Gaussian Process Regression and Classification: Elliptical Slice Sampling

Ricardo Marques Salgado

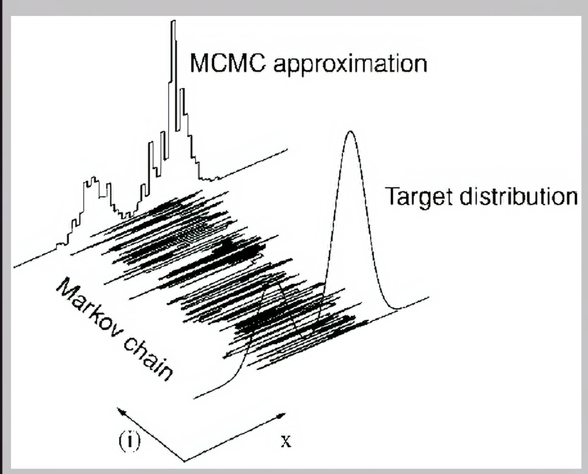
Supervisor: Dr Alta de Waal

Abstract

The use of Gaussian processes to perform classification and regression is considered, however Bayesian inference can become difficult to perform, as Bayes theorem requires a closed form solution of an integral which is often intractable. This issue is addressed through the use of the probabilistic approach in sampling, particularly Markov chain Monte Carlo inference. Within Markov chain Monte Carlo inference, an explanation as to why this is a suitable approach, such that the difficulties arising from Bayes theorem and it's implementation are avoided. In addition to this, a comparison of the degree to which several Markov chain Monte Carlo techniques address this obstacle is provided.

Why Markov Chain Monte Carlo Inference

The process of MCMC is illustrated below and there are several reasons as to why it works [3]:

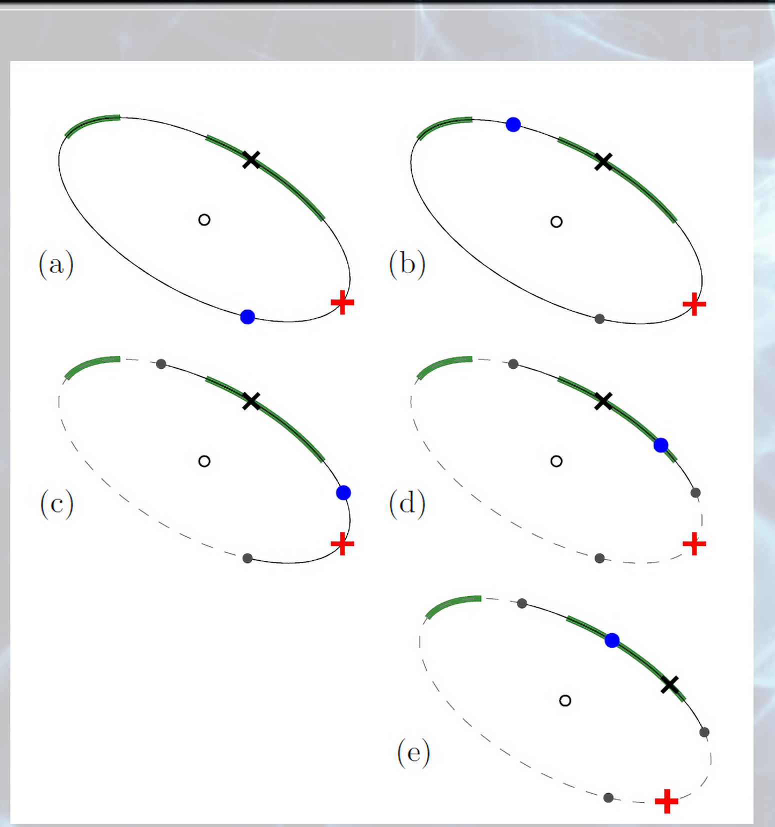


1. The Markov chain will possess a unique and stationary distribution.
2. The acceptance-rejection criteria for samples guarantees that not every sample is selected, but only those that ensure the target distribution is appropriately approximated.
3. The Markov property ensures that after a certain amount of iterations, the Markov chain will have "forgotten" the initial values, resulting in the Markov chain converging to a stationary distribution.
4. The posterior distribution of interest does not need to be known, only a distribution that is proportional to it needs to be known.

