Annals of Mathematics and Artificial Intelligence (2005) 45: 215–239
DOI: 10.1007/s10472-005-9007-9

Robust inference of trees

Marco Zaffalon* and Marcus Hutter**

IDSIA, Galleria 2, CH-6928 Manno (Lugano) Switzerland
E-mail: {zaffalon; marcus}@idsia.ch

Received 11 August 2002; Revised 5 July 2005; Accepted 27 August 2005

This paper is concerned with the reliable inference of optimal tree-approximations to the dependency structure of an unknown distribution generating data. The traditional approach to the problem measures the dependency strength between random variables by the index called *mutual information*. In this paper reliability is achieved by Walley's *imprecise Dirichlet model*, which generalizes Bayesian learning with Dirichlet priors. Adopting the imprecise Dirichlet model results in posterior interval expectation for mutual information, and in a set of plausible trees consistent with the data. Reliable inference about the actual tree is achieved by focusing on the substructure common to all the plausible trees. We develop an exact algorithm that infers the substructure in time $O(m^4)$, m being the number of random variables. The new algorithm is applied to a set of data sampled from a known distribution. The method is shown to reliably infer edges of the actual tree even when the data are very scarce, unlike the traditional approach. Finally, we provide lower and upper credibility limits for mutual information under the imprecise Dirichlet model. These enable the previous developments to be extended to a full inferential method for trees.

Keywords: robust inference, spanning trees, intervals, dependence, graphical models, mutual information, imprecise probabilities, imprecise Dirichlet model

AMS subject classification: 05C05, 41A58, 62G35, 62H20, 68T37, 90C35

1. Introduction

This paper deals with the following problem. We are given a random sample of n observations, which are jointly categorized according to a set of m nominal random variables $\imath$, $\jmath$, κ, etc. The dependency between two variables is measured by the information-theoretic symmetric index called *mutual information* [16]. If the chances[1] $\boldsymbol{\pi}$ of all instances defined by the co-occurrence of $\imath = i$, $\jmath = j$, $\kappa = \dot{\kappa}$, etc., were known, it would be possible to approximate the distribution by another, for which all the dependencies are bivariate and can graphically be represented as an undirected tree T, that is the optimal approximating tree-dependency distribution (section 2). This result

* Research partially supported by the Swiss NSF grant 2100-067961.

** Research partially supported by the Swiss NSF grant 2000-61847.00 to Jürgen Schmidhuber.

[1] We denote vectors by $\boldsymbol{x} := (x_1, \ldots, x_d)$ for $\boldsymbol{x} \in \{\boldsymbol{n}, \boldsymbol{t}, \boldsymbol{u}, \boldsymbol{\pi}, \ldots\}$.

is due to Chow and Liu [5], who use Kullback–Leiber's divergence [17] to measure the similarity of two distributions.

Since only a sample is available, the joint distribution $\boldsymbol{\pi}$ is unknown and an inferential approach is necessary. Prior uncertainty about the vector $\boldsymbol{\pi}$ is described by the *imprecise Dirichlet model* (IDM) [26]. This is an inferential model that generalizes Bayesian learning with Dirichlet priors, by using a set of prior densities to model prior (near-) ignorance. Using the IDM results in posterior uncertainty about $\boldsymbol{\pi}$, the mutual information and the tree T (section 2). In general, this makes a set of trees $\mathcal{T}$ consistent with the data.

Robust inference about T is achieved by identifying the edges common to all the trees in $\mathcal{T}$, called *strong edges* (section 3). An exact and an approximate algorithm are developed that detect strong edges in times $O(m^4)$ and $O(m^3)$ respectively. The former is applied to a set of data sampled from a known distribution, and is compared with the original algorithm from Chow and Liu (section 5). The new algorithm is shown to reliably infer partial trees (we call them *forests*), which quickly converge to the actual complete tree as the sample grows. Unlike the traditional approach based on precise probabilities, the new algorithm avoids drawing wrong edges by suspending the judgement on those for which the information is poor.

Many technical issues are addressed in the paper to develop the new algorithm. The identification of strong edges involves solving a problem on graphs. We develop original exact and approximate algorithms for this task in section 3. Robust inference involves computing bounds for the lower and upper expectation of mutual information under the IDM (section 4). We provide conservative (i.e., over-cautious) bounds that at most make an error of magnitude $O(n^{-2})$.

These results lead to important extensions, reported in section 6. Inference on mutual information is extended by providing lower and upper credibility limits under the IDM (i.e., intervals that depend on a given guarantee level). The overall approach extends accordingly. Furthermore, we discuss alternatives to the strong edges algorithm proposed in this paper, aiming to exploit the results presented here in wider contexts.

To our knowledge, the literature only reports two other attempts to infer robust structures of dependence. Kleiter [14] uses approximate confidence intervals on mutual information[2] to measure the dependence between random variables. Kleiter's work is different in spirit from ours. We look for tree structures that are optimal in some sense, by using systematic and reliable interval approximations to the actual mutual information. Kleiter focuses on general graphical structures and is not concerned with questions of optimality.

Bernard [3] describes a method to build a directed graph from a multivariate binary database. The method is based on the IDM and Bayesian implicative analysis. The connection with our work is looser here since the arcs of the graph are interpreted as logical implications rather than probabilistic dependencies.

[2] Note that accurate expressions for credible mutual information intervals have been derived in [9, 11].

2. Background

2.1. Maximum spanning trees

This paper is concerned with trees. In the undirected case, trees are undirected connected graphs with m nodes and $m-1$ edges. Undirected trees are such that for each pair of nodes there is only one path that connects them [20, Proposition 2]. Directed trees can be constructed from undirected ones, orienting the arrows in such a way that each node has at most a single direct predecessor (or *parent*). When used to represent dependency structures, the nodes of a tree are regarded as random variables and the tree itself represents the dependencies between the variables. It is a well-known result that all the directed trees that share the same undirected structure represent the same set of dependencies [24]. This is the reason why the inference of directed trees from data focuses on recovering the undirected structure; and it is also the reason why this paper is almost entirely concerned with undirected trees (called more simply 'trees' in the following).

Chow and Liu [5] address the problem of approximating the actual pattern of dependencies of a distribution by an undirected tree. Their work is based on mutual information. Given two random variables $\imath$, $\jmath$ with values in $\{1,\dots,d_\imath\}$ and $\{1,\dots,d_\jmath\}$, respectively, the *mutual information* is defined as

$$\mathcal{I}(\boldsymbol{\pi}) = \sum_{i=1}^{d_\imath}\sum_{j=1}^{d_\jmath} \pi_{ij} \log \frac{\pi_{ij}}{\pi_{i+}\pi_{+j}},$$

where π_{ij} is the actual chance of (i, j), and $\pi_{i+} := \sum_j \pi_{ij}$ and $\pi_{+j} := \sum_i \pi_{ij}$ are marginal chances. Chow and Liu's algorithm works by computing the mutual information for all the pairs of random variables. These values are used as edge *weights* in a fully connected graph. The output of the algorithm is a tree for which the sum of the edge weights is maximum. In the literature of graph algorithms, the general version of the last problem is called the *maximum spanning tree* [20, p. 271]. Its construction takes $O(m^2)$ time. This is also the computational complexity of the above procedure. The tree constructed as above is shown to be an optimal tree-approximation to the actual dependencies when the similarity of two distributions is measured by Kullback-Leiber's divergence [17].

Chow and Liu extend their procedure to the inference of trees from data by replacing the mutual information with the *sample mutual information* (or *empirical mutual information*). This approximates the actual mutual information by using, in the expression for mutual information, the sample relative frequencies instead of the chances π_{ij}, which are typically unknown in practice.

2.2. Robust inference

Using empirical approximations for unknown quantities, as described in the previous section, can lead to fragile models. Fragile models produce quite different outputs depending on the random fluctuations involved in the generation of the sample.

Reliability can be achieved by robust inferential tools. In this paper we consider the imprecise Dirichlet model [4, 26]. The IDM is a model of inference for multivariate categorical data. It models prior uncertainty using a set of Dirichlet prior densities and does posterior inference by combining them with the likelihood function (see section 4.1 for details). The IDM rests on very weak prior assumptions and is therefore a very robust inferential tool.

The IDM leads to lower and upper expectations for mutual information (and, possibly, lower and upper credibility limits), i.e., to intervals. This is a complication for the discovery of tree structures from data. In fact, the maximum spanning tree problem assumes that the edge weights can be totally ordered. Now, multiple values of mutual information are generally consistent with the given intervals. In general, this prevents us from having a total order on the edges: not all the pairs of edges can be compared.

The generalization of Chow and Liu's approach is achieved via the definition of more general graphs that can deal with multiple edge weights. This is done in the next section.

3. Set-based weighted graphs

Consider an undirected fully connected graph $G_w = \langle V, E \rangle$, with $m = |V|$ nodes, and where E denotes the set of edges $[(v,v) \notin E$ for each $v \in V]$. G_w is also a weighted graph, in the sense that each edge $e \in E$ is associated with the real number $w(e)$, which in this paper will be a value of mutual information. Consider a set of graphs with the same topological structure but different weight functions w in a non-empty set $W : \mathcal{G} = \{G_w : w \in W\}$. We call $\mathcal{G}$ a *set-based weighted graph*. Note that $\mathcal{G}$ can be thought of also as a single graph G, on each edge e of which there is a set of real weights: $\{w(e) : w \in W\}$. Yet, for the latter view to be equivalent to the former, one should pay attention to the fact that there could be logical dependencies between weights of two different sets; in other words, it could be the case that not all the pairs of weights in the cartesian product of two sets appear in a single graph of $\mathcal{G}$.

In order to extend the notion of maximum spanning tree to set-based weighted graphs, we define the solution of the maximum spanning tree problem generalized to set-based weighted graphs, as the set $\mathcal{T}$ of maximum spanning trees originated by the graphs in $\mathcal{G}$.

