Chapter 6

Dependency exploration and learning metrics

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

6.1 Introduction

We develop statistical machine learning methods for extracting useful regularities from large, high-dimensional data sets. We have divided this task of *statistical data mining* or exploration into three subtypes of problems:

- *Unsupervised mining*, where regularities or dependencies *within* one data set are sought. Standard clustering, component models, and data mining belongs to this category. We have recently developed new nonlinear projection methods for information visualization.

- *Supervised mining*, where one data set supervises the mining of another. We have introduced a principle of learning metrics, where the distance metric is learned in a supervised way, to guide subsequent unsupervised learning. Hence, this line of work has alternatively been coined *supervised unsupervised learning*.

- *Dependency mining* or exploration, where the supervision is symmetric and the task is to find dependencies *between* data sets. In this subtask we have introduced new clustering and component models.

Unsupervised mining is the most common task of these three but also the most difficult, because it suffers from the garbage in—garbage out problem; unsupervised learning cannot distinguish relevant from irrelevant variation in a data set. Supervised mining and dependency mining are two solutions to this problem that are applicable in different kinds of settings. In supervised mining, variation within one of the sets is assumed irrelevant and the other set is assumed relevant enough to be useful for supervising the other. An example is analysis of measurement data of cancer tissues where known cancer labels (the other set) are clearly extremely relevant, and variation in tissue samples not related to cancer classes is irrelevant.

In dependency mining the within-set variation is assumed irrelevant in all data sets, and only between-set variation is important. An example is measurement of several noisy signals from a common source, when characteristics of the noise are not known. More examples are given in Section 5.

6.2 Supervised unsupervised learning

Many unsupervised methods rely on a distance measure that tells how far apart two data samples are. Usually the measure is a simple one such as Euclidean distance. However, such measures do not take into account that two samples can differ in many ways, and not all differences are relevant for the analysis. How relevant a difference is depends on what the analyst is interested in; the relevance can vary in different parts of the data space.

In supervised settings we can learn what is relevant by learning a distance measure, that is, a metric. The idea of learning such metrics has been coined the *learning metrics principle* [1].

We assume there is paired data: the primary data $\mathbf{x}$ that we want to explore are paired to *auxiliary data* c that guide the exploration. The learning metrics principle assumes that variation of the primary data is important only to the extent it causes variation in auxiliary data.

Technically, we use an information-geometric definition: 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.,

$$d_L^2(\mathbf{x}, \mathbf{x} + d\mathbf{x}) \equiv D_{\mathrm{KL}}(p(c|\mathbf{x}) \| p(c|\mathbf{x} + d\mathbf{x})) = \; d\mathbf{x}^T \mathbf{J}(\mathbf{x}) d\mathbf{x} \; , \tag{6.1}$$

where $\mathbf{J}(\mathbf{x})$ is the Fisher information matrix. The metric is learned from the data that defines the distributions $p(c|\mathbf{x})$. This yields a Riemannian metric that preserves the topology of the feature space, yet can flexibly change what is locally relevant. The preservation of topology and the capability to ignore noisy dimensions not related to the auxiliary data are demonstrated (with empirical experiments) in [2].

In practice, the learning metrics principle 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 both are given below.

Visualization and component models

The self-organizing map learns to visualize multidimensional data on a two-dimensional regular grid by finding and optimizing winner units for data samples. The winner is the closest map unit for a particular sample, and we simply choose the winner by distance in the learning metric. A practical approximation to the distance that in principle is defined by path integrals is the so-called T-point distance approximation [2] defined as

$$d_T(\mathbf{x}_1, \mathbf{x}_2) = \frac{1}{T} \sum_{i=0}^{T-1} \left(\mathbf{r}^T \mathbf{J} \left(\mathbf{x} + \frac{i}{T} \mathbf{r} \right) \mathbf{r} \right)^{1/2} \tag{6.2}$$

where $\mathbf{r} = \mathbf{x}_2 - \mathbf{x}_1$. This approximation assumes the shortest path is a line and computes the local metric at T points between the start and end point.

