EE4016 Engineering Applications of Artificial Intelligence

Project 2

Low-Rank Matrix Completion for Movie Recommendation

Intended Learning Outcomes

On completion of this project, you should be able to

- Apply low-rank matrix factorization to find the missing entries of incomplete matrices.
- Evaluate the recommender performance of a matrix factorization method using real movie rating data.

Deliverable

- Each student is required to submit a report on or before Week 13 (22 Nov.).

Background

A recommender system can be modelled as an incomplete matrix whose column and row indices represent the user and item identity numbers, while the known entries are the observed ratings. The recommendation task is to predict/estimate the unobserved ratings or missing entries of the matrix.

Mathematically, the incomplete matrix $\boldsymbol{X}_\Omega \in \mathbb{R}^{n_1 \times n_2}$ can be expressed as:

$$[\boldsymbol{X}_\Omega]_{ij} = \begin{cases} \boldsymbol{X}_{ij} = x_{i,j}, & \text{if } (i,j) \in \Omega \\ 0, & \text{otherwise} \end{cases}$$

where Ω is a known subset of the complete set of entries while the unknown entries are set to zero for simplicity.

To perform matrix completion, we assume that $\boldsymbol{X}$ is of rank $r << \min(n_1, n_2)$, and hence it can be represented as:

$$\boldsymbol{X} = \boldsymbol{U}\boldsymbol{V}, \quad \boldsymbol{U} \in \mathbb{R}^{n_1 \times r}, \ \boldsymbol{V} \in \mathbb{R}^{r \times n_2}$$

The solution is given by:

$$\min_{\boldsymbol{U},\boldsymbol{V}} \ J(\boldsymbol{U},\boldsymbol{V}) := \|(\boldsymbol{U}\boldsymbol{V})_\Omega - \boldsymbol{X}_\Omega\|_F^2$$

which can be realized via alternating minimization:

$$\boldsymbol{V}^{k+1} = \arg\min_{\boldsymbol{V}} \ \|(\boldsymbol{U}^k\boldsymbol{V})_\Omega - \boldsymbol{X}_\Omega\|_F^2$$

and

$$\boldsymbol{U}^{k+1} = \arg\min_{\boldsymbol{U}} \ \|(\boldsymbol{U}\boldsymbol{V}^{k+1})_\Omega - \boldsymbol{X}_\Omega\|_F^2$$

Upon convergence of $\boldsymbol{U}^{k+1}$ and $\boldsymbol{V}^{k+1}$, $\boldsymbol{X}$ is estimated as:

$$\boldsymbol{M} = \boldsymbol{U}^{k+1}\boldsymbol{V}^{k+1}$$

In this project, this matrix factorization method is applied to evaluate the movie recommender system performance using the MovieLens 100K dataset [1]:

- 100000 ratings from 943 users on 1682 movies, i.e., $\boldsymbol{X}_{\Omega} \in \mathbb{R}^{1682\times 943}$ with around 94% missing entries
- Rating is from 1 to 5
- Each user has rated at least 20 movies
- More information can be found in README.txt

Two main tasks are:

- Estimate the matrix rank of the MovieLens 100 K dataset
- Compute the empirical mean square error (MSE)

Experimental Works

Python and/or MATLAB can be used. Nevertheless, a MATLAB version for the matrix factorization algorithm is provided.

1. Data Preprocessing

Outline:

- Download MovieLens 100K dataset at [1].
- Study the dataset.
- Extract the first 3 columns of `u.data`, which correspond to the user identity number, item identity number and rating, respectively. Note that there are a total of 100000 observed ratings.
- The matrix dimensions are 1682 x 943. Remove any row or column which has less than 2 entries. This means that the incomplete matrix is $\boldsymbol{X}_{\Omega} \in \mathbb{R}^{n_1\times n_2}$ where $n_1 \leq 1682$ and $n_2 \leq 943$, and the number of observed entries in $\boldsymbol{X}_{\Omega}$ is at most 100000.
- Obtain two incomplete matrices from $\boldsymbol{X}_{\Omega}$, namely, $\boldsymbol{X}_{\Omega_1}$ and $\boldsymbol{X}_{\Omega_2}$, where each of them contain around 50000 observed ratings such that $\Omega = \Omega_1 \cup \Omega_2$ and $\Omega_1 \cap \Omega_2 = \varnothing$.
- Make sure each student will work on a distinct pair of workable $\boldsymbol{X}_{\Omega_1}$ and $\boldsymbol{X}_{\Omega_2}$. There are many solutions to achieve this. For example, for each row, assign one entry to $\boldsymbol{X}_{\Omega_1}$, and then for each column, assign one entry to $\boldsymbol{X}_{\Omega_1}$. Apply this procedure to $\boldsymbol{X}_{\Omega_2}$ as well. In doing so, $\boldsymbol{X}_{\Omega_1}$ and $\boldsymbol{X}_{\Omega_2}$ should be workable in the sense they have no all-zero row or column. The next step is to fill $\boldsymbol{X}_{\Omega_1}$ and $\boldsymbol{X}_{\Omega_2}$ with the remaining entries randomly so that each of the $\boldsymbol{X}_{\Omega_1}$ and $\boldsymbol{X}_{\Omega_2}$ has around 50000 ratings.

