Geostatistical Methods in R

Adéla Volfová, Martin Šmejkal
Students of Geoinformatics Programme
Faculty of Civil Engineering
Czech Technical University in Prague

Abstract

Geostatistics is a scientific field which provides methods for processing spatial data. In our project, geostatistics is used as a tool for describing spatial continuity and making predictions of some natural phenomena. An open source statistical project called R is used for all calculations. Listeners will be provided with a brief introduction to R and its geostatistical packages and basic principles of kriging and cokriging methods. Heavy mathematical background is omitted due to its complexity. In the second part of the presentation, several examples are shown of how to make a prediction in the whole area of interest where observations were made in just a few points. Results of these methods are compared.

Keywords: geostatistics, R, kriging, cokriging, spatial prediction, spatial data analysis

1. Introduction

Spatial data, also known as geospatial data or shortly geodata, carry information of a natural phenomenon including a location. This location allows us to georeference the described phenomenon to a region on the Earth. It is usually specified by coordinates such as longitude and latitude. By mapping spatial data, a data model is created. We will focus on a raster data model which provides value of the phenomena at each pixel of the area of interest. Mapping is a very common process in sciences such as geology, biology, and ecology. Geostatistics is a set of tools for predicting values in unsampled locations knowing spatial correlation between neighboring observations.

Making use of geostatistics requires difficult matrix computations briefly described in chapters Ordinary Kriging and Multivariate Geostatistics. In order to make our predictions easier, we are going to use methods from R geostatistical packages introduced in chapter R Basics. The best known geostatistical prediction methods are called kriging (for univariate data set) and cokriging (for multivariate data set) — examples of their use are shown in chapter Example of Kriging and Cokriging in R.

2. R Basics

There any many applications implementing geostatistical methods. Most of them are complex GIS and most of them are commercial. This does not hold for project R. R is a language and environment for any statistical computations and creating graphics. R is available as Free Software under the terms of the Free Software Foundation's GNU General Public License. R is multi–platform, easy to learn, and with a huge amount of additional packages that extend its functionality. For instance, such packages serve for special branches of statistics such as geostatistics.

2.1. First Steps in R

After downloading and installing R, open the default R console editor, figure 1. A standard prompt `>` appears at the last line. When this prompt is shown, R is ready to accept a command. Another prompt exists (`+`) which is for multiple row commands. There are two symbols for assigning a value to a variable: `<-` and `=`. When working in the *R Console*, the command is executed right after hitting *Enter*. If the user wishes to create a set of commands that are saved and can be used later, it is necessary to create a script (*File — New script*). For executing commands from a script, select all commands to be executed and press *Ctrl+R*. The results are shown in the *R Console*. The `#` symbol is used for comments. The following brief list of examples is the basic overview of commands to get you started in R. Please refer to countless internet tutorials for more advanced examples.

```
# Help and packages
help.start()                # Load online HTML help
help(function)              # Show online help for "function"
?function                   # Show online help for "function"
help.search("keyword")      # Open RGui dialog for packages/functions
                            # ... or classes connected to "keyword"
q()                         # Quit R
library()                   # List downloaded libraries
library(package)            # Load "package"
install.packages(package)   # Download "package"

# Assigning values
a <- 3
b = 8
x <- c(5, 2, 7)             # Vector
y <- 2*x^2                  # Evaluate elements of vector x 1 by 1,
                            # ... dimension of x and y matches
y                           # Print content of a variable
    [1] 50  8 98            # ... result
1:5                         # Create sequence
    [1] 1 2 3 4 5           # ... result
seq(1,5)                    # Another way to create sequence
    [1] 1 2 3 4 5           # ... result
x[1]                        # Print 1st element of x
x[1:4]                      # Print 1st through 4th element of x
    5  2  7 NA              # ... result (when out of range,
                            # ... NA value is printed)
x[length(x)]                # Print last element of vector

# Input table from a file
# --- file data.txt ---
Id  Price   Brand
1   3.5     Goldfish
```

```
2    7.0      Wolf
3    1.5      Seal
4    3.0      Goldfish
# ----------------------

# Read table, a relative path can be used
# \ needs to be doubled
T = read.table("C:\\Data\\data.txt", header=T)

T                          # Print table T
T["Id"]                    # Print column Id
T[2]                       # Print 2nd column as data.frame
T[[2]]                     # Print 2nd column as vector
T[2,]                      # Print 2nd row
T[3,"Brand"]               # Print 3rd row in Brand column
T$Price                    # Print Price column into a row vector
colnames(T)                # Print column names
colnames(T) <- c(1,2,"x") # Rename columns

# Plotting
x <- 1:5; y <- x^2
plot(x,y)                  # Plot data
points(x,y,pch="+")        # Add new data (use + symbols)
lines(x,y)                 # Add a solid line
text(10,12,"Some text")    # Write text on plot
abline(h=7)                # Add a horizontal line
abline(v=6,col="red")      # Add a vertical line
```

All necessary manuals and package documentation are stored in the Comprehensive R Archive Network (CRAN) [5].

2.2. Geostatistical Packages in R

A surprisingly large number of packages implementing geostatistical principles have been released. We are going to use only a few of them, the most common ones – `geoR`, `gstat`, and `sp`. However, for those who wish to explore more packages, a list of some spatial packages with a brief description is provided below. These packages were developed by different communities, therefore their functionality overlaps sometimes. Some of them are out of date and are not recommended for use anymore. Please refer to online documentation for more information on each package and its methods description.

`geoR` — is probably the most important package for geostatistical analysis and prediction.

`geoRglm` — extends functionality of `geoR` package. It is designed for dealing with generalized linear spatial models.

`gstat` — provides users with vast number of methods for both univariate and multivariate geostatistics, variogram modeling, and very useful plotting functions.

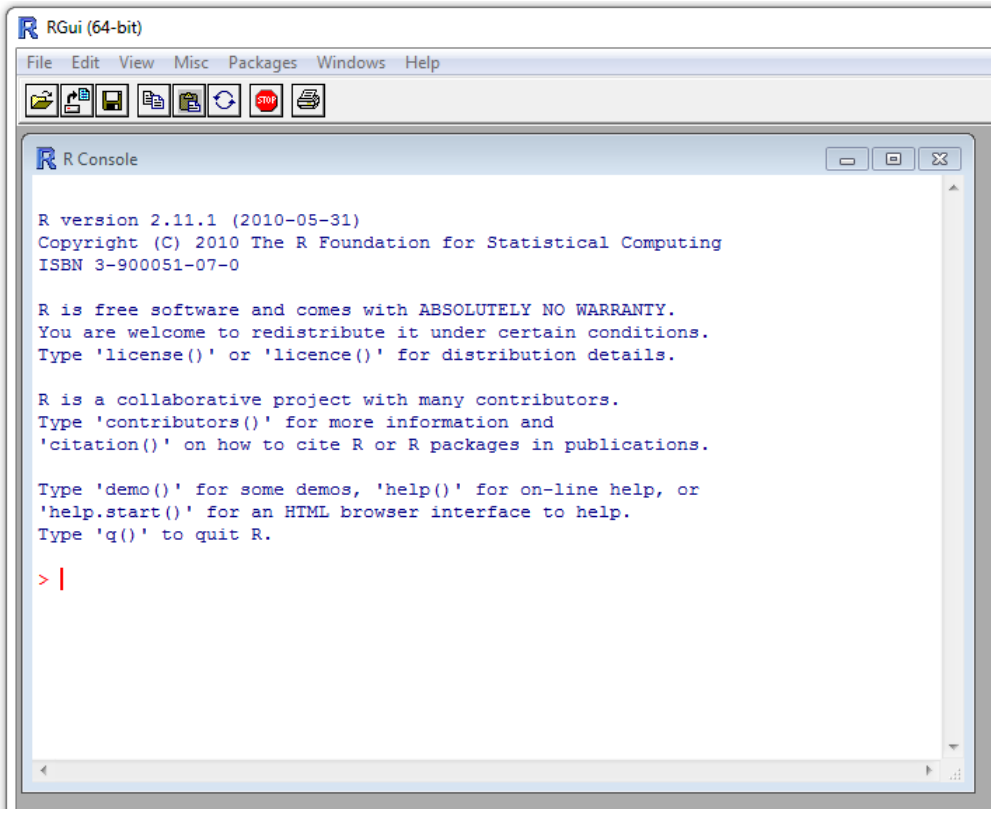

Figure 1: Default R interface.

