MAPPING RADIOACTIVITY FROM MONITORING DATA

AUTOMATING THE CLASSICAL GEOSTATISTICAL APPROACH

Edzer J. Pebesma

Dept of Physical Geography, Faculty of Geosciences, Utrecht University, Netherlands
Correspondence to: e.pebesma@geo.uu.nl

In the context of a comparison of spatial prediction algorithms, we applied the classical geostatistical approach to see how well it would automate, and how well it performed in case of an unexpected anomaly. In case of a test without anomaly, the method performed well. In the anomaly case, automatic variogram modelling was hindered seriously, and in terms of RMSE best results were obtained by using the variogram from the test data without the anomaly. Although the 10 days of available training data showed a strong temporally persistent spatial pattern, cokriging did not improve predictions.

1 INTRODUCTION

Spatial interpolation comparison 2004 (SIC2004) was an open contest, held over the internet (http://www.ai-geostats.org/events/sic2004/), for the comparison of interpolation procedures. An earlier spatial interpolation comparison was held in 1997, as reported by Dubois (2003). The issue of SIC2004 is about "automatic mapping": participants had to estimate levels of natural radioactivity by means of interpolation algorithms that are fully automatic. The automatic mapping procedures were evaluated for the case of anomalies and emergencies.

The variable used in this exercise was natural ambient radioactivity measured in Germany. The data, provided by the German Federal Office for Radiation Protection (BfS), are gamma dose rates reported by means of the national automatic monitoring network (IMIS).

In the frame of SIC2004, a rectangular area was used to select 1008 monitoring stations (from a total of around 2000 stations). For these 1008 stations, 11 days of measurements were selected randomly during the last 12 months up to the start of the exercise (May 1st, 2004) and the average daily dose rates calculated for each day. Hence, we ended up having 11 data sets.

During the training phase (May 1st to Sept 15, 2004), 10 of the 11 data sets were released. During the test phase, (September 4 until October 1, 2004) the eleventh data set was released. Within the test phase, participants were challenged to download the data, and return the predictions for the 808 locations as fast as possible after download, with the emphasis on automatic mapping. Details about the procedure to be used for the automatic mapping should have been given to the jury before downloading the test data.

After October 1, 2004, the evaluation phase started, during which the observed data at the 808 test locations were made available to the participants.

The main objectives of the SIC2004 algorithms were:

1. to generate results that are reliable, that is the estimation errors should be minimum: the primary criteria used to assess the performance of the algorithm will be the Mean Square Error;
2. to generate results in the shortest amount of time (less than 24 hours);
3. to generate results automatically, that is without any human intervention;
4. to be able to deal with anomalies, that is the algorithm should be able to respond properly to situations in which levels reported by the monitoring network exceed usual background levels and present different spatial patterns as it is generally the case with accidents.

Shortly before the test data were released, participants were informed that instead of one, two test sets were to be released, one called ”input” (or first), the other called ”joker” (or second); the latter referring to the fourth objective, presence of anomalies.

This paper tries to find a procedure that meets the first three objectives, and we will look at what happens when the proposed procedure is applied to the joker data, testing the fourth objective. The procedure tries to build on the traditional geostatistical approach, but without human intervention. This choice was motivated by experience, curiosity, and by an ongoing need for software testing and development (notably the gstat program, Pebesma and Wesseling, 1998, here used as the gstat R/S-Plus library; Pebesma, 2004). Unlike the usual order, this paper is organized as the game was played, along the chronology the research took, and findings of the training, test and evaluation phase (including a major mistake with interesting consequences that was made) will be given.

2 THE TRAINING PHASE

2.1 EXPLORATORY DATA ANALYSIS

The training data consist of ten days of daily averaged natural ambient radioactivity measured in Germany, where the ten days were chosen randomly from one year (May 1st, 2003 to May 1st, 2004). Because all measured days have measurements on the same 200 locations, we can easily find out how much of the total variability is due to variation in time, in space or their interaction. For this, we analysed the (saturated) ANOVA model

$$y_{ij} = \mu + A_i + B_j + C_{ij} , i = 1, ..., 10, j = 1, ..., 200$$

with y_{ij} the observation at day i and location j, μ the grand mean, A_i the space-averaged deviation from the mean level at day i, B_j the time-averaged deviation from μ at location j, and C_{ij} the space-time interaction, or the deviation from the station/day expected value when no interaction is present, $\mu + A_i + B_j$. The total sum of squares is divided as demonstrated in Table 1:

	Degrees of freedom	Sum of squares	Mean squares
day	9	16057	1784
station	199	592787	2979
day:station	1791	14531	8

Table 1

Here day:station refers to the interaction term. The percentage variation explained by factor station (R^2 adjusted) is 0.95, by day (time, as factor) is 0.02, and by both is 0.97.

When the ten days are seen as 10 variables, the leading principal component calculated over them captures 98% of the variation. From this, we can conclude that most of the variation is (persistent) spatial variation. The size of the space-time interaction variance component (day:station) is marginal, meaning that to a large extent the temporal pattern is constant (not varying) over space, or, equivalently, that the spatial pattern is to a large extent persistent over time (Gneiting, 2002).

