

UNIVERSITY OF COLOGNE



PRACTICAL COURSE M
COMPUTATIONAL PHYSICS

Statistical Inference

Sangeet Srinivasan

Tutor: Kai Meinerz

September 28, 2023

Contents

Contents	1
1 Learning Thermal Distributions	2
1.1 Thermalising Monte-Carlo Sweeps	2
1.2 Boltzmann Learning	3
1.2.1 Results	6
1.3 Glauber Dynamics	7
1.3.1 Results	8
2 Salamander Brain	9
2.1 Equilibrium Model	9
2.1.1 Results	10
2.2 Non-Equilibrium Model	10
2.2.1 Results	10
Bibliography	12

Chapter 1

Learning Thermal Distributions

We generate random target fields h as a vector of normally distributed data with 0 mean and standard deviation 1. Random target couplings J are generated as a symmetric matrix of normally distributed data with 0 mean and standard deviation $1/N$ where N is the system size. Random states are generated as an N -dimensional vector of ± 1 values. We use the Metropolis-Hastings algorithm (MHA) that executes Monte-Carlo (MC) sweeps where individual spins are flipped based on the energy differences associated to the Hamiltonian given by,

$$H(\{s_i\}) = \sum_i h_i s_i + \sum_{i < j} J_{ij} s_i s_j. \quad (1.1)$$

When the state reaches equilibrium, the algorithm gives us the resulting magnetisation and correlation matrix along with the spin configuration history. and The aim is to use a Boltzmann machine to check how well the inferred fields and couplings match the target using the magnetisations and correlations.

1.1 Thermalising Monte-Carlo Sweeps

A threshold number of MC sweeps are required to be made in order to have the spin configuration attain equilibrium. To this end, we conservatively pick the following parameters -

PARAMETERS	VALUES
Size N	5 spins
Burn-in Sweeps n	0 MC sweeps
Total Sweeps S	2^{15} MC sweeps

We perform the MHA by setting no thermalisation time and directly drawing magnetisations M and correlations C_{ij} from 2^{15} configurations, each generated by MC sweeping the previous one. To check whether the system has attained equilibrium, we plot the differences in magnetisation ΔM and correlations ΔC_{ij} between consecutive sweeps (see fig 1.1). Clearly, the fluctuations become negligible as $S \rightarrow 2^{11}$ and thus the states reach equilibrium. Henceforth, we set the Burn-in Sweeps for the MHA as $n = 2^{11}$. Furthermore, we set a relaxation time of 5 sweeps.

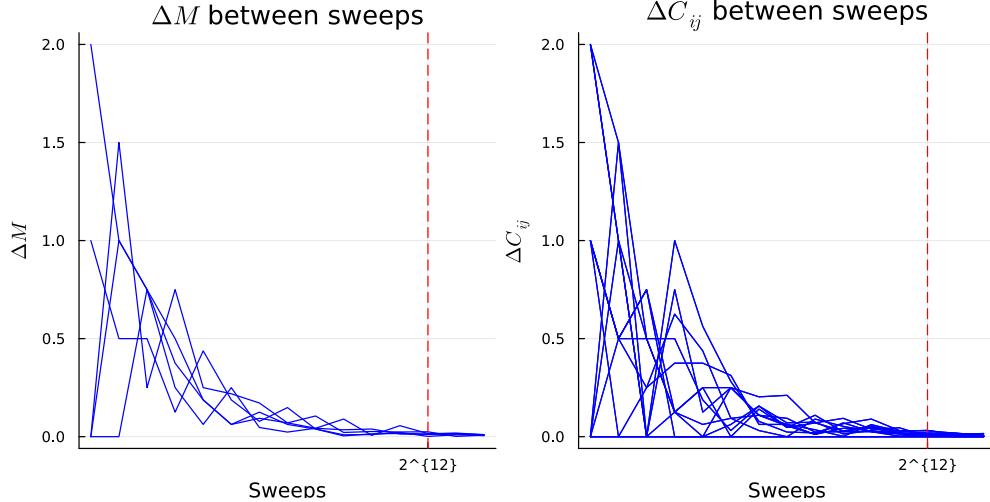


Figure 1.1: Fluctuations of Magnetisation and Correlation over iterations vs # Sweeps using MHA. Equilibrium is attained at the 2^{11} th sweep.

