ESANN'2005 proceedings - European Symposium on Artificial Neural Networks
Bruges (Belgium), 27-29 April 2005, d-side publi., ISBN 2-930307-05-6.

The dynamics of Learning Vector Quantization

Michael Biehl[1], Anarta Ghosh[1], and Barbara Hammer[2]

1- Rijksuniversiteit Groningen - Mathematics and Computing Science
P.O. Box 800, NL-9700 AV Groningen - The Netherlands

2- Clausthal University of Technology - Institute of Computer Science
D-98678 Clausthal-Zellerfeld - Germany

Abstract. Winner-Takes-All (WTA) algorithms offer intuitive and powerful learning schemes such as Learning Vector Quantization (LVQ) and variations thereof, most of which are heuristically motivated. In this article we investigate in an exact mathematical way the dynamics of different vector quantization (VQ) schemes including standard LVQ in simple, though relevant settings. We consider the training from high-dimensional data generated according to a mixture of overlapping Gaussians and the case of two prototypes. Simplifying assumptions allow for an exact description of the on-line learning dynamics in terms of coupled differential equations. We compare the typical dynamics of the learning processes and the achievable generalization error.

1 Introduction

Learning vector quantization as proposed by Kohonen has been widely used in a variety of areas due to its flexibility and simplicity of application [1, 9]. A couple of modifications have been developed to achieve a larger flexibility, faster convergence, more flexible metrics, or better adaptation to Bayesian decision boundaries, to name just a few [4, 8, 9, 14]. The motivation of the methods differ. Most learning schemes such as basic LVQ are based on heuristics and their learning behavior is not yet precisely investigated. Others can be derived from a cost function such as GRLVQ [8] or LVQ2.1, the latter being a limit case of a statistical model [12, 13]. Thereby, the connection of the cost functions to the generalization ability is not clear, and only few models explicitely include regularization terms [7]. In addition, some learning rules suffer from divergent behavior and modifications such as the window rule for LVQ2.1 become necessary. Thus, an exact mathematical investigation of typical learning scenarios and their limit behavior would be valuable to judge the performance of the models.

In this work we introduce a theoretical framework in which to analyze and compare different LVQ algorithms. It considers on-line learning from a sequence of uncorrelated, random training data generated according to a model distribution, whereby the training schemes do not make use of the form of this distribution. The dynamics of training is studied along the successful theory of on-line learning [2, 5, 11], considering the limit $N \to \infty$ where N is the data dimensionality. In this limit, the typical system dynamics can be described by ordinary differential equations for a small number of characteristic quantities and the model behavior and generalization ability can be evaluated.

We apply this formalism to example scenarios of WTA-algorithms such that it becomes possible to judge the generalization ability of different parameter choices which correspond to underlying design criteria. The analysis could readily be extended to more general schemes, approaching the ultimate goal to devise novel and efficient LVQ training algorithms with exact mathematical foundation.

2 Winner-Takes-All algorithms

We study situations in which vectors $\boldsymbol{\xi} \in I\!R^N$ belong to one of two possible classes denoted as $\sigma = \pm 1$. We restrict to the case of two prototype vectors $\{w_+, w_-\}$ corresponding to the data classes. In WTA-schemes, the squared Euclidean distances $d_S(\boldsymbol{\xi}) = (\boldsymbol{\xi} - w_S)^2$ are evaluated for $S = \pm 1$ and the vector $\boldsymbol{\xi}$ is assigned to class σ if $d_{+\sigma} < d_{-\sigma}$. We investigate incremental learning schemes in which a sequence of single uncorrelated examples $\{\boldsymbol{\xi}^\mu, \sigma^\mu\}$ is presented to the system. Here, we treat updates of the form

$$\boldsymbol{w}_S^\mu = \boldsymbol{w}_S^{\mu-1} + \Delta \boldsymbol{w}_S^\mu \quad \text{with} \quad \Delta \boldsymbol{w}_S^\mu = \tfrac{\eta}{N}\, \Theta_S^\mu\, g(S, \sigma^\mu)\, \left(\boldsymbol{\xi}^\mu - \boldsymbol{w}_S^{\mu-1}\right) \qquad (1)$$

