An Unsupervised Bayesian Distance Measure

Petri Kontkanen, Jussi Lahtinen, Petri Myllymäki, and Henry Tirri

Complex Systems Computation Group (CoSCo)
P.O.Box 26, Department of Computer Science
FIN-00014 University of Helsinki, Finland
http://www.cs.Helsinki.FI/research/cosco/

Abstract. We introduce a distance measure based on the idea that two vectors are considered similar if they lead to similar predictive probability distributions. The suggested approach avoids the scaling problem inherent to many alternative techniques as the method automatically transforms the original attribute space to a probability space where all the numbers lie between 0 and 1. The method is also flexible in the sense that it allows different attribute types (discrete or continuous) in the same consistent framework. To study the validity of the suggested measure, we ran a series of experiments with publicly available data sets. The empirical results demonstrate that the unsupervised distance measure is sensible in the sense that it can be used for discovering the hidden clustering structure of the data.

1 Introduction

Machine learning techniques usually aim at compressing available sample data into more compact representations called *models*. These models can then be used for solving different explorative (data mining) or predictive inference tasks. As opposed to traditional model-based machine learning, *lazy learning* [2] refers to methods that defer all essential computation (including the compression of sample data into a model or models) until the specific prediction task is completely determined. This type of predictive inference is also known as *case-based* or *instance-based reasoning* [1, 3, 15, 22, 25], reflecting the fact that the methods use a set of sample instances — the case base — in a central role in the predictive inference process.

In the model-based approach predictive inference problems are solved by building from sample data (and possibly domain-specific prior information) a mathematical model with a set of parameters, which then implements a function that can be used for solving the problem at hand. In the lazy learning approach, on the other hand, no such model is explicitly defined, but prediction problems are solved by comparing the input vector against the samples available in the case base. The actual prediction is typically done by using some *distance function* [15], so that the vectors closest to the input vector affect the result most. This means that from the predictive accuracy point of view, determining the distance function is the key issue in building CBR systems. However, defining a good distance function is difficult, as one has to solve problems related to

E. Blanzieri and L. Portinale (Eds.): EWCBR 2000 LNAI 1898, pp. 148–160, 2000

attribute scaling, handling of different attribute types, and so on. We argue that probability theory offers a general framework where many of these problems disappear, while the approach offers at the same time a theoretically solid approach for building accurate predictive models. In this paper we discuss alternative ways for linking the probabilistic predictive models into the lazy learning framework. To be able to do this, we must first formalize lazy learning as a concept.

Although the lazy learning scenario appears at first sight to be model-free, it is important to realize that this view is flawed: all distance functions are implicitly based on a model (a set of assumptions) that determine how the distance between two high-dimensional vectors is defined. For instance, it is easy to see that the Euclidean distance is based on an underlying probabilistic model with normally distributed, independent variables, while the Mahalanobis distance (see, e.g., [7]) assumes the multivariate normal model. Consequently, lazy-learning cannot be simply defined as the “model-free” approach to machine learning.

In [18] we made an attempt towards a formal definition of lazy learning, and suggested that if the basic idea is to refrain from using a single model (or a set of models), mathematically this corresponds to a requirement for marginalizing (integrating) over all the possible model families and individual models within such families. This type of integration implicitly suggests the lazy learning type of an approach. To see this, note that in the model-based approach the sample data can be compressed into a set of numbers called sufficient statistics [4], and these numbers are all that is required for determining the model parameters (hence the phrase “sufficient”). In the lazy learning approach, on the other hand, the integration over several model families means that a single set of sufficient statistics is not enough: the sufficient statistics with respect to a single family of models may not be sufficient with respect to some other family. This suggests that we should define lazy learning as a set of methods for which the minimal sufficient statistics is the data itself; hence all lazy learning methods are inherently based on using the sample data as it is, in other words, “case-based” by nature.

Inspired by this general theoretical framework, in [19] we suggested a novel distance measure for case-based reasoning tasks. However, although lazy or model-free in its general form, for computational reasons the system was implemented by using a single model as the basis for the suggested distance measure. What is more, the metric was mainly planned for a particular subtask in case-based reasoning, case-retrieval, and was not symmetric in nature. Furthermore, the approach was based on point probabilities of different events while it is clear that similarity should actually be based on properties of probability distributions, not individual probabilities.

