

An Experimental Study On Combining Euclidean Distances

Wan-Jui Lee, Robert P.W. Duin, Marco Loog and Alessandro Ibba

Pattern Recognition Laboratory

Delft University of Technology, The Netherlands

Email: w.j.lee@tudelft.nl

r.duin@ieee.org

m.loog@tudelft.nl

a.ibba@tudelft.nl

Abstract—Combining different distance matrices or dissimilarity representations usually can increase the performance of individual ones. In this work, we experimentally study on the performance of combining Euclidean distances and its relationship with the non-Euclideanness produced from combining Euclidean distances. Also, the relationship between the degree of non-Euclideanness from combining Euclidean distances and the correlations between these Euclidean distances are also investigated in the experiments.

From the experimental results, we observe that combining dissimilarities computed with Euclidean distances usually performs better than combining dissimilarities computed with squared Euclidean distances. Also, the improvements are highly related to the degree of non-Euclideanness. Moreover, the degree of non-Euclideanness is relatively high if two highly uncorrelated dissimilarity matrices are combined. And the degree of non-Euclideanness becomes lower if two dissimilarity matrices to be combined are more correlated.

I. INTRODUCTION

In stead of feature-based representations, pairwise distances are also commonly used in classification problems, especially in applications where features are difficult to obtain or are with too high dimensionalities. However, selecting a good dissimilarity measure for the problem at hand is a difficult task. Recently, an interesting development seeks to construct a good dissimilarity from a series of dissimilarities [1], [2], [3], [4], [5], [6], [7], [8], [9]. The most simple way to combine dissimilarities is by averaging them. But not every dissimilarity should receive the same weight in the decision process, and therefore the main force of such combination study which is called the distance metric learning [1], [2], [3], [4], [5], [6], [7], [8] is to determine the optimal weight for each dissimilarity. The criterion for searching these weights is mainly based on Fisher discriminant that maximizes the ratio of the between-class variance and the within-class variance. Optimization methods and heuristic approaches are also used for obtaining the weights. Interestingly, in distance metric learning, the squared version of dissimilarities are very commonly used and also the optimal weights for the dissimilarities are usually derived with the constraint that the combined dissimilarity should be metric. In spite of the convenience of such a setting for learning the weighted average of dissimilarities, do the choice of squared dissimilarities and the constraint of

metric dissimilarity give better combination results than other possibilities? In this work, we investigate the differences of combining dissimilarities computed with squared Euclidean distances and combining those computed with Euclidean distances. Combining squared Euclidean distances and then taking the squared root of the combined distance results in proper Euclidean distances. However, combining Euclidean distances, on the other hand, may result in non-Euclidean distances. Also, combining squared Euclidean distances equals to extending feature spaces by directly concatenating features and computing Euclidean distances in the extended space. But combining Euclidean distances extends the feature space not only by concatenating features but also considering the correlations between features. We will discuss further in the next section how this non-Euclideanness arises and investigate the influences of the non-Euclideanness of combining dissimilarities in the experiments. We will also experimentally investigate the relationship between the degree of non-Euclideanness from combining Euclidean distances and the correlations between these Euclidean distances.

The rest of the paper is organized as follows. The differences between combining dissimilarities computed with Euclidean distances and combining those computed with squared Euclidean distances are discussed in Section II. Simulation results and the influences of the non-Euclideanness of combining dissimilarities are presented in Section III. Finally, a conclusion is given in Section IV.

II. COMBINING EUCLIDEAN DISTANCES

In this section, we would like to discuss the differences between combining dissimilarities computed with Euclidean distances and combining those computed with squared Euclidean distances.

