

# Chapter5

## Spatial non-stationarity

20200908

金森由妃

# 概要

## “non-stationary”・・・空間相関が均一でない場合

- (i) 共変量を使って共分散をモデリングする  
(空間相関に関するパラメータを応答変数、空間の不均一性を表現するような情報（例えば標高）を説明変数として線形回帰する)
  - (ii) 物理的障壁を作る  
(普通の地図と障壁の地図から地点間の関係性に強弱をつけていく)
- 
- 5.1 Explanatory variables in the covariance; (i)の方法
  - 5.2 The Barrier model; (ii)の方法
  - 5.3 Barrier model for noise data in Albacete (Spain); (ii)の方法
  - 5.3.0.4 Space-time model

# 5.1 Explanatory variables in the covariance

In this section we present an example of the model proposed in [Ingebrigtsen, Lindgren, and Steinsland \(2014\)](#).

This example describes a way to include explanatory variables (i.e., covariates) in both of the SPDE model parameters.

原著論文の方が分かりやすいので、 そちらで説明

GRF with explanatory variables in the covariance parameters. For further theoretical details on the approach refer to [Lindgren et al. \(2011\)](#), and for a more practical introductory tutorial on how to use SPDE based GRFs using the `r-inla` package, refer to [Lindgren \(2012\)](#).

### 3.1. Background

Using SPDEs in geostatistical modelling is a relatively new approach. It was introduced by [Lindgren et al. in 2011](#) and has since been extended and applied in various contexts ([Bolin and Lindgren, 2011, 2013](#); [Simpson et al., 2012a,b](#); [Cameletti et al., 2013](#)). However, the original idea dates back to the work of [Whittle \(1954; 1963\)](#), where it is shown that the solution to the following SPDE

$$(\kappa^2 - \Delta)^{\alpha/2}x(\mathbf{u}) = \mathcal{W}(\mathbf{u}), \quad \mathbf{u} \in \mathbb{R}^d, \quad \alpha = \nu + d/2, \quad \kappa > 0, \quad \nu > 0, \quad (1)$$

is a Gaussian random field with Matérn covariance function. The innovation process  $\mathcal{W}$  on the right hand side of (1) is spatial Gaussian white noise, and  $\Delta$  is the Laplacian. The Matérn covariance function between locations  $\mathbf{u}$  and  $\mathbf{v}$  in  $\mathbb{R}^d$  is given by

$$C(\mathbf{u}, \mathbf{v}) = \frac{\sigma^2}{2^{\nu-1}\Gamma(\nu)}(\kappa\|\mathbf{v} - \mathbf{u}\|)^{\nu}K_{\nu}(\kappa\|\mathbf{v} - \mathbf{u}\|), \quad (2)$$

where  $K_{\nu}$  is the modified Bessel function of the second kind and order  $\nu > 0$ ,  $\kappa$  is a positive scaling parameter, and  $\sigma^2$  is the marginal variance.

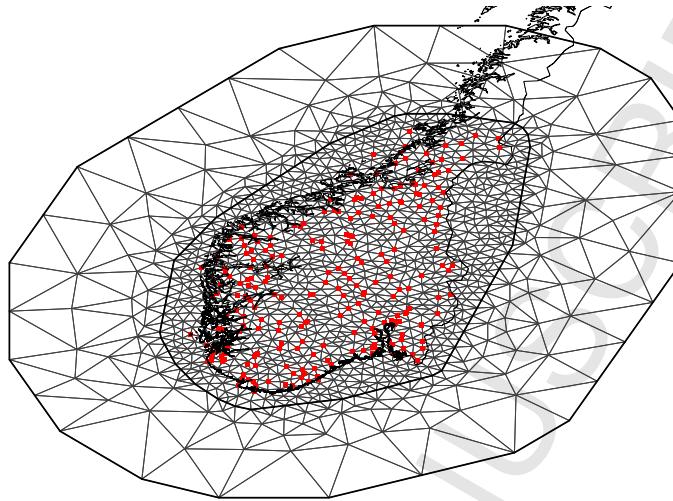
[Lindgren et al. \(2011\)](#) show that an approximation to the solution of the SPDE in (1) can be obtained using a finite element method (FEM), a numerical technique for solving partial differential equations (see e.g. [Brenner and Scott \(2008\)](#)). The approximated solution is computationally efficient for inferential purposes: instead of using a GRF with dense covariance matrix, the computations can be carried out with a GMRF with sparse precision matrix (the inverse of the covariance matrix).

The first step is to replace the infinite dimensional GRF with a finite basis function representation

$$x(\mathbf{u}) = \sum_{i=1}^m \psi_i(\mathbf{u})w_i. \quad (3)$$

Here, the  $w_i$ 's are weights and the  $\psi_i$ 's are piece-wise linear basis functions defined on a triangulation of the domain, i.e. a subdivision into non-intersecting triangles with  $m$  nodes; see Figure 2 for the triangulation of southern Norway that will be used for the analysis of the precipitation dataset in Section 5.1.

The basis functions  $\{\psi_i\}_{i=1,\dots,m}$  are deterministic and defined for every node in the triangulation, or mesh. [Lindgren et al. \(2011\)](#) use piece-wise linear basis functions, which have compact support; they are equal to one in the mesh nodes and zero in all other nodes. The stochastic properties of the approximation are determined by the weights  $\mathbf{w} = (w_1, \dots, w_m)$ . These are chosen so that the representation in (3) approximates the distribution of the solution to the SPDE in (1). The distribution of the weights  $\mathbf{w}$  is Gaussian, with Markov properties



**Figure 2:** Triangulation of southern Norway. The red dots indicate the observation locations of the annual precipitation data.

determined by the triangulation, leading to a sparse precision matrix. The elements of the precision matrix depend on  $\alpha$  and  $\kappa$ , see Result 2 in [Lindgren et al. \(2011\)](#).

### 3.2. Non-stationarity with explanatory variables in the dependence structure

One of the main advantages with the SPDE approach is its flexibility. Properties of the random field are now characterised by an SPDE rather than a covariance function, which enables us to modify the SPDE instead of the covariance function to obtain GRFs with other dependence structures than the stationary Matérn covariance. The local nature of the differential operators allows local specification of parameters, and only mild regularity conditions are necessary to ensure a valid model.

In the following, we assume a two-dimensional spatial domain, i.e.  $d = 2$ , and fix  $\nu = 1$ , which implies  $\alpha = 2$ . The parameter  $\nu$  in the Matérn covariance function determines the mean-square differentiability of the field ([Diggle and Ribeiro, 2007](#), Chapter 3.4.1), which can influence predictions made by the model. As  $\nu$  is difficult to identify from data it is common to either fix  $\nu$  or to fit models for half-integer values of  $\nu$  and choose its value by model selection ([Diggle and Ribeiro, 2007](#), Chapter 5.4). For stationary models, the authors' discussion response from [Lindgren et al. \(2011\)](#) provides GMRFs corresponding to any  $\nu$ , but no work has yet been done to carry that out for non-stationary models.

The parameter  $\kappa$  is a scaling parameter, linked to the range  $\rho$  by the empirically derived relationship  $\rho = \sqrt{8}/\kappa$ . For a GRF with Matérn covariance with

parameters  $\kappa$  and  $\nu = 1$  spatial correlation is 0.1 at distance  $\rho$ . Thus, we can think of  $\kappa$  as a range parameter governing the spatial dependence structure of the GRF. We add more flexibility to the dependence structure by rescaling the field  $x(\mathbf{u})$  with a variance parameter  $\tau$ , which yields the marginal variance

$\mathbf{u}(\mathbf{A})$

$$\sigma^2 = \frac{1}{4\pi\kappa^2\tau^2}. \leftarrow \text{stationary if } \kappa \text{ is constant} \quad (4)$$

Motivated by the annual precipitation data we introduce space-dependent covariance parameters, i.e.  $\tau$  and  $\kappa$  are now functions of the spatial location  $\mathbf{u}$ ,  $\mathbf{u} \in \mathcal{D}$ , where  $\mathcal{D}$  is the spatial domain. The modified SPDE becomes

$$(\kappa(\mathbf{u})^2 - \Delta)(\tau(\mathbf{u})x(\mathbf{u})) = \mathcal{W}(\mathbf{u}), \quad \mathbf{u} \in \mathbb{R}^2, \quad (5)$$

where the solutions  $x(\mathbf{u})$  are non-stationary GRFs because  $\tau$  and  $\kappa$  vary with location, with the consequence that the variance and correlation range vary with location.

Define  $\log \tau(\mathbf{u})$  and  $\log \kappa(\mathbf{u})$  as sums of basis functions,

$$\log \tau(\mathbf{u}) = \theta_1^\tau + \sum_{k=2}^N b_k^\tau(\mathbf{u})\theta_k^\tau, \quad \log \kappa(\mathbf{u}) = \theta_1^\kappa + \sum_{k=2}^N b_k^\kappa(\mathbf{u})\theta_k^\kappa, \quad (6)$$

where the  $\theta$ 's are weight parameters and  $b_k^\tau(\cdot)$  and  $b_k^\kappa(\cdot)$  are deterministic basis functions defined on the domain  $\mathcal{D}$ , in practice in the nodes of the triangulation. The basis functions  $b_k^\tau(\cdot)$  and  $b_k^\kappa(\cdot)$  should typically be smoothly varying with respect to the triangulation to ensure that they can be adequately represented on the mesh. In Lindgren et al. (2011) the basis functions were generic low order spherical harmonics and B-splines. In this work we investigate the use of parametric models to control the SPDE parameters, using explanatory variables as basis functions.

As noted by Lindgren et al. (2011) the deformation method for non-stationary models due to Sampson and Guttorp (1992) can be reformulated as a non-stationary SPDE in the special case of deformations within a plane. However, for deformations into a higher dimensional space the two approaches differ considerably. Some deformation and SPDE models can have similar properties, but in general the two model classes are not equivalent, since the SPDE approach measures distances within or along a manifold. The non-stationary model produced by (5) and (6) leads to local changes of distance metric that can not be represented by a simple deformation model, and hence leads to different models than the explanatory variable models from Schmidt et al. (2011). Since the non-stationary parameters control the local distance metric in the manifold they must be defined on the mesh.

