Pp. 254–258 in *Proceedings of The Fourth International Conference on Knowledge Discovery and Data Mining (KDD'98)*, edited by R. Agrawal, P. Stolorz and G. Piatetsky-Shapiro. AAAI Press, Menlo Park, CA, 1998.

BAYDA: Software for Bayesian Classification and Feature Selection

Petri Kontkanen, Petri Myllymäki, Tomi Silander, Henry Tirri
Complex Systems Computation Group (CoSCo)
P.O.Box 26, Department of Computer Science, FIN-00014 University of Helsinki, Finland
cosco@cs.Helsinki.FI, http://www.cs.Helsinki.FI/research/cosco/

Abstract

BAYDA is a software package for flexible data analysis in predictive data mining tasks. The mathematical model underlying the program is based on a simple Bayesian network, the Naive Bayes classifier. It is well-known that the Naive Bayes classifier performs well in predictive data mining tasks, when compared to approaches using more complex models. However, the model makes strong independence assumptions that are frequently violated in practice. For this reason, the BAYDA software also provides a feature selection scheme which can be used for analyzing the problem domain, and for improving the prediction accuracy of the models constructed by BAYDA. The scheme is based on a novel Bayesian feature selection criterion introduced in this paper. The suggested criterion is inspired by the Cheeseman-Stutz approximation for computing the marginal likelihood of Bayesian networks with hidden variables. The empirical results with several widely-used data sets demonstrate that the automated Bayesian feature selection scheme can dramatically decrease the number of relevant features, and lead to substantial improvements in prediction accuracy.

Introduction

In this paper we are interested in the problem of finding the most relevant subset of features for coding our domain. We take the *predictive data mining* point of view, and focus on finding those features that have predictive power (for a general discussion on various definitions of "relevance", see (Blum & Langley 1997)). A recent pragmatic survey of this area can be found in (Weiss & Indurkhya 1998).

Feature selection is often motivated by performance issues: in many cases the purpose of feature selection is elimination of irrelevant or redundant features without sacrificing prediction performance. From purely theoretical standpoint, having more features should always give us better classification performance. In practice, however, this is not generally the case for two reasons. First, finding the optimal (or even approximately optimal) predictive model becomes computationally intractable for most model classes with the large number of features present in data mining applications. Thus reducing the search space allows better models to be found with the available computing resources. More fundamentally, most classification algorithms can be viewed as performing (a biased form of) estimation of the probability of the class value distribution, given a set of features. In domains with many features, this distribution is complex and of high dimension, and the parameter estimation task with a limited data set is difficult. From the classical statistics point of view, this difficulty exhibits itself as the bias-variance trade-off (Geman, Bienenstock, & Wilson 1992; Kohavi & Wolpert 1996): the trade-off of allowing more parameters with the problem of accurately estimating these parameters. From the Bayesian perspective, any classification algorithm performs the search in a set of possible models (e.g., certain types of decision trees or neural network structures), and thus makes an underlying assumption of the dependency structure of the features. If these assumptions are violated in the full feature set, better classification result can be obtained by restricting the set of features to subsets for which violations of these dependency assumptions is less severe. For this reason, *pruning irrelevant features can improve the predictive accuracy of a classifier even when an "optimal" model can be found for the full feature set.*

In this paper we describe BAYDA, a software tool for Bayesian classification and feature selection. The mathematical model underlying the program is based on a simple Bayesian network, the *Naive Bayes classifier*. However, it should be observed that in contrast to most Naive Bayes classifiers reported in the literature (see, e.g., (Duda & Hart 1973; Langley, Iba, & Thompson 1992; Friedman, Geiger, & Goldszmidt 1997)), the BAYDA model is fully Bayesian in the sense that the program produces the predictive distribution for classification by integrating over all the possible parameter instantiations. Use of this kind of marginal likelihood predictive distributions gives theoretically more accurate predictions than a single model with maximum likelihood or maximum posterior parameters — a fact that has also been verified empirically (Kontkanen *et al.* 1997b; 1997c).

