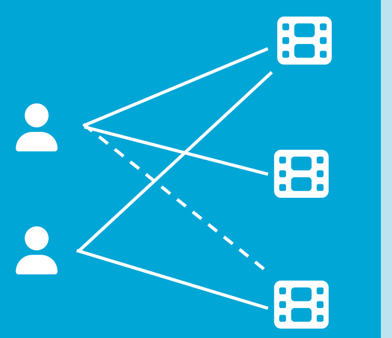
# Evaluating the performance of sparsified precision VNNs as a graph collaborative filter

Author: Jort Boon (j.j.boon@student.tudelft.nl) Responsible professor: Elvin Isufi Supervisors: Andrea Cavallo, Chengen Liu

## Introduction



- **Recommender systems** via collaborative filtering (CF) are a crucial part in many applications, such as social media apps, media streaming services, and many more.
- **Graph Neural Networks (GNNs)** are effective for creating such CF models, as they model relational data as a graph naturally.
- **coVariance Neural Networks (VNNs)** [1,2] are GNNs that define graph connectivity by using the sample covariance matrix.
- VNNs are able to capture non-linear relations better and are more stable to data perturbations than PCA.
- The **Precision Matrix** (the inverse of the covariance matrix) captures conditional dependence between users.
- Since not all covariance matrices are invertible, the Precision Matrix has to be **estimated** using **Graphical Lasso.**
- By **sparsifying** the estimated Precision Matrix, noisy estimates are being filtered out, and computational complexity improves.
- The performance and computational complexity of using the sparsified Precision Matrix in a GNN recommender system are being evaluated.

## Background

For estimating the Precision Matrix, Graphical Lasso is used. Graphical Lasso uses the following minimization objective derived from a multivariate gaussian distribution to estimate the precision matrix:

$$\min_{\Theta^* \succ 0} \left\{ \underbrace{-\log\det(\Theta^*)}_{\text{Log-determinant}} + \underbrace{\mathrm{tr}(\hat{C}\Theta^*)}_{\text{Data fitting}} + \underbrace{\lambda\|\Theta^*\|_1}_{\text{Sparsity penalty}} \right\}$$

- The **Log determinant** term ensures the estimated precision matrix has large enough eigenvalues, making it numerically stable.
- The **Data fitting** term ensures the estimated precision matrix closely relates to the covariance matrix.
- The **Sparsity penalty** term sparsifies the precision matrix, by driving some values in the L1-norm to zero.

## Methodology

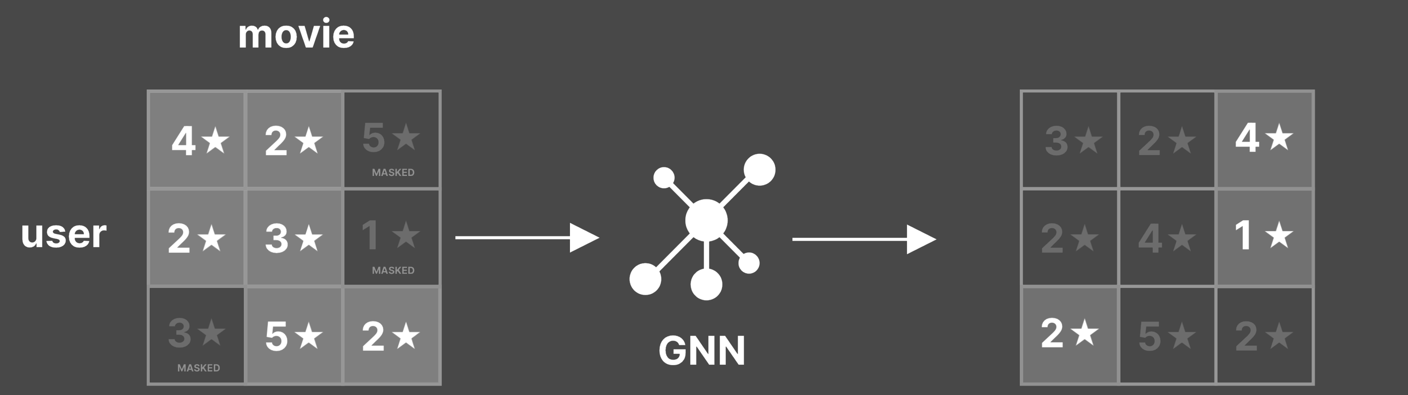
### Dataset

The dataset used is MovieLens-100K [3], which contains movie ratings between 1 and 5, with a total of 943 users and 1682 movies.

