Sum-of-norms regularized Nonnegative Matrix Factorization [*]

[1]Andersen Ang Waqas Bin Hamed [2]Hans De Sterck

[1]School of Electronics and Computer Science, University of Southampton, UK
[2]Department of Applied Mathematics, University of Waterloo, Canada

September 24, 2025

Abstract

When applying nonnegative matrix factorization (NMF), the rank parameter is generally unknown. This rank, called the nonnegative rank, is usually estimated heuristically since computing its exact value is NP-hard. In this work, we propose an approximation method to estimate the rank on-the-fly while solving NMF. We use the sum-of-norm (SON), a group-lasso structure that encourages pairwise similarity, to reduce the rank of a factor matrix when the initial rank is overestimated. On various datasets, SON-NMF can reveal the correct nonnegative rank of the data without prior knowledge or parameter tuning.

SON-NMF is a nonconvex, nonsmooth, non-separable, and non-proximable problem, making it nontrivial to solve. First, since rank estimation in NMF is NP-hard, the proposed approach does not benefit from lower computational complexity. Using a graph-theoretic argument, we prove that the complexity of SON-NMF is essentially irreducible. Second, the per-iteration cost of algorithms for SON-NMF can be high. This motivates us to propose a first-order BCD algorithm that approximately solves SON-NMF with low per-iteration cost via the proximal average operator.

SON-NMF exhibits favorable features for applications. Besides the ability to automatically estimate the rank from data, SON-NMF can handle rank-deficient data matrices and detect weak components with small energy. Furthermore, in hyperspectral imaging, SON-NMF naturally addresses the issue of spectral variability.

Keywords: nonnegative matrix factorization, rank, regularization, sum-of-norms, nonsmooth nonconvex optimization, algorithm, proximal gradient, proximal average, complete graph

[*] Andersen Ang (andersen.ang@soton.ac.uk) is the corresponding author. Part of the work of this paper was done when Andersen Ang was a post-doctoral fellow and when Waqas Bin Hamed was a master student, both at the University of Waterloo. Funding: Andersen Ang acknowledge the supported in part by a joint postdoctoral fellowship by the Fields Institute for Research in Mathematical Sciences and the University of Waterloo, and in part by Discovery Grants from the Natural Sciences and Engineering Research Council (NSERC) of Canada.

1 Introduction

Nonnegative Matrix Factorization (NMF) Denote $\texttt{NMF}(\boldsymbol{M}, r)$ as the following problem: given a matrix $\boldsymbol{M} \in \mathbb{R}_+^{m \times n}$, find two factor matrices $\boldsymbol{W} \in \mathbb{R}_+^{m \times r}$ and $\boldsymbol{H} \in \mathbb{R}_+^{r \times n}$ such that $\boldsymbol{M} = \boldsymbol{W}\boldsymbol{H}$. NMF Paatero and Tapper (1994); Gillis (2020) describes a cone: $\boldsymbol{M}$ is a point cloud (of n points) in $\mathbb{R}_+^m$, contained in a polyhedral cone generated by the r columns of $\boldsymbol{W}$, with nonnegative weights encoded in $\boldsymbol{H}$. Here, H_{ij} represents the contribution of column $\boldsymbol{w}_i$ to the representation of data column $\boldsymbol{m}_j$; see, e.g., (Leplat et al., 2019, Fig.1).

Nonnegative rank Let $r = \text{rank}_+(\boldsymbol{M})$ denote the nonnegative rank of a matrix, where r is the minimal number of nonnegative rank-1 components required to represent $\boldsymbol{M}$ (Berman and Plemmons, 1994, Sect.4), (Gillis, 2020, Sect.3), i.e.,

$$\boldsymbol{M} = \boldsymbol{W}\boldsymbol{H} = \begin{bmatrix}\boldsymbol{w}_1 \dots \boldsymbol{w}_r\end{bmatrix} \begin{bmatrix}\boldsymbol{h}^1 \\ \vdots \\ \boldsymbol{h}^r\end{bmatrix} = \boldsymbol{w}_1\boldsymbol{h}^1 + \cdots + \boldsymbol{w}_r\boldsymbol{h}^r = \sum_{\ell=1}^{r} \boldsymbol{w}_\ell \boldsymbol{h}^\ell, \quad (\texttt{NMF}(\boldsymbol{M}, r))$$

where $\boldsymbol{w}_j \geq \boldsymbol{0}$ is the jth column of $\boldsymbol{W}$, and $\boldsymbol{h}^j \geq \boldsymbol{0}$ is the jth row of $\boldsymbol{H}$. Here, $\boldsymbol{w}_j\boldsymbol{h}^j$ represents the jth rank-1 factor in $\boldsymbol{W}\boldsymbol{H}$.

r **is important** The parameter r controls the model complexity of NMF and plays a critical role in data analysis. In signal processing Leplat et al. (2020), r represents the number of sources in an audio signal. If r is overestimated, overfitting occurs, where the extra components in the model capture noise (e.g., piano mechanical noise (Ang, 2020, Sect.4.2)) rather than meaningful information.

r **is unknown** Generally, r is unknown. Finding r in $\texttt{NMF}(\boldsymbol{M}, r)$ for $\text{rank}_+(\boldsymbol{M}) \geq 3$ is NP-hard Vavasis (2010)[1]. In many cases, $\text{rank}(\boldsymbol{M})$ and/or $\text{rank}_+(\boldsymbol{M})$ are small since $\boldsymbol{M}$ is approximately low-rank Udell and Townsend (2019) and/or has low nonnegative rank (Gillis, 2020, Sect.9.2). Heuristics have been proposed to find r. Besides trial-and-error, the two main groups of methods for estimating r are stochastic/information-theoretic and algebraic/deterministic. The first group includes Bayesian methods Tan and Févotte (2012), the cophenetic correlation coefficient Esposito et al. (2020), and minimum description length Squires et al. (2017). The second group includes fooling sets Cohen and Rothblum (1993) and the f-vector in combinatorics Dewez et al. (2021). See (Gillis, 2020, Sect.3) for a summary of the algebra of rank_+.

In this work, we focus on approximately solving $\texttt{NMF}(\boldsymbol{M}, r)$ without knowing r in advance. This is achieved by imposing a "rank penalty" on NMF. Instead of using the nuclear norm nor the rank itself as a penalty term, we consider a clustering regularizer called Sum-of-norms (SON): we propose SON-NMF to "relax" the assumption of knowing r. Before we introduce SON-NMF, we first review the SON term.

[1] Note that $\text{rank}_+(\boldsymbol{M})$ is not the same as $\text{rank}(\boldsymbol{M})$, which can be computed by eigendecomposition or singular value decomposition. See Gillis (2020) for solving $\texttt{NMF}(\boldsymbol{M}, r)$ in the case $\text{rank}_+(\boldsymbol{M}) \leq 2$.

Matrix $\ell_{p,q}$-norm The $\ell_{p,q}$-norm of a matrix $\boldsymbol{X} \in \mathbb{R}^{m \times n}$ is defined as

$$\|\boldsymbol{X}\|_{p,q} \;:=\; \left(\sum_{j=1}^{n}\left(\sqrt[p]{\sum_{i=1}^{m} X_{ij}^{p}}\right)^{q}\right)^{\frac{1}{q}} \;=\; \left\|\begin{bmatrix}\|\boldsymbol{x}_1\|_p \\ \vdots \\ \|\boldsymbol{x}_n\|_p\end{bmatrix}\right\|_q ,$$

where, in the last equality, we first take the p-norm of each column and then take the q-norm of the resulting vector. A popular choice of the $\ell_{p,q}$-norm is the $\ell_{2,1}$-norm, which is widely used in the multiple measurement vector problem Cotter et al. (2005), sparse coding Nie et al. (2010), and robust NMF Kong et al. (2011).

Sum-of-norms (SON) We define the SON of a matrix $\boldsymbol{X}$ as the $\ell_{2,1}$-norm of $P(\boldsymbol{X})$, where $\boldsymbol{X} \mapsto P(\boldsymbol{X})$ is all the pairwise difference $\boldsymbol{x}_i - \boldsymbol{x}_j$. As $\|\boldsymbol{x}_i - \boldsymbol{x}_j\|_2 = \|\boldsymbol{x}_j - \boldsymbol{x}_i\|_2$, there are $\frac{n^2-n}{2}$ terms in SON of $\boldsymbol{X}$. In this work, we propose using $\text{SON}_{2,1}(\boldsymbol{W})$ as a regularizer for NMF, to be presented in the next section. Below, we give remarks on $\text{SON}_{2,q}(\boldsymbol{W})$ for other choices of q.

- $\text{SON}_{2,0}(\boldsymbol{W})$ with $q = 0$: It is trivial that rank$(\boldsymbol{W}) \leq \text{SON}_{2,0}(\boldsymbol{W})$, because the set of linearly independent vectors is a subset of the set of unequal vector pairs. By the combinatorial nature of the ℓ_0-norm, minimizing $\text{SON}_{2,0}(\boldsymbol{W})$ is NP-hard, and its complexity scales with r. Therefore, $\text{SON}_{2,0}(\boldsymbol{W})$ is computationally unfavourable for NMF applications with large $r \approx (m, n)$, which is the case in this work.
- $\text{SON}_{2,2}(\boldsymbol{W})$ with $q = 2$: By definition, this is the Frobenius norm of $P(\boldsymbol{W})$. This SON has been used in graph-regularized NMF Cai et al. (2010), but it differs from (SON-NMF) for two reasons: (1) the graph regularizer is a weighted squared $\text{SON}_{2,2}$ norm, which is everywhere differentiable, unlike $\text{SON}_{2,1}(\boldsymbol{W})$; and (2) $\text{SON}_{2,2}(\boldsymbol{W})$ does not induce sparsity, whereas $\text{SON}_{2,1}(\boldsymbol{W})$ does.
- $\text{SON}_{2,\infty}(\boldsymbol{W})$ with $q \to \infty$: This term focuses on the pair $(\boldsymbol{w}_i, \boldsymbol{w}_j)$ that is mutually furthest apart, ignoring the rest. This is unfavourable for removing redundant $\boldsymbol{w}_j$ in NMF for the purposes of this work.

We are now ready to introduce SON-NMF.

SON-NMF We propose to regularize NMF by $\text{SON}_{2,1}(\boldsymbol{W}) = \sum_{i \neq j} \|\boldsymbol{w}_i - \boldsymbol{w}_j\|_2$ as

$$\begin{aligned}\operatorname*{argmin}_{\boldsymbol{W},\boldsymbol{H}} F(\boldsymbol{W},\boldsymbol{H}) := \;& \frac{1}{2}\|\boldsymbol{W}\boldsymbol{H} - \boldsymbol{M}\|_F^2 + \lambda \sum_{i \neq j} \|\boldsymbol{w}_i - \boldsymbol{w}_j\|_2 \\ & + \gamma \sum_i \|\max\{-\boldsymbol{w}_i, \boldsymbol{0}\}\|_1 + \iota_{\Delta^r}(\boldsymbol{H}),\end{aligned} \tag{SON-NMF}$$

where $\frac{1}{2}\|\boldsymbol{M} - \boldsymbol{W}\boldsymbol{H}\|_F^2 : \mathbb{R}^{m \times n} \times \mathbb{R}^{m \times r} \times \mathbb{R}^{r \times n} \to \mathbb{R}$ is a smooth, nonconvex data-fitting term, the constants $\lambda > 0$ and $\gamma > 0$ are parameters, the functions $\sum_i \|\max\{-\boldsymbol{w}_i, \boldsymbol{0}\}\|_1$ and $\iota_{\Delta^r}(\boldsymbol{H}) = \sum_j \iota_{\Delta^r}(\boldsymbol{h}_j)$ are nonsmooth, lower-semicontinuous, proper convex functions representing model constraints: respectively, the nonnegativity of $\boldsymbol{w}_j$ (i.e., $\boldsymbol{W} \geq$

$\mathbf{0}$) and the requirement that $\boldsymbol{h}_j$ lies in the r-dimensional unit simplex (i.e., $\boldsymbol{H}$ is elementwise nonnegative and $\boldsymbol{H}^\top \mathbf{1}_r \leq \mathbf{1}_n$, where $\mathbf{1}_r \in \mathbb{R}^r$ denotes a vector of ones). Note that in (SON-NMF) we use the penalty $\sum_i \| \max\{-\boldsymbol{w}_i, \mathbf{0}\}\|_1$, which enforces nonnegativity $\boldsymbol{W} \geq \mathbf{0}$ for sufficiently large λ, this will be explained in section 4. We defer the definition of symbols used in (SON-NMF) to the end of this section.

