TU Delft

Applied Mathematics

Bachelor End Project

Maximum likelihood estimation of parameters in the exponential random graph model

Author:
Douwe Bosma
Student number: 4379624

Supervisor:
dr. ir. Joris Bierkens

To obtain the degree of
Bachelor of Science in **Applied Mathematics**
at the Delft University of Technology.

November 24, 2017

TU Delft

Applied Mathematics

Bachelor End Project

Members of evaluation commission
dr. ir. Joris Bierkens
dr. ir. Frank van der Meulen
dr. ir. Bart van den Dries

Abstract

In this report the method of Markov chain Monte Carlo maximum likelihood estimation was used to estimate parameters in the Ising model and the exponential random graph model. The method and the models where described mathematically and problems that occurred during the estimation process where discussed. A package that executes the method was built in programming language Julia and is tested on precision. It was concluded that the precision is high in most situation and that, in these situations, the speed of convergence of the estimation can be found in the results.

Contents

1 Introduction

Lately, the application of statistical models for networks are becoming more and more interesting. The few billion people in the world have never been this connected in history and this social system can be seen as an enormous construction of dots and lines. Examples of applications of network models in this field can be found in [RPKL07]. Furthermore, artificial intelligence and machine-learning are hot topics that correspond with a lot of knowledge of neural networks. Take [SHM+16] for example. A neural networks can also be found in our brain, which is so complex that we hardly know anything about the overall process at all. An application of networks can be found in [SHL11]. Though a lot of interesting applications of networks are known, it is necessary to start with the fundamentals of these networks. In this report the focus will be the most basic networks, their properties in context of the exponential random graph model (ERGM) and the method of Markov chain Monte Carlo maximum likelihood estimation (MCMCMLE).

The only networks that are considered are undirected networks, so only vertices and connections between those vertices without direction. In networks certain properties can be looked at. (Such as the number of edges, triangles, stars, sub networks, etc.) The ERGM is a statistical model that analyses networks. In the model it is assumed that the group of networks with a fixed amount of vertices follow a probability distribution. This probability distribution depends on one or more properties of the networks. The importance of a property is scaled with a vector of parameters.

In the end it is desired to estimate the parameters of a network, because it will then be possible to compare different networks in a statistical way. Since the amount of networks with a fixed number of vertices can be extremely large, it is necessary to sample from the probability distribution in a more efficient way. MCMCMLE is a method that finds the most likely values for the parameters of a network making use of a smart way to sample from the probability distribution.

A motive for this report is the extension of my own knowledge of this subject. I think it is a fascinating component of mathematics and it is known that it has a lot of direct and indirect applications. Another goal is to write a package in a programming language that executes the method of MCMCMLE. The software can be found in [Bos17a] and [Bos17b]. The programming language is chosen because of its high performance (See [BKSE12]) and it is called Julia. Finally, some questions concerning the method and the package are answered. Can the method always been carried out? How precise are the results of the package? Are the results in line with the expectation?

First a more simple version of the ERGM, the Curie-Weiss model, and the ERGM itself are discussed. The Curie-Weiss model is a form of the Ising model, a well known model in the world of physics. It will help understanding both models in a easier way. The section is followed by a mathematical introduction to the method of MCMCMLE. The mathematical idea behind Markov chains, Monte Carlo and maximum likelihood estimation will be presented separately. In the next section some lemmas necessary to make the model mathematical

correct are given and proved. After that the results of the Julia program are given in the form of diagrams. Finally in the conclusion and discussion the results will be discussed and future research will be suggested.

2 Mathematical background

In this report two models are studied: the Curie-Weiss model and the exponential random graph model (ERGM). In this section it will first be examined what both models mean. How do they work? What are the properties of the models? It will be noticed that both models are quite similar. The second part of the section will elaborate more on the method we are going to use. In steps Markov chains, Markov chain Monte Carlo and maximum likelihood estimation will be treated.

2.1 Models

2.1.1 The Curie-Weiss model

In this model a set of a fixed amount of d vertices is given. In context of the model the vertices are called *kernels*. The kernels have a positive or a negative spin indicated by $S = \{+1, -1\}$ and are represented by the vertices of the set V. V has d elements. We consider the complete graph/network on the vertices from V. Each combination of positive and negative kernels from V is called a *configuration* and it represents a spin system of atom kernels. The set Ω of all configurations is called the *state space* and is given by

$$\Omega = S^V = \{+1, -1\}^V.$$

There are $|S|^{|V|} = 2^d$ elements in Ω. By $|V|$ we mean the number of elements in set V.

In the model it is assumed that a configuration is a random outcome following a probability distribution p on the set Ω of all configurations. Below we will describe the form of p.

In the model each kernel is connected with each other kernel and they have influence on each other. If, for example, almost all kernels have a positive spin $(+1)$, there is a very high probability that a randomly chosen kernel will become/stay positive.

Each configuration $x \in \Omega$ has a certain amount of energy in the system, given by

$$H(x) = -\frac{1}{2d} \sum_{i,j=1}^{d} x_i x_j$$

found in [LPW09]. Here x_i represents the sign of kernel i of configuration $x \in \Omega$, so x_i is an element of S. The summation runs through all combinations of vertices in V including the combination of vertices with itself. It follows that the energy in configuration x is higher when less pairs of vertices coincide in sign and $H(x) < 0$. The probability distribution p on Ω is given by

$$p_\theta(x) = \frac{e^{-\theta H(x)}}{Z(\theta)}, \quad x \in \Omega. \tag{1}$$

([LPW09] page 43). Here θ is a parameter that can be seen as the inverse of temperature. Physically $\theta \geq 0$ holds, but in our model negative values for θ are tolerated. $Z(\theta)$ is a normalisation function assuring that p_θ is a probability distribution and is given by

$$Z(\theta) = \sum_{x \in \Omega} e^{-\theta H(x)}. \tag{2}$$

2.1.2 Exponential random graph models

In the ERGM model we consider a state space Ω of networks with d vertices, similar to the Curie-Weiss model. Now, however, it is variable whether an edge in the network is present, indicated by $S = \{0, 1\}$. Ω is defined by all possible networks that can be constructed with a fixed amount of d vertices. That is

$$\Omega = \{(V, E) : \#V = d\},$$

where (V, E) is a network with vertices in vertex set V and edges in edge set E. Note that only undirected networks are considered, though the model works in a similar way with directed networks.

The number of possible edges in a network of d vertices is given by $\frac{d(d-1)}{2}$. It follows that the number of elements in Ω is given by $|V|^{|E|} = 2^{\frac{d(d-1)}{2}}$.

As before it is assumed that a network $y \in \Omega$ is a random outcome following a probability distribution p.

To construct the probability distribution on Ω we first define the following vector of functions:

$$\boldsymbol{s}(x) = [s_1(x)\ s_2(x)\ \ldots\ s_{q-1}(x)\ s_q(x)]^{\mathrm{T}}$$

Here $\boldsymbol{s} : \Omega \to \mathbb{R}^q$ is a function.

Example Let s_1 be the function that counts the number of edges in a network (see figure 1). s_1 is defined by

$$s_1(x) = \frac{1}{2} \sum_{i,j \in V} x_{ij}, \quad x \in \Omega.$$

with

$$x_{ij} = \begin{cases} 1, & \text{if graph } x \text{ has an edge between vertices } i \text{ and } j \\ 0, & \text{else.} \end{cases}$$

For $s_2 \ldots s_q$ we count other structures in a network, such as triangles and n-stars of varying order. (See figure 1 and [RPKL07] page 183.) Note that these functions all play a similar role in the ERGM as the function $H(x)$ plays in the Curie-Weiss model.

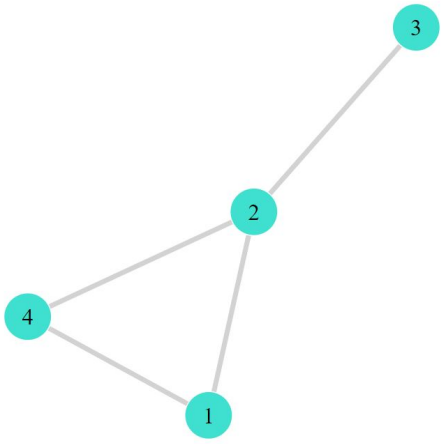

Figure 1: A network with four edges; one triangle through vertices 1, 2 and 4; one 3-star on vertex 2; five 2-stars, two on vertices 1 and 4 and three on vertex 2.

The probability distribution p on Ω is given by

$$p_{\boldsymbol{\theta}}(x) = \frac{\exp[\boldsymbol{\theta} \cdot \boldsymbol{s}(x)]}{Z(\boldsymbol{\theta})}, \quad x \in \Omega. \tag{3}$$

Here $\boldsymbol{\theta}$ is a vector of parameters that defines the weight of the functions $s_1, \ldots, s_q$ and $Z(\boldsymbol{\theta})$ is the same normalisation function to ensure that $p_{\boldsymbol{\theta}}$ is a probability distribution. It is given by

$$Z(\boldsymbol{\theta}) = \sum_{y \in \Omega} \exp[\boldsymbol{\theta} \cdot \boldsymbol{s}(y)]. \tag{4}$$

2.1.3 General model

In summary, both models that where discussed are based on the same principle. In general a set Ω with a certain amount of elements is given. This set is called the state space of the model. It is assumed that the structure of $x \in \Omega$ can be described by a (vector of) function(s) denoted by $\boldsymbol{s}$ or H given by

$$H : x \to \mathbb{R} \quad \text{or} \quad \boldsymbol{s} : x \to \mathbb{R}^q.$$

Furthermore it is assumed that each element x of Ω is a random outcome following distribution

$$p_{\boldsymbol{\theta}}(x) = \frac{\exp[\boldsymbol{\theta} \cdot \boldsymbol{s}(x)]}{Z(\boldsymbol{\theta})}.$$

$p_{\boldsymbol{\theta}}(x)$ can also be notated as $p(x|\boldsymbol{\theta})$.

2.2 Methods

2.2.1 Markov chains

In this section Markov chains are being introduced. Much of the information in this paragraph can be found in [LPW09] on pages 3, 4 and 8.

Markov chains In a Markov chain process, in time it is moved between the elements (states) of a finite set (state space), say Ω. When the Markov process is at $y \in \Omega$, the next step is determined by a probability distribution $p(y, \cdot)$ on Ω. These probabilities can be put into a matrix $\boldsymbol{P}$ that is called the *transition matrix* of the Markov chain in question. See the next example.

A Markov chain will be defined in a more formal way now.

Definition 2.1. (See page 2 of [Nor98].) We say that $(Y_n)_{n\geq 0}$ is a Markov chain with initial distribution λ and transition matrix $\boldsymbol{P}$ if

1. Y_0 has distribution λ;
2. for $n \geq 0$, conditional on $Y_n = y$, Y_{n+1} has distribution $(p_{yz} : z \in \Omega)$ and is independent of $Y_0, \ldots, Y_{n-1}$.

This last point is called the Markov property. It means that state z at time $t+1$ is determined by state y at time t only, no matter what the sequence $(y_0\ y_1 \ldots y_{t-1})$ of states may be.

$(Y_n)_{n\geq 0}$ is said to be $Markov(\lambda, \boldsymbol{P})$ for short.

Example Suppose a network with two vertices is given, representing two locations and the connection between them (see figure 2) and suppose that at $t = 0$ we are at location 1. Let us say that the probability to move from vertex 1 to 2 is equal to p and to move from 2 to 1 is equal to q. This means that the probability to stay on vertex 1 is $1 - p$ and the probability to stay on 2 is $1 - q$. In matrix-form this becomes

$$\boldsymbol{P} = \begin{bmatrix} 1-p & p \\ q & 1-q \end{bmatrix}.$$

A sequence of random variables $(Y_0\ Y_1\ Y_2\ \ldots)$ following the above probabilities to move to/stay on a vertex is called a Markov chain with transition matrix $\boldsymbol{P}$.

Let $i, j \in \Omega$. We say that i leads to j and write $i \to j$ if

$$P(Y_n = j \text{ for some } n \geq 0 | Y_m = i) > 0 \quad \text{for some } m < n.$$

In words $i \to j$ means that j can be reached with a positive probability within a finite amount of steps from starting state i.

Definition 2.2. Let λ be any distribution. A Markov chain that is Markov$(\lambda, \boldsymbol{P})$ with state space Ω is irreducible when $i \to j$ holds for all $i, j \in \Omega$

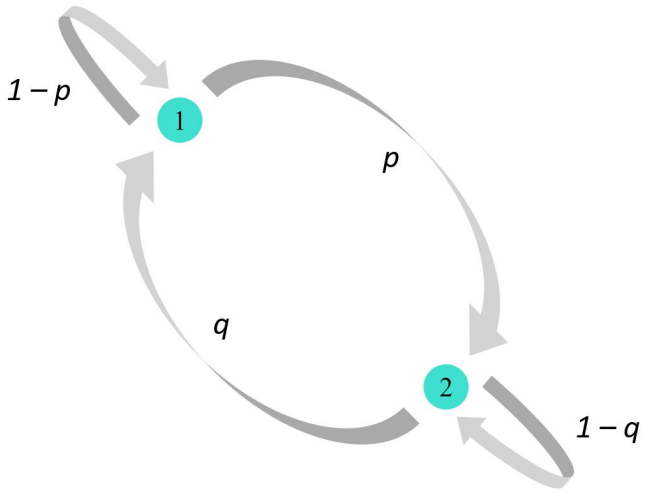

Figure 2: A network with two vertices and one edge.

