Generalizing Dissimilarity Representations Using Feature Lines

Mauricio Orozco-Alzate[1,2], Robert P.W. Duin[1],
and César Germán Castellanos-Domínguez[2]

[1] Information and Communication Theory Group, Faculty of Electrical Engineering,
Mathematics and Computer Science, Delft University of Technology, P.O. Box 5031,
2600GA Delft, The Netherlands
{m.orozcoalzate,r.p.w.duin}@tudelft.nl

[2] Grupo de Control y Procesamiento Digital de Señales, Universidad Nacional de
Colombia Sede Manizales, Carrera 27 # 64-60, Manizales (Caldas), Colombia
{morozcoa,cgcastellanosd}@unal.edu.co

Abstract. A crucial issue in dissimilarity-based classification is the choice of the representation set. In the small sample case, classifiers capable of a good generalization and the injection or addition of extra information allow to overcome the representational limitations. In this paper, we present a new approach for enriching dissimilarity representations. It is based on the concept of feature lines and consists in deriving a generalized version of the original dissimilarity representation by using feature lines as prototypes. We use a linear normal density-based classifier and the nearest neighbor rule, as well as two different methods for selecting prototypes: random choice and a length-based selection of the feature lines. An important observation is that just a few long feature lines are needed to obtain a significant improvement in performance over the other representation sets and classifiers. In general, the experiments show that this alternative representation is especially profitable for some correlated datasets.

Keywords: Dissimilarity, representation, feature lines, generalization.

1 Introduction

The nearest neighbor method (k-NN) [1] is a simple and asymptotically well-behaved classifier, which classifies an object x by assigning it the class label[1] $\hat{c}$ most frequently represented among the k nearest training objects. In a conventional feature space representation, x is represented as a feature point $\boldsymbol{x}$. Consider a training set $T = \{\boldsymbol{x}_i^c, 1 \leq c \leq C, 1 \leq i \leq n_c\}$, where C is the number

[1] In order to simplify the notation, ours differs from the usual way to denote the set of class labels, i.e. $\Omega = \{\omega_1, \ldots, \omega_c\}$. In this paper, we denote the membership or association to one of the C classes by using the letter c as a variable running from 1 to C. Besides, when a particular value of c is used as a subscript, it is written between round brackets.

L. Rueda, D. Mery, and J. Kittler (Eds.): CIARP 2007, LNCS 4756, pp. 370–379, 2007.

of classes and n_c the number of objects per class. For $k = 1$, the rule can be written as follows:

$$d(\boldsymbol{x}, \boldsymbol{x}_{\hat{i}}^{\hat{c}}) = \min_{1 \leq c \leq C,\ 1 \leq i \leq n_c} d(\boldsymbol{x}, \boldsymbol{x}_i^c), \tag{1}$$

where $d(\boldsymbol{x}, \boldsymbol{x}_i^c) = \|\boldsymbol{x} - \boldsymbol{x}_i^c\|$ is usually the (weighted) Euclidean or the city block norm. Using the entire training set implies $N = \sum_{c=1}^{C} n_c$ distance calculations; as a result, considerable space requirements to store T and a high computational effort for the evaluation of new objects might be required. A straightforward solution to this drawback is selecting a representation set R, which is chosen to be a subset of T ($R \subseteq T$) or even a distinct set having a cardinality n lower than that of T.

More generally, d might be a dissimilarity measure, metric or not, computed or derived from the objects directly, their sensor representations, or some initial representation [2]; in other words, if a companion feature representation is not necessarily involved, $d(x, p_i)$ denotes a dissimilarity measure between an object and one of the representative objects (prototypes) from R. Those measures, arranged as a vector $D(x, R) = [d(x, p_1), d(x, p_2), \ldots, d(x, p_n)]$, constitute a *dissimilarity representation* of x. For the training set T, it extends to an $N \times n$ dissimilarity matrix $D(T, R)$ and a set S of new objects is provided in terms of their distances to R, i.e. as a matrix $D(S, R)$. Analogously to (1), the 1-NN rule in the dissimilarity representation assigns a new object to the class of its nearest neighbor from R by finding the minimum in the rows of $D(S, R)$.