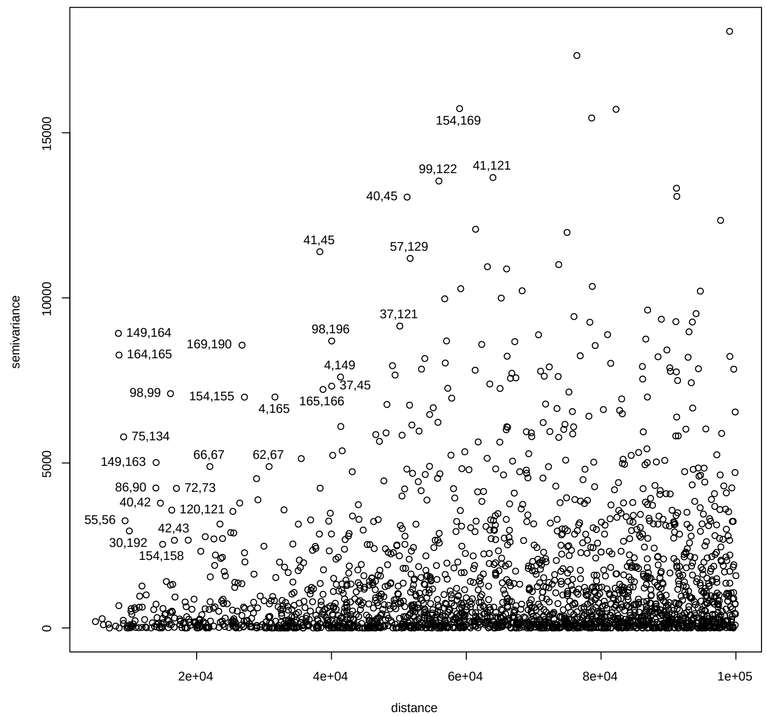

Figure 1
Variogram cloud for day01, with record id (from 1 to 200) pairs denoting which records contributed to manually selected variogram cloud point; record with id 164 is present twice in point pairs with high semivariance at very short distance. Units for this and all other variograms: distance in m, semivariance in $(nSv/h)^2$

From this we can tentatively entertain the hypothesis that for studying spatial patterns, at an early stage it is not of very much importance whether we look at a single day or to leading principal component scores, because the days are very much alike.

Figure 1 shows a variogram cloud for day01, with record identifiers shown for the point pairs at short distance and large differences. For each pair of observations $y(s)$ and $y(s+h)$, the variogram cloud shows half the squared differences in measurements, $0.5(y(s)-y(s+h))^2$ as a function of their separation distance in geographic space, h. The figure suggests that data row 164 (original record number 838) has an unexpected value: it is close to two other observations (149, 165) but differs strongly from them. Figure 2 shows the sample variograms for all data (left) and for the case where record number 838 was removed (right). Removing this record has the effect that the first variogram estimate is no longer strongly influenced by a single, unexpected observation, and we will remove this observation in further analysis.

2.2 VARIOGRAM MODEL SELECTION AND FITTING

For fitting a variogram model, we first have to choose a suitable model. The models we evaluated were exponential, Gaussian, spherical, linear, and the Matern family of variogram models (Chilès and Delfiner, 1999; Stein, 1999) with smoothing parameter values in {0.05, 0.1, 0.2, 0.3, ... , 1.9, 2, 5, 10}.

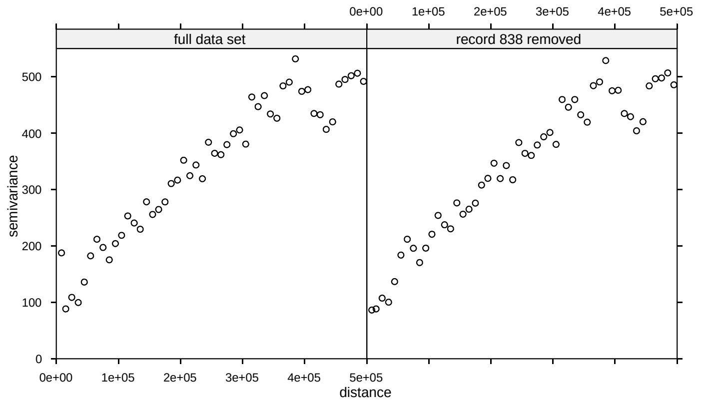

Figure 2
Variograms for day01, for all data (left) and for the case where record 838 (row number 164) was left out; cut-off distance is 500,000 m, lag distance intervals are 10,000 m

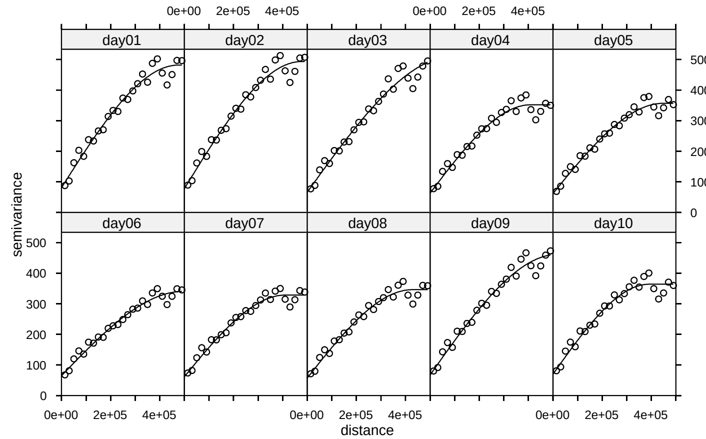

Figure 3
Variograms with fitted spherical models for each of the 10 days

Model selection was done as follows: for each of the ten days, variogram models were fitted to sample variograms (see e.g. figure 3), and used in a leave-one-out cross validation to obtain a cross validation root mean square error (RMSE) by

$$RMSE = \sqrt{\frac{1}{1990} \sum_{i=1}^{10} \sum_{j=1}^{199} (y_{ij} - \hat{y}_{[ij]})2}$$

with $\hat{y}_{[ij]}$ the predicted value using all observations of that day, except the observation y_{ij} itself (i.e., we leave one observation out). Three trend models were considered: a constant trend, a linear and a quadratic trend in the coordinates; when the trends were linear or quadratic the sample variogram was calculated from residuals with respect to a trend estimated by ordinary least squares.

