

# Constrained clustering with $k$ -means

Ziqiu Su\*

Department of Mathematics and Statistics  
University of Maryland Baltimore County

Text Mining Workshop 2009  
Sparks, NV

May 2, 2009

---

\*Joint work with Jacob Kogan, Charles Nicholas

# Overview

- $k$ -means with no links
- Clustering with cannot-link constraints
- Elimination of must-link constraints
- Numerical Experiments
- Conclusion and future research directions

# Problem Setting

- A partition  $\Pi = \{\pi_1, \dots, \pi_k\}$  of  $\mathcal{A} = \{\mathbf{a}_1, \dots, \mathbf{a}_m\} \subset \mathbf{R}^n$  is where

$$\bigcup_{i=1}^k \pi_i = \mathcal{A}, \pi_i \cap \pi_j = \emptyset, i \neq j$$

- For a finite vector set  $\pi \subset \mathbf{R}^n$  and a distance-like function  $d$ , we define the centroid  $\mathbf{c}(\pi)$  as

$$\mathbf{c}(\pi) = \arg \min \left\{ \sum_{\mathbf{a} \in \pi} d(\mathbf{x}, \mathbf{a}), \mathbf{x} \in \mathbf{R}^n \right\}$$

# Partitions Approach

- The quality of a cluster  $\pi$ :  $Q(\pi) = \sum_{\mathbf{a} \in \pi} d(\mathbf{c}(\pi), \mathbf{a})$
- The quality of a partition  $\Pi$ :  $Q(\Pi) = \sum_{i=1}^k Q(\pi_i)$
- We are going to find a  $k$  cluster partition  $\Pi^0$  so that  $Q(\Pi^0) \leq Q(\Pi)$  for each  $k$  cluster partition  $\Pi$
- In the following,  $d = \|\cdot\|^2$

# Batch $k$ -means Algorithm

J. MacQueen, 1967.

- 1 Choose the number of clusters,  $k$ .
- 2 Randomly generate  $k$  cluster centroids.
- 3 Assign each point to the closest cluster centroid.
- 4 Recompute the new cluster centroids.
- 5 Repeat the two previous steps until the partition doesn't change.

# Incremental Step

Suppose  $\mathbf{a} \in \pi_i$ .

- $\Delta_k = \|\mathbf{c}_i - \mathbf{a}\|^2 - \|\mathbf{c}_j - \mathbf{a}\|^2$ .
- If  $\Delta_k > 0$ , do the batch step.

## First Variation

*A first variation of a partition  $\Pi$  is a partition  $\Pi'$  obtained from  $\Pi$  by removing a single vector  $\mathbf{a}$  from a cluster  $\pi_i$  of  $\Pi$  and assigning this vector to an existing cluster  $\pi_j$  of  $\Pi$ .*

- $\Delta = \frac{|\pi_i|}{|\pi_i|-1} \|\mathbf{c}_i - \mathbf{a}\|^2 - \frac{|\pi_j|}{|\pi_j|+1} \|\mathbf{c}_j - \mathbf{a}\|^2 > \Delta_k$
- If  $\Delta > 0$ , do the incremental step.

## nextFV

*The partition  $\text{nextFV}(\Pi)$  is a first variation of  $\Pi$  so that for each first variation  $\Pi'$  one has  $Q(\text{nextFV}(\Pi)) \leq Q(\Pi')$ .*

# Algorithm Outline

**Cannot-Link** Two data points must not be in the same cluster.

- $p : \mathbf{R}^n \times \mathbf{R}^n \rightarrow \mathbf{R}_+$  is a symmetric penalty function:  
 $p(\mathbf{a}, \mathbf{a}) = 0, p(\mathbf{a}, \mathbf{a}') = p(\mathbf{a}', \mathbf{a}) \geq 0$