Elliptical Slice Sampling

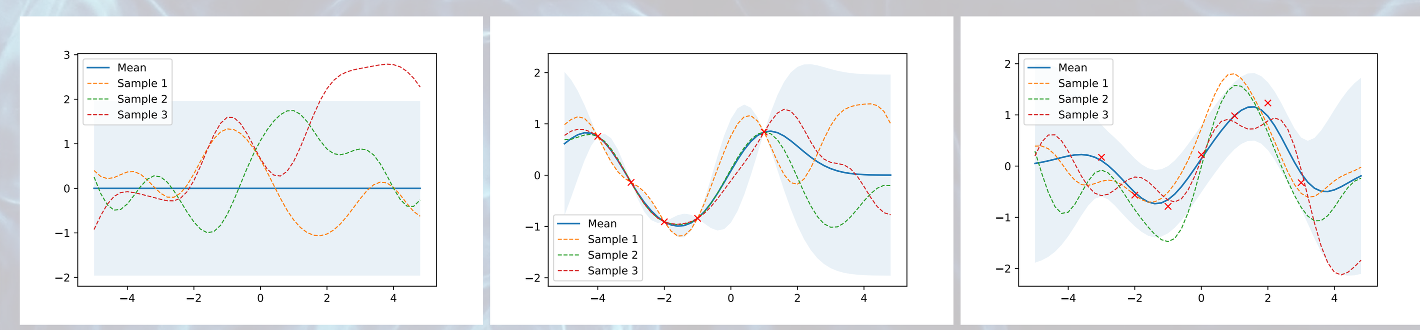
ESS can be described using the following illustration [1], that is:

a) A random variate, +, is drawn. This defines an ellipse centred at the origin. A likelihood threshold (slice), -, is then determined. An initial proposal, •, is drawn, which is not on the 'slice'.
b) Another proposal is drawn, and is not on the slice, so the bracket is shrunk to the two proposed values while still including the input **X**.
c) The above process is repeated.
d) A proposal is accepted, hence the bracket is not shrunk. This process is repeated until the required samples are obtained.

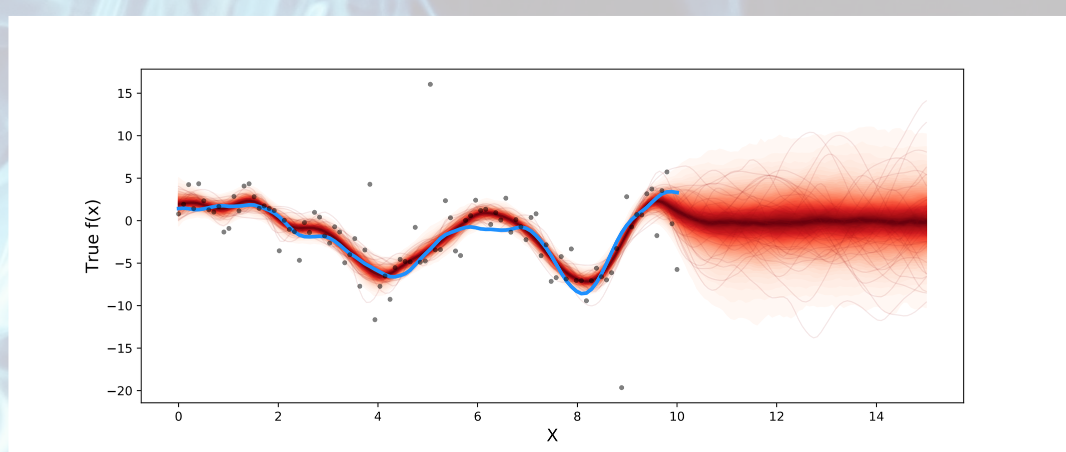


Gaussian Process Regression

GPs are a non-parametric modelling technique for regression and classification. A GP is a collection of random variables $\{X_t : t \in S\}$, such that any finite subset, $\{X_{t_0}, X_{t_1}, ..., X_{t_N}\}$, has a joint Gaussian distribution.



