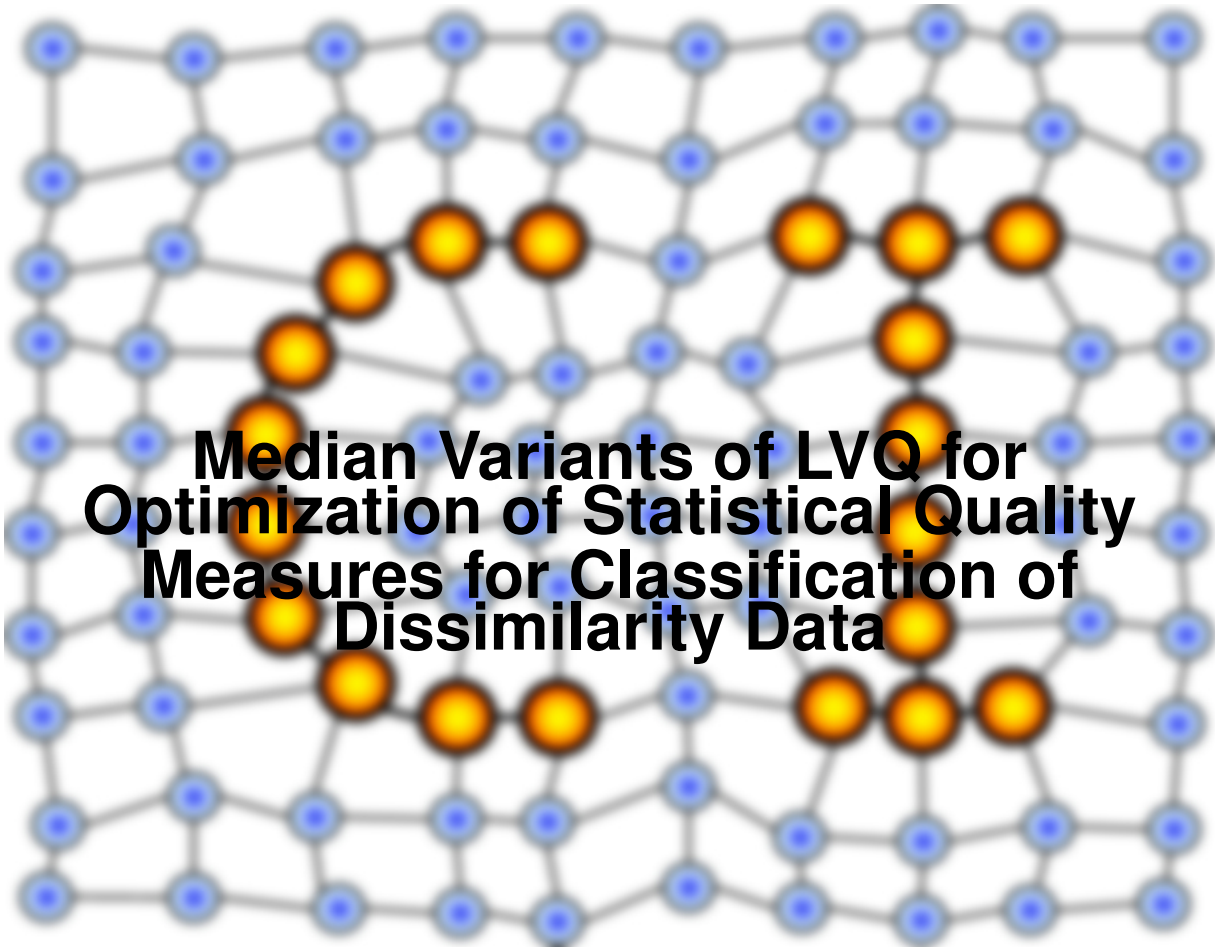
MACHINE LEARNING REPORTS



Median Variants of LVQ for Optimization of Statistical Quality Measures for Classification of Dissimilarity Data

Report 03/2014
Submitted: 01.09.2014
Published: 11.09.2014

David Nebel[1] and Thomas Villmann[1]
(1)University of Applied Sciences Mittweida, Fac. of Mathematics/Natural and Computer Sciences, Computational Intelligence Group, Mittweida, Technikumplatz 17, 09648 Mittweida, Germany

Abstract

We consider in this article median variants of the learning vector quantization classifier for classification of dissimilarity data. particularly we are interested in optimization of advanced classification quality measures like sensitivity, specificity or the F_β-measure. These measures are frequently more appropriate than simple accuracy, in particular, if the training data are imbalanced for the investigated data classes. We present the mathematical theory for this approach based on a generalized Expectation-Maximization-scheme.

Median Variants of LVQ for Optimization of Statistical Quality Measures for Classification of Dissimilarity Data

David Nebel and Thomas Villmann

University of Applied Sciences Mittweida,
Fac. of Mathematics/Natural and Computer Sciences,
Computational Intelligence Group,
Germany

Abstract

We consider in this article median variants of the learning vector quantization classifier for classification of dissimilarity data. particularly we are interested in optimization of advanced classification quality measures like sensitivity, specificity or the F_β-measure. These measures are frequently more appropriate than simple accuracy, in particular, if the training data are imbalanced for the investigated data classes. We present the mathematical theory for this approach based on a generalized Expectation-Maximization-scheme.

1 Introduction and Notations

In this section we briefly motivate the new median variants of learning vector quantization approach and clarify notations and abbreviations.

1.1 Introduction

Learning vector quantization (LVQ) as introduced by Teuvo Kohonen is a popular approach for classification of vector data [15, 16, 17]. The basic idea of this approach is to represent the data classes by prototype vectors. Many variants of the basic Hebbian learning scheme were developed since the initial work by Kohonen. An actual overview can be found in [14]. Yet, the main learning task, the optimization of the classification accuracy, as well as the differentiable dissimilarity measure in data space for comparison of prototypes and data were kept all the time.

Recently, the focus was shifted to more advanced classification goals like optimization of sensitivity, specificity or the F_β-measure developed by C.J. VAN RIJSBERGEN [24], which are based on the evaluation of the confusion matrix. These statistical quality measures are more adequate for class-imbalanced training data [13]. Sensitivity and specificity are closely related to the Receiver-Operating-Characteristics (ROC), which is an important tool for performance comparison of binary classifiers [5]. LVQ-like optimization of the area under the ROC-curve (AUROC) was proposed in [1, 2]. But still, the differentiable dissimilarity measure is necessary to derive the gradient based learning rules for the prototypes.

