Evaluation Measures for Learning Probabilistic and Possibilistic Networks

Christian Borgelt and Rudolf Kruse
Dept. of Information and Communication Systems
Otto-von-Guericke-University of Magdeburg
D-39106 Magdeburg, Germany
e-mail: borgelt@iik.cs.uni-magdeburg.de

Abstract: Evidence propagation in inference networks, probabilistic or possibilistic, can be done in two different ways — using a product/sum scheme or using a minimum/maximum scheme — depending on the type of answers one expects from the network. Usually the first is seen in connection with probabilistic, the second in connection with possibilistic reasoning, although we argue that both schemes are applicable in both settings. We discuss learning inference networks from data and examine some evaluation measures with respect to the chosen propagation method.

Keywords: probabilistic networks, possibilistic networks, evidence propagation, evaluation measures, learning from data

1 Introduction

Since reasoning in multi-dimensional domains tends to be infeasible in the domains as a whole — and the more so, if uncertainty is involved — decomposition techniques, that reduce the reasoning process to computations in lower-dimensional subspaces, have become very popular. For example, decomposition based on dependence and independence relations between variables has extensively been studied in the field of graphical modeling [16]. Some of the best-known approaches are Bayesian networks [22], Markov networks [19], and the more general valuation-based networks [27]. They all led to the development of efficient implementations, for example HUGIN [1], PULCINELLA [26], PATHFINDER [11] and POSSINFER [7].

In this paper we examine two propagation methods used for reasoning in probabilistic and possibilistic networks. Although these two methods are usually seen as tied to the underlying uncertainty calculus, i.e. probability or possibility theory, we argue that both are applicable in both domains. Hence inference networks should not only be characterized by the uncertainty calculus but also by the propagation method used.

Since a large part of recent research has been devoted to learning inference networks from data [4, 12, 8], we examine, with respect to the propagation method of the network to be learned, some evaluation measures that can be used in learning algorithms.

2 Degrees of Possibility

Before we discuss propagation in inference networks, we explain our interpretation of a degree of possibility, which is based on the context model [6, 17]. In this model possibility distributions are interpreted as information-compressed representations of (not necessarily nested) random sets, a degree of possibility as the one-point coverage of a random set [21].

More intuitively, a degree of possibility is the least upper bound on the probability of the possibility of a value. We explain this interpretation in three steps. In the first place, the possibility of a value is just what we understand by this term in daily life: whether a value is possible or not. At this point we do not assume intermediate degrees, i.e. if a value is possible, we cannot say more than that. We can not give a probability for that value. All we know is that if a value is not possible, its probability must be zero.

Secondly, imagine that we can distinguish between certain disjoint contexts or scenarios, to each of which we can assign a probability and for each of which we can state whether in it the value under consideration is possible or not. Then we can assign to the value as a degree of possibility the sum of the probabilities of the contexts in which it is possible. Thus we arrive at a degree of possibility as the probability of the possibility of a value.

Thirdly, we drop the requirement that the contexts or scenarios must be disjoint. They may overlap, but

we assume, that we do not know how. This seems to be a sensible assumption, since we should be able to split contexts, if we knew how they overlap. If we now assign to a value as the degree of possibility the sum of the probabilities of the contexts in which it is possible, this value may exceed the actual probability, because of the possible overlap. But since we do not know which contexts overlap and how they overlap, this is the least upper bound consistent with the available information.

Note, that in this interpretation probability distributions are just special possibility distributions. If we have disjoint contexts and if in all contexts in which a value is possible it has the probability 1, the degree of possibility is identical to the probability. Note also, that in this interpretation the degree of possibility can not be less than the probability.

3 Two Propagation Methods

The basic presupposition underlying every inference network, probabilistic or possibilistic, is that a multi-dimensional distribution can be decomposed without much loss of information into a set of (overlapping) lower-dimensional distributions.[1] This set of lower-dimensional distributions is usually represented as a hypergraph, in which there is a node for each variable and a hyperedge for each distribution of the decomposition. To each node and to each hyperedge a projection of the multi-dimensional distribution (a *marginal distribution*) is assigned: to the node a projection to its variable and to a hypergraph a projection to the set of variables connected by it. Thus hyperedges represent direct influences that the connected variables have on each other, i.e. how constraints on the value of one variable affect the probabilities or possibilities of the values of the other variables in the hyperedge. Reasoning in such a hypergraph is done by propagating evidence, i.e. observed constraints on the possible values of a subset of all variables, along the hyperedges.

