

A Manifold Proximal Linear Method for Sparse Spectral Clustering with Application to Single-Cell RNA Sequencing Data Analysis*

Zhongruo Wang[†] Bingyuan Liu[‡] Shixiang Chen[§] Shiqian Ma[†] Lingzhou Xue[‡]
Hongyu Zhao[¶]

October 30, 2020

Abstract

Spectral clustering is one of the fundamental unsupervised learning methods and is widely used in data analysis. Sparse spectral clustering (SSC) imposes sparsity to the spectral clustering and it improves the interpretability of the model. This paper considers a widely adopted model for SSC, which can be formulated as an optimization problem over the Stiefel manifold with nonsmooth and nonconvex objective. Such an optimization problem is very challenging to solve. Existing methods usually solve its convex relaxation or need to smooth its nonsmooth part using certain smoothing techniques. In this paper, we propose a manifold proximal linear method (ManPL) that solves the original SSC formulation. We also extend the algorithm to solve the multiple-kernel SSC problems, for which an alternating ManPL algorithm is proposed. Convergence and iteration complexity results of the proposed methods are established. We demonstrate the advantage of our proposed methods over existing methods via the single-cell RNA sequencing data analysis.

Keywords— Riemannian Optimization, Manifold Proximal Linear Method, Sparse Spectral Clustering, Single-Cell RNA Sequencing Data Analysis

1 Introduction

Clustering is a fundamental unsupervised learning problem with wide applications. The hierarchical clustering, K -means clustering and spectral clustering (SC) methods are widely used in practice [Friedman et al., 2001]. It is known that interpretation of the dendrogram in hierarchical clustering can be difficult in practice, especially for large datasets. The K -means clustering, closely related to Lloyd’s algorithm, does not guarantee to find the optimal solution and performs poorly for non-linearly separable or non-convex clusters. SC [Chung and Graham, 1997, Shi and Malik, 2000, Ng et al., 2002] is a graph-based clustering method and it provides a promising alternative for identifying locally connected clusters.

Given the data matrix $X = [\mathbf{x}_1, \dots, \mathbf{x}_n] \in \mathbb{R}^{p \times n}$, where n is the number of data points and p is the feature dimension, SC constructs a symmetric affinity matrix $S = (s_{ij})_{n \times n}$, where $s_{ij} \geq 0$ measures the pairwise similarity between two data samples \mathbf{x}_i and \mathbf{x}_j for $i, j = 1, \dots, n$. Denote diagonal matrix $D = \text{Diag}(d_1, \dots, d_n)$ with $d_i = \sum_{j=1}^n s_{ij}$. The main step of SC is to compute the following eigenvalue decomposition:

$$\min_{U \in \mathbb{R}^{n \times C}} \langle UU^\top, L \rangle, \text{ s.t., } U^\top U = I_C, \quad (1.1)$$

*The first two authors contributed equally to this paper.

[†]Department of Mathematics, University of California, Davis

[‡]Department of Statistics, The Pennsylvania State University

[§]Department of Industrial & Systems Engineering, Texas A&M University

[¶]Department of Biostatistics, Yale University

where $L = I_n - D^{-1/2}SD^{-1/2}$ is the normalized Laplacian matrix, I_C denotes the $C \times C$ identity matrix, and C is the number of clusters. The rows of U can be regarded as an embedding of the data X from \mathbb{R}^p to \mathbb{R}^C . The cluster assignment is then decided after using a standard clustering method such as the K -means clustering on the estimated embedding matrix \hat{U} obtained by solving (1.1). Ideally, \hat{U} should be a sparse matrix such that $\hat{U}_{ij} \neq 0$ if and only if the sample i belongs to the j -th cluster. Therefore $\hat{U}\hat{U}^\top$ should be a block diagonal matrix which is also sparse. To this end, the sparse spectral clustering (SSC) [Lu et al., 2016, 2018, Park and Zhao, 2018] is proposed to impose sparsity on $\hat{U}\hat{U}^\top$, which leads to the following optimization problem:

$$\min_{U \in \mathbb{R}^{n \times C}} \langle UU^\top, L \rangle + \lambda \|UU^\top\|_1, \text{ s.t., } U^\top U = I_C, \quad (1.2)$$

where $\|Z\|_1 = \sum_{ij} |Z_{ij}|$ is the entry-wise ℓ_1 norm of Z and it promotes the sparsity of Z , and $\lambda > 0$ is a weighting parameter.

In practice, the performance of SSC is sensitive to a single measure of similarity between data points, and there are no clear criteria to choose an optimal similarity measure. Moreover, for some very complex data such as the single-cell RNA sequencing (scRNA-seq) data [Kiselev et al., 2019], one may benefit from considering multiple similarity matrices because they provide more information to the data. The next-generation sequencing technologies provide large detailed catalogs of the transcriptomes of massive cells to identify putative cell types. Clustering high-dimensional scRNA-seq data provides an informative step to disentangle the complex relationship between different cell types. For example, it is important to characterize the patterns of monoallelic gene expression across mammalian cell types [Deng et al., 2014], explore the mechanisms that control the progression of lung progenitors across distinct cell types [Treutlein et al., 2014], or study the functionally distinct lineage in the bone marrow across mouse conventional dendritic cell types [Schlitzer et al., 2015]. To this end, Park and Zhao [2018] suggest the following similarity matrices which lead to multiple-kernel SSC (MKSSC):

$$K_{\delta,m}(i,j) = \exp\left(\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2\epsilon_{ij}^2}\right), \epsilon_{ij} = \frac{\delta(\mu_i + \mu_j)}{2}, \mu_i = \frac{\sum_{\ell \in \text{KNN}(i)} \|\mathbf{x}_i - \mathbf{x}_\ell\|}{m},$$

where $\text{KNN}(i)$ represents a set of sample indices that are the top m nearest neighbors of the sample \mathbf{x}_i . The parameters δ and m control the width of the neighborhoods. We use $S(\delta)$ and $S(m)$ to denote the sets of possible choices of δ and m , respectively. Then the total number of similarity matrices is equal to $T = |S(\delta)| \cdot |S(m)|$. We denote the normalized Laplacian matrices corresponding to these T similarity matrices as $L^{(\ell)}$, $\ell = 1, \dots, T$. The MKSSC can be formulated as the following optimization problem:

$$\begin{aligned} \min_{U \in \mathbb{R}^{n \times C}, w \in \mathbb{R}^T} \quad & \bar{F}(U, w) \equiv \left\langle UU^\top, \sum_{\ell=1}^T w_\ell L^{(\ell)} \right\rangle + \lambda \|UU^\top\|_1 + \rho \sum_{\ell=1}^T w_\ell \log(w_\ell) \\ \text{s.t.} \quad & U^\top U = I_C, \sum_{\ell=1}^T w_\ell = 1, w_\ell \geq 0, \ell = 1, \dots, T, \end{aligned} \quad (1.3)$$

where $w_\ell, \ell = 1, \dots, T$ are unknown weightings of the kernels, and $\rho \sum_{\ell=1}^T w_\ell \log(w_\ell)$ serves as an entropy regularization term, and λ, ρ are two regularization parameters.

Note that both SSC (1.2) and MKSSC (1.3) are nonconvex and nonsmooth with Riemannian manifold constraints. Therefore, they are both numerically challenging to solve. In this paper, we propose a manifold proximal linear (ManPL) method for solving the SSC (1.2) in Section 2, and an alternating ManPL (AManPL) method for solving the MKSSC (1.3) in Section 4. The convergence rates of ManPL and AManPL are rigorously analyzed. Moreover, we present numerical results in Section 5 to demonstrate the advantages of ManPL and AManPL in benchmark datasets, synthetic datasets, and real datasets in scRNA-seq data analysis.

Our contributions lie in several folds.

- (i) We propose ManPL method for solving SSC (1.2) and AManPL method for solving MKSSC (1.3).
- (ii) We analyze the convergence and iteration complexity of both ManPL and AManPL.

- (iii) We propose a proximal point algorithm based on a semi-smooth Newton method for solving the convex subproblems arising from ManPL and AManPL.
- (iv) We apply our proposed methods to clustering of scRNA-seq data.

Notation. Throughout this paper, we use \mathcal{M} to denote the Stiefel manifold. The smoothness, convexity, and Lipschitz continuity of a function f are always interpreted as the function is considered in the ambient Euclidean space. We use \mathbb{S}_+^n to denote the set of $n \times n$ positive semidefinite matrices, and $\text{Tr}(Z)$ to denote the trace of matrix Z .

2 A Manifold Proximal Linear Method for SSC

Since SSC (1.2) is both nonsmooth and nonconvex, it is numerically challenging to solve. In the literature, convex relaxations and smooth approximations of (1.2) have been suggested. In particular, Lu et al. [2016] proposed to replace UU^\top with a positive semidefinite matrix P and solve the following convex relaxation:

$$\min_{P \in \mathbb{S}_+^n} \langle P, L \rangle + \lambda \|P\|_1, \text{ s.t.}, 0 \preceq P \preceq I, \text{Tr}(P) = C. \quad (2.1)$$

This convex problem (2.1) can be solved by classical optimization algorithms such as ADMM. Denote the solution of (2.1) by \hat{P} , the solution of (1.2) can be approximated by the top C eigenvectors of \hat{P} . In another work, Lu et al. [2018] proposed a nonconvex ADMM to solve the following smooth variant of (1.2):

$$\min_{U \in \mathbb{R}^{n \times C}, P \in \mathbb{S}_+^n} \langle UU^\top, L \rangle + g_\sigma(P), \text{ s.t.}, P = UU^\top, U^\top U = I_C, \quad (2.2)$$

where $g_\sigma(\cdot)$ is a smooth function with smoothing parameter $\sigma > 0$ that approximates the ℓ_1 regularizer $\lambda \|\cdot\|_1$. In [Lu et al., 2018], the authors used the following smooth function:

$$g_\sigma(P) := \max_Z \langle P, Z \rangle - \frac{\sigma}{2} \|Z\|_F^2, \text{ s.t.}, \|Z\|_\infty \leq \lambda, \quad (2.3)$$

where $\|Z\|_\infty = \max_{ij} |Z_{ij}|$. The nonconvex ADMM for solving (2.2) typically iterates as

$$U^{k+1} := \underset{U \in \mathbb{R}^{n \times C}}{\text{argmin}} \mathcal{L}(U, P^k; \Lambda^k), \text{ s.t.}, U^\top U = I_C, \quad (2.4a)$$

$$P^{k+1} := \underset{P \in \mathbb{S}_+^n}{\text{argmin}} \mathcal{L}(U^{k+1}, P; \Lambda^k), \quad (2.4b)$$

$$\Lambda^{k+1} := \Lambda^k - \mu(P^{k+1} - U^{k+1}U^{k+1\top}), \quad (2.4c)$$

where the augmented Lagrangian function \mathcal{L} is defined as

$$\mathcal{L}(U, P; \Lambda) := \langle UU^\top, L \rangle + g_\sigma(P) - \langle \Lambda, P - UU^\top \rangle + \frac{\mu}{2} \|P - UU^\top\|_F^2,$$

and $\mu > 0$ is a penalty parameter. The two subproblems (2.4a) and (2.4b) are both relatively easy to solve. The reason to use the smooth function $g_\sigma(\cdot)$ to approximate $\lambda \|\cdot\|_1$ in (2.2) is for the purpose of convergence guarantee. In [Lu et al., 2018], the authors proved that any limit point of the sequence generated by the nonconvex ADMM (2.4) is a stationary point of (2.2). This result relies on the fact that function g_σ is smooth. If one applies ADMM to the original SSC (1.2), then no convergence guarantee is known.

