Non-exhaustive, Overlapping Clustering via Low-Rank Semidefinite Programming

Yangyang Hou\(^1\)*, Joyce Jiyoung Whang\(^2\)*
David F. Gleich\(^1\)  Inderjit S. Dhillon\(^2\)

\(^1\)Purdue University
\(^2\)The University of Texas at Austin
(* first authors)

ACM SIGKDD Conference on Knowledge Discovery and Data Mining
Non-exhaustive, Overlapping Clustering
  NEO-K-Means Objective
  NEO-K-Means Algorithm

Semidefinite Programming (SDP) for NEO-K-Means

Low-Rank SDP for NEO-K-Means

Experimental Results

Conclusions
Clustering

- Clustering: finding a set of cohesive data points

- Traditional disjoint, exhaustive clustering (e.g., $k$-means)
  - Every single data point is assigned to exactly one cluster.

- Non-exhaustive, overlapping clustering
  - A data point is allowed to be outside of any cluster.
  - Clusters are allowed to overlap with each other.
The NEO-K-Means objective function

Overlap and non-exhaustiveness - handled in a unified framework

\[
\min_U \sum_{j=1}^{k} \sum_{i=1}^{n} u_{ij} \|x_i - m_j\|^2, \quad \text{where} \quad m_j = \frac{\sum_{i=1}^{n} u_{ij} x_i}{\sum_{i=1}^{n} u_{ij}}
\]

s.t. \( \text{trace}(U^T U) = (1 + \alpha)n, \sum_{i=1}^{n} I\{ (U1)_i = 0 \} \leq \beta n. \)

\( \alpha: \) overlap, \( \beta: \) non-exhaustiveness
\( \alpha = 0, \beta = 0: \) equivalent to the standard k-means objective

\[U = \begin{bmatrix} c_1 & c_2 & c_3 \\ 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 1 \end{bmatrix}, \quad U^T U = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 1 \end{bmatrix} = \begin{bmatrix} 2 & 1 & 0 \\ 1 & 2 & 1 \\ 0 & 1 & 1 \end{bmatrix}, \quad U1 = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \\ 0 \\ 2 \end{bmatrix}\]

\[\text{cluster sizes, no. of clusters a data point belongs to}\]
Normalized Cut for Overlapping Community Detection

(a) Disjoint communities:
\[ \text{ncut}(G) = \frac{2}{14} + \frac{2}{4} \]

(b) Overlapping communities:
\[ \text{ncut}(G) = \frac{2}{14} + \frac{3}{9} \]

Weighted Kernel NEO-K-Means objective is equivalent to the extended normalized cut objective.

---

The NEO-K-Means Algorithm is a simple iterative algorithm that monotonically decreases the NEO-K-Means objective.

- $\alpha = 0, \beta = 0$: identical to the standard $k$-means algorithm

Example ($n = 20, \alpha = 0.15, \beta = 0.05$)

- Assign $n - \beta n (=19)$ data points to their closest clusters.
- Make $\beta n + \alpha n (=4)$ assignments by taking minimum distances.

---

Motivation

- **NEO-K-Means Algorithm**
  - Fast iterative algorithm
  - Susceptible to initialization
  - Can be trapped in local optima

LRSDP initialization allows the NEO-K-Means algorithm to consistently produce a reasonable clustering structure.
Goal: more accurate and more reliable solutions than the iterative NEO-K-Means algorithm by paying additional computational cost

- **NEO-K-Means Objective Iterative NEO-K-Means Algorithm**: Fast and scalable, Trapped in local optima
- **Convex SDP relaxation CVX**: Slow and not scalable, Globally optimized
- **Low-Rank SDP Augmented Lagrangian**: Faster than CVX, Locally optimized
Background: Semidefinite Programs (SDPs)

- Semidefinite Programming (SDP)
  - Convex problem (→ globally optimized via a variety of solvers)
  - The number of variables is quadratic in the number of data points.
  - Problems with fewer than 100 data points

- Low-rank SDP
  - Non-convex (→ locally optimized via an augmented Lagrangian method)
  - Problems with tens of thousands of data points

Canonical SDP

\[
\begin{align*}
\text{maximize} & \quad \text{trace}(CX) \\
\text{subject to} & \quad X \succeq 0, X = X^T, \\
& \quad \text{trace}(A_iX) = b_i, \\
& \quad i = 1, \ldots, m
\end{align*}
\]

Low-rank SDP

\[
\begin{align*}
\text{maximize} & \quad \text{trace}(CYY^T) \\
\text{subject to} & \quad Y : n \times k, \\
& \quad \text{trace}(A_iYY^T) = b_i \\
& \quad i = 1, \ldots, m
\end{align*}
\]
NEO-K-Means as an SDP