1.2 Boltzmann Learning

The task of a Boltzmann machine is to infer optimal parameters that describe our model by minimising a cost function such as the negative log-likelihood (NLL) given by,

$$\mathcal{L}(\theta) = -\frac{1}{N} \sum_k \log P_\theta(\mathbf{s}^{(k)}), \quad (1.2)$$

where N is the sample size, θ is a model parameter and $P_\theta(\mathbf{s}^{(k)})$ corresponds to the probability of being in a particular spin configuration $\mathbf{s}^{(k)}$ of the sample. This function is minimised by means of a gradient descent. For our model parameters h and J , we have the relations,

$$\begin{aligned} h_i^{(n+1)} &= h_i^{(n)} + \eta \left(\langle s_i \rangle^D - \langle s_i \rangle_\theta \right) \\ J_{ij}^{(n+1)} &= J_{ij}^{(n)} + \eta \left(\langle s_i s_j \rangle^D - \langle s_i s_j \rangle_\theta \right). \end{aligned}$$

Here, $\langle \cdot \rangle^D$ denotes the expectation value of training data and the $\langle \cdot \rangle_\theta$ denotes the expectation value with respect to P_θ . η is what we call the learning rate of the algorithm. We run a Boltzmann machine that accepts initial magnetisations and correlations of a MHA for a random choice of model parameters h and J of a configuration of $N = 5$ random spins with the following parameters -

PARAMETERS	VALUES
Sample Size S	10000 samples
Learning Rates η	0.05, 0.1, 0.5
Iterations	500 steps

We plot the NLL with respect to the number of iterations (see fig 1.2) and we notice that there is an overall decreasing trend as expected for all the learning rates. However, for higher rates, the initial decrease is much more rapid and the NLL fluctuates around a value for a longer number of iterations, suggesting a compromise in the accuracy of our inference. This suspicion is confirmed

when we plot the inferred parameters against the targets (fig 1.3 and 1.4) as we see closest agreement to a perfect fit when $\eta = 0.1$.

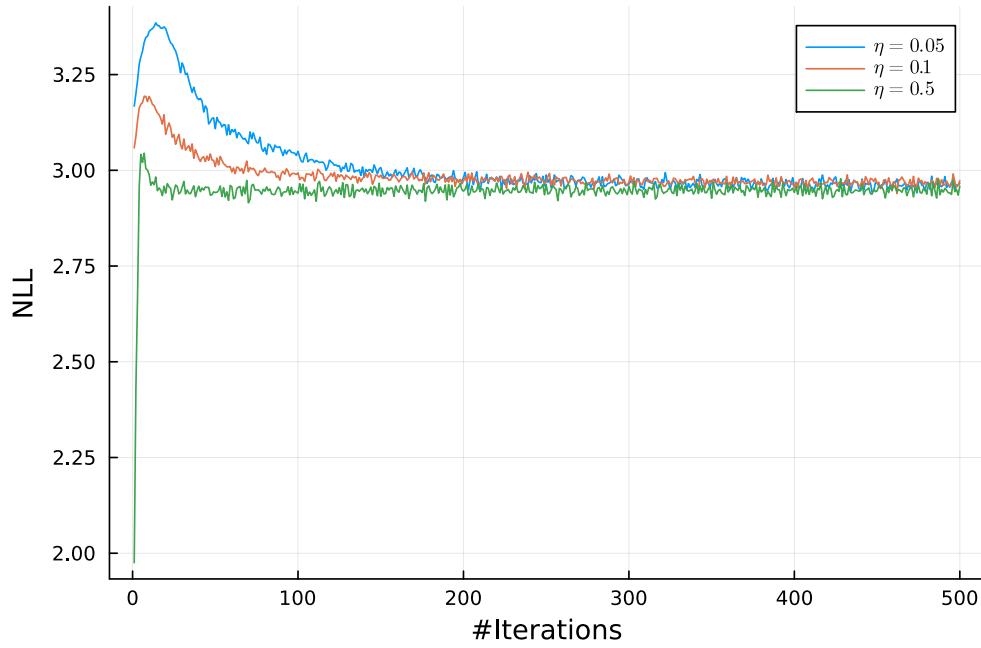


Figure 1.2: Monitoring NLL over Iterations for different learning rates. Best is $\eta = 0.05$.

