Online Nonnegative Matrix Factorization and Applications:

Using matrix factorizations for interpretability

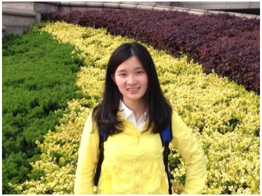
Deanna Needell

Collaborators

Dr. Hanbaek Lyu
(now U. Wisconsin)

Prof. Laura Balzano
(Univ Michigan)

Chris Strohmeier
(PhD student, UCLA)

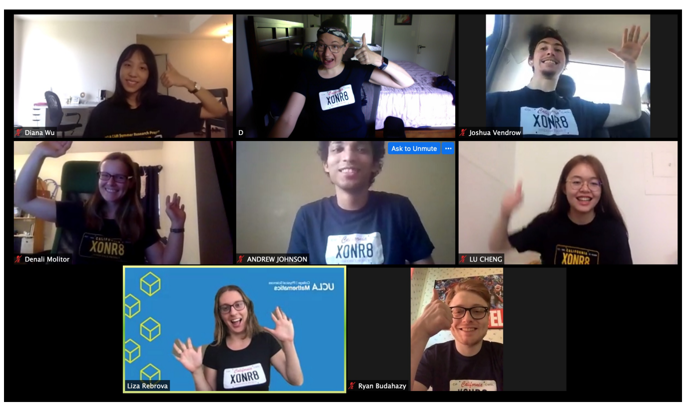


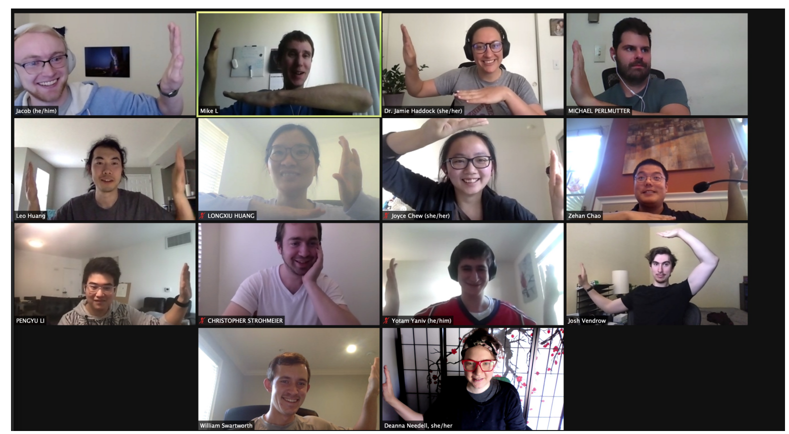
Dr. Longxiu Huang, UCLA
On job market! 🔥🔥

Prof. Keaton Hamm
U Texas, Arlington

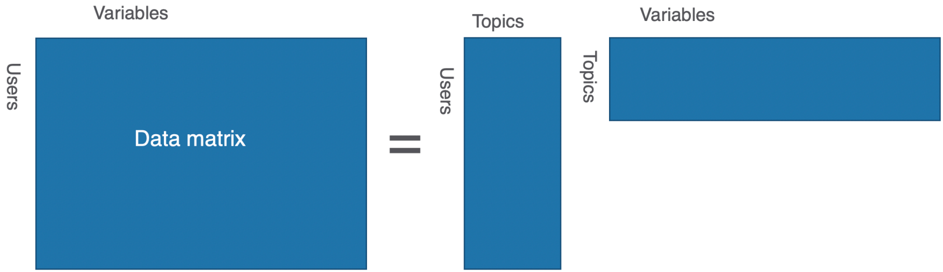
Dr. Hanqin Cai, UCLA
On job market! 🔥🔥

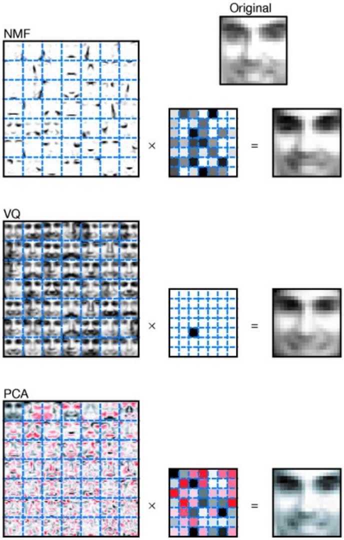
Joint work with





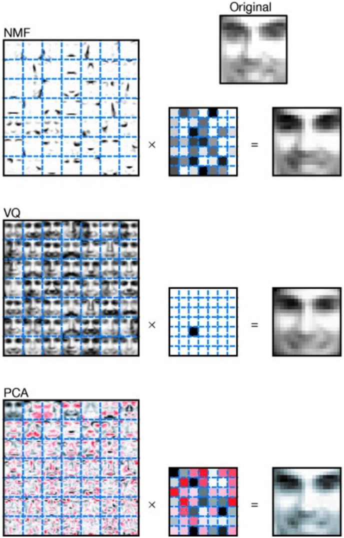
Non-negative matrix factorization





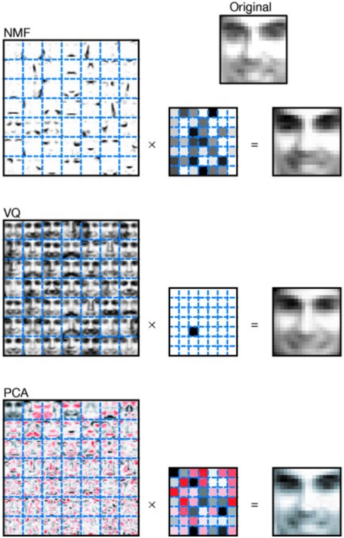
- In NMF, each column of the data matrix has to be represented as a non-negative linear combination of dictionaries

Static NMF algorithms



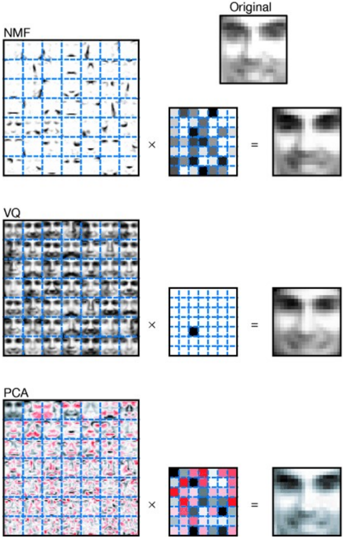
- In NMF, each column of the data matrix has to be represented as a non-negative linear combination of dictionaries
- Hence the dictionaries must be "positive parts" of the columns of the data matrix

Static NMF algorithms



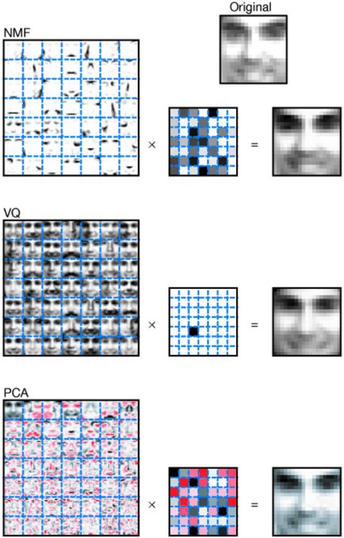
- In NMF, each column of the data matrix has to be represented as a non-negative linear combination of dictionaries
- Hence the dictionaries must be "positive parts" of the columns of the data matrix
- When each column consists of a human face image, NMF learns the parts of human face (e.g., eyes, nodes, mouth)