In addition to the storage and computational disadvantages, the NN rule suffers from other limitations, e.g. sensitivity to noise and potential loss of accuracy when a limited number of prototypes is available or when their representational capacity is not enough to cover the possible variations of data. A number of strategies have been proposed to handle such situations, e.g., modifying the rule [3, 4, 5], adapting the distance measure [6, 7, 8, 9], expanding the representational capacity of the available feature points [10, 11] and building Bayesian classifiers on the dissimilarity representations [12, 13]. Combining some of those strategies, taking advantage of their individual properties, may be effective. In particular, we will study the use of the nearest feature line method [10] for generalizing dissimilarity representations and constructing Bayesian classifiers in such a generalized dissimilarity space. The generalization procedure is intended for small sample cases. Its basic rationale is that to enhance the representation using feature lines and to achieve a better generalization, building a Bayesian classifier in the enhanced representation, may improve the performance of both techniques when they are used separately. Our experiments show that the proposed procedure is specially profitable for correlated (cigar-like or elongated) datasets.

The remainder of this paper is organized as follows. Section 2 describes the proposed procedure for generalizing dissimilarity representations. Experiments and results on artificial and real data sets are described in Section 3. Section 4 presents the conclusions and discusses some possibilities for future work.

2 Generalization Procedure

The procedure consists in creating the generalized dissimilarity representation $D_L(T, R_L)$, where L denotes that the representation set is composed by feature lines. In the original dissimilarity space approach, one considers a data-depending mapping $D(x, R) : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}^n$ to the so-called *dissimilarity space*, where each dimension corresponds to a dissimilarity $D(\cdot, p_i)$ to a particular object $p_i \in R$. Analogously, for a generalized dissimilarity space, the considered mapping is $D(x, R_L) : \mathcal{X} \times \mathcal{X}_L \rightarrow \mathbb{R}^{n_L}$. As a result, a *generalized dissimilarity representation* of x corresponds to the vector $D(x, R_L) = [d(x, L_1), d(x, L_2), \ldots, d(x, L_{n_L})]$. In this section we review the nearest feature line method as it was originally proposed for feature space representations. After that, we describe how to build a generalized dissimilarity representation using only the information available at $D(T, R)$; that is, without recurring to an associated feature representation. Indeed, feature vectors might be not available, e.g. when dissimilarities are directly derived from the objects.

2.1 Feature Lines

The *Nearest Feature Line* rule, or *NFL* [10], is an extension of the NN rule. It generalizes each pair of prototype feature points belonging to the same class: $\{\boldsymbol{x}_i^c, \boldsymbol{x}_j^c\}$ by a linear function L_{ij}^c, which is called the *feature line*. The line L_{ij}^c is expressed by the span $L_{ij}^c = \mathrm{sp}(\boldsymbol{x}_i^c, \boldsymbol{x}_j^c)$. The query $\boldsymbol{x}$ is projected onto L_{ij}^c as a point $\boldsymbol{p}_{ij}^c$ (see Fig. 1). This projection can be computed as

$$\boldsymbol{p}_{ij}^c = \boldsymbol{x}_i^c + \tau(\boldsymbol{x}_j^c - \boldsymbol{x}_i^c), \tag{2}$$

where $\tau = (\boldsymbol{x} - \boldsymbol{x}_i^c) \cdot (\boldsymbol{x}_j^c - \boldsymbol{x}_i^c) / \|\boldsymbol{x}_j^c - \boldsymbol{x}_i^c\|^2 \in \mathbb{R}$; τ is called the *position parameter*. The classification of $\boldsymbol{x}$ is done by assigning it the class label $\hat{c}$ most frequently represented among the k nearest feature lines; for $k = 1$ that means:

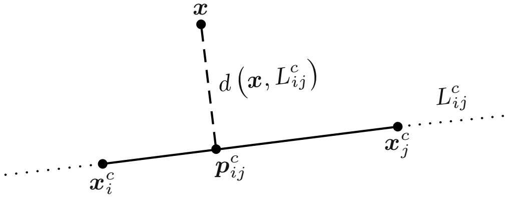

Fig. 1. Feature line L_{ij}^c and computation of the distance to it

$$d(\boldsymbol{x}, L_{\hat{i}\hat{j}}^{\hat{c}}) = \min_{\substack{1 \leq c \leq C,\ 1 \leq i,j \leq n_c \\ i \neq j}} d(\boldsymbol{x}, L_{ij}^c) \tag{3}$$

where $d(\boldsymbol{x}, L_{ij}^c) = \|\boldsymbol{x} - \boldsymbol{p}_{ij}^c\|$.

