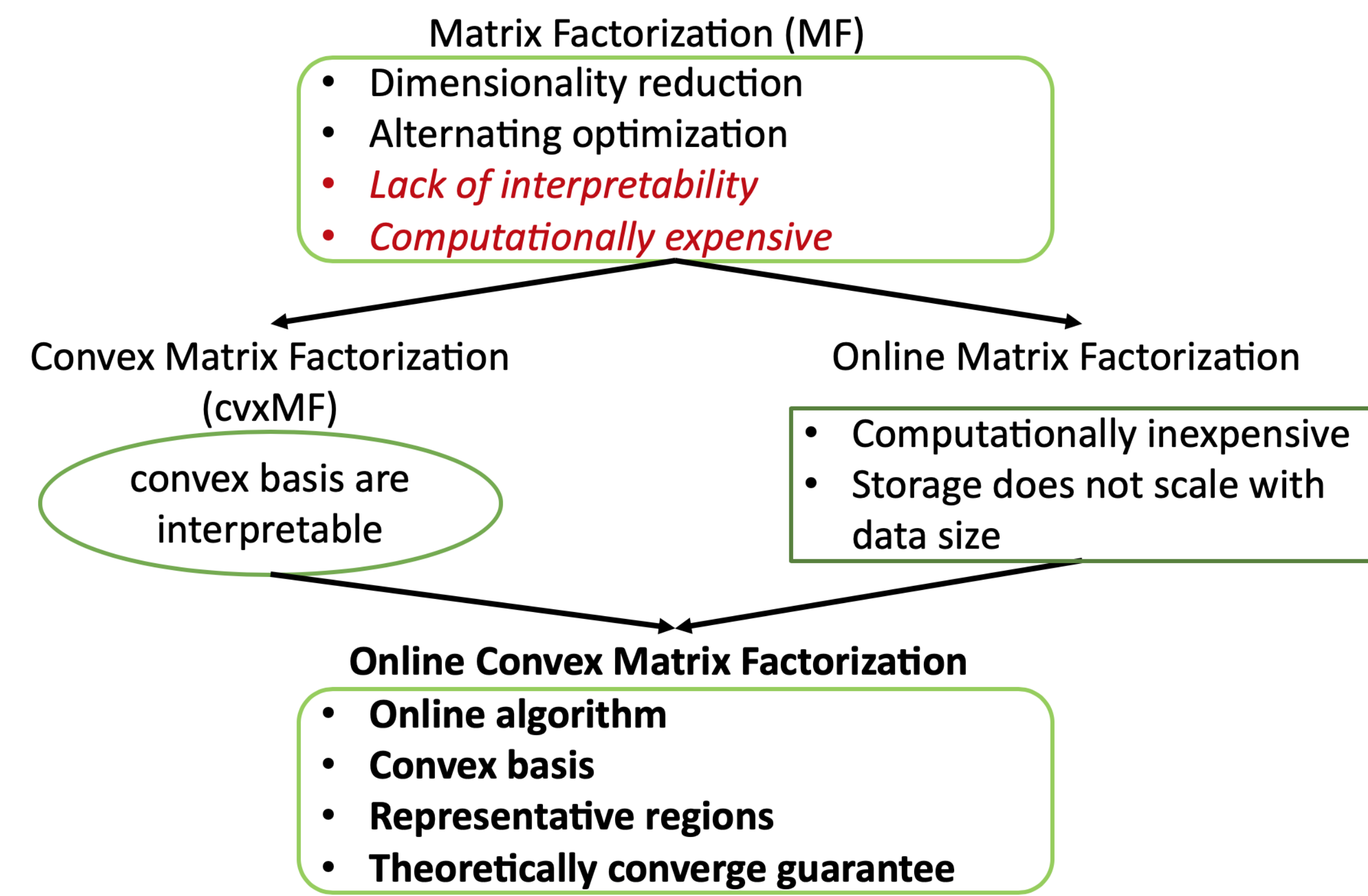# Online Convex Matrix Factorization with Representative Regions

Abhishek Agarwal, Jianhao Peng, Olgica Milenkovic

Electrical and Computer Engineering, University of Illinois at Urbana-Champaign

Matrix Factorization Problem

Matrix Factorization (MF)
- Dimensionality reduction
- Alternating optimization
- *Lack of interpretability*
- *Computationally expensive*

