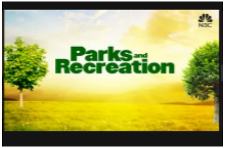
# Improving the Quality of Top-$N$ Recommendation

Evangelia Christakopoulou

Computer Science & Engineering
University of Minnesota, Twin Cities

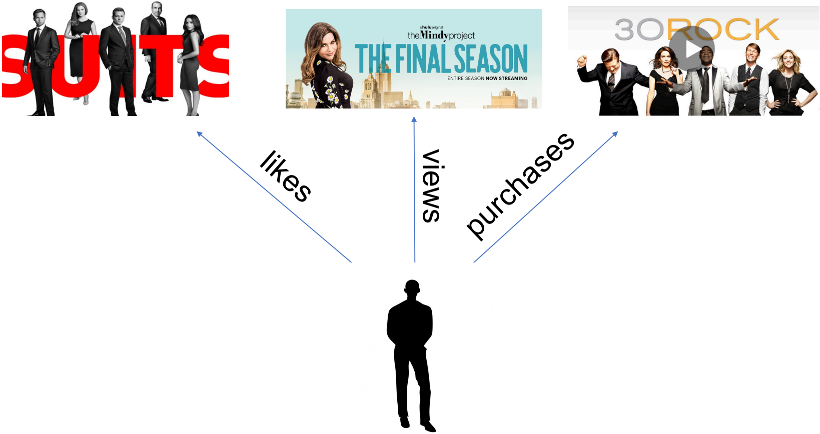
January 31st, 2018

# Recommender Systems - Introduction

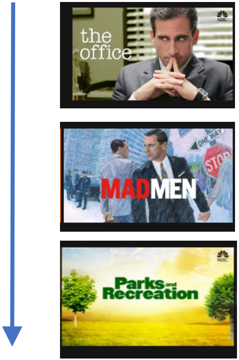


# Recommender Systems - Introduction




A good recommendation system takes into account users' needs, while going through all possible items.

# Top-$N$ Recommender Systems




Top-$N$ recommender systems recommend to the user a ranked list of $N$ items.

# Thesis Focus

Developing algorithms to
*improve the quality of top-N recommendation systems*,
utilizing implicit feedback data.

# Related Publications


- **Evangelia Christakopoulou** and George Karypis. HOSLIM: higher-order sparse linear method for top-n recommender systems. *In Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 38–49. Springer, Cham, 2014.
- **Evangelia Christakopoulou**. Moving beyond linearity and independence in top-n recommender systems. *In Proceedings of the* 8*th ACM Conference on Recommender systems*, pages 409–412. ACM, 2014.
- **Evangelia Christakopoulou** and George Karypis. Local item-item models for top-n recommendation. *In Proceedings of the 10th ACM Conference on Recommender Systems*, pages 67–74. ACM, 2016.
- David Anastasiu, **Evangelia Christakopoulou**, Shaden Smith, Mohit Sharma and George Karypis. Big Data and Recommender Systems. *In Novatica: Journal of the Spanish Computer Scientist Association*. 2016.

# Related Publications

- **Evangelia Christakopoulou**, Shaden Smith, Mohit Sharma, Alex Richards, David Anastasiu and George Karypis. Scalability and distribution of collaborative recommenders. *In Collaborative Recommendations: Algorithms, Practical Challenges and Applications*. 2018.
- **Evangelia Christakopoulou** and George Karypis. Using the error for top-$N$ recommendation. *Ready for submission*.
- **Evangelia Christakopoulou** and George Karypis. Local Latent Space Models for Top-$N$ Recommendation. *Ready for submission*.

# Notations

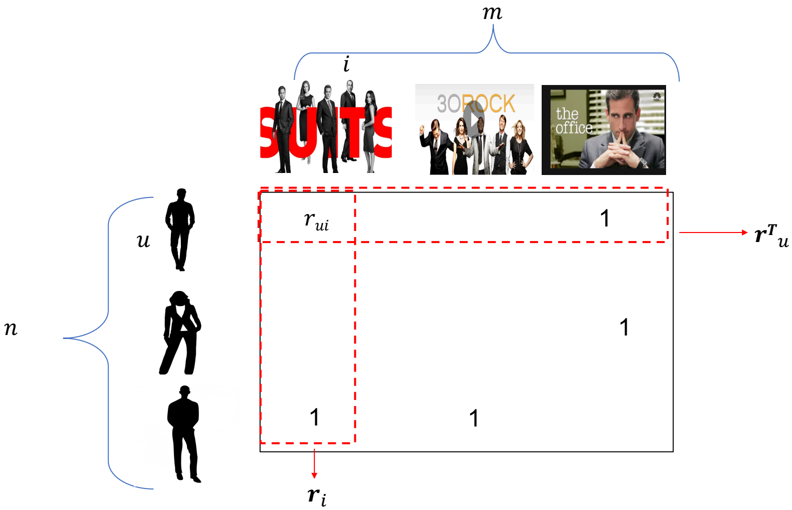
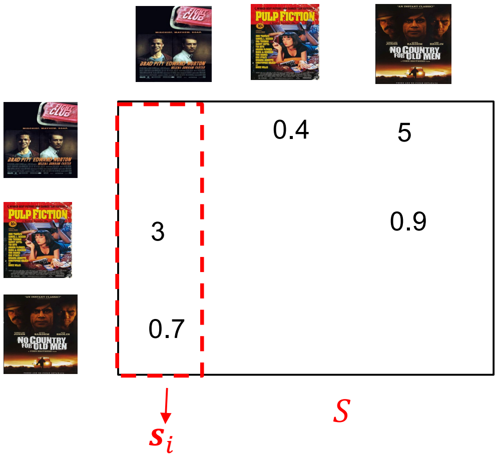



Figure: User-item implicit feedback matrix **R**.

