A Comparison of Software Implementations of SOM Clustering Procedures

by

Haiyan Li, University of Maryland
Bruce Golden, University of Maryland
Edward Wasil, American University
Paul Zantek, University of Maryland

Presented at ANNIE 2002, November 2002

Focus of the Paper

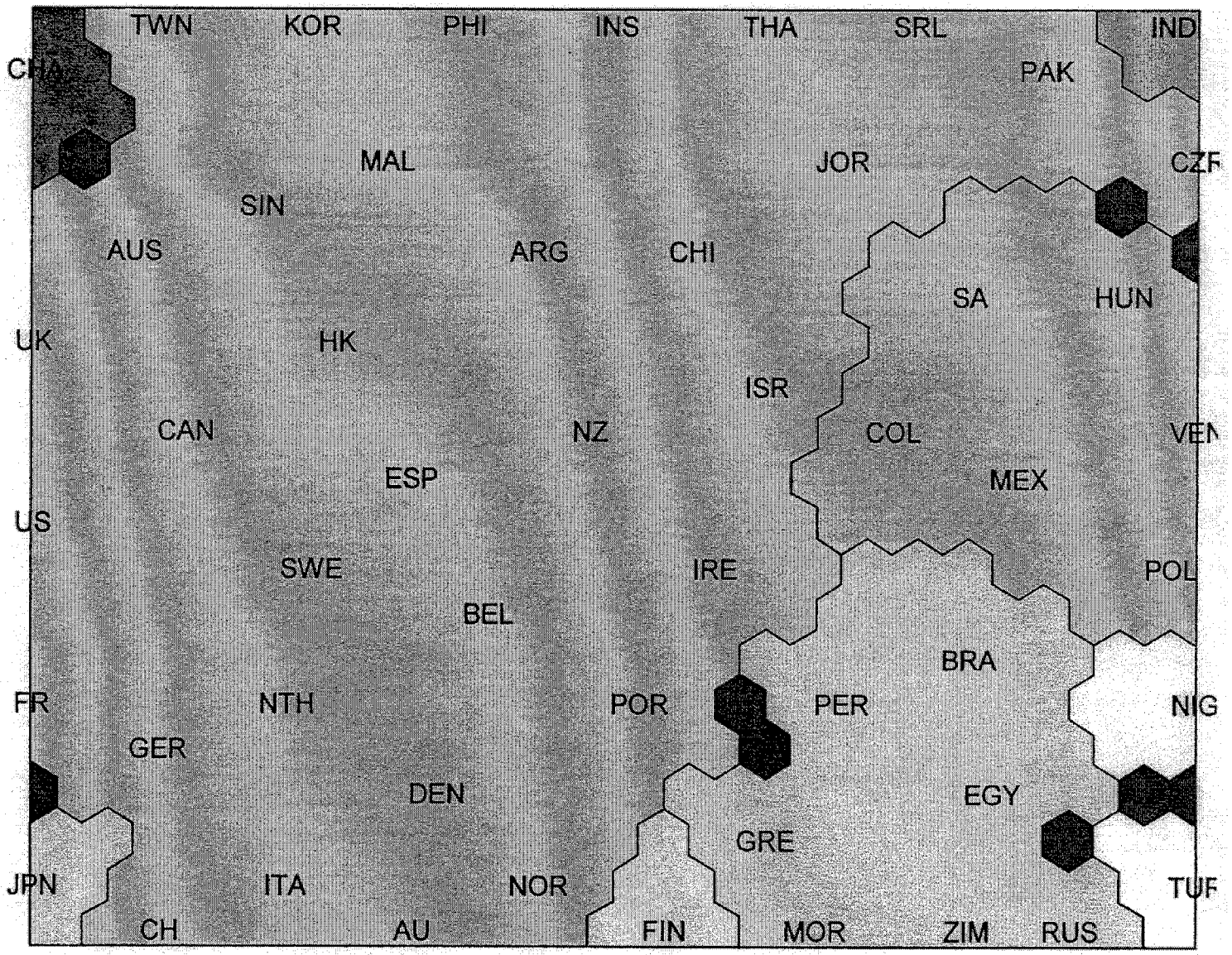
- Self-organizing maps (SOMs) are routinely used in clustering
- New software for SOM-based clustering continues to emerge
- How well do these software packages perform?
- We construct 96 data sets and evaluate the performance of 4 SOM-based clustering procedures as well as the K-means algorithm
- Classification accuracy is measured using the cluster recovery rate and the Rand statistic.