Let $\boldsymbol{P}$ be a transition matrix on state space Ω. $i \in \Omega$ is called *aperiodic* if $p_{ii}^{(n)} > 0$ for all sufficient large n.

Lemma 2.1. *Suppose $\boldsymbol{P}$ is irreducible and has an aperiodic state i. Then, for all states j and k, $p_{jk}^{(n)} > 0$ for all sufficient large n. In particular, all states are aperiodic.*

Proof. See page 41 in [Nor98]. □

Let $\boldsymbol{P}$ be a transition matrix on state space Ω. Distribution π is called the *stationary distribution* of $\boldsymbol{P}$ if

$$\pi = \boldsymbol{P}\pi$$

holds.

Lemma 2.2. *Let $\boldsymbol{P}$ be irreducible and aperiodic, and suppose that $\boldsymbol{P}$ has an stationary distribution π. Let λ be any distribution. Suppose that $(X_n)_{n\geq 0}$ is Markov$(\lambda, \boldsymbol{P})$. Then*

$$P(Y_n = j) \to \pi_j \quad \text{as } n \to \infty \quad \text{for all } j.$$

In particular,

$$p_{ij}^{(n)} \to \pi_j \quad \text{as } n \to \infty \text{ for all } i, j.$$

Proof. See pages 41 and 42 in [Nor98]. □

Example (continued) Suppose that the vector

$$\mu_t = [P(Y_t = 1|Y_0 = 1), P(Y_t = 2|Y_0 = 1)] \tag{5}$$

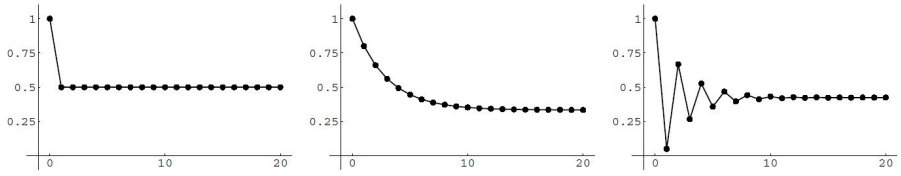

Figure 3: Source: [LPW09] page 4. Three plots of the probability to be at location 1 (starting on location 1). (a) $p = q = 0.5$, (b) $p = 0.2$ and $q = 0.1$, (c) $p = 0.95$ and $q = 0.7$.

is defined. It follows that $\mu_1 = \mu_0 \boldsymbol{P}$ and $\mu_2 = \mu_0 \boldsymbol{P}^2$ etc. In general $\mu_t = \mu_0 \boldsymbol{P}^t$ holds.

Figure 3 suggests that eventually $\pi = \pi \boldsymbol{P}$ holds for certain distribution π. We will take the limit of μ_t.

$$\lim_{t \to \infty} \mu_t = \pi.$$

π is called the stationary distribution of matrix $\boldsymbol{P}$. In this case π is given by

$$\pi(1) = \frac{p}{p+q} \text{ and } \pi(2) = \frac{q}{p+q}.$$

In general the same can be done. Again let λ be any distribution. let $(Y_0\ Y_1\ Y_2\ \ldots)$ be Markov$(\lambda, \boldsymbol{P})$ with state space Ω. Let μ_t be a row vector with the distribution of Y_t:

$$\mu_t(y) = P(Y_t = y) \quad \text{for all } y \in \Omega. \tag{6}$$

When μ_{t+1} is constructed, matrix $\boldsymbol{P}$ is used:

$$\mu_{t+1}(z) = \sum_{y \in \Omega} P(Y_t = y) \boldsymbol{P}(y, z) = \sum_{y \in \Omega} \mu_t(y) \boldsymbol{P}(y, z) \quad \text{for all } z \in \Omega. \tag{7}$$

In other words

$$\mu_{t+1} = \mu_t \boldsymbol{P}.$$

This implies

$$\mu_t = \mu_0 \boldsymbol{P}^t.$$

All Markov chains that are considered in this report are irreducible and aperiodic. So suppose that λ is any distribution and $(Y_n)_{n \geq 0}$ is a Markov chain used in this report that is Markov$(\lambda, \boldsymbol{P})$ with stationary distribution π. Then

$$\lim_{t \to \infty} \mu_t = \pi. \tag{8}$$

holds.

From practical perspective it is wanted to prevent the limit from t to ∞ to obtain the stationary distribution π. Therefore the amount of time t such that μ_t is close to π is needed. This time is called the *mixing time* of a Markov chain. In [DLP09] a more formal definition can be found.

Glauber Dynamics A special form of Markov chains is called Glauber dynamics. Let S and V be finite sets. Let the state space Ω be given by S^V. (In the Curie-Weis model, for example, $S = \{1, -1\}$ and V is the set of all kernles.) Let π be a probability distribution on Ω. The Glauber dynamics for π is a Markov chain with state space Ω and stationary distribution π.

Let $x \in \Omega$ and let $v \in V$. (In the Curie-Weiss model, by x_v we mean the sign of kernel v in configuration x.) Let $x_{\hat{v}} = \{x_1, \ldots, x_{v-1}, x_{v+1}, \ldots, x_d\}$. A step in the Glauber dynamics starting on x is determined by the following two iterations:

- Choose $v \in V$ uniformly.
- Choose $y \in \Omega$ randomly from distribution $\pi(x_v|x_{\hat{v}})$.

In words, a step from state $x \in \Omega$ in the Glauber dynamics is determined by choosing a vertex/element $v \in V$ uniformly. A new state y is chosen randomly from distribution π conditional on $x_{\hat{v}}$. This means that x and y agree on each element of V except for v.

Before it can be confirmed that the stationary distribution of the Glauber dynamics for π is indeed π itself, a definition and a theorem are presented.

Definition 2.3. A stochastic matrix $\boldsymbol{P}$ and distribution π is said to be in *detailed balance* if

$$\pi(x)\boldsymbol{P}(x, y) = \pi(y)\boldsymbol{P}(y, x) \quad \text{for } x, y \in \Omega$$

For the next theorem and proof, see [Nor98] page 48.

Theorem 2.1. *If $\boldsymbol{P}$ and π are in detailed balance, then π is invariant for $\boldsymbol{P}$. That is, π is the stationary distribution of the Markov chain with transition matrix $\boldsymbol{P}$.*

Lemma 2.3. *The Glauber dynamics for π has stationary distribution π.*

Proof. Take $x, y \in \Omega$. It follows that the transition matrix $\boldsymbol{P}$ for the Glauber dynamics for π is given by

$$\boldsymbol{P}(x, y) = \frac{1}{|V|} \sum_{v \in V} \pi(x_v = y_v|x_{\hat{v}}) \cdot \mathbb{1}_{x_{\hat{v}} = y_{\hat{v}}}.$$

Furthermore we know that

$$\pi(x) = \pi(x_{\hat{v}})\pi(x_v|x_{\hat{v}}).$$

So it follows that

$$\begin{aligned} \pi(x)\boldsymbol{P}(x, y) &= \frac{1}{|V|} \sum_{v \in V} \pi(x)\pi(x_v = y_v|x_{\hat{v}}) \cdot \mathbb{1}_{x_{\hat{v}} = y_{\hat{v}}} \\ &= \frac{1}{|V|} \sum_{v \in V} \pi(x_{\hat{v}})\pi(x_v|x_{\hat{v}})\pi(x_v = y_v|x_{\hat{v}}) \cdot \mathbb{1}_{x_{\hat{v}} = y_{\hat{v}}}. \end{aligned}$$

By interchanging x_v and y_v we obtain an expression that is equal to

$$\pi(y)\boldsymbol{P}(y,x).$$

It can be concluded that $\boldsymbol{P}$ and π are in detailed balance. By theorem 2.1 the stationary distribution of this Glauber dynamics is equal to π. □

2.2.2 Markov chain Monte Carlo

Suppose we have probability distribution π. Sometimes direct sampling from a probability distribution like π is hard. This can be a consequence of a computationally expansive normalisation function of the distribution. Luckily there is another, more effective, way to sample from such a distribution.

As seen before, the Markov chains that are considered converge to a stationary distribution. Suppose a Markov chain with stationary distribution π is given. A sample from a distribution close to π can be generated by taking enough steps in the Markov chains. This follows from equation (8). In this way we could create a sequence of samples from a distribution very close to π.

The process of sampling from a distribution making use of Markov chains is called Markov chain Monte Carlo (MCMC). There are different algorithms for the process. The one that we are using in this report is called Gibbs sampling or the Metropolis-Hastings algorithm.

Example (ERGM) Suppose we have a Markov chain with transition matrix $\boldsymbol{P}$ and state space Ω. Here Ω is the set of all possible graphs with vertices V from paragraph 2.1.2. Suppose $\boldsymbol{\theta}$ is given. The probability distribution corresponding to this vector of parameters $p_{\boldsymbol{\theta}}$ is given by equation (3). Now we can generate a network based on the parameters by making use of the Glauber dynamics with stationery distribution $p_{\boldsymbol{\theta}}$. For the Curie-Weiss model it works in an analogous way.

Suppose that we repeat this process for large number of steps. After enough outcomes we could approach the real probability distribution in equation (3) associated with $\theta_1,\ \theta_2,\ \theta_3$ without actually calculating the $Z(\theta_1,\theta_2,\theta_3)$ function.

For this method it is important to know the mixing time of the Glauber dynamics for both models. In [DLP09] the mixing time for the Curie-Weiss model is examined. It follows that the mixing time of the Glauber dynamics for the Curie-Weiss model is of order $d\log d$.

2.2.3 Maximum likelihood estimation

Suppose $X_1,\dots,X_n$ are independent and identically distributed random variables following distribution $f(\cdot|\boldsymbol{\theta})$. Given the observed values $x_1,\dots,x_n$ we can construct the likelihood function:

$$\mathcal{L}(\boldsymbol{\theta};x_1,\dots,x_n) = f(x_1,\dots,x_n|\boldsymbol{\theta}) = \prod_{i=1}^{n} f(x_i|\boldsymbol{\theta}). \tag{9}$$

This function is the probability of observing the given data as function of $\boldsymbol{\theta}$. Let $\hat{\boldsymbol{\theta}}$ be the value of $\boldsymbol{\theta}$ corresponding to the global maximum of the function. $\hat{\boldsymbol{\theta}}$ is called the maximum likelihood estimator of equation (9) and it is the most likely value for the parameter $\boldsymbol{\theta}$ corresponding to $x_1, \ldots, x_n$.

An easier way to find $\hat{\boldsymbol{\theta}}$ is by making use of the log-likelihood function. This function has the same maximum likelihood estimator $\hat{\boldsymbol{\theta}}$ as the likelihood function. The log-likelihood function is defined by

$$l(\boldsymbol{\theta}; x_1, \ldots, x_n) = \log\left\{\prod_{i=1}^{n} f(x_i|\boldsymbol{\theta})\right\} = \sum_{i=1}^{n} \log[f(x_i|\boldsymbol{\theta})]. \tag{10}$$

Here we use the fact that the logarithm is an increasing function.

In general the form of the likelihood function for both the Curie-Weiss model and the ERGM is given by

$$\mathcal{L}(\boldsymbol{\theta}; x) = p(x|\boldsymbol{\theta}) = \frac{h_{\boldsymbol{\theta}}(x)}{Z(\boldsymbol{\theta})} \tag{11}$$

where $h_{\boldsymbol{\theta}}(x) = \exp[\boldsymbol{\theta} \cdot \boldsymbol{s}(x)]$. It follows that the log-likelihood function is

$$l(\boldsymbol{\theta}; x) = \boldsymbol{\theta} \cdot \boldsymbol{s}(x) - \log\left\{\sum_{y \in \Omega} \exp[\boldsymbol{\theta} \cdot \boldsymbol{s}(y)]\right\} \tag{12}$$

To find the maximum of this function, the gradient is used. However, when we are dealing with larger networks, it is very hard to differentiate the second term of equation (12). Therefore another way to find the maximum is needed.

