Chapter 11

Learning metrics

Samuel Kaski, Janne Sinkkonen, Jaakko Peltonen, Jarkko Venna, Arto Klami, Jarkko Salojärvi

11.1 Introduction

Unsupervised learning such as clustering and information visualization suffers from the garbage in—garbage out problem. The ultimate goal is to make discoveries in data, that is, to find new things without speficying them in advance. The problem is that unsupervised learning cannot distinguish relevant variation from irrelevant variation in data. Structured noise becomes modeled as well as relevant structure.

Hence, all successful unsupervised learning must have been supervised implicitly or explicitly, by feature extraction or model selection. Our goal is to automate (part of) this implicit supervision by learning from a supervising signal. The difference from standard supervised learning is that the goal is to explore new things in the primary data given the supervision, whereas in supervised learning the goal is simply to predict the supervisory signal. The task could be coined supervised unsupervised learning.

Sample applications include exploration of factors leading to bankruptcy, where primary data are financial indicators and supervisory signal is the bankruptcy risk. Another is exploration of gene expression, supervised by functional classes of the genes.

For methods that are based on distance computations, the supervision can be conveniently incorporated in the distance measure. The idea of deriving information-geometric metrics to data spaces from paired data has been coined the learning metrics principle. It is assumed that variation of the primary data $\mathbf{x} \in \mathbb{R}^n$ is important only to the extent it causes variation in *auxiliary data* c, the supervisory signal, which is available paired to the primary data.

In other words, important variation in $\mathbf{x}$ is supposed to be revealed locally by variation in the conditional density $p(c|\mathbf{x})$. The distance d between two close-by data points $\mathbf{x}$ and $\mathbf{x} + d\mathbf{x}$ is defined as the difference between the corresponding distributions of c, measured by the Kullback-Leibler divergence D_{KL}, i.e.

$$d_L^2(\mathbf{x}, \mathbf{x} + d\mathbf{x}) \equiv D_{\mathrm{KL}}(p(c|\mathbf{x}) \| p(c|\mathbf{x} + d\mathbf{x})) = \; d\mathbf{x}^T \mathbf{J}(\mathbf{x}) d\mathbf{x} \; , \tag{11.1}$$

where $\mathbf{J}(\mathbf{x})$ is the Fisher information matrix. The Riemannian metric depends on $\mathbf{x}$ and hence is more general than a global scaling of the feature space.

The Fisher information matrix has earlier been used to construct metrics to spaces of probability models (see, e.g., [1]). The novelty here is that the information matrix is applied in the data space to construct a new metric there. The coordinates of data are considered as parameters

In practice, the idea can be applied in two ways. One can estimate $p(c|\mathbf{x})$ first and then plug the new metric, computed from the estimates, into a standard unsupervised method. Another possibility is to more directly insert the new metric into the cost function of a suitable method. Examples of these approaches are discussed in more detail below.

11.2 Learning metrics for information visualization

Explicit estimation of learning metrics by approximations to (11.1) is generally applicable to explicitly supervise unsupervised metric-based methods. The choice of auxiliary data determines what is important, without need for hand-tuned feature extraction.

We have so far applied learning metrics to two widely used unsupervised information visualization methods: the Self-Organizing Map and Sammon's mapping, a sample Multidimensional Scaling (MDS) method.

Computation of approximations to the metric

Globally the learning metric (11.1) becomes minimal path integrals of local distances. The local distances in turn are based on conditional auxiliary densities $p(c|\mathbf{x})$. For practical computation, the densities must be estimated and the minimal path integrals approximated. We have developed several approximations; the choice needs a tradeoff between computation time and accuracy.

Several semiparametric estimators of the conditional density $p(c|\mathbf{x})$ are available. The still open theoretical question is how to choose the estimator rigorously.