The SOM in learning metric has been shown shown to outperform SOM in Euclidean metric, as well as a supervised variant of SOM [2]. It provides visualizations that tell more about the auxiliary data, and on top of that allows visualizing how relevant the input features locally are. Example visualizations of SOMs computed using the learning metric are provided in Figure 6.1.

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

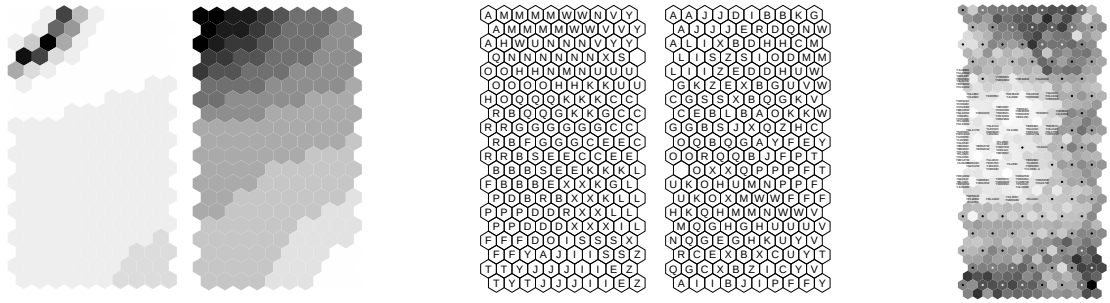

Figure 6.1: Example applications of SOM in learning metrics. **a** Analysis of financial data reveals that the importance of profitability (left) in avoiding bankruptcies depends on the state of the company, here illustrated by the actual profitability indicator (right). **b** Similar written characters are grouped more tighly in learning metrics (left) than in Euclidean metric (right). **c** Part of yeast gene expression data visualization emphasizing the functional categories.

representation for the data that aims to preserve the distance matrix, and are ideal candidates for applying computationally more intensive approximations of learning metrics distances since the distances need to be computed only once.

In this kind of application the assumption of minimal path being a line can be relaxed. Instead, a graph whose edge weights are pairwise T-point distances between data points is formed and a graph search for the minimal path is performed, providing a piece-wise linear path. In [2] this approximation generalized a specific MDS method called Sammon's mapping.

Relevant Component Analysis is a new data analysis tool that finds components of data in the learning metrics. Instead of unsupervised components like principal components, we wish to find components that contain information about, or are relevant for, classes of the data. Such components can be used to reduce dimensionality, in order to explore and visualize class separation, and to study the contribution of original data variables to it. We call the task of finding the components Relevant Component Analysis (RCA).

A classical method for this task is Linear Discriminant Analysis (LDA), which finds linear components but makes restrictive assumptions about the data distribution. We have introduced an improved method that removes the assumptions and finds components by optimizing a simple nonparametric generative model for class labels.

Technically, the method maximizes a simple likelihood criterion:

$$\sum_{(\mathbf{x},c)} \log \hat{p}(c|\mathbf{W}^T\mathbf{x}) \tag{6.3}$$

where the $\mathbf{x}$ are samples, c are their classes, $\hat{p}$ is the nonparametric estimator, and the columns of $\mathbf{W}$ are the component directions. This is equivalent (asymptotically, when $\hat{p}$ is consistent) to maximizing the mutual information of the component projections and the class labels. The components can be interpreted (asymptotically and approximately) as principal components in learning metrics.

The main merits of the new method, compared to other generalizations of LDA, are its theoretical simplicity and good performance. The method has been applied to exploration of sound samples from different phonemes [3], expressions of genes from different functional

