



HAL
open science

Unsupervised classification of multivariate geostatistical data: Two algorithms

Thomas Romary, Fabien Ors, Jacques Rivoirard, Jacques Deraisme

► To cite this version:

Thomas Romary, Fabien Ors, Jacques Rivoirard, Jacques Deraisme. Unsupervised classification of multivariate geostatistical data: Two algorithms. *Computers & Geosciences*, 2015, Statistical learning in geoscience modelling: Novel algorithms and challenging case studies, 85, pp.96-103. 10.1016/j.cageo.2015.05.019 . hal-01219704

HAL Id: hal-01219704

<https://minesparis-psl.hal.science/hal-01219704v1>

Submitted on 23 Oct 2015

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

10 **1. Introduction**

11 In mining assessment, a partitioning of the data is often to be conducted
12 prior to evaluate the reserves. This is necessary to design the mineralization
13 envelopes corresponding to several lithofacies where the grades of the ore
14 to be mined may have different spatial behavior, in terms of mean, variability
15 and spatial structure. In remote sensing of environmental variables, a
16 similar problem may be encountered when the characteristics of the variable
17 of interest is governed by a hidden variable, e.g. the component of a mixture
18 model, accounting for a particular local behaviour. A typical example
19 in soil sciences consists in the the retrieval of soil classes over a region from
20 the observation of continuous variables.

21 A natural solution to this problem is to cluster the data. Clustering a
22 dataset consists in partitioning the observations into subsets (called clusters)
23 so that observations in the same cluster are similar in some sense.
24 Clustering is used in many fields, including machine learning, data mining,
25 pattern recognition, image analysis, information retrieval and bioinformatics
26 (Hastie et al., 2009). It is an unsupervised classification problem where the
27 goal is to determine a structure among the data, with no response variable
28 to lead the process.

29 While a wide range of methods exist for independent (Hastie et al., 2009) or
30 gridded spatial observations (in the image processing literature), not much
31 attention has been paid to the case of irregularly spaced data. Indeed, in
32 a geostatistical context, one expects to obtain a classification of the data
33 that presents some spatial continuity. This is especially the case with mining
34 data, where the geologist wishes to delineate homogeneous areas in a
35 deposit to facilitate its evaluation and exploitation.

36 Clustering in a spatial framework has been mainly studied in the image
37 analysis context where the data is organized on a grid. The model is usually
38 a hidden Markov random field. In this model, label properties and
39 pixel values need only to be conditioned on nearest neighbors instead of on
40 all pixels of the map, see e.g. Guyon (1995) for a review and Celeux et al.
41 (2003) for more recent developments. In Ambroise et al. (1995), the authors
42 proposed to use this approach directly to irregularly sampled data using a
43 neighborhood defined by the Delaunay graph of the data. As the length of
44 the edges of the graph are not accounted for in the approach, this neighborhood
45 structure does not reflect a structure in the data, rather a structure in

46 the sampling scheme. This disqualifies this approach especially for mining
47 data, where the samples are located along drillholes: two neighbors on a
48 same drillhole are distant a few centimeters while two neighbors from two
49 different drillholes may be distant several decimeters.

50 Oliver and Webster (Oliver and Webster, 1989) were the first to propose a
51 method for the clustering of multivariate non-lattice data. They proposed
52 to modify the dissimilarity matrix of the data, used e.g. in a hierarchical
53 algorithm, by multiplying it by a variogram matrix. This terms to smooth
54 the dissimilarity matrix for close pairs of points. However, this will not en-
55 force the connexity of the resulting clusters, it will rather blur the borders
56 between geologically different areas, making them difficult to differentiate,
57 as our practice showed.

58 In Allard and Guillot (2000), the authors proposed a clustering method
59 based on a mixture of random functions models where an approximation
60 of the expectation-maximization (EM, see Dempster et al., 1977) algorithm
61 is used to estimate the parameters and the labels. It has been later extended
62 to multivariate data in Guillot et al. (2006). However this method relies on
63 strong assumptions that are not likely to be encountered in practice: the
64 data are assumed to be Gaussian and data belonging to different clusters
65 are assumed independent. Moreover, the estimation algorithm requires the
66 computation of the maximum likelihood estimator of the random function
67 model at each iteration of the EM algorithm, which involves the inversion of
68 the covariance matrix and is not computationally tractable for large, mul-
69 tivariate datasets. Indeed, a single iteration requires several inversions of a
70 $(n \times p) \times (n \times p)$ matrix, where n is the number of data and p is the number of
71 variables. Using composite likelihood techniques (Varin et al., 2011) could
72 be useful to alleviate the computational burden but it will add a degree of
73 approximation while still not allowing to deal with categorical data.

74 The approaches developped in this paper are model free and do not involve
75 complex computations. Therefore, they are able to process large, multi-
76 variate datasets. The first one, already outlined in Romary et al. (2012),
77 is based on an agglomerative hierarchical algorithm with complete linkage
78 (see e.g. Hastie et al., 2009), where the connexity of the resulting clusters is
79 enforced through the use of a graph structuring the data. It only involves
80 the computation of distances along the edges of the graph which has a sparse
81 structure. Its sequential nature makes it practical for reasonable volumes of
82 data. An alternative for large datasets consists however in running first the
83 algorithm on a subsample, then training a supervised classifier and finally
84 applying it to the rest of the data. The second proposed algorithm provides
85 a non-hierarchical alternative to partition the same graph. It is an adap-

86 tation of the spectral clustering algorithm (Ng et al., 2002; von Luxburg,
87 2007) to geostatistical data. The computations involve only sparse matrices,
88 therefore this second algorithm is adapted to large volumes of data.
89 The paper is organized as follows: in section 2, we describe both algorithms
90 as well as a method to classify newly available data based on the results
91 of a preceding clustering. In section 3, we show the performance of each
92 algorithm on a synthetic dataset as well as on a mining dataset.

