

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

(collaborator:)

沒經過 `normalize` 的分數：0.84706

有經過 `normalize` 的分數：0.84260

可見有 `normalize`，分數較佳。

$$\min_{\substack{x^{(1)}, \dots, x^{(n_m)} \\ \theta^{(1)}, \dots, \theta^{(n_u)}}} \frac{1}{2} \sum_{(i,j):r(i,j)=1} ((\theta^{(j)})^T x^{(i)} - y^{(i,j)})^2 + \frac{\lambda}{2} \sum_{i=1}^{n_m} \sum_{k=1}^n (x_k^{(i)})^2 + \frac{\lambda}{2} \sum_{j=1}^{n_u} \sum_{k=1}^n (\theta_k^{(j)})^2$$

以上為 MF 要 minimize 的 object。θ 是 user latent，x 是 movie latent，y 是 rating。

2. (1%)比較不同的 latent dimension 的結果。

(collaborator:)

我用的 latent dimension 為 120，kaggle 分數為 0.84260。

latent 60 分數則為 0.84870

3. (1%)比較有無 bias 的結果。

(collaborator:)

有：0.84260、無：0.84541

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。

(collaborator:)

DNN model：

```
user_id_input = Input(shape=(1, ), name="UserID")
movie_id_input = Input(shape=(1, ), name="MovieID")

user_latent = Embedding(
    num_users,
    latent_dimension,
    input_length=1,
    embeddings_initializer="random_uniform",
    embeddings_regularizer=l2(lamda) if is_regularized else None,
    name="UserLatent")(user_id_input)
movie_latent = Embedding(
    num_movies,
    latent_dimension,
    input_length=1,
    embeddings_initializer="random_uniform",
    embeddings_regularizer=l2(lamda) if is_regularized else None,
    name="MovieLatent")(movie_id_input)

user_latent = Flatten(name="FlattenedUserLatent")(user_latent)
movie_latent = Flatten(name="FlattenedMovieLatent")(movie_latent)

features = Concatenate(
    name="ConcatenatedUserMovieLatent")([user_latent, movie_latent])
x = Dense(64, activation="elu", name="Dense-1")(features)
x = Dense(128, activation="elu", name="Dense-2")(x)
x = Dense(256, activation="elu", name="Dense-3")(x)

output = Dense(1, activation="elu", name="Rating")(x)

model = Model([user_id_input, movie_id_input], output)

model.compile(optimizer="adam", loss="mse", metrics=["mse"])
```

DNN 的分數相較 MF 下降了 2% 左右，根據助教所提示的，可能是參數還需要再調整，否則 DNN 應該有機會超越 MF。

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