Helsinki University of Technology
Dissertations in Computer and Information Science
Espoo 2003 Report D2

LEARNING METRICS AND DISCRIMINATIVE CLUSTERING

Janne Sinkkonen

Dissertation for the degree of Doctor of Philosophy to be presented with due permission of the Department of Computer Science and Engineering for public examination and debate in Auditorium T2 at Helsinki University of Technology (Espoo, Finland) on the 21th of November, 2003, at 12 o'clock noon.

Helsinki University of Technology
Department of Computer Science and Engineering
Laboratory of Computer and Information Science
P.O.Box 5400
FIN-02015 HUT
Finland

Distribution:

Helsinki University of Technology
Laboratory of Computer and Information Science
P.O.Box 5400
FIN-02015 HUT
Finland

Tel. +358-9-451 3272
Fax +358-9-451 3277
http://www.cis.hut.fi/

Available in pdf format at http://lib.hut.fi/Diss/2003/isbn9512267977/

ISBN 951-22-6796-9 (printed version)
ISBN 951-22-6797-7 (electronic version)
ISSN 1459-7020

Otamedia Oy
Espoo 2003

Sinkkonen, J. (2003): **Learning Metrics and Discriminative Clustering.** Doctoral thesis, Helsinki University of Technology, Dissertations in Computer and Information Science, Report D2, Espoo, Finland.

Keywords: clustering, discriminative clustering, exploratory data analysis, feature extraction, information bottleneck, information geometry, learning metrics, mutual information, supervised unsupervised learning.

ABSTRACT

In this work methods have been developed to extract relevant information from large, multivariate data sets in a flexible, nonlinear way. The techniques are applicable especially at the initial, explorative phase of data analysis, in cases where an explicit indicator of relevance is available as part of the data set.

The unsupervised learning methods, popular in data exploration, often rely on a distance measure defined for data items. Selection of the distance measure, part of which is feature selection, is therefore fundamentally important.

The *learning metrics principle* is introduced to complement manual feature selection by enabling automatic modification of a distance measure on the basis of available relevance information. Two applications of the principle are developed. The first emphasizes relevant aspects of the data by directly modifying distances between data items, and is usable, for example, in information visualization with the self-organizing maps. The other method, *discriminative clustering*, finds clusters that are internally homogeneous with respect to the interesting variation of the data. The techniques have been applied to text document analysis, gene expression clustering, and charting the bankruptcy sensitivity of companies.

In the first, more straightforward approach, a new local metric of the data space measures changes in the conditional distribution of the relevance-indicating data by the Fisher information matrix, a local approximation of the Kullback-Leibler distance. Discriminative clustering, on the other hand, directly minimizes a Kullback-Leibler based distortion measure within the clusters, or equivalently maximizes the mutual information between the clusters and the relevance indicator. A finite-data algorithm for discriminative clustering is also presented. It maximizes a partially marginalized posterior probability of the model and is asymptotically equivalent to maximizing mutual information.

CONTENTS

PREFACE

This work has been carried out in the Neural Networks Research Centre, Laboratory of Computer and Information Science, Helsinki University of Technology, in 2000-2003. The work was mainly funded by the graduate school Computational Methods of Information Technology (ComMIT). Significant funding was also received in the form of project money from the Academy of Finland (project numbers are given in the publications).

Special thanks go to Docent Samuel Kaski, my long-time friend, collaborator, and the leader of our research group. Sami's contribution is evident from the publications. He has also been very valuable by encouraging me to attack tedious tasks that otherwise become easily postponed.

I wish to thank my supervisor Professor Erkki Oja for making it, in the first place, possible for me to work in the laboratory. He has also given valuable comments on the manuscript.

I express my sincere gratitude for the comments given by Professor Petri Myllymäki and Doctor Kari Torkkola, the reviewers of the manuscript.

I wish to thank my co-authors Janne Nikkilä and Jaakko Peltonen for interesting and often highly stimulating collaboration. Thanks also belong to other people in the Lab, especially our group and the HTC portion, including Arto, Eika, two Jarkko's, one Kai, another Jaakko, Leo, and Merja. These people have created a friendly and intellectually refreshing atmosphere.

I am of course indebted to many of my friends. Explicitly I have to thank Jouko for regularly questioning me about the status of my thesis for the last seven or eight years.

Finally, thanks to my wife Laura for being present, especially at the final stages of this work.

Ruoholahti, October 27, 2003
Janne Sinkkonen

LIST OF PUBLICATIONS

The thesis consists of an introduction and the following publications:

1. Samuel Kaski and Janne Sinkkonen (2000) Metrics that learn relevance. In *Proceedings of IJCNN-2000, International Joint Conference on Neural Networks*, vol. V, pp. 547–552. IEEE, Piscataway, NJ.

2. Janne Sinkkonen and Samuel Kaski (2000) Clustering by similarity in an auxiliary space. In *Proceedings of IDEAL 2000, Second International Conference on Intelligent Data Engineering and Automated Learning.*, pp. 3–8. Springer, London.

3. Samuel Kaski, Janne Sinkkonen, and Jaakko Peltonen (2001) Bankruptcy analysis with self-organizing maps in learning metrics. *IEEE Transactions on Neural Networks* 12, pp. 936–947.

4. Samuel Kaski and Janne Sinkkonen (2001) A topography-preserving latent variable model with learning metrics. In N. Allinson, H. Yin, L. Allinson, and J. Slack, editors, *Advances in Self-Organizing Maps*, pp. 224–229. Springer, London.

5. Janne Sinkkonen and Samuel Kaski (2002) Clustering based on conditional distributions in an auxiliary space. *Neural Computation* 14, pp. 217–239.

6. Samuel Kaski and Janne Sinkkonen (2002) Principle of learning metrics for exploratory data analysis. *The Journal of VLSI Signal Processing-Systems for Signal, Image, and Video Technology, special issue on Data Mining and Biomedical Applications of Neural Networks*, forthcoming.

7. Janne Sinkkonen, Samuel Kaski, and Janne Nikkilä (2002) Discriminative clustering: optimal contingency tables by learning metrics. In T. Elomaa, H. Mannila, and H. Toivonen, editors, *Proceedings of the ECML'02, 13th European Conference on Machine Learning*, pp. 418–430. Springer, London.

8. Jaakko Peltonen, Janne Sinkkonen and Samuel Kaski (2002) Discriminative clustering of text documents. In L. Wang, J. C. Rajapakse, K. Fukushima, S.-Y. Lee, and X. Yao, editors, *Proceedings of ICONIP'02, 9th International Conference on Neural Information Processing*, vol. 4, pp. 1956-1960. IEEE, Piscataway, NJ.

THE AUTHOR'S CONTRIBUTION

Most of the ideas presented in the publications have been developed as team work, mostly in collaboration with Sami Kaski. Giving individual credit is therefore, to a large part, not meaningful.

Publication 1 introduces the ideas of discriminative clustering and deriving a metric from auxiliary data. It also suggests that a metric generated from a density estimate, implicit in the clustering, would be suitable for further data analysis. The ideas were jointly developed, the mutual information framework perhaps more due to S. Kaski and the application of Fisher metrics in the data space more due to the author. The toy example presented in the paper was a joint effort.

In Publication 2 the discriminative clustering algorithm was generalized for von Mises–Fisher basis functions, applied to two real-life data sets, text documents and financial statements of companies, and found out to perform better than an alternative method based on joint density estimation by a well-known algorithm (MDA2). Here the contributions of the authors are roughly equal and quite inseparable.

In Publication 3 about bankruptcy analysis with self-organizing maps the author had a key role in the development of the principle, including the use of the natural gradient, the performance measures, and the relevance index presented in the paper. The earlier idea of using Fisher metrics with self-organizing maps, presented in Publication 1, was developed here into a practical form and successfully applied.

The idea of constrained discriminative clustering, presented in Publication 4, was jointly developed. S. Kaski contributed the algorithm, as well as most of the simulations.

Publication 5 summarizes the version of discriminative clustering algorithm based on mutual information, improves the optimization algorithm, draws connections to density estimation and related works of other authors, and applies the algorithm to gene expression data. The author contributed to the whole paper, especially to the optimization improvements and drawing connections to density estimation. Also the simulations were to a large part performed by the author.

The main novelty of the otherwise review-like Publication 6 lies in the asymptotic connections presented for the learning metrics principle and the discriminative clustering. The new theoretical results are again a joint effort, but the author's contribution was crucial. The term *discriminative clustering* was coined in this paper.

The application of discriminative clustering to finite data is put onto a firmer theoretical ground in Publication 7 by introducing a generative model with maximum a posteriori cost function, and by proving that it asymptotically becomes the earlier mutual information criterion. The author contributed a large part of the theory and the simulations.

Finally, Publication 8 applies the mutual information version of discriminative clustering to text documents by generalizing the basis functions into a distribution space. Most of the simulations are due to the author, but he also took part into the development of the theory.

The writing of the papers in the sense of producing the actual text has in all cases been a very collaborative effort.

1 INTRODUCTION

This work is about explorative data analysis, where the goal is to reveal and understand the structure of large data sets. Emphasis is on methods that allow automatic direction of exploration by a criterion of relevance that is available in the data itself. Examples of applications include the following:

- A two-dimensional visual mapping of the (continuous) state space of an industrial manufacturing process is wanted for process monitoring purposes. To project future states onto the display, the mapping needs to be defined not only for the data set at hand but for the whole data space. A standard method would be projection onto the first two principal components or the self-organizing map, accompanied by heuristic feature selection.

  In the process monitoring setting, the most useful features of the state are part of its dynamics: They help to predict the future states. Features not correlating with future states are then given less importance, or even ignored. Naturally, future states would not be available during the on-line application phase of the system.

  A method that produces two-dimensional visualizations by weighting features according to their predictive value would then be needed. Publications 3 and 4 of the present work tackle the problem by *learning metrics*.

- A company wishes to cluster its customers, both current and potential future ones, for marketing purposes. Existing customers carry a profile of past purchases, but such a profile is not available for potential new customers. The clustering of new customers must therefore be based on “background information”, such as age, sex, residential address, etc. On the other hand, the buying behavior is operationally important, so the company wants to cluster only on the basis of background information that is strongly correlated to purchasing behavior. *Discriminative clustering*, presented in its current state in Publication 7, is a potential solution.

- Genes are nowadays analyzed by measuring their activation in a massively parallel way in a number of experimental situations. The dimensionality of the feature vectors associated to genes may be large, and we may not have much prior information available about the importance of the dimensions. Known functional classification of the genes may be used to focus the analysis on relevant dimensions of gene activity. The hope is then to discover misclassifications, substructures, and relationships of the known functional classes of genes.

In all these examples, continuous-valued, vectorial *primary* samples become paired to *auxiliary* data. Primary data are clustered or otherwise explored, while auxiliary data serve as an indicator of important variation.

1.1 Data analysis and other close-by fields

The work has two main themes, coined learning metrics and discriminative clustering. Learning metrics is a way to take auxiliary data into account as an indicator of useful variation. It does not constitute an analysis method but is more like a principle from which analysis methods can be derived. Discriminative clustering segments data by interesting variation, with interestingness again defined by auxiliary data. It has a justification of its own as a generative statistical model, but asymptotically it is also an application of learning metrics.

Although learning metrics and discriminative clustering are best categorized into the field of explorative data analysis, it may be informative to relate them to other nearby fields as well.

Exploratory data analysis [99] aims at maximizing understanding of data with a variety of mostly graphical and often relatively simple techniques [71]. The techniques include *clustering*, and mapping into low-dimensional spaces and other visualization methods. *Data mining*, "science of extracting useful information from large data sets or databases" [38] is related, although often associated to rule extraction rather than to finding visual descriptions. The methods of this work fall well to these broad categories, but they are more about visualizing continuous data than finding rules from discrete data.

Machine learning, including *pattern recognition*, has a long history, both in the form of symbolic artificial intelligence (AI) and *neural computation* (see, e.g., [27, 43, 44, 67]. Gradually, at least from the perspective of the author, it has become increasingly difficult to separate neural computation, machine learning and pattern recognition from *statistics* (see, e.g., [10, 39, 79]). The flexible models developed in this work are agnostic enough to be categorized as machine learning techniques, and they certainly have been influenced by neural computation. Some of the concepts used, such as the Fisher information metrics, are familiar from statistics, and discriminative clustering is a statistical model in the sense that it has a likelihood.

Classification refers to the effort of predicting classes of some entities, usually vectors, on the basis of a set of examples. Although the methods developed in this thesis take very similar data, consisting of vectors with class-like labels attached, they are not classification methods. In technical terms, a good classifier models the (Bayes) decision boundary, while the methods introduced in this work model the whole data space. Conditional *density estimation* becomes much closer, to the extent of being formally the cost function of discriminative clustering. The aim of our team has not been to find a good density estimator, however, but to find a clustering algorithm with desirable properties that include a statistical justification.

Discriminative clustering is of course closely related to standard *clustering*. More exactly, it partitions the data space and is therefore a vector quantization method with a new kind of cost function. It has interpretations from the classic *information-theoretic* point of view of coding and compression. For instance, the method has a close relationship with the Information Bottleneck (Section 3.4.2).

The idea of learning metrics, finding a local metric to the data space that would reflect changes of a conditional distribution, is a *feature extraction* principle, and discussed from this viewpoint in Section 2.

The methods fall between *unsupervised* and *supervised* learning: They extract variation by criteria of supervised learning, but model it by methods familiar from unsupervised learning.

1.2 Briefly about the contents

Learning metrics, or the idea of generating local Fisher metrics to data spaces, is reviewed in Section 2. Its application to clustering and self-organizing maps is reviewed in Sections 3 and 2.6.2, respectively.

Discriminative clustering, discussed in Section 3, finds variation that depends on relevance-indicating auxiliary data as separate clusters, and hides other kind of variation inside the clusters. DC has connections to learning metrics (Sect. 3.1.3), mutual information (3.1.1), generative models (3.1.4, 3.2.1), and to the information bottleneck (3.4.2).

Methods from the classic fields of density estimation, discrimination, clustering, and projective explorative methods are reviewed in Section 4. Part of them are presented as general background information, while others are used in the Publications as benchmarks for the new methods. (Although in retrospect this section may seem somewhat artificial and has produced a lot of cross-references, integrating it would have caused other kind of distortions in the flow of the main chapters.)

Section 5 is devoted to concrete data analysis. It summarizes the case studies appearing in the Publications. Conclusions are drawn in Section 6. Some mathematical background for the methods is introduced in Appendix A.

2 LEARNING METRICS

2.1 Feature selection and exploratory data analysis

In unsupervised learning and exploratory data analysis, it is tempting to assume a Euclidean geometry to the data space. This validates the use of many familiar concepts, including global Euclidean distances, orthogonality, parallelism, and projection residuals. One may then search for subspaces and smooth manifolds near which most of the data samples reside. Methods of this flavor include principal component analysis, factor analysis, independent component analysis, multidimensional scaling, Sammon's mappings, principal curves, self-organizing maps, and hierarchical clustering algorithms, i.e., most of the classical unsupervised methods.

With the Euclidean geometry, many explorative methods seem quite elegant and well-defined until one realizes that the geometry is based on the (numerical) *representation* of data. The representation is often rather arbitrary in nature, with the overall level, scale, or even more about the values being arbitrary. As an example, the data may include measures of physical quantities such as weight, temperature, and motion with little hint on whether the weight is best expressed in kilograms or pounds, temperature in kelvins or Celsius degrees, or motion as kinetic energy, impulse, or speed. The relativity of representation makes methods based on the geometry of the data space, including many explorative data analysis methods, equally relative, for the meaning of the geometrical relationships is tied to the more or less arbitrary representation instead of fundamental properties of data that would be clearly derived from the goal of exploration or the phenomenon under analysis.

When no criteria based on deeper properties of data are available or easily usable, the choice of representation, or *feature extraction*, is then left to the analyst. It includes selection, scaling, and transforming the data into a suitable form. If nothing else, at least the variables included in the analysis must be chosen, either before or after their measurement. Note that feature extraction and changing the following analysis procedure itself are trivially complementary. The representation of data can be changed to be suitable for the model, or alternatively, the model to be fitted to the data can be modified to interpret the representation differently. [1]

Feature extraction can be seen as incorporation of prior (expert) information into the analysis process. In probabilistic modeling the complementary approach to incorporating prior information is model selection, which in Bayesian modeling involves the prior. The most convenient way to take prior information into

[1] K-means clustering or, equivalently, vector quantization is typically performed in the Euclidean or another isotropic metric. This is because the isotropic metric, a kind of symmetry of the model, makes the model mathematically easier. In a sense, K-means is, although theoretically flexible, "practically inflexible" which makes the approach of simple model and complex feature selection attractive. The same probably applies to many other models as well.

account seems to be different for different problems. Sometimes prior information is easiest to code into the representation of data, while in other cases it may conveniently be expressed as characteristics of the model family.

Feature selection is highly resource-intensive and may take a considerable part of the overall resources of an analysis project. It requires manual work of participants who have knowledge both on the application and on the technical side of the problem. And as long as feature extraction brings prior, that is, data-independent information to the analysis, its full automatization is impossible.

2.2 Setup of the work: augmented data

Setups exist, however, where partial feature selection is justified by objective criteria. In this work, we have *auxiliary data* (c) paired to the data to be explored, hereby called *primary data* (x), with the assumption that variation of auxiliary data signifies interesting variation of primary data. We are then able to explore only the essential part of primary data variation, which makes the results of the analysis more interesting, accurate and understandable. Implicitly, the intrinsic dimensionality of primary data is reduced by a kind of automatic feature selection.

As an example, the primary data to be explored could be financial statements of companies. If variables indicating later success are used as the auxiliary data, visualization of the financial numbers then illustrates properties of the companies that predict their future success. Other examples are given in Section 1.

Before the concept is formalized below, a philosophical viewpoint to feature extraction is outlined to justify the metric approach.

2.3 Generalization and topology

One of the basic assumptions behind learning metrics is the intimate relationship between inductive reasoning and topology.

Continuous-valued data has a property important for learning: As continuous random variables can take infinitely many values, the probability of encountering exactly the same value twice within a finite sample is zero. Thereby, any practical inference from past to future events on the basis of continuously valued data requires inductive reasoning, generalization from the events that have been seen to new events never seen before. Fortunately, continuous data usually comes with topology on which generalization can be based.

Most data analysis methods assume neighborhood similarity of inferred properties: If two points x_1 and x_2 of the data space are close to each other, then the inferred properties for the points are also assumed to be similar. For example, likelihoods of close-by data in generative models are similar, almost similar data points are mapped to almost the same location in a visualization, and interpretations made by a speech recognition system of close-by word utterances are probably similar. (To a degree, biological systems may also obey this rule.)

Neighborhood similarity is a vague concept. Theoretically, however, the neighborhoods can be taken to the limit, and instead of preserving proximity, we are left with continuity of the inference: The mapping from data to inferred properties should be continuous.[2] This assumes the existence of a natural topology for the inferred properties, of course.

The requirement of continuity could be applied to preprocessing, including feature extraction, by arguing that, given lack of details of further processing, preprocessing should be continuous just in case the original topology is later needed for generalization. Discontinuities of preprocessing would tear the original data space and introduce artificial boundaries that would at later stages be harder or impossible to cross.

