Spatial statistics and image analysis (TMS016/MSA301)

Statistical image modeling: Markov random fields

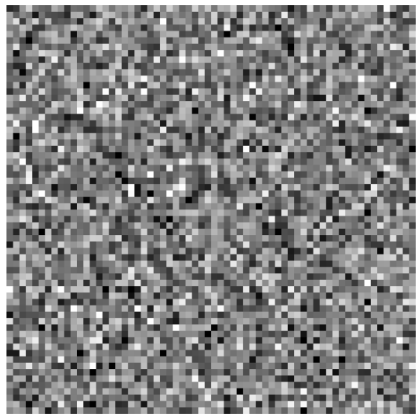
2022-04-25

Practical information: project part 3

- Deadline for the planning report is today.
- Deadline for the preliminary report is May 18.
- The projects will be presented on the 23rd and 25th of May. Each group gives a short presentation (appr. 15 min) of their work.
- Deadline for the final report is May 27

Simple models with independent pixel values

- Binary image: randomly assign the value "black" with probability p and "white" with probability $1 - p$ independently for each pixel (top, $p = 0.5$)
- Grey level image: randomly assign a value from $N(\mu, \sigma^2)$ distribution independently for each pixel (bottom, $N(0.5, 0.2^2)$)



Markov random fields (MRF)

MRFs can be used to model the dependence between the pixel values. They can be used

- in classification/segmentation
- in pattern recognition
- in image restoration
- as priors for the unobserved image in Bayesian image reconstruction
- as approximations for Gaussian fields (Gaussian MRF)

Markov chain (one dimensional)

A random sequence X_t which takes values in a finite or countable set V is a Markov chain if

$$\begin{aligned} &\mathrm{P}(X_{i+1} = x | X_0 = x_0, X_1 = x_1, ..., X_{i-1} = x_{i-1}, X_i = x_i) \\ &\quad = \mathrm{P}(X_{i+1} = x | X_i = x_i), \quad x \in V. \end{aligned}$$

This means that predictions regarding future outcomes can be made based only on the present state (value) and such predictions are just as good as the ones that would be made knowing the full history of the process.

Question: How can this be generalized to two (or higher) dimensions?

Markov random field (two dimensional)

We have a random image $X = (X_s, s \in S)$, where S is the set of sites (or pixel locations). For each site s, we define a neighbourhood $N_s \subset S$ such that

- $s \notin N_s$, i.e. s is not its own neighbour
- $t \in N_s$ if and only if $s \in N_t$ (symmetry)

Often, the neighbourhood is defined as the set of the four nearest or the eight nearest neighbours.

Markov random field: definition

$X = (X_s,\, s \in S)$ is a set of discrete random variables taking values in V is a Markov random field with respect to the neighbourhood (or system of neighbourhoods) $(N_s,\, s \in S)$ if

$$\mathrm{P}(X_s = x | X_t, t \neq s) = \mathrm{P}(X_s = x | X_t,\, t \in N_s).$$

This means that the value of the pixel in s depends only on the pixel values in the neighbourhood of s.

Boundary condition

When predicting a value of a pixel on the boundary of the region, we do not have information on all the neighbouring values.

One possible solution is to use periodic boundaries, i.e. to wrap the rectangular area onto a torus (doughnut) by identifying the opposite edges. For example, to predict the value of pixel (i, n) $(2 \leq i \leq m-1)$ in an $m \times n$ image, the four nearest neighbour neighbourhood of the pixel would be

$$\{(i-1, n), (i+1, n), (i, n-1), (i, 1)\}$$

Example: Ising model

Ising (1925): A mathematical model of ferromagnetism in statistical mechanics.

The model consists of a set of spins X_s at locations $s \in S$, where S is an $n \times m$ lattice (typically large in physics applications) with periodic boundary conditions and with the four nearest neighbour neighbourhood.

Each spin is pointing either up or down, i.e.

$$X_s = \begin{cases} +1 & \text{if spin up} \\ -1 & \text{if spin down} \end{cases}$$

Ising model

The probability distribution of $X = (X_s, s \in S)$ can be given by a Gibbs distribution

$$\mathrm{P}(X = x) = \frac{1}{Z} \exp(\beta \sum_{s \sim t} x_s x_t),$$

where

- Z is a normalizing constant (a sum over all possible configurations of spins up and down),
- $\beta > 0$ (interpreted as inverse temperature)
- $s \sim t$ denotes that s and t are neighbours.
- $\sum_{s \sim t} x_s x_t$ is the difference between the number of neighbour pairs with similar spins and the number or neighbour pairs with dissimilar spins.

