Project2

Gaussian random field with application of INLA

Yaolin Ge, Florian Beiser

Contents

Part I Multivariate normal distribution **1**
Solution to Part I 2

Part II Gaussian random fields and Kriging **8**
2.1 Simulation 9
2.2 Paramter estimation 10
2.3 Kriging 12

Part III Integrated nested Laplace Approximations (INLA) **13**
3.1 Simple Linear Regression 13
3.2 GLMM with random effects 15

Part I Multivariate normal distribution

Let $\boldsymbol{x} = (x_1, \ldots, x_n), n = 100$ be multivariate normal distributed with $E(x_i) = 0, Var(x_i) = 1$, and $Corr(x_i, x_j) = e^{-0.1|i-j|}$

a) Compute and image the covariance matrix $\boldsymbol{\Sigma}$ of $\boldsymbol{x}$

b) Find the lower Cholesky factor $\boldsymbol{L}$, such that $\boldsymbol{L}\boldsymbol{L}^T = \boldsymbol{\Sigma}$, of this covariance matrix, and image.

c) Sample $\boldsymbol{x} = \boldsymbol{L}\boldsymbol{z}$, where $\boldsymbol{z}$ is a length n random vector of independent standard normal variables. Plot the sample.

d) Find the precision matrix $\boldsymbol{Q}$ of the covariance matrix, and compute the lower Cholesky factor $\boldsymbol{L}_Q$, such that $\boldsymbol{L}_Q\boldsymbol{L}_Q^T = \boldsymbol{Q}$, of this matrix. Image these matrices and compare them to the images obtained in a) and b)

e) Sample $\boldsymbol{x}$ by solving $\boldsymbol{L}_Q^T\boldsymbol{x} = \boldsymbol{z}$, where $\boldsymbol{z}$ is a length n random vector of independen standard normal variables. Plot the sample.

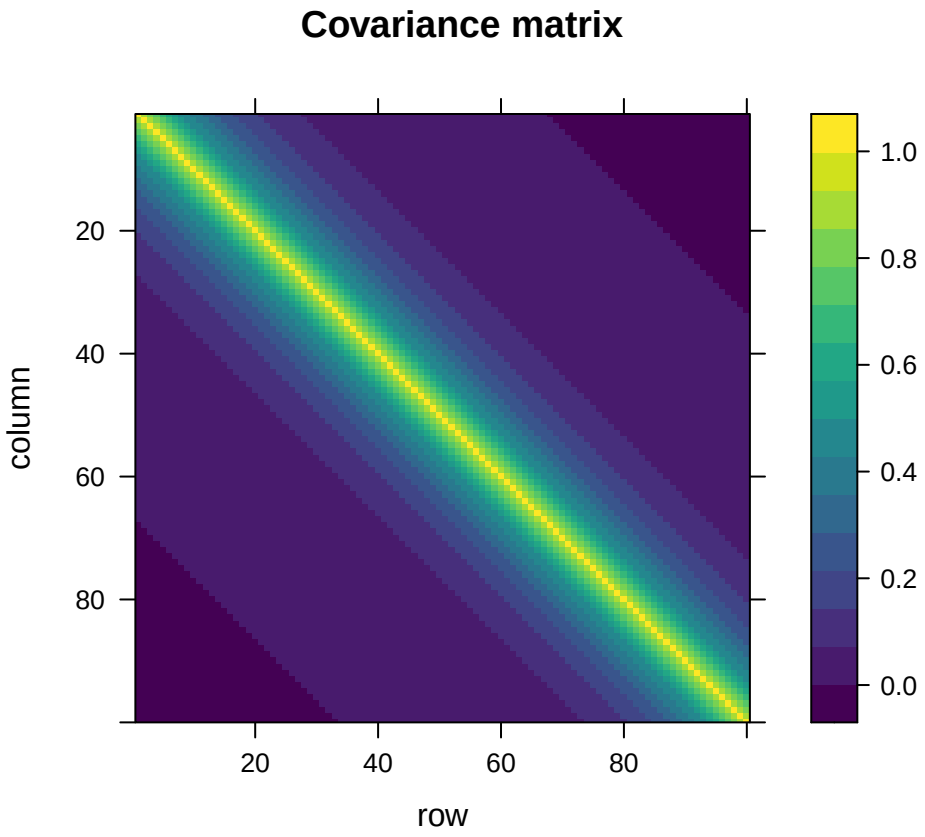
f) Permute the ordering of variables in $\boldsymbol{x}$, and redo the exercises.

Solution to Part I

a)

Given that $\mathbf{\Sigma} = e^{-0.1|i-j|}$. The covariance matrix can be expressed as follows:

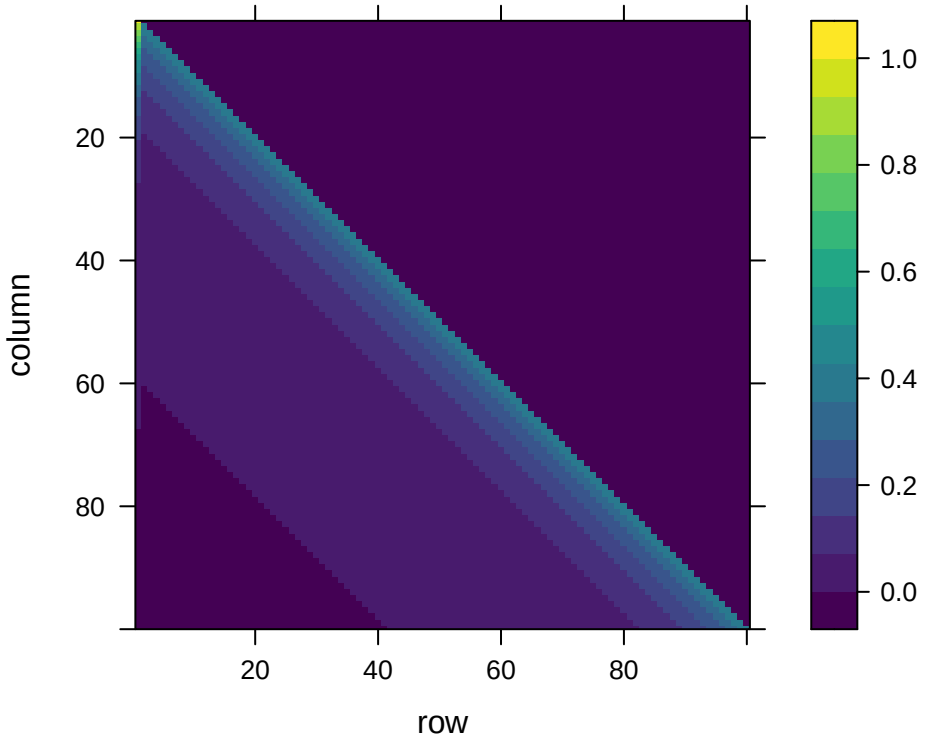
$$\Sigma = \begin{pmatrix} 1 & e^{-0.1h_{12}} & \dots & e^{-0.1h_{1n}} \\ e^{-0.1h_{21}} & 1 & \dots & e^{-0.1h_{2n}} \\ \vdots & \vdots & \ddots & \vdots \\ e^{-0.1h_{n1}} & e^{-0.1h_{n2}} & \dots & 1 \end{pmatrix}$$



The covariance has the biggest entries on and around the main diagonal. However, it is fully dense.

b)

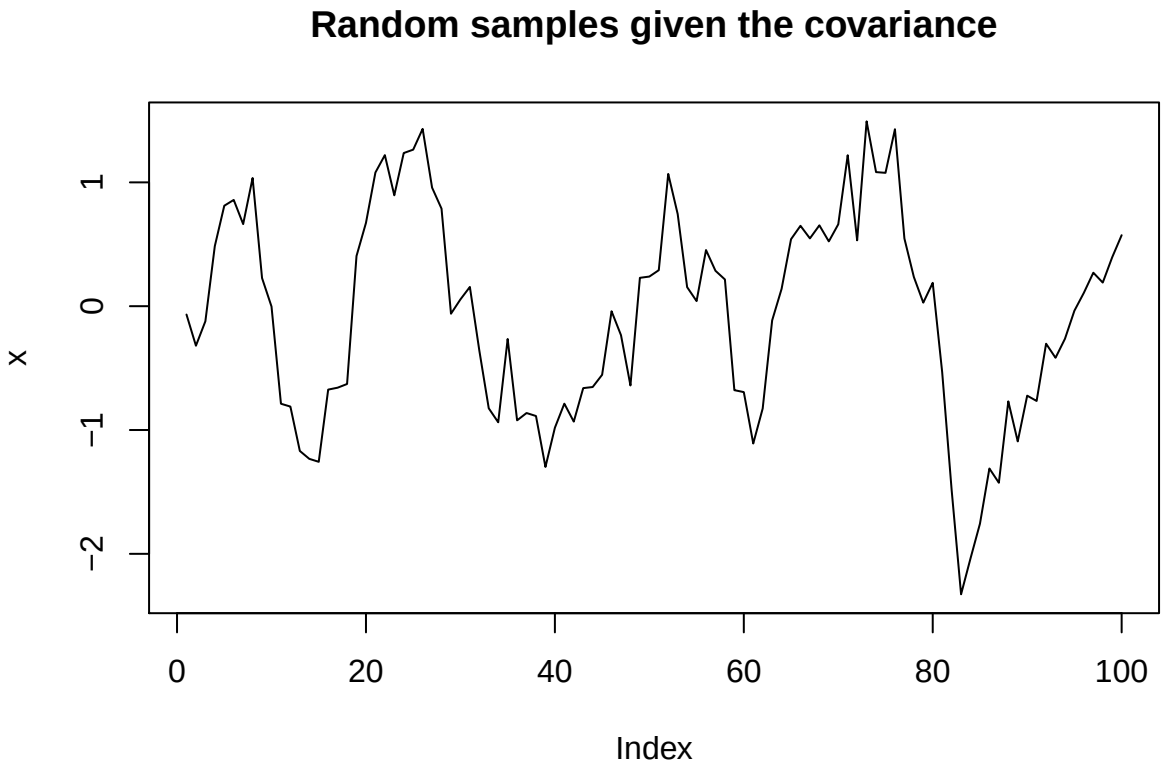
According to the cholesky decomposition rule, $\boldsymbol{L}$ is the lower triangular matrix for $\mathbf{\Sigma}$, it can be easily computed from R using `L = chol(Sigma)`. It is then plotted as below.



