`GeoModels` Tutorial: simulation, estimation and prediction of spatial data with asymmetric and heavy tails using sinh-arcsinh (SAS) random fields

Christian Caamaño-Carrillo
Moreno Bevilacqua

September 18, 2025

Introduction

In this tutorial we show how to analyze spatial data with heavy tails using sinh-arcsinh (SAS) random fields (Blasi et al., 2022) with the R package `GeoModels` (Bevilacqua et al. (2025)). This process incorporates flexible marginal distributions involving two additional parameters that model heavier or lighter tails than those induced by Gaussian processes and/or possible asymmetries. One advantage of the SAS compared to other recent proposals (e.g., Xua and Genton, 2017) is that the transformation involved is explicitly invertible, which allows likelihood-based methods of estimation to be applied directly.

We first load the R libraries needed in this tutorial and set the name of the model in the `GeoModels` package.

```
rm(list=ls());
install.packages("GeoModels");
library(GeoModels)
require(fields);
model="SinhAsinh";  # model name in the  GeoModels package
set.seed(91);
```

Simulation of sinh-arcsinh (SAS) random fields

The definition of a SAS random field starts by considering a 'parent' Gaussian random field $G = \{G(\mathbf{s}),\ \mathbf{s} \in \mathfrak{D}\}$, where $\mathbf{s}$ represents a location in the domain $\mathfrak{D}$. In this tutorial we consider the spatial case *i.e.* $\mathfrak{D} \subseteq \mathbb{R}^2$. However, the package `GeoModels` allows to work also with spatio-temporal data or data defined on a sphere of arbitrary radius. The Gaussian field G is assumed weakly stationary with zero mean, unit variance and correlation function $\rho(\mathbf{h}) = \mathrm{cor}(G(\mathbf{s}+\mathbf{h}), G(\mathbf{s}))$.

Let us consider the continuous, strictly monotonic function $S_{a,b} : \mathbb{R} \to \mathbb{R}$ defined as

$$S_{a,b}(y) = \sinh\big(b \sinh^{-1}(y) - a\big), \tag{1}$$

with $b > 0$ and $a \in \mathbb{R}$. Specifically, we define $Y^*_{\eta,\kappa} = \{Y^*_{\eta,\kappa}(\mathbf{s}),\ \mathbf{s} \in \mathfrak{D}\}$ with $Y^*_{\eta,\kappa}(\mathbf{s}) := S_{-\frac{\eta}{\kappa},\frac{1}{\kappa}}\big(G(\mathbf{s})\big)$, a standard SAS random field. One particularly appealing property of the SAS transformation is that its parameters are clearly interpretable. The parameters η and κ can be interpreted as skewness and kurtosis parameters, respectively. If they are studied

separately, the parameter $\eta \in \mathbb{R}$ controls the distribution's skewness. In particular, $\eta > 0$ and $\eta < 0$ yield a right-skewed and a left-skewed distribution, respectively. The parameter κ controls the tail-weights, where tail-weights decreases with increasing κ. Specifically, $\kappa < 1$ and $\kappa > 1$ yield heavier and lighter tails than the Gaussian distribution, respectively. Additionally, the special case where $\eta = 0$ and $\kappa = 1$ leads to the identity transformation, i.e., yielding the Gaussian random field.

In addition, the SAS RF has marginal distribution given by:

$$f_{Y^*_{\eta,\kappa}}(y) = \kappa\left(\frac{1 + S^2_{\eta,\kappa}(y)}{2\pi\,(1+y^2)}\right)^{1/2} \exp\left(-\frac{1}{2}S^2_{\eta,\kappa}(y)\right) \tag{2}$$

with mean and variance

$$\mathbb{E}\big(Y^*_{\eta,\kappa}(\mathbf{s})\big) = \frac{\sigma \sinh(\eta/\kappa)e^{1/4}}{\sqrt{8\pi}}\left(K_{\frac{\kappa+1}{2\kappa}}(1/4) + K_{\frac{1-\kappa}{2\kappa}}(1/4)\right),$$
$$\mathrm{var}\big(Y^*_{\eta,\kappa}(\mathbf{s})\big) = \frac{\cosh(2\eta/\kappa)e^{1/4}}{\sqrt{32\pi}}\left(K_{\frac{\kappa+2}{2\kappa}}(1/4) + K_{\frac{2-\kappa}{2\kappa}}(1/4)\right) - \frac{1}{2} - \big(\mathbb{E}(Y^*_{\eta,\kappa}(\mathbf{s}))\big)^2,$$

