Clustering Gene Expression Data by Mutual Information with Gene Function

Samuel Kaski, Janne Sinkkonen, and Janne Nikkilä

Neural Networks Research Centre
Helsinki University of Technology
P.O.Box 5400, FIN-02015 HUT, Finland
{samuel.kaski, janne.sinkkonen, janne.nikkila}@hut.fi

Abstract. We introduce a simple on-line algorithm for clustering paired samples of continuous and discrete data. The clusters are defined in the continuous data space and become local there, while within-cluster differences between the associated, implicitly estimated conditional distributions of the discrete variable are minimized. The discrete variable can be seen as an indicator of relevance or importance guiding the clustering. Minimization of the Kullback-Leibler divergence-based distortion criterion is equivalent to maximization of the mutual information between the generated clusters and the discrete variable. We apply the method to a time series data set, i.e. yeast gene expressions measured with DNA chips, with biological knowledge about the functions of the genes encoded into the discrete variable.

1 Introduction

Clustering methods depend crucially on the criterion of similarity. In this work we present a new algorithm which uses additional information to decide the relevance of a dissimilarity in the data. This information is available as *auxiliary samples* c_k; they form pairs $(\mathbf{x}_k, c_k)$ with the primary samples $\mathbf{x}_k$. In this study the extra information is discrete and consists of labels of functional classes of genes.

It is assumed that differences in the auxiliary data indicate what is important in the primary data space. Intuitively speaking, the auxiliary data guides the clustering to emphasize the (locally) important dimensions of the primary data space, and to disregard the rest. This automatic relevance detection is the main practical motivation for the present work.

The distance between two close-by points $\mathbf{x}$ and $\mathbf{x}'$ is measured by the differences between the *distributions* of c, given $\mathbf{x}$ and $\mathbf{x}'$. Our aim is to minimize the within-cluster dissimilarities in terms of the (implicitly estimated) distributions $p(c|\mathbf{x})$ and still define the clusters by localized basis functions within the primary space.

Put in different words, we supervise the metric of the data space to concentrate on the important or relevant aspects of the data. The main goal is data-driven unsupervised exploration of the data in the new metric to make new discoveries of its statistical properties. Here the changes in the distribution of the auxiliary data provide the supervised metric, and the unsupervised task is clustering. The method could perhaps be called *semisupervised*, or clustering "guided" by the supervisory signal.

In an earlier work [8] we constructed explicit estimates for the conditional distributions $p(c|\mathbf{x})$ and used them to compute the local distances. In the present work we use an alternative method [11] which does not need an estimate of the conditional distributions $p(c|\mathbf{x})$ as an intermediate step.

Minimizing the within-cluster distortion is equivalent to maximizing the mutual information between the clusters (where cluster membership is interpreted as a value of a multinomial random variable) and the auxiliary data. Maximization of mutual information has been previously used for constructing neural representations [1, 2]. Other related works and paradigms include learning from (discrete) dyadic data [12], and distributional clustering [10] with the information bottleneck [13] principle. These methods start from an estimate of the joint distribution $p(c, x)$, which must be available for all discrete x. Our method places and defines clusters directly in a continuous input space; new data $\mathbf{x}$ can then be assigned to clusters even if the auxiliary data is not available.

In this work the method is applied to gene expression data, which has previously been clustered with conventional clustering algorithms (see, e.g., [5]), and classified with supervised methods (Support Vector Machines, [3]). Our goal is to use the known functional classification of the genes to implicitly define which aspects of the expression data are important in the data, and only utilize the important aspects, local factors or dimensions, in clustering. The difference from ordinary supervised classification methods is that while they cannot surpass the original classes our method is not tied to the classification and may reveal substructures within and relations between the known functional classes.

In this case study we compare our method empirically with alternative methods, and demonstrate the potential usefulness of the results. More detailed biological interpretation of the results will be presented in subsequent papers.

