Multi-objective clustering

Tomas Barton
tomas.barton@fit.cvut.cz
Goal

- goal: a result matching human judgment
- how many objectives do we need to obtain this?

k-means clustering

- most algorithms optimize single objective
- e.g. minimize square distance inside a cluster
Single-Link clustering

- capable of discovering arbitrary shaped clusters
- but too sensitive to noise
Limitation of k-means
Problems

k-means ($k = 3$)
Interesting clustering approaches


2010 – Forestier, Gançarski, Wemmert. "Collaborative clustering with background knowledge." Data & Knowledge Engineering
Cluster Ensembles
Strehl and Gosh (2002)

CSPA – Cluster-based Similarity Partitioning Algorithm: combines multiple clustering based pairwise similarity of instances in clusters

HGPA – Hyper-Graph Partitioning Algorithm: tries partitioning a hypergraph where hyperedges represent clusters

MCLA – Meta-CLustering Algorithm: tries to identify groups of clusters (meta-clusters) and consolidate them
Multiobjective data clustering
Law, Topchy

- uses resampling and combines clusterings
- multi-objective clustering as NP-hard combinatorial optimization problem
MOCK

Multiobjective clustering with automatic k-determination
Possible objectives

A: Compactness
B: Connectedness
C: Spatial separation
MOCK
Connectivity

\[ \text{Conn}(\mathbb{C}) = \sum_{i=1}^{N} \left( \sum_{j=1}^{L} x_{i,nn_{ij}} \right), \quad (1) \]

where

\[ x_{r,s} = \begin{cases} \frac{1}{j}, & \text{if } \#C_k : r \in C_k \land s \in C_k \\ 0, & \text{otherwise}, \end{cases} \]

\( nn_{ij} \) is the \( j \)th nearest neighbour of item \( i \), \( N \) is the size of the data set and \( L \) is a parameter determining the number of neighbours that contribute to the connectivity measure.
MOCK

Deviation (Compactness)

\[
\text{Dev}(\mathbb{C}) = \sum_{C_k \in \mathbb{C}} \sum_{i \in C_k} \delta(i, \mu_k)
\]  

(2)

where \( \mathbb{C} \) is a set of all clusters, \( \mu_k \) is the centroid of the cluster \( C_k \) and \( \delta(., .) \) is a chosen distance function.
MOCK
Solution front

Square 4

k=1

Average link
Single link
k-means
Control

single link: k=25

k-means: k=25

Normalised connectivity
Let’s go back...
Karypis, George, Eui-Hong Han, and Vipin Kumar. "Chameleon: Hierarchical clustering using dynamic modeling." Computer 32.8
k-means
Chameleon algorithm

1. Create the k-nearest neighbor graph
2. Partition the graph
3. Merge partitions

Data set

Diagram:
- Construct a sparse graph
- k-nearest neighbor graph
- Partition the graph
- Merge partitions
- Final clusters
Chameleon

1. k-nn

- dataset represented as a graph
Chameleon

2. partitioning

• split graph into many components
• minimize egdes cut
• optimal bisection is NP-complete, thus we use approximation:
  • Kerighan-Lin $O(n^3)$
  • Spectral bisection $O(n^3)$
  • Fiduccia-Matheyses $O(|E|)$
  • METIS / hMETIS
Complexity of partitioning phase

Time spent on bisection of randomly generated datasets

- hMETIS
- Recursive bisection
$\bar{\phi}(C_i, C_j)$
$\bar{\phi}(C_i), \bar{\phi}(C_j)$
Chameleon

3. merging

- find best candidate for merging

\[
R_{IC}(C_i, C_j) = \frac{\phi(C_i, C_j)}{\phi(C_i) + \phi(C_j)} \cdot \frac{2\phi(C_i, C_j)}{\phi(C_i) + \phi(C_j)}
\]

\[
R_{RC}(C_i, C_j) = \frac{\bar{\phi}(C_i, C_j)}{|C_i| + |C_j|} \cdot \frac{\bar{\phi}(C_i)}{|C_i| + |C_j|} + \frac{|C_j|}{|C_i| + |C_j|}\phi(C_j)
\]

\[
Sim(C_i, C_j) = R_{CL}(C_i, C_j)^{\alpha} \cdot R_{IC}(C_i, C_j)^{\beta}
\]
Chameleon
Combination of objectives

$$Sim(C_i, C_j) = R_{CL}(C_i, C_j)^\alpha \cdot R_{IC}(C_i, C_j)^\beta$$

- $\alpha, \beta$ – user defined priorities
- $\alpha = 2$
- $\beta = 1$
Chameleon MO

- inspired by NSGA-II
- uses same objectives as Chameleon
- relatively “fast”
Chameleon 1

Standard similarity
Chameleon MO
Chameleon MO

- hierarchical merging has complexity $O(n^2)$
- $\Rightarrow$ multi-objective sorting (2 objectives) $O(n^4)$
Chameleon MO

