Online robust matrix factorization for dependent data streams

Hanbaek Lyu

Department of Mathematics, University of California, Los Angeles

Seminar on applied math and data Science, HKUST

Joint work with HanQin Cai and Deanna Needell

Mar. 24, 2019

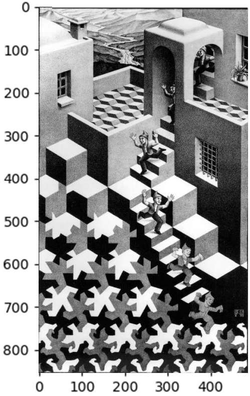
Overview

1. Introduction
2. ORMF algorithm and convergence result
3. Applications: Dictionary learning from networks

1. Introduction

Learning parts of images – Image reconstruction

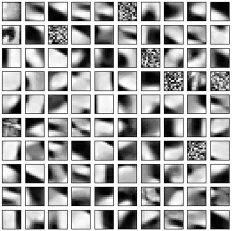
Original image -
Cycle by M. C. Escher (1928)



- Dictionary learning enables a compressed representation of complex objects using a few dictionary elements.
- Used in data compression, reconstruction, transfer learning, etc.

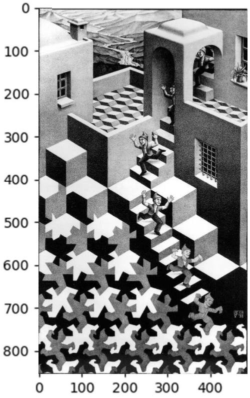
Learning parts of images – Image reconstruction

Dictionary learned from
Cycle by M. C. Escher



(basis)

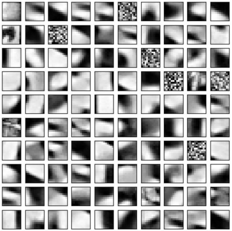
Original image -
Cycle by M. C. Escher (1928)



- Dictionary learning enables a compressed representation of complex objects using a few dictionary elements.
- Used in data compression, reconstruction, transfer learning, etc.

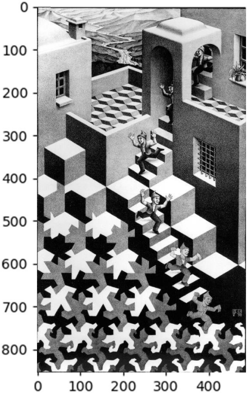
Learning parts of images – Image reconstruction

Dictionary learned from
Cycle by M. C. Escher

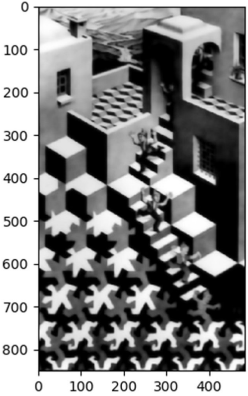


(basis)

Original image -
Cycle by M. C. Escher (1928)

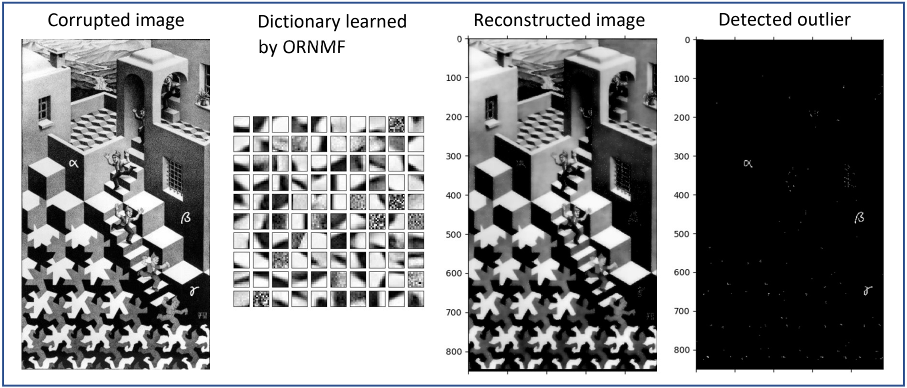


Reconstructed image
using learned dictionary



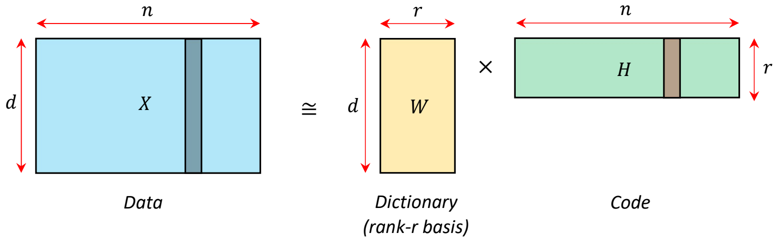
- Dictionary learning enables a compressed representation of complex objects using a few dictionary elements.
- Used in data compression, reconstruction, transfer learning, etc.
- Img recons. = (local approx. by dict.) + (Averaging)

Simultaneous dictionary learning and outlier detection

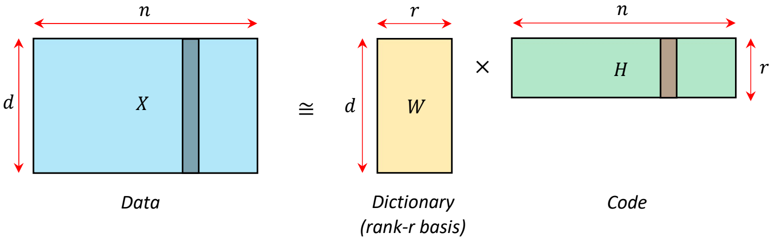


- What defines an outlier? How can we detect them?
- Low-rank based approach – Outlier = Data - Low-rank approx.
- Dictionary-based approach – Outlier = Data - Reconstruction from dictionary
- Dictionary learning has to be done in a robust way

- Matrix Factorization is a fundamental tool in dictionary learning problems.



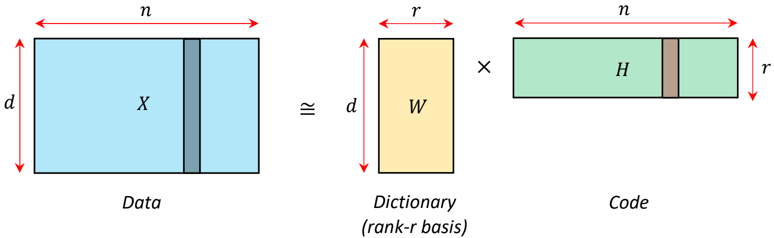
- Matrix Factorization is a fundamental tool in dictionary learning problems.



- Formulated as an optimization problem:

$$\begin{aligned} \text{minimize} \quad & \|X - WH\| + \lambda_1 \|H\|_1 \quad \text{(Reconstruction error)} \\ \text{subject to} \quad & W \in \mathcal{C},\ H \in \mathcal{C}' \quad (\textit{Constraints}) \end{aligned}$$

- Matrix Factorization is a fundamental tool in dictionary learning problems.