2.4 Local variation and metrics

For the continuous-valued, vectorial data $x \in \mathbb{X} \subset \mathbb{R}^n$, we may consider infinitesimal variation between two close-by points, x and $x + dx$. On the basis of the previous Section, we treat feature extraction as a continuous function f from $\mathbb{X}$ to another vector space, say $f(x) \in \mathbb{X}'$. Then feature extraction can be characterized locally: variation of data at a point x in a direction dx is mapped[3] to another small variation dx'.

Local effects of feature extraction. To get forward, the mapping is also assumed to be differentiable, with the derivative or Jacobian matrix $F(x) \equiv \partial f(x)/\partial x$. Then for a small local variation $dx' = F dx$, and for the the magnitude of the variation

$$\|dx'\|^2 = dx^T F(x)^T F(x) dx \; .$$

If the goal is to characterize the effects of local feature selection, we could denote $J(x) \equiv F(x)^T F(x)$, and

$$\|dx'\|^2 = dx^T J(x) dx \; . \tag{1}$$

The matrix field $J(x)$ over all x completely characterizes local effects of feature selection, but it does not require a corresponding transformation $f(x)$ to exist.

Locally defined feature extraction. We can take this approach a bit further, and define local feature extraction in terms of $J(x)$ alone. Assuming $J(x)$ changes smoothly and is non-singular, such a matrix field defines metric relationships for a local neighbourhood of points x, within which $J(x)$ is arbitrarily close to a

[2] Inference often destroys information, which means that as a mapping, it is projective. Although being non-tearing, projective mappings do not preserve topology.

[3] The notation dx' is adopted instead of the obvious df, because we will later get rid of the function f.

constant. Locally the feature extraction problem then becomes the problem of finding a metric.

One can see the local metric as an extra step in a familiar nested sequence of generalizations of the Euclidean metric. Starting from

$$\|\Delta x'\|^2 = \|\Delta x\|^2 = (\Delta x)^T I (\Delta x)$$

with the identity matrix I, we may generalize first to a metric stretching the space in the directions of the coordinate axes:

$$\|\Delta x'\|^2 = (\Delta x)^T W (\Delta x) \ .$$

The identity matrix has been replaced by a diagonal matrix of weights, W. In the next step, general global stretching is allowed; then W may be any positive semidefinite symmetric matrix. The Mahalanobis distance is a special case, with W set to the inverse of a covariance matrix. If we additionally make W local in the feature space by allowing it to depend on x, the local metric (1) results.

The usability of the local metric approach depends on the application. Many analysis methods are based on metric relationships or proximity, and in these cases a *global* metric would suffice, without an explicit new representation or feature extraction for x. A global version of the distances (1) would just be plugged in to the analysis method. Even the local distances (1) can be extended to a global *Riemannian metric* (for text books, see [4, 6, 57, 68]) by describing global distances as the shortest paths over the local distances.

Sometimes an analysis method relies on relatively short distances, and then (1) may be usable as such, as an approximation. We have applied this approach to the self-organizing maps (Section 2.6.2).

In the next Section, a Fisher metric of the form (1) is defined that takes into account the distribution of auxiliary data. Before that, a couple of notes on the generality of the metric approach and on the case of singular $J(x)$ are in order, however.

Local scaling vs. transformation. Feature extraction by local scaling is potentially more general than feature extraction by a smooth (differentiable) transformation. Every smooth transformation f is locally a linear scaling, expressable by its Jacobian matrix. But local scalings exist whose overall effect cannot be described by a dimensionality-preserving smooth transformation. As an example, the geometry of a square on a plane can be converted to the geometry of a hemisphere (a half of a ball) by a local scaling. (This is analogous to feature extraction by local metrics.) Representing the change of the geometry by a transformation on the plane, however, is impossible. If such a transformation existed, we would have distortion-free planar maps of the Earth. Tranformation into a higher dimensionality, of course, is able to produce the desired local geometry seen on the sphere.

Projective feature extraction. If $J(x)$ is singular, distances between non-identical points become zero and $dx^T J(x) dx$ is not a metric even locally. We may then collect points x into equivalence classes, here called x^+. If for all close-by points $x_1 \in x_1^+$ and $x_2 \in x_2^+$ the quadratic form $dx^T J(x) dx$ depends only on the equivalence classes x_1^+ and x_2^+ and not on the exact identity of the points, then Eq. 1 defines a local metric for the equivalence classes x^+. This holds naturally for the Fisher metric of the next Section, so the complication is more theoretical than practical.

2.5 Learning metrics: Fisher metrics in the data space

Next we apply the local approach of last Section to paired data (x, c) by giving c the role of an indicator of important variation in x. Real-world examples of paired-data exploration problems are given in Section 1.

Feature extraction of x will be characterized by the local metric (1), expressed as a smooth matrix field $J(x)$. The matrix $J(x)$ is made to measure the speed of change of the conditional distributions $p(c|x)$ of the random variable C. Then distances of the new metric become long where the distribution $p(c|x)$ changes quickly.

The distributional difference between two points x_1 and x_2 is measured by the Kullback-Leibler divergence

$$d_{KL}^2(x_1, x_2) \equiv \int_c p(c|x_1) \log \frac{p(c|x_1)}{p(c|x_2)} dx \ .$$

The divergence is asymmetric in general, but for close-by points it is asymptotically symmetric (see Section A.1.8 for a proof). If the conditional densities $p(c|x)$ are assumed to be differentiable with respect to x, we may express small divergences by

$$d_{KL}^2(x, x + dx) = dx^T J(x) dx \ , \tag{2}$$

where the elements of the matrix $J(x)$ are defined by

$$\begin{aligned} \{J(x)\}_{ij} &= -\int_c p(c|x) \, \frac{\partial^2}{\partial x_i \partial x_j} \log p(c|x) \; dx \\ &= \int_c p(c|x) \, \frac{\partial}{\partial x_i} \log p(c|x) \, \frac{\partial}{\partial x_j} \log p(c|x) \; dx \ . \end{aligned} \tag{3}$$

The quadratic form follows from a second-order Taylor approximation of the divergence ([66]; a proof is again presented in Section A.1.8). The matrix $J(x)$ is analogous to the Fisher information matrix (see, e.g., [77, 78]), but here small changes of the "model" $p(c|x)$ over the primary data space are considered instead of its variation over a parameter space. The Riemannian metric of the parameter space generated by the Fisher information matrix is called *information metric* by

Amari [4, 6]. Our metric can therefore seen as an application of information metric, with parameters replaced by a covariate. In the Publications, the approach has been coined *learning metrics*.

For a $J(x)$ with a lower dimensionality than that of $\mathbb{X}$, $d(x, x + dx)$ becomes a metric for the equivalence sets consisting of points with zero mutual distance—see the previous Section for the definition of these sets. In practice singularity of $J(x)$ may be problematic, and then the regularized form

$$d(x, x + dx) = dx^T \left(J(x) + \lambda I\right) dx \tag{4}$$

might be more useful, with I being the identity matrix and λ a positive scalar constant.

For theoretical analysis we may just assume $p(c|x)$ known. In concrete data analysis $p(c|x)$ can be estimated from data $\{(x_k, c_k)\}$. Alternatively, one may find an analysis method that at the limit of large data sets behaves like learning metrics but which for finite data sets uses a conventional cost function without explicitly estimating $p(c|x)$. Discriminative clustering, described in Section 3, is an example.

If an explorative method relies on local distances, non-differential (short) distances can be approximated directly by the quadratic form (2). This approach has been successful with the self-organizing maps (Section 2.6.2). The local approximation can be improved by various heuristics [73], and the regularization (4) may cancel part of the adverse effects caused by the inaccuracy of (2).

Although (2) is a local approximation of the Kullback-Leibler divergence, its extension to a global Riemannian metric as minimal path integrals does not produce Kullback-Leibler divergences. Contrary to the Kullback-Leibler divergence, the Riemannian metric would be symmetric, and contrary to the Kullback-Leibler divergence, distances between areas of a fixed $p(c|x)$ are non-zero if they are not connected.

Learning metrics have a connection to mutual information. In Publication 6 it is conjectured that making local, asymptotically small partitions of the $\mathbb{X}$-space spherical in the learning metric maximizes the mutual information between the partitioning and the variable C.

2.6 Realizations

To apply feature extraction of the learning metric type into real data analysis, one must somehow close the gap between the distances based on conditional distributions $p(c|x)$, and a set of samples $\{(x, c)_k\}$. At least three alternatives exist.

2.6.1 Implementation alternatives

The first option is to straightforwardly plug a density estimator $\hat{p}(c|x)$ into the procedure, and derive distances from the estimated probabilities, either analytically or numerically. This approach is conceptually easy and often not difficult to implement, but suffers from the drawback that currently no justification exists for the selection of the estimator. One will then have two non-commensurable cost functions, one for the estimator and one for the analysis method in which the metric is used. We have used explicit density estimaton with self-organizing maps as described in Section 2.6.2 below.

Another approach is to estimate the gradients of $p(c|x)$ in (3) directly without constructing an estimate of the probabilities $p(c|x)$ itself. The mean-shift approach [20] seems to work [82], although there currently exists no justification for the exact form of the gradient estimators.

The third alternative is to totally integrate the metrics into the main analysis method, instead of generating a metric as an intermediate step. The explorative model would then somehow, for example by the shape of clusters, reflect the metric (2) in its own structure. This has essentially been the approach in developing the discriminative clustering, introduced in Section 3.

2.6.2 Self-organizing maps in learning metrics

The self-organizing map (SOM) forms a semicontinuous projection of vectorial or other kind of data onto a usually two-dimensional discrete grid of *prototype vectors* m_j. The prototype vectors are optimized to be similar to data, and simultaneously be similar to neighbourhood vectors on the grid. For more details see Section 4.5.

The stochastic approximation algorithm (Section A.3.1), if used to optimize a SOM, relies on the metric of the data space in two steps. First, a winning prototype m_w for each data sample x is found by minimizing the distance $d(x, m_w)$. Then the winning unit and its neighbours on the grid are updated in the direction of steepest descent of the quantization error $d(x, m_w)$, which in the Euclidean metric co-occurs with the direction of the gradient, $\Delta m_i \propto \partial d^2(x, m_i)/\partial m_i$. The magnitude of the update depends on the neighbourhood relation (i, w) on the grid and the sequential position of the update step in the whole iteration: Units on the grid further away from the winner are updated less, and the updates become smaller towards the end of the stochastic iteration. Many kind of similarity measures have been used with the SOM, but usually $d(\cdot, \cdot)$ is the Euclidean distance.

As described in Publication 3, in learning metrics one can simply find the winning unit in the metric (2), with $J(x)$ most conveniently evaluated at x. This of course assumes that the local approximation of the metric holds. It will hold relatively well for the real winning unit, which is likely to be close to x, but we must additionally assume that errors in computing distances to other units do not come large enough to introduce a false winner. Currently there exists no theoretical quarantee for this assumption. It has, however, worked well enough to make the learning metrics approach beneficial in the experiments of the Publication 3, where we have charted the bankruptcy sensitivity of companies on the basis of financial statements.

The second part of SOM learning, the update of m_w and its neighbours, should be in the direction of maximal decrease of the quantization error $d(x, m_w)$. Here, curiously, the direction and magnitude of the steepest descent are *not* expressed by the gradient but by $J^{-1}(x)\ \partial d^2(x, m_i)/\partial m_i$, the natural gradient. Natural gradient has earlier been succesfully used in training multilayer perceptrons (MLP's) [5].

Another kind of approach to self-organizing maps in learning metrics, related to discriminative clustering (Sect. 3), is presented in Publication 4.

2.7 Related works

In this section the learning metrics idea is related to other conceptually similar approaches. As it in itself is not a data analysis method but rather a new principle for developing algorithms, direct comparisons with related algorithms, especially performance-wise, are difficult.

Discriminative methods in general are related to learning metrics, and this relationship is discussed first (Section 2.7.1). Learning metrics has a connection to the mutual information maximizing paradigm, in which discrete or otherwise reduced representations are found on the basis of a relationship with another variable. These methods, notably the information bottleneck, are mostly discussed in the context of discriminative clustering in Section 3.3, except for the distributional clustering viewpoint which comes close to learning metrics (Section 2.7.2). Finally, a diverse set of metric-modifying approaches and methods are listed in Section 2.7.3.

2.7.1 Discriminative methods

Classic linear discriminants. If the goal of learning metrics is taken to be the exploration of variation correlated to another data, then certainly the classic linear discriminant analysis (LDA) and logistic regression become close in spirit. The basics of these methods are reviewed in Section 4.3.2. They are also introduced in many textbooks ([39, 92], for example).

Connection between learning metrics and the linear discriminative methods can be sought in two directions.

First, LDA and logistic regression produce a low-dimensional representation of data, in terms of the discriminants. If distances between the original data points and their projections were compared, we would observe a projective change of the metric. The justification for such a change of the metric, however, is global discrimination in terms of second moments, not a local criterion as in learning metrics.

Second, logistic regression produces estimates for the conditional distribution $p(c|x)$ of the class variable, and these estimates could be used to generate a learning metric to the primary data space ($\mathbb{X}$-space). With the Bayes rule and the generative interpretation presented in Section 4.3.2, similar metric could be derived from LDA as well. Again, the criteria for the generated metrics are global, and their relationship to the learning metric would probably not be very informative. Note that the metric obtained this way would not necessarily be equivalent to the Euclidean metric obtained by the first method described above, projection of data to the discriminants.

Canonical correlation analysis. A classic non-probabilistic method for two sets of continuous variables is canonical correlation analysis (CCA; Section 4.3.3; see [39, 92] for textbook accounts). It chooses low-dimensional linear representations for both variable sets by maximizing their mutual correlations. Canonical correlation analysis is symmetric with respect to the two continuous variable spaces, but one may see it as feature selection in either of the spaces, with the intention of finding linear features that are maximimally linearly predictive of the other variable. In this sense, CCA is feature selection by discrimination, as is learning metrics.

Interestingly, under the assumption of Gaussian or in general elliptically symmetric distributions, CCA asymptotically amounts to maximization of mutual information between the canonical variates [58]. Potential deeper connections to learning metrics are unexplored, but mutual information maximization is at least related to discriminative clustering, an application of learning metrics (Section 3). Many other, relatively recent works exist on algorithms that maximize empirical mutual information. These will be reviewed in Section 3.3.

Linear projections by maximizing entropy-like criteria. Torkkola and colleagues have suggested linear [98, 96] and nonlinear [97] projections that maximize the class separability on the projection. Analogously to maximizing the conventional mutual information, which is the KL-divergence between $p(c, x)$ and $p(c)p(x)$, class separability in Torkkola's approach is measured by quadratic divergence between the joint distribution $p(c, x)$ and the factored distribution $p(c)p(x)$ (both after projection). Renyi entropy [30, 76] is a related quadratic entropy-like concept. Torkkola's approach is similar to learning metrics in that class separability is maximized, but the criterion is different, i.e., quadratic instead of Shannonian. Further, linear projection is directly sought instead of a metric.

Flexible discriminative methods as a source of metric. The most important relationship between learning metrics and discriminative learning, however, is that the methods producing an explicit estimate $p(c|x)$ for the conditional densities of the auxiliary data can be used as estimators from which the metric (2) is produced. The metric can then be utilized in many conventional unsupervised methods. Of course, joint distribution models can always be marginalized to produce a model for the conditional distribution $p(c|x)$ of the auxiliary variable. It seems that for generating metrics the density estimator should be smoother than what would be optimal in terms of the likelihood (Publication 3).

2.7.2 Distributional clustering and maximization of mutual information

Plenty of research has been done on *co-occurrence* data consisting of discrete values. Most of these works are on the two-variable case. Here we denote the samples (x, y).

Co-occurrence models are reviewed more extensively in Section 3.4 in the context of discriminative clustering. Here they deserve mention because from a viewpoint of one variable, say, X, a conditional distribution of the other variable, $p(y|x)$, is associated to each value x. In the typical case of the discrete values lacking any other attributes and relationships, the data can be seen as *distributional*, consisting of multinomial distributions $p(y|x)$ with some samples y available for each x. Each distribution has its own, unknown parameters θ_x.

This kind of conceptualization is natural for example in the case of text documents, where one may identify x with the documents and y with the vocabulary of the language used in the documents. Then samples from $p(y|x_\theta)$ are words occurring in the documents—here all their mutual relationships, including tendencies to co-occur, are of course ignored.

One may then for example cluster, agglomerate or compute distances for other kind of use on the basis of the conditional distributions. Clustering by distributions is called *distributional clustering* [74]. The distributions are often compared by the Kullback-Leibler divergence, making the problem setting apparently similar to learning metrics.

Differences arise from the continuity of x in the learning metrics setup. The continuity enables one to define a local metric to the $\mathbb{X}$-space, giving the theory a new dimension. On the other hand, because the continuous x almost never gets the same value twice, in practical applications variation of $p(c|x)$ over x needs to be modeled to attain generalization capability. Both the theoretical and the practical parts of co-occurrence data analysis are therefore quite fundamentally affected by the continuity of the covariant variable x.

Distributional clustering and analysis of co-occurrence data in general are also closely related to maximization of mutual information, including the information bottleneck. The connection between learning metrics and mutual information is, however, easier to understand after the introduction of discriminative clustering, and is therefore postponed to Section 3.3.

2.7.3 Related methods based on modified distance measures

Hastie and Tibshirani have proposed a method called DANN, or Discriminant Adaptive Nearest Neighbors [40], that locally performs a regularized LDA-like (see Section 4.3.2) operation for nearest neighbours of the queried data point. This results in a local metric expressable as a function of the within-class and between-class covariance matrices of local data. The metric can then be used in nearest-neighbour classification. The authors show that their metric approximates the Chi-squared distance

$$d^2(x,m) = \sum_c \frac{(p(c|x) - p(c|m))^2}{p(c|m)} \ .$$

When $p(c|x)$ is close to $p(c|m)$, the metric approximates the Kullback-Leibler distance between $p(c|x)$ and $p(c|m)$, and therefore also our metric (2) with the real Fisher matrix $J(x)$.

Domeniconi and colleagues [25, 26] start from the Chi-squared distance and define a more heuristic measure of relevance as a sum of relevances along the coordinate axes. They avoid the construction of potentially high-dimensional matrices, but on the other hand the approach is not invariant to rotations of the data space.

Kontkanen et al. have proposed a setup very similar to ours in that similarities are based on predictions of an “auxiliary” variable [64, 65]. Being based on Bayes networks and a heuristic similarity measure, the method, however, is more suitable for discrete data.

Xing et al. have introduced a way to learn a global flat metric into the data space [101]. They minimize squared distances between given pairs of points under a constraint that prevents distances from collapsing to zero. If values of a (nominal) auxiliary variable are available as in the learning metrics approach, one can minimize within-class distances at the cost of between class distances, which leads into a metric that, like our approach, emphasizes distances where the conditional distributions $p(c|x)$ are different. The metric, of the global quadratic form, is defined for all pairs of points of the data space.

Structures related to metrics have also been generated to discrete data spaces. The *Fisher kernel* [50, 51], very useful in the context of kernel methods, is an inner product defined for all pairs of points of the data space. The Fisher kernel depends on a generative model specified for the data, and it can be defined for continuous data. Tipping has considered its use in clustering but ended up using another, heuristic metric, because the Fisher kernel method proved ineffective [93].