Static NMF algorithms



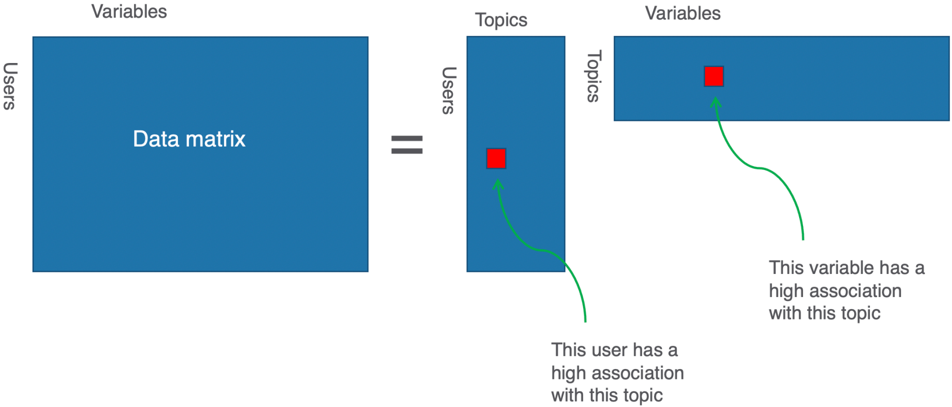
- In NMF, each column of the data matrix has to be represented as a non-negative linear combination of dictionaries
- Hence the dictionaries must be "positive parts" of the columns of the data matrix
- When each column consists of a human face image, NMF learns the parts of human face (e.g., eyes, nodes, mouth)
- This is in contrast to principal component analysis and vector quantization: Due to cancellation between eigenvectors, each 'eigenface' does not have to be parts of face

Static NMF algorithms

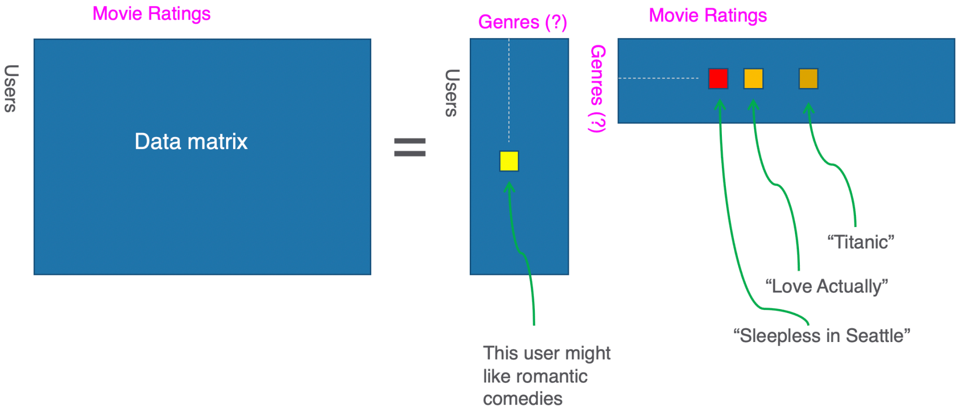


- In NMF, each column of the data matrix has to be represented as a non-negative linear combination of dictionaries
- Hence the dictionaries must be "positive parts" of the columns of the data matrix
- When each column consists of a human face image, NMF learns the parts of human face (e.g., eyes, nodes, mouth)
- This is in contrast to principal component analysis and vector quantization: Due to cancellation between eigenvectors, each 'eigenface' does not have to be parts of face
- NMF was popularized by Lee and Seung in their Nature paper in 1999

Non-negative matrix factorization



What is nonnegative matrix factorization?

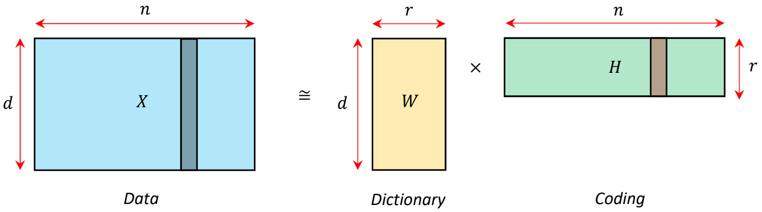


- The goal of **nonnegative matrix factorization** (NMF) is to factorize a data matrix $X \in \mathbb{R}^{d\times n}_{\geq 0}$ into a pair of low-rank nonnegative matrices $W \in \mathbb{R}^{d\times r}$ and $H \in \mathbb{R}^{r\times n}$ by solving the following optimization problem

$$\inf_{W\in\mathbb{R}^{d\times r}_{\geq 0},\, H\in\mathbb{R}^{r\times n}_{\geq 0}} \|X - WH\|_F^2,$$

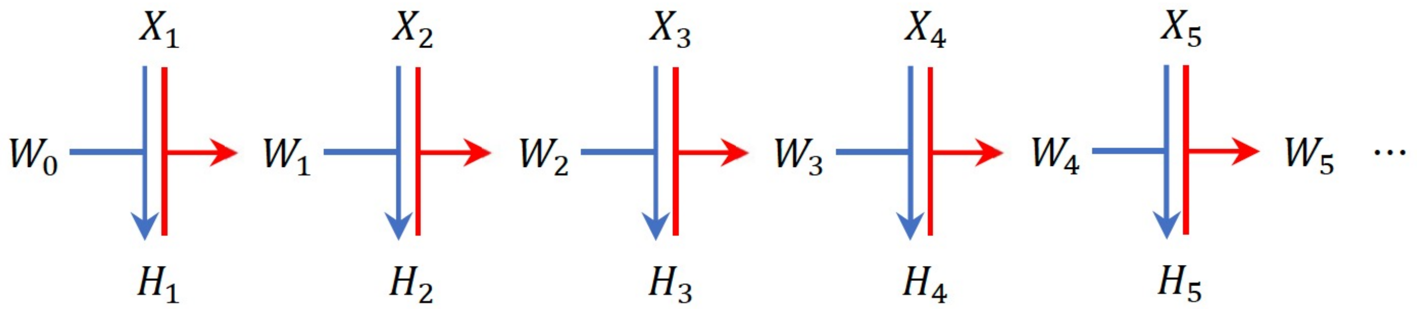
  where $\|A\|_F^2 = \sum_{i,j} A_{ij}^2$ denotes the matrix Frobenius norm.

- $\texttt{Data} \approx \texttt{Dictionary} \times \texttt{Coding}$



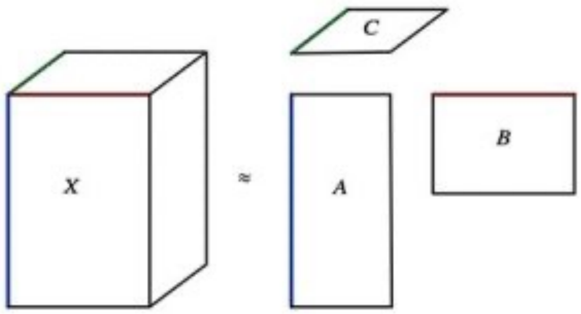
Online NMF

- Considers data that is streaming in over time
- *Learns a factorization that is best (in expectation)*
- Can be used for prediction in time series data
  - Uses "windows" across time to update factors and then predicts into a future window using one of the factors

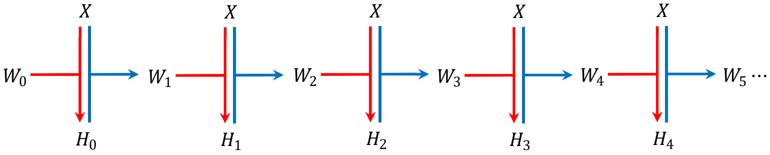


Non-negative Tensor Factorization (NTF)

- Can be extended to tensors in a (nontrivial but) analogous way

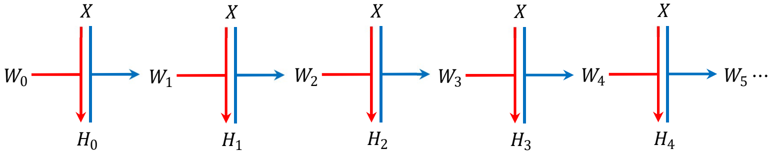


- In order to minimize $\|X - WH\|_F$, one can use block coordinate descent, by iteratively fixing W or H and minimizing the error w.r.t. the other factor



Static NMF algorithms

- In order to minimize $\|X - WH\|_F$, one can use block coordinate descent, by iteratively fixing W or H and minimizing the error w.r.t. the other factor

