Graph Signal Sampling for Inductive One-Bit Matrix Completion: a Closed-form Solution

Chao Chen*, Haoyu Geng, Gang Zeng, Zhaobin Han,
Hua Chai, Xiaokang Yang, Junchi Yan

ICLR

SHANGHAI JIAO TONG UNIVERSITY

滴滴

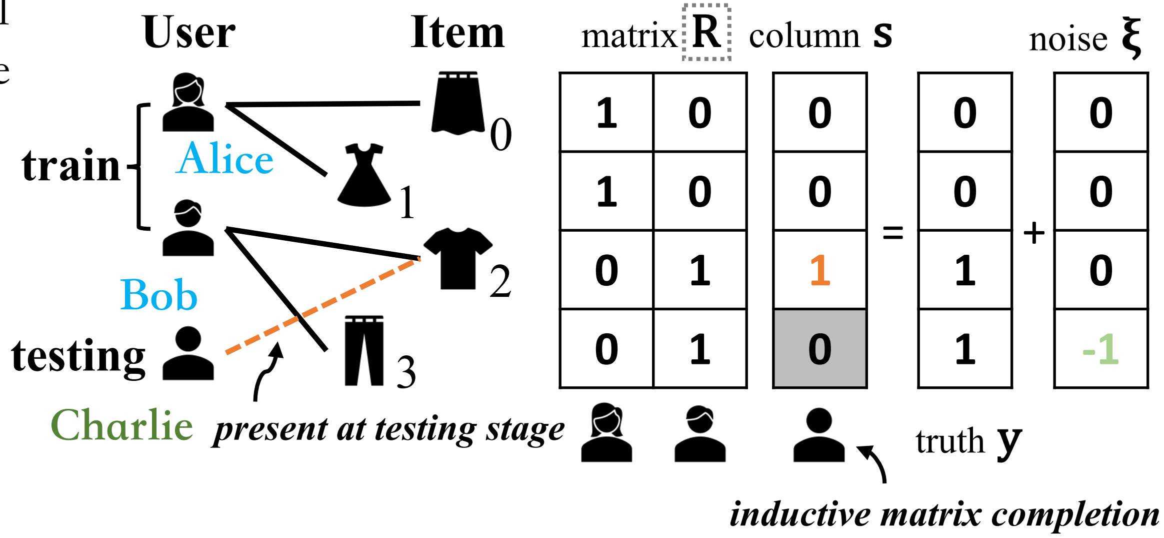
Introduction

❖ Motivation

In recommender systems, for each user the goal is to provide a list of items (e.g., products, movies etc.) based on her historical records (e.g., click, favorite or buy). In the majority of time, the system has the information of the users during the training.

❖ What if we need to serve new users?

This motivates the problem of the Inductive One-bit Matrix Completion: given a set of observations Ω_+ consisting only of ones but no zeros, the goal is to recover the underlying vector $\boldsymbol{y}$ from Ω_+.



❖ Data Modelling

Let $\boldsymbol{s}$ denote a user's history (e.g., Charlie), where $\boldsymbol{s}_i = 1$ only when $i \in \Omega_+$, then

$$\boldsymbol{s} = \boldsymbol{y} + \boldsymbol{\xi}\ ,$$

where $\boldsymbol{\xi}$ is the *discrete* noise that flips ones to zeros.

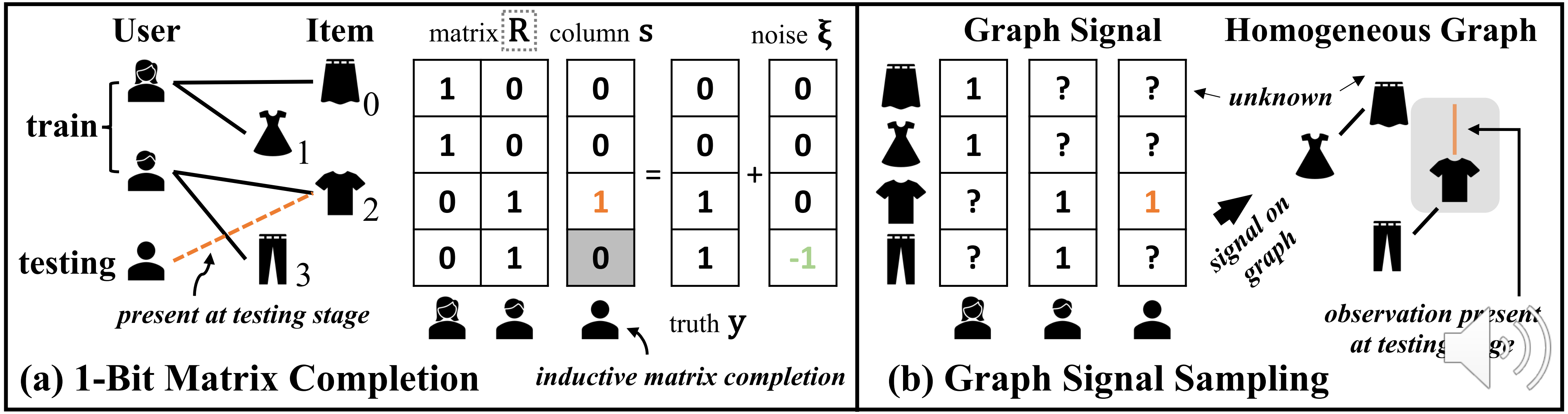
GSIMC：Closed Formed Solution for Inductive Learning

