Dr. Tim Brereton
Lisa Handl

Summer Term 2015

## Stochastic Simulation

# Problem Sheet 6

Deadline: June 11, 2015 at noon before the exercises

*Please email your code to lisa.handl@uni-ulm.de AND hand in a printed copy of the code!*

**Exercise 1 (theory)** (2 + 2 points)

a) Let $\{X_n\}_{n\in\mathbb{N}}$ be a Markov chain. Show that $\{Y_n\}_{n\in\mathbb{N}}$, where $Y_n = X_{2n}$ for all $n \in \mathbb{N}$, is a Markov chain.

b) Let $\{X_s\}_{s\in S}$ be a Markov random field with $S = \mathbb{Z}_m^2$ and neighborhood system, $\mathcal{N}$, given by

$$\mathcal{N}_s = \{(i,j) \in \mathbb{Z}_m^2 : |i - s_1| + |j - s_2| = 1\}$$

for all $s \in S$. Now, consider the case where we can only observe the values of $\{X_s\}_{s\in S}$ on the set

$$\widetilde{S} = \{(i,j) \in \mathbb{Z}_m^2 : i + j \text{ is even}\}.$$

Show that $\{X_s\}_{s\in\widetilde{S}}$ is a Markov random field with respect to the neighborhood system, $\widetilde{\mathcal{N}}$, given by

$$\widetilde{\mathcal{N}}_s = \{(i,j) \in \mathbb{Z}_m^2 : |i - s_1| + |j - s_2| = 2\}$$

for all $s \in \widetilde{S}$.

**Exercise 2 (theory)** (3 + 2 points)

Consider the model given in Exercise 5 of the last problem sheet (this is called the *hardcore model with fugacity*). Recall we defined the set, $A$, of all $5 \times 5$ matrices with values in $\{0,1\}$ and the subset $\Omega \subset A$ of all such matrices which have no 1s directly next to each other in a row or column. We put the following distribution on $\Omega$:

$$\boldsymbol{\pi}_T(\boldsymbol{x}) = \frac{1}{Z_T} \mathrm{e}^{-\frac{1}{T}\mathcal{E}(\boldsymbol{x})} \quad \forall \boldsymbol{x} \in A, \tag{1}$$

with the energy function

$$\mathcal{E}(\boldsymbol{x}) = \begin{cases} -\sum_{s\in S} x_s \log(\lambda) & \text{if } \boldsymbol{x} \in \Omega, \\ \infty & \text{otherwise,} \end{cases}$$

where $T = 1$ and $\lambda \in (0, \infty)$. Here $S$ denotes the set of all sites in the matrix $\boldsymbol{x}$.

a) Is $\pi_T$ the distribution of a Gibb's field (i.e., is the energy function a sum over a Gibb's potential)? If so, give the cliques (you can just draw them if you would like) and specify the Gibb's potential.

b) Give the local energy and local specification of $x_s$.

**Exercise 3 (programming)** (3 points)

Write a program to sample from the model described in Exercise 2 using the Gibb's sampler, run it for $\lambda = 0.5$ and $\lambda = 1.5$ and at least $N = 10^5$ steps, and estimate the expected number of ones in the matrix.

**Exercise 4 (programming)** (4 points)

The *Pott's model* is a generalization of the Ising model, where each variable is able to take values in the space $\Lambda = \{1, \ldots, M\}$. Everything else is as in the Ising model, except that the energy function is given by

$$\mathcal{E}(x) = -J \sum_{(s,t)} \delta(x_s, x_t) - \sum_{s \in S} h_s(x_s),$$

where

$$\delta(x, y) = \begin{cases} 1 & \text{if } x = y, \\ 0 & \text{otherwise,} \end{cases}$$

$J \in \mathbb{R}$, and the $\{h_s\}$ are some functions. Assuming that $h_s(x) = 0$ for all $x \in \Lambda$ and $s \in S$, we have the simplified model with energy function

$$\mathcal{E}(x) = -J \sum_{(s,t)} \delta(x_s, x_t).$$

Write a program to simulate from this simplified model on a $10 \times 10$ lattice with periodic boundary conditions, temperature $T = 1$, $J = 3$ and $M = 5$ using the Gibb's sampler. Run it for at least $N = 10^5$ steps and estimate the expected sum of all values in the random field.