where K_ζ is the modified Bessel function of the second kind of order ζ (Jones and Pewsey, 2009). In addition, the correlation function of the SAS random field can be written as

$$\rho_{Y^*_{\eta,\kappa}(\mathbf{s})}(h;\boldsymbol{\vartheta}) = \sum_{j=1}^{\infty}\frac{\xi_j(\eta,\kappa)^2}{j!}\rho(h;\boldsymbol{\vartheta})^j. \tag{3}$$

A closed form expression for the coefficients $\xi_j(\eta,\kappa)$ can be found in Blasi et al. (2022).

A location and scale transformation gives a non-standard SAS random field $Y_{\eta,\kappa} = \big\{Y_{\eta,\kappa}(\mathbf{s}),\ \mathbf{s} \in \mathfrak{D} \subset \mathbb{R}^d\big\}$, defined as

$$Y_{\eta,\kappa}(\mathbf{s}) = \mu(\mathbf{s}) + \sigma Y^*_{\eta,\kappa}(\mathbf{s}), \tag{4}$$

where $\mu(\mathbf{s}) \in \mathbb{R}$ is a spatially varying location parameter that, can be expressed as $\mu(\mathbf{s}) = X(\boldsymbol{s})^T\boldsymbol{\beta}$, and $\sigma > 0$ is a scale parameter. $X(\boldsymbol{s})$ is a k-dimensional vector of covariates (the first entry is one for the intercept) and $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_k)^T$ is a k-dimensional vector of (unknown) parameters. In this tutorial we assume $k = 2$.

To obtain a simulation from $Y_{\eta,\kappa}(\mathbf{s})$ we need to specify regression mean (in this example we specify two regression parameters), skew, tail and variance parameters *i.e.* β_1, β_2, η, κ and σ^2 respectively. Additionally, we need to specify a parametric correlation $\rho(\mathbf{h})$ for the 'parent' Gaussian random field. We first set the spatial coordinates:

```
N=1500;
coords=cbind(runif(N),runif(N));
plot(coords ,pch=20,xlab="",ylab="");
```

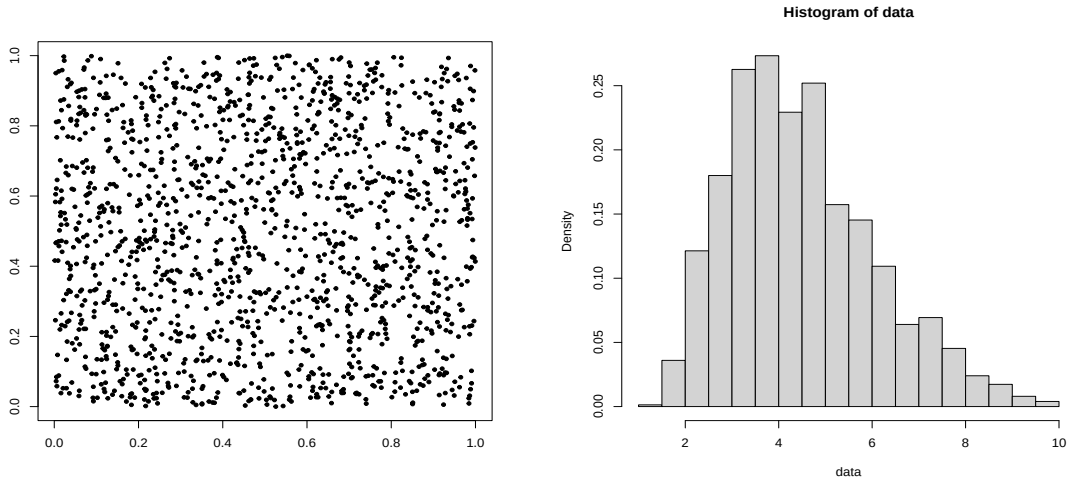

Figure 1: From left to right: Spatial location sites used in the tutorial and histogram of realization SAS random fields.