Thus, the topic of LVQ-extension for classification of dissimilarity or relational data emerged, as such variants are already known for unsupervised vector quantization [3, 7, 11, 10]. For relational approaches the prototypes are assumed as linear combination of the data. For general dissimilarity data prototypes are restricted to be data samples. The latter strategy is known as median-learning. First attempts for relational and median LVQ-variants optimizing the classification accuracy were provided in [8, 23, 22]. In the present publication, we extend these ideas to the previously mentioned statistical measures derived from the confusion matrix as well as to the ROC-analysis.

1.2 Notations and Abbreviations

In the following we clarify notation and abbreviations. We suppose data objects $\mathbb{X} = \{x_i\}_{i=1,\ldots,N}$ and M prototypes $\theta_k \in \Theta$, i.e. the cardinality of Θ is M. We assume a binary classification problem with the classes $C = \{\oplus, \ominus\}$. Let $c(\cdot)$ be the formal class label function, which assigns to each data object the class label $y_i = c(x_i)$. Analogously, $c_j = c(\theta_j)$ returns the predefined class label of the prototype. Further, M^+ denotes the number of prototypes assigned to the class $\oplus$. We introduce prototype dependent Kronecker-symbol abbreviations like

$$\delta_k^+ = \begin{cases} 1 \text{ if } & c_k = \oplus \\ 0 \text{ if } & c_k = \ominus \end{cases}$$

and

$$\delta_k^- = \begin{cases} 1 \text{ if } & c_k = \ominus \\ 0 \text{ if } & c_k = \oplus \end{cases}$$

as short-hand notations. Analogously, we define

$$\delta^+(x_i) = \begin{cases} 1 \text{ if } & y_i = \oplus \\ 0 \text{ if } & y_i = \ominus \end{cases}$$

and

$$\delta^-(x_i) = \begin{cases} 1 \text{ if } & y_i = \ominus \\ 0 \text{ if } & y_i = \oplus \end{cases}$$

as data dependent Kronecker-symbols. Furthermore, we define the set

$$X = \{(x_i, x_j) | y_i = \oplus \wedge y_j = \ominus\} \tag{1}$$

of all *ordered pairs* of data objects generated from $\mathbb{X}$. The *cardinality* of a set S is denoted by $|S|$.

2 The Mathematical Theory of the Generalized Expectation Maximization Approach

In this section we develop the general mathematical theory for maximization of a cost function $K(\mathbb{X})$ in the form

$$K(\mathbb{X}) = \sum_i g(x_i, \Theta) \tag{2}$$

with positive, bounded real functions $g(x_i, \Theta)$. It is based on the idea of the usual expectation-maximization (EM) strategy but adapted to the specific conditions. Later in this article it will be shown that median LVQ-variants for optimization of statistical measures and AUROC can be treated exactly in this manner.

Because the logarithm function is monotonically increasing the location of maximum for $K(\mathbb{X})$ is the same as for

$$\bar{K}(\mathbb{X}) = \ln\left(\sum_i g(x_i, \Theta)\right) \tag{3}$$

such that one can try to maximize the logarithmic cost function (LCF) $\bar{K}(\mathbb{X})$ instead of $K(\mathbb{X})$. Using the general functions $g(x_i, \Theta)$ we can formulate a formal probability

$$p(x_i) = \frac{g(x_i, \Theta)}{\sum_i g(x_i, \Theta)}$$

for a data object x_i. Furthermore, we assume arbitrary real numbers γ_i fulfilling the restrictions

$$\begin{aligned} \gamma_i &\geq 0 \\ \sum_{i=1}^{N} \gamma_i &= 1 \end{aligned}$$

which can be also interpreted as formal probability values. Thus we can define a formal Kullback-Leibler-divergence (KLD)

$$\mathcal{K}(\gamma || p) = \sum_i \gamma_i \ln\left(\frac{\gamma_i}{p(x_i)}\right) \tag{4}$$

being always non-negative and

$$\mathcal{K}(\gamma || p) = 0 \Leftrightarrow \gamma_i = p(x_i),\ \forall i$$

is valid [18]. Further, we define the loss term

$$\mathcal{L}(\gamma, \Theta) = \sum_i \gamma_i \ln\left(\frac{g(x_i, \Theta)}{\gamma_i}\right) \tag{5}$$

which allows a decomposition

$$\bar{K}(\mathbb{X}) = \mathcal{L}(\gamma, \Theta) + \mathcal{K}(\gamma||p) \tag{6}$$

of the LCF according to the following calculations:

$$\begin{aligned}
\bar{K}(\mathbb{X}) &= \mathcal{L}(\gamma, \Theta) + \mathcal{K}(\gamma||p) \\
&= \sum_i \gamma_i \ln\left(\frac{g(x_i, \Theta)}{\gamma_i}\right) + \sum_i \gamma_i \ln\left(\frac{\gamma_i}{p(x_i)}\right) \\
&= \sum_i \gamma_i \ln\left(\frac{g(x_i, \Theta)}{\gamma_i}\right) - \sum_i \gamma_i \ln\left(\frac{p(x_i)}{\gamma_i}\right) \\
&= \sum_i \gamma_i \ln\left(\frac{g(x_i, \Theta)}{\gamma_i}\right) - \sum_i \gamma_i \ln\left(\frac{g(x_i, \Theta)}{\left(\sum_k g(x_k, \theta)\right)\gamma_i}\right) \\
&= \sum_i \gamma_i \ln\left(\frac{g(x_i, \Theta)}{\gamma_i}\right) - \sum_i \gamma_i \ln\left(\frac{g(x_i, \Theta)}{\gamma_i}\right) - \sum_i \gamma_i \ln\left(\frac{1}{\left(\sum_k g(x_k, \theta)\right)}\right) \\
&= \sum_i \gamma_i \ln\left(\sum_k g(x_k, \theta)\right) \\
&= \underbrace{\left(\sum_i \gamma_i\right)}_{=1} \ln\left(\sum_k g(x_k, \theta)\right) \\
&= \ln\left(\sum_k g(x_k, \theta)\right) \\
&= \bar{K}(\mathbb{X})
\end{aligned}$$

At this point we recognize the fact that $\mathcal{L}(\gamma, \Theta)$ is a lower bound for the LCF $\bar{K}(\mathbb{X})$ due to the non-negativeness of the KLD $\mathcal{K}(\gamma||p)$. Using this property we obtain the following maximizing strategy for the LCF $\bar{K}(\mathbb{X})$:

1. **Expectation-step (E-step)**: set

$$\begin{aligned}
\gamma_i &:= p(x_i) \\
&\Rightarrow \\
\mathcal{K}(\gamma||p) &= 0 \\
&\Rightarrow \\
\bar{K}(\mathbb{X}) &= \mathcal{L}(\gamma, \Theta)
\end{aligned}$$

   Note that the cost function value $\bar{K}(\mathbb{X})$ does not change in this E-step, because $\bar{K}(\mathbb{X})$ is independent from the parameters γ_i.

