ELSEVIER

BioSystems 57 (2000) 37–48

BioSystems

www.elsevier.com/locate/biosystems

Minimizing stochastic complexity using local search and GLA with applications to classification of bacteria

P. Fränti [c], H.G. Gyllenberg [e], M. Gyllenberg [a,b], J. Kivijärvi [a], T. Koski [a,d], T. Lund [a,*], O. Nevalainen [a,b]

[a] *Department of Mathematical Sciences, University of Turku, FIN-20014 Turku, Finland*
[b] *Turku Center for Computer Science (TUCS), University of Turku, FIN-20014 Turku, Finland*
[c] *Department of Computer Science, University of Joensuu, P.O. Box 111, FIN-80101 Joensuu, Finland*
[d] *Department of Mathematics, Royal Institute of Technology, 10044 Stockholm, Sweden*
[e] *Institute of Biotechnology, University of Helsinki, 00014 Helsinki, Finland*

Received 2 September 1999; received in revised form 13 April 2000; accepted 17 April 2000

Abstract

In this paper, we compare the performance of two iterative clustering methods when applied to an extensive data set describing strains of the bacterial family Enterobacteriaceae. In both methods, the classification (i.e. the number of classes and the partitioning) is determined by minimizing stochastic complexity. The first method performs the minimization by repeated application of the generalized Lloyd algorithm (GLA). The second method uses an optimization technique known as local search (LS). The method modifies the current solution by making global changes to the class structure and it, then, performs local fine-tuning to find a local optimum. It is observed that if we fix the number of classes, the LS finds a classification with a lower stochastic complexity value than GLA. In addition, the variance of the solutions is much smaller for the LS due to its more systematic method of searching. Overall, the two algorithms produce similar classifications but they merge certain natural classes with microbiological relevance in different ways.

Keywords: Classification; Stochastic complexity; GLA; Local search; Numerical taxonomy

1. Introduction

Minimization of stochastic complexity (SC) (Rissanen, 1989) has proven to be an efficient method in numerical taxonomy, that is, in the classification and identification of bacteria based on phenetic features (Gyllenberg et al., 1997a, 1998, 2000). Stochastic complexity is an extension of Shannon's idea of information. Whereas in Shannon's case there is only one completely known probability model, stochastic complexity is computed with the help of a model class with unknown parameters. Stochastic complexity is taken to represent the information in a sequence of data with regard to the model class. In our adaptation of this notion to classification prob-

* Corresponding author.

0303-2647/00/$ - see front matter
PII: S0303-2647(00)00085-X

lems, we presuppose one of the standard statistical model classes in classification studies, the class of finite mixtures of multivariate Bernoulli distribution (Bock, 1996). Gyllenberg et al. (1997a, 2000) classified a large set of strains of Enterobacteriaceae by this method. They compared the classification found by minimization of SC to a well-established classification of the same material (Farmer et al., 1985). The study revealed similarity as well as some important differences between the two classifications.

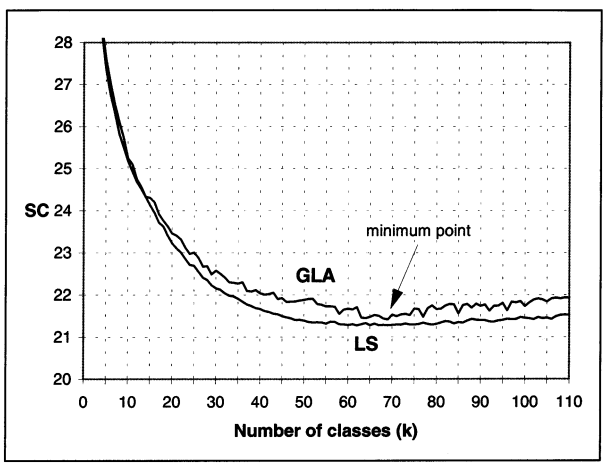

Fig. 1. SC curve as function of k produced by the LS and the GLA.

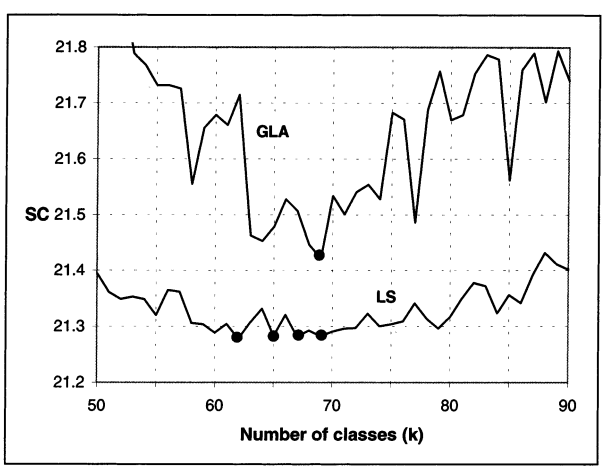

Fig. 2. SC curve from the interesting range. The black dots point the candidates for being minimum points within the range.