Given a n -dimensional data space which is with n features $f = \{1, 2, \dots, n\}$. Suppose there are two points $\vec{p} = (p_1, p_2, \dots, p_n)$ and $\vec{q} = (q_1, q_2, \dots, q_n)$ in this n -dimensional data space, where p_i and q_i are the feature values in dimension i . The Euclidean distance from \vec{p} to \vec{q} is therefore computed

by

$$d(\vec{p}, \vec{q}) = \sqrt{\sum_{i=1}^n (p_i - q_i)^2}. \quad (1)$$

The n -dimensional data space can also be divided into two subspaces by separating the feature set f into two different sets $f_1 = \{1, 2, \dots, k\}$ and $f_2 = \{k+1, k+2, \dots, n\}$. As given in Eq.(1), we can also derive the Euclidean distances of points \vec{p} and \vec{q} in these two different subspaces as $d(p_{f_1}, q_{f_1}) = \sqrt{\sum_{i=1}^k (p_i - q_i)^2}$ and $d(p_{f_2}, q_{f_2}) = \sqrt{\sum_{i=k+1}^n (p_i - q_i)^2}$. Obviously, $\sqrt{d^2(p_{f_1}, q_{f_1}) + d^2(p_{f_2}, q_{f_2})} = \sqrt{d^2(\vec{p}, \vec{q})}$. This means that combining the dissimilarities computed with squared Euclidean distances and then taking squared root of the combined distance results in proper Euclidean distances. Also, combining squared Euclidean distances is the same as concatenating the features first and then computing the distance in the concatenated space.

However, for Euclidean distance, this is not true because

$$\sqrt{\sum_{i=1}^k (p_i - q_i)^2} + \sqrt{\sum_{i=k+1}^n (p_i - q_i)^2} \neq \sqrt{\sum_{i=1}^n (p_i - q_i)^2},$$

and therefore combining Euclidean distances from two different spaces does not equal to concatenating these two spaces and then computing the Euclidean distances in this space.

But if we take the square of the combined Euclidean distances which is

$$\begin{aligned} & \left(\sqrt{\sum_{i=1}^k (p_i - q_i)^2} + \sqrt{\sum_{i=k+1}^n (p_i - q_i)^2} \right)^2 \\ &= \sum_{i=1}^n (p_i - q_i)^2 + \sqrt{\sum_{i=1}^k (p_i - q_i)^2 \sum_{i=k+1}^n (p_i - q_i)^2}, \end{aligned} \quad (2)$$

and then take the squared root on both sides of Eq.(3) as

$$\begin{aligned} & \sqrt{\sum_{i=1}^k (p_i - q_i)^2} + \sqrt{\sum_{i=k+1}^n (p_i - q_i)^2} \\ &= \sqrt{\sum_{i=1}^n (p_i - q_i)^2 + \sqrt{\sum_{i=1}^k (p_i - q_i)^2 \sum_{i=k+1}^n (p_i - q_i)^2}}, \end{aligned} \quad (3)$$

we can observe that combining Euclidean distances from two different spaces is to first compute the Euclidean distances in the concatenated space and then cross-reference the distances from two different spaces. Also, combining Euclidean distances from two different spaces will only equal to computing the Euclidean distances in the concatenated space under the condition that these two different spaces are actually identical or the cross-term $\sqrt{\sum_{i=1}^k (p_i - q_i)^2 \sum_{i=k+1}^n (p_i - q_i)^2}$ remains a constant. If the Euclidean distances computed in these two spaces are not very similar, the combined distances are probably not going to be Euclidean. On the other hand, if

the Euclidean distances computed in these two different spaces differ more, the degree of non-Euclidean of the combined dissimilarity is also likely to be higher.

III. EXPERIMENTS

In this section, we compare the performances of combining dissimilarities computed with Euclidean distances and combining those computed with squared Euclidean distances, respectively. Linear discriminant classifier (ldc), quadratic discriminant classifier (qdc) and 1-nearest neighbor classifier (1-nnc) are adopted in the experiments and are built with PRTOOLS [13]. Nine real-world datasets [11], [12], [13] as shown in Table I with the number of features, objects and classes, are used to have a more general investigation. All the features in the datasets are normalized with respect to their individual variances. Each dataset is also divided into two subsets by randomly splitting the features into two sets f_1 and f_2 . The results in all the experiments are the average of randomly splitting features into two sets for 1000 times. These subsets are further used for building dissimilarity matrices with Euclidean or squared Euclidean distances. For each dataset, two dissimilarity matrices D_1 and D_2 are computed with Euclidean distances based on the feature subsets f_1 and f_2 , respectively. Similarly, dissimilarity matrices D_1^2 and D_2^2 are those computed with squared Euclidean distances. The negative eigenfractions (NEF) [10] of the combined dissimilarities computed with Euclidean distances are also given in Table I. The negative eigen-fraction describes the contribution of negative eigenvalues and is captured by

$$\frac{\sum_{\lambda_i < 0} |\lambda_i|}{\sum_{j=1}^N |\lambda_j|}, \quad (4)$$

where λ_i is an eigenvalue of the dissimilarity matrix, and N is the number of objects and also the size of the dissimilarity matrix. A higher value means the negative eigenvalues have more contributions in the dissimilarity matrix.

