Hybridized System Design for Feature Selection in High Dimensional Gene Expression Data

Haider Banka
Associate Professor, Department of Computer Science & Engineering, Indian Institute of Technology (ISM), Dhanbad 826004
email: haider@iitism.ac.in

Outline

- Need for feature selection in gene expression data
- Reasons for using rough sets and multi-objective GA
- Basics of rough sets and GA and MOGA
- Pre-processing of gene expression data
- Formulation in multi-objective framework
- The Complete algorithm using NSGA II
- Experimental results and comparisons on bench mark data sets
- Conclusions and future directions

Basic Need: High complexity of high dimensional data

Reduce dimensionality

Remove irrelevant data

Improve learning accuracy

Enhancing output comprehensibility

Feature selection methods

- ***Filter methods:***

(Yu et al. (2003), Dash et al. (2002)) are based on performance evaluation functions calculated directly from the training data such as **distance, information, dependency, and consistency, and select features subsets** *without involving any* ***learning algorithm.***

- ***Wrapper methods:***

(Kohavi et al. (1997a)) require ***one predetermined learning algorithm*** and use its estimated ***performance as the evaluation criterion***. They attempt to find features better suited to the learning algorithm aiming to improve performance.

Generally, the ***wrapper method achieves better performance*** than the filter method, but tends to be ***more computationally expensive*** than the filter approach.

Challenges of Gene Expression Data

- High dimensional (curse of dimensionality)
- Highly redundant (performance degraded)
- Many missing values
- Noisy

Need for Feature Selection in gene expression data

- Large number of features (genes), the majority of which are not relevant to the description of the problem
- Can degrade the classification/clustering performance by masking the contribution of the relevant features.

Data sets Used	Classes	#Samples	#Attributes	The Three Data sets
Colon	Cancer Normal	40 22	2000	
Lymphoma	Other type B-cel-lymp.	54 42	4026	
Leukemia	ALL AML	47 25	7129	

A partial view of gene expression data

17 conditions (out of thousands conds.)

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
139	69	0	69	139	139	139	139	69	0	0	69	110	0	69	0	0
0	69	69	69	110	110	110	110	69	0	69	69	110	110	69	0	69
139	110	0	69	69	110	139	139	139	0	69	69	139	69	69	0	0
139	110	0	69	110	110	110	139	110	0	69	110	139	69	69	0	69
208	179	110	69	110	110	110	161	161	0	69	69	110	0	69	0	69
0	0	0	69	69	139	161	179	139	0	69	0	110	0	69	0	69
0	0	0	0	110	110	110	69	110	0	0	0	69	0	69	0	69
179	161	69	69	69	110	69	110	110	0	0	69	0	0	69	0	69
69	110	69	110	110	161	110	69	139	69	69	110	110	139	110	69	110
69	0	69	69	110	139	110	0	0	0	69	69	110	69	69	0	0
139	161	110	110	139	179	139	110	139	69	69	69	110	110	110	69	69
179	179	161	139	161	195	161	161	161	110	161	161	139	139	161	110	110
179	240	161	195	195	256	220	208	240	139	195	195	195	161	195	161	110
161	161	69	110	139	161	139	110	161	69	110	139	69	69	110	69	69
208	283	240	248	264	304	283	283	283	195	220	240	240	240	248	195	208
161	195	110	139	195	248	179	161	220	110	179	195	161	179	208	110	110
139	161	139	161	139	179	161	139	69	69	139	69	69	179	179	110	69
304	326	304	322	326	350	340	376	318	248	314	283	314	318	326	264	264
69	69	0	69	110	110	69	0	69	0	69	69	139	69	69	0	0
283	208	220	277	289	326	289	289	248	220	271	240	271	294	277	230	208
337	383	383	413	414	403	381	393	343	350	369	358	347	358	356	314	289
161	161	220	195	161	195	161	110	110	110	195	179	179	69	139	110	110
208	195	220	161	139	161	161	110	139	110	195	195	195	69	161	139	139
248	230	330	300	277	240	240	179	195	220	277	289	240	240	220	161	161
264	300	289	264	277	277	289	277	300	248	283	271	294	256	264	271	283
230	240	289	264	240	256	220	208	220	248	271	256	256	240	220	179	208
439	442	464	456	451	422	417	403	432	510	438	442	450	462	419	476	476
256	230	208	240	230	248	240	283	248	220	230	230	220	240	248	220	240
374	322	322	300	330	356	361	333	369	376	369	374	369	343	361	393	399
139	195	161	139	161	139	161	139	179	110	110	139	139	139	110	161	161
230	277	256	248	264	271	248	240	256	220	230	230	256	208	208	240	230
494	470	498	488	477	460	466	484	449	532	485	473	464	487	477	492	484
326	248	240	289	300	294	289	264	277	248	283	283	277	283	277	271	283
179	139	110	69	69	110	69	69	69	69	69	69	69	69	110	69	69
326	411	397	383	371	347	314	277	330	264	289	283	304	264	264	340	343
161	220	220	220	208	208	161	161	208	179	195	179	179	161	139	161	139
220	271	248	230	240	248	240	179	248	208	208	220	230	220	179	230	230
220	271	230	208	161	195	161	161	195	161	208	195	220	161	179	195	220
179	195	110	161	139	179	161	179	161	69	110	139	139	139	161	139	161

40 genes

Genes

Samples

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
139	69	0	69	139	139	139	139	69	0	0	69	110	0	69	0	0
0	69	69	69	110	110	110	110	69	0	69	69	110	110	69	0	69
139	110	0	69	69	110	139	139	139	0	69	69	139	69	69	0	0
139	110	0	69	110	110	110	139	110	0	69	110	139	69	69	0	69
208	179	110	69	110	110	110	161	161	0	69	69	110	0	69	0	69
0	0	0	69	69	139	161	179	139	0	69	0	110	0	69	0	69
0	0	0	0	110	110	110	69	110	0	0	0	69	0	69	0	69
179	161	69	69	69	110	69	110	110	0	0	69	0	0	69	0	69
69	110	69	110	110	161	110	69	139	69	69	110	110	139	110	69	110
69	0	69	69	110	139	110	0	0	0	69	69	110	69	69	0	0
139	161	110	110	139	179	139	110	139	69	69	69	110	110	110	69	69
179	179	161	139	161	195	161	161	161	110	161	161	139	139	161	110	110
179	240	161	195	195	256	220	208	240	139	195	195	195	161	195	161	110
161	161	69	110	139	161	139	110	161	69	110	139	69	69	110	69	69
208	283	240	248	264	304	283	283	283	195	220	240	240	240	248	195	208
161	195	110	139	195	248	179	161	220	110	179	195	161	179	208	110	110
139	161	139	161	139	179	161	139	69	69	139	69	69	179	179	110	69
304	326	304	322	326	350	340	376	318	248	314	283	314	318	326	264	264
69	69	0	69	110	110	69	0	69	0	69	69	139	69	69	0	0
283	208	220	277	289	326	289	289	248	220	271	240	271	294	277	230	208
337	383	383	413	414	403	381	393	343	350	369	358	347	358	356	314	289
161	161	220	195	161	195	161	110	110	110	195	179	179	69	139	110	110
208	195	220	161	139	161	161	110	139	110	195	195	195	69	161	139	139
248	230	330	300	277	240	240	179	195	220	277	289	240	240	220	161	161
264	300	289	264	277	277	289	277	300	248	283	271	294	256	264	271	283
230	240	289	264	240	256	220	208	220	248	271	256	256	240	220	179	208
439	442	464	456	451	422	417	403	432	510	438	442	450	462	419	476	476
256	230	208	240	230	248	240	283	248	220	230	230	220	240	248	220	240
374	322	322	300	330	356	361	333	369	376	369	374	369	343	361	393	399
139	195	161	139	161	139	161	139	179	110	110	139	139	139	110	161	161
230	277	256	248	264	271	248	240	256	220	230	230	256	208	208	240	230
494	470	498	488	477	460	466	484	449	532	485	473	464	487	477	492	484
326	248	240	289	300	294	289	264	277	248	283	283	277	283	277	271	283
179	139	110	69	69	110	69	69	69	69	69	69	69	69	110	69	69
326	411	397	383	371	347	314	277	330	264	289	283	304	264	264	340	343
161	220	220	220	208	208	161	161	208	179	195	179	179	161	139	161	139
220	271	248	230	240	248	240	179	248	208	208	220	230	220	179	230	230
220	271	230	208	161	195	161	161	195	161	208	195	220	161	179	195	220
179	195	110	161	139	179	161	179	161	69	110	139	139	139	161	139	161

Why Rough Sets and Multi-objective GA?

- Reduct in rough set theory correspond to the minimal feature sets necessary and sufficient to represent a correct classification decision.

- Reduct can
  - (i) classify among all elements of the universe with the same accuracy as the starting attribute set, and are
  - (ii) of small cardinality.

- The conflicting nature of these characteristics moots the suitability of multi-objective modelling.

Rough set preliminaries

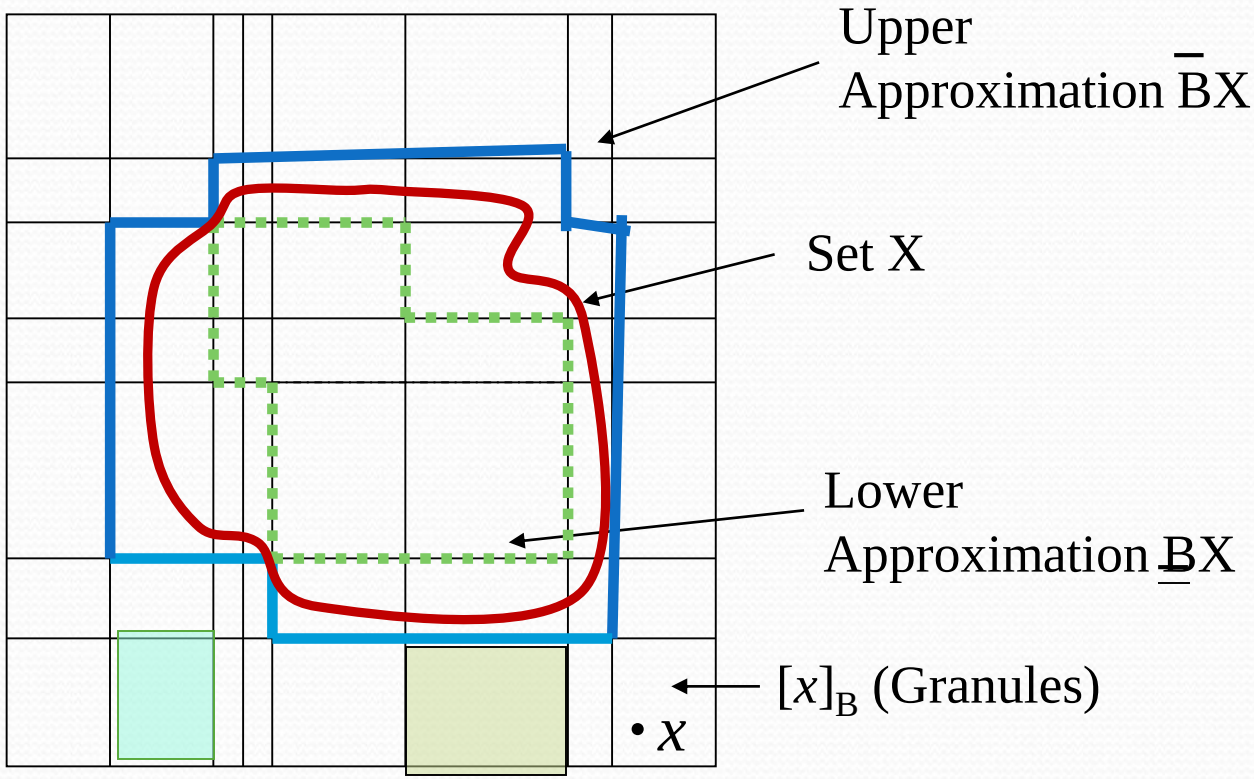
Information system {U, A(C,D)}

Indiscernibilty Relation, Discernibility matrix, Distinction table

Lower and Upper approximation

Reducts

Rough Sets



$[x]_B$ = set of all points belonging to the same granule as of the point x in feature space Ω_B.

$\Longrightarrow$ $[x]_B$ is the set of all points which are *indiscernible* with point x in terms of feature subset B.

