Near-Optimal Dimension Reduction for Facility Location

Lingxiao Huang [*] Shaofeng Jiang [†] Robert Krauthgamer [‡]
Di Yue [†]

June 2025

[*] Nanjing University
[†] Peking University
[‡] Weizmann Institute of Science

Dimension Reduction

Challenge: curse of dimensionality

Solution: dimension reduction

- Goal: find a mapping $\pi \colon \mathbb{R}^d \to \mathbb{R}^m$, where $m \ll d$
- Example: Johnson-Lindenstrauss transform

Dimension reduction in algorithm design

(1) Embed the high-dimensional data to a low-dimensional space: $X \subset \mathbb{R}^d \overset{\pi}{\mapsto} \pi(X) \subset \mathbb{R}^m$

(2) Compute the problem on the low-dimensional data $\pi(X)$

(3) Map the solution back to X

Dimension Reduction

Johnson-Lindenstrauss Lemma [JL84]

There exists a random map $\pi\colon \mathbb{R}^d \to \mathbb{R}^m$ for $m = O(\varepsilon^{-2}\log n)$, such that for every n-point set $X \subset \mathbb{R}^d$, w.h.p.

$$\forall x, y \in X, \qquad \|\pi(x) - \pi(y)\| \in (1 \pm \varepsilon)\, \|x - y\|\,.$$

Dimension Reduction

Johnson-Lindenstrauss Lemma [JL84]

There exists a random map $\pi\colon \mathbb{R}^d \to \mathbb{R}^m$ for $m = O(\varepsilon^{-2} \log n)$, such that for every n-point set $X \subset \mathbb{R}^d$, w.h.p.

$$\forall x, y \in X, \qquad \|\pi(x) - \pi(y)\| \in (1 \pm \varepsilon) \|x - y\| .$$

Good properties: linear, data oblivious

- Linear: $\pi\colon x \mapsto \frac{1}{\sqrt{m}} Gx$ (G is the Gaussian matrix)
- Data oblivious: many applications in streaming

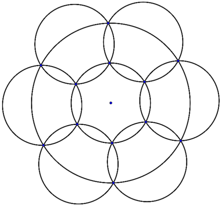
Limitations: target dimension $m = O(\varepsilon^{-2} \log n)$ is tight [LN17]

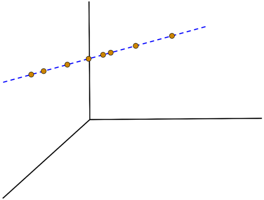
- TSP in $\mathbb{R}^{\Theta(\log n)}$ does not admit a PTAS [Tre00]
- Streaming MST in $\mathbb{R}^{\Theta(\log n)}$ requires $\Omega(\sqrt{n})$ bits [CCJ+23]

Doubling Dimension

Going beyond $O(\varepsilon^{-2}\log n)$: seek dependence on the intrinsic dimension of datasets

- High ambient dimension vs low intrinsic dimension
  - Points in a linear subspace
  - Points with sparse vector representation





- Doubling dimension $\mathrm{ddim}(X)$ [GKL03]: minimum $t \geq 0$, such that every ball in X can be covered by at most 2^t balls of half the radius
  - If $|X| = n$, then $\mathrm{ddim}(X) \leq \log n$

Dimension Reduction Meets Doubling Dimension

Goal: refine JL lemma, such that $m = m(\varepsilon, \mathrm{ddim}(X))$?

- Remains open
- Not possible for linear maps [IN07]

Dimension Reduction Meets Doubling Dimension

Goal: refine JL lemma, such that $m = m(\varepsilon, \mathrm{ddim}(X))$?

- Remains open
- Not possible for linear maps [IN07]

Goal: JL dimension reduction for specific problems, such that $m = m(\varepsilon, \mathrm{ddim}(X))$ and $\mathrm{opt}(\pi(X)) \in (1 \pm \varepsilon)\mathrm{opt}(X)$