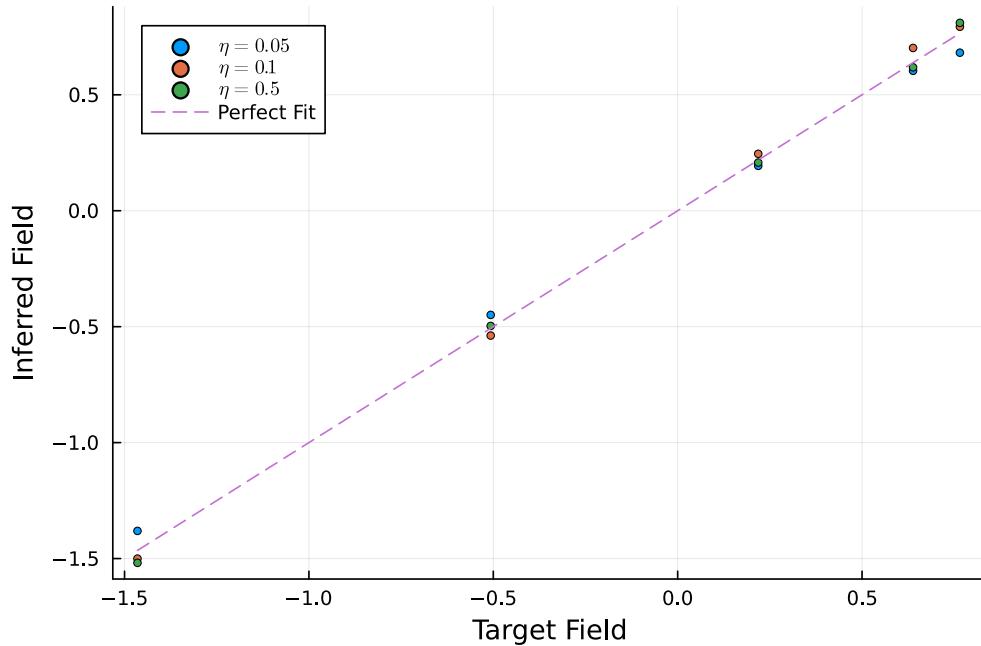


Figure 1.3: Fit of inferred fields vs target fields for different learning rates

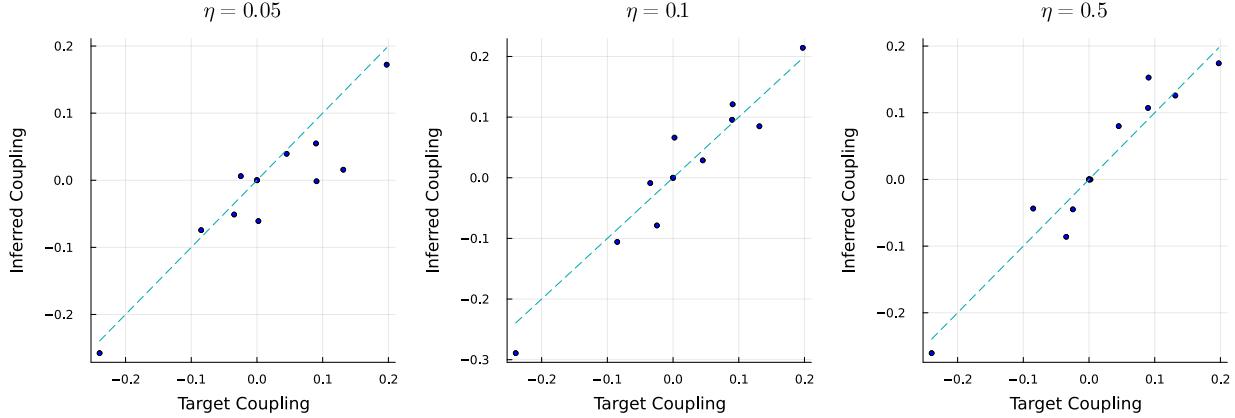


Figure 1.4: Fit of inferred couplings vs target couplings for different learning rates

PARAMETERS	VALUES
Sample Sizes S	$5 \times 10^3, 10^4, 10^5$ samples
Learning Rate η	0.05
Iterations	500 steps

Table 1.1: We further perform the simulation using the above parameters to observe performance with various system sizes.