Approximations of the set $X \subseteq U$ w.r.t feature subset B

B-lower: $\underline{B}X = \{x \in U : [x]_B \subseteq X\}$ Granules definitely belonging to *X*

B-upper: $\overline{B}X = \{x \in U : [x]_B \cap X \neq \phi\}$ Granules definitely and possibly belonging to *X*

If $\underline{B}X = \overline{B}X$, *X* is *B-exact* or *B-definable*

Otherwise it is *Roughly definable*

Discernibility Matrix

A discernibility matrix is defined as an m x m matrix of the information system with the $(i, j)^{th}$ entry c_{ij} given by

$$c_{ij} = \left\{ a \in C : a\left(x_i\right) \neq a\left(x_j\right), \wedge\left(x_i, x_j\right) \neq IND(D) \right\}, i, j \in \{1, 2, ..., m\}$$

Distinction Table

– A binary matrix $\frac{\left(m^2 - m\right)}{2} . N$ where N is the number of attributes. An entry $b((k, j), i)$ of the matrix corresponds to the attribute i and pair of objects (x_k, x_j) ,and is given by

$$b\left(\left(k, j\right), i\right) = \begin{cases} 1, if a_i(x_k) \neq a_i(x_j) \\ 0, if a_i(x_k) = a_i(x_j) \end{cases}$$

Rough Set Rule Generation

Decision Table:

Object	F_1	F_2	F_3	F_4	F_5	Decision
x_1	1	0	1	0	1	Class 1
x_2	0	0	0	0	1	Class 1
x_3	1	1	1	1	1	Class 1
x_4	0	1	0	1	0	Class 2
x_5	1	1	1	0	0	Class 2

Discernibility Matrix (c) for Class 1:

$$c_{ij} = \{a : a(x_i) \neq a(x_j)\}, 1 \leq i, j \leq p\}$$

Objects	x_1	x_2	x_3
x_1	ϕ	F_1, F_3	F_2, F_4
x_2		ϕ	F_1,F_2,F_3,F_4
x_3			ϕ

Using Distinction Table

	a1	a2	a3	a4	a5	a6	a7	a8
(m1,n1)	1	0	0	1	0	1	0	0
(m1,n2)	0	0	0	0	1	0	0	1
(m1,n3)	1	0	1	0	1	0	1	1
(m2,n1)	1	0	0	0	0	1	0	0
(m2,n2)	0	0	1	0	0	0	1	0
(m2,n3)	1	1	0	0	1	0	0	1

10001001

$$L_v^r = 3, N = 8, C_v^r = 5$$

$$m_1 = 2, m_2 = 3$$

$$f_1(\overset{r}{v}) = \frac{N - L_v^r}{N} = (8-3)/8 = 0.625$$

$$f_2(\overset{r}{v}) = \frac{C_v^r}{m_1 * m_2} = (5/6) = 0.833$$

(m * n) x N

Genetic Algorithm

- Genetic algorithms are global stochastic optimization techniques based on natural genetics.
- Robust and non-problem specific.
- GAs code the parameter set of the optimization problem as finite-length string.
- GAs start the searching from a population of random points, improve the quality of the population over time by genetic operations: selection, crossover, mutation;
- The best fitted solution will be evolved toward objective function.

Simple genetic Algorithm

Basic Operations:
Selection, Cross-over, Mutation

- Basic Term:

Chromosome, Population, Fitness/Objective Function

Multi-objective GA

Dominance Criteria

Non-dominated sorting

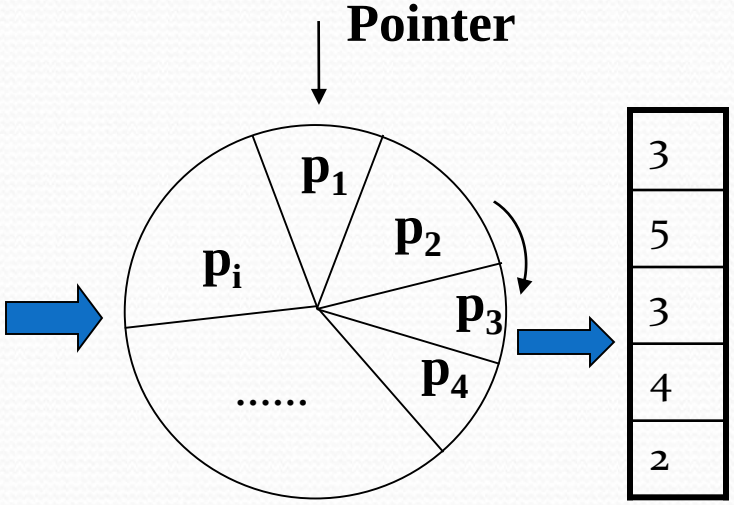
Crowding distance

Crowding tournament selection operator

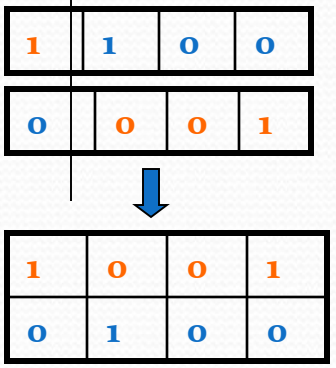
The Non-dominated sorting Genetic Algorithm (NSGA II)

Parent population

Populations of Chromosome (s)				x	$f(x) = x^* x$
1	0	1	0	10	100
0	1	1	0	6	36
1	1	0	0	12	144
1	0	1	0	10	100
0	0	0	1	3	1
0	0	1	1	3	9



Cross-over X-site



mutation

Child population

1	0	0	1	9	81
0	1	0	0	4	16
1	1	0	1	13	169
1	0	1	0	10	100
1	0	1	0	10	100
0	0	1	1	3	9

Multi-objective Genetic Algorithms (MOGAs)

- Deal with multiple, often competing objectives.
- Present a set of Pareto optimal solutions

- Example: purchasing of a car (finance available, comfort, distance to be driven by each day, no. of passengers riding, fuel consumption and cost, depreciation value, road conditions etc.)

Dominance Criteria

Definition: If there are *M* objective functions, a solution $x^{(1)}$ is said to dominate solution $x^{(2)}$, If both conditions 1 and 2 are true:

1) The solution $x^{(1)}$ is *no worse* than $x^{(2)}$ in all the *M* objectives.

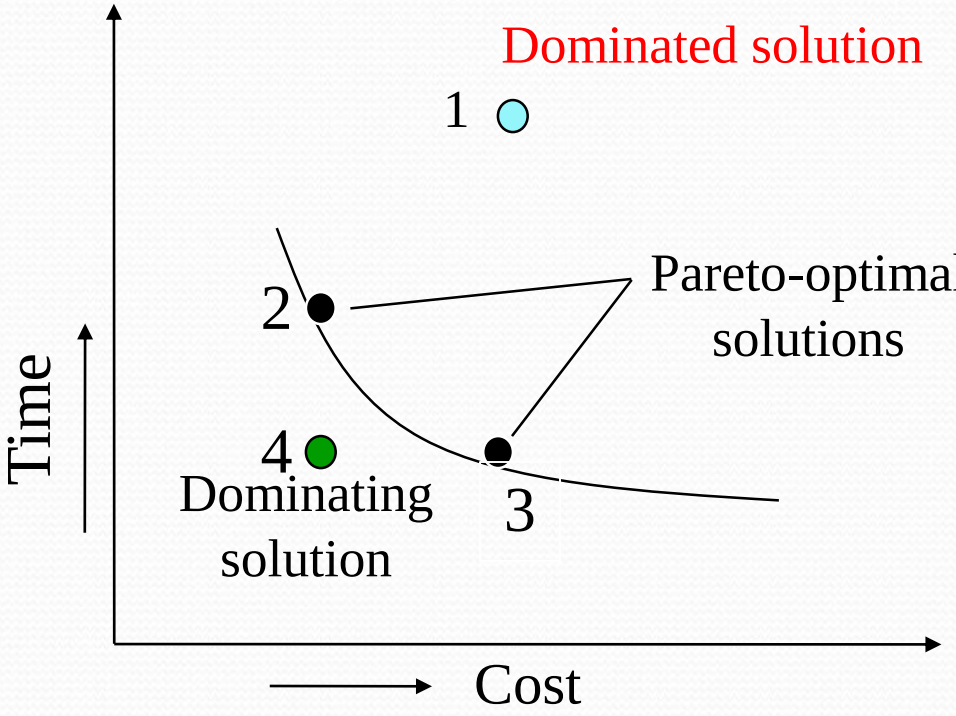
2) The solution $x^{(1)}$ is *Strictly better* than $x^{(2)}$ in *at least one* of the *M* objectives.

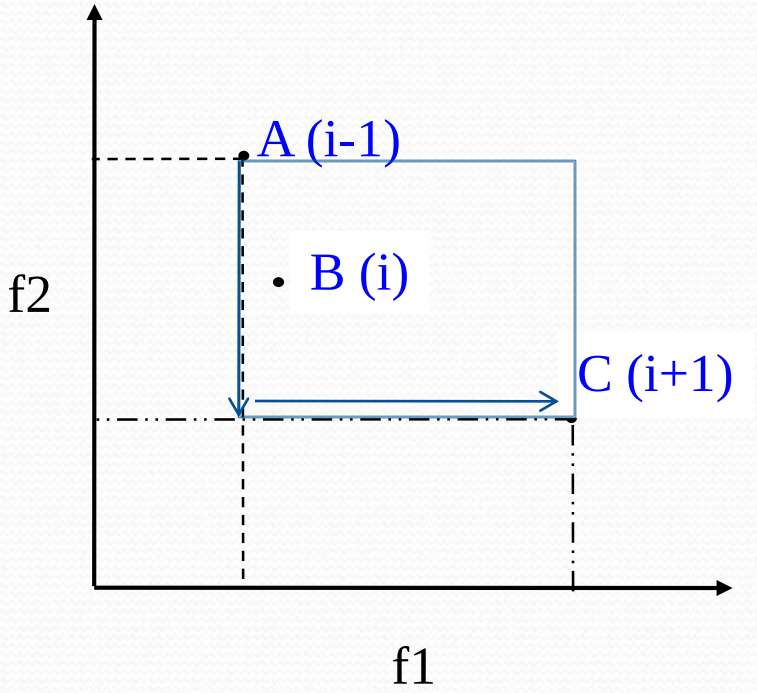
Otherwise, the two solutions are non-dominating to each other.

When a solution $x^{(1)}$ dominates a solution $x^{(2)}$, then rank of $x^{(1)} < x^{(2)}$

Pareto optimal solution

- A solution x is pareto-optimal if there doesn't exist any other solutions that dominate x.
- equally good; non-dominated;





Algorithm for finding non-dominated set in a population *P* of size |*P*|

Step 1) Set counter i=1 and $P' = \varnothing$

Step 2) for each solution $j \in P(j \neq i)$, check if j dominates i. If yes, go to Step 4)

Step 3) If more solution left in P, increment j by one and go to Step 2). Else Set $P' = P' \cup \{i\}$.

Step 4) Increment I by one. If $i \leq |P|$ then go to Step 2). Else declare P' as the non-dominated set

The process is repeated with $P = P - P'$ until $P = \varnothing$

Crowding distance

1) Let the no. of solutions in F be $l = |F|$ and assign $d_i = 0$ for $i = 1, 2, \ldots, l$

2) For each objective function $f_k, k = 1, 2, \ldots, M$, sort set in its worse order (ascending/descending)

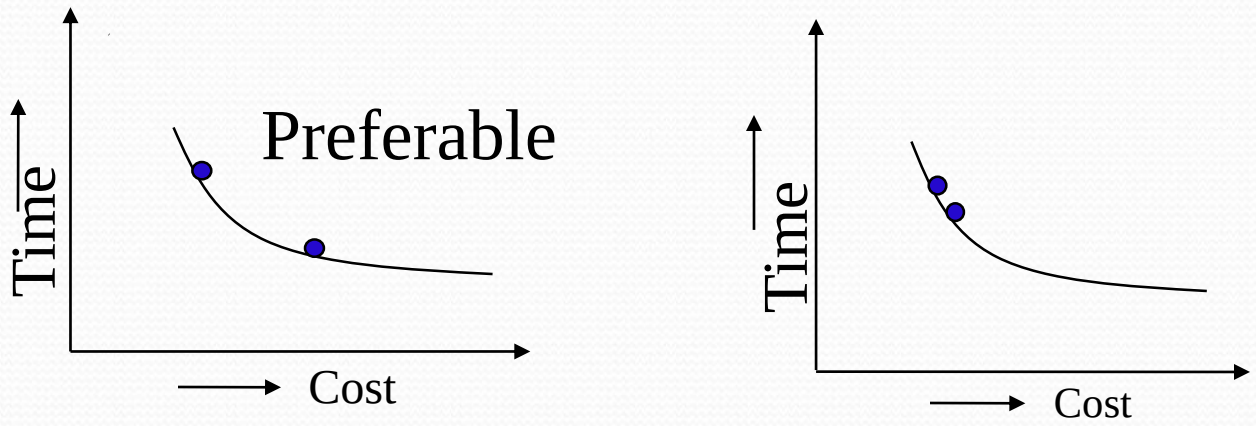
3) Set $d_1 = d_l = \infty.$

4) For $j = 2$ to $(l-1)$ increment d_j by $(f_{k_{j+1}} - f_{k_{j-1}})$

Goals of multi-objective solutions

✓Find a set of solutions as close as possible to the Pareto optimal front. (Non dominated sorting)

✓Find a set of sparsely spaced solutions, as far as possible. (Crowding distance operator)



Crowding Selection Operator

A solution i *wins tournament* with another solution j *if any one of* the following is true.