The approached likelihood function Suppose a fixed and arbitrary vector of parameters $\boldsymbol{\psi}$ is given. Then for a fixed observation x, $p(x|\boldsymbol{\psi})$ is a constant value. By subtracting the logarithm of this value from equation (12) nothing essential is changed in $l(\boldsymbol{\theta}; x)$. The function has got the same maximum likelihood estimator $\hat{\boldsymbol{\theta}}$. Therefore equation (12) is equivalent to

$$\begin{aligned} l(\boldsymbol{\theta}; x) =& \log\left\{\frac{p(x|\boldsymbol{\theta})}{p(x|\boldsymbol{\psi})}\right\} = \log\left\{\frac{h_{\boldsymbol{\theta}}(x)}{h_{\boldsymbol{\psi}}(x)}\right\} - \log\left\{\frac{Z(\boldsymbol{\theta})}{Z(\boldsymbol{\psi})}\right\} \\ =& (\boldsymbol{\theta} - \boldsymbol{\psi}) \cdot \boldsymbol{s}(x) - \log\left\{\mathbb{E}_{\boldsymbol{\psi}}\left[\frac{h_{\boldsymbol{\theta}}(Y)}{h_{\boldsymbol{\psi}}(Y)}\right]\right\} \end{aligned} \tag{13}$$

since

$$\mathbb{E}_{\boldsymbol{\psi}}\left[\frac{h_{\boldsymbol{\theta}}(Y)}{h_{\boldsymbol{\psi}}(Y)}\right] = \frac{Z(\boldsymbol{\theta})}{Z(\boldsymbol{\psi})}. \tag{14}$$

(See [Gey94].) In these equations Y is a random variable with probability distribution $p_{\boldsymbol{\psi}}$.

Calculating the expectation in equation (13) is hardly possible when the observed network is too large. A natural way to solve this problem is by making use of the 'empirical' expectation with respect to $p_{\boldsymbol{\psi}}$ denoted by $\mathbb{E}_{n,\boldsymbol{\psi}}$.

Let $Y_1, \ldots, Y_n$ be samples from $p_{\boldsymbol{\psi}}$ generated by the method of Markov chain Monte Carlo. $\mathbb{E}_{n,\boldsymbol{\psi}}$ is then given by

$$\mathbb{E}_{n,\boldsymbol{\psi}}\left[\frac{h_{\boldsymbol{\theta}}(Y)}{h_{\boldsymbol{\psi}}(Y)}\right] = \frac{1}{n}\sum_{i=1}^{n}\frac{h_{\boldsymbol{\theta}}(Y_i)}{h_{\boldsymbol{\psi}}(Y_i)}.$$

The Monte Carlo approximation of the log-likelihood function is thus given by

$$l_n(\boldsymbol{\theta}; x) = (\boldsymbol{\theta} - \boldsymbol{\psi}) \cdot \boldsymbol{s}(x) - \log\left\{\frac{1}{n}\sum_{i=1}^{n}\exp[(\boldsymbol{\theta} - \boldsymbol{\psi})\boldsymbol{s}(Y_i)]\right\}. \tag{15}$$

Proposition 2.1. *The maximum likelihood estimator $\hat{\boldsymbol{\theta}}_n$ of equation (15) converges to the maximum likelihood estimator $\hat{\boldsymbol{\theta}}$ of equation (12) when $n \to \infty$.*

Proof. It is known that the parameter set $\Theta = \mathbb{R}^d$, which implies that it is a separable metric space. Furthermore the function $\boldsymbol{\theta} \mapsto h_{\boldsymbol{\theta}}(x)$ is a continuous function. Thirdly all Markov chains we use in the Metropolis-Hastings (Markov chain Monte Carlo) algorithm in this report are irreducible.

These are sufficient conditions for theorem 1 ([Gey94] page 264). From the theorem it follows that $l_n(\boldsymbol{\theta}; x)$ (equation (15)) hypo-converges to $l(\boldsymbol{\theta}; x)$ (equation (12)). This means that if $l_n(\boldsymbol{\theta}; x) \to l(\boldsymbol{\theta}; x)$ as $n \to \infty$, than $\hat{\boldsymbol{\theta}}_n \to \hat{\boldsymbol{\theta}}$ as $n \to \infty$. □

2.2.4 Markov chain Monte Carlo maximum likelihood estimation

Given $x \in \Omega$, we know that it is not possible to calculate $\boldsymbol{\theta}$ in a direct way if x is too complex. This is a consequence of a normalisation function $Z(\boldsymbol{\theta})$ that is computational expensive. Therefore the method of maximum likelihood estimation is used.

The Glauber dynamics is used to generate a group of n samples from $p_{\boldsymbol{\psi}}$ without knowing $Z(\boldsymbol{\psi})$. These samples are used to calculate the empirical expectation from l_n in section 2.2.3. The more samples, the better l_n approximates the original likelihood l and $l_n \to l$ as $n \to \infty$ holds.

Finally the maximum likelihood estimator $\hat{\boldsymbol{\theta}}_n$ of l_n is found. It is known that $\hat{\boldsymbol{\theta}}_n \to \hat{\boldsymbol{\theta}}$ as $n \to \infty$. At this point the method of Markov chain Monte Carlo maximum likelihood estimation is used to estimate the value of $\boldsymbol{\theta}$ without calculating the giant $Z(\boldsymbol{\theta})$ function.

There are still a few questions, though.

1. In this report the method of MCMCMLE is carried out by a self written package in Julia. See [Bos17a] and [Bos17b]. This package makes use of an optimisation function to find the maximum of the log-likelihood function. How do we know for sure that this maximum is the global maximum of the function and not a local maximum?

2. The Glauber dynamics makes use of steps that are repeated for a number of times. After order $d \log d$ of steps the position of the method is independent of the starting position for $\theta < 0$ in the Curie-Weiss model and the mixing time is reached. See [DLP09]. How big is this mixing time exactly?

3. How do we calculate the probability $P_{\boldsymbol{\theta}}(x_v | x_{\hat{v}})$ that is used in the Glauber dynamics for both models?

In the next chapter among others, these questions are discussed.

3 Approach

In section 2.2 the general and mathematical idea behind our models and method was discussed. There was a description of the method of Markov chain Monte Carlo maximum likelihood estimation and introduction to the Curie-Weiss model and the exponential random graph model. In this section we are going to proof some essential mathematical statements.

First of all, both models are discussed in context of the Glauber dynamics. Two lemmas are given here. Secondly it is proven that under certain circumstances the previously mentioned likelihood function has only one maximum which implies that our method is not calculating the wrong (local) maximum.

3.1 Steps in the Glauber dynamics

3.1.1 Curie-Weiss model

As described in section 2.2.1, to move between the elements of Ω a random kernel is chosen uniformly and the probability for that kernel to become or stay positive/negative is determined. Remember that x_v is kernel v of configuration x. Let $x_{\hat{v}} = \{x_1, \ldots, x_{v-1}, x_{v+1}, \ldots, x_d\}$.

Lemma 3.1. *The probability of kernel $v \in V$ to become or stay positive, given the sign of all other kernels in V, is*

$$P(x_v = 1 | x_{\hat{v}}) = \frac{1 + \tanh[\theta H(x)]}{2} \tag{16}$$

Proof.

$$\begin{aligned} P(x_v = 1 | x_{\hat{v}}) &= \frac{P(x_v = 1, x_{\hat{v}})}{P(x_{\hat{v}})} = \frac{P(x_v = 1, x_{\hat{v}})}{P(x_v = -1, x_{\hat{v}}) + P(x_v = 1, x_{\hat{v}})} \\ &= \frac{e^{-\theta H(\{1, x_{\hat{v}}\})}}{e^{-\theta H(\{-1, x_{\hat{v}}\})} + e^{-\theta H(\{1, x_{\hat{v}}\})}} \\ &= \frac{e^{-\theta s(x,v)}}{e^{\theta s(x,v)} + e^{-\theta s(x,v)}} \\ &= \frac{1 + \tanh[\theta H(x)]}{2}. \end{aligned}$$

□

3.1.2 Exponential Random Graph Model

To move between the elements of Ω making use of the Glauber dynamics, two random kernels in V are chosen. After that the probability of the edge in between the two kernels to exist, given all other edges is calculated.

In the following part of this report the vector of functions $\boldsymbol{s}(x)$ will be given by $[s_1(x)\ s_2(x)]^T$ in context of the ERGM. s_1 and s_2 are respectively the functions that count the number of edges and triangles in a network. It follows that $\boldsymbol{\theta} = [\theta_1\ \theta_2]^T$.

let $x_{\hat{ij}} = E \setminus x_{ij}$ be the set of all existing and non-existing edges in network x. Let $\Delta(x_{ij})$ be the function that counts the number of triangles through edge $i \sim j$.

Lemma 3.2. *The probability of edge $i \sim j,\ i, j \in V$ to be added to a network x, given the (non)-existence of all other edges, is*

$$P(x_{ij} = 1 | x_{\hat{ij}}) = \frac{1}{1 + \exp[-\theta_1 - \theta_2 \cdot \Delta(x_{ij} = 1)]}$$

Proof.

$$\begin{aligned}
P(x_w = 1 | x_{\hat{ij}}) =& \frac{P(x_w = 1, x_{\hat{ij}})}{P(x_w = 0, x_{\hat{ij}}) + P(x_w = 1, x_{\hat{ij}})} \\
=& \frac{\exp[\boldsymbol{\theta} \cdot \boldsymbol{s}(x_{ij} = 1, x_{\hat{ij}})]}{\exp[\boldsymbol{\theta} \cdot \boldsymbol{s}(x_{ij} = 1, x_{\hat{ij}})] + \exp[\boldsymbol{\theta} \cdot \boldsymbol{s}(x_{ij} = 0, x_{\hat{ij}})]} \\
=& \frac{1}{1 + \exp[\boldsymbol{\theta} \cdot \boldsymbol{s}(x_{ij} = 0, x_{\hat{ij}}) - \boldsymbol{\theta} \cdot \boldsymbol{s}(x_{ij} = 1, x_{\hat{ij}})]} \\
=& \frac{1}{1 + \exp[-\theta_1 + \theta_2 (s_2(\{x_{ij} = 0, x_{\hat{ij}}\}) - s_2(\{x_{ij} = 1, x_{\hat{ij}}\})]} \\
=& \frac{1}{1 + \exp[-\theta_1 - \theta_2 \cdot \Delta(x_{ij} = 1)]}.
\end{aligned}$$

□

3.2 Global maximum of the log-likelihood functions

The (Monte Carlo) log-likelihood functions considered here, equation (15) and (12), give (an approximation of) the maximum likelihood estimator for the vector of parameter $\boldsymbol{\theta}$. This is done by calculating the extreme value of the function.

Because of the complex functions it is not certain whether our package in Julia is calculating the global or a local maximum of the log-likelihood function. To tackle this problem, it is shown that the functions have exactly one maximum. It is proven that $l_n(\boldsymbol{\theta}; x)$ and $l(\boldsymbol{\theta}; x)$ are concave functions **with** an extreme value for $\boldsymbol{\theta} \in \mathbb{R}^d$.

3.2.1 Example: the log-likelihood function for the one-dimensional case

Before the concavity and the presence of a maximum is proven in general, we will take a look at the Monte Carlo log-likelihood function corresponding to the Curie-Weiss model. This function is very much the same as the one dimensional version of the Monte Carlo log-likelihood in the ERGM. With this function the proof is more intuitive because it is one-dimensional only.

In the Curie-Weiss model the Monte Carlo log-likelihood function is given by

$$l_n(\theta; x) = (\psi - \theta)H(x) - \log\left\{\frac{1}{n}\sum_{i=1}^{n}\exp[(\psi - \theta)H(Y_i)]\right\}. \tag{17}$$

In this equation the Y_i's are the samples of $p_{\boldsymbol{\psi}}$, making use of the Glauber dynamics. x is the configuration we are observing. Note that substituting $s(x)$ for $-H(x)$ gives us the general one dimensional Monte Carlo log-likelihood function. In the this example we refer to equation (17) with $l_n(\theta; x)$.

Let $A_i = e^{(\psi-\theta)H(Y_i)}$. It can be checked that the first and second derivative of $l_n(\theta; x)$ are given by

$$\frac{\partial l_n(\theta; x)}{\partial \theta} = -H(x) + \frac{\sum_{i=1}^{n} H(Y_i) \cdot A_i}{\sum_{i=1}^{n} A_i} \quad \text{and} \tag{18}$$

$$\frac{\partial^2 l_n(\theta; x)}{\partial \theta^2} = -\frac{\left(\sum_{i=1}^{n} A_i\right)\left(\sum_{i=1}^{n} H(Y_i)^2 \cdot A_i\right) + \left(\sum_{i=1}^{n} H(Y_i) \cdot A_i\right)^2}{\left(\sum_{i=1}^{n} A_i\right)^2}. \tag{19}$$

Since $H(x) \neq 0$ for all $x \in \Omega$ it follows that, as long as $\theta, \psi \in (-\infty, \infty)$ holds,

$$\sum_{i=1}^{n} A_i > 0, \quad \sum_{i=1}^{n} H(Y_i)^2 \cdot A_i > 0.$$

Therefore

$$\frac{\partial^2 l_n(\theta; x)}{\partial \theta^2} < 0 \tag{20}$$

holds as long as $\theta, \psi \neq \pm\infty$. In this report the focus will be on parameters close to the high-temperature area and ψ is an arbitrary fixed parameter that

we choose as the value we expect θ to be a priori. It follows that θ and ψ are not close to $\pm\infty$ and $l_n(\theta; x)$ is a strictly concave function.