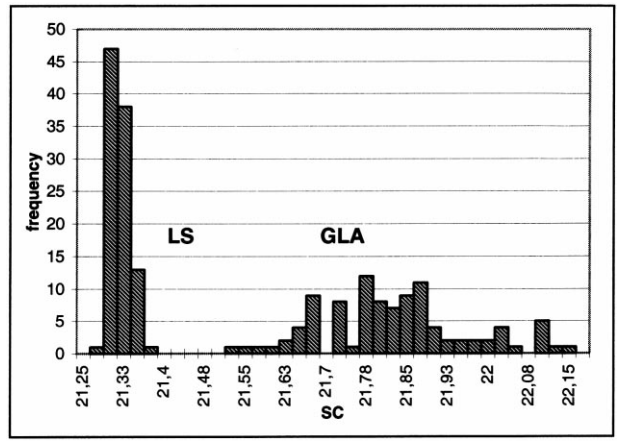

Fig. 3. Histograms of the SC-values produced by the two algorithms when 100 trials were performed.

In the present study, we consider the role of the clustering algorithm in numerical taxonomy. In particular, we want to answer two questions in this context. First, we compare the performance of the two clustering clustering algorithms measured by the best values of the cost function. Second, it is by no means evident that different algorithms would produce similar results. Even if the values of the cost function were similar, it would not imply that the classifications were the same. This is because the algorithms may converge to different local minima with the same value. The cost function, on the other hand, guides the clustering algorithm to find certain types of solution.

We use the classification of 5313 strains of bacteria belonging to the Enterobacteriaceae family as a case study. In particular, we discuss two efficient clustering algorithms, the generalized lloyd algorithm (GLA) (Linde et al., 1980); and the local search (LS) (Fränti et al., 1998; Fränti and Kivijärvi, 2000). The GLA has been applied to the Enterobacteriaceae data by Gyllenberg et al. (1997a), where a classification into 69 classes was obtained.

Local search (LS) is one alternative amongst effective clustering algorithms based on optimization techniques. It has originally been applied to the construction of the codebook in vector quantization. In this context, it has turned out to be

Table 1
SCENTE vs. LSENTE concordance matrix, labels are class numbers (LS on the rows)[a]

k	1	2	3	4	5	6	61	7	8	9	10	21	11	12	69	13	33	14	15	16	17	18	19	
1	250				4							1							7					265
3		219											1			2								222
36		22												2							2			52
5			206																					206
56			20																					27
2	1			224		1						3	12			2			11					258
4					207							6							4					217
15						98	8																	115
31						61	4																	66
9								159																159
8							3		156															160
10										110												10		147
47										38														38
7											146	40					2							188
14													124			2			2					128
12		1												134	1									136
6				4								40	1			88	60			1				202
13		1											3	4		4	2	113	5		1			135
11	13				1							3				38			81	2				139
17																				106				106
16																					106			106
19																						98		98
20																							97	97
32																								62
50																								36
21																								93
22																								83
18																								101
23																								81
28																								75
24																								78
25																								77
29																								73
44																								45
27			17																					75
69																								1
30																								68
34																								59
26																								76
41																								48
35																								54
33																								62
37																								51
48																								37
43												1							1					45
39																	1							49
40																							1	49
42																								46
45																								44
53																								33
61																								19
62																								19
46																								39
55																								29
67																								12
52																1								34
49																								36
51																								35
54																								30
38																								50
57																								26
64																								13
63																								16
59																								19
60																								19
66																								13
65																								13
58		4										1				1								20
68																								3
s	264	247	243	228	212	160	15	159	156	148	146	95	141	140	1	138	65	113	111	109	109	108	98	