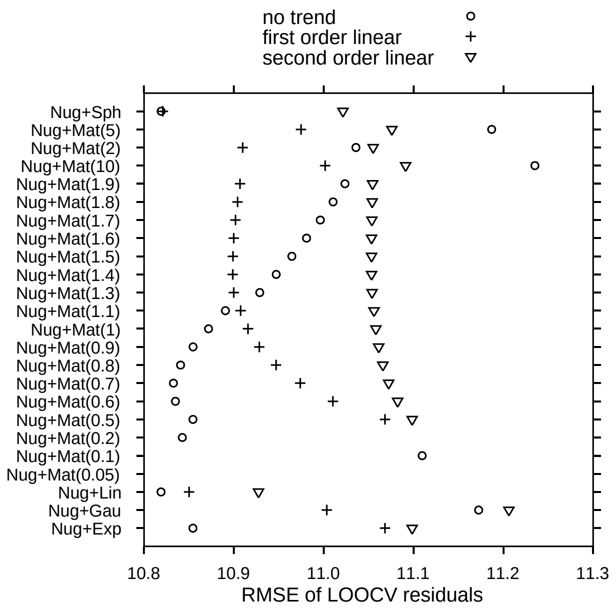

Figure 4
Leave-one-out cross validation (LOOCV) RMSE values ($(nSv/h)^2$) obtained for a series of variogram models and trend functions; each time the variogram was fitted to the data from day01 and used for subsequent days

Figure 4 shows the RMSE values for each of the variogram models fitted, and for each of the trend models. From this figure it can be concluded that (i) the interpolation with a constant mean as trend model performs best, or equally well as higher order trends, and (ii) the linear + nugget and spherical + nugget model perform best. These two models, fitted to the sample variogram of day01 are shown in figure 5. We finally chose for the spherical + nugget model because it is more flexible than the linear model: it effectively includes the linear model as a special case. After selecting a variogram model, we verified for directional dependence in semivariance (anisotropy). Figure 6 shows directional variograms for day01, along with the fitted omnidirectional (isotropic) model. The figure does not give reason to model anisotropy.

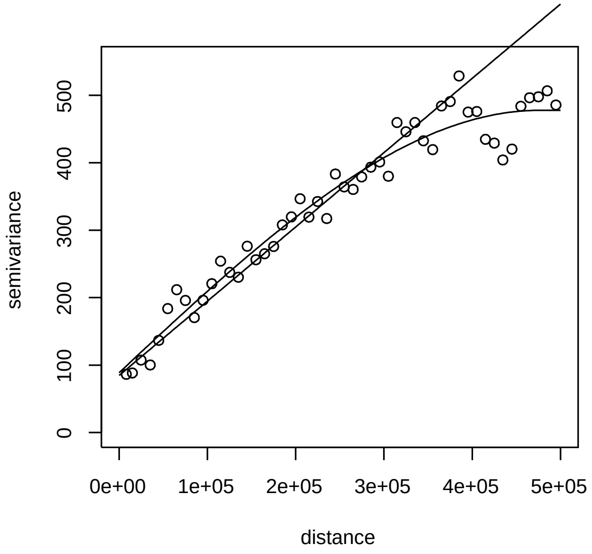

Figure 5
The two best fitting variograms for day01, in terms of cross validation RMSE after removal of record 838: linear (straight line) and spherical; both having a RMSE of 10.82

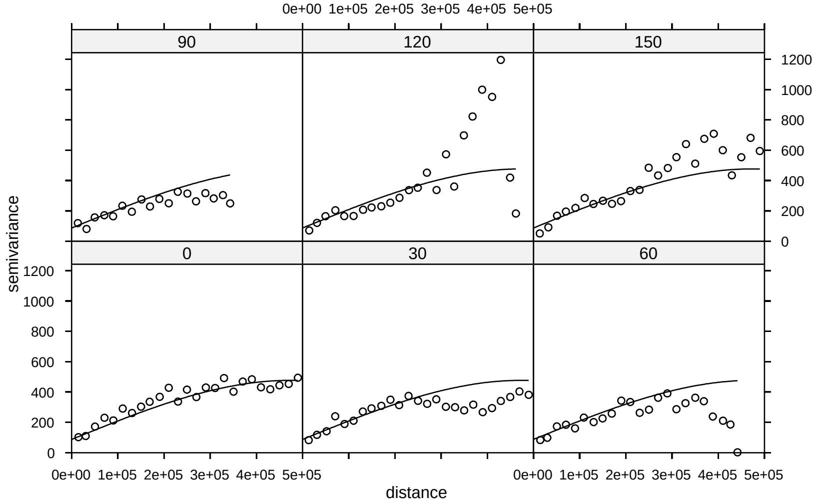

Figure 6
Directional sample variograms (+) for day01: 0 refers to north-south, 90 to east-west; the fitted model (line) is the isotropic spherical variogram model fitted to the omnidirectional variogram for this day

2.3 CROSS CORRELATION AND COKRIGING

Cross correlation between the available days is expected to be strong, because the spatial pattern appeared to be mostly persistent over time. Figure 7 shows direct and cross variograms for each of the 10 days and the 45 possible pairs of days; the fitted model is a spherical model that obeys the linear model of coregionalization (Chilès and Delfiner, 1999). Cross validation using 10-variable cokriging, each time leaving a full location (i.e. data for all available days), out for each multivariable (cokriging) prediction did not lead to improvement: RMSE values raised from 10.8 to 11.05. This result may be surprising, and a possible cause for it is given in the discussion.