- Three key variables to model the assignment structure U
  - Co-occurrence matrix $Z = \sum_{c=1}^{k} \frac{W_u c (W_u c)^T}{u_c^T W_u c}$
  - $f$: overlap, $g$: non-exhaustiveness

$$U = \begin{bmatrix} c_1 & c_2 \\ 1 & 0 \\ 1 & 1 \\ 0 & 1 \\ 0 & 0 \end{bmatrix}$$

$$g = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 0 \end{bmatrix}$$

$$f = \begin{bmatrix} 1 \\ 2 \\ 1 \\ 0 \end{bmatrix}$$

$$Z = \begin{bmatrix} w_1^2 & w_1 w_2 & 0 & 0 \\ w_2 w_1 & w_1 w_2 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 & w_2^2 & 0 \\ 0 & 0 & w_2 w_3 & 0 \\ 0 & 0 & w_2 + w_3 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$
SDP-like Formulation for NEO-K-Means

- NEO-K-Means with a discrete assignment matrix
  - Non-convex, combinatorial problem

\[
\begin{align*}
\text{maximize} & \quad \text{trace}(KZ) - f^T d \\
\text{subject to} & \quad \text{trace}(W^{-1}Z) = k, \quad (a) \\
& \quad Z_{ij} \geq 0, \quad (b) \\
& \quad Z \geq 0, \ Z = Z^T \quad (c) \\
& \quad Ze = Wf, \quad (d) \\
& \quad e^T f = (1 + \alpha)n, \quad (e) \\
& \quad e^T g \geq (1 - \beta)n, \quad (f) \\
& \quad f \geq g, \quad (g) \\
& \quad \text{rank}(Z) = k, \quad (h) \\
& \quad f \in \mathbb{Z}^n_{\geq 0}, \ g \in \{0, 1\}^n. \quad (i)
\end{align*}
\]

- Z must arise from an assignment matrix
- Overlap & non-exhaustiveness constraints
- Combinatorial problem
SDP for NEO-K-Means

- Convex relaxation of NEO-K-Means
  - Any local optimal solution must be a global solution.

\[
\begin{align*}
\text{maximize} & \quad \text{trace}(KZ) - f^T d \\
\text{subject to} & \quad \text{trace}(W^{-1}Z) = k, \quad (a) \\
& \quad Z_{ij} \geq 0, \quad (b) \\
& \quad Z \succeq 0, \ Z = Z^T \quad (c) \\
& \quad Ze = Wf, \quad (d) \\
& \quad e^T f = (1 + \alpha)n, \quad (e) \\
& \quad e^T g \geq (1 - \beta)n, \quad (f) \\
& \quad f \geq g, \quad (g) \\
& \quad 0 \leq g \leq 1 \quad (h)
\end{align*}
\]

- Z must arise from an assignment matrix
- Overlap & non-exhaustiveness constraints
- Relaxation
Low-Rank SDP

- Low-Rank SDP
  - Low-rank factorization of $\mathbf{Z}$: $\mathbf{Y}\mathbf{Y}^T$ ($\mathbf{Y}$: $n \times k$, non-negative)
  - $s, r$: slack variables
  - Lose convexity but only requires linear memory

\[
\begin{align*}
\text{minimize} & \quad \mathbf{f}^T \mathbf{d} - \text{trace}(\mathbf{Y}^T \mathbf{K}\mathbf{Y}) \\
\text{subject to} & \quad k = \text{trace}(\mathbf{Y}^T \mathbf{W}^{-1}\mathbf{Y}) \\
& \quad 0 = \mathbf{Y}\mathbf{Y}^T\mathbf{e} - \mathbf{W}\mathbf{f} \\
& \quad 0 = \mathbf{e}^T\mathbf{f} - (1 + \alpha)n \\
& \quad 0 = \mathbf{f} - \mathbf{g} - s \\
& \quad 0 = \mathbf{e}^T\mathbf{g} - (1 - \beta)n - r \\
& \quad Y_{ij} \geq 0, s \geq 0, r \geq 0 \\
& \quad 0 \leq \mathbf{f} \leq k\mathbf{e}, 0 \leq \mathbf{g} \leq 1
\end{align*}
\]
LRSDP: optimize the NEO-K-Means Low-Rank SDP

Augmented Lagrangian method:
minimizing an augmented Lagrangian of the problem that includes
- Current estimate of the Lagrange multipliers
- Penalty term that derives the solution towards the feasible set

\[
\mathcal{L}_A(Y, f, g, s, r; \lambda, \mu, \gamma, \sigma) = \underbrace{f^T d - \text{trace}(Y^T K Y)}_{\text{the objective}}
- \lambda_1(\text{trace}(Y^T W^{-1} Y) - k) + \frac{\sigma}{2}(\text{trace}(Y^T W^{-1} Y) - k)^2
- \mu^T (Y Y^T e - W f) + \frac{\sigma}{2} (Y Y^T e - W f)^T (Y Y^T e - W f)
- \lambda_2(e^T f - (1 + \alpha)n) + \frac{\sigma}{2} (e^T f - (1 + \alpha)n)^2
- \gamma^T (f - g - s) + \frac{\sigma}{2} (f - g - s)^T (f - g - s)
- \lambda_3(e^T g - (1 - \beta)n - r) + \frac{\sigma}{2} (e^T g - (1 - \beta)n - r)^2
\]
Algorithmic Validation