2.2 Distances to Feature Lines in Terms of Dissimilarities

Given a dissimilarity matrix $D(T, R)$, deriving the distances to feature lines consists in computing the height h of a scalene triangle as shown in Fig. 2 . Note that d_{ij} must be an intraclass distance. In addition, since any metric triplet d_{ij}, d_{ik} and d_{jk} is Euclidean (i.e. it constitutes a Euclidean triangle), we restrict our experiments to metric distance matrices. Such a restriction does not imply a loss of generality because an embedding can be found to correct a non-metric D, e.g. through a pseudo-Euclidean embedding [14].

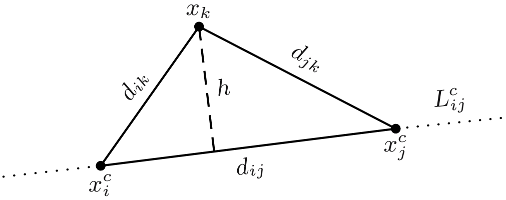

Fig. 2. Scalene triangle for computing the distance to a feature line in terms of dissimilarities

Let define $s = (d_{jk} + d_{ij} + d_{ik})/2$. Then, the area of the triangle is given by:

$$A = \sqrt{s(s - d_{jk})(s - d_{ij})(s - d_{ik})}; \tag{4}$$

but we also know that area, assuming d_{ij} as base, is:

$$A = \frac{d_{ij}h}{2} \tag{5}$$

We can solve (4) and (5) for h, which is the distance to the feature line, i.e. $d(x_k, L^c_{ij})$. The generalized dissimilarity representation for a particular object x_k is constructed by arranging all the $n_L = \sum_{c=1}^{C} n_c(n_c - 1)/2$ distances to the feature lines in a vector $D(x_k, R_L)$. As for the original dissimilarity representations, for a training set T it extends to a $N \times n_L$ dissimilarity matrix $D(T, R_L)$. In general, $D(T, R_L)$ is not square and has two zeros elements per column. The information on a set S of new incoming objects is provided in terms of their distances to R_L, i.e., as a generalized dissimilarity matrix $D(S, R_L)$.

2.3 Classification in Dissimilarity Spaces

As $D(\cdot, p_i)$, a dissimilarity $D(\cdot, L_i)$ to a particular feature line $L_i \in R_L$ can be interpreted as an attribute, allowing for building classifiers in such a space. Previous studies [12, 13] showed that building Bayesian classifiers in dissimilarity spaces, e.g. a linear normal density based classifier, often outperforms the k-NN rule, especially for small representation sets or non-representative training sets. The use of normal density based classifiers in dissimilarity spaces is suggested because the summation-based distances are often approximately normally distributed (in fact, a clipped normal distribution due to the nonnegativity of dissimilarities) [13]. There is no practical difference between constructing a classifier on generalized dissimilarities and to build it on non-generalized dissimilarities. Thereby, the classifier definition is the same either the representation is generalized or not. For a two-class problem, a linear decision function (BayesNL) based on the representation set R is given by (The same applies for R_L)

$$\begin{aligned} f(D(x,R)) = \left[D(x,R) - \frac{1}{2}\left(\boldsymbol{m}_{(1)} + \boldsymbol{m}_{(2)}\right)\right]^T \\ \times \boldsymbol{C}^{-1}\left(\boldsymbol{m}_{(1)} - \boldsymbol{m}_{(2)}\right) + \log\frac{P_{(1)}}{P_{(2)}}, \end{aligned} \tag{6}$$

where $\boldsymbol{C}$ is the sample covariance matrix, $\boldsymbol{m}_{(1)}$ and $\boldsymbol{m}_{(2)}$ are the mean vectors, $P_{(1)}$ and $P_{(2)}$ are the class prior probabilities. When $\boldsymbol{C}$ becomes singular, it is regularized by using for example the following strategy [15]: $\boldsymbol{C}_{reg}^{\lambda} = (1-\lambda)\,\boldsymbol{C} + \lambda \mathrm{diag}(\boldsymbol{C})$. In practice, λ equals 0.01 or less [2]. We keep it fixed to 0.01 in our experiments.