The idea of propagation can be understood best by a simple example. Imagine three variables, A, B, and C (all with a finite number of values — throughout this paper we assume that all variables have a finite number of values), and a (hyper)graph A—B—C. When evidence about A is fed into the network it is propagated like this: The constraints on the values of variable A stated by the evidence are extended to the space $A \times B$ to obtain constraints on tuples (a_i, b_j), which are then projected to the variable B to compute the constraints on the values of this variable. These constraints are then in turn extended to the subspace $B \times C$ and projected to variable C.

For this scheme to be feasible, the main operations, projection and extension of distributions, have to satisfy certain preconditions [27]. We consider here two pairs of operations satisfying these preconditions: product/sum and minimum/maximum.

The first pair, product/sum, is used in probability theory, in which the marginal distribution of e.g. a two-dimensional distribution is calculated by summing over one dimension, that is $P(a_i) = \sum_j P(a_i, b_j)$. Extension consists in multiplying the prior probability distribution on the superset with the quotient of posterior and prior probability on the subset.

An example is given in figures 1 and 2. Figure 1 shows a three-dimensional probability distribution on the joint domain of the variables $A = \{a_1, a_2, a_3, a_4\}$, $B = \{b_1, b_2, b_3\}$, and $C = \{c_1, c_2, c_3\}$, and the marginal distributions calculated by summing over lines/columns. Since in this distribution the equations $\forall i, j, k : P(a_i, b_j, c_k) = \frac{P(a_i, b_j) P(b_j, c_k)}{P(b_j)}$ hold, it can be decomposed into the marginal distributions on the subspaces $A \times B$ and $B \times C$. Therefore it is possible to propagate the observation that variable A has value a_4 using the scheme in figure 2.[2] One can easily check that the resulting marginal distributions are the same as those that can be computed from the three-dimensional distribution directly.

The second pair of operations, minimum/maximum, is used in possibility theory. The projection of e.g. a two-dimensional distribution is calculated by determining the maximum over one dimension, extension by calculating the minimum of the prior joint distribution on the superset and the posterior marginal distribution.

An example is given in figures 3 and 4. Figure 3 shows a three-dimensional possibility distribution on the joint domain of the variables A, B, and C and various marginal distributions determined by computing the maximum over lines/columns. Since in this distribution the equations $\forall i, j, k : \pi(a_i, b_j, c_k) = \min_j(\max_k \pi(a_i, b_j, c_k), \max_i \pi(a_i, b_j, c_k))$ hold, it can be decomposed into marginal distributions on the subspaces $A \times B$ and $B \times C$. Therefore it is possible to propagate the observation that variable A has value a_4 using the scheme in figure 4. Again the marginal distributions obtained are the same as those that can be computed directly from the three-dimensional distribution.

[1] Of course, this presupposition need not hold. A distribution need not be decomposable, even if one accepts a certain limited loss of information. But in such a situation inference networks cannot be used.

[2] This scheme is a simplification and does not lend itself to direct implementation. Especially joining evidence from two (hyper)edges needs additional computations, which we omitted here.