Convex Matrix Factorization (cvxMF)
- convex basis are interpretable

Online Matrix Factorization
- Computationally inexpensive
- Storage does not scale with data size

Online Convex Matrix Factorization
- Online algorithm
- Convex basis
- Representative regions
- Theoretically converge guarantee

Stochastic Loss function :
$$g(\mathbf{D}) \triangleq \mathbb{E}\left[\min_{\alpha} \|X - \mathbf{D}\alpha\|^2 + \lambda\|\alpha\|_1\right]$$

$X \in \Omega \subset \mathbb{R}^m$: Random Variable input

$\mathbf{D} \in \mathbb{R}^{m \times k}$: Basis Vector matrix

$\alpha \in \mathbb{R}^k$: Coefficient vector corresponding to input X

$\lambda \in \mathbb{R}$: Sparsity parameter

Approach and Challenges

Let $\ell(x_t, \mathbf{D}, \alpha) \triangleq \|x_t - \mathbf{D}\alpha\|^2 + \lambda\|\alpha\|_1$ be the loss for a single input x_t.
Then, $g(\mathbf{D}) = \mathbb{E}[\min_\alpha \ell(X, \mathbf{D}, \alpha)] = \mathbb{E}[\ell(X, D)]$
An empirical approximation of $g(D)$ can be defined as:
$$g_t(\mathbf{D}) = \sum_{n \le t} \min_\alpha \|x_t - \mathbf{D}\alpha\|^2 + \lambda\|\alpha\|_1, \mathbf{D} \in \Omega$$

Minimizing $g_t(D)$ using an online algorithm is infeasible without keeping all the history data.

- We choose a surrogate function
$$\hat{g}_t(\mathbf{D}) = \sum_{n \le t} \|x_t - \mathbf{D}\alpha_n\|^2 + \lambda\|\alpha_n\|_1, \mathbf{D} \in \Omega$$
- Even optimizing the surrogate function $\hat{g}_t(\mathbf{D})$ for $\mathbf{D} \in \Omega$ is infeasible for an online algorithm, since Ω is unknown
- Instead we compute $\min_{\mathbf{D} \in \mathrm{cvx}(\hat{X}_t)} \hat{g}_t(\mathbf{D})$ in each step, while updating the region $\mathrm{cvx}(\hat{X}_t)$ in each iteration
- **New objective**: $\min_{\mathbf{D} \in \mathrm{cvx}(\hat{X}_t)} \hat{g}_t(\mathbf{D})$

We prove that the results are asymptotically the same as optimizing for $g(D), \mathbf{D} \in \Omega$

Algorithm Setting

We consider two versions of the algorithm
- $\mathcal{R}_u : \mathbf{D}_j \in \Omega, \forall j \in [k]$
- $\mathcal{R}_r : \mathbf{D}_j \in \Omega_j, \forall j \in [k]$ where Ω_j corresponds to the cluster j in the data
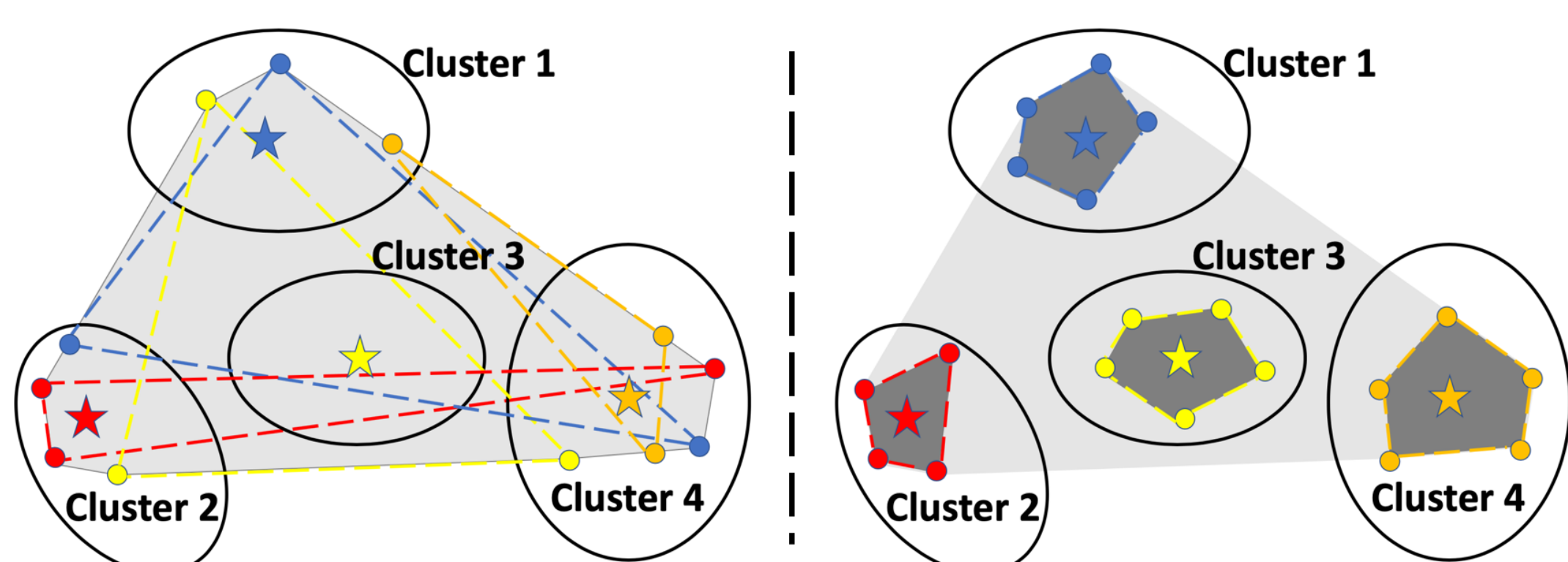

