Convolution-Consistent Collective Matrix Completion

Xu Liu
Arizona State University
xliu338@asu.edu

Jingrui He
University of Illinois at Urbana-Champaign
jingrui@illinois.edu

Sam Duddy
Allstate Northern Ireland
sam.duddy@allstate.com

Liz O'Sullivan
Allstate Northern Ireland
losul@allstate.com

ABSTRACT

Collective matrix completion refers to the problem of simultaneously predicting the missing entries in multiple matrices by leveraging the cross-matrix information. It finds abundant applications in various domains such as recommender system, dimensionality reduction, and image recovery. Most of the existing work represents the cross-matrix information in a shared latent structure constrained by the Euclidean-based pairwise similarity, which may fail to capture the nonlinear relationship of the data. To address this problem, in this paper, we propose a new collective matrix completion framework, named C^4, which uses the graph spectral filters to capture the non-Euclidean cross-matrix information. To the best of our knowledge, this is the first effort to represent the cross-matrix information in the graph spectral domain. We benchmark our model against 8 recent models on 10 real-world data sets, and our model outperforms state-of-the-art methods in most tasks.

CCS CONCEPTS

• **Information systems → Social recommendation**.

ACM Reference Format:

Xu Liu, Jingrui He, Sam Duddy, and Liz O'Sullivan. 2019. Convolution-Consistent Collective Matrix Completion. In *The 28th ACM International Conference on Information and Knowledge Management (CIKM '19), November 3–7, 2019, Beijing, China.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3357384.3358111

1 INTRODUCTION

Collective Matrix Completion (CMC) is a fundamental data mining problem, where the goal is to collectively complete multiple incomplete matrices by leveraging the cross-matrix information. Each matrix, also known as one view, corresponds to one type of measurement, while multiple views contain complementary information from various sources. Many high impact applications, such as recommender system [8], neuroimages analyzing [16], crowdsourcing [19, 20], and rare category detection [17, 18], bring us huge amount of incomplete data from multiple sources. CMC benefits from the correlation among such multi-view data, and aims to predict their missing entries with a high accuracy.

There are two approaches extensively applied in the CMC studies when formulating the cross-matrix information: *(i) Low-rank Latent Structure*: the cross-matrix information is encoded into a single measurement matrix, which has the overall minimum Euclidean distance to each view [7]. *(ii) Graph Knowledge*: the samples' pairwise similarity is enforced by the graph-based regularizer with respect to the observed information [2]. More recently, through revisiting deep learning with graph spectral theory, graph knowledge is decoded by the deep model as a graph convolution network by introducing graph filters. In particular, the authors of [4] showed that the Chebyshev polynomial approximation can well estimate the graph filters, and the authors of [3] introduced such graph filters into convolution neural networks for handling the graph structured data. Furthermore, the authors of [5] simplified the graph convolution process and widely applied since its inception [15].

However, state-of-the-art techniques for CMC currently face the following challenges. (**C1**) The low-rank latent structure cannot fully exploit the cross-matrix information, especially when the data exhibit non-Euclidean structure. (**C2**) The graph convolution method is dependent on Chebyshev polynomial expansions. As we will demonstrate in this paper, different K values (i.e., how many polynomial bases are taken, also known as K-order graph filter) can significantly impact the effectiveness. Therefore, it is preferred to have K adjusted according to the specific applications. However, almost all the existing methods use a fixed K value, e.g., [5] sets the order K=2, [3] adopts K=5, and [16] conducts the experiments with K=30. (**C3**) The graph convolution method is originally designed with a single matrix input, which cannot be readily applied to multiple matrices. To address these challenges, in this paper, we propose a Convolution-Consistent Collective Matrix Completion framework, named C^4. Our contributions are summarized as follows.

- **Formulation**: The CMC problem is formulated as a joint graph convolution problem. The essential non-Euclidean cross-matrix information is encoded in the graph spectral domain with the adapted data-driven filter order.
- C^4 **Framework**: We propose a novel framework for CMC with adaptive graph filter order K^*, such that the information loss is minimized for all the matrices.
- **Experiments**: Our model C^4 performs the best when compared with 8 state-of-the-art methods on 10 real-world data sets.

