Budapest University of Technology and Economics (BME)

A Distributed Genetic Algorithm for Graph-Based Clustering

Krisztian Buza
Budapest University of Technology and Economics, Hungary
buza@cs.bme.hu

Antal Buza, Piroska Buzáné Kis
Polytechnic of Dunaújváros, Hungary
{buza,piros}@mail.duf.hu

Introduction: Clustering

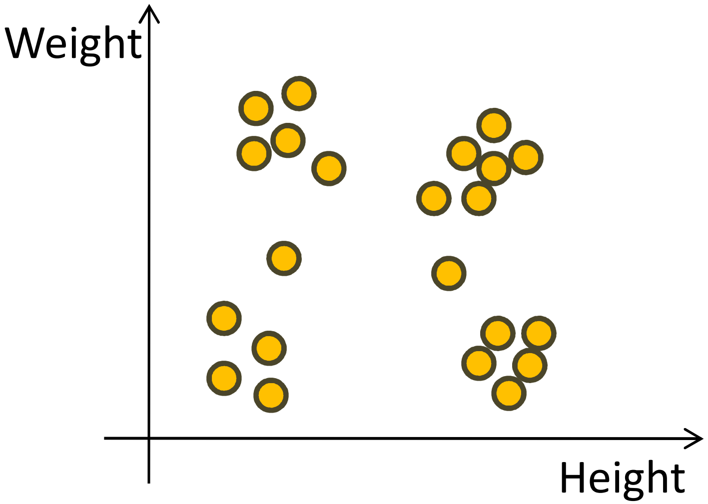
ID	Weight (kg)	Height (cm)
1	105	182
2	95	195
3	60	160
…	…	…

- Fundamental approach in analysis of massive datasets: explore the (high-level) structure of data

- Identification of groups so that similar objects belong to the same group, dissimilar objects belong to different groups

Introduction: Clustering

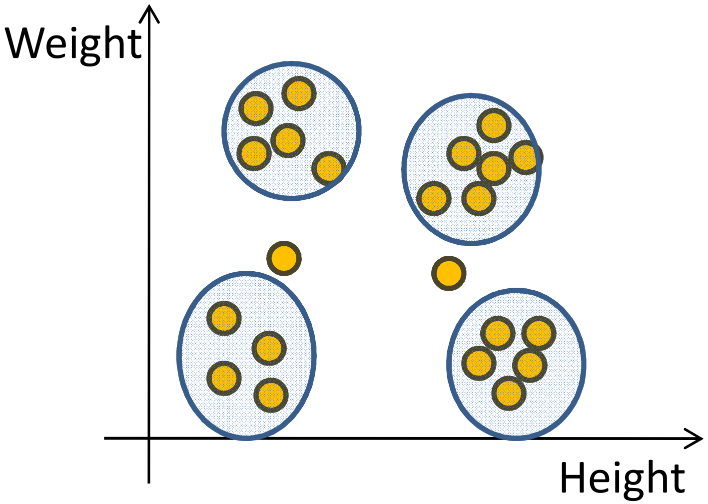
ID	Weight (kg)	Height (cm)
1	105	182
2	95	195
3	60	160
…	…	…



- Fundamental approach in analysis of massive datasets: explore the (high-level) structure of data
- Identification of groups so that similar objects belong to the same group, dissimilar objects belong to different groups

