RC23602 (W0505-053) May 6, 2005
Computer Science

IBM Research Report

Kikuchi-Bayes: Markov Networks for Classification

Aleks Jakulin
Jozef Stefan Institute
Jamova 39
SI-1000 Ljubljana
Slovenia

Irina Rish
IBM Research Division
Thomas J. Watson Research Center
P.O. Box 704
Yorktown Heights, NY 10598

Research Division
Almaden - Austin - Beijing - Haifa - India - T. J. Watson - Tokyo - Zurich

Kikuchi-Bayes: Markov Networks for Classification

Aleks Jakulin
Jožef Stefan Institute
Jamova 39, SI-1000 Ljubljana, Slovenia

Irina Rish
IBM T.J. Watson Research Center
19 Skyline Drive, Hawthorne, NY 10532, USA

Abstract

We propose a simple and efficient approach to building undirected probabilistic classification models (Markov networks) that extend the naïve Bayes classifiers and outperform the existing directed probabilistic classifiers of similar complexity (e.g. Bayesian network with same cluster size). The models are represented as sets of cliques, not necessarily maximal, and the probability density functions can be estimated in closed form that mirrors the cluster variation method (Kikuchi approximation). We employ a highly efficient Bayesian learning algorithm, based on integrating along a hill-climb in the structure space. We present promising empirical results on 46 benchmarks.

1 INTRODUCTION

In this paper, we focus on the problem of building probabilistic classifiers from data. It is often an ill-defined problem to learn the 'true' global probabilistic model from a limited amount of data, especially in high-dimensional domains. Even with potentially unlimited data, the model complexity might be intractable. Therefore, various approximation techniques are used. A common approach is to use 'local' models which limit the complexity of probabilistic functions used to describe the model, effectively making certain independence assumptions: for example, naïve Bayes only uses class-conditional distributions $P(X_i|Y)$ for each attribute X_i, tree-augmented naïve Bayes extends it to pairwise class-conditional dependencies such as $P(X_i|X_j, Y)$, and Bayesian network classifiers learn a collections of conditional distributions $P(X_i|\Pi_{X_i}, Y)$ where Π_{X_i} is some limited-size subset of attributes.

Another approach is to seek a joint probability mass function (PMF) consistent with a set of marginal probability mass functions on subsets of attributes $P(X_1, \ldots, X_k, Y)$ that act as constraints (Ireland & Kullback, 1968). There are many such joint functions, and we pick the maximum entropy one, meaning that no information is assumed by the joint PMF $P(\mathbf{X}, Y)$ beyond what is given by the constraints. Unfortunately, computationally expensive iterative methods are usually required to identify the global PMF.

Note that the approaches mentioned above aim at learning a joint probability distribution in an explicit (normalized) form, which is not really necessary (as we show later) for estimating the class predictive probability $P(Y|\mathbf{X})$. Herein, we propose a simple approach that only learns a collection of marginal distributions over subsets of variables, which corresponds to an undirected model – Markov network – and an implicit (non-normalized) representation of a joint distribution. While general inference in Markov networks is hard, computing predictive class probabilities $P(Y|\mathbf{X})$ is easy given the observed $\mathbf{X} = \mathbf{x}$.

Our approach is based on the *cluster variation method* originally proposed as a way of approximating free energy of a complex system from 'local' energies over subsets of variables (Kikuchi, 1951) (so-called *Kikuchi approximations*). There is a close link between approximating free energy of a system and approximating the corresponding probability distribution, namely, minimizing KL-divergence between the true and approximate distribution is equivalent to minimizing the corresponding Kikuchi approximation to free energy[1]. This link explains recent popularity of Kikuchi approximations within the probabilistic inference community.

In this paper, we will focus on the tasks of predictive class probability estimation and classification, where the query attribute is the label, and all the other at-

[1]It is also shown that the fixed points of *(generalized) belief propagation*, a popular approximate inference algorithm, are at the stationary points of the Kikuchi free energy (Yedidia et al., 2004).

tributes are the evidence. We will not use the cluster variation method to model the free energy, but instead to model the global PMF directly. We will employ a parsimonious Bayesian prior to select the set of regions (subsets of variables) with the maximum posterior probability, without restricting ourselves to hierarchical, graphical or decomposable probabilistic models. However, we will treat structure as a variable, and integrate it out when performing prediction, using a highly efficient algorithm that combines posterior sampling with lookahead-enabled heuristic search.

2 RELATED WORK

Learning Markov networks from data has a long history. Typically, it focuses on learning bounded-treewidth models, since general inference in such models is exponential in their treewidth. For example, Chow and Liu (1968) proposed a simple algorithm for learning optimal tree distributions, and Srebro (2001) generalized this approach to hypertrees, i.e. Markov networks of bounded treewidth. Projecting a distribution on a Markov network of bounded treewidth (i.e. finding such network that is closest to the true distribution in terms of KL-divergence) is shown to reduce to finding a triangulated graph that maximizes the total sum of weights over its cliques with respect to some monotone weight function. Unfortunately, learning optimal hypertrees is an NP-hard problem, but there are approximation algorithms with provable performance guarantees (Srebro, 2001).

Note, however, that in case of classification with a model $\mathcal{M}$, we are not concerned with bounding the treewidth of the model as we only use it for computing $P(y|\mathbf{x}, \mathcal{M})$, and this can be done using non-normalized products of potentials that approximate the true model (as we show below). Thus, we are only concerned with bounding the clique size in the original (non-triangulated) network, due to computational complexity of our structure search being exponential in the maximum clique size. The advantage of not learning an explicit distribution and settling for an non-normalized product of potentials enables us to account for more k-way interactions between the variables than the corresponding triangulated model would do, keeping the treewidth fixed.