From a technical point of view, having space-dependent  $\tau$  and  $\kappa$  only changes the elements of the precision matrix of the Markov representation of the GRF. The weight vector  $\mathbf{w}$  from the representation in (3) is  $\mathcal{N}(\mathbf{0}, \mathbf{Q}(\boldsymbol{\theta})^{-1})$ , where  $\boldsymbol{\theta} = (\theta_1^\tau, \dots, \theta_N^\tau, \theta_1^\kappa, \dots, \theta_N^\kappa)$ , i.e. the elements of the precision matrix  $\mathbf{Q}(\boldsymbol{\theta})$  are

functions of the  $\theta$ 's in (6). Following Lindgren et al. (2011), we introduce two finite element structure matrices,  $\mathbf{C}$  and  $\mathbf{G}$ , and write the precision matrix as

$$\mathbf{Q}(\boldsymbol{\theta}) = \mathbf{T}(\mathbf{K}^2 \mathbf{C} \mathbf{K}^2 + \mathbf{K}^2 \mathbf{G} + \mathbf{G} \mathbf{K}^2 + \mathbf{G} \mathbf{C}^{-1} \mathbf{G}) \mathbf{T}, \quad (7)$$

where  $\mathbf{T}$  and  $\mathbf{K}$  are diagonal matrices with  $T_{ii} = \tau(\mathbf{u}_i)$  and  $K_{ii} = \kappa(\mathbf{u}_i)$ , and  $i$  is an index over the  $m$  nodes in the triangulation. The matrix  $\mathbf{C}$  is diagonal, with  $C_{ii} = \int \psi_i(\mathbf{u}) d\mathbf{u}$ , and  $\mathbf{G}$  is sparse positive semi-definite, with  $G_{ij} = \int \nabla \psi_i(\mathbf{u}) \cdot \nabla \psi_j(\mathbf{u}) d\mathbf{u}$ . Recall that the  $\psi_i$ 's are the basis functions from the representation in (3).

The relationship between the model parameters and the marginal variance given by (4) is valid in the stationary case but not in the non-stationary case. However, by disregarding the spatial interaction between the non-stationary parameter fields we obtain nominal approximations to the variance and correlation range of the non-stationary GRF as functions of elevation;

$$\sigma^2(\mathbf{u}) \approx \frac{1}{4\pi\kappa(\mathbf{u})^2\tau(\mathbf{u})^2}, \quad (8)$$

and

$$\rho(\mathbf{u}) \approx \frac{\sqrt{8}}{\kappa(\mathbf{u})}. \quad (9)$$

These approximations are valid for slowly varying  $\kappa(\mathbf{u})$ , and can be used for easy interpretation of the parameters. In Section 5.1, Figure 5 we check how close the approximation is to the actual field variances, which can be obtained numerically (see Rue et al., 2009). Such a comparison for the spatial range is not done, since using a single value to represent the local spatial range in a non-stationary model is inherently problematic, and (9) is used only as a convenient qualitative measure of local range.

#### 4. Models

We suggest two Bayesian hierarchical models for the annual precipitation data: one where the spatial field is a stationary GRF and one where it is a non-stationary GRF. In Section 5.1 the two models are compared in terms of model fit and predictive performance using the data from southern Norway.

##### 4.1. Annual precipitation model

Let the spatial process  $\{\xi(\mathbf{u}): \mathbf{u} \in \mathcal{D}\}$ , represent the true level of annual precipitation in southern Norway. We assume that this process is observed with additive measurement error at the  $n = 233$  weather stations. This yields the following data model for the observations  $y_1, \dots, y_n$

$$y_i = \xi(\mathbf{u}_i) + \epsilon_i, \quad i = 1, \dots, n, \quad (10)$$

where the noise terms  $\epsilon_1, \dots, \epsilon_n$  are iid  $\mathcal{N}(0, 1/\tau_\epsilon)$ , and independent of  $\xi(\cdot)$ .

Furthermore, we assume that the precipitation process can be modelled by three parts: an intercept  $\beta_0$ , a linear effect of elevation  $\beta_1$ , and a spatial field  $x(\mathbf{u})$ . The process model can be written as

$$\xi(\mathbf{u}) = \beta_0 + \beta_1 h(\mathbf{u}) + x(\mathbf{u}), \quad \mathbf{u} \in \mathcal{D}, \quad (11)$$

where  $h(\mathbf{u})$  is the elevation at location  $\mathbf{u}$ .

#### 4.2. Stationary and non-stationary annual precipitation model

The two precipitation models we compare in this work are obtained by using different models for the spatial field  $x(\mathbf{u})$ : 1) a stationary GRF,  $x_S(\mathbf{u})$ , and 2) a non-stationary GRF,  $x_{N-S}(\mathbf{u})$ . We refer to the models as the stationary and non-stationary model, respectively, and for later reference we write the two process models here

$$\xi_S(\mathbf{u}) = \beta_0 + \beta_1 h(\mathbf{u}) + x_S(\mathbf{u}), \quad \mathbf{u} \in \mathcal{D}, \quad (12)$$

$$\xi_{N-S}(\mathbf{u}) = \beta_0 + \beta_1 h(\mathbf{u}) + x_{N-S}(\mathbf{u}), \quad \mathbf{u} \in \mathcal{D}. \quad (13)$$

For the GRFs we use the SPDE based model as defined in Section 3.2 and (6). For the stationary  $x_S$ ,

$$\log \tau = \theta_1^\tau \quad \text{and} \quad \log \kappa = \theta_1^\kappa. \quad (14)$$

And for the non-stationary  $x_{N-S}$ ,

$$\log \tau(\mathbf{u}) = \theta_1^\tau + h(\mathbf{u}) \theta_h^\tau \quad \text{and} \quad \log \kappa(\mathbf{u}) = \theta_1^\kappa + h(\mathbf{u}) \theta_h^\kappa, \quad (15)$$

i.e. elevation is included as a log-linear effect on  $\kappa$  and  $\tau$ . Note that the stationary model is equal to the non-stationary model with  $\theta_h^\tau = \theta_h^\kappa = 0$ .

#### 4.3. Identifiability constraints

In both the stationary and the non-stationary model, the three components of the model are non-identifiable, since the spatial field  $x(\mathbf{u})$  can capture both an overall mean and an elevation dependent effect. To separate the linear effect of elevation, and to make the model identifiable we require two linear orthogonality constraints for the spatial field:

$$\int_{\mathcal{D}} x(\mathbf{u}) d\mathbf{u} = \int_{\mathcal{D}} h(\mathbf{u}) x(\mathbf{u}) d\mathbf{u} = 0.$$

In the implementation, this is accomplished by discretising the integrals on the mesh into

$$\mathbf{1}^T \mathbf{C} \mathbf{w} = \mathbf{h}^T \mathbf{C} \mathbf{w} = 0,$$

where  $\mathbf{C}$  is the same diagonal matrix as in (7), and  $\mathbf{w}$  and  $\mathbf{h}$  are the vectors of weights for the basis function representations of  $x(\mathbf{u})$  and  $h(\mathbf{u})$ .

# 5.1 Explanatory variables in the covariance

In this section we present an example of the model proposed in [Ingebrigtsen, Lindgren, and Steinsland \(2014\)](#).

This example describes a way to include explanatory variables (i.e., covariates) in both of the SPDE model parameters.

$$\log(\tau(s)) = \underline{b_0^{(\tau)}(s)} + \sum_{k=1}^p \underline{b_k^{(\tau)}(s)\theta_k}$$

inla.spde2.matern() の B.tau

$$\log(\kappa(s)) = \underline{b_0^{(\kappa)}(s)} + \sum_{k=1}^p \underline{b_k^{(\kappa)}(s)\theta_k}$$

inla.spde2.matern() の B.kappa

## 5.1 Explanatory variables in the covariance

Both the marginal standard deviation and the range can be modeled considering a regression model as detailed in Lindgren and Rue (2015). This is the case of considering that the log of  $\sigma$  and log of  $\rho$  are modeled by a regression on basis functions as

$$\log(\sigma(\mathbf{s})) = b_0^\sigma(\mathbf{s}) + \sum_{k=1}^p b_k^\sigma(\mathbf{s})\theta_k$$
$$\log(\rho(\mathbf{s})) = b_0^\rho(\mathbf{s}) + \sum_{k=1}^p b_k^\rho(\mathbf{s})\theta_k$$

where  $b_0^\sigma(\mathbf{s})$  and  $b_0^{\rho(\mathbf{s})}$  are offsets,  $b_k^\sigma()$  and  $b_k^{\rho(\mathbf{s})}$  are basis functions that can be defined on spatial locations or covariates, each one with an associated  $\theta_k$  parameter. Notice that `B.tau` and `B.kappa` are basis functions evaluated at each mesh node. Therefore, to include a covariate in  $\sigma^2$  or in the range we do need to have it available at each mesh node.

In our example we consider the domain area as the rectangle  $(0, 10) \times (0, 5)$  and the range as a function of the first coordinate of the location defined by  $\rho(\mathbf{s}) = \exp(\theta_2 + \theta_3(s_{i,1} - 5)/10)$ . This gives

$$\begin{aligned}\log(\sigma) &= \log(\sigma_0) + \theta_1 \\ \log(\rho(\mathbf{s})) &= \log(\rho_0) + \theta_2 + \theta_3 b(\mathbf{s})\end{aligned}$$

where  $b(\mathbf{s}) = (s_{i,1} - 5)/10$ .