93 2. Algorithms

94 Both algorithms proposed rely on the same basic idea. The latter consists
95 in structuring the available data in a graph in the geographical space
96 made of a unique connex component. This graph is then partitioned into
97 clusters either hierarchically or directly by decomposition. The structure
98 thus imposed ensures the spatial coherency of the resulting clusters.
99 We consider a sample of georeferenced data $(x_1, \dots, x_n) \in \mathbb{R}^{n \times p}$, where p
100 is the number of variables, coordinates included. We also consider that the
101 data have been standardized preliminary to the application of the clustering
102 algorithms. It may also be useful to gaussianize the variables, e.g. by
103 anamorphosis (Chilès and Delfiner, 2012), for skewed data. This preliminary
104 processing allows to make the variables comparable. We describe in
105 this section the different ingredients required to implement both algorithms
106 as well as their core.

107 2.1. Structuring the data

Being either regular or not, the spatial sampling of a geostatistical dataset defines a geometric set, namely a set of points in the geographical space. From this set, a neighborhood system can be built. This can be represented by an undirected graph where each vertex represents an observation and each edge shows the relation of neighborhood shared by close points (Geman and Geman, 1984). We call this graph the sampling graph. Several methods can be applied to build it such as Delaunay triangulation, Gabriel graph or a graph based on the neighborhood structure defined by moving neighborhood algorithms used in kriging, for instance based on octants (see e.g. Chilès and Delfiner, 2012). Particular shapes can also be obtained by using non-isotropic distances or coordinates transformation. The graph should be parsimonious whilst containing enough edges to support a variety of configurations for the clusters. In our experience, the Delaunay graph and a graph based on a neighborhood selection algorithm give good results. Once the graph G has been built, two observations x_i and $x_j, i \neq j$, are said

to be connected if there exists an edge in G linking x_i and x_j . This is denoted by $x_i \leftrightarrow x_j$. G can also be represented by its adjacency matrix with general term $(G_{ij})_{i,j \in \{1, \dots, n\}}$:

$$G_{ij} = \begin{cases} 1 & \text{if } x_i \leftrightarrow x_j, \\ 0 & \text{otherwise.} \end{cases}$$

108 Note that an individual is not considered to be connected with itself.

109 *2.2. Choice of a distance*

The second basic ingredient of both algorithms is a distance or metric measuring the dissimilarity between two observations. The aim of clustering algorithms is to group similar observations, hence the need to define *similar*. We define the distance d between two observations x_i and x_j by:

$$d(x_i, x_j) = \sum_{k=1}^p \sum_{l=1}^p \omega_{k,l} d^{(k,l)}(x_i^{(k)}, x_j^{(l)}), \quad (1)$$

110 where $(\omega_{k,l})_{(k,l) \in (1, \dots, p)^2}$ are the entries of a positive definite matrix Ω and
 111 $(d^{(k,l)})_{k=1, \dots, p}$ is a set of coupled distances, each one chosen according to
 112 the corresponding couple of variables. d is therefore a weighted sum of
 113 distances. The weights are to be chosen by the user, depending on the
 114 relative importance the variables should have and their possible correlation.
 115 As noted above, the variables have been preliminary standardized so as to
 116 avoid any scale effect between the variables. In practice, Ω is generally
 117 chosen to be diagonal and only individual distances are thus involved. The
 118 use of the squared Mahalanobis distance, where Ω is the inverse of the
 119 empirical covariance matrix could be considered so as to account for possible
 120 correlations between variables, but has not proven useful in our experiments.
 121 The individual distances are chosen according to the associated variable: if
 122 the latter is quantitative, the squared euclidean distance is advocated from
 123 its strong relation with the variogram as a measure of the local continuity; if
 124 it is a categorical variable, an ad-hoc distance is used. Such a distance may
 125 take the value 0 when both observations have an equal value for this variable
 126 and 1 otherwise. Other options are also available, see e.g. Hastie et al.
 127 (2009) for a comprehensive view.

128 It is worth noting that the coordinates are also included in (1). Indeed,
 129 although the spatial location of the data is already accounted for by the
 130 graph structure, this allows to account for the length of the edges. By doing
 131 this, we promote short connections.

132 Concerning the setting up of the weights, we generally recommend to put
 133 5% to 30% of the total on the coordinates and to set the other variables
 134 to 1 at a first guess, then to progressively tune the weights of the variables
 135 according to the outcome of the algorithm.

136 2.3. Geostatistical Hierarchical Clustering

137 The distance defined above is only valid between pairs of observations.
 138 Agglomerative hierarchical clustering algorithms require a linkage criterion
 139 which specifies the dissimilarity of sets as a function of the pairwise distances
 140 of observations in the sets. Lance and Williams formula (Lance and Williams,
 141 1966) enables the use of a unique recurrence formula to update the distances
 142 when merging two clusters for a large family of criteria, including the maxi-
 143 mum, minimum or average distance, respectively named complete, single or
 144 average linkage criteria or Ward’s criterion which computes the intra-cluster
 145 variance, see e.g. Milligan (1979).

