Machine Learning HW6

TAs
ntu.mlta@gmail.com

Outline

1. Task Introduction
2. Kaggle
3. Deadline and Policy
4. FAQ

Task Introduction

Matrix Factorization

Task Introduction

- Given the user's rating history on items, we want to predict the rating of unseen (user,item) pairs.
- We want you to implement matrix factorization to predict the missing value on user-item matrix.

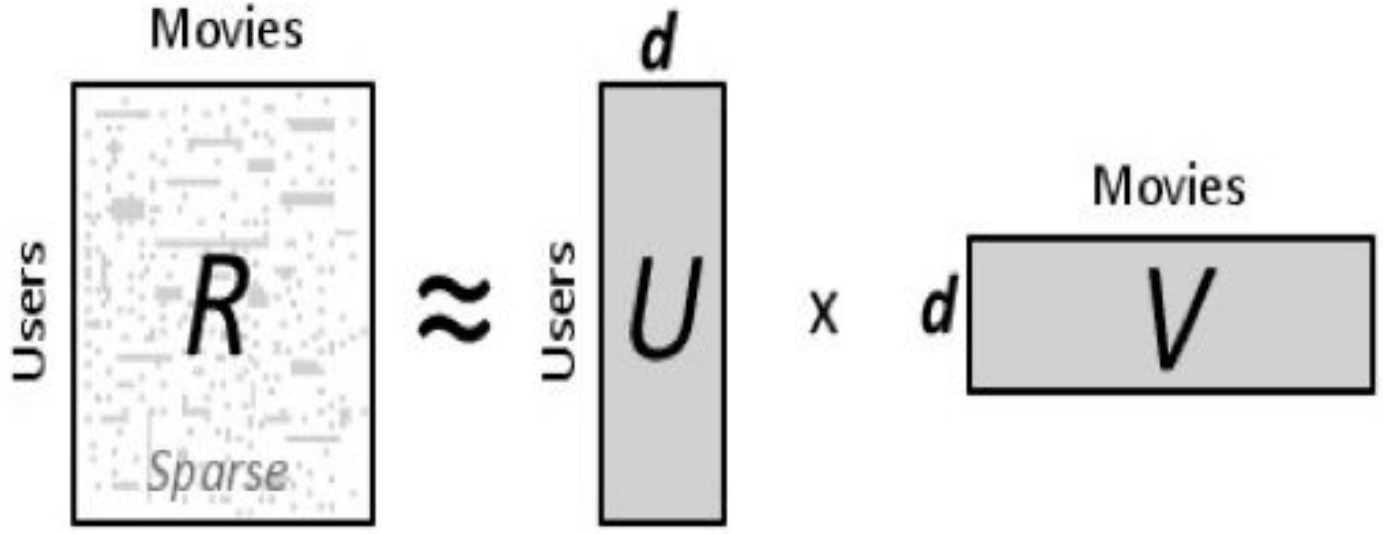
Matrix Factorization(1/4)

- Predict missing values.

	涼宮春日的憂鬱	4月是你的謊言	科學超電磁砲
大木博士	5	N/A	4
小智	N/A	3	N/A
小茂	2	N/A	2
吸盤魔偶	4	2	N/A

Matrix Factorization(2/4)

$$R \approx \hat{R} = U \cdot V^T$$



Matrix Factorization(3/4)

- Minimize loss function by gradient descent.

$$L = \sum_{i,j} (R_{ij} - U_i \cdot V_j)^2$$

	涼宮春日的憂鬱	4月是你的謊言	科學超電磁砲
大木博士	4.7	2.7	3.9
小智	3.2	3.5	3.7
小茂	1.9	2.5	2.2
吸盤魔偶	4.1	1.8	1.2

Matrix Factorization(4/4)