Table 1 (*Continued*)

k	20	22	23	24	25	26	27	28	29	30	31	32	34	35	54	36	37	38	39	40	41	42	43	
1						1																1		265
3																								222
36					2										9	6			2	2				52
5																								206
56																								27
2																					4			258
4																								217
15						4							5											115
31						1																		66
9																								159
8											1													160
10		1								26														147
47																								38
7																								188
14																								128
12																								136
6			6																		1	1		202
13																					2			135
11																								139
17																								106
16																								106
19																								98
20																								97
32	62																							62
50	36																							36
21		93																						93
22			83																					83
18				88																				101
23					81																			81
28						75																		75
24							78																	78
25								77																77
29				1					72															73
44										44														45
27											49							9						75
69											1													1
30												68												68
34													59											59
26														57	19									76
41																48								48
35																	54							54
33											17							45						62
37																			51					51
48																				37				37
43																					43			45
39																						47		49
40															1								47	49
42																								46
45																								44
53																								33
61																								19
62																								19
46																	1							39
55																								29
67																								12
52																								34
49																								36
51															1									35
54																								30
38																				13				50
57																								26
64																								13
63																								16
59																								19
60											1													19
66																								13
65																								13
58			1								1					1								20
68																								3
s	98	94	90	89	83	81	78	77	72	70	70	68	64	57	30	55	55	54	53	52	50	49	47	

Table 1 (*Continued*)

k	44	45	46	47	48	49	50	51	66	52	53	55	65	56	57	58	59	60	62	63	64	67	68	
1					1																			265
3																								222
36																5								52
5																								206
56			7																					27
2																								258
4																								217
15																								115
31																								66
9																								159
8																								160
10																								147
47																								38
7																								188
14																								128
12																								136
6																								202
13																								135
11							1																	139
17																								106
16																								106
19																								98
20																								97
32																								62
50																								36
21																								93
22																								83
18															13									101
23																								81
28																								75
24																								78
25																								77
29																								73
44									1															45
27																								75
69																								1
30																								68
34																								59
26																								76
41																								48
35																								54
33																								62
37																								51
48																								37
43																								45
39												1												49
40																								49
42	46																							46
45		44																						44
53			33																					33
61				19																				19
62				19																				19
46					38																			39
55						29																		29
67						9				3														12
52							33																	34
49								35	1															36
51				1					1	32														35
54											30													30
38												27	10											50
57														26										26
64															13									13
63																16								16
59																	19							19
60																		18						19
66																			13					13
65																		1		12				13
58							1														10			20
68																						2	1	3
s	46	44	40	39	39	38	35	35	3	35	30	28	10	26	26	21	19	19	13	12	10	2	1	

[a] The values represent the number of the vectors appearing in both the classifications in the particular class.

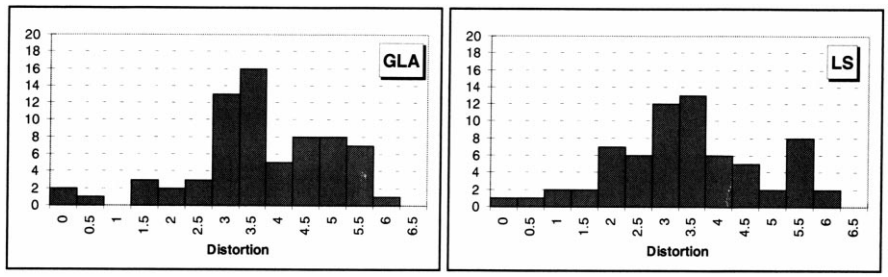

Fig. 4. Frequency histograms of the distortion values for the GLA- and LS-classifications. Distortion intervals (length of 0.5) are on x-axis and counts on y-axis.

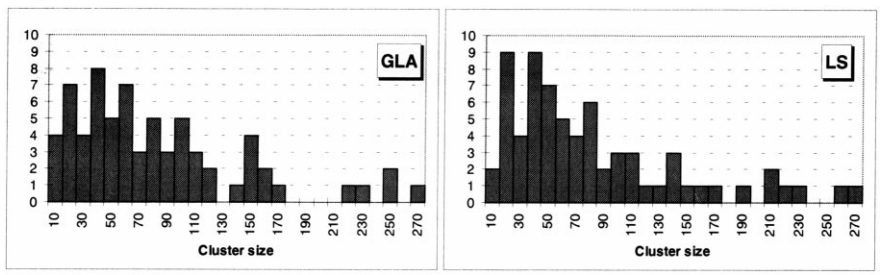

Fig. 5. Cluster size histograms of the GLA- and LS-classifications. Size intervals (length of 10) are on x-axis and counts on y-axis.

very competitive finding extremely effective codebooks in reasonable time (Fränti and Kivijärvi, 2000). In vector quantization, the LS has been applied using the mean square error as the cost function. In the present paper, we give necessary modifications for the method in order to apply stochastic complexity as the cost function in the LS. Other well known methods that could be for the clustering by SC are for example genetic algorithms (Fränti et al., 1997) and simulated annealing (Zeger and Gersho, 1989) but the simplicity of the LS makes it more suitable with the SC.

