Log-determinant Non-negative Matrix Factorization via Successive Trace Approximation

Optimizing an optimization algorithm

Andersen Ang

Mathématique et recherche opérationnelle
UMONS, Belgium

Email: manshun.ang@umons.ac.be
Homepage: angms.science

File created : November 30, 2017
Last update : May 18, 2018

Joint work with my supervisor : Nicolas Gillis (UMONS, Belgium)

Overview

1. **Background and the research problem**
   - Non-negative Matrix Factorization
   - Separable NMF
   - Why Separable NMF is not enough
   - The minimum volume criterion
2. **Minimum volume log-determinant (logdet) NMF**
   - The logdet regularization
   - A logdet inequality
   - An upper bound of logdet
3. **Solving logdet NMF - Successive Trace Approximation**
   - STA algorithm
   - The NNQPs
   - Optimizing STA
4. **Experiments**
5. **Discussions, extensions and directions**
   - Coordinate acceleration by randomization
   - On robustness and robust formulations
     - De-noising norm
     - Iterative Reweighted Least Sqaures
   - Theoretical convergences
   - Comparisons with benchmark algorithms (skiped here)
   - On selecting the regularization parameter λ
   - On automatic detection of r
   - Further acceleration with weighted column fitting norms

What is Non-negative Matrix Factorization (NMF) ?

Given $\mathbf{X} \in \mathbb{R}_{+}^{m \times n}$ and integer r, find matrices $\mathbf{W} \in \mathbb{R}_{+}^{m \times r}, \mathbf{H} \in \mathbb{R}_{+}^{r \times n}$ s.t.

$$\mathbf{X} = \mathbf{WH}.$$

1. This is called : extact NMF and it is NP-hard (Vavasis2007).

What is Non-negative Matrix Factorization (NMF) ?

Given $\mathbf{X} \in \mathbb{R}_+^{m \times n}$ and integer r, find matrices $\mathbf{W} \in \mathbb{R}_+^{m \times r}, \mathbf{H} \in \mathbb{R}_+^{r \times n}$ s.t.

$$\mathbf{X} = \mathbf{W}\mathbf{H}.$$

1. This is called : extact NMF and it is NP-hard (Vavasis2007).
2. We consider
   - low rank/complexity model $1 \leq r \leq \min\{m, n\}$.
   - *approximate* NMF

$$[\mathbf{W}, \mathbf{H}] = \underset{\mathbf{W} \geq 0, \mathbf{H} \geq 0}{\arg\min} \frac{1}{2} \|\mathbf{X} - \mathbf{W}\mathbf{H}\|_F^2.$$

What is Non-negative Matrix Factorization (NMF) ?

Given $\mathbf{X} \in \mathbb{R}_+^{m \times n}$ and integer r, find matrices $\mathbf{W} \in \mathbb{R}_+^{m \times r}, \mathbf{H} \in \mathbb{R}_+^{r \times n}$ s.t.

$$\mathbf{X} = \mathbf{W}\mathbf{H}.$$

1. This is called : extact NMF and it is NP-hard (Vavasis2007).
2. We consider
   - low rank/complexity model $1 \leq r \leq \min\{m, n\}$.
   - *approximate* NMF

$$[\mathbf{W}, \mathbf{H}] = \underset{\mathbf{W} \geq 0, \mathbf{H} \geq 0}{\arg\min} \frac{1}{2} \|\mathbf{X} - \mathbf{W}\mathbf{H}\|_F^2.$$

3. Such minimization problem is
   - also NP-hard
   - a non-convex problem
   - an ill-posed problem

What is Non-negative Matrix Factorization (NMF) ?

Given $\mathbf{X} \in \mathbb{R}_{+}^{m \times n}$ and integer r, find matrices $\mathbf{W} \in \mathbb{R}_{+}^{m \times r}, \mathbf{H} \in \mathbb{R}_{+}^{r \times n}$ s.t.

$$\mathbf{X} = \mathbf{W}\mathbf{H}.$$

1. This is called : extact NMF and it is NP-hard (Vavasis2007).
2. We consider
   - low rank/complexity model $1 \leq r \leq \min\{m, n\}$.
   - *approximate* NMF

$$[\mathbf{W}, \mathbf{H}] = \underset{\mathbf{W} \geq 0, \mathbf{H} \geq 0}{\arg\min} \frac{1}{2} \|\mathbf{X} - \mathbf{W}\mathbf{H}\|_F^2.$$

3. Such minimization problem is
   - also NP-hard
   - a non-convex problem
   - an ill-posed problem
4. Assumptions (1) $\mathbf{W}$, $\mathbf{H}$ full rank, (2) r is known.
5. Notation note : we use $\mathbf{W}\mathbf{H}$ instead of $\mathbf{W}\mathbf{H}^\top$

Separable NMF

Given $\mathbf{X} \in \mathbb{R}_{+}^{m \times n}$ and integer r, find matrices $\mathbf{W} \in \mathbb{R}_{+}^{m \times r}, \mathbf{H} \in \mathbb{R}_{+}^{r \times n}$ s.t.

$$\mathbf{X} = \mathbf{WH} \text{ and } \underbrace{\mathbf{W} = \mathbf{X}(:,\mathcal{K})}_{\text{separability}}.$$

1. Separable NMF = NMF + additional condition $\mathbf{W} = \mathbf{X}(:,\mathcal{K})$
   - Meaning : $\mathbf{W}$ is some columns of $\mathbf{X}$.
   - $\mathcal{K}$: column index set
   - $|\mathcal{K}| = r$

Separable NMF

Given $\mathbf{X} \in \mathbb{R}_{+}^{m \times n}$ and integer r, find matrices $\mathbf{W} \in \mathbb{R}_{+}^{m \times r}, \mathbf{H} \in \mathbb{R}_{+}^{r \times n}$ s.t.

$$\mathbf{X} = \mathbf{WH} \text{ and } \underbrace{\mathbf{W} = \mathbf{X}(:, \mathcal{K})}_{\text{separability}}.$$

1. Separable NMF = NMF + additional condition $\mathbf{W} = \mathbf{X}(:, \mathcal{K})$
   - Meaning : $\mathbf{W}$ is some columns of $\mathbf{X}$.
   - $\mathcal{K}$: column index set
   - $|\mathcal{K}| = r$
2. Not NP-hard anymore (by Donohol, Arora et al., etc.)
3. Existing algorithms that solve Separable NMF :
   - XRAY (Kumar,2013)
   - SPA, SNPA (Gillis, 2013)

Illustrative examples

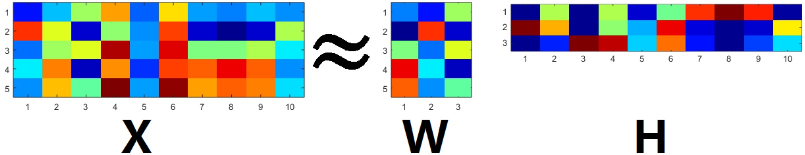


Figure: NMF, $m = 5, n = 10, r = 3$.

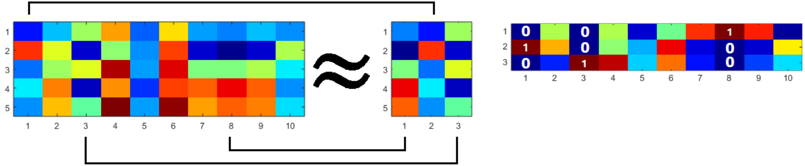


Figure: Separable NMF, $\mathcal{K} = \{8, 1, 3\}$. $\mathbf{H}$ has some special columns : only one '1', and other elements are 0.

"Pure pixels" assumption : matrix $\mathbf{H} = [\,\mathbf{I}_r \;\; \mathbf{H}'\,]\mathbf{\Pi}$.

Related terms : self-expressive, self-dictionary

Motivation — why study NMF

- In hyperspectral imaging application, $\mathbf{X}$ = image.
- $\mathbf{W}$ = absorption behaviour of materials : **non-negative** spectrum.
- $r = \#$ fundamental materials (e.g. rock, vegetation, water).
- $\mathbf{H}$ = abundance of materials : **non-negative** and sum-to-1.

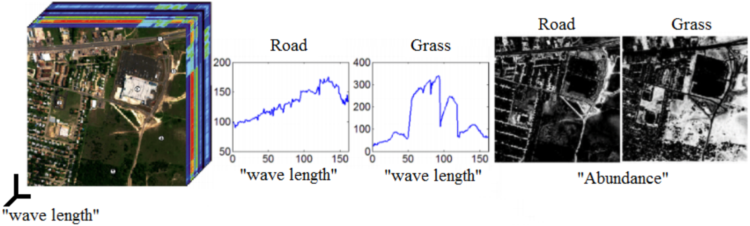


Figure: Hypersectral images decomposition. Figure copied shamelessly from N. Gillis.

Interpretation in short :

$$\mathbf{X} = \text{data}, \ \mathbf{W} = \text{basis}, \ r = \#\text{basis}, \ \mathbf{H} = \text{membership of data w.r.t. basis}$$

NMF has many other signal processing applications.

NMF vs SVD : SVD has lower fitting error (in fact SVD achieve the optimal fitting), but basis of SVD are not interpretable. Separable NMF basis comes form data, interpretable !

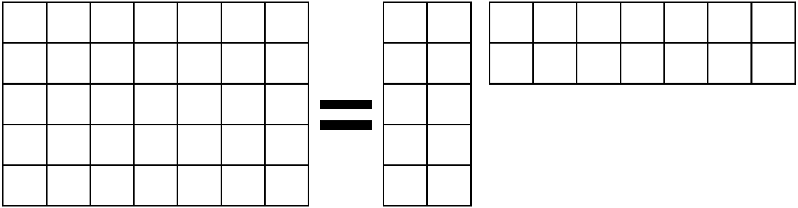
Geometry of Separable NMF : normalizing columns of $\mathbf{H}$

$\mathbf{H}$ tells the membership of data points in $\mathbf{X}$ w.r.t basis $\mathbf{W}$.

Column form expression of $\mathbf{X} = \mathbf{WH}$ is

$$\mathbf{X}(:,j) = \mathbf{WH}(:,j).$$

- Example. $m = 5, n = 7, r = 2$



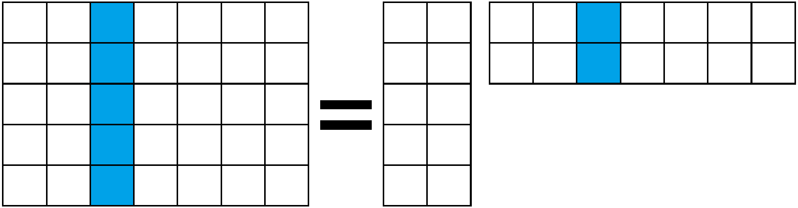
Geometry of Separable NMF : normalizing columns of $\mathbf{H}$

$\mathbf{H}$ tells the membership of data points in $\mathbf{X}$ w.r.t basis $\mathbf{W}$.

Column form expression of $\mathbf{X} = \mathbf{W}\mathbf{H}$ is

$$\mathbf{X}(\,:,\, j\,) = \mathbf{W}\mathbf{H}(\,:,\, j\,).$$

- Consider $j = 3$



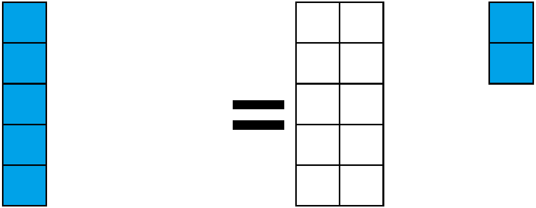
Geometry of Separable NMF : normalizing columns of $\mathbf{H}$

$\mathbf{H}$ tells the membership of data points in $\mathbf{X}$ w.r.t basis $\mathbf{W}$.

Column form expression of $\mathbf{X} = \mathbf{W}\mathbf{H}$ is

$$\mathbf{X}(:,j) = \mathbf{W}\mathbf{H}(:,j).$$

- A linear combination !



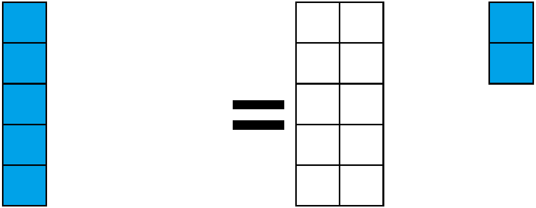
Geometry of Separable NMF : normalizing columns of $\mathbf{H}$

$\mathbf{H}$ tells the membership of data points in $\mathbf{X}$ w.r.t basis $\mathbf{W}$.

Column form expression of $\mathbf{X} = \mathbf{W}\mathbf{H}$ is

$$\mathbf{X}(:,j) = \mathbf{W}\mathbf{H}(:,j).$$

- Nonnegativity : $\mathbf{H}_{ij} \geq 0$ so it is in fact conical combination !



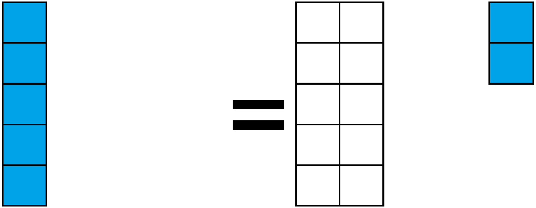
Geometry of Separable NMF : normalizing columns of $\mathbf{H}$

$\mathbf{H}$ tells the membership of data points in $\mathbf{X}$ w.r.t basis $\mathbf{W}$.

