

USER MANUAL

Introductory Notes

This software is a simple interface for compiling R script interactively. After the script is compiled it runs it in batch mode. All the computations are done using packages already present in R and NOT CREATED BY THE AUTHOR OF THIS SOFTWARE INTERFACE.

The packages with the relative authors are listed below:

- **GSTAT**: Pebesma, E.J., 2004. Multivariable geostatistics in S: the gstat package. Computers & Geosciences, 30: 683-691.
- **sp**: Pebesma, E.J., R.S. Bivand, 2005. Classes and methods for spatial data in R. R News 5 (2), <http://cran.r-project.org/doc/Rnews/>. Roger S. Bivand, Edzer J. Pebesma, Virgilio Gomez-Rubio, 2008. Applied spatial data analysis with R. Springer, NY. <http://www.asdar-book.org/>
- **maptools**: Nicholas J. Lewin-Koh and Roger Bivand
- **rgdal**: Timothy H. Keitt, Roger Bivand, Edzer Pebesma, Barry Rowlingson
- **lattice**: Sarkar, Deepayan (2008) Lattice: Multivariate Data Visualization with R. Springer, New York. ISBN 978-0-387-75968-5
- **akima**: Albrecht Gebhardt (1998). Akima: Interpolation of irregularly spaced data

Note on R

R is a very powerful statistical language. However it is also a slow language and this is clear in geostatistical interpolations. For this reason, especially when it is interpolating with inverse distance, it takes some time to finish the process. Do not worry too much and let it works.

System Requirements

The system requirements are the same of R.

In this version of IntR3D is included the version 2.12.0 of R, that is used to run the scripts. This version of R is compatible with 32bit machines. On these machines R can run but it has some memory limitations. It can use a maximum of 4 Gb of RAM (even though in some machines the amount is limited at 2 or 3 Gb of RAM). This means that sometimes R will not be able to complete the process because it is not able to allocate the data onto the RAM.

However, there are some ways to limit the amount of memory necessary for complete the process. The common tip is to reduce the size of data and covariates. In IntR3D the user can choose in every step of the process the resolution of the ascii grid and the resolution of the map that he wants to create. This way, if the user notices that R does not complete the process because of a memory related problem, he can just decrease the resolution and try again. The only way to increase the memory that R can use for its computation is to use a 64bit machine and IntR3Dx64.

Another limitation regards the use of the CPU. By default R can use only one core. Therefore, in the task manager, when R is performing the prediction will indicate that the CPU usage is low. This is because is using a fraction of the virtual power of the machine.

Data Format

IntR works with data file arrange in a textual table, in txt format. The structure of the file must be like the following example:

ID	Lat	Lon	value	depth
1	xxxxxx.xx	yyyyyy.yy	ii	dd
2	xxxxxx.xx	yyyyyy.yy	iii	ddd
3	xxxxxx.xx	yyyyyy.yy	iiii	dddd
4	xxxxxx.xx	yyyyyy.yy	iiii	dddd

The format of the two spatial coordinates is not important, because within the sp package in R there is the possibility to set the projection of your data, according to the proj4string format (<http://spatialreference.org/>). It is important tough, the format of the depth values, they have to be in centimetres and positives (not -10 but 10).

Once the user insert the correct projection of his data, all the interpolation will consider the Euclidean distance between data points.

The important thing is that the header of the file is exactly the same of the above example.

Covariates format

All the covariates need to be in ascii grid. In IntR there is a module that converts the data between the ESRI shape file and the ascii grid format. In this module the user can convert a raster into an ascii grid, selecting the resolution of the final grid.

Starting IntR

When the user starts IntR, the software shows a window with a button for each module of the program. The user can, very intuitively, click on the button that corresponds to the module he wants to start. To consult this user manual the user can click on the menu “File” and on the menu item “User Guide”. Each module corresponds to a different task that will be compiled interactively in IntR and executed with R. Following there are detailed instructions for each module.