In [16] we proposed and analyzed a supervised, probabilistic model-based data reduction scheme where two vectors are considered similar if they lead to similar predictive distributions, when the corresponding attribute-value pairs are given as input to the same probabilistic model. Unlike the metric in [19], this similarity metric is symmetric and is hence suitable for example for visualizing high-dimensional data, as demonstrated in [16]. However, as can be seen in Section 2, the metric is inherently supervised in nature, and cannot be applied in unsupervised domains. What is more, the method is essentially based on a single probabilistic model, and is hence not case-based in the sense discussed above.

To overcome these limitations, in Section 3 we introduce a novel, unsupervised Bayesian distance measure based on an extension of the earlier supervised approach. In order to validate our approach, empirical results obtained with this distance measure are reported in Section 4. In the experiments reported here, the unsupervised distance measure was computed by using a "pool" of supervised Naive Bayes classifier models. The validity of the suggested distance measure was evaluated by both a set of subjective visual tests, and a set of simple crossvalidation experiments.

2 A Supervised Bayesian Distance Metric

Let $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ denote our case base, a collection of N vectors, and let us assume that each vector $\mathbf{x}_i$ consists of values of m attributes $X_1, \ldots, X_m$. In the sequel we will assume the attributes X_i to be discrete (or discretized) — extending the approach to continuous attributes is left as a goal for future work.

As discussed earlier, we argue that there exist no such thing as model-free reasoning, and we suggest that all our assumptions concerning the domain space should be explicitly listed and exploited by using formal models of the problem domain. By a model M we mean here a parametric (probabilistic) model form so that each parameterized instance (M, θ) of the model produces a probability distribution $P(X_1, \ldots, X_m | M, \theta)$ on the space of possible data vectors $\mathbf{x}$. To make our presentation more concrete, for the remainder of the paper we assume that the models M represent different *Bayesian network structures* (for an introduction to Bayesian network models, see e.g., [24, 23, 13, 6]).

The general idea suggested in [16] can be summarized as follows: *two vectors are considered similar if they lead to similar predictive distributions*, when the corresponding attribute-value pairs are given as input to the same Bayesian network model M. To make this idea more precise, we must first define the predictive distribution used in the above informal definition. In [16] this predictive distribution was determined with respect to a special *target variable* X_m, resulting in the conditional distribution

$$P(X_m \mid X_1, \ldots, X_{m-1}, M). \tag{1}$$

Data vectors $\mathbf{x}_i$ and $\mathbf{x}_j$ are now considered similar if the corresponding predictive distributions are similar, i.e.

$$P(X_m \mid \mathbf{x}_i^{-m}, M) \approx P(X_m \mid \mathbf{x}_j^{-m}, M), \tag{2}$$

where $\mathbf{x}_i^{-m}$ denotes the attribute values in vector $\mathbf{x}_i$ without the value of the class variable X_m.

This type of similarity measures lead to supervised distance measures, and we can easily change the focus of the metric by changing the target variable X_m. The scheme is also scale invariant as we have moved from the original attribute space to the probability space where all the numbers lie between 0 and 1. This also allows us to handle different attribute types (discrete or continuous) in the same consistent framework. Furthermore, the framework fulfills the requirement stated earlier: the approach is theoretically on a solid basis as all our domain assumptions must be formalized in the model M.

The above scheme still leaves us with the question of defining a similarity measure between two predictive distributions. The standard solution for computing the distance between two distributions is to use the Kullback-Leibler divergence (see, e.g, [10]). However, this asymmetric measure is not (in its basic form) a distance metric in the geometric sense. In the empirical experiments reported in [16] it was observed that the following simpler distance metric yields good results in practice:

$$d_m(\mathbf{x}_i, \mathbf{x}_j) = 1.0 - P(\mathrm{MAP}_m(\mathbf{x}_i) = \mathrm{MAP}_m(\mathbf{x}_j)), \tag{3}$$

where $\mathrm{MAP}_m(\mathbf{x}_i)$ denotes the *maximum posterior probability* value of target variable X_m with respect to the predictive distribution (1), conditioned on the values of variables $X_1, \ldots, X_{m-1}$ in $\mathbf{x}_i$:

$$\mathrm{MAP}_m(\mathbf{x}_i) = \arg\max_k P(X_m = k \mid \mathbf{x}_i^{-m}, M). \tag{4}$$

