Correlation Clustering: Introduction and Innovations

Nathan Cordner

Boston University

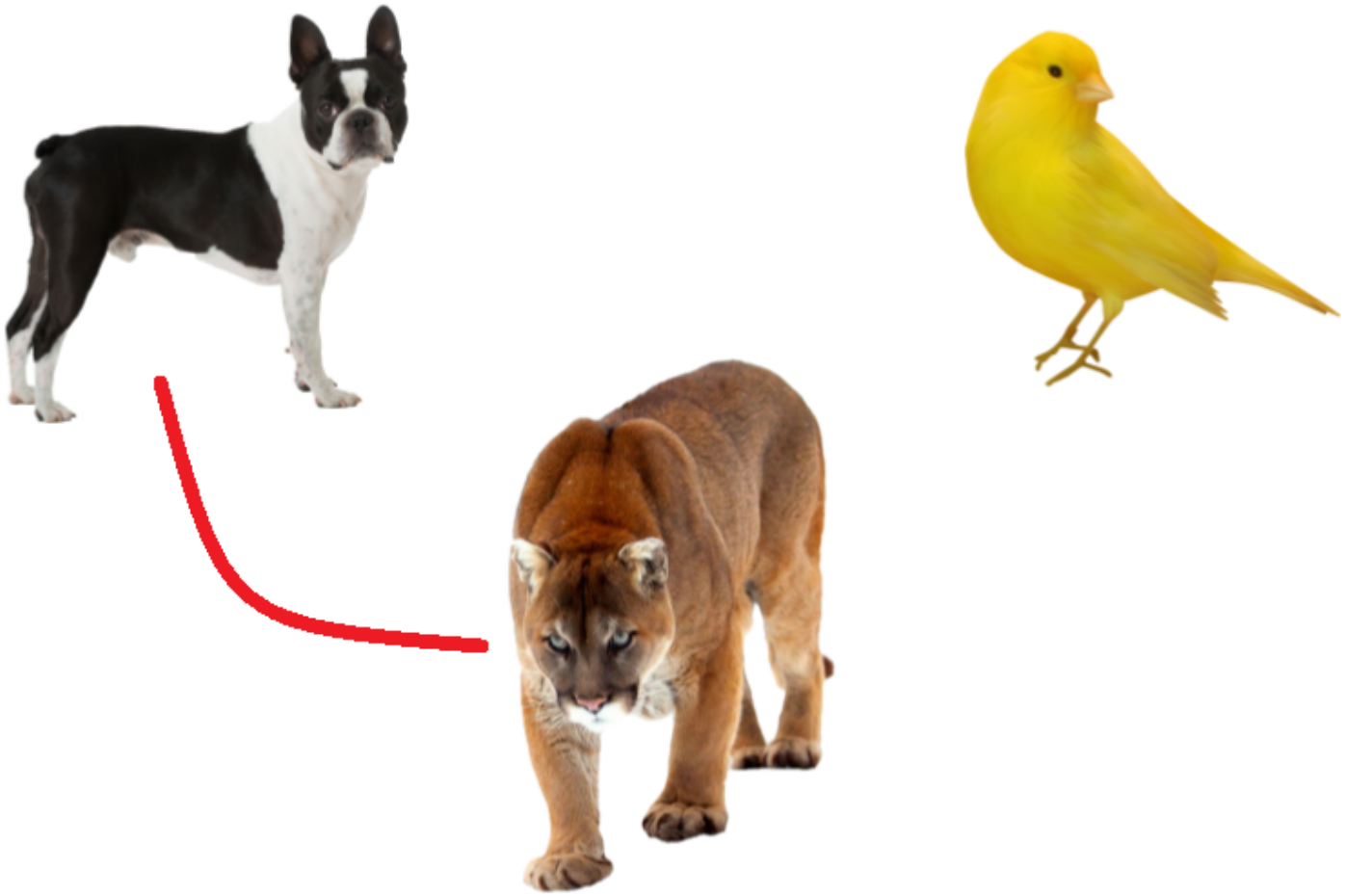
20 October 2022

Clustering Problem

Which animals belong together?

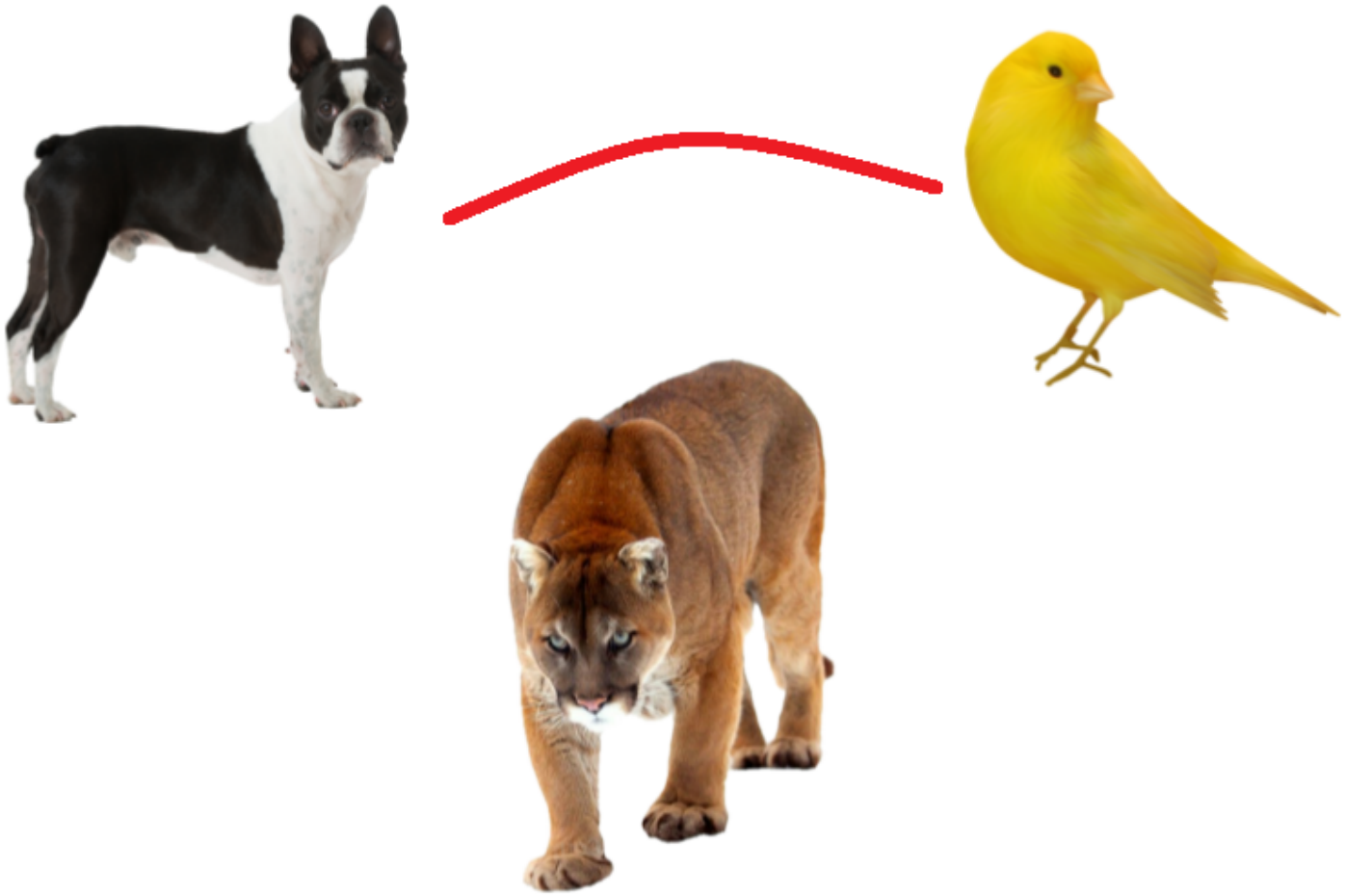
Clustering Problem

Which animals belong together?