2. The classification problem

Let $S=\{\bar{x}^{(l)}|l=1, 2, \ldots, t\}$ be a set of t elements (feature vectors) of the form $\bar{x}^{(l)}=(x_1^{(l)}, x_2^{(l)}, \ldots, x_d^{(l)})$; $x_i^{(l)}\in\{0, 1\}$ for all $l\in\{1, 2, \ldots, t\}$, $i\in\{1, 2, \ldots, d\}$, d being the dimension of the vectors. Our task is to determine a classification of S into k classes so that the cost of the classification is minimal. We must consider three sub problems when generating a classification,

1. selecting a measure for the cost of the classification;
2. determining the number of classes used in the classification; and
3. selecting a suitable clustering algorithm.

These sub problems are interrelated but there are still many degrees of freedom in each selection. The choice of the number of classes has been discussed in (Gyllenberg et al., 1997b). In this paper, our main interest is to understand how the clustering algorithm itself affects the outcome.

Table 2
Distance between the classifications according to Eq. (5)

	CFARM	SCENTE	LSENTE
CFARM	*		
SCENTE	1441.0	*	
LSENTE	1388.5	565.0	*

Table 3
Most important differences in microbiological context appearing in Table 2

Class in SCENTE	Class in LSENTE	Nomenspecies or genus	Actions in LSENTE
1	1	*E. coli*	16 *E. coli* strains moved to class 11
2	3+36	*Shigella*	~20 *S. dysenteriae* separated to class 36
3	5	*Klebsiella*	~20 *E. aerogenes* separated to class 56 ~15 *K. pneumoniae* strains moved to class 27
6+61	15	*C. freundii*	~60 strains separated to 31
9	10+47	*E. cloacae*	~40 strains separated to 47
13+21+33	6	*E. coli*	~40 strains of SCENTE(21) moved to class 7 ~40 strains of SCENTE(33) moved to class 11
15	11	*E. coli*	10 strains separated to class 2
18	60	*Enterobacter*	~10 other than *E. taylorae* strains moved to class 10
20	32+50	*Providencia*	~35 *P. rustigianii* strains separated to class 50
30	44	*Enterobacter*	~25 *E. cloacae* strains moved to class 10
31	27	*Klebsiella*	~15 *K. rhinoscleromatis* strains moved to class 33
35+54	54	*E. americana*	~10 *E. americana* strains of SCENTE(54) moved to class 36
38	33	*Klebsiella*	~10 strains moved to class 27
40	48	*Salmonella*	~10 *S. enteritidis* moved to class 38
47	61+62	*Serratia*	~20 *S. odorifera* strains separated to class 62, other strains separated to class 61
55+65	38	*Salmonella*	
57	64	*Hafnia*	~10 *Hafnia* strains moved to class 18
		+ *Koserella*	
67+68	68	Trash classes	
69		Trash class	Single strain joined to class 12

2.1. Stochastic complexity

We describe mathematically a classification of strains with d binary (0 or 1) features into k classes by the numbers

$$\lambda_j;\ j=1,\ \ldots,\ k;\ \lambda_j \geq 0;\ \sum_{j=1}^{k} \lambda_j = 1$$

and

$$\theta_{ij};\ i=1,\ \ldots,\ d;\ j=1,\ \ldots,\ k (0 \leq \theta_{ij} \leq 1),$$

where λ_j is the relative frequency of strains in the jth class and θ_{ij} is the relative frequency of 1 s in the ith position in the jth class. The centroid of the jth class is the vector $(\theta_{1j},\ \theta_{2j},\ \ldots,\ \theta_{dj})$. The distribution of feature vectors $\mathbf{x} = (x_1,\ x_2,\ \ldots,\ x_d)$, $(x_i = 1$ or $0)$ of strains in class j is given by

$$p_j(\mathbf{x}) = \prod_{i=1}^{d} \theta_{ij}^{x_i} (1-\theta_{ij})^{1-x_i} \quad (1)$$

(Dybowski & Franklin, 1968; Willcox et al., 1980)

As a statistical model of the classification we, therefore, choose the distribution

$$p(\mathbf{x}) = \sum_{j=1}^{k} \lambda_j p_j(\mathbf{x}) \quad (2)$$

with the numbers k, λ_j and θ_{ij} being the parameters of the model. We emphasize that this statistical representation is simply a mathematically convenient way of defining the classification model and it does not imply any randomness in the data.

