Set-Valued Probabilistic
Sentential Decision Diagrams

Alessandro Antonucci and Alessandro Facchini

Istituto Dalle Molle di Studi sull'Intelligenza Artificiale
Manno-Lugano (Switzerland)
{alessandro,alessandro.facchini}@idsia.ch

Abstract. *Probabilistic sentential decision diagrams* are a class of arithmetic circuits locally annotated by probability mass functions. This allows for a compact representation of joint mass functions in the presence of logical constraints. Here we propose a *set-valued* generalisation of the probabilistic quantification in these models, that allows to replace the sharp specification of the local probabilities with linear constraints over them. This induces a (convex) set of joint probability mass functions, all consistent with the assigned logical constraints. These models should be adopted for a cautious learning of the local parameters if only small amount of data are available. Algorithmic strategies to compute the lower and upper bounds of marginal and conditional queries with respect to these sets of joint mass functions are sketched. The task can be achieved in linear time with respect to the diagram size for marginal queries and, if the diagram is singly connected, the same can be done for conditional queries.

A simple example showing the need of coherently combining logic and probability is considered in [7]. Assume we are given a report of the results of 100 students in four subjects, namely logic (L), knowledge representation (K), probability (P) and AI (A). From those data, depicted in Table 1, we thus want to learn a probabilistic model such as a Bayesian network (BN) [8]. The white nodes in Figure 1 show a BN model that assumes conditional independence between K and P given A and L. To learn BN parameters we use a Laplace prior of strength one. This assigns probability $\Pr(X = x) = [n(X = x) + |X|^{-1}]/[n(\cdot) + 1]$ to the state x of variable X, where $|X|$ is the cardinality of X and n is a counting function for the data set (e.g., $\Pr(L = 0) = 70.5/101$). Out of sixteen possible combinations, Table 1 reports no observations for eight cases (grey rows). Yet, seven of these eight cases (dark grey) are not observed because of logical constraints: $L \vee P$ (it is compulsory for a student to either take logic or probability), $A \to P$ (probability is prerequisite for AI), and $K \to A \vee L$ (knowledge representation is a prerequisite to either AI or logic). The light grey row corresponds instead to a possible configuration for which we just do not have observations. Yet, the BN cannot distinguish between those two cases and non-zero probabilities can be therefore assigned to impossible events (e.g., $\Pr(A = 1, P = 0) \simeq 0.005$). The logical constraints can then be embedded in the BN by adding auxiliary nodes

(grey nodes in Figure 1). Each constraint induces a Boolean variable which is a child of all the variables involved in the logical constraint and whose conditional probability table has zero/one values modelling the fact that the variable is true if and only if the logical constraint is satisfied. The model is therefore queried given the fact that the auxiliary variables implementing the constraints are in their true state. Such a BN approach to the learning of probability mass functions that are subject to domain constraints might increase the *treewidth* of the original graphical model, thus potentially affecting the speed of the inferences (exact inference is exponential in this parameter).

L	K	P	A	#
0	0	0	0	0
0	0	0	1	0
0	0	1	0	6
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	10
1	0	0	0	5
1	0	0	1	0
1	0	1	0	1
1	0	1	1	0
1	1	0	0	13
1	1	0	1	0
1	1	1	0	8
1	1	1	1	3

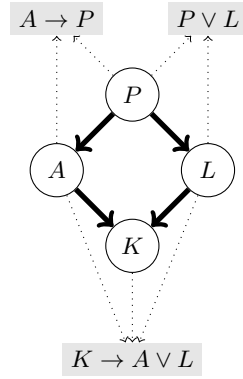**Table 1.** Results of 100 students in four subjects.

Fig. 1. A Bayesian network over four variables (white nodes). Grey nodes implement logical constraints.

Probabilistic sentential decision diagrams (PSDDs for short) [7] offer an elegant alternative approach to the learning of probabilistic models from data subject to logical constraints. A PSDD is a circuit representation of a joint probability mass function assigning zero probability to the impossible states of the logical constraints. More specifically, it is a parametrised directed acyclic graph. Each non-root node is either a logical AND gate with two inputs, or a logical OR gate with an arbitrary number of inputs and whose incoming wires are annotated with the probabilities of a conditional mass function (that can be learned from the data). These types of nodes alternate. Each root node is a univariate mass function, assigning probability one to X, when X is always true, to $\neg X$ when X is always false, or assigning θ to X (written as $(\theta : X)$) when X is true with probability θ (conditional to a certain *context* encoded by the circuit). E.g., the circuit in Figure 2 with sharp numbers instead of intervals associated to the wires defines a PSDD consistent with the logical constraints in the example.