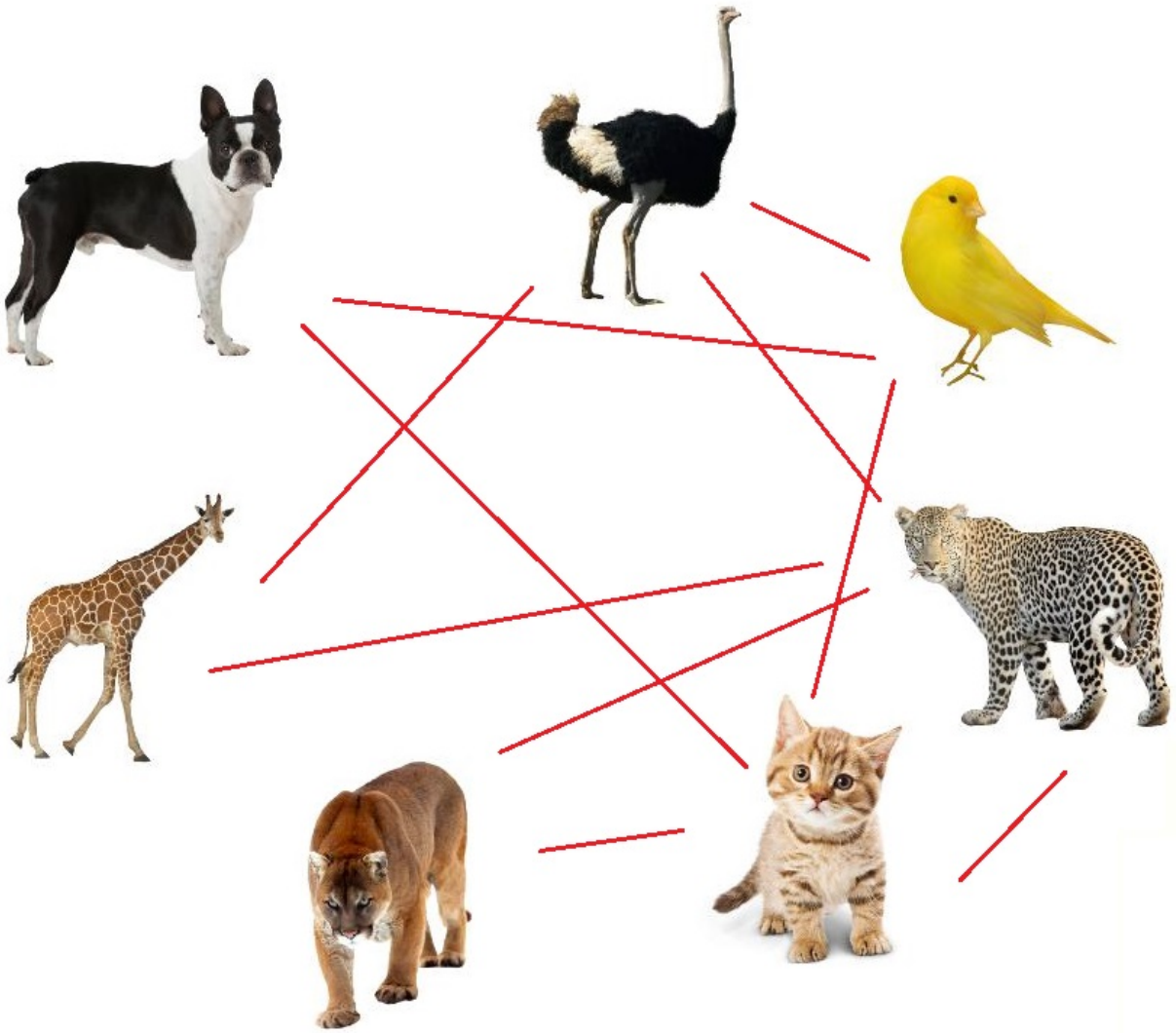
Clustering Problem

Which animals belong together?