Ising model

The conditional probability of a spin being up given the spins in its neighbourhood is

$$\mathrm{P}(X_s = +1 | X_t,\, t \in N_s) = \frac{\exp(2\beta(X_s^+ - X_t^-))}{1 + \exp(2\beta(X_s^+ - X_t^-))},$$

where

- X_s^+ is the number of neighbours of site s that have positive spins (value $+1$)
- X_s^- is the number of neighbours of site s that have negative spins (value -1)

If $X_s^+ > X_s^-$, then the probability of having a spin up, i.e. $\mathrm{P}(X_s = +1)$, is greater than $1/2$.

Ising model: phase transition

Ising model has a critical value β_c meaning that the behaviour of the model is different above and below the critical value.

For $\beta > \beta_c$ (or temperature below the critical temperature)

- long range dependencies
- possible phase transitions: either a clear majority with spin up ($X_s = +1$) or a clear majority with spin down ($X_s = -1$)

For $\beta < \beta_c$ (or temperature above the critical temperature)

- no phase transitions
- in a large area, the average value of X_s is close to zero (as many spins up as spins down).

Onsager (1944) showed that $\beta_c = \frac{1}{2}\log(1+\sqrt{2}) \approx 0.44069$.

Ising model: simulated realizations

Realizations (binary images) of the Ising model with (from left to right) $\beta = 0.11, 0.22, 0.4407, 0.88$ and 1.76.

Autonormal MRF model

Let us have an MRF model, where $X_s,\ s \in S$, are continuous real-valued random variables. The Markov property can now be written as

$$\mathrm{P}(X_s \in A | X_t, t \neq s) = \mathrm{P}(X_s \in A | X_t, t \in N_s),\ A \subseteq \mathbb{R}, s \in S$$

for all considered subsets A of $\mathbb{R}$.

An example of such models is the so-called autonormal model, where

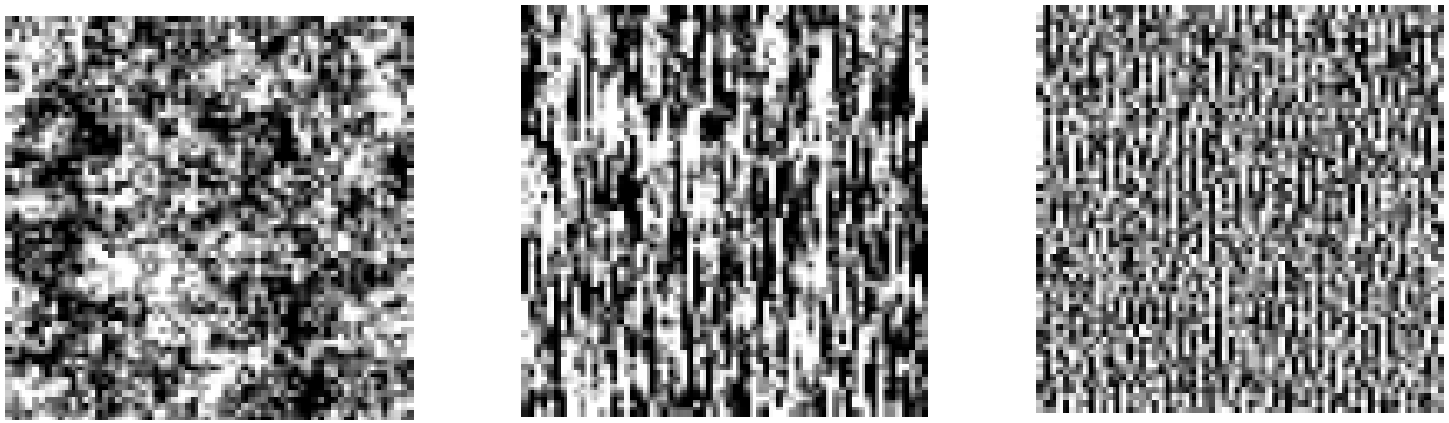
$$X_s | N_s \sim N(\mathrm{E}(X_s | X_t, t \in N_s), \sigma^2)$$

and the expectation is a linear combination of the neighbouring values. For example, if we consider the four nearest neighbours s_W (west), s_N (north), s_E (east), and s_S (south) of s, we can write

$$\begin{aligned}\mathrm{E}(X_s | X_t, t \in N_s) = \mu &+ \beta_W(X_{s_W} - \mu) + \beta_N(X_{s_N} - \mu) \\ &+ \beta_E(X_{s_E} - \mu) + \beta_S(X_{s_S} - \mu).\end{aligned}$$

Autonormal model: simulated images

Realizations of the autonormal model ($\mu = 0.5$, $\sigma = 0.3$)



- $\beta_W = \beta_E = \beta_N = \beta_S = 0.24$ (left)
- $\beta_W = \beta_E = 0$ and $\beta_N = \beta_S = 0.48$ (middle)
- $\beta_W = \beta_E = -0.24$ and $\beta_N = \beta_S = 0.24$ (right)

Simulation of MRFs

We want to simulate realizations from the distribution $\mathrm{P}(\mathrm{x}) = \mathrm{P}(x_1, ..., x_n)$ but it is difficult due to the normalizing constant Z.