Recall that Kruskal's algorithm only needs a total order on the edges to build a unique maximum spanning tree [15]. Therefore, in order to focus on $\mathcal{T}$, we can equivalently focus on the set $\mathcal{O}_\mathcal{T}$ of total orders that are consistent with the graphs in $\mathcal{G}$. In the following we find it more convenient not to directly deal with $\mathcal{O}_\mathcal{T}$. Rather, we first show how to construct a partial order that is consistent with all the total orders in $\mathcal{O}_\mathcal{T}$, and then we consider all the total orders that extend the partial order. Initially, we need the following definition.

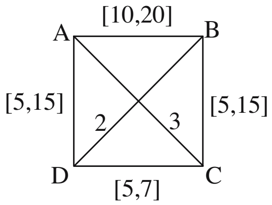

Figure 1. An example set-based weighted graph. The sets for the edges are specified separately by intervals that in two cases degenerate to real numbers.

Definition 1. We say that edge e *dominates* edge e' if $w(e) > w(e')$ for all $w \in W$.

By applying the above definition to all the distinct pairs of edges in G we obtain the sought partial order. To see that the order is only partial in general, consider the example graph in figure 1. We have defined such a graph G by drawing the graphical structure and specifying set-based weights by placing intervals on the edges in a separate way (i.e., assuming logical independency between different intervals). That is, the example graph is equivalent to the set $\mathcal{G}$ of graphs obtained by choosing real weights within the intervals in all the possible ways. Now observe that the intervals for the edges (A,B) and (B,C) overlap, so that there is no dominance in either direction. Figure 2 shows the overall partial order on the edges for the graph in figure 1.

Now we consider the set $\mathcal{O}$ of all the total orders that extend the partial order induced by Definition 1. Of course, $\mathcal{O}$ includes $\mathcal{O}_{\mathcal{T}}$. They coincide if for each total order in $\mathcal{O}$, there is a graph $G_w \in \mathcal{G}$ in which $w(e) > w(e')$ if e dominates e' in the given total order. This is the case, for example, when mutual information is separately specified via intervals on the edges.

3.1. Exact detection of strong edges

We call *strong edges* the edges of G that are common to all the trees in $\mathcal{T}$. Identifying the strong edges allows us to robustly infer dependencies that belong to the unknown optimal approximating trees. The following theorem is the central tool for the identification.

Theorem 2. Assume $\mathcal{O} = \mathcal{O}_{\mathcal{T}}$. An edge e of G is strong if and only if in each simple[3] cycle that contains e there is an edge e' dominated by e.

[3] This is a cycle in which the nodes are all different. In the following we will simply refer to simple cycles as cycles.

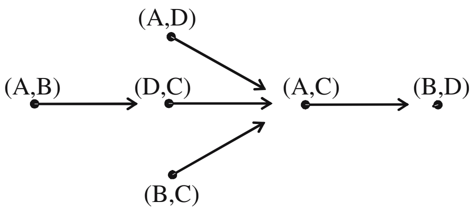

Figure 2. The partial order on the edges of the graph in the preceding figure. Here an arrow from e to e' means that e dominates e'.

Proof. ($\Leftarrow$) By contradiction, assume that there is a graph $G_w \in \mathcal{G}$ for which an optimal tree T does not contain e. By adding e to T we create a cycle [20, Proposition 2]. By hypothesis, in such a cycle there must exist an edge e' dominated by e, so $w(e) > w(e')$. Removing e', we obtain a new tree that improves upon T, so that T cannot be optimal for G_w.

($\Rightarrow$) By contradiction, assume that there is a cycle C in G where e does not dominate any edge. Then there is a total order in $\mathcal{O}$ in which e is dominated by any other edge e' in C. Since $\mathcal{O} = \mathcal{O}_{\mathcal{T}}$, there must also exist a related graph G_w for which $w(e) \leq w(e')$ for any edge e' in C. Call T the related tree. By removing e from T we create two subtrees, say T' and T''. One of these can possibly be a degenerate tree composed by a single node. Now consider that there must be an edge e_C of C that connects a node of T' with one of T''. If there was not, there would be no way to start from an endpoint of e in T' and reach the other endpoint, because all the paths would be confined within T'. The graph composed by T', T'' and e_C has $m-1$ edges, spans all the nodes of G, and therefore it is a tree, say T^* [20, Proposition 2]. If $w(e) < w(e_C)$, T^* improves upon T, so that T cannot be optimal for G_w. If $w(e) = w(e_C)$, both T^* and T are optimal, but their intersection does not contain e, so $e \notin \mathcal{T}$. □

Theorem 2 directly leads to a procedure that determines whether or not a given edge e is strong. It suffices to consider the graph G' obtained from G by removing e and the edges that e dominates (see the Procedure ‘DetectStrongEdges’ in table III). Edge e is strong if and only if its endpoints are not connected in G'. By applying this procedure to the graph in figure 1, we conclude that only (A,B) is strong.

Note that Theorem 2 assumes that $\mathcal{O}$ coincides with $\mathcal{O}_{\mathcal{T}}$. If this failed to be true, $\mathcal{O}_{\mathcal{T}} \subset \mathcal{O}$ would still hold, making Theorem 2 work with a set of trees larger than $\mathcal{T}$, eventually leading to an excess of caution: the edges determined by the above procedure would anyway be strong, but there might be strong edges that the procedure would not be able to determine.

As for computational considerations, note that testing whether or not two nodes are connected in a graph demands $O(m^2)$ time. Repeating the test for all the edges $e \in E$, we have the computational complexity of the overall procedure, $O(m^4)$.

3.2. Approximate detection of strong edges

This section presents a procedure that approximately detects the strong edges, reducing the complexity to $O(m^3)$ with respect to the exact procedure given in section 3.1.

Consider the algorithm outlined in a pseudo programming language in table I. It takes as input a fully connected graph $G = \langle V, E \rangle$. In the algorithm, a tree with a number of nodes in $\{2, \ldots, m-1\}$ is called *subtree*.

The following proposition shows that the algorithm in table I returns only strong edges.

Proposition 1. *SE* is a subset of the strong edges of G.

Proof. Consider the first possible insertion in Step 2(a)i. The cycles that encompass (v,v') must pass through the set of edges $\{(v,v''): v'' \in V, v'' \neq v'\}$. Since (v,v') dominates all the edges in the preceding set, for each cycle passing through (v,v') there is an edge in the cycle that is dominated by (v,v'), so that (v,v') is strong, by Theorem 2.

The algorithm can insert an edge in *SE* also in Step 3(c)i. Recall that each subtree is a connected acyclic graph. It is clear that any cycle that contains e' must pass through an edge e'' that has one endpoint in the nodes of the subtree and the other outside. But e' dominates e'' by Step 3c. This holds for all the cycles, so e' is strong by Theorem 2. □

The logic of the algorithm in table I is to move from subtrees made of strong edges to adjacent nodes, in order to detect the strong edges of a graph. This policy

Table I
Approximate procedure to detect strong edges.

```
1. Let $SE = \emptyset$;
2. for each $v \in V$
   a) if there is a node $v' \in V$ such that $(v,v') \notin SE$ and it dominates $(v,v'')$ for each $v'' \in V$, $v'' \neq v'$ then
      i) add $(v, v')$ to $SE$
3. if there is a subtree in $SE$ then
   a) make it the current subtree;
   b) consider the set of edges $E' \subseteq E$ with one endpoint in the nodes of the current subtree and the other outside;
   c) if there is an edge $e' \in E'$ that dominates all the other edges in $E'$ then
      i) add $e'$ to $SE$ and to the current subtree;
      ii) go to 3b;
   d) else
      i) if there is another subtree in $SE$ not considered yet then
         A) go to 3a;
      ii) else output $SE$.
```

does not allow all the strong edges to be determined in general. For example, the approximate algorithm cannot determine that the edge (A,B) in figure 1 is strong.

The heuristic policy implements a trade-off between computational complexity and the capability to fully detect the strong edges. This choice does not seem critical to the specific extent of discovering tree-dependency structures. In fact, the knowledge of the actual mutual information increases with the sample size, becoming a number in the limit. It is easy to check that in these conditions the exact and the approximate procedure produce the same set of edges.

3.2.1. Computational complexity

The assumption behind the following analysis is that the comparison of two edges can be done in constant time. In this case, given a set E' of edges, there is a procedure that determines in time $O(|E'|)$ if there is an edge $e' \in E'$ that dominates all the others. The first step of the procedure selects an edge that is candidate to be dominant. This is made by doing pairwise comparisons of edges and by always discarding the non-dominant edge (or edges) in the comparison. After at most $|E'|-1$ comparisons, we know whether there is a candidate or not. If there is, the second step of the procedure compares such candidate e' with all the other edges, deciding if e' dominates all the others. This requires $|E'|-1$ comparisons. The two steps of the procedure take $O\big(|E'|\big)$ time.

Let us now focus on the algorithm in table I. The loop 2 is repeated $m = |V|$ times. Each time the test 2a decides whether there is a dominant edge out of $m-1$ edges (each node is connected to all the others). By the previous result, such task takes $O(m)$ time. Then the loop requires $O(m^2)$ time.

Now consider the two nested loops made by the instructions 3a, 3b, 3(c)ii, and 3(d)iA. Each time the instruction of jump 3(c)ii is executed, a new edge has been added to *SE*. Each time 3(d)iA is executed, a new subtree is considered. Since *SE* can have $m-1$ edges at most and m is also an upper bound on the number of different subtrees, the two loops can jointly require $2m-1$ iterations at most. Each such iteration executes the test 3c. By using m^2 as an upper bound on $|E'|$, we need $O(m^2)$ time to detect whether the dominant edge exists. The overall time required by the loops is $O(m^3)$. This is also the computational complexity of the entire procedure.

4. Robust comparison of edges

So far we have focused on the detection of strong edges, taking for granted that there exists a method to partially compare edges based on imprecise knowledge of mutual information. We provide such a method in the following sections. We will first present a formal introduction to the imprecise Dirichlet model in section 4.1. Section 4.2 will make a first step by computing robust estimates for the entropy. These will be used in section 4.3 to derive robust estimates of mutual information. Finally, the method to compare edges will be given in section 4.4.