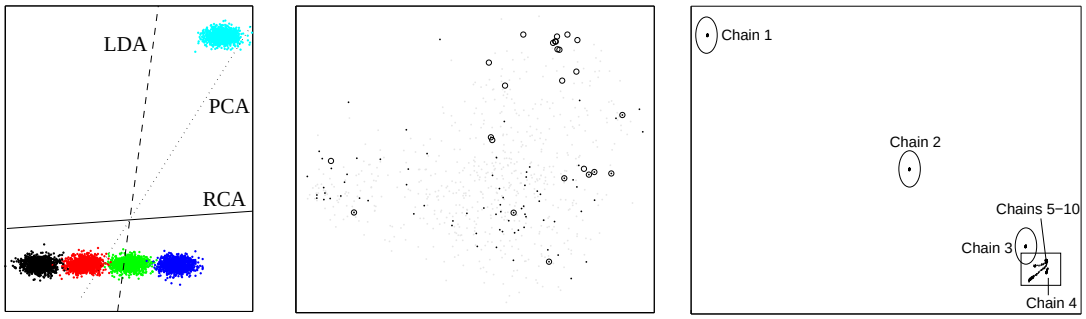

Figure 6.2: Left: The new method, RCA, finds a component that is more informative of classes than the classical methods. Center: two components of gene expression data that are informative of functional classes (only two classes shown). Right: two components of MCMC samples that show differences between sampling chains.

classes [4], and posterior samples from different MCMC chains (see below). Figure 6.2 shows examples on both toy data and real applications.

Visualizing convergence problems in MCMC simulation. 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 [5] that the main multivariate convergence measure, the multivariate potential scale reduction factor (MPSRF) developed by Brooks and Gelman [6], equals the cost function of a one-dimensional linear discriminant analysis (LDA), a method that discriminates between data classes. Traditional methods of visualizing MCMC simulations 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 focus on features that are relevant to convergence, and thus reduce the number of visualizations.

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 RCA [5].

Clustering

Discriminative clustering is a method for clustering continuous data so that the clusters become informative of auxiliary data paired with the data samples. While originally motivated by its asymptotic equivalence to vector quantization in learning metrics [7], the current algorithm is a probabilistic algorithm that maximizes the dependency of a contingency table and handles parameter uncertainty in a Bayesian manner [8]. It provides a link between learning metrics and generative modeling.

The key observation is that from the viewpoint of clustering, the parameters defining class distributions within clusters are not interesting and can then be marginalized out. This leads to a cost function only depending on the cluster prototype vectors $\{\mathbf{m}_j\}$:

$$L_{\mathrm{DC}}(\{\mathbf{m}_j\}) \propto \sum_{ij} \log\Gamma(n_i^0 + n_{ji}) - \sum_j \log\Gamma(N^0 + N_j) \;, \tag{6.4}$$

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 $N_j = \sum_i n_{ji}$, $N^0 = \sum_i n_i^0$.

In [8] it was shown that (6.4) can be efficiently optimized by a conjugate gradient algorithm operating on a slightly modified cost function (the counts are smoothed). It gave performance comparable to directly optimizing the original cost function with a time-consuming simulated annealing algorithm. On top of that, the model can be regularized by two alternative ways, either by an information theoretic equalization of cluster sizes, or by a Bayesian way of modeling also the primary data to improve generalization. The latter provides an interesting compromise between modeling conditional and joint densities, and the experimental results show that with small training data sets including a term modeling the covariates improves the accuracy.

Finite-data sequential information bottleneck Count data such as frequencies of words in text documents can be represented as a table of co-occurrence counts, called a contingency table. For clustering such data, successful Information Bottleneck-based methods treat the table as a probability distribution. However, in practice the table is sparsely populated by the finite number of counts, and it is necessary to take the sampling uncertainty into account.

We have introduced a new rigorous method for this. It is a variant of the sequential Information Bottleneck algorithm [9], with a new cost function directly defined for counts. The new cost function, a Bayes factor, compares the posterior probabilities of two alternative probabilistic models for the contingency table. It turns out that we can integrate over model parameters which takes the finite co-occurrence numbers into account. On the other hand, if there is much data, the cost function becomes equivalent to that of the sequential Information Bottleneck.

The new formulation extends discriminative clustering, defined earlier for continuous data, to count data, for which there exist powerful optimization algorithms. The new method, finite sequential Information Bottleneck (fsIB; [10]) outperformed the previous sequential Information Bottleneck in clustering sparse subsets of document corpora, as measured by a precision measure with respect to known categories of the documents.

References

[1] Samuel Kaski and Janne Sinkkonen. Principle of learning metrics for exploratory data analysis. *Journal of VLSI Signal Processing, special issue on Machine Learning for Signal Processing*, 37:177–188, 2004.