146 In our context, the spatial continuity needs to be taken into account during
 147 the linkage process. In the proposed algorithm, two clusters can merge if
 148 and only if they are connected in the graph structure G . When two clusters
 149 merge, the resulting cluster inherits all connections of its components. This
 150 point is the only departure from the original hierarchical clustering algo-
 151 rithm.

152 The geostatistical hierarchical clustering algorithm (GHC) is described in
 153 pseudo code in algorithm 1 under the complete linkage criterion.

Algorithm 1 Geostatistical Hierarchical Clustering algorithm (GHC)

- 1: Compute the distance matrix $D \in \mathbb{R}^{n \times n}$, such that $D_{ij} = d(x_i, x_j)$, $j < i$, if $i \leftrightarrow j$, 0 otherwise
- 2: **repeat**
- 3: Find k and l , $k < l$, such that $D_{lk} = \min_{\{i,j,i \leftrightarrow j\}} D_{ij}$
- 4: Merge k and l in $\{kl\}$, and update D such that

$$\begin{aligned}
 D_{ki} &= \max(D_{ki}, D_{li}) \text{ if } i \leftrightarrow \{kl\} \text{ and } i < k \\
 D_{ik} &= \max(D_{ik}, D_{li}) \text{ if } i \leftrightarrow \{kl\} \text{ and } k < i < l \\
 D_{ik} &= \max(D_{ik}, D_{il}) \text{ if } i \leftrightarrow \{kl\} \text{ and } i > l
 \end{aligned}$$

discard line and column l from D

- 5: **until** D is a scalar
-

154 In algorithm 1, the value D_{lk} can be interpreted as the inner distance
 155 or dissimilarity of the cluster obtained when merging clusters k and l . The

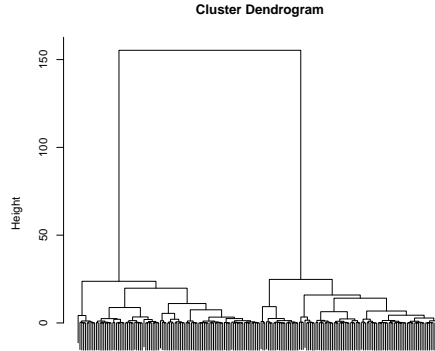


Figure 1: Example of a dendrogram

156 notation $i \leftrightarrow \{kl\}$ means i is connected with the cluster $\{kl\}$, that is $i \leftrightarrow k$
 157 or $i \leftrightarrow l$.

158 Since two clusters are merged when they realize the minimum distance
 159 among the connected pairs of clusters, they may not realize the minimum
 160 distance in absolute, depending on the chosen linkage criterion. In partic-
 161 ular, more dissimilar points may merge into clusters before having merged
 162 points which are actually more similar but not directly connected. That
 163 is why we advocate the use of the complete linkage criterion which is, to
 164 our knowledge, the only way to preserve the ultrametric property in our
 165 algorithm. The ultrametric property means a monotonic increase of the dis-
 166 similarity value of the clusters, see Milligan (1979) for further details. In
 167 particular, the ultrametric property allows to build a dendrogram.

168 The dendrogram is a very practical tool to select the final number of clusters,
 169 see an example in figure 1. It represents the evolution of the intra-cluster
 170 dissimilarity along the agglomeration process. A long branch means that the
 171 merge between two clusters leads to a much less homogeneous one. There-
 172 fore the tree should be pruned at the level where the branches are long. The
 173 number of pruned branches gives the number of clusters to consider, 2 in
 174 the example of figure 1.

175 The computational efficiency of this algorithm relies on the graph struc-
 176 ture employed and especially on the number of connections. Indeed, only
 177 the distances between connected points are required at the beginning of the

178 algorithm, which makes the matrix D sparse and allows fast computations.
 179 Then, the computation of the distances between connected points required
 180 at step 4 can be performed on the fly.

181 2.4. Geostatistical Spectral Clustering

182 The Geostatistical Spectral Clustering (GSC) is an adaptation of the
 183 spectral clustering algorithm where the graph used is the sampling graph
 184 defined above instead of a graph based on the similarity. Contrarily to GSC,
 185 it requires a preselection of the number K of desired clusters and does not
 186 rely on an iterative procedure. This is not a major drawback however. Once
 187 computed the quantities required for a given maximum number of classes, it
 188 is straightforward to compute the outcome for a smaller number of classes.
 189 The different steps of the algorithm are described in algorithm 2.

Algorithm 2 Geostatistical Spectral Clustering algorithm (GSC)

1: Compute the similarity or weighted adjacency matrix W :

$$W_{ij} = \begin{cases} \exp\left(-\frac{d(x_i, x_j)}{\sigma^2}\right) & \text{if } i \leftrightarrow j \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

2: Compute the degree matrix D :

$$D_{ii} = \sum_{j=1}^n W_{ij}$$

3: Compute the graph Laplacian matrix

$$L = D^{-1/2} W D^{-1/2}$$

4: Compute the K largest eigenvalues of L and form the matrix $V \in \mathbb{R}^{n \times K}$
 whose columns are the associated K first eigenvectors of L

5: Apply the K -means algorithm to the lines of V

6: Assign observation x_i to the same class the line i of V has been assigned

190 This algorithm consists in representing the data into an infinite dimen-
 191 sional space (the reproducing kernel Hilbert space associated to the kernel
 192 used in (2), here the Gaussian (or radial basis function kernel) where they
 193 are easily clustered through K -means.

