# CS4033/5033: Assignment 5

Antonio Natusch Zarco

The University of Oklahoma

**Topic 1: Low-Rank Matrix Factorization**

Implement low-rank matrix factorization based on the alternate least square technique. Evaluate the technique on a real-world user-movie rating matrix. In this matrix, each row corresponds to a user and each column corresponds to a movie. Ratings take value in {1, 1.5, 2, 2.5, 3, 3.5, 4, 4.5, 5}; you should treat '0' entries in the matrix as (truly) missing ratings and do not use it for either training or testing. For convenience, we have separated the set of observed ratings that should be used for training (stored in 'rate_train.csv') from the set of observed ratings that should be used for testing (stored in 'rate_test.csv').

Task 1. Learn a rank-$k$ factorization of the rating matrix from the training set and evaluate it on the testing set. Report testing error versus the number of ALS updates in Figure 1. Pick the value for $k$, $\lambda_1$, $\lambda_2$ and the maximum number of ALS updates by yourself.

Tip: you may round the predicted rates (given by $UV$) to the nearest value in {1, 1.5, 2, 2.5, 3, 3.5, 4, 4.5, 5} to get a more meaningful prediction and possibly higher prediction accuracy.

Task 1 - Solution.

We model the rating matrix $X$ as a low-rank product $X \approx UV$ with $U \in \mathbb{R}^{n \times k}$, $V \in \mathbb{R}^{\urcorner \times >}$. Let $O \subset \{1, \ldots, n\} \times \{1, \ldots, m\}$ be the set of observed (non-zero) entries in $X$. The ridge-regularized objective function is:

$$\min_{U,V} \sum_{(i,j) \in O} (X_{ij} - U_{i:} V_{:j})^2 + \lambda_1 \|U\|_F^2 + \lambda_2 \|V\|_F^2 \tag{1}$$

Holding $V$ fixed, each user row $U_{i:}$ solves a regularized least-squares normal equation, and symmetrically for each movie column $V_{:j}$. We alternate these updates for a fixed number of iterations.

$$U_{i:} = (\sum_{j:(i,j) \in O} X_{ij} V_{:j}^T)(\sum_{j:(i,j) \in O} V_{:j} V_{:j}^T + \lambda_1 I)^{-1} \tag{2}$$

,

$$V_{:j} = (\sum_{i:(i,j) \in O} U_{i:}^T U_{i:} + \lambda_2 I)^{-1}(\sum_{i:(i,j) \in O} X_{ij} U_{i:}^T) \tag{3}$$

