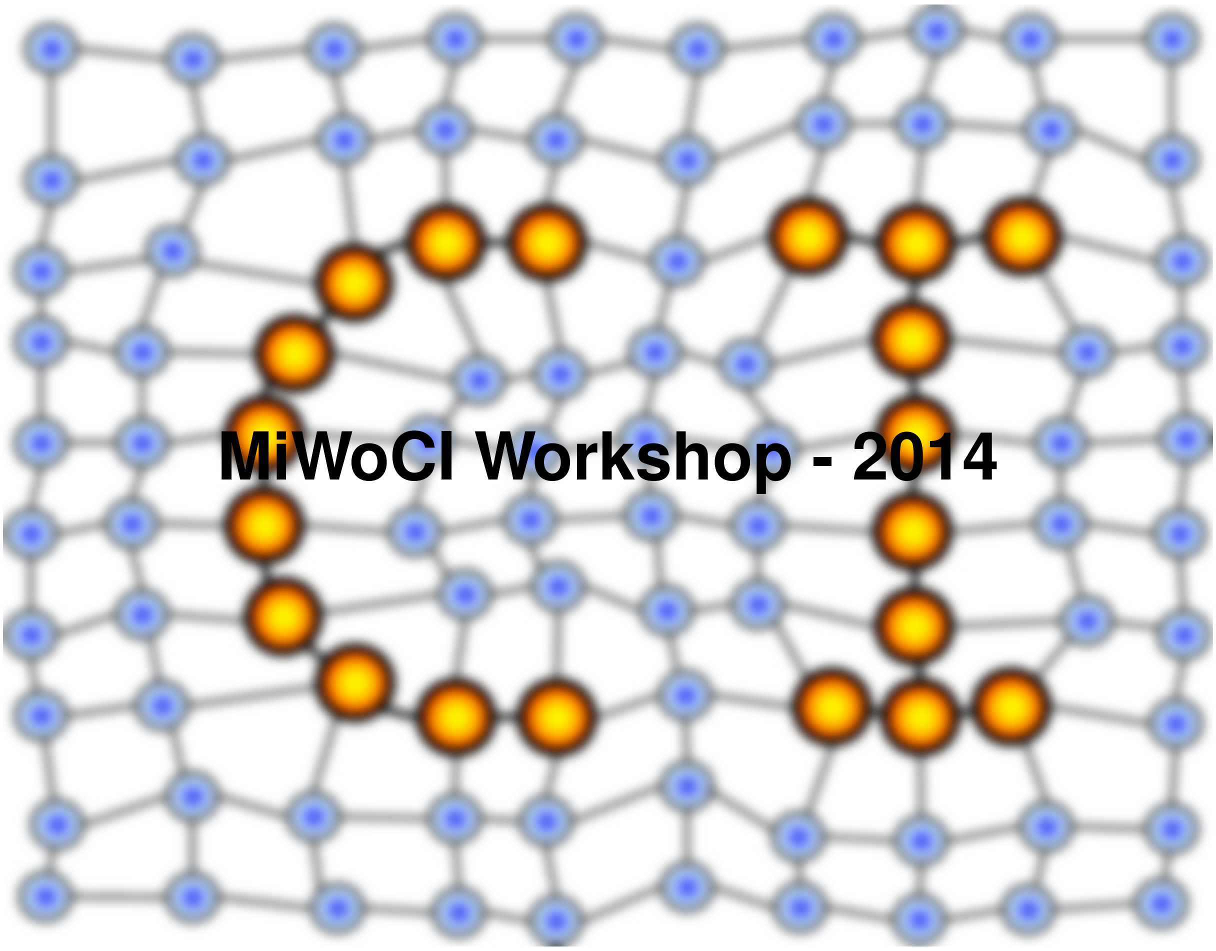
MACHINE LEARNING REPORTS



MiWoCI Workshop - 2014

Report 01/2014
Submitted: 25.06.2014
Published: 01.07.2014

Frank-Michael Schleif[1], Thomas Villmann[2] (Eds.)
(1) University of Birmingham, School of Computer Science,
Edgbaston, B15 2TT Birmingham, UK
(2) University of Applied Sciences Mittweida, Technikumplatz 17, 09648 Mittweida, Germany

Machine Learning Reports
http://www.techfak.uni-bielefeld.de/∼fschleif/mlr/mlr.html

Contents

Impressum

Publisher: University of Applied Sciences Mittweida
Technikumplatz 17,
09648 Mittweida, Germany
Editor: Prof. Dr. Thomas Villmann
Dr. Frank-Michael Schleif

Techical-Editor: PD Dr. rer. nat. habil. Frank-Michael Schleif
Contact: f.schleif@cs.bham.ac.uk
URL: http://techfak.uni-bielefeld.de/∼ fschleif/mlr/mlr.html
ISSN: 1865-3960

1 Sixth Mittweida Workshop on Computational Intelligence

From 02. Juli to 04 Juli 2014 we had the pleasure to organize and attend the sixth Mittweida Workshop on Computational Intelligence (MiWoCi 2014) as a satellite event of 10th Workshop on Self Organizing Maps (WSOM'14). Multiple scientists from the University of Bielefeld, HTW Dresden, the University of Groningen (NL), the SOM Japan Inc (Japan), the University of Birmingham (UK) and the University of Applied Sciences Mittweida met in Mittweida, Germany, to continue the tradition of the Mittweida Workshops on Computational Intelligence - *MiWoCi'2014*.

The aim was to present their current research, discuss scientific questions, and exchange their ideas. The seminar centered around topics in machine learning, signal processing and data analysis, covering fundamental theoretical aspects as well as recent applications, partially in the frame of innovative industrial cooperations. This volume contains a collection of extended abstracts and short papers which accompany some of the discussions and presented posters of the MiWoCi Workshop.

Apart from the scientific merrits, this year's seminar came up with the great chance to attend the 10th Workshop on Self Organizing Maps (WSOM'14). WSOM is the major anchor conference focusing on Self Organizing Maps and is not only a perfect chance to met high renowned researchers in the field but also to attend the three invited plenaray talks given during WSOM 2014:

- Prof. Dr. Michael Biehl, University Groningen (NL), Johann-Bernoulli-Institute of Mathematics and Computer Sciences
- Prof. Dr. Erzsebet Merenyi, Rice University Houston (USA), Department of Statistics and Department of Electrical and Computer Engineering
- Prof. Dr. Fabrice Rossi, Universite Paris1- Pantheon-Sorbonne, Department Statistique, Analyse, Modelisation Multidisciplinaire (SAMM)

This year the MiWoCi Workshop was also accompanied by a poster spotlight at the WSOM 2014 for each poster contribution and a best poster award was announced.

Our particular thanks for a perfect local organization of the workshop go to Thomas Villmann as spiritus movens of the seminar and his PhD and Master students.

Mittweida, July, 2014
Frank-M. Schleif

[1] E-mail: fschleif@techfak.uni-bielefeld.de
[2] University of Bielefeld, CITEC, Theoretical Computer Science, Leipzig, Germany

Interpretation of linear mappings employing $l1$ regularization

Alexander Schulz[1], Daniela Hofmann[1],
Michael Biehl[2] and Barbara Hammer[1]
(1) Bielefeld University, CITEC - Center of Excellence, Germany
(2) University of Groningen, Johann Bernoulli Institute for Mathematics
and Computer Science, The Netherlands

Abstract

In this contribution we propose a new technique to judge the relevance of features for a given linear mapping, thereby taking redundancy and interdependence of features into account.

We employ a two step optimization strategy: In the first step, we linearly minimize the $l1$-norm of the linear mapping $\sum_i |w_i|$, while taking redundancies in the data distribution into account. Since the first step does not necessarily yield a unique solution, we search in this solution space by minimizing/maximizing the absolute value of each single feature, respectively.

Thus, we obtain a lower and upper bound of relevance for each feature, indicating how important it minimally and maximally is (similar to strong and weak relevance in the literature).

Acknowledgement

Funding by the CITEC center of excellence is gratefully acknowledged.

Hellinger divergence in information theoretic novelty detection

Paul Stürmer* and Thomas Villmann
Computational Intelligence Group
University of Applied Sciences Mittweida

Abstract

A novelty detection framework proposed in 2009 by M. Filippone and G. Sanguinetti [6] is considered, which is suitable for small training sample sizes and allows control over the false positive rate. It is based on estimating the information content a test sample yields via the Kullback-Leibler divergence. In case of a Gaussian density estimation this approach is analytically tractable and for Gaussian mixtures appropriate approximations are provided. Here the framework is expanded by allowing the use of the Hellinger divergence [10] instead, summarizing the work done in [16].

1 Introduction

Outlier detection is an important task in machine learning where *outliers* – observations which deviate markedly from a given sample of training data [9] – are to be identified. There are many applications such as fault detection [2, 5] or monitoring medical conditions [14, 17], where such problems arise. Novelty detection concerns the case that no anomalous data is available in the training phase of the system. This is the case when outliers are costly or difficult to obtain, or when anomalous data can not to be modeled in advance. For instance, it would be unreasonable to sabotage an aircraft engine just to obtain anomalous observation data [13] and a new method of fraud is unlikely to be modeled in advance [4, 11].

Since outliers (*true positives*) are rare by definition, the accuracy alone is not sufficient for evaluating a novelty detection system. The framework presented here allows control over the *false positive rate* (*fpr*), which is the rate at which normal data samples (*true negatives*) give rise to an outlier alarm.

The key idea of the information theoretic approach is to first model the training data alone, and to train a second model which also takes a test point into account. The two models are then compared using a divergence. This is in [6] the Kullback-Leibler divergence. Here we consider the Hellinger divergence instead, which is symmetric and therefore easier to be interpreted as a distance-like measure. If the test point is a true negative, the two models are expected to be similar, resulting in a low divergence; and if it is a true positive the second model is expected to be strongly adapted, hence inducing a high divergence. Since the divergences considered here are defined for probability density functions, it is necessary to model the data via a probabilistic approach.

The goal is then to find a divergence threshold of acceptance for a test sample, which is found via Monte Carlo simulation. In this simulation phase a statistical test – in this work referred to as the F-test – is implemented, which significantly improves the performance of the framework. The framework is restricted to *Gaussian mixture models* (*GMM*s), hence assuming the data to be distributed normally. In case of single-component Gaussian densities it is analytically tractable.

[*] corresponding author, stuermer@hs-mittweida.de

The paper is structured as follows: In Section 2 the process of modeling data via *GMM*s is considered. In Section 3 the F-test is reviewed in order to incorporate it into the framework. In Section 4 the Hellinger divergence is approximated to allow an implementation. The performance of the framework is illustrated on an artificial dataset and validated on the well-known Iris dataset in Section 5, before a conclusion is given in Section 6.

2 Data model

Let $X = \{\boldsymbol{x}_1, \dots, \boldsymbol{x}_n\} \subset \mathbb{R}^d$ be the training set of samples, where each $\boldsymbol{x}_i$ is drawn from a normal distribution. Let f be the *GMM* with c components *fitted* to the training data (*training model*)

$$f(\boldsymbol{x}) = \sum_{k=1}^{c} \hat{\pi}_k \mathcal{N}(\boldsymbol{x}|\hat{\boldsymbol{m}}_k, \hat{S}_k);$$
$$\mathcal{N}(\boldsymbol{x}|\hat{\boldsymbol{m}}_k, \hat{S}_k) = \frac{1}{\sqrt{|2\pi\hat{S}_k|}} \exp\Big((\boldsymbol{x}_* - \hat{\boldsymbol{m}}_k)^T \, \hat{S}_k^{-1} \, (\boldsymbol{x}_* - \hat{\boldsymbol{m}}_k) \Big),$$

where the parameters $\hat{\pi}_i$ denote the estimated mixing coefficients, $\hat{\boldsymbol{m}}_i$ the estimated means and $\hat{S}_i$ the sample covariances for each component $i \in [c] := \{1, \dots, c\}$. The distinction between parameter estimations $\hat{\pi}_i, \hat{\boldsymbol{m}}_i, \hat{S}_i$ and the true parameters $\pi_i, \boldsymbol{m}_i, S_i$ of the generating density will become important in the next section. Such a *GMM* with maximal likelihood

$$\sum_{i=1}^{n} f(\boldsymbol{x}_i) \to \max$$

can be obtained via the *Expectation/Maximization* (*E/M*) algorithm [3], where the user has to know the number of components c in advance. For a test sample $\boldsymbol{x}_*$ the training model f is adjusted to the *adapted model* f^*

$$f^*(\boldsymbol{x}) = \sum_{k=1}^{c} \hat{\pi}_k^* \mathcal{N}(\boldsymbol{x}|\hat{\boldsymbol{m}}_k^*, \hat{S}_k^*),$$

which is obtained by performing only a single *E/M*-step on $X \cup \{\boldsymbol{x}_*\}$, starting from the parameters of f. Under the assumption that adding a single point to the fitted dataset leads to small changes in the parameter estimations of the *GMM*, it is reasonable that a single *E/M* step might already give a good estimate of the new parameters. The reason for doing this is that the updated parameters then can be denoted in closed form. It is then eventually possible to formulate an explicit divergence approximation which only depends on $\boldsymbol{x}_*$ and the parameters of f.

The Expectation step does not affect any responsibility u_{ik} of component k for the training sample $\boldsymbol{x}_i$, whereas the responsibilities for $\boldsymbol{x}_*$ are:

$$u_{*k} = \frac{\hat{\pi}_k \mathcal{N}(\boldsymbol{x}_*|\hat{\boldsymbol{m}}_k, \hat{S}_k)}{\sum_{r=1}^{c} \hat{\pi}_r \mathcal{N}(\boldsymbol{x}_*|\hat{\boldsymbol{m}}_r, \hat{S}_r)}$$

The updated cardinality of component k is $n_k^* = n_k + u_{*k} = \sum_{i=1}^{n} u_{ik} + u_{*k}$ and the updated parameter estimates are

$$\hat{\pi}_k^* = \frac{n\hat{\pi}_k + u_{*k}}{n+1}; \qquad \hat{\boldsymbol{m}}_k^* = \hat{\boldsymbol{m}}_k + \frac{u_{*k}}{n_k^*}\tilde{\boldsymbol{x}}_{*k}; \quad \hat{S}_k^* = \frac{n_k}{n_k^*}\left(\hat{S}_k + \frac{u_{*k}}{n_k^*}\tilde{\boldsymbol{x}}_{*k}\tilde{\boldsymbol{x}}_{*k}^T \right),$$

where $\tilde{\boldsymbol{x}}_{*k} := (\boldsymbol{x}_* - \hat{\boldsymbol{m}}_k)$.

3 The F-test

Let $X = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}$ be the training set, where each $\boldsymbol{x}_i \in \mathbb{R}^d$ is drawn from a multivariate Gaussian distribution $\mathcal{N}(\boldsymbol{m}, S)$. The maximum likelihood estimation of the generating distribution based on X, $\mathcal{N}(\hat{\boldsymbol{m}}, \hat{S})$, is unique and known explicitly with parameters

$$\hat{\boldsymbol{m}} = \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{x}_i, \qquad\qquad \hat{S} = \frac{1}{n}\sum_{i=1}^{n} \left(\boldsymbol{x}_i - \hat{\boldsymbol{m}}\right)\left(\boldsymbol{x}_i - \hat{\boldsymbol{m}}\right)^T.$$

One could find a decision threshold for a test sample $\boldsymbol{x}_*$ via generating a user-specified number of vectors from this estimated density (Monte Carlo simulation). These simulated vectors yield a distribution of density evaluations, which allows to choose a threshold θ based on the preferred *false positive rate.*

The major flaws of this approach are that (a) it is a multivariate test for the intrinsically one-dimensional decision $f(\boldsymbol{x}_*) \gtreqless \theta$ and (b) the generating function of the simulated samples depends on the parameter estimations $\hat{\boldsymbol{m}}$, $\hat{S}$. These are random variables themselves and their prediction quality is significantly impaired for small values of $n = |X|$:

$$\hat{\boldsymbol{m}} \sim \mathcal{N}(\boldsymbol{m}, \frac{1}{n}S), \qquad\qquad n\hat{S} \sim \mathcal{W}_{(n-1)}(S), \tag{1}$$

where $\mathcal{W}_{(\nu)}$ denotes the Wishart distribution with ν degrees of freedom. In order to improve the Monte Carlo simulation, the distribution of the *squared Mahalanobis distance* $\hat{z}^2$ between a test point $\boldsymbol{x}_*$ and the sample mean $\hat{\boldsymbol{m}}$

$$\hat{z}^2 = \left(\boldsymbol{x}_* - \hat{\boldsymbol{m}}\right)^T \hat{S}^{-1}\left(\boldsymbol{x}_* - \hat{\boldsymbol{m}}\right)$$

is considered. The following result from statistics is used [1]:
Suppose that $\boldsymbol{y} \sim \mathcal{N}(\boldsymbol{0}, aS)$ and $A \sim \mathcal{W}_\nu(S)$. Then:

$$\boldsymbol{y}^T A^{-1} \boldsymbol{y} \sim \frac{ad}{\nu - d + 1} F_{(d, \nu - d + 1)}$$

The F-distribution, named after R. A. Fisher [7], is the distribution of the quotient of two χ^2-distributed variables. The degrees of freedom of these χ^2-distributed variables are the parameters of the F-distribution.

Under the null hypothesis that $\boldsymbol{x}_*$ was generated by the same distribution as the training set, we have

$$\left(\boldsymbol{x}_* - \hat{\boldsymbol{m}}\right) \sim \mathcal{N}\left(0, \left(1 + \frac{1}{n}\right) S\right)$$

and therefore, eventually

$$\hat{z}^2 \sim \frac{(n+1)d}{n-d} F_{(d, n-d)}. \tag{2}$$

The statistical test based on this result is referred to as the F-test. It takes the uncertainty caused by the number of training samples n and the number of dimensions d into account. It is furthermore optimal in the sense that the distribution of $\hat{z}^2$ is independent of the estimated parameters $\hat{\boldsymbol{m}}$ and $\hat{S}$. Furthermore, since this imposes a univariate test, multidimensional calculations can be circumvented completely in the simulation phase for Gaussian density estimations.

When *GMM*s are used, however, certain multidimensional calculations are necessary. The position of a simulated point $\boldsymbol{x}_*$ determines all $\hat{z}_k^2$'s

$$\forall k \in [c] : \hat{z}_k^2 = \left(\boldsymbol{x}_* - \hat{\boldsymbol{m}}_k\right)^T \hat{S}_k^{-1}\left(\boldsymbol{x}_* - \hat{\boldsymbol{m}}_k\right), \tag{3}$$

which makes it necessary to compute the corresponding Mahalanobis distances $\hat{z}_i^2$, $i \in [c] \backslash \{k\}$ when $\hat{z}_k^2$ is generated. This can be done [6] via

$$\hat{z}_j^2 = \hat{z}_k^2 \left\| \hat{S}_j^{-\frac{1}{2}} \hat{S}_k^{\frac{1}{2}} \hat{\boldsymbol{v}}_k \right\|^2 + \left\| \hat{S}_j^{-\frac{1}{2}} \left(\hat{\boldsymbol{m}}_k - \hat{\boldsymbol{m}}_j\right) \right\|^2 + 2\sqrt{\hat{z}_k^2}\left(\hat{\boldsymbol{m}}_k - \hat{\boldsymbol{m}}_j\right)^T \hat{S}_j^{-1} \hat{S}_k^{\frac{1}{2}} \hat{\boldsymbol{v}}_k, \tag{4}$$

where $\hat{\boldsymbol{v}}_k$ is a randomly generated unit norm vector.