Another related approach is learning max-margin Markov networks for classification (Taskar et al., 2003). However, that work was focused on the interactions among the labels rather than among the features, e.g. in sequential and spatial data where there are strong correlations between the labels and the usual i.i.d. assumption does not capture the problem's structure. Our goal is different as we consider the standard classification task but wish to learn the structure of the Markov network over the attributes and the class, i.e. select an appropriate subset of clusters (which also can be viewed as a feature selection problem in exponential-family models).

3 NOTATION AND DEFINITIONS

Let $\mathbf{X} = \{X_1, \ldots, X_n\}$ be a set of observed random variables, called *attributes*, and let $\mathbf{x} = (x_1, \ldots, x_n)$ be a vector of values assigned to the variables in $\mathbf{X}$. Herein, we assume discrete-valued attributes, i.e. $\mathbf{x} \in \mathcal{X} = \mathcal{X}_1 \times \ldots \times \mathcal{X}_n$ where each range $\mathcal{X}_i$ is a set of possible values of X_i. Let Y denote an unobserved random variable called the *class*, where $y \in \mathcal{Y}, |\mathcal{Y}| = m$. The set of attributes together with the class (i.e., all variables) is denoted $\mathbf{V} = \mathbf{X} \cup \{Y\}$. An assignment $\mathbf{v} = (\mathbf{x}, y)$ of values to the attributes and the class is called an *instance*, or *example*. We will use a short notation $P(\mathbf{v}) = P(\mathbf{x}, y) = P(x_1, \ldots, x_n, y)$ to describe the joint probability distribution $P(X_1 = x_1, \ldots, X_n = x_n, Y = y)$. A subset of variables $R \in \mathbf{V}$ is called a *region* (or *cluster*), and a value assignment to R is denoted $\mathbf{v}_R$.

A *classifier* is a mapping $h : \mathcal{X} \to \mathcal{Y}$ that assigns a class value to any given instance. In particular, the *Bayes classifier* $h^*(\mathbf{x}) = \arg\max_y P(y|\mathbf{x}) = \arg\max_y P(\mathbf{x}, y)$ selects the most-likely value of class given the observed attributes, and is provably optimal (i.e. has the lowest error probability, or lowest *risk*, among all classifiers). However, in practice, the true underlying distribution $P(\mathbf{x}, y)$ (or, respectively, $P(y|\mathbf{x})$) is not available, or is hard to estimate from limited data.

A common approach to this problem is to assume a certain simplified family $\mathcal{M}$ of joint probability mass functions $P(y, \mathbf{x}|\mathcal{M}) = \hat{P}(y, \mathbf{x})$ that approximate $P(y, \mathbf{x})$. For example, one of the simplest and perhaps most popular probabilistic classifiers is the *naïve Bayes* that assumes attribute independence given the class, thus approximating $P(\mathbf{x}, y)$ by $\hat{P}(\mathbf{x}, y) = \prod_i P(x_i|y)P(y)$. Other approaches include less restrictive assumptions on the structure of $\hat{P}(\mathbf{V})$, such as trees (e.g., Tree-Augmented Naïve Bayes (TAN) or, more generally, Bayesian networks (Friedman et al., 1997). We will use undirected graphical models such as Markov networks. A *Markov network*, or *Markov random field* on random variables $\mathbf{X}$ is defined as $\langle \mathcal{G}, \mathcal{S} \rangle$ where $\mathcal{G}$ is an undirected graph and $\mathcal{S} = (\Phi_1, \ldots, \Phi_m)$ is a set of positive functions, called *potentials* for each of the m cliques in $\mathcal{G}$, such that the joint distribution $P(\mathbf{x})$ factorizes over them: $P(\mathbf{x}) = (1/Z) \prod_i \Phi(\mathbf{x}_i)$ where Z is a normalization constant.[2]

[2]Without loss of generality we could restrict the set of all cliques to the set of all *maximal* cliques.

4 MAIN IDEA

Our approach is motivated by the following simple observation:

Lemma 1 *Given a set of random variables* $\mathbf{V} = \mathbf{X} \cup \{Y\}$*, a set* $\mathcal{R} = \{R|R \subseteq \mathbf{V}\}$ *of subsets (regions) of* $\mathbf{V}$*, where* Y *belongs to at least one region, and a product* $\Phi(\mathbf{v}) = \Phi(\mathbf{x}, y) = \prod_{R\in\mathcal{R}} \Phi_R(\mathbf{v}_R)$ *of non-negative functions (potentials) defined on these regions, let* $\hat{P}(\mathbf{v}) = (1/Z)\Phi(\mathbf{v})$ *be the corresponding joint probability distribution over* $\mathbf{V}$*, where* Z *is a normalization constant. Then:*

1. *Computing* $\hat{P}(Y|\mathbf{x})$ *does not require global normalization, i.e.* $\hat{P}(Y|\mathbf{x}) = \Phi(\mathbf{x}, Y)/\sum_{y'} \Phi(\mathbf{x}, y')$*;*
2. *Bayesian classifier can be computed using a product of only those potentials that contain* Y*, i.e.* $h^*(\mathbf{x}) = \arg\max_y \prod_{\{R\in\mathcal{R}|Y\in R\}} \Phi_R(\mathbf{v}_R)$.

Proof: The first claim follows from $\hat{P}(y|\mathbf{x}) = \hat{P}(\mathbf{x}, y)/\hat{P}(\mathbf{x}) = (1/Z)\Phi(\mathbf{x}, y)/\sum_{y'}(1/Z)\Phi(\mathbf{x}, y')$, since by definition $\Phi(\mathbf{v}) = \Phi(\mathbf{x}, y)$. The second claim is easily obtained from the definition of Bayesian classifier, $h^*(\mathbf{x}) = \arg\max_y \hat{P}(y|\mathbf{x})$, and the following observation:

$$\hat{P}(y|\mathbf{x}) = \frac{\Phi(\mathbf{x}, y)}{\sum_{y'} \Phi(\mathbf{x}, y')} = \frac{\prod_{\{Q\in\mathcal{R}|Y\notin Q\}} \Phi(\mathbf{v}_Q)}{\sum_{y'} \Phi(\mathbf{x}, y)} \prod_{\{R\in\mathcal{R}|Y\in R\}} \Phi_R(\mathbf{v}_R),$$

where $(\prod_{\{Q\in\mathcal{R}|Y\notin Q\}} \Phi(\mathbf{v}_Q))/\sum_{y'} \Phi(\mathbf{x}, y)$ is independent of Y. ■

This observation tells us that a non-normalized representation of a joint distribution over a Markov network can be used for a specific inference problem of computing the predictive class probability $P(y|\mathbf{x})$.