Column form expression of $\mathbf{X} = \mathbf{W}\mathbf{H}$ is

$$\mathbf{X}(:,j) = \mathbf{W}\mathbf{H}(:,j).$$

- If $\mathbf{H}(:,j)$ are normalized $\implies 0 \leq \mathbf{H}_{ij} \leq 1 \implies$ convex combination



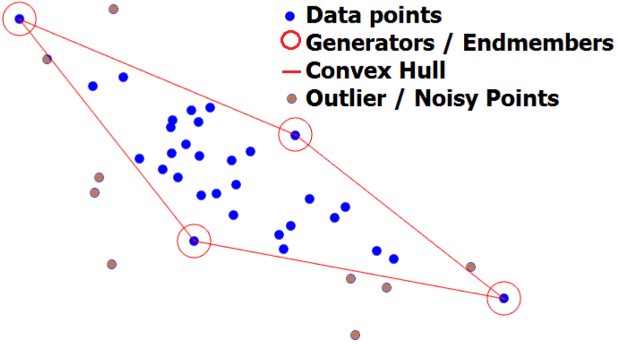
Geometry of Separable NMF : normalizing columns of $\mathbf{H}$

$\mathbf{H}$ tells the membership of data points in $\mathbf{X}$ w.r.t basis $\mathbf{W}$.

Column form expression of $\mathbf{X} = \mathbf{W}\mathbf{H}$ is

$$\mathbf{X}(:, j) = \mathbf{W}\mathbf{H}(:, j).$$

- If columns of $\mathbf{H}$ are normalized : it means $\mathbf{W}$ is forming a *convex hull* encapsulating the data columns of $\mathbf{X}$.



Geometry of Separable NMF : normalizing columns of $\mathbf{H}$

$\mathbf{H}$ tells the membership of data points in $\mathbf{X}$ w.r.t basis $\mathbf{W}$.

Column form expression of $\mathbf{X} = \mathbf{W}\mathbf{H}$ is

$$\mathbf{X}(:,j) = \mathbf{W}\mathbf{H}(:,j).$$

- Algebarically, column normalization of $\mathbf{H}$ removes the scaling ambiguity of factorization, prevents huge $\mathbf{H}$ and super small $\mathbf{W}$ as

$$\mathbf{W}_1\mathbf{H}_1 = \underbrace{\mathbf{W}_1\Lambda\Pi}_{\mathbf{W}_2}\underbrace{\Pi^{-1}\Lambda^{-1}\mathbf{H}_1}_{\mathbf{H}_2}$$

No normalization : convex hull $\to$ conical hull $\implies$ scaling ambiguity!

What has been done in literature

The problem statement : given the data points that has pure pixel (i.e. data points are **distributed in the entire** data subspace).
Goal : find the vertices = (1) find r (#vertices), (2) locate them.

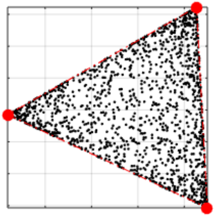


Figure: A 2D PCA projection of a high dimensional data, showing the data points (black dots) encapsulated inside convex hull spanned by the generator (vertices).

This problem $\subset$ Blind source identification with even distributed data

Existing methods : XRAY, SPA, SNPA ...

This work : what if pure pixel ($\mathbf{H}_{ij} = 1$) is hidden in data?

The problem statement : given the data points that are at least $1 - \theta$ away from the vertices (i.e. the pure pixel are hidden, data points are **not distributed in the entire** data subspace).

Goal – find the vertices : (1) find r (#vertices), (2) locate them.

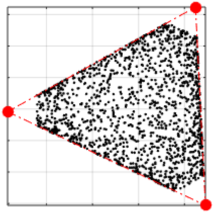


Figure: A 2D PCA projection of a high dimensional data, showing the **data points** encapsulated inside convex hull spanned by the **generator vertices**.

This work : what if pure pixel ($\mathbf{H}_{ij} = 1$) is hidden in data?

The problem statement : given the data points that are at least $1 - \theta$ away from the vertices (i.e. the pure pixel are hidden, data points are **not distributed in the entire** data subspace).

Goal – find the vertices : (1) find r (#vertices), (2) locate them.

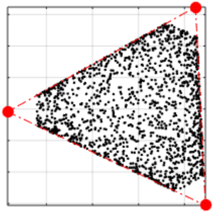


Figure: A 2D PCA projection of a high dimensional data, showing the **data points** encapsulated inside convex hull spanned by the **generator vertices**.

- θ : "purity" of the data, $\theta \in [0 \;\; 1]$.
- $\theta = 1$: Separable NMF.
- Problem (1) is not considered here : $r = \#$ vertices (red dots) is assume known.

Why separable NMF is not enough ... (1/2)

Existing algorithms aimed solving the Separable NMF ($\theta = 1$) work poorly on the problems with $\theta < 1$.

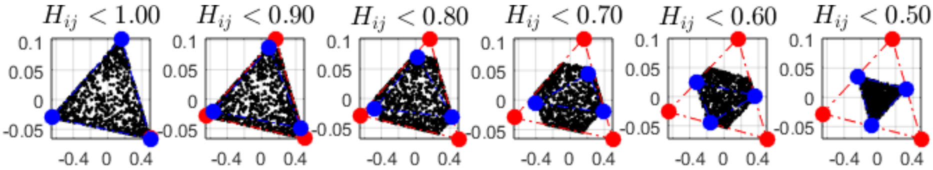


Figure: Results (2d PCA projection) of SNPA with decreasing θ. The dimensions are $(m, n, r) = (8, 1000, 3)$.

Why separable NMF is not enough ... (2/2)

Due to the nature of high dimensional geometry, when (m, r) increase, the data points are getting more and more concentrated around the annulus of the origin and thus they are **not distributed in entire data subspace**. Making approches that use L_2 norm of data points (such as SNPA) less workable.

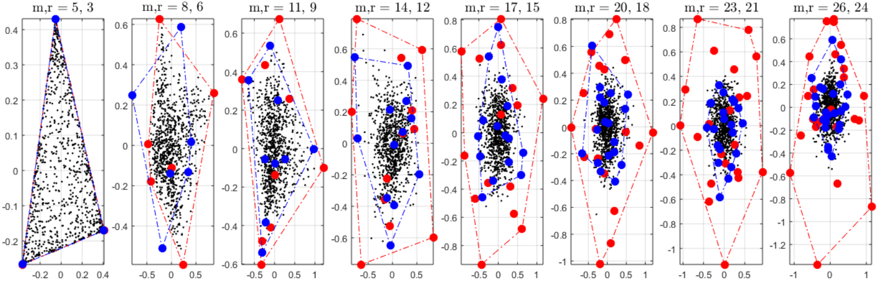


Figure: Results (2d PCA proj.) of SNPA with increasing (m, r). Red dots : ground truth vertices. Blue dots : estimated vertices. The dimensions are $(n, \theta) = (1000, 0.999)$.

The research problem : find the vertices from the data without the pure pixel ($\mathbf{H}_{ij} = 1$)

- An idea from 1994[1] : fit a low rank convex hull with minimum volume.

[1] Craig, Minimum-volume transforms for remotely sensed data. IEEE Trans. Geosci. Remote Sensing

[2] Lin, et. al, Identifiability of the simplex volume minimization criterion for blind hyperspectral unmixing: The no-pure-pixel case. IEEE Trans Geosci. Remote Sensing

The research problem : find the vertices from the data without the pure pixel ($\mathbf{H}_{ij} = 1$)

- An idea from 1994[1] : fit a low rank convex hull with minimum volume.
- Theoretical result in 2015[2] : such hull is identifiable if the data points are *well spreaded* (the underlying θ is not too small)
  $\Longrightarrow$ **volume** regularized NMF.

[1] Craig, Minimum-volume transforms for remotely sensed data. IEEE Trans. Geosci. Remote Sensing

[2] Lin, et. al, Identifiability of the simplex volume minimization criterion for blind hyperspectral unmixing: The no-pure-pixel case. IEEE Trans Geosci. Remote Sensing

The research problem : find the vertices from the data without the pure pixel ($\mathbf{H}_{ij} = 1$)

- An idea from 1994[1] : fit a low rank convex hull with minimum volume.
- Theoretical result in 2015[2] : such hull is identifiable if the data points are *well spreaded* (the underlying θ is not too small)
  $\implies$ **volume** regularized NMF.
- Volume is related to determinant $\implies \det \mathbf{W}$ regularization

[1] Craig, Minimum-volume transforms for remotely sensed data. IEEE Trans. Geosci. Remote Sensing

[2] Lin, et. al, Identifiability of the simplex volume minimization criterion for blind hyperspectral unmixing: The no-pure-pixel case. IEEE Trans Geosci. Remote Sensing

The research problem : find the vertices from the data without the pure pixel ($\mathbf{H}_{ij} = 1$)

- An idea from 1994[1] : fit a low rank convex hull with minimum volume.
- Theoretical result in 2015[2] : such hull is identifiable if the data points are *well spreaded* (the underlying θ is not too small)
  $\implies$ **volume** regularized NMF.
- Volume is related to determinant $\implies$ $\det \mathbf{W}$ regularization
- $\det \mathbf{W}$ only works for sqaure $\mathbf{W}$ $\implies$ consider $\det \mathbf{W}^\top \mathbf{W}$ or $\log \det \mathbf{W}^\top \mathbf{W}$

[1] Craig, Minimum-volume transforms for remotely sensed data. IEEE Trans. Geosci. Remote Sensing

[2] Lin, et. al, Identifiability of the simplex volume minimization criterion for blind hyperspectral unmixing: The no-pure-pixel case. IEEE Trans Geosci. Remote Sensing

The research problem : find the vertices from the data without the pure pixel ($\mathbf{H}_{ij} = 1$)

- An idea from 1994[1] : fit a low rank convex hull with minimum volume.
- Theoretical result in 2015[2] : such hull is identifiable if the data points are *well spreaded* (the underlying θ is not too small)
  $\Longrightarrow$ **volume** regularized NMF.
- Volume is related to determinant $\Longrightarrow$ $\det \mathbf{W}$ regularization
- $\det \mathbf{W}$ only works for sqaure $\mathbf{W}$ $\Longrightarrow$ consider $\det \mathbf{W}^\top \mathbf{W}$ or $\log\det \mathbf{W}^\top \mathbf{W}$

$$\text{detNMF}: \quad \min_{\substack{\mathbf{W}\geq 0\\ \mathbf{H}\geq 0\\ \mathbf{1}_r^\top \mathbf{H}\leq \mathbf{1}_n}} \underbrace{\frac{1}{2}\|\mathbf{X}-\mathbf{WH}\|_F^2}_{\text{data fitting term } \mathcal{F}} + \frac{\lambda}{2} \underbrace{\det(\mathbf{W}^\top\mathbf{W})}_{\text{volume regularizer } \mathcal{G}} .$$

$$\text{logdetNMF}: \quad \min_{\substack{\mathbf{W}\geq 0\\ \mathbf{H}\geq 0\\ \mathbf{1}_r^T \mathbf{H}\leq \mathbf{1}_n}} \underbrace{\frac{1}{2}\|\mathbf{X}-\mathbf{WH}\|_F^2}_{\text{data fitting term } \mathcal{F}} + \frac{\lambda}{2} \underbrace{\log\det(\mathbf{W}^\top\mathbf{W}+\delta\mathbf{I})}_{\text{volume regularizer } \mathcal{G}} .$$

[1] Craig, Minimum-volume transforms for remotely sensed data. IEEE Trans. Geosci. Remote Sensing

[2] Lin, et. al, Identifiability of the simplex volume minimization criterion for blind hyperspectral unmixing: The no-pure-pixel case. IEEE Trans Geosci. Remote Sensing

Log-determinant Non-negative Matrix Factorization

Given $\mathbf{X} \in \mathbb{R}^{m \times n}$, find matrices $\mathbf{W} \in \mathbb{R}_+^{m \times r}, \mathbf{H} \in \mathbb{R}_+^{r \times n}$ by solving

$$\min_{\substack{\mathbf{W} \geq 0 \\ \mathbf{H} \geq 0 \\ \mathbf{1}_r^\top \mathbf{H} \leq \mathbf{1}_n}} \underbrace{\frac{1}{2} \|\mathbf{X} - \mathbf{W}\mathbf{H}\|_F^2}_{\text{data fitting term}} + \frac{\lambda}{2} \underbrace{\log\det(\mathbf{W}^\top \mathbf{W} + \delta \mathbf{I}_r)}_{\text{volume regularizer}}.$$

- $\lambda > 0$: tuning parameters (regularization parameter).
- ($r \in \mathbb{N}_+$ assumed known, $\mathbf{W}, \mathbf{H}$ assumed full rank).

