Geometric Matrix Completion: A Functional View

Abhishek Sharma, Maks Ovsjanikov

Geometric Matrix Completion: A Functional View

Abhishek Sharma
LIX, École Polytechnique
kein.iitian@gmail.com

Maks Ovsjanikov
LIX, École Polytechnique
maks@lix.polytechnique.fr

Abstract

We propose a totally functional view of geometric matrix completion problem. Differently from existing work, we propose a novel regularization inspired from the functional map literature that is more interpretable and theoretically sound. On synthetic tasks with strong underlying geometric structure, our framework outperforms state of the art by a huge margin (two order of magnitude) demonstrating the potential of our approach. On real datasets, we achieve state-of-the-art results at a fraction of the computational effort of previous methods. Our code will be publicly available at https://github.com/Not-IITian/functional_matrix_completion

1 Introduction

Matrix completion deals with the recovery of missing values of a matrix of which we have only measured a subset of the entries,

$$\text{Find } \boldsymbol{X} \text{ s.t. } \boldsymbol{X} \odot \boldsymbol{S} = \boldsymbol{M} \odot \boldsymbol{S}. \tag{1}$$

Here $\boldsymbol{X}$ stands for the unknown matrix, $\boldsymbol{M} \in \mathbb{R}^{m \times n}$ for the ground truth matrix, $\boldsymbol{S}$ is a binary mask representing the input support, and $\odot$ denotes the Hadamard product. In general, without any constraints, this problem (1) is ill-posed and not solvable. However if the rank of underlying matrix is small, the number of degrees of freedom decreases and thus, it is common to find the lowest rank matrix that agrees with known measurements. Under this low rank assumption, the matrix completion problem can be rewritten as,

$$\min_{\boldsymbol{X}} \text{ rank}\,(\boldsymbol{X}) + \frac{\mu}{2} \left\|(\boldsymbol{X} - \boldsymbol{M}) \odot \boldsymbol{S}\right\|_F^2. \tag{2}$$

Various problems in collaborative filtering can be posed as a *matrix completion* problem Kalofolias et al. [2014], Rao et al. [2015], where for example the columns and rows represent users and items, respectively, and matrix values represent a score determining whether a user would like an item or not. This setting was particularly popularized by the Netflix challenge Koren et al. [2009]. Often, additional structural information is available in the form of column and row graphs representing similarity of users and items, respectively. Such geometric information is not exploited by rank based prior work that only seeks a purely algebraic solution by optimizing for low rank Candès and Recht [2009]. Prior work that incorporates geometric structure into matrix completion problemsMonti et al. [2017] obtains state of the art results using powerful pattern extraction ability of graph CNN but falls short of giving a principled framework to model such geometric information.

Boyarski et al. [2020] makes an attempt to build a principled framework that is based on a functional map representation Ovsjanikov et al. [2012] and also compete empirically with highly engineered models such as multi-graph CNNMonti et al. [2017]. One of the advantages of working with the functional map representation is that its size is typically much smaller, and is only controlled by the size of the basis, independent of the number of nodes in graphs, resulting in simpler optimization problems Although Boyarski et al. [2020] obtains state-of-the-art results on both synthetic and

Preprint. Under review.

real datasets, it introduces several non-convex regularization terms thereby, making the overall optimization harder to optimize. To address this challenge, we propose a simple formulation based on functional map, consisting of a single regularizer, that mitigates the problems associated with Boyarski et al. [2020].

Contributions. Our contributions are threefold. First, we propose a novel functional view of geometric matrix completion problem that is convex in formulation and theoretically grounded. Second, our method, on synthetic tasks with strong underlying geometric model, sets new benchmarks in modelling the geometric information that are 100 times superior over prior work. Third, our proposed model obtains state-of-the-art results on various real world recommendation systems datasets while being more intuitive and simpler to optimize and thereby, easier to analyze and reproduce.

2 Related work

Matrix completion has been studied with many viewpoints and thus, exhaustive coverage of prior work is beyond the scope of this paper. In this section, we mainly describe related work on geometric matrix completion.