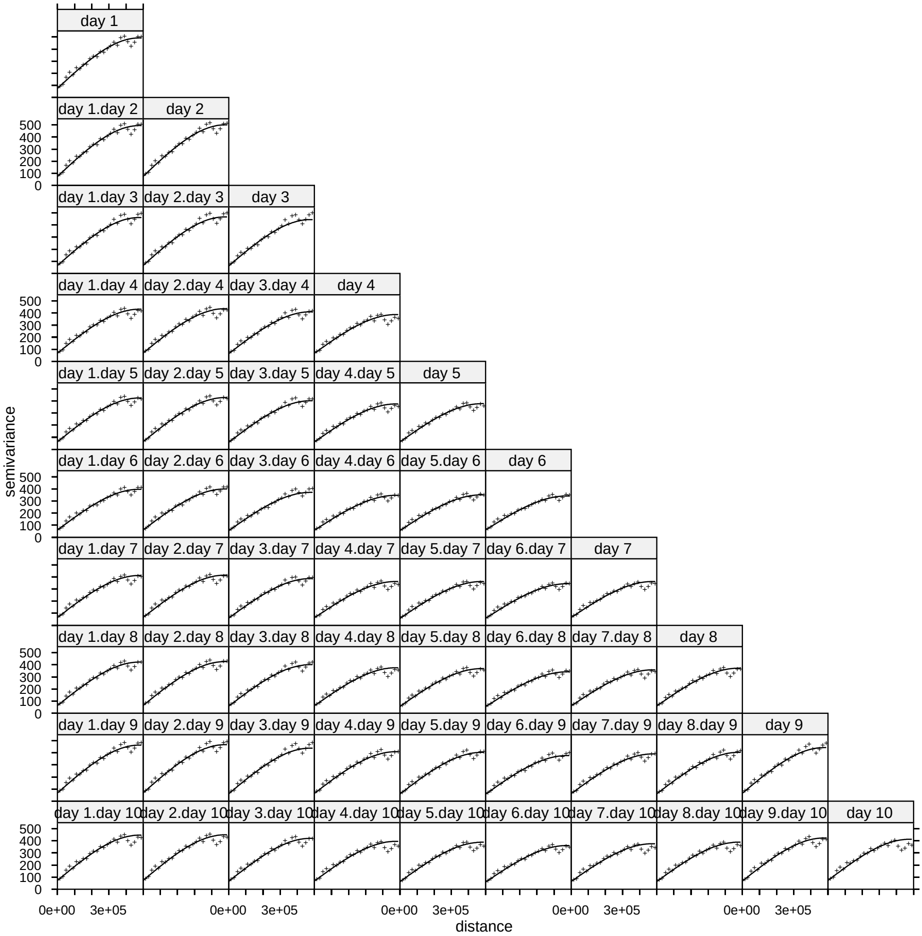

Figure 7
All direct and cross sample variograms (+) for the 10 days of data (direct variograms on the diagonal) along with a fitted linear model of coregionalization (lines); lag distance interval width is 20,000 m, cut-off 500,000 m

2.4 KRIGING NEIGHBOURHOOD SIZE

Up to now, for practical (speed) reasons, local kriging neighbourhoods with size 40 were used, meaning that each kriging interpolation used the nearest 40 observations (for the each variable, when in a cokriging setting). The choice of kriging neighbourhood size may have some influence on the success of a kriging procedure, and we varied it over the values {10, 20, 30, 40, 50, 75, 100, 125, 150, 200}, the last effectively being a global neighbourhood. Figure 8 shows how marginally the leave-one-out cross validation RMSE varies as a function of kriging neighbourhood size. We settled on the “best” value of 125; a fairly large, but in terms of computation load by far not prohibitively large neighbourhood.

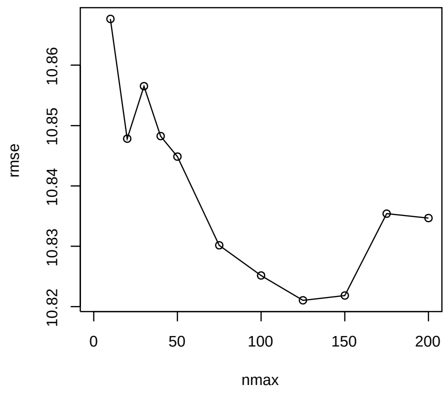

Figure 8
RMSE as a function of kriging neighbourhood size (nmax), leave-one-out cross validation results using the spherical variogram

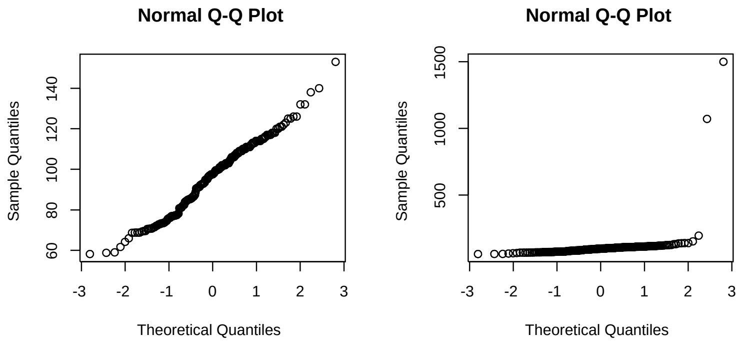

Figure 9
Normal probability plots for the observations of the first test data set, input (left) and the second test data set, joker (right); Theoretical Quantiles are those of the standard normal distribution (-); Sample Quantiles are those of the measurements (nSv/h)

2.5 THE AUTOMATIC MAPPING PROCEDURE

This led to a procedure where we:

1. omit record 838;
2. calculate a omnidirectional sample variogram from the raw data, using a cut-off of 500,000 and an interval (lag) width of 20,000 m, as in figure 3;
3. fitted a spherical variogram model with nugget to this sample variogram, using weighted least squares (Cressie, 1993) fitting with weights $h^{-1}N_h$, with distance h and number of point pairs N_h;
4. use ordinary kriging for prediction, given the observed data and the fitted variogram model, using a local kriging neighbourhood of 125 observations.

The implemented R code is shown in the ANNEX. During the test phase, a number of mistakes were made; these are also pointed out in the ANNEX.

