Probabilistic Programming Reading List

Steven Holtzen
s.holtzen@northeastern.edu

July 18, 2021

Abstract

This document contains the Fall 2021 reading list for CS7480: Topics in Probabilistic Programming. The goal of this document is to be a representative list of papers from which students can draw from for presentations. It is not exhaustive, but if anyone would like to add an article, please feel free to make a pull request at the GitHub repository https://github.com/SHoltzen/CS7480-Reading-Fall21. This document will likely be updated through the course of the semester, so check back regularly.

Contents

1 Semantics

1. *Semantics of probabilistic programs*, [Kozen, 1981]: One of the original semantics papers on probabilistic programs, designed for verifying and representing randomized algorithms.
2. *PMAF: an algebraic framework for static analysis of probabilistic programs*, [Wang and Hoffmann, 2018]: Gives an algebraic semantics for manipulating probabilistic programs; very recent.
3. [Borgström et al., 2013]: Measure transformer semantics
4. [Ścibior et al., 2017]: Describe a semantics of recursive probabilistic programs, discrete inference

5. [Vákár et al., 2019]: Domain theory for higher-order probabilistic programs
6. *Contextual equivalence for a probabilistic language with continuous random variables and recursion*, [Wand et al., 2018]

2 Inference & Systems

In general, inference methods can be divided into several broad categories:

1. *Approximation* methods, which perform approximate inference either via sampling or optimization.
2. *Exact* methods, which aim to give exact answers to probabilistic inference queries.

This section attempts to organize some existing probabilistic programming systems based on their inference methods. Some sections defy such organization, often by having more than one supported inference, and these are listed at the end.

2.1 Approximation Methods

2.1.1 Sampling

1. [Hur et al., 2015]: Utilizes program analysis to improve Markov-Chain Monte Carlo.
2. [Nori et al., 2014]: R2 sampling method
3. [Goodman et al., 2012]: Church
4. [Wingate et al., 2011]: Sampling with program transformation
5. [Carpenter et al., 2017]: Stan
6. *Accelerating Metropolis-Hastings with Lightweight Inference Compilation*, [Liang et al., 2021]

2.1.2 Variational Approximations

Good introductions found in Murphy [2012, Chapter 21] and Bishop [2006, Chapter 10].

1. *Automatic variational inference in Stan*, [Kucukelbir et al., 2015]
2. *Probabilistic Programming with Variational Inference:Under the Hood*, `https://willcrichton.net/notes/probabilistic-programming-under-the-hood/`
3. *Deep amortized inference for probabilistic programs*, [Ritchie et al., 2016]
4. *Learning Proposals for Probabilistic Programs with Inference Combinators*, [Stites et al., 2021]

2.2 Exact Inference

2.2.1 Inference via Compilation

1. [Sampson et al., 2014]: Verifies that probabilistic assertions hold by compiling the program to a graphical model.
2. [McCallum et al., 2009]: Factorie, a language for specifying factor graphs.
3. [Minka et al., 2014]: Infer.NET, compiles probabilistic programs to factor graphs.

4. [Pfeffer, 2009]: Figaro, compiles to factor graphs
5. [Pfeffer, 2001]: Ibal, an early PPL which uses variable elimination
6. [Fierens et al., 2013]: ProbLog, compiles probabilistic logic programs to weighted Boolean formulae.
7. [Claret et al., 2013]: Performs inference by compiling probabilistic programs to algebraic decision diagrams (ADDs).

2.2.2 Path-based Methods

1. [Sankaranarayanan et al., 2013]: Approximates the probability with analyzing a finite subset of paths
2. [Albarghouthi et al., 2017]: FairSquare, performs inference by approximating integrating under each path in the program.
3. [Belle et al., 2015]: Approximate weighted model integration, a generalization of SMT-solvers to perform integrals instead of just finding a satisfying assignment.
4. [Chistikov et al., 2015]: Performs inference using weighted model integration.

2.3 Systems

1. *Deep Probabilistic Programming* (the Edward system), [Tran et al., 2017]

3 Probabilistic Program Analysis

1. *Conditional Independence by Typing*, [Gorinova et al., 2020]
2. [Morgan et al., 1996]: Probabilistic predicate transformers
3. [Ferrer Fioriti and Hermanns, 2015]: Analysis of probabilistic termination.
4. *pGCL: Formal reasoning for random algorithms* [Morgan and McIver, 1999].