Geometric matrix completion. A prominent relaxation of the rank operator in Eq. (2) is to constrain the space of solutions to be smooth w.r.t. some geometric structure of the matrix rows and columns. There exist several prior work on geometric matrix completion problem that exploit such geometric information Berg et al. [2017], Kalofolias et al. [2014], Rao et al. [2015] such as graphs encoding relation between rows and columns. More recent work leverages deep learning on geometric domain Berg et al. [2017], Monti et al. [2017] to extract relevant information from geometric data such as graphs. As argued in Boyarski et al. [2020], while these techniques achieve state-of-the-art results, their design is highly engineered and thus, non-intuitive.

Functional Maps. Our work is mainly inspired from the functional map framework [Ovsjanikov et al., 2012] used ubiquitously in non-rigid shape correspondence, and has been extended to handle challenging partial matching cases, e.g. Litany et al. [2017]. This framework has recently been adapted for geometric matrix completion in Boyarski et al. [2020], where the authors propose to build a functional map between graphs of rows and columns. As noted in several works, isometry between two spaces is a key to functional map representation. Assuming isometry between real world graphs is however over optimistic. Thus, one way to work under relaxed isometry condition is to instead align the eigen basis with additional transformation matrix to achieve diagonal functional map matrixLitany et al. [2017]. Boyarski et al. [2020] achieve this with a range of transformation on eigen basis of graph Laplacian. However, they 1) impose several regularization terms each with a scaling hyperparameter and some even with different initialization 2) explore a huge range of hyperparameter space and question remains on why does it work so well.

3 Preliminaries

In this section, we cover some preliminaries about product graphs and functional maps.

Product graphs Let $\mathcal{G} = (V, E, W)$ be a (weighted) graph with its vertex set V and edge set E and adjacency matrix denoted by W. Graph Laplacian $\boldsymbol{L}$ is given by $\boldsymbol{L} = \boldsymbol{D} - \boldsymbol{W}$, where $\boldsymbol{D} = \mathrm{diag}(\boldsymbol{W}\mathbf{1})$ is the *degree matrix*. $\boldsymbol{L}$ is symmetric and positive semi-definite and therefore admits a spectral decomposition $\boldsymbol{L} = \boldsymbol{\Phi}\boldsymbol{\Lambda}\boldsymbol{\Phi}^\top$. It is well-known that spectrum of the Laplacian contains the structural information about the graph Spielman [2009]. Let $\mathcal{G}_1 = (V_1, E_1, W_1)$, $\mathcal{G}_2 = (V_2, E_2, W_2)$ be two graphs, with $\boldsymbol{L}_1 = \boldsymbol{\Phi}\boldsymbol{\Lambda}_1\boldsymbol{\Phi}^\top$, $\boldsymbol{L}_2 = \boldsymbol{\Psi}\boldsymbol{\Lambda}_2\boldsymbol{\Psi}^\top$ being their corresponding graph Laplacians. The bases $\boldsymbol{\Phi}, \boldsymbol{\Psi}$ can be used to represent functions on these graphs. We define the Cartesian product of $\mathcal{G}_1$ and $\mathcal{G}_2$, denoted by $\mathcal{G}_1 \square \mathcal{G}_2$, as the graph with vertex set $V_1 \times V_2$, on which two nodes $(u, u'), (v, v')$ are adjacent if either $u = v$ and $(u', v') \in E_2$ or $u' = v'$ and $(u, v) \in E_1$.

