SSC : Statistical Subspace Clustering

L. Candillier$^{1,2}$, I. Tellier$^1$, F. Torre$^1$, O. Bousquet$^2$

$^1$GRAppA, Lille 3 University / $^2$Pertinence, Paris
MLDM, 9 july 2005

1. Subspace clustering
   - Overview
   - Limits

2. SSC algorithm
   - Motivations
   - Clusters detection
   - Output presentation

3. Experiments and results
   - Artificial data
   - Real data

4. Conclusions and future work
I. Subspace clustering [Parsons et al., 2004]

Different clusters may exist in different subspaces

\[ X \times Z \]

\[ X \times Z \]

\[ Y \times Z \]

\[ X \times Y \]
Stakes of subspace clustering

- capture the subspaces specific to each cluster
- generalize clustering and feature selection
- tackle the problem of *curse of dimensionality*
- get reduced description of the clusters
Limits of existing methods [Agrawal et al., 1998]

- consider only numerical data
- need user parameters difficult to set
- no interpretable output

Fig.: Results of the 2 principal types of subspace clustering methods on an artificial example.
EM methods [Dempster et al., 1977]

Fig.: Result of EM-like methods on the artificial example.

- may be slow to converge
- need a model able to mix continuous and discrete dimensions
- no interpretable output
II. SSC algorithm

**Goals**: handle discrete features, minimize the number of user parameters, interpretable output

**Method**: probabilistic model & EM algorithm

**Assumption**: data generated by a mix of probability distributions independent on each dimension $\theta = (\theta_1, ..., \theta_K)$

$$\theta_k = \left( \pi_k, \{(\mu_{kd}, \sigma_{kd})\}_d \text{ continuous}, \{\text{Freqs}_{kd}\}_d \text{ discrete} \right)$$

- Allow to capture irrelevant dimensions
- Allow an interpretable representation as a set of rules (hypercubes) defined with few dimensions
- Speed up the algorithm
Given $K$ the number of expected clusters

1. **Clusters detection**
   - Iterate $R$ times
     - Initialize the model (randomly)
     - Optimize the model parameters ($EM$)
     - Compute the log-likelihood $LL$
   - Select the model that maximizes $LL$

2. **Output presentation**
   - Create the rules associated with the clusters
   - Simplify the rules
   - Select the more visual couples of dimensions
EM algorithm

Find the model parameters that best fit the data
⇒ optimize the \( \text{log-likelihood} \ LL \) of the model to the data

Iterate 2 steps:

1. **Expectation**: find the membership probabilities of the data to the clusters according to the current model parameters

2. **Maximization**: update the model parameters according to the new membership probabilities

Stop when \( LL^{t+1} - LL^t < \delta \)
Specificities of SSC

- Membership probability of a data point to a cluster
  \[= \text{product of membership probabilities on each dimension}\]

- Acceleration heuristic
  add a \textit{k-means like} stopping criterion: stop whenever the membership of each data point to their most probable cluster does not change
  \[\Rightarrow S_k = \text{set of data points whose membership probability to cluster } C_k \text{ is the highest}\]
As a set of rules (hypercubes)

- smallest interval on continuous dimensions
- most probable category on discrete dimensions
Output presentation

As a set of rules (hypercubes)

- smallest interval on continuous dimensions
- most probable category on discrete dimensions

+ select as few dimensions as possible
Compare the likelihood of our model on each independent dimension of each cluster with the likelihood of a uniform model

$$\sum_{\bar{x}_i \in S_k} \log P(X_{id} | \theta_{kd}) > \sum_{\bar{x}_i \in S_k} \log P(X_{id} | \theta_U)$$
Dimension pruning

1. \( W_{kd} = \) weight of dimension \( d \) for cluster \( C_k \)
   - ratio between local and global standard deviation for continuous dimensions
   - relative frequency of the most probable category for discrete dimensions

2. Delete, in ascending order of their weights, the dimensions from the rule if their deletion does not modify its support

3. Visual weights of the couples of relevant dimensions

\[
V_{ij} = \sum_{k=1}^{K} \max(W_{ki}, W_{kj})
\]
III. Experiments and results

Numerical artificial databases

- Quality measure: purity of clusters (initial concept)
- Effectiveness of the acceleration heuristic
- Noise resistance (random data points)
- Influence of parameter $K$ (number of expected clusters)
- Comparison with LAC [Domeniconi et al., 2004]
Effectiveness of the acceleration heuristic

![Graph showing the effectiveness of the acceleration heuristic for different data sizes.](image-url)