The simple approximation for the distance between two points $\mathbf{x}_1$ and $\mathbf{x}_2$ is the local quadratic form [2]

$$d_1^2(\mathbf{x}_1, \mathbf{x}_2) = (\mathbf{x}_2 - \mathbf{x}_1)^T \mathbf{J}(\mathbf{x}_1)(\mathbf{x}_2 - \mathbf{x}_1) \tag{11.2}$$

called the 1-point approximation. An improved version called the T-point approximation [3] computes the metric at T points between the start and end point, yielding

$$d_T(\mathbf{x}_1, \mathbf{x}_2) = \frac{1}{T} \sum_{i=0}^{T-1} \left(\mathbf{r}^T \mathbf{J} \left(\mathbf{x} + \frac{i}{T} \mathbf{r} \right) \mathbf{r} \right)^{1/2} , \tag{11.3}$$

where $\mathbf{r} = \mathbf{x}_2 - \mathbf{x}_1$.

Both approximations assume the minimal path is a line. A further improvement is to form a graph whose edge weights are pairwise T-point distances between data points and perform a graph search for the minimal path [4]. This is called the graph approximation; it allows both linear and piecewise linear paths. Since data points are used as graph vertices, distances are computed more accurately where the data is dense.

Information visualization methods

The sequential SOM algorithm iterates *winner selection* and *adaptation*. In the learning metric the winner is sought by

$$w(\mathbf{x}(t)) = \arg\min_i d_L^2(\mathbf{x}(t), \mathbf{m}_i(t)) . \tag{11.4}$$

where t is the iteration, $\mathbf{x}(t)$ is the input and d_L can be either the local distance approximation d_1 or the T-point approximation d_T. The latter is more accurate, but computationally heavier.

For the local approximation the adaptation step can be shown to equal the familiar SOM learning rule,

$$\mathbf{m}_i(t+1) = \mathbf{m}_i(t) + \alpha(t) h_{wi}(t)(\mathbf{x}(t) - \mathbf{m}_i(t)) , \tag{11.5}$$

were $\alpha(t)$ is the learning rate and $h_{wi}(t)$ is the neighbourhood function.

In empirical tests the SOM-L with the improved (T-point) distance approximation significantly outperforms the 1-point SOM-L as well as classical SOM and a supervised SOM [3, 4].

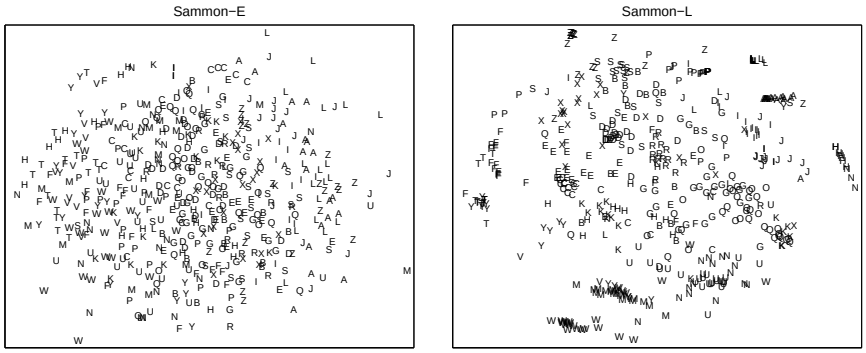

Figure 11.1: Sammon's mapping in learning metrics (right) separates the different letters of the Letter Recognition data (from UCI Machine Learning Repository) set clearly better than the Sammon's mapping in the Euclidean metric (left).

Metric multidimensional scaling methods (MDS) are used for visualizing similarities of data samples based on a pairwise distance matrix. They construct a low-dimensional representation for the data that aims to preserve the distance matrix.

Sammon's mapping, as well as the other MDS methods, are based on the pairwise distance matrix, are ideal candidates for the graph approximation since they are based on the pairwise distance matrix. The distances need to be computed only once.

The difference to the traditional Sammon's mapping where the pairwise distance matrix is computed in the Euclidean metric is illustrated in Figure 11.1 [4]. The class separation is clearly increased when the learning metrics is used, but the topology of the samples is still retained.