Functional maps. Let $\boldsymbol{X}$ be a function defined on $\mathcal{G}_1 \square \mathcal{G}_2$. It can be encoded as a matrix of size or $|V_1| \times |V_2|$. Then it can be represented using the bases $\boldsymbol{\Phi}, \boldsymbol{\Psi}$ of the individual graph Laplacians,

$\boldsymbol{C} = \boldsymbol{\Phi}^\top \boldsymbol{X} \boldsymbol{\Psi}$. In the shape processing community, such $\boldsymbol{C}$ is called a *functional map*, as it it used to map between the functional spaces of $\mathcal{G}_1$ and $\mathcal{G}_2$. One of the advantages of working with the functional map representation $\boldsymbol{C}$ rather than the matrix $\boldsymbol{X}$ is that its size is typically much smaller, and is only controlled by the size of the basis, independent of the number of nodes in G_1 and G_2, resulting in simpler optimization problems. Moreover, the projection onto a basis also provides a strong regularization, which can itself be beneficial for both shape matching, and, as we show below, matrix completion. For example, given two functions, $\boldsymbol{x} = \boldsymbol{\Phi}\boldsymbol{\alpha}$ on $\mathcal{G}_1$ and $\boldsymbol{y} = \boldsymbol{\Psi}\boldsymbol{\beta}$ on $\mathcal{G}_2$, one can use $\boldsymbol{C}$ to map between their representations $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$, i.e., $\boldsymbol{\alpha} = \boldsymbol{\Phi}^\top \boldsymbol{x} = \boldsymbol{C}\boldsymbol{\Psi}^\top \boldsymbol{y} = \boldsymbol{C}\boldsymbol{\beta}$.

4 Functional Geometric Matrix Completion

We assume that we are given a set of samples from some unknown matrix $\boldsymbol{M} \in \mathbb{R}^{m \times n}$, along with a binary indicator mask $\boldsymbol{S}$ that is 1 for measured samples and 0 for missing ones. In addition, we are given two graphs $\mathcal{G}_\mathrm{r}, \mathcal{G}_\mathrm{c}$, encoding relations between the rows and the columns, respectively. We represent the Laplacians of these graphs and their spectral decompositions by $\boldsymbol{L}_\mathrm{r} = \boldsymbol{\Phi}\boldsymbol{\Lambda}_\mathrm{r}\boldsymbol{\Phi}^\top$, $\boldsymbol{L}_\mathrm{c} = \boldsymbol{\Psi}\boldsymbol{\Lambda}_\mathrm{c}\boldsymbol{\Psi}^\top$. We minimize the objective function of the following form:

$$\min_{\boldsymbol{X}} \; E_\mathrm{data}(\boldsymbol{X}) + \mu E_\mathrm{reg}(\boldsymbol{X}) \tag{3}$$

with E_data denoting a data term of the form

$$E_\mathrm{data}(\boldsymbol{X}) = \|(\boldsymbol{X} - \boldsymbol{M}) \odot \boldsymbol{S}\|_F^2 \,, \tag{4}$$

As observed in Boyarski et al. [2020], we can decompose $\boldsymbol{X} = \boldsymbol{\Phi}\boldsymbol{C}\boldsymbol{\Psi}^\top$. Remarkably, the data term itself, as we show in our experiments later, when expressed through the functional map i.e.$\boldsymbol{X} = \boldsymbol{\Phi}\boldsymbol{C}\boldsymbol{\Psi}^\top$ already recovers low-rank matrices and outperforms the approach of Boyarski et al. [2020] on synthetic geometric experiments. Before we explain the choice and motivation of our regularizer E_reg, we explain next why the data term itself already works remarkably well on rank constrained geometric problems.

4.1 Low Rank Geometric Matrix Completion

Our first observation is that by using a reduced basis to represent a function $\boldsymbol{X}$ on the product space $\mathcal{G}_1 \square \mathcal{G}_2$ already provides a strong regularization, which can be sufficient to recover a low rank matrix approximation from a sparse signal.

Specifically, suppose that we constrain $\boldsymbol{X}$ to be a matrix such that $\boldsymbol{X} = \boldsymbol{\Phi}\boldsymbol{C}\boldsymbol{\Psi}^\top$ for some matrix $\boldsymbol{C}$. Note that if $\boldsymbol{\Phi}$ and $\boldsymbol{\Psi}$ have k columns each then $\boldsymbol{C}$ must be a $k \times k$ matrix. We would like to argue that solving Eq. (4) under the constraint that $\boldsymbol{X} = \boldsymbol{\Phi}\boldsymbol{C}\boldsymbol{\Psi}^\top$ will recover the underlying ground truth signal $\boldsymbol{Y}$ if it is low rank and satisfies an additional condition that we call basis consistency.