Introduction

- Clustering is a common activity in data mining
- The goal is to partition the observations of a data set into clusters
- The observations within a cluster should be similar
- Observations in different clusters should be dissimilar
- Numerous applications in biology, business, and engineering

Self-Organizing Maps (SOMs)

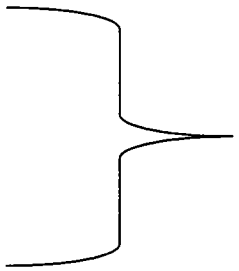
- Developed by Teuvo Kohonen in early 1980s
- Observations are mapped onto a two-dimensional hexagonal grid
- Related to MDS and Sammon maps, but ensures better spacing
- Colors are used to indicate clusters
- Software: SOM_PAK (Public domain, WWW), Viscovery (Eudaptics, Austria)



Clustering of countries based on country risk measures using Viscovery

Software Implementations Studied

- SOM-based Viscovery procedures
  - Ward clustering
  - modified-Ward clustering
  - single linkage clustering

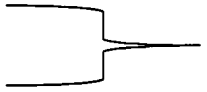


Viscovery SOMine 4.0

- Classic SOM clustering

SOM_Pak

- K-means algorithm



Clementine

Our Approach

- Start with "easy" problems
- Apply the procedures to problems for which the clusters are already known
- We construct 96 data sets in which the clusters are well separated
- In Figure 1, we see a two-dimensional plot of a four-cluster data set

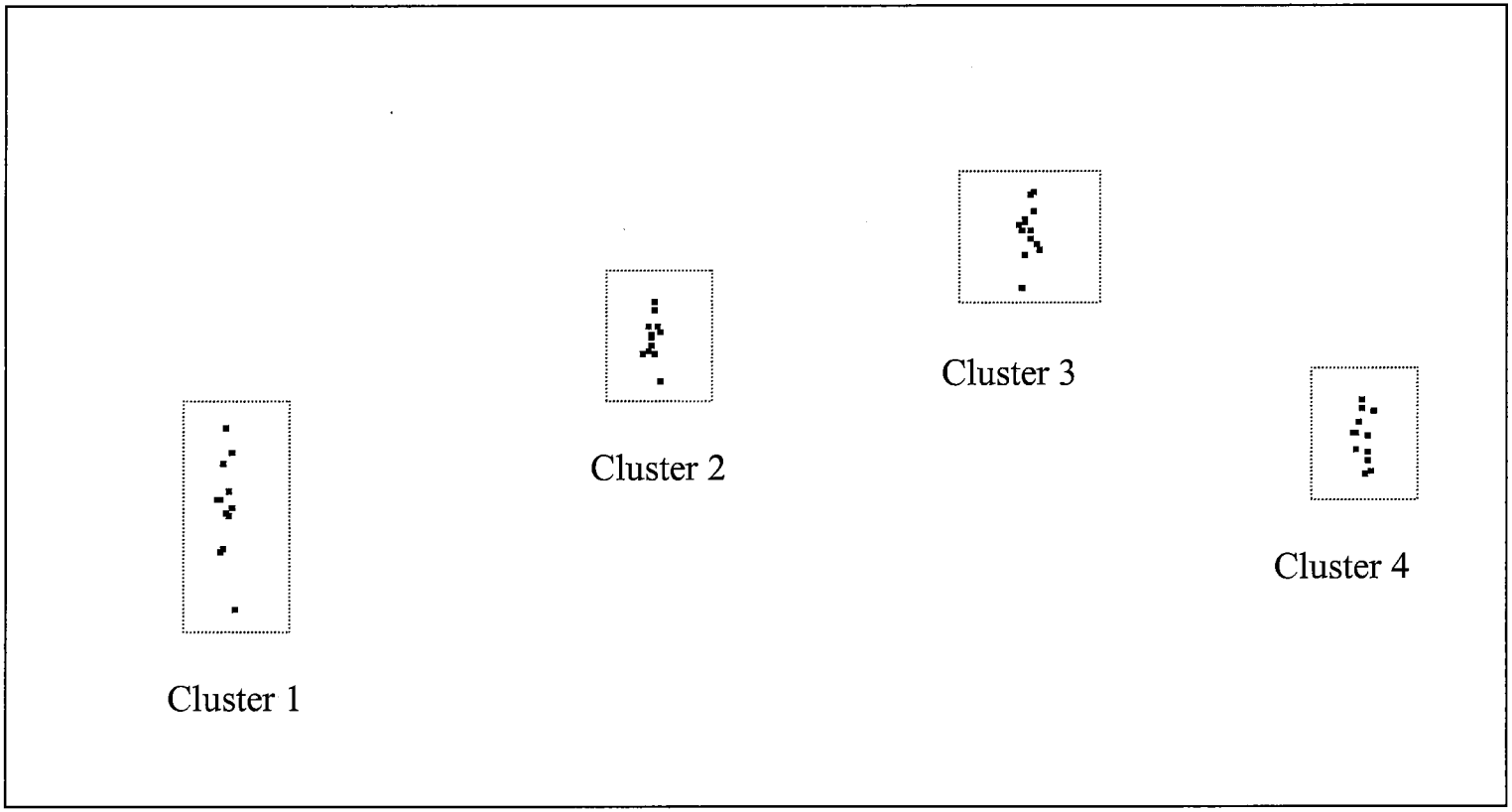


Figure 1. An example of a four-cluster data set.