3 THE TEST PHASE

During the test phase, two files were made available for download, input and joker. The sample distributions, shown in the normal probability plots of figure 9 clearly show that the joker data set contains a huge outlier. In normal probability plots normally distributed samples plot as a straight line. For radiation values without an anomaly, the data are close to normal. This plot type can therefore be helpful in an early screening stage to detect the presence of outliers.

Running the (corrected, see ANNEX) script, the following output resulted from the essential part, the two calls to do.sic (ANNEX) R script:

```
> output = do.sic(input, sic.pred)
Loading required package: gstat
Loading required package: lattice
[using ordinary kriging]
> joker.output = do.sic(joker, sic.pred)
Warning: singular model in variogram fit
[using ordinary kriging]
```

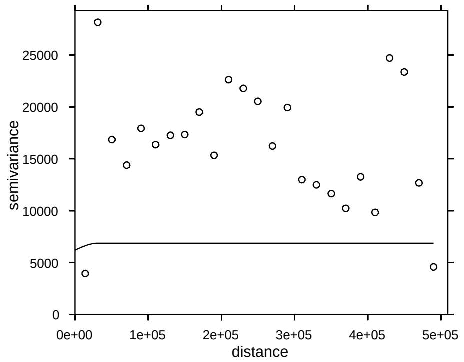

Figure 10
Sample variogram for joker, along with last fitted variogram model before the singularity warning was issued

The warning did not stop the execution, and a new pair of prediction data sets were formed, and submitted as the final prediction.

The cross validation RMSE for this second run was identical for the input data, but was 121.6 for joker. Crucially, the “correct” run issues a warning message that was not present in the original run: Warning: singular model in variogram fit. We will now look into detail what this means.

The variogram model fitted to the sample variogram in the last iteration before the warning message (the model that was returned by the function) was a spherical model with a nugget of 6189, a partial sill of 676 and a range of 30696 (Figure 10).

The first two sample variogram estimates were:

```
> variogram(dayx~x, ~x+y, joker, width=2e4, cutoff=5e5)[1:2,]
np dist gamma dir.hor dir.ver id
1 103 13864.26 3884.342 0 0 var1
2 274 31057.91 27845.673 0 0 var1
```

Clearly, the range at the last iteration is smaller than the distance value of the second sample variogram estimate. This means that at the next variogram fitting iteration, two parameters (say the nugget and the range) had to be fitted from one single sample variogram estimate. Just like fitting a straight line through a single point this leads to a singularity, and the warning is issued. It basically means that in the context of this fitting procedure, a large number of variogram models are now possible, a situation where an automatic fitting routine can make no sensible choice. Given that the correct procedure was used in the first place, the warning message should be taken so serious as to recall the resulting predictions. After having submitted the predictions from the procedure with the error, and after having corrected the error, my final submission for joker was: *my procedure fails on joker*.

3.1 WHAT IF'S

So far, a number of decisions in the modelling process have been fairly ad hoc. The consequences of two of them, the interval lag width and the variogram model type, will be examined in some more detail.

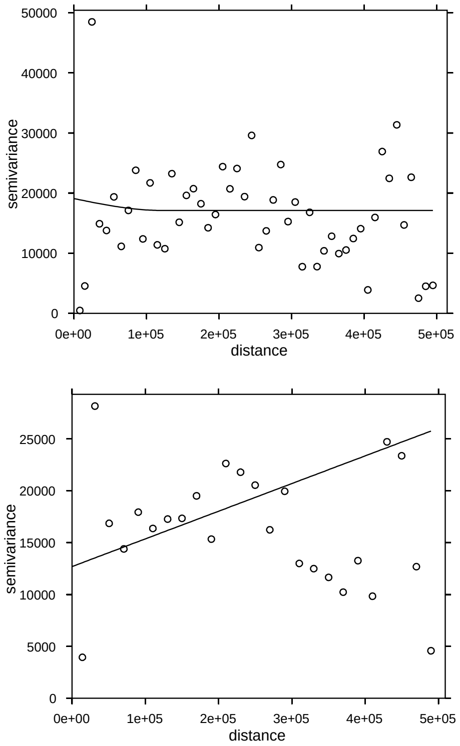

Figure 11
Sample variogram for joker, along with fitted (a) fitted spherical variogram model when a lag interval width of 10,000 is used, instead of 20,000, (b) fitted linear variogram model

What would have happened if instead of a 20,000 interval width the value 10,000 were used? The variogram fitting function does in this case not give a warning, but returns a fitted spherical model with a nugget of 19076, a range of 121543, and a *negative* partial sill of -1967 (Figure 11a), which is of course physically impossible. Using this variogram in the kriging prediction leads, perhaps surprisingly, not to singular systems of kriging equations, but does lead to rubbish predictions: 15% of the predictions are negative, with a minimum value of -469, and the cross validation RMSE is 199. An example of *garbage in, garbage out*, not caught by warning or error messages. The reason for the variogram fitting routine returning negative sill values is that a single routine can be used for fitting direct and cross variograms. Although a minimal lag width does help in revealing outliers or anomalies (as the variogram cloud does), too small lag width values lead to noisy variograms, which may be problematic for variogram model fitting.

When we chose the spherical model, cross validation statistics suggested that the linear model without range would yield equal results (figures 4 and 5). What would have happened if the linear model were used instead of the spherical? The fitted variogram is shown in figure 11b; the cross validation RMSE obtained is 118.8, which is the best we have seen so far. An error checking R function that breaks (stops execution) when a singular variogram fit is encountered is given in ANNEX A.4.

