Nonnegative Matrix Factorization

Complexity, Algorithms and Applications

Nicolas Gillis†
Advisor: François Glineur
Université catholique de Louvain

†Université de Mons
Department of Mathematics and Operational Research
Belgium

We are here

Mons is here

1992 MAGELLAN Geographic SMS Santa Barbara, CA (800) 929-4627
Nonnegative Matrix Factorization (NMF)

Given a matrix $M \in \mathbb{R}^{p \times n}$ and a factorization rank $r \ll \min(p, n)$, find $U \in \mathbb{R}^{p \times r}$ and $V \in \mathbb{R}^{r \times n}$ such that

$$
\min_{U \geq 0, V \geq 0} ||M - UV||_F^2 = \sum_{i,j} (M - UV)_{ij}^2.
$$

(NMF)

NMF is a linear dimensionality reduction technique for nonnegative data:

$$
M(:, j) \approx \sum_{k=1}^{r} U(:, k) V(k, j) \quad \text{for all } j.
$$

Why nonnegativity?

→ **Interpretability**: Nonnegativity constraints lead to easily interpretable factors (and a sparse and part-based representation).

→ **Many applications**: image processing, text mining, recommender systems, community detection, clustering, etc.
Nonnegative Matrix Factorization (NMF)

Given a matrix \( M \in \mathbb{R}^{p \times n} \) and a factorization rank \( r \ll \min(p, n) \), find \( U \in \mathbb{R}^{p \times r} \) and \( V \in \mathbb{R}^{r \times n} \) such that

\[
\min_{U \geq 0, V \geq 0} \| M - UV \|_F^2 = \sum_{i,j} (M - UV)_{ij}^2. \tag{NMF}
\]

NMF is a linear dimensionality reduction technique for nonnegative data:

\[
M(:, j) \approx \sum_{k=1}^r U(:, k) V(k, j) \geq 0 \quad \forall j.
\]

Why nonnegativity?

→ **Interpretability**: Nonnegativity constraints lead to easily interpretable factors (and a sparse and part-based representation).

→ **Many applications**: image processing, text mining, recommender systems, community detection, clustering, etc.
Nonnegative Matrix Factorization (NMF)

Given a matrix $M \in \mathbb{R}^{p \times n}$ and a factorization rank $r \ll \min(p, n)$, find $U \in \mathbb{R}^{p \times r}$ and $V \in \mathbb{R}^{r \times n}$ such that

$$\min_{U \geq 0, V \geq 0} \| M - UV \|_F^2 = \sum_{i,j} (M - UV)_{ij}^2.$$  

(NMF)

NMF is a linear dimensionality reduction technique for nonnegative data:

$$M(:, j) \approx \sum_{k=1}^{r} U(:, k) \cdot V(k, :) \quad \text{for all } j.$$  

Why nonnegativity?

→ **Interpretability**: Nonnegativity constraints lead to easily interpretable factors (and a sparse and part-based representation).

→ **Many applications.** image processing, text mining, recommender systems, community detection, clustering, etc.
Example: blind hyperspectral unmixing

Figure: Urban hyperspectral image with 162 spectral bands and 307-by-307 pixels.
Example: blind hyperspectral unmixing

- Basis elements allow to recover the different materials;
- Weights allow to know which pixel contains which material.
Example: blind hyperspectral unmixing

- Basis elements allow to recover the different materials;
- Weights allow to know which pixel contains which material.
Example: blind hyperspectral unmixing

- Basis elements allow to recover the different materials;
- Weights allow to know which pixel contains which material.
Example: blind hyperspectral unmixing

\[
\mathbf{M}(::, j) \approx \sum_{k=1}^{\infty} \mathbf{U}(::, k) \mathbf{V}(k, j)
\]

*spectral signature of jth pixel*

**Figure**: Decomposition of the Urban dataset.
Example: blind hyperspectral unmixing

\[ M(:,j) \approx \sum_{k=1} \underbrace{U(:,k)}_{\text{spectral signature of } k\text{th endmember}} \underbrace{V(k,j)}_{\text{ }}. \]

**Figure:** Decomposition of the Urban dataset.
Example: blind hyperspectral unmixing

\[ M(\cdot,j) \approx \sum_{k=1}^{\cdot} U(\cdot,k) V(k,j) \]

- **M(\cdot,j)**: spectral signature of \( j \)th pixel
- **U(\cdot,k)**: spectral signature of \( k \)th endmember
- **V(k,j)**: abundance of \( k \)th endmember in \( j \)th pixel

**Figure**: Decomposition of the Urban dataset.
Outline

1. Algorithms and Applications
   - General framework for NMF algorithms
   - Solving NMF sequentially with underapproximations

2. Complexity and Bounds
   - Nonnegative rank
   - rank-one subproblems, weights and missing data
How can we ‘solve’ NMF problems?