A. Combining Dissimilarities

In Table II, the leave-one-out errors of the 1-nearest neighbor classifiers are computed in subspaces consisting features f_1 and f_2 , the combined dissimilarity computed with Euclidean distances, and the combined dissimilarity computed with squared Euclidean distances. The features f_1 and f_2 are splitted randomly for 1000 times, and the leave-one-out errors of the 1-nearest neighbor classifiers are the average of these 1000 times. Also, the errors with combining Euclidean distances are given in bold if they are smaller than those of combining squared Euclidean distances. The linear and quadratic discriminant classifiers are not used in these experiments because they cannot be built on dissimilarity matrices directly but can only be used in vector spaces. We will use dissimilarity representation to form such vector space for ldc and qdc in the next section.

From Table I and Table II, combining Euclidean distances performs better than combining squared Euclidean distances with datasets biomed, diabetes, ecoli, heart, imox and wine,

TABLE I
DATASETS.

dataset	# features	# classes	# objects	NEF of $D_1 + D_2$
biomed	5	2	194	0.1220±0.0188
diabetes	8	2	768	0.1608±0.0090
ecoli	7	3	272	0.1147±0.0345
heart	13	2	297	0.1319±0.0087
imox	8	4	192	0.1091±0.0125
ionosphere	34	2	251	0.0354 ± 0.0106
iris	4	3	150	0.0715±0.0275
sonar	60	2	208	0.0192 ± 0.0079
wine	13	3	178	0.0704± 0.0099

TABLE II
LEAVE-ONE-OUT ERROR OF 1-NEAREST NEIGHBOR CLASSIFIERS IN SUBSPACES F_1 AND F_2 , COMBINED SQUARED EUCLIDEAN DISSIMILARITY AND COMBINED EUCLIDEAN DISSIMILARITY, RESPECTIVELY.

dataset	f_1	f_2	$\sqrt{D_1^2 + D_2^2}$	$D_1 + D_2$
biomed	0.2009±0.0549	0.2181±0.0659	0.1443±0.0018	0.1341±0.0148
diabetes	0.3475±0.0336	0.3438±0.0315	0.3168±0.0104	0.3034±0.0021
ecoli	0.2203±0.1299	0.1723±0.1077	0.0846±0.0072	0.0761±0.0080
heart	0.3248±0.0751	0.2778±0.0492	0.2318±0.0149	0.2222± 0.0032
imox	0.2715± 0.1973	0.1616± 0.1294	0.0625 ± 0.0008	0.0554 ± 0.0109
ionosphere	0.1049 ± 0.0603	0.0718± 0.0155	0.0627 ± 0.0051	0.0632 ± 0.0064
iris	0.1715± 0.1490	0.1382 ± 0.1258	0.0603 ± 0.0118	0.0800 ±0.0097
sonar	0.2094 ± 0.0756	0.1681 ± 0.0255	0.1490 ± 0.0077	0.1632 ± 0.0132
wine	0.1418± 0.1273	0.0752 ± 0.0517	0.0337 ± 0.0065	0.0311 ± 0.0082

which happen to have relatively high negative eigen-fractions. For datasets ionosphere, iris and sonar, it is the other way around. This suggests that negative eigen-fraction is an important factor that makes combining Euclidean distances perform better than combining squared Euclidean distances. The negative eigen-fraction is actually the degree of non-Euclideaness in a dissimilarity matrix, and therefore, the non-Euclideaness is actually beneficial for combining Euclidean distances.