The reasons for using the Naive Bayes model in BAYDA can be justified along two lines. Firstly, it is well-known that the Naive Bayes classifier performs well when compared to approaches using more complex models (Friedman, Geiger, & Goldszmidt 1997; Kohavi & John 1997; Kontkanen *et al.* 1997b; Langley, Iba, & Thompson 1992), and it has proven to be viable for real-world predictive data mining tasks (the first and the third prize of the KDD'97 Data Mining CUP were given to Naive Bayes classifiers). Secondly, the Naive Bayes classifier makes strong independence assumptions frequently violated in practice, which is an obvious source of classification errors (Langley & Sage 1994). Using feature selection to improve prediction accuracy is thus particularly relevant in the context of Naive Bayes classifiers.

In the BAYDA software, both filter and wrapper feature selection schemes (see (John, Kohavi, & Pfleger 1994)) are available. For the wrapper approach, the user has the opportunity to manually select different feature subsets, which can then be evaluated by using leave-one-out crossvalidated classification error. For the filter approach, the program uses automated search with a novel, theoretically justified Bayesian criterion for feature subset selection. The criterion is based on the *supervised marginal likelihood* (closely related to the conditional log likelihood in (Friedman, Geiger, & Goldszmidt 1997)) of the class value vector, given the rest of the data. Unfortunately it turns out that this criterion cannot be efficiently computed even for cases where computing the corresponding unsupervised marginal likelihood is straightforward to do (including the class of Naive Bayes models). However, we introduce an efficient method, inspired by (Cheeseman & Stutz 1996), for computing this criterion approximately.

In the empirical part of the paper we present results with several real-world data sets, and demonstrate that the proposed feature selection criterion correlates well with the predictive performance of the resulting classifier. Furthermore, the results show that the automated Bayesian feature selection scheme dramatically decreases the number of relevant features, thus offering a valuable tool for data mining applications. The results also show that pruning of irrelevant features not only improves the execution performance of the system, but it can also lead to substantial improvements in prediction accuracy.

The Naive Bayes classifier

In the Naive Bayes classifier, the discrete domain attributes (features) $A_1,\ldots,A_m$ are assumed to be independent, given the values of the class attribute C. It follows that the joint probability distribution for a data vector $(\mathbf{d},c) = (A_1 = d_1,\ldots,A_m = d_m, C = c)$ can be written as

$$P(\mathbf{d},c) = P(C=c)\prod_{i=1}^{m} P(A_i = d_i | C = c).$$

Consequently, in the Naive Bayes model case, distribution $P(\mathbf{d},c)$ can be uniquely determined by fixing the values of the parameters $\Theta = (\alpha,\Phi)$, $\alpha = (\alpha_1,\ldots,\alpha_K)$, and $\Phi = (\Phi_{11},\ldots,\Phi_{1m},\ldots,\Phi_{K1},\ldots,\Phi_{Km})$,where the value of parameter α_k gives the probability $P(C=k)$, and $\Phi_{ki} = (\phi_{ki1},\ldots,\phi_{kin_i})$, where $\phi_{kil} = P(A_i = l | C = k)$. Here n_i denotes the number of possible values for attribute A_i, and K the number of possible classes (the number of values for attribute C). Using these denotations, we can now write $P(\mathbf{d},c|\Theta) = \alpha_c \prod_{i=1}^{m} \phi_{cid_i}$.

In the following we assume that both the class attribute distribution $P(C)$ and the intra-class conditional distributions $P(A_i|C=k)$ are multinomial, i.e., $C \sim \text{Multi}(1;\alpha_1,\ldots,\alpha_K)$, and $A_{i|k} \sim \text{Multi}(1;\phi_{ki1},\ldots,\phi_{kin_i})$. Since the family of Dirichlet densities is *conjugate* to the family of multinomials, it is convenient to assume that the prior distributions of the parameters are from this family (see, e.g., (Heckerman, Geiger, & Chickering 1995)). More precisely, let $(\alpha_1,\ldots,\alpha_K) \sim \text{Di}(\mu_1,\ldots,\mu_K)$, and $(\phi_{ki1},\ldots,\phi_{kin_i}) \sim \text{Di}(\sigma_{ki1},\ldots,\sigma_{kin_i})$, where $\{\mu_k,\sigma_{kil} | k = 1,\ldots,K; i = 1,\ldots,m; l = 1,\ldots,n_i\}$ are the *hyperparameters* of the corresponding distributions. A more detailed discussion on the priors can be found in (Kontkanen *et al.* 1998; 1997b).

