A HYBRID MATRIX FACTORIZATION VIA GRAPH SIGNAL PROCESSING

NETWORK TOUR OF DATA SCIENCE

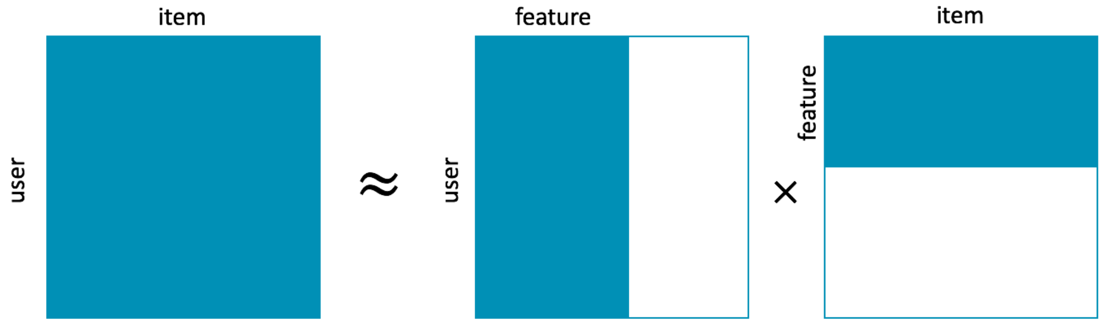
EPFL

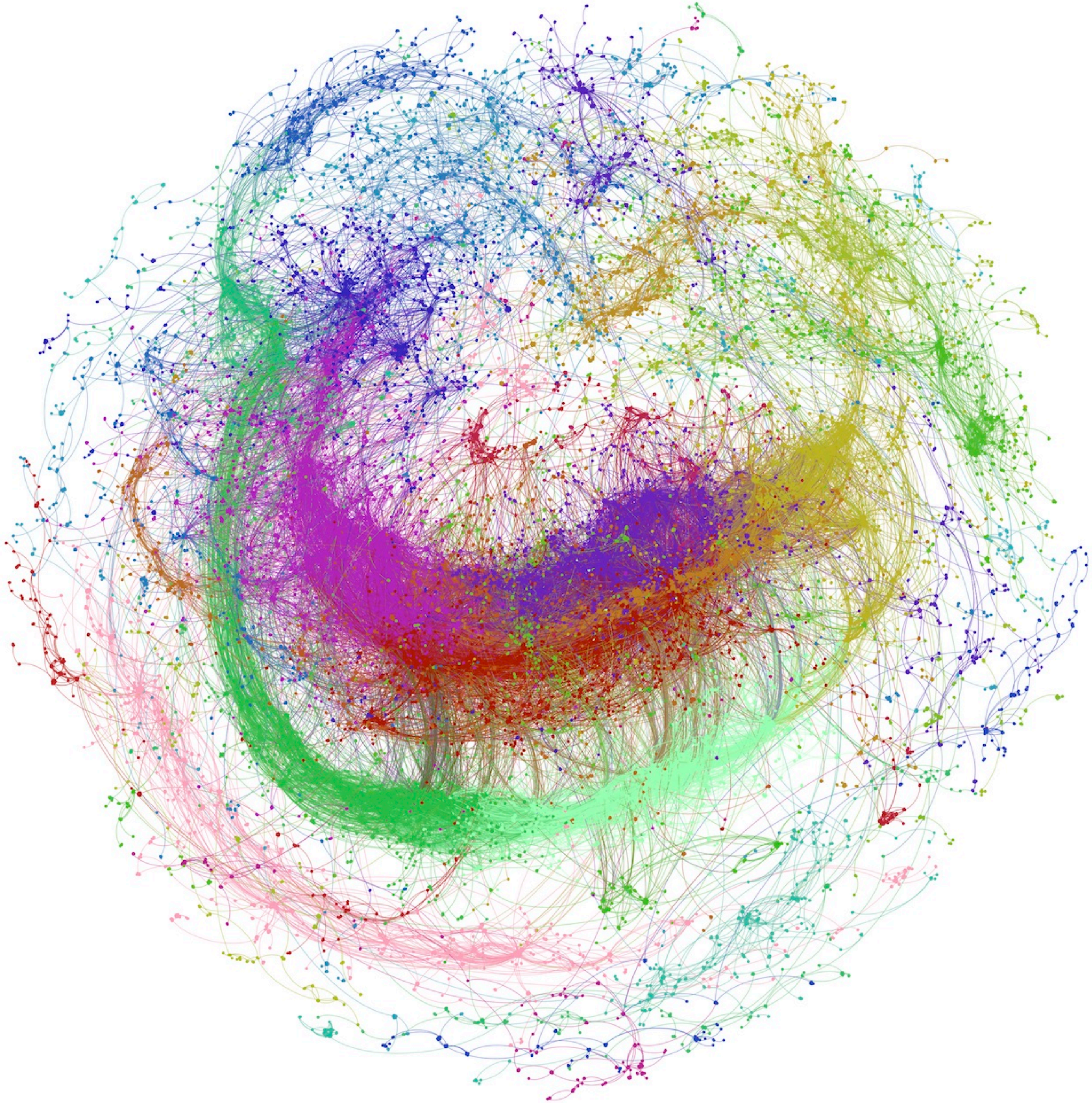
TEAM 17: LJUPCHE MILOSHESKI, ALEN CARIN, BLAGOJ MITREVSKI, MLADEN KORUNOSKI

WHAT DO THIS SERVICES HAVE IN COMMON?

RECOMMENDER SYSTEMS

- Netflix prize
  - Winning solution: Matrix Factorization (MF)
  - Year 2009
- Can we improve it?