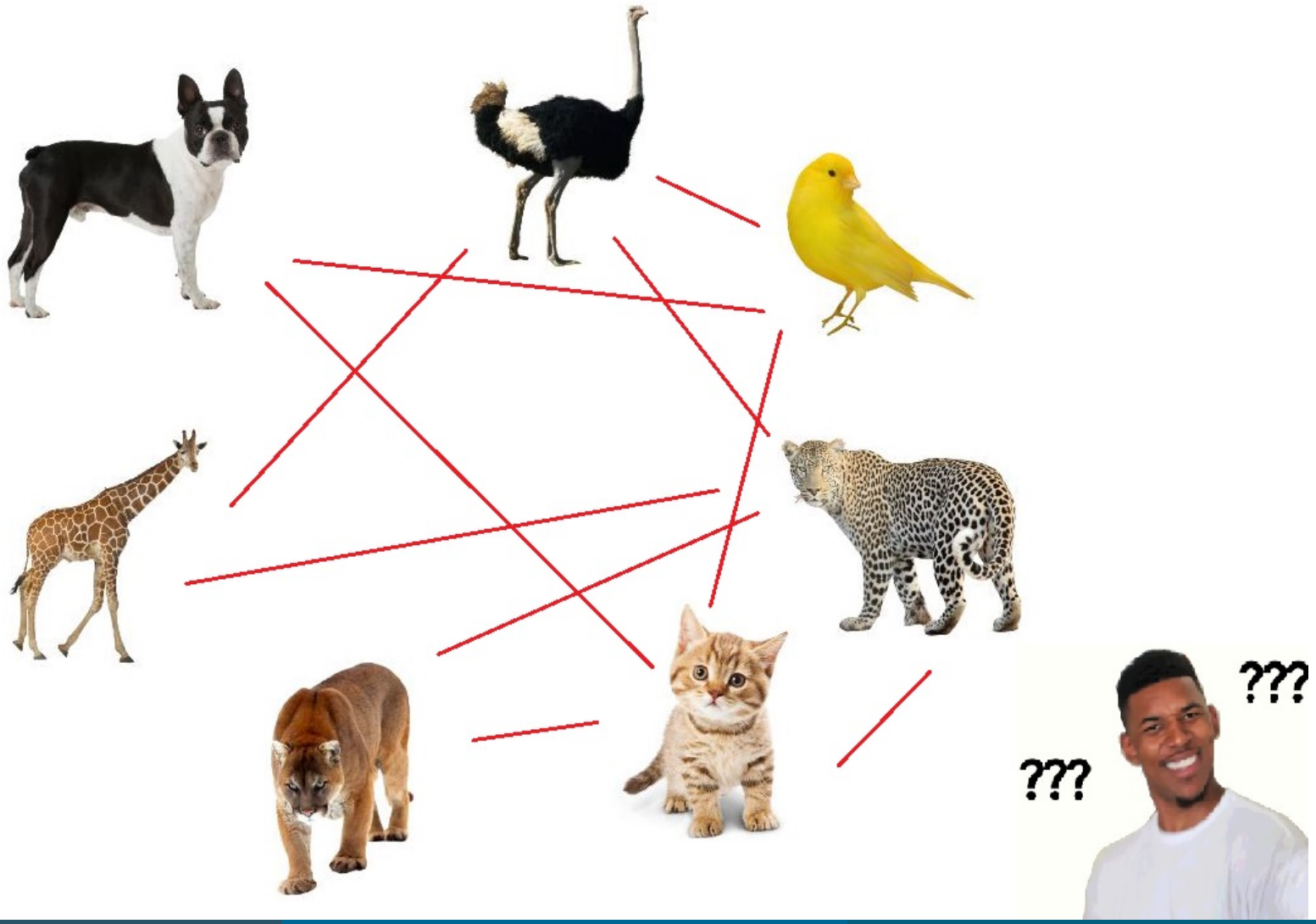
Clustering Problem

Which animals belong together?



Clustering Problem

Which animals belong together?



Correlation Clustering [BBC04]

Given a graph $G = (V, E)$

Want to cluster edges and separate non-edges

- Maximize Agreements
- **Minimize Disagreements**

Some Applications

- Classification
- Entity Resolution
- Communities in Social Networks

The Pivot Algorithm [ACN08]

Pivot(V, E):

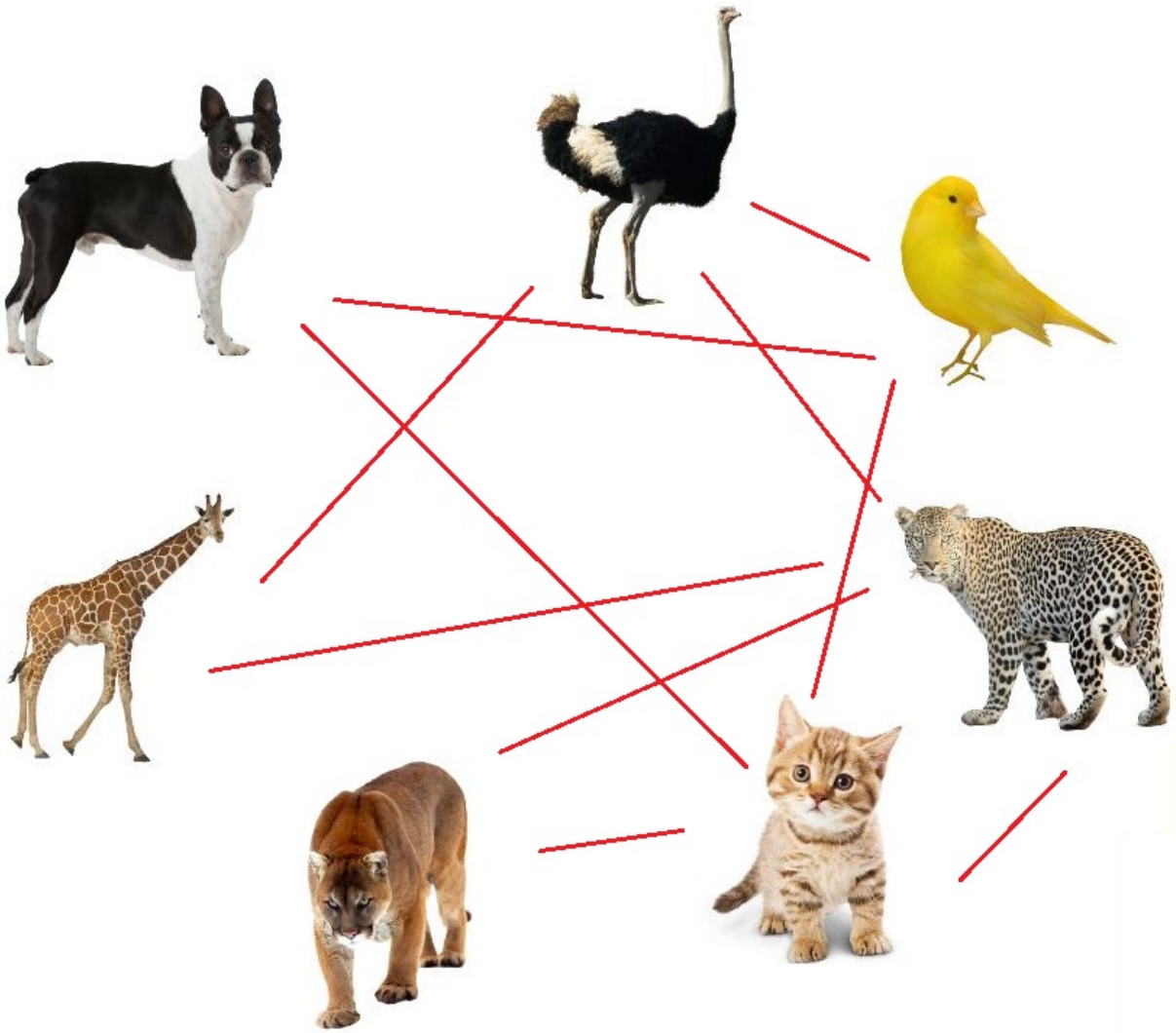
- Pick random pivot node $u \in V$
- Set $C = \{u\}$
- For all $v \in V \setminus \{u\}$:
  - If $(u, v) \in E$: Add v to C
- Repeat on $V = V \setminus C$ until empty
- Return completed clustering

Runs in $O(|V| + |E|)$ time

Randomized expected 3-approximation

The Pivot Algorithm [ACN08]

Example:



LP Methods

Linear Program for Correlation Clustering:

$$\min \sum_{(u,v)\in E} x_{uv} + \sum_{(u,v)\notin E} (1 - x_{uv})$$

$$\begin{aligned} x_{uv} + x_{vw} &\geq x_{uw} && \text{for all } u, v, w \in V \\ x_{uv} &\in [0,1] && \text{for all } u, v \in V \end{aligned}$$

- $O(|V|^2)$ variables, $O(|V|^3)$ constraints!

