**Setting**

Hierarchical Clustering: Iteratively group clusters using a linkage function. Given \( G \) and \( G' \):
- Single Linkage (SL): \( W(G, G') = \max_{x \in G} \min_{y \in G'} w_{xy} \)
- Complete Linkage (CL): \( W(G, G') = \min_{x \in G} \max_{y \in G'} w_{xy} \)
- Average Linkage (AL): \( W(G, G') = \frac{1}{\binom{|G|}{2}} \sum_{x \in G} \sum_{y \in G'} w_{xy} \)

**Planted Model:** A noisy hierarchical block matrix with \( L \) levels, \( 2^L \) pure clusters of size \( N_0 \) and signal to noise \( \delta \).

**Quadruplets Kernel Average Linkage (4K–AL)**

Summary: Use the quadruplets to derive a proxy for the similarities between the examples.

**Kernel function:** Two similar objects should behave similarly with respect to any third object.
- Active comparisons: Let \( w_{ij} \) be a reference similarity and \( \mathcal{D} \) a set of landmarks:
  \[
  K_{ij} = \sum_{x \in \mathcal{D}, y \in \mathcal{D}} \left( \mathbb{I}_{x \sim y} - w_{ij} \right) \left( \mathbb{I}_{x \sim y} - w_{ij} \right)
  \]
- Passive comparisons: Use all the similarities as references and all the examples as landmarks:
  \[
  K_{ij} = \sum_{x \in \mathcal{D}, y \in \mathcal{D}} \left( \sum_{k \leq j} \mathbb{I}_{x \sim y} - w_{ik} \right) \left( \sum_{k \leq j} \mathbb{I}_{x \sim y} - w_{ik} \right)
  \]

**Quadruplets-Based Average Linkage (4–AL)**

Summary: Use passive comparisons to define a cluster-level similarity function.

**Cluster-level similarity:** Clusters \( G_1, G_2 \) are more similar to each other than \( G_3, G_4 \) if their objects are, on average, more similar to each other than the objects of \( G_1 \) and \( G_2 \):

\[
W_{4}(G_1, G_2, G_3, G_4) = \sum_{x \in G_1} \sum_{y \in G_2} \sum_{z \in G_3} \sum_{w \in G_4} \mathbb{I}_{x \sim y} \mathbb{I}_{z \sim w} - w_{xy} w_{zw}
\]

Averaging over all cluster pairs gives rise to the following linkage function:

\[
W'(G_i, G_j) = \frac{1}{K(K-1)} \sum_{x \in G_i} \sum_{y \in G_j} W_{4}(G_1, G_2, G_3, G_4)
\]

**Theory**

Summary: 4K–AL and 4–AL have better guarantees than SL and CL and use less quadruplets.

<table>
<thead>
<tr>
<th>Method</th>
<th>Queries</th>
<th>Number of queries</th>
<th>Sufficient conditions</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>SL, CL</td>
<td>Active</td>
<td>( O(N^3) )</td>
<td>( N )</td>
<td>Tight!</td>
</tr>
<tr>
<td>4K–AL</td>
<td>Passive</td>
<td>( O(N \log N) )</td>
<td>( N )</td>
<td>Near-optimal number of queries.</td>
</tr>
<tr>
<td>4–AL</td>
<td>Passive</td>
<td>( O(N \log N) )</td>
<td>( N )</td>
<td>Needs initial clusters of size ( \Omega(N_0) ).</td>
</tr>
</tbody>
</table>

**Experiments**

Planted Model: SL and CL only recover the hierarchy for large signal to noise ratios while 4K–AL and 4–AL exactly recover the hierarchy for smaller signal to noise ratios.

Evaluation: Average Adjusted Rand Index (AARI, higher is better).

4K–AL, 4–AL, and SL have better guarantees than embedding based methods.

Evaluation: Dasgupta’s cost (lower is better).