3 Experiments and Results

We test the application of the generalization method on several artificial and real-world datasets. Two selection procedures, random and length-based, are used for selecting prototype feature lines. Due to space constraints and in order to illustrate when the generalization is advantageous, we only present results for some datasets which were found to be benefited by the generalization. In other words, we are not claiming that our strategy gives an overall best solution, but the results do show that there exist problems for which the proposed method is beneficial. The presented results correspond to the following artificial and real-world problems:

Difficult normally distributed classes. It corresponds to a two-dimensional and two-class dataset having very different class variances for the dimensions (see `gendatd` function in [15]). Separation is thereby, for small sample sizes, difficult.

Highleyman classes. A two-dimensional and two-class dataset generated by the Highleyman distribution [16] (see also `gendath` function in [15]).

Wine data. The *Wine* data come from Machine Learning Repositary [17] and describe three types of wine by 13 features.

Laryngeal data. The *Laryngeal* dataset comes from the Bulgarian Academy of Sciences and is available at [18]. The set was originally used for a computer decision support system, in order to aid diagnosis of laryngeal pathology and especially in detecting its early stages. Normal and pathological voices are described by 16 parameters in the time, spectral and cepstral domains.

In all the experiments, a Euclidean distance was chosen for the original dissimilarity representations. The reported results are based on 25 repetitions; however, in order to maintain the clarity of the plots, we do not present the resulting standard deviations. In general, we found that they vary between 2% and 6% of the averaged errors. Implementations are done using PRTools [15].

Figs. 3 and 4 show the classification errors of the 1-NN and BayesNL classifiers applied to the generalized dissimilarity representations, as a function of the number of prototype feature lines chosen by two length-based selection methods: ascending and descending orders. The initial representation set R_L for the ascending method is the shortest feature line, i.e. the shortest base of the triangles (see Fig. 2). Then, the second shortest feature line is added to R_L, followed by the third shortest one and so on. The reverse case corresponds to the selection in descending order. In brief, the first m reported errors (m left most values) in Figs. 3 and 4 correspond to classification using the m shortest/largest feature lines. At the end, when all the $\sum_{c=1}^{C} n_c(n_c-1)/2$ feature lines are included, the length-ranked representation sets are flipped versions of each other. In these experiments, we use $n_c = 15$. In order to explore its influence, we performed additional experiments for $n_c = 10$ and $n_c = 20$; however, it was not observed a significant difference in the general behavior. The same figures[1] show the best results obtained by the 1-NN and the BayesNL rules in the original dissimilarity spaces. They are plotted as horizontal lines and constitute our reference. In both cases, the representation set R is chosen by random selection. In consequence, such best results do not necessarily correspond to the case of using the entire T for representation.

The BayesNL classifier based on the descending ranked R_L outperforms both the best results in the original dissimilarity space and the other studied alternatives in the generalized one. Comparing the two feature line selection criteria, it is noteworthy that few long feature lines are needed to yield a good result with the BayesNL classifier. This fact may be explained as follows: long feature lines, which are chosen at first by the descending order selection, provide continua across the main direction of data. Such continua, in the case of elongated datasets, resemble the principal axis of an hyperellipse. More generally, they can be interpreted as a piecewise description. In principle (in absence of outliers), the feature lines represent the data as a structural model, i.e. through a *generalization* of their geometric spread. In contrast, for small representation sets and the descending order, the 1-NN method is negatively affected. As claimed in [12] for non-generalized representations, a possible interpretation is that when R or R_L are small, they refer to the objects that differ much from each other, potentially including also outliers.

[1] Note that the 1-NN rule, directly applied to the dissimilarity representations $D(S,R)$ or $D(S,R_L)$, consists in looking for the minima in the rows of the matrices. Thereby, its application to those representations corresponds to the 1-NN and the NFL classifiers, respectively. In other words, we are not deriving a new distance representation from the vectors $D(x,R)$ or $D(x,R_L)$.