4.1. The imprecise Dirichlet model

4.1.1. Random i.i.d. processes

We consider a discrete random variable $\imath$ and a related i.i.d. random process with samples $i \in \{1,\ldots,d\}$ drawn with probability π_i. The chances $\boldsymbol{\pi}$ form a probability distribution, i.e., $\boldsymbol{\pi} \in \Delta := \{\boldsymbol{x} \in \mathbb{R}^d : x_i \geq 0 \forall i, x_+ = 1\}$, where we have used the abbreviation $x_+ := \sum_{i=1}^d x_i$. The likelihood of a specific data set $\boldsymbol{D} = (i_1,\ldots,i_n)$ with n_i samples i and total sample size $n = n_+ = \sum_i n_i$ is $p(\boldsymbol{D}|\boldsymbol{\pi}) \propto \prod_i \pi_i^{n_i}$. Quantities of interest are, for instance, the entropy $\mathcal{H}(\boldsymbol{\pi}) = -\sum_i \pi_i \log \pi_i$, where log denotes the natural logarithm. The chances π_i are usually unknown and have to be estimated from the data.

4.1.2. Second order p(oste)rior

In the Bayesian approach one models the initial uncertainty in $\boldsymbol{\pi}$ by a (second order) prior distribution $p(\boldsymbol{\pi})$ with domain $\boldsymbol{\pi} \in \Delta$. The Dirichlet priors $p(\boldsymbol{\pi}) \propto \prod_i \pi_i^{n_i'-1}$, where n_i' comprises prior information, represent a large class of priors. n_i' may be interpreted as (possibly fractional) 'virtual' sample numbers. High prior belief in i can be modelled by large n_i'. It is convenient to write $n_i' = s \cdot t_i$ with $s := n_+'$, hence $\boldsymbol{t} \in \Delta$. Examples for s are 0 for Haldane's prior [8], 1 for Perks' prior [22], $\frac{d}{2}$ for Jeffreys' prior [12], and d for Bayes-Laplace's uniform prior [7] (all with $t_i = \frac{1}{d}$). These are also called *non-informative priors*. From the prior and the data likelihood one can determine the posterior $p(\boldsymbol{\pi}|\boldsymbol{D}) = p(\boldsymbol{\pi}|\boldsymbol{n}) \propto \prod_i \pi_i^{n_i+st_i-1}$. The expected value or mean $u_i := E_{\boldsymbol{t}}[\pi_i] = \frac{n_i+st_i}{n+s}$ is often used for estimating $\boldsymbol{\pi}_i$ (the accuracy may be obtained from the covariance of $\boldsymbol{\pi}$). The expected entropy is $E_{\boldsymbol{t}}[\mathcal{H}] = \int_\Delta \mathcal{H}(\boldsymbol{\pi}) p(\boldsymbol{\pi}|\boldsymbol{n}) d\boldsymbol{\pi}$. An approximate solution can be obtained by exchanging E with $\mathcal{H}$ (exact only for linear functions): $E_{\boldsymbol{t}}[\mathcal{H}(\boldsymbol{\pi})] \approx \mathcal{H}(E_{\boldsymbol{t}}[\boldsymbol{\pi}]) = \mathcal{H}(\boldsymbol{u})$. The approximation error is typically of the order $\frac{1}{n}$. In [9, 11, 27] exact expressions have been obtained:

$$E_{\boldsymbol{t}}[\mathcal{H}] = H(\boldsymbol{u}) := \sum_i h(u_i) \text{ with}$$
$$h(u) = u \cdot [\psi(n+s+1) - \psi((n+s)u+1)], \tag{1}$$

where $\psi(x) = d \log \Gamma(x)/dx$ is the logarithmic derivative of the Gamma function. There are fast implementations of ψ and its derivatives and exact expressions for integer and half-integer arguments (see Appendix).

4.3.1. Definition of the imprecise Dirichlet model

There are several problems with noninformative priors. First, the inference generally depends on the arbitrary definition of the sample space. Second, they assume exact prior knowledge $p(\boldsymbol{\pi})$. The solution to the second problem is to model our ignorance by considering sets of priors $p(\boldsymbol{\pi})$, a model that is part of the wider theory of *imprecise*[4] *probabilities* [25]. The specific imprecise Dirichlet model [26] considers the set of all[5] $\boldsymbol{t} \in \Delta$, i.e., $\{p(\boldsymbol{\pi}) : \boldsymbol{t} \in \Delta\}$, which solves also the first problem. Walley

[4] In the following we will avoid the term *imprecise* in favor of *robust*, since expressions like 'exact imprecise intervals' sound confusing.

suggests to fix the hyperparameter s somewhere in the interval [1, 2]. A set of priors results in a set of posteriors, set of expected values, etc. For real-valued quantities like the expected entropy $E_t[\mathcal{H}]$ the sets are typically intervals:

$$E_t[\mathcal{H}] \in [\min_{\boldsymbol{t}\in\Delta} E_{\boldsymbol{t}}[\mathcal{H}],\ \max_{\boldsymbol{t}\in\Delta} E_{\boldsymbol{t}}[\mathcal{H}]] =: [\underline{H}, \overline{H}].$$

In the next section we derive approximations for

$$\overline{H} = \max_{\boldsymbol{t}\in\Delta} H(\boldsymbol{u}) \text{ and } \underline{H} = \min_{\boldsymbol{t}\in\Delta} H(\boldsymbol{u}).$$

One can show that $h(u)$ is strictly concave (see Appendix), i.e., $h''(u) < 0$ and that h'' is monotone increasing ($h''' > 0$), which we exploit in the following. The results for the entropy serve as building blocks to derive similar results for the needed mutual information. We define the general correspondence

$$u_i^{\cdots} = \frac{n_i + st_i^{\cdots}}{n+s}, \quad \text{where} \cdots \text{can be various superscripts.}$$

4.2. *Robust entropy estimates*

4.2.1. *Taylor expansion of H(u)*

In the following we derive reliable approximations for $\overline{H}$ and $\underline{H}$. If n is not too small these approximations are close to the exact values. More precisely, the length of interval $[\underline{H}, \overline{H}]$ is $O(\sigma)$, where $\sigma := \frac{s}{n+s}$, while the approximations will differ from $\overline{H}$ and $\underline{H}$ by at most $O(\sigma^2)$. Let $t_i \in [0,1]$ and $u_i^* = \frac{n_i + st_i^*}{n+s}$. This implies

$$u_i - u_i^* = \sigma \cdot (t_i - t_i^*) \text{ and } |u_i - u_i^*| = \sigma|t_i - t_i^*| \le \sigma. \tag{2}$$

Hence we may Taylor-expand $H(\boldsymbol{u})$ around $\boldsymbol{u}^*$. H is approximately linear in $\boldsymbol{u}$ and hence in $\boldsymbol{t}$. A linear function on a simplex assumes its extreme values at the vertices of the simplex. The most natural point for expansion is $t_i^* = \frac{1}{d}$ in the center of Δ. For this choice the bound (2) and most of the following bounds can be improved to $\sigma \rightsquigarrow \sigma|1 - \frac{1}{d}|$. Other, even data-dependent choices like $t_i^* = \frac{n_i}{n} = u_i^*$, are possible. The only property we use in the following is that[6] $\text{argmax}_i u_i^* = \text{argmax}_i n_i$ and $\text{argmin}_i u_i^* = \text{argmin}_i n_i$. We have

$$H(\boldsymbol{u}) = \overbrace{H(\boldsymbol{u}^*)}^{H_0 = O(1)} + \overbrace{\sum_i h'(u_i^*)(u_i - u_i^*)}^{H_1 = O(\sigma)} + \overbrace{\frac{1}{2}\sum_i h''(\breve{u}_i)(u_i - u_i^*)^2}^{H_R = O(\sigma^2)}.$$

For suitable $\breve{u}_i$ between u_i^* and u_i this expansion is exact (H_R is the exact remainder).

[5] Strictly speaking, Δ should be the open simplex [26], since $p(\boldsymbol{\pi})$ is improper for $\boldsymbol{t}$ on the boundary of Δ. For simplicity we assume that, if necessary, considered functions of $\boldsymbol{t}$ can be, and are, continuously extended to the boundary of Δ, so that, for instance, minima and maxima exist. All considerations can straightforwardly, but cumbersomely, be rewritten in terms of an open simplex. Note that open/closed Δ result in open/closed robust intervals, the difference being numerically/practically irrelevant.

[6] $\text{argmin}_i n_i$ is the i for which n_i is minimal. Ties can be broken arbitrarily. Kronecker's $\delta_{i,j} = 1$ for $i = j$ and $\delta_{i,j} = 0$ for $i \neq j$.

4.2.2. *Approximation of $\overline{H}$*

Inserting (2) into H_1 we get

$$H_1 = \sum_i h'(u_i^*)(u_i - u_i^*) = \sigma \sum_i h'(u_i^*)(t_i - t_i^*).$$

Ignoring the $O(\sigma^2)$ remainder H_R, in order to maximize $H(\boldsymbol{u})$ we only have to maximize $\sum_i h'(u_i^*)t_i$ (the only $\boldsymbol{t}$-dependent part). A linear function on Δ is maximized by setting the t_i component with largest coefficient to 1. Due to concavity of h, $h'(u_i^*)$ is largest for the smallest u_i^*, i.e., for smallest n_i, i.e., for $i = \bar{i} := \text{argmin}_i n_i$. Hence $\overline{H}_1 = H_1(\overline{\boldsymbol{u}})$, where $\bar{t}_i := \delta_{i,\bar{i}}$ and $\overline{\boldsymbol{u}}$ follows from $\bar{\boldsymbol{t}}$ by the general correspondence. $H_0 + \overline{H}_1$ is an $O(\sigma^2)$ approximation of $\overline{H}$. Consider now the remainder H_R:

$$H_R = \frac{1}{2}\sigma^2 \sum_i h''(\check{u}_i)|t_i - t_i^*|^2 \le 0 =: H_R^{ub}$$

due to $h'' < 0$. This bound cannot be improved in general, since $H_R = 0$ is attained for $t_i = t_i^*$. Non-positivity of H_R shows that $H_0 + \overline{H}_1$ is an upper bound of $\overline{H}$. Since $\overline{H} \ge H(\boldsymbol{u})$ for all $\boldsymbol{u}$, $H(\overline{\boldsymbol{u}})$ in particular is a lower bound on $\overline{H}$, and moreover also an $O(\sigma^2)$ approximation. Together we have

$$\underbrace{H(\overline{\boldsymbol{u}})}_{\overline{H} - O(\sigma^2)} \le \overline{H} \le \underbrace{H_0 + \overline{H}_1}_{\overline{H} + O(\sigma^2)}.$$

For robust estimates, the upper bound is, of course, more interesting.