## The batch $k$ -means algorithm

- Given a partition  $\Pi = \{\pi_1, \dots, \pi_k\}$  of the set  $\mathcal{A}$  compute the corresponding centroids  $\{\mathbf{c}_1, \dots, \mathbf{c}_k\}$  by

$$\mathbf{c}_i = \arg \min_{\mathbf{x}} \left\{ \sum_{\mathbf{a} \in \pi_i} \|\mathbf{x} - \mathbf{a}\|^2 + \frac{1}{2} \sum_{\mathbf{a}, \mathbf{a}' \in \pi_i} p(\mathbf{a}, \mathbf{a}') \right\}$$

- Compute the new partition  $\Pi' = \{\pi'_1, \dots, \pi'_k\}$  where  $\pi'_i =$

$$\left\{ \mathbf{a} : \|\mathbf{c}_i - \mathbf{a}\|^2 + \sum_{\mathbf{a}' \in \pi_i} p(\mathbf{a}, \mathbf{a}') \leq \|\mathbf{c}_j - \mathbf{a}\|^2 + \sum_{\mathbf{a}' \in \pi_j} p(\mathbf{a}, \mathbf{a}'), \forall j \right\}$$

# New Quality Function

For a cluster  $\pi \subseteq \mathcal{A}$

$$Q(\pi) = \sum_{\mathbf{a} \in \pi} \|\mathbf{c}(\pi) - \mathbf{a}\|^2 + \frac{1}{2} \sum_{\mathbf{a}, \mathbf{a}' \in \pi} p(\mathbf{a}, \mathbf{a}')$$

We want find a  $k$ -cluster partition of  $\mathcal{A}$

$$\Pi^0 = \{\pi_1^0, \dots, \pi_k^0\}$$

that minimizes  $Q(\pi_1) + \dots + Q(\pi_k)$ .

# New Quality Function

For a cluster  $\pi \subseteq \mathcal{A}$

$$Q(\pi) = \sum_{\mathbf{a} \in \pi} \|\mathbf{c}(\pi) - \mathbf{a}\|^2 + \frac{1}{2} \sum_{\mathbf{a}, \mathbf{a}' \in \pi} p(\mathbf{a}, \mathbf{a}')$$

We want find a  $k$ -cluster partition of  $\mathcal{A}$

$$\Pi^0 = \{\pi_1^0, \dots, \pi_k^0\}$$

that minimizes  $Q(\pi_1) + \dots + Q(\pi_k)$ .

The previous algorithm may lead to erroneous result.

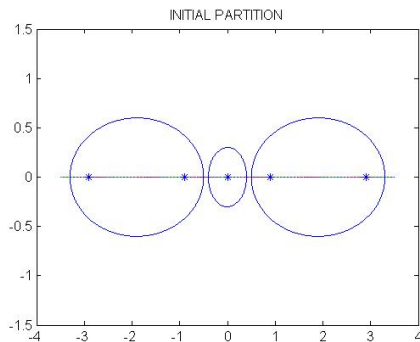
# Example

Consider  $\mathcal{A} = \{\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3, \mathbf{a}_4, \mathbf{a}_5\} = \{-2.9, -0.9, 0, 0.9, 2.9\}$  with  $\rho(\mathbf{a}_i, \mathbf{a}_j) = 4$  when  $i \neq j$ . Initially we have

$\Pi = \{\{\mathbf{a}_1, \mathbf{a}_2\}, \{\mathbf{a}_3\}, \{\mathbf{a}_4, \mathbf{a}_5\}\}$ .

$Q(\Pi) = (2 + p) + 0 + (2 + p) = 4 + 2p = 12$

Figure: initial three cluster partition



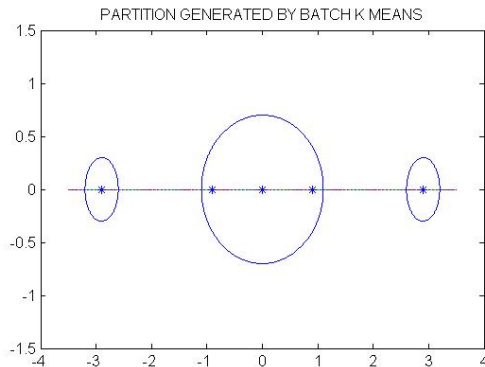
# Example

If we apply the previous assignment step, it leads to

$\Pi' = \{\{\mathbf{a}_1\}, \{\mathbf{a}_2, \mathbf{a}_3, \mathbf{a}_4\}, \{\mathbf{a}_5\}\}$  and

$Q(\Pi') = 0 + (3\rho + 2(0.9)^2) + 0 = 1.62 + 3\rho = 13.62$ .

