

# OpenMarkov, an Open-Source Tool for Probabilistic Graphical Models

Manuel Arias, Jorge Pérez-Martín, Manuel Luque and Francisco J. Díez

Universidad Nacional de Educación a Distancia (UNED), Madrid, Spain

{marias, jperezmartin, mluque, fjdiez}@dia.uned.es

## Abstract

OpenMarkov is a Java open-source tool for building and evaluating probabilistic graphical models, including Bayesian networks, influence diagrams, and some Markov models. With more than 100,000 lines of code, it offers some features for interactive learning, explanation of reasoning, and cost-effectiveness analysis that are not available in any other tool. OpenMarkov has been used in universities, research centers, and large companies in more than 30 countries on four continents. Several models, some of them for real-world medical applications, built with OpenMarkov, are publicly available on Internet.

## 1 Introduction

Probabilistic graphical models (PGMs) [Pearl, 1988; Koller and Friedman, 2009] have been used in artificial intelligence to build many real-world applications. Their main advantage is the possibility of representing a joint probability distribution of many variables as a product of potentials, each involving a very small number of variables. (A potential is a function that maps the configurations of a set of variables onto  $\mathbb{R}$ .) The graph of a PGM represents (some of) the conditional independences of the distribution and determines the structure of the potentials. This facilitates the construction of these models from experts' knowledge, from data, or from a combination of both.

Kevin Murphy maintained until 2014 a famous list of 69 software packages for PGMs, [www.cs.ubc.ca/~murphyk/Software/bnsoft.html](http://www.cs.ubc.ca/~murphyk/Software/bnsoft.html).

To our knowledge, OpenMarkov, whose development started in 2002, is the only open-source tool in that list that is still under active development. Some R packages, such as `gRain`, `gRbase`, `bnlearn`, and `bnclassify`, were developed later, but they can only evaluate Bayesian networks (BNs) and none has a graphical user interface for editing PGMs.

The purpose of this demo is to present to the artificial intelligence community the main features of Open-

Markov, discussing the facilities it offers and its current limitations.

## 2 OpenMarkov

### 2.1 Representation of PGMs

Currently, OpenMarkov can edit and evaluate four types of PGMs: BNs [Pearl, 1988], influence diagrams (IDs) [Howard and Matheson, 1984], Markov IDs [Díez *et al.*, 2017], decision analysis networks (DANs) [Díez *et al.*, 2018a]. DANs are especially interesting because they can represent problems involving constraints and partially ordered decisions, which IDs cannot.

Potentials can be represented using tables, canonical models (noisy OR, MAX, AND...), trees, algebraic decision diagrams (ADDs), univariate probability distributions, user-defined functions, etc. There are also some potentials for temporal models, such as exponential and Weibull distributions [Arias *et al.*, 2011]. The uncertainty about the potentials can be represented by several types of distributions (Beta, Gamma, Dirichlet...) and covariance matrices.

OpenMarkov's native format for encoding models, ProbModelXML [Arias *et al.*, 2011], has also been implemented in other software tools, such as `pgmpy`, `MADP` and `PILGRIM`.<sup>1</sup> Several examples of PGMs built with OpenMarkov, most of them for medical problems, are available at [www.probmodelxml.org/networks](http://www.probmodelxml.org/networks).

### 2.2 Inference

OpenMarkov is able to perform several types of inference. The default algorithm for all of them is a generalized version of variable elimination [Jensen and Nielsen, 2007; Luque and Díez, 2010]. It also implements other algorithms for BNs, such as clustering [Lauritzen and Spiegelhalter, 1988], logic sampling [Henrion, 1988], and likelihood weighting [Fung and Chang, 1990]. The basic operations with potentials (sum, multiplication, marginalization...) have been optimized by using "accumulated offsets" [Arias and Díez, 2007], which significantly speeds up inference.

<sup>1</sup>See [www.probmodelxml.org](http://www.probmodelxml.org), <https://github.com/pgmpy>, [www.fransoliehoek.net/index.php?fuseaction=software.madp](http://www.fransoliehoek.net/index.php?fuseaction=software.madp), <http://pilgrim.univ-nantes.fr>.

Multicriteria models can be evaluated by combining all the criteria into a single one (weighted sum) or, in the case of cost-effectiveness models, with new algorithms developed by our research group [Arias and Díez, 2015; Díez *et al.*, 2017], which are able to solve much larger problems than standard techniques. There are also specific algorithms for temporal reasoning.

In other tools for PGMs the optimal strategy for a decision model, such as an ID or a DAN, is implicit in the set of optimal policies (one table for each decision node), which contain many irrelevant columns. In addition to showing those policies, OpenMarkov is able to build a tree that compactly represents the optimal intervention; for example, the evaluation of Mediastinet, a large ID for lung cancer, returns 5 policy tables, one of them containing 15,552 columns, while the strategy tree only has 5 leaves—one for each relevant column in that table [Luque *et al.*, 2017].