The rest of the paper is organized as follows. In Sect. 2, we introduce the preliminary knowledge and present our model C^4. Sect. 3 shows the experimental results and we conclude the paper in Sect. 4.

2 PROPOSED MODEL

In this section, we first review the preliminary knowledge about spectral graph filter and graph convolution (Sect. 2.1). Then we present the proposed model C^4, in which we aim to solve two subproblems that: (i) how to select the graph filter order K^* (Sect. 2.2) and (ii) how to further capture the cross-matrix information through the K^*-th order joint graph convolution process (Sect. 2.3). Figure 1 shows the overall structure of our model. **Notation**: In general,

CIKM '19, November 3–7, 2019, Beijing, China

ACM ISBN 978-1-4503-6976-3/19/11...$15.00
https://doi.org/10.1145/3357384.3358111

we use bold-face uppercase letter $\mathbf{X}$ to represent a matrix. $\mathbf{X}_{[i,j]}$ denotes its entry in the i^{th} row and j^{th} column, and $X^\top$ denotes its transpose. We use bold-face lowercase letter $\mathbf{u}$ to denote a column vector, whose i^{th} entry is denoted $\mathbf{u}_i$. The uppercase Greek letters are used to represent scalars. **Goal**: Given multiple incomplete matrices $\{\mathbf{X}_t\}_{t=1}^T \subset R^{m\times n_t}$, our goal is to predict their missing entries simultaneously. To this end, the proposed C^4 framework jointly factorizes all these T matrices into $\{\mathbf{U}_t\}_{t=1}^T \subset R^{m\times c_t}$ and $\{\mathbf{V}_t\}_{t=1}^T \subset R^{n_t\times c_t}$ by incorporating the cross-matrix information from graph spectral domain. The products $\{\tilde{\mathbf{X}}_t = \mathbf{U}_t\mathbf{V}_t^\top\}_{t=1}^T$ contain the estimated missing entries.

2.1 Spectral Graph Filter and Convolution

A graph $\mathcal{G}$ with m nodes is presented as $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{A})$, with the adjacency matrix $\mathbf{A} \in R^{m\times m}$, vertex set $\mathcal{V}$ and edge set $\mathcal{E}$. The normalized graph Laplaican matrix $\mathbf{\Delta} = \mathbf{I} - \mathbf{D}^{-\frac{1}{2}}\mathbf{A}\mathbf{D}^{-\frac{1}{2}}$ with the diagonal degree matrix $\mathbf{D}_{ii} = \sum_j^m \mathbf{A}_{ij}$ and identity matrix $\mathbf{I} \in R^{m\times m}$. As $\mathbf{\Delta}$ is positive semidefinite, it has a complete set of eigenvalues $\mathbf{\Lambda} = (\lambda_1, \lambda_2, \ldots, \lambda_m)$ and eigenvectors $\mathbf{\Phi} = (\phi_1, \phi_2, \ldots, \phi_m)$ for its eigendecomposition $\mathbf{\Delta} = \mathbf{\Phi\Lambda\Phi}^\top$. Eigenvalues $\{\lambda_i\}_{i=1}^m$ are identified as graph spectral frequencies and eigenvectors $\{\phi_i\}_{i=1}^m$ are identified as graph Fourier basis. For a graph signal $\mathbf{x} \in R^m$, the *graph Fourier transform* is defined as $\tilde{\mathbf{x}} = \mathbf{\Phi}^\top\mathbf{x}$ and its inverse transform $\mathbf{x} = \mathbf{\Phi}\tilde{\mathbf{x}}$. The graph convolutional operation $*_\mathcal{G}$ for the graph signals $\mathbf{x}$ and $\mathbf{y}$ is then defined on the graph spectral domain as $\mathbf{x} *_\mathcal{G} \mathbf{y} = \mathbf{\Phi}(\mathbf{\Phi}^\top\mathbf{x}) \odot (\mathbf{\Phi}^\top\mathbf{y}) = \mathbf{\Phi}g_\theta(\mathbf{\Lambda})\tilde{\mathbf{x}}$, where $\odot$ denotes element wise product. $g_\theta(\mathbf{\Lambda})$ is recognized as θ-parameterized graph filter. More recently, the graph filter $g_\theta(\mathbf{\Lambda})$ is re-modeled to decrease its complexity by being expanded by the Chebyshev polynomial as:

$$g_\theta(\mathbf{\Lambda}) = \sum_{k=0}^K \theta_k T_k(\tilde{\mathbf{\Delta}}) = \sum_{k=0}^K \theta_k \mathbf{\Phi} T_k(\tilde{\mathbf{\Lambda}})\mathbf{\Phi}^\top \tag{1}$$

where the modified graph Laplacian $\tilde{\mathbf{\Delta}} = \frac{2\Delta}{\lambda_{max}} - \mathbf{I}_m$ and its eigenvalues $\tilde{\mathbf{\Lambda}}$ fall into the range $[-1, 1]$. $T_k(\cdot)$ represents the k-th order Chebyshev polynomial abiding by the recursive manner $T_k(\lambda) = 2\lambda T_{k-1}(\lambda) - T_{k-2}(\lambda)$ with $T_0(\lambda) = 1$ and $T_1(\lambda) = \lambda$.

2.2 Adapted Data-Driven Filter Order

In this subsection, we introduce our proposed techniques for selecting the filter order $K*$. The problem definition is as follows.

Problem 1: *Selecting Filter Order K^**

Input: *Incomplete matrices $\{\mathbf{X}_t\}_{t=1}^T$ with missing entries.*

Output: *Adapted graph filter order K^*.*

The output K^* denotes the adapted graph filter order derived from the cross-matrix information observed in the input matrices $\{\mathbf{X}_t\}_{t=1}^T$. Emphasized in [5], the order K^* plays a decisive role in adopting the graph structure knowledge, i.e., only the nodes within maximum K^* steps away from the central node are taken into consideration. The data-adaptive order K^* can improve the completion performance significantly, while none analytic guidance address it yet.

In our model, we propose to settle the order K^* with respect to the minimum information loss considering from the graph spectral domain. Being expanded by K^*-th order Chebyshev polynomial, each matrix is reconstructed as the weighted combination of its own graph structure knowledge from K^* level. The expectation is that K^*-th order graph filters are capable enough to preserve the observed knowledge as much as possible. The completion results $\{\tilde{\mathbf{X}}_t\}_{t=1}^T$ are expected to be consistent with the observed entries in $\{\mathbf{X}_t\}_{t=1}^T$ as the reconstruction error between $\{\tilde{\mathbf{X}}_t\}_{t=1}^T$ and $\{\mathbf{X}_t\}_{t=1}^T$ is minimum, which is defined as solving the problem:

$$\min_{\mathbf{U}_t, \mathbf{V}_t} \sum_{t=1}^T \left\|\mathbf{X}_t - \mathbf{U}_t\mathbf{V}_t^\top\right\|_{F,\Omega_t}^2 \qquad s.t.\ \{\mathbf{U}_t\}_{t=1}^T \subset R_+^{m\times c_t},\ \{\mathbf{V}_t\}_{t=1}^T \subset R_+^{n_t\times c_t} \tag{2}$$

where $\mathbf{U}_t \in R^{m\times c_t}$ and $\mathbf{V}_t \in R^{n_t\times c_t}$. R_+ denotes the non-negative real numbers and the completion results are $\{\tilde{\mathbf{X}}_t = \mathbf{U}_t\mathbf{V}_t^\top\}_{t=1}^T$. The index matrix $\mathbf{\Omega}_t \in R^{m\times n_t}$ contains $\mathbf{\Omega}_{t[i,j]} = 1$ if $\mathbf{X}_{t[i,j]}$ is observed, otherwise 0. To simplify the expression, $\|\mathbf{X}\|_{F,\Omega}^2$ is equivalent to the expression of $\|\mathbf{X} \odot \mathbf{\Omega}\|_F^2$, in which $\odot$ denotes the Hadamard product. In Eq. (2), the $\mathbf{U}_t$ is polynomial expanded by Eq. (1) as:

$$\begin{aligned}&\min_{\mathbf{U}_t,\mathbf{V}_t,\theta_{k,t}} \sum_{t=1}^T ||\mathbf{X}_t - (\sum_{k=0}^K \theta_{k,t}T_k(\tilde{\mathbf{\Delta}}_{r,t})\mathbf{U}_t)\mathbf{V}_t^\top||_{F,\Omega_t}^2 \\ \Leftrightarrow &\min_{\mathbf{U}_t,\mathbf{V}_t,\theta_{k,t}} \sum_{t=1}^T ||\mathbf{X}_t - (\theta_{0,t}T_0(\tilde{\mathbf{\Delta}}_{r,t})\mathbf{U}_t + \theta_{1,t}T_1(\tilde{\mathbf{\Delta}}_{r,t})\mathbf{U}_t + \\ &\qquad \cdots + \theta_{K,t}T_K(\tilde{\mathbf{\Delta}}_{r,t})\mathbf{U}_t)\mathbf{V}_t^\top||_{F,\Omega_t}^2\end{aligned} \tag{3}$$

constrained by $\{\mathbf{U}_t\}_{t=1}^T \subset R_+^{m\times c_t}$ and $\{\mathbf{V}_t\}_{t=1}^T \subset R_+^{n_t\times c_t}$. For the t-th view, parameter $\theta_{k,t}$ weights the k-th order graph filter $T_k(\tilde{\mathbf{\Delta}}_{r,t})$. $\tilde{\mathbf{\Delta}}_{r,t}$ denotes the normalized row-wise Laplacian matrix. To be more specific, the factor $\mathbf{U}_t$ is described as the weighted combination of K-th order graph structure knowledge, which is purposeful to reinforced the estimation of the matrix $\mathbf{U}_t$ by the localized graph knowledge from K-th level neighboring information.

Here, what makes Eq. (3) more attractive is that the reconstruction error is altered according to the graph filter order K. The low-order graph filters capture the nearest neighborhood knowledge surrounding each node, which shows the similar patterns existing in each view. While as the order increases, the less similarity has been preserved by the far-away neighborhoods. Even worse, we found that the model would be impaired when incorporating the far-away neighborhoods into cross-matrix information. Thus, we settle the order K^* for each data set which brings in the minimum effect when removing the graph filters higher than K^*. The superiority of this strategy is shown in the experiments.

Algorithm 1 - C^4 Updating Procedure

1: **Input**: (1) multiple matrices $\{\mathbf{X}_t\}_{t=1}^T$ (with missing entries). (2) normalized Laplacian matrices $\{\tilde{\mathbf{\Delta}}_{r,t}\}_{t=1}^T$. (3) filter order K^*.
2: **Initialize**: $\{\mathbf{U}_t\}_{t=1}^T$, $\{\mathbf{V}_t\}_{t=1}^T$, $\mathbf{\Theta}_k$ and $\mathcal{W}_k$ randomly.
3: **Repeat**: Stochastic Gradient Descent (SGD) algorithm to update $\{\mathbf{U}_t\}_{t=1}^T$, $\{\mathbf{V}_t\}_{t=1}^T$, $\mathbf{\Theta}_k$ and $\mathcal{W}_k$ one at a time.
4: **Until**: Eq. (5) converges.
5: **Output**: $\{\tilde{\mathbf{X}}_t\}_{t=1}^T$: completion results

2.3 Quantifying Cross-Matrix Information

In this subsection, we present our proposed techniques for quantifying cross-matrix information. The problem definition is as follows.

Problem 2: *Quantifying Cross-Matrix Information*

Input: *(1) Incomplete matrices $\{\mathbf{X}_t\}_{t=1}^T$. (2) Graph filter order K^* obtained from Problem 1.*