Given a matrix $M \in \mathbb{R}^{m \times n}_+$ and a factorization rank $r \in \mathbb{N}$:

$$
\min_{U \in \mathbb{R}^{m \times r}_+, V \in \mathbb{R}^{r \times n}_+} \|M - UV\|_F^2 = \sum_{i,j} (M - UV)_{ij}^2 . \tag{NMF}
$$

0. Initialize $(U, V)$. Then, alternatively update $U$ and $V$:

1. Update $V \approx \operatorname{argmin}_{X \geq 0} \|M - UX\|_F^2 . \tag{NNLS}$
2. Update $U \approx \operatorname{argmin}_{Y \geq 0} \|M - YV\|_F^2 . \tag{NNLS}$

HALS algorithm: Use block-coordinate descent on NNLS subproblems $\rightarrow$ closed-form solutions for the columns of $U$ and rows of $V$:

$$
U^*_k = \operatorname{argmin}_{U:k \geq 0} \|R_k - U:k V_k:\|_F^2 = \max \left( 0, \frac{R_k V_k^T}{\|V_k:\|_2^2} \right) \quad \forall k,
$$

where $R_k = M - \sum_{j \neq k} U:j V_j:,$ and similarly for $V$.

HALS can be accelerated ordering updates efficiently.

How can we ‘solve’ NMF problems?

Given a matrix $M \in \mathbb{R}^{m \times n}$ and a factorization rank $r \in \mathbb{N}$:

$$\min_{U \in \mathbb{R}^{m \times r}, V \in \mathbb{R}^{r \times n}} ||M - UV||_F^2 = \sum_{i,j}(M - UV)_{i,j}^2. \quad \text{(NMF)}$$

0. Initialize $(U, V)$. Then, alternatively update $U$ and $V$:

1. Update $V \approx \arg\min_{X \geq 0} ||M - UX||_F^2$. \quad \text{(NNLS)}

2. Update $U \approx \arg\min_{Y \geq 0} ||M - YV||_F^2$. \quad \text{(NNLS)}

**HALS algorithm**: Use block-coordinate descent on NNLS subproblems $\rightarrow$ closed-form solutions for the columns of $U$ and rows of $V$:

$$U^*_k = \arg\min_{U_{:k} \geq 0} ||R_k - U_{:k}V_{k:}||_F^2 = \max \left(0, \frac{R_kV_{k:}^T}{||V_{k:}||_2^2}\right) \quad \forall k,$$

where $R_k = M - \sum_{j \neq k} U_{:j}V_{j:}$, and similarly for $V$.

HALS can be accelerated ordering updates efficiently.

How can we ‘solve’ NMF problems?

Given a matrix $M \in \mathbb{R}^{m \times n}$ and a factorization rank $r \in \mathbb{N}$:

$$
\min_{U \in \mathbb{R}^{m \times r}_+, V \in \mathbb{R}^{r \times n}_+} \|M - UV\|^2_F = \sum_{i,j} (M - UV)_{ij}^2. \quad \text{(NMF)}
$$

0. Initialize $(U, V)$. Then, alternatively update $U$ and $V$:

1. Update $V \approx \arg\min_{X \geq 0} \|M - UX\|^2_F$. \quad \text{(NNLS)}

2. Update $U \approx \arg\min_{Y \geq 0} \|M - YV\|^2_F$. \quad \text{(NNLS)}

**HALS algorithm**: Use block-coordinate descent on NNLS subproblems → closed-form solutions for the columns of $U$ and rows of $V$:

$$
U^*_{:k} = \arg\min_{U_{:k} \geq 0} \|R_k - U_{:k}V_{k:}\|^2_F = \max \left( 0, \frac{R_k V_{k:}^T}{\|V_{k:}\|_2^2} \right) \quad \forall k,
$$

where $R_k \doteq M - \sum_{j \neq k} U_{:j}V_{j:}$, and similarly for $V$.

HALS can be accelerated ordering updates efficiently.

Drawbacks of standard NMF Algorithms

1. The optimal solution is, in most cases, non-unique and the problem is ill-posed. Many variants of NMF impose additional constraints (e.g., sparsity, smoothness, spatial information, etc.).

2. In practice, it is difficult to choose the factorization rank (in general, trial and error approach or estimation using the SVD).

A possible way to overcome these drawbacks is to use underapproximation constraints to solve NMF sequentially.
Drawbacks of standard NMF Algorithms

1. The optimal solution is, in most cases, non-unique and the problem is ill-posed. Many variants of NMF impose additional constraints (e.g., sparsity, smoothness, spatial information, etc.).

2. In practice, it is difficult to choose the factorization rank (in general, trial and error approach or estimation using the SVD).

A possible way to overcome these drawbacks is to use underapproximation constraints to solve NMF sequentially.
Drawbacks of standard NMF Algorithms

1. The optimal solution is, in most cases, non-unique and the problem is ill-posed. Many variants of NMF impose additional constraints (e.g., sparsity, smoothness, spatial information, etc.).

2. In practice, it is difficult to choose the factorization rank (in general, trial and error approach or estimation using the SVD).

A possible way to overcome these drawbacks is to use underapproximation constraints to solve NMF sequentially.
**Nonnegative Matrix Underapproximation (NMU)**

It is possible to solve NMF **sequentially**, solving at each step

\[
\min_{u \geq 0, v \geq 0} \| M - uv^T \|_F^2 \quad \text{such that} \quad uv^T \leq M \iff M - uv^T \geq 0.
\]

NMU is yet another **linear dimensionality reduction** technique. However,

- As PCA/SVD, it is sequential and is well-posed.
- As NMF, it leads to a separation by parts. Moreover the additional underapproximation constraints enhance this property.
- In the presence of pure-pixels, the NMU recursion is able to detect materials individually.