# Nearest Neighborhood Approaches

# SLIM




$$\tilde{r}_{ui} = \mathbf{r}_u^T \mathbf{s}_i \tag{1}$$

$$\begin{aligned} \underset{\mathbf{S}}{\text{minimize}} \quad & \frac{1}{2}\sum_{u,i}(r_{ui} - \tilde{r}_{ui})^2 + \frac{\beta}{2}||\mathbf{S}||_F^2 + \lambda||\mathbf{S}||_1 \\ \text{subject to} \quad & \mathbf{S} \geq 0, diag(\mathbf{S}) = 0. \end{aligned} \tag{2}$$

# Latent Space Approaches


$$
\boldsymbol{R} \simeq
\begin{array}{c}
\begin{array}{ccc} \text{Comedy} & \text{Workplace} & \text{Awarded} \end{array} \\
\boxed{\begin{array}{ccc} 1.2 & 0.6 & 0.9 \\ 0.4 & 1.2 & -0.2 \\ 0.3 & 0.7 & 1.8 \end{array}}
\end{array}
\;
\begin{array}{c}
\boxed{\begin{array}{ccc} 1.2 & 0.7 & 1.6 \\ 1.9 & 1.7 & 0.06 \\ 2.0 & 0.02 & 0.47 \end{array}}
\end{array}
\begin{array}{l} \text{Comedy} \\ \text{Workplace} \\ \text{Awarded} \end{array}
$$

- PureSVD

$$\tilde{r}_{ui} = \mathbf{p}_u^T \boldsymbol{\Sigma}_r \mathbf{q}_i \tag{3}$$

# Outline

| Research Questions | Methods |
|---|---|
| How do we exploit higher-order sets beyond pairs, to perform top-N recommendation? | Item-based Approaches |
| How do we personalize more to individual users, beyond a global model? | Item-based Approaches |
| How do we apply a global and local model to latent space approaches? | Latent Space Approaches |
| Since missing entries are treated as zeros, recommendations come from entries with highest error. Which are its properties? | Item-based & Latent Space Approaches |

# Datasets

| Name | #Users | #Items | #Non-zeros | Density |
|---|---|---|---|---|
| synthetic | 5,000 | 1,000 | 73,597 | 1.47% |
| ml100k | 943 | 1,681 | 100,000 | 6.30% |
| bms1 | 26,667 | 496 | 116,704 | 0.88% |
| delicious | 2,989 | 2,000 | 246,430 | 4.12% |
| ctlg3 | 56,593 | 39,079 | 451,247 | 0.02% |
| retail | 85,146 | 16,470 | 905,560 | 0.06% |
| jester | 57,732 | 150 | 1,760,039 | 20.32% |
| groceries | 63,034 | 15,846 | 2,060,719 | 0.21% |
| bms-pos | 435,319 | 1,657 | 3,286,742 | 0.46% |
| flixster | 29,828 | 10,085 | 7,356,146 | 2.45% |
| ml10m | 69,878 | 10,677 | 10,000,054 | 1.34% |
| netflix | 274,036 | 17,770 | 31,756,784 | 0.65% |

# Evaluation Methodology

- Leave-one-out cross-validation.
- Performance metrics:

$$HR = \frac{\#hits}{\#users} \tag{4}$$

$$ARHR = \frac{1}{\#users} \sum_{i=1}^{\#hits} \frac{1}{p_i} \tag{5}$$

| Research Questions | Methods |
|---|---|
| How do we exploit higher-order sets beyond pairs, to perform top-N recommendation? | Item-based Approaches |
| How do we personalize more to individual users, beyond a global model? | Item-based Approaches |
| How do we apply a global and local model to latent space approaches? | Latent Space Approaches |
| Since missing entries are treated as zeros, recommendations come from entries with highest error. Which are its properties? | Item-based &<br>Latent Space Approaches |

# Identifying and exploring higher-order sets of items

# Higher-Order Sparse LInear Method (HOSLIM)

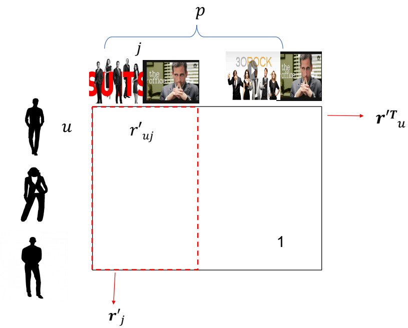



Figure: Compute the itemsets and construct $\mathbf{R}'$.

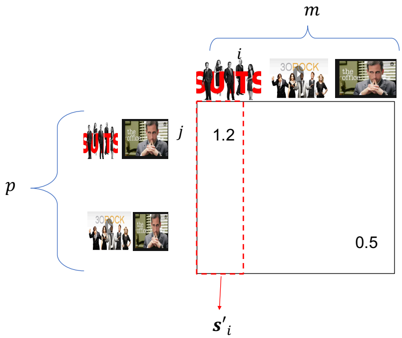


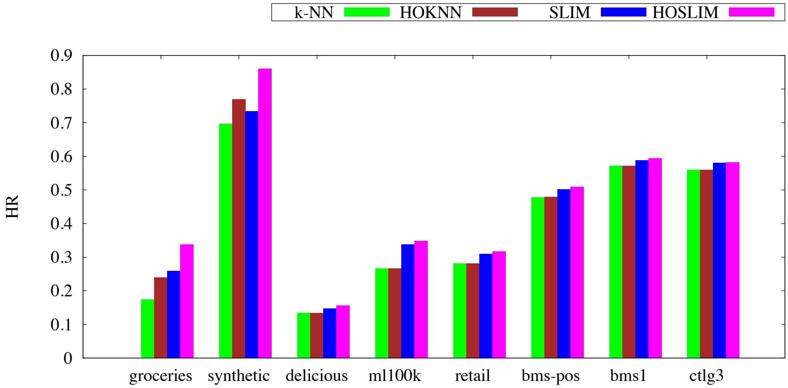

Figure: Estimate $\mathbf{S}$ and $\mathbf{S}'$ (pictured).

# Estimation of $\mathbf{S}$ and $\mathbf{S}'$

$$\tilde{r}_{ui} = \mathbf{r}_u^T \mathbf{s}_i + \mathbf{r}_u'^T \mathbf{s}_i' \tag{6}$$

$$\begin{aligned}
\underset{\mathbf{S},\mathbf{S}'}{\text{minimize}} \quad & \frac{1}{2}\sum_{u,i}(r_{ui} - \tilde{r}_{ui})^2 + \frac{\beta}{2}||\mathbf{S}||_F^2 + \frac{\beta}{2}||\mathbf{S}'||_F^2 + \lambda||\mathbf{S}||_1 + \lambda||\mathbf{S}'||_1 \\
\text{subject to} \quad & \mathbf{S} \geq 0 \\
& \mathbf{S}' \geq 0 \\
& diag(\mathbf{S}) = 0, \text{ and} \\
& s'_{ji} = 0, \text{ where}\{i \in \mathcal{I}_j\}, \forall i.
\end{aligned} \tag{7}$$

