Recommender Systems Lecture —4

Matrix Factorization for Clustering

Matrix Factorization (MF) isn't just for recommendations or dimensionality reduction; it plays a crucial role in clustering, such as KMeans and Gaussian Mixture Models (GMM).

KMeans Clustering as MF:
- **Objective:** Minimize intracluster distances, represented as minimizing the distance between data points and cluster centroids.

Matrix Representation:
- X: Data matrix with points as rows.
- C: Matrix with cluster centroids as columns.
- Z: Cluster Assignment Matrix, indicating if a point belongs to a cluster (1 if it belongs, 0 otherwise).

Optimization Problem Transformation:
- Transformed into minimizing the Frobenius norm of XZC, aligning with MF's goal of approximating a matrix with the product of two matrices.

Frobenius Norm:
- Measures the "size" of a matrix, equivalent to taking the square root of the sum of the absolute squares of its elements.

Gaussian Mixture Model (GMM) and MF:
- **GMM:** A more flexible version of KMeans that allows for "soft" cluster assignment, making it akin to MF with less rigid constraints on the cluster assignment matrix Z.
- **Difference from KMeans:** Z doesn't have binary constraints, allowing for probabilistic membership values.

Takeaway:
- MF underpins various data analysis techniques, showing its versatility beyond its conventional applications.
- Both KMeans and GMM can be viewed through the MF lens, emphasizing the interconnectedness of machine learning concepts.

Hyperparameter 'd' in Matrix Factorization for Recommender Systems

Hyperparameter tuning is crucial in optimizing matrix factorization (MF) models for recommender systems. The parameter 'd' represents the dimensionality of the latent feature space shared by the user and item matrices.

Optimization Problem:

- The objective is to minimize the Frobenius norm of the difference between the original ratings matrix A and the product of the user matrix U and the item matrix V, focusing only on nonempty cells in A.

Finding 'd':

- 'd' is a hyperparameter indicating the number of latent features.
- Unlike 'k' in clustering, 'd' is chosen through experimentation, affecting the model's ability to approximate A accurately.

Constraints on 'd':

- $d > 0$, and ideally, $d < \min(n, m)$ to ensure a lower-dimensional representation.

Effect of Increasing 'd':

- Higher 'd' values increase model complexity, potentially capturing more information but also risking overfitting.
- Loss typically decreases as 'd' increases due to better approximation capabilities.

Selecting Optimal 'd':

- An elbow or inflection point in the plot of loss vs. 'd' can indicate an optimal tradeoff point.
- Overfitting is a risk if 'd' is chosen solely based on minimizing loss.

Avoiding Overfitting:

- Split the data, using 80% for training and 20% for testing, to evaluate different 'd' values based on actual performance rather than just loss minimization.
- Methods like Stochastic Gradient Descent or Coordinate Descent Algorithm can be used for optimization.

Interpretability of 'd':

- The latent features represented by 'd' do not have direct interpretability; they are numerical values that facilitate the approximation of A.
- Post-factorization techniques like UMAP or tSNE can project high-dimensional user or item vectors into 2D space for visualization, offering insights into user similarity and potential clustering.

In summary, choosing the right 'd' in MF models is a balance between capturing sufficient information to make accurate recommendations and avoiding overfitting by keeping the model sufficiently general. This process requires careful experimentation and validation against a held-out dataset.

Non-Negative Matrix Factorization (NMF) with Equation

NMF is a matrix factorization technique where all elements in the matrices involved are constrained to be non-negative. This approach is particularly useful for data sets where the attributes represent some measurable quantity that cannot be negative.

NMF Equation:

- Given a non-negative data matrix A, NMF aims to find two non-negative matrices W and H such that:
  - $A \approx W \times H$
    - A: Data matrix of dimensions $n \times m$, where n is the number of data points and m is the number of features.
    - W: Basis matrix of dimensions $n \times d$, representing the latent features associated with the data points.
    - H: Coefficient matrix of dimensions $d \times m$, representing how those latent features combine to approximate the original data matrix.

Key Points:

- **Non-Negativity:** Both W and H are constrained to have only non-negative elements, $W \geq 0$ and $H \geq 0$, which ensures that the decomposed components have a meaningful interpretation as quantities or counts.
- **Dimensionality d:** The choice of d (the number of latent features) is a hyperparameter that influences the granularity of the decomposition. It needs to be selected carefully to balance between overfitting and underfitting.
- **Optimization:** Finding W and H typically involves minimizing the difference between A and the product $W \times H$, often measured by the Frobenius norm or other distance metrics, subject to the non-negativity constraints.

Applications and Scenarios Enhanced:

- **Equation Interpretation:** The NMF equation provides a mathematical framework for breaking down high-dimensional data into a lower-dimensional, interpretable space, facilitating applications in recommender systems, image processing, and text mining.

By incorporating the NMF equation into its methodology, the technique enables a structured approach to identify patterns and features within data, making it invaluable for extracting insights and making predictions in various fields.