























- initial universe definition,
- a function from a universe representing the distribution at one time point to a universe representing the distribution at the next time point. [PfeffRutHowCon]

All in all, Figaro is one of the most viable probabilistic tools nowadays, because it provides an extensive flexibility defining data models and running inference algorithms on them. Figaro's compatibility with Java environment allows to integrate its functionality in a wide range of IT projects.

## CONCLUSION

Probabilistic programming is one of the most rapidly growing areas of IT research nowadays arousing interest in academic circles (including research groups in MIT and Oxford), acknowledged IT leaders as Microsoft, well-established industrial customers and IT community all over the world.

The increasing interest of the international IT community to this relatively new direction can be accounted by practical applicability of probabilistic programming concepts in the context of machine learning.

Probabilistic programming explores possibilities of mapping theoretical concepts of probability theory onto suitable practical programming techniques to reason under uncertainty.

Probabilistic programmes operate with variables holding the quantified knowledge about the constituent elements of the modelled situation. There exist two general types of dependencies representing relations of variables in probabilistic programmes: directed and undirected. Bayesian networks are, as a rule, used to express directed dependencies, whereas Markov networks represent undirected dependencies.

An active interest of the academic community to the probabilistic programming encouraged appearance of various tools designed to perform tasks of probabilistic inference. These tools include both frameworks of already existent general purpose programming languages and "purely" probabilistic programming systems (many of them not Turing complete). A large number of current probabilistic programming tools are implemented on the basis of functional programming paradigm. However, OOP paradigm is considered to be promising in the context of probabilistic programming, as it suggests natural mechanisms of modelling the reality in terms of objects and enables reuse of code.

With respect to inference algorithms, there can be differentiated two major groups, i.e. exact (e.g. Variable Elimination algorithm) and approximate inference algorithms (e.g. sampling family). Inference complexity makes it especially important to choose the right algorithm for each particular situation.

Although probabilistic programming has managed to arouse the interest of the international IT community and to achieve positive results in a number of research projects worldwide, there are still things to be done for probabilistic programming to prove itself as a generally

accepted standard. In particular, it's needed to work out a unified basis for different approaches within probabilistic programming and develop "best practices" of it. Second, probabilistic programming has to be explored and tested in large-scale industrial IT projects, outside purely academic environment.

## LITERATURE

- Cronin B. "What is probabilistic programming?" Retrieved 20.01.2017 from <http://radar.oreilly.com/2013/04/probabilistic-programming.html>
- CS181 Lectures 20 - 21 - "Bayesian Networks". Retrieved 20.01.2017 from <http://isites.harvard.edu>
- Goodman N. D. "The Principles and Practice of Probabilistic Programming". In Principles of Programming languages (POPL), 2013
- Goodman N. D., Tenenbaum J. B. (electronic). "Probabilistic Models of Cognition". Retrieved 20.01.2017 from <http://probmods.org>.
- Gordon A. D., Henzinger T. A., Nori A. V. "Probabilistic Programming". Proc. of the International Conference on Software Engineering (ICSE, FOSE track), 2014
- Hicks M. "What is probabilistic programming?" Retrieved 20.01.2017 from <http://www.pl-enthusiast.net/2014/09/08/probabilistic-programming>
- Karkera K. R. "Building Probabilistic Graphical Models with Python". Packt Publishing, 2014.
- Koller D., Friedman N. "Probabilistic Graphical Models. Principles and Techniques". MIT Press, 2009
- Pfeffer A., Ruttenberg B., Howard M., O'Connor A. "Figaro Tutorial". Charles River Analytics. Retrieved 20.01.2017 from [https://www.cra.com/sites/default/files/pdf/Figaro\\_Tutorial.pdf](https://www.cra.com/sites/default/files/pdf/Figaro_Tutorial.pdf)
- Pfeffer A. "Figaro: An object-oriented probabilistic programming language". Charles River Analytics Technical Report, 2009
- Pfeffer A. "Practical Probabilistic Programming". MEAP Edition, Manning Publications, 2015
- Pfeffer A. "What Probabilistic Programming is and How to Use it". Manning Publications. Free Content. 2014
- Poole D. "Probabilistic Programming Languages: Independent Choices and Deterministic Systems". In R. Dechter, H. Geffner, and J. Y. Halpern, editors, Heuristics, Probability and Causality: A Tribute to Judea Pearl. College Publications, 2010
- Russell S. J., Norvig P. "Artificial Intelligence. A Modern Approach". Third Edition. Prentice Hall, 2010
- Sucar L. E. "Probabilistic Graphical Models. Principles and Applications". Springer, 2015
- Wiki. "Probabilistic Programming". Retrieved 20.01.2017 from <http://probabilistic-programming.org/wiki/Home>

Wingate D., Stuhlmüller A., Goodman N. D.  
"Lightweight Implementations of Probabilistic  
Programming Languages Via Transformational  
Compilation". Proc. of the 14th Artificial  
Intelligence and Statistics, 2011

Wood F., van de Meent J. W., Mansinghka V. "A New  
Approach to Probabilistic Programming Inference".  
AISTATS, 2014

## **CONTACT**

Olga Ivanova, software developer,  
ollyenn@gmail.com