



UNIVERSITY OF CALIFORNIA, LOS ANGELES  
DEPARTMENT OF STATISTICS

---

# Potts Models

---

MULTIPLE IMPUTATION OF MISSING VALUES

*Author:*

Thomas MAIERHOFER  
SID 905078249

*Stats 202C:*

Prof. Mark Handcock

Final Project Report

Spring Quarter 2019,  
June 14, 2019

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Potts Model</b>	<b>2</b>
2.1	Temperature Estimation . . . . .	2
2.1.1	Maximum Pseudolikelihood Estimator . . . . .	3
2.1.2	MCMC Estimator . . . . .	4
2.2	Sampling Potts Models . . . . .	5
2.2.1	Gibbs Sampler . . . . .	5
2.2.2	Swendsen Wang Algorithm . . . . .	6
2.3	Treatment of Missing Values . . . . .	7
2.3.1	Missing Values in Temperature Estimation . . . . .	9
2.3.2	Missing Values in Model Sampling . . . . .	9
<b>3</b>	<b>Future Research</b>	<b>11</b>

## **Abstract**

Potts models are a well studied subject of statistical mechanics. This introduces a novel Markov Chain Monte Carlo approach for estimating the joint distribution of missing values in a Potts model. The Swendsen Wang algorithm is used to increase computational efficiency.

# 1 Introduction

The Potts model describes a system of interacting spins on a crystalline lattice. Its predecessor, the Ising model was originally described in Lenz (1920) to model atomic spins and was solved for the 2 dimensional case by Onsager (1944). In its generalization for more than two classes, the Potts model is often interpreted as an image, where every point in the lattice is a pixel taking one color in a discrete set. Usually, all pixels are assumed to be fully observed, which is not necessarily the case in non-experimental settings. This project proposes a method for estimating the joint distribution of missing pixels in a Potts model of unknown temperature which is based on a Markov Chain Monte Carlo (MCMC) algorithm.

The remainder of this report is structured as follows: Section 2 formally introduces the Potts model, outlines strategies for estimating the inverse temperature and based on this, sampling from a completely observed Potts models given the inverse temperature. Subsequently, a new method is proposed for estimating the joint distribution of missing values in a Potts model using multiple imputation. The report concludes in an outlook on unanswered questions for future research (Section 3).

## 2 Potts Model

The Potts model is a direct generalization of the Ising model where the class labels (or spin directions) take  $L \in \{2, 3, 4, \dots\}$  possible values. See topright and bottomleft of Figure 1 for an example of a four class Potts model in equilibrium. The probability of a given  $N \times N$  dimensional lattice  $X = (X)_{ij}$ ,  $i, j = 1, \dots, N$ , taking values  $x = (x)_{ij} \in \mathfrak{X} := \{1, 2, \dots, L\}^{N \times N}$ , is given as

$$P(X = x|\beta) = \frac{1}{C(\beta)} \exp \left( -\beta \sum_{i=1}^N \sum_{j=1}^N \sum_{(k,l) \in E_{ij}} \mathbb{1}\{x_{ij} = x_{kl}\} \right), \quad (1)$$

where  $\beta = 1/(kT)$  is the inverse temperature  $T$  with Boltzmann constant  $k$ ,  $C(\beta)$  is the integration constant ensuring that  $P(X|\beta)$  is a proper probability distribution

$$C(\beta) = \sum_{x \in \mathfrak{X}} \exp \left( -\beta \sum_{i=1}^N \sum_{j=1}^N \sum_{(k,l) \in E_{ij}} \mathbb{1}\{x_{ij} = x_{kl}\} \right), \quad (2)$$

$E_{ij}$  is the set of all indices sharing an edge with  $X_{ij}$ , i.e. the indices  $\{(i+1, j), (i-1, j), (i, j+1), (i, j-1)\}$ , and  $\mathbb{1}\{\cdot\} \in \{0, 1\}$  the indicator function which is 1 if  $\cdot$  is true. Note that for notational simplicity the lattice is assumed to be a torus, i.e. the pixels  $X_{1j}$  share an edge with  $X_{Nj}$  and the pixels  $X_{i1}$  share an edge with  $X_{iN}$  for all  $i, j$ , and  $\mathfrak{X}$  is the set of all possible colorings  $\{1, 2, \dots, L\}^{N \times N}$  of size  $|\mathfrak{X}| = L^{N^2}$ .