**Exercise 5 (theory)** (3 + 2 points)

a) Prove that the Gibb's sampler is a special case of the Metropolis-Hastings algorithm, i.e., specify the proposal distributions $(Q)_{i,j}$ and show that the acceptance probability is always 1.

b) Is the Gibb's sampler also a special case of the Metropolis algorithm? Justify your answer.

**Exercise 6 (programming)** (4 + 3 points)

In a knapsack problem you consider a set of $n$ objects for which you know the weights $w_1, \ldots, w_n$ and the values $v_1, \ldots, v_n$. The goal is to pack a selection of these objects into a knapsack (backpack), so that the total value of things in the knapsack is as large as possible, but the total weight does not exceed a given limit, $w_{max}$.

a) Explain how you could use simulated annealing to solve this problem. In particular:

- Define an appropriate state space $S$.

- Define an appropriate cost function $f$ on $S$. Explain how you include the weight limit.
- Explain how you can take a "random step" in $S$, such that the resulting Markov chain is irreducible.
- Write out the acceptance probability of such a random step.

b) Write a Matlab program solving the following knapsack problem:

Imagine you want to go on a day trip into mountains. You have many things you want to take with you, but you can only carry a maximum weight of 400. The following list contains the things you would like to take with you, their weights and their values to you.

| object | weight | value |
|---|---|---|
| apple | 39 | 40 |
| banana | 27 | 60 |
| beer | 52 | 10 |
| book | 30 | 10 |
| camera | 32 | 30 |
| cheese | 23 | 30 |
| compass | 13 | 35 |
| glucose | 15 | 60 |
| map | 9 | 150 |
| note-case | 22 | 80 |
| sandwich | 50 | 160 |
| socks | 4 | 50 |
| sunglasses | 7 | 20 |
| suntan cream | 11 | 70 |
| t-shirt | 24 | 15 |
| tin | 68 | 45 |
| towel | 18 | 12 |
| trousers | 48 | 10 |
| umbrella | 73 | 40 |
| water | 153 | 200 |
| waterproof overclothes | 43 | 75 |
| waterproof trousers | 42 | 70 |

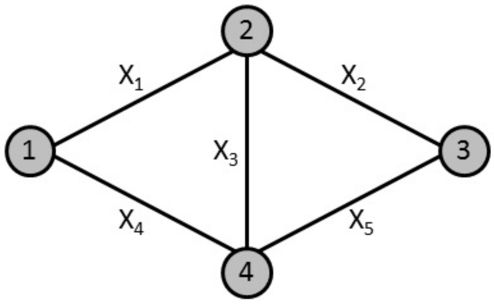
Using an initial temperature of $T_0 = 100$, geometric cooling with $\beta = 0.999$ and an arbitrary initial state, run the algorithm for at least $N = 10^4$ iterations. Plot the total value and weight of the objects in the knapsack in each iteration and list the final (best) set of objects you should take with you.

*Hints:*

- You can use `dlmread` to read the data from the file *hiking.txt* on the course website.
- If `a` and `b` are vectors, you can use `setdiff(a,b)` to get all entries of `a` which are are not in `b`.

**Exercise 7 (programming)** (3 + 1 points)

Consider the graph




with random independent edge lengths $X_1, \dots, X_5 \sim \text{Poi}(2)$.

a) Write a Matlab program to sample from $(X_1, \dots, X_5)$ **conditional on the shortest path from vertex 1 to vertex 3 being longer than 4** using the Gibb's sampler. and start from the initial state $(4, 4, 4, 4, 4)$.

b) Run the algorithm for $N = 10^5$ iterations and estimate the mean length of the shortest path from 1 to 3 in this model.

*Hint:* You can use `poissrnd` to sample from a Poisson distribution.