It was shown by Gyllenberg et al. (1997b) that the stochastic complexity SC of a set of t strains with respect to the above model is

$$\mathrm{SC} = \log_2\left(\frac{t!}{t_1!\ \ldots\ t_k!}\right) + \log_2\left(\frac{t+k-1}{t}\right) + \sum_{j=1}^{k}\sum_{i=1}^{d} \log_2\left(\frac{(t_j+1)!}{t_{ij}!(t_j-t_{ij})!}\right) \quad (3)$$

where t_j is the number of strains in class j and t_{ij}

is the number of strains in class j with ith feature equal to one. The two first terms in (Eq. (3)) describe the complexity of the classification and the third term, the complexity of the strains with respect to the classification.

To classify the feature vectors by minimizing stochastic complexity, we apply the Shannon-codelength instead of the L_2-distance that was originally used by Linde et al. (1980) (see also Gersho and Gray, 1992)

$$\mathrm{CL}(\mathbf{x},C_j) = -\sum_{i=1}^{d}\left((1-x_{ji})\log_2\left(1-\frac{t_{ji}}{t_j}\right)+x_{ji}\log_2\frac{t_{ji}}{t_j}\right) - \log_2\frac{t_j}{t}, \quad (4)$$

The vector $(t_{1j}/t_j, \ldots t_{dj}/t_j)$ is the centroid of the class θ_j and the number t_j/t is the estimate of the parameter λ_j and is called weight of the class. Minimization of expression Eq. (4) clearly minimizes the expression Eq. (3).

2.2. The generalized Lloyd algorithm

Given the number k of classes, the GLA for minimizing SC works as follows,

Step 1. Draw k initial centroids randomly from the set of input vectors S.
REPEAT TEN TIMES.
  Step 2.1. Assign each input vector to the closest cluster centroid with L_2-distance.
  Step 2.2. Calculate new cluster centroids.
REPEAT.
  Step 3.1. Assign each input vector to the closest cluster centroid with Formula 4.
  Step 3.2. Calculate new cluster centroids and class weights, and evaluate the distortion with Eq. (3).
UNTIL no more improvement in distortion value.

The algorithm (also known as LBG or k-means) (McQueen, 1967; Linde et al., 1980) starts with an initial solution, which is iteratively improved using the two step procedure. In the first step (step 2.1), the data objects are partitioned into a set of k clusters by mapping each object to the closest cluster centroid. In the second step (step 2.2), the centroids and the weights of the clusters are updated. We represent the centroids by floating point numbers (i.e. the components of a centroid are the relative frequencies of one-bits). The process is first repeated ten times using the L_2-distance for the following reasons, (i) an initial solution is needed for calculating the cluster weights; (ii) the CL-distance is highly dependent of the quality of the initial solution; and, therefore, we need more than one iteration with L_2; (iii) the use of the L_2-distance reduces the computational load of the algorithm. In the next steps (steps 3.1 and 3.2), the GLA is performed using the CL-distance. This process is repeated as long as improvement is achieved in the cost function value (SC).

Another problem is connected with the so-called orphaned centroids (also known as the empty cell problem). Orphaned centroids have no vectors assigned to them in the steps 2.2 and 3.2 of the algorithm. The orphaned centroids must be eliminated, otherwise the number of non-empty classes would be less than the prescribed k. Whenever an orphaned centroid occurs, it is replaced by splitting an existing class having the greatest distortion. The centroid of the new class is taken as the vector with the greatest distance to the original centroid of the split class, as proposed by Kaukoranta et al. (1996). The GLA iterations then continue with the modified set of centroids.

A significant benefit of the GLA is its fast operation. On the other hand, the main drawback of the GLA is that it is only a simple descent method and, therefore, converges at the first local minimum. The result depends strongly on the quality of the initial solution. An easy way to improve the method is to repeat the algorithm for several different initial solutions and select the one with minimal value of the cost function (Eq. (3)). In the following, we use this approach.

2.3. The local search algorithm

Local search (LS) uses a different approach to avoid the problem of local minima. Next we give an improved version of the LS algorithm by Fränti et al. (1998) which is simplified for gaining speed (Fränti and Kivijärvi, 2000), and modified

to use SC as the cost function instead of the MSE. The algorithm forms an initial classification and iteratively improves it by applying a randomizing function, and the GLA. Given the number of classes k, the control flow of the LS method is as follows.