2. ***generalized* Maximization-step (gM-step)**: take the parameters γ_i as fixed and find new prototypes Θ^{new}, such that:

$$\mathcal{L}(\gamma, \Theta^{new}) \quad \geq \quad \mathcal{L}(\gamma, \Theta^{old})$$

3. **Convergence criterion**: if $\Theta^{new} = \Theta^{old}$ stop. Else goto 1.

We remark that the new prototypes Θ^{new} maybe found by any search procedure. Thus it is not required to apply a gradient learning scheme. If we apply a sophisticated discrete search, with prototypes restricted to be selected from the data objects, a median-like optimization scheme is obtained. Further, because the new prototypes Θ^{new} have not to be maximizing the function $\mathcal{L}$, it is not a precise maximization step and, therefore, we denote it as a generalized M-step (gM-step) and the overall procedure a *generalized* EM-optimization (gEM).

3 LVQ-Classifier Functions and Confusion Matrix Entries

In the following we will consider statistical classifier functions based on the the confusion matrix as depicted in Tab.3.

		true		
		$\oplus$	$\ominus$	
predicted	$\oplus$	TP	FP	$\tilde{N}_+$
	$\ominus$	FN	TN	$\tilde{N}_-$
		N_+	N_-	N

Table 1: Confusion matrix

Thereby, the goal is to express the entries of the confusion matrix in terms of LVQ-classifier functions based on prototypes, which than can be used later for design of more complex statistical measures to be optimized by a LVQ approach. Here, the concrete choice of the classifier function depends on the considered LVQ-variant.

3.1 Classifier Functions of LVQ-Variants

As mentioned above, the concept of LVQ was introduced by Teuvo Kohonen in the late eighties of the last century. The original variants LVQ1 ... LVQ3 collected and described in [17] have in common that they heuristically optimize the crisp misclassification rate. However, the learning schemes are not mathematically exact optimization strategies. Several attempts were made to over come this disadvantage. The Generalized LVQ (GLVQ, [27]) approximates the classification error by a smoothed version to obtain a gradient descent learning scheme. Probabilistic LVQ-classifier schemes are the Robust Soft LVQ (RSLVQ,

[30]) and the Soft Nearest Prototype Classifier (SNPC, [29]) keeping the idea of prototypes but relaxing the restriction of crisp classification.

Interestingly, RSLVQ and GLVQ, are based on so-called classifier function, which we will use later to determine the values of the confusion matrix entries. Therfore, we consider them more detailed in the following.

3.1.1 Robust Soft LVQ Classifier Function

A probabilistic LVQ-classifier based on a likelihood ratio cost function is the Robust Soft LVQ (RSLVQ) [30]. In particular, RSLVQ represents data in terms of a mixture model with the prototypes $\Theta = \{\theta_1, \ldots, \theta_M\}$ taking as model parameters. For this purpose, a RSLVQ-classifier function $\mu_{RSLVQ}(\kappa|x_i)$ is considered describing that a data object x_i is assigned to class $\kappa \in C$.

Supposing a binary RSLVQ classifier, the probability that an arbitrary data point is assigned to the class $\oplus$ by RSLVQ is given by

$$p(\oplus|x_i, \Theta) = \frac{\sum_j \delta_j^+ p(x_i|\theta_j)}{\sum_k p(x_i|\theta_k)}$$

as conditional mixture model, where the conditional probabilities

$$p(x_i|\theta_j) = \exp\left(-\left(\frac{d(x_i, \theta_j)}{\sigma_j}\right)^2\right) \tag{7}$$

are Gaussians with width's $\sigma_j > 0$. Here, $d(x_i, \theta_j)$ is a dissimilarity measure between data objects and prototypes. Then

$$\mu_{RSLVQ}(\oplus|x_i) = p(\oplus|x_i, \Theta) \tag{8}$$

is determined. Analogously, the probability that an arbitrary data point is assigned to the class $\ominus$ is given by

$$p(\ominus|x_i, \Theta) = \frac{\sum_j \delta_j^- p(x_i|\theta_j)}{\sum_k p(x_i|\theta_k)}$$

as conditional mixture model and

$$\mu_{RSLVQ}(\ominus|x_i) = p(\ominus|x_i, \Theta)\,. \tag{9}$$

Note at this point that if prototypes are restricted to be data objects, i.e $\Theta \subseteq \mathbb{X}$, only the dissimilarities between the data objects are required to calculate both probabilities $p(\oplus|x_i, \Theta)$ and $p(\ominus|x_i, \Theta)$.

To keep the model simple, we assume $\sigma = \sigma_j$ for all $j = 1 \ldots M$ in the following. Now we make the important observation that in the limit $\sigma \to 0$ the conditional probability $p(\oplus|x_i, \Theta)$ becomes crisp, i.e. we have

$$p(\oplus|x_i, \Theta) \underset{\sigma \to 0}{\longrightarrow} \begin{cases} 1 & \text{if } y_i = \oplus \\ 0 & \text{else} \end{cases}$$

and, therefore, $p(\oplus|x_i, \Theta)$ is an indicator function for the class $\oplus$ in this limit. Analogously, we have

$$p(\ominus|x_i, \Theta) \underset{\sigma \to 0}{\longrightarrow} \begin{cases} 1 & \text{if } y_i = \ominus \\ 0 & \text{else} \end{cases}$$

for the the conditional probability $p(\ominus|x_i, \Theta)$.

Hence, both quantities $\mu_{RSLVQ}(\oplus|x_i)$ and $\mu_{RSLVQ}(\ominus|x_i)$ can be used to count approximately the correctly classified data objects x_i.