For this suppose that the ground truth hidden signal $\boldsymbol{M}$ has rank r. Consider its singular value decomposition $\boldsymbol{M} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^\top$. If $\boldsymbol{M}$ has rank r, then $\boldsymbol{\Sigma}$ is a diagonal matrix with r non-zero entries. We will call $\boldsymbol{M}$ *basis-consistent* with respect to $\boldsymbol{\Phi}, \boldsymbol{\Psi}$ if the first r left singular vectors U_r (i.e., those corresponding to non-zero singular values) lie in the span of $\boldsymbol{\Phi}$, and the first r right singular vectors V_r lie in the span of $\boldsymbol{\Psi}$. In other words, there exist some matrices $\boldsymbol{R}, \boldsymbol{Q}$ s.t. $U_r = \boldsymbol{\Phi}\boldsymbol{R}$ (note that this implies $k \geq r$) and $V_r = \boldsymbol{\Psi}\boldsymbol{Q}$.

Given this definition, it is easy to see that all basis-consistent matrices with rank $r \leq k$ can be represented by some functional map $\boldsymbol{C}$. In other words, given $\boldsymbol{Y}$ that is basis-consistent, there is some functional map $\boldsymbol{C}$ s.t. $Y = \boldsymbol{\Phi}\boldsymbol{C}\boldsymbol{\Psi}^T$. Conversely any $\boldsymbol{X} = \boldsymbol{\Phi}\boldsymbol{C}\boldsymbol{\Psi}^T$ has rank at most k and must be basis-consistent by construction.

Second, suppose we are optimizing Eq (4) under the constraint $\boldsymbol{X} = \boldsymbol{\Phi}\boldsymbol{C}\boldsymbol{\Psi}^\top$ and that the optimum, i.e., the ground truth matrix $\boldsymbol{M}$, is basis-consistent. Then since the energy $E(C)$ is convex and there are enough known samples to full constrain the corresponding linear system, then we are guaranteed to recover the optimum low-rank basis-consistent matrix. We note briefly that the argument above can also be made approximate, when the ground truth matrix is not exactly, but only approximately basis consistent, by putting appropriate error bounds.

This simple observation suggests that by restricting $\boldsymbol{X} = \boldsymbol{\Phi}\boldsymbol{C}\boldsymbol{\Psi}^\top$ and optimizing over the matrices $\boldsymbol{C}$ instead of $\boldsymbol{X}$ already provides a strong regularization that can help recover appropriate low-rank signals even without any other regularization. In practice, we observe that a weak additional regularization is often sufficient to obtain state-of-the-art results.

4.2 Functional Regularization

For clarity, we first describe the regularization terms introduced in Boyarski et al. [2020] briefly

$$E_{\text{reg}} = \mu_r E^r_{\text{dir}} + \mu_c E^c_{\text{dir}} + \rho_c E^c_{\text{diag}} + \rho_r E^r_{\text{diag}} \tag{5}$$

E^r_{dir} is the Dirichlet energy of $\boldsymbol{X}$ on row graph, given by $E^r_{\text{dir}}(\boldsymbol{X}) = \text{tr}\left(\boldsymbol{X}^\top \boldsymbol{L}_\text{r} \boldsymbol{X}\right)$. Similar term is used for column graph $E^c_{\text{dir}}(\boldsymbol{X}) = \text{tr}\left(\boldsymbol{X} \boldsymbol{L}_\text{c} \boldsymbol{X}^\top\right)$ In addition to Dirichlet energy, Boyarski et al. [2020] also introduces two regularization on the transformation matrix $\boldsymbol{P}, \boldsymbol{Q}$. As described earlier, the purpose of these transformation is to rotate the original eigen basis $\boldsymbol{\Phi}$ $\boldsymbol{\Psi}$ that will simplify the structure of $\boldsymbol{C}$. $E^r_{\text{diag}}(\boldsymbol{P}) = \|\text{off}\left(\boldsymbol{P}^\top \boldsymbol{\Lambda}_\text{r} \boldsymbol{P}\right)\|^2_F$, where $\text{off}(\cdot)$ denotes the off-diagonal elements. A similar treatment to the columns graph gives,$E^c_{\text{diag}}(\boldsymbol{Q}) = \|\text{off}\left(\boldsymbol{Q}^\top \boldsymbol{\Lambda}_\text{c} \boldsymbol{Q}\right)\|^2_F$.. We briefly mention that in addition to SGMC, Boyarski et al. [2020] also proposes a multi-resolution spectral loss named SGMC-Zoomout(SGMC-Z) with its own hyperparameters (step size between different resolutions) besides the four hyperparameters in Eq. 5.