The nearest-neighbour structure of the data set, measured in the Euclidean metric, can be used to generate kernels by a diffusion process. The kernels can then be used for clustering [95] or learning classification from partially labelled data [90].

Unrelated to exploration, metrics of the data space can be modified in the hope of improving predictive methods. For example, Amari has improved the performance of a predictive kernel method by isotropic magnification of the (estimated) class boundary [3].

3 DISCRIMINATIVE CLUSTERING

Roughly speaking, *discriminative clustering (DC)* clusters by interesting variation in data, with interestingness measured by variation of another data. Less interesting variation is hidden inside the clusters. Technically, discriminative clusters are Voronoi regions of vectorial data $x \in \mathbb{X} \subset \mathbb{R}^n$ that are homogeneous by conditional distributions $p(c|x)$ of a random variable C. The metric behind Voronoi regions does not need to be Euclidean, although the Euclidean metric is used in the standard version.

Like other kinds of clustering, DC is a simplification tool useful for exploration and reduction. Contrary to conventional clustering, DC can also refine or coarsify an existing clustering or classification. Discriminative clusters are partitions of the data space, and cluster memberships of new samples are therefore easy to compute. Examples of applications are listed in Section 1.

Two versions of DC have been developed so far. The older, introduced in Publication 5, is defined for probability distributions, and has direct connections to learning metrics and maximization of mutual information. The newer, introduced in Publication 7, is a generative model for finite data sets but asymptotically equivalent to the older DC. The presentation here proceeds in the historical order, by starting from the DC formulated for probability distributions or "infinite data".

3.1 The principle

3.1.1 Between-cluster heterogeneity and mutual information

This section presents the discriminative clustering cost function for probability distributions. Instead of having a paired data set $\{(x,c)_k\}$ available, we assume that the joint distribution $p(c,x)$ of the random variables X and C is completely known, or that we are doing on-line clustering within the stochastic approximation framework (Section 3.2.6). Clustering of finite data sets in considered later (from Section 3.1.4 on).

The goal is to find a partitioning of $\mathbb{X}$ with two properties. First, the partitions should be compact by values of x, and easily interpretable. Second, as we are clustering variation of X that correlates with the relevance indicator C, the partitions should be maximally different by their "contents of C", which under the uncertainty is best summarized by $p(c|x)$. (Intuitively, one would expect large between-cluster variation of C to imply small within-cluster variation, and this indeed turns out to be the case.)

The focus will first be on the second criterion, with the following notation. A partitioning could be interpreted as a random variable V with values v. (Below, v is often used as an index instead of the more redundant notation v_i.) The variable V has a deterministic relation to X to be clustered: given x, v is completely determined. We could therefore also denote by $v(x)$ the function that maps a sample x to the partition v. Denote additionally the sets of x with a constant $v = v(x)$ by $\mathbb{V}(v)$. That is, v is the "partition identity" while $\mathbb{V}(v) \subset \mathbb{X}$ is the actual partition of the $\mathbb{X}$-space, and $\mathbb{V}(v(x))$ is the partition into which x belongs. Although the notation with multiple V's may seem confusing, in practice it is easy to associate a single letter, v, to partitions, and the exact meaning is always clear from the context.

Although many kind of divergences have been proposed, criteria based on the Shannon entropy are an obvious choice for making comparions between distributions. One starting point would be to maximize the heterogeneity of $p(c|v)$ compared to their average by, e.g., the Kullback-Leibler divergence. Averaged over clusters v by weighting with the cluster sizes, such an heterogeneity measure would be

$$E = \sum_v p(v) D_{KL}\left(p(c|v), E_v p(c|v)\right) \ . \tag{5}$$

Because $E_v p(c|v) = \sum_v p(c|v)p(v) = p(c)$, we could write

$$E = \sum_v p(v) D_{KL}(p(c|v), p(c)) = \sum_v p(v) \sum_c p(c|v) \log \frac{p(c|v)}{p(c)} \ .$$

Note that altering the partitioning changes the meaning of v and consequently $p(v)$ and $p(c|v)$, but leaves $p(c)$ intact. Therefore

$$E = \sum_v p(v) \sum_c p(c|v) \log p(c|v) + \text{const.} = -\sum_v p(v) H(C|v) + \text{const.} \tag{6}$$

and E, somewhat counter-intuitively, can be interpreted as a measure of average cluster entropy, which in discriminative clustering then becomes minimized. More interestingly

$$E = \sum_{cv} p(c,v) \log \frac{p(c,v)}{p(c)p(v)} = I(C;V) \ ,$$

that is, E is the mutual information, a measure of statistical dependence, between the partitioning V and the variable C. After a second thought this is not a surprise, for the heterogeneity of $p(c|v)$ makes the prediction of c easy given the partition v.

In addition to mutual information and between-cluster distributional homogeneity (5), DC is also expressible as within-cluster homogeneity.

3.1.2 Within-cluster homogeneity and prototype distributions

Taking a look at the error criteria E of the last section from the point of view of the vectorial variable x reveals that the between-cluster heterogeneity corresponds to a measure of within-cluster homogeneity.

The partitions V are defined in terms of X only—once x is known, the partition membership of x does not depend on C. In probabilistic terms, the partition random variable V and the variable C are *conditionally independent* given X: $p(c,v|x) = p(c|x)p(v|x)$. Substitution of another manifestation of the independence,

$$p(v)p(c|v) = p(c,v) = \int_{\mathbb{X}} p(c,v|x)p(x)\,dx = \int_{\mathbb{X}} p(c|x)p(v|x)p(x)\,dx \;,$$

into (6) gives the intermediate result

$$E = \sum_v \int_{\mathbb{X}} p(v|x) \sum_c p(c|x) \log p(c|v)\, p(x)\, dx + \text{const.}\;, \tag{7}$$

from which

$$E = \text{const.} - \sum_v \int_{\mathbb{V}(v)} D_{KL}\left(p(c|x), \psi_v\right)\, p(x)\, dx \tag{8}$$

is obtained after introducing the notation $\psi_v \equiv p(c|v)$ and observing that $p(v|x) = 1$ if $x \in \mathbb{V}(v)$ and zero otherwise. Note that the symbol ψ_v refers to the whole conditional distribution in partition v over all values of c. (Later the notation $\psi_{vc} = \psi_{v,c}$ will be used to refer to probabilities for single values of c.) From (8) it is evident that E, besides being a measure of between-cluster heterogeneity, also measures within-cluster homogeneity of $p(c|x)$ around the average or *prototype distribution* ψ_v.

For later use, it will be convenient to write the heterogeneity in yet another form. Starting from (7), using the conditional independency of c and v, and denoting the probabilities of the distribution $\psi_v = p(c|v)$ for single c's by ψ_{vc} results in

$$E + \text{const.} = \sum_v \int_{\mathbb{X}} \sum_c \log p(c|v)\, p(c,v,x)\, dx = \sum_c \int_{\mathbb{X}} \log \psi_{vc}\, p(c,v,x)\, dx \;. \tag{9}$$

The sum over v disappears because it can be moved into the integral, where it is redundant because v is completely determined by x.

3.1.3 Connection to learning metrics

Discriminative clustering is, under some assumptions, connected to learning metrics.

For partitions local in $\mathbb{X}$, the within-cluster heterogeneity (8) becomes well approximated by the quadratic distortion of the Fisher metrics (2) spanned by the conditional distributions $p(c|x)$. The distributional prototypes ψ_v then become translated into points $E_{p(x)|x\in\mathbb{V}(v)}x$ of the $\mathbb{X}$-space, around which the distortion is computed.

In other words, DC approximates vector quantization in Fisher metrics, the metrics being defined by the conditional distributions $p(c|x)$. Details are available in Publication 6.

3.1.4 Homogeneity as a likelihood

The theoretical DC criterion of previous sections, formulated for probability distributions, is applicable to real data through the so called empirical distribution. This opens up a connection to generative models (which will later be improved by partial marginalization).

First, however, the distributional prototypes ψ_v need to be turned to parameters. If we write

$$E' \equiv \sum_v p(v) \sum_c p(c|v) \log q(c|v) \ ,$$

it is immediately clear from the properties of entropy [22] that E' is maximized when $q(c|v) = p(c|v) \equiv \psi_v$. Then E' is equal to the heterogeneity E as expressed in (6).

In other words, when discriminative clusters are found by maximizing E with respect to the partitions V, it is equivalent to maximize E' with respect to the partitions *and* ψ_v. The now free parameter ψ_v automatically becomes the correct distributional prototype $p(c|v)$.

A finite data set $\{(x_k, c_k)\}_k$ defines the empirical probability density $\hat{p}(c, x) = \sum_k \delta(x = x_k, c = c_k)$, a sum of "label-wise" delta functions located at the points x_k. For such a distribution, the heterogeneity (9) becomes

$$\sum_k \log \psi_{v(x_k),c_k} \tag{10}$$

which is the log-likelihood of a simple generative model: For a sample (x_k, c_k), the model predicts the conditional density $p(c_k|x_k)$ to be

$$p(c_k|x_k) = p(c_k|v(x_k)) \equiv \psi_{v(x_k),c_k}$$

The predictions are constant within $\mathbb{V}(v(x))$. In the model, the prototypes ψ are treated as free parameters. The model, if fitted to data, can be optimized with respect to ψ and the partitioning defined by V (or $v(x)$).[4]

Discriminative clustering by maximizing between-cluster heterogeneity (5) therefore becomes conditional density estimation, if applied to the empirical distribution generated by a data set. In a sense, the information-theoretic formulation has suggested a generative model.

Conversely, because $\hat{p}(c, x)$ approaches $p(c, x)$ as the number of data points $n(k)$ grows,[5] the conditional density estimator (10) asymptotically has all the

[4] Although the results have been presented for general partitionings V without any restrictions, note that in practice we optimize the model with respect to some *parameters* $\{m_v\}_v$ of the partitionings, not over all possible partitionings. Exhaustive optimization over all possible partitiongs of a continuous space would not be possible or desirable. In practice the partitions have been Voronoi regions.

[5] Again, a rigorous treatment would deal with the convergence of the measures in the probabilistic sense.

nice properties of discriminative clustering: it maximizes the mutual information between partitions and C, minimizes within-cluster heterogeneity of $p(c|x)$ etc.

In summary so far, we have defined the discriminative clustering criterion (5) which is equivalent to mutual information of partitions and the relevance data, and also measures within-cluster homogeneity. The criterion is for probabilities, but asymptotically the likelihood of a simple generative model is equivalent to it. From now on, we accept the generative model as the practical form of discriminative clustering.

3.2 Discriminative clustering models

On the basis of last section, we take the model

$$\hat{p}(c|x) = \psi_{v(x)} \tag{11}$$

of the conditional densities $p(c|x)$ as the starting point of practical discriminative clustering.

The model generates piece-wise constant conditional densities: Distribution of c within a partition v is ψ_v, or $\psi_{vc} = \hat{p}(c|v)$ if values of c are also indexed. The partition of x is denoted by $v(x)$, but otherwise the shape of the partitions is so far undefined. Some options are discussed in Section 3.2.2 below.

Clustering occurs when the model is fitted to the data with respect to the partitions and the distributional parameters ψ. For example, the log-likelihood

$$\ell(V, \psi) \equiv \sum_k \log \psi_{v(x_k),c_k} \tag{12}$$

over the data $\{(x_k, c_k)\}_k$ could be maximized. Although nothing can yet be said of the solution with respect to the unspecified partitioning V, the maximum likelihood solution w.r.t. ψ is simply

$$\hat{\psi}_{vc} = \frac{n_{vc}}{n_v} \ , \tag{13}$$

the relative frequency of nominal values c within partitions v ($n_v = \sum_c n_{cv}$ denotes the amount of data falling to partition v.)

3.2.1 Partial marginalization

Although (11) is formally a conditional density model, in discriminative clustering our interest is in the partitions V and not in the predictions ψ per se. A definitive clustering solution is needed, but we could in many situations do without a point estimate of ψ.

A solution taking into account the uncertainty of ψ may even improve the clustering. The maximum likelihood solution (13) for ψ is based on partition-wise *relative* counts of data. On average, the ratio $\hat{\psi}_{vc}$ is invariant to the overall number of data, but it becomes more variable for small counts. From the maximum-likelihood point of view, partitions with small counts and randomly deviating relative frequencies of classes seem heterogeneous and therefore desirable, although the deviations are of purely random nature arising from the limited sample size.

A remedy to the problem of limited counts is to take the uncertainty into account, and look for an *average* likelihood over the possible values of $\hat{\psi}_{vc}$ behind the observed counts.

The Bayesian framework offers a formal justification and further support for the procedure. Instead of maximizing the likelihood $\ell(V, \psi) = p(C|\psi, V, X)$ of data ($\{(x_k, c_k)\}_k$ denoted for brevity by $D \equiv (X, C)$), we maximize the *marginalized likelihood*

$$\ell_M(V) \equiv p(C|V, X) = \int_\psi p(C|\psi, V, X) p(\psi)\, d\psi$$

with respect to the partitions V only. From the Bayesian point of view, integrating ψ away as they are not needed is very natural.

With a suitable prior for V, the marginalized likelihood is directly proportional to the posterior probability $P(V|D)$ of the model, and then the approach could as well be called *maximum a posteriori* or *MAP* as in the Publication 7. The prior $p(\psi)$ is in the paper chosen to be conjugate Dirichlet with symmetry over partitions, mainly for computational convenience. The marginalized log-likelihood then becomes (more details in Publication 7)

$$\ell_M(V) \propto \sum_{vc} \log \Gamma(n_c^0 + n_{vc}) - \sum_v \log \Gamma(n^0 + n_v)\ , \tag{14}$$

with Γ being the gamma function and n_c^0 and $n^0 = \sum_c n_c^0$ prior parameters. Marginalization has improved results in the experiments of Publication 7.

The results so far are theoretically applicable to any kind of partitioning V, but on the other hand without any hint of how optimization would occur. Marginalization does not change this fact. We will next take a brief look at estimation with different partitions and algorithms.

3.2.2 Parameterizing the partitions

Discriminative clustering seeks partitions of $x \in \mathbb{X} \subset \mathbb{R}^n$ mutually heterogeneous by their distributional content $p(c|x)$ or, equivalently, partitions internally homogeneous by $p(c|x)$. In last sections various criteria were developed for the distributional content, but it is clear that the shape of the partitions $\mathbb{V}$ in a continuous space must be somehow restricted.

Our approach is to parameterize the partitions as *Voronoi regions*: for the distances $d(x, m_v)$ from $x \in \mathbb{V}_v$ to *prototypes* m_v, it holds $d(x, m_v) \leq d(x, m_{v'})$ for all $v' \neq v$. The partitions are mutually exclusive except for the ambiguously partitioned borders $d(x, m_{v'}) = d(x, m_v)$ that have no practical significance.[6] Voronoi partitions are familiar from K-means clustering and vector quantization.

Euclidean partitions. The original DC partitions, since Publication 1, were Voronoi regions of the Euclidean space. A very simple optimization algorithm follows, detailed in Section 3.2.3 and in many Publications, including 5.

Other kind of partitions. For text document classification under the so called vector space model, Voronoi partitions of a hypersphere with the inner product metric have been useful (Publications 2, 5, and 8). Discriminative clustering of the distributional space ($0 \leq x \leq 1$, $\sum_i x_i = 1$) was introduced in Publication 8.

3.2.3 Estimation

Estimation means finding the partitions V that would maximize the marginalized likelihood $\ell_M(V)$ of (14), or finding both the partitions and the distributional prototypes ψ to maximize the unmarginalized $\ell(V, \psi)$ in (12). For Voronoi regions, the partitioning is defined by the prototype vectors $\{m_v\}_v$.

Because the likelihoods (12) and (14) are functions of sample counts of partitions, they are non-continuous with respect to the prototype vectors $\{m_v\}_v$—when an m_v changes, Voronoi regions shift until a sample suddenly jumps from one region to another. Gradients of the likelihoods with respect to the parameters $\{m_v\}_v$ then become zero or non-existent, and many conventional optimization algorithms are not applicable.

Two kinds of remedy exist to the problem. First, algorithms not relying on the gradients, such as simulated annealing, could be used [56, 60]. The other approach is to modify the cost function by smoothing the partitions, as discussed next.

3.2.4 Estimation by smoothing

The idea is to make partition memberships gradual and continuous with respect to the partition parameters. The advantage is that (modified) likelihood functions then have gradients making their optimization much easier.

Partition memberships of data x have been interpreted as values v of the random variable V. The interpretation has been somewhat artificial, for the relationship is deterministic for hard partitions and denotable simply as a function $v(x)$. The conditional probabilities $p(v|x)$ are then either zero or one.

Soft partitions utilize V better. We may denote degrees of memberships of x by $y_v(x)$, with $\sum_v y_v(x) = 1$. It is then technically convenient to interpret the partial memberships as (artificial) uncertainty, that is, we set $p(v|x) \equiv y_v(x)$.

[6] We assume that both the Lebesque measure and the measure $P(x)$ of the borders is zero.

The random variables V and C are still conditionally independent given X (see Section 3.1.2). Most of the theory of discriminative clustering holds also for the soft partitions, including the equivalence of between-partition heterogeneity (5), internal homogeneity (8), and mutual information $I(C;V)$.

The soft versions of the likelihoods (12) and (14) have a gradient with respect to partition parameters $\{m_v\}_v$. Optimization can then be performed by stochastic approximation (early Publications), conjugate gradients (Publication 7), or other gradient-based algorithms.

The results would, of course, be not exactly optimal for hard partitions. An unexplored compromise would be to start with very soft partitions, and gradually shift towards hard partitions during the optimization.

3.2.5 Improving results with regularization and good initialization

In experiments (Publication 7, [56, 60]), modified or 'regularized' versions of the marginalized discriminative clustering cost (14) have worked better than the original. They seem to help with the optimization algorithms that otherwise may produce clusters with little or no data.

In a modified version, coined entropy regularization, extra weight has been put onto the second term:

$$\ell'_M(V) \propto \sum_{vc} \log \Gamma(n^0_c + n_{vc}) - (1+\lambda) \sum_v \log \Gamma(n^0 + n_v) \ ,$$

with $\lambda > 0$, usually $\lambda \approx 0.2$. Weighting the latter term favors partitions of equal sizes in terms of data $\{n_v\}_v$, for $\log \Gamma(\cdot)$ is a convex function and $\sum_v n_v$ is fixed. In addition, for increasing $\{n_v\}_v$ the term approaches Shannon entropy, making it in a sense a natural measure of evenness.

Another regularization method introduces an additional term to the likelihoods that can be interpreted as a model for the density $p(x)$. No performance difference between the two methods has been found [60].

The same set of simulations emphasized the importance of good initialization with the current optimization methods. Initialization by K-means seems to work well.

3.2.6 A simple on-line algorithm

A particularly simple algorithm follows if stochastic approximation (Sect. A.3.1) is applied to optimizing the likelihood (12) with softened Euclidean partitions (Publication 5). The update rule for model vectors $\{m_v\}_v$ and the random sample (x, c) is essentially

$$\Delta m_v \propto (m_v - x) \log \frac{\psi_{vc}}{\psi_{v'c}} \ , \tag{15}$$

with v and v' being the two neighborhood partitions of x. Similarity to the LVQ updating rule (Section 4.4) is striking. Details of the stochastic optimization are perhaps best presented in Publication 5, with convergence proved in a technical report [55]. No equally simple algorithm has been found for the marginalized likelihood (14), but on the other hand the results with (14) optimized by a conjugate gradient algorithm are better than with (15).

