Mining Media Data I Winter Semester 25/26

Assignment-02

Prof. Dr. Rafet Sifa Armin Berger Dr. Lorenz Sparrenberg

17.12.2025

Introduction

In this assignment, you will investigate methods for matrix factorization. We will start by implementing and comparing two algorithms for factorizing a data matrix. All programming tasks must be completed in **Python 3.10+**. You are encouraged to use `NumPy` for numerical computations and `Matplotlib` for visualization.

This assignment contributes a total of **30 points** to your final grade.

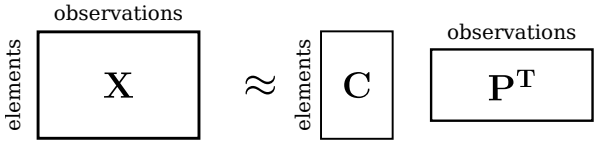
1 Matrix Factorization (30 Pts.)

1.1 Implementation of the Rotated Matrix Regression Algorithm (8 Pts.)

In this part, you will implement the *Rotated Matrix Regression Algorithm* to approximate the data matrix

$$\mathbf{X} \approx \mathbf{C}\mathbf{P}^T,$$

where $\mathbf{X} \in \mathbb{R}^{m \times n}$ is the observed data, $\mathbf{C} \in \mathbb{R}^{m \times k}$ is the basis (or component) matrix, and $\mathbf{P} \in \mathbb{R}^{n \times k}$ is the coefficient (or loading) matrix.



Use the data in Table 1 as your initial matrix $\mathbf{X} \in \mathbb{R}^{m \times n}$. Each row represents a movie, each column a user, and a checkmark indicates that a user likes a movie. Assume that users have underlying preferences for a small number of latent "movie types" (genres).

Tasks.

1. Construct the matrix $\mathbf{X}$ from Table 1, using a suitable numerical encoding (e.g. 1 for a checkmark, 0 otherwise). Briefly document your encoding choice.
2. Implement the Rotated Matrix Regression Algorithm from the lecture to factorize $\mathbf{X}$ into $\mathbf{C}$ and $\mathbf{P}$ for a given latent dimension k.
3. Experiment with several values of k (e.g. $k = 1, 2, 3, 4$):
   (a) For each k, compute an error measure between $\mathbf{X}$ and its reconstruction $\hat{\mathbf{X}} = \mathbf{C}\mathbf{P}^T$.
   (b) Plot the reconstruction error as a function of k.
4. Answer and discuss:
   - What happens to the reconstruction quality as k increases?
   - Which error measure from the lecture do you use, and why is it appropriate for this problem?

1.2 Compare to Gradient Descent (8 Pts.)

Next, you will solve the same matrix factorization problem using (batch) gradient descent instead of the Rotated Matrix Regression updates.

Tasks.

1. Starting from the objective function
$$L(\mathbf{C}, \mathbf{P}) = \left\|\mathbf{X} - \mathbf{C}\mathbf{P}^T\right\|_F^2,$$
derive the gradients $\nabla_{\mathbf{C}} L$ and $\nabla_{\mathbf{P}} L$.
2. Implement gradient descent updates for $\mathbf{C}$ and $\mathbf{P}$ using your derived gradients.
3. Introduce a learning rate (step size) parameter η:

Table 1: Example movie data

	user 1	user 2	user 3	user 4	user 5	user 6	user 7	user 8	user 9	user 10	user 11
movie 1					✓	✓	✓				
movie 2	✓		✓	✓							
movie 3									✓		✓
movie 4		✓		✓							
movie 5					✓						
movie 6								✓	✓	✓	
movie 7					✓	✓					
movie 8								✓		✓	✓
movie 9	✓	✓	✓	✓							

- Explain why a learning rate is necessary in gradient descent.
- Experiment with at least three different learning rates (e.g. small, medium, large). For each, plot the reconstruction error versus the number of gradient descent steps.

4. Compare the two approaches (Rotated Matrix Regression vs. gradient descent):
   - How many iterations/steps are required to reach a "good" level of convergence for each method?
   - How stable are the methods with respect to initialization and choice of hyperparameters (e.g. learning rate)?
   - Provide a short explanation for the observed differences in convergence speed and stability.

1.3 Choosing k for Latent Pattern Mining (8 Pts.)

In latent pattern mining, our goal is not only to reconstruct the original matrix $\mathbf{X}$ but to find useful representations (latent factors) that can support downstream tasks such as building a recommender system.

Tasks.

1. Explain in your own words:
   - Why we are *not* necessarily interested in a perfect reconstruction of $\mathbf{X}$ using $\mathbf{C}$ and $\mathbf{P}$.
   - How overfitting can occur if k is chosen too large, and how underfitting can occur if k is chosen too small.
2. Propose and implement a strategy to select a "good" value for k. Clearly describe your chosen strategy.
3. For a reasonable range of k (e.g. $k = 1, \ldots, 6$):
   (a) Compute the reconstruction error for each k.
   (b) Visualize your results using `Matplotlib`.
4. Based on your experiments:
   - Which value of k would you choose for this data and why?
   - How do the latent patterns for this k relate to intuitive user groups or movie genres in the table?

1.4 Noisy Data Interpretation (6 Pts.)

In reality, user feedback is often noisy: users may make mistakes, have inconsistent tastes, or give random ratings.

Tasks.

1. Construct a noisy version of your data matrix $\mathbf{X}$, denoted $\mathbf{X}^{\text{noisy}}$. For example, you may:
   - Flip each entry of $\mathbf{X}$ (from 0 to 1 or from 1 to 0) with a small probability

p (e.g. $p = 0.1$), or

Describe briefly how you generated the noise.

2. Repeat your matrix factorization on $\mathbf{X}^{\text{noisy}}$ using the value of k you previously selected. Compare:

   - Reconstruction errors for $\mathbf{X}$ vs. $\mathbf{X}^{\text{noisy}}$,
   - Heatmaps of $\hat{\mathbf{X}}$ vs. $\hat{\mathbf{X}}^{\text{noisy}}$ (e.g. using a clustered heatmap as before).

   How does the additional noise affect the clarity of the latent blocks (clusters of users and movies)?

3. In the lecture you learned about *Archetypal Analysis (AA)*. Briefly explain the main idea of AA (what archetypes are and how data points are represented using them), and discuss how such an archetype-based representation could help to interpret more realistic, noisy rating data in which users may have mixed preferences.

2 Presenting the results

The results will be presented as either a Jupyter Notebook or a PDF with an accompanying code base. They will be discussed in an exercise session.

Submission

All the submissions will be made electronically by sending a single `.zip` file (including your Python code and PDF or Jupyter Notebook) to sparrenberg@bit.uni-bonn.de by the submission deadline with the title `MMD WS2025 Assignment 02 [GroupID]`, where `[GroupID]` refers to your group id (name).

Submissions sent after the deadline and not following the title convention will not be evaluated. The submission deadline for this assignment is on 16/12/2025 at 23:59 (CET).

A Note on Plagiarism

Work containing plagiarism will not be graded. After a second warning, a disciplinary process will be started, and the corresponding students will not be allowed to attend the final examination.