Contributions to unsupervised learning from massive high-dimensional data streams: structuring, hashing and clustering

PhD Thesis defense

Anne MORVAN

CEA, LIST, LADIS
Université Paris-Dauphine, PSL Research University, CNRS, UMR 7243, LAMSADE

November 12, 2018

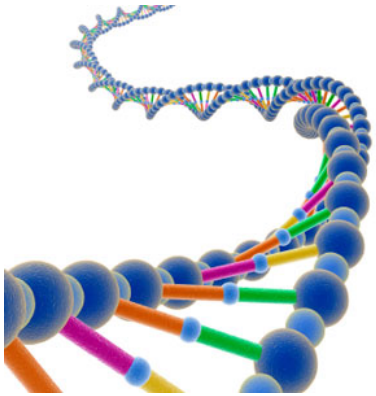
Plan

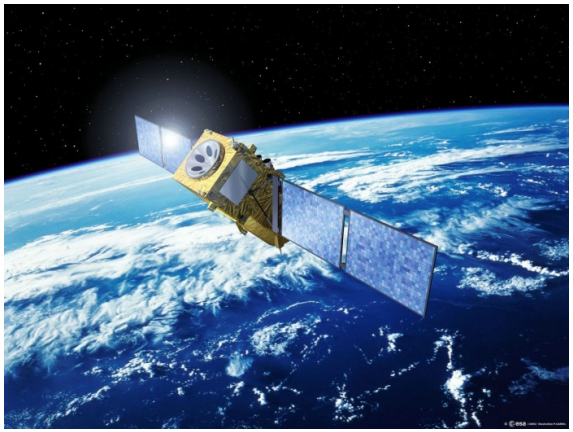
Plan

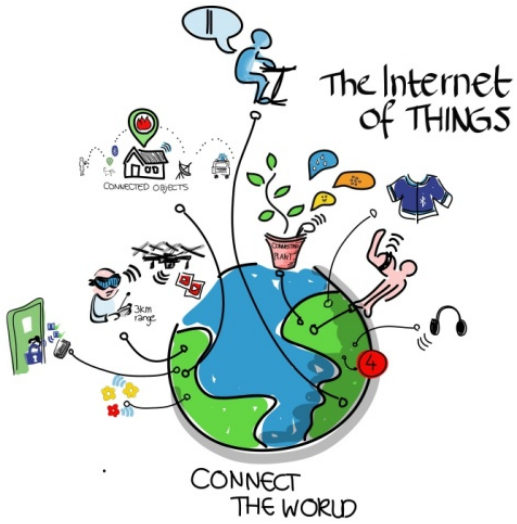
Context & motivation

Big Data era

- Huge quantity
- High dimensionality
- Distribution or real time extraction
- Unlabelled data



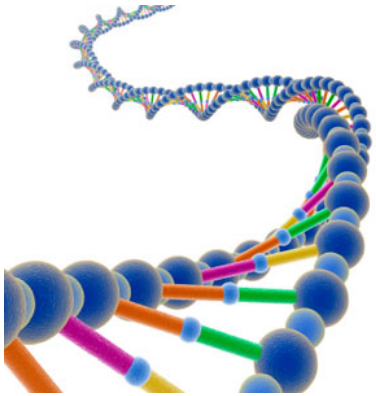


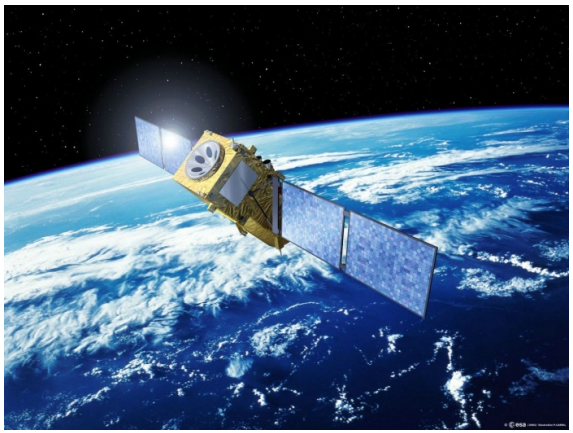


Context & motivation

Consecutive issues

- Huge quantity
- High dimensionality
- Distribution or real time extraction
- Unlabelled data

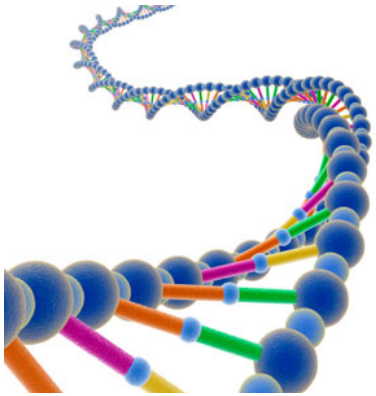


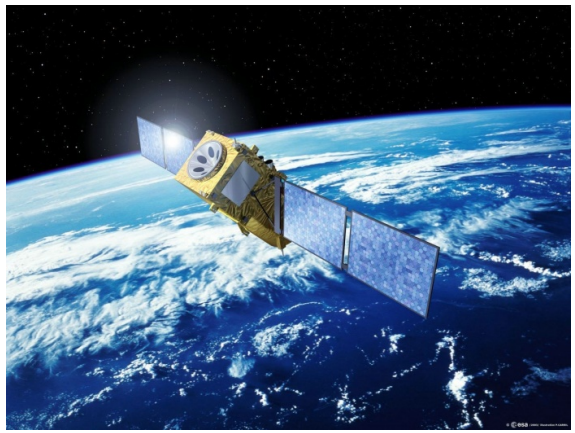


Context & motivation

Consecutive issues

- Huge quantity $\rightarrow$ limited time and space complexities
- High dimensionality
- Distribution or real time extraction
- Unlabelled data

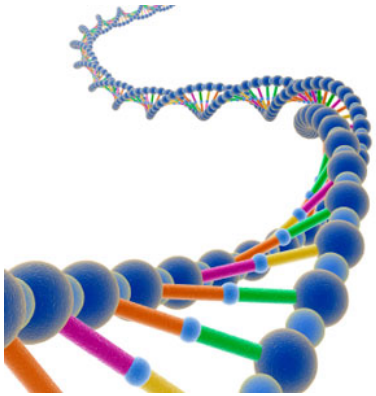


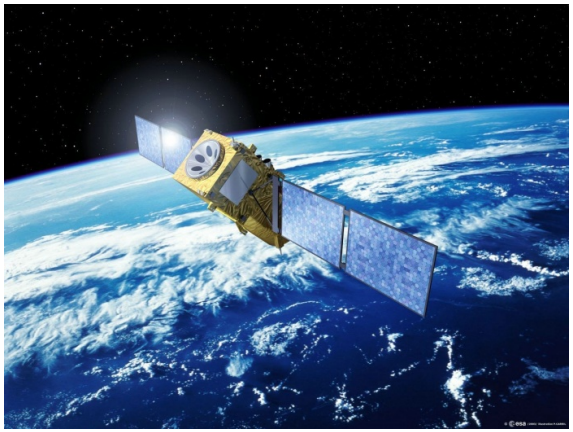


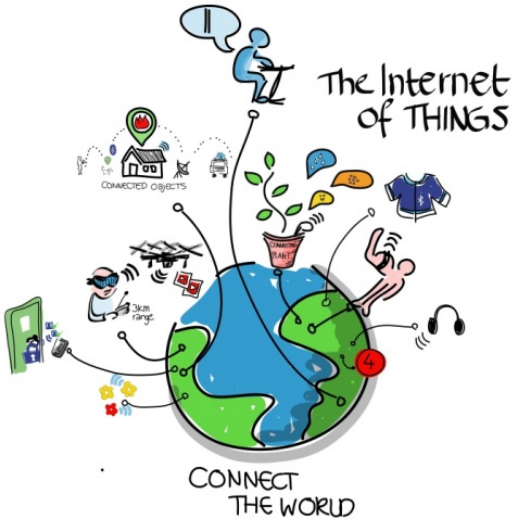
Context & motivation

Consecutive issues

- Huge quantity → limited time and space complexities
- High dimensionality → the *curse of dimensionality*
- Distribution or real time extraction
- Unlabelled data



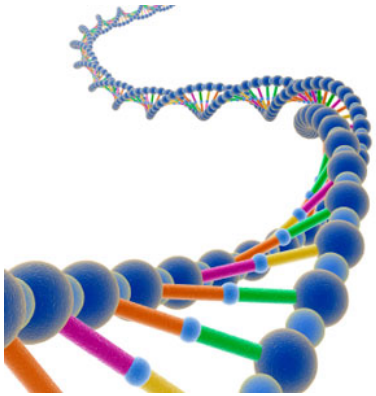


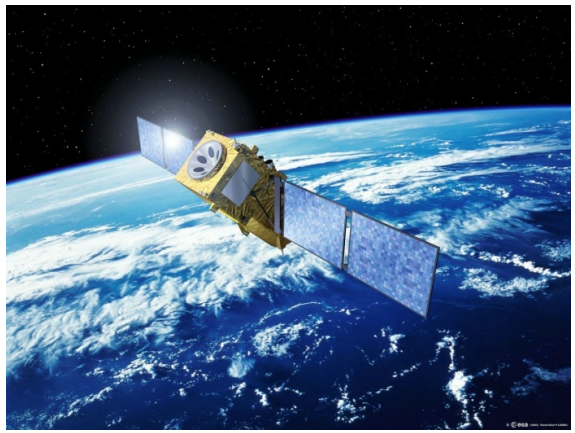


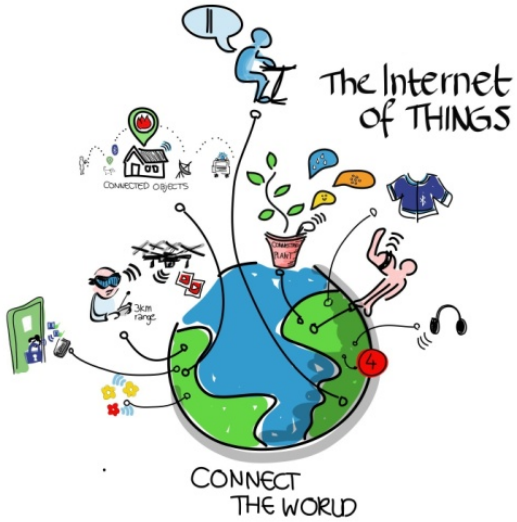
Context & motivation

Consecutive issues

- Huge quantity → limited time and space complexities
- High dimensionality → the *curse of dimensionality*
- Distribution or real time extraction → streaming data, online analysis
- Unlabelled data



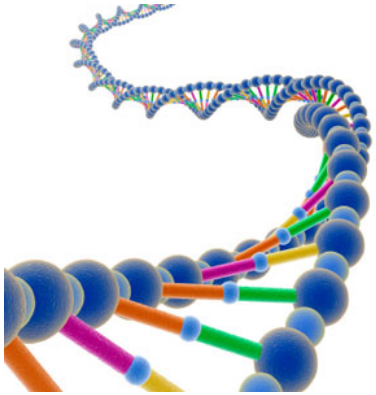


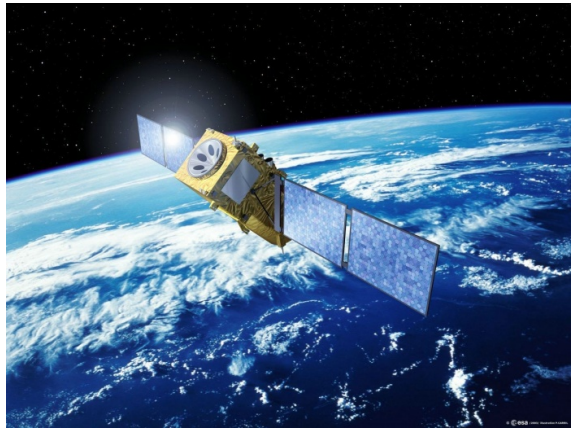


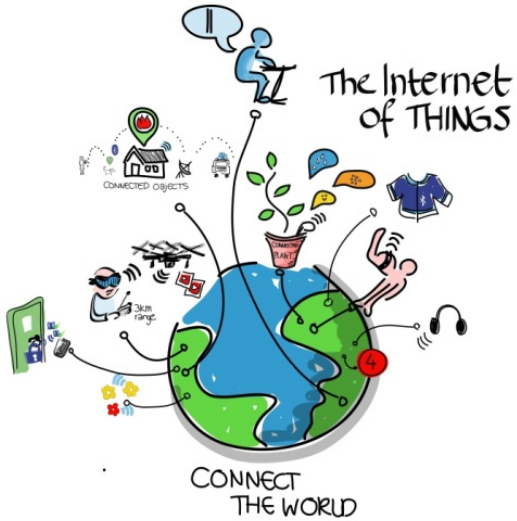
Context & motivation

Consecutive issues

- Huge quantity $\rightarrow$ limited time and space complexities
- High dimensionality $\rightarrow$ the *curse of dimensionality*
- Distribution or real time extraction $\rightarrow$ streaming data, online analysis
- Unlabelled data $\rightarrow$ unsupervised learning

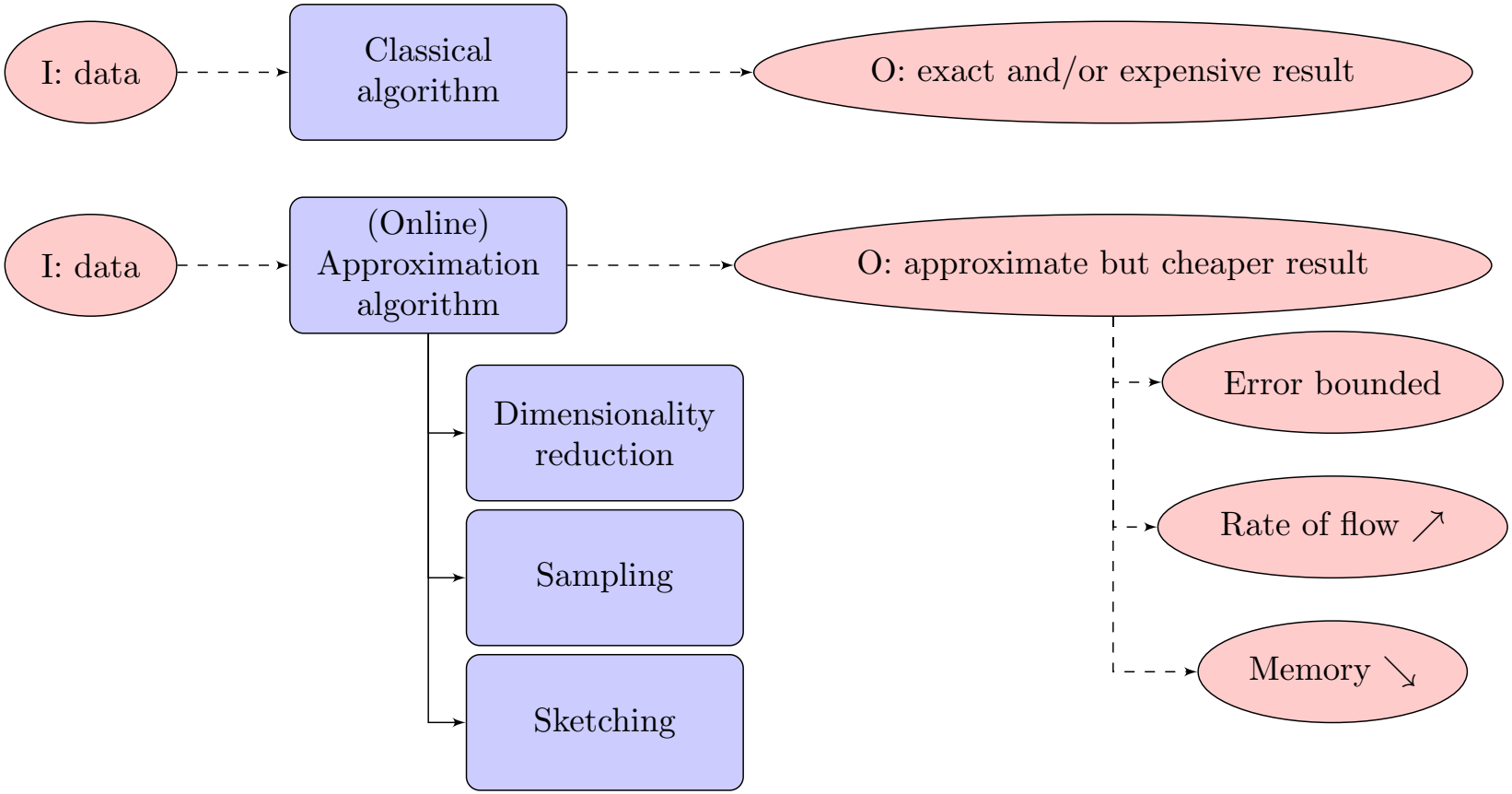






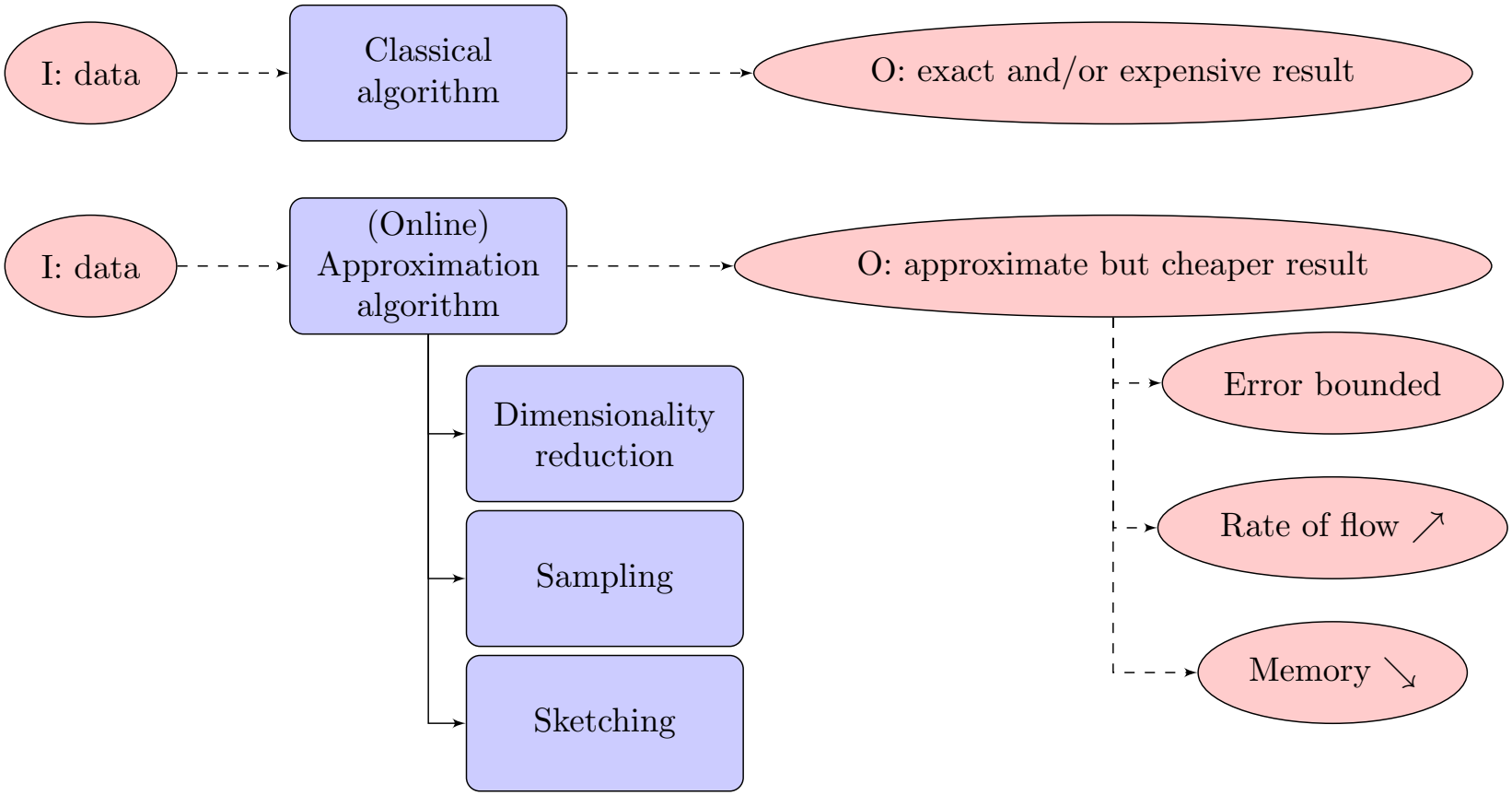
Considered approach

Approximate data *distance - or structure - preserving transformations*

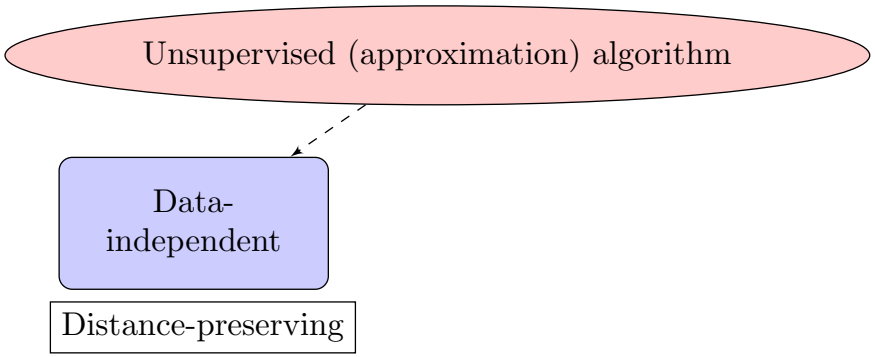


Goal of the PhD thesis

How to perform efficiently unsupervised machine learning such as the nearest neighbor search and clustering tasks, under time and space constraints for high-dimensional datasets?

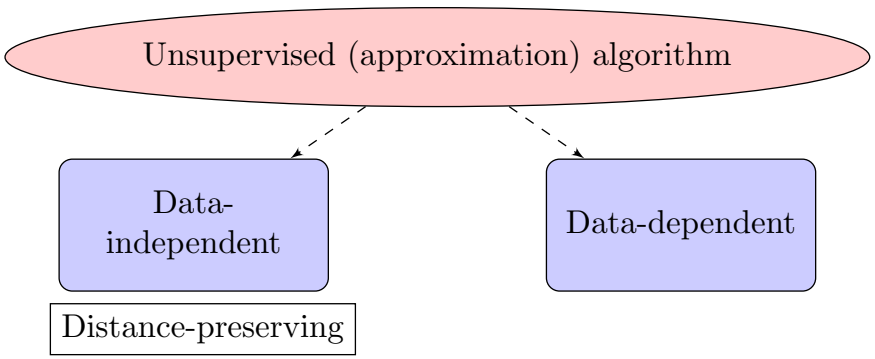


Some approaches in unsupervised learning to consider for approximation



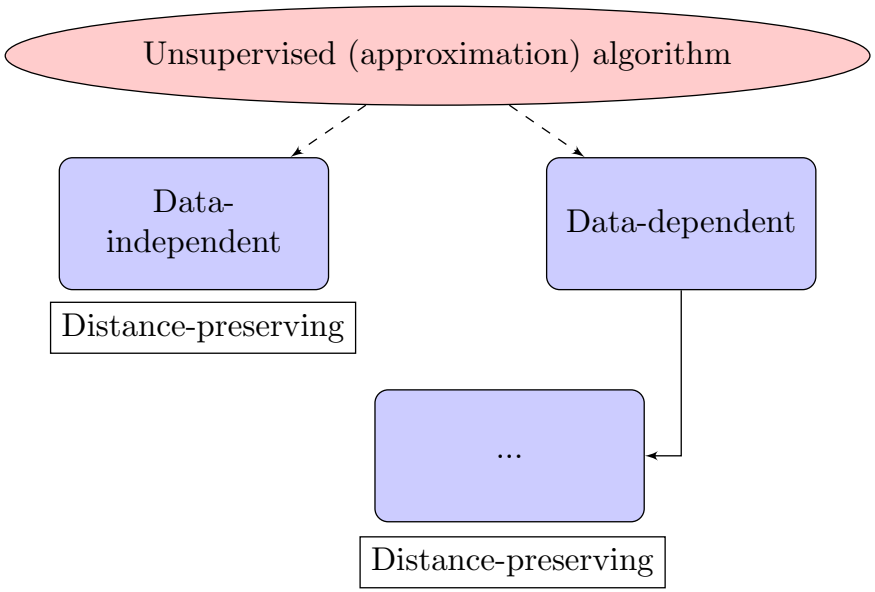
- **Data-independent:**
  Random projection [Johnson and Lindenstrauss, 1984] and structured variants [Ailon and Chazelle, 2006, Andoni et al., 2015]
  ✗ theoretical guarantees for $\mathbf{HD}_3\mathbf{HD}_2\mathbf{HD}_1$ [Andoni et al., 2015]

Some approaches in unsupervised learning to consider for approximation



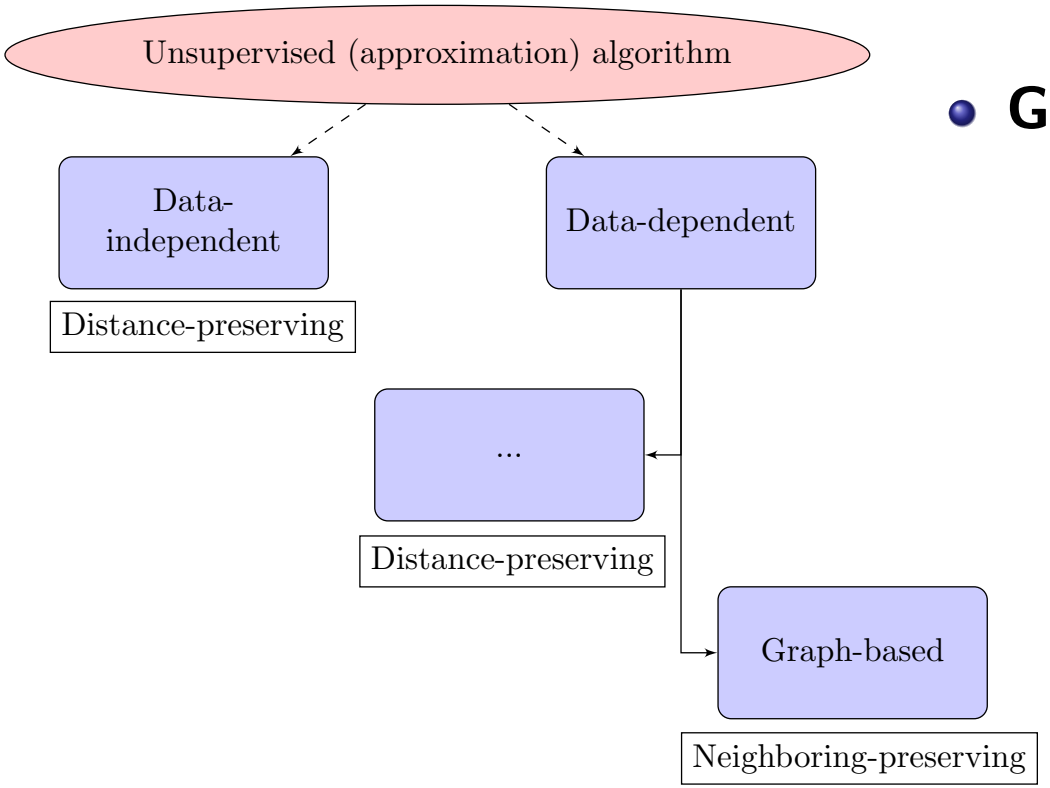
- **Data-independent:**
  Random projection [Johnson and Lindenstrauss, 1984] and structured variants [Ailon and Chazelle, 2006, Andoni et al., 2015]
  ✗ theoretical guarantees for $\mathbf{HD}_3\mathbf{HD}_2\mathbf{HD}_1$ [Andoni et al., 2015]
- **Data-dependent**

Some approaches in unsupervised learning to consider for approximation



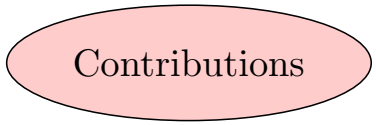
- **Data-dependent:** Learning to hash [Wang et al., 2018], ex: ITQ [Gong et al., 2013], Iso-Hash [Kong and Li, 2012], OSH [Leng et al., 2015]
  - ✓ accuracy
  - ≈ online
  - ✗ theoretical guarantees

Some approaches in unsupervised learning to consider for approximation

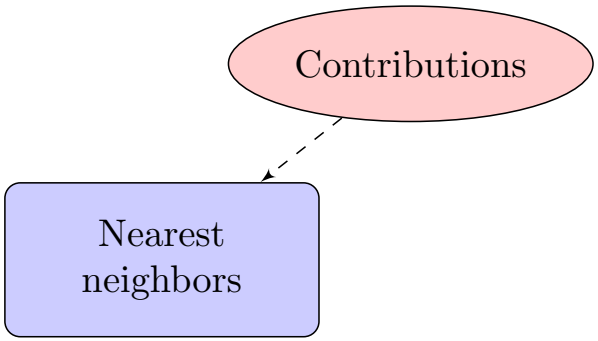


- **Graph-based:**
  - Convex-optimization,
    ✓ accuracy
    ✓ theoretical guarantees
    ✗ scalability
  - MST-based, etc.
    ✗ accuracy
    ✗ theoretical guarantees

