

Some More Useful Programs

John G. Manchuk and Clayton V. Deutsch

The variety of problems that are worked on at the CCG occasionally requires the development of customized programs to perform specific tasks. These programs are not always distributed with the annual report because they are typically not generalized and may not apply to other similar problems due to some minor difference. This note provides a summary of some of the general programs developed by the authors that were used for various research purposes over the 2011 – 2012 term. Most of these programs are like typical GSLIB programs that take a parameter file; however, some of them are small requiring only two or three parameters so these were written as command-line programs that do not take a parameter file. All programs work with space delimited data files with the following format:

Line	Content
1	Title
2	Number of variables or columns, n
3	Variable 1 name
4	Variable 2 name
5	...
6	Variable n name
7+	Numeric data

Programs and parameters

summarystats

This program provides a summary of the statistics of a set of variables from a data file, possibly separated by a categorical variable if specified. Statistics include the sample size, number trimmed based on user specified bounds, mean, standard deviation, minimum, maximum, P5, P50, and P95. Users can gain a quick understanding of their data and how each variable is distributed. The minimum and maximum can be useful in other programs such as SGSIM where they are used to limit the magnitude of a variable during the back-transformation process. Parameters are as follows:

Line	Parameters
1	START OF PARAMETERS:
2	data.dat -file with data
3	5 7 8 9 10 11 - number of variables and columns
4	6 - column for category, 0 if none
5	0 0 - number of categories and codes, 0 for all
6	-998. 99999. - trimming limits
7	summarystats.out -file for output summary statistics

splitbylith

This is a simple command-line utility program that extracts all the lines from a data file associated with a specific category and saves them to a new data file. In many cases, we assess statistics of data such as the histogram and variogram by category; however, many programs that calculate these statistics do not allow users to specify a category so the data must be partitioned first. At a command prompt such as DOS or Cygwin, the program is executed with the following command:

```
splitbylith data.dat 3 51 data_51.out
```

Where the first argument is the program name, second is the input data file, third is the column with the categorical variable, fourth is the category of interest, and fifth is the output data file. If file names contain spaces, they must be specified in double quotations, for example:

```
splitbylith "data file.dat" 3 51 data_51.out
```

Executing the program with no arguments will output usage instructions.

gsl2csv

This is a command-line utility that converts a data file from the typical GSLIB format to a comma-delimited file and vice versa. Resulting files can be loaded quickly into software such as Microsoft Excel. The comma-delimited format is also used by some of the commercial geomodeling software packages. At a command prompt, the program is executed with the following command:

```
gsl2csv data.dat data.csv
```

Where the first argument is the program name, second is the input data file, and third is the output data file. The program determines the type of conversion based on the format of the input data file which is determined automatically. Executing the program with no arguments will output usage instructions.

maxcolumnindex

This is a command-line program that reads through an input data file and records the column number that contains the maximum value of all columns for each row. When this program was written, it's purpose was to process data files that contain categorical proportions that may have originated from trend modeling or kriging. In this case, by recording the column with maximum proportion, we can quickly generate a majority-vote categorical model. At a command prompt, the program is executed with the following command:

```
maxcolumnindex data.txt -98 99 maxcol.out
```

Where the first argument is the program name, second is the input data file, third and fourth are minimum and maximum trimming limits respectively, and fifth is the output file name. This program assumes all columns in the input file are relevant. Executing the program with no arguments will output usage instructions.

subgrid

This program is used to extract a portion of data from a file that contains data stored on a GSLIB grid. Users can extract a specific section of a grid defined by index ranges along each grid axis. The extracted portion is written to a new file that can be processed independently of the rest of the grid from which it was extracted. Parameters are as follows:

Line	Parameters	
1	START OF PARAMETERS:	
2	sgsim.out	-file with gridded data
3	50 50 50	- nx, ny, nz
4	10 40	-index range in x to export
5	10 40	-index range in y to export
6	5 10	-index range in z to export
7	(f9.3,x,f9.3,x,f9.3)	-format for output in brackets
8	sgsim-sub.out	-file for output

For these parameters, the gridded input file on Line 2 is subset from a 50 by 50 by 50 grid into a 30 by 30 by 5 grid according to the indexes on Lines 4, 5 and 6. The format specification on Line 7 is used to control the format of the output numbers. Format specifiers have the following syntax: AB.C where A is the numeric format (F for real and i for integer), B is record width (the number is allocated 9 total spaces in the example parameters), the period is specified for real numbers only, and C is the number of spaces allocated after the decimal. In the example parameters, the number would look like 12345.678 which has 9 total spaces including the decimal and 3 values after the decimal. The primary purpose of Line 7 was for

fast reading and writing of data from and to the hard drive and will likely be removed in future versions with the increased use of solid-state drives as it is somewhat cumbersome.

gridsmooth

This program applies a high pass filter to a data file containing numbers stored on a GSLIB grid. The filter is an ellipsoidal volume. For any given grid cell being smoothed, all grid cells within the ellipsoid are used in the averaging process. A variety of different weight functions can be used to assign weight to each of the grid cells based on their distance from the cell being smoothed. Parameters are as follows:

Line	Parameters	
1	START OF PARAMETERS:	
2	sgsim.out	-file with gridded data
3	5	- column for variable
4	100 100 1	- nx, ny, nz
5	1.0 1.0 1.0	- cell size in x, y, z
6	-998. 9999.	- trimming limits
7	2 2 1	-search radii in x, y, z
8	90.0 0.0 0.0	-search angles
9	4	-weight function type
10	smooth.out	-file for output