3.1.2 Generalized LVQ Classifier Function

The objective of the GLVQ is to optimize the hypothesis margin [4, 9, 28]. It is based on the classifier function $\mu_{GLVQ}(\kappa|x_i)$ determining whether a data object x_i is assigned to the class $\kappa \in C$. For a binary classifier it is defined as

$$\mu_{GLVQ}(\oplus|x_i) = f\left(\frac{d_i^- - d_i^+}{d_i^+ + d_i^-}\right)$$

and

$$\mu_{GLVQ}(\ominus|x_i) = f\left(\frac{d_i^+ - d_i^-}{d_i^+ + d_i^-}\right)$$

where

$$d_i^+ = \min_{\theta_k : c_k = \oplus} d(x_i, \theta_k)$$

and

$$d_i^- = \min_{\theta_k : c_k = \ominus} d(x_i, \theta_k)$$

with $d(x_i, \theta_k)$ again being a dissimilarity measure. The function f is assumed to be monotonically increasing [27]. Thus, d_i^+ is the dissimilarity of the given data object x_i to the best matching prototype responsible for class $\oplus$ and d_i^- is the equivalent value for the class $\ominus$. Thus, the classifier function $\mu_{GLVQ}(\kappa|x_i)$ is positive, if the object class y_i matches the considered class κ, i.e. $y_i = \kappa$.

In the case that $f(z)$ is the Heaviside function $H(z)$, the classifier function $\mu_{GLVQ}(\kappa|x_i)$ detects all correctly classified data objects x_i. Note that the Heaviside function can be approximated by sigmoid function

$$sgd(z) = \frac{1}{1 + \exp(-\frac{z}{\sigma})}$$

in the limit $0 \leq \sigma \searrow 0$. Hence, the classifier function $\mu_{GLVQ}(\kappa|x_i)$ can be used to count correctly classified data objects in this approximation.

3.2 Approximation of the Confusion Matrix Entries Using the Classifier Functions

As we have seen in the previous subsections, both classifier functions $\mu_{RSLVQ}(\kappa|x_i)$ and $\mu_{GLVQ}(\kappa|x_i)$ can be used to detect correctly classified data objects x_i. To unify these approaches, we simply use the notation $\mu(\kappa|x_i)$ in the following considerations.

With the introduced classifier functions $\mu(\oplus|x_i)$ and $\mu(\ominus|x_i)$ we are now able to calculate all entries of the confusion matrix from Tab.3. In particular, we obtain

$$\begin{aligned} TP &= \sum_i \delta^+(x_i) \cdot \mu(\oplus|x_i) \\ FP &= \sum_i \delta^-(x_i) \cdot \mu(\oplus|x_i) \\ FN &= \sum_i \delta^+(x_i) \cdot \mu(\ominus|x_i) \\ TN &= \sum_i \delta^-(x_i) \cdot \mu(\ominus|x_i) \end{aligned}$$

as LVQ-based quantities. Again, we emphasize at this point that only dissimilarities between the data objects and the prototypes are required for the calculation of these quantities. However, this approach is similar to the approach as presented for vector data and prototypes in [13].

4 Median-LVQ-variants for Optimization of Statistical Measures based on the Confusion Matrix

In this chapter we will describe several statistical quality measures for classification in the form of (2), which allows to apply the gEM-optimization scheme provided in Sec.2. For this purpose, we will use the approximation of the entries of the confusion matrix introduced in Sec.3.2

4.1 Simple Classification Quality Measures

In this sub-chapter we consider simple quality measure, which are directly derived from the confusion matrix. We will write them in the form of (2) and specify the respective choice of $g(x_i, \Theta)$.

Sensitivity ρ or true positive rate (TPR) The sensitivity ρ, some times also called *recall*, is given by

$$\begin{aligned} \rho &= \frac{TP}{TP + FN} \\ &= \frac{1}{N^+} \sum_i g(x_i, \Theta) \end{aligned} \tag{10}$$

with $g(x_i, \Theta) = \delta^+(x_i) \cdot \mu(\oplus|x_i)$.

- range of values

$$0 \leq TPR \leq 1$$

$$0 \leq g(x_i, \Theta) \leq 1$$

- maximization/positivity/numerical remarks
  - The sensitivity ρ has to be maximized. The resulting functions $g(x_i, \Theta)$ are positive.
  - For numerical reasons it is be better to maximize
  - $$\frac{1}{N^+} \sum_i (g(x_i, \Theta) + 1) = \frac{1}{N^+} \sum_i \bar{g}(x_i, \Theta)$$

    instead of the term $\frac{1}{N^+} \sum_i g(x_i, \Theta)$ to avoid numerical instabilities due to the evaluation of the logarithm in (3).

Specificity ς or true negative rate (TNR) The specificity ς writes as

$$\begin{aligned} \varsigma &= \frac{TN}{FP + TN} \\ &= \frac{1}{N^-} \sum_i g(x_i, \Theta) \end{aligned} \tag{11}$$

with $g(x_i, \Theta) = \delta^-(x_i) \cdot \mu(-|x_i)$.

- range of values

$$0 \leq \varsigma \leq 1$$

$$0 \leq g(x_i, \Theta) \leq 1$$

- maximization/positivity/numerical remarks
  - The statistical value ς has to be maximized. The resulting functions $g(x_i, \Theta)$ are positive.
  - For numerical reasons it is be better to maximize
  - $$\frac{1}{N^-} \sum_i (g(x_i, \Theta) + 1) = \frac{1}{N^-} \sum_i \bar{g}(x_i, \Theta)$$

    instead of $\sum_i g(x_i, \Theta)$ to avoid numerical instabilities due to the evaluation of the logarithm in (3).

Precision π or positive predictive value (PPV) The precision π is given as

$$\begin{aligned} \pi &= \frac{TP}{TP+FP} \\ &= \sum_i g(x_i, \Theta) \end{aligned} \tag{12}$$

with

$$g(x_i, \Theta) = \frac{\delta^+(x_i) \cdot \mu(\oplus|x_i)}{\sum_j \delta^+(x_j) \cdot \mu(\oplus|x_j) + \sum_j \delta^-(x_j) \cdot \mu(\oplus|x_j)}$$

- range of values

$$0 \leq \pi \leq 1$$

$$0 \leq g(x_i, \Theta) \leq 1$$

- maximization/positivity/numerical remarks
  - The statistical value π has to be maximized. The resulting functions $g(x_i, \Theta)$ are positive.
  - For numerical reasons it is be better to maximize

$$\sum_i (g(x_i, \Theta) + 1) = \sum_i \bar{g}(x_i, \Theta)$$

    instead of $\sum_i g(x_i, \Theta)$ to avoid numerical instabilities due to the evaluation of the logarithm in (3).

Negative prediction value ν (NPV) The negative prediction value ν is defined as

$$\begin{aligned} \nu &= \frac{TN}{TN+FN} \\ &= \sum_i g(x_i, \Theta) \end{aligned} \tag{13}$$

with

$$g(x_i, \Theta) = \frac{\delta^-(x_i) \cdot \mu(\ominus|x_i)}{\sum_j \delta^-(x_j) \cdot \mu(\ominus|x_j) + \sum_i \delta^+(x_j) \cdot \mu(\ominus|x_j)}$$

- range of values

$$0 \leq \nu \leq 1$$

$$0 \leq g(x_i, \Theta) \leq 1$$

- maximization/positivity/numerical remarks
  - The statistical value ν has to be maximized. The resulting functions $g(x_i, \Theta)$ are positive.