Log-determinant Non-negative Matrix Factorization

Given $\mathbf{X} \in \mathbb{R}^{m \times n}$, find matrices $\mathbf{W} \in \mathbb{R}_+^{m \times r}, \mathbf{H} \in \mathbb{R}_+^{r \times n}$ by solving

$$\min_{\substack{\mathbf{W} \geq 0 \\ \mathbf{H} \geq 0 \\ \mathbf{1}_r^\top \mathbf{H} \leq \mathbf{1}_n}} \underbrace{\frac{1}{2}\|\mathbf{X} - \mathbf{W}\mathbf{H}\|_F^2}_{\text{data fitting term}} + \frac{\lambda}{2} \underbrace{\log\det(\mathbf{W}^\top \mathbf{W} + \delta \mathbf{I}_r)}_{\text{volume regularizer}}.$$

- $\lambda > 0$: tuning parameters (regularization parameter).
- δ : fix small positive constant (e.g. 1).
- Why $\delta \mathbf{I}_r$: to bound $\log\det$ (otherwise $\lim\limits_{\|\mathbf{W}\| \to 0} \log\det \mathbf{W}^\top \mathbf{W} \to -\infty$).
- ($r \in \mathbb{N}_+$ assumed known, $\mathbf{W}, \mathbf{H}$ assumed full rank).

Two Block Coordinate Descent solution framework

The optimization problem : given $\mathbf{X} \in \mathbb{R}^{m \times n}$ and $r \geq 1$

$$\min_{\substack{\mathbf{W} \geq 0 \\ \mathbf{H} \geq 0 \\ \mathbf{1}_r^\top \mathbf{H} \leq \mathbf{1}_n}} \|\mathbf{X} - \mathbf{W}\mathbf{H}\|_F^2 + \lambda \log\det(\mathbf{W}^\top \mathbf{W} + \delta \mathbf{I}_r),$$

1: INPUT : $\mathbf{X} \in \mathbb{R}^{m \times n}$, $r \in \mathbb{N}_+$ and $\lambda \geq 0$

2: OUTPUT : $\mathbf{W} \in \mathbb{R}_+^{m \times r}$ and $\mathbf{H} \in \mathbb{R}_+^{r \times n}$

3: INITIALIZATION : $\mathbf{W} \in \mathbb{R}_+^{m \times r}$ and $\mathbf{H} \in \mathbb{R}_+^{r \times n}$

4: **for** $k = 1$ to itermax **do**

5: $\quad \mathbf{W} \leftarrow \arg\min_{\mathbf{W} \geq 0} \|\mathbf{X} - \mathbf{W}\mathbf{H}\|_F^2 + \lambda \log\det(\mathbf{W}^\top \mathbf{W} + \delta \mathbf{I}_r)$.

6: $\quad \mathbf{H} \leftarrow \arg\min_{\mathbf{H} \geq 0, \mathbf{1}_r^\top \mathbf{H} \leq \mathbf{1}_n} \|\mathbf{X} - \mathbf{W}\mathbf{H}\|_F^2 + \lambda \log\det(\mathbf{W}^\top \mathbf{W} + \delta \mathbf{I}_r)$.

7: **end for**

From now on, line 1-3 will be skipped (for space)

Subproblems lines 5-6 can be solve by projected gradient.

On solving $\mathbf{H}$ and $\mathbf{W}$

To solve

$$\mathbf{H} \leftarrow \underset{\substack{\mathbf{H}\geq 0 \\ \mathbf{1}_r^\top \mathbf{H} \leq \mathbf{1}_n}}{\arg\min} \|\mathbf{X} - \mathbf{W}\mathbf{H}\|_F^2 + \underbrace{\lambda \log\det(\mathbf{W}^\top \mathbf{W} + \delta \mathbf{I}_r)}_{\text{independent of } \mathbf{H}},$$

the FGM (Fast gradient method on constrainted least sqaure on unit simplex) from N. Gillis† will be used :

1: **for** $k = 1$ to itermax **do**
2: $\quad \mathbf{W} \leftarrow \underset{\mathbf{W}\geq 0}{\arg\min} \|\mathbf{X} - \mathbf{W}\mathbf{H}\|_F^2 + \lambda \log\det(\mathbf{W}^\top \mathbf{W} + \delta \mathbf{I}_r)$.
3: $\quad$ Update $\mathbf{H}$ using FGM† with $\{\mathbf{X}, \mathbf{W}, \mathbf{H}\}$.
4: **end for**

---

† N. Gillis, "Successive Nonnegative Projection Algorithm for Robust Nonnegative Blind Source Separation", SIAM J. on Imaging Sciences 7 (2), pp. 1420-1450, 2014.

On solving $\mathbf{H}$ and $\mathbf{W}$

So the key problem is to solve for W.

- Data fitting part $\|\mathbf{X} - \mathbf{W}\mathbf{H}\|_F^2$ is easy to handle.
- The regularizer $\log\det(\mathbf{W}^\top\mathbf{W} + \delta\mathbf{I}_r)$ is problematic : non-convex, column-coupled, non-proximable.
- *Don't forget we can always solve this problem just by vanilla projected gradient, but such approach does not utilize the structure of the problem — not good !

---

for $k = 1$ to itermax **do**
  $\mathbf{W} \leftarrow \underset{\mathbf{W}\geq 0}{\arg\min} \|\mathbf{X} - \mathbf{W}\mathbf{H}\|_F^2 + \lambda \log\det(\mathbf{W}^\top\mathbf{W} + \delta\mathbf{I}_r)$.
  Update $\mathbf{H}$ using FGM† with $\{\mathbf{X}, \mathbf{W}, \mathbf{H}\}$.
end for

---

† N. Gillis, "Successive Nonnegative Projection Algorithm for Robust Nonnegative Blind Source Separation", SIAM J. on Imaging Sciences 7 (2), pp. 1420-1450, 2014.

Previous lines of attack on $\log\det$

Previous lines of attack (fails or no result) :

- Proximal operator on $+\log\det \mathbf{W}^\top \mathbf{W}$
- Hadamard's inequality : $|\det(\mathbf{A})| < \prod_i \|a_i\|_2^2$
  (Upper bound of volume spanned by $\mathbf{A} = [a_1\, a_2\, ...]$)
- exp-trace-log equation : $\det(\mathbf{I}_r + \mathbf{A}) = \exp\operatorname{tr}\log(\mathbf{I}_r + \mathbf{A})$
- Approximating the determinant 1. Diagonal Approximations
- Approximating the determinant 2. Eigenspectrum approximations

$$
\begin{aligned}
\det(\mathbf{I}_r + \delta\mathbf{A}) &= \det(\mathbf{I}_r + \delta P^{-1}JP) \\
&= \det(P^{-1}(\mathbf{I} + \delta J)P) \\
&= \det(P^{-1}P(\mathbf{I} + \delta J)) \\
&= \prod_i (1 + \delta J_{ii}) \\
&= 1 + \delta \sum_i J_{ii} + \delta^2 \sum_{i,j,j\neq i} J_{ii}J_{jj} + ...
\end{aligned}
$$

- A bound from telecom research :
  $\log\det(\mathbf{W}^\top\mathbf{W} + \delta\mathbf{I}_r) \leq \operatorname{tr}(D^{\text{Taylor}}\mathbf{A}^\top\mathbf{A}) - \log\det(\mathbf{D}^{\text{Taylor}}) - r$ where $\mathbf{D}^{\text{Taylor}} = (\mathbf{W}_{-1}^\top\mathbf{W}_{-1} + \delta\mathbf{I}_r)^{-1}$, this is infact the first order Taylor convex upper bound of the function $\log\det(X)$. Reference includes : S.Christensen et al., "Weighted Sum-Rate Maximization using Weighted MMSE for MIMO-BC Beamforming Design", IEEE Trans. Wireless Com., pp. 4792-4799, vol. 7, issue 12, 2008

The key iequality : logdet-trace inequality

Given a positive definite matrix $\mathbf{A} \in \mathbb{R}^{r \times r}$, we have

$$\operatorname{tr}(\mathbf{I}_r - \mathbf{A}^{-1}) \leq \log\det \mathbf{A} \leq \operatorname{tr}(\mathbf{A} - \mathbf{I}_r)$$

Sowe now have an upper bound : put $\mathbf{A} = \mathbf{W}^\top \mathbf{W} + \delta \mathbf{I}_r$

$$\begin{aligned} \log\det(\mathbf{W}^t \mathbf{W} + \delta \mathbf{I}_r) &\leq \operatorname{tr}(\mathbf{W}^\top \mathbf{W} + (\delta - 1)\mathbf{I}_r) \\ \|\mathbf{X} - \mathbf{W}\mathbf{H}\|_F^2 + \lambda \log\det(\mathbf{W}^\top \mathbf{W} + \delta \mathbf{I}_r) &\leq \|\mathbf{X} - \mathbf{W}\mathbf{H}\|_F^2 + \lambda \operatorname{tr}(\mathbf{W}^\top \mathbf{W} + (\delta - 1)\mathbf{I}_r) \end{aligned}$$

- $\log\det(\mathbf{W}^\top \mathbf{W} + \delta \mathbf{I}_r)$ is not convex w.r.t. $\mathbf{W}$ but the trace is.

The key iequality : logdet-trace inequality

Given a positive definite matrix $\mathbf{A} \in \mathbb{R}^{r \times r}$, we have

$$\operatorname{tr}(\mathbf{I}_r - \mathbf{A}^{-1}) \leq \log\det \mathbf{A} \leq \operatorname{tr}(\mathbf{A} - \mathbf{I}_r)$$

Sowe now have an upper bound : put $\mathbf{A} = \mathbf{W}^\top \mathbf{W} + \delta \mathbf{I}_r$

$$\begin{aligned} \log\det(\mathbf{W}^t\mathbf{W} + \delta\mathbf{I}_r) &\leq \operatorname{tr}(\mathbf{W}^\top\mathbf{W} + (\delta - 1)\mathbf{I}_r) \\ \|\mathbf{X} - \mathbf{W}\mathbf{H}\|_F^2 + \lambda \log\det(\mathbf{W}^\top\mathbf{W} + \delta\mathbf{I}_r) &\leq \|\mathbf{X} - \mathbf{W}\mathbf{H}\|_F^2 + \lambda \operatorname{tr}(\mathbf{W}^\top\mathbf{W} + (\delta - 1)\mathbf{I}_r) \end{aligned}$$

- $\log\det(\mathbf{W}^\top\mathbf{W} + \delta\mathbf{I}_r)$ is not convex w.r.t. $\mathbf{W}$ but the trace is.
- Algorithm that minimizes this upper bound :

1: **for** $k = 1$ to itermax **do**
2: $\quad \mathbf{W} \leftarrow \underset{\mathbf{W} \geq 0}{\arg\min} \|\mathbf{X} - \mathbf{W}\mathbf{H}\|_F^2 + \lambda \operatorname{tr}(\mathbf{W}^\top\mathbf{W} + (\delta - 1)\mathbf{I}_r)$.
3: $\quad$ Update $\mathbf{H}$ using FGM with $\mathbf{X}, \mathbf{W}, \mathbf{H}$.
4: **end for**

The key iequality : logdet-trace inequality

Given a positive definite matrix $\mathbf{A} \in \mathbb{R}^{r \times r}$, we have

$$\operatorname{tr}(\mathbf{I}_r - \mathbf{A}^{-1}) \leq \log\det \mathbf{A} \leq \operatorname{tr}(\mathbf{A} - \mathbf{I}_r)$$

Sowe now have an upper bound : put $\mathbf{A} = \mathbf{W}^\top \mathbf{W} + \delta \mathbf{I}_r$

$$\begin{aligned} \log\det(\mathbf{W}^t \mathbf{W} + \delta \mathbf{I}_r) &\leq \operatorname{tr}(\mathbf{W}^\top \mathbf{W} + (\delta - 1)\mathbf{I}_r) \\ \|\mathbf{X} - \mathbf{W}\mathbf{H}\|_F^2 + \lambda \log\det(\mathbf{W}^\top \mathbf{W} + \delta \mathbf{I}_r) &\leq \|\mathbf{X} - \mathbf{W}\mathbf{H}\|_F^2 + \lambda \operatorname{tr}(\mathbf{W}^\top \mathbf{W} + (\delta - 1)\mathbf{I}_r) \end{aligned}$$

- $\log\det(\mathbf{W}^\top \mathbf{W} + \delta \mathbf{I}_r)$ is not convex w.r.t. $\mathbf{W}$ but the trace is.
- Algorithm that minimizes this upper bound :

1: **for** $k = 1$ to itermax **do**
2: $\quad \mathbf{W} \leftarrow \underset{\mathbf{W} \geq 0}{\arg\min} \|\mathbf{X} - \mathbf{W}\mathbf{H}\|_F^2 + \lambda \operatorname{tr}(\mathbf{W}^\top \mathbf{W} + (\delta - 1)\mathbf{I}_r)$.
3: $\quad$ Update $\mathbf{H}$ using FGM with $\mathbf{X}, \mathbf{W}, \mathbf{H}$.
4: **end for**

- Don't stop here, it can be better !!