SON encourages multicollinearity and rank-deficiency for NMF The SON term encourages the pairwise difference in $\|\boldsymbol{w}_i - \boldsymbol{w}_j\|_2$ to be small, potentially resulting in multicollinearity in the matrix $\boldsymbol{W}$. Note that in traditional regression models, multicollinearity is strongly discouraged due to its negative statistical effects on the variables Farrar and Glauber (1967). In this work, we intentionally promote multicollinearity in $\boldsymbol{W}$ to encourage rank deficiency, which helps reduce an overestimated rank during rank estimation. In other words, SON-NMF can be seen as the ordinary NMF model with a multicollinearity regularizer: the rank of $\boldsymbol{W}$ is overestimated at the first iteration, and the regularizer gradually reduces it to the correct value during the algorithmic process.

There is a "price to pay" for such multicollinearity. If $\boldsymbol{W}$ is near-multicollinear, its condition number is large, making $\boldsymbol{W}^\top\boldsymbol{W}$ ill-conditioned and negatively affecting the process of updating $\boldsymbol{H}$. See the discussion in Section 3.

Contributions We introduce a new problem (SON-NMF) with the contributions:

- **Empirically rank-revealing.** On synthetic and real-world datasets, we empirically show that model (SON-NMF), free from tuning the rank r, will itself find the correct r in the data automatically when r is overestimated. This is due to the sparsity-inducing property of the $\ell_{2,1}$ norm in $\text{SON}_{2,1}$.
  - **Rank-deficient compatibility.** SON-NMF can handle rank-deficient problems, i.e., data matrices whose true rank is smaller than the overestimated parameter r. This has two advantages. First, it prevents overfitting. Second, compared with existing NMF models such as minimum-volume NMF Ang and Gillis (2018); Leplat et al. (2020), which were shown to exhibit rank-finding ability Leplat et al. (2019), SON-NMF is applicable to rank-deficient matrices.
- **Irreducible computational complexity.** As computing rank_+ is NP-hard, the SON approach, as a "work-around" method to estimate rank_+, cannot reduce computational complexity. We prove (Theorem 1) that the complexity of the SON term is *almost irreducible*. Precisely, we show that in the best case, to recover the r^* columns of the true $\boldsymbol{W}^*$ using $\boldsymbol{W}$ obtained from SON-NMF with rank $r > r^*$, the complexity of the SON term cannot be reduced from $r(r-1)/2$ to below $r(r - \lceil r/r \rceil)/2$.
- **Fast algorithm by proximal-average.** Solving (SON-NMF) is nontrivial: the $\boldsymbol{W}$-subproblem is nonsmooth, non-separable, and non-proximable, so existing proximal-based methods Tseng and Yun (2009); Xu and Yin (2013); Razaviyayn et al. (2013); Bolte et al. (2014); Le et al. (2020) cannot efficiently solve the problem. For non-proximal problems, dual approaches such as Lagrange multipliers or ADMM are typically used. However, SON-NMF involves $\mathcal{O}(r^2)$ non-proximal terms, and this

complexity is irreducible (Theorem 1). Therefore, dual and 2d-order methods are inefficient due to their high per-iteration cost. We propose a low-cost proximal average approach Yu (2013) based on the Moreau-Yosida envelope Bauschke et al. (2008).

We review the literature, focusing on the background and motivation of this work.

Review of NMF: minimum-volume and rank-deficiency SON-NMF is related to minimum-volume (minvol) NMF Ang and Gillis (2018, 2019). Recently, it was observed in Leplat et al. (2019) that when using volume regularization in the form of $\log\det(\boldsymbol{W}^\top\boldsymbol{W} + \delta\boldsymbol{I}_r)$, minvol NMF applied to a rank-deficient matrix $\boldsymbol{M}$ (i.e., when the r parameter is overestimated) can zero out the extra components in $\boldsymbol{W}$ and $\boldsymbol{H}$. This phenomenon was also observed in audio blind source separation Leplat et al. (2020), where a rank-7 factorization was applied to a dataset with 3 sources: the minvol NMF was able to set the redundant components to zero. However, minvol NMF is not suitable for rank-deficient $\boldsymbol{W}$: if $\delta = 0$, then $\log\det(\boldsymbol{W}^\top\boldsymbol{W}) = \log 0 = -\infty$. Even if $\delta \neq 0$, a rank-deficient $\boldsymbol{W}$ provides little information in the log-det term. Furthermore, in Leplat et al. (2020), when using an overestimated rank in minvol NMF, it is the redundant components in $\boldsymbol{H}$ that are set to zero, rather than those in $\boldsymbol{W}$. We remark that this rank-revealing property of minvol NMF motivated the first author to propose SON-NMF.

Review of SON SON was originally proposed in Pelckmans et al. (2005); Lindsten et al. (2011) for clustering. Because minimizing $\text{SON}(\boldsymbol{W})$ forces the pairwise differences $\boldsymbol{w}_i - \boldsymbol{w}_j$ to be small, SON is also referred to as a "fusion penalty" Hocking et al. (2011). Later, Niu et al. (2016) considered SON with $0 < p < 1$, and more recently, Jiang and Vavasis (2020) showed that SON-based clustering can provably recover Gaussian mixtures under certain assumptions. $\text{SON}_{2,0}$ has also been applied in graph trend filtering Huang et al. (2025). We note that these works differ from SON-NMF: they involve single-variable problems, whereas NMF is a bi-variate, nonconvex problem with nonnegativity constraints.

SON solution approaches The approach we propose to solve the SON problem differs from existing methods such as quadratic programming with convex hull Pelckmans et al. (2005), active-set methods Hocking et al. (2011), interior-point methods Lindsten et al. (2011), trust-region methods with smoothing Niu et al. (2016), Lagrange multiplier methods (Beck, 2017, 12.3.8), and semi-smooth Newton methods Yuan et al. (2018). These approaches were all designed for single-variable clustering problems (i.e., involving $\boldsymbol{W}$ only) without nonnegativity constraints. In contrast, we leverage the proximal average Bauschke et al. (2008); Yu (2013), which is computationally inexpensive to compute, with a per-iteration cost of $\mathcal{O}(m)$, where m is the dimension of $\boldsymbol{w}_j$. This significantly lowers the per-iteration cost for SON in our setting. All the aforementioned methods are either unable to handle the SON problem with nonnegativity constraints on $\boldsymbol{W}$ or incur higher per-iteration costs. See details in section 4.

History: the geometric median and the Fermat-Torricelli-Weber problem SON was proposed in the 2000s Pelckmans et al. (2005); Hocking et al. (2011); Lindsten et al.

(2011), it is closely related to an older problem known as the Fermat-Torricelli-Weber problem Krarup and Vajda (1997); Nam et al. (2014), also called as the geometric median (Beck, 2017, E.g.3.66). The analysis of the geometric median does not directly apply to SON-NMF, but it provides a geometric interpretation: SON-NMF produces an r^*-cluster of points that minimizes the geometric median distance to the dataset.

Rank estimation in NMF Existing methods for rank estimation in NMF are not applicable in the setting of this paper. Algebraic approaches, such as fooling sets Cohen and Rothblum (1993) and the f-vector Dewez et al. (2021), only provide loose bounds on $\text{rank}_+(\boldsymbol{M})$ and are computationally expensive to implement. Statistical approaches Tan and Févotte (2012); Squires et al. (2017); Esposito et al. (2020) assume that $\boldsymbol{W}$ and $\boldsymbol{H}$ follow predefined distributions or require heavy post-processing. SON-NMF makes none of these assumptions and requires no post-processing.

A "drawback" of SON-NMF Finding rank_+ in NMF is NP-hard, and the search space of r in NMF is the set of natural numbers $\mathbb{N}$, which is countably infinite. In SON-NMF, we do not need to estimate the rank r, but we must provide a regularization parameter λ, whose search space is the set of nonnegative real numbers $\mathbb{R}_+$. By Cantor's diagonal argument Cantor (1890), the cardinality of the real numbers is uncountably infinite. Hence, theoretically, SON-NMF replaces the search space $\mathbb{N}$ of NMF with the much larger space $\mathbb{R}_+$, suggesting that SON-NMF could be even more difficult to solve than the already NP-hard NMF. We remark, however, that this is not an issue in practice: many datasets are hierarchically clustered in the latent space, so a simple tuning of λ is sufficient for SON-NMF to recover the true rank.

Paper organization We present the theory of SON-NMF section 2. We describe how to solve SON-NMF in section 3 and section 4. In section 5, we show experimental results, and section 6 concludes the paper.

Notation The notation "$\{x, y\}$ denotes $\{X, Y\}$" means that X is denoted by x and Y by y, respectively (resp.). We use $\{\mathbb{R}, \mathbb{R}_+, \overline{\mathbb{R}}, \mathbb{R}^m, \mathbb{R}^{m\times n}\}$ to denote {reals, nonnegative reals, extended reals, m-dimensional reals, m-by-n reals}, we use {lowercase italic, bold lowercase italic, bold uppercase letters} to denote {scalar, vector, matrix}. Given a matrix $\boldsymbol{M}$, we denote $\{\boldsymbol{m}^i, \boldsymbol{m}_j\}$ the $\{i$th row, jth column$\}$ of $\boldsymbol{M}$. Given a convex set $C \subset \mathbb{R}^n$, the indicator function of C at $\boldsymbol{x}$ is defined as $\iota_C(\boldsymbol{x}) = 0$ if $\boldsymbol{x} \in C$ and $\iota_C = +\infty$ if $\boldsymbol{x} \notin C$, and $\text{proj}_C(\boldsymbol{x})$ denotes the projection of $\boldsymbol{x}$ onto C. The projection of $\{\boldsymbol{v} \in \mathbb{R}^n, \boldsymbol{V} \in \mathbb{R}^{m\times n}\}$ onto the nonnegative orthant $\{\mathbb{R}^n_+, \mathbb{R}^{m\times n}_+\}$ is denoted by the element-wise max operator $\{[\boldsymbol{v}]_+, [\boldsymbol{V}]_+\}$. Lastly, $\Delta^r \subset \mathbb{R}^r$ denotes the unit simplex, and $\mathbf{1}_r \in \mathbb{R}^r$ is the vector of ones.

Remark. *The constraint on $\boldsymbol{H}$ removes the scaling ambiguity of the factorization. I.e., for $\boldsymbol{M} = \boldsymbol{W}_1\boldsymbol{H}_1$, there does not exist a diagonal matrix $\boldsymbol{D}$ such that $\boldsymbol{M} = (\boldsymbol{W}_1\boldsymbol{D})(\boldsymbol{D}^{-1}\boldsymbol{H}_1) =: \boldsymbol{W}_2\boldsymbol{H}_2$ with $\boldsymbol{W}_1 \neq \boldsymbol{W}_2$ and $\boldsymbol{H}_1 \neq \boldsymbol{H}_2$.*

2 Theory of SON-NMF

In this section, we present the theory of $\text{SON}_{2,1}$-NMF. We first examine $\text{SON}_{2,1}$ in detail, then motivate why reducing its computational complexity is desirable, and provide a bound showing that this complexity is essentially irreducible.

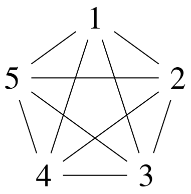
2.1 $\text{SON}_{2,1}$ has r^2 terms and its minimum occurs at maximal cluster imbalance

Let $\boldsymbol{X} = [\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots]$ have five columns. The pairwise differences $\boldsymbol{x}_i - \boldsymbol{x}_j$ in SON yield $5^2 - 5 = 20$ pairs. In general, if $\boldsymbol{X}$ has r columns, there are $r(r-1)$ ordered pairs $(\boldsymbol{x}_i, \boldsymbol{x}_j)$. By symmetry, $|\boldsymbol{x}_i - \boldsymbol{x}_j|_2 = |\boldsymbol{x}_j - \boldsymbol{x}_i|_2$, so we consider only the $r(r-1)/2$ *distinct* pairs in SON. We can describe this in graph-theoretic terms. Let $G(V, E)$ be a simple, undirected, unweighted graph with $|V|$ nodes and $|E|$ edges, and let K_r be the complete graph on r nodes. Then

$$\text{SON}_{2,1}(\boldsymbol{X}) = \sum_{(i,j)\in E(K_r)} \|\boldsymbol{x}_i - \boldsymbol{x}_j\|_2$$

Since $|E(K_r)| = r(r-1)/2$, $\text{SON}_{2,1}$ contains $\mathcal{O}(r^2)$ terms.

