New integer linear programming models for a variant of correlation clustering problem

Morshinin Aleksandr
Sobolev Institute of Mathematics SB RAS
The research was supported by RSF grant № 22-71-10015

Clustering problems

Clustering problems form an important section of data analysis. In these problems one has to partition a given set of objects into several subsets (*clusters*) basing only on similarity of the objects.

The aim of *correlation clustering* is to group the vertices of a graph into clusters taking into consideration the edge structure of the graph whose vertices are objects and edges represent similarities between the objects.

Basic definitions

A simple graph is called a *cluster graph* if each of its components is a complete graph. Components is called *clusters*.

Denote cluster graph with s components as $C(V_1, V_2, ..., V_s)$.

The *distance* $\rho(G_1, G_2)$ between $G_1 = (V, E_1)$ and $G_2 = (V, E_2)$

$$\rho(G_1, G_2) = | E_1 \setminus E_2 | + | E_2 \setminus E_1 | .$$

Correlation clustering

CC. For given graph $G = (V, E)$ find the nearest to G cluster graph $C(V_1, V_2, ..., V_s)$ with $2 \leq s \leq |V|$ clusters.

$\text{CC}_{\leq k}$. For given graph $G = (V, E)$ find the nearest to G cluster graph $C(V_1, V_2, ..., V_s)$ with $2 \leq s \leq k < |V|$ clusters.

Both problems are *NP*-hard.

Integer linear programming

Charikar, Guruswami, Wirth (2005) developed binary integer linear programming model for CC.

$$x_{ij} = \begin{cases} 0, \text{ if } i \text{ and } j \text{ in same cluster} \\ 1, \text{ otherwise} \end{cases}$$

If $x_{ij} = 0$ and $x_{jr} = 0$ then $x_{ir} = 0$. Therefore, $x_{ir} \leq x_{ij} + x_{jr}$ for all $i, j, r \in V$.

Integer linear programming

$$\sum_{ij \in E} x_{ij} + \sum_{ij \notin E} (1 - x_{ij}) \to \min$$

$$x_{ir} \le x_{ij} + x_{jr}, \text{ for all } i, j, r \in V$$

$$x_{ij} \in \{0, 1\}, \text{ for all } i, j \in V$$

Dewan and Hassini (2022) in their literature review of correlation clustering added the following constraint:

$$x_{ij} = x_{ji}, \text{ for all } i, j \in V$$

Integer linear programming

We can rewriter triangle inequality to use only unordered variables.

$$\sum_{ij \in E} x_{ij} + \sum_{ij \notin E} (1 - x_{ij}) \to \min$$

$$\begin{cases} x_{ir} \le x_{ij} + x_{jr}, \\ x_{ij} \le x_{ir} + x_{jr}, \text{ for all } i, j, r \in V \\ x_{jr} \le x_{ij} + x_{ir}, \end{cases}$$

$$x_{ij} \in \{0, 1\}, \text{ for all } i, j \in V$$

Integer linear programming for $\mathrm{CC}_{\le k}$

Feasible solution for $\mathrm{CC}_{\le k}$ must not contain null subgraph O_{k+1}.

$$x_{i_1 i_2} + \ldots + x_{i_k i_{k+1}} \le \frac{(k+2)(k-1)}{2}, \text{ for all } i_1, \ldots, i_{k+1} \in V$$

It's easy to see that both models (ordered and unordered) contain $O(n^2)$ variables and $O(n^{k+1})$ constraints. Thus, they are difficult to use for applications.

We will look at another approach that allows to reduce the number of variables and constraints.

Integer programming for $CC_{\leq 2}$

For each vertex $i \in V$ we introduce the following variables.

$$x_i = \begin{cases} 0, \text{ if } i \in V_1 \\ 1, \text{ if } i \in V_2 \end{cases}$$

Case 1. $ij \in E$. Then $|x_i - x_j| = 0$ only if $x_i = x_j$ and $|x_i - x_j| = 1$ only if $x_i \neq x_j$.
Case 2. $ij \notin E$. Then $|x_i + x_j - 1| = 0$ only if $x_i \neq x_j$ and $|x_i + x_j - 1| = 1$ only if $x_i = x_j$.

Integer programming for $CC_{\le 2}$

$$\sum_{ij\in E} |x_i - x_j| + \sum_{ij\notin E} |x_i + x_j - 1| \to \min$$

$$x_i \in \{0,1\}, \text{ for all } i \in V$$

This model is not linear. We need to make substitution to get linear model. Let's represent each modulus in objective as a sum of two binary variables $u_{ij} + v_{ij}$. We should also add constraints to model.