Introduction: Clustering

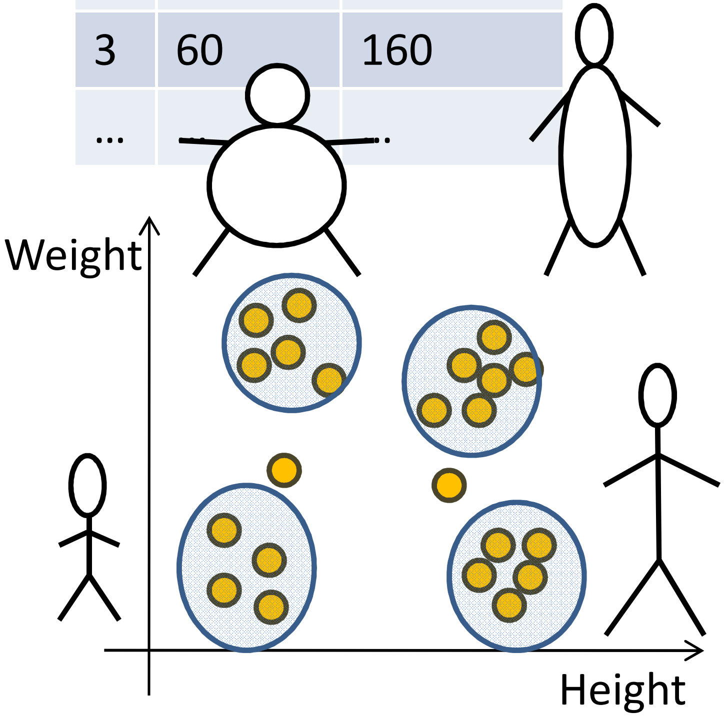
ID	Weight (kg)	Height (cm)
1	105	182
2	95	195
3	60	160
…	…	…



- Fundamental approach in analysis of massive datasets: explore the (high-level) structure of data
- Identification of groups so that similar objects belong to the same group, dissimilar objects belong to different groups

Introduction: Clustering

ID	Weight (kg)	Height (cm)
1	105	182
2	95	195
3	60	160
...		...



- Fundamental approach in analysis of massive datasets: explore the (high-level) structure of data

- Identification of groups so that similar objects belong to the same group, dissimilar objects belong to different groups

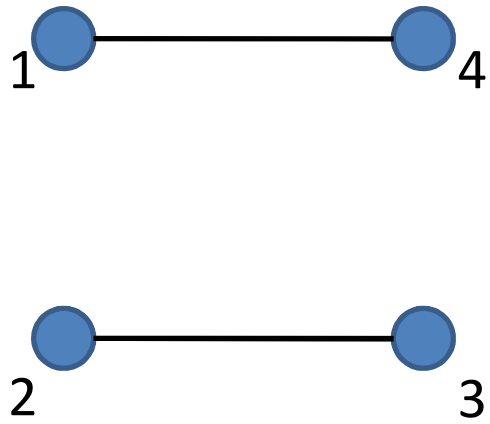
Problems with conventional clustering algorithms

- Mapping complex data into a vector space (categorical attributes, sequences, transactional data, heterogeneous data…)
- Curse of dimensionality
  - Distances and density become less meaningful
- Alleviation of the problem:
  - Objects → vertices of a graph
  - Graph-based clustering algorithms

Graph-based cluserting

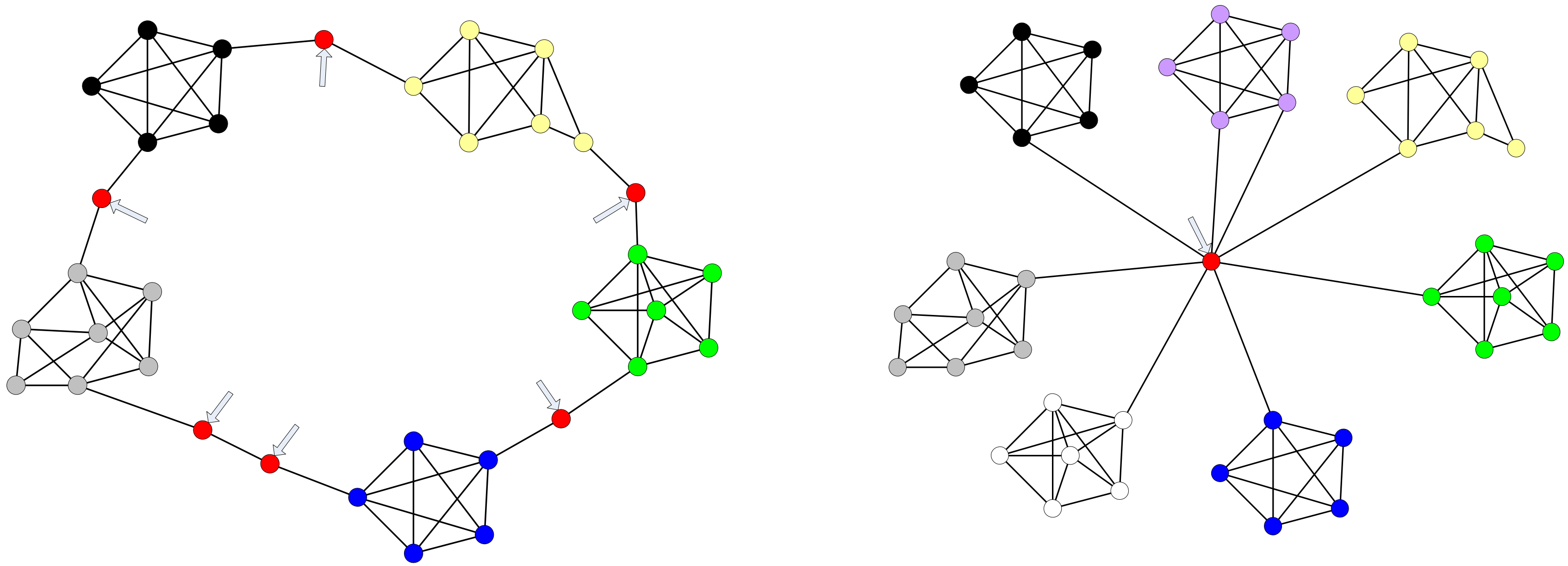
- Objects: vertices
- Two objects are similar → edge between the corresponding vertices

