4033/5033: Assignment 5

Your Name

Due: Nov 11 (by 11:59pm)

In this assignment, we will implement the low-rank matrix factorization technique using alternating least square for the collaborative filtering problem. We have studied how to derive the update rule of $A_{k:}$ in class. Your job is to derive the update rule of $B_{:k}$ and implement this technique in Python.

Notation: Let M be an n-by-m rating matrix and O be its set of observed entries. Let A be an n-by-k matrix and B be a k-by-m matrix. Let $A_{k:}$ be the k_{th} row of matrix A and $B_{:k}$ be the k_{th} column of matrix B. Recall we will solve the following optimization problem using alternating least square:

$$\min_{A,B} \sum_{(i,j)\in O} (A_{i:}B_{:j} - M_{ij})^2 + \lambda_1 ||A||_F^2 + \lambda_2 ||B||_F^2, \tag{1}$$

where λ_1, λ_2 are hyper-parameters.

Task 1. Derive the following update rule of $B_{:k}$

$$B_{:k} = \left(\sum_{i\in O'_k} A_{i:}^T A_{i:} + \lambda I \right)^{-1} \left(\sum_{i\in O'_k} M_{ik} A_{i:}^T \right) \tag{2}$$

Task 2. Implement the alternating least square algorithm on the lectured slides and evaluate the prediction performance on the given data sets[1]. Below are detailed instructions:

– You are given a rate_train.csv file and a rate_test.csv file. They both store ratings of of the same set of 43 users on the same set of 18 movies. Each file contains a rating matrix, where each row corresponds to a user and each column corresponds to a movie. We should treat all zero entries as unobserved ratings and do not use them in training or testing.

– Train your model (i.e. matrices A and B) based on the observed ratings in 'rate_train.csv'.

– Evaluate your model (i.e. matrices A and B) based on the observed ratings in 'rate_test.csv'.

– Evaluate prediction performance using rooted-mean-squared-error.

– It is recommended to initialize all entries in A and B using Gaussian distribution.

Draw Two Figures:

1. Pick a proper k and draw a curve of the prediction error of your model versus the rounds of updates – in each round, we update all rows in A and all columns in B. (In the figure, y-axis is testing error and x-axis is number of rounds.) Pick proper λ_1 and λ_2 so we can observe a smooth and convergent error curve.

2. Draw a curve of the prediction error versus the value of k. Each point on the curve is the (converged) error of your prediction model based on one choice of k.

[1] These are dense subsets of the user-movie rating data set at `https://www.kaggle.com/datasets/shubhammehta21/movie-lens-small-latest-dataset`.

Submission Instruction

Please submit two files to Canvas. (Do not zip them. Upload them separately.)

(i) All your mathematical and experimental results should be presented in a single pdf file named as 'hw5.pdf'.

(ii) A Python source code for the implementation of alternating least square named 'hw5_als.py'