TMA4300 Computer Intensive Statistical Methods
Exercise 2, Spring 2014

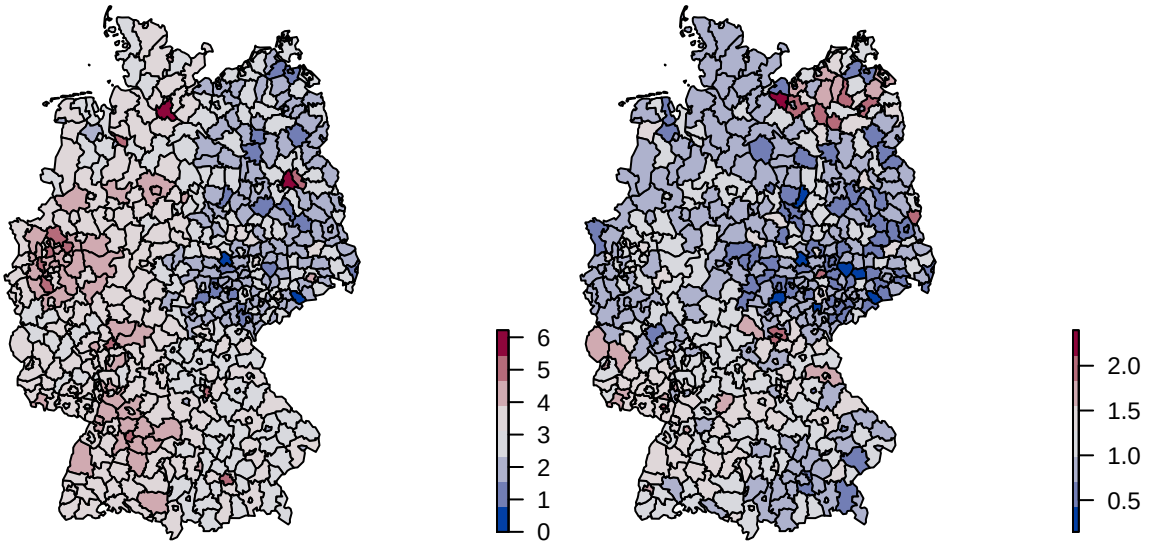
Note: The solution to ALL exercises must be handed in no later than **April 1th 2014**.
Preliminary steps: Please install the packages `fields` and `INLA` by typing in the R-terminal

```
install.packages("fields")
source("http://www.math.ntnu.no/inla/givemeINLA-testing.R")
```

The goal is to carry out a spatial analysis on mortality rates of oral cavity cancer in Germany during a 5-year period, 1986–1990, for $n = 544$ districts. Observed counts are y_i, expected counts are e_i, and log relative risk is η_i. The data is available in `R` by

```
library(spam)          # load the data
str(Oral)              # see structure of data
#'data.frame': 544 obs. of  3 variables:
# $ Y  : int  18 62 44 12 18 27 20 29 39 21 . . .
# $ E  : num  16.4 45.9 44.7 16.3 26.9 . . .
# $ SMR: num  1.101 1.351 0.985 0.735 0.668 . . .
attach(Oral)           # allow direct referencing to Y and E
# generate some plots
library(fields, warn.conflict=FALSE)
library(colorspace)
col <- diverge_hcl(8)      # blue - red
# alternative colors
# col <- rev(gray(0:8 / 8))  # gray scales
# col <- rev(heat_hcl(64))
# use the function provided by spam
map.landkreis(log(Oral$Y),col=col)
map.landkreis(Oral$Y/Oral$E,col=col)
```

On the left, log counts $\log(y_i)$ are shown. On the right the standardised mortality rates (SMR) $\frac{y_i}{e_i}$ are shown.



Assuming observed counts to be conditionally independent Poisson, the model is

$$y_i \mid \eta_i \sim \text{Pois}(e_i \exp(\eta_i)), \qquad i = 1, \ldots, n. \tag{1}$$

The log relative risk, $\boldsymbol{\eta} = (\eta_1, \ldots, \eta_n)^T$, is then decomposed into

$$\boldsymbol{\eta} = \boldsymbol{u} + \boldsymbol{v}.$$

Component $\boldsymbol{u} = (u_1, \ldots, u_n)^T$ is spatially structured with smoothing parameter κ_u. Component $\boldsymbol{v} = (v_1, \ldots, v_n)^T$ is unstructured white-noise with precision parameter κ_v, i.e. $\mathcal{N}(\mathbf{0}, \kappa_v^{-1}\mathbf{I})$. (Note: An equivalent model would be $\boldsymbol{\eta} = \mu\mathbf{1} + \boldsymbol{u} + \boldsymbol{v}$, but would then require an additional sum to zero constraint on $\boldsymbol{u}$.) The distribution of $\boldsymbol{\eta}$, conditional on the spatial component $\boldsymbol{u}$ and κ_v, is now

$$\boldsymbol{\eta} \mid \boldsymbol{u}, \kappa_v \sim \mathcal{N}(\boldsymbol{u}, \kappa_v^{-1}\mathbf{I}). \tag{2}$$

