Discriminant Analysis on Dissimilarity Data: A New Fast Gaussian-like Algorithm

A. Guérin-Dugué, G. Celeux
INRIA-IS2, 655 Avenue de l'Europe, F-38330 Montbonnot cedex

Abstract Classifying objects according to their proximity is the fundamental task of pattern recognition and arises as a classification problem or discriminant analysis in experimental sciences. Here we consider a particular point of view on discriminant analysis from a dissimilarity data table. We develop a new approach, inspired from the Gaussian model in discriminant analysis, which defines a set a decision rules from simple statistics on the dissimilarity matrix between observations. This matrix can be only sparse dealing with huge databases. Numerical experiments on artificial and real data (proteins classification) show interesting behaviour compared to a KNN classifier, (i) equivalent error rate, (ii) dramatically lower CPU times and (iii) more robustness with sparse dissimilarity structure up to 40% of actual dissimilarity measures.

1 Introduction

Most of classification approaches concern situations where an observation is described by its coordinates in metric space. But, for many applications such vector description is not available, and only pairwise dissimilarity data are provided. Such applications are usual in psychology, biology, genetic, signal processing... As far as we know, only two approaches dealing with the classification problem in this context have been proposed. The first one is based on the "K Nearest Neighbors" (KNN) method [3] which is a rather slow method and non suited to non-spherical class shapes but efficient with non-connected classes. The second one transforms the problem to a metric one using Multidimensional Scaling techniques [4], [2]. But, this approach can introduce important distortion in the Euclidian representation of the observations and the estimation of the intrinsic dimension of the Euclidian space is a difficult open problem.

Our motivation is to propose alternative classification techniques from dissimilarity tables whose advantages are rapidity, data driven versatility and adaptation to incomplete dissimilarity data. All these features are discussed in the following.

The set of proposed decision rules starts from the simplest case which is equivalent to the linear discriminant analysis. A pseudo Euclidian distances is defined using averages estimated for each class ω_k from the dissimilarity matrix. Moreover, and this is one of the originalities of this proposal, non linearity is introduced by the way of the class variances on this same set of dissimilarities. This quantity takes into account the "shape" and the intrinsic dimension of the classes in a global way or in a local way. This leads to a quadratic-like classifier based on a pseudo Mahalanobis distance.

In the following, we present the justification of the proposed method, the decision rules, the practical implementation of the learning algorithms, and finally some experimental results.

2 Statistics on Distance Data

Let us consider a set $\mathcal{X}$of N objects, linked by pairwise distance values gathered in a $N \times N$ matrix $D = (d(i,j), i,j \in \mathcal{X})$. Acting as if the matrix D defines Euclidian distances between the objects, we define for $e \in \mathcal{X}$

$$\overline{d(e)^2} = \frac{1}{N} \sum_{i \in \mathcal{X}} d^2(e,i). \tag{1}$$

This quantity can be regarded as the inertia of $\mathcal{X}$ with respect to e. The pseudo-centre o of $\mathcal{X}$ is defined as

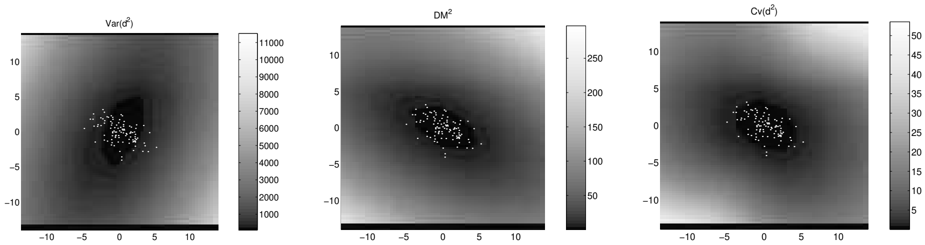
$$o = \text{argmin}_{e \in \mathcal{X}} \overline{d(e)^2} \tag{2}$$

and the inertia I of $\mathcal{X}$ is defined as $I = \overline{d(o)^2}$. Now, if D defines an Euclidian distance matrix, we have from Huygens theorem