Simulating realizations from $\mathrm{P}(x_s|x_t, t \neq s)$ (or $\mathrm{P}(x_s|N_s)$) is, therefore, quite easy by using Markov chain Monte Carlo (MCMC) methods

- Gibbs sampling
- Metropolis-Hastings algorithm

Simulation of MRFs: Gibbs sampling

Step 1: Choose a starting value (configuration) $\mathbf{x}^0$

Step 2: Repeat for $i = 1, ..., N$:

- Draw $x_1^{(i)}$ from $\mathrm{P}(x_1|x_2^{(i-1)}, ..., x_n^{(i-1)})$
- Draw $x_2^{(i)}$ from $\mathrm{P}(x_2|x_1^{(i)}, x_3^{(i-1)}, ..., x_n^{(i-1)})$
- ...
- Draw $x_n^{(i)}$ from $\mathrm{P}(x_n|x_1^{(i)}, ..., x_{n-1}^{(i)})$

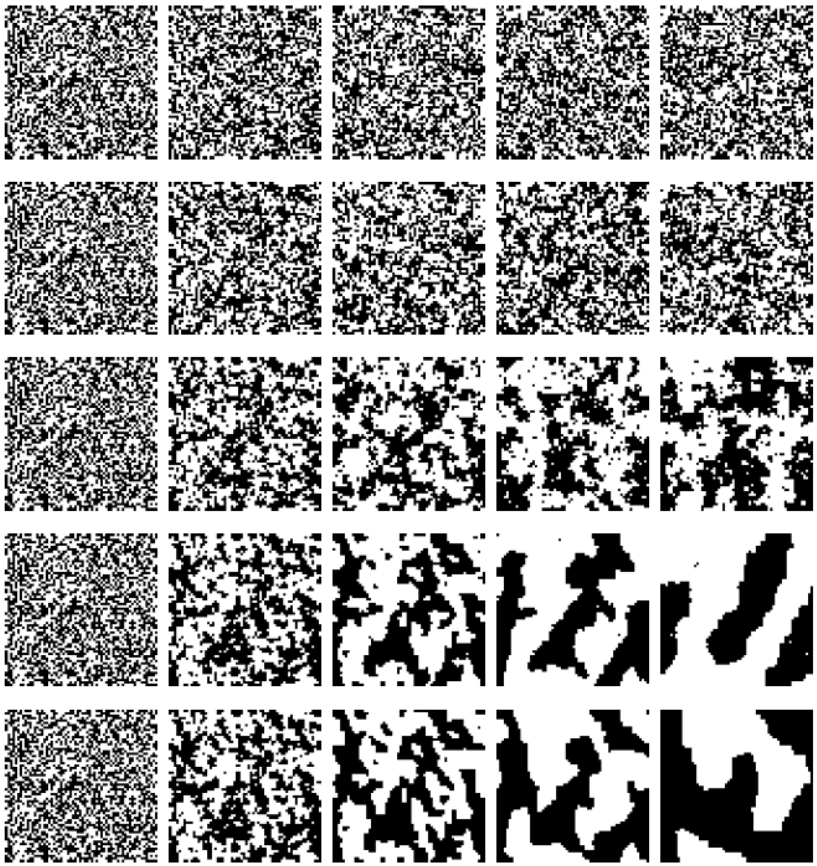
Step 3: Use $\mathbf{x}^{(K)}, ..., \mathbf{x}^{(N)}$ as a sequence of dependent draws

Note that to compute $\mathrm{P}(x_1|x_2^{(i-1)}, ..., x_n^{(i-1)})$ (and the other probabilities above), it is enough to condition on the pixel values in the neighbourhood of the pixel 1 (or the pixel we are updating).

