Exploiting the trade-off – the benefits of multiple objectives in data clustering

Julia Handl, Joshua Knowles

Denise de Oliveira, Marcelo C. Toyama

November 4, 2005

Outline

Introduction

- Clustering is finding groups with similar properties;
- It is a intuitive and loose concept;
- One evaluation function X multiple evaluation functions;

Introduction

- Clustering is finding groups with similar properties;
- It is a intuitive and loose concept;
- One evaluation function X multiple evaluation functions;

Introduction

- Clustering is finding groups with similar properties;
- It is a intuitive and loose concept;
- One evaluation function X multiple evaluation functions;

Outline

MOCK - multiobjective clustering with automatic determination of the number of clusters

- Multiple evaluation functions:
  - Compactness;
  - Connectedness;
- Automatic detection of number of clusters;

MOCK - multiobjective clustering with automatic determination of the number of clusters

- Multiple evaluation functions:
  - Compactness;
  - Connectedness;
- Automatic detection of number of clusters;

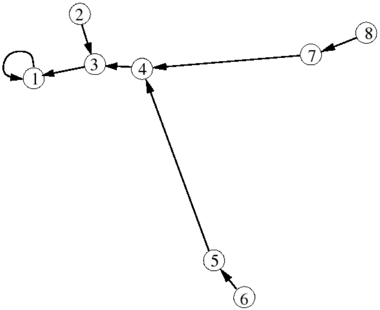
MOCK - multiobjective clustering with automatic determination of the number of clusters

- Multiple evaluation functions:
  - Compactness;
  - Connectedness;
- Automatic detection of number of clusters;

MOCK - multiobjective clustering with automatic determination of the number of clusters

- Multiple evaluation functions:
  - Compactness;
  - Connectedness;
- Automatic detection of number of clusters;

Genetic representation and operators



Order of connection:

1 to 1
3 to 1
4 to 3
2 to 3
7 to 4
8 to 7
5 to 4
6 to 5

Genotype:

1	3	1	3	4	5	4	7

Genetic representation and operators

- initialization exploits the link-based encoding and uses minimum spanning trees.
- Operators:
  - Uniform crossover.
  - Mutation operator that significantly reduces the size of the search space: each data item can only be linked to one of its L nearest neighbors.

Genetic representation and operators

- initialization exploits the link-based encoding and uses minimum spanning trees.
- Operators:
  - Uniform crossover.
  - Mutation operator that significantly reduces the size of the search space: each data item can only be linked to one of its L nearest neighbors.

Genetic representation and operators

- initialization exploits the link-based encoding and uses minimum spanning trees.
- Operators:
  - Uniform crossover.
  - Mutation operator that significantly reduces the size of the search space: each data item can only be linked to one of its L nearest neighbors.

Genetic representation and operators

- initialization exploits the link-based encoding and uses minimum spanning trees.
- Operators:
  - Uniform crossover.
  - Mutation operator that significantly reduces the size of the search space: each data item can only be linked to one of its L nearest neighbors.

Outline

Compactness of clusters

$$Dev(C) = \sum_{C_k \in C} \sum_{i \in C_k} \delta(i, \mu_k)$$

Where:

- C is the set of all clusters
- μ_k is the center of cluster C_k
- $\delta(i, \mu_k)$ is the chosen distance function.

As an objective, overall deviation should be minimized.

Compactness of clusters

$$Dev(C) = \sum_{C_k \in C} \sum_{i \in C_k} \delta(i, \mu_k)$$

Where:

- C is the set of all clusters
- μ_k is the center of cluster C_k
- $\delta(i, \mu_k)$ is the chosen distance function.

As an objective, overall deviation should be minimized.

Compactness of clusters

$$Dev(C) = \sum_{C_k \in C} \sum_{i \in C_k} \delta(i, \mu_k)$$

Where:

- C is the set of all clusters
- μ_k is the center of cluster C_k
- $\delta(i, \mu_k)$ is the chosen distance function.

As an objective, overall deviation should be minimized.

Connectedness of data points

$$Conn(C) = \sum_{i=1}^{N} \left(\sum_{j=1}^{L} x_{i,nm_i(j)} \right),$$

$$\text{where } x_{i,nm_i(j)} = \begin{cases} \frac{1}{j} & if \nexists C_k : i, nn_i(j) \in C_k \\ 0 & otherwise \end{cases}$$

Where:

- $nn_i(j)$ is the jth nearest neighbor of datum i
- L is a parameter determining the number of neighbors that contribute to the connectivity measure

As a objective, connectivity should be minimized.

