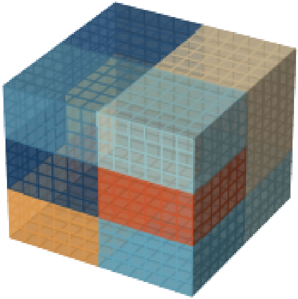
Heteroskedastic Tensor Clustering

Yuxin Chen, Wharton Statistics & Data Science

Yuchen Zhou

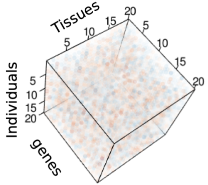
UIUC Statistics

Tensors: high-order arrays



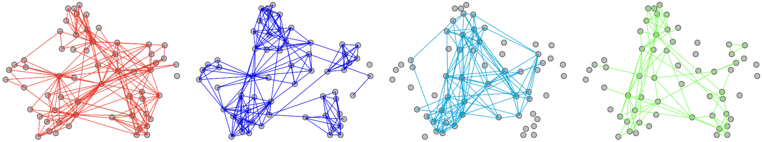
- vectors and matrices are order-1 and order-2 tensors
- $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{n_1 \times \cdots \times n_d}$: order-$d$ tensor

Tensor clustering



Multi-tissue gene expression

fig. credit: Wang et al. '19

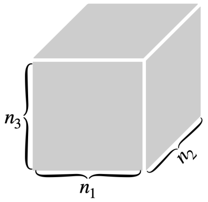


(a) Work. (b) Lunch. (c) Facebook. (d) Friend.

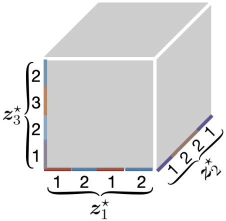
Multilayer network analysis

fig. credit: Kim and Lee '15

Tensor block model

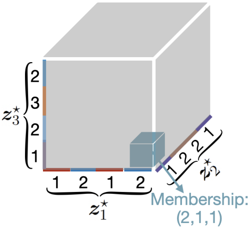


Tensor block model



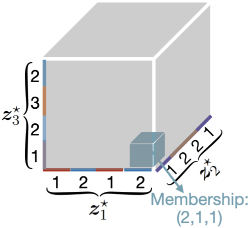
- $z_i^\star \in [k_i]^{n_i}$: cluster assignment vector along the i-th mode
  - $z_{i,j}^\star = \ell$ if the jth index falls within cluster ℓ

Tensor block model



- $z_i^\star \in [k_i]^{n_i}$: cluster assignment vector along the i-th mode
  - $z_{i,j}^\star = \ell$ if the jth index falls within cluster ℓ
  - The membership of (i_1, i_2, i_3) is $(z_{1,i_1}^\star, z_{2,i_2}^\star, z_{3,i_3}^\star)$

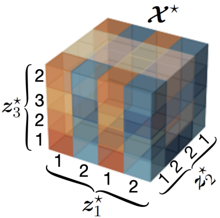
Tensor block model



- $z_i^\star \in [k_i]^{n_i}$: cluster assignment vector along the i-th mode

- $\mathcal{S}^\star \in \mathbb{R}^{k_1 \times k_2 \times k_3}$: block/clustering mean

Tensor block model



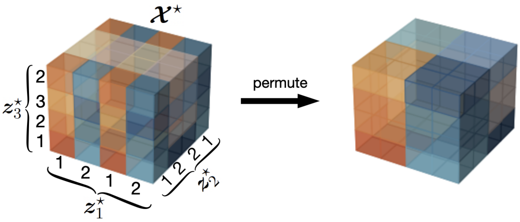
- $\boldsymbol{z}_i^\star \in [k_i]^{n_i}$: cluster assignment vector along the i-th mode

- $\boldsymbol{\mathcal{S}}^\star \in \mathbb{R}^{k_1 \times k_2 \times k_3}$: block/clustering mean

- **Truth:** for all $(i_1, i_2, i_3) \in [n_1] \times [n_2] \times [n_3]$,

$$X^\star_{i_1,i_2,i_3} = S^\star_{z^\star_{1,i_1}, z^\star_{2,i_2}, z^\star_{3,i_3}}$$

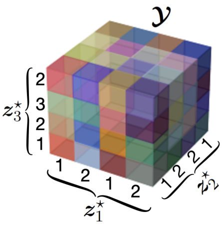
Tensor block model



- $z_i^\star \in [k_i]^{n_i}$: cluster assignment vector along the i-th mode
- $\mathcal{S}^\star \in \mathbb{R}^{k_1 \times k_2 \times k_3}$: block/clustering mean
- **Truth:** for all $(i_1, i_2, i_3) \in [n_1] \times [n_2] \times [n_3]$,

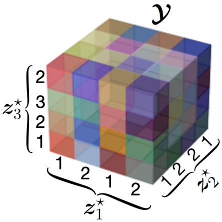
$$X^\star_{i_1,i_2,i_3} = S^\star_{z^\star_{1,i_1}, z^\star_{2,i_2}, z^\star_{3,i_3}}$$