3.2.7 SOM-like constrained discriminative clustering

The current DC parameterization can be modified by replacing the Voronoi parameters $\{m_v\}_v$ by something different or by constraining the current parameterization.

An example of the latter is presented in Publication 4, the rationale being to replace the learning metrics SOM scheme of Section 2.6.2 with a more justified, integrated approach. The prototypes $\{m_v\}_v$ were themselves parameterized as linear (convex) combinations of second-level prototypes $\{m'_v\}_v$. The mapping was such that nearby $\{m_v\}_v$, as arranged to a SOM-like rectangular grid, were to a large degree influenced by the same second-level parameters, the aim being to get an ordered map with similarity of close-by prototypes.

The approach showed some success in experiments with toy data but needs further development.

3.3 Related approaches

This section discusses methods that are either alternative to the kind of discriminative clustering presented above, or near enough by their spirit, theoretical background, or applicability to be confusing or interesting.

A section of its own has been devoted to the information bottleneck and analysis of co-occurrence data in general (Section 3.4). Elsewhere in this work, the asymptotic connection between DC and learning metrics is covered in Section 3.1.3, and the context of learning metrics, thereby somewhat relevant to DC, is outlined in Section 2.7. See also Section 4.2.2 for introduction of K-means, and 3.2.6 and 4.4 on the similarity of the stochastic DC update rule to LVQ.

Section 4 deals with research directions associated to the present work, including learning metrics.

3.3.1 Discriminative clustering by explicit density estimation

Both discriminative clustering and learning metrics aim at modeling variation of data correlated to some other data. DC does that directly in the context of clustering, while learning metric is an intermediate step usable together with traditional unsupervised methods.

Learning metrics can also be applied to clustering very much like it was applied to SOM's in Section 2.6.2: K-means is computed in the approximate Fisher metrics (2), based on an external conditional density estimator $\hat{p}(c|x)$, and evaluated either at the data points or at the prototype vectors $\{m_v\}$ [82]. The update rule is then similar to the LM-SOM update rule (Publication 3) but without a neighborhood.

3.3.2 Concatenation

A trivial although theoretically unjustified method for clustering mixed-type co-occurrence data $\{(x_k, c_k)\}_k$ would be to concatenate C, encoded in the one-of-C fashion, to X, and then cluster the resulting vectors by classic methods.

Concatenation would deviate from DC in several ways. The model would be for the joint distribution $p(c, x)$ instead of DC's $p(c|x)$, which is potentially suboptimal as explained in Section 3.3.5. The clusters would by default be for (X, C), and may be difficult to translate to clusters in X. No obvious principle exists for choosing the relative scales of C and X.

3.3.3 Conditional density estimation

Discriminative clustering in the sense of maximizing the likelihood (12) *is* a special case of conditional density estimation. The DC model is, additionally, easily interpretable as clusters and has its known asymptotic connections to mutual information (Section 3.1.1) and learning metrics (Sections 3.1.3, 2).

The likelihood (14) goes a bit further and marginalizes the conditional density estimates away, leaving only the cluster structure. This emphasizes the difference between DC and density estimation: although the model is similar, DC clusters while estimation predicts.

The structure of some mixture models of conditional density estimates might be suitable for discriminative clustering. For Euclidean partitions, the simplest candidate would probably be

$$p(c|x) = \sum_v \psi_{vc} \frac{G(x; m_v, \sigma)}{\sum_{v'} G(x; m'_v, \sigma)} \equiv \sum_v p(c, v|x) \ , \tag{16}$$

where ψ_{vc} are distributional parameters resembling the "prototypes" of discriminative clustering, and G is an isotropic Gaussian kernel parameterized by the locations m_v and the dispersion σ. The model could be conceptualized as a very simple *mixture of experts* [52] with Gaussian gating.

When $\sigma \to 0$, the "partitions" $G_v / \sum_{v'} G_{v'}$ become mutually exclusive and the model translates to discriminative clustering of type (12). From the viewpoint of hard partitioning, therefore, the mixture model would just be another smoothing technique for estimating the partitions. Its advantages are so far unexplored. Especially, the technique has not been properly compared to that presented in Section 3.2.4 and used in the Publications.

Note that the conditional independence assumption of the mixture models (16)–(18) is the more conventional $p(c, x|v) = p(c|v)p(x|v)$ instead of DC's $p(c, v|x) = p(c|x)p(v|x)$ (on DC, see Section 3.1.2).

Bayesian justification for modeling conditional density instead of the joint density is shortly discussed in Section 4.1.1.

3.3.4 Classification

Here classification means the effort of *forced-choice* prediction of c given x and some training data $\{(x_k, c_k)\}_k$. Costs and benefits are then a function of the frequencies of misclassifications only, and no probability estimate $\hat{p}(c|x)$ is necessarily defined by the model. Classifier effectively partitions the $\mathbb{X}$ into a finite number of regions of constant answers. The theoretical optimal decision boundary is called Bayes boundary.[7]

Asymptotically, with increasing data, the resources of a good classifier go into representing the decision boundary well. This is in contrast to conditional density estimation (see 3.3.3 above), where the increasing accuracy offered by extra data is divided evenly over $p(x)$.

Discriminative clustering partitions the $\mathbb{X}$-space like a classifier does, but it is not tied to finding and representing class boundaries. DC resembles more conditional density estimation than clustering in that it models $p(c|x)$ over the whole $p(x)$. If DC were used for classification, by fitting $n(c)$ clusters to data $\{(x_k, c_k)\}_k$, the clusters V would predict C well in the statistical sense of mutual information $I(X, V)$, but there would be no guarantees of a one-to-one correspondence between the clusters and the classes c. If variation of $p(c|x)$ is large enough somewhere in $\mathbb{X}$, the area would get finely clustered regardless of the number of contributing classes (values of C).

3.3.5 Clustering the joint distribution

Within the mixture approach (Sect. 4.1.2), the obvious alternative to performing discriminative clustering by modeling the conditional distribution $p(c|x)$ (Sect. 3.3.3) would be to model the full joint distribution

$$p(c, x) \equiv \sum_v p(c, x, v) \tag{17}$$

instead. This kind of models are usually optimizable by the EM algorithm (Section A.3.2), and hard partitioning of $\mathbb{X}$ can be derived by mapping each x to the partition v with maximal $p(v|x)$ (obtainable by the Bayes rule).

An obvious choice for the Euclidean case would be

$$p(c, x) = \sum_v \pi_v \psi_{vc} G(x; m_v, \sigma) \equiv \sum_v p(c, x, v) \; . \tag{18}$$

[7] In the context of classification, methods that directly model the class boundaries or the conditional class probabilities are called *discriminative*, in contrast to informative methods that model each of the class densities separately and derive the conditional class probabilities via the Bayes rule [81]. This kind of discriminativity is the origin for the name *discriminative clustering*.

The model, called MDA2 by its authors [42, 41], is equivalent to (16) but without the implicit marginalization and with the cluster-wise coefficients $\{\pi_v\}_v$ added.

Intuitively, however, modeling the whole joint distribution *generatively* would waste resources on the details of $p(x)$ at the cost of the margin $p(c|x)$. The result would then not optimally discriminate between areas of different distributions $p(c|x)$, but rather be a compromise between modeling $p(x)$ and *discriminating*, or modeling $p(c|x)$. The prevailing folk wisdom is reported to recommend the discriminative approach for discriminative problems, except perhaps for very small data sets [28, 69]. In the experiments reported in the Publications 2 and 5, DC generally outperforms MDA2-based discriminative clustering, which supports the conclusion that at least for large data sets the discriminative approach works well.

3.4 Related work: co-occurrence analysis and information bottleneck

Co-occurrence data consists of paired samples $\{(x_k, y_k)\}_k$ of two *discrete* random variables X and Y. Interest lies in modeling the dependency of X and Y. From the viewpoint of vectorial data and Section 2.3, the situation here is in some sense harder but interesting in that no predefined topology is present to justify generalization over the categorical variables.

Often the samples $\{(x_k, y_k)\}_k$ are independent over k, and then a convenient representation is the *contingency table*, consisting of the frequencies of all possible co-occurrences of the margin variables X and Y.

Most popular goals of analysis include finding components of the joint distribution or clusters of one of the marginals. Clustering a margin of a contingency table is almost synonymous to *distributional clustering* [74] of discrete distributions.

Analysis of contingency tables is a traditional problem in statistics. Perhaps the most prominent modern application area of co-occurrences is *text document analysis*. If one discards word order and assumes the (conditional) independency of words, pieces of text can be treated as "bags of words", analogously to bags of marbles with various colors (see Section 5.3 for more). A contingency table would then describe a document collection with the columns corresponding to words and the rows to the documents.

Discriminative clustering and learning metrics deal with paired data where the member to be clustered is continuous, and are therefore not directly related to the analysis of co-occurrence data. Three indirect connections, however, make this discussion worthwhile. First and foremost, the justification of DC is very close to the *Information Bottleneck (IB)* framework used in co-occurrence analysis. Second, if DC is applied to text document analysis, usually in connection with the *bag of the words* and *vector-space models* (Section 5.3), the setup becomes confusingly similar to the prime applications of co-occurrence methods. Third, DC can be seen as the optimization of the margin of a contingency table. Before returning to these topics, we will for completeness take a look at classical and generative contingency table models.

3.4.1 Generative co-occurrence models

The analysis of contingency tables and co-occurrence data is a large and relatively old field.

One of the simplest forms of analysis are tests for independence of the row and column variables. The very well known, classic method relies on a large-sample approximation leading to a statistic following the chi-square distribution under null hypothesis (see for example [78]). Also an exact test is available, following from the assumption of fixed marginals and the hypergeometric distribution [78].

Classic models for analyzing the structure of the dependencies are well reviewed in Goodman's 1985 paper [37]. These are of the general form

$$p_{ij} \propto \sum_k \exp \rho_k \alpha_{ki} \beta_{kj} \ ,$$

with some orthogonality and normalization constraints (association model; for details see [37]), or of the form

$$p_{ij} \propto 1 + \sum_k \rho_k \alpha_{ki} \beta_{kj} \ ,$$

which can be seen as a linearized version of the multiplicative model for small values of ρ. The latter model class includes correspondence analysis. Hofmann has relatively recently—compared to the classic models—proposed the model

$$p_{ij} = \sum_k \rho_k \alpha_{kl} \beta_{kj} \ ,$$

which requires extra (entropic) regularization of ρ_k to be practical [47, 48]. The model, if parameterized asymmetrically, is called probabilistic latent semantic indexing (pLSI; [46]). The asymmetric clustering model (ACM), also by Hofmann [47, 48], is genuinely asymmetric and close to some manifestations of the information bottleneck principle to be introduced in Section 3.4.2 below.

Although the models above are generative for the training data, none of them includes a generative process for the margins. This poses at least an aesthetic problem for many applications where it is necessary to relate new margin values, like new text documents, into the existing model.

Mixture of unigrams [70] is generative for a margin, as is the recent Latent Dirichlet Allocation (LDA; [12, 13]), perhaps more properly called multinomial PCA [17]. Especially the latter model is interesting, but out of the scope of this thesis. Other, traditional non-generative models for text documents are shortly discussed in Section 5.3 and in Publication 8.

3.4.2 Information bottleneck

The Information Bottleneck (IB) [94] is an information-theoretic principle used to build representations of random variables. An important practical application is in the clustering of co-occurrence data. We will here first outline the idea, then relate it to DC.

Given a joint distribution $p(x, c)$ of two discrete variables X and C, IB defines a *representation* of X that is maximally informative of C. The representation is defined as a discrete random variable V taking values v, and encoded into the conditional distribution $p(v|x)$. The variable V can be said to represent X for two reasons. First, it depends only on X and not other variables (C) in the setup, and second, it is artificial in that its relationship to X is completely determined by its parameterization. Clearly $p(v|x)$ defines a soft partitioning of the $\mathbb{X}$-space in the sense of Section 3.2.4. Consequently, the partitioning V of discriminative clustering (Section 3.1.1) is a "representation of X".

The complexity of V as a representation of X can in a sense be restricted by limiting the amount of information V carries of X, i.e., limiting the mutual information $I(X;V)$. On the other hand, V is made as informative of C as possible by maximizing $I(C;V)$. Combined, these requirements lead to the variational criterion

$$I(X;V) - \beta I(C;V) \ ,$$

which is to be minimized with respect to the representation, that is, $p(v|x)$. The compromise between complexity and informativeness is chosen by the parameter β and gives the information bottleneck its name: V works as an information bottleneck between X and C.

Finding an optimal $p(v|x)$ for the IB becomes a problem of rate distortion theory [22], if one identifies X with the signal to be sent over a channel, V with the codebook, and $E_{X,V} d(x, v) \equiv -I(C;V)$ with the average distortion. Then we would seek a codebook V with maximal rate $I(X;V)$ and minimal average distortion $E_{X,V} d(x, v)$. The solution is

$$p(v|x) = \frac{p(v)e^{-\beta d(x,v)}}{\sum_{v'} p(v')e^{-\beta d(x,v')}} \ , \tag{19}$$

with $d(x, v)$ being the Kullback-Leibler divergence $D_{KL}(p(c|x), p(c|v))$. An iterative algorithm, resembling the Blahut-Arimoto algorithm of the rate distortion theory, can be used to find a concrete solution from the self-referential (19).

IB and distributional clustering. Distributional (*not* discriminative) clustering (Section 2.7.2; [74]) refers to clustering of co-occurrence data (x, c) by comparing the conditional distributions $p(c|x)$. From this viewpoint, IB performs distributional clustering with soft clusters V, the cluster memberships being $p(v|x)$. From (19) we see that each cluster membership is a monotonically decreasing function of the KL-divergence from the corresponding prototype. IB gives a justification for the use of KL-divergence as the distortion measure of distributional clustering (as also noted in, e.g., [85, 94]).

Estimation. Like the information-theoretic formulation of DC (Section 3.1.1), the formulation of IB presented above is for a known joint distribution $p(c, x)$. Because IB as such is not a generative model, the common likelihood-based estimation criteria, such as maximum likelihood, are not applicable. One can then rely on the empirical distribution $\hat{p}(c, x)$, computed as the normalized frequency counts. The IB optimization criteria with the empirical distribution does, however, become maximum likelihood in some special cases, including that of $\beta \to \infty$ [89].

DC vs. IB. Discriminative clustering is independently developed, but its original, information-theoretic formulation can be seen as the information bottleneck generalized for a continuous X, taken to the limit $\beta \to \infty$, and with the representation $p(v|x)$ parameterized to enforce the resulting clusters to the Voronoi shape.

Although originally so formulated, IB is in principle not limited to discrete random variables. For a continuous $x \in \mathbb{X} \subset \mathbb{R}^n$, when $\beta \to \infty$ the partitions (19) would become Voronoi regions in the space consisting of the *distributions* of C, and the Voronoi regions would minimize $D_{KL}(p(c|x), p(c|v))$. We could map the distributional partitions, indexed by v, into sets $\mathbb{V}_v \subset \mathbb{X}$ by defining: $x \in \mathbb{V}_v$ if $p(c|x)$ belongs to the Voronoi region with prototype $p(c|v)$. Then we have partitions of $\mathbb{X}$ as in DC, but they may be disconnected.

Compared to DC, IB has an extra term $I(X; V)$ in its cost function. From this point of view, DC is a continuous information bottleneck taken to the Voronoi limit $\beta \to \infty$ (which makes the extra term essentially disappear).

DC partitions are parameterized as Voronoi regions of the $\mathbb{X}$-space. Besides being a necessity to get the partitions of the the continuous space manageable, the parameterization keeps the clusters connected.

For both DC and IB, the gap between an information-theoretic formulation and a practical clustering criterion is bridged with the empirical distribution $\hat{p}(c, x)$. For hard clusters, a generative interpretation is available and opens a route to Bayesian estimation of the clusters (Section 3.2.1).

Applications and extensions of IB. The IB principle has been applied to agglomerative [87] and non-agglomerative [85] form of marginal clustering of co-occurrence data and, besides text documents, for example to galaxy spectra [86]. Approaches to two-margin clustering have been presented [88, 31], and multivariate extensions exist [31, 19]. IB has also been used to define kernels into a set of continuous data [95]. A recent development called Sufficient Dimensionality Reduction is also related to the IB [35].

3.4.3 Discriminative clustering and Bayesian contingency tables

Because discriminative clustering maximizes mutual information between its partitions V and the guiding random variable C, it is expected that other dependency measures might be maximized as well.

It is shown in Publication 7 that the likelihood of the marginalized generative DC (Section 3.2.1; $\ell_M(V)$) is equivalent to a Bayesian likelihood ratio, the probability of data under a full Dirichlet prior for the table vs. a prior decomposable into Dirichlet margins [36]. Thus, while the information-theoretic formulation of DC maximizes mutual information, the generative version of DC finds a margin V for the contingency table that maximizes the Bayesian measure of dependency. Asymptotically, for increasing number of data, the dependency measures coincide.

4 Other paradigms

A few methods from the classic fields of density estimation, discrimination, clustering, and projective explorative methods are reviewed in this section. Part of them are useful as a general background material, while others are used in the Publications as benchmarks for the new methods. Methods directly related to learning metrics or discriminative clustering are not discussed here but in Sections 2.7, 3.3 and 3.4.

4.1 Density estimation

Density estimation refers to the estimation of the density $p(X)$ given data X. The data usually comes in the form of independent observations: $X = \{x_k\}_k$, and then we are actually interested in estimating the density of single observations $p(x)$ [79]. As implied by the word "density", at least part of the structure inside samples X is usually continuous (vectorial).

In a sense, generative modeling (Section A.2) is always density estimation, for the models generate a density $p(x)$ to the data space that in some sense fit to the data X well. The set of potential density estimation methods is therefore very wide.

Here we limit our attention to two aspects of density estimation. First, the justification for conditional density estimation, important for explicit generation of learning metrics (Publication 3, Section 2.6.2) as well as for generative discriminative clustering (Section 3.2.1) is viewed from the perspective of generative models and Bayesianism. Second, some simple generative models, including mixtures, MDA2, and Parzen estimates are reviewed. These are used in generating the density estimates for learning metrics (Publication 3) and as benchmark methods for discriminative clustering.

Standard mixture and Parzen estimators are introduced for example in the text book of Ripley [79].

4.1.1 Bayesian perspective to conditional likelihood

Given vectorial samples x paired to nominal values c (class labels in the context of classification), two choices are available for estimating the conditional densities $p(c|x)$. We may either devise a model for the joint distribution $p(c, x)$ and obtain the conditional density estimates by the Bayes rule, or model the conditional probabilities directly. The first option is generatively well justified, but the second case requires more attention. (Section 3.3.5 on the merits of the two approaches for clustering is also partially relevant here.)

In the conditional case, maximum likelihood would straightforwardly maximize $p(C|X, \theta)$ with respect to the parameters θ. The variable X is a covariate: it has a role analogous to that of an explanatory variable in regression analysis.

How does the conditional approach relate to Bayesianism? In short, it is not contradictory, but it contains a hidden assumption of separable parameter priors [32].