Inference in PSDDs, intended here as the computation of the probability of marginal or conditional queries for single variables, takes time linear in the size of the graph. Dedicated algorithms have been developed to identify the smallest circuit modeling a given formula [2]. This makes inferences based on a PSDD typically faster than those based on a BN with auxiliary nodes implementing the same constraints [7]. Yet, PSDDs embed context-specific independence relations and the problem of trading off the search for a small circuit with the learning of proper dependence relations from the data is still relatively unexplored [9], while many techniques have been proposed for the structural learning of BNs [8].

From sharp to interval probabilities. Both BN and PSDD parameters are probabilities to be learned from the data set. These are sharp numerical values corresponding to the mean of a posterior distribution or the maximum of a likelihood. Yet, standard inference algorithms adopt only the sharp estimators, while propagating the whole distribution might increase the inferential complexity (e.g., [6]). This might be an issue especially if few data are available [13] or if some data are missing according to an asystematic incompleteness process [4]. In these cases, coping with sets instead of single probability mass functions might offer a more cautious, and hence reliable, approach to probabilistic modelling.[1]

In particular the imprecise Dirichlet model (IDM) [13] replaces the above considered estimate for $P(X = x)$ with the interval $[n(x), n(x) + |X|^{-1}]/(n(\cdot) + 1)$. Thus, for instance, $P(L = 0) \in [70, 71]/101$. IDM has been often applied to BN quantification from data (e.g., [1]). The approach transforms the BN into a *credal* network [3], and the local conditional probability mass functions, associated to each variable and each configuration of the parent variables are replaced by sets of probability mass functions consistent with such interval constraints. The same can be done with PSDDs by replacing each probability mass function annotating the incoming wires of a OR gate or a root node with a IDM-based set of probability mass function.[2] This might be valuable as the local parameters of a PSDD are conditional probabilities based on a *context*. Indeed, the higher is the number of variables in the context, the less data might be available to learn the probabilities, and therefore IDM appears as more reliable from a modelling perspective. As an example, Figure 2 depicts a PSDD with interval-valued weights obtained in this way. We call such a model *credal PSDD* (CPSDD). Exactly as a PSDD defines a joint probability mass function which assigns non-zero probability only to the joint states satisfying the logical constraints, a CPSDD defines a (credal) set of joint probability mass function assigning non-zero proba-

---

[1] Other extensions of the classical probability theory have been proposed. Among others, we mention the evidence theory [11], the possibility theory [5], and fuzzy sets theory [15]. The set-valued approach we consider here is based instead on the *imprecise probability* theory [12] and can be regarded as a generalization of all the previous ones [14], with a direct sensitivity analysis interpretation. To the best of our knowledge, none of the other formalisms has been applied to PSDDs so far.

[2] If the application of the IDM involves zero counts because of the logical constraints, the IDM interval associated to the impossible event is replaced by a sharp zero.