194 The parameter σ^2 is chosen as the empirical variance of the variable, fol-
 195 lowing von Luxburg (2007). Note that a local adaptive approach could be
 196 considered for the setting of σ^2 , as proposed in Zelnik-Manor and Perona
 197 (2004). However, this refinement has not proven useful in our practice.
 198 Also, the lines of V can be optionally normalized prior to step 5, as pro-
 199 posed in Ng et al. (2002). The differences when using the normalization or
 200 not did not appear sensible in our experimentations.

201 The number of clusters to consider can be chosen by studying the eigen-
 202 values of L . A small eigenvalue signifies that the associated eigenvector is
 203 not relevant to discriminates the data. In practice, we advocate to compute
 204 a given maximum number of eigenvalues (10 to 20), which corresponds to
 205 the maximum number of clusters we want, and then to plot them. A large
 206 difference between two eigenvalues means that the smaller one is not so rel-
 207 evant.

208 As the graph structure is sparse, all the computations required in algorithm
 209 2 can be carried out using sparse linear matrix algebra, which makes GSC
 210 computationally efficient and adapted to large multivariate datasets.

211 2.5. *Classifying new data*

212 Sometimes, the sampling of the variables of interest on a domain can be
 213 performed in several steps. For instance, new drillholes can be added to an
 214 initial sampling campaign. In the case where a clustering has already been
 215 performed, we may want to classify the new data into the classes resulting
 216 from that previous run. An other occurrence when we want to classify data
 217 upon the results of a previous clustering is when dealing with very large
 218 datasets with the GHC. In that case, we propose to run first the algorithm
 219 on a subsample, then train a supervised classifier and finally apply the latter
 220 to the remaining data.

221 It is particularly difficult to incorporate new data into the clustering results
 222 with simple rules. Indeed, when new data are added, the sampling graph
 223 gets modified and the outcome of GHC and GSC may change dramatically.
 224 Therefore, the idea developed here is to learn a classification rule based
 225 on the initial clustering results. This can be achieved for instance through
 226 support vector machines (SVM, see Hastie et al. (2009)). In the case of two
 227 classes, the basic principle is to find $f(x) = \alpha_0 + \sum_{i=1}^N \alpha_i \Phi(x, x_i)$, where the
 228 $(\alpha_i)_{i=0, \dots, N}$ are scalars and Φ a given kernel function, that minimizes

$$\sum_{i=1}^N (1 - y_i f(x_i))_+ + \lambda \alpha^t \Phi \alpha, \quad (3)$$

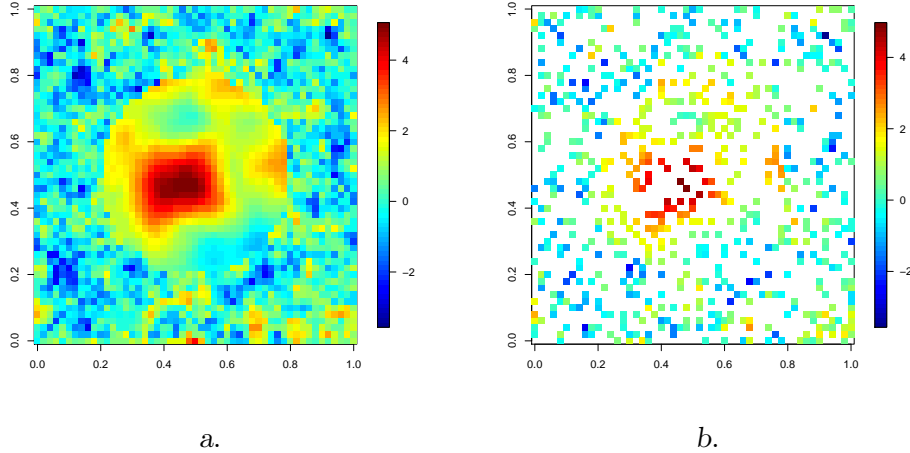


Figure 2: One realization of the random function *a.* and sampling performed *b.*

229 as a function of $(\alpha_i)_{i=0,\dots,N}$ and where the underscript $+$ means the maxi-
 230 mum between 0 and the quantity between parenthesis, and λ is a penalty
 231 parameter. For multi-class classification, several options are available among
 232 which we retain the standard “one versus all” implemented in LIBSVM
 233 (Chang and Lin, 2011). The penalty parameter λ is set through cross-
 234 validation. Applying the rule to a new observation allows to assign it to
 235 an existing class.

236 3. Results

237 3.1. Toy dataset

238 Here, we describe a 2D example on which we have evaluated the perfor-
 239 mances of several methods including GHC and GSC. We consider a random
 240 function on the unit square which is made of a Gaussian random function
 241 with mean 2 and a cubic covariance with range 0.3 and sill 1 on the disk of
 242 radius 0.3 and center (0.5,0.5) and a Gaussian random function with mean
 243 0 and an exponential covariance with range 0.1 and sill 1 elsewhere. This
 244 model is made to mimick a mineralization area in a mining deposit, where
 245 high grades are more likely to be found within the disk. A realization is
 246 shown in figure 2 *a.* while figure 2 *b.* corresponds to the sampling performed
 247 by picking 650 points out of the 2601 points of the complete realization.