In contrast to SGMC or SGMC-Z, our E_{reg} contains a single regularization term on the functional map induced between row space and column space described next.

Laplacian Commutativity as a Regularizer Our main idea is to use the simplest possible regularizer, which furthermore leads to a convex optimization problem and can achieve state-of-the-art results. For this we borrow a condition that is prominent in the functional map literature Ovsjanikov et al. [2016]. Namely, in the context of surfaces, the functional map is often expected to *commute with the Laplace-Beltrami operator*:

$$E_{\text{reg}} = \left\|\boldsymbol{C}\boldsymbol{\Lambda}_\text{r} - \boldsymbol{\Lambda}_\text{c}\boldsymbol{C}\right\|^2, \tag{6}$$

where $\boldsymbol{\Lambda}_\text{r}$ and $\boldsymbol{\Lambda}_\text{c}$ are diagonal matrices of Laplacian eigenvalues of the source and target.

For shape matching problems this constraint helps to find better mappings because functional maps that commute with the Laplacian must arise from near isometric point-to-point correspondences Rosenberg [1997], Ovsjanikov et al. [2012]. More broadly commutativity with the Laplacian imposes a diagonal structure of the functional map, which intuitively promotes preservation of low frequency eigenfunctions used in the basis. In the context of matrix completion this can be interpreted simply as approximate preservation of global low frequency signals defined on the two graphs.

Given these above definitions, our problem defined in Eq. (3) reduces to

$$\min_{\boldsymbol{C}} \|(\boldsymbol{X} - \boldsymbol{M}) \odot \boldsymbol{S}\|^2_F + \left\|\boldsymbol{C}\boldsymbol{\Lambda}_\text{r} - \boldsymbol{\Lambda}_\text{c}\boldsymbol{C}\right\|^2, \text{ where } \boldsymbol{X} = \boldsymbol{\Phi}\boldsymbol{C}\boldsymbol{\Psi}^\top \tag{7}$$

In practice, however, we observe faster convergence if we replace C with $\boldsymbol{P}\boldsymbol{C}\boldsymbol{Q}^\top$ and therefore, let all three be free variables.

4.3 Implementation

The optimization is carried out using gradient descent in Tensorflow Abadi et al. [2015].

Initialization Similar to Boyarski et al. [2020], we initialize the $\boldsymbol{P}$ and $\boldsymbol{Q}$ with an identity matrix with size equal to that of underlying matrix $\boldsymbol{M}$ corresponding to respective dataset and $\boldsymbol{C}$ by projecting $\boldsymbol{X} \odot \boldsymbol{S}$ on the first eigen vector of $\boldsymbol{L}_\text{c}$ and $\boldsymbol{L}_\text{r}$.

Hyperparameters Our formulation contains two hyperparameters namely the size of $\boldsymbol{C}$ and the weighing scalar μ. We divide the number of available entries in the matrix randomly into training and validation set in a 95 to 5 ratio respectively. We set μ to be $.00001$ and learning rate to $.000001$ for all the experiments. Size of C is different for different datasets and set according to the performance on the validation set of each dataset.