– For numerical reasons it is be better to maximize

$$\sum_i \left(g(x_i,\Theta)+1\right) = \sum_i \bar{g}(x_i,\Theta)$$

instead of $\sum_i g(x_i,\Theta)$ to avoid numerical instabilities due to the evaluation of the logarithm in (3).

Fall-out or false positive rate (FPR) The false positive rate (FPR) is given as

$$\begin{aligned} \varphi &= \frac{FP}{FP+TN} \\ &= 1-\mu \\ &= \frac{1}{N_-}\sum_i g(x_i,\Theta) \end{aligned} \tag{14}$$

with $g(x_i,\Theta) = \delta^-(x_i)\cdot\mu(\oplus|x_i)$.

- range of values

$$0 \leq FPR \leq 1$$

$$0 \leq g(x_i,\Theta) \leq 1$$

- maximization/positivity/numerical remarks

  – The statistical value FPR has to be minimized. To get a maximization problem and to ensure the positivity of the functions $g(x_i,\Theta)$ as well a numerical stable cost we maximize

$$\frac{1}{N_-}\sum_i \left(2-g(x_i,\Theta)\right) = \frac{1}{N_-}\sum_i \bar{g}(x_i,\Theta)$$

  instead of the minimization of $\frac{1}{N_-}\sum_i g(x_i,\Theta)$

False discovery rate (FDR) The false discovery rate is defined as

$$\begin{aligned} FDR &= \frac{FP}{FP+TP} \\ &= \sum_i g(x_i,\Theta) \end{aligned}$$

with

$$g(x_i,\Theta) = \frac{\delta^-(x_i)\cdot\mu(\oplus|x_i)}{\sum_j \delta^-(x_j)\cdot\mu(\oplus|x_j) + \sum_j \delta^+(x_j)\cdot\mu(\oplus|x_j)}$$

- range of values

$$0 \leq FDR \leq 1$$

$$0 \leq g(x_i,\Theta) \leq 1$$

- maximization/positivity/numerical remarks
  - The statistical value FDR has to be minimized.
  - To get a maximization problem and to ensure the positivity of the functions $g(x_i, \Theta)$ as well a numerical stabel cost we maximize

$$\sum_i (2 - g(x_i, \Theta)) = \sum_i \bar{g}(x_i, \Theta)$$

    instead of the minimization of $\sum_i g(x_i, \Theta)$

Miss Rate or False Negative Rate (FNR) The false negative rate is given by

$$\begin{aligned} FNR &= \frac{FN}{FN + TP} \\ &= \frac{FN}{N_+} \\ &= \frac{1}{N_+} \sum_i g(x_i, \Theta) \end{aligned}$$

with $g(x_i, \Theta) = \delta^+(x_i) \cdot \mu(\ominus | x_i)$

- range of values

$$0 \leq FNR \leq 1$$

$$0 \leq g(x_i, \Theta) \leq 1$$

- maximization/positivity/numerical remarks
- The statistical value FNR has to be minimized.
- To get a maximization problem and to ensure the positivity of the functions $g(x_i, \Theta)$ as well a numerical stable cost we maximize

$$\frac{1}{N_+} \sum_i (2 - g(x_i, \Theta)) = \frac{1}{N_+} \sum_i \bar{g}(x_i, \Theta)$$

  instead of the minimization of $\frac{1}{N_+} \sum_i g(x_i, \Theta)$.

4.2 F_β-measure

The F_β-measure developed by C.J. VAN RIJSBERGEN [24] is an advanced quality measure. It combines precision π from (12) and recall (sensitivity) from (10) into a single quantity

$$F_\beta = \frac{(1 + \beta^2)\pi\rho}{\beta^2\pi + \rho} \tag{15}$$

depending on the balancing parameter β. Frequently, this balancing parameter is chosen as $\beta = 2$ yielding the measure to be the ratio of the arithmetic and the geometric mean between both quantities precision and recall. Using the quantities from the confusion matrix, we can rewrite (15) as

$$\begin{aligned} F_\beta &= \frac{(1+\beta^2)TP}{(1+\beta^2)\,TP + \beta^2 FN + FP} \\ &= \sum_i g(x_i, \Theta) \end{aligned}$$

with

$$g(x_i, \Theta) = \frac{\delta^+(x_i) \cdot (1+\beta^2)\mu(+|x_i)}{\sum_j \left((1+\beta^2)\,\delta^+(x_j) \cdot \mu(\oplus|x_j) + \beta^2 \delta^+(x_j) \cdot \mu(\ominus|x_j) + \delta^-(x_j) \cdot \mu(\oplus|x_j)\right)}$$

- range of values

$$0 \leq F_\beta \leq 1$$

$$0 \leq g(x_i, \Theta) \leq 1$$

- maximization/positivity/numerical remarks
  - The statistical value F_β has to be maximized. The resulting functions $g(x_i, \Theta)$ are positive.
  - For numerical reasons it could be better to maximize

$$\sum_i \left(g(x_i, \Theta) + 1\right) = \sum_i \bar{g}(x_i, \Theta)$$

  instead of $\sum_i g(x_i, \Theta)$ to avoid numerical problems.

4.3 Jaccard Index

A widely used measure is the Jaccard-index

$$J = \frac{TP}{FP + TP + FN}$$

as explained in [12, 6]. It is related to the Tanimoto distances [25]. Again, we rewrite this index J in the form

$$J = \sum_i g(x_i, \Theta)$$

with

$$g(x_i, \Theta) = \frac{\delta^+(x_i) \cdot \mu(\oplus|x_i)}{\sum_j \left(\delta^-(x_j) \cdot \mu(\oplus|x_j) + \delta^+(x_j) \cdot \mu(\oplus|x_j) + \delta^+(x_j) \cdot \mu(\ominus|x_j)\right)}$$

- range of values

$$0 \leq J \leq 1$$

$$0 \leq g(x_i, \Theta) \leq 1$$

- maximization/positivity/numerical remarks

  - The statistical value J has to be maximized. The resulting functions $g(x_i, \Theta)$ are positive. For numerical stability reasons it is better to maximize

$$\sum_i \left(g(x_i, \Theta) + 1 \right) = \sum_i \bar{g}(x_i, \Theta)$$

    instead of $\sum_i g(x_i, \Theta)$

.