A closer look on the logdet-trace inequality

- Given a positive definite matrix $\mathbf{A} \in \mathbb{R}^{r \times r}$, we have

$$\log\det \mathbf{A} \le \operatorname{tr}(\mathbf{A} - \mathbf{I}_r).$$

A closer look on the logdet-trace inequality

- Given a positive definite matrix $\mathbf{A} \in \mathbb{R}^{r \times r}$, we have

$$\log\det \mathbf{A} \le \mathrm{tr}(\mathbf{A} - \mathbf{I}_r).$$

- Let μ denotes eigenvalues, we have

$$\det \mathbf{A} = \prod_i \mu_i \quad \text{and} \quad \mathrm{tr}\, \mathbf{A} = \sum_i \mu_i.$$

- $\Longrightarrow$

$$\log\det \mathbf{A} \le \mathrm{tr}(\mathbf{A} - \mathbf{I}_r) \iff \sum_i \log \mu_i \le \sum_i (\mu_i - 1).$$

A closer look on the logdet-trace inequality

- Given a positive definite matrix $\mathbf{A} \in \mathbb{R}^{r \times r}$, we have

$$\log\det\mathbf{A} \le \operatorname{tr}(\mathbf{A} - \mathbf{I}_r).$$

- Let μ denotes eigenvalues, we have

$$\det\mathbf{A} = \prod_i \mu_i \quad \text{and} \quad \operatorname{tr}\mathbf{A} = \sum_i \mu_i.$$

- $\Longrightarrow$

$$\log\det\mathbf{A} \le \operatorname{tr}(\mathbf{A} - \mathbf{I}_r) \iff \sum_i \log\mu_i \le \sum_i (\mu_i - 1).$$

- $\log\mu_i$ means matrix $\mathbf{A}$ has to be positive definite ($\mu_i > 0 \,\forall\, i$), which is satisfied for $\mathbf{A} = \mathbf{W}^\top\mathbf{W} + \delta\mathbf{I}_r$.

A closer look on the logdet-trace inequality

- Given a positive definite matrix $\mathbf{A} \in \mathbb{R}^{r \times r}$, we have

$$\log\det \mathbf{A} \leq \operatorname{tr}(\mathbf{A} - \mathbf{I}_r).$$

- Let μ denotes eigenvalues, we have

$$\det \mathbf{A} = \prod_i \mu_i \quad \text{and} \quad \operatorname{tr} \mathbf{A} = \sum_i \mu_i.$$

- $\implies$

$$\log\det \mathbf{A} \leq \operatorname{tr}(\mathbf{A} - \mathbf{I}_r) \iff \sum_i \log \mu_i \leq \sum_i (\mu_i - 1).$$

- $\log \mu_i$ means matrix $\mathbf{A}$ has to be positive definite ($\mu_i > 0 \,\forall\, i$), which is satisfied for $\mathbf{A} = \mathbf{W}^\top \mathbf{W} + \delta \mathbf{I}_r$.
- $\sum_i \log \mu_i \leq \sum_i (\mu_i - 1) \;\Leftarrow\; \log \mu_i \leq \mu_i - 1 \,\forall\, i$. We can focuse on the inequality $\log \mu_i \leq \mu_i - 1$ with $\mu_i \geq 0$

On $\log x \leq x - 1$, $x \geq 0$

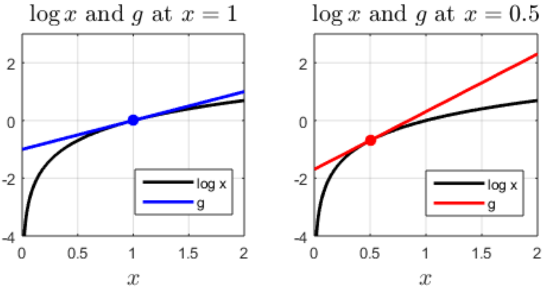
- $\log x$ is concave.
- $x - 1$ is the first order Taylor approximation of $\log x$ at $x = 1$.
- $x - 1$ is the only **convex-tight** upper bound of $\log x$.[†]
- Tight : $x - 1$ touch $\log x$ at the point $x = 1$.

† Higher order Taylor approximation of $\log x$ is tight, more accurate but not convex.

On $\log x \leq x - 1$, $x \geq 0$

- $\log x$ is concave.
- $x - 1$ is the first order Taylor approximation of $\log x$ at $x = 1$.
- $x - 1$ is the only **convex-tight** upper bound of $\log x$.†
- Tight : $x - 1$ touch $\log x$ at the point $x = 1$.
- Generalize to point x_0 : $\log x \leq g(x|x_0) = a_1(x_0)x + a_0(x_0)$ is

$$\log x \leq \frac{1}{x_0}x + \log x_0 - 1.$$



† Higher order Taylor approximation of $\log x$ is tight, more accurate but not convex.

A parametric trace upper bound for $\log \det \mathbf{A}$

$$\begin{aligned}
\log \det \mathbf{A} &= \sum \log \mu_i \\
&\leq \sum \frac{1}{\mu_i^-}\mu_i + \log \mu_i^- - 1 \\
&\leq \sum \frac{1}{\mu_{\min}^-}\mu_i + \log \mu_i^- - 1 \\
&= \operatorname{tr}(\mathbf{D}^1\mathbf{A} + \mathbf{D}^0)
\end{aligned}$$

$\mathbf{D}^1 = \dfrac{1}{\mu_{\min}^-}\mathbf{I}_r$, $\mathbf{D}^0 = \text{Diag}(\log \mu_i^- - 1)$, μ_i^- is μ_i of the previous step

Put $\mathbf{A} = \mathbf{W}^\top\mathbf{W} + \delta\mathbf{I}_r$, we have

$$\begin{aligned}
\log\det(\mathbf{W}^\top\mathbf{W} + \delta\mathbf{I}_r) &\leq \operatorname{tr}(\mathbf{D}^1\mathbf{W}^\top\mathbf{W} + \delta\mathbf{D}^1 + \mathbf{D}^0) \\
\text{(ignore constants)} &= \operatorname{tr}\mathbf{D}^1\mathbf{W}^\top\mathbf{W}
\end{aligned}$$

A note on weighted sum of eigenvalues and trace

Consider the matrix A has eigen-decomposition as $\mathbf{A} = \mathbf{V}\Lambda\mathbf{V}^T$.
Let the weighting $\frac{1}{\mu_i^-}$ be a_i, then $\sum_i \frac{1}{\mu_i^-}\mu_i = \sum_i a_i\mu_i$ and

$$
\begin{aligned}
\sum_i a_i\mu_i &= \operatorname{tr}\begin{bmatrix} a_1\mu_1 & & \\ & a_2\mu_2 & \\ & & \ddots \end{bmatrix} = \operatorname{tr}\underbrace{\begin{bmatrix} a_1 & & \\ & a_2 & \\ & & \ddots \end{bmatrix}}_{D^a}\underbrace{\begin{bmatrix} \mu_1 & & \\ & \mu_2 & \\ & & \ddots \end{bmatrix}}_{\Lambda = V^T AV} \\
&= \operatorname{tr}\mathbf{D}^a\mathbf{V}^\top\mathbf{A}\mathbf{V} \\
&\neq \operatorname{tr}\mathbf{D}^a\mathbf{A}
\end{aligned}
$$

Comparing upper bounds for $\log\det \mathbf{A}$

The original function $F(\mathbf{W}) = \log\det(\mathbf{W}^\top\mathbf{W} + \delta\mathbf{I}_r)$ is upper bounded by :

$$\begin{aligned} \text{Eigen bound} \quad (B_1): &\quad \operatorname{tr} \mathbf{D}^1\mathbf{W}^\top\mathbf{W} + \text{constants.} \\ \text{Taylor bound} \quad (B_2): &\quad \operatorname{tr} \mathbf{D}^{\text{Taylor}}\mathbf{W}^\top\mathbf{W} + \text{constants.} \end{aligned}$$

- Constants $\mathbf{D}^1$, $\mathbf{D}^0$ are defined as before, and constant $\mathbf{D}^{\text{Taylor}} = (\mathbf{W}_{-1}^\top\mathbf{W}_{-1} + \delta\mathbf{I}_r)^{-1}$.
- Both bounds are trace functional with an **relaxation gap** :
  - (B_1) has eigen gap $\mu_i \geq \mu_{\min}$
  - (B_2) has convexification gap
  - $\mathbf{D}^1$ is **diagonal** but $\mathbf{D}^{\text{Taylor}}$ is not (it is dense) $\implies$ column-wise decomposition is possible

Successive Trace Approximation (STA)

Algorithm 1 Successive Trace Approximation

```
INPUT: $\mathbf{X} \in \mathbb{R}_+^{m\times n}$, $r \in \mathbb{N}_+$, $\lambda > 0$, $\delta > 0$.
OUTPUT: $\mathbf{W} \in \mathbb{R}_+^{m\times r}$ and $\mathbf{H} \in \mathbb{R}_+^{r\times n}$.
INITIALIZATION : $\mathbf{W} \in \mathbb{R}_+^{m\times r}$ , $\mathbf{H} \in \mathbb{R}_+^{r\times n}$, $\mathbf{D}^1 = \mathbf{I}_r$
for $K = 1$ to itermax do
    for $k = 1$ to itermax do
        $\mathbf{W} \leftarrow \arg\min_{\mathbf{W}\geq 0} \|\mathbf{X} - \mathbf{W}\mathbf{H}\|_F^2 + \lambda \operatorname{tr} \mathbf{D}^1\mathbf{W}^\top\mathbf{W}$.
        Update $\mathbf{H}$ using FGM with $\mathbf{X}, \mathbf{W}, \mathbf{H}$.
    end for
    $\mu_i \leftarrow$svd$(\mathbf{W}^\top\mathbf{W} + \delta\mathbf{I}_r)$, $\mathbf{D}^1$ =Diag$(\mu_{\min}^{-1})$
end for
```

Small summary : a model relaxation

$$\|\mathbf{X}-\mathbf{W}\mathbf{H}\|_F^2 \longrightarrow \|\mathbf{X}-\mathbf{W}\mathbf{H}\|_F^2 + \lambda \log\det(\mathbf{W}^\top\mathbf{W}+\delta\mathbf{I}_r)$$

$$\downarrow$$

$$\|\mathbf{X}-\mathbf{W}\mathbf{H}\|_F^2 + \lambda \operatorname{tr}(\mathbf{W}^\top\mathbf{W}+(\delta-1)\mathbf{I}_r)$$

$$\downarrow$$

$$\|\mathbf{X}-\mathbf{W}\mathbf{H}\|_F^2 + \lambda \operatorname{tr}(\mathbf{D}^1\mathbf{W}^\top\mathbf{W}+\mathbf{D}^0)$$

In one sentence : Eigenval-wise convex relaxation of a non-convex problem using logdet – trace ineqaulity.

D^1 is diagonal : column decoupling and column-wise BCD

Consider one vector w_i while fixing all other things :

$$
\begin{aligned}
&\|\mathbf{X} - \mathbf{W}\mathbf{H}\|_F^2 + \lambda \mathrm{tr}(\mathbf{D}^1 \mathbf{W}^\top \mathbf{W} + \mathbf{D}^0) \\
&= \quad \|\mathbf{X} - \sum_i w_i h_i\|_F^2 + \lambda \sum_i \left(D^1_{ii} \|w_i\|_2^2 + \mathbf{D}^0_{ii} \right)
\end{aligned}
$$

D^1 is diagonal : column decoupling and column-wise BCD

Consider one vector w_i while fixing all other things :

$$
\begin{aligned}
&\|\mathbf{X} - \mathbf{W}\mathbf{H}\|_F^2 + \lambda \mathrm{tr}(\mathbf{D}^1 \mathbf{W}^\top \mathbf{W} + \mathbf{D}^0) \\
&= \|\mathbf{X} - \sum_i w_i h_i\|_F^2 + \lambda \sum_i \left(D^1_{ii} \|w_i\|_2^2 + \mathbf{D}^0_{ii}\right) \\
&= \|\underbrace{\left(X - \sum_{j \neq i} w_j h_j\right)}_{\mathbf{X}_i} - w_i h_i\|_F^2 + \lambda \Bigg(\mathbf{D}^1_{ii}\|w_i\|_2^2 + \underbrace{\sum_{j \neq i} \left(\mathbf{D}^1_{jj}\|w_j\|_2^2 + \mathbf{D}^0_{ii}\right)\Bigg)}_{c}
\end{aligned}
$$

D^1 is diagonal : column decoupling and column-wise BCD

Consider one vector w_i while fixing all other things :

$$
\begin{aligned}
&\|\mathbf{X}-\mathbf{WH}\|_F^2 + \lambda \mathrm{tr}(\mathbf{D}^1\mathbf{W}^\top\mathbf{W} + \mathbf{D}^0) \\
&= \|\mathbf{X} - \sum_i w_i h_i\|_F^2 + \lambda \sum_i \left(D_{ii}^1 \|w_i\|_2^2 + \mathbf{D}_{ii}^0\right) \\
&= \|\underbrace{\left(X - \sum_{j\neq i} w_j h_j\right)}_{\mathbf{X}_i} - w_i h_i\|_F^2 + \lambda \Big(\mathbf{D}_{ii}^1\|w_i\|_2^2 + \underbrace{\sum_{j\neq i}\left(\mathbf{D}_{jj}^1\|w_j\|_2^2 + \mathbf{D}_{ii}^0\right)\Big)}_{c} \\
&= \|\mathbf{X}_i - w_i h_i\|_F^2 + \lambda \mathbf{D}_{ii}^1 \|w_i\|_2^2 + c
\end{aligned}
$$

