Memory-Efficient Separable Simplex-Structured Matrix Factorization via the Frank-Wolfe Method

Tri Nguyen

Qualifying Exam
Oregon State University

May 4, 2022

Outline

Simplex Structured Matrix Factorization

Simplex Structured Matrix Factorization (SSMF)

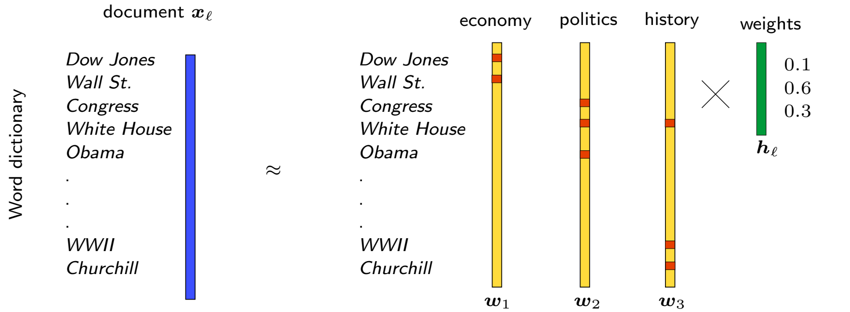
Data matrix $\boldsymbol{X} \in \mathbb{R}^{N \times M}$ is assumed to be generated by $\boldsymbol{W} \in \mathbb{R}^{N \times K}, \boldsymbol{H} \in \mathbb{R}^{K \times M}, K \ll \min(M, N)$ such that

$$\boldsymbol{X} = \boldsymbol{W}\boldsymbol{H} + \boldsymbol{V} \quad \text{subject to } \boldsymbol{H} \geq 0, \mathbf{1}^\top \boldsymbol{H} = \mathbf{1}^\top$$

Given $\boldsymbol{X}$, how do we find the latent factors $\boldsymbol{W}, \boldsymbol{H}$?

- Closely related to nonnegative matrix factorization.
- Has received significant attention across many domains [S. Arora et al. 2012; Sanjeev Arora et al. 2013; T.-H. Chan et al. 2008; X. Fu et al. 2016; Huang et al. 2019; Keshava et al. 2002; Mao et al. 2017b; Panov et al. 2017; Recht et al. 2012]

Application: Topic Modeling



$\boldsymbol{x}_\ell \approx 0.1 \times \boldsymbol{w}_1 + 0.6 \times \boldsymbol{w}_2 + 0.3 \times \boldsymbol{w}_3$

doc. $\approx 0.1 \times$ **economy** $+ 0.6 \times$ **politics** $+ 0.3 \times$ **history**

A demonstration of $\boldsymbol{x}_\ell \approx \boldsymbol{W}\boldsymbol{h}_\ell$

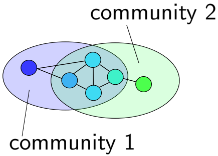
- $\boldsymbol{X}$ is a vocab-document matrix, then $\boldsymbol{X} = \boldsymbol{W}\boldsymbol{H}$ where
  - $\boldsymbol{H} \geq 0, \mathbf{1}^\top \boldsymbol{H} = \mathbf{1}^\top$
  - K is number of topics
- This model has been used in [S. Arora et al. 2012; Sanjeev Arora et al. 2013, 2016; Huang et al. 2016; Recht et al. 2012]

Application: Community Detection

- The mixed membership stochastic blockmodels [Airoldi et al. 2008]

$$P_{i,j} = \boldsymbol{h}_i^\top \boldsymbol{B} \boldsymbol{h}_j$$
$$\boldsymbol{A}(i,j) = \boldsymbol{A}(j,i) \sim \text{Bernoulli}(\boldsymbol{P}(i,j))$$

where $\boldsymbol{h}_i = [h_{1,i}, \ldots, h_{K,i}]^\top$ represents membership of node i, $\boldsymbol{B}$ represents community-community connection.



Demonstration of a graph with $K = 2$ communities

- By physical interpretation, $\boldsymbol{H} \geq 0, \mathbf{1}^\top \boldsymbol{H} = \mathbf{1}^\top$.
- Range space of $\boldsymbol{H}$ can be estimated from K leading eigenvectors of $\boldsymbol{A}$ (denoted as matrix $\boldsymbol{X}$). [Lei et al. 2015; Mao et al. 2017a,b; Panov et al. 2017]

$$\boldsymbol{X} = \boldsymbol{W}\boldsymbol{H} + \boldsymbol{N}$$

Identifiability

- Given a SSMF model with $\boldsymbol{X} = \boldsymbol{W}^\star \boldsymbol{H}^\star$, finding $\boldsymbol{W}^\star, \boldsymbol{H}^\star$ is a difficult problem.

$$
\begin{aligned}
&\text{find} && \boldsymbol{W}, \boldsymbol{H} && \text{(1a)}\\
&\text{subject to} && \boldsymbol{X} = \boldsymbol{W}\boldsymbol{H} && \text{(1b)}\\
& && \boldsymbol{H} \geq 0, \mathbf{1}^\top \boldsymbol{H} = \mathbf{1}^\top && \text{(1c)}
\end{aligned}
$$

- The solution is not unique. There exists non-singular $\boldsymbol{Q}$ such that

$$\boldsymbol{X} = \boldsymbol{W}^\star \boldsymbol{H}^\star = (\underbrace{\boldsymbol{W}^\star \boldsymbol{Q}^{-1}}_{\boldsymbol{W}'})(\underbrace{\boldsymbol{Q}\boldsymbol{H}^\star}_{\boldsymbol{H}'}), \text{ and } \boldsymbol{H}' \geq 0, \mathbf{1}^\top \boldsymbol{H}' = \mathbf{1}^\top$$

