



# MCMC methods for Gaussian process models using fast approximations for the likelihood

Chunyi Wang, Radford M. Neal

(Submitted on 10 May 2013)

Gaussian Process (GP) models are a powerful and flexible tool for non-parametric regression and classification. Computation for GP models is intensive, since computing the posterior density,  $\pi$ , for covariance function parameters requires computation of the covariance matrix,  $C$ , a  $p \times p$  operation, where  $p$  is the number of covariates and  $n$  is the number of training cases, and then inversion of  $C$ , an  $n^3$  operation. We introduce MCMC methods based on the "temporary mapping and caching" framework, using a fast approximation,  $\tilde{\pi}$ , as the distribution needed to construct the temporary space. We propose two implementations under this scheme: "mapping to a discretizing chain", and "mapping with tempered transitions", both of which are exactly correct MCMC methods for sampling  $\pi$ , even though their transitions are constructed using an approximation. These methods are equivalent when their tuning parameters are set at the simplest values, but differ in general. We compare how well these methods work when using several approximations, finding on synthetic datasets that a  $\tilde{\pi}$  based on the "Subset of Data" (SOD) method is almost always more efficient than standard MCMC using only  $\pi$ . On some datasets, a more sophisticated  $\tilde{\pi}$  based on the "Nyström-Cholesky" method works better than SOD.

Subjects: **Computation (stat.CO)**

Cite as: [arXiv:1305.2235](#) [stat.CO]

(or [arXiv:1305.2235v1](#) [stat.CO] for this version)

## Submission history

From: Radford M. Neal [[view email](#)]

[v1] Fri, 10 May 2013 00:59:32 GMT (31kb,D)

*[Which authors of this paper are endorsers?](#)*

## Download:

- [PDF](#)
- [Other formats](#)

Current browse context:

stat.CO

[< prev](#) | [next >](#)

[new](#) | [recent](#) | [1305](#)

Change to browse by:

[stat](#)

## References & Citations

- [NASA ADS](#)

## Bookmark (what is this?)