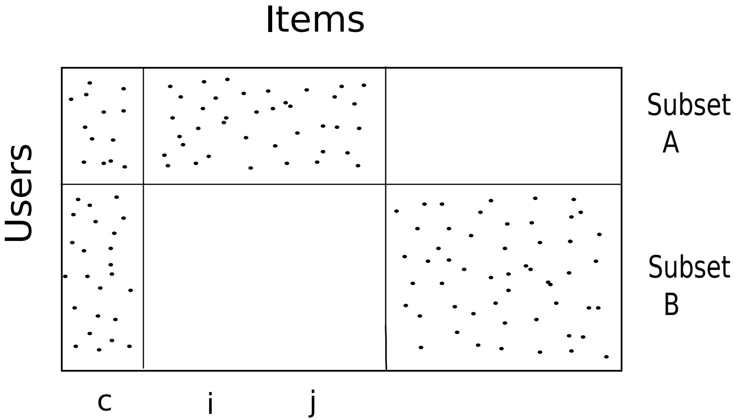
# Performance comparison



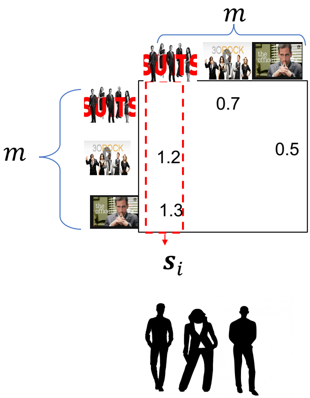

Pearson correlation coefficient between coverage and performance gains is 0.712.

| Research Questions | Methods |
|---|---|
| How do we exploit higher-order sets beyond pairs, to perform top-N recommendation? | Item-based Approaches |
| How do we personalize more to individual users, beyond a global model? | Item-based Approaches |
| How do we apply a global and local model to latent space approaches? | Latent Space Approaches |
| Since missing entries are treated as zeros, recommendations come from entries with highest error. Which are its properties? | Item-based &<br>Latent Space Approaches |

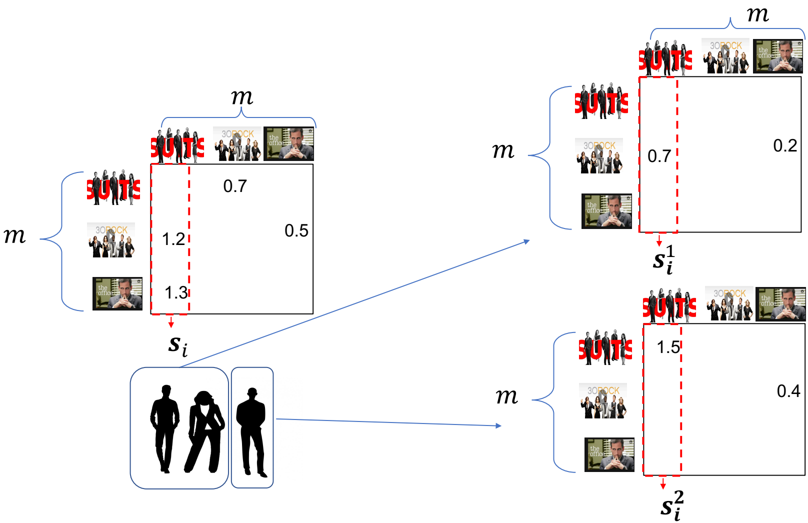
# Estimating multiple user-subset-specific item-item models



# Global and Local Sparse LInear Method (GLSLIM)



# Global and Local Sparse LInear Method (GLSLIM)

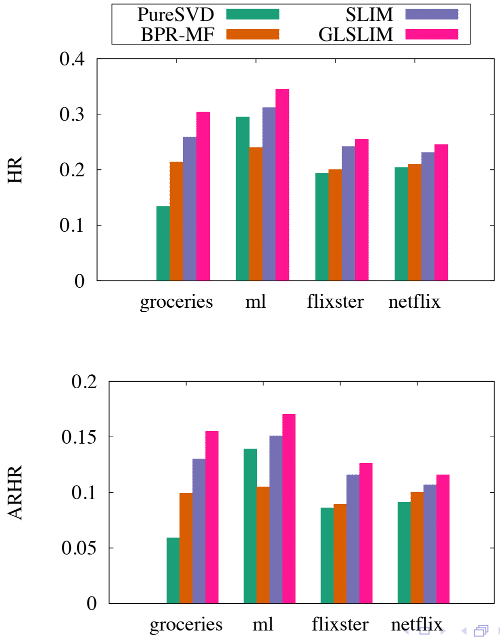


# Model Estimation

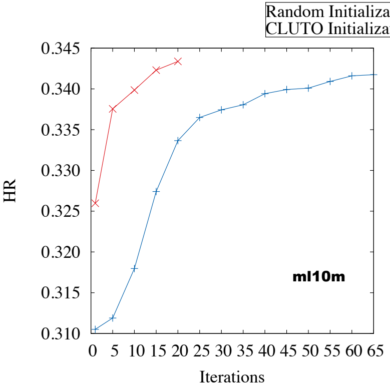
$$\tilde{r}_{ui} = \mathbf{r}_u^T (g_u \mathbf{s}_i + (1 - g_u)\mathbf{s}_i^{p_u}). \tag{8}$$

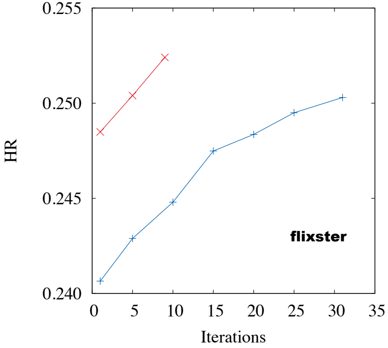
$$\begin{aligned}
\underset{\mathbf{S},\{\mathbf{S}^1,\ldots,\mathbf{S}^k\},\mathbf{p},\mathbf{g}}{\text{minimize}} \quad & \frac{1}{2}\sum_{u,i}(r_{ui} - \tilde{r}_{ui})^2 + \quad \frac{1}{2}\beta_g ||\mathbf{S}||_F^2 + \lambda_g ||\mathbf{S}||_1 + \\
& \qquad \textstyle\sum_{p_u=1}^{k} \frac{1}{2}\beta_l ||\mathbf{S}^{p_u}||_F^2 + \lambda_l ||\mathbf{S}^{p_u}||_1, \\
\text{subject to} \quad & 0 \leq g_u \leq 1, \\
& p_u \in \{1, \ldots, k\}, \forall u, \\
& \mathbf{S} \geq 0, \\
& diag(\mathbf{S}) = 0 \\
& \mathbf{S}^{p_u} \geq 0, \qquad \forall p_u \in \{1, \ldots, k\} \\
& diag(\mathbf{S}^{p_u}) = 0, \qquad \forall p_u \in \{1, \ldots, k\}
\end{aligned} \tag{9}$$