`sp` — is a package for various work with spatial data — plotting, spatial selection, summary etc.. It also provides a very good training data set called `meuse`.

`intamap` — provides classes and methods for automated spatial interpolation.

`fields` — is a package with functionality similar to the `gstat` package. It is useful for curve, surface and function fitting, manipulating spatial data and spatial statistics. A covariance function implemented in R with the `fields` interface can be used for spatial prediction. This package also includes methods for visualization of spatial data.

`RandomFields` — provides methods for simulation and analysis of random spatial data sets. It also provides prediction methods such as kriging.

`vardiag` — allows to diagnose variogram interactivelly.

`sgeostat` — is an object–oriented framework for geostatistical modeling in S+[1].

`spatial` — contains methods for kriging and point pattern analysis.

`spatstat` — is another very extensive package for analysis of spatial data. Both 2D and 3D data sets can be processed. It contains over 1000 functions for plotting spatial data, exploratory data analysis, model–fitting, simulation, spatial sampling, model diagnostics, and

[1]S+ is a language for data analysis and statistics. It is possible to use the `sgeostat` package in R as well.

formal inference. Data types include point patterns, line segment patterns, spatial windows, pixel images, and tessellations.

There are many other packages dealing with spatial data in some way. A description of all of them is beyond the scope of this paper. For more information please refer to [4].

3. Spatial Statistics Basics

Spatial statistics is a set of statistical tools where location of data is considered. The main goal of geostatistics is to make a prediction of data x_i $(i = 1, 2, \ldots, n)$ within an area of interest A where sample observations Z_i have been made. Each observation Z_i is dependent on values of a stochastic process $S(x)$ of spatial continuity in corresponding points x_i.

Functions in the `geoR` package are based on a Gaussian model. According to [8], chapter 3:

Gaussian stochastic processes are widely used in practice as models for geostatistical data. These models rarely have any physical justification. Rather, they are used as convenient empirical models which can capture a wide range of spatial behaviour according to the specification of their correlation structure.

Please, refer this book for more information on Gaussian processes.

3.1. Univariate Geostatistics

There are several statistics of spatial data that serve for general overview of the data set, show potential outliers among the observations, and describe the distribution. These features are shown in examples in section Analysis of Univariate Data.

Now, let's focus on the best known geostatistical method for prediction — kriging. There are many kinds of kriging. Each type determines a linear constraint on weights implied by the unbiasedness condition[2]. We are going to focus on ordinary kriging that assumes a constant but unknown mean.

In order to predict the phenomenon in the unsampled locations, we need to specify the spatial dependence. A geostatistical tool describing this dependence is called a variogram (figure 2). From now on, we assume isotropy in our data, then the variogram is so–called omnidirectional variogram. It is defined as the variance of the difference between field values at two locations across realizations of the field [6]. When shown as a plot, the x–axis represents the distance h between two observations. The maximum size h should be set such that we can expect two observations in this distance independent. The variance is depicted on the y–axis and is defined as:

$$\gamma(h) = \frac{1}{2n} \sum_{i=1}^{n} [Z(x_i) - Z(x_i + h)]^2,$$

where $Z(x_i)$ is an observed value of a random field and h is a distance between two observations. If the data are anisotropic, h becomes a vector and we need more variograms, each for a different angle. A variogram determined directly from the measurement is called empirical. For a prediction, we need to create a theoretical variogram that fits the empirical one as good as possible. The necessity of having a theoretical variogram lies in its continuity, so we can

[2] http://en.wikipedia.org/wiki/Kriging

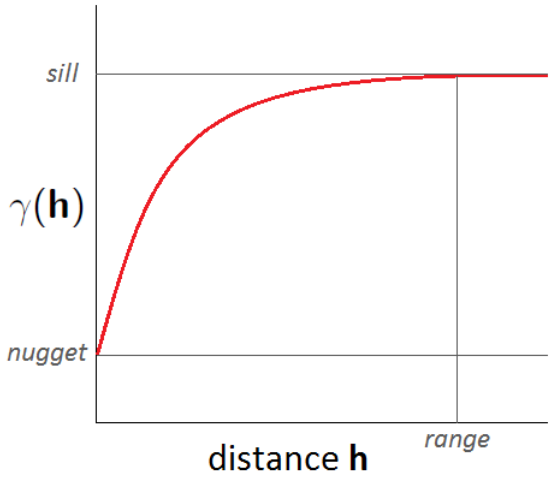

Figure 2: Variogram.

obtain the variance for any distance h. The kriging matrices based on such variogram must be positive definite.

There are three important characteristics of a variogram:

- *range* — a value of a variogram increases with increasing distance h up to a certain distance. Further than this, the variogram does not change much and we expect two observation independent behind this range,
- *sill* — the upper value of a variogram,
- *nugget* — the value of a variogram for zero h is strictly zero, nevertheless for the shortest distance h the variogram is computed, its value jumps from zero to a certain value (a nugget). This is called a nugget effect and it is caused, for instance, by an error of a measurement.

3.2. Ordinary Kriging

Since we have a model of spatial dependence (i.e. we know the formula of our theoretical variogram), we can predict the phenomenon in an unsampled location. Let us call this location x_0, then

$$Z^*(x_0) = \sum_{\alpha=1}^{n} \lambda_\alpha Z(x_\alpha),$$

where λ_α is a weight for value $Z(x_\alpha)$ at x_α.

Ordinary kriging is aliased BLUP (best linear unbiased predictor) and therefore the following conditions hold:

- a sum of weights is equal to 1 (guarantees the unbiasedness of the prediction),
- a variance of estimation errors is minimal.

There is one thing left to determine for the prediction — the vector of weights.

$$\begin{bmatrix} \lambda_1 \\ \vdots \\ \lambda_n \\ \mu \end{bmatrix} = \begin{bmatrix} C_{11} & \cdots & C_{1n} & 1 \\ \vdots & \ddots & \vdots & \vdots \\ C_{n1} & \cdots & C_{nn} & 1 \\ 1 & \cdots & 1 & 0 \end{bmatrix}^{-1} \begin{bmatrix} C_{10} \\ \vdots \\ C_{n0} \\ 1 \end{bmatrix},$$

where μ is a Lagrange parameter and C_{ij} is a covariance between $Z(x_i)$ and $Z(x_j)$. A relationship between a covariance and a variogram is following:

$$C_{ij} = Cov(Z(x_i), Z(x_j)) = C(0) - \gamma(x_i - x_j),$$

where $C(0)$ is the *sill* of the variogram model.

More detailed mathematical description is out of the scope of this paper. For more, please refer to [1,2,7].

3.3. Multivariate Geostatistics

Natural phenomena from one region can show some measure of dependency between each other. In such case, we can take one variable for prediction (primary) and the other variable(s) (secondary) to enhance the prediction. This is applied in cases where obtaining data of the primary variable is expensive, technically very difficult, or for any other reason we have an insufficient number of obtained data. In that case, we can look for some dependent variables in the region which we can measure in a much easier or cheaper way. Beside other advantages, we can reveal extreme values of the primary variable at locations where its measurement have not even been made.

3.4. Covariables Dependency

We assume to have only one secondary variable from now on. In case we have the covariables measured exactly at the same locations, we can easily tell the strength of their dependency by computing a correlation coefficient and/or by plotting a *scatterplot*. A *scatterplot* is a figure with axes corresponding to values of variables, one axis for each variable. In case we do not have measurement at the same locations, the best way to reveal the dependency is to compare variograms of the variables. We use so–called coregionalization when a cross–variogram is created [1].