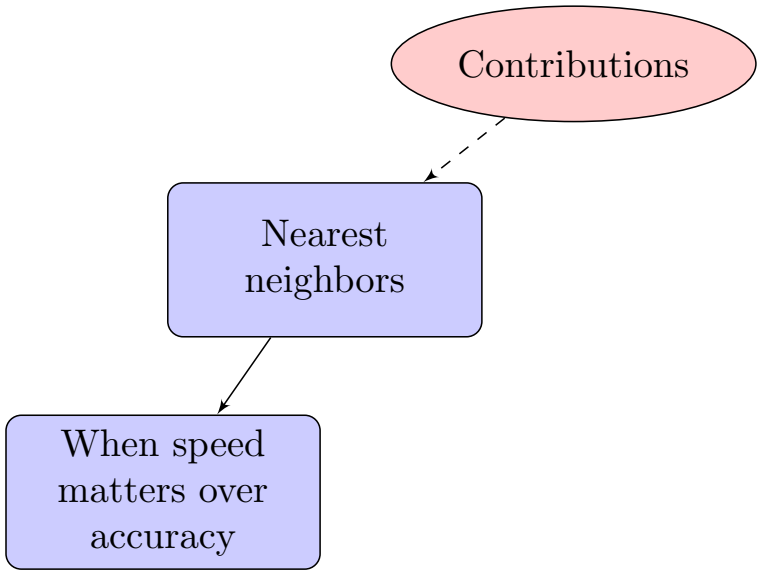
Overview of the contributions



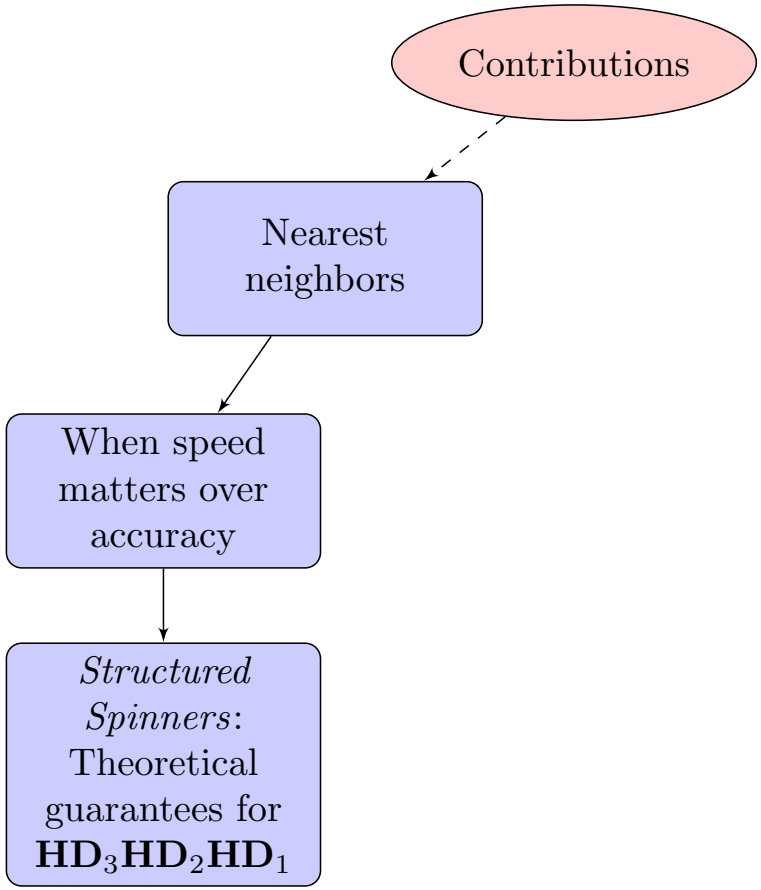
Overview of the contributions



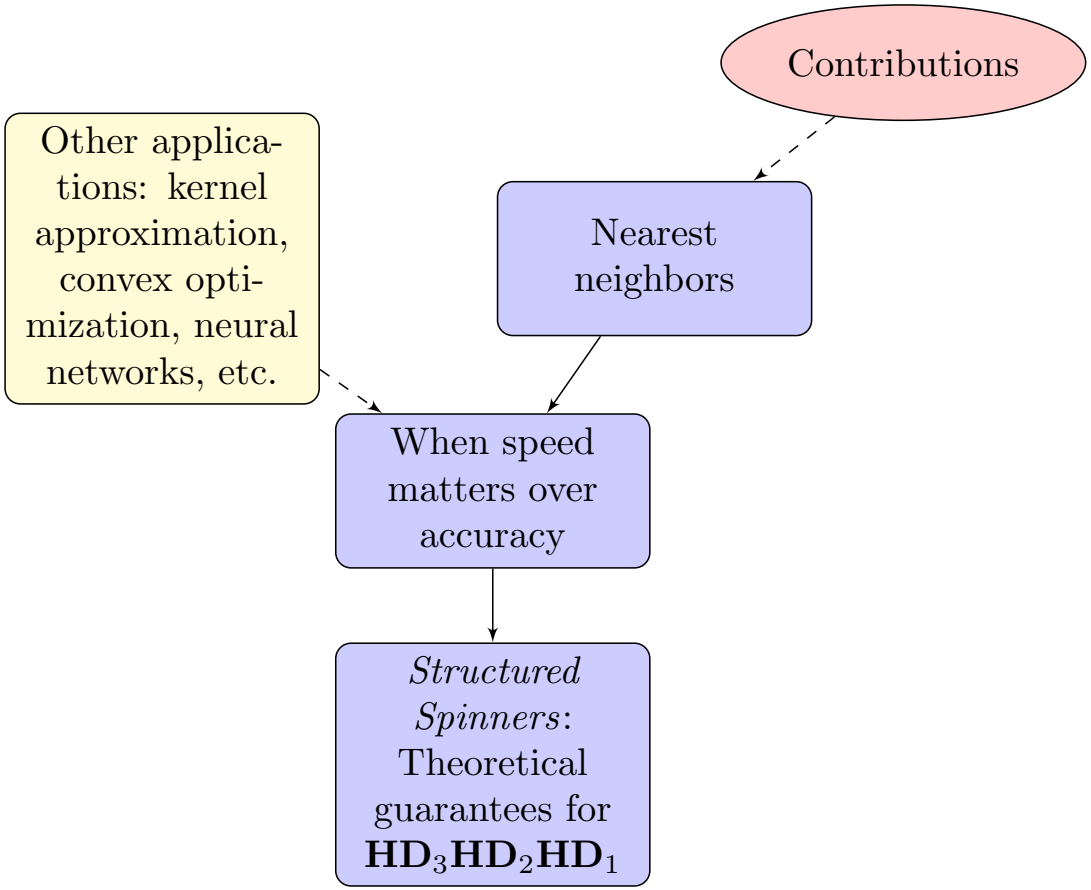
Overview of the contributions



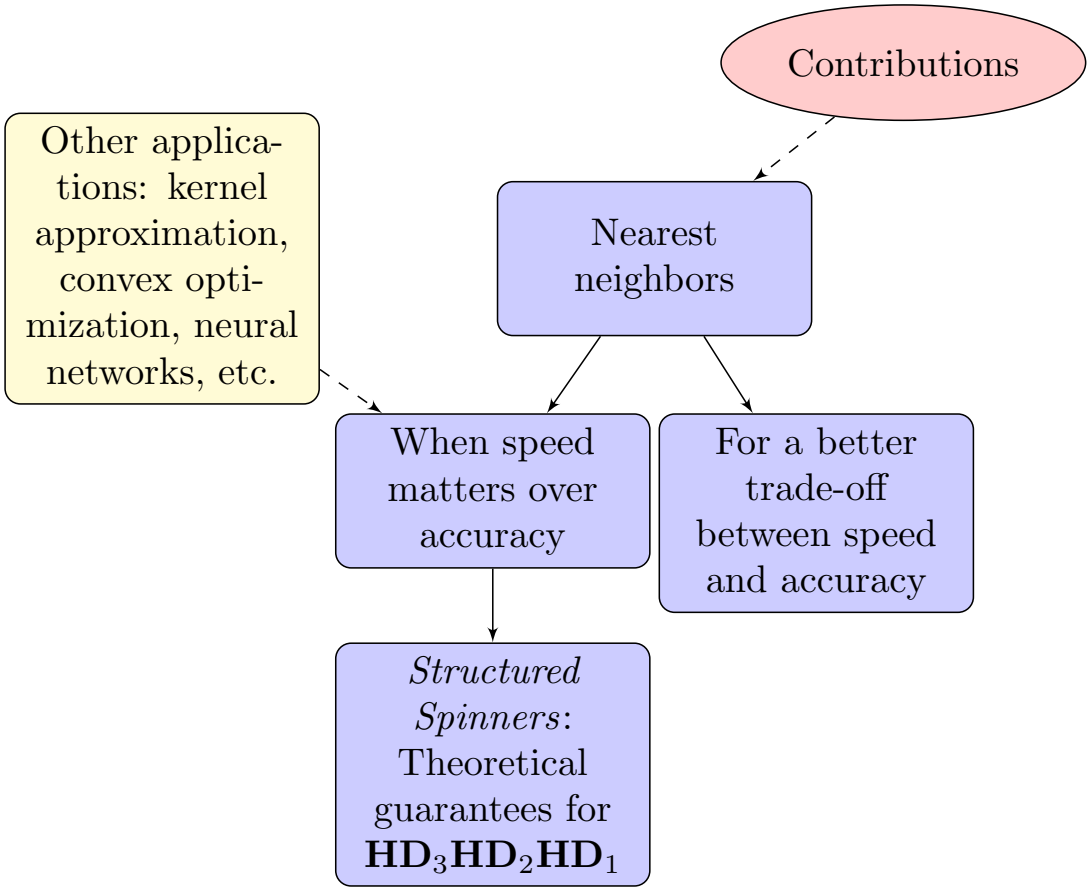
Overview of the contributions



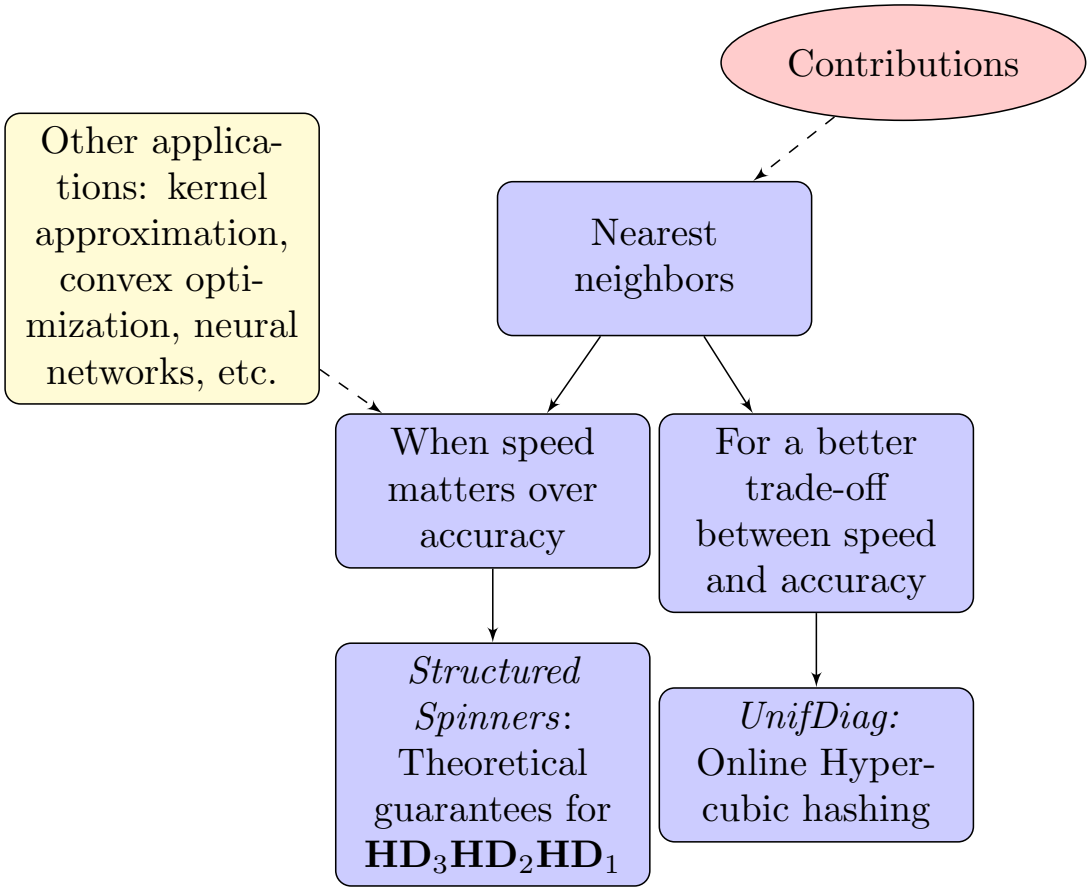
Overview of the contributions



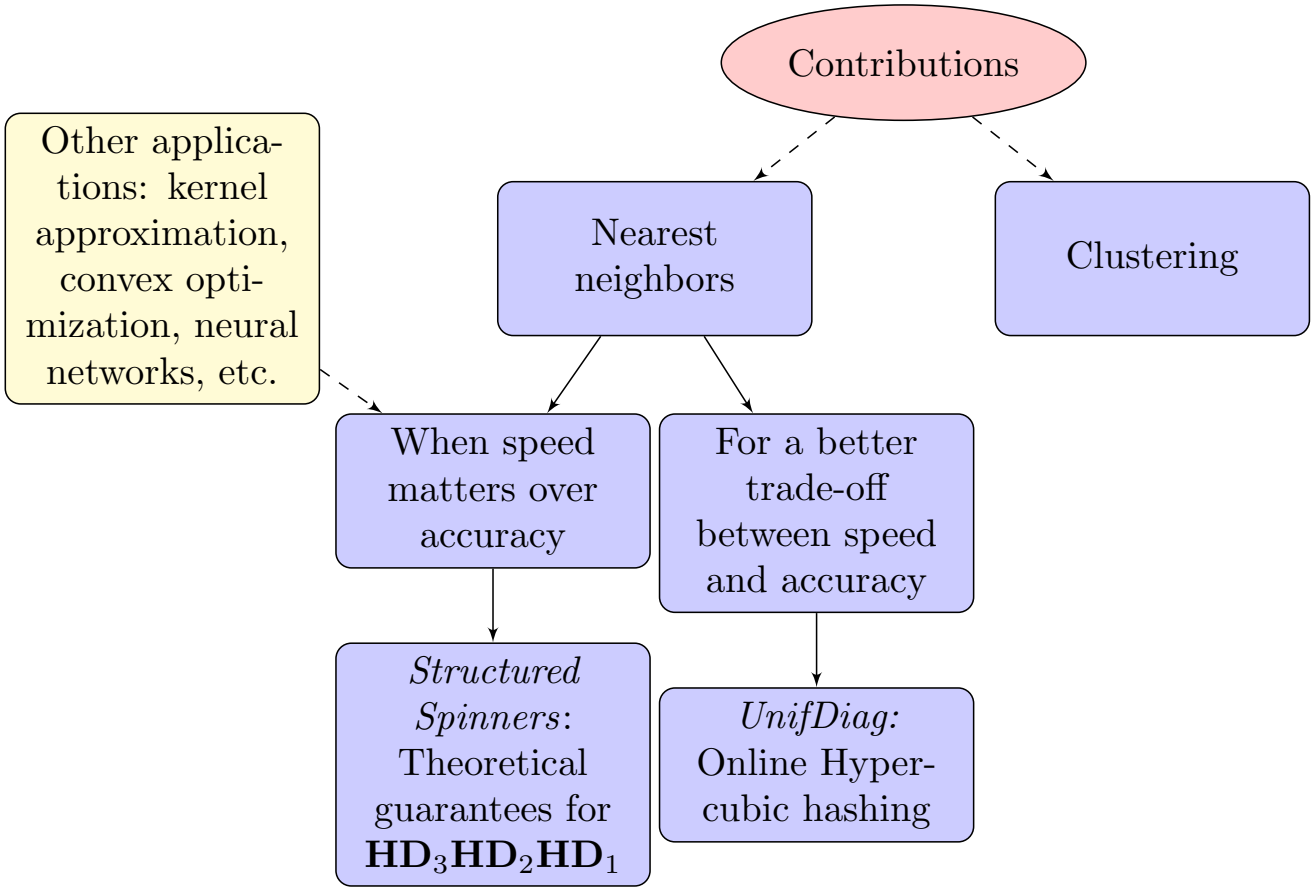
Overview of the contributions



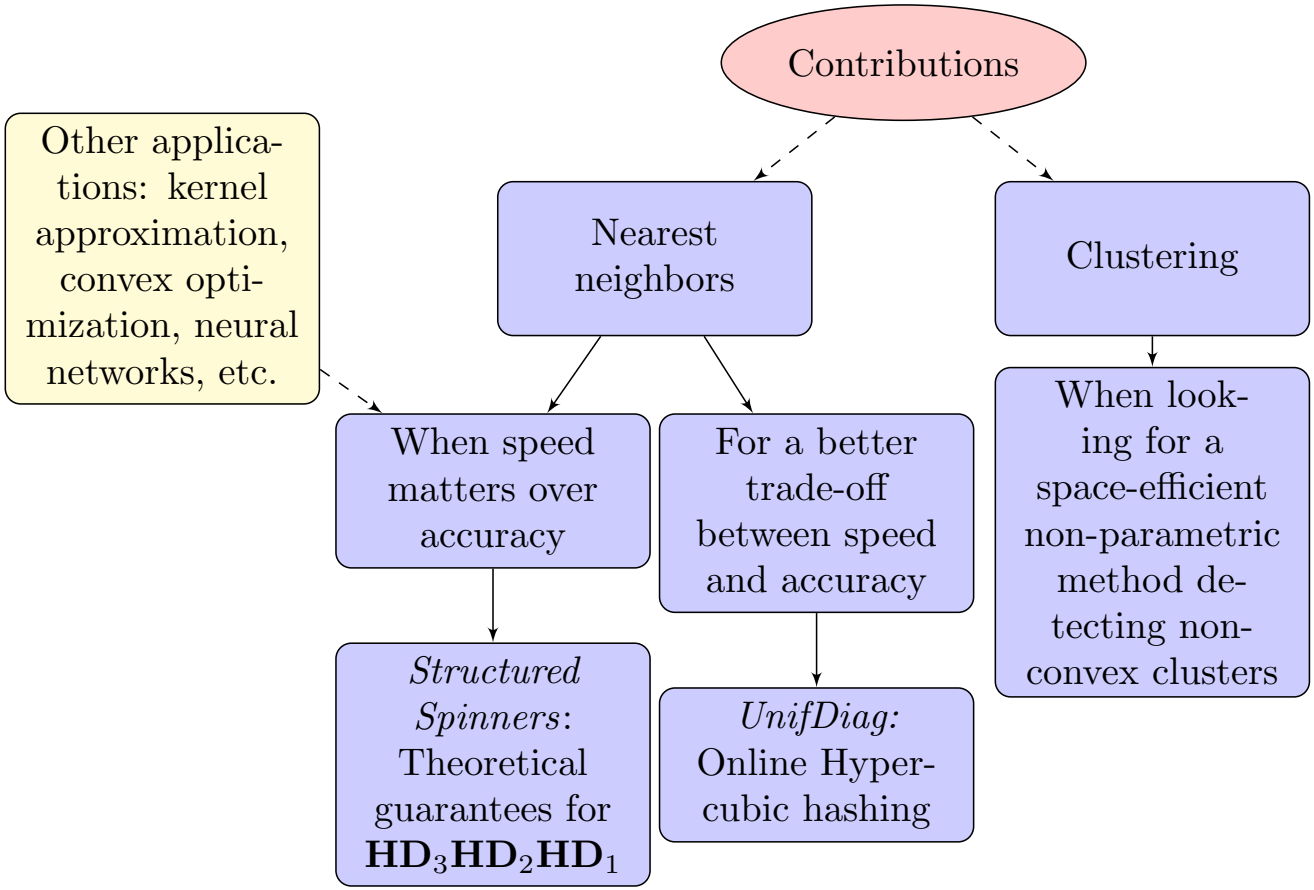
Overview of the contributions



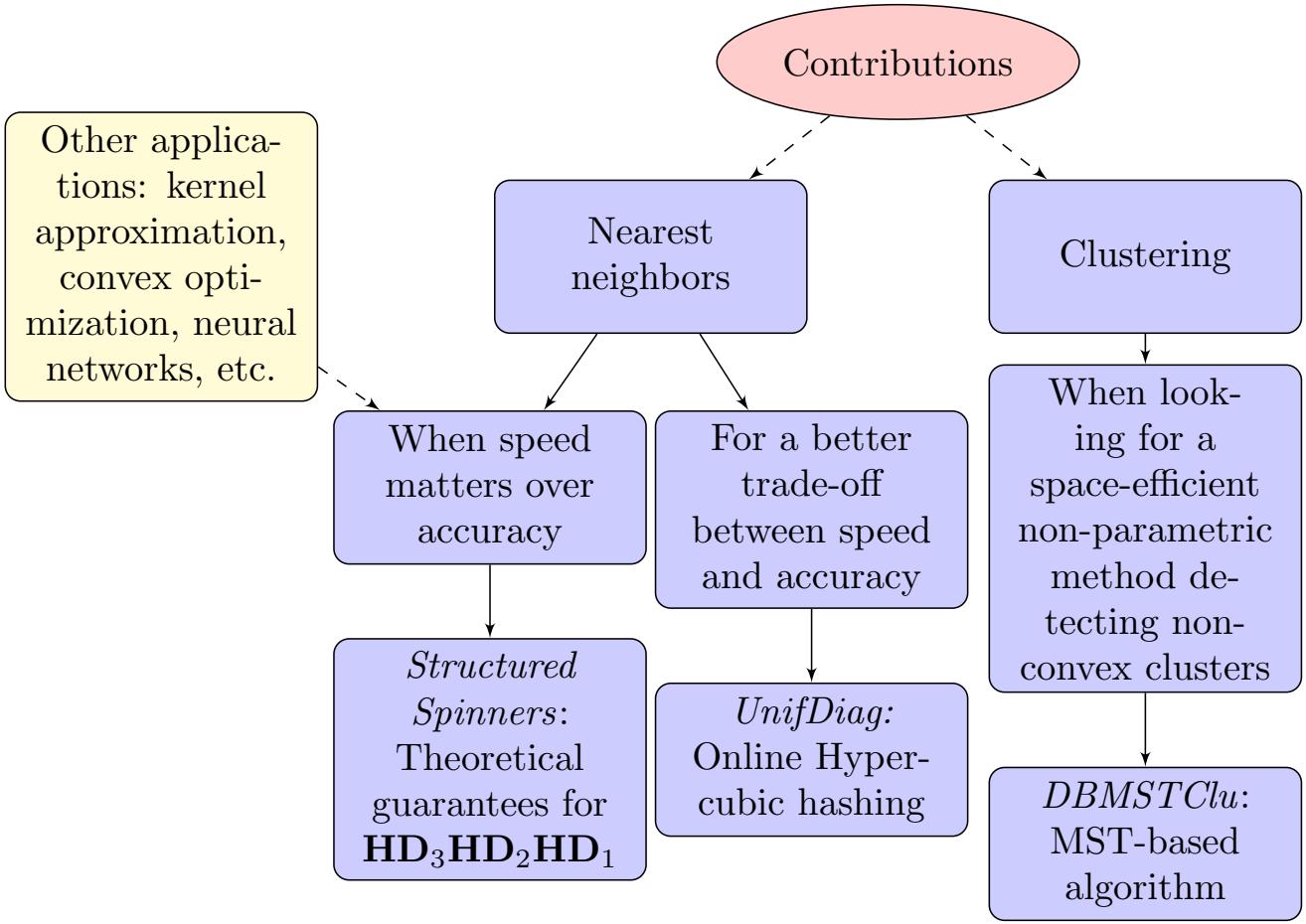
Overview of the contributions



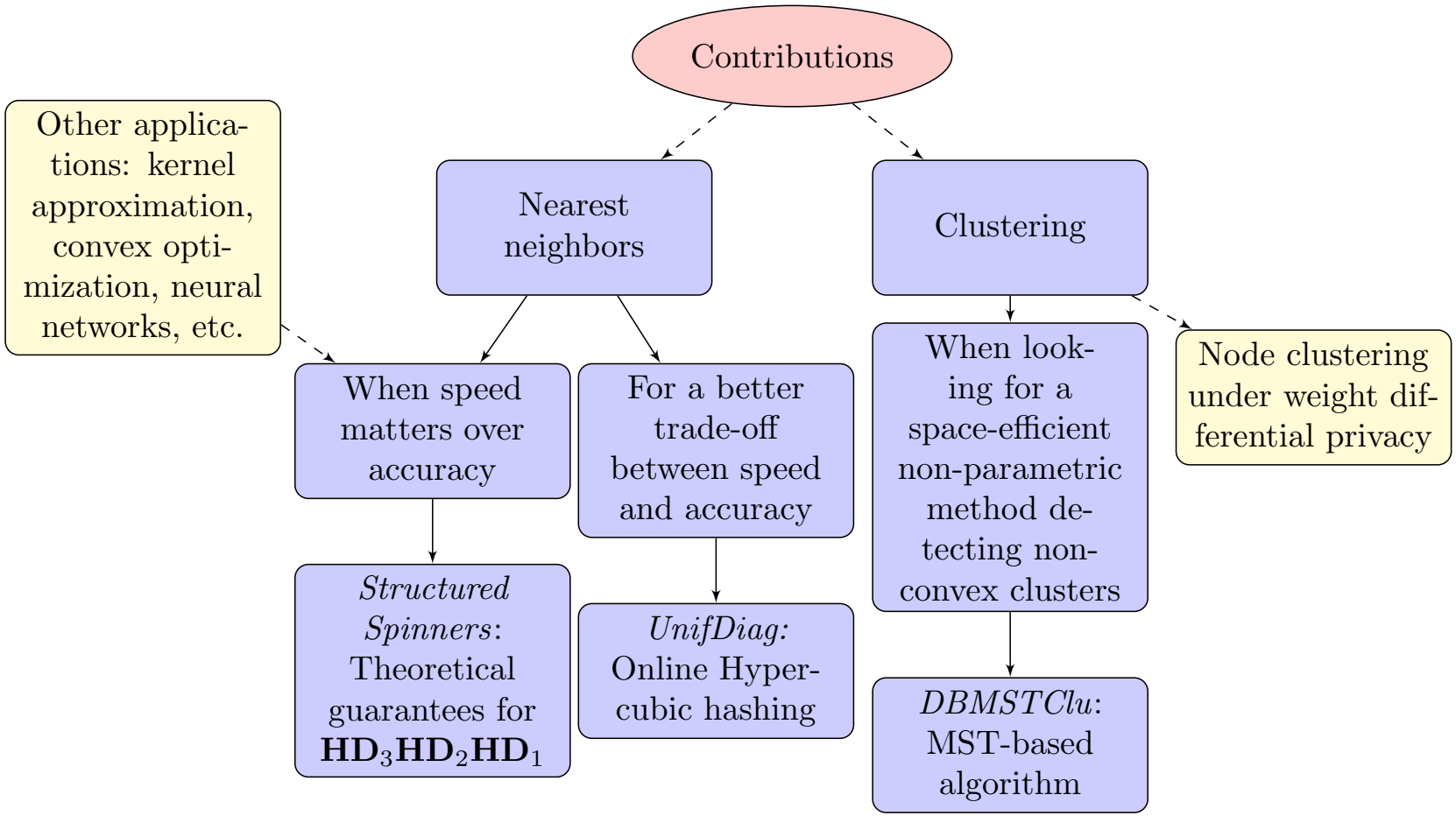
Overview of the contributions



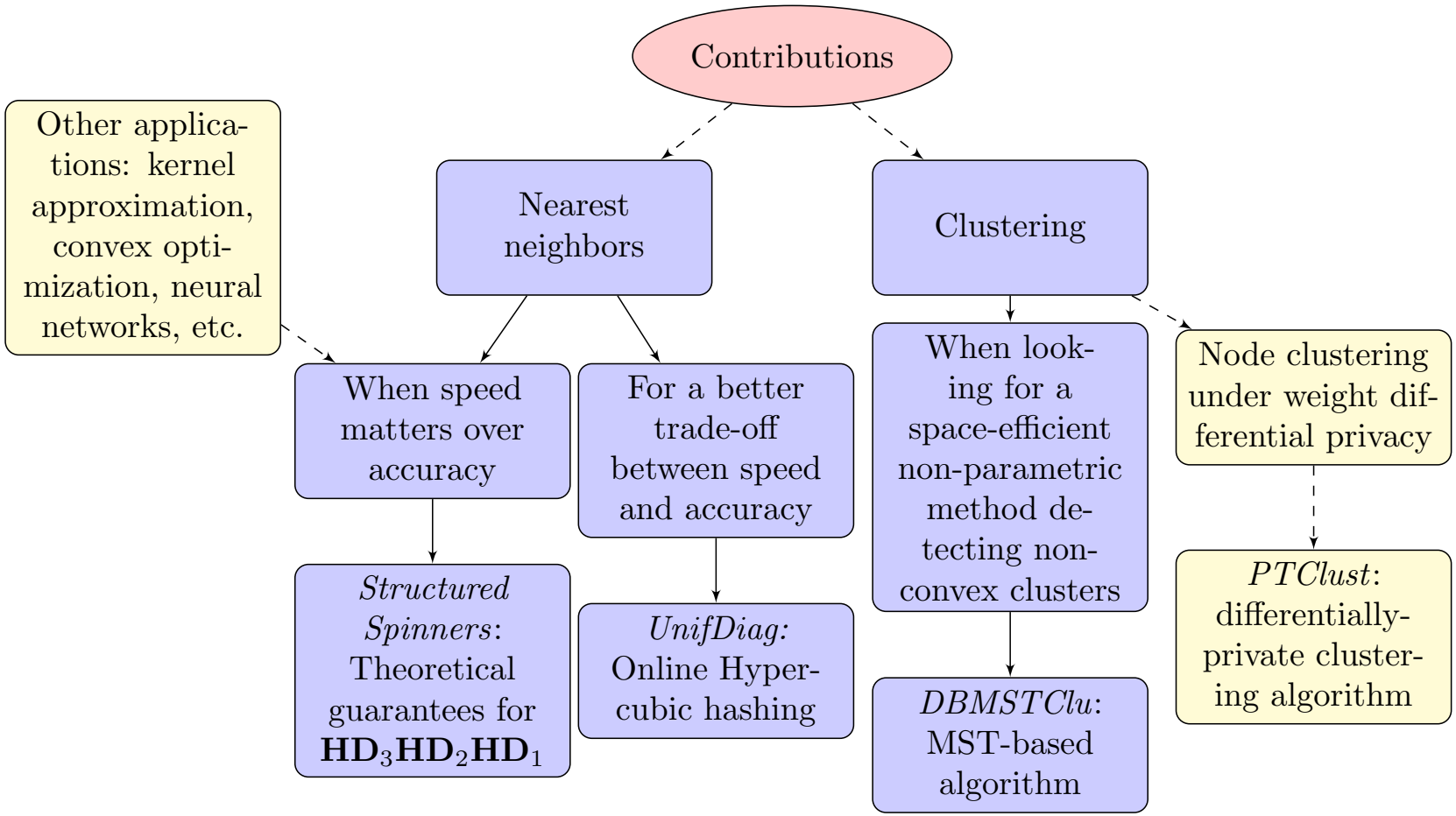
Overview of the contributions



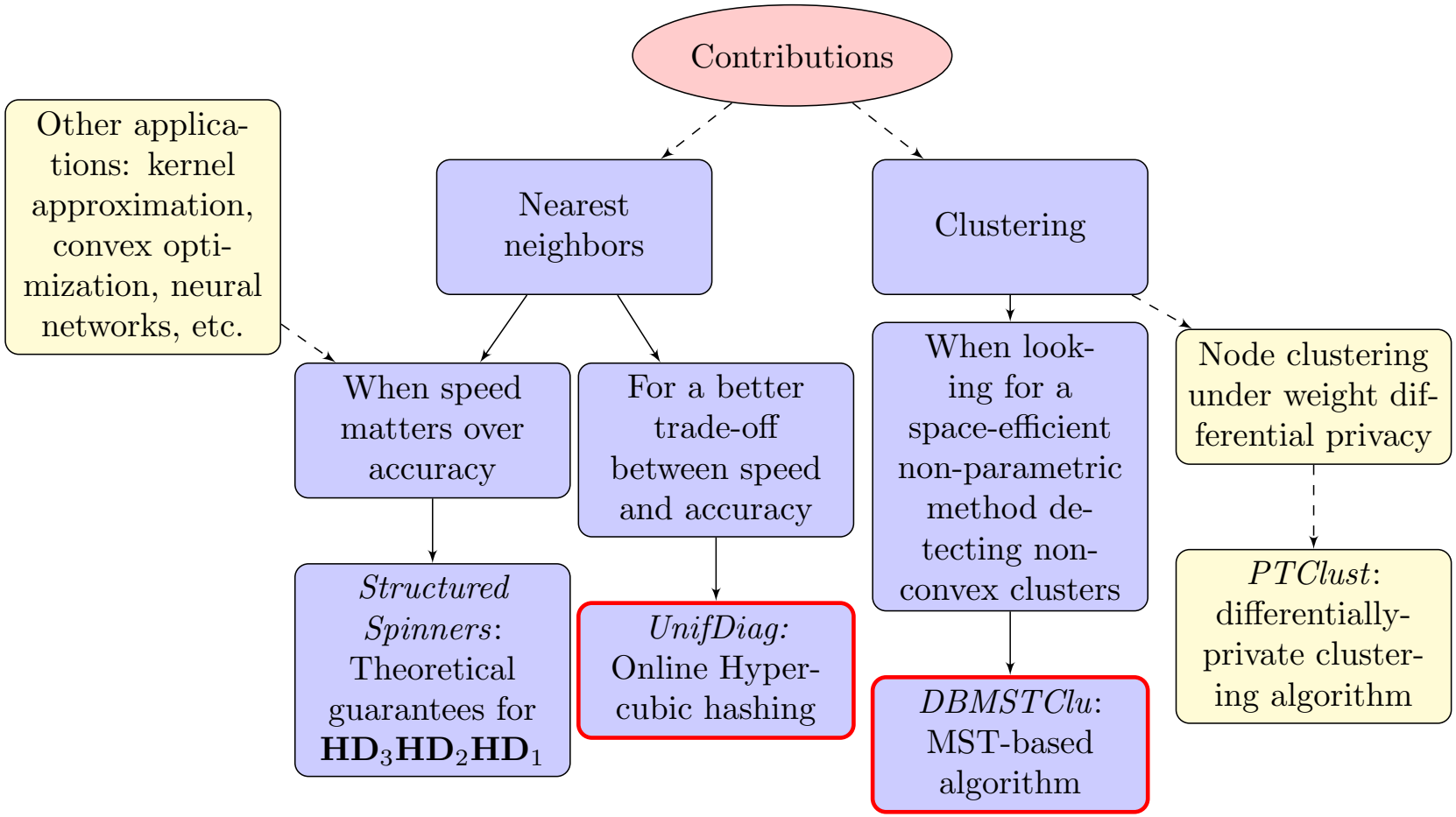
Overview of the contributions



Overview of the contributions



Overview of the contributions



Plan

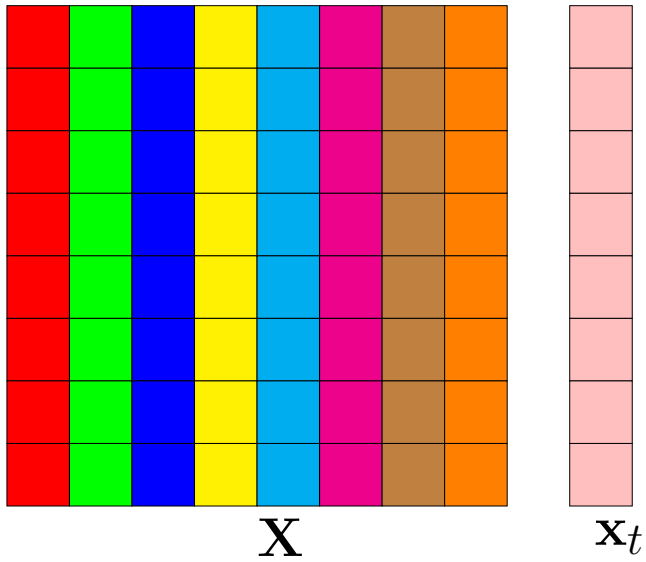
Learning compact binary codes of massive data streams

UnifDiag

A new online method for computing distance-preserving compact c-bits codes of high-dimensional data stream to perform efficient similarity search.

Streaming

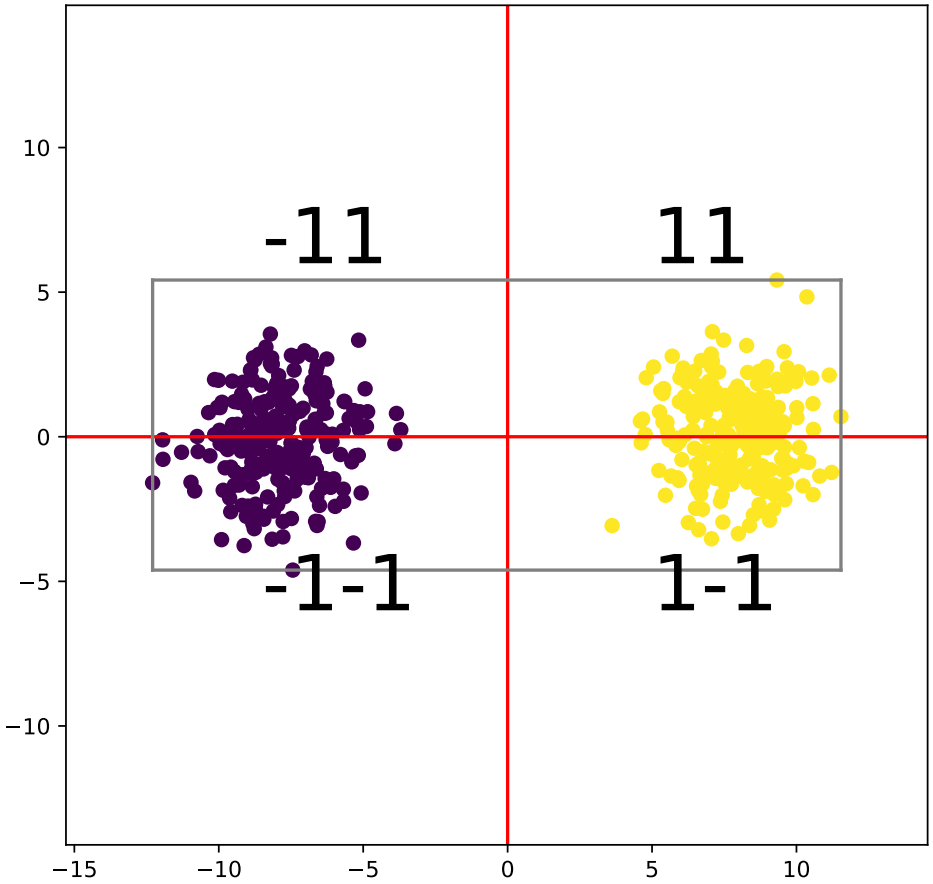
- Process one point at a time
- Sublinear space cost / time cost measured per input data



Let's take a look at offline ITQ [Gong et al., 2013]

- $\mathbf{X} \in \mathbb{R}^{d \times N}$, dataset
- For $x \in \mathbb{R}$, $\operatorname{sign}(x) = 1$ if $x \geq 0$ and -1 otherwise. On vectors, applied pointwise.

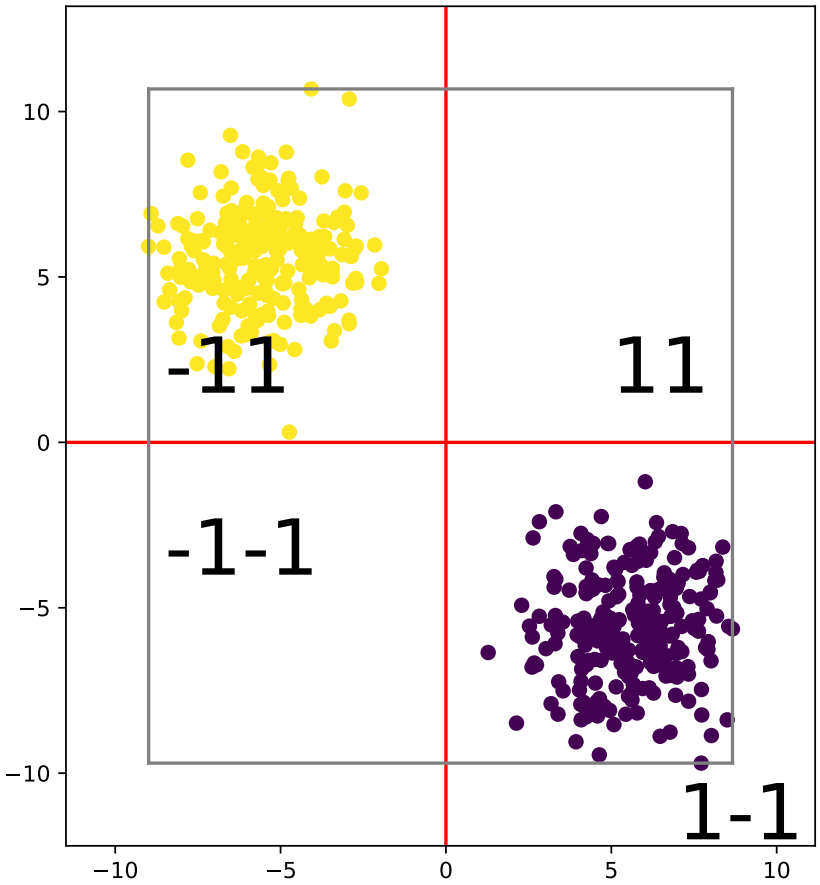
ITQ Principle



after PCA projection

- $\mathbf{X} \in \mathbb{R}^{d \times N}$, dataset
- $\mathbf{W} \in \mathbb{R}^{c \times d}$, $c \ll d$, PCA projection
- $\mathbf{V} = \mathbf{W}\mathbf{X} \in \mathbb{R}^{c \times N}$, PCA-projected dataset

ITQ Principle



after ITQ rotation

- $\mathbf{Y} = \mathbf{RV}$, rotated PCA-projected dataset
- $\mathbf{R}\mathbf{R}^T = \mathbf{R}^T\mathbf{R} = \mathbf{I}_c$
- $\tilde{\mathbf{W}} = \mathbf{RW}$
- $\mathbf{B} = \text{sign}(\mathbf{Y}) \in \{-1, 1\}^{c \times N}$
- $\mathbf{R}^* = \text{argmin}_{\mathbf{B},\mathbf{R}} \|\mathbf{B} - \tilde{\mathbf{W}}\mathbf{X}\|_F^2$

Key challenges in the online setting

- Define $\tilde{\mathbf{W}}_t \in \mathbb{R}^{c \times d}$ s.t. $c \ll d$ and the c-bits binary code

$$\mathbf{b}_t = \text{sign}(\tilde{\mathbf{W}}_t \mathbf{x}_t).$$

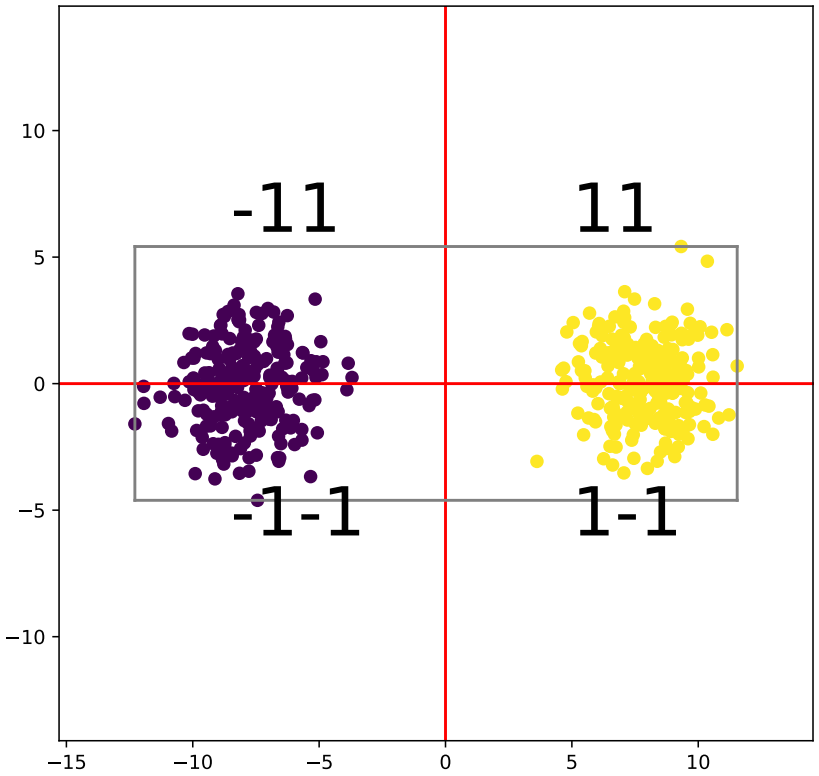
$$\tilde{\mathbf{W}}_t = \mathbf{R}_t \mathbf{W}_t$$

with $\mathbf{W}_t \in \mathbb{R}^{c \times d}$ principal subspace and