[2] Jaakko Peltonen, Arto Klami, and Samuel Kaski. Improved learning of Riemannian metrics for exploratory analysis. *Neural Networks*, 17:1087–1100, 2004.

[3] Jaakko Peltonen and Samuel Kaski. Discriminative components of data. *IEEE Transactions on Neural Networks*, 16(1):68–83, 2005.

[4] 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.

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

[6] 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.

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

[8] S. Kaski, J. Sinkkonen, and A. Klami. Discriminative clustering. *Neurocomputing*, 69:18–41, 2005.

[9] Noam Slonim, Nir Friedman, and Naftali 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, New York, NY, USA, 2002.

[10] Jaakko Peltonen, Janne Sinkkonen, and Samuel Kaski. Sequential information bottleneck for finite data. In Russ Greiner and Dale Schuurmans, editors, *Proceedings of the Twenty-First International Conference on Machine Learning (ICML 2004)*, pages 647–654. Omnipress, Madison, WI, 2004.

6.3 Dependency exploration

In dependency exploration we assume that the relevant aspects of data are shared by several information sources. This assumption opens up a new principled framework to combine various information sources. In particular, the effects due to data set-specific noise can be filtered out. Instead of having to specify a model for the noise, the data analyst needs to be able to choose a set of data sources.

The key idea in dependency exploration is to build models for two (or more) information sources such that their statistical dependency is maximized. Given two sets of real-valued vectorial features from two sources, $\mathbf{x} \in \mathbb{R}^n$ and $\mathbf{y} \in \mathbb{R}^m$, and two representations for them, $v = f_x(\mathbf{x})$ and $w = f_y(\mathbf{y})$, dependency exploration models maximize some estimate of the statistical dependency between v and w. One of the most popular estimators is mutual information $I(V, W) = \sum \sum p(v, w) \log \frac{p(v,w)}{p(v)p(w)}$. The main drawback of mutual information is that it is defined for (known) probability distributions, and may have strong biases when estimated from the finite data of practical applications. We have developed several methods that (i) use mutual information, or (ii) use a finite-data version of mutual information.

Associative clustering

Associative clustering (AC) [1] is a clustering method suitable for dependency exploration of two data sources. It clusters each data source separately, such that the dependency between the clusterings is maximized. If the different sources describe the same object, then the clusterings are as similar as possible in the sense of maximizing statistical dependency, but yet the objects may belong to different clusters in different contexts (sources) if necessary.

The dependencies between the two clusterings are represented with a two-way contingency table formed by cross-tabulation of the data items in the two sets of clusters. AC is an extension of discriminative clustering to two continuous margin spaces, and the techniques similar to discriminative clustering can be applied, including the regularization methods and smoothed partitions. The uncertainty in the clustering result can be dealt with by bootstrap. Figure 6.3 gives on overview of AC.

Generalized canonical correlation analysis

The classical canonical correlation analysis (CCA) finds maximally correlating components from two feature sets. It is equivalent to finding mutual information-maximizing components if the data is normally distributed. A generalized version of CCA, gCCA, extends the method to several feature sets [2].

We have developed a novel way to use the gCCA to integrate multiple data sets in such a way that their statistical dependencies are maximally preserved in the new, integrated representation [3]. As gCCA is computationally efficient it can be used as a preprocessing step for more complicated dependency analysis tasks, such as associative clustering. A sample application is given in Section 5.

Non-parametric dependent components

Canonical correlation analysis can also be extended to capture more general dependencies instead of mere correlation. In [4] we introduced a method coined *dependent component analysis* that estimates the mutual information (or more presicely, the likelihood ratio between dependent and independent hypotheses) using non-parametric density estimates