Definition (Identifiability [Xiao Fu et al. 2019])

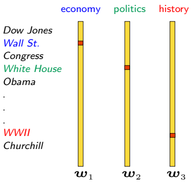
A SSMF model where $\boldsymbol{X} = \boldsymbol{W}^\star \boldsymbol{H}^\star$ is called identifiable respect to criterion (1) if for all $\boldsymbol{W}, \boldsymbol{H}$ satisfying criterion (1), it holds that $\boldsymbol{W} = \boldsymbol{W}^\star \boldsymbol{\Pi}, \boldsymbol{H} = \boldsymbol{\Pi}^\top \boldsymbol{H}^\star$, where $\boldsymbol{\Pi}$ is a permutation matrix.

Separability Condition

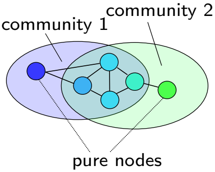
Separability condition [Donoho et al. 2003]

There exists set $\mathcal{K}$ so that $\boldsymbol{H}^\star(:,\mathcal{K}) = \boldsymbol{I}$.

- Have been adapted in many works [Sanjeev Arora et al. 2016; Tsung-Han Chan et al. 2011; Gillis et al. 2014a; Nascimento et al. 2005]
- Finding $\mathcal{K}$ is the key to estimate ground truth $\boldsymbol{W}^\star, \boldsymbol{H}^\star$.
  - In noiseless case, $\boldsymbol{X}(:,\mathcal{K}) = \boldsymbol{W}^\star \boldsymbol{H}^\star(:,\mathcal{K}) = \boldsymbol{W}^\star$.
- Physical interpretation
  - Anchor word [S. Arora et al. 2012] in topic modeling
  - Pure node [Mao et al. 2017b] in community detection



Demonstration of anchor word



Demonstration of pure node

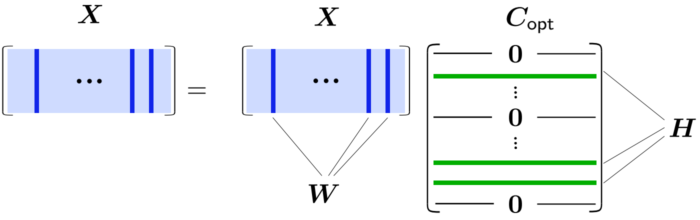
  - Expert annotator in crowd-sourcing [Ibrahim et al. 2019]
  - Pure pixels in hyperspectral unmixing [Ma et al. 2014]

A Self-Dictionary Perspective

- Consider the self-dictionary and sparse regression formulation,
  [Elhamifar et al. 2012; Esser et al. 2012; Iordache et al. 2014; Recht et al. 2012]

$$
\begin{aligned}
\underset{\boldsymbol{C}}{\text{minimize}} \quad & \|\boldsymbol{C}\|_{\text{row-0}} \\
\text{subject to} \quad & \boldsymbol{X} = \boldsymbol{X}\boldsymbol{C} \\
& \boldsymbol{C} \geq 0, \mathbf{1}^\top \boldsymbol{C} = \mathbf{1}^\top
\end{aligned}
$$

- $\boldsymbol{C}_{\text{opt}}(\mathcal{K},:) = \boldsymbol{H}, \boldsymbol{C}_{\text{opt}}(\mathcal{K}^c,:) = \mathbf{0}$ is an optimal solution point.
  - $\|\boldsymbol{C}_{\text{opt}}\|_{\text{row-0}} = K$.
  - For a full rank $\boldsymbol{W}$, one needs at least K non-zero rows of $\boldsymbol{C}$ to construct $\boldsymbol{X}$.



Row-sparsity matrix $\boldsymbol{C}_{\text{opt}}$

Greedy Approach

$$\begin{aligned} \underset{\boldsymbol{C}}{\text{minimize}} \quad & \|\boldsymbol{C}\|_{\text{row-0}} \\ \text{subject to} \quad & \boldsymbol{X} = \boldsymbol{X}\boldsymbol{C} \\ & \boldsymbol{C} \geq 0, \mathbf{1}^\top \boldsymbol{C} = \mathbf{1}^\top \end{aligned}$$

- The greedy approach identifies the set $\mathcal{K}$ by adding one index at a time [Xiao Fu et al. 2015b].
- Successive projection algorithm (SPA) [Araújo et al. 2001] is a representative.
- Extracting $\mathcal{K}$ is guaranteed even in noisy case [Gillis et al. 2014a].
- All greedy-based methods have a Gram-Schmidt structure which is prone to error propagation under noisy conditions.

Convex Relaxation Approach

- Relax the problem to a convex optimization problem [Ammanouil et al. 2014; Elhamifar et al. 2012; Gillis 2013; Gillis et al. 2018, 2014b; Recht et al. 2012]
- An example of this approach is [Esser et al. 2012; Xiao Fu et al. 2015a; Gillis et al. 2018]

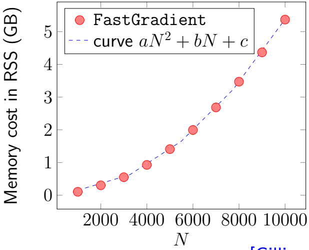
$$\begin{aligned} \underset{\boldsymbol{C}}{\text{minimize}} \quad & \frac{1}{2}\|\boldsymbol{X} - \boldsymbol{X}\boldsymbol{C}\|_{\mathrm{F}}^2 + \lambda R(\boldsymbol{C}) \\ \text{subject to} \quad & \boldsymbol{C} \geq 0, \mathbf{1}^\top \boldsymbol{C} = \mathbf{1}^\top \end{aligned}$$

  where $R(\boldsymbol{C})$ is some regularization term to promote row-sparsity.
- $\mathcal{K}$ is identified in noisy conditions.
- Often more robust than greedy approach.

