TMA4300 Computer Intensive Statistical Methods
Exercise 2, Spring 2016

Note: The solution to ALL exercises must be handed in no later than **13$^{\text{th}}$ of April 2016, 12:00 noon**.
Note 2: All "helper" files mentioned throughout the exercise text are available from the course webpage.

Preliminary steps: Please install the packages `fields`, `INLA` and `spam` by typing in the R-terminal

```
install.packages("fields")
install.packages("spam")
install.packages("INLA", repos="http://www.math.ntnu.no/inla/R/testing")
```

The goal is to carry out a spatial analysis on mortality rates of oral cavity cancer in males in Germany during a 5-year period, 1986–1990, for $n = 544$ districts. Observed counts are y_i, expected counts are E_i. Figure 1 shows standardised mortality rates (SMR) $\frac{y_i}{E_i}$.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
# load some libraries to generate nice map plots
library(fields)
library(colorspace)
col <- diverge_hcl(8)      # blue - red
# use a function provided by spam to plot the map together with the mortality rates
germany.plot(Oral$Y/Oral$E, col=col, legend=TRUE)
```

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. $\boldsymbol{v} \sim \mathcal{N}(\boldsymbol{0}, \kappa_v^{-1}\mathbf{I})$. (Note: An equivalent model would be $\boldsymbol{\eta} = \mu\boldsymbol{1} + \boldsymbol{u} + \boldsymbol{v}$, but would then require an additional sum to zero constraint on $\boldsymbol{u}$ to make the intercept μ identifiable.)

A common way to introduce a spatially correlated effect is to assume that neighbouring districts are more similar than distant districts. For this purpose, 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 (**?**) with density

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

$$= \kappa_u^{(n-1)/2} \exp\left(-\frac{\kappa_u}{2} \boldsymbol{u}^T \mathbf{R} \boldsymbol{u}\right). \tag{3}$$

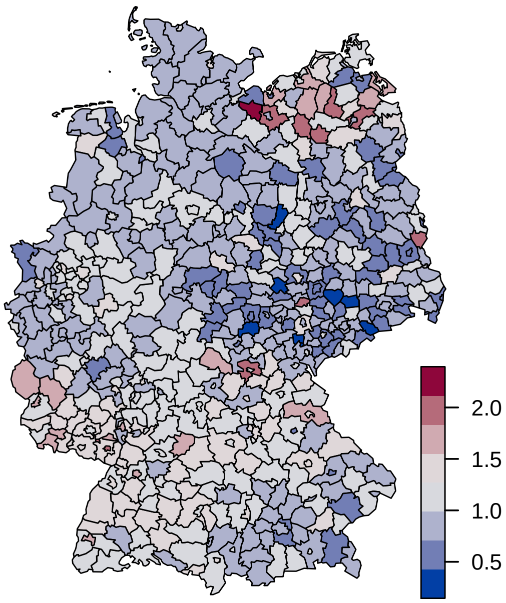

Figure 1: Standardised mortality rates (SMR) y_i/E_i.

The sum in (2) goes over all pairs of neighbouring districts $i \sim j$ which is defined by the geographical map. The neighbourhood structure is consequently defined in the structure matrix $\mathbf{R}$ where

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

Here, n_i denotes the number of neighbouring districts of district i. The distribution of $\boldsymbol{\eta}$, conditional on the spatial component $\boldsymbol{u}$ and κ_v, is

$$\boldsymbol{\eta} \mid \boldsymbol{u}, \kappa_v \sim \mathcal{N}(\boldsymbol{u}, \kappa_v^{-1}\mathbf{I}). \tag{4}$$

The precision terms κ_v and κ_u are assigned gamma prior distributions

$$\kappa_u \sim \text{Gamma}(\alpha_u, \beta_u),$$
$$\kappa_v \sim \text{Gamma}(\alpha_v, \beta_v).$$

where $\text{Gamma}(\alpha, \beta)$ denotes the gamma density with shape parameter α and rate parameter β. Here, we will use $\alpha_u = \alpha_v = 1$ and $\beta_u = \beta_v = 0.01$.

To analyse the data and its underlying spatial structure we will implement a Gibbs sampler with individual parameter updates based on the full conditional distributions. One parameter, $\boldsymbol{\eta}$, does not have a "standard" full conditional density and will require a Metropolis-Hastings step.

Exercise 1 (Derivations [*no programming needed*])

(a) Derive the posterior distribution $\mathrm{p}(\boldsymbol{\eta}, \boldsymbol{u}, \kappa_u, \kappa_v \mid \boldsymbol{y})$ and show that it is proportional to

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

(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, here a Gaussian, approximate the function

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

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

$$\widetilde{f}(\eta_i) = a_i + b_i\eta_i - \frac{1}{2}c_i\eta_i^2, \tag{5}$$

with $a_i = E_i\exp(z_i)\cdot(z_i - \frac{1}{2}z_i^2 - 1)$, $b_i = y_i + E_i\exp(z_i)\cdot(z_i-1)$ and $c_i = E_i\exp(z_i)$.

(c)

- 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 can be written as

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

and consequently be approximated using (5) by a normal density written in canonical form as

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

where $\boldsymbol{b} = (b_1, \ldots, b_n)^T$, $\boldsymbol{c} = (c_1, \ldots, c_n)^T$. Of note, the first argument in the condition of $q(\cdot \mid \cdot)$, denoted by $\boldsymbol{z}$, represents the vector at which the Taylor approximation is computed.

Exercise 2 (Implementation of the MCMC sampler)

After a suitable burn-in period, a posterior sample size of at least $M = 50000$ (depending on the initial values) is recommended. Let m index the current iteration. The steps required for a single iteration are:

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 $q(\boldsymbol{\eta}^{\star} \mid \boldsymbol{z}, \boldsymbol{y}, \boldsymbol{u}^{(m)}, \kappa_u^{(m)}, \kappa_v^{(m)})$, with approximation around $\boldsymbol{z} = \boldsymbol{\eta}^{(m-1)}$.
5. Set $\boldsymbol{\eta}^{(m)} = \boldsymbol{\eta}^{\star}$ with probability

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

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

Computational details:

- The matrix $\mathbf{R}$, also called `R`, in (3) is available from the course webpage and can be loaded using

