Optimal Decision Trees for Interpretable Clustering with Constraints

POUYA SHATI, ELDAN COHEN, SHEILA MCILRAITH

UNIVERSITY OF TORONTO

VECTOR INSTITUTE FOR ARTIFICIAL INTELLIGENCE

IJCAI-23

AUGUST 2023

Overview

- **Decision trees** as **interpretable** clustering solutions
  - usually found via **local search heuristics**
  - no **exact optimization** nor **support for constraints**

- **Our contribution:** the first exact optimization approach
  - **MaxSAT-based** encoding allows **optimality** and **constraint support**
  - finds ϵ**-approximation** of a well-studied **bi-criteria** objective

- Our experiments show
  - **tree clustering** outperforms **state-of-the-art** non-tree clustering in **ARI** scores
  - the **bi-criteria** objective complements tree clustering
  - tree solutions are well-suited to **benefit from constraints**

Background

Encoding

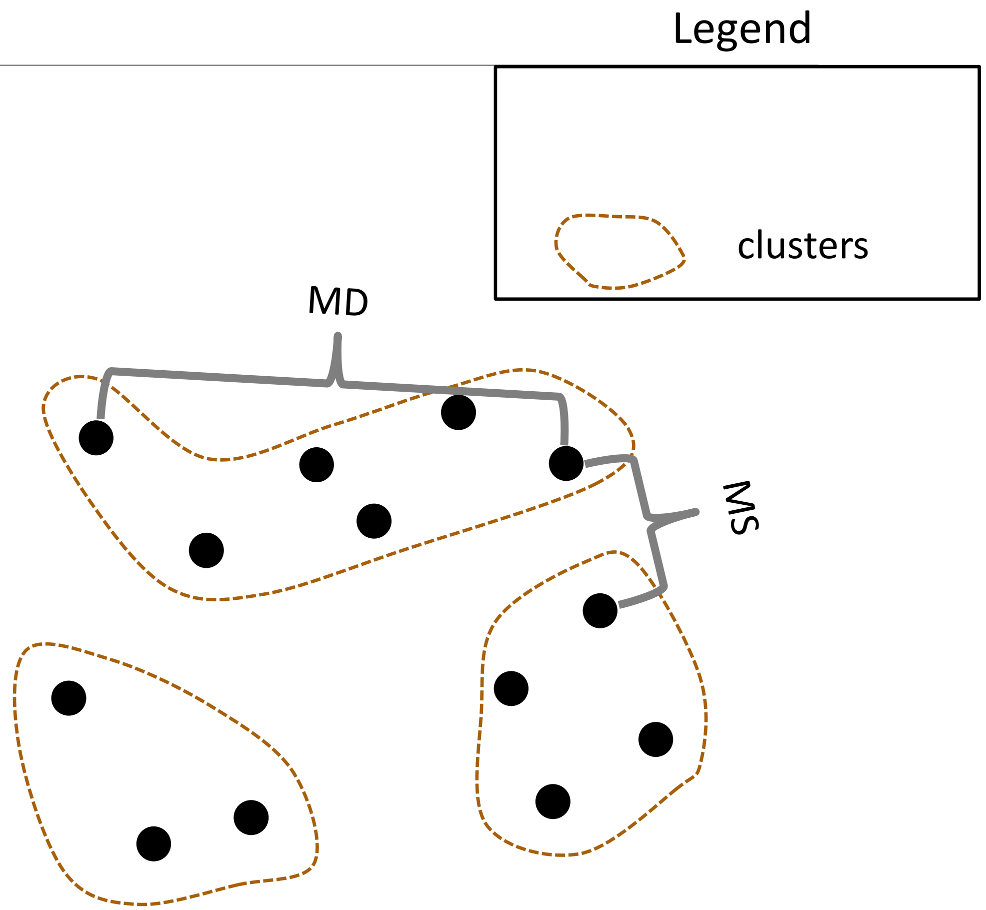
Experiments

- Constrained clustering
- Decision trees
- Tree clustering
- MaxSAT

Constrained Clustering

- A semi-supervised machine learning task

- Bi-criteria objective:
  - **maximize minimum split (MS)** between clusters
  - **minimize maximum diameter (MD)** within clusters