5 Results

This section is divided into two subsections. The goal of first subsection is to extensively compare between our approach and Spectral geometric matrix completion **(SGMC)**Boyarski et al. [2020] on

Table 1: Comparative results to test the dependence of SGMC and our method on the rank of the underlying random matrix of size 150×200

Rank	Ours	Ours-FM	SGMC
5	1e-7	2e-5	1e-4
10	2e-7	2e-5	2e-4
12	5e-7	4e-5	9e-4
15	6e-3	1e-3	1e-2
20	3e-2	1e-2	5e-2

Table 2: Comparative results to test the dependence of SGMC and our method on the density of the sampling set in $\%$ of the number of matrix elements, for a random rank 10 matrix of size 150×200.

Density	Ours	Ours-FM	SGMC
1	2e-2	2e-2	1e-1
5	8e-7	1e-3	5e-4
10	2e-7	5e-5	2e-4
20	1e-7	2e-5	1e-4

a synthetic example of a community structured graph. In the second subsection, we compare with all approaches on various real world recommendation benchmarks. Note that we use SGMC and Boyarski et al. [2020] interchangeably in this section.

5.1 Experimental study on synthetic dataset

For a fair comparison with Boyarski et al. [2020], we use the graphs taken from the synthetic Netflix dataset. Synthetic Netflix is a small synthetic dataset constructed by Kalofolias et al. [2014] and Monti et al. [2017], in which the user and item graphs have strong communities structure. It is useful in conducting controlled experiments to understand the behavior of geometry-exploiting algorithms. In all our tests, we use a randomly generated band-limited matrix on the product graph $\mathcal{G}_{\mathrm{c}} \square \mathcal{G}_{\mathrm{r}}$.

Baselines

- **Ours-FM**; This baseline only optimizes for $\boldsymbol{C}$ without any regularization. All results are obtained with $\boldsymbol{C}$ of size 30×30.
- **SGMC**: All results are obtained with their open source code with their optimal parameters.

Test Error. To evaluate the performance of the algorithms in this section, we report the *root mean squared error*,

$$\mathrm{RMSE}(\boldsymbol{X}, \boldsymbol{S}) = \sqrt{\frac{\|(\boldsymbol{X} - \boldsymbol{M}) \odot \boldsymbol{S}\|_F^2}{\sum_{i,j} \boldsymbol{S}_{i,j}}} \tag{8}$$

computed on the complement of the training set. Here $\boldsymbol{X}$ is the recovered matrix and $\boldsymbol{S}$ is the binary mask representing the support of the set on which the RMSE is computed.

We compare the two approaches on different constraints as follows:

Rank of the underlying matrix. We explore the effect of the rank of the underlying matrix, showing that as the rank increases upto 15 to 20, it becomes harder for both methods to recover the

Table 3: Comparative results to test the robustness of our method in the presence of noisy graphs.

Noise	Ours	Ours-FM	SGMC
5	1e-3	2e-3	5e-3
10	4e-3	3e-3	1e-2
20	6e-3	6e-3	1e-2

Table 4: Test error on Synthetic Netflix [Monti et al., 2017], Flixster [Jamali and Ester, 2010], and Movielens-100K [Harper and Konstan, 2016]

Model	**Synthetic Neflix**	**Flixster**	**ML-100K**
MC Candès and Recht [2009]	–	1.533	0.973
GMC Kalofolias et al. [2014]	0.3693	–	0.996
GRALS Rao et al. [2015]	0.0114	1.245	0.945
RGCNN Monti et al. [2017]	0.0053	0.926	0.929
GC-MC Berg et al. [2017]	–	0.917	0.910
Ours-FM	0.0064	1.02	1.12
DMF Arora et al. [2019]	0.0468	1.06	0.922
SGMC	0.0021	0.900	0.912
SGMC-Z	0.0036	0.888	0.913
Ours	0.0022	0.888	0.915

matrix. As the rank increases, the reconstruction error increases, but it increases slower for us than for SGMC. For the training set we used 10% of the points chosen at random (same training set for all experiments summarized in Table 1). We remark that Ours-FM consistently outperforms SGMC for all rank.