Lemma 3.3. *Suppose* $H_{\min} = \min\{H(Y_i) : 1 \leq i \leq n\}$ *and* $H_{\max} = \max\{H(Y_i) : 1 \leq i \leq n\}$. *Then* $l_n(\theta; x)$ *has a maximum on* $\theta \in \mathbb{R}$ *if and only if* $H_{\min} < H(x) < H_{\max}$ *holds.*

Proof. Let $I_{\min} = \{i : H(Y_i) = H_{\min}\}$ and let, in an analogous way, $I_{\max} = \{i : H(Y_i) = H_{\max}\}$. It can be checked that

$$\frac{\partial l_n(\theta; x)}{\partial \theta} = \sum_{i=1}^{n} H(Y_i) \cdot p_i^\theta - H(x)$$

with

$$\begin{aligned} p_i^\theta =& \frac{\exp[(\psi - \theta)H(Y_i)]}{\sum_j \exp[(\psi - \theta)H(Y_j)]} \cdot \frac{\exp[(\theta H_{\min}]}{\exp[(\theta H_{\min}]} \\ =& \frac{\exp[\psi H(Y_i) + \theta(H_{\min} - H(Y_i))]}{\sum_j \exp[\psi H(Y_j) + \theta(H_{\min} - H(Y_j))]}. \end{aligned}$$

By taking the limit we obtain

$$\lim_{\theta \to \infty} p_i^\theta = \begin{cases} \frac{1}{|I_{\min}|}, & \text{if } i \in I_{\min} \\ 0, & \text{else} \end{cases}$$

We can conclude that

$$\begin{aligned} \lim_{\theta \to \infty} \frac{\partial l_n(\theta; x)}{\partial \theta} &= |I_{\min}| \cdot H_{\min} \cdot \frac{1}{|I_{\min}|} - H(x) \\ &= H_{\min} - H(x) \\ &= \begin{cases} 0, & H(x) = H_{\min} \\ < 0, & H(x) > H_{\min} \end{cases}. \end{aligned} \tag{21}$$

In an analogous way we can see that

$$\begin{aligned} \lim_{\theta \to -\infty} \frac{\partial l_n(\theta; x)}{\partial \theta} &= H_{\max} - H(x) \\ &= \begin{cases} 0, & H(x) = H_{\max} \\ > 0, & H(x) < H_{\max} \end{cases}. \end{aligned} \tag{22}$$

From equation (20) we know that the first derivative of $l_n(\theta; x)$ is a strictly decreasing function. Suppose $H_{\min} < H(x) < H_{\max}$, from equations (21) and (22) we can conclude that the derivative of the log-likelihood function as exactly one root. It follows that $l_n(\theta; x)$ from equation (17) has exactly one maximum for $\theta \in \mathbb{R}$.

Now suppose that $H(x) = H_{\min}$ or $H(x) = H_{\max}$. It follows that

$$\frac{\partial l_n(\theta; x)}{\partial \theta} \geq 0 \text{ or } \frac{\partial l_n(\theta; x)}{\partial \theta} \leq 0$$

hold respectively and the function has no root for $\theta \in \mathbb{R}$. Therefore $l_n(\theta; x)$ has no maximum on $\mathbb{R}$. By negating this logical sentence it is found that if $l_n(\theta; x)$ has no maximum, then $H(x) = H_{\min}$ or $H(x) = H_{\max}$ holds. □

At this point it is proven that $l_n(\theta; x)$ has a maximum if and only if $H(x)$ is not an extreme value. Of course the next question is, when does it occur that $H(x) = H_{\max}$ or $H(x) = H_{\min}$? This will happen with a higher probability as the parameter θ gets bigger. In figure 4 this is shown in an intuitive way.

In this report we focus on values for θ close to 0. So the problem of no maximum in the log-likelihood won't occur very often, though it can still happen.

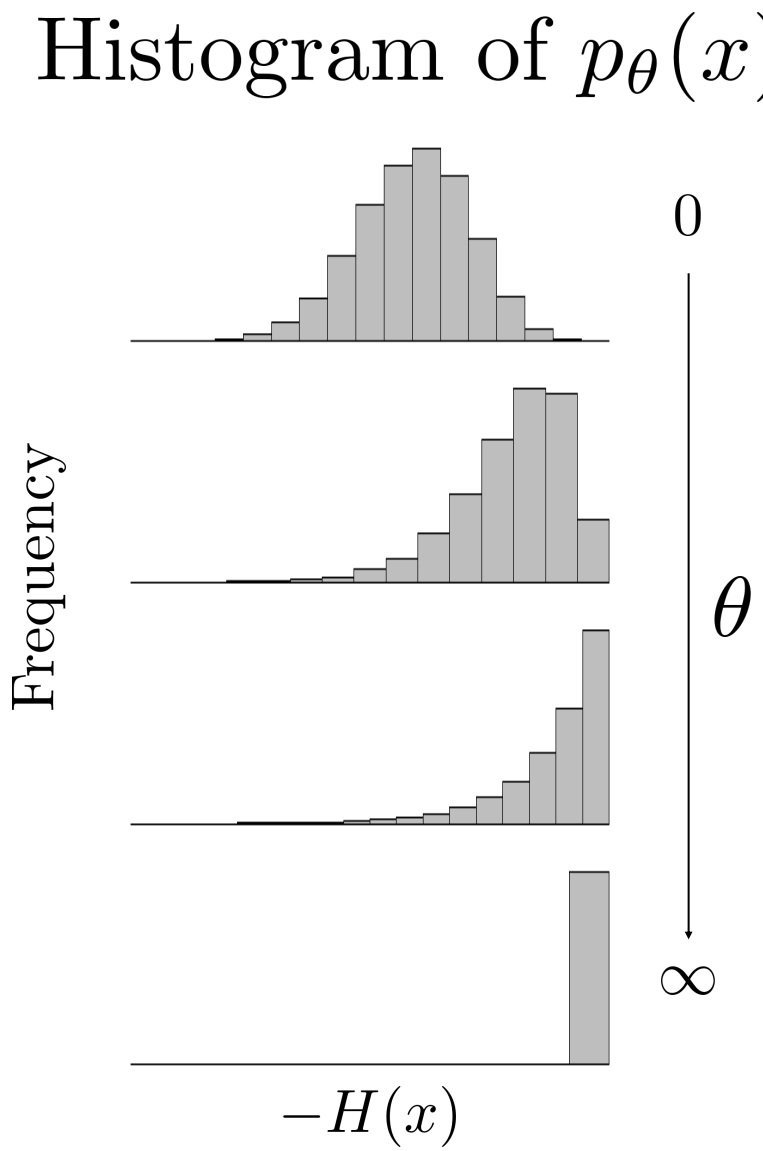

Figure 4: Four histograms of samples from $p_\theta(x)$. In the figure it can be seen that the bigger θ gets the higher is the probability of a configuration x with a high $-H(x)$ (or low energy).

3.2.2 Proofs for concavity of the log-likelihood functions

In the remaining part of this section the proofs are given concerning concavity and the maximum of both log-likelihood functions. With $l(\boldsymbol{\theta}; x)$ we refer to equation (11) and with $l_n(\boldsymbol{\theta}; x)$ to equation (15). $Y_1, \ldots, Y_n$ are still samples from distribution $p_{\boldsymbol{\psi}}$ and Y is a random variable with distribution $p_{\boldsymbol{\psi}}$. Furthermore $Y^{(n)}$ is the empirical random variable based on samples $Y_1, \ldots, Y_n$. This means that $Y^{(n)}$ is a discrete random variable with state space $S = \{Y_1, \ldots, Y_n\}$ and pmf

$$p_{Y^{(n)}}(x) = \frac{1}{n} \sum_{y \in S} \mathbb{1}_y(x).$$

Lemma 3.4. *The Hessian $\boldsymbol{H}_l$ of $l(\boldsymbol{\theta}; x)$ is equal to $-\mathrm{Cov}_{\boldsymbol{\theta}}(\boldsymbol{s}(Y))$.*

Proof. First of all remember that the probability distribution related to this likelihood function is

$$p_{\boldsymbol{\theta}}(y) = \frac{e^{\boldsymbol{\theta} \cdot \boldsymbol{s}(y)}}{\sum_{z \in \Omega} e^{\boldsymbol{\theta} \cdot \boldsymbol{s}(z)}}, \quad y \in \Omega.$$

In general, the expectation is given by

$$\mathbb{E}_{\boldsymbol{\theta}}[f(Y)] = \sum_{y \in \Omega} p_{\boldsymbol{\theta}}(y) \cdot f(y)$$

Now the gradient of $l(\boldsymbol{\theta}; x)$ is calculated. Suppose that $A_y = \exp[\boldsymbol{\theta} \cdot \boldsymbol{s}(y)]$. We take a look at the derivative with respect to the i'th element of parameter vector $\boldsymbol{\theta}$:

$$\frac{\partial l(\boldsymbol{\theta}; x)}{\partial \theta_i} = s_i(x) - \frac{\sum_{y \in \Omega} s_i(y) \cdot A_y}{\sum_{z \in \Omega} A_y}. \tag{23}$$

Remember that

$$\frac{\sum_{y \in \Omega} s_i(y) \cdot A_y}{\sum_{z \in \Omega} A_z} = \sum_{y \in \Omega} s_i(y) \frac{A_y}{\sum_{z \in \Omega} A_z} = \sum_{y \in \Omega} s_i(y) \cdot p_{\boldsymbol{\theta}}(y) = \mathbb{E}_{\boldsymbol{\theta}}[s_i(Y)].$$

Now we want to calculate the Hessian matrix, $\boldsymbol{H}_l$. This time the derivative of (23) with respect to θ_j is calculated. This gives us

$$\begin{aligned} \frac{\partial^2 l(\boldsymbol{\theta}; x)}{\partial \theta_i \partial \theta_j} &= -\left[\frac{\sum_{y \in \Omega} A_y \cdot \sum_{y \in \Omega} s_i(y) s_j(y) A_y - \left(\sum_{y \in \Omega} s_i(y) A_y\right) \cdot \left(\sum_{y \in \Omega} s_j(y) A_y\right)}{\left(\sum_{z \in \Omega} A_z\right)^2} \right] \\ &= \mathbb{E}_{\boldsymbol{\theta}}[s_i(Y)] \cdot \mathbb{E}_{\boldsymbol{\theta}}[s_j(Y)] - \mathbb{E}_{\boldsymbol{\theta}}[s_i(Y) \cdot s_j(Y)] = -\mathrm{Cov}_{\boldsymbol{\theta}}(s_i(Y),\ s_j(Y)). \end{aligned}$$

It follows that $\boldsymbol{H}_l = -\mathrm{Cov}_{\boldsymbol{\theta}}(\boldsymbol{s}(Y))$. □

Lemma 3.5. *The log-likelihood function, $l(\boldsymbol{\theta}; x)$, is a strictly concave function if $\mathrm{Cov}_{\boldsymbol{\theta}}(\boldsymbol{s}(Y))$ is positive definite for all $\boldsymbol{\theta}$.*

Proof. Suppose that $\mathrm{Cov}_{\boldsymbol{\theta}}(\boldsymbol{s}(Y))$ is positive definite for all $\boldsymbol{\theta}$. It follows that $-\boldsymbol{H}_l$ is positive definite and $-l(\boldsymbol{\theta};x)$ is a strictly convex function. This implies that $l(\boldsymbol{\theta};x)$ is a strictly concave function. □

Similar lemmas can be proven concerning $l_n(\boldsymbol{\theta};x)$.

Lemma 3.6. *The Hessian $\boldsymbol{H}_{l_n}$ of $l_n(\boldsymbol{\theta};x)$ is equal to $-\mathrm{Cov}_{\boldsymbol{\theta}-\boldsymbol{\psi}}(\boldsymbol{s}(Y^{(n)}))$.*

Proof. The proof is very analogous to that of lemma 3.4. Let

$$p_{\boldsymbol{\psi}}^{(n)}(Y_i) = \frac{\exp[\boldsymbol{\psi}\boldsymbol{s}(Y_i)]}{\sum_{j=1}^{n}\exp[\boldsymbol{\psi}\boldsymbol{s}(Y_j)]}$$

be the empirical probability distribution defined by samples $Y_1,\ldots,Y_n$.

