Hierarchical Clustering in Large Networks

Vladimir Batagelj
Anuška Ferligoj
Andrej Mrvar
University of Ljubljana

Detection, evolution and visualization of communities in complex networks
Louvain-la-Neuve, Belgium, March 13-14, 2008.
Outline

1 Regionalization problem .................................................. 1
2 Clustering with relational constraint .................................... 2
5 Agglomerative method for relational constraints ...................... 5
7 Dissimilarities between clusters ......................................... 7
10 Reducibility ................................................................. 10
11 Example: US counties / maximum, $t = 1400$ .......................... 11
13 Hierarchical clustering in two-mode networks ....................... 13
16 Conditions for hierarchical methods .................................... 16
19 References ................................................................. 19

Based on Hierarchical clustering with relational constraints of large data sets presented at 6th Slovenian International Conference on Graph Theory, Bled, Slovenia, 24 – 30 June 2007.

Regionalization problem

Group given territorial units into regions such that units inside the region will be similar according to selected properties (attributes, variables) and form contiguous part of the territory.

In Ferligoj and Batagelj (1982 and 1983) we generalized this problem to clustering with relational constraints problem and proposed some algorithms to solve it.
Clustering with relational constraint

Suppose that the units are described by attribute data \( a: \mathcal{U} \to [\mathcal{U}] \) and related by a binary relation \( R \subseteq \mathcal{U} \times \mathcal{U} \) that determine the relational data \( (\mathcal{U}, R, a) \).

We want to cluster the units according to the similarity of their descriptions, but also considering the relation \( R \) – it imposes constraints on the set of feasible clusterings, usually in the following form:

\[
\Phi(R) = \{ C \in \mathcal{P}(\mathcal{U}) : \text{each cluster } C \in C \text{ induces a subgraph } (C, R \cap C \times C) \text{ in the graph } (\mathcal{U}, R) \text{ of the required type of connectedness} \}
\]
...Clustering with relational constraints

We can define different types of sets of feasible clusterings for the same relation $R$. Some examples of types of relational constraint $\Phi^i(R)$ are

<table>
<thead>
<tr>
<th>type of clusterings</th>
<th>type of connectedness</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\Phi^1(R)$</td>
<td>weakly connected units</td>
</tr>
<tr>
<td>$\Phi^2(R)$</td>
<td>weakly connected units that contain at most one center</td>
</tr>
<tr>
<td>$\Phi^3(R)$</td>
<td>strongly connected units</td>
</tr>
<tr>
<td>$\Phi^4(R)$</td>
<td>clique</td>
</tr>
<tr>
<td>$\Phi^5(R)$</td>
<td>the existence of a trail containing all the units of the cluster</td>
</tr>
</tbody>
</table>

Trail – all arcs are distinct.

A set of units $L \subseteq C$ is a center of cluster $C$ in the clustering of type $\Phi^2(R)$ iff the subgraph induced by $L$ is strongly connected and $R(L) \cap (C \setminus L) = \emptyset$. 
Some graphs of different types

1. A clique

2. Strongly connected units

3. Weakly connected units

4. Weakly connected units with a center \{1, 2, 4\}
Agglomerative method for relational constraints

We can use both hierarchical and local optimization methods for solving some types of problems with relational constraint (Ferligoj, Batagelj 1983).

1. \( k := n; \mathbf{C}(k) := \{\{X\} : X \in \mathcal{U}\} \);  
2. \( \textbf{while } \exists C_i, C_j \in \mathbf{C}(k): (i \neq j \land \psi(C_i, C_j)) \textbf{ repeat} \)
   2.1. \((C_p, C_q) := \text{argmin}\{D(C_i, C_j) : i \neq j \land \psi(C_i, C_j)\}\);  
   2.2. \( C := C_p \cup C_q; k := k - 1; \)
   2.3. \( \mathbf{C}(k) := \mathbf{C}(k + 1) \setminus \{C_p, C_q\} \cup \{C\}; \)
   2.4. determine \( D(C, C_s) \) for all \( C_s \in \mathbf{C}(k) \)
   2.5. adjust the relation \( R \) as required by the clustering type
3. \( m := k \)

The fusibility condition \( \psi(C_i, C_j) \) is equivalent to \( C_i R C_j \) for tolerant, leader and strict method; and to \( C_i R C_j \land C_j R C_i \) for two-way method.
Adjusting relation after joining

\( \Phi^1 \) – tolerant
\( \Phi^2 \) – leader
\( \Phi^4 \) – two-way
\( \Phi^5 \) – strict
Dissimilarities between clusters

In the original approach a complete dissimilarity matrix is needed. To obtain fast algorithms we propose to consider only the dissimilarities between linked units.

Let \((\mathcal{U}, R), R \subseteq \mathcal{U} \times \mathcal{U}\) be a graph and \(\emptyset \subset S, T \subset \mathcal{U}\) and \(S \cap T = \emptyset\).