- hierarchical merging has complexity $O(n^2)$
- $\Rightarrow$ multi-objective sorting (2 objectives) $O(n^4)$

approximation required

- limited number of fronts
- blacklist
- 3rd level sorting
- heap rebuild
Chameleon 2
Chameleon 2
improved similarity measure

\[ Sim_{\text{shat}}(C_i, C_j) = R_{\text{CLS}}(C_i, C_j)^\alpha \cdot R_{\text{ICS}}(C_i, C_j)^\beta \cdot \gamma(C_i, C_j) \]

\[ R_{\text{CLS}}(C_i, C_j) = \frac{\bar{s}(C_i, C_j)}{|E_{C_i}| + |E_{C_j}|} \frac{\bar{s}(C_i) + |E_{C_i}|}{|E_{C_i}| + |E_{C_j}|} \bar{s}(C_j) \]

\[ R_{\text{ICS}}(C_i, C_j) = \frac{\min\{\bar{s}(C_i), \bar{s}(C_j)\}}{\max\{\bar{s}(C_i), \bar{s}(C_j)\}} \]

\[ \gamma(C_i, C_j) = \frac{|E_{C_{i,j}}|}{\min(|E_{C_i}|, |E_{C_j}|)} \]

Where \( \bar{s}(C_i) \) is defined as sum of edges’ weights in a cluster.
Chameleon 2
compound dataset
(a) CLUTO, NMI = 0.18  
(b) Ch2, NMI = 0.76
Chameleon 2
cluto-t7.10k
Chameleon 2
cluto-t8.8k
Chameleon 2
CURE
Chameleon 2
aggregation
Chameleon 2
diamond9
Pattern recognition benchmark (30 datasets)

k-means
HAC-CL
HAC-WL
HAC-AL
CURE
CL-G
HAC-SL
DBSCAN
Ch1
Ch2
Clustering evaluation

(unsupervised)
Clustering objectives

objective function

most metrics considers following criteria:

\[ p = \frac{\sum \text{distances in a cluster}}{\sum \text{distances between clusters}} \]
Clustering objectives

C-index

$$f_{c\text{-index}}(C) = \frac{S_w - S_{min}}{S_{max} - S_{min}}$$

where

- $S_w$ is the sum of the within cluster distances
- $S_{min}$ is the sum of the $N_w$ smallest distances between all the pairs of points in the entire dataset. There are $N_t$ such pairs
- $S_{max}$ is the sum of the $N_w$ largest distances between all the pairs of points in the entire dataset
Clustering objectives

Davies-Bouldin

Davies-Bouldin indexes combines two measures, one related to dispersion and the other to the separation between different clusters

\[ f_{DB}(C) = \frac{1}{K} \sum_{i=1}^{K} \max_{i \neq j} \left( \frac{\bar{d}_i + \bar{d}_j}{d(c_i, c_j)} \right) \]

where \( d(c_i, c_j) \) corresponds to the distance between the center of clusters \( C_i \) and \( C_j \), \( \bar{d}_i \) is the average within-group distance for cluster \( C_i \).

\[ \bar{d}_i = \frac{1}{|C_i|} \sum_{l=1}^{|C_i|} d(x_i(l), \bar{x}_i) \]
Clustering objectives

- Dunn index (1974)
- AIC (1974)
- C-index (1976)
- Gamma (1976)
- BIC (1978)
- Davies and Bouldin (1979)
- Silhouette (1987)
- Gap statistic (2001)
- Connectivity (2004)
- Compactness (2006)
Simplified problem

• given a set $\mathcal{C}$ of clustering solution $\{C_1, C_2, \ldots, C_k\}$ created from the same dataset

• we use a supervised function as reference

$$f_{\text{supervised}}(\mathcal{C}) \rightarrow \text{rank}\{r_1, r_2, \ldots, r_k\}$$

• and an unsupervised function

$$g_{\text{unsupervised}}(\mathcal{C}) \rightarrow \text{rank}\{r_1, r_2, \ldots, r_k\}$$
• aggregation dataset – 7 clusters
Visualization of objectives

supervised objective

unsupervised objective

\[ \text{NMI-sqrt} \]

\[ \text{C-index} \]
• over-optimized clustering (highest C-index)
Ideal objective
Dunn
Davies-Bouldin
Point-Bi serial
C-index (Iris dataset)

- correlation \(-0.81\)
AIC (Iris dataset)

- correlation = 0.13
fast non-dominated sort
Pareto front projection
AIC & C-index (Iris dataset)

• correlation $= -0.47$
AIC & Davies-Bouldin (Iris dataset)

- correlation = 0.12
AIC & Point BiSerial (Iris dataset)

- correlation = 0.62
Conclusion

- after 50 years of research we have clustering objective that *actually works*
- there are combinations of objectives that work in many cases
- combining AIC (or BIC) with other objectives improves quality of the result
Questions?

Thank you for your attention

tomas.barton@fit.cvut.cz