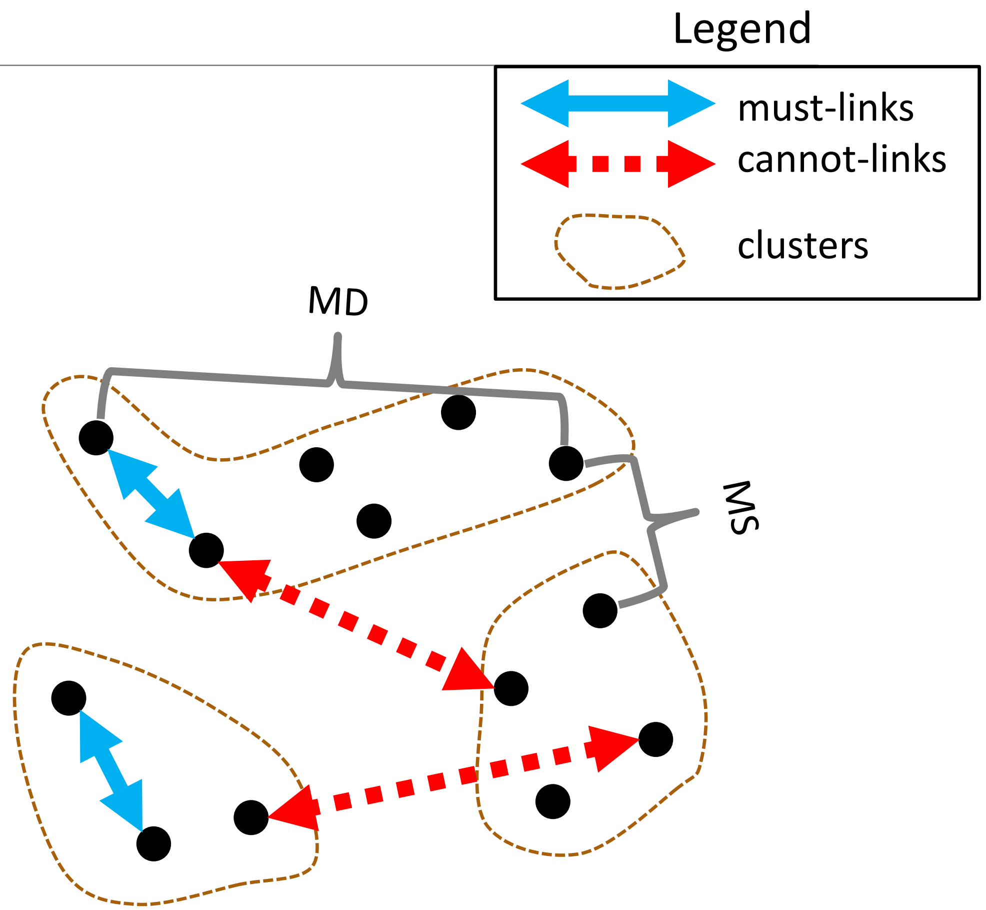


Constrained Clustering

- A semi-supervised machine learning task

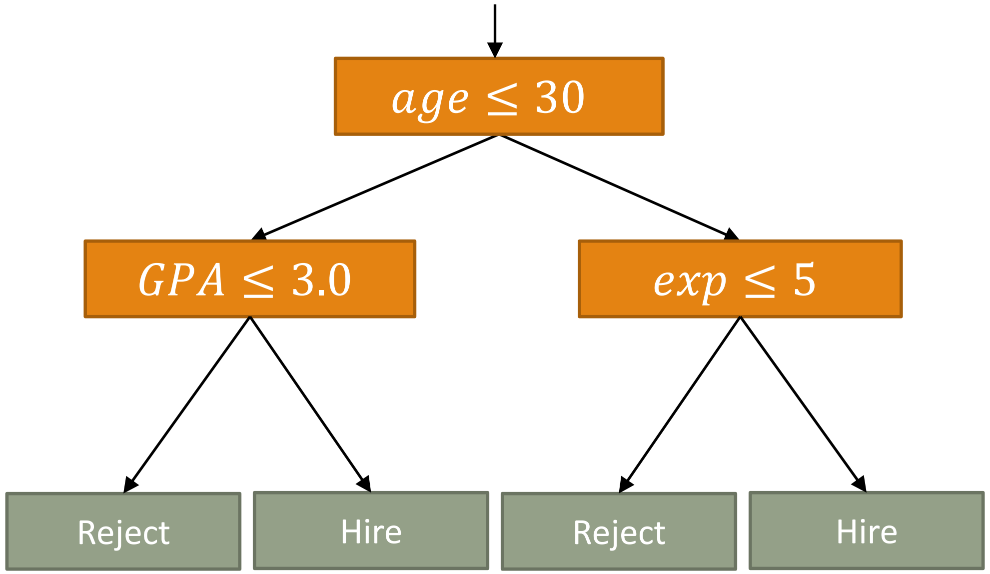
- Bi-criteria objective:
  - **maximize minimum split (MS)** between clusters
  - **minimize maximum diameter (MD)** within clusters

- **Domain-Independent** Constraints:
  - **must-links:** pairs that should be in the same cluster
  - **cannot-links:** pairs that should be in different clusters



Decision Trees

- **Decision trees:**
  - feature selection
  - threshold selection
  - leaf labelling

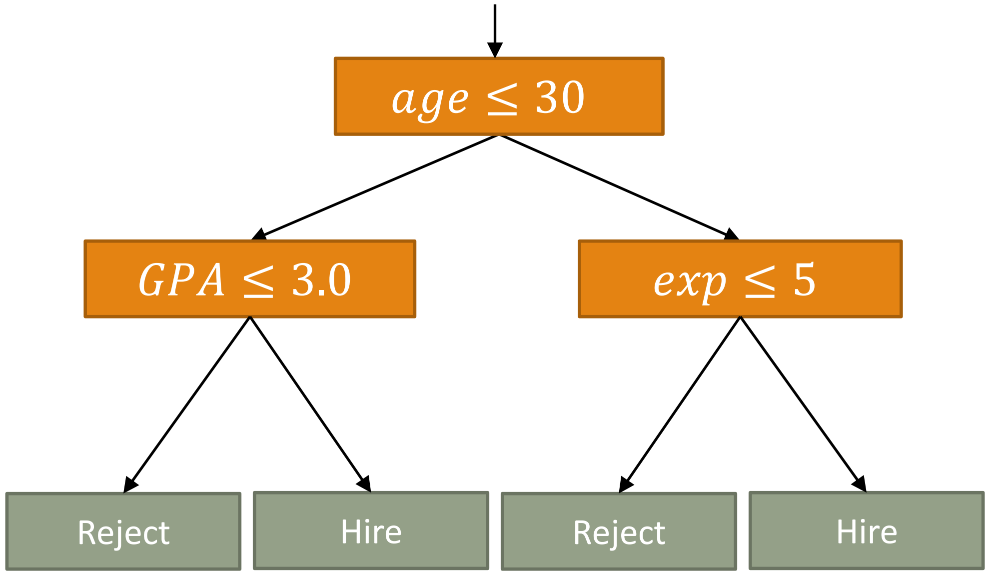


Decision Trees

- **Decision trees:**
  - feature selection
  - threshold selection
  - leaf labelling

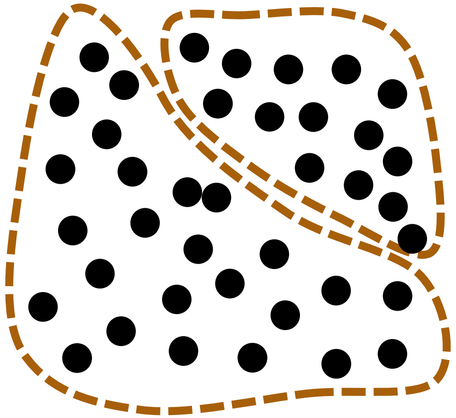
- They are **interpretable**:
  - yet competitive in **accuracy**