In this paper we however use a slightly different distance function $d(\mathbf{x}_i, \mathbf{x}_j)$ based on a straightforward logarithmic transformation of the predictive probabilities:

$$d_m(\mathbf{x}_i, \mathbf{x}_j) = -\log P(\mathrm{MAP}_m(\mathbf{x}_i) = \mathrm{MAP}_m(\mathbf{x}_j)). \tag{5}$$

As noted in [16], extending this general approach to cases with two or more target variables is straightforward.

3 An Unsupervised Bayesian Distance Metric

The distance metric described in Section 2 is inherently supervised in nature, as it requires us to choose one (or more) of the domain variables to be used as the target variable. Consequently, this approach cannot be directly used for cases where no natural candidate for such a target variable exists, and we would like to process our data in a purely unsupervised manner. For such unsupervised domains, we propose the following unsupervised extension of the suggested supervised distance metric:

$$d(\mathbf{x}_i, \mathbf{x}_j) = \sum_{k=1}^{m} d_k(\mathbf{x}_i, \mathbf{x}_j). \tag{6}$$

Consequently, the distance between two vectors $\mathbf{x}_i$ and $\mathbf{x}_j$ is computed by taking each of the variables X_k in its turn as the target variable, and summing the resulting m supervised distance measures computed by formula (5). Intuitively speaking, this means that two vectors $\mathbf{x}_i$ and $\mathbf{x}_j$ are considered similar, if the most probable outcome is the same in both cases in all the m individual supervised prediction tasks based on the conditional distributions

$$P(X_k \mid X_1, \ldots, X_{k-1}, X_{k+1}, \ldots, X_m, M), k = 1, \ldots, m. \tag{7}$$

As we are using a sum of logarithms in the definition of our overall unsupervised distance function (6), this means that we are basically treating all these separate supervised prediction tasks independently.

The above scenario still leaves us with a very pragmatic and important question: what is the model structure M to be used for defining the conditional distributions (7)? Traditionally, this question is answered in the Bayesian framework by using the model M maximizing the posterior probability $P(M \mid \mathbf{X})$. Assuming the uniform prior for the model structures, this is equivalent to using the model with the highest *marginal likelihood* or *evidence*,

$$M = \arg\max_{M'} P(\mathbf{X} \mid M') = \arg\max_{M'} \int P(\mathbf{X} \mid \Theta, M') P(\Theta \mid M') d\Theta. \tag{8}$$

With certain technical assumptions, this model selection criterion can be computed in closed form [8, 11]. The required conditional distribution (1) can then be computed by marginalizing the joint probability distribution $P(\mathbf{x}_i \mid \mathbf{X}, M)$ appropriately:

$$\begin{aligned} P(X_m = x_{im} \mid \mathbf{x}_i^{-m}, \mathbf{X}, M) &= \frac{P(X_m = x_{im}, \mathbf{x}_i^{-m}, \mathbf{X}, M)}{P(\mathbf{x}_i^{-m}, \mathbf{X}, M)} \\ &= \frac{P(\mathbf{x}_i \mid \mathbf{X}, M)}{\sum_k P(X_m = k, \mathbf{x}_i^{-m} \mid \mathbf{X}, M)}. \end{aligned} \tag{9}$$

We see that this approach is consistent with our definition of lazy learning in the sense that we marginalize in (8) over all the possible parameter instantiations. To follow this line of reasoning further, we see that should actually marginalize also over all the possible model structures M as well, yielding

$$P(\mathbf{x} \mid \mathbf{X}) = \sum_M P(\mathbf{x} \mid \mathbf{X}, M) P(M \mid \mathbf{X}) \propto \sum_M P(\mathbf{x} \mid \mathbf{X}, M) P(\mathbf{X} \mid M), \tag{10}$$

assuming a uniform prior $P(M)$ for the model structures. Consequently, we should weight the prediction obtained by using model structure M by the marginal likelihood $P(\mathbf{X}|M)$.