4.2.3. *Approximation of $\underline{H}$*

The determination of $\underline{H_1}$ follows the same scheme as for $\overline{H_1}$. We get $\underline{H_1} = H_1(\underline{\boldsymbol{u}})$ with $\underline{t}_i := \delta_{i,\underline{i}}$ and $\underline{i} := \text{argmax}_i n_i$. Using $|t_i - t_i^*| \le 1, \check{u}_i \ge \frac{n_i}{n+s}, h'' < 0$ and that h'' is monotone increasing ($h''' > 0$) we get the following lower bound on the remainder H_R:

$$H_R = \frac{1}{2}\sigma^2 \sum_i h''(\check{u}_i)|t_i - t_i^*|^2 \ge \frac{1}{2}\sigma^2 \sum_i h''\left(\frac{n_i}{n+s}\right) =: H_R^{lb}.$$

Putting everything together we have

$$\underbrace{H_0 + \underline{H_1}}_{\underline{H} - O(\sigma^2)} + \underbrace{H_R^{lb}}_{O(\sigma^2)} \le \underline{H} \le \underbrace{H(\underline{\boldsymbol{u}})}_{\underline{H} + O(\sigma^2)}.$$

For robust estimates, the lower bound is more interesting. General approximation techniques for other quantities of interest are developed in [10]. Exact expressions for $[\underline{H}, \overline{H}]$ are also derived there.

4.3. *Robust estimates for mutual information*

4.3.1. *Mutual information*

Here we generalize the bounds for the entropy found in section 4.2 to the mutual information of two random variables $\imath$ and $\jmath$ that take values in $\{1,\ldots,d_\imath\}$ and $\{1,\ldots,d_\jmath\}$, respectively. Consider an i.i.d. random process with samples $(i,j) \in \{1,\ldots,d_\imath\} \times \{1,\ldots,d_\jmath\}$ drawn with joint probability π_{ij}, where $\boldsymbol{\pi} \in \Delta := \{\boldsymbol{x} \in \mathbb{R}^{d_\imath \times d_\jmath} : x_{ij} \geq 0 \forall ij, x_{++} = 1\}$. We are interested in the mutual information of $\imath$ and $\jmath$:

$$\begin{aligned}\mathcal{I}(\boldsymbol{\pi}) &= \sum_{i=1}^{d_\imath}\sum_{j=1}^{d_\jmath} \pi_{ij} \log \frac{\pi_{ij}}{\pi_{i+}\pi_{+j}} \\ &= \sum_{ij} \pi_{ij} \log \pi_{ij} - \sum_i \pi_{i+} \log \pi_{i+} - \sum_j \pi_{+j} \log \pi_{+j} \\ &= \mathcal{H}(\boldsymbol{\pi}_{\imath+}) + \mathcal{H}(\boldsymbol{\pi}_{+\jmath}) - \mathcal{H}(\boldsymbol{\pi}_{\imath\jmath}).\end{aligned}$$

$\pi_{i+} = \sum_j \pi_{ij}$ and $\pi_{+j} = \pi \sum_i \pi_{ij}$ are marginal probabilities. Again, we assume a Dirichlet prior over $\boldsymbol{\pi}_{\imath\jmath}$, which leads to a Dirichlet posterior $p(\boldsymbol{\pi}_{\imath\jmath}|\boldsymbol{n}) \propto \prod_{ij} \pi_{ij}^{n_{ij}+st_{ij}-1}$. The expected value of π_{ij} is

$$u_{ij} := E_{\boldsymbol{t}}[\pi_{ij}] = \frac{n_{ij} + st_{ij}}{n + s}.$$

The marginals $\boldsymbol{\pi}_{i+}$ and $\boldsymbol{\pi}_{+j}$ are also Dirichlet with expectation u_{i+} and u_{+j}. The expected mutual information $E_{\boldsymbol{t}}[\mathcal{I}]$ can, hence, be expressed in terms of the expectations of three entropies

$$\begin{aligned}I(\boldsymbol{u}) &:= H(\boldsymbol{u}_{\imath+}) + H(\boldsymbol{u}_{+\jmath}) - H(\boldsymbol{u}_{\imath\jmath}) = H_{left} + H_{right} - H_{joint} \\ &= \sum_i h(u_{i+}) + \sum_j h(u_{+j}) - \sum_{ij} h(u_{ij})\end{aligned}$$

where here and in the following we index quantities with *joint*, *left*, and *right* to denote to which distribution the quantity refers. Using (1) we get $E_{\boldsymbol{t}}[\mathcal{I}] = I(\boldsymbol{u})$.

4.3.2. *Crude bounds for I(**u**)*

Estimates for the IDM interval $[\min_{\boldsymbol{t}\in\Delta} E_{\boldsymbol{t}}[\mathcal{I}], \max_{\boldsymbol{t}\in\Delta} E_{\boldsymbol{t}}[\mathcal{I}]]$ can be obtained by minimizing/maximizing $I(\boldsymbol{u})$. A crude upper bound can be obtained as

$$\begin{aligned}\overline{I} &:= \max_{\boldsymbol{t}\in\Delta} I(\boldsymbol{u}) = \max[H_{left} + H_{right} - H_{joint}] \\ &\leq \max H_{left} + \max H_{right} - \min H_{joint} = \overline{H}_{left} + \overline{H}_{right} - \underline{H}_{joint},\end{aligned}$$

where upper and lower bounds to $\overline{H}_{left}, \overline{H}_{right}$ and $\overline{H}_{joint}$ have been derived in section 4.2. Similarly $\underline{I} \geq \underline{H}_{left} + \underline{H}_{right} - \overline{H}_{joint}$. The problem with these bounds is that, although good in some cases, they can become arbitrarily crude. In the following we derive bounds similar to the entropy case with $O(\sigma^2)$ accuracy.

4.3.3. $O(\sigma^2)$ *bounds for* $I(\boldsymbol{u})$

We expand $I(\boldsymbol{u})$ around $\boldsymbol{u}^*$ with a constant term I_0, a term I_1 linear in σ and an exact $O(\sigma^2)$ remainder.

$$I(\boldsymbol{u}) = I_0 + I_1 + I_R, \quad I_0 = H_{0left} + H_{0right} - H_{0joint} = I(\boldsymbol{u}^*),$$

$$\begin{aligned} I_1 &= H_{1left} + H_{1right} - H_{1joint} \\ &= \sum_i h'(u^*_{i+})(u_{i+} - u^*_{i+}) + \sum_j h'(u^*_{+j})(u_{+j} - u^*_{+j}) \sum_{ij} h'(u^*_{ij})(u_{ij} - u^*_{ij}) \\ &= \sigma \sum_{ij} g_{ij}(t_{ij} - t^*_{ij}) \quad \text{with} \quad g_{ij} := h'(u^*_{+}) + h'(u^*_{+j}) - h'(u^*_{ij}). \end{aligned}$$

$\underline{I_1}$ is maximal if $\sum_{ij} g_{ij} t_{ij}$ is maximal. This is maximal if $t_{ij} = \bar{t}_{ij} := \delta_{(ij),\overline{(ij)}}$ and $\overline{(ij)} := \text{argmax}_{(ij)} g_{ij}$, hence $\overline{I_1} = I_1(\overline{\boldsymbol{u}})$, and $I_0 + \overline{I_1}$ and $I(\overline{\boldsymbol{u}})$ being $O(\sigma^2)$ approximations to $\bar{I}$. Replacing all max's by min's we get $I_0 + \underline{I_1}$ and $I(\underline{\boldsymbol{u}})$ as $O(\sigma^2)$ approximations to $\underline{I}$. To get robust bounds we need bounds on $I_R = H_{R\,left} + H_{Rright} - H_{Rjoint}$.

$$\begin{aligned} I_R &\le \max_{\boldsymbol{u},\check{\boldsymbol{u}}} [H_{Rleft} + H_{Rright} - H_{Rjoint}] \\ &\le H^{ub}_{Rleft} + H^{ub}_{Rright} - H^{lb}_{Rjoint} = -H^{lb}_{Rjoint} =: I^{ub}_R. \\ I_R &\ge \min_{\boldsymbol{u},\check{\boldsymbol{u}}} [H_{Rleft} + H_{Rright} - H_{Rjoint}] \\ &\ge H^{lb}_{Rleft} + H^{lb}_{Rright} = H^{ub}_{Rjoint} = H^{lb}_{Rleft} + H^{lb}_{Rright} =: I^{lb}_R. \end{aligned}$$

Note that for H_R we can tolerate such a crude approximation, since H_R (and $H_R^{ub/lb}$) are small $O(\sigma^2)$ corrections. In summary we have

$$\begin{aligned} \overbrace{I(\overline{\boldsymbol{u}})}^{\bar{I}-O(\sigma^2)} &\le \bar{I} \le \overbrace{I_0 + \overline{I_1}}^{\bar{I}+O(\sigma^2)} + \overbrace{I^{ub}_R}^{O(\sigma^2)} \text{ and} \\ \underbrace{I_0 + \underline{I_1}}_{\underline{I}-O(\sigma^2)} + \underbrace{I^{lb}_R}_{O(\sigma^2)} &\le \underline{I} \le \underbrace{I(\underline{\boldsymbol{u}})}_{\underline{I}+O(\sigma^2)}. \end{aligned}$$