Potential Memory Issue

The variable $\boldsymbol{C}$ has size $N \times N$.

A dense matrix $\boldsymbol{C}$ with $N = 100000$ requires 74.5GB.



Memory consumption of `FastGradient` [Gillis et al. 2018]

Proposal: Frank-Wolfe

In order to gain noise robustness and memory efficiency while obtaining identifiability,

- We follow the convex relaxation approach.
- We propose to use Frank-Wolfe as the optimization method to guarantee $O(KN)$ memory consumption.

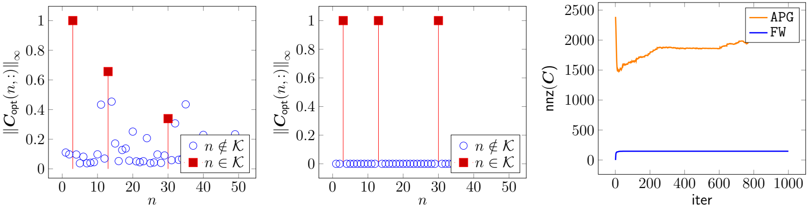
Warm-up with the Noiseless Case

$$\underset{\boldsymbol{C}}{\text{minimize}} \quad \frac{1}{2}\|\boldsymbol{X} - \boldsymbol{X}\boldsymbol{C}\|_{\mathrm{F}}^2 \tag{2a}$$

$$\text{subject to} \quad \boldsymbol{C} \geq 0, \mathbf{1}^\top \boldsymbol{C} = \mathbf{1}^\top \tag{2b}$$

Problem (2) can have several solutions

- A desired solution $\boldsymbol{C}^\star(\mathcal{K},:) = \boldsymbol{H}, \boldsymbol{C}^\star(\mathcal{K}^c,:) = \boldsymbol{0}$
- A trivial solution $\boldsymbol{I}_N$



APG - Objective value: $3.99e{-5}$ FW - Objective value: $2.86e{-5}$ Number of nonzeros (nnz) of $\boldsymbol{C}$

Accelerated proximal gradient (APG) vs Frank-Wolfe (FW). Unlike APG, FW outputs exact $\boldsymbol{C}^\star$ and keeps $\boldsymbol{C}$ sparse during its procedure. $M = 10, N = 50, K = 3$

Frank-Wolfe (FW) method [Frank et al. 1956]

- Assume $f(\boldsymbol{x})$ is convex and $\mathcal{D}$ is a compact convex constraint

$$\begin{aligned} \underset{\boldsymbol{x}}{\text{minimize}} \quad & f(\boldsymbol{x}) \\ \text{subject to} \quad & \boldsymbol{x} \in \mathcal{D} \end{aligned}$$

- FW's standard procedure: at iteration t,

$$\begin{aligned} \boldsymbol{s}^t &\leftarrow \underset{\boldsymbol{s} \in \mathcal{D}}{\arg\min} \ \nabla f(\boldsymbol{x}^t)^\top \boldsymbol{s} \\ \boldsymbol{x}^{t+1} &\leftarrow \boldsymbol{x}^t + \alpha^t(\boldsymbol{s}^t - \boldsymbol{x}^t), \quad \alpha^t = 2/(2+t) \end{aligned} \tag{3}$$

- For our problem,

When $\mathcal{D} = \left\{\boldsymbol{x} \in \mathbb{R}^n \mid \boldsymbol{x} \geq 0, \mathbf{1}^\top \boldsymbol{x} = 1\right\}$, solving (3) only cost $O(n)$, i.e.,

$$\boldsymbol{s} = \boldsymbol{e}_{n^\star}, \ n^\star = \arg\min_n [\nabla f(\boldsymbol{x}^t)]_n$$

FW in the Noiseless Case

- The original problem can be solved for each column $\boldsymbol{c}$ independently.

$$\begin{aligned} \underset{\boldsymbol{c}\in\mathbb{R}^N}{\text{minimize}} \quad & \frac{1}{2}\|\boldsymbol{x}-\boldsymbol{X}\boldsymbol{c}\|_{\mathrm{F}}^2 := f(\boldsymbol{c}) \\ \text{subject to} \quad & \boldsymbol{c}\geq 0, \mathbf{1}^\top\boldsymbol{c}=1 \end{aligned}$$

- Updating procedure:

$$\begin{aligned} \boldsymbol{s}^t &\leftarrow \boldsymbol{e}_{n^\star}, \quad n^\star = \arg\min_n\,[\nabla f(\boldsymbol{x}^t)] \\ \boldsymbol{c}^{t+1} &\leftarrow \boldsymbol{c}^t + \alpha^t(\boldsymbol{s}^t - \boldsymbol{c}^t), \quad \alpha^t = 2/(2+t) \end{aligned}$$

- If FW picks $n^\star\in\mathcal{K}$ in all iterations, then with $\boldsymbol{c}^0=\mathbf{0}$,

$$\mathsf{supp}(\boldsymbol{c}^t)\subseteq\mathcal{K}$$

holds in all iterations t until FW terminates.