# Performance against competing approaches



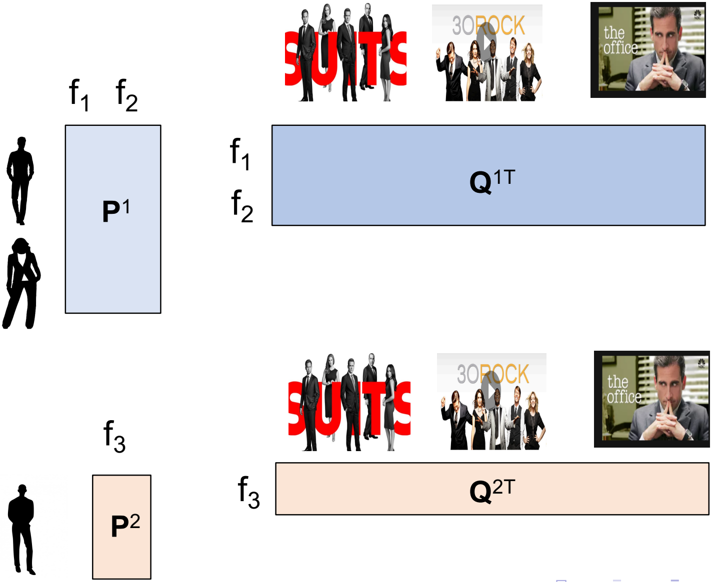
# Initializing with random user subsets



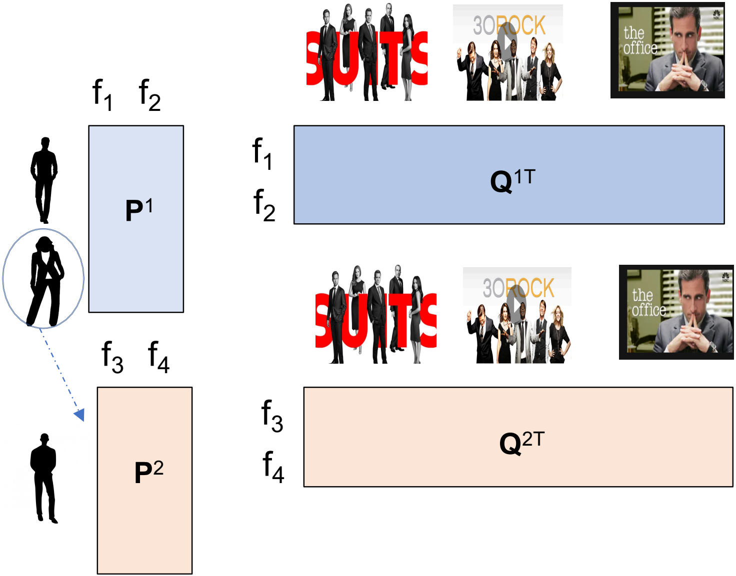



| Research Questions | Methods |
|---|---|
| How do we exploit higher-order sets beyond pairs, to perform top-N recommendation? | Item-based Approaches |
| How do we personalize more to individual users, beyond a global model? | Item-based Approaches |
| How do we apply a global and local model to latent space approaches? | Latent Space Approaches |
| Since missing entries are treated as zeros, recommendations come from entries with highest error. Which are its properties? | Item-based &<br>Latent Space Approaches |

# Varying Global and Local Singular Value Decomposition (vGLSVD)



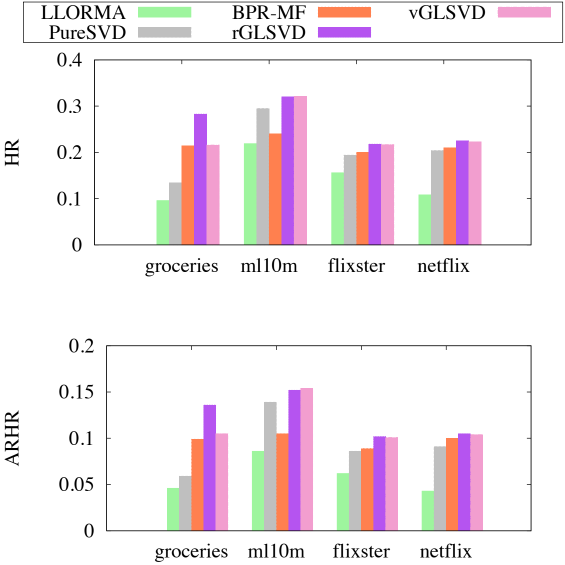
# Refined Global and Local Singular Value Decomposition (rGLSVD)



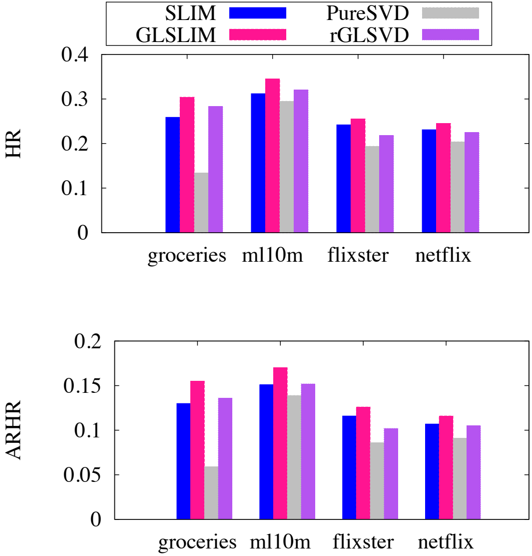
# vGLSVD & rGLSVD

$$\tilde{r}_{ui} = g_u \mathbf{p}_u^T \boldsymbol{\Sigma}_r \mathbf{q}_i + (1 - g_u) \mathbf{p}_u^{cT} \boldsymbol{\Sigma}_r^c \mathbf{q}_i^c \tag{10}$$

# Performance against competing approaches



# Comparing standard global approaches with global & local approaches



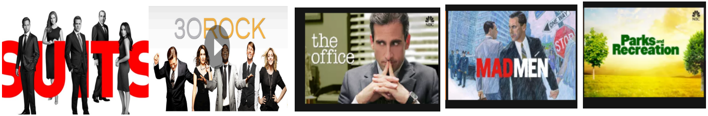
# Comparing standard global approaches with global & local approaches

Table: The training time for *ml10m* dataset with 5 clusters on one node ($=24$ cores).