**Figure:** three cluster partition generated by batch  $k$ -means



# Incremental Step with cannot-link constraints

Suppose  $\mathbf{a} \in \pi_i$ ,

- $$\Delta = \frac{|\pi_i|}{|\pi_i| - 1} \|\mathbf{c}_i - \mathbf{a}\|^2 - \frac{|\pi_j|}{|\pi_j| + 1} \|\mathbf{c}_j - \mathbf{a}\|^2 + \sum_{\mathbf{a}' \in \pi_i} p(\mathbf{a}, \mathbf{a}') -$$

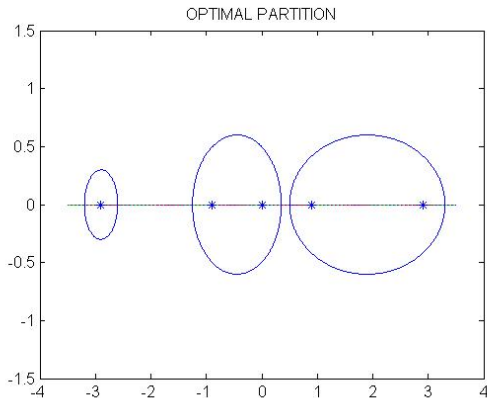
$$\sum_{\mathbf{a}' \in \pi_j} p(\mathbf{a}, \mathbf{a}')$$

- If  $\Delta > 0$ , do the incremental step.

# Example

If we apply the incremental step, it leads to  $\Pi'' = \{\{\mathbf{a}_1\}, \{\mathbf{a}_2, \mathbf{a}_3\}, \{\mathbf{a}_4, \mathbf{a}_5\}\}$  and  $Q(\Pi'') = 10.405$ .

**Figure:** optimal three cluster partition generated by incremental  $k$ -means



# The Incremental $k$ -means Algorithm

For a user supplied non-negative tolerance  $\text{tol} \geq 0$  do the following:

- 1 Start with an initial partition  $\Pi^{(0)} = \{\pi_1^{(0)}, \dots, \pi_k^{(0)}\}$ . Set the index of iteration  $t = 0$ .
- 2 Generate the partition  $\text{nextFV}(\Pi^{(t)})$ .  
if  $[Q(\Pi^{(t)}) - Q(\text{nextFV}(\Pi^{(t)}))] > \text{tol}$   
set  $\Pi^{(t+1)} = \text{nextFV}(\Pi^{(t)})$ .  
increment  $t$  by 1.  
go to 2
- 3 Stop.

# Transitive Closure of must-link constraints

**Must-Linked** Two data points have to be in the same cluster.

- $\pi_i = \{\mathbf{a}_{i_1}, \dots, \mathbf{a}_{i_p}\}$  are must-linked
- $\mathbf{b}_i = \mathbf{c}(\pi_i) = \frac{1}{|\pi_i|} \sum_{\mathbf{a} \in \pi_i} \mathbf{a}$
- $q_i = Q(\pi_i)$
- $m(\mathbf{b}_i) = m_i = |\pi_i|$  to represent each closure
- We want to partition  $\mathcal{B} = \{\mathbf{b}_1, \dots, \mathbf{b}_M\}$  and recover the induced partition of  $\mathcal{A}$