A common way to introduce a spatially correlated effect is to assume that neighbouring districts are more similar than distant districts, therefore for a valid prior definition, a neighbourhood has to be defined for each district. In geographical applications a common assumption is that two districts are neighbours if they share a common border. If we consider a single district, and condition on only the neighbours with which it shares a border, this is a first-order autoregressive process, or intrinsic Gaussian Markov random field (Rue and Held, 2005). The density is then

$$\text{p}(\boldsymbol{u} \mid \kappa_u) \propto \kappa_u^{(n-1)/2} \exp\left(-\frac{\kappa_u}{2} \sum_{i \sim j} (u_i - u_j)^2\right) \tag{3}$$

$$= \kappa_u^{(n-1)/2} \exp(-\frac{\kappa_u}{2} \boldsymbol{u}^T \mathbf{R} \boldsymbol{u}). \tag{4}$$

The sum in (3) goes over all pairs of neighbouring regions $i \sim j$ and the structure matrix $\mathbf{R}$ defines the neighbour structure:

$$R_{ij} = \begin{cases} n_i & i = j \\ -1 & i \sim j \\ 0 & \text{otherwise,} \end{cases}$$

where n_i denotes the number of neighbouring regions of region i. The precision terms are assigned the prior distributions

$$\kappa_u \sim \text{Gamma}(\alpha_u, \beta_u), \tag{5}$$

$$\kappa_v \sim \text{Gamma}(\alpha_v, \beta_v). \tag{6}$$

The analysis will require implementation of a MCMC sampler. There are two strategies that will be covered, separate parameter updates (GI) and a block update (BL):

(GI) A Gibbs sampler with individual parameter updates that utilise the full conditional distributions. One parameter, $\boldsymbol{\eta}$, does not have a "standard" full conditional density and will require a Metropolis-Hastings step.

(BL) A Metropolis-Hastings step to block update the latent parameters $\boldsymbol{x} = (\boldsymbol{u}^T, \boldsymbol{\eta}^T)^T$ jointly. The hyperparameters κ_u and κ_v are updated separately by sampling from its corresponding full-conditional distributions.

Exercise 1 (Derivation of Posterior [*no programming needed*])

(a) Show that the full joint posterior $\mathrm{p}(\boldsymbol{\eta}, \boldsymbol{u}, \kappa_u, \kappa_v \mid \boldsymbol{y})$ is proportional to

$$\kappa_u^{\frac{n-1}{2}+\alpha_u-1}\kappa_v^{\frac{n}{2}+\alpha_v-1}\exp\left(-\beta_u\kappa_u-\beta_v\kappa_v-\frac{\kappa_v}{2}(\boldsymbol{\eta}-\boldsymbol{u})^T(\boldsymbol{\eta}-\boldsymbol{u})-\frac{\kappa_u}{2}\boldsymbol{u}^T\mathbf{R}\boldsymbol{u}+\sum_i y_i\eta_i-\exp(\eta_i)e_i\right).$$

(b) Due to the non-normality, sampling from the posterior will require a Metropolis–Hastings step. To obtain a proposal distribution that is easy to sample from, i.e. Gaussian, approximate the function

$$f(\eta_i) = y_i\eta_i - \exp(\eta_i)e_i$$

with a second order Taylor series expansion, $\widetilde{f}(\eta_i)$ at the point η_{0_i}. Show that the approximation can be written as

$$\widetilde{f}(\eta_i) = a_i + b_i\eta_i - \frac{1}{2}c_i\eta_i^2, \tag{7}$$

with $a_i = e_i\exp(\eta_{0_i})\cdot(\eta_{0_i}-\frac{1}{2}\eta_{0_i}^2-1)$, $b_i = y_i + e_i\exp(\eta_{0_i})\cdot(\eta_{0_i}-1)$ and $c_i = e_i\exp(\eta_{0_i})$.