Note that both the convex relaxation (2.1) and the smooth approximation (2.2) are only approximations to the original SSC (1.2). In this section, we introduce our ManPL algorithm that solves the original SSC (1.2) directly. For the ease of presentation, we rewrite (1.2) as

$$\min_U F(U) \equiv f(U) + h(c(U)), \text{ s.t.}, U \in \mathcal{M}, \quad (2.5)$$

where $f(U) = \langle UU^\top, L \rangle$, $h(\cdot) = \lambda \|\cdot\|_1$, $c(U) = UU^\top$, $\mathcal{M} = \{U \in \mathbb{R}^{n \times C} \mid U^\top U = I_C\}$ is the Stiefel manifold. Moreover, note that f and c are smooth mappings, and h is nonsmooth but convex in the ambient Euclidean

space. Therefore, (2.5) is a Riemannian optimization problem with nonsmooth and nonconvex objective function. Furthermore, throughout this paper, we use L_f, L_c, L_h to denote the Lipschitz constants of $\nabla f, \nabla c$, and h , respectively. Riemannian optimization has drawn much attention recently, due to its wide applications, including low rank matrix completion [Boumal and Absil, 2011], phase retrieval [Bendory et al., 2018, Sun et al., 2018], phase synchronization [Boumal, 2016, Liu et al., 2017], and dictionary learning [Cherian and Sra, 2017, Sun et al., 2017]. Several important classes of algorithms for Riemannian optimization with a smooth objective function were covered in the monograph [Absil et al., 2008]. On the other hand, there has been very limited number of algorithms for Riemannian optimization with nonsmooth objective until very recently. The most natural idea for this class of optimization problems is the Riemannian subgradient method (RSGM) [Ferreira and Oliveira, 1998, Grohs and Hosseini, 2016, Hosseini and Uchmajew, 2017]. Recently, Li et al. [2019] studied the RSGM for Riemannian optimization with weakly convex objective. In particular, they showed that the number of iterations needed by RSGM for obtaining an ϵ -stationary point is $O(\epsilon^{-4})$. Motivated by the proximal gradient method for solving composite minimization in Euclidean space, Chen et al. [2020c] proposed a manifold proximal gradient method (ManPG) for solving the following Riemannian optimization problem:

$$\min_X f(X) + h(X), \text{ s.t., } X \in \mathcal{M}, \quad (2.6)$$

where \mathcal{M} is the Stiefel manifold, f is a smooth function, and h is a nonsmooth and convex function. A typical iteration of ManPG for solving (2.6) is:

$$V^k := \operatorname{argmin}_V \langle \nabla f(X^k), V \rangle + h(X^k + V) + \frac{1}{2t} \|V\|_F^2, \text{ s.t., } V \in \mathbb{T}_{X^k} \mathcal{M}, \quad (2.7a)$$

$$X^{k+1} := \operatorname{Retr}_{X^k}(\alpha_k V^k), \quad (2.7b)$$

where $\mathbb{T}_U \mathcal{M}$ denotes the tangent space of \mathcal{M} at U , $\alpha_k > 0$ is a step size, and Retr denotes the retraction operation. For the Stiefel manifold, $\mathbb{T}_U \mathcal{M} := \{V \in \mathbb{R}^{n \times C} \mid V^\top U + U^\top V = 0\}$. Comparing with (2.6), we note that (2.5) is more difficult to solve, because of the nonconvex term $c(U)$. In fact, ManPG cannot be used to solve the SSC (1.2) because of the existence of the nonconvex term UU^\top composite with the ℓ_1 norm. As a result, new algorithm is demanded for solving SSC (1.2). The iteration complexity of ManPG is proved to be $O(\epsilon^{-2})$ for obtaining an ϵ -stationary point of (2.6) [Chen et al., 2020c], which is better than the complexity of RSGM [Li et al., 2019]. Variants of ManPG have been designed for different applications, such as alternating ManPG for sparse PCA and sparse CCA [Chen et al., 2020b], FISTA for sparse PCA [Huang and Wei, 2019a], manifold proximal point algorithm for robust subspace recovery and orthogonal dictionary learning [Chen et al., 2019, 2020a], and stochastic ManPG [Wang et al., 2020] for online sparse PCA. Moreover, ManPG has been extended to more general Riemannian proximal gradient method [Huang and Wei, 2019b]. Motivated by the success of ManPG and its variants, we propose a manifold proximal linear algorithm for solving SSC (2.5).

The proximal linear method has recently drawn great research attentions. It targets to solve the optimization problem in the form of (2.5) without the manifold constraint, i.e.,

$$\min_{x \in \mathbb{R}^n} f(x) + h(c(x)), \quad (2.8)$$

where $f : \mathbb{R}^n \rightarrow \mathbb{R}$ and $c : \mathbb{R}^n \rightarrow \mathbb{R}^m$ are smooth mappings, $h : \mathbb{R}^m \rightarrow \mathbb{R}$ is convex and nonsmooth. The proximal linear method for solving (2.8) iterates as follows:

$$x^{k+1} := \operatorname{argmin}_x \langle \nabla f(x^k), x - x^k \rangle + h(c(x^k)) + J(c(x^k))(x - x^k) + \frac{1}{2t} \|x - x^k\|_2^2, \quad (2.9)$$

where $J(x) = \nabla c(x)$ is the Jacobian of c , and $t > 0$ is a step size. Note that since h is convex, the update (2.9) is a convex problem. This method has been studied recently by [Lewis and Wright, 2016, Drusvyatskiy and Paquette, 2018, Duchi and Ruan, 2018] and applied to solving many important applications such as robust phase retrieval [Duchi and Ruan, 2017], robust matrix recovery [Charisopoulos et al., 2019a], and robust blind deconvolution [Charisopoulos et al., 2019b].

Due to the nonconvex constraint $U \in \mathcal{M}$, solving (2.5) is more difficult than (2.8). Motivated by ManPG and the proximal linear method (2.9), we propose a ManPL algorithm for solving (2.5). A typical iteration of

the ManPL algorithm for solving (2.5) is:

$$V^k := \operatorname{argmin}_V \langle \nabla f(U^k), V \rangle + h(c(U^k) + J(U^k)V) + \frac{1}{2t} \|V\|_F^2, \text{ s.t., } V \in \mathbf{T}_{U^k} \mathcal{M}, \quad (2.10a)$$

$$U^{k+1} := \operatorname{Retr}_{U^k}(\alpha_k V^k). \quad (2.10b)$$

Similar to (2.7a), the equation (2.10a) computes the descent direction V by minimizing a convex function over the tangent space of \mathcal{M} . However, solving (2.10a) is more difficult than (2.7a) because of the non-trivial affine function, i.e., $c(U^k) + J(U^k)V$, composite with the nonsmooth function h . Moreover, the difference of (2.10a) and (2.9) is the constraint in (2.10a), which is needed in the Riemannian optimization setting. Fortunately, (2.10a) can still be solved efficiently by a proximal point algorithm combined with a semi-smooth Newton method, which will be elaborated in Section 3. The retraction step (2.10b) brings the iterate back to the manifold \mathcal{M} .

The complete description of the ManPL for solving SSC (2.5) is given in Algorithm 1. The step (2.11) is a line search step to find the step size α_k such that there is a sufficient decrease on the function F .

Algorithm 1 The ManPL for SSC (2.5)

Input: initial point $U^0 \in \mathcal{M}$, parameters $\gamma \in (0, 1)$, $t > 0$

for $k = 0, 1, \dots$ **do**

 Calculate V^k by solving (2.10a)

 Let j_k be the smallest nonnegative integer such that

$$F(\operatorname{Retr}_{U^k}(\gamma^{j_k} V^k)) \leq F(U^k) - \frac{\gamma^{j_k}}{2t} \|V^k\|_F^2 \quad (2.11)$$

 Let $\alpha_k = \gamma^{j_k}$ and compute U^{k+1} by (2.10b)

end for

The main convergence and iteration complexity result of ManPL (Algorithm 1) is given in Theorem 2.1. Its proof is given in the appendix.

Theorem 2.1. *Assume $F(U)$ is lower bounded by F^* . The limit point of the sequence $\{U^k\}$ generated by ManPL (Algorithm 1) is a stationary point of (2.5). Moreover, ManPL returns an ϵ -stationary point of (2.5) in $O(\epsilon^{-2})$ iterations.*

3 A Semi-Smooth Newton-based Proximal Point Algorithm for the Subproblem

In this section, we introduce a proximal point algorithm (PPA) combined with a semi-smooth Newton method (SSN) for solving the subproblem (2.10a) in ManPL. The notion of semi-smoothness was originally introduced by [Mifflin, 1977] for real valued functions and later extended to vector-valued mappings by [Qi and Sun, 1993]. The SSN method has recently received significant amount of attention due to its success in solving structured convex problems to a high accuracy in problems such as LASSO [Li et al., 2018, Yang et al., 2013], convex clustering [Wang et al., 2010], SDP [Zhao et al., 2010], and convex composite problems [Xiao et al., 2018].