❖ A Graph Signal Sampling Perspective

1. **Graph Definition.** Let $\boldsymbol{R}$ denotes an item-user rating matrix, then we can have an item-item graph, for example,

$$\boldsymbol{A} = \boldsymbol{R}\boldsymbol{D}_u^{-1}\boldsymbol{R}^T$$

2. **Graph Signal Definition.** Recall that $\boldsymbol{s}$ signifies a user's history, it can be regarded as the values residing on the item vertices, namely a graph signal.



GSIMC：Closed Formed Solution for Inductive Learning (cont.)

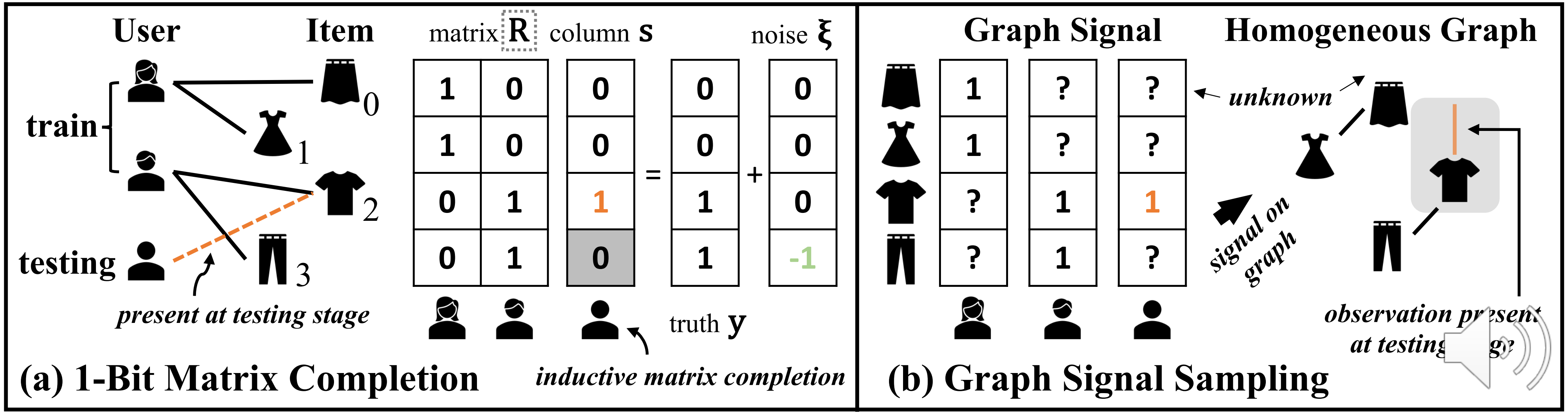
❖ A Graph Signal Sampling Perspective (cont.)

3. **Graph Signal Sampling Formulation.** Let $\boldsymbol{L}$ denote the graph Laplacian matrix, our goal is to recover the ground-truth $\boldsymbol{y}$ from the observations on a graph vertex subset, namely $\boldsymbol{s}$.

$$\min_{\boldsymbol{f}} \frac{1}{2}\langle \boldsymbol{f}, K(\boldsymbol{L})\boldsymbol{f}\rangle + \frac{\varphi}{2}\|\boldsymbol{f}-\boldsymbol{s}\|^2$$

where K(■) represents the kernel function on graph. This has a closed-formed solution:

$$\hat{\boldsymbol{y}} = (\boldsymbol{I} + K(\boldsymbol{L})/\varphi)^{-1}\boldsymbol{s}$$



GSIMC：Closed Formed Solution for Inductive Learning (cont.)