248 We can clearly see a smooth surface with high values in the central disk
 249 in figure 2 a. and this is the area we would like to retrieve from the 650 ob-
 250 servations plotted in figure 2 b.. We test the performances of five different
 251 methods for this task: K -means, complete linkage hierarchical clustering
 252 (HC), Oliver and Webster’s method (O&W), GHC and GSC.
 253 For every method, the three variables are scaled such that the coordinates
 254 are given a weight of 10% in the computation of the distance. This prelim-
 255 inary treatment makes the different methods comparable. In HC, O&W,
 256 GHC and GSC, we use the squared euclidean distance. K -means does not
 257 need any parameterization. For O&W, several variogram models and sets
 258 of parameters have been considered, without much success. The results pre-
 259 sented here are obtained with an exponential variogram with range 0.5. The
 260 Delaunay graph has been used for both GHC and GSC. Concerning GSC,
 261 normalizing the rows of V (see algorithm 2) gave similar results as without
 262 normalization. Consequently, only the results without normalization are
 263 presented.

264 Figures 3 and 4 show the results obtained by each five methods on the
 265 realization depicted in figure 2. Each subpicture represents the dataset on
 266 scatterplots with respect to the coordinates (x and y) and the sampled value
 267 (Z). K -means (a.) identifies well the central area. The result lacks of con-
 268 nectivity however. In particular, large values outside of the disk are classified
 269 as belonging to the disk and low values within the disk are missclassified as
 270 well. It can be seen that the method only discriminates between low and
 271 high values of Z : the limiting value between the two clusters can be read
 272 as more or less 0.5. HC (b.) also discriminates between low and high value
 273 but the limiting value is higher, around 2. To sum up, those two classical
 274 methods in an independent observations context fail to produce spatially
 275 connected clusters. O & W’s approach has been tested with various vari-
 276 ograms and variogram parameter values but it never showed any structured
 277 result (c.). Our interpretation is that multiplying the dissimilarity matrix
 278 by a variogram may erase some dissimilarities, inducing a loss in the struc-
 279 ture of the data. The GHC algorithm succeeds in providing a clustering
 280 with spatial connectivity (d.) though non perfect. A part of the area sur-
 281 rounding the disk is misclassified however. If we turn back to the complete
 282 realization in figure 2 a, we can see that the misclassified area corresponds
 283 to high values of the realization around the border of the disk that are very
 284 close to the values taken inside the disk and are thus difficult to classify
 285 correctly. Finally, the GSC algorithm performed a geometrical classification
 286 by making a cut along the axis of the first coordinate, see figure 4. However,
 287 when looking at the result obtained when asking for five classes, it provided

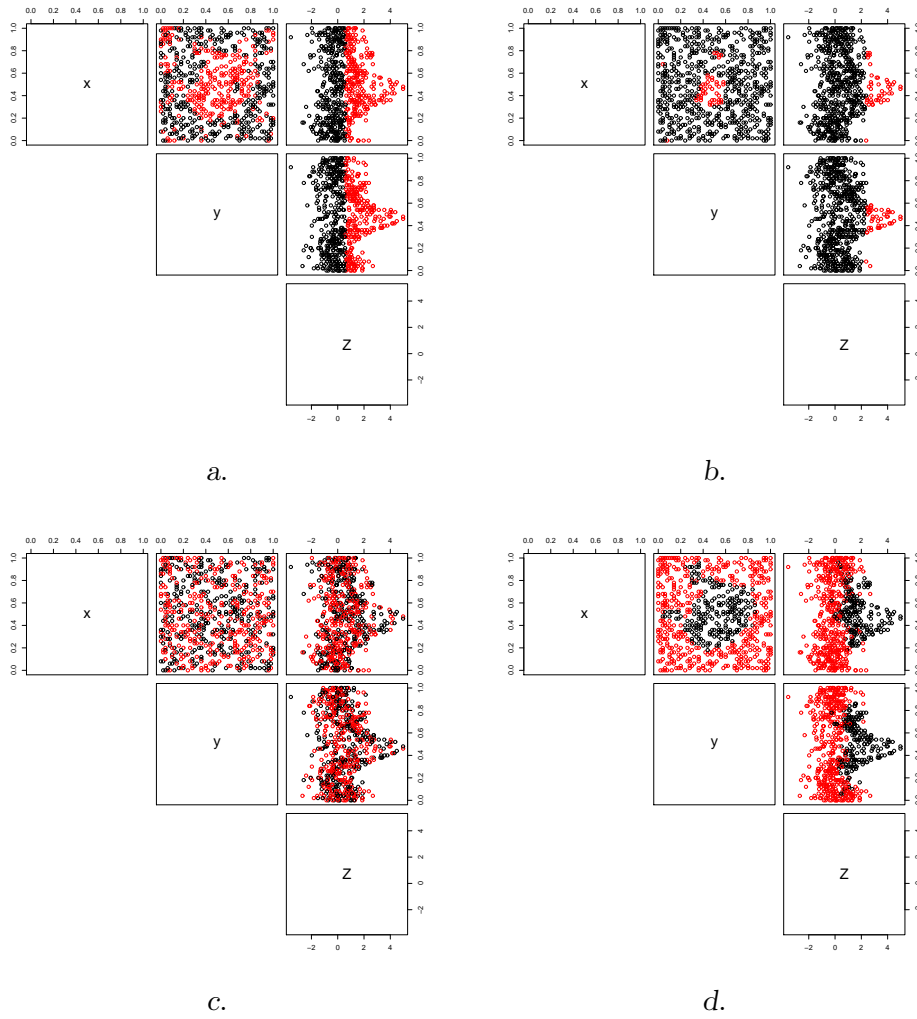


Figure 3: Results of K -means *a.*, hierarchical clustering *b.*, Oliver and Webster's method *c.* and geostatistical hierarchical clustering *d.*