- A weaker requirement than preserving pairwise distances

Dimension Reduction for Specific Problems

Problems	Apx.	Target Dimension	Ref.
Max-Cut	$1+\varepsilon$	$\tilde{O}(\varepsilon^{-2})$	CJK23
k-Median/Means	$1+\varepsilon$	$O(\varepsilon^{-2}\log k)$	MMR19
k-Subspace Apx.	$1+\varepsilon$	$\tilde{O}(\varepsilon^{-3}k^2)$	CW25
Nearest Neighbor	$1+\varepsilon$	$\tilde{O}(\varepsilon^{-2}\mathrm{ddim})$	IN07
k-Center Clustering	$1+\varepsilon$	$O(\varepsilon^{-2}(\mathrm{ddim}+\log k))$	JKS24
MST	$1+\varepsilon$	$\tilde{O}(\varepsilon^{-2}(\mathrm{ddim}+\log\log n))$	NSIZ21
UFL	$O(1)$	$O(\mathrm{ddim})$	NSIZ21
UFL	$1+\varepsilon$	$\tilde{O}(\varepsilon^{-2}\mathrm{ddim})$	This work

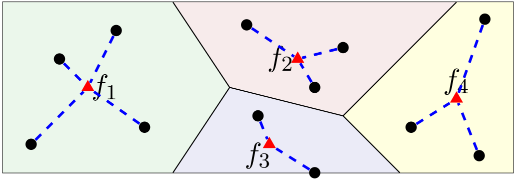
Uniform Facility Location (UFL)

Input: n-point set $X \subset \mathbb{R}^d$

Objective: find a facility set $F \subset \mathbb{R}^d$, to minimize

$$\mathrm{cost}(X, F) := |F| + \sum_{x \in X} \mathrm{dist}(x, F)$$

Optimal value: $\mathrm{ufl}(X) := \min_{F \subset \mathbb{R}^d} \mathrm{cost}(X, F)$



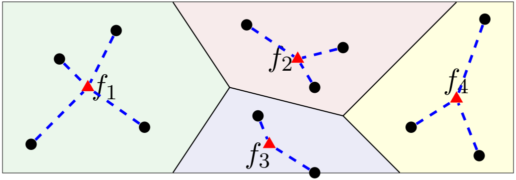
Uniform Facility Location (UFL)

Input: n-point set $X \subset \mathbb{R}^d$

Objective: find a facility set $F \subset \mathbb{R}^d$, to minimize

$$\mathrm{cost}(X, F) := |F| + \sum_{x \in X} \mathrm{dist}(x, F)$$

Optimal value: $\mathrm{ufl}(X) := \min_{F \subset \mathbb{R}^d} \mathrm{cost}(X, F)$



Problem (Dimension reduction for UFL)

Given $\varepsilon \in (0,1)$, decide a target dimension $m = m(\varepsilon, \mathrm{ddim}(X))$, such that w.h.p. $\mathrm{ufl}(\pi(X)) \in (1 \pm \varepsilon)\,\mathrm{ufl}(X)$

Main Result: Dimension Reduction

Theorem (Dimension reduction for UFL)

Consider $m = \tilde{O}(\varepsilon^{-2}\mathrm{ddim})$. Then for every finite $X \subset \mathbb{R}^d$ with $\mathrm{ddim}(X) \le \mathrm{ddim}$,

$$\Pr[\mathrm{ufl}(\pi(X)) \in (1 \pm \varepsilon)\,\mathrm{ufl}(X)] \ge 0.99$$

Main Result: Dimension Reduction

Theorem (Dimension reduction for UFL)

Consider $m = \tilde{O}(\varepsilon^{-2}\mathrm{ddim})$. Then for every finite $X \subset \mathbb{R}^d$ with $\mathrm{ddim}(X) \le \mathrm{ddim}$,

$$\Pr[\mathrm{ufl}(\pi(X)) \in (1 \pm \varepsilon)\,\mathrm{ufl}(X)] \ge 0.99$$

- Improvements over previous results
  - [NSIZ21]: $O(1)$-apx, target dimension $m = O(\mathrm{ddim})$
  - [MMR19]: $(1+\varepsilon)$-apx, target dimension $m = O(\varepsilon^{-2}\log n)$
- Handles a regime between low and high dimension
  - Data: low doubling dimension
  - Facilities: high dimension

Corollary: Streaming Algorithm

Corollary (Streaming algorithm for UFL)

There is a streaming algorithm that, given as input a set $X \subseteq [\Delta]^d$ presented as a stream, and an upper bound ddim, uses space $\tilde{O}(d \cdot \mathrm{polylog}(\Delta) + (\varepsilon^{-1} \log \Delta)^{\tilde{O}(\mathrm{ddim})})$ and outputs w.h.p a $(1 + \varepsilon)$-apx to $\mathrm{ufl}(X)$.