Figure 1:A multi-cluster dataset: Stars represent the learned bases, while circles denote representative samples for the basis of the same color. Left: The representative sets for the individual basis elements are unrestricted. Right: The representative sets are restricted to lie within their corresponding clusters.
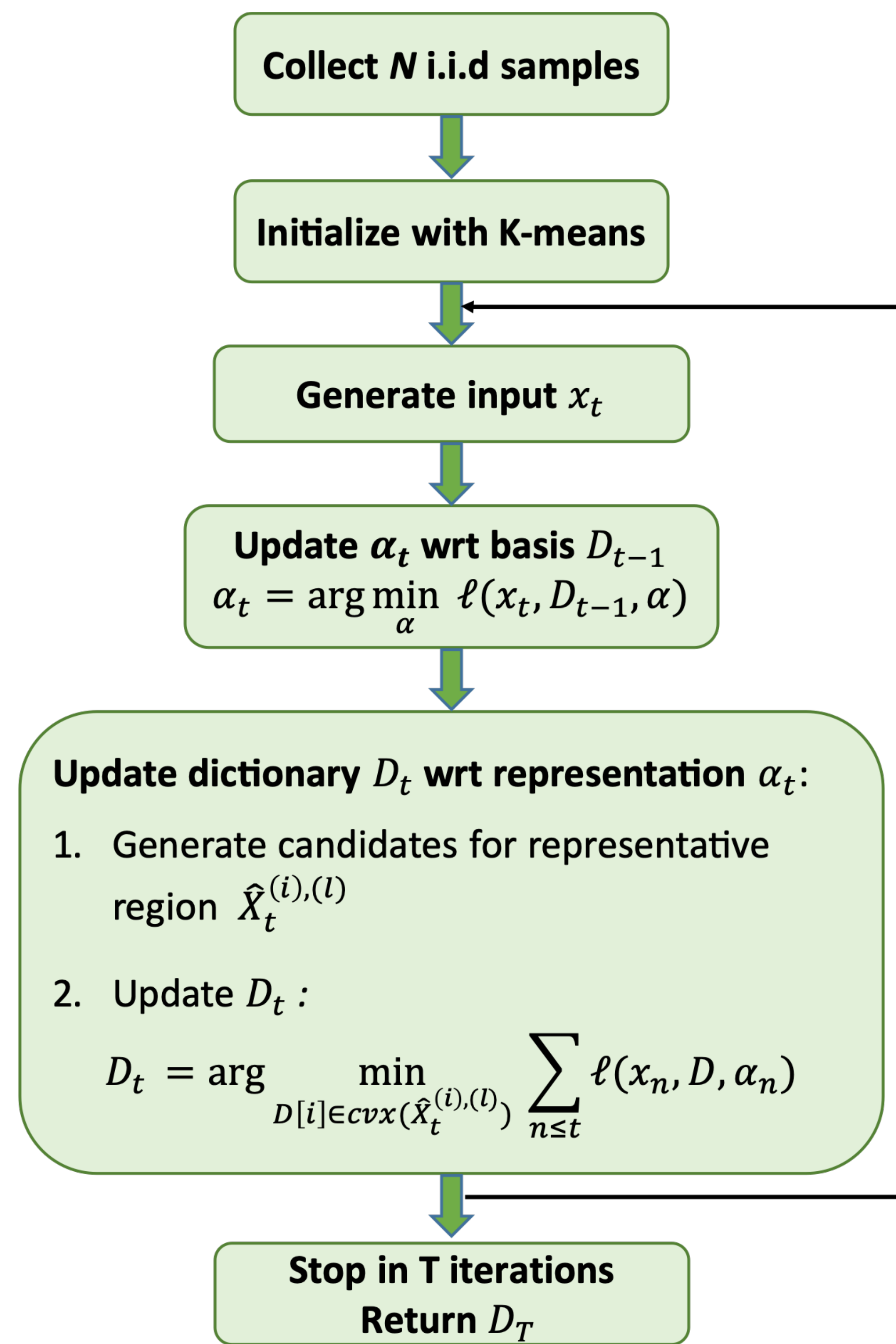
Algorithm