Nonnegative Matrix Underapproximation (NMU)

It is possible to solve NMF sequentially, solving at each step

\[
\min_{u \geq 0, v \geq 0} \| M - uv^T \|_F^2 \quad \text{such that} \quad uv^T \leq M \iff M - uv^T \geq 0.
\]

NMU is yet another linear dimensionality reduction technique. However,

- As PCA/SVD, it is sequential and is well-posed.
- As NMF, it leads to a separation by parts. Moreover the additional underapproximation constraints enhance this property.
- In the presence of pure-pixels, the NMU recursion is able to detect materials individually.


Nonnegative Matrix Underapproximation (NMU)

It is possible to solve NMF sequentially, solving at each step

$$\min_{u \geq 0, v \geq 0} \| M - uv^T \|_F^2$$

such that

$$uv^T \leq M \iff M - uv^T \geq 0.$$ 

NMU is yet another linear dimensionality reduction technique. However,

- As PCA/SVD, it is sequential and is well-posed.
- As NMF, it leads to a separation by parts. Moreover the additional underapproximation constraints enhance this property.
- In the presence of pure-pixels, the NMU recursion is able to detect materials individually.

Nonnegative Matrix Underapproximation (NMU)

It is possible to solve NMF sequentially, solving at each step

$$\min_{u \geq 0, v \geq 0} \| M - uv^T \|_F^2 \quad \text{such that} \quad uv^T \leq M \iff M - uv^T \geq 0.$$ 

NMU is yet another linear dimensionality reduction technique. However,

- As PCA/SVD, it is sequential and is well-posed.
- As NMF, it leads to a separation by parts. Moreover the additional underapproximation constraints enhance this property.
- In the presence of pure-pixels, the NMU recursion is able to detect materials individually.


NMU on the Urban dataset
NMU on the Urban dataset
NMU on the Urban dataset
NMU on the Urban dataset
NMU on the Urban dataset
Identifying Lighting Orientations with NMU

A static scene is illuminated from many directions.

NMU is able to detect the different lighting orientations.

NMU has also been successfully used in text mining for anomaly detection, and in image processing for segmentation of medical images.
Identifying Lighting Orientations with NMU

A static scene is illuminated from many directions.

NMU is able to detect the different lighting orientations.

NMU has also been successfully used in text mining for anomaly detection, and in image processing for segmentation of medical images.
Identifying Lighting Orientations with NMU

A static scene is illuminated from many directions.

NMU is able to detect the different lighting orientations. NMU has also been successfully used in text mining for anomaly detection, and in image processing for segmentation of medical images.
Outline

1. Algorithms and Applications
   - General framework for NMF algorithms
   - Solving NMF sequentially with underapproximations

2. Complexity and Bounds
   - Nonnegative rank
   - Rank-one subproblems, weights and missing data
What is the nonnegative rank?

The nonnegative rank of a nonnegative matrix $M \in \mathbb{R}^{m \times n}_+$ is the minimum number $r$ of nonnegative rank-one factors needed to reconstruct $M$:

$$M = \sum_{i=1}^{r} u_i v_i^T, \quad u_i \in \mathbb{R}^m_+, v_i \in \mathbb{R}^n_+,$$

that is, the minimum $r$ such that an exact NMF exists:

$$M = UV = \sum_{i=1}^{r} U_{i:} V_{i:}, \quad U \geq 0, V \geq 0.$$

The nonnegative rank of $M$ is denoted $\text{rank}_+(M)$. Clearly,

$$\text{rank}(M) \leq \text{rank}_+(M) \leq \min(m, n).$$
What is the nonnegative rank?

The nonnegative rank of a nonnegative matrix $M \in \mathbb{R}^{m \times n}_+$ is the minimum number $r$ of nonnegative rank-one factors needed to reconstruct $M$:

$$M = \sum_{i=1}^{r} u_i v_i^T, \quad u_i \in \mathbb{R}^m_+, v_i \in \mathbb{R}^n_+,$$

that is, the minimum $r$ such that an exact NMF exists:

$$M = UV = \sum_{i=1}^{r} U_i V_i, \quad U \geq 0, V \geq 0.$$

The nonnegative rank of $M$ is denoted $\text{rank}_+(M)$. Clearly,

$$\text{rank}(M) \leq \text{rank}_+(M) \leq \min(m, n).$$
An amazing result

Let $\mathcal{P}$ be a polytope

$$\mathcal{P} = \{ x \in \mathbb{R}^k \mid b_i - A(i,:)x \geq 0 \text{ for } 1 \leq i \leq m \},$$

and let $v_j$'s ($1 \leq j \leq n$) be its vertices.

We define the $m$-by-$n$ slack matrix $S_\mathcal{P}$ of $\mathcal{P}$ as follows:

$$S_\mathcal{P}(i,j) = b_i - A(i,:)v_j \geq 0 \quad 1 \leq i \leq m, 1 \leq j \leq n.$$ 

An extended formulation of $\mathcal{P}$ is higher dimensional polyhedron $Q \subseteq \mathbb{R}^{k+p}$ that (linearly) projects onto $\mathcal{P}$. The minimum number of facets of such a polytope is called the extension complexity $\text{xp}(\mathcal{P})$ of $\mathcal{P}$.