Corollary: Streaming Algorithm

Corollary (Streaming algorithm for UFL)

There is a streaming algorithm that, given as input a set $X \subseteq [\Delta]^d$ presented as a stream, and an upper bound ddim, uses space $\tilde{O}(d \cdot \mathrm{polylog}(\Delta) + (\varepsilon^{-1} \log \Delta)^{\tilde{O}(\mathrm{ddim})})$ and outputs w.h.p a $(1+\varepsilon)$-apx to $\mathrm{ufl}(X)$.

- The first streaming algorithm for UFL that utilize the doubling dimension (generalization of [CLMS13])
- Break the $\Omega(\sqrt{n})$ barrier in [CJK+22]

Main Result: PTAS

Theorem (PTAS for UFL)

There exists a randomized algorithm that computes a $(1+\varepsilon)$-apprximation for UFL in time $(2^{m'}d + 2^{2^{m'}}) \cdot \tilde{O}(n)$, for

$$m' = O\Big(\mathrm{ddim}(X) \cdot \log(\mathrm{ddim}(X)/\varepsilon)\Big)$$

Main Result: PTAS

Theorem (PTAS for UFL)

There exists a randomized algorithm that computes a $(1+\varepsilon)$-apprximation for UFL in time $(2^{m'}d + 2^{2^{m'}}) \cdot \tilde{O}(n)$, for

$$m' = O\Big(\mathrm{ddim}(X) \cdot \log(\mathrm{ddim}(X)/\varepsilon)\Big)$$

- Facilities are allowed to be picked from the ambient space $\mathbb{R}^d$
  - [CFS21]: PTAS for UFL in time $2^{2^{O(\mathrm{ddim}^2)}} \cdot d \cdot \tilde{O}(n)$, with facilities restricted to the dataset

Technical Overview

Theorem (Dimension reduction for UFL)

Consider $m = \tilde{O}(\varepsilon^{-2}\mathrm{ddim})$. Then for every finite $X \subset \mathbb{R}^d$ with $\mathrm{ddim}(X) \leq \mathrm{ddim}$,

$$\Pr[\mathrm{ufl}(\pi(X)) \in (1 \pm \varepsilon)\,\mathrm{ufl}(X)] \geq 0.99$$

Technical Overview

Theorem (Dimension reduction for UFL)

Consider $m = \tilde{O}(\varepsilon^{-2}\mathrm{ddim})$. Then for every finite $X \subset \mathbb{R}^d$ with $\mathrm{ddim}(X) \le \mathrm{ddim}$,

$$\Pr[\mathrm{ufl}(\pi(X)) \in (1 \pm \varepsilon)\, \mathrm{ufl}(X)] \ge 0.99$$

- Easy direction: $\mathrm{ufl}(\pi(X)) \le (1+\varepsilon)\, \mathrm{ufl}(X)$
  - Let F^* be the optimal solution for X, then $\pi(F^*)$ is a feasible solution for $\pi(X)$
  - $\mathrm{ufl}(\pi(X)) \le \mathrm{cost}(\pi(X), \pi(F^*)) \lesssim \mathrm{cost}(X, F^*) = \mathrm{ufl}(X)$
  - Suffices to preserve the cost of one solution

Technical Overview

Theorem (Dimension reduction for UFL)

Consider $m = \tilde{O}(\varepsilon^{-2}\mathrm{ddim})$. Then for every finite $X \subset \mathbb{R}^d$ with $\mathrm{ddim}(X) \le \mathrm{ddim}$,

$$\Pr[\mathrm{ufl}(\pi(X)) \in (1 \pm \varepsilon)\,\mathrm{ufl}(X)] \ge 0.99$$

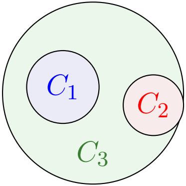
- Easy direction: $\mathrm{ufl}(\pi(X)) \le (1+\varepsilon)\,\mathrm{ufl}(X)$
  - Let F^* be the optimal solution for X, then $\pi(F^*)$ is a feasible solution for $\pi(X)$
  - $\mathrm{ufl}(\pi(X)) \le \mathrm{cost}(\pi(X), \pi(F^*)) \lesssim \mathrm{cost}(X, F^*) = \mathrm{ufl}(X)$
  - Suffices to preserve the cost of one solution
- Hard direction: $\mathrm{ufl}(\pi(X)) \ge (1-\varepsilon)\,\mathrm{ufl}(X)$
  - Optimal solution F^*_π for $\pi(X)$ is random
  - Need to preserve the cost of all solutions
  - Idea: metric decomposition