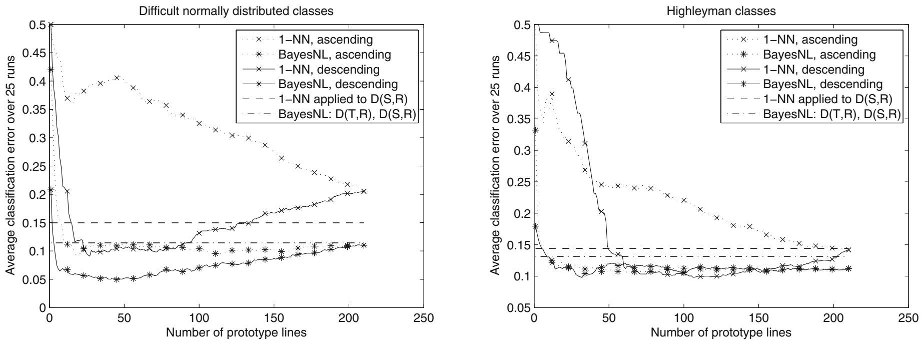

Fig. 3. Artificial data. Average classification errors of the BayesNL and 1-NN classifiers in the generalized dissimilarity space. Feature lines are incrementally included according to their length. Horizontal lines are the best results achieved in the original dissimilarity representations.

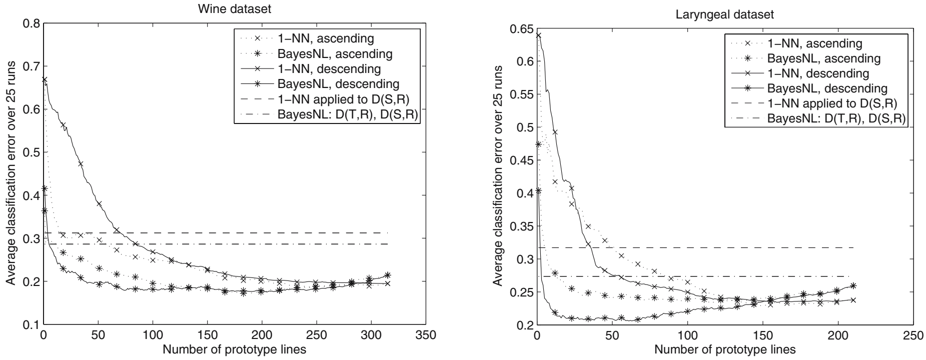

Fig. 4. Real-world data. Average classification errors of the BayesNL and 1-NN classifiers in the generalized dissimilarity space. Feature lines are incrementally included according to their length. Horizontal lines are the best results achieved in the original dissimilarity representations.

Figs. 5 and 6 show the results when the number of the selected prototypes, for both points and lines, is fixed to be a proportion of the cardinality of T: $n = n_L = n_c C/5$. For instance, for the Highleyman classes (two-class problem) and 12 training objects per class, the number of prototypes (points or lines) selected for representation is 5. Again and as expected, the BayesNL classifiers yield a better performance than the 1-NN rules based on the same representations sets either R or R_L.

As an additional criterion to evaluate the discriminative capacity of the generalized dissimilarity representations, we examine the Mahalanobis distance $d_{(i,j)}$ between each pair of classes. The larger Mahalanobis distance, the larger discriminative capacity between data classes. A clear enlargement of such a capacity is observed in Figs. 7 and 8.

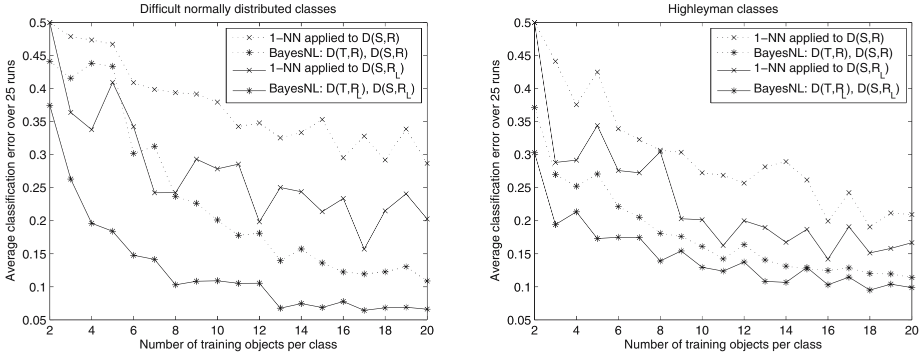

Fig. 5. Artificial data. Average classification errors of the BayesNL and 1-NN classifiers in the original and the generalized dissimilarity spaces. A rule-of-thumb of selecting $n_cC/5$ prototypes is used.