- Bias term

$$r_{i,j} = U_i \cdot V_j + b_i^{user} + b_j^{movie}$$

Useful function in Keras

1. keras.layers.Embedding : the user matrix and item matrix can be viewed as two embedding matrix
2. keras.layers.Flatten: the output tensor shape of embedding layer would be [batch_size,1,embedding_dim], you need this function to reshape the tensor to [batch_size,embedding_dim]

Useful function in Keras

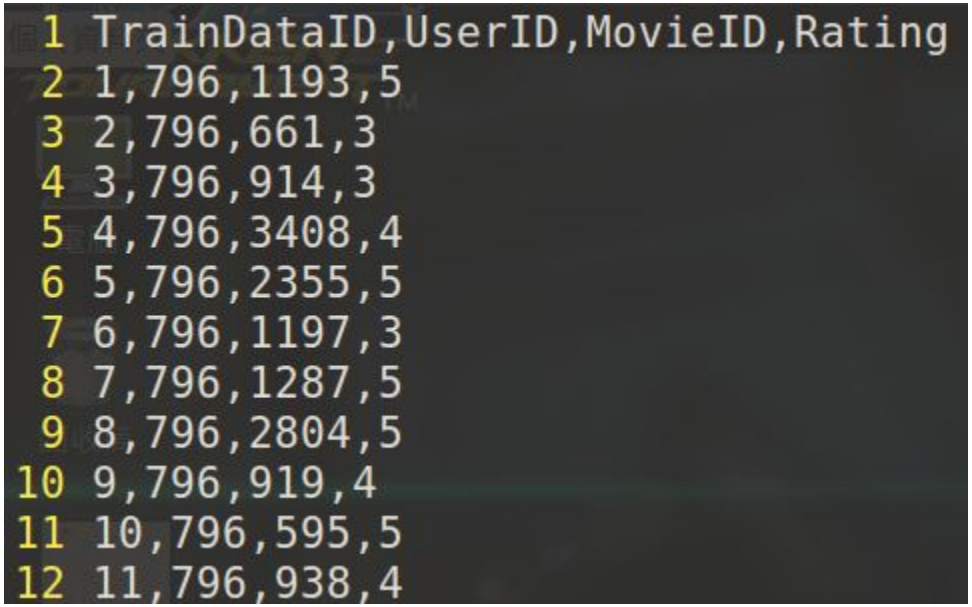
1. keras.layers.Dot : if applied to two tensors a and b of shape (batch_size, n), the output will be a tensor of shape (batch_size, 1) where each entry i will be the dot product between a[i] and b[i].
2. keras.layers.Add : add all tensors
3. keras.layers.Concatenate : concatenate two tensor

Example code:

```
import numpy as np
import keras
a = np.ones([2,5])
b = np.zeros([2,5])
input_a = keras.layers.Input(shape=[5])
input_b = keras.layers.Input(shape=[5])
out = keras.layers.Add()([input_a,input_b])
model = keras.models.Model([input_a,input_b],out)
print(model.predict([a,b]))
```

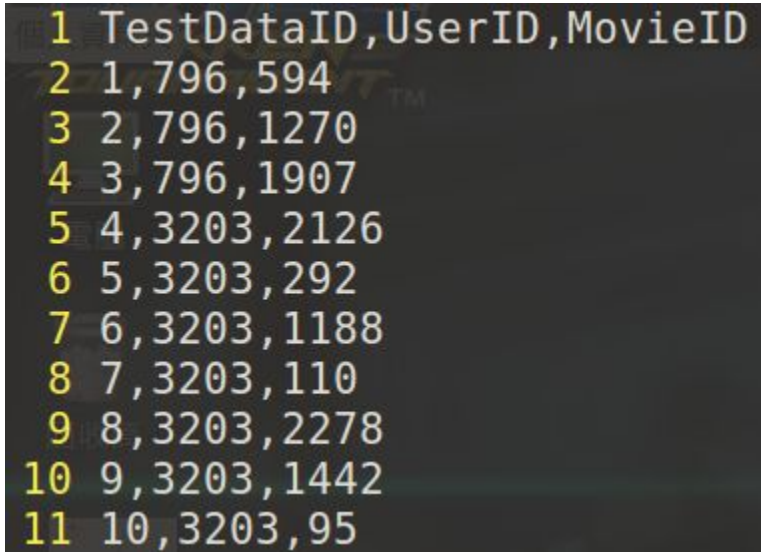
Data format(1/5)