2 Clustering Based on the Kullback-Leibler Divergence

We seek to categorize items $\mathbf{x}$ of the data space by utilizing the information within a set of samples $(\mathbf{x}_k, c_k)$ from a random variable pair (X, C). Here $\mathbf{x} \in \mathbb{X} \subset \mathbb{R}^n$, and the c_k are discrete, multinomial values. We wish to *define* the clusters in terms of $\mathbf{x}$ only, and to keep the clusters local with respect to $\mathbf{x}$. The distortions minimized by the clustering are, however, measured between distributions in the auxiliary space, although it is not necessary to explicitly compute these distributions.

Vector quantization (VQ) or, equivalently, K-means clustering, is one approach to categorization. In ("soft") VQ the goal is to minimize the average distortion E between the data and the prototypes or codebook vectors $\mathbf{m}_j$, defined by

$$E = \sum_j \int y_j(\mathbf{x}) D(\mathbf{x}, \mathbf{m}_j)\, p(\mathbf{x})\, d\mathbf{x}\ . \tag{1}$$

Here $D(\mathbf{x}, \mathbf{m}_j)$ is the distortion between $\mathbf{x}$ and $\mathbf{m}_j$, and $y_j(\mathbf{x})$ is the "soft" cluster membership function that fulfills $0 \leq y_j(\mathbf{x}) \leq 1$ and $\sum_j y_j(\mathbf{x}) = 1$.

We measure distortions by the Kullback–Leibler divergence $D_{KL}(p, \boldsymbol{\psi}) \equiv \sum_i p_i \log(p_i/\psi_i)$, where p_i is the multinomial distribution in the auxiliary space that corresponds to the data $\mathbf{x}$, that is, $p_i \equiv p(c_i|\mathbf{x})$. The second distribution is a prototype for the cluster; let us denote the jth prototype by $\boldsymbol{\psi}_j$.

Although the distortions are measured between distributions of the auxiliary variable, the cluster memberships are still defined in the primary data space, by the membership functions $y_j(\mathbf{x})$. In our application the data lies on a hypersphere, and hence we use membership functions that are defined for spheres. The membership functions are normalized von Mises–Fisher ($\mathrm{vMF}(\mathbf{x}; \boldsymbol{\theta}_j)$) kernels [9], spherical analogues of Gaussians, parameterized by their centers $\boldsymbol{\theta}_j$ (and widths κ): $y_j(\mathbf{x}; \boldsymbol{\theta}_j) = \mathrm{vMF}(\mathbf{x}; \boldsymbol{\theta}_j)/\sum_k \mathrm{vMF}(\mathbf{x}; \boldsymbol{\theta}_k)$, where $\mathrm{vMF}(\mathbf{x}; \boldsymbol{\theta}_j) = \text{const.} \times \exp(\kappa \boldsymbol{x}^T \boldsymbol{\theta}_j / \|\boldsymbol{\theta}_j\|)$.

The final cost function to be optimized is (1), with $y_j(\mathbf{x})$ replaced by the parameterized $y_j(\mathbf{x}; \boldsymbol{\theta}_j)$, and $D(\mathbf{x}, \mathbf{m}_j)$ replaced by the divergence $D_{KL}(p(c|\mathbf{x}), \boldsymbol{\psi}_j)$.

Connection to mutual information. Denote by V the random variable that indicates which cluster a data sample belongs to. That is, the value of V is v_i if the sample belongs to cluster i. It can be shown that our cost function is equal to the (negative) mutual information between V and C, $-I(C;V)$, up to an additive constant.

The algorithm. It can be shown [7] that the average distortion can be minimized with stochastic approximation that samples from $y_j(\mathbf{x}) y_l(\mathbf{x}) p(c_i, \mathbf{x}) = p(v_j, v_l, c_i, \mathbf{x})$. This leads to an on-line algorithm in which the following steps are repeated for $t = 0, 1, \ldots$ with $\alpha(t)$ gradually decreasing towards zero:

1. Draw a data sample $(\mathbf{x}(t), c(t))$. The discrete sample $c(t) = c_i$ defines the value of i in the following steps.
2. Draw two basis functions, j and l, from a multinomial distribution with probabilities $\{y_k(\mathbf{x}(t))\}_k$.
3. Adapt the parameters $\boldsymbol{\theta}_l$ and γ_{lm}, $m = 1, \ldots, N_c$, by

$$\boldsymbol{\theta}_l(t+1) = \boldsymbol{\theta}_l(t) - \alpha(t)(\mathbf{x}(t) - \mathbf{x}(t)^T \boldsymbol{\theta}_l(t) \boldsymbol{\theta}_l(t)) \log \frac{\psi_{ji}(t)}{\psi_{li}(t)} \qquad (2)$$

$$\gamma_{lm}(t+1) = \gamma_{lm}(t) - \alpha(t)(\psi_{lm}(t) - \delta_{mi}) \,, \qquad (3)$$

where N_c is the number of possible values of the random variable C, δ_{mi} is the Kronecker delta, and the γ_l are prototype distributions reparameterized to keep their sum at unity: $\log \psi_{ji} = \gamma_{ji} - \log \sum_m exp(\gamma_{jm})$. Due to the symmetry between j and l, it is possible to adapt the parameters twice for each t by swapping j and l in (2) and (3) for the second adaptation step. Note that $\boldsymbol{\theta}_l(t+1) = \boldsymbol{\theta}_l(t)$ if $j = l$.

3 Case Study: Clustering of Gene Expression data

We clustered 2,467 genes of the budding yeast *Saccharomyces cerevisiae* by their activity or expression in eight experimental conditions (see [5]), measured as a time series with DNA chips. The total of 79 time points were collected into the feature vector. The data was preprocessed in the same way as in [3], and divided into a training set (two thirds) and a test set (the remaining third).

3.1 Alternative models

We compared our model with two standard state-of-the-art mixture density models. The goal of these models is slightly different from ours but we use them to provide baseline results. According to our knowledge they are the most closely related existing models.

The first is a totally unsupervised mixture of von Mises–Fisher distributions. The model is analogous to the usual mixture of Gaussians; the Gaussian mixture components are simply replaced by the von Mises–Fisher components.

In the second model, MDA2 [6], the joint distribution between the functional classes c and the expression data $\mathbf{x}$ is modeled by a set of additive components denoted by u_j:

$$p(c_i, \mathbf{x}) = \sum_j p(c_i|u_j)p(\mathbf{x}|u_j)p_j \ , \tag{4}$$

where $p(c_i|u_j)$ and p_j are parameters to be estimated, and $p(\mathbf{x}|u_j) = \mathrm{vMF}(\mathbf{x}; \boldsymbol{\theta}_j)$. Both models are fitted to the data by likelihood maximization with the EM algorithm [4].

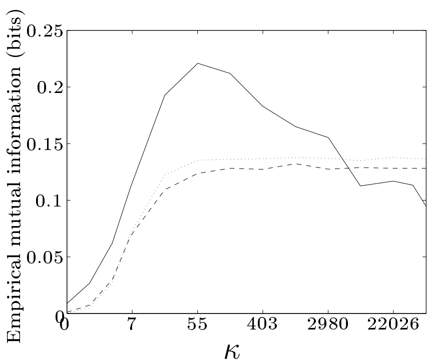
3.2 The experiments

We compared the three models, each with 8 clusters and optimized to near convergence. All models were run three times with different random initializations, and the best of the three results was chosen.

The quality of the resulting clusterings was measured by the average distortion error or, equivalently, the empirical mutual information. The results in Figure 1 **(A)** show that our model clearly outperforms the others for a wide range of widths of the kernels and produces the best overall performance. The clusters of our model therefore convey more information about the functional classification of the genes than the alternative models.

There is a somewhat surprising sideline in the results: The vMF mixture model is slightly better than the MDA2, although it does not utilize the class information at all. The reason is probably in some special property of the data since the likelihood of the paired data (not shown), the measure that the MDA2 optimizes, was larger for MDA2 than for the simple mixture model in which the class distribution in the clusters was estimated afterwards. Moreover, for other data sets the MDA2 has outperformed the plain mixture model.