4 D_H approximation

In order to implement the information theoretic approach, it is necessary to compare a training model to an adapted model. This can be done via divergences. In [6] the well-known Kullback-Leibler divergence (D_{KL}) [12] is used, which is based on the concept of self-information introduced by Shannon [15]. We propose the Hellinger divergence

$$D_H(f||g) = 1 - \int_{\mathbb{R}^d} \sqrt{f \cdot g}\ \mathrm{d}\boldsymbol{x}, \tag{5}$$

to be implemented instead, which is defined for probability densities f, g. In contrast to D_{KL}, D_H is symmetric, which makes it easier to be interpreted as a distance-like measure. It is not to be mistaken as a metric, however, since it violates the triangle inequality.

Now, the goal is to derive an explicit formula of the divergence between a training model f fitted to X and an adapted model f^* fitted to $X \cup \{\boldsymbol{x}_*\}$. In case the densities are *GMM*s, the integral in Equation (5) is not analytically tractable, making approximations necessary for efficient computing. For *GMM*s the above equation translates to:

$$D_H(f||f^*) = 1 - \int_{\mathbb{R}^d} \sqrt{f \cdot f^*}\ \mathrm{d}\boldsymbol{x} = 1 - \int_{\mathbb{R}^d} \sqrt{\sum_{k=1}^{c} \hat{\pi}_k \mathcal{N}_k \cdot \sum_{r=1}^{c} \hat{\pi}_r^* \mathcal{N}_r^*}\ \mathrm{d}\boldsymbol{x}$$
$$\mathcal{N}_i := \mathcal{N}(\boldsymbol{x}|\hat{\boldsymbol{m}}_i, \hat{S}_i), \quad \mathcal{N}_i^* := \mathcal{N}(\boldsymbol{x}|\hat{\boldsymbol{m}}_i^*, \hat{S}_i^*)$$

The root of the sum is no further tractable analytically. *Jensen's Inequality* is used to obtain an upper bound:

$$1 - \int_{\mathbb{R}^d} \sqrt{\sum_{k=1}^{c} \hat{\pi}_k \mathcal{N}_k \cdot \sum_{r=1}^{c} \hat{\pi}_r^* \mathcal{N}_r^*}\ \mathrm{d}\boldsymbol{x} \leq\ 1 - \sum_{k=1}^{c}\sum_{r=1}^{c} \hat{\pi}_k \hat{\pi}_r^* \int_{\mathbb{R}^d} \sqrt{\mathcal{N}_k \cdot \mathcal{N}_r^*}\ \mathrm{d}\boldsymbol{x}$$

The remaining integral is analytically tractable [16]:

$$\int_{\mathbb{R}^d} \sqrt{\mathcal{N}_k \cdot \mathcal{N}_r^*}\ \mathrm{d}\boldsymbol{x} = \left(\frac{\exp\left(-(\hat{\boldsymbol{m}}_k - \hat{\boldsymbol{m}}_r^*)^T \left(\hat{S}_r^* + \hat{S}_k\right)^{-1} (\hat{\boldsymbol{m}}_k - \hat{\boldsymbol{m}}_r^*) \right)}{\left| \frac{1}{4} \left(\hat{S}_r^* + \hat{S}_k\right) \left(\hat{S}_r^{*-1} + \hat{S}_k^{-1}\right) \right|} \right)^{\frac{1}{4}} \tag{6}$$

This yields a formula for D_H that depends on $\boldsymbol{x}_*$ (implicitly) and the parameters of the model f. However, a framework based on this formula imposes questionable rejection regions, as discussed in [16]. A second approximation is done by neglecting the above term for $k \neq r$, leading to another upper bound for the Hellinger divergence:

$$D_H(f||f^*) \leq 1 - \sum_{k=1}^{c} \hat{\pi}_k \hat{\pi}_k^* \int_{\mathbb{R}^d} \sqrt{\mathcal{N}_k \cdot \mathcal{N}_k^*}\ \mathrm{d}\boldsymbol{x} \tag{7}$$

A similar approximation is done for D_{KL} in [6]. The last approximation furthermore allows the following step: In order to incorporate the F-test it is necessary to reformulate the above formula in terms of $\hat{z}_1^2, \dots, \hat{z}_c^2$ instead of $\boldsymbol{x}_*$. The detailed calculation is rather intricate, applying the *Sherman-Morrison formula* and using a certain reasoning from [6] to explicitly evaluate the determinant in the denominator on the right-hand side of (6). It can be found in [16]. The resulting formula for the integral can then be expressed as:

$$\left(\int_{\mathbb{R}^d} \sqrt{\mathcal{N}_k \cdot \mathcal{N}_k^*}\ \mathrm{d}\boldsymbol{x} \right)^4 =$$
$$\exp\left(\frac{((n_k - n_k^*)u_{*k}z_k^2 - n_k^{*2})u_{*k}^2 z_k^2}{(n_k + n_k^* + u_{*k}z_k^2)n_k^{*3}} \right) \cdot \frac{(4n_k)^d n_k^{*(d-1)} (n_k^* + u_{*k}z_k^2)\,(n_k + n_k^*)^{2(1-d)}}{[(n_k + n_k^*)^2 + (2(n_k + n_k^*) + u_{*k}z_k^2)\,u_{*k}z_k^2]} \tag{8}$$

The notation from the end of Section 2 is used.

5 Framework & Experiments

Algorithm 1 Novelty detection framework for D_H

```
Require: training set $X$, #$GMM$ components $c$, #Monte Carlo samples $m$, false positive
  rate $r$, test sample $\boldsymbol{x}_*$
  Fit $GMM$ $f$ to $X$ with the $E/M$ algorithm

  /* Monte Carlo simulation */
  for i = 1, ..., c do
    generate $m\hat{\pi}_i$ values of $\hat{z}_i^2$ using (2)
    generate $m\hat{\pi}_i$ unit norm vectors $\boldsymbol{v}_i$
    $\forall j \in [c]\backslash\{i\}$ : calculate $\hat{z}_j^2$'s using (4)
  end for
  evaluate $D_H$ with (7) and (8) for each sample
  return (1-$r$)-th quantile $\theta$ of these evaluations

  evaluate $D_H(\boldsymbol{x}_*)$ using (3) and (8)
  if $D_H(\boldsymbol{x}_*) \geq \theta$ then
    $\boldsymbol{x}_*$ is outlier
  else
    $\boldsymbol{x}_*$ is normal
  end if
```

5.1 Artificial dataset

In order to illustrate the behavior of the approximation D_H of the Hellinger divergence, an artificial dataset is considered first. In this example 100 data points are drawn from $c = 3$ bivariate Gaussians (about 33 each), which resemble the letter "A" (Figure 1). A *GMM* is fitted to this data, and a Monte Carlo simulation according to the F-distribution from Section 3 is performed. This way, using the approximations from Section 4 for D_H and those from [6] for D_{KL}, decision thresholds for D_H and D_{KL} are obtained. The statistical test L_F from [16] is also considered. The *false positive rate* is chosen to be $fpr = 0.03$. The outlier decision is made for a $10^3 \times 10^3$ test grid of points over $[-5, 5]^2$, which is depicted in Figure 2. The main differences between the three methods are visible in areas where the decision borders of components overlap or are in close proximity.

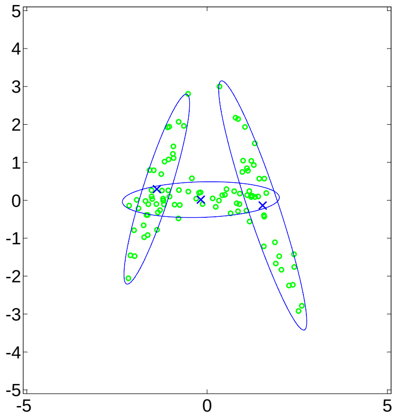
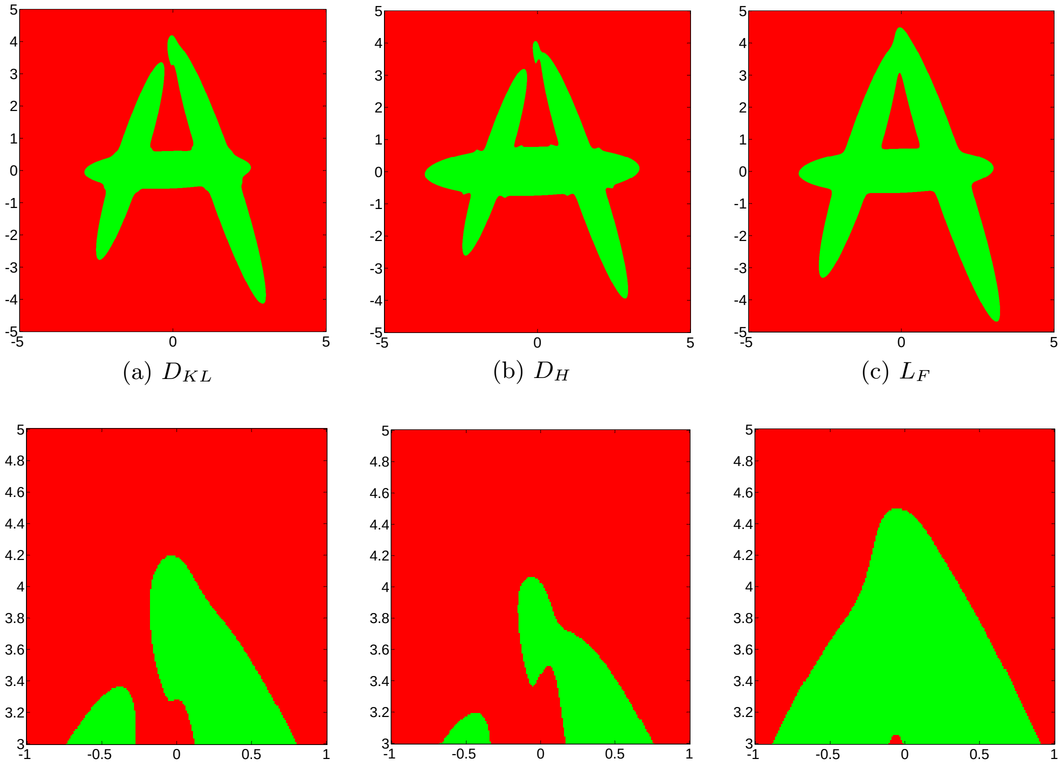


Figure 1: Artificial dataset with *GMM* estimation

5.2 Iris

The framework is validated on the well-known Iris dataset, which was introduced by R. A. Fisher in 1936 [8]. The data consists of 150 data samples (Iris plants) in 3 classes – "Iris setosa", "Iris versicolor" and "Iris virginica", each consisting of 50 samples. Each observation is made of $d = 4$ features – sepal length, sepal width, petal length and petal width. The classes "versicolor" and "virginica" are combined into one pool of 100 normal datasamples and the 50 "setosa" class members are considered anomalous. A sample of size



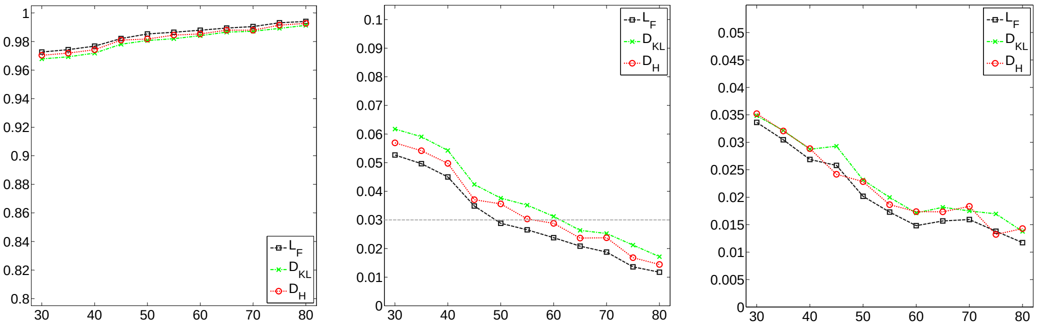
(a) D_{KL} (b) D_H (c) L_F

(d) D_{KL} excerpt (e) D_H excerpt (f) L_F excerpt

Figure 2: Decision maps for the artificial dataset, $fpr = 0.03$.

$n \in \{30, 35, \ldots, 80\}$ is randomly drawn from the normal data and the remaining $150 - n$ points serve as the test set of samples. The *GMM* is chosen to have $c = 2$ components and the fpr is chosen to be 0.03; 10^6 Monte Carlo samples are simulated to obtain the corresponding decision threshold. For each n, this experiment is repeated 100 times, the averages of the achieved accuracy and false positive rate are depicted in Figure 3. Three methods are considered: a purely statistical test L_F [16], as well as the above framework using the approximations of the Hellinger divergence D_H and the Kullback-Leibler divergence D_{KL} [6].

Other validation examples can be found in [16].



(a) average accuracy (b) average *false positive rate* (c) fpr standard deviation

Figure 3: Evaluation of the Iris experiment

6 Conclusion

With the formulas presented here it is possible to use the symmetric Hellinger divergence instead of its Kullback-Leiber counterpart in the framework from [6]. The measures D_H and D_{KL} perform similarly, whereas the former is easier to interpret as a distance-like measure. However, the performance of both methods is questioned by that of a purely statistical approach based on (2) – both in robustness and in control of the fpr. This is discussed in more detail in [16].

References

[1] T. W. Anderson. *An Introduction to Multivariate Statistical Analysis.* Wiley, New York, NY, 3rd edition, 2003.

[2] C. Archer, T. K. Leen, and A. Baptista. Parameterized novelty detection for environmental sensor monitoring. In *Advances in Neural Information Processing Systems 16*, pages 619–624, 2004.

[3] C. M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics).* Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

[4] R. J. Bolton and D. J. Hand. Unsupervised profiling methods for fraud detection. In *Proc. Credit Scoring and Credit Control VII*, pages 5–7, 2001.

[5] D. Clifton, L. Tarassenko, N. McGrogan, D. King, S. King, and P. Anuzis. Bayesian extreme value statistics for novelty detection in gas-turbine engines. In *Aerospace Conference, 2008 IEEE*, pages 1–11, March 2008.

[6] M. Filippone and G. Sanguinetti. Information theoretic novelty detection. Technical report, University of Sheffield, 2009. URL `http://www.dcs.gla.ac.uk/ maurizio /Publications/`.

[7] R. A. Fisher. The goodness of fit of regression formulae and the distribution of regression coefficients. *Journal of the Royal Statistical Society*, 85:597–612, 1922.

[8] R. A. Fisher. The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, 7(7):179–188, 1936.

[9] F. E. Grubbs. Procedures for detecting outlying observations in samples. *Technometrics*, 11(1):1–21, February 1969.

[10] E. Hellinger. *Neue Begründung der Theorie quadratischer Formen von unendlichvielen Veränderlichen.* 1909.

[11] J. Hollmn and V. Tresp. Call-based fraud detection in mobile communication networks using a hierarchical regime-switching model. In *Proceedings of the 1998 conference on Advances in neural information processing systems II*, pages 889–895. MIT Press, 1999.

[12] S. Kullback and R. A. Leibler. On information and sufficiency. *The Annals of Mathematical Statistics*, 22(1):79–86, 03 1951.

[13] A. Nairac, N. Townsend, R. Carr, S. King, P. Cowley, and L. Tarassenko. A system for the analysis of jet engine vibration data. *Integrated Computer-Aided Engineering*, 6(1): 53–66, Jan. 1999.

[14] J. Quinn and C. Williams. Known unknowns: Novelty detection in condition monitoring. In *Pattern Recognition and Image Analysis*, volume 4477 of *Lecture Notes in Computer Science*, pages 1–6. Springer Berlin Heidelberg, 2007.

[15] C. Shannon. A mathematical theory of communication. *Bell System Technical Journal*, 27:379–423, 623–656, July, October 1948.

[16] P. Stürmer. Hellinger divergence in information theoretic novelty detection. Master's thesis, Hochschule Mittweida, University of Applied Sciences, 2014.

[17] C. Williams, J. Quinn, and N. Mcintosh. Factorial switching kalman filters for condition monitoring in neonatal intensive care. In *Advances in Neural Information Processing Systems 18*, pages 1513–1520. MIT Press, 2006.

The new proposal of the calculation for the significance degree by once SOM learning

-using iris, gene, and other data-

H. TOKUTAKA [1)], M. OHKITA [1)], M. OYABU [2)], M. SENO [3)], M. OHKI [4)],

1) SOM Japan Inc., 2) Kanazawa Inst. of Tech., 3) Okayama Univ. 4) Tottori Univ.

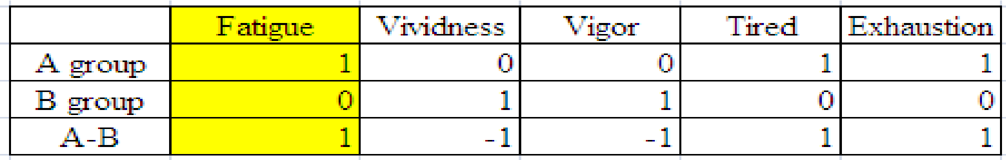
Abstract—**The significance degree of each component was calculated only by once SOM learning where the Spherical Self-Organizing-Map (SSOM) was used for the demonstration. In the method, kinds of specimens of the data are inserted in each column as each specimen. The method is first demonstrated using the iris data and then gene data. The method can be used for other data with successful results. The method can also be processed by the usual planar SOM.**

Keywords—Self organizing map; Significance degree;

I. Introduction

Here, the aim of the new proposal is described. Table 1 is the explanation of the concept of the significance degree. Relation between the fatigue and the other components about it is shown in Table 1. There is fatigue in the A group but there is none in the B group.

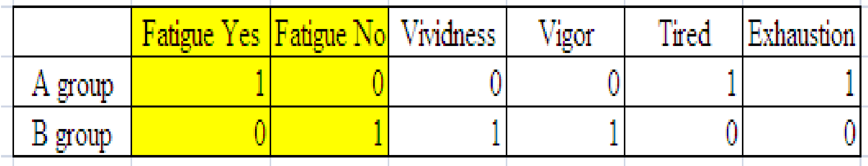
Table 1 Experimental condition for testing presence or absence of fatigue among other components.



