HY573 - Assignment #1

Grigorios Tsagkatakis

30/10/2023

Submission Guidelines

- **Due date: 12/11/2023**
- Submission via e-learn
- Provide one zip file with the written solutions and the code.
  - Provide a report discussing the different topics.
  - Incorporate your code, visualizations, and intermediate outputs in a Jupyter notebook.

1 Exercise 1: Singular Value Decomposition and Principal Component Analysis on the paper

1.1 Singular Value Decomposition (SVD)

Find the SVD of the matrix:

$$A = \begin{bmatrix} 1 & 3 \\ 3 & 1 \\ 1 & 1 \end{bmatrix}$$

(a) Find the eigenvalues and unit-length eigenvectors for $A^T A$ and AA^T. What do the singular values represent? What is the rank of A? Can a matrix's rank be greater than its dimensions?

(b) Calculate the three matrices U, Σ, and V^T in the SVD and observe that $A = U\Sigma V^T$. Please explain clearly how you obtain these matrices.

(c) Explain how you can use the SVD of matrix A to perform dimensionality reduction. How would the reduced U_r, Σ_r, and V_r look like in this case? How can matrix A be expressed according to these reduced matrices?

1.2 Principal Component Analysis (PCA)

(a) Find the principal components of matrix A. What is the relationship between the singular values and the principal components?

(b) Project the data into 2 and 3 dimensions.

2 Exercise 2: Interpretation and Reduction of the MovieLens Dataset Using SVD and PCA

Welcome to the world of movie recommendations! In this exercise, you will delve into the MovieLens dataset, a treasure trove of movie ratings and user preferences. MovieLens is a platform where movie enthusiasts rate films and apply tags to share their opinions and categorize movies. This dataset allows us to explore user behavior and preferences, offering insights into movie recommendations and user-item interactions. So, grab your popcorn, and let's begin!

2.1 The MovieLens Dataset

Download the dataset from the MovieLens website. Choose the "Small" dataset, which includes 9724 movies scored by 610 users. After obtaining the dataset, load it into our preferred environment (e.g. Jupyter notebook on google colab).

2.2 Data Exploration

Now that we have the dataset, it's time to explore the movie universe:

(a) Calculate the number of users, movies, and ratings in the dataset, setting the stage for our analysis.

(b) Visualize the distribution of movie ratings. Are viewers generally generous or critical in their ratings?

(c) Do any gaps exist in our data? Identify and visualize any missing ratings to understand data completeness

(d) Generate the user rating - movie matrix from the data (it should be 610×9724).

(e) Visualize the sparsity of the user-item matrix, e.g. via a heat map.

2.3 Data interpretation using SVD

Using Singular Value Decomposition (SVD) you can reveal the secrets hidden within the dataset.

(a) Decompose the user-item matrix and dissect its components.

(b) Identify and plot the singular values. What is the information included in these values?

(c) Examine the left singular vectors (related to users) and the right singular vectors (associated with movies). Are there patterns or clusters that illuminate user preferences and movie genres?

(d) How do missing values affect the performance? What is the impact of replacing nan with zeros?

2.4 Reducing Complexity with PCA

Use Principal Component Analysis (PCA) to simplify and visualize the data. If needed, standardize the user-item matrix to ensure our analysis is consistent.

(a) Implement PCA for dimensionality reduction.

(b) Project the data onto the first principal component. What is the information given by this projection?

(c) Explore further by projecting the data onto the first two and then three principal components. Discuss the variance retained by these projections. Are there any patterns or clusters revealed? (Hint: Use scatter and 3D plots.)

3 User Similarity via PCA Projections

Objective: To apply PCA on a subset of users, then validate the derived low-dimensional space using a different set of users, and find the nearest neighbors in this reduced dimensionality.

Tasks:

A. Train-Validation Split:

1. Randomly select a subset of users (e.g., 70% of total users) as your training set. The remaining users will serve as your validation set.

B. PCA on Training Data:

1. For the training set, create a user-item matrix. Remember that this matrix might be very sparse, with a high percentage of missing values.

2. Apply PCA on this user-item matrix to reduce its dimensionality. For instance, project the data onto the first two or three principal components.

3. Plot the projected users in the low-dimensional space to visualize the distribution.

C. Projecting Validation Users:

1. For each user in the validation set, project them onto the PCA space derived from the training set. This can be achieved using the PCA transformation matrix obtained from the training phase.

2. Plot these new projections on the same PCA space to differentiate between training and validation users.

D. Finding Nearest Neighbors in Low-Dimensional Space:

1. For each user in the validation set, find the nearest neighbor (or neighbors, e.g., top-5 closest users) from the training set within the PCA space.

2. Use a suitable distance metric (like Euclidean distance) to determine similarity.

3. Analyze these pairings: Do these nearest neighbors have similar movie preferences? Validate by checking common movies and their ratings.

1. Reflect on the effectiveness of PCA in capturing user similarity. How did dimensionality reduction impact the ability to find similar users?

2. Discuss potential improvements or alternative methods to further refine user similarity detection in reduced spaces.