Complementi di Algoritmi e Strutture Dati

Correlation Clustering

Docente: *Nicolò Cesa-Bianchi* versione 17 aprile 2025

Clustering is a central problem in unsupervised learning. A clustering problem is typically represented by a set of elements together with a notion of similarity (or dissimilarity) between them. When the elements are points in a metric space, dissimilarity can be measured via a distance function. In more general settings, when the elements to be clustered are members of an abstract set V, similarity is defined by an arbitrary symmetric function σ defined on pairs of distinct elements in V.

Correlation Clustering (CC) is a well-known special case where σ is a $\{-1,+1\}$-valued function establishing whether any two distinct elements of V are similar or not. The objective of CC is to cluster the points in V so to minimize the number of errors, where an error is given by any pair of elements having similarity -1 and belonging to the same cluster, or having similarity $+1$ and belonging to different clusters. Importantly, there are no a priori limitations on the number of clusters or their sizes: all partitions of V, including the trivial ones, are valid. Given V and σ, the error achieved by an optimal clustering is known as the *Correlation Clustering index*, denoted by OPT.

Note that OPT $= 0$ implies that V can be perfectly clustered: any two elements in the same cluster have similarity $+1$ and any two elements in different clusters have similarity -1. Since its introduction, CC has attracted a lot of interest and has found numerous applications in entity resolution, image analysis, and social media analysis.

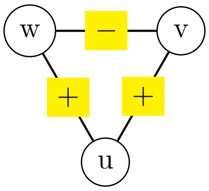
Minimizing the correlation clustering error is hard, and the best efficient algorithm found so far achieves 2OPT (the actual coefficient multiplying OPT is slightly smaller than 2). A very simple and elegant algorithm for approximating CC is KwikCluster. At each round, KwikCluster draws a random pivot π_r from V, queries the similarities between π_r and every other node in V, and creates a cluster C containing π_r and all points u such that $\sigma(\pi_r, u) = +1$. The algorithm then recursively invokes itself on $V \setminus C$. On any instance of CC, KwikCluster achieves an expected error bounded by 3OPT.

Algoritmo 1 KwikCluster

```
Parameters: residual node set $V_r$, round index $r$
if $|V_r| = 0$ then RETURN
end if
if $|V_r| = 1$ then output singleton cluster $V_r$ and RETURN
end if
Draw pivot $\pi_r$ u.a.r. from $V_r$
$C_r \leftarrow \{\pi_r\}$                                  ▷ Create new cluster and add the pivot to it
$C_r \leftarrow C_r \cup \{u \in V_r : \sigma(\pi_r, u) = +1\}$             ▷ Populate cluster
Output cluster $C_r$
KwikCluster($V_r \setminus C_r, r+1$)                        ▷ Recursive call on the remaining nodes
```

We denote by $V \equiv \{1,\dots,n\}$ the set of input nodes, by $\mathcal{E} \equiv \binom{V}{2}$ the set of all pairs $\{u,v\}$ of distincts nodes in V, and by $\sigma : \mathcal{E} \to \{-1,+1\}$ the binary similarity function. A clustering $\mathcal{C}$ is a partition of V in disjoint clusters $C_i : i = 1,\dots,k$. Given $\mathcal{C}$ and σ, the set $\Gamma_{\mathcal{C}}$ of mistaken edges contains all pairs $\{u,v\}$ such that $\sigma(u,v) = -1$ and u,v belong to same cluster of $\mathcal{C}$ and all pairs $\{u,v\}$ such that $\sigma(u,v) = +1$ and u,v belong to different clusters of $\mathcal{C}$. The cost of $\mathcal{C}$ is $\left|\Gamma_{\mathcal{C}}\right|$. The correlation clustering index is $\mathrm{OPT} = \min_{\mathcal{C}} \left|\Gamma_{\mathcal{C}}\right|$, where the minimum is over all clusterings $\mathcal{C}$.

A triangle is any unordered triple $T = \{u,v,w\} \subseteq V$. We denote by $e = \{u,w\}$ a generic triangle edge; we write $e \subset T$ and $v = T \setminus e$. We say T is a *bad triangle* if the labels $\sigma(u,v), \sigma(u,w), \sigma(v,w)$ are $\{+,+,-\}$ (the order is irrelevant), see the figure below here.

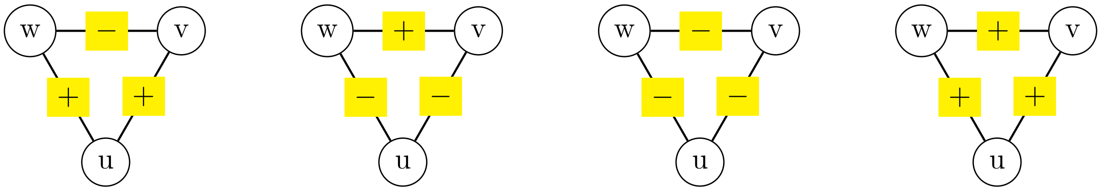


We denote by $\mathcal{T}$ the set of all bad triangles in V and also define $\mathcal{T}(e) \equiv \{T \in \mathcal{T} : e \subset T\}$. It is easy to see that the number of edge-disjoint bad triangles is a lower bound on OPT: no matter how we cluster its nodes, a bad triangle contributes by at least 1 to the total cost of the partition. The following lemma (which we state without proof) shows that also the weighted sum of all bad triangles is a lower bound on OPT, provided the sum of the weights of all bad triangles insisting on any single edge e is at most 1.