FW in the Noiseless Case

FW always picks $n^\star \in \mathcal{K}$.

- Gradient

$$\nabla f(\mathbf{c}) = [\boldsymbol{h}_1^\top \boldsymbol{q}, \ldots, \boldsymbol{h}_N^\top \boldsymbol{q}]^\top, \quad \boldsymbol{q} = \boldsymbol{W}^\top \boldsymbol{W}(\boldsymbol{H}\boldsymbol{c} - \boldsymbol{h})$$

- For $n^\star = \arg\min_n \boldsymbol{h}_n^\top \boldsymbol{q}$, either
  - $\boldsymbol{h}_{n^\star} = \boldsymbol{e}_{k^\star}$, where $k^\star = \arg\min_{k \in [K]} q_k$. By definition, $n^\star \in \mathcal{K}$.
  - $\boldsymbol{q} = \mathbf{0} \Rightarrow$ desired solution $\boldsymbol{c}^\star$ is found because,

$$\boldsymbol{q} = \mathbf{0} \Leftrightarrow \boldsymbol{H}\boldsymbol{c} = \boldsymbol{h} \xLeftrightarrow{\text{assume } \mathcal{K} = [K]} \begin{bmatrix} \boldsymbol{I} & \boldsymbol{H}' \end{bmatrix} \boldsymbol{c} = \boldsymbol{h} \Leftrightarrow \boldsymbol{c} = \begin{bmatrix} \boldsymbol{h} \\ \mathbf{0} \end{bmatrix} = \boldsymbol{c}^\star$$

To sum up, in the noiseless case, with $\boldsymbol{c}^0 = \mathbf{0}$,

- $\text{supp}(\boldsymbol{c}^t) \subseteq \mathcal{K}$ for all t.
- FW terminates when $\boldsymbol{c}^t = \boldsymbol{c}^\star = \begin{bmatrix} \boldsymbol{h} \\ \mathbf{0} \end{bmatrix}$.

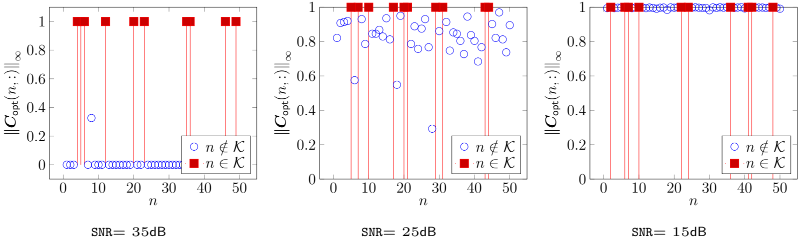
Therefore, FW outputs $\boldsymbol{C}_{\text{opt}} = \boldsymbol{C}^\star$ using only $O(KN)$ memory.

FW in the Noisy Case

- In the noisy case, i.e., $\boldsymbol{X} = \boldsymbol{W}\boldsymbol{H} + \boldsymbol{V}, \boldsymbol{V} \neq \boldsymbol{0}$, the gradient is

$$\nabla f(\boldsymbol{c}) = [\boldsymbol{h}_1^\top \boldsymbol{q}, \ldots, \boldsymbol{h}_n^\top \boldsymbol{q}] + \boldsymbol{n}, \quad (\boldsymbol{n} \text{ depends on the noise } \boldsymbol{V})$$

  then the picked index $n^\star$ could be outside of $\mathcal{K}$.
- FW is no longer guaranteed to output $\boldsymbol{C}^\star$.



$\boldsymbol{C}_{\text{opt}}$ obtained by FW; $M = 40, N = 50, K = 10$.

Enhancement in the Noisy Case

- Different regularizations have been used to promote row-sparsity [Elhamifar et al. 2012; Esser et al. 2012; Xiao Fu et al. 2015a; Gillis et al. 2018, 2014b; Recht et al. 2012]. For example, [Esser et al. 2012; Xiao Fu et al. 2015a] use

$$\|\boldsymbol{C}\|_{\infty,1} := \sum_{i=1}^{N} \|\boldsymbol{C}(i,:)\|_{\infty}$$

- FW works best with smooth functions

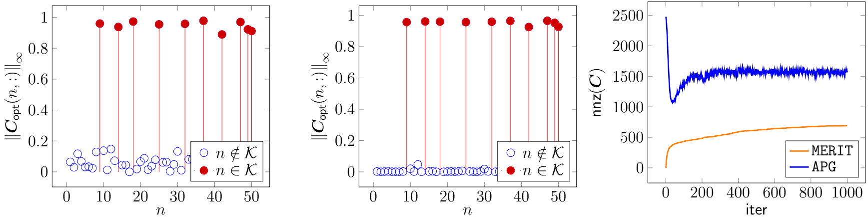
$$\varPhi_\mu(\boldsymbol{C}) = \sum_{i=1}^{N} \varphi_\mu(\boldsymbol{C}(i,:)), \quad \varphi_\mu(\boldsymbol{C}(i,:)) = \mu \log\left(\frac{1}{N}\sum_{j=1}^{N} \exp\left(\frac{c_{i,j}}{\mu}\right)\right)$$

- We propose `MERIT`, a FW-based algorithm for solving:

$$\begin{aligned} \underset{\boldsymbol{C}}{\text{minimize}} \quad & \frac{1}{2}\|\boldsymbol{X} - \boldsymbol{X}\boldsymbol{C}\|_{\mathrm{F}}^2 + \lambda \varPhi_\mu(\boldsymbol{C}) \\ \text{subject to} \quad & \boldsymbol{C} \geq 0, \boldsymbol{1}^\top \boldsymbol{C} = \boldsymbol{1}^\top \end{aligned}$$

Identifiability

- With regularization, we can guarantee the extraction of $\mathcal{K}$ exactly in the noisy case under some reasonable assumptions [Nguyen et al. 2021].
- This result is obtained using a similar idea to [Xiao Fu et al. 2015a].
- Any convex optimization method can be used to obtain $\mathcal{K}$ via solution $\boldsymbol{C}_{\text{opt}}$.