We may introduce a separate model for X, parameterized by ψ. Then a full model for the joint density exists, and it has the special factorial form

$$p(X, C|\theta, \psi) = p(C|X, \theta)p(X|\psi) \; .$$

If the priors of the parameters are separable, $p(\psi, \theta) = p(\psi)p(\theta)$, their posterior factors into

$$p(\theta, \psi|X, C) \propto p(\psi|X)\, p(\theta|X, C) \; .$$

We may then optimize θ, the parameters of the conditional model $p(C|X, \theta)$, by treating $p(\psi|X)$ as an unknown constant. The marginal model for X may even be left totally unspecified.

The assumption of separable priors holds naturally if at least one of the priors is vague (constant). On the other hand, if no uncertainty about X exists, as in some experimental settings, then the "model" for X has no parameters.

4.1.2 Gaussian mixture, Parzen, vMF, and MDA2

Gaussian mixtures (see, e.g., [79]) model the density $p(x)$ by the convex combination

$$p(x) = \sum_j \pi_j G(x; \sigma, m_j) \; ,$$

where $\{m_j\}_j$ and σ are the location and spread parameters of the Gaussians G, and π_j are component-specific weights, sometimes referred to as "prior probabilities." The normalization $\int G(x)dx = 1$ must hold. The model is usually fitted by the EM algorithm of Section A.3.2. Gaussian mixture is easily adaptable to other kernel types. If data are normalized to reside on a hypersphere, the von Mises Fisher (vMF) kernel $vMF(x) \propto \exp -\kappa x^T m$ defined on the sphere can be used as in Publications 2 and 5.

MDA2, or "Mixture Density Analysis of Type 2" [42, 41] is a model for the joint density of a nominal c and a vectorial x:

$$p(x, c) = \sum_j \pi_j \xi_{jc} G(x; \sigma, m_j) \; ,$$

with $\sum_c \xi_{jc} = 1$. Again, the model is easily adapted for other kernel types, including vMF. An estimate of the conditional density $p(c|x)$ is obtained from the Bayes formula: $p(c|x) = p(c, x)/p(x) = p(c, x)/\sum_{c'} p(c', x)$, and can be used to derive a metric to the $\mathbb{X}$-space (Section 2; Publication 3).

In both kinds of models the spread parameter σ can be optimized as a part of the M-step of the EM algorithm. Straightforward optimization often causes problems, however, for the parameter is closely tied to the model complexity:

kernels with a large spread tend to produce smooth estimates and vice versa. In our own works the parameter has been chosen by cross-validation.

Parzen estimates ([79] is again a good textbook account) are also additive mixtures of kernels, but a kernel is tied to each data sample, all kernels are usually equally weighted, and the spread parameter, if optimized at all, is chosen by some kind of cross-validation:

$$p(x) = \sum_k G(x; \sigma, x_k) .$$

Except for the spread, there are no parameters to be optimized, which makes Parzen estimates almost non-parametric. An MDA2-like version, suitable for joint density estimation of a discrete and a continuous variable, is obtained by restricting the summation to class-specific samples of the learning set:

$$p(x, c) = \sum_{k:c_k=c} G(x; \sigma, x_k) .$$

The joint density models appearing in the Publications are relatively simple. Partly this has been due to the general problem settings of exploration and clustering, where little prior information is available or typically used. The structurally simple Parzen and mixture estimates are also transparent enough to keep track of their smoothness, which is important especially when the local approximation (2) of Section 2.5 is used.[8] In benchmarking discriminative clustering, an approach with a cluster interpretation is essential, making mixture models a natural alternative.

4.2 Clustering

Clustering refers to a diverse set of methods. The common general goal is to divide data into groups which are more homogeneous internally than mutually. Sometimes the possibility to map future data into the old clusters is essential, and then essentially the whole data space, not only the learning set, must be segmented.

The most common application for clustering is data exploration, but the methods are also applicable to complexity reduction, including compression for transmission and storage, and probably for many other purposes.

A taxonomy for clustering methods is presented in [53]. Clustering may be either hierarchical, with an explicit (hierarchical) structure generated for the clusters, or partitional with no such structure. Partitions may be overlapping (inclusive) or mutually exclusive, and the clustering may be supervised or unsupervised.

[8] The structural simplicity has, for example, allowed the observation that self-organizing maps (Section 2.6.2) are best based on a slightly more smoothed model than the likelihood-optimal one.

In this taxonomy, discriminative clustering is partially supervised, partitional clustering that produces non-overlapping segmentation of the whole data space. "Partially supervised" refers to the fact that supervision is used, but no exact correspondence between the clusters and the given nominal labels is required.

4.2.1 The similarity measure and feature selection

Feature selection has already been discussed on a rather general level in Section 2.1. We will here review the subject from the viewpoint of clustering.

Because the goal of clustering is to produce homogeneous clusters, or clusters of similar data items, the (dis)similarity measure, or proximity measure, is a crucial part of any clustering effort. Similarities may be explicitly given in the form of a similarity matrix, or the data may be available in a raw form leaving the choice of the similarity measure free.

In data analysis, *feature selection* refers to selection, scaling and transforming of input variables before the actual analysis. Because such modifications of data can always be incorporated into the dissimilarity measure, we may conveniently ignore feature selection, at least in theoretical descriptions, and consider only the properties of (dis)similarity measures.

The representation of data may have varying degrees of structure in it, and the degree of given structure restricts the set of applicable proximity measures. The classic categories of binary, nominal (discrete), interval and ratio scales are examples of structure existing in the representation, but the concept is more general: For example tree-structured data items may be clustered given a suitable similarity measure. Co-occurrence data and distributional clustering (Section 3.4) provide another, by classic standards "atypical" structure that has recently found many applications.

Traditionally, multivariate vectorial data have been a very common target of clustering. Sometimes the variables are of a homogeneous origin, with some kind of symmetry (or a lack of knowledge about asymmetry) present in the measurements, that allows us to treat the variables identically. Examples are wind direction (north–south vs. east–west; a physical symmetry exists), pixels of a photograph, and to a degree genes in a microarray experiment.

Often the variable set is heterogeneous, sometimes with no common physical scale. Even different variable types, nominal and discrete, may be intermixed. Unless other criteria are available, preferably an expert of the application field should decide transformation and weights for the input variables. The procedure will be heuristic, for the goal of end result of the clustering is typically poorly defined, therefore invalidating the use of any formal methods. Statistical criteria for the goodness of a clustering solution will then refer to the internal relationships of the data rather than some external costs and benefits. Learning metrics and discriminative clustering potentially help with these classic problems—in situations where an external, objective criterion of relevance is available in the form of data.

Especially in explorative context, however, the feature selection and clustering stages need not be fully separated. Instead, the process can be highly interactive and iterative. When interactivity increases, the immediate understandability of the clustering solutions becomes essential, and can be greatly aided by good visualization [100]. Clustering can for example be performed on the self-organizing map of Section 2.6.2.

4.2.2 Clustering by mixture models and K-means

Of the almost innumerable set of existing clustering algorithms, only those most closely related to the present work, that is, mixture models and K-means, are reviewed here.

Mixture models, discussed from the viewpoint of density estimation in Section 4.1, can be used for clustering simply by associating each kernel with a cluster. Then the kernels of a generative mixture become interpreted as noisy sources. The posterior probability of a source having generated a data item is obtained by the Bayes rule,

$$p(h_i|x,\theta) = \frac{p(h_i, x|\theta)}{\sum_l p(h_l, x|\theta)} = \frac{G_i(x;\theta)}{\sum_l G_l(x;\theta)} \ ,$$

where h_i is a value of the hidden variable indicating kernel identities, and $G_i(x;\theta)$ is the kernel i with parameters θ, evaluated at x. (Incidentally, the hidden variable interpretation is similar to that created for the EM algorithm; c.f. Section A.3.2.)

The posterior probabilities may be interpreted as soft cluster memberships. Hard cluster memberships are conveniently obtained by choosing the cluster with the highest posterior probability.

For data $\{x_k\}_k$, the aim of the **K-means** algorithm is to find a partitioning with minimal total within-cluster distance or *quantization error*

$$E(m,c) = \sum_k d^2(m_{c(x_k)}, x_k) \ .$$

The distances inside the clusters are measured to the prototypical locations $m = \{m_j\}_j$, also called model vectors. Above, the notation $c(x_k)$ is used to denote the cluster membership (or partition index) of the data item x_k. The minimum of the quantization error is sought with respect to both the memberships c and the prototypes m.

If the model vectors m are fixed, the criterion $E(m,c)$ is clearly minimized when each data item is associated to the cluster with the closest model vector: a point x_k belongs to the partition j, i.e. $c(x_k) = j$, if the distance $d(x_k, m_j) \leq d(x_k, m_l)$ for all $l \neq j$. This criterion is defined for all points x of the data space, not just for the data x_k in the learning set, and it divides the space into non-overlapping partitions called *Voronoi regions.* (Ambiguity remains at the borders of the Voronoi regions, but these are of measure zero.)

On the other hand, if the memberships $c(x_k)$ are fixed, the K-means criterion is minimized at

$$m_j = \frac{1}{N(x_k : c(x_k) = j)} \sum_{x_k : c(x_k) = j} x_k \, . \tag{20}$$

We obtain an iterative optimization algorithm, the K-means algorithm, by alternating between these two steps of assigning data to clusters and updating prototypes. The quantization error is downward-bounded by zero and the optimizations always decrease the cost except at a local optimum. Convergence to a local minimum is therefore guaranteed. In practice, to obtain a good solution, it is advisable to run the algorithm several times with different initializations, or to use one of its less aggressive variations.

The K-means algorithm bears a striking resemblance to the EM algorithm of Section A.3.2. For a mixture of isotropic Gaussians of equal spread, the M-step becomes essentially similar to that of (20), with the sum taken over all data with the posterior of the hidden variables, i.e. the mixture proportions $p(h_j|x_k, m)$, as weights. The K-means algorithm can therefore be interpreted as an EM for a Gaussian mixture with all-or-none mixture proportions, which are obtained by letting the spread parameter of the Gaussians to go to zero.

The K-means costs can also be minimized by stochastic approximation (Section A.3.1) which leads to the update rule $m_j := m_j - \alpha(x - m_{c(x)})$, where x has been randomly chosen from the learning set and the parameter α goes slowly to zero during the iteration [62].

The K-means quantization error is essentially identical to the distortion criterion of *vector quantization*, used in compression [33].

4.3 Linear and other projective methods

By projective methods we mean here methods that find continuous representations of data instead of discrete clusters.

Linear methods typically find linear functions of data and are therefore inherently projective. The self-organizing map and related algorithms, reviewed separately in Section 4.5, fall between clustering and projection as they produce discrete representations that has a predefined topography. The results are typically easy to visualize, for the linear functions of data can be used as coordinates.

Other projective methods include multidimensional scaling (MDS) and Sammon mapping. Here the data items get a continuous, vectorial representation but it is not expressible in a closed form as a function of data vectors—except in the case of classic (metric) MDS [14].

Projection into a lower dimensionality naturally works best when the *intrinsic dimensionality* of the data is low, i.e. when the data residing in a high-dimensional space can be described with little residual by a linear or smooth non-linear slow-dimensional manifold.

In exploration, supervised projective methods are often just as useful as the unsupervised ones, because the end result, a projection to a (low-dimensional) continuum is comparatively easy to visualize and understand. Here we start with linear methods and the unsupervised PCA, but later focus on supervised methods, for their spirit is somewhat similar to learning metrics and discriminative clustering.

4.3.1 Principal component analysis (PCA)

Principal component analysis is not directly related to any of the methods introduced or used in the Publications. It is, however, worth introducing here, for it is a prototypical and classic projective unsupervised learning method, highlights important theoretical concepts, and works as an introduction to other, more interesting methods.

By a partial orthonormal basis we mean here a subset of the vectors of an orthonormal basis. Such a basis defines a subspace, and conversely, each subspace has such a basis. The coordinates y of a vector x in a partial orthonormal basis can be conveniently represented as a projective linear mapping: $y = L^T x$, where columns of the matrix L are the basis vectors (note that $L^T L = I$). PCA assumes the reconstruction $\hat{x} = LL^T x$, which includes the projection $L^T x$ into the subspace defined by L. A basis L^T is then found to minimize the average least-squares error $E\left(\|x - \hat{x}\|^2\right)$ between the original, centered vectors[9] x with $E(x) = 0$, and their reconstructions [91]. The average $E(\cdot)$ is either the expectation over a probability density or the mean over a learning set.

It can be shown that the base producing minimal reconstruction error is a composition of the eigenvectors of the covariance matrix $C = E(xx^T)$ associated to its largest eigenvalues [91]. The vectors of the base also coincide with the directions of the largest variance in the data.

In the context of PCA, the variables $y = L^T x$, or the projections of the data onto the eigenvectors, are called *principal components*. The principal components are uncorrelated, with their variances equal to the largest eigenvalues.

For visualization, new variables or their normalized versions can be interpreted as coordinates on a plane or in a 3-D space. This kind of exploratory use of PCA becomes close to classic (metric) multidimensional scaling (MDS; see, e.g., [14]).

Although PCA is unique as an eigensolution for a C-matrix with no equal eigenvalues, the compression $y = L^T x$ is not unique. Identical reconstructions $\hat{x}$ can be produced from all (orthonormal) rotations Ry by $\hat{x} = LL^T x = LR^T RL^T x$. Therefore, each basis RL^T is in the least-squares sense equally good to the basis L^T, and *from the viewpoint of explanatory power (reconstruction error), the PCA solution is not unique.* In a 2-D solution used for visualization this indeterminacy

[9] Centering of data, assumed for simplicity here, can be justified by adding location parameters to the reconstruction model and showing that centering produces an optimal reconstruction [39].

does not matter as long as one does not put too much meaning into the coordinate axes. It may, however, cause serious misinterpretations if pairs of principal components from a higher-dimensional solutions are visualized, or if semantics for single variables are deduced from the coefficients of the basis vectors.

4.3.2 Linear methods supervised by discrete data

An analysis of labeled data may aim either at prediction, e.g., building a classifier for future samples, or on exploring the variation that is somehow related to the labels. In the latter case the most popular goal is to find variation correlating with the labels, although the opposite possibility is also imaginable.

With linear methods the distinction between exploration and black-box predictive models is not as significant as with nonlinear methods, for the linear structure makes even the models optimized for prediction easy enough to interpret.

The description of linear methods below is mainly based on the books of Ripley [79] and Hastie et al. [39].

LDA. Linear discriminant analysis or LDA can be justified either from a generative point of view, or as a linear way to extract variation that is interesting in terms of covariances. We start with the generative motivation.

In general, a *discriminant* is a function associated to a classifier. It maps data x to real values that can be compared either to a threshold or to the values of other discriminants to get the class identity of x.

Linear discriminants are linear functions of data. They define a projection to a lower-dimensional discriminant space, which can be used for explorative purposes. 2-D projections are especially useful for visualization, and linearity makes interpretations easy.

A theoretically optimal classifier is defined in terms of the probabilities $p(c|x)$, so discriminants derived from a generative model would naturally be functions of the conditional probabilities $p(c|x)$ of classes or labels c. They could either be estimated directly, or derived from a model for $p(c, x)$ via the Bayes rule (discriminative and informative learning, respectively, as defined by Rubinstein and Hastie [81] and in Section 3.3.4). Linear discriminants would be justified when the isosurfaces $p(c_i|x) = p(c_j|x)$ or, in the more general case, the surfaces $p(c_i|x) \propto p(c_j|x)$, are affine subspaces. Note that in the case of more than two clusters, the space becomes divided into partitions with potentially complex shapes, although with piece-wise linear borders.

The most popular generative model with such a property is the conditional Gaussian distribution with equal covariances, i.e., the distributions $p(x|c)$ are supposed to be Gaussian with a common covariance matrix W (standing for Within-group). The use of linear discriminants is justified by this generative Gaussian model for $p(c, x)$.

Simply estimating the means and the covariance of the Gaussians separately and plugging them into the optimal linear discriminants leads to *linear discriminant analysis* (LDA). Estimating the model parameters by maximum likelihood gives the same result. LDA can therefore be seen to be based on the multi-gaussian generative model for the *joint* distribution $p(c, x)$. This distinguishes it from logistic regression (see below), which is based on a model for the conditional distribution $p(c|x)$.

Given there are K different labels c with the associated Gaussians and their centroids, the discriminants would span a subspace with dimension $K - 1$. If $K - 1$ is much higher than the easily visualized two or three dimensions, the data may be projected to a (still) lower dimension while trying to maintain interesting variation.

Fisher's linear discriminant. The whole linear discrimination problem can also be formulated directly as a maximization of ratio of interesting and non-interesting variation, defined in terms of variances. One first finds a direction of the original data space where the between-class variance is maximized relative to the within-class variance. This will be the first discriminant direction. Other directions are then found recursively by projecting the data to a subspace normal to the extracted directions, and then again finding an interesting direction by comparing the ratios of the variances. This is the interpretation of linear discriminant analysis as originally proposed by Fisher.

In practice, the data can be 'whitened' by finding coordinates that diagonalize the within-group covariance. A low-dimensional subspace with high variance of the *centroids* is then found by PCA.

LDA has been generalized by using a nonlinear mapping into a high-dimensional space as a preprocessing step [8, 80].

Logistic regression. Logistic regression is a simple model for the dependency of a nominal variable c on a vectorial variable x. The probabilities $p(c|x)$ are expressed as (constant) functions of linear combinations of x:

$$\hat{p}(c|x) = \frac{\exp x^T \beta_c}{\sum_{c'} \exp x^T \beta_{c'}} \ ,$$

with $\{\beta_c\}_c$ being parameters of the class-wise linear combinations. (For full linear generality, the variable x can be thought to be augmented with an extra dimension with the constant value 1.) An indeterminacy exists in this parameterization, for all $\beta_c := \beta_c - \beta'$ with arbitrary β' produce the same probabilities $\hat{p}(c|x)$. This is cured by, e.g., setting $\sum_c \beta_c = 0$. The model is fitted by maximum likelihood.

In the traditional case of two classes the model simplifies into

$$\hat{p}(c|x) = \frac{\exp x^T \beta_c}{\exp x^T \beta_c + \exp x^T \beta_{c'}} = \frac{1}{1 + \exp x^T (\beta_{c'} - \beta_c)} \equiv \operatorname{logit} x^T \beta \ ,$$

with $\beta \equiv \beta_{c'} - \beta_c$.

Note that even in the case of multiple classes

$$\log \frac{\hat{p}(c'|x)}{\hat{p}(c|x)} = x^T(\beta_{c'} - \beta_c) \ ,$$

which could be interpreted as a discriminant for classes c' and c. Somewhat surprisingly, discriminants produced by LDA can be written into the same form (up to a constant, by applying the Bayes rule to $p(c, x)$; see, e.g., [39]). Therefore, although LDA is a joint distribution model (likelihood of $\hat{p}(c, x)$ optimized) while logistic regression is a conditional model (likelihood of $\hat{p}(c|x)$ optimized), from the viewpoint of discrimination the models are identical. The potential merits of conditional and joint modeling approaches are discussed by Efron [28], Ng [69], and more informally in textbooks [39, 79].

Ordinary linear least-squares regression has the probabilistic interpretation as a maximum likelihood regression model with a Gaussian response variable. From this perspective, logistic regression is a generalization of classic linear regression for multinomial responses. In general, such generalizations for various kind of response variables are called generalized linear models (GLM's; see, e.g., [32] for a Bayesian introduction).