First l_n is differentiated to one of its variables to find the gradient of the function. Suppose $A_i = \exp[(\boldsymbol{\theta}-\boldsymbol{\psi})\boldsymbol{s}(Y_i)]$. This gives us

$$\frac{\partial l_n(\boldsymbol{\theta};x)}{\partial\theta_k} = s_k(x) - \frac{\sum_{i=1}^{n} s_k(Y_i)\cdot A_i}{\sum_{j=1}^{n} A_j}. \tag{24}$$

Again, remember that

$$\begin{aligned}\frac{\sum_{i=1}^{n} s_k(Y_i)\cdot A_i}{\sum_{j=1}^{n} A_j} &= \sum_{i=1}^{n} s_k(Y_i)\frac{A_i}{\sum_{j=1}^{n} A_j}\\ &= \sum_{i=1}^{n} s_k(Y_i)\cdot p_{\boldsymbol{\theta}-\boldsymbol{\psi}}^{(n)}(Y_i) = \mathbb{E}_{\boldsymbol{\theta}-\boldsymbol{\psi}}[s_k(Y^{(n)})]\end{aligned}$$

By differentiating equation (24) with respect to θ_l we find

$$\begin{aligned}\frac{\partial^2 l_n(\boldsymbol{\theta};x)}{\partial\theta_k\partial\theta_l} &= -\left[\frac{\sum_{i=1}^{n} A_i\cdot\sum_{i=1}^{n} s_k(Y_i)A_i - (\sum_{i=1}^{n} s_k(Y_i)s_l(Y_i)A_i)\cdot(\sum_{i=1}^{n} s_l(Y_i)A_i)}{\left(\sum_{j=1}^{n} A_j\right)^2}\right]\\ &= \mathbb{E}_{\boldsymbol{\theta}-\boldsymbol{\psi}}[s_k(Y^{(n)})]\cdot\mathbb{E}_{\boldsymbol{\theta}-\boldsymbol{\psi}}[s_l(Y^{(n)})] - \mathbb{E}_{\boldsymbol{\theta}-\boldsymbol{\psi}}[s_k(Y^{(n)})\cdot s_l(Y^{(n)})]\\ &= -\mathrm{Cov}_{\boldsymbol{\theta}-\boldsymbol{\psi}}[s_k(Y^{(n)}),\ s_l(Y^{(n)})].\end{aligned}$$

In other words, the Hessian $\boldsymbol{H}_{l_n}$ of the Monte Carlo log-likelihood function is equal to $-\mathrm{Cov}_{\boldsymbol{\theta}-\boldsymbol{\psi}}(\boldsymbol{s}(Y^{(n)}))$. □

Lemma 3.7. *The Monte Carlo log-likelihood function, $l_n(\boldsymbol{\theta};x)$, is a strictly concave function if $\mathrm{Cov}_{\boldsymbol{\theta}-\boldsymbol{\psi}}(\boldsymbol{s}(Y^{(n)}))$ is positive definite for all $\boldsymbol{\theta}-\boldsymbol{\psi}$.*

Proof. Suppose that $\mathrm{Cov}_{\boldsymbol{\theta}-\boldsymbol{\psi}}(\boldsymbol{s}(Y^{(n)}))$ is positive definite for all $\boldsymbol{\theta}-\boldsymbol{\psi}$. It follows that $-\boldsymbol{H}_{l_n}$ is positive definite and $-l_n(\boldsymbol{\theta};x)$ is a strictly convex function. This implies that $l_n(\boldsymbol{\theta};x)$ is a strictly concave function. □

3.2.3 Proofs for the global maximum of the log-likelihood function

We know now that (under certain conditions) both the original log-likelihood function and the Monte Carlo log-likelihood function $l_n(\boldsymbol{\theta}; x)$ are strictly concave functions. Yet it is not clear whether both functions have a maximum on $\mathbb{R}^q$. It will be proven that, under certain other conditions, this is the case.

Definition 3.1. A function $f : \mathbb{R}^n \to \mathbb{R}$ is said to be coercive if for every sequence $\{\boldsymbol{x}_\nu\} \subset \mathbb{R}^n$ with $||\boldsymbol{x}_\nu|| \to \infty$ as $\nu \to \infty$, $f(\boldsymbol{x}_\nu) \to \infty$ as $\nu \to \infty$ holds.

Proposition 3.1. *The log-likelihood function, $l(\boldsymbol{\theta}; x)$, has exactly one maximum on $\mathbb{R}^q$ if for all $\tilde{\boldsymbol{\theta}} \in S^{q-1}$ with $||\tilde{\boldsymbol{\theta}}|| = 1$ there exists an $y_0 \in \Omega$ such that $\tilde{\boldsymbol{\theta}} \cdot \boldsymbol{s}(y_0) > \tilde{\boldsymbol{\theta}} \cdot \boldsymbol{s}(x)$ holds and if $\mathrm{Cov}_{\boldsymbol{\theta}}(\boldsymbol{s}(Y^{(n)})$ is positive definite for all $\boldsymbol{\theta}$.*

Proof. Suppose for all $\tilde{\boldsymbol{\theta}} \in S^{q-1}$ with $||\tilde{\boldsymbol{\theta}}|| = 1$ there exists an $y_0 \in \Omega$ such that $\tilde{\boldsymbol{\theta}} \cdot \boldsymbol{s}(y_0) > \tilde{\boldsymbol{\theta}} \cdot \boldsymbol{s}(x)$ holds. Let $||\tilde{\boldsymbol{\theta}}|| = 1$ and $\boldsymbol{\theta} = r\tilde{\boldsymbol{\theta}}$, $r > 0$. It follows that $l(\boldsymbol{\theta}; x)$ is given by

$$\begin{aligned} l(\boldsymbol{\theta}; x) &= \log \left\{ \frac{\exp[r\tilde{\boldsymbol{\theta}}\boldsymbol{s}(x)]}{\sum_{y \in \Omega} \exp[r\tilde{\boldsymbol{\theta}}\boldsymbol{s}(y)]} \right\} \\ &\leq \log \left\{ \frac{\exp[r\tilde{\boldsymbol{\theta}}\boldsymbol{s}(x)]}{\exp[r\tilde{\boldsymbol{\theta}}\boldsymbol{s}(y_0)]} \right\} \\ &= r\tilde{\boldsymbol{\theta}}(s(x) - s(y_0)) \to -\infty, \quad \text{as } r \to \infty. \end{aligned} \tag{25}$$

This implies that the function $-l$ is a coercive function.

Consider the superlevel sets $L_\alpha(l) = \{\boldsymbol{\theta} \in \mathbb{R}^q : l(\boldsymbol{\theta}; x) \geq \alpha, \alpha \in \mathbb{R}\}$. Because l is a continuous function $L_\alpha(l)$ are closed sets. $L_\alpha(l)$ are bounded since $-l$ is coercive. It follows that $L_\alpha(l)$ are compact sets. It is known that a continuous function on a compact set attains its extreme values on that set, therefore l on $L_\alpha(l)$ reaches its maximum on $L_\alpha(l)$. This maximum is global and unique as a consequence of the strict concavity of l. (See figure 5.)

□

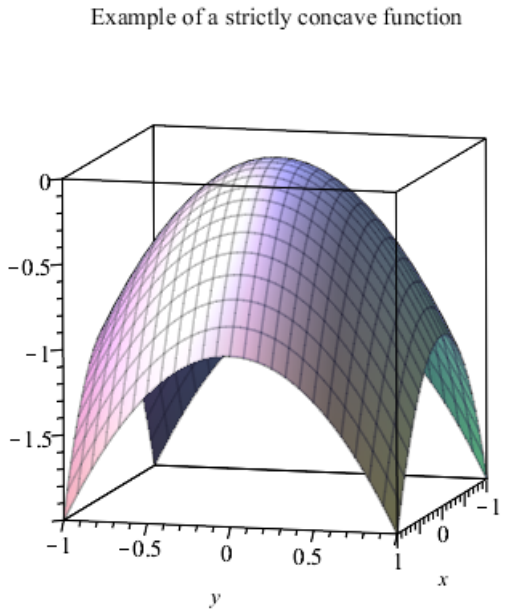

Figure 5: An example of a strictly concave function, $f(x, y)$. If $f(x, y) \to -\infty$ as $||[x\ y]^T|| \to \infty$ then it has a global maximum.

Proposition 3.2. *The Monte Carlo log-likelihood function, $l_n(\boldsymbol{\theta}; x)$, has exactly one maximum on $\mathbb{R}^q$ if for all $\tilde{\boldsymbol{\tau}} \in S^{q-1}$ with $||\tilde{\boldsymbol{\tau}}|| = 1$ there exists an $y_0 \in \{Y_1, \ldots, Y_n\}$ such that $\tilde{\boldsymbol{\tau}} \cdot \boldsymbol{s}(y_0) > \tilde{\boldsymbol{\tau}} \cdot \boldsymbol{s}(x)$ holds and if $\mathrm{Cov}_{\boldsymbol{\theta}-\boldsymbol{\psi}}(\boldsymbol{s}(Y^{(n)})$ is positive definite for all $\boldsymbol{\theta} - \boldsymbol{\psi}$.*

Proof. Let $\boldsymbol{\tau} = \boldsymbol{\theta} - \boldsymbol{\psi}$. Take $\tilde{\boldsymbol{\tau}} \in S^{q-1}$, $||\tilde{\boldsymbol{\tau}}|| = 1$. Suppose there exists a $y_0 \in \{Y_1, \ldots, Y_n\}$ such that $\tilde{\boldsymbol{\tau}}\boldsymbol{s}(y_0) > \tilde{\boldsymbol{\tau}}\boldsymbol{s}(x)$. Now $\boldsymbol{\theta} - \boldsymbol{\psi} = \boldsymbol{\tau} = r \cdot \tilde{\boldsymbol{\tau}}$ with $r > 0$. Thus l_n is given by

$$\begin{aligned} l_n(\boldsymbol{\theta}; x) &= r\tilde{\boldsymbol{\tau}}\boldsymbol{s}(x) - \log\left\{\frac{1}{n}\sum_{i=1}^{n}\exp[r\tilde{\boldsymbol{\tau}}\boldsymbol{s}(Y_i)]\right\} \\ &= r\tilde{\boldsymbol{\tau}}\boldsymbol{s}(y_0) + r[\tilde{\boldsymbol{\tau}}(\boldsymbol{s}(x) - \boldsymbol{s}(y_0))] \\ &\quad - \log\left\{\frac{1}{n}\sum_{i=1}^{n}\exp[r\tilde{\boldsymbol{\tau}}(\boldsymbol{s}(Y_i) - \boldsymbol{s}(y_0))]\right\} + r\tilde{\boldsymbol{\tau}}\boldsymbol{s}(y_0) \\ &= r[\tilde{\boldsymbol{\tau}}(\boldsymbol{s}(x) - \boldsymbol{s}(y_0))] - \log\left\{\frac{1}{n}\sum_{i=1}^{n}\exp[r\tilde{\boldsymbol{\tau}}(\boldsymbol{s}(Y_i) - \boldsymbol{s}(y_0))]\right\} \end{aligned}$$

It is easily checked that

$$\sum_{i=1}^{n}\exp[r\tilde{\boldsymbol{\tau}}(\boldsymbol{s}(Y_i) - \boldsymbol{s}(y_0))] \geq \sum_{\boldsymbol{s}(Y_i)\geq\boldsymbol{s}(y_0)}\exp[r\tilde{\boldsymbol{\tau}}(\boldsymbol{s}(Y_i) - \boldsymbol{s}(y_0))].$$

It follows that

$$\begin{aligned} &\log\left\{\frac{1}{n}\sum_{i=1}^{n}\exp[r\tilde{\boldsymbol{\tau}}(\boldsymbol{s}(Y_i) - \boldsymbol{s}(y_0))]\right\} \\ \leq &\log\left\{\sum_{i=1:\ \boldsymbol{s}(Y_i)\geq\boldsymbol{s}(y_0)}^{n}\exp[r\tilde{\boldsymbol{\tau}}(\boldsymbol{s}(Y_i) - \boldsymbol{s}(y_0))]\right\} \leq 0, \quad \text{as } r \to \infty \end{aligned}$$

and that

$$r[\tilde{\boldsymbol{\tau}}(\boldsymbol{s}(x) - \boldsymbol{s}(y_0))] \to -\infty \quad \text{as } r \to \infty.$$

In conclusion it is found that

$$l_n(\boldsymbol{\theta}; x) \to -\infty, \quad \text{as } r \to \infty.$$

Furthermore, by lemma 3.7 $l_n(\boldsymbol{\theta}; x)$ is a concave function. Following the same argumentation as in the proof of proposition 3.1 it can be concluded that $l_n(\boldsymbol{\theta}; x)$ has exactly one maximum. □

4 Results

In this section the results of the report are presented. At the end the result are used to state a conclusion of the precision of the model and of the method of MCMCMLE.

4.1 Software

The results are obtained by the Julia-package that is written to execute the method of this research. In the code-comments there is a detailed description of the package. In this way it is easier to follow the logical steps taken in the programs. The software can be found on software platform **GitHub**: See [Bos17a] and [Bos17b] or The Curie-Weiss model software and The ERGM software for direct hyperlinks. Make sure to read the README file first, in there it will be explained which file is used for what.

4.2 Curie-Weiss model

In the diagrams the response variable (that is $\hat{\theta}$) will always be on the vertical axis. For each data point in a diagram the package calculates a group of 30 outcomes of $\hat{\theta}$ behind the scenes. These outcomes are used to give the mean and standard deviation of the group so that the precision of the estimator can be measured.

The horizontal axis is labelled with different explanatory variables that can be found in table 1. Only one of them at a time is the explanatory variable. The others will be kept in place. In the next pages all results for the Curie-Weiss model will be given.

When one runs the package, there are two options. Either a observation x is chosen on forehand or a value for θ^* is given to generate an observation x making use of the Glauber dynamics for p_{θ^*}. This last option is more interesting because it will show whether the package gives back a $\hat{\theta}$ close to θ^* or not. On the other hand, when x is chosen the precision of the results will be higher.

The package consists of two parts. One is making use of the Glauber dynamics to generate the n samples and the second one samples directly from the probability distribution p_θ. (This is only possible when $d \leq 10$, otherwise p_θ will be too computational expensive.)