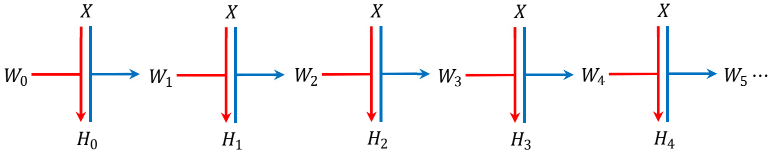


- One of the most popular static NMF algorithm is the **Multiplicative Update** by Lee and Seung: Update all entries of H and W alternatively using the following update

$$H_{ij} \leftarrow H_{ij} \frac{[W^T X]_{ij}}{[W^T W X]_{ij}}, \qquad W_{ij} \leftarrow W_{ij} \frac{[X H^T]_{ij}}{[X H H^T]_{ij}}.$$

Static NMF algorithms

- In order to minimize $\|X - WH\|_F$, one can use block coordinate descent, by iteratively fixing W or H and minimizing the error w.r.t. the other factor



- One of the most popular static NMF algorithm is the **Multiplicative Update** by Lee and Seung: Update all entries of H and W alternatively using the following update

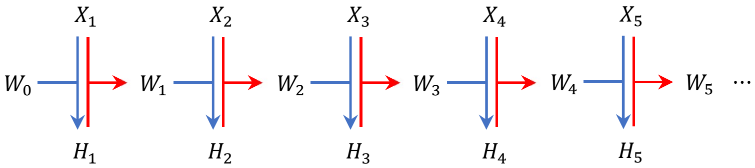
$$H_{ij} \leftarrow H_{ij} \frac{[W^T X]_{ij}}{[W^T W X]_{ij}}, \qquad W_{ij} \leftarrow W_{ij} \frac{[X H^T]_{ij}}{[X H H^T]_{ij}}.$$

- It is known that the error $\|X - WH\|_F^2$ is non-increasing under the above update, but there is no guarantee to converge to a stationary point.

Online NMF

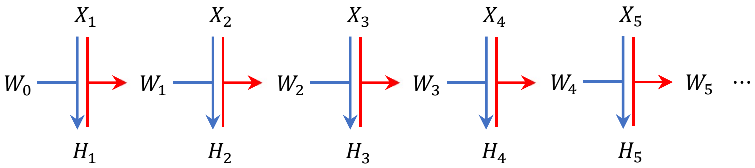
- If the data matrix X is randomly drawn from a sample space $\Omega \subseteq \mathbb{R}^{d\times n}_{\geq 0}$ according to a distribution π, can we still learn the 'best dictionaries' that describe X in law?

- If the data matrix X is randomly drawn from a sample space $\Omega \subseteq \mathbb{R}^{d\times n}_{\geq 0}$ according to a distribution π, can we still learn the 'best dictionaries' that describe X in law?
- The **online Non-negative Matrix Factorization** (ONMF) problem concerns a similar matrix factorization problem for a sequence of input matrices $(x_T)_{t\geq 0}$.



Online NMF

- If the data matrix X is randomly drawn from a sample space $\Omega \subseteq \mathbb{R}^{d\times n}_{\geq 0}$ according to a distribution π, can we still learn the 'best dictionaries' that describe X in law?
- The **online Non-negative Matrix Factorization** (ONMF) problem concerns a similar matrix factorization problem for a sequence of input matrices $(x_T)_{t\geq 0}$.



- Suppose $(X_t)_{t\geq 1}$ is an irreducible Markov chain on a sample space Ω with unique stationary measure π. The goal of ONMF problem is to construct a sequence $(W_t, H_t)_{t\geq 1}$ of dictionary $W_t \in \mathbb{R}^{r\times d}$ and a coding $H_t \in \mathbb{R}^{r\times n}_{\geq 0}$ such that (almost surely)

$$\|X_t - W_{t-1}H_t\|_F^2 \longrightarrow \inf_{W\in\mathbb{R}^{d\times r},\, H\in\mathbb{R}^{r\times n}} \mathbb{E}_{X\sim\pi}\left[\|X - WH\|_F^2\right]$$

Asymptotic solution minimizing surrogate loss function

- Mairal, Bach, Ponce, and Sapiro gave an influential solution to the ONMF problem with a rigorous derivation of almost sure convergence of the empirical loss over time for i.i.d. data matrices.

Asymptotic solution minimizing surrogate loss function

- Mairal, Bach, Ponce, and Sapiro gave an influential solution to the ONMF problem with a rigorous derivation of almost sure convergence of the empirical loss over time for i.i.d. data matrices.
- The idea is to solve the following approximate problem

$$\textit{Upon arrival of } X_t\textit{:} \qquad \begin{cases} H_t = \operatorname{argmin}_{H \in \mathbb{R}^{r \times n}_{\geq 0}} \|X_t - W_{t-1}H\|_F^2 + \lambda \|H\|_1 \\ W_t = \operatorname{argmin}_{W \in \mathcal{C}} \hat{f}_t(W), \end{cases}$$

where $\hat{f}_t(W)$ is a convex upper bounding **surrogate** for $f_t(W)$ defined by

$$\hat{f}_t(W) = \frac{1}{t} \sum_{s=1}^{t} (\|X_s - WH_s\|_F^2 + \lambda \|H_s\|_1).$$

Asymptotic solution minimizing surrogate loss function

- Mairal, Bach, Ponce, and Sapiro gave an influential solution to the ONMF problem with a rigorous derivation of almost sure convergence of the empirical loss over time for i.i.d. data matrices.
- The idea is to solve the following approximate problem

$$\textit{Upon arrival of } X_t: \qquad \begin{cases} H_t = \text{argmin}_{H \in \mathbb{R}^{r\times n}_{\geq 0}} \|X_t - W_{t-1}H\|_F^2 + \lambda \|H\|_1 \\ W_t = \text{argmin}_{W \in \mathcal{C}} \hat{f}_t(W), \end{cases}$$

where $\hat{f}_t(W)$ is a convex upper bounding **surrogate** for $f_t(W)$ defined by

$$\hat{f}_t(W) = \frac{1}{t}\sum_{s=1}^{t}(\|X_s - WH_s\|_F^2 + \lambda\|H_s\|_1).$$

- Namely, we recycle the previously found coding $H_1, \cdots, H_t$ and use them as approximate solutions of the sub-problems. Hence, there is only a single optimization for W_t in the above relaxed problem

Asymptotic solution minimizing surrogate loss function

- Mairal, Bach, Ponce, and Sapiro gave an influential solution to the ONMF problem with a rigorous derivation of almost sure convergence of the empirical loss over time for i.i.d. data matrices.
- The idea is to solve the following approximate problem

$$\textit{Upon arrival of } X_t: \qquad \begin{cases} H_t = \operatorname{argmin}_{H\in\mathbb{R}^{r\times n}_{\geq 0}} \|X_t - W_{t-1}H\|_F^2 + \lambda\|H\|_1 \\ W_t = \operatorname{argmin}_{W\in\mathcal{C}} \hat{f}_t(W), \end{cases}$$

where $\hat{f}_t(W)$ is a convex upper bounding **surrogate** for $f_t(W)$ defined by

$$\hat{f}_t(W) = \frac{1}{t}\sum_{s=1}^{t}(\|X_s - WH_s\|_F^2 + \lambda\|H_s\|_1).$$

- Namely, we recycle the previously found coding $H_1, \cdots, H_t$ and use them as approximate solutions of the sub-problems. Hence, there is only a single optimization for W_t in the above relaxed problem
- But we still need to store the entire history $X_1, \cdots, X_t$ and $H_1, \cdots, H_t$. Do we?

Asymptotic solution minimizing surrogate loss function

- In fact, the approximate ONMF problem is equivalent to

$$\textit{Upon arrival of } X_t\textit{:} \quad \begin{cases} H_t = \operatorname{argmin}_{H \in \mathbb{R}^{r\times n}_{\geq 0}} \|X_t - W_{t-1}H\|_F^2 + \lambda \|H\|_1 \\ A_t = t^{-1}((t-1)A_{t-1} + H_t H_t^T) \\ B_t = t^{-1}((t-1)B_{t-1} + H_t X_t^T) \\ W_t = \operatorname{argmin}_{W \in \mathcal{C} \subseteq \mathbb{R}^{d\times r}_{\geq 0}} \left(\operatorname{tr}(W A_t W^T) - 2\operatorname{tr}(W B_t)\right), \end{cases}$$

where A_0 and B_0 are zero matrices of size $r \times r$ and $r \times d$, respectively.