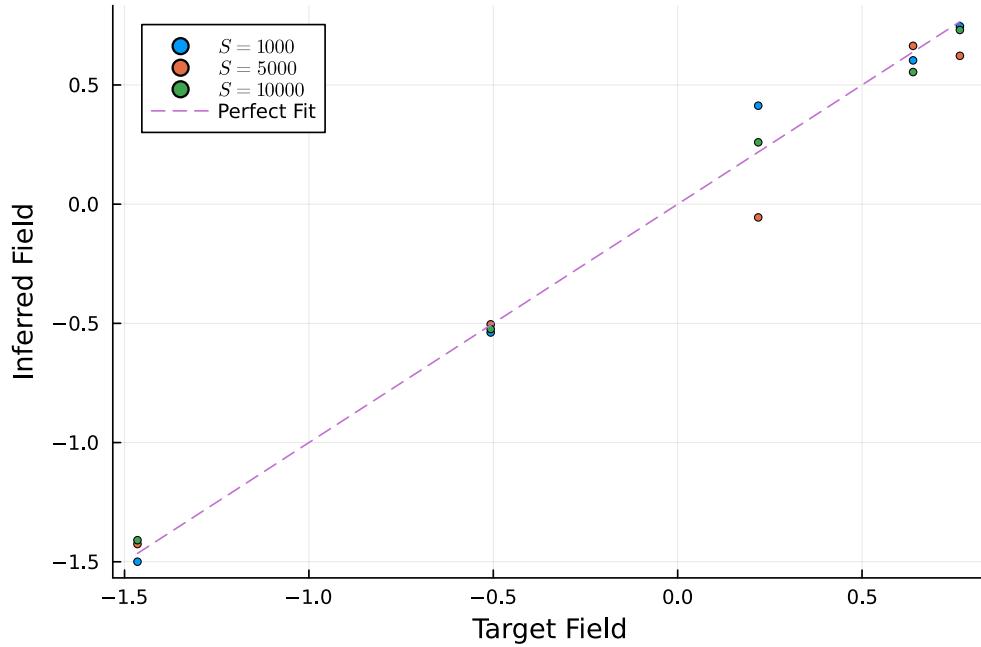


Figure 1.5: Fit of inferred fields vs target fields for different sample sizes

As one can see (fig 1.5, 1.6), the fits seem to improve with sample size.

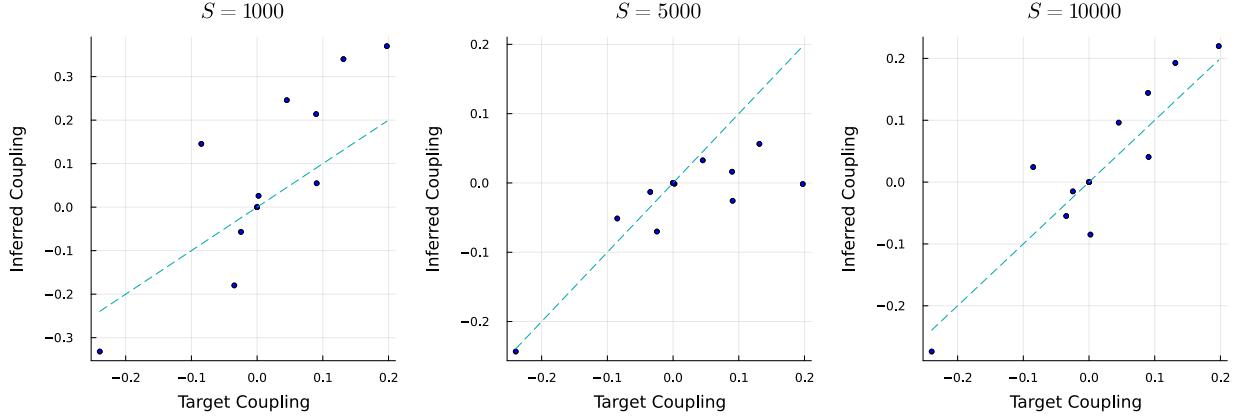


Figure 1.6: Fit of inferred couplings vs target couplings for different sample sizes

