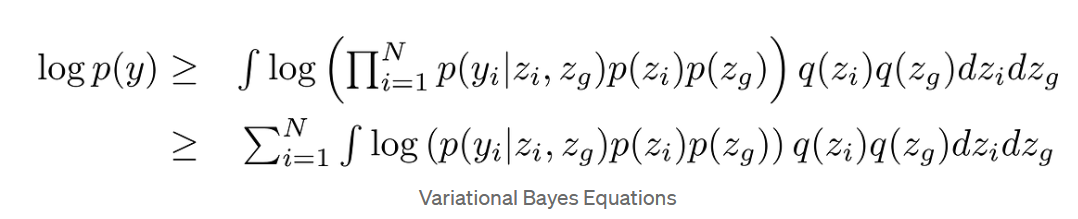
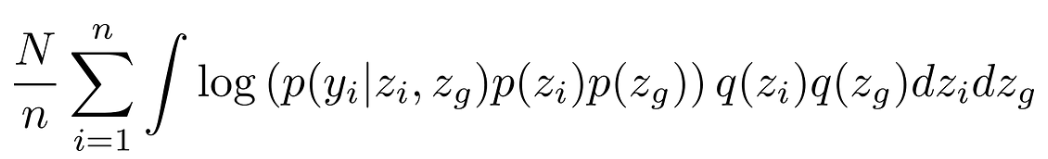
PyMC: backend by Theano. Bayesian models really struggle when it has to deal with a reasonably large amount of data (~10000+ data points). Variational inference is one way of doing approximate Bayesian inference. Both Stan and PyMC3 has this. But it is the extra step that PyMC3 has taken of expanding this to be able to use mini batches of data.





The PyMC package is an MCMC implementation written in Python and Fortran. It makes use of the classic Metropolis-Hastings MCMC sampler, and includes many built-in features, such as support for efficient sampling of common prior distributions.

PyMC includes HMC, SMC and black-box variational inference. It is based on PyTensor,3 a Python mathematics library that is a fork of Aesara, and continuation of the no longer developed Theano, which was the PyMC3 backend up to the current major release 4.0 and the renaming to PyMC. The computational graphs in PyMC are transpiled to C-code, Numba or JAX (a highperformance AD library for running Pyton/NumPy code on CPU, GPU and TPU), which allows for highly optimized code.

Stan: making use of a Hamiltonian MCMC using a No U-Turn Sampler.

Stan is implemented in C++, has a standalone command line interface, but also has mature interfaces for Python and R (RStan, PyStan) and lightweigt wrappers for Python and R (CmdStanPy , CmdStanR, BridgeStan2). There are also interfaces for most languages that are traditionally used for data analysis: Matlab (Matlabstan), Julia (Stan.jl), Stata (StataStan), Mathematica (MathematicaStan), Scala (ScalaStan) and http request-based interface (httpstan).While Stan implements black-box variational inference, Laplace approximation and standard optimization methods, the core Bayesian computation method is no-U-turn sampler, NUTS, a variant of HMC. Stan has a rich mathematics library with AD, and OpenCL-based GPU support with kernel fusion.

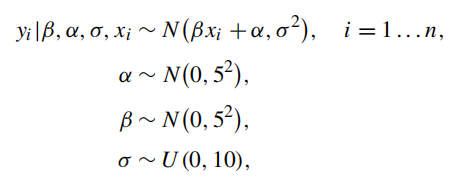
The performance between PyMC and Stan, compared with max( rss, vms, pss, uss), and running time, CPU time

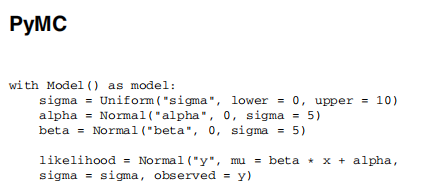
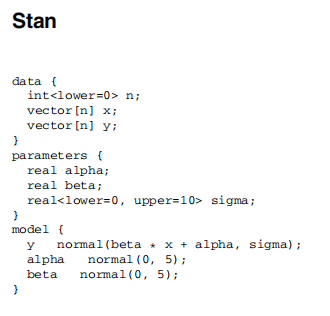
\*\*\*\* rss, vms, pss, uss是不同类型的内存

[pymc3-stan-comparison/docs/index.html at master · jhrcook/pymc3-stan-comparison · GitHub](https://github.com/jhrcook/pymc3-stan-comparison?tab=readme-ov-file" \l "running-the-pipeline)

[index (jhrcook.github.io)](https://jhrcook.github.io/pymc3-stan-comparison/)

Modelling example:





Edward: Backend by Tensorflow