4.4 Averaged conditional classification probability

The averaged conditional classification probability (ACP) is the weighted average

$$\begin{array}{rcl} ACP & = & \alpha\rho + \beta\pi + \gamma\nu + \eta\varsigma \\ & \text{with} & \alpha + \beta + \gamma + \eta = 1 \end{array}$$

of recall, precision, negative prediction value ν, and specificity ς with non-negative weights $\alpha, \beta, \gamma, \eta \geq 0$ and the normalization condition $\alpha + \beta + \gamma + \eta = 1$. We calculate

$$ACP = \sum_i g(x_i, \Theta)$$

with

$$\begin{array}{rcl} g(x_i, \Theta) & = & \alpha \dfrac{\delta^+(x_i) \cdot \mu(\oplus|x_i)}{N^+} + \beta \dfrac{\delta^+(x_i) \cdot \mu(\oplus|x_i)}{\sum_i \delta^+(x_j) \cdot \mu(\oplus|x_j) + \sum_i \delta^-(x_j) \cdot \mu(\oplus|x_j)} \\ & & + \gamma \dfrac{\delta^-(x_i) \cdot \mu(\ominus|x_i)}{\sum_i \delta^-(x_j) \cdot \mu(\ominus|x_j) + \sum_i \delta^+(x_j) \cdot \mu(\ominus|x_j)} + \eta \dfrac{\delta^-(x_i) \cdot \mu(\ominus|x_i)}{N^-} \end{array}$$

- range of values

$$0 \leq ACP < N$$

$$0 \leq g(x_i, \Theta) \leq N$$

- maximization/positivity/numerical remarks

  - The statistical value ACP has to be maximized. The resulting functions $g(x_i, \Theta)$ are positive.
  - For numerical reasons it is better to maximize $\sum_i \bar{g}(x_i, \Theta) = \sum_i \left(g(x_i, \Theta) + 1 \right)$

4.5 Matthews correlation coefficient

Another popular measure is the Matthews correlation coefficient

$$MCC = \frac{TP \cdot TN - FP \cdot FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}} \tag{16}$$

which is equivalent to the χ^2-statistics for a 2×2 contingency table [21]. In particular,

$$|MCC| = \sqrt{\frac{\chi^2}{N}} \tag{17}$$

is valid [21, 26]. We can write the MCC as

$$\begin{aligned} MCC &= \frac{\sum_i \delta^+(x_i) \cdot \mu(\oplus|x_i) \cdot \sum_j \delta^-(x_j) \cdot \mu(\ominus|x_j) - \sum_i \delta^-(x_i) \cdot \mu(\oplus|x_i) \cdot \sum_j \delta^+(x_j) \cdot \mu(\ominus|x_j)}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}} \\ &= \frac{\sum_{i,j} (\delta^+(x_i) \cdot \delta^-(x_j) \cdot \mu(\oplus|x_i)\mu(\ominus|x_j)) - \sum_{i,j} (\delta^-(x_i) \cdot \delta^+(x_j) \cdot \mu(\oplus|x_i)\mu(\ominus|x_j))}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}} \\ &= \sum_{i,j} g(x_i, x_j, \Theta) \end{aligned}$$

with

$$g(x_i, x_j, \Theta) = \frac{\delta^+(x_i) \cdot \delta^-(x_j) \cdot \mu(\oplus|x_i)\mu(\ominus|x_j) - \delta^-(x_i) \cdot \delta^+(x_j) \cdot \mu(\oplus|x_i)\mu(\ominus|x_j)}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}}$$

- range of values

$$-1 \leq MCC \leq 1$$

$$-1 \leq g(x_i, \Theta) \leq 1$$

- maximization/positivity/numerical remarks
  - The statistical value MCC has to be maximized. The resulting functions $g(x_i, \Theta)$ are not positive.
  - To ensure positivity and numerical stability it is recommended to maximize

$$\sum_{i,j} (g(x_i, x_j, \Theta) + 2) = \sum_i \bar{g}(x_i, \Theta)$$

    instead of original MCC.

4.6 ROC Analysis (area under the curve)

The Receiver Operator Characteristic (ROC) is a graphical tool for comparison of classifiers with respect to their performance [5]. These performances are measured in terms of the true positive rate (recall/sensitivity) ρ from (10) and the false positive rate φ from (14).

If a parametrized classifier is considered, the resulting pairs of these values may be plotted into two-dimensional diagram - the so-called ROC-curve. The area A_{ROC} under this ROC-curve (AUROC) is a performance measure for this parametrized classifier. The higher the AUROC-value, the better the classifier. Assuming a binary classifier with with classifier function $\mu(\kappa|x_i)$. Then the area A_{ROC} has a probabilistic interpretation:

$$A_{ROC} = P(\mu(\oplus|x_i) > \mu(\oplus|x_j)) \tag{18}$$

for a randomly chosen *ordered* pair $(x_i, x_j) \in X$ of the data set $\mathbb{X}$ as defined in (1) [5], which yields due to the underlying rank statistics [20, 31]. If we define for the prototype-based classifier the ordering function

$$O(x_i, x_j) = H(\mu(\oplus|x_i) - \mu(\oplus|x_j)) \tag{19}$$

where H is the Heaviside function

$$H(x) = \begin{cases} 0 & if \quad x \leq 0 \\ 1 & else \end{cases}, \tag{20}$$

th probability P in (18) can be estimated by:

$$P = \frac{1}{|X|} \sum_{(x_i,x_j)\in X} O(x_i, x_j)$$

as proposed in [1].

gEM for AUC-optimizing classifier

A binary GLVQ-variant for data vectors $x_k \in \mathbb{X} \subset \mathbb{R}^n$ was recently published in [1, 2]. Here we propose a respective median variant using a gEM-scheme, only assuming a prototype based binary classifier with classifier function $\mu(\kappa|x_i)$.

With the same arguments as in 2 we can maximize the following logarithmic probability

$$P_{log} = \ln\left(\frac{1}{|X|}\sum_{X} O(x_i, x_j))\right)$$

taking the formal probabilities $p((x_i, x_j))$ for P as

$$p((x_i, x_j)) = \frac{g((x_i, x_j), \Theta)}{\sum_{l,k} g((x_l, x_k), \Theta)}$$

with

$$g((x_i, x_j), \Theta) = O(x_i, x_j)$$

for the respective gEM scheme to maximize the estimated area A_{ROC} under the ROC-curve.