Note that typical probabilistic classifiers (e.g., naïve Bayes, TAN, Bayesian networks) build an *explicit* probabilistic model $\hat{P}(y, \mathbf{x})$ by assuming a certain structure of the probability distribution (e.g., a tree-structure in TAN, or a particular factorization of $\hat{P}(y, \mathbf{x})$ according to the Bayesian network structure). However, as shown above, learning an explicit (normalized) probabilistic model is not needed if our objective is only to estimate the predictive class probability $P(y|\mathbf{x})$, and may potentially introduce unnecessary constraints on the set of interactions we wish to include in the model.

In this paper, we propose an alternative approach that models $\hat{P}(y, \mathbf{x})$ *implicitly* by using a collection of marginal distributions defined over (potentially all) subsets, or clusters, of the variables (clearly, the subset size is limited to a reasonable value to make the approach tractable). The marginal distribution for a particular subset of variables R is $P(\mathbf{v}_R)$ that is estimated directly from the data, is referred to as the *submodel*. The set of such clusters corresponds to an undirected graphical model, i.e. a Markov network. The main advantage of our approach is that it allows to take into account potentially any subset of k-way interactions instead of limiting ourselves to interactions consistent with the DAG structure in Bayesian network learning. We are also not restricting the models to bounded-treewidth Markov networks as done in (Srebro, 2001). Thus, our approach imposes no a priori constraints on the structure, other than that the structure is specified in terms of subsets of variables.

For example, consider an $n \times n$ grid Markov network (e.g., Ising model). It is well-known that its treewidth equals n, so a bounded-treewidth model with bound $k < n$ would have to ignore many pairwise interactions, while an non-normalized model with the bound of 2 on the (initial) clique size could include all of them. While none of the approaches is an absolute winner in all cases, empirical results demonstrate that the ability to incorporate more interactions while staying tractable is a clear advantage of our approach.

Given the factorization of variables into the set of clusters, the question is how to reconstruct the joint distribution $P(\mathbf{v})$ only knowing the set of submodels. Our approach to region selection is inspired by the *cluster variation method* (CVM) (Yedidia et al., 2004), also known as *Kikuchi approximation* of free energy (Kikuchi, 1951). The cluster variation technique is used only for representation of the joint probability distribution, not for performing inference. The issue of interaction selection is handled later in the paper. An overview of our approach is given below:

Kikuchi-Bayes classification algorithm:

1. Given $\mathbf{Y} = \mathbf{X} \cup \{y\}$, and a bound k on region size, select the set of interactions $\mathcal{M} = \{M|M \subseteq \mathbf{Y}\}$ using the approach described in Sect. 6.
2. Given $\mathcal{M}$, compute an extended set of regions and their counting numbers forming a region graph $\mathcal{R}$ using the *cluster variation method* where each interaction corresponds to an initial region (see Sect. 5). For each region R estimate the submodel $P(\mathbf{v}_R)$ from data.
3. Approximate $P(\mathbf{v})$ by the (non-normalized) product $\Phi(\mathbf{v}) = \prod_{\langle R, c_R\rangle\in\mathcal{R}} P(\mathbf{v}_R)^{c_R}$ where c_R is the *counting number* for region R.
4. Compute $\hat{P}(y|\mathbf{x}) = \Phi(\mathbf{x}, y)/\sum_{y'} \Phi(\mathbf{x}, y')$ and

classify $y^*(\mathbf{x}) = \arg\max_y P(y|\mathbf{x})$.

5 KIKUCHI APPROXIMATION TO PROBABILITY DISTRIBUTIONS

We will now elaborate on the second step of the Kikuchi-Bayes algorithm. Let us consider a problem of approximating a joint PMF $P(\mathbf{V})$ using its marginals over subsets of $n+1$ random variables $\mathbf{V} = \{X_1, X_2, \ldots, X_n, Y\}$. Given the set of interactions $\mathcal{M} = \{R_1, R_2, \ldots, R_\ell\}$ in the set $\mathbf{V}$ and the submodel for each region, $P_R = P(\mathbf{v}_R) = P(v_{R,1}, v_{R,2}, \ldots, v_{R,k})$, we will find an (non-normalized) approximation $\Phi_{\mathcal{M}}(\mathbf{v})$ of the intractable $P(\mathbf{y})$ using the set of $\{P(\mathbf{v}_R); R \in \mathcal{M}\}$.

Our approach to the joint PMF approximation is inspired by the *cluster variation method* (CVM) (Yedidia et al., 2004), also known as the Kikuchi approximation of free energy (Kikuchi, 1951). We apply the cluster variation method (Yedidia et al., 2004) to the learned set of interactions $\mathcal{M}$ to obtain a *region graph*. The region graph includes the interactions as initial regions, their intersections, intersections of intersections, and so on. For each region R, there is a corresponding *counting number* c_R, that accounts for the region overlaps and avoids double-counting when using the *region-based approximation* of the *free energy* (Yedidia et al., 2004). The details of cluster variation algorithm are described as Algorithm 1.