For the correlation function $\rho(\boldsymbol{h})$ of the 'parent' Gaussian random field G we assume an isotropic Matérn model (Matérn, 1986):

$$\rho_{\alpha,\gamma}(\boldsymbol{h}) = \frac{2^{1-\gamma}}{\Gamma(\gamma)} \left(\frac{\|\boldsymbol{h}\|}{\alpha} \right)^{\gamma} \mathcal{K}_{\gamma} \left(\frac{\|\boldsymbol{h}\|}{\alpha} \right), \qquad \|\boldsymbol{h}\| \geq 0. \tag{5}$$

where $\mathcal{K}_{\gamma}$ is a modified Bessel function of the second kind of order γ, $\gamma > 0$ is the smoothness parameter and $\alpha > 0$ the spatial scale parameter. Then, we set the parameter associated to this correlation model:

```
corrmodel = "Matern";          ## correlation model
scale = 0.15/3;                 ## scale parameter
smooth=0.5;                    ## smooth parameter
nugget=0;                      ## nugget parameter
```

and we set the skew, tail and variance parameters of the SAS random field:

```
skew=3.5;   ## skew parameter
tail=2;     ## tail parameter
sill= 1.5; ## variance parameter
```

Finally we set the mean regression parameters and the regression matrix:

```
mean = 1; mean1= -1 # regression paramteres
a0=rep(1,N);a1=runif(N,-1,1)
X=cbind(a0,a1); ## regression matrix
```

We are now ready to simulate a realization of the SAS random field $Y_{\eta,\kappa}(\mathbf{s})$ using the function `GeoSim`. Simulation is performed exploiting the stochastic representation (4), where the underlying the Gaussian field is generated with Cholesky decomposition. Another way to simulate SAS random fields is by using two approximate simulation methods—circulant embedding and turning bands—through the function `GeoSimApprox` from the `GeoModels` package (Bevilacqua et al. (2025)).

```
param=list(nugget=nugget,mean=mean,mean1=mean1, scale=scale,
           smooth=smooth, sill=sill,skew=skew,tail=tail);
data <- GeoSim(coordx=coords ,corrmodel=corrmodel,
          param=param ,model=model ,X=X)$data;
hist(data,prob=TRUE,breaks=20)
```

The marginal distribution of the data (see the histogram in the right panel of Figure 1) exhibits a certain degree of asymmetry and heavy tails as expected.

Estimation of SAS random fields

Given $\{\mathbf{s}_1,\ldots,\mathbf{s}_n\} \in \mathfrak{D}$ a set of distinct locations, let $\boldsymbol{\Omega} = [\rho(||\mathbf{s}_i - \mathbf{s}_j||;\boldsymbol{\psi})]_{i,j=1}^n$, the correlation matrix associated with a parametric correlation model of the process $G(\mathbf{s})$, can be performed using the `GeoFit` and `GeoFit2` functions. Let $S_{\eta,\kappa}(\mathbf{s})$ be the componentwise SAS transformation of the elements of the vector $\boldsymbol{G}$, where $\boldsymbol{G} = \big(G(\mathbf{s}_1),\ldots,G(\mathbf{s}_n)\big)^\top$ is a Gaussian random vector, where $G(\mathbf{s}_i) = S_{\eta,\kappa}\big(\sigma^{-1}(y_i - \mu_i)\big)$ for $i = 1,\ldots,n$. Thus, maximum likelihood estimation that is the maximization of the log-likelihood function associated to the random vector $\boldsymbol{Y} = \big(Y_{\eta,\kappa}(\mathbf{s}_1),\ldots,Y_{\eta,\kappa}(\mathbf{s}_n)\big)^\top$ is given by

$$
\begin{aligned}
l(\boldsymbol{\vartheta};\mathbf{y}) = &-\frac{n}{2}\log(2\pi\sigma^2) - \frac{1}{2}\log\det\boldsymbol{\Omega}(\boldsymbol{\psi}) - \frac{1}{2}S_{\eta,\kappa}\big(\sigma^{-1}(\mathbf{y}-\mathbf{X}\boldsymbol{\beta})\big)^\top\boldsymbol{\Omega}(\boldsymbol{\psi})^{-1}S_{\eta,\kappa}\big(\sigma^{-1}(\mathbf{y}-\mathbf{X}\boldsymbol{\beta})\big)\\
&+ n\log(\kappa) - \frac{1}{2}\sum_{i=1}^{n}\Big(\log\Big(1+S^2_{\eta,\kappa}\big(\sigma^{-1}(y_i-\mathbf{x}_i\boldsymbol{\beta})\big)\Big) - \log\Big(1+\big(\sigma^{-1}(y_i-\mathbf{x}_i\boldsymbol{\beta})\big)^2\Big)\Big),
\end{aligned}
\tag{6}
$$