**Theorem** (Yannakakis, 1991).

$$\text{rank}_+(S_\mathcal{P}) = \text{xp}(\mathcal{P}).$$

**Remark.** Other closely related problems in communication complexity, probability, graph theory.
An amazing result

Let $\mathcal{P}$ be a polytope

$$\mathcal{P} = \{x \in \mathbb{R}^k \mid b_i - A(i,:)x \geq 0 \text{ for } 1 \leq i \leq m\},$$

and let $v_j$'s ($1 \leq j \leq n$) be its vertices.

We define the $m$-by-$n$ slack matrix $S_{\mathcal{P}}$ of $\mathcal{P}$ as follows:

$$S_{\mathcal{P}}(i, j) = b_i - A(i,:)v_j \geq 0 \quad 1 \leq i \leq m, 1 \leq j \leq n.$$  

An extended formulation of $\mathcal{P}$ is higher dimensional polyhedron $Q \subseteq \mathbb{R}^{k+p}$ that (linearly) projects onto $\mathcal{P}$. The minimum number of facets of such a polytope is called the extension complexity $\text{xp}(\mathcal{P})$ of $\mathcal{P}$.

**Theorem** (Yannakakis, 1991).

$$\text{rank}_+(S_{\mathcal{P}}) = \text{xp}(\mathcal{P}).$$

**Remark.** Other closely related problems in communication complexity, probability, graph theory.
An amazing result

Let $\mathcal{P}$ be a polytope

$$\mathcal{P} = \{ x \in \mathbb{R}^k \mid b_i - A(i,:)x \geq 0 \text{ for } 1 \leq i \leq m \},$$

and let $v_j$’s ($1 \leq j \leq n$) be its vertices.

We define the $m$-by-$n$ slack matrix $S_\mathcal{P}$ of $\mathcal{P}$ as follows:

$$S_\mathcal{P}(i,j) = b_i - A(i,:)v_j \geq 0 \quad 1 \leq i \leq m, 1 \leq j \leq n.$$

An extended formulation of $\mathcal{P}$ is higher dimensional polyhedron $Q \subseteq \mathbb{R}^{k+p}$ that (linearly) projects onto $\mathcal{P}$. The minimum number of facets of such a polytope is called the extension complexity $\text{xp}(\mathcal{P})$ of $\mathcal{P}$.

Theorem (Yannakakis, 1991).

$$\text{rank}_+(S_\mathcal{P}) = \text{xp}(\mathcal{P}).$$

Remark. Other closely related problems in communication complexity, probability, graph theory.
An amazing result

Let $\mathcal{P}$ be a polytope

$$\mathcal{P} = \{ x \in \mathbb{R}^k \mid b_i - A(i,:)x \geq 0 \text{ for } 1 \leq i \leq m \},$$

and let $v_j$’s ($1 \leq j \leq n$) be its vertices.

We define the $m$-by-$n$ slack matrix $S_\mathcal{P}$ of $\mathcal{P}$ as follows:

$$S_\mathcal{P}(i,j) = b_i - A(i,:)v_j \geq 0 \quad 1 \leq i \leq m, 1 \leq j \leq n.$$  

An extended formulation of $\mathcal{P}$ is higher dimensional polyhedron $Q \subseteq \mathbb{R}^{k+p}$ that (linearly) projects onto $\mathcal{P}$. The minimum number of facets of such a polytope is called the extension complexity $\text{xp}(\mathcal{P})$ of $\mathcal{P}$.

**Theorem** (Yannakakis, 1991).

$$\text{rank}_+(S_\mathcal{P}) = \text{xp}(\mathcal{P}).$$

**Remark.** Other closely related problems in communication complexity, probability, graph theory.
An amazing result

Let \( P \) be a polytope

\[
P = \{ x \in \mathbb{R}^k \mid b_i - A(i,:)x \geq 0 \text{ for } 1 \leq i \leq m \},
\]

and let \( v_j \)'s (1 \leq j \leq n) be its vertices.

We define the \( m \)-by-\( n \) slack matrix \( S_P \) of \( P \) as follows:

\[
S_P(i, j) = b_i - A(i,:)v_j \geq 0 \quad 1 \leq i \leq m, 1 \leq j \leq n.
\]

An extended formulation of \( P \) is higher dimensional polyhedron \( Q \subseteq \mathbb{R}^{k+p} \) that (linearly) projects onto \( P \). The minimum number of facets of such a polytope is called the extension complexity \( \text{xp}(P) \) of \( P \).

**Theorem** (Yannakakis, 1991).

\[
\text{rank}_+(S_P) = \text{xp}(P).
\]

**Remark.** Other closely related problems in communication complexity, probability, graph theory.
The Hexagon

\[ S_P = \begin{pmatrix}
0 & 1 & 2 & 2 & 1 & 0 \\
0 & 0 & 1 & 2 & 2 & 1 \\
1 & 0 & 0 & 1 & 2 & 2 \\
2 & 1 & 0 & 0 & 1 & 2 \\
2 & 2 & 1 & 0 & 0 & 1 \\
1 & 2 & 2 & 1 & 0 & 0
\end{pmatrix} \]
The Hexagon

\[ S_P = \begin{pmatrix} 0 & 1 & 2 & 2 & 1 & 0 \\ 0 & 0 & 1 & 2 & 2 & 1 \\ 1 & 0 & 0 & 1 & 2 & 2 \\ 2 & 1 & 0 & 0 & 1 & 2 \\ 2 & 2 & 1 & 0 & 0 & 1 \\ 1 & 2 & 2 & 1 & 0 & 0 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 1/2 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1/2 & 0 \\ 0 & 0 & 1 & 0 & 1/2 \\ 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1/2 \end{pmatrix} \begin{pmatrix} 0 & 1 & 2 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 2 \\ 0 & 0 & 0 & 2 & 2 & 0 \\ 2 & 2 & 0 & 0 & 0 & 0 \end{pmatrix}, \]

with

\[ \text{rank}(S_P) = 3 \leq \text{rank}_+(S_P) = 5 \leq \min(m, n) = 6. \]
Lower Bounds for the Nonnegative Rank