Tensor block model



- **Noisy observations:** $\mathcal{Y} = \mathcal{X}^\star + \underbrace{\mathcal{E}}_{\text{zero-mean ind. noise}} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$

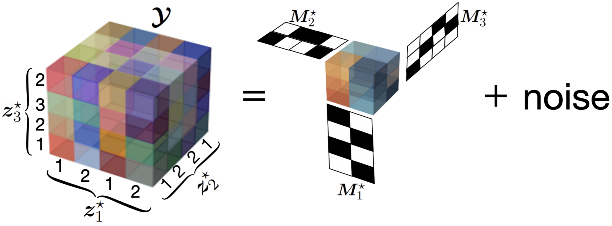
Tensor block model



- **Noisy observations:** $\mathcal{Y} = \mathcal{X}^\star + \underbrace{\mathcal{E}}_{\text{zero-mean ind. noise}} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$

- **Goal**: recover $z_i^\star$, $i \in [3]$ from $\mathcal{Y}$

Tensor block model



Equivalently, we observe

$$\mathcal{Y} = \mathcal{X}^\star + \underbrace{\mathcal{E}}_{\text{zero-mean ind. noise}} \quad \in \mathbb{R}^{n_1 \times n_2 \times n_3},$$

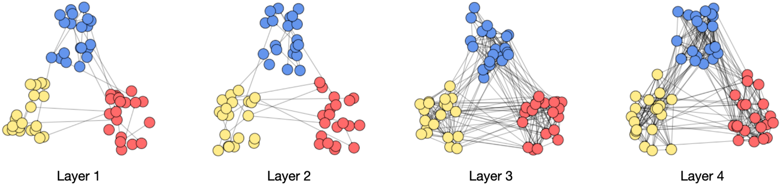
where $\mathcal{X}^\star = \mathcal{S}^\star \times_1 \boldsymbol{M}_1^\star \times_2 \boldsymbol{M}_2^\star \times_3 \boldsymbol{M}_3^\star$, with $\boldsymbol{M}_i^\star \in \{0,1\}^{n_i \times k_i}$ s.t.

$$(\boldsymbol{M}_i^\star)_{j,\ell} = \begin{cases} 1, & \text{if } z_{i,j}^\star = \ell \\ 0, & \text{else} \end{cases}$$

An important case: stochastic tensor block model

Stochastic tensor block model (STBM)

- generalization of bipartite stochastic block model
- each entry of $\boldsymbol{\mathcal{S}}^\star$ is connection probability
- $Y_{i,j,\ell} = \begin{cases} 1, & \text{with prob. } S^\star_{z^\star_{1,i}, z^\star_{2,j}, z^\star_{3,\ell}}, \\ 0, & \text{o.w.} \end{cases}$



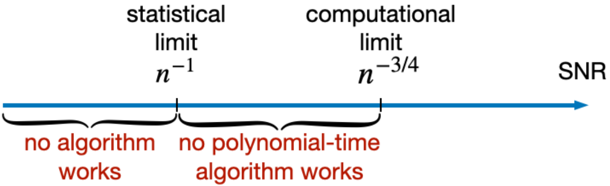
A natural approach

Least-square estimator (Wang and Zeng '19):

$$\left(\widehat{\boldsymbol{\mathcal{S}}}, \widehat{\boldsymbol{z}}_1, \widehat{\boldsymbol{z}}_2, \widehat{\boldsymbol{z}}_3\right) := \underset{\substack{\boldsymbol{\mathcal{S}} \in [k_1] \times [k_2] \times [k_3] \\ \boldsymbol{z}_i \in [k_i]^{n_i}}}{\arg\min} \sum_{i,j,\ell} \left(Y_{i,j,\ell} - S_{z_{1,i}, z_{2,j}, z_{3,\ell}}\right)^2 \tag{1}$$

- statistically accurate, computationally intractable!

Statistical-computational gap



SNR $:= \Delta_{\mathsf{min}}/\omega_{\mathsf{max}}$, where

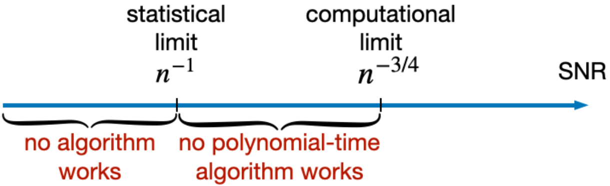
- signal strength:

$$\Delta_1 := \overbrace{\min_{1\le j_1\neq j_2\le k_1} \left\| S^\star_{j_1,:,:} - S^\star_{j_2,:,:}\right\|_2}^{\text{minimum seperation distance along mode}-1}$$

$$\Delta_{\mathsf{min}} := \min_{1\le i\le 3} \Delta_i$$

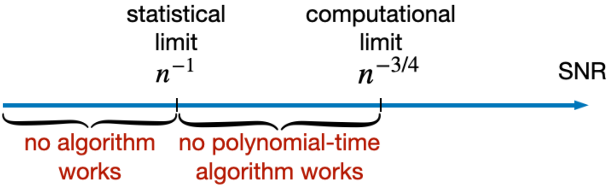
- noise level: ω_{max}

Statistical-computational gap



- Han et al. '22: HSC + HLloyd
    - spectral clustering + iterative refinements

Statistical-computational gap

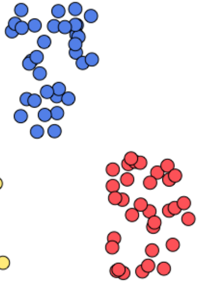


- Han et al. '22: HSC + HLloyd
  - spectral clustering + iterative refinements

- Under i.i.d. sub-Gaussian noise, HSC + HLloyd achieves exact clustering if SNR exceeds $n^{-3/4}$ (up to log factors)