| Method | mins |
|---|---|
| rGLSVD | 9.3 |
| SLIM | 39.9 |
| SLIM-warm | 2.6 |
| GLSLIM | 199.2 |
| GLSLIM-warm | 53.7 |

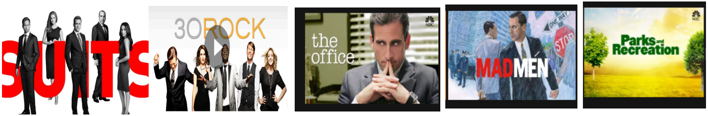
| Research Questions | Methods |
| --- | --- |
| How do we exploit higher-order sets beyond pairs, to perform top-N recommendation? | Item-based Approaches |
| How do we personalize more to individual users, beyond a global model? | Item-based Approaches |
| How do we apply a global and local model to latent space approaches? | Latent Space Approaches |
| Since missing entries are treated as zeros, recommendations come from entries with highest error. Which are its properties? | Item-based & Latent Space Approaches |

# Identifying the error properties of the top-$N$ recommendation models



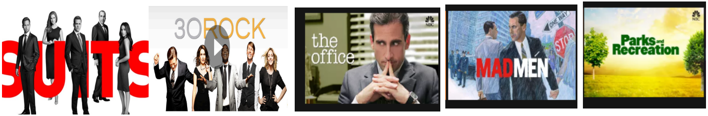

| | | | | |
|---|---|---|---|---|
| 1 | 1 | ? | ? | ? |
| ? | 1 | ? | ? | ? |
| ? | ? | 1 | ? | 1 |
| ? | ? | ? | 1 | ? |
| 1 | 1 | ? | ? | ? |

# Identifying the error properties of the top-$N$ recommendation models




| | | | | |
|---|---|---|---|---|
| 1 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 1 |
| 0 | 0 | 0 | 1 | 0 |
| 1 | 1 | 0 | 0 | 0 |

# Identifying the error properties of the top-$N$ recommendation models




| | | | | |
|---|---|---|---|---|
| | | 0.3 | 0.02 | 0.28 |
| 0.01 | | 0.8 | 0 | 0.77 |
| 0 | 0.1 | | 0 | |
| 0.9 | 0.01 | 0.88 | | 0.88 |
| | | 0.3 | 0.02 | 0.28 |

# Top-$N$ Recommendation Error

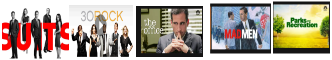
- SLIM models:

$$\dot{e}_{ui} = \begin{cases} \mathbf{r}_u^T \mathbf{s}_i, & \text{if } r_{ui} = 0 \\ 0, & \text{if } r_{ui} \neq 0. \end{cases} \tag{11}$$

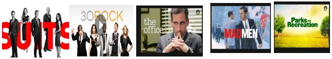
- PureSVD models:

$$\dot{e}_{ui} = \begin{cases} \mathbf{p}_u^T \mathbf{\Sigma}_r \mathbf{q}_i, & \text{if } r_{ui} = 0 \\ 0, & \text{if } r_{ui} \neq 0. \end{cases} \tag{12}$$

# Hypothesis




| | | | | | |
|---|---|---|---|---|---|
| $r_u$ | 1 | 1 | | | |
| | | 1 | | | |
| | | | 1 | | 1 |
| | | | | 1 | |
| $r_v$ | 1 | 1 | | | |




| | | | | | |
|---|---|---|---|---|---|
| $\dot{e}_u$ | | | 0.3 | 0.02 | 0.28 |
| | | 1 | | | |
| | | | 1 | | 1 |
| | | | | 1 | |
| $\dot{e}_v$ | | | 0.3 | 0.02 | 0.28 |

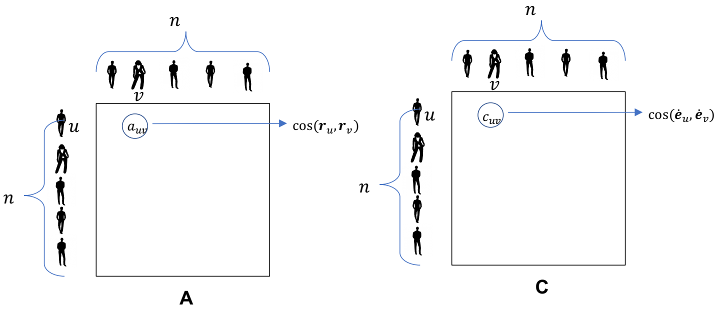
# Hypothesis

| SUITS | 30 ROCK | the office | MAD MEN | Parks and Recreation |
|---|---|---|---|---|
| 1 | 1 | | | |
| | 1 | | | |
| | | 1 | | 1 |
| | | | 1 | |
| 1 | 1 | | | |
| | | $\boldsymbol{r}_i$ | | $\boldsymbol{r}_j$ |

| SUITS | 30 ROCK | the office | MAD MEN | Parks and Recreation |
|---|---|---|---|---|
| 1 | 1 | 0.3 | | 0.28 |
| | 1 | 0.8 | | 0.77 |
| | | | | |
| | | 0.88 | 1 | 0.88 |
| 1 | 1 | 0.3 | | 0.28 |
| | | $\dot{\boldsymbol{e}}_i$ | | $\dot{\boldsymbol{e}}_j$ |

# Verifying that similar users should have similar error

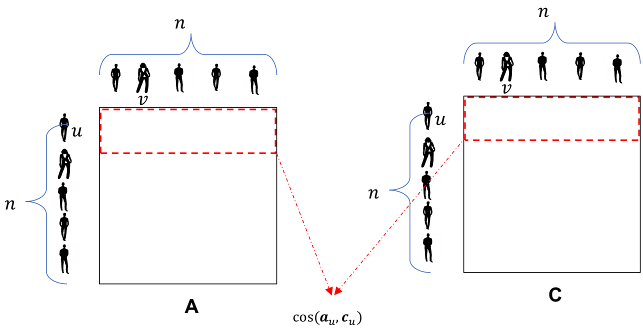
- Create the user similarity matrix **A** of rating similarities.
- Create the user similarity matrix **C** of error similarities.