$$\overline{d(e)^2} = d^2(o,e) + I, \tag{3}$$

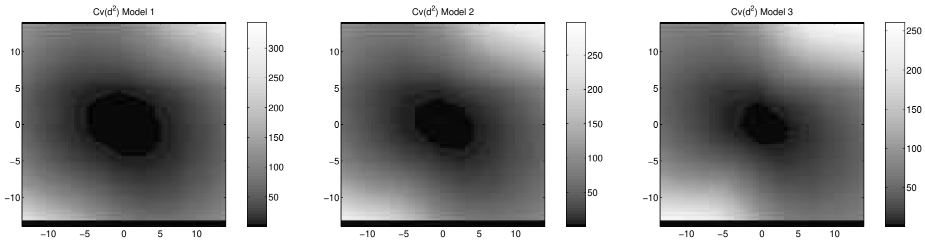
and, moreover, it can be seen that

$$I = \frac{1}{2N^2} \sum_{i,j \in \mathcal{X}} d^2(i,j). \tag{4}$$



(a) (b) (c)

Figure 1: Spatial evolution of (a) $\mathrm{var}(d(e)^2)$, (b) $D_M^2(E, O)$, (c) $\mathrm{Cv}(d(e)^2)$.



(a) (b) (c)

Figure 2: Spatial evolution of $\hat{D_M^2}$, according to (a) the model 1 $\alpha = 0.78$, (b) the model 2 $\beta = 5.56$, and (c) the model 3 $\alpha = 1.18$, $\beta = 21.64$.

Thus, in the Euclidian setting, there is no need to compute the pseudo-centre o, to get $d^2(o, e)$ for any object e. A first proposed decision rule (Section 3.1) is then naturally derived.

The empirical variance

$$\mathrm{var}(d(e)^2) = \frac{1}{N} \sum_{i \in \mathcal{X}} (d^2(e, i) - \overline{d(e)^2})^2 \tag{5}$$

is more complex, depending on high order $\mathcal{X}$ moments. Nevertheless, this quantity takes globally into account the "shape" and the intrinsic dimension of $\mathcal{X}$. Let us illustrate this behaviour on $\mathcal{X}$, a simple 2D Gaussian distribution (fig. 1).

For observations e lying in the direction of the main $\mathcal{X}$ orientation, $\mathrm{var}(d(e)^2)$ are greater than for observations lying in the opposite direction (fig. 1a). In order to take into account the "shape" of the set $\mathcal{X}$ like in the Mahalanobis distance $D_M(e, o)$ (fig. 1b), we use the variation coefficient (fig. 1c), defined as :

$$\mathrm{Cv}(d(e)^2) = \frac{(\overline{d(e)^2} - I)^2}{\mathrm{var}(d(e)^2)}. \tag{6}$$

The similar behaviour of these two quantities $D_M^2(e, o), \mathrm{Cv}(d(e)^2)$ can be refined by the following fitting equations. We have defined three fitting models (two with one parameter and one with two parameters) :

$$1: \quad \hat{D_M^2}(e, o) = \frac{(\overline{d(e)^2} - I)^2}{[\mathrm{var}(d(e)^2)]^\alpha}, \tag{7}$$

$$2: \quad \hat{D_M^2}(e, o) = \beta \cdot \frac{(\overline{d(e)^2} - I)^2}{\mathrm{var}(d(e)^2)}, \tag{8}$$

$$3: \quad \hat{D_M^2}(e, o) = \frac{\beta \cdot (\overline{d(e)^2} - I)^2}{[\mathrm{var}(d(e)^2)]^\alpha}. \tag{9}$$

Figure 2 illustrates the behaviour of the three models

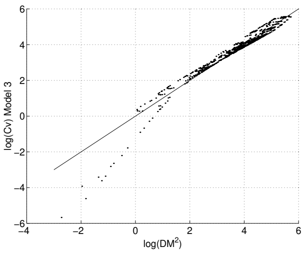


Figure 3: Fitting between the Mahalanobis distance and the variation coefficient the model 3.

on a 2D Gaussian distribution, and Figure 3 illustrates the fitting quality according to the model 3. The fitting parameters α and β are set by minimizing the mean squared error. For this example, the residual mean squared error is respectively 0.257, 0.116 and 0.081, for respectively the model 1, 2, and 3. Extensive simulations on Gaussian distributions from 2 to 10 dimensions and various "shapes" lead to the following remarks:

- Fitting with two parameters is better than fitting with one parameter.
- Models 2 and 3 are better than model 1.
- The parameters (α, β) depend strongly on the "shape" (covariance matrix) and the Euclidian dimension of the data.

Thus, to take into account a particular "shape", and the intrinsic dimension of a class, the proposed decision rules will use these Malahanobis-like estimators by these "modified" variation coefficients (Section 3.2). The procedure to estimate the learning parameters α and β is described in Section 4.