A toy example: STBM

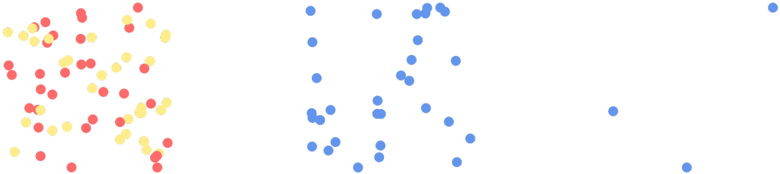
- 100-layer networks, with 100 nodes
- 3 clusters for layers, 3 clusters for nodes
- Nodes within the yellow/red/blue community are sparsely connected.



Layer 1

A toy example: STBM

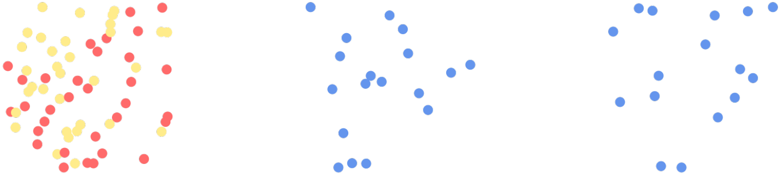
- 100-layer networks, with 100 nodes
- 3 clusters for layers, 3 clusters for nodes
- Nodes within the yellow/red/blue community are sparsely connected.



HSC

A toy example: STBM

- 100-layer networks, with 100 nodes
- 3 clusters for layers, 3 clusters for nodes
- Nodes within the yellow/red/blue community are sparsely connected.



HSC + HLloyd

A common senario: heteroskedastic noise

- $\underbrace{\text{noise variances } \{\mathbb{E}[E_{i,j,\ell}^2]\}}_{\text{unknown } \textit{a priori}}$ are location-varying
- discrete-valued observations: multi-tissue gene expression data, multilayer network data

A common senario: heteroskedastic noise

- $\underbrace{\text{noise variances } \{\mathbb{E}[E_{i,j,\ell}^2]\}}_{\text{unknown } a\ priori}$ are location-varying
- discrete-valued observations: multi-tissue gene expression data, multilayer network data
- fail dramatically in the face of **heteroskedastic** noise!

How to deal with heteroskedastic noise under essential SNR conditions?

Spectral clustering

- Step 1: estimate "important" subspace of $\boldsymbol{X}_i^\star = \mathcal{M}_i(\boldsymbol{\mathcal{X}}^\star) \Rightarrow \boldsymbol{U}_i$.

- Step 2: apply approximate k-means on the rows of

$$\widehat{\boldsymbol{B}}_1 = \mathcal{M}_1\Big(\boldsymbol{\mathcal{Y}} \underbrace{\times_1 \boldsymbol{U}_1 \boldsymbol{U}_1^\top}_{\text{denoising}} \underbrace{\times_2 \boldsymbol{U}_2 \times_3 \boldsymbol{U}_3}_{\substack{\text{dimension reduction} \\ \text{\& denoising}}}\Big) \qquad (\text{resp. } \widehat{\boldsymbol{B}}_2, \widehat{\boldsymbol{B}}_3)$$

  - can be efficiently done by using k-means++!

Spectral clustering

- Step 1: estimate "important" subspace of $\boldsymbol{X}_i^\star = \mathcal{M}_i(\boldsymbol{\mathcal{X}}^\star) \Rightarrow \boldsymbol{U}_i$.

Key challenges:

- ◦ unbalanced dimensionality
- ◦ heteroskedastic noise

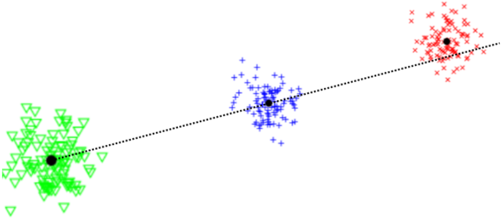
SVD can be highly sub-optimal $\Longrightarrow$ HSC fails drastically!

Spectral clustering

- Step 1: estimate "important" subspace of $\boldsymbol{X}_i^\star = \mathcal{M}_i(\boldsymbol{\mathcal{X}}^\star) \Rightarrow \boldsymbol{U}_i$.