where $\boldsymbol{w}_S^\mu$ denotes the prototype vectors after presentation of μ examples. The learning rate η is rescaled with the vector dimension N. The Heaviside term $\Theta_S^\mu := \Theta(d_{-S}^\mu - d_S^\mu)$ singles out the prototype $\boldsymbol{w}_S^{\mu-1}$ which is closest to the new input $\boldsymbol{\xi}^\mu$. d_S^μ is the squared distance $(\boldsymbol{\xi}^\mu - \boldsymbol{w}_S^{\mu-1})^2$. In this formulation, only the *winner* $\boldsymbol{w}_S$ can be updated whereas the *looser* $\boldsymbol{w}_{-S}$ remains unchanged. The change of the winner is always along the direction $\pm(\boldsymbol{\xi}^\mu - \boldsymbol{w}_S^{\mu-1})$. The function $g(S, \sigma^\mu)$ further specifies the update rule. We focus on three special cases:

a) **VQ:** $g(S, \sigma) = 1$. Unsupervised vector quantization disregards the actual data label and moves the winner towards the example input. The aim is a good representation of data in the sense of Euclidean distances.

b) **LVQ1:** $g(S, \sigma) = S\sigma = +1\,(\text{resp. } -1)$ for $S = \sigma\,(\text{resp. } S \neq \sigma)$. This extension of competitive learning to labeled data corresponds to Kohonen's original LVQ1. For a *correct winner*, the update is towards $\boldsymbol{\xi}^{\boldsymbol{\mu}}$. A *wrong winner* is moved away from the current input.

c) **LVQ+:** $g(S, \sigma) = \Theta(S\sigma) = +1\,(\text{resp. } 0)$ for $S = \sigma\,(\text{resp. } S \neq \sigma)$. In this scheme the update is non-zero only for a correct winner and, then, always positive, i.e., a prototype $\boldsymbol{w}_S$ can only accumulate updates from its own class $\sigma = S$. We will use the abbreviation LVQ+ for this prescription.

Note that the VQ procedure (a) can be readily formulated as a stochastic gradient descent with respect to the quantization error, see e.g. [6]. While intuitively clear and well motivated, LVQ1 (b) and LVQ+ (c) lack such an interpretation.

3 The model data

To analyze the behavior of these algorithms we assume that data are generated according to a model distribution $P(\boldsymbol{\xi})$. As a simple yet non-trivial situation we consider input data generated according to a binary mixture of Gaussian clusters

$$P(\boldsymbol{\xi}) = \textstyle\sum_{\sigma=\pm 1} p_\sigma P(\boldsymbol{\xi}\,|\,\sigma) \quad \text{with} \quad P(\boldsymbol{\xi}\,|\,\sigma) = \frac{1}{\sqrt{2\pi}^N} \exp\left[-\tfrac{1}{2}\,(\boldsymbol{\xi} - \lambda \boldsymbol{B}_\sigma)^2\right] \qquad (2)$$

where the weights p_σ correspond to the prior class membership probabilities and $p_+ + p_- = 1$. Clusters are centered about $\lambda \boldsymbol{B}_+$ and $\lambda \boldsymbol{B}_-$, respectively. W.l.o.g. we assume that $\boldsymbol{B}_\sigma \cdot \boldsymbol{B}_\tau = \Theta(\sigma\tau)$, i.e. $\boldsymbol{B}_\sigma^2 = 1$ and $\boldsymbol{B}_+ \cdot \boldsymbol{B}_- = 0$, thus the length scale and the location with respect to the origin are fixed.