The decomposed matrix shows a similar structure as the original covariance matrix, such that the biggest values are on and close to the main diagonal, but naturally the matrix is lower triangular.

c)

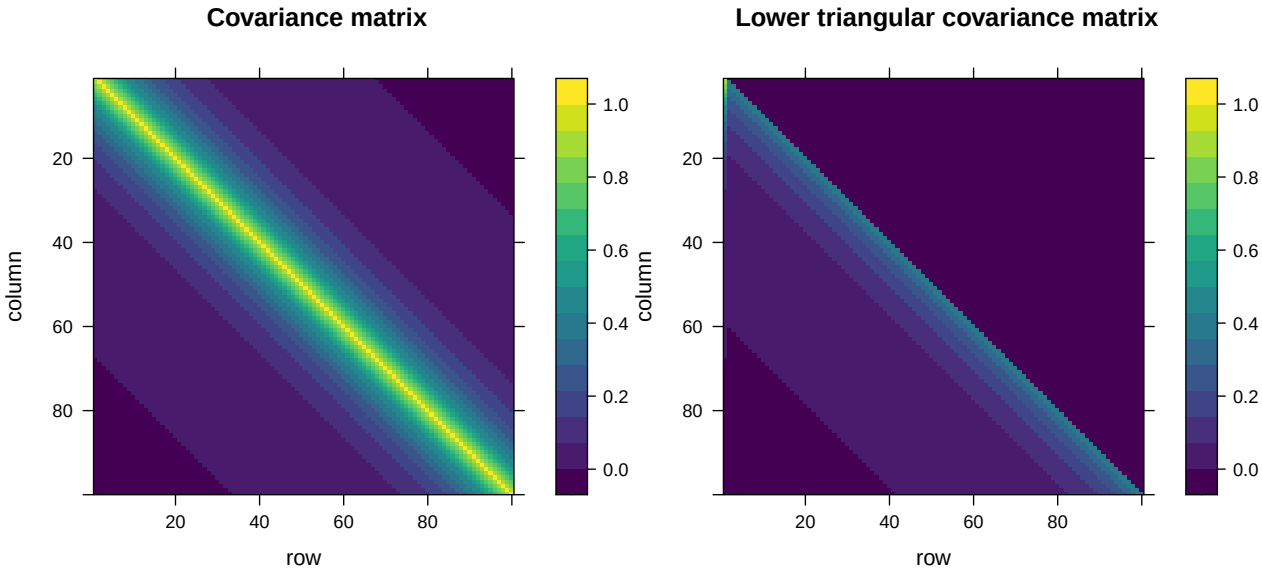
Sample using $\boldsymbol{x} = \boldsymbol{L}\boldsymbol{z}$ transforms the zero-mean, standard normal random variales to the random variables with the desired covariance matrix.

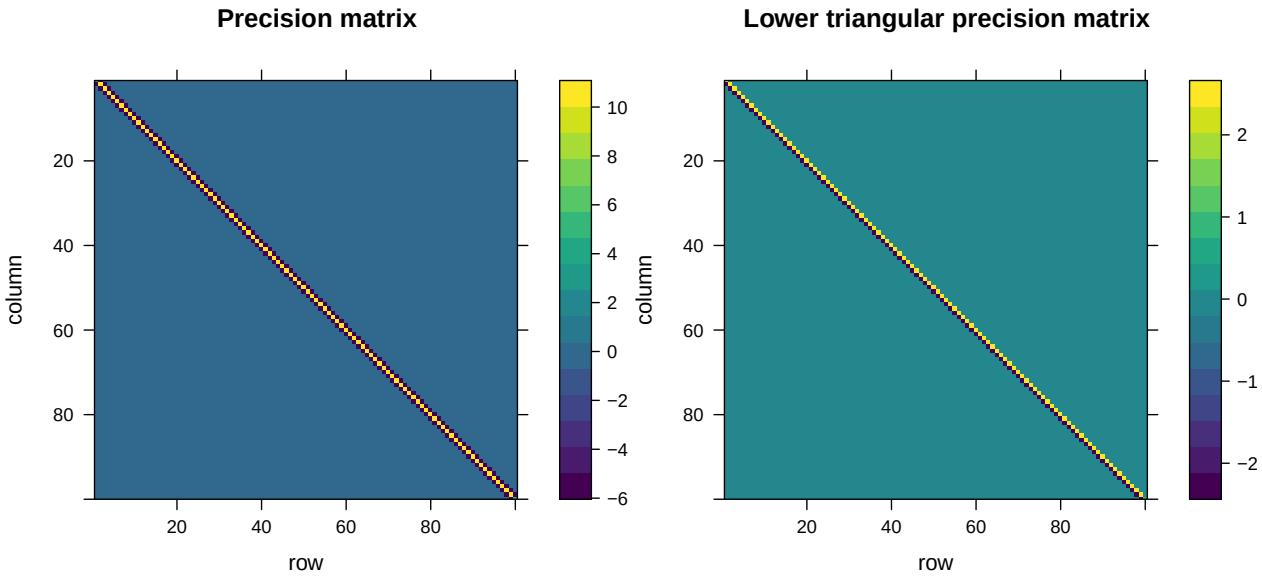


For the given covariance matrix, a sample of x obtains values around 0 and has a quite wriggling structure.

d)

The precision matrix $\boldsymbol{Q}$ is the inverse of the covariance matrix $\boldsymbol{\Sigma}$, it is computed using `Q = solve(Sigma)` in R. The three matrices are thereby depicted as follows. Since the covariance matrix is not singular, given that the it belongs to the Matern family, thus it is analytically guaranteed to have positive definite property. Therefore, both precision matrix and the lower triangular precision matrix exist.

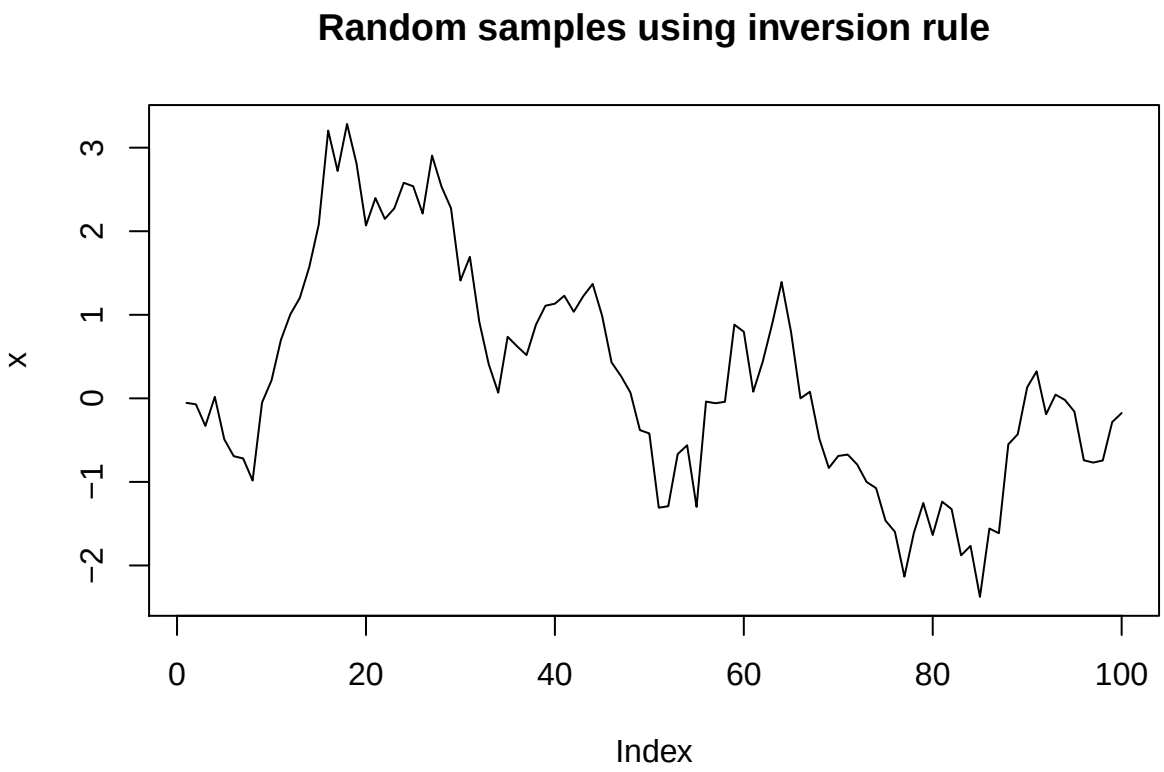




The precision matrix is even more diagonal dominant as the covariance matrix, i.e. values are further than the first off-diagonal are numerically negligible, which makes the precision matrix much more sparse. The same holds for the Cholesky decomposition of the precision matrix.

