias的結果。

當latent dimension= 120時，沒有加bias的準確度為0.85716，加上一些bias後準確度可以提升到0.85414，也發現加入age這個bias之後的進步最明顯。

1. (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。  
   我將user embedding以及movie embedding concatenate在一起再過DNN得出rating，DNN中有兩個hidden layer，每個layer有128個neural且dropout 0.1，output regression，當k=8時準確率為NN的**0.86285，而MF的準確度為0.86361，稍微比MF好一些，但NN相較之下需要train非常久，大概需要train到100個epoch，而MF大概只要2X個epoch就能train完了，所以如果k太大時NN就會非常難train(k=**latent dimension**)** ，應該是因為NN的參數量較多而且有drop才導致必須train那麼久。
2. (1%)請試著將movie的embedding用tsne降維後，將movie category當作label來作圖。

Comedy 0

Fantasy 50

Action 100

Romance 150

Sci-Fi 200

Documentary 250

War 300

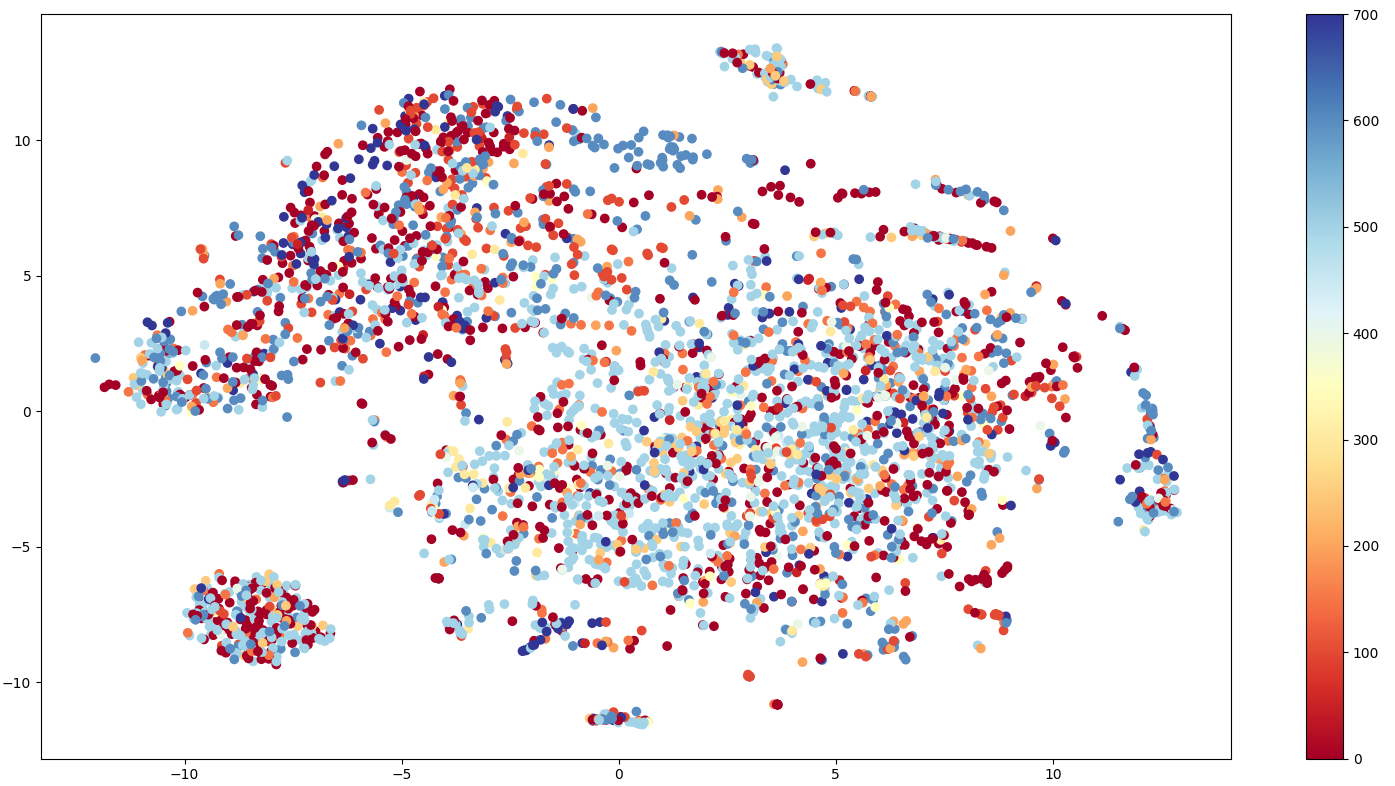
Mystery 350

Western 400

Film-Noir 450

Drama’,’Musical 500

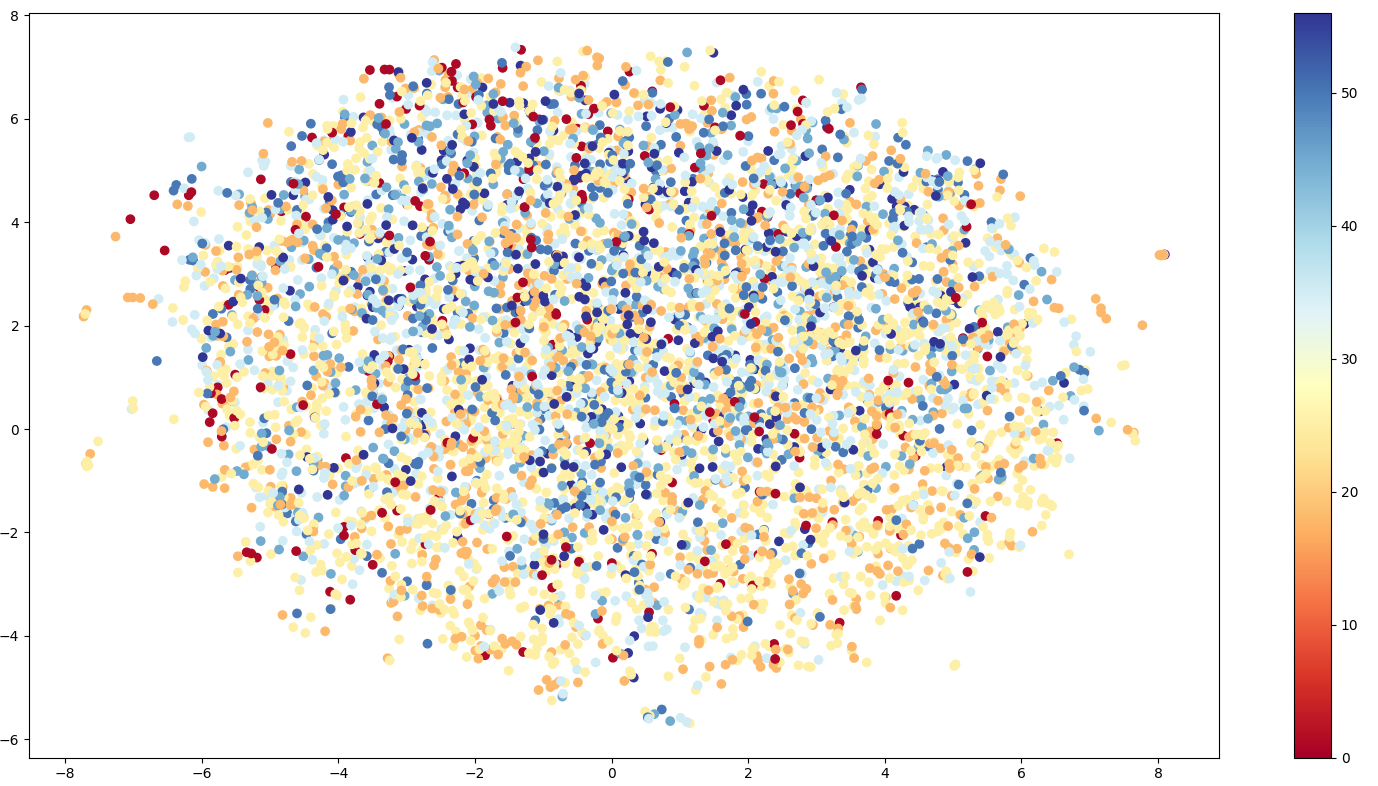
Thriller',’Horror’,’Crime 600

Adventure,Animation,Children's 700  


從這圖中可看到中間有很大的兩堆，上面那堆比較多喜劇、動作、恐怖片等，下面那堆則是比較多drama所以感覺很雜亂，因為drama很多不同的種類，而最上下分別有兩個很獨立的小堆，兩個小堆都有很多的drama，下面那堆感覺比較偏喜劇、愛情的drama，上面那堆則是比較偏神秘或戰爭類的drama。