1) Solution i *has better rank, i.e.,* $r_i < r_j$

2) Both the solutions are in the same front, i.e., $r_i = r_j$ *, but* solution i *is less densely located in the search space, i.e.,* $d_i > d_j$

NSGA II Algorithm

1. Initialize the population randomly.
2. Calculate the multi-objective fitness functions.
3. Rank the population using dominance criteria.
4. Calculate the crowding distance.
5. Do selection using crowding selection operator.
6. Do crossover and mutation to generate children population.
7. Combine parent and children population.
8. Replace the parent population by the best members of the combined population.

Evolutionary Rough Feature Selection

Preprocessing (Normalization, Thresholding, and Discretization)

Making of distinction table

Fitness functions for Evaluation

The complete algorithm

Experimental results and comparison (Colon, Leukemia, and Lymphoma datasets)

Pre-processing

- Normalize (attribute-wise).

$$a'_j(x_i) = \frac{a_j(x_i) - \min_j}{\max_j - \min_j}, \forall i$$

  - Choose Th_i and Th_f using quartiles (or partition values are the values of a variate which divides the total frequency into a number of equal parts)

quartile

$$Th_k = l_c + \frac{R_k - cfr_{c-1}}{fr_c} \cdot \delta$$

l_c is the lower limit of the C th class interval

$k = 1, 2, 3$ for four partition

$$R_k = \frac{N \cdot k}{4}$$

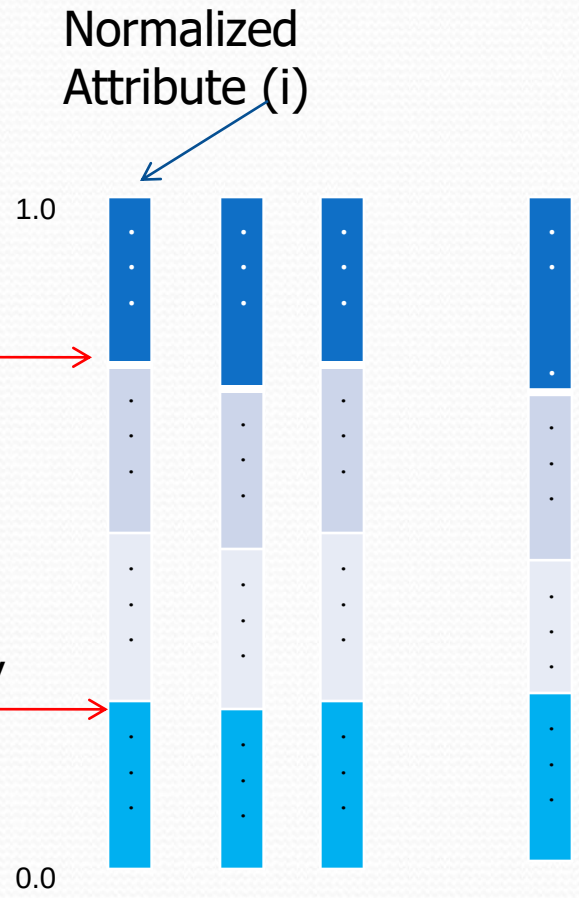
δ is the small class interval width and

fr_c is the count of the corresponding class frequency

cfr_{c-1} Cumulative frequency of preceding class interval

Such that $cfr_{c-1} \leq R_k \leq cfr_c$

We use $Th_i = Th_1, Th_f = Th_3$



Convert the attribute value table into binary

- if $a'(x_i) \leq Th_i$ then Put ‘o'
- else if $a'(x_i) > Th_f$ then Put 'ı'
- else Put '*' (don't care).
- Find the average '*' of the whole attributes (Choose this as threshold Th_a).
- Remove those attributes whose no. of '*' are $> Th_a$ from the table.

Creation of distinction table

- Objects in the same class do not constitute a row in the distinction table.
- if either of the objects in a pair has '*' entry under an attribute, then Put a 'o' in the entry of that attribute and pair in the table.
- An 'ı' in the table corresponds to the attributes of interest for classification decision.

Table after discretization

		a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8
Class 1	M1	0	1	*	0	1	*	0	1
	M2	1	0	1	1	0	1	1	0
Class 2	N1	1	1	0	1	1	0	1	1
	N2	0	0	*	0	0	*	0	0
	N3	*	1	1	*	1	1	*	1

Corresponding distinction table

Object Pairs	a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8
M1,N1	1	0	0	1	0	0	1	0
M1,N2	0	1	0	0	1	0	0	1
M1,N3	0	0	0	0	0	0	0	0
M2,N1	0	1	0	0	1	0	0	1
M2,N2	1	0	0	1	0	0	1	0
M2,N3	0	1	0	0	1	0	0	1

Fitness Function

- Solutions represented as binary strings of length *N*, where *N* is the number of condition attributes
- "1/0" indicates presence/absence of corresponding attribute,

L_v^r is no. of 1's in the reduct candidate,

C_v^r no. of objects discerned between,

m_1, m_2 no. of objects in two classes

- $f_1(\vec{v}^r) = \frac{N - L_v^r}{N}$ prefers candidates with less attributes

$f_2(\vec{v}^r) = \frac{C_v^r}{m_1 * m_2}$ determines discernibility among objects pairs

- In SGA we have $f = \alpha_1 f_1(\vec{v}) + \alpha_2 f_2(\vec{v})$

Algorithm:

- **Redundancy reduction** for high-dimensional microarray data to generate **reduced attribute** value table.
- d-distinction table generated for two classes being discerned
- **REPEAT** until pre-specified no. of generations
- Random **population** of size P generated.
- **Two fitness** values calculated for each individual.
- **Non-domination sorting** done to identify different fronts.
- Crowding sort, based on **crowding distance**, performed to get wide spread of solution.
- Offspring solution of size n created using **fitness tournament selection**, **crossover and mutation**.
- **Select best populations** of size P from both parent and offspring solutions, to generate a combined population of size P.

The Three Data sets

Data sets Used	Classes	#Samples	#Attributes	#Attributes after Prepocessing	#Attribute using MOEA	#Attribute using GA
Colon	Cancer Normal	40 22	2000	1102	9	15
Lymphoma	Other type B-cel-lymp.	54 42	4026	1867	2	18
Leukemia	ALL AML	47 25	7129	3783	2	19

Using Distinction Table

	a1	a2	a3	a4	a5	a6	a7	a8
(m1,n1)	1	0	0	1	0	1	0	0
(m1,n2)	0	0	0	0	1	0	0	1
(m1,n3)	1	0	1	0	1	0	1	1
(m2,n1)	1	0	0	0	0	1	0	0
(m2,n2)	0	0	1	0	0	0	1	0
(m2,n3)	1	1	0	0	1	0	0	1

10001001

$$L_v^r = 3, N = 8, C_v^r = 5$$

$$m_1 = 2, m_2 = 3$$

$$f_1^r(v) = \frac{N - L_v^r}{N} = (8-3)/8 = 0.625$$

$$f_2^r(v) = \frac{C_v^r}{m_1 * m_2} = (5/6) = 0.833$$

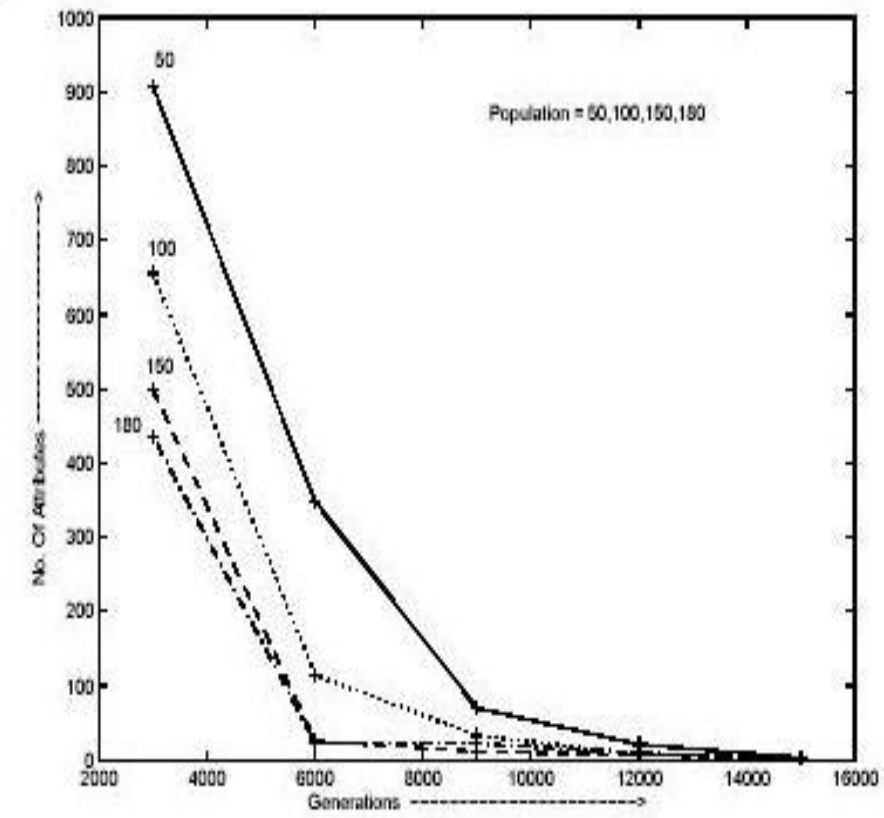
(m * n) x N

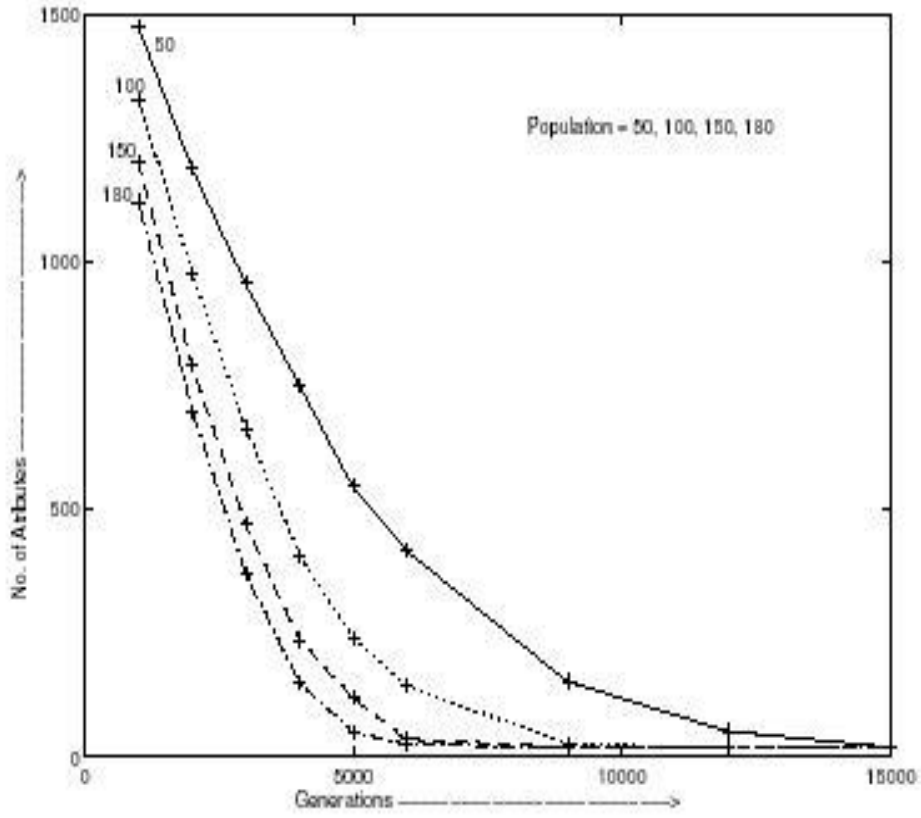
Recognition score (%) using *k*-NN classifier