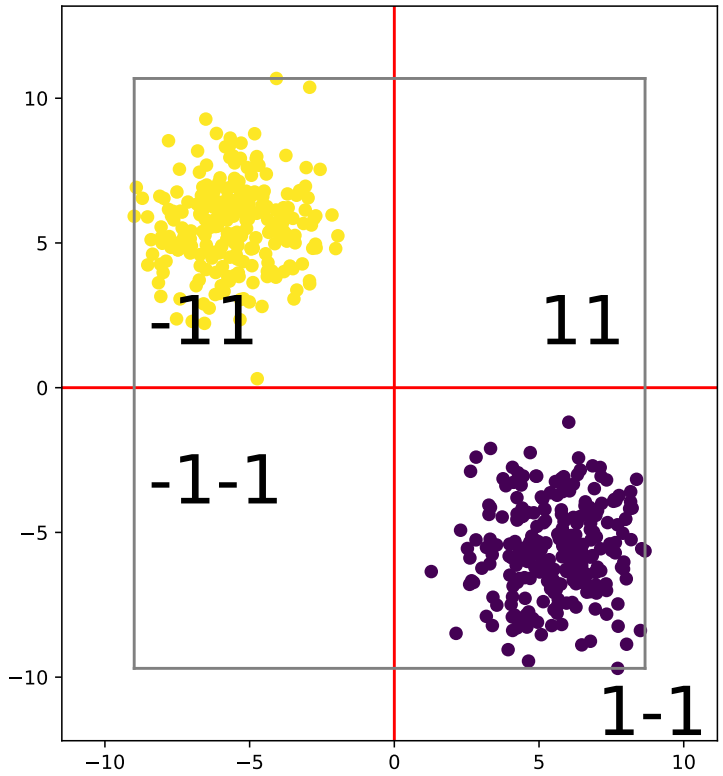
$$\mathbf{R}_t \mathbf{R}_t^T = \mathbf{R}_t^T \mathbf{R}_t = \mathbf{I}_c$$

- $\mathbf{W}_t$: how to estimate online the eigen subspace? $\rightarrow$ OPAST [Abed-Meraim et al., 2000]
- Online optimization for $\mathbf{R}_t$: $\mathbf{R}^* = \text{argmin}_{\mathbf{B},\mathbf{R}} \|\mathbf{B} - \tilde{\mathbf{W}}\mathbf{X}\|_F^2$

Key challenges in the online setting

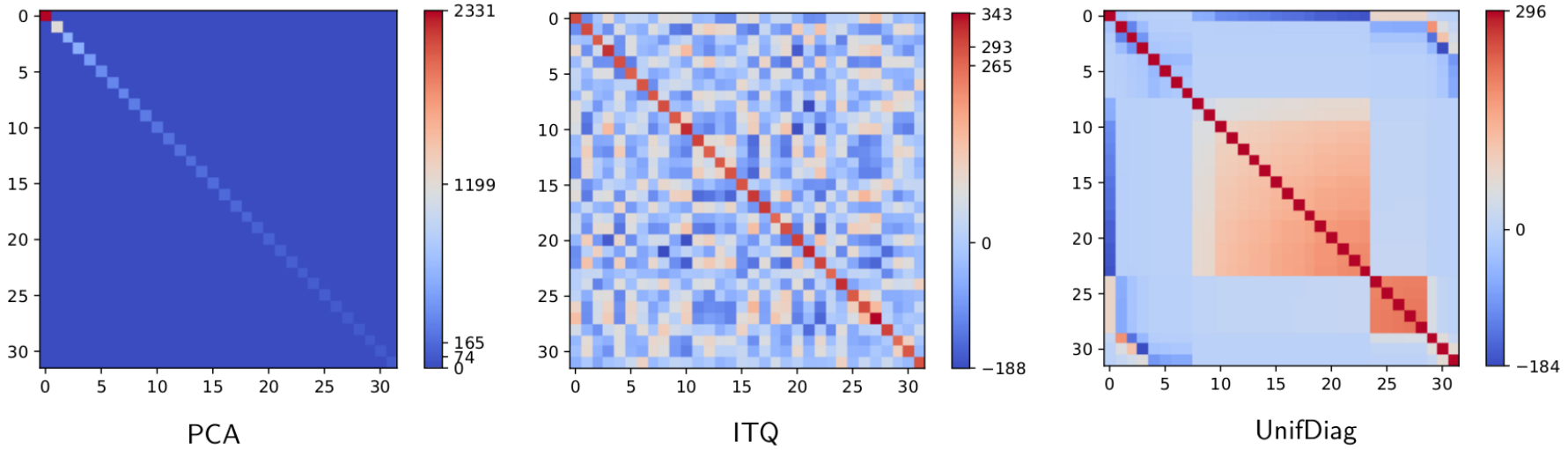


after PCA projection



after ITQ rotation

Key challenges in the online setting



Covariance matrices for 32-bits codes

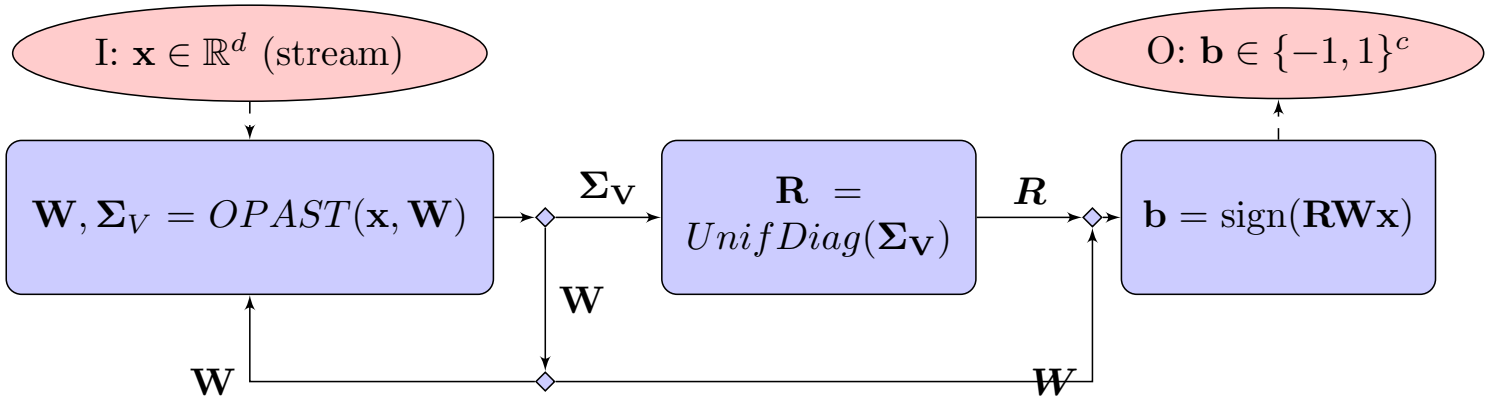
- Importance of $\mathbf{R}_t$: without, variance concentrated on the first dimensions
- How to define a rotation $\mathbf{R}_t$ balancing the variance over the different directions? (i.e. set to value $\tau = \mathrm{Tr}(\mathbf{\Sigma_V})/c$)

Givens rotation and notations

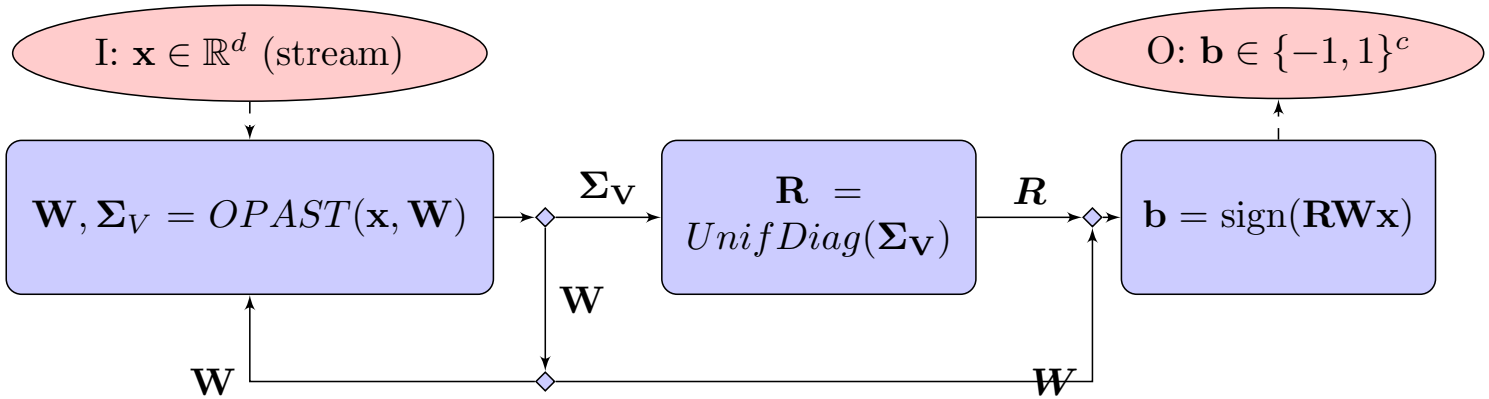
$$\mathbf{G}(i,j,\theta) = \begin{bmatrix} 1 & \cdots & 0 & \cdots & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & & \vdots & & \vdots \\ 0 & \cdots & cos(\theta) & \cdots & -sin(\theta) & \cdots & 0 \\ \vdots & & \vdots & \ddots & \vdots & & \vdots \\ 0 & \cdots & sin(\theta) & \cdots & cos(\theta) & \cdots & 0 \\ \vdots & & \vdots & & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & \cdots & 0 & \cdots & 1 \end{bmatrix}$$

- $\mathbf{R}$ defined as a product of $c-1$ Givens rotations $\{\mathbf{G}(i_r, j_r, \theta_r)\}_{1 \leq r \leq c-1}$ iteratively applied left and right to $\mathbf{\Sigma_V}$.

Learning compact binary codes of massive data streams



Learning compact binary codes of massive data streams



Method	Unity for time	Time $\mathbf{W}$	Space $\mathbf{W}$	Time $\mathbf{R}$	Space $\mathbf{R}$
UnifDiag + OPAST	per data point	$O(dc + c^2)$	$O(dc + c^2)$	$O(c^2)$	$O(c^2)$

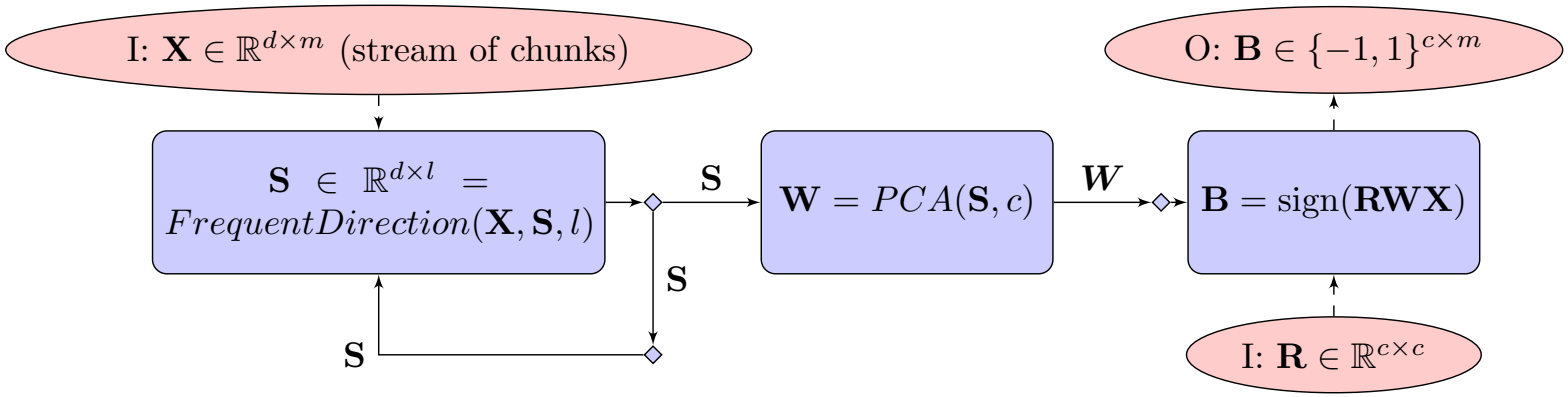
Comparison with IsoHash [Kong and Li, 2012]

$$\mathbf{R}^* = \underset{\mathbf{R}\in\mathcal{O}(c)}{\operatorname{argmin}} \frac{1}{2} \| \operatorname{diag}(\mathbf{R}^T \mathbf{\Sigma_V} \mathbf{R}) - \operatorname{diag}(\boldsymbol{\tau}) \|_F^2$$

$\mathcal{O}(c)$ set of orthogonal matrices in $\mathbb{R}^{c\times c}$

Method	Unity for time	Time $\mathbf{W}$	Space $\mathbf{W}$	Time $\mathbf{R}$	Space $\mathbf{R}$
UnifDiag + OPAST	per data point	$O(dc + c^2)$	$O(dc + c^2)$	$O(c^2)$	$O(c^2)$
IsoHash				$O(c^3)$	$O(c^2)$

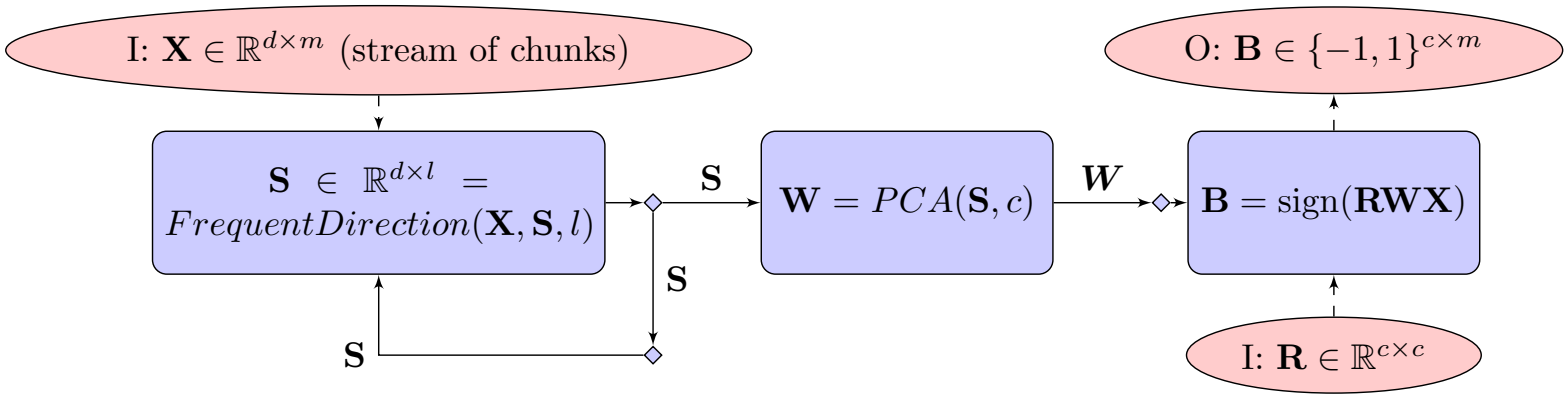
State-of-the-art: Online Sketching Hashing (OSH) [Leng et al., 2015]



$\mathbf{R}$ is a random orthonormal matrix.

