Scut: A Balanced Clustering Algorithm

Mikko Malinen

Speech and Image Processing Unit
School of Computing
University of Eastern Finland
2013
A more balanced clustering result is desirable in some applications (e.g. in photo query systems, workload balancing).

We present a clustering method, which weights TSE of a single cluster by number of points in cluster and uses those weighted TSE’s as a cost.

\[ TSE_i = \sum_{j, \text{so that } X_i \text{ belongs to cluster } i} |X_j - C_i|^2 \]

\[ \text{Cost} = n_1 \cdot TSE_1 + n_2 \cdot TSE_2 + \ldots + n_k \cdot TSE_k, \]

where \( n_i \) is number of points in \( i \text{th cluster.} \)
The cost function

\[ \text{Cost} = n_1 \cdot TSE_1 + n_2 \cdot TSE_2 + \ldots + n_k \cdot TSE_k, \]

where \( n_i \) is number of points in \( i \)th cluster.

may also be written as

\[ \text{Cost} = n_1^2 \cdot MSE_1 + n_2^2 \cdot MSE_2 + \ldots + n_k^2 \cdot MSE_k, \]

where \( n_i \) is number of points in \( i \)th cluster.

\[
(MSE_i = \sum_{j, \text{so that } X_j \text{ belongs to cluster } i} \frac{|X_j - C_i|^2}{n_i})
\]
MAX k-CUT

Partition $P_1$

Partition $P_2$

Partition $P_3$

Partition $P_4$

$\omega_{13} = 4$

cut = 27

MAX 4-CUT
Scut algorithm

Input: dataset X, number of clusters k
Output: partitioning of points P

Set the vertices of the graph the data points.
Set the weights of the edges of the graph the squared distances.
Compute MAX k-CUT
Output partitioning of points P
Balanced k-means clustering

Mikko Malinen

Speech and Image Processing Unit
School of Computing
University of Eastern Finland
Balanced clustering

Want to fix the number of points in clusters?

We have a solution for that.
Forming the assignment graph

Data points $X_i$
K-means –like process

1) **Initialization** of centroid locations

2) **Assignment phase.** Calculate arclengths and use Hungarian algorithm to assign.

3) **Update phase.** Calculate new centroid locations.

**Repeat** phases 2)-3) until centroid locations do not change.
Time complexity

Balanced k-means:

\[ T(n) = O(I \cdot n^3) \]

Constrained k-means (from literature):

\[ T(n) = O(I \cdot k^{3.5} n^{3.5}) \]

Where \( I \) = number of kmeans iterations
Experiment with dataset s2 of size n=5000

Execution time 1h 30min
Experiment with Iris dataset of 150 points and 4 dimensions

Execution time 0.4 s
Rottavalikoitu data, normalized, dimensions 2 and 3 out of 14

Execution time 2.6 seconds