Data explosion of recent years has promoted artificial intelligence to an important stature in a variety of applications. Studies estimate that in three recent years, more data are generated than during their preceding human history (Lyman and Varian 2003). The processing of the increasingly available data require advanced techniques in, among others, data mining (Han and Kamber 2005), pattern recognition (Duda et al. 2000) and machine learning (Mitchell 1997). Even with great progresses in these fields, this remains a highly challenging task. A well-funded framework to optimize existing methods and develop novel schemes is necessary to produce faster, more efficient data analysis.

The field of machine learning concerns the design of algorithms to enable an adaptive system to learn from the arrival of information and recognize complex patterns from a subset of examples. A concise history of machine learning can be found in, e.g. (Duda et al. 2000, Mitchell 1997, Ripley 1996). Machine learning can be broadly categorized into supervised learning, unsupervised learning and reinforcement learning. Supervised learning involves an external supervision providing correct responses to given inputs and aim to maximize generalization of novel data. Unsupervised learning, on the other hand, works without supervision with a common goal of discovering underlying structures within the data. Reinforcement learning is concerned with finding suitable actions to maximize long-term rewards, see e.g. (Sutton and Barto 1998). The analysis of pattern involves a number of tasks, including data representation, classification, clustering, density estimation, regression and feature extraction, to name a few.

We concentrate on supervised learning and unsupervised learning for mainly two objectives: representation, which focuses on identifying the characteristic features of the data, and classification, which deals with discriminant analysis to assign unknown data into discrete categories with high accuracy. One of the simplest pattern analysis for these tasks is to compare the unknown data to all known reference patterns on the basis of a similarity measure, known as Nearest Neighbor (NN) methods (Duda et al. 2000). Obviously, the required computation scales immediately with the number of known patterns, therefore it is useful to employ representations of data subsets from each class. Hence, the subsets are quantized into a
number of prototype vectors. In this thesis, we focus on the theoretical aspects of one particular implementation, termed Learning Vector Quantization (LVQ).

### 1.1 Learning Vector Quantization

Learning Vector Quantization constitutes a simple yet powerful family of prototype-based machine learning algorithms, originally proposed by Kohonen in (Kohonen 1997). In its general form, a number of designated labeled prototypes is adapted to input examples to simultaneously provide a proper representation of the data and good classification by means of Nearest Prototype Classification (NPC). Originally, these prototype-based methods are devised for unsupervised learning, termed simply Vector Quantization (VQ). Similar techniques are later developed to incorporate supervised learning problems, which are called Learning Vector Quantization (LVQ).

The use of prototypes in the same feature space of the data is a particularly attractive feature which facilitates a straightforward interpretation of the data, especially to researchers outside the field of machine learning. This is in contrast to, for instance, the black-box approach of feed-forward neural networks or Support Vector Machines. The success of LVQ learning schemes are well documented in the fields of spectral analysis, medical data and gene expression (Bojer et al. 2003, Kuncheva 2004, Schleif et al. 2006, Villmann et al. 2003) among others. An extensive, up-to-date bibliography on LVQ systems are managed in (Neural Networks Research Centre, Helsinki 2002).

LVQ methods are fast, easy to use and well equipped for multi-class classification problems. Despite its apparent simplicity, LVQ is particularly flexible and can be adapted to specific problems with increasing complexity. The adaptation prescription vary greatly and can be modified into various algorithms. Numerous variants have been proposed, including LVQ 1, LVQ 2.1, LVQ 3, Optimized LVQ1 (Kohonen 1990), Generalized LVQ (Sato and Yamada 1995) and Soft LVQ (See and Obermayer 2003). Several algorithms focus on discovering distance metrics with highly discriminative orientations, including Distinction Selective LVQ (Pregenzer et al. 1996), Relevance LVQ (Bojer et al. 2001), Generalized Relevance LVQ (Hammer and Villmann 2002) and Matrix Relevance LVQ (Schneider et al. 2009). The framework of unsupervised prototype-based models can also be derived into popular many-prototype methods such as the popular Self Organising Maps (SOM) and Neural Gas methods (NG) which allow topologically correct representations of the high dimension data. Many modifications allow the LVQ systems to achieve performances comparable to, or better than, many other machine learning techniques, see
for example a comparative study of LVQ and Support Vector Machines (SVM) can be found in (Hammer et al. 2004).