e)

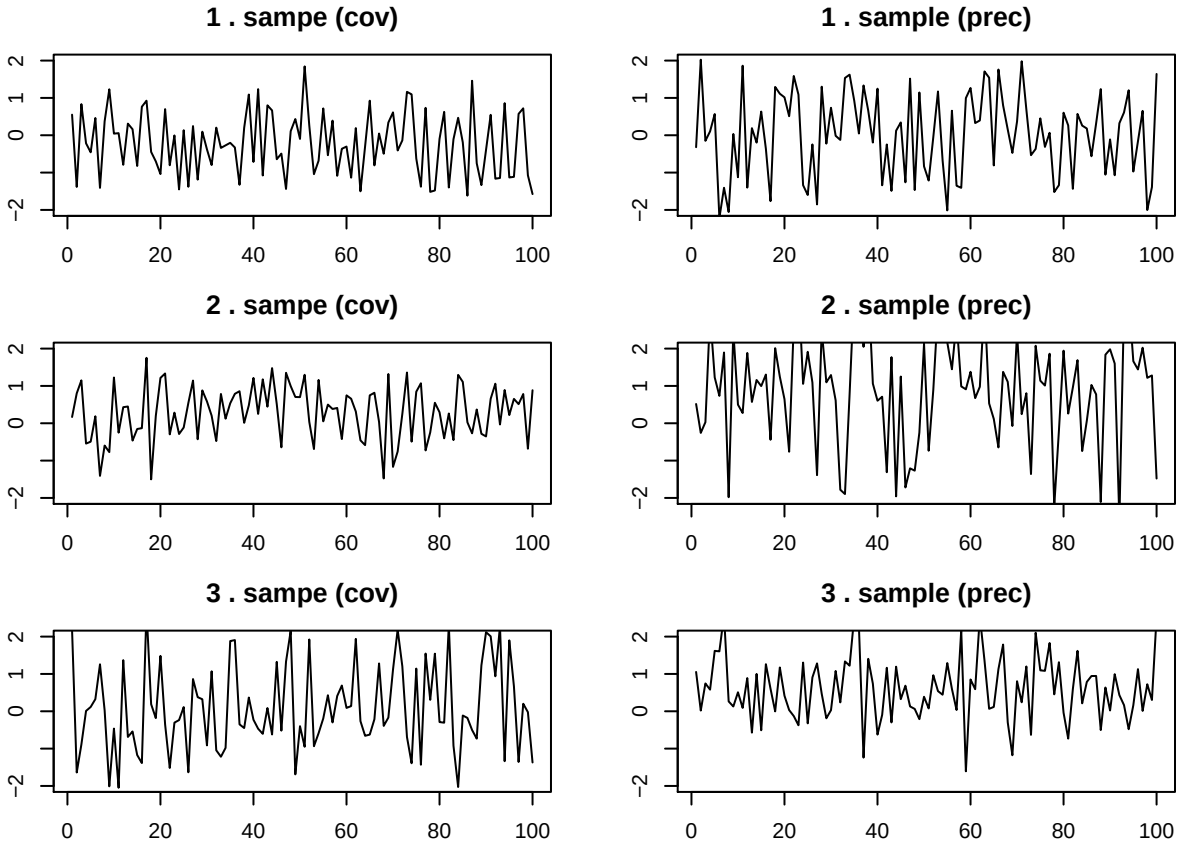
Similarly, the expected random samples can be generated using the inversion of the above formula, thus $\boldsymbol{L}_Q^T \boldsymbol{x} = \boldsymbol{z}$



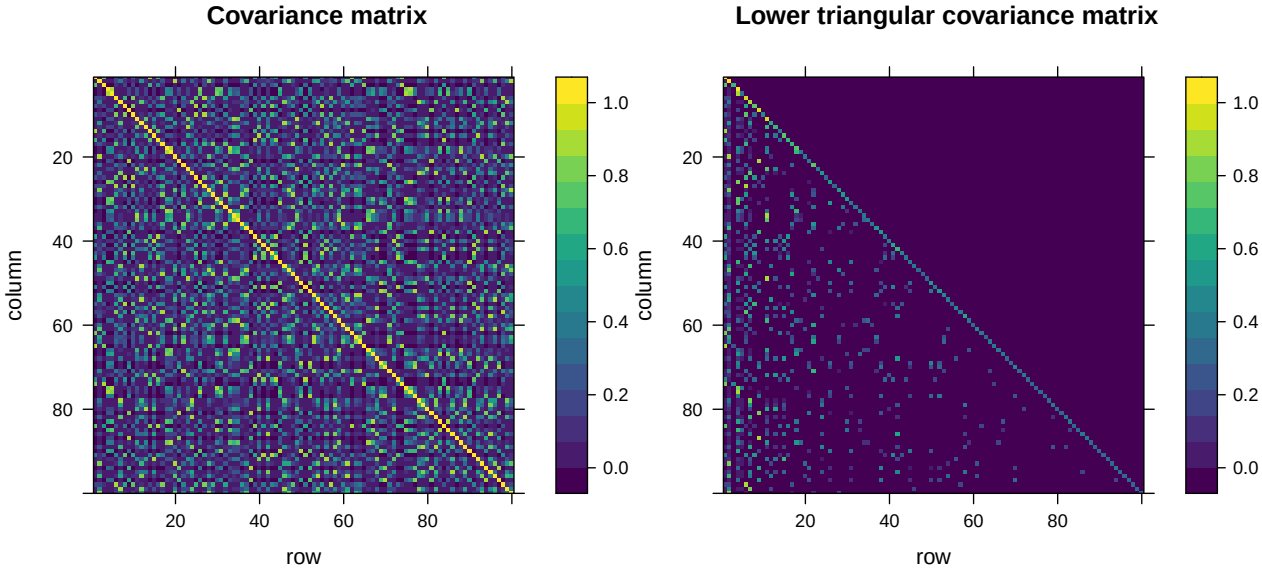
The sample shows the same characterisitics as samples which are calculated directly.

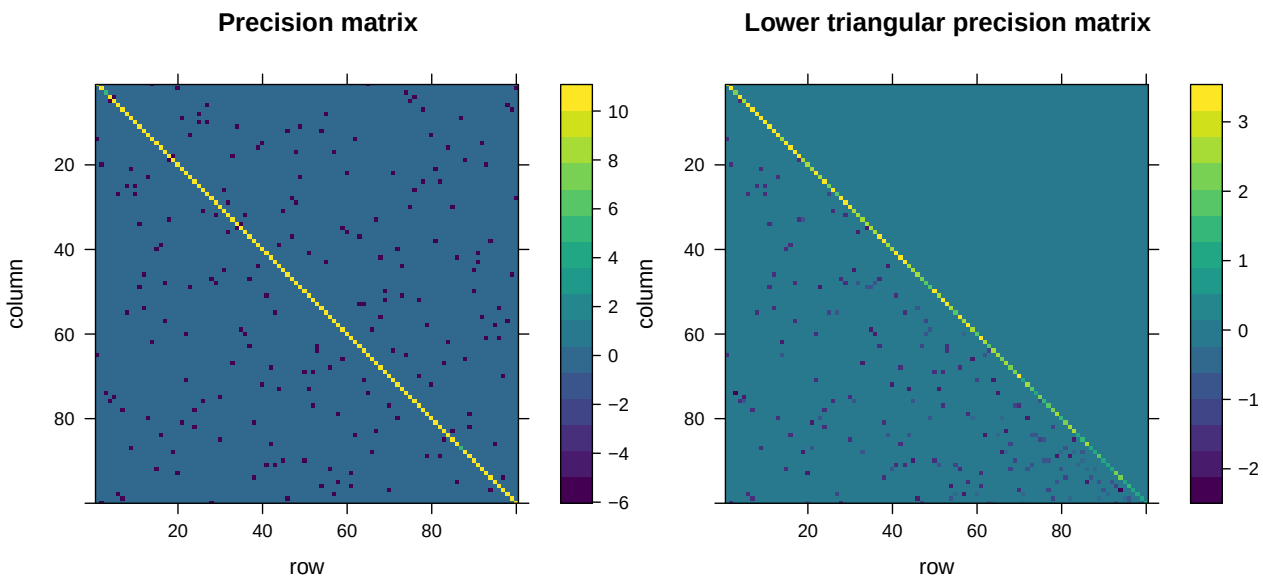
f)