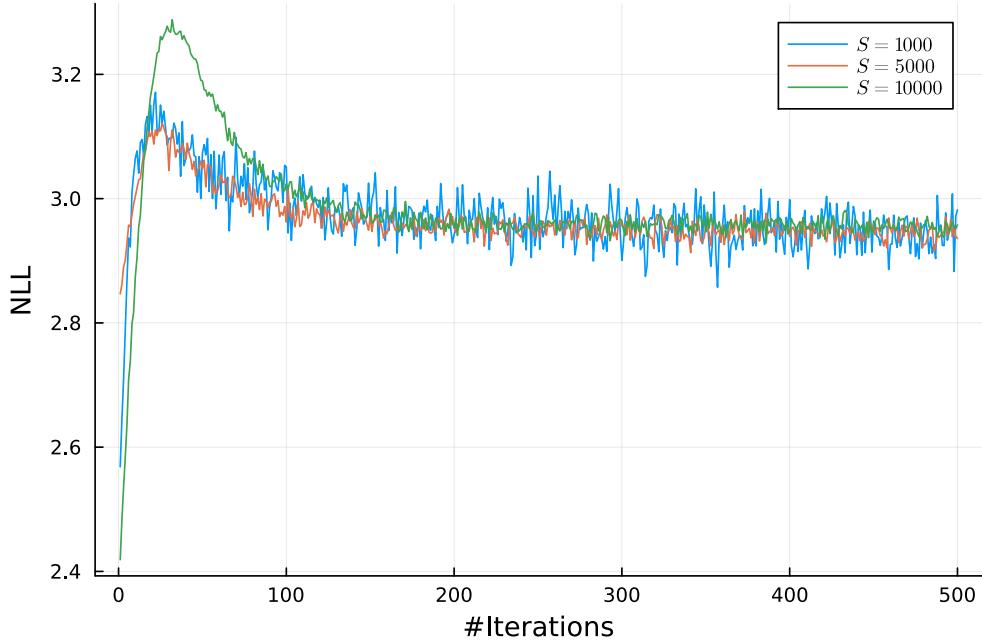


Figure 1.7: Monitoring NLL over Iterations for different sample sizes.

1.2.1 Results

We note that the inference gets better with slower learning rate and higher sample size as expected. The NLL converges slower at lower learning rates but for different sample sizes at the same rate the convergence rate does not change appreciably.

1.3 Glauber Dynamics

A Boltzmann machine inherently assumes that the couplings are symmetric for the given dataset, for if not, then the equilibrium condition of detailed balance is broken[1]. In such cases, we perform the inference using Glauber Dynamics instead which is used to learn from non-equilibrium distributions, such as a time-series. The key difference is a modified NLL given by,

$$\mathcal{L}(\theta) = \frac{1}{M} \sum_{t=1}^{M-1} \sum_i [s_i(t+1)\Theta_i(t) - \ln 2 \cosh(\Theta_i(t))], \quad (1.3)$$

where, $\Theta_i(t) = \sum_j J_{ij}s_j(t) + h_i$ is called an effective field. The spin flip is done using,

$$P\{\vec{s}(t)\} = \prod_i \frac{\exp\{[s_i(t+1)\Theta_i(t)]\}}{2 \cosh(\Theta_i(t))}, \quad (1.4)$$

which is the joint probability of flipping the spins. This leads to a slightly different gradients[2],

$$\begin{aligned} \nabla_{h_i}\mathcal{L}(\theta) &= \frac{1}{M} \sum_{t=1}^{M-1} \sum_i [s_i(t+1) - \tanh \Theta_i(t)] \\ \nabla_{J_{ij}}\mathcal{L}(\theta) &= \frac{1}{M} \sum_{t=1}^{M-1} \sum_i [s_i(t+1)s_j(t) - \tanh(\Theta_i(t))s_j(t)] \end{aligned} \quad (1.5)$$

for our model parameters h and J which are updated by $\eta \nabla_{h_i}\mathcal{L}(\theta)$ and $\eta \nabla_{J_{ij}}\mathcal{L}(\theta)$ respectively. We choose a burn-in of 2^{11} once again as a result of an identical simulation as for the MHA burn-in,

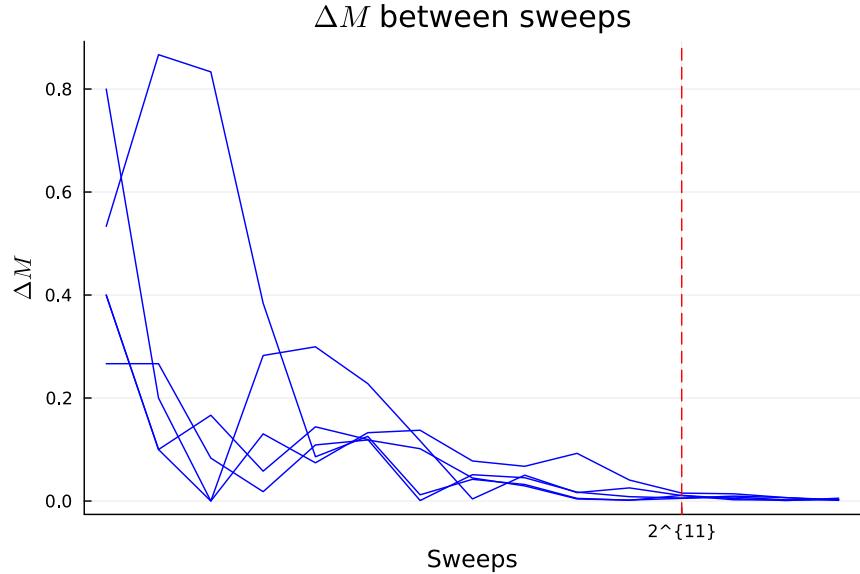


Figure 1.8: Fluctuations in Magnetisation for Glauber Updates

We thus perform our simulations for $N = 5$ random spins using the below parameters,