11.3 Discriminative Clustering (DC)

The original motivation for discriminative clustering was its asymptotic equivalence to vector quantization in learning metrics. DC turned out to have other interesting interpretations as well: It extends earlier works on mutual information maximization (IMAX [5], Information Bottleneck [6]) and connects learning metrics to generative models and contingency tables.

DC partitions a vectorial data space to a set of connected partitions that are homogeneous by distributions of an auxiliary variable or variables present in the data [7]. The homogeneity criterion of partitions turns out to be equivalent to informativeness of the partitions of the auxiliary variable(s). Membership of a sample in a partition then tends to predict the value of the auxiliary variable well, and vice versa. Still, the partitions are solely defined in terms of the primary data, without reference to the values of auxiliary data. Hence future data without the associated auxiliary variable can be partitioned. The relative locality of the clusters in the primary data space makes them useful for exploratory analysis.

A prototypical application would be segmenting customers of a company in terms of background information, but by using buying behaviour as the criterion of segment homogeneity. Buying behaviour guides the segmentation but does not directly define the segments. Incoming customers without buying history can then be immediately assigned to the predefined segments. Other applications include, e.g., understanding company bankruptcy, finding relationships between gene expression databases (Section 10.3), and guiding text document clustering with classifications of informaticians.

For densities. The original formulation for DC is for probability densities $p(c, \mathbf{x})$ of auxiliary data c and primary data $\mathbf{x}$. This version is easy to understand, but directly applicable only for large data/cluster rations.

Partitions of the primary data are restricted to be Voronoi regions, which makes them connected, relatively local, and therefore easy to interpret. Homogeneity of the auxiliary data distributions within the clusters is measured by the intra-cluster Kullback-Leibler divergence

$$E = \int D_{\mathrm{KL}}(p(c|\mathbf{x}) \| \boldsymbol{\psi}_{j(\mathbf{x})})\, p(\mathbf{x})\, d\mathbf{x} \,, \tag{11.6}$$

which is minimized with respect to the distributions of auxiliary data within clusters, $p(c|\mathit{Cluster}_j) \equiv \boldsymbol{\psi}_j$ and the Voronoi partitioning defined by the centroid parameters $\mathbf{m}_j$ (implicit in assignments $j(\mathbf{x})$). Minimizing the distortion is equivalent to maximizing the informativeness of the clusters about the values of the auxiliary variable, in the sense of mutual information. Gradient algorithms can be applied if the partitions are first smoothed. An extremely simple on-line learning rule results.

For data sets. The log-likelihood of a piece-wise constant model for the conditional densities $p(c|\mathbf{x})$ approaches the distortion (11.6) when the size of the data set grows. It is therefore a good candidate for the cost function of DC for finite data sets [8]. From the viewpoint of clustering, the distributional prototypes $\boldsymbol{\psi}$ are not interesting and can then be marginalized out, which leads to a likelihood only depending on the Voronoi partitioning $\{\mathbf{m}_j\}$:

$$L_{\mathrm{DC}}(\{\mathbf{m}_j\}) \propto \sum_{ij} \log \Gamma(n_i^0 + n_{ji}) - \sum_j \log \Gamma(N^0 + N_j) \,, \tag{11.7}$$

where n_{ji} denotes the number of samples in the cluster j with the value of auxiliary variable $c = i$. The parameters n_i^0 arise from a Dirichlet prior, and $N_j = \sum_i n_{ji}$, $N^0 = \sum_i n_i^0$.

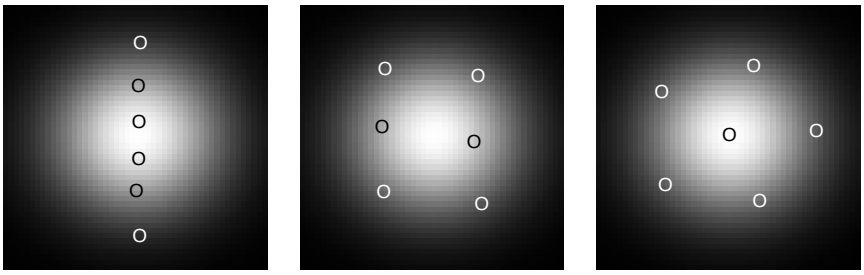

Figure 11.2: Discriminative clustering of simple toy data, where only the vertical direction is indicated to be relevant by auxiliary data associated to the 2D primary data. The primary data is sampled from the Gaussian distribution (grey shades), while the conditional distribution of the auxiliary data changes in the vertical direction. The regularized solution (middle) shares properties of the discriminative solution and the K-means solution (right).