(c)

(GI) Derive the full conditional densities $\mathrm{p}(\kappa_u \mid \boldsymbol{y}, \kappa_v, \boldsymbol{\eta}, \boldsymbol{u})$ and $\mathrm{p}(\kappa_v \mid \boldsymbol{y}, \kappa_u, \boldsymbol{\eta}, \boldsymbol{u})$. Derive full conditional posterior densities for $\mathrm{p}(\boldsymbol{u} \mid \boldsymbol{y}, \kappa_v, \kappa_u, \boldsymbol{\eta})$ and $\mathrm{p}(\boldsymbol{\eta} \mid \boldsymbol{y}, \kappa_v, \kappa_u, \boldsymbol{u})$. Notice that the first one is a standard density that is straightforward to sample from. Show that the second density is

$$\mathrm{p}(\boldsymbol{\eta} \mid \boldsymbol{y}, \kappa_v, \kappa_u, \boldsymbol{u}) \propto \exp\left(-\frac{1}{2}\boldsymbol{\eta}^T(\kappa_v\mathbf{I})\boldsymbol{\eta} + \boldsymbol{\eta}^T(\kappa_v\boldsymbol{u}) + \boldsymbol{\eta}^T\boldsymbol{y} - \exp(\boldsymbol{\eta})^T\boldsymbol{e}\right).$$

Then show, using (7), that this can be approximated by a normal density

$$q(\boldsymbol{\eta} \mid \boldsymbol{\eta}_0, \boldsymbol{y}, \boldsymbol{u}, \kappa_u, \kappa_v) \propto \exp\left(-\frac{1}{2}\boldsymbol{\eta}^T(\kappa_v\mathbf{I} + \mathrm{diag}(\boldsymbol{c}))\boldsymbol{\eta} + \boldsymbol{\eta}^T(\kappa_v\boldsymbol{u} + \boldsymbol{b})\right),$$

where $\boldsymbol{b} = (b_1, \ldots, b_n)^T$, $\boldsymbol{c} = (c_1, \ldots, c_n)^T$.

(BL) Here, we also need, as for the GI sampler, the full conditional densities $\mathrm{p}(\kappa_u \mid \boldsymbol{y}, \kappa_v, \boldsymbol{\eta}, \boldsymbol{u})$ and $\mathrm{p}(\kappa_v \mid \boldsymbol{y}, \kappa_u, \boldsymbol{\eta}, \boldsymbol{u})$. However, instead of separately, we update $\boldsymbol{u}$ and $\boldsymbol{\eta}$ now jointly. Recall that $\boldsymbol{x} = (\boldsymbol{u}^T, \boldsymbol{\eta}^T)^T$ and show that

$$\mathrm{p}(\boldsymbol{x} \mid \boldsymbol{y}, \kappa_u, \kappa_v) \propto \exp\left(\boldsymbol{\eta}^T\boldsymbol{y} - \exp(\boldsymbol{\eta})^T\boldsymbol{e} - \frac{1}{2}\boldsymbol{x}^T\begin{pmatrix}\kappa_u\mathbf{R}+\kappa_v\mathbf{I} & -\kappa_v\mathbf{I}\\ -\kappa_v\mathbf{I} & \kappa_v\mathbf{I}\end{pmatrix}\boldsymbol{x}\right),$$

and that this can be approximated around the point $\boldsymbol{x}_0 = (\mathbf{0}^T, \boldsymbol{\eta}_0^T)^T$ as a normal density

$$q(\boldsymbol{x} \mid \boldsymbol{x}_0, \boldsymbol{y}, \kappa_u, \kappa_v) \propto \exp\left(-\frac{1}{2}\boldsymbol{x}^T\begin{pmatrix}\kappa_u\mathbf{R}+\kappa_v\mathbf{I} & -\kappa_v\mathbf{I}\\ -\kappa_v\mathbf{I} & \kappa_v\mathbf{I}\end{pmatrix}\boldsymbol{x} - \frac{1}{2}\boldsymbol{\eta}^T\,\mathrm{diag}(\boldsymbol{c})\boldsymbol{\eta} + \boldsymbol{b}^T\boldsymbol{\eta}\right),$$

by using (7), where $\boldsymbol{b} = (b_1, \ldots, b_n)^T$, $\boldsymbol{c} = (c_1, \ldots, c_n)^T$.