The distribution of some functional *subclasses* of genes in the clusters is shown in Figure 1 **(B)**. Note that these subclasses were not included in the values of the auxiliary variable C. Instead, they were picked from the second



(A)

Class	Cluster number 1	2	3	4	5	6	7	8
a	0	1	6	0	1	0	0	0
b	0	1	16	0	0	0	0	0
c	1	5	39	1	1	4	14	3
d	0	3	8	1	0	0	0	2
e	122	1	0	2	0	2	44	2
f	3	3	1	20	0	46	2	12
g	0	1	0	21	0	0	0	7

(B)

Fig. 1. **(A)** Empirical mutual information between the generated gene expression clusters and the functional classes of the genes, as a function of parameter κ which governs the width of the basis functions. Solid line: our model, dotted line: mixture of vMFs, dashed line: MDA2. **(B)** Distribution of genes (learning and test set combined) of sample functional subclasses into the 8 clusters. These subclasses were not used in supervising the clustering. a: the pentose-phosphate pathway, b: the tricarboxylic-acid pathway, c: respiration, d: fermentation, e: ribosomal proteins, f: cytoplasmic degradation, g: organization of chromosome structure.

and third level of the functional classification, and they include groups that are known to be regulated in the experimental settings measured by the data [5]. Most of the subclasses are concentrated in one of the clusters. The first four (a-d) belong to the same first-level class and are placed (mostly) in the same cluster, number 3.

Three of the subclasses (c, e, and f) have been clearly divided into more than one cluster, indicating some potentially biologically interesting subdivision of genes inside the subclasses. It remains to be seen in further biological inspection whether such subdivisions provide new biological information; in this paper our goal is to demonstrate that the "semisupervised" clustering approach can be used to explore the data set and provide potential further hypotheses about its structure.

4 Conclusions

We have presented a new clustering method for continuous data which minimizes the within-cluster distortion between distributions of associated, discrete auxiliary data [11]. A simple on-line algorithm was used for optimizing the distortion measure.

We clustered a yeast gene expression data set in which auxiliary information was available as an independent, functional classification of the genes. In our semisupervised task the algorithm performed better than other algorithms available for continuous data, the mixture of vMFs and the MDA2 which models the joint density of the expression data and the classes.

Note that the number of extracted clusters can be arbitrary and it need not depend on the auxiliary data (here the number of classes). Depending on whether the number of clusters is larger or smaller than the number of classes the relationships of the classes or the substructures within the classes can be studied. Such studies are not possible with supervised methods.

It was shown that the obtained clusters mediate information about the function of the genes and, although the results have not yet been biologically analyzed, potentially suggest novel cluster structures for the yeast genes.

Acknowledgments. This work was supported by the Academy of Finland, in part by the grant 50061. We wish to thank Petri Törönen for his help with the data and Jaakko Peltonen with some of the programs.

References

1. S. Becker. Mutual information maximization: models of cortical self-organization. *Network: Computation in Neural Systems*, 7:7–31, 1996.
2. S. Becker and G. E. Hinton. Self-organizing neural network that discovers surfaces in random-dot stereograms. *Nature*, 355:161–163, 1992.
3. M. P. Brown, W. N. Grundy, D. Lin, N. Cristianini, C. W. Sugnet, T. S. Furey, M. Ares, Jr., and D. Haussler. Knowledge-based analysis of microarray gene expression data by using support vector machines. *Proceedings of the National Academy of Sciences, USA*, 97:262–267, 2000.
4. 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.
5. 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, USA*, 95:14863–14868, 1998.
6. 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, 1995.
7. 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.
8. S. Kaski, J. Sinkkonen and J. Peltonen. Bankruptcy Analysis with Self-Organizing Maps in Learning Metrics *IEEE Transactions on Neural Networks*, 2001, in press.
9. K. V. Mardia. Statistics of directional data. *Journal of the Royal Statistical Society. Series B*, 37:349–393, 1975.
10. 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. 1993.
11. J. Sinkkonen and S. Kaski. Clustering based on conditional distributions in an auxiliary space. *Neural Computation*, 2001, in press.
12. 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 Kauffmann Publishers, San Mateo, CA, 1998.
13. N. Tishby, F. C. Pereira, and W. Bialek. The information bottleneck method. In *37th Annual Allerton Conference on Communication, Control, and Computing*, Urbana, Illinois, 1999.