Key challenges:

- avoid unnecessary assumptions on condition number of $\boldsymbol{\mathcal{S}}^\star$



condi. number $\kappa := \frac{\max_i \sigma_1(\mathcal{M}_i(\boldsymbol{\mathcal{S}}^\star))}{\min_i \sigma_{k_i}(\mathcal{M}_i(\boldsymbol{\mathcal{S}}^\star))}$

Road map

Subspace estimation problem

Key challenges : heteroskedastic noise
unbalanced dimensionality
not well-conditioning

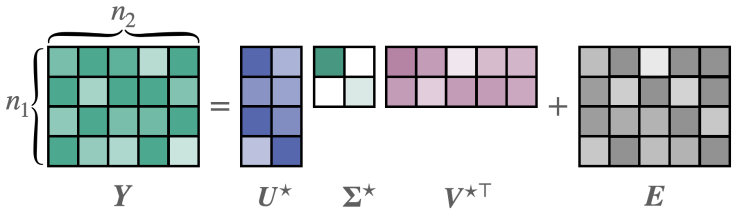
Propose Deflated-HeteroPCA! 

Spectral tensor clustering

Still need assumptions on least singular value!

Thresholded Deflated-HeteroPCA + k-means!

A detour: a subspace estimation / model

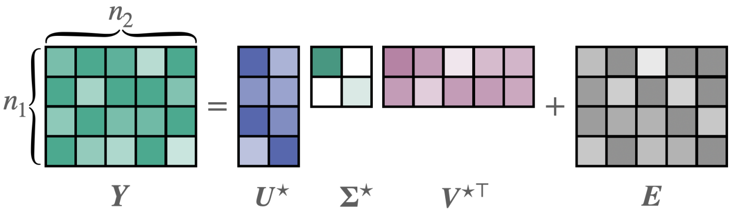


- **Ground truth:** rank-r matrix $\boldsymbol{X}^\star$ with SVD $(r \ll \min\{n_1, n_2\})$

$$\boldsymbol{X}^\star = \boldsymbol{U}^\star \boldsymbol{\Sigma}^\star \boldsymbol{V}^{\star\top} = \sum_{i=1}^{r} \sigma_i^\star \boldsymbol{u}_i^\star \boldsymbol{v}_i^{\star\top} \in \mathbb{R}^{n_1 \times n_2}$$

where $\boldsymbol{U}^\star \in \mathbb{R}^{n_1 \times r}$, $\boldsymbol{\Sigma}^\star = \mathrm{diag}\{\sigma_1^\star, \cdots, \sigma_r^\star\}$, $\boldsymbol{V}^\star \in \mathbb{R}^{n_2 \times r}$

A detour: a subspace estimation / model

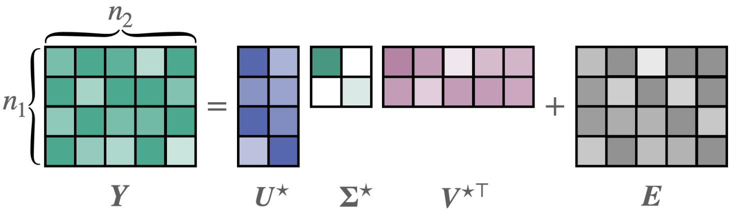


- **Ground truth:** rank-r matrix $\boldsymbol{X}^\star$ with SVD ($r \ll \min\{n_1, n_2\}$)

$$\boldsymbol{X}^\star = \boldsymbol{U}^\star \boldsymbol{\Sigma}^\star \boldsymbol{V}^{\star\top} = \sum_{i=1}^{r} \sigma_i^\star \boldsymbol{u}_i^\star \boldsymbol{v}_i^{\star\top} \in \mathbb{R}^{n_1 \times n_2}$$

- **Noisy observations:** $\boldsymbol{Y} = \boldsymbol{X}^\star + \underbrace{\boldsymbol{E}}_{\text{zero-mean ind. noise}}$

A detour: a subspace estimation / model



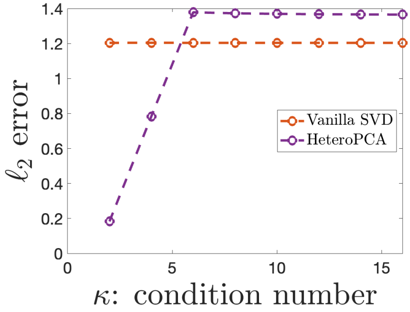
- **Ground truth:** rank-r matrix $X^\star$ with SVD ($r \ll \min\{n_1, n_2\}$)

$$X^\star = U^\star \Sigma^\star V^{\star\top} = \sum_{i=1}^{r} \sigma_i^\star u_i^\star v_i^{\star\top} \in \mathbb{R}^{n_1 \times n_2}$$

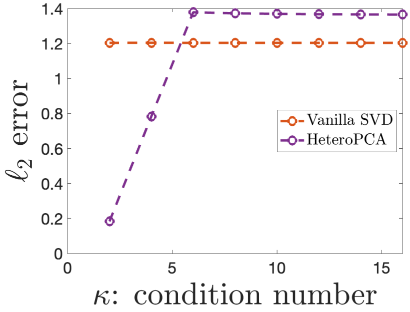
- **Noisy observations:** $Y = X^\star + \underbrace{E}_{\text{zero-mean ind. noise}}$
- **Goal**: estimate column subspace $U^\star \in \mathbb{R}^{n_1 \times r}$ based on Y