Method	Unity for time	Time $\mathbf{W}$	Space $\mathbf{W}$	Time $\mathbf{R}$	Space $\mathbf{R}$
UnifDiag + OPAST	per data point	$O(dc + c^2)$	$O(dc + c^2)$	$O(c^2)$	$O(c^2)$

State-of-the-art: Online Sketching Hashing (OSH) [Leng et al., 2015]

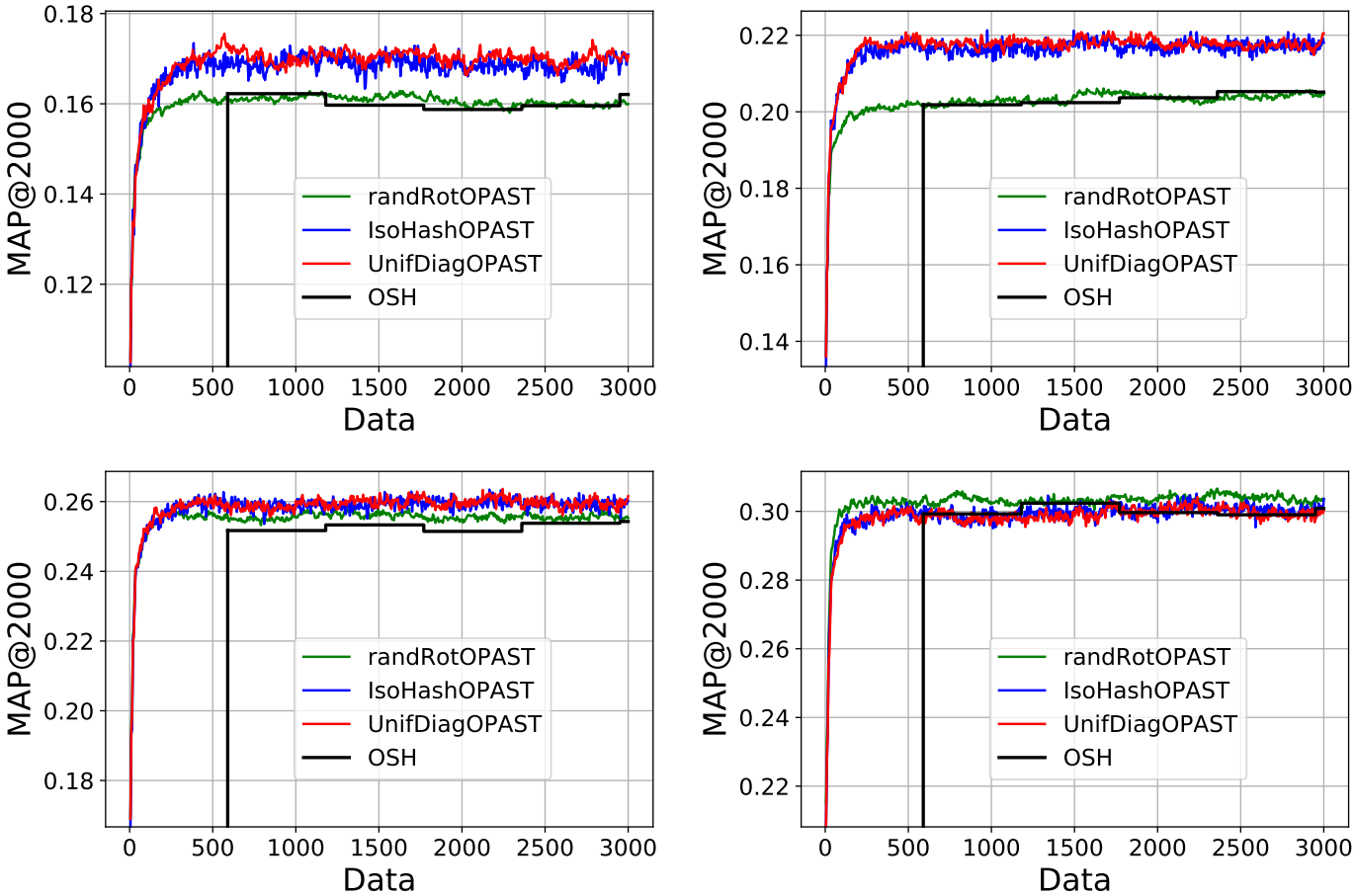


$\mathbf{R}$ is a random orthonormal matrix.

Method	Unity for time	Time $\mathbf{W}$	Space $\mathbf{W}$	Time $\mathbf{R}$	Space $\mathbf{R}$
UnifDiag + OPAST	per data point	$O(dc + c^2)$	$O(dc + c^2)$	$O(c^2)$	$O(c^2)$
OSH	per chunk	$O(dl^2 + l^3)$	FD [Liberty, 2013]: $O(dl)$ SVD: $O(dl + l^2)$	$O(c^3)$	$O(c^2)$

$$l \ll m \ll n,\ c \ll l \ll d$$

Comparison with online methods



CIFAR dataset: $MAP@2000$ in the online setting for various code lengths $c \in \{8, 16, 32, 64\}$.

Main results on the optimality of $\mathbf{R}$ for Hypercubic quantization hashing

$$\forall t \in [N],\ \mathbf{v}_t = \mathbf{W}\mathbf{x}_t$$

$$\mathbf{y}_t = \mathbf{R}\mathbf{v}_t$$

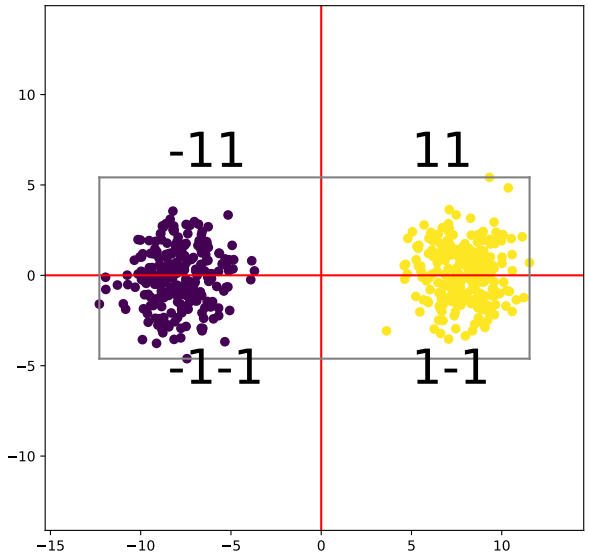
$$\mathbf{b}_t = \mathrm{sign}(\mathbf{y}_t)$$

Hypothesis 1 (H1) $\forall t \in [N]$, $(\mathbf{v}_t^{(1)}, \mathbf{v}_t^{(2)}, \ldots, \mathbf{v}_t^{(c)})^T \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma}_{\mathbf{V}}^{th})$ s.t. $\mathrm{diag}(\mathbf{\Sigma}_{\mathbf{V}}^{th}) = ({\sigma_1^{th}}^2, \ldots, {\sigma_c^{th}}^2)$.

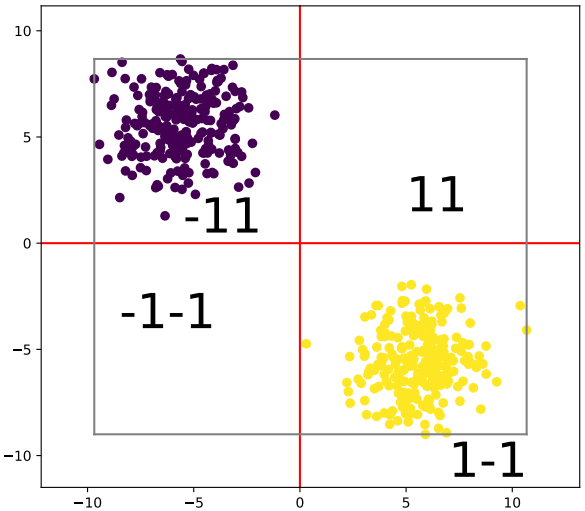
Main results on the optimality of $\mathbf{R}$ for Hypercubic quantization hashing

Theorem 1

Assume $\{\mathbf{x}_t \in \mathbb{R}^d\}_{1 \leq t \leq N}$ is a stream of N zero-centered data points following **H1**. Then, choosing $\mathbf{R} = UnifDiag(\mathbf{\Sigma}_{\mathbf{Y}}^{th}) \iff$ minimizing some upper bound on the probability that $\{\mathbf{y}_t \in \mathbb{R}^d\}_{1 \leq t \leq N}$ are close to an hyperplane delimiting an orthant.



after PCA projection



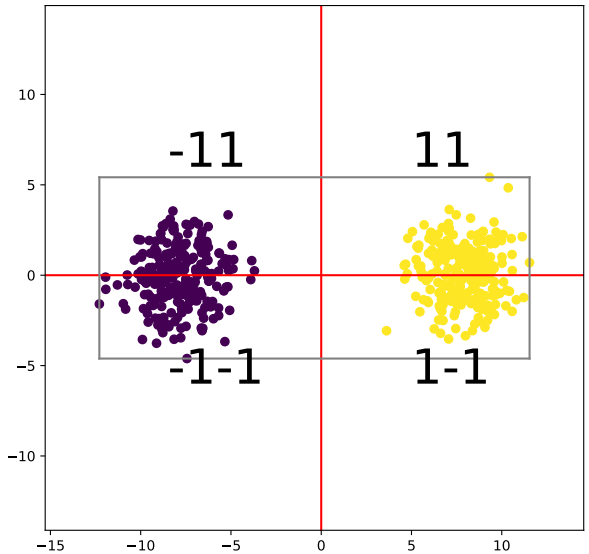
after UnifDiag rotation

Main results on the optimality of $\mathbf{R}$ for Hypercubic quantization hashing

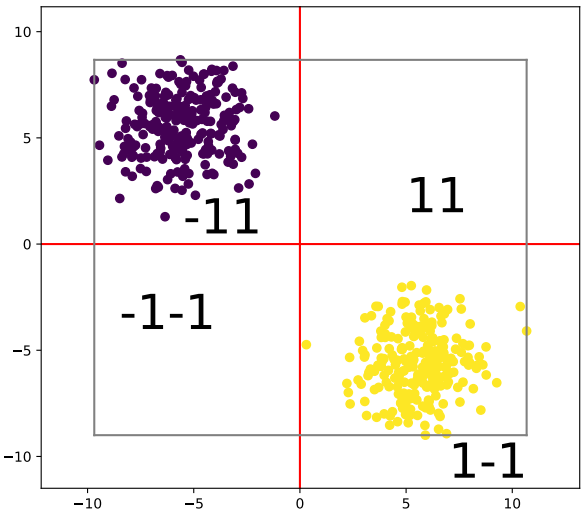
Theorem 2

Let $\mathbf{x}_{t_1} \in \mathbb{R}^d$ and $\mathbf{x}_{t_2} \in \mathbb{R}^d$ be two data points following **H1**, $\epsilon > 0$ s.t. $\|\mathbf{x}_{t_1} - \mathbf{x}_{t_2}\|_2 \leq \epsilon$ and $\mathbf{b}_{t_1} \in \{-1, 1\}^c$, $\mathbf{b}_{t_2} \in \{-1, 1\}^c$ with $\mathbf{b}_{t_i} = \text{sign}(\mathbf{RW}\mathbf{x}_{t_i})$ for $i \in \{1, 2\}$. Then,

$$\mathbb{P}[\, dist_H(\mathbf{b}_{t_1}, \mathbf{b}_{t_2}) > 0] \leq 2\epsilon\sqrt{\frac{2}{\pi}}c^{\frac{3}{2}} \left(\text{Tr}(\mathbf{\Sigma}_{\mathbf{V}}^{th})\right)^{-\frac{1}{2}}.$$



after PCA projection



after UnifDiag rotation

Conclusion for UnifDiag

Take-home message: UnifDiag is an ...

- online,
- theoretically-justified (not the case for IsoHash),
- Hypercubic quantization hashing technique for similarity search.

Method	Accuracy	Online	Space cost	Time cost	Theoretical guarantees
ITQ	✓	✗	✗	✗	✗
IsoHash	✓	≈	✓	≈	✗
OSH	✓	≈	✗	✗	✗
UnifDiag + OPAST	✓	✓	✓	✓	✓

`https://github.com/annemorvan/UnifDiagStreamBinSketching/`

Plan

Objectives

Context

Resources-limited devices collecting huge volume of data

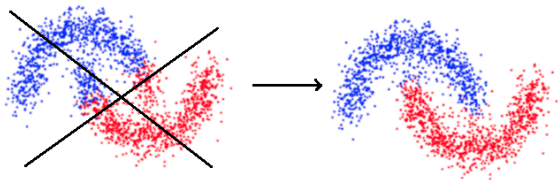
A clustering algorithm...

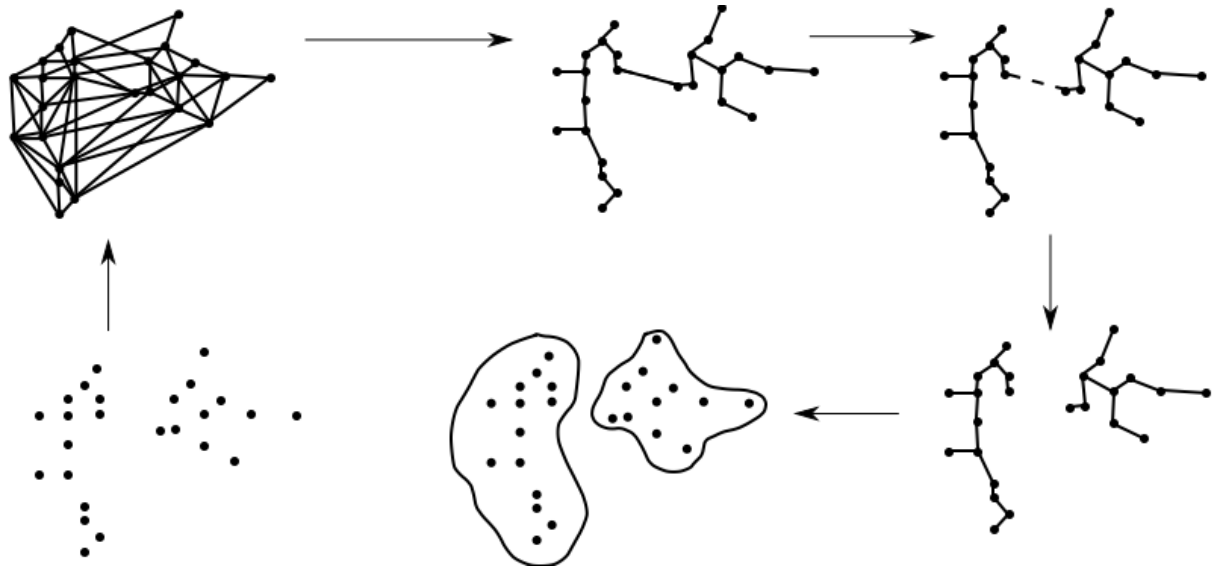
- Recognizing arbitrary non-convex cluster shapes
- With no parameter
- In a time linear to the number of points N
- Under high space constraints
- With theoretical guarantees

Principle

Minimum-Spanning-Tree-based (MST) clustering algorithm

- MST: A useful and compact summary of the data dissimilarity graph
- Appealing property: helping to recover arbitrarily-shaped clusters
- Idea: perform suitable cuts on the MST





Related work

Graph clustering [Schaeffer, 2007]

- DenGraph [Falkowski et al., 2007]: graph version of DBSCAN
- Convex optimization [Oymak and Hassibi, 2011, Chen et al., 2012, Chen et al., 2014a, Chen et al., 2014b]

MST-based clustering

- [Zahn, 1971, Asano et al., 1988, Grygorash et al., 2006]

Space-efficient clustering

- Streaming k-means [Ailon et al., 2009]: only the centroïd is stored
- CURE algorithm [Guha et al., 2001]: $O(N^2 \log(N))$ time complexity
- CluStream [Aggarwal et al., 2003] and DenStream [Cao et al., 2006]

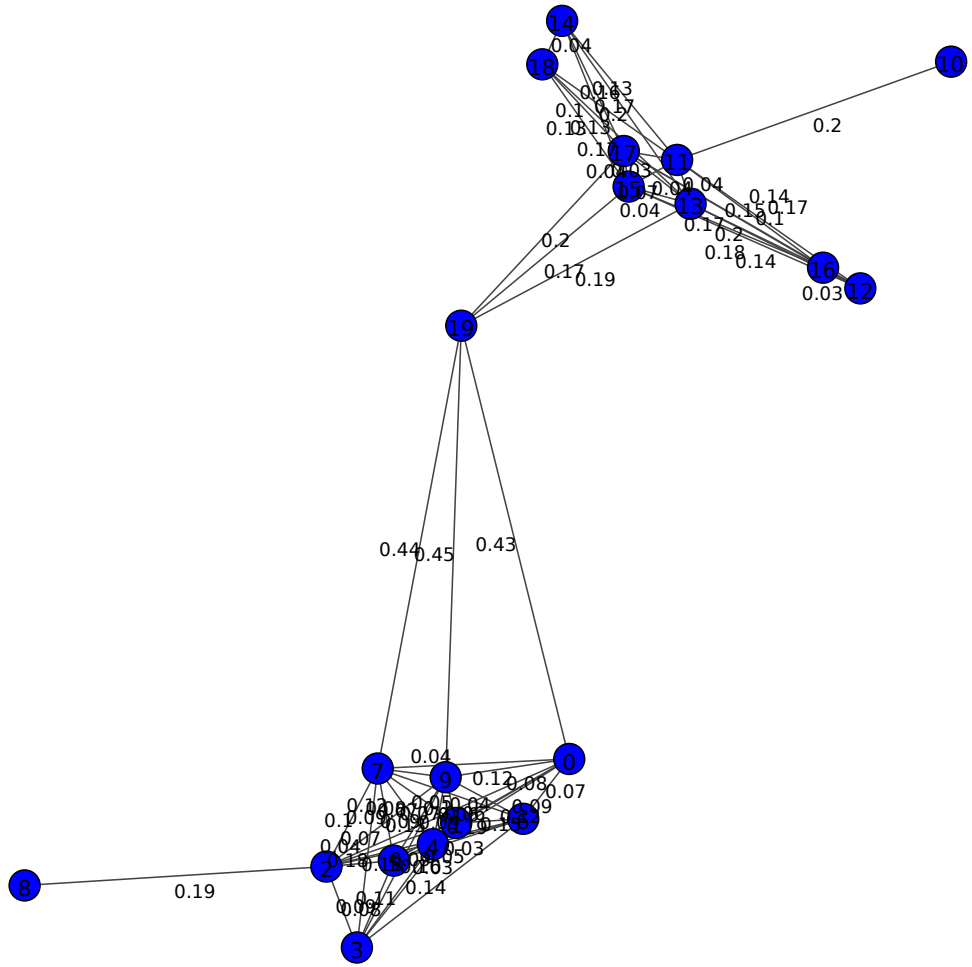
Optimality of MST-based clustering algorithms

Minimum path distance

Let be $\mathcal{G} = (V, E, w)$ and $u, v \in V$.

$$dist_{\mathcal{G}}(u, v) = \min_{\mathcal{P}_{u-v}} \sum_{e \in \mathcal{P}_{u-v}} w(e)$$

with $\mathcal{P}_{u-v}$ a path (edge version) from u to v in $\mathcal{G}$.



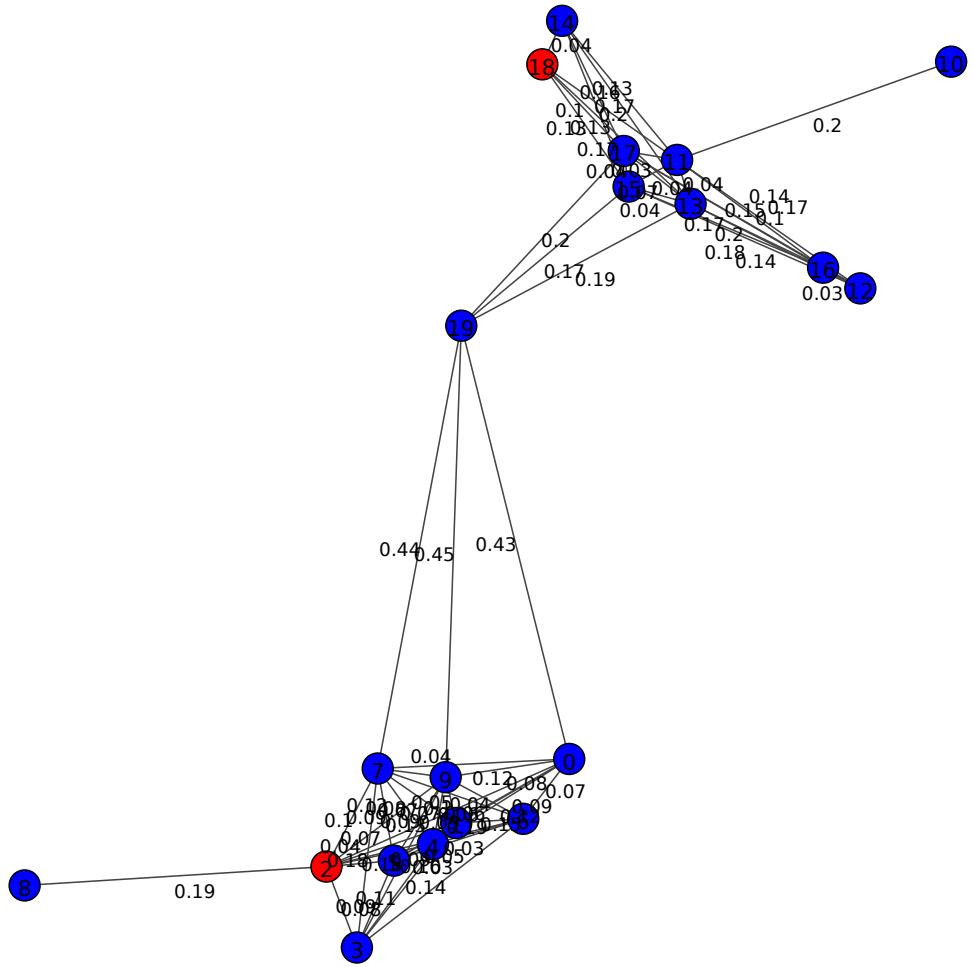
Optimality of MST-based clustering algorithms

Minimum path distance

Let be $\mathcal{G} = (V, E, w)$ and $u, v \in V$.

$$dist_{\mathcal{G}}(u, v) = \min_{\mathcal{P}_{u-v}} \sum_{e \in \mathcal{P}_{u-v}} w(e)$$

with $\mathcal{P}_{u-v}$ a path (edge version) from u to v in $\mathcal{G}$.



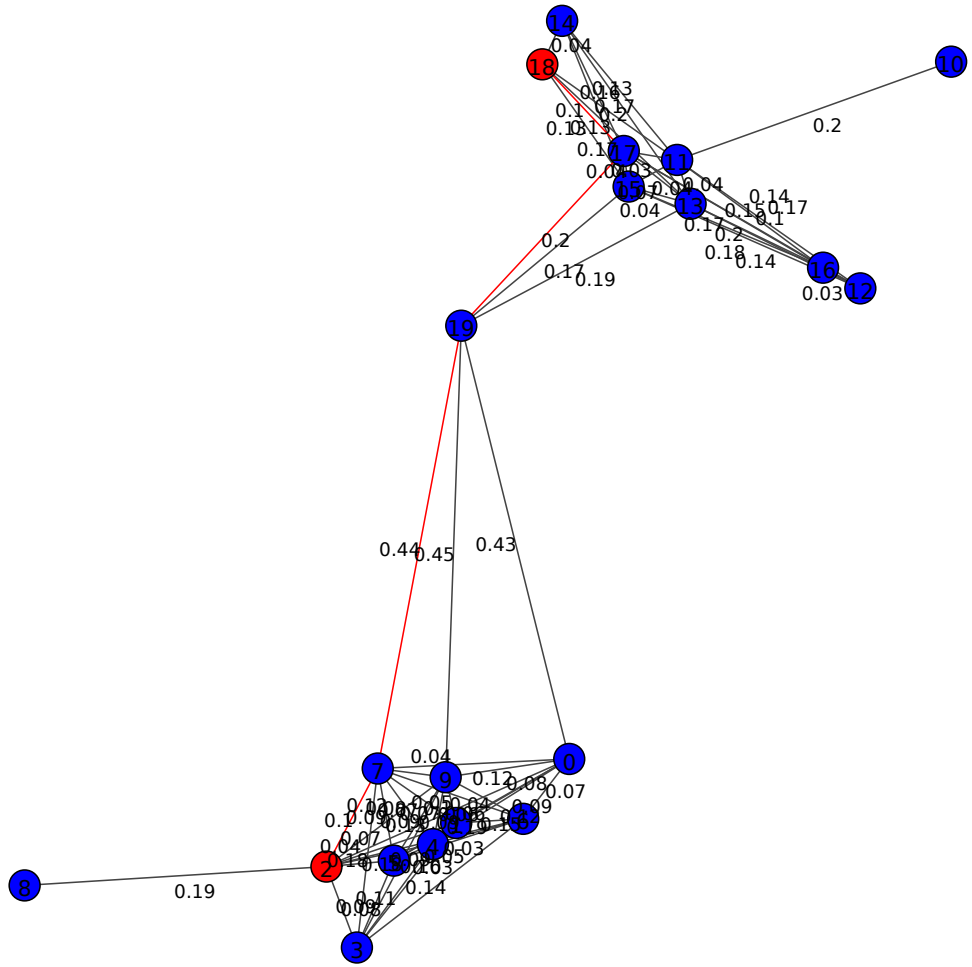
Optimality of MST-based clustering algorithms

Minimum path distance

Let be $\mathcal{G} = (V, E, w)$ and $u, v \in V$.

$$dist_{\mathcal{G}}(u, v) = \min_{\mathcal{P}_{u-v}} \sum_{e \in \mathcal{P}_{u-v}} w(e)$$

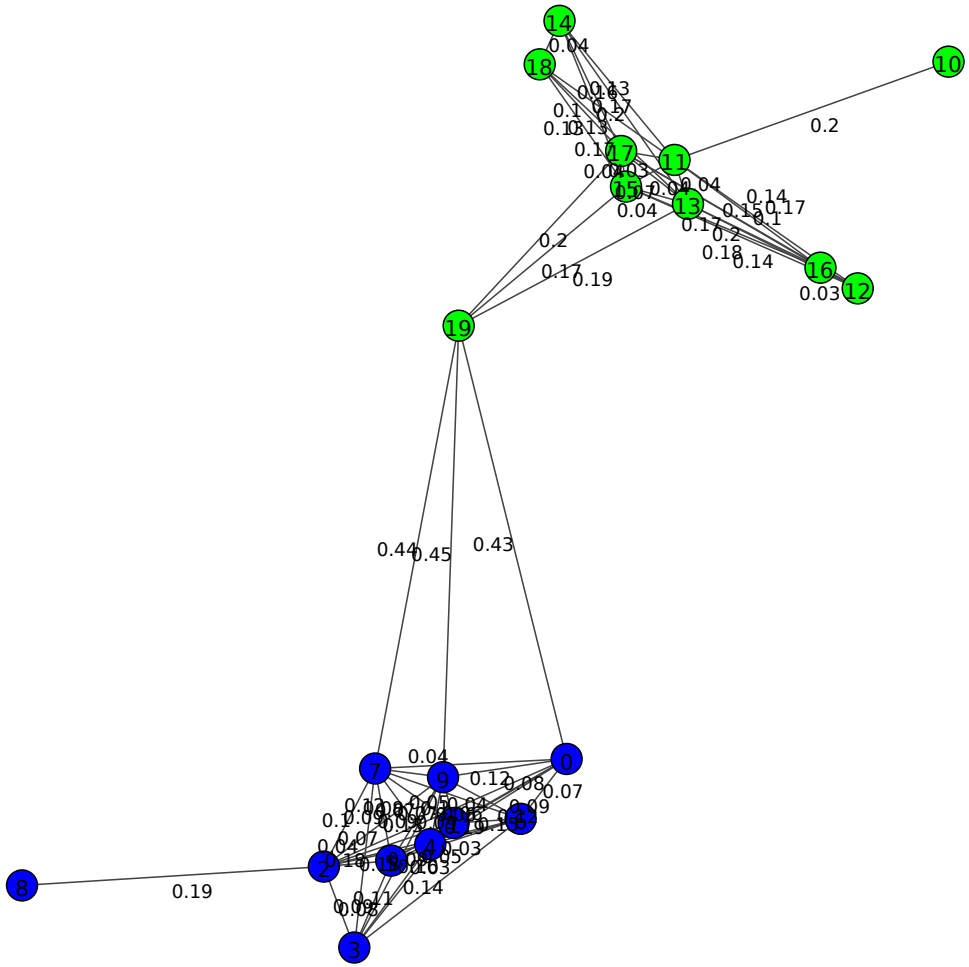
with $\mathcal{P}_{u-v}$ a path (edge version) from u to v in $\mathcal{G}$.