2. Performance Evaluation

Outline:

- Estimate the best rank r to approximate $\boldsymbol{X}_{\Omega}$ based on minimum empirical MSE. Try $r = 1, 2, 3, 4$ and 5.

- Compute the empirical MSE. For a given r, suppose the matrix estimate of $\boldsymbol{X}$ using $\boldsymbol{X}_{\Omega_1}$ is $\boldsymbol{M}_1$. The empirical MSE for $\boldsymbol{M}_1$ is computed using the squared difference between $\boldsymbol{M}_{1\Omega_2}$ and $\boldsymbol{X}_{\Omega_2}$:

$$\text{MSE}_{r,1} = \frac{\|\boldsymbol{M}_{1\Omega_2} - \boldsymbol{X}_{\Omega_2}\|_F^2}{|\Omega_2|}$$

where $\|\boldsymbol{M}_{1\Omega_2} - \boldsymbol{X}_{\Omega_2}\|_F^2$ is the sum of squared differences between the entries of $\boldsymbol{M}_{1\Omega_2}$ and $\boldsymbol{X}_{\Omega_2}$ within the set Ω_2, and $|\Omega_2|$ represents the number of observed ratings in $\boldsymbol{X}_{\Omega_2}$, which should be around 50000. As an example, suppose the estimated and true values of $\boldsymbol{X}_{\Omega}$ are:

$$\boldsymbol{M}_{\Omega} = \begin{bmatrix} 5.5 & 0.1 \\ 3.3 & \times \end{bmatrix}, \qquad \boldsymbol{X}_{\Omega} = \begin{bmatrix} 5 & 1 \\ 4 & \times \end{bmatrix}, \qquad |\Omega| = 3$$

where $\times$ denotes a missing entry in $\boldsymbol{X}_{\Omega}$. The MSE is then:

$$\text{MSE} = \frac{(5.5-5)^2 + (0.1-1)^2 + (3.3-4)^2}{3} \approx 0.52$$

Similarly, let the matrix estimate of $\boldsymbol{X}$ using $\boldsymbol{X}_{\Omega_2}$ be $\boldsymbol{M}_2$. The empirical MSE for $\boldsymbol{M}_2$ is given by:

$$\text{MAE}_{r,2} = \frac{\|\boldsymbol{M}_{2\Omega_1} - \boldsymbol{X}_{\Omega_1}\|_F^2}{|\Omega_1|}$$

The overall empirical MSE is:

$$\text{MSE}_r = \frac{\text{MSE}_{r,1} + \text{MSE}_{r,2}}{2}$$

Of course, weighted average can also be used when $|\Omega_1| \neq |\Omega_2|$.

- Note that the matrix factorization method can guarantee a local solution only. This means the obtained $\boldsymbol{M}_1$ (or $\boldsymbol{M}_2$) may differ when running the algorithm again. As a result, try the algorithm several times and choose the solution corresponding to the minimum value of $\|\boldsymbol{M}_{1\Omega_2} - \boldsymbol{X}_{\Omega_2}\|_F^2$ or ($\|\boldsymbol{M}_{2\Omega_1} - \boldsymbol{X}_{\Omega_1}\|_F^2$).

- It is known that the rating is restricted to an integer between 1 and 5. Use this prior knowledge to check if the recommender system performance can be improved or not. That is, compare the results using the computed $\boldsymbol{M}_{1\Omega_2}$ and $\boldsymbol{M}_{2\Omega_1}$, and their rounded versions. For the latter, each entry of $\boldsymbol{M}_{1\Omega_2}$ (or $\boldsymbol{M}_{2\Omega_1}$) should be rounded to the nearest integer from 1 to 5. For example, if the computed entries are -0.5, 3.3 and 6.1, they will be rounded to 1, 3 and 5, respectively.

Report Requirements

The report should include:

- Your data preprocessing procedure
- Your performance evaluation procedure
- Results and findings
- Discussions and conclusion

References:

[1] https://grouplens.org/datasets/movielens/100k/

[2] Y. Koren, R. Bell and C. Volinsky, “Matrix factorization techniques for recommender systems,” *Computer*, vol. 42, no. 8, Aug. 2009, pp. 30-37.