A curious phenomenon: curse of ill-conditioning

Somewhat surprising numerical example: $r = 2, n_1 = 200,\ n_2 = 40,000$



Somewhat surprising numerical example: $r = 2, n_1 = 200,\ n_2 = 40,000$



Previous methods degrade as $\underbrace{\textit{condition number of } \boldsymbol{X}^\star \textit{ increases!}}_{\text{but this actually makes problem info-theoretically easier} \ldots}$

Can we break the curse of ill-conditioning while accommodating widest SNR range?

Proposed algorithm: deflated-HeteroPCA

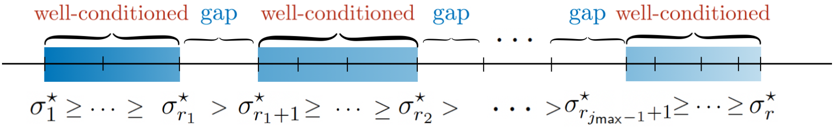
— Zhang et al. '18, Yan et al. '21

revisit HeteroPCA theory: works well if

- $\boldsymbol{X}^\star$ is well-conditioned
- least singular value $\sigma_r^\star$ (or spectral gap) is not buried by noise

$$\sigma_1^\star \geq \sigma_2^\star \geq \quad \cdots \quad \geq \quad \sigma_r^\star$$

Proposed algorithm: deflated-HeteroPCA



solution:

- divide eigenvalues into well-conditioned & well-separated subblocks (data-driven)
- estimate subblocks sequentially

More details: Y. Zhou, Y. Chen, "Deflated HeteroPCA: Overcoming the Curse of Ill-conditioning in Heteroskedastic PCA," arxiv:2303.06198, 2023

Performance of Deflated-HeteroPCA

Zhou, Chen '23a (informal):

Under essential assumptions on $\sigma_r^\star$ and regularity assumptions, Deflated-HeteroPCA

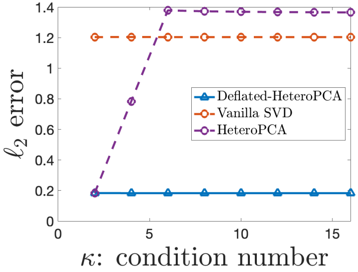
- can handle heteroskedastic noise
- enjoys near-optimal and condition-number-free guarantees

Performance of Deflated-HeteroPCA

Zhou, Chen '23a (informal):

Under essential assumptions on $\sigma_r^\star$ and regularity assumptions, Deflated-HeteroPCA

- can handle heteroskedastic noise
- enjoys near-optimal and condition-number-free guarantees



Back to tensor clustering...

Subspace estimation problem

Key challenges : heteroskedastic noise

unbalanced dimensionality

not well-conditioning

Propose Deflated-HeteroPCA!

Spectral tensor clustering

Still need assumptions on least singular value!

Thresholded Deflated-HeteroPCA + k-means!

Back to tensor clustering...

Subspace estimation problem

Key challenges :heteroskedastic noise

unbalanced dimensionality

not well-conditioning

Propose Deflated-HeteroPCA! 😎

↓

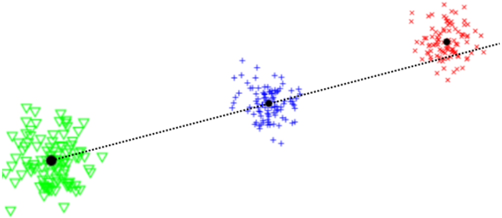
Spectral tensor clustering

Still need assumptions on least singular value!

Thresholded Deflated-HeteroPCA + k-means!

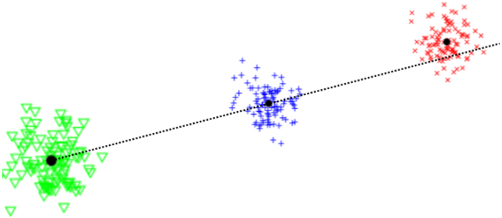
Back to tensor clustering...

- Least singular value assumption is unnecessary for clustering!



Back to tensor clustering...

- Least singular value assumption is unnecessary for clustering!



- Only large singular values matter! Add a thresholding procedure!