Returning to the example with five columns, let $\boldsymbol{X}$ be a rank-3 matrix with three clusters having centers $\boldsymbol{c}_1, \boldsymbol{c}_2, \boldsymbol{c}_3$: $\boldsymbol{X} = [\boldsymbol{x}_1 \mid \boldsymbol{x}_2 \mid \boldsymbol{x}_3\ \boldsymbol{x}_4\ \boldsymbol{x}_5] = [\boldsymbol{c}_1 \mid \boldsymbol{c}_2 \mid \boldsymbol{c}_3\ \boldsymbol{c}_3\ \boldsymbol{c}_3]$. Then $\text{SON}_{2,1}(\boldsymbol{X}) = \|\boldsymbol{c}_1 - \boldsymbol{c}_2\|_2 + 3\|\boldsymbol{c}_1 - \boldsymbol{c}_3\|_2 + 3\|\boldsymbol{c}_2 - \boldsymbol{c}_3\|_2$. The graph K_5 and the corresponding pairwise differences are illustrated below.



	1	2	3	4	5
1	0	$\boldsymbol{c}_1 - \boldsymbol{c}_2$	$\boldsymbol{c}_1 - \boldsymbol{c}_3$	$\boldsymbol{c}_1 - \boldsymbol{c}_3$	$\boldsymbol{c}_1 - \boldsymbol{c}_3$
2		0	$\boldsymbol{c}_2 - \boldsymbol{c}_3$	$\boldsymbol{c}_2 - \boldsymbol{c}_3$	$\boldsymbol{c}_2 - \boldsymbol{c}_3$
3			0	0	0
4				0	0
5					0

Now we generalize. Let $\boldsymbol{X}$ have r columns and $r^* \leq r$ clusters $C_1, \ldots, C_{r^*}$ with centers $\boldsymbol{c}_1, \ldots, \boldsymbol{c}_{r^*}$. Let $|C_i|$ denote the size of cluster C_i. Since $|C_1| + \cdots + |C_{r^*}| = r$, we have

$$\begin{aligned}\text{SON}_{2,1}(\boldsymbol{X}) = \sum_{(i,j)\in K_r} |C_i||C_j|\big\|\boldsymbol{c}_i - \boldsymbol{c}_j\big\|_2 &\leq \Big(\max_{i\in[r]} |C_i|\Big)\Big(\max_{i,j} \big\|\boldsymbol{c}_i - \boldsymbol{c}_j\big\|_2\Big) \sum_{(i,j)\in K_r} |C_j| \\ &\leq \Big(\max_{i\in[r]} |C_i|\Big)\Big(\max_{i,j} \big\|\boldsymbol{c}_i - \boldsymbol{c}_j\big\|_2\Big) r,\end{aligned}$$

giving a stopping criterion for the algorithm: we have r as input, so we just need to track the product $\big(\max_{i\in[r]} |C_i|\big)\big(\max_{i,j} \big\|\boldsymbol{c}_i - \boldsymbol{c}_j\big\|_2\big)$ for convergence. Furthermore, from the inequality, we can focus on the cluster sizes rather than the norms $\|\boldsymbol{c}_i - \boldsymbol{c}_j\|_2$, leading to the following lemma that characterizes the theoretical minimum of $\text{SON}_{2,0}$ as a proxy for $\text{SON}_{2,1}$:

Lemma 1 (Maximal cluster imbalance gives the minimum of $\mathrm{SON}_{2,0}$)**.** *For an n-column matrix $\boldsymbol{X}$ with K clusters $C_1, \ldots, C_K$, where all $\boldsymbol{x}_i \in C_i$ take the centroid $\boldsymbol{c}_i$,*

$$SON_{2,0}(\boldsymbol{X}) = \sum_{i,j} |C_i||C_j|$$

achieves its minimum when one cluster contains $n - K + 1$ columns of $\boldsymbol{X}$, and the remaining $K - 1$ clusters each have size one.

Proof. Directly by $|C_1| + \cdots + |C_K| = n$ and basic inequality manipulations. □

Remarks of Lemma 1

1. Since the ℓ_1-norm is a tight convex relaxation of the ℓ_0-norm (over the unit ball), Lemma 1 for the $\mathrm{SON}_{2,0}$ term provides a theoretical minimum for the $\mathrm{SON}_{2,1}$ term when the input matrix lies within a unit ball.

2. For any cluster C_i, the smallest cluster size is 1, so the $\mathrm{SON}_{2,0}$ term (and similarly the $\mathrm{SON}_{2,1}$ term) cannot "miss" a weak component in the data if one exists. This behavior is observed in experiments (see Figs. 7, 5). Furthermore, if the cluster centers $\boldsymbol{c}_i$ are approximately equidistant ($\|\boldsymbol{c}_i - \boldsymbol{c}_j\|_2 \approx \|\boldsymbol{c}_j - \boldsymbol{c}_k\|_2$), maximal cluster imbalance naturally occurs in applications (see Figs. 3, 4, 6, 7).

2.2 SON complexity is irreducible

We now see that SON has $\mathcal{O}(r^2)$ terms. In practice, SON-NMF often uses a large input rank r-potentially as large as the data dimensions m or n-to estimate the true rank r^*. This makes the SON term computationally expensive. A natural question arises: can we reduce the complexity of the SON term by removing some edges in K_r, thereby lowering the per-iteration cost of SON-NMF while preserving recovery performance? The answer is negative, as stated in Theorem 1: the complexity of the SON term is essentially irreducible.

Remark. *There are prior works explored similar ideas. For example, (Yuan et al., 2018, page 2) mentions approaches using k-nearest neighbors. However, these are data-dependent methods that leverage the data to learn a graph structure for reducing the complexity of the SON term. Our focus is different. We investigate the possibility of reducing the complexity of the SON term purely from a graph-theoretic perspective, independent of the data. That is, we are interested in whether a sparsest subgraph exists such that the reduced SON is "functionally the same" as the full-SON. Theorem 1 shows that such sparsest subgraph basically does not exist.*

We introduce notation. Let r^* be the true NMF rank of $\boldsymbol{W}$. For a graph $G(V, E)$, let $u, v \in V$ be two nodes connected by an edge, i.e., $(u, v) \in E$. The notation $G \setminus (u, v)$ denotes the subgraph obtained by removing the edge (u, v) from G. A *graph partition* of G is a set of subgraphs $S_1, S_2, \ldots$ such that the vertex sets $V(S_i)$ form a mutually exclusive partition of $V(G)$.

We now state a trivial fact.

Lemma 2. *Let $\boldsymbol{W}$ have NMF rank r^*. Then the graph generated by the columns of $\boldsymbol{W}$ satisfies the following property: for every partition of its nodes into r^* subgraphs, each subgraph must be connected.*

This lemma can be easily proved by contradiction. Next we have a lemma.

Lemma 3. *The only graph satisfying the condition of Lemma 2 is the complete graph.*

Proof. Let G be a graph whose nodes correspond to the columns of $\boldsymbol{W}$. Suppose some edge (u, v) is omitted from G. Then it is possible for the 'true' partition to place u and v in the same subgraph, while the remaining nodes of $G \setminus (u, v)$ are divided arbitrarily among the other $r^* - 1$ subgraphs. In this case, u and v are no longer directly connected, violating the connectivity requirement. Since u and v were arbitrary, no edge can be omitted from G, and thus G must be complete. □

This lemma implies that, apart from the complete graph, no other graph structure satisfies the connectivity condition. The following theorem quantifies how many edges can be removed from the complete graph, which tells essentially "none".

Theorem 1. *Let $(\boldsymbol{W}^*, \boldsymbol{H}^*) = \text{NMF}(\boldsymbol{M}, r^*)$ and let $(\boldsymbol{W}, \boldsymbol{H}) = \text{SON-NMF}(\boldsymbol{M}, r)$ with $r \geq r^*$. If we aim to recover $\boldsymbol{W}^*$ using r^* clusters among the r columns of $\boldsymbol{W}$, the number of pairwise difference terms $|\boldsymbol{w}_i - \boldsymbol{w}_j|_2$ in SON cannot be reduced below*

$$\frac{r}{2}\left(r - \left\lceil \frac{r}{r^*} \right\rceil\right).$$

Proof. Represent the r columns of $\boldsymbol{W}$ as nodes of a simple undirected graph $G(V, E)$, where each edge (u, v) corresponds to the term $|\boldsymbol{w}_i - \boldsymbol{w}_j|_2$. Recovering $\boldsymbol{W}^*$ by $\boldsymbol{W}$ with fewer SON terms can be translated as

we can identify r^* disjoint clusters in G with $|V| = r$
using a subgraph of K_r with fewer edges. (1)

We are going to show that the statement (1) is true, and at best such an improvement is from $r(r-1)/2$ to $r(r - \lceil r/r^* \rceil)/2$.

Assuming, in the best case that, each of these r^* columns of $\boldsymbol{W}^*$ is corresponds to exactly $\lceil r/r^* \rceil$ nodes in $\boldsymbol{W}$, represented by the nodes in the graph G. For any node $v \in V$, let $S(v) \subset V$ be the set of nodes disconnected[2] from v, and let T be a nonempty subset of $S(v)$. Then the recovery of the r^* clusters in G is impossible if a cluster in G is of the form $\{v\} \cup T$. The negation of this very last statement gives:

To recover the r^* clusters for all subset of nodes of size at least r/r^*,
we need $|T| < r/r^*$ for any such T.

The inequality $|T| < r/r^*$ holds for all subset T of $S(v)$, this implies $|S(v)| < r/r^*$. I.e., v has to connect to at least $r - \lceil r/r^* \rceil$ other nodes $u \notin S(v)$ in the graph G. This connectivity holds for every node $v \in V$, meaning that the minimum number of edges required is $r\big(r - \lceil r/r^* \rceil\big)/2$. □

[2] I.e., there is no path between $u \in S(v)$ and v

Theorem 1 shows that the number of edges in SON cannot be significantly reduced from $r(r-1)/2$. We define the reduction factor $R(r^*, r)$ as

$$\begin{aligned} R(r; r^*) &:= \frac{\text{full number of terms} - \text{reduced number of terms}}{\text{full number of terms}} \\ &= \frac{\frac{r}{2}(r-1) - \frac{r}{2}\big(r - \lceil r/r^* \rceil\big)}{\frac{r}{2}(r-1)}. \end{aligned}$$

The following lemma tells that the reduction factor is small.

Lemma 4. *For fixed* r^**, the value* $R(r^*, r)$ *approaches to* $1/r^*$ *as* $r \to \infty$*.*

Proof. Take the limit of $R(r; r^*)$ gives $\lim\limits_{r\to\infty} R(r; r^*) = \lim\limits_{r\to\infty} \dfrac{\lceil r/r^* \rceil - 1}{r-1} = \lim\limits_{r\to\infty} \dfrac{\lceil r/r^* \rceil}{r-1}$. Using $r \leq \lceil r \rceil \leq r+1$, we have $\lim\limits_{r\to\infty} \dfrac{r/\lceil r^* \rceil}{r-1} \leq \lim\limits_{r\to\infty} R(r; r^*) \leq \lim\limits_{r\to\infty} \dfrac{(r+1)/\lceil r^* \rceil}{r-1}$. By squeeze theorem, $\lim\limits_{r\to\infty} R(r; r^*) = 1/r^*$. □

The lemma shows that the SON term's complexity can only be reduced marginally. For $r^* \geq 3$ (NMF is trivial for $r^* \leq 2$ Gillis (2020)), the maximum reduction is about 33%. This reduction quickly decreases as r or r^* increases. For example, with $(r, r^*) = (1000, 25)$, using 1000 nodes to find 25 clusters, the number of edges in K_{1000} can be reduced by at most 5%, from $|K_{1000}| = 499{,}500$ to $500(1000 - \lceil 1000/25 \rceil) = 480{,}000$.

3 BCD algorithm and the H-subproblem

We now discuss solving the nonsmooth, nonconvex, nonseparable, and non-proximable problem (SON-NMF) via block coordinate descent (BCD) Hildreth (1957); Wright (2015). Let k denotes the iteration counter. Starting with an initial guess $(\boldsymbol{W}_1, \boldsymbol{H}_1)$, we perform alternate update : $\boldsymbol{H}_{k+1} \leftarrow \text{update}(\boldsymbol{H}_k; \boldsymbol{W}_k)$, $\boldsymbol{W}_{k+1} \leftarrow \text{update}(\boldsymbol{W}_k; \boldsymbol{H}_{k+1})$, where update() denotes an approximate solution to the respective subproblem. In this section, we describe the BCD framework and the update for $\boldsymbol{H}$. The $\boldsymbol{W}$-subproblem is discussed in the next section.

Algorithm 1 shows the pseudo-code of the BCD method for solving SON-NMF.

