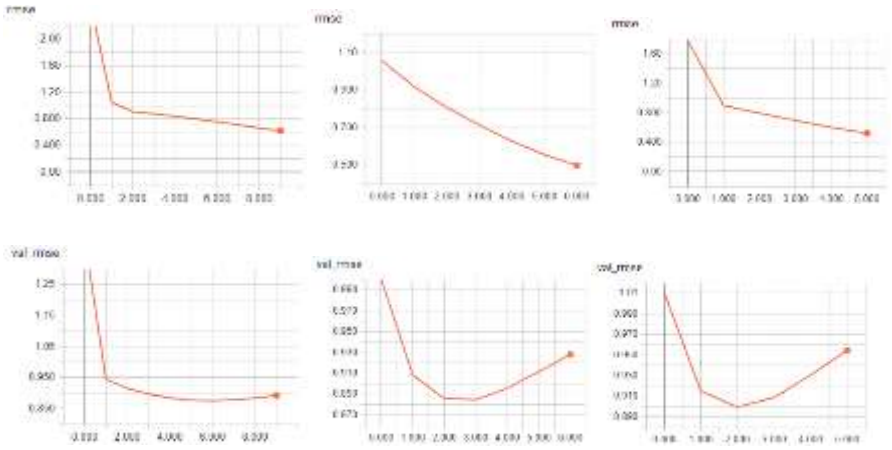
Machine Learning HW6

R06725035 資管所碩一 陳廷易

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

Normalize 方法嘗試了兩種做法，第一種為 Min-max normalization，也就是將各 rating 值減去 rating 最小值後再除以 rating 全距；而 predict 出結果時再乘回全距並加上最小值。第二種作法為 zero-mean normalization，也就是將各 rating 值減去 rating 平均值後再除以 rating 標準差進行標準化；而 predict 出結果時要再乘回標準差並加上平均值。

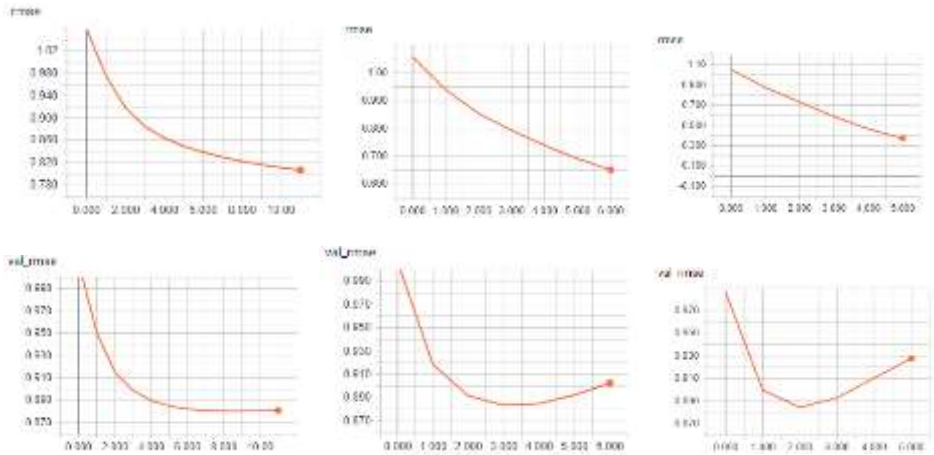
訓練過程如下所示，未 normalize(圖左)、zero-mean normalized(圖中)、min-max normalized(圖右)；



可發現此 task 很容易 overfitting，其中以 zero-mean normalized 的表現最好且收斂最快。未進行 normalize 的雖訓練時間較長，但事實上最終表現仍會有相當不錯的表現。*上傳至 kaggle 的成績依序為 0.87488、0.87311、0.88387。*

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

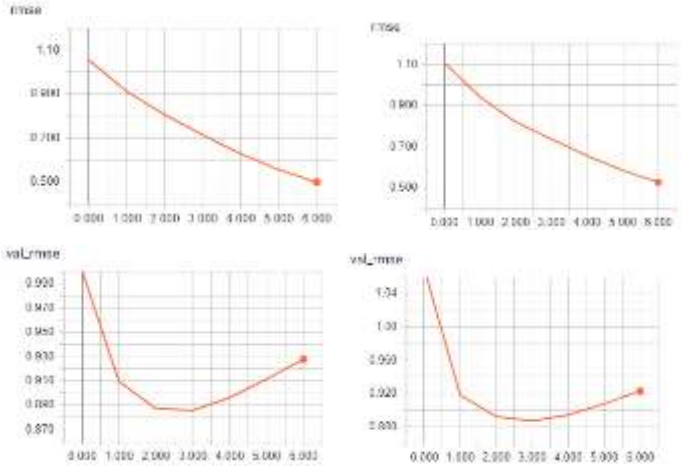
在預設參數下，比較 latent dimension 在 8(左)、64(中)、256(右)的結果：



根據比較結果可看發現，laten dimension 越大，訓練時的收斂速度越快，但也越容易發生 overfitting 問題，且表現通常不會比較好，在 laten_dim=8 的情況下是表現比較穩定的。