- Traditionally used for **classification**



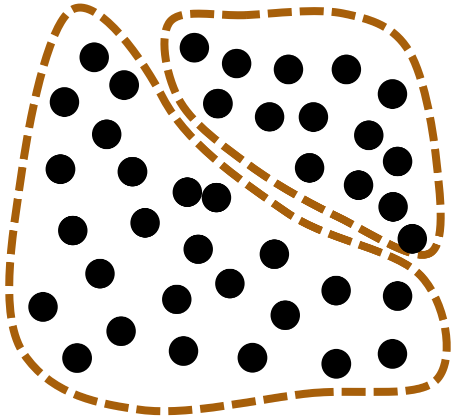
Tree Clustering

Non-tree

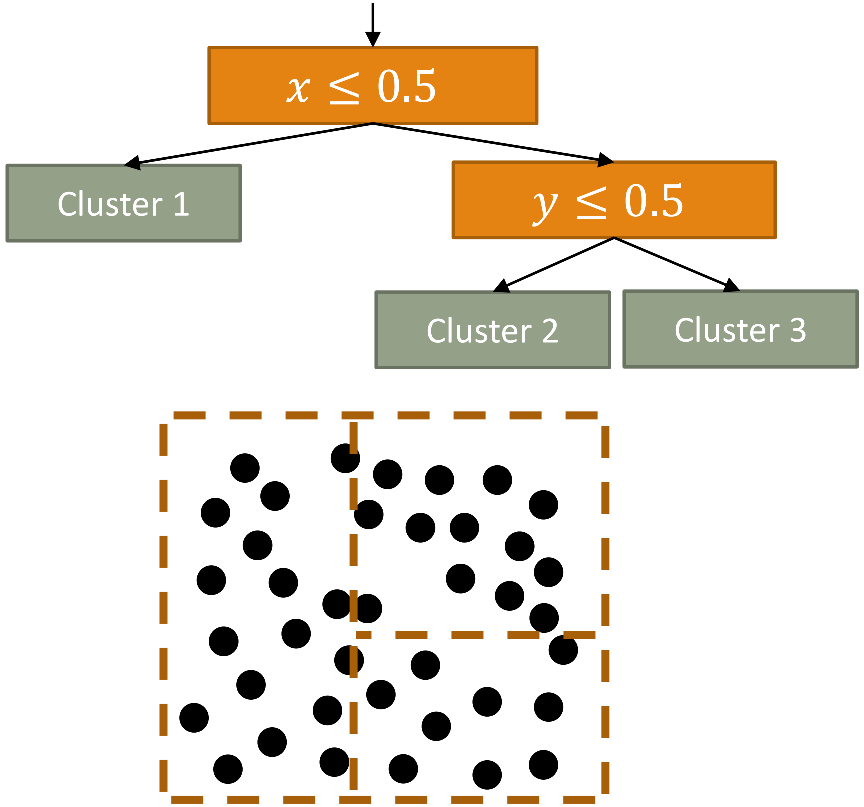


Tree Clustering

Non-tree

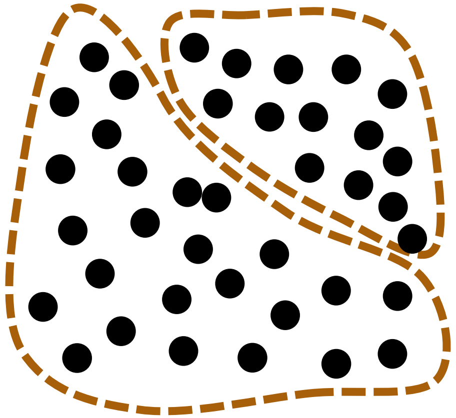


One cluster per leaf

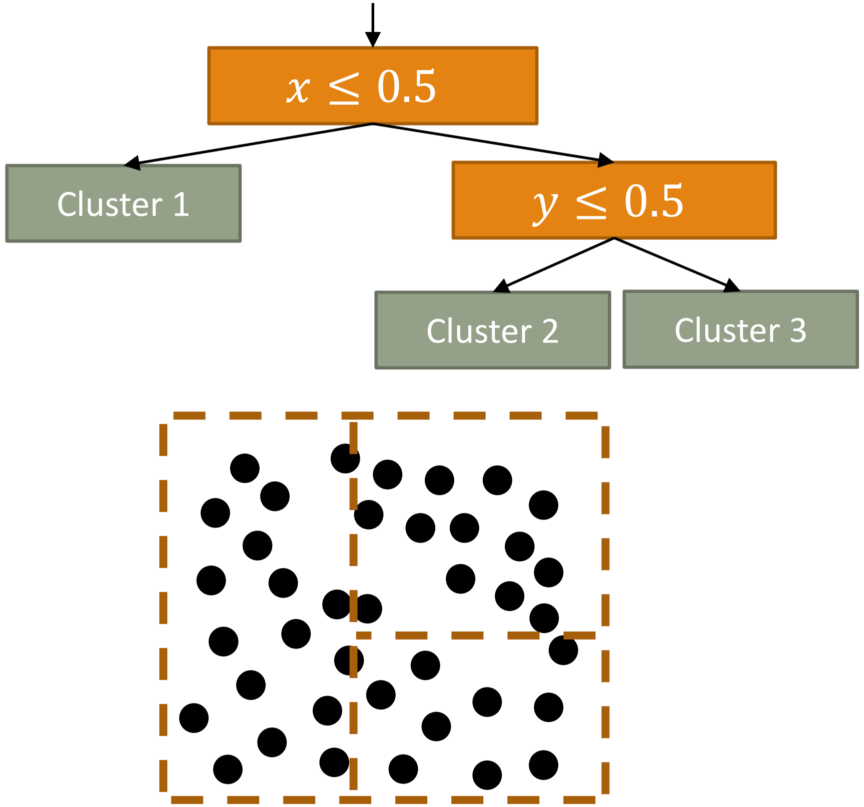


Tree Clustering

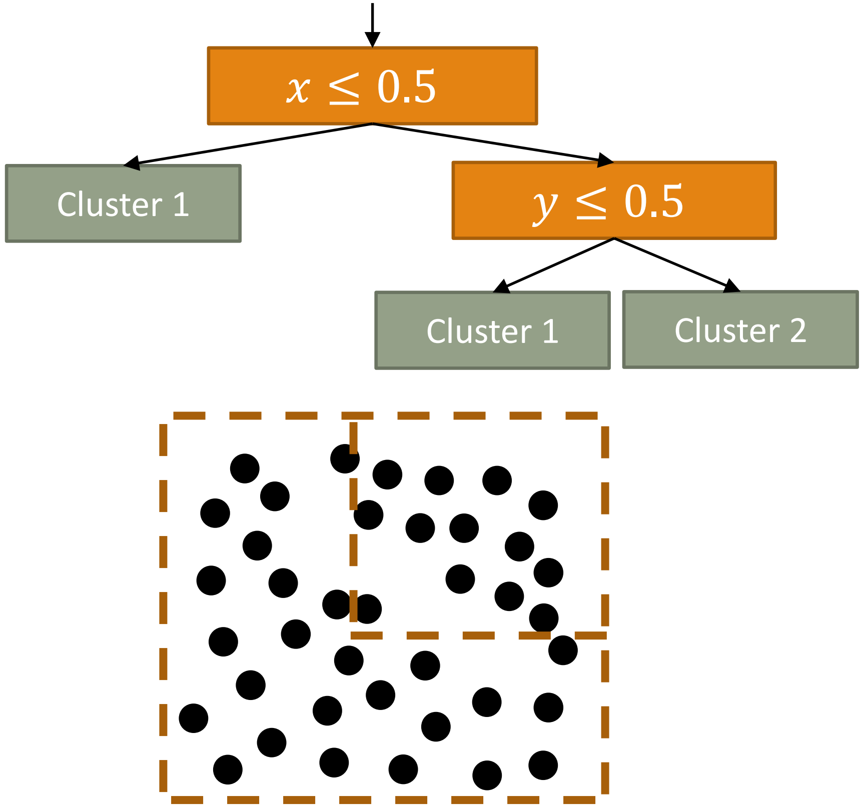
Non-tree



One cluster per leaf



Multi-leaf clusters



MaxSAT

- A set of binary variables $\mathcal{X} = \{x_0, x_1, \dots, x_n\}$
- A clause C_i is a subset of literals $\mathcal{X} \cup \neg\mathcal{X}$

- Satisfy all **hard** clauses $\mathcal{C}_h$
- Maximize the number of satisfied **soft** clauses $\mathcal{C}_s$