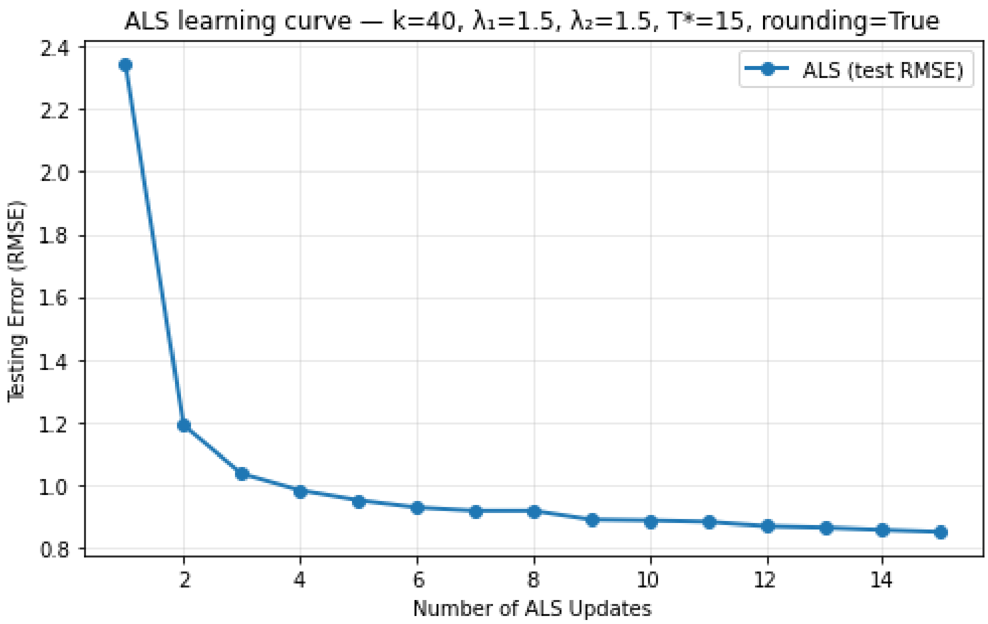


**Fig. 1.** Testing Error versus ALS Updates with $k = 40$

Task 2. Report testing error versus $k$ in Figure 2. Pick five values of $k$ and fix other configurations by yourself.

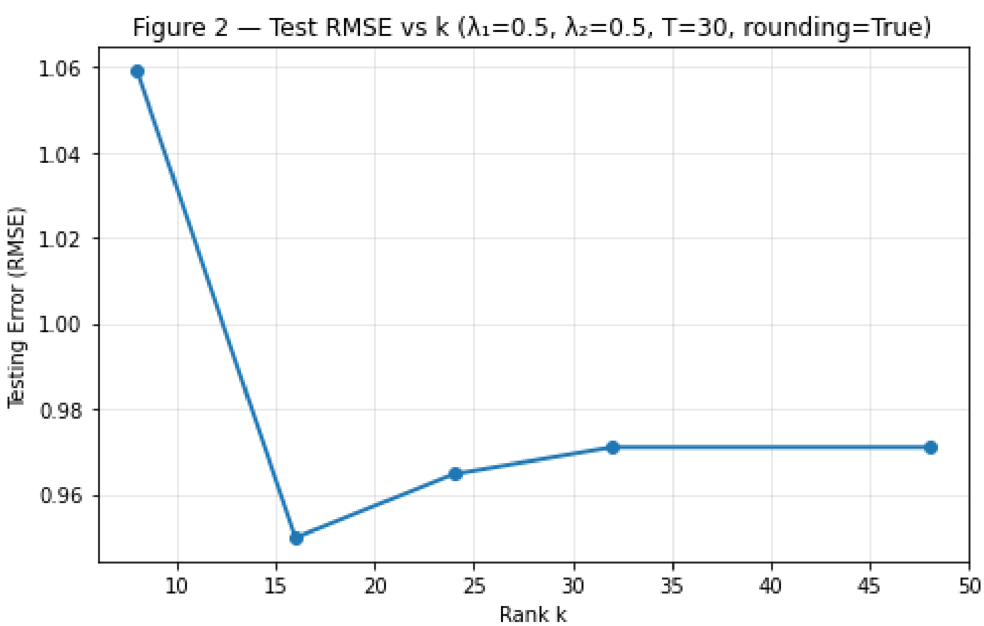


**Fig. 2.** Testing Error versus $k$

**Topic 2: Markov Model**

Suppose we want to learn a Markov model, and apply it to predict Sam's final grade as A or B. Our training data are shown in Figure 3. (The first row is Sam's record. Do not use it for training.)

Task 3. Assume 1st-order Markov chain. Evaluate the probability for Sam to get an A in the 5th semester. You need to elaborate on the estimation of key probabilities and the key formula such as Bayes theorem.

Task 3 - Solution. We model grades $X_t \in \{A, B\}$ as a time-homogeneous first-order Markov chain:

$$Pr(X_{t+1} = y | X_1, \ldots, X_t = x_t) = Pr(X_{t+1} = y | X_t = x_t) \equiv p_{x_t y} \tag{4}$$

We need $Pr(X_5 = A \mid X_4 = B)$ for Sam. By the property this is exactly the transition probability $p_{BA}$

Estimate $p_{xy}$ from the training rows (X1-X8 only) via MLE using bigram counts over adjacent semesters $t \to t+1$ (for $t = 1, 2, 3, 4$):

$$N_{xy} = \#\{t : X_t = x, X_{t+1} = y\}, \qquad N_x = \sum_{y \in \{A,B\}} N_{xy}, \qquad \hat{p}_{xy} = \frac{N_{xy}}{N_x} \tag{5}$$

Equivalently, by Bayes' rule,

$$Pr(X_{t+1} = A \mid X_t = B) = \frac{Pr(X_{t+1} = A)}{Pr(X_t = B)} \approx \frac{N_{BA}/N}{N_B/N} = \frac{N_{BA}}{N_B} \tag{6}$$

where $N = \sum_{x,y} N_{xy}$ is the total number of observed transitions; the common N cancels.