For simplicity of the notation, we omit the index k in (2.10a), and denote $Z_1 := \nabla f(U^k)$, $Z_2 := c(U^k)$, $J = J(U^k)$, and operator $\mathcal{A} : V \rightarrow V^\top U^k + (U^k)^\top V$. Therefore, (2.10a) reduces to the following form:

$$\min_{V, Y} \frac{1}{2t} \|V + Z_1\|_F^2 + h(Z_2 + Y), \text{ s.t., } \mathcal{A}(V) = 0, Y = JV. \quad (3.1)$$

Note that we have introduced a variable Y to replace JV . The Lagrangian function for (3.1) is given by:

$$\mathcal{L}(V, Y; \Gamma_1, \Gamma_2) = \frac{1}{2t} \|V + Z_1\|_F^2 + h(Z_2 + Y) - \langle \Gamma_1, \mathcal{A}(V) \rangle - \langle \Gamma_2, JV - Y \rangle,$$

where Γ_1 and Γ_2 are the Lagrange multipliers associated to the two equality constraints. Therefore, (3.1) is equivalent to

$$\min_{V,Y} \{G(V,Y) := \max_{\Gamma_1,\Gamma_2} \mathcal{L}(V,Y;\Gamma_1,\Gamma_2)\}. \quad (3.2)$$

The minimization problem (3.2) can be solved by a PPA, which iterates as:

$$\begin{aligned} (V^{k+1}, Y^{k+1}) &:= \operatorname{argmin}_{V,Y} G(V^k, Y^k) + \frac{1}{2\beta} (\|V - V^k\|_F^2 + \|Y - Y^k\|_F^2) \\ &:= \operatorname{argmin}_{V,Y} \max_{\Gamma_1,\Gamma_2} \mathcal{L}(V,Y;\Gamma_1,\Gamma_2) + \frac{1}{2\beta} (\|V - V^k\|_F^2 + \|Y - Y^k\|_F^2), \end{aligned} \quad (3.3)$$

where $\beta > 0$ is a parameter. The problem (3.3) is equivalent to:

$$\max_{\Gamma_1,\Gamma_2} \min_{V,Y} \mathcal{L}(V,Y;\Gamma_1,\Gamma_2) + \frac{1}{2\beta} (\|V - V^k\|_F^2 + \|Y - Y^k\|_F^2). \quad (3.4)$$

Note that the minimization part of (3.4) is strongly convex and admits a closed-form solution given by:

$$V = \frac{t\beta}{t+\beta} \left(\mathcal{A}^*(\Gamma_1) + J^\top \Gamma_2 - \frac{1}{t} Z_1 + \frac{1}{\beta} V^k \right), \quad Y = \operatorname{Prox}_{\beta h}(Z_2 + Y^k - \beta \Gamma_2) - Z_2, \quad (3.5)$$

where Prox_g denotes the proximal mapping of function g , which is defined as:

$$\operatorname{Prox}_g(Z) := \operatorname{argmin}_X g(X) + \frac{1}{2} \|X - Z\|_F^2.$$

For simplicity of the notation, we define function $E(\Gamma_2) := Z_2 + Y^k - \beta \Gamma_2$. Substituting (3.5) to (3.4), and using the Moreau identity, we know that (3.4) is equivalent to:

$$\begin{aligned} \max_{\Gamma_1,\Gamma_2} \Theta(\Gamma_1,\Gamma_2) &:= -\frac{1}{2} \frac{t\beta}{t+\beta} \left\| \frac{1}{t} Z_1 - \frac{1}{\beta} V^k - \mathcal{A}^*(\Gamma_1) - J^\top \Gamma_2 \right\|_F^2 \\ &\quad + h(\operatorname{Prox}_{\beta h} E(\Gamma_2)) + \beta \|\operatorname{Prox}_{h^*/\beta}(E(\Gamma_2)/\beta)\|_F^2 + \langle \Gamma_2, Y^k \rangle - \frac{\beta}{2} \|\Gamma_2\|_F^2, \end{aligned} \quad (3.6)$$

where h^* denotes the conjugate function of h . Now by denoting

$$\Psi(\Gamma_2) := \max_{\Gamma_1} \Theta(\Gamma_1,\Gamma_2), \quad (3.7)$$

it is easy to verify that $\Psi(\Gamma_2)$ is strongly concave and continuously differentiable [Li et al., 2018], and its unique maximizer is found by solving the following nonsmooth system:

$$\nabla \Psi(\Gamma_2) = 0. \quad (3.8)$$

Solving (3.8) can be done by using SSN [Li et al., 2018, Xiao et al., 2018]. After we obtain the solution to (3.8), the optimal Γ_1 can be found by solving the maximization problem in (3.7), which is an easy least-squares problem.

To summarize, the PPA for solving (2.10a) is given by (3.3), and its solution is given by (3.5). The required Γ_2 in (3.5) is obtained by solving (3.8) using SSN, and Γ_1 in (3.5) is obtained by solving the least-squares problem in (3.7). The convergence of the PPA and the SSN has been well studied in the literature [Li et al., 2018, Yang et al., 2013].

4 An Alternating ManPL Method for Multiple-Kernel SSC

In this section, we consider the multiple-kernel SSC (1.3). Park and Zhao [2018] consider to solve the following relaxation of (1.3) by letting $P = UU^\top$:

$$\begin{aligned} \min_{P,w} \quad & \left\langle P, \sum_{\ell=1}^T w_\ell L^{(\ell)} \right\rangle + \lambda \|P\|_1 + \rho \sum_{\ell=1}^T w_\ell \log(w_\ell) \\ \text{s.t.} \quad & \operatorname{Tr}(P) = C, 0 \preceq P \preceq I, \sum_{\ell=1}^T w_\ell = 1, w_\ell \geq 0, \ell = 1, \dots, T. \end{aligned} \quad (4.1)$$

Note that this is still a nonconvex problem due to the bilinear term in the objective function. Park and Zhao [2018] suggested to use an alternating minimization algorithm (AMA) to solve (4.1). Note that this method is named MPSSC in [Park and Zhao, 2018]. In the k -th iteration of AMA, one first fixes w as w^k and solves the resulting problem with respect to P to obtain P^{k+1} , and then fixes P as P^{k+1} and solves the resulting problem with respect to w to obtain w^{k+1} . In particular, when w is fixed as w^k , problem (4.1) reduces to

$$\min_P \left\langle P, \sum_{\ell=1}^T w_\ell^k L^{(\ell)} \right\rangle + \lambda \|P\|_1, \text{ s.t., } \text{Tr}(P) = C, 0 \preceq P \preceq I, \quad (4.2)$$

which is a convex problem and can be solved via convex ADMM algorithm. When P is fixed as P^{k+1} , problem (4.1) reduces to

$$\min_w c^\top w + \rho \sum_{\ell=1}^T w_\ell \log(w_\ell), \text{ s.t., } \sum_{\ell=1}^T w_\ell = 1, w_\ell \geq 0, \ell = 1, \dots, T, \quad (4.3)$$

where $c_\ell = \langle P^{k+1}, L^{(\ell)} \rangle$, $\ell = 1, \dots, T$. This is also a convex problem and it can be easily verified that (4.3) admits a closed-form solution given by

$$w_\ell = \frac{\exp(-c_\ell/\rho)}{\sum_{j=1}^T \exp(-c_j/\rho)}, \ell = 1, \dots, T. \quad (4.4)$$

In summary, a typical iteration of the AMA algorithm proposed by [Park and Zhao, 2018] is as follows:

$$\begin{cases} \text{update } P^{k+1} \text{ by solving (4.2)} \\ \text{update } w^{k+1} \text{ by solving (4.3)}. \end{cases} \quad (4.5)$$

Another approach to approximate (1.3) is to combine the idea of AMA (4.5) and the nonconvex ADMM for solving the smooth problem (2.2). In particular, one can solve the following smooth variant of (1.3):

$$\begin{aligned} \min_{U \in \mathbb{R}^{n \times C}, w \in \mathbb{R}^T} & \left\langle UU^\top, \sum_{\ell=1}^T w_\ell L^{(\ell)} \right\rangle + g_\sigma(UU^\top) + \rho \sum_{\ell=1}^T w_\ell \log(w_\ell) \\ \text{s.t.} & \quad U^\top U = I_C, \sum_{\ell=1}^T w_\ell = 1, w_\ell \geq 0, \ell = 1, \dots, T, \end{aligned} \quad (4.6)$$

where $g_\sigma(\cdot)$ is the smooth approximation to $\lambda \|\cdot\|_1$ defined in (2.3). When fixing w , (4.6) is in the same form as the smoothed SSC (2.2), so it can be solved by the nonconvex ADMM (2.4). When fixing U , (4.6) is in the same form as (4.3), and admits a closed-form solution (4.4). In summary, the AMA+ADMM algorithm for solving (4.6) works as follows:

$$\begin{cases} \text{update } U^{k+1} \text{ by solving (4.6) with } w \text{ fixed as } w^k \text{ using nonconvex ADMM (2.4)} \\ \text{update } w^{k+1} \text{ by solving (4.6) with } U \text{ fixed as } U^{k+1} \text{ using (4.4)}. \end{cases} \quad (4.7)$$

By exploiting the structure of (1.3), we propose to solve (1.3) by an alternating ManPL algorithm (AManPL). More specifically, in the k -th iteration of AManPL, we first fix w as w^k , then (1.3) reduces to

$$\min_U \left\langle UU^\top, \sum_{\ell=1}^T w_\ell^k L^{(\ell)} \right\rangle + \lambda \|UU^\top\|_1, \text{ s.t., } U \in \mathcal{M}, \quad (4.8)$$

which is in the same form of (2.5) with L in (2.5) replaced by $\bar{L} := \sum_{\ell=1}^T w_\ell^k L^{(\ell)}$. Therefore, (4.8) can also be solved by ManPL. Here we adopt one step of ManPL, i.e., (2.10) to obtain U^{k+1} . More specifically, U^{k+1} is computed by the following two steps:

$$V^k := \underset{V}{\text{argmin}} \langle \nabla_U f(U^k, w^k), V \rangle + h(c(U^k) + J(U^k)V) + \frac{1}{2t} \|V\|_F^2, \text{ s.t., } V \in \text{T}_{U^k} \mathcal{M}, \quad (4.9a)$$

$$U^{k+1} := \text{Retr}_{U^k}(\alpha_k V^k), \quad (4.9b)$$

where $f(U, w) := \langle UU^\top, \sum_{\ell=1}^T w_\ell L^{(\ell)} \rangle$, $h(\cdot) := \lambda \|\cdot\|_1$, and $c(U) = UU^\top$. Note that (4.9a) can be solved by the same PPA+SSN algorithm discussed in Section 3. We then fix U in (1.3) as U^{k+1} , and then (1.3) reduces to

$$\min_w c^\top w + \rho \sum_{\ell=1}^T w_\ell \log(w_\ell), \text{ s.t., } \sum_{\ell=1}^T w_\ell = 1, w_\ell \geq 0, \ell = 1, \dots, T, \quad (4.10)$$

where $c_\ell = \langle U^{k+1}U^{k+1\top}, L^{(\ell)} \rangle$, $\ell = 1, \dots, T$. We then obtain w^{k+1} by solving (4.10), which admits a closed-form solution given by (4.4). The AManPL is described in Algorithm 2.