- Find an assignment $\mathcal{M}: \mathcal{X} \rightarrow \{false, true\}$

Background

Encoding

Experiments

- Basis
- Approximating objective
- Smart pairs

Encoding Basis

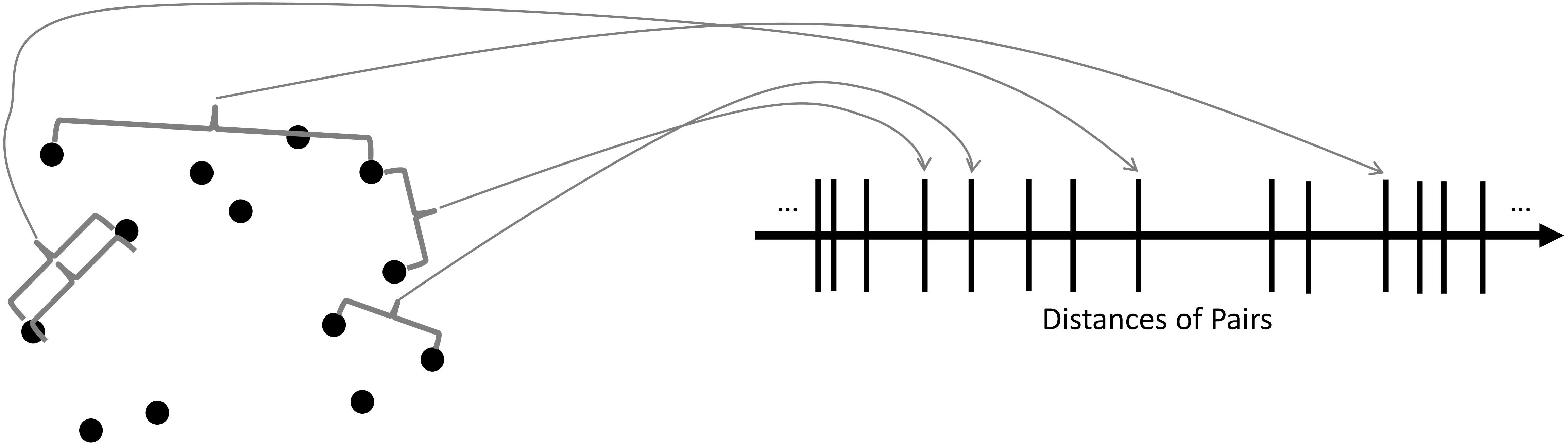
- Based on our previous work on decision tree classifiers

  [Shati, Cohen, McIlraith, CP2021]

- How to extend the encoding for constrained clustering:
  - model ϵ-approximation of the two objectives
  - support for pairwise constraints