- **Comparison of SDP and LRSDP**
  - LRSDP is roughly an order of magnitude faster than CVX.
  - The objective value are different in light of the solution tolerances.
  - dolphins ¹: 62 nodes, 159 edges, les miserables ²: 77 nodes, 254 edges

<table>
<thead>
<tr>
<th></th>
<th>Objective value</th>
<th>Run time</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>SDP</td>
<td>LRSDP</td>
</tr>
<tr>
<td>dolphins</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$k=2$, $\alpha=0.2$, $\beta=0$</td>
<td>-1.968893</td>
<td>-1.968329</td>
</tr>
<tr>
<td>$k=2$, $\alpha=0.2$, $\beta=0.05$</td>
<td>-1.969080</td>
<td>-1.968128</td>
</tr>
<tr>
<td>$k=3$, $\alpha=0.3$, $\beta=0$</td>
<td>-2.913601</td>
<td>-2.915384</td>
</tr>
<tr>
<td>$k=3$, $\alpha=0.3$, $\beta=0.05$</td>
<td>-2.921634</td>
<td>-2.922252</td>
</tr>
<tr>
<td>les miserables</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$k=2$, $\alpha=0.2$, $\beta=0$</td>
<td>-1.937268</td>
<td>-1.935365</td>
</tr>
<tr>
<td>$k=2$, $\alpha=0.3$, $\beta=0$</td>
<td>-1.949212</td>
<td>-1.945632</td>
</tr>
<tr>
<td>$k=3$, $\alpha=0.2$, $\beta=0.05$</td>
<td>-2.845720</td>
<td>-2.845070</td>
</tr>
<tr>
<td>$k=3$, $\alpha=0.3$, $\beta=0.05$</td>
<td>-2.859959</td>
<td>-2.859565</td>
</tr>
</tbody>
</table>

Rounding Procedure & Practical Improvements

- Problem $\rightarrow$ Relaxation $\rightarrow$ Rounding $\rightarrow$ Refinement

- Rounding procedure
  - $Y$: normalized assignment matrix
  - $f$: the number of clusters each data point is assigned to
  - $g$: which data points are not assigned to any cluster

- Refinement
  - Use LRSDP solution as the initial cluster assignment for the iterative NEO-K-Means algorithm

- Sampling
  - Run LRSDP on a 10% sample of the data points

- Two-level hierarchical clustering
  - First level: $k' = \sqrt{k}$, $\alpha' = \sqrt{1 + \alpha} - 1$ and unchanged $\beta$
  - Second level: $k'$, $\alpha'$ and $\beta' = 0$ for each cluster at level 1
Experimental Results on Synthetic Problems

- Overlapping community detection on a Watts-Strogatz cycle graph
- LRSDP initialization lowers the errors.
Experimental Results on Data Clustering

- Comparison of NEO-K-Means objective function values
  - Real-world datasets from Mulan\(^3\)
  - By using the LRSDP solution as the initialization of the iterative algorithm, we can achieve better objective function values.

<table>
<thead>
<tr>
<th></th>
<th>worst</th>
<th>best</th>
<th>avg</th>
</tr>
</thead>
<tbody>
<tr>
<td>yeast</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>kmeans+neo</td>
<td>9611</td>
<td>9495</td>
<td>9549</td>
</tr>
<tr>
<td>lrstdp+neo</td>
<td>9440</td>
<td>9280</td>
<td>9364</td>
</tr>
<tr>
<td>slrsdp+neo</td>
<td>9471</td>
<td>9231</td>
<td>9367</td>
</tr>
<tr>
<td>music</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>kmeans+neo</td>
<td>87779</td>
<td>70158</td>
<td>77015</td>
</tr>
<tr>
<td>lrstdp+neo</td>
<td>82323</td>
<td>70157</td>
<td>75923</td>
</tr>
<tr>
<td>slrsdp+neo</td>
<td>82336</td>
<td>70159</td>
<td>75926</td>
</tr>
<tr>
<td>scene</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>kmeans+neo</td>
<td>18905</td>
<td>18745</td>
<td>18806</td>
</tr>
<tr>
<td>lrstdp+neo</td>
<td>18904</td>
<td>18759</td>
<td>18811</td>
</tr>
<tr>
<td>slrsdp+neo</td>
<td>18895</td>
<td>18760</td>
<td>18810</td>
</tr>
</tbody>
</table>

\(^3\)http://mulan.sourceforge.net/datasets.html
Experimental Results on Data Clustering

- $F_1$ scores on real-world vector datasets
  - Low-rank SDP method improves the clustering results.