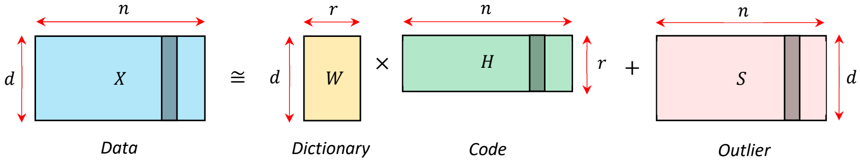


- Formulated as an optimization problem:

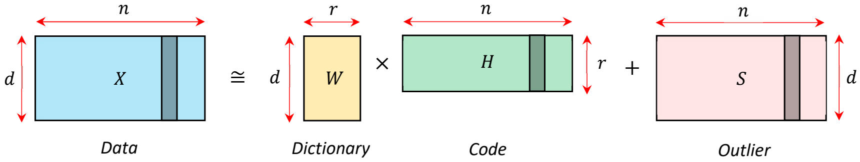
$$\begin{aligned} \text{minimize} \quad & \|X - WH\| + \lambda_1 \|H\|_1 \quad \text{(Reconstruction error)} \\ \text{subject to} \quad & W \in \mathcal{C},\ H \in \mathcal{C}' \quad (\textit{Constraints}) \end{aligned}$$

- Non-convex optimization problem $\rightarrow$ No guarantee for global convergence

- Robust Matrix Factorization enables simultaneous dictionary learning and outlier detection



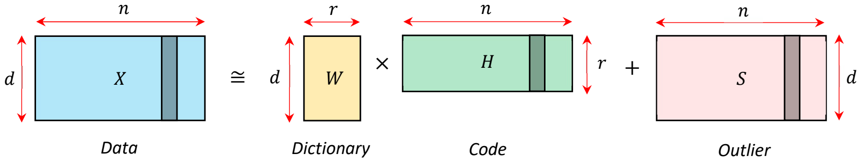
- Robust Matrix Factorization enables simultaneous dictionary learning and outlier detection



- Formulated as an optimization problem:

$$\begin{aligned} \text{minimize} \quad & \|X - WH - S\| + \lambda_1 \|H\|_1 + \lambda_2 \|S\|_1 \quad \text{(Reconstruction error)} \\ \text{subject to} \quad & W \in \mathcal{C},\ H \in \mathcal{C}' \quad (\textit{Constraints}) \end{aligned}$$

- Robust Matrix Factorization enables simultaneous dictionary learning and outlier detection



- Formulated as an optimization problem:

$$\begin{aligned} \text{minimize} \quad & \|X - WH - S\| + \lambda_1 \|H\|_1 + \lambda_2 \|S\|_1 \quad \text{(Reconstruction error)} \\ \text{subject to} \quad & W \in \mathcal{C},\ H \in \mathcal{C}' \quad (\textit{Constraints}) \end{aligned}$$

- Non-convex optimization problem $\rightarrow$ No guarantee for global convergence

Matrix Factorization - other examples

- Singular Value Decomposition (SVD):

$$\underset{W\in\mathbb{R}^{d\times r},\, H\in\mathbb{R}^{r\times n}}{\text{minimize}} \|X - WH\|_F$$

- Non-negative Matrix Factorization (NMF):

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

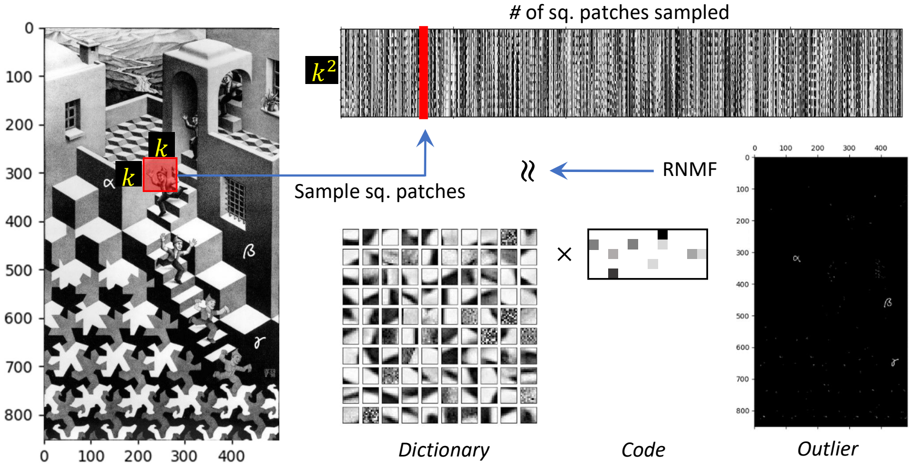
  - Corresponding dictionary columns can be interpreted as 'parts' of the data matrix (Lee, Seung '99 [**lee1999learning**])

- Subspace Clustering (may have $r > d$):

$$\underset{W\in\mathbb{R}^{d\times r},\, H \text{ group sparse}}{\text{minimize}} \|X - WH\|_F$$

Matrix Completion, Probabilistic PCA, Sparse PCA, Robust PCA, Poisson PCA, Heteroscedastic PCA, Bilinear Inverse Problems, Robust NMF, Max-Plus Factorization

Illustration of RMF application to images



Online RMF

- Data matrix could be too large to be loaded in a memory or processed at once

Online RMF

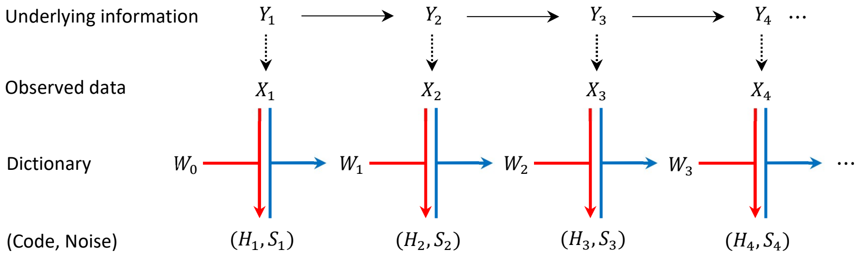
- Data matrix could be too large to be loaded in a memory or processed at once
- Only sub-matrices of a huge data set may be available through sampling

Online RMF

- Data matrix could be too large to be loaded in a memory or processed at once
- Only sub-matrices of a huge data set may be available through sampling
- We may want to learn from a complicated probability distribution on the sample space of data – e.g., posterior distribution

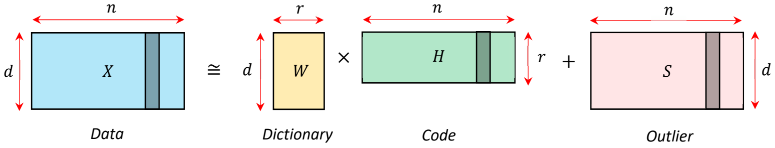
Online RMF

- Data matrix could be too large to be loaded in a memory or processed at once
- Only sub-matrices of a huge data set may be available through sampling
- We may want to learn from a complicated probability distribution on the sample space of data – e.g., posterior distribution
- The **Online Matrix Factorization** (OMF) problem concerns a similar matrix factorization problem for a sequence of input matrices $(X_t)_{t\geq 0}$.



Reminder of matrix factorization

- Robust Matrix Factorization



$$\begin{aligned} \text{minimize} \quad & \|X - WH - S\|_F + \lambda_1 \|H\|_1 + \lambda_2 \|S\|_2 \qquad \text{(Reconstruction error)} \\ \text{subject to} \quad & W \in \mathcal{C},\ H \in \mathcal{C}' \qquad \text{(Compact, convex)} \end{aligned}$$

- **Online RMF for streaming data**:

Learn Robust Dictionary W from a seq. of data matrices $(X_t)_{t \geq 0}$.

