Clustering with Same-Cluster Queries

Hassan Ashtiani, Shai Ben-David, Shrinu Kushagra

David Cheriton School of Computer Science
University of Waterloo
Motivation

- Clustering is the task of automatically partitioning a set of instances into meaningful (!) subsets
  - Well-separated subsets
  - Coherent subsets
  - ...

- In contrast to supervised learning, the goal is not clear/unique!
Clustering is the task of automatically partitioning a set of instances into meaningful subsets.
- Well-separated subsets
- Coherent subsets
- ...

In contrast to supervised learning, the goal is not clear/unique!
Motivation

- How does your fancy clustering method perform here?
Motivation

- How does your fancy clustering method perform here?

- Under-specificity is a real issue in clustering
- We need domain knowledge
Challenges of Clustering

- **Under-specificity**
  - Various modelling choices ... with contradicting outcomes
  - A systematic approach to incorporate domain knowledge?

- **High Computational Complexity**
  - Optimizing K-means, K-median, K-center, ... objective functions is NP-hard

- Seemingly orthogonal, but actually related!
Acquiring Domain Knowledge

- Trial-and-error/Intuitions
- Off-line Models
  - Constrained Clustering (Wagstaff et al. (2000))
  - Demonstration-based Clustering (Ashtiani and Ben-David (2015))
- Interactive Clustering
  - Merge/split Model (Balcan and Blum (2008))
  - Our approach
Our Approach

- Learner interacts with an expert/oracle to get “advice”

- **Same-Cluster Query**
  - Do $x_1$ and $x_2$ belong to the same cluster?

- A natural/user-friendly form of query
  - Record De-Duplication
  - Assisted Troubleshooting
  - ...

Ashtiani, Ben-David, Kushagra  
Interactive Clustering  
December 5, 2016  
7 / 20
Key Takeaways

Interactive clustering in the form of same-cluster queries can help us deal with

- Under-specificity
- Computational complexity (?!?)
Interactive clustering in the form of same-cluster queries can help us deal with

- Under-specificity
- Computational complexity (?!?)

The use of a few queries can make an otherwise NP-hard clustering problem tractable!
Problem Setting

- Input is $X = \{x\}_{i=1}^{n} \subset \mathbb{R}^d$
- Learner can ask same-cluster queries from the oracle
- The goal is to recover the oracle’s clustering $C^* = (C_1^*, \ldots, C_k^*)$
- $n$ and $d$ could be very large
Problem Setting

- Input is $X = \{x\}_{i=1}^n \subset \mathbb{R}^d$
- Learner can ask same-cluster queries from the oracle
- The goal is to recover the oracle’s clustering $C^* = (C_1^*, \ldots, C_k^*)$
- $n$ and $d$ could be very large

Still need more structure/inductive-bias ...

- “No-free-lunch” in clustering
  - Need to ask $\Omega(n)$ queries!
- Target $C^*$ is “nice”
$\mathcal{C} = \{C_1, \ldots, C_k\}$ with centers $\{\mu_1, \mu_2, \ldots, \mu_k\}$ satisfies the $\gamma$-margin property if for all $x \in C_i$ and $y \in X \setminus C_i$,

$$(1 + \gamma)d(x, \mu_i) < d(y, \mu_i)$$
Query complexity?

Computational complexity?
**Computational Complexity**

- **Special case:** oracle’s clustering is the solution of K-means

- Learner just wants to find K-means’ optimal solution
  - \( \min_{\{\mu\}_j} \sum_{x \in X} \min_j \|x - \mu_j\|_2^2 \)
  - Fixed objective, no under-specificity issue
  - Additional structure: \( \gamma \)-margin property

\[ \min_{\{\mu\}_j} \sum_{x \in X} \min_j \|x - \mu_j\|_2^2 \]
Computational Complexity

- **Special case:** oracle’s clustering is the solution of **K-means**
- Learner just wants to find K-means’ optimal solution
  - \[ \min_{\{\mu\}} \sum_{x \in X} \min_j \|x - \mu_j\|_2^2 \]
- Fixed objective, no under-speciality issue
- Additional structure: \(\gamma\)-margin property

**Theorem**

Solving Euclidean K-means clustering with no queries is **NP-hard** if \(\gamma \leq 0.84\).
Without queries:

- NP-hard for realistic values of $\gamma$
- Tractable only for unrealistically large values of $\gamma$
Positive Result (with Queries)

**Theorem**

There is an algorithm that finds $C^*$ for any $\gamma > 0$ (with constant probability) which

- Runs in $O(knd)$
- Asks $O(k^2 \log k + k \log n)$ queries

- Works for any "nice" target
- Query complexity is
  - Dimension-independent!
  - Only logarithmic in $n$
Surprising Conclusion

For the realistic situation $0 \leq \gamma \leq 0.84$

Clustering is NP-hard without queries

BUT

tractable with a small number of queries
Algorithm

**Algorithm’s Idea**

1. Estimate a center
   - Query uniformly till we have “enough” points from one cluster

2. Prune points belonging to that cluster
   - Binary search to find the “effective radius”

3. Repeat for the other clusters
Algorithm

Compute distance of all points to the 'green' point.
Sort them.

No need to know $k$ or $\gamma$ in advance
Hardness Result

Euclidean $k$-means is NP-hard even when the optimal solution satisfies the $\gamma$-margin property for $\gamma < 0.84$.

- True even with $O(\log n)$ same-cluster queries
- Reduction from Exact Cover by 3-Sets

Figure 1: Geometry of $H_{i,m}$. This figure is similar to Fig. 1 in [Vat09]. Reading from left to right, each row $R_i$ consists of a diamond $(s_i)$, $6m + 1$ bullets

Figure 2: The locations of $x_{i,j}$, $x'_{i,j}$, $y_{i,j}$
Clustering is an under-specified problem

Domain knowledge can be conveyed using interaction

An efficient algorithm for interactive clustering

Same-cluster queries can also reduce computational complexity

An NP-hard clustering problem became tractable using a few queries

Can we exploit same-cluster queries in other settings?
Thank You!