Proceedings of the 3rd International Workshop on Distributed Statistical Computing (DSC 2003)
March 20–22, Vienna, Austria *ISSN 1609-395X*
Kurt Hornik, Friedrich Leisch & Achim Zeileis (eds.)
http://www.ci.tuwien.ac.at/Conferences/DSC-2003/

Gstat: Multivariable Geostatistics for S

Edzer J. Pebesma

Abstract

This paper introduces the `gstat` package for the S language (R, S-PLUS). The package provides multivariable geostatistical modelling, prediction and simulation, as well as several visualisation functions. Gstat (`http://www.gstat.org`) was started 10 years ago and was released under the GPL in 1996; it is closely linked to several GIS systems. It was not initially written for teaching purposes, but for research purposes, emphasizing flexibility, scalability and portability. It can deal with a large number of practical issues in geostatistics, including change of support (block kriging), simple/ordinary/universal (co)kriging, fast local neighbourhood selection, flexible trend modelling, variables with different sampling configurations, and efficient simulation of large spatially correlated random fields, indicator kriging and simulation, and (directional) variogram and cross variogram modelling. The formula/models interface is used to define multivariable geostatistical models. This paper presents the functionality provided by the `gstat` S package, discusses a number of design and implementation issues, and advantages and shortcomings of the S environment for multivariable geostatistics.

1 Introduction

Geostatistics is not a new subject in the S community, and several packages or libraries are available. Some of these were developed for teaching purposes, and some have have very advanced, cutting edge functionality. Still, all of them lack a number of features that are commonly used in applied geostatistics (e.g., Isaaks and Srivastava, 1989), notably block kriging, kriging in a local neighbourhood, and multivariable variogram modelling, kriging and simulation.

Gstat (Pebesma and Wesseling, 1998) used to be a stand-alone computer program that provides all these features, but that has no graphics capabilities of its own. It has an interactive user interface for variogram modelling, but uses gnuplot for visualizing the variograms. Also, it can read and write point data and raster map data to and from 7 different geographic information systems (among which GRASS,

PCRaster, and GMT); graphical user interfaces that use gstat as a back-end have been developed within the Idrisi and ArcInfo environments.

The S (R/S-PLUS) environment has much to offer to a multivariable geostatistics program without graphics capabilities, especially with the Trellis/`lattice` graphics environment. The `gstat` S package, introduced in this paper, offers most of the functionality of the gstat stand-alone program to S users, and provides a number of useful wrapper functions to plot spatial point data, multiple grid maps, and multivariable or directional variograms.

2 Multivariable geostatistics

2.1 The univariable model

Let $Z(s)$ be a vector of length n with observations $Z(s_1), ..., Z(s_n)$ observed at spatial locations s_i arbitrarily spread in R^1, R^2 or R^3. The variability in observations $Z(s)$ is usually thought of as consisting of a trend and a residual, and the trend is modelled as a linear function,

$$Z(s) = \sum_{j=0}^{p} X_j(s)\beta_j + e(s) = X\beta + e(s) \tag{1}$$

with $X_j(s), j > 0$, the p explanatory or predictor variables, with β_0 usually being an intercept and $X_0(s) \equiv 1$, with β the vector with unknown regression coefficients, and with $e(s)$ the residual vector. For spatial data, residuals are usually spatially correlated, and given the covariance matrix V of $e(s)$, best linear unbiased prediction (kriging) of $Z(s_0)$ at an unobserved location s_0 is obtained by

$$\hat{Z}(s_0) = x(s_0)\hat{\beta} + v'V^{-1}(Z(s) - X\hat{\beta}) \tag{2}$$

with $x(s_0)$ the row of X that would have corresponded to $Z(s_0)$, with $\hat{\beta} = (X'V^{-1}X)^{-1}$ $X'V^{-1}Z(s)$ the generalised least squares estimate of the trend coefficients where X' denotes the transpose of X, and with $v = (\mathrm{Cov}(Z(s_0), Z(s_1)), ..., \mathrm{Cov}(Z(s_0), Z(s_n)))'$ where $\mathrm{Cov}(\cdot,\cdot)$ denotes covariance.

