Bayesian Factorisation as Adjoint

Bart Jacobs[1], Fabio Zanasi[2], and Octavio Zapata[2]

[1]Institute for Computing and Information Sciences, Radboud University Nijmegen,
`bart@cs.ru.nl`
[2]Department of Computer Science, University College London,
`f.zanasi@ucl.ac.uk, ocbzapata@gmail.com`

May 11, 2018

Bayesian probability theory is receiving increasing attention from the program semantics community. In the last few years, several frameworks [3, 7, 10, 5, 9, 2, 1] have been proposed modelling Bayesian reasoning in the style of programming language semantics. Algebraic methods, based on category theory, play a pivotal role in most of these approaches, offering a principled and expressive language to reason compositionally about probabilistic computation. This perspective has already proven to be useful in various ways: for instance, it clarifies the meaning of 'soft evidence' in Bayesian inference [6] and has recently suggested an alternative inference algorithm [4].

The present work goes in the same direction, offering a principled, categorical modelling for Bayesian probability. We focus on *learning*, one of the central tasks of Bayesian reasoning. Given a joint probability distribution ω, typically representing a collection of raw data, learning is the process of constructing a graph-like model, called a *Bayesian network*, in which nodes represent single events and edges represent statistical correlation of events. Such a correlation is expressed by conditional probabilities associated with each node of the network. When the graph structure is fixed, computing such conditional probabilities from the original ω is a deterministic process, and one says that ω *factorises* through the resulting Bayesian network G_ω.

Intuitively, factorisation is a sort of 'translation' from the language of probability distributions to the language of Bayesian networks. In practice, this is motivated by the fact that networks allow for more efficient representation of probabilistic information, which in joint distributions quickly become unmanageable via an exponential explosion. This translation procedure plays an important role in various algebraic approaches to Bayesian reasoning [2, 3, 6], but so far it has not been systematised as a categorical construction. The main result of this work is to make it categorical, by showing that factorisation can be seen as an adjoint functor Fact: Dst → BNet. Here the source category Dst has objects pairs (ω, G) of a probability distribution ω and a graph G (whose sets associated with the nodes together corresponds to the sample space of ω); the target category BNet has Bayesian networks as objects. In both categories, arrows are graph homomorphisms preserving probabilistic independencies. We retrieve a left adjoint Flat: BNet → Dst for the factorisation functor; we call it 'flattening', because it merges the conditional probabilities associated with a Bayesian network into a joint probability distribution.

Theorem 1. *There is an adjunction*

$$BNet \underset{Fact}{\overset{Flat}{\rightleftarrows}} Dst$$

Interestingly, the counit of this adjunction turns out to be given by one of the fundamental results relating Bayesian networks and probability distributions (see e.g. [8, Ch. 3]), namely that factorisation preserves the conditional independencies of the original distribution. The unit is actually an isomorphism: this allows us to give formal meaning, via the following statement, to the observation that Bayesian networks can be encoded as certain kinds of joint probabilities.

Corollary 1. *BNet is a coreflective subcategory of Dst.*

Further consequences of the theory remain to be explored. One is the investigation of coalgebras for the comonad Flat ∘ Fact on Dst, which can be computed as pairs (ω, G) where ω and the flattening of its factorisation have the same conditional independencies. It is interesting to notice that this bears analogies with the notion of 'perfect map' appearing in the context of traditional Bayesian probability, see e.g. [8, Ch. 3].

References

[1] Kenta Cho and Bart Jacobs. Disintegration and bayesian inversion, both abstractly and concretely. *arXiv preprint arXiv:1709.00322*, 2017.

[2] Florence Clerc, Vincent Danos, Fredrik Dahlqvist, and Ilias Garnier. Pointless learning. In *International Conference on Foundations of Software Science and Computation Structures*, pages 355–369. Springer, 2017.

[3] B. Fong. Causal theories: A categorical perspective on Bayesian networks. Master's thesis, Univ. of Oxford, 2012. see `arxiv.org/abs/1301.6201`.

[4] Bart Jacobs. A channel-based exact inference algorithm for bayesian networks. *arXiv preprint arXiv:1804.08032*, 2018.

[5] Bart Jacobs and Fabio Zanasi. A predicate/state transformer semantics for bayesian learning. *Electronic Notes in Theoretical Computer Science*, 325:185–200, 2016.

[6] Bart Jacobs and Fabio Zanasi. The logical essentials of bayesian reasoning. *arXiv preprint arXiv:1804.01193*, 2018.

[7] Nils Jansen, Benjamin Lucien Kaminski, Joost-Pieter Katoen, Federico Olmedo, Friedrich Gretz, and Annabelle McIver. Conditioning in probabilistic programming. *Electronic Notes in Theoretical Computer Science*, 319:199 – 216, 2015. The 31st Conference on the Mathematical Foundations of Programming Semantics (MFPS XXXI).

[8] Daphne Koller and Nir Friedman. *Probabilistic graphical models: principles and techniques*. MIT press, 2009.

[9] Radu Mardare, Prakash Panangaden, and Gordon D. Plotkin. Quantitative algebraic reasoning. In *Proceedings of the 31st Annual ACM/IEEE Symposium on Logic in Computer Science, LICS '16, New York, NY, USA, July 5-8, 2016*, pages 700–709, 2016.

[10] S. Staton, H. Yang, C. Heunen, O. Kammar, and F. Wood. Semantics for probabilistic programming: higher-order functions, continuous distributions, and soft constraints. In *Logic in Computer Science, LICS '16*, pages 525–534, 2016.