Collect N i.i.d samples

Initialize with K-means

Generate input x_t

Update α_t wrt basis D_{t-1}
$\alpha_t = \arg\min_\alpha \ell(x_t, D_{t-1}, \alpha)$

Update dictionary D_t wrt representation α_t:
1. Generate candidates for representative region $\hat{X}_t^{(i),(l)}$
2. Update D_t :
$$D_t = \arg\min_{D[i] \in cvx(\hat{X}_t^{(i),(l)})} \sum_{n \le t} \ell(x_n, D, \alpha_n)$$

Stop in T iterations
Return D_T

Figure 2:Flowchart of online cvxMF

Main Results

Theorem 1 *Under assumptions (A.1) to (A.3), the sequence $\{D_t\}_t$ converges almost surely to a stationary point of $g(D)$*

Lemma 2 *Let $\Delta_t \triangleq \hat{g}_t(D_t) - \hat{g}_t(D_t^*)$. Then*
$$\mathbb{E}[\Delta_t] \le O\left(\frac{1}{t}\right) + \mathbb{E}[\Delta_{t-1}] - V_m \mathbb{E}[\Delta_{t-1}]^{\frac{m+2}{2}}$$
Where V_m is a coefficient depends on m, κ, the minimum cluster probability p_i, and the running average of $\alpha_t \alpha_t^T$. Here κ is the same constant used in assumption (A.1).

Lemma 3 $\sum_t \frac{\|D_t - D_t^*\|}{t+1}$ *converges almost surely.*

Lemma 4 *The following claims hold true:*
P.1) $\hat{g}_t(D_t)$ and $\hat{g}_t(D_t^*)$ converge almost surely
P.2) $\hat{g}_t(D_t) - \hat{g}_t(D_t^*)$ converges almost surely to 0
P.3) $\hat{g}_t(D_t^*) - g(D_t^*)$ converges almost surely to 0
P.4) $g(D_t^*)$ converges almost surely

Algorithm (cont.)