4 Program Transformations

In standard program analysis, a program transformation is a rewriting procedure which preserves the underlying semantics of the program; for example, the optimization phase of a compiler. In the context of probabilistic programs, the goal is to generalize well-known rewriting procedures to apply to programs with probabilistic semantics, in the hopes of easing analyses such as inference.

1. [Hur et al., 2014]: Generalizes program slicing to the setting of probabilistic programs with observations.
2. [McIver and Morgan, 2005]: Abstraction and refinement in probabilistic systems; extending the framework of program abstraction to verifying probabilistic properties.
3. [Wang et al., 2018]: PMAF, abstract interpretation for lower/upper bounds on Bayesian inference
4. [Narayanan et al., 2016]: Hakaru, compiles programs into posterior distributions

5 Probabilistic Model Checking

1. [Baier et al., 1997]: Early work on symbolic model checking for probabilistic systems.
2. [Hermanns et al., 2008]: Probabilistic CEGAR, generalizes CEGAR to probabilistic systems.
3. [Dehnert et al., 2017]: Storm model checker
4. [Kwiatkowska et al., 2002]: PRISM model checker

6 Applications

- [Foster et al., 2016]: Probabilistic network verification (ProbNetKat).
- [Gordon et al., 2014]: Using probabilistic programs to define probabilistic databases.
- [Schkufza et al., 2013]: Stochastic super-optimization; treats optimization as a search through a probability space over programs.

References

Aws Albarghouthi, Loris D'Antoni, Samuel Drews, and Aditya Nori. Quantifying program bias, 2017.

Christel Baier, Edmund M. Clarke, Vasiliki Hartonas-Garmhausen, Marta Kwiatkowska, and Mark Ryan. Symbolic model checking for probabilistic processes. pages 430–440, 1997. ISSN 16113349. doi: 10.1007/3-540-63165-8_199. URL `http://link.springer.com/10.1007/3-540-63165-8{_}199`.

Vaishak Belle, Andrea Passerini, and Guy Van den Broeck. Probabilistic inference in hybrid domains by weighted model integration. In *Proc. of IJCAI*, pages 2770–2776, 2015.

Christopher M Bishop. *Pattern Recognition and Machine Learning*. Springer Science+Business Media, 2006. ISBN 0-387-31073-8.

Johannes Borgström, Andrew D Gordon, Michael Greenberg, James Margetson, and Jurgen Van Gael. Measure Transformer Semantics for Bayesian Machine Learning. *Proc .of ESOP*, 6602:77–96, 2013. doi: 10.2168/LMCS-9(3:11)2013.

Bob Carpenter, Andrew Gelman, Matthew D Hoffman, Daniel Lee, Ben Goodrich, Michael Betancourt, Marcus Brubaker, Jiqiang Guo, Peter Li, and Allen Riddell. Stan: A probabilistic programming language. *Journal of statistical software*, 76(1), 2017.

Dmitry Chistikov, Rayna Dimitrova, and Rupak Majumdar. Approximate counting in smt and value estimation for probabilistic programs. In *Proc. of TACAS*, pages 320–334, New York, NY, USA, 2015. Springer-Verlag New York, Inc. ISBN 978-3-662-46680-3. doi: 10.1007/978-3-662-46681-0_26. URL `http://dx.doi.org/10.1007/978-3-662-46681-0_26`.

Guillaume Claret, Sriram K Rajamani, Aditya V Nori, Andrew D Gordon, and Johannes Borgström. Bayesian Inference Using Data Flow Analysis. *J. Foundations of Software Engineering*, pages 92–102, 2013. doi: 10.1145/2491411.2491423.

Christian Dehnert, Sebastian Junges, Joost-Pieter Katoen, and Matthias Volk. A storm is coming: A modern probabilistic model checker. In *CAV*, 2017.

Luis María Ferrer Fioriti and Holger Hermanns. Probabilistic termination: Soundness, completeness, and compositionality. *SIGPLAN Not.*, 50(1):489–501, January 2015. ISSN 0362-1340. doi: 10.1145/2775051.2677001. URL `http://doi.acm.org/10.1145/2775051.2677001`.