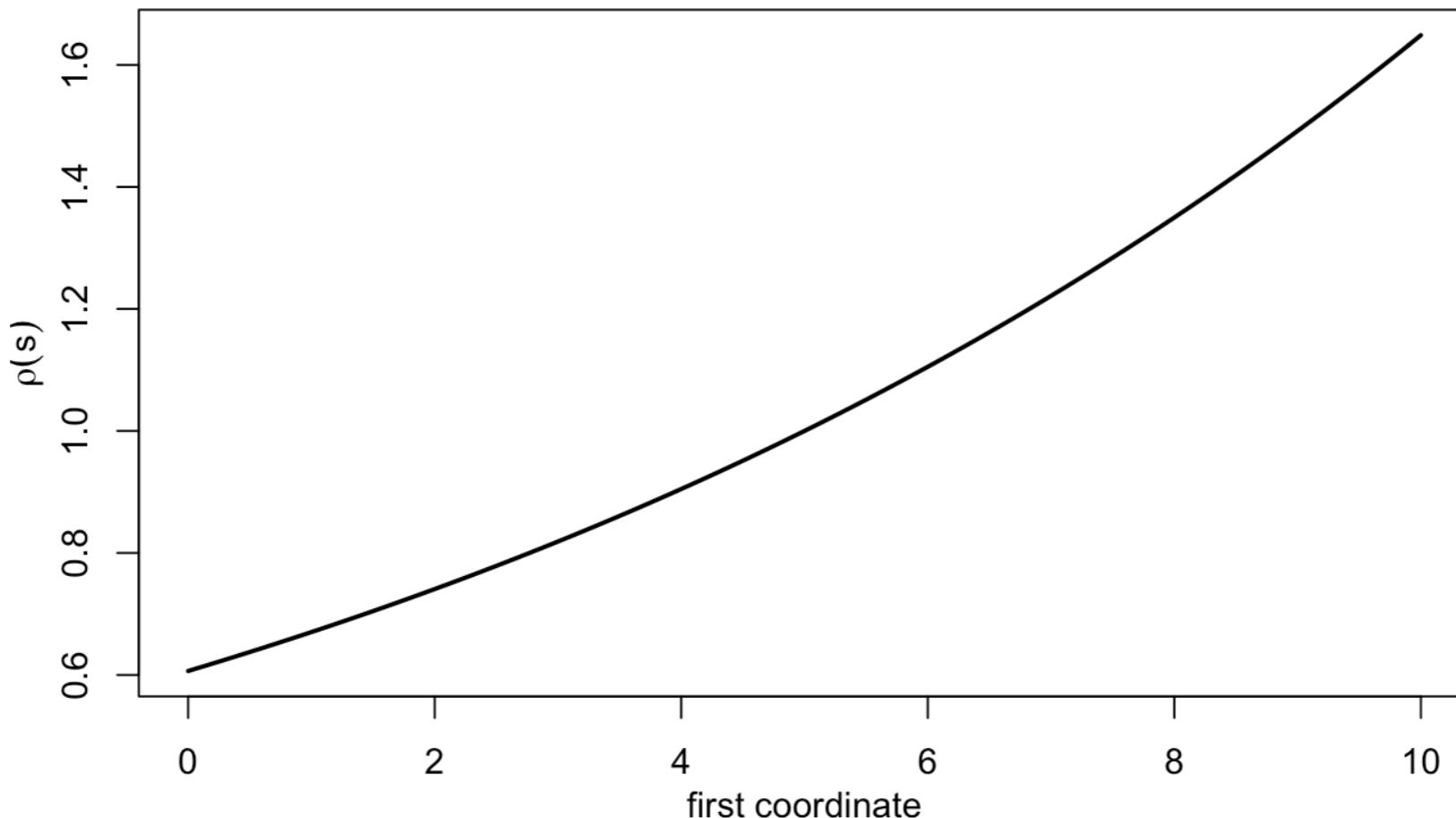


Figure 5.1: Range as function of the first coordinate.

We have to define the basis functions for  $\log(\tau)$  and  $\log(\kappa)$ . Following Lindgren and Rue (2015) we need to write  $\log(\kappa(s))$  and  $\log(\tau(s))$  as functions of  $\log(\rho(s))$  and  $\log(\sigma)$ . Substituting  $\log(\sigma)$  and  $\log(\rho(s))$  into the equations for  $\log(\kappa(s))$  and  $\log(\tau(s))$  we have

$$\begin{aligned}\log(\tau(s)) &= \log(\tau_0) - \theta_1 + \nu\theta_2 + \nu\theta_3 b(s) \\ \log(\kappa(s)) &= \log(\kappa_0) - \theta_2 - \theta_3 b(s).\end{aligned}$$

Therefore we have to include  $(s_{i,1} - 5)/10$  as a fourth column in `B.tau` and minus it in `B.kappa` as well.

$$\log(\tau) = \frac{1}{2} \log \left( \frac{\Gamma(\nu)}{\Gamma(\alpha)(4\pi)^{d/2}} \right) - \log(\sigma) - \nu \log(\kappa)$$

$$\log(\kappa) = \frac{\log(8\nu)}{2} - \log(\rho).$$

$\tau$ ,  $\kappa$ ,  $\sigma^2$ ,  $\rho$  の関係を使って導くみたいだけど、ちょっと分からなかった. . .

```
ドメイン pl01 <- cbind(c(0, 1, 1, 0, 0) * 10, c(0, 0, 1, 1, 0) * 5)
```

**メッシュ**      `mesh <- inla.mesh.2d(loc.domain = pl01, cutoff = 0.1,  
                      max.edge = c(0.3, 1), offset = c(0.5, 1.5))`

$$\begin{aligned}\log(\tau(\mathbf{s})) &= \log(\tau_0) - \theta_1 + \nu\theta_2 + \nu\theta_3 b(\mathbf{s}) \\ \log(\kappa(\mathbf{s})) &= \log(\kappa_0) - \theta_2 - \theta_3 b(\mathbf{s}) .\end{aligned}$$

```

nu <- 1

alpha <- nu + 2 / 2

# log(kappa)
logkappa0 <- log(8 * nu) / 2

# log(tau); in two lines to keep code width within range
logtau0 <- (lgamma(nu) - lgamma(alpha) -1 * log(4 * pi)) / 2
logtau0 <- logtau0 - logkappa0

# SPDE model

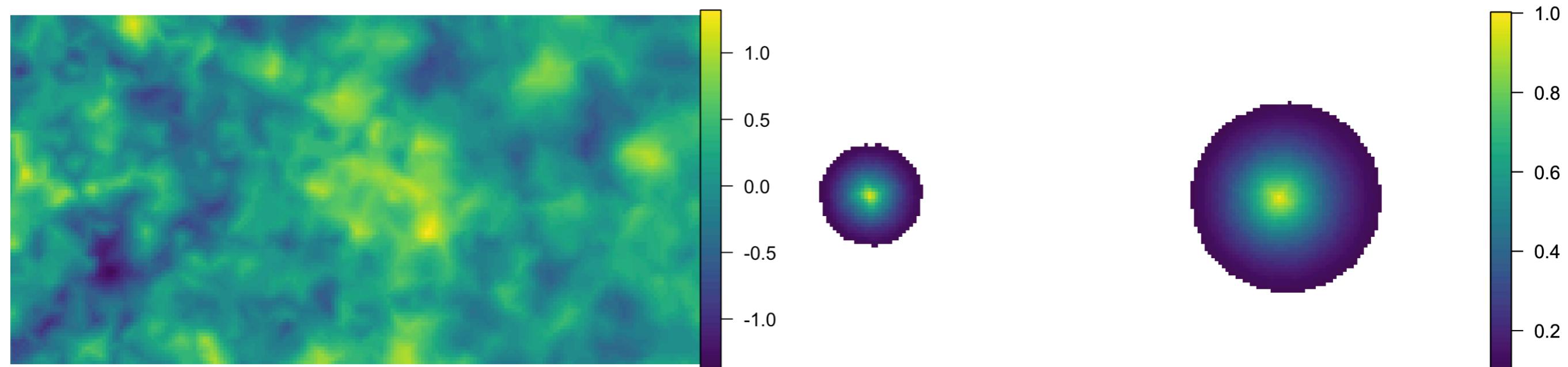
spde <- inla.spde2.matern(mesh,                                     共変量の設定
B.tau = cbind(logtau0, -1, nu, nu * (mesh$loc[,1] - 5) / 10),
B.kappa = cbind(logkappa0, 0, -1, -1 * (mesh$loc[,1] - 5) / 10),
theta.prior.mean = rep(0, 3),
theta.prior.prec = rep(1, 3))

Q <- inla.spde2.precision(spde, theta = theta)

```

## 5.1.3 Simulation at the mesh nodes

- `inla.qsample()`; データの発生
- `book.plot.field()`; シミュレーションデータのプロット
- `book.spatial.correlation()`; ある場所での空間相関をプロット



## 5.1.4 Estimation with data simulated at the mesh nodes

- ・ メッシュのノード = データ点なので、 predictor matrixとstack関数が必要ない

```
clik <- list(hyper = list(theta = list(initial = 20,  
fixed = TRUE)))
```

```
formula <- y ~ 0 + f(i, model = spde)
```

```
res1 <- inla(formula, control.family = clik,  
data = data.frame(y = sample, i = 1:mesh$n))
```

## 5.1.5 Estimation with location not at the mesh nodes

データ地点を作る

```
set.seed(2)
```

```
n <- 200
```

```
loc <- cbind(runif(n) * 10, runif(n) * 5)
```

メッシュのノードで発生したデータをデータ地点に写す

```
projloc <- inla.mesh.projector(mesh, loc)
x <- inla.mesh.project(projloc, sample)
```

データ地点≠メッシュのノードだからstackでデータを統合し、フィッティング

```
stk <- inla.stack(
  data = list(y = x),
  A = list(projloc$proj$A),
  effects = list(data.frame(i = 1:mesh$n)),
  tag = 'd')

res2 <- inla(formula, data = inla.stack.data(stk),
  control.family = clik,
  control.predictor = list(compute = TRUE, A = inla.stack.A(stk)))
```

## 5.2 The Barrier model

## 5.2 The Barrier model

- 最も一般的な空間モデルは, stationary isotropic model
- reason assumptionだけど, 物理的障壁がある時にはunreasonable
- 例えば, 沿岸域に生息する水性生物についてモデルを作る時
- Bakka et al. (2016)

- SGFで考慮すべき物理的障壁の問題; coastline problem
- 水までの最短距離や境界条件（ノイマンとかディリクレ）によって物理的障壁を決める方法が提案されてきたが、それぞれ問題があった

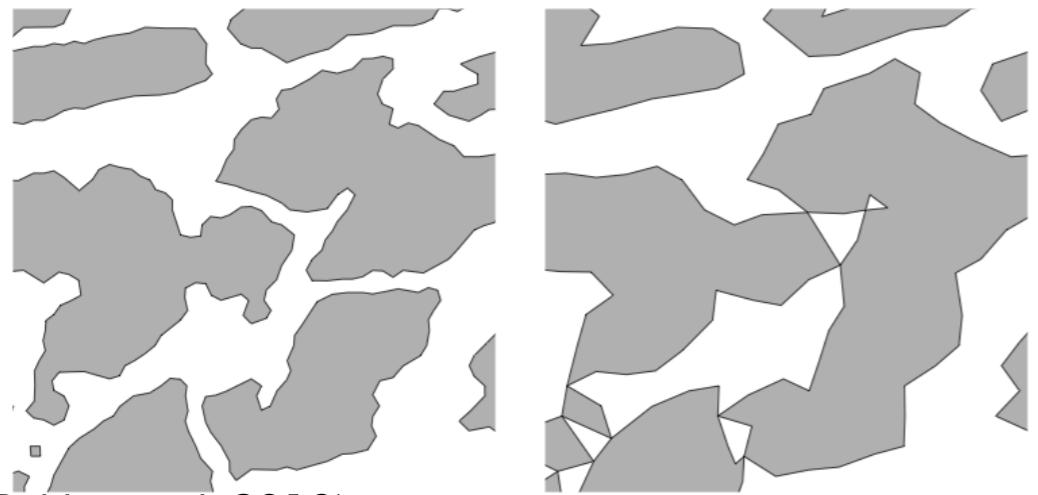


Fig. 2 (Bakka et al. 2019)

海岸線はポリゴンで表現するが、近似の仕方によって海岸線の形が変わる  
(boundary polygon selection process)  
→”水までの最短距離”という定義が  
口バストでない