3 Decision rules

Two kinds of decision rules have been designed. The first one is based on the mean distances, and the second one on the variation coefficients. The justification of these decision rules comes from analogies with Gaussian classifiers assuming that the dissimilarity measures are in fact Euclidian distance measures. Otherwise, the decision rules are simply applied from statistics on disimilarity values (means, variances, variation coefficients), but the exact relationships with inertia and centres are no longer valid.

3.1 Decision rules based on the mean values

Considering (1), the simplest rule to classify a new object e is

$$\mathrm{class}(e) = \mathrm{argmin}_k(\overline{d_k(e)^2} - I_k), \tag{10}$$

where I_k is the pseudo-inertia of class k, and $\overline{d_k(e)^2}$ is the mean value of the dissimilarities (1) restricted to class k. Applied on Euclidian distance data, this rule is exactly equivalent to a linear classifier (fig. 4a). It can be enhanced by taking into account the volume of each class by the way of the pseudo-inertia I_k, such as (fig 4b) :

$$\mathrm{class}(e) = \mathrm{argmin}_k(\frac{\overline{d_k(e)^2} - I_k}{I_k}). \tag{11}$$

3.2 Decision rules based on the variation coefficients

The last refinement of the decision rule is to take into account the "shape" of each class k using the variation coefficient $\mathrm{Cv}(d_k(e)^2$ in the following way

$$\mathrm{class}(e) = \mathrm{argmin}_k(\mathrm{Cv}(d_k(e)^2). \tag{12}$$

The boundaries obtained with this rule (fig. 4c) are compared with those obtained with a simple quadratic classifier (fig. 4d). This rule uses the variation coefficients defined by (6) without any additional fitting parameter. We present in the next section a fast and optimal learning procedure to both estimate the fitting parameters (α_k, β_k), and classify the observations. This learning strategy allows to take into account the database structures. So, it is a more powerful implementation than the simple rule (12) which corresponds to $(\alpha_k = \beta_k = 1)$ for all k.

4 Learning Procedure

The learning procedure is explained for the two models with one parameter for which it is optimal. An other procedure for the model 3 with two parameters can be easily derived by nesting the previous ones. But this resulting procedure is only sub-optimal.

	Without adaptation	Global adaptation	Local adaptation
Model 1	$\alpha = 1$	$\alpha \neq 1$ and $\alpha_k = \alpha^*, \forall k$	α_k
Model 2	$\beta = 1$	$\beta = 1$	β_k

Table 1: Principle of the local and global adaptation

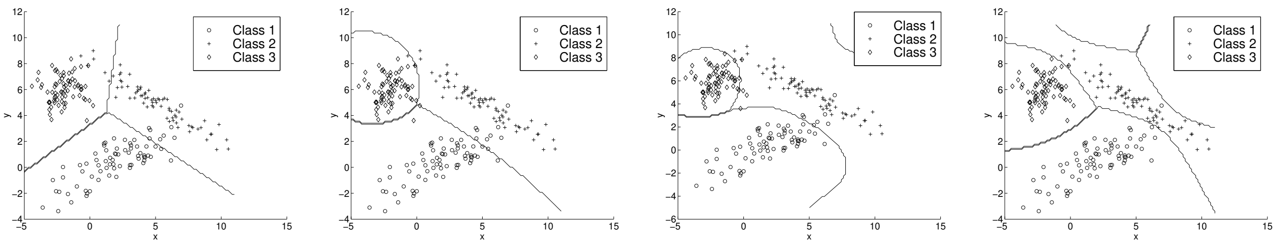
As usual, two cases are considered for the data-driven estimation of parameters α and β : a global estimation for all the classes and a local estimation for each class. Table 1 summarizes those different possibilities. The parameters are estimated relatively to each other, from a reference value, fixed to 1. In all the cases, the parameters optimize the cross-validated recognition rate. %

4.1 Global adaptation

Let us consider the model 1. Starting from $\alpha = 1$ for all the observations, α will be set to the value α^* maximizing the recognition rate estimated by cross-validation. For the model 2, since β is a proportionality factor on the variation coefficient, the global adaptation of its value does not make sense.

Let us notice $\omega_k(e)$, the true class k of an observation e belonging to the learning data set, and $cl_l(e)$, the class l selected from the decision rule.