It is difficult to compute the nonnegative rank (NP-hard). However, it is possible to compute lower bounds efficiently. For example,

\[ n = \# \text{ vertices}(\mathcal{P}) \leq 2^{r^+}. \]

What is the complexity of NMF?

Given a matrix \( M \in \mathbb{R}^{m \times n} \) and a factorization rank \( r \in \mathbb{N} \):

\[
\min_{U \in \mathbb{R}_{+}^{m \times r}, V \in \mathbb{R}_{+}^{r \times n}} ||M - UV||_F^2 = \sum_{i,j} (M - UV)_{ij}^2. \tag{NMF}
\]

- For \( r = 1 \): ‘easy’ [Perron-Frobenius and Eckart-Young theorems].
- \( \text{rank}(M) = 2 \): \( \text{rank}_{+}(M) = 2 \), ‘easy’.
- \( r \) part of the input: NP-hard (Vavasis, 2009).
- \( \text{rank}_{+}(M) = r \) not part of the input: polynomial \( \mathcal{O}(mn r^2) \) (Arora et al. 2012, Moitra 2013).
- \( \text{rank}(M) = k \) not part of the input: open problem.
What is the complexity of NMF?

Given a matrix $M \in \mathbb{R}^{m \times n}$ and a factorization rank $r \in \mathbb{N}$:

$$\min_{U \in \mathbb{R}_+^{m \times r}, V \in \mathbb{R}_+^{r \times n}} ||M - UV||_F^2 = \sum_{i,j} (M - UV)_{ij}^2. \quad \text{(NMF)}$$

- For $r = 1$: ‘easy’ [Perron-Frobenius and Eckart-Young theorems].
- $\text{rank}(M) = 2$: $\text{rank}_+(M) = 2$, ‘easy’.
- $r$ part of the input: NP-hard (Vavasis, 2009).
- $\text{rank}_+(M) = r$ not part of the input: polynomial $-\mathcal{O} \left( (mn)^{r^2} \right)$ (Arora et al. 2012, Moitra 2013).
- $\text{rank}(M) = k$ not part of the input: open problem.
What is the complexity of NMF?

Given a matrix $M \in \mathbb{R}^{m \times n}$ and a factorization rank $r \in \mathbb{N}$:

$$\min_{U \in \mathbb{R}_+^{m \times r}, V \in \mathbb{R}_+^{r \times n}} ||M - UV||^2_F = \sum_{i,j} (M - UV)_{ij}^2.$$  \hspace{1cm} \text{(NMF)}

- For $r = 1$: ‘easy’ [Perron-Frobenius and Eckart-Young theorems].
- $\text{rank}(M) = 2$: $\text{rank}_+(M) = 2$, ‘easy’.
- $r$ part of the input: \textbf{NP-hard} (Vavasis, 2009).
- $\text{rank}_+(M) = r$ not part of the input: polynomial $-\mathcal{O} \left((mn)^{r^2}\right)$ (Arora et al. 2012, Moitra 2013).
- $\text{rank}(M) = k$ not part of the input: open problem.
What is the complexity of NMF?

Given a matrix $M \in \mathbb{R}_{+}^{m \times n}$ and a factorization rank $r \in \mathbb{N}$:

$$\min_{U \in \mathbb{R}_{+}^{m \times r}, V \in \mathbb{R}_{+}^{r \times n}} \|M - UV\|_F^2 = \sum_{i,j} (M - UV)_{ij}^2. \quad \text{(NMF)}$$

- For $r = 1$: ‘easy’ [Perron-Frobenius and Eckart-Young theorems].
- $\text{rank}(M) = 2$: $\text{rank}_+(M) = 2$, ‘easy’.
- $r$ part of the input: NP-hard (Vavasis, 2009).
- $\text{rank}_+(M) = r$ not part of the input: polynomial $-\mathcal{O}\left((mn)^{r^2}\right)$ (Arora et al. 2012, Moitra 2013).
- $\text{rank}(M) = k$ not part of the input: open problem.
What about rank-one problems in NMF?

Solving NMF requires to finding \( r \) nonnegative rank-one factors \( U_k V_k \), each having to satisfy the following equality as well as possible

\[
U_k V_k \approx M - \sum_{j \neq k} U_j V_j = R_k \not\geq 0 \quad \forall k.
\]

These subproblems have the following form

\[
\min_{u \in \mathbb{R}^m, v \in \mathbb{R}^n} \| R - uv^T \|_F^2 \quad \text{such that } u \geq 0, v \geq 0.
\]

called rank-one nonnegative factorization.

Is this problem difficult?

If \( R \geq 0 \) : No.