❖ A Graph Signal Sampling Perspective (cont.)

Theorem 4 (Error Analysis, extension of Theorem 1.1 in (Pesenson, 2009)). *Given $R(\lambda)$ with $\lambda \leq R(\lambda)$ on graph $\mathcal{G} = (V, E)$, assume that $\Omega^c = V - \Omega$ admits the Poincare inequality $\| \phi \| \leq \Lambda \| \L\phi \|$ for any $\phi \in L_2(\Omega^c)$ with $\Lambda > 0$, then for any $\mathbf{y} \in \mathrm{PW}_\omega(\mathcal{G})$ with $0 < \omega \leq R(\omega) < 1/\Lambda$,*

$$\| \mathbf{y} - \hat{\mathbf{y}}_k \| \leq 2\Big(\Lambda R(\omega)\Big)^k \| \mathbf{y} \| \quad \text{and} \quad \mathbf{y} = \lim_{k\to\infty} \hat{\mathbf{y}}_k \tag{8}$$

where k is a pre-specified hyperparameter and $\hat{\mathbf{y}}_k$ is the solution of Eq. (5) with $\epsilon = 0$.

Theorem 5 (Error Analysis, with label noise). *Suppose that $\boldsymbol{\xi}$ is the random noise with flip rate ρ, and positive $\lambda_1 \leq \cdots \leq \lambda_n$ are eigenvalues of Laplacian $\mathbf{\L}$, then for any function $\mathbf{y} \in \mathrm{PW}_\omega(\mathcal{G})$,*

$$\mathbb{E}\Big[\mathrm{MSE}(\mathbf{y}, \hat{\mathbf{y}})\Big] \leq \frac{C_n^2}{n}\Big(\frac{\rho}{R(\lambda_1)(1 + R(\lambda_1)/\varphi)^2} + \frac{1}{4\varphi}\Big), \tag{9}$$

where $C_n^2 = R(\omega) \| \mathbf{y} \|^2$, φ is the regularization parameter and $\hat{\mathbf{y}}$ is defined in Eq. (7).

ICLR

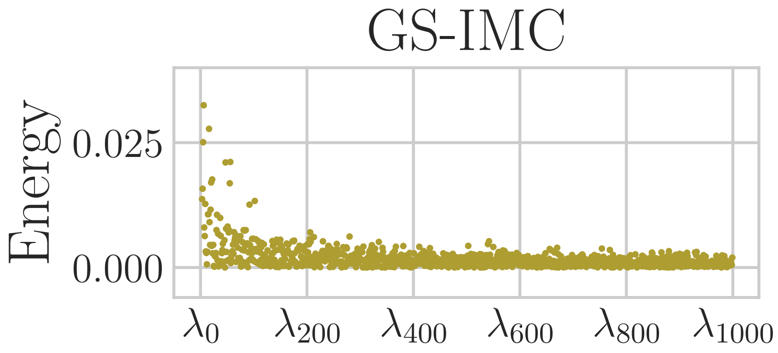
BGSIMC : Prediction-Correction Algorithm for Online Learning

