Clustering Problem

Which animals belong together?

![Dog](image1.png)
![Cougar](image2.png)
![Bird](image3.png)
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:

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

\[
x_{uv} + x_{vw} \geq x_{uw} \quad \text{for all } u, v, w \in V
\]

\[
x_{uv} \in [0, 1] \quad \text{for all } u, v \in V
\]

- \(O(|V|^2)\) 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:

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

$$w^+_{uv} + w^-_{uv} = 1$$
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]
Limited Cluster Sizes

- Given $K$, each cluster allowed at most $K$ elements
**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 $O(\sqrt{K|V||V|^2})$
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

**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
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
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.
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 $\text{Pivot}(G)$
- Let $G_i$ be the graph induced by $C_i$
- Return $\text{LocalSearch}(G_1), \ldots, \text{LocalSearch}(G_k)$

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!

Cordner (Boston University)  20 October 2022  Correlation Clustering
Inner Local Search

Examples:

<table>
<thead>
<tr>
<th>Name</th>
<th>V</th>
<th>E</th>
<th>d</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DBLP</td>
<td>317080</td>
<td>1049866</td>
<td>145.1</td>
<td>co-authors network</td>
</tr>
<tr>
<td>Amazon</td>
<td>334863</td>
<td>925872</td>
<td>105.0</td>
<td>joint purchases</td>
</tr>
<tr>
<td>YouTube</td>
<td>1134890</td>
<td>2987624</td>
<td>452.6</td>
<td>friend network</td>
</tr>
<tr>
<td>LiveJournal</td>
<td>3997962</td>
<td>34681189</td>
<td>895.6</td>
<td>friend network</td>
</tr>
<tr>
<td>Orkut</td>
<td>3072441</td>
<td>117185083</td>
<td>1287.0</td>
<td>friend network</td>
</tr>
</tbody>
</table>

snap.stanford.edu/data/#communities
**Examples:** ILS gives nearly the same improvement as LS, but in a lot less time!

![Median Clustering Cost Improvement over Pivot](chart1.png)

![Median Time Comparison to Pivot](chart2.png)

[snaptanford.edu/data/#communities](https://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:

$$3\left(\frac{1}{3}(\text{ILS triangle cost}) + \frac{2}{3}\right) = 2.5$$
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
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: $(1 + \epsilon) \text{ PTAS}$ [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

![Graph of Amazon: Times](snap.stanford.edu/data/com-Amazon.html)
$k$-CC Experiments

Correlation Clustering

[Graphs showing cost vs. number of clusters for different algorithms: PSmall, Blend, Vote, LS .05, LS .15, LS .25, Vote]

snap.stanford.edu/data/com-Aazon.html
$k$-CC Experiments

[Graphs showing Amazon: Cost and Amazon: % Cost Improvement]

snap.stanford.edu/data/com-Amazon.html
<table>
<thead>
<tr>
<th>Reference</th>
<th>Authors</th>
<th>Title</th>
<th>Year</th>
</tr>
</thead>
<tbody>
<tr>
<td>AEKM20</td>
<td>Ahmadian, Epasto, Kumar, and Mahdian</td>
<td>Fair correlation clustering. 2020</td>
<td></td>
</tr>
<tr>
<td>ACGM15</td>
<td>Ahn, Cormode, Guha, McGregor, and Wirth</td>
<td>Correlation clustering in data streams. 2015</td>
<td></td>
</tr>
<tr>
<td>ACN08</td>
<td>Ailon, Charikar, and Newman</td>
<td>Aggregating inconsistent information: ranking and clustering. 2008</td>
<td></td>
</tr>
<tr>
<td>BBC04</td>
<td>Bansal, Blum, and Chawla</td>
<td>Correlation clustering. 2004</td>
<td></td>
</tr>
<tr>
<td>BGTU15</td>
<td>Bonchi, Gionis, Gullo, Tsourakakis, and Ukkonen</td>
<td>Chromatic correlation clustering. 2015</td>
<td></td>
</tr>
<tr>
<td>BEK21</td>
<td>Bun, Elias, and Kulkarni</td>
<td>Differentially private correlation clustering. 2021</td>
<td></td>
</tr>
<tr>
<td>CMSY15</td>
<td>Chawla, Makarychev, Schramm, and Yaroslavtsev</td>
<td>Near optimal lp rounding algorithm for correlation clustering on complete and complete k-partite graphs. 2015</td>
<td></td>
</tr>
<tr>
<td>CDK14</td>
<td>Chierichetti, Dalvi, and Kumar</td>
<td>Correlation clustering in mapreduce. 2014</td>
<td></td>
</tr>
<tr>
<td>CMB17</td>
<td>Christiansen, Mobasher, and Burke</td>
<td>Using uncertain graphs to automatically generate event flows from news stories. 2017</td>
<td></td>
</tr>
<tr>
<td>CSW08</td>
<td>Coleman, Saunderson, and Wirth</td>
<td>A local-search 2-approximation for 2-correlation-clustering. 2008</td>
<td></td>
</tr>
<tr>
<td>DEFI06</td>
<td>Demaine, Emanuel, Fiat, and Immorlica</td>
<td>Correlation clustering in general weighted graphs. 2006</td>
<td></td>
</tr>
<tr>
<td>ES09</td>
<td>Elsner and Schudy</td>
<td>Bounding and comparing methods for correlation clustering beyond ILP. 2009</td>
<td></td>
</tr>
</tbody>
</table>
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


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