Despite its wide range of applications, the theoretical understanding of LVQ in general remains very limited. In its original form, LVQ 1 was developed by heuristic methods, unlike the statistical flavor of other machine learning methods such as Support Vector Machines and related theories (Cortes and Vapnik 1995, Vapnik 1995). Several approaches are developed in order to provide a mathematical understanding of LVQ, such as constructing algorithms from well-chosen cost functions (Sato and Yamada 1995, Seo and Obermayer 2003) or viewing LVQ systems as Large Margin theorems (Crammer et al. 2002), among others. Nevertheless, a variety of LVQ algorithms remain heuristic and their properties such as convergence speed, stability and overall performance are unknown. A theoretical study would provide great insights which would be highly beneficial to the development of novel, efficient learning schemes.

1.2 Statistical mechanics of learning

A statistical mechanics approach to learning from examples was first proposed to study the generalization ability of Boolean learning networks in (Carnevali and Patarnello 1987) and further elaborated in (Levin et al. 1990, Tishby et al. 1989). Since then, this framework has been applied with success to investigate various machine learning techniques, including perceptrons e.g. (del Giudice et al. 1989, Hansel and Sompolinsky 1990), feed forward neural networks e.g. (Engel and van den Broeck 2001, Seung et al. 1992). Furthermore, the treatment has been extended to prototype based systems, e.g. (Biehl et al. 2007, Engel and van den Broeck 2001, Saad 1999).

Along the lines of statistical mechanics developed by Maxwell, Gibbs and Boltzmann, see e.g. (Huang 1987), macroscopic thermodynamical laws are deduced from microscopic dynamical properties of the system. In parallel with physical system which study properties such as work, heat, energy and pressure, we analyse learning systems in terms of energy, quantization error, generalization error and entropy, among others. The system can be fully described in terms of a set of quantities which characterizes the system, which is referred to as order parameters in the statistical mechanics framework. It is shown later that from the order parameters, it is possible to derive any relevant quantities of the system.

A key ingredient is the high degree of freedom in the system, so-called the thermodynamic limit, where the number of $N$ degrees of freedom tends to infinity. A remarkable property in this limit is that the probability distribution of a relevant quantity becomes sharply peaked around its maximal values. The approach of sta-
Statistical mechanics allows exact description of the typical learning behavior.

Note that this approach is different from the probably almost correct (PAC) notion (Valiant 1984) which are highly applied in many mathematical learning problems. In the general PAC concept, strict bounds on the achievable learning are studied by choosing the worst possible combinations of the target rule and student complexity (Baum and Haussler 1989, Valiant 1984). However PAC is also overly pessimistic because the typical learning behaviors are often substantially better than the worst case scenario. Additionally, most PAC studies are limited to realizable tasks which can be perfectly obtained by the learning system, whereas many real world problems are unrealizable tasks which can only be approximated by the system.

Furthermore learning can be done in two general fashions: on-line learning and off-line learning. In on-line learning, training examples are presented sequentially and the prototypes are adapted to each example. Under a guiding cost function, this is equivalent to a stochastic gradient descent procedures (Bottou 1991). Meanwhile, off-line learning minimizes a properly constructed cost function, for instance by a gradient descent. Obviously, the selection of the cost function is highly relevant to the success of training. The statistical mechanics analyses treat on-line and off-line learning differently in many respects.

In the on-line learning analysis, we study the dynamics of the learning process with respect to learning time. Learning steps after each presentation of examples can be written as recursive relations of the order parameters. By capitalizing on the thermodynamic limit, we can remove the stochasticity from the recursion and describe the learning process as a set of ordinary differential equations of order parameters. This framework was applied with success to, e.g., perceptrons, feedforward neural networks, principal component analysis (Biehl and Caticha 2003, Engel and van den Broeck 2001, Saad 1999, Watkins et al. 1993).

This approach allows rigorous study of various LVQ algorithms with respect to training parameters and initialization of prototypes. We can compare learning curves and convergence speed to identify efficient training schemes. Given long learning times and fixed point analysis, we can also inspect the asymptotic properties of the algorithm.

In the off-line learning analysis, we study at the equilibrium state of the system with a fixed training set. We interpret learning as a stochastic minimization of the cost function under noise presented as the training temperature. A seminal work on this approach was applied to perceptrons, neural networks in (Seung et al. 1992). This approach generated a great interest leading to applications in (Biehl and Caticha 2003, Engel and van den Broeck 2001, Seung et al. 1992, Watkins et al. 1993), among others.