Technical Overview: Decomposition

Construct a partition $\mathbf{\Lambda}$ for X, s.t. for parameter $\kappa = \Theta(\mathrm{ddim}/\varepsilon)^{\Theta(\mathrm{ddim})}$

(a) Every cluster $C \in \mathbf{\Lambda}$ satisfies $\mathrm{ufl}(C) = \Theta(\kappa)$

(b) $\sum_{C \in \mathbf{\Lambda}} \mathrm{ufl}(C) \in (1 \pm \varepsilon) \cdot \mathrm{ufl}(X)$



$\mathbf{\Lambda} = \{C_1, C_2, C_3\}$

- Property (a): κ determines the target dimension $m = O(\varepsilon^{-2} \log \kappa) = \tilde{O}(\varepsilon^{-2}\mathrm{ddim})$
- Property (b): $(1 + \varepsilon)$-apx

Technical Overview

Step 1 Construct a partition $\mathbf{\Lambda}$ for X, s.t. for parameter $\kappa = \Theta(\mathrm{ddim}/\varepsilon)^{\Theta(\mathrm{ddim})}$

(a) Every cluster $C \in \mathbf{\Lambda}$ satisfies $\mathrm{ufl}(C) = \Theta(\kappa)$

(b) $\sum_{C\in\mathbf{\Lambda}} \mathrm{ufl}(C) \in (1 \pm \varepsilon) \cdot \mathrm{ufl}(X)$

Step 2 $\mathrm{ufl}(\pi(X)) \geq \sum_{C\in\mathbf{\Lambda}} \mathrm{ufl}(\pi(C)) - \varepsilon \cdot \mathrm{ufl}(X)$

- Property (b) carries over to the target space

Step 3 $\sum_{C\in\mathbf{\Lambda}} \mathrm{ufl}(\pi(C)) \geq (1 - \varepsilon) \sum_{C\in\mathbf{\Lambda}} \mathrm{ufl}(C)$

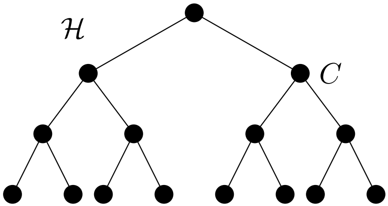
- Apply k-median results in [MMR19] to every cluster $C \in \mathbf{\Lambda}$
- Target dimension $m = O(\varepsilon^{-2} \log \kappa)$ suffices

Step 4 Apply property (b) $\sum_{C\in\mathbf{\Lambda}} \mathrm{ufl}(C) \geq \mathrm{ufl}(X)$

Our Decomposition Procedure

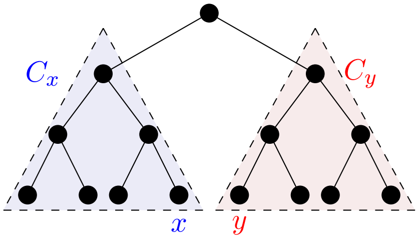
Hierarchical Decomposition [Tal04]

- A generalization of randomly shifted quadtree to doubling metrics



- Node $\leftrightarrow$ cluster, children $\subseteq$ parent
  - Root: X
  - Leaves: singletons
  - Level i: diameter $\Theta(2^i)$
- Each node (cluster) has $2^{O(\mathrm{ddim})}$ child nodes (clusters)

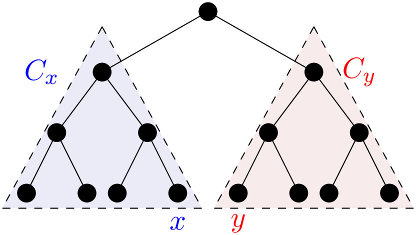
Cutting Probability



Cutting probability [Tal04]: $\forall x, y \in X$,

$\Pr_{\mathcal{H}}[x, y \text{ are in different clusters at level } i] \leq O(\mathrm{ddim}) \cdot \mathrm{dist}(x, y)/2^i$

Cutting Probability



Cutting probability [Tal04]: $\forall x, y \in X$,

$\Pr_{\mathcal{H}}[x, y$ are in different clusters at level $i] \le O(\mathrm{ddim}) \cdot \mathrm{dist}(x, y)/2^i$