- **Barrier model**; Matérn correlationを使う

- Matérn fieldを、グリッド上のSimultaneous Autoregressive (SAR) モデルとして解釈し、近似する

$$U_{i,j} - k(U_{i-1,j} + U_{i,j-1} + U_{i+1,j} + U_{i,j+1}) = z_{i,j}$$

$U_{\{i, j\}}$ ; グリッド*i, j*のランダム変数,  $z_{\{i, j\}}$ ; 平均0のガウス

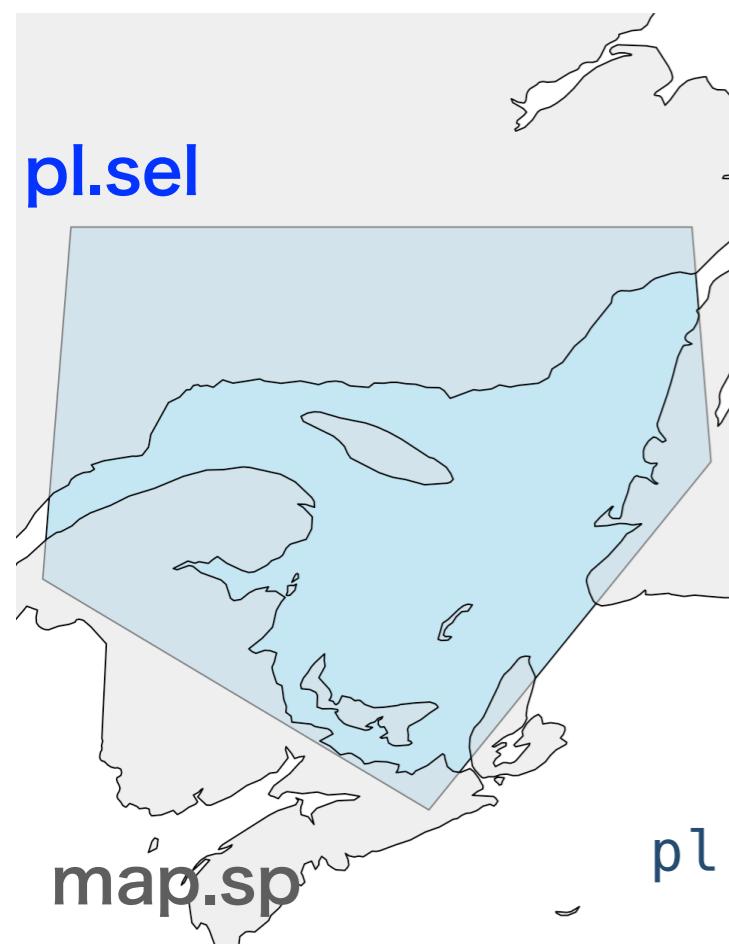
- 距離は最短距離ではなく、ある場所から別の場所への全てのパスの集まりと解釈される
- 2点間の関係は、それらの間に存在する全てのパスに依存
- 物理的な障壁の部分だけ関係性を0に近くする
- SARモデルは陸地も含めた全領域を含むが、フィッティングや予測は陸地では行われない
- 境界部分では

$$(1 - k_2)U_{0,0} - k(U_{-1,0} + U_{0,-1} + (1 - k_3)U_{1,0} + U_{0,1}) = z_{0,0}$$

陸地

- SARとMatérnモデルがSPDEを通してリンクしていないと、 $k_2$ と $k_3$ の適切な値は選択できない

## 5.2.1 Canadian coastline example



```
# Select region
map <- map("world", "Canada", fill = TRUE,
           col = "transparent", plot = FALSE)
IDs <- sapply(strsplit(map$names, ":"), function(x) x[1])
map.sp <- map2SpatialPolygons(
  map, IDs = IDs,
  proj4string = CRS("+proj=longlat +datum=WGS84"))
pl.sel <- SpatialPolygons(list(Polygons(list(Polygon(
  cbind(c(-69, -62.2, -57, -57, -69, -69),
        c(47.8, 45.2, 49.2, 52, 52, 48)),
  FALSE)), '0'))), proj4string = CRS(proj4string(map.sp)))
poly.water <- gDifference(pl.sel, map.sp)
```

→ 緯度経度座標からUTM座標に変換する (5.2.1.2)

## 5.2.1 Canadian coastline example

pl.sel



map.sp

```
# Select region  
map <- map("world", "Canada", fill = TRUE,  
           col = "transparent", plot = FALSE)  
IDs <- sapply(strsplit(map$names, ":"), function(x) x[1])  
map.sp <- map2SpatialPolygons(  
  map, IDs = IDs,  
  proj4string = CRS("+proj=longlat +datum=WGS84"))
```

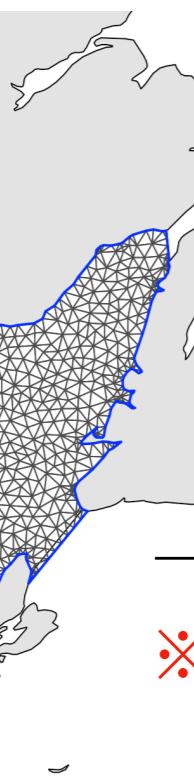
```
pl.sel <- SpatialPolygons(list(Polygons(list(Polygon(  
  cbind(c(-69, -62.2, -57, -57, -69, -69),  
  c(47.8, 45.2, 49.2, 52, 52, 48)),  
  FALSE)), '0'))), proj4string = CRS(proj4string(map.sp)))
```

```
poly.water <- gDifference(pl.sel, map.sp)
```

→ 緯度経度座標からUTM座標に変換する (5.2.1.2)

→ inla.mesh.2d()でメッシュを作成 (5.2.1.3)

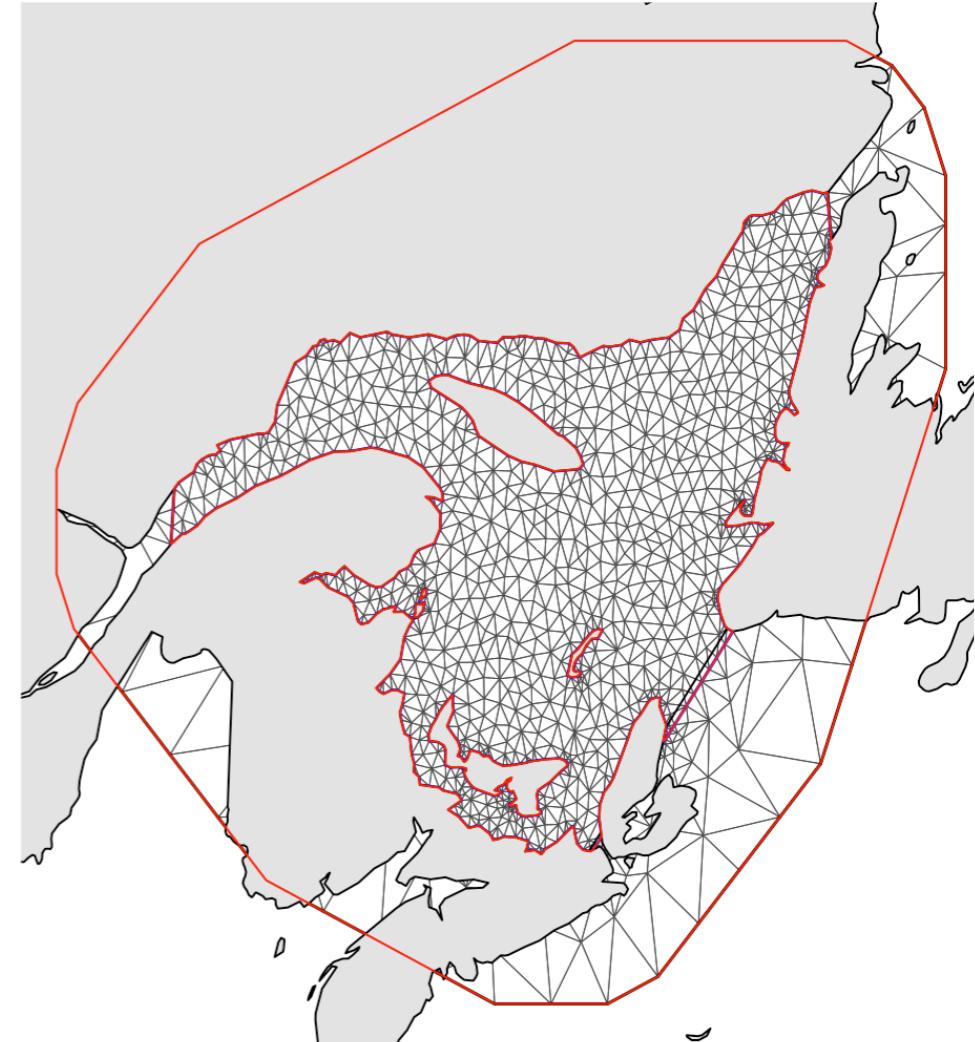
※Neumann boundary conditionsと呼ばれる隠れた仮定を置いている



## 5.2.1.4 Mesh over land and sea

we construct a mesh over both water and land.

```
max.edge = 30  
bound.outer = 150  
mesh <- inla.mesh.2d(boundary = poly.water,  
  max.edge = c(1,5) * max.edge,  
  cutoff = 2,  
  offset = c(max.edge, bound.outer))
```



Next, we select the triangles of the mesh that are inside `poly.water`.