DATA DESCRIPTION

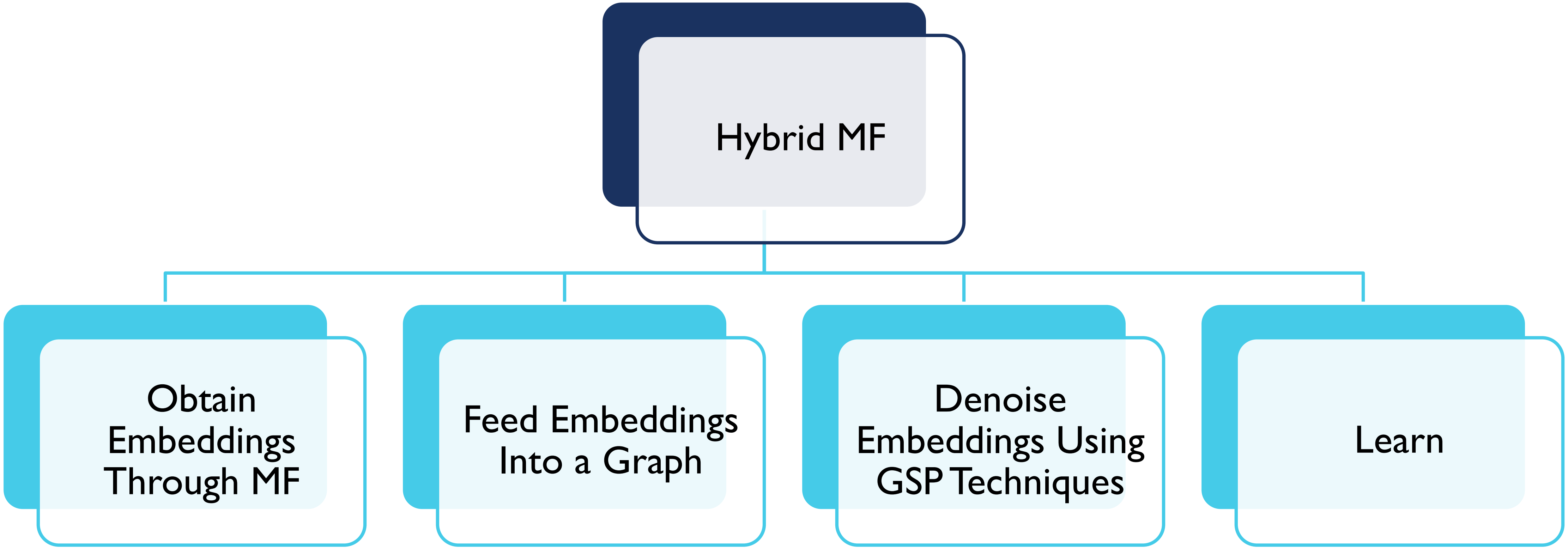
MOVIELENS 100K

grouplens

UNIVERSITY OF MINNESOTA

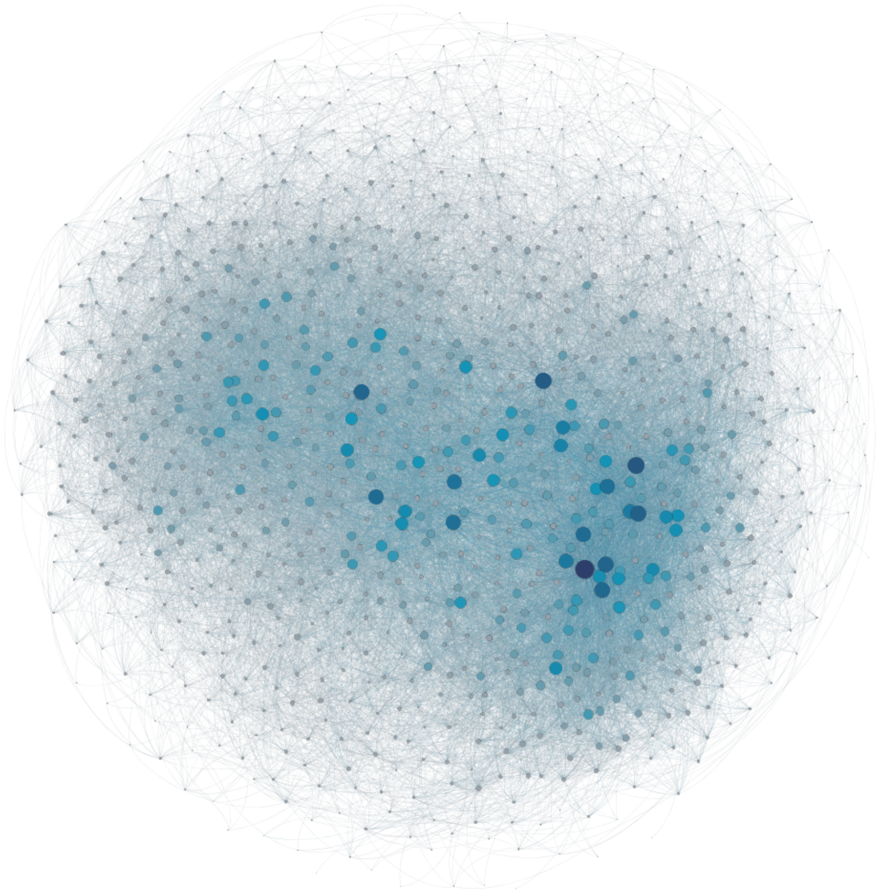
- 100.000 integer valued ratings
  - 943 users
  - 1682 movies
  - Age, gender, occupation, …
- September 19th, 1997 through April 22nd, 1998
- Cleaned
  - Without complete demographics information
  - Less than 20 ratings in total