It is obvious that computing the sum (10) in practice is not feasible as the number of possible Bayesian networks is super-exponential. The sum can be approximated efficiently if we can find the most important individual terms, i.e., model structures with high marginal likelihood. However, as noted in e.g. [12], finding high evidence model structures is an extremely difficult problem. This means that in practical situations we are dealing with model structures M that are possibly only poor models of the "true" joint domain probability distribution, and hence some of the probabilities obtained are not correct. As demonstrated in [20], instead of trying to find good models of the joint probability distribution, in supervised classification domains it makes sense to try to find a model (or a set of models) so that the errors affect the accuracy of the conditional distribution (1) as little as possible, while we can allow the joint probability distribution to be such that the predictions concerning some other variable would be quite inaccurate.

For this reason, we suggest that instead of using a single model structure M for determining the distance measure (6), we should use m supervised models $M_1, \ldots, M_m$, each chosen with respect to the corresponding predictive task. Unfortunately, as discussed in [12, 9, 20], finding accurate Bayesian network models for supervised prediction tasks is also a difficult problem. We return to this issue in Section 4 where we suggest computationally efficient heuristics for solving this problem.

4 Empirical Results

4.1 The setup

To illustrate the validity of the suggested similarity metric, we performed a series of experiments with publicly available classification data sets from the UCI data repository [5]. The 20 data sets used and their main properties are listed in Table 1. In the preprocessing phase of the experimental setup, all continuous attributes in the data sets were discretized by using a straightforward application of the k-means algorithm. Consequently, with respect to the empirical study reported here, all the data sets were discrete.

When computing the pairwise distances between vectors $\mathbf{x}_i$ and $\mathbf{x}_j$ according to Equation (6), we need to determine m predictive distributions $P(X_m \mid \mathbf{x}_i^{-m}, M_m)$. As discussed in Section 3, finding good Bayesian network models M_m for supervised prediction tasks is difficult in practice. On the other hand, as demonstrated in, for example, [26, 17], the structurally simple Naive Bayes classifier performs surprisingly well in many real-world classification domains, despite the fact that the model is extremely fast to construct and use. For this reason, in this series of experiments the predictive models M_m corresponded to m Naive Bayes models, each constructed with respect to the target variable X_m. The model parameters were set to their expected values (not to their maximum probability values). These parameter values lead to a predictive distribution equivalent to that obtained by integrating over the parameter values, and as demonstrated in [21], this results in improved classification accuracy in cases where the amount of data is very small.

To illustrate the validity of the suggested unsupervised distance measure, the data vectors in each data set were visualized in two-dimensional space by using a technique called *Sammon's mapping* (see e.g. [14]). In this method the data vectors are placed in low-dimensional space in such a manner that the pairwise

Table 1. The data sets used in the experiments.

Dataset	size	#attrs.	#classes
Australian Credit	690	15	2
Breast Cancer (Wisconsin)	699	11	2
Breast Cancer	286	10	2
Credit Screening	690	16	2
Pima Indians Diabetes	768	9	2
German Credit	1000	21	2
Heart Disease (Cleveland)	303	14	5
Heart Disease (Hungarian)	294	14	2
Heart Disease (Statlog)	270	14	2
Hepatitis	155	20	2
Ionosphere	351	35	2
Iris Plant	150	5	3
Liver Disorders	345	7	2
Lymphography	148	19	4
Mole Fever	425	33	2
Postoperative Patient	90	9	3
Thyroid Disease	215	6	3
Vehicle Silhouettes	846	19	4
Congressional Voting Records	435	17	2
Wine Recognition	178	14	3

distances in this space match as closely as possible to the pairwise distances in the original high-dimensional space. The objective function, corresponding to the relative error caused by the dimensionality reduction, is typically quite complex and finding the optimal visualization in this sense is not possible in practice, so we are left with approximative solutions. However, it should be pointed out that in this visualization context it is not necessary to aim at the absolutely optimal solution — for visualization purposes a reasonable approximation is usually quite sufficient. How to find effectively good approximations of the optimal visualization is, however, a challenging research problem on its own, and is not discussed in detail here. In the experiments reported here we used a simple iterative stochastic greedy algorithm where at each step the visual locations of two randomly chosen data vectors are optimized along the connecting line so that the objective function is optimized locally.