4.2.1 n vs $\hat{\theta}$

One of the things we are interested in is how precise our model is when we calculate $\hat{\theta}$ based on n outcomes of the Glauber dynamics and of the exact samples. The results can be found in figure 6.

It follows that the standard deviation gets smaller as n gets bigger when we look at the four top plots of the figure. The samples based on the exact distribution are a little more precise than those based on the Glauber Dynamics.

Variable name	Range when explanatory variable
d	[10, 20]
n	[50, 10000]
ψ	$[-\theta^*,\ \theta^*]$
Glauber steps	[10, 1000]

Table 1: Table of variables in the model that can be the explanatory variable in a diagram. Each time only one of the variables is the explanatory variable, the others are kept in place. θ^* is the value for θ that is used to generate the results.

We can conclude that a n of 1000 is high enough for a sufficient low standard deviation.

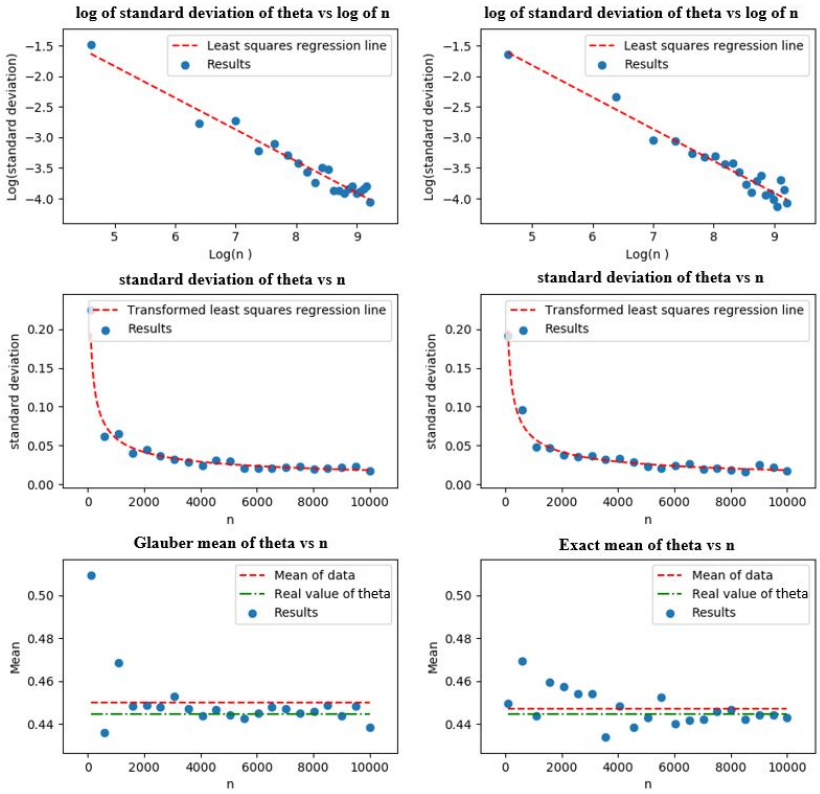

Figure 6: Plots of n vs variance and mean of sample group for θ. The samples are based on the Glauber dynamics (left side) and on the exact distribution (right side). The top figures are transformed to obtain a linear relation. Values of other variables are: $d = 10$, $\psi = 0.5$, *Glauber steps* $= 500$.

4.2.2 Glauber steps vs $\hat{\theta}$

The amount of *Glauber steps* in the program is important. If it is too low, the sample distribution will not be close to the original probability distribution and the mixing time of the Glauber dynamics is not respected. The corresponding diagrams can be found in figure 7 and 8. The results show that the SD of $\hat{\theta}$ decreases a *little* up to *Glauber steps* $= 200$. For a higher amount of steps almost no improvement can be found.

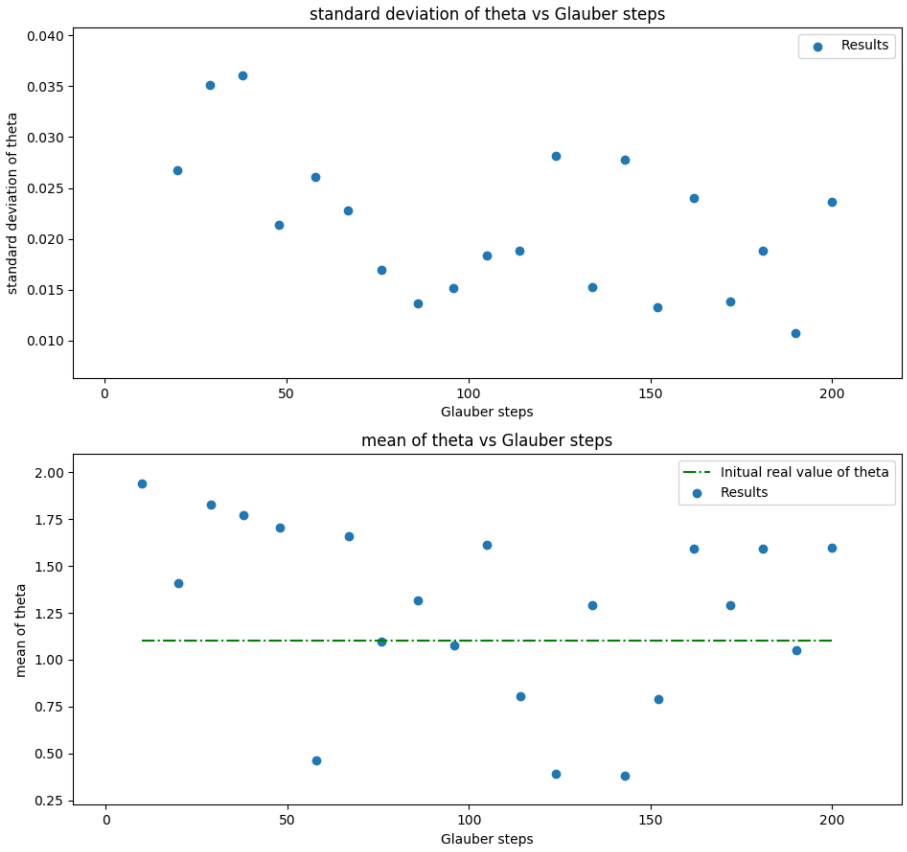

Figure 7: Plots of the amount of steps in the Glauber dynamics (low) against the standard deviation (above) and against the mean (below) of the group of 30 outcomes. Values of other variables are: $d = 15$, $n = 1000$, $\psi = 1.1$.

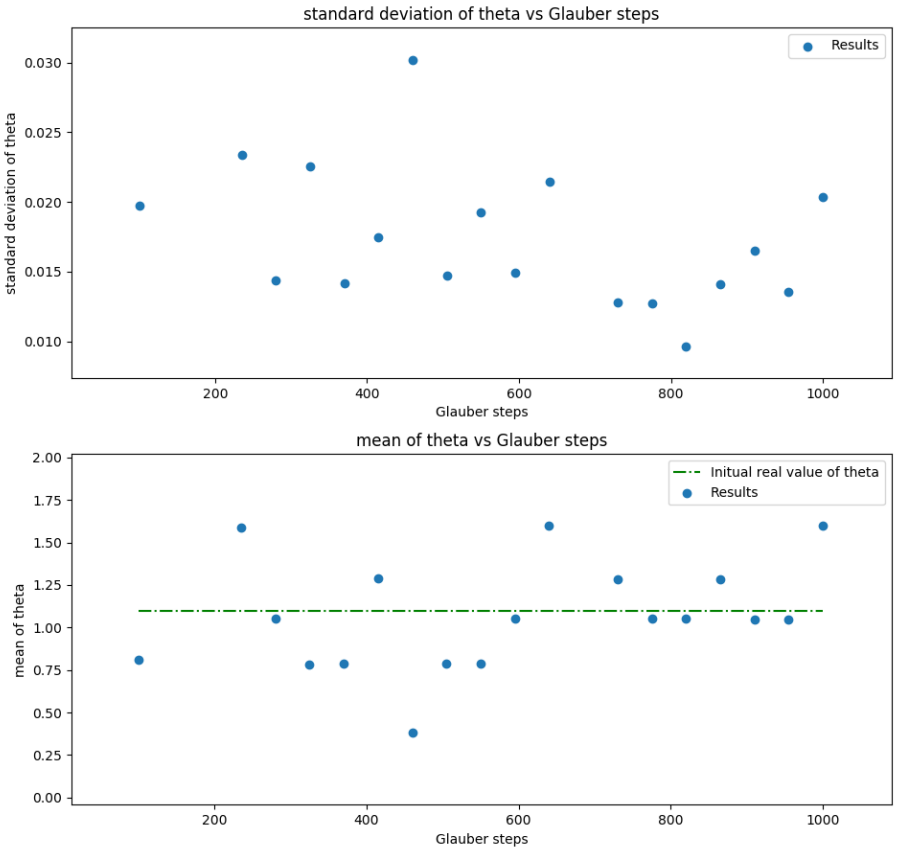

Figure 8: Plots of the amount of steps in the Glauber dynamics (high) against the standard deviation (above) and against the mean (below) of the group of 30 outcomes. Values of other variables are: $d = 15$, $n = 1000$, $\psi = 1.1$.

4.2.3 ψ vs $\hat{\theta}$

Now it is time to see what influence the value of ψ has on the outcomes of the program. ψ is the arbitrary parameter from equation 13 and 17.

In figure 9 the result is shown. In this diagram the observation x is chosen on forehand to increase precision.

It can be observed that the standard deviation of $\hat{\theta}$ is a its lowest point if $\psi = \theta$. This makes perfectly sense. The mean corresponding to this point, however is slightly lower than it should be.

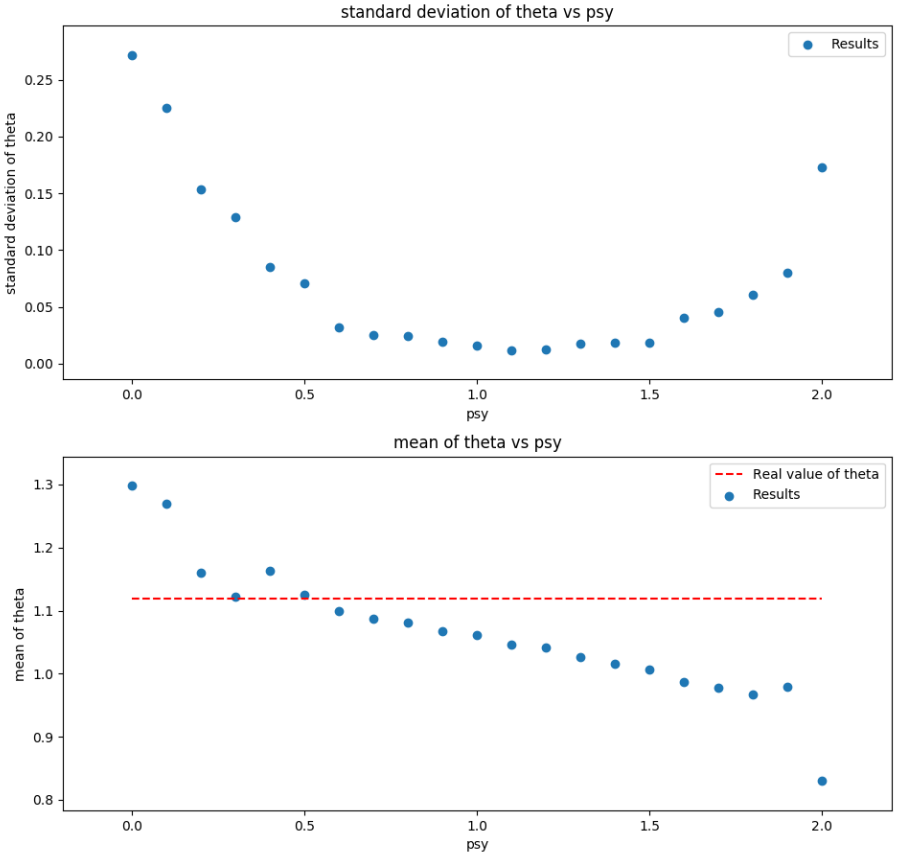

Figure 9: Plots of the parameter ψ against $\hat{\theta}$. An observation x was given on forehand (3 out of 15 kernels negative). The real value for theta is calculated afterwards to compare with the results. Values of other variables: $d = 15$, $n = 1000$, $Glauber\ steps = 100$.

4.3 ERGM

In the diagram corresponding to this model the horizontal and vertical axis are used for the parameters $\hat{\theta}_1$ and $\hat{\theta}_2$, respectively. The explanatory variable is given in the dimension of colours.

Just like in the Curie-Weiss Model, for each data point in a diagram a group of 30 outcomes is generated. In this way the mean and standard deviation can be found. Now the precision of the package can be measured. In each diagram the initial value of $\boldsymbol{\theta}$ is indicated by the black dotted lines.

4.3.1 n vs $\hat{\boldsymbol{\theta}}$

A few diagrams are given where the influence of n on the estimator can be visualised. (See figures 10 through 13.) In the diagrams the standard deviation of $\hat{\boldsymbol{\theta}}$ is higher with the rising of n. Likewise the higher n the closer the smaller the distance between the mean of $\hat{\boldsymbol{\theta}}$ and $\boldsymbol{\theta}$.