Algorithm 2 The AManPL Method for Solving MKSSC (1.3)

Input: parameter $\gamma \in (0, 1)$, initial point $U^0 \in \mathcal{M}$, let w^0 be the optimal solution to (4.10) for $c_\ell = \langle U^0U^{0\top}, L^{(\ell)} \rangle$

for $k = 0, 1, \dots$ **do**

 Calculate V^k by solving (4.9a)

 Let j_k be the smallest nonnegative integer such that

$$\bar{F}(\text{Retr}_{U^k}(\beta^{j_k} V^k), w^k) \leq \bar{F}(U^k, w^k) - \frac{\gamma^{j_k}}{2t} \|V^k\|_F^2 \quad (4.11)$$

 Let $\alpha_k = \gamma^{j_k}$ and compute U^{k+1} by (4.9b)

 Update w_ℓ^{k+1} by (4.4) with $c_\ell = \langle U^{k+1}U^{k+1\top}, L^{(\ell)} \rangle$, $\ell = 1, \dots, T$

end for

We have the following convergence and iteration complexity result for AManPL for solving MKSSC (1.3). Its proof is given in the appendix.

Theorem 4.1. *Assume $\bar{F}(U, w)$ in (1.3) is lower bounded by \bar{F}^* . The limit point of the sequence $\{U^k, w^k\}$ generated by AManPL (Algorithm 2) is a stationary point of problem (1.3). Moreover, to obtain an ϵ -stationary point of problem (1.3), the number of iterations needed by AManPL is $O(\epsilon^{-2})$.*

5 Numerical Experiments

In this section, we compare our proposed methods ManPL and AManPL with some existing methods for solving SSC and MKSSC. In particular, for SSC (1.2), we compare ManPL (Algorithm 1) with convex ADMM [Lu et al., 2016] (denoted by CADMM¹) for solving (2.1) and nonconvex ADMM [Lu et al., 2018] (denoted by NADMM) for solving (2.2). We also include the spectral clustering (denoted by SC) in the comparison. For MKSSC (1.3), we compare AManPL (Algorithm 2) with MPSSC (i.e., AMA+CADMM²) [Park and Zhao, 2018] and AMA+NADMM (4.7). All the algorithms were terminated when the absolute change of the objective value is smaller than 10^{-5} , which indicates that the algorithms were not making much progress.

5.1 UCI Datasets

We first compare the clustering performance of different methods on three benchmark datasets in UCI machine learning repository [Dua and Graff, 2017]. We list the parameters used in the algorithms in Table 1. These parameters are either suggested in the corresponding papers, or are tuned from a set of values and the ones that yielded the best normalized mutual information (NMI) score (see Table 2) were chosen. For instance, λ in CADMM was chosen from the set $\{10^{-3}, 10^{-4}, 10^{-5}\}$ and this was suggested in [Lu et al., 2016]. The values of λ and ρ in AMA+CADMM were suggested in [Park and Zhao, 2018]. We follow [Park and Zhao, 2018] to construct the similarity matrices and record the NMI scores to measure the performance

¹codes downloaded from <https://github.com/canyilu/LibADMM/blob/master/algorithms/sparsesc.m>

²codes downloaded from <https://github.com/ishspsy/project/tree/master/MPSSC>

Problem	Algorithm	Parameters
Convex SSC (2.1)	CADMM [Lu et al., 2016]	$\lambda = 10^{-4}$
Smoothed SSC (2.2)	NADMM (2.4) [Lu et al., 2018]	$\lambda = 10^{-4}, \sigma = 10^{-2}$
Original SSC (1.2)	ManPL (Algorithm 1)	$\lambda = 10^{-3}$
MKSSC (4.1)	AMA+CADMM (4.5) [Park and Zhao, 2018]	$\lambda = 10^{-4}, \rho = 0.2$
Smoothed MKSSC (4.6)	AMA+NADMM (4.7)	$\lambda = 10^{-4}, \rho = 0.2, \sigma = 10^{-2}$
Original MKSSC (1.3)	AManPL (Algorithm 2)	$\lambda = 5 \times 10^{-3}, \rho = 1$

Table 1: Algorithms and their parameters

of the clustering. Note that higher NMI scores indicate better clustering performance. The NMI scores are reported in Table 2. From Table 2 we see that the three algorithms for SSC usually outperform SC, and among the three algorithms for SSC, ManPL usually outperforms the other two methods. We also see that the MKSSC model usually generates higher NMI scores than SSC. Moreover, among all three algorithms for SSC and three algorithms for MKSSC, the AManPL for MKSSC always has the largest NMI score. This indicates the great potential of our AManPL method for solving MKSSC. Moreover, we show the heatmap of $|UU^T|$ for the Wine data set in Figure 1. From this figure we see that for SSC, the NADMM and ManPL generate much better results than CADMM in terms of recovering the block matrices, and for MKSSC, the AMA+NADMM and AManPL generate much better results than AMA+CADMM. The heatmaps for the other two data sets are similar, so we omit them for brevity.

Datasets	SC	SSC			MKSSC			C
		CADMM	NADMM	ManPL	AMA+CADMM	AMA+NADMM	AManPL	
Wine	0.8650	0.8650	0.8782	0.8782	0.8854	0.8854	0.8926	3
Iris	0.7496	0.7582	0.7582	0.7582	0.7665	0.7601	0.7705	3
Glass	0.3165	0.3418	0.2047	0.3471	0.2656	0.3315	0.3644	6

Table 2: Comparison of the NMI scores on the UCI data sets.

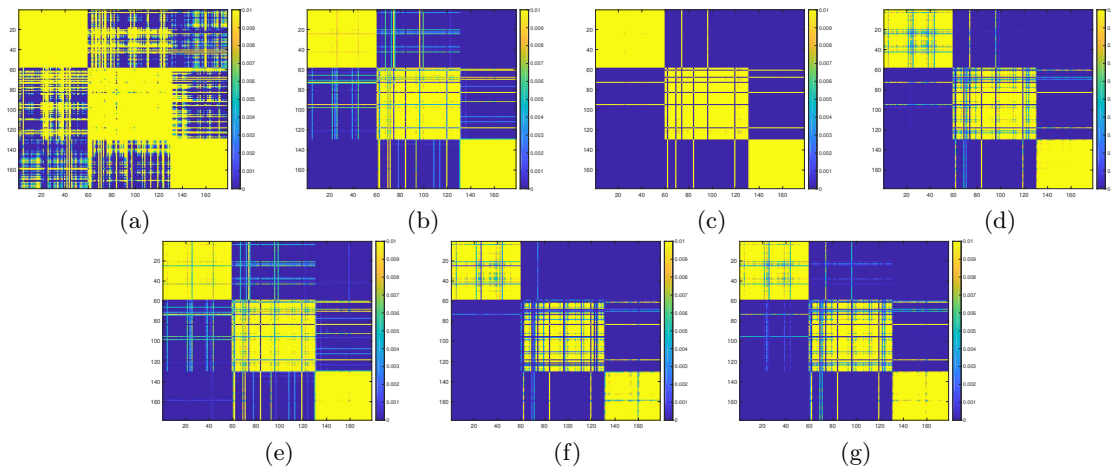


Figure 1: The heatmaps of $|UU^T|$ on the Wine dataset estimated by (a) SC, (b) CADMM for SSC, (c) NADMM for SSC, (d) ManPL for SSC, (e) AMA+CADMM for MKSSC, (f) AMA+NADMM for MKSSC, and (g) AManPL for MKSSC.

5.2 Synthetic Data

In this subsection, we follow [Park and Zhao, 2018] to evaluate the clustering performance of different methods on two synthetic datasets with $C = 5$ clusters. To compare ManPL, AManPL and existing methods, we select the regularization parameter λ using an independent tuning dataset and set other parameters as in Table 1. Specifically, we generate the tuning datasets using the same data generating process, and we select the parameter λ that maximizes the average NMI score over 50 independent repetitions. The two synthetic datasets are generated as follows.

- **Synthetic data 1.** We randomly generate C points in the 2-dimensional latent space spanning a circle as the centers of C clusters. For each cluster, we randomly generate the points by adding an independent noise to its center. We project these 2-dimensional data to a p -dimensional space using a linear projection matrix and then add the heterogeneous noise to obtain the data matrix X . The noise level is 30% of the radius of the circle in the embedded space.
- **Synthetic data 2.** We randomly generate a matrix $B' \in \mathbb{R}^{C \times d}$ by drawing its entries independently from Gaussian distributions, where $d < p$ and different rows of B' specify heterogeneous variances. We randomly assign the cluster labels $z_1, \dots, z_n \in [C]$. Let $B = [B', 0_{C \times (p-d)}]$ and $Z = (Z_{ij})_{n \times C} = (1_{\{z_i=j\}})_{n \times C}$. We generate $X = ZB + W$, where W is a noise matrix with independent standard normally distributed entries. The noise level is 20% of the radius of the circle in the embedded space.

Figure 2 visualizes one realization of the simulated data for these two settings. From Figure 2 we see that different clusters mix together and the variability between clusters varies. Since we found that MKSSC is better to handle the heterogeneous noise than SSC in both settings, we only focus on the comparison of AMA+CADMM, AMA+NADMM and AManPL for MKSSC. In Table 3, we report the NMI scores of the three algorithms for solving MKSSC. The NMI scores are averaged over 50 independent runs and the numbers in the parentheses are the standard deviation of the NMI scores. From Table 3 we see that AManPL consistently outperforms the other two methods in all tested instances. This is not surprising because AManPL solves the original MKSSC, and the other two methods only solve its approximations. Moreover, we show the heatmaps of $|UU^T|$ for synthetic data 1 in Figure 3 (synthetic data 2 is similar). We see that the block diagonal structure is well recovered by AManPL, which shows much better performance than the other two methods.

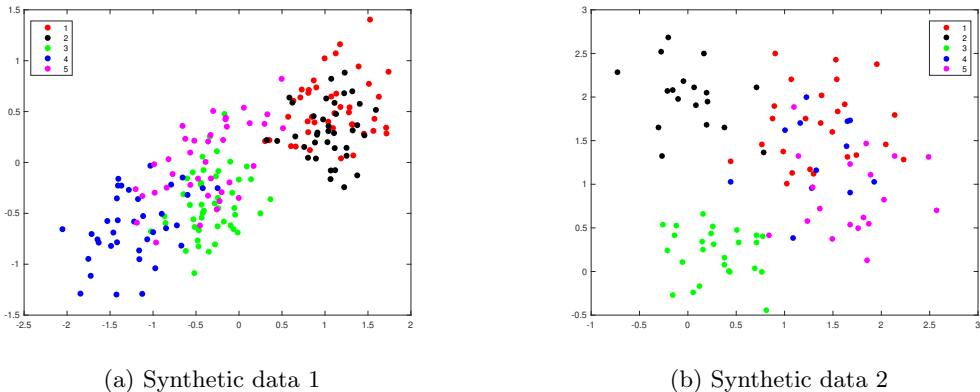


Figure 2: Illustration of one realization of the synthetic data.