Having now defined the prior distribution, we get a posterior distribution $P(\Theta|D)$ for the parameters. By choosing from this distribution the *maximum posterior probability (MAP)* parameter values $\hat{\Theta}$, we arrive at the following probability distribution:

$$P(\mathbf{d},c|\hat{\Theta}) = \frac{h_c+\mu_c-1}{N+\sum_{k=1}^{K}\mu_k-1}\prod_{i=1}^{m}\frac{f_{cid_i}+\sigma_{cid_i}-1}{h_c+\sum_{l=1}^{n_i}\sigma_{cil}-1}, \quad (1)$$

where h_c and f_{cil} are the *sufficient statistics* of the training data D: h_c is the number of data vectors where the class attribute has value c, and f_{cil} is the number of data vectors where the class attribute has value c and attribute A_i has value l. As discussed in, for example, (Tirri, Kontkanen, & Myllymäki 1996), a more accurate distribution can be obtained by integrating over all the individual models Θ, resulting in

$$\begin{aligned} P(\mathbf{d},c|D) &= \int P(\mathbf{d},c|D,\Theta)P(\Theta|D)d\Theta \\ &= \frac{h_c+\mu_c}{N+\sum_{k=1}^{K}\mu_k}\prod_{i=1}^{m}\frac{f_{cid_i}+\sigma_{cid_i}}{h_c+\sum_{l=1}^{n_i}\sigma_{cil}}, \qquad (2)\end{aligned}$$

This formula can be derived by using the results in (Cooper & Herskovits 1992; Heckerman, Geiger, & Chickering 1995). In the experiments reported in this paper, we used the uniform prior for the parameters (all the hyperparameters were set to 1), in which special case formula (2) yields the known *Laplace estimates* for the model parameters.

A Bayesian criterion for feature selection

Let $\mathcal{F}$ denote a subset of the set $\mathcal{A} = \{A_1,\ldots,A_m\}$ of all the domain features (attributes). Our goal is now to find a Bayesian criterion for feature selection, a measure $S(\mathcal{F})$ that can be used for evaluating different subsets $\mathcal{F}$. From the Bayesian point of view, in unsupervised learning different subsets $\mathcal{F}$ should be ranked according their posterior probability $P(\mathcal{F}|D_{\mathcal{A}},D_C)$,

$$P(\mathcal{F}|D_{\mathcal{A}},D_C) \propto P(D_C|D_{\mathcal{A}},\mathcal{F})P(D_{\mathcal{A}}|\mathcal{F})P(\mathcal{F}),$$

where $D_C = (c_1,\ldots,c_N)$ denotes a vector containing the values of the class variable C, and $D_{\mathcal{A}}$ is the data matrix without the column D_C: $D_{\mathcal{A}} = (\mathbf{d}_1,\ldots,\mathbf{d}_N)$, where $\mathbf{d}_j = (d_{j1},\ldots,d_{jm})$. Furthermore, assuming different subsets $\mathcal{F}$

to be equally probable a priori, in other words, $P(\mathcal{F})$ to be uniform, we get

$$P(\mathcal{F}|D_{\mathcal{A}},D_C) \propto P(D_C|D_{\mathcal{A}},\mathcal{F})P(D_{\mathcal{A}}|\mathcal{F}). \tag{3}$$

The first term of (3) is the conditional probability for the values of the class variable C, given the data with respect to the values of variables in $\mathcal{F}$, while the second term is the conditional probability of $D_{\mathcal{A}}$, given the fact that only the attributes in $\mathcal{F}$ are to be used. Another way to look at (3) is to say that the first term models the domain with respect to the class variable C, while the second term represent a model of the joint distribution for the other variables. As we are here interested in supervised classification only, we can now see that instead of $P(\mathcal{F}|D_{\mathcal{A}},D_C)$, we should use only the first term in the right hand side of Eq. (3) for ranking different attribute subsets. The proposed feature selection measure thus becomes $S(\mathcal{F}) = P(D_C|D_{\mathcal{A}},\mathcal{F})$.