Rounding Methods:

- 2.5-approx [ACN08]
- Later improved to 2.06-approx [CMSY15]
  - Integrality gap is 2, so this is near optimal

Correlation Clustering Generalizations

Edge Weights [BBC04, ACN08]

- Every pair of nodes u, v has weights $w^+_{uv}, w^-_{uv} \geq 0$
- Clustering Cost:

$$\sum_{u,v \text{ in different clusters}} w^+_{uv} + \sum_{u,v \text{ in same cluster}} w^-_{uv}$$

Correlation Clustering Generalizations

Edge Weights [BBC04, ACN08]

- Every pair of nodes u, v has weights $w^+_{uv}, w^-_{uv} \geq 0$
- Clustering Cost:

$$\sum_{u,v \text{ in different clusters}} w^+_{uv} + \sum_{u,v \text{ in same cluster}} w^-_{uv}$$

Pivot: create graph edges when $w^+_{uv} \geq w^-_{uv}$

- 5-approx with probability constraints [ACN08]:

$$w^+_{uv} + w^-_{uv} = 1$$

Correlation Clustering Generalizations

Edge Weights [BBC04, ACN08]

- Every pair of nodes u, v has weights $w^+_{uv}, w^-_{uv} \geq 0$
- Clustering Cost:

$$\sum_{u,v \text{ in different clusters}} w^+_{uv} + \sum_{u,v \text{ in same cluster}} w^-_{uv}$$

Pivot: create graph edges when $w^+_{uv} \geq w^-_{uv}$

- 5-approx with probability constraints [ACN08]:

$$w^+_{uv} + w^-_{uv} = 1$$

State of the Art: LP rounding

- 2.5-approx for probability weights [ACN08]
- $O(\log |V|)$-approx for general case [CGW05, DEFI06]

Correlation Clustering Generalizations

Limited Cluster Sizes

- Given K, each cluster allowed at most K elements

Correlation Clustering Generalizations

Limited Cluster Sizes

- Given K, each cluster allowed at most K elements

Pivot [PM15]: grow clusters until size limit reached

- 11-approx if done on the fly
- 7-approx with clever preprocessing
  - increases time complexity to $O(\sqrt{K|V|}|V|^2)$

Correlation Clustering Generalizations

Limited Cluster Sizes

- Given K, each cluster allowed at most K elements

Pivot [PM15]: grow clusters until size limit reached

- 11-approx if done on the fly
- 7-approx with clever preprocessing
  - increases time complexity to $O(\sqrt{K|V|}|V|^2)$

State of the Art: LP rounding

- 6-approx [PM15]
- Later improved to 5.37-approx [JCTZ21]

Correlation Clustering Generalizations

Chromatic Correlation Clustering [BGTU15]:

- Every edge has a color from label set L
- Assign a dominant color to each cluster formed
- Penalize all edges with non-dominant colors

Correlation Clustering Generalizations

Chromatic Correlation Clustering [BGTU15]:

- Every edge has a color from label set L
- Assign a dominant color to each cluster formed
- Penalize all edges with non-dominant colors

Pivot [BGGTU15]: ignore edge colors and run as usual

- Assign cluster colors by majority vote

Correlation Clustering Generalizations

Chromatic Correlation Clustering [BGTU15]:

- Every edge has a color from label set L
- Assign a dominant color to each cluster formed
- Penalize all edges with non-dominant colors

Pivot [BGGTU15]: ignore edge colors and run as usual

- Assign cluster colors by majority vote

State of the Art: Also Pivot!

- Color-blind Pivot is a 3-approx [KSZC21]
- Other methods have better experimental results

One Algorithm to Rule Them All?

Pivot has been successfully used to cluster

- Social network graphs [KPT11]
- Protein-protein interaction graphs [KPT11; HWH15]
- Event graphs generated from news stories [CMB17]

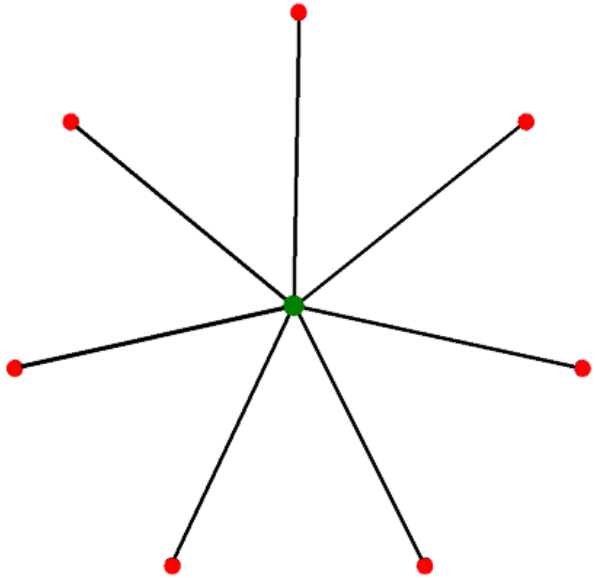
Pivot has been adapted for

- Probabilistic graphs [KPT11; MTG20]
- Fair correlation clustering [AEKM20]
- Data streaming and online settings [ACGM15; LMVW21]
- Query constraints [GKBT20]

Deterministic and parallel versions [ZW09; CDK14; PORJ15]

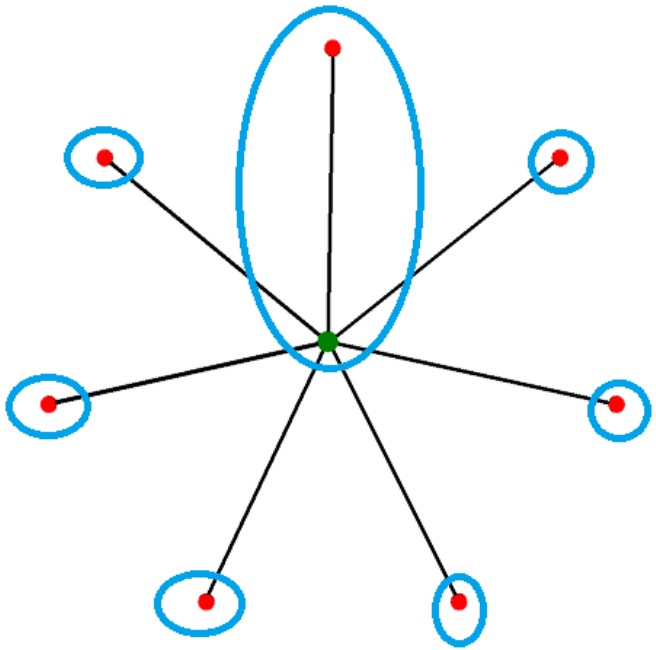
Drawbacks of Pivot

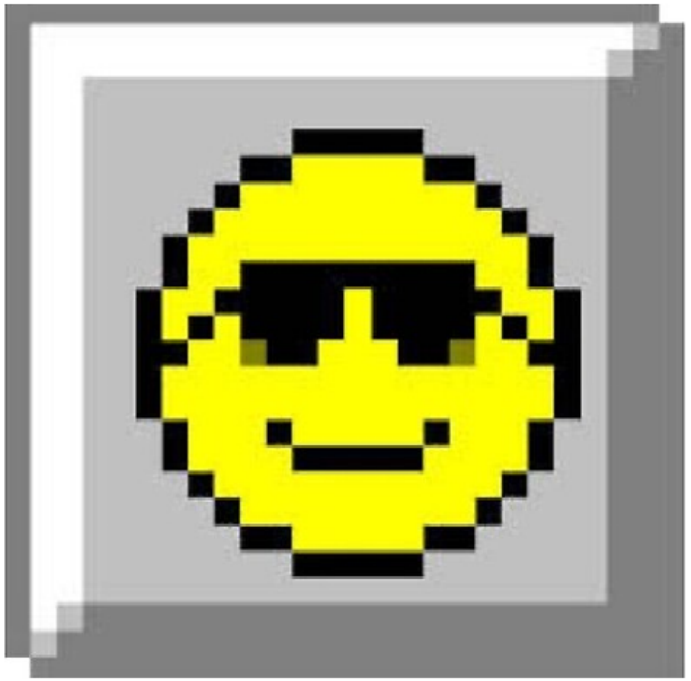
Pivot can form sparse, star-shaped clusters