PARAMETERS	VALUES
Sample Size S	10^5 samples
Learning Rate η	0.05
Iterations	500 steps

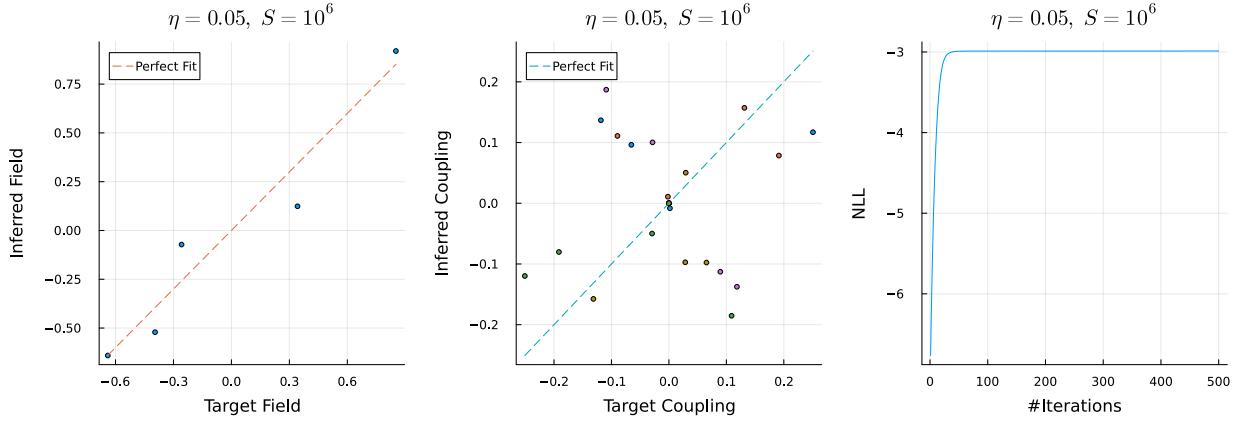


Figure 1.9: Glauber Inference for $N = 5$ random spins for asymmetric couplings

1.3.1 Results

We notice that the Boltzmann machine learns better under the same parameters, but this may be an artefact of the small system size. An important thing to note here is that the glauber inference unlike the boltzmann case maximises the negative log-likelihood and is thus performing a gradient ascent as opposed to a descent.

Chapter 2

Salamander Brain

2.1 Equilibrium Model

We choose one half of the salamander brain data for training and the other half for testing. The training data is put through the Boltzmann machine and we feed the magnetisations and correlations as directly computed from the training data. We use the following parameters for the Boltzmann machine -

PARAMETERS	VALUES
Sample Size S	1000 samples
Learning Rate η	0.05
Iterations	500 steps