5.3 Single-Cell RNA Sequencing Data Analysis

Clustering cells and identifying subgroups are important topics in high-dimensional scRNA-seq data analysis. The multiple kernel learning approach is vital as clustering scRNA-seq data is usually sensitive to the choice of the number of neighbors and scaling parameter. Recently, [Park and Zhao, 2018] showed that AMA+CADMM for MKSSC provides a promising clustering result and outperforms several state-of-art methods such as SC,

Method		AMA+CADMM	AMA+NADMM	AManPL
n	p	Synthetic data 1		
100	250	0.9852 (1.2e-3)	0.9852 (2.4e-3)	0.9933 (1.7e-4)
100	300	0.9789 (2.1e-2)	0.9844 (1.4e-2)	0.9917 (7.9e-4)
100	500	0.9787 (2.0e-3)	0.9834 (1.5e-3)	0.9956 (1.2e-4)
200	250	0.9821 (1.8e-3)	0.9803 (1.9e-3)	0.9955 (6.6e-5)
200	300	0.9830 (1.3e-3)	0.9833 (1.3e-3)	0.9844 (1.6e-4)
200	500	0.9607 (2.8e-3)	0.9606 (2.8e-3)	0.9867 (3.6e-4)
n	p	Synthetic data 2		
100	250	0.6491 (5.8e-3)	0.7163 (3.9e-3)	0.7334 (7.6e-3)
100	300	0.6304 (1.3e-3)	0.7466 (2.4e-3)	0.7881 (1.5e-3)
100	500	0.6253 (2.3e-3)	0.7289 (1.3e-3)	0.7308 (1.1e-3)
200	250	0.7977 (2.1e-3)	0.1371 (2.2e-3)	0.9182 (3.2e-4)
200	300	0.7380 (1.1e-3)	0.1034 (1.2e-3)	0.8773 (1.2e-3)
200	500	0.7130 (9.6e-3)	0.1220 (2.1e-3)	0.8199 (4.5e-3)

Table 3: NMI of three algorithms for solving MKSSC for synthetic data.

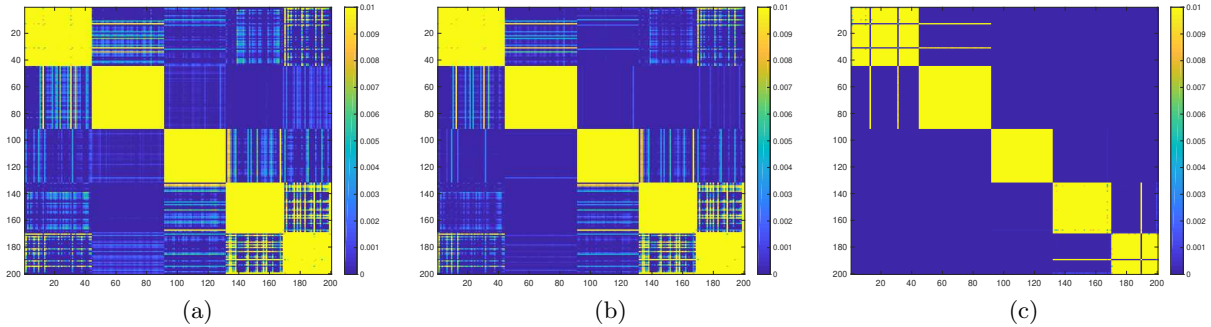


Figure 3: The heatmaps of $|\mathbf{U}\mathbf{U}^\top|$ on synthetic data 1 estimated by (a) AMA+CADMM, (b) AMA+NADMM, and (c) AManPL for MKSSC.

SSC, t-SNE [van der Maaten and Hinton, 2008], and SIMLR [Wang et al., 2017]. In what follows, we focus on the numerical comparison of AMA+CADMM, AMA+NADMM and AManPL to cluster high-dimensional scRNA-seq data on six real datasets. These six real datasets represent several types of important dynamic processes such as cell differentiation, and they include the information about single cell types. We follow the procedure of [Park and Zhao, 2018] to specify multiple kernels for clustering scRNA-seq data and choose the proper tuning parameters λ and ρ from a set of given values so that the best NMI scores were obtained in each case. The six datasets and the NMI scores of the three algorithms: AMA+CADMM, AMA+NADMM and AManPL for solving MKSSC, are summarized in Table 4. From Table 4 we observe that AManPL always achieves the highest NMI scores and consistently outperforms the other two methods on all six real datasets. We also demonstrate the performance of AManPL by showing the two-dimensional embedding estimated by AManPL in Figure 4. From this figure, we see that AManPL yielded clear and meaningful clusters, even for the Ginhous dataset [Schlitzer et al., 2015], which was known to be very challenging. This again demonstrates the great practical potential of our AManPL method for analyzing the scRNA-seq data.

6 Conclusion

Motivated by the recent need on analyzing single cell RNA sequencing data, we considered the sparse spectral clustering and multiple-kernel sparse spectral clustering in this paper. We proposed a manifold proximal

Datasets	AMA+CADMM	AMA+NADMM	AManPL	C
Deng [Deng et al., 2014]	0.7319	0.7389	0.7464	7
Ting [Ting et al., 2014]	0.9283	0.9524	0.9755	5
Treutlein [Treutlein et al., 2014]	0.7674	0.7229	0.8817	5
Buettner [Buettner et al., 2015]	0.7929	0.8744	0.8997	3
Ginhoux [Schlitzer et al., 2015]	0.6206	0.6398	0.6560	3
Pollen [Pollen et al., 2014]	0.9439	0.9372	0.9631	11

Table 4: NMI scores of three algorithms for MKSSC on six real scRNA-seq datasets.

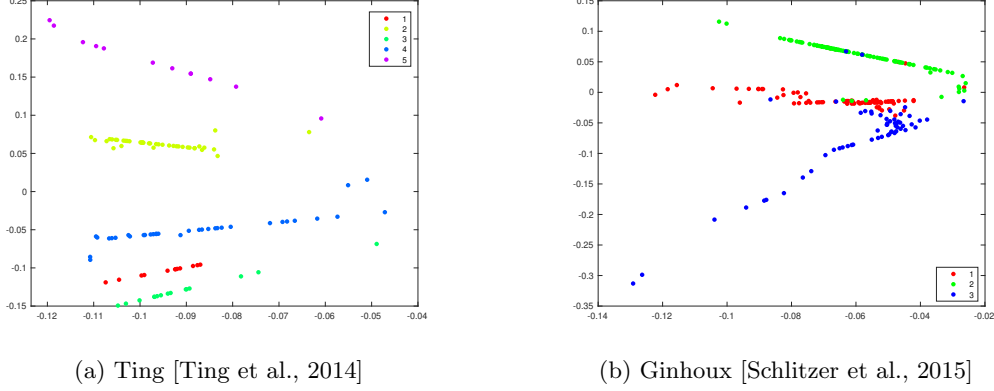


Figure 4: Visualization of the cells in 2-D embedded space using AManPL for MKSSC on two real datasets in high-dimensional scRNA-seq analysis.

linear method for solving SSC, and the alternating manifold proximal linear method for solving MKSSC. Convergence and iteration complexity of the proposed methods are analyzed. Numerical results on synthetic data and the single cell RNA sequencing data demonstrated the great potential of our proposed methods.

A Preliminaries

A.1 Optimality Conditions of Manifold Optimization

Definition A.1. (Generalized Clarke subdifferential [Hosseini and Pouryayevali, 2011]) For a locally Lipschitz function F on \mathcal{M} , the Riemannian generalized directional derivative of F at $X \in \mathcal{M}$ in direction V is defined by

$$F^\circ(X, V) = \limsup_{Y \rightarrow X, t \downarrow 0} \frac{F \circ \phi^{-1}(\phi(Y) + tD\phi(X)[V]) - F \circ \phi^{-1}(\phi(Y))}{t}, \quad (\text{A.1})$$

where (ϕ, U) is a coordinate chart at X and $D\phi(X)$ denotes the Jacobian of $\phi(X)$. The generalized gradient or the Clarke subdifferential of F at $X \in \mathcal{M}$, denoted by $\partial_R F(X)$, is given by

$$\partial_R F(X) = \{\xi \in T_X \mathcal{M} : \langle \xi, V \rangle \leq F^\circ(X, V), \forall V \in T_X \mathcal{M}\}. \quad (\text{A.2})$$

Definition A.2. ([Yang et al., 2014]) A function f is said to be regular at $X \in \mathcal{M}$ along $T_X \mathcal{M}$ if

- for all $V \in T_X \mathcal{M}$, $f'(X; V) = \lim_{t \downarrow 0} \frac{f(X+tV) - f(X)}{t}$ exists, and
- for all $V \in T_X \mathcal{M}$, $f'(X; V) = f^\circ(X; V)$.

For a smooth function f over Riemannian submanifold, if the metric on the manifold is induced by the Euclidean inner product in the ambient space, then we know that $\mathbf{grad} f(X) = \text{Proj}_{T_X \mathcal{M}} \nabla f(X)$. Here

$\mathbf{grad} f$ denotes the Riemannian gradient of f , and $\text{Proj}_{\mathbb{T}_X \mathcal{M}}$ denotes the projection onto $\mathbb{T}_X \mathcal{M}$. According to Lemma 5.1 in [Yang et al., 2014], for a regular function F , we have $\partial_R F(X) = \text{Proj}_{\mathbb{T}_X \mathcal{M}}(\partial F(X))$. Moreover, the function $h(c(U))$ in problem (2.5) is weakly convex and thus is regular according to Lemma 5.1 in [Yang et al., 2014]. By Theorem 4.1 in [Yang et al., 2014], the first-order optimality condition of problem (2.5) is given by

$$0 \in \text{Proj}_{\mathbb{T}_U \mathcal{M}} \nabla c(U)^\top \partial h(c(U)) + \mathbf{grad} f(U). \quad (\text{A.3})$$

Definition A.3. A point $U \in \mathcal{M}$ is called a stationary point of problem (2.5) if it satisfies the first-order optimality condition (A.3).