4 RESULTS: THE EVALUATION PHASE

4.1 GLOBAL STATISTICS

For the first data set, only a single run was performed. For the second data set, the following scenarios were compared: (i) "singular fit", which uses the variogram resulting from the singular fit (figure 10), (ii) the "linear fit" of figure 11b. As a standard benchmark, inverse distance weighted interpolation (idw), using distance weighting power 2 and a 125 nearest neighbourhood was added as a scenario for both data sets. Table 2 states the summary statistics for observed and predicted values for the first data set:

	Min	Max	Mean	Median	Std.dev
Observed	57	180	98.02	98.8	20.02
Predicted	69.92	125.14	96.8	99.14	14.47
Idw	70.69	129.58	96.93	99.51	12.03

Table 2

The summary statistics for the observed and predicted values for the second data set (joker), under several scenarios of predicting this secondary data set, are: min max mean median stddev scenario, as in Table 3:

	Min	Max	Mean	Median	Std.dev	Scenario
Observed	57	1528.2	105.4	99	83.7	
predicted	92.2	178.4	117.4	120.7	9.6	Singular fit
Predicted	73.2	255.2	109.4	103.6	31.8	Linear fit
Predicted	71.13	497.63	110.55	106.4	39.9	Inverse distance w.

Table 3

Summary statistics for the errors, here defined as predicted-observed, for the first data set and all scenarios for the second (joker) data set, are in Table 4:

Data set	*MAE*	*ME*	*Corr*	RMSE	Scenario
1	9.11	-1.22	0.788	12.44	
1	9.76	-1.09	0.778	13.10	inverse distance w
2	28.45	12.01	0.380	81.41	singular fit
2	23.26	4.00	0.417	76.19	linear fit
2	21.31	5.13	0.512	72.09	inverse distance w.

Table 4

In Table 4 *mae* refers to mean absolute error, *me* to mean error, *corr* to Pearson's correlation between observed and predicted, and *RMSE* to root mean square error.

It is surprising that the RMSE for the first data set is much larger than those obtained in figure 4. A possible cause is that for each day the variogram (including the nugget variance) changes (figure 3); the first test data set may be a day with larger (nugget) variance than the average over the 10 test days. Leave-one-out cross validation RMSE values using the same procedure for the 10 days range from 10.3 to 11.7.

The RMSE values for the four kriging scenarios for the second (joker) data set are fairly disastrous. Ironically, both in terms of RMSE as in terms of correlation between predicted and observed, inverse distance weighting interpolation slightly outperforms the kriging scenarios for the second data set. This is not the case for the first data set.

4.2 INTERPOLATED MAPS AND KRIGING STANDARD ERROR

Figure 12 shows the interpolated (kriged) maps for both data sets, based on the test data only, and both based on the variogram for the first data set. The kriging standard errors are valid for the first data set: both maps are based on the same variogram model, but only for the first data set is the variogram model the model fitted to the data used. For the second data set, in the area with the anomaly the kriging standard error will grossly underestimate prediction errors. Figure 13 shows a perspective plot (wireframe) of the predicted second data set (joker).

5 DISCUSSION

In this paper a classical geostatistical "algorithm" was designed for automatic mapping background radioactivity levels. The procedure addressed the following model decisions: (i) outlier detection, (ii) choice of an isotropic variogram model, (iii) choice of variogram cut-off distance and lag width, (iv) verification of directional dependence, (v) verification whether cokriging improved predictions, (vi) optimize kriging neighbourhood. Choices were based on visual inspection (i, iii, iv) and leave-one-out cross validation RMSE values (ii, v, vi). The issues were evaluated *in this particular order*; the idea behind it is that prior to the analysis we expected this to be the order of importance. Figure 8 e.g. confirms that the neighbourhood size has a very small influence on the cross validation errors. We do not claim that choosing another order through (i)–(vi) will lead to the same automatic mapping strategy. It is interesting to speculate what would happen if the outlier detection were automated: should the automatic mapping script be modified to filter outliers? Should they be ignored for sample variogram estimation or even for kriging as well? The current study suggests that for the anomaly, from all approaches tried, that based on a variogram fitted to the data without anomaly gives the best predictions, expressed in average errors. Obviously, this approach does not give valuable uncertainty measures (kriging prediction standard errors) for the predictions of the anomaly.

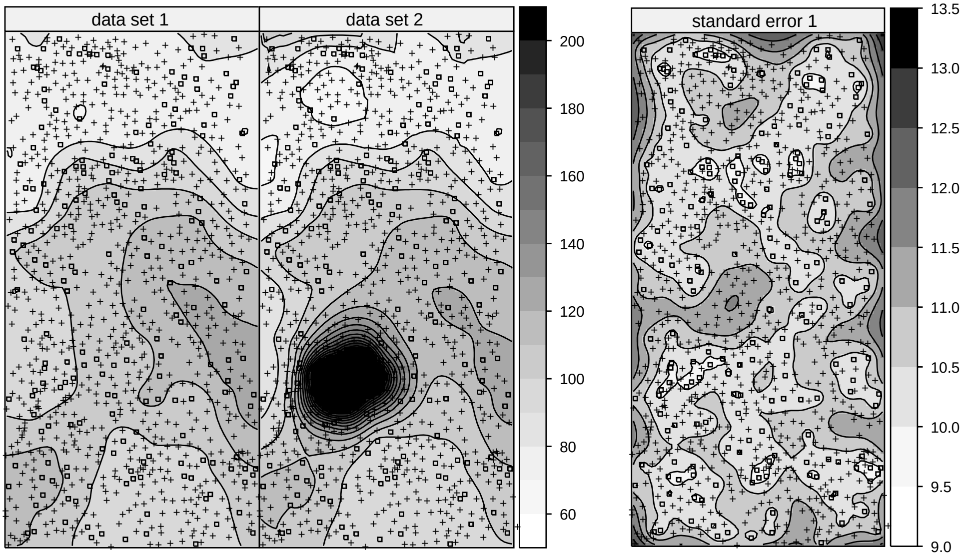

Figure 12
Left: predictions for day 11 and joker; prediction standard errors; perspective plot of joker predictions; Right: prediction standard errors (nSv/h)