Lemma 1 *If $\{\beta_T \geq 0 : T \in \mathcal{T}\}$ is a set of weights on the bad triangles such that $\sum_{T \in \mathcal{T}(e)} \beta_T \leq 1$ for all $e \in \mathcal{E}$, then $\sum_{T \in \mathcal{T}} \beta_T \leq \mathrm{OPT}$.*

We now bound the expected error of KwikCluster. We use V_r to denote the set of remaining nodes at the beginning of the r-th recursive call.

Let Γ_A be the set of mistaken edges for the clustering output by KwikCluster and let $\left|\Gamma_A\right|$ be the cost of this clustering.



Lemma 2 *For any $e \in \mathcal{E}$, $e \in \Gamma_A$ if and only if there exists a recursive call r and a bad triangle T such that $T \subseteq V_r$, $T \in \mathcal{T}(e)$, and $\pi_r = T \setminus e$.*

Dimostrazione. Fix any edge e and let r be the round in which at least one of the two nodes of e are removed from V_r. Then KwikCluster makes a mistake on e if and only if e forms a bad triangle with π_r and $\pi_r = T \setminus e$ (see the pictures above here). Hence, if $e \in \Gamma_A$, then there exists a round r

and a bad triangle $T \subseteq V_r$ such that: $\pi_r = T \setminus e$ and $T \in \mathcal{T}(e)$. We prove the other direction via a case analysis with $e = \{u, w\}$. Assume $T = \{u, \pi_r, w\} \subseteq V_r$, $T \in \mathcal{T}(e)$, and $\pi_r = T \setminus e$.

Case 1: $\sigma(u, w) = +1$. If $\sigma(u, w) = +1$, $\pi_r = T \backslash e$, and T is a bad triangle, then $\sigma(\pi_r, w) \neq \sigma(\pi_r, u)$. But then u and w must end up in different clusters, which implies that e is mistaken.

Case 2: $\sigma(u, w) = -1$. If $\sigma(u, w) = -1$ $\pi_r = T \setminus e$, and T is a bad triangle, then $\sigma(\pi_r, u) = \sigma(\pi_r, v) = +1$. But then u and w end up in the same cluster, which implies that e is mistaken. □

Lemma 2 implies that at round r we make a mistake on exactly one edge of every bad triangle T such that $T \subset V_r$ and $\pi_r \in T$. Recall that a bad triangle T can be mistaken only once, because when $\pi_r \in T$, π_r gets removed from V_r. Hence, for any realization $\pi_1, \pi_2, \dots$ of the random pivot sequence,

$$\big|\Gamma_A\big| = \sum_{T \in \mathcal{T}} \mathbb{I}\{(\exists\, r) \,:\, T \subseteq V_r \,\wedge\, \pi_r \in T\} \ .$$

For all $T \in \mathcal{T}$ tet A_T be the event $\big\{(\exists\, r) \,:\, T \subseteq V_r \,\wedge\, \pi_r \in T\big\}$ indicating that T contributes to one mistake.

Note that for any $e \in \Gamma_A$ and for any two distinct $T, T' \in \mathcal{T}(e)$, A_T and $A_{T'}$ can not both occur because e is mistaken when it forms a bad triangle with π_r, which can only occur for at most one between T and T'. Thus, for any e,

$$\sum_{T \in \mathcal{T}(e)} \mathbb{I}\{A_T \,\wedge\, e \in \Gamma_A\} = 1 \ .$$

Taking expectations with respect to the random choice of the pivot sequence,

$$1 = \sum_{T \in \mathcal{T}(e)} \mathbb{P}\big(A_T \,\wedge\, e \in \Gamma_A\big) = \sum_{T \in \mathcal{T}(e)} \mathbb{P}\big(e \in \Gamma_A \mid A_T\big)\mathbb{P}(A_T) = \sum_{T \in \mathcal{T}(e)} \frac{1}{3}\mathbb{P}(A_T) \ .$$

where $\mathbb{P}\big(e \in \Gamma_A \mid A_T\big) = \frac{1}{3}$ holds because, given some r such that $T \subseteq V_r$ and $\pi_r \in T$, e is mistaken only if $\pi_r \in T \setminus e$.

Applying Lemma 1 with $\beta_T = \frac{1}{3}\mathbb{P}(A_T)$ for each $T \in \mathcal{T}$, we get

$$\mathbb{E}\big[|\Gamma_A|\big] = \sum_{T \in \mathcal{T}} \mathbb{P}(A_T) = 3 \sum_{T \in \mathcal{T}} \beta_T \leq 3\mathrm{OPT} \ .$$