GRAPH ANALYSIS



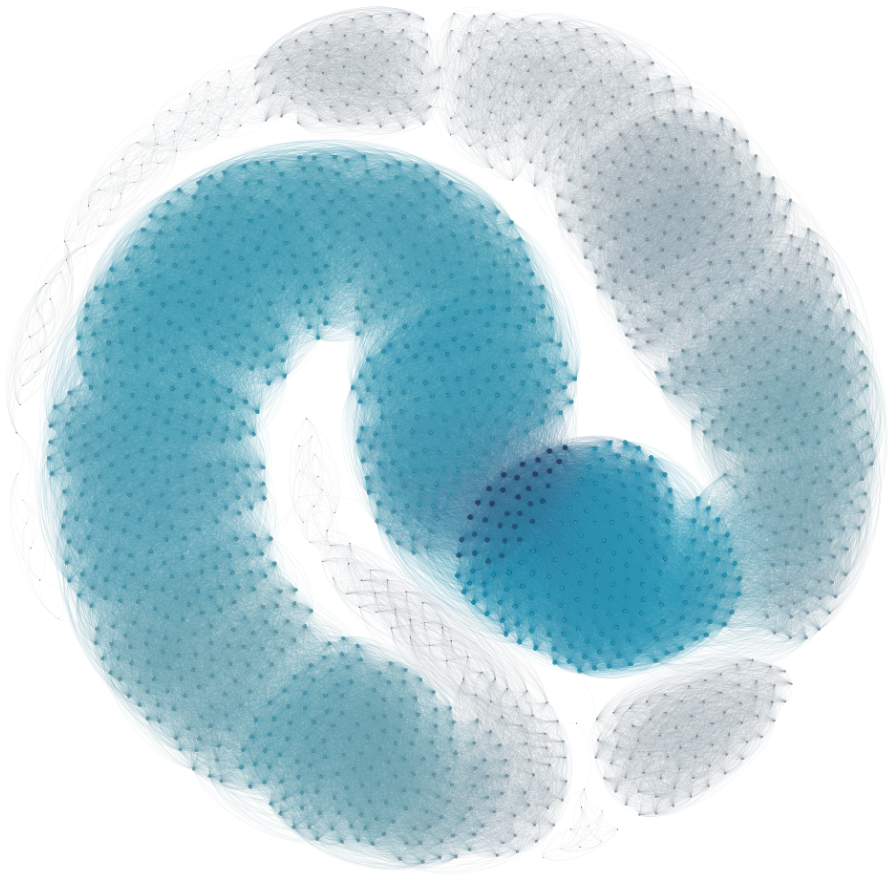
USER GRAPH

- Vector representation for each user
  - With subtracted mean
- Similarity matrix
  - Cosine similarity
- Pruning
  - Connected graph
  - Sparse (0.02%)
- No additional data needed



MOVIE GRAPH

- Assumption: *Movies with similar mean ratings are connected*
  - Sparsity (α=0.11)
- Clique if pairwise difference is no more than α
- Extreme case: clique for ratings close to 5 and most similar ratings lower than α
  - Fix: edge between most similar movies from consecutive groups



GRAPH PROPERTIES (I)

User Graph

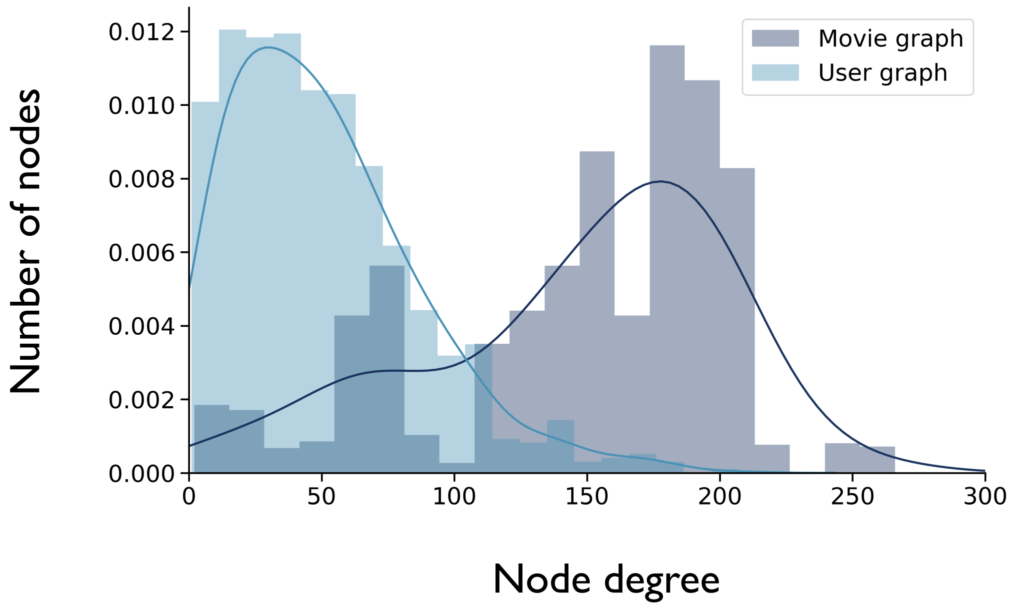
- Diameter: 5
- Degree distribution: scale-free + random (to some degree)
- Clustering coefficient: 0.25