From the training data, (rows $X_1 : X_8$, not using Sam), counting all $B \to \cdot$ transitions across the 4 semester steps gives

$$N_{BA} = 6, \quad N_{BB} = 4, \quad N_B = 10 \tag{7}$$

Hence

$$Pr(X_5 = A \mid X_4 = B) = \hat{p}_{BA} = \frac{6}{10} = 0.6 \tag{8}$$

So, under a 1st-order, time-homogeneous Markov model, the estimated probability that Sam gets an $A$ in the 5th semester (given $X_4 = B$) is 0.6.

**Note:**

**Estimate transition matrix**
The estimated transition matrix is:

$$\hat{P} = \begin{bmatrix} \hat{p}_{AA} & \hat{p}_{AB} \\ \hat{p}_{BA} & \hat{p}_{BB} \end{bmatrix} = \begin{bmatrix} \frac{1}{3} & \frac{2}{3} \\ \frac{3}{5} & \frac{2}{5} \end{bmatrix} \tag{9}$$

Task 4. Assume 2nd-order Markov chain. Evaluate the probability for Sam to get an A in the 5th semester. You need to elaborate on the estimation of key probabilities and the key formula such as Bayes theorem.

Task 4 - Solution. Let $X_t \in \{A, B\}$ denote the grade in semester $t$. We model grades as a time-homogeneous second-order Markov chain:

$$Pr(X_{t+1} = z \mid X_t = y, X_{t-1} = x) = \theta_{xy\to z}, \quad x, y, z \in \{A, B\} \tag{10}$$

For Sam we need

$$Pr(X_5 = A \mid X_4 = B, X_3 = A) = \theta_{AB\to A} \tag{11}$$

since Sam's history is $X_1 = B$, $X_2 = A$, $X_3 = A$, $X_4 = B$.

[MLE of the 2nd-order transition]

From the training rows only (X1-X8), form all adjacent triples $(X_t, X_{t+1}, X_{t+2})$ for $t = 1, 2, 3$ in each row.

For each context pair $(x, y)$, count

$$N_{xy\to z} = \#\{t : X_t = x, X_{t+1} = y, X_{t+2} = z\}, \quad N_{xy} = \sum_{z\in\{A,B\}} N_{xy\to z} \tag{12}$$

The MLE is

$$\hat{\theta}_{xy\to z} = \frac{N_{xy\to z}}{N_{xy}} \tag{13}$$

Equivalently, by Bayes' rule,

$$Pr(X_{t+2} = z \mid X_{t+1} = y, X_t = x) = \frac{Pr(x, y, z)}{Pr(x, y)} \approx \frac{N_{xy\to z}}{N_{xy}} \tag{14}$$

| Semester | 1st | 2nd | 3rd | 4th | 5th |
|---|---|---|---|---|---|
| Sam | B | A | A | B | ? |
| X1 | B | A | B | B | B |
| X2 | A | B | A | B | A |
| X3 | A | A | B | A | B |
| X4 | B | B | A | B | A |
| X5 | A | A | A | B | A |
| X6 | A | B | B | A | A |
| X7 | B | B | A | B | B |
| X8 | B | A | B | A | B |

**Fig. 3.** Sam's and other students' grade records

Submission Instruction

Please submit three files on Canvas.

(i) Submit a 'hw5.pdf'. It should contain your answers to all questions in this template.

(ii) Submit a 'hw5_als.py'. It should be the code that generates Figure 1.

(iii) Submit a 'hw5_k.py'. It should be the code that generates Figure 2.