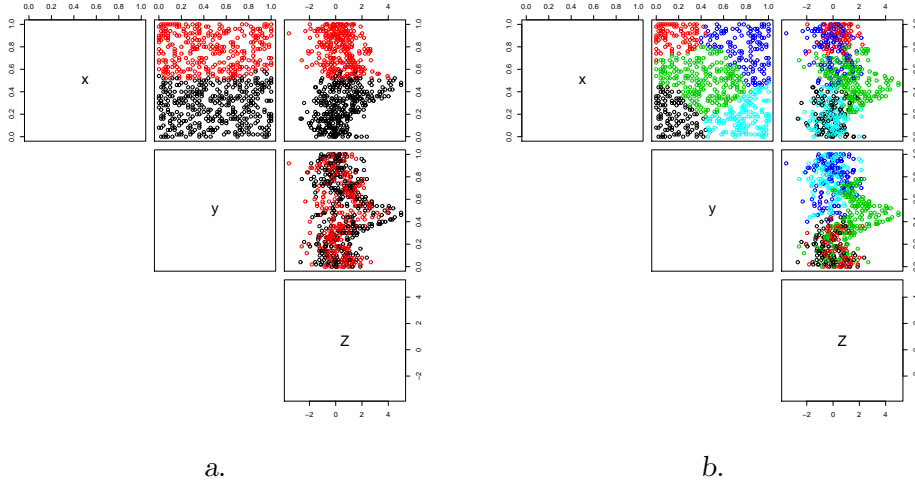


Figure 4: Results of GSC for two *a.* and five classes *b.*

288 a class delineating the disk fairly well. It seems that this algorithm tends to
 289 generate more compact subsets of the sampling graph.
 290 Each of the five algorithms are applied to 100 realizations of the same ran-
 291 dom function model, each with a different uniform random sampling. Then
 292 we compute the mean, median and 90% percentile of the rate of correctly
 293 classified points. Results are summarized in table 1.
 294 GHC exhibits the best performances overall with 85% correctly classified
 295 points in average while *K*-means providing similar results in average, GSC
 296 performing the worst with HC and O & W in between. If we look at the
 297 median however, GHC has the greatest one with a larger margin. The 90%
 298 percentile indicates that in the 10% most favorables cases, GHC misclassi-
 299 fied only 0.02% of the points, while all the other algorithms perform worse.
 300 It can also be seen that the 90% percentile are similar for the *K*-means and
 301 the HC. This means that the HC, and GHC (its worse result in this task
 302 was a misclassification of almost 50%, seemingly due to a high sensitivity
 303 to large values), can sometimes perform really bad, whereas the *K*-means
 304 algorithm gives more stable results, being less sensitive to extreme values.
 305 Indeed, in the presence of very large or very low value, it occurs that the
 306 algorithm comes out with a class made of a single point while the other
 307 contains all the other observations. In the favorable cases however, HC al-
 308 gorithm works as well as the *K*-means, while GHC outperforms clearly all
 309 other algorithms. Concerning GSC, the results obtained are extremely poor

	<i>K</i> -means	HC	O & W	GHC	GSC
Mean	0.86	0.70	0.65	0.85	0.52
Median	0.86	0.64	0.67	0.90	0.52
90% percentile	0.90	0.91	0.72	0.98	0.54

Table 1: Rates of correctly classified points for the 5 algorithms

310 but do not account for the interesting results obtained when considering
311 more classes.

312 It is worth noting that the drawbacks exhibited by GHC and GSC are far
313 from being prohibitive in practice. Indeed, when applying clustering al-
314 gorithms to real data the user generally observes the outcome for several
315 numbers of classes. This can be performed easily with both algorithms with
316 a negligible computational cost.

317 *3.2. Mining data example*

318 In this section, we present an application of both geostatistical clustering
319 algorithms to an ore deposit. We describe the different steps and exhibit
320 some results.

321 The first step is to select the data that will be used for the clustering. The
322 following variables are chosen:

- 323 • coordinates, X , Y and Z ,
- 324 • ore grades,
- 325 • a geological factor describing the basement vs. a sandy part on top of
326 it,
- 327 • the hematization degree.

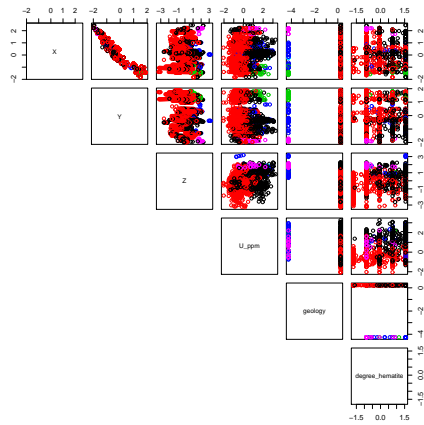
328 This choice is made upon an exploratory analysis of the data and discus-
329 sions with geologists. Some transformations of the data are preliminary
330 performed:

- 331 • coordinates are standardized,
- 332 • ore grades are log-transformed and standardized,
- 333 • the degree of hematization is transformed into a continuous variable,
334 then standardized.