Drawbacks of Pivot

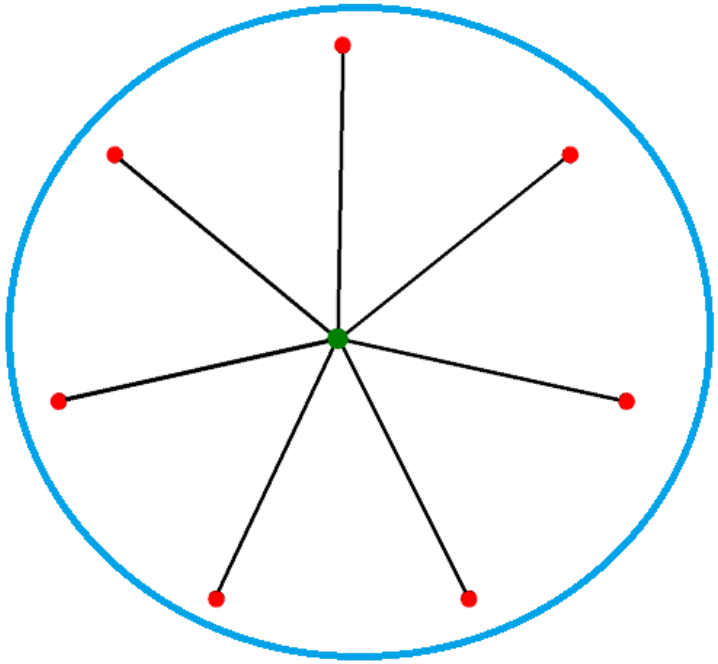
Pivot can form sparse, star-shaped clusters





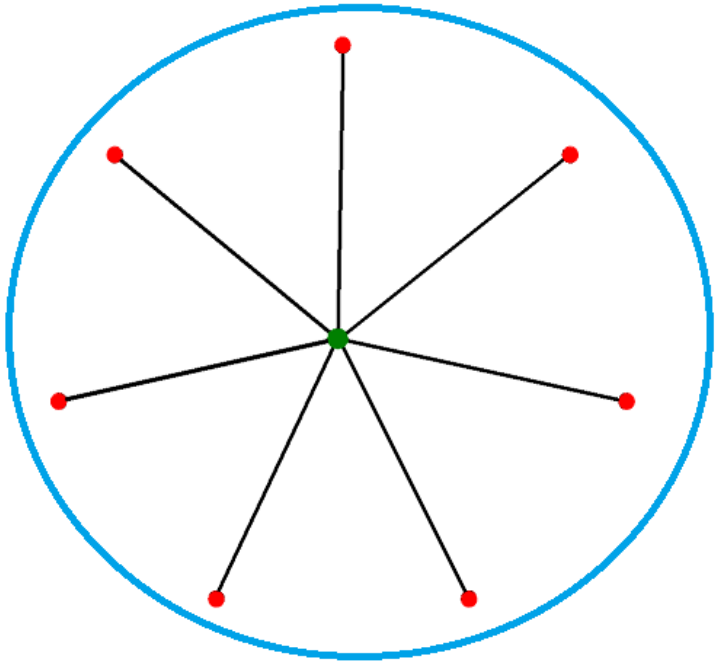
Drawbacks of Pivot

Pivot can form sparse, star-shaped clusters



Drawbacks of Pivot

Pivot can form sparse, star-shaped clusters



Can we do better and still maintain scalability?

Cluster Improvement

Local Search: given a clustering,

- Each node decides whether to stay or move clusters
- Iterate until improvements stop
- Slow: each iteration is $O(|V| + |E|)$
- Somewhat popular though [MTG20; AEKM20]

New Ideas:

- Idea 1: limit LS to just one iteration
- Idea 2: Run LS *inside* clusters only
  - Special ordering of nodes inside clusters maximizes expected cost improvement

Cluster Improvement: Idea 1

Clustering Precision: the ratio of edges inside clusters to the total number of node pairs within clusters

Lemma: clustering precision after one round of LocalSearch is at least 50%

Proof Idea: Consider node u

- If node u has fewer than 50% edges to other nodes in its current cluster, then separating u into its own cluster will decrease overall clustering cost
- Similarly, if u joins an existing cluster then it must have at least 50% edges present to other nodes in that cluster

Cluster Improvement: Idea 2

Focus on running LocalSearch inside a single cluster

Neighborhood: $N(u) = \{v \in V \mid (u, v) \in E\}$

- Order nodes in cluster by increasing size of $|N(u)|$
  - $O(|V| \log |V|)$ time
- Follow LocalSearch with this node order

Lemma: Following LocalSearch in this order maximizes expected cost decrease after one LocalSearch round

Proof Idea: Nodes with small neighborhood sizes decrease cost more when moved from the current cluster

Inner Local Search

Inner Local Search: on graph G

- Obtain clusters $C_1, \ldots, C_k$ from Pivot(G)
- Let G_i be the graph induced by C_i
- Return LocalSearch$(G_1), \ldots,$ LocalSearch(G_k)

Properties

- Nearly linear running time: $O(|V| \log d + |E|)$
  - d is size of largest Pivot cluster
- Easily run in parallel
- Immediately applies to CC generalizations
- Improves cluster costs from Pivot
- **Approximation Bound**: stay tuned!

Inner Local Search

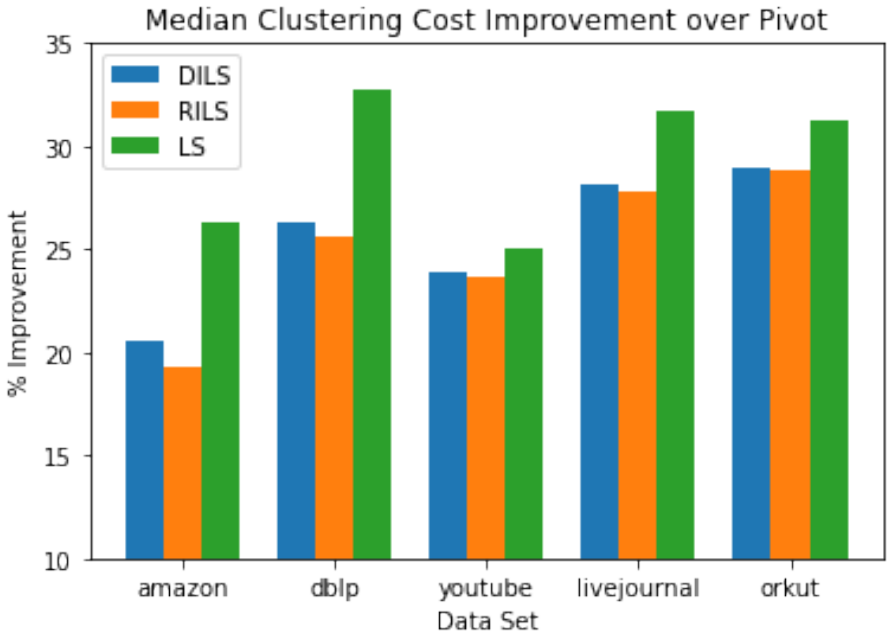
Examples:

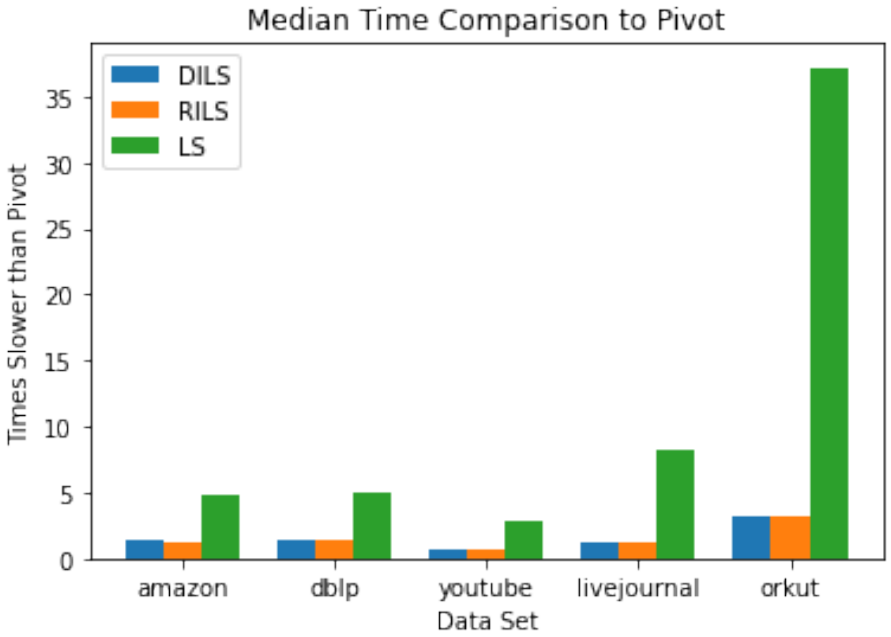
Name	$\lvert\mathbf{V}\rvert$	$\lvert\mathbf{E}\rvert$	$\mathbf{d}$	**Description**
DBLP	317080	1049866	145.1	co-authors network
Amazon	334863	925872	105.0	joint purchases
YouTube	1134890	2987624	452.6	friend network
LiveJournal	3997962	34681189	895.6	friend network
Orkut	3072441	117185083	1287.0	friend network

`snap.stanford.edu/data/#communities`

Inner Local Search

Examples: ILS gives nearly the same improvement as LS, but in a lot less time!

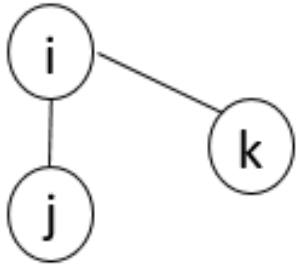




`snap.stanford.edu/data/#communities`

Inner Local Search

Theoretical Improvements



"Bad Triangles": i, j, k unclustered

- Two edges exist but the third is absent

Lemma [ACN08]: Approx bound of Pivot $\leq$ worst cost ratio for bad triangles

- Triangle completely inside cluster when i is chosen as pivot (1/3 chance)
- **Claim**: ILS reduces average cost of bad triangles inside Pivot clusters by half
- ILS approximation bound:

$$3((1/3)(\textbf{ILS triangle cost}) + 2/3) = 2.5$$

Applications and Limitations

ILS immediately applies to several CC generalizations:

- Weighted / probabilistic graphs
- Cluster size constraints
- Chromatic correlation clustering

Applications and Limitations

ILS immediately applies to several CC generalizations:

- Weighted / probabilistic graphs
- Cluster size constraints
- Chromatic correlation clustering

ILS does not work as well for some other variants:

- Fair correlation clustering
- Data streaming and online settings
- Query constraints
- **Constrained number of clusters**

Constrained Number of Clusters

Given k, find clustering with at most k clusters

Constrained Number of Clusters

Given k, find clustering with at most k clusters

- $k = 2$:
  - Pivot-like 3-approximation [BBC04]
  - Local search 2-approximation [CSW08]
  - Neither generalizes well for $k > 2$

Constrained Number of Clusters

Given k, find clustering with at most k clusters

- $k = 2$:
  - Pivot-like 3-approximation [BBC04]
  - Local search 2-approximation [CSW08]
  - Neither generalizes well for $k > 2$
- General case: $(1 + \epsilon)$ PTAS [GG06]
  - *Extremely* inefficient: $|V|^{O(9^k/\epsilon^2)} \log |V|$ running time
  - Still used from time to time [ACGM15; BEK21]

Constrained Number of Clusters

Given k, find clustering with at most k clusters

- $k = 2$:
  - Pivot-like 3-approximation [BBC04]
  - Local search 2-approximation [CSW08]
  - Neither generalizes well for $k > 2$
- General case: $(1 + \epsilon)$ PTAS [GG06]
  - *Extremely* inefficient: $|V|^{O(9^k/\epsilon^2)} \log |V|$ running time
  - Still used from time to time [ACGM15; BEK21]
- General case: LocalSearch [TCD19]
  - Faster, but still inefficient
  - No approximation guarantees

Constrained Number of Clusters

Given k, find clustering with at most k clusters

- $k = 2$:
  - Pivot-like 3-approximation [BBC04]
  - Local search 2-approximation [CSW08]
  - Neither generalizes well for $k > 2$
- General case: $(1 + \epsilon)$ PTAS [GG06]
  - *Extremely* inefficient: $|V|^{O(9^k/\epsilon^2)} \log |V|$ running time
  - Still used from time to time [ACGM15; BEK21]
- General case: LocalSearch [TCD19]
  - Faster, but still inefficient
  - No approximation guarantees

Goal: develop algorithm that is both time-efficient *and* has provable approximation guarantees

The Vote Algorithm [ES09]

Pick unclustered nodes one at a time

- First node creates its own cluster
- All others: add to existing cluster, or create own
- Greedily minimize increase in clustering cost