where $S_{\eta,\kappa}(\mathbf{y}) = [S_{\eta,\kappa}(y_i)]_{i=1}^n$ and $\boldsymbol{\vartheta} = (\boldsymbol{\beta}^\top,\boldsymbol{\psi}^\top,\sigma,\eta,\kappa)^\top \in \mathbb{R}^{k+m+3}$. If $\eta = 0$ and $\kappa = 1$, (6) simplifies to the Gaussian log-likelihood.

The pairwise likelihood estimator $\hat{\boldsymbol{\vartheta}}$ is obtained maximizing (6) with respect to $\boldsymbol{\vartheta}$. In the `GeoModels` package, we can choose the fixed parameters and the parameters that can be estimated.

Maximum likelihood estimation can be performed using the function `GeoFit2`. In this example, we perform optimization of (6) using the function `nlminb` that allows box-constrained optimization using PORT routines. However other type of optimization algorithms available in R can be used (`BFGS` or `Nelder-Mead` for instance).

```
optimizer="nlminb";
fixed<-list(nugget=nugget,smooth=smooth);
start<-list(mean=mean, mean1=mean1,scale=scale,sill=sill,
            skew=skew,tail=tail);
I=Inf;
lower<-list(mean=-I, mean1=-I,scale=0,sill=0,skew=-I,tail=0);
upper<-list(mean=I, mean1=I,scale=I,sill=I,skew=I,tail=I);
fit1 <- GeoFit2(data=data,coordx=coords,corrmodel=corrmodel,X=X,
               model=model,optimizer=optimizer,lower=lower,
               upper=upper,likelihood="Full",type="Standard",
               varest=TRUE,start=start,fixed=fixed)
```

The object `fit1` includes informations about the maximum likelihood estimation

```
fit1
#########################################################################
Maximum  Likelihood Fitting of SinhAsinh Random FieldsSetting:
Full Likelihood
Model: SinhAsinh
Distance: Eucl
Type of the likelihood objects: Standard
Covariance model: Matern
Optimizer: nlminb
Number of spatial coordinates: 1500
Type of the random field: univariate
Number of estimated parameters: 6

Type of convergence: Successful
Maximum log-Likelihood value: -1857.18
AIC : 3726
BIC : 3758

Estimated parameters:
```

```
    mean      mean1      scale       sill       skew       tail
 2.04476   -1.00415    0.04624    1.71296    1.98039    1.53782
Standard errors:
    mean      mean1      scale       sill       skew       tail
0.380464   0.021201   0.004139   0.424051   0.478868   0.193636
###################################################################
```

An alternative and much faster method of estimation is the weighted composite likelihood based on marginal pairs. Denote the log density of the bivariate random vector $(Y_{\eta,\kappa}(\boldsymbol{s}_i), Y_{\eta,\kappa}(\boldsymbol{s}_j))^\top$ by $l_{ij}(\boldsymbol{\vartheta}; \mathbf{y})$. Then the weighted pairwise likelihood function is defined as:

$$pl(\boldsymbol{\vartheta}) = \sum_{i=1}^{N} \sum_{j=1}^{N} l_{ij}(\boldsymbol{\vartheta}; \mathbf{y}) w_{ij}, \tag{7}$$

where w_{ij} are non-negative weights, not depending on $\boldsymbol{\vartheta}$. An efficient way to specify the weights from computational and efficient viewpoint is based on neighborhoods (Caamaño-Carrillo et al. (2024)):

$$w_{ij}(k) = \begin{cases} 1 & \mathbf{s}_i \in N_k(\mathbf{s}_j) \\ 0 & \text{otherwise.} \end{cases} \tag{8}$$

Here $N_k(\mathbf{s}_l)$ is the set of the neighbors of order $k = 1, 2, \ldots$ of the point $\mathbf{s}_l$ and in this case $\boldsymbol{\vartheta} = (\boldsymbol{\beta}^\top, \boldsymbol{\psi}^\top, \sigma^2, \eta, \kappa)^\top$. The pairwise likelihood estimator $\hat{\boldsymbol{\vartheta}}_{pl}$ is obtained maximizing (7) with respect to $\boldsymbol{\vartheta}$. In the `GeoModels` package, we can choose the fixed parameters and the parameters that can be estimated.