The initial step is $\alpha(e) = 1$, whatever the observations. For each observation e, the class $cl_l(e)$ is selected according to the decision rule (12) with Cv estimated by (7). If the selected class (l) is not equal to the true class (k), the parameter $\alpha(e)$ must be modified to correct this misclassification, that is to ensure that



(a) (b) (c) (d)

Figure 4: Examples of boundaries obtained with (a) equation 10, (b) equation 11, (c) equation 12, (d) a quadratic classifier.

$$\mathrm{Cv}(d_l(e)^2) > \mathrm{Cv}(d_k(e)^2)$$

$$\log[\frac{(\overline{d_l(e)^2} - I_l)^2}{[\mathrm{var}(d_l(e)^2)]^{\alpha(e)}}] > \log[\frac{(\overline{d_k(e)^2} - I_k)^2}{[\mathrm{var}(d_k(e)^2)]^{\alpha(e)}}] \quad (13)$$

Then to well-classify the observation e, the parameter $\alpha(e)$ must be set to this new value :

$$\alpha(e) = \frac{\log[\frac{(\overline{d_k(e)^2} - I_k)^2}{(\overline{d_l(e)^2} - I_l)^2}]}{\log[\frac{\mathrm{var}(d_k(e)^2)}{\mathrm{var}(d_l(e)^2)}]}. \quad (14)$$

At the end of the procedure, a set of possible values for the parameter α is obtained. The cardinal of this set is the number of misclassified observations. It is easily proven that the optimal value α^* belongs to this set: α^* is then selected by maximizing the cross-validated recognition rate over this finite set.

4.2 Local adaptation

For the local adaptation, since the parameters are optimized relatively to each others, $G-1$ parameters are adapted, G being the number of classes. The reference value is set to one for the first class, for example. If $G = 2$, only one parameter is to be estimated (α_2, or β_2): this leads to the basic learning procedure. If $G > 2$, the learning procedure is recursive, decomposing the multi-class problem as a sequence of two-class problems.

4.2.1 Learning for a two-class problem

The local adaptation allows to take into account different local structures for each class. Here, the parameter for the second class will be set relatively to the first class. Let us consider the two classes ω_1 and ω_2. For the class ω_1, the parameter is fixed and set to one. For the second class, the initial value of the parameter is also set to one. This value is only modified for misclassified observations. Two cases occur which are summarized in Table 2. For example, for case 1 with model 1, the inequality between the variation coefficients to well-classify the observation e is

	True class	Selected class	Action	Model 1	Model 2
Case 1	1	2	$Cv_2 \nearrow$	$\alpha_2(e) \searrow$	$\beta_2(e) \nearrow$
Case 1	2	1	$Cv_2 \searrow$	$\alpha_2(e) \nearrow$	$\beta_2(e) \searrow$

Table 2: Parameter modification on misclassified observations

$$\mathrm{Cv}(d_2(e)^2) > \mathrm{Cv}(d_1(e)^2)$$

$$\log[\frac{(\overline{d_2(e)^2} - I_2)^2}{[\mathrm{var}(d_2(e)^2)]^{\alpha_2(e)}}] > \log[\frac{(\overline{d_1(e)^2} - I_1)^2}{\mathrm{var}(d_1(e)^2)}] \quad (15)$$

To verify this inequality, the parameter $\alpha_2(e)$ must be set to :

$$\alpha_2(e) = \frac{\log[(\overline{d_2(e)^2} - I_2)^2] - \log[(\overline{d_1(e)^2} - I_1)^2] + \log[\mathrm{var}(d_1(e)^2)]}{\log[\mathrm{var}(d_2(e)^2)]} \quad (16)$$

For model 2, with a similar approach, the parameter $\beta_2(e)$ for the misclassified observations must be such that

$$\log[\beta_2(e)] = \log[Cv_1(e)] - \log[Cv_2(e)] \quad (17)$$

The final step of the procedure consists of selecting the best value among this set of candidates, maximizing the cross-validated recognition rate. The optimality of this procedure is illustrated in Figure 6.

4.2.2 Learning for a multi-class problem ($G > 2$)

These procedures can be easily extended to the general case, for a multi-class problem ($G > 2$). This extension is realized recursively from the procedure restricted to a two-class problem.

Let us notice $\mathcal{X}_{12\cdots k}$, the learning set restricted to the classes ω_1, ω_2, ... ω_k. Let us consider the class ω_1 as the reference, α_1 (or β_1) is constant and set to one. Starting from initial values set to one, the $G-1$ parameters, from α_2 (or β_2) to α_G (or β_G) are recursively optimized according to a $G-1$ steps procedure. At each step k, α_k^* (or β_k^*) is set, maximizing the cross-validated recognition rate on $\mathcal{X}_{12\cdots(k+1)}$.