The remainder of this section discusses two complementary approaches to the estimation of the inverse temperature  $\beta$  in a 2D Potts model in Section 2.1, a pseudolikelihood estimator in Section 2.1.1 and an MCMC estimator in Section 2.1.2. Section 2.2 introduces a Gibbs Sampler (Section 2.2.1) and the Swendsen Wang algorithm (Section 2.2.2) for sampling new observations from a Potts model. Section 2.3 discusses how to use these algorithms to estimate the temperature (Section 2.3.1) and sample over missing observations from a partially observed grid (Section 2.3.2).

### 2.1 Temperature Estimation

An important problem arising in Potts models is the estimation of the inverse temperature  $\beta$ . As there are no feasible analytical results, a maximum pseudolikelihood estimator (MPLE) (Section 2.1.1), see e.g. Levada et al. (2009), and a MCMC estimator (Section 2.1.2) are used. The MPLE estimator is computationally way more efficient (scaling with  $N^2 \cdot L$ ) than the MCMC algorithm (scaling with  $M \cdot N^2 \cdot L$ , where  $M$  is the number of MCMC samples drawn) but can be biased, particularly for large  $\beta$ . The MCMC algorithm has the disadvantage that it requires a prespecified starting value  $\beta_0^{\text{MCMC}}$ , which needs to be close to the true  $\beta$  for the estimator to converge within a practical number of MCMC iterations  $M$ . The strengths of these two approaches can be combined, by using the MPLE estimator to obtain an initial starting value for the MCMC algorithm,

$$\beta_0^{\text{MCMC}} \leftarrow \hat{\beta}^{\text{MPLE}},$$

which allows the MCMC algorithm to converge within a reasonable time.

### 2.1.1 Maximum Pseudolikelihood Estimator

The likelihood function for  $X$  does not be factored into likelihood functions for the individual  $X_{ij}$ , as the distribution of  $X_{ij}$  depends on its neighbors  $X_{lk}, (l, k) \in E_{ij}$ . Ignoring this dependence, a pseudolikelihood estimator can be obtained, which approximates the true likelihood

$$\begin{aligned}
\mathcal{L}(\beta|X) &= P(X = x|\beta) \\
&= \frac{1}{C(\beta)} \exp \left( -\beta \sum_{i=1}^N \sum_{j=1}^N \sum_{(k,l) \in E_{ij}} \mathbb{1}\{x_{ij} = x_{kl}\} \right) \\
&= \prod_{i=1}^N \prod_{j=1}^N \frac{1}{C(\beta)} \exp \left( -\beta \sum_{(k,l) \in E_{ij}} \mathbb{1}\{x_{ij} = x_{kl}\} \right) \tag{3}
\end{aligned}$$

with integration constant  $C(\beta)$  given in Equation (2), by the pseudolikelihood

$$\begin{aligned}
\mathcal{L}^*(\beta|X) &= \prod_{i=1}^N \prod_{j=1}^N P(X_{ij} = x_{ij}|\beta, X \setminus X_{ij}) \\
&= \prod_{i,j} P(X_{ij} = x_{ij}|\beta, X_{kl}), \text{ given all } (k, l) \in E_{ij} \\
&= \prod_{i,j} \frac{1}{C_{ij}(\beta)} \exp \left( -\beta \sum_{(k,l) \in E_{ij}} \mathbb{1}\{x_{ij} = x_{kl}\} \right), \tag{4}
\end{aligned}$$

with integration constant

$$C_{ij}(\beta) = \sum_{y=1}^L \exp \left( -\beta \sum_{(k,l) \in E_{ij}} \mathbb{1}\{x_{kl} = y\} \right). \tag{5}$$