We assume that the cluster membership σ coincides with the class label of the data. The corresponding classification scheme is not linearly separable because the Gaussians overlap. According to Eq. (2) a vector $\boldsymbol{\xi}$ consists of statistically independent components with unit variance. Denoting the average over $P(\boldsymbol{\xi}\,|\,\sigma)$ by $\langle\cdots\rangle_\sigma$ we have, for instance, $\langle\xi_j\rangle_\sigma = \lambda(\boldsymbol{B}_\sigma)_j$ for a component and correspondingly

$$\left\langle \boldsymbol{\xi}^2 \right\rangle_\sigma = \sum_{j=1}^N \left\langle \xi_j^2 \right\rangle_\sigma = \sum_{j=1}^N 1 + \langle \xi_j \rangle_\sigma^2 = N + \lambda^2.$$

Averages over the full $P(\boldsymbol{\xi})$ will be written as $\langle\cdots\rangle = \sum_{\sigma=\pm1} \langle\cdots\rangle_\sigma$.

Note that in high dimensions, i.e. for large N, the Gaussians overlap significantly. The cluster structure of the data becomes only apparent when projected into the plane spanned by $\{\boldsymbol{B}_+, \boldsymbol{B}_-\}$. However projections in a randomly chosen two-dimensional subspace would overlap completely. In an attempt to learn the classification scheme, the relevant directions $\boldsymbol{B}_\pm \in I\!R^N$ have to be identified to a certain extent. Obviously this task becomes highly non-trivial for large N.

4 The dynamics of learning

The following analysis is along the lines of on-line learning, see e.g. [2, 5, 11]. Here we give a brief summary of the results for LVQ and refer to [3] for details.

The actual configuration of prototypes is characterized by the projections

$$R_{S\sigma}^\mu = \boldsymbol{w}_S^\mu \cdot \boldsymbol{B}_\sigma \quad \text{and} \quad Q_{ST}^\mu = \boldsymbol{w}_S^\mu \cdot \boldsymbol{w}_T^\mu, \quad \text{for} \;\; S,T,\sigma = \pm 1 \tag{3}$$

The self-overlaps Q_{++} and Q_{--} specify the lengths of vectors $\boldsymbol{w}_+, \boldsymbol{w}_-$, whereas the remaining five overlaps correspond to projections, i.e. angles, between $\boldsymbol{w}_+$ and $\boldsymbol{w}_-$ and between the prototypes and the center vectors $\boldsymbol{B}_\pm$.

The algorithm (1) directly implies recursions for the above defined overlaps upon presentation of a novel example:

$$\begin{aligned} N(R_{S\sigma}^\mu - R_{S\sigma}^{\mu-1}) &= \eta\, \Theta_S^\mu\, g(S,\sigma^\mu) \left(y_S^\mu - R_{S\sigma}^{\mu-1} \right) \\ N(Q_{ST}^\mu - Q_{ST}^{\mu-1}) &= \eta\, \Theta_S^\mu\, g(S,\sigma^\mu) \left(x_T^\mu - Q_{ST}^{\mu-1} \right) + \eta\, \Theta_T^\mu\, g(T,\sigma^\mu) \left(x_S^\mu - Q_{ST}^{\mu-1} \right) \\ &\quad + \eta^2\, \Theta_S^\mu\, \Theta_T^\mu\, g(S,\sigma^\mu)\, g(T,\sigma^\mu) + \mathcal{O}(1/N) \end{aligned} \tag{4}$$

Here, the actual input $\boldsymbol{\xi}^\mu$ enters only through the projections

$$x_S^\mu = \boldsymbol{w}_S^{\mu-1} \cdot \boldsymbol{\xi}^\mu \quad \text{and} \quad y_\sigma^\mu = \boldsymbol{B}_\sigma \cdot \boldsymbol{\xi}^\mu, \tag{5}$$

note in this context that $\Theta_S^\mu = \Theta(Q_{-S-S}^{\mu-1} - 2x_{-S}^\mu - Q_{SS}^{\mu-1} + 2x_S^\mu)$.