D^1 is diagonal : column decoupling and column-wise BCD

Consider one vector w_i while fixing all other things :

$$
\begin{aligned}
&\|\mathbf{X}-\mathbf{W}\mathbf{H}\|_F^2 + \lambda \mathrm{tr}(\mathbf{D}^1\mathbf{W}^\top\mathbf{W}+\mathbf{D}^0) \\
= \;& \|\mathbf{X} - \sum_i w_i h_i\|_F^2 + \lambda \sum_i \left(D^1_{ii}\|w_i\|_2^2 + \mathbf{D}^0_{ii}\right) \\
= \;& \| \underbrace{\left(X - \sum_{j\neq i} w_j h_j\right)}_{\mathbf{X}_i} - w_i h_i\|_F^2 + \lambda \Big(\mathbf{D}^1_{ii}\|w_i\|_2^2 + \underbrace{\sum_{j\neq i}\left(\mathbf{D}^1_{jj}\|w_j\|_2^2 + \mathbf{D}^0_{ii}\right)\Big)}_{c} \\
= \;& \|\mathbf{X}_i - w_i h_i\|_F^2 + \lambda \mathbf{D}^1_{ii}\|w_i\|_2^2 + c \\
\leq \;& \|\mathbf{X}_i - w_i h_i\|_F^2 + \lambda \mathbf{D}^1_{ii}\|w_i\|_2^2 + \frac{\gamma}{2}\|w_i - w_i^-\|_2^2 + c.
\end{aligned}
$$

Ignoring constants, we have a constrained regularized QP

$$
\min_{w_i \geq 0} \|\mathbf{X}_i - w_i h_i\|_F^2 + \lambda \mathbf{D}^1_{ii}\|w_i\|_2^2 + \frac{\gamma}{2}\|w_i - w_i^-\|_2^2.
$$

where w_i^- is the previous iterate of w_i, $\gamma > 0$ is a (small) constant. The proximal term $\|w_i - w_i^-\|_2^2$ penalizes w for leaving w^- too far.

A non-negative quadratic program (NNQP)

$$\min_{w_i \geq 0} \|\mathbf{X}_i - w_i h_i\|_F^2 + \lambda \mathbf{D}_{ii}^1 \|w_i\|_2^2 + \frac{\gamma}{2}\|w_i - w_i^-\|_2^2.$$

Introducing the proximal term :

- turns the QP problem strongly convex.
- gurantees $\|h\|_2^2 + \lambda \mathbf{D}_{ii}^1 + \gamma > 0$:
  Lemma Given $\mathbf{X}, h, z, \alpha, \beta$, the optimal solution of

  $$\min_{w \geq 0} \|\mathbf{X} - wh\|_F^2 + \alpha \|w\|_2^2 + \beta \|w - z\|_2^2$$

  is

  $$w = \frac{\left[\mathbf{X}h^T + \beta z\right]_+}{\|h\|_2^2 + \alpha + \beta}.$$

  Proof: skiped.

Note. This is also related to solving the problem using Newton iteration.

Optimizing STA ... 1

The STA algorithm with decomposed $f(w_i)$

INPUT: $\mathbf{X} \in \mathbb{R}_+^{m \times n}$, $r \in \mathbb{N}_+$, $\lambda > 0$ and $\delta > 0$
OUTPUT: $\mathbf{W} \in \mathbb{R}_+^{m \times r}$ and $\mathbf{H} \in \mathbb{R}_+^{r \times n}$
INITIALIZATION : $\mathbf{W} \in \mathbb{R}_+^{m \times r}$, $\mathbf{H} \in \mathbb{R}_+^{r \times n}$ and $\mathbf{D}^1 = I_r$, $\gamma = 10^{-6}$
for $K = 1$ to itermax **do**
    for $k = 1$ to itermax **do**
        for $i = 1$ to r **do**
            $w_i = \underset{w_i \geq 0}{\arg\min} f(w_i) = \|\mathbf{X}_i - w_i h_i\|_F^2 + \lambda \mathbf{D}_{ii}^1 \|w_i\|_2^2 + \frac{\gamma}{2} \|w_i - w_i^-\|_2^2$
            Update $\mathbf{H}$ by FGM with $\mathbf{X}, \mathbf{W}, \mathbf{H}$
        end for
    end for
    $\mu_i \leftarrow$ svd$(\mathbf{W}^\top \mathbf{W} + \delta I)$ and $\mathbf{D}^1 =$Diag$(\mu_{\min}^{-1})$
end for

Optimizing STA ... 2

Apply close form solution to $f(w_i)$

INPUT: $\mathbf{X} \in \mathbb{R}_+^{m\times n}$, $r \in \mathbb{N}_+$, $\lambda > 0$ and $\delta > 0$
OUTPUT: $\mathbf{W} \in \mathbb{R}_+^{m\times r}$ and $\mathbf{H} \in \mathbb{R}_+^{r\times n}$
INITIALIZATION : $\mathbf{W} \in \mathbb{R}_+^{m\times r}$, $\mathbf{H} \in \mathbb{R}_+^{r\times n}$ and $\mathbf{D}^1 = \mathbf{I}_r$, $\gamma = 10^{-6}$
for $K = 1$ to itermax **do**
    for $k = 1$ to itermax **do**
        for $i = 1$ to r **do**
            $w_i = \dfrac{\left[X_i h_i^T + \gamma w_i^-\right]_+}{\|h_i\|_2^2 + \lambda \mathbf{D}_{ii}^1 + \dfrac{\gamma}{2}}$ where $\mathbf{X}_i = \mathbf{X} - \sum_{j\neq i} w_j h_j$
            Update $\mathbf{H}$ by FGM with $\mathbf{X}, \mathbf{W}, \mathbf{H}$
        end for
    end for
    $\mu_i \leftarrow$ svd$(\mathbf{W}^\top \mathbf{W} + \delta \mathbf{I}_r)$ and $\mathbf{D}^1 =$Diag$(\mu_{\min}^{-1})$
end for

Optimizing STA ... 3

Move the update of $\mathbf{H}$ outside the loop to reduce computation burden

INPUT: $\mathbf{X} \in \mathbb{R}_+^{m \times n}$, $r \in \mathbb{N}_+$, $\lambda > 0$ and $\delta > 0$
OUTPUT: $\mathbf{W} \in \mathbb{R}_+^{m \times r}$ and $\mathbf{H} \in \mathbb{R}_+^{r \times n}$
INITIALIZATION : $\mathbf{W} \in \mathbb{R}_+^{m \times r}$, $\mathbf{H} \in \mathbb{R}_+^{r \times n}$ and $\mathbf{D}^1 = \mathbf{I}_r$, $\gamma = 10^{-6}$
for $K = 1$ to itermax **do**
    for $k = 1$ to itermax **do**
        for $i = 1$ to r **do**
            $w_i = \dfrac{\left[\mathbf{X}_i h_i^T + \gamma w_i^-\right]_+}{\|h_i\|_2^2 + \lambda \mathbf{D}_{ii}^1 + \dfrac{\gamma}{2}}$ where $\mathbf{X}_i = \mathbf{X} - \sum_{j \neq i} w_j h_j$
        end for
        Update $\mathbf{H}$ by FGM with $\mathbf{X}, \mathbf{W}, \mathbf{H}$
    end for
    $\mu_i \leftarrow$ svd$(\mathbf{W}^\top \mathbf{W} + \delta \mathbf{I}_r)$ and $\mathbf{D}^1 =$Diag$(\mu_{\min}^{-1})$
end for

Optimizing STA ... 4

Move the update of D^1 inside the loop ($\mathbf{W}^\top\mathbf{W}$ is r-by-r, small !)

INPUT: $\mathbf{X} \in \mathbb{R}_+^{m\times n}$, $r \in \mathbb{N}_+$, $\lambda > 0$ and $\delta > 0$
OUTPUT: $\mathbf{W} \in \mathbb{R}_+^{m\times r}$ and $\mathbf{H} \in \mathbb{R}_+^{r\times n}$
INITIALIZATION : $\mathbf{W} \in \mathbb{R}_+^{m\times r}$, $\mathbf{H} \in \mathbb{R}_+^{r\times n}$ and $\mathbf{D}^1 = \mathbf{I}_r$, $\gamma = 10^{-6}$
for $K = 1$ to itermax **do**
  for $k = 1$ to itermax **do**
    for $i = 1$ to r **do**
      $$w_i = \frac{\left[\mathbf{X}_i h_i^T + \gamma w_i^-\right]_+}{\|h_i\|_2^2 + \lambda \mathbf{D}^1_{ii} + \dfrac{\gamma}{2}} \quad \text{where} \quad \mathbf{X}_i = \mathbf{X} - \sum_{j\neq i} w_j h_j$$
      $\mu_i \leftarrow$ svd$(\mathbf{W}^\top\mathbf{W} + \delta \mathbf{I}_r)$ and $\mathbf{D}^1 =$Diag$(\mu_{\min}^{-1})$
    end for
    Update $\mathbf{H}$ by FGM with $\mathbf{X}, \mathbf{W}, \mathbf{H}$
  end for
end for

Optimizing STA ... 5

Now consider line 7, it can be show that it can be further optimized[3].

1: INPUT: $\mathbf{X} \in \mathbb{R}_+^{m\times n}$, $r \in \mathbb{N}_+$, $\lambda > 0$ and $\delta > 0$

2: OUTPUT: $\mathbf{W} \in \mathbb{R}_+^{m\times r}$ and $\mathbf{H} \in \mathbb{R}_+^{r\times n}$

3: INITIALIZATION : $\mathbf{W} \in \mathbb{R}_+^{m\times r}$, $\mathbf{H} \in \mathbb{R}_+^{r\times n}$ and $\mathbf{D}^1 = \mathbf{I}_r$, $\gamma = 10^{-6}$

4: **for** $K = 1$ to itermax **do**

5:     **for** $k = 1$ to itermax **do**

6:         **for** $i = 1$ to r **do**

7: $$w_i = \frac{\left[\mathbf{X}_i h^T + \gamma w_i^-\right]_+}{\|h\|_2^2 + \lambda \mathbf{D}_{ii}^1 + \dfrac{\gamma}{2}} \quad \text{where} \quad \mathbf{X}_i = \mathbf{X} - \textstyle\sum_{j\neq i} w_j h_j$$

8:             $\mu_i \leftarrow$ svd$(\mathbf{W}^\top \mathbf{W} + \delta \mathbf{I}_r)$ and $\mathbf{D}^1 =$Diag$(\mu_{\min}^{-1})$

9:         **end for**

10:         Update $\mathbf{H}$ by FGM with $\mathbf{X}, \mathbf{W}, \mathbf{H}$

11:     **end for**

12: **end for**

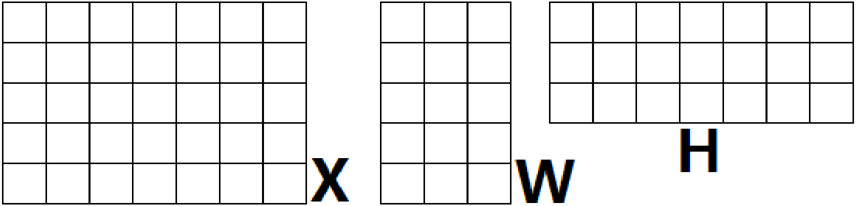
[3] N. Gillis and F. Glineur, "Accelerated Multiplicative Updates and Hierarchical ALS Algorithms for Nonnegative Matrix Factorization", Neural Computation 24 (4), 2012

On update of w_i

Line 7

$$w_i = \frac{\left[\mathbf{X}_i h_i^T + \gamma w_i^-\right]_+}{\|h_i\|_2^2 + \lambda \mathbf{D}^1_{ii} + \frac{\gamma}{2}} \quad \text{where} \quad \mathbf{X}_i = \mathbf{X} - \sum_{j \neq i} w_j h_j.$$

- Matrix $\mathbf{X}, \mathbf{W}$ and $\mathbf{H}$ with $m = 5, n = 7, r = 3$

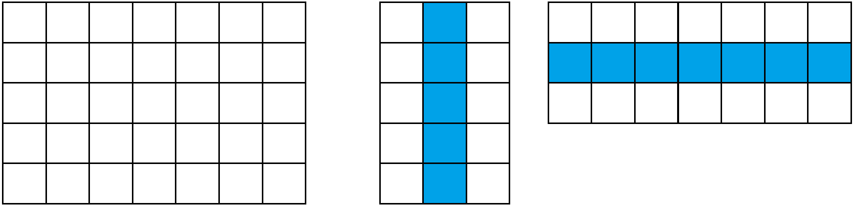


On update of w_i

Line 7

$$w_i = \frac{\left[\mathbf{X}_i h_i^T + \gamma w_i^-\right]_+}{\|h_i\|_2^2 + \lambda \mathbf{D}_{ii}^1 + \frac{\gamma}{2}} \quad \text{where} \quad \mathbf{X}_i = \mathbf{X} - \sum_{j \neq i} w_j h_j.$$

- Consider w_2.