2. ORMF algorithm and convergence result

Online MF as Empirical Loss Minimization

- Fix $\lambda > 0$ and define the following the **quadratic loss function**

$$\ell(X, W) = \inf_{H \in \mathcal{C}' \subseteq \mathbb{R}^{r \times n},\, S \in \mathbb{R}^{d \times n}} \|X - WH - S\|_F^2 + \lambda_1 \|H\|_1 + \lambda_2 \|S\|_1$$

Define the **expected loss** and **empirical loss** functions

$$f(W) = \mathbb{E}_{X \sim \pi}[\ell(X, W)], \qquad f_t(W) = \frac{1}{t} \sum_{s=1}^{t} \ell(X_s, W)$$

Online MF as Empirical Loss Minimization

- Fix $\lambda > 0$ and define the following the **quadratic loss function**

$$\ell(X, W) = \inf_{H \in \mathcal{C}' \subseteq \mathbb{R}^{r \times n},\, S \in \mathbb{R}^{d \times n}} \|X - WH - S\|_F^2 + \lambda_1 \|H\|_1 + \lambda_2 \|S\|_1$$

Define the **expected loss** and **empirical loss** functions

$$f(W) = \mathbb{E}_{X \sim \pi}[\ell(X, W)], \qquad f_t(W) = \frac{1}{t} \sum_{s=1}^{t} \ell(X_s, W)$$

- If $(X_t)_{t \geq 0}$ is i.i.d. with common distribution π, then by SLLN,

$$\lim_{t \to \infty} f_t(W) = f(W) \quad \text{a.s. for all } W \in \mathcal{C}.$$

Online MF as Empirical Loss Minimization

- Fix $\lambda > 0$ and define the following the **quadratic loss function**

$$\ell(X, W) = \inf_{H \in \mathcal{C}' \subseteq \mathbb{R}^{r \times n},\, S \in \mathbb{R}^{d \times n}} \|X - WH - S\|_F^2 + \lambda_1 \|H\|_1 + \lambda_2 \|S\|_1$$

Define the **expected loss** and **empirical loss** functions

$$f(W) = \mathbb{E}_{X \sim \pi}[\ell(X, W)], \qquad f_t(W) = \frac{1}{t} \sum_{s=1}^{t} \ell(X_s, W)$$

- If $(X_t)_{t \geq 0}$ is i.i.d. with common distribution π, then by SLLN,

$$\lim_{t \to \infty} f_t(W) = f(W) \quad \text{a.s. for all } W \in \mathcal{C}.$$

- Same holds if $(X_t)_{t \geq 0}$ is a Markov chain (irreducible, aperiodic, Harris recurrent) by Markov chain ergodic theorem.

Online MF as Empirical Loss Minimization

- Fix $\lambda > 0$ and define the following the **quadratic loss function**

$$\ell(X, W) = \inf_{H \in \mathcal{C}' \subseteq \mathbb{R}^{r \times n},\, S \in \mathbb{R}^{d \times n}} \|X - WH - S\|_F^2 + \lambda_1 \|H\|_1 + \lambda_2 \|S\|_1$$

  Define the **expected loss** and **empirical loss** functions

$$f(W) = \mathbb{E}_{X \sim \pi}[\ell(X, W)], \qquad f_t(W) = \frac{1}{t} \sum_{s=1}^{t} \ell(X_s, W)$$

- If $(X_t)_{t \geq 0}$ is i.i.d. with common distribution π, then by SLLN,

$$\lim_{t \to \infty} f_t(W) = f(W) \quad \text{a.s. for all } W \in \mathcal{C}.$$

- Same holds if $(X_t)_{t \geq 0}$ is a Markov chain (irreducible, aperiodic, Harris recurrent) by Markov chain ergodic theorem.
- Furthermore, for $\mathcal{C}$ compact, by Glivenko-Cantelli

$$\lim_{t \to \infty} \sup_{W \in \mathcal{C}} \|f_t(W) - f(W)\| \to 0 \quad \text{a.s.}$$

Online MF as Empirical Loss Minimization

- Fix $\lambda > 0$ and define the following the **quadratic loss function**

$$\ell(X, W) = \inf_{H \in \mathcal{C}' \subseteq \mathbb{R}^{r \times n}} \|X - WH - S\|_F^2 + \lambda \|H\|_1,$$

Define the **expected loss** and **empirical loss** functions

$$f(W) = \mathbb{E}_{X \sim \pi}[\ell(X, W)], \qquad f_t(W) = \frac{1}{t} \sum_{s=1}^{t} \ell(X_s, W)$$

- Empirical Loss (Risk) Minimization for Online RMF:

Input: (Markovian) Sequence of data matrices $(X_t)_{t \geq 0}$, $X_t \sim \pi$.

Objective: $$W_t = \operatorname{argmin}_{W \in \mathcal{C}} f_t(W)$$

Online MF as Empirical Loss Minimization

- Fix $\lambda > 0$ and define the following the **quadratic loss function**

$$\ell(X, W) = \inf_{H \in \mathcal{C}' \subseteq \mathbb{R}^{r \times n}} \|X - WH - S\|_F^2 + \lambda \|H\|_1,$$

  Define the **expected loss** and **empirical loss** functions

$$f(W) = \mathbb{E}_{X \sim \pi}[\ell(X, W)], \qquad f_t(W) = \frac{1}{t} \sum_{s=1}^{t} \ell(X_s, W)$$

- Empirical Loss (Risk) Minimization for Online RMF:

  Input: (Markovian) Sequence of data matrices $(X_t)_{t \geq 0}$, $X_t \sim \pi$.

  Objective: $W_t = \text{argmin}_{W \in \mathcal{C}} f_t(W)$

- But how do we minimize the empirical loss f_t?
  - f_t is non-convex
  - Each $\ell(X_s, W)$ involves separate optimization
  - Need to store all data $X_1, \cdots, X_t$.

Asymptotic solution minimizing surrogate loss function

- Online surrogate optimization algorithm:

$$\textit{Given } X_t\text{:} \quad \begin{cases} (H_t, S_t) = \operatorname{argmin}_{H\in\mathcal{C}'} \|X_t - W_{t-1}H - S\|_F^2 + \lambda_1\|H\|_1 + \lambda_2\|S\|_1 \\ W_t = \operatorname{argmin}_{W\in\mathcal{C}} \hat{f}_t(W), \end{cases}$$

where $\hat{f}_t(W)$ is a **surrogate loss** defined by

$$(f_t(W) \leq) \qquad \hat{f}_t(W) = \frac{1}{t}\sum_{s=1}^{t}(\|X_s - WH_s - S\|_F^2 + \lambda_1\|H_s\|_1 + \lambda_2\|S_s\|_1).$$

Asymptotic solution minimizing surrogate loss function

- Online surrogate optimization algorithm:

$$\textit{Given } X_t\text{:}\quad \begin{cases} (H_t, S_t) = \mathrm{argmin}_{H\in\mathcal{C}'} \|X_t - W_{t-1}H - S\|_F^2 + \lambda_1\|H\|_1 + \lambda_2\|S\|_1 \\ W_t = \mathrm{argmin}_{W\in\mathcal{C}} \hat{f}_t(W), \end{cases}$$

where $\hat{f}_t(W)$ is a **surrogate loss** defined by

$$(f_t(W) \leq) \qquad \hat{f}_t(W) = \frac{1}{t}\sum_{s=1}^{t}(\|X_s - WH_s - S\|_F^2 + \lambda_1\|H_s\|_1 + \lambda_2\|S_s\|_1).$$

- Recycle the previously found codes $H_1, \cdots, H_t$ and outliers $S_1, \cdots, S_t$ and use them as approximate solutions of the sub-problems.

Asymptotic solution minimizing surrogate loss function

- Online surrogate optimization algorithm:

$$\textit{Given } X_t\text{:} \quad \begin{cases} (H_t, S_t) = \operatorname{argmin}_{H \in \mathcal{C}'} \|X_t - W_{t-1}H - S\|_F^2 + \lambda_1 \|H\|_1 + \lambda_2 \|S\|_1 \\ W_t = \operatorname{argmin}_{W \in \mathcal{C}} \hat{f}_t(W), \end{cases}$$

where $\hat{f}_t(W)$ is a **surrogate loss** defined by

$$(f_t(W) \leq) \qquad \hat{f}_t(W) = \frac{1}{t}\sum_{s=1}^{t}(\|X_s - WH_s - S\|_F^2 + \lambda_1\|H_s\|_1 + \lambda_2\|S_s\|_1).$$

- Recycle the previously found codes $H_1, \cdots, H_t$ and outliers $S_1, \cdots, S_t$ and use them as approximate solutions of the sub-problems.
- Block optimization + Majorization - Minimization (MM) + Convex relaxation