We call a block of relation \(R\) for \(S\) and \(T\) its part \(R(S, T) = R \cap S \times T\).

The symmetric closure of relation \(R\) we denote with \(\hat{R} = R \cup R^{-1}\). It holds: \(\hat{R}(S, T) = \hat{R}(T, S)\).

For all dissimilarities between clusters \(D(S, T)\) we set:

\[
D(\{s\}, \{t\}) = \begin{cases} 
  d(s, t) & s \hat{R} t \\
  \infty & \text{otherwise}
\end{cases}
\]

where \(d\) is a selected dissimilarity between units.
Minimum

\[ D_{\min}(S, T) = \min_{(s, t) \in \hat{R}(S, T)} d(s, t) \]

\[ D_{\min}(S, T_1 \cup T_2) = \min(D_{\min}(S, T_1), D_{\min}(S, T_2)) \]

Maximum

\[ D_{\max}(S, T) = \max_{(s, t) \in \hat{R}(S, T)} d(s, t) \]

\[ D_{\max}(S, T_1 \cup T_2) = \max(D_{\max}(S, T_1), D_{\max}(S, T_2)) \]
\[ w : V \rightarrow \mathbb{R} \text{ – is a weight on units; for example } w(v) = 1, \text{ for all } v \in \mathcal{U}. \]

\[
D_a(S, T) = \frac{1}{w(\hat{R}(S, T))} \sum_{(s,t) \in \hat{R}(S, T)} d(s, t)
\]

\[
w(\hat{R}(S, T_1 \cup T_2)) = w(\hat{R}(S, T_1)) + w(\hat{R}(S, T_2))
\]

\[
D_a(S, T_1 \cup T_2) = \frac{w(\hat{R}(S, T_1))}{w(\hat{R}(S, T_1 \cup T_2))} D_a(S, T_1) + \frac{w(\hat{R}(S, T_2))}{w(\hat{R}(S, T_1 \cup T_2))} D_a(S, T_2)
\]
Reducibility

The dissimilarity $D$ has the reducibility property (Bruynooghe, 1977) iff

$$D(C_p, C_q) \leq \min(D(C_p, C_s), D(C_q, C_s)) \Rightarrow$$

$$\min(D(C_p, C_s), D(C_q, C_s)) \leq D(C_p \cup C_q, C_s)$$

or equivalently

$$D(C_p, C_q) \leq t, \ D(C_p, C_s) \geq t, \ D(C_q, C_s) \geq t \Rightarrow \ D(C_p \cup C_q, C_s) \geq t$$

**Theorem 1**  *If a dissimilarity $D$ has the reducibility property then $h_D$ is a level function.*

All three dissimilarities have the reducibility property. In this case also the nearest neighbors network for a given network is preserved after joining the nearest clusters. This allows us to develop a very fast agglomerative hierarchical clustering procedure. It is available in program **Pajek**.
Example: US counties / maximum, $t = 1400$

Example: US counties / maximum, $t = 200$
Hierarchical clustering in two-mode networks

At the Bertinoro workshop on graph drawing (9-14. March 2008) Katharina Zweig (Lehmann) asked for a method for clustering bipartite graphs. It turned out that our method can be easily adapted to solve also this problem.

Let \(((U, V), L, d)\), \(d : L \rightarrow \mathbb{R}^+_0\) be a weighted two-mode (bipartite) network. \(d\) is a dissimilarity measure between linked units (vertices). As an example of such dissimilarity we can take \(d(u, v) = d(v, u) = w^* - w_4(u, v)\) where \(w_4(u, v)\) is the number of different 4-cycles containing the line \((u, v)\) and \(w^* = \max_{(u,v) \in L} w_4(u, v)\).

We call a \textit{two-mode partition} a set

\[
C = \{ (C_1, D_1), (C_2, D_2), \ldots, (C_k, D_k) \}
\]

such that the sets \(\{C_i\} \ \backslash \ \emptyset\) and \(\{D_i\} \ \backslash \ \emptyset\) are partitions of sets \(U\) and \(V\); and \((\emptyset, \emptyset) \notin C\).
Hierarchical clustering in two-mode networks

We start the hierarchical clustering with the singeltons partition

\[ C_N = U^{(1)} \times \{ \emptyset \} \cup \{ \emptyset \} \times V^{(1)} = \{(\{u_1\}, \emptyset), (\{u_2\}, \emptyset), \ldots, (\emptyset, \{v_n\})\} \]

where \(U^{(1)} = \{\{u\} : u \in U\}\) and \(N = |U \cup V|\).

The fusibility condition is now \(\psi((C_i, D_i), (C_j, D_j)) = C_i RD_j \lor C_j RD_i\)

where \(C_i RD_j = \exists u \in C_i \exists v \in D_j : (u, v) \in L\).