# Clustering with the summaries

- $\pi^{\mathcal{B}} = \{\mathbf{b}_{i_1}, \dots, \mathbf{b}_{i_p}\} \subseteq \mathcal{B}$
- $P(\mathbf{b}_i, \mathbf{b}_j) = \frac{1}{2} \sum_{\mathbf{a} \in \pi_i, \mathbf{a}' \in \pi_j} p(\mathbf{a}, \mathbf{a}')$
- $Q_{\mathcal{B}}(\pi^{\mathcal{B}}) = \sum_{j=1}^p m_{i_j} \|\mathbf{c} - \mathbf{b}_{i_j}\|^2 + \frac{1}{2} \sum_{l,j} P(\mathbf{b}_{i_l}, \mathbf{b}_{i_j})$
- $\mathbf{c} = \frac{m_{i_1} \mathbf{b}_{i_1} + \dots + m_{i_p} \mathbf{b}_{i_p}}{m_{i_1} + \dots + m_{i_p}}$
- $Q(\bigcup_{j=1}^p \pi_{i_j}) = \sum_{j=1}^p q_{i_j} + Q_{\mathcal{B}}(\pi^{\mathcal{B}})$

# Incremental Step

- $M_I = \sum_{\mathbf{b} \in \pi_I^{\mathcal{B}}} m(\mathbf{b})$
- $\Delta = \frac{M_i \cdot m(\mathbf{b})}{M_i + m(\mathbf{b})} \|\mathbf{c}(\pi_i^{\mathcal{B}}) - \mathbf{b}\|^2 - \frac{M_j \cdot m(\mathbf{b})}{M_j + m(\mathbf{b})} \|\mathbf{c}(\pi_j^{\mathcal{B}}) - \mathbf{b}\|^2 + \sum_{\mathbf{b}' \in \pi_i^{\mathcal{B}}} P(\mathbf{b}, \mathbf{b}') - \sum_{\mathbf{b}' \in \pi_j^{\mathcal{B}}} P(\mathbf{b}, \mathbf{b}')$
- If  $\Delta > 0$ , do the incremental step.

# Data Set

Three document collection dataset classic3:

- DC0(Medlars Collection 1033 medical abstracts)
- DC1(CISI Collection 1460 information science abstracts)
- DC2(Cranfield Collection 1398 aerodynamics abstracts)

600 best terms are selected, 3891 vectors in  $\mathbf{R}^{600}$

# Numerical Experiments I

Cluster/DocCol	DC0	DC1	DC2
cluster 0	1362	13	6
cluster 1	7	1372	120
cluster 2	91	13	907

**Table:** PDDP generated “confusion” matrix with **250** “misclassified” documents

# Numerical Experiments II

Cluster/DocCol	DC0	DC1	DC2
cluster 0	1437	22	9
cluster 1	1	1360	5
cluster 2	22	16	1019

**Table:** PDDP followed by incremental  $k$ -means with no penalty generated “confusion” matrix with **75** “misclassified” documents

# Numerical Experiments III

Cluster/DocCol	DC0	DC1	DC2
cluster 0	1453	17	8
cluster 1	1	1377	4
cluster 2	6	4	1021

**Table:** PDDP followed by incremental  $k$ -means with  $p = 0.01$  generated “confusion” matrix with **40** “misclassified” documents

# Numerical Experiments IV

Cluster/DocCol	DC0	DC1	DC2
cluster 0	1460	0	0
cluster 1	0	1398	0
cluster 2	0	0	1033

**Table:** PDDP followed by incremental  $k$ -means with  $p = 0.09$  generated “confusion” matrix with **0** “misclassified” documents

# Numerical Experiments V

penalty	misclassification
0.00	75
0.01	40
0.02	20
0.03	17
0.04	8
0.05	5
0.06	4
0.07	2
0.08	1
0.09	0

**Table:** penalty vs. “misclassification” with  $r_0 = r_1 = r_2 = 1$

# Summary

- Constrained clustering algorithm
- Replace cannot-link constraints with a penalty function
- Eliminate must-link constraints by BIRCH-like technique
- Improved misclassification rate

# Future Work

- Substitute must-link constraints by *negative* penalties
- Replace constraints with a penalty function on centroids
- Constrained clustering with Bregman divergences
- Penalty on the number of clusters