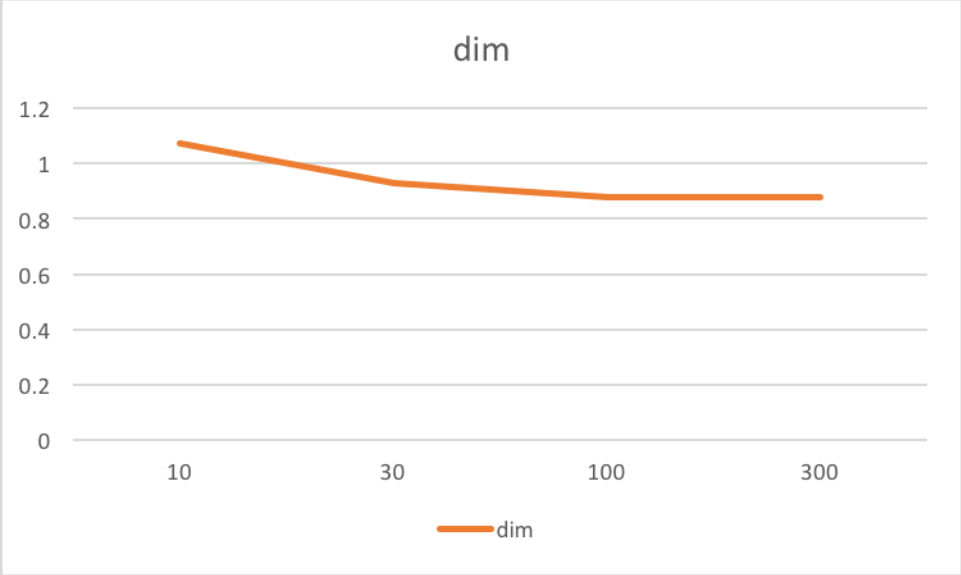
學號：B04902051 系級： 資工二 姓名：林承豫

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

   對 rating 取出後做 normalize，及減掉 mean（3.58）之後除上 variance(1.24)
   固定訓練 50epoch，發現在 kaggle 上未經過 normalized 的 loss 為 0.86373，經過 normalize 後的結果為 0.85643，結果稍微好一點。

2. (1%)比較不同的 latent dimension 的結果。
   固定訓練 30epoch，並切出 0.1 data 作為 validation set，作為觀察不同 latent dimension 的表現。下圖為不同 dimension 下的結果，分別從 10 到 300 共實驗四種不同 dimension，可以發現 dim 越大在 loss 表現上越好，但基本上大於 100 之後並沒有顯著的差異。



3. (1%)比較有無 bias 的結果。
   實驗方式為透過 MF model 在同樣模型下訓練 30epoch，並切出 0.1 的 training data 作為 validation data

   no bias:0.88014
   with bias:0.86213

   發現增加了 bias 之後，在 loss 的表現上會更好，在 training 的過程下降的較快，原因可能是 MF 的參數較多，增加 bias 可以讓模型參數更符合的正確的結果。但發現在上傳 kaggle 之後，結果差不多，但明顯可以讓 training 的過程加快。

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。

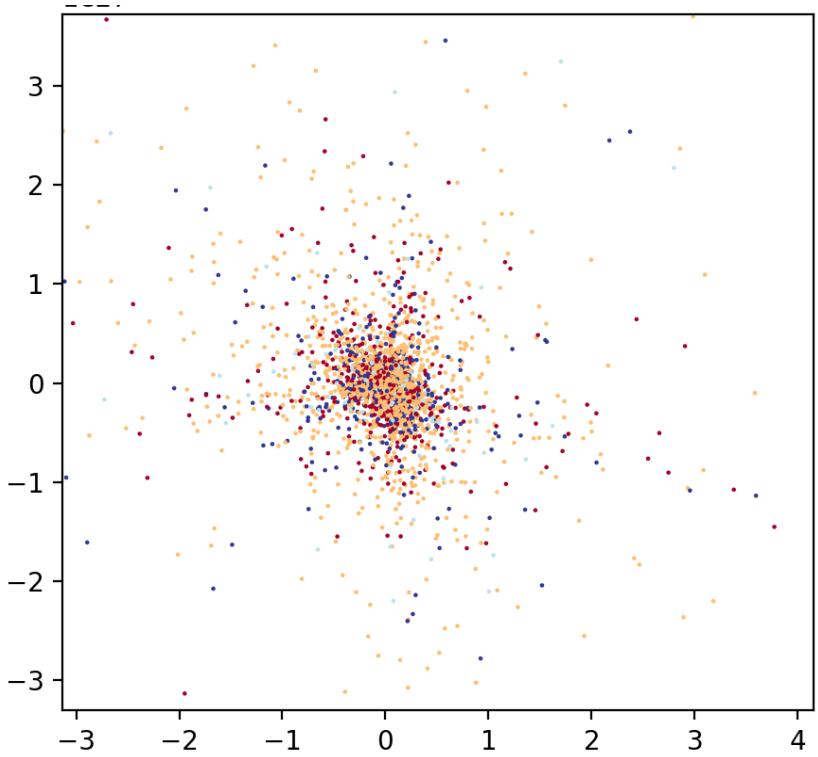
   實作方法為 embedding 轉成 100dim 之後 加入一層 dense(2048)，後再加入 dense(1) 輸出

   MF: 0.86625

   NN: 0.859998

   發現 NN 的結果稍微好一點，但基本上差距不太大，也曾試過增加 dense 的 layer 但發現表現並不會變好，反而是增加 dense 的參數會讓表現變好，但加到 2048 之後，即使參數更多 performance 也不會變好。

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



   圖中黃色的點 Drama,Romance,Musical,Comedy

   藍色的點為 Adventure,Action

   淡藍色的點為 Thriller,Crime

   紅色的點為其他類別

   結果發現用 movie category 來作圖，並沒辦法看出 model 的 embedding 能有把 movie 的 category 做 classify 的效果，感覺不同 category 都混雜在一起。

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。

作法為原本的 embedding model，但 bias 改為加入 user 的 age，即對每一筆 data 都多加一欄為 user age，並且在設計 model 時加為 bias
在 trainnig 過程中一樣乘第一題切出 0.1 的 data 作為 validation data，訓練 30epoch。
發現這樣的結果會變差

結果為 0.9038

原本的結果為 0.88014