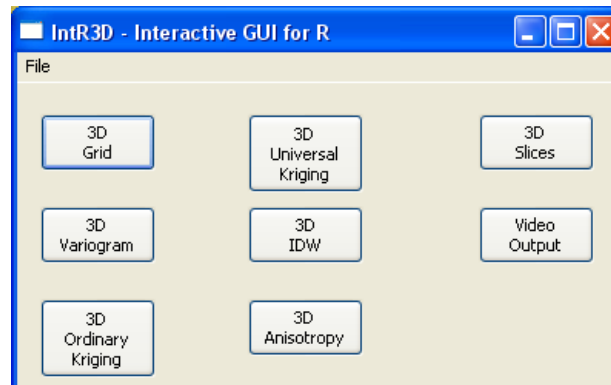


Figure 1: starting windows of IntR3D. The user here can intuitively select the algorithm that he needs to use.

3D Variogram & 3D Anisotropy

In IntR3D there are two modules that are used for performing some preliminary studies upon the study area. These two modules are “3D Variogram” and “3D Anisotropy”.

3D Variogram

This module studies the spatial variability of the dataset performing the variogram computation. The results are a series of three images, the omnidirectional variogram, four directional variograms a 0, 45, 90 and 135 degrees, a variogram map that can be used to assess the anisotropy of the field and a series of variogram with different dips and directions.

Installation

Simply run the script START.exe and select 3D Variogram from the buttons.

In order to run the software all the data and covariates that you need for running the interpolation must be in the folder c:\RHOME.

Running the Program

The working flow for Variogram Plot is the following:

- select data file
- input of ID number of the projection
- select number and covariates files
- input the linear regression model
- select the variogram model
- execute the cross-validation

This module can be used for an ordinary kriging variogram or for computing a regression variogram, which uses a linear regression module. Even if the user wants to perform a simple ordinary kriging variogram, he needs to insert a covariate file that will be used for the variogram boundary.

Results

At the end of the module, IntR3D displays a window with four buttons that are used to show the images of the variograms.

3D Anisotropy

Anisotropy is a module that is used to provide the user with the five values used to assess the geometrical anisotropy of the field.

A geometrical anisotropy is evident when the directional variograms show the same sill but different ranges. To correct this anisotropy the user needs to insert five parameters:

- the angle of the direction of maximum continuity, in positive degrees from north
- the dip in the direction of maximum continuity
- the rotation angle of the other two axis
- the last two values are ratios between the main axis and the other two

Installation

Simply run the script START.exe and select 3D Anisotropy from the buttons.

In order to run the software all the data that you need for running the interpolation must be in the folder c:\RHOME.

Running the Program

This module asks only for the data file, the projection and an ascii grid for bounding the variograms.

Results

At the end of the module, IntR3D displays a window with the values of the five parameters used to correct the geometrical anisotropy. Those 5 values are saved in a txt file called parameters.txt within the RHOME folder. To correct the anisotropy the user needs to insert manually the five parameters when asked in one of the kriging modules.

3D Grid

IntR3D has been created to be used like a unique working flow, rather than a series of separate modules like IntR. For this reason the creation of the 3D grid to be used as new data for the final prediction in the kriging modules is performed in a separate module.

Installation

Simply run the script START.exe and select 3D Grid from the buttons.

In order to run the software all the data that you need must be in the folder c:\RHOME.

Running the Program

This module asks for:

- the data file
- the horizontal size of the prediction grid (the vertical resolution is 1m)
- absolute or relative altitude
- an ESRI shp file of the area

The user can choose between absolute or relative altitude. If he select relative altitude, the software creates a grid that start from a depth of 0m and proceed by one till reaching the maximum depth of the data file. If the user select absolute, he needs to import also a DTM file in ascii grid and the program will correct all the depth based on the real altitude of the point.

Results

At the end of the module it does not show any summary windows it just close itself after it finished creating a file called Pred_Grid.txt into the RHOME folder.

This file can be used with the modules 3D Universal Kriging and 3D Ordinary Kriging.

3D Universal Kriging

This python script will help you compiling and running a script able to perform a 3D Universal kriging on your data. The kriging is performed with the package Gstat.

In the 3D Universal Kriging the cross validation is not performed with the default function implemented in GSTAT. This function in fact, performs an n-fold cross validation extracting n data and repredicting their value. But this approach, when the dataset has been created resampling a spline is not appropriate because the kriging algorithm will use neighbourhood values, from the same samples, to repredict the missing data. And the neighbourhood values are related, to the missing data, by a nonlinear regression equation that simplifies the prediction.