The free energy is then defined as $F_{\mathcal{R}} = U_{\mathcal{R}} - H_{\mathcal{R}}$, where $U_{\mathcal{R}}$ and $H_{\mathcal{R}}$ are the region-based approximations of the average energy and the entropy, respectively, and are given by: $U_{\mathcal{R}} = \sum_{R\in\mathcal{R}} c_R U_r(b_R)$, and $H_{\mathcal{R}} = \sum_{R\in\mathcal{R}} c_R H_R(b_R)$, where b_R is some marginal probability distribution over R, $U_R(b_R) = \sum_{\mathbf{y}_R} b_R(\mathbf{y}_R) E_R(\mathbf{y}_R)$ is the average energy, and $H_R(b_R) = \sum_{\mathbf{y}_R} b_R(\mathbf{y}_R) \ln b_R(\mathbf{y}_R)$ is the entropy of a region, respectively (Yedidia et al., 2004). Region-based approximation using CVM is considered a good approximation to the (intractable) true free energy, because it accounts for the overlaps between the regions.

Although the region graph is defined as a directed graph where the nodes are regions, and the links represent the region inclusion relationships (Yedidia et al., 2004), we will represent the region graph merely as a set of pairs $\mathcal{R} = \{\langle R, c_R\rangle, R \subseteq \mathbf{V}\}$, keeping the connectivity structure implicit. Generalizing the Kirkwood superposition approximation (Jakulin & Bratko, 2004), the joint *Kikuchi approximation* is then defined as:

$$P(\mathbf{v}) \triangleq \frac{1}{Z}\Phi_{\mathcal{M}}(\mathbf{v}), \;\; \Phi_{\mathcal{M}}(\mathbf{v}) \triangleq \prod_{\langle R, c_R\rangle \in \mathcal{R}} P(\mathbf{v}_R)^{c_R} \quad (1)$$

$\mathcal{R}_0 \leftarrow \{\emptyset\}$ {Redundancy-free set of interactions.}
for all $\mathcal{S} \in \mathcal{M}$ **do** {for each initial region}
  if $\forall \mathcal{S}' \in \mathcal{R}_0 : \mathcal{S} \not\subseteq \mathcal{S}'$ **then**
    $\mathcal{R}_0 \leftarrow \mathcal{R}_0 \cup \{\mathcal{S}\}$ {$\mathcal{S}$ is not redundant}
  end if
end for
$\mathcal{R} \leftarrow \{\langle \mathcal{S}, 1\rangle;\ \mathcal{S} \in \mathcal{R}_0\}$
$k \leftarrow 1$
while $|\mathcal{R}_{k-1}| > 2$ **do** {there are feasible subsets}
  $\mathcal{R}_k \leftarrow \{\emptyset\}$
  for all $\mathcal{I} = \mathcal{S}^\dagger \cap \mathcal{S}^\ddagger : \mathcal{S}^\dagger, \mathcal{S}^\ddagger \in \mathcal{R}_{k-1}, \mathcal{I} \notin \mathcal{R}_k$ **do** {feasible intersections}
    $c \leftarrow 1$ {the counting number}
    for all $\langle \mathcal{S}', c'\rangle \in \mathcal{R}, \mathcal{I} \subseteq \mathcal{S}'$ **do**
      $c \leftarrow c - c'$ {consider the counting numbers of all regions containing the intersection}
    end for
    $\mathcal{R} \leftarrow \mathcal{R} \cup \{\langle \mathcal{I}, c\rangle\}$
    $\mathcal{R}_k \leftarrow \mathcal{R}_k \cup \{\mathcal{I}\}$
  end for
end while
return $\{\langle R, c\rangle \in \mathcal{R}; c \neq 0\}$ {Region graph.}

Algorithm 1: Cluster variation method for constructing the region graph given the set of interactions $\mathcal{M} = \{\mathcal{S}_1, \mathcal{S}_2, \ldots, \mathcal{S}_\ell\}$.

The submodel over the attributes $\mathbf{V}_R$ is $P(\mathbf{v}_R)$, and Z is the normalization constant. Each $P(\mathbf{v}_R)$ can be computed by marginalization of some $P(\mathbf{v}_S), R \subseteq S$. It is also easy to show that the approximation in (1) is equivalent to the following definition of the joint PMF that uses a recursive definition for potentials ϕ_R:

$$P(\mathbf{v}) \triangleq \frac{1}{Z}\prod_{R\in\mathcal{R}} \phi_R(\mathbf{v}_R)$$
$$\phi_R(\mathbf{v}_R) \triangleq \frac{P(\mathbf{v}_R)}{\prod_{R'\subset R} \phi_{R'}(\mathbf{v}_{R'})} \quad (2)$$

The set of regions (clusters) $\mathcal{R}$ thus defines a Markov network. It is well-known (Pearl, 1988) that when the Markov network is triangulated and thus yields a clique tree, the distribution can be represented exactly through (1) (i.e., no normalization is needed), as $P(\mathbf{v}) = \prod_{R\in\mathcal{R}} \phi_R(\mathbf{v}_R)$, where the potentials $\phi_R(\mathbf{v}_R)$ are defined by (2). In general, when the counting numbers are greater than zero only for the initial regions, the Kikuchi approximation is exact (Yedidia et al., 2001).

When the resulting Markov network is not triangulated, the above set of regions nevertheless defines a so-called "valid approximation" to the free energy as a sum of weighted free energies over the regions (Yedidia et al., 2004); the weights (counting numbers) ensure that each potential function and each node will be counted exactly once in the approximation to the free energy. This motivated us to choose this approach to model approximation, although we do not always compute a set of regions that yields a triangulated graph.

6 MODEL LEARNING

In the present section we will deal with the choice of interactions or initial regions. We will adopt the Bayesian framework, based on an explicit description of the model in terms of its parameters $\phi = (\mathcal{M}, \mathbf{\Theta}, \vartheta)$. It is desirable that the structure $\mathcal{M}$ is independent of the submodel prior ϑ and the submodel parameters $\mathbf{\Theta}$. It would be paradoxical that some submodel $P(X_1, Y)$ changed because some new attribute X_2 was introduced into the model. Submodels and marginals of the joint should remain invariant given the overall model structure. Our goal is achieved by the following factorization of the prior $P(\phi) = P(\mathcal{M})P(\vartheta)P(\mathbf{\Theta}|\vartheta) = P(\mathcal{M})P(\vartheta)\prod_i P(\boldsymbol{\theta}_i|\vartheta)$. We will now address two additional constraints: first, the submodels must be a posteriori consistent in spite of the conditional independence of their parameters; second, the models should be parsimonious: as simple as possible but not simpler.