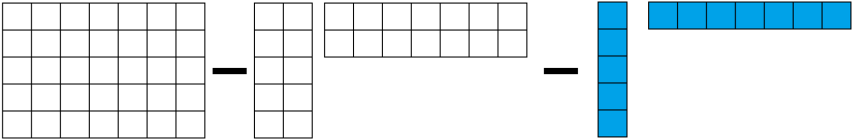


On update of w_i

Line 7

$$w_i = \frac{\left[\mathbf{X}_i h_i^T + \gamma w_i^-\right]_+}{\|h_i\|_2^2 + \lambda \mathbf{D}_{ii}^1 + \frac{\gamma}{2}} \quad \text{where} \quad \mathbf{X}_i = \mathbf{X} - \sum_{j \neq i} w_j h_j.$$

- $\mathbf{X} - \mathbf{WH} = \mathbf{X}_i - {\color{blue} w_i h_i}$.

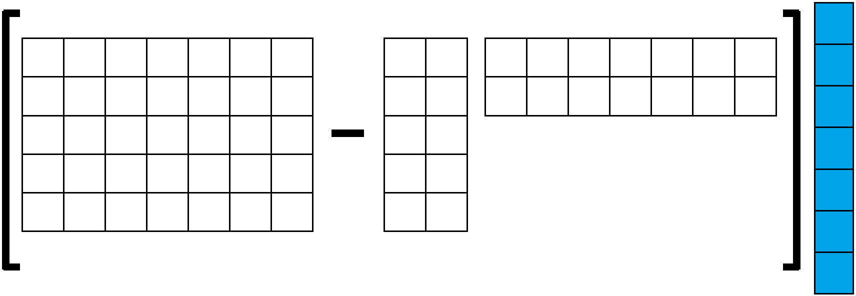


On update of w_i

Line 7

$$w_i = \frac{\left[\mathbf{X}_i h_i^T + \gamma w_i^-\right]_+}{\|h_i\|_2^2 + \lambda \mathbf{D}_{ii}^1 + \frac{\gamma}{2}} \quad \text{where} \quad \mathbf{X}_i = \mathbf{X} - \sum_{j \neq i} w_j h_j.$$

- The term $\mathbf{X}_i h_i^T$

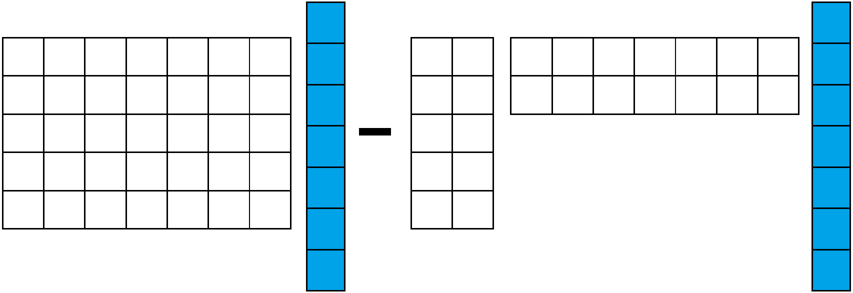


On update of w_i

Line 7

$$w_i = \frac{\left[\mathbf{X}_i h_i^T + \gamma w_i^-\right]_+}{\|h_i\|_2^2 + \lambda \mathbf{D}_{ii}^1 + \frac{\gamma}{2}} \quad \text{where} \quad \mathbf{X}_i = \mathbf{X} - \sum_{j \neq i} w_j h_j.$$

- $\mathbf{X}_i h_i^T = \mathbf{X} h_i^T - \sum_{j \neq i} w_j h_j h_i^T$

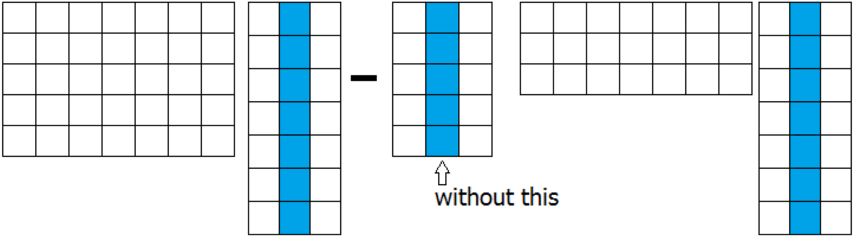


On update of w_i

Line 7

$$w_i = \frac{\left[\mathbf{X}_i h_i^T + \gamma w_i^-\right]_+}{\|h_i\|_2^2 + \lambda \mathbf{D}_{ii}^1 + \frac{\gamma}{2}} \quad \text{where} \quad \mathbf{X}_i = \mathbf{X} - \sum_{j \neq i} w_j h_j.$$

- $\mathbf{X} h_i^T - \sum_{j \neq i} w_j h_j h_i^T = [\mathbf{X}\mathbf{H}^\top](:, i) - \mathbf{W}(:, 1, ..., i-1, i+1, ..., r)[\mathbf{H}\mathbf{H}^\top](j, i)$



On update of w_i

Line 7

$$w_i = \frac{\left[\mathbf{X}_i h_i^T + \gamma w_i^-\right]_+}{\|h_i\|_2^2 + \lambda \mathbf{D}_{ii}^1 + \frac{\gamma}{2}} \quad \text{where} \quad \mathbf{X}_i = \mathbf{X} - \sum_{j \neq i} w_j h_j.$$

is equivalent to

$$w_i = \frac{\left[P_i - \sum_{j=1}^{i-1} w_j Q_{ji} - \sum_{j=i+1}^{r} w_j^- Q_{ji} + \gamma w_i^-\right]_+}{Q_{ii} + \lambda \mathbf{D}_{ii}^1 + \frac{\gamma}{2}}$$

where $P = \mathbf{X}\mathbf{H}^\top$, $P_i = P(:, i)$, $Q = \mathbf{H}\mathbf{H}^\top$ and $Q_{ji} = Q(j, i)$.
P and Q can be pre-computed.

Finally we have

Algorithm 2 STA

1: INPUT: $\mathbf{X} \in \mathbb{R}_+^{m\times n}$, $r \in \mathbb{N}_+$, $\lambda > 0$ and $\delta > 0$
2: OUTPUT: $\mathbf{W} \in \mathbb{R}_+^{m\times r}$ and $\mathbf{H} \in \mathbb{R}_+^{r\times n}$
3: INITIALIZATION : $\mathbf{W} \in \mathbb{R}_+^{m\times r}$, $\mathbf{H} \in \mathbb{R}_+^{r\times n}$ and $\mathbf{D}^1 = \mathbf{I}_r$, $\gamma = 10^{-6}$
4: **for** $K = 1$ to itermax **do**
5:     **for** $k = 1$ to itermax **do**
6:         $P = \mathbf{X}\mathbf{H}^\top$ and $Q = \mathbf{H}\mathbf{H}^\top$.
7:         **for** $i = 1$ to r **do**
8:             $w_i = \dfrac{\left[P_i - \sum_{j=1}^{i-1} w_j Q_{ji} - \sum_{j=i+1}^{r} w_j^- Q_{ji} + \gamma w_i^-\right]_+}{Q_{ii} + \lambda \mathbf{D}_{ii}^1 + \dfrac{\gamma}{2}}$
9:             $\mu_i \leftarrow$ svd$(\mathbf{W}^\top\mathbf{W} + \delta\mathbf{I}_r)$ and $D^1 =$Diag$(\mu_{\min}^{-1})$
10:         **end for**
11:         Update $\mathbf{H}$ by FGM with $\mathbf{X}, \mathbf{W}, \mathbf{H}$
12:     **end for**
13: **end for**

In fact, line $7 - 10$ can be run multiple times for "better" convergence.

Small summary

$$\|\mathbf{X} - \mathbf{W}\mathbf{H}\|_F^2$$

$$\|\mathbf{X} - \mathbf{W}\mathbf{H}\|_F^2 + \lambda \log\det(\mathbf{W}^\top \mathbf{W} + \delta \mathbf{I}_r)$$

$$\|\mathbf{X} - \mathbf{W}\mathbf{H}\|_F^2 + \lambda \operatorname{tr}(\mathbf{W}^\top \mathbf{W} + (\delta - 1) \mathbf{I}_r)$$

$$\|\mathbf{X} - \mathbf{W}\mathbf{H}\|_F^2 + \lambda \operatorname{tr}(\mathbf{D}^1 \mathbf{W}^\top \mathbf{W} + \mathbf{D}^0)$$

$$\sum_i \|\mathbf{X}_i - w_i h_i\|_F^2 + \lambda \mathbf{D}^1_{ii} \|w_i\|_2^2 + \frac{\gamma}{2} \|w_i - w_i^-\|_2^2$$

Experiments and fancy figures - setup (1/3)

Synthetic data

- Ground truth : $\mathbf{W}_0 \in \mathbb{R}_+^{m \times r}$ and $\mathbf{H}_0 \in \mathbb{R}_+^{r \times n}$ (with $\mathbf{H}^T\mathbf{1} = \alpha$)
- m, n are sizes, r is rank, $\alpha \in (0\ 1]$ tells how "well spread" the data are ($\alpha = 1$ means pure pixel)
- Form $\mathbf{X}_0$ as $\mathbf{X}_0 = \mathbf{W}_0\mathbf{H}_0 \in \mathbb{R}_+^{m \times n}$
- Add noise $N \in \mathbb{R}^{m \times n}$ and $N \sim \mathcal{N}(0, R)$ as $\mathbf{X} = \mathbf{X}_0 + N$.
- As $N \in \mathbb{R}$ not $\mathbb{R}_+$, corrupted data points may lie outside the convex hull.

Experiments and fancy figures - setup (2/3)

Real data (An example : Copperas Cove Texas Walmart)

Figure: RGB image of Copperas Cove Texas Walmart

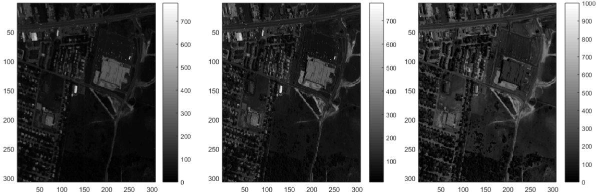


Figure: Three spectral images of Copperas Cove Texas Walmart

$\mathbf{X} \in \mathbb{R}_+^{94249 \times 162}$, or $\mathbf{X} \in \mathbb{R}_+^{307 \times 307 \times 162}$, pick $r = 5, 6, 7$.

Experiments and fancy figures - setup (3/3)

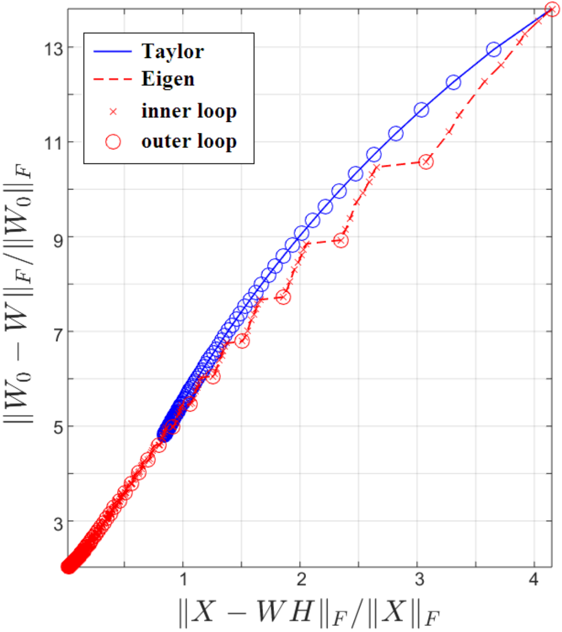
Performance measurements. Algorithm produces $\hat{\mathbf{W}}$ and $\hat{\mathbf{H}}$ and $\hat{\mathbf{X}} = \hat{\mathbf{W}}\hat{\mathbf{H}}$

- For simulation with known $\mathbf{W}_0$, $\mathbf{H}_0$, $\mathbf{X}_0$:
  - Data fitting errror : $\dfrac{\|\mathbf{X}_0 - \hat{\mathbf{X}}\|_F}{\|\mathbf{X}_0\|_F}$
  - Endmember fitting error : $\dfrac{\|\mathbf{W}_0 - \hat{\mathbf{W}}\|_F}{\|\mathbf{W}_0\|_F}$
  - Computational time
- For real data without knowing $\mathbf{W}_0$, $\mathbf{H}_0$, $\mathbf{X}_0$:
  - Data fitting errror : $\dfrac{\|\mathbf{X} - \hat{\mathbf{X}}\|_F}{\|\mathbf{X}\|_F}$
  - Volume of convex hull : $\log\det(\hat{\mathbf{W}}^\top \hat{\mathbf{W}} + \delta \mathbf{I}_r)$
  - Computational time

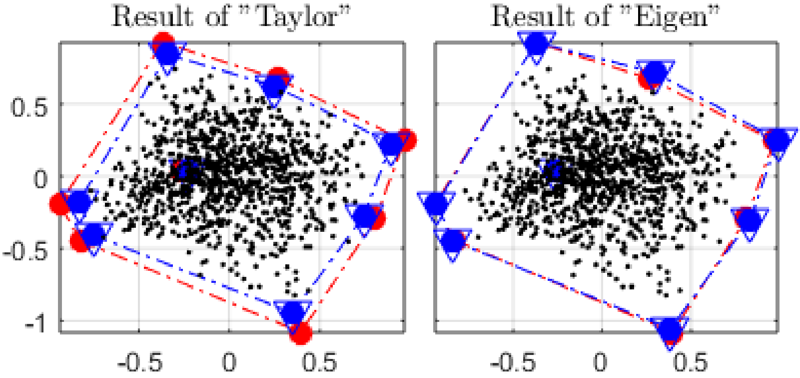
Results and fancy figures

- What iterations of the algorithm looks like : the gif file
- Effect of fix lambda : EXAMPLE Rotate
- Effect of very big lambda : EXAMPLE Big lambda

Comparing the two logdet inequality - error



Comparing the two logdet inequality - fitting



We have problems :

$$(P_0) \quad \text{minimizes} \quad \|\mathbf{X} - \mathbf{W}\mathbf{H}\|_F^2 + \lambda \log\det(\mathbf{W}^\top \mathbf{W} + \delta \mathbf{I}_r),$$
$$(P_1) \quad \text{minimizes} \quad \|\mathbf{X} - \mathbf{W}\mathbf{H}\|_F^2 + \lambda \operatorname{tr}(\mathbf{D}^1 \mathbf{W}^\top \mathbf{W} + \mathbf{D}^0),$$

under the constaints : $\mathbf{W} \geq 0, \mathbf{H} \geq 0, \mathbf{H}^\top \mathbf{1} \leq \mathbf{1}$.