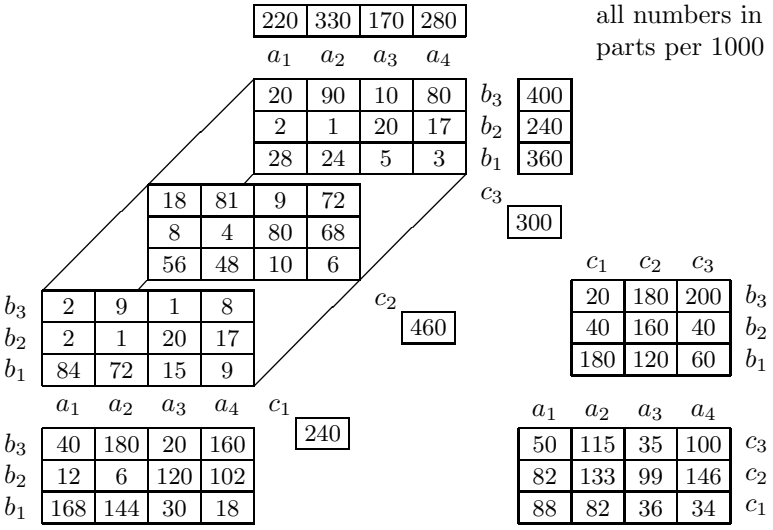


Figure 1: A three-dimensional probability distribution with marginal distributions calculated by summing over lines/columns. Since in this distribution the equations $\forall i, j, k:$ $P(a_i, b_j, c_k) = \frac{P(a_i, b_j) P(b_j, c_k)}{P(b_j)}$ hold, it can be decomposed into the marginal distributions on the subspaces $A \times B$ and $B \times C$.

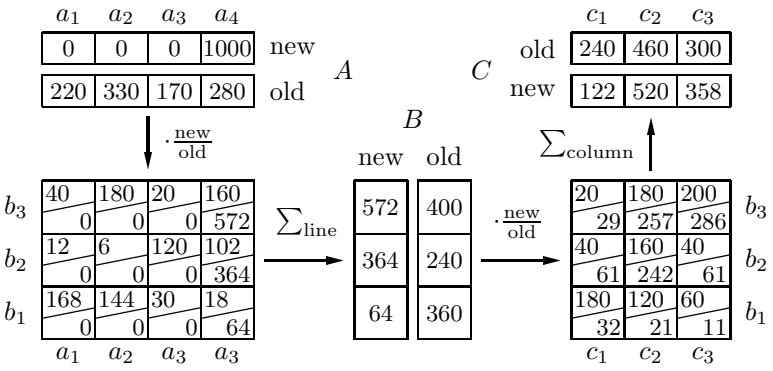


Figure 2: Product/sum propagation of the evidence that variable A has value a_4 in the three-dimensional probability distribution shown in figure 1 using the marginal probability distributions on the subspaces $A \times B$ and $A \times C$.

Note, that although we used a probability distribution as an example for product/sum propagation and a possibility distribution as an example for minimum/maximum propagation, both propagation schemes can be used in both cases. Whether a certain scheme is applicable or not depends only on the equations that have to hold (at least approximately) to render the distribution decomposable, but not on the interpretation of the distribution.

The only problem that can occur is that when using sum projection the values of marginal possibility distributions can get greater than 1. The obvious solution to this problem would be to use a bounded sum, but this would interfere with the propagation scheme. We therefore assume that the possibility degrees are bounded only when the results of a propagation are read from the network, but not during propagation.

Summing degrees of possibility should not be rejected in general. Given our interpretation of a degree of possibility, the only thing we can do, if we are given a joint possibility distribution and want to know the degree of possibility of a value of some variable, is to sum over the values of all other variables. This is consistent with our interpretation, since the result (bounded by 1, if necessary) is indeed the least upper bound on the probability of the possibility of that value that can be inferred from the available information. We could derive a lower bound only if we had information about the underlying contexts.

Which propagation scheme to choose, if both are possible, depends on the questions one wants the network to answer. If we are interested in asking questions like "What is the probability (or the degree of possibility) that variable A has value a?", the first scheme should be used, but if we are interested in asking questions like "What is the probability (or degree of possibility) of the most probable value vector (or the value vector with the highest degree of possibility) in which variable A has value a?", we may prefer the second.[3]