Definition A.4. A retraction on a differentiable manifold \mathcal{M} is a smooth mapping Retr from the tangent bundle TM onto \mathcal{M} satisfying the following two conditions (here Retr_X denotes the restriction of Retr onto $\mathbb{T}_X \mathcal{M}$)

- $\text{Retr}_X(0) = X, \forall X \in \mathcal{M}$, where 0 denotes the zero element of $\mathbb{T}_X \mathcal{M}$.
- For $X \in \mathcal{M}$, it holds that

$$\lim_{\mathbb{T}_X \mathcal{M} \ni \xi \rightarrow 0} \frac{\|\text{Retr}_X(\xi) - (X + \xi)\|_F}{\|\xi\|_F} = 0.$$

The retraction onto the Euclidean space is simply the identity mapping: $\text{Retr}_X(\xi) = X + \xi$. Common retractions include the polar decomposition:

$$\text{Retr}_X^{\text{polar}}(\xi) = (X + \xi)(I_r + \xi^\top \xi)^{-1/2},$$

the QR decomposition:

$$\text{Retr}_X^{\text{QR}} = \mathbf{qf}(X + \xi),$$

where $\mathbf{qf}(A)$ is the Q factor of the QR factorization of A , and the Cayley transformation:

$$\text{Retr}_X^{\text{cayley}}(\xi) = \left(I_n - \frac{1}{2}W(\xi)\right)^{-1} \left(I_n + \frac{1}{2}W(\xi)\right)X,$$

where $W(\xi) = (I_n - \frac{1}{2}XX^\top)\xi X^\top - X\xi^\top(I_n - \frac{1}{2}XX^\top)$

The following lemma is useful in our analysis.

Lemma A.5. [Boumal et al., 2018, Chen et al., 2020c] For all $X \in \mathcal{M}$ and $\xi \in \mathbb{T}_X \mathcal{M}$ there exist constants $M_1 > 0$ and $M_2 > 0$ such that the following two inequalities hold:

$$\|\text{Retr}_X(\xi) - X\|_F \leq M_1 \|\xi\|_F, \forall X \in \mathcal{M}, \xi \in \mathbb{T}_X \mathcal{M} \quad (\text{A.4a})$$

$$\|\text{Retr}_X(\xi) - (X + \xi)\|_F \leq M_2 \|\xi\|_F^2, \forall X \in \mathcal{M}, \xi \in \mathbb{T}_X \mathcal{M}. \quad (\text{A.4b})$$

Lemma A.6. The function h in (2.5) is L_h -Lipschitz continuous and the Jacobian $\nabla c(x)$ is L_c -Lipschitz continuous, where $L_h = n\lambda$ and $L_c = 2$.

Proof. Since $h(Z) = \lambda \|Z\|_1$ where $Z \in \mathbb{R}^{n \times n}$, we immediately get that h is $n\lambda$ -Lipschitz continuous. Secondly, observing that $\nabla c(U)V = UV^\top + VU^\top$ for any $V \in \mathbb{R}^{n \times k}$, we have

$$\|\nabla c(U_1) - \nabla c(U_2)\|_{\text{op}} = \max_{\|V\|_F=1} \|(U_1 - U_2)V^\top + V(U_1 - U_2)^\top\|_F \leq 2 \|U_1 - U_2\|_F.$$

Hence, $\nabla c(x)$ is 2-Lipschitz continuous. \square

B Proofs

B.1 Convergence Analysis of ManPL (Algorithm 1)

Before we present the convergence result of ManPL, we need the following useful lemma for proximal linear algorithm.

Lemma B.1. *[Drusvyatskiy and Paquette, 2018] For functions h and c satisfying the properties in Lemma A.6, we have the following result:*

$$-\frac{L_c L_h}{2} \|U_2 - U_1\|_F^2 \leq h(c(U_2)) - h(c(U_1) + \nabla c(U_1)(U_2 - U_1)) \leq \frac{L_c L_h}{2} \|U_2 - U_1\|_F^2. \quad (\text{B.1})$$

Proof. It follows from Lemma A.6 that

$$\begin{aligned} & \left| h(c(U_2)) - h(c(U_1) + \nabla c(U_1)(U_2 - U_1)) \right| \\ & \leq L_h \|c(U_2) - (c(U_1) + \nabla c(U_1)(U_2 - U_1))\|_F \\ & = L_h \left\| \int_0^1 (\nabla c(U_1 + t(U_2 - U_1)) - \nabla c(U_1))(U_2 - U_1) dt \right\|_F \\ & \leq L_h \int_0^1 \|(\nabla c(U_1 + t(U_2 - U_1)) - \nabla c(U_1))\|_{\text{op}} \|U_2 - U_1\|_F dt \\ & \leq L_h L_c \left(\int_0^1 t dt \right) \|U_2 - U_1\|_F^2 = \frac{L_c L_h}{2} \|U_2 - U_1\|_F^2. \end{aligned}$$

This completes the proof. \square

Now, we start to analyze the convergence and iteration complexity of our ManPL algorithm. The following lemma states that the minimizer of (2.10a) is a descent direction on the tangent space.

Lemma B.2. *Let V^k be the minimizer of (2.10a), the following holds for any $\alpha \in [0, 1]$, with any $t > 0$:*

$$\langle \nabla f(U^k), \alpha V^k \rangle + \frac{1}{2t} \|\alpha V^k\|_F^2 + h(c(U^k) + \alpha \nabla c(U^k)V^k) - h(c(U^k)) \leq \frac{\alpha^2 - 2\alpha}{2t} \|V^k\|_F^2. \quad (\text{B.2})$$

Proof. Since V^k is the minimizer of (2.10a), for any $\alpha \in [0, 1]$, we have:

$$\begin{aligned} & \langle \nabla f(U^k), \alpha V^k \rangle + \frac{1}{2t} \|\alpha V^k\|_F^2 + h(c(U^k) + \alpha \nabla c(U^k)V^k) \\ & \geq \langle \nabla f(U^k), V^k \rangle + \frac{1}{2t} \|V^k\|_F^2 + h(c(U^k) + \nabla c(U^k)V^k), \end{aligned}$$

which implies that

$$(1 - \alpha) \langle \nabla f(U^k), V^k \rangle + \frac{1 - \alpha^2}{2t} \|V^k\|_F^2 + h(c(U^k) + \nabla c(U^k)V^k) - h(c(U^k) + \alpha \nabla c(U^k)V^k) \leq 0.$$

Using the convexity of h , we have

$$\langle \nabla f(U^k), V^k \rangle + \frac{1 + \alpha}{2t} \|V^k\|_F^2 + h(c(U^k) + \nabla c(U^k)V^k) - h(c(U^k)) \leq 0.$$

Letting $\alpha \rightarrow 1$, we get

$$\langle \nabla f(U^k), V^k \rangle + h(c(U^k) + \nabla c(U^k)V^k) - h(c(U^k)) \leq -\frac{1}{t} \|V^k\|_F^2.$$

Finally, from the convexity of h we get

$$\begin{aligned}
& \langle \nabla f(U^k), \alpha V^k \rangle + \frac{1}{2t} \|\alpha V^k\|_F^2 + h\left(c(U^k) + \alpha \nabla c(U^k) V^k\right) - h\left(c(U^k)\right) \\
& \leq \alpha \left(\langle \nabla f(U^k), V^k \rangle + h\left(c(U^k) + \nabla c(U^k) V^k\right) - h\left(c(U^k)\right) \right) + \frac{\alpha^2}{2t} \|V^k\|_F^2 \\
& \leq \frac{\alpha^2 - 2\alpha}{2t} \|V^k\|_F^2,
\end{aligned}$$

which completes the proof. \square

The following lemma suggests that the new point $U^{k+1} = \text{Retr}_{U^k}(\alpha V^k)$ has a lower function value.

Lemma B.3. *Given any $t > 0$, for V^k and U^{k+1} computed in (2.10), there exists a constant $\bar{\alpha} > 0$ such that*

$$F(U^{k+1}) - F(U^k) \leq -\frac{\alpha}{2t} \|V^k\|_F^2, \quad \forall 0 \leq \alpha \leq \min\{1, \bar{\alpha}\}. \quad (\text{B.3})$$

Proof. We will prove by induction. Denote $U_+^k = U^k + \alpha V^k$. For $k = 0$, by using the convexity of h and Lipschitz continuity of c , we can show that:

$$\begin{aligned}
& h\left(c(U^{k+1})\right) - h\left(c(U^k) + \alpha \nabla c(U^k) V^k\right) \\
& = h\left(c(U^{k+1})\right) - h\left(c(U^k) + \nabla c(U^k)(U^{k+1} - U^k)\right) \\
& \quad + h\left(c(U^k) + \nabla c(U^k)(U^{k+1} - U^k)\right) - h\left(c(U^k) + \nabla c(U^k)(U_+^k - U^k)\right) \\
& \stackrel{(\text{B.1})}{\leq} \frac{L_h L_c}{2} \|U^{k+1} - U^k\|_F^2 + L_h \|\nabla c(U^k)(U^{k+1} - U_+^k)\|_F \\
& \stackrel{(\text{A.4a})}{\leq} \frac{\alpha^2 L_h L_c}{2} M_1^2 \|V^k\|_F^2 + L_h \|\nabla c(U^k)(U^{k+1} - U_+^k)\|_F \\
& \stackrel{(\text{A.4b})}{\leq} \frac{\alpha^2 L_c L_h}{2} M_1^2 \|V^k\|_F^2 + L_h L_c M_2 \alpha^2 \|V^k\|_F^2 = \left(\frac{1}{2} M_1^2 + M_2\right) L_c L_h \alpha^2 \|V^k\|_F^2.
\end{aligned} \quad (\text{B.4})$$

Since $\nabla f(X)$ is L_f Lipschitz continuous, we obtain

$$\begin{aligned}
f(U^{k+1}) - f(U^k) & \leq \langle \nabla f(U^k), U^{k+1} - U^k \rangle + \frac{L_f}{2} \|U^{k+1} - U^k\|_F^2 \\
& = \langle \nabla f(U^k), U^{k+1} - U_+^k + U_+^k - U^k \rangle + \frac{L_f}{2} \|U^{k+1} - U^k\|_F^2 \\
& \stackrel{(\text{A.4b})}{\leq} M_2 \|\nabla f(U^k)\|_F \|\alpha V^k\|_F^2 + \alpha \langle \nabla f(U^k), V^k \rangle + \frac{M_1^2 L_f}{2} \|\alpha V^k\|_F^2 \\
& \leq c_0 \alpha^2 \|V^k\|_F^2 + \alpha \langle \nabla f(U^k), V^k \rangle,
\end{aligned} \quad (\text{B.5})$$

where $c_0 = M_2 G + \frac{M_1^2 L_f}{2}$ and $G = \max_{U \in \mathcal{M}} \|\nabla f(U)\|_F$.