MERIT - Objective value: $3.58e{-}01$ APG - Objective value: $4.41e{-}01$ Number of nonzeros (nnz) of $\boldsymbol{C}$

$$M = 40, K = 10, N = 50, \texttt{SNR} = 30\text{dB}, \mu = 1e{-}6, \lambda = 0.1.$$

Memory

- The objective function

$$h(\boldsymbol{C}) = \underbrace{\frac{1}{2}\left\|\boldsymbol{X} - \boldsymbol{X}\boldsymbol{C}\right\|_{\mathrm{F}}^2}_{f(\boldsymbol{C})} + \lambda \mathit{\Phi}_\mu(\boldsymbol{C}) = f(\boldsymbol{C}) + \lambda \mathit{\Phi}_\mu(\boldsymbol{C})$$

- FW's updating procedure on this problem can be executed column by column sequentially

$$\boldsymbol{s}_\ell^t \leftarrow \boldsymbol{e}_{n^\star}, \quad n^\star = \arg\min_n \, [\nabla h(\boldsymbol{c}_\ell)]_n$$
$$\boldsymbol{c}_\ell^{t+1} \leftarrow \boldsymbol{c}_\ell^t + \alpha(\boldsymbol{s}_\ell^t - \boldsymbol{c}_\ell^t), \quad \alpha^t = 2/(2+t)$$

- Gradient is given by

$$\nabla h(\boldsymbol{c}_\ell) = \nabla f(\boldsymbol{c}_\ell) + \lambda [\nabla \mathit{\Phi}_\mu(\boldsymbol{C})]_{:,\ell}$$

- Question: If at iteration t, supp$(\boldsymbol{c}_\ell^t) \subseteq \mathcal{K}$, can FW pick

$$n^\star \in \mathcal{K},$$

where $n^\star := \arg\min_n \, [\nabla h(\boldsymbol{c}_\ell)]_n$ in iteration $t+1$?

Effect of Noise

- Gradient of $f(\boldsymbol{c}_\ell) = 1/2 \, \|\boldsymbol{X} - \boldsymbol{X}\boldsymbol{C}\|_{\mathrm{F}}^2$

$$\nabla f(\boldsymbol{c}_\ell) = [\boldsymbol{h}_1^\top \boldsymbol{q}_\ell, \ldots, \boldsymbol{h}_N^\top \boldsymbol{q}_\ell]^\top + \boldsymbol{n}_\ell$$

- A demonstration of effect of noise that causes

$$n^\star := \arg\min_n \, [\nabla f(\boldsymbol{c}_\ell)]_n \notin \mathcal{K}.$$

$$\nabla f(\boldsymbol{c}_\ell) = \begin{matrix} n^\star \dashrightarrow \\ \\ \\ \\ \\ \end{matrix} \begin{bmatrix} 0.5 \\ 1.5 \\ 1.5 \\ 2.0 \\ \vdots \\ 1.5 \end{bmatrix} \begin{matrix} \left.\begin{matrix} \\ \\ \end{matrix}\right\} \mathcal{K} \\ \\ \\ \end{matrix} + \begin{bmatrix} 0.5 \\ 1.5 \\ -0.5 \\ 1.0 \\ \vdots \\ -1.0 \end{bmatrix} = \begin{bmatrix} 1.0 \\ 3.0 \\ 1.0 \\ 3.0 \\ \vdots \\ 0.5 \end{bmatrix} \begin{matrix} \left.\begin{matrix} \\ \\ \end{matrix}\right\} \mathcal{K} \\ \\ \\ \dashleftarrow n^\star \end{matrix}$$

Regularization

- Gradient of the regularization

$$\boldsymbol{y}_\ell = [\nabla \Phi_\mu(\boldsymbol{C})]_{:,\ell}, \quad y_{n,\ell} = \frac{\exp(c_{n,\ell}/\mu)}{\sum_{i=1}^{N} \exp(c_{n,i}/\mu)}$$

- Assume that at iteration t, supp$(\boldsymbol{c}_\ell^t) \subseteq \mathcal{K}$ for all ℓ.
  - For $n \notin \mathcal{K}$, $y_{n,\ell} = 1/N$.
  - If $\exists n_0 \in \mathcal{K}$ such that $c_{n_0,\ell}$ is not the largest element in row n_0 (*), then $y_{n_0,\ell} < \exp((c_{n_0,\ell} - c_{n_0,\star})/\mu)$, $\quad c_{n_0,\star} = \max_i c_{n_0,i}$.
  - (*) can be enforced with some initialization.
- An example of $\boldsymbol{C}$ and $\boldsymbol{y}_\ell$,

$$\boldsymbol{C} = \begin{array}{ccc} & \begin{array}{ccc} \ell & \quad\quad & m \end{array} & \\ \begin{array}{c} \mathcal{K}\left\{\begin{array}{c} \\ \\ \\ \end{array}\right. \\ \\ \\ \end{array} & \begin{bmatrix} 0.3 & \dots & 0.1 \\ 0.2 & \dots & 0.6 \\ 0.5 & \dots & 0.3 \\ & & \\ \mathbf{0} & \vdots & \mathbf{0} \end{bmatrix} & \begin{array}{c} \\ \leftarrow n_0 \\ \\ \\ \\ \end{array} \end{array} \implies \boldsymbol{y}_\ell = \begin{bmatrix} 0.5 \\ 0.001 \\ 0.9 \\ 1/N \\ \vdots \\ 1/N \end{bmatrix} \begin{array}{c} \\ \leftarrow n_0 \\ \\ \\ \\ \\ \end{array}$$

Effect of Regularization

Regularization can ensure $n^\star \in \mathcal{K}$ under some reasonable assumptions.