Step 1. Draw k initial centroids (from the input data) randomly.
Step 2. Assign each input vector to the closest cluster centroid.
ITERATE T times
  Step 3.1. Replace a randomly chosen class.
  Step 3.2. Re-assign each input vector according to the changed class.
  Step 3.3. Perform two iterations of the GLA using L_2 distance.
  Step 3.4. Evaluate the cost function value Eq. (3).
  Step 3.5. If the modifications improved the classification, accept the new classification. Otherwise restore the previous classification.
Step 4. Perform the GLA using the CL-criterion Eq. (4) for the final solution.

The initial classification is generated as in the GLA, i.e. the algorithm selects items randomly as the class representatives (HMO, hypothetical mean organism). The rest of the strains are assigned to the closest class representative according to the L_2-distance. A new trial classification is then generated by making random modifications to the current one, a randomly chosen class is made obsolete and a new one is created by selecting any sample strain as the new HMO. The resulting classification is improved by the GLA. This generates changes in the class representatives of the trial classification. The trial classification is accepted if it improves the cost function value, which is the SC.

The advantage of the LS over the GLA is that it makes global changes to the clustering structure, and at the same time performs local fine tuning towards a local optimum. In our practical tests, we let the algorithm run for 5000 iterations in total, and two iterations of the GLA is applied for each trial solution. This gives us a good balance between the global changes in the clustering structure and the local modifications in the classification. During the search (step 3), we do not use the Shannon-codelength (CL) when assigning the vectors to the clusters because it is problematic with the chosen randomization function. It is also much slower and does not enhance the performance according to our tests. The final solution, however, is fine-tuned by the GLA using the CL-criterion. The steps 1–3 of our LS-algorithm, thus, form a preprocessing phase for the SC-clustering.

2.4. Number of classes

The optimum number of classes is, by definition, the value k^* at which the overall minimum of SC is attained. A straightforward solution is, therefore, to apply the GLA and the LS for every reasonable number of classes (1–110). From the results of each fixed k, we take the one with minimal stochastic complexity. The LS was performed only once with the iteration count 5000 whereas the GLA was repeated 50–200 times.

2.5. Difference between the classifications

There are basically two ways for measuring the difference between classifications, either we measure the smallest number of modifications (usually set operations) that are needed to make the two classifications equal, or we count some well defined differences between them (Day, 1981). Modifications and differences can be defined in various ways and they should be chosen carefully to suit the application. We illustrate the difference between two given classifications denoted by $P' = \{C'_1, \ldots, C'_{k'}\}$ and $P'' = \{C''_1, \ldots, C''_{k''}\}$. The Concordance matrix $\mathbf{M}$ is a k' by k'' matrix with entry $\mathbf{M}_{ij}$ defined as the number of elements in $C'_i \cap C''_j$. The distance between two classifications can now be calculated from the concordance matrix by

$$D = \frac{1}{2}\left[\sum_{i=1}^{k'}\left(\sum_{j=1}^{k''} M_{ij} - \max_{j=1}^{k''} M_{ij}\right) + \sum_{j=1}^{k''}\left(\sum_{i=1}^{k'} M_{ij} - \max_{i=1}^{k'} M_{ij}\right)\right] \quad (5)$$

The distance D can be interpreted as the average number of differently classified vectors when comparing P' to P'' and vice versa. The third

alternative is to trust a domain expert, here to a microbiologist, and let him judge the classifications P' and P'' against the best state of art knowledge of the taxonomy of the strains. This evaluation gives us insight of the usability of our algorithms and proximity functions.

3. Application to classification of Enterobacteriaceae

The data set (called ENTE) consists of $t = 5313$ strains of bacteria belonging to the Enterobacteriaceae family. The strains have been isolated during the years 1950–1988 and identified by CDC in Atlanta, Georgia (Farmer et al., 1985). The material consists of 104 nomenspecies. We refer to nomenspecies classification as CFARM ($SC/t = 23.12$). Each strain is characterized by a binary vector of $d = 47$ bits representing outcomes of biochemical tests. For a detailed description of the material, see Farmer et al. (1985).

4. Results

Gyllenberg et al. (1997a) calculated the SC minimizing classification of ENTE using GLA and obtained a classification into 69 classes with the SC-value 21.42. This classification is called SCENTE. At least 20 repetitions were run for each k, the best candidates were inspected 50–200 times. Thus, the total running time for SC-minimization was very long, a few days with an efficient computer, although the partitioning of one candidate is a quick operation (few minutes). The LS is about 100–200 times slower than the GLA for a single classification. The LS, on the other hand, is rather independent of the initialization and, therefore, only a single run is enough whereas the GLA, as mentioned above, had to be repeated 50–200 times. This balances the computational differences of the methods.