Weight function on Line 9 include (type): equal weighting (1), Gaussian weighting (2), quartic spline (3), and inverse distance (4).

sgsimstrat

This program is identical to the GSLIB program SGSIM with one additional parameter that is specified after the variogram and is the average variogram parameter, $\bar{\gamma}$. The value is the integral of the variogram specified for SGSIM through the volume covered by the grid that will be used for generating realizations on. Using the input $\bar{\gamma}$, the program computes the expected standard deviation of the mean of the realizations to be generated, which is equal to $\sigma = (1 - \bar{\gamma})^{1/2}$. According to the central limit theorem, the distribution of means will be normal with mean zero and standard deviation σ . The program then predetermines the mean that each realization will receive by sampling this normal distribution with L equally probable values, where L is the number of realizations to be generated. The primary application of this version of SGSIM is for cases where users are limited to a small number of realizations and where the variogram range is larger than the domain being modeled ($\bar{\gamma} < 1$). By stratifying the means of the realizations, we ensure a good spread through low to high with relatively few realizations. Parameters are as follows, details can be found in Deutsch and Journel (1998):

Line	Parameters	
31	4	- column for secondary variable
32	1 0.1	-nst, nugget effect
33	1 0.9 0.0 0.0 0.0	-it, cc, angl, ang2, ang3
34	10.0 10.0 10.0	-a_hmax, a_hmin, a_vert
35	0.5	-gammabar for stratified means

The average variogram parameter is on Line 35. If this line is omitted completely from the parameter file, the legacy version of SGSIM will execute.

blocksis

The original version of BLOCKSIS written by Deutsch (2006) was updated to handle locally varying anisotropy. Local parameters that are permitted to vary include the azimuth, dip, horizontal anisotropy, and vertical anisotropy. These parameters are not used for anisotropic distance calculations as in Boisvert et al (2009), rather, they are used to update the local search ellipsoid and covariances calculated for kriging. Parameters are as follows:

Line	Parameters
1	START OF PARAMETERS:
10	trendmodel.out -file with gridded prior mean values
11	2 1 - columns for each category
12	3 - 2-D areal map (2) or 3-D cube (3)
13	1 -Consider locally varying search, 1=yes
14	localsearch.out - file with gridded local azimuth and dip
15	2 1 0 0 - columns for azimuth and dip, haniso, vaniso
16	-998. 999 - trimming limits
17	keyout.dat -file with keyout array
18	1 - column for keyout indicator

Parameters are explained in Deutsch (2006). New parameters are on Lines 13 to 16. If these parameters are not present in the parameter file, or the parameter on Line 13 is set to 0, the original version of BLOCKSIS is executed. Not all local search variables are required; it is possible to use only a locally varying azimuth or dip, or only a varying anisotropy ratio. If the resulting realizations are cleaned based on Line 3, the local search parameters are used to compute the weights used in the image filter. An unconditional example is provided in Figure 1. In this example, the dip varied from 45 degrees at the top to zero degrees at an elevation of 25 units, below which the dip remained 0. The azimuth varied from 90 degrees at 0 units North to 0 degrees at 100 units North.

References

Boisvert, J.B., Manchuk, J.G., Deutsch, C.V., 2009. Kriging in the presence of locally varying anisotropy using non-Euclidean distance. *Mathematical Geosciences*, 41, 585-601

Deutsch, C.V., 2006. A sequential indicator simulation program for vategorical variables with point and block data: BlockSIS. *Computers & Geosciences*, 32(10), 1669-1681

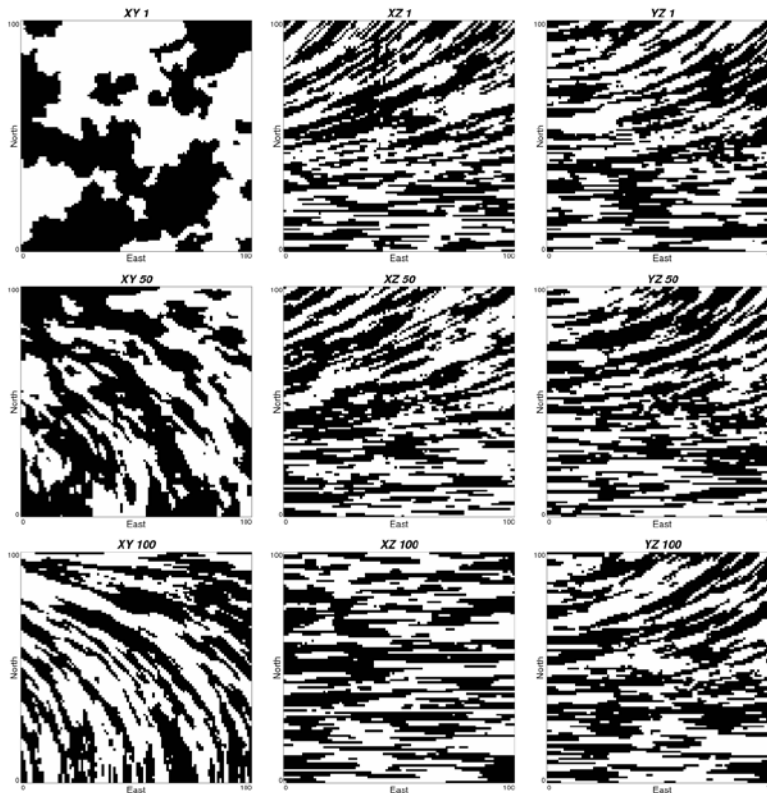


Figure 1: Slices from an example unconditional model using locally varying anisotropy in BLOCKSIS.