Recommender Systems via Covariance Neural Networks

How does sparsification affect the performance of covariance VNNs as graph-based collaborative filters?

Martin Angelov
Supervisors: Elvin Isufi, Chengen Liu, Andrea Cavallo

*EEMCS, TU Delft

Problem Statement

Covariance Neural Networks (VNNs) use the covariance of user-item interactions to construct graphs for graph-based collaborative filtering. However, empirical covariance matrices are:

- Often dense and noisy due to high dimensionality and limited data;
- Computationally expensive to process;
- Susceptible to overfitting from spurious correlations.

Goal: Investigate whether sparsifying these graphs through thresholding or probabilistic pruning improves generalization, stability, and runtime.

Mathematical Setup: Let $R \in \mathbb{R}^{n\times m}$ be the user-item rating matrix. The sample covariance is:

$$\Sigma = \frac{1}{n-1}(R-\bar{R})^\top(R-\bar{R})$$

This matrix is treated as a graph adjacency matrix $A = \Sigma$, which is then sparsified to $\tilde{\Sigma}$ before being used in a GNN-like model:

$$H^{(l+1)} = \sigma(\tilde{\Sigma}H^{(l)}W^{(l)})$$

Sparsification Techniques

Hard Thresholding:

$$[\tilde{\Sigma}]_{ij} = \begin{cases} \Sigma_{ij}, & \text{if } |\Sigma_{ij}| \geq \tau \\ 0, & \text{otherwise} \end{cases}$$

Soft Thresholding:

$$[\tilde{\Sigma}]_{ij} = \text{sign}(\Sigma_{ij}) \cdot \max\left(|\Sigma_{ij}| - \tau,\, 0\right)$$

ACV (Absolute Covariance Value): Each entry is retained with probability proportional to its magnitude:

$$p_{ij} \propto |\Sigma_{ij}|$$

RCV (Ranked Covariance Value): Retain top-p proportion of covariances by magnitude.

Motivations:

- Filter out noise and weak correlations.
- Lower training time via sparser matrix operations.
- Improve interpretability and robustness.

Experimental Setup

- **Dataset:** MovieLens 100K (943 users $\times$ 1682 movies)
- **Graph:** Covariance from mean-centered rating matrix
- **Model:** LocalGNN variant with 2 layers, 100 epochs
- **Metrics:** RMSE, Training time, Sparsity (% non-zero)

Results Summary

Test RMSE and Sparsity

Method	Parameter	Test RMSE	Sparsity (%)
Standard	–	0.9973	0.0
Soft Thresholding	$\tau = 8.74$	**0.9898**	25.4
RCV	$p = 0.25$	0.9922	74.8
ACV	–	0.9923	57.9
Hard Thresholding	$\tau = 8.74$	1.0064	25.4

Table 1: Performance of various sparsification methods.

Training Time (Avg)

Method	Parameter	Time (s)
Standard	–	62.93
Soft Thresholding	$\tau = 8.74$	**52.99**
RCV	$p = 0.25$	61.19
ACV	–	62.38

Table 2: Training time per method (avg. of 5 seeds).

Conclusions

- **Soft Thresholding** ($\tau = 8.74$) achieves best trade-off in accuracy and efficiency.
- **ACV/RCV** offer strong performance with minimal hyper-parameter tuning.
- **Sparsification** reduces overfitting and improves interpretability.

Takeaway: Sparsified covariance graphs improve VNN performance, robustness, and scalability in collaborative filtering.