Algorithm 1: (Inexact) BCD for solving SON-NMF

```
  Input: $\boldsymbol{M}, \boldsymbol{W}_1, \boldsymbol{H}_1, \lambda, \gamma$
1 for $k = 1, 2, \ldots$ do
2     $\boldsymbol{H}_{k+1} = \text{proj}_{\Delta^r}\Big(\boldsymbol{Q}\boldsymbol{H}_k + \boldsymbol{R}\Big)$ with $\boldsymbol{Q} = \boldsymbol{I}_n - \boldsymbol{W}_k^\top \boldsymbol{W}_k / \|\boldsymbol{W}_k^\top \boldsymbol{W}_k\|_2$ and
       $\boldsymbol{R} = \boldsymbol{W}_k^\top \boldsymbol{M} / \|\boldsymbol{W}_k^\top \boldsymbol{W}_k\|_2$
3     for $\ell = 1, 2, \ldots, \ell_{max}$, (e.g., 10) do
4         $\boldsymbol{W}_{k+1} = \text{update}(\boldsymbol{W}_k; \boldsymbol{H}_{k+1}, \boldsymbol{M}, \lambda, \gamma)$, see section 4.
```

We now explain Step 2 in Algorithm 1.

H-subproblem: projection onto unit simplex The step update$(\boldsymbol{H}_k; \boldsymbol{W}_k)$ solves the subproblem on $\boldsymbol{H}$, which consists of n parallel problems:

$$\operatorname*{argmin}_{\boldsymbol{h}_1,\ldots,\boldsymbol{h}_n} \frac{1}{2}\sum_{j=1}^{n} \|\boldsymbol{W}_k\boldsymbol{h}_j - \boldsymbol{m}_j\|_2^2 \;\text{ s.t. }\; \boldsymbol{h}_j \in \Delta^r \text{ for } j = 1, 2, \ldots, n, \tag{2}$$

where $\Delta^r := \big\{\boldsymbol{x} \in \mathbb{R}^r_+ : \sum_i x_i \leq 1\big\}$. Each column $\boldsymbol{h}_j$ solves a a constrained least-squares:

$$\operatorname*{argmin}_{\boldsymbol{x}\in\Delta^r} f(\boldsymbol{x}) \;=\; \frac{1}{2}\|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{b}\|_2^2 \;=\; \frac{1}{2}\langle \boldsymbol{A}^\top\boldsymbol{A}\boldsymbol{x}, \boldsymbol{x}\rangle - \langle \boldsymbol{A}^\top\boldsymbol{b}, \boldsymbol{x}\rangle, \tag{3}$$

where $\boldsymbol{x}$ is the variable $\boldsymbol{h}_j$, and $\boldsymbol{A} = \boldsymbol{W}^\top\boldsymbol{W}$ with $\boldsymbol{b} = \boldsymbol{W}^\top\boldsymbol{m}_j$. We use proximal gradient method (details in the next section) to update $\boldsymbol{x}$ in (3) iteratively as

$$\boldsymbol{x}_k^{\ell+1} = \text{proj}_{\Delta^r}\left(\boldsymbol{x}_k^\ell - \frac{\boldsymbol{A}^\top\boldsymbol{A}\boldsymbol{x}_k^\ell - \boldsymbol{A}^\top\boldsymbol{b}}{\|\boldsymbol{A}^\top\boldsymbol{A}\|_2}\right), \tag{4}$$

where $\boldsymbol{x}_k^\ell$ is the variable at iteration-k and inner-iteration-ℓ. For low per-iteration cost, we typically set $\ell = 1$.

Projection $\text{proj}_{\Delta^r}(\boldsymbol{x})$ projects $\boldsymbol{x} \in \mathbb{R}^r$ onto the unit simplex Δ^r with cost $\mathcal{O}(r \log r)$ Condat (2016), due to sorting when computing the Lagrange multiplier.

Matrix update The column-wise updates can be combined into a matrix update:

$$\boldsymbol{H}_{k+1} = \text{proj}_{\Delta^r}\left(\boldsymbol{H}_k - \frac{\boldsymbol{W}_k^\top\boldsymbol{W}_k\boldsymbol{H}_k - \boldsymbol{W}_k^\top\boldsymbol{M}}{\|\boldsymbol{W}_k^\top\boldsymbol{W}_k\|_2}\right),$$

with proj_{Δ^r} applied in parallel to each column. The total cost is $\mathcal{O}(nr \log r)$. For $r \approx n$, the cost is $\mathcal{O}(n^2 \log n)$, this partly explains why 2nd-order methods are impractical. Below we give another reason for not considering 2nd-order method for updating $\boldsymbol{H}$: the $\boldsymbol{W}$ is multicollinear.

Impact of $\boldsymbol{W}$-multicollinearity As SON encourages multicollinearity in $\boldsymbol{W}$, so $\boldsymbol{W}$ and $\boldsymbol{W}^\top\boldsymbol{W}$ may be ill-condotioned. This affects Problem (2):

1. The problem may not be strongly convex, possibly yielding multiple global minima.
2. Nesterov acceleration Nesterov (2003) becomes less effective due to the large condition number.
3. 2nd-order method are infeasible because $(\boldsymbol{W}_k^\top\boldsymbol{W}_k)^{-1}$ may not exists.
4. Duality-based tools (e.g., for stopping criteria) cannot be applied efficiently.

4 Proximal averaging on the W-subproblem

We now focus on solving the W-subproblem, the line update$(\boldsymbol{W};\boldsymbol{H},\boldsymbol{M},\lambda,\gamma)$ in Algorithm 1:

$$\underset{\boldsymbol{W}}{\operatorname{argmin}}\; F(\boldsymbol{W}) \;:=\; \frac{1}{2}\|\boldsymbol{W}\boldsymbol{H}-\boldsymbol{M}\|_F^2 + \lambda\sum_{i\neq j}\|\boldsymbol{w}_i-\boldsymbol{w}_j\|_2 + \gamma\sum_{j=1}^{r}\big\|\max\{-\boldsymbol{w}_j,\mathbf{0}\}\big\|_1. \tag{5}$$

The function $F(\boldsymbol{W})$ in (5) has the following properties:

- Convex and continuous: All terms are norms under convex-preserving maps. $\|\boldsymbol{w}_i-\boldsymbol{w}_j\|_2$, $\|\max\{-\boldsymbol{w}_j,\mathbf{0}\}\|_1$ are 1-Lipschitz.
- Nonsmooth: $\|\boldsymbol{w}_i-\boldsymbol{w}_j\|_2$ is non-differentiable at $\boldsymbol{w}_i=\boldsymbol{w}_j$ and $\|\max\{-\boldsymbol{w}_j,\mathbf{0}\}\|_1$ is non-differentiable at negative entries.
- Non-separable: $\boldsymbol{w}_i,\boldsymbol{w}_j$ are lumped together in SON, so $F(\boldsymbol{W})$ cannot be split into independent column-wise functions.
- Non-proximable: The prox operator for $\lambda\sum\|\boldsymbol{w}_i-\boldsymbol{w}_j\|_2+\gamma\sum\|\max\{-\boldsymbol{w}_j,\mathbf{0}\}\|_1$ has no closed-form solution and cannot be efficiently computed.
- Not dual-friendly: Introducing dual variables (e.g., for ADMM) increases the number of variables from r to r^2, which is impractical for large r.
- Not 2nd-order friendly: Computing Hessians or Newton steps is prohibitive, with per-iteration cost $\mathcal{O}(m^4)$ to $\mathcal{O}(m^5)$ if $r\sim m$.

Because of these properties, standard proximal gradient methods Tseng and Yun (2009); Xu and Yin (2013); Razaviyayn et al. (2013); Bolte et al. (2014); Le et al. (2020) are inefficient. Instead, we solve (5) using the Moreau-Yosida envelope with proximal averaging Yu (2013), which avoids parameter tuning required in inexact proximal Schmidt et al. (2011) or smoothing methods Nesterov (2005).

Remark. $\sum_i\|\max\{-\boldsymbol{w}_i,\mathbf{0}\}\|_1$ *enforces* $\boldsymbol{W}\geq\mathbf{0}$ *if* $\gamma>0$ *is sufficiently large.*

Column-wise update We solve (5) column by column. Consider the jth rank-1 component $\boldsymbol{w}_j\boldsymbol{h}^j$. Let $\boldsymbol{M}_j=\boldsymbol{M}-\boldsymbol{W}_{-j}\boldsymbol{H}^{-j}$ where $\boldsymbol{W}_{-j}$ is $\boldsymbol{W}$ without column $\boldsymbol{w}_j$ and $\boldsymbol{H}^{-j}$ is $\boldsymbol{H}$ without row $\boldsymbol{h}^j$. The subproblem (5) on $\boldsymbol{w}_j$ becomes

$$\boldsymbol{w}_j^* \;:=\; \underset{\boldsymbol{w}}{\operatorname{argmin}}\; \frac{\|\boldsymbol{h}^j\|_2^2}{2}\|\boldsymbol{w}\|_2^2 - \langle\boldsymbol{M}_j\boldsymbol{h}^{j\top},\boldsymbol{w}\rangle + \lambda\sum_{i\neq j}\|\boldsymbol{w}-\boldsymbol{w}_i\|_2 + \gamma\|\max\{-\boldsymbol{w},\mathbf{0}\}\|_1. \tag{6}$$

which can be cast in the general form

$$\underset{\boldsymbol{x}}{\operatorname{argmin}}\;\phi(\boldsymbol{x})+\psi(\boldsymbol{x}),\quad\text{where}\;\;\psi(\boldsymbol{x}):=\sum_{i=1}^{N}\alpha_i\psi_i(\boldsymbol{x}), \tag{7}$$

where $\phi : \mathbb{R}^m \to \mathbb{R}$ is closed, proper, convex, and smooth. Each $\psi_i : \mathbb{R}^m \to \overline{\mathbb{R}}$ is closed, proper, convex, and possibly nonsmooth. Coefficients $\alpha_i \geq 0$ are normalized ($\sum \alpha_i = 1$) by λ and γ. Note that ψ_i are non-separable, i.e., they share the same global variable $\boldsymbol{x}$.

Proximal gradient method A standard approach to solve minimization (7) is the proximal gradient method Passty (1979); Fukushima and Mine (1981); Combettes and Wajs (2005), in update under a stepsize $\mu > 0$ as $\boldsymbol{x}^+ = \mathrm{P}^\mu_\psi\big(\boldsymbol{x} - \mu\nabla\phi(\boldsymbol{x})\big)$, where P^μ_ψ is the proximal operator of ψ (see (8)). By $\psi(\boldsymbol{x}) := \sum_{i=1}^N \alpha_i\psi_i(\boldsymbol{x})$ in (7), we have $\mathrm{P}^\mu_\psi = \mathrm{P}^\mu_{\sum\alpha_i\psi_i}$ in which intractable, this is what we mean by ψ being "non-proximable". To address this, we employ the proximal average Bauschke et al. (2008); Yu (2013). Below we first give the background of proximal average for solving (7), then we explain its application to (6).

4.1 Proximal average

Given a point $\boldsymbol{v} \in \mathbb{R}^n$, a convex, closed, proper function $f : \mathbb{R}^n \to \mathbb{R} \cup \{+\infty\}$, and a parameter $\mu > 0$, the proximal operator of f at $\boldsymbol{v}$, denoted as $\mathrm{P}^\mu_f(\boldsymbol{v})$, and the Moreau-Yosida envelope of f at $\boldsymbol{v}$, denoted as $\mathrm{M}^\mu_f(\boldsymbol{v})$, are defined as

$$\begin{aligned}\mathrm{P}^\mu_f(\boldsymbol{v}) &:= \underset{\boldsymbol{\xi}}{\operatorname{argmin}}\ f(\boldsymbol{\xi}) + \frac{1}{2\mu}\|\boldsymbol{\xi} - \boldsymbol{v}\|_2^2,\\ \mathrm{M}^\mu_f(\boldsymbol{v}) &:= \min_{\boldsymbol{\xi}} f(\boldsymbol{\xi}) + \frac{1}{2\mu}\|\boldsymbol{\xi} - \boldsymbol{v}\|_2^2.\end{aligned} \tag{8}$$

Algorithm 2: Proximal average

```
for $k = 1, 2, \ldots$ do
    $\bar{\boldsymbol{x}} = \boldsymbol{x}_k - \mu\nabla\phi(\boldsymbol{x}_k)$
    $\boldsymbol{x}_{k+1} = \sum_{i=1}^N \alpha_i \mathrm{P}^\mu_{\psi_i}(\bar{\boldsymbol{x}})$
```

The idea of proximal average is that computing $\mathrm{P}^\mu_\psi = \mathrm{P}^\mu_{\sum\alpha_i\psi_i}$ directly is hard, while the individual $\mathrm{P}^\mu_{\psi_i}$ is easy to compute. We therefore approximate P^μ_ψ by $\sum_{i=1}^N \alpha_i \mathrm{P}^\mu_{\psi_i}$. Algorithm 2 implements this approach to solve (7). Under the assumptions that ϕ is L-smooth and ψ_i are all M_i-Lipschitz, the sequence $\{\boldsymbol{x}_k\}_{k\in\mathbb{N}}$ produced by Algorithm 2 converges to the minimizer of

$$\underset{\boldsymbol{x}}{\operatorname{argmin}}\ \phi(\boldsymbol{x}) + A(\boldsymbol{x}),\ \text{where}\ \mathrm{M}^\mu_A = \sum_i \alpha_i \mathrm{M}^\mu_{\psi_i}, \tag{\#}$$

with A called the proximal average of $\{\psi_1, \ldots, \psi_n\}$ Yu (2013). Moreover, we have $0 \leq \psi - A \leq \frac{\mu}{2}\sum_i \alpha_i M_i^2 < +\infty$ which implies that an $\epsilon-$solution for (#) is an 2ϵ-solution for (7).