The pseudolikelihood estimator is based on the (incorrect) assumption that the neighborhood of every  $X_{ij}$  is given and fixed. Additionally to allowing the factorization of the likelihood, the integration constant is simplified dramatically. Instead of summing over all  $L^{N^2}$  possible lattices  $x \in \{1, \dots, L\}^{N \times N}$ , we only have to sum over all  $LN^2$  possible values  $x_{ij} \in \{1, \dots, L\}$  given their fixed neighborhood  $E_{ij}$ . The maximum pseudolikelihood estimator is obtained maximizing the log-pseudo-

likelihood w.r.t.  $\beta$

$$\begin{aligned}
\ell^*(\beta|X) &= \log \left( \mathcal{L}^*(\beta|X) \right) \\
&= \sum_{i=1}^N \sum_{j=1}^N \log \left( \frac{1}{C_{ij}(\beta)} \exp \left( -\beta \sum_{(k,l) \in E_{ij}} \mathbb{1}\{x_{ij} = x_{kl}\} \right) \right) \\
&= \sum_{i,j} -\log \left( \sum_{y=1}^L \exp \left( -\beta \sum_{(k,l) \in E_{ij}} \mathbb{1}\{x_{kl} = y\} \right) \right) \\
&\quad -\beta \sum_{i,j} \sum_{(k,l) \in E_{ij}} \mathbb{1}\{x_{ij} = x_{kl}\} \\
&=: -\sum_{i,j} \log \left( \sum_{y=1}^L \exp \left( -\beta U_{ij}(y) \right) \right) - \beta \sum_{i,j} U_{ij}(x_{ij}), \tag{6}
\end{aligned}$$

introducing the notational abbreviation

$$U_{ij}(y) = \sum_{(k,l) \in E_{ij}} \mathbb{1}\{x_{kl} = y\}$$

for counts of class  $y$  in the neighborhood of  $X_{ij}$ . Maximization of the pseudolikelihood is achieved by equating its derivative w.r.t.  $\beta$  to 0,

$$\begin{aligned}
\frac{d}{d\beta} \ell^*(\beta|X) &= -\sum_{i,j} \frac{\sum_{y=1}^L \exp(-\beta U_{ij}(y)) (-U_{ij}(y))}{\sum_{y=1}^L \exp(-\beta U_{ij}(y))} - \sum_{i,j} U_{ij}(x_{ij}) \\
&\stackrel{!}{=} 0 \\
\Rightarrow \sum_{i,j} U_{ij}(x_{ij}) &= \sum_{i,j} \frac{\sum_{y=1}^L \exp(-\beta U_{ij}(y)) U_{ij}(y)}{\sum_{y=1}^L \exp(-\beta U_{ij}(y))},
\end{aligned}$$

which is solved by the maximum pseudolikelihood estimator

$$\hat{\beta}^{\text{MPLE}} = \arg \min_{\beta} \left| \sum_{i,j} U_{ij}(x_{ij}) - \sum_{i,j} \frac{\sum_{y=1}^L \exp(-\beta U_{ij}(y)) U_{ij}(y)}{\sum_{y=1}^L \exp(-\beta U_{ij}(y))} \right| \tag{7}$$

which can be solved using a simple grid search over possible  $\beta$  (or Newton-Raphson).

### 2.1.2 MCMC Estimator

The MCMC estimator obtains a more accurate estimate of the likelihood by approximating the integration constant  $C(\beta)$  using MCMC sampling. Relying on the approximation  $\mathfrak{X} \approx \{x_1, x_2, \dots, x_M\}$  for large  $M$  and arbitrary  $\beta_0$ , we can sample  $\{x_1, x_2, \dots, x_M\}$  using either sampling strategy described in Section 2.2 with inverse

temperature  $\beta_0$ . We estimate  $C(\beta)$ , see Equation (2), by its MCMC counterpart