Asymptotic solution minimizing surrogate loss function

- Online surrogate optimization algorithm:

$$\textit{Given } X_t\text{:} \quad \begin{cases} (H_t, S_t) = \operatorname{argmin}_{H\in\mathcal{C}'} \|X_t - W_{t-1}H - S\|_F^2 + \lambda_1\|H\|_1 + \lambda_2\|S\|_1 \\ W_t = \operatorname{argmin}_{W\in\mathcal{C}} \hat{f}_t(W), \end{cases}$$

where $\hat{f}_t(W)$ is a **surrogate loss** defined by

$$(f_t(W) \leq) \qquad \hat{f}_t(W) = \frac{1}{t}\sum_{s=1}^{t}(\|X_s - WH_s - S\|_F^2 + \lambda_1\|H_s\|_1 + \lambda_2\|S_s\|_1).$$

- Recycle the previously found codes $H_1, \cdots, H_t$ and outliers $S_1, \cdots, S_t$ and use them as approximate solutions of the sub-problems.
- Block optimization + Majorization - Minimization (MM) + Convex relaxation
- $W_t = \operatorname{argmin}_W \operatorname{tr}(WA_tW^T) - 2\operatorname{tr}(WB_t)$ for summary matrices A_t, B_t

Solving joint sparse coding problem

- We solve the following joint sparse coding problem by proximal gradient:

$$(H_t, S_t) = \operatorname{argmin}_{H \in \mathcal{C}'} \|X_t - W_{t-1}H - S\|_F^2 + \lambda_1 \|H\|_1 + \lambda_2 \|S\|_1 \qquad (1)$$

Solving joint sparse coding problem

- We solve the following joint sparse coding problem by proximal gradient:

$$(H_t, S_t) = \operatorname{argmin}_{H \in \mathcal{C}'} \|X_t - W_{t-1}H - S\|_F^2 + \lambda_1 \|H\|_1 + \lambda_2 \|S\|_1 \qquad (1)$$

- Fix $W_{t-1} \in \mathbb{R}^{d \times n}$ and parameters $\alpha, \beta > 0$. Define a $d \times (r+d)$ matrix

$$G_{t-1} = [W_{t-1}, \beta I_d]. \qquad (2)$$

Consider the following constrained LASSO problem

$$V_t = \operatorname*{argmin}_{\substack{V=[H,S'] \\ (H,S') \in \mathcal{C}^{\text{code}} \times \mathbb{R}^{d \times n}}} \|X_t - G_{t-1}V\|_F^2 + \alpha \|V\|_1. \qquad (3)$$

Solving joint sparse coding problem

- We solve the following joint sparse coding problem by proximal gradient:

$$(H_t, S_t) = \operatorname{argmin}_{H \in \mathcal{C}'} \|X_t - W_{t-1}H - S\|_F^2 + \lambda_1 \|H\|_1 + \lambda_2 \|S\|_1 \qquad (1)$$

- Fix $W_{t-1} \in \mathbb{R}^{d \times n}$ and parameters $\alpha, \beta > 0$. Define a $d \times (r + d)$ matrix

$$G_{t-1} = [W_{t-1}, \beta I_d]. \qquad (2)$$

  Consider the following constrained LASSO problem

$$V_t = \operatorname{argmin}_{\substack{V=[H,S'] \\ (H,S') \in \mathcal{C}^{\text{code}} \times \mathbb{R}^{d \times n}}} \|X_t - G_{t-1}V\|_F^2 + \alpha \|V\|_1. \qquad (3)$$

- Equivalent to the original problem for the choice $\alpha = \lambda_1$ and $\beta = \lambda_1/\lambda_2$:

$$\begin{aligned} \|X_t - G_{t-1}V\|_F^2 + \alpha\|V\|_1 &= \|X_t - W_{t-1}H - \beta S'\|_F^2 + \alpha\|H\|_1 + \alpha\|S'\|_1 \\ &= \|X_t - W_{t-1}H - S\|_F^2 + \alpha\|H\|_1 + (\alpha/\beta)\|S\|_1, \end{aligned}$$

  with change of variable $S = \beta S'$.

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

Theorem (Cai, **Lyu**, Needell '20+)

Suppose $(X_t)_{t\geq 0}$ is a Hidden Markov chain (irreducible, aperiodic, finite state). Let $(W_t, H_t, S_t)_{t\geq 1}$ be a solution to the ORMF algorithm before.

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

Theorem (Cai, **Lyu**, Needell '20+)

Suppose $(X_t)_{t \geq 0}$ is a Hidden Markov chain (irreducible, aperiodic, finite state). Let $(W_t, H_t, S_t)_{t \geq 1}$ be a solution to the ORMF algorithm before.

(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) $W_t \to$ *Set of critical points of* f *as* $t \to \infty$ *almost surely.*

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

Theorem (Cai, **Lyu**, Needell '20+)

Suppose $(X_t)_{t\geq 0}$ is a Hidden Markov chain (irreducible, aperiodic, finite state). Let $(W_t, H_t, S_t)_{t\geq 1}$ be a solution to the ORMF algorithm before.

(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) $W_t \to$ *Set of critical points of* f *as* $t \to \infty$ *almost surely.*

- IDEA: Condition on distant past + Control conditional error by MC mixing + Control unconditional error by MC uniform functional CLT

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

Theorem (Cai, **Lyu**, Needell '20+)

Suppose $(X_t)_{t\geq 0}$ is a Hidden Markov chain (irreducible, aperiodic, finite state). Let $(W_t, H_t, S_t)_{t\geq 1}$ be a solution to the ORMF algorithm before.

(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) $W_t \to$ *Set of critical points of* f *as* $t \to \infty$ *almost surely.*

- IDEA: Condition on distant past + Control conditional error by MC mixing + Control unconditional error by MC uniform functional CLT
- **First** convergence result for ORMF algorithms for Markovian input (even i.i.d.)

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

Theorem (Cai, **Lyu**, Needell '20+)

Suppose $(X_t)_{t\geq 0}$ is a Hidden Markov chain (irreducible, aperiodic, finite state). Let $(W_t, H_t, S_t)_{t\geq 1}$ be a solution to the ORMF algorithm before.

(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) $W_t \to$ *Set of critical points of* f *as* $t \to \infty$ *almost surely.*

- IDEA: Condition on distant past + Control conditional error by MC mixing + Control unconditional error by MC uniform functional CLT
- **First** convergence result for ORMF algorithms for Markovian input (even i.i.d.)
- A similar result was obtained for non-robust version by Lyu, Needell, and Balzano 19' for dependent data matrices.

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

Theorem (Cai, **Lyu**, Needell '20+)

Suppose $(X_t)_{t\geq 0}$ *is a Hidden Markov chain (irreducible, aperiodic, finite state). Let* $(W_t, H_t, S_t)_{t\geq 1}$ *be a solution to the ORMF algorithm before.*

(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) $W_t \to$ *Set of critical points of* f *as* $t \to \infty$ *almost surely.*

- IDEA: Condition on distant past + Control conditional error by MC mixing + Control unconditional error by MC uniform functional CLT
- **First** convergence result for ORMF algorithms for Markovian input (even i.i.d.)
- A similar result was obtained for non-robust version by Lyu, Needell, and Balzano 19' for dependent data matrices.
- The first result of this kind was obtained for non-robust version by MBPS 10' for i.i.d. data matrices.