As the first step, we classified the material with the LS for all k-values from 1 to 110 and compared the SC-values with the ones obtained previously by the GLA. They both give the minimum SC-values approximately in the same range (see Fig. 1). The curve for the LS is very flat for k-values in the range $k = 60$–72 (see Fig. 2). The LS found SC values from 21.28 to 21.33 with the mean 21.30 and S.D. 0.016 for these k-values. This means that there are several values of k which produce almost equally good classification in the sense of minimizing SC. The corresponding SC-values for the GLA vary from 21.42 to 21.72 with the mean 21.53 and S.D. 0.094.

We studied the methods more closely on the previously found minimum point ($k = 69$). The results obtained by the LS have less variation and are consistently better than the results of GLA (Fig. 3). For example, the LS always produced classification with smaller SC-value than the best SC-value obtained by the GLA. The corresponding SC-values for the LS vary from 21.27 to 21.35, and for the GLA from 21.51 to 22.18.

As a second step, we compared the two classifications by producing the class concordance matrix for them. We focus on differences between the LS and GLA classifications, see Table 1. The rows of the matrix stand for the classes of the LS-solution and the columns for the classes of the GLA-solution. Note that the class indices are selected by the size of the classes in descending order. If the two classifications resemble each other there is a good concordance between some corresponding pairs of classes from LS and GLA. To increase the level of illustration, we have additionally arranged the rows and columns of the matrix such that the greatest matrix-elements appear on the diagonal. The total number of elements of a particular class appear as row/column sum.

Table 1 shows that the class with index 6 of LSENTE consists of 303 strains which are scattered over classes 13 (88 strains), 33 (60), 21 (40), 23 (6), 4 (4), 11 (1), 41 (1), 42 (2) and 16 (1) of SCENTE. The joining of the three classes 13, 33 and 21 in other classification is natural because these represent the species *E. coli*. Similar phenomena can be found for example in *E. americana* strains with SCENTE classes 35 and 54. These two classes are joined in LSENTE into class 26. The phenomenon of splitting and merging of classes appears also in the other direction. The GLA class 6 contains most of the citrobacters. In

the LSENTE classification, the *C. freundii* strains are scattered over two major classes (15, 31).

Fig. 4 shows the distribution of the distortion values (average Hamming distance from HMO) for the two classifications. The distortion values of both LSENTE and SCENTE resemble normal distribution. Fig. 5 demonstrates the class size distributions, first bar represents number of classes having size in range 1–10, and second 11–20 and so forth. There are some small classes in both classifications and they are likely due to some badly fitting vectors in data (i.e. trash).

As a third step, we calculated the distance D between the classifications and CFARM. The results of Table 2 show that the LSENTE and SCENTE classifications, whose stochastic complexities were smaller, are closer to each other than to CFARM. On the other hand, the LSENTE and SCENTE classifications are almost as far from CFARM.

As a fourth step, we compared the classifications made by the classification algorithm to the expert class (CFARM). Our data had scientific names (104 nomenspecies) and identification numbers attached to the vectors. This allows us to analyze how well the obtained classifications conform to the currently established classification of Enterobacteriaceae. Most of the data represent species of *E. coli*, hence we concentrate on this part.

The *E. coli* strains tend to divide into two groups. These correspond to the so-called active and inactive *E. coli* (Farmer et al., 1985). The LSENTE classes 6, 13, 14 represent inactive *E. coli* group and the classes 1, 2, 4, 7, 17, 39, 43 active group. Similarly, *E. coli* is divided into two remote groups of SCENTE classes 15, 21, 33, 13, 11 and 42, 41, 4, 10, 5, 16, 1. We noted before that LSENTE class 6 was scattered to many classes in SCENTE (mainly 13, 21, 33). It seems that LS has managed to classify the inactive group of *E. coli* more efficiently. On the other hand, the active *E.coli's* are classified similarly in both classifications. We have listed most microbiologically relevant differences in Table 3. We find these phenomena similar to the findings of Gyllenberg et al. (1998) Gyllenberg et al. (2000).

5. Conclusions

We have compared two clustering algorithms using a taxonomic problem in microbiology as a case study. One of the algorithms is the classical GLA algorithm whereas the other applies an efficient local search strategy to improve the classification. A general observation is that the classifications do not differ radically when evaluated by SC. The LS was somewhat more effective than GLA.

The comparison of the classifications reveals several interesting facts. A class may be split into two or more classes in the classifications obtained by GLA and LS. The splitting of the clusters is usually reasonable from a microbiological point of view.