The practical relevance of a visualization can be indirectly measured through a data mining process, where domain experts try to capture interesting regularities from the visual image. In our case, however, no such domain experts were available, so we had to evaluate our visualization technique in a different manner. In this set of experiments, this was done by assuming that the clustering provided by the class labels in the UCI data sets is a reasonable clustering, in the sense that this clustering can be regarded as something that we should come up with, had we not seen the class labels originally. Following this line of reasoning, each classification data set was first pruned by removing the class labels, i.e., the column containing the values of the class variable X_m. The remaining data $\mathbf{X}$ was then visualized in two-dimensional space by using the unsupervised approached presented here. Finally, the produced visual images were colored according to the class labels that were not used at all in the visualization process. If the resulting image was visually pleasing in the sense that the different classes (different colors) were nicely separated in the picture, it can be said that we were able to recover the original clustering in a *totally unsupervised manner, without using the class label information.*

4.2 The results

The empirical results show that the suggested unsupervised distance measure is sensible in the sense that it can be used for discovering the underlying cluster structure: most of the produced visual images pass the class coloring clarity test explained above. A black-and-white example of the produced visualizations can be found in Figure 1. A full library of colored 2D images can be found at URL: http://www.cs.Helsinki.FI/research/cosco/Projects/Visual/EWCBR2000.

There is however a possible caveat in the above experimental procedure: basically there is no a priori reason why the unsupervised visual image produced should reflect the clustering provided by the class labels, especially if the original clustering is poor from the probabilistic modeling point of view. One way to measure the "goodness" of the clustering provided by the class labels is to evaluate the predictive accuracy of the Naive Bayes model, as this model is essentially based on clustering the data according to the class variable X_m. Leave-one-out cross-validated classification results of the Naive Bayes classifier can be found in the second column of Table 2.

We can now conjecture that the above "class-color clarity test" for the resulted visual images may fail with data sets where the performance of the Naive Bayes classifier is poor. The experimental results confirm this hypothesis: in cases where the leave-one-out cross-validated classification accuracy of the NB classifier is poor in the absolute sense (as with the Liver Disorders data set), or in the relative sense with respect to the default classification accuracy (as with the Postoperative Patient data set), the class labeled colored images are somewhat blurred. Nevertheless, we would like to emphasize that this does not mean that the unsupervised visualization technique has "failed" in these cases, but that in these (relatively few) cases the somewhat artificial empirical setup used here is not practically sensible in the first place. This means that if the data would in these cases be clustered according to the visual image produced, this could result in a probabilistic model producing more accurate predictions than the Naive Bayes classifier. This interesting idea is however not studied further here.

We believe that most people agree that the produced images (with the exception of the few cases discussed above) are visually pleasing in the sense that the original classes are clearly separable in the image. This however raises the

Fig. 1. The Ionosphere data set: an example of the unsupervised visualizations obtained with the suggested distance measure.

question of whether the quality of the visualization could be measured more objectively. Intuitively, we would like to measure how well the data in the visual image is clustered according to the (hidden) class label. We suggest that this can be done by using, for example, a simple k-NN (nearest neighbor) method, where each data point is classified as a member of the class containing the most representatives in the k nearest data points. The results with this type of k-NN (with k=9) classification method are summarized in Table 2. The method 9-NN refers to a simple 9-nearest neighbor classifier, where the distances of the (m-1)-dimensional (recall that the class labels were omitted from the data) vectors are computed by formula (6).

From Table 2 we can see that the classification accuracy obtained with the 9-NN classifier is quite comparable (and in some cases even better!) to the accuracy of the NB classifier, *even though the nearest neighbor classification was done in a purely unsupervised manner* — recall that the distance measure was computed without the class labels. This is even more surprising considering the fact that the 9-NN classifier used here was just a random and naive choice for performing CBR-type of classification, as the idea was only to demonstrate the potentiality of the suggested distance measure.

Table 2. Leave-one-out cross-validated classification results obtained with the data sets.