  ```
  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

  Please see their help pages in `R` for more information.
- To evaluate a multivariate normal density parameterised in canonical form you can use the provided function `dmvnorm.canonical(.)` implemented in the file `dmvnorm.R`.

While running the samplers keep track of the acceptance rates for the Metropolis-Hastings steps, i.e. print for example every 500 iterations the current percentage of accepted proposals. Use the function `system.time()` or `Sys.time()` to record how long the sampler needs to generate the M samples.

Exercise 3 (Convergence diagnostics)
Obtain the following diagnostic summaries for the precision parameters κ_u, κ_v, and exemplary for three randomly chosen components of $\boldsymbol{u}$ and $\boldsymbol{v}$.

(a) Trace plots.

(b) Autocorrelation plots.

(c) Use the function `geweke.diag()` of the `R`-package `coda` to check the Markov chains for convergence.

What is your conclusion?

Exercise 4 (Effective sample size)

- 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. Interpret the obtained values.
- Compute also the relative ESS, where you divide the ESS by the running time of the MCMC algorithm. For which purpose could this quantity be interesting?

Exercise 5 (Interpretation of results)
Plot the posterior median of $\exp(\boldsymbol{u})$ for all regions using the function `germany.plot()` provided in the R-package `spam` (see page 1 of this exercise) and interpret the obtained spatial pattern.

Exercise 6 (Comparison to INLA and inclusion of covariate information)

a) Implement the same model using `R-INLA`. Be thereby careful to use the same priors for κ_u and κ_v. The latent Gaussian model given in (2), which defines the spatial structure, is termed `besag` in `R-INLA` and requires the path to the geographical map (graph) of Germany assigned to the argument `graph`. You get the path to the Germany map as

```
g <- system.file("demodata/germany.graph", package="INLA")
```

Unstructured random effects are defined in the latent model `iid`. Please see the documentation provided on `www.r-inla.org` and the lecture notes for details. Within the `f(.)` functions used in the INLA formula object you will besides others need the arguments `hyper` and `constr` to define the equivalent model as you implemented using MCMC.

Compare the histograms of your MCMC-samples and the posterior marginals obtained by INLA for both precision parameters κ_u and κ_v, and three 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`.

b) Smoking is besides alcohol consumption regarded as one of the main risk factors of oral cavity cancer. There is covariate information regarding smoking consumption available in the file `smoking.dat`. You can read it in INLA using

```
  smoking = read.table("covariate.dat")
```

and add to the data.frame `Oral` by

```
  Oral["smoking"] = smoking
```

Extend the INLA model formulation of part a) to incorporate this covariate:

1) as a linear effect.
2) as a non-linear function using a random walk of second order (`rw2`). (You can use the default prior parameters of INLA for the gamma prior of the corresponding precision parameter.)

Compare extension 1) and 2) to the original model formulation without covariate information in part a) by means of the deviance information criterion (DIC). Interpret the results. Plot also the posterior median within 95% credible intervals of the non-linear covariate effect (2nd covariate model) and relate this to the obtained DIC values.

Oral presentations

Date	Exercise	Team
15.02.2016	1: Problem A1 and A2	Kristian Aga, Moritz Klaus Bogs
	1: Problem A3	Tomasz Kusmierczyk, Eliezer de Souza da Silva
	1: Problem B	John Darkwah, Audun Sektnan
22.02.2016	1: Problem C1 and C2	Fredrik Lundquist, Ole Bernhard Forberg
	1: Problem C3 and C4	Yingzi Jin, Himanshu Srivastav
	1: Problem D1 and D2	Eivind Berg Fosse, Espen Høegh Sørum
04.04.2016	2: 1a, b	Eyvind Thommesen
	2: 1c	Negar Olfati, Abdolreza Sabzi Shahrebabaki
	2: 2	Arnaud Vacogne, Truls Brubak
11.04.2016	2: 3	Sandra Sæther, Dechasa Obsi Gudeta
	2: 4, 5	Martina Hall, Jorid Ødegrd
	2: 6a, b	Øyvind Øksnes Dalheim, Anna Swider
26.04.2016	3	Gunhild Elisabet Berget, Ingeborg Gullikstad Hem
		Andreas Malmgård, Haris Fawad
		Filip Emil Schjerven, Rafael Schwarzenegger
		Inger Ellingsen