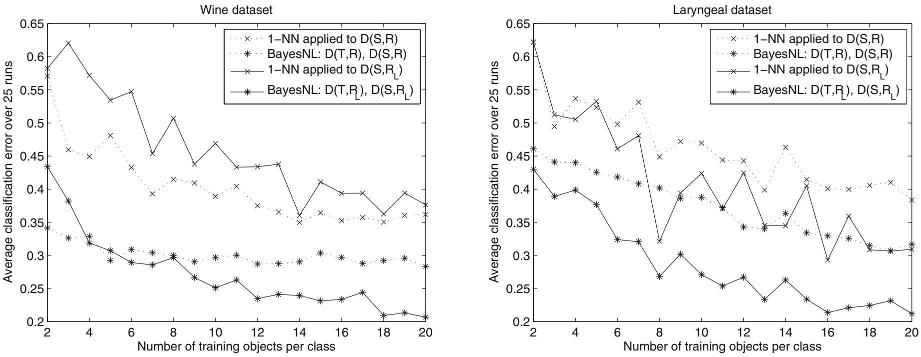

Fig. 6. Real-world data. Average classification errors of the BayesNL and 1-NN classifiers in the original and the generalized dissimilarity spaces. A rule-of-thumb of selecting $n_cC/5$ prototypes is used.

4 Conclusions

Here we have proposed a generalization procedure for dissimilarity representations. The method is based on the feature line concept, which was originally proposed for face recognition problems. Our experiments showed that the generalization procedure, when using a random and a length-based selection of prototype feature lines, seems to be especially profitable for elongated (cigar-like) datasets. Compared to the non-generalized dissimilarity representations, the generalized ones exploit more the intrinsic geometric information available at the pairwise dissimilarities, effectively finding an enriched representation. Additionally, the method is particularly advantageous for small sample size problems because in such sparse spaces, the feature lines are somewhat filling them. Further studies on prototype selection will be conducted as well as on generalization by using feature planes.

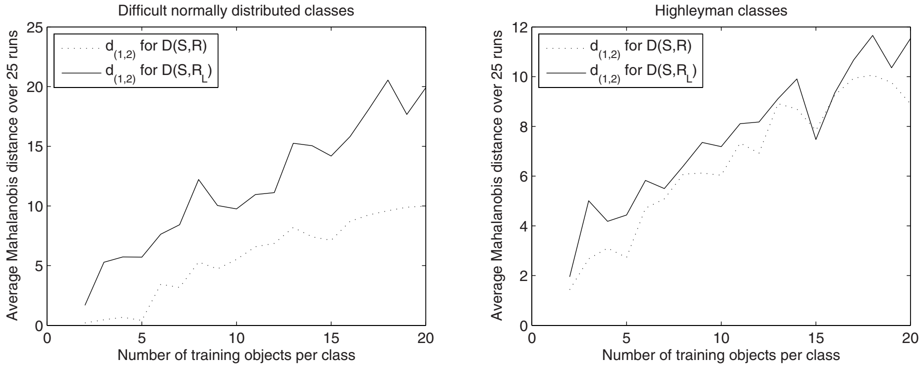

Fig. 7. Mahalanobis distance $d_{(1,2)}$ for the two-class artificial datasets

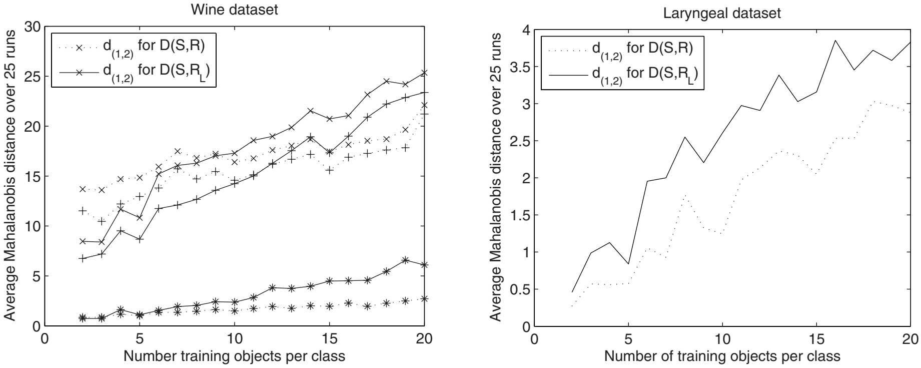

Fig. 8. Mahalanobis distances for the two real-world datasets. Pluses and stars in the left plot correspond to Mahalanobis distances between the other classes, $d_{(1,3)}$ and $d_{(2,3)}$ respectively. They are not specified in the legend for clarity reasons.