$$\begin{aligned}
\hat{C}(\beta) &\propto \sum_{x \in \{x_1, \dots, x_M\}} \frac{1}{P(x|\beta_0)} \exp \left( -\beta \sum_{i=1}^N \sum_{j=1}^N \sum_{(k,l) \in E_{ij}} \mathbb{1}\{x_{ij} = x_{kl}\} \right) \\
&= \sum_{x \in \{x_1, \dots, x_M\}} C(\beta_0) \frac{\exp \left( -\beta \sum_{i,j} U_{ij}(x_{ij}) \right)}{\exp \left( -\beta_0 \sum_{i,j} U_{ij}(x_{ij}) \right)} \\
&\propto \sum_{x \in \{x_1, \dots, x_M\}} \exp \left( -(\beta - \beta_0) \sum_{i,j} U_{ij}(x_{ij}) \right), \tag{8}
\end{aligned}$$

which can be evaluated for any  $\beta$ . Note that for  $\beta \not\approx \beta_0$  the estimator will diverge unless the MCMC sample size  $M$  is very large.

## 2.2 Sampling Potts Models

This section describes two approaches for sampling observations from an  $L$  class Potts model on an  $N \times N$  dimensional lattice. Due to the interrelatedness of neighboring pixels, and thus all pixels, the computation of independent observations is intractable and iterative methods have to be used.

### 2.2.1 Gibbs Sampler

An iterative Gibbs sampler (Geman and Geman; 1987) can be used to pass through the sampling space. The core idea is, to update a single random pixel probabilistically according to its neighbors and a fixed inverse temperature  $\beta$ . Over time, no matter the initial state of the grid, this process will converge to (not independent) draws from the sampling space of the Potts model implied by  $\beta$ . In pseudo code, the algorithm to obtain  $M$  samples from the Potts model with inverse temperature  $\beta$  can be expressed as:



### Potts Model Sampling

1. Initialize  $X^0$  randomly in  $\{1, 2, \dots, L\}^{N \times N}$
2. For  $b$  in  $1, 2, \dots, B$  (burn-in):
  - 2.1 Sample random pixel  $X_{ij}$  from  $X^{b-1}$  with label  $x_{ij}$
  - 2.2 Sample new label  $y$  randomly from  $\{1, \dots, L\}$
  - 2.3 Compute energy difference  $\Delta E := U_{ij}(x_{ij}) - U_{ij}(y)$  between current lattice  $X^{b-1}$  and proposed lattice  $\tilde{X}^b = X^{b-1} : X_{ij} = y$
  - 2.4 Set  $X^b = \tilde{X}^b$  with probability  $\min(\exp(-\beta \Delta E), 1)$
3. For  $m$  in  $B + 1, B + 2, \dots, B + M$  (sampling from distribution):
  - 3.1 Sample random pixel  $X_{ij}$  from  $X^{m-1}$  with label  $x_{ij}$
  - 3.2 Sample new label  $y$  randomly from  $\{1, \dots, L\}$
  - 3.3 Compute energy difference  $\Delta E := U_{ij}(x_{ij}) - U_{ij}(y)$  between current lattice  $X^{m-1}$  and proposed lattice  $\tilde{X}^m = X^{m-1} : X_{ij} = y$
  - 3.4 Set  $X^m = \tilde{X}^m$  with probability  $\min(\exp(-\beta \Delta E), 1)$
4. Return  $\{X^{B+1}, X^{B+2}, \dots, X^{B+M}\}$

A GIF of the Gibbs sampler sampling new observations from a Potts model in equilibrium can be found following this [link](#).