If \( R \not\geq 0 \) : Surprisingly, yes.

G., Glineur, A Continuous Characterization of the Maximum-Edge Biclique Problem, JOGO '12.
What about rank-one problems in NMF?

Solving NMF requires to finding \( r \) nonnegative rank-one factors \( U_k V_k \), each having to satisfy the following equality as well as possible

\[
U_k V_k \approx M - \sum_{j \neq k} U_j V_j = R_k \not\equiv 0 \quad \forall k.
\]

These subproblems have the following form

\[
\min_{u \in \mathbb{R}^m, v \in \mathbb{R}^n} \|R - uv^T\|_F^2 \quad \text{such that } u \geq 0, v \geq 0.
\] (R1NF)

called rank-one nonnegative factorization.

Is this problem difficult?

If \( R \geq 0 \): No.
If \( R \not\equiv 0 \): Surprisingly, yes.

G., Glineur, A Continuous Characterization of the Maximum-Edge Biclique Problem, JOGO ’12.
What about rank-one problems in NMF?

Solving NMF requires to finding \( r \) nonnegative rank-one factors \( U_k V_k \), each having to satisfy the following equality as well as possible

\[
U_k V_k \approx M - \sum_{j \neq k} U_j V_j = R_k \geq 0 \quad \forall k.
\]

These subproblems have the following form

\[
\min_{u \in \mathbb{R}^m, v \in \mathbb{R}^n} \| R - uv^T \|_F^2 \quad \text{such that } u \geq 0, v \geq 0. \tag{R1NF}
\]

called rank-one nonnegative factorization.

Is this problem difficult?

If \( R \geq 0 \) : No.

If \( R \not\geq 0 \) : Surprisingly, yes.

G., Glineur, A Continuous Characterization of the Maximum-Edge Biclique Problem, JOGO '12.
What about rank-one problems in NMF?

Solving NMF requires to finding $r$ nonnegative rank-one factors $U_k V_k$, each having to satisfy the following equality as well as possible

$$U_k V_k \approx M - \sum_{j \neq k} U_j V_j = R_k \neq 0 \quad \forall k.$$ 

These subproblems have the following form

$$\min_{u \in \mathbb{R}^m, v \in \mathbb{R}^n} \| R - u v^T \|_F^2 \quad \text{such that } u \geq 0, v \geq 0. \quad \text{(R1NF)}$$

called rank-one nonnegative factorization.

Is this problem difficult?
If $R \geq 0$ : No.
If $R \nexists 0$ : Surprisingly, yes.

G., Glineur, A Continuous Characterization of the Maximum-Edge Biclique Problem, JOGO '12.
What about weights and missing data?

In some cases, some entries are missing/unknown.

For example, we would like to predict how much someone is going to like a movie based on its movie preferences (e.g., 1 to 5 stars):

Users

\[
\begin{bmatrix}
2 & 3 & 2 & ? & ? \\
? & 1 & ? & 3 & 2 \\
1 & ? & 4 & 1 & ? \\
5 & 4 & ? & 3 & 2 \\
? & 1 & 2 & ? & 4 \\
1 & ? & 3 & 4 & 3
\end{bmatrix}
\]

Movies

Huge potential in electronic commerce sites (movies, books, music, ...). Good recommendations will increase the propensity of a purchase.
What about weights and missing data?

In some cases, some entries are missing/unknown.

For example, we would like to predict how much someone is going to like a movie based on its movie preferences (e.g., 1 to 5 stars):

\[
\begin{bmatrix}
2 & 3 & 2 & ? & ? \\
? & 1 & ? & 3 & 2 \\
1 & ? & 4 & 1 & ? \\
5 & 4 & ? & 3 & 2 \\
? & 1 & 2 & ? & 4 \\
1 & ? & 3 & 4 & 3
\end{bmatrix}
\]

Huge potential in electronic commerce sites (movies, books, music, ...). Good recommendations will increase the propensity of a purchase.
Low-rank model for recommendation systems

The behavior of users is modeled using linear combination of 'feature' users (related to age, sex, culture, etc.)

\[
M(:, j) \approx \sum_{k=1}^{r} U(:, k) V(k, j)
\]

Or equivalently, movies ratings are modeled as linear combinations of 'feature' movies (related to the genres - child oriented, serious vs. escapist, thriller, romantic, actors, etc.).

\[
M(i, :) \approx \sum_{k=1}^{r} U(i, k) V(k, :)
\]
Low-rank model for recommendation systems

The behavior of users is modeled using linear combination of 'feature' users (related to age, sex, culture, etc.)

\[
M(:, j) \approx \sum_{k=1}^{r} U(:, k) V(k, j)
\]

Or equivalently, movies ratings are modeled as linear combinations of 'feature' movies (related to the genres - child oriented, serious vs. escapist, thriller, romantic, actors, etc.).