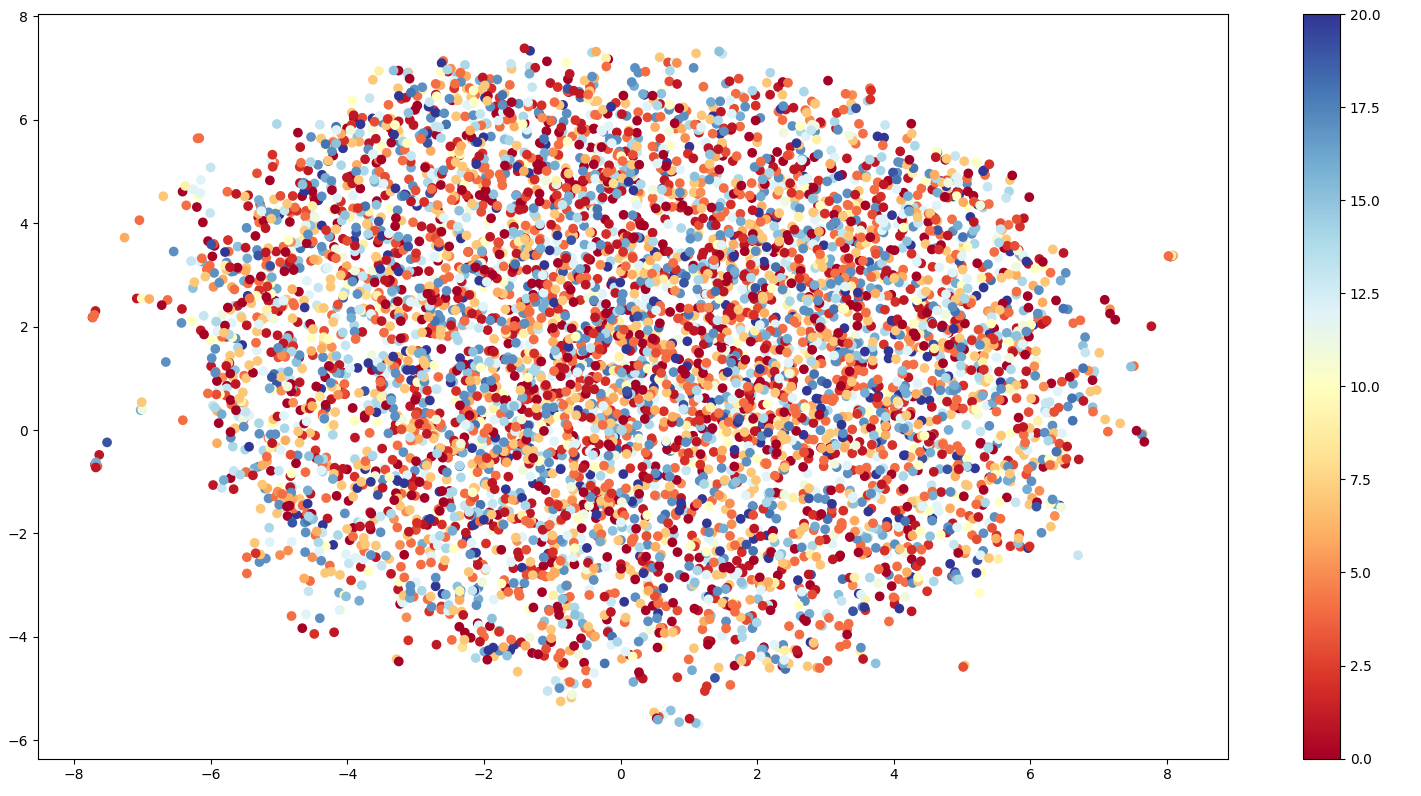
1. (BONUS)(1%)試著使用除了rating以外的feature, 並說明你的作法和結果，結果好壞不會影響評分。

我把users的age跟occupation拿來當作額外的feature跟tags，拿users的embedding用tsne降維作圖，下圖是age的結果:



很明顯的可以看到年紀較大也就是顏色較深的部分幾乎都集中在上半部，而年紀較小也就是較淺的點全部集中在下半部，代表年齡的確是有一些區分的效果，在做bias時確實也是加入age影響最大。

下圖是用occupation作圖的結果:



因為種類太多又不好分群，所以做出來的結果看不出什麼意義。