A Scalable Framework for Discovering Coherent Co-clusters in Noisy Data

M. Deodhar, H. Cho, G. Gupta, J. Ghosh, I. Dhillon

Electrical and Computer Engineering
University of Texas at Austin, USA

June, 2009

Table of contents

Small Clusters in Large Datasets

- Microarray data
- Market basket data
- Transaction data in fraud detection applications

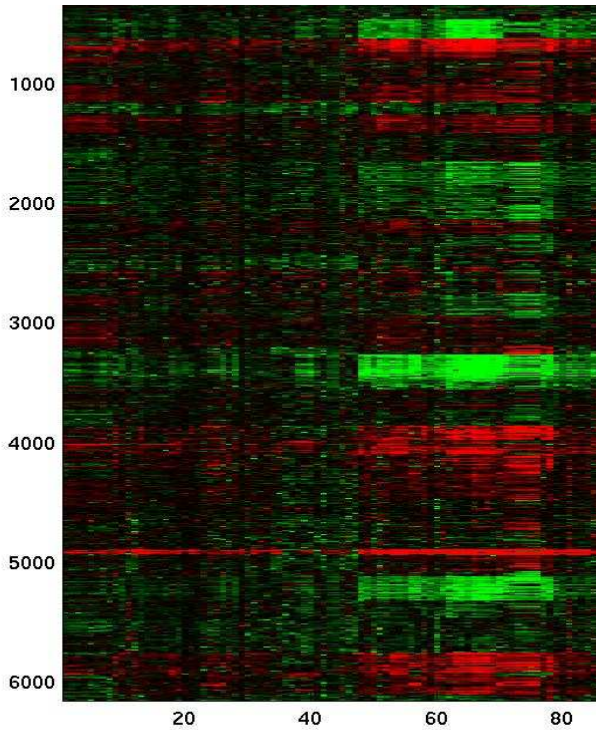


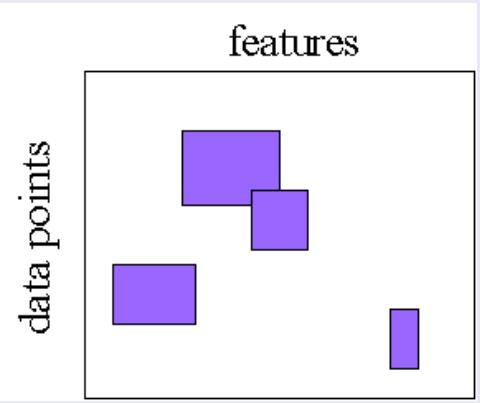
Figure: Gene-condition co-clusters

Clustering Challenges

- Coherent patterns obfuscated by large amount of noise
- Large number of irrelevant points and features
- Clusters exist in different subspaces of features
- Overlapping clusters

Problem

Want to find co-clusters with the most general structure



Related Work

- Density based clustering
  - e.g. DBSCAN, OPTICS, Bregman Bubble Clustering
  - Restricted to one-sided clustering
- Co-clustering
  - Iterative greedy algorithms - Biclustering, Plaid Model
  - Deterministic - OPSM
  - Partitional grid based - Bregman Co-clustering

Related Work

- Density based clustering
  - e.g. DBSCAN, OPTICS, Bregman Bubble Clustering
  - Restricted to one-sided clustering
- Co-clustering
  - Iterative greedy algorithms - Biclustering, Plaid Model
  - Deterministic - OPSM
  - Partitional grid based - Bregman Co-clustering
- Subspace clustering
  - e.g. CLIQUE, PROCLUS, pCluster
  - Shifting and scaling patterns - Reg-cluster model

Related Work

- Density based clustering
  - e.g. DBSCAN, OPTICS, Bregman Bubble Clustering
  - Restricted to one-sided clustering
- Co-clustering
  - Iterative greedy algorithms - Biclustering, Plaid Model
  - Deterministic - OPSM
  - Partitional grid based - Bregman Co-clustering
- Subspace clustering
  - e.g. CLIQUE, PROCLUS, pCluster
  - Shifting and scaling patterns - Reg-cluster model

Robust Overlapping Co-Clustering

- Discovering dense, overlapping co-clusters in large datasets
- Robust to presence of irrelevant points and features
  - Automatically detected and pruned away
- Robust to data noise model

