Representation Learning for Clustering: A Statistical Framework

Hassan Ashtiani and Shai Ben-David
School of Computer Science, University of Waterloo

Motivation

- There are tons of algorithmic choices for clustering.
- Each of these choices results in a different outcome.
- We have to use the domain knowledge to choose between these options.
- What kind of protocol/framework should we use to communicate prior knowledge?
- What kind of model should we use to leverage this knowledge?
- What kind of guarantees can we expect?

Contributions

- We propose a framework for incorporating domain knowledge into clustering.
- In this framework, the domain expert provides a clustering of a relatively small random sample of the data set
- An algorithm uses this to come up with a data representation under which k-means clustering results in a clustering that is consistent with the domain knowledge.
- We provide a formal statistical model for analyzing the sample complexity of learning a clustering representation with this paradigm.
- We introduce a notion of capacity of a class of possible representations, in the spirit of the VC-dimension, showing that classes of representations that have finite such dimension can be successfully learned with sample size error bounds

Definitions

- X: The domain
- $f : X \mapsto \mathbb{R}^d$
- C_X^f: The clustering of X induced by first mapping the data by f and then doing k-means clustering
- $\mathcal{F}$: A class of mappings from X to $\mathbb{R}^d$
- C^*: Optimal (unknown) k-clustering of X
- Algorithm $A(S, C_S^*)$ takes a sample $S \subset X$ and its clustering C_S^*, and outputs a mapping $f_A \in \mathcal{F}$
- The error is the $\Delta_X(C^*, C_X^{f_A})$ (the difference between C^* and the clustering induced by f_A)
- A natural choice of distance between two k-clusterings:

$$\Delta_X(C^1, C^2) = \min_{\sigma \in \pi^k} \frac{1}{|X|} \sum_{i=1}^{k} |C_i^1 \Delta C_{\sigma(i)}^2|$$

PAC-Type Framework

Let $\mathcal{F}$ be a set of mappings from X to $\mathbb{R}^d$. A representation learning algorithm A is a PAC-SRLK with sample complexity $m_{\mathcal{F}} : (0,1)^2 \mapsto \mathbb{N}$ with respect to $\mathcal{F}$, if for every $(\epsilon, \delta) \in (0,1)^2$, every domain set X and every clustering of X, C^*, the following holds:

For every X and C^*, if S is a randomly (uniformly) selected subset of X of size at least $m_{\mathcal{F}}(\epsilon, \delta)$, then with probability at least $1-\delta$

$$\Delta_X(C^*, C_X^{f_A}) \leq \inf_{f \in \mathcal{F}} \Delta_X(C^*, C_X^f) + \epsilon$$

TERM Algorithm

A Transductive Empirical Risk Minimizer (TERM) for $\mathcal{F}$ takes as input a sample $S \subset X$ and its clustering Y and outputs:

$$A^{TERM}(S, Y) = \arg\min_{f \in \mathcal{F}} \Delta_S(C_X^f\big|_S, Y)$$

- It finds the mapping based on which if you cluster X, the *empirical error* will be minimized.

Result

- Sample complexity of PAC-SRLK:

$$m_{\mathcal{F}}(\epsilon, \delta) \leq \mathcal{O}(\frac{k + Pdim(\mathcal{F}) + \log(\frac{1}{\delta})}{\epsilon^2})$$

where $\mathcal{O}$ hides logarithmic factors.

- Let $\mathcal{F}$ be a set of *linear* mappings from $\mathbb{R}^{d_1}$ to $\mathbb{R}^{d_2}$. Then

$$m_{\mathcal{F}}(\epsilon, \delta) \leq \mathcal{O}(\frac{k + d_1 d_2 + \log(\frac{1}{\delta})}{\epsilon^2})$$

- Pseudo-dimension: the size of the largest pseudo-shattered set (real-valued functions)
- We have defined a vector-valued version of it

Uniqueness Assumption

- k-means' solution may not be unique for some mappings
- Such mappings should not be selected!
- We should compare the the output of the algorithm only to those mappings in $\mathcal{F}$ that have unique solutions
- (η, ϵ)-*Uniqueness*: Every η-optimal solution to k-means' cost is ϵ-close to the optimal solution

Proof Sketch

Sketch:

1. Bound $Pdim(\mathcal{F})$
2. Bound $\mathcal{N}(\mathcal{F}, d_{L_1}^X, \epsilon)$ based on $Pdim(\mathcal{F})$ and ϵ
3. Bound $\mathcal{N}(\mathcal{F}, \Delta_X, \epsilon)$ based on $\mathcal{N}(\mathcal{F}, d_{L_1}^X, \epsilon)$
4. Bound the $m_{UC}^{\mathcal{F}}(\epsilon, \delta)$ based on δ and $\mathcal{N}(\mathcal{F}, \Delta_X, \epsilon)$
5. Bound $m^{\mathcal{F}}(\epsilon, \delta)$ based on $m_{UC}^{\mathcal{F}}(\epsilon, \delta)$

Covering Number

- $d(.,.)$: a metric over $\mathcal{F}$
- Δ-distance between two mappings:

$$\Delta_X(f_1, f_2) = \Delta_X(C_X^{f_1}, C_X^{f_2})$$

- L_1 distance between two mappings:

$$d_{L_1}^X(f_1, f_2) = \frac{1}{|X|} \sum_{x \in X} \|f_1(x) - f_2(x)\|_2$$

- $\mathcal{N}(\mathcal{F}, d, \epsilon)$ or covering number: Roughly, the number of ϵ-different members of $\mathcal{F}$ with respect to $d(.,.)$