### Methods

- In the preprocessing phase, all movie ratings are normalized between -1 and 1.
- The data is then split into training / validation / testing sets with ratios 80% / 10% / 10%.
- The precision matrix for the users is being estimated using the regression-based Graphical Lasso model on the training data.
- The GNN learning strategy uses a matrix reconstruction approach, where 30% of the ratings are masked, and the GNN is tasked to predict these masked ratings.
  For evaluation, the RMSE metric is used.

### Visualization of GNN matrix reconstruction strategy



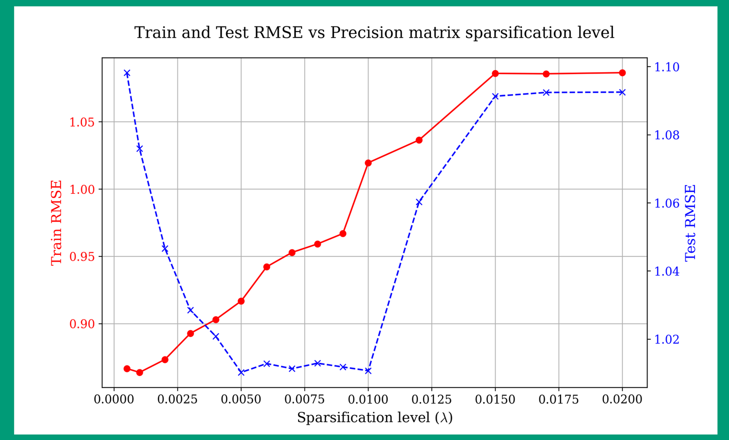

Input ratings
30% of the ratings masked

Predicted ratings
Calculate loss on masked ratings only

## Results

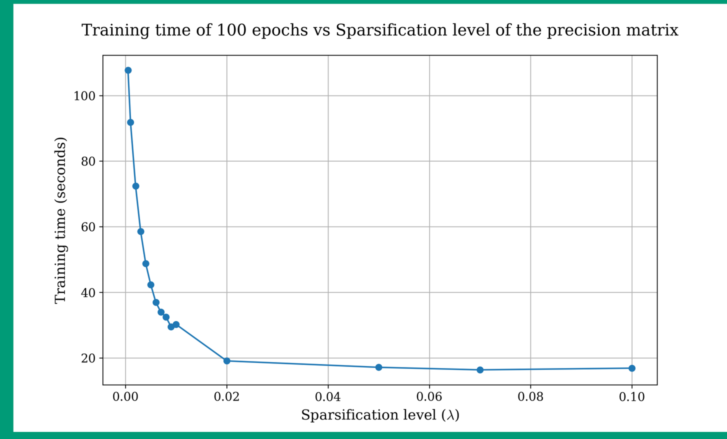
### Performance

Because the user-preference data is sparse, the estimated precision matrix contains a lot of noise. This results in overfitting on the training data, shown by the large gap in Train and Test RMSE. When sparsifying, the noise gets filtered out, resulting in improved Test RMSE (visible at the dip).




### Time complexity

The runtime of the training process significantly reduces with higher levels of sparsification, and maps directly to the number of non-zero elements in the precision matrix.




## Conclusions and future work

- Because data is sparse, the precision matrix estimate contains a lot of noise.
- After sparsifying, both performance and computational complexity improved significantly.
- Further work is needed to evaluate performance on denser datasets.

**References**
[1] Saurabh Sihag et al. *coVariance Neural Networks.* 2022. arXiv: 2205.15856 [cs.LG]. url: https://arxiv.org/abs/2205.15856
[2] Andrea Cavallo, Zhan Gao, and Elvin Isufi. Sparse Covariance Neural Networks. 2024. arXiv: 2410.01669 [cs.LG]. url: https://arxiv.org/abs/2410.01669.
[3] Harper, F. M., & Konstan, J. A. (2015). The MovieLens Datasets: History and context [Data set]. ACM Transactions on Interactive Intelligent Systems (TiiS), 5(4), 1–19. https://doi.org/10.1145/2827872

TU Delft