4.4. *Comparing edges*

For two edges a and b with no common vertex, the reliable interval containing $[\underline{I}, \bar{I}]$ of section 4.3 can be used separately for a and b. For edges with a common vertex the results of section 4.3 may still be used, but they may no longer be reliable or good from a global perspective. Consider the subgraph $\imath \overset{a}{—} \jmath \overset{b}{—} \kappa$, joint probabilities $\pi_{\imath\jmath\kappa}$ of vertices $\imath$, $\jmath$, κ, a Dirichlet posterior $\prod_{ij\dot\kappa} \pi_{ij\dot\kappa}^{n_{ij\dot\kappa}+st_{ij\dot\kappa}-1}$, $u_{ij\dot\kappa} = E_t[\pi_{ij\dot\kappa}] = \frac{n_{ij\dot\kappa}+st_{ij\dot\kappa}}{n+s}$, etc. The expected mutual information between node $\imath$ and j is $I^a := I(\boldsymbol{u}^a)$ and $I^b := I(\boldsymbol{u}^b)$ between $\jmath$ and κ, where $u_{ij}^a = u_{ij+}$ and $u_{j\dot\kappa}^b = u_{+j\dot\kappa}$. The weight of edge a is w^a = [min I^a, max I^a], where min and max are w.r.t. $t_{ij}^a := t_{ij+}$. Similarly, the weight of edge b is w^b = [min I^b, max I^b], where min and max is w.r.t. $t_{j\dot\kappa}^b := t_{+j\dot\kappa}$. The results of section 4.3 can be used to determine the intervals. Unfortunately this procedure neglects the constraint $t_{+j}^a = t_{j+}^b$. The correct treatment is to define w^a larger than w^b as follows:

$$[w^a > w^b] \;\Leftrightarrow\; [I^a > I^b \text{ for all } t_{\imath\jmath\kappa} \in \Delta] \;\Leftrightarrow\; \min_{\boldsymbol{t}}[I^a - I^b] > 0.$$

The crude approximation min $[I^a - I^b] \geq \min I^a - \max I^b$ gives back the above naive interval comparison procedure. This shows that the naive procedure is reliable, but the approximation may be crude. For good estimates we proceed similar as in section 4.3 to get $O(\sigma^2)$ approximations and bounds on $I^a - I^b$.

$$\overbrace{I_0^a - I_0^b + I_1^a(\underline{\boldsymbol{u}}) - I_1^b(\underline{\boldsymbol{u}})}^{\min[I^a - I^b - O(\sigma^2)]} + \overbrace{I_R^{a.lb} - I_R^{b.ub}}^{O(\sigma^2)} \leq \min_{\boldsymbol{t}\in\Delta}[I^a - I^b] \leq \overbrace{I^a(\underline{\boldsymbol{u}}) - I^b(\underline{\boldsymbol{u}})}^{\min[I^a - I^b] + O(\sigma^2)}$$

$$\underline{(ij\dot\kappa)} := \arg\min_{ij\dot\kappa}[h'(u_{i++}^*) - h'(u_{ij+}^*) - h'(u_{++\dot\kappa}^*) + h'(u_{+j\dot\kappa}^*)]$$
$$= \arg_{ij\dot\kappa}\{\min_j[\min_i(h'(u_{i++}^* - h'(u_{ij+}^*)) + \min_{\dot\kappa}(h'(u_{+j\dot\kappa}^*) - h'(u_{++\dot\kappa}^*))]\}$$

and $\underline{t}_{ij\dot\kappa} := \delta_{(ij\dot\kappa),\underline{(ij\dot\kappa)}}$, and, for instance, choosing $t_{ij\dot\kappa}^* = \frac{1}{d_\imath d_\jmath d_\kappa}$ or $t_{ij\dot\kappa}^* = \frac{n_{ij\dot\kappa}}{n} = u_{ij\dot\kappa}^*$. The second representation for $\underline{(ij\dot\kappa)}$ shows that $\underline{(ij\dot\kappa)}$, and hence the bounds, can be computed in time $O(d^2)$ rather than $O(d^3)$. Note that $\min_i$ and $\min_{\dot\kappa}$ determine i and $\dot\kappa$ as a function of j, then $\min_j$ determines $\underline{j}$, which can be used to get $\underline{i} = i(\underline{j})$ and $\underline{\dot\kappa} = \dot\kappa(\underline{j})$. This lower bound on $\min[I^a - I^b]$ is used in the next section to robustly compare weights.

5. An example

This section illustrates the application of the developed methodology to an artificial problem.

The graph in figure 3 models the domain by relationships of direct dependency, represented by directed arcs. Each node represents a binary (yes-no) variable that is associated with the probability distribution of the variable itself conditional on the state of the parent node. The distributions are given in table II.

A model made by the graph and the probability tables, as the one above, is called a *Bayesian network* [21]. We used the Bayesian network to sample units from the joint distribution of the variables in the graph. Each unit is a vector that represents a joint

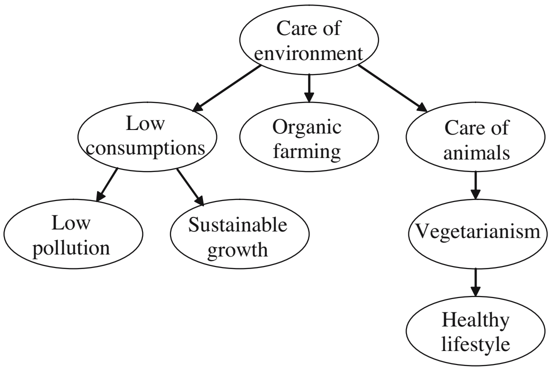

Figure 3. A graph that models the dependencies between the random variables of an artificial domain.

instance of all the variables. By the generated data set we can test our algorithm for the discovery of strong edges, and compare it with Chow and Liu's algorithm.

The 'strong edges algorithm' is summarized for clarity in table III. The main procedure is called 'DetectStrongEdges' and it implements the exact procedure from section 3.1. The comparison of edges needed by 'DetectStrongEdges' is implemented by the subprocedure 'TestDominance.' The test 2(a)vii there exploits the bounds defined in section 4.3 (we have added superscripts a and b to the terms of the bounds to make it clear to which edge they refer). For edges with a common node, the test 2(b)vi exploits the bounds given in section 4.4. For the dominance tests we have used the value 1 for the IDM hyper-parameter s (see section 4.1). We have also chosen $t^*_{ij} = \frac{1}{d_i d_j}, t^*_{ij\kappa} = \frac{1}{d_i d_j d_\kappa},$ etc.

Figures 4–7 show the progression of the models discovered by the two algorithms as more instances are read. The strong edges algorithm appears to behave more reliably than Chow and Liu's algorithm. It suspends the judgment on ambiguous

Table II
Conditional probability distribution for the variables of the example in figure 3.

Variable	P(variable = yes|parent = yes)	P(variable = yes|parent = no)
Care of environment	0.366	0.366
Low consumptions	0.959	0.460
Organic farming	0.950	0.450
Care of animals	0.801	0.332
Low pollution	1.000	0.208
Sustainable growth	0.951	0.200
Vegetarianism	0.993	0.460
Healthy lifestyle	0.920	0.300

The distribution of 'Care of environment' is represented in this table though it is actually unconditional.

Table III

A summary view of the strong edges algorithm. Remember that $\sigma = \frac{s}{n+s}$, n is the sample size, $u_{\dots} = \frac{n_{\dots}+t_{\dots}}{n+s}$ denotes the expectation of a certain chance, $u^*_{\dots}$ the expectation taken for a specific value $t^*_{\dots}$ of hyper-parameter $t_{\dots}$ finally, ψ denotes the ψ function, described in the Appendix.

```
1. Procedure **DetectStrongEdges** (a set-based weighted graph $G$)
   a) forest := $\emptyset$
   b) for each edge $e \in E$
         i) consider $G'$ obtained from $G$ dropping $e$ and the edges it dominates;
        ii) if the endpoints of $e$ are not connected in $G'$, add $e$ to forest;
   c) return forest.
2. Procedure **TestDominance** (edge $a$, edge $b$)
   a) if $a$ and $b$ do not share nodes then (i.e., the edges are $\imath \overset{a}{—} \jmath$ and $\tilde{\imath} \overset{b}{—} \tilde{\jmath}$)
         i) $I_0^a := \sum_i h(u^*_{i+}) + \sum_j h(u^*_{+j}) - \sum_{ij} h(u^*_{ij})$;
        ii) $I_0^b := \sum_{\tilde{i}} h(u^*_{\tilde{i}+}) + \sum_{\tilde{j}} h(u^*_{+\tilde{j}}) - \sum_{\tilde{i}\tilde{j}} h(u^*_{\tilde{i}\tilde{j}})$;
       iii) $\underline{I_1^a} := \quad \sigma \min_{ij}[h'(u^*_{i+}) + h'(u^*_{+j}) - h'(u^*_{ij})]$
                $-\sigma \sum_{ij} t^*_{ij}[h'(u^*_{i+}) + h'(u^*_{+j}) - h'(u^*_{ij})]$;
        iv) $\overline{I_1^b} := \quad \sigma \max_{\tilde{i}\tilde{j}}[h'(u^*_{\tilde{i}+}) + h'(u^*_{+\tilde{j}}) - h'(u^*_{\tilde{i}\tilde{j}})]$
                $-\sigma \sum_{\tilde{i}\tilde{j}} t^*_{\tilde{i}\tilde{j}}[h'(u^*_{\tilde{i}+}) + h'(u^*_{+\tilde{j}}) - h'(u^*_{\tilde{i}\tilde{j}})]$;
         v) $I_R^{a.lb} := \frac{1}{2}\sigma^2 \sum_i h''(\frac{n_{i+}}{n+s}) + \frac{1}{2}\sigma^2 \sum_j h''(\frac{n_{+j}}{n+s})$;
        vi) $I_R^{b.ub} := -\frac{1}{2}\sigma^2 \sum_{\tilde{i}\tilde{j}} h''(\frac{n_{\tilde{i}\tilde{j}}}{n+s})$;
       vii) if $I_0^a - I_0^b + \underline{I_1^a} - \overline{I_1^b} + I_R^{a.lb} - I_R^{b.ub} > 0$, return ‘true’;
   b) else (i.e., the edges are $\imath \overset{a}{—} \jmath \overset{b}{—} \kappa$)
         i) $I_0^a := \sum_i h(u^*_{i++}) + \sum_j h(u^*_{+j+}) - \sum_{ij} h(u^*_{ij+})$;
        ii) $I_0^b := \sum_j h(u^*_{+j+}) + \sum_{\dot{\kappa}} h(u^*_{++\dot{\kappa}}) - \sum_{j\dot{\kappa}} h(u^*_{+j\dot{\kappa}})$;
       iii) $\underline{I_1^a - I_1^b} := \sigma \min_j[\min_i(h'(u^*_{i++}) - h'(u^*_{ij+})) + \min_{\dot{\kappa}}(h'(u^*_{+j\dot{\kappa}}) - h'(u^*_{++\dot{\kappa}}))]$
            $-\sigma \sum_j[d_\kappa \sum_i t^*_{ij\dot{\kappa}}(h'(u^*_{i++}) - h'(u^*_{ij+})) + d_\imath \sum_{\dot{\kappa}} t^*_{ij\dot{\kappa}}(h'(u^*_{+j\dot{\kappa}}) - h'(u^*_{++\dot{\kappa}}))]$;
        iv) $I_R^{a.lb} := \frac{1}{2}\sigma^2 \sum_i h''(\frac{n_{i++}}{n+s}) + \frac{1}{2}\sigma^2 \sum_j h''(\frac{n_{+j+}}{n+s})$;
         v) $I_R^{b.ub} := -\frac{1}{2}\sigma^2 \sum_{j\dot{\kappa}} h''(\frac{n_{+j\dot{\kappa}}}{n+s})$;
        vi) if $I_0^a - I_0^b + \underline{I_1^a - I_1^b} + I_R^{a.lb} - I_R^{b.ub} > 0$, return ‘true’;
   c) return ‘false’.
3. Procedure $\mathbf{h}(u)$ return $u\psi(n+s+1) - u\psi(nu+su+1)$;
4. Procedure $\mathbf{h}'(u)$ return $\psi(n+s+1) - \psi(nu+su+1) - u(n+s)\psi'(nu+su+1)$;
5. Procedure $\mathbf{h}''(u)$ return $-2(n+s)\psi'(nu+su+1) - u(n+s)^2\psi''(nu+su+1)$;
```