3.5. Cross-variogram

A cross–variogram describes correlation between covariables and is given by:

$$\gamma_{12}(h) = \frac{1}{2} E[(Z_1(x+h) - Z_1(x))(Z_2(x+h) - Z_2(x))],$$

where Z_1 and Z_2 are primary and secondary variables. In some cases (e.g. in R methods), a pseudo cross–variogram is computed. There are inconsistent opinions on its use [1] (p. 150), however, its advantage is to gain much more points for an empirical cross–variogram. The pseudo cross–variogram is given by:

$$\psi_{12}(h) = \frac{1}{2} E[(Z_1(x+h) - Z_2(x))^2].$$

3.6. Ordinary Cokriging

Since we have the variogram and cross–variogram models, we can use ordinary cokriging for prediction. A value of a primary variable in an unsampled location is given by the following equation.

$$Z^*(x_0) = \sum_{S_1} \lambda_{1\alpha} Z_1(x_\alpha) + \sum_{S_2} \lambda_{2\alpha} Z_2(x_\alpha),$$

where S_1 and S_2 are sets of samples for the primary and secondary variables respectively.

The following hold:

- the sum of weights $\lambda_{1\alpha}$ is equal to 1 and the sum of weights $\lambda_{2\alpha}$ is equal to 0 (guarantees the unbiasedness of the prediction),
- a variance of estimation errors is minimal.

A relationship between a cross–variogram and a cross–covariance is:

$$\gamma_{12}(h) = C_{12}(0) - \frac{C_{12}(h) + C_{21}(h)}{2},$$

Then, ordinary cokriging system in matrix form is given as:

$$\begin{bmatrix} C_{11} & C_{12} & 1 & 0 \\ C_{21} & C_{22} & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} \lambda_1 \\ \lambda_2 \\ \mu_1 \\ \mu_2 \end{bmatrix} = \begin{bmatrix} C_{01} \\ C_{02} \\ 1 \\ 0 \end{bmatrix},$$

where C_{11} and C_{22} are covariance matrices of primary and secondary variables respectively, and C_{12} is a cross–covariance matrix.

For more detailed mathematical explanation of ordinary cokriging including proves, please follow [1,2].

3.7. Sampling Density and Location of Primary and Secondary Variables

There are several cases of how the covariables can be measured:

- samples of both, the primary and secondary variable, are obtained at exactly identical locations — this case is not very often because we either have a sufficient data set for the primary variable and so the secondary is not necessary to include in a prediction, or we do not have enough samples of the primary variable for creating a valid prediction by ordinary kriging and the secondary variable will not provide us with more useful information about it,
- the secondary variable is measured with higher density and all the primary variable samples overlap in location with the secondary variable — this is one of the most common cases of use of ordinary cokriging that leads to the best results; the primary variable measurement is not dense enough to make a good ordinary kriging prediction, so a significantly dependent secondary variable substantially increases the density of sampled locations and enhances the prediction of the primary variable,

– the secondary variable is measured with higher density and the covariables sample locations do not overlap — another very common case, however negatively effected by worse determination of the model of coregionalization which leads to not–so–good improvement of the prediction (compared to the previous case),

– number of samples of the secondary variable is smaller then number of samples of the primary variable — this case is useless for getting better prediction,

– the secondary variable samples correspond with all locations for prediction of the primary variable (very dense sampling) — for such a case another type of kriging is recommended — *kriging with external drift* [7]; the more dense the samples of the secondary variable, the harder the prediction to process when using ordinary cokriging.

See figure 3 for examples.

4. Spatial Statistics in R

For purposes of this chapter, sample data set from [2] has been used. These data are derived from a digital elevation model (DEM) of Walker Lake area (Nevada, USA) and are available in the `gstat` package, hence anyone can obtain the same data set a get to the same results.

4.1. Sample Data Set

The Walker Lake DEM has been modified for the sake of generality. There are 1.95 million points in the original data set. These points were divided into blocks of 5 by 5 points and final values were derived from them. There are two variables which we are going to use:

– U variance of the 25 values given by equation

$$U = \sigma^2 = \frac{1}{25}\sum_{i=1}^{25}(x_i - \bar{x})^2,$$

where $x_1, x_2, \ldots, x_{25}$ is elevation in meters,[3]

– V is function of mean and variance

$$V = [\bar{x} * \log(U + 1)]/10,$$

There are 78 000 values in a grid of 260 by 300 points. 470 points were chosen across the area to represent measurement.

4.2. Analysis of Univariate Data

A sample of 100 values in regular grid of 10 by 10 points is used for a following basic data description.

We can obtain a lot of useful information when we arrange the data according to some order, plot them or make some summary statistics. Beside other things, this is good for searching for outliers and errors in the measured data set.

[3]It is obvious that a flat terrain has low value of U, whereas hilly terrain has this variable very high. That is why this variable is also called as *topographic roughness index*.

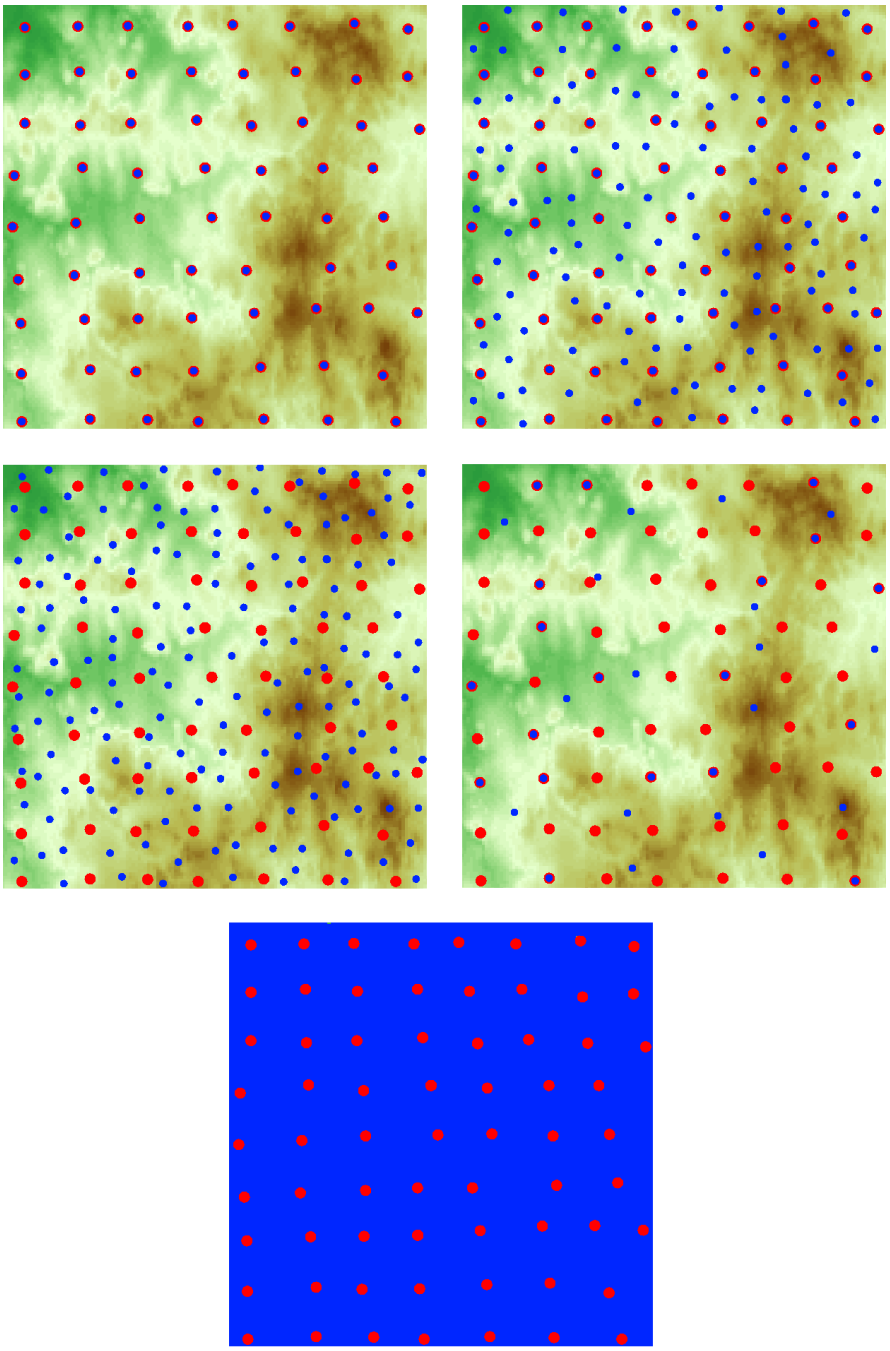

Figure 3: Location of primary and secondary variables. Order of the examples corresponds with chapter Sampling Density and Location of Primary and Secondary Variables, red dots stand for primary variable samples, blue represents secondary variable samples.