### 2.2.2 Swendsen Wang Algorithm

The Swendsen Wang Algorithm (SW) is a generalization of the model sampling presented above (Section 2.2.1). Instead of proposing individual pixels  $X_{ij}$  to be flipped with probability less than one, SW creates probabilistic clusters from which one will be selected to be flipped in its entirety. SW for a given lattice  $X$  and inverse temperature  $\beta$  is structured the same way as the Potts model sampling in Section 2.2.1, but instead of updating individual pixels in Steps 2.1.-2.4. and 3.1.-3.4., entire clusters of pixels are updated using the following algorithm:

### Swendsen Wang Algorithm

1. Clustering step
  - 1.1 Sample random edge set  $E^*$  that contains an edge between to neighbors  $X_{ij}, X_{lk}, (l, k) \in E_{ij}$  with probability  $1 - \exp(-\beta)$  if  $x_{ij} = x_{lk}$
  - 1.2 Define clusters as all pixels that are connected in  $E^*$
2. Flipping step
  - 2.1 Sample new label  $y$  randomly from  $\{1, \dots, L\}$
  - 2.2 Sample random pixel  $X_{ij}$  from  $X$
  - 2.3 Assign new label  $y$  to all pixels in the cluster containing  $X_{ij}$
3. Return  $X$

This can increase computational efficiency, especially for large  $\beta$ , as a Potts model in equilibrium with a high  $\beta$  has a very small flipping probability for any individual pixel. SW avoids this problem by creating increasingly large clusters for increasing  $\beta$ . This means, that for large  $\beta$  the flipping probability is still one, but the probability of neighboring pixels with the same color being assigned different colors in an update step converges to 0 for large  $\beta$ . Figure 1 shows one update step from topright to bottomleft, where most of the yellow cluster in the center is flipped to green. From topleft to topright shows how a large number of SW updates move a randomly initialized grid into its equilibrium given inverse temperature  $\beta$ . A GIF of SW sampling new observations in a Potts model in equilibrium can be found following this [link](#).

## 2.3 Treatment of Missing Values

The main goal of the project is to implement an algorithm to estimate a distribution over missing values by means of multiple imputation. The algorithms above for estimating the inverse temperature  $\beta$  (Section 2.1) and model sampling (Section 2.2) assume a completely observed lattice  $X$ . The problem of estimating  $\beta$  and calculating the model when missing values are present can be circumvented using a Gibbs sampler. The Gibbs sampler for obtaining  $M$  samples of the complete lattice works as follows:

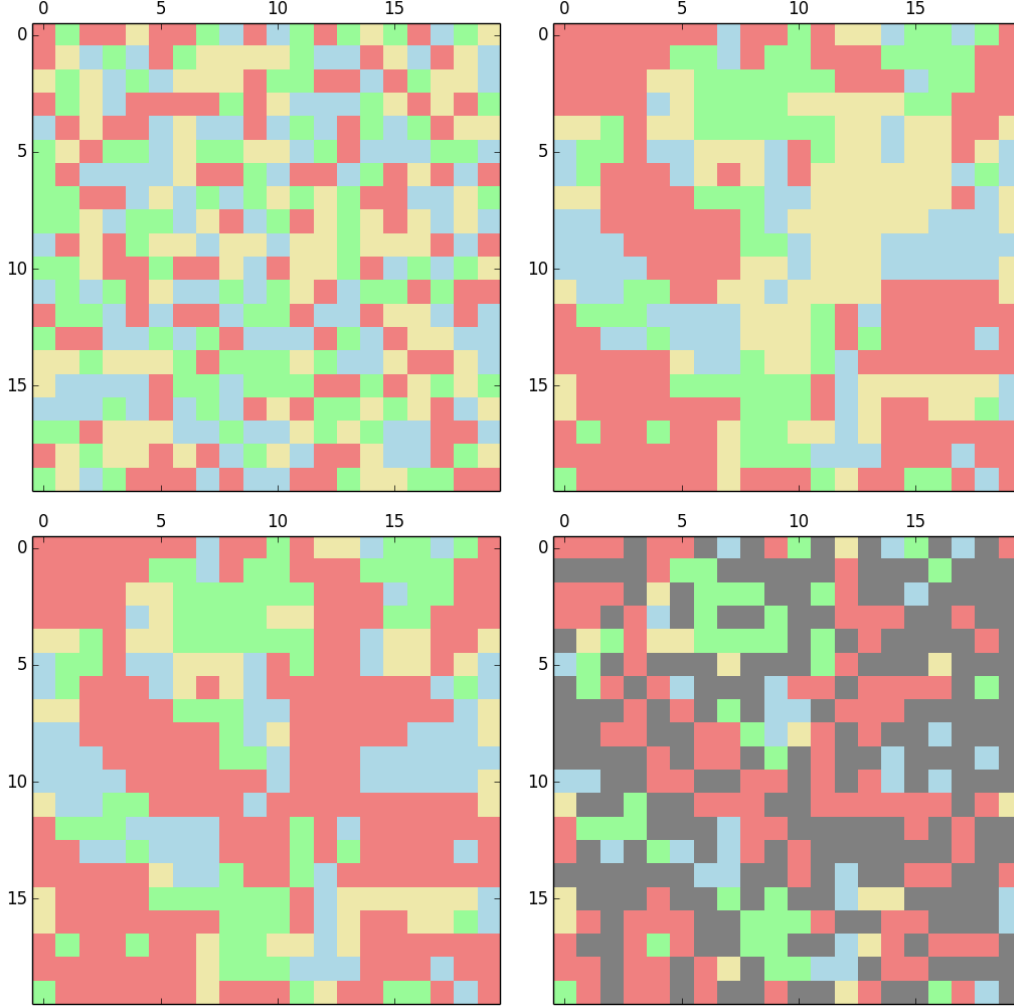


Figure 1: Potts model with  $L = 4$  classes on an  $N \times N = 20 \times 20$  dimensional grid. Topleft: random initialization; Topright: state after 401 iterations; bottomleft: state after 402 iterations of the Swendsen Wang Algorithm described in Section 2.2.2 with inverse temperature  $\beta = 1$ , note the flip of the central yellow cluster to green; bottomright: randomly sampled 50% missing values in grey.

### Multiple Imputation of Missing Values in Potts Model

1. Initialize all missing values  $X_{\text{miss}}$  randomly from  $\{1, 2, \dots, L\}$  to obtain  $X^0$
2. Initialize  $\beta^0$
3. For  $b$  in  $1, 2, \dots, B$  (burn-in):
  - 3.1 Estimate  $\beta^b$  from  $X^{b-1}$  using MPLE (7)
  - 3.2 Estimate  $X_{\text{miss}}^b$  using SW
4. For  $m$  in  $B + 1, B + 2, \dots, B + M$  (sampling from distribution):
  - 4.1 Estimate  $\beta^m$  from  $X^{m-1}$  using MCMC (8)
  - 4.2 Estimate  $X_{\text{miss}}^m$  using SW
5. Return  $\{X^{B+1}, X^{B+2}, \dots, X^{B+M}\}$

#### 2.3.1 Missing Values in Temperature Estimation

The presence of missing values does not affect the estimation of the inverse temperature  $\beta$  in Steps 3.1 and 4.1. The grid can be assumed to be fully observed and the Gibbs sampler ensures asymptotically unbiased estimates of  $\beta$ . Initializing missing values randomly will result in a dramatical underestimation of the inverse temperature  $\beta$ . A long burnin could be necessary to move the grid into an equilibrium state with stable estimated  $\beta$  when  $\beta$  is large and/or a large number of values in the grid are missing, see Figure 2.

#### 2.3.2 Missing Values in Model Sampling

Caution should be exercised in Step 3.2 as standard SW (Section 2.2.2) will flip any  $X_{ij} \in X$  regardless of whether are observed or missing. In this application, only missing pixels  $X_{ij} \in X_{\text{miss}}$  should be flipped, as observed pixels  $X_{kl} \in X_{\text{obs}}$  are, as the name suggests, observed and thus fixed. Proposing edge sets  $E^*$  that result in clusters containing only unobserved pixels is non-trivial.<sup>1</sup> An ad-hoc solution to this problem is to keep proposing new edge sets and thus clusters until all observations in the cluster selected for flipping are unobserved. Then, the cluster can be flipped with probability 1. A GIF of SW converging towards its equilibrium in the presence of missing values in a Potts model can be found following this [link](#). Note that the clusters grow in size over time and the number of pixels flipped per update step also increases. A GIF of SW sampling observations for missing values in a Potts model

---

<sup>1</sup>It is actually trivial to propose edge sets and thus clusters that contain only unobserved pixels, but quite difficult to find a clustering strategy for which the acceptance probabilities are not computationally complex, differ by proposed class, and are definitely not equal to 1.