bility to the non-impossible configurations. The model semantics is as expected, each element of this set of joint mass functions corresponds to a PSDD whose (sharp) parameters have values consistent with the intervals in the CPSDD. Inferences in CPSDDs are thence intended as the computation of the lower and upper bounds of a query with respect to such a joint set of probability mass functions. Notably, it seems possible to apply the same strategy proposed by [10] for sum-product networks (SPNs) with set-valued specifications of the parameters to PSDDs. Yet, compared to SPNs, PSDDs allows to cope with logical constraints. For joint or marginal queries, the inference corresponds to a simple bottom-up propagation which can be easily achieved by interval arithmetic. E.g., the query $\Pr(A = 0, P = 0)$ corresponds to a product of IDM intervals and has value $[\frac{540}{3131}, \frac{19}{101}]$. For conditional queries the situation might be more involved. This is because local (linear) optimisation should be solved during the intermediate steps of the propagation. By a derivation analogous to that in [10], it is possible to argue that for CPSDDs with singly connected diagrams, i.e., such that each node has exactly one child, inference can be computed in time linear with respect to the size of the circuit.
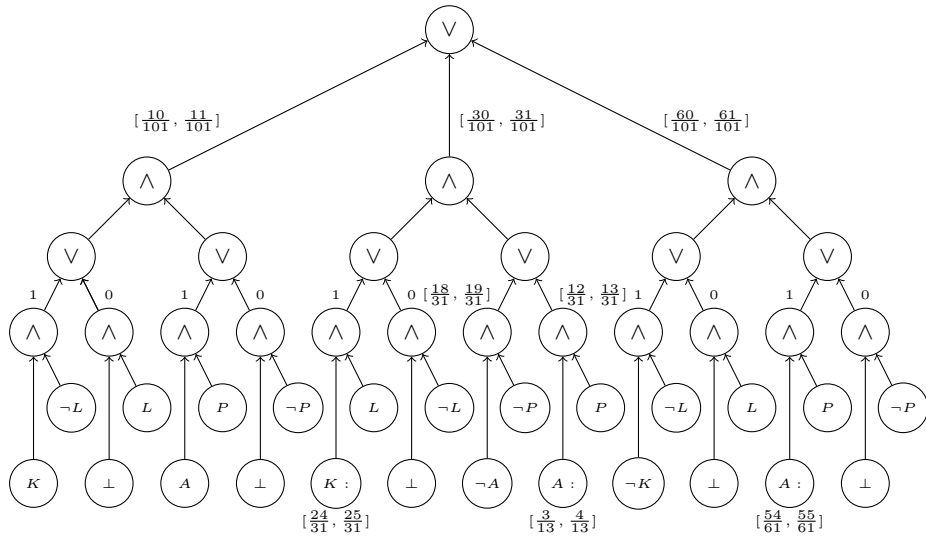

Fig. 2. A credal PSDD.

References

1. Alessandro Antonucci and Giorgio Corani. The multilabel naive credal classifier. *International Journal of Approximate Reasoning*, 83:320–336, 2016.
2. Arthur Choi and Adnan Darwiche. Dynamic minimization of sentential decision diagrams. In *AAAI*, 2013.
3. Fabio G. Cozman. Credal networks. *Artificial Intelligence*, 120:199–233, 2000.
4. Gert de Cooman and Marco Zaffalon. Updating beliefs with incomplete observations. *Artificial Intelligence*, 159(1–2):75–125, 2004.
5. Didier Dubois and Henri Prade. Possibility theory. In *Computational Complexity*, pages 2240–2252. Springer, 2012.
6. Lance Kaplan and Magdalena Ivanovska. Efficient belief propagation in second-order Bayesian networks for singly-connected graphs. *International Journal of Approximate Reasoning*, 93:132–152, 2018.
7. Doga Kisa, Guy Van den Broeck, Arthur Choi, and Adnan Darwiche. Probabilistic sentential decision diagrams. In *KR*, 2014.
8. Daphne Koller and Nir Friedman. *Probabilistic graphical models: principles and techniques.* MIT press, 2009.
9. Yitao Liang, Jessa Bekker, and Guy Van den Broeck. Learning the structure of probabilistic sentential decision diagrams. In *UAI*, 2017.
10. Denis Deratani Mauá, Diarmaid Conaty, Fabio Gagliardi Cozman, Katja Poppenhaeger, and Cassio Polpo de Campos. Robustifying sum-product networks. *International Journal of Approximate Reasoning*, 2018.
11. Glenn Shafer. *A Mathematical Theory of Evidence.* Princeton University Press, 1976.
12. Peter Walley. *Statistical Reasoning with Imprecise Probabilities.* Chapman and Hall, 1991.
13. Peter Walley. Inferences from multinomial data: learning about a bag of marbles. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 3–57, 1996.
14. Peter Walley. Towards a unified theory of imprecise probability. *International Journal of Approximate Reasoning*, 24:125–148, 2000.
15. Lotfi A. Zadeh. A computational approach to fuzzy quantifiers in natural languages. *Computers & Mathematics with applications*, 9(1):149–184, 1983.