Figure 3 also shows one step in the online cvxMF algorithms for a three clusters toy dataset.
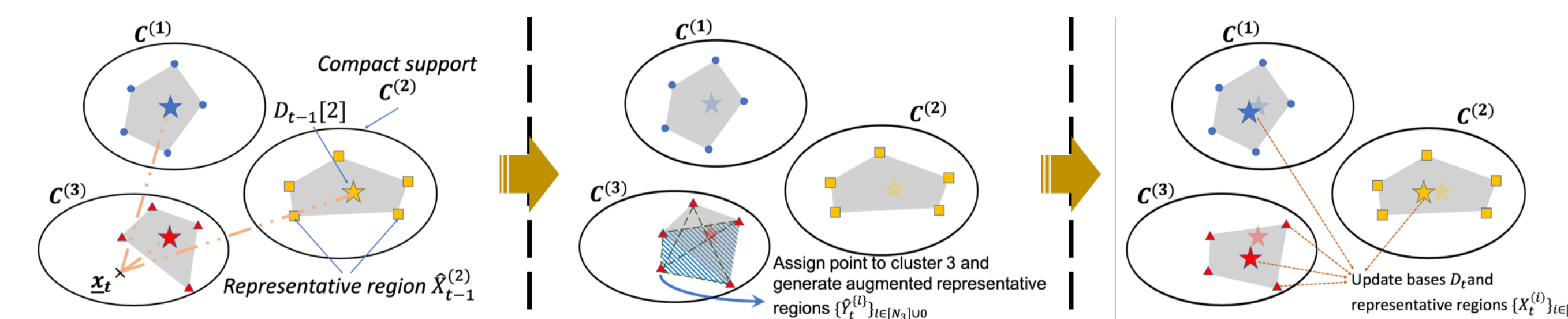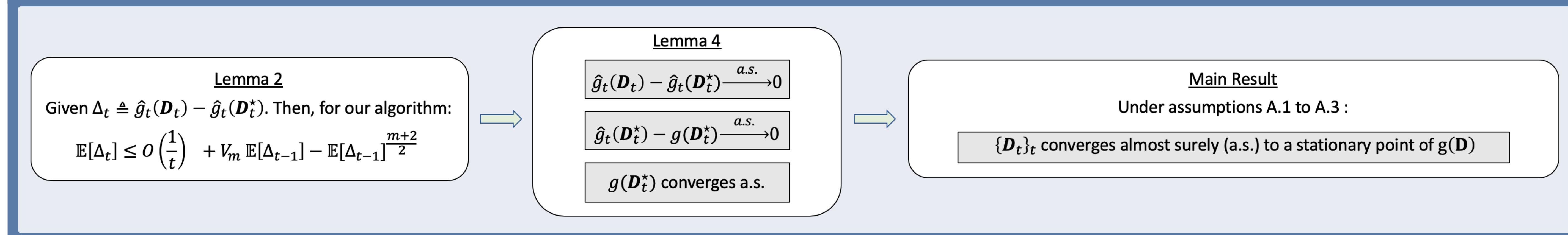

Figure 3:One step of the online cvxMF algorithm with multiple-representative regions.

Assumptions

(A.1) **The data distribution on a compact support set C has bounded "skewnes".** The bounded skewness assumption for the distribution of $\underline{X}$ reads as
$$P(\|\underline{X} - p\| \le r \mid \underline{X} \in \mathcal{C}) \ge \kappa \operatorname{vol}(B(r,p))/\operatorname{vol}(\mathcal{C}) \quad (1)$$

(A.2) **The quadratic surrogate functions $\hat{g}_t$ are strictly convex**, and have Hessians that are lower-bounded by a positive constant $\kappa_1 > 0$.

(A.3) **The approximation-error function $\ell(x, D)$ is "well-behaved".** We assume that $\ell(x, D)$ is continuously differentiable and its expectation $g(D)$ is continuously differentiable and Lipschitz on the compact set $\mathcal{C}$

Key Proof Ideas

Lemma 2
Given $\Delta_t \triangleq \hat{g}_t(\boldsymbol{D}_t) - \hat{g}_t(\boldsymbol{D}_t^*)$. Then, for our algorithm:
$$\mathbb{E}[\Delta_t] \le O\left(\frac{1}{t}\right) + V_m \mathbb{E}[\Delta_{t-1}] - \mathbb{E}[\Delta_{t-1}]^{\frac{m+2}{2}}$$

Lemma 4
$\hat{g}_t(\boldsymbol{D}_t) - \hat{g}_t(\boldsymbol{D}_t^*) \xrightarrow{a.s.} 0$
$\hat{g}_t(\boldsymbol{D}_t^*) - g(\boldsymbol{D}_t^*) \xrightarrow{a.s.} 0$
$g(\boldsymbol{D}_t^*)$ converges a.s.