Let us consider the step k on $\mathcal{X}_{12\cdots(k+1)}$. Only two following misclassification cases are considered2 :

- $cl(e) = k+1$ and $w(e) \neq cl(e)$. Then α_{k+1} must be decreased relatively to $\alpha_{\omega(e)}$ according to (16) (or increase β_{k+1} relatively to $\beta_{\omega(e)}$ according to (17)),
- $w(e) = k+1$ and $w(e) \neq cl(e)$. Then α_{k+1} must be increased relatively to $\alpha_{\omega(e)}$ according to (16) (or decrease β_{k+1} relatively to $\beta_{\omega(e)}$ according to (17)).

The other misclassification cases are ignored since they do not concern class ω_{k+1}. This step k is completed by the selection of the optimal parameter maximizing the cross-validated recognition rate on $\mathcal{X}_{12\cdots(k+1)}$. This procedure is running up to the step $G-1$. The $G-1$ parameters are optimal maximizing the recognition rate on $\mathcal{X}$. Actually, the recognition rate τ is the sum of elementary recognition rate on each class : $\tau = \sum_{k=1}^{G} \tau(k)$. And, each parameter α_k^* (or β_k^*) optimizes the partial sum $\sum_{j=1}^{k} \tau(j)$, with α_1 (or β_1) = 1.

5 Experimental Results

To illustrate the decision rules based on variation coefficients, experiments have been realized on a database of 449 observations, This distance data set has been designed with protein sequences from *Bacillus subtilis* extracted from the SWISSPROT databank release 38 (see [1]). Those proteins were classified into 2 categories according to their "subcellular location" keyword: 151 cytoplasmic proteins and 298 integral menbrane proteins. The amino-acid usage of each protein (i.e. the frequency of each of the 20 amino-acid) was computed and give rise to the distance table at hand.

Five decision algorithms have been benchmarked in this context. Three decision rules are based on variation coefficients : simple CV (12), CV through the model 1 (7) and 2 (8) with a local adaptation. The two other algorithms are the KNN and the 1NN classifier. The recognition rate is estimated by an "Half Sampling" learning procedure. The database is split into 2 parts. In a first step, the learning parameters (α_2 for CV-Mod1, β_2 for CV-Mod2, and K for KNN) are optimized by cross validation with the first part of the database. These parameter values are then used, for validation, to classify the second part of the database. This leads to a first recognition rate τ_{v2}. In a second step, the role of the two database parts are inverted and a second recognition rate τ_{v1} is also processed for validation with the part 1. The final rate (τ_{hs}) is the average of these two estimates. Table 3 and Figure 5 summarize the mean behaviour on 10 experiments (10 random partitions into two parts). With

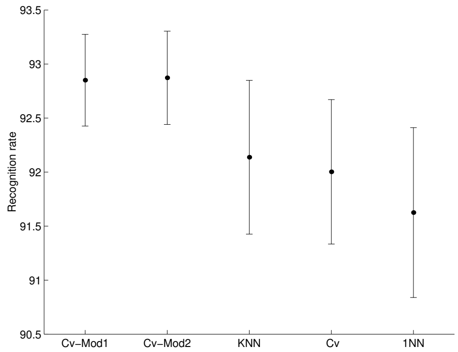


Figure 5: Half sampling procedure : Recognition rate (mean, standard deviation) on 10 partitions

this database, the recognition rates are high for the five methods. Nevetheless, differences appear between the algorithms :

- As expected, algorithms with a data driven learning parameter give better results.
- Decision rules based on adaptive variation coefficients give better recognition rates (higher mean, reduced standard deviation).
- The dependence between the optimal parameters and a data partition is smaller with the " CV rules" than with the "KNN rule" (reduced deviation between τ_l and τ_v). Then a better generalization can be expected with such "CV methods".