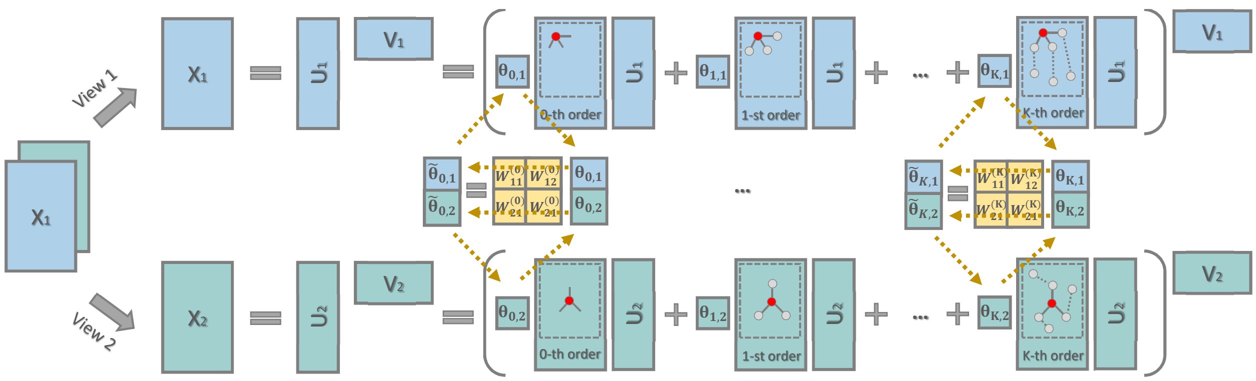

Figure 1: The overall architecture of our framework with two views.

Output: *(1) Matrix-stitch unit* $\mathcal{W}_k$. *(2) Completion results* $\{\tilde{\mathrm{X}}_t\}_{t=1}^{T}$.
There are two main challenges arise in Problem 2: (**P2-a**) how to capture the cross-matrix information when data implicate the non-Euclidean structure, e.g., graph-structured data. (**P2-b**) how to quantify the matrices' interactive impacts. Either positive or negative impacts exist between the matrices, e.g., how much knowledge does the view 2 contribute to predicting the missing entries in view 1? Is the view 2 more competent than view 3 for predicting the missing entries in view 1? Existing methods fail to address these issues, while we make an effort to correct the cross-matrix information.

We propose the **Matrix-Stitch Unit** $\mathcal{W}_k$ to answer Problem 2. For illustration purpose, only two views are considered (T=2), while in practice, the unit $\mathcal{W}_k$ is feasible to the arbitrary number of views ($T \geq 2$), which has been demonstrated in our experiments. Based on the Eq. (3) when ($t = 1, 2,\ k = 1, 2, \ldots, K$), the factors $\mathbf{U}_t$ are expanded by $T_k(\tilde{\Delta}_{r,t})$ and $\theta_{k,t}$, where $\theta_{k,t}$ is the learnable weighted parameter reflecting how does the k-th localized graph knowledge $T_k(\tilde{\Delta}_{r,t})$ impact in each view separately. The Matrix-Stitch Unit considers the impacts from both view itself and all the other views. The unit $\mathcal{W}_k$ is designed as a weight matrix between the parameters $\{\theta_{k,t}\}_{t=1}^{T=2}$ for each level of the graph localized knowledge:

$$\tilde{\Theta}_k = \begin{bmatrix} \tilde{\theta}_{k,1} \\ \tilde{\theta}_{k,2} \end{bmatrix} = \begin{bmatrix} w_{11} & w_{12} \\ w_{21} & w_{22} \end{bmatrix}_k \begin{bmatrix} \theta_{k,1} \\ \theta_{k,2} \end{bmatrix} = \mathcal{W}_k \Theta_k \tag{4}$$

where $\tilde{\Theta}_k, \Theta_k \in R^{T\times 1}$. The matrix-stitch unit $\mathcal{W}_k \in R^{T\times T}$. There are total K^* units when adopting K^* order graph filter. Incorporating Eq. (4) with Eq. (3), the C^4 objective function is written as:

$$\begin{aligned} &\min_{\mathrm{U}_t, \mathrm{V}_t, \Theta_k, \mathcal{W}_k} \sum_{t=1}^{T} ||\mathbf{X}_t - (\sum_{k=0}^{K} \tilde{\Theta}_{k_{[t,1]}} T_k(\tilde{\Delta}_{r,t}) \mathrm{U}_t) \mathrm{V}_t^{\top}||_{F,\Omega_t}^2 \\ \Leftrightarrow &\min_{\mathrm{U}_t, \mathrm{V}_t, \Theta_k, \mathcal{W}_k} \sum_{t=1}^{T} ||\mathbf{X}_t - (\sum_{k=0}^{K} \mathcal{W}_{k_{[t,:]}} \Theta_k T_k(\tilde{\Delta}_{r,t}) \mathrm{U}_t) \mathrm{V}_t^{\top}||_{F,\Omega_t}^2 \end{aligned} \tag{5}$$