- train.csv
- TrainDataID, UserID,MovieID,Rating



Data format(2/5)

- test.csv
- TestDataID,UserID,MovieID



Data format(3/5)

- movies.csv
- movieID::Title::Genres

```
 1 movieID::Title::Genres
 2 1::Toy Story (1995)::Animation|Children's|Comedy
 3 2::Jumanji (1995)::Adventure|Children's|Fantasy
 4 3::Grumpier Old Men (1995)::Comedy|Romance
 5 4::Waiting to Exhale (1995)::Comedy|Drama
 6 5::Father of the Bride Part II (1995)::Comedy
 7 6::Heat (1995)::Action|Crime|Thriller
 8 7::Sabrina (1995)::Comedy|Romance
 9 8::Tom and Huck (1995)::Adventure|Children's
10 9::Sudden Death (1995)::Action
11 10::GoldenEye (1995)::Action|Adventure|Thriller
12 11::American President, The (1995)::Comedy|Drama|Romance
13 12::Dracula: Dead and Loving It (1995)::Comedy|Horror
```

Data format(4/5)

- users.csv
- UserID::Gender::Age::Occupation::Zip-code

```
 1 UserID::Gender::Age::Occupation::Zip-code
 2 796::F::1::10::48067
 3 3203::M::56::16::70072
 4 4387::M::25::15::55117
 5 4771::M::45::7::02460
 6 1191::M::25::20::55455
 7 2868::F::50::9::55117
 8 1070::M::35::1::06810
 9 5074::M::25::12::11413
10 5585::M::25::17::61614
11 3402::F::35::1::95370
12 5500::F::25::1::04093
```

Data format(5/5)

1. Training data: 899873
2. Testing data: 100336, half private set.

Evaluation

$$RMSE = \sqrt{\sum_{t=1}^{n} \frac{(\hat{y}_t - y_t)^2}{n}}$$

RMSE numpy implementation

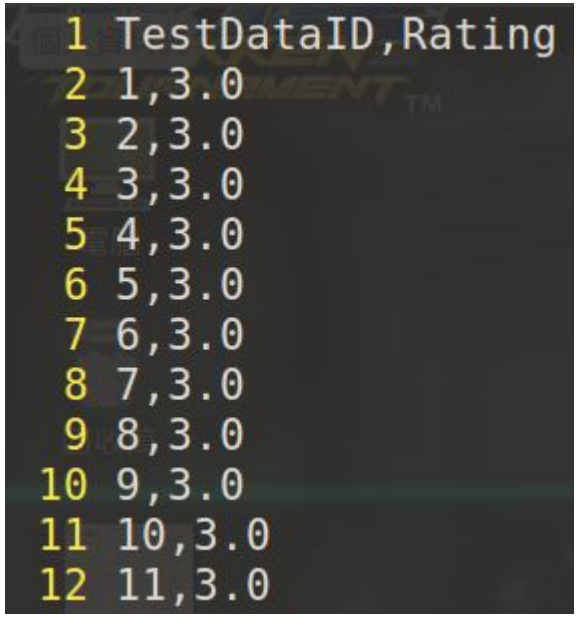
```
np.sqrt(((y_pred - y_true)**2).mean())
```