The most common presentation of the data is a frequency table and a histogram. The frequency table arranges the data into intervals and shows how many observations fall into each interval. An example of a frequency table for V data set is shown in table 1. See the datasets in the following listing:

$$U =$$

15	12	24	27	30	0	2	18	18	18
16	7	34	36	29	7	4	18	18	20
16	9	22	24	25	10	7	19	19	22
21	8	27	27	32	4	10	15	17	19
21	18	20	27	29	19	7	16	19	22
15	16	16	23	24	25	7	15	21	20
14	15	15	16	17	18	14	6	28	25
14	15	15	15	16	17	13	2	40	38
16	17	11	29	37	55	11	3	34	35
22	28	4	32	38	20	0	14	31	34

$$V =$$

81	77	103	112	123	19	40	111	114	120
82	61	110	121	119	77	52	111	117	124
82	74	97	105	112	91	73	115	118	129
88	70	103	111	122	64	84	105	113	123
89	88	94	110	116	108	73	107	118	127
77	82	86	101	109	113	79	102	120	121
74	80	85	90	97	101	96	72	128	130
75	80	83	87	94	99	95	48	139	145
77	84	74	108	121	143	91	52	136	144
82	100	47	111	124	109	0	98	134	144

A histogram o V values is shown in figure 4. An R function `hist` serves for plotting histogram.

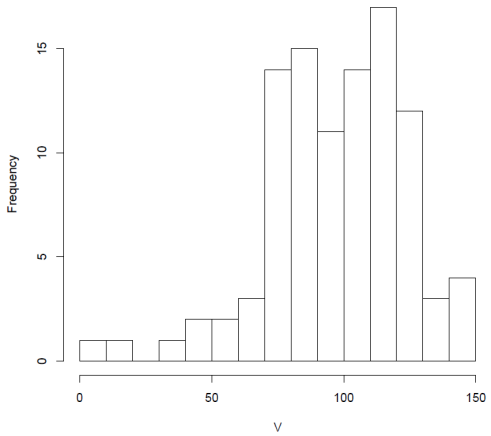

Figure 4: Histogram of V, `hist(V)`.

Some methods used later in this paper, works better for normally distributed data. We can tell whether the data are normally distributed from a plot with the measurement on the x–axis and cumulative frequency on the y–axis. In case of normal distribution, the points are arranged into a line. Example of this plot is in figure 5, it was created in R by calling `qqnorm`

Interval of V	Count
$(0, 10\rangle$	1
$(10, 20\rangle$	1
$(20, 30\rangle$	0
$(30, 40\rangle$	1
$(40, 50\rangle$	2
$(50, 60\rangle$	2
$(60, 70\rangle$	3
$(70, 80\rangle$	14
$(80, 90\rangle$	15
$(90, 100\rangle$	11
$(100, 110\rangle$	14
$(110, 120\rangle$	17
$(120, 130\rangle$	12
$(130, 140\rangle$	3
$(140, \infty\rangle$	4

Table 1: Frequency table for V.

function. Outliers and erroneous data, if some, can be observed in this plot.

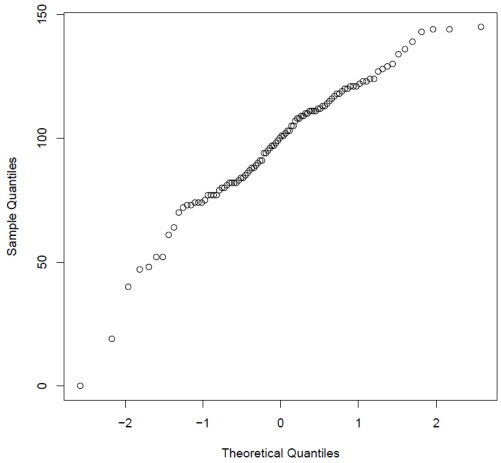

Figure 5: Visual test for normal distribution of V, `qqnorm(V)`.

Another plot good for visual exploration of the data is so–called box–and–whisker plot (figure 6). One half of the data lies inside the box. The line inside the box is median. The whisker lines represent a multiple of border values of the box (in this case a default 1.5 multiple was maintained). One can tell that values of $V(right)$ have larger variance in this case. The circles represent outliers. An R function `boxplot` has been used to create this plot.

For further data description, we look at the summary statistics such as the minimal and maximal value, mean, median, mode, quantiles, standard deviation, variance, interquartile range, coefficient of skewness, coefficient of variation etc.. First five mentioned statistics tell us about the location of important parts of the distribution. Next three values signify the variability of the distribution. The coefficient of skewness and coefficient of variation describe

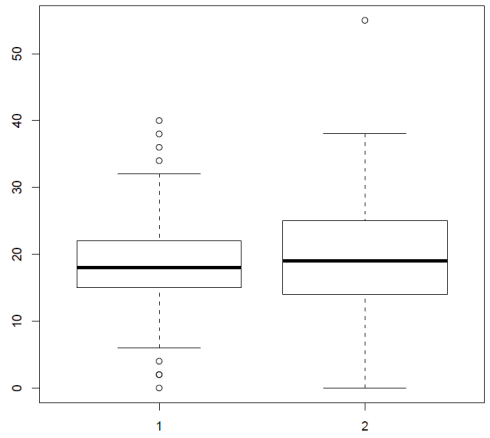

Figure 6: Box–and–whisker plot for U (left) and V (right), `boxplot(matrix(U,V))`.

the shape of the distribution and help to reveal potential erroneous observations.

We can obtain the basic statistics using the `summary` function. Summary statistics for V is listed in the following example.

```
> summary(V)
Min.    :  0.00
1st Qu.: 81.75
Median :100.50
Mean    : 97.50
3rd Qu.:116.25
Max.    :145.00

# variance
> var = sum((V-97.5)^2)/length(V)
689.69

# interquartile range
> IQR = 116.25-81.75
34.5

# coefficient of skewness
> CS = sum((V-97.5)^3)/sqrt(var)^3/length(V)
-0.771

# coefficient of variation
> CV = sqrt(var)/97.5
0.269
```

The coefficient of skewness is, in this case, negative which means the distribution rises slowly from the left and the median is greater then the mean. The closer the coefficient of skewness to zero, the more symmetrical the distribution. Hence the difference between median and

mean is getting smaller.

The coefficient of variation is quite low. If this value is greater then 1, a search of erroneous observations is recommended.

4.3. Variogram

In order to plot an empirical variogram, we need to set a proper distance for the lag (x–axis on the plot). When the lag is too small, the variogram would go up and down despite its theoretical increasing trend before the range distance and constant trend for distance larger than the range. When we set the lag too large, we gain just a small number of values (breaks) on the variogram curve and we would not see the important characteristics of the variogram such as range, sill etc..

In our example we set the lag for 10 m. A variogram cloud (all pairs of points) and an empirical variogram with given lag for the V variable is in figure 7. These variograms were created by `variog` function. The theoretical variogram is modeled with `lines.variogram`

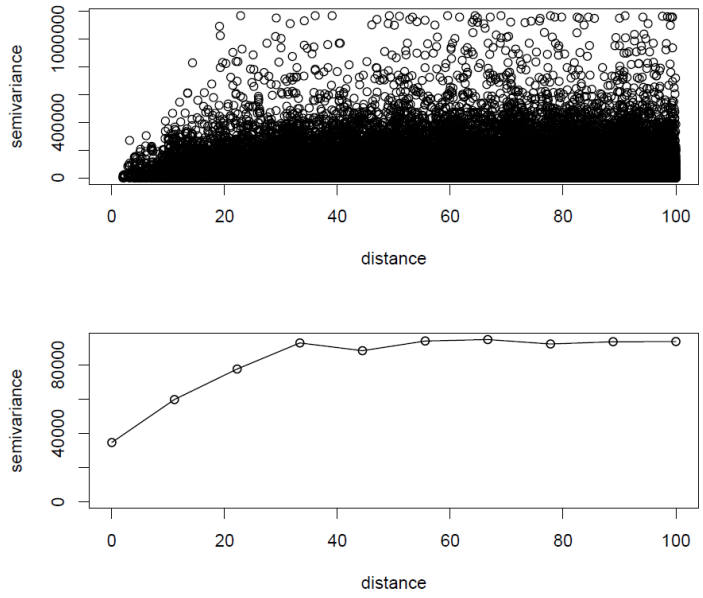

Figure 7: Variogram cloud and empirical variogram ($lag = 10\ m$) of V.

function or with an interactive tool `eyefit`. In our example in figure 8 we set the maximal distance to 100 m, the covariance model as exponential, the range to 25 m, the sill to 65000, and nugget to 34000.