The corresponding prediction error variance is

$$\sigma^2(s_0) = \sigma_0^2 - v'V^{-1}v + (x(s_0) - v'V^{-1}X)(X'V^{-1}X)^{-1}(x(s_0) - v'V^{-1}X)' \tag{3}$$

where σ_0^2 is $\mathrm{Var}(Z(s_0))$.

2.2 The multivariable model

Multivariable prediction involves multiple, spatially and cross-variable correlated variables. Consider m distinct variables, and let $Z_i(s), X_i, \beta^i, e_i(s), x_i(s_0), v_i$ and V_i all correspond to the i-th variable. Next, let $\mathbf{Z}(s) = (Z_1(s)', ..., Z_m(s)')'$, $\mathbf{B} = (\beta^{1\prime}, ..., \beta^{m\prime})'$, $\mathbf{e}(s) = (e_1(s)', ..., e_m(s)')'$,

$$\mathbf{X} = \begin{bmatrix} X_1 & 0 & ... & 0 \\ 0 & X_2 & ... & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & ... & X_m \end{bmatrix}, \quad \mathbf{x}(s_0) = \begin{bmatrix} x_1(s_0) & 0 & ... & 0 \\ 0 & x_2(s_0) & ... & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & ... & x_m(s_0) \end{bmatrix}$$

with 0 conforming zero matrices, and

$$\mathbf{v} = \begin{bmatrix} v_{1,1} & v_{1,2} & \dots & v_{1,m} \\ v_{2,1} & v_{2,2} & \dots & v_{2,m} \\ \vdots & \vdots & \ddots & \vdots \\ v_{m,1} & v_{m,2} & \dots & v_{m,m} \end{bmatrix}, \quad \mathbf{V} = \begin{bmatrix} V_{1,1} & V_{1,2} & \dots & V_{1,m} \\ V_{2,1} & V_{2,2} & \dots & V_{2,m} \\ \vdots & \vdots & \ddots & \vdots \\ V_{m,1} & V_{m,2} & \dots & V_{m,m} \end{bmatrix}$$

where element i of $v_{k,l}$ is $\mathrm{Cov}(Z_k(s_i), Z_l(s_0))$, and where element (i,j) of $V_{k,l}$ is $\mathrm{Cov}(Z_k(s_i), Z_l(s_j))$.

The multivariable prediction equations equal (2) and (3) when all matrices are substituted by their multivariable (bold case) forms (see also Ver Hoef and Cressie, 1993), and when in (3) σ_0^2 is substituted by Σ with $\mathrm{Cov}(Z_i(s), Z_j(s))$ in its (i,j)-th element. Note that (3) is now a prediction error covariance matrix.

The implementation of this model in `gstat` does not pose restrictions to the number of variables m, and each variable can have its own set of predictor variables, number of observations and unique observation locations. Covariances are specified by ways of variogram functions and cross variogram functions.

Gstat provides a number of highly useful extensions to the straightforward application of (2) and (3):

Kriging in a local neighbourhood Instead of using all data, only data in a local neighbourhood around s_0 are used for predicting $\mathbf{Z}(s_0)$, where neighbourhood can be defined in terms of distance to s_0 or in terms of the number of nearest observations. There are two good reasons for restricting kriging to a local neighbourhood. First, the system $V^{-1}X$ becomes prohibitively large when data are abundant ($n \gg 10^3$) or when sequential simulation is used to simulate large fields. Second, the assumption of spatially constant trend coefficients in (1) may need to be relaxed to apply only to local neighbourhoods. Gstat takes care of cases where one or more of the variables are missing in a local neighbourhood, defined by a distance criterion.