Connectedness of data points

$$Conn(C) = \sum_{i=1}^{N} \left(\sum_{j=1}^{L} x_{i,nm_i(j)} \right),$$

$$\text{where } x_{i,nm_i(j)} = \begin{cases} \frac{1}{j} & \textit{if} \nexists C_k : i, nn_i(j) \in C_k \\ 0 & \textit{otherwise} \end{cases}$$

Where:

- $nn_i(j)$ is the jth nearest neighbor of datum i
- L is a parameter determining the number of neighbors that contribute to the connectivity measure

As a objective, connectivity should be minimized.

Outline

Incrementing the number of clusters k:

- Improvement in overall deviation δD;
- Degradation in connectivity δC.

Incrementing the number of clusters k:

- Improvement in overall deviation δD;
- Degradation in connectivity δC.

- Number of cluster k smaller than the true number $\rightarrow R = \frac{\delta D}{\delta C}$
  - Large R!
  - The separation of two clusters will trigger a great decrease in overall deviation, with only a small or no increase in connectivity.

- Number of cluster k smaller than the true number $\rightarrow R = \frac{\delta D}{\delta C}$
  - Large R!
  - The separation of two clusters will trigger a great decrease in overall deviation, with only a small or no increase in connectivity.

- Number of cluster k smaller than the true number $\rightarrow R = \frac{\delta D}{\delta C}$
  - Large R!
  - The separation of two clusters will trigger a great decrease in overall deviation, with only a small or no increase in connectivity.

- Number of cluster k larger than the true number:
  - Small R!
  - the decrease in overall deviation will be less significant but come at a high cost in terms of connectivity because a true cluster is being split!

- Number of cluster k larger than the true number:
  - Small R!
  - the decrease in overall deviation will be less significant but come at a high cost in terms of connectivity because a true cluster is being split!

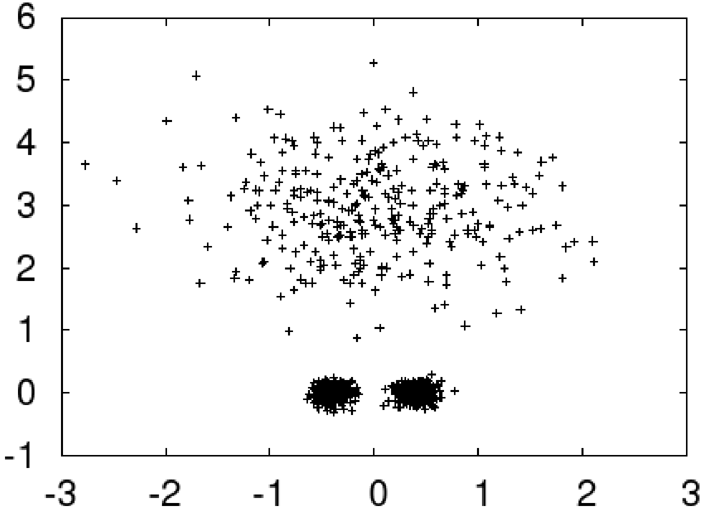
- Number of cluster k larger than the true number:
  - Small R!
  - the decrease in overall deviation will be less significant but come at a high cost in terms of connectivity because a true cluster is being split!

Outline

Parameter settings

Parameter	*setting*
Number of generations	200
External population size	1000
Internal population size	$\max(50, \frac{N}{20})$
Initialization	Minimum spanning tree
Mutation type	L nearest neighbours ($L = 20$)
Mutation rate p_m	$1/N$
Recombination	Uniform crossover
Recombination rate p_r	0.7
Objective functions	Overall deviation and connectivity ($L = 20$)
Constraints	$k \in \{1, \ldots, 25\}$, cluster size > 2
Number of reference distributions	5