Previous Results

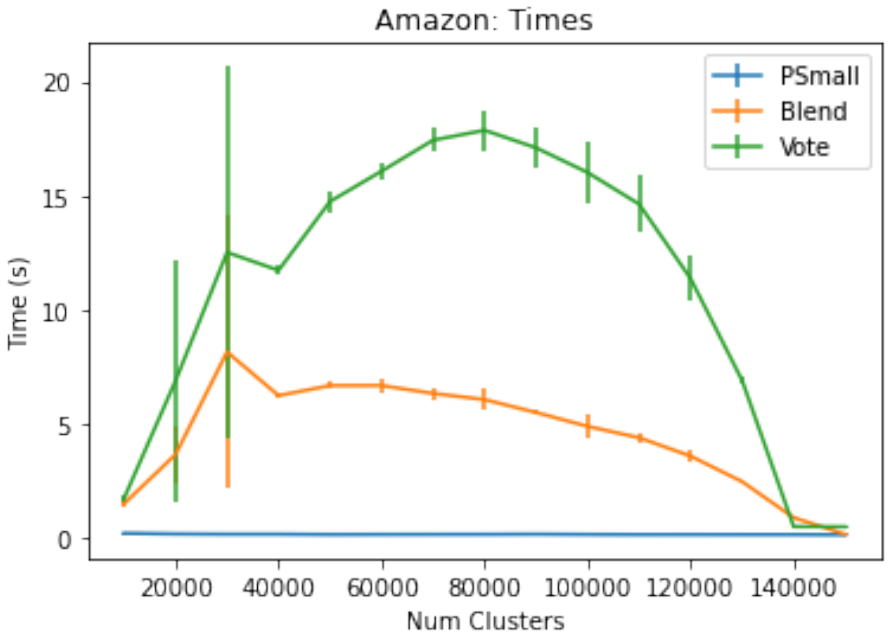
- Experimentally better than Pivot (e.g. [ES09])
- Much slower, though complexity is still $O(|V| + |E|)$

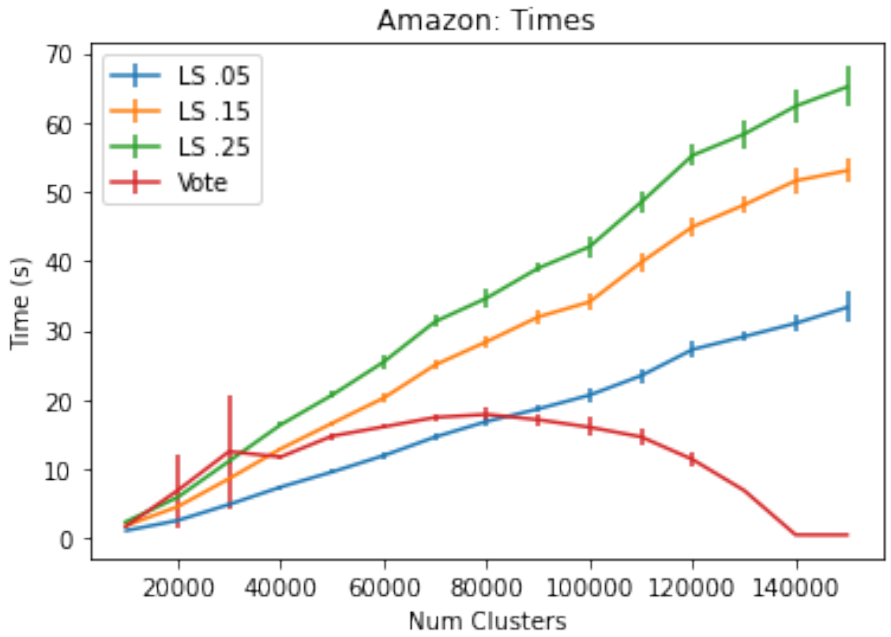
New k-CC Strategies

1. Run Pivot until k clusters are formed, then...
   - Merge new Pivot clusters into existing ones
     - Merge at random
     - Merge in order
     - **Merge to the current smallest cluster**
   - Add remaining nodes to existing clusters using the Vote algorithm
2. Run Vote until k clusters are formed, then continue without the option to form new clusters