Sampling density. We investigate the effect of the number of samples on the reconstruction error. We demonstrate that in the data-poor regime, our regularization is strong enough to recover matrix, compared to performance achieved by incorporating geometric regularization through SGMC. These experiments are summarized in Table 2. Note that gap between us and SGMC remains high even when the sample density increases to 20%. Even when using 1% of the samples, we perform better than SGMC. We also remark that Ours-FM outperforms SGMC only when density is sufficient.

Noisy graphs. We study the effect of noisy graphs on the performance. We follow the same experimental setup as Boyarski et al. [2020] and perturb the edges of the graphs by adding random Gaussian noise with zero mean and tunable standard deviation to the adjacency matrix. Table 3 mentions the value of this tunable standard deviation. We discard the edges that became negative as a result of the noise, and symmetrized the adjacency matrix. Table 3 demonstrates that our method is robust to substantial amounts of noise in graphs. Surprisingly, Ours-FM demonstrates even stronger resilience to noise.

5.2 Results on recommender systems datasets

In addition to synthetic Netflix, we also validate our method on two more recommender systems datasets for which row and column graphs are available. Movielens-100K Harper and Konstan [2016] contains ratings of 1682 items by 943 users whereas Flixter [Jamali and Ester, 2010] contains ratings of 3000 items by 3000 users. All baseline numbers, except Ours-FM, are taken from Monti et al. [2017] and Boyarski et al. [2020].

Baselines

- **SGMC(Z)**: In addition to SGMC, Boyarski et al. [2020] also proposed a multi resolution spectral loss named SGMC-Zoomout.
- **DMF**: This is a matrix factorization approach that was adapted for matrix completion tasks by Boyarski et al. [2020]. Note that this approach does not incorporate any geometric information.
- **Ours-FM**: This method only optimizes the data term, over $\boldsymbol{C}$, without any regularization. All results are obtained with $\boldsymbol{C}$ of size 30×30.

We explain several observations from Table 4: First, our baseline, Ours-FM, obtains surprisingly good performance across datasets. This underscores the regularization brought in by the laplacian eigen basis of row and column graphs. Second, non geometric model such as DMF shows competitive performance with all the other methods on ML-100K. This suggests that the geometric information is not very useful for this dataset. Third, our proposed algorithm is competitive with the other methods while being simple and interpretable. On Synthetic Netflix, we obtain best results with randomization

on underlying graph structure. We explore its effect in detail in supplement. Furthermore, it should be noted that non geometric models such as DMF performs poorly on both synthetic datasets compared to ours and SGMC. Lastly, these experimental results validate the effectiveness of our single regularization when compared to the combination of several regularizations introduced in Boyarski et al. [2020].

Computation Issues Our method depends on the eigenvalue decomposition of graph Laplacian matrix which is the main bottleneck to scale our approach for large scale deployment. We intend to address this issue in our future work.

6 Conclusion

In this work, we propose a functional view for geometric matrix completion, building upon the recent work of Boyarski et al. [2020]. We establish empirically and theoretically that using a reduced basis to represent a function on the product space of two graphs already provides a strong regularization, which is sufficient to recover a low rank matrix approximation from a sparse signal. Moreover, we propose a novel regularization and show, through extensive experimentation on real and synthetic datasets, that our single regularization is very competitive when compared to the combination of several different regularizations proposed before.

7 Acknowledgement

Parts of this work were supported by the ERC Starting Grant StG-2017-758800 (EXPROTEA), KAUST OSR Award No. CRG-2017-3426, and a gift from Nvidia.

8 Broader Impact

In addition to collaborative filtering Feng et al. [2019], extracting geometric information from graph structured data is a core task in several domains from few shot learning Gidaris and Komodakis [2019], zero shot learning Wang et al. [2018] in computer vision, machine learning to knowledge graph based problems in NLP to name a few since graphs appear everywhere. Graph CNN forms backbone of all state-of-the-art solutions in such domains. Although it has recently been demonstrated that some graph CNN architectures can be greatly simplified, and still perform competitively on several graph analysis tasks Wu et al. [2019], our work takes a different approach and treats graphs as functional spaces that can be easily manipulated and analyzed using functional maps and is competitive with graph CNN in modelling geometric information. We believe this is a very promising research direction and future work in this direction has the potential to alter the course in geometric deep learning or graph signal processing in general.