Parameter settings



Parameter settings

Problem	single-link	average-link	*k*-means	clustering ensemble	*MOCK*
Long1	0.666444 (0.333556)	0.665104 (0.005896)	0.521989 (0.015211)	**1.0** (0.0)	**1.0** (0.0)
Long2	0.678444 (0.000178)	0.67714 (0.011155)	0.520026 (0.01683)	0.902 (0.0)	**1.0** (0.0)
Long3	0.777895 (0.000556)	0.77514 (0.009774)	0.566661 (0.017321)	0.730591 (0.001802)	**1.0** (0.006)
Sizes1	0.428323 (0.000242)	0.977935 (0.007071)	**0.989003** (0.005013)	0.747616 (0.030904)	0.987 (0.006)
Sizes2	0.522742 (0.000477)	0.981947 (0.009885)	0.987051 (0.004999)	0.633283 (0.002187)	**0.988** (0.006)
Sizes3	0.600841 (0.000782)	0.98502 (0.00905)	0.987114 (0.006899)	0.562078 (0.00984)	**0.99** (0.005)
Sizes4	0.658308 (0.000676)	0.983953 (0.005826)	0.985274 (0.006851)	0.506595 (0.261149)	**0.989** (0.0041)
Sizes5	0.702411 (0.001261)	0.986976 (0.007064)	0.984288 (0.005843)	0.487591 (0.311809)	**0.9909** (0.0079)
Smile1	**1.0** (0.0)	0.753036 (0.0)	0.665609 (0.009407)	**1.0** (0.0)	**1.0** (0.0)
Smile2	**1.0** (0.0)	0.725156 (0.0)	0.586508 (0.009967)	0.91 (0.0)	**1.0** (0.0)
Smile3	**1.0** (0.0)	0.549761 (0.0)	0.505994 (0.007393)	0.776494 (0.001284)	**1.0** (0.0)
Spiral	**1.0** (0.0)	0.576 (0.0)	0.593 (0.002)	1.0 (0.0)	**1.0** (0.0)
Square1	0.399759 (8e-05)	0.977997 (0.015005)	**0.987006** (0.004982)	0.984 (0.008006)	0.985 (0.0051)
Square2	0.399759 (0.0)	0.961982 (0.009888)	**0.976019** (0.007988)	0.97 (0.008002)	0.973 (0.009)
Square3	0.399759 (8e-05)	0.934935 (0.016238)	**0.956933** (0.00802)	0.94599 (0.015982)	0.946 (0.0172)
Square4	0.399759 (8e-05)	0.883035 (0.02214)	**0.919999** (0.008024)	0.908 (0.019006)	0.9041 (0.0184)
Square5	0.399759 (8e-05)	0.720672 (0.107357)	**0.86798** (0.014231)	0.842965 (0.033088)	0.8361 (0.0324)
Triangle1	**1.0** (0.0)	0.997 (0.004001)	0.98486 (0.00613)	0.999 (0.001)	**1.0** (0.0)
Triangle2	0.45193 (0.116834)	0.986979 (0.013638)	0.957697 (0.011837)	0.810492 (0.068513)	**0.995** (0.004)

Outline

Conclusion about MOCK

- Existing algorithms deal only with one clustering objective;
- MOCK is a multi-objective clustering algorithm!

Conclusion about MOCK

- Existing algorithms deal only with one clustering objective;
- MOCK is a multi-objective clustering algorithm!

Outline

How to use Multi-objective clustering with ACO?

- 2 types of ants: each one for a different objective function;
- All ants have a common pheromone matrix;
- Each ant has a probability of choosing an item according to its type and the hormonal densities;
- Number of ants is related to the total number of elements to be clusterized.
- MOCACO - Multiobjective Clustering using Ant Colony Optimization! :-)

How to use Multi-objective clustering with ACO?

- 2 types of ants: each one for a different objective function;
- All ants have a common pheromone matrix;
- Each ant has a probability of choosing an item according to its type and the hormonal densities;
- Number of ants is related to the total number of elements to be clusterized.
- MOCACO - Multiobjective Clustering using Ant Colony Optimization! :-)

How to use Multi-objective clustering with ACO?

- 2 types of ants: each one for a different objective function;
- All ants have a common pheromone matrix;
- Each ant has a probability of choosing an item according to its type and the hormonal densities;
- Number of ants is related to the total number of elements to be clusterized.
- MOCACO - Multiobjective Clustering using Ant Colony Optimization! :-)

How to use Multi-objective clustering with ACO?

- 2 types of ants: each one for a different objective function;
- All ants have a common pheromone matrix;
- Each ant has a probability of choosing an item according to its type and the hormonal densities;
- Number of ants is related to the total number of elements to be clusterized.
- MOCACO - Multiobjective Clustering using Ant Colony Optimization! :-)

How to use Multi-objective clustering with ACO?

- 2 types of ants: each one for a different objective function;
- All ants have a common pheromone matrix;
- Each ant has a probability of choosing an item according to its type and the hormonal densities;
- Number of ants is related to the total number of elements to be clusterized.
- MOCACO - Multiobjective Clustering using Ant Colony Optimization! :-)

References

- Handl, J. and Knowles, J. (2005) Exploiting the trade-off – the benefits of multiple objectives in data clustering *Third International Conference on Evolutionary Multi-Criterion Optimization* 547-560