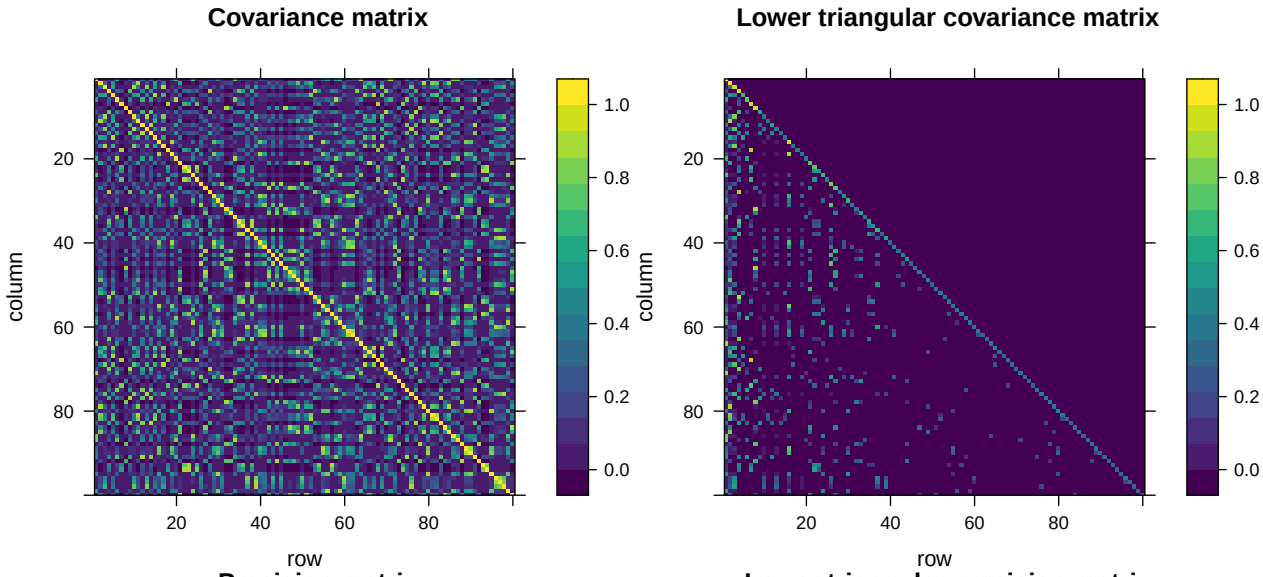
Permute $\boldsymbol{x}$ to make randomise the ordering of the grid, the associated covariance matrix can be thereby modified in a sparse way.

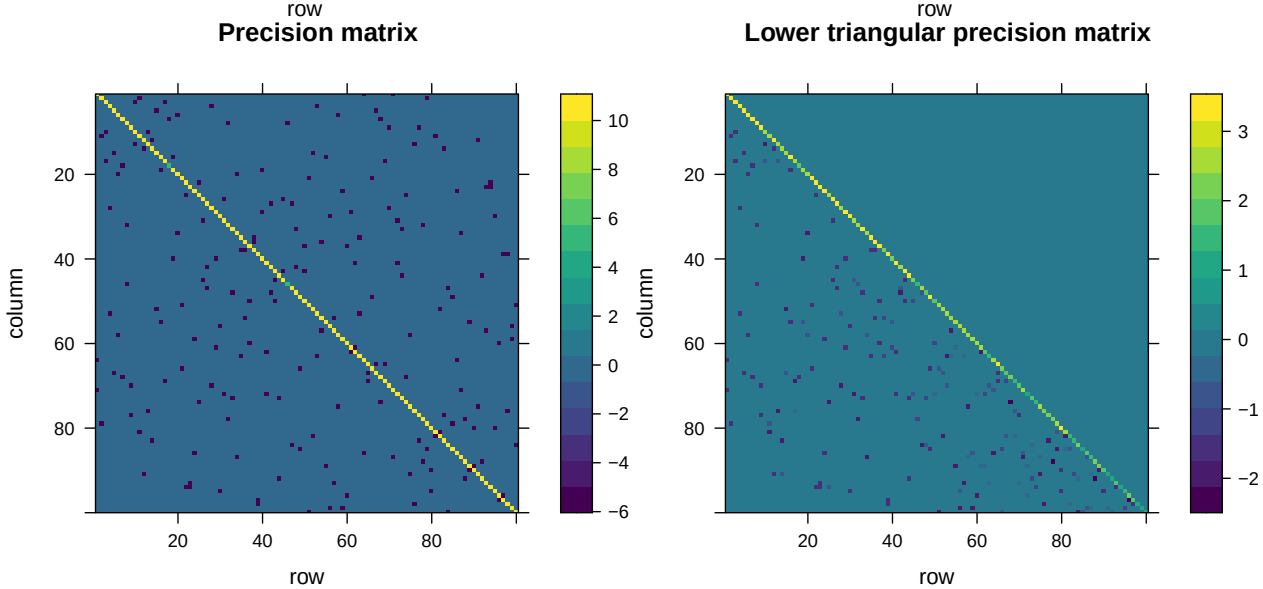


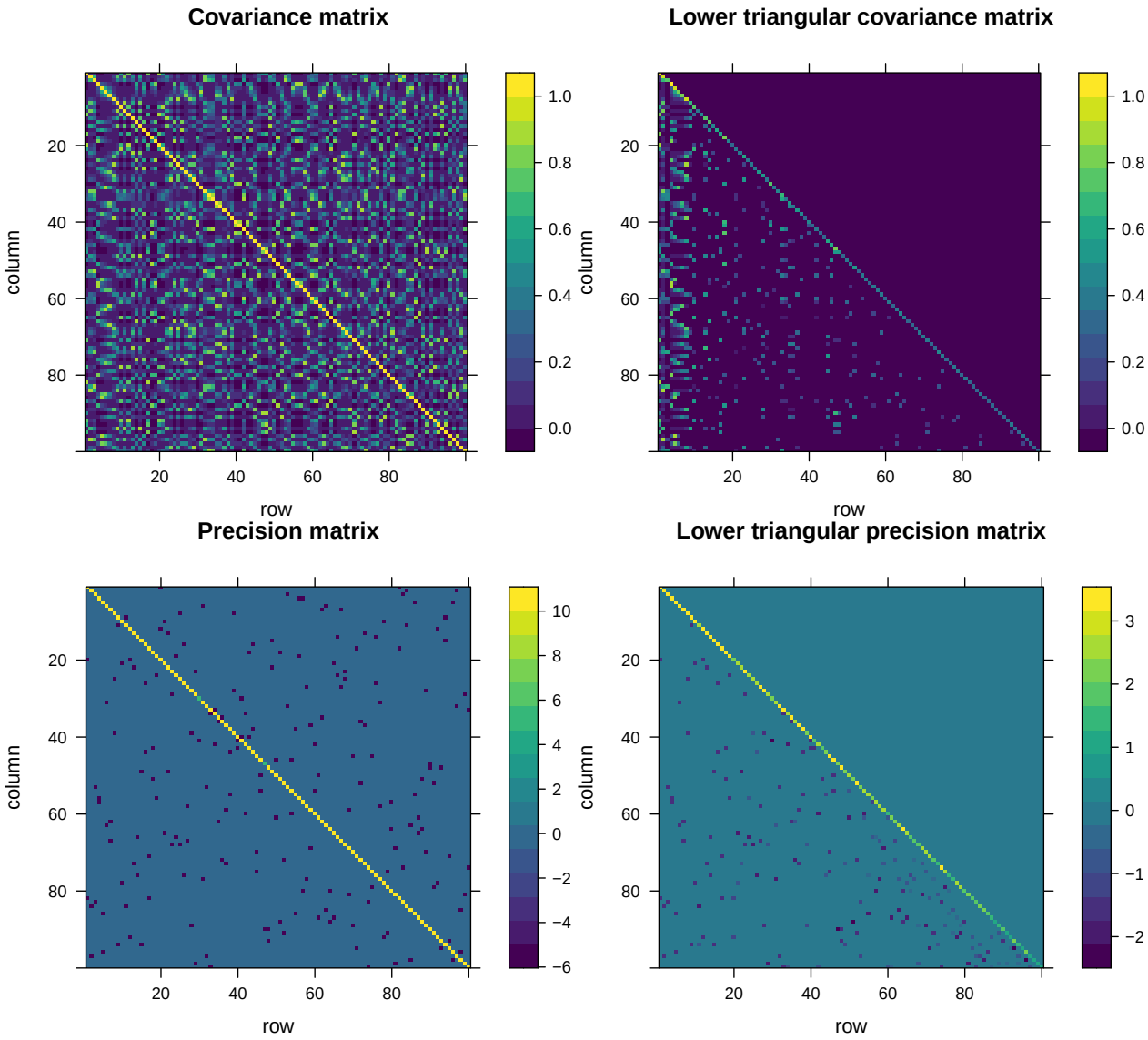
All samples show same characterisics (as far as we can tell from a single realisation).




Precision matrix
Lower triangular precision matrix
column
row
20
40
60
80
10
8
6
4
2
0
−2
−4
−6
3
1
−1


Covariance matrix
Lower triangular covariance matrix
column
row
20
40
60
80
1.0
0.8
0.6
0.4
0.2
0.0


Precision matrix
Lower triangular precision matrix
column
row
20
40
60
80
10
8
6
4
2
0
−2
−4
−6
3
1
−1



In contrast to similar sample results, the permutation have a significant influence on the structure of the matrices. The covarinace matrix looses its clear pattern, which leads to a non-sparse precision matrix!