Block kriging Instead of predicting $Z(s_0)$ (point kriging), block kriging aims at predicting the average of $Z(\cdot)$ over a larger support (area or volume) B_0: $Z(B_0) = |B_0|^{-1} \int_{B_0} Z(s)ds$, with $|B_0|$ the area (or volume) of B_0. Blocks B_0 may be rectangular or irregular (specified by a number of points discretizing B_0). Although the interest was originally limited to mining applications, block kriging is now widely used in environmental applications when spatially aggregated predictions for larger areas are required, or when point support predictions are too inaccurate.

Simple and ordinary kriging In certain cases, the trend coefficients can be assumed known e.g. when an other mechanism, such as an external deterministic model takes care of estimating them. In this case, called simple kriging, β is substituted for $\hat{\beta}$ in (2), and the third term on the right hand side of (3) disappears. Another simplified version of universal kriging is ordinary kriging, which contains only an intercept ($p = 0$).

Shared trend coefficients Suppose a single variable is measured by two different devices, each having its own noise characteristics. In this case, their variability can be thought of as consisting of a common trend and separate residual characteristics. Examples are found in Isaaks and Srivastava (1989, pp 409-416),

or references to collocated cokriging found in Goovaerts (1997) or Wackernagel (1998).

Debugging results Near-singularities may occur for a number of reasons, such as near-zero distances between data points, or linear dependencies among columns of a (locally formed) matrix X. Gstat has many debug modes for obtaining information on all aspects of the systems, and can verify that estimated condition numbers of V and $X'V^{-1}X$ stay below a user-specified threshold.

2.3 Sequential simulation

Sequential simulation (Johnson, 1987; Gómez-Hernández and Journel, 1993) involves the generation of many independent realisations of a Gaussian (or in case of indicator simulation, binary) random field, conditional to observed data, that honour the variogram (covariance) of the random field. Gstat uses the sequential simulation algorithm because it is versatile, efficient, and suitable for large to very large fields (number of nodes $\gg 10^6$).

Traditionally, simulation algorithms only involved the simulation of the residual part of (1), although some attempts to stretch this have been reported (Goovaerts, 1997). This can be seen as the simulation equivalent of simple kriging. Gstat implements a wider class that also addresses estimation errors of the trend coefficients, and uses an algorithm that was reported (although somewhat hidden) in Abrahamsen and Espen Benth (2001). It involves the simulation of trend coefficients drawn from $N(\hat{\beta}, (X'V^{-1}X)^{-1})$, followed by simulating residuals with respect to the trend coefficients drawn. It is the simulation equivalent of universal kriging.

2.4 Variogram modelling

All methods mentioned above assume that the residual covariance is known. A common convention is to enter the covariance by ways of the variogram. Gstat calculates direct sample variograms, cross variograms ("classical" cross variograms for variables that have identical locations, pseudo-cross variograms (Ver Hoef and Cressie, 1993) when they don't), and can fit nested variogram models to sample variograms. In fitting direct and cross variogram models, it can also guarantee that the fitted model obeys the linear model of coregionalisation (Goovaerts, 1997), ensuring that cross covariance matrices are positive definite. Furthermore, `gstat` can calculate and visualize directional variograms, variogram clouds, and provides identification through interactive examination (for example of extreme points) in the variogram cloud.

Variogram models may consist of the sum of one or more basic models, that include the Nugget, Exponential, Spherical, Gaussian, Linear, Power model. Each basic model can have its own 2D or 3D geometric or zonal anisotropy parameters defined. The `gstat` S package also includes the Matérn class, strongly recommended by Stein (1999), but does not (yet, as of version 0.9-4) fit its parameters.

3 Implementation

3.1 User interface

The `gstat` S package provides a formula-based interface such as found in `lm()` and its family. One formula is used to define how the response depends on the predictor variables, and a second formula defines the spatial coordinates. Calculating a residual variogram of `log(zinc)` as a function of `dist` with spatial coordinates in `x` and `y`, found in data frame `meuse`, looks like:

```
variogram(log(zinc)~dist, ~x+y, meuse)
```

Univariate universal kriging on locations defined in `meuse.grid`, using a fitted (residual) variogram model `vgm.mod` is obtained by

```
krige(log(zinc)~dist, ~x+y, meuse, meuse.grid, vgm.mod)
```

and 50 conditional simulations are obtained by

```
krige(log(zinc)~dist, ~x+y, meuse, meuse.grid, vgm.mod,
        nmax = 20, nsim = 50)
```

where `nmax` refers to the neighbourhood size, needed by the sequential simulation algorithm.