Notations

- Fix $\lambda > 0$ and define the following the **quadratic loss function**

$$\ell(X, W) = \inf_{H \in \mathcal{C} \subseteq \mathbb{R}^{r \times n}, S \in \mathbb{R}^{r \times d}} \|X - WH - S\|_F^2 + \lambda_1 \|H\|_1 + \lambda_2 \|S\|_1,$$

Define the **expected loss** and **empirical loss** functions

$$f(W) = \mathbb{E}_{X \sim \pi}[\ell(X, W)], \qquad f_t(W) = \frac{1}{t} \sum_{s=1}^{t} \ell(X_s, W)$$

Notations

- Fix $\lambda > 0$ and define the following the **quadratic loss function**

$$\ell(X, W) = \inf_{H \in \mathcal{C} \subseteq \mathbb{R}^{r \times n}, S \in \mathbb{R}^{r \times d}} \|X - WH - S\|_F^2 + \lambda_1 \|H\|_1 + \lambda_2 \|S\|_1,$$

  Define the **expected loss** and **empirical loss** functions

$$f(W) = \mathbb{E}_{X \sim \pi}[\ell(X, W)], \qquad f_t(W) = \frac{1}{t} \sum_{s=1}^{t} \ell(X_s, W)$$

- Online surrogate optimization algorithm:

$$\textit{Given } X_t\text{:} \quad \begin{cases} (H_t, S_t) = \operatorname{argmin}_{H \in \mathcal{C}'} \|X_t - W_{t-1} H - S\|_F^2 + \lambda_1 \|H\|_1 + \lambda_2 \|S\|_1 \\ W_t = \operatorname{argmin}_{W \in \mathcal{C}} \hat{f}_t(W), \end{cases}$$

  where $\hat{f}_t(W)$ is a **surrogate loss** defined by

$$(f_t(W) \leq) \qquad \hat{f}_t(W) = \frac{1}{t} \sum_{s=1}^{t} (\|X_s - WH_s - S\|_F^2 + \lambda_1 \|H_s\|_1 + \lambda_2 \|S_s\|_1).$$

Notations

- Fix $\lambda > 0$ and define the following the **quadratic loss function**

$$\ell(X, W) = \inf_{H \in \mathcal{C} \subseteq \mathbb{R}^{r \times n}, S \in \mathbb{R}^{r \times d}} \|X - WH - S\|_F^2 + \lambda_1 \|H\|_1 + \lambda_2 \|S\|_1,$$

  Define the **expected loss** and **empirical loss** functions

$$f(W) = \mathbb{E}_{X \sim \pi}[\ell(X, W)], \qquad f_t(W) = \frac{1}{t} \sum_{s=1}^{t} \ell(X_s, W)$$

- Online surrogate optimization algorithm:

$$\textit{Given } X_t\textit{:} \quad \begin{cases} (H_t, S_t) = \operatorname{argmin}_{H \in \mathcal{C}'} \|X_t - W_{t-1} H - S\|_F^2 + \lambda_1 \|H\|_1 + \lambda_2 \|S\|_1 \\ W_t = \operatorname{argmin}_{W \in \mathcal{C}} \hat{f}_t(W), \end{cases}$$

  where $\hat{f}_t(W)$ is a **surrogate loss** defined by

$$(f_t(W) \leq) \qquad \hat{f}_t(W) = \frac{1}{t} \sum_{s=1}^{t} (\|X_s - WH_s - S\|_F^2 + \lambda_1 \|H_s\|_1 + \lambda_2 \|S_s\|_1).$$

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

Proposition

(i) $\hat{f}_{t+1}(W_{t+1}) - \hat{f}_t(W_t) \le \frac{1}{t+1}\left(\ell(X_{t+1}, W_t) - f_t(W_t)\right)$.

(ii) $0 \le \frac{1}{t+1}\left(\hat{f}_t(W_t) - f_t(W_t)\right) \le \frac{1}{t+1}\left(\ell(X_{t+1}, W_t) - f_t(W_t)\right) + \hat{f}_t(W_t) - \hat{f}_{t+1}(W_{t+1})$.

Sketch of main argument:

- $\sum_{t=0}^{\infty} \mathbb{E}\left[\frac{1}{t+1}\left(\ell(X_{t+1}, W_t) - f_t(W_t)\right)^+\right] < \infty$ implies $\mathbb{E}[\hat{f}_t(W_t)]$ converges.
- $\sum_{t=0}^{\infty} \mathbb{E}\left[\frac{1}{t+1}\left(\hat{f}_t(W_t) - f_t(W_t)\right)\right] < \infty$ implies $\hat{f}_t(W_t) - f_t(W_t) \to 0$ a.s.
- $f_t \le \hat{f}_t$, $W_t = \operatorname{argmin} \hat{f}_t$, $\hat{f}_t(W_t) - f_t(W_t) \to 0$ a.s. imply

$$W_t \to \text{Set of critical points of } f \text{ a.s.}$$

Suffices to show
$$\sum_{t=0}^{\infty} \left| \mathbb{E}\left[\frac{1}{t+1}\left(\ell(X_{t+1}, W_t) - f_t(W_t)\right)\right]\right| < \infty$$

Key estimate in the i.i.d. case

- Suffices to show $\sum_{t=0}^{\infty} \left| \mathbb{E}\left[\frac{1}{t+1} \left(\ell(X_{t+1}, W_t) - f_t(W_t) \right) \right] \right| < \infty$

Key estimate in the i.i.d. case

- Suffices to show $\sum_{t=0}^{\infty} \left| \mathbb{E}\left[\frac{1}{t+1} \left(\ell(X_{t+1}, W_t) - f_t(W_t) \right) \right] \right| < \infty$
- Suppose data matrices X_t are i.i.d. and let $\mathcal{F}_t$ denote the information up to time t. Then

$$\left| \mathbb{E}\left[\ell(X_{t+1}, W_t) - f_t(W_t) \,\middle|\, \mathcal{F}_t \right] \right| \le \left| \mathbb{E}_{X\sim\pi}[\ell(X, W_t)] - f_t(W_t) \right|$$
$$= |f(W_t) - f_t(W_t)| \le \|f - f_t\|_\infty$$

Key estimate in the i.i.d. case

- Suffices to show $\sum_{t=0}^{\infty} \left| \mathbb{E}\left[\frac{1}{t+1} \left(\ell(X_{t+1}, W_t) - f_t(W_t) \right) \right] \right| < \infty$
- Suppose data matrices X_t are i.i.d. and let $\mathcal{F}_t$ denote the information up to time t. Then

$$\left| \mathbb{E}\left[\ell(X_{t+1}, W_t) - f_t(W_t) \,\middle|\, \mathcal{F}_t \right] \right| \leq \left| \mathbb{E}_{X\sim\pi}[\ell(X, W_t)] - f_t(W_t) \right| \\ = |f(W_t) - f_t(W_t)| \leq \|f - f_t\|_\infty$$

- $\|f - f_t\|_\infty \to 0$ Glivenko-Cantelli Thm. ($W \in$ Compact set)

Key estimate in the i.i.d. case

- Suffices to show $\sum_{t=0}^{\infty} \left| \mathbb{E}\left[\frac{1}{t+1} \left(\ell(X_{t+1}, W_t) - f_t(W_t) \right) \right] \right| < \infty$
- Suppose data matrices X_t are i.i.d. and let $\mathcal{F}_t$ denote the information up to time t. Then

$$\begin{aligned} \left| \mathbb{E}\left[\ell(X_{t+1}, W_t) - f_t(W_t) \,\middle|\, \mathcal{F}_t \right] \right| &\leq |\mathbb{E}_{X\sim\pi}[\ell(X, W_t)] - f_t(W_t)| \\ &= |f(W_t) - f_t(W_t)| \leq \|f - f_t\|_\infty \end{aligned}$$

- $\|f - f_t\|_\infty \to 0$ Glivenko-Cantelli Thm. ($W \in$ Compact set)
- $\mathbb{E}[t^{1/2}\|f - f_t\|_\infty] = O(1)$ by uniform functional CLT

Key estimate in the i.i.d. case

- Suffices to show $\sum_{t=0}^{\infty} \left| \mathbb{E}\left[\frac{1}{t+1} \left(\ell(X_{t+1}, W_t) - f_t(W_t) \right) \right] \right| < \infty$
- Suppose data matrices X_t are i.i.d. and let $\mathcal{F}_t$ denote the information up to time t. Then

$$\begin{aligned} \left| \mathbb{E}\left[\ell(X_{t+1}, W_t) - f_t(W_t) \,\middle|\, \mathcal{F}_t \right] \right| &\leq \left| \mathbb{E}_{X\sim\pi}[\ell(X, W_t)] - f_t(W_t) \right| \\ &= |f(W_t) - f_t(W_t)| \leq \|f - f_t\|_\infty \end{aligned}$$