[3]Note, that it is always possible to answer questions of the

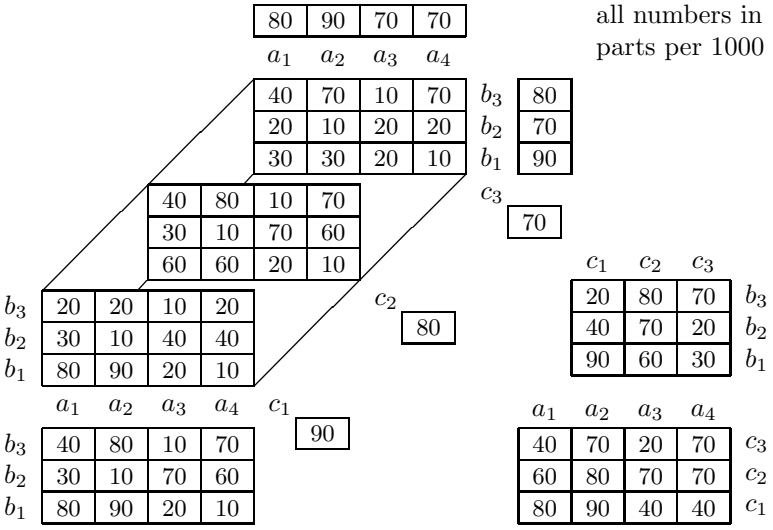


Figure 3: A three-dimensional possibility distribution with marginal distributions calculated by determining the maximum over lines/columns. Since in this distribution the equations $\forall i, j, k : \pi(a_i, b_j, c_k) = \min_j(\max_k \pi(a_i, b_j, c_k), \max_i \pi(a_i, b_j, c_k))$ hold, it can be decomposed into marginal distributions on the subspaces $A \times B$ and $B \times C$.

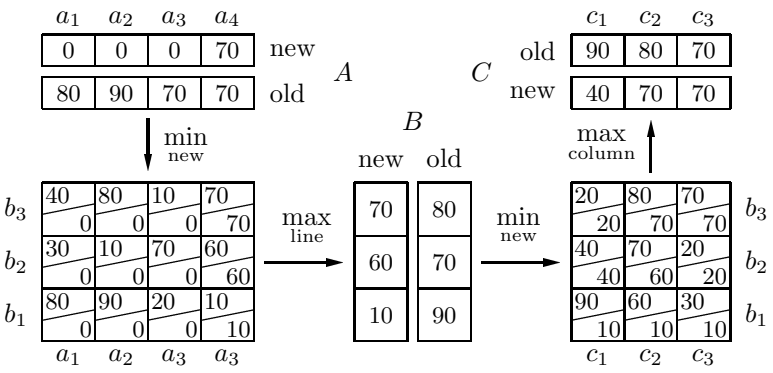


Figure 4: Minimum/maximum propagation of the evidence that variable A has value a_4 in the three-dimensional possibility distribution shown in figure 3 using the marginal distributions on the subspaces $A \times B$ and $A \times C$.

Note, that the minimum/maximum propagation scheme enforces to define some special terms and notations. When using sum projection for probability distributions one can speak of a marginal probability distribution, but this is not possible, if maximum projection is used. Although the values of a maximum-projected distribution are probabilities, they need not add up to one. We will therefore call them *probability maxima* and denote them by $P_{\max}$. Analogously we use the term *possibility degree maximum* and $\pi_{\max}$.

4 Evaluation Measures

An algorithm for learning inference networks consists always of two parts: an evaluation measure and a search method. The evaluation measure estimates the quality of a given decomposition (a given hypergraph) and the search method determines which decompositions (which hypergraphs) are inspected. Often the search is guided by the value of the evaluation measure, since it is usually the goal to maximize (or to minimize) its value.

A desirable property of an evaluation function is a certain locality, i.e. the possibility to evaluate subgraphs, at best single hyperedges, separately. This is desirable, not only because it facilitates computation, but also because some search methods can make use of such locality. For example, in [3] the value of an evaluation measure is computed on all two-dimensional distributions and then the Kruskal algorithm is applied to determine a maximum weight spanning tree.

In this section we review some evaluation functions that can be used for learning inference networks from data. All of them estimate the quality of single hyperedges and are based on the empirical probability or possibility distributions found in the database. That is, if N is the total number of tuples in the database

second type from a network designed for questions of the first type by determining the joint distribution. At least this holds in theory, in practice the computational costs may be too high.

and N_i the number of tuples in which variable A has value a_i, then $P(a_i) = \frac{N_i}{N}$.

We do not examine search methods, since the evaluation functions presented here can be used with any general heuristic search algorithm, e.g. simulated annealing or genetic programming. Of course, there are also special search methods like the K2 algorithm [4] (start with a topological order on the variables to restrict the permissible network structures and then carry out a greedy search to determine the parents of each variable), but limits of space prevent us from discussing them in detail.

4.1 Product/Sum Propagation

The basic idea of nearly all evaluation measures used for learning networks based on product/sum propagation is to compare the joint distribution with the product of the marginal distributions. This seems to be reasonable, since the more these two distributions differ, the more dependent the variables are on each other. Nevertheless this can lead to superfluous edges, because a dependence that is not genuine but mediated through another variable, i.e. a situation of conditional independence, can not be recognized as such.