The final result of our inference based on data $\mathcal{D}$ will be the following class predictive distribution:

$$\hat{P}(y|\mathbf{x}) \propto \int P(\phi|\mathcal{D})\hat{P}(y|\mathbf{x}, \phi)d\phi \tag{3}$$

For prediction we thus integrate the model structure out (Buntine, 1991; Cerquides & López de Màntaras, 2003). In the following sections we will discuss our priors and our greedy algorithm for efficiently integrating in the space of ϕ. Still, the value of ϕ with the maximum a posteriori probability is interesting as the best individual model in the ensemble.

6.1 CONSISTENT SUBMODELS

The submodels have no specific ordering, and should be estimated independently from data $\mathcal{D}$. After the estimation, we work with posterior predictive distributions $P(\mathbf{v}_R|\mathcal{D})$, without referring back to their parameters. It is important, however, to assure that the predictive submodels are in fact consistent. Consistency means that there exist some global predictive distribution $P(\mathbf{v}|\mathcal{D})$ so that the submodels could be obtained from it by marginalization.

While maximum likelihood estimation would result in consistent submodels, Bayesian modelling requires some forethought. Namely, each submodel is modelled based on the same prior, but independently of other submodels, including those that overlap with it. Some popular choices of parameter priors, such as the Laplacean prior, would result in inconsistent submodel posteriors. Imagine estimating two entangled coins using the Laplacean prior. If a single coin c_1 is estimated independently, we will obtain the posterior predictive probability of $p_H = (1 + \#_{c_1=H})/(2 + \#)$. If we estimate two co-tossed coins simultaneously, and marginalize c_2 out, we obtain a non-matching

$$p_H = \frac{2 + \#(c_1 = H, c_2 = H) + \#(c_1 = H, c_2 = T)}{4 + \#}.$$

Let us now consider a submodel on attributes $\boldsymbol{X}_s = \{X_1, X_2, \ldots, X_k\}$. All the attributes are assumed to be nominal, and the multinomial submodel would be appropriate. The multinomial submodel is parameterized by the vector $\boldsymbol{\theta}_s$ whose dimensionality corresponds to the cardinality of $\prod_{i=1}^{k} |\mathcal{X}_i|$. A coordinate $\boldsymbol{\theta}_{s:x_1,\ldots,x_k}$ can be interpreted as the probability of occurrence of $(x_1, \ldots x_k)$. What we need is a prior $P(\boldsymbol{\theta}_s)$ that assures that the posterior predictive distribution $P(\boldsymbol{x}_s|\mathcal{D}) = \int P(\boldsymbol{\theta}_s|\mathcal{D})P(\boldsymbol{x}_s|\boldsymbol{\theta}_s)d\boldsymbol{\theta}_s$ will be consistent with all submodels that share attributes with $\boldsymbol{X}_s$.

It is quite easy to see that the following choice of the symmetric Dirichlet prior fulfills the demand of predictive consistency, if the same value of ϑ is used for all the submodels:

$$P(\boldsymbol{\theta}_s|\vartheta) = \text{Dirichlet}(\alpha, \ldots, \alpha), \quad \alpha = \frac{\vartheta}{\prod_{i=1}^{k} |\mathcal{X}_i|} \tag{4}$$

This prior is best understood as the expected number of outliers: to any data set, we add ϑ uniformly distributed instances. There is also an implied assumption of no structural zeros: not making such an assumption may result in zero likelihood of the test data.

6.2 PARSIMONIOUS STRUCTURES

The structure in the context of Kikuchi-Bayes is simply a selection of the submodels. $P(\mathcal{M})$ models our prior expectations about the structure of the model. Parsimony means that we should not select all the submodels, and the motivation for this is not just the subjective desire for simplicity but also the frequentist problem of objective identifiability and the decision-theoretic desire to minimize the expected loss. We will now provide a parsimonious prior that asserts a higher prior probability to simpler selections of submodels.

The primary question is how to quantify the complexity of the set of submodels. Neither the number of submodels nor the total number of parameters across the submodels in $\mathcal{M}$ would be sensible choices: some submodels describe attributes with a greater number of values, and some submodels may be partly contained within other submodels. An interesting quantification of complexity that solves this dilemma is given by Krippendorff (1986) in the context of loglinear models without structural zeros. Let us assume a set of overlapping submodels of the attribute vector $\mathbf{V}$, and the resulting region graph $\mathcal{R}$ obtained using the CVM.

The number of *degrees of freedom* of the joint model $\mathcal{M}$ with a corresponding region graph $\mathcal{R}$ is:

$$df_{\mathcal{M}} \triangleq \sum_{\langle \mathcal{S},c\rangle \in \mathcal{R}} c\Big(-1 + \prod_{X \in \mathcal{S}} |\mathcal{X}|\Big) \qquad (5)$$

The overlap between submodels is hence handled in an analogous way both for fusion in (1) and for the assessment of degrees of freedom.

The following prior corresponds to the assumption of exponentially decreasing prior probability of a structure with an increasing number of degrees of freedom (or effective parameters):

$$P(\mathcal{M}) \overset{\triangle}{\propto} \exp\left\{ -\frac{m\ df_{\mathcal{M}}}{m - df_{\mathcal{M}} - 1} \right\} \qquad (6)$$

We discourage the degrees of freedom from exceeding the number of training instances m. This choice of the prior has a frequentist justification: it corresponds to the Akaike information criterion (AIC) with small-sample correction (Burnham & Anderson, 2002). Performing MAP inference of the structure parameter $\mathcal{M}$ with such a prior would correspond to maximizing the AIC. Thus, our prior corresponds to the subjective choice of the frequentist paradigm along with a particular loss function. A Bayesian will make sure that the prior is properly normalized, of course.