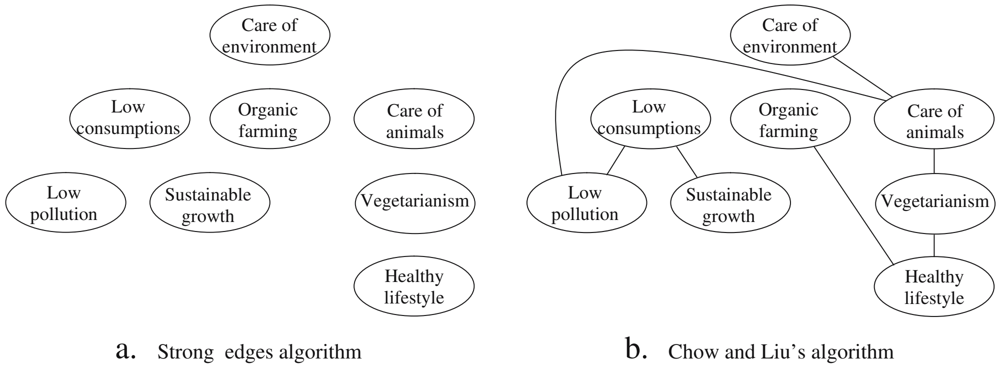

Figure 4. The outputs of the two algorithms after reading 20 instances.

cases and outputs forests. These are always composed of edges of the actual graph. Chow and Liu's algorithm always produces complete trees, but these misrepresent the actual tree until 50 instances have been read. At this point Chow and Liu's algorithm detects the right tree. The cautious approach implemented by the strong edges algorithm needs other 20 instances to produce the same complete tree.

6. Extensions

The methodology developed so far leads naturally to other possible extensions of Chow and Liu's approach. We briefly report on two different types of extensions in the following.

Section 6.1 discusses the question of tree-dependency structures vs. forest-dependency structures under several respects. The discussion focuses both on algorithms that are alternative to the strong edges one, and that aim at yielding trees, and on the other hand on algorithms that emphasize the inference of forest-dependency structures from data.

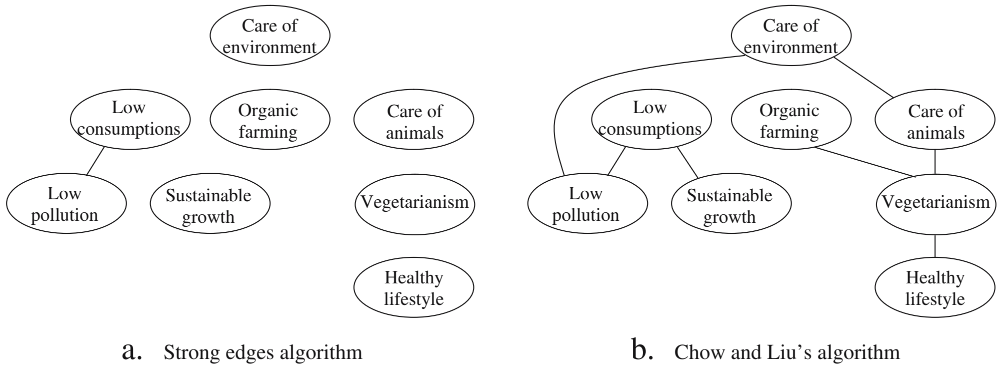

Figure 5. The outputs of the two algorithms after reading 30 instances.

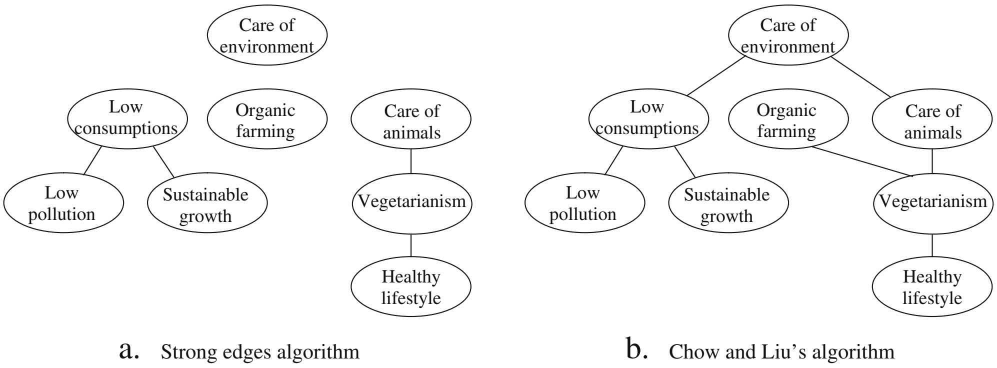

Figure 6. The outputs of the two algorithms after reading 40 instances.

In section 6.2 we extend the computation of lower and upper expectations of mutual information to the computation of robust credible limits. These are intervals for mutual information obtained from the IDM that contain the actual value with given probability. This result is useful in order to produce dependency structures that provide the user with a given guarantee level. In principle the extension to credible limits can be applied both to the computation of strong edges and to that of robust trees, as defined in the next section, although the results of sections 6.1 and 6.2 are actually independent, in the sense that one does not need to use them together.

6.1. *Forests vs. trees*

It may be useful to critically re-consider Chow and Liu's algorithm in the following respect. Chow and Liu's algorithm yields always a tree by construction, and hence this happens also when the actual (but usually unknown) dependency structure is a forest. This is a questionable characteristic of the algorithm, as in the mentioned case yielding a tree seems to be hard to justify. There are indeed approaches in the

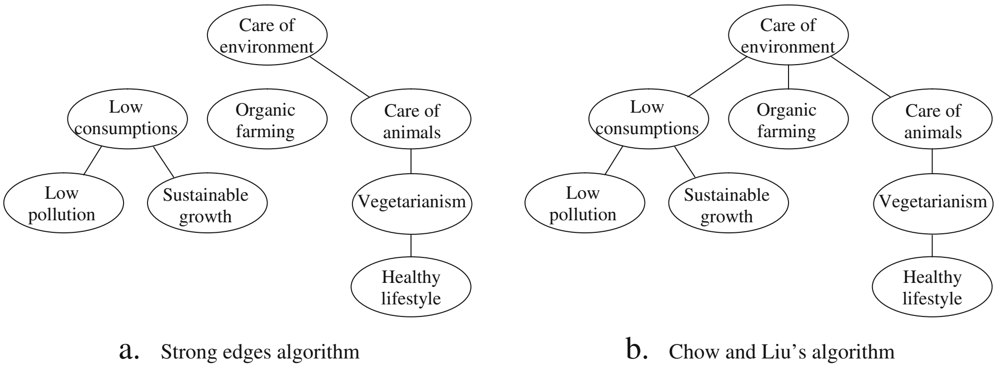

Figure 7. The outputs of the two algorithms after reading 50 instances.

literature of precise probability that suppress the edges of a maximum spanning tree for which the mutual information is not large enough, yielding a forest. This is typically implemented using a numerical threshold ε, sometimes computed via statistical tests. Such approaches can be used immediately also within the imprecise-probability framework introduced in this paper; it is sufficient to suppress the edges for which the upper value of mutual information [i.e., $\max_{w \in W} w(e)$] does not exceed ε. In contrast with the precise-probability approach, the latter should have the advantage to better deal with the problem to suppress edges by mistake, as a consequence of the variability of the inferred values of mutual information. This should be especially true once forests are inferred using the credible limits for mutual information introduced in the next section.