4.1.1 The χ^2-Measure

The χ^2-measure directly implements the idea to compare the joint distribution and the product of the marginal distributions by computing their squared difference. For two variables A and B it is defined as

$$\chi^2 = \sum_{i,j} N \frac{(P(a_i)P(b_j) - P(a_i, b_j))^2}{P(a_i)P(b_j)},$$

where N is the number of tuples in the database to learn from. An extension to m variables $A^{(1)}, \ldots, A^{(m)}$ can be obtained in two different ways

$$\chi^2_1 = \sum_{i_1,\ldots,i_m} N \frac{\left(\prod_{k=1}^{m} P(a^{(k)}_{i_k}) - P(a^{(1)}_{i_1}, \ldots, a^{(m)}_{i_m})\right)^2}{\prod_{k=1}^{m} P(a^{(k)}_{i_k})}$$

and

$$\chi^2_2 = \sum_{i_1,\ldots,i_m} N \frac{\left(P\left(a^{(1)}_{i_1},\ldots,a^{(m-1)}_{i_{m-1}}\right)P\left(a^{(m)}_{i_m}\right) - P\left(a^{(1)}_{i_1},\ldots,a^{(m)}_{i_m}\right)\right)^2}{P\left(a^{(1)}_{i_1},\ldots,a^{(m-1)}_{i_{m-1}}\right)P\left(a^{(m)}_{i_m}\right)}$$

The second version, in which all but one variable are combined into one pseudo-variable, is especially suited for directed edges, since the parent variables are the obvious ones to combine. If not stated otherwise, all measures described for two variables in the following can be extended in these two ways.

4.1.2 Entropy-based Measures

In [3] the (two-dimensional) edges of a tree-decomposition of a multi-dimensional distribution are selected with the help of *mutual information*. Under the name of *information gain* this measure was later used for the induction of decision trees [23, 24], which is closely related to learning inference networks (with directed edges): A hyperedge consisting of a variable and its parents can be seen as a decision tree with the restriction that all leaves have to lie on the same level and all decisions in the same level of the tree have to be made on the same attribute.

Mutual information implements the idea to compare the joint distribution and the product of the marginal distributions by computing their quotient. For two variables A and B it is defined as

$$\begin{aligned} I_{\text{mutual}} &= \sum_{i,j} P(a_i, b_j) \log_2 \frac{P(a_i, b_j)}{P(a_i)P(b_j)} \\ &= H_A + H_B - H_{AB} \quad = \quad I_{\text{gain}}, \end{aligned}$$

where H is the Shannon entropy [28]. It can be shown, that mutual information is always greater or equal to zero, and equal to zero, if and only if the joint distribution and the product of the marginal distributions coincide [18]. Hence it can be seen as measuring the difference of the two distributions. In the interpretation as information gain, it measures the information (in bits) gained about the value of one variable from the knowledge of the value of the other variable.

When using information gain for decision tree induction, it was discovered that information gain is biased towards many-valued attributes. To adjust for this bias the *information gain ratio* was introduced, which is defined as the information gain divided by the entropy of the split attribute [23, 24]:

$$I_{\text{gr}} = \frac{I_{\text{gain}}}{H_A} = \frac{I_{\text{gain}}}{-\sum_i P(a_i) \log_2 P(a_i)}.$$

Transfered to learning inference networks this means to divide the generalized information gain by the sum of the entropies of the parent variables. (Obviously this is only applicable when directed edges are used. Otherwise there would be no "split attribute" in contrast to the "class attribute.")

An alternative is the *symmetric information gain ratio* defined in [20], which is the information gain divided by the entropy of the joint distribution:

$$I_{\text{sgr}} = \frac{I_{\text{gain}}}{H_{AB}} = \frac{I_{\text{gain}}}{-\sum_{i,j} P(a_i, b_j) \log_2 P(a_i, b_j)}.$$

Because of its symmetry it is also applicable for undirected edges.