6.3 THE PRIOR AND THE LIKELIHOOD FUNCTION FOR CLASSIFICATION

Our objective is predictive class probability estimation with the Kikuchi approximation (1). We need to define the prior on the structure variable $\mathcal{M}$ and the likelihood of $\mathcal{M}$ given the data. If $\mathcal{M}$ is going to be used for prediction, the effective degrees of freedom are fewer ("Conditional density estimation is easier than joint density estimation."). Assuming a single attribute X_i, the degrees of freedom of the conditional model $P(Y|X_i)$ correspond to the difference between the cardinality of the range of both Y and X_1 at once less the cardinality of the range of X_1 alone: $df_{Y|X_i} = |\mathcal{X}_i \times \mathcal{Y}| - |\mathcal{X}_1|$. In general, if we condition upon a subset of attributes $\mathcal{Y} \subseteq \mathcal{X}$, the degrees of freedom of the resulting conditional model will be defined as:

$$df_{\mathcal{M}_{\mathcal{Y}}} \triangleq \sum_{\langle \mathcal{S},c\rangle \in \mathcal{R}} c\left(\prod_{X \in \mathcal{S}} |\mathcal{X}| - \prod_{\substack{X \in \mathcal{S} \\ X \notin \mathcal{Y}}} |\mathcal{X}| \right) \qquad (7)$$

The prior $P(\mathcal{M}_{\mathcal{Y}})$ is obtained by plugging (7) into (6).

The growth of structures should be guided by whether the addition of a submodel is of benefit in predicting the label. The following conditional likelihood function takes this into account:

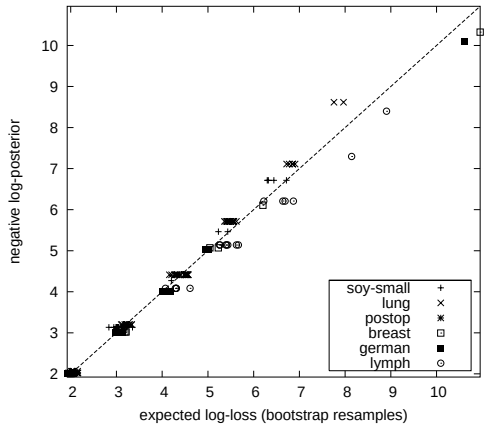

Figure 1: The logarithm of the parsimonious prior is well-aligned with the expected log-loss across bootstrap resamples in conditional density estimation.

$$\hat{P}(\mathbf{v}^{(1)\ldots(m)}|\mathcal{M}_{\mathcal{Y}}) \triangleq \prod_{i=1}^{m} \hat{P}(\mathbf{y}^{(i)}|\mathbf{x}^{(i)}, \mathcal{M}_{\mathcal{Y}}) \qquad (8)$$

Because $\mathcal{M}$ was assumed to be independent of ϑ and $\boldsymbol{\Theta}$, we prepare $\boldsymbol{\Theta}$ in advance, before assessing $\mathcal{M}$. The $\hat{P}(\mathbf{y}^{(i)}|\mathbf{x}^{(i)}, \mathcal{M})$ is obtained by applying the Bayes rule on a Kikuchi approximation. A different approach to submodel fusion is conceivable, e.g., based on replacing the Kikuchi approximation with a maximum entropy one.

We can examine the resulting conditional prior empirically on several smaller benchmark domains with the number of instances in the order of magnitude of 100 and discretized attributes: 'soybean-small', 'lung', 'post-op', 'lymphography', 'german credit' and 'breast-cancer'. We have compared the posterior log-likelihood of a particular model $\mathcal{M}$, $P(\mathbf{v}^{(1)\ldots(m)}|\mathcal{M}_{\mathcal{Y}}, \vartheta = 0)P(\mathcal{M}_{\mathcal{Y}})$ with the expected total log-loss of the maximum likelihood estimates on nonparametric bootstrap resamples $\mathcal{D}^*$, across many resamples: $\mathbb{E}_{\mathcal{D}^*}\{-\sum_{\mathbf{v}^{(i)} \in \mathcal{D}^*} \log P(y^{(i)}|\mathbf{x}^{(i)})\}$. The sampling zeros were assumed to be structural zeros, i.e., if a particular attribute-class combination did not appear in the training data, it was assumed to be impossible and did not count towards the *df* (Jakulin & Bratko, 2004). The result is shown in Fig. 1.

7 STRUCTURE SEARCH

Although integrating the structure out using (3) is theoretically simple, we need to sample in the space of $\mathcal{M}$. It is very expensive to exhaustively survey the whole lattice of possible structures. Even if we did that, we would not be able to explain what our model is. We will adopt a simpler approach: hull-climbing. We

will greedily ascend to the local maximum a posteriori structure by including the best individual region at each step, one that maximizes the posterior structure probability. Even once we get there, we keep descending for a while, as long as the structure's likelihood keeps increasing. On the termination of ascent, we *integrate out the stage of the path*. In other words, we perform Bayesian model averaging with respect to the *length* of the greedy path to the top and beyond.

This way, we obtain a compact depiction of the optimal hilltop (maximum a posteriori structure), the continuing of the path towards the dangerous peak (maximum likelihood structure). Integrating out the stage of the path prevents overconfidence in a particular structure and overfitting on the test data. Furthermore, the model average is essentially transparent and interpretable, as we can easily present the ordering of the regions as they were included into the model.

During the climb, we are guided by one-level lookahead (Buntine, 1991). This can be done efficiently with Kikuchi-Bayes using the tricks of Caruana et al. (2004): including a new region corresponds to just multiplying the approximate joint PMF with another term and renormalizing for each instance. With the considerable increase in performance that ensues, we can afford to find the best region at every step of the forward selection.