Dataset	Population size	No. of attributes	*k*-nearest neighbors classification (%) on test set								
			$k = 1$			$k = 3$			$k = 5$		
			C1	C2	Net	C1	C2	Net	C1	C2	Net
Colon:	50	10	80.0	90.9	83.9	75.0	90.9	80.6	75.0	81.8	77.4
# Genes 2000	100	9	90.0	90.9	90.3	90.0	90.9	90.3	90.0	81.8	87.1
Reduce to 1102	200	8	85.0	90.9	87.1	90.0	81.8	87.1	90.0	90.9	90.3
	300	8	75.0	72.7	74.2	80.0	72.7	77.4	80.0	63.6	74.2
Lymphoma:	50	2	92.6	90.5	91.7	96.3	95.2	95.8	96.3	95.2	95.8
# Genes 4026	100	3	92.6	90.5	91.7	96.3	95.2	95.8	96.3	95.2	95.8
Reduce to 1867	200	3	96.3	90.5	93.8	96.3	95.2	95.8	96.3	95.2	95.8
	300	2	92.6	90.5	91.7	96.3	95.2	95.8	96.3	95.2	95.8
Leukemia:	50	3	100.0	85.7	94.1	100.0	78.6	91.2	100.0	78.6	91.2
# Genes 7129	100	3	100.0	78.6	91.2	95.0	85.7	91.2	100.0	78.6	91.2
Reduce to 3783	150	2	90.0	71.4	82.4	90.0	100.0	94.1	90.0	85.7	88.2
	180	2	95.0	71.4	85.3	100.0	71.4	88.2	100.0	71.4	88.2

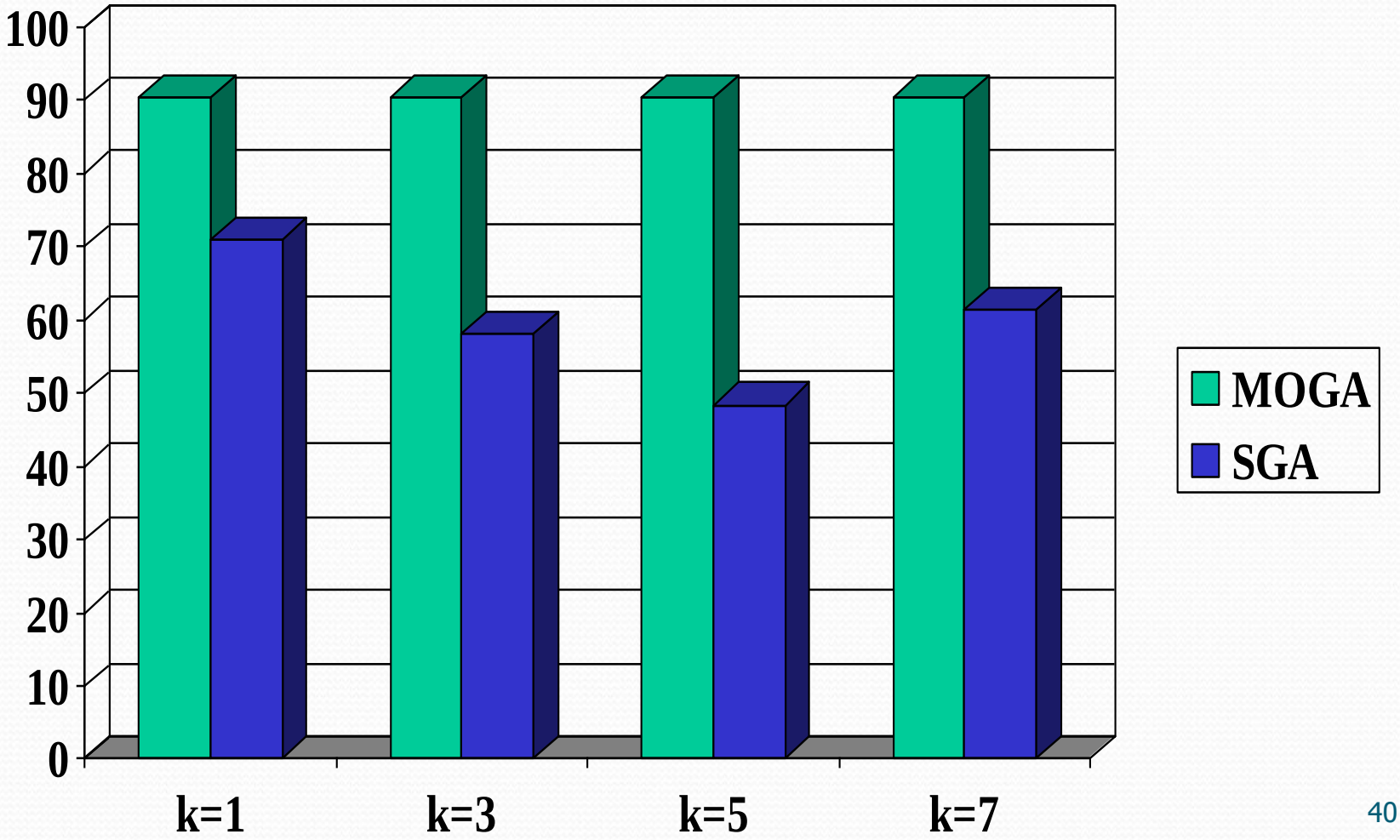
Reduction of Features with generation

MOGA vs SGA (Leukemia data)

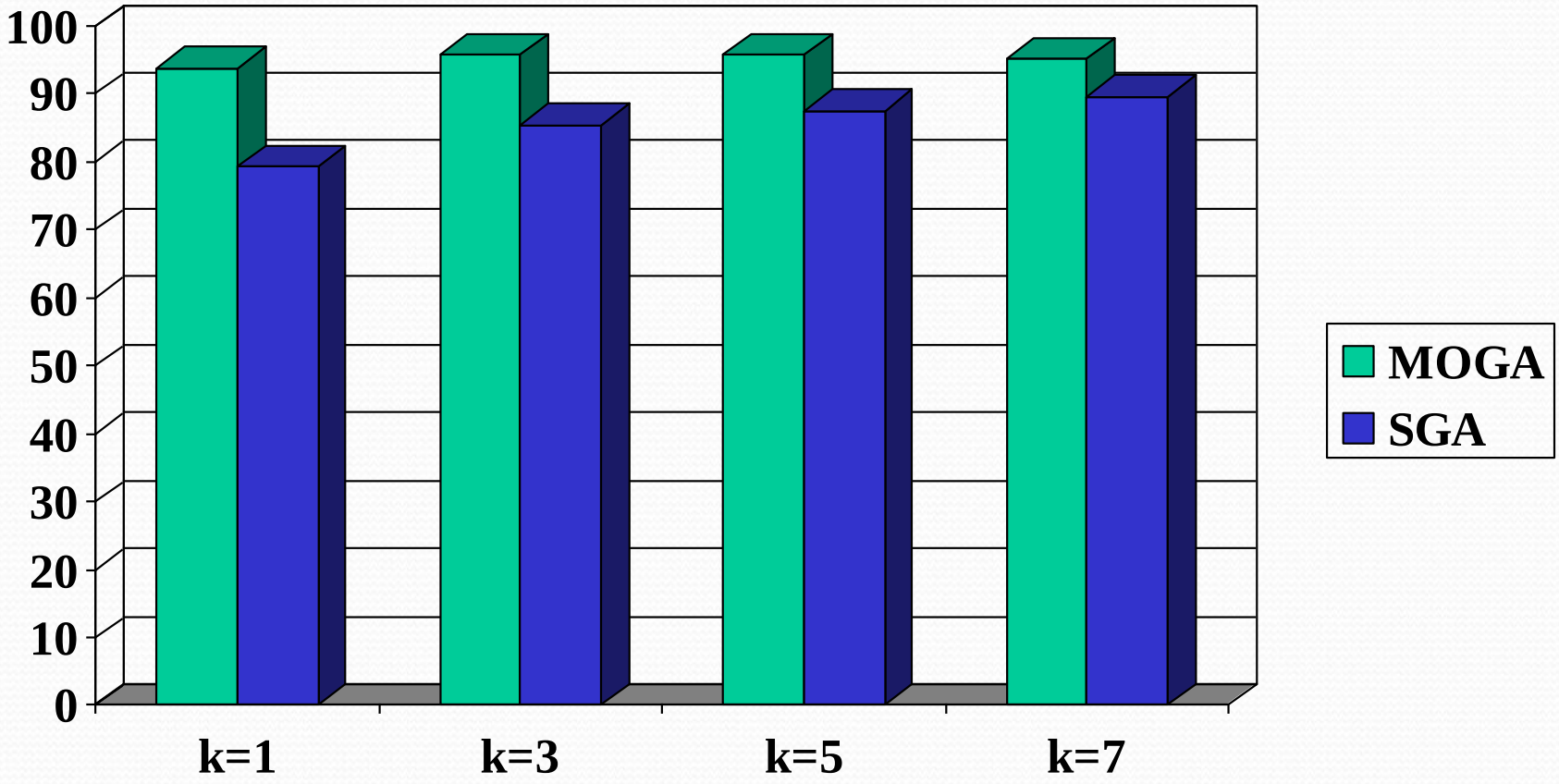




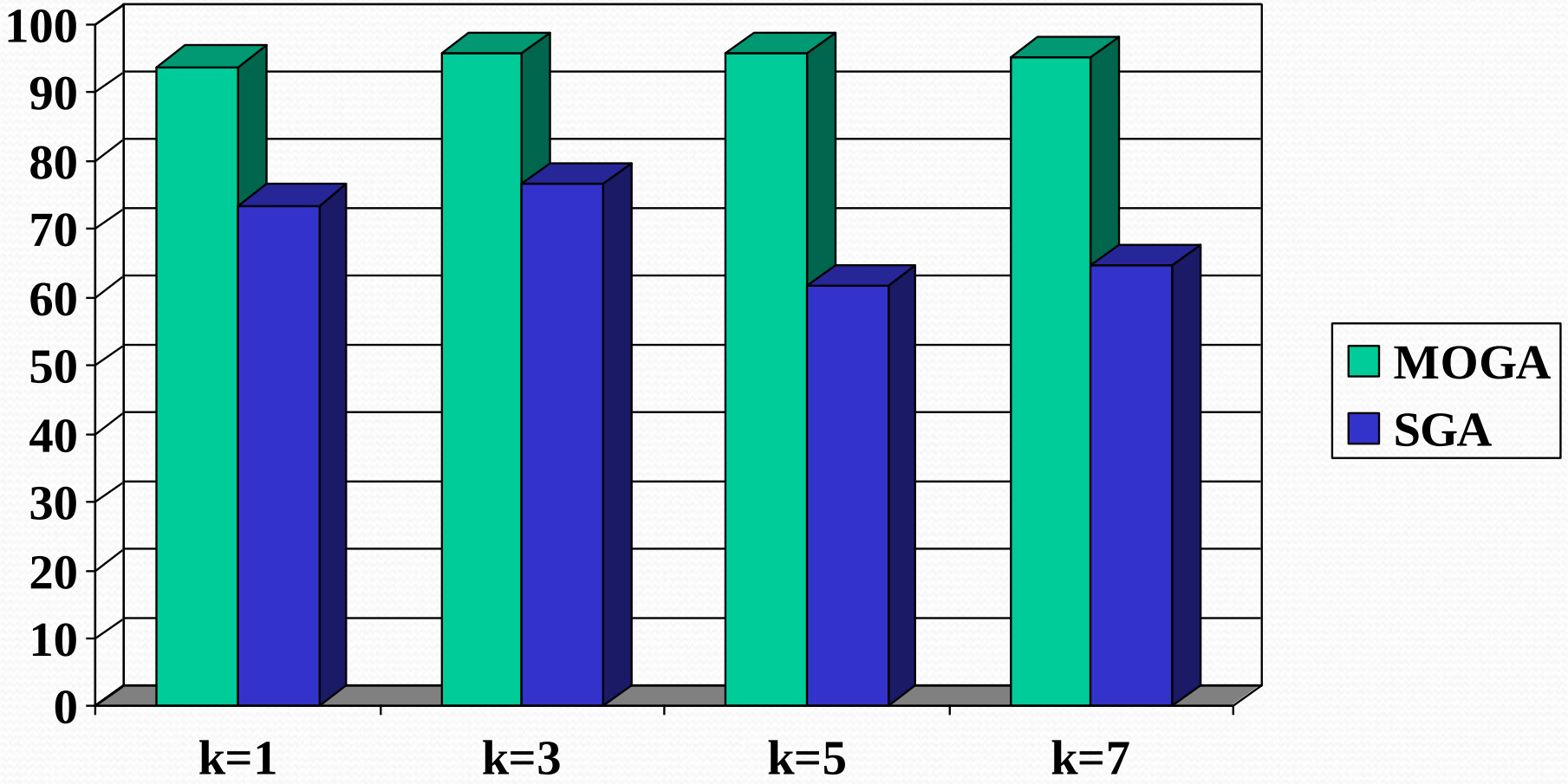
Recognition score (%)
MOGA vs SGA (Colon data)