4.2 Update on w

We now explain how to use proximal average (Algorithm 2) to solve the W-subproblem. First, let $\sigma = (r-1)\lambda + \gamma$ be a normalization factor and rewrite (6) as

$$\underset{\boldsymbol{w}}{\operatorname{argmin}}\ \frac{\|\boldsymbol{h}^j\|_2^2}{2}\|\boldsymbol{w}\|_2^2 - \langle \boldsymbol{M}_j\boldsymbol{h}^{j\top}, \boldsymbol{w}\rangle + \sigma\Big(\sum_{1\leq i\neq j\leq r}^{r} \frac{\lambda}{\sigma}\|\boldsymbol{w} - \boldsymbol{w}_i\|_2 + \frac{\gamma}{\sigma}\|\max\{-\boldsymbol{w}, \boldsymbol{0}\}\|_1\Big). \tag{9}$$

Since argmin $F =$ argmin αF for all $\alpha > 0$, we scale by $1/\sigma$ to get

$$\operatorname*{argmin}_{\boldsymbol{w}} \underbrace{\frac{\|\boldsymbol{h}^j\|_2^2}{2\sigma}\|\boldsymbol{w}\|_2^2 - \Big\langle \frac{\boldsymbol{M}_j \boldsymbol{h}^{j\top}}{\sigma}, \boldsymbol{w} \Big\rangle}_{\phi} + \sum_{1 \le i \neq j \le r}^{r} \frac{\lambda}{\sigma}\|\boldsymbol{w} - \boldsymbol{w}_i\|_2 + \frac{\gamma}{\sigma}\|\max\{-\boldsymbol{w}, \boldsymbol{0}\}\|_1, \tag{10}$$

which satisfies the assumptions required for proximal averaging.

The gradient of ϕ is $\nabla\phi(\boldsymbol{w}) = \|\boldsymbol{h}^j\|_2^2 \boldsymbol{w}/\sigma - \boldsymbol{M}_j \boldsymbol{h}^{j\top}/\sigma$ and it is $(\|\boldsymbol{h}^j\|_2^2/\sigma)$-Lipschitz. Thus, the gradient step in Algorithm 2 becomes

$$\overline{\boldsymbol{w}} \;=\; \boldsymbol{w} - \frac{1}{L}\nabla\phi(\boldsymbol{w}) \;=\; \boldsymbol{w} - \frac{1}{\|\boldsymbol{h}^j\|_2^2/\sigma}\Big(\frac{\|\boldsymbol{h}^j\|_2^2}{\sigma}\boldsymbol{w} - \frac{\boldsymbol{M}_j \boldsymbol{h}^{j\top}}{\sigma}\Big) \;=\; \frac{\boldsymbol{M}_j \boldsymbol{h}^{j\top}}{\|\boldsymbol{h}^j\|_2^2}.$$

Next we recall three useful lemmas for computing the prox of each nondifferentiable terms:

Lemma 5 (Scaling)**.** *If $\nu > 0, \mu > 0$ then $\mathrm{P}^{\mu}_{\nu\psi} = \mathrm{P}^{\nu\mu}_{\psi}$.*

Lemma 6. *The proximal operator of $\|\boldsymbol{x} - \boldsymbol{c}\|_2$ with parameter μ is*

$$\mathrm{P}^{\mu}_{\|\boldsymbol{x}-\boldsymbol{c}\|_2}(\boldsymbol{v}) = \boldsymbol{v} - \frac{\boldsymbol{v} - \boldsymbol{c}}{\max\Big\{1, \big\|\frac{\boldsymbol{v}-\boldsymbol{c}}{\mu}\big\|_2\Big\}}.$$

Lemma 7. *Let $\boldsymbol{1}$ be the vector of ones, the proximal operator of $\mu\|\max\{-\boldsymbol{x}, \boldsymbol{0}\}\|_1$ has the closed-form expression median$\big(\boldsymbol{v} + \mu\boldsymbol{1}, \boldsymbol{0}, \boldsymbol{v}\big)$, i.e.,*

$$\Big[\mathrm{P}^{1}_{\mu\|\max\{-\cdot,\boldsymbol{0}\}\|_1}(\boldsymbol{v})\Big]_i \;=\; \begin{cases} v_i + \mu & v_i + \mu < 0, \\ 0 & v_i \le 0 \le v_i + \mu, \\ v_i & v_i > 0. \end{cases}$$

Based on the three lemmas, the proximal step for the SON terms is

$$\mathrm{P}^{\frac{1}{L_j}\frac{\lambda}{\sigma}}_{\|\cdot - \boldsymbol{w}_i\|_2}(\bar{\boldsymbol{w}}) \;=\; \mathrm{P}^{\frac{\lambda}{\|\boldsymbol{h}^j\|_2^2}}_{\|\cdot - \boldsymbol{w}_i\|_2}(\bar{\boldsymbol{w}}) \;=\; \bar{\boldsymbol{w}} - \frac{\bar{\boldsymbol{w}} - \boldsymbol{w}_i}{\max\Big\{1, \Big\|\frac{\bar{\boldsymbol{w}} - \boldsymbol{w}_i}{\lambda/\|\boldsymbol{h}^j\|_2^2}\Big\|_2\Big\}},$$

and the proximal step for the penalty term is

$$\mathrm{P}^{1}_{\frac{1}{L_j}\frac{\gamma}{\sigma}\|\max\{-\cdot,\boldsymbol{0}\}\|_1}(\bar{\boldsymbol{w}}) \;=\; \text{median}\Big(\bar{\boldsymbol{w}} + \frac{1}{L_j}\frac{\gamma}{\sigma}\boldsymbol{1}, \boldsymbol{0}, \bar{\boldsymbol{w}}\Big) \;=\; \text{median}\Big(\bar{\boldsymbol{w}} + \frac{\gamma}{\|\boldsymbol{h}^j\|_2^2}\boldsymbol{1}, \boldsymbol{0}, \bar{\boldsymbol{w}}\Big).$$

Algorithm 3 performs one proximal-average iteration for update$(\boldsymbol{W}_k; \boldsymbol{H}_{k+1})$ in the BCD framework. Repeating these iterations solve the W-subproblem (5). The per-iteration cost of the for-loop in Algorithm 3 is $\mathcal{O}(r^2 m)$, or $\mathcal{O}(m^3)$ if $r \approx m$.

Remark (Why not enforce $\boldsymbol{W} \ge \boldsymbol{0}$ as hard constraints?)**.** *In NMF, nonnegativity is often imposed via an indicator function $\iota_+(\boldsymbol{W})$, where $\iota_+(W_{ij}) = 0$ if $W_{ij} \ge 0$ and $\iota_+(W_{ij}) = +\infty$ otherwise. For SON-NMF, enforcing $\boldsymbol{W} \ge \boldsymbol{0}$ as a hard constraint may cause the proximal-average update to produce infeasible $\boldsymbol{W}$, resulting in the objective jumping to $+\infty$, and destroys the convergence of the whole method.*

Algorithm 3: A iteration of update$(\boldsymbol{W}_k; \boldsymbol{H}_{k+1}, \boldsymbol{M}, \lambda, \gamma)$

```
for $j = 1, 2, ..., r$ do
    Compute $\|\boldsymbol{h}^j\|_2^2$, $\boldsymbol{M}_j = \boldsymbol{M} - \boldsymbol{W}\boldsymbol{H} + \boldsymbol{w}_j\boldsymbol{h}^j$
    Update $\boldsymbol{w}_j$ by solving (6) using one iteration of proximal-average as:
        Compute $\bar{\boldsymbol{w}} = \boldsymbol{M}_j\boldsymbol{h}^{j\top}/\|\boldsymbol{h}^j\|_2^2$
        For $i \neq j$, compute
        $\mathrm{P}^{\frac{1}{L_j}\frac{\lambda}{\sigma}}_{\|\cdot-\boldsymbol{w}_i\|_2}(\bar{\boldsymbol{w}}) = \mathrm{P}^{\frac{\lambda}{\|\boldsymbol{h}^j\|_2^2}}_{\|\cdot-\boldsymbol{w}_i\|_2}(\bar{\boldsymbol{w}}) = \bar{\boldsymbol{w}} - \dfrac{\bar{\boldsymbol{w}} - \boldsymbol{w}_i}{\max\left\{1, \left\|\frac{\bar{\boldsymbol{w}}-\boldsymbol{w}_i}{\lambda/\|\boldsymbol{h}^j\|_2^2}\right\|_2\right\}}$
        Compute $\mathrm{P}^1_{\frac{1}{L_j}\frac{\gamma}{\sigma}\|\max\{-\cdot,\boldsymbol{0}\}\|_1}(\bar{\boldsymbol{w}}) = \mathrm{median}\Big(\bar{\boldsymbol{w}} + \frac{\gamma}{\|\boldsymbol{h}^j\|_2^2}\boldsymbol{1}, \boldsymbol{0}, \bar{\boldsymbol{w}}\Big)$
        $\boldsymbol{w} = \sum_{i\neq j}^{r} \frac{\lambda}{\sigma}\mathrm{P}^{\frac{1}{L_j}\frac{\lambda}{\sigma}}_{\|\cdot-\boldsymbol{w}_i\|_2}(\bar{\boldsymbol{w}}) + \frac{\gamma}{\sigma}\mathrm{P}^1_{\frac{1}{L_j}\frac{\gamma}{\sigma}\|\max\{-\cdot,\boldsymbol{0}\}\|_1}(\bar{\boldsymbol{w}})$
```

Post-processing to extract columns of $\boldsymbol{W}$ After minimizing the SON$_{2,1}$ norm with an overestimated rank, we select one representative column from each cluster to form the final rank-reduced matrix $\boldsymbol{W}$.

5 Experiment

In this section we present numerical results to demonstrate the effectiveness of our algorithm for solving SON-NMF, and showcase SON-NMF's ability to identify the rank without prior knowledge. In section 5.1, we evaluate SON-NMF's rank-revealing capability. In section 5.2, we compare the proposed algorithm's speed against ADMM and Nesterov's smoothing. All the experiments were conducted on a Apple MacBook Air[3] in Python[4].

5.1 SON-NMF identifies the rank without prior knowledge

We test SON-NMF on datasets with known true NMF rank r^*. The rank parameter r is intentionally overestimated ($r > r^*$) to demonstrate that SON-NMF can correctly recover r^*.

5.1 Synthetic data

We first use the synthetic dataset from Leplat et al. (2019): $\boldsymbol{Z} = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{bmatrix}$ with rank$(\boldsymbol{Z}) = 3 < 4 =$ rank$_+(\boldsymbol{Z})$.

[3] M2 chipset, 8 CPU cores, 8 GPU cores with a 3.5GHz CPU and 8 GB RAM

[4] Code is available: `https://github.com/waqasbinhamed/sonnmf`

Dataset generation Let $\boldsymbol{W}_{\text{true}} = \boldsymbol{Z}$. The ground truth $\boldsymbol{H}_{\text{true}}$ is generated by sampling each column from a Dirichlet distribution with parameter $\alpha = 0.05$. The data matrix is then $\boldsymbol{M} = \boldsymbol{W}_{\text{true}}\boldsymbol{H}_{\text{true}} + \boldsymbol{N}$ where $\boldsymbol{N} \sim \mathcal{N}(0, 1)$ is Gaussian noise.