4.4. Analysis of Multivariate Data

Since we wish to take advantage of spatial dependency of a primary and a secondary variable, we need to analyze the data sets. The goal is to examine whether the covariates are dependent enough so the secondary variable can improve prediction of the primary variable.

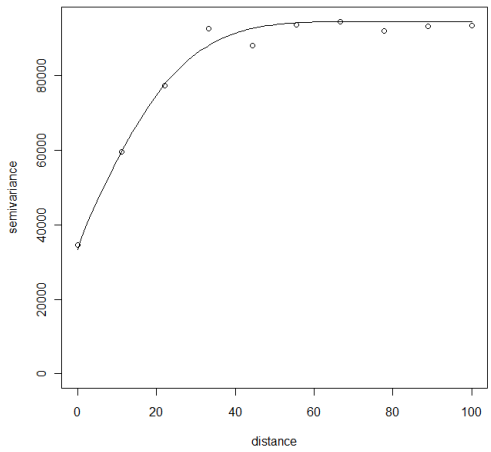

Figure 8: Theoretical variogram of V.

The first thing we can try is to compare the shape of histograms. Very similar shapes (i.e. similar distribution) indicates a certain degree of dependency.

By using `cor(U,V)` function in R we can get a correlation coefficient (in this case 0.837). Its value is always within the interval $\langle -1, 1 \rangle$. The closer to zero, the less dependent the data sets are.

In order to compare two distributions, we can visualize so–called q–q plot (`qqplot` function in R). Each axis represents quantiles of one data set (see figure 9). If the plotted data are close to $y = x$ line, the variables are strongly dependent. If the data make a straight line that has a different direction than $y = x$, the variables still have similar distribution but with different mean and variance.

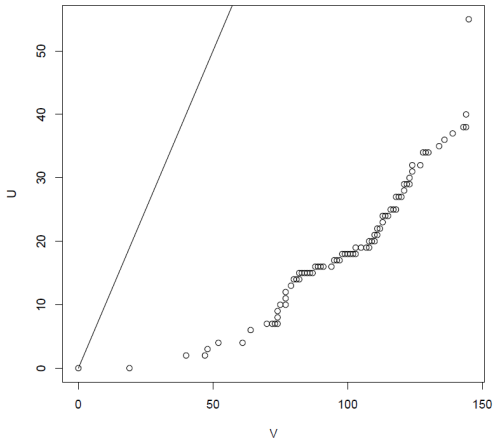

Figure 9: Q–q plot, straight line represents $y = x$, `qqplot(V,U)`.

Another graphical tool for testing the dependency of two spatial data sets is so–called *scatterplot*. Pairs made of primary variable value and secondary variable value at the same location are visualized as points in this plot. The result is a cloud of points (see figure 10 for our

example on U and V data). The narrower the cloud, the higher the degree of dependency. A *scatterplot* has one another big advantage — outliers and measurement errors lie outside the cloud. We can then easily check these points and in case they are wrong we would take them out of the data set. The dependency of two variables can be approximated by linear

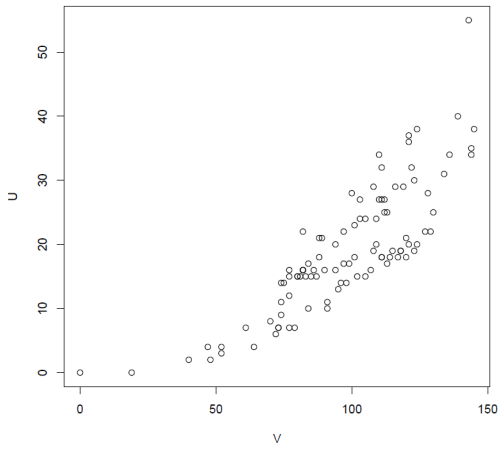

Figure 10: *Scatterplot*, `plot(V,U)`.

regression given by $y = ax + b$. How to do this in R is shown in the following code.

```
# method for linear regression
model = lm(U~V)

# plot
plot(V,U,main="Scatterplot and linear regression")
abline(model)

# model parameters
summary(model)
```

The plot from the previous example is in figure 11. An alternative for a linear regression can be a graph of conditional expectation where one variable is divided into classes (such as when we create a histogram) and a mean of the other variable is calculated within these classes, see figure 12.

Since we explored the data sets, did basic geostatistical analysis and determined the spatial continuity and covariables dependence, we might proceed to prediction. From now on, we are going to use a new data set that is more suitable as an example for prediction by (co)kriging.

4.5. Example of Kriging and Cokriging in R

In the following part of this paper, we are going to make two predictions — one using only primary variable on its own and ordinary kriging method, and the other using secondary variable and ordinary cokriging method. We are going to compare these two methods using some graphical and tabular outputs.

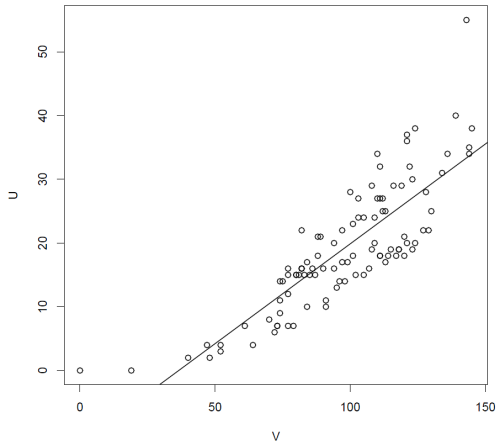

Figure 11: Linear regression $U = 0.314V - 11.5$.

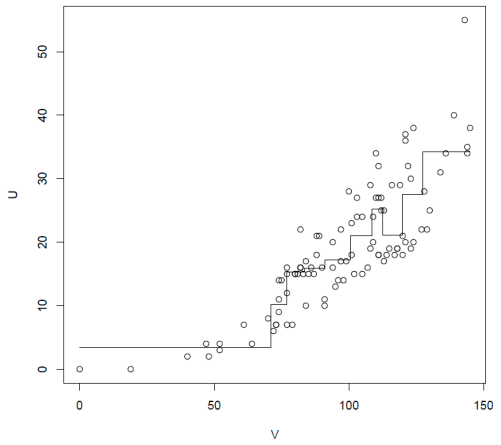

Figure 12: Conditional expectation of V within classes defined on U values.

4.6. Data Description

The phenomena we use in this example are simulated random fields in a square region of size of 50 pixels (i.e. 2500 pixels/values in total). We randomly[4] select some values and state them for measurement. After the prediction is made, we can easily compare the results with the original data set. This is not how it works in reality — we do not have values of the variable at each location of the region, that is why we do the prediction. However, for educational purposes, comparison of predicted and real values is a good way to show how these methods work and how well they work.

Simulation of Gauss Random Fields was chosen to create our phenomena by method `grf` in R. This method is able to create a random raster which can represent continuous spatial phenomenon. Gaussianity of the spatial random process is an assumption common for most standard applications in geostatistics. However non-Gaussian data are often provided. How

[4]The layout of the samples is not random — we try to cover the whole region and arrange the samples in a grid. However, the samples are randomly chosen from a neighborhood of each node of the grid.

to deal with this sort of data is decribed in detail in [9].