Asymptotic solution minimizing surrogate loss function

- In fact, the approximate ONMF problem is equivalent to

$$\textit{Upon arrival of } X_t\textit{:} \quad \begin{cases} H_t = \operatorname{argmin}_{H \in \mathbb{R}^{r \times n}_{\geq 0}} \|X_t - W_{t-1}H\|_F^2 + \lambda \|H\|_1 \\ A_t = t^{-1}((t-1)A_{t-1} + H_t H_t^T) \\ B_t = t^{-1}((t-1)B_{t-1} + H_t X_t^T) \\ W_t = \operatorname{argmin}_{W \in \mathcal{C} \subseteq \mathbb{R}^{d \times r}_{\geq 0}} \left(\operatorname{tr}(W A_t W^T) - 2\operatorname{tr}(W B_t) \right), \end{cases}$$

where A_0 and B_0 are zero matrices of size $r \times r$ and $r \times d$, respectively.

- So we only need to store two summary matrices $A_t \in \mathbb{R}^{r \times r}_{\geq 0}$ and $B_t \in \mathbb{R}^{r \times d}$.

Asymptotic solution minimizing surrogate loss function

- In fact, the approximate ONMF problem is equivalent to

$$\textit{Upon arrival of } X_t\textit{:} \quad \begin{cases} H_t = \operatorname{argmin}_{H \in \mathbb{R}^{r\times n}_{\geq 0}} \|X_t - W_{t-1}H\|_F^2 + \lambda \|H\|_1 \\ A_t = t^{-1}((t-1)A_{t-1} + H_t H_t^T) \\ B_t = t^{-1}((t-1)B_{t-1} + H_t X_t^T) \\ W_t = \operatorname{argmin}_{W \in \mathcal{C} \subseteq \mathbb{R}^{d\times r}_{\geq 0}} \left(\operatorname{tr}(W A_t W^T) - 2\operatorname{tr}(W B_t)\right), \end{cases}$$

  where A_0 and B_0 are zero matrices of size $r \times r$ and $r \times d$, respectively.
- So we only need to store two summary matrices $A_t \in \mathbb{R}^{r\times r}_{\geq 0}$ and $B_t \in \mathbb{R}^{r\times d}$.
- Computing W_t also requires solving only a single optimization instance

Convergence under i.i.d. assumption

Upon arrival of X_t:
$$\begin{cases} H_t = \operatorname{argmin}_{H\in\mathbb{R}^{r\times n}_{\geq 0}} \|X_t - W_{t-1}H\|_F^2 + \lambda\|H\|_1 \\ A_t = t^{-1}((t-1)A_{t-1} + H_tH_t^T) \\ B_t = t^{-1}((t-1)B_{t-1} + H_tX_t^T) \\ W_t = \operatorname{argmin}_{W\in\mathcal{C}\subseteq\mathbb{R}^{d\times r}_{\geq 0}} \left(\operatorname{tr}(WA_tW^T) - 2\operatorname{tr}(WB_t)\right), \end{cases}$$

$$f_t = \text{empirical loss}, \qquad \hat{f}_t = \text{surrogate loss}, \qquad f = \text{expected loss}$$

Theorem (Mairal, Bach, Ponce, and Sapiro '10)

Suppose $(X_t)_{t\geq 0}$ are i.i.d. with common distribution π. Let $(W_{t-1}, H_t)_{t\geq 1}$ be the optimal solution to the above ONMF algorithm.

Convergence under i.i.d. assumption

Upon arrival of X_t:
$$\begin{cases} H_t = \text{argmin}_{H\in\mathbb{R}^{r\times n}_{\geq 0}} \|X_t - W_{t-1}H\|_F^2 + \lambda\|H\|_1 \\ A_t = t^{-1}((t-1)A_{t-1} + H_t H_t^T) \\ B_t = t^{-1}((t-1)B_{t-1} + H_t X_t^T) \\ W_t = \text{argmin}_{W\in\mathcal{C}\subseteq\mathbb{R}^{d\times r}_{\geq 0}} \left(\text{tr}(WA_tW^T) - 2\text{tr}(WB_t)\right), \end{cases}$$

$$f_t = \text{empirical loss}, \qquad \hat{f}_t = \text{surrogate loss}, \qquad f = \text{expected loss}$$

Theorem (Mairal, Bach, Ponce, and Sapiro '10)

Suppose $(X_t)_{t\geq 0}$ are i.i.d. with common distribution π. Let $(W_{t-1}, H_t)_{t\geq 1}$ be the optimal solution to the above ONMF algorithm.

(i) $(f_t(W_t))_{t\geq 1}$ *and* $(\hat{f}_t(W_t))_{t\geq 1}$ *converge to the same constant almost surely.*

$$\textit{Upon arrival of } X_t\text{:}\quad \begin{cases} H_t = \operatorname{argmin}_{H \in \mathbb{R}^{r\times n}_{\geq 0}} \|X_t - W_{t-1}H\|_F^2 + \lambda \|H\|_1 \\ A_t = t^{-1}((t-1)A_{t-1} + H_t H_t^T) \\ B_t = t^{-1}((t-1)B_{t-1} + H_t X_t^T) \\ W_t = \operatorname{argmin}_{W \in \mathcal{C} \subseteq \mathbb{R}^{d\times r}_{\geq 0}} \left(\operatorname{tr}(W A_t W^T) - 2\operatorname{tr}(W B_t)\right), \end{cases}$$

$$f_t = \text{empirical loss}, \qquad \hat{f}_t = \text{surrogate loss}, \qquad f = \text{expected loss}$$

Theorem (Mairal, Bach, Ponce, and Sapiro '10)

Suppose $(X_t)_{t\geq 0}$ are i.i.d. with common distribution π. Let $(W_{t-1}, H_t)_{t\geq 1}$ be the optimal solution to the above ONMF algorithm.

(i) $(f_t(W_t))_{t\geq 1}$ *and* $(\hat{f}_t(W_t))_{t\geq 1}$ *converge to the same constant almost surely.*

(ii) $\limsup_{t\to\infty} \|\nabla f(W_t)\|_{\text{op}} = 0$ *almost surely.*

Convergence under Markovian dependence

Upon arrival of X_t:

$$\begin{cases} H_t = \text{argmin}_{H\in\mathbb{R}^{r\times n}_{\geq 0}} \|X_t - W_{t-1}H\|_F^2 + \lambda\|H\|_1 \\ A_t = t^{-1}((t-1)A_{t-1} + H_t H_t^T) \\ B_t = t^{-1}((t-1)B_{t-1} + H_t X_t^T) \\ W_t = \text{argmin}_{W\in\mathcal{C}\subseteq\mathbb{R}^{d\times r}_{\geq 0}} \left(\text{tr}(WA_tW^T) - 2\text{tr}(WB_t)\right), \end{cases}$$

$f_t =$ empirical loss, $\hat{f}_t =$ surrogate loss, $f =$ expected loss

Theorem (Balzano, Lyu, Needell '19+)

Suppose $(X_t)_{t\geq 0}$ is an irreducible MC on a finite state space with unique stationary distribution π. Let $(W_{t-1}, H_t)_{t\geq 1}$ be a solution to the above ONMF algorithm. Then the following hold.

(i) $\lim_{t\to\infty} \mathbb{E}[f_t(W_t)] = \lim_{t\to\infty} \mathbb{E}[\hat{f}_t(W_t)] < \infty$.

