Scalable Online Machine Learning: Adaptive Mass Matrices in SMC Proposals

Andrew Millard, Supervised by Professor Simon Maskell (UoL) and Dr Simon Goodchild (STFC Hartree Centre)
Sponsored by GCHQ

Project Background and Research Aims

Background:
Currently, one of the gold standard sampling methods used by programming languages like STAN is the NUTS Sampling Method [1]. This is due to several reasons; because of the dual averaging equations [2]the step size is adapted based on the shape of the covariance and also the No-U-Turn mechanism draws a good approximation of the path length so that the sampler does not retrace its own path.

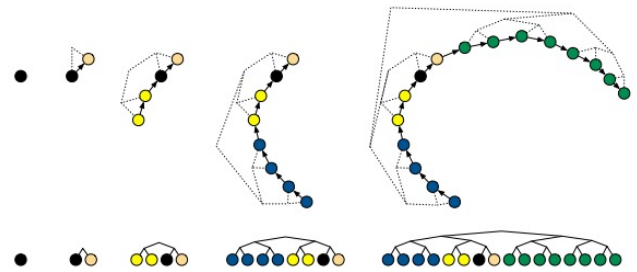


Figure 1: Binary Tree and Path Length Diagram [1]

However, there are some limitations. After the burn in, period the mass matrix is fixed instead of continually adapting due to complications of not satisfying detailed balance that is required by a MCMC algorithm.

Aim:
To get around this, NUTS has instead been implemented as a forward proposal in an SMC Sampler [3] but this still has a fixed mass matrix. The aims of my project are as follows:

- Developing a forward proposal that uses an adaptive mass matrix in an SMC Sampler
- Use the mathematics of the Adam Optimizer [4] to not only have a mass in dimension, but changing mass in each dimension
- Test on real world high dimensional datasets and see whether it adapts well to static datasets with rare events to detect and/or datasets which experience concept drift
- Make sure the proposal is built so that it can be distributed and therefore can scale well to a never ending data stream
- Develop a fast method that can compress information from a never ending data stream in a way that Bayesian inference is representative of the uncompressed dataset

Current Results

SMC Sampler Base
I have started to build out an SMC Sampler library written within the PyTorch framework. So far, I have several different proposals (including NUTS) implemented and also uses an adaptive mass matrix.

Linear and Logistic Regression with SMC
I am using SMC samplers for linear and logistic regression problems and comparing the results to SKLearns built-in solvers. SKLearns make_regression module allows us to output the linear regression coefficients that have been used to generate the data so we can easily test how well our sampler converges to these parameters (seen in Figure 2). I have also plotted the MSE of our sampler in Figure 3.

The Logistic Regression tests are still at the early stages but so far we have seen that in low dimensions, the classification accuracy is the same as SKLearns LogisticRegression model. The added benefit of the SMC Sampler is it gives a variance estimate as well as a mean estimate for each parameter which the SKLearn methods do not however, the accuracy of this is hard to measure in these tests as we do not know the true value of the variances in each dimension.

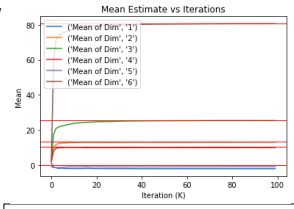


Figure 2: Convergence of Linear Regression coefficients as estimated by SMC Sampler (NUTS proposal).

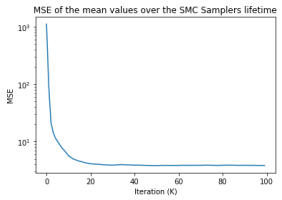


Figure 3: MSE of the SMC Sampler Linear Regression coefficient estimation over the lifetime of the sampler.

Future Work

Distributing the Sampler
A key part of my work will be being able to use the sampler in as efficient a way as possible. A key to this is being able to distribute the calculations of the sampler so that it can work on a HPC such as Barkla or Scarfell Pike.

Use in Bayesian Neural Networks
There has been some work on using SMC methods as an alternative to backpropagation [5] with some good effect. Current Bayesian approaches to NN's are either very slow (MCMC) or give bad variance approximations (Variational Inference). With Parallelised SMC we hope to be able to get the accuracy of MCMC approaches but with the speed of VI.