Movie Graph

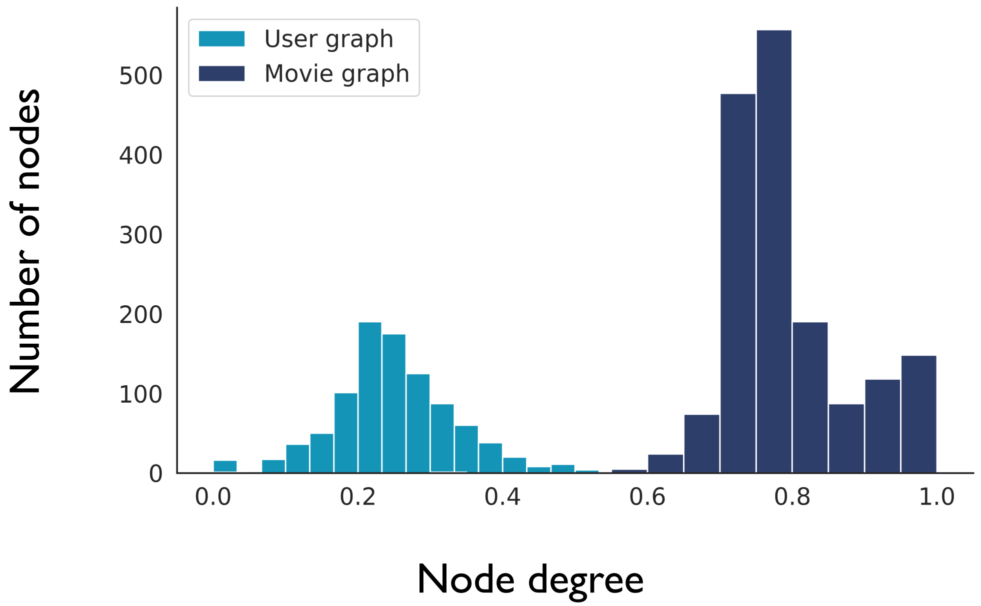
- Diameter: 39
- Degree distribution: unidentifiable
- Clustering coefficient: 0.79

GRAPH PROPERTIES (II)

Degree Distribution



Distribution of Clustering Coefficients



MODELS AND METHODS

MATRIX FACTORIZATION WITH BIAS TRICK

$$\min_{P,Q} L(P,Q) = \min_{P,Q} \|M - PQ^\intercal\|_F$$

- Representations
- Non-convex, non-identifiable
  - Local minimum
- Bias trick
  - Modification of Stochastic Gradient Descent
  - For *users* and *movies*

MODELS AND METHODS

GRAPH SIGNAL PROCESSING

$$\hat{x} = U^{\intercal} x$$

$$x = U \hat{x}$$

$$GS = U \Lambda U^{\intercal}$$

$$h(L) = U h(\Lambda) U^{\intercal}$$

- Signal
  - Per node
- Graph Fourier Transform and Graph Filters
- Graph Shift Matrix
  - Weighted Adjacency Matrix
  - Laplacian Matrix
- Vertex to spectral domain and back
- Denoising

PSEUDOCODE FOR (MF + USER) MOVIE MODEL

procedure (MF + USER) MOVIE MODEL

  $\mathbf{P}, \mathbf{Q} \leftarrow \mathit{learn\_SGD}(\mathbf{M})$

  $\mathit{outer\_iter} \leftarrow 0$

  while $\mathit{outer\_iter} \leq \mathit{filter\_max\_iter}$ **do**

    $\mathit{outer\_iter} \leftarrow \mathit{outer\_iter} + 1$

    $\mathbf{Q}^* \leftarrow \mathit{denoise}(\mathbf{Q})$

    $\mathit{inner\_iter} \leftarrow 0$

    while $\mathit{inner\_it} \leq \mathit{filter\_max\_it}$ **do**

      $\mathit{inner\_iter} \leftarrow \mathit{inner\_iter} + 1$

      $\mathbf{P}^* \leftarrow \mathit{denoise}(\mathbf{P})$

      $\hat{\mathbf{P}}, \hat{\mathbf{Q}} \leftarrow \mathit{learn\_SGD}(\mathbf{M}, \mathbf{P}^*, \mathbf{Q}^*)$

      $\mathbf{P} \leftarrow \hat{\mathbf{P}}$

    $\mathbf{Q} \leftarrow \hat{\mathbf{Q}}$

end procedure

RESULTS

- 5-fold cross-validation
- Hand-crafted low-pass filters
  - Approximated by a polynomial
- Root Mean Square Error (RMSE)

Algorithm	RMSE
Standard MF	0.914
MF with bias trick	0.906
MF + Movie filtering	0.903
MF + User filtering	0.886
MF + User & Movie	0.887
(MF + Movie) User filtering	0.882
(MF + User) Movie filtering	**0.877**