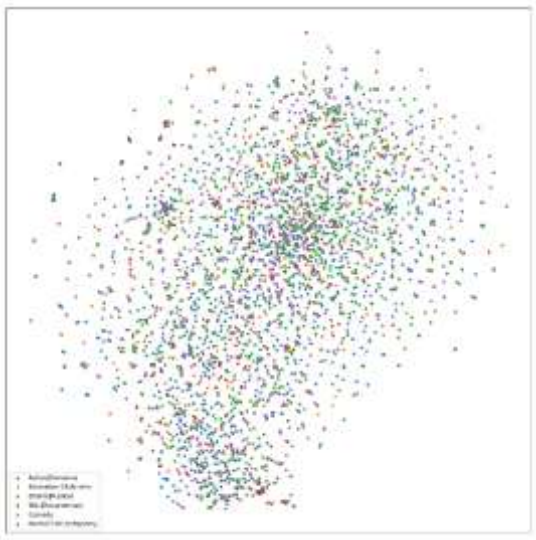
3. (1 %)比較有無 bias 的結果。

左圖為有 bias 的訓練成果，右圖為無 bias 的訓練成果：



從比較結果中可看出有增加 bias 的效果不論在 training 或 validation 表現都略好一點，雖沒有明顯的進步，但或許是因為每個 user 與 movie 仍存在不同的 rating 傾向，因此能略為提升效果。*上傳至 kaggle 的分數分別為 0.88148、0.88345。*

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

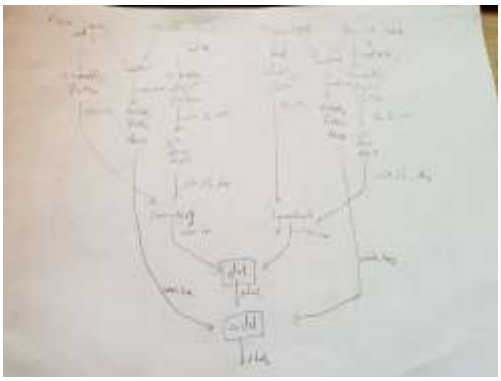


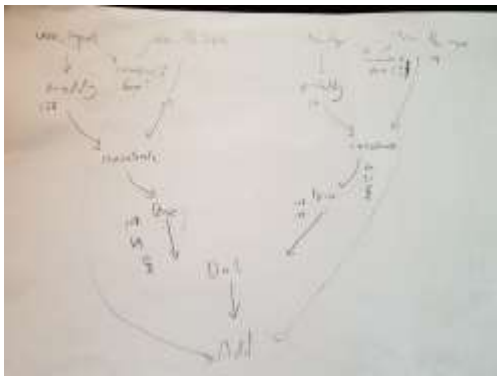
從左圖中可看出綠色(音樂戲劇類)、藍色(動作浪漫)與紫色(喜劇)，除了在中間區域有集中的現象，幾乎散布於整張圖中較無規律性，或許因為多數電影皆有相關元素有關。而橘色(兒童卡通)則分布於偏中上方的部分。咖啡色(恐怖懸疑驚悚)主要位於下方，有較明顯的一群，可見其元素特徵較為獨特。

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。

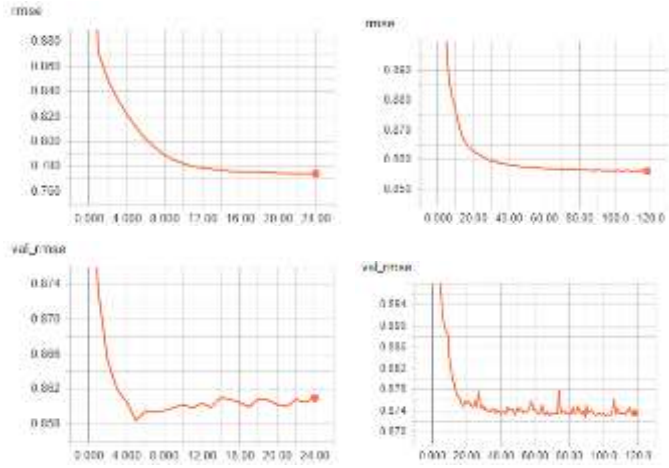
在 best model 中，仍是圍繞以 MF 為中心想法，差別在於將額外 feature (如:電影種類、職業別、年齡等)透過 one-hot encoding 方式分別 append 於 movie ID embedding 後面與 user ID embedding 後面，再利用 DNN

至相同維度進行 DOT，再 ADD 上 movie bias 與 user bias。以此作法共設計了兩種 model，架構如下：





訓練結果如下：



最後上傳至 kaggle 的成績為 0.85106/0.85816 、0.87311/0.87386

※若未提及之參數皆預設如下：

- patient=3, max_epoch=350
- latent_dim = 128
- batchSize=1024
- bias = True
- normalize=zero-mean normalization
- optimizer=Adam()
- loss='mse'

※Reference：

- 歷屆助教 sample code
- 小老師手把手教學
- 其他網路上分享之教學文章