- $\|f - f_t\|_\infty \to 0$ Glivenko-Cantelli Thm. ($W \in$ Compact set)
- $\mathbb{E}[t^{1/2}\|f - f_t\|_\infty] = O(1)$ by uniform functional CLT
- Averaging over $\mathcal{F}_t$, this gives

$$\begin{aligned} \left| \mathbb{E}\left[\frac{1}{t+1} \left(\ell(X_{t+1}, W_t) - f_t(W_t) \right) \right] \right| &\leq \mathbb{E}\left[\left| \mathbb{E}\left[\frac{\left(\ell(X_{t+1}, W_t) - f_t(W_t) \right)}{t+1} \,\middle|\, \mathcal{F}_t \right] \right| \right] \\ &\leq t^{-3/2}\mathbb{E}[t^{1/2}\|f - f_t\|_\infty] \\ &= O(t^{-3/2}). \end{aligned}$$

Key estimate in the Markovian case

- If $(X_t)_{t\geq 0}$ is Markovian, then

$$\mathbb{E}[\ell(X_{t+1}, W) \,|\, \mathcal{F}_t] \neq \mathbb{E}_{X\sim\pi}[\ell(X, W)] = f(W_t).$$

Key estimate in the Markovian case

- If $(X_t)_{t\geq 0}$ is Markovian, then

$$\mathbb{E}[\ell(X_{t+1}, W) \,|\, \mathcal{F}_t] \neq \mathbb{E}_{X\sim\pi}[\ell(X, W)] = f(W_t).$$

- Instead, **condition on a distant past** $\mathcal{F}_{t-N}$ and see how much the chain mixes to the stationary distribution during $[t-N, t]$.

$$\left|\mathbb{E}[\ell(X_{t+1}, W) \,|\, \mathcal{F}_{t-N}] - f(W)\right| \leq 2\|\ell(\cdot, W)\|_\infty \|P^{N+1}(\mathbf{x}, \cdot) - \pi\|_{TV}.$$

Key estimate in the Markovian case

- If $(X_t)_{t\geq 0}$ is Markovian, then

$$\mathbb{E}[\ell(X_{t+1}, W) \,|\, \mathcal{F}_t] \neq \mathbb{E}_{X\sim\pi}[\ell(X, W)] = f(W_t).$$

- Instead, **condition on a distant past** $\mathcal{F}_{t-N}$ and see how much the chain mixes to the stationary distribution during $[t-N, t]$.

$$\left| \mathbb{E}[\ell(X_{t+1}, W) \,|\, \mathcal{F}_{t-N}] - f(W) \right| \leq 2\|\ell(\cdot, W)\|_\infty \|P^{N+1}(\mathbf{x}, \cdot) - \pi\|_{TV}.$$

- The TV distance decays exponentially in N

Key estimate in the Markovian case

- If $(X_t)_{t\geq 0}$ is Markovian, then

$$\mathbb{E}[\ell(X_{t+1}, W) \,|\, \mathcal{F}_t] \neq \mathbb{E}_{X\sim\pi}[\ell(X, W)] = f(W_t).$$

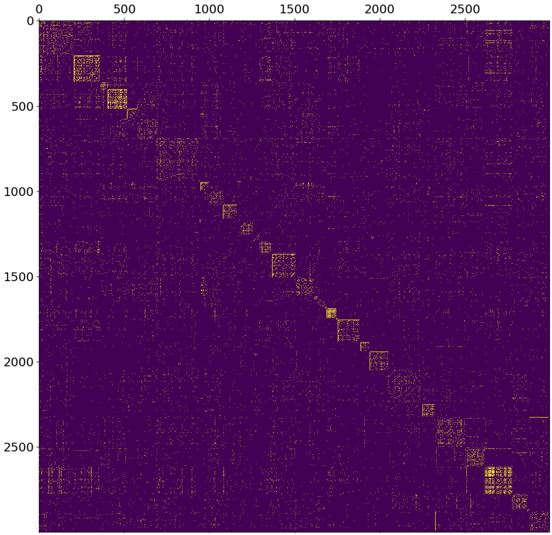
- Instead, **condition on a distant past** $\mathcal{F}_{t-N}$ and see how much the chain mixes to the stationary distribution during $[t-N, t]$.

$$\left| \mathbb{E}[\ell(X_{t+1}, W) \,|\, \mathcal{F}_{t-N}] - f(W) \right| \leq 2\|\ell(\cdot, W)\|_\infty \|P^{N+1}(\mathbf{x}, \cdot) - \pi\|_{TV}.$$

- The TV distance decays exponentially in N
- Choose $N = N(t)$ appropriately and average over $\mathcal{F}_{t-N}$.

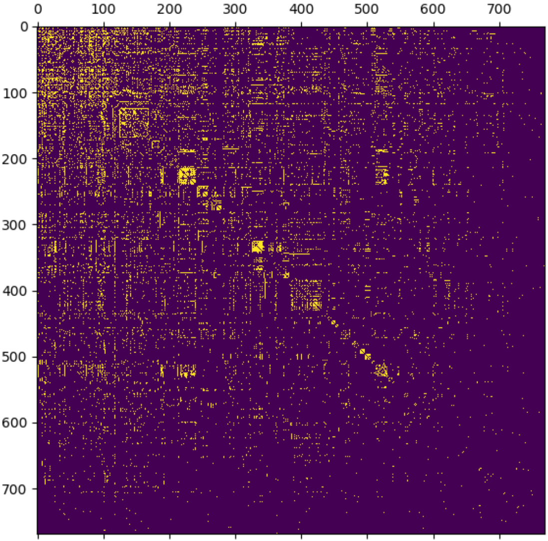
3. Applications: Dictionary learning from Facebook networks

Facebook100 network data - UCLA26



- Traud, Mucha, Porter '12
- Snapshot of UCLA FB ntwk on Sep. 2005
- (i, j)-entry $=$ $\mathbf{1}$(user i and j are friends)
- Number of nodes $= 20467$
- Number of edges $= 747613$
- Edge density $= 0.00357$
- Figure shows only the network on first 3000 nodes

Facebook100 network data - Caltech36



- Traud, Mucha, Porter '12
- Snapshot of Caltech FB ntwk on Sep. 2005
- (i, j)-entry $=$ $\mathbf{1}$(user i and j are friends)
- Number of nodes $= 769$
- Number of edges $= 8328$
- Edge density $= 0.05640$

Cycle (1938) by M.C. Escher

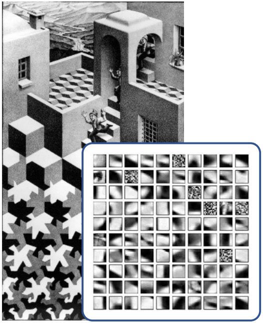
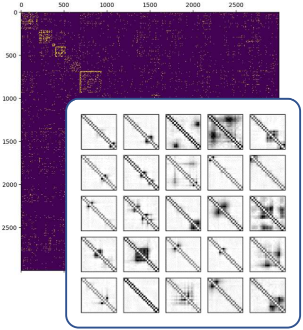


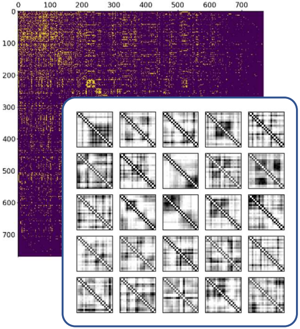
Image Dictionary

UCLA26 Facebook network



Network Dictionary

Caltech36 Facebook network



Network Dictionary

Main question: Can we learn parts of networks like we do for the images?

