Clustering

$k$-means like algorithms and beyond

CIKM 2003

Jacob Kogan
Charles Nicholas
Marc Teboulle
Outline of the talk

- how to build a partition
- how to improve a partition
- how to evaluate a partition
Partitions

$X = \{x_1, \ldots, x_m\}$ is a set of vectors in $\mathbb{R}^n$.

A partition $\Pi$ of $X$ is

$$\Pi = \{\pi_1, \ldots, \pi_k\}$$

$$\pi_1 \cup \ldots \cup \pi_k = \{x_1, \ldots, x_m\}, \text{ and } \pi_i \cap \pi_j = \emptyset \text{ if } i \neq j.$$

$q$ is a real valued function whose domain is the set of subsets of $\{x_1, \ldots, x_m\}$.

The quality of the partition is given by

$$Q(\Pi) = q(\pi_1) + \ldots + q(\pi_k).$$
What do we want?

To identify an optimal partition

\[ \Pi^o = \{\pi_1^o, \ldots, \pi_k^o\} , \]

i.e., one that optimizes

\[ Q(\Pi) = q(\pi_1) + \ldots + q(\pi_k) . \]

In general the solution is available when the dimension of the vector space is \textbf{ONE}.
Data sets

Vector sets generated for large document collections contain vectors which are:

- sparse
- high dimensional
- have non-negative entries
- normalized (usually with $l_2$ norm)

For example the Reuters business news collection (available from David D. Lewis’ home page: http://www.research.att.com/~lewis) contains 19043 non-empty documents with 44749 unique words.
The simplest way to go

Given a partition \( \Pi^{(t)} = \left\{ \pi_1^{(t)}, \ldots, \pi_k^{(t)} \right\} \),

build a partition \( \Pi^{(t+1)} = \left\{ \pi_1^{(t+1)}, \ldots, \pi_k^{(t+1)} \right\} \),

such that:

- there are clusters \( \pi_i^{(t)}, \pi_j^{(t)} \), and \( x \in \pi_i^{(t)} \),

- \( \pi_i^{(t+1)} = \pi_i^{(t)} - \{x\} \), \( \pi_j^{(t+1)} = \pi_j^{(t)} \cup \{x\} \)

and

\[ q \left( \pi_i^{(t+1)} \right) + q \left( \pi_j^{(t+1)} \right) < q \left( \pi_i^{(t)} \right) + q \left( \pi_j^{(t)} \right). \]
\[ \Pi(t) = \left\{ \pi_1(t), \pi_2(t) \right\} \]
Pick a vector
New partition

\[ \Pi^{(t+1)} \]
Centroid
Centroid
Centroid

\[ f(y) = d(y, x_1) + d(y, x_2) + d(y, x_3) \]
Distance-like function

d(y, x) and q can be associated.

The relation between q and d can be defined through a centroid c of a cluster π

\[ c = c(\pi) = \arg\min \left\{ \sum_{x \in \pi} d(y, x), \ y \in C \right\}. \]

If q(π) is defined as \( \sum_{x \in \pi} d(c(\pi), x) \), then centroids and partitions can be associated.
Centroid-partition association

1. For a set of centroids \( \{c_1, \ldots, c_k\} \) define a partition \( \{\pi_1, \ldots, \pi_k\} \) of the set \( X \) by:

\[
\pi_i = \{x \mid d(c_i, x) \leq d(c_j, x) \text{ for each } j \neq i\}
\]

(we break ties arbitrarily).

2. Given a partition \( \{\pi_1, \ldots, \pi_k\} \) of the set \( X \) define the corresponding centroids \( \{c_1, \ldots, c_k\} \) by

\[
c_i = \text{arg min} \left\{ \sum_{x \in \pi_i} d(y, x), \; y \in C \right\}.
\]
Example

"distance-like" function

\[ d(y, x) = \| x - y \|^2 \] and \( C = \mathbb{R}^n \)

if \( \pi = \{x_1, \ldots, x_l\} \), then

\[
\begin{align*}
c &= \text{arg min} \left\{ \sum_{x \in \pi} d(y, x), \ y \in C \right\} = \frac{1}{l} \sum_{i=1}^{l} x_i
\end{align*}
\]

and

\[
q(\pi) = \sum_{x \in \pi} \| x - c \|^2
\]
Example
Example
Example

REASSIGN VECTORS
convexity of final partition
optimal 1D two cluster partition
Batch k-means like algorithms

1. For a set of centroids \( \{c_1, \ldots, c_k\} \) define a partition \( \{\pi_1, \ldots, \pi_k\} \) of the set \( X \) by:

\[
\pi_i = \left\{ x \mid \|c_i - x\|^2 \leq \|c_j - x\|^2 \text{ for each } j \neq i \right\}
\]

(we break ties arbitrarily).