Part II Gaussian random fields and Kriging

The purpose of this computer exercise is to give an introduction to parameter estimation and kriging for Gaussian random field models for spatial data.

We assume the following observation model on the unit square:

$$y(\boldsymbol{s}_j) = x(\boldsymbol{s}_j) + \epsilon_j, \quad j = 1, \ldots, N,$$

where $\epsilon_j \;\; N(0, \tau^2)$ are independent measurement noise terms. Further, consider a Matérn covariance function for the Gaussian random field $\boldsymbol{x(s)}$:

$$Cov(x(\boldsymbol{s}_i), x(\boldsymbol{s}_j)) = \Sigma_{i,j} = \sigma^2(1 + \phi h)\exp(-\phi h),$$

where h denotes the Euclidean distance between the two sites $\boldsymbol{s}_i$ and $\boldsymbol{s}_j$.

We assume the mean increases with east and north coordinates as follows: $\mu_j = \alpha((s_{j1} - 0.5) + (s_{j2} - 0.5))$, for site $\boldsymbol{s}_j = (s_{j1}, s_{j2})$ on the unit square.

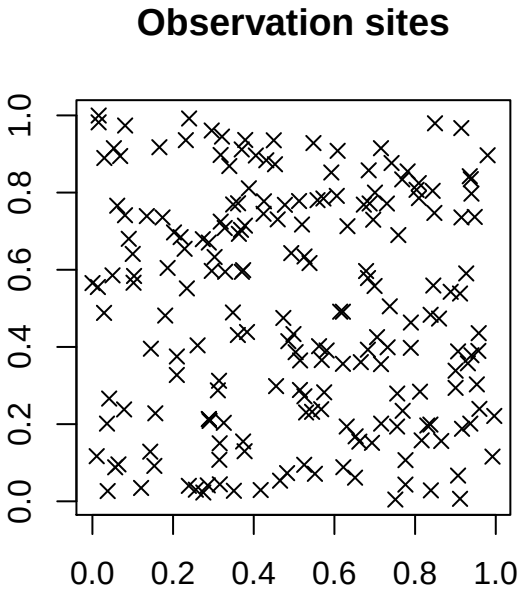
2.1 Simulation

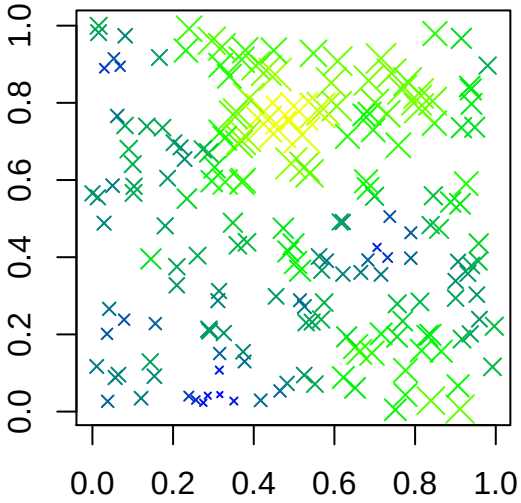
Simulate $N = 200$ random sites in the unit square and plot them. Form the covariance matrix using $\sigma = 1, \phi = 10, \tau = 0.05$. Take its Cholesky decomposition and simulate dependent zero-mean Gaussian data variables, then add the mean using $\alpha = 1$. Plot your observations.

The true mean of the field is expressed as

$$\mu_i = \alpha((s_{i1} - 0.5) + (s_{i2} - 0.5))$$

where s_{i1}, s_{i2} are the location from east and north direction in the grid.





In the observation plot, bigger crosses and brighter colors indicate higher observed values. Then we can see tendency towards greater values towards the upper right corner (as we would expect from the bias).

2.2 Paramter estimation

We will now use the simulated data to estimate the model parameters $\alpha, \sigma^2, \tau^2, \phi$ using maximum likelihood estimation. Iterate between the update for the mean parameter, and updating the covariance parameters. Monitor the likelihood function at each step of the algorithm to check convergence.

The mean of the field is modelled by $p(\boldsymbol{x})$ and the imperfect information $\boldsymbol{y} = (y_1, \ldots, y_m)$ conditional on $\boldsymbol{x}$ can be modelled by $p(\boldsymbol{y}|\boldsymbol{x})$, which can be expressed as follows:

$$p(\boldsymbol{x}) = N(\mu, \boldsymbol{\Sigma}), \quad p(\boldsymbol{y}|\boldsymbol{x}) = N(\boldsymbol{x}, \boldsymbol{\Sigma} + \boldsymbol{T})$$

Therefore, the marginal likelihood of the data is

$$p(\boldsymbol{y}) = N(\boldsymbol{\mu}, \boldsymbol{C}), \quad \boldsymbol{C} = \boldsymbol{\Sigma} + \boldsymbol{T}, \quad \boldsymbol{T} = \tau^2 \boldsymbol{I}$$

The log-likelihood as a function of α and unknown fixed nuisance parameters $\boldsymbol{\theta} = (\sigma, \phi, \tau)$ in the prior covariance matrix $\boldsymbol{\Sigma} = \boldsymbol{\Sigma}(\boldsymbol{\theta})$, and/or the likelihood noise matrix $\boldsymbol{T} = \boldsymbol{T}(\boldsymbol{\theta})$ becomes

$$l(\boldsymbol{\theta}, \boldsymbol{\alpha}) = -\frac{N}{2}\log(2\pi) - \frac{1}{2}\log|\boldsymbol{C}| - \frac{1}{2}(\boldsymbol{y} - \boldsymbol{mu}(\boldsymbol{\alpha}))^T \boldsymbol{C}^{-1}(\boldsymbol{y} - \boldsymbol{mu}(\boldsymbol{\alpha}))$$

The MLEs of $\boldsymbol{\alpha}$ and $\boldsymbol{\theta}$ are obtained by

$$(\hat{\boldsymbol{\alpha}}, \hat{\boldsymbol{\theta}}) = \arg\max_{\boldsymbol{\alpha}, \boldsymbol{\theta}} \{l(\boldsymbol{\alpha}, \boldsymbol{\theta})\}$$

For fixed $\boldsymbol{\theta}$, the MLEs of $\boldsymbol{\beta}$ can be determined analytically.

$$\hat{\boldsymbol{\alpha}} = (\boldsymbol{\mu}^T \boldsymbol{C}^{-1} \boldsymbol{\mu})^{-1} \boldsymbol{\mu}^T \boldsymbol{C}^{-1} \boldsymbol{y}$$

Whereas for fixed $\boldsymbol{\alpha}$, the MLE of nuisance parameters $\boldsymbol{\theta}$ can be obtained by numerical maximization. Let $\boldsymbol{Z} = \boldsymbol{y} - \boldsymbol{\mu}(\boldsymbol{\alpha})$, and $\boldsymbol{Q} = \boldsymbol{C}^{-1}$. For each component of $\boldsymbol{\theta}_r$, $r = 1, \ldots, d$, in this case, θ has 3 components (σ, η, τ). The score of the log-likelihood becomes

$$\frac{dl}{d\theta_r} = -\frac{1}{2}\text{trace}(\boldsymbol{Q}\frac{d\boldsymbol{C}}{d\theta_r}) + \frac{1}{2}\boldsymbol{z}^T \boldsymbol{Q}\frac{d\boldsymbol{C}}{d\theta_r}\boldsymbol{Q}\boldsymbol{Z}$$

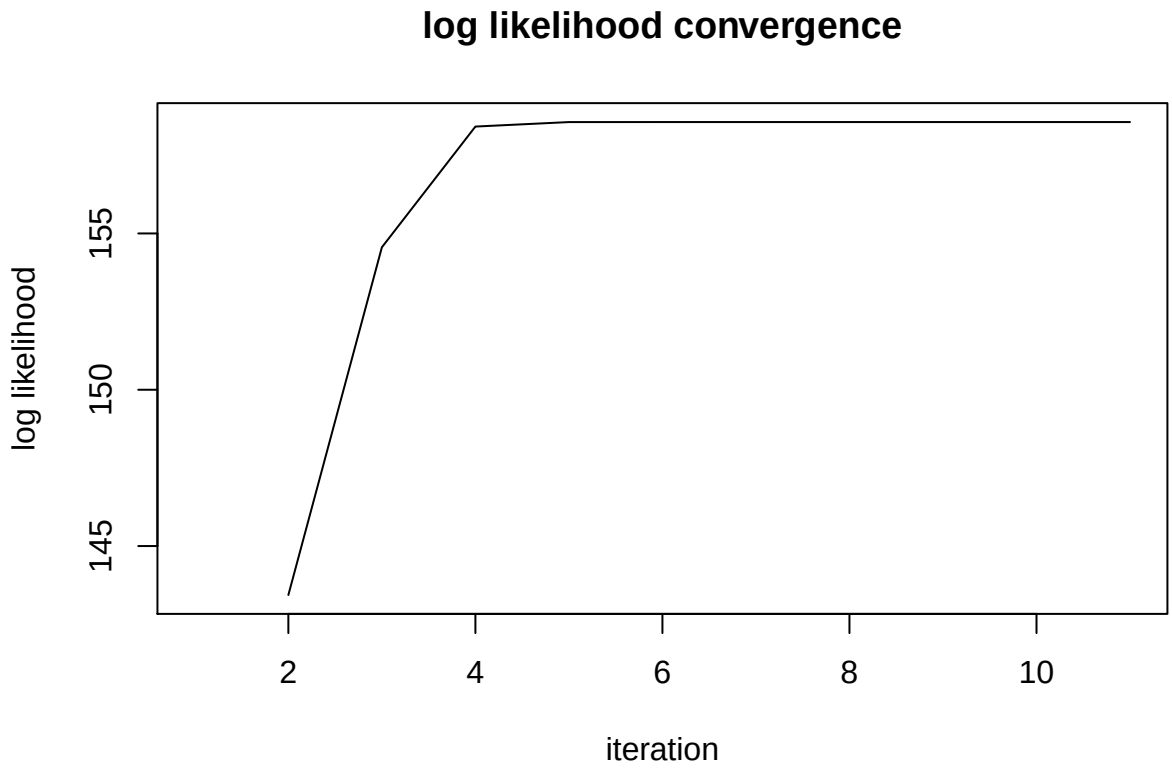
The above mentioned score can be solved iteratively using Fisher scoring algorithm. To achieve the numerical stability of the algorithm, the expected Hessian is applied, which is