A more subtle question is how the forests inferred using the above threshold procedure relate to the forests that are naturally produced by the strong edges algorithm in its original form. Remind that the strong edges algorithm produces a forest rather than a tree when there is more than one optimal tree consistent with the available data; indeed the algorithm aims at yielding the graphical structure made of the intersection of all such trees. The situation may be clarified by focusing on a special case: consider a problem in which the true dependency structure is a tree in which there are edges with the same value of mutual information, say μ. In this case the strong edges algorithm will never produce a tree, only a forest, also in the limit of infinitely many data. The reason is that there will always be multiple optimal trees consistent with the data, just because multiple optimal trees are a characteristic of the problem. In particular, there would arise a forest because some edges with weight μ would never belong to the set of strong edges. Now suppose that $\mu > \varepsilon$. In this case, the previous threshold procedure would not suppress the edges with mutual information equal to μ. In other words, the two procedures suppress edges under different conditions: the strong edges algorithm may suppress edges because they have equal true values of mutual information, despite those values may be high; the threshold procedure only suppresses edges with low value of mutual information. For this reason it could make sense to apply the threshold procedure also as a post-processing step of the strong edges algorithm.

The discussion so far has highlighted an interesting point. By focusing on the intersection of all the trees consistent with the data, the strong edges algorithm appears to be well suited as a tool to recover the actual dependency structure underlying the data. This is because the algorithm does not aim at recovering just any of the equivalent structures, rather, it focuses on the common pattern to all of them, which is obviously part of the actual structure. In this sense, the strong edges algorithm might be well suited for applications concerned with the recovery of causal patterns.

On the other hand, one can think of applications for which the algorithm is probably not so well suited. For instance, in (precise-probability) problems of pattern classification based on Bayesian networks [6], it is important to recover any tree (or forest) structure for which the sum of the edge weights is maximized. In this case, suppressing edges with large weights only because they are not strong might lead to low classification accuracy. In these cases, the extension of those precise approaches to the IDM-based inferential approach should probably follow other lines than those described here. One possibility could be to exploit existing results in the literature of robust optimization; the work of Yaman et al. [28] seems to be particularly worthy of consideration. Yaman et al. consider a problem of maximum spanning tree for a graph with weights specified by intervals (the weights are given no particular interpretation), which is a special case of a set-based weighted graph. They define the *relative robust spanning tree* as follows (using our notations): let T be a generic tree spanning G, and denote by T_w^* a maximum spanning tree of $G_w \in \mathcal{G}$. Let S_w^* resp. S_w be the sum of the edge weights of T_w^* resp. T, with respect to the weight function w. A relative robust spanning tree T^* is one that solves the optimization problem $\min_T \max_{w \in W}(S_w^* - S_w)$, i.e., one that minimizes the largest deviation $S_w^* - S_w$ among all the possible graphs $G_w \in \mathcal{G}$. In this sense the approach adopted by Yaman et al. is in the long tradition of the popular *maximin* (or *minimax*) decision criterion. From the computational point of view, although the problem is *NP-complete* [2], recent results show that relatively large instances of the problem can be solved efficiently [19]. The trees defined by Yaman et al. could probably be combined with the IDM-based inferential approach presented here, suitably modified for classification problems, in order to yield relative robust classification trees. Here, too, it could make sense to post-process the relative robust trees in order to suppress edges with small upper (or even lower) values of mutual information, yielding a forest.

6.2. Robust credible limits for mutual information

In this section we develop a full inferential approach for mutual information under the IDM.

An α-credible interval for the mutual information $\mathcal{I}$ is an interval $[\underset{\sim}{\mathcal{I}}, \tilde{\mathcal{I}}]$ which contains $\mathcal{I}$ with probability at least α, i.e., $\int_{\underset{\sim}{\mathcal{I}}}^{\tilde{\mathcal{I}}} p(\mathcal{I}) d\mathcal{I} \geq \alpha$. We define α-credible intervals w.r.t. distribution $p_t(\mathcal{I})$ as

$$[\underset{\sim}{\mathcal{I}}_t, \tilde{\mathcal{I}}_t] = [E_t[\mathcal{I}] - \Delta \underset{\sim}{\mathcal{I}}_t, E_t[\mathcal{I}] + \widetilde{\Delta \mathcal{I}}_t \quad \text{such that} \quad \int_{\underset{\sim}{\mathcal{I}}_t}^{\tilde{\mathcal{I}}_t} p_t(\mathcal{I}) d\mathcal{I} \geq \alpha,$$

where $\widetilde{\Delta \mathcal{I}}_t := \tilde{\mathcal{I}}_t - E_t[\mathcal{I}]$ ($\underset{\sim}{\Delta \mathcal{I}}_t := E_t[\mathcal{I}] - \underset{\sim}{\mathcal{I}}_t$) is the distance from the right boundary $\tilde{\mathcal{I}}_t$ (left boundary $\underset{\sim}{\mathcal{I}}_t$) of the α-credible interval $[\underset{\sim}{\mathcal{I}}_t, \tilde{\mathcal{I}}_t]$ to the mean $E_t[\mathcal{I}]$ of $\mathcal{I}$ under distribution p_t. We can use

$$[\underset{\approx}{\mathcal{I}}, \overset{\approx}{\mathcal{I}}] := [\min_t \underset{\sim}{\mathcal{I}}_t, \max_t \tilde{\mathcal{I}}_t] = \bigcup_t [\underset{\sim}{\mathcal{I}}_t, \tilde{\mathcal{I}}_t]$$

as a robust credible interval, since $\int_{\underset{\approx}{\mathcal{I}}}^{\bar{\tilde{\mathcal{I}}}} p_{\boldsymbol{t}}(\mathcal{I})d\mathcal{I} \geq \int_{\underset{\sim}{\mathcal{I}}_t}^{\tilde{\mathcal{I}}_t} p_{\boldsymbol{t}}(\mathcal{I})d\mathcal{I} \geq \alpha$ for all $\boldsymbol{t}$. An upper bound for $\bar{\tilde{\mathcal{I}}}$ (and similarly lower bound for $\underset{\approx}{\mathcal{I}}$) is

$$\bar{\tilde{\mathcal{I}}} = \max_{\boldsymbol{t}}(E_{\boldsymbol{t}}[\mathcal{I}] + \widetilde{\Delta\mathcal{I}}_{\boldsymbol{t}}) \leq \max_{\boldsymbol{t}} E_{\boldsymbol{t}}[\mathcal{I}] + \max_{\boldsymbol{t}} \widetilde{\Delta\mathcal{I}}_{\boldsymbol{t}} = \overline{E[\mathcal{I}]} + \overline{\widetilde{\Delta\mathcal{I}}}$$

Good upper bounds on $\bar{I} = \overline{E[\mathcal{I}]}$ have been derived in section 4.3.

For not too small n, $\underset{\sim}{p_{\boldsymbol{t}}}(\mathcal{I})$ is close to Gaussian due to the central limit theorem. So we may approximate $\widetilde{\Delta\mathcal{I}}_{\boldsymbol{t}} \approx r\sigma_{\boldsymbol{t}}$ with r given by $\alpha = \operatorname{erf}(r/\sqrt{2})$, where erf is the error function (e.g., $r = 2$ for $\alpha \approx 95\%$) and $\sigma_{\boldsymbol{t}}$ is the variance of $p_{\boldsymbol{t}}$, keeping in mind that this could be a non-conservative approximation. In order to determine $\overline{\widetilde{\Delta\mathcal{I}}}$ we only need to estimate $\max_{\boldsymbol{t}}\sqrt{\mathrm{Var}_{\boldsymbol{t}}[\mathcal{I}]} = O(\frac{1}{n})$. The variation of $\sqrt{\mathrm{Var}_{\boldsymbol{t}}[\mathcal{I}]}$, with $\boldsymbol{t}$ is of order $n^{-3/2}$. If we regard this as negligibly small, we may simply fix some $\boldsymbol{t}^* \in \Delta$. So the robust credible interval for $\mathcal{I}$ can be estimated as

$$\bar{\tilde{\mathcal{I}}} \leq \bar{I} + \overline{\widetilde{\Delta\mathcal{I}}} \leq I_0 + \overline{I_1} + I_R^{ub} + \overline{\widetilde{\Delta\mathcal{I}}} \approx I_0 + \overline{I_1} + I_R^{ub} + r\sqrt{\mathrm{Var}_{\boldsymbol{t}*}[I]}.$$

Expressions for the variance of $\mathcal{I}$ have been derived in [9, 11]:

$$\mathrm{Var}_{\boldsymbol{t}}[\mathcal{I}] = \frac{1}{n+s}\sum_{ij} u_{ij}\left(\log\frac{u_{ij}}{u_i + u_{+j}}\right)^2 - \frac{1}{n+s}\left(\sum_{ij} u_{ij}\log\frac{u_{ij}}{u_{i+}u_{+j}}\right)^2 + O(n^{-2}).$$

Higher order corrections to the variance and higher moments have also been derived, but are irrelevant in light of our other approximations. In sections 4.4 and 5 we also needed a lower bound on $I^a - I^b$. Taking credible intervals into account we need a robust upper α-credible limit for $\mathcal{I}^{ba} := \mathcal{I}^b - \mathcal{I}^a$. Similarly as for the variance one can derive the following expression:

$$\begin{aligned}\overline{\widetilde{\mathcal{I}^{ba}}} \leq\; & I_0^b - I_0^a + \overline{I_1^b} - \underline{I_1^a} + I_R^{b\cdot ub} - I_R^{a\cdot lb} \\ & + r\sqrt{\mathrm{Var}_{\boldsymbol{t}^*}[\mathcal{I}^b - \mathcal{I}^a]} + O(n^{-3/2}),\end{aligned},$$

$$\mathrm{Var}_{\boldsymbol{t}}[\mathcal{I}^b - \mathcal{I}^a] = \mathrm{Var}_{\boldsymbol{t}}[\mathcal{I}^b] + \mathrm{Var}_{\boldsymbol{t}}[\mathcal{I}^a] - 2\,\mathrm{Cov}_{\boldsymbol{t}}[\mathcal{I}^b, \mathcal{I}^a],$$

$$\begin{aligned}\mathrm{Cov}_{\boldsymbol{t}}[\mathcal{I}^b, \mathcal{I}^a] = & \frac{1}{n+s}\sum_{ij\dot{\kappa}} u_{ij\dot{\kappa}}\left(\log\frac{u_{ij}^a}{u_{i+}^a u_{+j}^a}\log\frac{u_{j\dot{\kappa}}^b}{u_{j+}^b u_{+\dot{\kappa}}^b}\right) \\ & - \frac{1}{n+s}\left(\sum_{ij} u_{ij}^a \log\frac{u_{ij}^a}{u_{i+}^a u_{+j}^a}\right)\left(\sum_{j\dot{\kappa}} u_{j\dot{\kappa}}^b \log\frac{u_{j\dot{\kappa}}^b}{u_{j+}^b u_{+\dot{\kappa}}^b}\right) + O(n^{-2}).\end{aligned}$$

Variances are typically of order $1/n$, so for large n, credible intervals $\overset{\approx}{I} - \underset{\approx}{I} = O(1/\sqrt{n})$ are much wider than expected intervals $\bar{I} - \underline{I} = O(1/n)$.