- Gradient

$$\nabla h(\boldsymbol{c}_\ell) = \nabla f(\boldsymbol{c}_\ell) + \lambda \boldsymbol{y}_\ell,$$

- We have

$$\begin{cases} y_{n,\ell} = 1/N & \text{if } n \notin \mathcal{K} \\ y_{n_0,\ell} \approx 0 & \text{for some } n_0 \in \mathcal{K} \end{cases}$$

$$\Rightarrow n^\star := \arg\min_n \, [\nabla h(\boldsymbol{c}_\ell)]_n = n_0 \quad \text{for some } \lambda$$

- An example of $\boldsymbol{C}$ and $\nabla h(\boldsymbol{c}_\ell)$,

$$\Longrightarrow \nabla h(\boldsymbol{c}_\ell) = \begin{array}{c} \mathcal{K}\left\{\begin{array}{c} \\ \\ \\ \end{array}\right. \\ \\ \\ \\ \\ \end{array} \begin{bmatrix} 1.0 \\ 3.0 \\ 1.0 \\ 3.0 \\ \vdots \\ 0.5 \end{bmatrix} + \lambda \begin{bmatrix} 0.5 \\ 0.001 \\ 0.9 \\ 1/N \\ \vdots \\ 1/N \end{bmatrix}$$

the smallest element

MERIT in Noisy Case

To sum up, in the noisy case, under some reasonable assumptions, the proposed method MERIT can

- Extract $\mathcal{K}$ exactly.
- If $\boldsymbol{C}^t$ satisfies $\text{supp}(\boldsymbol{c}_\ell^t) \subseteq \mathcal{K}$ for all ℓ, then $\text{supp}(\boldsymbol{c}_\ell^{t+1}) \subseteq \mathcal{K}$ for all ℓ, and hence MERIT can guarantee a memory consumption of $O(KN)$.

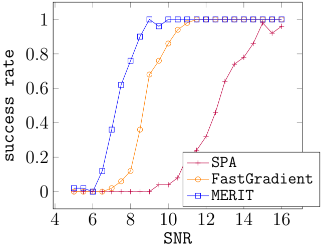
Synthetic Data

Data generation

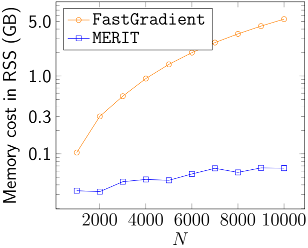
- $\boldsymbol{W} \sim \mathcal{U}(0,1)$
- $\boldsymbol{H} \sim \text{Dir}(\mathbf{1}), \boldsymbol{H}(:,1:K) = \boldsymbol{I}$
- $\boldsymbol{V} \sim \mathcal{N}(0,\sigma)$
- After shuffling $\boldsymbol{H}$, $\boldsymbol{X} = \boldsymbol{W}\boldsymbol{H} + \boldsymbol{V}$
- Noise level is measured in $\texttt{SNR} = 10\log_{10}(\sum_{\ell=1}^{N} \|\boldsymbol{W}\boldsymbol{h}_\ell\|_2^2)/(MN\sigma^2)\text{dB}$

Metric

- $\texttt{success rate} = P(\mathcal{K} = \widehat{\mathcal{K}})$
- Estimate `success rate` by 50 trials



(a) `success rate` under different SNRs; $N = 200$, $M = 50$, $K = 40$.



(b) Memory consumption under different N's; SNR $= 10$dB, $M = 50$, $K = 40$.

Performance of MERIT compared to baselines

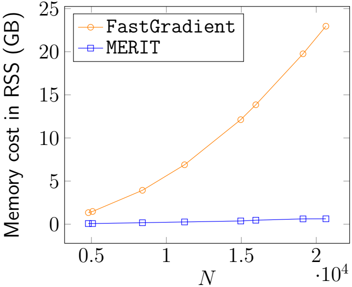
Real Data: Topic Modeling

Accuracy

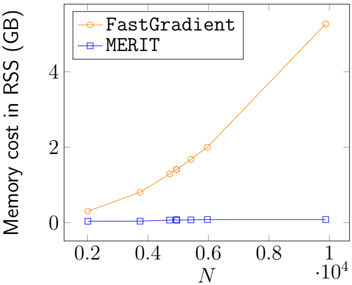
	Method \ K	3	4	5	6	7	8	9	10
TDT2	SPA	0.87	0.83	0.81	0.81	0.78	0.76	0.75	0.72
	FastAnchor	0.77	0.72	0.67	0.63	0.66	0.63	0.65	0.65
	XRAY	0.87	0.82	0.80	0.81	0.78	0.75	0.75	0.71
	LDA	0.78	0.77	0.74	0.75	0.73	0.72	0.68	0.70
	FastGradient	0.70	0.71	0.65	0.64	0.61	0.56	0.58	0.57
	MERIT	**0.88**	**0.88**	**0.85**	**0.86**	**0.84**	**0.82**	**0.80**	**0.77**
Reuters-21578	SPA	0.64	0.57	0.54	0.51	0.49	0.44	0.42	0.40
	FastAnchor	0.60	0.57	0.52	0.52	0.46	0.42	0.38	0.37
	XRAY	0.63	0.57	0.54	0.51	0.49	0.45	0.42	0.40
	LDA	0.63	0.57	0.53	0.51	0.46	0.44	0.41	0.42
	FastGradient	0.62	0.57	**0.56**	0.51	0.50	**0.48**	**0.44**	**0.46**
	MERIT	**0.66**	**0.62**	0.53	**0.53**	**0.51**	**0.48**	0.43	0.45