(Execution time for $N$ between 200 and 2000)
Artificial data

Resistance to noise varying between 0 and 20%

(600 data points, 30 dimensions, 5 clusters, mean of 3 relevant dimensions per cluster, standard deviation between 2 and 5)
Influence of parameter $K$ (number of expected clusters)

$$BIC = -2 \times LL(\theta|D) + m_M \log N$$

($m_M$ number of free parameters of the model)
Real data

*Automobile* database from UCI repository [Blake and Merz, 1998]

- Descriptions of 205 cars
- 26 features: 16 continuous and 10 discrete
- $K = 3$

Result of SSC: 3 ranges of cars with a mean of 4 dimensions

- $C_0$: num-of-doors = four; wheel-base $\in [94.5, 103.3]$; length $\in [165.3, 177.8]$; curb-weight $\in [2017, 3110]$; price $\in [6989, 13950]$;
- $C_1$: drive-wheels = fwd; width $\in [60.3, 64.4]$; compression-ratio $\in [7.6, 10.1]$; price $\in [5118, 9980]$;
- $C_2$: drive-wheels = rwd; length $\in [168.9, 202.6]$. 
Projection on *drive-wheels* and *curb-weight*
IV. Conclusions and future work

New *subspace clustering* algorithm

- Handle discrete features
- One user parameter: $K$ (or $BIC$)
- **Interpretable and visual output**
- Resistant to noise
- Effective acceleration heuristic
- Incremental / [Pelleg and Moore, 2001]

Future work

- Perform **hard feature selection** during the learning process
- Extension to supervised, semi-supervised, partially-supervised learning, local feature selection


In SIAM Int. Conf. on Data Mining.


Example

Rules generated:

- $R_0 \rightarrow d_2 \in [84, 104], d_8 \in [27, 47], d_{10} = A, d_{13} = D, d_{14} = D$
- $R_1 \rightarrow d_5 \in [25, 43], d_8 \in [51, 68]$
- $R_2 \rightarrow d_2 \in [55, 79], d_4 \in [75, 92], d_9 \in [69, 92], d_{14} = E$

Result of SSC (10 runs):

- $R_0 \rightarrow d_2 \in [84, 104], d_{10} = A, d_{14} = D$
- $R_1 \rightarrow d_5 \in [25, 43], d_8 \in [51, 68]$
- $R_2 \rightarrow d_2 \in [55, 79], d_4 \in [75, 92], d_{14} = E$
Projection on $d_8$ and $d_{14}$
Projection on $d_2$ and $d_5$
More

- Robust even if data generated uniformly inside given intervals
- Resistant to missing data (ignore)
Feature Selection

\[ W_{kd} = \begin{cases} 
1 - \frac{\sigma_{kd}^2}{\sigma_d^2}, \text{ with } \sigma_d^2 = \frac{\sum_i (X_{id} - \mu_{kd})^2}{N} & \text{if } d \text{ continuous} \\
\frac{\text{Freqs}_{kd} \,(\text{cat}) - \text{Frequencies}_d \,(\text{cat})}{1 - \text{Frequencies}_d \,(\text{cat})} & \text{if } d \text{ discrete} \\
\text{with } \text{cat} = \text{Argmax}_{c \in \text{Categories}_d} \text{Freqs}_{kd} \,(c) & 
\end{cases} \]
Expectation

Membership probability of a data point to a cluster $P(\vec{X}_i|\theta_k)$

= product of membership probabilities on each dimension

$$\Rightarrow \frac{1}{\sqrt{2\pi\sigma_{kd}}} e^{-\frac{1}{2} \left( \frac{x_{id} - \mu_{kd}}{\sigma_{kd}} \right)^2} \quad \text{if } d \text{ continuous}$$

$$\Rightarrow Freq_{kd}(X_{id}) \quad \text{if } d \text{ discrete}$$
Maximization

\[
\pi_k = \frac{1}{N} \sum_i P(\theta_k | \vec{X}_i)
\]

\[
\mu_{kd} = \frac{\sum_i X_{id} \times P(\theta_k | \vec{X}_i)}{\sum_i P(\theta_k | \vec{X}_i)}
\]

\[
\sigma_{kd} = \sqrt{\frac{\sum_i P(\theta_k | \vec{X}_i) \times (X_{id} - \mu_{kd})^2}{\sum_i P(\theta_k | \vec{X}_i)}}
\]

\[
\text{Freqs}_{kd}(\text{cat}) = \frac{\sum \{i | X_{id} = \text{cat} \} P(\theta_k | \vec{X}_i)}{\sum_i P(\theta_k | \vec{X}_i)} \quad \forall \text{ cat } \in \text{Categories}_d
\]