It turns out, however, that the suggested Bayesian supervised feature selection criterion is computationally infeasible in general. This can be seen by writing

$$P(D_C|D_{\mathcal{A}},\mathcal{F}) = \frac{P(D_C,D_{\mathcal{A}}|\mathcal{F})}{P(D_{\mathcal{A}}|\mathcal{F})} = \frac{P(D_C,D_{\mathcal{A}}|\mathcal{F})}{\sum_{D'_C} P(D'_C,D_{\mathcal{A}}|\mathcal{F})},$$

where the summation goes over all the possible value configurations on the class variable vector D_C. These configurations are clearly exponential in number, so we need to find an approximation to this theoretically correct measure.

The *Cheeseman-Stutz* (C-S) approximation (or measure) used in the Autoclass system (Cheeseman & Stutz 1996) has been found to be computationally feasible, yet quite accurate when compared to alternative approximative methods for computing marginal likelihoods with incomplete data (Kontkanen *et al.* 1997a; Kontkanen, Myllymäki, & Tirri 1996; Chickering & Heckerman 1997). Inspired by the C-S approximation, we start with the equality

$$P(D_C|D_{\mathcal{A}},\mathcal{F}) = \frac{\int P(D_C,D_{\mathcal{A}}|\Theta,\mathcal{F})P(\Theta|\mathcal{F})d\Theta}{\int P(D_{\mathcal{A}}|\Theta,\mathcal{F})P(\Theta|\mathcal{F})d\Theta}.$$

Now assuming that both integrands peak at the same (MAP) point $\hat{\Theta}$ maximizing the posterior

$$P(\Theta|D_C,D_{\mathcal{A}},\mathcal{F}) \propto P(D_C,D_{\mathcal{A}}|\Theta,\mathcal{F})P(\Theta|\mathcal{F}), \tag{4}$$

we can replace both integrals with a single term:

$$P(D_C|D_{\mathcal{A}},\mathcal{F}) \approx \frac{P(D_C,D_{\mathcal{A}}|\hat{\Theta},\mathcal{F})}{P(D_{\mathcal{A}}|\hat{\Theta},\mathcal{F})} = P(D_C|D_{\mathcal{A}},\hat{\Theta},\mathcal{F}).$$

It is important to note that the model $\hat{\Theta}$ used in this approximation should be the MAP model maximizing the posterior (4), not the maximum likelihood model maximizing the likelihood $P(D_C|D_{\mathcal{A}},\Theta,\mathcal{F})$, as is easily assumed by the form of the criterion.

The derived criterion can be computed efficiently for several model families of practical importance, including the Bayesian network model family with local distribution functions from the exponential family (see, e.g., (Heckerman, Geiger, & Chickering 1995)). In particular, in the Naive Bayes case the criterion can be computed in $O(N|\mathcal{F}|)$ time by using

$$S(\mathcal{F}) \approx P(D_C|D_{\mathcal{A}},\hat{\Theta},\mathcal{F}) = \prod_{j=1}^{N} P(c_j|\mathbf{d}_j,\hat{\Theta},\mathcal{F})$$
$$= \prod_{j=1}^{N} \frac{P(c_j,\mathbf{d}_j|\hat{\Theta},\mathcal{F})}{P(\mathbf{d}_j|\hat{\Theta},\mathcal{F})} = \prod_{j=1}^{N} \frac{P(c_j,\mathbf{d}_j|\hat{\Theta},\mathcal{F})}{\sum_{k=1}^{K} P(C=k,\mathbf{d}_j|\hat{\Theta},\mathcal{F})},$$

where $P(c_j,\mathbf{d}_j|\hat{\Theta},\mathcal{F})$ can be computed by using formula (1) with index i going through attributes in $\mathcal{F}$.