Bold, and blue indicate the best and second best scores, resp.

Real Data: Topic Modeling



TDT2



Reuters-21578

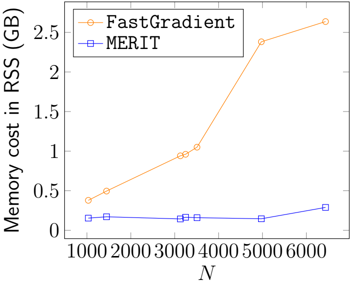
Memory consumption of `FastGradient` and `MERIT`

Real Data: Community detection

- Metric: Spearman's rank correlation (SRC). SRC $\in [-1, 1]$, higher value is better.
- Data: co-authorship networks, a community ground truth is defined by
  - DBLP: group of conferences
  - MAG: "field of study" tag

Dataset	GeoNMF	SPOC	FastGradient	MERIT
DBLP1	0.2974	0.2996	**0.3145**	0.2937
DBLP2	0.2948	0.2126	0.3237	**0.3257**
DBLP3	0.2629	**0.2972**	0.1933	0.2763
DBLP4	0.2661	0.3479	0.1601	**0.3559**
DBLP5	0.1977	0.1720	0.0912	**0.1983**
MAG1	**0.1349**	0.1173	0.0441	0.1149
MAG2	0.1451	0.1531	**0.2426**	0.2414

SRC Performance on DBLP and MAG. **Bold** and blue indicate the best and second best scores.



Memory consumption of FastGradient and MERIT

Conclusion

- FW is proposed as a memory efficient method for solving separable simplex-structured matrix factorization via convex relaxation.
- When noise is absent, using FW can bring identification with memory $O(KN)$
- For the noisy case, we have proposed using a smooth regularization to guarantee identifiability.
- For the noisy case, we have also shown that running FW only cost $O(KN)$ under some reasonable assumptions.

The talk is based on [Tri Nguyen et al. "Memory-efficient convex optimization for self-dictionary separable nonnegative matrix factorization: A frank-wolfe approach". In: *arXiv preprint arXiv:2109.11135* [2021], IEEE TSP, revised. (2nd round revision).]

Reference I

[1] Edoardo M Airoldi et al. "Mixed membership stochastic blockmodels". In: *Journal of Machine Learning Research* 9.Sep (2008), pp. 1981–2014.

[2] R. Ammanouil et al. "Blind and Fully Constrained Unmixing of Hyperspectral Images". In: *IEEE Trans. Image Process.* 23.12 (Dec. 2014), pp. 5510–5518.

[3] U.M.C. Araújo et al. "The successive projections algorithm for variable selection in spectroscopic multicomponent analysis". In: *Chemometrics and Intelligent Laboratory Systems* 57.2 (2001), pp. 65–73.

[4] S. Arora et al. "Learning topic models–going beyond SVD". In: *2012 IEEE 53rd Annual Symposium on Foundations of Computer Science*. 2012, pp. 1–10.

[5] Sanjeev Arora et al. "A practical algorithm for topic modeling with provable guarantees". In: *International Conference on Machine Learning*. 2013, pp. 280–288.

Reference II

[6] Sanjeev Arora et al. "Computing a Nonnegative Matrix Factorization—Provably". In: *SIAM Journal on Computing* 45.4 (2016), pp. 1582–1611. DOI: 10.1137/130913869.

[7] T.-H. Chan et al. "A Convex Analysis Framework for Blind Separation of Non-Negative Sources". In: *IEEE Trans. Signal Process.* 56.10 (Oct. 2008), pp. 5120–5134.

[8] Tsung-Han Chan et al. "A simplex volume maximization framework for hyperspectral endmember extraction". In: *IEEE Trans. Geosci. Remote Sens.* 49.11 (2011), pp. 4177–4193.

[9] D. Donoho et al. "When does non-negative matrix factorization give a correct decomposition into parts?" In: *Advances in Neural Information Processing Systems*. Vol. 16. 2003, pp. 1141–1148.

[10] E. Elhamifar et al. "See all by looking at a few: Sparse modeling for finding representative objects". In: *IEEE Conference on Computer Vision and Pattern Recognition*. 2012, pp. 1600–1607.

Reference III

[11] Ernie Esser et al. "A convex model for nonnegative matrix factorization and dimensionality reduction on physical space". In: *IEEE Trans. Image Process.* 21.7 (2012), pp. 3239–3252.

[12] Marguerite Frank et al. "An algorithm for quadratic programming". In: *Naval Research Logistics Quarterly* 3.1-2 (1956), pp. 95–110.

[13] X. Fu et al. "Robust Volume Minimization-Based Matrix Factorization for Remote Sensing and Document Clustering". In: *IEEE Trans. Signal Process.* 64.23 (Dec. 2016).