ID	Age	Weight	Height
1	59	70	170
2	35	85	180
3	36	86	179
4	64	69	172



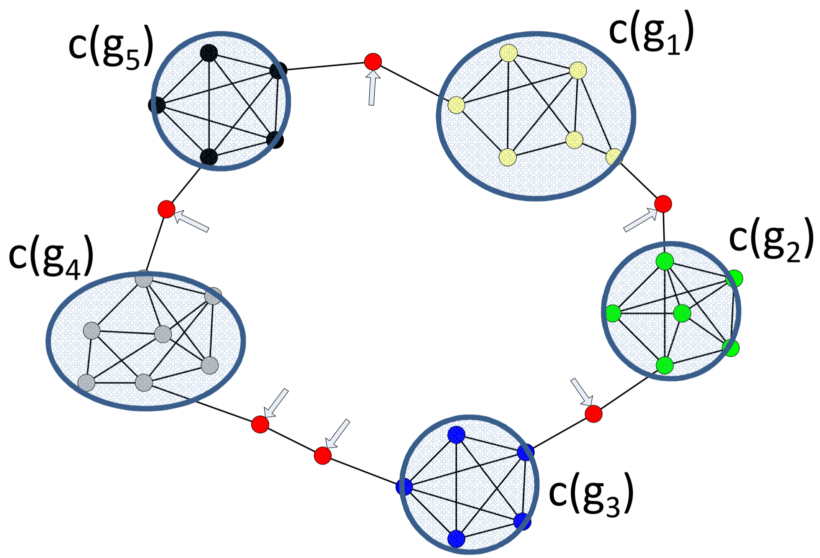
Our approach

- Search for a set of cutting vertices



Our approach

- Clustering kernel
  - subgraph-quality function: c(g): G → R
  - clustering-quality function: h: R^n → R
- Search for a set of cutting vertices so that the value returned by the clustering kernel is maximized