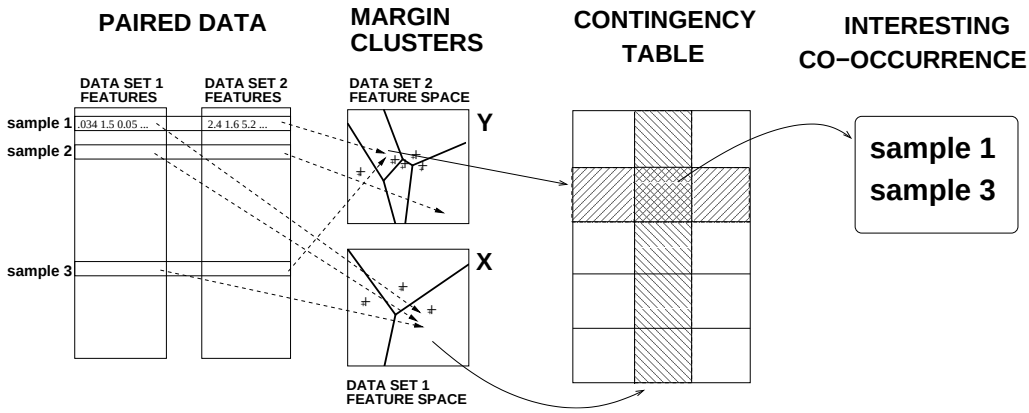

Figure 6.3: Associative clustering (AC) in a nutshell. Two data sets are clustered into margin clusters represented as Voronoi regions with prototype vectors. A one-to-one correspondence between the two data sets exists: each sample is presented in both sets. As each sample falls to one cluster in each data set, we get a contingency table by placing the two sets of clusters as rows and columns, and by counting samples in each combination of row and column clusters. *AC* by definition finds Voronoi prototypes that maximize the dependency seen in the contingency table. Maximization of dependency in a contingency table results in a maximal amount of counts not explainable by the margin distributions which can then be interpreted as maximally dependent clusters.

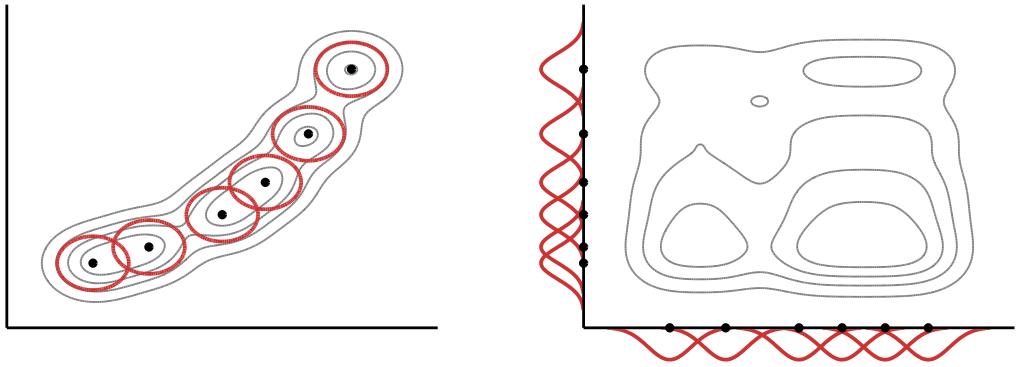

Figure 6.4: Dependency can be measured by studying how closely a joint distribution (left) can be approximated by a product of marginal distributions (right). For independent variables the distributions are identical. Here the gray density estimate in both pictures is computed using Gaussian mixtures, and the representations have relatively clear but non-linear dependency.

in the projection space, and maximizes the estimated dependency with respect to the parameters of the projections using a conjugate gradient algorithm.

The non-parametric measure for dependency is illustrated in Figure 6.4, where the two density estimates required for estimating the likelihood ratio are depicted. The closer the components are to being independent, the closer the two distributions are.

The algorithm was demonstrated to better find the correct component when used on toy data that had dependent non-Gaussian variables. It was also shown that in a real-life application to yeast stress measurements the the projection space found by the method gave more information about general stress than the space found by gCCA or KernelCCA described in [2].

References

[1] Samuel Kaski, Janne Nikkilä, Janne Sinkkonen, Leo Lahti, Juha Knuuttila, and Christophe Roos. Associative clustering for exploring dependencies between functional genomics data sets. *IEEE/ACM Transactions on Computational Biology and Bioinformatics, Special Issue on Machine Learning for Bioinformatics – Part 2*, 2(3):203–216, 2005.

[2] F. R. Bach and M. I. Jordan. Kernel independent component analysis. *Journal of Machine Learning Research*, 3:1–48, 2002.