In our paper, two fields were created by function `grf`, each representing one variable (called A and B). A is our primary variable for which the prediction will be made. B is just an auxiliary variable for forming the secondary variable – C. C is strongly correlated with A, the correlation coefficient is about 0.93. All three fields are shown in figure 13. The R code of creating and plotting these three fields is following:

```
library(geoR)
library(gstat)

set.seed(1)
# creates regular grid of 50 by 50 pixels
# the covariance parameters are sigma^2 (partial sill)
# and phi (range parameter)
A = grf(50^2,grid="reg",cov.pars=c(1,0.25))
# all values of A are non-negative
A$data = (A$data+abs(min(A$data)))*100

set.seed(1)
# covariance model is set to matern
# smoothness parameter kappa is set 2.5
B = grf(50^2,grid="reg",cov.pars=c(1200,0.1),
    cov.model="mat",kappa=2.5)

C = A
C$data = A$data-B$data
# all values of C are non-negative
C$data = C$data+abs(min(C$data))

library(fields)
img_A = xyz2img(data.frame(A))
img_B = xyz2img(data.frame(B))
img_C = xyz2img(data.frame(C))

par(mfrow=c(2,2))
image.plot(img_A, col=terrain.colors(64), main="A",
           asp=1, bty="n", xlab="", ylab="")
image.plot(img_B, col=terrain.colors(64), main="B",
           asp=1, bty="n", xlab="", ylab="")
image.plot(img_C, col=terrain.colors(64), main="C",
           asp=1, bty="n", xlab="", ylab="")
```

Both, A and C, have normal distribution, and all values are non–negative for sake of easier presentation. The coordinates are in range $\langle 0, 1\rangle$. The basic statistics are in table 2.

The sample data set consist of 166 measured values of C and 63 values of A. The primary variable fully overlaps the samples of the secondary variable and the secondary variable sample

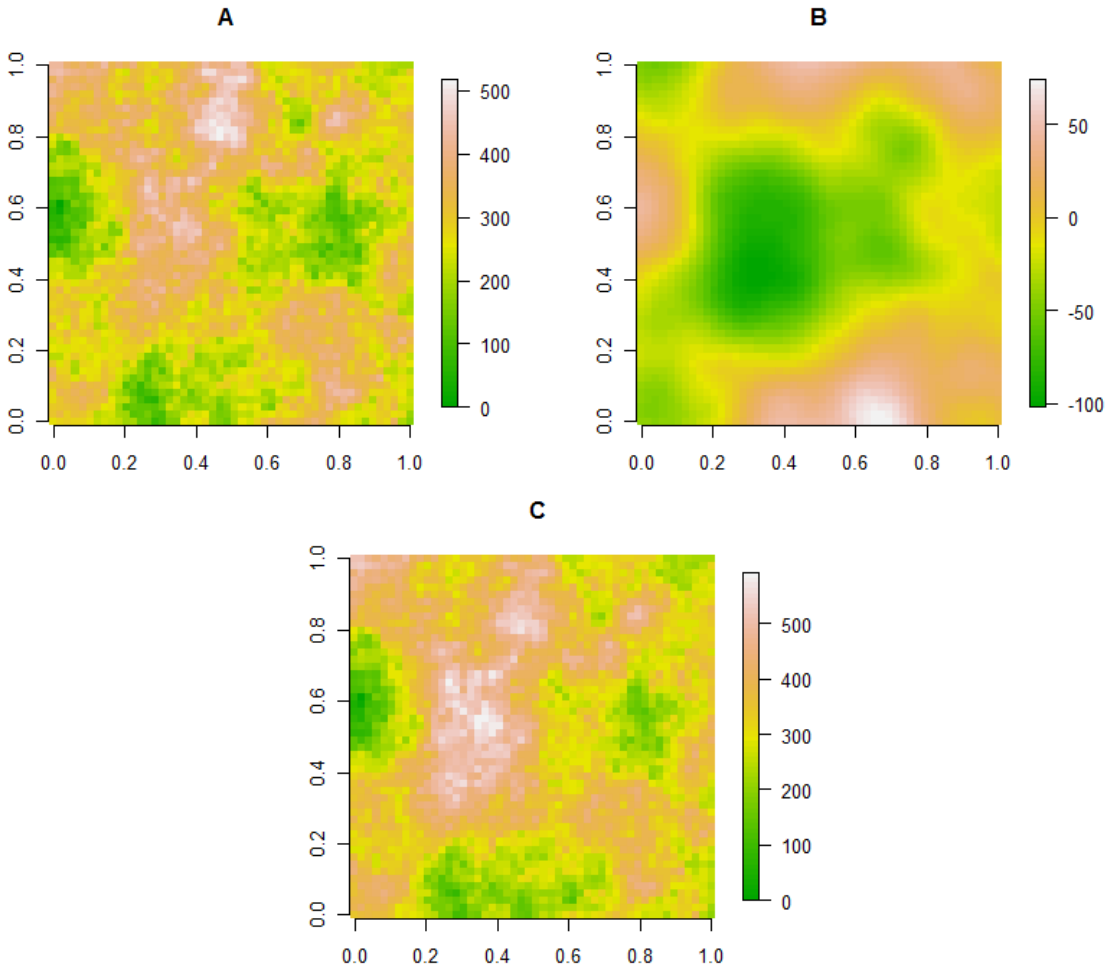

Figure 13: Simulated random variables.

grid is much more dense (see later in figure 18). Let us have a look at some analyzing graphical tools — histograms of samples are shown in figure 14, q–q plots are shown in figure 15, and a *scatterplot* is shown in figure 16. According to these plots, we can conclude that the samples have normal distribution and the distributions are quite similar which confirms the strong correlation of the variables.

4.7. Prediction Using Ordinary Kriging

Use of ordinary kriging in R is very simple. Once we determined the theoretical variogram we can proceed to the prediction. See the following code:

```
# create a grid for the prediction
gr = data.frame(Coord1=A$coords[,"x"],Coord2=A$coords[,"y"])
gridded(gr) = ~Coord1+Coord2

# assign coordinates to variable A
```

Variable	Number of values	Minimum	Median	Mean	Maximum
A (primary)	2500	0.0	289.8	286.1	517.0
C (secondary)	2500	0.0	342.8	342.6	590.5

Table 2: Basic statistics of primary and secondary variable.

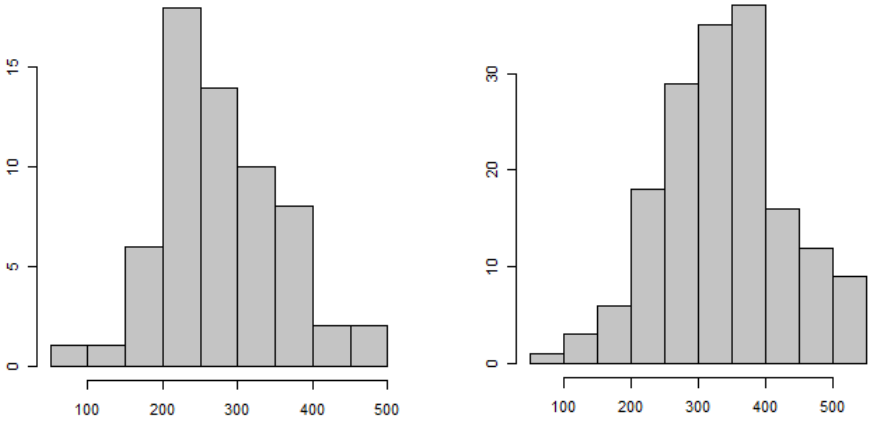

Figure 14: Histograms of A (left) and C (right).