<table>
<thead>
<tr>
<th></th>
<th>moc</th>
<th>esp</th>
<th>isp</th>
<th>okm</th>
<th>kmeans+neo</th>
<th>lrspdp+neo</th>
<th>slrspdp+neo</th>
</tr>
</thead>
<tbody>
<tr>
<td>yeast</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>worst</td>
<td>-</td>
<td>0.274</td>
<td>0.232</td>
<td>0.311</td>
<td>0.356</td>
<td>0.390</td>
<td>0.369</td>
</tr>
<tr>
<td>best</td>
<td>-</td>
<td>0.289</td>
<td>0.256</td>
<td>0.323</td>
<td>0.366</td>
<td>0.391</td>
<td>0.391</td>
</tr>
<tr>
<td>avg.</td>
<td>-</td>
<td>0.284</td>
<td>0.248</td>
<td>0.317</td>
<td>0.360</td>
<td>0.391</td>
<td>0.382</td>
</tr>
<tr>
<td>music</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>worst</td>
<td>0.530</td>
<td>0.514</td>
<td>0.506</td>
<td>0.524</td>
<td>0.526</td>
<td>0.537</td>
<td>0.541</td>
</tr>
<tr>
<td>best</td>
<td>0.544</td>
<td>0.539</td>
<td>0.539</td>
<td>0.531</td>
<td>0.551</td>
<td>0.552</td>
<td>0.552</td>
</tr>
<tr>
<td>avg.</td>
<td>0.538</td>
<td>0.526</td>
<td>0.517</td>
<td>0.527</td>
<td>0.543</td>
<td>0.545</td>
<td>0.547</td>
</tr>
<tr>
<td>scene</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>worst</td>
<td>0.466</td>
<td>0.569</td>
<td>0.586</td>
<td>0.571</td>
<td>0.597</td>
<td>0.610</td>
<td>0.605</td>
</tr>
<tr>
<td>best</td>
<td>0.470</td>
<td>0.582</td>
<td>0.609</td>
<td>0.576</td>
<td>0.627</td>
<td>0.614</td>
<td>0.625</td>
</tr>
<tr>
<td>avg.</td>
<td>0.467</td>
<td>0.575</td>
<td>0.598</td>
<td>0.573</td>
<td>0.610</td>
<td>0.613</td>
<td>0.613</td>
</tr>
</tbody>
</table>
Experimental Results on Graph Clustering

- Conductance-vs-graph coverage
  - The lower curve indicates better communities.
Experimental Results on Graph Clustering

- AUC of conductance-vs-graph coverage
  - Real-world networks from SNAP\(^4\)
  - LRSDP produces the best quality communities in terms of AUC score.
  - The largest graph: AstroPh (17,903 nodes, 196,972 edges)

<table>
<thead>
<tr>
<th></th>
<th>Facebook1</th>
<th>Facebook2</th>
<th>HepPh</th>
<th>AstroPh</th>
</tr>
</thead>
<tbody>
<tr>
<td>bigclam</td>
<td>0.830</td>
<td>0.640</td>
<td>0.625</td>
<td>0.645</td>
</tr>
<tr>
<td>demon</td>
<td>0.495</td>
<td>0.318</td>
<td>0.503</td>
<td>0.570</td>
</tr>
<tr>
<td>oslom</td>
<td>0.319</td>
<td>0.445</td>
<td>0.465</td>
<td>0.580</td>
</tr>
<tr>
<td>nise</td>
<td>0.297</td>
<td>0.293</td>
<td>0.102</td>
<td>0.153</td>
</tr>
<tr>
<td>neo</td>
<td>0.285</td>
<td>0.269</td>
<td>0.206</td>
<td>0.190</td>
</tr>
<tr>
<td>LRSDP</td>
<td><strong>0.222</strong></td>
<td><strong>0.148</strong></td>
<td><strong>0.091</strong></td>
<td><strong>0.137</strong></td>
</tr>
</tbody>
</table>

\(^4\)http://snap.stanford.edu/

Joyce Jiyoung Whang, The University of Texas at Austin
Conclusions

- We propose a convex SDP relaxation of a k-means-like objective that handles non-exhaustive, overlapping clustering problems.

- We formulate a low-rank factorization of the SDP problem and implement the scalable LRSDP algorithm.

- We also propose a series of initialization and rounding strategies that accelerate the convergence of our optimization procedures.

- Experiments show that our LRSDP approach gives reliable solutions on both data clustering and overlapping community detection problems.

http://www.cs.utexas.edu/~joyce/