Optimality of MST-based clustering algorithms

Cluster Let be $\mathcal{G} = (V, E, w)$ a graph, $w := E \to (0, 1]$, $(V, dist_{\mathcal{G}})$ a metric space based on $dist_{\mathcal{G}}$ defined on $\mathcal{G}$ and $D \subset V$ a node set. $C \subset D$ is a cluster if and only if $|C| > 2$ and $\forall C_1, C_2$ s.t. $C = C_1 \cup C_2$ and $C_1 \cap C_2 = \emptyset$, one has:

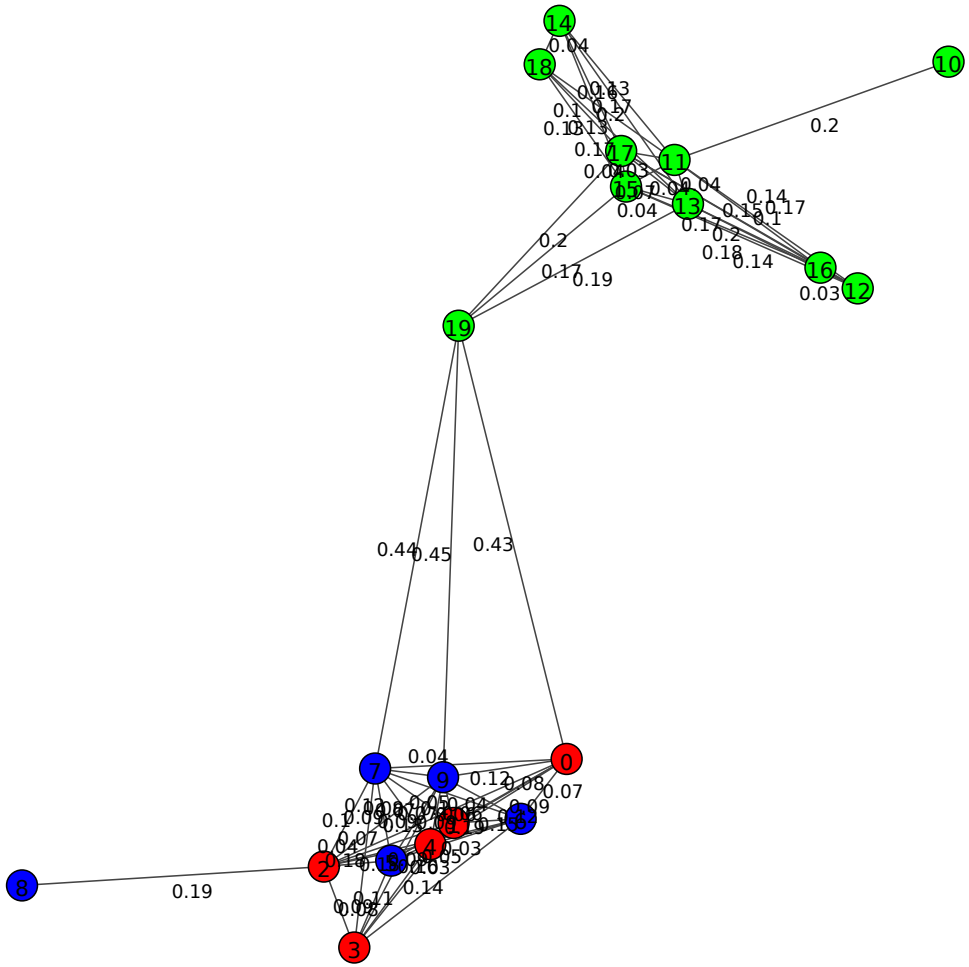
$$\underset{z \in D \setminus C_1}{\operatorname{argmin}} \{ \min_{v \in C_1} dist_{\mathcal{G}}(z, v) \} \subset C_2$$



Optimality of MST-based clustering algorithms

Cluster Let be $\mathcal{G} = (V, E, w)$ a graph, $w := E \to (0, 1]$, $(V, dist_{\mathcal{G}})$ a metric space based on $dist_{\mathcal{G}}$ defined on $\mathcal{G}$ and $D \subset V$ a node set. $C \subset D$ is a cluster if and only if $|C| > 2$ and $\forall C_1, C_2$ s.t. $C = C_1 \cup C_2$ and $C_1 \cap C_2 = \emptyset$, one has:

$$\underset{z \in D \setminus C_1}{\operatorname{argmin}} \{ \min_{v \in C_1} dist_{\mathcal{G}}(z, v) \} \subset C_2$$



Optimality of MST-based clustering algorithms

Theorem

Let be $\mathcal{G} = (V, E, w)$ a graph and $\mathcal{T}$ a minimum spanning tree of $\mathcal{G}$. Let also be C a Cluster and two vertices $v_1, v_2 \in C$. Then, $V_{\mathcal{P}_{v_1-v_2}} \subset C$ with $\mathcal{P}_{v_1-v_2}$ a path from v_1 to v_2 in $\mathcal{G}$, and $V_{\mathcal{P}_{v_1-v_2}}$ the set of vertices contained in $\mathcal{P}_{v_1-v_2}$.

Interpretation

- Given a graph $\mathcal{G}$, an MST $\mathcal{T}$, and any two nodes of C, every node in the path between them is in C.
- A cluster can be characterized by a subtree of $\mathcal{T}$.
- It justifies the use of all MST-based methods for data clustering or node clustering in a graph.

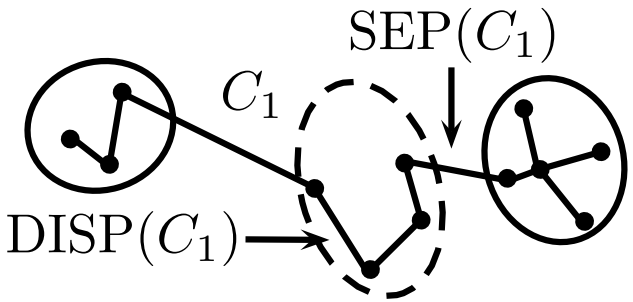
Cluster Dispersion and Separation

Cluster Dispersion

$$\forall i \in [K],\ \mathrm{DISP}(C_i) = \begin{cases} \max\limits_{j,\ e_j \in C_i} w_j & \text{if } |E(C_i)| \neq 0 \\ 0 & \text{otherwise.} \end{cases}$$

Cluster Separation

$$\forall i \in [K],\ \mathrm{SEP}(C_i) = \begin{cases} \min\limits_{j,\ e_j \in Cuts(C_i)} w_j & \text{if } K \neq 1 \\ 1 & \text{otherwise.} \end{cases}$$



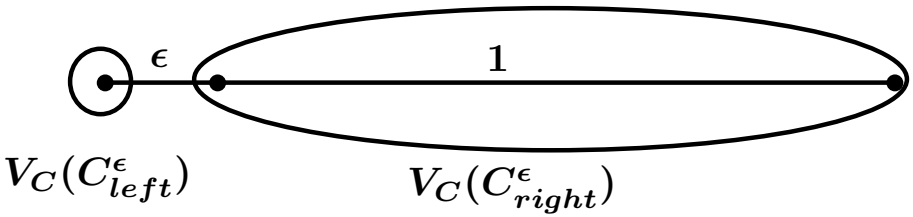
Validity Index of a Cluster and of a Clustering Partition

Validity Index of a Cluster

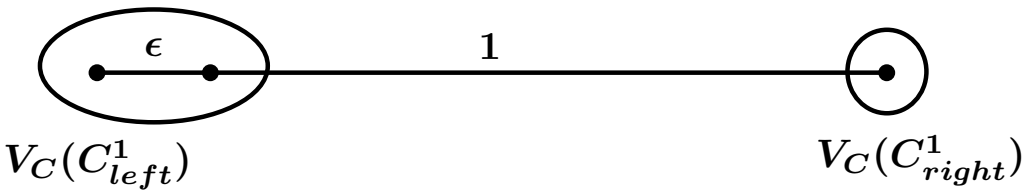
$$V_C(C_i) = \frac{\mathrm{SEP}(C_i) - \mathrm{DISP}(C_i)}{\max(\mathrm{SEP}(C_i), \mathrm{DISP}(C_i))} \in [-1, 1]$$

Validity Index of a Clustering partition

$$\mathrm{DBCVI}(\Pi) = \sum_{i=1}^{K} \frac{|C_i|}{N} V_C(C_i) \in [-1, 1]$$

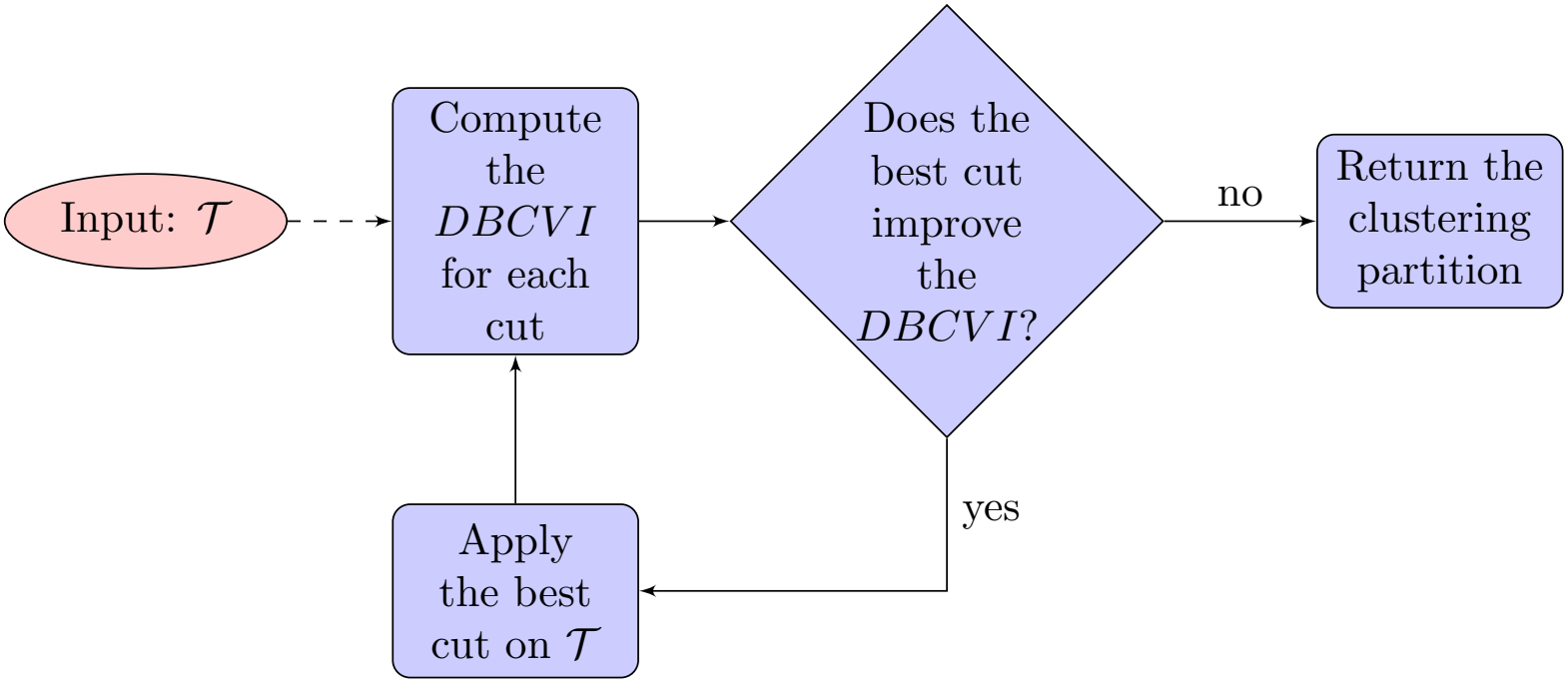


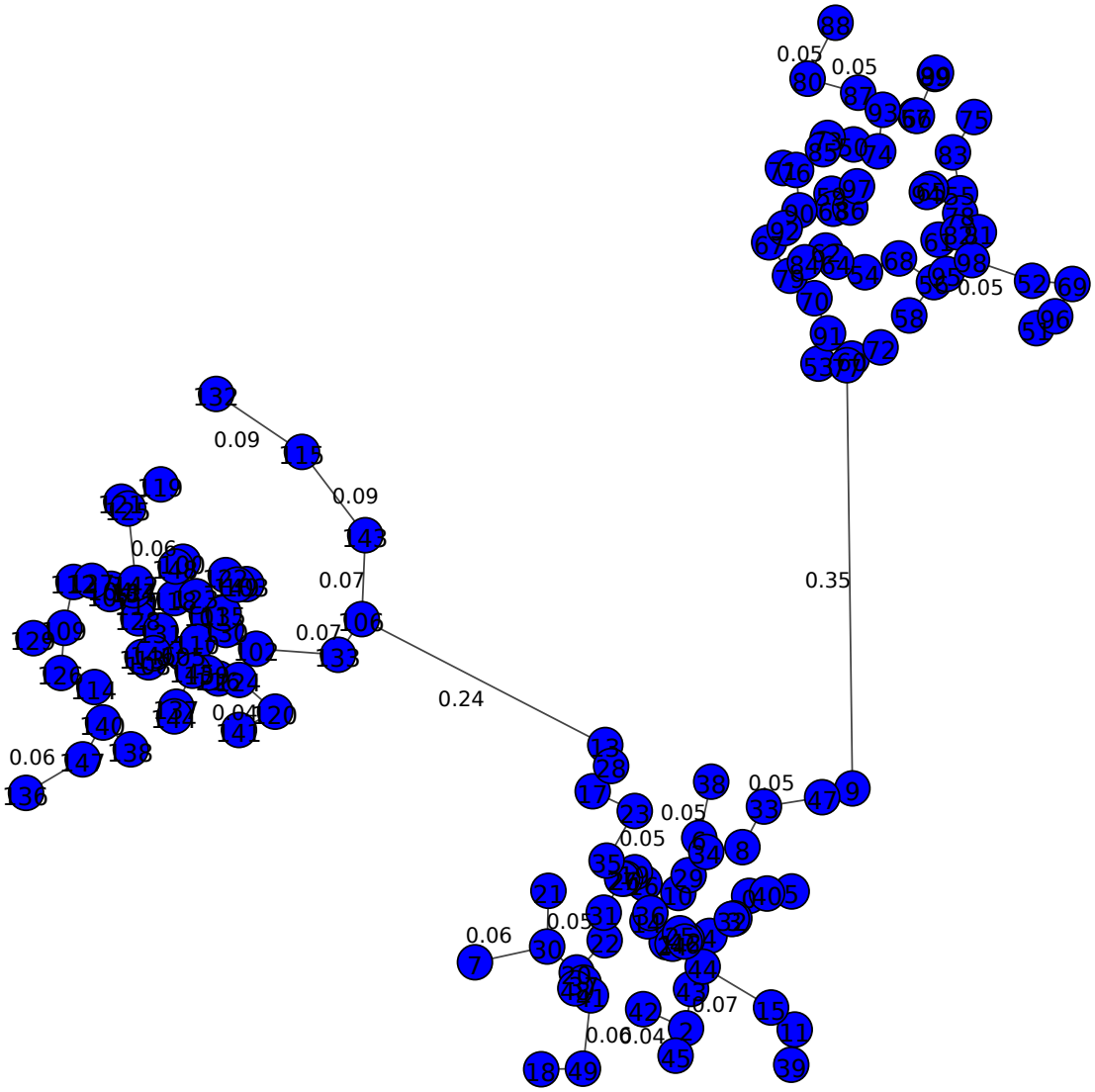
$$V_C(C^{\epsilon}_{left}) = 1$$
$$V_C(C^{\epsilon}_{right}) = \epsilon - 1 < 0$$

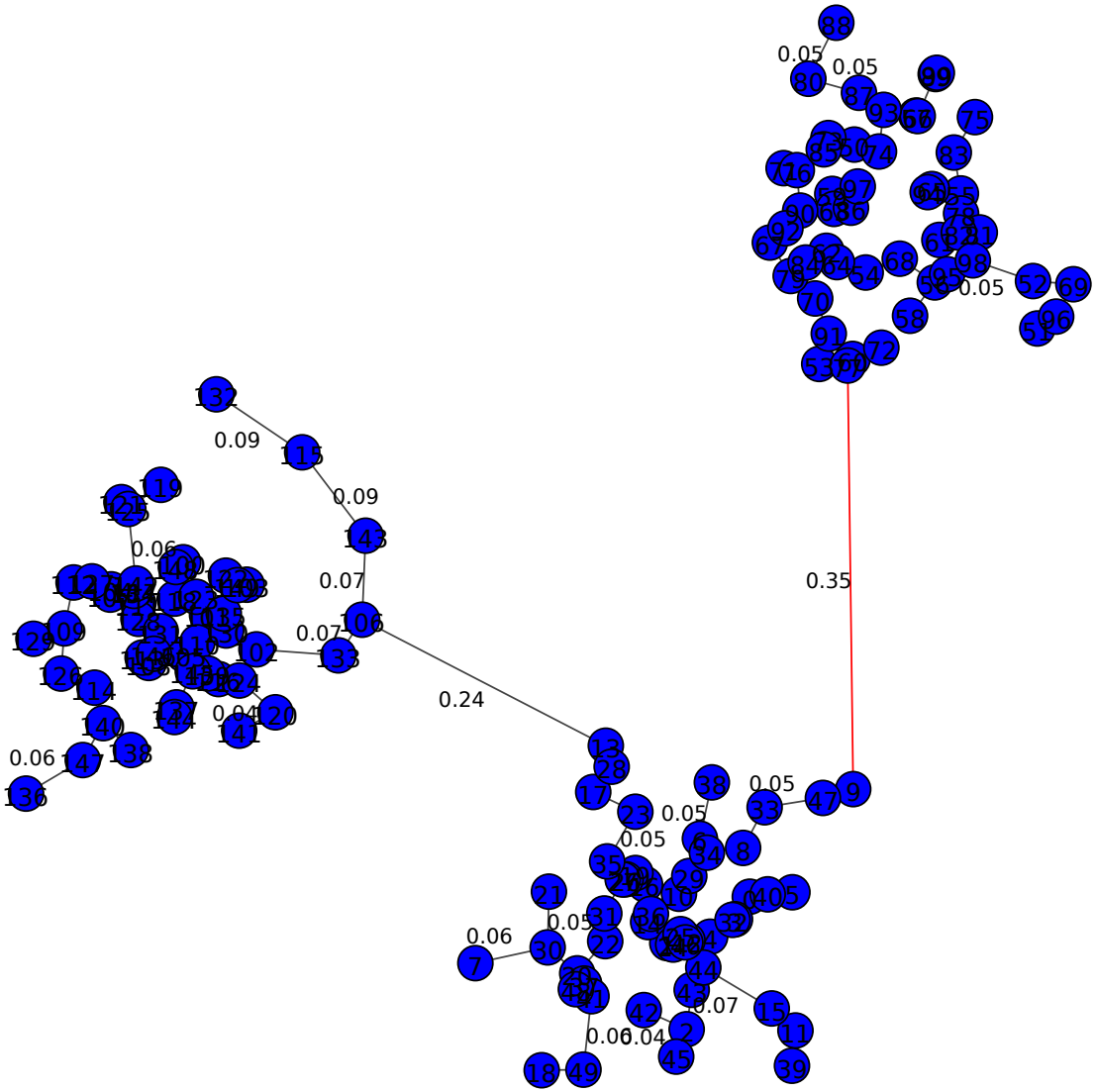


$$V_C(C^{1}_{left}) = 1 - \epsilon > 0$$
$$V_C(C^{1}_{right}) = 1$$

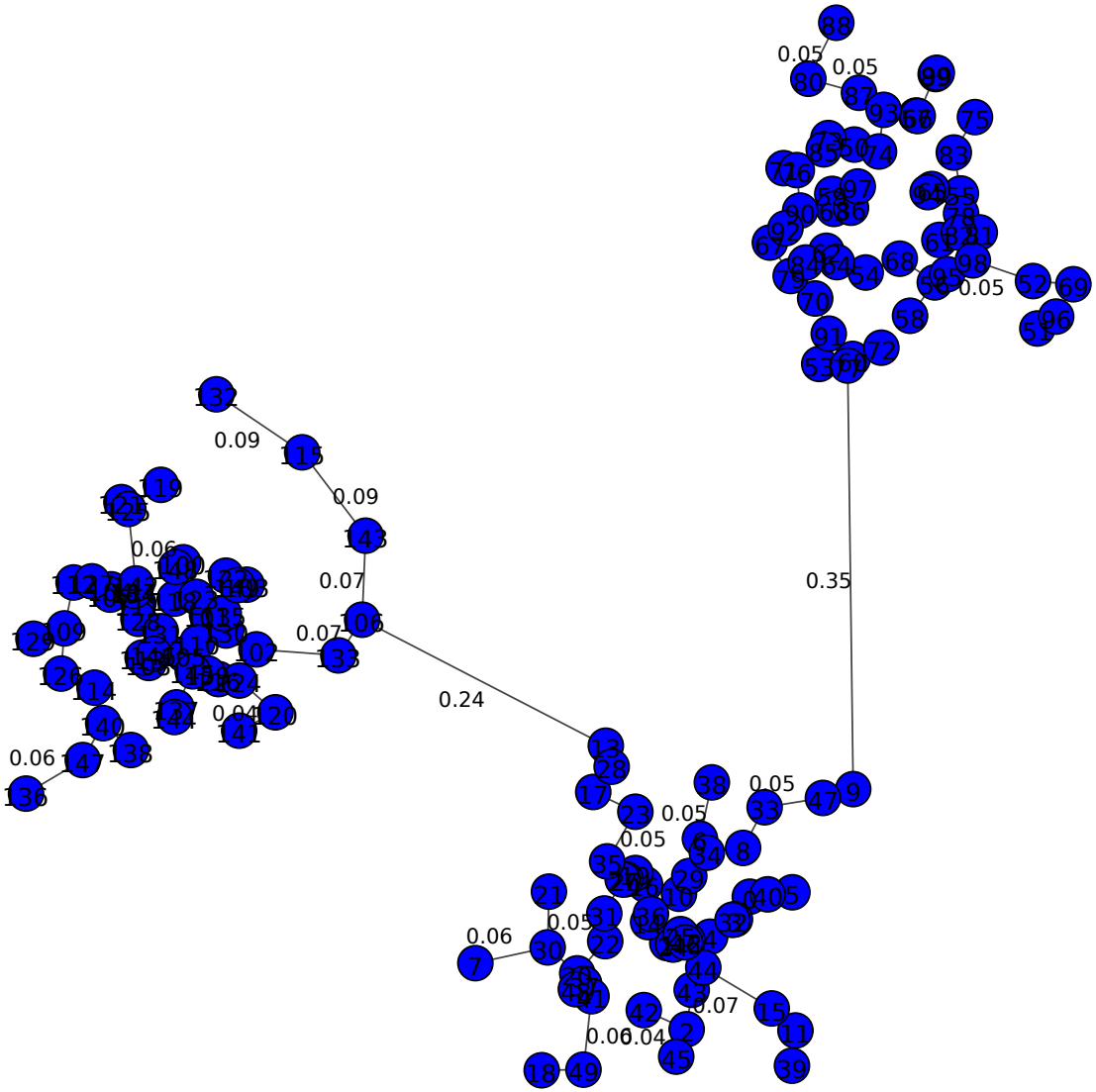
Algorithm DBMSTClu($\mathcal{T}$)



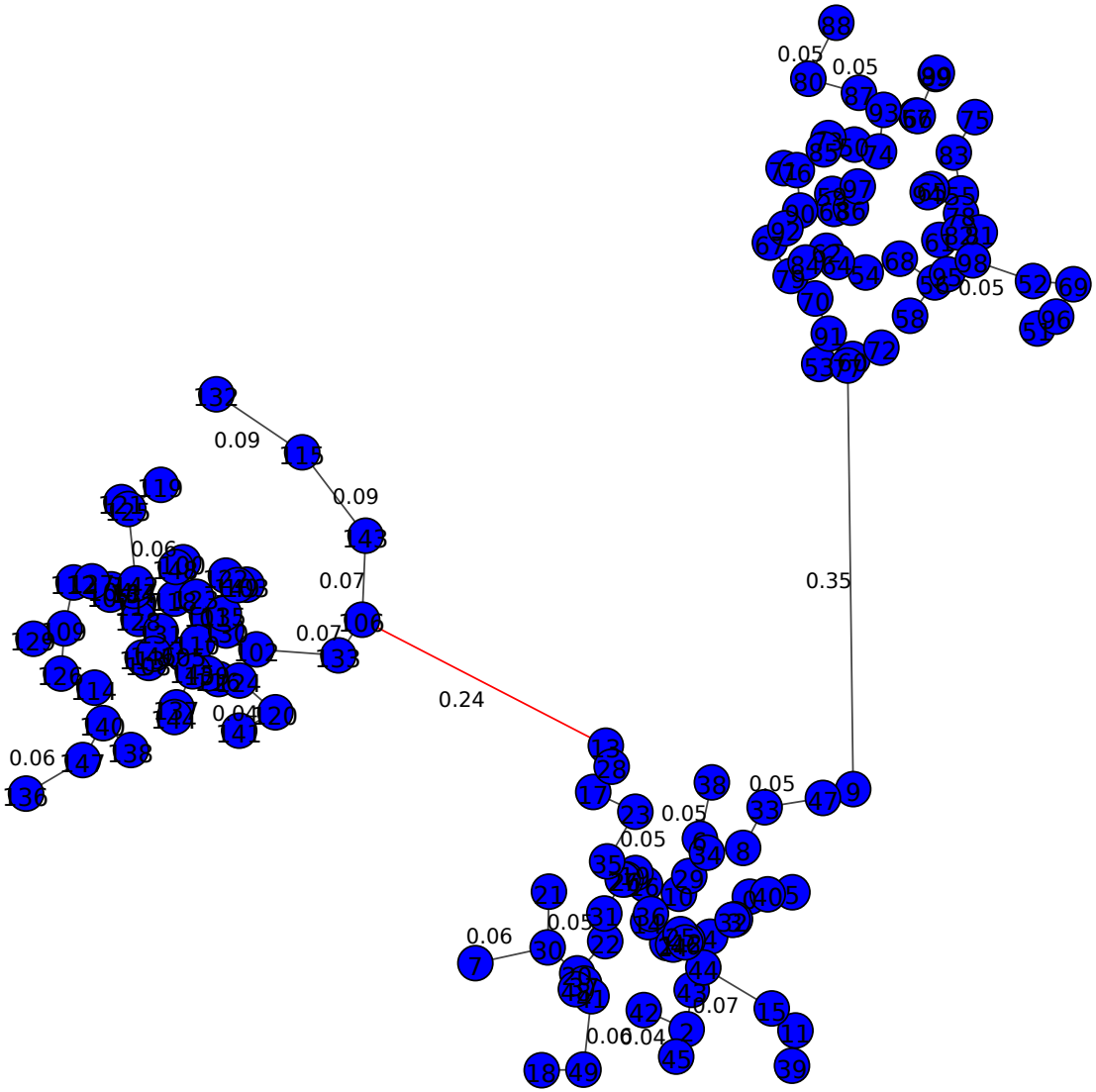

0.05
0.05
0.05
0.09
0.09
0.06
0.07
0.07
0.35
0.24
0.06
0.05
0.05
0.05
0.05
0.06
0.07
0.06
0.04