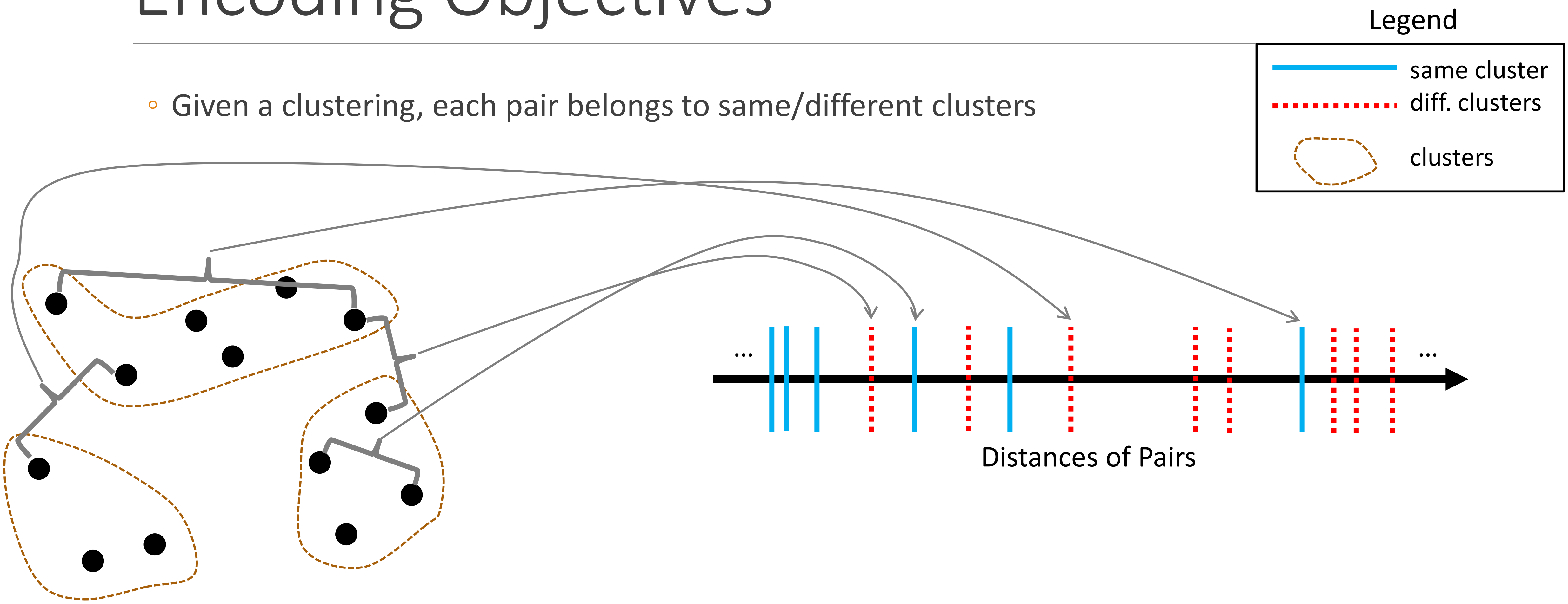
Encoding Objectives

- Our objectives involve sorting distances of pairs



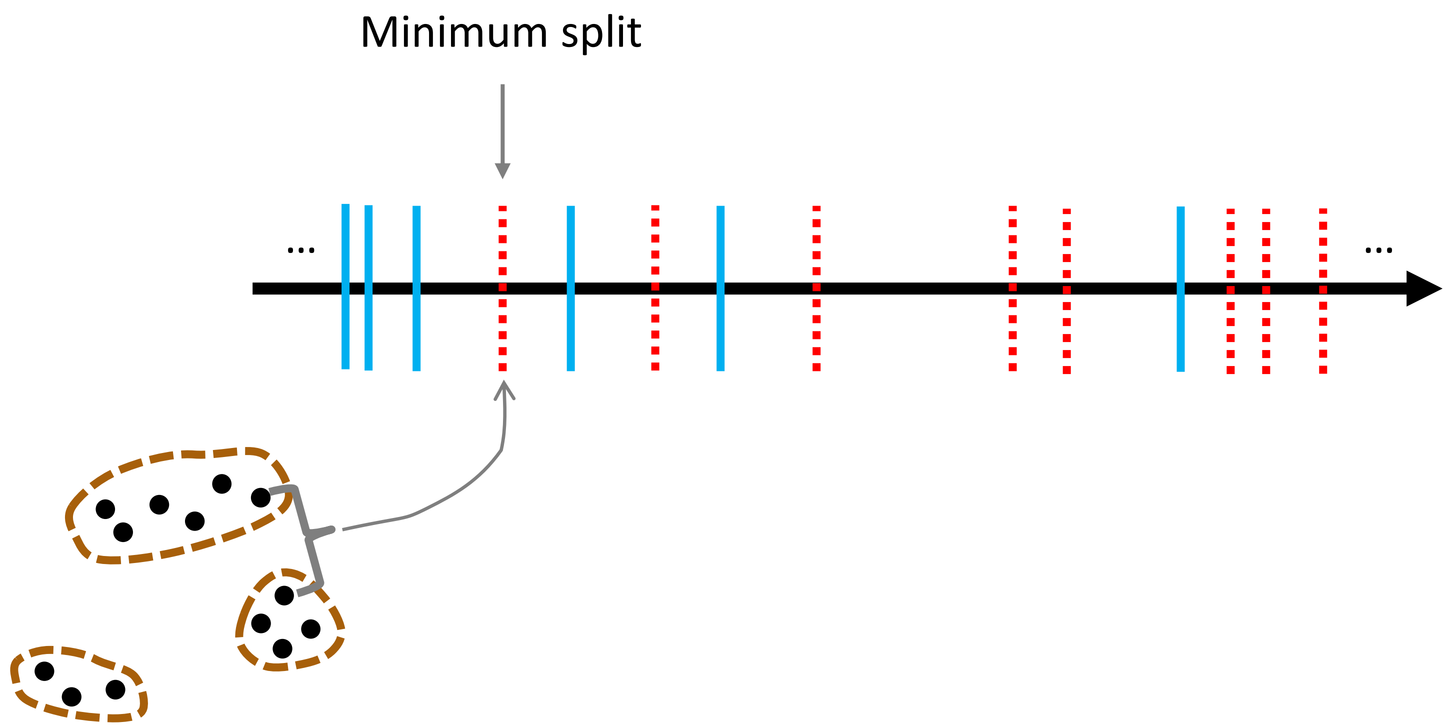
Encoding Objectives

- Given a clustering, each pair belongs to same/different clusters



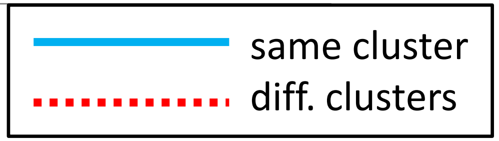
Encoding Objectives

- Minimum split and maximum diameter are points along the axis

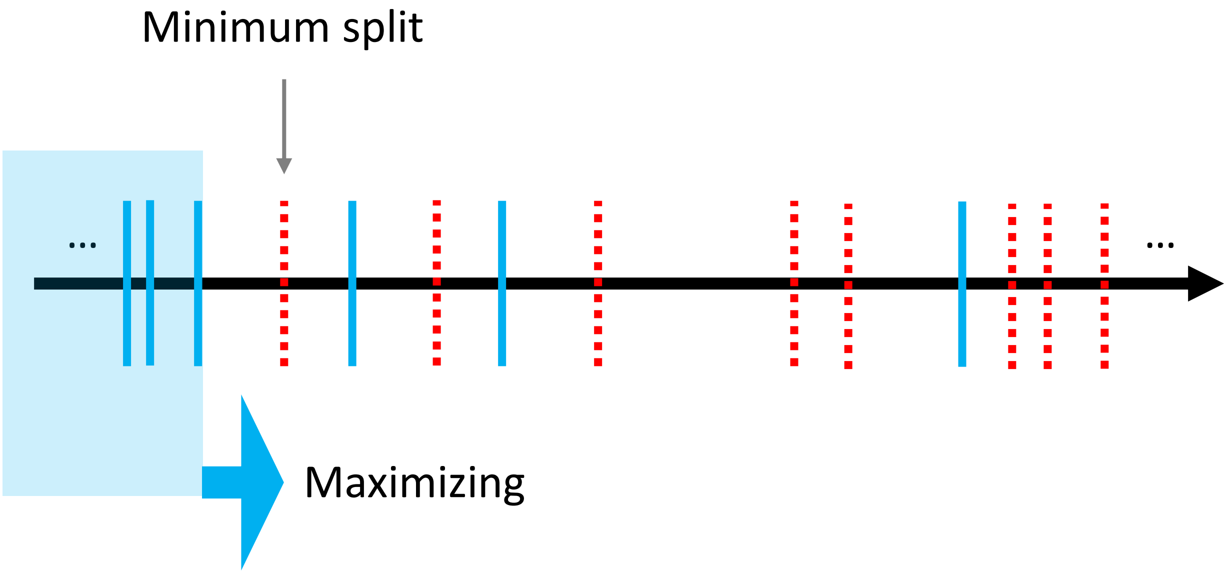


Encoding Objectives

Legend

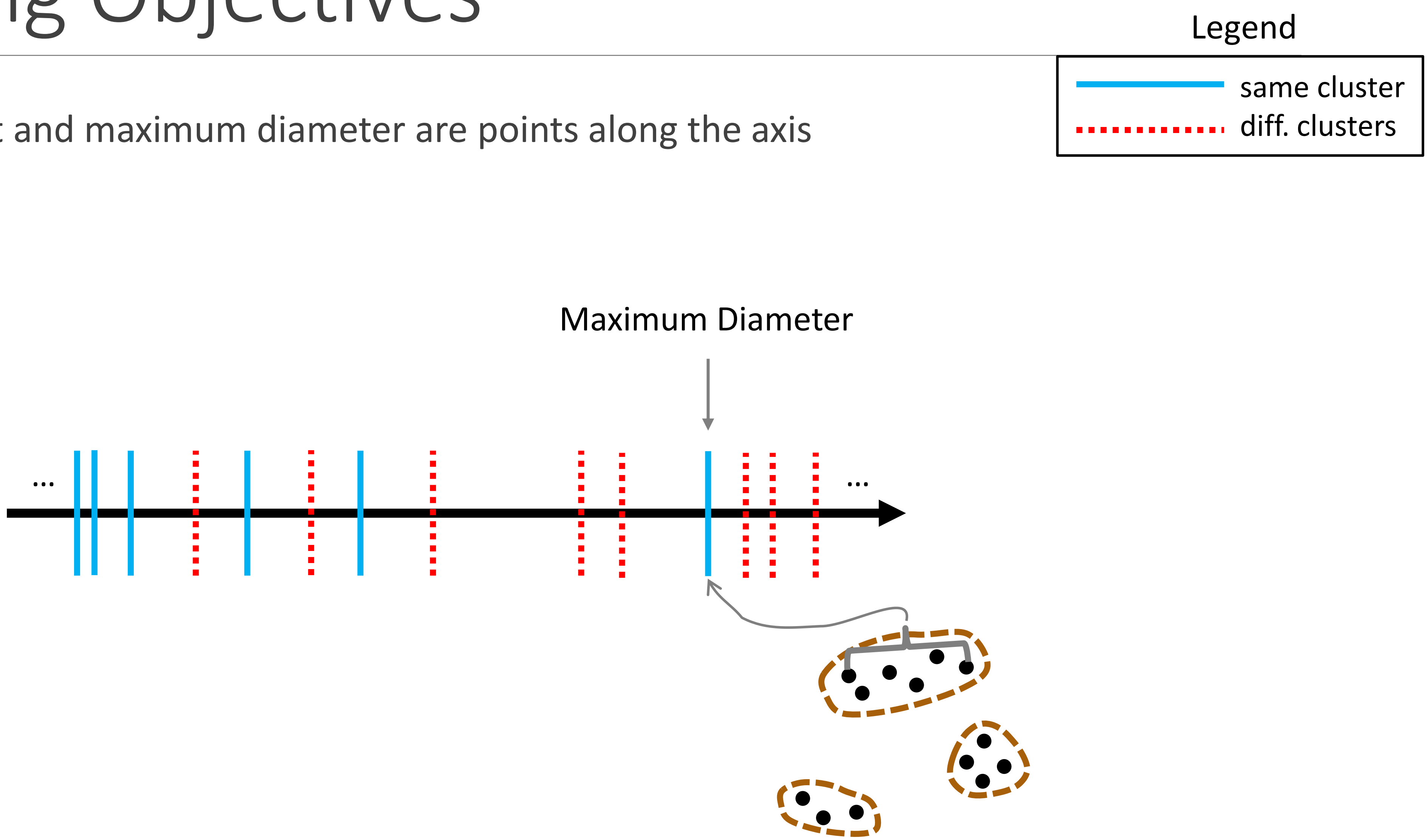


- Minimum split and maximum diameter are points along the axis