\[
M(i, :) \approx \sum_{k=1}^{r} U(i, k) V(k, :)
\]
**Example**

\[
M = \begin{pmatrix}
2 & 3 & 2 & ? & ? \\
? & 1 & ? & 3 & 2 \\
1 & ? & 4 & 1 & ? \\
5 & 4 & ? & 3 & 2 \\
? & 1 & 2 & ? & 4 \\
1 & ? & 3 & 4 & 3
\end{pmatrix}
\]

\[
\approx \begin{pmatrix}
0.5 & 0.6 & -0.1 \\
0.8 & -0.2 & -0.3 \\
0.8 & -0.7 & 0.6 \\
-2 & 2.3 & 1.8 \\
-0.2 & 0.3 & 0.9 \\
1 & -0.2 & -0.2
\end{pmatrix}
\begin{pmatrix}
1.7 & 2.1 & 3.7 & 5 & 4.1 \\
2.2 & 3.2 & 0.8 & 5 & 0.5 \\
2 & 0.6 & 2.6 & 0.9 & 5
\end{pmatrix} = UV
\]

\[
= \begin{pmatrix}
2 & 2.9 & 2.1 & 5.4 & 1.9 \\
0.3 & 0.9 & 2 & 2.7 & 1.7 \\
1 & -0.2 & 4 & 1 & 5.9 \\
5.3 & 4.2 & -0.9 & 3.1 & 2 \\
2.1 & 1.1 & 1.8 & 1.3 & 2.8 \\
0.9 & 1.3 & 3 & 3.8 & 3
\end{pmatrix}
\]
Example

\[ M = \begin{pmatrix} 2 & 3 & 2 & ? & ? \\ ? & 1 & ? & 3 & 2 \\ 1 & ? & 4 & 1 & ? \\ 5 & 4 & ? & 3 & 2 \\ ? & 1 & 2 & ? & 4 \\ 1 & ? & 3 & 4 & 3 \end{pmatrix} \approx \begin{pmatrix} 0.5 & 0.6 & -0.1 \\ 0.8 & -0.2 & -0.3 \\ 0.8 & -0.7 & 0.6 \\ -2 & 2.3 & 1.8 \\ -0.2 & 0.3 & 0.9 \\ 1 & -0.2 & -0.2 \end{pmatrix} \begin{pmatrix} 1.7 & 2.1 & 3.7 & 5 & 4.1 \\ 2.2 & 3.2 & 0.8 & 5 & 0.5 \\ 2 & 0.6 & 2.6 & 0.9 & 5 \end{pmatrix} = UV \]

\[ = \begin{pmatrix} 2 & 2.9 & 2.1 & 5.4 & 1.9 \\ 0.3 & 0.9 & 2 & 2.7 & 1.7 \\ 1 & -0.2 & 4 & 1 & 5.9 \\ 5.3 & 4.2 & -0.9 & 3.1 & 2 \\ 2.1 & 1.1 & 1.8 & 1.3 & 2.8 \\ 0.9 & 1.3 & 3 & 3.8 & 3 \end{pmatrix} \]
Example

\[ M = \begin{pmatrix}
2 & 3 & 2 & ? & ? \\
? & 1 & ? & 3 & 2 \\
1 & ? & 4 & 1 & ? \\
5 & 4 & ? & 3 & 2 \\
? & 1 & 2 & ? & 4 \\
1 & ? & 3 & 4 & 3
\end{pmatrix} \approx \begin{pmatrix}
0.5 & 0.6 & -0.1 \\
0.8 & -0.2 & -0.3 \\
0.8 & -0.7 & 0.6 \\
-2 & 2.3 & 1.8 \\
-0.2 & 0.3 & 0.9 \\
1 & -0.2 & -0.2
\end{pmatrix} \begin{pmatrix}
1.7 & 2.1 & 3.7 & 5 & 4.1 \\
2.2 & 3.2 & 0.8 & 5 & 0.5 \\
2 & 0.6 & 2.6 & 0.9 & 5
\end{pmatrix} = UV
\]

\[ = \begin{pmatrix}
2 & 2.9 & 2.1 & 5.4 & 1.9 \\
0.3 & 0.9 & 2 & 2.7 & 1.7 \\
1 & -0.2 & 4 & 1 & 5.9 \\
5.3 & 4.2 & -0.9 & 3.1 & 2 \\
2.1 & 1.1 & 1.8 & 1.3 & 2.8 \\
0.9 & 1.3 & 3 & 3.8 & 3
\end{pmatrix} \]
Complexity

Weighted low-rank approximation is the following problem

\[
\inf_{U \in \mathbb{R}^{m \times r}, V \in \mathbb{R}^{r \times n}} \| M - UV^T \|_W^2 = \sum_{ij} W_{ij} (M - UV^T)_{ij}^2 ,
\]

(WLRA)

where \( W \geq 0 \) is the weighting matrix. A zero in \( W \) represents a missing/unknown entry in \( M \).

For \( r = 1 \) and \( M \geq 0 \), this is equivalent to Weighted NMF.

Theorem

It is \( \mathcal{NP} \)-hard to find an approximate solution of WLRA, for any \( r \geq 1 \).

Other applications: computer vision, microarray data analysis, 2-D digital filter, etc.

G., Glineur, Low-Rank Matrix Approximation with Weights or Missing Data is NP-hard, SIMAX '11.
Complexity

Weighted low-rank approximation is the following problem

\[
\inf_{U \in \mathbb{R}^{m \times r}, V \in \mathbb{R}^{r \times n}} ||M - UV^T||^2_W = \sum_{ij} W_{ij} (M - UV^T)^2_{ij}, \quad \text{(WLRA)}
\]

where \( W \geq 0 \) is the weighting matrix. A zero in \( W \) represents a missing/unknown entry in \( M \).