Convergence under Markovian dependence

$$\text{Upon arrival of } X_t: \quad \begin{cases} H_t = \operatorname{argmin}_{H \in \mathbb{R}^{r\times n}_{\geq 0}} \|X_t - W_{t-1}H\|_F^2 + \lambda\|H\|_1 \\ A_t = t^{-1}((t-1)A_{t-1} + H_t H_t^T) \\ B_t = t^{-1}((t-1)B_{t-1} + H_t X_t^T) \\ W_t = \operatorname{argmin}_{W \in \mathcal{C} \subseteq \mathbb{R}^{d\times r}_{\geq 0}} \left(\operatorname{tr}(WA_tW^T) - 2\operatorname{tr}(WB_t)\right), \end{cases}$$

$$f_t = \text{empirical loss}, \qquad \hat{f}_t = \text{surrogate loss}, \qquad f = \text{expected loss}$$

Theorem (Balzano, Lyu, Needell '19+)

Suppose $(X_t)_{t\geq 0}$ is an irreducible MC on a finite state space with unique stationary distribution π. Let $(W_{t-1}, H_t)_{t\geq 1}$ be a solution to the above ONMF algorithm. Then the following hold.

(i) $\lim_{t\to\infty} \mathbb{E}[f_t(W_t)] = \lim_{t\to\infty} \mathbb{E}[\hat{f}_t(W_t)] < \infty$.

(ii) $f_t(W_t) - \hat{f}_t(W_t) \to 0$ *as* $t \to \infty$ *almost surely.*

Convergence under Markovian dependence

Upon arrival of X_t:
$$\begin{cases} H_t = \text{argmin}_{H \in \mathbb{R}^{r\times n}_{\geq 0}} \|X_t - W_{t-1}H\|_F^2 + \lambda \|H\|_1 \\ A_t = t^{-1}((t-1)A_{t-1} + H_t H_t^T) \\ B_t = t^{-1}((t-1)B_{t-1} + H_t X_t^T) \\ W_t = \text{argmin}_{W \in \mathcal{C} \subseteq \mathbb{R}^{d\times r}_{\geq 0}} \left(\text{tr}(W A_t W^T) - 2\text{tr}(W B_t)\right), \end{cases}$$

$$f_t = \text{empirical loss}, \qquad \hat{f}_t = \text{surrogate loss}, \qquad f = \text{expected loss}$$

Theorem (Balzano, Lyu, Needell '19+)

Suppose $(X_t)_{t\geq 0}$ is an irreducible MC on a finite state space with unique stationary distribution π. Let $(W_{t-1}, H_t)_{t\geq 1}$ be a solution to the above ONMF algorithm. Then the following hold.

(i) $\lim_{t\to\infty} \mathbb{E}[f_t(W_t)] = \lim_{t\to\infty} \mathbb{E}[\hat{f}_t(W_t)] < \infty$.

(ii) $f_t(W_t) - \hat{f}_t(W_t) \to 0$ *as* $t \to \infty$ *almost surely.*

(iii) $\limsup_{t\to\infty} \|\nabla f(W_t)\|_{\text{op}} = 0$ *almost surely.*

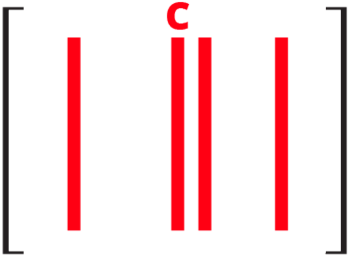
CUR Decomposition

- $A \in \mathbb{R}^{d \times d}$,

$$\begin{bmatrix} \quad \end{bmatrix}$$

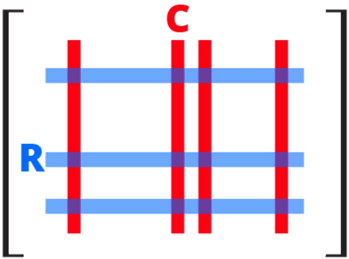
CUR Decomposition

- $A \in \mathbb{R}^{d \times d}$,
- $C \in \mathbb{R}^{d \times k}$: k columns of A



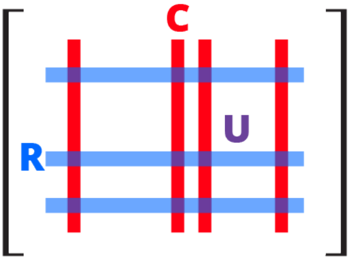
CUR Decomposition

- $A \in \mathbb{R}^{d \times d}$,
- $C \in \mathbb{R}^{d \times k}$: k columns of A
- $R \in \mathbb{R}^{s \times d}$: s rows of A



CUR Decomposition

- $A \in \mathbb{R}^{d \times d}$,
- $C \in \mathbb{R}^{d \times k}$: k columns of A
- $R \in \mathbb{R}^{s \times d}$: s rows of A
- $U \in \mathbb{R}^{s \times k}$: the intersection of C and R



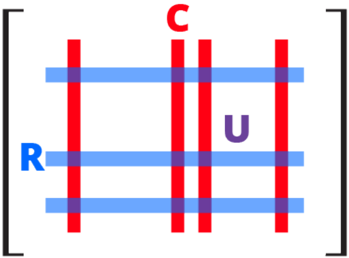
CUR Decomposition

- $A \in \mathbb{R}^{d\times d}$,
- $C \in \mathbb{R}^{d\times k}$: k columns of A
- $R \in \mathbb{R}^{s\times d}$: s rows of A
- $U \in \mathbb{R}^{s\times k}$: the intersection of C and R

Theorem

If $\mathrm{rank}(U) = \mathrm{rank}(A)$, *then*

$$A = CU^{\dagger}R.$$



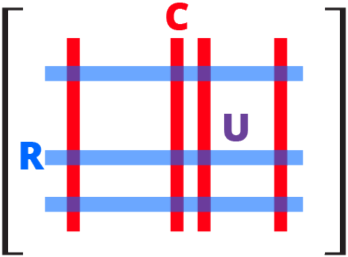
Question

Can the matrix CUR decomposition be generalized to the multidimensional data structure (i.e., tensor)?

Tensor CUR Decompositions

Motivation

Let $A \in \mathbb{R}^{d\times d}$ with CUR decomposition of $A = CU^{\dagger}R$. Then
$A = CU^{\dagger}R = CU^{\dagger}UU^{\dagger}R = U \times_1 (CU^{\dagger}) \times_2 (R^T(U^T)^{\dagger})$.



Characterizations of Tensor CUR Decompositions

(A taste...)

Theorem (Cai–Hamm–Huang–N, 2021)

(Chidori CUR) Let $\mathcal{A} \in \mathbb{R}^{d\times\cdots\times d}$ *with* $\text{rank}(\mathcal{A}) = (r, \dots, r)$*. Let* $I_i \subseteq [d]$*. Set* $\mathcal{R} = \mathcal{A}(I_1, \cdots, I_n)$, $C_i = \mathcal{A}_{(i)}(:, J_i := \otimes_{j\neq i} I_j)$ *and* $U_i = C_i(I_i, :)$*. Then the following are equivalent:*

1. $\text{rank}(U_i) = r$,
2. $\mathcal{A} = \underbrace{\mathcal{R} \times_1 (C_1 U_1^\dagger) \times_2 \cdots \times_n (C_n U_n^\dagger)}_{CUR}$,
3. $\text{rank}(\mathcal{R}) = (r, \cdots, r)$,
4. $\text{rank}(\mathcal{A}_{(i)}(I_i, :)) = r$ *for all* $i \in [n]$.

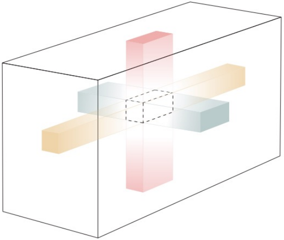
Moreover, if the above statements hold, then $\mathcal{A} = \mathcal{A} \times_{i=1}^n (C_i C_i^\dagger)$.