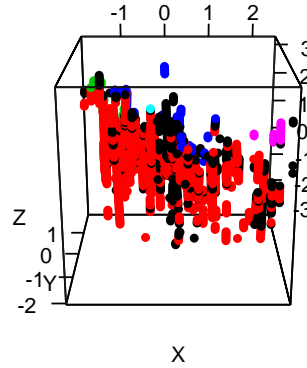
335 The next step consists of building the sampling graph connections between
336 geographically close samples. The graph is here built from the neighbour-
337 ing structure induced by the moving neighbourhood kriging algorithm of
338 Isatis® 2013 (Geostatistics, 2013). At each point, the space is split into 16
339 hexadecants: 8 above and 8 below the horizon. One neighbor per hexade-
340 cant is authorized at most for each point with no more than 2 from the same
341 drillhole. The search ellipse is of infinite size so as to connect even possibly
342 distant points. The angles of the search ellipse are chosen so that to take
343 into account the horizontal shape of the mineralization of the deposit.
344 Then the dissimilarity matrix is built. All variables listed above are used.
345 A particular distance for the geological factor is considered: it is chosen to
346 be 1 when the samples have different factor values and 0 otherwise. This
347 distance is scaled to maintain the coherency with the other individual dis-
348 tances. Weights are set step by step, as advocated in section 2.2: we begin
349 by giving an equal weight to all variables with a 30% contribution to the
350 coordinates. Finally, the contribution of the coordinates is lowered to 10%
351 while the other variables are assigned equal weights. The same set of weights
352 is used for both algorithms. Practice shows indeed that setting low weights
353 to the coordinates leads to better results, as the spatial aspect is already
354 somehow taken into account by the sampling graph. However, the coordi-
355 nates needs to be included in the distance so as to account for different
356 length of the edges in the graph. This is especially important for drillholes
357 data where two neighbors along a drillhole are generally much closer than
358 two neighbors belonging to two different drillholes.
359 Finally, we can run both GHC and GSC algorithms described in section 2.
360 We choose to represent 6 clusters as the intra cluster dissimilarity at that
361 step of the GHC shows a great increase. The results are depicted in figure
362 5 for GHC and 6 for GSC.

363 GHC separates the basement into two classes, the black one being richer
364 than the red one. Note that the black cluster is mainly present in the mid-
365 dle of the deposit. The sandy part on top of the basement is splitted into
366 3 separate classes plus one single observation (in cyan), see figure 5. The
367 discrimination between the three sandy classes seems to rely on geographical
368 considerations.

369 As for GSC, it splits the basement into 5 classes and puts every observation
370 on top of it into one single class. Some similarities can be observed between
371 the clustering results obtained with the two algorithms however. In particu-
372 lar, both make a clear distinction between the basement and the sand on top
373 of it, emphasizing the variable 'geology'. They also both exhibit the desired
374 connexity properties. Both also reveal a high grades area in the center of

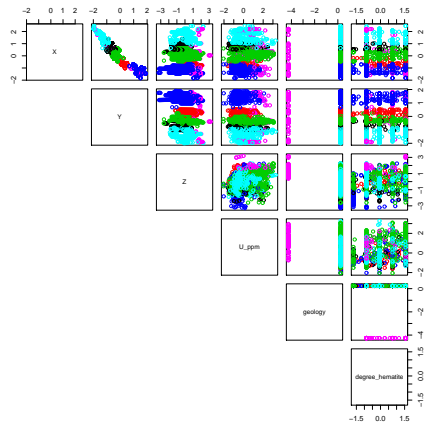


a.

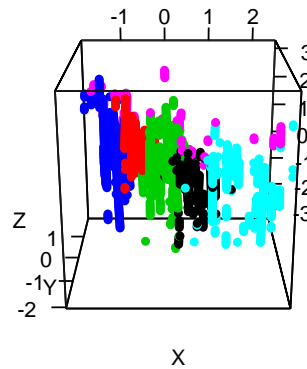


b.

Figure 5: Resulting clusters for the GHC algorithm from the variables point of view a. and in 3D b.



a.



b.

Figure 6: Resulting clusters for the GSC algorithm from the variables point of view a. and in 3D b.

375 the deposit (the black cluster in both figures), whose retrieval was the goal
376 of the experimentation. As already noticed in the previous paragraph, GSC
377 tends to produce more compact clusters than GHC who can follow awkward
378 routes along the graph.

379 **4. Conclusion**

380 In this paper, we presented two clustering procedures adapted to irreg-
381 ularly sampled spatial data. Both algorithms allow to process large multi-
382 variate datasets. They rely on a partition of a graph structuring the data in
383 the geographical space, thus ensuring the spatial coherency of the resulting
384 clusters. Two applications have been provided, the first one on a toy exam-
385 ple and the second on mining data.