To examine the cause of non-Euclideaness when combining two Euclidean distances, the correlation between the dissimilarity matrices computed by these two Euclidean distances and its relationship with the degree of non-Euclideaness of combining these two dissimilarity matrices are given in Figure 1. The variance of the cross-terms for combining two Euclidean distances and its relationship with the non-Euclideaness of combining these two distances for each dataset is given in Figure 2. In Figure 1, features are randomly splitted into two subsets for 1000 times and each point indicates one randomly splitted pair of features. Theoretically, there should be 1000 points in each subfigure. But for datasets like biomed and iris, the number of features is very small and therefore there are many identical splittings. As a result, only few points can be seen in the figure because most of them are lying on top of each other. For each pair of randomly splitted features, the correlations between the dissimilarity matrices computed with these two different subsets of features and the negative eigen-fractions of the dissimilarity matrix which are combined with these two dissimilarity matrices are computed.

From Figure 1, we can observe that when the correlation between two dissimilarity matrices are high, the degree of non-Euclideaness for combining these two dissimilarity matrices is

low. On the other hand, when the correlation between two dissimilarity matrices are low, the degree of non-Euclideaness for combining two dissimilarity matrices becomes high. Together with the results of Table II, we can conclude that combining two dissimilar distance measurements might probably perform better than combining two similar distance measurements.

Also, from Figure 2, higher variances in cross-terms usually result in high non-Euclideaness. But it is obviously the other way around in datasets ionosphere and sonar which happen to be the datasets that non-Euclideaness do not contribute much to the performance of the classifier as given in Table II.

B. Combining Dissimilarity Representations

To compare dissimilarities with other classifiers, a representation for embedding dissimilarities into data spaces is necessary. Here, we adopt the dissimilarity representation [9], [10] that the dissimilarity space uses (selected) object dissimilarities as axes and objects as points. That is, axis 1 is the dissimilarity to object 1, axis 2 the dissimilarity to object 2 and so on. In other words, the dissimilarity or distances values becomes features. Object points are located in this space by their dissimilarities to all (selected) objects. These selected objects are also called the representation set. With this setting, we can project the objects into a vector space and build a classifier in it. In Table III, the errors of 1-nnc, ldc and qdc in the dissimilarity space using 20% of objects as the representation set are the 20-fold cross-validation errors. Combining Euclidean distances is not really better than combining squared Euclidean distances with ldc. But it is significantly much better than combining squared Euclidean distances with qdc. The exceptions are the biomed