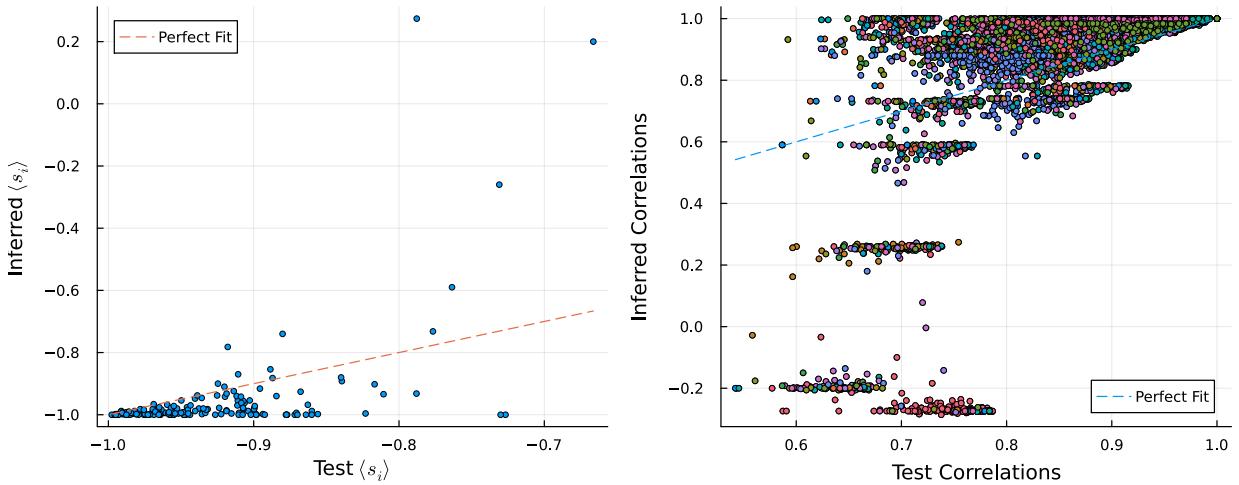


Figure 2.1: Inferred Magnetisations vs Test Magnetisations (Left); Inferred Correlations vs Test Correlations

We compute the magnetisations and correlations using the inferred parameters and plot them against the of the test data. We also plot the spin-spin-spin triplet correlations the same way.

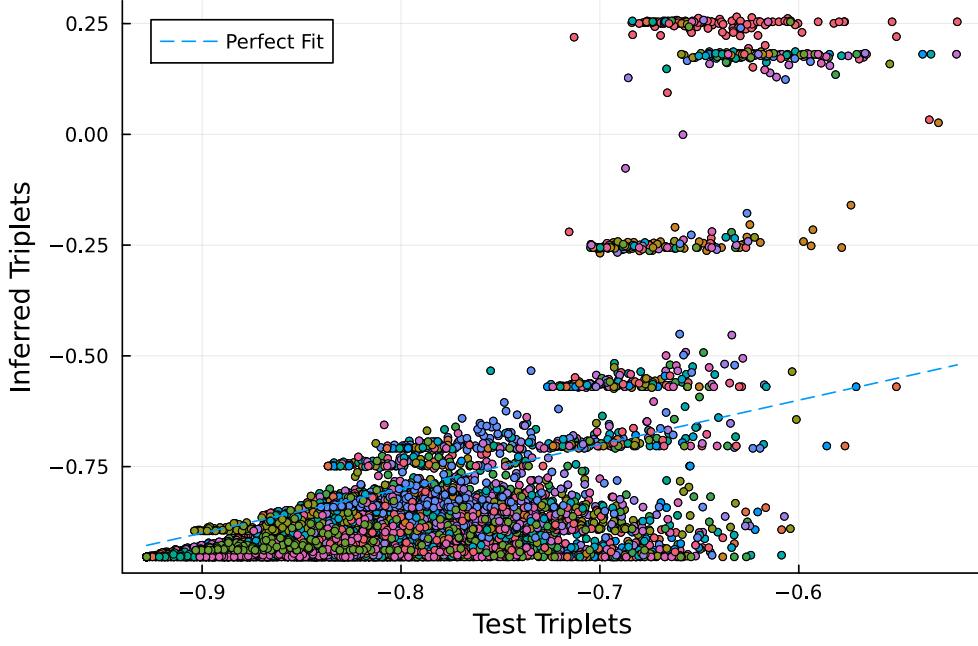


Figure 2.2: Inferred Triplets vs Test Triplets (Left); Inferred Correlations

2.1.1 Results

We observe that the closer the inferred magnetisation is to the ground state ($\langle s_i \rangle \rightarrow -1$), the closer it is to the line of perfect fit, but few of the less negative ones show much higher variance. For the correlations, the perfectly correlated points fall on the straight line. This scatterplot indicates a rather average inference for both the doublets and the triplets.

2.2 Non-Equilibrium Model

We choose the same sample size for inference using glauber dynamics, but set $\eta = 0.3$ and iterations = 100 and plot the differences of temporal correlations between the inferred and test data as a scatterplot for both symmetric and asymmetric couplings.

2.2.1 Results

As can be seen from the fits, both the symmetric nor the asymmetric couplings learn better than the Boltzmann machine. This suggests that the salamander data was not drawn from an equilibrium distribution. The NLL which is slightly slower to converge for the asymmetric couplings.