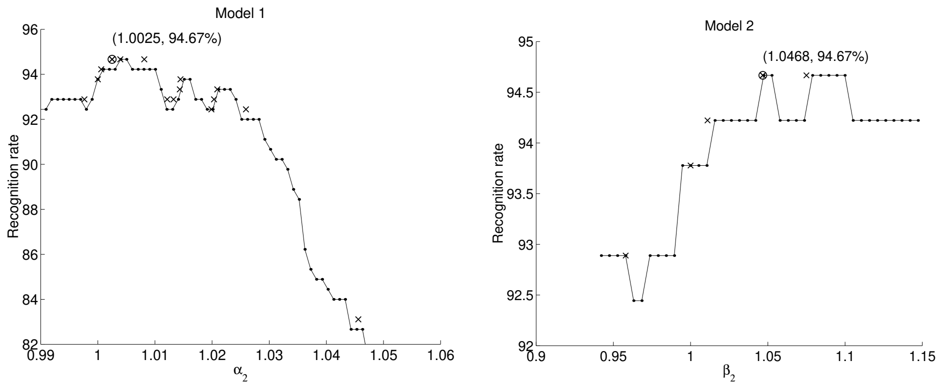
Figure 6 illustrates this optimal learning procedure by the variation of the recognition rate versus the fitting parameter. On this figure, the different possible values for each misclassified observation are marked by crosses and the selected optimal value by a circle. In this interval, for regularly sampled parameter values (dot), the recognition rate are always lower than the maximum cross-validated recognition rate.

Concerning the processing time, with predifined parameters, the"CV algorithms" are in average 20 times faster than the "KNN" one. The processing time for the learning step depend on the number of misclassified observations. For this example, the learning time is in average 10 times longer than the test time on the finite dataset.

Dealing with incomplete dissimilarity table, the "CV algorithms" have nice behaviour. The principle is the same but the statistics are only set on the known dissimilarities. Numerical experiments not reported here

Algorithm	CV-Mod 1	CV-Mod2	KNN	CV	1NN
Learning parameter on part 1	$\alpha_2 = 1.01$	$\beta_2 = 1.18$	$K = 8.4$	-	-
Learning on part 1 τ_{l1} (%)	94.1	94.1	93.7	-	-
Validation on part 2 τ_{v2} (%)	92.8	92.8	91.9	91.4	91.5
Learning parameter on part 2	$\alpha_2 = 1.01$	$\beta_2 = 1.29$	$K = 7.4$	-	-
Learning on part 2 τ_{l2} (%)	93.2	93.2	93.5	-	-
Validation on part 1 τ_{v1} (%)	92.9	92.9	92.4	92.6	91.8
Final rate $\tau_{hs} = \frac{\tau_{v1}+\tau_{v2}}{2}$ (%)	92.8	92.9	92.1	92	91.6
Deviation $\tau_l - \tau_v$ (%)	0.82	0.80	1.47	-	-

Table 3: Recognition rates on the proteins database on 10 random partitions.



(a) (b)

Figure 6: Result of a learning step : Recognition rate vs the fitting parameter for the model 1 (a) and the model 2 (b). See text for more details.

show that the recognition rate is robust to the sparse dissimilarity structure up to 40% of unknown dissimilarities.

6 Conclusions

The development of "data mining" techniques enhances the great need to have multiple classification tools adapted to various data structures. The dissimilarity tables are one of these structures. In this domain, we have presented a new sensible classification framework inspired from the Euclidian Gaussian model. The proposed set of decision rules is an alternative to the well-known "KNN" rule. The characteristics of these decision rules are simplicity, rapidity (recursive implementation, few adaptive parameters), robustness to the size of the dataset (based on first and second order statistics on dissimilarity values), data driven versatility (adaptive parameters to learn the "shape" and the intrinsic dimension of each class), adaptation to incomplete dissimilarity data (statistics only on known values). This last property is very important for applications dealing with huge databases, since the dissimilarity table is a quadratic data structure.

A simple illustrative example for a protein classification problem shows already a very interesting behaviour compared to the "KNN" rule. Extensive experiments with more complex data must be performed to completely validate this new concept of classification from dissimilarity tables.

References

[1] A. Bairoch and R. Apweiler. The swiss-prot protein sequence database and its supplement trembl in 2000. *Nuclear Acids Research*, 28:45–48, 2000.

[2] I. Borg and P. Groenen. *Modern Multidimensional Scaling : Theory and Applications*. Springer-Verlag New-York, Inc., 1997.

[3] K. Fukunaga. *Statistical Pattern Recognition*. Academic Press, Inc., 2nd edition, 1990.

[4] M.E. Tipping. *Topographic Mappings and Feed-Forward Neural Networks*. PhD thesis, Univ. of Aston., UK, 1996. http://www.ncrg.aston.ac.uk.