Characterizations of Tensor CUR Decompositions

Theorem (Cai–Hamm–Huang–N, 2021)

(Fiber CUR): Let $\mathcal{A} \in \mathbb{R}^{d\times\cdots\times d}$ *with* $\operatorname{rank}(\mathcal{A}) = (r, \dots, r)$. *Let* $I_i \subseteq [d]$ *and* $J_i \subseteq [d^{n-1}]$. *Set* $\mathcal{R} = \mathcal{A}(I_1, \cdots, I_n)$, $C_i = \mathcal{A}_{(i)}(:, J_i)$ *and* $U_i = C_i(I_i, :)$. *Then the following statements are equivalent*

1. $\operatorname{rank}(U_i) = r$,
2. $\mathcal{A} = \underbrace{\mathcal{R} \times_1 (C_1 U_1^\dagger) \times_2 \cdots \times_n (C_n U_n^\dagger)}_{CUR}$,
3. $\operatorname{rank}(C_i) = r$ *for all* $i \in [n]$ *and* $\operatorname{rank}(\mathcal{R}) = (r, \cdots, r)$,
4. $\operatorname{rank}(C_i) = r$ *and* $\operatorname{rank}(\mathcal{A}_{(i)}(I_i, :)) = r$ *for all* $i \in [n]$.



(Thanks Dustin Mixon)

Figure 1: Illustration of Chidori CUR decomposition à la Theorem 3.1 of a 3-mode tensor in the case when the indices I_i are each an interval and $J_i = \otimes_{j \neq i} I_j$. The matrix C_1 is obtained by unfolding the red subtensor along mode 1, C_2 by unfolding the green subtensor along mode 2, and C_3 by unfolding the yellow subtensor along mode 3. The dotted line shows the boundaries of $\mathcal{R}$. In this case $U_i = \mathcal{R}_{(i)}$ for all i.

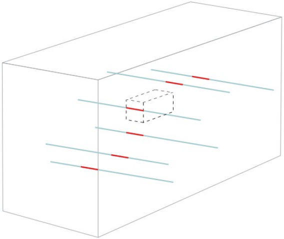


Figure 2: Illustration of the Fiber CUR Decomposition of Theorem 3.3 in which J_i is not necessarily related to I_i. The lines correspond to rows of C_2, and red indices within correspond to rows of U_2. Note that the lines may (but do not have to) pass through the core subtensor $\mathcal{R}$ outlined by dotted lines. Fibers used to form C_1 and C_3 are not shown for clarity.

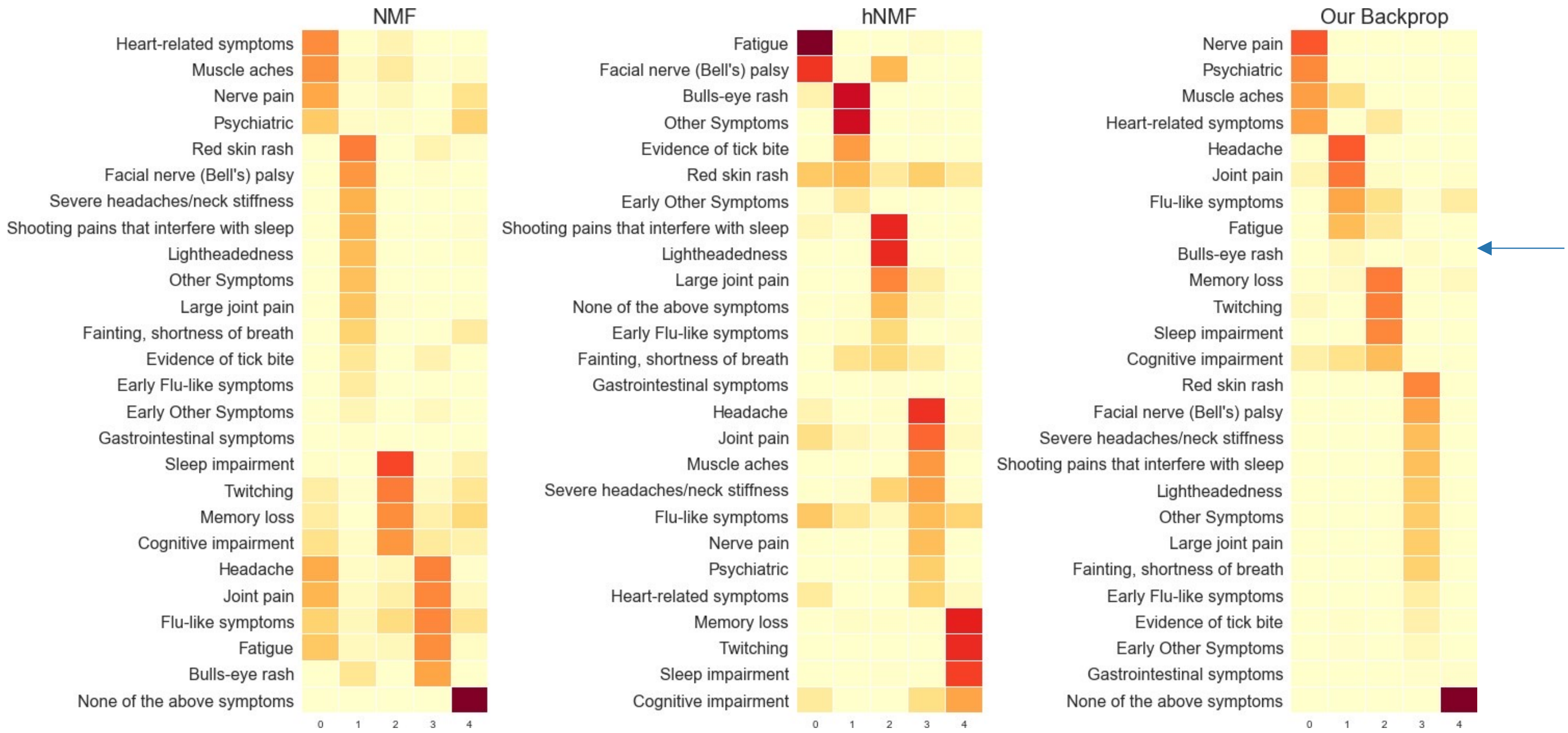
Applications of ONMF

MyLymeData

- Lyme disease a vector-borne disease typically transmitted by tick or insect bite or blood-blood contact
  - Symptoms often mimic those of others, e.g. MS / ALS / Parkinsons / FMA … and can become chronic
- CDC estimates 300,000 new diagnoses each year
  - Likely a grandiose underestimate
- Poorly understood, poorly funded, poorly diagnosed, poorly treated

MyLymeData
A PROJECT OF LYMEDISEASE.ORG

Comparisons on Lyme data



The hidden topics here may provide insight on how symptoms manifest themselves

ONMF for image reconstruction

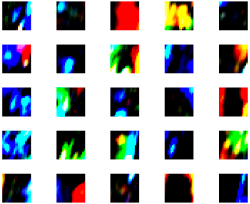
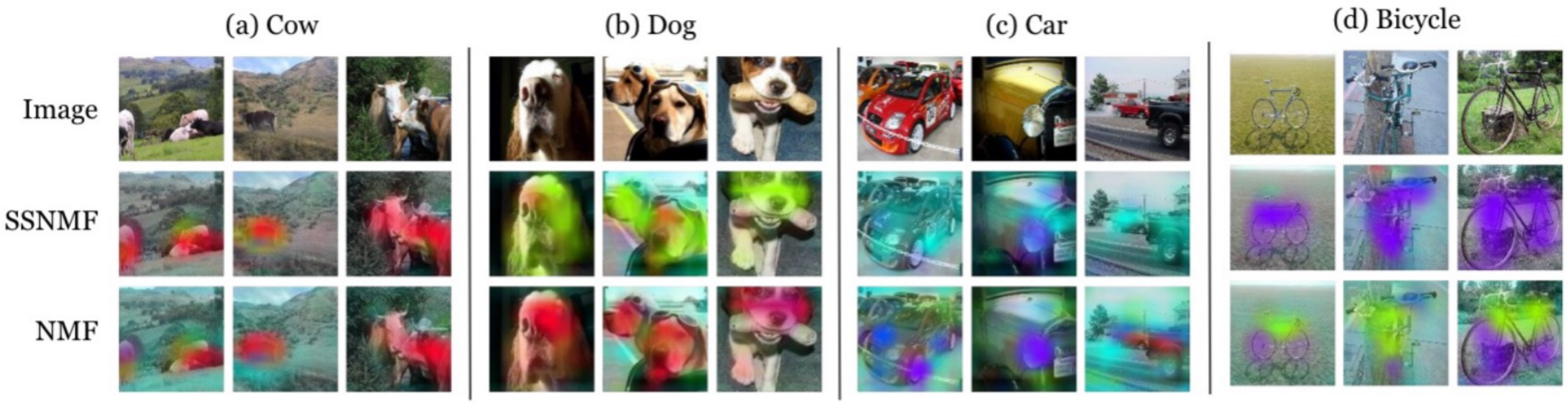


Fig. 7: Image Compression Via ONMF. (Top) uncompressed image of Leonid Afremov's famous painting "Rain's Rustle." (Middle) 25 of the 100 learned dictionary elements, reshaped from their vectorized form to color image patch form. (Bottom): Painting compressed using a dictionary of 100 vectorized 20×20 color image patches obtained from 30 data samples of ONMF, each consisting of 1000 randomly selected sample patches. We used an overlap length of 15 in the patch averaging for the construction of the compressed image.