The measures discussed above are all based on Shannon entropy, which can be seen as a special case (for $\beta \to 1$) of *generalized entropy* [5]:

$$H^\beta(p_1, \ldots, p_r) = \sum_{i=1}^{r} p_i \frac{2^{\beta-1}}{2^{\beta-1}-1}(1 - p_i^{\beta-1})$$

Setting $\beta = 2$ yields the *quadratic entropy*

$$H^2(p_1, \ldots, p_r) = \sum_{i=1}^{r} 2p_i(1-p_i) = 2 - 2\sum_{i=1}^{r} p_i^2.$$

Using it in a similar way as Shannon entropy leads to the so-called Gini index

$$Gini = \frac{1}{2}(H_A^2 - H_{A|B}^2),$$

a well known measure for decision tree induction [2, 29]. Because of its asymmetry (in general it is $H_{A|B}^2 \neq H_{B|A}^2$) it can be used only with directed graphs. An extension to more than two variables should be achieved by combining all conditioning variable into one pseudo-variable.

4.1.3 MDL-based Measures

Information gain can also be seen as measuring the reduction in the description length of a dataset, if the values of a set of variables are encoded together (one symbol per tuple) instead of separately (one symbol per value). The *minimum description length principle* [25] in addition takes into account the information needed to transmit the coding scheme, thus adding a "penalty" for making the model more complex by enlarging a hyperedge.

Unfortunately limits of space prevent us from a detailed discussion of these measures, especially the extension of the two MDL-based measures suggested in [15] for decision tree induction, to learning inference networks.

4.1.4 Bayesian Measures

In [4] as an evaluation measure the g-function is used, which is defined as

$$g(A, \mathrm{par}_A) = c \cdot \prod_{j=1}^{n_{\mathrm{par}_A}} \frac{(n_A - 1)!}{(N_j + n_A - 1)!} \prod_{i=1}^{n_A} N_{ij}!,$$

where A is a variable and par_A the set of its parents. n_{par_A} is the number of distinct instantiations (value vectors) of the parent variables that occur in the database to learn from and n_A the number of values of variable A. N_{ij} is the number of cases (tuples) in the database in which variable A has the ith value *and* the parent variables are instantiated with the jth value vector, N_j the number of cases in which the parent variables are instantiated with the the jth value vector, that is $N_j = \sum_{i=1}^{n_A} N_{ij}$. c is a constant prior probability, which is usually set to 1, since with common search methods only the relation between the values of the evaluation measure for different sets of parent variables matters.

The g-function estimates (for a certain value of c) the probability of finding the joint distribution of the variable and its parents that is present in the database. That is, assuming that all network structures are equally likely, and that, given a certain structure, all conditional probability distributions compatible with the structure are equally likely, it uses Bayesian reasoning to compute the probability of the network structure given the database from the probability of the database given the network structure. This function seems to be applicable only in the probabilistic setting.

4.2 Minimum/Maximum Propagation

Analogous to the product/sum case the idea of some of the measures presented in this section is to compare the joint distribution with the minimum (instead of the product) of the marginal distributions.

4.2.1 Comparison-based Measures

For networks based on product/sum propagation the χ^2-measure and mutual information both compare directly the joint distribution and the product of the marginal distributions: the first by the difference, the latter by the quotient. Hence the idea suggests itself to apply the same scheme to networks based on minimum/maximum propagation, replacing the product by the minimum and the sum by the maximum.

We thus obtain for two variables A and B

$$d_{\chi^2} = \sum_{i,j} \frac{(\min(\pi_{\max}(a_i), \pi_{\max}(b_j)) - \pi(a_i, b_j))^2}{\min(\pi_{\max}(a_i), \pi_{\max}(b_j))}$$

as the analogon of the χ^2-measure and

$$d_{\mathrm{mi}} = -\sum_{i,j} \pi(a_i, b_j) \log_2 \frac{\pi(a_i, b_j)}{\min(\pi_{\max}(a_i), \pi_{\max}(b_j))}$$

as the analogon of mutual information. Since both measures are always greater or equal to zero, and zero, if and only if the two distributions coincide, they can be seen as measuring how much the two distributions differ.

4.2.2 Nonspecificity-based Measures

A possibilistic evaluation measure can also be derived from the U-uncertainty measure of *nonspecificity* of a possibility distribution [14], which is defined as

$$\mathrm{nsp}(\pi) = \int_0^{\sup(\pi)} \log_2 |[\pi]_\alpha| d\alpha$$

and can be justified as a generalization of Hartley information [10] to the possibilistic setting [13]. $\mathrm{nsp}(\pi)$ reflects the expected amount of information (measured in bits) that has to be added in order to identify the actual value within the set $[\pi]_\alpha$ of alternatives, assuming a uniform distribution on the set $[0, \sup(\pi)]$ of possibilistic confidence levels α [9].