Now we need to show that the objective value decreases. It follows that

$$\begin{aligned}
& F(U^{k+1}) - F(U^k) \\
& \stackrel{(\text{B.5})}{\leq} \alpha \langle \nabla f(U^k), V^k \rangle + c_0 \alpha^2 \|V^k\|_F^2 + h\left(c(U^{k+1})\right) - h\left(c(U^k) + \alpha \nabla c(U^k) V^k\right) \\
& \quad + h\left(c(U^k) + \alpha \nabla c(U^k) V^k\right) - h\left(c(U^k)\right) \\
& \stackrel{(\text{B.4})}{\leq} \alpha \langle \nabla f(U^k), V^k \rangle + c_0 \alpha^2 \|V^k\|_F^2 + \left(\frac{1}{2} M_1^2 + M_2\right) L_c L_h \alpha^2 \|V^k\|_F^2 \\
& \quad + \alpha \left(h\left(c(U^k) + \nabla c(U^k) V^k\right) - h\left(c(U^k)\right) \right) \\
& \stackrel{(\text{B.2})}{\leq} \left[\left(c_0 + \left(\frac{1}{2} M_1^2 + M_2\right) L_c L_h \right) \alpha^2 - \frac{\alpha}{t} \right] \|V^k\|_F^2.
\end{aligned}$$

Letting $\bar{\alpha} = \frac{1}{2\left(c_0 + \left(\frac{1}{2}M_1^2 + M_2\right)L_c L_h\right)t}$, we get that for any $0 \leq \alpha \leq \min\{1, \bar{\alpha}\}$,

$$F(\text{Retr}_{U^k}(\alpha V^k)) - F(U^k) \leq -\frac{\alpha}{2t} \|V^k\|_F^2. \quad (\text{B.6})$$

Thus, the result (B.3) holds for $k = 0$. Using induction and the same procedure as above, we can show that (B.3) holds for all $k \geq 1$. The proof is completed. \square

Following [Chen et al., 2020c], we now define the ϵ -stationary point of (2.5).

Definition B.4. *Given any $t > 0$, U^k is called an ϵ -stationary point of (2.5), if V^k computed in (2.10a) satisfies $\|V^k/t\|_F \leq \epsilon$.*

The following lemma indicates that Definition B.4 is a reasonable definition for the ϵ -stationary point.

Lemma B.5. *If $V^k = 0$, then U^k is a stationary point of problem (2.5).*

Proof. The optimality conditions of (2.10a) are given by:

$$0 \in \text{Proj}_{\mathbb{T}_{U^k} \mathcal{M}} \nabla c(U^k)^\top \partial h(c(U^k) + \nabla c(U^k) V^k) + \frac{1}{t} V^k + \mathbf{grad} f(U^k).$$

If $V^k = 0$, it follows that

$$0 \in \text{Proj}_{\mathbb{T}_{U^k} \mathcal{M}} \nabla c(U^k)^\top \partial h(c(U^k)) + \mathbf{grad} f(U^k),$$

which is the first-order optimality condition of (2.5). \square

Proof. Proof of Theorem 2.1. By using Lemma B.3, the claimed result directly follows from the proof of [Chen et al., 2020c, Theorem 5.5]. We omit the details for brevity. \square

B.2 Convergence Analysis of AManPL (Algorithm 2)

Lemma B.6. *(Sufficient decrease in the w subproblem) From (4.10), we have*

$$\bar{F}(U^{k+1}, w^{k+1}) - \bar{F}(U^{k+1}, w^k) \leq -\frac{\rho}{2} \|w^{k+1} - w^k\|_1^2. \quad (\text{B.7})$$

Proof. This follows from the ρ -strong convexity of the objective function of (4.10) over the probability simplex. \square

Lemma B.7. *Let (V^k, U^{k+1}) be computed from (4.9). There exists $\hat{\alpha} > 0$ such that for any $0 \leq \alpha \leq \min\{1, \hat{\alpha}\}$, it holds that*

$$\bar{F}(U^{k+1}, w^{k+1}) - \bar{F}(U^k, w^k) \leq -\left(\frac{\alpha}{2t} \|V^k\|_F^2 + \frac{\rho}{2} \|w^{k+1} - w^k\|_1^2\right). \quad (\text{B.8})$$

Proof. Firstly we show that the subproblem (4.8) is of the same form as (2.5). That is, we show that $f(U, w^k) = \sum_{\ell=1}^T w_\ell^k \langle L^{(\ell)}, UU^\top \rangle$ is smooth with respect U and its gradient with respect U is Lipschitz continuous. Once we prove this, then (B.8) follows by combining Lemma B.6 and the proof of Lemma B.3. Note that for any $w \in \mathbb{R}^T$, we have:

$$\begin{aligned} \|\nabla_U f(U_1, w^k) - \nabla_U f(U_2, w^k)\|_F &= \left\| 2 \sum_{\ell=1}^T w_\ell^k L^{(\ell)} (U_1 - U_2) \right\|_F \\ &\leq 2 \sum_{\ell=1}^T w_\ell^k \|L^{(\ell)}\|_2 \|U_1 - U_2\|_F \leq 2 \|U_1 - U_2\|_F, \end{aligned}$$

where we use the fact that $\|L^{(\ell)}\|_2 = 1$ for every ℓ (followed by the property of the affinity matrix). This implies that for fixed w^k , $f(\cdot, w^k)$ is smooth and its gradient is Lipschitz continuous with Lipschitz constant $L_f = 2$. Now we can use Lemma B.3 and let $\hat{\alpha} = \frac{1}{2\left(c_1 + \left(\frac{1}{2}M_1^2 + M_2\right)L_c L_b\right)t}$, where $c_1 = M_2 G_1 + \frac{L_f M_1^2}{2}$ and

$$G_1 = \max_{U \in \mathcal{M}, \sum_{\ell} w_{\ell} = 1, 0 \leq w \leq 1} \|\nabla_U f(U, w)\|_F,$$

M_1 and M_2 are constants defined in Lemma A.5. From Lemma B.3 we get that for any $0 \leq \alpha \leq \min\{1, \hat{\alpha}\}$,

$$\bar{F}(\text{Retr}_{U^k}(\alpha V^k), w^k) - \bar{F}(U^k, w^k) \leq -\frac{\alpha}{2t} \|V^k\|_F^2,$$

which together with (B.7) yields the desired result (B.8). \square

Since for the w -subproblem (4.3), one always has (B.10). This motivates us to define the following ϵ -stationary point for problem (1.3).

Definition B.8. We call a point (U^k, w^k) an ϵ -stationary point of problem (1.3) if

$$\|V^k/t\|_F \leq \epsilon,$$

where V_k is the optimal solution to (4.9a).

The following lemma indicates that Definition B.8 is a reasonable definition for the ϵ -stationary point of problem (1.3).

Lemma B.9. For given (U^k, w^k) generated from the k -th iteration of AManPL, if $V^k = 0$, where V^k is the optimal solution to (4.9a), then (U^k, w^k) is a stationary point of problem (1.3).

Proof. Since $V^k = 0$, the optimality condition of (4.9a) implies that

$$0 \in \text{Proj}_{\mathbb{T}_{U^k, \mathcal{M}}} \left(2 \sum_{\ell=1}^T w_{\ell}^k L^{(\ell)} U^k + \nabla c(U^k)^{\top} \partial h(c(U^k)) \right). \quad (\text{B.9})$$

Moreover, the optimality condition to (4.10) implies that

$$0 \in c_{\ell} + \rho(\log(w_{\ell}^k) + 1) + \partial_{\ell} \mathbb{I}(w^k), \quad (\text{B.10})$$

where $c_{\ell} = \langle U^k U^{k\top}, L^{(\ell)} \rangle$, $\ell = 1, \dots, T$, and $\mathbb{I}(w)$ denotes the indicator function of the probability simplex constraint $\mathcal{S} := \{w \mid \sum_{\ell=1}^T w_{\ell} = 1, w_{\ell} \geq 0, \ell = 1, \dots, T\}$, i.e., $\mathbb{I}(w) = 0$ if $w \in \mathcal{S}$, and $\mathbb{I}(w) = +\infty$ otherwise. Moreover, since

$$\partial_R(h(c(U)), \mathbb{I}(w)) = \text{Proj}_{\mathbb{T}_U, \mathcal{M}}(\nabla c(U)^{\top} \partial h(c(U))) \times \partial \mathbb{I}(w),$$

the first-order optimal conditions of (1.3) are given by

$$\begin{aligned} 0 &\in \text{Proj}_{\mathbb{T}_U, \mathcal{M}} \left(2 \sum_{\ell=1}^T w_{\ell} L^{(\ell)} U + \nabla c(U)^{\top} \partial h(c(U)) \right), \\ 0 &\in \langle L^{(\ell)}, U U^{\top} \rangle + \rho(\log(w_{\ell}) + 1) + \partial_{\ell} \mathbb{I}(w), \quad \forall \ell. \end{aligned} \quad (\text{B.11})$$

From (B.9) and (B.10) we know that (U^k, w^k) satisfies the optimality condition (B.11). Therefore (U^k, w^k) is a stationary point of (1.3). \square

Proof. Proof of Theorem 4.1. Since the feasible region of (1.3) is bounded, there exists a convergent subsequence of $\{(U^k, w^k)\}$. We denote a limit point of $\{(U^k, w^k)\}$ as $\{(U^*, w^*)\}$. Lemma B.7 indicates that $\{\bar{F}(U^k, w^k)\}$ monotonically decreases, and therefore we have

$$\lim_{k \rightarrow \infty} \left\| V^k / t \right\|_F^2 + \|\rho(w^{k+1} - w^k)\|_1^2 = 0.$$

Note that the function $h(c(U) + \nabla c(U)V)$ is convex with respect to V , therefore, taking limit for (B.9) and (B.10) gives

$$0 \in \text{Proj}_{T_{U^*} \mathcal{M}} \left(2 \sum_{\ell=1}^T w_\ell^* L^{(\ell)} U^* + \nabla c(U^*)^\top \partial h(c(U^*)) \right),$$

and

$$0 \in c_\ell^* + \rho(\log(w_\ell^*) + 1) + \partial_\ell \mathbb{I}(w^*), \quad \ell = 1, \dots, T.$$

This suggests that (U^*, w^*) is a stationary point.