The relation \(R\) update rules are

\[ C_{pq} RC_s = C_p RC_s \lor C_q RC_s \quad \text{and} \quad C_s RC_{pq} = C_s RC_p \lor C_s RC_q \]
...Hierarchical clustering in two-mode networks

For a pair of two-mode clusters \((C_p, D_p)\) and \((C_q, D_q)\), such that \(C_p RD_q\), let \(\delta(p, q)\) denote the difference of cluster \(p\) from cluster \(q\). \(\delta\) is determined as follows

\[
\delta(\{u\}, \{v\}) = d(u, v)
\]

and the update formulae for \(\delta\) after merging are: if \(C_p RD_q \land C_q RD_p\) then

\[
\delta((pq), s) = \oplus(\delta(p, s), \delta(q, s)) \quad \delta(s, (pq)) = \oplus(\delta(s, p), \delta(s, q))
\]

where \(\oplus \in \{\min, \max, \text{ave}\}\); otherwise only values of existing links are considered.

Using \(\delta\) the clustering dissimilarity \(D(p, q)\) is defined as

\[
D(p, q) = \oplus(\delta(p, q), \delta(q, p))
\]

Again the link needs not to exist in both ’directions’.
Conditions for hierarchical methods

The set of feasible clusterings \( \Phi \) determines the \textit{feasibility predicate} \( \Phi(C) \equiv C \in \Phi \) defined on \( \mathcal{P}(\mathcal{P}(U) \setminus \{\emptyset\}) \); and conversely \( \Phi \equiv \{C \in \mathcal{P}(\mathcal{P}(U) \setminus \{\emptyset\}) : \Phi(C)\} \).

In the set \( \Phi \) the relation of \textit{clustering inclusion} \( \sqsubseteq \) can be introduced by
\[
C_1 \sqsubseteq C_2 \equiv \forall C_1 \in C_1, C_2 \in C_2 : C_1 \cap C_2 \in \{\emptyset, C_1\}
\]

we say also that the clustering \( C_1 \) is a \textit{refinement} of the clustering \( C_2 \).

It is well known that \( (\Pi(U), \sqsubseteq) \) is a partially ordered set (even more, semimodular lattice). Because any subset of partially ordered set is also partially ordered, we have: Let \( \Phi \subseteq \Pi(U) \) then \( (\Phi, \sqsubseteq) \) is a partially ordered set.

The clustering inclusion determines two related relations (on \( \Phi \)):
\[
C_1 \subset C_2 \equiv C_1 \sqsubseteq C_2 \land C_1 \neq C_2 \quad \text{– strict inclusion, and}
\]
\[
C_1 \sqsubset C_2 \equiv C_1 \sqsubseteq C_2 \land \neg \exists C \in \Phi : (C_1 \subset C \land C \sqsubset C_2) \quad \text{– predecessor.}
\]
Conditions on the structure of the set of feasible clusterings

We shall assume that the set of feasible clusterings $\Phi \subseteq \Pi(\mathcal{U})$ satisfies the following conditions:

**F1.** $\emptyset \equiv \{ \{X\} : X \in \mathcal{U} \} \in \Phi$

**F2.** The feasibility predicate $\Phi$ is _local_ – it has the form $\Phi(C) = \bigwedge_{C \in \mathcal{C}} \varphi(C)$ where $\varphi(C)$ is a predicate defined on $\mathcal{P}(\mathcal{U}) \setminus \{\emptyset\}$ (clusters).

The intuitive meaning of $\varphi(C)$ is: $\varphi(C) \equiv$ the cluster $C$ is ’good’. Therefore the locality condition can be read: a ’good’ clustering $C \in \Phi$ consists of ’good’ clusters.

**F3.** The predicate $\Phi$ has the property of _binary heredity_ with respect to the _fusibility_ predicate $\psi(C_1, C_2)$, i.e.,

$$C_1 \cap C_2 = \emptyset \land \varphi(C_1) \land \varphi(C_2) \land \psi(C_1, C_2) \Rightarrow \varphi(C_1 \cup C_2)$$

This condition means: in a ’good’ clustering, a fusion of two ’fusible’ clusters produces a ’good’ clustering.
...conditions

**F4.** The predicate $\psi$ is *compatible* with clustering inclusion $\sqsubseteq$, i.e.,

$$\forall C_1, C_2 \in \Phi : (C_1 \sqsubseteq C_2 \land C_1 \setminus C_2 = \{C_1, C_2\} \Rightarrow \psi(C_1, C_2) \lor \psi(C_2, C_1))$$

**F5.** The *interpolation* property holds in $\Phi$, i.e., $\forall C_1, C_2 \in \Phi :$

$$(C_1 \sqsubseteq C_2 \land \text{card}(())C_1) > \text{card}(())C_2 + 1 \Rightarrow \exists C \in \Phi : (C_1 \sqsubseteq C \land C \sqsubseteq C_2))$$

These conditions provide a framework in which the hierarchical methods can be applied also for constrained clustering problems $\Phi_k(U) \subset \Pi_k(U)$.

In the ordinary problem both predicates $\varphi(C')$ and $\psi(C_p, C_q)$ are always true – all conditions F1-F5 are satisfied.
References