$$E(\frac{d^2 l}{d\theta_r d\theta_{\bar{r}}}) = -\frac{1}{2}\text{trace}(\boldsymbol{Q}\frac{d\boldsymbol{C}}{d\theta_{\bar{r}}}\boldsymbol{Q}\frac{d\boldsymbol{C}}{d\theta_r})$$

The pseudo code for the Fisher scoring algorithm can then be expressed as follows:

Data: initial α_0, θ_0
Result: Converged $\hat{\alpha}$, $\hat{\theta}$
while *not converged* **do**
  $\boldsymbol{C} = \boldsymbol{C}(\boldsymbol{\theta}^b)$;
  $\boldsymbol{\alpha}^{b+1} = [\boldsymbol{\mu}^T \boldsymbol{C}^{-1} \boldsymbol{\mu}]^{-1} \boldsymbol{\mu}^T \boldsymbol{C}^{-1} \boldsymbol{y}$;
  $\boldsymbol{Z} = \boldsymbol{y} - \boldsymbol{mu}(\boldsymbol{\alpha}^{\boldsymbol{b+1}})$;
  $\boldsymbol{Q} = \boldsymbol{C}^{-1}$;
  $\boldsymbol{dC}_r = \frac{d\boldsymbol{C}(\boldsymbol{\theta}^b)}{d\theta_r}, \quad r = 1, \ldots, 3$;
  $\boldsymbol{dl}_r = \frac{dl}{d\theta_r} = -\frac{1}{2}\text{trace}(\boldsymbol{QdC}_r) + \frac{1}{2}\boldsymbol{z}^T \boldsymbol{QdC}_r \boldsymbol{Qz}, \quad r = 1, \ldots, 3;$
  $\boldsymbol{Hess}_{r,s} = E(\frac{d^2 l}{d\theta_r d\theta_r}) = -\frac{1}{2}\text{trace}(\boldsymbol{QdC}_r \boldsymbol{QdC}_s), \quad r, s = 1, \ldots, 3;$
  $\boldsymbol{\theta}^{b+1} = \boldsymbol{\theta}^b - \boldsymbol{Hess}^{-1} \boldsymbol{dl}$;
  $b = b + 1$
end

```
## [1] "The parameter estimation coverges after  11 iterations"
## [1] "The estimate for sigma (true value  1.0) is  0.8686"
## [1] "The estimate for phi   (true value 10.0) is  11.5657"
## [1] "The estimate for tau   (true value 0.01) is  0.031"
## [1] "The estimate for alpha (true value  1.0) is  1.1755"
```



We see that the parameter estimation comes closer to the true values as the initial guess, but does not fully hit them - however, in a probabilistic framework we cannot expect more. Whereas the likelihood converges very quickly.

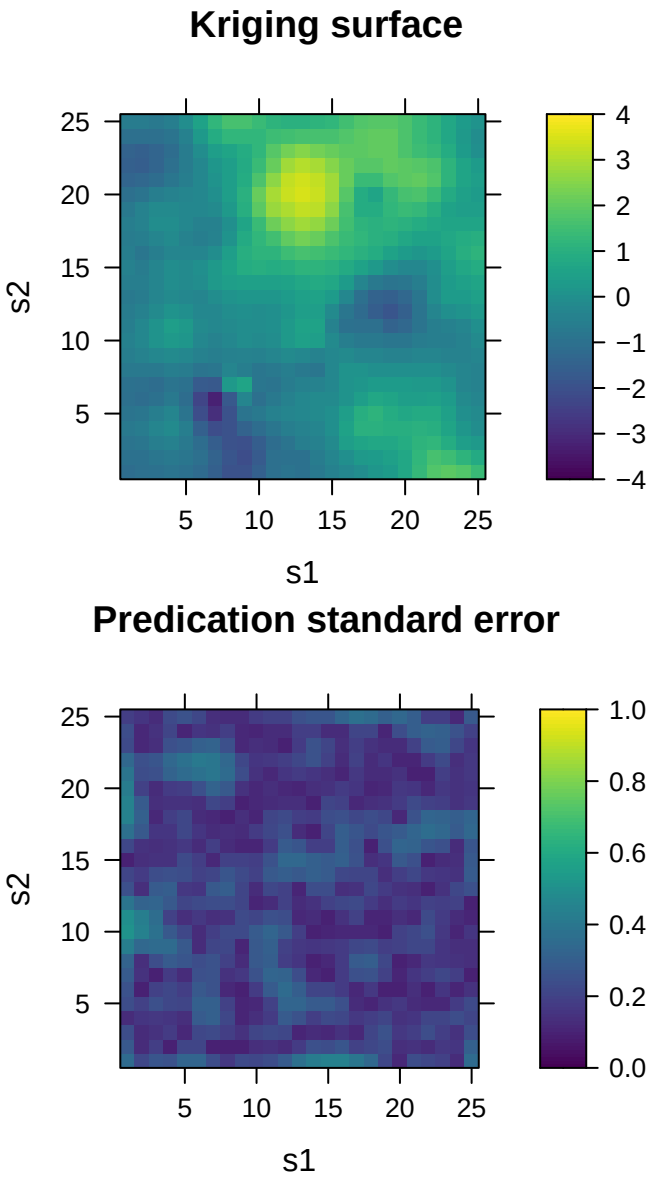
2.3 Kriging

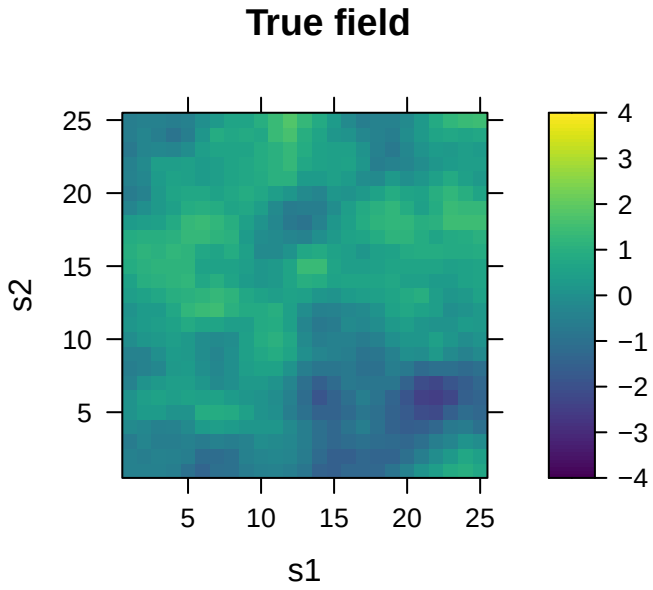
We will now use the estimated model parameters to perform kriging prediction. Predict variables $x(s)$, where predictions sites lie on a regular grid of size 25x25 for the unit square. Visualize the Kriging surface and the prediction standard error. Compare with the true field. The expression for the kriging can be shown as below:

$$\hat{Y}_0 = E(Y_0|\boldsymbol{Y}) = \boldsymbol{X}_0\hat{\beta} + \boldsymbol{C}_{0,.}\boldsymbol{C}^{-1}(\boldsymbol{Y} - \boldsymbol{X}\hat{\boldsymbol{\beta}})$$

$$Var(Y_0|\boldsymbol{Y}) = \boldsymbol{C}_0 - \boldsymbol{C}_{0,.}\boldsymbol{C}^{-1}\boldsymbol{C}_{0,.}^{'}$$

By depicting the posterior mean and covariance, one can find that Gaussian process model is suitable for predicting the field surface with relatively low mean squared error. To make the predicted surface of the field, the grid needs to be represented





From the prediction error plot, one can tell that once it is observed, the corresponding prediction error shrinks to small values. So one can tell by observing the field using Guassian process, it provides a suitable way to predict the desired field.

Part III Integrated nested Laplace Approximations (INLA)

In the last part of this exercise, we explore the `R-INLA` package along two examples.