	Fatigue	Vividness	Vigor	Tired	Exhaustion
A group	1	0	0	1	1
B group	0	1	1	0	0
A-B	1	-1	-1	1	1

This time, the way of allocating Table 2 is considered for being fatigued or not being fatigued like in the two dimensional space.

Table 2 Fatigue in Table 1 is separated by Yes and No



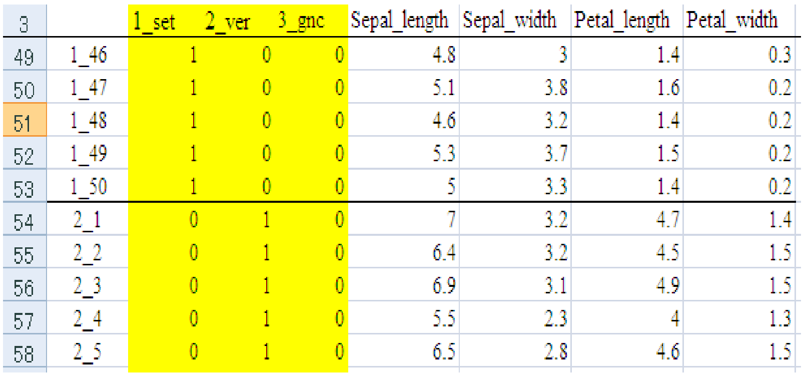
	Fatigue Yes	Fatigue No	Vividness	Vigor	Tired	Exhaustion
A group	1	0	0	0	1	1
B group	0	1	1	1	0	0

In the way in Table 2, the fatigue can be equally evaluated from the A group, as well as from the B group. In other words, if gathering the one where 1 stands in feeling fatigued, the significance degree of feeling fatigued is found from the each item. Also, if gathering the one where 1 stands feeling not fatigued, the significance degree of feeling not fatigued is found from the each item. This time, the proposed method is applied to the iris data which were used last time.

II. The algorithm of calculating the significance degree

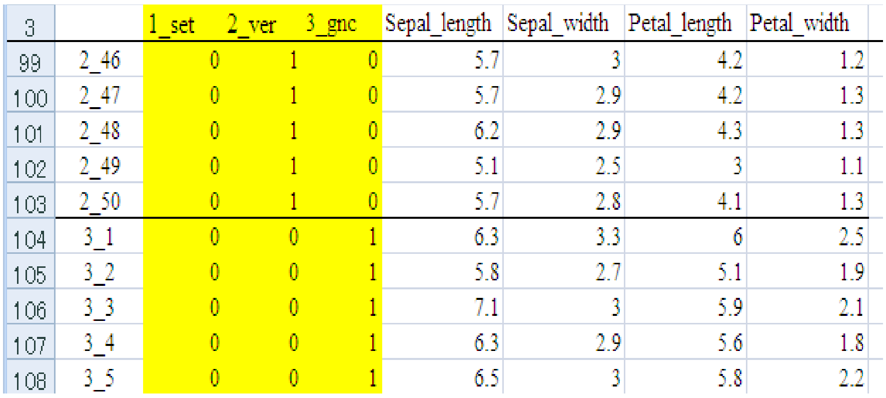
The contents are first described using iris data [1]. Using Spherical Self-Organizing Maps (SSOM) method [2,3,4] The significance degree calculation was previously introduced [5]. This time, the proposed method is as follows that the significance degree between each label pairs is computed by the once learning a lot of label data groups. First, the contents are described by the iris data. Next, the generality of the concerned method is described by using Gene and Tof-SIMS data [6]. The part of the original iris data is shown in tables 3 and 4. The data are composed of 1 (setosa) , 2 (versicolor) , and 3(virginica) , (each 50 stocks of 1, 2, and 3). A learning result is shown in Fig. 1.

Table 3 Part of the original iris data with 4 components. The classifications are added as setosa, versicolor and virginica by 3 columns



3		1_set	2_ver	3_gnc	Sepal_length	Sepal_width	Petal_length	Petal_width
49	1_46	1	0	0	4.8	3	1.4	0.3
50	1_47	1	0	0	5.1	3.8	1.6	0.2
51	1_48	1	0	0	4.6	3.2	1.4	0.2
52	1_49	1	0	0	5.3	3.7	1.5	0.2
53	1_50	1	0	0	5	3.3	1.4	0.2
54	2_1	0	1	0	7	3.2	4.7	1.4
55	2_2	0	1	0	6.4	3.2	4.5	1.5
56	2_3	0	1	0	6.9	3.1	4.9	1.5
57	2_4	0	1	0	5.5	2.3	4	1.3
58	2_5	0	1	0	6.5	2.8	4.6	1.5

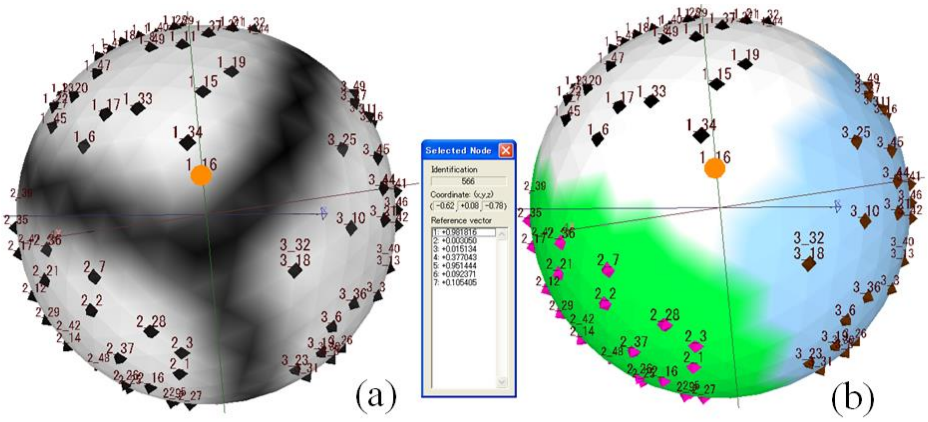
Table 4 The border between 2 (versicolor) and 3 (virginica)



3		1_set	2_ver	3_gnc	Sepal_length	Sepal_width	Petal_length	Petal_width
99	2_46	0	1	0	5.7	3	4.2	1.2
100	2_47	0	1	0	5.7	2.9	4.2	1.3
101	2_48	0	1	0	6.2	2.9	4.3	1.3
102	2_49	0	1	0	5.1	2.5	3	1.1
103	2_50	0	1	0	5.7	2.8	4.1	1.3
104	3_1	0	0	1	6.3	3.3	6	2.5
105	3_2	0	0	1	5.8	2.7	5.1	1.9
106	3_3	0	0	1	7.1	3	5.9	2.1
107	3_4	0	0	1	6.3	2.9	5.6	1.8
108	3_5	0	0	1	6.5	3	5.8	2.2

The raw data are normalized in the column wise direction. As an example, when the comparison between the classification 1_set and 2_ver is carried out, 3_gnc in Table 3 must be erased. Leave the label, 1 and 2 of each 50 stocks, 100 stocks in amount as shown in Table 8. Then, continue the calculation.

The result is shown in Fig.1. (a) is the U-matrix display which shows the distances between the nodes. (b) is the color display where the boundaries among the clusters of 1, 2, and 3 are clearly classified. The red mark in (b) on the spherical surface is the one of iris classification 1(setosa) where the minimum value (0.981816) is shown. Also, 2_11 in classification 2 and 3_20, in classification 3, shows the minimum value where label code books have minimum value of 0.991068 in 2_11 and 0.987421 in 3_20. They are on the other side of the sphere.



Selected Node

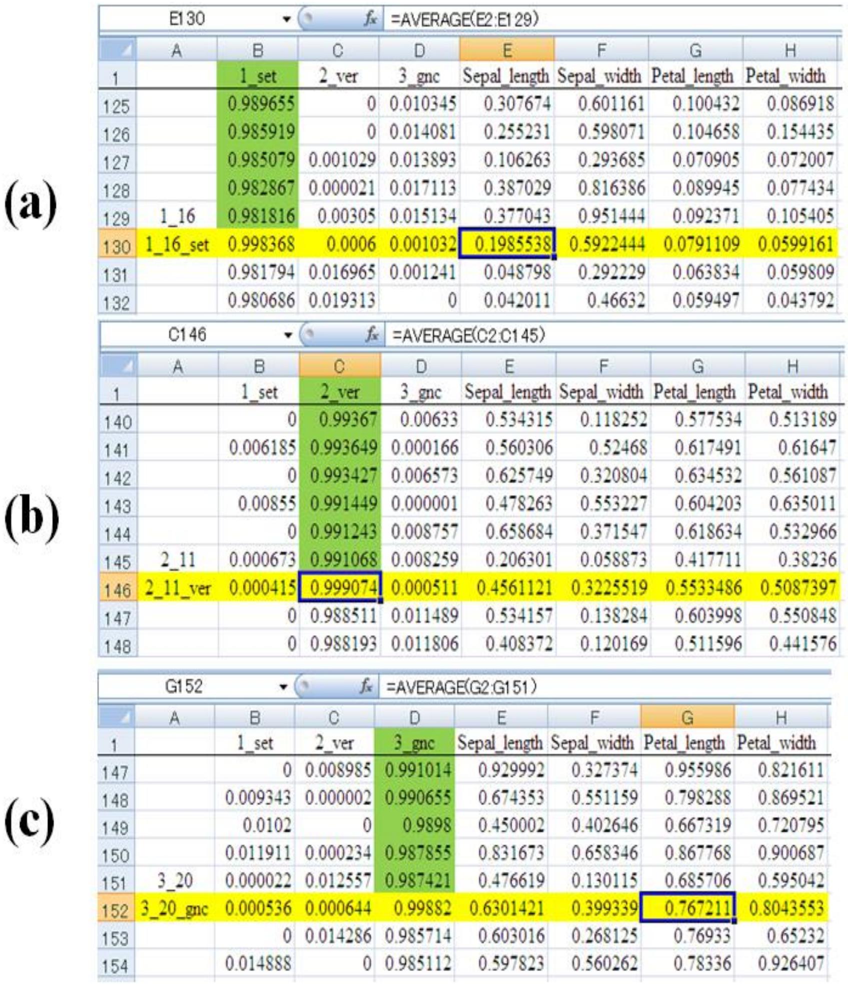
Identification
566

Coordinate: (x,y,z)
(-0.62, +0.08, -0.78)

Reference vector
1: +0.981816
2: +0.003050
3: +0.015134
4: +0.377043
5: +0.951444
6: +0.092371
7: +0.105405

Fig.1 (a) The U- matrix display at Griff value (0). (b) The boundary in coloring display of 1_set, 2_ver, and 3_gnc region.

Table 5 Descending sort of code book vector regarding to class identifiers



E130 =AVERAGE(E2:E129)

	A	B	C	D	E	F	G	H
1		1_set	2_ver	3_gnc	Sepal_length	Sepal_width	Petal_length	Petal_width
125		0.989655	0	0.010345	0.307674	0.601161	0.100432	0.086918
126		0.985919	0	0.014081	0.255231	0.598071	0.104658	0.154435
127		0.985079	0.001029	0.013893	0.106263	0.293685	0.070905	0.072007
128		0.982867	0.000021	0.017113	0.387029	0.816386	0.089945	0.077434
129	1_16	0.981816	0.00305	0.015134	0.377043	0.951444	0.092371	0.105405
130	1_16_set	0.998368	0.0006	0.001032	0.1985538	0.5922444	0.0791109	0.0599161
131		0.981794	0.016965	0.001241	0.048798	0.292229	0.063834	0.059809
132		0.980686	0.019313	0	0.042011	0.46632	0.059497	0.043792

C146 =AVERAGE(C2:C145)

	A	B	C	D	E	F	G	H
1		1_set	2_ver	3_gnc	Sepal_length	Sepal_width	Petal_length	Petal_width
140		0	0.99367	0.00633	0.534315	0.118252	0.577534	0.513189
141		0.006185	0.993649	0.000166	0.560306	0.52468	0.617491	0.61647
142		0	0.993427	0.006573	0.625749	0.320804	0.634532	0.561087
143		0.00855	0.991449	0.000001	0.478263	0.553227	0.604203	0.635011
144		0	0.991243	0.008757	0.658684	0.371547	0.618634	0.532966
145	2_11	0.000673	0.991068	0.008259	0.206301	0.058873	0.417711	0.38236
146	2_11_ver	0.000415	0.999074	0.000511	0.4561121	0.3225519	0.5533486	0.5087397
147		0	0.988511	0.011489	0.534157	0.138284	0.603998	0.550848
148		0	0.988193	0.011806	0.408372	0.120169	0.511596	0.441576

G152 =AVERAGE(G2:G151)

	A	B	C	D	E	F	G	H
1		1_set	2_ver	3_gnc	Sepal_length	Sepal_width	Petal_length	Petal_width
147		0	0.008985	0.991014	0.929992	0.327374	0.955986	0.821611
148		0.009343	0.000002	0.990655	0.674353	0.551159	0.798288	0.869521
149		0.0102	0	0.9898	0.450002	0.402646	0.667319	0.720795
150		0.011911	0.000234	0.987855	0.831673	0.658346	0.867768	0.900687
151	3_20	0.000022	0.012557	0.987421	0.476619	0.130115	0.685706	0.595042
152	3_20_gnc	0.000536	0.000644	0.99882	0.6301421	0.399339	0.767211	0.8043553
153		0	0.014286	0.985714	0.603016	0.268125	0.76933	0.65232
154		0.014888	0	0.985112	0.597823	0.560262	0.78336	0.926407

Data processing is as follows. (a) Arrange the table in the descending order of a B column of 1_set in the code book vector of 642 nodes. Then, search the minimum value of 1_16. Above it, compute the average and the result in the 130th line. (b) Next, carry the same procedure of a) in C column of 2_ver And search the minimum code book label of 2_11. (c) In the same way, arrange and search the minimum value as 3_20 to the D column, in the descending order of 3_gnc. Then, computes the average values at 146th line, and 152th line, respectively.

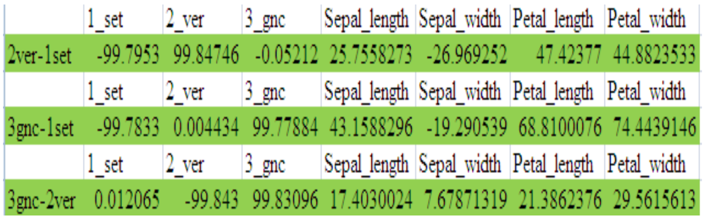
Table 6 shows the summary of Table 5. Data processing is as follows. Copy the 130th line in (a), 146th line in (b) and 152th line in (c), in the previous Table 5 and multiply them by 100 times.

Table 6 Summary of Table 5

	1_set	2_ver	3_gnc	Sepal_length	Sepal_width	Petal_length	Petal_width
1_16_set	99.83685	0.059979	0.103174	19.85538	59.224439	7.9110891	5.9916141
	1_set	2_ver	3_gnc	Sepal_length	Sepal_width	Petal_length	Petal_width
2_11_ver	0.0415	99.90744	0.051059	45.611207	32.255187	55.334859	50.873967
	1_set	2_ver	3_gnc	Sepal_length	Sepal_width	Petal_length	Petal_width
3_20_gnc	0.053565	0.064413	99.88202	63.014209	39.9339	76.721097	80.435529

Table 7 shows the differences of 2ver-1set, 3gnc-1set and 2ver-3gnc. Those were calculated using the averaged values with each line, based on Table 6. The direction of the subtraction is for the comparison between the conventional method and the proposed method for the 2 clusters.

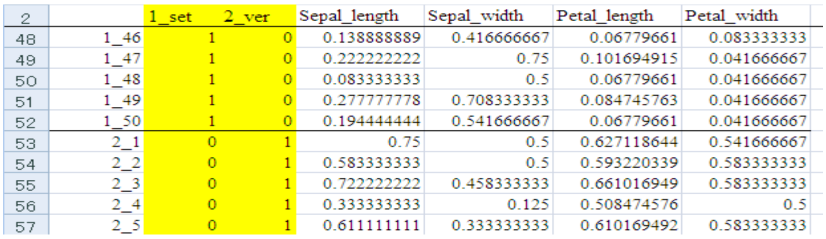
Table 7 The differences of 2ver-1set, 3gnc-1set and 2ver-3gnc



	1_set	2_ver	3_gnc	Sepal_length	Sepal_width	Petal_length	Petal_width
2ver-1set	-99.7953	99.84746	-0.05212	25.7558273	-26.969252	47.42377	44.8823533
	1_set	2_ver	3_gnc	Sepal_length	Sepal_width	Petal_length	Petal_width
3gnc-1set	-99.7833	0.004434	99.77884	43.1588296	-19.290539	68.8100076	74.4439146
	1_set	2_ver	3_gnc	Sepal_length	Sepal_width	Petal_length	Petal_width
3gnc-2ver	0.012065	-99.843	99.83096	17.4030024	7.67871319	21.3862376	29.5615613

Next, normalization of the iris raw data of Table 3 and 4 was carried out. Those are shown in Table 8. Normalization data of only setosa 1 and versicolor 2, with 100 stocks are used.

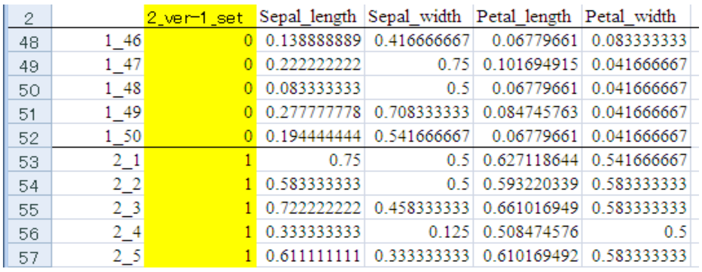
Table 8 Normalization of the iris data of 1_set and 2_ver (yellow columns)



2		1_set	2_ver	Sepal_length	Sepal_width	Petal_length	Petal_width
48	1_46	1	0	0.138888889	0.416666667	0.06779661	0.083333333
49	1_47	1	0	0.222222222	0.75	0.101694915	0.041666667
50	1_48	1	0	0.083333333	0.5	0.06779661	0.041666667
51	1_49	1	0	0.277777778	0.708333333	0.084745763	0.041666667
52	1_50	1	0	0.194444444	0.541666667	0.06779661	0.041666667
53	2_1	0	1	0.75	0.5	0.627118644	0.541666667
54	2_2	0	1	0.583333333	0.5	0.593220339	0.583333333
55	2_3	0	1	0.722222222	0.458333333	0.661016949	0.583333333
56	2_4	0	1	0.333333333	0.125	0.508474576	0.5
57	2_5	0	1	0.611111111	0.333333333	0.610169492	0.583333333

Table 9 shows the same data of Table 8. The classification column is different as the class identifier allocated to one column.