Experiment We solve (SON-NMF) using inexact-BCD (Algorithm 1) with proximal average (Algorithm 3) with:

- Random initialization of $\boldsymbol{W}$ and $\boldsymbol{H}$ in $[0, 1)$.
- 1 update of $\boldsymbol{H}$ per 10 updates of $\boldsymbol{W}$.
- Stopping criterion: relative change $(F_k - F_{k-1})/F_{k-1} < 10^{-6}$ or maximum iterations reached.
- Table 1 shows the parameters used in the experiments.

Table 1: Parameters used in the algorithm in the experiments

	r	λ	γ	max iteration
synthetic data experiment 1	4	10^{-6}	10	1000
synthetic data experiment 2	8	10^{-6}	1.5	1000
swimmer	50	0.5	10	1000
Jasper experiment 1	64	40000	10000	2000
Jasper experiment 2	100	1000	0.001	1000
Jasper experiment 3	20	1000000	1000000	1000
Urban	20	1000000	1000000	1000

Result Fig. 1 shows that SON-NMF reconstructs the data more accurately than standard NMF. Fig. 2 compares the convergence of W-subproblem updates using BCD with proximal averaging, BCD with ADMM, and BCD with Nesterov's smoothing, showing faster convergence for the proposed method.

5.2 The swimmer dataset

We next use the `swimmer` dataset[5] introduced by Donoho and Stodden (2003). It consists of 256 images of size 20×11 pixels representing a skeleton "swimming" (top row of Fig. 3). By inspection, the true NMF rank is $r^* = 17$: 1 for the torso and 16 for the four limbs (4 movements per limb). We apply SON-NMF with an overestimated rank $r = 50 > r^*$. All 17 true components are successfully recovered, while the extra components capture small-energy noise. Using a simple greedy search to select columns of $\boldsymbol{W}$, the score plot (right of Fig. 2) shows a clear cut-off at $r = 17$, accurately identifying the true rank.

[5] `https://gitlab.com/ngillis/nmfbook/`

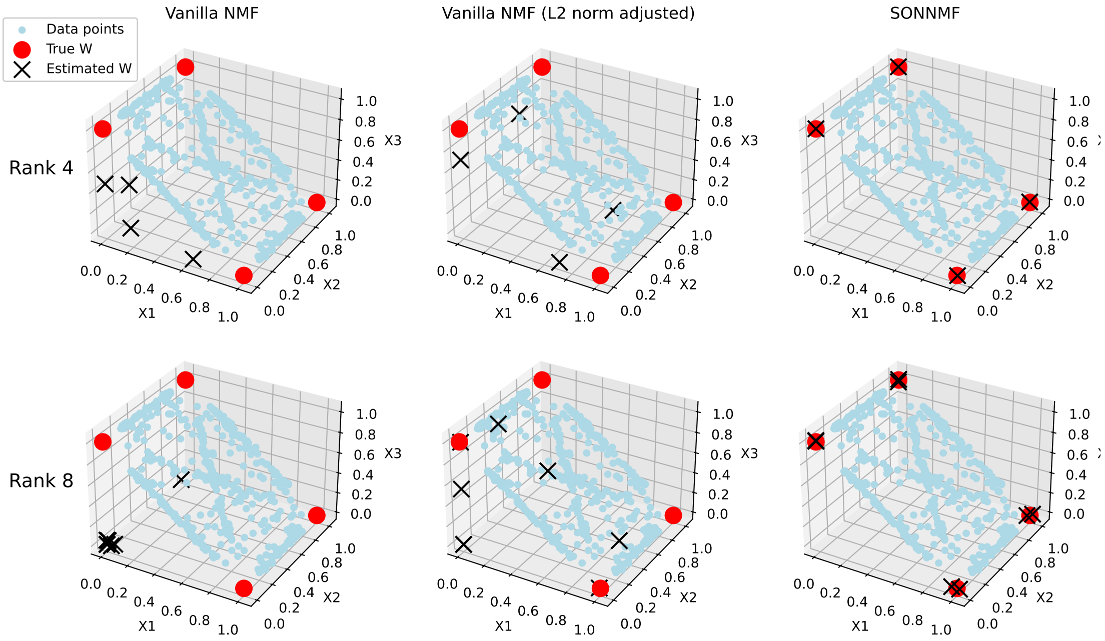


Figure 1: Reconstructed columns of $\boldsymbol{W}$ (crosses) by NMF and SON-NMF, compared with the ground truth columns (red dots). **Left**: $\boldsymbol{W}$ from NMF; **Middle**: NMF with columns normalized to 1; **Right**: $\boldsymbol{W}$ from SON-NMF. In both cases $r = 4$ and $r = 8$, the crosses given by SON-NMF fit numerically with the red dots.

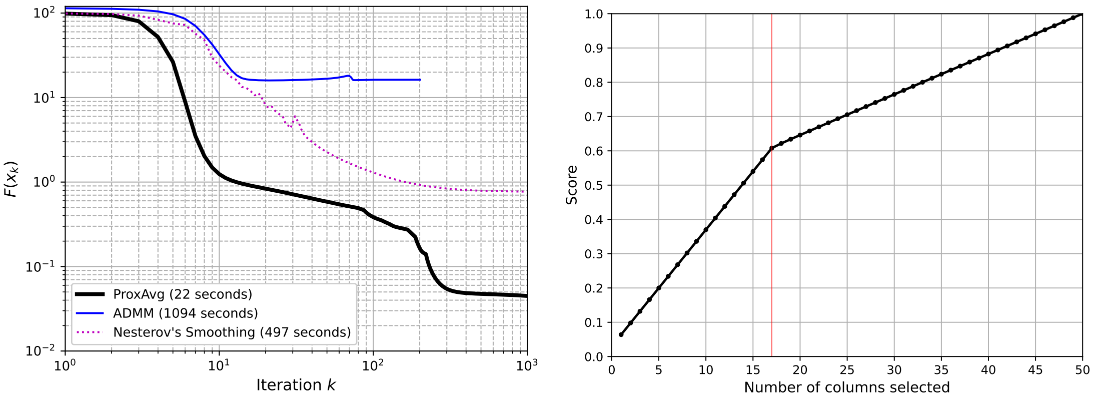


Figure 2: **Left**: Convergence of the SON-NMF cost function on synthetic data (Experiment 2). We compare three BCD algorithms for solving the W-subproblem: proximal average (this work), ADMM, and Nesterov's smoothing. Computation time (in seconds) is also shown. Proximal average achieves the fastest convergence. **Right**: Column-selection score (SON term) on the swimmer dataset using a simple greedy search. The red line at $r = 17$ marks the cut-off, matching the true number of components in the dataset.

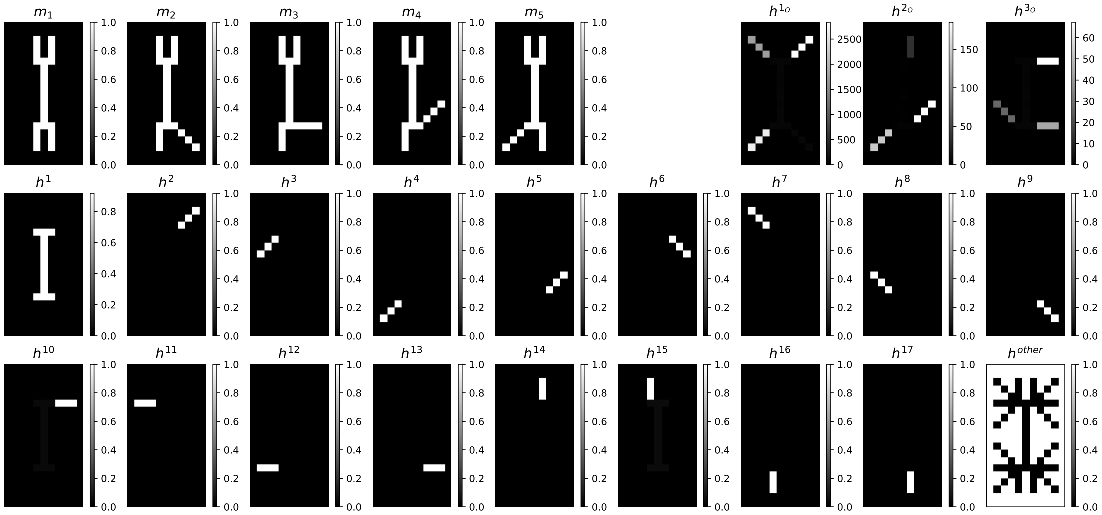


Figure 3: Top row (left): first 5 images ($\boldsymbol{m}^1, \boldsymbol{m}^2, \ldots, \boldsymbol{m}^5$) in the swimmer dataset, showing a swimmer swimmer in motion. Top row (right): 3 $\boldsymbol{h}_{\mathrm{O}}^j$ obtained from rank-50 vanilla NMF (subscript O denotes the vanilla NMF). These components exhibit mixed or overlapping factors. Bottom rows: Decomposition from rank-50 SON-NMF. Here $\boldsymbol{h}^1$ captures the torso, $\boldsymbol{h}^2, \boldsymbol{h}^3, \ldots, \boldsymbol{h}^{17}$ capture the four limbs. The remaining components are aggregated into $\boldsymbol{h}^{\text{other}}$, which represents noise. $\boldsymbol{h}^{\text{other}}$ is complementary to all $\boldsymbol{h}^1, \ldots, \boldsymbol{h}^{17}$, and its corresponding $\boldsymbol{w}$ has negligible energy (not plotted).

5.3 Jasper ridge hyperspectral dataset

We next evaluate SON-NMF on the Jasper Ridge hyperspectral dataset[6] This dataset has dimensions $100 \times 100 \times 198$, corresponding to 100×100 spatial pixels and 198 spectral bands (wavelength channels). Background on applying NMF to hyperspectral images can be found in (Gillis, 2020, Sect. 1.3.2). Fig. 4 shows a photograph of the Jasper Ridge site along with the three spatial regions selected for our experiments. Because the dataset entries have relatively large numerical values, we scale the SON regularization parameter λ to a higher magnitude (see Table 1).

Jasper experiment 1 We apply rank-64 SON-NMF to an 8×8 region containing vegetation and soil. Here we set $r = 64 = mn$, i.e., the rank is as large as the size of the dataset. Fig. 4 shows the matrix $\boldsymbol{W}$ obtained from SON-NMF. By inspection, region 1 consists of two end-member materials: soil and vegetation. SON-NMF successfully identifies these two materials, confirming its ability to recover the correct number of components without prior knowledge of the true rank.

Jasper experiment 2 We apply rank-100 SON-NMF to a 10×10 region consisting purely of water. Since this region contains only one material, the correct decomposition rank is 1. SON-NMF successfully recovers the water component, effectively reducing a rank-100 initialization to a rank-1 solution.

[6] Available in MATLAB: https://uk.mathworks.com/help/images/explore-hyperspectral-data-in-

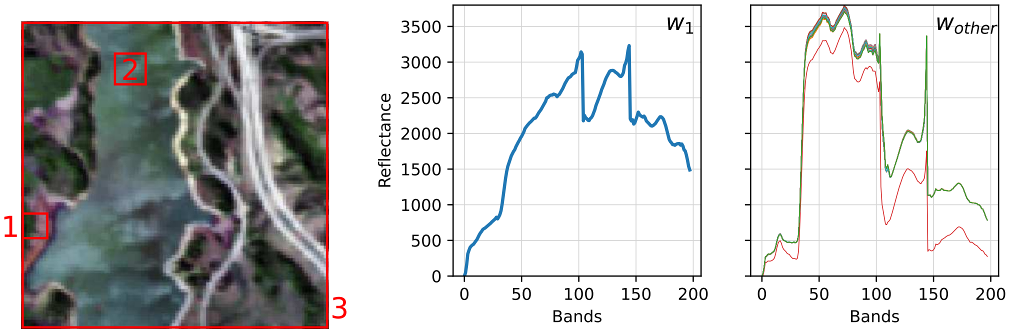


Figure 4: **Left**: Photograph of the Jasper Ridge site with the three selected regions highlighted in red. **Right**: Result for Jasper experiment 1. SON-NMF separates two distinct materials: soil (captured by $\boldsymbol{w}_1$) and vegetation (captured collectively by $\boldsymbol{w}_{\text{other}}$, i.e., all remaining columns of $\boldsymbol{W}$).

We remark that in this special case, the rank-1 factorization can also be obtained algebraically. By the Perron–Frobenius theorem, the leading component in the eigendecomposition of the covariance matrix yields the exact rank-1 NMF solution (see Proposition 1).