constrained by $\{\mathrm{U}_t\}_{t=1}^{T} \subset R_+^{m\times c_t}$ and $\{\mathrm{V}_t\}_{t=1}^{T} \subset R_+^{n_t\times c_t}$. The C^4 updating procedure is summarized in Algorithm 1 omitted to space.

3 EXPERIMENTAL RESULTS

In this section, we first present our data sets and experiment setting. Then, we present matrix completion results compared with others and further discussion about the effectiveness of filter order K^*.

3.1 Experiment Settings

Data Sets. Table. 1 shows ten data sets collected from Amazon datum [8]. Seven of them contain two views (ID 1-7) and three of them contain three views (ID 8-10). Taking data set (ID 1) as an example, view 1 '*Electronics*' contains 6352 users and their 39574 ratings for 12836 products, and view 2 '*Video Games*' contains 27712 rating for 12,836 products from the same users group. In each view, 30% ratings of each item are removed and serves as the ground-truth for the completion results.
Baselines. Our model is compared with 8 state-of-the-art methods, including GROUSE [1], IALM [6], LMaFit [12], MC-NMF [13], OR1MP [11], RMAMR [14], ScGrassMC [9], and multiNMF [7]. Parameters are initialized as suggested in [10].

3.2 Results and Discussions

As shown in Table 2, the completion results are evaluated by the mean squared error (MSE) between the ground-truth and prediction values. Our model C^4 achieves the best completion performance compared with state-of-the-art methods.

For various data sets, the ideal filter order can be obtained by iterating over every possible value, however, it is infeasible to handle large data set. Hereby, we estimate the order K^* based on sample proportion of the observed data $\{X_t\}_{t=1}^{T}$. Within the realm of affordable computation cost, the larger portion sampled, the more precise order K^* can be estimated. Red stars in Fig. 2 (graph filter order v.s. reconstruction MSE) and Fig. 3 (graph filter order v.s. prediction MSE) denote K^* estimations. The blue stars in Fig. 3 denote the ideal filter order identified through the offline iterative searching. In most cases, the estimation filter orders (K^* in red stars) are close

ID	Views	|User|	|Item1|	|Item2|	|Item3|	|Rating|
1	Electronics & Video Games	6352	12836	8059	-	39574
2	Patio & Tools	3778	4077	7813	-	27712
3	Beauty Product & Clothing	1318	3406	7261	-	12691
4	Art & Musical Instruments	1412	602	988	-	8520
5	Electronics & Kindle Store	1050	3956	1614	-	7431
6	Beauty Product & Jewelry	266	1458	730	-	3870
7	Kindle Store & Software	190	637	627	-	2885
8	Electronics & Video Games & Software	1724	4383	2487	3845	12741
9	Patio & Tools & Pet Supplies	652	812	1564	715	8125
10	Beauty Product & Clothing & Jewelry	571	1845	2377	492	4298

Table 1: Multi-View Amazon Review Data Sets.

Table 2: Matrix Completion MSE w.r.t. ground-truth and missing entries.

