Development Plan for OSS World Challenge 2012

Registration No.	2012- ※Registration No. need not be written.
Program title	Stan, a C++ library for probability and sampling

1. Program Overview

1) Development goals (background, goals etc.)

We propose to develop Stan, an extensible, open-source, cross-platform, software framework and compiler for Bayesian statistical modeling. Our emphasis is on efficient (i.e., fast and small) and scalable (i.e., able to deal with large data sets) implementations of full Bayesian inference through sampling.

The primary components for the Stan C++ framework are

- a fully templated density and distribution library including multivariate and matrix densities,
- an extensive library of templated special functions which are useful for probabilistic modeling,
- a fully templated vector, matrix, and linear algebra library,
- an extensible, thread-safe, object-oriented reverse-mode algorithmic differentiation package for com- puting gradients of arbitrary functions (including special functions, matrix operations, and densities),
- an implementation of a novel, adaptive Hamiltonian Monte Carlo sampler for simulating continuous densities over blocks of variables,
- an implementation of Gibbs samplers and random-walk Metropolis for blocks of bounded and unbounded discrete parameters,
- variable transforms (and reverse transforms) with log absolute Jacobian determinants to convert uni- variate and multivariate densities with constrained support to densities without constraints,
- an efficient,direct top-down sampler for simulating data from a given model for the puposes of model checking and simulation, and
- L-BFGS and conjugate gradient optimization routines for maximum likelihood and maimum a posteriori point estimation.

Development will include algorithmic design and coding in templated, object-oriented, thread-safe C++.

Linear algebra functionality will be based on the open-source Eigen library. Some special functions including cumulative densities, program parsing for the compiler, cross-platform thread management, (multithreaded) random-number generation, binding for lazy evaluation, and template metaprogramming facilities will be based on the open-source Boost C++ library.

2) System configuration

Stan is designed to work on any platform that supports C++, including Windows, Macintosh, and Unix/LInux. It will be integrated so as to be callable from Python, R, and MATLAB.

3) Menus

Not applicable.

4) Language used for development

C++.

5) Systems used

Stan is actively tested on Windows 7, Mac OS X, and a couple flavors of unix.

6) Plan for each stage of development

We will be releasing version 1.0 in July 2012 (planned). For the next version, we plan on having improvements to the underlying algorithms, more features, and more optimized code.

7) Number of personnel input and work assignment

Over the past year, we have had 7 people actively work on the project (with discussions with some more). Bob Carpenter is the lead and has designed and coded much of Stan. Daniel Lee has assisted in the design and is responsible for a lot of coding. Matthew Hoffman has provided much of the algorithm development, design, and coding. Jiqiang Guo has helped with the coding, especially with the package for use within R. Wei Wang has helped with implementing some models. Ben Goodrich has helped with the multivariate distributions and additional coding. Marcus Brubaker has also helped with the multivariate distributions and additional coding. Andrew Gelman has provided guidance and motivation for the project.

2. Long-term prospects of the program developed (No specific form is required.)

Stan is designed to facilitate practical, end-to-end applied Bayesian data analysis for scientists and engineers. Bayesian inference involves three steps, applied iteratively: (1) developing a probabilistic model of a scientific phenomenon including the data collection process, (2) fitting the model to the observed data, and (3) evaluating the fit of the model and its implications. Larger data sets combined with richer, more realistic probabilistic models require more efficient and scalable inference engines. Stan is aimed at automating Bayesian model fitting and inference, allowing a rapid exploration of potential modeling choices.

Stan will be compiled rather than interpreted, greatly speeding up core operations such as loops and function calls. Stan employs a newly-developed, automatically-tuned, adaptive Hamiltonian Monte Carlo algorithm, which uses gradient and directional information to accelerate convergence and subsequent exploration of the neighborhood of the fit. The result is one or more orders of magnitude speedup over existing general methods, such as Gibbs sampling or random-walk Metropolis, with greater speedups for larger data sets and models with higher dimensionality or more correlated parameters. Together, these improvements will make simple Bayesian analysis routine and enable more complex analyses than are currently possible.

Hamiltonian Monte Carlo has not been implemented in a general-purpose framework previously be- cause of several technical difficulties involving unconstrained variables, gradient calculations, and perfor- mance tuning parameters. Another obstacle is the complexity involved in compilation, threading, memory management, lazy evaluation through templates and binding, and numerical software.

The team proposed to develop Stan is interdisciplinary by design. The PI, Andrew Gelman, is a statistician specializing in Bayesian modeling and computation; he also collaborates on applications in many problems in the social and natural sciences. The co-PI, Bob Carpenter, is a computer scientist specializing in parsing and statistical natural language processing; he has spent the last ten years in industry developing production software libraries and applications. Both the PI and co-PI have written multiple textbooks, teach tutorials around the world, and maintain active and widely read blogs engaging their respective research communities. The senior researcher, Matthew D. Hoffman, is a computer scientist working on scalable Bayesian inference with applications in signal processing; he also has production coding experience.

Broader Impacts

Stan's intended audience includes any scientist or engineer with measured or simulated data. Bayesian statistical inference has already found a wide range of applications in a diverse range of scientific fields ranging from epidemiology to education, molecular physics to the social and behavioral sciences, and image processing to climate science.

To support the broadest range of participation among the current and next generation of scientists, Stan will be usable by general researchers on their existing desktop and notebook computers (in a manner similar to the Bayesian inference program WinBUGS), while at the same time being flexible enough to be programmed directly by more sophisticated users and distributed over a cluster.

For maximum ease of access, Stan will be licensed as free, open-source software. Stan will be integrated with scientists' existing workflows through the most popular data analysis and visualization tools, Python, R, and Matlab.