Table 9 Same data of Table 8. 1 dentifier column is different



2		2_ver-1_set	Sepal_length	Sepal_width	Petal_length	Petal_width
48	1_46	0	0.138888889	0.416666667	0.06779661	0.083333333
49	1_47	0	0.222222222	0.75	0.101694915	0.041666667
50	1_48	0	0.083333333	0.5	0.06779661	0.041666667
51	1_49	0	0.277777778	0.708333333	0.084745763	0.041666667
52	1_50	0	0.194444444	0.541666667	0.06779661	0.041666667
53	2_1	1	0.75	0.5	0.627118644	0.541666667
54	2_2	1	0.583333333	0.5	0.593220339	0.583333333
55	2_3	1	0.722222222	0.458333333	0.661016949	0.583333333
56	2_4	1	0.333333333	0.125	0.508474576	0.5
57	2_5	1	0.611111111	0.333333333	0.610169492	0.583333333

The results of the proposed method are shown in Fig. 2. In Fig. 2(b), the proposed method is applied only to two cases of Table 9.

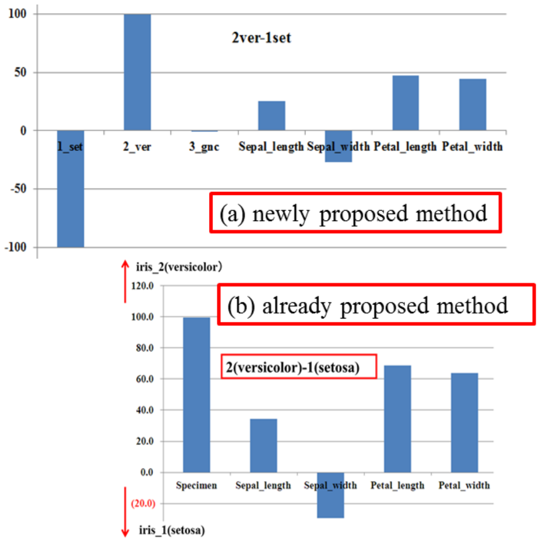


Fig. 2 (a) The significance degree among versicolor : setosa calculated using Tables 3 and 4 (b) using Table 9

Fig.3 is moreover compared adding with the conventional method in Table 9. Hereinafter, other combinations of the significance degree of virginica : versicolor and also of virginica : setosa are included. The comparison among the total 3 methods of the proposed 2 methods and a conventional method is shown for these 3 cases in Fig. 3. The significance degree are calculated and compared among 2ver-1set of vericolor: setosa, 3gnc-1set of versinica : setosa, and 3gnc-2ver of virginica : versicolor. The significance degree among gnc-ver of virginica : versicolor doesn't agree only a little as shown in the figure.

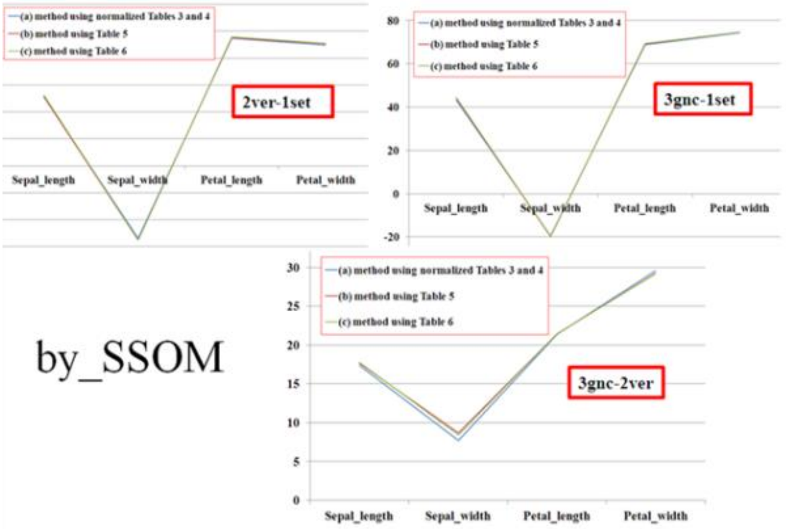


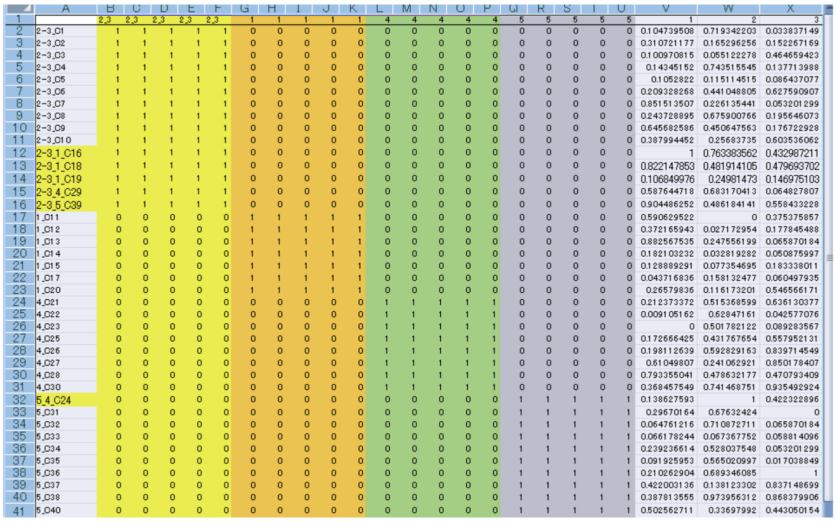
Fig. 3 By 3 methods of (a) using Tables 3 and 4, (b) Table 8, and (c) Table 9 in each figure

In case of the iris data, in the first, the data of three kinds of the total 150 stocks were normalized in the column direction. In the case of the computations for 2 methods, this original normalized data were used and computed. In this case, all 3 methods approximately fully agreed as shown in Fig. 3.

III. The Application to the Gene data

Here, we have the data where the relation between the breast cancer and the gene are examined, respectively. Breast cancer is classified into 4 steps to the breast cancer level 5, 4, 2-3 from 1 of the health. Here, 1 of the health and 5 of the breast cancer are compared in the significance degree. 40 cases are classified into 1-5 level of the cancer by the doctor. Each of the gene of 40 cases was named in 1-831. Further, the level 2 and 3 are gathered altogether as 2-3 for simplicity. By the method which is described in the Tables 3 and 4, the flags of 4 steps of 1, 2-3, 4, and 5 were put up as shown in Table 10. The data of Table 10 was studied by the Spherical SOM. Incidentally, the gene data was normalized at the maximum-the minimum of the gene to 1-831 in each sample. This is the so-called line normalization. The number of component is large as 831. Therefore, the number of the flags was set as the weight. Total 20 columns in amount are the flags of 4 clusters with five lines of each cluster as shown in Table 10. Four kinds of the label of 1, 2-3, 4, and 5 are used as classification. Learning data and a learning result are shown in Table 10, and Fig. 4, respectively.

Table 10 The learning data normalized at the line (row) direction



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
1		2_3	2_3	2_3	2_3	2_3	1	1	1	1	1	4	4	4	4	4	5	5	5	5	5	1	2	3
2	2-3_C1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.104739508	0.719342203	0.033837149
3	2-3_C2	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.310721177	0.165296256	0.152267169
4	2-3_C3	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.100970815	0.055122278	0.464659423
5	2-3_C4	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.14345152	0.743515545	0.137713988
6	2-3_C5	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.1052822	0.115114515	0.086437077
7	2-3_C6	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.209328268	0.441048805	0.627590907
8	2-3_C7	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.851513507	0.226135441	0.053201299
9	2-3_C8	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.243728895	0.675900766	0.195646073
10	2-3_C9	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.645682586	0.450647563	0.176722928
11	2-3_C10	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.387994452	0.25683735	0.603536062
12	2-3_1_C16	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0.763383562	0.432987211
13	2-3_1_C18	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.822147853	0.481914105	0.479693702
14	2-3_1_C19	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.106849976	0.24981473	0.146975103
15	2-3_4_C29	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.587644718	0.683170413	0.064827807
16	2-3_5_C39	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.904486252	0.486184141	0.558433228
17	1_C11	0	0	0	0	0	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0.590629522	0	0.375375857
18	1_C12	0	0	0	0	0	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0.372165943	0.027172954	0.177845488
19	1_C13	0	0	0	0	0	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0.882567535	0.247556199	0.065870184
20	1_C14	0	0	0	0	0	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0.182103232	0.032819282	0.050875997
21	1_C15	0	0	0	0	0	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0.128889291	0.077354695	0.183338011
22	1_C17	0	0	0	0	0	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0.043716836	0.158132477	0.060497935
23	1_C20	0	0	0	0	0	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0.26579836	0.116173201	0.546566171
24	4_C21	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	0	0	0	0	0	0.212373372	0.515368599	0.636130377
25	4_C22	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	0	0	0	0	0	0.009105162	0.62847161	0.042577076
26	4_C23	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	0	0	0	0	0	0	0.501782122	0.089283567
27	4_C25	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	0	0	0	0	0	0.172666425	0.431767654	0.557952131
28	4_C26	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	0	0	0	0	0	0.198112639	0.592829163	0.839714549
29	4_C27	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	0	0	0	0	0	0.61049807	0.241062921	0.850178407
30	4_C28	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	0	0	0	0	0	0.793355041	0.478632177	0.470793409
31	4_C30	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	0	0	0	0	0	0.368457549	0.741468751	0.935492924
32	5_4_C24	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	0.138627593	1	0.422322896
33	5_C31	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	0.29670164	0.67632424	0
34	5_C32	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	0.064761216	0.710872711	0.065870184
35	5_C33	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	0.066178244	0.067367752	0.058814096
36	5_C34	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	0.239236614	0.528037548	0.053201299
37	5_C35	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	0.091925953	0.565020997	0.017038849
38	5_C36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	0.210262904	0.689346085	1
39	5_C37	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	0.422003136	0.138123302	0.837148699
40	5_C38	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	0.387813555	0.973956312	0.868379906
41	5_C40	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	0.502562711	0.33697992	0.443050154

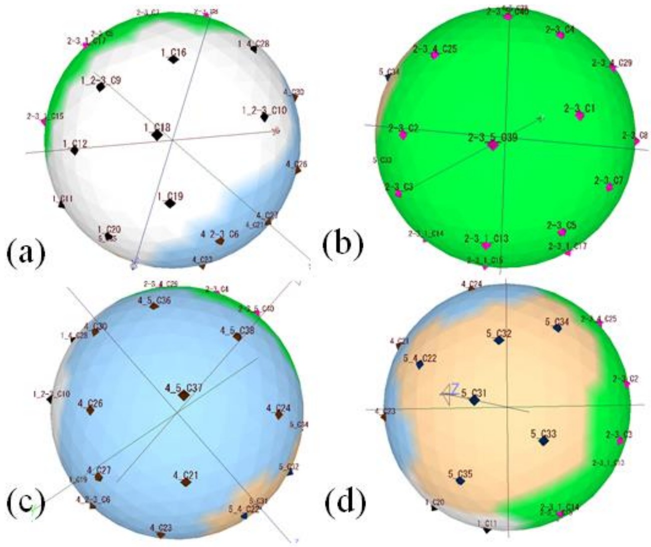
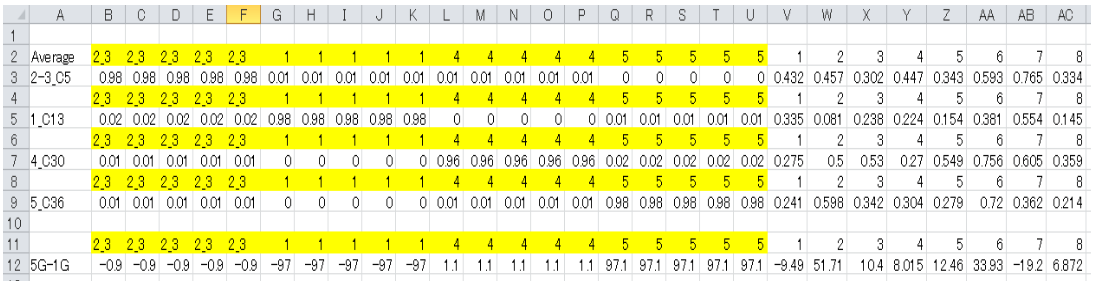


Fig. 4 The SOM learning result using the data in Table 10. (a) 1 cluster, (b) 2-3 cluster, (c) 4cluster, and (d) 5 cluster

Using the learning codebook vector of Fig. 4, the calculation of the same as Tables 5 and 6 was carried out. The average of each cluster was calculated. The 5th line was subtracted from the 9th line. The significance degree of 5-1 is tabled on the 12th line. The colored flags of the clusters at the 12th line were deleted here. Only the significance degree of the gene was graphed. The results are shown in Table 11.

Table 11 The significance degree of 5-1



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	AA	AB	AC
1																													
2	Average	2_3	2_3	2_3	2_3	2_3	1	1	1	1	1	4	4	4	4	4	5	5	5	5	5	1	2	3	4	5	6	7	8
3	2-3_C5	0.98	0.98	0.98	0.98	0.98	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0	0	0	0	0	0.432	0.457	0.302	0.447	0.343	0.593	0.765	0.334
4		2_3	2_3	2_3	2_3	2_3	1	1	1	1	1	4	4	4	4	4	5	5	5	5	5	1	2	3	4	5	6	7	8
5	1_C13	0.02	0.02	0.02	0.02	0.02	0.98	0.98	0.98	0.98	0.98	0	0	0	0	0	0.01	0.01	0.01	0.01	0.01	0.335	0.081	0.238	0.224	0.154	0.381	0.554	0.145
6		2_3	2_3	2_3	2_3	2_3	1	1	1	1	1	4	4	4	4	4	5	5	5	5	5	1	2	3	4	5	6	7	8
7	4_C30	0.01	0.01	0.01	0.01	0.01	0	0	0	0	0	0.96	0.96	0.96	0.96	0.96	0.02	0.02	0.02	0.02	0.02	0.275	0.5	0.53	0.27	0.549	0.756	0.605	0.359
8		2_3	2_3	2_3	2_3	2_3	1	1	1	1	1	4	4	4	4	4	5	5	5	5	5	1	2	3	4	5	6	7	8
9	5_C36	0.01	0.01	0.01	0.01	0.01	0	0	0	0	0	0.01	0.01	0.01	0.01	0.01	0.98	0.98	0.98	0.98	0.98	0.241	0.598	0.342	0.304	0.279	0.72	0.362	0.214
10																													
11		2_3	2_3	2_3	2_3	2_3	1	1	1	1	1	4	4	4	4	4	5	5	5	5	5	1	2	3	4	5	6	7	8
12	5G-1G	-0.9	-0.9	-0.9	-0.9	-0.9	-97	-97	-97	-97	-97	1.1	1.1	1.1	1.1	1.1	97.1	97.1	97.1	97.1	97.1	-9.49	51.71	10.4	8.015	12.46	33.93	-19.2	6.872

Only the part of the gene with the 12th line of Table 11 was graphed in Fig. 5. In Fig. 5(a), To the 1-150th gene and skipping a little interval, (b) To 301-450th gene. + side is intentional gene with 5, the breast cancer. - side is the healthily intentional gene with 1 cluster.

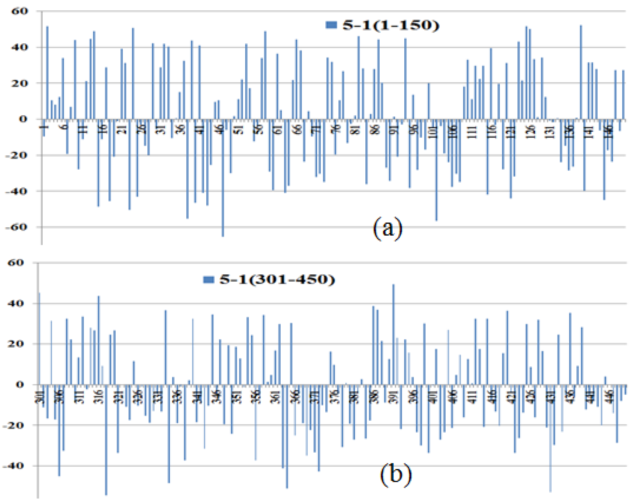


Fig. 5 Graph of significance degree of Table 8. (a) to 1-150th gene, (b) to 301-450th gene

The characteristics of the significance degree between 5-1 can be seen from Figs. 5 and 6. It has begun with Fig. 5(a) already, but the bigger the gene number becomes, the smaller significance degree becomes on the healthy 1 side. Specifically, in figure 6(b), most peaks are on the 5 sides of the breast cancer.

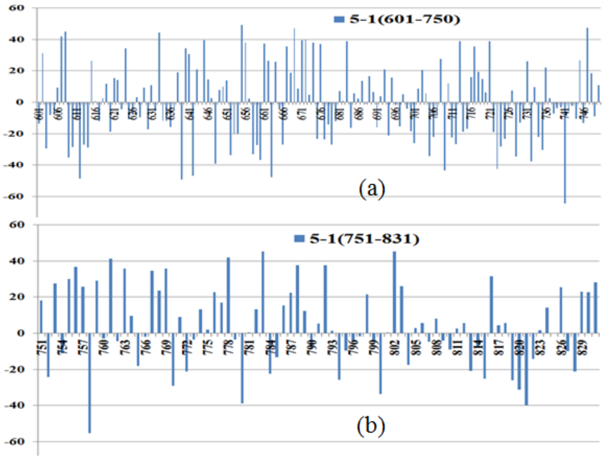
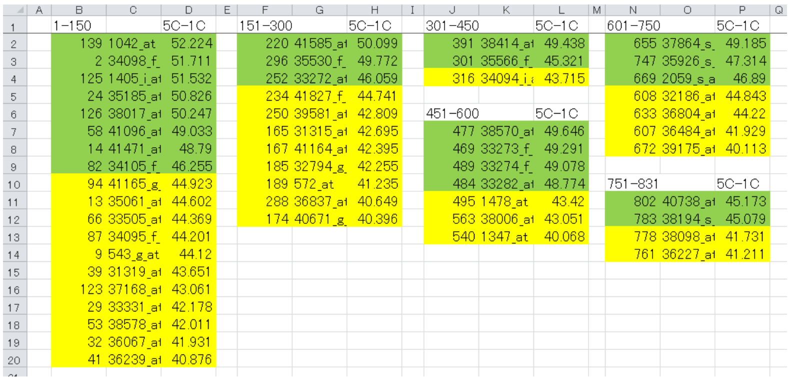


Fig. 6 Signficance degree. (a) to 601-750th gene, (b) to 751-831th gene. A little interval is skipped.

Table 12 shows the significance degree on the 5 side of the breast cancer equal to or more than 40 degree was chosen in each gene number range. The number of the higher significance degree is especially high in 1-150th gene number range. The green color range shows equal to or more than 45.