[14] Xiao Fu et al. "Nonnegative Matrix Factorization for Signal and Data Analytics: Identifiability, Algorithms, and Applications". In: *IEEE Signal Process. Mag.* 36.2 (Mar. 2019), pp. 59–80.

[15] Xiao Fu et al. "Robustness analysis of structured matrix factorization via self-dictionary mixed-norm optimization". In: *IEEE Signal Processing Letters* 23.1 (2015), pp. 60–64.

[16] Xiao Fu et al. "Self-Dictionary Sparse Regression for Hyperspectral Unmixing: Greedy Pursuit and Pure Pixel Search are Related". In: *IEEE J. Sel. Topics Signal Process.* 9.6 (2015), pp. 1128–1141.

Reference IV

[17] Nicolas Gillis. "Robustness analysis of hottopixx, a linear programming model for factoring nonnegative matrices". In: *SIAM Journal on Matrix Analysis and Applications* 34.3 (2013), pp. 1189–1212.

[18] Nicolas Gillis et al. "A fast gradient method for nonnegative sparse regression with self-dictionary". In: *IEEE Trans. Image Process.* 27.1 (2018), pp. 24–37.

[19] Nicolas Gillis et al. "Fast and robust recursive algorithms for separable nonnegative matrix factorization". In: *IEEE Trans. Pattern Anal. Mach. Intell.* 36.4 (2014), pp. 698–714.

[20] Nicolas Gillis et al. "Robust near-separable nonnegative matrix factorization using linear optimization". In: *The Journal of Machine Learning Research* 15.1 (2014), pp. 1249–1280.

[21] Kejun Huang et al. "Anchor-free correlated topic modeling: Identifiability and algorithm". In: *Advances in Neural Information Processing Systems*. 2016, pp. 1786–1794.

Reference V

[22] Kejun Huang et al. "Detecting Overlapping and Correlated Communities without Pure Nodes: Identifiability and Algorithm". In: *International Conference on Machine Learning*. Sept. 2019, pp. 2859–2868.

[23] Shahana Ibrahim et al. "Crowdsourcing via Pairwise Co-occurrences: Identifiability and Algorithms". In: *Advances in Neural Information Processing Systems*. 2019, pp. 7847–7857.

[24] Marian Daniel Iordache et al. "Collaborative sparse regression for hyperspectral unmixing". In: *IEEE Trans. Geosci. Remote Sens.* 52.1 (2014), pp. 341–354.

[25] Nirmal Keshava et al. "Spectral unmixing". In: *IEEE signal processing magazine* 19.1 (2002), pp. 44–57.

[26] Jing Lei et al. "Consistency of spectral clustering in stochastic block models". In: *The Annals of Statistics* 43.1 (2015), pp. 215–237.

Reference VI

[27] Wing-Kin Ma et al. "A signal processing perspective on hyperspectral unmixing: Insights from remote sensing". In: *IEEE Signal Process. Mag.* 31.1 (2014), pp. 67–81.

[28] Xueyu Mao et al. "Estimating Mixed Memberships with Sharp Eigenvector Deviations". In: *arXiv* (2017), pp. 1–46. ISSN: 23318422. arXiv: 1709.00407.

[29] Xueyu Mao et al. "On Mixed Memberships and Symmetric Nonnegative Matrix Factorizations". In: *International Conference on Machine Learning*. 2017, pp. 2324–2333.

[30] José MP Nascimento et al. "Vertex component analysis: A fast algorithm to unmix hyperspectral data". In: *IEEE Trans. Geosci. Remote Sens.* 43.4 (2005), pp. 898–910.

[31] Tri Nguyen et al. "Memory-efficient convex optimization for self-dictionary separable nonnegative matrix factorization: A frank-wolfe approach". In: *arXiv preprint arXiv:2109.11135* (2021).

Reference VII

[32] Maxim Panov et al. "Consistent estimation of mixed memberships with successive projections". In: *International Workshop on Complex Networks and their Applications* (2017), pp. 53–64.

[33] Ben Recht et al. "Factoring nonnegative matrices with linear programs". In: *Advances in Neural Information Processing Systems*. 2012, pp. 1214–1222.

Condition (*)

Claim:

$\exists n_0 \in \mathcal{K}$ such that $\boldsymbol{C}(n_0,:)$ is not a constant
$\Rightarrow \exists n_0 \in \mathcal{K}$ such that $c_{n_0,\ell}$ is not the largest element in row n_0 (*).

- Assume that for all $n \in \mathcal{K}$, $c_{n,\ell}$ is the largest element in row n.
- Then for row n_0 such that $\boldsymbol{C}(n_0,:)$ is not a constant,

$$\exists m, \quad c_{n_0,\ell} > c_{n_0,m}$$

- That leads to

$$1 = \mathbf{1}^\top \boldsymbol{c}_\ell > \mathbf{1}^\top \boldsymbol{c}_m = 1$$

- The contradiction concludes our claim.

An example of $\boldsymbol{C}$,

$$\boldsymbol{C} = \begin{array}{cc} & \begin{array}{ccc} \overbrace{\ell} & \quad\quad & \overbrace{m} \end{array} \\ \mathcal{K}\left\{ \begin{array}{c} \\ \\ \\ \end{array} \right. & \begin{bmatrix} \vdots & \cdots & \vdots \\ 0.6 & \cdots & 0.5 \\ \vdots & \cdots & \vdots \\ & & \\ \mathbf{0} & \vdots & \mathbf{0} \end{bmatrix} \end{array}$$