Recognition score (%)
MOGA vs SGA (Lymphoma data)



Recognition score (%)
MOGA vs SGA (Leukemia data)



Comparison with Other Methods

1. RSA
2. PCA
3. GA
4. Others (Chu et al.), 5 feature (lymphoma)

COMPARATIVE PERFORMANCE AS NUMBER OF MISCLASSIFICATIONS

Dataset	*Leukemia*		*Colon*		
# Genes:	2	3	8	9	10
# Misclassification for: Evolutionary-Rough	2	2	3	3	5
# Genes:	2	4	2	4	8
# Misclassification for: RSA	4	2	8.5 ± 0.58	6.5 ± 1.73	5.0 ± 1.41

Future Work

- Feature selection for multi-class problem is still challenging.
- Suitable for other applications. (e.g., facial recognition)
- Validation of the findings by biologists

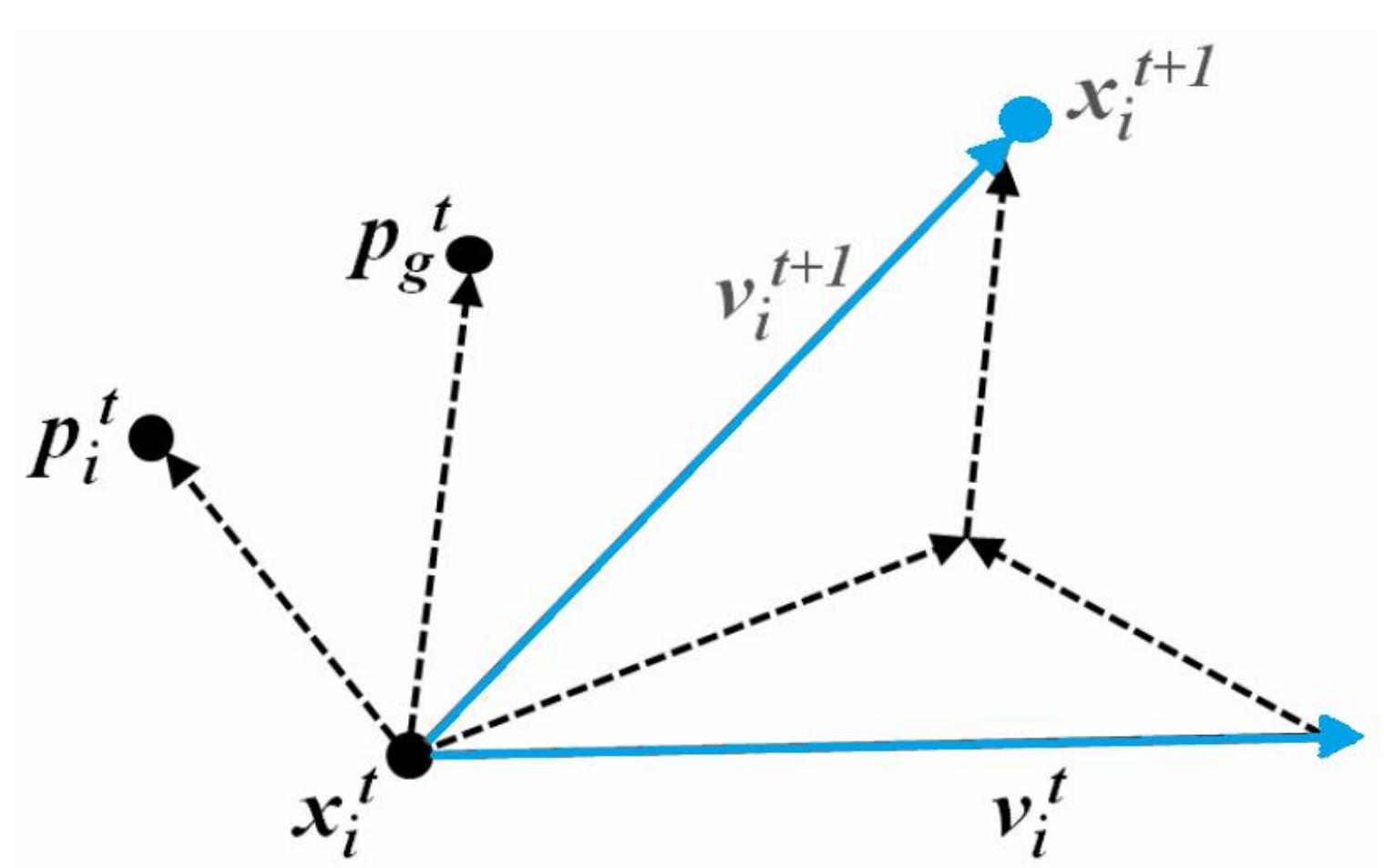
A Binary PSO based Feature Selection Algorithm for Gene expression data Classification

Particle Swarm Optimization (PSO)

- Introduced by Kennedy & Eberhart in 1995
- Inspired by social behaviour and movement dynamics of insects, birds and fish
- Global gradient-less stochastic search method, Suited to continuous variable problems

- PSO is a robust stochastic optimization technique based on the movement and intelligence of swarms.
- It uses a number of agents (particles) that constitute a swarm moving around in the search space looking for the best solution.
- Each particle is treated as a point in a N-dimensional space which adjusts its “flying” according to its own flying experience as well as the flying experience of other particles.

PSO Architecture



Standard PSO Algorithm

x_i^t - Particle position

v_i^t - Particle velocity

p_i^t - Best "remembered" individual particle position

p_g^t - Best "remembered" swarm position

c1,c2 - Cognitive and social parameters

R1,R2 - Random numbers between 0 and 1

w – inertia weight

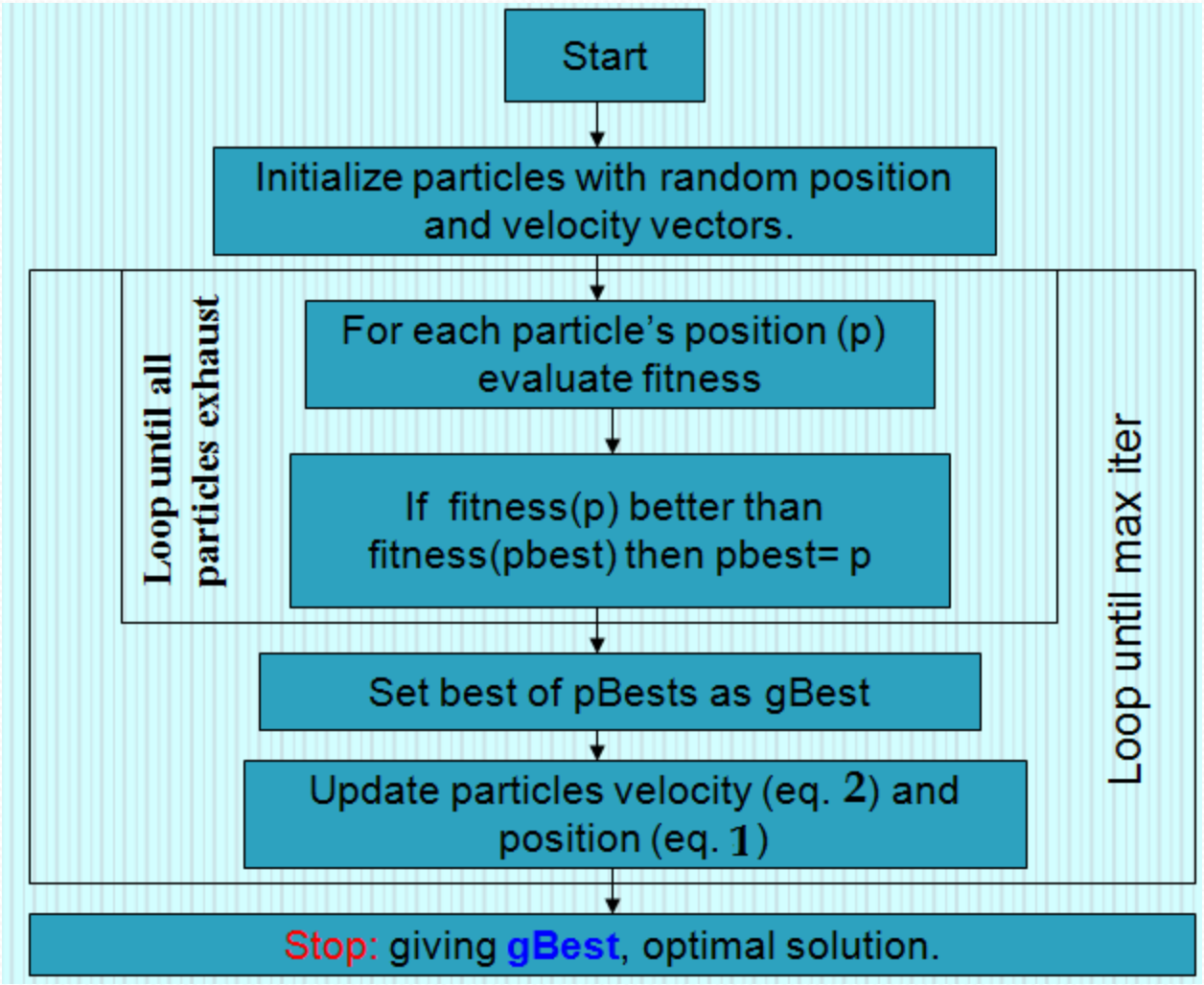
Position of individual particles updated as follows:

$$x_{ij}(t+1) = x_{ij} + v_{ij}(t+1) \quad \text{------(1)}$$

with the velocity calculated as follows:

$$v_{ij}(t+1) = w*v_{ij}(t) + c1*R1*(p_{ij}(t) - x_{ij}(t)) + c2*R2*(p_{gj}(t) - x_{ij}(t)) \quad \text{----(2)}$$

Flow chart depicting the standard PSO Algorithm:



Why PSO..

- Simple implementation, very few algorithm parameters
- Good exploration abilities, very efficient global search algorithm
- Insensitive to scaling of design variables
- Easily parallelized for concurrent processing

Fitness Function

- Solutions represented as binary strings of length N, where N is the number of condition attributes
- "1/0" indicates presence/absence of corresponding attribute,

L_v^r is no. of 1's in the reduct candidate ,

C_v^r no. of objects discerned between L_v^r,

m_1, m_2 no. of objects in two classes

$$f_1(\overset{r}{v}) = \frac{N - L_v^r}{N}$$ prefers candidates with less attributes

$$f_2(\overset{r}{v}) = \frac{C_v^r}{m_1 * m_2}$$ determines discernibility among objects pairs

- Final fitness function: $f = \alpha_1 f_1(\overset{1}{v}) + \alpha_2 f_2(\overset{1}{v})$

where $\alpha_1 + \alpha_1 = 1$

Three Data sets

Data sets Used	Classes	# Samples/ Instances	Features/ Attributes	#Features After Preprocessing
Colon	Cancer normal	40 22	2000	1102
Lymphoma	Other type B-cell lymphoma	54 42	4026	1867
Leukemia	ALL AML	47 25	7129	3783

Proposed Binary PSO for Features selection

- The PSO was extended to Binary Particle Swarm Optimization by Kennedy and Eberhart, In this the position and velocities are restricted to either 0 or 1.
- In BPSO, the updated positions using as follows
-

$$x_{id} = \begin{cases} 1, rand < S(v_{id}) \\ 0, otherwise \end{cases} \qquad \text{---------(3)}$$

where $$S(v_{id}) = \frac{1}{1+e^{-v_{id}}}$$

Example.

x_{id}	1	2	3	…	…	…	…	d
x_1	1	0	0	1	0	0	1	0
v_1	1	0	1	0	1	1	0	0
x_2	0	1	0	0	1	0	0	1
v_2	0	0	0	1	0	1	0	1
..	0	0	0	1	0	1	0	0
x_n	0	1	0	0	1	0	0	1
v_n	1	0	1	0	0	0	1	1

Here, x_1 to x_n are particles, d is dimension, and v is velocity of each particle . Assume pbest=4, gbest=6 .