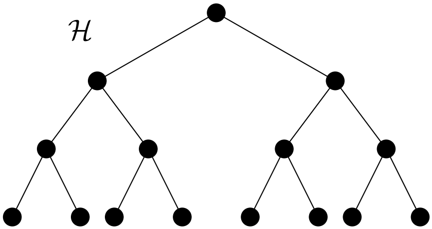
Badly-cut pair [CFS21]: say (x, y) is badly cut, if x, y are in different clusters at level $\log(\varepsilon^{-1}\mathrm{ddim} \cdot \mathrm{dist}(x, y))$

- $\Pr[(x, y)$ is badly cut$] \le O(\varepsilon)$

Property (a): Bottom-up Construction

Property (a): Every cluster $C \in \mathbf{\Lambda}$ satisfies $\mathrm{ufl}(C) = \Theta(\kappa)$

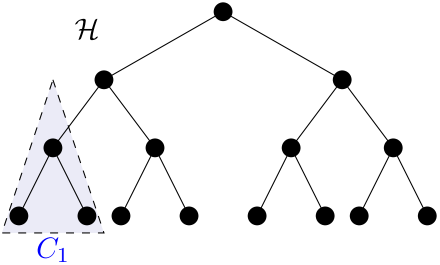
Given threshold $\kappa = \Theta(\mathrm{ddim}/\varepsilon)^{O(\mathrm{ddim})}$, find the lowest level "heavy cluster" $(\mathrm{ufl}(C) \geq \kappa)$ in a bottom-up manner



Property (a): Bottom-up Construction

Property (a): Every cluster $C \in \mathbf{\Lambda}$ satisfies $\mathrm{ufl}(C) = \Theta(\kappa)$

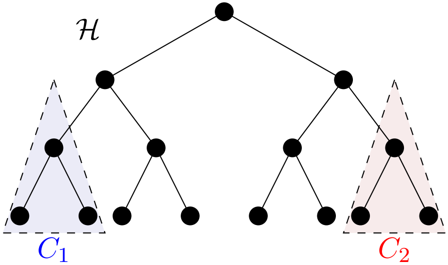
Given threshold $\kappa = \Theta(\mathrm{ddim}/\varepsilon)^{O(\mathrm{ddim})}$, find the lowest level "heavy cluster" $(\mathrm{ufl}(C) \geq \kappa)$ in a bottom-up manner



Property (a): Bottom-up Construction

Property (a): Every cluster $C \in \mathbf{\Lambda}$ satisfies $\mathrm{ufl}(C) = \Theta(\kappa)$

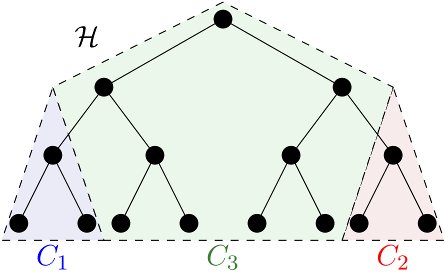
Given threshold $\kappa = \Theta(\mathrm{ddim}/\varepsilon)^{O(\mathrm{ddim})}$, find the lowest level "heavy cluster" ($\mathrm{ufl}(C) \geq \kappa$) in a bottom-up manner



Property (a): Bottom-up Construction

Property (a): Every cluster $C \in \mathbf{\Lambda}$ satisfies $\mathrm{ufl}(C) = \Theta(\kappa)$

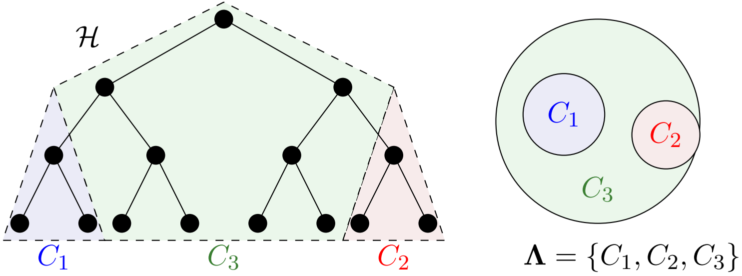
Given threshold $\kappa = \Theta(\mathrm{ddim}/\varepsilon)^{O(\mathrm{ddim})}$, find the lowest level "heavy cluster" ($\mathrm{ufl}(C) \geq \kappa$) in a bottom-up manner



Property (a): Bottom-up Construction

Property (a): Every cluster $C \in \mathbf{\Lambda}$ satisfies $\mathrm{ufl}(C) = \Theta(\kappa)$