Integer linear programming for $\mathrm{CC}_{\leq 2}$

$$\sum_{i,j \in V} u_{ij} + v_{ij} \to \min$$

$$x_i - x_j + u_{ij} - v_{ij} = 0, \text{ for all } i, j \in V,\ ij \in E$$

$$x_i + x_j - 1 + u_{ij} - v_{ij} = 0, \text{ for all } i, j \in V,\ ij \notin E$$

$$x_1 = 0$$

$$x_i \in \{0,1\}, \text{ for all } i \in V$$

$$u_{ij} \in \{0,1\}, \text{ for all } i, j \in V$$

$$v_{ij} \in \{0,1\}, \text{ for all } i, j \in V$$

This model contains $O(n^2)$ variables and $O(n^2)$ constraints.

One-hot encoding for $\mathrm{CC}_{\leq k}$

For $\mathrm{CC}_{\leq k}$, $k \geq 3$ we can't use binary variables for belonging of vertices to clusters. But we can use *one-hot vector*.

$$x_{is} = \begin{cases} 1, \text{ if } i \in V_s \\ 0, \text{ otherwise} \end{cases}$$

For each $i \in V$ we build binary vector $[x_1, \ldots, x_k]$ with only one $x_{is} = 1$.

One-hot encoding for $\mathrm{CC}_{\le k}$

Case 1. $ij \in E$. Then $\frac{1}{2}\sum_{s=1}^{k} | x_{is} - x_{js} | = 0$ only if i and j belong to the same cluster. Otherwise, $\frac{1}{2}\sum_{s=1}^{k} | x_{is} - x_{js} | = 1$ only if i and j belong to different clusters.

Case 2. $ij \notin E$. Then $\frac{1}{2}\left(\sum_{s=1}^{k} | x_{is} + x_{js} - 1 | - k + 2\right) = 0$ only if i and j belong to different clusters. Otherwise, $\frac{1}{2}\left(\sum_{s=1}^{k} | x_{is} + x_{js} - 1 | - k + 2\right) = 1$ only if i and j belong to the same cluster.

Integer linear programming for $\mathrm{CC}_{\leq k}$

$$\sum_{i,j\in V}\sum_{s=1}^{k} u_{ijs} + v_{ijs} \to \min$$

$$x_{is} - x_{js} + u_{ijs} - v_{ijs} = 0, \text{ for all } i, j \in V,\ ij \in E,\ s = 1,\ldots,k$$

$$x_{is} + x_{js} - 1 + u_{ijs} - v_{ijs} = 0, \text{ for all } i, j \in V,\ ij \notin E,\ s = 1,\ldots,k$$

$$\sum_{s=1}^{k} x_{is} = 1, \text{ for all } i \in V$$

$$x_{11} = 0$$

$$x_i \in \{0,1\}, \text{ for all } i \in V$$

$$u_{ij} \in \{0,1\}, \text{ for all } i, j \in V$$

$$v_{ij} \in \{0,1\}, \text{ for all } i, j \in V$$

Integer linear programming for $\mathrm{CC}_{\leq k}$

For each vertex we have k variables and for each pair of vertices we have constraint. So, this model contains $O(k^2 n^2)$ variables and $O(k^2 n^2)$ constraints.

Experimental study

We tested three models (Ordered, Unordered and Modulus) for $CC_{\leq 2}$. All models were programming in Python with MIP library. We use Neos-Server as backend (CPLEX).

Random graphs were generated by *Erdos-Renyi model* $G(n, p)$ with the parameter $p \in \{0.33, 0.5, 0.67\}$.

Optimal solutions were found for graphs which contain from 20 to 50 vertices. For each pair of parameters 100 problems were solved.

Experimental study

n	$p = 0.33$			$p = 0.5$			$p = 0.67$		
	Or	Unor	Mod	Or	Unor	Mod	Or	Unor	Mod
20	4.96	0.54	0.16	5.87	0.77	0.24	3.27	0.27	0.20
25	32.13	3.78	1.22	43.96	5.25	2.08	21.08	0.96	0.63
30	348.34	21.17	4.51	319.26	43.85	9.79	59.60	3.45	3.06
35	-	123.0	18.21	-	240.38	36.31	-	7.34	6.78
40	-	812.46	96.29	-	952.19	162.91	-	16.68	24.38
45	-	-	499.40	-	-	623.76	-	28.83	26.08
50	-	-	3598.54	-	-	2153.23	-	63.90	53.17

Thank you!