Take x_1 As input particle, using some initial random velocities (0 or 1) for each dimension.
Assume $pbest_1=1$, and $gbest_1=0$

- Update position eq.(3) based on velocity update eq.(2)
- For $x_{11,}$ R1= 0.846004, R2= 0.521804,
- Equation(2)

$$v_{ij}(t+1) = w*v_{ij}(t) + c1*R1*(p_{ij}(t) - x_{ij}(t)) + c2*R2*(p_{gj}(t) - x_{ij}(t))$$

$$= 0.5*(1)+2*0.846004*(1-1)+2*0.521804*(0-1)$$

$$= 0.5+0+(-1.043608)$$

$$= \mathbf{-0.543608}$$

- update velocity, if velocity exceeds the boundary, set boundary between Vmin= -4.0, Vmax= 4.0
- Then apply eq.(3) ,
- random value=0.131199, sigmoid(eq.2)=0.367349,
- If Sigmoid(V)>rand
- 0.367349>0.131199, then update position $\mathbf{x_{11}}$ as **1**, else 0
- Similarly apply to all dimension $\mathbf{x_{11}}$ **to** $\mathbf{x_{1d,}}$ and apply to all particles $\mathbf{x_1}$ **to** $\mathbf{x_n}$

BPSO-Feature Selection algorithm

Input : c1, c2,w, Vmin, Vmax, d-Distinction table
Output: Reduct features
Start: Initialize random population
While(max iterations)
  For: 1→ n particles(x_1, x_2,…x_n)
    find fitness value (fit) using eq.(3)
    if(fitness of x_i >pbest) then update x_i as pbest
    if(fitness of x_i >gbest) then update x_i as gbest
      For: 1 →D (D dimensions for each particle)
        update velocity of using eq.(2)
        set velocity with in Vmax, Vmin
        based on eq. (3), update positions
      End for
  End for
End while
End

Implementation

- We set parameter values as
- two accelerator coefficients c1, c2=2
- Vmax = 4, Vmin = -4
- Inertia weight w= 0.4 to 0.9
- Population size 10, 20, 30, and 50

Feature subsets selection

Population size	Selected feature subsets using BPSO		
	colon	lymphoma	leukemia
10	10	13	13
20	8	11	10
30	7	8	10
50	6	6	9

Experimental Results

Correct classification score in % using K-NN classifier

Data sets	population	Selected feature subset	K=1		K=3		K=5		K=7	
			Correct	Incorrect	Correct	Incorrect	Correct	Incorrect	Correct	Incorrect
Colon	10	10	100	0	80.65	19.35	80.65	19.65	74.29	20.80
	20	8	100	0	77.42	22.58	74.20	25.80	67.75	32.25
	30	7	100	0	77.42	22.58	64.35	35.48	64.35	35.48
	50	6	100	0	80.65	19.35	74.20	25.80	74.20	25.80
lymphoma	10	13	100	0	97.92	2.08	93.75	6.25	89.58	10.42
	20	11	100	0	89.59	10.41	81.25	18.75	77.09	22.91
	30	8	100	0	89.59	10.41	85.42	14.58	81.25	18.75
	50	6	100	0	79.17	20.83	81.25	18.75	70.84	29.16
leukemia	10	13	100	0	84.22	15.78	86.85	13.15	76.32	23.68
	20	10	100	0	86.85	13.15	81.58	18.42	76.32	23.68
	30	10	100	0	92.11	7.89	84.22	15.78	86.85	13.15
	50	9	100	0	89.48	10.52	84.22	15.78	89.48	10.52

Bayes family classifiers

For colon data

Selected feature subset	used classification method					
	BLR *		BayesNet		NaiveBayes	
	Correct	Incorrect	Correct	Incorrect	Correct	Incorrect
10	93.55	6.45	83.88	16.12	93.55	6.45
8	87.09	12.91	64.52	35.48	83.87	16.13
7	80.65	19.35	64.52	35.48	7.96	29.04
6	64.52	35.48	64.52	35.48	77.42	22.58

* Bayesian Logistic Regression

Bayes family classifiers For lymphoma data

Selected feature subset	used classification method					
	BLR *		BayesNet		NaiveBayes	
	Correct	Incorrect	Correct	Incorrect	Correct	Incorrect
13	97.91	2.09	93.75	6.25	95.84	4.16
11	93.75	6.25	85.41	14.59	95.84	4.16
8	93.75	6.25	85.41	14.59	91.66	8.34
6	81.25	18.75	68.75	31.25	79.17	20.83

* Bayesian Logistic Regression

Bayes family classifiers
For leukemia data

Selected feature subset	used classification method					
	BLR *		BayesNet		NaiveBayes	
	Correct	Incorrect	Correct	Incorrect	Correct	Incorrect
12	94.74	5.26	86.85	13.15	92.11	7.89
10	92.11	7.89	89.48	10.52	94.74	5.26
10	71.05	28.95	71.05	28.95	92.11	7.89
9	73.68	26.32	86.85	13.15	89.48	10.52

* Bayesian Logistic Regression

Tree Based Classification

For colon data

Selected Feature subset	Used classifiers methods													
	BFT		DS		FT		J48		LMT		RF		REPT	
	Correct	Incorrect	Correct	Incorrect	Correct	Incorrect	Correct	Incorrect	Correct	Incorrect	Correct	Incorrect	Correct	Incorrect
10	90.33	9.67	83.88	16.12	100	0	96.78	3.22	96.78	3.22	96.78	3.22	90.32	9.68
8	90.33	9.67	80.65	19.35	100	0	93.55	6.45	87.09	12.91	100	0	90.32	9.68
7	64.52	35.48	74.19	25.81	100	0	83.88	16.12	80.65	19.35	100	0	80.65	19.35
6	87.09	12.91	64.52	35.48	83.88	16.12	90.32	9.68	80.65	19.35	100	0	64.52	35.48

Best-First decision Tree (BFTree), Decision Stump (DS), Functional Trees (FT), J48, Logistic Model Trees (LMT), Random Forest (RF), REPTree (REPT)

Tree Based Classification

For lymphoma data

Selected Feature subset	Used classifiers methods													
	BFT		DS		FT		J48		LMT		RF		REPT	
	Correct	Incorrect	Correct	Incorrect	Correct	Incorrect	Correct	Incorrect	Correct	Incorrect	Correct	Incorrect	Correct	Incorrect
13	97.91	2.09	87.5	12.5	100	0	97.91	2.09	97.91	2.09	97.91	2.09	97.91	2.09
11	95.84	4.16	75.0	25.0	100	0	93.75	6.25	95.84	4.16	100	0	93.75	6.25
8	91.66	8.34	81.25	18.75	95.84	4.16	93.75	6.25	95.84	4.16	100	0	95.84	4.16
6	85.42	14.58	68.75	31.25	81.25	18.75	97.92	2.08	79.17	20.83	100	0	79.17	20.83

Best-First decision Tree (BFTree), Decision Stump (DS), Functional Trees (FT), J48, Logistic Model Trees (LMT), Random Forest (RF), REPTree (REPT)

Tree Based Classification

For leukemia data

Selected Feature subset	Used classifiers methods													
	BFT		DS		FT		J48		LMT		RF		REPT	
	Correct	Incorrect	Correct	Incorrect	Correct	Incorrect	Correct	Incorrect	Correct	Incorrect	Correct	Incorrect	Correct	Incorrect
12	86.85	13.15	86.85	13.15	94.74	5.26	97.37	2.63	89.48	10.52	100	0	97.91	2.09
10	92.11	7.89	92.11	7.89	97.37	2.63	97.37	2.63	94.74	5.26	100	0	92.11	7.89
10	86.85	13.15	73.68	26.32	100	0	92.11	7.89	86.85	13.15	100	0	71.05	28.95
9	92.11	7.89	78.95	21.05	97.37	2.63	97.37	2.63	100	0	100	0	86.85	13.15

Best-First decision Tree (BFTree), Decision Stump (DS), Functional Trees (FT), J48, Logistic Model Trees (LMT), Random Forest (RF), REPTree (REPT)

Function based classification

For colon data

Selected feature subset	used classifier method											
	Liblinear		LibSVM		Logistic		MLP		SGD		SPegasos	
	Correct	Incorrect	Correct	Incorrect	Correct	Incorrect	Correct	Incorrect	Correct	Incorrect	Correct	Incorrect
10	100	0	100	0	100	0	100	0	96.78	3.22	96.78	3.22
8	83.87	16.13	100	0	83.87	16.13	96.78	3.22	90.32	9.68	90.32	9.68
7	83.87	16.13	100	0	80.65	19.35	93.55	6.45	83.88	16.12	74.19	25.81
6	74.19	25.81	100	0	77.42	22.58	87.09	12.91	74.19	25.81	74.19	25.81

Multi Layer Perceptron (MLP), Stochastic Gradient Descent (SGD)

Function based classification

For lymphoma data

Selected feature subset	used classifier method											
	Liblinear		LibSVM		Logistic		MLP		SGD		SPegasos	
	Correct	Incorrect	Correct	Incorrect	Correct	Incorrect	Correct	Incorrect	Correct	Incorrect	Correct	Incorrect
13	100	0	91.66	8.34	100	0	100	0	97.91	2.09	96.78	3.22
11	93.75	6.25	95.84	4.16	91.66	8.34	95.84	4.16	93.75	6.25	93.75	6.25
8	95.84	4.16	93.75	6.25	95.84	4.16	97.91	2.09	93.75	6.25	95.84	4.16
6	79.17	20.83	87.5	12.5	81.25	18.75	91.67	8.33	81.25	18.75	79.17	20.83

Multi Layer Perceptron (MLP), Stochastic Gradient Descent (SGD)

Function based classification

For leukemia data

Selected feature subset	used classifier method											
	Liblinear		LibSVM		Logistic		MLP		SGD		SPegasos	
	Correct	Incorrect	Correct	Incorrect	Correct	Incorrect	Correct	Incorrect	Correct	Incorrect	Correct	Incorrect
12	94.74	5.26	91.66	8.34	100	0	100	0	89.48	10.52	94.74	5.26
10	100	0	100	0	100	0	100	0	89.48	10.52	92.11	7.89
10	84.22	15.78	100	0	100	0	97.37	2.63	86.85	13.15	84.22	15.78
9	100	0	100	0	100	0	100	0	92.11	7.89	97.37	2.63

Multi Layer Perceptron (MLP), Stochastic Gradient Descent (SGD)

Comparison with other Methods

- Probabilistic neural network for feature selection
- Feature selection using Bayesian approach
- Feature selection using PSO
- GA and NSGA-II
- PCA

Comparison with feature selection using Bayesian approach

Data	Method	Classifier Method		
		DS	LibSVM	Logistic
Colon	Bayes-FS	72.6	77.4	71.0
	BPSO	83.88	100	100
leukemia	Bayes-FS	95.8	94.4	97.2
	BPSO	92.11	100	100

Comparison with GA with single objective function

data	Selected subset	Method	K=1			K=3			K=5			K=7		
			C1	C2	Net	C1	C2	Net	C1	C2	Net	C1	C2	Net
colon	15	GA	75	63.6	71	70	36.4	58.1	75	0	48.4	90	9.1	61.3
		BPSO	70	54.55	64.52	80	72.73	77.42	95	36.37	74.2	95	18.19	67.75
lymphoma	18	GA	85.2	71.40	79.2	81.5	90.5	85.4	92.6	81	87.5	92.6	85.7	89.6
		BPSO	92.6	85.71	89.59	88.89	90.49	89.59	88.89	100	93.76	92.6	95.24	93.76
leukemia	19	GA	90	50	73.50	90	57.1	76.5	95	14.3	61.7	100	14.3	64.70
		BPSO	90	50	73.53	100	35.72	73.53	100	7.15	61.77	100	21.43	67.65

Thank you…

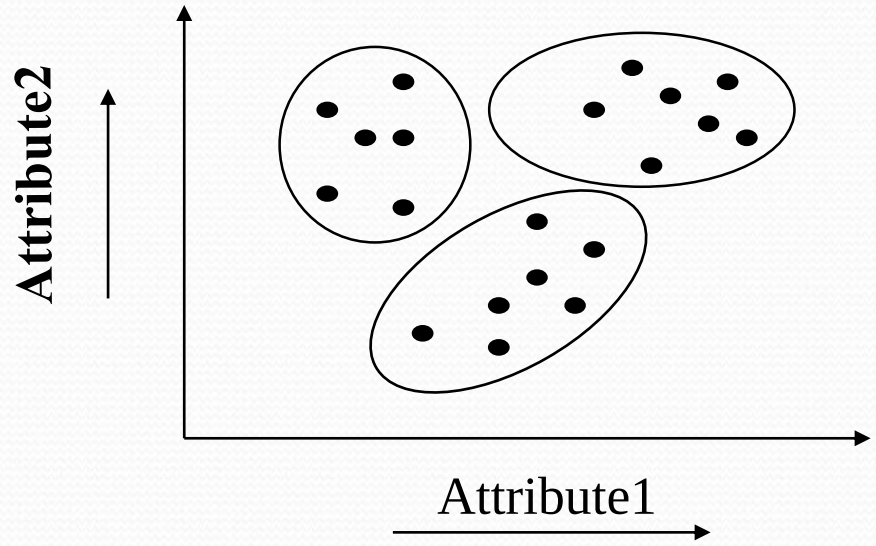
Rough Fuzzy Clustering

Outline of chapter 3

- Clustering algorithms (c-means & rough C-means)
- Clustering validity indices
- Formulation in rough framework
- Concept of collaborating clustering
- Experimental results
- Conclusion & future work
- References

Clustering

The objects within each group should be more similar to each other than to objects in any other group.