Connection to learning metrics and discriminative clustering In its original formulation due to R. A. Fisher, LDA finds a projection to a linear subspace that in a sense of second-order moments maximizes the discriminativity of classes. Logistic regression, on the other hand, explains conditional probabilities $p(c|x)$ of classes c as functions of the inner products $\{x^T b_j\}_j$. A projection is essentially performed here as well, for data affects $\hat{p}(c|x)$ only in the subspace spanned by $\{b_j\}_j$. Discriminativity of the logistic regression projection is maximized in the sense of the likelihood of $\hat{p}(c|x)$ on a data set, just as in generative discriminative clustering (DC; Section 3), an application of learning metrics. Both DC and the linear projective methods, especially logistic regression, therefore maximize discriminativity of c on a representation, which is either a set of clusters or a linear projection.

The learning metric, as defined in Section 2.5, measures distances as a function of local changes of $p(c|x)$ or its estimate $\hat{p}(c|x)$. In this context, a discriminative linear projection producing an estimate $\hat{p}(c|x)$ can be interpreted as a projective change of the metric of the data space. For example, the metric matrix $J(x)$ (see Section 2.5) generated by the two-class logistic regression model is of the form $J(x) = 2p(1-p)bb^T$, where $p \equiv \hat{p}(c|x)$ is the prediction of the model (for a class) and b are the coefficients of the linear model. The properties of metrics generated in this way from predictive linear projections are largely unexplored.

4.3.3 Supervision by vectorial data: canonical correlations

Canonical correlation analysis (CCA) is a technique somewhat similar to LDA, but for pairs of *two* vectorial variables, here x and y ([15] is a good introduction, and cites [49] as the origin of CCA; see also [92]). Linear combinations $\{a_j^T x\}_j$ and $\{b_k^T y\}_k$ of x and y are found that are maximally correlated for $j = k$, while being uncorrelated (but not orthogonal) in $\mathbb{X}$ and $\mathbb{Y}$, i.e., $a_j^T C_{xx} a_k = b_j^T C_{yy} b_k = 0$ for $j \neq k$, where C_{xx} and C_{yy} are covariance matrices of x and y, respectively. In addition, the cross-covariance $a_j^T C_{xy} b_k$ is required to be zero for $j \neq k$. Computationally, CCA reduces to an eigenvalue problem.

Like LDA, CCA is originally motivated as a criteria based on the second moments. CCA is scale invariant: non-singular affine transformations of x and y do not essentially change the solution. The *canonical variates* $\{(a_j\}_j$ and $\{b_k\}_k$ can be used for visualization, discrimination, or as features for further processing.

4.3.4 Multidimensional scaling and Sammon mapping

Multidimensional scaling (MDS; [14]) and Sammon mapping (e.g. [39]) are techniques for representing samples x in low-dimensional coordinates with the intention of preserving given *distances* of the samples. Thus the samples do not need to be vectorial, although vectorial samples with a distance measure can be used. The low-dimensional representation is found by optimizing a *stress function* (a cost function) with respect to the unknown low-dimensional coordinates of data.

As no absolute representation of the samples is used, no mapping from the space of samples to the lower-dimensional space can be generated. Finding a representation for new, unseen samples in an existing projection may therefore be hard. On the other hand, working on the dissimilarities alone makes MDS much more generally applicable than for example the vector-based methods.

MDS and Sammon mapping are not directly related to learning metrics, except as potential applications.

4.4 Learning Vector Quantization (LVQ)

The *Learning Vector Quantization* (LVQ; [62]) family of algorithms is worth mentioning here, because the stochastic learning rule of the discriminative clustering somewhat resembles the learning rule of LVQ.

The goal of the LVQ algorithms is to find a classifier given vector-valued data (x, c) with the class labels c. The classifier is parameterized by the prototype vectors m_j with associated class identities c_j, and a mapping of points of the X-space (and hence the points of the learning set) to the model vectors is defined as in the K-means algorithm (see Section 4.2.2). The mapping defines Voronoi regions into the data space.

In LVQ1, the model vectors of the model are updated by a rule motivated by the stochastic approximation theory:

$$m_j := m_j - \alpha(x - m_j)\, s(c, c_j) \ ,$$

where c is the class of the sample x, and the function $s(c, c_j)$ takes the values ± 1, depending on whether $c = c_j$ or not. In LVQ2.1 [63] two best-matching model vectors are updated, with the sign $s(c, c_j)$ reversed for the second-closest match. It is also proposed that no update should take place unless the sample x lies within a narrow region around the boundary of the Voronoi regions.

The resemblance of the windowed LVQ2.1 algorithm to the stochastic update rule (15) of discriminative clustering is striking.

In discriminative clustering, the window of LVQ2.1 is replaced by the product $y_j(x)y_l(x)$. If the spreads of the kernels $y(x)$ are small enough, the product gets significantly non-zero values only on a narrow, even-width region around the border of the Voronoi regions j and l, which is essentially a smoothed version of the window of LVQ2.1.

But in contrast to the LVQ, in discriminative clustering the sign and amplitude of the updates is (effectively) dynamically adjusted according to the class frequencies of the data falling into the Voronoi regions of the best-matching model vectors. Therefore discriminative clustering does not try to represent only the class boundary but the overall changes in the conditional class densities.

4.5 Self-organizing maps

From the clustering point of view, the self-organizing map (SOM; [61, 63]) is clustering with topological constraints. The “atomic clusters”, also called map units, neurons, or model vectors, are ordered onto a usually two-dimensional rectangular or hexagonal grid. The grid dimensionality or its regularity are not critical; even adaptive grid topologies may be used. One-dimensional maps are sometimes used in special applications, e.g. to prove theoretical results.

The procedures for training the map constrain the solution such that clusters residing nearby on the grid become relatively similar. In an imprecise sense, the grid can therefore be seen as an approximation to a continuous manifold. As the parameters of single clusters have a one-to-one correspondence to the locations of the feature space, the manifold can be thought of as embedded into the feature space. Additionally, a SOM defines a position on the manifold (a cluster identity) not only for the data in the learning set but for all points of the feature space. Taken together, these properties of the algorithm allow us to say that the SOM performs *nonlinear dimensionality reduction.*

More formally, the SOM is parameterized by the *model vectors* m_j. Each vector has a fixed position on the SOM grid. The training algorithm, however, is affected by the grid locations only through the neighborhood structure defined for pairs of model vectors in the form of positive and symmetric coefficients h_{jl}, i.e. $h_{jl} \geq 0$ and $h_{jl} = h_{lj}$. Often these coefficients are given in the form of a continuous *neighborhood function* $h(j, l)$, which is a monotonically decreasing function of the Euclidean distance between the coordinates of the model vectors j and l on the grid.

Mapping of points of feature space to the model vectors is defined in terms of the shortest distance or "best match": points are mapped to their closest model vectors.[10] Closeness, of course, is relative to a metric, which is typically Euclidean or an inner product metric on the unit hypersphere.

The original SOM was defined by the concepts introduced above and a stochastic learning algorithm. For finite data, Euclidean distances and a fixed neighborhood h_{jl}, the SOM has the cost function

$$E(m) = \sum_l \sum_k \|x_k - m_l\|^2 h_{c(x_k)\,l} \ ,$$

where $c(x)$ is the index of the model vector closest to the sample x [45]. For a density $P(x)$ with non-pointwise support there exists no cost function [45]. Note that the cost $E(m)$ has discontinuities at parameter changes that cause the indices $c(x_k)$ to change. Therefore gradient-based algorithms are guaranteed to optimize the parameters $m = \{m_j\}_j$ only locally in the parameter space, within a region on which all the indices $c(x)$ are constant. Still, especially the stochastic gradient-based algorithms seem to work well in practice.

The SOM optimizing algorithms are *competitive learning* in the sense that they can be seen as a realization of a competition process between neighborhoods of the model vectors m_j. Both a stochastic on-line algorithm with a varying neighborhood, and a batch-style algorithm with a fixed neighborhood are commonly used.

The **on-line stochastic algorithm** is motivated by the stochastic approximation theory. At each round of the iteration, a single sample is chosen, and model vectors are moved into the direction of the sample, the magnitude of the move being relative to the product $\alpha_t h_{c(x)l}$, where c is the unit closest to the sample x. Training is repeated with more samples drawn from the distribution $P(x)$ (in practice often from the learning set) and with decreasing updates α_t. In theory, the sequence should satisfy conditions from the stochastic approximation theory that are necessary for convergence, namely $\sum_t \alpha_t = \infty$ and $\sum_t \alpha_t^2 < \infty$, but in practice piecewise-linear sequences are common and found to produce good results.

It is common to reduce the neighborhood size during the stochastic iteration. For a regular 2D-grid with a Gaussian neighborhood this means decreasing the spread of the Gaussian. Such an iteration process is regularizing; the map tends to become globally better ordered, avoiding twisting of the manifold in the feature

[10] This definition is ambivalent for points which are equally close to more than one model vectors. Usually, however, these points are or are assumed to be from a set of measure zero (w.r.t. both the Lebesque measure of the feature space and the distribution of data within the space.)

space. An initially large neighborhood size therefore often leads to a smaller cost for the final neighborhood than would have been achieved by iterating with the final neighborhood alone.

An iteration of the **batch algorithm**, which is essentially a fixed-point iteration resembling the K-means algorithm, updates model vectors to the centroids of all data weighted by the model vector-specific coefficients $h_{c(x_k)l}$.

Note that if the neighborhood coefficients h_{lj} become zero for $l \neq j$ and otherwise a positive constant, the batch SOM algorithm becomes the traditional K-means iteration, i.e. clustering without topographic constants, and the on-line algorithm becomes “stochastic vector quantization” with the quantization error $\|x - m_{c(x)}\|$ as its cost function.

4.5.1 Other similar algorithms

Many modifications of the SOM on-line and batch algorithms exist. With these, a main motivation has been finding an algorithm with a continuous cost function that would also generalize to the asymptotic case of ’infinite data’ or probability distributions without losing its essential properties. Heskes [45] finds the best-matching (winner) unit on the basis of neighborhood averages $\{\sum_l m_l h_{kl}\}_k$ instead of the bare model vectors. As a result of this modification, a hidden variable interpretation appears, and the cost function becomes continuous. Practical advantages of the approach are so far unclear.

Generative topographic mapping (GTM; [11, 9]) is a probabilistic alternative to self-organizing maps, that generates a probability distribution to the $\mathbb{X}$-space as a mixture of Gaussians. Data are mapped to a low-dimensional manifold by maximum likelihood.

5 APPLICATIONS

This section shortly lists the practical applications of learning metrics and discriminative clustering presented in the Publications.

On a more abstract level, discriminative clustering (DC; Section 3) can itself be seen as an application of the learning metrics principle (Section 2). Learning metrics have also been applied to self-organizing maps (Section 2.6.2).

Practical applications include visualizing bankruptcy of companies (Publication 3), text document clustering (Publications 2, 8), and clustering of gene expression (Publication 5).

5.1 SOM, learning metrics and bankruptcy

Expectation of positive future returns is the basis of all financing. Consequently, quantitative prediction of corporate success, including bankruptcy, is a well-studied subject [1, 2]. A divergence from the tradition is to quantitatively *understand* bankruptcy by explorative analysis methods such as the self-organizing maps (SOM) [59].

We combined visualization and prediction in Publication 3. Financial statements of Finnish companies were mapped onto a self-organizing map, using the technique of Section 2.6.2, with the metric computed from a predictor $\hat{p}(c|x)$ of bankruptcy. Two kinds of predictive models were tried: a Parzen kernel estimator and a mixture of Gaussians (MDA2). Despite the low number of bankruptcy companies in the training data, the results were successful in that the SOM's were visually acceptable and separated failed companies well from the mass.

5.2 DC and learning metrics for gene expression

DNA microarrays enable simultaneous measurement of the activity of thousands of genes. In a living cell, genetic code is continuously *transcripted* from DNA to so called messenger RNA (mRNA), the eventual goal being protein synthesis. Microarrays measure the level of this activity, gene-wise, and provide a window to the internal state of the cell (see [21] for a review, and [18] for a textbook account). Transcription is usually measured in several different experimental treatments, and often serially to provide a time series. The large size of microarray data sets and relative lack of knowledge of the regulatory mechanisms of cells constitute a potentially fertile setup for explorative data analysis.

Since Eisen introduced hierarchical clustering into microarray data analysis in 1998 [29], expression data have been clustered in numerous ways. Genes have also been categorized by experts into (hierarchical) groups reflecting the current knowledge of their functions. We have combined these two approaches by clustering gene expression data with DC using the existing functional grouping as a guide. DC would then, hopefully, concentrate on relevant variation in the very high-dimensional expression data, and reveal structure between or inside the existing functional groups. This application is an example of *reclustering*, refining or coarsening an existing clustering, with DC (Publication 5).

In addition, a paper reporting biological interpretations of the DC clusters found from the expression data is in preparation. Outside the scope of this thesis, SOM's in learning metrics have been applied to gene expression data as well [72].

5.3 DC for text documents

Although DC is a clustering model for continuous data, it can be applied to text documents if the documents are first somehow mapped onto a continuous space. Two major alternatives for such a mapping are the *vector-space model*, and a generative probabilistic approach.

Although other kind of feature extraction, such as bigrams, would be possible, we have handled text documents within the classic *bags of words* model. In it, words are thought to occur independently, which erases all structure but word frequencies from the documents and reduces the documents to a form quite analogous to bags of marbles.

In Publication 2 the frequencies were treated using the popular vector space model (VSM; [83]) which represents the documents as real-valued vectors, each word corresponding to a dimension. The vectors are then normalized onto a hypersphere. With the TF-IDF weighting (term frequency, inverse document frequency; [83]) the word frequencies are, before normalization, multiplied by a monotonic function of the number of documents in which they occur. This emphasizes rare words. Irrespective of the weighting scheme, however, the documents get mapped onto the unit hypersphere, with a vector x, $\sum_i x_i^2 = 1$, representing each document.

In Publication 8, the bags of marble analogy was taken a bit further, and the marbles or words in the bag or document were supposed to be sampled from a multinomial distribution with unknown parameters. An estimate of the parameters was then used to encode the documents. In this scheme each document gets mapped onto a distributional space: A distribution x, $\sum_i x_i = 1$, $\forall i : x_i \geq 0$ represents a document. It would be more appropriate to take the sampling uncertainty into account, and to represent documents by the full posterior distributions of the parameters. This, however, turned out to be computationally very difficult.

Discriminative clustering would then be applicable if we had a suitable relevance criterion, in the form of auxiliary data, for the document vectors. The partitions would reside either on the unit hypersphere, or in a distributional space, with the dot product metric and the Kullback-Leibler divergence being suitable choices, respectively, for parameterizing the Voronoi regions. (Kullback-Leibler divergence is not a metric and, although used in Publication 8, may pose certain problems if used for Voronoi regions.)

In our works, we have studied the problem of clustering scientific abstracts. Because keywords chosen by the authors of the abstracts are available, we have guided the feature selection in documents by the keywords. Once the clusters have been found, they would be usable for new documents without the keywords. Measured by mutual information to classification created by informaticians, and compared to MDA2 and non-supervised clustering, the approach proved successful both within the VSM (Publication 8) and in the distributional (Publication 8) approach.

Although the setup is confusingly close to that of simple co-occurrence analysis with two variables, it is different, for it actually has *three* random variables: documents, potential words appearing in the abstracts, and potential keywords. The DC model is for the relationship $p(c|x)$ of keywords and documents, rather than for the documents themselves. The mutual information between clusters and keywords is (asymptotically) maximized—in this sense the clusters try to predict the keywords given by the authors. In the application phase the keywords are not needed, because the clusters are originally defined in terms of the abstracts alone.

6 CONCLUSIONS

The conceptual contributions of this work have been learning metrics, a new kind of exploration principle, and its applications, especially discriminative clustering. Roughly put, learning metrics allows one to restrict exploration onto a manifold spanned by dependency to an auxiliary random variable. Learning metric is, in certain kind of setups, able to improve visualizations by self-organizing maps. Discriminative clustering (DC) asymptotically works on such a manifold, but in practice is a semi-generative model with connections to Bayesian contingency tables. While these methods are related to many existing frameworks, especially DC is related to the information bottleneck, this work provides a more or less integrated viewpoint to the discriminative exploration of continuous data.

In the experiments, a self-organizing map (SOM) implemented in the new metric was able to separate bankruptcy companies (statistically significantly) better than a standard SOM, while it maintained the good visual characteristics of the usual self-organizing map (evaluated subjectively). The approach is generally useful at least as far as it is regularized with Euclidean metric, with the weight of the two metrics evaluated by cross-validation. Discriminative clustering, in turn, has been applied to analysis of gene function and clustering of text document. In both applications, it has statistically significantly outperformed alternative methods, including joint distribution clustering. Gene clusters obtained by DC and related methods have also offered new biological hypotheses.

The learning metric principle—as implied by its name—defines a metric to the primary data space, usable by many general-purpose explorative methods. The principle, however, does not define how to estimate the metric in a practical data analysis task. Two approaches are obviously available. The first straightforwardly estimates conditional densities of auxiliary data, from which the metric can be derived, while the second in a way or another embeds the density estimation into the exploratory method. Self-organizing maps in learning metrics are an example of the former, while discriminative clustering demonstrates the latter approach. Both ways have their pros and cons: A separate density estimator is easy to implement and intuitive, but has its own, separate criterion of optimality, while integration of density estimation into the explorative method requires theoretical work and is able to follow the principal idea of learning metrics only asymptotically.

With the separated density estimation approach, approximations to the full Riemannian metric are a necessity. In addition to the simple local approximation introduced in this work, more accurate alternative approximations would increase accuracy, but they would also increase computational load significantly above the standard SOM. Another potential improvement would be to take the uncertainty of the density estimates into account.

The optimization of DC has room for improvement. An efficient batch-mode algorithm is not totally out of the question. A big question not even touched is model selection: Besides cross-validation, how to apply known model selection criteria to decide an optimal number of clusters. Finally, DC is a kind of simplistic prototype with extension potential. The auxiliary variable could have a more complex distribution. It could be continuous, and with a structure of its own to be optimized. The cluster structure could be constrained to be topographical or agglomerative. For example, the topological DC pursued in Publication 4 needs further study.

APPENDICES

A Theoretical background

This appendix briefly reviews some mathematical and learning-theoretic concepts appearing in this work. (In addition, a few classic learning algorithms are reviewed in Appendix 4.)

A.1 Information theory and information geometry

Information theory is central to the formulation of learning metrics, discriminative clustering for probability distributions, and information bottleneck, a closely related paradigm. A few basic concepts are listed here, mostly on the level of formulas. Many textbooks are available on the subject, including one from Cover and Thomas [22]. The classic book by Kullback [66] deals with applications to statistics, including the Fisher information matrix (Section A.1.5).

A.1.1 Entropy and cross-entropy

The *entropy* $H(X)$ of a discrete random variable X with values x_i is

$$H(X) = -\sum_i P(x_i) \log P(x_i) \ . \tag{21}$$

In coding terms, the entropy measures the average length of a theoretical optimal code for random symbols x_i [84, 22]. The *cross-entropy* $-\sum_i P(x_i) \log Q(x_i)$, on the other hand, measures the average length of codes for symbols following P when the code is optimized for Q.