For this reason in this module has been implemented a method that extract an entire sample and repredict it using all the others. This way the result of the validation is more accurate.

Installation

Simply run the script START.exe and select 3D Universal Kriging from the buttons.

In order to run the software all the data and covariates that you need for running the interpolation must be in the folder c:\RHOME. All the covariates need to be in ascii grid (*.asc) format.

Running the Program

The working flow for 3D Universal Kriging is the following:

- select data file
- input of ID number of the projection
- select the prediction-grid size
- select number and covariates files
- input the linear regression model
- select the variogram model
- execute the cross-validation
- perform the kriging

The first information the program needs, after the data file, is the ID number of the projection. For more information see paragraph "Projection" below. The data file as be a txt and the format has to be the same of the one presented in the Data Format paragraph.

In IntR3D the Universal Kriging module does not create the prediction grid itself but it imports it from the output of the module 3D Grid.

Subsequently, the program will ask for the covariate number and to select the covarites files.

The user must be careful during this step because if he needs to insert 2 covariates, for example, the second file selection window, for selecting the second covariate, will appear very shortly after the first.

At this point the user must insert the linear regression model. The model has to be in the form:
covariate1+covariate2+.....+covariateN

Ex.

if the two covariates are named slope.asc and em38h.asc the model will be:

slope.asc+em38h.asc

it is important to insert the entire name of the file, extension included.

When the user finishes input the linear regression model, IntR asks for the variogram model. It important to insert the name as it is shown on the screen.

Ex.

on the screen there will be a message like,

"Please write the model of the variogram.

Type:

- 0 for the default model,
- Sph for Spherical Model,
- Exp for Exponential model,

- Gau for Gaussian model
- Mat for the Matern. "

if the user wants the spherical model, he needs to insert Sph, with the first letter in capital.

Subsequently a dialog appears and asks for the five anisotropy parameters. The user, if an anisotropy is presents, needs to insert the five numbers separated by a comma. If no anisotropy is present in the field, the user needs to insert a single 0.

At this point, the cross-validation starts.

After it finishes the cross-validation, it shows a window with the R Squared value and a button that shows the variogram image, which is also saved on "C:\".

If the result of the cross-validation is satisfying, you can click on "YES" and automatically proceed with the kriging with the same linear regression model. If the result is not good you can click on "NO" and repeat the cross-validation selecting again the linear regression model and all the parameters of the variogram.

Projection

To set the projection of the map you need to insert the ID number of the coordinates system. You can find a list of ID number and the relative system here:

<http://spatialreference.org/ref/?search=utm&srtext=Search>

At the moment, IntR supports only EPSG projections, so you need to insert a number of an EPSG projection. If your data are unprojected insert 0 in the apposite space.

It is important to insert the projection ID, if the data are projected, because the package GSTAT has been designed to work with projections. This means that if the projection is available, the interpolation is done using Euclidean Distances.

Output

The output of the script are:

- variogram plot in jpg
- prediction map in jpg
- kriging error map in jpg
- cross-validation of the last used model in txt
- kriging results table in txt
- ascii grid of the map
- shp file with predicted points

3D Ordinary Kriging

The Ordinary kriging script is similar to the other two kriging script.

The format of the data file is the same, coordinates columns called Lat and Lon and the data column called value.

The difference of course is that with the ordinary kriging you don't need to insert covariates. However you still need to insert an ascii grid for interpolating the final prediction map.

The ascii grid can be the result of the SHPtoASC script.

The series of information needed by the software are similar to the one needed for universal kriging.

Installation

Simply run the script START.exe and select Ordinary Kriging from the buttons.

In order to run the software all the data and covariates that you need for running the interpolation must be in the folder c:\RHOME.

All the covariates need to be in ascii grid (*.asc) format. To convert a table with Latitude, Longitude and covariate value into .asc you can use the Inverse Distance algorithm in IntR or any other GIS software. To convert an ESRI raster into an ascii grid use the module from SHP to ASC.

Running the Program

The working flow for Universal Kriging is the following:

- input of ID number of the projection
- select data file
- select the prediction-grid size
- select number and covariates files
- input the linear regression model
- select the variogram model
- execute the cross-validation
- perform the kriging

The first information the program needs is the ID number of the projection. For more information see paragraph "Projection" below.

After that, the program will ask you to choose the data file.