Main Result
Under assumptions A.1 to A.3 :
$\{\boldsymbol{D}_t\}_t$ converges almost surely (a.s.) to a stationary point of $g(\boldsymbol{D})$

Experimental Validation

We compare the approximation error and running time of our proposed online cvxMF algorithm with non-negative MF (NMF), cvxMF [1] and online MF [2]

Synthetic data: The synthetic datasets were generated by sampling from a 3σ-truncated Gaussian mixture model with 5 components, and with samples-sizes in $[10^3, 10^6]$.
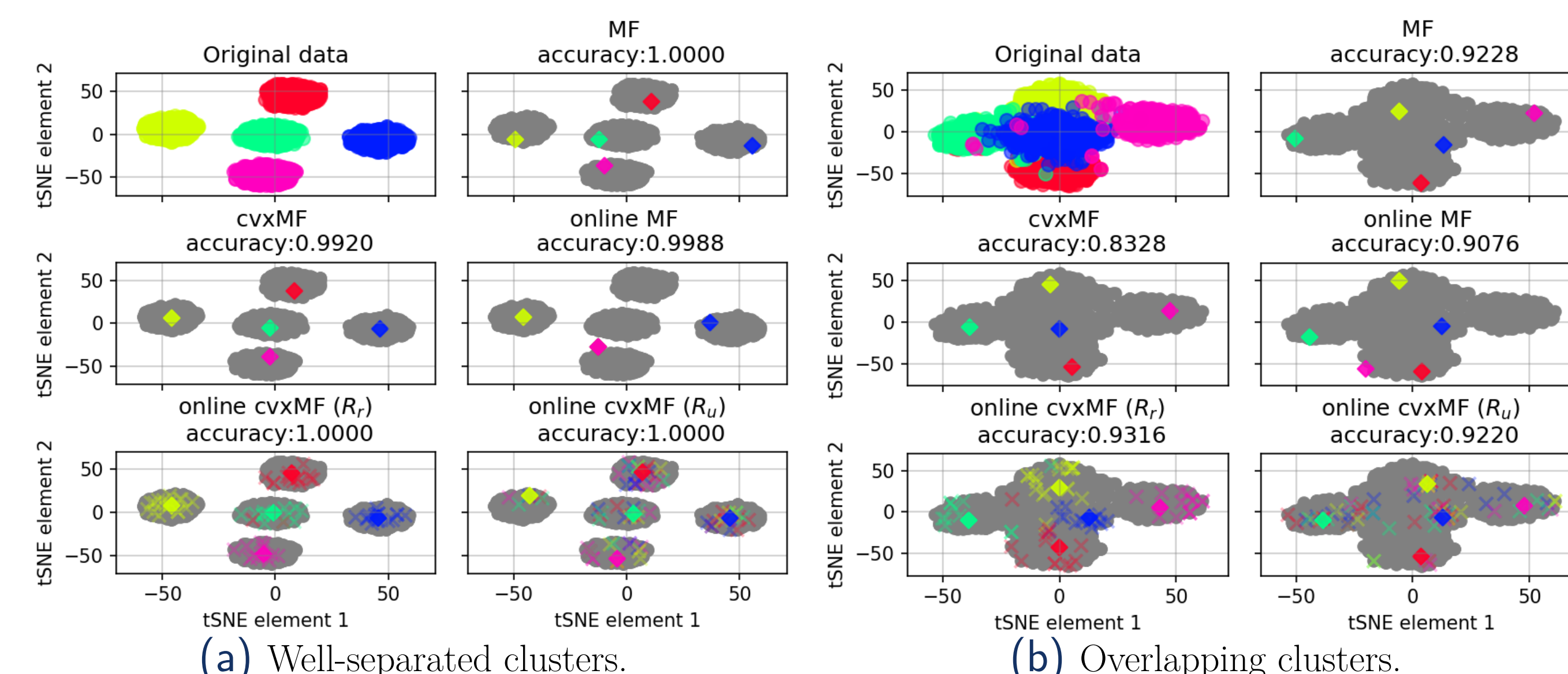

Figure 4:Results for Gaussian mixtures with color-coded clusters. Color-coded circles represent samples, diamonds represent basis vectors learned by the different algorithms, while crosses describe samples in the representative regions. The "interpretability property" can be easily observed visually.