Gibbs sampling

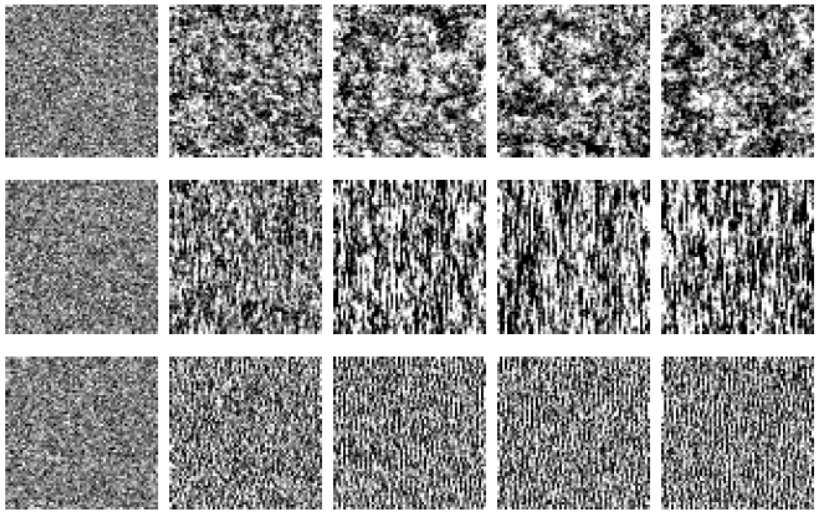
- Samples are approximately from the distribution $\mathrm{P}(\mathrm{x})$. Under some mild conditions, $\mathrm{P}(x^{(i)})$ converges to $\mathrm{P}(\mathrm{x})$.
- K should be chosen large enough so that the chain has converged.

Ising model: simulation



- From top to bottom, $\beta = 0.11, 0.22, 0.4407, 0.88$ and 1.76.
- From left to right, a purely random start configuration, after 1 sweep, after 4 sweeps, after 16 sweeps, and after 64 sweeps.

Autonormal model: simulation



- $\mu = 0.5$, $\sigma = 0.3$, and
  - $\beta_W = \beta_E = \beta_N = \beta_S = 0.24$ (top)
  - $\beta_W = \beta_E = 0$, $\beta_N = \beta_S = 0.48$ (middle)
  - $\beta_W = \beta_E = -0.24$, $\beta_N = \beta_S = 0.24$ (bottom)
- From left to right, a purely random start configuration, after 1 sweep, after 16 sweeps, after 128 sweeps, and after 256 sweeps.

Metropolis-Hastings algorithm

We update the sites in a similar manner as in Gibbs sampling (one at the time) but the proposal distribution depends on the current state:

- Site s is in state x_s and the neighbouring sites are in state $x(N_s)$.
- Propose a new state y for the site s by using a proposal density $g_s(y|x_s)$.
- Accept the proposal with probability (Hastings ratio)

$$A(y, x_s) = \min\left\{1, \frac{\mathrm{P}(y|x(N_s))g_s(x_s|y)}{\mathrm{P}(x_s|x(N_s))g_s(y|x_s)}\right\}$$

- The proposal density can depend on the site s and the values x_s. In the case where the MRF takes only a finite number of values, g_s can be uniform.

Gibbs sampling or Metropolis-Hastings algorithm?

- In general, Gibbs sampler tends to converge more slowly than Metropolis-Hastings
- For binary (black and white) MRFs, it is trivial to apply the Gibbs sampler but when the number of possible values at each site is large, Gibbs sampler can be cumbersome and costly in computer time.
- Gibbs sampler is a special case of Metropolis-Hastings, where the proposal distribution is $\mathrm{P}(y|x(N_s))$.
- Metropolis-Hastings algorithm updates always a site if the proposal is "better" than the current value, Gibbs sampler does not.

Bayesian image restauration

The true (random) image $X = (X_s, s \in S)$ is given by an MRF but we observe a distorted image $Y = (Y_s, s \in S)$.

Question: How to reconstruct X from Y?

Simple model for the observed Y: We assume that

- the Y_s variables are independent given X
- the distribution of Y_s only depends on X_s, i.e.

$$\mathrm{P}(Y = y|X) = \prod_{s \in S} \mathrm{P}(Y_s = y_s|X_s). \tag{1}$$

Restauration of X: To find the original image, we would need to compute the posterior distribution (Bayes theorem)

$$\mathrm{P}(X = x|Y = y) = \frac{\mathrm{P}(Y = y|X = x)\mathrm{P}(X = x)}{\mathrm{P}(Y = y)} \tag{2}$$

which can be difficult computionally.

Bayesian image restauration

- Markov chain Monte Carlo algorithms can be used to compute the posterior distribution
- MRF models can be used as priors for the unobserved image X but it can be difficult to specify realistic priors for typical images.