Dataset ID	C^4	GROUSE	IALM	LMaFit	MC-NMF	OR1MP	RMAMR	ScGrassMC	multiNMF
1	1.206±0.031	**1.181±0.162**	1.689±0.004	1.429±0.002	1.986±2.542E-6	1.590±4.590E-31	1.344±0.003	1.347±1.275E-30	3.817±1E-9
2	**1.350±0.007**	1.368±0.133	1.423±0.012	1.452±0.004	1.946±1.104E-5	1.464±1.653E-30	1.416±0.007	1.943±1.275E-31	2.298±1E-9
3	**1.027±0.004**	1.282±0.087	1.446±0.005	2.005±0.002	2.045±1.447E-5	1.392±2.040E-31	1.536±0.003	1.733±1.275E-30	4.301±1E-9
4	**1.016±0.002**	1.320±0.087	1.478±0.008	1.664±0.004	2.059±3.419E-6	1.796±4.590E-31	1.506±0.005	1.814±2.648E-31	2.961±1E-9
5	1.341±0.092	**1.268±0.093**	1.807±0.003	1.894±0.002	2.008±4.946E-6	1.475±1.275E-30	1.387±0.006	1.641±8.161E-31	2.888±1E-9
6	**1.235±0.031**	1.328±0.134	1.494±0.007	2.082±0.008	2.057±8.806E-6	1.808±2.684E-31	1.498±0.001	2.047±3.264E-30	4.253±1E-9
7	**1.174±0.056**	1.217±0.015	1.759±0.003	2.028±0.008	1.984±3.216E-5	1.444±2.040E-31	1.517±0.006	1.903±4.590E-31	2.403±1E-9
8	**1.256±0.081**	1.335±0.153	1.812±0.015	1.896±0.002	2.009±8.37E-6	1.590±1.154E-9	1.437± 0.005	1.676± 5.478E-32	3.674±1E-9
9	**1.207±0.048**	1.310±0.089	1.504±0.020	2.080±0.001	2.058±1.27E-5	1.264±1.348E-9	1.9431±0.004	2.044±2.191E-31	2.479±1E-9
10	**1.243±0.032**	1.279±0.032	1.732±0.003	1.880±0.004	1.981±5.14E-5	1.292±1 .674E-9	1.742±0.002	1.841±4.213E-32	2.738±1E-9

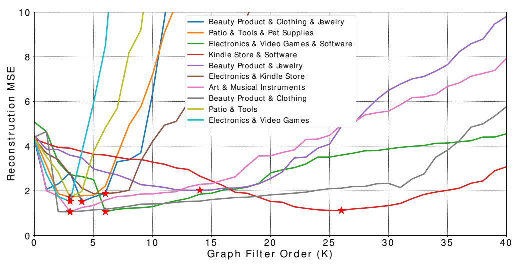

Figure 2: Red stars (K^*) leads to minimum reconstruction.

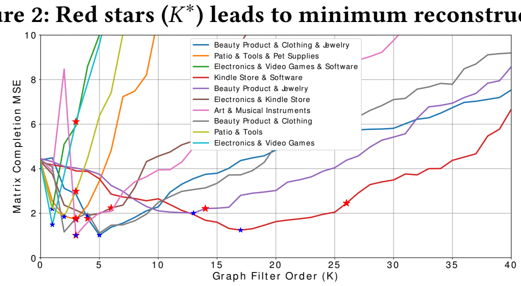

Figure 3: Red stars are the same stars in Fig. 2. Blue stars denote ideal filter order identified through offline searching.

to the ideal order (blue stars). The exception happens in the data set with ID 7 as the data volume is relatively small. In addition to the order estimation, we verify the assumption in Problem 1 that larger K does not signify the authentic cross-matrix information. As shown in Fig. 3, the completion performance declines as the MSE value bounces back after a certain point of the increasing filter order, as the cross-matrix information being impaired when incorporating the nodes far away from the neighborhood.

4 CONCLUSION

In this paper, we have proposed a novel multi-view graph convolution framework (C^4) for collective matrix completion. We make the first effort to decode the essential cross-matrix information in the graph spectral domain and quantify the matrices' interactive impacts. Experimental results on ten real-world data sets demonstrate the effectiveness of C^4 as compared to state-of-the-art methods.

ACKNOWLEDGEMENT

This work is supported by National Science Foundation under Grant No. IIS-1552654 and Grant No. IIS-1813464, the U.S. Department of Homeland Security under Grant Award Number 17STQAC00001-02-00, and an IBM Faculty Award. The views and conclusions are those of the authors and should not be interpreted as representing the official policies of the funding agencies or the government.

REFERENCES

[1] Laura Balzano and Stephen J Wright. 2013. On GROUSE and incremental SVD. In *Computational Advances in Multi-Sensor Adaptive Processing, 2013 IEEE 5th International Workshop on.* IEEE, 1–4.