References

Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. URL `https://www.tensorflow.org/`. Software available from tensorflow.org.

Sanjeev Arora, Nadav Cohen, Wei Hu, and Yuping Luo. Implicit regularization in deep matrix factorization. *arXiv preprint arXiv:1905.13655*, 2019.

Rianne van den Berg, Thomas N Kipf, and Max Welling. Graph convolutional matrix completion. *arXiv preprint arXiv:1706.02263*, 2017.

Amit Boyarski, Sanketh Vedula, and Alex Bronstein. Deep matrix factorization with spectral geometric regularization. *arXiv preprint arXiv:1911.07255*, 2020.

Emmanuel J Candès and Benjamin Recht. Exact matrix completion via convex optimization. *Foundations of Computational mathematics*, 9(6):717, 2009.

C. Feng, Z. Liu, S. Lin, and T. Q. S. Quek. Attention-based graph convolutional network for recommendation system, 2019.

Spyros Gidaris and Nikos Komodakis. Generating classification weights with gnn denoising autoencoders for few-shot learning. *CVPR*, 2019.

F Maxwell Harper and Joseph A Konstan. The movielens datasets: History and context. *Acm transactions on interactive intelligent systems (tiis)*, 5(4):19, 2016. URL `https://grouplens.org/datasets/movielens/`.

Mohsen Jamali and Martin Ester. A matrix factorization technique with trust propagation for recommendation in social networks. In *Proceedings of the fourth ACM conference on Recommender systems*, pages 135–142. ACM, 2010.

Vassilis Kalofolias, Xavier Bresson, Michael Bronstein, and Pierre Vandergheynst. Matrix completion on graphs. *arXiv preprint arXiv:1408.1717*, 2014.

Yehuda Koren, Robert Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42:30–37, 2009.

Or Litany, Emanuele Rodolà, Alexander M Bronstein, and Michael M Bronstein. Fully spectral partial shape matching. In *Computer Graphics Forum*, volume 36, pages 247–258. Wiley Online Library, 2017.

Federico Monti, Michael Bronstein, and Xavier Bresson. Geometric matrix completion with recurrent multi-graph neural networks. In *Advances in Neural Information Processing Systems*, pages 3697–3707, 2017.

Maks Ovsjanikov, Mirela Ben-Chen, Justin Solomon, Adrian Butscher, and Leonidas Guibas. Functional maps: a flexible representation of maps between shapes. *ACM Transactions on Graphics (TOG)*, 31(4):30, 2012.

Maks Ovsjanikov, Etienne Corman, Michael Bronstein, Emanuele Rodolà, Mirela Ben-Chen, Leonidas Guibas, Frederic Chazal, and Alex Bronstein. Computing and processing correspondences with functional maps. In *SIGGRAPH ASIA 2016 Courses*, page 9. ACM, 2016.

Nikhil Rao, Hsiang-Fu Yu, Pradeep K Ravikumar, and Inderjit S Dhillon. Collaborative filtering with graph information: Consistency and scalable methods. In *Advances in neural information processing systems*, pages 2107–2115, 2015.

Steven Rosenberg. *The Laplacian on a Riemannian manifold: an introduction to analysis on manifolds*, volume 31. Cambridge University Press, 1997.

Daniel Spielman. Spectral graph theory. *Lecture Notes, Yale University*, pages 740–0776, 2009.

Xiaolong Wang, Yufei Ye, and Abhinav Gupta. Zero-shot recognition via semantic embeddings and knowledge graphs. *CVPR*, 2018.

Felix Wu, Amauri Souza, Tianyi Zhang, Christopher Fifty, Tao Yu, and Kilian Weinberger. Simplifying graph convolutional networks. In *ICML*, pages 6861–6871. PMLR, 2019.