Experimental Design

Factors	Values
# of clusters	3, 4, 5, 6
# of dimensions	3, 4
# of data points	50, 100, 150, 200
amount of internal dispersion	low, medium, high

- Using this design, we construct 4x2x4x3=96 data sets

Constructing Data Sets and Measuring Performance

- The multivariate normal distribution is used to construct clusters that exhibit external isolation and internal cohesion (see paper for details)

- Cluster recovery rate
  - the proportion of times a clustering procedure correctly recovers the cluster structure
  - the percentage of times a procedure correctly determines the cluster membership of each and every data point.

- The Rand Statistic

The Rand Statistic

Table 1. Pairwise classification notation.

Clustering Procedure Solution	Correct Solution: Pair in Same Cluster	Correct Solution: Pair Not in Same Cluster
Pair in Same Cluster	*A*	*B*
Pair Not in Same Cluster	*C*	*D*

- The Rand statistic provides the proportion of correct pairwise classifications for the data set and equals (A+D)/(A+B+C+D)

- The Rand statistic equals one when the solution generated by the clustering procedure is correct

Results

Table 2. Cluster recovery rates (in %).

SOM-Ward	SOM-Modified Ward	SOM-Single Linkage	SOM-Classic	*K*-Means
92.7	91.7	82.3	14.6	80.2

- SOM-Ward recovers the true clusters in 89 of 96 data sets (89/96=.927)
- SOM-Ward and SOM-Modified Ward perform very well
- SOM-Single Linkage and K-Means perform well
- SOM-Classic performs poorly

Results - - continued

Table 3. Cluster recovery rates (in %) by level of dispersion.

	Level of Dispersion		
Procedure	Low	Medium	High
SOM-Ward	100	94	84
SOM-Modified Ward	100	91	84
SOM-Single Linkage	100	91	56
SOM-Classic	19	16	9
K-Means	88	78	75

- As the intra-cluster dispersion increases, internal cohesion of clusters is reduced and cluster recovery rates decrease

- At all levels of dispersion, the first two procedures perform best

Results - - continued

Table 4. Values of the Rand statistic.

Clustering Procedure	Number of Clusters 3	4	5	6	Row Average
	Low Intra-Cluster Dispersion				
SOM-Ward	1.000	1.000	1.000	1.000	1.000
SOM-Modified Ward	1.000	1.000	1.000	1.000	1.000
SOM-Single Linkage	1.000	1.000	1.000	1.000	1.000
SOM-Classic	0.846	0.899	0.911	0.886	0.886
K-Means	1.000	1.000	0.988	0.965	0.988
	Medium Intra-Cluster Dispersion				
SOM-Ward	0.994	0.989	1.000	1.000	0.996
SOM-Modified Ward	0.995	0.986	1.000	0.997	0.995
SOM-Single Linkage	1.000	0.999	1.000	0.999	0.999
SOM-Classic	0.898	0.878	0.893	0.893	0.890
K-Means	0.995	1.000	0.988	0.931	0.979
	High Intra-Cluster Dispersion				
SOM-Ward	0.914	0.954	0.984	1.000	0.963
SOM-Modified Ward	0.951	0.971	0.996	1.000	0.980
SOM-Single Linkage	0.946	0.977	0.992	0.991	0.977
SOM-Classic	0.915	0.931	0.876	0.898	0.905
K-Means	1.000	0.960	0.990	0.950	0.975

Results - - continued

- Each entry in Table 4 is an average over 8 data sets
- As the level of dispersion increases (especially for 3 or 4 clusters), the performance of each procedure generally deteriorates
- Both SOM-Classic and K-Means require the user to specify the number of clusters in advance
- Viscovery, on the other hand, does not have this requirement
- Viscovery can determine the number of clusters on its own

Results - - continued

Table 5. Cluster recovery rates (in %) for Viscovery (number of clusters is not specified).

SOM-Ward	SOM-Modified Ward	SOM-Single Linkage
83.3	82.3	71.9

- For each procedure, the recovery rate drops about 10 percentage points from the recovery rate generated when number of clusters was specified (Table 2)

- These results are still competitive with the recovery rate from K-Means (80.2%) when K-Means has the advantage of knowing the true number of clusters

Conclusions

- We evaluated the performance of 4 SOM-based clustering procedures when the clusters are well separated

- The three procedures in Viscovery SOMine 4.0 performed well, better than K-Means, and much better than the procedure in SOM_Pak

- Viscovery users who are not sure of the number of clusters may rely on the package to determine the number of clusters

- Bottom line: Viscovery seems to do reasonably well