For multivariable prediction or simulation, we need to specify for each variable at least two formula's and a data frame. All this information is stored in a (nested) list of class `gstat`, which is built one variable at a time, by a function (surprisingly) called `gstat`:

```
meuse.g <- gstat(id = "log-zn", formula = log(zinc)~1, locations = ~x+y,
         data = meuse)
meuse.g <- gstat(meuse.g, id = "log-cu", log(copper)~1, ~x+y, meuse)
...
```

that can accumulate an arbitrary number of variables. Suppose `meuse.g` is filled with the four heavy metal variables measured in the `meuse` data set, then the four commands

```
meuse.g <- gstat(meuse.g, model=vgm(1, "Sph", 900, nugget = 1), fill.all=T)
x <- variogram(meuse.g, cutoff=1000)
meuse.fit = fit.lmc(x, meuse.g)
plot(x, model = meuse.fit)
meuse.cok <- predict(meuse.fit, newdata = meuse.grid)
```

(i) fill all variogram models with the same initial (Nugget + Spherical) variogram model, (ii) calculate sample variograms and cross variograms, (iii) fit a linear model of coregionalisation to direct and cross variograms, (iv) plot the variograms and fitted models (Figure 1), and (v) store four-variate cokriging predictions and prediction error (co)variances in `meuse.cok`.

The prediction function, `predict.gstat`, is the prediction and simulation engine of `gstat`. Depending on the data it is fed with, it decides what to do; Figure 2 shows the decision tree. The list of user functions in package `gstat` is shown in Table 1

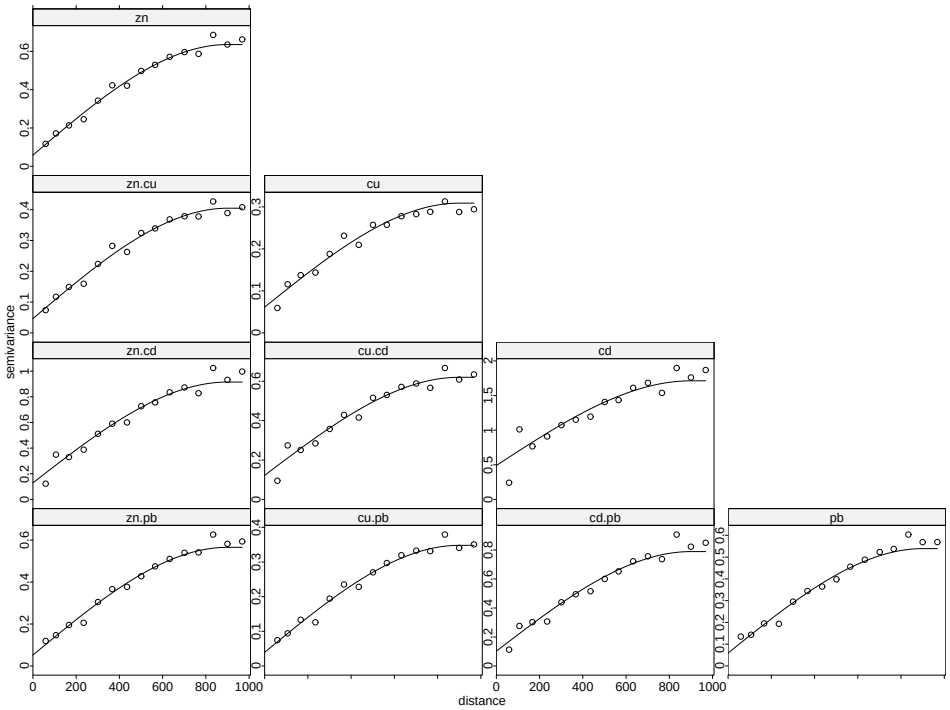

Figure 1: direct sample variograms (diagonal), cross variograms (off-diagonal) and fitted linear model of coregionalisation for the four heavy metal variables in the `meuse` data set