The role nonspecificity plays in possibility theory is similar to that of Shannon entropy in probability theory. Thus the idea suggests itself to construct an evaluation measure from nonspecificity in the same way as information gain and (symmetric) information gain ratio are constructed from Shannon entropy.

By analogy to information gain we define *specificity gain* as

$$S_{\mathrm{gain}} = \mathrm{nsp}(\pi_{\max A}) + \mathrm{nsp}(\pi_{\max B}) - \mathrm{nsp}(\pi_{AB}),$$

or for more than two variables

$$S_{\mathrm{gain}} = \sum_{k=1}^{m} \mathrm{nsp}(\pi_{\max A^{(k)}}) - \mathrm{nsp}(\pi_{A^{(1)},\ldots,A^{(m)}}).$$

This measure is equivalent to the one defined in [9]. Then, just like information gain ratio and symmetric information gain ratio, *specificity gain ratio*

$$S_{\mathrm{gr}} = \frac{S_{\mathrm{gain}}}{\mathrm{nsp}(\pi_{\max A})}$$

and *symmetric specificity gain ratio*

$$S_{\mathrm{sgr}} = \frac{S_{\mathrm{gain}}}{\mathrm{nsp}(\pi_{AB})}$$

can be defined. Extensions to more than two variables are obtained in the same way as above.

The idea of specificity gain is illustrated in figure 5. The joint possibility distribution is seen as a set of relational cases, one for each α-level. Specificity gain aggregates the gain in Hartley information for these relational cases by computing the integral over all α-levels.

5 Experimental Results

For the two examples described in section 3 the correct decomposition is found by all measures corresponding to the used propagation method and optimum weight spanning tree construction. First results on larger datasets, which we obtained with a prototype implementation, indicate that all presented measures are well suited for learning inference networks. For product/sum propagation the g-function and the minimum description length measures, for minimum/maximum propagation d_{mi} seem to yield the best results. But more tests are necessary for definite conclusions.

Acknowledgments

We are grateful to J. Gebhardt for fruitful discussions, especially about the topic of the two propagation schemes for inference networks.

References

[1] S.K. Andersen, K.G. Olesen, F.V. Jensen, and F. Jensen. HUGIN — A shell for building Bayesian belief universes for expert systems. *Proc. 11th Int. J. Conf. on Artificial Intelligence*, 1080–1085, 1989

[2] L. Breiman, J.H. Friedman, R.A. Olshen, and C.J. Stone. *Classification and Regression Trees*, Wadsworth International Group, Belmont, CA, 1984

[3] C.K. Chow and C.N. Liu. Approximating Discrete Probability Distributions with Dependence Trees. *IEEE Trans. on Information Theory* 14(3):462–467, IEEE 1968

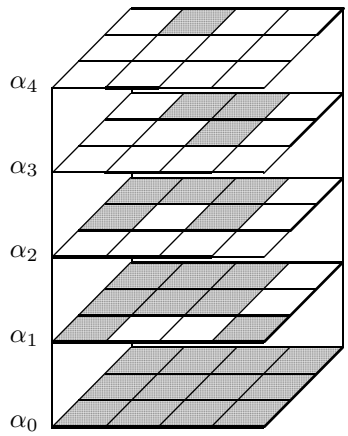
[4] G.F. Cooper and E. Herskovits. A Bayesian Method for the Induction of Probabilistic Networks from Data. *Machine Learning* 9:309–347, Kluwer 1992

[5] Z. Daróczy. Generalized Information Functions. *Information and Control* 16:36–51, 1970

[6] J. Gebhardt and R. Kruse. A Possibilistic Interpretation of Fuzzy Sets in the Context Model. *Proc. IEEE Int. Conf. on Fuzzy Systems*, 1089-1096, San Diego 1992.