$$DBCVI(0.35) = 0.50$$

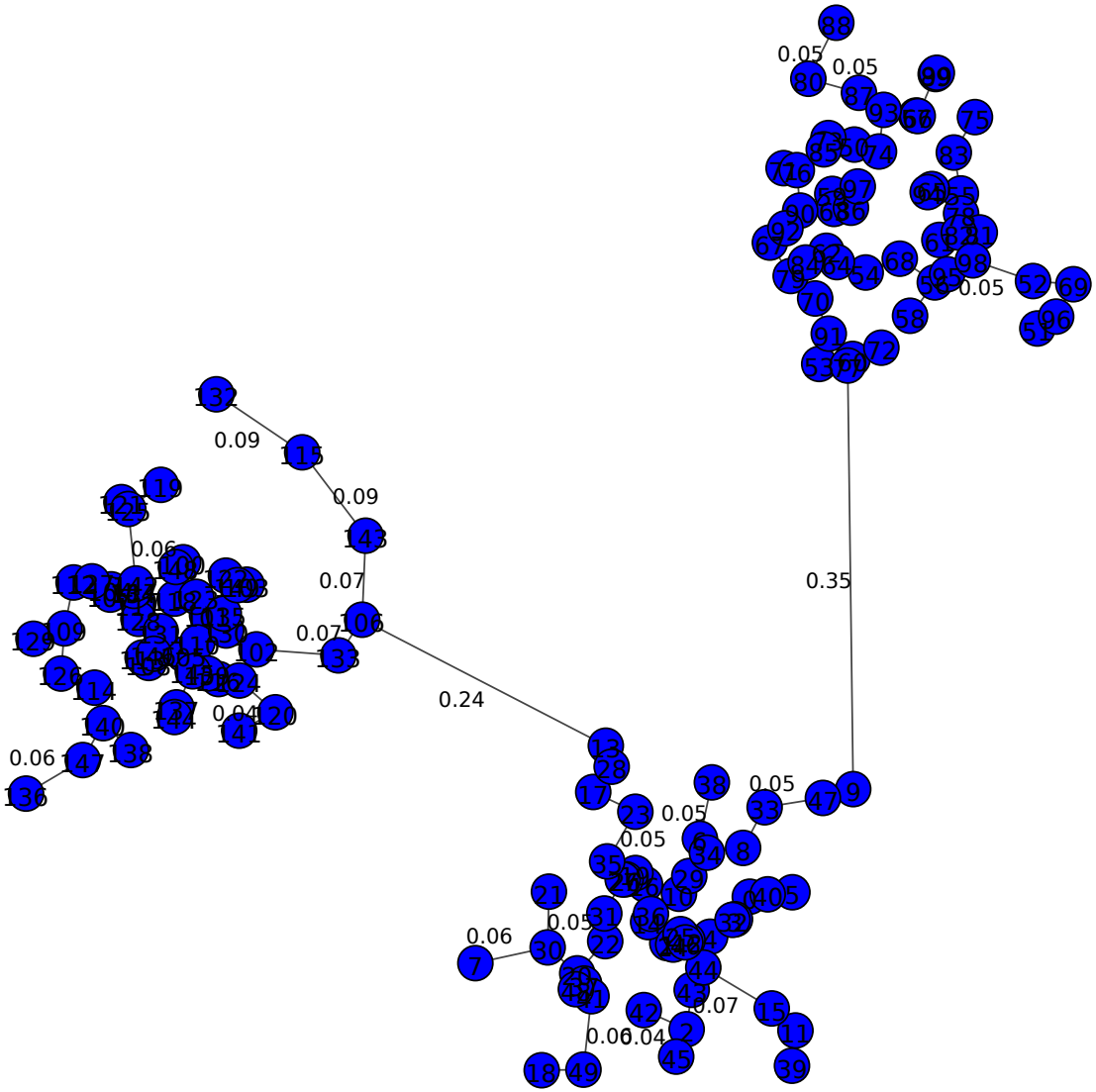


$$DBCVI(0.35) = 0.50$$

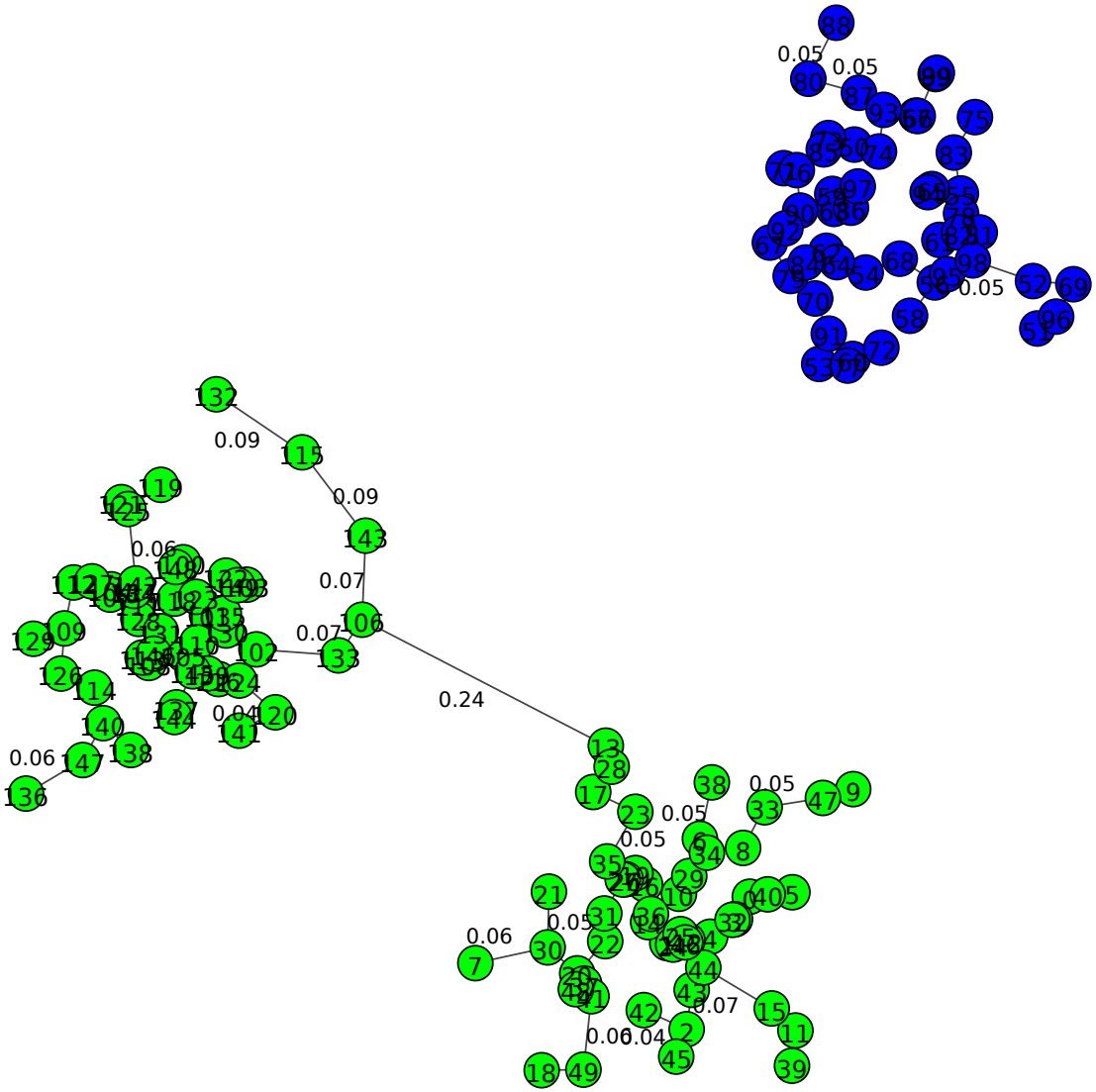


$$DBCVI(0.35) = 0.50$$

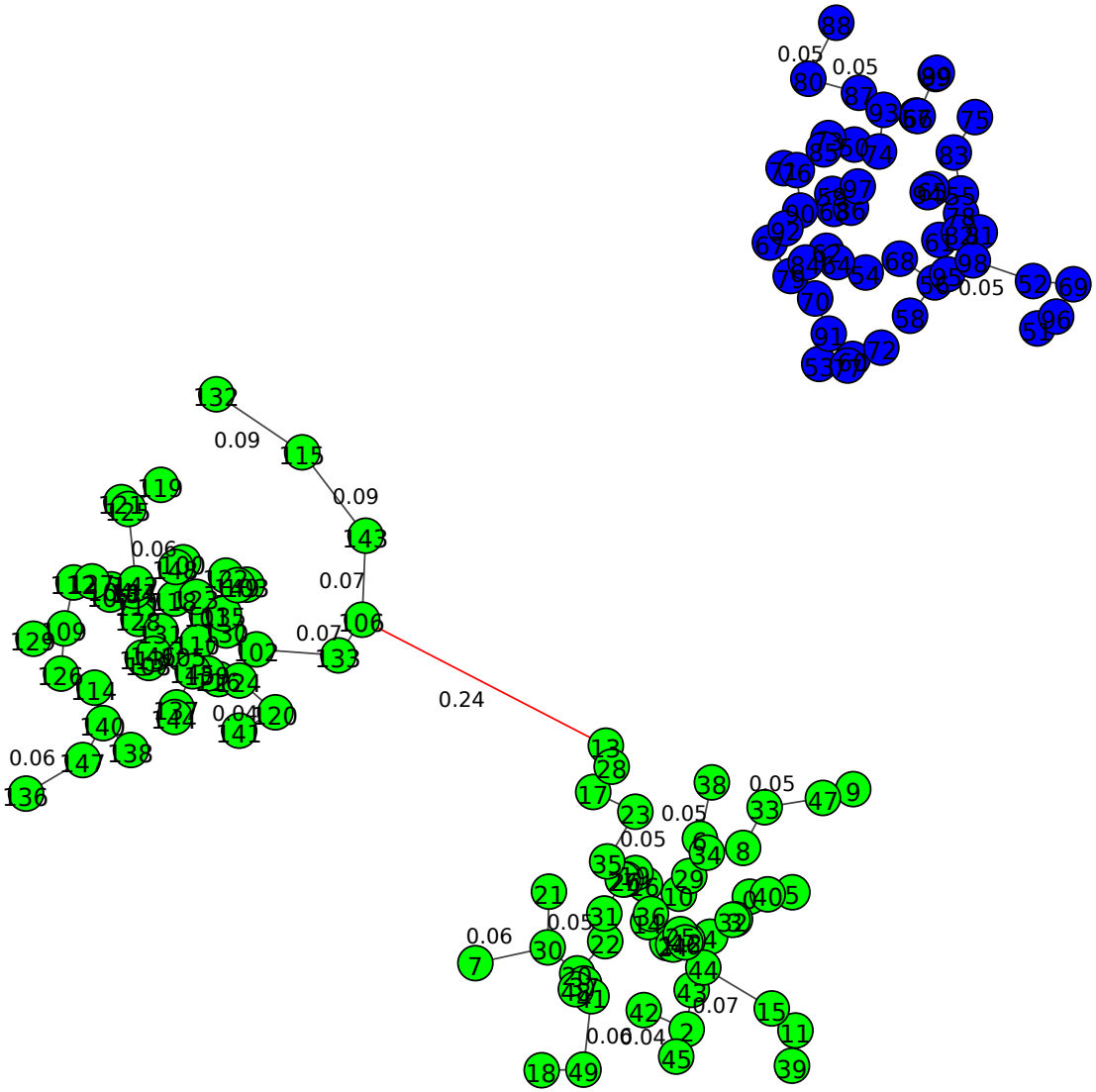
$$DBCVI(0.24) = -0.001$$



$$DBCVI(0.35) = 0.50$$

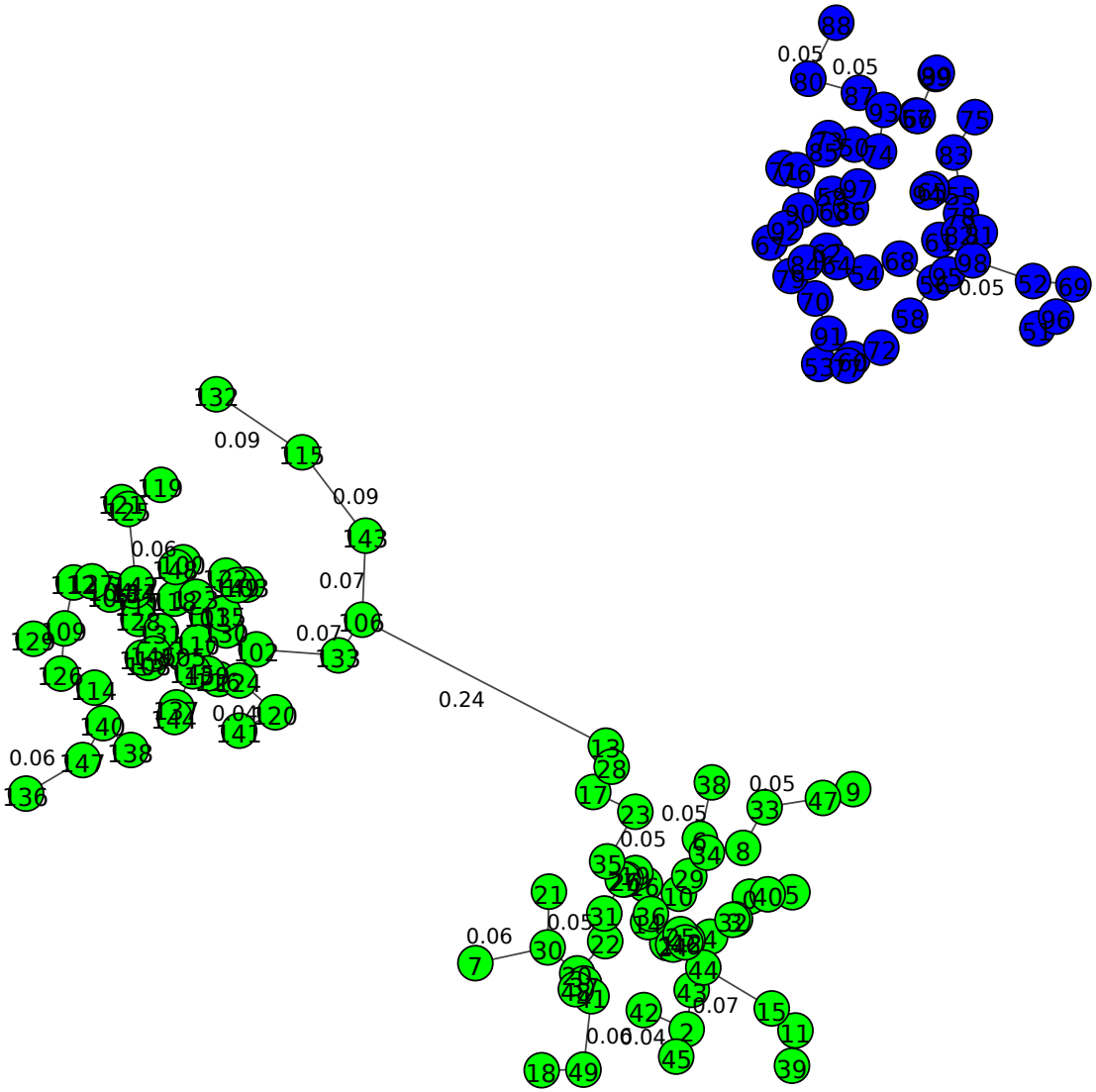

0.05
0.05
0.05
0.09
0.09
0.06
0.07
0.07
0.24
0.04
0.06
0.05
0.05
0.05
0.06
0.07
0.06
0.04

$$DBCVI = 0.50$$



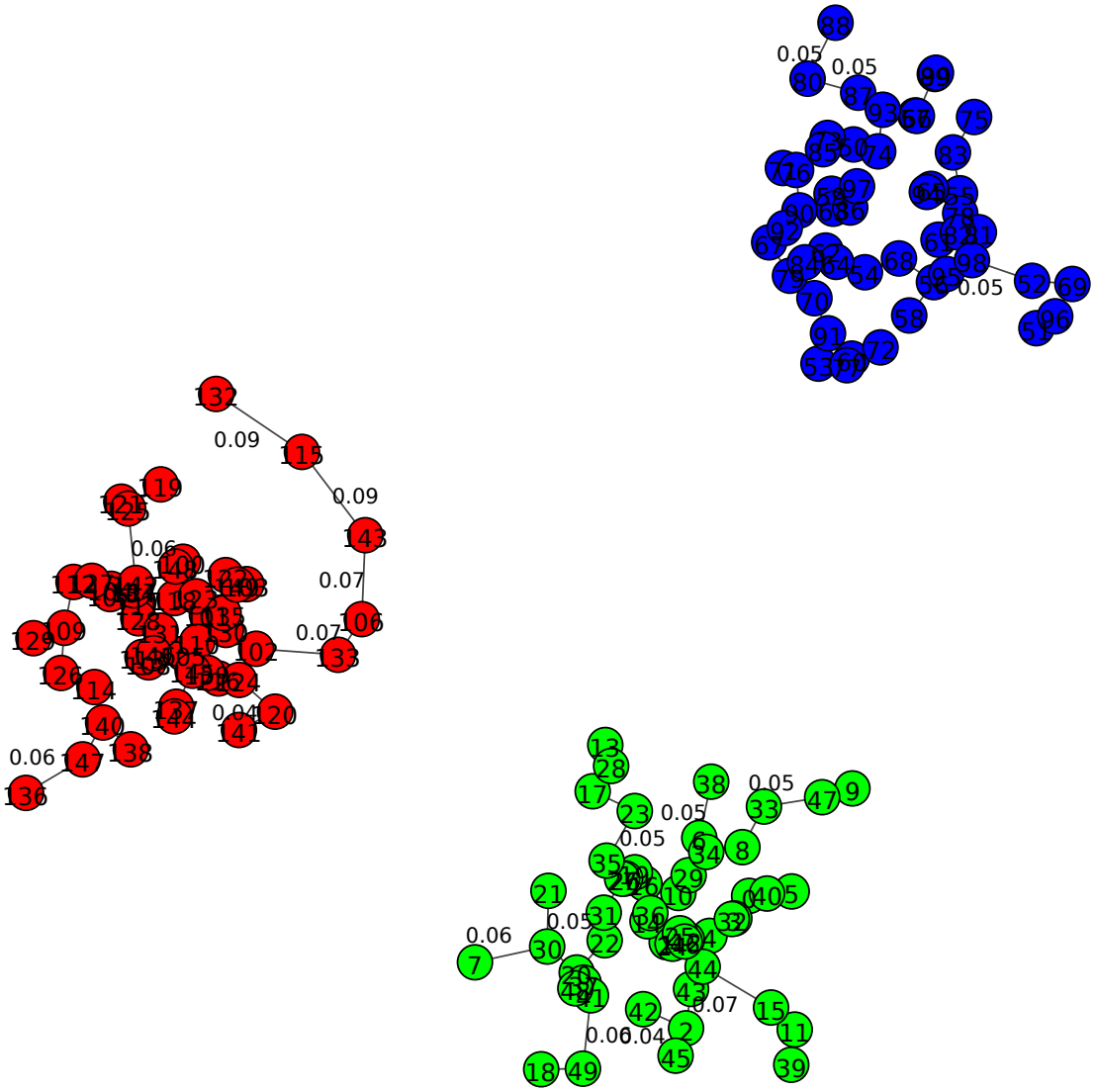
$$DBCVI = 0.50$$

$$DBCVI(0.24) = 0.73$$



$$DBCVI = 0.50$$

$$DBCVI(0.24) = 0.73$$

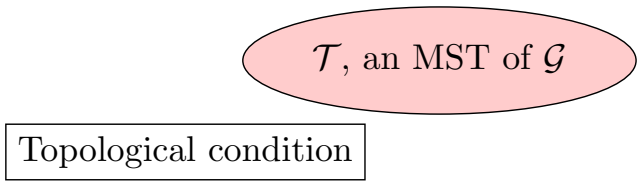


$$DBCVI = 0.73$$

Accuracy guarantees

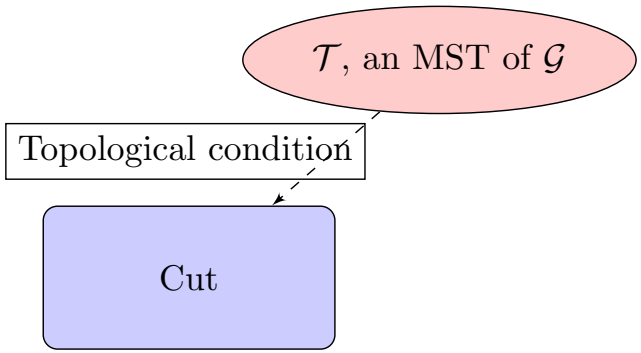
- Graph $\mathcal{G} = (V, E, w)$ with K clusters $(C_i^*)_{i \in [K]}$
- $\mathcal{T}$ an MST of $\mathcal{G}$

Accuracy guarantees



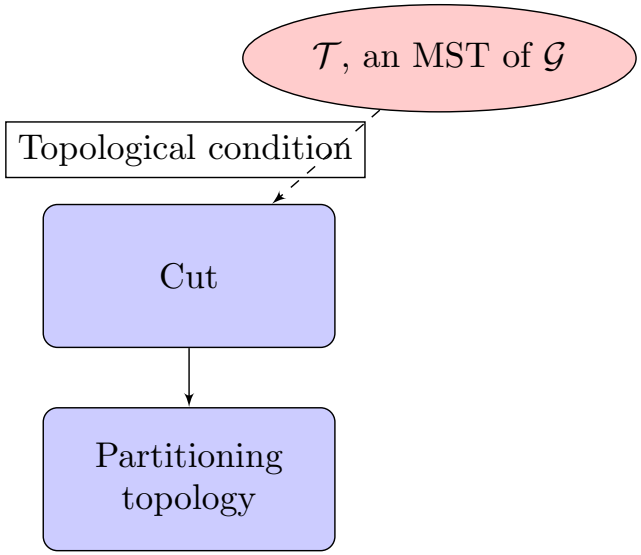
- Graph $\mathcal{G} = (V, E, w)$ with K clusters $(C_i^*)_{i \in [K]}$
- $\mathcal{T}$ an MST of $\mathcal{G}$

Accuracy guarantees



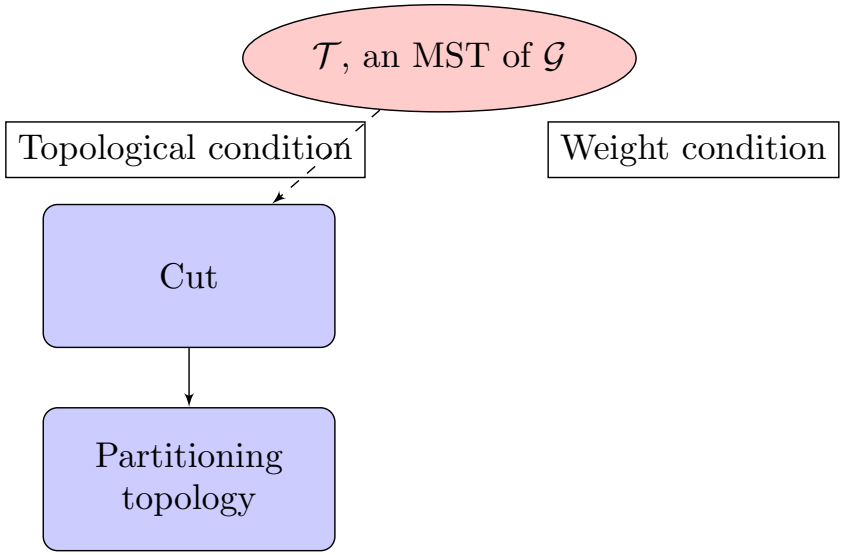
- $Cut_{\mathcal{G}}(\mathcal{T})$
  the set of effective cuts to perform on $\mathcal{T}$ in order to ensure the exact recovery of the clustering partition.
- $e^{(ij)} \in Cut_{\mathcal{G}}(\mathcal{T})$ the edge between cluster C_i^* and C_j^*.

Accuracy guarantees



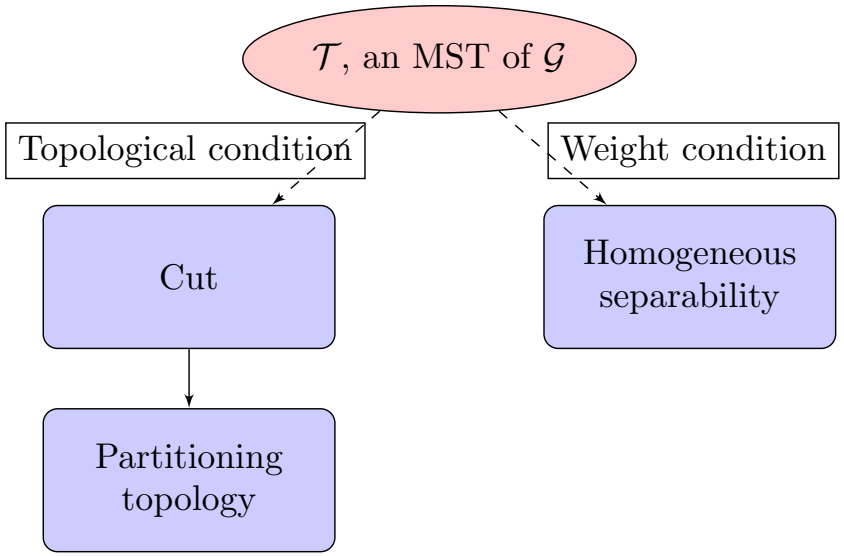
- Trees on which $Cut_{\mathcal{G}}(.)$ enables to find the right partition are said to be a **partitioning topology**.

Accuracy guarantees



- Trees on which $Cut_{\mathcal{G}}(.)$ enables to find the right partition are said to be a **partitioning topology**.

Accuracy guarantees



- $\mathcal{T}$ is **homogeneously separable** by s, if

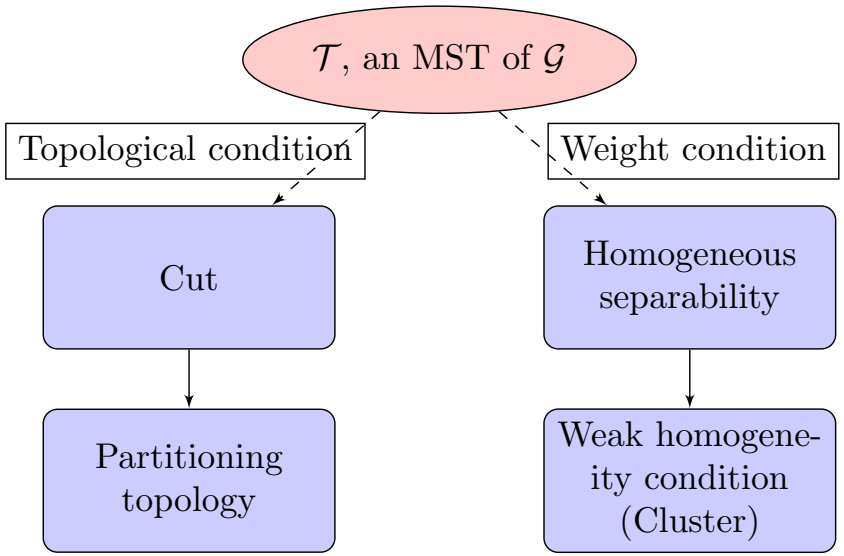
$$\alpha_{\mathcal{T}} \max_{e \in E(\mathcal{T})} w(e) < w(s)$$

with

$$\alpha_{\mathcal{T}} = \frac{\max_{e \in E(\mathcal{T})} w(e)}{\min_{e \in E(\mathcal{T})} w(e)} \geq 1.$$

- Notation: $H_{\mathcal{T}}(s)$ is verified.

Accuracy guarantees



- C_i^*, $i \in [K]$, is **weakly homogeneous** if:

  for all $\mathcal{T}$ an MST of $\mathcal{G}$, and $\forall j \in [K]$, $j \neq i$, s.t.

  $$e^{(ij)} \in Cut_{\mathcal{G}}(\mathcal{T}),$$

  $H_{\mathcal{T}_{|C_i^*}}(e^{(ij)})$ is verified.

Accuracy guarantees

Graph $\mathcal{G} = (V, E, w)$ with K homogeneous clusters $C_1^*, \ldots, C_K^*$ and $\mathcal{T}$ an MST of $\mathcal{G}$.

Theorem 1

Let assume that at step $k < K - 1$, DBMSTClu built $k + 1$ subtrees $\mathcal{C}_1, \ldots, \mathcal{C}_{k+1}$ by cutting $e_1,\ e_2,\ \ldots,\ e_k \in E$.
Then,
$Cut_k := Cut_{\mathcal{G}}(\mathcal{T}) \setminus \{e_1,\ e_2,\ \ldots,\ e_k\} \neq \emptyset \implies \mathrm{DBCVI}_{k+1} \geq DBCVI_k$,
i.e. if there are still edges in Cut_k, the algorithm will continue to perform some cut.

Accuracy guarantees

Graph $\mathcal{G} = (V, E, w)$ with K homogeneous clusters $C_1^*, \ldots, C_K^*$ and $\mathcal{T}$ an MST of $\mathcal{G}$.

Theorem 2

Assume that at step $k < K - 1$, DBMSTClu built $k + 1$ subtrees $\mathcal{C}_1, \ldots, \mathcal{C}_{k+1}$ by cutting $e_1,\ e_2,\ \ldots,\ e_k \in E$.
If $Cut_k \neq \emptyset$ then $\underset{e \in \mathcal{T} \setminus \{e_1,\ e_2,\ \ldots,\ e_k\}}{\operatorname{argmax}} DBCVI_{k+1}(e) \subset Cut_k$ i.e. the cut edge at step $k + 1$ is in Cut_k.

Accuracy guarantees

Graph $\mathcal{G} = (V, E, w)$ with K weakly homogeneous clusters $C_1^*, \ldots, C_K^*$ and $\mathcal{T}$ an MST of $\mathcal{G}$.

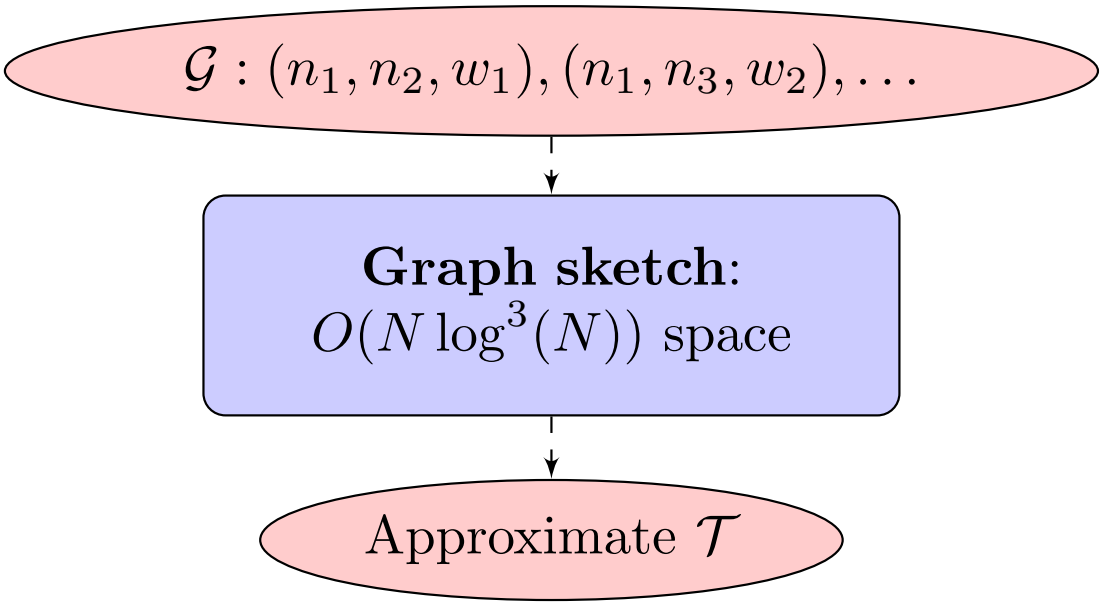
Theorem 3

Let assume that at step $K-1$, DBMSTClu built K subtrees $\mathcal{C}_1, \ldots, \mathcal{C}_K$ by cutting $e_1,\ e_2,\ \ldots,\ e_{K-1} \in E$.
Then, for all $e \in \mathcal{T} \backslash \{e_1,\ e_2,\ \ldots,\ e_{K-1}\}$, $DBCVI_K(e) < DBCVI_{K-1}$
i.e. the algorithm stops: no edge gets cut during step K.

Corollary

$DBMSTClu(\mathcal{T})$ stops after $K-1$ iterations and the K subtrees produced match exactly the clusters i.e. under homogeneity condition, the algorithm finds automatically the underlying clustering partition.

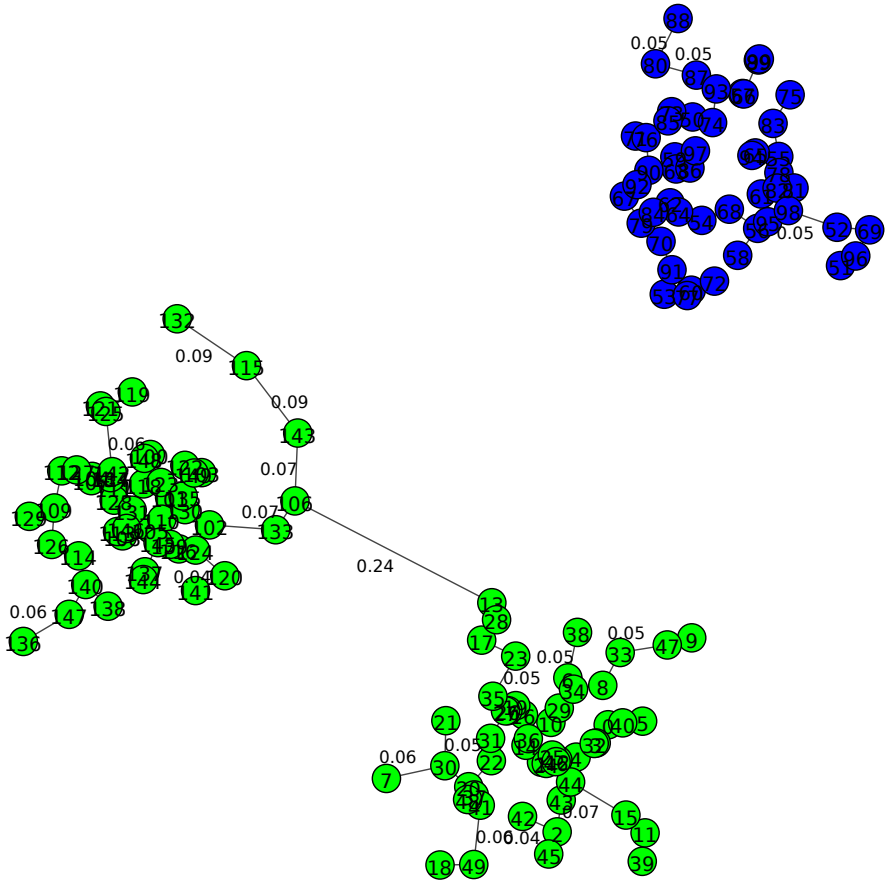
Scalability for computing an MST



Graph sketching [Ahn et al., 2012]

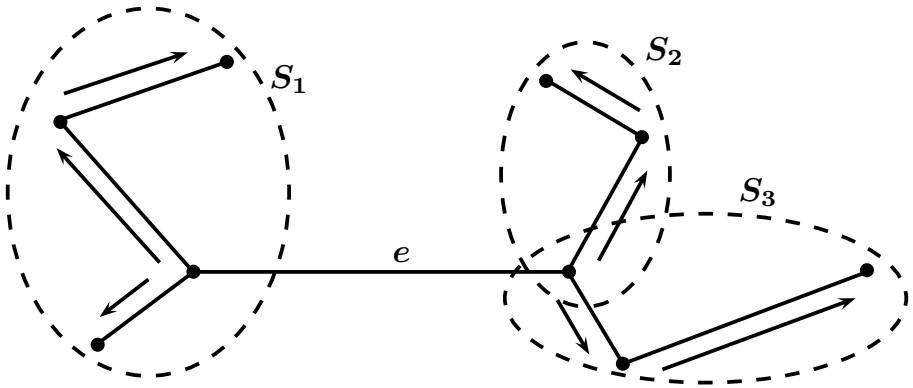
Scalability of the clustering algorithm

- A cut in cluster C_i lets $V_C(C_j),\ \forall j \neq i$ unchanged.