A major assumption is that all examples in the training sequence are independently drawn from the model distribution and, hence, are uncorrelated with previous data and with $\boldsymbol{w}_\pm^{\mu-1}$. As a consequence, the statistics of the projections (5) are well known for large N. By means of the Central Limit Theorem their joint density becomes a mixture of Gaussians, which is fully specified by the corresponding conditional means and variances:

$$\langle x_S^\mu \rangle_\sigma = \lambda R_{S\sigma}^{\mu-1}, \quad \langle y_\tau^\mu \rangle_\sigma = \lambda \Theta(S\sigma), \quad \langle x_S^\mu x_T^\mu \rangle_\sigma - \langle x_S^\mu \rangle_\sigma \langle x_T^\mu \rangle_\sigma = Q_{ST}^{\mu-1}$$

$$\langle x_S^\mu y_\tau^\mu \rangle_\sigma - \langle x_S^\mu \rangle_\sigma \langle y_\tau^\mu \rangle_\sigma = R_{S\tau}^{\mu-1}, \quad \left\langle y_\rho^\mu y_\tau^\mu \right\rangle_\sigma - \left\langle y_\rho^\mu \right\rangle_\sigma \langle y_\tau^\mu \rangle_\sigma = \Theta(\rho\tau) \tag{6}$$

This observation enables us to perform an average of the recursions w.r.t. the latest example data in terms of Gaussian integrations. Details of the calculation are presented in [3]. On the right hand sides of (4) terms of order $(1/N)$ have been neglected, e.g. using $\left\langle \boldsymbol{\xi}^2 \right\rangle / N = 1 + \lambda^2/N \approx 1$ for large N.

The limit $N \to \infty$ has further simplifying consequences. First, the recursions can be interpreted as ordinary differential equations (ODE) in *continuous training time* $\alpha = \mu/N$. Second, the overlaps $\{R_{S\sigma}, Q_{ST}\}$ as functions of α become *self–averaging* with respect to the random sequence of examples. Fluctuations of these quantities, as for instance observed in computer simulations of the learning process, vanish with increasing N and the description in terms of mean values is sufficient. For a detailed mathematical discussion of this property see [10].

Given initial conditions $\{R_{S\sigma}(0), Q_{ST}(0)\}$, the resulting system of coupled ODE can be integrated numerically. This yields the evolution of overlaps with increasing α in the course of training. The behavior of the system will depend on the characteristics of the data, i.e. λ, the learning rate η, and the actual algorithm as specified by the choice of $g(S, \sigma)$ in Eq. (1). Monte Carlo simulations of the learning process are in excellent agreement with the $N \to \infty$ theory for dimensions as low as $N = 200$, already.

The success of learning can be quantified as the probability of misclassifying novel random data, the *generalization error* $\epsilon_g = \sum_{\sigma=\pm 1} p_\sigma \left\langle \Theta_{-\sigma} \right\rangle_\sigma$. Performing the averages is done along the lines discussed above [3] and yields ϵ_g as a function of the overlaps $\{Q_{ST}, R_{S\sigma}\}$. Hence, we can obtain the learning curve $\epsilon_g(\alpha)$, the typical generalization error achieved from training with αN examples.

5 Results – dynamics

The dynamics of unsupervised VQ has been studied for $p_+ = p_-$ in an earlier publication [6]. Because data labels are disregarded or unavailable, the prototypes could be exchanged with no effect on the achieved quantization error. This permutation symmetry is reflected in a weakly repulsive fixed point (f.p.) of the