- range of values

$$0 \leq P \leq 1$$
$$0 \leq g\left(\left(x_i, x_j\right), \Theta\right) \leq 1$$

- maximization/positivity/numerical remarks
  - The resulting functions $g(x_i, \Theta)$ are positive.
  - For numerical stability it is better to maximize

$$\ln\left(\frac{1}{|X|}\sum_X \left(g\left(\left(x_i, x_j\right), \Theta\right) + 1\right)\right) \;=\; \ln\left(\frac{1}{|X|}\sum_X \bar{g}\left(\left(x_i, x_j\right), \Theta\right)\right)$$

    instead of P_{log}

4.7 The specificity related geometric mean (G-measure)

We define two G-measures for a classifier as the geometric mean of the specificity ς from (11) with the negative prediction value ν from (13)

$$G_\nu = \sqrt{\varsigma\nu}$$

and the recall ρ from (10)

$$G_\rho = \sqrt{\varsigma\rho}$$

respectively. These G-measures are mathematically interesting, because the square root requires special treatment for an appropriate gEM-scheme decomposition in comparison to the gEM-scheme decomposition developed for the logarithm function, involved in the measure so far in the previous chapters.

For this purpose, we start with the G_ρ-measure and obtain

$$\begin{aligned} G_\rho &= \sqrt{\frac{TN}{FP+TN}\frac{TP}{TP+FN}} \\ &= \sqrt{\sum_{i,j} g_\rho(x_i, x_j)} \end{aligned}$$

with the non-negative function

$$g_\rho(x_i, x_j) = \frac{\delta^-\left(x_i\right) \cdot \delta^+\left(x_j\right) \cdot \mu(-|x_i)\mu(+|x_j)}{N^+N^-}$$

inside the sum. Assuming arbitrary non-negative numbers $\gamma_{i,j} \geq 0$ with the additional restriction $\sum_{i,j} \gamma_{i,j} = 1$ and using the Jensen-inequality for concave functions [19], we get

$$\begin{aligned} G_\rho &= \sqrt{\sum_{i,j} \gamma_{i,j}\left(\frac{g_\rho(x_i, x_j)}{\gamma_{i,j}}\right)} \\ &\geq \sum_{i,j} \gamma_{i,j}\sqrt{\frac{g(x_i, x_j)}{\gamma_{i,j}}} \end{aligned} \tag{21}$$

We determine

$$\mathcal{L}_\rho(\gamma,\Theta)=\sum_{i,j}\gamma_{i,j}\sqrt{\frac{g_\rho(x_i,x_j)}{\gamma_{i,j}}}$$

and

$$R_\rho(\gamma)=G_\rho-\mathcal{L}_\rho(\gamma,\Theta)$$

such that $\mathcal{L}_\rho(\gamma,\Theta)$ determines a lower bound for

$$G_\rho=\mathcal{L}_\rho(\gamma,\Theta)+R_\rho(\gamma)$$

because $R_\rho(\gamma)\geq 0$ is valid due to the Jensen-inequality (21).

The G_ν-measure can be decomposed analogously using the respective quantities $g_\nu(x_i,x_j)$, $\mathcal{L}_\nu(\gamma,\Theta)$, and $R_\nu(\gamma)$.

gEM for G-measures

In the following we will not distinguish between G_ν and G_ρ, because they can be treated equivalently. Thus, we generally suppose a decomposition

$$G=\mathcal{L}(\gamma,\Theta)+R(\gamma)$$

with an underlying non-negative function $g(x_i,x_j)$. For the special choice of $\gamma_{i,j}=\frac{g(x_i,x_j)}{\sum_{k,l}g(x_k,x_l)}$ the value of the lower bound $\mathcal{L}(\gamma,\Theta)$ is equal to the costfunction value of G:

$$\begin{aligned}
\mathcal{L}(\gamma,\Theta) &= \sum_{i,j}\gamma_{i,j}\sqrt{\frac{g(x_i,x_j)}{\gamma_{i,j}}}\\
&= \sum_{i,j}\gamma_{i,j}\sqrt{\frac{g(x_i,x_j)}{\frac{g(x_i,x_j)}{\sum_{k,l}g(x_k,x_l)}}}\\
&= \sum_{i,j}\gamma_{i,j}\sqrt{\sum_{k,l}g(x_k,x_l)}\\
&= \underbrace{\left(\sum_{i,j}\gamma_{i,j}\right)}_{=1}\sqrt{\sum_{k,l}g(x_k,x_l)}\\
&= \sqrt{\sum_{k,l}g(x_k,x_l)}\\
&= G
\end{aligned}$$

From this it follows immediately that $R(\gamma)=0$ is valid for this special choice of $\gamma_{i,j}$.

Utilization of these properties as before in chapter (2) results in a generalized maximizing strategy:

1. **Expectation-step (E-step)**: set

$$\begin{aligned} \gamma_{i,j} &:= \frac{g(x_i, x_j)}{\sum_{k,l} g(x_k, x_l)} \\ &\Rightarrow \\ R(\gamma) &= 0 \\ &\Rightarrow \\ G &= \mathcal{L}(\gamma, \Theta) \end{aligned}$$

   The cost function value remains unchanged, because G does not depend on the parameters $\gamma_{i,j}$.

2. ***generalized* Maximization-step (gM-step)**: fix the parameter $\gamma_{i,j}$ and find new prototypes Θ^{new}, such that

$$\mathcal{L}(\gamma, \Theta^{new}) \geq \mathcal{L}(\gamma, \Theta^{old})$$

   is valid

3. if $\Theta^{new} = \Theta^{old}$ stop. Else goto 1.

Again we obtain a *generalized* EM-optimization scheme, because the new prototypes Θ^{new} have not to be maximizing the function $\mathcal{L}$ as required for a precise maximization.

5 Conclusion

In this contribution we provide the mathematical framework for prototype-based classifiers derived from GLVQ for general dissimilarity data of data objects. The classifier cost functions investigated here are several statistical classification quality measures and the prototypes of the classifiers are required to be data exemplars. Thus, so-called median-variants are studied. The statistical quality measures for classification include accuracy, sensitivity, specificity, F_β-measure, area under the ROC-curve and other. The respective underlying optimization scheme is a generalized Expectation-Maximization-procedure.

Acknowledgment

D. Nebel was supported by a grant of the European Social Fund, Saxony (ESF).

References

[1] M. Biehl, M. Kaden, P. Stürmer, and T. Villmann. ROC-optimization and statistical quality measures in learning vector quantization classifiers. *Machine Learning Reports*, 8(MLR-01-2014):23–34, 2014. ISSN:1865-3960, http://www.techfak.uni-bielefeld.de/~fschleif/mlr/mlr_01_2014.pdf.