Entropy is also generally used as an evenness measure for probability distributions. It is not directly relevant to this work in its basic form, but serves as a foundation for the other concepts introduced below. An exception is the entropic regularization of discriminative clustering (Section 3.2.5).

A.1.2 Kullback-Leibler divergence

The Kullback-Leibler divergence, or shortly the KL-divergence, is a popular measure of similarity for two distributions defined over the same set of events. For discrete distributions, the definition is

$$D_{\mathrm{KL}}(P, Q) = \sum_i P(x_i) \log \frac{P(x_i)}{Q(x_i)} \ . \tag{22}$$

Note that $D_{\mathrm{KL}}(P, Q) \geq 0$, and that $D_{\mathrm{KL}}(P, Q) = 0$ if and only if $P = Q$.

The definition for continuous distributions is analogous. In general the KL-divergence is asymmetric. It follows that the divergence does not generate a metric for the distributions. For nearby distributions, however, the divergence is asymptotically symmetric.

Notably, the continuous form of KL-divergence is “measure invariant” in the sense that it does not depend on the (Lebesque) measure with respect to which densities are expressed. (Measure invariance makes the mutual information, a special form of KL-divergence, similarly invariant.)

A.1.3 Entropy for continuous distributions

Differential entropy, or the entropy of continuous random variables, is not a critical part of this work. It is, however, worth a short discussion because of a subtlety in the concept of entropy for continuous variables.

Definition of entropy for continuous variables faces some difficulties which are easily ignored if the summation in (21) is simply changed to an integral. The continuous case would be most conveniently defined as a limit of ordinary entropy when the number of possible values of x goes to infinity, and to keep the definition tied to ordinary entropy the limit would be most natural as a binning (discretization) process of the continuous variable, with an increasing number of bins. Unfortunately, such an entropy would be infinite, which also implies a fundamental conceptual difficulty: Symbols from an infinite code book would really carry an infinite amount of information.

A satisfactory compromise is to accept the infinity of ordinary entropy, and define another concept, called *differential entropy*, just as the integral

$$h(p(x)) = -\int_X p(x) \log p(x) dx = -\int_X dP(x) \log p(x) \ .$$

Note that unlike the KL-divergence, differential entropy is not a property of the random variable alone, for it depends on the measure with respect to which the density $p(x)$ is expressed.

The connection between differential entropy and ordinary entropy is the following [22]. If $p(x)$ is regular enough and the variable X is quantized to regions of equal size Δ to get the discrete variable X^Δ, then

$$H(X^\Delta) + \log \Delta \to h(p(x)) \text{ when } \Delta \to 0 \ . \tag{23}$$

That is, the two entropies differ by a “constant” that just happens to be infinite. Measure dependency of the differential information swims in through the equality of the bin sizes, which actually refer to the Lebesque measure. For KL-divergence, the constants $\log \Delta$ cancel out, which makes KL-divergence invariant to the underlying measure and also erases the conceptual difference between the continuous and the discrete case: Binning continuous variables and measuring the KL-divergence of the resulting discrete variables produces good, converging estimates of the KL-divergence of the continuous variables.

A.1.4 Mutual information

Mutual information is the measure of dependence used by discriminative clustering and the information bottleneck.

For two discrete random variables X and Y, it is defined by

$$I(X;Y) = \sum_{ij} P(x_i, y_j) \log \frac{P(x_i, y_j)}{P(x_i)P(y_j)} \ ,$$

and for continuous variables by

$$I(X;Y) = \int_{xy} dP(x,y) \log \frac{p(x,y)}{p(x)p(y)} \ .$$

Mutual information is the Kullback-Leibler divergence (Section A.1.2) between the joint distributions and the product of the marginals, and therefore $I(X;Y) = 0$ if and only if X and Y are independent. Mutual information is perhaps the most widely used measure of dependency for probability distributions.

A.1.5 Fisher information matrix

The Fisher information matrix

$$\begin{aligned} J(\theta) = &\int_D \frac{\partial^2}{\partial \theta^2} \log p(D|\theta) \, dP(D|\theta) \\ &= \int_D \left[\frac{\partial}{\partial \theta} \log p(D|\theta) \right] \left[\frac{\partial}{\partial \theta} \log p(D|\theta) \right]^T dP(D|\theta) \ . \end{aligned}$$

measures the sensitivity of a generative model $p(D|\theta)$ to differential changes of the parameters θ, in the sense that the quadratic form $d\theta^T J(\theta) d\theta$ approximates the Kullback-Leibler divergence, $D_{\mathrm{KL}}(p(D|\theta), p(D|\theta + d\theta))$, of the predicted distributions of data [66].

In the learning metrics approach, a metric is defined to a *data space* $\mathbb{X}$, on the basis of conditional distributions $p(c|x)$ conditioned by points of the space. Distances are measured by a quadratic form of a matrix analogous to the Fisher matrix, with x taking the role of θ.

A.1.6 Amari's information metric

Information geometry [4, 6, 57, 68] is a framework for statistical inference based on differential geometry. A basic concept of information geometry is the *Fisher metric* or *information metric* [6], that is essentially similar to the "learning metric" of this work, except for its context.

For neighbourhood points θ and $\theta + d\theta$, the information matrix is defined by the quadratic form of the Fisher information matrix,

$$d_F(\theta, \theta + d\theta) = d\theta^T J(\theta) d\theta \ .$$

Longer distances are defined as integrals along the shortest paths between the points. This generates a *Riemannian metric* into the parameter space of a model family.

Note that one could also obtain a kind of global similarity measure by measuring differences between points θ_0 and θ_1, potentially residing far from each other, directly by the KL-divergence between the conditional distributions $p(D|\theta_0)$ and $p(D|\theta_1)$. Such similarities would not generate a metric, however, for they would not in general be symmetric. They would also break the topology of the parameter space, for disconnected regions with identical generated distributions $p(D|\theta)$ would be at the distance zero from each other.

A.1.7 Differential geometry: coordinates and tensors

Differential geometry is potentially useful for the learning metrics approach, but too involved for a proper treatment here. We will therefore adhere to an informal briefing of some basic concepts related to this work. The books on information geometry [4, 6, 57, 68] would probably serve as good introductions to the basic concepts of differential geometry as well.

Differential geometry deals with "smooth continuous point sets", or *manifolds*, and concentrates on the properties of the manifolds that are invariant to the coordinate system used to represent the points. In this context, the coordinates are a *representation* for the points p of the manifold. Instead of denoting a point of the data space by x, one would then write $x(p)$ to emphasize the relativity: x is the vectorial value of a more or less artificial coordinate function $x(\cdot)$ for the point p. For differential geometry, the interesting properties of the points p are those that do not depend on x.

A metric of the data space is a relationship between pairs of points, and therefore in principle coordinate-free. For example, the information metric of the previous section (A.1.6) is originally defined for the parameters θ. From the viewpoint of differential geometry, however, θ are just coordinates, and the metric is really for the models M, with $\theta(\cdot)$ being just a convenient representation.

It is possible to use other coordinates $\theta'(M)$ in place of $\theta(M)$ and still refer to the same metric for the models M. The Fisher matrix $J(\theta)$ is a function of the derivatives $\partial p(D|\theta(M))/\partial\theta(M)$, and therefore it changes with the coordinate system according to some transformation rules. For $\theta'(\cdot)$ we would then have the corresponding $J'(\cdot)$.

Clearly, just as the coordinates are a representation for the points, the matrix J is a representation for an underlying geometric object. The object is called a *tensor*, and the collection of the metric-defining tensors over all models M is a tensor field. Once a coordinate system is chosen, tensors can be represented by matrices.

A.1.8 Kullback-Leibler divergence and Fisher information

Symmetry of the KL-divergence for close-by distributions. Let $\{\epsilon_i\} = \{p_i - q_i\}$ be the difference in probabilities assigned by distributions P and Q for the nominal values i. From the definition (22) we may then write

$$D_{\mathrm{KL}}(P,Q) = \sum_i p_i \log \frac{p_i}{p_i+\epsilon_i} = -H(P) - \sum_i p_i \left(\log p_i + \frac{\epsilon_i}{p_i} - \frac{\epsilon_i^2}{p_i^2} + O(\epsilon_i^3) \right)$$
$$= \sum_i \frac{\epsilon_i^2}{p_i} + O(\epsilon_{max}^3) \ , \quad (24)$$

where ϵ_{max} is the largest of $\{\epsilon_i\}$ measured by their absolute value. To get the KL-divergence in the opposite direction, from Q to P, we substitute $p_i := p_i + \epsilon_i$ and $\epsilon_i := -\epsilon_i$ to the previous result, to obtain

$$D_{\mathrm{KL}}(Q,P) = \sum_i \frac{\epsilon_i^2}{p_i+\epsilon_i} + O(\epsilon_{max}^3) = \sum_i \frac{\epsilon_i^2}{p_i} + O(\epsilon_{max}^3) \ ,$$

with the latter equality resulting from $1/(p_i+\epsilon_i) = 1/p_i + O(\epsilon_i)$. The symmetry of the KL-divergence therefore holds up to $O(\epsilon_{max}^3)$.

KL-divergence and Fisher information. If the distributions P and Q are conditional on or parameterized by a vectorial variable x, and denoted just by their probabilities $\{p_i(x)\}$ and $\{p_i(x+\Delta x)\}$, we may approximate

$$\epsilon_i \equiv p_i(x+\Delta x) - p_i(x) = \left(\frac{\partial p_i}{\partial x} \right)^T \Delta x + O(\|\Delta x\|^2) \ ,$$

assuming the mapping from x to $\{p_i\}$ is continuous and differentiable. Substitution to (24) gives

$$D_{\mathrm{KL}}(p(x), p(x+\Delta x)) =$$
$$\sum_i \frac{1}{p_i} (\Delta x)^T \left(\frac{\partial p_i}{\partial x} \right) \left(\frac{\partial p_i}{\partial x} \right)^T (\Delta x) + \sum_i O(\epsilon_i \|\Delta x\|^2) + O(\epsilon_{max}^3) \ .$$

Because $\{\epsilon_i\}$ as well as ϵ_{max} are $O(\|\Delta x\|)$, the O-terms reduce to $O(\|\Delta x\|^3)$. The first term may be conceptualized either as a square of inner products (which it originally was), or as a quadratic form with the outer product of the gradients forming a matrix. With the latter interpretation, we may move the sum inwards and finally arrive at

$$
\begin{aligned}
&D_{\mathrm{KL}}(p(x), p(x+\Delta x)) = \\
&\qquad (\Delta x)^T \left[\sum_i \frac{1}{p_i} \left(\frac{\partial p_i}{\partial x} \right) \left(\frac{\partial p_i}{\partial x} \right)^T \right] (\Delta x) + O(\|\Delta x\|^3) \\
&\quad = (\Delta x)^T \left[\sum_i p_i \left(\frac{\partial \log p_i}{\partial x} \right) \left(\frac{\partial \log p_i}{\partial x} \right)^T \right] (\Delta x) + O(\|\Delta x\|^3) \\
&\qquad\qquad = \sum_i (\Delta x)^T J(x) (\Delta x) + O(\|\Delta x\|^3) \ ,
\end{aligned}
$$

where $J(x)$ is the Fisher information matrix of Sections 2.5 and A.1.5.

A.2 Generative models and their estimation

As their name indicates, *generative probabilistic models* generate a probability distribution to the space of possible data. The model is estimated by optimizing a probabilistic measure of match between the data and the model. Of these, maximum likelihood and maximum posterior estimates are used in the Publications.

For practicality, the model is usually constrained to a prechosen *model family* parameterized by a vector θ. Then the model, or its *likelihood*, is written $p(D|\theta)$, with D being data and θ the parameters. With the model family M made explicit, one can also write $p(D|\theta, M)$.

Bayesianism refers to the philosophical view that allows probabilities to be subjective instead of requiring them all to refer to objective frequencies of events. An interesting philosophical discussion of the history of probability from the Bayesian point of view is offered by Jaynes [54], and the book by Gelman et al. is a thoroughful introduction to practical Bayesian data analysis [32]. Bayesianism does not change the formal structure of probability, but extends its applicability: Classic laws of probability can be deduced also from the subjectivist ground [23].

For example, statistical inference on means of two populations is classically based on the notion of repeated tests: The probability of achieving the observed result in repeated tests, assuming no difference in means, is evaluated. Once the representation of subjective beliefs as probabilities is allowed, one can actually compute the probability of a certain kind of difference in means, *given* the prior subjective information.

The subjectivity, although ideal for many cases of machine learning, is actually problematic in many practical applications. Requirements of objectivity probably set more limits for the applicability of Bayesianism than any philosophical dispute.

In modelling, the Bayesian viewpoint justifies the inversion of the likelihood or, more accurately, makes the result obtained by the Bayes rule acceptable:

$$p(\theta|D) \propto p(D|\theta)p(\theta) \ .$$

It has been possible to compute the probabilities of various models, parameterized by θ. On the right-hand side, $p(\theta)$ is the *prior* over models. It represents the modeller's conception of probable values of θ, or in other words, his or her prior knowledge of θ. In a case of a strong, or informative, prior, new data makes little difference. Modeller's ignorance, on the other hand, makes $p(\theta)$ (in some sense) even and smooth, and then the likelihood $p(D|\theta)$ dominates.

Note that because the prior is subjective, the posterior probabilities $p(\theta|D)$ are also subjective. We are actually just manipulating subjective beliefs on the basis of emerged data, in a justified way. It is also important to realize that the uncertainty forces the theoretical Bayesian modeller to consider *all* models (within the chosen set, or model family) as potential explanations for the world. In practice the posterior is approximated by one or a few models, or it may be approximated by a simple analytical form effectively parameterized as one model.

The generative approach, including Bayesianism, is in wide use, not only with small models tuned for a particular purpose, but also in flexible models optimized for big data sets. Although the Bayesian viewpoint has a strong theoretical support as manipulation of uncertain beliefs, it is less clear how useful probability distributions are in representing prior and posterior beliefs in the case of complex models, where representation resources for the posteriors may be sparse, computational resources limited, and the semantics of the parameters not known even after fitting the model. The Bayesian methods of model fitting, however, may be valuable even when priors become just a technical property of the model.

In the frequentist setting, maximum likelihood is the criterion for choosing models. The model maximizing the probability of observed data is chosen, i.e. $p(D|\theta)$ is maximized with respect to θ. From the Bayesian perspective, as $p(D|\theta) \propto p(\theta|D)/p(\theta)$, maximum likelihood amounts to finding the most probable model with a uniform (constant) prior $p(\theta)$. Of course, such a prior is rather artificial unless the parameterization by θ, including its Lebesque measure with respect to which $p(\theta)$ is represented, happens to be somehow special.

In the maximum a posteriori (MAP) estimation, the model which maximizes $p(\theta|D)$ for another, explicitly set prior $p(\theta)$, is chosen. MAP avoids the theoretical problem of an arbitrary prior, but may find isolated peaks of $p(\theta|D)$ far away from a real concentration of probable models.

Note that these theoretical considerations have little relevance to the application of MAP to discriminative clustering, where priors are unimodal and do not, at least in their current state, really present prior knowledge.

A.3 Optimization

In addition to a more or less standard conjugate gradient method (from [75]), two statistically oriented optimization algorithms, stochastic approximation and the EM algorithm, have been used in the Publications.

A.3.1 Stochastic approximation

Stochastic approximation (see [16]) is applied to self-organizing maps in Publication 3, and to discriminative clustering in Publication 5. In both cases a function of the form

$$F(\theta) = \int f(\theta, x) p(x) dx \tag{25}$$

is optimized. The theory of stochastic approximation allows this to be done by sampling x repeatedly from $p(x)$ and updating the parameters θ by

$$\theta_{t+1} = \theta_t \pm \alpha_t \nabla_\theta f(\theta_t, x_t) \ ,$$

the sign depending on the polarity of the desired optimum. The coefficients $\alpha_t > 0$ decrease monotonically toward zero. The procedure converges if certain, quite general conditions are fulfilled. Theoretically necessary but practically necessarily neglected conditions for convergence are $\sum_t \alpha_t = \infty$ and $\sum_t \alpha_t^2 < \infty$.

The principal advantage of stochastic approximation is the disapperance of the integral from the update formula, which often leads to very simple, computationally efficient and intuitive optimization rules, such as (15) in Section 3.2.6. Stochastic approximation is often used for *on-line learning*, conceptualizable as the optimization of a cost function of type (25).

A.3.2 The EM algorithm

The EM or expectation maximization algorithm is popular in the optimization of the likelihood of mixture models (although this is not its only application). MDA2 and other mixture models (Section 4.1) are used in Publications 2, 5, 6, and 3 as benchmark methods. The classic reference in EM is the paper of Dempster et al. [24], but perhaps more useful treatments, parts of which have been used as a foundation for the introduction below, appear in more modern sources [7, 17, 34].

Suppose we wish to fit a parameterized generative model by maximizing the log-likelihood $L(\theta) \equiv \log p(D|\theta)$. Sometimes the problem can be converted to an iteration of easier optimization problems by introducing additional, imaginary and unknown data H, and by expanding the likelihood into the form

$$L(\theta) = \sum_H q(H|\theta) \log P(D|\theta) = \mathcal{F}(q, \theta) + D_{\mathrm{KL}}(q(H|\theta), p(H|D, \theta)) \ , \tag{26}$$

where

$$\mathcal{F}(q, \theta) \equiv \sum_H q(H|\theta) \log p(D, H|\theta) - \sum_H q(H|\theta) \log q(H|\theta) \ ,$$

and $q(H|\theta)$ is an artificial distribution over the hidden data, to be treated as a set of extra parameters—hence optimization of L with respect to $q(H|\theta)$ makes sense[11]. One then maximizes $\mathcal{F}(q,\theta)$ by alternating maximization with respect to the original parameters θ and the distribution $q(H|\theta)$ (this leads to the original goal of maximizing $L(\theta)$ as explained below):

E-step: As $L(\theta)$ in (26) does not really depend on q, maximization of $\mathcal{F}$ with respect to q with θ fixed leads to the minimization of the Kullback-Leibler divergence, i.e., to the theoretical optimum $q = p(H|D,\theta)$, or in some applications to a best possible match. This part of the maximization procedure is usually called the E-step, for it can be interpreted as finding the posterior $p(H|D,\theta)$ or its approximation, over which an expectation is later taken in the M-step.

M-step: Once q is fixed to $p(H|D,\theta)$ or its approximation, maximization of $\mathcal{F}$ with respect to θ becomes simply maximization of the expectation

$$\sum_H q(H|\theta)\log p(D,H|\theta) = E_{q(H|\theta)}\log p(D,H|\theta)\ ,$$

for the latter term of $\mathcal{F}$, the entropy of q, is independent of θ. This maximization is the M-step, and again it increases $\mathcal{F}$ unless θ already happens to be optimal. The algorithm then continues from the E-step.

If optimization with respect to $q(H|\theta)$ in the E-step is exact, $q(H|\theta) = p(H|D,\theta)$ and the Kullback-Leibler divergence in (26) is zero, and the iteration leads to the original goal of maximal $L(\theta)$. If the $q(H|\theta)$ found in the E-step fails to be exactly $p(H|D,\theta)$, for example due to being parameterized as a simple functional form, the Kullback-Leibler divergence in (26) remains positive and we have maximized a lower bound of the likelihood.