The BAYDA software

The Bayesian classification and feature selection approach described above is implemented in the BAYDA (Bayesian Discriminant Analysis) software programmed in the Java language. BAYDA has several unique features that distinguish it from standard statistical software for classification (such as predictive discriminant analysis (Huberty 1994)), or from machine learning classifier packages. First of all, for classification BAYDA uses a "fully Bayesian" Naive Bayes classifier with the marginal likelihood (evidence) predictive distribution (2). As demonstrated in (Kontkanen *et al.* 1997b; 1997c), using model parameter averaging improves classification performance substantially, especially with small samples. Second, BAYDA combines both manual and automatic feature selection. For automated feature selection the current version of BAYDA supports only forward selection and backward elimination (as described in, e.g., (Kohavi & John 1997)), but the future releases will add stochastic greedy methods and simulated annealing as search engines. Third, BAYDA graphical interface is built using the "intelligent document" paradigm: the user interface is embedded into an HTML-document, and for each classification task performed, BAYDA produces the corresponding adapted HTML-report explaining the results of the analysis. These results include the leave-one-out crossvalidated estimate of the overall classification accuracy, and the accuracy of the prediction by classes, both in graphical and textual format. In addition, the predictions for each individual data vector are also available. All the results can be stored in HTML-format.

BAYDA is available free of charge for research and teaching purposes from the CoSCo group home page[1], and it is currently tested on Windows'95/NT, SunOS and Linux platforms. However, being implemented in 100% Java, it should be executable on all platforms supporting Java Runtime Environment 1.1.3 or later.

Empirical results

Although elaborate empirical techniques, such as k-fold crossvalidation (Stone 1974), has been developed for validating different prediction methods, making a fair comparison of experimental results is difficult. One of the reasons for this is that the result of a single k-fold crossvalidation run is highly sensitive to the way the data is partitioned into the k folds, as demonstrated in (Tirri 1997). A

[1]http://www.cs.Helsinki.FI/research/cosco/

more objective comparison is possible if one uses an average of several independent crossvalidation runs (with different data partitionings), or the *leave-one-out crossvalidation* scheme. Here we have chosen the latter alternative, and emphasize that the results reported here are pessimistic in the sense that they could be easily "boosted up" by several percentage units by using k-fold crossvalidation with a single, conveniently chosen data partitioning. For this reason, we argue that our 0/1-score classification results, being better than those reported in, e.g., (Kohavi & John 1997; Friedman, Geiger, & Goldszmidt 1997), compare quite favorably to other empirical results.

The results of the first set of experiments can be found in Table 1. These tests are of the "brute-force" type, as in all cases the number of attributes was small enough to allow us to go through all the possible feature subsets. After the name of each dataset is the corresponding number of attributes, and the leave-one-out crossvalidated log-score and 0/1-score results obtained by using all the attributes. The log-score is obtained by computing minus the logarithm of the probability given to the correct class (thus the smaller the score, the better the result). It should be noted that although the simple 0/1-score is a more commonly used measure for classification accuracy, the log-score is an important measure from the decision-theoretic point of view: the Bayesian approach produces predictive distributions that can be used in decision-theoretic risk analysis of the outcomes of the actions based on our predictions, and the log-score is a natural measure for the accuracy of such predictive distributions.

Dataset	m	log	0/1	$\lvert\mathcal{F}\rvert$	log	0/1
Australian	14	0.46	84.78	6	0.34	86.83
Breast C.	9	0.64	72.03	7	0.59	73.78
Diabetes	8	0.56	75.91	4	0.47	76.56
Glass	9	0.98	66.82	7	0.93	66.82
Heart D.	13	0.44	83.70	10	0.38	82.96
Iris	4	0.13	94.00	2	0.10	98.00
Lymphog.	18	0.44	85.81	12	0.33	84.46

Table 1: The UCI datasets used and the leave-one-out crossvalidated results. Here m denotes the total number of attributes, and $\lvert\mathcal{F}\rvert$ the size of the feature subset maximizing the Bayesian feature selection criterion.

From Table 1 we can see that as a result of the Bayesian feature selection scheme, in all the cases the number of attributes has decreased a considerable amount (in some cases by more than 50%) from the original value, while the predictive accuracy has never decreased a significant amount. As a matter of fact, in the log-score sense the results are *consistently better* with the pruned attribute sets than with all the attributes, and in the 0/1-score sense feature selection usually improves the results too. This means that the Bayesian feature selection criterion chooses features that can indeed be said to be relevant with respect to the classification problem in question.