The training temperature acts as a smoothing function over the jagged landscape.
of the actual cost function, allowing for an easier examination of present local and
global minima. However, even with the smoothing function, such analysis tend
to be rather complicated due to the valleys of the cost function and requires more
advanced methods used in statistical mechanics, such as replica method or cavity
method (Mezard et al. 1987), in order to understand the process at finite learning
temperatures. An approximation which has been found to be extremely useful as a
good predictor of these analyses is the so-called annealed approximation (Biehl and
Caticha 2003, Seung et al. 1992, Solla and Levin 1992). Both annealed approxima-
tion and the replica method yield the same exact result in the limit of high training
temperatures.

1.3 Thesis organization

Here we briefly address the topics of the chapters of this thesis. The material is
mainly composed of two parts: the analysis of on-line and off-line learning systems.
The former part spans from Chapters 3 to 4, while the latter part is explained in
Chapters 5 and 6.

Chapter 2 briefly presents the learning model and generic LVQ algorithms to be
studied throughout the thesis. The extents of the input density models and their
relevance in practical situations are discussed.

Chapter 3 explains in detail the analysis of on-line VQ and LVQ systems. We
describe the learning dynamics as ordinary differential equations in terms of the
order parameters and obtain results for both transient and asymptotic properties.
In particular, this chapter focuses on unsupervised learning using VQ and Neural
Gas (NG). We compare the convergence speed and asymptotic properties of Winner-
Takes-All methods to rank-based Neural Gas and demonstrate robustness of NG
under various initial conditions.

Chapter 4 deals with on-line supervised learning using multiple prototype LVQ
algorithms, where we investigate the influence of additional competing prototypes
to fine tune the decision boundary. We analyse and compare various LVQ prescrip-
tions, and the effectiveness of a window-based approach which selects examples
during learning.

Chapter 5 presents the detailed analysis of off-line learning for VQ and LVQ
cost functions along the lines of equilibrium physics. In this chapter, we analyse
unsupervised VQ and NG scenarios based on the high training temperature limit
and the annealed approximation. We demonstrate the phase transition behavior in
the learning curves which depend on the training set size and temperature.

Chapter 6 extends the analysis of high temperature off-line learning to super-

vised methods. We compare the performances of cost-function based LVQ algorithms depending on the size of the training set under the influence of an additional weight decay.

A summary of all chapters is written in chapter 7. The chapter also discusses the relation of on-line and off-line learning and the outlook of further statistical physics analysis of prototype-based learning systems.
Nomenclature

\[ \alpha \] rescaled number of examples \( \alpha = P/N \)

\[ \hat{\alpha} \] number of examples rescaled with temperature \( \hat{\alpha} = \beta \alpha \)

\[ \tilde{\alpha} \] number of examples rescaled with learning rate \( \tilde{\alpha} = \eta \alpha \)

\[ \beta \] inverse training temperature

\[ B \] data cluster centers

\[ \gamma \] weight decay parameter

\[ \delta \] window parameter

\[ \ell_{\sigma} \] separation between cluster centers

\[ \epsilon_g \] generalization error

\[ \epsilon_t \] training error

\[ \eta \] learning rate

\[ \mathcal{D} \] training data set

\[ \lambda \] rank parameter for neural gas

\[ \mu \] example index

\[ \sigma_{\mu} \] cluster generating the \( \mu \)-th example

\[ \Theta \] Heaviside function

\[ v_G \] softness parameter for GLVQ
\( v_{\text{soft}} \)  softness parameter for RSLVQ

\( \mathbf{W} \)  set of prototypes

\( \mathbf{w}_s \)  \( s \)-th prototypes

\( \xi^\mu \)  \( \mu \)-th example

\( b_\sigma \)  projection of examples to cluster centers  \( h_S = \xi \cdot \mathbf{B}_\sigma \)

\( C \)  covariance matrix

\( c_s \)  class of \( s \)-th prototypes

\( d \)  squared Euclidean distance

\( E \)  quantization error

\( F \)  quenched free energy

\( f \)  free energy function

\( H \)  cost function

\( h_S \)  projection of examples to prototypes  \( h_S = \xi \cdot \mathbf{w}_S \)

\( K \)  number of prototypes

\( L \)  likelihood ratio of RSLVQ mixture model

\( M \)  number of clusters

\( N \)  dimensionality of data

\( P \)  number of examples

\( P(\xi) \)  probability density of data

\( Q_{ST} \)  order parameter  \( Q_{S\sigma} = \mathbf{w}_S \cdot \mathbf{w}_T \)

\( r_S \)  rank of \( \mathbf{w}_S \)

\( R_{S\sigma} \)  order parameter  \( R_{S\sigma} = \mathbf{w}_S \cdot \mathbf{B}_\sigma \)

\( T \)  training temperature

\( Y \)  number of classes

\( y \)  class of example

\( Z \)  partition function