Pairwise likelihood estimation can be performed using the function `GeoFit2`. In this example, we perform optimization of (7) using the function `nlminb`. We use the following code to estimate the parameters $\boldsymbol{\vartheta}$ of the SAS random field (the option `neighbs` set the order of neighbors k in the weight function (8)).

```
fit2 <- GeoFit2(data=data,coordx=coords,corrmodel=corrmodel,X=X,
                model=model,optimizer=optimizer,lower=lower,
                upper=upper,neighb=2,likelihood="Marginal",
                type="Pairwise",sensitivity=TRUE,start=start,
                fixed=fixed)
```

The object `fit2` include informations about the conditional pairwise likelihood estimation:

```
fit2
###################################################################
```

```
Maximum  Composite-Likelihood Fitting of SinhAsinh Random Fields
Setting: Marginal Composite-Likelihood
Model: SinhAsinh
Distance: Eucl
Type of the likelihood objects: Pairwise
Covariance model: Matern
Optimizer: nlminb
Number of spatial coordinates: 1500
Type of the random field: univariate
Number of estimated parameters: 6

Type of convergence: Successful
Maximum log-Composite-Likelihood value: -8884.95

Estimated parameters:
    mean     mean1     scale      sill      skew      tail
 2.18161  -0.98421   0.04628   1.75475   1.77453   1.47713
###################################################################
```

The standard error estimation can be performed using parametric bootstrap using the `GeoVarestbootstrap` function of the `GeoModels` package.

```
fit2=GeoVarestbootstrap(fit2,K=500,parallel=TRUE)
```

Standard error estimates and interval confidence for both methods of estimation can be obtained with as

```
 round(fit1$stderr,5);round(fit2$stderr,5)
   mean    mean1    scale     sill     skew     tail
0.38046  0.02120  0.00414  0.42405  0.47887  0.19364
   mean    mean1    scale     sill     skew     tail
0.40349  0.02291  0.00466  0.60544  0.60791  0.21759
round(fit1$conf.int,5);round(fit2$conf.int,5)
       mean     mean1    scale     sill     skew     tail
low  2.02091  -1.00548  0.04598  1.68637  1.95036  1.52568
upp  2.06862  -1.00282  0.04650  1.73956  2.01042  1.54996
         mean     mean1    scale     sill     skew     tail
Lower  1.39079  -1.02912  0.03714  0.56810  0.58305  1.05067
Upper  2.97244  -0.93930  0.05542  2.94139  2.96600  1.90359
```

As expected standard errors for maximum likelihood are slightly smaller than the standard errors for the composite likelihood.

Checking model assumptions

Hereafter we consider the ML estimation for checking assumption and prediction. Given the estimation of the mean regression and sill parameters, the estimated residuals

$$\widehat{Y^*_{\eta,\kappa}(\mathbf{s}_i)} = \frac{y(\mathbf{s}_i) - X(\mathbf{s}_i)^\top \widehat{\boldsymbol{\beta}}}{(\widehat{\sigma}^2)^{\frac{1}{2}}} \quad i = 1, \dots N$$

can be viewed as a realization of the process $Y^*_{\eta,\kappa}(\mathbf{s})$. The residuals can be computed using the `GeoResiduals` function:

```
res=GeoResiduals(fit1); # computing residuals
```

The marginal distribution assumption on the residuals can be graphically checked trough a qq-plot using the `GeoQQ` function (see Figure 2, left part):

```
### checking model  residuals assumptions: marginal distribution
GeoQQ(res)
```

The covariance model assumption can be checked comparing the empirical and the estimated semi-variogram using the `GeoVariogram` and `GeoCovariogram` functions (see Figure 2, right part):

```
### checking model  residuals assumptions: covariance model
vario <- GeoVariogram(data=res$data,coordx=coords,maxdist=0.4);
GeoCovariogram(res,show.vario=TRUE, vario=vario,pch=20);
```

The semi-variogram is computed using the correlation function (3).