[2] M. Biehl, M. Kaden, and T. Villmann. Statistical quality measures and ROC-optimization by learning vector quantization classifiers. In H. Kestler, M. Schmid, L. Lausser, and J. Kraus, editors, *Proceedings of the 46th Workshop on Statistical Computing (Ulm/Reisensburg 2014)*, number 2014-04 in Ulmer Informatik-Berichte, pages 1–6. University Ulm, Germany, 2014.

[3] M. Cottrell, B. Hammer, A. Hasenfuß, and T. Villmann. Batch and median neural gas. *Neural Networks*, 19:762–771, 2006.

[4] K. Crammer, R. Gilad-Bachrach, A. Navot, and A.Tishby. Margin analysis of the LVQ algorithm. In S. Becker, S. Thrun, and K. Obermayer, editors, *Advances in Neural Information Processing (Proc. NIPS 2002)*, volume 15, pages 462–469, Cambridge, MA, 2003. MIT Press.

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

[6] T. Geweniger, L. Fischer, M. Kaden, M. Lange, and T. Villmann. Clustering by fuzzy neural gas and evaluation of fuzzy clusters. *Computational Intelligence and Neuroscience*, 2013:Article ID 165248, 2013. http://dx.doi.org/10.1155/2013/165248.

[7] B. Hammer and A. Hasenfuss. Topographic mapping of large dissimilarity data sets. *Neural Computation*, 22(9):2229–2284, 2010.

[8] B. Hammer, D. Hofmann, F.-M. Schleif, and X. Zhu. Learning vector quantization for (dis-)similarities. *Neurocomputing*, page in press, 2013.

[9] B. Hammer, M. Strickert, and T. Villmann. Relevance LVQ versus SVM. In L. Rutkowski, J. Siekmann, R. Tadeusiewicz, and L. Zadeh, editors, *Artificial Intelligence and Soft Computing (ICAISC 2004)*, Lecture Notes in Artificial Intelligence 3070, pages 592–597. Springer Verlag, Berlin-Heidelberg, 2004.

[10] R. Hathaway and J. Bezdek. NERF c-means: Non-Euclidean relational fuzzy clustering. *Pattern recognition*, 27(3):429–437, 1994.

[11] R. Hathaway, J. Davenport, and J. Bezdek. Relational duals of the c-means clustering algorithms. *Pattern recognition*, 22(3):205–212, 1989.

[12] P. Jaccard. The distribution of the flora in the alpine zone. *New Phytologist*, 11:37–50, 1912.

[13] M. Kaden, W. Hermann, and T. Villmann. Optimization of general statistical accuracy measures for classification based on learning vector quantization. In M. Verleysen, editor, *Proc. of European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (ESANN'2014)*, pages 47–52, Louvain-La-Neuve, Belgium, 2014. i6doc.com.

[14] 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.

[15] T. Kohonen. Learning vector quantization for pattern recognition. Report TKK-F-A601, Helsinki University of Technology, Espoo, Finland, 1986.

[16] T. Kohonen. Learning Vector Quantization. *Neural Networks*, 1(Supplement 1):303, 1988.

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

[18] S. Kullback and R. Leibler. On information and sufficiency. *Annals of Mathematical Statistics*, 22:79–86, 1951.

[19] D. Mackay. *Information Theory, Inference and Learning Algorithms*. Cambridge University Press, 2003.

[20] 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.

[21] B. Matthews. Comparison of the predicted and observed secondary structure of T4 phage lysozyme. *Biochimica et Biophysica Acta*, 405:442–451, 1975.

[22] 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.

[23] D. Nebel and T. Villmann. A median variant of generalized learning vector quantization. In M. Lee, A. Hirose, Z.-G. Hou, and R. Kil, editors, *Proceedings of International Conference on Neural Information Processing (ICONIP)*, volume II of *LNCS*, pages 19–26, Berlin, 2013. Springer-Verlag.

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

[25] D. J. Rogers and T. Tanimoto. A computer program for classifying plants. *Science*, 132(3434):1115–1118, 1960.

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

[27] 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.

[28] P. Schneider, B. Hammer, and M. Biehl. Distance learning in discriminative vector quantization. *Neural Computation*, 21:2942–2969, 2009.

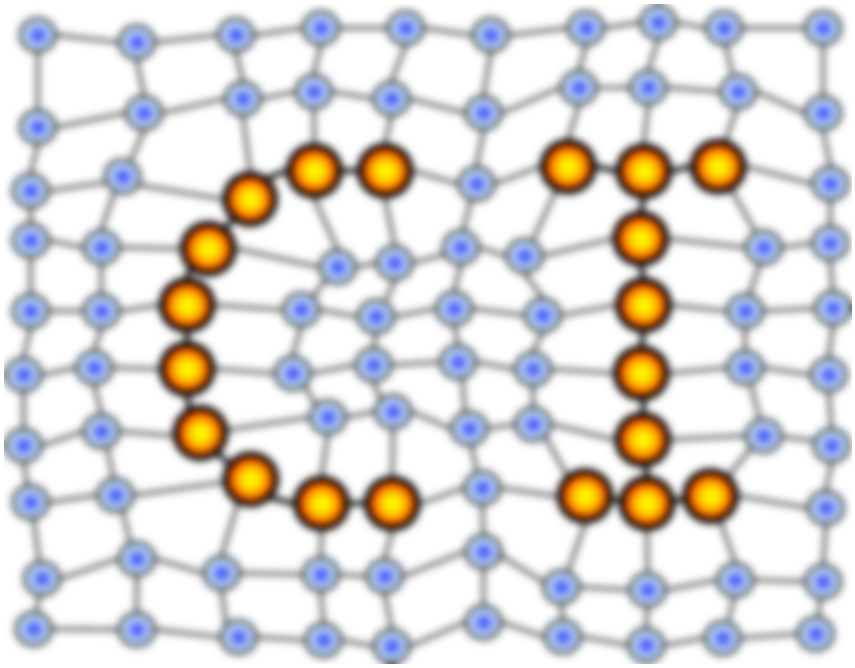
[29] S. Seo, M. Bode, and K. Obermayer. Soft nearest prototype classification. *IEEE Transaction on Neural Networks*, 14:390–398, 2003.

[30] S. Seo and K. Obermayer. Soft learning vector quantization. *Neural Computation*, 15:1589–1604, 2003.

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

MACHINE LEARNING REPORTS

Report 03/2014



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 Bielefeld
Universitätsstrasse 21-23, 33615 Bielefeld, Germany
• http://www.cit-ec.de/tcs/about

▽ Acknowledgments

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