In addition to the look-ahead, we use the step-wise forward selection algorithm designed for loglinear models (Jobson, 1992). We first ascend by adding regions of size k attributes, and only when no further ascent is possible, we continue by ascending through addition of regions of size $k+1$ attributes. The purpose of this step-wise approach is both to increase the performance by decreasing the fanout in the search tree and to smooth the path. For example, we prevent immediately adding the interaction ABY if adding AY and BY is just as good. Still, we grow models faster than we would by only attempting unitary increases in their degrees of freedom: we skip forward by adding whole regions. In all, other search algorithms could also be used, especially stochastic ones, but we should be careful as counting the same model structure multiple times would interfere with the prior. An example of such a search is shown in Fig. 2.

It must be noted that adding a region in the context of Kikuchi-Bayes may sometimes reduce the model's likelihood, not just its posterior probability. We refer to this as *approximation error* due to the use of suboptimal Kikuchi approximation. There would be no joint approximation error had we used MaxEnt instead or if we only used the maximal cliques of the Markov network as initial regions.

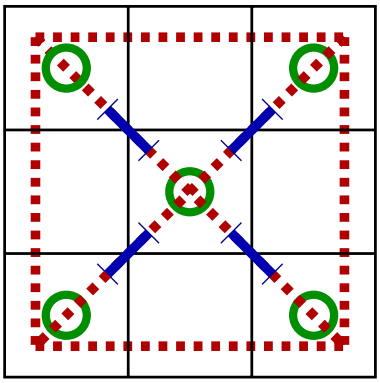

Figure 2: Interactions of size 4 are not merely a theoretical curiosity. In this illustration we show the Tic-Tac-Toe game board, which comprises 9 squares, each corresponding to a 3-valued attribute with the range $\{\times, \circ, \_\}$. The goal is to develop a predictive model that will indicate if a board position is winning for $\times$ or not: this is the 2-valued class attribute. The illustration shows the interactions in the MAP model identified by our algorithm: 2-way interaction (5 green circles), 3-way interactions (4 blue serif lines), and 4-way interaction (6 red dashed lines). Each region includes the class.

8 RESULTS AND CONCLUSIONS

Judging from the rankings in Table 1, Kikuchi-Bayes with path averaging manages to outperform all of the today's most frequently used probabilistic classifiers: multinomial logistic regression with the baseline, tree-augmented naïve Bayes and the naïve Bayesian classifier, in spite of the fact that it is based on only an approximation to the Boltzmann distribution. At the same time, Kikuchi-Bayes is highly efficient in spite of the fact that it follows a fully Bayesian approach by treating the structure as a nuisance variable and that it uses exhaustive lookahead in exploring the structure space: most data sets were processed in a fraction of a second. The single non-branching greedy path of ascent in the structure space is highly interpretable.

We have noticed no deterioration by increasing the maximum interaction size, so the prior effectively prevents overfitting. However, attempting the inclusion of large regions is sometimes futile: the interactions of order 3 or even just 2 were perfectly sufficient in many natural data sets. Although these higher-order interactions are relatively rare in real-life data, we should have the capacity to handle them.

However, there appears to be an interesting phenomenon: the mismatch between the underlying assumptions of cross-validation and the i.i.d. Specifically, cross-validation seems not to overly penalize the classifiers that fit many parameters. The domains with many attributes and few examples are marked with '*' in Table 1, and we can see that Kikuchi-Bayes is con-

servative with respect to model complexity in those domains. In all, because cross-validation is not enforcing the i.i.d., methods that assume i.i.d. have suboptimal performance. We can see that ordinary naïve Bayes and tree-augmented naïve Bayes would get completely eliminated from the competition if it was not for such domains.

References

Buntine, W. (1991). Classifiers: A theoretical and empirical study. *Int. Joint Conf. on AI*. Sydney, Australia.

Burnham, K. P., & Anderson, D. R. (2002). *Model selection and multimodel inference*. Springer. 2nd edition.

Caruana, R., Niculescu, A., Crew, G., & Ksikes, A. (2004). Ensemble selection from libraries of models. *Proc. of 21st International Conference on Machine Learning (ICML)*. Banff, Alberta, Canada.

Cerquides, J., & López de Màntaras, R. (2003). Tractable Bayesian learning of tree augmented naive Bayes classifiers. *Proc. of the 20th International Conference on Machine Learning* (pp. 75–82).

Chow, C. K., & Liu, C. N. (1968). Approximating discrete probability distributions with dependence trees. *IEEE Transactions on Information Theory*, *14*, 462–467.

Friedman, N., Geiger, D., & Goldszmidt, M. (1997). Bayesian network classifiers. *Machine Learning*, *29*, 131–163.

Ireland, C. T., & Kullback, S. (1968). Contingency tables with given marginals. *Biometrika*, *55*, 179–188.

Jakulin, A., & Bratko, I. (2004). Testing the significance of attribute interactions. *Proc. of 21st International Conference on Machine Learning (ICML)* (pp. 409–416). Banff, Alberta, Canada.

Jobson, J. D. (1992). *Applied multivariate data analysis, volume II: Categorical and multivariate methods*. New York: Springer-Verlag.

Kikuchi, R. (1951). A theory of cooperative phenomena. *Physical Review*, *81*, 988–1003.

Krippendorff, K. (1986). *Information theory: Structural models for qualitative data*, vol. 07–062 of *Quantitative Applications in the Social Sciences*. Beverly Hills, CA: Sage Publications, Inc.

Pearl, J. (1988). *Probabilistic reasoning in intelligent systems*. San Francisco, CA, USA: Morgan Kaufmann.

Srebro, N. (2001). Maximum likelihood bounded tree-width Markov networks. *Proceedings of the 17th Conference on Uncertainty in Artificial Intelligence (UAI)* (pp. 504–511).

Taskar, B., Guestrin, C., & Koller, D. (2003). Max-margin Markov networks. *Neural Information Processing Systems Conference 16*. Vancouver, Canada.