Anomaly Detection and Concept Drift
As our samplers give variance estimates on the weights of NN's, by setting a variance threshold, we may be able to detect anomalies in datasets by not classifying data which parameter values fall outside of this threshold. Increasing anomaly detections may indicate towards concept drift in the data so the combination of anomaly detection and rate of change of anomalies could be used to identify concept drift for time series data streams.

Ongoing Work

Mass Matrices
When samplers are not using an adaptive mass matrix, the mass matrix can be thought of as being fixed to the identity. Currently the mass matrix is commonly implemented via calculating the inverse of the covariance.

$$M = \left(\frac{\Sigma_T \cdot n + \Sigma_A \cdot N}{n + N}\right)^{-1}$$

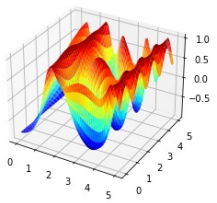
The momentum is picked from the diagonal elements of the mass matrix.

$$p_i \sim \mathcal{N}(0, M_{ii}), i = 1, ..., d$$

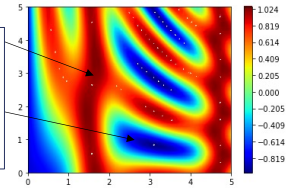
However, this method assumes that your current samples are already giving a good picture of what the covariance of the distribution looks like. So instead, we have come up with a method of calculating the mass matrix by using the Fisher Information Matrix.

$$\mathcal{M} = (\nabla log(\pi(x))\nabla log(\pi(x))^T)\tilde{w}_{1:t}$$

This method uses the gradient information in order to estimate the inverse of the covariance, which is also the mass matrix. We can use a combination of the gradient information from each sample to estimate a global mass matrix, or we can use the gradient information to give each sample an individual mass matrix which will determine the momentum for the next trajectory within the NUTS proposal. This method means that the current position and gradient of the sample dictates its momentum, as opposed to the position and gradient of other samples. This method is similar to how the Adam Optimizer [6] is constructed to finding the optima in complex probability landscapes. To avoid samples near minima getting stuck, we would use a combination of global and local mass matrices to determine the momentum of each particle.



We can imagine that a sample in the red region would have a higher mass than in the blue as it is further away from a minima.



Adaptive Numerical Integrator
Dual averaging is currently used to adapt the step size in NUTS. However, this has issues in high dimensional problems as the step size is changed by using an ideal acceptance ratio. In high dimensions, to achieve this acceptance ratio, the step size is often reduced to such a small number that the sampler can get stuck and not fully explore the covariance. Therefore, we are working on an alternative method by using a higher order numerical integrator.

References

[1] – Hoffman, M. and Gelman, A., 2022. *The No-U-Turn Sampler: Adaptively Setting Path Lengths in Hamiltonian Monte Carlo*. [online] arXiv.org. Available at: https://arxiv.org/abs/1111.4246
[2] – Nesterov, Y. (2005). Primal-dual Subgradient Methods for Convex Problems. *SSRN Electronic Journal*. doi:10.2139/ssrn.912637.
[3] - Devlin, L., Horridge, P., Green, P.L. and Maskell, S. (2021). The No-U-Turn Sampler as a Proposal Distribution in a Sequential Monte Carlo Sampler with a Near-Optimal L-Kernel. *arXiv:2108.02498 [stat]*. [online] Available at: https://arxiv.org/abs/2108.02498 .
[4] – Kingma, D.P. and Ba, J. (2014). *Adam: A Method for Stochastic Optimization*. [online] arXiv.org. Available at: https://arxiv.org/abs/1412.6980.
[5] – De Freitas, J., Niranjan, M., Gee, A. and Doucet, A. (n.d.). *LETTER Communicated by Sequential Monte Carlo Methods to Train Neural Network Models*. [online] Available at: http://www.gatsby.ucl.ac.uk/~byron/nlds/freitas2000b.pdf.
[6] - Staib, M., Reddi, S.J., Kale, S., Kumar, S. and Sra, S. (2020). Escaping Saddle Points with Adaptive Gradient Methods. *arXiv:1901.09149 [cs, math, stat]*. [online] Available at: https://arxiv.org/abs/1901.09149 .

UKRI Engineering and Physical Sciences Research Council
UNIVERSITY OF LIVERPOOL
GCHQ