PyMC's Big Adventure: Lessons Learned from the Development of Open-source Software for Bayesian Modeling

Christopher Fonnesbeck
Department of Biostatistics
Vanderbilt University Medical Center
Nashville, TN 37203
chris.fonnesbeck@vanderbilt.edu

The PyMC project is a team of open source developers devoted to the development of software for applied Bayesian statistics and probabilistic machine learning. Broadly, our objective is to produce Python implementations of state-of-the-art methods that can be used by a wide range of non-expert analysts, thereby democratizing probabilistic programming and putting powerful Bayesian methods in the hands of those who need them most: economists, astronomers, epidemiologists, ecologists, and more. Our current product, PyMC3, allows users to implement arbitrary probabilistic models using a high-level API that is analogous to specifying a model on a whiteboard.

The first version of PyMC was created over fifteen years ago, as a solo academic project. Its original purpose was to provide an open-source alternative to WinBUGS, a proprietary single-platform software package that dominated applied Bayesian analysis at the time. PyMC2 (Patil *et al.* 2010), released in 2009, represented a transition to a high-performance code base, using optimized Fortran extensions, and was the first truly collaborative release, based largely on the efforts of a team of three developers. The current major release, PyMC3 (Salvatier *et al.* 2010), is the product of a complete re-write of the code base, moving to Theano (Bergstra *et al.* 2010), a deep learning framework, as the computational backend. This allowed for the next generation of Bayesian model fitting methods to be implemented, including gradient-based (Hamiltonian) Monte Carlo and variational inference. Today, PyMC3 is supported by over a dozen core developers, and is used widely to support academic and industry research and decision-making.

Most of PyMC's history was characterized by substantial contributions by a few talented but ephemeral developers. The relative success and stability that the project enjoys today is the result of the hard work of a small but enthusiastic core of users and contributors, the emergence of tools for building software communities online, namely GitHub, StackOverflow, and Slack, and a small dose of good fortune. I will share some of the lessons learned since 2003, and provide a summary of best practices and potential pitfalls associated with growing and sustaining a community-driven scientific software project.

Project Links

PyMC2 `https://github.com/pymc-devs/pymc`

PyMC3 `https://github.com/pymc-devs/pymc3`

References

[1] Patil, A., Huard, D. & Fonnesbeck, C.J. (2010). PyMC: Bayesian Stochastic Modelling in Python. Journal of Statistical Software 35 (4): 1–81.

Preprint. Work in progress.

[2] Salvatier, J., Wiecki, T. & Fonnesbeck, C.J. (2016) Probabilistic Programming in Python Using PyMC3. PeerJ Computer Science 2 (April): e55.

[3] Bergstra, J., Breuleux, O., Bastien, F., Lamblin, P., Pascanu, R., Desjardins, G., Turian, J., Warde-Farley, D., and Bengio, Y. (2010). Theano: A CPU and GPU Math Compiler in Python. In Proc. 9th Python in Science Conf. Vol. 1.