[3] Janne Nikkilä, Christophe Roos, Eerika Savia, and Samuel Kaski. Explorative modeling of yeast stress response and its regulation with gcca and associative clustering. *International Journal of Neural Systems*, 15(4):237–246, 2005.

[4] Arto Klami and Samuel Kaski. Non-parametric dependent components. In *Proceedings of ICASSP'05, IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages V–209–V–212. IEEE, 2005.

6.4 Discriminative learning

The more traditional counterpart to supervised mining is *discriminative learning* where the data set is the same but the task is different. Given paired data $(\mathbf{x}, c)$, the task is to predict c for a test set where only the values of $\mathbf{x}$ are known.

There exist two traditional modeling approaches for predicting c, discriminative and generative. Discriminative models optimize the conditional probability $p(c|\mathbf{x})$ (or some other discriminative criterion) directly. The models are good classifiers, since they do not waste resources on modeling those properties of the data that do not affect the value of c, that is, the distribution of $\mathbf{x}$. The alternative approach is generative modeling of the joint distribution $p(c, \mathbf{x})$. Generative models add prior knowledge of the distribution of $\mathbf{x}$ into the task. This facilitates for example inferring missing values, since the model is assumed to generate also the covariates $\mathbf{x}$. The models are often additionally simpler to construct, and their parameters offer simple explanations in terms of expected sufficient statistics.

Discriminative Joint Density Models. One way of constructing discriminative classifiers is to take a joint density model, and then change the objective function from the joint likelihood $p(c, \mathbf{x}|\theta)$ to the conditional likelihood $p(c|\mathbf{x}, \theta)$. The obtained solution is (asymptotically) optimal for discrimination [1], given the model family. Compared to pure discriminative models, the benefit of the approach is that prior knowledge about $\mathbf{x}$ can be brought in. We call such a model a discriminative joint density model. The models operate in the same parameter space as ordinary discriminative models, but the generative formulation constrains the model manifold within the space.

Another advantage is that even after converting a joint density model to a discriminative model, the model still constructs a density estimate for $\mathbf{x}$. In [1] we show that this information may be useful, even if the model is inaccurate, for example in predicting missing values of $\mathbf{x}$.

Discriminative Expectation Maximization. Discriminative joint density models have been put to extensive use in speech processing applications, where good results have been obtained using discriminative hidden Markov models [2]. Current state-of-the-art method within the field used for optimizing the models is called extended Baum Welch, EBW [3, 2]. During the last 15 years, considerable effort has been made in order to find the best form of update formulas, but the method is still partly heuristic and has no solid theory explaining the update formulas. In [4] we introduce a discriminative Expectation Maximization -type algorithm that relies on a derivation of a global lower bound for conditional probability densities, alternatively to the practically too complex [5]. The derivation suggests a practical algorithm that results from slightly relaxing the requirement of the globality of the lower bound. The benefit of the algorithm is computational efficiency and simpler update formulas. The resulting update formulas are very close to current extended Baum Welch formulas, for which they give a theoretical basis that justifies the heuristics.

References

[1] Jarkko Salojärvi, Kai Puolamäki, and Samuel Kaski. On discriminative joint density modeling. In João Gama, Rui Camacho, Pavel Brazdil, Alípio Jorge, and Luís Torgo, editors, *Machine Learning: ECML 2005*, Lecture Notes in Artificial Intellligence 3720, pages 341–352, Berlin, Germany, 2005. Springer-Verlag.

[2] D. Povey, P.C. Woodland, and M.J.F. Gales. Discriminative MAP for acoustic model adaptation. In *IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. Proceedings. (ICASSP'03)*, volume 1, pages 312–315, 2003.

[3] P.S. Gopalakrishnan, Dimitri Kanevsky, Arthur Nádas, and David Nahamoo. An inequality for rational functions with applications to some statistical estimation problems. *IEEE Transactions on Information Theory*, 37(1):107–113, 1991.

[4] Jarkko Salojärvi, Kai Puolamäki, and Samuel Kaski. Expectation maximization algorithms for conditional likelihoods. In Luc De Raedt and Stefan Wrobel, editors, *Proceedings of the 22nd International Conference on Machine Learning (ICML-2005)*, pages 753–760, New York, USA, 2005. ACM press.