Proposition 1. *Given a data matrix $\boldsymbol{M} \in \mathbb{R}_+^{m\times n}$ with NMF $\boldsymbol{M} = \boldsymbol{W}\boldsymbol{H}$, and columns of $\boldsymbol{W}$ ordered according to $\|\boldsymbol{w}_j \boldsymbol{h}^j\|$, then for $r = 1$ (rank-1 NMF), the leading column $\boldsymbol{w}_1$ of $\boldsymbol{W}$ is given by the leading eigenvector of $\boldsymbol{M}\boldsymbol{M}^\top$.*

Proof. $\boldsymbol{M}\boldsymbol{M}^\top = \boldsymbol{W}\boldsymbol{H}\boldsymbol{H}^\top\boldsymbol{W}^\top = \boldsymbol{W}\boldsymbol{G}\boldsymbol{W}^\top$ with $\boldsymbol{G} := \boldsymbol{H}\boldsymbol{H}^\top$. Let $\boldsymbol{G} = \boldsymbol{V}\boldsymbol{\Sigma}\boldsymbol{V}^\top$ and $\boldsymbol{M}\boldsymbol{M}^\top = \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^\top$ be the eigendecompositions. Then $\boldsymbol{W}\boldsymbol{V}\boldsymbol{\Sigma}\boldsymbol{V}^\top\boldsymbol{W}^\top = \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^\top$ implies

$$\boldsymbol{W}\boldsymbol{V} = \boldsymbol{U} \implies (\boldsymbol{W}\boldsymbol{V})_{:,1} = \boldsymbol{U}_{:,1} \iff \boldsymbol{W}\boldsymbol{v}_1 = \boldsymbol{u}_1.$$

Both $\boldsymbol{G} = \boldsymbol{H}\boldsymbol{H}^\top$ and $\boldsymbol{M}\boldsymbol{M}^\top$ are nonnegative square matrices. By the Perron–Frobenius theorem, both $\boldsymbol{u}_1$ and $\boldsymbol{v}_1$ are nonnegative. Thus $\boldsymbol{u}_1 \in \text{cone}(\boldsymbol{W})$, and for $\text{rank}(\boldsymbol{W}) = 1$, we have $\boldsymbol{u}_1 = \boldsymbol{w}_1$. □

Fig. 5 shows that SON-NMF recovers the water spectrum with a relative error of 0.006, matching the exact eigendecomposition solution.

Jasper experiment 3 n this experiment, we run a rank-20 SON-NMF on the full Jasper Ridge dataset. SON-NMF extracts four distinct materials, as shown in Fig. 6. The extracted materials correspond to water, vegetation, soil, and road, and agree well with results obtained from other hyperspectral unmixing methods.

5.4 The Urban hyperspectral dataset

In this section, we conduct an experiment on a large-scale dataset with approximately 1.5×10^7 data points. We use the Urban dataset[7], which is a $307 \times 307 \times 162$ data

[7] Available at https://gitlab.com/ngillis/nmfbook/

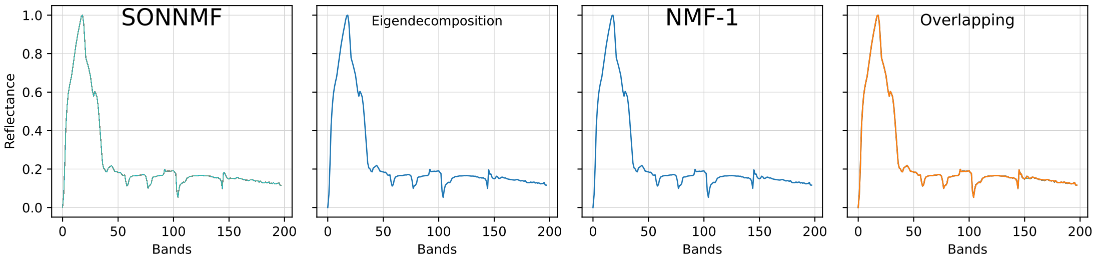


Figure 5: Result for Jasper experiment 2. The rank-100 SON-NMF (with $r = 100$, much larger than the true rank $r^* = 1$) identifies the water spectrum. **Left:** All 100 columns of $\boldsymbol{W}$ from SON-NMF share the same waveform. **Middle:** $\boldsymbol{W}$ obtained from eigendecomposition and rank-1 vanilla NMF. **Right:** Overlapping plot of all $\boldsymbol{W}$ columns, showing SON-NMF agrees with the vanilla NMF solution. All columns are normalized to unit ℓ_∞-norm for clarity.

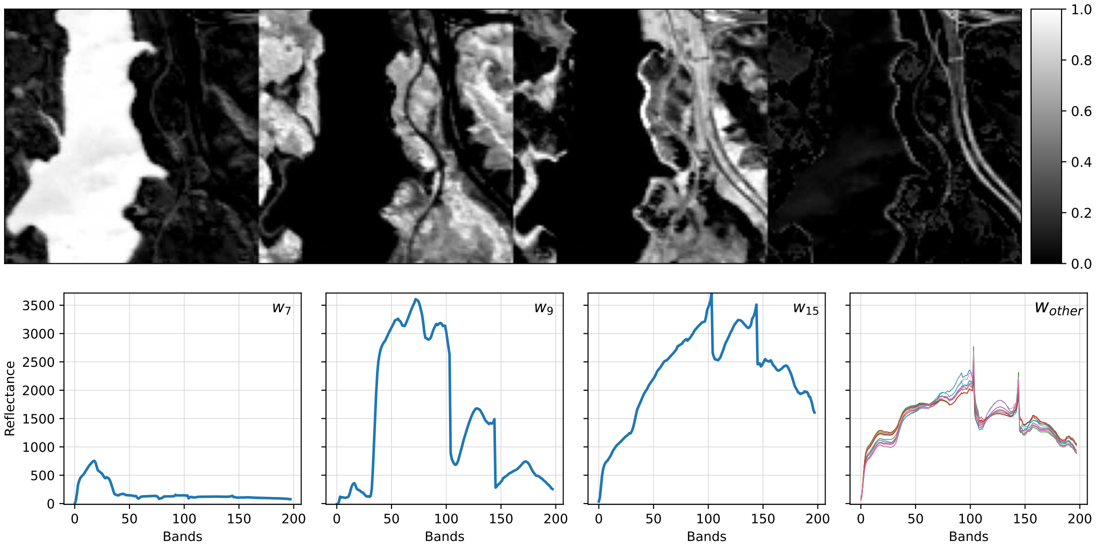


Figure 6: Result for Jasper experiment 3. Four material are extracted: (from left to right) water, vegetation, soil and road.

cube, with pixel dimensions 307×307 and 162 spectral bands. We run a rank-20 SON-NMF on this dataset with parameters $\lambda = \gamma = 10^6$, allowing up to 1000 iterations. SON-NMF successfully identifies five distinct material clusters, see Fig.7.

5.2 Speed of the algorithm

Fig. 2 shows the convergence of BCD (Algorithm 1) using proximal average to solve the W-subproblem (Algorithm 3), compared with BCD with ADMM and BCD with Nesterov's smoothing. The results clearly indicate that proximal average outperforms the other methods. For a detailed discussion on why proximal average performs better than smoothing, see Yu (2013).

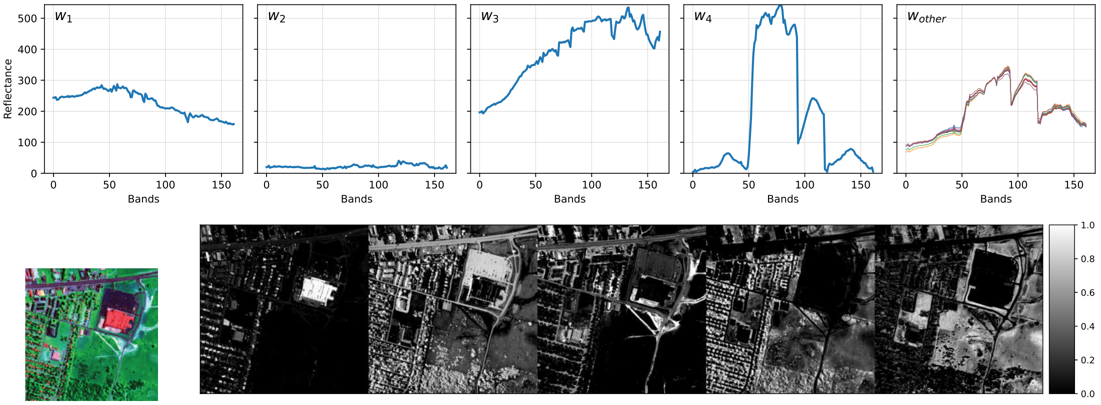


Figure 7: SON-NMF decomposition of the Urban hyperspectral dataset using rank-20. **Top:** Overlapped spectral signatures (columns of $\boldsymbol{W}$) of the five extracted materials: roof, asphalt, soil, tree, and grass (from left to right). **Bottom left:** The true photo of the Urban dataset. **Bottom right:** Summed abundances ($\boldsymbol{H}$) of the extracted materials across the spatial domain. Note that the weak component, asphalt, is successfully extracted by SON-NMF, which is not the case with classical NMF or other rank estimation approaches.

ADMM is not suitable for SON-NMF: high per-iteration cost The subproblem (7) of size $n \times 1$ can be solved using multi-block ADMM, which introduces N auxiliary variables and N Lagrange multipliers. The resulting augmented Lagrangian has size $n \times (1 + 2N)$, leading to a large computational overhead. Specifically, the mapping $\boldsymbol{W} \mapsto P(\boldsymbol{W})$ is $m \times r$ to $m \times r(r-1)/2$ due to the numerous nonsmooth terms $\|\boldsymbol{w}_i - \boldsymbol{w}_j\|_2$ in the SON regularization. For each column $\boldsymbol{w}_i$, the number of nonsmooth terms is r, making the per-iteration complexity for multi-block ADMM $\mathcal{O}(2mr^2 + mr)$. In contrast, proximal average has per-iteration complexity $\mathcal{O}(mr)$. When $r \sim m$, this means ADMM requires $\mathcal{O}(2m^3 + m^2)$ per iteration, whereas proximal average only requires $\mathcal{O}(m^2)$. Combined with the generally slower convergence of ADMM, this makes it inefficient for solving the W-subproblem in SON-NMF.

5.3 Discussion: favourable features of SON-NMF for applications

We summarize the key advantages of SON-NMF observed in our experiments.

Empirically rank-revealing and handles rank deficiency All seven experiments in section 5 demonstrate that SON-NMF can learn the factorization rank without prior knowledge. SON-NMF is effective on datasets with rank deficiency, a feature not present in other regularized NMF models such as minvol NMF Leplat et al. (2019), despite their empirical rank-revealing capabilities.

Detects weak components The clustering property of the SON term allows SON-NMF to identify weak components that vanilla NMF often misses. For example, in the Jasper dataset (section 5), the water component contributes only 9% of the total energy

$\|\boldsymbol{w}_{\text{water}}\boldsymbol{h}^{\text{water}}\|_F/\|\boldsymbol{M}\|_F$, compared with 54% for vegetation. The squared Frobenius norm in vanilla NMF emphasizes high-energy components, potentially ignoring weaker ones. In contrast, SON-NMF, through the term $\|\boldsymbol{w}_{\text{other}} - \boldsymbol{w}_{\text{water}}\|_2$, successfully extracts the water component. Lemma 1 further guarantees that the smallest possible cluster identified by the SON term has size at least 1.

Handles spectral variability The $\boldsymbol{W}$ in hyperspectral experiments (Figs. 4, 5, 6, 7) naturally capture spectral variability Borsoi et al. (2021). Thus, SON-NMF can effectively model such variability without requiring complex preprocessing pipelines.

Hierarchical clustering capability The number of clusters obtained by SON-NMF depends on the regularization parameter λ. Smaller λ values yield finer clusters, while larger values produce coarser clusters. This hierarchical nature is advantageous for datasets with multi-scale structure, as demonstrated in hyperspectral experiments, see Figs. 5.3.

6 Conclusion

In this paper, we proposed a sum-of-norm (SON) regularized NMF model, designed to estimate the factorization rank in NMF on-the-fly. The resulting SON-NMF problem is nonconvex, nonsmooth, non-separable, and non-proximal. To solve it, we developed a block coordinate descent (BCD) algorithm combined with proximal averaging.

Theoretically, we showed that the complexity of the SON term in SON-NMF is irreducible, implying that the computational cost of solving SON-NMF can be very high. This is expected, as estimating the rank in NMF is an NP-hard problem.

Empirically, we demonstrated that SON-NMF can accurately detect the correct factorization rank and extract weak components, making it particularly suitable for applications in imaging and hyperspectral data analysis. Its hierarchical clustering property and ability to handle spectral variability further highlight its practical advantages.