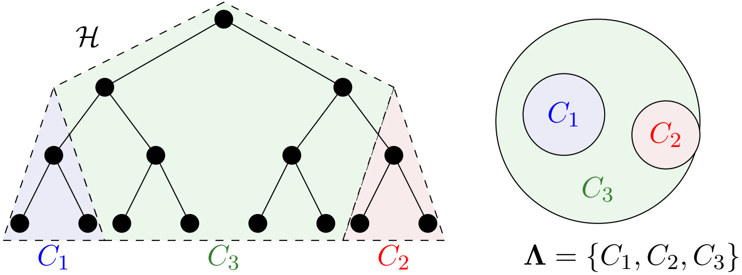
Given threshold $\kappa = \Theta(\mathrm{ddim}/\varepsilon)^{O(\mathrm{ddim})}$, find the lowest level "heavy cluster" $(\mathrm{ufl}(C) \geq \kappa)$ in a bottom-up manner



Property (a): Bottom-up Construction

Property (a): Every cluster $C \in \mathbf{\Lambda}$ satisfies $\mathrm{ufl}(C) = \Theta(\kappa)$

Given threshold $\kappa = \Theta(\mathrm{ddim}/\varepsilon)^{O(\mathrm{ddim})}$, find the lowest level "heavy cluster" $(\mathrm{ufl}(C) \geq \kappa)$ in a bottom-up manner



- Every cluster $C \in \mathbf{\Lambda}$ satisfies $\kappa \leq \mathrm{ufl}(C) \leq 2^{O(\mathrm{ddim})}\kappa$
- Previous top-down construction [CLMS13]: $\mathrm{polylog}(n)$ upper bound
  - Extra $\log \log n$ factor in the target dimension

Property (b): Proof Idea

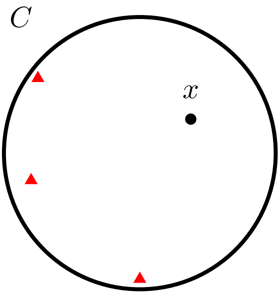
Property (b): $\underbrace{\sum_{C\in\mathbf{\Lambda}} \mathrm{ufl}(C)}_{\text{local}} \leq (1+\varepsilon)\cdot \underbrace{\mathrm{ufl}(X)}_{\text{global}}$

Idea: adding extra facilities to F^* to serve each $C \in \mathbf{\Lambda}$ locally

- $F' := F^* \cup (\bigcup_{C\in\mathbf{\Lambda}} \underbrace{N_C}_{\text{a net on } C})$
- Connection cost: $\forall C \in \mathbf{\Lambda}, \forall x \in C$, $\mathrm{dist}(x, F' \cap C) \leq (1+\varepsilon)\,\mathrm{dist}(x, F^*)$
- Opening cost: $|F'| - |F^*| \leq \varepsilon \sum_{C\in\mathbf{\Lambda}} \mathrm{ufl}(C)$

Property (b): Connection Cost

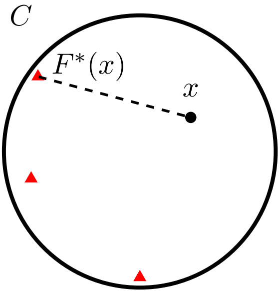
Goal: $\mathrm{dist}(x, F' \cap C) \leq (1+\varepsilon)\,\mathrm{dist}(x, F^*)$



Let F^* be the optimal solution for X and assume $F^* \subseteq X$ for simplicity

Property (b): Connection Cost

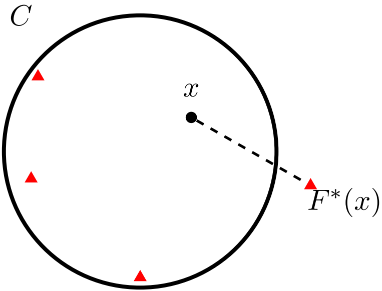
Goal: $\operatorname{dist}(x, F' \cap C) \leq (1+\varepsilon) \operatorname{dist}(x, F^*)$



If $F^*(x) \in C$, then ✓

Property (b): Connection Cost

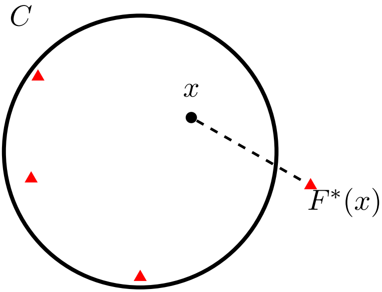
Goal: $\mathrm{dist}(x, F' \cap C) \leq (1+\varepsilon)\,\mathrm{dist}(x, F^*)$



What if $F^*(x) \notin C$?