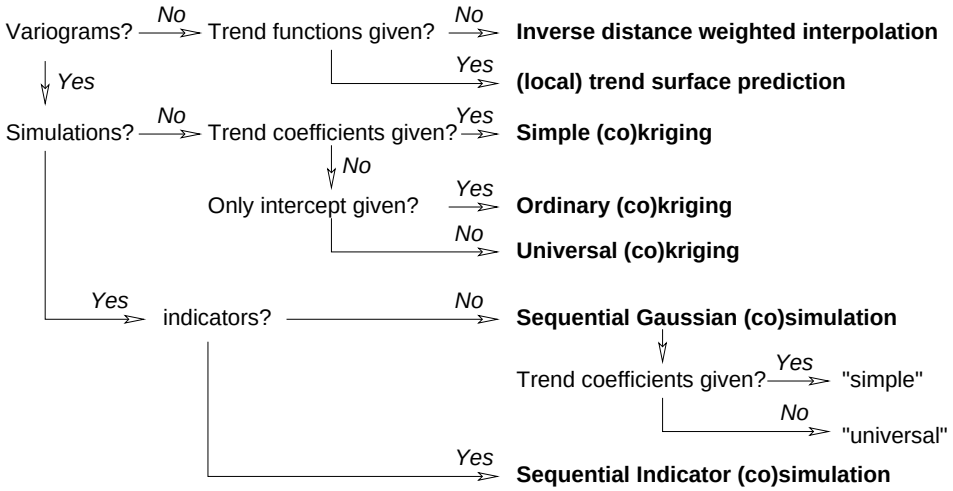

Figure 2: Prediction method decision tree of `predict.gstat` (or `krige`); each of the prediction methods may apply to points, rectangular blocks, or irregular blocks, and may use all data or a selection of local data in a local neighbourhood around each prediction location

3.2 C code

The gstat C code used for the `gstat` package consists of approximately 25000 lines of "native" gstat code, and 14000 lines of C code in the Meschach matrix library (Stewart and Leyk, 1994) used by `gstat`. Because originally gstat was written as a stand alone program (Pebesma and Wesseling, 1998), a large part of the effort of writing a `gstat` package was dedicated towards making the code suitable as a callable library. This involved removing many static variables, re-initialising the full state of the library after every call from S, and writing wrapper functions around log, warning and error messages.

Two important optimizations are implemented in the gstat C code. The first is a fast neighbourhood search algorithm, based on the PR-bucket quadtree search index structure (Hjaltason and Samet, 1995). The second is the realisation of many simulated random fields in a single call following a single random path through the simulation locations, re-using the expensive results, i.e. the neighbourhood selection and $V^{-1}X$.

All variogram models are defined in the `gstat` package are in the gstat C code (although the Matérn class uses functions from `libR`), and provides not an easy way to use variogram functions defined in S. Adding a function to the gstat C code is a fairly straightforward, though.

4 Relation to other geostatistics packages

Ripley (2001) gives a short overview of available R packages for spatial statistics. Geostatistics packages include `spatial`, `sgeostat`, `geoR`, and `RandomFields`. Most of these packages provide variogram modelling, trend surface analysis and/or universal kriging. None of them provides kriging in a local neighbourhood, block kriging, cokriging, or three-dimensional kriging. S-PLUS has a commercial module,