Exercise 2 (Implementation of MCMC Sampler)
After a suitable burn-in period, a posterior sample size of $M = 10000$, after thinning has been done, is recommended. Let m index the current iteration. The steps required for a single iteration are:

(GI)
1. Draw $\boldsymbol{\kappa}_u^{(m)}$ using full conditional $\mathrm{p}(\kappa_u \mid \boldsymbol{y}, \kappa_v^{(m-1)}, \boldsymbol{\eta}^{(m-1)}, \boldsymbol{u}^{(m-1)})$.
2. Draw $\boldsymbol{\kappa}_v^{(m)}$ using full conditional $\mathrm{p}(\kappa_v \mid \boldsymbol{y}, \kappa_u^{(m)}, \boldsymbol{\eta}^{(m-1)}, \boldsymbol{u}^{(m-1)})$.
3. Draw $\boldsymbol{u}^{(m)}$ using full conditional $\mathrm{p}(\boldsymbol{u} \mid \boldsymbol{y}, \kappa_u^{(m)}, \kappa_v^{(m)}, \boldsymbol{\eta}^{(m-1)})$.
4. Draw $\boldsymbol{\eta}^\star$ with proposal density $q(\boldsymbol{\eta}^\star \mid \boldsymbol{\eta}_0, \boldsymbol{y}, \boldsymbol{u}^{(m)}, \kappa_u^{(m)}, \kappa_v^{(m)})$, with approximation around $\boldsymbol{\eta}_0 = \boldsymbol{\eta}^{(m-1)}$.
5. Set $\boldsymbol{\eta}^{(m)} = \boldsymbol{\eta}^\star$ with probability

$$\alpha = \min\left(1, \frac{\mathrm{p}(\boldsymbol{\eta}^\star \mid \boldsymbol{y}, \kappa_v^{(m)}, \kappa_u^{(m)}, \boldsymbol{u}^{(m)})}{\mathrm{p}(\boldsymbol{\eta}^{(m-1)} \mid \boldsymbol{y}, \kappa_v^{(m)}, \kappa_u^{(m)}, \boldsymbol{u}^{(m)})} \frac{q(\boldsymbol{\eta}^{(m-1)} \mid \boldsymbol{\eta}^\star, \boldsymbol{y}, \kappa_v^{(m)}, \kappa_u^{(m)}, \boldsymbol{u}^{(m)})}{q(\boldsymbol{\eta}^\star \mid \boldsymbol{\eta}^{(m-1)}, \boldsymbol{y}, \kappa_v^{(m)}, \kappa_u^{(m)}, \boldsymbol{u}^{(m)})}\right),$$

otherwise $\boldsymbol{\eta}^{(m)} = \boldsymbol{\eta}^{(m-1)}$.

(BL)
1. Draw $\boldsymbol{\kappa}_u^{(m)}$ using full conditional $\mathrm{p}(\kappa_u \mid \boldsymbol{y}, \kappa_v^{(m-1)}, \boldsymbol{\eta}^{(m-1)}, \boldsymbol{u}^{(m-1)})$.
2. Draw $\boldsymbol{\kappa}_v^{(m)}$ using full conditional $\mathrm{p}(\kappa_v \mid \boldsymbol{y}, \kappa_u^{(m)}, \boldsymbol{\eta}^{(m-1)}, \boldsymbol{u}^{(m-1)})$.
3. Draw $\boldsymbol{x}^\star$ with proposal density $q(\boldsymbol{x}^\star \mid \boldsymbol{x}_0, \boldsymbol{y}, \kappa_u^{(m)}, \kappa_v^{(m)})$, so that approximation is around $\boldsymbol{x}_0 = \boldsymbol{x}^{(m-1)}$.
4. Set $\boldsymbol{x}^{(m)} = \boldsymbol{x}^\star$ with probability

$$\alpha = \min\left(1, \frac{\mathrm{p}(\boldsymbol{x}^\star \mid \boldsymbol{y}, \kappa_u^{(m)}, \kappa_v^{(m)})}{\mathrm{p}(\boldsymbol{x}^{(m-1)} \mid \boldsymbol{y}, \kappa_u^{(m)}, \kappa_v^{(m)})} \frac{q(\boldsymbol{x}^{(m-1)} \mid \boldsymbol{x}^\star, \boldsymbol{y}, \kappa_u^{(m)}, \kappa_v^{(m)}}{q(\boldsymbol{x}^\star \mid \boldsymbol{x}^{(m-1)}, \boldsymbol{y}, \kappa_u^{(m)}, \kappa_v^{(m)})}\right),$$

otherwise $\boldsymbol{x}^{(m)} = \boldsymbol{x}^{(m-1)}$.
(Caution: Be careful to involve all terms of q that do not cancel in the acceptance ratio).