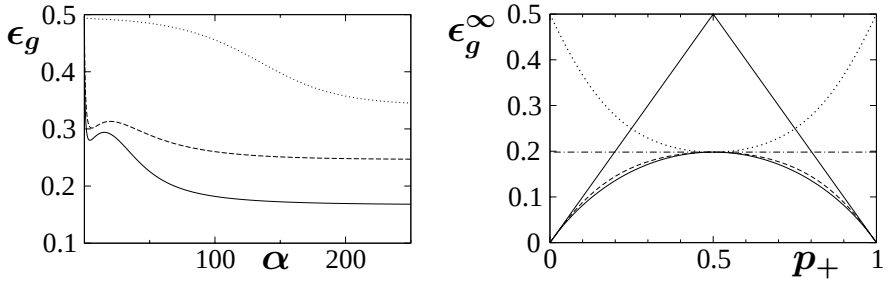

Fig. 1: Left panel: Typical learning curves $\epsilon_g(\alpha)$ of unsupervised VQ (dotted), LVQ+ (dashed) and LVQ1 (solid line) for $\lambda = 1.0$, $\eta = 0.2$, and $p_+ = 0.8$.
Right panel: asymptotic ϵ_g for $\eta \to 0, \eta\alpha \to \infty$ for $\lambda = 1.2$ as a function of the prior weight p_+. The lowest, solid curve corresponds to the optimum ϵ_g^{min} whereas the dashed line represents the typical outcome of LVQ1. The horizontal line is the $p_\pm$–independent result for LVQ+, it can even exceed $\min\{p_+, p_-\}$ (thin solid line). VQ yields an asymptotic ϵ_g as marked by the dotted line.

ODE in which all $R_{S\sigma}$ are equal. Generically, the prototypes remain *unspecialized*, i.e. orthogonal to $(\boldsymbol{B}_+ - \boldsymbol{B}_-)$, up to rather large values of α, depending on the precise initial conditions. Without prior knowledge, of course, $R_{S\sigma}(0) \approx 0$ holds. This feature is discussed in [6] and persists for $p_+ \neq p_-$. While VQ does not aim at good generalization, we can still obtain $\epsilon_g(\alpha)$ from the prototype configuration, see Fig. 1 for an example. The very slow initial decrease relates to the above mentioned delayed specialization.

In LVQ1, data and prototypes are labeled and, hence, specialization is enforced as soon as $\alpha > 0$. The corresponding ϵ_g displays a very fast initial decrease, cf. Fig. 1. The nonmonotonic intermediate behavior of $\epsilon_g(\alpha)$ is particularly pronounced for very different prior weights (e.g. $p_+ > p_-$) and for strongly overlapping clusters (small λ).
Qualitatively, the typical behavior of LVQ+ is similar to that of LVQ1. However, unless $p_+ = p_-$, the achieved $\epsilon_g(\alpha)$ is much larger, cf. Fig. 1. The effect becomes clearer from the discussion of asymptotic configurations in the next section.

6 Results – asymptotic configurations

For stochastic gradient descent procedures like VQ, the expectation value of the associated cost function is minimized in the simultaneous limits of $\eta \to 0$ and many examples such that $\tilde{\alpha} = \eta\alpha \to \infty$. In the absence of a cost function we can still consider the above limit, in which the system of ODE simplifies and can be expressed in the rescaled $\tilde{\alpha}$ after neglecting terms $\propto \eta^2$. A f.p. analysis then yields a well defined asymptotic configuration, see also [6].

For symmetry reasons, the decision boundary with minimal generalization error ϵ_g^{min} is given by the plain orthogonal to $(\boldsymbol{B}_+ - \boldsymbol{B}_-)$ which contains all $\boldsymbol{\xi}$ with $p_+ P(\boldsymbol{\xi}|+1) = p_- P(\boldsymbol{\xi}|-1)$ [4]. The lowest, solid line in Fig. 1 represents $\epsilon_g^{\min}$ for $\lambda = 1.2$ as a function of p_+. For comparison, the trivial classification according to the priors $p_\pm$ yields $\epsilon_g^{triv} = \min\{p_-, p_+\}$ is also included.

In unsupervised VQ, a strong prevalence, e.g. $p_+ \approx 1$, will be accounted for by placing both vectors *inside* the stronger cluster. Obviously this yields a poor classification as indicated by $\epsilon_g^\infty = 1/2$ in the limiting cases $p_+ = 0$ or 1. For equal priors, $p_+ = 1/2$, the aim of representation coincides with good generalization and ϵ_g becomes optimal, indeed.

LVQ1 yields a classification scheme which is very close to being optimal for all values of p_+, cf. Fig. 1. For $p_\sigma > p_{-\sigma}$ the prototype $\boldsymbol{w}_\sigma$ is placed closer to the center of the stronger cluster, hence the prevalence is taken advantage of.