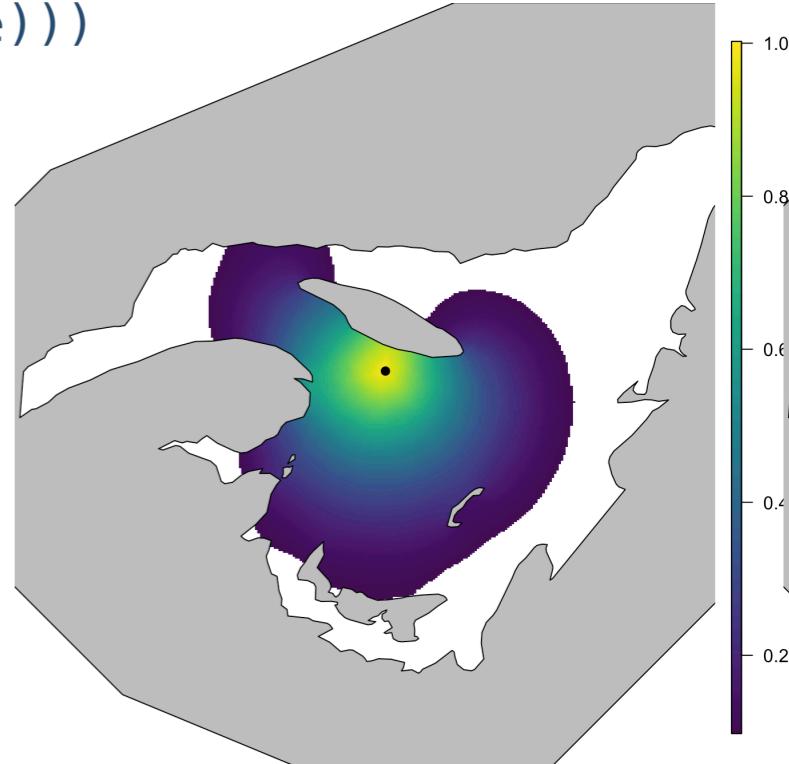
```
water.tri = inla.over_sp_mesh(poly.water, y = mesh,  
  type = "centroid", ignore.CRS = TRUE)  
num.tri = length(mesh$graph$tv[, 1])  
barrier.tri = setdiff(1:num.tri, water.tri)  
poly.barrier = inla.barrier.polygon(mesh,  
  barrier.triangles = barrier.tri)
```

- non-stationaryとstationaryで空間相関を比較

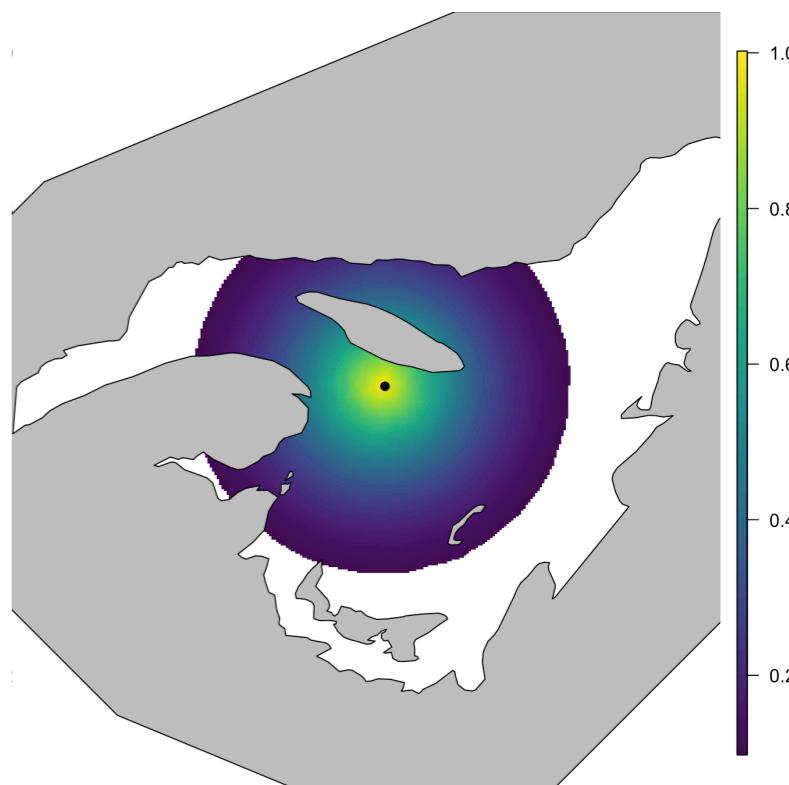
```
range <- 200

barrier.model <- inla.barrier.pcmatern(mesh,
  barrier.triangles = barrier.tri)

Q <- inla.rgeneric.q(barrier.model, "Q", theta = c(0, log(range)))
```