Acknowledgement

Andersen Ang thanks Steve Vavasis for the discussion on graph theory and the complexity of SON-NMF.

References

Ang, A. M. S. and Gillis, N. (2019). Algorithms and comparisons of nonnegative matrix factorizations with volume regularization for hyperspectral unmixing. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12(12):4843–4853.

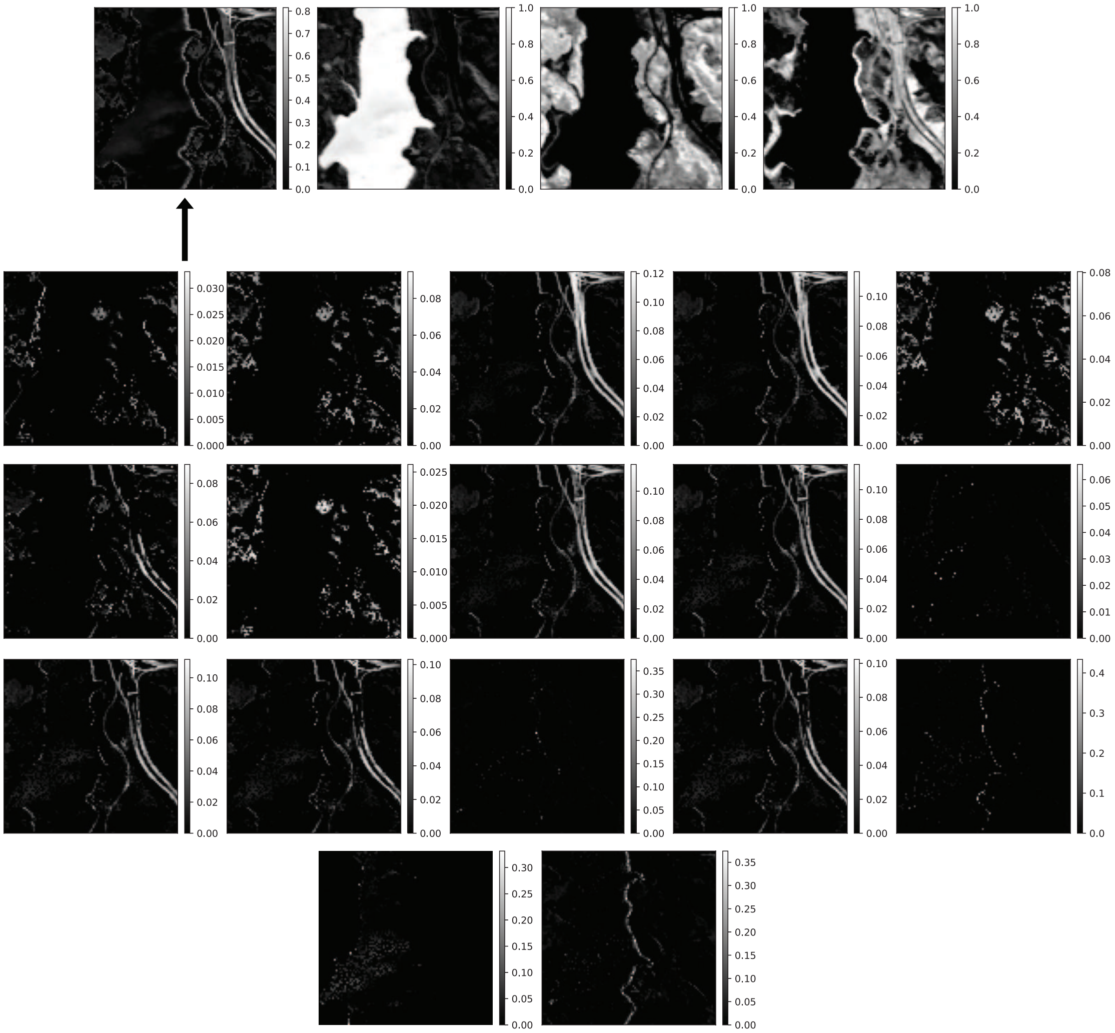


Figure 8: Full decomposition map of SON-NMF ($r = 20$) on Jasper dataset ($r^* \approx 4$). Here the road endmember consists of 17 components.

Ang, M. A. and Gillis, N. (2018). Volume regularized non-negative matrix factorizations. In *2018 9th Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS)*, pages 1–5. IEEE.

Ang, M. S. (2020). Nonnegative matrix and tensor factorizations: Models, algorithms and applications. *Ph. D. thesis*.

Bauschke, H. H., Goebel, R., Lucet, Y., and Wang, X. (2008). The proximal average: basic theory. *SIAM Journal on Optimization*, 19(2):766–785.

Beck, A. (2017). *First-order methods in optimization*. SIAM.

Berman, A. and Plemmons, R. J. (1994). *Nonnegative matrices in the mathematical sciences*. SIAM.

Bolte, J., Sabach, S., and Teboulle, M. (2014). Proximal alternating linearized minimization for nonconvex and nonsmooth problems. *Mathematical Programming*, 146(1-2):459–494.

Borsoi, R. A., Imbiriba, T., Bermudez, J. C. M., Richard, C., Chanussot, J., Drumetz, L., Tourneret, J.-Y., Zare, A., and Jutten, C. (2021). Spectral variability in hyperspectral data unmixing: A comprehensive review. *IEEE geoscience and remote sensing magazine*, 9(4):223–270.

Cai, D., He, X., Han, J., and Huang, T. S. (2010). Graph regularized nonnegative matrix factorization for data representation. *IEEE transactions on pattern analysis and machine intelligence*, 33(8):1548–1560.

Cantor, G. (1890). Ueber eine elementare frage der mannigfaltigketislehre. *Jahresbericht der Deutschen Mathematiker-Vereinigung*, 1:72–78.

Cohen, J. E. and Rothblum, U. G. (1993). Nonnegative ranks, decompositions, and factorizations of nonnegative matrices. *Linear Algebra and its Applications*, 190:149–168.

Combettes, P. L. and Wajs, V. R. (2005). Signal recovery by proximal forward-backward splitting. *Multiscale modeling & simulation*, 4(4):1168–1200.

Condat, L. (2016). Fast projection onto the simplex and the l1 ball. *Mathematical Programming*, 158(1-2):575–585.

Cotter, S. F., Rao, B. D., Engan, K., and Kreutz-Delgado, K. (2005). Sparse solutions to linear inverse problems with multiple measurement vectors. *IEEE Transactions on Signal Processing*, 53(7):2477–2488.

Dewez, J., Gillis, N., and Glineur, F. (2021). A geometric lower bound on the extension complexity of polytopes based on the f-vector. *Discrete Applied Mathematics*, 303:22–38.

Donoho, D. and Stodden, V. (2003). When does non-negative matrix factorization give a correct decomposition into parts? *Advances in neural information processing systems*, 16.

Esposito, F., Boccarelli, A., and Del Buono, N. (2020). An NMF-Based Methodology for Selecting Biomarkers in the Landscape of Genes of Heterogeneous Cancer-Associated Fibroblast Populations. *Bioinformatics and Biology Insights*, 14:1177932220906827.

Farrar, D. E. and Glauber, R. R. (1967). Multicollinearity in regression analysis: the problem revisited. *The Review of Economic and Statistics*, pages 92–107.

Fukushima, M. and Mine, H. (1981). A generalized proximal point algorithm for certain non-convex minimization problems. *International Journal of Systems Science*, 12(8):989–1000.

Gillis, N. (2020). *Nonnegative matrix factorization*. SIAM.

Hildreth, C. (1957). A quadratic programming procedure. *Naval research logistics quarterly*, 4(1):79–85.

Hocking, T. D., Joulin, A., Bach, F., and Vert, J.-P. (2011). Clusterpath: an algorithm for clustering using convex fusion penalties. In *28th international conference on machine learning*, page 1.

Huang, X., Ang, A., Huang, K., Zhang, J., and Wang, Y. (2025). Inhomogeneous graph trend filtering via a $l_{2,0}$-norm cardinality penalty. *IEEE Transactions on Signal and Information Processing over Networks*.

Jiang, T. and Vavasis, S. (2020). Certifying clusters from sum-of-norms clustering. *arXiv preprint arXiv:2006.11355*.

Kong, D., Ding, C., and Huang, H. (2011). Robust nonnegative matrix factorization using l21-norm. In *Proceedings of the 20th ACM international conference on Information and knowledge management*, pages 673–682.

Krarup, J. and Vajda, S. (1997). On torricelli's geometrical solution to a problem of fermat. *IMA Journal of Management Mathematics*, 8(3):215–224.

Le, H., Gillis, N., and Patrinos, P. (2020). Inertial block proximal methods for non-convex non-smooth optimization. In *International Conference on Machine Learning*, pages 5671–5681. PMLR.

Leplat, V., Ang, A. M., and Gillis, N. (2019). Minimum-volume rank-deficient nonnegative matrix factorizations. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3402–3406. IEEE.

Leplat, V., Gillis, N., and Ang, A. M. (2020). Blind audio source separation with minimum-volume beta-divergence nmf. *IEEE Transactions on Signal Processing*, 68:3400–3410.

Lindsten, F., Ohlsson, H., and Ljung, L. (2011). Clustering using sum-of-norms regularization: With application to particle filter output computation. In *2011 IEEE Statistical Signal Processing Workshop (SSP)*, pages 201–204. IEEE.

Nam, N. M., An, N. T., Rector, R. B., and Sun, J. (2014). Nonsmooth algorithms and nesterov's smoothing technique for generalized fermat–torricelli problems. *SIAM Journal on Optimization*, 24(4):1815–1839.

Nesterov, Y. (2003). *Introductory lectures on convex optimization: A basic course*, volume 87. Springer Science & Business Media.

Nesterov, Y. (2005). Smooth minimization of non-smooth functions. *Mathematical programming*, 103:127–152.

Nie, F., Huang, H., Cai, X., and Ding, C. (2010). Efficient and robust feature selection via joint $\ell 2$, 1-norms minimization. *Advances in neural information processing systems*, 23.

Niu, L., Zhou, R., Tian, Y., Qi, Z., and Zhang, P. (2016). Nonsmooth penalized clustering via ℓ_p regularized sparse regression. *IEEE transactions on cybernetics*, 47(6):1423–1433.

Paatero, P. and Tapper, U. (1994). Positive matrix factorization: A non-negative factor model with optimal utilization of error estimates of data values. *Environmetrics*, 5(2):111–126.

Passty, G. B. (1979). Ergodic convergence to a zero of the sum of monotone operators in hilbert space. *Journal of Mathematical Analysis and Applications*, 72(2):383–390.

Pelckmans, K., De Brabanter, J., Suykens, J. A., and De Moor, B. (2005). Convex clustering shrinkage. In *PASCAL workshop on statistics and optimization of clustering workshop*.

Razaviyayn, M., Hong, M., and Luo, Z.-Q. (2013). A unified convergence analysis of block successive minimization methods for nonsmooth optimization. *SIAM Journal on Optimization*, 23(2):1126–1153.

Schmidt, M., Roux, N., and Bach, F. (2011). Convergence rates of inexact proximal-gradient methods for convex optimization. *Advances in neural information processing systems*, 24.

Squires, S., Prügel-Bennett, A., and Niranjan, M. (2017). Rank selection in nonnegative matrix factorization using minimum description length. *Neural computation*, 29(8):2164–2176.

Tan, V. Y. and Févotte, C. (2012). Automatic relevance determination in nonnegative matrix factorization with the β-divergence. *IEEE transactions on pattern analysis and machine intelligence*, 35(7):1592–1605.

Tseng, P. and Yun, S. (2009). A coordinate gradient descent method for nonsmooth separable minimization. *Mathematical Programming*, 117:387–423.

Udell, M. and Townsend, A. (2019). Why are big data matrices approximately low rank? *SIAM Journal on Mathematics of Data Science*, 1(1):144–160.

Vavasis, S. A. (2010). On the complexity of nonnegative matrix factorization. *SIAM Journal on Optimization*, 20(3):1364–1377.

Wright, S. J. (2015). Coordinate descent algorithms. *Mathematical programming*, 151(1):3–34.

Xu, Y. and Yin, W. (2013). A block coordinate descent method for regularized multiconvex optimization with applications to nonnegative tensor factorization and completion. *SIAM Journal on imaging sciences*, 6(3):1758–1789.

Yu, Y.-L. (2013). Better approximation and faster algorithm using the proximal average. *Advances in neural information processing systems*, 26.

Yuan, Y., Sun, D., and Toh, K.-C. (2018). An efficient semismooth newton based algorithm for convex clustering. In *International Conference on Machine Learning*, pages 5718–5726. PMLR.