`gstat`	add variable definition to gstat object
variogram modelling:	
`variogram`	calculate sample variogram, directional sample variograms, or direct and cross variograms
`fit.variogram`	fit variogram model coefficients to sample variogram
`fit.lmc`	fit a linear model of coregionalisation to direct and cross variograms
`variogram.line`	calculates variogram values from a variogram model
prediction/simulation:	
`predict.gstat`	spatial prediction or simulation, see figure 2
`krige`	univariable wrapper around `gstat` and `predict.gstat`
`krige.cv`	LOO or n-fold cross validation wrapper for `krige`
`zerodist`	detect observations with the same location
graphics:	
`bubble`	bubble scatter plot for data or residuals (using color for sign, size for value)
`plot.variogram`	plot sample variogram (optional with number of point pairs) and fitted model; uses conditioning plots for directional or multivariable variograms (Figure 1)
`plot.variogram.cloud`	plot variogram cloud, with options for interactive point pairs identification
`plot.point.pairs`	plot point pairs, identified by `plot.variogram.cloud`, in a map
`image.data.frame`	draw image for (x, y, z) values, stored in columns of a data frame
`map.to.lev`	stack data in the form $(x, y, z_1, z_2, ..., z_n)$ to a form, suitable for plotting with `levelplot`
`mapasp`	calculate aspect ratio for geographically correct levelplot

Table 1: user functions in package `gstat`

S+SpatialStats, that provides block kriging. Large parts of the `geoR` code (and its extension `geoRglm`) address the uncertainty of estimated covariance parameters in a Bayesian framework (sometimes called *model-based* kriging), an issue that seems to be relevant especially for smaller data sets (Moyeed and Papritz, 2002).

5 Discussion

5.1 Flexibility

The `gstat` package provides a robust and flexible suite of univariable or multivariable geostatistical methods. From the following five items:

- 1-D, 2-D or 3-D
- point, regular block, or irregular block
- univariable, multiple (uncorrelated), or multivariable (correlated) cokriging
- global kriging or kriging in a local neighbourhood
- (co)kriging, unconditional or conditional (co)simulation

any combination (e.g., 3-D universal irregular block co-simulation) can be obtained by the `gstat` package. Also, routines are available for very fast fitting of large numbers of direct and cross variograms. The objection to cokriging that the modelling of a large number of (cross) variograms is prohibitively tedious can now only be put in the past tense.

5.2 S visualisation

The major reason why S is a suitable environment for doing multivariable geostatistics with `gstat` is its graphics capabilities. The `gstat` package gratefully uses the Trellis/`lattice` functions to visualize its results, notably

- `xyplot` for visualising directional variograms and multivariable (direct and cross) variograms (e.g. Figure 1), and to visualize spatial data and cross validation residuals;
- `levelplot` for visualising (multiple) grid maps, using the `aspect` argument to make them geographically correct (1 km north equals 1 km east, a convention that even S+SpatialStats ignores);
- `image` for fast display of many grid maps; and
- `plot` and `identify` to identify extreme point pairs in a variogram cloud.

The graphics functions in table 1 are no more than simple wrapper functions around the S graphics functions, but may be the most important ones to make an analysis successful.

5.3 S4 classes

The `gstat` package was written using S3 classes, mainly for efficiency (time) reasons. Using S4 classes might make the package more robust, and would definitely be the way to go when maintainers of the major R geostats packages could agree on a set of useful classes, e.g. for spatial data, grid maps, sample variograms, variogram models, and maybe even kriging/simulation specifications.

5.4 Gstat features missing in the S package

The major functionality of stand-alone gstat is made available in the package, but a number of features are missing. Most of them have to do with the lack of (explicit) spatial data structures for grid or vector data. *Stratified mode:* the gstat program has an efficient way of dealing with a stratification, where each stratum has its own data, variogram and prediction locations. *Variogram maps* are two-dimensional variograms, calculated on a regular grid. *Efficient variogram calculation for gridded data:* knowing the gridded topology of data, sample variograms can be calculated in $O(N)$, instead of $O(N^2)$. *Multi-step simulation:* (Gómez-Hernández, Journel, 1993) the gstat code can use a recursively refining random visiting sequence (Pebesma and Wesseling, 1998) for sequential simulation, but needs to know the grid topology of prediction locations; currently a simple random path is chosen. *Edges:* open or closed polygons can be defined to further constrain the search neighbourhood. *Quadrant/octant search neighbourhoods, variogram distance:* are other methods to refine search neighbourhoods based on direction or correlation. *Latin hypercube sampling of Gaussian random fields* (Pebesma and Heuvelink, 1999) should be fairly easy to re-implement in S.