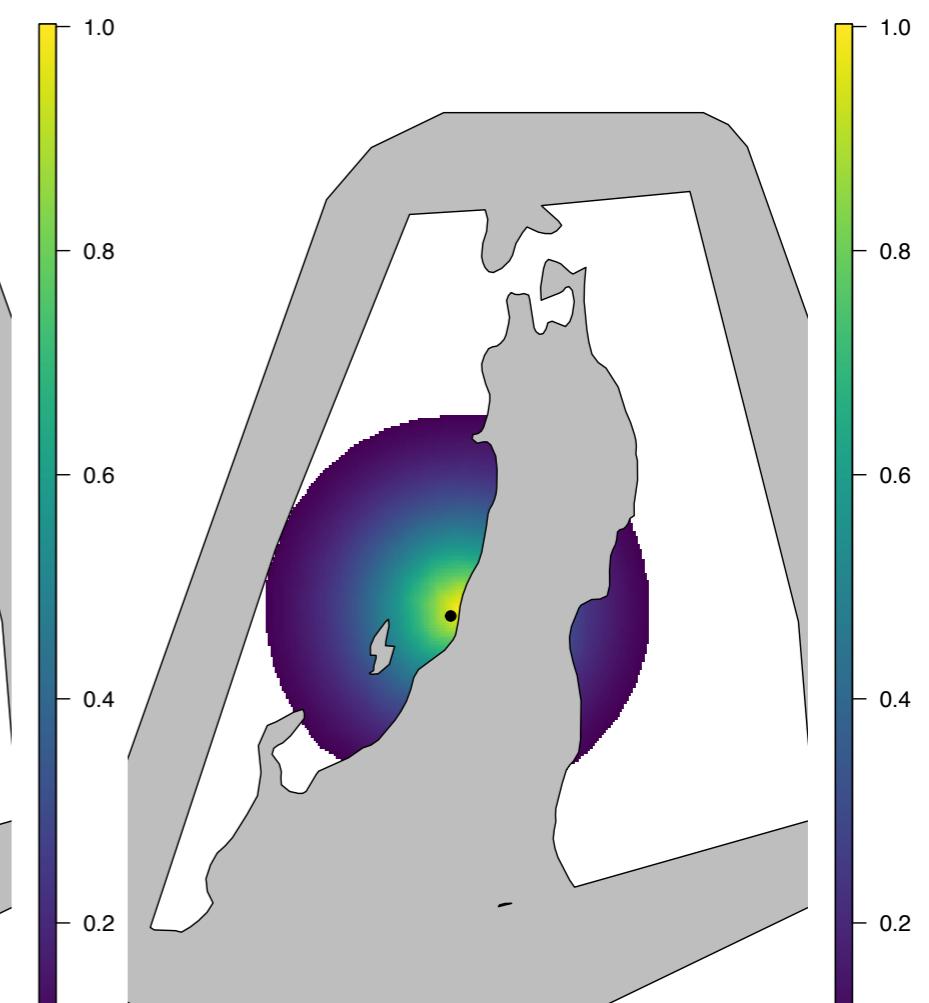
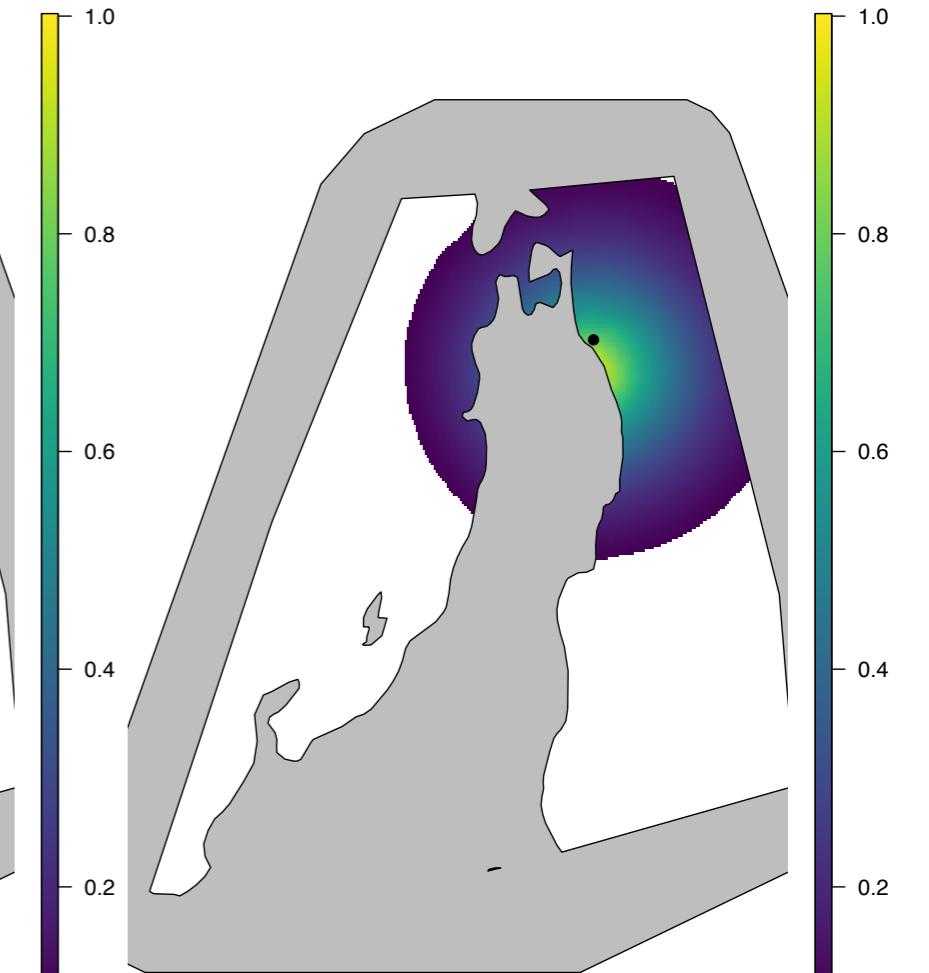
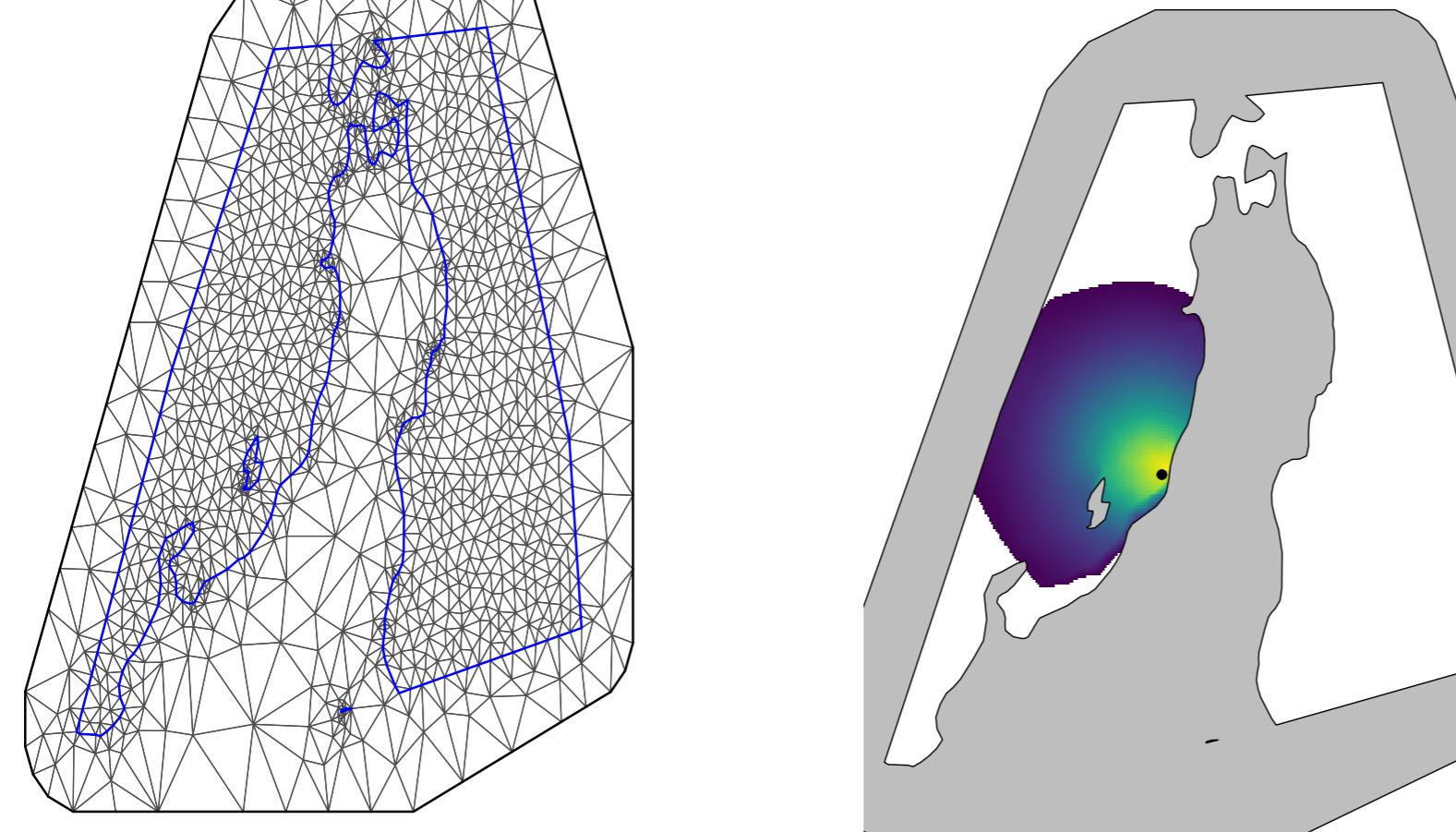
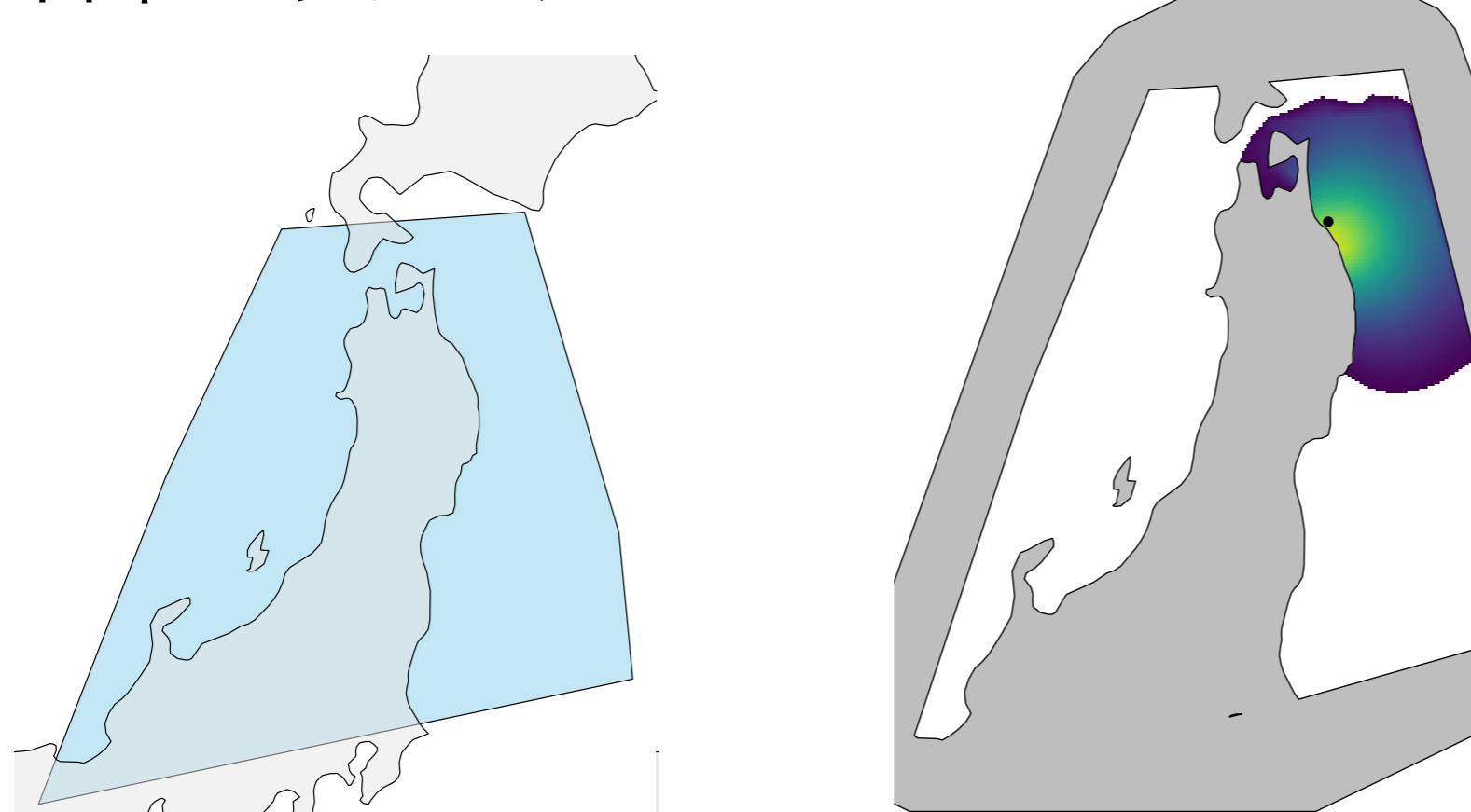
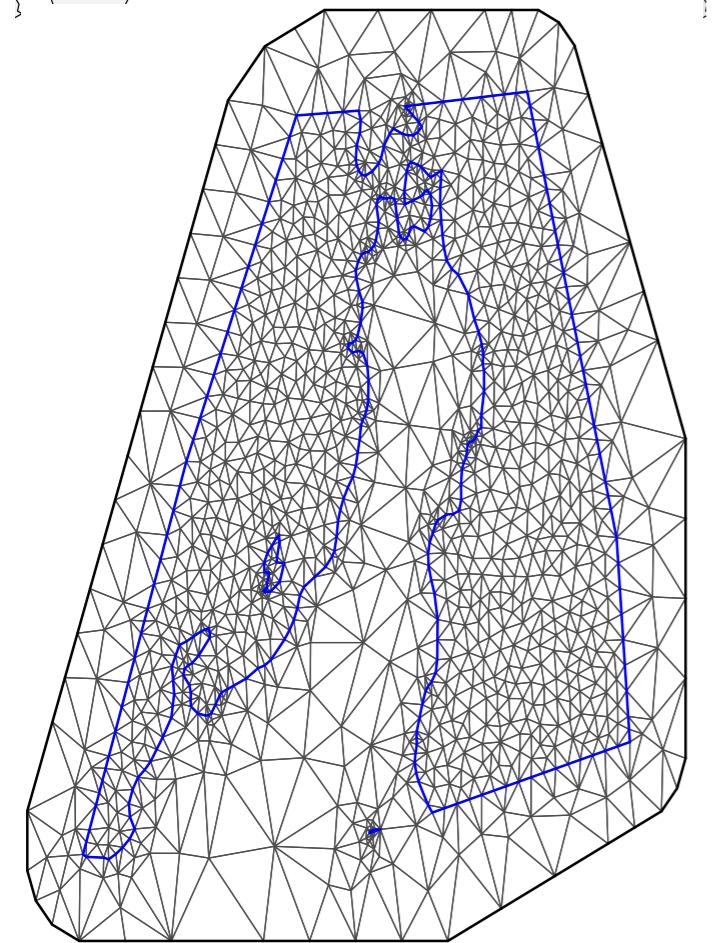
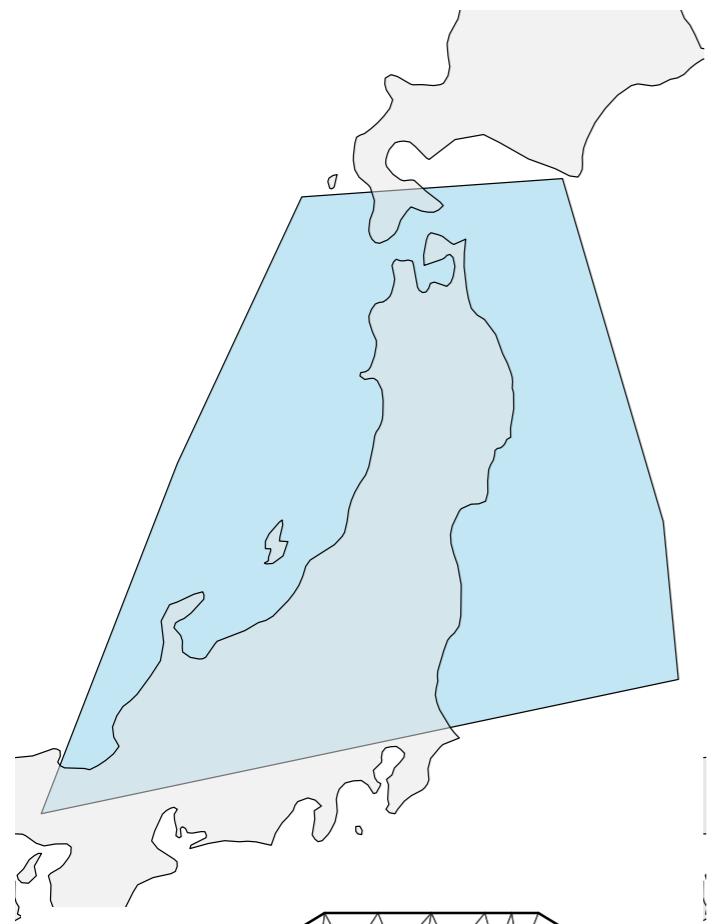
```
stationary.model <- inla.spde2.pcmatern(mesh,
  prior.range = c(1, 0.1), prior.sigma = c(1, 0.1))
Q.stat <- inla.spde2.precision(stationary.model,
  theta = c(log(range), 0))
```