Kaggle

Kaggle

- kaggle_url：https://inclass.kaggle.com/c/ml2017-hw6
- 請至kaggle創帳號登入，需綁定NTU信箱。
- 個人進行，不需組隊。
- 隊名:學號_任意名稱(ex. b02902000_日本一級棒)，旁聽同學請避免學號開頭。
- 每日上傳上限10次。
- test set的資料將被分為兩份，一半為public，另一半為private。
- 最後的計分排名將以2筆自行選擇的結果，測試在private set上的準確率為準。
- kaggle名稱錯誤者將不會得到任何kaggle上分數。

Submission Format



```
 1 TestDataID,Rating
 2 1,3.0
 3 2,3.0
 4 3,3.0
 5 4,3.0
 6 5,3.0
 7 6,3.0
 8 7,3.0
 9 8,3.0
10 9,3.0
11 10,3.0
12 11,3.0
```

format: TestDataID,Rating

Deadline and Policy

Deadline

1. Kaggle: 6/8 23:59 (GMT+8)
2. Report and source code: 6/9 21:00 (GMT+8)

Policy I - Repository

- github上ML2017/hw6/裡面請至少包含:
  - Report.pdf
  - hw6.sh
  - hw6_best.sh
  - your python files
  - your model files (can be loaded by your python file)
- **請不要上傳dataset**
- **hw6.sh 必須是MF的實作**
- 如果你的model超過github的最大容量，可以考慮把model放在其他地方(同hw3)。
- model可以是多個檔案(ex:keras model, movies id mapping file)，檔名沒有規定，shell script裡寫死相對路徑即可(助教批改時會cd進同學的目錄裡)。

Policy II – Source Code

- **Python Only**，請使用Python 3.6, Python 2.7, Keras 2.0.4, Tensorflow1.1.0, h5py2.7.0
- **只可使用限定的package** (Keras、Tensorflow、Numpy、Pandas、h5py)，**以及python內建的package，並且限定使用Tensorflow作為Keras的backend.** 若import其他東西，或是使用不同版本，造成批改錯誤，將不接受修正。
- 不能使用額外data來training (包括 pre-training)
- 不能call 其他線上 API (Project Oxford...)
- 請附上訓練好的model (及其參數)，hw6.sh 和 hw6_best.sh 要在10分鐘內跑完

Policy II – Source Code

- 與之前作業相同，請在script中寫清楚使用python版本
- 以下的路徑，助教在跑的時候會另外指定，請保留可更改的彈性，不要寫死
  - Script usage:
    bash hw6.sh <data directory> <prediction file>
    bash hw6_best.sh <data directory> <prediction file>
    data directory: 放test.csv, users.csv, movies.csv的資料夾，批改時資料夾中檔案名稱會與上述相同。directory最後會有 '/' 字元。
    prediction file: 結果的csv檔路徑

Policy III - Report

- 請使用中文作答，若使用英文請確定語意清楚☺。
- 請交pdf檔，檔名為Report.pdf
- Report Template： Link

Policy IV - Score

- Kaggle Rank
  - (1%) 超過public leaderboard的simple baseline分數
  - (1%) 超過public leaderboard的strong baseline分數
  - (1%) 超過private leaderboard的simple baseline分數
  - (1%) 超過private leaderboard的strong baseline分數
  - (1%) 5/31 23:59 (GMT+8)前超過public simple baseline
  - (BONUS) kaggle排名前五名(且願意上台跟大家分享的同學)
- 前五名排名以public以及private平均為準，屆時助教會公布名單
- **hw6.sh的結果必須超過public simple baseline否則程式部分將不會有任何分數。**