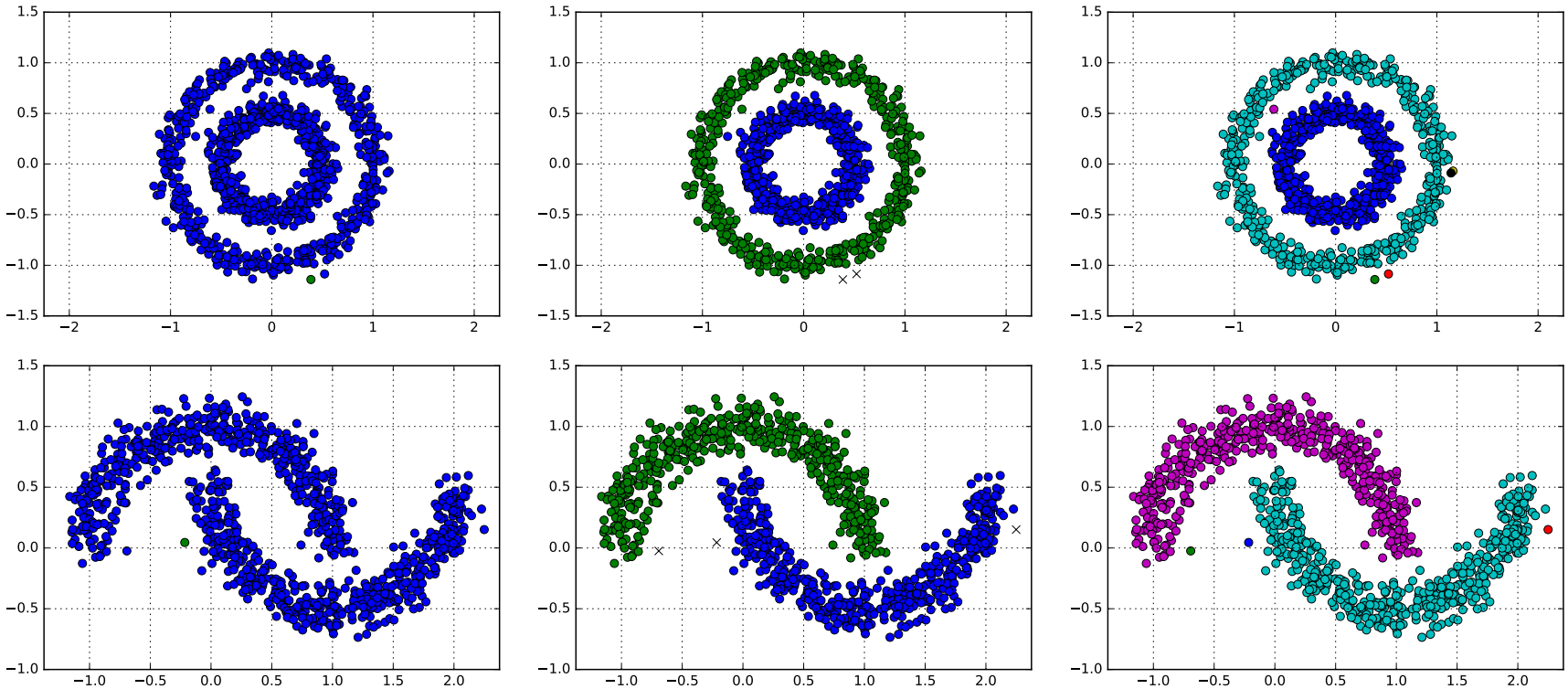


Scalability of the clustering algorithm

- Recurrence relationship of SEP and DISP in $\mathcal{T}$. Iterative version of the Depth-First Search to determine DBCVI for each cut left and right: Double Depth-First Search.



Safety of the sketching

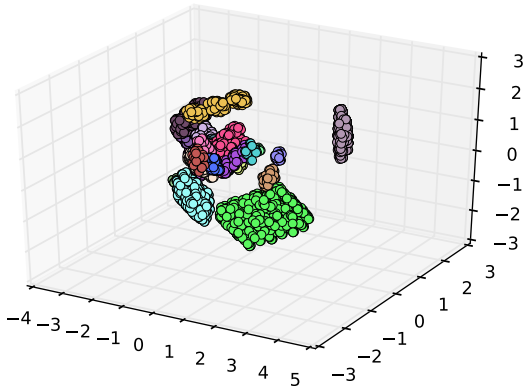


SEMST DBSCAN DBMSTClu

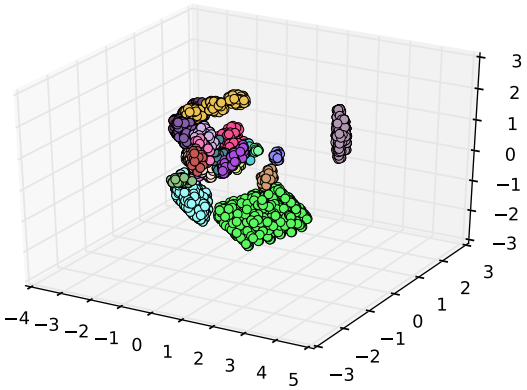
	Silhouette coeff.		ARI		DBCVI	
SEMST	**0.16**	−0.12	0	0	0.001	0.06
DBSCAN	0.02	**0.26**	**0.99**	**0.99**	−0.26	**0.15**
DBMSTClu	−0.26	**0.26**	**0.99**	**0.99**	**0.18**	**0.15**

Scalability of the clustering

Mushroom dataset (8124 nodes), time to recover 23 clusters:



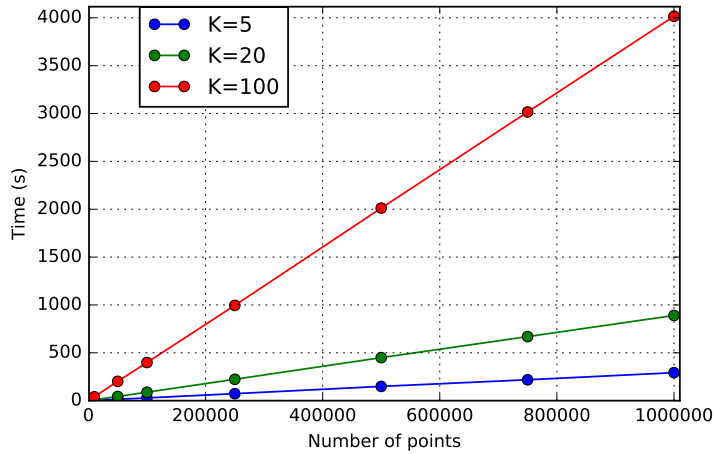
DBSCAN: 9s



DBMSTClu: 3.36s

In the Stochastic Block Model, time (s) to recover the K clusters w.r.t N:

$K \backslash N$	1000	10000	50000	100000	250000	500000	750000	1000000
5	0.34	2.96	14.37	28.91	73.04	148.85	218.11	292.25
20	0.95	8.73	43.71	88.51	223.18	449.37	669.29	889.88
100	4.36	40.25	201.76	398.41	995.42	2011.79	3015.61	4016.13
"100/5"	12.82	13.60	14.04	13.78	13.63	13.52	13.83	13.74



Conclusion for DBMSTClu

Take-home message: DBMSTClu is an ...

- MST-based
- parameter-free
- space-efficient clustering algorithm
- for arbitrarily-shaped clusters

`https://github.com/annemorvan/DBMSTClu`

What is also in the thesis...

- A Differentially Private clustering algorithm based on a private release of an MST

Plan

Conclusion and perspectives

Main results

- Theoretical guarantees for $\mathbf{HD_3HD_2HD_1}$
- UnifDiag: New online hypercubic quantization-based hashing technique
  - Theoretical guarantees on optimality of the rotation
- Proof of optimality of MST-based clustering
- DBMSTClu: New non-parametric space-efficient MST-based clustering algorithm
  - Results on the clustering partition accuracy

Conclusion and perspectives

Perspectives

- UnifDiag
  - How to find the optimal rotation among the ones uniformizing the diagonal of the PCA-projected covariance matrix?
  - Linear vs nonlinear embeddings (kernel methods)
  - Regret-type results by changing the online PCA estimation
- DBMSTClu
  - Full streaming MST computation + DBMSTClu

Publications I

- Mariusz Bojarski, Anna Choromanska, Krzysztof Choromanski, Francois Fagan, Cédric Gouy-Pailler, Anne Morvan, Nouri Sakr, Tamás Sarlós, Jamal Atif, *Structured adaptive and random spinners for fast machine learning computations*, Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS'17), 54, pp.1020-1029, Fort Lauderdale, FL, USA, 20-22 Apr.
- Anne Morvan, Antoine Souloumiac, Cédric Gouy-Pailler, Jamal Atif, *Streaming Binary Sketching based on Subspace Tracking and Diagonal Uniformization*, Proceedings of the 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2018), Calgary, Alberta, Canada, 15-20 Apr.

Publications II

- Anne Morvan, Krzysztof Choromanski, Cédric Gouy-Pailler, Jamal Atif, *Graph sketching-based Space-efficient Data Clustering*, Proceedings of the SIAM International Conference on DATA MINING (SDM'18), pp.10-18, San Diego, CA, USA, 3-4 May.
- Rafaël Pinot, Anne Morvan, Florian Yger, Cédric Gouy-Pailler, Jamal Atif, *Graph-based Clustering under Differential Privacy*, Proceedings of the International Conference on Uncertainty in Artificial Intelligence (UAI 2018), Monterey, CA, USA, 6-10 Aug.
- Anne Morvan, Antoine Souloumiac, Krzysztof Choromanski, Cédric Gouy-Pailler, Jamal Atif, *On the Needs for Rotations in Hypercubic Quantization Hashing*, Journal paper in preparation.

References I

Abed-Meraim, K., Chkeif, A., and Hua, Y. (2000).
Fast Orthonormal PAST Algorithm.
IEEE Signal Processing Letters, (3):60 – 62.

Aggarwal, C. C., Han, J., Wang, J., and Yu, P. S. (2003).
A Framework for Clustering Evolving Data Streams.
In Proceedings of the 29th International Conference on Very Large Data Bases - Volume 29, VLDB '03, pages 81–92. VLDB Endowment.

Ahn, K. J., Guha, S., and McGregor, A. (2012).
Analyzing Graph Structure via Linear Measurements.
In Proceedings of the Twenty-third Annual ACM-SIAM Symposium on Discrete Algorithms, SODA '12, pages 459–467, Philadelphia, PA, USA. Society for Industrial and Applied Mathematics.

Ailon, N. and Chazelle, B. (2006).
Approximate nearest neighbors and the fast Johnson-Lindenstrauss transform.
In Proceedings of the 38th STOC, pages 557–563. ACM.

Ailon, N., Jaiswal, R., and Monteleoni, C. (2009).
Streaming k-means approximation.
In Bengio, Y., Schuurmans, D., Lafferty, J. D., Williams, C. K. I., and Culotta, A., editors, Advances in Neural Information Processing Systems 22, pages 10–18. Curran Associates, Inc.

Andoni, A., Indyk, P., Laarhoven, T., Razenshteyn, I., and Schmidt, L. (2015).
Practical and Optimal LSH for Angular Distance.
In Proceedings of the 28th International Conference on Neural Information Processing Systems, NIPS'15, pages 1225–1233, Cambridge, MA, USA. MIT Press.

References II

Asano, T., Bhattacharya, B., Keil, M., and Yao, F. (1988).
Clustering Algorithms Based on Minimum and Maximum Spanning Trees.
In Proceedings of the Fourth Annual Symposium on Computational Geometry, SCG '88, pages 252–257, New York, NY, USA. ACM.

Cao, F., Ester, M., Qian, W., and Zhou, A. (2006).
Density-based clustering over an evolving data stream with noise.
In In 2006 SIAM Conference on Data Mining, pages 328–339.

Chen, Y., Jalali, A., Sanghavi, S., and Xu, H. (2014a).
Clustering Partially Observed Graphs via Convex Optimization.
J. Mach. Learn. Res., 15(1):2213–2238.

Chen, Y., Lim, S. H., and Xu, H. (2014b).
Weighted Graph Clustering with Non-uniform Uncertainties.
In Proceedings of the 31st International Conference on International Conference on Machine Learning - Volume 32, ICML'14, pages II–1566–II–1574. JMLR.org.

Chen, Y., Sanghavi, S., and Xu, H. (2012).
Clustering Sparse Graphs.
In Pereira, F., Burges, C. J. C., Bottou, L., and Weinberger, K. Q., editors, Advances in Neural Information Processing Systems 25, pages 2204–2212. Curran Associates, Inc.

Cormode, G. and Firmani, D. (2014).
A unifying framework for l_0-sampling algorithms.
Distributed and Parallel Databases, 32(3):315–335.
Special issue on Data Summarization on Big Data.

References III

Falkowski, T., Barth, A., and Spiliopoulou, M. (2007).
DENGRAPH: A Density-based Community Detection Algorithm.
In Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence, WI '07, pages 112–115, Washington, DC, USA. IEEE Computer Society.

Gong, Y., Lazebnik, S., Gordo, A., and Perronnin, F. (2013).
Iterative Quantization: A Procrustean Approach to Learning Binary Codes for Large-Scale Image Retrieval.
IEEE Transactions on Pattern Analysis and Machine Intelligence, (12):2916–2929.

Grygorash, O., Zhou, Y., and Jorgensen, Z. (2006).
Minimum Spanning Tree Based Clustering Algorithms.
In 2006 18th IEEE International Conference on Tools with Artificial Intelligence (ICTAI'06), pages 73–81.

Guha, S., Rastogi, R., and Shim, K. (2001).
Cure: An Efficient Clustering Algorithm for Large Databases.
Inf. Syst., 26(1):35–58.

Johnson, W. and Lindenstrauss, J. (1984).
Extensions of Lipschitz mappings into a Hilbert space.
In Conference in modern analysis and probability (New Haven, Conn., 1982), volume 26 of Contemporary Mathematics, pages 189–206. American Mathematical Society.

Kong, W. and Li, W.-j. (2012).
Isotropic Hashing.
In NIPS, pages 1646–1654.

References IV

Leng, C., Wu, J., Cheng, J., Bai, X., and Lu, H. (2015).
Online sketching hashing.
In CVPR, pages 2503–2511.

Liberty, E. (2013).
Simple and Deterministic Matrix Sketching.
In Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '13, pages 581–588, New York, NY, USA. ACM.

Oxley, J. G. (2006).
Matroid Theory (Oxford Graduate Texts in Mathematics).
Oxford University Press, Inc., New York, NY, USA.

Oymak, S. and Hassibi, B. (2011).
Finding Dense Clusters via "Low Rank + Sparse" Decomposition.
CoRR, abs/1104.5186.

Schaeffer, S. E. (2007).
Survey: Graph Clustering.
Comput. Sci. Rev., 1(1):27–64.

Wang, J., Zhang, T., j. song, Sebe, N., and Shen, H. T. (2018).
A Survey on Learning to Hash.
IEEE Trans. on Pattern Anal. and Mach. Intell., PP(99):1–1.

References V

Zahn, C. T. (1971).
Graph-Theoretical Methods for Detecting and Describing Gestalt Clusters.
IEEE Trans. Comput., 20(1):68–86.

Thank you for your attention!

anne.morvan@cea.fr

↓

amorvan@expedia.com

Plan

Submodular?

Let $f : 2^E \to \mathbb{R}$ be the DBCVI function. f is clearly nonlinear. Suppose that $S \subset T$ and $e \notin T$.

Is the marginal DBCVI value of adding e to T is not larger than the marginal DBCVI value of adding e to S?

$$\forall S \subset T \subset T \cup \{e\}, \quad \underbrace{f(T \cup \{e\}) - f(T)}_{\text{marginal DBCVi value } +e} \leq f(S \cup \{e\}) - f(S) \quad (1)$$

If f satisfies Eq.2, f is submodular.

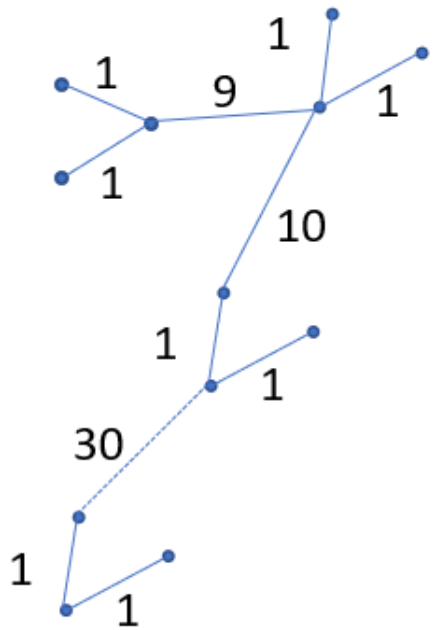
f submodular?

Suppose that $S \subset T$ and $e \notin T$.

Is the marginal DBCVI value of adding e to T is not larger than the marginal DBCVI value of adding e to S?

$$S = \{1, 1, \ldots, 1\}$$
$$T = \{1, 1, \ldots, 1, 9\}$$
$$e = 10$$

$$0.06 = \underbrace{f(T \cup \{e\})}_{0.58} - \underbrace{f(T)}_{0.52} > \underbrace{f(S \cup \{e\})}_{0.41} - \underbrace{f(S)}_{0.66} = -0.15$$



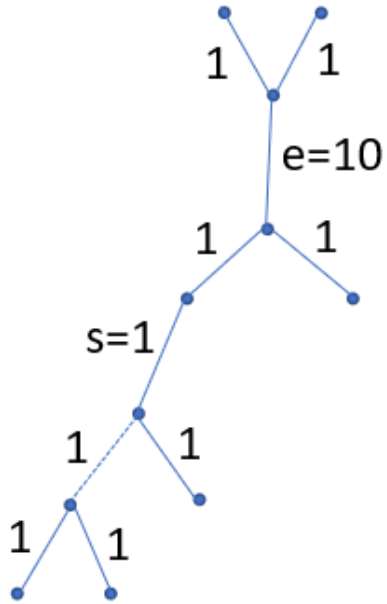
$-f$ submodular?

$$\forall S \subset T \subset T \cup \{e\}, \quad \underbrace{f(T \cup \{e\}) - f(T)}_{\text{marginal DBCVi value } +e} \geq f(S \cup \{e\}) - f(S) \quad (2)$$

If $-f$ satisfies Eq.2, $-f$ is submodular.

$$S = \{1, 1, \ldots, 1\}$$
$$T = \{1, 1, \ldots, 1, s\}$$
$$e = 10$$

$$-0.91 = \underbrace{f(T \cup \{e\})}_{-0.68} - \underbrace{f(T)}_{0.23} < \underbrace{f(S \cup \{e\})}_{-0.45} - \underbrace{f(S)}_{0.23} = -0.68$$



Matroid

Theorem [Oxley, 2006]

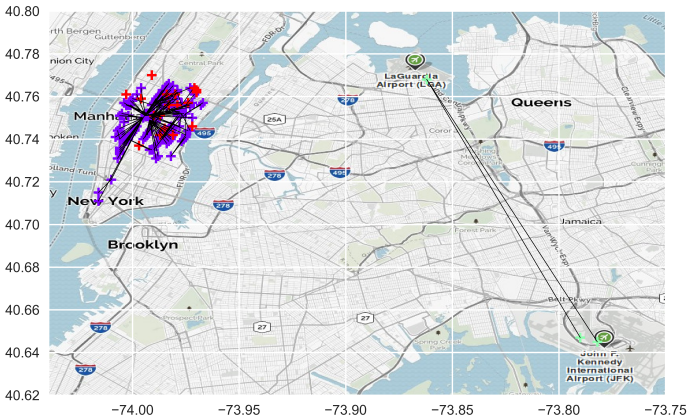
Let $\mathcal{I}$ be a collection of subsets of E. The $(E, \mathcal{I})$ is a matroid iff $\mathcal{I}$ satisfies the following conditions:

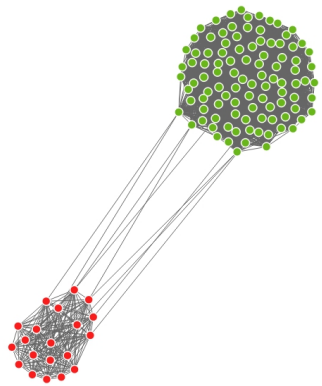
- $\emptyset \subset \mathcal{I}$
- If $I \in \mathcal{I}$ and $I' \in \mathcal{I}$.
- For all weight functions: $w : 2^E \rightarrow \mathbb{R}$, the greedy algorithm produces a maximal member of $\mathcal{I}$ of maximum weight.

Applications

For graph data or build a dissimilarity graph (weights are distances or dissimilarities). Examples:

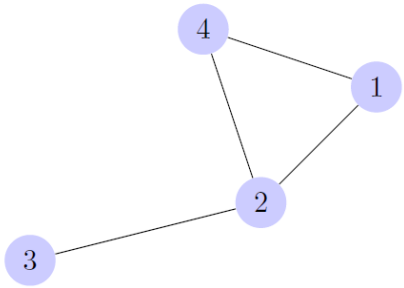
- Vertices: locations in NYC, weights: $max - \#$ trips between two points, goal: recover geographical zones
- V: webpages, w: $max - \#$users getting from one page to another, goal: identify controlled webpages,
- V: individuals, w: dissimilarity, goal: identify terrorist groups in social network





Graph sketching [Ahn et al., 2012, Cormode and Firmani, 2014]

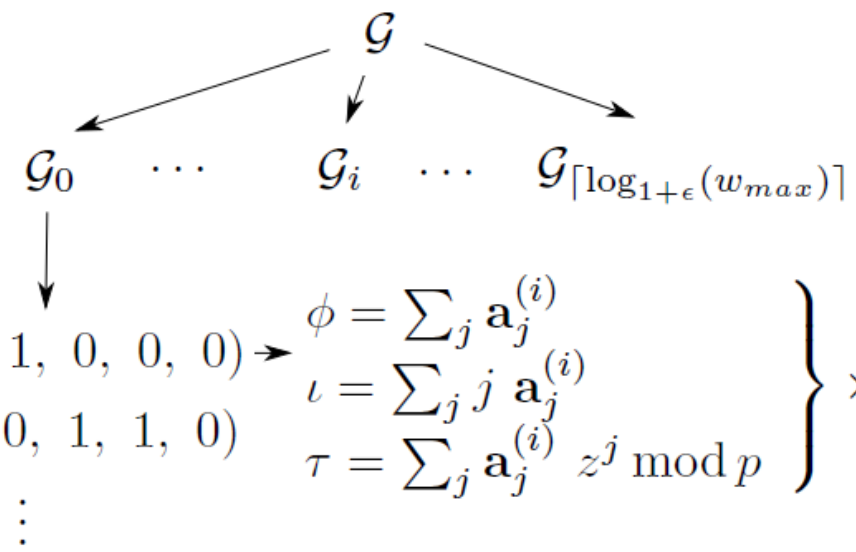
A compact structure for $\mathcal{G}$ in $O(N \log^3(N))$



weighted

unweighted

$\mathcal{G}_i \subset \mathcal{G}_{i+1}$



L levels of representation:

$$\begin{cases} h : [|E|] \to [L] \\ Pr[h(j) = l] = \frac{1}{2^l} \end{cases}$$

1-sparsity test

If $\tau = \phi\ z^{\frac{\iota}{\phi}} \bmod p$ then $\mathbf{a}^{(i)}$ is 1-sparse. If $\mathbf{a}^{(i)}$ is 1-sparse: always $+$ answer, otherwise $-$ answer with prob. at least $1 - |E|/p$ with p a suitably large prime and $z \in \mathbb{Z}_p$

UnifDiag Model

- For any matrix $\mathbf{M}$, $\mathbf{\Sigma_M} = \mathbf{M}\mathbf{M}^T$.
  $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n] \in \mathbb{R}^{d \times n}$; $\mathbf{V} = \mathbf{W}\mathbf{X} \in \mathbb{R}^{c \times n}$; $\mathbf{Y} = \mathbf{R}\mathbf{V} \in \mathbb{R}^{c \times n}$
- Role of $\mathbf{R}$: balancing variance over the c directions
- Equivalence: equalizing the diagonal coefficients of $\mathbf{\Sigma_Y}$ to the same value

$$\tau = \mathrm{Tr}(\mathbf{\Sigma_V})/c$$

- $\mathbf{\Sigma_V}$ dynamically computed as a new data data point is seen during update of $\mathbf{W}$ with OPAST

UnifDiag Model

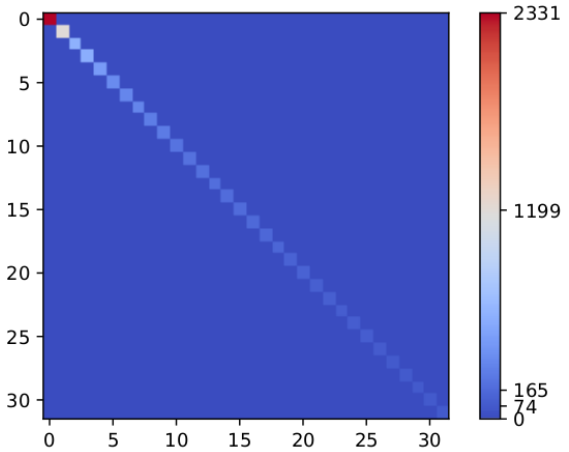
- For $r \in \{1, \ldots, c-1\}$, given i_r, j_r, θ_r, with $(\mathbf{\Sigma_Y})_0 = \mathbf{\Sigma_V}$, $\mathbf{R}_0 = \mathbf{I}_c$

rows columns

$$(\mathbf{\Sigma_Y})_r \leftarrow \mathbf{G}(i_r, j_r, \theta_r)\ (\mathbf{\Sigma_Y})_{r-1}\ \mathbf{G}(i_r, j_r, \theta_r)^T$$

$$\mathbf{R}_r \leftarrow \mathbf{R}_{r-1}\ \mathbf{G}(i_r, j_r, \theta_r)^T.$$

- At each step r, i_r and j_r are chosen to be the indices of some diagonal coefficients of $(\mathbf{\Sigma_Y})_{r-1}$ below, respectively above τ.



PCA

UnifDiag Model

- For $r \in \{1, \dots, c-1\}$, given i_r, j_r, θ_r, with $(\mathbf{\Sigma_Y})_0 = \mathbf{\Sigma_V}$, $\mathbf{R}_0 = \mathbf{I}_c$

rows columns

$$(\mathbf{\Sigma_Y})_r \leftarrow \mathbf{G}(i_r, j_r, \theta_r)\ (\mathbf{\Sigma_Y})_{r-1}\ \mathbf{G}(i_r, j_r, \theta_r)^T$$

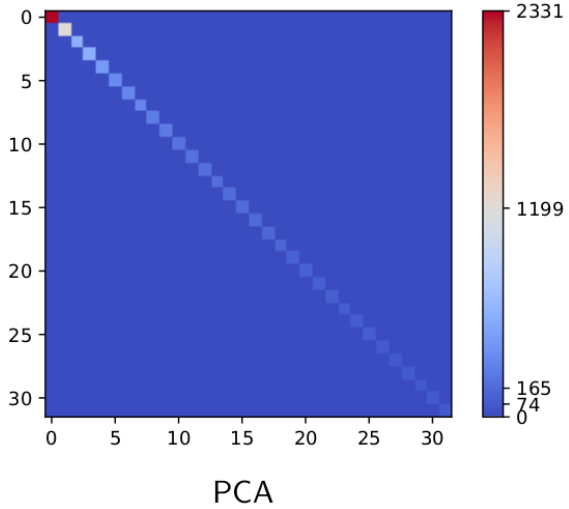
$$\mathbf{R}_r \leftarrow \mathbf{R}_{r-1}\ \mathbf{G}(i_r, j_r, \theta_r)^T.$$

- $c = \cos(\theta_r)$, $s = \sin(\theta_r)$, $a = \mathbf{\Sigma_V}_{j,j}$, $d = \mathbf{\Sigma_V}_{i,i}$, $b = \mathbf{\Sigma_V}_{j,i} = \mathbf{\Sigma_V}_{i,j}$

$$\begin{pmatrix} a' & b' \\ b' & d' \end{pmatrix} := \begin{pmatrix} c & -s \\ s & c \end{pmatrix} \begin{pmatrix} a & b \\ b & d \end{pmatrix} \begin{pmatrix} c & s \\ -s & c \end{pmatrix}$$

$$a' = \tau,\ d' = a + d - \tau$$

θ_r is computed accordingly.



PCA

- End of step r: r diagonal coefficients of $(\mathbf{\Sigma_Y})_r$ are equal to τ.