The data file can be a txt and the format has to be the same of the one presented in the Data Format paragraph.

In the next input windows IntR asks to insert the grid size of the prediction grid.

If the user inserts 10, for example, it means that IntR will predict a value on a square grid of dimension 10x10 m.

Subsequently, the program will ask for an ascii grid. This will be used to bound the variogram.

Ex.

on the screen you will see a message like,

"Please write the model of the variogram.

Type:

- 0 for the default model,
- Sph for Spherical Model,
- Exp for Exponential model,
- Gau for Gaussian model
- Mat for the Matern. "

if the user wants the spherical model, he needs to insert Sph, with the first letter in capital.

Subsequently a dialog appears and asks for the two anisotropy parameter. The user, if an anisotropy is presents, needs to insert the two numbers separated by a comma. If no anisotropy is present in the field, the user needs to insert a single 0.

Ex.

Parameter p is 45 and the parameter s is 0.5.

The user has to insert: 45,0.5

At this point, the cross-validation starts.

After it finishes the cross-validation, it shows a window with the R Squared value of it and a button that shows the variogram image, which is also saved on "C:\".

If the result of the cross-validation is satisfying, you can click on "YES" and automatically proceed with the kriging with the same linear regression model.

If the result is not good you can click on "NO" and repeat the cross-validation selecting again the linear regression model and all the parameters of the variogram.

Projection

To set the projection of the map you need to insert the ID number of the coordinates system. You can find a list of ID number and the relative system here:

<http://spatialreference.org/ref/?search=utm&srtext=Search>

At the moment, IntR supports only EPSG projections, so you need to insert a number of an EPSG projection.

If your data are unprojected insert 0 in the apposite space.

It is important to insert the projection ID, if the data are projected, because the package GSTAT has been designed to work with projections. This means that if the projection is available, the interpolation is done using Euclidean Distances.

Output

The output of the script are:

- variogram plot in jpg
- prediction map in jpg
- kriging error map in jpg
- cross-validation of the last used model in txt
- kriging results table in txt
- ascii grid of the map
- shp file with predicted points

3D Inverse Distance

This program performs an inverse distance interpolation in 3D.

The data file can be a txt and the format has to be the same of the one presented in the Data Format paragraph.

The main purpose of this module is to create a continuous lattice from a series of data points in 3D, which are the result of the 3D kriging.

At the end of the 3D kriging, in fact, the program performs an interpolation in each point of the prediction grid. The result is a series of points with a predicted value for the property of interest. However, there is no way, in R, to visualize this prediction as a continuous surface. In order to do so, the only way is to perform a further interpolation.

Running the Program

The working flow for Inverse Distance is the following:

- select the SHP file with the prediction area
- select the size of the prediction grid
- select data file
- perform the inverse distance

This module was included for increasing the number of point in the prediction grid. The main purpose is to visualize the results, properly.

This module can be used to perform an inverse distance interpolation on a 3D dataset, but it lacks the cross-validation part. So the user can not be sure about the precision of the prediction.

This module is very easy to use, because it asks only for few information and performs the interpolation.

Results

The result is a file, called “3Dinverse.txt” which can be used with the subsequent module “3D Slices”.

3D Slices

This module creates a series of slice images from the 3D dataset. This module works by selecting, from the 3D file, a single value of depth, extrapolating all the values from that depth and creating an interpolated image of the selected depth. The same procedure is done also for latitude and longitude.

Running the Program

The working flow for Inverse Distance is the following:

- select the 3D file
- select the type of image
- select the depth interval
- select the latitude interval
- select the longitude interval

The execution of this module is very simple. The user needs to select the 3D file, usually a txt generated by the module 3D IDW.

Then he needs to select the type of image he wants as output. The user has three choices: a coloured image, a coloured image with contours and finally a contour image.

The next step is to select the depth, latitude and longitude interval of the slicing process. The user can type 0 to let the program create a slice for each depth, latitude or longitude. But he can also select an interval, if he is not interested in the entire field.

3D Video

This module copies the file mencoder.exe (www.mplayerhq.hu), which is a very powerful encoder that works with command lines, into the folders created by the module “3D slices”. After this step, the module starts mencoder and performs the creating of the 3 videos, one for depth, one for latitude and one for longitude.