2. Given a partition \( \{\pi_1, \ldots, \pi_k\} \) of the set \( X \) define the corresponding centroids \( \{c_1, \ldots, c_k\} \) by

\[
c_i = \arg\min_{\pi_i} \left\{ \sum_{x \in \pi_i} \|y - x\|^2, \ y \in \mathbb{R}^n \right\}.
\]
Deficiencies

- the “right” number of clusters should be supplied,
- the initial partition

\[ \Pi^{(0)} = \left\{ \pi_1^{(0)}, \ldots, \pi_k^{(0)} \right\} \]

should be supplied,

- the batch \( k \)-means often gets trapped at a local minimum.
Incremental $k$–means

$X = \{0, \frac{2}{3}, 1\}$, $\pi_1^{(0)} = \{0, \frac{2}{3}\}$, $\pi_2^{(0)} = \{1\}$. 
Enhanced k-means algorithm

1. Set $t = 0$.

2. Start with an arbitrary partitioning $\Pi^{(t)} = \left\{ \pi_1^{(t)}, \ldots, \pi_k^{(t)} \right\}$.

3. Run batch k-means until no vector movement is detected.

4. Run one iteration of incremental k-means.
   if (vector movement is detected) go to Step 3.

5. Stop.
The decision whether a vector $x \in \pi_i$ should be moved from cluster $\pi_i$ with $m_i$ vectors to cluster $\pi_j$ with $m_j$ vectors made by the batch $k$–means algorithm based on the sign of

$$\Delta = -\|x - c(\pi_i)\|^2 + \|x - c(\pi_j)\|^2.$$

The vector $x$ is moved by the batch $k$–means algorithm if $\Delta < 0$.

The exact change in the value of the objective function caused by the move is

$$\Delta_{\text{exact}} = -\frac{m_i}{m_i - 1} \|x - c(\pi_i)\|^2 + \frac{m_j}{m_j + 1} \|x - c(\pi_j)\|^2.$$
"distance–like" functions

A vector $\mathbf{x} = (x[1], \ldots, x[n])^T \in \mathbb{R}^n$.

- $d(\mathbf{c}, \mathbf{x}) = \|\mathbf{c} - \mathbf{x}\|^2$
- $d(\mathbf{c}, \mathbf{x}) = \mathbf{c}^T \mathbf{x}$
- $d(\mathbf{c}, \mathbf{x}) = \sum_{j=1}^{n} \left[ x[j] \log \frac{x[j]}{c[j]} + c[j] - x[j] \right]$
- $d(\mathbf{c}, \mathbf{x}) = \sum_{j=1}^{n} \left[ c[j] \log \frac{c[j]}{x[j]} + x[j] - c[j] \right]$
\[ \Pi = \{\pi_1, \ldots, \pi_k\}, \quad q(\pi) = \left\| \sum_{x \in \pi} x \right\| \]
optimal 1D two cluster partition

when the data belongs to $S^1$
the optimal two cluster partition can be obtained by splitting the circle $S^1$ into two semi-circles by a line passing through the origin
data

- DC0 (Medlars Collection, 1033 medical abstracts).
- DC1 (CISI Collection, 1460 information science abstracts).
- DC2 (Cranfield Collection, 1400 aerodynamics abstracts).

<table>
<thead>
<tr>
<th></th>
<th>DC0</th>
<th>DC1</th>
<th>DC2</th>
</tr>
</thead>
<tbody>
<tr>
<td>cluster 0</td>
<td>1004</td>
<td>5</td>
<td>4</td>
</tr>
<tr>
<td>cluster 1</td>
<td>18</td>
<td>1440</td>
<td>16</td>
</tr>
<tr>
<td>cluster 2</td>
<td>11</td>
<td>15</td>
<td>1380</td>
</tr>
</tbody>
</table>

69 "misclassified" documents using 4099 terms
The Pythagorean Theorem employed 24 words, the Lord’s Prayer has 66 words, Archimedes Principle has 67 words, the 10 Commandments have 179 words, the Gettysburg Address had 286 words, the Declaration of Independence has 1,300 words and finally the European Commission’s regulation on the sale of cabbage: 26,911 words.
what do we want to do:

- to use the same data
- to select SMALLER set of terms (and to reduce the dimensionality of the problem)
- to apply a hybrid clustering scheme (a sequence of clustering algorithms so that the output of algorithm \( i \) becomes the input of algorithm \( i + 1 \)).

and to get better clustering results
PDDP

Principal Direction Divisive Partitioning

or

HOW TO GET A REASONABLE INITIAL PARTITION
PDDP

Principal Direction Divisive Partitioning

- means and beyond – p.35/53
projection
partition one dimensional set
partition original set
How to find the principal direction

- Look at the "term by document" matrix

\[ X = [x_1, \ldots, x_m]. \]

- compute the mean \( m = \frac{x_1 + \ldots + x_m}{m} \)

- compute the first singular vector \( v_1 \) of the matrix

\[ X - m e^T = [x_1 - m, \ldots, x_m - m]. \]

\( v_1 \) is the principal direction vector
## clustering results

<table>
<thead>
<tr>
<th></th>
<th>DC0</th>
<th>DC1</th>
<th>DC2</th>
</tr>
</thead>
<tbody>
<tr>
<td>cluster 0</td>
<td>272</td>
<td>9</td>
<td>1379</td>
</tr>
<tr>
<td>cluster 1</td>
<td>4</td>
<td>1285</td>
<td>11</td>
</tr>
<tr>
<td>cluster 2</td>
<td>757</td>
<td>166</td>
<td>8</td>
</tr>
<tr>
<td>&quot;empty&quot; documents</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>cluster 3</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

PDDP generated initial "confusion" matrix with 470 "misclassified" documents using 600 terms
constrained data

PDDP as well as the classical $k$–means algorithm are general clustering algorithms capable of handling general datasets in $\mathbb{R}^n$.