# The location we find the correlation with respect to

```
loc.corr <- c(500, 5420)
corr <- book.spatial.correlation(Q, loc = loc.corr, mesh)
corr.stat <- book.spatial.correlation(Q.stat, loc = loc.corr,
  mesh)
```

- 日本でやってみた



- ・シミュレーションデータの作成, モデルの作成, 予測

```
set.seed(201805)

loc.data <- spsample(poly.water, n = 1000, type = "random")
loc.data <- loc.data@coords

# Seed is the month the code was first written times some number
u <- inla.qsample(n = 1, Q = Q, seed = 201805 * 3)[, 1]
A.data <- inla.spde.make.A(mesh, loc.data)
u.data <- A.data %*% u

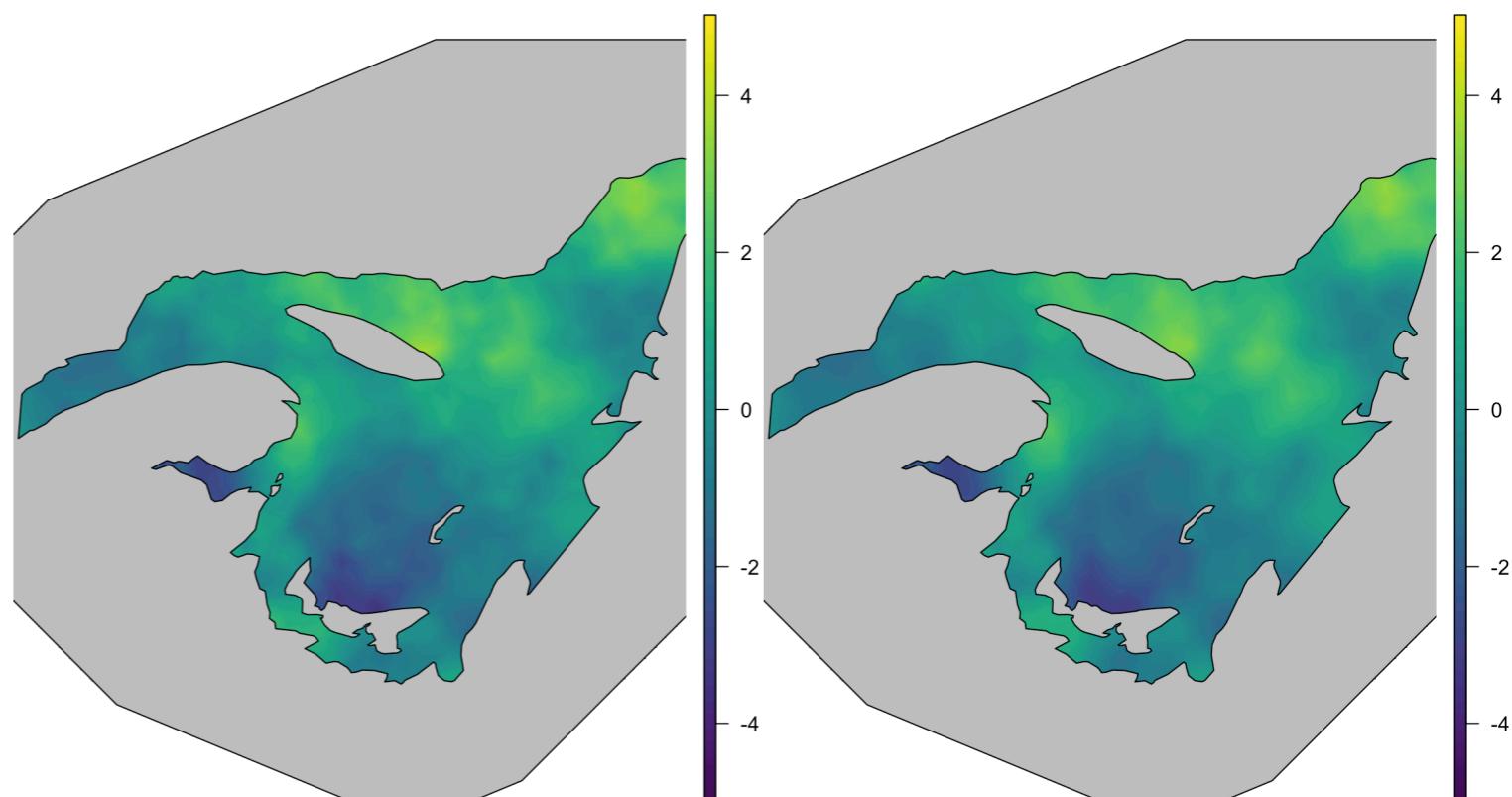
# df is the dataframe used for modeling
df <- data.frame(loc.data)
names(df) <- c('locx', 'locy')
# Size of the spatial signal
sigma.u <- 1
# Size of the measurement noise
sigma.epsilon <- 0.1
df$y <- drop(sigma.u * u.data + sigma.epsilon * rnorm(nrow(df)))
```

- ・シミュレーションデータの作成, モデルの作成, 予測

```
stk <- inla.stack(  
  data = list(y = df$y),  
  A = list(A.data, 1),  
  effects =list(s = 1:mesh$n, intercept = rep(1, nrow(df))),  
  tag = 'est')  
  
form.barrier <- y ~ 0 + intercept + f(s, model = barrier.model)  
  
res.barrier <- inla(form.barrier, data = inla.stack.data(stk),  
  control.predictor = list(A = inla.stack.A(stk)),  
  family = 'gaussian',  
  control.inla = list(int.strategy = "eb"))
```

These have a close match.

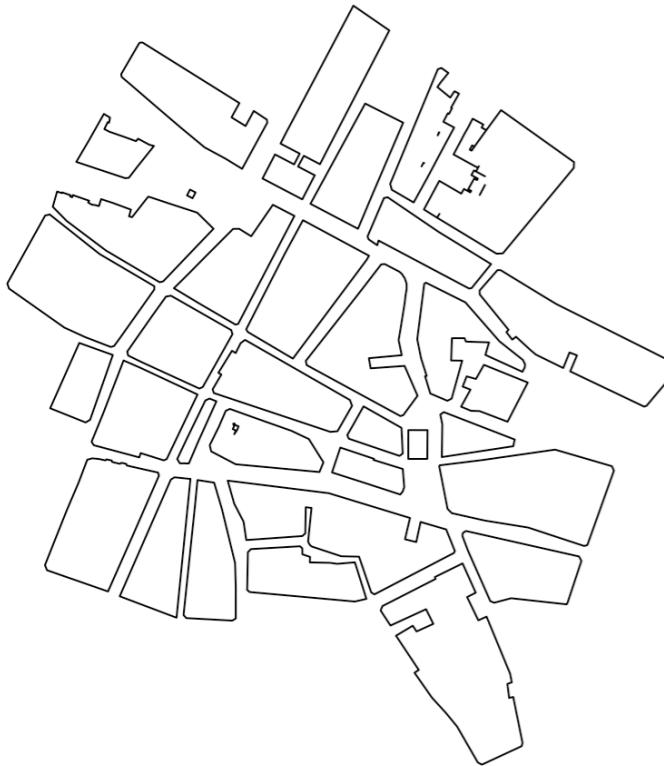
We observe quick changes across islands or small peninsulas, which may be smoothed out by a stationary model.



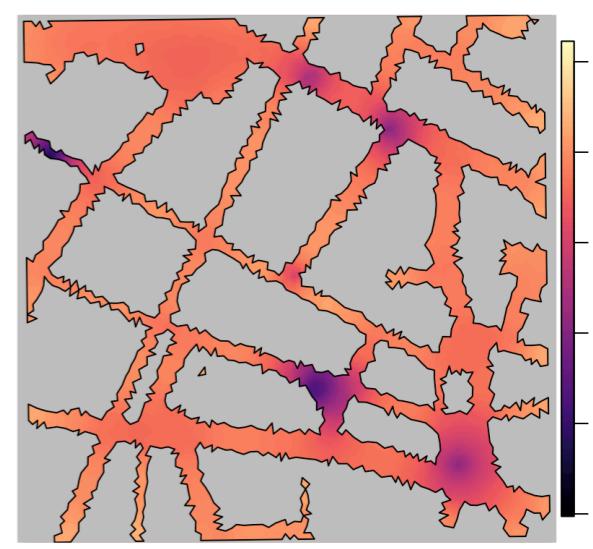
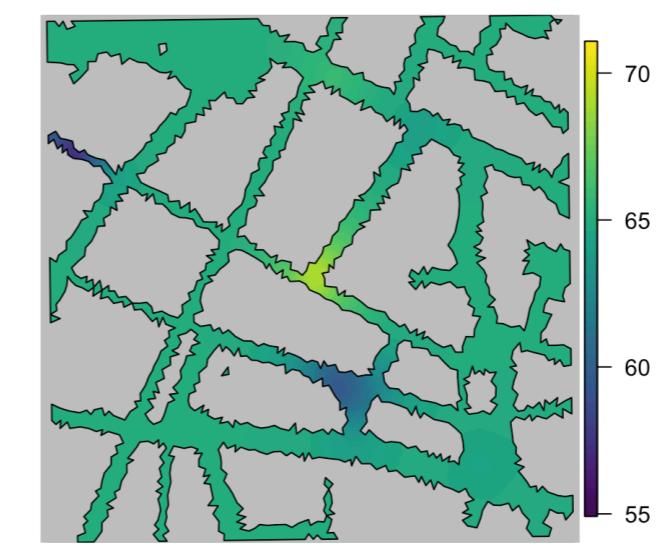
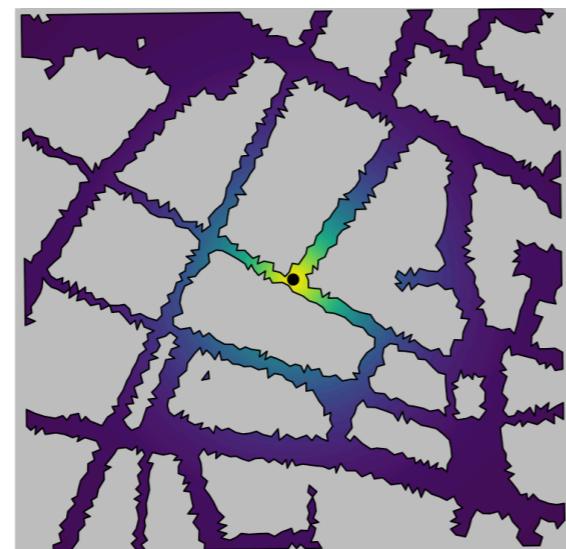
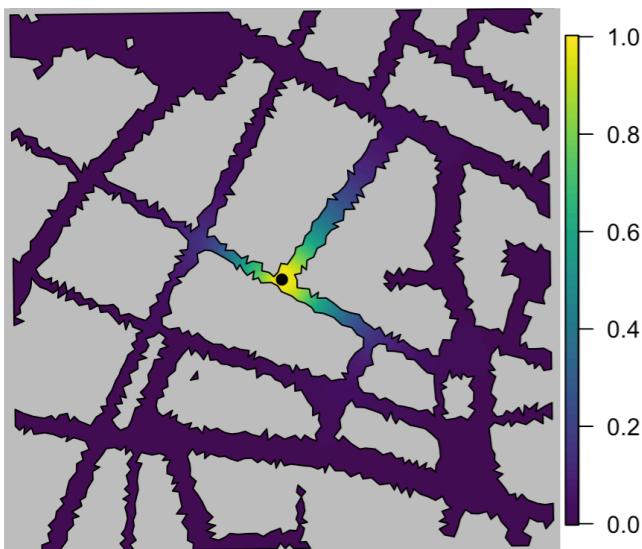
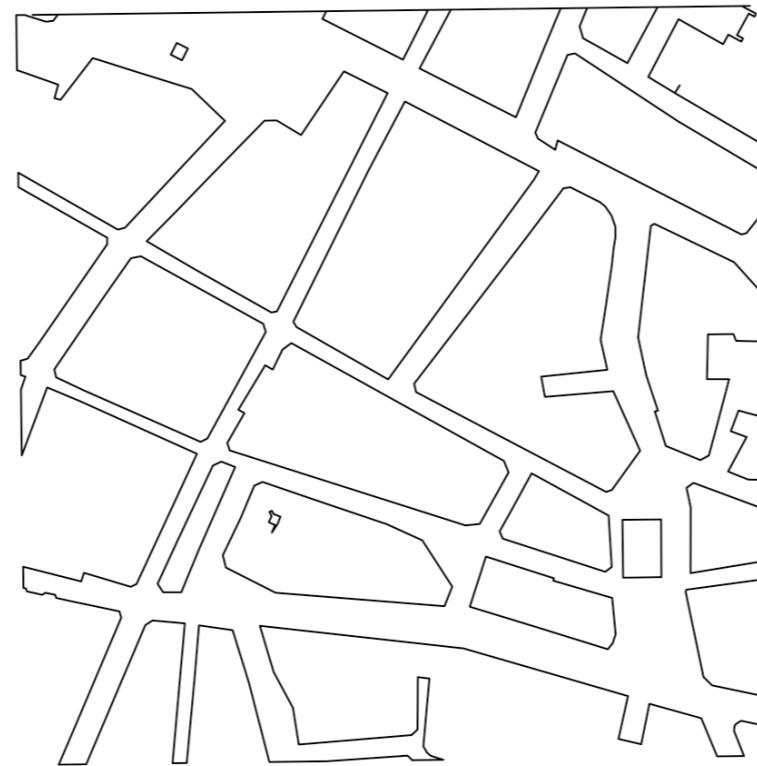
# 5.3