After the partitions are smoothed the new cost function can be optimized by gradient algorithms. Direct optimization by simulated annealing is also possible, but a simple conjugate gradient algorithm with smoothed partitions leads to equally good results and is faster. In empirical comparisons the marginaled finite-data model has been found to outperform the simple on-line algorithm resulting from (11.6).

Regularization. Tests indicate that the performance of the purely discriminative DC algorithms is improved if the cost functions are 'regularized' by partially taking into account the margin distribution $p(\mathbf{x})$ in one way or another (Fig. 11.2). Note that taking it fully into account would lead to modeling the joint distribution $p(c, \mathbf{x})$, which is different by its goal and empirically shown to be inferior in the task of DC.

Non-Euclidean spaces. DC has been extended for data on hyperspheres and on distributional spaces. The latter formulation is applicable to text documents under the usual 'bags of words' assumption, where word frequencies are analyzed and the order of words in the documents is ignored. The method has been applied to scientific texts from the INSPEC database [9], by using keywords chosen by the document authors as auxiliary data. Keywords improve feature selection in the full documents and therefore improve clustering results compared to classic methods.

Connection to learning metrics. For a large number of clusters, DC performs vector quantization in learning metrics (Section 11.1): The Euclidean distortion of normal vector quantization becomes replaced with a distortion computed in the Fisher metric (11.1). The Fisher metric measures changes in the conditional distributions $p(c|\mathbf{x})$ of the auxiliary variable [10].

The asymptotic connection was utilized in practice by plugging a local approximation of Fisher metrics to standard K-means clustering [11]. The adaptable metric frees the Voronoi partitions from being defined in Euclidean metric and allows more optimal shapes. In tests the resulting algorithms, although computationally heavy, have outperformed the plain DC.

Figure 11.3: The Helsinki capital area segmented into Voronoi regions maximally informative of demographics. Associative clustering of geographic coordinates and vectorial sosiodemographic data finds segments for both 'margin spaces'. In the figure, only one margin space, the geography, is shown. Demographically distinct and homogeneous regions such as the downtown become clearly separated. Similar clusters become defined to the high-dimensional sosiodemographic space.

Associative clustering: bidirectional DC

Discriminative clustering quantizes a continuous variable and then maximizes statistical dependency between two discrete variables: the partitions and the auxiliary variable guiding the partitioning. Contingency tables are a classic framework for quantifying and testing such dependencies. In this framework, the cost (11.7) is interpretable as a *Bayes factor* between the hypotheses of dependent and independent margins [12].

In DC one margin is fixed. We have called the generalization to two adjustable margins *associative clustering* (AC; [13]). Then two vectorial variables are quantized by Voronoi partitionings, and the partitionings are adjusted to maximize their mutual dependency in the sense of the Bayes factor. Techniques similar to discriminative clustering can be applied, including the regularization methods and smoothed partitions. A demonstration of AC is shown in Figure 11.3.

11.4 Discriminative components

Unsupervised principal components and factor analyses search for components of data that can be used for data exploration, visualization and dimensionality reduction.

A classical method for supervising the components is linear discriminant analysis (LDA). It has been commonly used for two tasks: the more common one is linear classification (supervised learning), but the components can also be used for exploring and visualizing class differences. We have generalized LDA for this latter purpose, but searching for linear components that are more generally *informative of* or *relevant to* the the classes of samples. The task of extracting components relevant to auxiliary data could perhaps be called Relevant Component Analysis.