ROCC: Key Idea

- 2 step approach
  - Prune data points and features
  - Agglomerate co-clusters

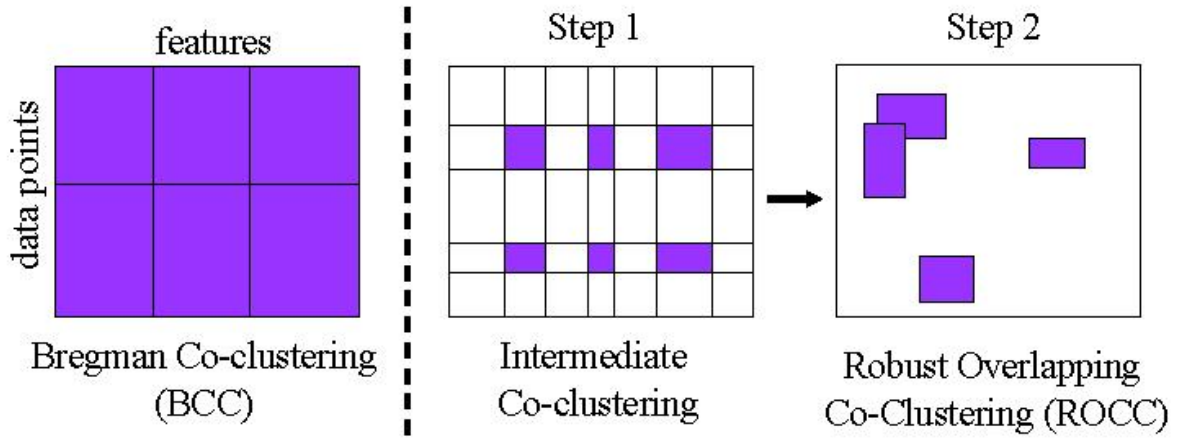


Figure: ROCC Concept

Distinguishing Features

- Systematically developed cost function
- Applicability to all Bregman divergences
- Ability to handle missing values
- Simultaneous detection of all co-clusters
- Scalability to large, high-D datasets

Problem Definition

- Step 1
  - Matrix $Z_{m \times n}$, s_r, s_c - # rows and columns to be clustered
  - $\mathcal{K}, \mathcal{L}$ - sets of clustered rows and columns
  - ρ - mapping from s_r rows $\in \mathcal{K}$ to k row clusters
  - γ - mapping from s_c columns $\in \mathcal{L}$ to l column clusters

Objective function

- Find $\mathcal{K}, \mathcal{L}, (\rho, \gamma)$ to minimize

$$\sum_{u \in \mathcal{K}, v \in \mathcal{L}} w_{uv}(z_{uv} - \hat{z}_{uv}(\rho(u), \gamma(v)))^2$$

- Squared error only over the $s_r \times s_c$ clustered elements

Problem Definition

- Step 1
  - Matrix $Z_{m \times n}$, s_r, s_c - # rows and columns to be clustered
  - $\mathcal{K}, \mathcal{L}$ - sets of clustered rows and columns
  - ρ - mapping from s_r rows $\in \mathcal{K}$ to k row clusters
  - γ - mapping from s_c columns $\in \mathcal{L}$ to l column clusters

Objective function

- Find $\mathcal{K}, \mathcal{L}, (\rho, \gamma)$ to minimize

$$\sum_{u \in \mathcal{K}, v \in \mathcal{L}} w_{uv}(z_{uv} - \hat{z}_{uv}(\rho(u), \gamma(v)))^2$$

- Squared error only over the $s_r \times s_c$ clustered elements

- Step 2
  - Agglomerate co-clusters using distance measure
    dist(cc1,cc2) = approximation error of combined co-cluster

Problem Definition

- Step 1
  - Matrix $Z_{m \times n}$, s_r, s_c - # rows and columns to be clustered
  - $\mathcal{K}, \mathcal{L}$ - sets of clustered rows and columns
  - ρ - mapping from s_r rows $\in \mathcal{K}$ to k row clusters
  - γ - mapping from s_c columns $\in \mathcal{L}$ to l column clusters

Objective function

- Find $\mathcal{K}, \mathcal{L}, (\rho, \gamma)$ to minimize

$$\sum_{u \in \mathcal{K}, v \in \mathcal{L}} w_{uv}(z_{uv} - \hat{z}_{uv}(\rho(u), \gamma(v)))^2$$

- Squared error only over the $s_r \times s_c$ clustered elements

- Step 2
  - Agglomerate co-clusters using distance measure
    dist(cc1,cc2) = approximation error of combined co-cluster