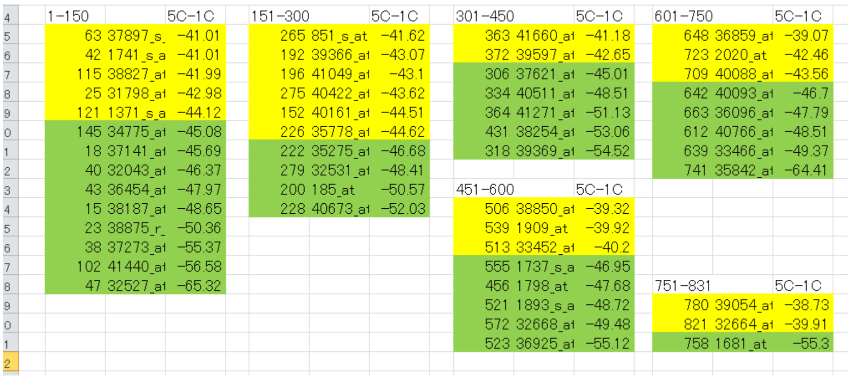
Table 12 The significance degree on the 5 side of the breast cancer



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1		1−150		5C−1C		151−300		5C−1C		301−450		5C−1C		601−750		5C−1C	
2		139	1042_at	52.224		220	41585_at	50.099		391	38414_at	49.438		655	37864_s_	49.185	
3		2	34098_f_	51.711		296	35530_f_	49.772		301	35566_f_	45.321		747	35926_s_	47.314	
4		125	1405_i_at	51.532		252	33272_at	46.059		316	34094_i_a	43.715		669	2059_s_a	46.89	
5		24	35185_at	50.826		234	41827_f_	44.741						608	32186_at	44.843	
6		126	38017_at	50.247		250	39581_at	42.809		451−600		5C−1C		633	36804_at	44.22	
7		58	41096_at	49.033		165	31315_at	42.695		477	38570_at	49.646		607	36484_at	41.929	
8		14	41471_at	48.79		167	41164_at	42.395		469	33273_f_	49.291		672	39175_at	40.113	
9		82	34105_f_	46.255		185	32794_g_	42.255		489	33274_f_	49.078					
10		94	41165_g_	44.923		189	572_at	41.235		484	33282_at	48.774		751−831		5C−1C	
11		13	35061_at	44.602		288	36837_at	40.649		495	1478_at	43.42		802	40738_at	45.173	
12		66	33505_at	44.369		174	40671_g_	40.396		563	38006_at	43.051		783	38194_s_	45.079	
13		87	34095_f_	44.201						540	1347_at	40.068		778	38098_at	41.731	
14		9	543_g_at	44.12										761	36227_at	41.211	
15		39	31319_at	43.651													
16		123	37168_at	43.061													
17		29	33331_at	42.178													
18		53	38578_at	42.011													
19		32	36067_at	41.931													
20		41	36239_at	40.876													

Next, the side of 1 health is shown in Table 13. In the range of 1-151, the number of the genes which have the absolute value of the significance degree equal to or more than -40 is more than on the side above of 5 breast cancer (cf -65). Equal to or less than 40 genes are equally distributed approximately compared with Table 12. On the significance degree for the side of the health, it can be seen that the high significance gene is the 47th and 741st gene from the table.

Table 13 Significance degree of the side of 1 health



4		1−150		5C−1C		151−300		5C−1C		301−450		5C−1C		601−750		5C−1C
5		63	37897_s_	−41.01		265	851_s_at	−41.62		363	41660_at	−41.18		648	36859_at	−39.07
6		42	1741_s_a	−41.01		192	39366_at	−43.07		372	39597_at	−42.65		723	2020_at	−42.46
7		115	38827_at	−41.99		196	41049_at	−43.1		306	37621_at	−45.01		709	40088_at	−43.56
8		25	31798_at	−42.98		275	40422_at	−43.62		334	40511_at	−48.51		642	40093_at	−46.7
9		121	1371_s_a	−44.12		152	40161_at	−44.51		364	41271_at	−51.13		663	36096_at	−47.79
0		145	34775_at	−45.08		226	35778_at	−44.62		431	38254_at	−53.06		612	40766_at	−48.51
1		18	37141_at	−45.69		222	35275_at	−46.68		318	39369_at	−54.52		639	33466_at	−49.37
2		40	32043_at	−46.37		279	32531_at	−48.41						741	35842_at	−64.41
3		43	36454_at	−47.97		200	185_at	−50.57		451−600		5C−1C				
4		15	38187_at	−48.65		228	40673_at	−52.03		506	38850_at	−39.32				
5		23	38875_r_	−50.36						539	1909_at	−39.92				
6		38	37273_at	−55.37						513	33452_at	−40.2				
7		102	41440_at	−56.58						555	1737_s_a	−46.95				
8		47	32527_at	−65.32						456	1798_at	−47.68		751−831		5C−1C
9										521	1893_s_a	−48.72		780	39054_at	−38.73
0										572	32668_at	−49.48		821	32664_at	−39.91
1										523	36925_at	−55.12		758	1681_at	−55.3
2																

Tables 12 and 13 are generalized. The genetic code which shows the characteristic of the gene is next to the gene number. For the gene of the higher strength of the green color, the highest value for the 1 health side shows -65. In other words, the healthy gene has the clear characteristics. However, there are many, too, numbers of the manifestation in the genes on the 5 side of the breast cancer. The strength is uniformly, too. The gene comparison between the breast cancer and the health, was evaluated using the method of this significance degree. As for the manifestation gene of the breast cancer, both the manifestation quantity and the number of the manifestation are high compared with the side of the health. In this way, it is possible to diagnose breast cancer from the genetic code.

Incidentally, it was examined the breast cancer of which level the strange sample is in. By considering in which area of 4 areas of Fig. 4 there is a sample of strange (UK), the cancer level of the sample can be distinguished.

IV. Conclusions

Thus, a new method for calculating the significance degree was proposed using the Spherical Self-Organizing Maps (SSOM). It was verified by the iris 3 data. The procedure is of leaving the same flag which is equal to each label of 3 kinds of data. In this way, each label (the classification) could be equally compared. When the significance degree which was sought in this method of using all 3 data and the significance degree which were sought in two combinations, were compared, the results reasonably agreed as shown in Fig. 3. Also, using this method of the significance degree, the gene comparison between the breast cancer (stage 5) and the health (stage 1), was evaluated. As for the breast cancer, both the manifestation quantity and the number of the manifestation of the gene are high compared with the side of the health. Thus, it is possible to diagnose breast cancer from the gene examination. The iris, and the gene data are the examples which the human being classified a cluster by some procedure. However, in case of the Tof-SIMS data of the unknown clustering, it was automatically classified using Spherical SOM. However, the results of the analysis for the TOF-SIMS data are deleted due to insufficient space.

As the conclusion, a general procedure is described as follows:

1. Any data can be learned by the Spherical SOM.

2. The spherical surface was deformed considering the distance (U-matrix) among the learned nodes. Then, a classification is carried out and a dendrogram is constructed.

3. Here, the optional group to be analyzed can be chosen. 1 or 0 of the classification was assigned to the chosen group for the number of the classification like Tables 3, 4, and 8.

4. This data is once again, learned by the Spherical SOM method. The significance degree among two kinds of classification for each is evaluated by the procedure which is shown in Tables 5, 6, and 7.

Incidentally, a little, the precision falls however, it is possible to compute the significance degree by the plane SOM. When the classification becomes three kinds for example, in multiple regression [7], it cannot avoid allocating the classification with -1, 0, 1 (or 0, 1, 2). To the group which doesn't have an order, it is improper to allocate an order having to do with a number. It is possible to be solved if making the classification of three-dimensional as (1,0,0・・) as having proposed this time.

References

[1] http://www.ics.uci.edu/~mlearn/databases/

[2] H. Tokutaka, K. Fujimura, M. Ohkita: Cluster Analysis using Spherical SOM, in Japanese, Journal of Biomedical Fuzzy Systems Association, Vol. 8, No.1, pp.29-39, 2006.

[3] M. Ohkita, H. Tokutaka, K. Fujimura and E. Gonda: Self-Organizing Maps and the tool, in Japanese, Springer Japan Inc., 2008

[4] http://www.somj.com

[5] H. Tokutaka: The calculation of the significance degree among the data components of the various distinction data by the Spherical Self-Organizing Maps (SSOM) method, in Japanese, The 40th SASJ (Surface Analysis Society of Japan) seminar material, The Ota-Ku (Ward) industry plaza, Feb. 21st, 2013.

[6] K. Yoshihara, H. Tokutaka: The TOF-SIMS spectrum analysis by PCA and the Spherical SOM method, in Japanese, The 40th SASJ (Surface Analysis Society of Japan) seminar material, Nagoya Congress Center Japan, June 17th , 2013

[7] M. Nakano: The delightful multivariate analysis for nursing, health care and the medical care, in Japanese, The helicity publishing, Kobe, Japan, pp.1-212, 2009.

LEARNING ALGORITHMS FOR NON-METRIC SPACES - A SHORT SURVEY -

Frank-M. Schleif[1] and Peter Tino[1]
{f.schleif,p.tino}@cs.bham.ac.uk
[1]Department of Computer Science, University of Birmingham, UK

1. INTRODUCTION

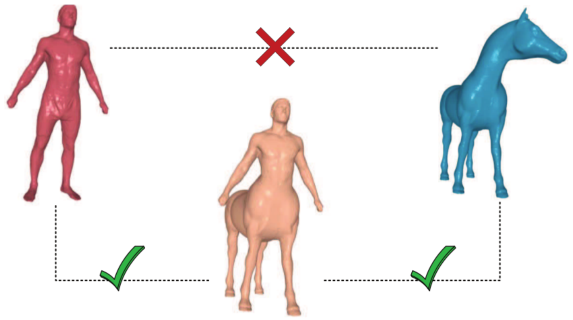
Machine learning is a key element in data analysis systems and often used to analyze

OLD *standard vectorial data with the Euclidean distance.*

With the advent of web and social-media data this has rapidly changed. Many data are now given as

NEW *non-standard or structured data with indefinite proximities*

by means of dedicated, often non-metric proximity measures. Due to the mathematical power and efficiency of the Euclidean space only few methods were proposed for non-Euclidean data and even less for non-metric data analysis. In this contribution we provide a short review of the few available algorithms and concepts for non-metric data analysis as available today.



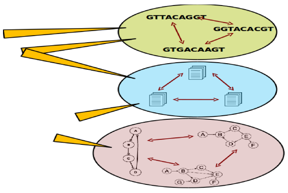
Non-metric data representation is daily life

So what?: most classical learning algorithms like the Support Vector Machine expect metric inputs (mercer kernels). If non-mercer or non-psd kernels are used the employed mathematical theory is not any longer valid and your preferred kernel method can easily **fail** , all guarantees (convergences) and bounds become **invalid** .

2. COMMON (NON-METRIC) SIMILARITY MEASURES

Non-metric proximities (similarities and dissimilarities) are frequent if domain specific measures are used. There is often not even an explicit vector space available.

- **alignment** (Bioinformatics)
- Levenstein (Textprocessing)
- Hamming (Information theory)
- Geodesic distance (Geometry)
- Jaccard index (Statistics)
- **Compression distance**
- dynamic time warping (time-series)



Other examples for indefinite proximities:

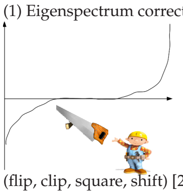
- Manhattan kernel: $K(\mathbf{x},\mathbf{x}') = -||\mathbf{x}-\mathbf{x}'||_1$
- indefinite sigmoid kernel $K(\mathbf{x},\mathbf{x}') = \tanh(a\langle\mathbf{x},\mathbf{x}'\rangle + r)$ (for some parameters a, r)
- Many divergence measures popular in the field of spectroscopy

Outcome: (non-)metric proximities with **negative** eigenvalues in the eigenspectrum.

3. CURRENT APPROACHES - MAKE IT GLOBAL PSD

Consider negative eigenvalues as noise - correct the eigenspectrum to psd.

(1) Eigenspectrum correction

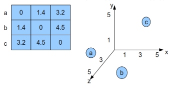


(flip, clip, square, shift) [20, 6]

(2) Proxy alignment (find a similar **psd** matrix)

[4, 10]

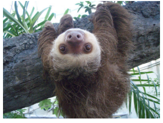
(3) Similarity space approach



a	0	1.4	3.2
b	1.4	0	4.5
c	3.2	4.5	0

Construct a vector space from the proximity matrix [17]

(4) Do nothing (can easily fail)



[19]

NON-METRIC SPACES

OLD In metric spaces similarities between two objects $\mathbf{x},\mathbf{w} \in X^D$ calculated as a mapping $\phi : \mathbf{x} \in \mathbf{X} \subseteq \mathbb{R} \mapsto \phi(\mathbf{x}) \in \mathbf{F}$ using the kernel trick [21], k: $\mathbf{X} \times \mathbf{X} \rightarrow \mathbf{F}$ with $k(\mathbf{x},\mathbf{x}') = \langle\phi(\mathbf{x}),\phi(\mathbf{x}')\rangle,\ \forall \mathbf{x},\mathbf{x}' \in \mathbf{X}$. Thereby it is assumed that the kernel function $k(\mathbf{x},\mathbf{x}')$ is positive semi definite (psd).

NEW For non-psd $k(\cdot,\cdot)$ a Krein space has to be used which for finite dimensions is a pseudo-Euclidean space (P_E). We can always embed K into P_E for symmetric *dissimilarities* with constant zero diagonal [9].

Definition 1 (Pseudo-Euclidean space [16]) *A pseudo-Euclidean space* (P_E) $\xi = \mathbb{R}^{(p,q)}$ *is a real vector space equipped with a non-degenerate, indefinite inner product* $\langle .,. \rangle_\xi$. ξ *admits a direct orthogonal decomposition* $\xi = \xi_+ \oplus \xi_-$ *where* $\xi_+ = \mathbb{R}^p$ *and* $\xi_- = \mathbb{R}^q$ *and the inner product is positive definite on* ξ_+ *and negative definite on* ξ_-.

A symmetric bilinear form in this space is given by

$$\langle \mathbf{x},\mathbf{y}\rangle_{p,q} = \sum_{i=1}^{p} x_i y_i - \sum_{i=p+1}^{p+q} x_i y_i = \mathbf{x}^\top \mathbf{I}_{p,q}\mathbf{y}$$

where $\mathbf{I}_{p,q}$ is a diagonal matrix with p entries 1 and q entries -1. The eigendecomposition of $\mathbf{S} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^\top$ provides a vectorial representation $\mathbf{V}$ in P_E:

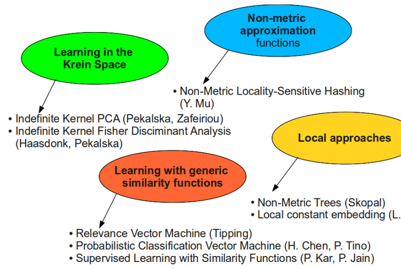
$$\mathbf{V} = \mathbf{U}_{p+q} \left|\mathbf{\Lambda}_{p+q}\right|^{1/2} \qquad (1)$$

For symmetric (non-)psd similarities with low intrinsic dimension this can be calculated **exact and in linear time** [20].

USAGE:

- use the decomposition to learn the model in $\mathcal{K}_+$ and $\mathcal{K}_-$ by late recombination [14],
- incredients of the model (e.g. scatter matrix) can be calculated on the decomposition [18, 24]
- just learn e.g. a generic regression function $f(x) = \sum_{i=1}^{N} w_i \phi_{i,\theta}(\mathbf{x}) + b$ which can related to the indefinite $k(\mathbf{x},\cdot)$ [11, 3]
- local metric adaptations [2, 5] to correct violated triangle inequalities.

LEARNING MODELS IN NON-METRIC SPACES



Non-metric approximation functions

- Non-Metric Locality-Sensitive Hashing (Y. Mu)

Learning in the Krein Space

- Indefinite Kernel PCA (Pekalska, Zafeiriou)
- Indefinite Kernel Fisher Disciminant Analysis (Haasdonk, Pekalska)

Local approaches

- Non-Metric Trees (Skopal)
- Local constant embedding (L. Chen)

Learning with generic similarity functions

- Relevance Vector Machine (Tipping)
- Probabilistic Classification Vector Machine (H. Chen, P. Tino)
- Supervised Learning with Similarity Functions (P. Kar, P. Jain)

4. SUMMARY

1. Learning in non-metric spaces is relevant [7, 16, 19, 13]
2. Only few approaches around - still often limited (theory, runtime, scalability, . . .
3. Specific effects of transformations not really understood [8, 12, 15]
4. First steps on establishing a theory on learning in non-metric spaces [1, 23]
5. Most approaches focus on supervised learning or retrieval

REFERENCES

References

[1] Maria-Florina Balcan, Avrim Blum, and Nathan Srebro. A theory of learning with similarity functions. *Machine Learning*, 72(1-2):89–112, 2008.
[2] Benjamin Bustos and Tomás Skopal. Non-metric similarity search problems in very large collections. In Serge Abiteboul, Klemens Böhm, Christoph Koch, and Kian-Lee Tan, editors, *ICDE*, pages 1362–1365. IEEE Computer Society, 2011.
[3] Huanhuan Chen, Peter Tino, and Xin Yao. Probabilistic classification vector machines. *IEEE Transactions on Neural Networks*, 20(6):901–914, 2009.
[4] J. Chen and J. Ye. Training svm with indefinite kernels. pages 136–143, 2008. cited By (since 1996)8.
[5] Lei Chen and Xiang Lian. Efficient similarity search in nonmetric spaces with local constant embedding. *IEEE Trans. Knowl. Data Eng.*, 20(3):321–336, 2008.
[6] Yihua Chen, Eric K. Garcia, Maya R. Gupta, Ali Rahimi, and Luca Cazzanti. Similarity-based classification: Concepts and algorithms. *JMLR*, 10:747–776, 2009.
[7] Robert P. W. Duin and Elzbieta Pekalska. Non-euclidean dissimilarities: Causes and informativeness. In *Structural, Syntactic, and Statistical Pattern Recognition, Joint IAPR International Workshop, SSPR&SPR 2010, Cesme, Izmir, Turkey, August 18-20, 2010. Proceedings*, pages 324–333, 2010.
[8] M. Filippone. Dealing with non-metric dissimilarities in fuzzy central clustering algorithms. *International Journal of Approximate Reasoning*, 50(2):363–384, 2009. cited By (since 1996)8.
[9] L. Goldfarb. A unified approach to pattern recognition. *Pattern Recognition*, 17(5):575 – 582, 1984.
[10] S. Gu and Y. Guo. Learning svm classifiers with indefinite kernels. volume 2, pages 942–948, 2012. cited By (since 1996)0.
[11] P. Kar and P. Jain. Supervised learning with similarity functions. volume 1, pages 215–223, 2012. cited By (since 1996)1.
[12] Julian Laub. *Non-metric pairwise proximity data*. PhD thesis, 2004.
[13] Julian Laub, Volker Roth, Joachim M. Buhmann, and Klaus-Robert Müller. On the information and representation of non-euclidean pairwise data. *Pattern Recognition*, 39(10):1815–1826, 2006.
[14] Yadong Mu and Shuicheng Yan. Non-metric locality-sensitive hashing. In Maria Fox and David Poole, editors, *AAAI*. AAAI Press, 2010.
[15] A. MuÃśoz and I.M. De Diego. From indefinite to positive semi-definite matrices. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 4109 LNCS:764–772, 2006. cited By (since 1996)2.
[16] E. Pekalska and R. Duin. *The dissimilarity representation for pattern recognition*. World Scientific, 2005.
[17] E. Pekalska and R.P.W. Duin. Beyond traditional kernels: Classification in two dissimilarity-based representation spaces. *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews*, 38(6):729–744, 2008. cited By (since 1996)16.
[18] Elsbieta Pekalska and Bernard Haasdonk. Kernel discriminant analysis for positive definite and indefinite kernels. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(6):1017–1032, 2009.
[19] Elzbieta Pekalska, Robert P. W. Duin, Simon Günter, and Horst Bunke. On not making dissimilarities euclidean. In *Structural, Syntactic, and Statistical Pattern Recognition, Joint IAPR International Workshops, SSPR 2004 and SPR 2004, Lisbon, Portugal, August 18-20, 2004 Proceedings*, pages 1145–1154, 2004.
[20] F.-M. Schleif and A. Gisbrecht. Data analysis of (non-)metric proximities at linear costs. In *Proceedings of SIMBAD 2013*, page accepted, 2013.
[21] J. Shawe-Taylor and N. Cristianini. *Kernel Methods for Pattern Analysis and Discovery*. Cambridge University Press, 2004.
[22] L. Van Der Maaten and G. Hinton. Visualizing non-metric similarities in multiple maps. *Machine Learning*, 87(1):33–55, 2012. cited By (since 1996)3.
[23] Liwei Wang, Masashi Sugiyama, Cheng Yang, Kohei Hatano, and Jufu Feng. Theory and algorithm for learning with dissimilarity functions. *Neural Computation*, 21(5):1459–1484, 2009.
[24] Stefanos Zafeiriou. Subspace learning in krein spaces: Complete kernel fisher discriminant analysis with indefinite kernels. In Andrew W. Fitzgibbon, Svetlana Lazebnik, Pietro Perona, Yoichi Sato, and Cordelia Schmid, editors, *ECCV (4)*, volume 7575 of *Lecture Notes in Computer Science*, pages 488–501. Springer, 2012.