Cycle (1938) by M.C. Escher

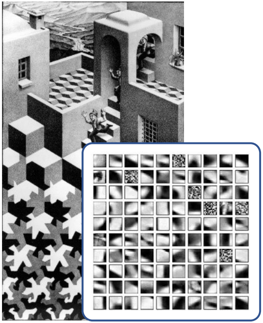
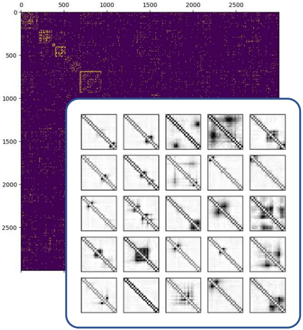


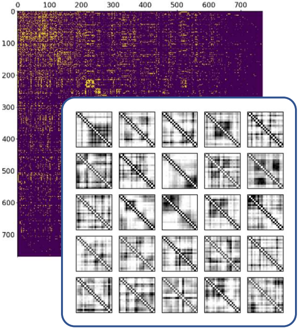
Image Dictionary

UCLA26 Facebook network



Network Dictionary

Caltech36 Facebook network



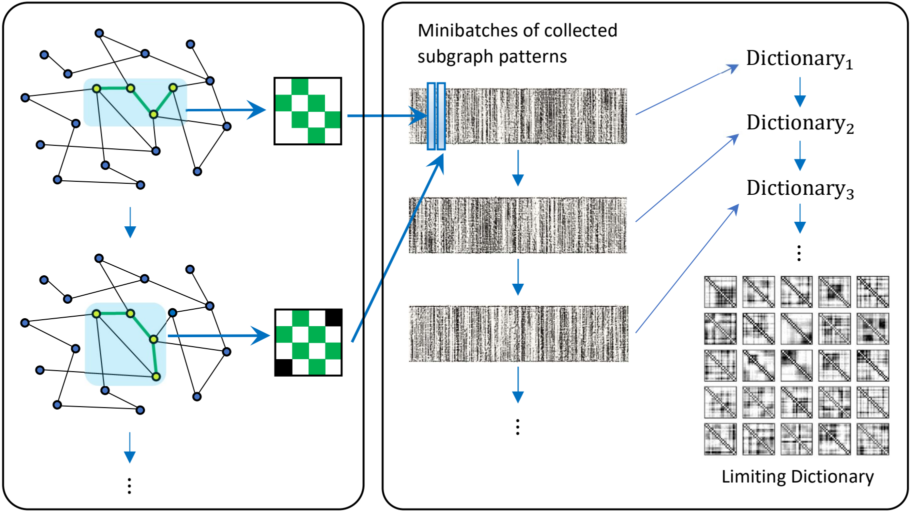
Network Dictionary

Main question: Can we learn parts of networks like we do for the images?

Answer: Network Dictionary Learning (**Lyu**, Needell, and Balzano '19)

- Theoretical background: MCMC, motif sampling, Markov chains, Optimizaion, Online Matrix Factorization.
- Applications: Network + (compression, completion, comparison, classification, visualization, inference)

MCMC motif sampling + OMF dictionary learning

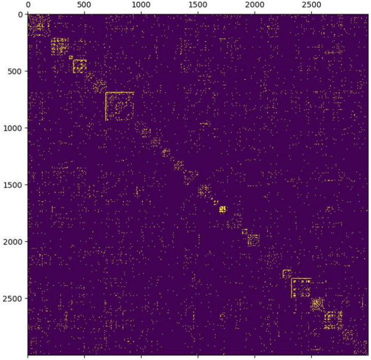


MCMC Motif sampling from network

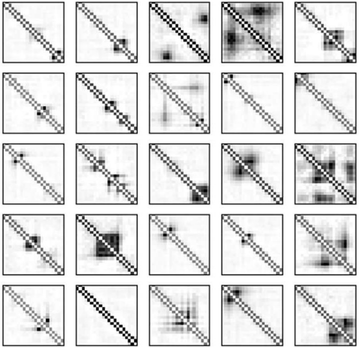
Online Nonnegative Matrix Factorization

Network Dictionary Learning – UCLA26

Original UCLA26 FB Ntwk



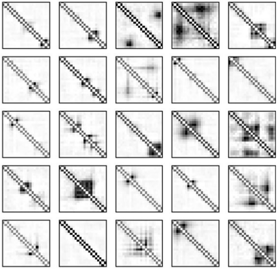
25 Dictionary of size 21
learned from UCLA26 FB ntwk



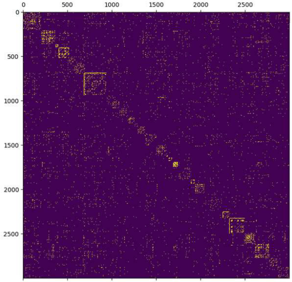
- Extract k-node subgraph patterns by k-chain motif sampling from UCLA26
- Let $k = 21$, so that $\dim(\text{all subgraph patterns}) = \binom{21}{2} - 20 = 200$.
- On the right: rank-25 (approximate) basis for subgraph patterns in UCLA26

Network Dictionary Learning – Reconstructing UCLA from UCLA

25 Dictionary learned from UCLA26 FB ntwk



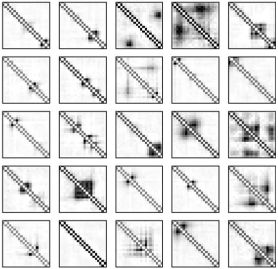
Original UCLA26 FB Ntwk



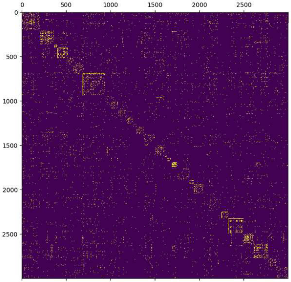
- Can we reconstruct the original network using the learned dictionary?

Learning parts of networks – Reconstructing UCLA from UCLA

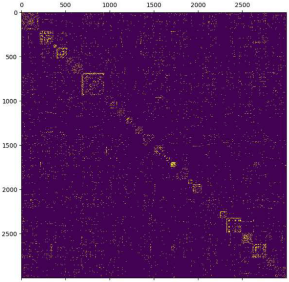
25 Dictionary learned from UCLA26 FB ntwk



Original UCLA26 FB Ntwk



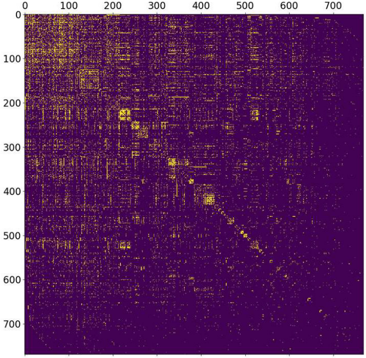
Reconstructed UCLA Ntwk using Dict. learned from UCLA



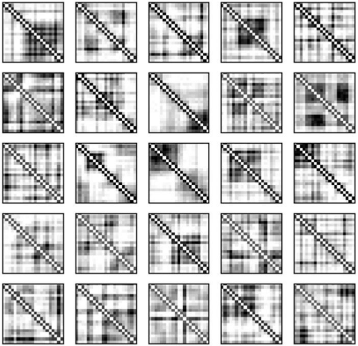
- 95% of reconstruction accuracy (# common edges)/(# edges in original)
- Ntwk recons. = (local approx. by dict.) + (Averaging) + (Rounding)

Network Dictionary Learning – Caltech36

Original Caltech FB Ntwk



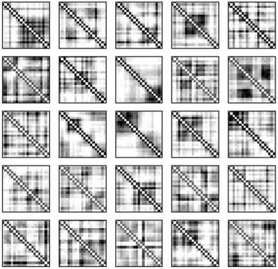
25 Dictionary of size 21
learned from Caltech36 FB ntwk



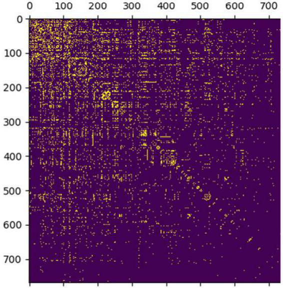
- Extract k-node subgraph patterns by k-chain motif sampling from Caltech36
- We choose $k = 21$, so that $\dim(\text{all subgraph patterns}) = \binom{21}{2} - 20 = 200$.
- On the right: rank-25 (approximate) basis for subgraph patterns in Caltech36