Property (b): Connection Cost

Goal: $\mathrm{dist}(x, F' \cap C) \leq (1+\varepsilon)\,\mathrm{dist}(x, F^*)$

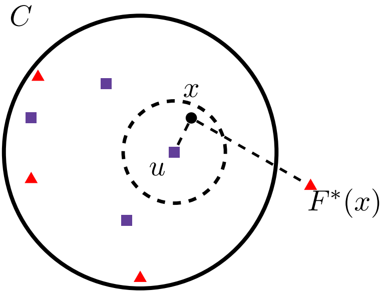


Key observation: separation property

- Badly-cut pair: say (x, y) is badly cut, if x, y are in different clusters at level $\log(\varepsilon^{-1}\mathrm{ddim} \cdot \mathrm{dist}(x, y))$
- Separation property: if $(x, F^*(x))$ is not badly cut, then $x \in C,\ F^*(x) \notin C \implies \mathrm{dist}(x, F^*(x)) \geq \varepsilon \cdot \mathrm{diam}(C)$
  - An $\varepsilon \cdot \mathrm{diam}(C)$-net N_C suffices

Property (b): Connection Cost

Goal: $\mathrm{dist}(x, F' \cap C) \le (1+\varepsilon)\,\mathrm{dist}(x, F^*)$

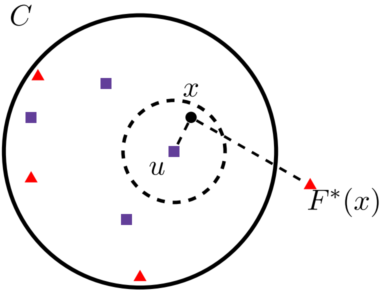


Key observation: separation property

- Badly-cut pair: say (x, y) is badly cut, if x, y are in different clusters at level $\log(\varepsilon^{-1}\mathrm{ddim} \cdot \mathrm{dist}(x, y))$
- Separation property: if $(x, F^*(x))$ is not badly cut, then $x \in C,\ F^*(x) \notin C \Longrightarrow \mathrm{dist}(x, F^*(x)) \ge \varepsilon \cdot \mathrm{diam}(C)$
  - An $\varepsilon \cdot \mathrm{diam}(C)$-net N_C suffices

Property (b): Opening Cost

Goal: $|F'| - |F^*| \leq \varepsilon \sum_{C \in \mathbf{\Lambda}} \mathrm{ufl}(C)$



- $|N_C|$ is small: $|N_C| \leq (1/\varepsilon)^{O(\mathrm{ddim})} \leq \varepsilon\kappa \leq \varepsilon \cdot \mathrm{ufl}(C)$
- $|F'| - |F^*| = \sum_{C \in \mathbf{\Lambda}} |N_C| \leq \varepsilon \sum_{C \in \mathbf{\Lambda}} \mathrm{ufl}(C)$

Eliminate Badly-Cut Pairs

Remaining issue: what if $(x, F^*(x))$ is badly-cut?

Prior solution [CFS21]

- Move x to $F^*(x)$, creating a new instance X'
  - Algorithmically, only feasible to eliminate badly-cut pairs $(x, F_0(x))$ for some approximation solution F_0
- X' is random (randomness comes from $\mathcal{H}$)
- Hard to deal with remaining badly-cut pairs $(x, F^*(x))$

Eliminate Badly-Cut Pairs

Remaining issue: what if $(x, F^*(x))$ is badly-cut?

Prior solution [CFS21]

- Move x to $F^*(x)$, creating a new instance X'
  - Algorithmically, only feasible to eliminate badly-cut pairs $(x, F_0(x))$ for some approximation solution F_0
- X' is random (randomness comes from $\mathcal{H}$)
- Hard to deal with remaining badly-cut pairs $(x, F^*(x))$

Our approach: modify $\mathcal{H}$ instead of X

- At each level, force close points to be clustered together
- Construct $\mathbf{\Lambda}$ accordingly
- Can still apply cutting probability to handle remaining badly-cut pairs

Thank you!