Dataset	Default	NB	9-NN
Australian Credit	55.5	87.1	82.0
Breast Cancer (Wisconsin)	65.5	97.4	97.3
Breast Cancer	70.3	72.3	71.7
Credit Screening	55.5	86.2	82.9
Pima Indians Diabetes	65.1	77.9	72.7
German Credit	70.0	74.9	67.4
Heart Disease (Cleveland)	54.1	57.8	57.4
Heart Disease (Hungarian)	63.9	83.3	82.7
Heart Disease (Statlog)	55.6	85.2	83.7
Hepatitis	79.4	83.2	82.6
Ionosphere	64.1	92.9	91.2
Iris Plant	33.3	94.0	88.7
Liver Disorders	58.0	63.2	58.8
Lymphography	54.7	85.8	81.8
Mole Fever	67.1	87.8	89.2
Postoperative Patient	71.1	67.8	71.1
Thyroid Disease	69.8	99.1	97.2
Vehicle Silhouettes	25.8	64.7	66.5
Congressional Voting Records	61.4	90.1	88.0
Wine Recognition	39.9	97.2	96.1

However, we must admit that the empirical setup used is not quite satisfactory as the two classifiers work in a different space. If one corrects this deficiency by simply incorporating the class label information into the Sammon's mapping process, then it was observed that the 9-NN classifier outperforms the NB classifier consistently in every case. Nevertheless, it is important to notice that this type of an empirical setup is not absolutely fair either: if one incorporates all class label information in the Sammon's mapping process, this means that when performing leave-one-out cross-validation, some information about the test vector is included in the training data, which is not acceptable. To make the comparison completely fair in every respect, the Sammon's mapping procedure should be run N times, separately for each test vector during the loo-cross-validation cycle. This is however computationally quite demanding, and was left for a task in our future work.

Nevertheless, the results clearly suggest that the probabilistic distance-based approach used here offers a promising approach for producing accurate classifiers. This hypothesis is supported by the following argument: as described in Section 3, the unsupervised distance measure is computed by using a "pool" of $m-1$ Naive Bayes models, one model for each of the variables $X_1, \ldots, X_{m-1}$, whereas the standard Naive Bayes classifier uses a single NB model, constructed with respect to the class variable X_m. As the relatively unrealistic independence assumptions made in the Naive Bayes approach do not always hold very well, the resulting predictive distributions may be inaccurate. We conjecture that errors caused by this violation of assumptions are "smoothed" in the unsupervised case as a result of the distance measure being essentially a sum of several supervised distance measures produced by independent NB classifiers.

5 Conclusion

In this paper we presented an unsupervised extension of the supervised Bayesian distance metric suggested earlier. The suggested method is based on the idea that two vectors are considered similar if they lead to similar predictive probability distributions. In the earlier supervised approach this predictive distribution was defined with respect to a specific target variables (or a group of variables), whereas the suggested unsupervised metric is computed in "crossvalidated manner" by changing the target variable so that each of the variables serves in turn in this role.

The distance measure was validated in a series of experiments where the hidden structure of the data was discovered by producing a 2-dimensional visual picture of the domain. Furthermore, the distance measure was also applied in a simple nearest neighbor classifier, and the results support the visual observations: the distance measure obviously captures interesting regularities in the data. It should be emphasized that the classification results are surprisingly good with respect to those obtained by the Naive Bayes classifier, considering the fact that comparing the results is not at all fair as the nearest neighbor classifier worked in a totally unsupervised manner *without any knowledge of the class labels in the data*, whereas the Naive Bayes classifier had access to all data. This suggests that this type of an approach offers a very promising framework for building accurate case-based classifiers, but this is left as a topic for future work.

Acknowledgements

This research has been supported by the National Technology Agency, and the Academy of Finland.

References

1. D. Aha. *A Study of Instance-Based Algorithms for Supervised Learning Tasks: Mathematical, Empirical, an Psychological Observations.* PhD thesis, University of California, Irvine, 1990.
2. D. Aha, editor. *Lazy Learning.* Kluwer Academic Publishers, Dordrecht, 1997. Reprinted from Artificial Intelligence Review, 11:1–5.
3. C. Atkeson, A. Moore, and S. Schaal. Locally weighted learning. In Aha [2], pages 11–73.
4. J.O. Berger. *Statistical Decision Theory and Bayesian Analysis.* Springer-Verlag, New York, 1985.

