For clustering problems with submodular objective functions, we introduce the minimum average cost criterion.

The proposed algorithm
- does not require # of clusters in advance
- computes an optimal # of clusters and an optimal partition in polynomial time
- uses the theory of intersecting submodular functions

Keywords
clustering, submodular functions, combinatorial optimization
• \( V = \{1, \ldots, n\} \) is a finite set of data points

• A function \( f \) defined on \( 2^V = \{S : S \subseteq V\} \) is **submodular** if
  \[
  f(S) + f(T) \geq f(S \cup T) + f(S \cap T), \quad \forall S, T \subseteq V
  \]

  ➢ A generalization of cut functions, entropy functions, etc.

**Clustering problem with submodular objective function**

[\text{Narasimhan-Jojic-Bilmes, NIPS 2005}]

Given set \( V \), integer \( k(\leq n) \), and submodular function \( f \), find a \( k \)-partition of \( V \), \( \{S_1, \ldots, S_k\} \) that minimizes \( \sum_{i=1}^{k} f(S_i) \)

In the case of a network, \( \sum_{i} f(S_i) = 2 \times \#(\text{red edge}) \)

\( (V \) is a set of nodes, and \( f \) is a cut function)
**Optimal $k$-clustering problem** [Narasimhan et al., NIPS 2005]

\[
\begin{align*}
\min & \sum_{i=1}^{k} f(S_i) \\
\text{s. t.} & \{S_1, \ldots, S_k\} \text{ is a } k\text{-partition of } V
\end{align*}
\]

- $k$ (# of clusters) should be computed via some method
- NP-hard

**Minimum Average Cost (MAC) clustering** [This work]

\[
\begin{align*}
\min & \sum_{S \in \mathcal{P}} f(S) / (|\mathcal{P}| - \beta) \\
\text{s. t.} & \mathcal{P} \text{ is a partition of } V \\
& |\mathcal{P}| > \beta
\end{align*}
\]

where $0 \leq \beta < n$

- averaged objective function
- $k = |\mathcal{P}|$ and a partition $\mathcal{P}$ are determined at the same time
- Solvable in poly time
- competitive with other methods

If $\beta$ is small, $\mathcal{P}$ is coarse. If $\beta$ is big, $\mathcal{P}$ is fine.
Theorem [This work]. There is an algorithm that computes all the $\beta$-MAC clusterings in polynomial time in total.

Observation. Suppose that a partition $P$ is a $\beta$-MAC clustering for some $\beta$, and let $k = |P|$. Then, $P$ is a $k$-optimal clustering.

The information about MAC clusterings gives a portion of the information about optimal $k$-clusterings.

(remember that an optimal $k$-clustering problem is NP-hard)

Example

In this case, our algorithm computes optimal $k$-clusterings for $k = 1, 3, & 4$

For more information, please visit Poster T26