Daan Fierens, Guy Van den Broeck, Joris Renkens, Dimitar Shterionov, Bernd Gutmann, Ingo Thon, Gerda Janssens, and Luc De Raedt. Inference and learning in probabilistic logic programs using weighted boolean formulas. *J. Theory and Practice of Logic Programming*, 15(3):358 – 401, 2013.

Nate Foster, Dexter Kozen, Konstantinos Mamouras, Mark Reitblatt, and Alexandra Silva. Probabilistic netkat. In *Proceedings of the 25th European Symposium on Programming Languages and Systems - Volume 9632*, pages 282–309, New York, NY, USA, 2016. Springer-Verlag New York, Inc. ISBN 978-3-662-49497-4. doi: 10.1007/978-3-662-49498-1_12. URL `http://dx.doi.org/10.1007/978-3-662-49498-1_12`.

Noah Goodman, Vikash Mansinghka, Daniel M Roy, Keith Bonawitz, and Joshua B Tenenbaum. Church: a language for generative models. *arXiv preprint arXiv:1206.3255*, 2012.

Andrew D Gordon, Thore Graepel, Nicolas Rolland, Claudio Russo, Johannes Borgstrom, and John Guiver. Tabular: a schema-driven probabilistic programming language. *Proc. of POPL*, (1):321–334, 2014. ISSN 15232867. doi: 10.1145/2535838.2535850.

Maria I Gorinova, Andrew D Gordon, Charles Sutton, and Matthijs Vakar. Conditional independence by typing. *arXiv preprint arXiv:2010.11887*, 2020.

Holger Hermanns, Björn Wachter, and Lijun Zhang. *Probabilistic CEGAR*, pages 162–175. Springer Berlin Heidelberg, Berlin, Heidelberg, 2008. ISBN 978-3-540-70545-1. doi: 10.1007/978-3-540-70545-1_16.

Chung-Kil Hur, Aditya V. Nori, Sriram K. Rajamani, and Selva Samuel. Slicing probabilistic programs. *Proc. of PLDI*, pages 133–144, 2014. doi: 10.1145/2594291.2594303.

Chung-Kil Hur, Aditya Nori, and Sriram Rajamani. A provably correct sampler for probabilistic programs. In *Foundations of Software Technology and Theoretical Computer Science (FSTTCS)*. Leibniz International Proceedings in Informatics, December 2015. URL `https://www.microsoft.com/en-us/research/publication/a-provably-correct-sampler-for-probabilistic-programs/`.

Dexter Kozen. Semantics of probabilistic programs. *Journal of Computer and System Sciences*, 22(3):328 – 350, 1981. ISSN 0022-0000. doi: https://doi.org/10.1016/0022-0000(81)90036-2. URL `http://www.sciencedirect.com/science/article/pii/0022000081900362`.

Alp Kucukelbir, Rajesh Ranganath, Andrew Gelman, and David M Blei. Automatic variational inference in stan. *arXiv preprint arXiv:1506.03431*, 2015.

Marta Kwiatkowska, Gethin Norman, and David Parker. Prism: Probabilistic symbolic model checker. In *International Conference on Modelling Techniques and Tools for Computer Performance Evaluation*, pages 200–204. Springer, 2002.

Feynman Liang, Nimar Arora, Nazanin Tehrani, Yucen Li, Michael Tingley, and Erik Meijer. Accelerating metropolis-hastings with lightweight inference compilation. In *International Conference on Artificial Intelligence and Statistics*, pages 181–189. PMLR, 2021.

A McCallum, K Schultz, and S Singh. Factorie: Probabilistic programming via imperatively defined factor graphs. *Proc. of NIPS*, 22:1249–1257, 2009. ISSN 03643417.

Annabelle McIver and Carroll Morgan. Abstraction and refinement in probabilistic systems. *ACM SIGMETRICS Performance Evaluation Review*, 32:41–47, 2005. ISSN 01635999. doi: 10.1145/1059816.1059824.

T. Minka, J.M. Winn, J.P. Guiver, S. Webster, Y. Zaykov, B. Yangel, A. Spengler, and J. Bronskill. Infer.NET 2.6, 2014. Microsoft Research Cambridge. http://research.microsoft.com/infernet.

Carroll Morgan and Annabelle McIver. pGCL: Formal reasoning for random algorithms. 1999.

Carroll Morgan, Annabelle McIver, and Karen Seidel. Probabilistic predicate transformers. *J. Programming Languages & Systems*, 18(3):325–353, 1996. ISSN 01640925. doi: 10.1145/229542.229547.