# Verifying that similar users should have similar error

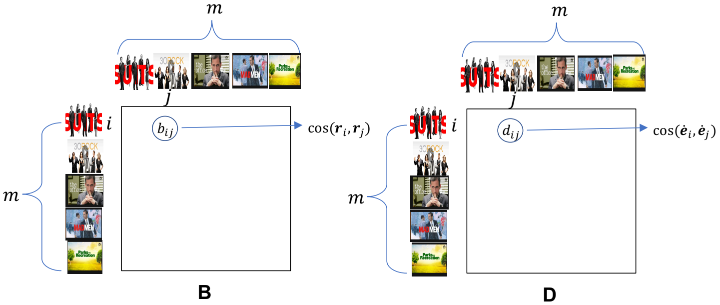
- Create the user similarity matrix **A** of rating similarities.
- Create the user similarity matrix **C** of error similarities.
- Compute the similarity of the rating-based and error-based representations:

$$\text{User Rating.Error Similarity} = \frac{\sum_{u=1}^{n} cos(\mathbf{a}_u, \mathbf{c}_u)}{n}. \qquad (13)$$



# Verifying that similar items should have similar error

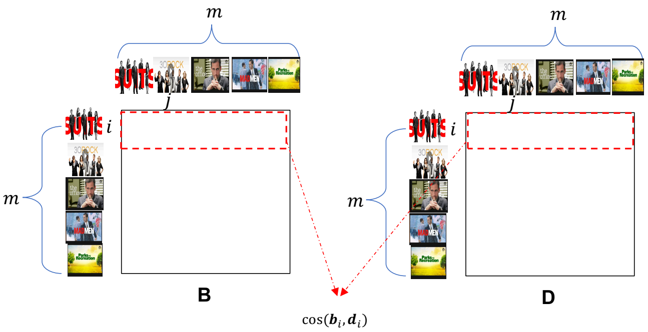
- Create the item similarity matrix **B** of rating similarities.
- Create the item similarity matrix **D** of error similarities.



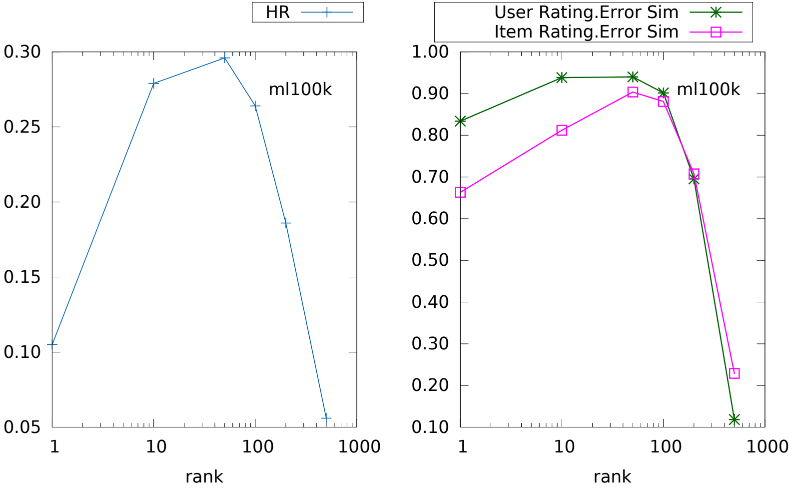
# Verifying that similar items should have similar error

- Create the item similarity matrix **B** of rating similarities.
- Create the item similarity matrix **D** of error similarities.
- Compute the similarity of the rating-based and error-based representations:

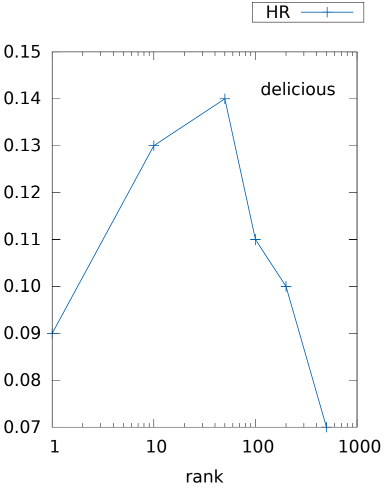
$$\text{Item Rating.Error Similarity} = \frac{\sum_{i=1}^{m} cos(\mathbf{b}_i, \mathbf{d}_i)}{m}. \tag{14}$$



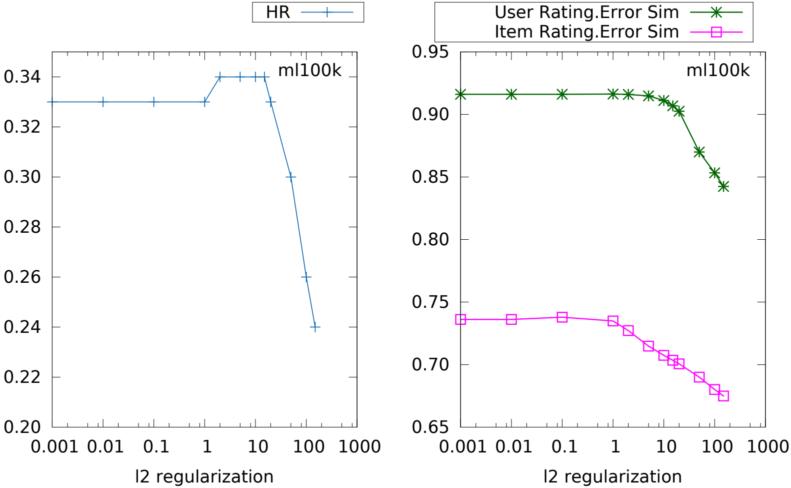
# Testing the hypothesis on PureSVD models



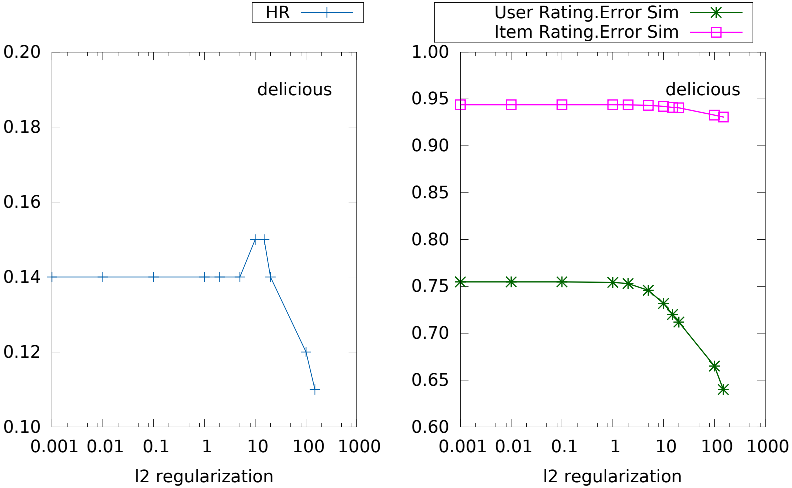
# Testing the hypothesis on PureSVD models