We search for linear relevant components [14] by optimizing the linear projection $\mathbf{y} = \mathbf{f}(\mathbf{x}) = \mathbf{W}^T\mathbf{x}$, where the columns of $\mathbf{W}$ are the component directions. The criterion is simply maximization of the log-likelihood of the auxiliary data given the projection, i.e.,

$$L = \sum_{(\mathbf{x},c)} \log \hat{p}(c|\mathbf{f}(\mathbf{x})) \tag{11.8}$$

where c are the auxiliary data and $\hat{p}$ is an estimator computed after the projection.

The key point in this method is its simplicity. The likelihood is a well-defined, simple criterion for fitting a projection to finite data, yet it has interesting theoretical connections and works better than alternative methods in practice. Maximizing the likelihood is asymptotically equivalent to maximizing the mutual information $I(C, \mathbf{f}(X))$ when consistent estimators $\hat{p}$ are used. Moreover, maximizing the likelihood is asymptotically approximately equivalent to minimizing a reconstruction error in learning metrics under some assumptions, so the components can be considered principal components in learning metrics.

The method has empirically outperformed classical and recent [16] methods. It has been applied to bioinformatics (Chapter 10) and assessing convergence of MCMC simulations (below).

11.5 Visualization of posterior distributions

Probabilistic generative modeling is one of the theoretical foundations of current mainstream machine learning and data analysis. Bayesian inference is potentially very powerful but closed-form solutions are seldom available. Inference has to be based on either approximation methods or simulations with Markov Chain Monte Carlo (MCMC) sampling.

The main practical problem of MCMC is how to assess whether the simulation has converged. The resulting samples come from the true distribution only after convergence. It turns out [17] that the main multivariate convergence measure, the multivariate potential scale reduction factor (MPSRF) developed by Brooks and Gelman [18], equals the cost function of a one-dimensional linear discriminant analysis (LDA), a method that discriminates between data classes.

MCMC chains have traditionally been visualized by time series plots, marginal histograms or 2-dimensional scatter plots of two variables. The problem with these visualizations is that they do not scale up to large models with lots of parameters. As the cost function of LDA is the equivalent to the MPSRF measure, we can use LDA to reduce the number of visualizations. A scatter plot of a projection on the two best discriminative components (see Figure 11.4) is the single best two-dimensional image in the sense of the MPSRF measure.

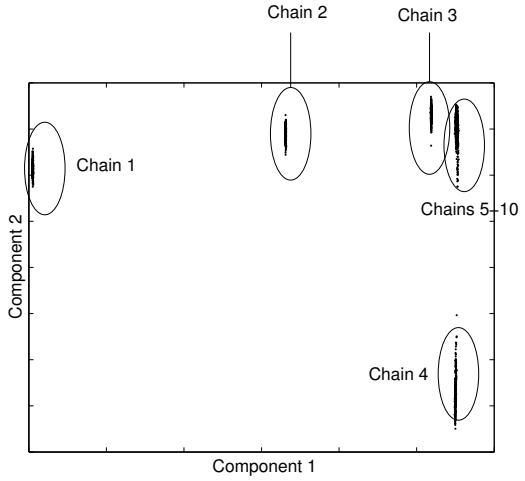

Figure 11.4: Two-dimensional LDA projection of samples from a MCMC simulation that does not converge. Chains 1-4 have gotten stuck in a degenerate state. The ellipses have been drawn by hand to mark the chains.

LDA assumes that each class is normally distributed with the same covariance matrix in each class. This does not hold in general, in particular not before MCMC convergence for small data. To address the above problem, we suggest to complement LDA-based analysis with the generalization of LDA introduced in Section 11.4.

Sometimes we are interested in visualizing the posterior distribution for other reasons than studying convergence of a sampler. We might for example be interested how the parameters affect the model output. Toward this end, we have proposed [19] a method that uses the Fisher metric of the model with a non-linear projection method, to create visualizations of the posterior that reflect the effect parameters have on the output.

References

[1] Shun-ichi Amari and Hiroshi Nagaoka. *Methods of Information Geometry*, volume 191. American Mathematical Society and Oxford University Press, 2000.