[2] Deng Cai, Xiaofei He, Jiawei Han, and Thomas S Huang. 2010. Graph regularized nonnegative matrix factorization for data representation. *IEEE transactions on pattern analysis and machine intelligence* 33, 8 (2010), 1548–1560.

[3] Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. 2016. Convolutional neural networks on graphs with fast localized spectral filtering. In *Advances in Neural Information Processing Systems.* 3844–3852.

[4] David K Hammond, Pierre Vandergheynst, and Rémi Gribonval. 2011. Wavelets on graphs via spectral graph theory. *Applied and Computational Harmonic Analysis* 30, 2 (2011), 129–150.

[5] Thomas N Kipf and Max Welling. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907* (2016).

[6] Zhouchen Lin, Minming Chen, and Yi Ma. 2010. The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices. *arXiv preprint arXiv:1009.5055* (2010).

[7] Jialu Liu, Chi Wang, Jing Gao, and Jiawei Han. 2013. Multi-view clustering via joint nonnegative matrix factorization. In *Proceedings of the 2013 International Conference on Data Mining.* SIAM, 252–260.

[8] Julian McAuley and Jure Leskovec. 2013. Hidden factors and hidden topics: understanding rating dimensions with review text. In *Proceedings of the 7th ACM conference on Recommender systems.* ACM, 165–172.

[9] Thanh Ngo and Yousef Saad. 2012. Scaled gradients on Grassmann manifolds for matrix completion. In *Advances in Neural Information Processing Systems.*

[10] A. Sobral and E. Zahzah. 2016. Matrix and Tensor Completion Algorithms for Background Model Initialization: A Comparative Evaluation. *Pattern Recognition Letters* (2016). https://doi.org/10.1016/j.patrec.2016.12.019

[11] Zheng Wang, Ming-Jun Lai, Zhaosong Lu, Wei Fan, Hasan Davulcu, and Jieping Ye. 2015. Orthogonal rank-one matrix pursuit for low rank matrix completion. *SIAM Journal on Scientific Computing* 37, 1 (2015), A488–A514.

[12] Zaiwen Wen, Wotao Yin, and Yin Zhang. 2012. Solving a low-rank factorization model for matrix completion by a nonlinear successive over-relaxation algorithm. *Mathematical Programming Computation* 4, 4 (2012), 333–361.

[13] Yangyang Xu, Wotao Yin, Zaiwen Wen, and Yin Zhang. 2012. An alternating direction algorithm for matrix completion with nonnegative factors. *Frontiers of Mathematics in China* 7, 2 (2012), 365–384.

[14] Xinchen Ye, Jingyu Yang, Xin Sun, Kun Li, Chunping Hou, and Yao Wang. 2015. Foreground–background separation from video clips via motion-assisted matrix restoration. *IEEE Transactions on Circuits and Systems for Video Technology* 25, 11 (2015), 1721–1734.

[15] Si Zhang, Hanghang Tong, Jiejun Xu, and Ross Maciejewski. 2018. Graph convolutional networks: Algorithms, applications and open challenges. In *International Conference on Computational Social Networks.* Springer, 79–91.

[16] Xi Zhang, Lifang He, Kun Chen, Yuan Luo, Jiayu Zhou, and Fei Wang. 2018. Multi-View Graph Convolutional Network and Its Applications on Neuroimage Analysis for Parkinson's Disease. *arXiv preprint arXiv:1805.08801* (2018).

[17] Dawei Zhou, Jingrui He, K Seluk Candan, and Hasan Davulcu. 2015. MUVIR: multi-view rare category detection. In *24th IJCAI.*

[18] Dawei Zhou, Jingrui He, Hongxia Yang, and Wei Fan. 2018. Sparc: Self-paced network representation for few-shot rare category characterization. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining.* ACM, 2807–2816.

[19] Yao Zhou and Jingrui He. 2016. Crowdsourcing via Tensor Augmentation and Completion.. In *IJCAI.* 2435–2441.

[20] Yao Zhou and Jingrui He. 2017. A randomized approach for crowdsourcing in the presence of multiple views. In *2017 IEEE International Conference on Data Mining (ICDM).* IEEE, 685–694.