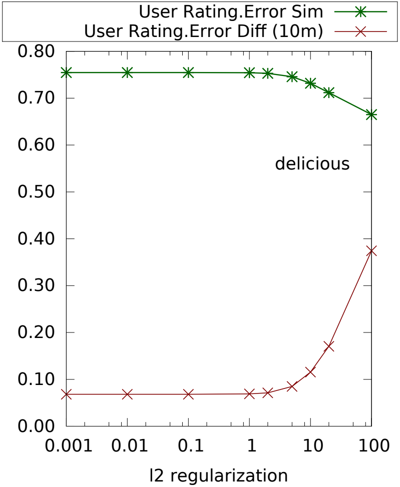
# Testing the hypothesis on SLIM models

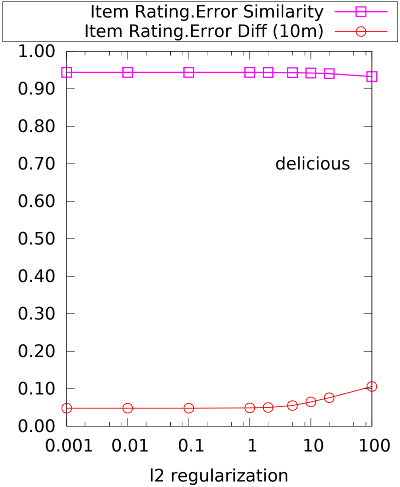


# Testing the hypothesis on SLIM models



# Rating.Error Representation



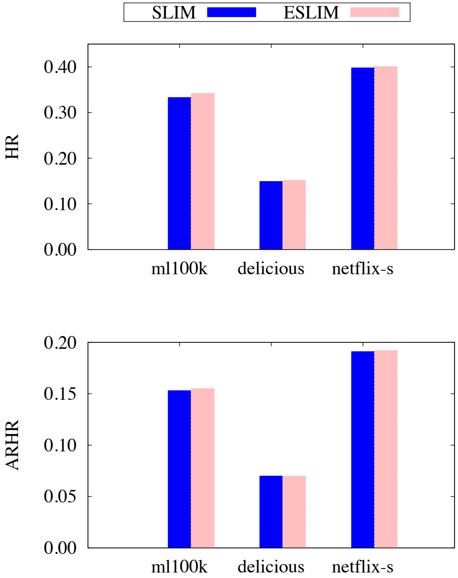



# Error-constrained Sparse LInear Method (ESLIM)

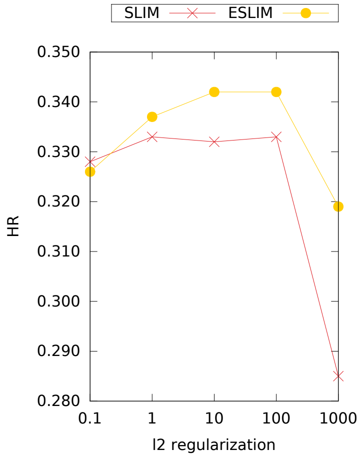
$$\tilde{r}_{ui} = \mathbf{r}_u^T \mathbf{s}_i \tag{15}$$

$$\begin{aligned} \underset{\mathbf{S}}{\text{minimize}} \quad & \frac{1}{2} \sum_{u,i} (r_{ui} - \tilde{r}_{ui})^2 + \frac{\beta}{2} ||\mathbf{S}||_F^2 + \frac{\lambda_u}{2} ||\mathbf{C} - \mathbf{A}||_F^2 + \frac{\lambda_i}{2} ||\mathbf{D} - \mathbf{B}||_F^2 \\ \text{subject to} \quad & \mathbf{S} \geq 0 \\ & diag(\mathbf{S}) = 0. \end{aligned} \tag{16}$$

# Performance of ESLIM vs SLIM



# Performance of ESLIM vs SLIM



# Conclusion

- Utilizing higher-order sets improves performance, when such sets are present.
- Global and Local approaches outperform standard global models.
- Insight into how error at the missing entries correlates with performance.

# Future Research

- Combination of vGLSD and rGLSVD (with regularized SVD).
- Error-constrained approach with underlying latent space model.

# Acknowledgements

Thank you all!

# Appendix

Appendix Slides

# HOSLIM datasets

Table: The average basket size of datasets we evaluated HOSLIM on.

| Name | Average Basket Size |
|---|---:|
| groceries | 32.69 |
| synthetic | 14.72 |
| delicious | 82.45 |
| ml100k | 106.04 |
| retail | 10.64 |
| bms-pos | 7.55 |
| bms1 | 4.38 |
| ctlg3 | 7.97 |

# Verifying the existence of higher-order relations

- Found all frequent itemsets.
- Computed quality metrics:

$$dependency\_max = \frac{P(ABC)}{\max(P(AB)P(C), P(AC)P(B), P(BC)P(A))}, \tag{17}$$

and

$$dependency\_min = \frac{P(ABC)}{\min(P(AB)P(C), P(AC)P(B), P(BC)P(A))}. \tag{18}$$

- Selected itemsets that have quality metrics exceeding threshold.

# Coverage

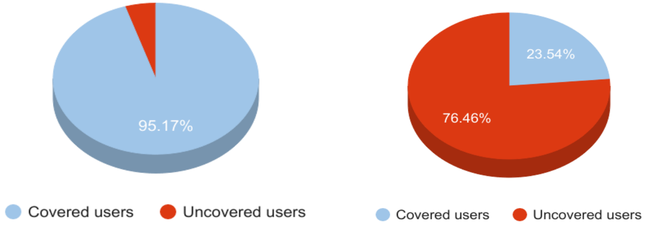



Figure: Coverage by affected users.

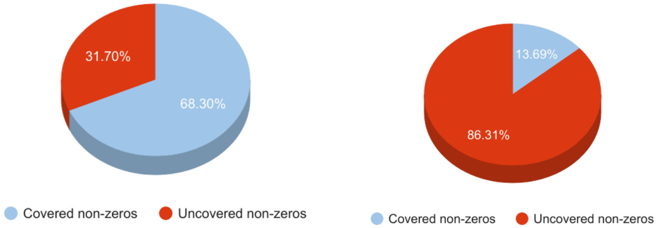
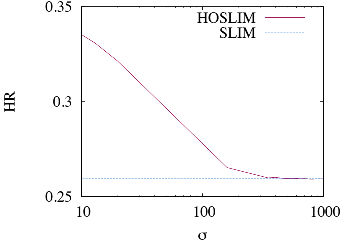
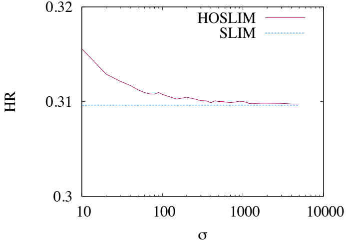



Figure: Coverage by affected non-zeros.