Convergence properties :

1. STA algorithm produces a stationary point for problem P_1.
2. The solution of P_1 obtained by STA converges to the solution of P_0.
3. Convergence rate of STA algorithm.

Idea : consider $\mathbf{W}$, $\mathbf{H}$ and $\mathbf{D}^1$, (and $\mathbf{D}^0$) as variables and treat STA is as an *Inexact Block Coordinate Descent* (BCD) algorithm / *Block Successive Upper bound Minimization* (BSUM) : at each iteration on variable $\mathbf{W}$, we are not considering the original problem but an upper bound function (with the proximal term) $\|\mathbf{X}_i - w_i h_i\|_F^2 + \lambda \mathbf{D}_{ii}^1 + \frac{\gamma}{2}\|w_i - w_i^-\|_2^2$, notice that the inexactness is also contributed by the eigen-gap introduced by $\mu_i \geq \mu_{\min}$.

Extension : accelerated coordinate descent ... (1/3)

The "simplified" STA algorithm (the updates of $\mathbf{D}$ and $\mathbf{H}$ are hidden here)

Algorithm 3 STA with cyclic indexing

for $k = 1$ to itermax **do**
    for $i = 1$ to r **do**
        Update w_i by doing something
    end for
end for

STA is a Inexact BCD with cycling indexing. That is, w_i is selected according to $i = 1, 2, ..., r, 1, 2, ..., r, ...$

In fact, cyclic indexing is not optimal. Acceleration can be made on using random indexing and extrapolation. The next page will discuss accelerated STA that, **in expectation**, converges faster.

Extension : accelerated coordinate descent ... (2/3)

Algorithm 4 Accelerated STA with random indexing

Set $v_i^k = w_i$ for $i = 1, 2, ..., r$.
for $k = 1$ to itermax **do**
    Pick $i = i_k$ by random (with uniform probability)
    $y_i^k = a_k v_i^k + (1 - a_k) w_i^k$
    $w_i^{k+1} = y_i^k - \dfrac{1}{L_{i_k}} \nabla_{i_k} f(y_i^k)$
    $v_i^{k+1} = b_k v_i^k + (1 - b^k) y^k - \dfrac{c_k}{L_{i_k}} \nabla_{i_k} f(y_i^k)$
end for

where σ is strong convexity parameter and L_{i_k} is smoothness parameter of function $f(w_i)$. And parameters a, b, c are obtained by solving the following non-linear equations

$$c_k^2 - \frac{c_k}{r} = \left(1 - \frac{c_k \sigma}{r}\right) c_{k-1}^2, \; a_k = \frac{r - c_k \sigma}{c_k (r^2 - \sigma)}, \; b_k = 1 - \frac{c_k \sigma}{r}$$

Idea is similar to the "extrapolation induced acceleration" of the Nesterov's accelerated (full-)graident.

The "simplified" acceelrated STA algorithm :

Algorithm 5 Accelerated STA with random indexing

1: **for** $k = 1$ to itermax **do**
2: Pick $i = i_k$ by random
3: update w_i by doing something, together with two additional series v_i^k and y_i^k
4: **end for**

The trade off of faster convergence is the "randomness" : picking up repeated index is possible $i = 1, \mathbf{3}, \mathbf{3}, \mathbf{3}, 4, 2, 5, ...$

Solution is to use random shuffle instead of totally random. For example, $r = 3$, and

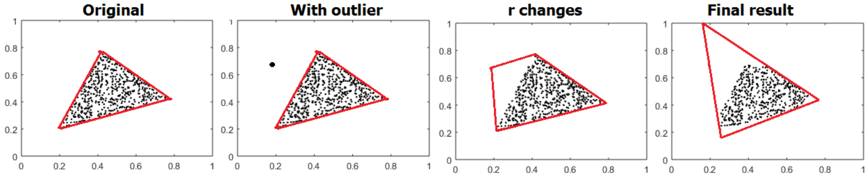
$$i = [1,3,2],[3,2,1],[3,1,2],[1,2,3],[2,1,3],...$$

But the analysis of the convergence property becomes complicated.

Extension : noise, outlier and robustness (1/3)

Formulation (P_0) and (P_1) are sensitive to **outlier** and **noise**.

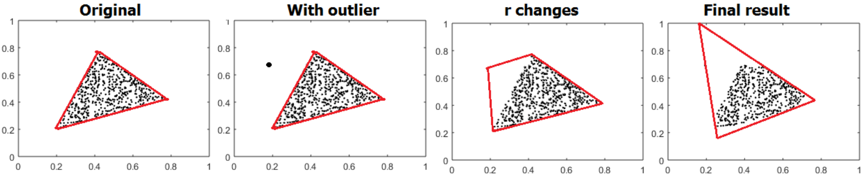
- **Outlier** : a single outlier **can kill** the algorithm.



Extension : noise, outlier and robustness (1/3)

Formulation (P_0) and (P_1) are sensitive to **outlier** and **noise**.

- **Outlier** : a single outlier **can kill** the algorithm.



- Solution : **robust norm** on data fitting:

$$\|\mathbf{X} - \mathbf{WH}\|_F^2 \xrightarrow{\text{changed to}} \|\mathbf{X} - \mathbf{WH}\|_\phi^2$$

- Examples of ϕ
  - $\|\mathbf{X} - \mathbf{WH}\|_{2-1}^2 + \lambda\,\mathrm{tr}(\mathbf{D}^1\mathbf{W}^\top\mathbf{W} + \mathbf{D}^0)$ (L_{2-1} norm, column robust)
  - $\|\mathbf{X} - \mathbf{WH}\|_1^2 + \lambda\,\mathrm{tr}(\mathbf{D}^1\mathbf{W}^\top\mathbf{W} + \mathbf{D}^0)$ (L_1 norm, matrix robust)
  - $\|\mathbf{X} - \mathbf{WH}\|_p^2 + \lambda\,\mathrm{tr}(\mathbf{D}^1\mathbf{W}^\top\mathbf{W} + \mathbf{D}^0)$ (L_p norm, tunable)
  - $\|\mathbf{X} - \mathbf{WH}\|_B^2 = \sum_{ij} B_{ij}(\mathbf{X} - \mathbf{WH})_{ij}^2 = \|B \odot (\mathbf{X} - \mathbf{WH})\|_F^2$

Extension : noise, outlier and robustness (2/3)

Formulation (P_0) and (P_1) are sensitive to **outlier** and **noise**.

- **Noise** : F-norm is for additive Gaussian noise.

Extension : noise, outlier and robustness (2/3)

Formulation (P_0) and (P_1) are sensitive to **outlier** and **noise**.

- **Noise** : F-norm is for additive Gaussian noise.
- Other noises and corresponding denoising norms :
  - Kullback Leibler divergence for Poisson noise
  - Itakura Saito divergence for Gamma Expoential noise
  - Laplacian / Double Expoential noise , Uniform noise, Lorentz noise

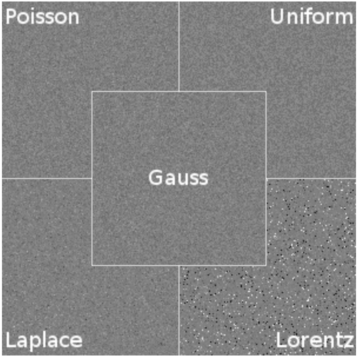


Figure: Noises. Source : https://gimper.net/resources/noise-generator.576

Extension : noise, outlier and robustness (3/3)

Formulations (P_0) and (P_1) are sensitive to **outlier** and **noise**.
Solution - 1 : solve $\|\mathbf{X} - \mathbf{W}\mathbf{H}\|_p^2 + \lambda \operatorname{tr}(\mathbf{D}^1 \mathbf{W}^\top \mathbf{W} + \mathbf{D}^0)$ by **Iterative Reweighted Least Squares** (IRLS)
Idea :

$$\|x\|_p = \Big(\sum_i |x_i|^p \Big)^{1/p} = \Big(\sum_i w_i^2 |x_i|^{\color{red}{2}} \Big)^{\color{red}{1/2}}$$

where weight $w_i = x_i^{\frac{p-2}{2}}$. For application w_i can be set as x_i^{-}.

i.e. Approximate $\|\mathbf{X} - \mathbf{W}\mathbf{H}\|_p^2$ as a weighted L_2 norm problem.

Extension : noise, outlier and robustness (3/3)

Formulations (P_0) and (P_1) are sensitive to **outlier** and **noise**.
Solution - 1 : solve $\|\mathbf{X} - \mathbf{W}\mathbf{H}\|_p^2 + \lambda \operatorname{tr}(\mathbf{D}^1\mathbf{W}^\top\mathbf{W} + \mathbf{D}^0)$ by **Iterative Reweighted Least Squares** (IRLS)
Idea :

$$\|x\|_p = \Big(\sum_i |x_i|^p\Big)^{1/p} = \Big(\sum_i w_i^2 |x_i|^{2}\Big)^{1/2}$$

where weight $w_i = x_i^{\frac{p-2}{2}}$. For application w_i can be set as x_i^{-}.

i.e. Approximate $\|\mathbf{X} - \mathbf{W}\mathbf{H}\|_p^2$ as a weighted L_2 norm problem.

Or, approximate matrix L_1 norm by IRLS :

$$\begin{aligned}\|\mathbf{X} - \mathbf{W}\mathbf{H}\|_B^2 &= \sum_{ij} B_{ij}(\mathbf{X} - \mathbf{W}\mathbf{H})_{ij}^2 \\ &\approx \|\mathbf{X} - \mathbf{W}\mathbf{H}\|_1^2\end{aligned}$$

for $B_{ij} = \dfrac{1}{\|\mathbf{X} - \mathbf{W}\mathbf{H}\|_{ij} + \epsilon}$

Extension : noise, outlier and robustness (3/3)

Formulations (P_0) and (P_1) are sensitive to **outlier** and **noise**.
Solution - 2 : use L_{2-1} norm, or more specific :

$$\sum_{j=1}^{n} \frac{1}{2}\big(\|\mathbf{X}(:,j) - \mathbf{W}\mathbf{H}(:,j)\|_2^2 + \epsilon\big)^{\frac{p}{2}} + \lambda \, \mathrm{tr}(\mathbf{D}^1\mathbf{W}^\top\mathbf{W} + \mathbf{D}^0)$$

where $\epsilon > 0$ is smoothness constant and $p \in (0 \; 2]$ is robustness parameter.

FGM cannot be used directly on this formulation, need some modifications.

Small summary

$$\|\mathbf{X} - \mathbf{W}\mathbf{H}\|_F^2$$

$$\|\mathbf{X} - \mathbf{W}\mathbf{H}\|_F^2 + \lambda \log\det(\mathbf{W}^\top \mathbf{W} + \delta I)$$

$$\|\mathbf{X} - \mathbf{W}\mathbf{H}\|_\phi^2$$

$$\|\mathbf{X} - \mathbf{W}\mathbf{H}\|_F^2 + \lambda \operatorname{tr}(\mathbf{W}^\top \mathbf{W} + (\delta - 1)\mathbf{I}_r)$$

$$\|\mathbf{X} - \mathbf{W}\mathbf{H}\|_F^2 + \lambda \operatorname{tr}(\mathbf{D}^1 \mathbf{W}^\top \mathbf{W} + \mathbf{D}^0)$$

$$\sum_i \|\mathbf{X}_i - w_i h_i\|_F^2 + \lambda \mathbf{D}^1_{ii}\|w_i\|_2^2 + \frac{\gamma}{2}\|w_i - w_i^-\|_2^2$$

Further discussions

Standard open problems

- How to tune λ
  - small noise $\lambda \to 0$
  - large noise $\lambda \to \infty$
  - $\lambda(N) =?$
- How to find r (in real world application you don't know the r !)