Acknowledgment A Marie Curie Intra-European Fellowship (IEF): FP7-PEOPLE-2012-IEF (FP7-327791-ProMoS) is grateful acknowledged. PT was supported by the EPSRC grant EP/L000296/1, "Personalized Health Care through Learning in the Model Space". Frank-M. Schleif is affiliated member of the University of Applied Sciences Mittweida.

ROC-Optimization and Statistical Quality Measures in Learning Vector Quantization Classifiers

M. Biehl[1], M. Kaden*[2], P. Stürmer[2], and T. Villmann[2]†

[1]Johann-Bernoulli-Institute for Mathematics and Computer Sciences,
University Groningen, The Netherlands

[2]Computational Intelligence Group,
University of Applied Sciences Mittweida, Germany

Abstract

This paper deals with the integration of statistical measures into the framework of prototype-based learning vector quantization for learning of binary classification problems. In particular, the evaluation and optimization of the confusion matrix by means of a learning vector quantizer is considered keeping the Hebbian learning paradigm for prototype adaptation. In a further step, receiver operating characteristic curves are investigated. The area under the respective curves, which can be equivalently interpreted by a rank-statistics model, serves as an alternative quality measure for parametrized classifiers. As we show, this statistical approach can also be integrated into the learning vector quantization scheme whereas the precision-recall-curve counterpart is not suitable for such a model approach.

1 Introduction - Classification by Learning Vector Quantization

Learning vector quantization (LVQ) models are prototype-based adaptive classifiers for processing vectorial data [15]. Training samples are assumed to be of the form $\mathbf{v} \in V \subseteq \mathbb{R}^n$ with class labels $x_{\mathbf{v}} = x(\mathbf{v}) \in \mathcal{C} = \{1, \ldots, C\}$. The set of prototypes $W = \{\mathbf{w}_j \in \mathbb{R}^n, j = 1 \ldots M\}$ contains representatives of the classes carrying prototype

*supported by the European Social Foundation Saxony (ESF)

†*corresponding author - email: thomas.villmann@hs-mittweida.de*

labels $y_j \in \mathcal{C}$. Classification decisions for unknown data samples $\tilde{\mathbf{v}}$ are usually made according to a winner take all rule, i.e.

$$x_{\tilde{\mathbf{v}}} := y_{s(\tilde{\mathbf{v}})} \text{ with } s\left(\tilde{\mathbf{v}}\right) = argmin_j \left(d\left(\tilde{\mathbf{v}}, \mathbf{w}_j\right)\right)$$

where $d\left(\tilde{\mathbf{v}}, \mathbf{w}_j\right)$ is a dissimilarity measure in the data space, frequently chosen as the Euclidean distance. LVQ training amounts to distributing the prototypes in the data space such that the classification error is minimized. Stochastic gradient descent learning have been introduced which is based on objective function

$$E\left(W, f\right) = \frac{1}{2} \sum_{\mathbf{v} \in V} f\left(\mu\left(\mathbf{v}\right)\right) \tag{1}$$

approximating the classification error [22]. Here, the function

$$\mu\left(\mathbf{v}\right) = \frac{d^+\left(\mathbf{v}\right) - d^-\left(\mathbf{v}\right)}{d^+\left(\mathbf{v}\right) + d^-\left(\mathbf{v}\right)} \tag{2}$$

is the so-called classifier function. This approach is known as Generalized LVQ (GLVQ) [22]. Here $d^+\left(\mathbf{v}\right) = d\left(\mathbf{v}, \mathbf{w}^+\right)$ denotes the dissimilarity between the data vector $\mathbf{v}$ and the closest prototype $\mathbf{w}^+ = \mathbf{w}_{s^+}$ with the same class label $y_{s^+} = x_{\mathbf{v}}$, while $d^-\left(\mathbf{v}\right) = d\left(\mathbf{v}, \mathbf{w}^-\right)$ is the distance from the best matching prototype $\mathbf{w}^-$ with a class label y_{s^-} different from $x_{\mathbf{v}}$. The *modulation function* f in (1) is a monotonically increasing function usually chosen as a sigmoid or the identity function. A typical choice is the Fermi function

$$f_\theta\left(x\right) = \frac{1}{1 + a \cdot \exp\left(-\frac{(x - x_0)}{2\theta^2}\right)} \tag{3}$$

with $x_0 = 0$ and $a = 1$ as standard parameter values. The parameter θ determines the slope of f_θ but is frequently fixed as $\theta = 1$.

Stochastic gradient learning performs update steps of the form

$$\triangle \mathbf{w}^\pm \propto -\frac{\partial f_\theta\left(\mu\left(\mathbf{v}\right)\right)}{\partial \mu\left(\mathbf{v}\right)} \cdot \frac{\partial \mu\left(\mathbf{v}\right)}{\partial d^\pm\left(\mathbf{v}\right)} \cdot \frac{\partial d^\pm\left(\mathbf{v}\right)}{\partial \mathbf{w}^\pm} \tag{4}$$

for a randomly chosen data sample $\mathbf{v}$.

2 Classification Accuracy and Statistical Measures in GLVQ

In the following we demonstrate how to realize a classifier optimizing a statistical measure based on the confusion matrix by means of GLVQ. GLVQ is the preferred choice to keep the intuitive approach of prototype based classification.

labels	true			
		C_+	C_-	
predicted	C_+	TP	FP	$\widehat{N}_+$
	C_-	FN	TN	$\widehat{N}_-$
		N_+	N_-	N

Table 1: Contingency / Confusion matrix: TP - true positives, FP - false positives, TN - true negatives, FN - false negatives, $N_\pm$- number of positive/negative data, $\widehat{N}_+$ - number of predicted positive/negative samples.

In this view, first we observe that the classifier function $\mu(\mathbf{v})$ from (2) becomes negative if the data point $\mathbf{v}$ is correctly classified, i.e. if $x_\mathbf{v} = y_{s(\mathbf{v})}$ is valid. Further, in the limit $\theta \searrow 0$ the sigmoid f_θ (3) becomes the Heaviside function

$$H(x) = \begin{cases} 0 & if \quad x \leq 0 \\ 1 & else \end{cases} , \tag{5}$$

such that *border sensitive classification learning* takes place [13]. Thus, in this limit, $E(W, H)$ counts the *misclassifications*. Considering a two-class problem with a positive class C_+ labeled by '$\oplus$' and a negative class C_- with class label '$\ominus$', these misclassifications are distinguished as the false positives (FP) and false negatives (FN) according to the contingency table Tab. 1.

Yet, counting of misclassifications is not always an appropriate evaluation of classifier, in particular, if the data are imbalanced [21]. In statistical analysis contingency table evaluations are well-known to deal with this problem more properly. Several measures were developed to judge the classification quality based on the confusion matrix emphasizing different aspects. For example, *precision* π and *recall* ρ, defined as

$$\pi = \frac{TP}{TP + FP} = \frac{TP}{\widehat{N}_+} \tag{6}$$

and

$$\rho = \frac{TP}{TP + FN} = \frac{TP}{N_+} \tag{7}$$

respectively, are used in the widely applied F_β-measure

$$F_\beta = \frac{(1 + \beta^2) \cdot \pi \cdot \rho}{\beta^2 \cdot \pi + \rho} \tag{8}$$

developed by C.J. van Rijsbergen [20].

To integrate these contingency quantities into a GLVQ-like cost function, we have to approximate them properly while ensuring their dependence on the prototypes is differentiable. For this purpose we introduce the quantity $\hat{\mu}(\mathbf{v}) = f_\theta(-\mu(\mathbf{v}))$ with

$\hat{\mu}(\mathbf{v}) \approx 1$ iff the data point $\mathbf{v}$ is correctly classified and $\hat{\mu}(\mathbf{v}) \approx 0$ otherwise for small values θ, with the derivative

$$\frac{\partial \hat{\mu}(\mathbf{v})}{\partial \mathbf{w}^{\pm}} = -\frac{\partial \hat{\mu}(\mathbf{v})}{\partial f_{\theta}} \cdot \frac{\partial f_{\theta}}{\partial \mu} \cdot \frac{\partial \mu}{\partial d^{\pm}(\mathbf{v})} \cdot \frac{\partial d^{\pm}(\mathbf{v})}{\partial \mathbf{w}^{\pm}} \,.$$

Thus we can express all quantities of the confusion matrix in terms of the new classifier function $\hat{\mu}(\mathbf{v})$:

$$TP = \sum_{\mathbf{v}} \delta_{\oplus, x_{\mathbf{v}}} \cdot \hat{\mu}(\mathbf{v}) \,,$$

$$FP = \sum_{\mathbf{v}} \delta_{\ominus, x_{\mathbf{v}}} \cdot (1 - \hat{\mu}(\mathbf{v})) \,,$$

$$FN = \sum_{\mathbf{v}} \delta_{\oplus, x_{\mathbf{v}}} \cdot (1 - \hat{\mu}(\mathbf{v}))$$

and

$$TN = \sum_{\mathbf{v}} \delta_{\ominus, x_{\mathbf{v}}} \cdot \hat{\mu}(\mathbf{v})$$

with $\delta_{\oplus, x_{\mathbf{v}}}$ is the Kronecker symbol and $\delta_{\ominus, x_{\mathbf{v}}} = 1 - \delta_{\oplus, x_{\mathbf{v}}}$. Obviously, all these quantities are also differentiable with respect to $\hat{\mu}(\mathbf{v})$ and, hence, also with respect to the prototypes $\mathbf{w}_k$. In consequence, an arbitrary general statistical measure can be optimized by a GLVQ-like stochastic gradient learning of the prototypes, if it is *continuous and differentiable* with respect to TP, FP, FN, and TN. Clearly, the above mentioned quantities precision π and recall ρ as well as the F_{β}-measure belong to this function class and, therefore, can be plugged into the GLVQ learning scheme.

3 Receiver Operation Characteristic Optimization and GLVQ

The Receiver Operation Characteristic (ROC) is an important tool for performance comparison of binary classifiers. A classifier is considered superior if it delivers a higher value of the *area under the ROC-curve* (AUC). Following [4], the AUC refers to the true distribution of positive and negative instances, but it can be estimated using a sample. The normalized Wilcoxon-Mann-Whitney statistic [25, 18] reveals the maximum likelihood of the true AUC for a given classifier [26]. Several method were developed to maximize AUC directly including gradient descent learning [11], approximated AUC optimization [5], reject option optimization [17], AUC optimization by linear programming [1] or ranking based optimization [8], to name just a few of the recently proposed approaches. Yet, for prototype based classification based on LVQ, which can be seen as a robust variant of the nearest-neighbor classifier [12], a direct optimization scheme for AUC is not known so far. As we will show in this chapter, the GLVQ variant of the basic LVQ scheme can be easily adapted for AUC optimization.

3.1 Probability Interpretation of AUC

Suppose the binary classification problem for classes A and B and related datasets V_A and V_B with cardinalities $\#V_A$, $\#V_B$, respectively. Further assume that a classifier delivers a continuous output (discriminant function) ϑ used for the classification decision. Then the AUC can be interpreted as the probability P_{AB} that a classifier will rank a randomly chosen A-instance $\mathbf{v}_A \in V_A$ higher than a randomly chosen B-instance $\mathbf{v}_B \in V_B$ [7]. In this view we can formulate an equivalent cost function introducing the (local) ordering function

$$O_\theta\left(\mathbf{v}_A, \mathbf{v}_B\right) = f_\theta\left(\vartheta\left(\mathbf{v}_A\right) - \vartheta\left(\mathbf{v}_B\right)\right) \tag{9}$$

for an ordered pair $\left(\mathbf{v}_A, \mathbf{v}_B\right)$ of vectors. We approximate P_{AB} by

$$P_{AB}\left(\theta\right) = \frac{1}{\#V_{AB}} \sum_{\left(\mathbf{v}_A, \mathbf{v}_B\right)} O_\theta\left(\mathbf{v}_A, \mathbf{v}_B\right) \tag{10}$$

depending on the slope parameter θ of the sigmoid function $f_\theta\left(x\right)$ from (3). If $\theta \searrow 0$ holds, $P_{AB}\left(\theta\right)$ converges to P_{AB} for $\#V_{AB} \to \infty$ according to the underlying rank statistics [18, 25] and paying attention to the functional limit $f_\theta \to H$ for $\theta \searrow 0$ with the Heaviside function H from (5).

3.2 A cost function for AUC based on GLVQ

The probabilistic interpretation of the AUC introduced in the previous subsection can be facilitated in the GLVQ-framework. To this end, the discriminant function ϑ in (9) is replaced by a discriminat function μ_{AB} specifically designed for the GLVQ and, hence based on the prototypes used in GLVQ. In particular, we define

$$\mu_{AB}\left(\mathbf{v}, \gamma\right) = \frac{d^B\left(\mathbf{v}\right) - d^A\left(\mathbf{v}\right)}{d^A\left(\mathbf{v}\right) + d^B\left(\mathbf{v}\right)} - \gamma \tag{11}$$

with $d^A\left(\mathbf{v}\right) = d^A\left(\mathbf{v}, \mathbf{w}_A^*\left(\mathbf{v}\right)\right)$ where $\mathbf{w}_A^*\left(\mathbf{v}\right)$ is the closest prototype to $\mathbf{v}$ responsible for class A. Analogously, $\mathbf{w}_B^*$ and $d^B\left(\mathbf{v}\right)$ are defined in the same manner. The parameter $\gamma \in [-1, 1]$ defines a threshold shifting the decision boundary between $\mathbf{w}_A^*$ and $\mathbf{w}_B^*$, which plays the role of the varying parameter for the ROC-curve. The unbiased case is obtained for the choice $\gamma = 0$. With these settings the ROC cost function for a respective GLVQ-scheme reads as

$$E_{ROC}\left(\theta, V_A, V_B, W\right) = \frac{1}{\#V_{AB}} \sum_{\left(\mathbf{v}_A, \mathbf{v}_B\right)} f_\theta\left(\mu_{AB}\left(\mathbf{v}_A, \gamma\right) - \mu_{AB}\left(\mathbf{v}_B, \gamma\right)\right) \tag{12}$$

again depending on the slope parameter θ of the sigmoid function $f_\theta\left(x\right)$ from (3). Hence, border sensitive learning in this ROC-GLVQ, i.e. forcing $\theta \searrow 0$ in (9), leads to the limit

$$E_{ROC}\left(\theta, V_A, V_B, W\right) \overset{\theta \searrow 0}{\longrightarrow} P_{AB}\ . \tag{13}$$

Further, using the derivatives

$$\frac{\partial \mu_{AB}(\mathbf{v},\gamma)}{\partial \mathbf{w}_A^*(\mathbf{v})} = \frac{d^B(\mathbf{v})}{\left(d^A(\mathbf{v}) + d^B(\mathbf{v})\right)^2} \cdot \frac{\partial d^A(\mathbf{v})}{\partial \mathbf{w}_A^*(\mathbf{v})}$$

and

$$\frac{\partial \mu_{AB}(\mathbf{v},\gamma)}{\partial \mathbf{w}_B^*(\mathbf{v})} = -\frac{d^A(\mathbf{v})}{\left(d^A(\mathbf{v}) + d^B(\mathbf{v})\right)^2} \cdot \frac{\partial d^B(\mathbf{v})}{\partial \mathbf{w}_B^*(\mathbf{v})}$$

we can calculate the gradients of the GLVQ-adapted ordering function

$$O_\theta^{\mu_{AB}}(\mathbf{v}_A,\mathbf{v}_B) = f_\theta\left(\mu_{AB}(\mathbf{v}_A,\gamma) - \mu_{AB}(\mathbf{v}_B),\gamma\right) \tag{14}$$

regarding to both $\mathbf{v}_A$ and $\mathbf{v}_B$, respectively:

$$\frac{\partial O_\theta^{\mu_{AB}}(\mathbf{v}_A,\mathbf{v}_B)}{\partial \mathbf{w}_A^*(\mathbf{v}_A)} = \left.\frac{\partial f_\theta}{\partial z}\right|_z \cdot \left(\frac{\partial \mu_{AB}(\mathbf{v}_A,\gamma)}{\partial \mathbf{w}_A^*(\mathbf{v}_A)} - \frac{\partial \mu_{AB}(\mathbf{v}_B,,\gamma)}{\partial \mathbf{w}_A^*(\mathbf{v}_A)}\right) \tag{15}$$

$$\frac{\partial O_\theta^{\mu_{AB}}(\mathbf{v}_A,\mathbf{v}_B)}{\partial \mathbf{w}_A^*(\mathbf{v}_B)} = \left.\frac{\partial f_\theta}{\partial z}\right|_z \cdot \left(\frac{\partial \mu_{AB}(\mathbf{v}_A,\gamma)}{\partial \mathbf{w}_A^*(\mathbf{v}_B)} - \frac{\partial \mu_{AB}(\mathbf{v}_B,\gamma)}{\partial \mathbf{w}_A^*(\mathbf{v}_B)}\right) \tag{16}$$

$$\frac{\partial O_\theta^{\mu_{AB}}(\mathbf{v}_A,\mathbf{v}_B)}{\partial \mathbf{w}_B^*(\mathbf{v}_A)} = \left.\frac{\partial f_\theta}{\partial z}\right|_z \cdot \left(\frac{\partial \mu_{AB}(\mathbf{v}_A,\gamma)}{\partial \mathbf{w}_B^*(\mathbf{v}_A)} - \frac{\partial \mu_{AB}(\mathbf{v}_B,\gamma)}{\partial \mathbf{w}_B^*(\mathbf{v}_A)}\right) \tag{17}$$

$$\frac{\partial O_\theta^{\mu_{AB}}(\mathbf{v}_A,\mathbf{v}_B)}{\partial \mathbf{w}_B^*(\mathbf{v}_B)} = \left.\frac{\partial f_\theta}{\partial z}\right|_z \cdot \left(\frac{\partial \mu_{AB}(\mathbf{v}_A,\gamma)}{\partial \mathbf{w}_B^*(\mathbf{v}_B)} - \frac{\partial \mu_{AB}(\mathbf{v}_B,\gamma)}{\partial \mathbf{w}_B^*(\mathbf{v}_B)}\right) \tag{18}$$

with $z = \mu_{AB}(\mathbf{v}_A,\gamma) - \mu_{AB}(\mathbf{v}_B,\gamma)$.