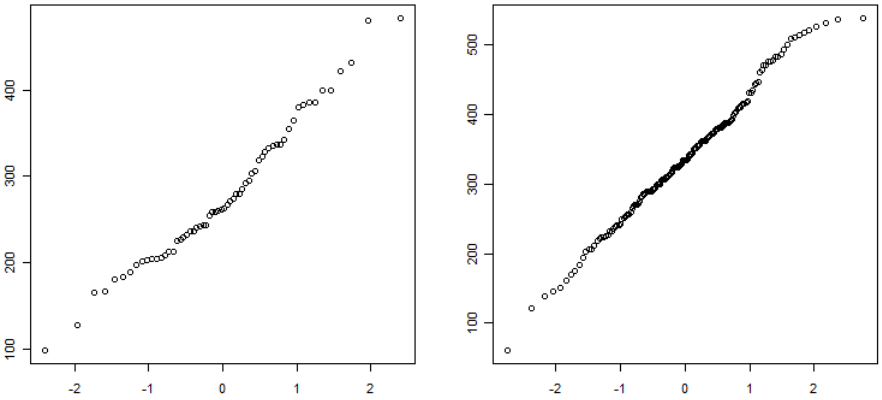

Figure 15: Q–q plots of A (left) and C (right).

```
coordinates(dataFrameA) = ~Coord1+Coord2

# variogram model
vm = variogram(data~1,dataFrameA)
vm.fit = fit.variogram(vm, vgm(6500, "Sph", 0.3, 50))

# prediction using ordinary kriging
OK_A = krige(data~1,dataFrameA,gr,vm.fit)
```

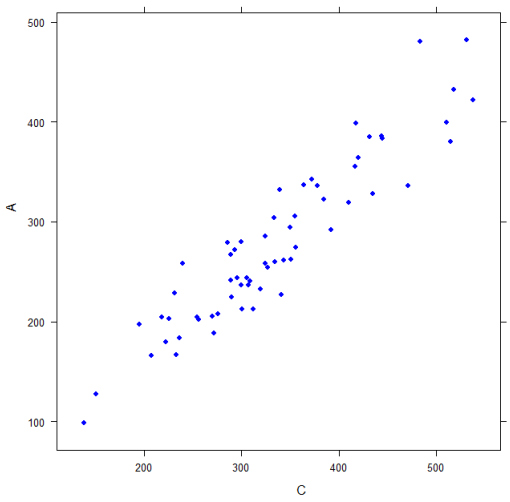

Figure 16: *Scatterplot* of A a C.

This is all we need to do to get prediction in unsampled locations when input is only the primary variable A. The results are shown in figures 18 and 19. Let us have a look how the process changes when we wish to include the secondary variable.

4.8. Prediction Using Ordinary Cokriging

A detailed description of how to process ordinary cokriging prediction in R is decribed in [3].

We already concluded that the variables A and C are spatially dependent. The most difficult step in prediction by ordinary cokriging is to set a linear model of coregionalization (in other words, to describe the spatial dependence between the covariables). We need to fit the samples into proper variogram and cross–variogram models. Follow the example in the code below:

```
# create a gstat object g
# (necessary for correct use in following methods)
# varibles A and C are saved in class data.frame
# add A and C to object g
g <- gstat(NULL, id = "A", form = data ~ 1, data=dataFrameA)
g <- gstat(g, id = "C", form = data ~ 1, data=dataFrameC)

# empirical variogram and cross-variogram
v.cross <- variogram(g)
plot(v.cross, pl=T)

# add variogram to object g
# vmA_fit is previously created variogram model
g <- gstat(g, id = "A", model = vmA_fit, fill.all=T)

#create linear model of coregionalization
g <- fit.lmc(v.cross, g)
```

```
plot(variogram(g), model=g$model)
```

The model of coregionalization is shown in figure 17. The upper figure is variogram of samples of A. The empirical variogram does not look good due to small number of input samples. Look at the improvement of variogram for C (lower right) where the number of samples is about three times larger. The lower left figure is the pseudo cross–variogram. The covariance model is identical (spherical in this case) for all three variograms, as well as the range was maintained (about 0.3). This means that the covariables behave similarly in space — they show the same degree of dependence for given distance. Since we gained linear model of coregionalization, we can proceed to prediction using ordinary cokriging.

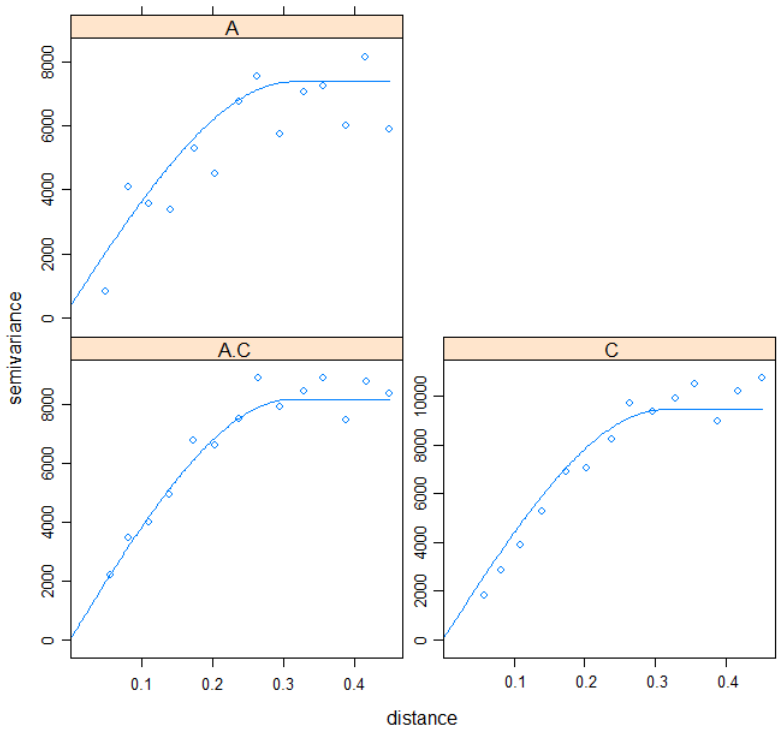

Figure 17: Variogram and pseudo cross–variogram of A and C.

The prediction step in R is actually very simple. It is literally a single command of method `predict.gstat` method. This method distinguishes (based on input data) what prediction method to use. There are actually two predictions made. One for our primary variable and one for the secondary one, because the method does not make a difference between those variables (i.e. we never specify which one is the primary one).

```
# gr is the prediction grid
CK <- predict.gstat(g, gr)
```

Comparisons of some statistics are listed in table 3. The contribution of C variable to the prediction of A is obvious. The extreme values got closer to real extreme values of A. The same holds for the median and mean. Values of variation of prediction got significantly lower.

Data	Min.	Med.	Mean	Max.	Mean of var.pred.	Max.var. of pred.
A real	0.0	289.8	286.1	517.0	–	–
OK A	98.2	277.4	280.8	482.4	2804	5329
CK A, C	42.3	279.6	281.0	482.4	1617	3293
Data	**Min. diff.**	**Mean diff.**	**Max. diff.**	**Med. of diff.**	**Med. of abs(diff.)**	**RMSE**
OK A	-153.5	-5.1	179.1	-4.1	32.2	49.1
CK A, C	-177.0	-5.1	166.4	-3.0	30.8	46.8

Table 3: Comparison of ordinary kriging (OK) a ordinary cokriging (CK).

RMSE stands for root mean square error:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(Z^*(x_i) - Z(x_i))^2}.$$

The best result presentation is visualization of the predictions (figure 18) and the prediction errors (figure 19). It is obvious that the cokriging prediction describes the regions with extreme values more precisely. However, we can see that the kriging prediction did a good job too. It is thanks to relatively sufficient number of samples and (more importantly) their proper layout. It is only on us to decide whether this prediction is accurate enough or not. If not, we need to provide the prediction with samples of another variable that is highly correlated with the primary one and that has more dense sampling. The question is whether the improvement is worth the cost of the secondary variable data set. Let us pay attention to the errors figure, particularly on the middle map with real errors. We can see that in case of ordinary cokriging a red cloud of errors appeared in the middle. This is a somewhat negative impact of the C samples. Let us recall that the C variable is derived not only from A but also from B variable (figure 13) that has a large region of negative values exactly in the place where the red cloud of errors appeared. This region effected the C samples as well as the final prediction of A. This may have a dangerous impact on the prediction when using a secondary variable. This is why the degree of dependency of the covariables has to be really high.