Network Dictionary Learning – Reconstructing Caltech from Caltech

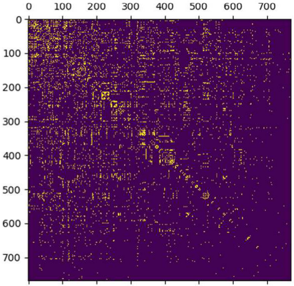
25 Dictionary learned from Caltech36 FB ntwk



Original Caltech36 FB Ntwk

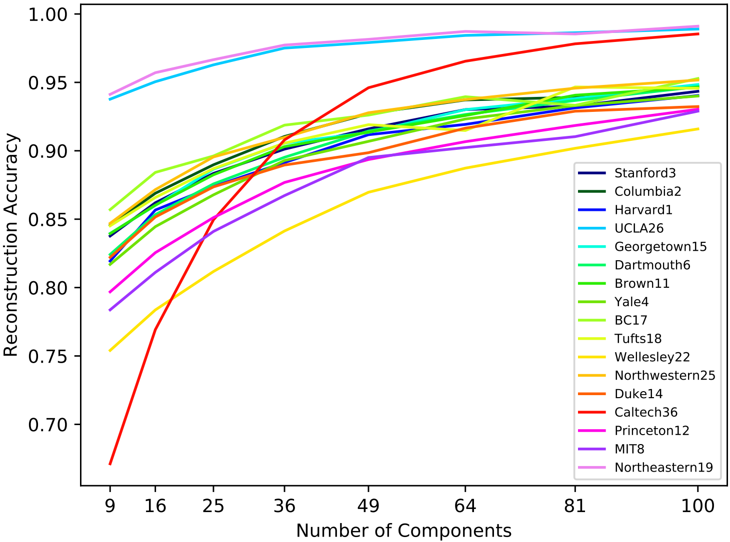


Recons. Caltech ntwk using Dict. learned from Caltech



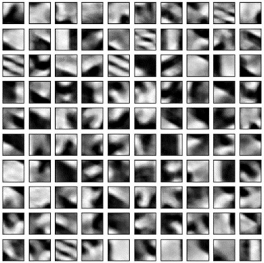
- 85% of reconstruction accuracy (# common edges)/(# edges in original)
- Ntwk recons. = (local approx. by dict.) + (Averaging) + (Rounding)

Network Dictionary Learning - Self-reconstruction accuracies

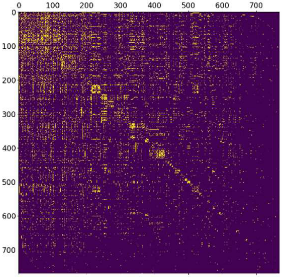


Learning parts of networks – Reconstructing Caltech from Escher

100 Dictionary learned from
Cycle by M. C. Escher



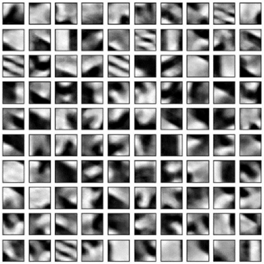
Original Caltech FB Ntwk



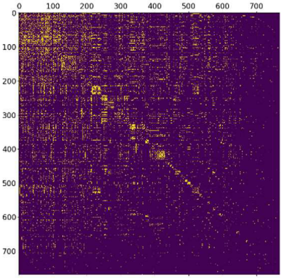
- Can we use dictionary learned from Escher to reconstruct Caltech?

Learning parts of networks – Reconstructing Caltech from Escher

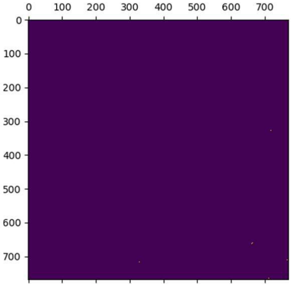
100 Dictionary learned from Cycle by M. C. Escher



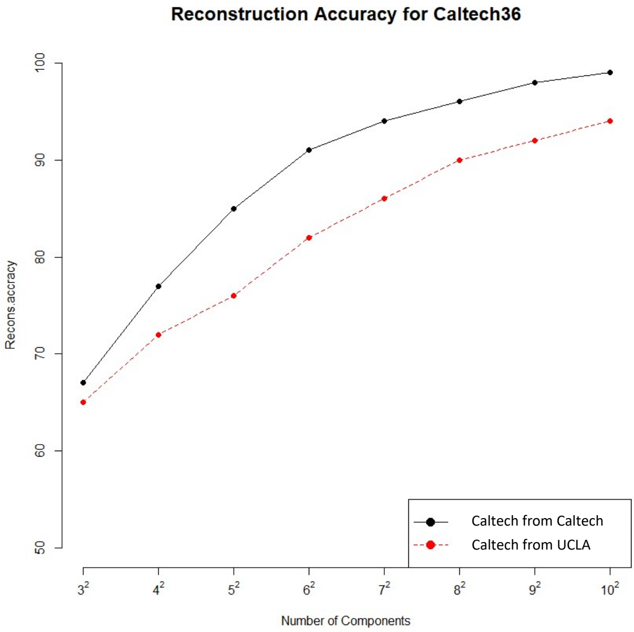
Original Caltech FB Ntwk



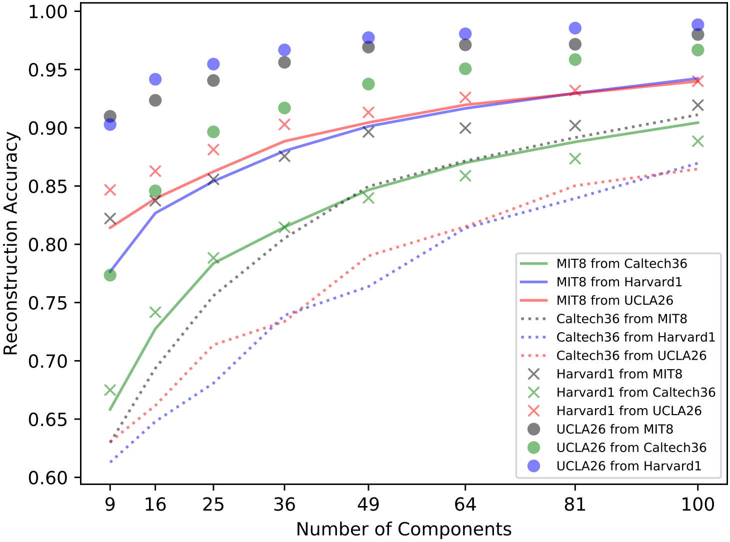
Reconstructed Caltech Ntwk using Dict. learned from Escher



- # edges in original ntwk = 16656
- # edges in reconstructed ntwk = 34
- # common edges = 0. (Zero reconstruction accuracy)
- Non-example of transfer learning


Reconstruction Accuracy for Caltech36
Recons.accracy
50
60
70
80
90
100
3^2
4^2
5^2
6^2
7^2
8^2
9^2
10^2
Number of Components
Caltech from Caltech
Caltech from UCLA

Network Dictionary Learning - Cross-reconstruction accuracies



Related current/future works

1. **Applications/Implications of Network Dictionary Learning**
   - Completion, inference, and transfer learning for social network data (joint with Kureh and Porter)
   - Edge completion, outlier detection on networks
2. **Deep neural networks + Matrix factorization**
   - Topic-aware chatbot using Recurrent NN and NMF (joint with summer REU students and Needell)
3. **Learning parts of tensor data**
   - Hyper-motif sampling from hyper-networks
   - Online tensor factorization for Markovian data (joint with Needell, Strohmeier) (c.f., no convergence known even for the i.i.d. case)

   Applications: Dict. learning for video, and trajectory of evolving networks, dynamic topic modeling
4. **Further extension of Online Matrix Factorization**
   - OMF for variable number of dictionaries (added optimization dimension)
   - OMF for non-stationary data matrices (what do we want to learn in this case?)

Thanks!