Moreover, given $N \geq \frac{2t\hat{\alpha}\gamma(\bar{F}(U^0, w^0) - \bar{F}^*)}{\epsilon^2}$ it follows from Lemma B.7 that

$$\min_{k=1, \dots, N} \left\| V^k / t \right\|_F^2 \leq \epsilon^2.$$

The proof is completed. □

References

- P.-A. Absil, R. Mahony, and R. Sepulchre. *Optimization Algorithms on Matrix Manifolds*. Princeton University Press, Princeton, NJ, 2008. ISBN 978-0-691-13298-3.
- T. Bendory, Y. C. Eldar, and N. Boumal. Non-convex phase retrieval from STFT measurements. *IEEE Transactions on Information Theory*, 2018.
- N. Boumal. Nonconvex phase synchronization. *SIAM Journal on Optimization*, 26(4):2355–2377, 2016.
- Nicolas Boumal and Pierre-Antoine Absil. RTRMC: A Riemannian trust-region method for low-rank matrix completion. In *Advances in neural information processing systems*, pages 406–414, 2011.
- Nicolas Boumal, Pierre-Antoine Absil, and Coralia Cartis. Global rates of convergence for nonconvex optimization on manifolds. *IMA Journal of Numerical Analysis*, 39(1):1–33, 2018.
- Florian Buettner, Kedar N Natarajan, F Paolo Casale, Valentina Proserpio, Antonio Scialdone, Fabian J Theis, Sarah A Teichmann, John C Marioni, and Oliver Stegle. Computational analysis of cell-to-cell heterogeneity in single-cell RNA-sequencing data reveals hidden subpopulations of cells. *Nature Biotechnology*, 33(2):155, 2015.
- V. Charisopoulos, Y. Chen, D. Davis, M. Diaz, L. Ding, and D. Drusvyatskiy. Low-rank matrix recovery with composite optimization: Good conditioning and rapid convergence. <https://arxiv.org/pdf/1904.10020.pdf>, 2019a.
- Vasileios Charisopoulos, Damek Davis, Mateo Díaz, and Dmitriy Drusvyatskiy. Composite optimization for robust blind deconvolution. *arXiv preprint arXiv:1901.01624*, 2019b.
- S. Chen, Z. Deng, S. Ma, and A. M.-C. So. Manifold proximal point algorithms for dual principal component pursuit and orthogonal dictionary learning. In *Proceedings of the 2019 Asilomar Conference on Signals, Systems, and Computers.*, 2019.

- S. Chen, Z. Deng, S. Ma, and A. M.-C. So. Manifold proximal point algorithms for dual principal component pursuit and orthogonal dictionary learning. *arXiv:2005.02356*, 2020a.
- S. Chen, S. Ma, L. Xue, and H. Zou. An alternating manifold proximal gradient method for sparse principal component analysis and sparse canonical correlation analysis. *INFORMS Journal on Optimization*, 2(3):192–208, 2020b.
- Shixiang Chen, Shiqian Ma, Anthony Man-Cho So, and Tong Zhang. Proximal gradient method for nonsmooth optimization over the Stiefel manifold. *SIAM J. Optimization*, 30(1):210–239, 2020c.
- A. Cherian and S. Sra. Riemannian dictionary learning and sparse coding for positive definite matrices. *IEEE Transactions on Neural Networks and Learning Systems*, 28(12):2859–2871, 2017.
- Fan RK Chung and Fan Chung Graham. *Spectral Graph Theory*. American Mathematical Society, 1997.
- Qiaolin Deng, Daniel Ramsköld, Björn Reinius, and Rickard Sandberg. Single-cell RNA-seq reveals dynamic, random monoallelic gene expression in mammalian cells. *Science*, 343(6167):193–196, 2014.
- Dmitriy Drusvyatskiy and Courtney Paquette. Efficiency of minimizing compositions of convex functions and smooth maps. *Mathematical Programming*, pages 1–56, 2018.
- Dheeru Dua and Casey Graff. UCI machine learning repository, 2017. URL <http://archive.ics.uci.edu/ml>.
- J. C. Duchi and F. Ruan. Stochastic methods for composite and weakly convex optimization problems. *SIAM Journal on Optimization*, 28(4):3229–3259, 2018.
- John C Duchi and Feng Ruan. Solving (most) of a set of quadratic equalities: Composite optimization for robust phase retrieval. *arXiv preprint arXiv:1705.02356*, 2017.
- O. P. Ferreira and P. R. Oliveira. Subgradient algorithm on Riemannian manifolds. *Journal of Optimization Theory and Applications*, 97(1):93–104, 1998.
- Jerome Friedman, Trevor Hastie, and Robert Tibshirani. *The Elements of Statistical Learning*. Springer Series in Statistics New York, 2001.
- P. Grohs and S. Hosseini. ϵ -subgradient algorithms for locally lipschitz functions on Riemannian manifolds. *Advances in Computational Mathematics*, 42(2):333–360, 2016.
- S Hosseini and MR Pouryayevali. Generalized gradients and characterization of epi-lipschitz sets in Riemannian manifolds. *Nonlinear Analysis: Theory, Methods & Applications*, 74(12):3884–3895, 2011.
- S. Hosseini and A. Uschmajew. A Riemannian gradient sampling algorithm for nonsmooth optimization on manifolds. *SIAM Journal on Optimization*, 27(1):173–189, 2017.
- Wen Huang and Ke Wei. Extending FISTA to Riemannian optimization for sparse PCA. Technical report, 2019a.
- Wen Huang and Ke Wei. Riemannian proximal gradient methods. Technical report, 2019b.
- Vladimir Yu Kiselev, Tallulah S Andrews, and Martin Hemberg. Challenges in unsupervised clustering of single-cell RNA-seq data. *Nature Reviews Genetics*, page 1, 2019.
- Adrian S Lewis and Stephen J Wright. A proximal method for composite minimization. *Mathematical Programming*, 158(1-2):501–546, 2016.
- X. Li, D. Sun, and K.-C. Toh. A highly efficient semismooth Newton augmented Lagrangian method for solving Lasso problems. *SIAM J. Optimization*, 28(1):433–458, 2018.

- X. Li, S. Chen, Z. Deng, Q. Qu, Z. Zhu, and A. M.-C. So. Weakly convex optimization over Stiefel manifold using Riemannian subgradient-type methods. <https://arxiv.org/abs/1911.05047>, 2019.
- H. Liu, M.-C. Yue, and A. M.-C. So. On the estimation performance and convergence rate of the generalized power method for phase synchronization. *SIAM Journal on Optimization*, 27(4):2426–2446, 2017.
- Canyi Lu, Shuicheng Yan, and Zhouchen Lin. Convex sparse spectral clustering: Single-view to multi-view. *IEEE Transactions on Image Processing*, 25(6):2833–2843, 2016.
- Canyi Lu, Jiashi Feng, Zhouchen Lin, and Shuicheng Yan. Nonconvex sparse spectral clustering by alternating direction method of multipliers and its convergence analysis. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.
- Robert Mifflin. Semismooth and semiconvex functions in constrained optimization. *SIAM Journal on Control and Optimization*, 15(6):959–972, 1977.
- Andrew Y Ng, Michael I Jordan, and Yair Weiss. On spectral clustering: Analysis and an algorithm. In *Advances in Neural Information Processing Systems*, pages 849–856, 2002.
- Seyoung Park and Hongyu Zhao. Spectral clustering based on learning similarity matrix. *Bioinformatics*, 34(12):2069–2076, 2018.
- Alex A Pollen, Tomasz J Nowakowski, Joe Shuga, Xiaohui Wang, Anne A Leyrat, Jan H Lui, Nianzhen Li, Lukasz Szpankowski, Brian Fowler, Peilin Chen, et al. Low-coverage single-cell mRNA sequencing reveals cellular heterogeneity and activated signaling pathways in developing cerebral cortex. *Nature Biotechnology*, 32(10):1053, 2014.
- Liqun Qi and Jie Sun. A nonsmooth version of Newton’s method. *Mathematical Programming*, 58(1-3):353–367, 1993.
- Andreas Schlitzer, V Sivakamasundari, Jinmiao Chen, Hermi Rizal Bin Sumatoh, Jaring Schreuder, Josephine Lum, Benoit Malleret, Sanqian Zhang, Anis Larbi, Francesca Zolezzi, et al. Identification of cDC1-and cDC2-committed DC progenitors reveals early lineage priming at the common DC progenitor stage in the bone marrow. *Nature Immunology*, 16(7):718, 2015.
- Jianbo Shi and Jitendra Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.
- J. Sun, Q. Qu, and J. Wright. Complete dictionary recovery over the sphere I: Overview and the geometric picture. *IEEE Transactions on Information Theory*, 63(2):853–884, 2017.
- J. Sun, Q. Qu, and J. Wright. A geometrical analysis of phase retrieval. *Foundations of Computational Mathematics*, 18(5):1131–1198, 2018.
- David T Ting, Ben S Wittner, Matteo Ligorio, Nicole Vincent Jordan, Ajay M Shah, David T Miyamoto, Nicola Aceto, Francesca Bersani, Brian W Brannigan, Kristina Xega, et al. Single-cell RNA sequencing identifies extracellular matrix gene expression by pancreatic circulating tumor cells. *Cell Reports*, 8(6):1905–1918, 2014.
- Barbara Treutlein, Doug G Brownfield, Angela R Wu, Norma F Neff, Gary L Mantalas, F Hernan Espinoza, Tushar J Desai, Mark A Krasnow, and Stephen R Quake. Reconstructing lineage hierarchies of the distal lung epithelium using single-cell RNA-seq. *Nature*, 509(7500):371, 2014.
- Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9(Nov):2579–2605, 2008.
- B. Wang, S. Ma, and L. Xue. Riemannian stochastic proximal gradient methods for nonsmooth optimization over the Stiefel manifold. <https://arxiv.org/pdf/2005.01209.pdf>, 2020.

- Bo Wang, Daniele Ramazzotti, Luca De Sano, Junjie Zhu, Emma Pierson, and Serafim Batzoglou. SIMLR: a tool for large-scale single-cell analysis by multi-kernel learning. *bioRxiv*, page 118901, 2017.
- Chengjing Wang, Defeng Sun, and Kim-Chuan Toh. Solving log-determinant optimization problems by a Newton-CG primal proximal point algorithm. *SIAM Journal on Optimization*, 20(6):2994–3013, 2010.
- Xiantao Xiao, Yongfeng Li, Zaiwen Wen, and Liwei Zhang. A regularized semi-smooth Newton method with projection steps for composite convex programs. *Journal of Scientific Computing*, pages 1–26, 2018.
- Junfeng Yang, Defeng Sun, and Kim-Chuan Toh. A proximal point algorithm for log-determinant optimization with group Lasso regularization. *SIAM Journal on Optimization*, 23(2):857–893, 2013.
- Wei Hong Yang, Lei-Hong Zhang, and Ruyi Song. Optimality conditions for the nonlinear programming problems on Riemannian manifolds. *Pacific Journal of Optimization*, 10(2):415–434, 2014.
- Xin-Yuan Zhao, Defeng Sun, and Kim-Chuan Toh. A Newton-CG augmented Lagrangian method for semidefinite programming. *SIAM Journal on Optimization*, 20(4):1737–1765, 2010.