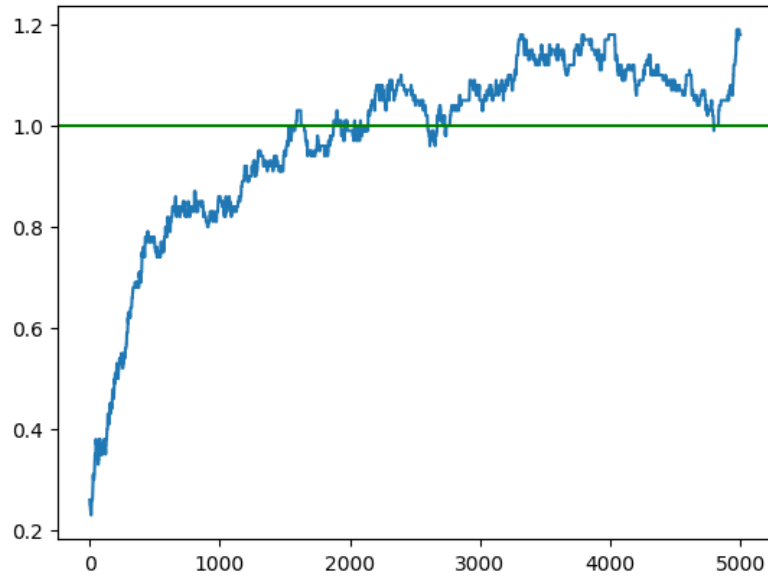


Figure 2: Burnin of the grid shown in the bottomleft of Figure 1 when missing values are initialized randomly. The plot shows the number of update steps on the  $x$  axis vs. estimated  $\beta$  on the  $y$  axis. The true  $\beta = 1$  is shown in green.

in equilibrium can be found following this [link](#). Note that the observed values in the grid never change.

### 3 Future Research

There are a number of open research question that could not be addressed due to time constraints of this projects. The first question is how the ratio of missing values affect the estimation of  $\beta$ . Undoubtedly, the variance will increase, but a bias towards smaller values of  $\beta$  could be introduced, as fewer and fewer (in the extreme case no) neighboring pixels are observed. The estimator could also run into the risk of diverging for small proportions of observed values.

Another interesting problem is the analysis of the run-time of SW and the Gibbs sampler for missing values for varying proportions of missing values and  $\beta$ . The Gibbs sampler would fail to work for large  $\beta$  and the proposed SW algorithm would have a low probability of sampling clusters with only unobserved variables and thus a low probability of actually executing a flip.

## References

- Geman, S. and Geman, D. (1987). Stochastic relaxation, gibbs distributions, and the bayesian restoration of images, *Readings in computer vision*, Elsevier, pp. 564–584.
- Lenz, W. (1920). Beitrage zum verstandnis der magnetischen eigenschaften in festen korpern, *Physikalische Zeitschrift* **21**: 613–615.
- Levada, A. L., Mascarenhas, N. D., Tannús, A. et al. (2009). Pseudo-likelihood equations for potts model on higher-order neighborhood systems: A quantitative approach for parameter estimation in image analysis, *Brazilian Journal of Probability and Statistics* **23**(2): 120–140.
- Onsager, L. (1944). Crystal statistics. i. a two-dimensional model with an order-disorder transition, *Physical Review* **65**(3-4): 117.

## Digital Appendix

The Python implementation of all methods, figures, and GIFs presented in this report can be found in a publicly accessible [Github repository](#). The files `potts_SW_missing.py` and `animations.py` can be executed to recreate all figures and GIFs. The repository also contains all figures, GIFs, and the Latex code used to generate this report.