[7] J. Gebhardt and R. Kruse. POSSINFER — A Software Tool for Possibilistic Inference. In: D. Dubois, H. Prade, and R. Yager, eds. *Fuzzy Set Methods in Information Engineering: A Guided Tour of Applications*, Wiley 1995

[8] J. Gebhardt and R. Kruse. Learning Possibilistic Networks from Data. *Proc. 5th Int. Workshop on Artificial Intelligence and Statistics*, 233–244, Fort Lauderdale, 1995



$$\log_2 1 + \log_2 1 - \log_2 1 = 0$$

$$\log_2 2 + \log_2 2 - \log_2 3 \approx 0.42$$

$$\log_2 3 + \log_2 2 - \log_2 5 \approx 0.26$$

$$\log_2 4 + \log_2 3 - \log_2 8 \approx 0.58$$

$$\log_2 4 + \log_2 3 - \log_2 12 = 0$$

Figure 5: Illustration of the idea of specificity gain. A two-dimensional possibility distribution is seen as a set of relational cases, one for each α-level. In each relational case, stating the allowed coordinates is compared to stating the allowed value pairs. Specificity gain aggregates the gain in Hartley information that can be achieved on each α-level by computing the integral over all α-levels.

[9] J. Gebhardt and R. Kruse. Tightest Hypertree Decompositions of Multivariate Possibility Distributions. *Proc. Int. Conf. on Information Processing and Management of Uncertainty in Knowledge-based Systems*, 1996

[10] R.V.L. Hartley. Transmission of Information. *The Bell Systems Technical Journal* 7:535–563, 1928

[11] D. Heckerman. *Probabilistic Similarity Networks*. MIT Press 1991

[12] D. Heckerman, D. Geiger, and D.M. Chickering. Learning Bayesian Networks: The Combination of Knowledge and Statistical Data. *Machine Learning* 20:197–243, Kluwer 1995

[13] M. Higashi and G.J. Klir. Measures of Uncertainty and Information based on Possibility Distributions. *Int. Journal of General Systems* 9:43–58, 1982

[14] G.J. Klir and M. Mariano. On the Uniqueness of a Possibility Measure of Uncertainty and Information. *Fuzzy Sets and Systems* 24:141–160, 1987

[15] I. Kononenko. On Biases in Estimating Multi-Valued Attributes. *Proc. 1st Int. Conf. on Knowledge Discovery and Data Mining*, 1034–1040, Montreal, 1995

[16] R. Kruse, E. Schwecke, and J. Heinsohn. *Uncertainty and Vagueness in Knowledge-based Systems: Numerical Methods*. Series: Artificial Intelligence, Springer, Berlin 1991

[17] R. Kruse, J. Gebhardt, and F. Klawonn. *Foundations of Fuzzy Systems*, John Wiley & Sons, Chichester, England 1994

[18] S. Kullback and R.A. Leibler. On Information and Sufficiency. *Ann. Math. Statistics* 22:79–86, 1951

[19] S.L. Lauritzen and D.J. Spiegelhalter. Local Computations with Probabilities on Graphical Structures and Their Application to Expert Systems. *Journal of the Royal Statistical Society, Series B*, 2(50):157–224, 1988

[20] R.L. de Mantaras. A Distance-based Attribute Selection Measure for Decision Tree Induction. *Machine Learning* 6:81–92, Kluwer 1991

[21] H.T. Nguyen. Using Random Sets. *Information Science* 34:265–274, 1984

[22] J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference (2nd edition)*. Morgan Kaufman, New York 1992

[23] J.R. Quinlan. Induction of Decision Trees. *Machine Learning* 1:81–106, 1986

[24] J.R. Quinlan. *C4.5: Programs for Machine Learning*, Morgan Kaufman, 1993

[25] J. Rissanen. A Universal Prior for Integers and Estimation by Minimum Description Length. *Annals of Statistics* 11:416–431, 1983

[26] A. Saffiotti and E. Umkehrer. PULCINELLA: A General Tool for Propagating Uncertainty in Valuation Networks. *Proc. 7th Conf. on Uncertainty in AI*, 323–331, San Mateo 1991

[27] G. Shafer and P.P. Shenoy. Local Computations in Hypertrees. Working Paper 201, School of Business, University of Kansas, Lawrence 1988

[28] C.E. Shannon. The Mathematical Theory of Communication. *The Bell Systems Technical Journal* 27:379–423, 1948

[29] L. Wehenkel. On Uncertainty Measures Used for Decision Tree Induction. *Proc. IPMU*, 1996