Prediction of SAS random fields

For a given spatial location $\boldsymbol{s}_0$ with associated covariates $X(\boldsymbol{s}_0)$, the optimal linear prediction of a SAS random field is given by:

$$\widehat{Y}_{\eta,\kappa}(\mathbf{s}_0) = X(\mathbf{s}_0)^\top \boldsymbol{\beta} + \sum_{i=1}^{N} \lambda_i [y(\mathbf{s}_i) - X(\mathbf{s}_i)^\top \boldsymbol{\beta}] \tag{9}$$

where the vector of weights $\boldsymbol{\lambda} = (\lambda_1, \dots, \lambda_N)^\top$ is given by $\boldsymbol{\lambda} = R_h^{-1} \boldsymbol{c}_{\eta,\kappa}$ and

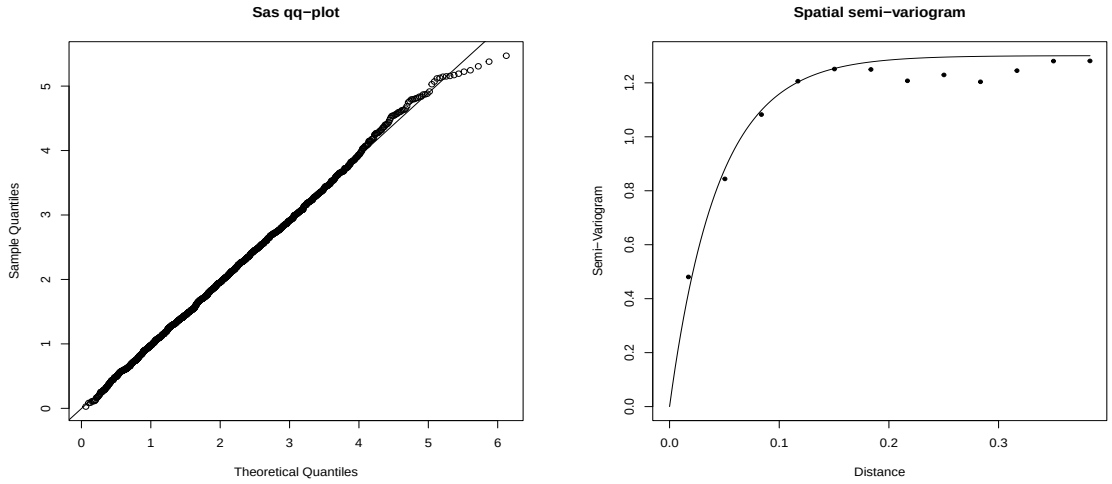

Figure 2: From left to right: qq-plot of the residuals using the $Y^*_{1.98,1.53}$ distribution and empirical vs estimated semi-variogram for the residuals.

- $\boldsymbol{c}_{\eta,\kappa} = (cor(Y_{\eta,\kappa}(\mathbf{s}_0), Y_{\eta,\kappa}(\boldsymbol{s}_1)), \ldots, cor(Y_{\eta,\kappa}(\mathbf{s}_0), Y_{\eta,\kappa}(\mathbf{s}_N)))^T$.
- $R_{\eta,\kappa} = [cor(Y_{\eta,\kappa}(\mathbf{s}_i), Y_{\eta,\kappa}(\mathbf{s}_j))]]_{i,j=1}^N$ is the correlation matrix.

Moreover the associated mean square error (MSE) is given by:

$$MSE(\widehat{Y}_{\eta,\kappa}(\mathbf{s}_0)) = \sigma^2 \text{var}\big(Y^*_{\eta,\kappa}(\mathbf{s})\big)(1 - \boldsymbol{c}^\top_{\eta,\kappa} R^{-1}_{\eta,\kappa} \boldsymbol{c}_{\eta,\kappa}). \tag{10}$$

The predictor can be viewed as an optimal Gaussian predictor assuming (3) as correlation function. If the parameters are unknown, both (9) and (10) can be computed replacing the parameters with the pairwise likelihood estimates. In particular, $R_{\eta,\kappa}$ and $\boldsymbol{c}_{\eta,\kappa}$ can be computed using (3) coupled with the estimates of the Matérn correlation function and of the degrees of freedom.