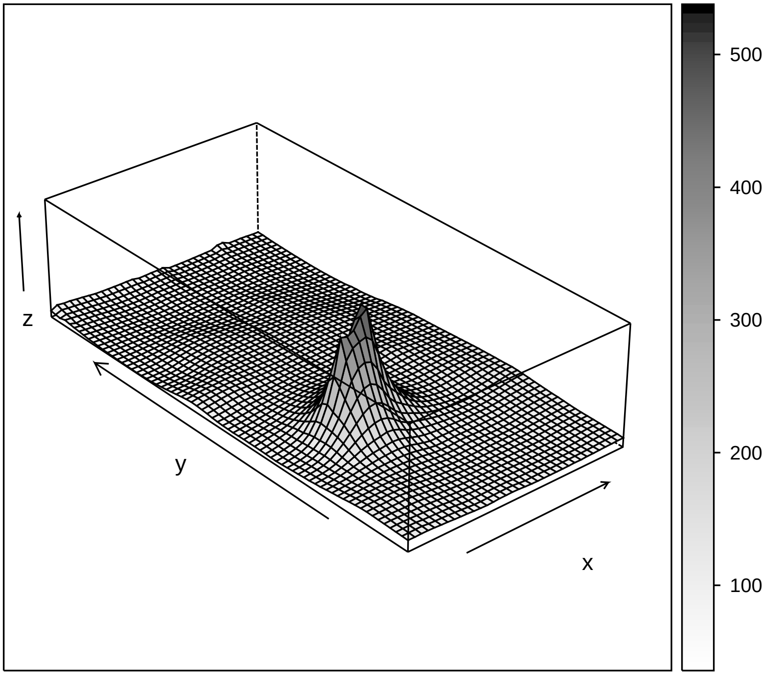

Figure 13
Perspective plot with wire frame for predictions on second (joker) data set, using the test data (nSv/h)

The question remains why cokriging did not lead to improvement in the cross validation setting. A possible cause is isotopy: all ten days have data on a the same set of locations; the variography may be close to the intrinsic correlation model in which case cokriging is equivalent to ordinary kriging (Wackernagel, 1998). Furthermore, fitting a linear model of coregionalization imposes quite strict limits, especially to the variogram range parameter. Isotopic cokriging would improve over kriging only if the secondary variable was much better correlated in space than the primary variable (Goovaerts, 1997).

One issue not addressed in this study is the weighting scheme when using weighted least squares to fit variogram models to sample variograms. We used here weights proportional to $h^{-1}N_h$, giving larger weights to smaller distances (h) and to variogram estimates based on the larger number of point pairs (N_h). Other weighting schemes incorporate the semivariance value, or address correlation among variogram estimates. For extreme sample variograms such as shown in figure 10 and 11, the weighting scheme chosen may play a considerable role.

5.1 AUTOMATION AND ANOMALIES

In principle, we believe that kriging interpolation is a suitable method for automatic interpolation. Automation of the manual steps carried out in this study, notably outlier rejection, variogram model choice and variogram parameter estimation ("model fitting") may not be advisable in cases where anomalies can be expected or should be detected.

The automatic mapping algorithm devised in this study failed tremendously when confronted with an anomaly (joker, or second test set). Although at that moment this came rather as a shock, in hindsight it is no surprise at all: first of all the algorithm had not been subject to strong testing, and second no testing at all had been done with data containing an anomaly. In this light, the current procedure should be seen as a first attempt, containing many aspects that can be improved. Using a scripting language (such as R) is in so far an advantage that errors can be caught afterwards—it is much harder to detect errors if the procedure involved numerous interactive clicks with a mouse. It should be noted that the mistakes found refer to errors in the scripts written for this particular automation project, not to errors in the gstat software (Pebesma, 2004).

In general, training algorithms with data that contain no anomalies in order to obtain an algorithm that can handle an anomaly when it occurs seems to be a silly undertaking. It is more advisable to add anomalies such as the second (joker) data set to the training data in order to develop robust procedures. For example, one could look for (more) robust estimators of variograms, and one could constrain the fitting of a variogram model by fixing the nugget variance. We hold it likely that the second test data contains too little information about the anomaly to reliably predict its spatial extent.

5.2 LESSONS LEARNED AND SOFTWARE DESIGN

In this study, we encountered a singularity during the automatic fitting of a variogram model to the sample variogram. What should the software do when this happens? Currently, gstat issues a warning message; it is shown how scripts can be built to turn this into an error message; an error message stops the script and no (automatic) map will be produced. A singularity as found here is not specific to the gstat software, and any automatic mapping software should catch it, and deal with it.

It should be noted that singularity during variogram model fit is not *always* a problem: in some cases spherical or exponential models converge to an effectively linear model, in which case fitting three parameters is a singular problem but the resulting variogram becomes useful, i.e. it fits the relevant part of the sample variogram.

A lesson learned in this study is that negative partial sills in direct variograms should lead to an error message. This is and now implemented in the gstat R/S-Plus library. Another message is the danger that lies in S scoping rules: objects present in top level environments (or system environments in the search path) are visible as global variables.

5.3 COMPUTING TIME

The average computing time for variogram computation, variogram modelling and kriging prediction of a single variable on 808 locations was around 3 seconds on a 2GHz standard laptop.