Clustering algorithms

- C-means
- Rough c-means

C-Means Algorithm

- Assign initial means m_i.
- Assign each X_k to the cluster U_i for the closest mean.
- Compute new mean for each cluster using

$$m_i = \frac{\sum_{X_k \in U_i} X_k}{|c_i|}$$

where $|c_i|$ is the number of objects in cluster U_i.

- Iterate until convergence.

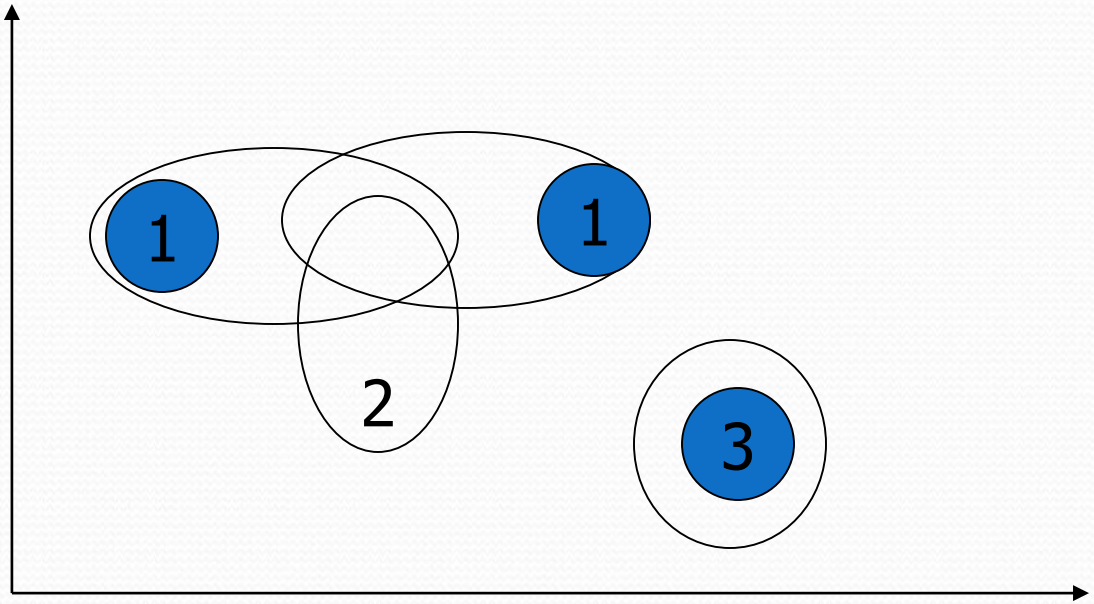
Rough Clustering

Clusters having objects/patterns

1). both in its lower and upper approximation.

2). Only in upper approximation.

3). Only in lower approximation.



Rough k-means algorithm

- Calculate new means as:

$$m_i = \begin{cases} w_{low} \dfrac{\sum_{X_k \in \underline{BU}_i} X_k}{|\underline{BU}_i|} + w_{up} \dfrac{\sum_{X_k \in (\overline{BU}_i - \underline{BU}_i)} X_k}{|\overline{BU}_i - \underline{BU}_i|}, & if\ \underline{BU}_i \neq \phi \wedge \overline{BU}_i - \underline{BU}_i \neq \phi \\ \dfrac{\sum_{X_k \in (\overline{BU}_i - \underline{BU}_i)} X_k}{|\overline{BU}_i - \underline{BU}_i|}, & if\ \underline{BU}_i = \phi \wedge \overline{BU}_i - \underline{BU}_i \neq \phi \\ \dfrac{\sum_{X_k \in \underline{BU}_i} X_k}{|\underline{BU}_i|}, & otherwise. \end{cases}$$

RCM algorithm

- Assign initial means m_i for the c clusters.
- Assign each X_k to lower approximation $|\underline{B}U_i|$ or upper approximation $|\overline{B}U_i|$, $|\overline{B}U_j|$ of cluster pairs U_i, U_j by computing the difference in its distance $d_{ik} - d_{jk}$ from cluster centroid pairs m_i and m_j

- **If** $d_{jk} - d_{ik}$ is less than some *threshold* **then**

  $X_k \in |\overline{B}U_i|$ and $X_k \in |\overline{B}U_j|$ and X_k cannot be a member of any lower approximation,

  else

  $$X_k \in |\underline{B}U_i|$$

- Compute new mean for each cluster U_i
- Iterate **until** convergence.

Cluster validity indices

- Davies-Bouldin Index

$$DB = \frac{1}{c}\sum_{k=1}^{c} \max_{l \neq k} \left\{ \frac{S(U_k) + S(U_l)}{d(U_k, U_l)} \right\}$$

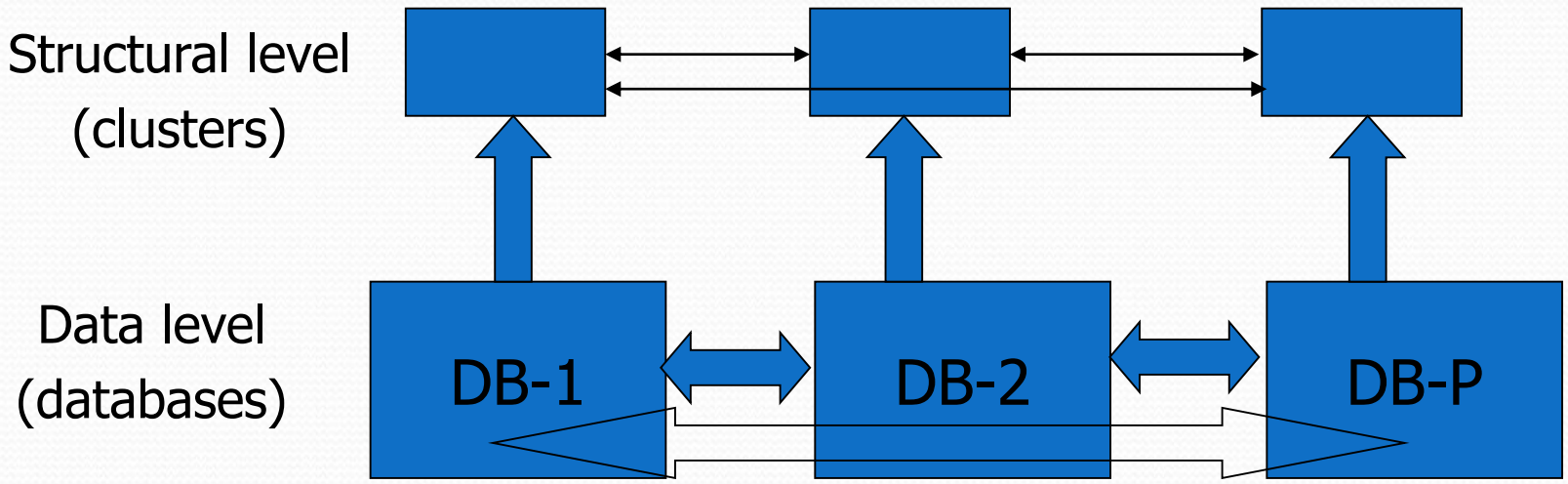
- Dunn index

$$D = \min_{j} \left\{ \min_{i \neq j} \left\{ \frac{d(U_i, U_j)}{\max_k S(U_k)} \right\} \right\}$$

DB for rough framework

$$S_r(U_i) = \begin{cases} w_{low} \dfrac{\sum_{X_k \in \underline{B}U_i} \| X_k - m_i \|^2}{|\underline{B}U_i|} + w_{up} \dfrac{\sum_{X_k \in \overline{B}U_i - \underline{B}U_i} \| X_k - m_i \|^2}{|\overline{B}U_i - \underline{B}U_i|}, & if\ \underline{B}U_i \neq \phi \wedge \overline{B}U_i - \underline{B}U_i \neq \phi \\ \dfrac{\sum_{X_k \in \overline{B}U_i - \underline{B}U_i} \| X_k - m_i \|^2}{|\overline{B}U_i - \underline{B}U_i|}, & if\ \underline{B}U_i = \phi \wedge \overline{B}U_i - \underline{B}U_i \neq \phi \\ \dfrac{\sum_{X_k \in \underline{B}U_i} \| X_k - m_i \|^2}{|\underline{B}U_i|} & otherwise. \end{cases}$$

Concept of collaboration



Collaborative RCM algorithm

1. Phase 1 (generation of rcm clusters for each module) $0.5 < w_{low} < 1$

2. Phase 2 (merging among modules) $0 < w_{low} < 0.5$

Phase 2:

A cluster U_j may be merged with an overlapping cluster U_i **if** $|\underline{B}U_i| \leq |BU_i - \underline{B}U_i|$ **and**

$\mathbf{m_j}$ is closest to $\mathbf{m_i}$ in the feature space with $(|\overline{B}U_i - \underline{B}U_i| - |\underline{B}U_i|)$ being the maximum among all overlapping clusters.