386 The results shown on the toy example validate both algorithms as they are
387 able to produce compact, connected clusters. The results obtained for the
388 mining application are also satisfactory as they highlight a homogeneous
389 area with high grades. Thanks to the sequential nature of GHC, it gen-
390 erates a whole ensemble of coherent clusterings that can be useful to the
391 user: he can visualize the results at different hierarchical levels which helps
392 the interpretation and the choice of the final number of clusters for the end
393 user. Note that GSC does not enjoy this property as the results may change
394 dramatically from one desired number of clusters to another. The main
395 drawback of GHC is its limitation to datasets of reasonable size. It becomes
396 slow when the number of observations goes beyond 10000. In the case of
397 large datasets, a two step approach based on subsampling and supervised
398 classification is proposed.

399 Finally, setting the distance used to compute the graph and the weights as-
400 sociated to each variable allows the practitioner to get different clusterings,
401 according to its knowledge of the geology and the variables he wants to be
402 emphasized in the results. The main difficulty in handling these algorithms
403 is their sensitivity to the different parameters used. Moreover the results are
404 difficult to validate except from the computation of indices of compactness
405 of the clusters or of heterogeneity between them. They are mostly to be
406 validated by the eye of the practitioner whose knowledge of the data should
407 guide in the step by step parameterization of the approach.

408 **Acknowledgement**

409 This work has been partly funded by the Geological and Geostatistical
410 Domaining (G²DC) consortium, conducted by Geovariances. The authors

411 are grateful to all members of the consortium for helpful discussions, namely
412 AngloGold Ashanti, Areva, BHP Billiton, Eramet and Vale.

413 **References**

414 Allard, D., Guillot, G., 2000. Clustering geostatistical data. In: Proceedings
415 of the sixth geostatistical conference.

416 Ambroise, C., Dang, M., Govaert, G., 1995. Clustering of spatial data by
417 the EM algorithm. In: et al., A. S. (Ed.), *geoENV I - Geostatistics for*
418 *Environmental Applications*. Kluwer Academic Publishers, pp. 493–504.

419 Celeux, G., Forbes, F., Peyrard, N., 2003. Em procedures using mean field-
420 like approximations for markov model-based image segmentation. *Pattern*
421 *recognition* 36 (1), 131–144.

422 Chang, C.-C., Lin, C.-J., 2011. LIBSVM: A library for sup-
423 port vector machines. *ACM Transactions on Intelligent Sys-*
424 *tems and Technology* 2, 27:1–27:27, software available at
425 <http://www.csie.ntu.edu.tw/~cjlin/libsvm>.

426 Chilès, J. P., Delfiner, P., 2012. *Geostatistics, Modeling Spatial Uncertainty*,
427 2nd Edition. John Wiley & Sons, New-York.

428 Dempster, A. P., Laird, N. M., Rubin, D. B., 1977. Maximum likelihood
429 from incomplete data via EM algorithm (with discussion). *Journal of the*
430 *Royal Statistical Society, Ser. B* 39, 1–38.

431 Geman, S., Geman, D., 1984. Stochastic relaxation, gibbs distributions,
432 and the bayesian restoration of images. *Pattern Analysis and Machine*
433 *Intelligence, IEEE Transactions on* (6), 721–741.

434 Geovariances, 2013. *Isatis technical references*. Version 13.

435 Guillot, G., Kan-King-Yu, D., Michelin, J., Huet, P., 2006. Inference of
436 a hidden spatial tessellation from multivariate data: application to the
437 delineation of homogeneous regions in an agricultural field. *Journal of the*
438 *Royal Statistical Society: Series C (Applied Statistics)* 55 (3), 407–430.

439 Guyon, X., 1995. *Random fields on a network*. Springer.

440 Hastie, T., Tibshirani, R., Friedman, J., 2009. *The elements of statistical*
441 *learning*, 2nd Edition. Springer.

- 442 Lance, G., Williams, W., 1966. A generalized sorting strategy for computer
443 classifications. *Nature*.
- 444 Milligan, G. W., 1979. Ultrametric hierarchical clustering algorithms. *Psy-*
445 *chometrika* 44 (3), 343–346.
- 446 Ng, A., Jordan, M., Weiss, Y., 2002. On spectral clustering: analysis and
447 an algorithm. In: Dietterich, T., Becker, S., Ghahramani, Z. (Eds.), *Ad-*
448 *vances in Neural Information Processing Systems*. Vol. 14. MIT Press, p.
449 849 – 856.
- 450 Oliver, M., Webster, R., 1989. A geostatistical basis for spatial weight-
451 ing in multivariate classification. *Mathematical Geology* 21, 15–35,
452 10.1007/BF00897238.
453 URL <http://dx.doi.org/10.1007/BF00897238>
- 454 Romary, T., Rivoirard, J., Deraisme, J., Quinones, C., Freulon, X., 2012.
455 Domaining by clustering multivariate geostatistical data. In: *Geostatistics*
456 *Oslo 2012*. Springer, pp. 455–466.
- 457 Varin, C., Reid, N. M., Firth, D., 2011. An overview of composite likelihood
458 methods. *Statistica Sinica* 21 (1), 5–42.
- 459 von Luxburg, U., 2007. A tutorial on spectral clustering. *Statistics and Com-*
460 *puting* 17 (4).
- 461 Zelnik-Manor, L., Perona, P., 2004. Self-tuning spectral clustering. In: *Ad-*
462 *vances in Neural Information Processing Systems* 17. MIT Press, pp.
463 1601–1608.