## ・市街地の騒音（建物が障壁）

Buildings



Streets



空間相関。

左がbarrier modelで右がstationary model

1 amの予測と97.5%quantile

- ・時空間モデルにしてみる（詳細は7章8章で）

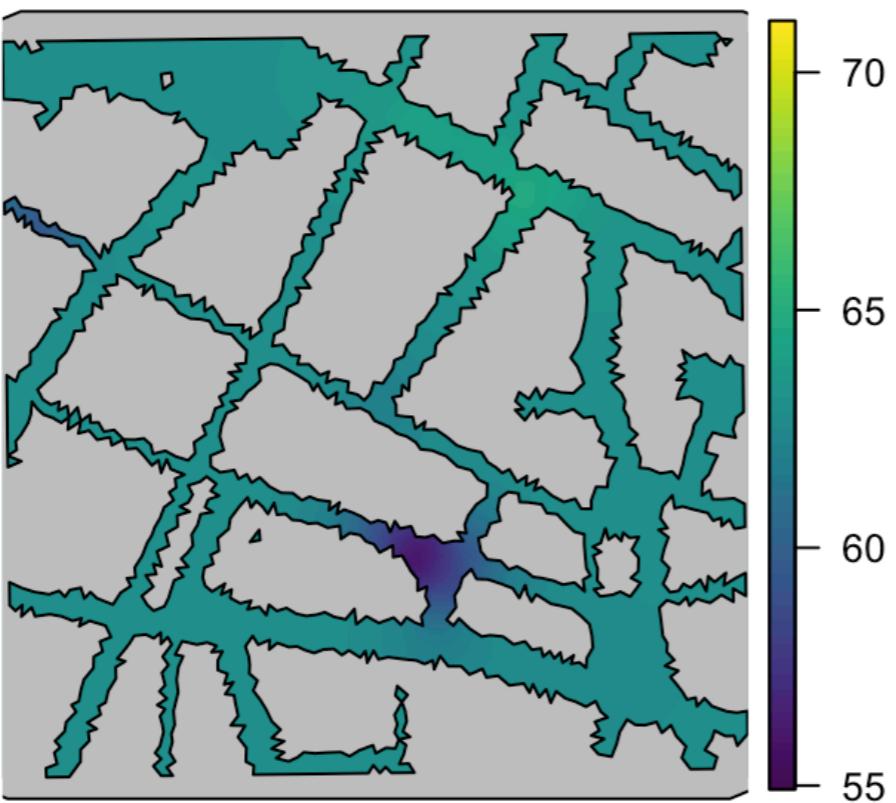
```
A.st <- inla.spde.make.A(mesh = mesh,
  loc = coordinates(noise)[rep(1:7, 24), ])

stk.st <- inla.stack(
  data = list(y = unlist(noise@data[, 2 + 1:24])),
  A = list(A.st, 1, 1),
  effects = list(s = 1:mesh$n,
    intercept = rep(1, 24 * nrow(noise)),
    time = rep(1:24, each = nrow(noise)))))

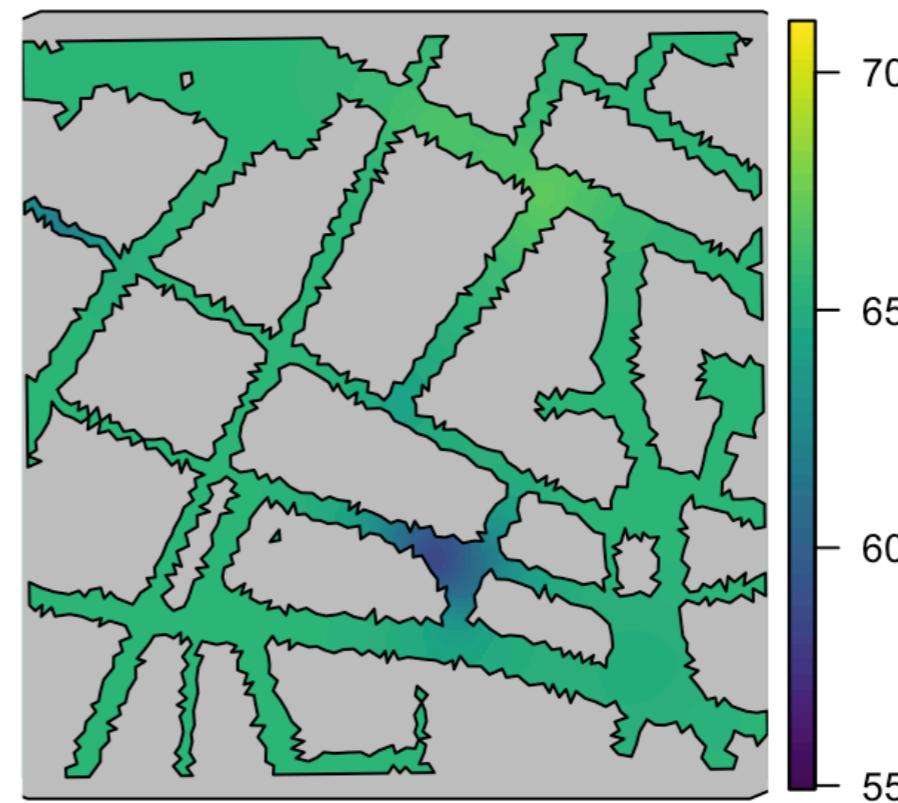
# Model formula
form.barrier.st <- y ~ 0 + intercept +
  f(s, model = barrier.model) +
  f(time, model = "rw1", cyclic = TRUE, scale.model = TRUE,
    hyper = list(theta = stdev.pcprior))
```

inla()はいつも通り。PC priorを使用

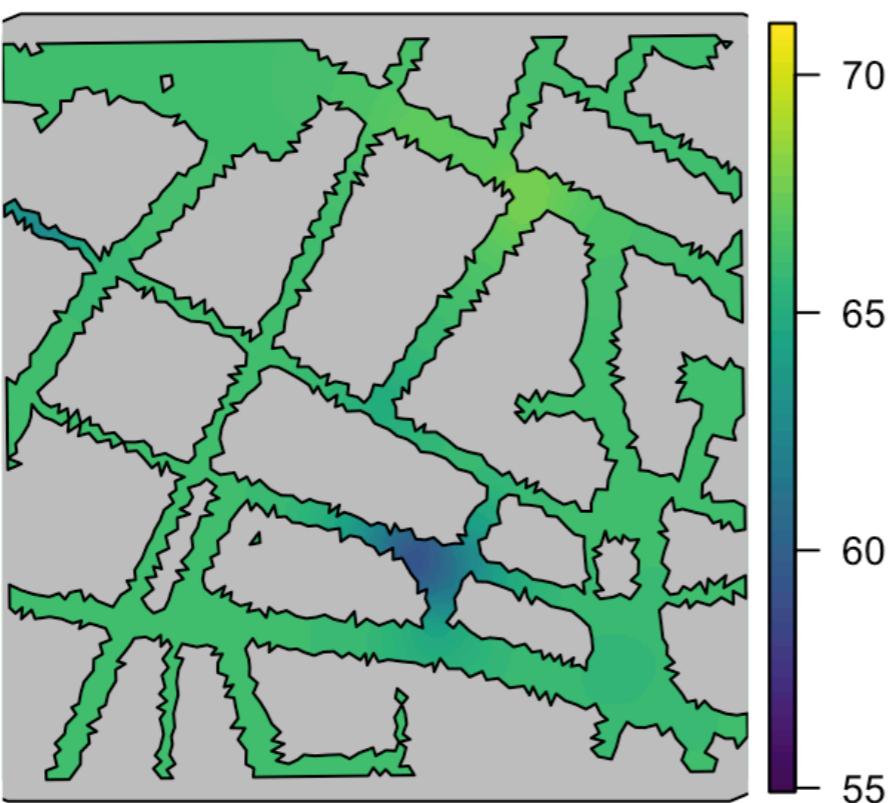
Time: 6



Time: 12



Time: 18



Time: 24