[5] Tony Jebara and Alex Pentland. On reversing Jensen's inequality. In Todd K. Leen, Thomas G. Dietterich, and Volker Tresp, editors, *Advances in Neural Information Processing Systems 13*, Cambridge, MA, April 2001. MIT Press.

6.5 Visualization methods

Visualization of mutual similarities of entries in large data sets is a central subproblem in exploratory analysis and mining. "Looking at the data" may be an invaluable sanity check, and often gives new insights on the data. We have introduced new nonlinear projection methods, and measures for better quantifying the necessary tradeoffs such methods need to make.

Continuity and trustworthiness

We have introduced methods to quantify a key question in visualization, namely the preservation of the original similarity relationships. In general, it is impossible to preserve all the similarities in the data set when projecting it to a lower-dimensional display. Hence, all visualization methods make a compromise between two goals. On the one hand the visualizations should be *trustworthy*, in the sense that samples that are near each other, i.e., in the same neighborhood, in the visualization can be trusted to actually be similar. On the other hand all the original similarities should become visualized. Some of the original similarities might be missing from the visualization because of *discontinuities* in the mapping. The tradeoff between these two goals is quite similar to the precision recall tradeoff in information retrieval.

We compared the trustworthiness and continuity of a set of state-of-the-art methods in a gene expression data visualization task [1, 2]. The Self-organizing map and another nonlinear dimensionality reduction method, Curvilinear Component Analysis, were found to be more trustworthy than other methods while visualizations produced by principal component analysis typically have a good continuity.

Local multidimensional scaling

It would be best if the user could select the tradeoff between trustworthiness and continuity explicitly instead of having to settle for the implicit tradeoff inherent in each method. This idea lead us to develop a new visualization method coined local MDS [3]. It extends a nonlinear dimensionality reduction method, curvilinear component analysis, with the ability to tune the tradeoff.

The cost function of local MDS is

$$\begin{aligned} E = & \frac{1}{2} \sum_i \sum_{j \neq i} [(1-\lambda)(d(\mathbf{x}_i, \mathbf{x}_j) - d(\mathbf{y}_i, \mathbf{y}_j))^2 F(d(\mathbf{y}_i, \mathbf{y}_j), \sigma_i) + \\ & + \lambda (d(\mathbf{x}_i, \mathbf{x}_j) - d(\mathbf{y}_i, \mathbf{y}_j))^2 F(d(\mathbf{x}_i, \mathbf{x}_j), \sigma_i)] \; . \end{aligned}$$

The first part of the cost function focuses on preserving distances that are within the area of influence $F(d(\mathbf{y}_i, \mathbf{y}_j), \sigma_i)$ around a data point i in the output space, and is the same as the original cost function of curvilinear component analysis. The second part focuses on preserving distances that are within the area of influence in the input space. The weighting between these two parts, and the tradeoff between trustworthiness and continuity, is controlled with the parameter λ. The effect of changing λ is illustrated in Figure 6.5.

References

[1] Samuel Kaski, Janne Nikkilä, Merja Oja, Jarkko Venna, Petri Törönen, and Eero Castrén. Trustworthiness and metrics in visualizing similarity of gene expression. *BMC Bioinformatics*, 4:48, 2003.

Figure 6.5: Three projections of a three-dimensional spherical cell with local MDS. On the left, trustworthiness of the projection is maximized by selecting $\lambda = 0$. In the middle and right, discontinuity of the projection is penalized as well, by setting $\lambda = 0.1$ and $\lambda = 0.9$, respectively. When λ is increased the edges where continuity is violated the worst get pulled closer together to minimize the number of neighborhoods that become split, and to reduce the distance between those neighborhoods that cannot be connected.

[2] Jarkko Venna and Samuel Kaski. Visualized atlas of a gene expression databank. In *Proceedings of Symposium of Knowledge Representation in Bioinformatics*, pages 30–36, Espoo, Finland, 2005.

[3] Jarkko Venna and Samuel Kaski. Local multidimensional scaling with controlled tradeoff between trustworthiness and continuity. In *Proceedings of 5th Workshop on Self-Organizing Maps*, pages 695–702, Paris, France, 2005.