For \( r = 1 \) and \( M \geq 0 \), this is equivalent to Weighted NMF.

**Theorem**

*It is \( \mathcal{NP} \)-hard to find an approximate solution of WLRA, for any \( r \geq 1 \).*

**Other applications:** computer vision, microarray data analysis, 2-D digital filter, etc.

G., Glineur, *Low-Rank Matrix Approximation with Weights or Missing Data is NP-hard*, SIMAX '11.
Complexity

Weighted low-rank approximation is the following problem

\[ \inf_{U \in \mathbb{R}^{m \times r}, \quad V \in \mathbb{R}^{r \times n}} ||M - UV^T||^2_W = \sum_{ij} W_{ij} (M - UV^T)_{ij}^2, \quad \text{(WLRA)} \]

where \( W \geq 0 \) is the weighting matrix. A zero in \( W \) represents a missing/unknown entry in \( M \).

For \( r = 1 \) and \( M \geq 0 \), this is equivalent to Weighted NMF.

**Theorem**

It is \( \mathcal{NP} \)-hard to find an approximate solution of WLRA, for any \( r \geq 1 \).

**Other applications:** computer vision, microarray data analysis, 2-D digital filter, etc.

G., Glineur, *Low-Rank Matrix Approximation with Weights or Missing Data is NP-hard*, SIMAX ’11.
Maximum-edge biclique problem

The reductions for NMU, R1NF and WLRA are from the maximum-edge biclique problem: Given two sets of objects interacting together (a bipartite graph), find highly connected subgroups.

This is the maximum-edge complete bipartite subgraph.

Applications in clustering: text mining, web community discovery . . .
Maximum-edge biclique problem

The reductions for NMU, R1NF and WLRA are from the maximum-edge biclique problem: Given two sets of objects interacting together (a bipartite graph), find highly connected subgroups.

This is the maximum-edge complete bipartite subgraph.

Applications in clustering: text mining, web community discovery...
Maximum-edge biclique problem

The reductions for NMU, R1NF and WLRA are from the maximum-edge biclique problem: Given two sets of objects interacting together (a bipartite graph), find highly connected subgroups.

This is the maximum-edge complete bipartite subgraph.

Applications in clustering: text mining, web community discovery . . .
**Example**

Let $M$ be the biadjacency matrix of the bipartite graph representing the interactions:

$$M = \begin{pmatrix}
1 & 0 & 1 \\
0 & 1 & 1 \\
1 & 0 & 1
\end{pmatrix} = \begin{pmatrix}
1 & 0 \\
0 & 1 \\
1 & 0
\end{pmatrix} \begin{pmatrix}
1 & 0 & 1 \\
0 & 0 & 0 \\
0 & 0 & 0
\end{pmatrix}$$

$$= \begin{pmatrix}
1 & 0 & 1 \\
0 & 0 & 0 \\
1 & 0 & 1
\end{pmatrix} + \begin{pmatrix}
0 & 0 & 0 \\
0 & 1 & 1 \\
0 & 0 & 0
\end{pmatrix}$$

Each rank-one factor corresponds to a *community*.


Conclusion

Low-rank matrix approximation can be used for linear dimensionality reduction. It is a powerful tool for data analysis.

Nonnegativity renders the problem difficult.

However, it enhances significantly its applicability in many areas (by improving interpretability), e.g., image processing, text mining, hyperspectral unmixing, clustering, community detection, and recommender systems.

Still many open questions for NMF and related problems, e.g., subclass of matrices for which NMF can be computed efficiently (separable NMF), bounding and computing nonnegative ranks, non-uniqueness characterization, complexity when using other norms, etc.
Conclusion

Low-rank matrix approximation can be used for linear dimensionality reduction. It is a powerful tool for data analysis.

Nonnegativity renders the problem difficult.

However, it enhances significantly its applicability in many areas (by improving interpretability), e.g., image processing, text mining, hyperspectral unmixing, clustering, community detection, and recommender systems.

Still many open questions for NMF and related problems, e.g., subclass of matrices for which NMF can be computed efficiently (separable NMF), bounding and computing nonnegative ranks, non-uniqueness characterization, complexity when using other norms, etc.
Conclusion

Low-rank matrix approximation can be used for linear dimensionality reduction. It is a powerful tool for data analysis.

Nonnegativity renders the problem difficult.

However, it enhances significantly its applicability in many areas (by improving interpretability), e.g., image processing, text mining, hyperspectral unmixing, clustering, community detection, and recommender systems.

Still many open questions for NMF and related problems, e.g., subclass of matrices for which NMF can be computed efficiently (separable NMF), bounding and computing nonnegative ranks, non-uniqueness characterization, complexity when using other norms, etc.
Conclusion

Low-rank matrix approximation can be used for linear dimensionality reduction. It is a powerful tool for data analysis.

Nonnegativity renders the problem difficult.

However, it enhances significantly its applicability in many areas (by improving interpretability), e.g., image processing, text mining, hyperspectral unmixing, clustering, community detection, and recommender systems.

Still many open questions for NMF and related problems, e.g., subclass of matrices for which NMF can be computed efficiently (separable NMF), bounding and computing nonnegative ranks, non-uniqueness characterization, complexity when using other norms, etc.
Thank you for your attention!

Code and papers available on
https://sites.google.com/site/nicolasgillis/