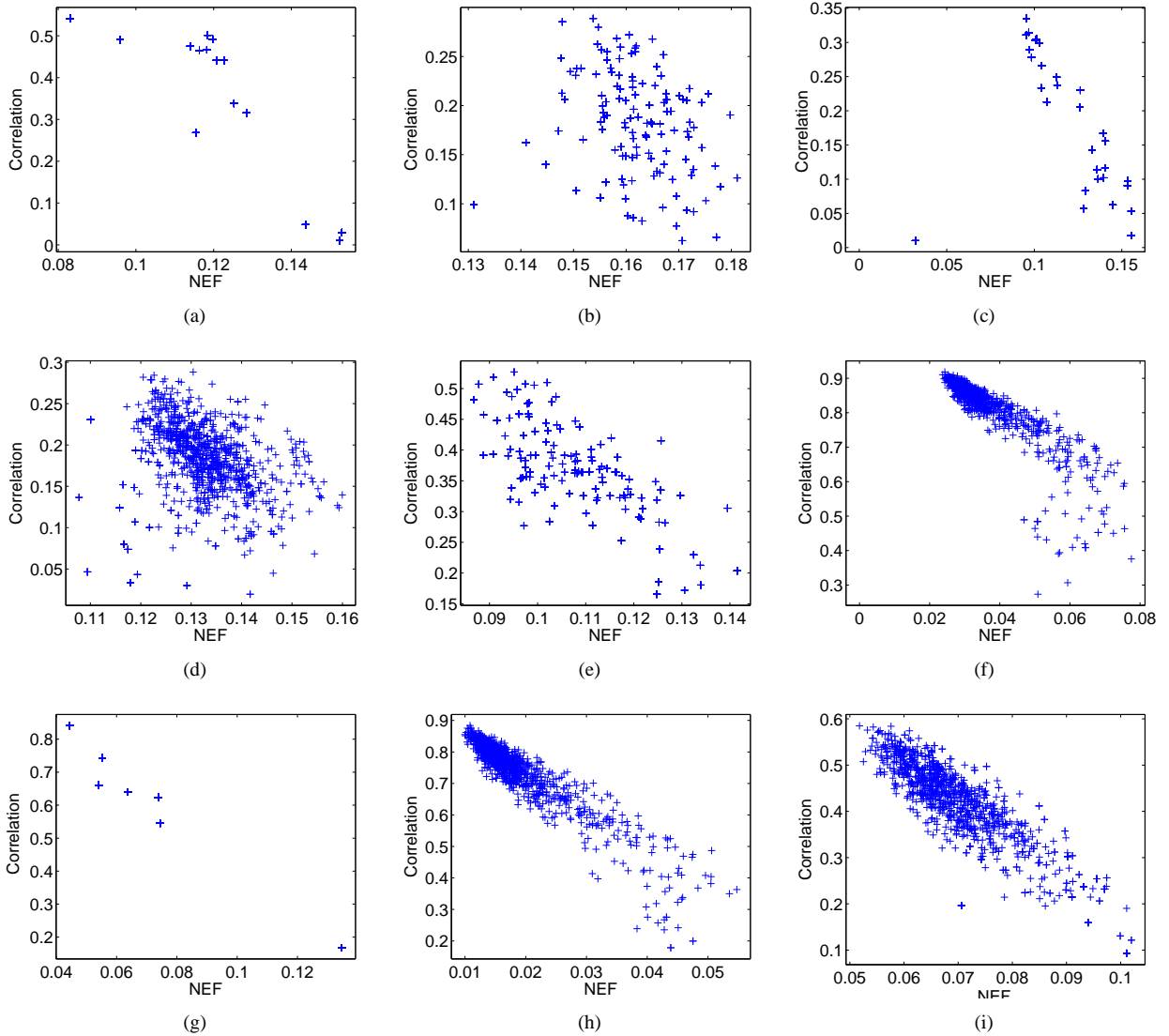


Fig. 1. Relationships between NEF of the combined distance and the correlation of the distances to be combined in dataset (a) biomed, (b) diabetes, (c) ecoli, (d) heart, (e) imox, (f) ionsphere, (g) iris, (h) sonar and (i) wine, respectively.

and wine dataset. The difference between their errors with qdc is rather small to be significant. But the difference in wine dataset is much bigger and the dataset is again with a small negative eigenfraction value. Therefore, in combining dissimilarity representations, combining Euclidean distances is very likely a better solution than combining squared Euclidean distances, especially when the non-Euclideanness can contribute.

IV. CONCLUSIONS

In this work, we investigate the differences of combining dissimilarities computed with Euclidean distances and combining those computed with squared Euclidean distances. Combining dissimilarities computed with Euclidean distances will very likely result in a non-Euclidean dissimilarity. But the non-Euclideanness is actually helpful in many situations. Also, the degree of non-Euclideanness is highly correlated with

the correlations between distances to be combined. If two distances are more correlated, the combined distance will probably have a lower degree of non-Euclideanness and vice versa. For the future directions, we would like to know whether these phenomena also hold for other distance measurements. Because this work is based on a very simple setting which only averages two dissimilarities, it will also be interesting to extend the work to a larger number of different dissimilarities and also to dissimilarities carrying different weights.

ACKNOWLEDGMENT

We acknowledge financial support from the FET programme within the EU FP7, under the SIMBAD project (contract 213250).