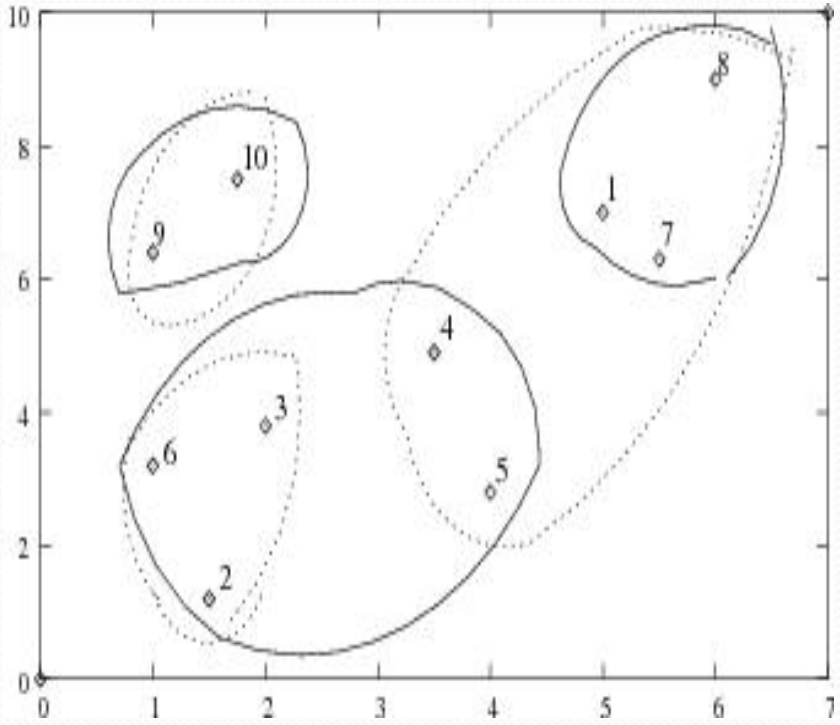
Collaborative rcm algorithm

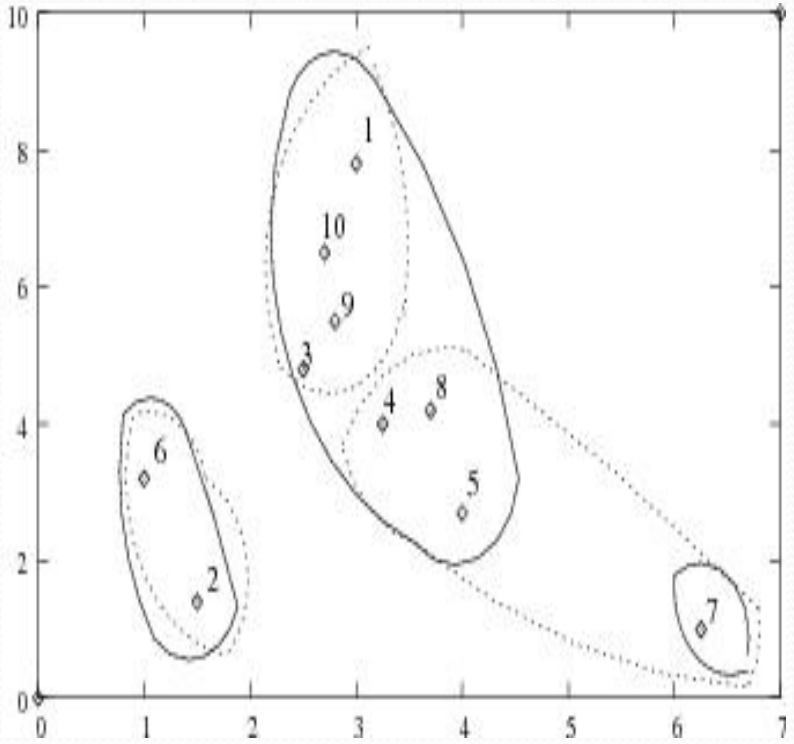
1) Split the large dataset into P modules.
2) For each module *p*=1, 2, ….., P **do**
   generate ***c*** *rcm clusters with* $0.5 < w_{low} < 1$
3) For each module *p* **do** collaboration.
   a. *accept* ***c*** *. (P-1) cluster prototypes from remaining (* P-*1) modules.*
   b. *assign each* X_k *to lower or upper approx. of the* C(=c.P) *collaborative rcm clusters with* $0 < w_{low} < 0.5$
   c. *merge overlapping clusters* ***while*** *merging condition hold.*
      i). compute new prototype for merged clusters U_i *and* U_j *as the mean of* m_i *and* m_j.
      ii). reduce C *by one.*
      iii). reassign each point X_k *to lower or upper approx of the C collaborative rcm clusters.*
      iv). Compute Davies-Bouldin and Dunn index.

Results

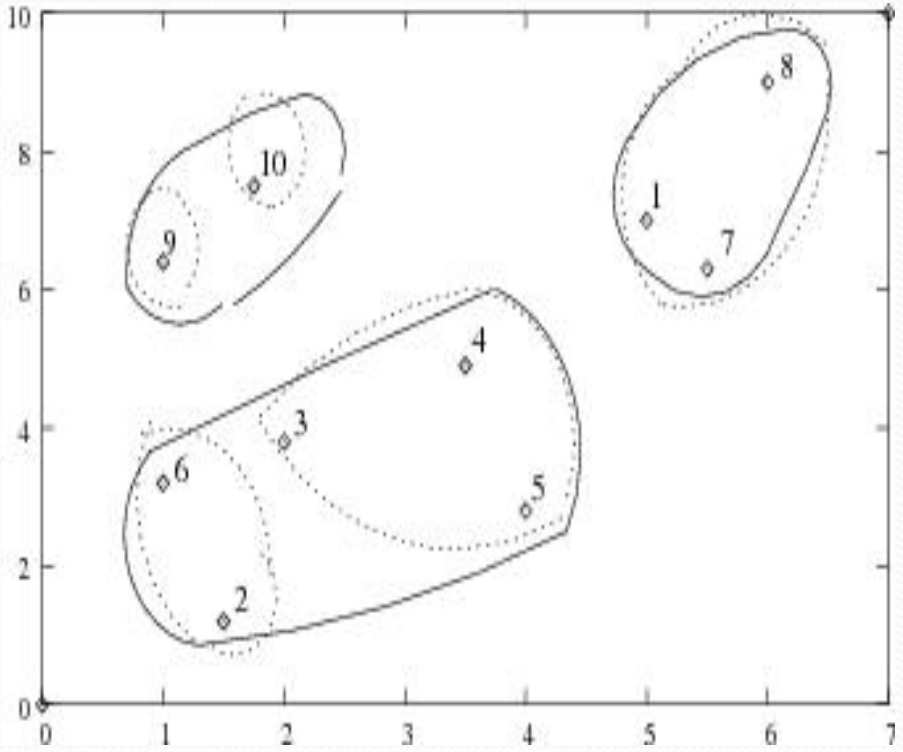
Module	Before Collaboration			After Collaboration		
	Prototypes	Samples in lower approx.	**DB**	Prototypes	Samples in lower approx.	**DB**
A	(3.35,6.42) (6.0,9.0) (2.13,2.75)	1, 4, 7, 9, 10 8 2, 3, 5, 6	0.79	(1.75,7.5) (5.5,7.43) (3.17,3.83) (1.0,4.63) (1.25,2.3)	10 1, 7, 8 3, 4, 5 9 2, 6	0.73
B	(6.25,1.0) (3.14,5.07) (1.25,2.3)	7 1, 3, 4, 5, 8, 9, 10 2, 6	0.57	(2.85,7.15) (4.0,2.7) (6.25,1.0) (3.06,4.63) (1.25,2.3)	1, 10 5 7 3, 4, 8, 9 2, 6	0.44

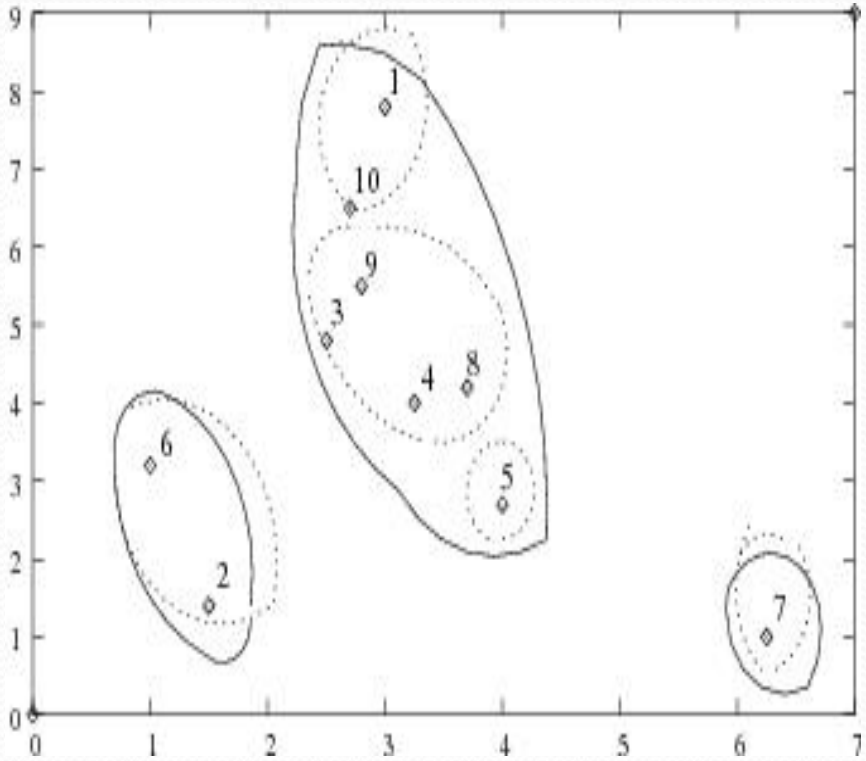
Collaboration for synthetic data A and B using Fuzzy clustering



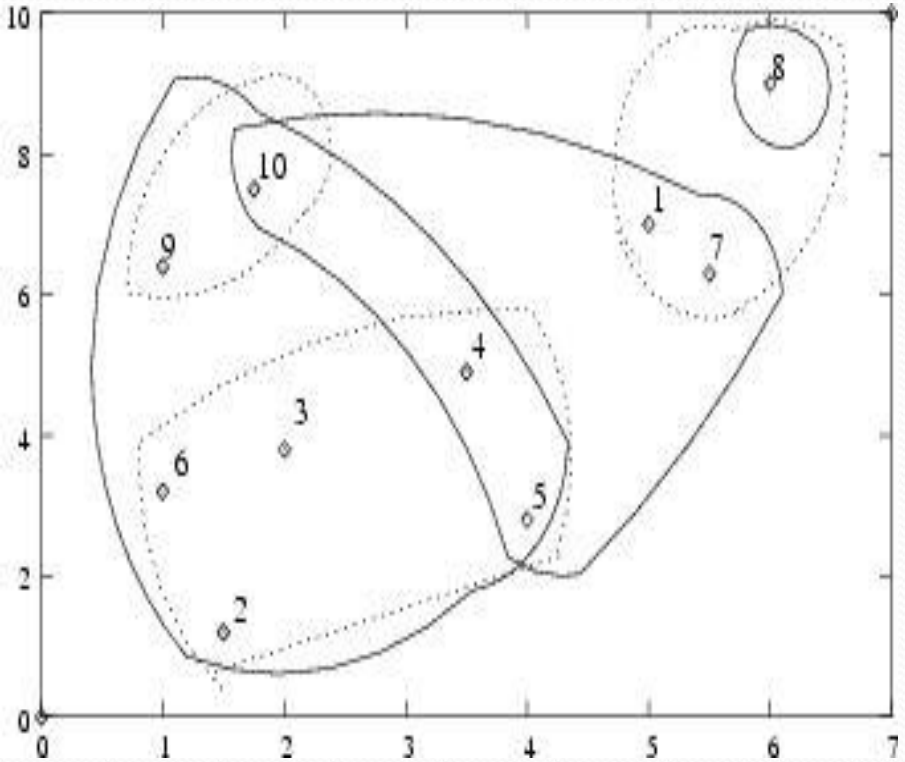


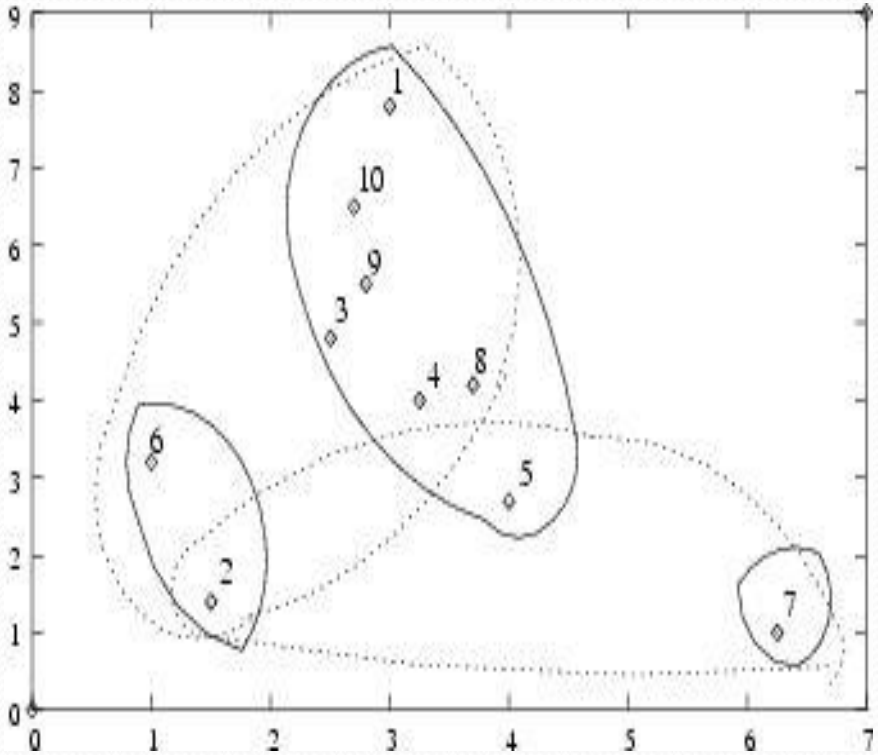
Collaboration for synthetic data A and B using Rough clustering





Collaboration for synthetic data A and B using Rough-Fuzzy clustering





Conclusion and future work

- Handling of heterogeneous data and/or cardinality of the data subsets.
- Various applications.