Kevin P Murphy. *Machine learning: a probabilistic perspective*. MIT press, 2012.

Praveen Narayanan, Jacques Carette, Wren Romano, Chung-chieh Shan, and Robert Zinkov. Probabilistic inference by program transformation in hakaru (system description). In *International Symposium on Functional and Logic Programming*, pages 62–79. Springer, 2016.

Aditya Nori, Chung-Kil Hur, Sriram Rajamani, and Selva Samuel. R2: An efficient mcmc sampler for probabilistic programs. In *Twenty-Eighth AAAI Conference on Artificial Intelligence*, 2014.

Avi Pfeffer. Ibal: A probabilistic rational programming language. In *IJCAI*, pages 733–740. Citeseer, 2001.

Avi Pfeffer. Figaro: An object-oriented probabilistic programming language. *Charles River Analytics Technical Report*, 137:96, 2009.

Daniel Ritchie, Paul Horsfall, and Noah D Goodman. Deep amortized inference for probabilistic programs. *arXiv preprint arXiv:1610.05735*, 2016.

Adrian Sampson, Pavel Panchekha, Todd Mytkowicz, Kathryn S. McKinley, Dan Grossman, and Luis Ceze. Expressing and verifying probabilistic assertions. In *Proceedings of the 35th ACM SIGPLAN Conference on Programming Language Design and Implementation*, PLDI, pages 112–122, New York, NY, USA, 2014. ACM. ISBN 978-1-4503-2784-8.

Sriram Sankaranarayanan, Aleksandar Chakarov, and Sumit Gulwani. Static analysis for probabilistic programs: Inferring whole program properties from finitely many paths. *SIGPLAN Not.*, 48(6):447–458, June 2013. ISSN 0362-1340. doi: 10.1145/2499370.2462179.

Eric Schkufza, Rahul Sharma, and Alex Aiken. Stochastic superoptimization. *SIGPLAN Not.*, 48(4):305–316, March 2013. ISSN 0362-1340. doi: 10.1145/2499368.2451150. URL `http://doi.acm.org/10.1145/2499368.2451150`.

Adam Ścibior, Ohad Kammar, Matthijs Vákár, Sam Staton, Hongseok Yang, Yufei Cai, Klaus Ostermann, Sean K. Moss, Chris Heunen, and Zoubin Ghahramani. Denotational validation of higher-order bayesian inference. *Proc. ACM Program. Lang.*, 2(POPL):60:1–60:29, December 2017. ISSN 2475-1421. doi: 10.1145/3158148. URL `http://doi.acm.org/10.1145/3158148`.

Sam Stites, Heiko Zimmermann, Hao Wu, Eli Sennesh, and Jan-Willem van de Meent. Learning proposals for probabilistic programs with inference combinators. *arXiv preprint arXiv:2103.00668*, 2021.

Dustin Tran, Matthew D Hoffman, Rif A Saurous, Eugene Brevdo, Kevin Murphy, and David M Blei. Deep probabilistic programming. *arXiv preprint arXiv:1701.03757*, 2017.

Matthijs Vákár, Ohad Kammar, and Sam Staton. A domain theory for statistical probabilistic programming. *Proc. ACM Program. Lang.*, 3(POPL):36:1–36:29, January 2019. ISSN 2475-1421. doi: 10.1145/3290349. URL `http://doi.acm.org/10.1145/3290349`.

Mitchell Wand, Ryan Culpepper, Theophilos Giannakopoulos, and Andrew Cobb. Contextual equivalence for a probabilistic language with continuous random variables and recursion. *Proceedings of the ACM on Programming Languages*, 2(ICFP):1–30, 2018.

Di Wang and Jan Hoffmann. PMAF: An Algebraic Framework for Static Analysis of Probabilistic Programs. *PLDI*, 2018.

Di Wang, Jan Hoffmann, and Thomas Reps. Pmaf: an algebraic framework for static analysis of probabilistic programs. In *Proceedings of the 39th ACM SIGPLAN Conference on Programming Language Design and Implementation*, pages 513–528. ACM, 2018.

David Wingate, Andreas Stuhlmüller, and Noah Goodman. Lightweight implementations of probabilistic programming languages via transformational compilation. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, pages 770–778, 2011.