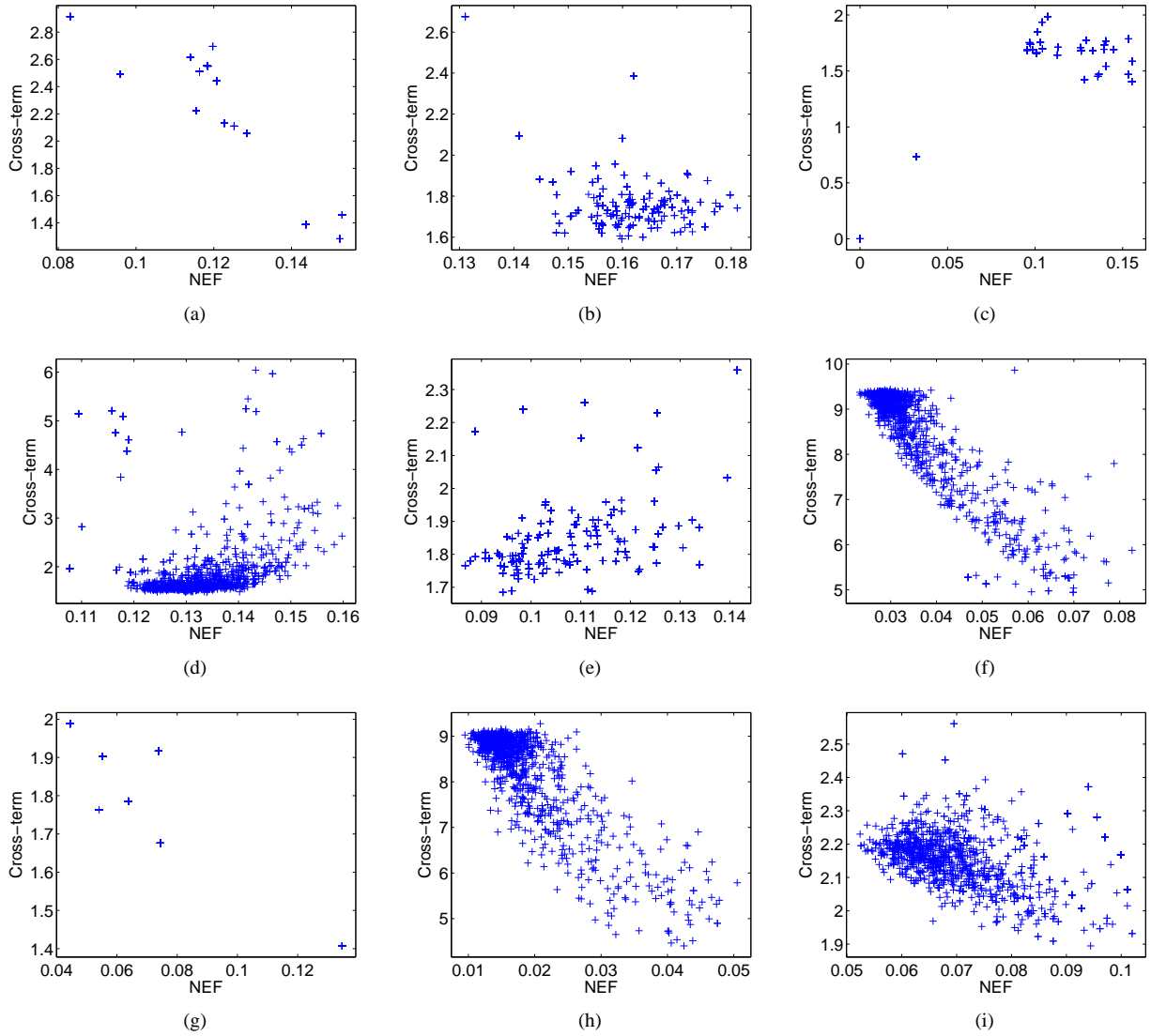


Fig. 2. Relationships between NEF of the combined distance and the variance of the cross-terms in dataset (a) biomed, (b) diabetes, (c) ecoli, (d) heart, (e) imox, (f) ionosphere, (g) iris, (h) sonar and (i) wine, respectively.

TABLE III
DISSIMILARITY REPRESENTATIONS: PROTOTYPE SELECTION USING I-NNC, LDC AND QDC.