# Sensitivity to the support of the itemsets



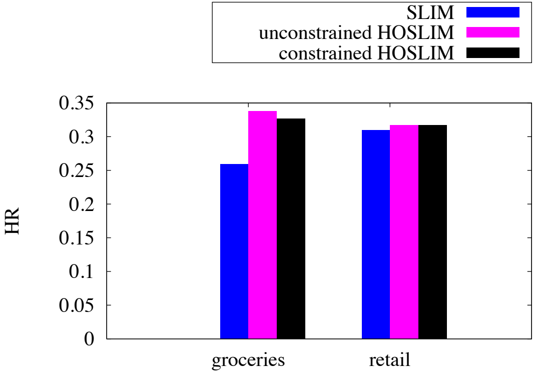

Groceries dataset.



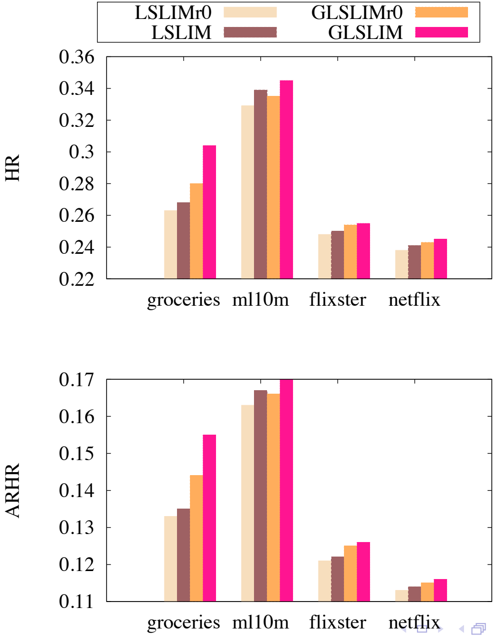

Retail dataset.

# Efficient recommendation by controlling the complexity

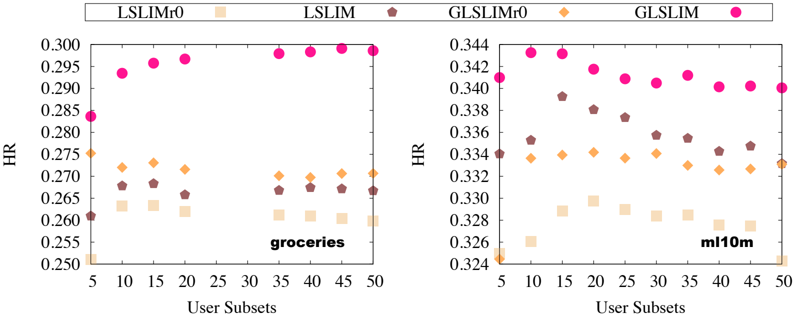
$$nnz(\mathbf{S}'_{HOSLIM}) + nnz(\mathbf{S}_{HOSLIM}) \leq 2nnz(\mathbf{S}_{SLIM}). \qquad (19)$$



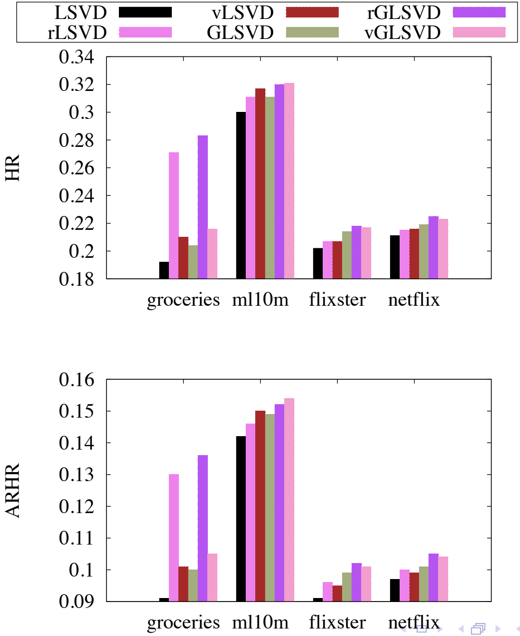
# Comparison among proposed approaches



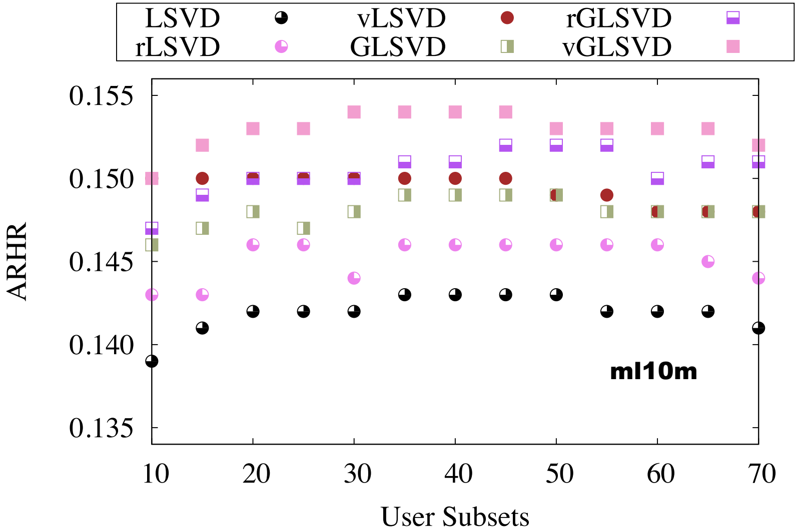
# Sensitivity on the number of Clusters



# Comparison among proposed approaches



# Sensitivity on the number of user subsets



$$\frac{\dot{\mathbf{e}}_u^T \dot{\mathbf{e}}_v}{||\dot{\mathbf{e}}_u||^T ||\dot{\mathbf{e}}_u||} = \frac{\mathbf{r}_u^T \mathbf{r}_v}{||\mathbf{r}_u||||\mathbf{r}_u||} \frac{\sum_{i \in \mathcal{N}_u \cap \mathcal{N}_v} ||\mathbf{s}_i||^2}{\sqrt{\sum_{i \in \mathcal{N}_u} ||\mathbf{s}_i||^2 \sum_{i \in \mathcal{N}_v} ||\mathbf{s}_i||^2}} \tag{20}$$