On the contrary, LVQ+ updates each $\boldsymbol{w}_S$ only with data from class S. As a consequence, the asymptotic positions of the $\boldsymbol{w}_\pm$ is always symmetric about the geometric center $(\boldsymbol{B}_+ + \boldsymbol{B}_-)/2$ and ϵ^∞ is independent of the priors $p_\pm$. Thus, LVQ+ is robust w.r.t. a variation of $p_\pm$, i.e. here, it is optimal in the sense of the minmax-criterion $\inf \sup_{p_\pm} \epsilon_g(\alpha)$ [4].

7 Conclusions

We investigated different variants of WTA algorithms in an exact mathematical way by means of the theory of on-line learning. For $N \to \infty$, the system dynamics can be described by few characteristic quantities, and the generalization

ability can be evaluated also for heuristic settings where a global cost function is lacking, like standard LVQ. Interestingly, common features of the learning curves (such as an initial plateau) but also fundamentally different limit solutions can already be observed for slightly different intuitive learning rules, as shown for the variants VQ, LVQ, and LVQ+. The final generalization ability of the algorithms differs in particular for unbalanced class distributions where the goals of minimizing the quantization error, the minmax error, and the generalization error do not coincide. It is quite remarkable that the simple learning rule of LVQ shows near optimum generalization error for all choices of the prior distribution.

It should be mentioned that the simple setting of two prototypes considered above captures important behavior for realistic settings: setting (b) describes the competition at class borders whereas setting (a) takes place within larger classes represented by more than one prototype. Setting (c) investigates an intuitive variant which can be seen as a mixture of VQ and LVQ, i.e. VQ within the classes. The learning rules tackled so far are restricted to iterative winner updates as take place in standard LVQ. It is possible to extend the same technique to more complex update formulas such as LVQ2.1 or the recent proposals [12, 13], which adapt more than one prototype at a time. Preliminary studies along this line by the authors show remarkable differences of the generalization behavior for these versions of LVQ-type algorithms. However, only situations where the model complexity and data complexity coincide have been considered so far. Interesting further computations can tackle the typical case of a different number of modes of the model distribution and clusters of the vector quantizer (e.g. the extreme case $p_+ = 0$ for simple VQ) in which overfitting can occur.

References

[1] *Bibliography on the Self-Organizing Map (SOM) and Learning Vector Quantization (LVQ)*, Neural Networks Research Centre, Helsinki University of Technology, 2002.

[2] M. Biehl and N. Caticha, *The statistical mechanics of on-line learning and generalization.* In M.A. Arbib, *The Handbook of Brain Theory and Neural Networks*, MIT Press, 2003.

[3] M. Biehl, A. Freking, A. Ghosh, and G. Reents, *A theoretical framework for analysing the dynamics of LVQ*, Technical Report 2004-9-02, Mathematics and Computing Science, University Groningen, P.O. Box 800, 9700 AV Groningen, The Netherlands, December 2004, available from `www.cs.rug.nl/~biehl`.

[4] R. Duda, P. Hart, and D. Storck. *Pattern Classification*, Wiley, 2001.

[5] A. Engel and C. van den Broeck, editors. *The Statistical Mechanics of Learning*, Cambridge University Press, 2001.

[6] A. Freking, G. Reents, and M. Biehl, *The dynamics of competitive learning*, Europhysics Letters 38: 73–78, 1996.

[7] B. Hammer, M. Strickert and T. Villmann, *On the generalization capability of GRLVQ networks*, to appear in Neural Processing Letters.

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

[9] T. Kohonen, *Self-organizing maps*, Springer, Berlin, 1995.

[10] G. Reents and R. Urbanczik, *Self-averaging and on-line learning*, Physical Review Letters 80: 5445-5448, 1998.

[11] D. Saad, editor, *Online learning in neural networks*, Cambridge University Press, 1998.

[12] S. Seo, M. Bode, and K. Obermayer, *Soft nearest prototype classification*, IEEE Transations on Neural Networks 14(2): 390-398, 2003.

[13] S. Seo and K. Obermayer, *Soft Learning Vector Quantization.* Neural Computation 15(7): 1589-1604, 2003.

[14] P. Somervuo and T. Kohonen, *Self-organizing maps and learning vector quantization for feature sequences*, Neural Processing Letters 10(2): 151-159, 1999.