dataset	l-nc		ldc		qdc	
	$\sqrt{D_1^2 + D_2^2}$	$D_1 + D_2$	$\sqrt{D_1^2 + D_2^2}$	$D_1 + D_2$	$\sqrt{D_1^2 + D_2^2}$	$D_1 + D_2$
biomed	0.1560 ± 0.0021	0.1423 ± 0.0114	0.1427 ± 0.0011	0.1419 ± 0.0115	0.1559 ± 0.0021	0.1564 ± 0.0202
diabetes	0.3123 ± 0.0018	0.3234 ± 0.0078	0.2934 ± 0.0009	0.2890 ± 0.0067	0.2971 ± 0.0008	0.2926 ± 0.0088
ecoli	0.0797 ± 0.0074	0.0781 ± 0.0016	0.0734 ± 0.0062	0.0711 ± 0.0009	0.0882 ± 0.0066	0.0871 ± 0.0013
heart	0.2335 ± 0.0028	0.2308 ± 0.0080	0.1921 ± 0.0027	0.1865 ± 0.0057	0.1993 ± 0.0019	0.1936 ± 0.0073
imox	0.0560 ± 0.0025	0.0573 ± 0.0058	0.1493 ± 0.0012	0.1538 ± 0.0095	0.1473 ± 0.0027	0.1427 ± 0.0146
ionosphere	0.0614 ± 0.0015	0.0607 ± 0.0061	0.2883 ± 0.0013	0.2886 ± 0.0106	0.2886 ± 0.0106	0.2576 ± 0.0141
iris	0.0685 ± 0.0013	0.0637 ± 0.0094	0.1133 ± 0.0019	0.1025 ± 0.0199	0.1034 ± 0.0017	0.0942 ± 0.0177
sonar	0.1830 ± 0.0037	0.1825 ± 0.0068	0.3946 ± 0.0022	0.3918 ± 0.0076	0.3696 ± 0.0032	0.3639 ± 0.0086
wine	0.0325 ± 0.0012	0.0361 ± 0.0065	0.0406 ± 0.0026	0.0580 ± 0.0219	0.0408 ± 0.0014	0.0575 ± 0.0207

REFERENCES

- [1] B. Alipanahi, D. R. Cheriton, M. Biggs and A. Ghodsi, *Distance metric learning vs. Fisher discriminant analysis*, Proceedings of the 23rd National Conference on Artificial Intelligence, pp. 598-603, 2008.
- [2] L. Yang, R. Jin, R. Sukthankar and Y. Liu, *An efficient algorithm for local distance metric learning*, Proceedings of the 21st National Conference on Artificial Intelligence, pp. 543-548, 2006.
- [3] A. Woznica, A. Kalousis and M. Hilario, *Learning to combine distances for complex representations*, Proceedings of the 24th International Conference on Artificial Intelligence, pp. 1034-1041, 2009.

- ference on Machine Learning, pp. 1031-1038, 2007.
- [4] A. Ghodsi, D. F. Wilkinson and F. Southey, *Improving embeddings by flexible exploitation of side information*, Proceedings of the International Joint Conference on Artificial Intelligence, pp. 810-816, 2007.
 - [5] J. Goldberger, S. Roweis, G. Hinton and R. Salakhutdinov, *Neighbourhood components analysis*, Advances in Neural Information Processing Systems 17, pp. 513-520, 2005.
 - [6] K. Weinberger, J. Blitzer and L. Saul, *Distance metric learning for large margin nearest neighbor classification*, Advances in Neural Information Processing Systems 18, pp. 1473-1480, 2006.
 - [7] K. Weinberger and L. Saul, *Distance metric learning for large margin nearest neighbor classification*, Journal of Machine Learning Research, no. 10, pp. 207-244, 2009.
 - [8] A. Globerson and S. Roweis, *Metric learning by collapsing classes*, Advances in Neural Information Processing Systems 18, pp. 451-458, 2006.
 - [9] E. Pełkalska and R. P. W. Duin, *On combining dissimilarity representations*, Proceedings of International Workshop on Multiple Classifier Systems, pp. 359-368, 2001.
 - [10] E. Pełkalska and R. P. W. Duin, *The Dissimilarity Representation for Pattern Recognition. Foundations and Applications*, World Scientific, Singapore, 2005.
 - [11] A. Asuncion and D. J. Newman, UCI Machine Learning Repository [<http://www.ics.uci.edu/~mlearn/MLRepository.html>], Irvine, CA: University of California, Department of Information and Computer Science, 2007.
 - [12] A. K. Jain and M. D. Ramaswami, *Classifier design with Parzen window*, Pattern Recognition and Artificial Intelligence, Netherlands: Elsevier, 1988.
 - [13] R. P. W. Duin, P. Juszczak, P. Paclik, E. Pełkalska, D. de Ridder and D. M. J. Tax, *PRTOOLS4, A Matlab Toolbox for Pattern Recognition*, [<http://www.prtools.org>], the Netherlands, Delft University of Technology, Pattern Recognition Laboratory, 2004.