$h((c(g_1), c(g_2), c(g_3), c(g_4), c(g_5)))$

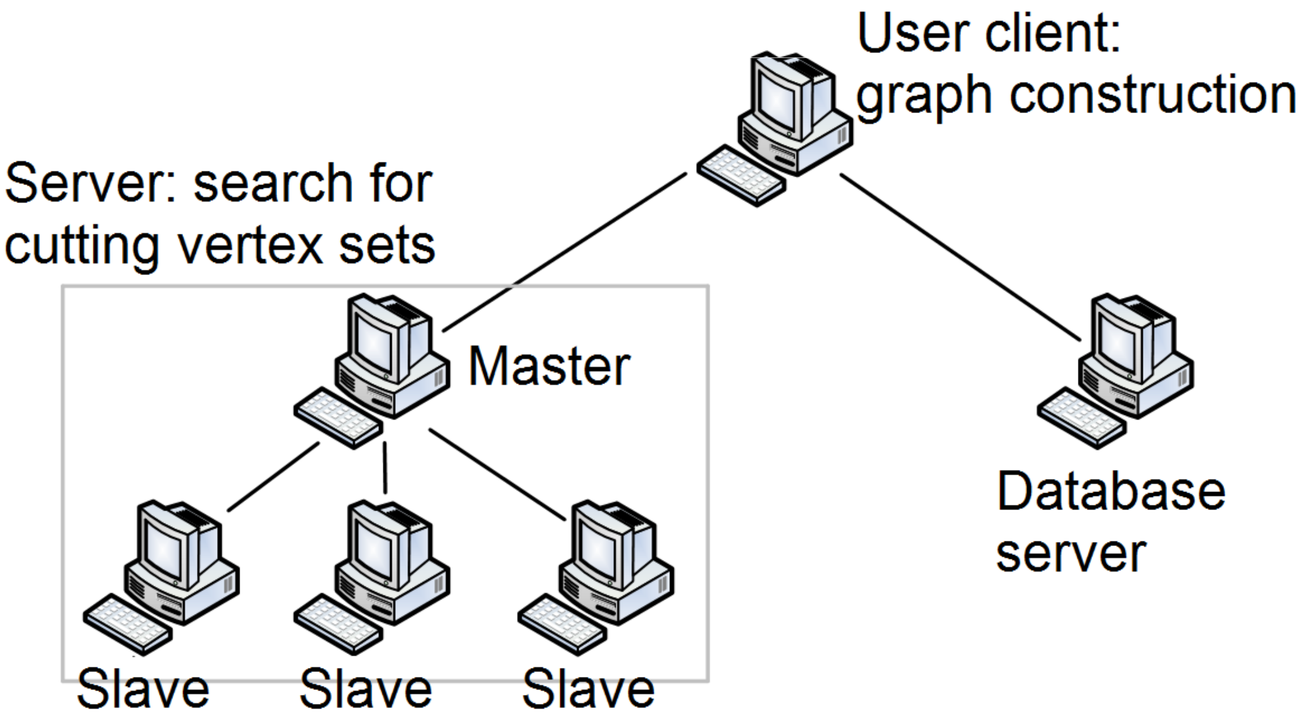
The above-defined problem is **NP-hard**
Proof: See Paper (reduction of the Clique-problem to our problem).

Our approach

- Search the set of cutting vertices with genetic algorithm
- Individuals: vertex sets
- Fitness function: clustering kernel
- Descendant s_3 of two individuals s_1 and s_2
  - Put all vertices of $s_1 \cap s_2$ into s_3.
  - Put each vertex from $s_1 \backslash s_2$ (and from $s_2 \backslash s_1$ respectively) into s_3 with a probability of 0.5.
  - We add (or remove) some random vertices to (from) s_3.

Our approach

- Implementation:
  distributed architecture – parallelize calculations



Experiments

- 3 benchmark tasks: Ring, Star, Tricky Star
- Clustering kernel:

$$c(g) = \frac{2|E(g)|}{|V(g)|(|V(g)|-1)} + \left(1 - \frac{1}{|V(g)|}\right) \quad , \quad h(\{x_1, ..., x_n\}) = \frac{x_1 + ... + x_n}{n}$$

- Vary the number/size of clusters

Table 1 Average number of generations in the genetic algorithm

Size of the graph[a]	Graph Type					
	STAR I	T.STAR I	RING I	STAR II	T.STAR II	RING II
100	21.00	19.33	18.66	20.00	19	28.5
200	25.00	24.67	32.66	26.00	24	33
300	27.67	27.00	41.00	27.00	25	40
400	29.67	28.33	43.66	30.00	30	40.5
500	31.00	29.33	56.00	30.50	30.5	40.5
...	...	...	...	...	...	...
1000	36.00	36.00	44.33	36.50	35	53

[a] total number of vertices, without the central vertex in case of Star and TrickyStar

Conclusion

- Graph-based clustering algorithm
- Quality function: clustering kernel
- Search for a set of cutting vertices with genetic algorithm (NP-hard in general)
- Implementation: distributed environment
- Experiments on benchmark tasks show scalability of the approach
- Experiments on real data: meaningful clusters