❖ **How to update the model when new data comes?**

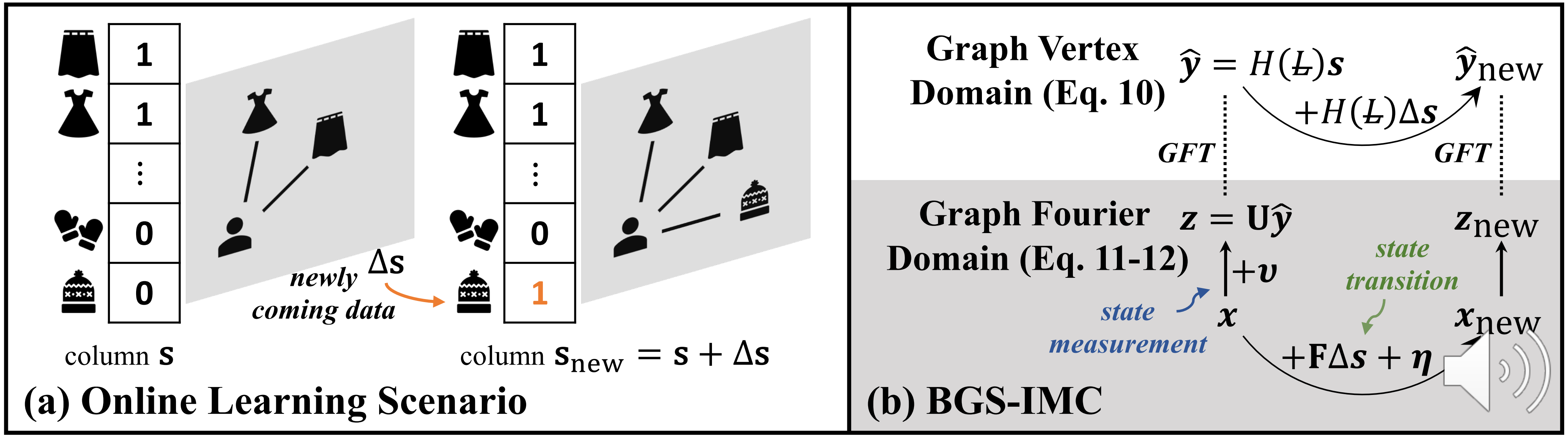
1. **Data Modeling.** We consider the problem in a dynamic state-space form:

$$\boldsymbol{x}_{\text{new}} = \boldsymbol{x} + \boldsymbol{F}\Delta\boldsymbol{s} + \boldsymbol{\eta}$$
$$\boldsymbol{z}_{\text{new}} = \boldsymbol{x}_{\text{new}} + \boldsymbol{\nu}$$

where $\boldsymbol{z}_{\text{new}}$ is a measure of the new user state $\boldsymbol{x}_{\text{new}}$, $\boldsymbol{x}$ is the user state of last time and $\Delta\boldsymbol{s}$ is the newly coming data (e.g., buying a hat in the example).



How $\boldsymbol{z}_{\text{new}}$ looks like in GSIMC?



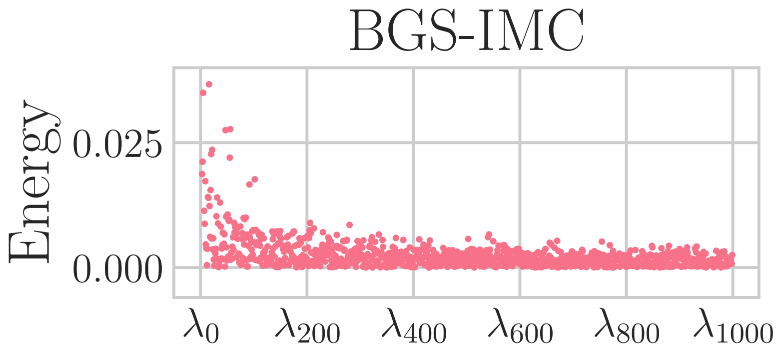
BGSIMC : Prediction-Correction Algorithm for Online Learning (cont.)

❖ How to update the model when new data comes? (cont.)

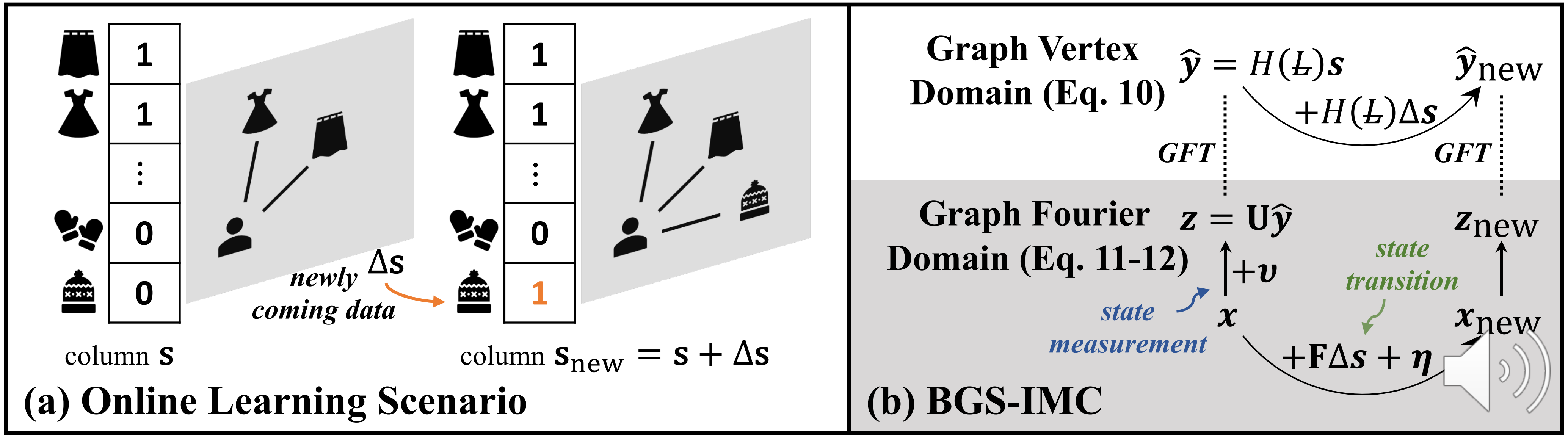
2. **Kalman filtering.** We propose a prediction-correction update algorithm:

$$\hat{\boldsymbol{x}}_{\text{new}} = \overline{\boldsymbol{x}}_{\text{new}} + \boldsymbol{K}(\boldsymbol{z}_{\text{new}} - \overline{\boldsymbol{x}}_{\text{new}})$$
$$\boldsymbol{P}_{\text{new}} = (\boldsymbol{I} - \boldsymbol{K})\overline{\boldsymbol{P}}_{\text{new}}(\boldsymbol{I} - \boldsymbol{K})^T$$
$$\boldsymbol{K} = \overline{\boldsymbol{P}}_{\text{new}}(\overline{\boldsymbol{P}}_{\text{new}} + \boldsymbol{\Sigma}_v)^{-1}$$

where $\overline{\boldsymbol{x}}_{\text{new}} = \hat{\boldsymbol{x}} + \boldsymbol{F}\Delta\boldsymbol{s}$ and $\overline{\boldsymbol{P}}_{\text{new}} = \boldsymbol{P} + \boldsymbol{\Sigma}_\eta$.



How $\boldsymbol{z}_{\text{new}}$ looks like in BGSIMC?



Experiments: Accuracy Comparison

Table 2: **Hit-Rate** results against the baselines for inductive top-N ranking. Note that SGMC (Chen et al., 2021) is a special case of our method using the cut-off regularization, and MRFCF (Steck, 2019) is the full rank version of our method with (one-step) random walk regularization. The standard errors of the ranking metrics are **less than 0.005** for all the three datasets.

	Koubei, Density=0.08%			Tmall, Density=0.10%			Netflix, Density=1.41%		
Model	H@10	H@50	H@100	H@10	H@50	H@100	H@10	H@50	H@100
IDCF* (Wu et al., 2021)	0.14305	0.20335	0.24285	0.16100	0.27690	0.34573	0.08805	0.19788	0.29320
IDCF+GAT (Veličković et al., 2017)	0.19715	0.26440	0.30125	**0.20033**	0.32710	0.39037	0.08712	0.19387	0.27228
IDCF+GraphSAGE (Hamilton et al., 2017)	0.20600	0.27225	0.30540	0.19393	0.32733	0.39367	0.08580	0.19187	0.26972
IDCF+SGC (Wu et al., 2019)	0.20090	0.26230	0.30345	0.19213	0.32493	0.38927	0.08062	0.18080	0.26720
IDCF+ChebyNet (Defferrard et al., 2016)	0.20515	0.28100	0.32385	0.18163	0.32017	0.39417	0.08735	0.19335	0.27470
IDCF+ARMA (Bianchi et al., 2021)	0.20745	0.27750	0.31595	0.17833	0.31567	0.39140	0.08610	0.19128	0.27812
MRFCF (Steck, 2019)	0.17710	0.19300	0.19870	0.19123	0.28943	0.29260	0.08738	0.19488	0.29048
SGMC (Chen et al., 2021)	0.23290	0.31655	0.34500	0.13560	0.31070	0.40790	0.09740	0.22735	0.32193
GS-IMC (ours, Sec 3)	0.23460	0.31995	0.35065	0.13677	0.31027	0.40760	0.09725	0.22733	0.32225
BGS-IMC (ours, Sec 4)	**0.24390**	**0.32545**	**0.35345**	0.16733	**0.34313**	**0.43690**	**0.09988**	**0.23390**	**0.33063**

1. BGSIMC consistently outperforms GSIMC;
2. BGSIMC achieves the state-of-the-art performances;

ICLR