However, when the initial value of $\theta_2 < 0$ (That is, triangles are considered in the Glauber dynamics.), the standard deviation becomes significant higher and only drops a little when n gets bigger. Furthermore, the means of $\hat{\boldsymbol{\theta}}$ are scattered in the θ_1, θ_2 plane. Make note of the line that is formed through the initial value of $\boldsymbol{\theta}$.

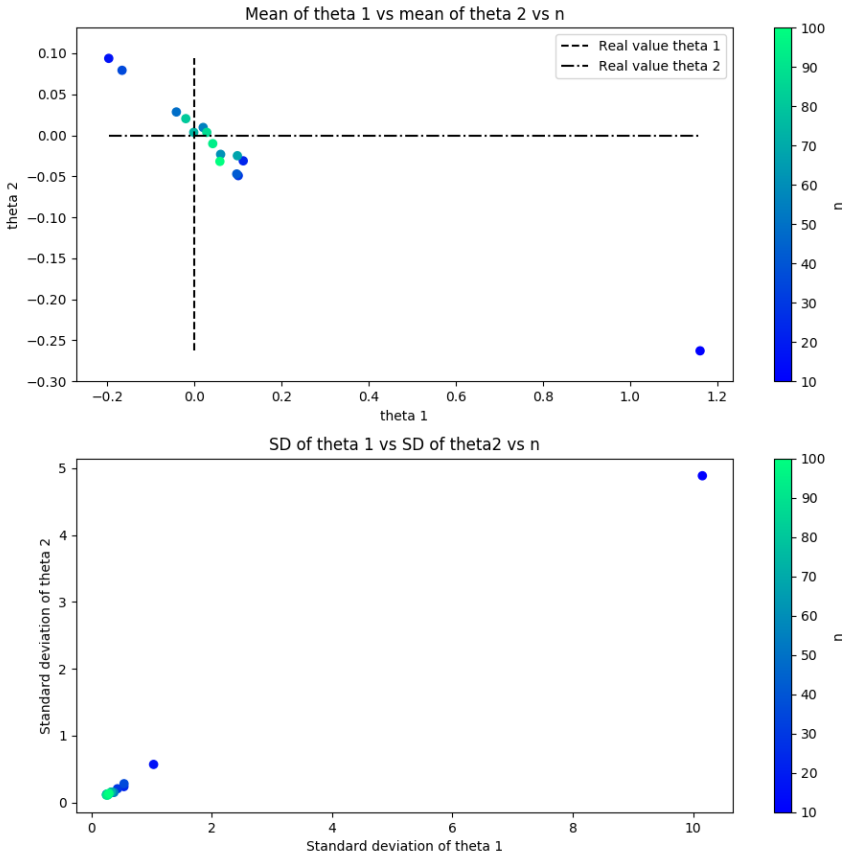

Figure 10: Diagram of θ_1 and θ_2 against n. Values for the other variables: *Glauber steps* $= 100$, $\boldsymbol{\psi} =$ real $\boldsymbol{\theta}$

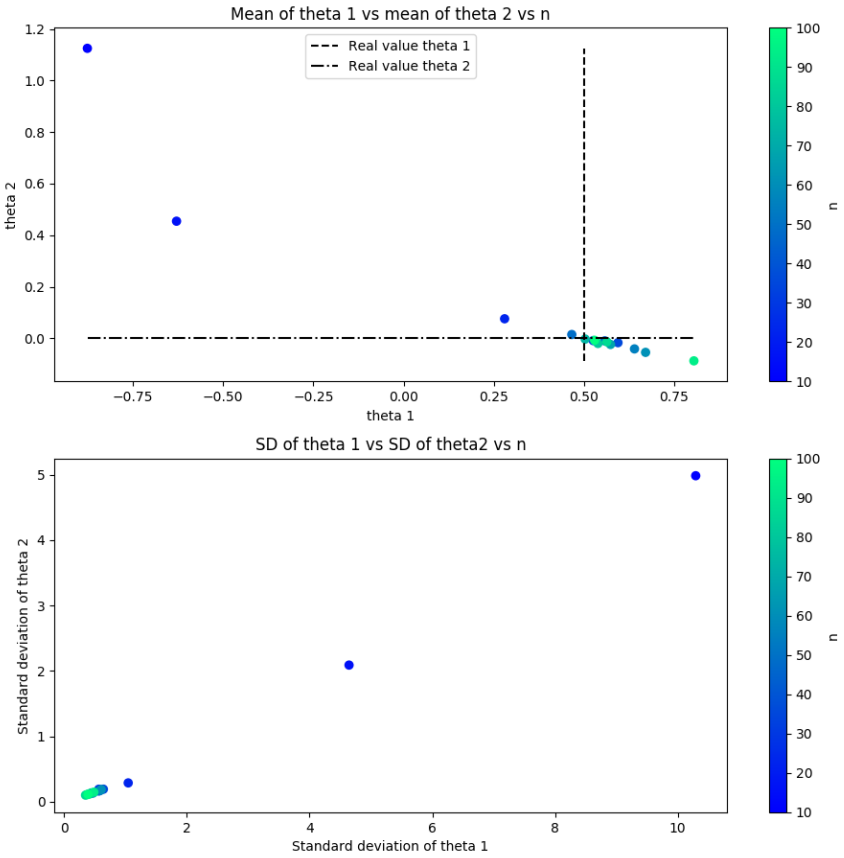

Figure 11: Diagram of θ_1 and θ_2 against n. Values for the other variables: *Glauber steps* $= 500$, $\boldsymbol{\psi} =$ real $\boldsymbol{\theta}$

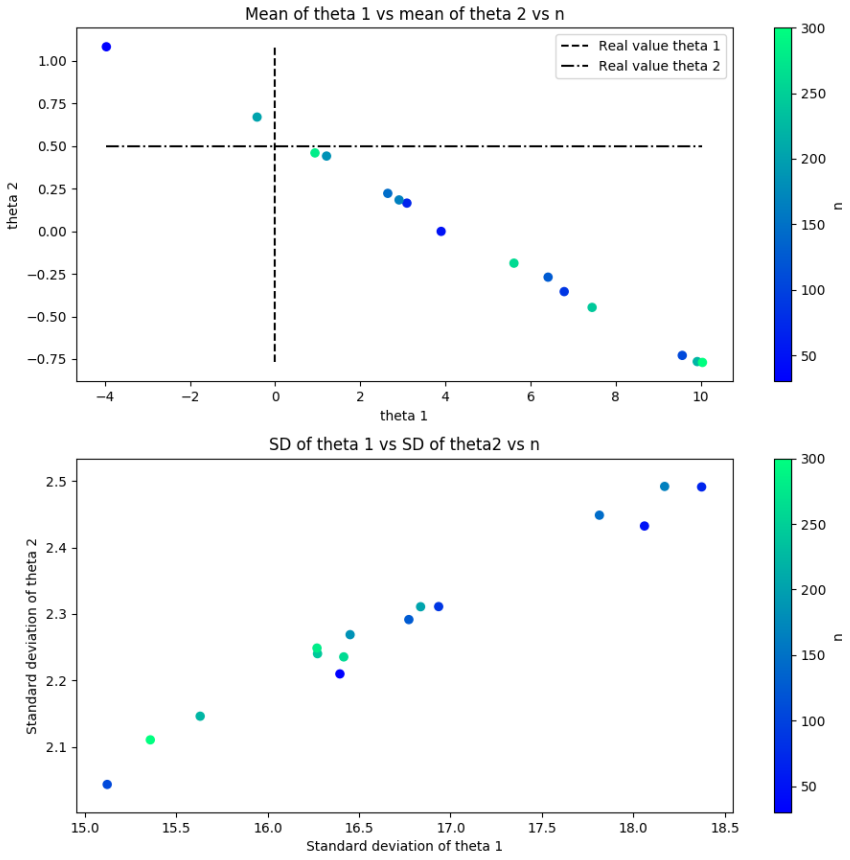

Figure 12: Diagram of θ_1 and θ_2 against n. Values for the other variables: *Glauber steps* $= 500$, $\boldsymbol{\psi} =$ real $\boldsymbol{\theta}$

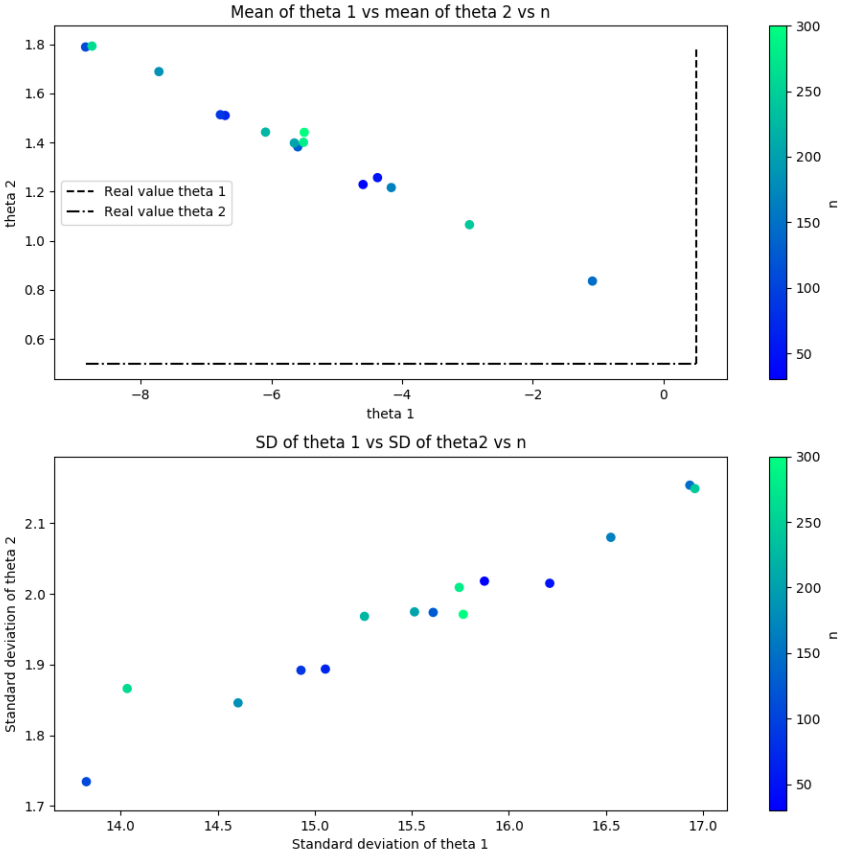

Figure 13: Diagram of θ_1 and θ_2 against n. Values for the other variables: *Glauber steps* $= 500$, $\boldsymbol{\psi} =$ real $\boldsymbol{\theta}$

4.3.2 *Glauber steps* vs $\hat{\boldsymbol{\theta}}$

The same is done for the amount of Glauber steps. The results can be found in figures 14 and 15. When the parameters are 0 and especially when $\theta_2 = 0$, a converging trend can be found in the diagram. When both parameters are nonzero it looks like the Markov chains do not converge.

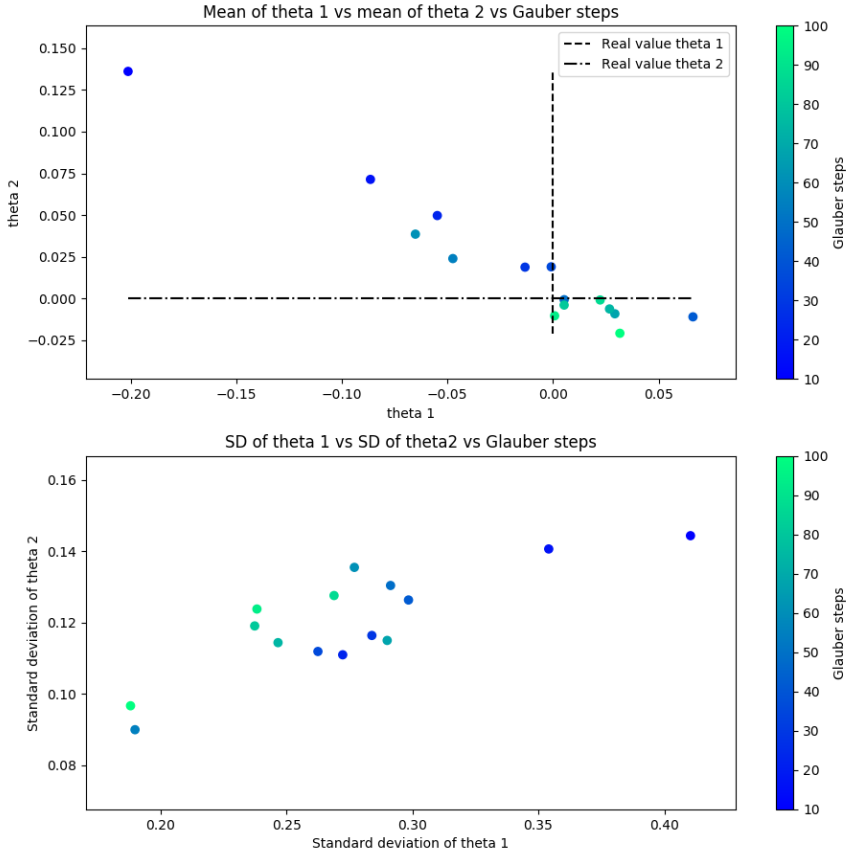

Figure 14: Diagram of θ_1 and θ_2 against *Glauber steps*. The initial values for $\boldsymbol{\theta}$ are given by the the dotted lines. Values for the other variables: $n = 500$, $\boldsymbol{\psi} =$ initial $\boldsymbol{\theta}$