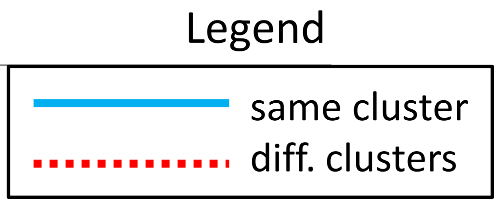
Encoding Objectives

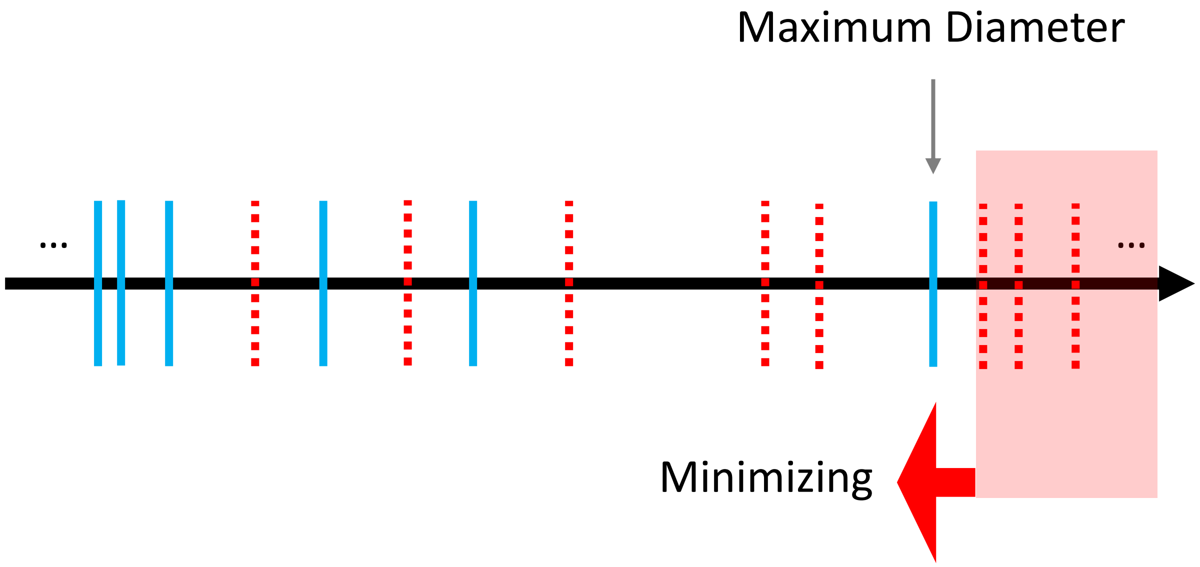
- Minimum split and maximum diameter are points along the axis



Encoding Objectives

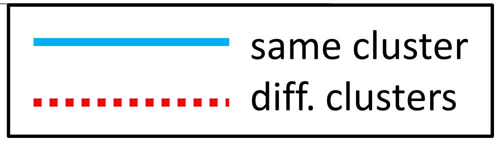
- Minimum split and maximum diameter are points along the axis



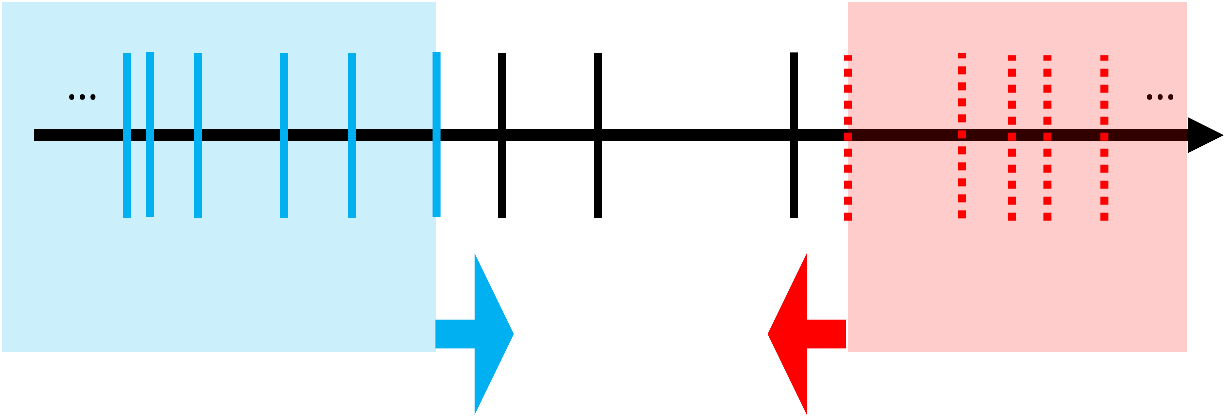


Encoding Objectives

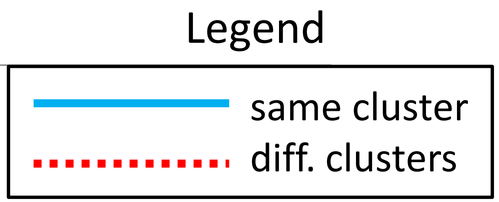
Legend



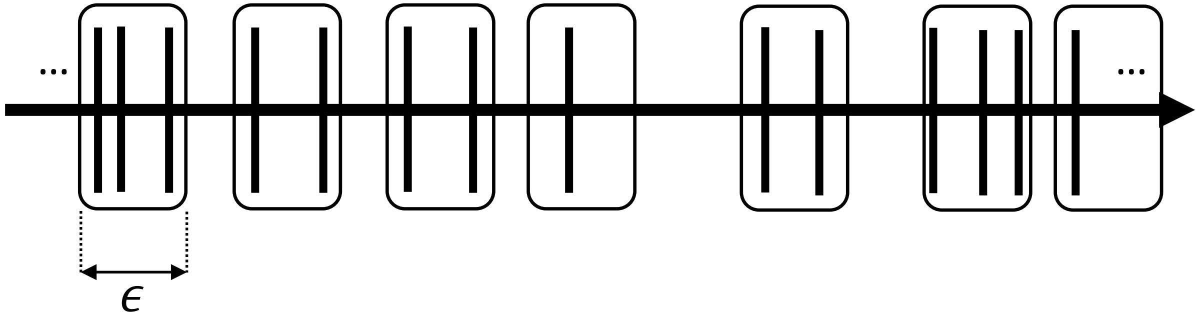
- Optimize the two objectives simultaneously to get Pareto optimality