In consequence, GLVQ-like stochastic gradient learning is possible also for the ROC cost function E_{ROC} from (12). However, for this purpose a *structured input*

$$\mathbf{v}_{AB} = (\mathbf{v}_A,\mathbf{v}_B)$$

is required in GLVQ learning. Thus, stochastic gradient descent learning on $E_{ROC-GLVQ}$ takes place with respect to $\mathbf{w}_A^*(\mathbf{v}_A)$, $\mathbf{w}_A^*(\mathbf{v}_B)$, $\mathbf{w}_B^*(\mathbf{v}_A)$, and $\mathbf{w}_B^*(\mathbf{v}_B)$ using the gradients (15)–(18) of the GLVQ-adapted ordering function $O_\theta^{\mu_{AB}}$ from (14) depending on the randomly selected structured input $\mathbf{v}_{AB}$. We emphasize at this point that the ROC-GLVQ delivers an AUC-optimizing scheme only in the limit $\theta \searrow 0$ of border sensitive learning.

3.3 The ROC-LVQ model

In the previous section we introduced a cost function for AUC based on the GLVQ-paradigm. Although frequently assumed, the standard GLVQ does not guarantees the prototypes to being class representative after learning [19, 9]. To enhance this property a generative cost function amount

$$E_{GEN}\left(\theta, V_A, V_B, W\right) = \sum_{\left(\mathbf{v}_A, \mathbf{v}_B\right)} d^A\left(\mathbf{v}_A\right) + d^B\left(\mathbf{v}_B\right) \tag{19}$$

has to be added to E_{ROC} from (12) such that the overall ROC-GLVQ cost function becomes

$$E_{ROC-GLVQ}\left(\theta, V_A, V_B, W, \alpha\right) = (1-\alpha) \cdot E_{GEN} + \gamma \cdot E_{ROC} \tag{20}$$

with the balancing parameter $\alpha \in [0,1]$ weighting both aspects classification-separation versus description of class distribution [12].

4 Precision-Recall-Curves and GLVQ

Precision-Recall-Curves (PR-curves) are closely related to ROC but do not provide the identical information [2, 6, 16]. Therefore, they can provide additional insides. Precision and recall as introduced in (6) and (7), respectively, are intensively used for test statistics and classification problems in medical applications. In this area, the recall ρ is often denoted as *sensitivity* describing the ability of the classifier to detect positive samples accurately. The counterpart of the sensitivity is the *specificity* value

$$\varsigma = \frac{TN}{TN+FP} = \frac{TN}{N_-} \tag{21}$$

judging the ability for detecting negative samples. Precision-recall-relations are frequently investigated taking the F_β-measure from (8) with $\beta = 1$ [16]. This is just the ratio of the arithmetic and the geometric mean of precision and recall, i.e.

$$F_1 = \frac{2 \cdot \pi \cdot \rho}{\pi + \rho} \tag{22}$$

However, for evaluation of PR-curves, only a few approaches were proposed [3, 2, 16]. In the following we will identify the difficulties arising, if one would like to adapt the ideas of AUC-maximization learning in GLVQ to PR-curve optimization by GLVQ.

4.1 Basic Definitions and Notations

We follow the explanations in [3]: We denote the class A as *positive* class and B is the *negative*. The real-valued model output for the positive samples $\mathbf{v}_A$ is y^A and, analogously, negative samples $\mathbf{v}_B$ generate y^B. The class skew S is defined as the probability $S = P(A)$ and also known as prevalence or a prior class distribution. The *recall* can be written as a probability

$$\rho(c) = P\left(y^A > c\right) \tag{23}$$

whereas the precision π is a conditional probability

$$\pi(c) = P\left(\mathbf{v} \in A | z > c\right) \ . \tag{24}$$

In this formula, $z = z(\mathbf{v})$ is the overall real-valued output of the classifier model for a given unclassified sample $\mathbf{v}$ being a mixture of y^A and y^B. With this definitions, the precision-recall-curve PR is the set

$$PR = \{(\rho(c), \pi(c)), -\infty < c < \infty\} \ .$$

We assume larger outputs to be associated with positive samples. In consequence, as c decreases, the recall $\rho(c)$ increases to one and the precision $\pi(c)$ approaches to S. The *area* Θ *under the precision-recall-curve* (AUCPR) is an average of the precision weighted by the probability of a given theshold c:

$$\Theta = \int_{-\infty}^{\infty} \pi(c)\, dP\left(y^A \leq c\right) \tag{25}$$

Since, $\pi(c)$ and $P\left(y^A \leq c\right)$ are both bounded on the unit square, the inequality $0 \leq \Theta \leq 1$ holds. Therefore, Θ can be interpreted as a probability. According to [3], the integral (25) can be interpreted as the fraction of positive examples among those examples whose output values exceed a randomly selected threshold c. Eq.(25) can be written equivalently as

$$\Theta = \int_0^1 \pi(\rho(c))\, d\rho(c) \tag{26}$$

paying attention to the fact that for $-\infty \leq c \leq \infty$ the range $\rho(c) \in [0,1]$ is valid [16, 14].

There exist several estimators for Θ in case of real datasets $V = V_A \cup V_B$ [3]. One powerful estimator avoiding the explicit determination of the empirical curve PR is the the averaged precision

$$\hat{\Theta} = \frac{1}{\#V_A} \sum_{i=1}^{\#V_A} \hat{\pi}\left(y_i^A\right) \tag{27}$$

and

$$\hat{\pi}(x) = \frac{S \cdot \hat{\rho}(x)}{S \cdot \hat{\rho}(x) + \frac{(1-S)}{\#V_B} \sum_{j=1}^{\#V_B} I\left(y_j^B > x\right)} \tag{28}$$

is the empirical precision estimate with

$$\hat{\rho}(x) = \frac{1}{\#V_A} \sum_{i=1}^{\#V_A} I\left(y_i^A > x\right) \tag{29}$$

being the empirical estimate of the recall $\rho(x)$ and $I(E)$ is the indicator function of the event E.

4.2 PRC-LVQ

According to the previous ROC-GLVQ model assumptions we have to define a GLVQ-output in the sense of a discriminat function. For this purpose we make use of the already declared and define

$$\hat{\mu}_{AB}(\mathbf{v}) = -\mu_{AB}(\mathbf{v}) \tag{30}$$

as PR-LVQ discrimant function, which is in agreement with the assumption that larger outputs should be associated with positive samples. Further, we replace the indicator function $I\left(y > x\right)$ by the Heaviside-function $H\left(y - x\right)$ from (5) and approximate the latter one by the sigmoid function f_θ from (3). Doing so and estimating the skew S as $S = \frac{\#V_A}{\#V_B}$, the estimator $\hat{\Theta}$ from (27) for the AUCPR can be written as a cost function

$$E_{PRC}\left(\theta, V_A, V_B, W\right) = \frac{1}{\#V_A} \sum_{i=1}^{\#V_A} \frac{1}{1 + \frac{(1-S)\cdot\sum_{j=1}^{\#V_B} f_\theta\left(\hat{\mu}_{AB}\left(\mathbf{v}_j^B\right) - \hat{\mu}_{AB}\left(\mathbf{v}_i^A\right)\right)}{\sum_{k=1}^{\#V_A} f_\theta\left(\hat{\mu}_{AB}\left(\mathbf{v}_k^A\right) - \hat{\mu}_{AB}\left(\mathbf{v}_i^A\right)\right)}} \tag{31}$$

to be minimized in dependence on the prototype set W. In analogy to the above ROC-LVQ cost function we finally obtain the formal PRC-LVQ cost function as

$$E_{PRC-LVQ}\left(\theta, V_A, V_B, W, \gamma\right) = (1 - \gamma) \cdot E_{GEN} + \gamma \cdot E_{PRC} \tag{32}$$

with E_{GEN} being the generative part (19).

However, this cost function E_{PRC} contains nested sums over the single events $\mathbf{v}_k^A$ and $\mathbf{v}_j^B$ in contrast to the pairwise events $(\mathbf{v}_A, \mathbf{v}_B)$ considered in the cost function (12) of AUCs for GLVQ. Thus stochastic gradient descent learning would become complicate for this model, because of the nested sums. Therefore, other optimization strategies like Expectation-Maximization should be investigated instead. However, this is behind the scope of this introdution article and will be studied in the future.

5 Conclusion

We present in this article the mathematical framework for learning of prototype-based LVQ-classifiers to optimize statistical quality measures based on the confusion matrix or receiver operating characteristic. We further obtained a GLVQ modification for maximizing explicitly the area under the ROC-curve, whereas a derivation of a similar method for precision-recall-curve optimization failed.

Obviously, the obtained approaches can be easily combined with other advanced GLVQ-techniques like relevance and matrix learning or kernelized variants [10, 23, 24].

References

[1] K. Ataman, W. Street, and Y. Zhang. Learning to rank by maximizing AUC with linear programming. In *Proceedings of the IEEE International Joint Conference on Neural Networks (IJCNN)*, pages 123–129. IEEE Press, 2006.

[2] K. Boyd, V. Costa, J. Davis, and C. Page. Unachievable region in precision-recall space and its effect on empirical evaluation. In *Proceedings of the 12th ICML Edinburgh*, 2012.

[3] K. Boyd, K. Eng, and C. Page. Area under the precision-recall curve: Point estimates and confidence intervals. In H. Blockeel, K. Kersting, S. Nijssen, and F. Zelezný, editors, *Machine Learning and Knowledge Discovery in Databases*, volume 8190 of *LNCS*, pages 451–466, Berlin Heidelberg, 2013. Springer-Verlag.

[4] U. Brefeld and T. Scheffer. AUC maximizing support vector learning. In *Proceedings of ICML 2005 workshop on ROC Analysis in Machine Learning*, pages 377–384, 2005.

[5] T. Calders and S. Jaroszewicz. Efficient AUC optimization for classification. In J. Kok, J. Koronacki, R. L. de Mantaras, S. Matwin, D. Mladenic, and A. Skowron, editors, *Knowledge Discovery in Databases: PKDD 2007*, volume 4702 of *LNCS*, pages 42–53. Springer-Verlag, 2007.

[6] J. Davis and M. Goadrich. The relationship between precision-recall and ROC curves. In *Proceedings of the 23rd International Conference on Machine Learning*, ICML '06, pages 233–240, New York, NY, USA, 2006. ACM.

[7] T. Fawcett. An introduction to ROC analysis. *Pattern Recognition Letters*, 27:861–874, 2006.

[8] H. Güvenir and M. Kurtcephe. Ranking instances by maximizing the area under ROC curve. *IEEE Transactions on Knowledge and Data Engineering*, 25(10):2356–2366, 2013.

[9] B. Hammer, D. N. M. Riedel, and T. Villmann. Generative versus discriminative prototype based classification. In T. Villmann, F.-M. Schleif, M. Kaden, and M. Lange, editors, *Advances in Self-Organizing Maps and Learning Vector Quantization: Proceedings of 10th International Workshop WSOM 2014, Mittweida*, volume 295 of *Advances in Intelligent Systems and Computing*, pages 123–132, Berlin, 2014. Springer.

[10] B. Hammer and T. Villmann. Generalized relevance learning vector quantization. *Neural Networks*, 15(8-9):1059–1068, 2002.

[11] A. Herschtal and B. Raskutti. Optimising area under the ROC curve using gradient descent. In *Proceedings of the 21st International Conference on Machine Learning (Banff, Canada)*, pages 49–56, 2004.

[12] M. Kaden, M. Lange, D. Nebel, M. Riedel, T. Geweniger, and T. Villmann. Aspects in classification learning - Review of recent developments in Learning Vector Quantization. *Foundations of Computing and Decision Sciences*, 39(2):79–105, 2014.

[13] M. Kästner, M. Riedel, M. Strickert, W. Hermann, and T. Villmann. Border-sensitive learning in kernelized learning vector quantization. In I. Rojas, G. Joya, and J. Cabestany, editors, *Proc. of the 12th International Workshop on Artificial Neural Networks (IWANN)*, volume 7902 of *LNCS*, pages 357–366, Berlin, 2013. Springer.

[14] J. Keilwagen, I. Grosse, and J. Grau. Area under precision-recall curves for weighted and unweighted data. *PLOS|ONE*, 9(3 / e92209):1–13, 2014.

[15] T. Kohonen. *Self-Organizing Maps*, volume 30 of *Springer Series in Information Sciences*. Springer, Berlin, Heidelberg, 1995. (Second Extended Edition 1997).

[16] T. Landgrebe, P. Paclìk, R. Duin, and A. Bradley. Precision-recall operating characteristic (P-ROC) curves in imprecise environments. In *Proceedings of ICPR*, 2006.

[17] T. Landgrebe, D. Tax, P. Paclìk, and R. Duin. The interaction between classification and reject performance for distance-based reject-option classifiers. *Pattern Recognition Letters*, 27:908–917, 2005.

[18] H. Mann and D. Whitney. On a test whether one of two random variables is stochastically larger than the other. *Annals of Mathematical Statististics*, 18:50–60, 1947.

[19] D. Nebel, B. Hammer, and T. Villmann. Supervised generative models for learning dissimilarity data. In M. Verleysen, editor, *Proc. of European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (ESANN'2014)*, pages 35–40, Louvain-La-Neuve, Belgium, 2014. i6doc.com.

[20] C. Rijsbergen. *Information Retrieval.* Butterworths, London, 2nd edition edition, 1979.

[21] L. Sachs. *Angewandte Statistik.* Springer Verlag, 7-th edition, 1992.

[22] A. Sato and K. Yamada. Generalized learning vector quantization. In D. S. Touretzky, M. C. Mozer, and M. E. Hasselmo, editors, *Advances in Neural Information Processing Systems 8. Proceedings of the 1995 Conference*, pages 423–9. MIT Press, Cambridge, MA, USA, 1996.

[23] P. Schneider, B. Hammer, and M. Biehl. Adaptive relevance matrices in learning vector quantization. *Neural Computation*, 21:3532–3561, 2009.

[24] T. Villmann, S. Haase, and M. Kaden. Kernelized vector quantization in gradient-descent learning. *Neurocomputing*, page in press, 2014.

[25] F. Wilcoxon. Andividual comparisons by ranking methods. *Biometrics*, 1:80–83, 1945.

[26] L. Yan, R. Dodier, M. Mozer, and R. Wolniewicz. Optimizing classifier performance via approximation to the Wilcoxon-Mann-Witney statistic. In *Proceedings of the 20th International Conference on Machine Learning*, pages 848–855, Menlo Park, CA, 2003. AAAI Press.

About the Generalization of the Eigen-Problem for Semi-Inner Products in Minkowski-ℓ_p-Spaces

S. Saralajew, M. Lange, and T. Villmann
University of Applied SciencesMittweida
Computational Intelligence Group

Abstract

Semi-inner products as generalization of inner products are recently discussed in several machine learning approaches for classification and vector quantization. This technical paper considers the eigen-problem from the perspective of semi-inner products and discusses related numerical and algebraic problems for its solution.

1 Introduction - the Usual Eigen-Problem

We start introducing the usual eigen-problem (EP) based on the usual inner product. This is done to clarify notations and to relate the later eigen-problem in case of semi-inner products to the usual ones.

For this purpose let V be a vector space over the field $\mathbb{K}$ assumed to be $\mathbb{R}$ or $\mathbb{C}$. The linear map $f : V \longrightarrow V$ is supposed to be an endomorphism. The EP consists in determination of a pair $(\lambda, \mathbf{v})$ such that $f(\mathbf{v}) = \lambda\mathbf{v}$ is valid with $\lambda \in \mathbb{K}$, $\mathbf{v} \in V$ and $\mathbf{v} \neq \mathbf{0}$. Then $\mathbf{v}$ is denoted as eigenvector of f and λ the respective eigenvalue. If we further assume that $\dim(V) = n < \infty$ is finite then f uniquely corresponds to a $n \times n$-matrix $\mathbf{A}$ over the field $\mathbb{K}$, i.e. $\mathbf{A} \in \mathbb{K}^{n\times n}$ and the EP becomes

$$\mathbf{A}\mathbf{v} = \lambda\mathbf{v} \tag{1.1}$$

also denoted more precisely as the right-side EP (REP) with $\mathbf{v}$ being a column vector. Analogously, we can define the left-side EP (LEP) as

$$\mathbf{v}^T\mathbf{A} = \lambda\mathbf{v}^T$$

where $\mathbf{v}^T$ denotes the transposed vector.

In this paper we will use the following notations for a given matrix $\mathbf{B} \in \mathbb{K}^{m\times n}$

- $B_{i\rightarrow}$: row $i \in \{1, 2, ..., m\}$ of the matrix $\mathbf{B}$
- $B_{j\downarrow}$; column $j \in \{1, 2, ..., n\}$ of the matrix $\mathbf{B}$
- $B_{i,j} = (\mathbf{B})_{i,j}$ matrix element at the position i, j of the matrix $\mathbf{B}$
- $\mathbf{I}$ is the n-dimensional unity matrix

The vector space as $V := \mathbb{C}^n$ together with the Euclidean inner product $\langle\bullet,\bullet\rangle_E$ is a Hilbert space. Thus the REP (1.1) writes as

$$\mathbf{A}\mathbf{v} = \lambda\mathbf{v} \Longleftrightarrow \begin{pmatrix} \langle A^T_{1\rightarrow}, \mathbf{v}\rangle_E \\ \langle A^T_{2\rightarrow}, \mathbf{v}\rangle_E \\ \vdots \\ \langle A^T_{n\rightarrow}, \mathbf{v}\rangle_E \end{pmatrix} = \lambda\mathbf{v} \tag{1.2}$$

with the induced norm $\|\mathbf{x}\|_2 = \sqrt{\langle\mathbf{x},\mathbf{x}\rangle_E}$.