Claim: k-Vote and k-Pivot-and-Vote ("Blend") are 7-approximation algorithms

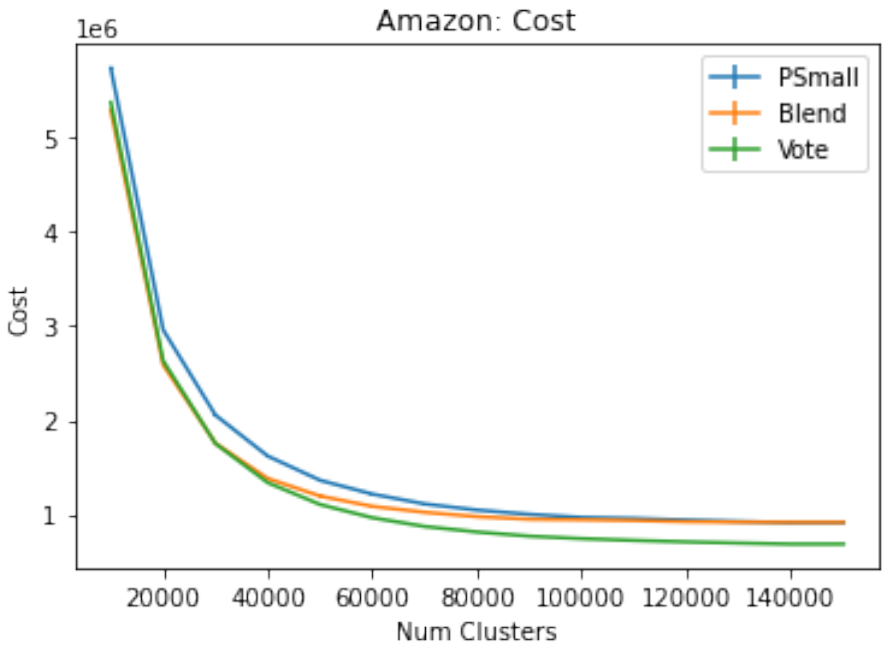
k-CC Experiments

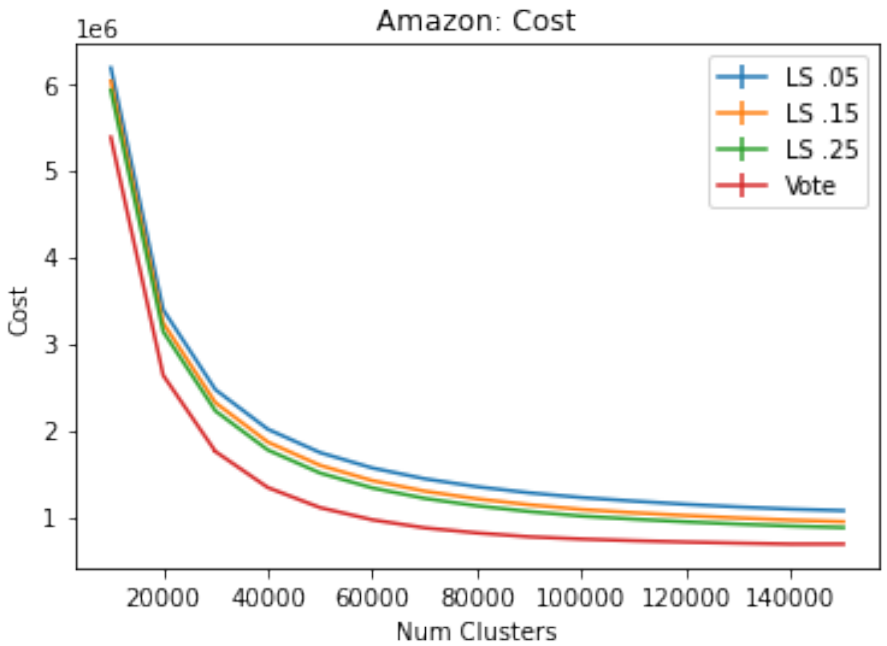




`snap.stanford.edu/data/com-Amazon.html`

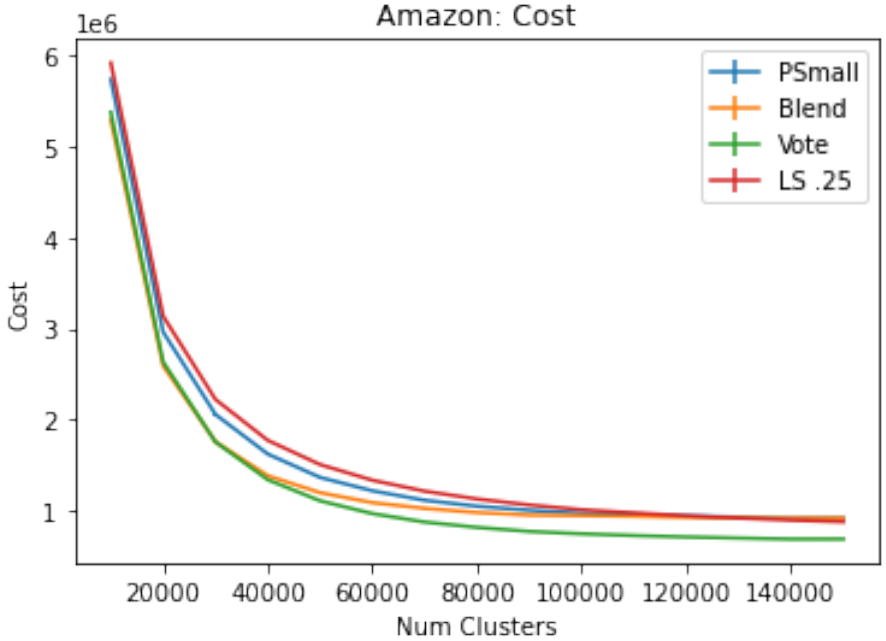
k-CC Experiments

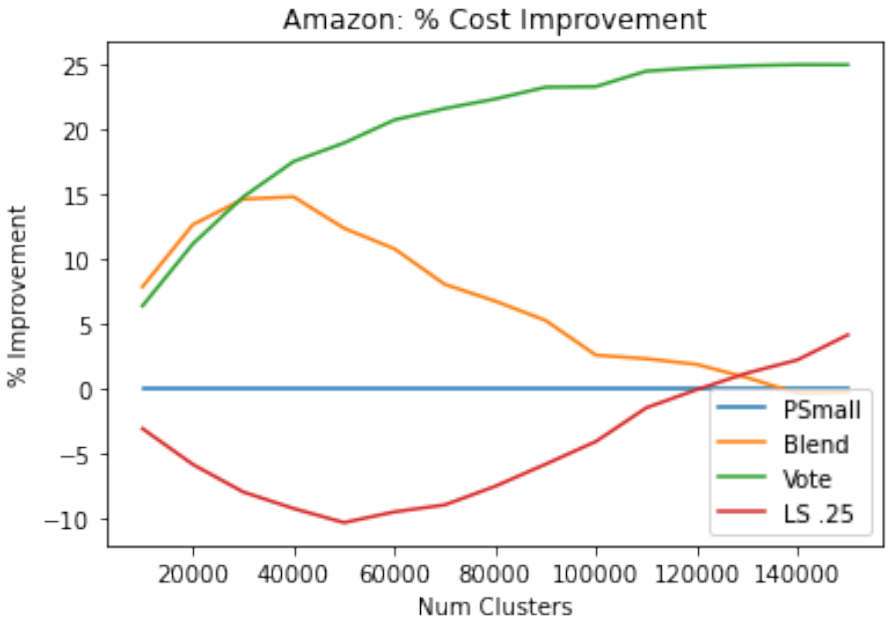




snap.stanford.edu/data/com-Amazon.html

k-CC Experiments





snap.stanford.edu/data/com-Amazon.html

References

AEKM20 Ahmadian, Epasto, Kumar, and Mahdian. *Fair correlation clustering*. 2020

ACGM15 Ahn, Cormode, Guha, McGregor, and Wirth. *Correlation clustering in data streams*. 2015

ACN08 Ailon, Charikar, and Newman. *Aggregating inconsistent information: ranking and clustering*. 2008

BBC04 Bansal, Blum, and Chawla. *Correlation clustering*. 2004

BGTU15 Bonchi, Gionis, Gullo, Tsourakakis, and Ukkonen. *Chromatic correlation clustering*. 2015

BEK21 Bun, Elias, and Kulkarni. *Differentially private correlation clustering*. 2021

CMSY15 Chawla, Makarychev, Schramm, and Yaroslavtsev. *Near optimal lp rounding algorithm for correlation clustering on complete and complete k-partite graphs*. 2015

CDK14 Chierichetti, Dalvi, and Kumar. *Correlation clustering in mapreduce*. 2014

CMB17 Christiansen, Mobasher, and Burke. *Using uncertain graphs to automatically generate event flows from news stories*. 2017

CSW08 Coleman, Saunderson, and Wirth. *A local-search 2-approximation for 2-correlation-clustering*. 2008

DEFI06 Demaine, Emanuel, Fiat, and Immorlica. *Correlation clustering in general weighted graphs*. 2006

ES09 Elsner and Schudy. *Bounding and comparing methods for correlation clustering beyond ILP*. 2009

References

GKBT20 García-Soriano, Kutzkov, Bonchi, and Tsourakakis. *Query-efficient correlation clustering*. 2020

GG06 Giotis and Guruswami. *Correlation clustering with a fixed number of clusters*. 2006

HWH15 Halim, Waqas, and Hussain. *Clustering large probabilistic graphs using multi-population evolutionary algorithm*. 2015

JCTZ21 Ji, Cheng, Tan, and Zhao. *An improved approximation algorithm for capacitated correlation clustering problem*. 2021

KSZC21 Klodt, Seifert, Zahn, Casel, Issac, and Friedrich. *A color-blind 3-approximation for chromatic correlation clustering and improved heuristics*. 2021

KPT11 Kollios, Potamias, and Terzi. *Clustering large probabilistic graphs*. 2011

LMVW21 Lattanzi, Moseley, Vassilvitskii, Wang, and Zhou. *Robust online correlation clustering*. 2021

MTG20 Mandaglio, Tagarelli, and Gullo. *In and out: optimizing overall interaction in probabilistic graphs under clustering constraints*. 2020

PORJ15 Pan, Papailiopoulos, Oymak, Recht, Ramchandran, and Jordan. *Parallel correlation clustering on big graphs*. 2015

PM15 Puleo and Milenkovic. *Correlation clustering with constrained cluster sizes and extended weights bounds*. 2015

TCD19 Thiel, Chehreghani, and Dubhashi. *A non–convex optimization approach to correlation clustering*. 2019

ZW09 Zuylen and Williamson. *Deterministic pivoting algorithms for constrained ranking and clustering problems*. 2009