Encoding Objectives

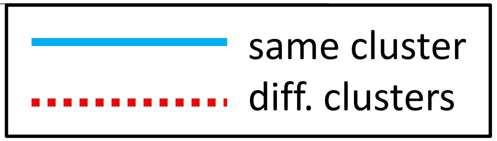


- Use **distance classes** instead of individual pairs

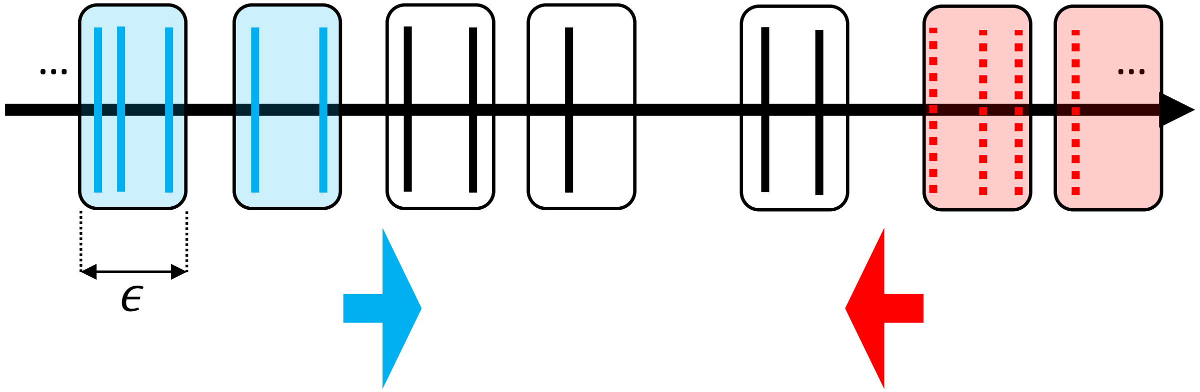


Encoding Objectives

Legend



- Use **distance classes** instead of individual pairs



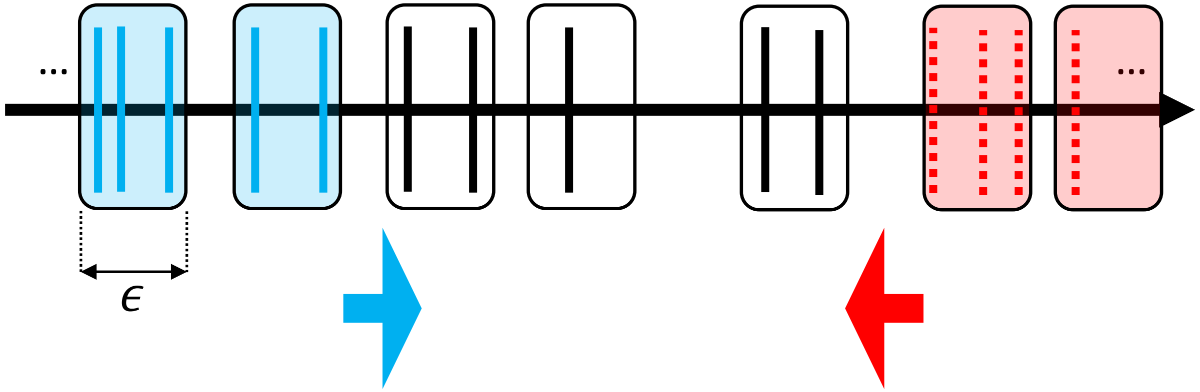
Encoding Objectives

Legend

same cluster

diff. clusters

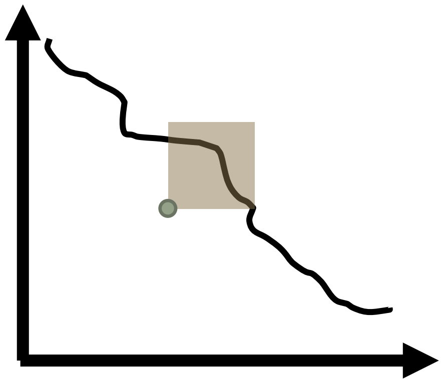
- Use **distance classes** instead of individual pairs



A Pareto-optimal solution in the number of classes

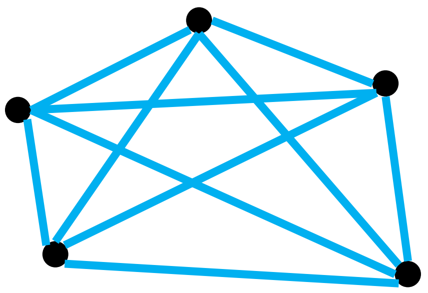
In ϵ-neighborhood of

A Pareto-optimal solution in Max Diameter and Min Split



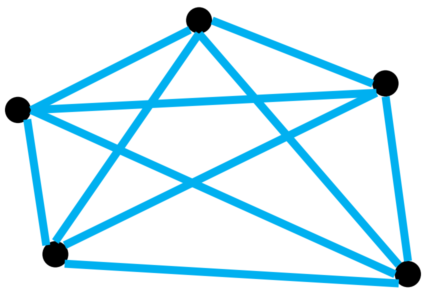
Smart Pairs

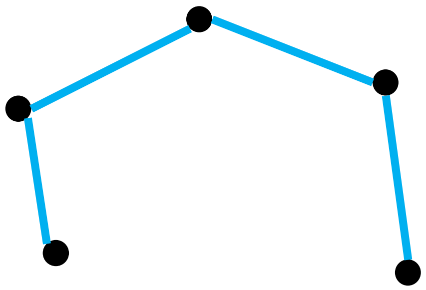
- **Quadratic** number of clauses for naively enforcing must-links



Smart Pairs

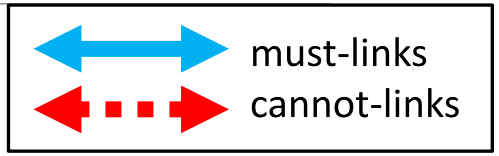
- **Quadratic** number of clauses for naively enforcing must-links
- But only a **linear** number of edges is needed for connecting all points





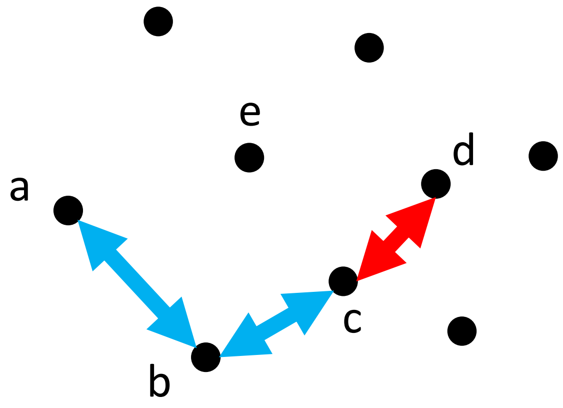
Smart Pairs

Legend



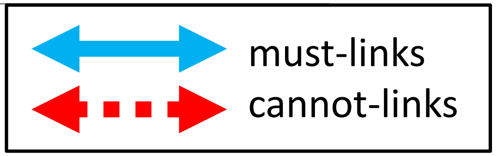
- Must-links, cannot-links, the minimum split, and the maximum diameter interact

- When adding a clause for a pair to be clustered together or separately
  - **Redundancy** or **infeasibility** is detected



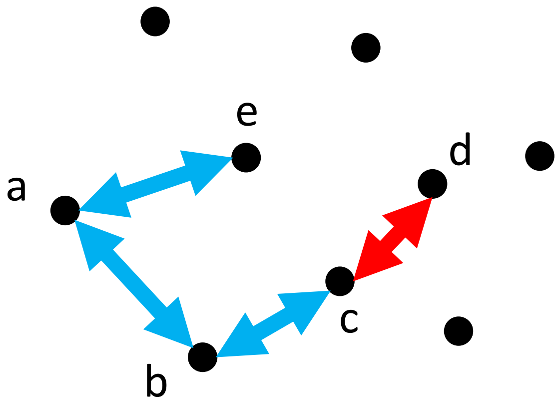
Smart Pairs

Legend

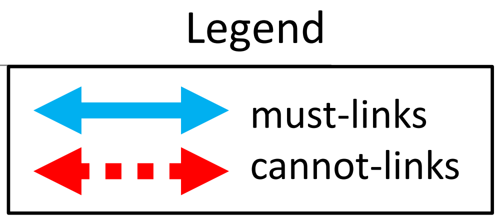


- Must-links, cannot-links, the minimum split, and the maximum diameter interact

- When adding a clause for a pair to be clustered together or separately
  - **Redundancy** or **infeasibility** is detected