7. Conclusions

This paper has tackled the problem to reliably infer trees from data. We have provided an exact procedure that infers strong edges in time $O(m^4)$, and have shown that it performs well in practice on an example problem. We have also developed an approximate algorithm that works in time $O(m^3)$.

Reliability follows from using the IDM, a robust inferential model that rests on very weak prior assumptions. Working with the IDM involves computing lower and upper estimates, i.e., solving global optimization problems. These can hardly be tackled exactly, as they are typically non-linear and non-convex. A substantial part of the present work has been devoted to provide systematic approximations to the exact intervals with a guaranteed worst case of $O(\sigma^2)$. This was achieved by optimizing approximating functions, obtained by Taylor-expanding the original objective function. We have taken care to make these approximations conservative, i.e., they always include the exact interval. This is the necessary step to ultimately obtain over-cautious rather than overconfident models.

More broadly speaking, the same approach has been used also for another approximation, concerned with the representation level chosen for the IDM. In principle, one might use the IDM for the joint realization of all the m random variables. In this paper we have used one IDM for each bivariate (and tri-variate, in some cases) realization. Using separate IDMs simplifies the treatment, but it may give rise to global inconsistencies (in the same lines of the discussion on comparing edges with a common vertex, in section 4.4). However, their effect is only to make $\mathcal{O}$ strictly include $\mathcal{O}_{\mathcal{T}}$, thus producing an excess of caution, as discussed in section 3.1.

We have already reported two developments that follow naturally from the work described above. The first involves the computation of robust trees, which widens the scope of this paper to other applications. The second is in the direction of even greater robustness by providing robust credibile limits for mutual information, which provide the user with a guarantee level on the inferred dependency structures.

Other extensions of the present work could be considered that need further research in order to be realized. Obviously, it would be worth extending the work to the robust inference of more general dependency structures. This could be achieved, for example, in a way similar to Kleiter's work [14]. One could also extend our approach to dependency measures other than mutual information, like the statistical coefficient ϕ^2 [13, pp. 556–561]. This would require new approximations to be derived for the new index under the IDM, but the first part of the paper on the detection of strong edges could be applied as it is.

Another important extension could be realized by considering the inference of dependency structures from incomplete samples. Recent research has developed robust approaches to incomplete samples that make very weak assumptions on the mechanism responsible for the missing data [18, 23, 29]. This would be an important step towards realism and reliability in structure inference.

Appendix

Properties of the digamma ψ function

The digamma function ψ is defined as the logarithmic derivative of the Gamma function. Integral representations for ψ and its derivatives are

$$\psi(z) = \frac{d \ln \Gamma(z)}{dz} = \frac{\Gamma'(z)}{\Gamma(z)} = \int_0^\infty \left[\frac{e^{-t}}{t} - \frac{e^{-zt}}{1-e^{-t}}\right] dt,$$
$$\psi^{(\ell)}(z) = (-1)^{\ell+1} \int_0^\infty \frac{t^\ell e^{-zt}}{1-e^{-t}} dt \quad \text{for} \quad \ell > 0.$$

The h function (1) and its derivatives are

$$h^{(\ell)}(u) = u^{(\ell)}\psi(n+s+1) - \ell(n+s)^{\ell-1}\psi^{(\ell-1)}((n+s)u+1)$$
$$-u(n+s)^\ell\psi^{(\ell)}((n+s)u+1).$$

At argument $u_i = \frac{n_i+st_i}{n+s}$ we get for h, h' and h''

$$h(u_i) = (n_i+st_i)[\psi(n+s+1) - \psi(n_i+st_i+1)]/(n+s),$$
$$h'(u_i) = \psi(n+s+1) - \psi(n_i+st_i+1) - (n_i+st_i)\psi'(n_i+st_i+1),$$
$$h''(u_i) = -2(n+s)\psi'(n_i+st_i+1) - (n_i+st_i)(n+s)\psi''(n_i+st_i+1),$$

For integral arguments the following closed representations for ψ, ψ', and ψ'' exist:

$$\psi(n+1) = -\gamma + \sum_{i=1}^n \frac{1}{i}, \psi'(n+1) = \frac{\pi^2}{6} - \sum_{i=1}^n \frac{1}{i^2}, \psi''\ (n+1) = -2\zeta(3) + 2\sum_{i=1}^n \frac{1}{i^3}$$

where γ = 0.5772156... is Euler's constant and $\zeta(3)$ = 1.202569... is Riemann's zeta function at 3. Closed expressions for half-integer values and fast approximations for arbitrary arguments also exist. The following asymptotic expansion can be used if one is interested in $O((\frac{s}{n+s})^2)$ approximations only (and not rigorous bounds):

$$\psi(z+1) = \log z + \frac{1}{2z} - \frac{1}{12z^2} + O(\frac{1}{z^4}).$$

See [1] for details on the ψ function and its derivatives. From the above expressions one may show $h'' < 0$ and $h''' > 0$.

References

[1] M. Abramowitz and I.A. Stegun, eds., *Handbook of Mathematical Functions* (Dover, 1974).

[2] I.D. Aron and P. Van Hentenryck, On the complexity of the robust spanning tree problem with internal data, Operations Research Letters 32 (2004) 36–40.

[3] J.-M. Bernard, 2001, Non-parametric inference about an unknown mean using the imprecise Dirichlet model, in: *ISIPTA'01*, eds. G. de Cooman, T. Fine and T. Seidenfeld (The Netherlands, 2001) pp. 40–50.

[4] J.-M. Bernard, An introduction to the imprecise Dirichlet model for multinomial data, International Journal of Approximate 39(2–3) (2005) 123–150.

[5] C.K. Chow and C.N. Liu, Approximating discrete probability distributions with dependence tress, IEEE Transactions on Information Theory, IT-14(3) (1968) 462–468.

[6] N. Friedman, D. Geiger and M. Goldszmidt, Bayesian networks classifiers, Machine Learning 29(2/3) (1997) 131–163.

[7] A. Gelman, J.B. Carlin, H.S. Stern and D.B. Rubin, *Bayesian Data Analysis* (Chapman, 1995).

[8] J.B.S. Haldane, The precision of observed values of small frequencies, Biometrika 35 (1948) 297–300.

[9] M. Hutter, Distribution of mutual information, in: *Proceedings of NIPS*2001*, eds. T.G. Dietterich, S. Vecker and Z. Ghahramani (Cambridge, MA, 2001).

[10] M. Hutter, Robust estimators under the imprecise dirichlet model, in: *Proc. 3rd International Symposium on Imprecise Probalities and Their Application (ISIPTA-2003)*, *Proceedings in Informatics* Vol. 18 (Canada, 2003) pp. 274–289.

[11] M. Hutter and M. Zaffalon, Distribution of mutual information from complete and incomplete data, Computational Statics & Data Analysis 48(3) (2005) 633–657.

[12] H. Jeffreys, An invariant form for the prior probability in estimation problems, in: *Proceedings Royal Society London A*, 186 (1946) pp. 453–461.

[13] M.G. Kendall and A. Stuart, *The Advanced Theory of Statistics*, 2nd edition. (Griffin, London, 1967).

[14] G.D. Kleiter, The posterior probability of Bayers nets with strong dependences, Soft Computing 3 (1999) 162–173.

[15] J.B. Kruskal Jr., On the shortest spanning subtree of a graph and the traveling salesman problem, in: *Proceedings of the American Mathematical Society* 7 (1956) 48–50.

[16] S. Kullback, *Information Theory and Statistics* (Dover, 1968).

[17] S. Kullback and R.A. Leiber, On information and sufficiency, Annals of Mathematical Statistics 22 (1951) 79–86.

[18] C. Manski, *Partial Identification of Probability Distributions* (Department of Economics, Northwestern University, USA: Draft book, 2002).

[19] R. Montemanni, A Benders decomposition approach for the robust spanning tree problem with interval data, European Journal of Operational Research. Forthcoming.

[20] H. Papadimitriou and K. Steiglitz, *Combinatorial Optimization: Algorithms and Complexity* (Prentice Hall, New York, 1982).

[21] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference* (Morgan Kaufmann, San Mateo, 1988).

[22] W. Perks, Some observations on inverse probability, Journal of the Institute of Actuaries 73 (1947) 285–312.

[23] M. Ramoni and P. Sebastiani, Robust learning with missing data, Machine Learning 45(2) (2001) 147–170.

[24] T. Verma and J. Pearl, Equivalence and synthesis of causal models, in: *UAI'90*, eds. P.P. Bonissone, M. Henrion, L.N. Kanal and J.F. Lemmer (New York, 1990) pp. 220–227.

[25] P. Walley, *Statistical Reasoning with Imprecise Probabilities* (Chapman and Hall, New York, 1991).

[26] P. Walley, Inferences from multinomial data: learning about a bag of marbles, Journal of the Royal Statistical Society B 58(1) (1996) 3–57.

[27] D.H. Wolpert and D.R. Wolf, Estimating functions of distributions from a finite set of samples, Physical Review E 52(6) (1995) 6841–6854.

[28] H. Yaman, O.E. Karaşan and M.C. Pinar, The robust spanning tree problem with interval data, Operations Research Letters 29 (2001) 31–40.

[29] M. Zaffalon, Exact credal treatment of missing data, Journal of Statistical Planning and Inference 105(1) (2002) 105–122.