### 2.3 Learning Bayesian Networks

In addition to creating BNs from causal knowledge, it is possible to learn them from data. OpenMarkov currently implements the two classic algorithms: hill climbing, based on search and score, with different metrics [Cooper and Herskovits, 1991], and PC, based on the detection of probabilistic independences [Spirtes *et al.*, 2000]. Even though other tools offer many more algorithms, OpenMarkov is the only one able to learn BNs interactively: the GUI proposes a list of edits (changes in the network, such as adding, removing, or inverting a link) and the user can accept any of them (not necessarily the one with the highest score) or manually do a different edit. Several facilities for saving time when learning BNs from the same database with different options are described in OpenMarkov’s tutorial. Interactive learning is useful not only for didactic purposes but also because even a little causal knowledge can lead to a significantly more accurate BN [Bermejo *et al.*, 2012].

### 2.4 Graphical User Interface (GUI)

OpenMarkov’s GUI can be used to built 13 types of PGMs: in addition to those already mentioned, it can edit factored Markov decision processes (MDPs) and partially observable MDPs (POMDPs), limited memory influence diagrams (LIMIDs), dynamic LIMIDs, and others—see [Arias *et al.*, 2011] for the references. Even though it cannot evaluate all these models, it can be used in combination with other software packages; for example, the MADP tool, implemented in C++, which does not have a GUI, can evaluate POMDPs built with OpenMarkov and stored in ProbModelXML [Oliehoek *et al.*, 2017].

OpenMarkov’s GUI stands out in the explanation of reasoning. One of the most interesting features—developed for OpenMarkov’s predecessor, Elvira [Elvira Consortium, 2002]—is the possibility of showing several bars at each node, representing its posterior probability (for chance nodes) or its expected value (for utility nodes) [Lacave and Díez, 2002]. It is also possible

to introduce two types of evidence in IDs and to impose sub-optimal policies for what-if reasoning [Lacave *et al.*, 2007]. These facilities have been very useful for teaching PGMs [Díez *et al.*, 2018b] and when collaborating with medical doctors in the construction of several BNs and IDs [Lacave *et al.*, 2006].

### 2.5 Software Engineering

Currently OpenMarkov contains more than 100,000 lines of code, licensed under GPLv3. More than 15 developers have contributed, including undergraduate, master, and doctoral students. Its architectural design allows using OpenMarkov either as a standalone application or as an application program interface (API). The code is structured in around 50 Maven projects, each containing a good number of JUnit tests. It is stored in a Git server of Bitbucket, which also includes a bugtracker. For continuous integration, we use Nexus and Jenkins.

### 2.6 Support for Users and Developers

The site [www.openmarkov.org](http://www.openmarkov.org) contains a compiled version of OpenMarkov (.jar file) and a tutorial. Developers interested in using OpenMarkov as an API can consult the wiki, the Javadoc pages, and the list of issues.<sup>2</sup> There are email addresses and Google groups for users and for developers.

## 3 Discussion

One of the strengths of OpenMarkov is that it offers for free several features that are not available in any other software package, either commercial or open-source, such as a rich variety of PGMs and potentials. It implements novel algorithms for cost-effectiveness analysis and an advanced GUI with several explanation facilities and interactive learning of BNs. It has been carefully designed so that Java programmers can add new features with relative ease. All these features make OpenMarkov a very useful tool for teaching PGMs [Díez *et al.*, 2018b].

Its main limitation is that the lack of human resources in our research group makes maintenance difficult: debugging, adding new features, and improving the documentation progress much more slowly than we wished. In spite of that, OpenMarkov has already being used for research and for building real-world applications at universities, government institutes, and large companies in more than 30 countries on four continents.

### Acknowledgments

This work has been supported by the Spanish Government under grants TIN2009-09158, PI13/02446, and TIN2016-77206-R, and co-financed by the European Regional Development Fund (ERDF). It has also received support from projects 262266 and 324401 (FP7-PEOPLE-2012-IAPP) of the European Union. J.P. received a predoctoral grant from the Ministry of Education (FPU14/02577).

<sup>2</sup>See [www.openmarkov.org/docs/tutorial](http://www.openmarkov.org/docs/tutorial), <http://wiki.openmarkov.org>, [www.openmarkov.org/javadoc](http://www.openmarkov.org/javadoc), and <http://issues.openmarkov.org>.