A more detailed analysis showed that the Bayesian feature selection criterion correlates extremely well with the prediction accuracy. The general trend with all the datasets is that the higher the value of the feature selection criterion, the better the prediction accuracy. This leaves us with the following question: in cases where there are too many attribute subsets for exhaustively searching through all the possibilities, is it possible to find suitable subsets through some search algorithm, or are the search spaces too complex for finding good attribute subsets in feasible time? For studying this question, we chose two additional datasets with a large number of attributes. The first dataset is a real-world educational dataset with 60 attributes, and the other the DNA dataset from the UCI repository, with 180 attributes. In this set of experiments, we used two greedy search methods: forward selection and backward elimination. The results with the educational data can be found in Table 2. From this table we can see that both algorithms have reduced the number of attributes approximately to one half. Furthermore, the predictive accuracy has improved dramatically. Both in terms of the criterion and the predictive accuracy, the differences between the search algorithms are negligible. In Table 3 are

Attributes	$\lvert\mathcal{F}\rvert$	$\log S(\mathcal{F})$	log	0/1
All	60	-1890.59	0.825	0.865
Forward sel.	22	-480.63	0.213	0.921
Backward elim.	28	-463.94	0.209	0.921

Table 2: Results with the educational data.

the results with the DNA data. The conclusions for this case are similar to the previous one, except that the number of selected attributes varies greatly from the 59 obtained by forward selection, to 125 produced by backward elimination. This result is not surprising since forward selection starts with an empty subset of attributes, and is hence more likely to get stuck in a local minima with less attributes than backward elimination, which starts with all the attributes.

Attributes	$\lvert\mathcal{F}\rvert$	$\log S(\mathcal{F})$	log	0/1
All	180	-376.14	0.214	0.941
Forward sel.	59	-194.20	0.106	0.971
Backward elim.	125	-199.23	0.115	0.967

Table 3: Results with the DNA data.

Conclusion

In this paper we have addressed the problem of classification and feature selection in predictive data mining tasks, where feature relevance is naturally defined by using prediction accuracy. Our approach, implemented in a software package BAYDA, was motivated by the fact that the models used for predictive data mining make independence assumptions of various degrees, the commonly used Naive Bayes classifier being an extreme case, and these assumptions are frequently violated in practice. Therefore feature subset is not only useful for data reduction, but it can also be expected to improve the classifier performance, if the features violating these assumptions are discarded.

In the theoretical part of the paper, we discussed a Bayesian criterion for feature selection. This criterion is based on the supervised marginal likelihood of the class value vector, given the rest of the data. Although the exact criteria is computationally intractable, we introduced an efficient approximation that can be used in practice. In the empirical part of the paper we used several real-world data sets, and demonstrated that the Bayesian feature selection scheme can dramatically decrease the number of features, while at the same time the predictive accuracy of the original Naive Bayes classifier can be improved. These results show that in addition of producing well-performing classifiers, BAYDA software offers a useful tool for the analysis of relevant features in classification domains.

Acknowledgments. This research has been supported by the Technology Development Center (TEKES), and by the Academy of Finland.

References

Blum, A., and Langley, P. 1997. Selection of relevant features and examples in machine learning. *Artificial intelligence* 97:245–271.

Cheeseman, P., and Stutz, J. 1996. Bayesian classification (AutoClass): Theory and results. In Fayyad, U.; Piatetsky-Shapiro, G.; Smyth, P.; and Uthurusamy, R., eds., *Advances in Knowledge Discovery and Data Mining*. Menlo Park: AAAI Press. chapter 6.

Chickering, D., and Heckerman, D. 1997. Efficient approximations for the marginal likelihood of Bayesian networks with hidden variables. *Machine Learning* 29(2/3):181–212.

Cooper, G., and Herskovits, E. 1992. A Bayesian method for the induction of probabilistic networks from data. *Machine Learning* 9:309–347.