5. C. Blake, E. Keogh, and C. Merz. UCI repository of machine learning databases, 1998. URL: http://www.ics.uci.edu/~mlearn/MLRepository.html.
6. E. Castillo, J. Gutiérrez, and A. Hadi. *Expert Systems and Probabilistic Network Models*. Monographs in Computer Science. Springer-Verlag, New York, NY, 1997.
7. C. Chatfield and A. Collins. *Introduction to Multivariate Analysis*. Chapman and Hall, New York, 1980.
8. G. Cooper and E. Herskovits. A Bayesian method for the induction of probabilistic networks from data. *Machine Learning*, 9:309–347, 1992.
9. N. Friedman, D. Geiger, and M. Goldszmidt. Bayesian network classifiers. *Machine Learning*, 29:131–163, 1997.
10. A. Gelman, J. Carlin, H. Stern, and D. Rubin. *Bayesian Data Analysis*. Chapman & Hall, 1995.
11. D. Heckerman, D. Geiger, and D.M. Chickering. Learning Bayesian networks: The combination of knowledge and statistical data. *Machine Learning*, 20(3):197–243, September 1995.
12. D. Heckerman and C. Meek. Models and selection criteria for regression and classification. In D. Geiger and P. Shenoy, editors, *Uncertainty in Arificial Intelligence 13*, pages 223–228. Morgan Kaufmann Publishers, San Mateo, CA, 1997.
13. F. Jensen. *An Introduction to Bayesian Networks*. UCL Press, London, 1996.
14. T. Kohonen. *Self-Organizing Maps*. Springer-Verlag, Berlin, 1995.
15. J. Kolodner. *Case-Based Reasoning*. Morgan Kaufmann Publishers, San Mateo, 1993.
16. P. Kontkanen, J. Lahtinen, P. Myllymäki, T. Silander, and H. Tirri. Using Bayesian networks for visualizing high-dimensional data. *Intelligent Data Analysis*, 2000. To appear.
17. P. Kontkanen, P. Myllymäki, T. Silander, and H. Tirri. BAYDA: Software for Bayesian classification and feature selection. In R. Agrawal, P. Stolorz, and G. Piatetsky-Shapiro, editors, *Proceedings of the Fourth International Conference on Knowledge Discovery and Data Mining (KDD-98)*, pages 254–258. AAAI Press, Menlo Park, 1998.
18. P. Kontkanen, P. Myllymäki, T. Silander, and H. Tirri. Bayes optimal instance-based learning. In C. Nédellec and C. Rouveirol, editors, *Machine Learning: ECML-98, Proceedings of the 10th European Conference*, volume 1398 of *Lecture Notes in Artificial Intelligence*, pages 77–88. Springer-Verlag, 1998.
19. P. Kontkanen, P. Myllymäki, T. Silander, and H. Tirri. On Bayesian case matching. In B. Smyth and P. Cunningham, editors, *Advances in Case-Based Reasoning, Proceedings of the 4th European Workshop (EWCBR-98)*, volume 1488 of *Lecture Notes in Artificial Intelligence*, pages 13–24. Springer-Verlag, 1998.
20. P. Kontkanen, P. Myllymäki, T. Silander, and H. Tirri. On supervised selection of Bayesian networks. In K. Laskey and H. Prade, editors, *Proceedings of the 15th International Conference on Uncertainty in Artificial Intelligence (UAI'99)*, pages 334–342. Morgan Kaufmann Publishers, 1999.
21. P. Kontkanen, P. Myllymäki, T. Silander, H. Tirri, and P. Grünwald. On predictive distributions and Bayesian networks. *Statistics and Computing*, 10:39–54, 2000.
22. A. Moore. Acquisition of dynamic control knowledge for a robotic manipulator. In *Seventh International Machine Learning Workshop*. Morgan Kaufmann, 1990.
23. R.E. Neapolitan. *Probabilistic Reasoning in Expert Systems*. John Wiley & Sons, New York, NY, 1990.
24. J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann Publishers, San Mateo, CA, 1988.

25. C. Stanfill and D. Waltz. Toward memory-based reasoning. *Communications of the ACM*, 29(12):1213–1228, 1986.
26. H. Tirri, P. Kontkanen, and P. Myllymäki. Probabilistic instance-based learning. In L. Saitta, editor, *Machine Learning: Proceedings of the Thirteenth International Conference (ICML'96)*, pages 507–515. Morgan Kaufmann Publishers, 1996.