Parameters of the gamma prior distributions should be set as $\alpha_u = \alpha_v = 1$ and $\beta_u = \beta_v = 0.01$. The matrix $\mathbf{R}$, in (4), can be calculated from an adjacency table, but has already been defined and is available with `load("tma4300_ex2_Rmatrix.Rdata")`. Note that log densities should always be used. For efficient computation, the sparsity of the precision matrices should be exploited. This will be done using the library `spam`. Some functions that may be of use are

- `diag.spam()` - create a diagonal matrix that is a sparse matrix object
- `rmvnorm.canonical()` - sample from a normal distribution using canonical parameterisation

Refer to their help pages for more information. In particular, note that the warning

```
Warning message:
Increased 'nnzcolindices' with 'NgPeyton' method
(currently set to 6467 from 5173)
```

can be avoided by adding `memory=list(nnzcolindices=6467)` as a function argument.
While running the samplers keep track of the acceptance rates for the Metropolis-Hastings steps. Further, use the function `system.time()` or `Sys.time()` to save information on how long the sampler needs to generate the M samples.

Exercise 3 (Convergence diagnostics)
For both BL) and GI) obtain the following diagnostic summaries for the precision parameters κ_u, κ_v, and exemplary for five randomly chosen components of $\boldsymbol{u}$ and $\boldsymbol{v}$.

(a) Trace plots.

(b) Autocorrelation plots.

(c) With `library(coda)`, use the functions `geweke.diag()` and `geweke.plot()` to check the Markov chains for convergence.

What do you observe? How do you interpret the results?

Exercise 4 (Effective sample size)
For both algorithms calculate the effective sample size (ESS) for the precision parameters κ_u and κ_v, as discussed in the lecture using the provided R-script ess.R. Calculate also the relative ESS, where you divide ESS by the running time needed for the algorithm. What is your conclusion when you compare both algorithms?

Exercise 5 (Performance)
Inspecting the acceptance rates, what do you guess might be a reason for the poor behaviour of the block sampler. In which range should the acceptance rates of the Metropolis-Hasting step ideally be? Do you have a proposal how to improve the sampler?

Exercise 6 (Comparison to INLA)
See the INLA documentation for an illustrated example of this analysis at:
http://www.r-inla.org/examples/volume-1/code-for-bym-example
The INLA example differs only in that different data and that an intercept term μ is used. Thus, to obtain results that can be directly compared with the MCMC analysis use

```
> g <- system.file("demodata/germany.graph", package="INLA")
> formula <- Y ~ f(region.struct, model="besag", graph=g,
+                     hyper=list(prec=list(param=c(1,0.01))), constr=F) +
+                 f(region, model="iid", hyper=list(prec=list(param=c(1,0.01)))) - 1
```

where you define `region` and `region.struct` using

```
 Oral<-cbind(Oral,region.struct=1:544, region=1:544)}
```

Compare the histograms of your MCMC-samples and the posterior marginals obtained by INLA for both precision parameters κ_u and κ_v, and five randomly chosen components of $\boldsymbol{u}$ and $\boldsymbol{v}$. Note, improved estimates of the posterior marginals for the precision parameters can be obtained by applying `inla.hyperpar(result)` on the original INLA results object `result`.

Exercise 7 (Interpretation of results)
Plot the posterior median of $\exp(\boldsymbol{u})$ (obtained from MCMC or INLA) for all regions using the function `map.landkreis()` provided in the R-package `spam` and interpret the obtained spatial pattern.

References

Rue, H. and Held, L. (2005). *Gaussian Markov Random Fields: Theory and Applications*, Chapman & Hall, London.

Oral presentations

Date	Exercise	Team
20.02.2014	1: Problem A1 and A2	Marius Møller Rokstad
	1: Problem A3	Ilmo Räisänen
	1: Problem B	Lars Kristian Steffensen, Shipra Sachdeva
	1: Problem C1 and C2	Tygve Bertelsen Wiig
27.02.2014	1: Problem C3	Henrik Vikøren, Edvard Hove
	1: Problem C4 and D1	Elise Landsem
	1: Problem D2	Mateusz Samiec
20.03.2014	2: 1 a, b	Tore Bredre
	2: 1c (GI)	Andrea Casati
	2: 1c (BL)	Gunnhild Hadersen Presthus, Harald Svandal Bø
	2: 2 (GI)	Ekaterina Fedorova, Beate Sildnes
27.03.2014	2: 2 (BL)	Pål Christie Ryalen
	2: 3	Odd Eirik Farestveit, Susanne Kjølen
	2: 4,5	Marius Fagerland, Tobias Bjormyr
	2: 6,7	Torgeir Rimstad, Kristoffer Berg
10.04.2014	3	Brandon Bergerud
		Kristoffer Kofoed Rødvei, Sverre Thommesen
		Kristin M. Drahus
		James Korley Attuquaye, Mireia Duaso