Mixture models are typically of the form $p(x|\theta) = \sum_i \pi_i \exp C_i(x;\theta)$, where $\exp C_i$ are the mixture components, often either explicitly exponential or combinatorial so that taking a logarithm is easy and beneficial. For simplicity, we will below consider the unrealistic case of a data set with only one sample: $D = \{x\}$; the realistic case of many samples is essentially similar but adds confusing extra summations and indexing.

The log-likelihood $\log \sum_i \pi_i \exp C_i(D;\theta)$ is hard to optimize as the summation over mixture components prevents any simplification. If one interprets $p(D,H|\theta) = \exp C_i(D;\theta)$ with $q(H) = \{\pi_i\}_i$, the marginalized, original version $p(D|\theta)$ of the model remains intact except for the parameters π_i becoming replaced by q, which is a purely metaphysical difference. The E-step can then always be solved analytically:

$$q = p(H|D,\theta) = \left\{\frac{\exp C_i(D;\theta)}{\sum_{i'}\exp C_{i'}(D;\theta)}\right\}_i\ .$$

In the M-step one maximizes the weighted sum

$$\sum_H q(H|\theta)\log p(D,H|\theta) = \sum_H q(H|\theta)\log\exp C_i(D;\theta) = \sum_H q(H|\theta) C_i(D;\theta)\ ,$$

[11] The conditioning of $q(H|\theta)$ with respect to θ may confuse, but is best to be thought of just as “parameters q for a certain value of θ”.

which is usually much easier to optimize than the original likelihood of the form $\log \sum \exp(\cdot)$.

REFERENCES

[1] E. I. Altman. Financial ratios, discriminant analysis, and the prediction of corporate bankruptcy. *The Journal of Finance*, pages 589–609, 1968.

[2] E. I. Altman. *A complete guide to predicting, avoiding, and dealing with bankruptcy.* Wiley, New York, 1983.

[3] S. Amari and S. Wu. Improving support vector machine classifiers by modifying kernel functions. *Neural Networks*, 12:783–789, 1999.

[4] S. Amari. *Differential-Geometrical Methods in Statistics.* Springer, New York, 1990.

[5] S. Amari. Natural gradient works efficiently in learning. *Neural Computation*, 10:251–276, 1998.

[6] S. Amari and H. Nagaoka. *Methods of Information Geometry*, volume 191. American Mathematical Society and Oxford University Press, 2000.

[7] P. Baldi and S. Brunak. *Bioinformatics: the machine learning approach.* Bradford, London, 3rd edition, 1999.

[8] G. Baudat and F. Anouar. Generalized discriminant analysis using a kernel approach. *Neural Computation*, 12:2385–2404, 2000.

[9] C. M. Bishop, M. Svensén, and C. K. I. Williams. Developments of the generative topographic mapping. *Neurocomputing*, 21:203–224, 1998.

[10] C. M. Bishop. *Neural Networks for Pattern Recognition.* Oxford University Press, Oxford, 1995.

[11] C. M. Bishop, M. Svensén, and C. K. I. Williams. GTM: A principled alternative to the self-organizing map. In M. C. Mozer, M. I. Jordan, and T. Petche, editors, *Advances in Neural Information Processing Systems 9*, pages 354–360. MIT Press, Cambridge, MA, 1997.

[12] D. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. In T. G. Dietterich, S. Becker, and Z. Ghahramani, editors, *Advances in Neural Information Processing Systems 14.* MIT Press, Cambridge, MA, 2002.

[13] D. Blei, A. Y. Ng, and M. I. Jordan. Latent Dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

[14] I. Borg and P. Groenen. *Modern Multidimensional Scaling.* Springer, New York, 1997.

[15] M. Borga. Canonical correlation - A tutorial.
`http://www.isy.liu.se/~magnus/cca/`, 1999.

[16] L. Bottou. On-line learning and stochastic approximations. In D. Saad, editor, *On-line learning in neural networks*, pages 9–42. Cambridge University Press, Cambridge, UK, 1998.

[17] W. Buntine. Variational extensions to EM and multinomial PCA. In T. Elomaa, H. Mannila, and H. Toivonen, editors, *Proceedings of the ECML'02, 13th European Conference on Machine Learning*, Lecture Notes in Artificial Intelligence 2430, pages 23–34. Springer, Berlin, 2002.

[18] N. A. Campbell, J. B. Reece, and L. G. Mitchell. *Biology.* Benjamin Cummings, San Francisco, 6th edition, 2001.

[19] G. Chechik and N. Tishby. Extracting relevant structures with side information. In *Advances in Neural Information Processing Systems 14*. MIT Press, Cambridge, MA, 2002.

[20] Y. Cheng. Mean shift, mode seeking, and clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17:790–799, 1995.

[21] Chipping forecast. *Nature Genetics*, 2002. Supplement to vol. 32.

[22] T. M. Cover and J. A. Thomas. *Elements of Information Theory.* Wiley, New York, 1991.

[23] R. T. Cox. Probability, frequency and reasonable expectation. *American Journal of Physics*, 17:1–13, 1946.

[24] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B*, 39:1–38, 1977.

[25] C. Domeniconi, J. Peng, and D. Gunopulos. An adaptive metric machine for pattern classification. In T. K. Leen, T. G. Dietterich, and V. Tresp, editors, *Advances in Neural Information Processing Systems 13*, pages 458–464. MIT Press, 2001.

[26] C. Domeniconi, J. Peng, and D. Gunopulos. Locally adaptive metric nearest-neighbor classificaton. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(9):1281–1285, 2002.

[27] H. L. Dreyfus. *What Computers Can't Do.* Harper Collins, 1979. Revised edition. Originally published 1972.

[28] B. Efron. The efficiency of logistic regression compared to normal discriminant analysis. *Journal of the American Statistical Association*, 70:892–898, 1975.

[29] M. B. Eisen, P. T. Spellman, P. O. Brown, and D. Botstein. Cluster analysis and display of genome-wide expression patterns. *Proceedings of the National Academy of Sciences*, 95:14863–14868, 1998.

[30] J. W. Fisher III and J. Principe. A methodology for information theoretic feature extraction. In A. Stuberud, editor, *Proceedings of the IEEE International Joint Conference on Neural Networks (IJCNN'98)*, volume 3, pages 1712–1716. IEEE, Piscataway, NJ, 1998.

[31] N. Friedman, O. Mosenzon, N. Slonim, and N. Tishby. Multivariate information bottleneck. In *Proc. Seventeenth Conference on Uncertainty in Artificial Intelligence (UAI)*, pages 152–161. Morgan Kaufmann Publishers, San Francisco, CA, 2001.

[32] A. Gelman, J. B. Carlin, H. S. Stern, and D. B. Rubin. *Bayesian Data Analysis.* Chapman & Hall, Boca Raton, FL, 1995.

[33] A. Gersho and R. M. Gray. *Vector quantization and signal compression.* Kluwer, Boston, 1992.

[34] Z. Ghahramani and S. Roweis. Probabilistic models for unsupervised learning. NIPS Tutorial, 1999.

[35] A. Globerson and N. Tishby. Sufficient dimensionality reduction. *Journal of Machine Learning Research*, 3:1307–1331, 2003.

[36] I. J. Good. On the application of symmetric Dirichlet distributions and their mixtures to contingency tables. *Annals of Statistics*, 4(6):1159–1189, 1976.

[37] L. A. Goodman. The analysis of cross-classified data having ordered and/or unordered categories: Association models, correlation models, and asymmetry models for contingency tables with or without missing entries. *Annals of Statistics*, 13(1):10–69, 1985.

[38] D. Hand, H. Mannila, and P. Smyth. *Principles of Data Mining.* MIT Press, Cambridge, MA, 2001.

[39] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning.* Springer, New York, 2001.

[40] T. Hastie and R. Tibshirani. Discriminant adaptive nearest neigbor classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18:607–616, 1996.

[41] T. Hastie and R. Tibshirani. Discriminant analysis by Gaussian mixtures. *Journal of the Royal Statistical Society B*, 58:155–176, 1996.

[42] T. Hastie, R. Tibshirani, and A. Buja. Flexible discriminant and mixture models. In J. Kay and D. Titterington, editors, *Neural Networks and Statistics.* Oxford University Press, Oxford, 1995.

[43] R. Hecht-Nielsen. *Neurocomputing.* Addison-Wesley, 1990.

[44] J. Hertz, A. Krogh, and R. G. Palmer. *Introduction to the Theory of Neural Computation.* Addison-Wesley, 1991.

[45] T. Heskes. Energy functions for self-organizing maps. In E. Oja and S. Kaski, editors, *Kohonen Maps*, pages 303–316. Elsevier, Amsterdam, 1999.

[46] T. Hofmann. Unsupervised learning by probabilistic latent semantic analysis. *Machine Learning*, 42:177–196, 2001.

[47] T. Hofmann and J. Puzicha. Statistical models for co-occurrence data. A.I. Memo 1625, MIT, 1998.

[48] T. Hofmann, J. Puzicha, and M. I. Jordan. Learning from dyadic data. In M. S. Kearns, S. A. Solla, and D. A. Cohn, editors, *Advances in Neural Information Processing Systems 11*, pages 466–472. Morgan Kaufmann Publishers, San Mateo, CA, 1999.

[49] H. Hotelling. Relations between two sets of variates. *Biometrika*, 28:321–377, 1936.

[50] T. Jaakkola, M. Diekhans, and D. Haussler. Using the Fisher kernel method to detect remote protein homologies. In *Proceedings of the 7th Annual International Conference on Intellegent Systems for Molecular Biology (ISMB'99)*, pages 1149–1158. AAAI Press, Menlo Park, CA, 1999.

[51] T. S. Jaakkola and D. Haussler. Exploiting generative models in discriminative classifiers. In M. S. Kearns, S. A. Solla, and D. A. Cohn, editors, *Advances in Neural Information Processing Systems 11*, pages 487–493. Morgan Kaufmann Publishers, San Mateo, CA, 1999.

[52] R. A. Jacobs, M. I. Jordan, S. J. Nowlan, and G. E. Hinton. Adaptive mixtures of local experts. *Neural Computation*, 3:79–87, 1991.

[53] A. K. Jain and R. C. Dubes. *Algorithms for Clustering Data.* Prentice Hall, Englewood Cliffs, New Jersey, 1988.

[54] E. T. Jaynes. Where do we stand on maximum entropy. In R.D. Levine and M. Tribus, editors, *The Maximum Entropy Formalism*, pages 15–117. MIT Press, Cambridge, MA, 1979.

[55] S. Kaski. Convergence of a stochastic semisupervised clustering algorithm. Technical Report A62, Helsinki University of Technology, Publications in Computer and Information Science, Espoo, Finland, 2000.

[56] S. Kaski and J. Sinkkonen. Discriminative clustering: Vector quantization in learning metrics. In *Studies in Classification, Data Analysis, and Knowledge Organization. Proceedings of 26th Annual Conference of the Gesellschaft für Klassifikation (GfKl) July 22-24, 2002, University of Mannheim, Germany.* Springer, 2004. To appear.

[57] R. E. Kass and P. W. Vos. *Geometrical Foundations of Asymptotic Inference.* Wiley, New York, 1997.

[58] J. Kay. Feature discovery under contextual supervision using mutual information. In *Proceedings of IJCNN'92, International Joint Conference on Neural Networks*, pages 79–84. IEEE, Piscataway, NJ, 1992.

[59] K. Kiviluoto. Predicting bankruptcies with the self-organizing map. *Neurocomputing*, 21(1–3):191–201, 1998.

[60] A. Klami. Regularized discriminative clustering. Master's thesis, Helsinki University of Technology, Espoo, Finland, 2003.

[61] T. Kohonen. Self-organized formation of topologically correct feature maps. *Biological Cybernetics*, 43:59–69, 1982.

[62] T. Kohonen. *Self-organization and associative memory.* Springer, Berlin, 3rd edition, 1989.

[63] T. Kohonen. *Self-Organizing Maps.* Springer, Berlin, 3rd edition, 2001.

[64] P. Kontkanen, J. Lahtinen, P. Myllymäki, T. Silander, and H. Tirri. Supervised model-based visualization of high-dimensional data. *Intelligent Data Analysis*, 4:213–227, 2000.

[65] P. Kontkanen, J. Lahtinen, P. Myllymäki, and H. Tirri. Unsupervised Bayesian visualization of high-dimensional data. In R. Ramakrishnan, S. Stolfo, R. Bayardo, and I. Parsa, editors, *Proceedings of the Sicth International Conference on Knowledge Discovery and Data Mining (KDD-2000)*, pages 325–329. ACM, New York, 2000.

[66] S. Kullback. *Information Theory and Statistics.* Wiley, New York, 1959.

[67] T. H. Leahey. *A History of Psychology: Main Currents in Psychological Thought.* Prentice Hall, 5th edition, 1999.

[68] M. K. Murray and J. W. Rice. *Differential Geometry and Statistics.* Chapman & Hall, London, 1993.

[69] A. Y. Ng and M. I. Jordan. On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes. In T. G. Dietterich, S. Becker, and Z. Ghahramani, editors, *Advances in Neural Information Processing Systems 14*. MIT Press, Cambridge, MA, 2002.

[70] K. Nigam, A. K. McCallum, S. Thrun, and T. M. Mitchell. Text classification from labeled and unlabeled documents using EM. *Machine Learning*, 39(2-3):103–134, 2000.

[71] NIST/SEMATECH e-Handbook of statistical methods.
`http://www.itl.nist.gov/div898/handbook/`, 2003.

[72] M. Oja, J. Nikkilä, P. Törönen, E. Castrén, and S. Kaski. Learning metrics for visualizing gene functional similarities. In P. Ala-Siuru and S. Kaski, editors, *STeP 2002 — Intelligence, The Art of Natural and Artificial. The 10th Finnish Artificial*

Intelligence Conference, pages 31–40. Finnish Artificial Intelligence Society, 2002. Oulu, Finland.

[73] J. Peltonen, A. Klami, and S. Kaski. Learning more accurate metrics for self-organizing maps. In J. R. Dorronsoro, editor, *Artificial Neural Networks—ICANN 2002*, pages 999–1004. Springer, Berlin, 2002.

[74] F. Pereira, N. Tishby, and L. Lee. Distributional clustering of English words. In *Proceedings of the 30th Annual Meeting of the Association for Computational Linguistics*, pages 183–190. ACL, Columbus, OH, 1993.

[75] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling. *Numerical Recipes: The Art of Scientific Computing.* Cambridge University Press, Cambridge, UK, 2nd edition, 1992.

[76] J. C. Principe, J. W. Fisher III, and D.X. Xu. Information-theoretic learning. In S. Haykin, editor, *Unsupervised Adaptive Filtering.* Wiley, New York, 2000.

[77] C. R. Rao. Information and the accuracy attainable in the estimation of statistical parameters. *Bull. Calcutta Math. Soc.*, 37:81–91, 1945.

[78] C. R. Rao. *Linear Statistical Inference and its Applications.* Wiley, 2nd edition, 1973. Originally published 1965.

[79] B. D. Ripley. *Pattern Recognition and Neural Networks.* Cambridge University Press, Cambridge, UK, 1996.

[80] V. Roth and V. Steinhage. Nonlinear discriminant analysis using kernel functions. In S. A. Solla, T. K. Leen, and K.-R. Müller, editors, *Advances in Neural Information Processing Systems 12*, pages 568–574. MIT Press, Cambridge, MA, 2000.

[81] Y. D. Rubinstein and T. Hastie. Discriminative vs informative learning. In D. Heckerman, H. Mannila, D. Pregibon, and R. Uthurusamy, editors, *Proc. ACM KDD*, pages 49–53. AAAI Press, 1997.

[82] J. Salojärvi, S. Kaski, and J. Sinkkonen. Discriminative clustering in Fisher metrics. In *Proceedings of ICANN/ICONIP*. 2003. To appear.

[83] G. Salton and M. J. McGill. *Introduction to modern information retrieval.* McGraw-Hill, New York, 1983.

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

[85] N. Slonim, N. Friedman, and N. Tishby. Unsupervised document classification using sequential information maximization. In *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 129–136. ACM Press, 2002.

[86] N. Slonim, R. Somerville, N. Tishby, and O. Lahav. Objective classification of galaxy spectra using the information bottleneck method. *Monthly Notes of the Royal Astronomical Society*, 323:270–284, 2001.

[87] N. Slonim and N. Tishby. Agglomerative information bottleneck. In S. A. Solla, T. K. Leen, and K.-R. Müller, editors, *Advances in Neural Information Processing Systems 12*, pages 617–623. MIT Press, Cambridge, MA, 2000.

[88] N. Slonim and N. Tishby. Document clustering using word clusters via the information bottleneck method. In *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 208–215. ACM Press, 2000.

[89] N. Slonim and Y. Weiss. Maximum likelihood and the information bottleneck. In *Advances in Neural Information Processing Systems 14*. MIT Press, Cambridge, MA, 2002.

[90] M. Szummer and T. Jaakkola. Partially labeled classification with Markov random walks. In *Advances in Neural Information Processing Systems 14*. MIT Press, Cambridge, MA, 2002.

[91] C. W. Therrien. *Decision, estimation and classification.* Wiley, New York, 1989.

[92] N. H. Timm. *Applied Multivariate Analysis.* Springer, New York, 2002.

[93] M. E. Tipping. Deriving cluster analytic distance functions from Gaussian mixture models. In *Proceedings of ICANN99, Ninth International Conference on Artificial Neural Networks*, pages 815–820. IEE, London, 1999.

[94] N. Tishby, F. C. Pereira, and W. Bialek. The information bottleneck method. In *37th Annual Allerton Conference on Communication, Control, and Computing*, pages 368–377. Urbana, Illinois, 1999.

[95] N. Tishby and N. Slonim. Data clustering by Markovian relaxation and the information bottleneck method. In T. Leen, T. Dietterich, and V. Tresp, editors, *Advances in Neural Information Processing Systems 13*, pages 640–646. MIT Press, Cambridge, MA, 2001.

[96] K. Torkkola. Learning discriminative feature transforms to low dimensions in low dimentions. In T. G. Dietterich, S. Becker, and Z. Ghahramani, editors, *Advances in Neural Information Processing Systems 14*. MIT Press, Cambridge, MA, 2002.

[97] K. Torkkola. Nonlinear feature transforms using maximum mutual information. In *Proceedings of IJCNN'01, International Joint Conference on Neural Networks*, pages 2756–2761. IEEE, Piscataway, NJ, 2001.

[98] K. Torkkola and W. Campbell. Mutual information in learning feature transformations. In *Proceedings of the 17th International Conference on Machine Learning*, pages 1015–1022. Morgan Kaufmann, Stanford, CA, 2000.

[99] J. Tukey. *Exploratory data analysis.* Addison-Wesley, 1977.

[100] J. Vesanto. Data exploration process based on the self-organizing map. *Acta Polytechnica Scandinavica, Mathematics, Computing and Management in Engineering Series No. 115*, 2002. D.Sc.(Tech) Thesis, Helsinki University of Technology, Finland.

[101] E. P. Xing, A. Y. Ng, M. I. Jordan, and S. Russell. Distance metric learning, with application to clustering with side information. In *Advances in Neural Information Processing Systems 14*. MIT Press, 2002.