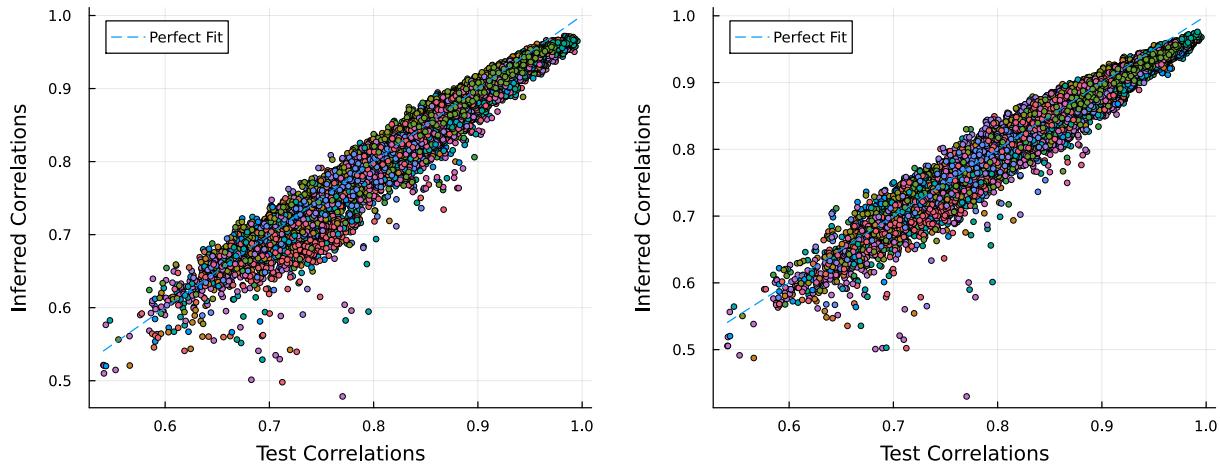


Figure 2.3: Symmetric Couplings (Left) Asymmetric Couplings (Right)

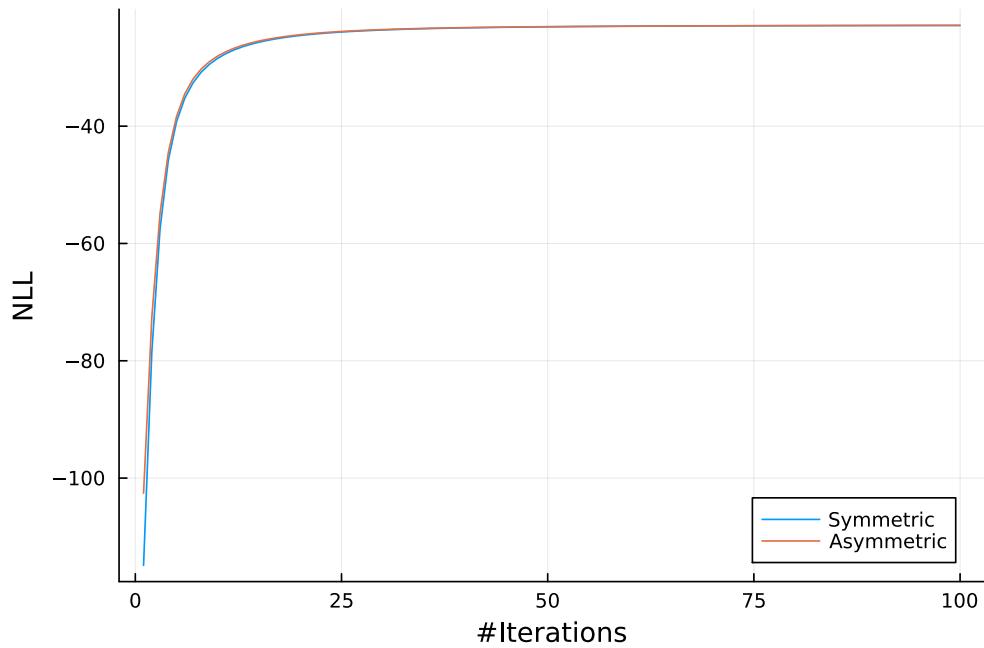


Figure 2.4: NLL comparision for Glauber Inference

Bibliography

- [1] Johannes Berg and Markus Schmitt. *MLab Computational Physics - Statistical Inference*. University of Cologne, 2023.
- [2] H. Chau Nguyen, Riccardo Zecchina, and Johannes Berg. Inverse statistical problems: from the inverse ising problem to data science. *Advances in Physics*, 66:197 – 261, 2017. URL: <https://api.semanticscholar.org/CorpusID:18745274>.