Experimental Validation (cont.)

we present running times and times to convergence (or, if convergence is slow, the maximum number of iterations) in figure 5.
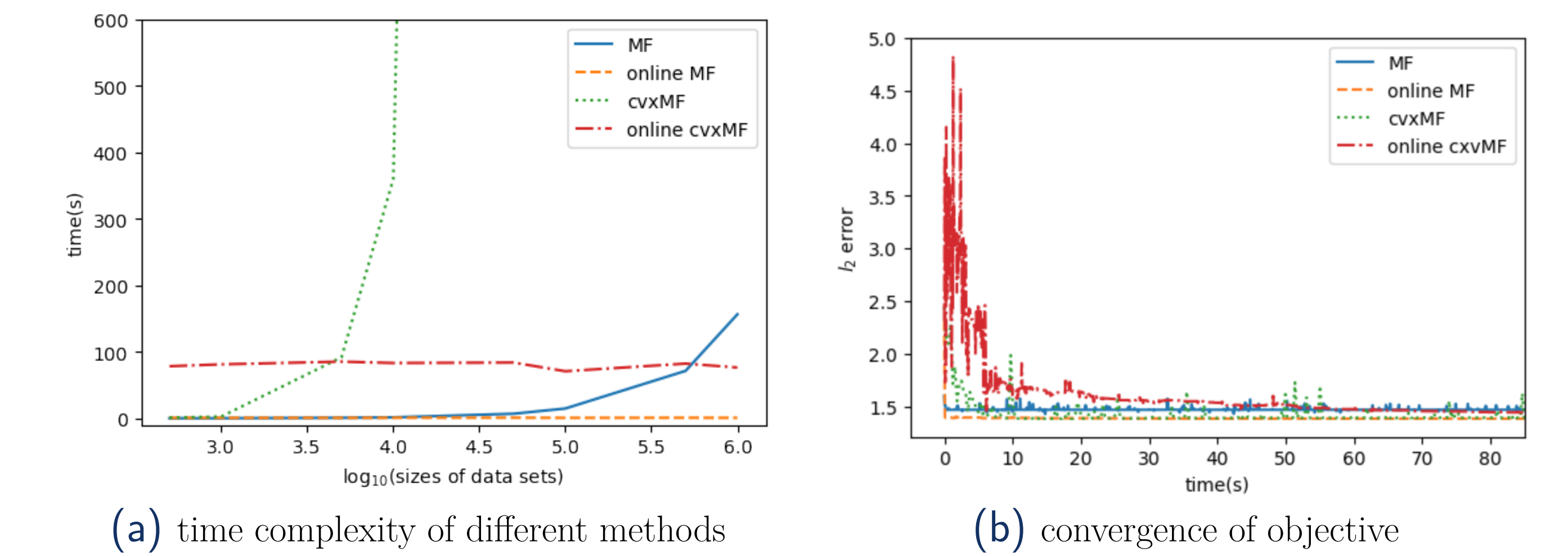

(a) time complexity of different methods
(b) convergence of objective

Figure 5:(a): Running times (s) vs. the log of the dataset sizes; (b) Running times (s) vs. the l_2 error.

The MNIST Dataset:

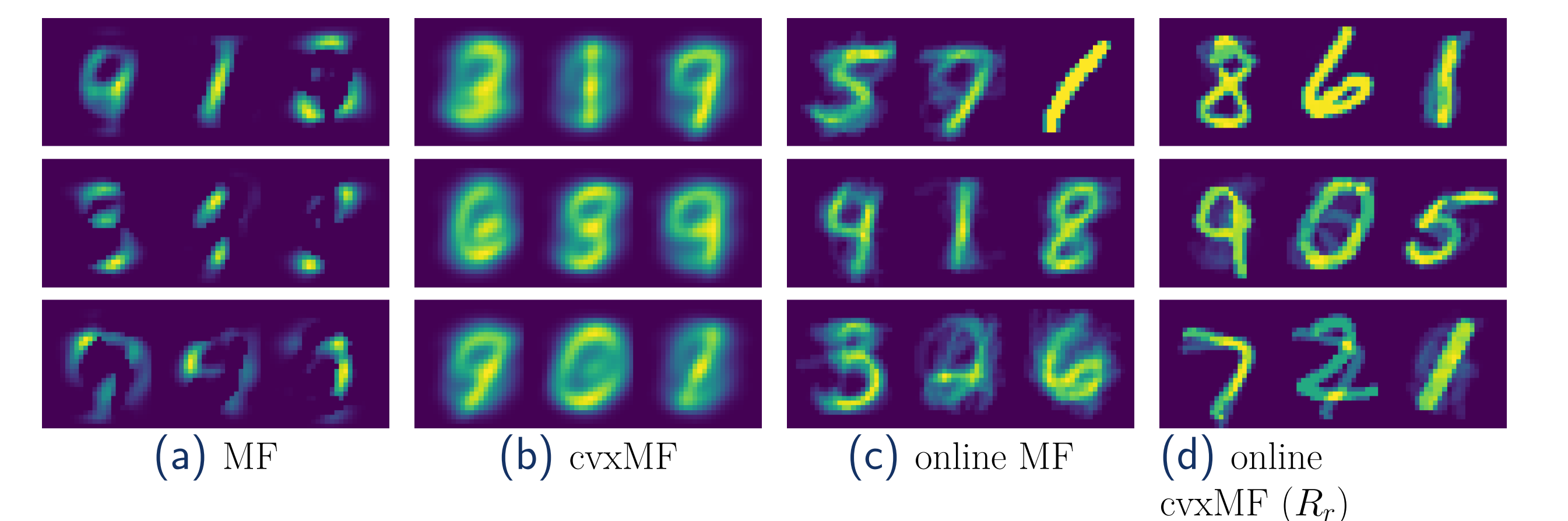(a) MF (b) cvxMF (c) online MF (d) online cvxMF (R_r)