Acknowledgments. This work is supported by a TU Delft Research Grant, the Scholarship Program for Outstanding Postgraduate Students from Universidad Nacional de Colombia and the Research Project "Técnicas de computación de alto rendimiento en la interpretación automatizada de imágenes médicas y bioseñales". The authors would like to thank Dr. Elżbieta Pękalska for her useful comments on the manuscript.

References

[1] Cover, T.M., Hart, P.E.: Nearest neighbor pattern classification. Theory IT-13(1), 21–27 (1967)

[2] Pękalska, E., Duin, R.P.W., Paclík, P.: Prototype selection for dissimilarity-based classifiers. Pattern Recognition 39, 189–208 (2006)

[3] Hastie, T., Tibshirani, R.: Discriminant adaptive nearest neighbor classification and regression. In: Touretzky, D.S., Mozer, M.C., Hasselmo, M.E. (eds.) Advances in Neural Information Processing Systems, vol. 8, pp. 409–415. The MIT Press, Cambridge (1996)

[4] Sánchez, J.S., Pla, F., Ferri, F.J.: Improving the k-NCN classification rule through heuristic modifications. Pattern Recognition Letters 19(13), 1165–1170 (1998)

[5] Domeniconi, C., Peng, J., Gunopulos, D.: Locally adaptive metric nearest-neighbor classification. IEEE Trans. Pattern Anal. Mach. Intell. 24(9), 1281–1285 (2002)

[6] Wilson, D.R., Martínez, T.R.: Improved heterogeneous distance functions. J. Artif. Intell. Res (JAIR) 6, 1–34 (1997)

[7] Avesani, P., Blanzieri, E., Ricci, F.: Advanced metrics for class-driven similarity search. In: DEXA 1999: Proceedings of the 10th International Workshop on Database & Expert Systems Applications, p. 223. IEEE Computer Society, Washington, DC, USA (1999)

[8] Paredes, R., Vidal, E.: A class-dependent weighted dissimilarity measure for nearest neighbor classification problems. Pattern Recogn. Lett. 21(12), 1027–1036 (2000)

[9] Wang, J., Neskovic, P., Cooper, L.N.: Improving nearest neighbor rule with a simple adaptive distance measure. Pattern Recogn. Lett. 28(2), 207–213 (2007)

[10] Li, S.Z., Lu, J.: Face recognition using the nearest feature line method. Neural Networks 10(2), 439–443 (1999)

[11] Chien, J.T., Wu, C.C.: Discriminant waveletfaces and nearest feature classifiers for face recognition. IEEE Trans. Pattern Anal. Machine Intell. 24(12), 1644–1649 (2002)

[12] Pękalska, E., Duin, R.P.W.: Dissimilarity representations allow for building good classifiers. Pattern Recognition Lett. 23, 943–956 (2002)

[13] Pękalska, E., Duin, R.P.W.: The Dissimilarity Representation for Pattern Recognition: Foundations and Applications. World Scientific, Singapore (2005)

[14] Duin, R.P.W., Pękalska, E.: Possibilities of zero-error recognition by dissimilarity representations. In: Inesta, J.M., Mico, L. (eds.) PRIS 2002, Alicante, Spain, pp. 20–32. ICEIS Press (April 2002)

[15] Duin, R.P.W., Juszczak, P., de Ridder, D., Paclík, P., Pękalska, E., Tax, D.M.J.: PRTools4: a Matlab Toolbox for Pattern Recognition. Technical report, Information and Communication Theory Group: Delft University of Technology, The Netherlands (2004), http://www.prtools.org/

[16] Highleyman, W.H.: Linear decision functions, with application to pattern recognition. Proceedings of the IRE 50(6), 1501–1514 (1962)

[17] Newman, D.J., Hettich, S.C.L.B., Merz, C.J.: UCI repository of machine learning databases (1998), http://www.ics.uci.edu/~mlearn/MLRepository.html

[18] Kuncheva, L.I.: Real medical data sets (2005) http://www.informatics.bangor.ac.uk/~kuncheva/activities/real_data_full_set.htm