Thresholded Deflated-HeteroPCA$(\boldsymbol{Y}, r, \tau)$

- **Initialize:** $\boldsymbol{G}_0 = \mathcal{P}_{\text{off-diag}}(\boldsymbol{Y}\boldsymbol{Y}^\top), j = 0, r_0 = 0$
- **Sequential updates:** while $r_j < r$ and $\sigma_{r_j+1}(\boldsymbol{G}_j) > \underbrace{\tau}_{\text{data-driven threshold}}$

$$j = j + 1$$

select r_j in a data-driven manner

$$(\boldsymbol{G}_j, \boldsymbol{U}_j) = \text{HeteroPCA}(\boldsymbol{G}_{j-1}, r_j)$$

- **Output:** $\boldsymbol{U} \coloneqq \boldsymbol{U}_j \quad \longrightarrow \quad$ estimate of $\boldsymbol{U}^\star$

Proposed algorithm: High-order HeteroClustering

- Step 1: estimate "important" subspaces of $\boldsymbol{X}_i^\star = \mathcal{M}_i(\boldsymbol{\mathcal{X}}^\star)$

  - Obtain $\boldsymbol{U}_i$ via Thresholded Deflated-HeteroPCA$(\boldsymbol{X}_i^\star, k_i, \tau)$

- Step 2: apply approximate k-means on the rows of

$$\widehat{\boldsymbol{B}}_1 = \mathcal{M}_1(\boldsymbol{\mathcal{Y}} \underbrace{\times_1 \boldsymbol{U}_1 \boldsymbol{U}_1^\top}_{\text{denoising}} \underbrace{\times_2 \boldsymbol{U}_2 \times_3 \boldsymbol{U}_3}_{\substack{\text{dimension reduction} \\ \& \text{ denoising}}}) \qquad \text{(resp. } \widehat{\boldsymbol{B}}_2, \widehat{\boldsymbol{B}}_3)$$

- Optional update: HLloyd (Han et al. '22)

Assumptions (ignoring log factors)

- dimension $n_1 \asymp n_2 \asymp n_3 \asymp n$
- **heteroskedasticity**: $\boldsymbol{\mathcal{E}}$ has indep. zero-mean entries obeying
  - $\mathsf{Var}[E_{i,j,\ell}] \leq \omega_{\mathsf{max}}^2$
  - $|E_{i,j,\ell}| \lesssim \omega_{\mathsf{max}} n^{3/4}$ with high prob.
  - examples: sub-Gaussian, centered Poisson, STBM

Assumptions (ignoring log factors)

- dimension $n_1 \asymp n_2 \asymp n_3 \asymp n$
- **heteroskedasticity**: $\mathcal{E}$ has indep. zero-mean entries obeying
  - $\mathsf{Var}[E_{i,j,\ell}] \le \omega_{\mathsf{max}}^2$
  - $|E_{i,j,\ell}| \lesssim \omega_{\mathsf{max}} n^{3/4}$ with high prob.
  - examples: sub-Gaussian, centered Poisson, STBM
- **signal-to-noise ratio (SNR)**:

$$\frac{\Delta_{\mathsf{min}}}{\omega_{\mathsf{max}}} \gtrsim n^{-3/4}$$

  - match the computational limit (Han et al. '22)

Assumptions (ignoring log factors)

- dimension $n_1 \asymp n_2 \asymp n_3 \asymp n$
- **heteroskedasticity**: $\mathcal{E}$ has indep. zero-mean entries obeying
  - $\mathsf{Var}[E_{i,j,\ell}] \leq \omega_{\mathsf{max}}^2$
  - $|E_{i,j,\ell}| \lesssim \omega_{\mathsf{max}} n^{3/4}$ with high prob.
  - examples: sub-Gaussian, centered Poisson, STBM
- **signal-to-noise ratio (SNR)**:

$$\frac{\Delta_{\mathsf{min}}}{\omega_{\mathsf{max}}} \gtrsim n^{-3/4}$$

  - match the computational limit (Han et al. '22)
- balanced cluster sizes: $\left|\left\{j \in [n_i] : (\boldsymbol{z}_i^\star)_j = \ell\right\}\right| \asymp n_i/k_i$
- number of clusters $k_i = O(1)$

Theoretical guarantees

Theorem 1 (Zhou, Chen '23)

Assume that either (1) or (2) is satisfied:

(1). *noise is not too spiky (e.g.,* $\mathrm{Var}\left[E_{i,j,\ell}\right] \asymp \omega_{\mathsf{max}}^2$*);*

(2). *The observation model is the stochastic tensor block model.*

W.h.p., HHC and HHC + HLloyd achieve exact clustering.

Theoretical guarantees

Theorem 1 (Zhou, Chen '23)

Assume that either (1) or (2) is satisfied:

(1). *noise is not too spiky (e.g.,* $\mathsf{Var}\left[E_{i,j,\ell}\right] \asymp \omega_{\mathsf{max}}^2$*);*

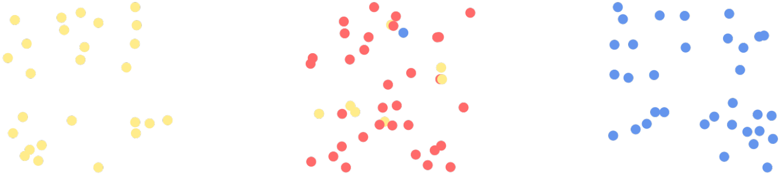
(2). *The observation model is the stochastic tensor block model.*

W.h.p., HHC and HHC + HLloyd achieve exact clustering.