Other directions

- On solving $\mathbf{H}$
- On super-big data
- **Parallelism**: divide-and-conquer / decompose-and-recombine.
- Acceleration by weighted formulation

Further discussions : how to tune λ

A very difficult problem. How to tune λ dynamically within the iterations ?? That is, find an expression in the form as :

$$\lambda = \lambda(\mathbf{X}, \mathbf{W}_k, \mathbf{H}_k, k)$$

where k is the current number of iteration.
Don't expect I can give a global solution ! A rough idea of approaches :

- Simulated Annealing
- Dynamic approach
- Hybrid approach

Pros : easy to implement
Cons : very hard to establish theoretical convergence gurantee

Simulated Annealing parameter tuning of λ

- Annealing (metallurgy) $=$ heat treatment of metal
- At the beginning, start with very high temperature to make coarse adjustment of the metal (hammering)
- Temperature is gradually decrease in the process, and graudally moving from coarse adjustment to fine adjustment
- Finally the metal is cooled down

Simulated Annealing parameter tuning of λ

- Annealing (metallurgy) = heat treatment of metal
- At the beginning, start with very high temperature to make coarse adjustment of the metal (hammering)
- Temperature is gradually decrease in the process, and graudally moving from coarse adjustment to fine adjustment
- Finally the metal is cooled down

On the problem $f(\mathbf{W}, \mathbf{H}) + \lambda G(\mathbf{W})$,

- high temperature = starting with very large λ
- Coarse adjustment of the metal = rotation of the convex hull
- Temperature is gradually decrease = gradually decrease the value of λ
- Fine adjustment = growth of convex hull
- metal is cooled down = λ_k is very close to zero
- (Reminder to myself : refer to the "rotate.gif")

Some discussions on the tuning

- Very primitive, not robust, non-determinstic
- Only works with cases that $\mathbf{W}_0$ is known : is $\log\det(\mathbf{W}^\top\mathbf{W} + \delta\mathbf{I}_r)$ really a good estimator of $\|\mathbf{W}_0 - \hat{\mathbf{W}}\|_F$? How about $\det\mathbf{W}^\top\mathbf{W}$?

$\Longrightarrow$ Compare the models $\log\det(\mathbf{W}^\top\mathbf{W} + \delta\mathbf{I}_r)$ with $\det\mathbf{W}^\top\mathbf{W}$

Equivalent problems:

$$
\begin{aligned}
(P_0) \quad & \text{min} \quad \|\mathbf{X} - \mathbf{W}\mathbf{H}\|_F^2 + \lambda \log\det(\mathbf{W}^\top\mathbf{W} + \delta\mathbf{I}_r), \\
(P_1) \quad & \text{min.} \quad \mathrm{tr}(\mathbf{D}^1\mathbf{W}^\top\mathbf{W} + \mathbf{D}^0) \text{ s.t. } \|\mathbf{X} - \mathbf{W}\mathbf{H}\|_F^2 \leq \epsilon
\end{aligned}
$$

For every ϵ in P_1, there exists a λ in P_0 that both of them share the same solution. Solving P_1 does not involve parameter tuning.

Further improving STA : on $\mathbf{H}$ (1/3)

Recall the STA algorithm (with first 3 lines remoeved)

---

for $K = 1$ to itermax **do**
    for $k = 1$ to itermax **do**
        $P = \mathbf{X}\mathbf{H}^\top$ and $Q = \mathbf{H}\mathbf{H}^\top$.
        for $i = 1$ to r **do**
            $w_i = \frac{\left[P_i - \sum_{j=1}^{i-1} w_j Q_{ji} - \sum_{j=i+1}^{r} w_j^- Q_{ji} + \gamma w_i^-\right]_+}{Q_{ii} + \lambda \mathbf{D}^1_{ii} + \frac{\gamma}{2}}$
            $\mu_i \leftarrow$ svd$(\mathbf{W}^\top\mathbf{W} + \delta\mathbf{I}_r)$ and $\mathbf{D}^1 =$Diag$(\mu_{\min}^{-1})$
        end for
        Update $\mathbf{H}$ by FGM with $\mathbf{X}, \mathbf{W}, \mathbf{H}$
    end for
end for

This line is to solve

$$\mathbf{H} \leftarrow \underset{\mathbf{H}\geq 0,\, \mathbf{1}_r^\top \mathbf{H} \leq \mathbf{1}_n}{\arg\min} \|\mathbf{X} - \mathbf{W}\mathbf{H}\|_F^2 + \underbrace{\lambda \log\det(\mathbf{W}^\top\mathbf{W} + \delta\mathbf{I}_r)}_{\text{a constant for } \mathbf{H}}.$$

Further improving STA : on $\mathbf{H}$ (2/3)

Is the FGM (Fast gradient method on constrainted least sqaure on unit simplex) really the best ?

---

for $K = 1$ to itermax **do**

 for $k = 1$ to itermax **do**

  $P = \mathbf{X}\mathbf{H}^\top$ and $Q = \mathbf{H}\mathbf{H}^\top$.

  for $i = 1$ to r **do**

   $w_i = \frac{\left[P_i - \sum_{j=1}^{i-1} w_j Q_{ji} - \sum_{j=i+1}^{r} w_j^- Q_{ji} + \gamma w_i^-\right]_+}{Q_{ii} + \lambda \mathbf{D}^1_{ii} + \frac{\gamma}{2}}$

   $\mu_i \leftarrow$ svd$(\mathbf{W}^\top\mathbf{W} + \delta\mathbf{I}_r)$ and $\mathbf{D}^1 =$Diag$(\mu_{\min}^{-1})$

  end for

  Update $\mathbf{H}$ by FGM with $\mathbf{X}, \mathbf{W}, \mathbf{H}$

 end for

end for

Further improving STA : on $\mathbf{H}$ (3/3)

Currently, yes.

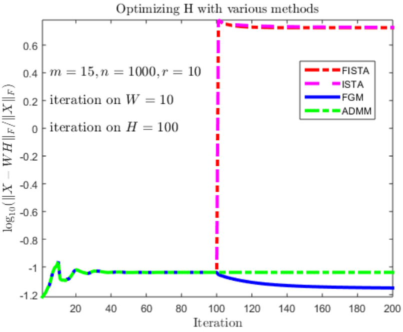


Figure: Four methods on H.

* primal algorithm vs primal-dual algorithm.

Further issues : on r

We assumed r is known, it is only true for synthetic experiment. In real application, no one know the true r.

Idea : if input r is larger than the true r, when the minimum volume is 'achieved', then there will be (almost-)colinear columns in $\mathbf{W} \implies$ a way to auto-detect r!!

Reminder to me : run the large_r.gif in chrome.

Further issues : acceleration by weighted sum formulation

Since

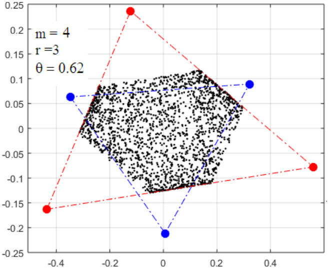
$$\|\mathbf{X} - \mathbf{WH}\|_F^2 = \sum_j \|\mathbf{X}(:,j) - \mathbf{WH}(:,j)\|_2^2.$$

What if the data fitting terms becomes

$$\sum_j \alpha_i \|\mathbf{X}(:,j) - \mathbf{WH}(:,j)\|_2^2,$$

where α_i are weights.

Idea : increase the weight on the points that $\notin$ conv$(\mathbf{W})$ to speed up the fitting of the vertices of the next iteration



Last page - summary

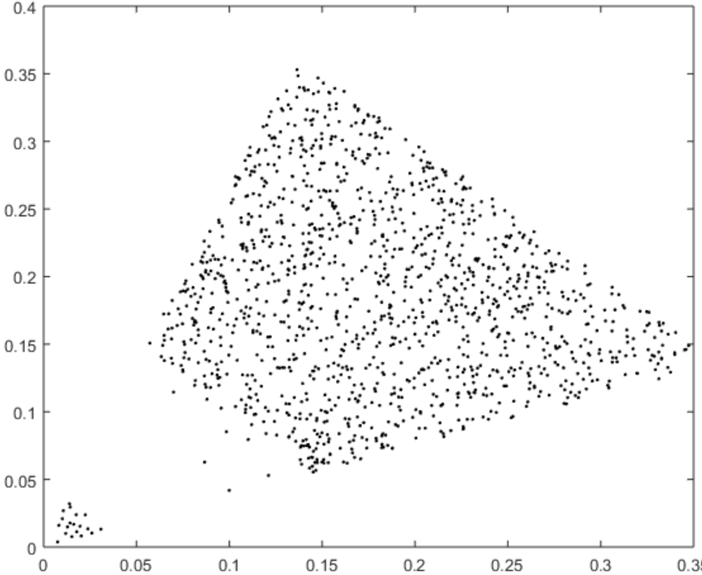
- Introduce / review NMF $\|\mathbf{X} - \mathbf{W}\mathbf{H}\|_F^2$
- Minimum volume NMF $\|\mathbf{X} - \mathbf{W}\mathbf{H}\|_F^2 + \lambda \log\det(\mathbf{W}^\top\mathbf{W} + \delta\mathbf{I}_r)$
- Successive Trace Approximation $\mathrm{tr}(\mathbf{D}^1\mathbf{W}^\top\mathbf{W} + \mathbf{D}^0)$
- The STA algorithm and refinments
- BCD acceleration by randomization (on index)
- Convergence of STA (just rough idea)
- Robust STA (just rough idea) : $\|\mathbf{X} - \mathbf{W}\mathbf{H}\|_\phi$, $\phi \in \{1 \leq p \leq 2, 2-1\}$, Iterative reweighted least sqaures
- Experiments : some rough illustrations
- Some open / unsolved problems and further refinments

Slides (and code) avaliable at *angms.science*

– END OF PRESENTATION –

Extra - 1. What if some elements of $\mathbf{H}$ is very very small

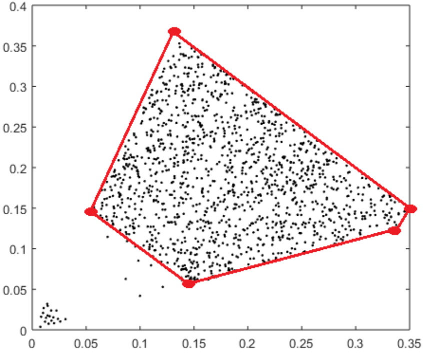
If some $\mathbf{H}$ of the data are very small, convex hull looks like conical hull



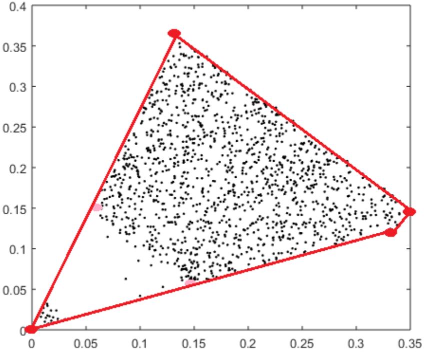
Extra - 1. What if some elements of $\mathbf{H}$ is very very small

Solution-1 See them as outliers
Their $\mathbf{H}_{ij}$ small $\implies$ norm small $\implies$ error small $\implies$ ok if only a few of them

Solution-2 Augmenting $\mathbf{W} = [\mathbf{W}' \ \mathbf{a}]$ where $\mathbf{a}$ is a small vector.
Note : r changes, and $\mathbf{a}$ cannot be $\mathbf{0}$ as matrix $[\mathbf{W} \ \mathbf{0}]$ is not full rank



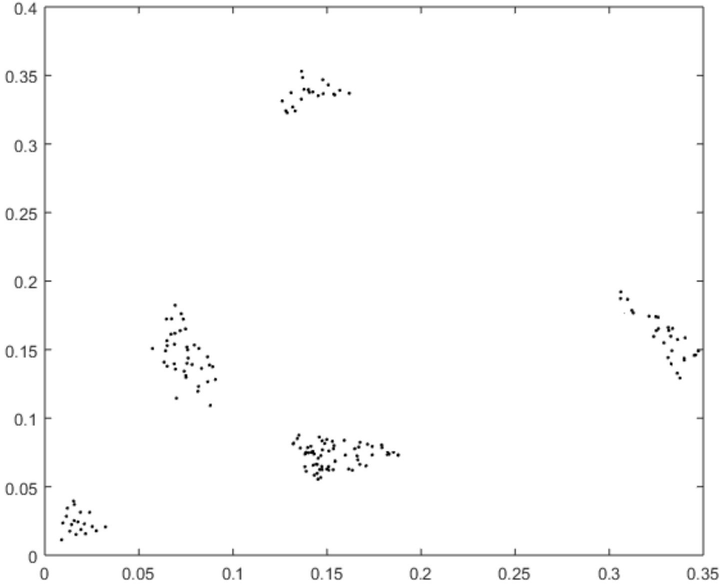
W



$[W' \ \mathbf{a}]$

Extra - 2. What if data are clustered

What if :



Extra - 2. What if data are clustered

Still works, but other method will be better (e.g. some clustering method such as K-means)