1 Introduction

The general relative variogram (Deutsch and Journel, 1997) is defined as

$$\gamma(h) = \frac{1}{2N_h} \sum_{i=1}^{N_h} \left( \frac{2(Z(s_i) - Z(s_i + h))}{Z(s_i) + Z(s_i + h)} \right)^2.$$

It is claimed to reveal spatial structure (correlation) better when data are skewed and/or clustered. The cluster.dat data set used in this vignette, from the GSLIB distribution\(^1\), seems to confirm this.

From version 1.02 on, R package gstat provides computation of the pairwise relative semivariogram. The following code provides an example and verification of the computation using direct R code and using the GSLIB program gamv.

The following code imports the cluster.dat data from GSLIB, which has been converted to have a single-line header containing column names, packaged with the R gstat package, and converts it into a SpatialPointsDataFrame object:

```r
> library(gstat)
> cluster = read.table(system.file("external/cluster.txt", package="gstat"), header = TRUE)
> summary(cluster)
```

1

---

\(^1\) F77 source code for Linux, downloaded Aug 28, 2011 from http://www.gslib.com/
Median : 1.012
Mean : 1.000
3rd Qu.: 1.416
Max. : 2.023

> library(sp)
> coordinates(cluster) = ~X+Y

The following commands specify a sequence of lag boundaries that correspond to the GSLIB conventions, and compute a regular variogram using these boundaries:

> bnd = c(0,2.5,7.5,12.5,17.5,22.5,27.5,32.5,37.5,42.5,47.5,52.5)
> variogram(Primary~1, cluster, boundaries = bnd)

<table>
<thead>
<tr>
<th>np</th>
<th>dist</th>
<th>gamma</th>
<th>dir.hor</th>
<th>dir.ver</th>
<th>id</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>149</td>
<td>1.527974</td>
<td>58.07709</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>624</td>
<td>5.472649</td>
<td>54.09188</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>989</td>
<td>10.150607</td>
<td>48.86144</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>1249</td>
<td>15.112173</td>
<td>40.08909</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>1148</td>
<td>20.033244</td>
<td>42.45081</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>1367</td>
<td>25.020160</td>
<td>48.60365</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>7</td>
<td>1311</td>
<td>29.996102</td>
<td>46.88879</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>8</td>
<td>1085</td>
<td>34.907219</td>
<td>44.36890</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>9</td>
<td>904</td>
<td>39.876469</td>
<td>47.34666</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>10</td>
<td>611</td>
<td>44.716540</td>
<td>38.72725</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>11</td>
<td>219</td>
<td>49.387310</td>
<td>30.67908</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

To compute the relative pairwise variogram, the logical argument PR ("pairwise relative") needs to be set to TRUE:

> variogram(Primary~1, cluster, boundaries=bnd, PR = TRUE)

<table>
<thead>
<tr>
<th>np</th>
<th>dist</th>
<th>gamma</th>
<th>dir.hor</th>
<th>dir.ver</th>
<th>id</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>149</td>
<td>1.527974</td>
<td>0.3608431</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>624</td>
<td>5.472649</td>
<td>0.6307083</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>989</td>
<td>10.150607</td>
<td>0.8376443</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>1249</td>
<td>15.112173</td>
<td>0.7769083</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>1148</td>
<td>20.033244</td>
<td>0.8745999</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>1367</td>
<td>25.020160</td>
<td>0.8961016</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>7</td>
<td>1311</td>
<td>29.996102</td>
<td>0.902297</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>8</td>
<td>1085</td>
<td>34.907219</td>
<td>0.964305</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>9</td>
<td>904</td>
<td>39.876469</td>
<td>0.9055426</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>10</td>
<td>611</td>
<td>44.716540</td>
<td>0.7554474</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>11</td>
<td>219</td>
<td>49.387310</td>
<td>0.8226759</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Figure 1 shows the two variograms, as plots, side by side

2 Verification with plain R code

The following R code reproduces the relative pairwise semivariogram values for the first three lags, i.e. 0-2.5, 2.5-7.5 and 7.5-12.5.
Figure 1: Regular variogram (left) and pairwise relative variogram (right) for the GSLIB data set \texttt{cluster.dat}.
\[ z = \text{cluster}\$Primary \]
\[ d = \text{spDists}((\text{cluster})) \]
\[ zd = \text{outer}(z, z, "-") \]
\[ zs = \text{outer}(z, z, "+") \]
\[ pr = (2 * zd / zs)^2 \]
\[ prv = \text{as.vector}(pr) \]
\[ dv = \text{as.vector}(d) \]
\[ \text{mean}(prv[dv > 0 & dv < 2.5])/2 \]
\[ [1] \ 0.3608431 \]
\[ \text{mean}(prv[dv > 2.5 & dv < 7.5])/2 \]
\[ [1] \ 0.6307083 \]
\[ \text{mean}(prv[dv > 7.5 & dv < 12.5])/2 \]
\[ [1] \ 0.8376443 \]

### 3 Verification with GSLIB

In a verification with the GSLIB (Deutsch and Journel, 1997) code of gamv, the following file was used:

```
Parameters for GAMV
***************

START OF PARAMETERS:
./data/cluster.dat
file with data
1 2 0
columns for X, Y, Z coordinates
1 3
number of variables,column numbers
-1.0e21 1.0e21
trimming limits
gamv.out
file for variogram output
10
number of lags
5.0
lag separation distance
2.5
lag tolerance
1
number of directions
0.0 90.0 50.0 0.0 90.0 50.0
azm,atol,bandh,dip,dtol,bandv
0
standardize sills? (0=no, 1=yes)
```
number of variograms
1 1 1
tail var., head var., variogram type
1 1 6
tail var., head var., variogram type

Running this program with these parameters gave the following output:

As can be seen, the values in the third column (semivariogram for the first section, pairwise relative semivariogram for the second) correspond to the output generated by variogram of package gstat. Two differences with respect to the gstat output are:

- for the first lag with distance zero, GSLIB reports that the semivariance value is zero based on 280 point pairs;
- the number of point pairs in GSLIB is double the number reported by gstat.

The ground for these differences seems that the GSLIB gamv uses a single routine for computing variograms as well as cross variograms and cross covariances. For cross variograms or covariograms, considering two variables $Z_a$
and $Z_b$ each having $N$ observations, the $N^2$ point pairs $Z_a(s_i), Z_b(s_i + h)$ and $Z_a(s_i + h), Z_b(s_i)$ need to be evaluated, and all contribute information.

For direct (non-cross) variograms or covariograms, $Z_a = Z_b$ and the $N^2$ pairs considered contain the $N$ trivial pairs $(Z(s_i) - Z(s_i))^2 = 0$, which contribute no information, as well as all duplicate pairs, i.e. in addition to $(Z(s_i) - Z(s_i + h))^2$, the identical pair $(Z(s_i + h) - Z(s_i))^2$ is also considered. This leads to correct variogram value estimates, but incorrect unique point pair numbers. (Data set cluster contains $N = 140$ observations.)

In contrast, gstat considers (and reports) only the number of unique pairs for each lag.

References