Policy IV - Score

- Report problem**(Q1,Q2,Q3,Q5都限定使用MF的model)**

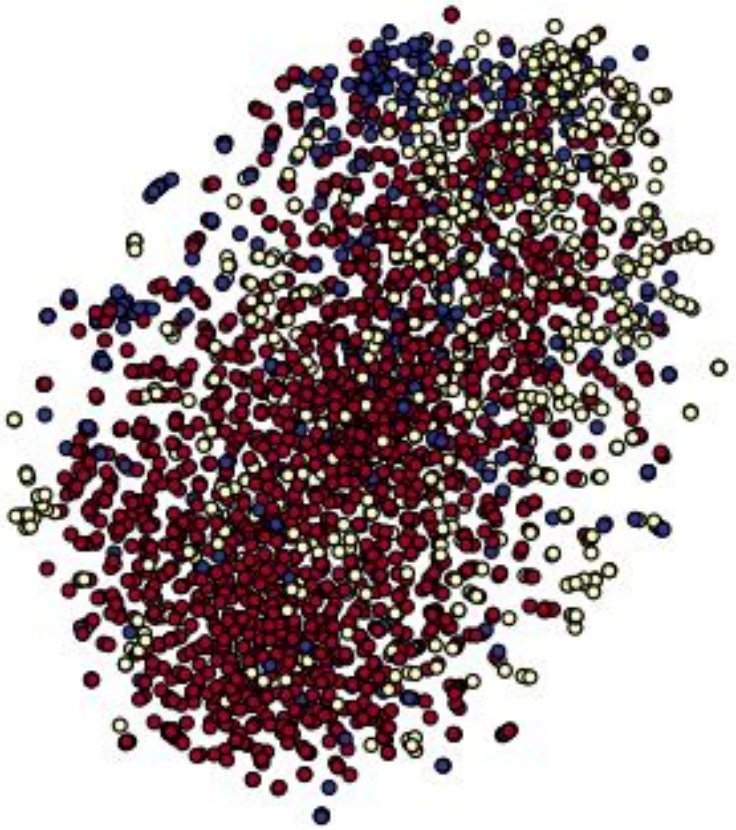
1. (1%)請比較有無normalize(在rating上)的差別。並說明如何normalize.
2. (1%)比較不同的latent dimension的結果。
3. (1%)比較有無bias的結果(p.8)。
4. (1%)請試著用DNN(p.28)來解決這個問題，並且說明實做的方法(方法不限)。並比較MF和NN的結果，討論結果的差異。
5. (1%)請試著將movie的embedding用tsne降維後，將movie category當作label來作圖(p.29)。
6. (BONUS)(1%)試著使用除了rating以外的feature, 並說明你的作法和結果，結果好壞不會影響評分。

DNN問題

1. DNN input: 舉例來說可以把user的embedding以及movie的embedding連接在一起，作為DNN的input
2. DNN output: 可以把這個問題視為regression問題，又或者將1-5每種分數都視為不同類別，再去做5個類別的分類問題

Movie category作圖範例

- 由於dataset給的movie分類有幾類滿像的，這張圖是把’Drama’,’Musical’作為一類，'Thriller',’Horror’,’Crime’作為一類，Adventure,Animation,Children's做為一類所畫的圖。在畫圖時同學的類別可以自訂，有區分效果即可。**有多個分類時，可以隨機選擇一個。**
- 米色是’Drama’,’Musical’,
  紅色是'Thriller',’Horror’,’Crime’
  藍色是Adventure,Animation,Children's



Other Policy

1. script 錯誤，直接0分。若是格式錯誤，請在公告時間內找助教修好，修完kaggle分數*0.7。
2. Kaggle超過deadline直接shut down，可以繼續上傳但不計入成績。
3. Github遲交一天(*0.7)，不足一天以一天計算，不得遲交超過兩天，有特殊原因請找助教。
4. Github遲交表單：https://goo.gl/forms/pEruhVFMr2uDnHFo2(遲交才需填寫)
5. 遲交請「先上傳程式」Github再填表單，助教會根據表單填寫時間當作繳交時間。
6. 請勿使用任何其他非助教提供的data，否則以0分計算

FAQ

FAQ

- 作業網址:
  https://inclass.kaggle.com/c/ml2017-hw6
- 若有其他問題，請po在FB社團裡或寄信至助教信箱，**請勿直接私訊助教**。
- 助教信箱：ntu.mlta@gmail.com