## References

- [Arias and Díez, 2007] M. Arias and F. J. Díez. Operating with potentials of discrete variables. *International Journal of Approximate Reasoning*, 46:166–187, 2007.
- [Arias and Díez, 2015] M. Arias and F. J. Díez. Cost-effectiveness analysis with influence diagrams. *Methods of Information in Medicine*, 54:353–358, 2015.
- [Arias et al., 2011] M. Arias, F. J. Díez, and M. P. Palacios. ProbModelXML. a format for encoding probabilistic graphical models. Technical Report CISIAD-11-02, UNED, Madrid, Spain, 2011.
- [Bermejo et al., 2012] I. Bermejo, J. Oliva, F. J. Díez, and M. Arias. Interactive learning of Bayesian networks with OpenMarkov. In A. Cano, M. Gómez, and T. D. Nielsen, editors, *Proceedings of the Sixth European Workshop on Probabilistic Graphical Models (PGM’12)*, pages 27–34, Granada, Spain, 2012.
- [Cooper and Herskovits, 1991] G. F. Cooper and E. Herskovits. A Bayesian method for constructing Bayesian belief networks from databases. In B. D’Ambrossio, P. Smets, and P. Bonissone, editors, *Proceedings of the Seventh Conference on Uncertainty in Artificial Intelligence (UAI’91)*, pages 86–94, San Mateo, CA, 1991. Morgan Kaufmann.
- [Díez et al., 2017] F. J. Díez, M. Yebra, I. Bermejo, M. A. Palacios-Alonso, M. Arias, M. Luque, and J. Pérez-Martín. Markov influence diagrams: A graphical tool for cost-effectiveness analysis. *Medical Decision Making*, 37:183–195, 2017.
- [Díez et al., 2018a] F. J. Díez, M. Luque, and I. Bermejo. Decision analysis networks. *International Journal of Approximate Reasoning*, 96:1–17, 2018.
- [Díez et al., 2018b] F. J. Díez, I. París, J. Pérez-Martín, and M. Arias. Teaching Bayesian networks with OpenMarkov. In *Proceedings of the Ninth European Workshop on Probabilistic Graphical Models (PGM’18)*, Prague, Czech Republic, 2018.
- [Elvira Consortium, 2002] Elvira Consortium. Elvira: An environment for creating and using probabilistic graphical models. In J. A. Gámez and A. Salmerón, editors, *Proceedings of the First European Workshop on Probabilistic Graphical Models (PGM’02)*, pages 1–11, Cuenca, Spain, 2002.
- [Fung and Chang, 1990] R. Fung and K. C. Chang. Weighing and integrating evidence for stochastic simulation in Bayesian networks. In P. Bonissone, M. Henrion, L. N. Kanal, and J. F. Lemmer, editors, *Uncertainty in Artificial Intelligence 6 (UAI’90)*, pages 209–219, Amsterdam, The Netherlands, 1990. Elsevier Science Publishers.
- [Henrion, 1988] M. Henrion. Propagation of uncertainty by logic sampling in Bayes’ networks. In R. D. Shachter, T. Levitt, L. N. Kanal, and J. F. Lemmer, editors, *Uncertainty in Artificial Intelligence 4 (UAI’88)*, pages 149–164, Amsterdam, The Netherlands, 1988. Elsevier Science Publishers.
- [Howard and Matheson, 1984] R. A. Howard and J. E. Matheson. Influence diagrams. In R. A. Howard and J. E. Matheson, editors, *Readings on the Principles and Applications of Decision Analysis*, pages 719–762. Strategic Decisions Group, Menlo Park, CA, 1984.
- [Jensen and Nielsen, 2007] F. V. Jensen and T. D. Nielsen. *Bayesian Networks and Decision Graphs*. Springer-Verlag, New York, second edition, 2007.
- [Koller and Friedman, 2009] D. Koller and N. Friedman. *Probabilistic Graphical Models: Principles and Techniques*. The MIT Press, Cambridge, MA, 2009.
- [Lacave and Díez, 2002] C. Lacave and F. J. Díez. Explanation for causal Bayesian networks in Elvira. In *Proceedings of the Workshop on Intelligent Data Analysis in Medicine and Pharmacology (IDAMAP-02)*, pages 43–48, Lyon, France, 2002.
- [Lacave et al., 2006] C. Lacave, A. Oniško, and F. J. Díez. Use of Elvira’s explanation facilities for debugging probabilistic expert systems. *Knowledge-Based Systems*, 19:730–738, 2006.
- [Lacave et al., 2007] C. Lacave, M. Luque, and F. J. Díez. Explanation of Bayesian networks and influence diagrams in Elvira. *IEEE Transactions on Systems, Man and Cybernetics—Part B: Cybernetics*, 37:952–965, 2007.
- [Lauritzen and Spiegelhalter, 1988] 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*, 50:157–224, 1988.
- [Luque and Díez, 2010] M. Luque and F. J. Díez. Variable elimination for influence diagrams with super-value nodes. *International Journal of Approximate Reasoning*, 51:615–631, 2010.
- [Luque et al., 2017] M. Luque, M. Arias, and F. J. Díez. Synthesis of strategies in influence diagrams. In *Proceedings of the Thirty-third Conference on Uncertainty in Artificial Intelligence (UAI’17)*, pages 1–9, Corvallis, OR, 2017. AUAI Press.
- [Oliehoek et al., 2017] F. A. Oliehoek, M. T. J. Spaan, B. Terwijn, P. Robbel, and J. V. Messias. The MADP Toolbox: An open source library for planning and learning in (multi-)agent systems. *Journal of Machine Learning Research*, 18(89):1–5, 2017.
- [Pearl, 1988] J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, San Mateo, CA, 1988.
- [Spirtes et al., 2000] P. Spirtes, C. Glymour, and R. Scheines. *Causation, Prediction and Search*. The MIT Press, Cambridge, Massachusetts, second edition, 2000.