Yedidia, J., Freeman, W. T., & Weiss, Y. (2001). Generalized belief propagation. *NIPS 13* (pp. 689–695). MIT Press.

Yedidia, J. S., Freeman, W. T., & Weiss, Y. (2004). *Constructing free energy approximations and generalized belief propagation algorithms* (Technical Report TR2004-040). MERL.

				log-loss / instance				
domain	t(s)	n	df	NB	LR	TAN	kMAP	kBMA
horse-colic	1.89	369	228	1.67	1.81	5.97	√0.83	**0.83**
hepatitis	0.47	155	48	0.78	√0.77	1.31	√0.48	**0.43**
ionosphere	3.71	351	129	√0.64	0.69	0.74	√0.39	**0.33**
vehicle	0.42	846	205	1.78	0.93	1.14	√0.69	**0.66**
voting	0.23	435	48	0.60	0.37	0.53	√0.21	**0.15**
monk2	0.01	601	17	0.65	0.65	0.63	√0.45	**0.45**
p-tumor*	0.39	339	552	√3.17	√2.76	4.76	2.65	**2.61**
heart	0.15	920	167	1.25	1.24	1.53	√1.11	**1.10**
post-op	0.01	88	19	√0.93	√0.81	√1.78	√0.79	**0.67**
wdbc	0.57	569	61	0.26	0.42	0.29	√0.15	**0.13**
promoters*	37.5	106	227	√0.60	√0.70	3.14	√0.59	**0.54**
lymph	0.39	148	94	√1.10	√0.91	√1.25	√0.98	**0.86**
cmc	0.04	1473	55	1.00	0.97	1.03	√0.93	**0.92**
adult	1.11	3.2e4	134	0.42	0.35	0.33	0.30	**0.30**
crx	0.19	690	58	√0.49	√0.39	0.93	√0.37	**0.36**
krkp	6.52	3196	69	0.29	0.08	0.19	√0.06	**0.05**
glass	0.03	214	90	√1.25	√1.07	√1.76	1.12	**1.05**
australian	0.16	690	49	√0.46	√0.39	0.94	√0.41	**0.38**
titanic	0.01	2201	8	0.52	0.50	√0.48	√0.48	**0.48**
segment	0.74	2310	617	0.38	0.45	1.06	**0.17**	**0.17**
lenses	0.00	24	14	√2.44	√0.89	2.99	**0.34**	0.39
monk1	0.01	556	16	0.50	0.50	0.09	**0.01**	√0.02
soy-small*	5.29	47	115	√0.00	0.15	**0.00**	0.00	0.00
mushroom	1.33	8124	72	0.01	0.00	**0.00**	0.00	0.00
shuttle	0.01	253	15	0.16	√0.10	**0.06**	√0.07	√0.07
car	0.02	1728	48	0.32	0.33	**0.18**	0.19	0.19
breast-LJ	0.03	286	24	√0.62	**0.58**	√0.89	√0.67	√0.58
monk3	0.01	554	17	0.20	**0.10**	√0.11	√0.11	√0.11
bupa	0.01	345	12	√0.62	**0.60**	√0.60	√0.62	√0.61
tic-tac-toe	0.03	958	27	0.55	**0.06**	0.49	√0.08	√0.07
pima	0.02	768	19	√0.50	**0.46**	√0.49	√0.51	√0.48
iris	0.00	150	15	√0.27	**0.21**	√0.32	√0.27	√0.23
spam	39.9	4601	156	0.53	**0.16**	0.32	0.19	√0.19
breast-wisc	0.03	683	28	√0.21	**0.13**	0.23	√0.21	√0.18
german	0.64	1000	68	√0.54	**0.52**	1.04	0.65	√0.59
anneal	6.16	898	204	√0.07	**0.02**	0.17	0.11	0.11
ecoli	0.01	336	92	√0.89	**0.68**	√0.94	√0.85	√0.83
hayes-roth	0.00	160	24	0.46	**0.26**	1.18	0.45	0.45
balance-scale	0.00	625	40	0.51	**0.28**	1.13	0.51	0.51
soy-large*	5.95	683	822	√0.57	**0.37**	√0.47	0.68	0.68
o-ring	0.00	23	7	√0.83	**0.66**	√0.76	√1.41	√1.00
lung-cancer*	35.0	32	233	5.41	**1.24**	6.92	√2.37	√1.62
audiology*	81.2	226	1783	3.55	**1.40**	5.56	2.24	2.23
wine	0.10	178	50	**0.06**	√0.09	√0.29	√0.19	√0.14
yeast-class*	138	186	376	**0.01**	0.90	√0.03	0.25	0.23
zoo*	0.25	101	124	**0.32**	√0.37	√0.42	√0.72	√0.70
avg rank	(log-loss)			3.68	√2.54	•3.95	2.88	**1.95**
avg rank	(error rate)			2.98	•3.34	3.20	√2.87	**2.62**

Table 1: For each of the 46 UCI data sets, we performed 5 replications of 5-fold cross-validation. The data sets were discretized with the Fayyad-Irani method and the missing values were interpreted as special values. The best result is typeset in bold, and the results of those methods that outperformed the best method in at least 2 of the 25 experiments are √-tagged. df are the degrees of freedom of the ordinary naïve Bayesian classifier, and n is the number of instances. The sparse data sets with fewer instances than degrees of freedom are tagged with '*'.

The Kikuchi-Bayes algorithm with model averaging (kBMA) outperforms all competing approaches (naïve Bayes (NB), logistic regression (LR), tree-augmented naïve Bayes (TAN), and the single best structure (kMAP)) both in terms of the log loss and in terms of the error rate. Logistic regression had the worst error rate, and TAN the worst log loss (•).

$t(s)$ marks the time in seconds spent by the Kikuchi-Bayes algorithm for learning the model structure using k-way interactions, $k \leq 4$, on a notebook computer: although performance does drop with an increasing number of attributes, most ordinary data sets were processed in under a second, and it never took more than 2.4 minutes.