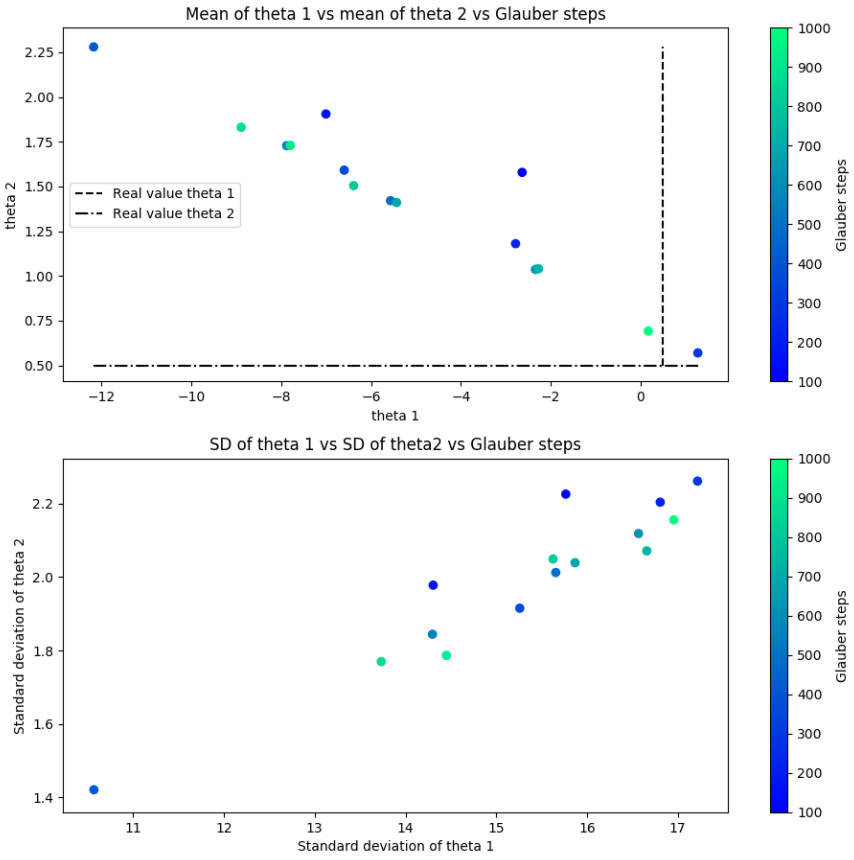

Figure 15: Diagram of θ_1 and θ_2 against *Glauber steps*. The initial values for $\boldsymbol{\theta}$ are given by the the dotted lines. Values for the other variables: $n = 500$, $\boldsymbol{\psi} = \text{initial } \boldsymbol{\theta}$

4.3.3 ψ against $\hat{\boldsymbol{\theta}}$

Finally, diagrams concerning the relation between ψ_1 and $\hat{\boldsymbol{\theta}}$, and ψ_2 and $\hat{\boldsymbol{\theta}}$ are given in figures 16 and 17. In the first diagram $\hat{\boldsymbol{\theta}}$ closest to the real value of $\boldsymbol{\theta}$ when $\psi_1 = \theta_1$, exactly like expected. Though in the second figure something happens that is more difficult to interpret.

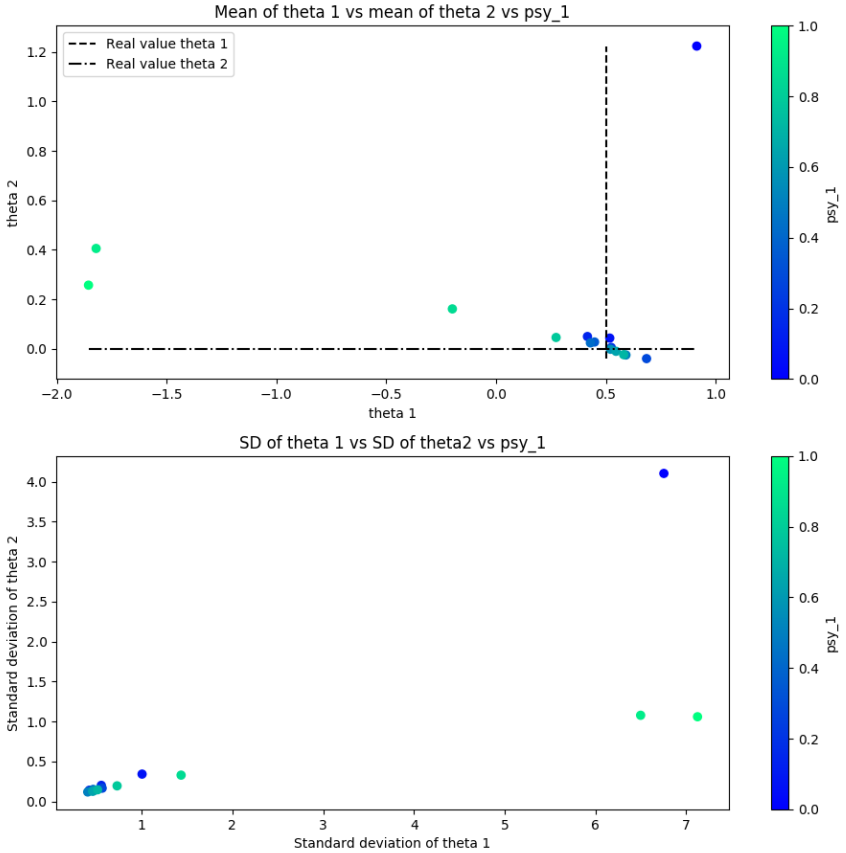

Figure 16: Plot of θ_1 and θ_2 against ψ_1

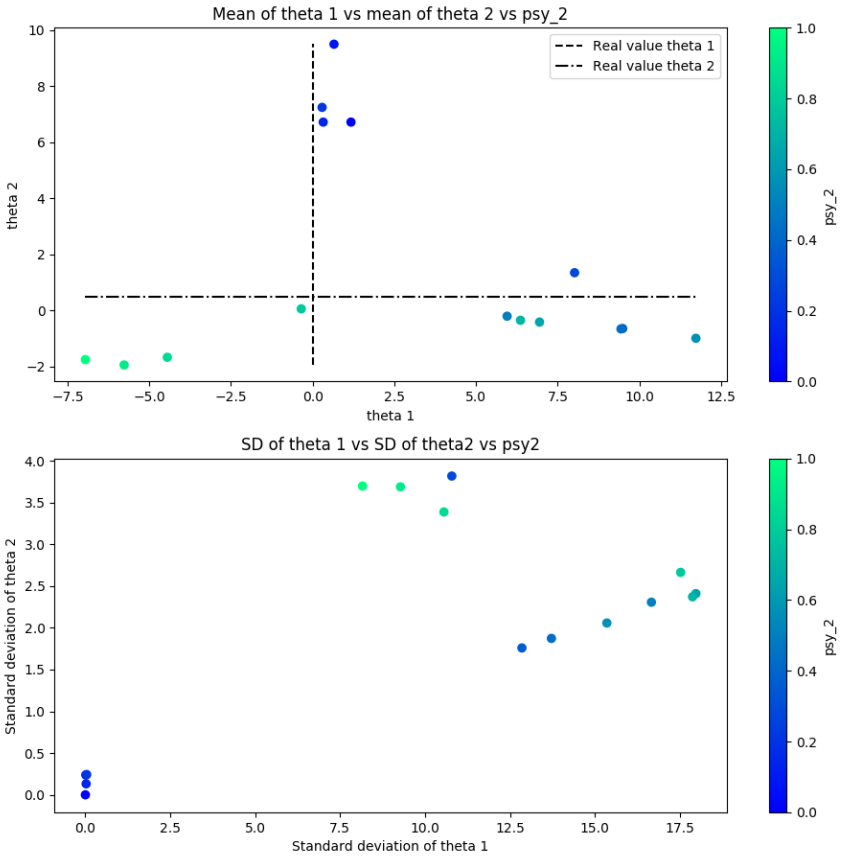

Figure 17: Plot of θ_1 and θ_2 against ψ_2

5 Discussion

In this report a self-written package that executes the method of MCMCMLE for two different models was examined. The package that executes the method for the Curie-Weiss model finds a most likely value for θ that is quite close to the real value. The standard deviation (the precision) and mean of the estimations against different variables in the model can be found in chapter 4. Though the results suggest quite a high precision, there is still some randomness in the model. Firstly, there are no conclusions that can be drawn from the results concerning the number of Glauber steps against the precision. Secondly, the diagrams show that the estimation is not optimal when $\psi = \theta$ while this is expected.

In the results of the package for the ERGM a clear convergence of the estimations of $\boldsymbol{\theta}$ is found as n rises and as soon as θ_2 remains (close to) zero. The same convergence appears as the number of Glauber steps rises. The estimation is at its best if $\psi_1 = \theta_1$. When $\theta_2 \neq 0$ it seems that no convergence of $\hat{\boldsymbol{\theta}}$ to $\boldsymbol{\theta}$ is found as n or *Glauber step* rises. The outcomes of the program do form a line through $\boldsymbol{\theta}$. This may be explained by the fact that the amount of edges and the amount of triangles in a network are obviously correlated. As it happens this implies that the number of combinations of θ_1 and θ_2 that can correspond to a network is bigger than one.

Finally it can be stated that I, as the writer of this report, have learned a lot. All the information in the area of mathematics, coding and researching that was treated has been added to my own knowledge and experience. It has inspired me to continue with this subject and to look for applications of networks that I am interested in.

Suggestions for future research In future research it is suggested that more information is found regarding the effect of the amount of Glauber steps on the precision of the method. In other words, it should be made clear what is the exact mixing time of the Glauber dynamics of both models. Secondly it should be investigated what is the probability of a non-estimable likelihood of θ. (That is, problems with a likelihood function without a maximum.) A possible source for this problem is found in [RPF11]. Furthermore it is found that the estimations of the parameters are not converging when triangles are considered in the ERGM. Yet it is not proven whether this is known problem in the world of ERGM and other network models or that it is a consequence of an error in the code of the Julia-package. It is suggested to treat this subject in the future as well. Besides that it is suggested to compare the package that execute the method in this research to other yet existing packages for the method of MCMCMLE. In this way it can be verified whether there are any errors left the package. Finally in the future it is a possibility to extend the research and the current package for directed networks and/or networks with connections that variate in strength. (Like our brain.)

Conclusion It can be concluded that the package that was written is quite precise for the Curie-Weiss model and that the results are in line with the theoretical expectation. The package that was written for the ERGM gives some good results as well, but something is not right when the parameter corresponding to the triangles in a network is unequal to zero. Whether this is due to an error in the package or an external factor is not clear and more research will have to be done. In general the speed of convergence of $\hat{\boldsymbol{\theta}}$ against n can be found in this report in an empirical way.

References

[BKSE12] Jeff Bezanson, Stefan Karpinski, Viral B Shah, and Alan Edelman. Julia: A fast dynamic language for technical computing. *arXiv preprint arXiv:1209.5145*, 2012.

[Bos17a] Douwe P Bosma. Curie-weiss-bep. *Available at* https://github.com/DPBosma/Curie-Weiss-BEP, 2017.

[Bos17b] Douwe P Bosma. Ergm-bep. *Available at* https://github.com/DPBosma/ERGM-BEP, 2017.

[DLP09] Jian Ding, Eyal Lubetzky, and Yuval Peres. The mixing time evolution of glauber dynamics for the mean-field ising model. *Communications in Mathematical Physics*, 289(2):725–764, 2009.

[Gey94] Charles J Geyer. On the convergence of monte carlo maximum likelihood calculations. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 261–274, 1994.

[LPW09] David Asher Levin, Yuval Peres, and Elizabeth Lee Wilmer. *Markov chains and mixing times*. American Mathematical Soc., 2009.

[Nor98] James R Norris. *Markov chains*. Number 2. Cambridge university press, 1998.

[RPF11] Alessandro Rinaldo, Sonja Petrovic, and Stephen E Fienberg. Maximum likelihood estimation in network models. *Arxiv preprint*, 2011.

[RPKL07] Garry Robins, Pip Pattison, Yuval Kalish, and Dean Lusher. An introduction to exponential random graph (p*) models for social networks. *Social networks*, 29(2):173–191, 2007.

[SHL11] Sean L Simpson, Satoru Hayasaka, and Paul J Laurienti. Exponential random graph modeling for complex brain networks. *PloS one*, 6(5):e20039, 2011.

[SHM+16] David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. Mastering the game of go with deep neural networks and tree search. *Nature*, 529(7587):484–489, 2016.