3.1 Simple Linear Regression

First, we analyse the ski jumping data set using a linear regression model, which can be phrased as Latent Gaussian model suitable for INLA. Therefore, we start with loading the INLA package and exploring the dataset.

The ski jumping data set contains 26 observations of measured lengths in ski jumping competitions (in meters) between the years 1961 and 2011.

In Figure 1 we depict the 26 observations given their year. We observe a clear (almost linear) trend in the measured jumping lengths to increase with the years.

Suitable for this model assumption on the data, we use linear regression approach for the statistical modelling of this data, where the years x_i are the covariates and the lengths y_i are the responses for $i = 1, \ldots, 26$:

$$\mathbb{E}[y_i] = \mu + \beta x_i, \quad \mathbb{V}\partial\backslash[y_i] = \tau^{-1}.$$

This can be posed as a latent Gaussian model suited for the INLA framework.

1. The response depends on the linear predictors η as $y|x, \theta = \Pi\pi(y_i|\eta_i, \tau)$ with Gaussian likelihood $\pi(y_i|\eta_i, \tau) \sim \mathcal{N}(\eta_i, \tau^{-2})$
2. The parameters μ and β of the linear predictor $eta_i = \mu + x_i\beta$ are independent Gaussian with a fixed huge variance and mean zero. Note that no additional hyperparameter is introduced here.
3. The model's hyperparameter τ is only one-dimensional and is equipped with a Gaussian prior by default with out specification.

The distributions which are not specified in detail here use default settings in the `R-INLA` package, which are naturally compatible with the LGM construction.

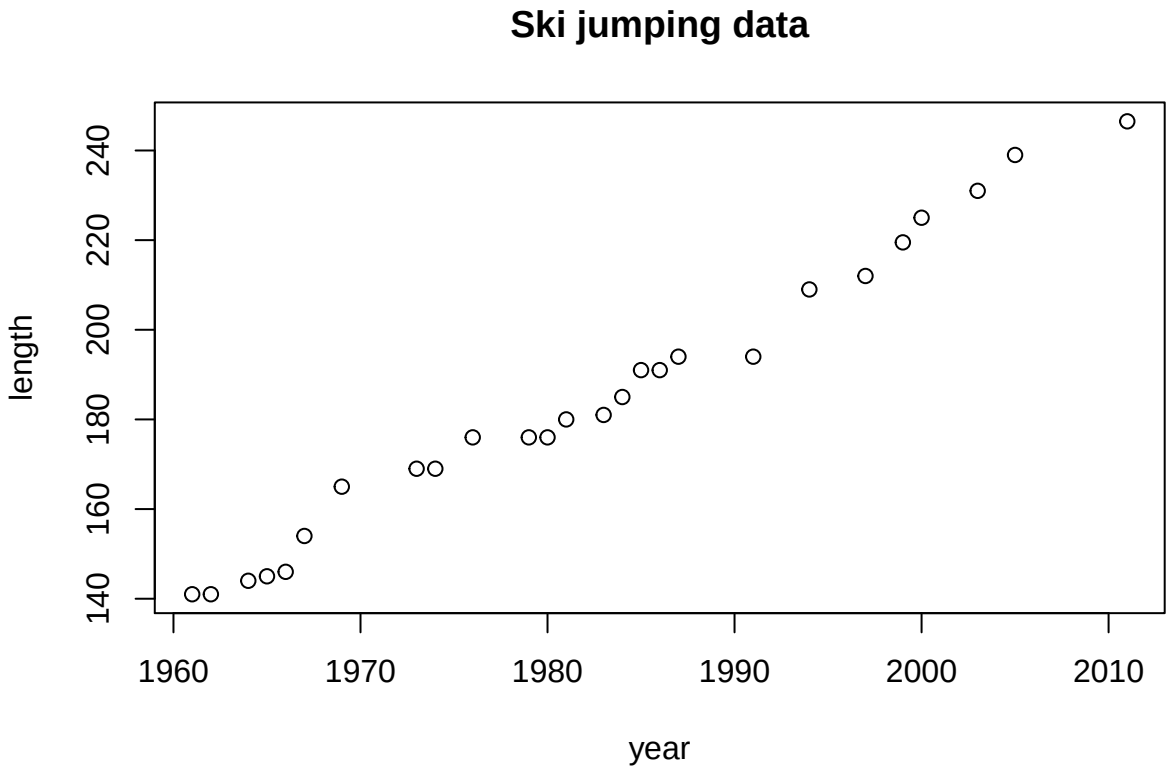


Figure 1: Visualisation of the ski jumping data set

```
##
## Call:
##    "inla(formula = Length ~ Year, data = skiData, control.predictor =
##    list(compute = TRUE))"
## Time used:
##     Pre = 3.83, Running = 0.585, Post = 0.21, Total = 4.63
## Fixed effects:
##                   mean      sd 0.025quant  0.5quant 0.975quant      mode kld
## (Intercept) -4029.646 106.615  -4240.649 -4029.650  -3818.934 -4029.646   0
## Year            2.126   0.054      2.019     2.126      2.232     2.126   0
##
## Model hyperparameters:
##                                          mean   sd 0.025quant 0.5quant
## Precision for the Gaussian observations 0.072 0.02      0.038     0.07
##                                         0.975quant  mode
## Precision for the Gaussian observations      0.116 0.066
##
## Expected number of effective parameters(stdev): 2.00(0.00)
## Number of equivalent replicates : 13.00
##
## Marginal log-Likelihood:  -89.58
## Posterior marginals for the linear predictor and
##  the fitted values are computed
```

The INLA run generates posterior estimates for the fixed effects μ and β, which will be investigated blow. In the summary, we read the precision of the distribution for the hyperparameter τ which is rather small, meaning that the variance will be rather big. However, in a ski jumping competition we can expect a variance

of several meters such that this is reasonable.

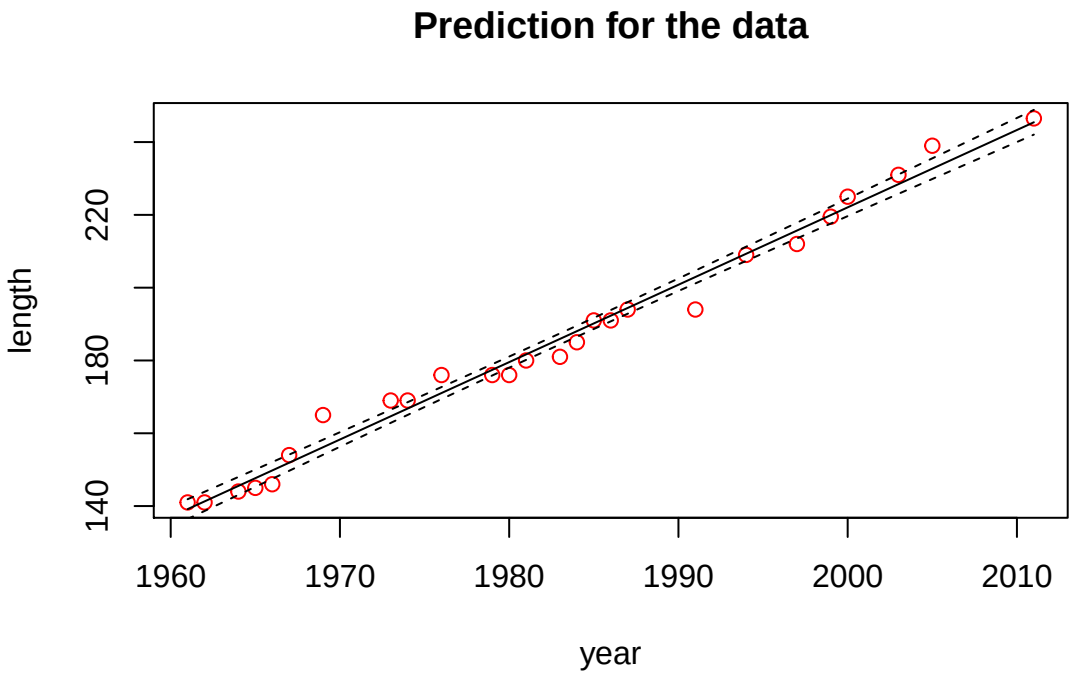


Figure 2: Linear regression with INLA

In Figure 2 we see that the linear trend is really well captured by the model (black line), whereas the 95% credibility interval (dashed line) does not cover all data points.

```
## [1] "Estimates for sigma:"
```

```
## Mean            3.83993
## Stdev           0.552976
## Quantile  0.025 2.93482
## Quantile  0.25  3.44617
## Quantile  0.5   3.77761
## Quantile  0.75  4.16357
## Quantile  0.975 5.09958
```

In Figure 3, we depict the marginal posterior distributions for all variables of interest. In particular, the transformed marginal analysis confirms our interpretation that the variance in the magnitude of several meters (roughly around 4 meters), but again this is very reasonable for this application.