Approximation Schemes

- 6 ways of approximating matrix entries $\hat{z}_{uv}$
- Block co-clusters
  - Matrix entries approximated by co-cluster mean
  - Finds uniform blocks
- Pattern based co-clusters
  - Matrix entries approximated by co-cluster row mean + co-cluster col mean - co-cluster mean
  - Captures trends in data values

ROCC Meta-Algorithm

Step 1

Begin with an initial co-clustering (ρ, γ)

Iterate until convergence

(1) Update co-cluster models

(2a) Update row clusters

Assign each row to the closest row cluster

Pick s_r rows, closest to assigned row clusters

(2b) Update col clusters

Assign each col to the closest col cluster

Pick s_c cols, closest to assigned col clusters

Step 2

Prune less coherent co-clusters

Merge similar co-clusters

Addressing the Local Minimum Problem

- Random initialization for ρ, γ could lead to poor local minima
- Our strategy
  - Cluster all the data
  - Iteratively shave off data points and features till s_r rows and s_c columns are left
  - $s_r^{(1)}$ and $s_c^{(1)}$ initialized to m and n
  - Decayed exponentially till $s_r^{(j)} = s_r$ and $s_c^{(j)} = s_c$

Generative Model Intuition

- Mixture of $k \times l$ exponential family distributions and a background uniform distribution
- Element z_{ij} generated by

$$P(z_{ij}) = \sum_{I=1}^{k} \sum_{J=1}^{l} \alpha_{IJ} f_{\psi}(z_{ij}|\theta_{i,j,I,J}) + \alpha_0 p_0$$

- Each element belongs to $k \times l$ components and background with certain probability
- EM algorithm to fit *soft* ROCC model

Results on Synthetic Datasets

Dataset	m,n	# co-clusters	Type
1	500, 500	3	block
2	500, 200	4	block
3	500, 500	3	pattern
4	500, 200	4	pattern