Example, initial $\mathbf{\Sigma_V}$

$$c = 8$$

$$(\mathbf{\Sigma_Y})_0 = \begin{pmatrix} \mathbf{101.94} & 12.14 & 6.95 & -6.79 & -3.30 & -5.03 & -14.92 & 0.6801 \\ 12.14 & \mathbf{107.73} & 5.03 & 5.81 & -4.07 & -10.28 & -12.06 & 7.96 \\ 6.95 & 5.03 & \mathbf{90.98} & 12.37 & 9.79 & 18.91 & 5.66 & -5.48 \\ -6.79 & 5.81 & 12.4 & \mathbf{90.24} & -8.05 & 27.72 & 19.99 & -18.49 \\ -3.30 & -4.07 & 9.79 & -8.05 & \mathbf{107.59} & 10.20 & 11.97 & -5.12 \\ -5.03 & -10.28 & 18.91 & 27.72 & 10.20 & \mathbf{107.54} & -2.99 & -17.50 \\ -14.92 & -12.06 & 5.66 & 19.99 & 11.97 & -2.99 & \mathbf{102.42} & 5.07 \\ 0.68 & 7.96 & -5.48 & -18.49 & -5.12 & -17.50 & 5.07 & \mathbf{90.20} \end{pmatrix}$$

$$\tau = \frac{\mathrm{Tr}(\mathbf{\Sigma}_V)}{c} = 99.83$$

The two first coefficients to change

$$(\mathbf{\Sigma_Y})_0 = \begin{pmatrix} \mathbf{101.94} & 12.14 & 6.95 & -6.79 & -3.30 & -5.03 & -14.92 & 0.6801 \\ 12.14 & \mathbf{107.73} & 5.03 & 5.81 & -4.07 & -10.28 & -12.06 & 7.96 \\ 6.95 & 5.03 & \mathbf{90.98} & 12.37 & 9.79 & 18.91 & 5.66 & -5.48 \\ -6.79 & 5.81 & 12.37 & \mathbf{90.24} & -8.05 & 27.72 & 19.99 & -18.49 \\ -3.30 & -4.07 & 9.79 & -8.05 & \mathbf{107.59} & 10.20 & 11.97 & -5.12 \\ -5.03 & -10.28 & 18.91 & 27.72 & 10.20 & \mathbf{107.54} & -2.99 & -17.50 \\ -14.92 & -12.06 & 5.66 & 19.99 & 11.97 & -2.99 & \mathbf{102.42} & 5.07 \\ 0.68 & 7.96 & -5.48 & -18.49 & -5.12 & -17.50 & 5.07 & \mathbf{90.20} \end{pmatrix}$$

$$i_1 = 2$$

$$j_1 = 0$$

$$i_1 > j_1$$

The two first coefficients to change

$$c_1 = 0.138$$

$$s_1 = 0.990$$

$$\mathbf{R}_1 = \begin{pmatrix} c_1 & 0 & s_1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -s_1 & 0 & c_1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

$$(\mathbf{\Sigma_Y})_1 = \begin{pmatrix} \mathbf{93.09} & 6.66 & -8.18 & 11.31 & 9.24 & 18.03 & 3.54 & -5.34 \\ 6.66 & \mathbf{107.73} & -11.33 & 5.81 & -4.07 & -10.28 & -12.06 & 7.96 \\ -8.18 & -11.33 & \mathbf{99.83} & 8.43 & 4.62 & 7.56 & 15.56 & -1.43 \\ 11.31 & 5.81 & 8.43 & \mathbf{90.24} & -8.05 & 27.72 & 19.99 & -18.49 \\ 9.29 & -4.07 & 4.62 & -8.05 & \mathbf{107.59} & 10.20 & 11.97 & -5.12 \\ 18.03 & -10.23 & 7.56 & 27.72 & 10.20 & 107.54 & -2.99 & -17.50 \\ 3.54 & -12.06 & 15.56 & 19.99 & 11.97 & -2.99 & \mathbf{102.42} & 5.07 \\ -5.34 & 7.95 & -1.43 & -18.49 & -5.12 & -17.50 & 5.07 & \mathbf{90.20} \end{pmatrix}$$

Step $r = 2$

$$(\mathbf{\Sigma_Y})_1 = \begin{pmatrix} \mathbf{93.09} & 6.66 & -8.18 & 11.31 & 9.24 & 18.03 & 3.54 & -5.34 \\ 6.66 & \mathbf{107.73} & -11.33 & 5.81 & -4.07 & -10.23 & -12.04 & 7.96 \\ -8.18 & -11.33 & \mathbf{99.83} & 8.43 & 4.62 & 7.56 & 15.56 & -1.43 \\ 11.31 & 5.81 & 8.43 & \mathbf{90.24} & -8.05 & 27.72 & 19.99 & -18.49 \\ 9.29 & -4.07 & 4.62 & -8.05 & \mathbf{107.59} & 10.20 & 11.97 & -5.12 \\ 18.03 & -10.23 & 7.56 & 27.72 & 10.20 & \mathbf{107.54} & -2.99 & -17.50 \\ 3.54 & -12.06 & 15.56 & 19.99 & 11.97 & -2.99 & \mathbf{102.42} & 5.07 \\ -5.34 & 7.95 & -1.43 & -18.49 & -5.12 & -17.50 & 5.07 & \mathbf{90.20} \end{pmatrix}$$

$$i_2 = 0$$

$$j_2 = 1$$

$$i_2 < j_2$$

Step $r = 2$

$$c_2 = 0.431$$

$$s_2 = 0.902$$

$$\mathbf{R}_2 = \mathbf{R}_1 . \begin{pmatrix} c_2 & -s_2 & 0 & 0 & 0 & 0 & 0 & 0 \\ s_2 & c_2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

$$(\mathbf{\Sigma_Y})_1 = \begin{pmatrix} \mathbf{99.83} & -9.88 & 6.70 & -0.37 & 7.66 & 17.04 & 12.41 & -9.48 \\ -9.88 & \mathbf{100.99} & -12.27 & 12.71 & 6.58 & 11.85 & -2.00 & -1.39 \\ 6.70 & -12.27 & \mathbf{99.83} & 8.43 & 4.62 & 7.56 & 15.56 & -1.43 \\ -0.37 & 12.71 & 8.43 & \mathbf{90.24} & -8.05 & 27.72 & 19.99 & -18.49 \\ 7.66 & 6.58 & 4.62 & -8.05 & \mathbf{107.59} & 10.20 & 11.97 & -5.12 \\ 17.05 & 11.85 & 7.56 & 27.72 & 10.20 & \mathbf{107.54} & -2.99 & -17.50 \\ 12.41 & -1.20 & 15.56 & 19.99 & 11.97 & -2.99 & \mathbf{102.42} & 5.07 \\ -9.48 & -1.39 & -1.43 & -18.49 & -5.12 & -17.50 & 5.07 & \mathbf{90.20} \end{pmatrix}$$

Step $r = 3$

$$(\mathbf{\Sigma_Y})_2 = \begin{pmatrix} \mathbf{99.83} & -9.88 & 6.70 & -0.37 & 7.66 & 17.04 & 12.41 & -9.48 \\ -9.88 & \mathbf{100.99} & -12.27 & 12.71 & 6.58 & 11.85 & -2.00 & -1.39 \\ 6.70 & -12.27 & \mathbf{99.83} & 8.43 & 4.62 & 7.56 & 15.56 & -1.43 \\ -0.37 & 12.71 & 8.43 & \mathbf{90.24} & -8.05 & 27.72 & 19.99 & -18.49 \\ 7.66 & 6.58 & 4.62 & -8.05 & \mathbf{107.59} & 10.20 & 11.97 & -5.12 \\ 17.05 & 11.85 & 7.56 & 27.72 & 10.20 & \mathbf{107.54} & -2.99 & -17.50 \\ 12.41 & -1.20 & 15.56 & 19.99 & 11.97 & -2.99 & \mathbf{102.42} & 5.07 \\ -9.48 & -1.39 & -1.43 & -18.49 & -5.12 & -17.50 & 5.07 & \mathbf{90.20} \end{pmatrix}$$

$$i_3 = 3$$

$$j_3 = 1$$

$$i_3 > j_3$$

Step $r = 3$

$$c_3 = 0.045$$

$$s_3 = 0.999$$

$$\mathbf{R}_3 = \mathbf{R}_2 . \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & c_3 & 0 & s_3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -s_3 & 0 & c_3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

$$(\mathbf{\Sigma_Y})_3 = \begin{pmatrix} \mathbf{99.83} & -0.81 & 6.70 & 9.85 & 7.66 & 17.04 & 12.41 & -9.48 \\ -0.81 & \mathbf{91.40} & 7.87 & -13.14 & -7.75 & 28.22 & 19.88 & -18.53 \\ 6.70 & 7.87 & \mathbf{99.83} & 12.63 & 4.62 & 7.60 & 15.56 & -1.43 \\ 9.85 & -13.14 & 12.63 & \mathbf{99.83} & -6.94 & -10.59 & 2.89 & 0.56 \\ 7.66 & -7.75 & 4.62 & -6.94 & \mathbf{107.59} & 10.20 & 11.97 & -5.12 \\ 17.04 & 28.22 & 7.56 & -10.59 & 10.20 & \mathbf{107.54} & -2.99 & -17.50 \\ 12.41 & 19.88 & 15.56 & 2.89 & 11.97 & -2.99 & \mathbf{102.42} & 5.07 \\ -9.48 & -18.53 & -1.43 & 0.56 & -5.12 & -17.50 & 5.07 & \mathbf{90.20} \end{pmatrix}$$

Step $r = 4$

$$(\mathbf{\Sigma_Y})_3 = \begin{pmatrix} \mathbf{99.83} & -0.81 & 6.70 & 9.85 & 7.66 & 17.04 & 12.41 & -9.48 \\ -0.81 & \mathbf{91.40} & 7.87 & -13.14 & -7.75 & 28.22 & 19.88 & -18.53 \\ 6.70 & 7.87 & \mathbf{99.83} & 12.63 & 4.62 & 7.60 & 15.56 & -1.43 \\ 9.85 & -13.14 & 12.63 & \mathbf{99.83} & -6.94 & -10.59 & 2.89 & 0.56 \\ 7.66 & -7.75 & 4.62 & -6.94 & \mathbf{107.59} & 10.20 & 11.97 & -5.12 \\ 17.04 & 28.22 & 7.56 & -10.59 & 10.20 & \mathbf{107.54} & -2.99 & -17.50 \\ 12.41 & 19.88 & 15.56 & 2.89 & 11.97 & -2.99 & \mathbf{102.42} & 5.07 \\ -9.48 & -18.53 & -1.43 & 0.56 & -5.12 & -17.50 & 5.07 & \mathbf{90.20} \end{pmatrix}$$

$$i_4 = 1$$

$$j_4 = 4$$

$$i_4 < j_4$$

Step $r = 4$

$$c_4 = 0.913$$

$$s_4 = 0.407$$

$$\mathbf{R}_4 = \mathbf{R}_3 . \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & c_4 & 0 & 0 & -s_4 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & s_4 & 0 & 0 & c_4 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

$$(\mathbf{\Sigma_Y})_4 = \begin{pmatrix} \mathbf{99.83} & -3.85 & 6.70 & 9.85 & 6.67 & 17.04 & 12.41 & -9.48 \\ -3.85 & \mathbf{99.83} & 5.31 & -9.18 & -11.20 & 21.64 & 13.29 & -14.85 \\ 6.70 & 5.31 & \mathbf{99.83} & 12.63 & 7.42 & 7.56 & 15.56 & -1.43 \\ 9.85 & -9.18 & 12.63 & \mathbf{99.83} & -11.68 & -10.59 & 2.89 & 0.56 \\ 6.67 & -11.20 & 7.42 & -11.68 & \mathbf{99.16} & 20.79 & 19.02 & -12.21 \\ 17.04 & 21.64 & 7.56 & -10.59 & 20.79 & \mathbf{107.54} & -2.99 & -17.50 \\ 12.41 & 13.29 & 15.56 & 2.88 & 19.02 & -2.99 & \mathbf{102.42} & 5.07 \\ -9.48 & -14.85 & -1.43 & 0.56 & -12.21 & -17.50 & 5.07 & \mathbf{90.20} \end{pmatrix}$$

Step $r = 5$

$$(\mathbf{\Sigma_Y})_4 = \begin{pmatrix} \mathbf{99.83} & -3.85 & 6.70 & 9.85 & 6.67 & 17.04 & 12.41 & -9.48 \\ -3.85 & \mathbf{99.83} & 5.31 & -9.18 & -11.20 & 21.64 & 13.29 & -14.85 \\ 6.70 & 5.31 & \mathbf{99.83} & 12.63 & 7.42 & 7.56 & 15.56 & -1.43 \\ 9.85 & -9.18 & 12.63 & \mathbf{99.83} & -11.68 & -10.59 & 2.89 & 0.56 \\ 6.67 & -11.20 & 7.42 & -11.68 & \mathbf{99.16} & 20.79 & 19.02 & -12.21 \\ 17.04 & 21.64 & 7.56 & -10.59 & 20.79 & \mathbf{107.54} & -2.99 & -17.50 \\ 12.41 & 13.29 & 15.56 & 2.88 & 19.02 & -2.99 & \mathbf{102.42} & 5.07 \\ -9.48 & -14.85 & -1.43 & 0.56 & -12.21 & -17.50 & 5.07 & \mathbf{90.20} \end{pmatrix}$$

$$i_5 = 4$$

$$j_5 = 5$$

$$i_5 < j_5$$

Step $r = 5$

$$c_5 = 0.182$$

$$s_5 = 0.983$$

$$\mathbf{R}_5 = \mathbf{R}_4 . \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & c_5 & -s_5 & 0 & 0 \\ 0 & 0 & 0 & 0 & s_5 & c_5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

$$(\mathbf{\Sigma_Y})_5 = \begin{pmatrix} \mathbf{99.83} & -3.85 & 6.70 & 9.85 & -15.55 & 9.65 & 12.41 & -9.48 \\ -3.85 & \mathbf{99.83} & 5.31 & -9.18 & -23.32 & -7.08 & 13.29 & -14.85 \\ 6.70 & 5.31 & \mathbf{99.83} & 12.63 & -6.12 & 8.68 & 15.56 & -1.43 \\ 9.85 & -9.18 & 12.63 & \mathbf{99.83} & 8.29 & -13.41 & 2.89 & 0.56 \\ -15.55 & -23.32 & -6.12 & 8.29 & \mathbf{99.83} & -20.92 & 6.40 & 14.99 \\ 9.65 & -7.08 & 8.68 & -13.41 & -20.92 & \mathbf{106.87} & 18.16 & -15.19 \\ 12.41 & 13.29 & 15.56 & 2.89 & 6.40 & 18.16 & \mathbf{102.42} & 5.07 \\ -9.48 & -14.85 & -1.43 & 0.56 & 14.99 & -15.19 & 5.07 & \mathbf{90.20} \end{pmatrix}$$

Step $r = 6$

$$(\mathbf{\Sigma_Y})_5 = \begin{pmatrix} \mathbf{99.83} & -3.85 & 6.70 & 9.85 & -15.55 & 9.65 & 12.41 & -9.48 \\ -3.85 & \mathbf{99.83} & 5.31 & -9.18 & -23.32 & -7.08 & 13.29 & -14.85 \\ 6.70 & 5.31 & \mathbf{99.83} & 12.63 & -6.12 & 8.68 & 15.56 & -1.43 \\ 9.85 & -9.18 & 12.63 & \mathbf{99.83} & 8.29 & -13.41 & 2.89 & 0.56 \\ -15.55 & -23.32 & -6.12 & 8.29 & \mathbf{99.83} & -20.92 & 6.40 & 14.99 \\ 9.65 & -7.08 & 8.68 & -13.41 & -20.92 & \mathbf{106.87} & 18.16 & -15.19 \\ 12.41 & 13.29 & 15.56 & 2.89 & 6.40 & 18.16 & \mathbf{102.42} & 5.07 \\ -9.48 & -14.85 & -1.43 & 0.56 & 14.99 & -15.19 & 5.07 & \mathbf{90.20} \end{pmatrix}$$

$$i_6 = 7$$

$$j_6 = 5$$

$$i_6 > j_6$$

Step $r = 6$

$$c_6 = 0.959$$

$$s_6 = 0.284$$

$$\mathbf{R}_6 = \mathbf{R}_5 . \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & c_6 & 0 & s_6 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & -s_6 & 0 & c_6 \end{pmatrix}$$

$$(\mathbf{\Sigma_Y})_6 = \begin{pmatrix} \mathbf{99.83} & -3.85 & 6.70 & 9.85 & -15.55 & 6.56 & 12.40 & -11.83 \\ -3.85 & \mathbf{99.83} & 5.31 & -9.18 & -23.32 & -11.01 & 13.29 & -12.22 \\ 6.70 & 5.31 & \mathbf{99.83} & 12.63 & -6.12 & 7.91 & 15.56 & -3.84 \\ 9.85 & -9.18 & 12.63 & \mathbf{99.83} & 8.29 & -12.70 & 2.89 & 4.35 \\ -15.55 & -23.32 & -6.12 & 8.29 & \mathbf{99.83} & -15.79 & 6.40 & 20.32 \\ 6.56 & -11.02 & 7.91 & -12.70 & -15.79 & \mathbf{97.24} & 18.85 & -17.28 \\ 12.41 & 13.29 & 15.56 & 2.89 & 6.40 & 18.85 & \mathbf{102.42} & -0.31 \\ -11.83 & -12.22 & -3.84 & 4.35 & 20.32 & -17.28 & -0.31 & \mathbf{99.83} \end{pmatrix}$$

Step $r = 7$

$$(\mathbf{\Sigma_Y})_6 = \begin{pmatrix} \mathbf{99.83} & -3.85 & 6.70 & 9.85 & -15.55 & 6.56 & 12.40 & -11.83 \\ -3.85 & \mathbf{99.83} & 5.31 & -9.18 & -23.32 & -11.01 & 13.29 & -12.22 \\ 6.70 & 5.31 & \mathbf{99.83} & 12.63 & -6.12 & 7.91 & 15.56 & -3.84 \\ 9.85 & -9.18 & 12.63 & \mathbf{99.83} & 8.29 & -12.70 & 2.89 & 4.35 \\ -15.55 & -23.32 & -6.12 & 8.29 & \mathbf{99.83} & -15.79 & 6.40 & 20.32 \\ 6.56 & -11.02 & 7.91 & -12.70 & -15.79 & \mathbf{97.24} & 18.85 & -17.28 \\ 12.41 & 13.29 & 15.56 & 2.89 & 6.40 & 18.85 & \mathbf{102.42} & -0.31 \\ -11.83 & -12.22 & -3.84 & 4.35 & 20.32 & -17.28 & -0.31 & \mathbf{99.83} \end{pmatrix}$$

$$i_7 = 5$$

$$j_7 = 6$$

$$i_7 < j_7$$

Step $r = 7$

$$c_7 = 0.068$$

$$s_7 = 0.998$$

$$\mathbf{R}_7 = \mathbf{R}_6 . \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & c_7 & -s_7 & 0 \\ 0 & 0 & 0 & 0 & 0 & s_7 & c_7 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

$$(\mathbf{\Sigma_Y})_7 = \begin{pmatrix} \mathbf{99.83} & -3.85 & 6.70 & 9.85 & -15.55 & -11.93 & 7.39 & -11.83 \\ -3.85 & \mathbf{99.83} & 5.31 & -9.18 & -23.32 & -14.02 & -10.08 & -12.22 \\ 6.70 & 5.31 & \mathbf{99.83} & 12.63 & -6.12 & -14.99 & 8.96 & -3.84 \\ 9.85 & -9.18 & 12.63 & \mathbf{99.83} & 8.29 & -3.75 & -12.47 & 4.35 \\ -15.55 & -23.32 & -6.12 & 8.29 & \mathbf{99.83} & -7.46 & -15.32 & 20.32 \\ -11.93 & -14.02 & -14.99 & -3.75 & -7.46 & \mathbf{99.83} & -19.03 & -0.87 \\ 7.39 & -10.08 & 8.96 & -12.47 & -15.32 & -19.03 & \mathbf{99.83} & -17.26 \\ -11.83 & -12.22 & -3.84 & 4.35 & 20.32 & -0.87 & -17.26 & \mathbf{99.83} \end{pmatrix}$$

End

$$c_7 = 0.068$$

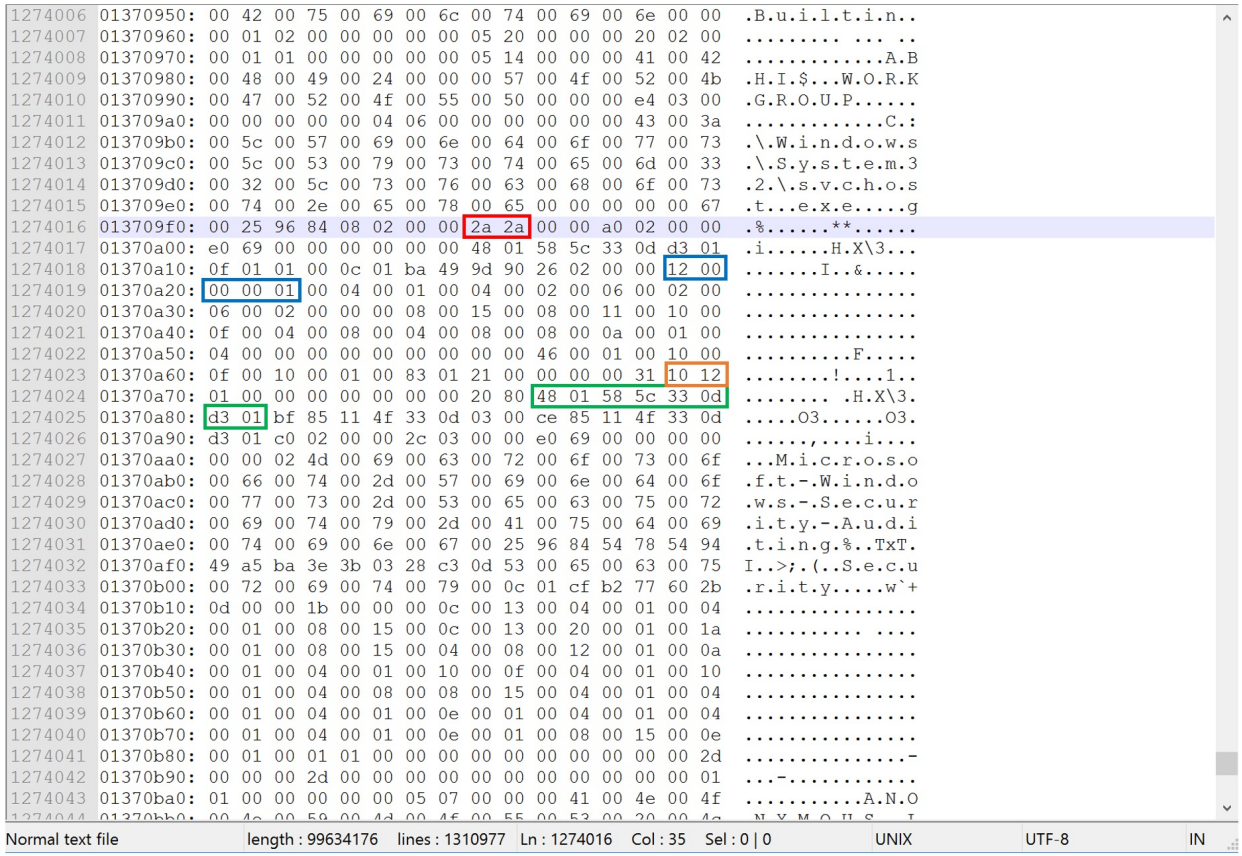
$$s_7 = 0.998$$

$$\mathbf{R}_7 = \mathbf{R}_6 . \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & c_7 & -s_7 & 0 \\ 0 & 0 & 0 & 0 & 0 & s_7 & c_7 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

$$(\mathbf{\Sigma_Y})_7 = \begin{pmatrix} \mathbf{99.83} & -3.85 & 6.70 & 9.85 & -15.55 & -11.93 & 7.39 & -11.83 \\ -3.85 & \mathbf{99.83} & 5.31 & -9.18 & -23.32 & -14.02 & -10.08 & -12.22 \\ 6.70 & 5.31 & \mathbf{99.83} & 12.63 & -6.12 & -14.99 & 8.96 & -3.84 \\ 9.85 & -9.18 & 12.63 & \mathbf{99.83} & 8.29 & -3.75 & -12.47 & 4.35 \\ -15.55 & -23.32 & -6.12 & 8.29 & \mathbf{99.83} & -7.46 & -15.32 & 20.32 \\ -11.93 & -14.02 & -14.99 & -3.75 & -7.46 & \mathbf{99.83} & -19.03 & -0.87 \\ 7.39 & -10.08 & 8.96 & -12.47 & -15.32 & -19.03 & \mathbf{99.83} & -17.26 \\ -11.83 & -12.22 & -3.84 & 4.35 & 20.32 & -0.87 & -17.26 & \mathbf{99.83} \end{pmatrix}$$

Applications

- Images, text, numerical data etc.
- Application field example: cybersecurity with events monitoring, pattern recognition, etc.



```
1274006 01370950: 00 42 00 75 00 69 00 6c 00 74 00 69 00 6e 00 00  .B.u.i.l.t.i.n..
1274007 01370960: 00 01 02 00 00 00 00 00 05 20 00 00 00 20 02 00  ......... ... ..
1274008 01370970: 00 01 01 00 00 00 00 00 05 14 00 00 00 41 00 42  .............A.B
1274009 01370980: 00 48 00 49 00 24 00 00 00 57 00 4f 00 52 00 4b  .H.I.$...W.O.R.K
1274010 01370990: 00 47 00 52 00 4f 00 55 00 50 00 00 00 e4 03 00  .G.R.O.U.P......
1274011 013709a0: 00 00 00 00 00 04 06 00 00 00 00 00 00 43 00 3a  .............C.:
1274012 013709b0: 00 5c 00 57 00 69 00 6e 00 64 00 6f 00 77 00 73  .\.W.i.n.d.o.w.s
1274013 013709c0: 00 5c 00 53 00 79 00 73 00 74 00 65 00 6d 00 33  .\.S.y.s.t.e.m.3
1274014 013709d0: 00 32 00 5c 00 73 00 76 00 63 00 68 00 6f 00 73  .2.\.s.v.c.h.o.s
1274015 013709e0: 00 74 00 2e 00 65 00 78 00 65 00 00 00 00 00 67  .t...e.x.e.....g
1274016 013709f0: 00 25 96 84 08 02 00 00 2a 2a 00 00 a0 02 00 00  .%......**......
1274017 01370a00: e0 69 00 00 00 00 00 00 48 01 58 5c 33 0d d3 01  .i......H.X\3...
1274018 01370a10: 0f 01 01 00 0c 01 ba 49 9d 90 26 02 00 00 12 00  .......I..&.....
1274019 01370a20: 00 00 01 00 04 00 01 00 04 00 02 00 06 00 02 00  ................
1274020 01370a30: 06 00 02 00 00 00 00 08 00 15 00 08 00 11 00 10  ................
1274021 01370a40: 0f 00 04 00 08 00 04 00 08 00 08 00 0a 00 01 00  ................
1274022 01370a50: 04 00 00 00 00 00 00 00 00 00 46 00 01 00 10 00  ..........F.....
1274023 01370a60: 0f 00 10 00 01 00 83 01 21 00 00 00 00 31 10 12  ........!....1..
1274024 01370a70: 01 00 00 00 00 00 00 00 00 20 80 48 01 58 5c 33  ......... .H.X\3.
1274025 01370a80: d3 01 bf 85 11 4f 33 0d 03 00 ce 85 11 4f 33 0d  .....O3......O3.
1274026 01370a90: d3 01 c0 02 00 00 2c 03 00 00 e0 69 00 00 00 00  ......,....i....
1274027 01370aa0: 00 00 02 4d 00 69 00 63 00 72 00 6f 00 73 00 6f  ...M.i.c.r.o.s.o
1274028 01370ab0: 00 66 00 74 00 2d 00 57 00 69 00 6e 00 64 00 6f  .f.t.-.W.i.n.d.o
1274029 01370ac0: 00 77 00 73 00 2d 00 53 00 65 00 63 00 75 00 72  .w.s.-.S.e.c.u.r
1274030 01370ad0: 00 69 00 74 00 79 00 2d 00 41 00 75 00 64 00 69  .i.t.y.-.A.u.d.i
1274031 01370ae0: 00 74 00 69 00 6e 00 67 00 25 96 84 54 78 54 94  .t.i.n.g.%..TxT.
1274032 01370af0: 49 a5 ba 3e 3b 03 28 c3 0d 53 00 65 00 63 00 75  I..>;.(..S.e.c.u
1274033 01370b00: 00 72 00 69 00 74 00 79 00 0c 01 cf b2 77 60 2b  .r.i.t.y.....w`+
1274034 01370b10: 0d 00 00 1b 00 00 00 0c 00 13 00 04 00 01 00 04  ................
1274035 01370b20: 00 01 00 08 00 15 00 0c 00 13 00 20 00 01 00 1a  ........... ....
1274036 01370b30: 00 01 00 08 00 15 00 04 00 08 00 12 00 01 00 0a  ................
1274037 01370b40: 00 01 00 04 00 01 00 10 00 0f 00 04 00 01 00 10  ................
1274038 01370b50: 00 01 00 04 00 08 00 08 00 15 00 04 00 01 00 04  ................
1274039 01370b60: 00 01 00 04 00 01 00 0e 00 01 00 04 00 01 00 04  ................
1274040 01370b70: 00 01 00 04 00 01 00 0e 00 01 00 08 00 15 00 0e  ................
1274041 01370b80: 00 01 00 01 01 00 00 00 00 00 00 00 00 00 00 2d  ...............-
1274042 01370b90: 00 00 00 2d 00 00 00 00 00 00 00 00 00 00 00 01  ...-............
1274043 01370ba0: 01 00 00 00 00 00 05 07 00 00 00 41 00 4e 00 4f  ...........A.N.O
```

Normal text file | length : 99634176 | lines : 1310977 | Ln : 1274016 | Col : 35 | Sel : 0 | 0 | UNIX | UTF-8 | IN

- Experiments here with images.

Use case

Principle

- Lots of similarity queries to perform on a high-dimensional dataset
- Speedup: similarity search directly on the binary codes
- Online computation of the transformation on the first data points: easier to process + gain in memory

Example

- CIFAR-10 60000 960-D GIST descriptors each (similar results for GIST1M dataset)
- Space cost: $960 \times 60000 \times 8\ o \approx 460\ Mo$ vs $60000/8 \times c = 7500 \times c$ octets
- Time cost: Euclidean distance vs Hamming distance for binary codes

Evaluation in the online setting [Gong et al., 2013, Leng et al., 2015]

- 1000 queries randomly sampled and the remaining data as training set
- Offline: Compute the Euclidean distance between all query and training points to determine the ground truth
- Online: Process one point at a time to update the transformation $\tilde{\mathbf{W}}_t$
- Apply the hashing scheme $\mathbf{B} = \mathrm{sign}(\tilde{\mathbf{W}}_t \mathbf{X})$ on the whole training set
- Compute the Hamming distance between all query points and training points and rank
- Compare the ranked training points to the ground truth with Mean Average Precision (mAP)
- 3 tested online baselines with hashing scheme $\Phi(\mathbf{x}_t) = \mathrm{sign}(\tilde{\mathbf{W}}_t \mathbf{x}_t)$

Thank you for your attention!

anne.morvan@cea.fr

↓

amorvan@expedia.com