Several numerical methods were developed to solve the REP, like

- v.-Mises-iteration with deflation
- inverse v.-Mises-iteration
- the QR-algorithm
- Krylow-subspace-method

to name just a few. They make intensively use of the Hermitian symmetry of the inner product $\langle\bullet,\bullet\rangle_E$, which implies the sesqui-linearity, i.e.

$$\langle\mathbf{v}, \lambda\cdot\mathbf{w}\rangle_E = \overline{\lambda}\cdot\langle\mathbf{v},\mathbf{w}\rangle_E$$

and linearity in the first argument. We refer to [3, 8], for further reading.

2 The Eigen Problem in the Minkowski-p-Space

Now we turn to consider normed complete vector spaces, i.e. Banach spaces. A prominent example is the n-dimensional Minkowski-p-space ℓ_p^n over the complex numbers $\mathbb{C}$ equipped with the Minkowski-p-norm

$$\|\mathbf{x}\|_p = \sqrt[p]{\sum_{k=1}^{n} |x_k|^p}$$

for $1 \leq p \leq \infty$.

For Banach spaces does not necessarily exist an inner product. However, a weaker concept can be identified - semi-inner products (SIPs) as introduced by G. Lumer [5]:

Definition 1 *A* semi-inner product $[\bullet,\bullet]$ *of a vector spaces* V *over the field* $\mathbb{K}$ *is a map*

$$[\bullet,\bullet] : V \times V \longrightarrow \mathbb{K}$$

with the following properties

1. $[\bullet,\bullet]$ *is semi-definite*

$$\forall \mathbf{x} \in V : [\mathbf{x},\mathbf{x}] \geq 0 \textit{ und } [\mathbf{x},\mathbf{x}] = 0 \Longleftrightarrow \mathbf{x} = \mathbf{0}$$

2. $[\bullet,\bullet]$ *is linear with respect to the first argument, i.e.*

$$\forall \mathbf{x},\mathbf{y},\mathbf{z} \in V, \forall \xi \in \mathbb{K} : \xi \cdot [\mathbf{x},\mathbf{z}] + [\mathbf{y},\mathbf{z}] = [\xi \cdot \mathbf{x} + \mathbf{y},\mathbf{z}]$$

3. $[\bullet,\bullet]$ *fulfills the Cauchy-Schwarz inequality*

$$\forall \mathbf{x},\mathbf{y} \in V : |[\mathbf{x},\mathbf{y}]|^2 \leq [\mathbf{x},\mathbf{x}]\,[\mathbf{y},\mathbf{y}]$$

Note that the Hermitian symmetry, as it is valid for inner products according to (1), is not required for SIPs and the triangle inequality for inner products is replaced by the Cauchy-Schwarz inequality. LUMER has shown that each Banach space $\mathbb{B}$ with norm $\|\bullet\|_{\mathbb{B}}$ can be equipped with a SIP $[\bullet,\bullet]_{\mathbb{B}}$ such that the norm is generated, i.e.

$$\|\mathbf{x}\|_{\mathbb{B}} = \sqrt{[\mathbf{x},\mathbf{x}]_{\mathbb{B}}}\,.$$

Generally, several SIPs may deliver the same norm. Uniqueness can be obtained by additional requirements, like differentiability in the second argument and other. We refer to [2] for details.

The previously mentioned Banach space ℓ_p^n obeys the unique SIP

$$[\mathbf{x},\mathbf{y}]_p = \frac{1}{(\|\mathbf{y}\|_p)^{p-2}} \sum_{\substack{k=1\\ y_k \neq 0}}^{n} x_k \cdot \bar{y}_k \cdot |y_k|^{p-2} \tag{2.1}$$

as shown in [2]. In the next step we introduce for this space an analog procedure to the Euclidean matrix multiplication from (1.2). For matrices $\mathbf{A} \in \mathbb{K}^{m\times l}$ and $\mathbf{B} \in \mathbb{K}^{l\times n}$ we define the operation $\diamond$ with respect to the SIP (2.1) for the ℓ_p^n-space as

$$\mathbf{A} \diamond \mathbf{B} := \begin{pmatrix} \left[A_{1\to}^T, B_{1\downarrow}\right]_p & \left[A_{1\to}^T, B_{2\downarrow}\right]_p & \cdots & \left[A_{1\to}^T, B_{n\downarrow}\right]_p \\ \left[A_{2\to}^T, B_{1\downarrow}\right]_p & \left[A_{2\to}^T, B_{2\downarrow}\right]_p & \cdots & \left[A_{2\to}^T, B_{n\downarrow}\right]_p \\ \vdots & \vdots & & \vdots \\ \left[A_{m\to}^T, B_{1\downarrow}\right]_p & \left[A_{m\to}^T, B_{2\downarrow}\right]_p & \cdots & \left[A_{m\to}^T, B_{n\downarrow}\right]_p \end{pmatrix} \tag{2.2}$$

denoted as SIP-matrix-multiplication (SIP-MM). The following lemma can be stated:

Lemma 2 *Let* $\mathbf{A} \in \mathbb{K}^{m\times k}$, $\mathbf{B} \in \mathbb{K}^{k\times l}$ *and* $\mathbf{C} \in \mathbb{K}^{l\times n}$ *be matrices. Then*

$$\mathbf{A} \cdot (\mathbf{B}\diamond\mathbf{C}) = (\mathbf{A} \cdot \mathbf{B})\diamond\mathbf{C}$$

is valid, whereby $\mathbf{A}\cdot\mathbf{B}$ *denotes the Euclidean matrix multiplication (EMM) with respect to the Euclidean inner product in agreement with (1.2).*

Proof. We consider an arbitrary matrix element $(\mathbf{A} \cdot (\mathbf{B}\diamond\mathbf{C}))_{i,j}$ and show that

$$(\mathbf{A} \cdot (\mathbf{B}\diamond\mathbf{C}))_{i,j} = ((\mathbf{A} \cdot \mathbf{B})\diamond\mathbf{C})_{i,j} \quad \forall i = 1, ..., m \text{ and } \forall j = 1, ..., n$$

is valid: Using the linearity of the SIP with respect to the first argument we derive

$$\begin{aligned}
(\mathbf{A} \cdot (\mathbf{B}\diamond\mathbf{C}))_{i,j} &= A_{i\rightarrow} \cdot \begin{pmatrix} \left[B^T_{1\rightarrow}, C_{j\downarrow}\right]_p \\ \left[B^T_{2\rightarrow}, C_{j\downarrow}\right]_p \\ \vdots \\ \left[B^T_{k\rightarrow}, C_{j\downarrow}\right]_p \end{pmatrix} \\
&= \sum_{h=1}^{k} A_{i,h} \cdot \left[B^T_{h\rightarrow}, C_{j\downarrow}\right]_p \\
&= \sum_{h=1}^{k} \left[A_{i,h} \cdot B^T_{h\rightarrow}, C_{j\downarrow}\right]_p \\
&= \Big[\sum_{h=1}^{k} A_{i,h} \cdot B^T_{h\rightarrow}, C_{j\downarrow}\Big]_p \\
&= \Big[\left(\sum_{h=1}^{k} A_{i,h} \cdot B_{h\rightarrow}\right)^T, C_{j\downarrow}\Big]_p \\
&= \left[(A_{i\rightarrow} \cdot \mathbf{B})^T, C_{j\downarrow}\right]_p \\
&= ((\mathbf{A} \cdot \mathbf{B})\diamond\mathbf{C})_{i,j}
\end{aligned}$$

showing the stated property. ■

Now we are able to define the REP and the LEP in Banach spaces with respect to a given SIP:

Definition 3 *Let* $\mathbb{B}$ *be a* n*-dimensional Banach space over the field* $\mathbb{K}$ *with the norm* $\|\bullet\|_{\mathbb{B}}$*. Let* $[\bullet, \bullet]_{\mathbb{B}}$ *be a SIP with* $\|\mathbf{x}\|_{\mathbb{B}} = \sqrt{[\mathbf{x}, \mathbf{x}]_{\mathbb{B}}}$*. The REP for a matrix* $\mathbf{A} \in \mathbb{K}^{n\times n}$ *with respect to the SIP* $[\bullet, \bullet]_{\mathbb{B}}$ *is defined as the determination of the pair* $(\lambda, \mathbf{v})$ *with* $\lambda \in \mathbb{K}$ *and* $\mathbf{v} \in \mathbb{B}$ *such that*

$$\mathbf{A} \diamond \mathbf{v} = \lambda\mathbf{v} \Longleftrightarrow \begin{pmatrix} \left[A^T_{1\rightarrow}, \mathbf{v}\right]_{\mathbb{B}} \\ \left[A^T_{2\rightarrow}, \mathbf{v}\right]_{\mathbb{B}} \\ \vdots \\ \left[A^T_{n\rightarrow}, \mathbf{v}\right]_{\mathbb{B}} \end{pmatrix} = \lambda\mathbf{v} \tag{2.3}$$

is valid. The related LEP reads as

$$\mathbf{v}^T \diamond \mathbf{A} = \lambda \mathbf{v}^T \Longleftrightarrow \left(\ [\mathbf{v}, A_{1\downarrow}]_{\mathbb{B}}\,, \quad [\mathbf{v}, A_{2\downarrow}]_{\mathbb{B}}\,, \quad \cdots\,, \quad [\mathbf{v}, A_{n\downarrow}]_{\mathbb{B}} \ \right) = \lambda \mathbf{v}^T \qquad (2.4)$$

and we refer to these as sREP and sLEP, respectively. $(\lambda, \mathbf{v})$ *is denoted as an eigen-pair.*

3 Numerical Approaches for the sREP and the sLEP

The sREP introduced in (2.3) offers a serious difficulty for its numerical solution. As it was explained above, most of the known numerical methods for the REP in Hilbert spaces utilize the Hermitian linearity (1) of the inner product in the second argument, which is not valid in case of SIPs. To our best knowledge, there is no general way to solve this problem so far by means of classical numerical approaches. An alternative was proposed in [1, 4] based on Hebbian learning for the special case of covariance matrices for a givenset $S \subset \mathbb{R}^n$ of data vectors $\mathbf{s} \in S$: For randomly presented $\mathbf{s}$ and randomly initialized vector $\mathbf{w} \in \mathbb{R}^n$ the adaptation

$$\mathbf{w} = \mathbf{w} + \triangle \mathbf{w}$$

with

$$\triangle \mathbf{w} = \varepsilon \cdot [\mathbf{s}, \mathbf{w}]_p \left(\mathbf{s} - [\mathbf{s}, \mathbf{w}]_p \cdot \mathbf{w} \right) \qquad (3.1)$$

is applied. The positive learning rate ε has to be small and decreasing during time, i.e. $1 \gg \varepsilon > 0$ and $\lim_{t \to \infty} \varepsilon(t) = 0$ with $\sum_t (\varepsilon(t))^2 = \infty$. The update scheme (3.1) is known as *Oja's rule* in the literature originally introduced for the REP [6, 7]. It delivers the eigenvector belonging to the maximum eigenvalue and can be extended also to calculate more than one eigenvector [9, 1, 4].

For the sLEP the situation is easier, because we can use the linearity of the SIP with respect to the first argument. As the main result of this paper we state the following theorem:

Theorem 4 (Saralajew&Villmann) *Let* $\mathbf{Q} \in \mathbb{K}^{n \times n}$ *be a regular matrix over the field* $\mathbb{K}$ *and* $(\lambda, \mathbf{v})$ *be an eigen-pair of the matrix* $\mathbf{A} \in \mathbb{K}^{n \times n}$ *with respect to the sLEP defined in (2.4). Then* $(\lambda, \mathbf{w})$ *is an eigen-pair of LEP for the matrix* $\mathbf{B} = (\mathbf{Q}^{-1} \diamond \mathbf{A})\,\mathbf{Q}$ *with respect to the* ***Euclidean*** *inner product and* $\mathbf{w} = \mathbf{Q}^T\mathbf{v}$ *holds.*

Proof. Using Lemma 2 we calculate

$$\begin{aligned}
\mathbf{w}^T\mathbf{B} &= \left(\mathbf{Q}^T\mathbf{v}\right)^T \mathbf{B} \\
&= \mathbf{v}^T\mathbf{Q}\left(\mathbf{Q}^{-1} \diamond \mathbf{A}\right)\mathbf{Q} \\
&= \mathbf{v}^T\left(\mathbf{I} \diamond \mathbf{A}\right)\mathbf{Q} \\
&= \left(\mathbf{v}^T \diamond \mathbf{A}\right)\mathbf{Q} \\
&= \lambda \mathbf{v}^T\mathbf{Q} \\
&= \lambda \mathbf{w}^T\,,
\end{aligned}$$

which is the desired result. ■

This theorem allows to transfer the sLEP to the usual Euclidean REP:

Corollary 5 (Saralajew&Villmann) *We consider the matrix* $\mathbf{C} = (\mathbf{I} \diamond \mathbf{A})^T$ *for* $\mathbf{A} \in \mathbb{K}^{n \times n}$ *being a matrix over the field* $\mathbb{K}$. *Then the sLEP* $\mathbf{v}^T \diamond \mathbf{A} = \lambda \mathbf{v}^T$ *can be translated equivalently into the Euclidean REP* $\mathbf{C}\mathbf{v} = \lambda \mathbf{v}$, *i.e.*

$$\mathbf{v}^T \diamond \mathbf{A} = \lambda \mathbf{v}^T \Longleftrightarrow \mathbf{C}\mathbf{v} = \lambda \mathbf{v}$$

holds.

Proof. The corollary immediately follows according to

$$\begin{aligned} \mathbf{C}\mathbf{v} = \lambda \mathbf{v} &\Longleftrightarrow (\mathbf{I} \diamond \mathbf{A})^T \mathbf{v} = \lambda \mathbf{v} \\ &\Longleftrightarrow \mathbf{v}^T (\mathbf{I} \diamond \mathbf{A}) = \lambda \mathbf{v}^T \\ &\Longleftrightarrow \mathbf{v}^T \diamond \mathbf{A} = \lambda \mathbf{v}^T \end{aligned}$$

applying the previous theorem with $\mathbf{Q} := \mathbf{I}$. ■

Thus it is possible to solve the sLEP using an arbitrary numerical approach for the Euclidean REP.

4 Conclusion

In this paper we briefly investigate the numerical solution of eigen-problems in Banach spaces, where no inner product is available as they are known from Hilbert spaces. Instead, semi-inner products with weaker requirements are the counterparts there. Whereas the right eigen-problem is difficult to handle, we present a solution for the left-side problem translating it to a right-side problem in an Euclidean space equipped with the Euclidean inner product.

References

[1] M. Biehl, M. Kästner, M. Lange, and T. Villmann. Non-Euclidean principal component analysis and Oja's learning rule – theoretical aspects. In P. Estevez, J. Principe, and P. Zegers, editors, *Advances in Self-Organizing Maps: 9th International Workshop WSOM 2012 Santiago de Chile*, volume 198 of *Advances in Intelligent Systems and Computing*, pages 23–34, Berlin, 2013. Springer.

[2] J. Giles. Classes of semi-inner-product spaces. *Transactions of the American Mathematical Society*, 129:436–446, 1967.

[3] I. Joliffe. *Principal Component Analysis*. Springer, 2nd edition, 2002.

[4] M. Lange, M. Biehl, and T. Villmann. Non-Euclidean principal component analysis by Hebbian learning. *Neurocomputing*, page in press, 2014.

[5] G. Lumer. Semi-inner-product spaces. *Transactions of the American Mathematical Society*, 100:29–43, 1961.

[6] E. Oja. Neural networks, principle components and subspaces. *International Journal of Neural Systems*, 1:61–68, 1989.

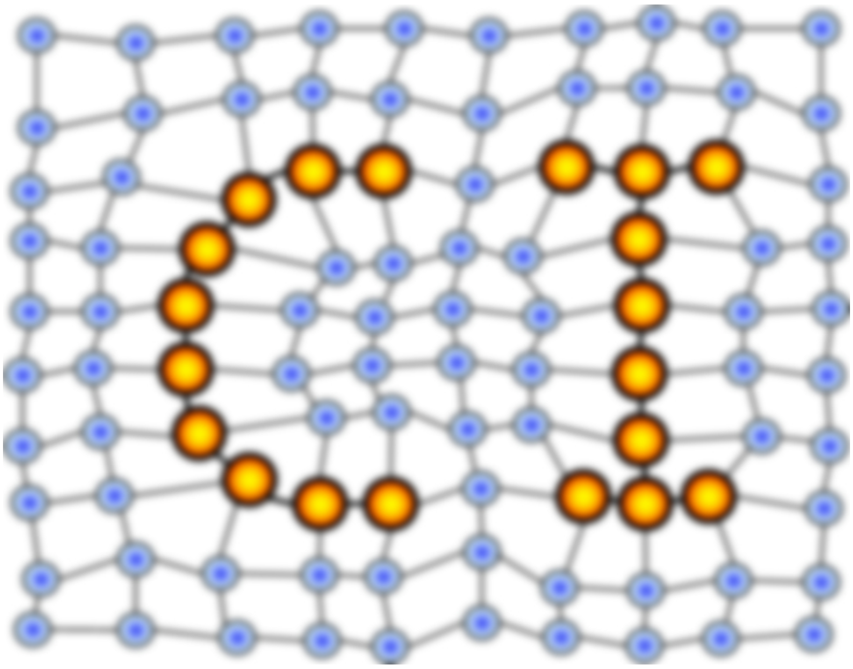
[7] E. Oja. Nonlinear PCA: Algorithms and applications. In *Proc. Of the World Congress on Neural Networks Portland*, pages 396–400, Portland, 1993.

[8] W. Press, S. Teukolsky, W. Vetterling, and B. Flannery. *Numerical Recipes in C*. Cambridge University Press, 1999.

[9] T. Sanger. Optimal unsupervised learning in a single-layer linear feedforward neural network. *Neural Networks*, 12:459–473, 1989.

Machine Learning Reports

Report 01/2014



Impressum

Machine Learning Reports ISSN: 1865-3960

▽ Publisher/Editors

Prof. Dr. rer. nat. Thomas Villmann
University of Applied Sciences Mittweida
Technikumplatz 17, 09648 Mittweida, Germany
• http://www.mni.hs-mittweida.de/

Dr. rer. nat. Frank-Michael Schleif
University of Birmingham
Edgbaston, B15 2TT Birmingham, UK,
• www.cs.bham.ac.uk/∼schleify/

▽ Acknowledgments

We would like to thank the reviewers for their time and patience.