More applications

(O)NMF for image co-segmentation



ONMF as a pattern detection and prediction tool – COVID19

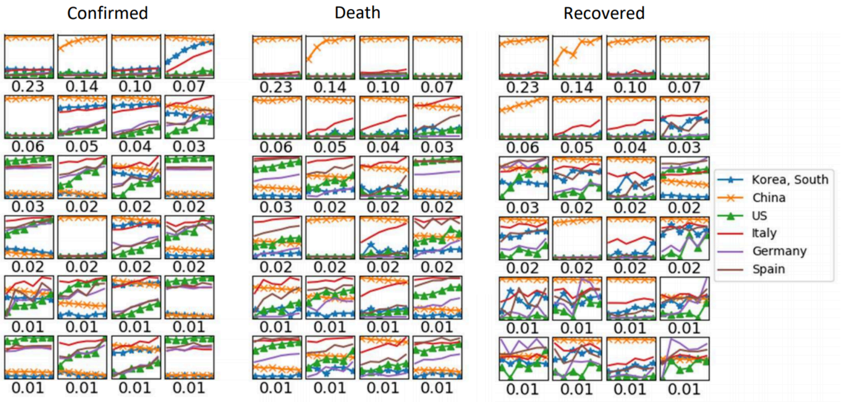


Fig. 2. 24 Joint dictionary atoms of 6-day evolution patterns of new daily cases (confirmed/death/recovered) in six countries (S. Korea, China, US, Italy, Germany, and France). Each dictionary atom is a $6*6*3 = 108$ dimensional vector corresponding to $\texttt{time} * \texttt{country} * \texttt{case type}$. The corresponding importance metric is shown below each atom. 50 atoms are learned and the figure shows top 24 with the highest importance metric.

ONMF as a pattern detection and prediction tool – COVID19

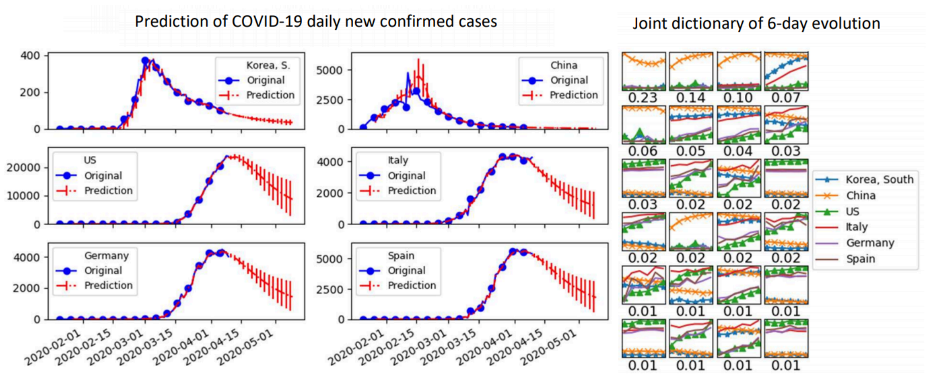
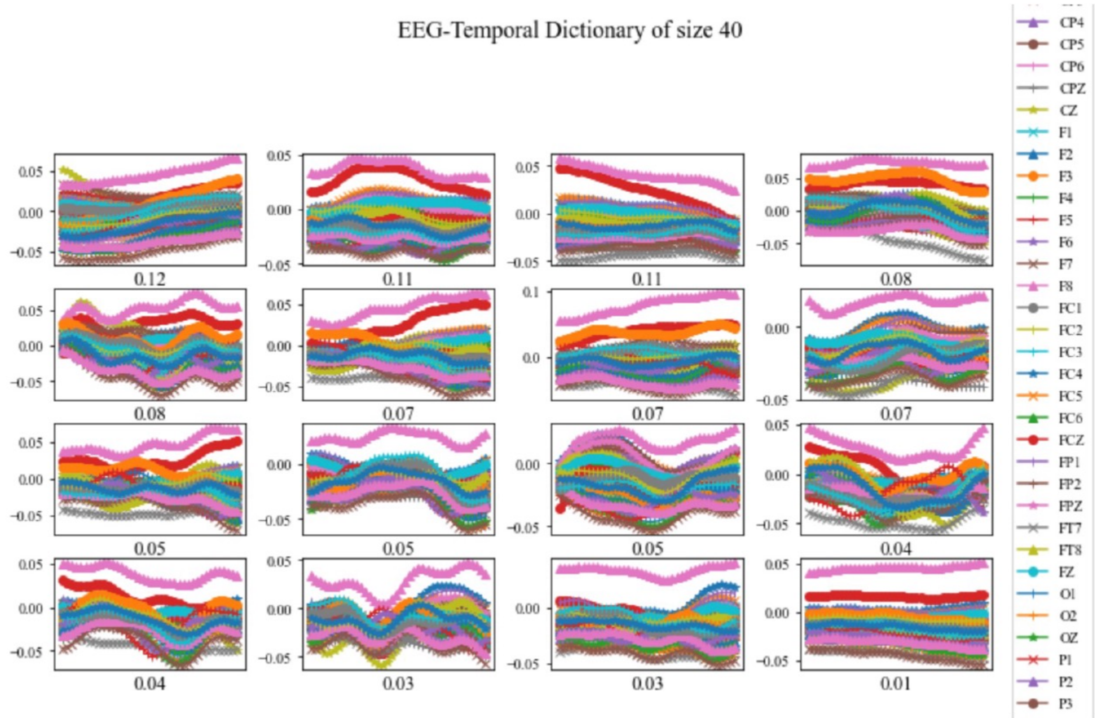


Fig. 3. Joint dictionary learning and prediction for the time-series of new daily cases (confirmed/death/recovered) in six countries (S. Korea, China, US, Italy, Germany, and France). After joint dictionary atoms are learned by minibatch learning, they are further adapted to the time-series data by concurrent online learning and predictions. (Right) Joint dictionary atoms of 6-day evolution patterns of new confirmed cases. The corresponding importance metric is shown below each atom. (Left) Plot of the original and predicted daily new confirmed cases of the six countries. The errorbar in the red plot shows standard deviation of 1000 trials.