- (almost) necessary SNR condition among poly-time algorithms
- handle heteroskedastic noise & no superfluous assumptions on $\boldsymbol{\mathcal{S}}^\star$

Numerical experiments

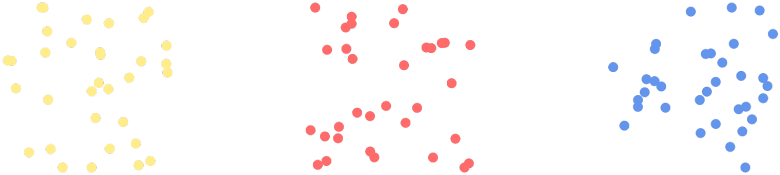
The toy example



HHC

Numerical experiments

The toy example



HHC + HLloyd

Real data example: the flight route network

- OpenFlights Airports Database: global fight information*
- top 50 airports based on the number of flights $\to$ 39 airlines†
- data $\boldsymbol{\mathcal{Y}} \in \{0,1\}^{39\times 50\times 50}$:

$$Y_{i,j,\ell} = \begin{cases} 1, & \text{if airline } i \text{ operates a flight route b/w airports } j, \ell, \\ 0, & \text{otherwise.} \end{cases}$$

*original data: https://openflights.org/data

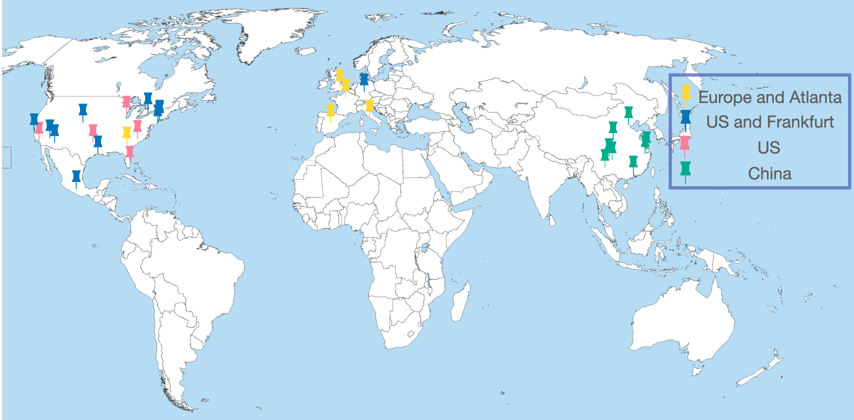
†processed data: https://github.com/Rungang/HLloyd/blob/master/experiment/flight_route.RData

Real data example: the flight route network

Select the clustering sizes based on BIC: $(k_1, k_2, k_3) = (5, 5, 5)$

Real data example: the flight route network

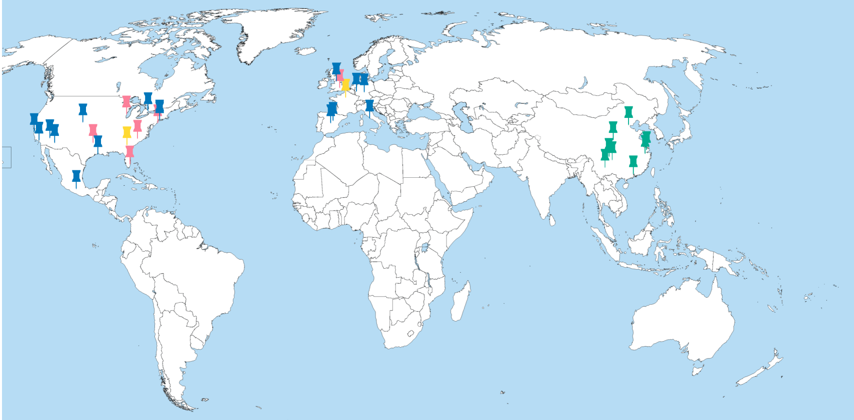
Select the clustering sizes based on BIC: $(k_1, k_2, k_3) = (5, 5, 5)$



Ours

Real data example: the flight route network

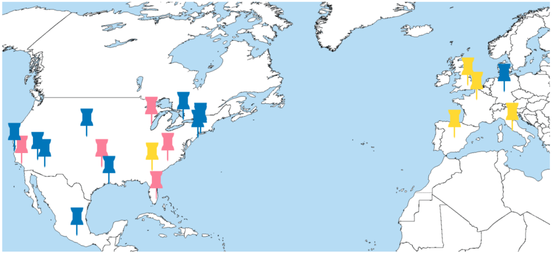
Select the clustering sizes based on BIC: $(k_1, k_2, k_3) = (5, 5, 5)$



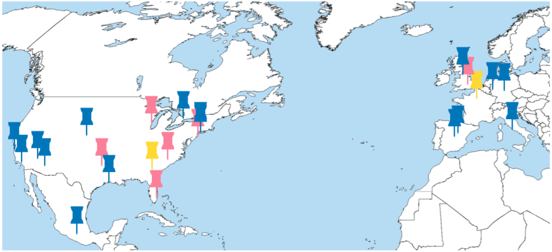
Han et al.'22

Real data example: the flight route network

Ours



Han et al.’22



Summary

Tensor clustering: a novel method that

- can handle heteroskedastic noise
- achieves exact clustering if SNR exceeds computational limit