The use of GPs is illustrated in the three figures above. The left shows 3 samples from a GP prior, while the middle shows the effect of conditioning the samples on observations, and how the functions are "pulled" through these noise-free data points. The figure on the right illustrates the effect noise within the data has.



The use of GPs can then be extended to make use of sampling, and in particular MCMC algorithms to make predictions regarding the functions behaviour for unobserved x values.

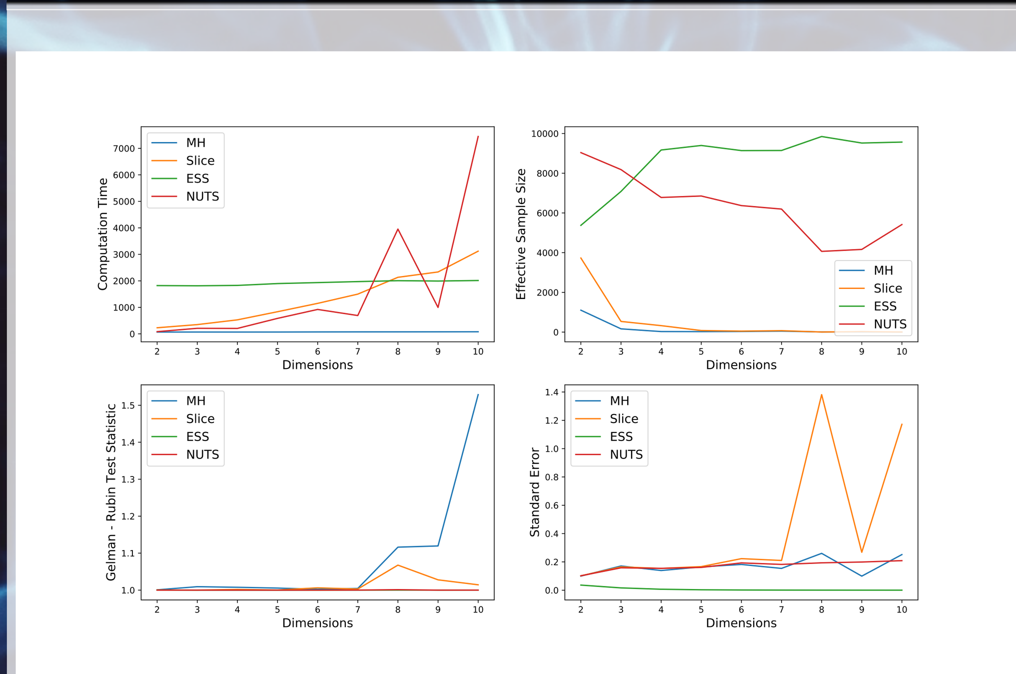
The Algorithm

Input: current state **f**, a routine to sample from $N(0, \Sigma)$, log-likelihood function *logL*.

Output: a new state **f**$'$. When **f** is drawn from $p^*(\boldsymbol{f}) \propto N(\boldsymbol{f}; 0, \Sigma)L(\boldsymbol{f})$, the marginal distribution of **f**$'$ is also p^*.