Kriging and associated MSE can be obtained using the `GeoKrig` function. We first need to specify the spatial locations to predict and, in this example, we consider a spatial regular grid:

```
xx=seq(0,1,0.012)
loc_to_pred=as.matrix(expand.grid(xx,xx))
Nloc=nrow(loc_to_pred)
Xloc=cbind(rep(1,Nloc),runif(Nloc))
```

Then the optimal linear prediction (9), using the estimated parameters, can be performed using the `GeoKrig` function (computation can be time consuming):

```
pr=GeoKrig(fit1,loc=loc_to_pred,Xloc=Xloc,mse=TRUE)
```

Finally, a kriging map with associate mean square error (Figure 3) can be obtained with the following code:

```
par(mfrow=c(1,3));
#### map of  data
quilt.plot(coords[,1], coords[,2], data,main="Data")
# linear kriging
map=matrix(pr$pred,ncol=length(xx))
image.plot(xx,xx,map,xlab="",ylab="",main="SimpleKriging")
#associated mean squared error
map_mse=matrix(pr$mse,ncol=length(xx))
image.plot(xx,xx,map_mse,xlab="",ylab="",main="MSE")
```

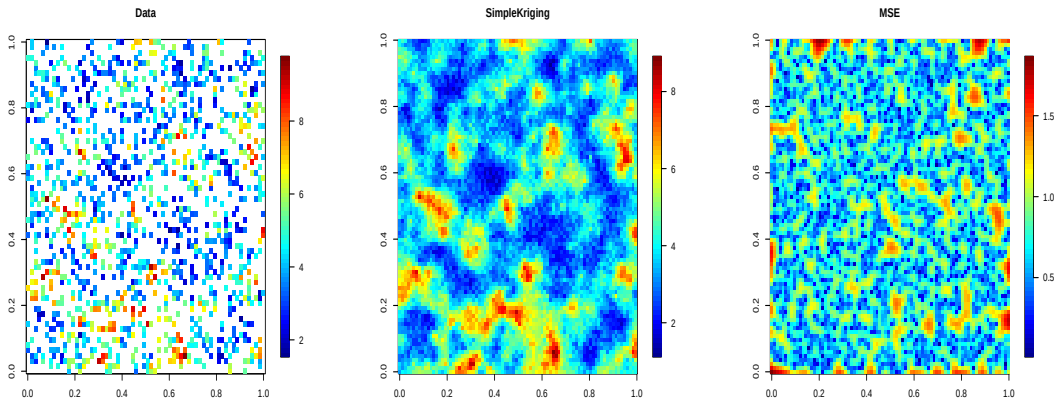

Figure 3: From left to right: observed spatial data, associated kriging map and mean square error map.

References

Bevilacqua, M., V. Morales-Oñate, C. Caamaño-Carrillo, and F. Cuevas-Pacheco (2025). *GeoModels: Procedures for Gaussian and Non Gaussian Geostatistical (Large) Data Analysis*. R package version 2.1.8.

Blasi, F., C. Caamaño-Carrillo, M. Bevilacqua, and R. Furrer (2022). A selective view of climatological data and likelihood estimation. *Spatial Statistics 50*, 100596. Special Issue: The Impact of Spatial Statistics.

Caamaño-Carrillo, C., M. Bevilacqua, C. López, and V. Morales-Oñate (2024). Nearest neighbors weighted composite likelihood based on pairs for (non-)gaussian massive spatial data with an application to tukey-hh random fields estimation. *Computational Statistics and Data Analysis 191*, 107887.

Jones, M. C. and A. Pewsey (2009). Sinh-arcsinh distributions. *Biometrika 96(4)*, 761–780.

Matérn, B. (1986). *Spatial Variation: Stochastic Models and their Applications to Some Problems in Forest Surveys and Other Sampling Investigations* (2nd ed.). Heidelberg: Springer.

Xua, G. and M. G. Genton (2017). Tukey g-and-h random fields. *Journal of the American Statistical Association 112*, 1236 –1249.