In spite of the different strategies in the GLA and the LS, the distributions of the class sizes of the two classifications look similar. This is most likely due to the data used in this experiment. It turned out that the repeated use of GLA with random initial solutions can produce good classifications. Iterative optimization techniques like LS are, however, more reliable in the sense that they search more systematically for the global optimum. To demonstrate this, we calculated the histogram of SC values, which shows that results of LS are more biased to better SC-values. Even though the implementation of the LS and the GLA is a relatively easy task, we were confronted by some practical issues, which need special consideration. These include the case when a cluster consists of a single strain only. In this case, vectors are seldom mapped to this new cluster when the entropy distance (6) is used. The use of the L_2-distance solves this problem. The overall evaluation of the preference of the two classifiers favor the LS because the GLA is more dependent on setting of the initial centroids, and as shown above, the LS will more likely produce a good solution.

Acknowledgements

This work has been supported by the Academy of Finland, The Swedish Natural Sci-

ence Research Council (NFR) and by the Knut and Alice Wallenberg Foundation.

References

Bock, H.-H., 1996. Probability models and hypothesis testing in partitioning cluster analysis. In: Arabie, C.P., Hubert, L.J., De Soete, G. (Eds.), Clustering and Classification. World Scientific, Singapore, pp. 377–453.

Day, W.H.E., 1981. The complexity of computing matrix distances between partitions. Math. Soc. Sci. 1, 269–287.

Dybowski, W., Franklin, D.A., 1968. Conditional probability and identification of bacteria. J. Gen. Microbiol. 54, 215–229.

Farmer, J.J., Davis, B.R., Hickman-Brenner, F.W., McWhorter, A., Huntley-Carter, G.P., Asbury, M.A., Riddle, C., Wahten-Grady, H.G., Elias, C., Fanning, G.R., Steigerwalt, A.G., O'Hara, C.M., Morris, G.K., Smith, P.B., Brenner, D.J., 1985. Biochemical identification of new species and biogroups of Enterobacteriaceae isolated from clinical specimens. J. Clin. Microbiol. 21, 46–76.

Fränti, P., Kivijärvi, J., 2000. Randomized local search algorithm for the clustering problem. Pattern Analysis and Applications 3, in press.

Fränti, P., Kivijärvi, J., Kaukoranta, T., Nevalainen, O., 1997. Genetic algorithms for large scale clustering problems. Comput. J. 40 (9), 547–554.

Fränti, P., Kivijärvi, J., Nevalainen, O., 1998. Tabu search algorithm for codebook generation in vector quantization. Pattern Recognition 31 (8), 1139–1148.

Gersho, A., Gray, R.M., 1992. Vector Quantization and Signal Compression. Kluwer Academic Publishers, Dordrecht.

Gyllenberg, H.G., Gyllenberg, M., Koski, T., Lund, T., Schindler, J., Verlaan, M., 1997a. Classification of Enterobacteriaceae by minimization of stochastic complexity. Microbiology 143, 721–732.

Gyllenberg, M., Koski, T., Verlaan, M., 1997b. Classification of binary vectors by stochastic complexity. J. Multivariate Anal. 63, 47–72.

Gyllenberg, H.G., Gyllenberg, M., Koski, T., Lund, T., 1998. Stochastic complexity as a taxonomic tool. Comput. Methods Programs Biomed. 56, 11–22.

Gyllenberg, H.G., Gyllenberg, M., Koski, T., Lund, T., Schindler, J., 2000. Enterobacteriaceae taxonomy approached by stochastic complexity. Quant. Microbiol. 1, 157–170.

Kaukoranta, T., Fränti, P., Nevalainen, O., 1996. Reallocation of GLA codevectors for evading local minima. Electron. Lett. 32, 1563–1564.

Linde, Y., Buzo, A., Gray, R.M., 1980. An algorithm for vector quantizer design. IEEE Trans. Commun. 28, 84–95.

McQueen, J.B., 1967. Some methods of classification and analysis of multivariate observations. Proceedings of Fifth Berkeley Symposium of Mathematical Statistics Probability, 1, Berkeley, CA, pp. 281–296.

Rissanen, J., 1989. Stochastic Complexity in Statistical Inquiry. World Scientific, Singapore.

Willcox, W.R., Lapage, S.P., Holmes, B., 1980. A review of numerical methods in bacterial identification. Antonie Leeuwenhoek 46, 233–299.

Zeger, K., Gersho, A., 1989. Stochastic relaxation algorithm for improved vector quantiser design. Electron. Lett. 25, 896–898.