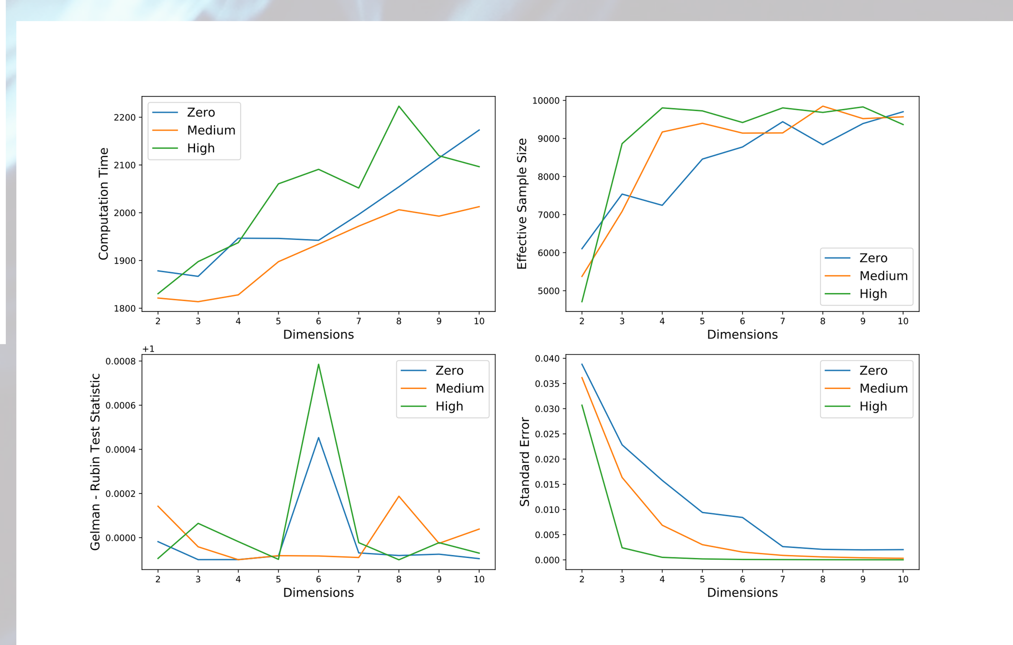
1 Choose ellipse: $\nu \sim N(0, \Sigma)$
2 Log-likelihood threshold:
$u \sim Uniform[0, 1]$ **and** $log(y) \leftarrow logL(\boldsymbol{f}) + log(u)$
3 Draw an initial proposal, also defining a bracket:
$\theta \sim Uniform[0, 2\pi]$ **and** $[\theta_{min}, \theta_{max}] \leftarrow [\theta - 2\pi, \theta]$
4 $\mathbf{f}' \leftarrow \mathbf{f}\cos(\theta) + \nu sin(\theta)$
5 **if** $logL(\mathbf{f}') > log(y)$ **then**: **return f**$'$
7 **else:**
Shrink the bracket and try a new point:
8 **if** $\theta < 0$ **then**: $\theta_{min} \leftarrow \theta$ **else**: $\theta_{max} \leftarrow \theta$
9 $\theta \sim Uniform[\theta_{min}, \theta_{max}]$, **Go to 4.**

Results



A comparison of the various performance metrics considered across the different MCMC algorithms implemented.

A comparison of the varying correlation structures effect on the performance of elliptical slice sampling.



Conclusion

Elliptical slice sampling far outperformed the other algorithms considered within this research with respect to the 4 measures considered. This improved performance came at the cost of an additional specification required by PyMC3 [2]. However, this requirement limits elliptical slice sampling's use as 'black-box' approach, when compared to methods such as MH and NUTS. Particularly when MH was significantly quicker and NUTS achieved the highest effective sample size of any algorithm in lower dimensions. It was also seen that the varying covariance and subsequent correlation structure between variables had little to no impact on any of the algorithms performance's.

References
1. Iain Murray, Ryan Prescott Adams, and David J. C. MacKay. Elliptical slice sampling. AISTATS13, 9:541–548, 2010.
2. John Salvatier, Thomas V. Wiecki, and Christopher Fonnesbeck. Probabilistic programming in Python using PyMC3.PeerJ Computer Science, 2:e55, 2016.
3. Kevin P. Murphy. Machine Learning: A Probabilistic Perspective. The MIT Press, 2012

UNIVERSITEIT VAN PRETORIA
UNIVERSITY OF PRETORIA
YUNIBESITHI YA PRETORIA