5.5 Handling spatial data in S

Prediction locations are often gridded, and observations sometimes are. As noted above, a number of efficiency gains can be obtained when the grid topology of data, if present, is available to gstat. Storing prediction results as grids (2D matrices) can be wasteful, because large part of the area may be filled with NA's. Currently, `gstat` resolves coordinates and explanatory variables at prediction locations using `model.matrix`, which needs both observation data and prediction locations in data frames. Storing output of `predict.gstat` as grids may be beneficial when they are plotted with `image`, but not when they are plotted with `levelplot`. The conversion of table data to gridded data seems to be $O(N)$ (see function `xyz2img` in package `gstat`).

As a result of the workshop on spatial data handling in R, a group of developers working on spatial statistics will work towards a common class for spatial data. The simples form of this is a data frame that documents which columns store the spatial coordinates. If such a class is agreed upon, it would be trivial to modify the `gstat` package such that the `locations` argument becomes obsolete.

5.6 Gstat S objects

Currently, gstat S objects store the complete data frame for each variable defined. This decision was made for convenience, but may be very inefficient when working with large data sets. Ideally, only references to data frames should be stored, along

with the frame number, such that the reference can be resolved only when the actual data are needed (i.e., during prediction, not while building gstat objects). At time of this writing, work is in progress to resolve this issue.

Acknowledgements

The development of the `gstat` S package was supported financially by the Dutch National Institute for Coastal and Marine Management (RIKZ).

References

Abrahamsen, P., F. Espen Benth, 2001. Kriging with inequality constraints. Mathematical Geology 33 (6), 719–744.

Goovaerts, P., 1997. Geostatistics for Natural Resources Evaluation. Oxford University Press.

Hjaltason, G. and H. Samet, Ranking in spatial databases, 1995. In: Advances in Spatial Databases - 4th Symposium, SSD'95, M. J. Egenhofer and J. R. Herring (eds), Lecture Notes in Computer Science, 951, Springer-Verlag, Berlin, 83-95 See also http://www.cs.umd.edu/~brabec/quadtree/index.html

Isaaks, E., R.M. Srivastava, 1989. An Introduction to Applied Geostatistics. Oxford University Press.

Johnson, M.E., 1987, Multivariate Statistical Simulation. Wiley, New York, 230 pp.

Journel, A.G. and Huijbregts, Ch.J., 1978. Mining Geostatistics. Academic Press, London.

Moyeed, R.A., A. Papritz, 2002. An empirical comparison of kriging methods for nonlinear spatial point prediction. Mathematical Geology 34(4), 365–386.

Pebesma, E.J. and Wesseling, C.G., 1998. Gstat, a program for geostatistical modelling, prediction and simulation. Computers & Geosciences, 24 (1), pp. 17–31; http://www.gstat.org/

Pebesma, E.J., G.B.M. Heuvelink, 1999. Latin hypercube sampling of Gaussian random fields. Technometrics 41 (4), pp. 303–312.

Ripley, B.D. Spatial Statistics in R. R News, Vol 1/2, 14–15.

Stein, M.L., 1999. Interpolation of spatial data: some theory for kriging. Springer, New York.

Stewart, D.E., Z. Leyk, 1994. Meschach: Matrix Computations in C. Proceedings of the Centre for Mathematics and its Applications, Australian National University 32, 240 pp, 1994. Source code available from netlib.

Ver Hoef, Jay M., Noel Cressie, 1993. Multivariable Spatial Prediction. Mathematical Geology, 25 (2), pp. 219–240.

Wackernagel, H., 1998. Multivariate Geostatistics, an introduction with applications; second edition. Springer, Berlin.

Affiliation

Edzer J. Pebesma
Dept. of Physical Geography
Utrecht University,
P.O. Box 80.115
3508 TC Utrecht
Netherlands
E-mail: e.pebesma@geog.uu.nl