Table: Synthetic Datasets

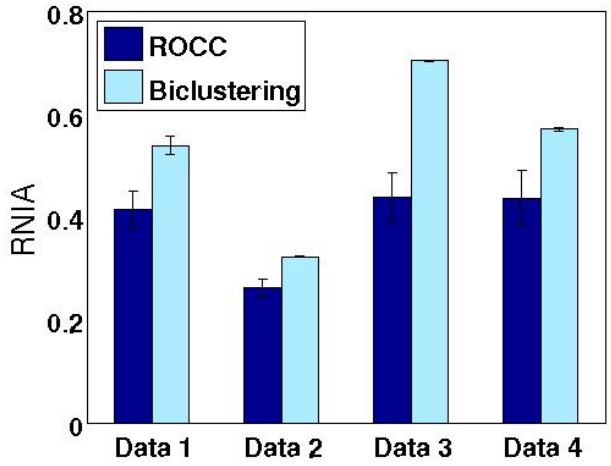


Figure: Comparison of ROCC and Biclustering

Matrix Reconstruction

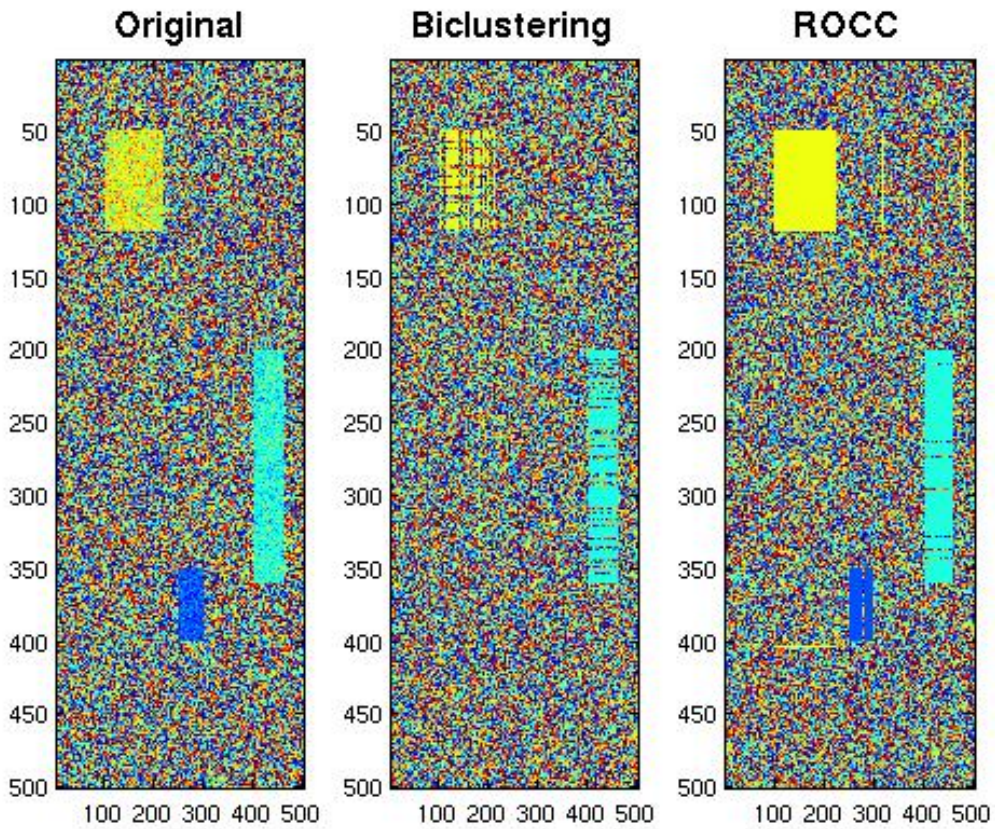
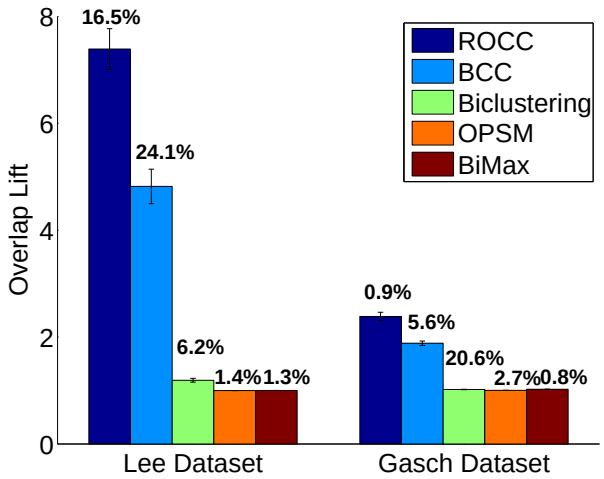


Figure: Reconstruction by Biclustering and ROCC

Microarray Datasets

- Lee dataset - 5612 yeast genes, 591 experiments
- Gasch dataset - 6151 yeast genes, 173 environmental stress conditions
- Ground truth: pairwise linkages between functionally related genes
- Evaluation measure
  - Overlap Lift: how many times more correct links are predicted as compared to random chance
- Aim: find most coherent 150-200 co-clusters

Results on Microarray Data



genes	Category(Coverage)	p-value
20	tRNA ligase (8/36)	6.63e-14
63	ER membrane (14/84)	3.886e-14
20	PF00270-DEAD (12/51)	<1e-14
12	Glycolysis (8/16)	<1e-14
24	PF00660-SRP1-TIP1 (22/30)	<1e-14

Table: Biologically significant clusters found by ROCC on the Lee dataset.

Simultaneous Feature Selection and Clustering

- Classification of lung cancer tissue samples
  - 12533 genes and 181 human tissue samples
  - 2 classes: malignant pleural mesothelioma (31 samples) and adenocarcinoma (150 samples)
- Recover 2 sample classes using genes as features
- Presence of redundant, non-informative genes

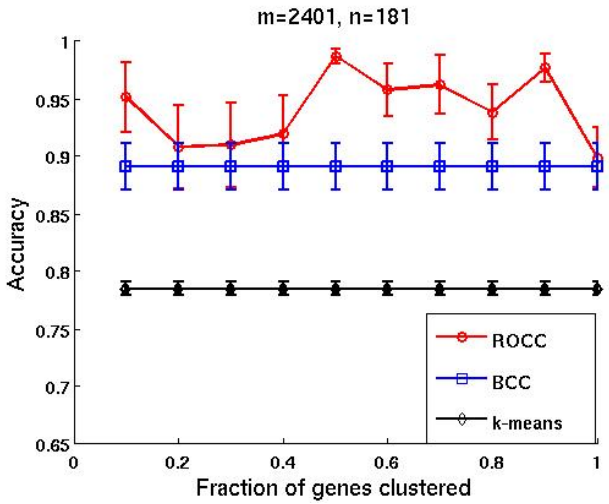


Figure: Lung Cancer data: sample clustering accuracy

Concluding Remarks

- ROCC accurately finds dense, overlapping co-clusters in noisy, high-D data
  - by identifying and pruning away irrelevant parts of the data
- Robust to choice of parameters
  - e.g. # rows/cols to be clustered need not be exactly specified
- Interesting applications in market data analysis, text mining