Figure 6:MNIST results (as the eigenimage set is overcomplete, clustering accuracy is omitted).
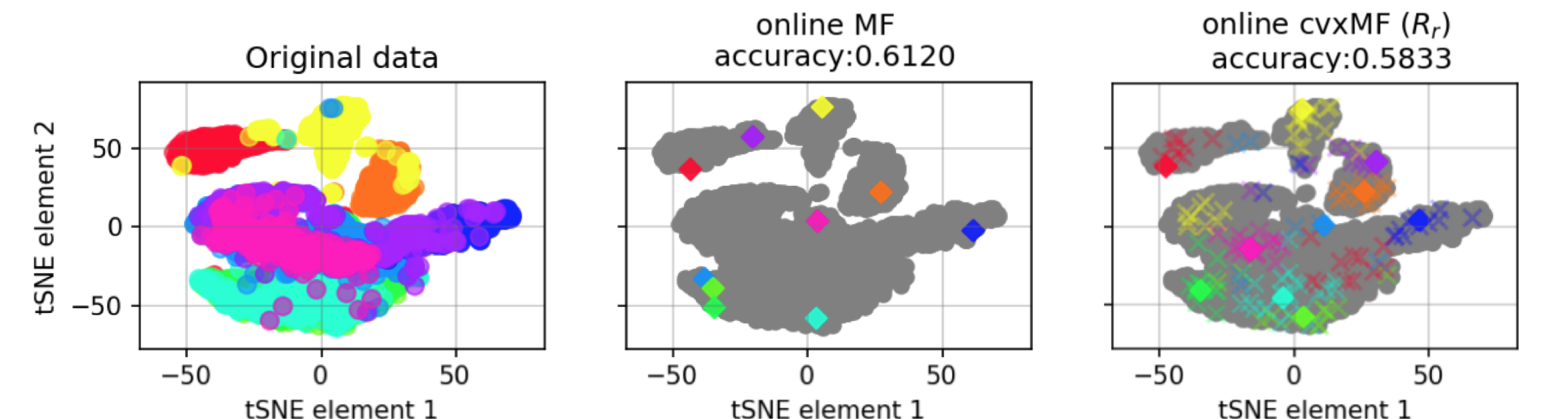
Single-Cell (sc) RNA Data:

Figure 7:Results for the online methods executed on a blood-cell scRNA dataset.

References

[1] Chris HQ Ding, Tao Li, and Michael I Jordan.
Convex and semi-nonnegative matrix factorizations.
IEEE transactions on pattern analysis and machine intelligence, 32(1):45–55, 2010.

[2] Julien Mairal, Francis Bach, Jean Ponce, and Guillermo Sapiro.
Online learning for matrix factorization and sparse coding.
Journal of Machine Learning Research, 11(Jan):19–60, 2010.

Acknowledgements

Contact Information

GitHub: https://github.com/jianhao2016/online_cvxMF
Email: jianhao2@illinois.edu