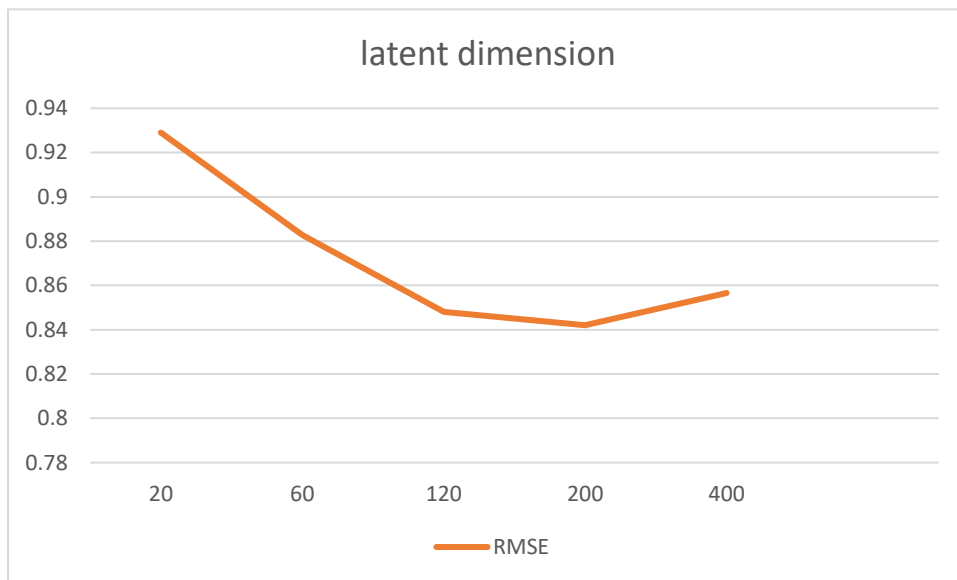


1. (1%)請比較有無 `normalize(rating)` 的差別。並說明如何 `normalize`。

有對 `rating` 做 `normalize` 的情況下，所輸出出來的結果會比較差(`normalize` 的做法是對 `train data` 的 `rating` 減平均後除標準差然後對輸出的 `rating` 乘上原本的標準差後加平均)，而我認為原因可能是因為 `normalize` 之後數值的差異變小了，而且造成了一些縮放誤差導致最後出來的結果沒有比較好。

2. (1%)比較不同的 `latent dimension` 的結果。



我們從上圖可以發現，當 `latent dimension` 的大小在 200 附近的時候，效果最好。

3. (1%)比較有無 `bias` 的結果。

我發現不管在有沒有處理 `bias` 的情況下，所做出的結果都差不多，結果(RMSE)會在 0.84 附近。我覺得有可能的原因是因為我有設定 `Droupout` 導致 `bias` 的效果沒有比較明顯的影響。

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。

我發現直接使用 `DNN` 所做出的結果(0.99)並沒有比用 `MF` 所做出來的結果(0.84)好，而我 `DNN` 的做法是用 `Concatenate` 把兩個 `user` 和 `movie` 兩個 `vector` 連起來後，在丟入兩層 `DNN` 中。

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



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。