Summary

Tensor clustering: a novel method that

- can handle heteroskedastic noise
- achieves exact clustering if SNR exceeds computational limit

a detour: new method for subspace estimation

- overcome curse of ill-conditioning & near-optimal guarantees

Summary

Tensor clustering: a novel method that

- can handle heteroskedastic noise
- achieves exact clustering if SNR exceeds computational limit

a detour: new method for subspace estimation

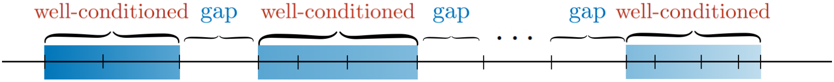
- overcome curse of ill-conditioning & near-optimal guarantees

papers:

Y. Zhou, Y. Chen, "Deflated HeteroPCA: Overcoming the Curse of Ill-conditioning in Heteroskedastic PCA," arxiv:2303.06198, 2023

Y. Zhou, Y. Chen, "Heteroskedastic Tensor Clustering," arxiv:2311.02306, 2023

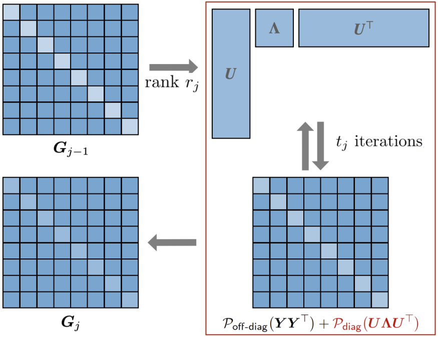
Proposed algorithm: deflated-HeteroPCA



$$\sigma_1^\star \geq \cdots \geq \sigma_{r_1}^\star > \sigma_{r_1+1}^\star \geq \cdots \geq \sigma_{r_2}^\star > \cdots > \sigma_{r_{j_{\max}-1}+1}^\star \geq \cdots \geq \sigma_r^\star$$

- sequentially choose ranks $\underbrace{r_0 = 0 < r_1 < \cdots < r_{j_{\max}} = r}_{\text{data-driven}}$ s.t.
    - $\sigma_{r_{j-1}+1}^\star / \sigma_{r_j}^\star$ is small
    - sufficient gap between $\sigma_{r_j}^\star$ and $\sigma_{r_j+1}^\star$

Proposed algorithm: deflated-HeteroPCA



- sequentially choose ranks $\underbrace{r_0 = 0 < r_1 < \cdots < r_{j_{\max}} = r}_{\text{data-driven}}$ s.t.
  - $\sigma^\star_{r_{j-1}+1}/\sigma^\star_{r_j}$ is small
  - sufficient gap between $\sigma^\star_{r_j}$ and $\sigma^\star_{r_j+1}$
- invoke HeteroPCA$(\underbrace{G_{j-1}}_{\text{input}}, \underbrace{r_j}_{\text{rank}})$ to impute diagonals & obtain G_k

Proposed algorithm: deflated-HeteroPCA

- **Initialize:** $G_0 = \mathcal{P}_{\text{off-diag}}(\boldsymbol{Y}\boldsymbol{Y}^\top), k = 0, r_0 = 0$
- **Sequential updates:** while $r_k < r$

$$
\begin{aligned}
& j = j + 1 \\
& \text{select } r_j \text{ in a data-driven manner} \\
& (\boldsymbol{G}_j, \boldsymbol{U}_j) = \text{HeteroPCA}(\underbrace{\boldsymbol{G}_{j-1}}_{\text{input}}, \underbrace{r_j}_{\text{rank}})
\end{aligned}
$$

- **Output:** $\boldsymbol{U} \coloneqq \boldsymbol{U}_j \quad \longrightarrow \quad$ estimate of $\boldsymbol{U}^\star$

Proposed algorithm: deflated-HeteroPCA

- **Initialize:** $G_0 = \mathcal{P}_{\text{off-diag}}(YY^\top), k = 0, r_0 = 0$
- **Sequential updates:** while $r_k < r$

$$j = j + 1$$

select r_j in a data-driven manner

$$(G_j, U_j) = \text{HeteroPCA}(\underbrace{G_{j-1}}_{\text{input}}, \underbrace{r_j}_{\text{rank}})$$

- **Output:** $U := U_j \quad \longrightarrow \quad$ estimate of $U^\star$

Select $r_j = \begin{cases} \max \mathcal{R}_j, & \text{if } \mathcal{R}_j \neq \emptyset, \\ r, & \text{otherwise.} \end{cases}$ Here,

$$\mathcal{R}_j := \{r' : r_{j-1} < r' \leq r, \overbrace{\sigma_{r_{j-1}+1}(G_{j-1}) / \sigma_{r'}(G_{j-1}) \leq 4}^{\text{well-conditioned}} \; \& \; \underbrace{\sigma_{r'}(G_{j-1}) - \sigma_{r'+1}(G_{j-1}) \geq \sigma_{r'}(G_{j-1}) / r}_{\text{gap}}\}.$$