Duda, R., and Hart, P. 1973. *Pattern classification and scene analysis*. John Wiley.

Friedman, N.; Geiger, D.; and Goldszmidt, M. 1997. Bayesian network classifiers. *Machine Learning* 29:131–163.

Geman, S.; Bienenstock, E.; and Wilson, R. 1992. Neural networks and the bias/variance dilemma. *Neural Computation* 4:1–48.

Heckerman, D.; Geiger, D.; and Chickering, D. 1995. Learning Bayesian networks: The combination of knowledge and statistical data. *Machine Learning* 20(3):197–243.

Huberty, C. 1994. *Applied Discriminant Analysis*. John Wiley & Sons.

John, G.; Kohavi, R.; and Pfleger, P. 1994. Irrelevant features and the subset selection problem. In *Machine Learning: Proceedings of the Eleventh International Conference*, 121–129. Morgan Kaufmann Publishers.

Kohavi, R., and John, G. 1997. Wrappers for feature subset selection. *Artificial Intelligence* 97:273–324.

Kohavi, R., and Wolpert, D. 1996. Bias plus variance decomposition for zero-one loss functions. In Saitta, L., ed., *Machine Learning: Proceedings of the Thirteenth International Conference*, 275–283. Morgan Kaufmann Publishers.

Kontkanen, P.; Myllymäki, P.; Silander, T.; and Tirri, H. 1997a. On the accuracy of stochastic complexity approximations. In *Proceedings of the Causal Models and Statistical Learning Seminar*, 103–117.

Kontkanen, P.; Myllymäki, P.; Silander, T.; Tirri, H.; and Grünwald, P. 1997b. Comparing predictive inference methods for discrete domains. In *Proceedings of the Sixth International Workshop on Artificial Intelligence and Statistics*, 311–318.

Kontkanen, P.; Myllymäki, P.; Silander, T.; Tirri, H.; and Grünwald, P. 1997c. On predictive distributions and Bayesian networks. In Daelemans, W.; Flach, P.; and van den Bosch, A., eds., *Proceedings of the Seventh Belgian-Dutch Conference on Machine Learning (BeNeLearn'97)*, 59–68.

Kontkanen, P.; Myllymäki, P.; Silander, T.; Tirri, H.; and Grünwald, P. 1998. Bayesian and information-theoretic priors for Bayesian network parameters. In Nédellec, C., and Rouveirol, C., eds., *Machine Learning: ECML-98, Proceedings of the 10th European Conference*, Lecture Notes in Artificial Intelligence, Vol. 1398. Springer-Verlag. 89–94.

Kontkanen, P.; Myllymäki, P.; and Tirri, H. 1996. Comparing Bayesian model class selection criteria by discrete finite mixtures. In Dowe, D.; Korb, K.; and Oliver, J., eds., *Information, Statistics and Induction in Science*, 364–374. Proceedings of the ISIS'96 Conference, Melbourne, Australia: World Scientific, Singapore.

Langley, P., and Sage, S. 1994. Induction of selective Bayesian classifiers. In *Proceedings of the Tenth Conference on Uncertainty in Artificial Intelligence*, 399–406. Seattle, Oregon: Morgan Kaufmann Publishers, San Francisco, CA.

Langley, P.; Iba, W.; and Thompson, K. 1992. An analysis of Bayesian classifiers. In *Proceedings of the 10th National Conference on Artificial Intelligence*, 223–228. San Jose, CA: MIT Press.

Stone, M. 1974. Cross-validatory choice and assessment of statistical predictions. *Journal of the Royal Statistical Society (Series B)* 36:111–147.

Tirri, H.; Kontkanen, P.; and Myllymäki, P. 1996. Probabilistic instance-based learning. In Saitta, L., ed., *Machine Learning: Proceedings of the Thirteenth International Conference*, 507–515. Morgan Kaufmann Publishers.

Tirri, H. 1997. *Plausible Prediction by Bayesian Inference*. Ph.D. Dissertation, Report A-1997-1, Department of Computer Science, University of Helsinki.

Weiss, S., and Indurkhya, N. 1998. *Predictive Data Mining*. Morgan Kaufmann Publishers.