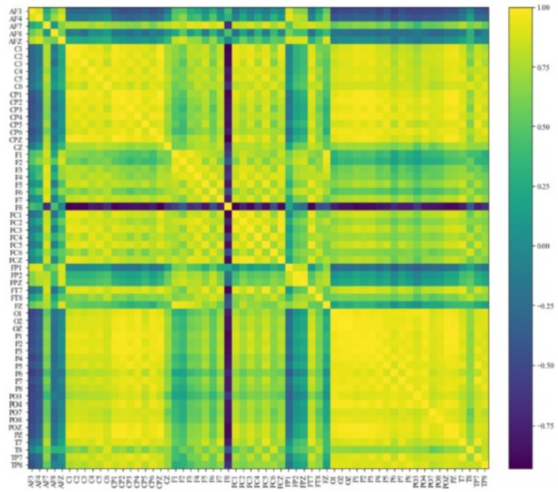
More applications

ONMF on EEG node correlations (UCI EEG Alcoholism data, 64 electrodes)

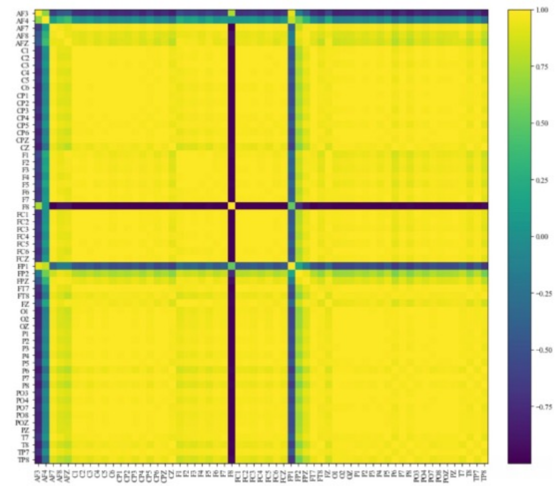


More applications

ONMF on EEG data (node correlations)



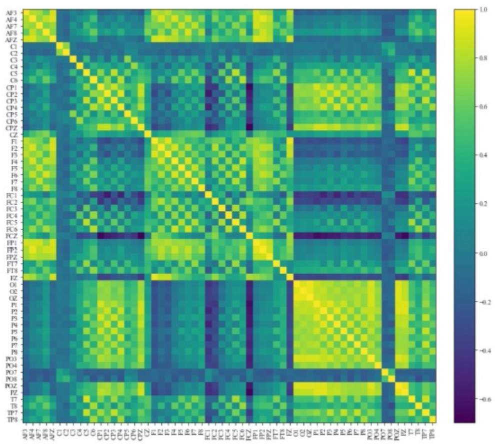
(Pearson)



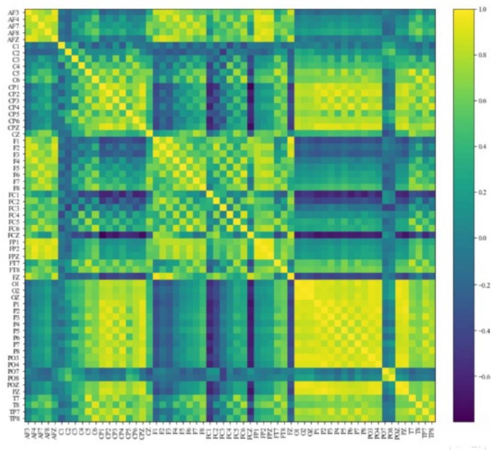
(ONMF)

More applications

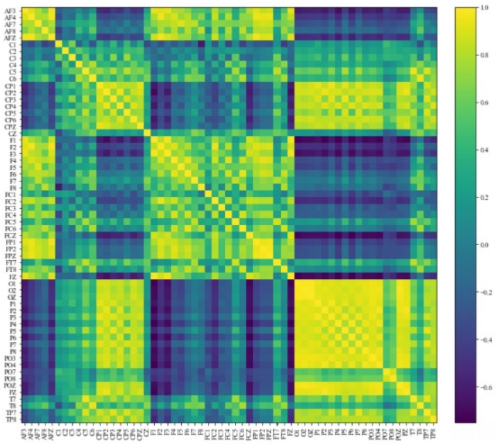
ONMF on EEG data (node correlations)



(Pearson w/ gradient)



(ONMF w/ gradient)



(ONMF w/o gradient, r=16)

ONTF to learn activation patterns in mouse cortex

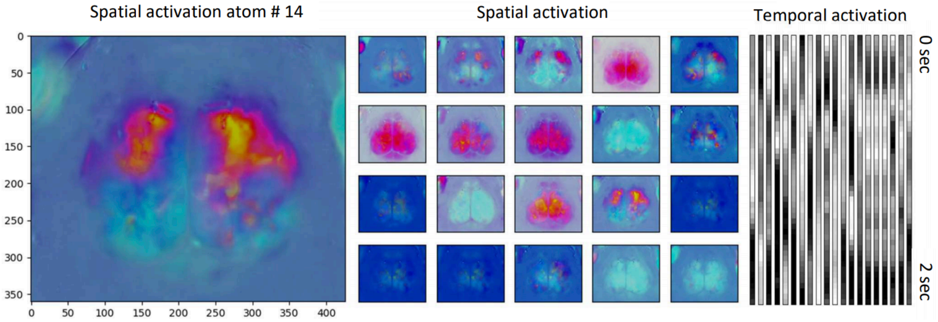
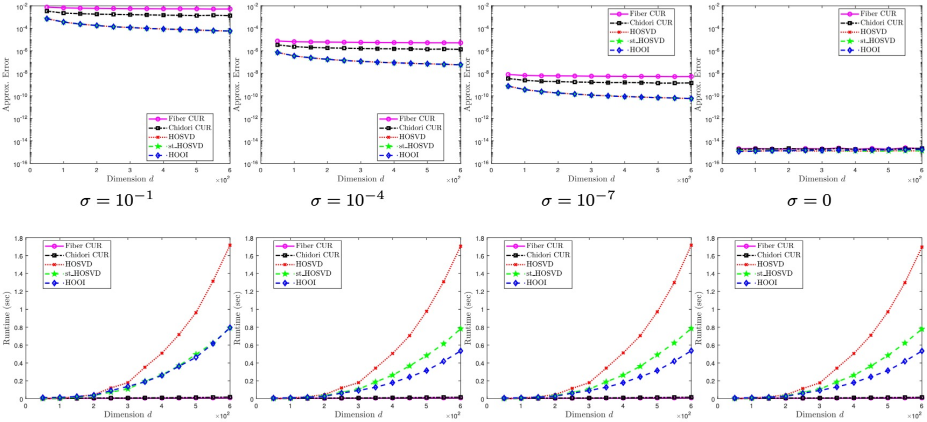


FIGURE 4. Learning 20 CP-dictionary patches from video frames on brain activity across the mouse cortex.

Performance of tensor CUR methods



		Ribeira	Braga	Ruivaes
Size		$1017 \times 1340 \times 33$	$1021 \times 1338 \times 33$	$1017 \times 1338 \times 33$
Rank		$(60, 60, 7)$	$(60, 60, 5)$	$(65, 65, 4)$
Runtime (seconds)	Fiber CUR	**0.29**	**0.26**	**0.31**
	Chidori CUR	0.66	0.59	0.55
	HOSVD	1.49	1.41	1.42
	st_HOSVD	0.83	0.77	0.76
	HOOI	2.29	2.67	3.30
SNR (dB)	Fiber CUR	24.14	17.93	15.53
	Chidori CUR	**24.39**	**18.56**	**15.84**
	HOSVD	22.99	17.70	15.48
	st_HOSVD	22.18	17.90	15.49
	HOOI	24.33	18.00	15.61

Figure 5: Visual comparison of the original and compressed hyperspectral images. From top to bottom, each row of the images are for the datasets Ribeira, Braga and Ruivaes, respectively.

Robustness of tensor CUR methods

	frame size	frame number	runtime (sec)	
			RCUR	RPCA
Shoppingmall	256×320	1000	7.69	44.30
Restaurant	120×160	3055	3.48	31.63
OSU	240×320	1506	10.39	68.62

Figure 1: *Restaurant*: The first column contains three randomly selected frames from the original video. The middle two columns are the separated background and foreground outputs of RCUR, respectively. The right two columns are the separated background and foreground outputs of RPCA, respectively.

Robustness of tensor CUR methods

Figure 5: Face modeling on *ExtYaleB*: Visual comparison of the outputs by RCUR and RPCA for face modeling task. The first row contains the original face images. The second and third rows are the face models and the facial occlusions outputted by RCUR, respectively. The last two rows are the face models and the facial occlusions outputted by RPCA, respectively.

UCLA

Thank you for listening!

- deanna@math.ucla.edu
- math.ucla.edu/~deanna