[2] Samuel Kaski, Janne Sinkkonen, and Jaakko Peltonen. Bankruptcy analysis with self-organizing maps in learning metrics. *IEEE Transactions on Neural Networks*, 12:936–947, 2001.

[3] Jaakko Peltonen, Arto Klami, and Samuel Kaski. Learning more accurate metrics for self-organizing maps. In José R. Dorronsoro, editor, *Artificial Neural Networks—ICANN 2002*, pages 999–1004. Springer, Berlin, 2002.

[4] Jaakko Peltonen, Arto Klami, and Samuel Kaski. Learning metrics for information visualization. In *Proceedings of the Workshop on Self-Organizing Maps (WSOM'03)*, pages 213–218. Hibikino, Kitakyushy, Japan, September 2003.

[5] Suzanna Becker. Mutual information maximization: models of cortical self-organization. *Network: Computation in Neural Systems*, 7:7–31, 1996.

[6] Naftali Tishby, Fernando C. Pereira, and William Bialek. The information bottleneck method. In *37th Annual Allerton Conference on Communication, Control, and Computing*, pages 368–377. Urbana, Illinois, 1999.

[7] Janne Sinkkonen and Samuel Kaski. Clustering based on conditional distributions in an auxiliary space. *Neural Computation*, 14:217–239, 2002.

[8] Janne Sinkkonen, Samuel Kaski, and Janne Nikkilä. 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*, pages 418–430. Springer, Berlin, 2002.

[9] Jaakko Peltonen, Janne Sinkkonen, and Samuel Kaski. Discriminative clustering of text documents. In Lipo Wang, Jagath C. Rajapakse, Kunihiko Fukushima, Soo-Young Lee, and Xin Yao, editors, *Proceedings of ICONIP'02, 9th International Conference on Neural Information Processing*, pages 1956–1960. IEEE, Piscataway, NJ, 2002.

[10] Samuel Kaski and Janne Sinkkonen. Principle of learning metrics for 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*, accepted for publication.

[11] Jarkko Salojärvi, Samuel Kaski, and Janne Sinkkonen. Discriminative clustering in Fisher metrics. In O. Kaynak, E. Alpaydin, E. Oja, and L. Xu, editors, *Artificial Neural Networks and Neural Information Processing - Supplementary proceedings ICANN/ICONIP 2003*, pages 161–164. Istanbul, Turkey, June 2003. To appear.

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

[13] J. Sinkkonen, J. Nikkilä, L. Lahti, and S. Kaski. Associative clustering by maximizing a bayes factor. Technical Report A68, Helsinki University of Technology, Laboratory of Computer and Information Science, Espoo, Finland, 2003.

[14] Samuel Kaski and Jaakko Peltonen. Informative discriminant analysis. In *Proceedings of the Twentieth International Conference on Machine Learning (ICML-2003)*, pages 329–336. AAAI Press, Menlo Park, CA, 2003.

[15] Kari Torkkola and William 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.

[16] Kari Torkkola. Feature extraction by non-parametric mutual information maximization. *Journal of Machine Learning Research*, 3:1415–1438, 2003.

[17] Jarkko Venna, Samuel Kaski, and Jaakko Peltonen. Visualizations for assessing convergence and mixing of mcmc. In N. Lavrac, D. Gamberger, H. Blockeel, and L. Todorovsk, editors, *Proceedings of the 14th European Conference on Machine Learning (ECML 2003)*, pages 432–443, Berlin, 2003. Springer.

[18] Stephen Brooks and Andrew Gelman. General methods for monitoring convergence of iterative simulations. *Journal of Computational and Graphical Statistics*, 7(4):434–456, Dec 1998.

[19] Jarkko Venna and Samuel Kaski. Visualizing high-dimensional posterior distributions in bayesian modeling. In O. Kaynak, E. Alpaydin, E. Oja, and L. Xu, editors, *Artificial Neural Networks and Neural Information Processing - Supplementary proceedings ICANN/ICONIP 2003*, pages 165–168, Istanbul, Turkey, June 2003.