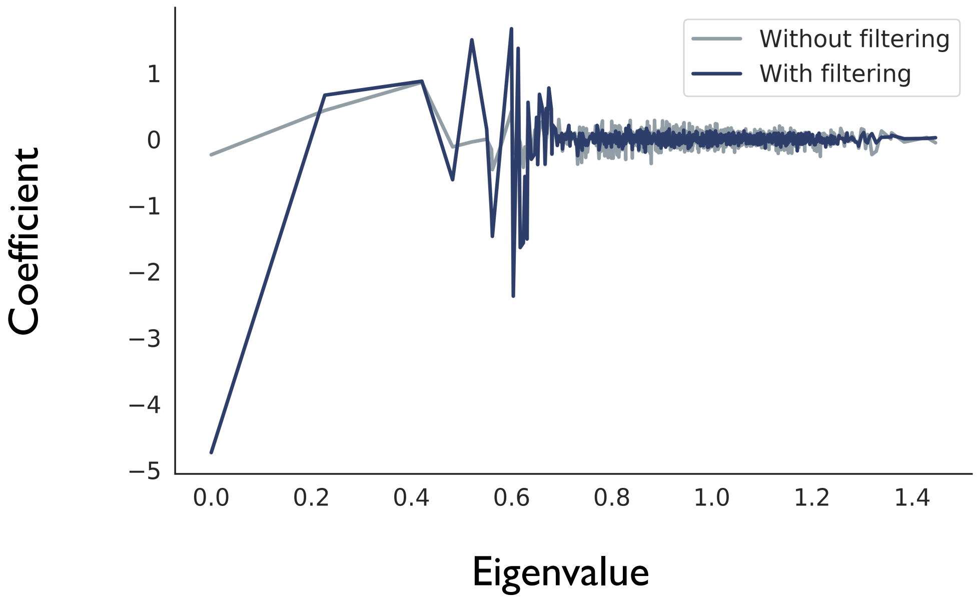
DISCUSSION (I)

- User graph has better quality
  - Movie graph has low minimum for ratings
  - Filtering does not improve much
- Our results outperform Monti et al. (RMSE: 0.905) and Berg et al. (RMSE: 0.929)
  - Floating point precision
- Similar results to Huang et al. (RMSE: 0.867)
  - Eigenvectors on the covariance matrices

DISCUSSION (II)

User Graph

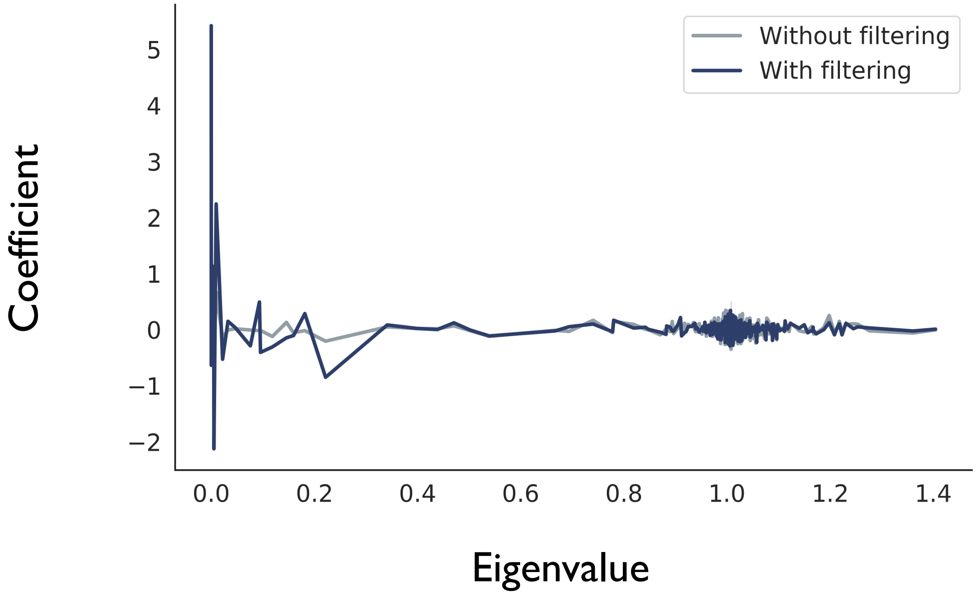
- Exploiting the graph structure
  - We could jump to better local minimum
- Coefficients with high eigenvalues are decreased
  - Decreasing noise
- Coefficients with eigenvalues around 0.6 are boosted



DISCUSSION (III)

Movie Graph

- Smooth eigenvectors are boosted to some degree
- Non-smooth are shrunk
- Still, significant amount of noise around 1
  - Escaping local minimum



CONCLUSION

1) No additional data needed
2) Independent of the recommender system problem
3) Requires at least several ratings for an item
4) CNNs can improve performance
5) Hand-crafted filters are difficult to tune
6) Learning one filter per latent dimension should give best results