3.2 GLMM with random effects

Last, we use INLA to analyse the "Seeds" data set. This data concerns the portion of seeds that germinated on a sample set of 21 plates. The plates are equipped with one of two types of seeds and one of two types of extracts. The characteristics of plate i are described by the covariates $x_{1,i}$ which is the type of seed and $x_{2,i}$ which is the type of root extract - these covariates are either 0 or 1 for the different possibilies. Then the number of germinated seeds r_i on plate i is counted in contrast to the number of total seeds n_i on that plate. Having p_i as the probability of germination on plate i, a binomial model for this example is

$$r_i \sim Binomial(p_i, n_i)$$
$$\text{logit}(p_i) = a_0 + a_1 x_{1,i} + a_2 x_{2,i} + \varepsilon_i$$

where ε_i is some iid noise. As above non-specified prior- and hyperparameter-distributions use the default of `R-INLA`. This model is then again implemented in `R-INLA`.

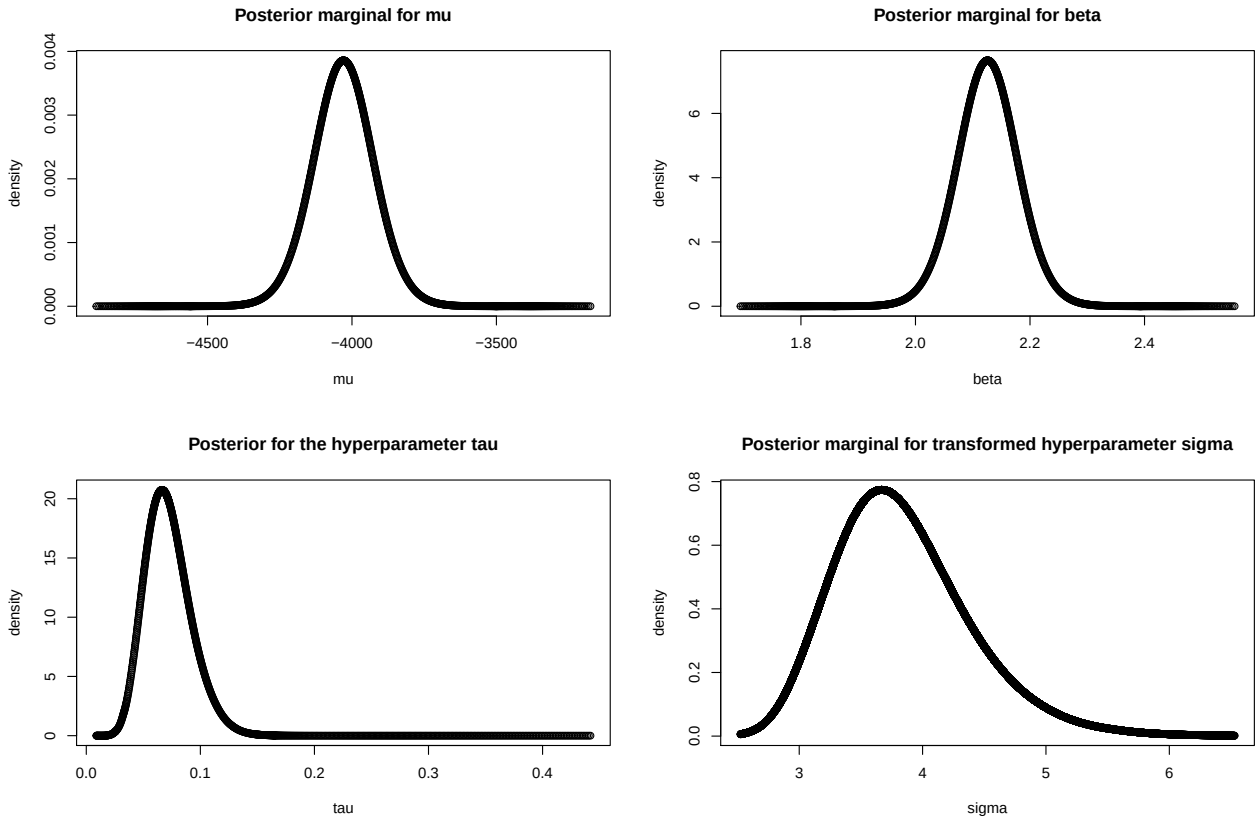


Figure 3: Posterior marginals for selected effects

```
## [1] "The x1 covariate:"
##  [1] 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1
## [1] "The x2 covariate:"
##  [1] 0 0 0 0 0 1 1 1 1 1 1 1 0 0 0 0 0 1 1 1 1 1
##
## Call:
##    c("inla(formula = formula, family = \"binomial\", data = data, Ntrials
##    = n, ", " control.predictor = list(compute = TRUE), control.family =
##    list(link = \"logit\"))" )
## Time used:
##     Pre = 4.39, Running = 0.426, Post = 0.21, Total = 5.03
## Fixed effects:
##               mean    sd 0.025quant 0.5quant 0.975quant   mode kld
## (Intercept) -0.429 0.115     -0.656   -0.429     -0.204 -0.428   0
## x1          -0.272 0.156     -0.580   -0.272      0.033 -0.271   0
## x2           1.066 0.146      0.782    1.066      1.353  1.065   0
##
## Random effects:
##   Name     Model
##     plate IID model
##
## Model hyperparameters:
##                           mean       sd 0.025quant 0.5quant 0.975quant mode
## Precision for plate 18930.69 20071.40      30.63 12701.76   73765.97 7.83
##
## Expected number of effective parameters(stdev): 3.25(1.16)
## Number of equivalent replicates : 6.46
##
## Marginal log-Likelihood:  -73.75
## Posterior marginals for the linear predictor and
##  the fitted values are computed
```

Remark the structured pattern in the covariates. Furthermore, here the precision is rather higher yielding a small variance in the estimate.

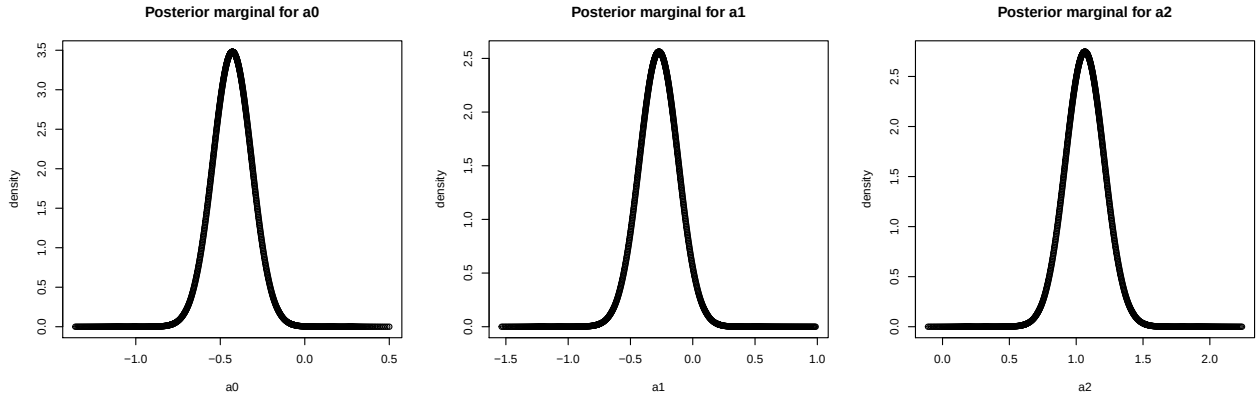


Figure 4: Posterior marginals for selected effects

In Figure 4, we see rather confident estimates for the fixed effects of the model. In particular, we note that the effect a_2 will dominate the model since we only have 0 and 1 values for the covariates. The effect a_0 is

common for all plates and the absolute value of a_1 for the seed type is a way smaller than the one for the root extract.

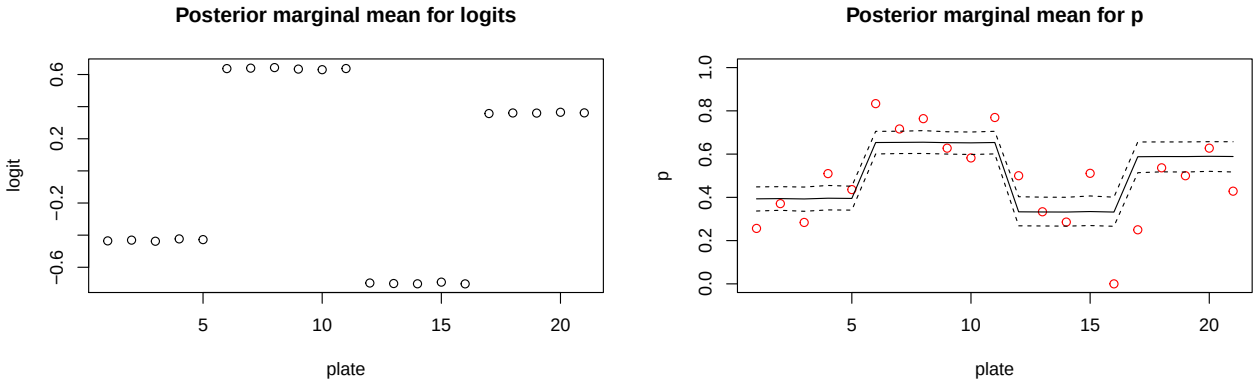


Figure 5: GLMM with INLA

In Figure 5, we recognize a clear pattern with high values when the root extract with value 1 is chosen. The choice of the seed only has minor influence on the prediction. Moreover, the variance of this model cannot cover all data points, which is a common issue in binomial models with few data though.