"document–vectors" reside on an $n - 1$ dimensional sphere $S^{n-1}$. 
sPDDP

1. Given a set \( X \subset S^{n-1}_2 \) determine the two dimensional plane \( P \) that approximates \( X \) in the “best possible way”.
2. Project \( X \) onto \( P \) and denote the projection by \( Y \).
3. “Push” \( Y \) to the great circle, i.e., \( y \rightarrow z = \frac{y}{\|y\|} \).
4. Partition \( Z \subset S^1_2 \) into two clusters \( Z_1 \) and \( Z_2 \).
5. Generate the induced partition \( \{X_1, X_2\} \) of \( X \) as follows:
   \[
   X_1 = \{x \mid z \in Z_1\}, \text{ and } X_2 = \{x \mid z \in Z_2\}.
   \]
first two dimensional projection
circular partition
second plane projection
### clustering results

<table>
<thead>
<tr>
<th>Cluster</th>
<th>DC0</th>
<th>DC1</th>
<th>DC2</th>
</tr>
</thead>
<tbody>
<tr>
<td>cluster 0</td>
<td>1000</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>cluster 1</td>
<td>8</td>
<td>10</td>
<td>1376</td>
</tr>
<tr>
<td>cluster 2</td>
<td>25</td>
<td>1447</td>
<td>21</td>
</tr>
<tr>
<td>&quot;empty&quot; docs</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>cluster 3</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

sPDDP generated initial "confusion" matrix with 68 "misclassified" documents using 600 terms
### Clustering Results

<table>
<thead>
<tr>
<th># of terms</th>
<th>0 vec</th>
<th>alone</th>
<th>$k$-means</th>
<th>sph $k$-means</th>
<th>IT-$k$-means</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>12</td>
<td>383</td>
<td>258</td>
<td>229</td>
<td>168</td>
</tr>
<tr>
<td>200</td>
<td>3</td>
<td>277</td>
<td>133</td>
<td>143</td>
<td>116</td>
</tr>
<tr>
<td>300</td>
<td>0</td>
<td>228</td>
<td>100</td>
<td>104</td>
<td>81</td>
</tr>
<tr>
<td>400</td>
<td>0</td>
<td>88</td>
<td>80</td>
<td>78</td>
<td>56</td>
</tr>
<tr>
<td>500</td>
<td>0</td>
<td>76</td>
<td>62</td>
<td>57</td>
<td>40</td>
</tr>
<tr>
<td>600</td>
<td>0</td>
<td>68</td>
<td>62</td>
<td>54</td>
<td>44</td>
</tr>
</tbody>
</table>
Low Bound for $Q(\Pi)$

$$\Pi = \{\pi_1, \ldots, \pi_k\}, \quad |\pi_i| = m_i$$

$$Y = \begin{bmatrix}
\frac{e_{m_1}}{\sqrt{m_1}} & \cdots & \cdots & \\
\cdots & \frac{e_{m_2}}{\sqrt{m_2}} & \cdots & \\
\cdots & \cdots & \cdots & \\
\cdots & \cdots & \cdots & \frac{e_{m_k}}{\sqrt{m_k}}
\end{bmatrix} \quad (\ast)$$

$$Y^TY = I_k$$

$$Q(\Pi) = \text{trace} \left( X^T X \right) - \text{trace} \left( Y^T X^T X Y \right).$$

To minimize $Q(\Pi)$ solve

$$\max \left\{ \text{trace} \left( Y^T X^T X Y \right) : Y \text{ is of the form } (\ast) \right\}.$$
Relaxed Maximization Problem

\[ \max \left\{ \text{trace} \left( Y^T X^T XY \right) : Y^T Y = I_k \right\}. \]

**Theorem** (Ky Fan) If \( H \) is a symmetric matrix with eigenvalues

\[ \lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n, \]

then

\[ \max_{Y^T Y = I_k} \text{trace} \left( Y^T H Y \right) = \lambda_1 + \lambda_2 + \ldots + \lambda_k. \]
Low Bound for $Q(\Pi)$

As a by-product we have

$$Q(\Pi) \geq \text{trace} \left( X^T X \right) - \max_{Y^T Y = I_k} \text{trace} \left( Y^T X^T X Y \right)$$

$$= \min\{m,n\}$$

$$\sum_{i=k+1}^{\min\{m,n\}} \sigma_i^2 (X).$$