5. Conclusion

Both methods, ordinary kriging and ordinary cokriging, were shown to lead to a successful prediction. As we expected, the gain of the secondary variable was obvious. However, we always need to consider the cost of obtaining it and a the quality of the prediction without it. We did much more combinations of covariables during this project that were not mentioned in the paper. We worked with yet another variable that was not so correlated to the primary one. The results in that case were not good which we expected. We tried different sample layouts for primary and secondary variable. The biggest gain in prediction was achieved when the primary data set was so sparse that prediction by ordinary kriging was almost impossible to process (we cannot create the variogram). By adding the secondary variable, the prediction gave us quite decent results. We also tried to use the same primary variable as in this paper and the secondary variable just with the difference in sample locations — they did not overlap with the primary variable samples (their count was still about three

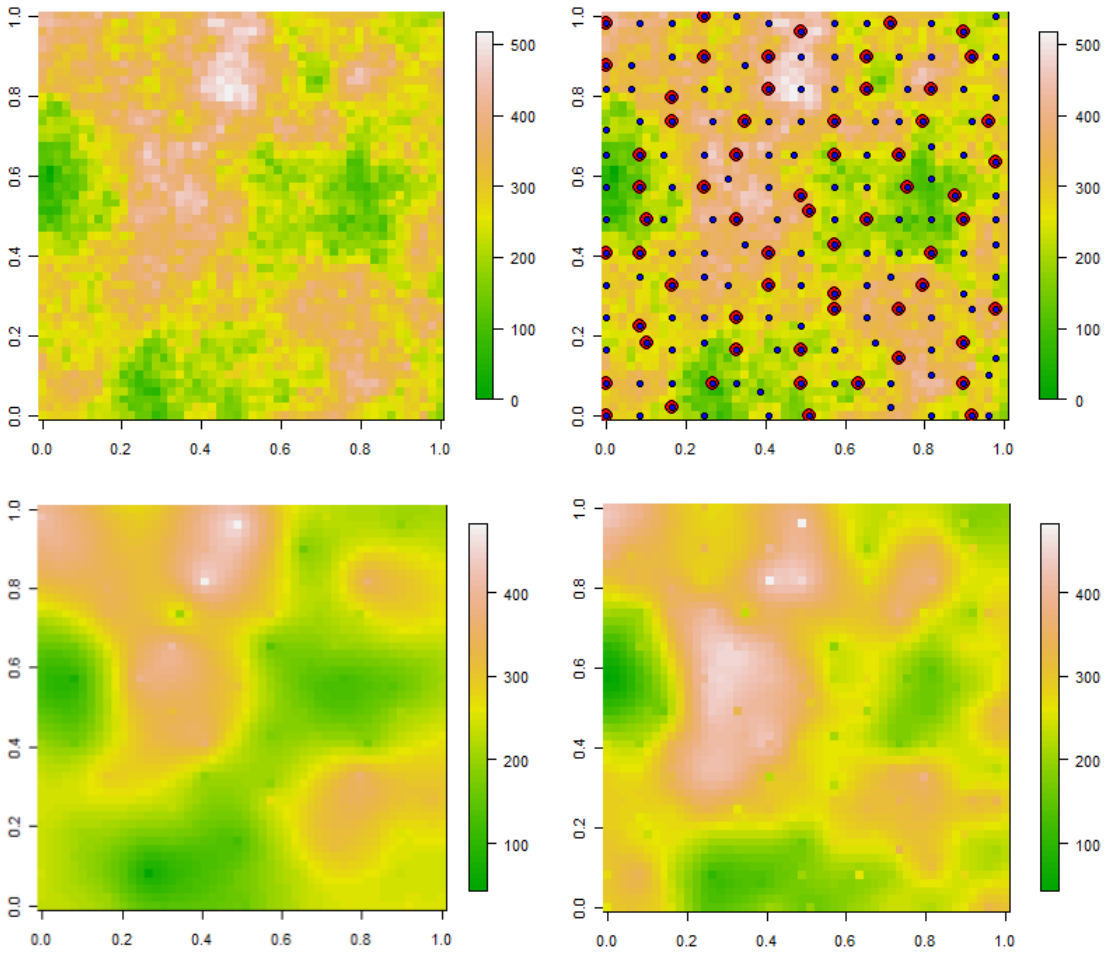

Figure 18: Ordinary kriging and ordinary cokriging for A and C (left upper – real values of A, right upper – samples (red – A, blue – C), left lower – ordinary kriging, right lower – ordinary cokriging).

times higher than number of samples for primary variable). This is the case where we cannot tell how good the spatial dependency of the covariables is and so it is harder to create the linear model of coregionalization. Results of such prediction were not that good as in the case presented in this paper, however we still managed to enhance the prediction of the primary variable.

This paper was originally made for educational purposes. It shows how to do basic spatial data analysis and how to predict values of some phenomenon in unsampled locations. Two methods were described — ordinary kriging and ordinary cokriging. Readers of this paper were provided with a step–by–step prediction process in R environment.

Acknowledgments *The project was supported by grant SGS11/003/OHK1/1T/11. Many thanks belong to Prof. Dr. Jürgen Pilz who became a great inspiration leading to including geostatistics and project R into the Geoinformatics programme at the Czech Technical University.*

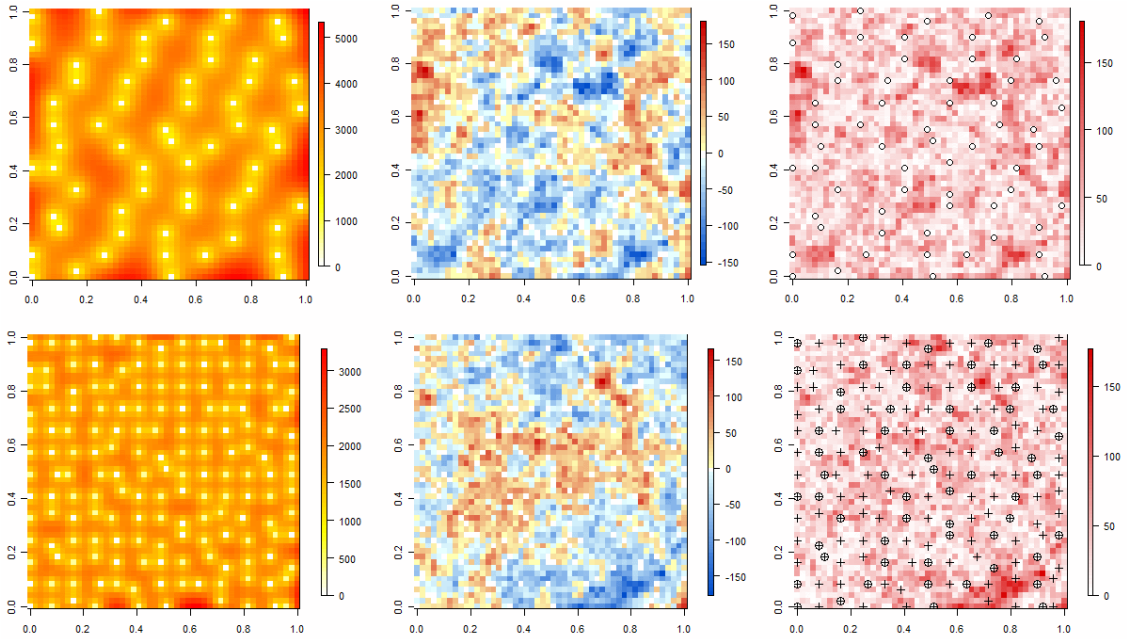

Figure 19: Prediction errors. Upper row for ordinary kriging, lower row for ordinary cokriging; Left: Variation of prediction, middle: Real estimation errors, right: Absolute values of estimation errors (circle – A, plus – C).

References

[1] Wackernagel, H. (2003): Multivariate Geostatistics. - 3rd edition. - Springer, Germany.

[2] Isaaks, E. H.; Srivastava, R. M. (1989): Applied Geostatistics. - Oxford University Press, New York.

[3] Rossiter, D. G.: Co-kriging with the gstat Package of the R Environment for Statistical Computing. - Web: http://www.itc.nl/ rossiter/teach/R/R ck.pdf.

[4] CRAN Task View: Analysis of Spatial Data. - Web: http://cran.r-project.org/web/views/Spatial.html.

[5] The Comprehensive R Archive Network. - Web: http://cran.r-project.org.

[6] Cressie, N. (1993): Statistics for spatial data. - Wiley Interscience.

[7] Hengl, T.: A Practical Guide to Geostatistical Mapping. - 2nd edition. - Office for Official Publications of the European Communities, Luxembourg. - Web: http://spatial-analyst.net/book/.

[8] Diggle, P. J.; Riberio, P. J. Jr. (2007): Model–based Geostatistics. - Springer.

[9] Pilz, J. (Ed.) (2009): Interfacing Geostatistics and GIS. - Paper: Bayesian Trans-Gaussian Kriging with Log-Log Transformed Skew Data by Spöck G., Kazianka H., and Pilz J.. - Springer.