CODES

All computations were done in R (Ihaka and Gentleman, 1996; Dalgaard, 1998), using the gstat package for geostatistical computations (Pebesma, 2004); this package also provides the SIC2004 data. To re-run part of the analysis and reproduce figure 12, proceed as follows: (i) download R, (ii) start R, (iii) install package gstat from CRAN, and (iv) type on the R prompt:

```
> demo(sic2004)
```

ANNEX

A.1 R SCRIPT

The complete R script for the automatic mapping; the two calls to do.sic process input and joker. Note that the function contains two uncorrected errors, noted by bold font (see text).

```
# everything after a # sign denotes comments
# read input:
sic.pred = read.csv("sic2004_out.csv", header=f)
names(sic.pred) = c("record", "x", "y")
input = read.csv("sic2004_input.csv", header=f)
names(input) = c("record", "x", "y", "dayx")
joker = read.csv("sic2004_joker.csv", header=f)
names(joker) = c("record", "x", "y", "dayx")
# define the working function:
do.sic = function(sic.input, sic.output) {
  require(gstat)
  # remove the unexpected observation:
  sic.input = sic.input[sic.input$record != 838, ]
  trend = dayx~ x # i.e., the model has a constant mean
  # (error: x should be 1)
  locations = ~x+y # names of x and y coordinates
  width = 2e4 # interval lag width
  cutoff = 5e5 # cutoff
  v.sample = variogram(trend, locations, input, width = width,
  cutoff = cutoff) # (error: input should be sic.input)
  # initial spherical model for fit: nug=100, p.sill=400,range=4.5e5
  initial.model = vgm(psill = 400, mod = "sph", ran = 450000, nug =
100)
  # fit the nugget, sill and range to the sample variogram:
  v.fitted = fit.variogram(v.sample, initial.model, fit.method = 2)
  # the function returns the output from the call to krige():
```

```
  krige(trend, locations, sic.input, sic.output, model = v.fitted,
    nmax = 125)
}
# do the spatial interpolations:
output = do.sic(input, sic.pred)
joker.output = do.sic(joker, sic.pred)
# write output to .csv
write.table(output, "sic2004_output.csv",
sep=",", col.names=FALSE, row.names=FALSE)
write.table(joker.output, "sic2004_joker_output.csv",
sep=",", col.names=FALSE, row.names=FALSE)
```

A.2 INITIAL SUBMISSION

During the test phase, two files were made available for download, input and joker. To test the function do.sic(), we ran the available data a few times, renaming them to the input and joker. The results were checked by cross validating them, using a call to krige.cv instead of krige in the function body, and then calculating the RMSE. RMSE values found were very close to the smallest values found in figure 8 and 4, and from this it was concluded that the procedure worked.

Next, the files were downloaded, run through the function designed, and the results were uploaded (or rather emailed, because the upload function was not functioning yet). Execution time was about 7 seconds, and no human intervention took place apart from the download – run – return cycle.

A.3 A MISTAKE

After that, one could argue that the human intervention part started, and by that the evaluation phase started; in some sense the test phase has ended. We will however not be looking at the real evaluation phase before the observations at the 808 locations have become available. Looking at the leave-one-out cross validation RMSE, the input data set gave a value of 10.74, the joker set of 120, more then 10 times as high. Clearly, joker contains an anomaly, as can be seen from figure 9.

Looking at the summaries of predicted values and predicted variances, it turned out that the prediction variances were equal for both sets. This could only be caused by the fact that both were based on exactly the same fitted variogram model (because they have the same topology), and led to the discovery of a bug in the script given the ANNEX: in the function used, input was passed to the variogram() function, instead of sic.input. The data set input is globally visible because it was assigned before the function call.

After correcting this (changing input into sic.input in the call to variogram), the singular variogram fit was obtained.

A.4 ERROR CHECKING

A do.sic function definition that would have given the results wanted, so far has the two error checks, for singularity during fit and for negative partial sills. In case of error it prints a helpful error message, indicating which problem occured: singularity or negative partial sill, it shows the variogram fitted by then, and then, most importantly, it stops execution.

```
do.sic = function(sic.input, sic.output) {
  require(gstat)
  sic.input = sic.input[sic.input$record != 838, ]
  v.sample = variogram(dayx~1, ~x+y, sic.input, width = 2e4, cutoff
=
      5e5)
```

```
  initial.model = vgm(psill = 400, mod = "Sph", range = 450000, nug
=
      100)
   v.fitted = fit.variogram(v.sample, initial.model, fit.method = 2)
   if (attr(v.fitted, "singular") || any(v.fitted$psill < 0)) {
      print(plot(v.sample, v.fitted))
      stop(paste("could not fit sensible model:",
ifelse(attr(v.fitted,
       "singular"), "singular variogram fit", "negative sills")))
   } else
      krige(dayx~1, ~x+y, sic.input, sic.output, model = v.fitted,
         nmax = 125)
}
```

REFERENCES

Chilès, J-P. and Delfiner, P. Geostatistics, modeling spatial uncertainty. Wiley, New York; 1999

Dalgaard, P. Introductory statistics with R. Springer, Berlin; 2002.

Dubois, G. Spatial Interpolation Comparison 97. Introduction and description of the dataset. In: *"Mapping Radioactivity in the environment. Spatial Interpolation Comparison. 1997"*. Dubois, G., Malczewski, J., De Cort, M. (Eds), pp. 39-44. EUR 20667 EN, Office for Official Publications of the European Communities, Luxembourg; 2003.

Dubois, G., Galmarini, S. *"Introduction to the Spatial Interpolation Comparison (SIC)2004 exercise and presentation of the data sets"*. *Applied GIS*; 1 (2): 09-01 to 09-11.

Gneiting, T. Nonseparable, stationary covariance functions for space-time data. Journal of the American Statistical Association, 97, 590-600; 2002.

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

Ihaka, R., Gentleman, R. R: A Language for Data Analysis and Graphics. Journal of Computational and Graphical Statistics 5(3) 299–314; 1996.

Pebesma, E.J. Multivariable geostatistics in S: the gstat package. Computers & Geo-sciences, 30: 683-691; 2004.

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

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

Wackernagel, H. Multivariate Geostatistics, An Introduction with Applications. Second edition. Springer-Verlag, Berlin; 1998.

Cite this article as: Pebesma, Edzer J. 'Mapping radioactivity from monitoring data: automating the classical geostatistical approach'. *Applied GIS*, Vol 1, No 2, 2005. pp. 11-01 to 11-17. DOI: 10.2104/ag050011