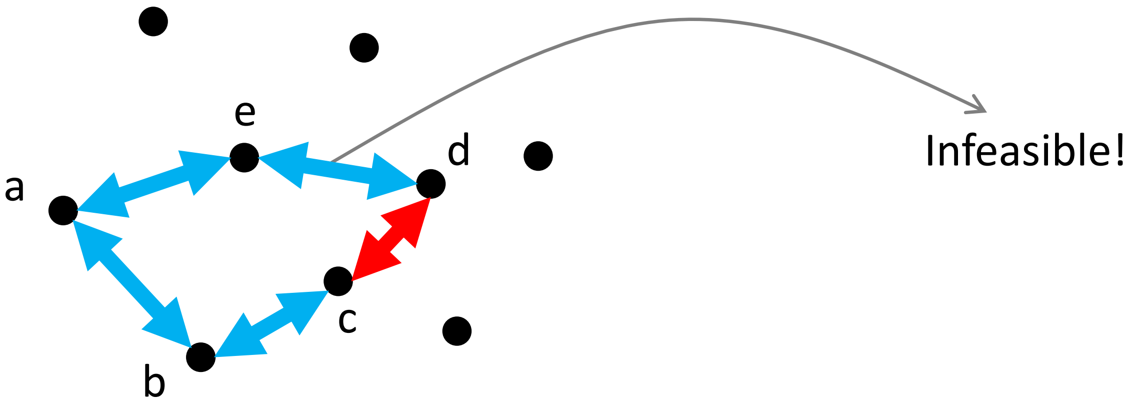


Smart Pairs



- Must-links, cannot-links, the minimum split, and the maximum diameter interact

- When adding a clause for a pair to be clustered together or separately
  - **Redundancy** or **infeasibility** is detected



Background

Encoding

Experiments

- Setup
- Results
- Ablation

Setup

- Baselines:
  - **Constrained Clustering:** not restricted to conform to a tree, max diameter only
  - **Mixed Integer Optimization:**

    [Dimitris Bertsimas, Agni Orfanoudaki, Holly Wiberg, Machine Learning, 2021]

- Datasets: seven real datasets from the **UCI** repository and four synthetic datasets from **FCPS**

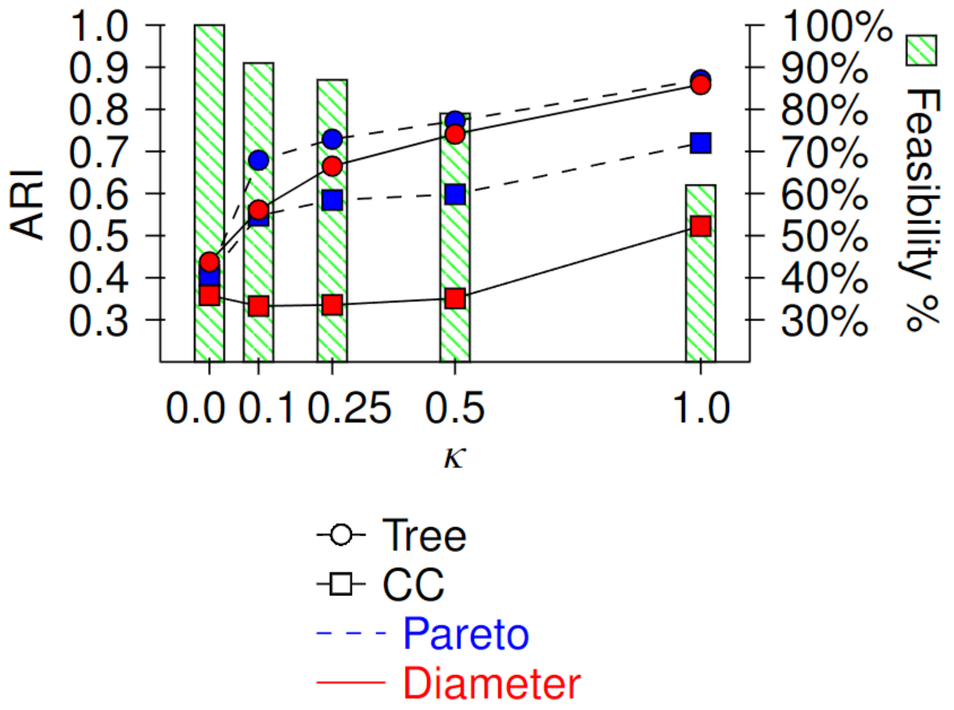
- Solver: **Loandra** with 30 minutes time limit

Better Score + Better Interpretability

- Our approach manages to produce high quality solutions in a short time

- The 3 aspects fit well together:
    - Tree clustering outperforms non-tree
    - Pareto objective outperforms only MD
    - Both utilize constraints more

- There is a trade-off between quality and feasibility



Better Performance

- Smart pairs and approximation help with performance and memory
- Approximation does not hurt the quality significantly

Dataset	Setting	ARI	Time (s)	# Clauses
Libras	SP & ϵ=0.1	0.18	866.4	2,082,261.2
	ϵ=0.1	0.16	822.0	3,888,452.0
	ϵ=0.0	0.16	1197.1	4,140,872.0
Spam	SP & ϵ=0.1	Inf.	151.6	3,823,479.2
	ϵ=0.1	Inf.	332.7	24,980,546.4
	ϵ=0.0	Inf./Unk.	864.0	69,166,751.4
WingN	SP & ϵ=0.1	1.00	1.7	95,879.25
	ϵ=0.1	1.00	4.2	1,128,700.4
	ϵ=0.0	OOM[†]	98.3	3,449,740.4

[†]OOM indicates an out-of-memory error.

Better Performance

- Smart pairs and approximation help with performance and memory
- Approximation does not hurt the quality significantly

Dataset	Setting	ARI	Time (s)	# Clauses
Libras	SP & ϵ=0.1	0.18	866.4	2,082,261.2
	ϵ=0.1	0.16	822.0	3,888,452.0
	ϵ=0.0	0.16	1197.1	4,140,872.0
Spam	SP & ϵ=0.1	Inf.	151.6	3,823,479.2
	ϵ=0.1	Inf.	332.7	24,980,546.4
	ϵ=0.0	Inf./Unk.	864.0	69,166,751.4
WingN	SP & ϵ=0.1	1.00	1.7	95,879.25
	ϵ=0.1	1.00	4.2	1,128,700.4
	ϵ=0.0	OOM†	98.3	3,449,740.4

†OOM indicates an out-of-memory error.

Summary

- First exact optimization approach to decision tree clustering
  - finds ϵ-approximation of max diameter and min split
  - supports pairwise constraints

- Smart pairs algorithm to detect redundancy and infeasibility

- Results show:
  - higher scores than non-tree clustering
  - decision trees, bi-criteria objective, and constraints complement each other

- **Future work**: see our paper