CONDITIONAL GAUSSIAN MIXTURE MODELS FOR ENVIRONMENTAL RISK MAPPING

Nicolas Gilardi, Samy Bengio and Mikhail Kanevski
Dalle Molle Institute for Perceptual Artificial Intelligence
CP 592, rue du Simplon 4
1920 Martigny, Switzerland
E-mail: {gilardi,bengio,kanevski}@idiap.ch

Abstract. This paper proposes the use of Gaussian Mixture Models to estimate conditional probability density functions in an environmental risk mapping context. A conditional Gaussian Mixture Model has been compared to the geostatistical method of Sequential Gaussian Simulations and shows good performances in reconstructing local PDF. The data sets used for this comparison are parts of the digital elevation model of Switzerland.

INTRODUCTION

Environmental survey needs very reliable tools in order to facilitate decision making. An important category of these tools is called "Risk Maps". It consists of drawing various kinds of probability maps, such as "indicator maps" (probability of exceeding a threshold), the "value at risk" (quantile map), etc.

These problems can be solved using classical regression models such as K-Nearest Neighbors, Inverse Distance, Indicator Kriging, Artificial Neural Networks, etc. However, it is known that regression models based on minimization of the expected error have a smoothing effect and do not recover the variability of data. In the case of risk mapping, this smoothing effect is not acceptable as we are especially interested in unusual events, i.e. events that are not necessarily extreme but often far from the mean value. It is thus necessary to develop alternate prediction methods which could concentrate on reconstructing not only the mean but also the variability and eventually the whole distribution of the data.

In Geostatistics, Stochastic Simulations [5] were developed to solve these particular problems. However, these methods have some drawbacks. The modelization process is usually very complicated and necessitates a strong expert knowledge; they are often based on some assumptions about data distribution (stationarity, normality, ...); they do not provide any analytical model of the local distribution of a sample point which could be reused for other tasks.

In this paper, we propose a method that can estimate the local probability density function (PDF) for each data point, without making any assumption on the distribution of the data. It is based on the use of Gaussian Mixture Models (GMM) for conditional density estimation, by conditioning a global PDF model on the sample location.

To evaluate the relative performance of this method, we compare it to the well-known Geostatistical method of Sequential Gaussian Simulations (SGS).

In the following, we first present the principles of conditional GMM and SGS algorithms. We then describe the methodology used to build, use and compare the models during the experiments. Finally, we present the experiments themselves, the results and some conclusions on the efficiency of conditional GMM for local PDF estimation.

ALGORITHMS DESCRIPTION

Gaussian Mixture Models

Gaussian Mixture Models have the property of being able to represent any distribution as long as the number of Gaussians in the mixture is large enough. The PDF of a vector $\mathbf{v}$ can be modeled as:

$$p(\mathbf{v}) = \sum_{i=1}^{n} w_i \cdot \mathcal{N}(\mathbf{v}, \mu_i, \Sigma_i) \tag{1}$$

where w_i, μ_i and Σ_i are respectively the weight, the mean vector and the covariance matrix of the i^{th} of the n Gaussians of the model. All w_i are positive and sum to 1.

In the present study, we are interested in modeling the distribution $p(y|\mathbf{x})$ of a variable y given its position $\mathbf{x}$. An interesting solution is to use a neural network with inputs $\mathbf{x}$ and which outputs the parameters of a mixture of Gaussians on y, the whole thing being optimized by gradient ascent [6][1]. However such solution, while appealing, often does not work in practice as it suffers from initialization problems: if the Gaussians are not properly initialized, the learning algorithm is often stuck in poor local optima, and with such solution, we only control the parameters of the neural network. Hence, an other solution is to use the definition:

$$p(y|\mathbf{x}) = \frac{p(y, \mathbf{x})}{p(\mathbf{x})}. \tag{2}$$

The method developped for these experiments was found to be similar to the Distorted Probability Mixture Network described in [6]. The idea is to use the property of diagonal GMM [1] allowing to write:

[1]i.e. a GMM where the covariance matrix of each Gaussian is diagonal. Hence, for each Gaussian: $p(\mathbf{v}) = \prod_i p(v_i)$

$$p(y, \mathbf{x}) = \sum_{i=1}^{n} w_i \mathcal{N}(y, \mu_{yi}, \sigma_{yi}) \mathcal{N}(\mathbf{x}, \mu_{\mathbf{x}i}, \Sigma_{\mathbf{x}i}). \tag{3}$$

It is then possible to derive $p(\mathbf{x})$ from this model, simply by "removing" the contribution of y to the model. The expression $p(y|\mathbf{x})$ then becomes:

$$p(y|\mathbf{x}) = \sum_{i=1}^{n} W_i(\mathbf{x}) \mathcal{N}(y, \mu_{yi}, \sigma_{yi}) \tag{4}$$

with:

$$W_i(\mathbf{x}) = \frac{w_i \mathcal{N}(\mathbf{x}, \mu_{\mathbf{x}i}, \Sigma_{\mathbf{x}i})}{\sum_{j=1}^{n} w_j \mathcal{N}(\mathbf{x}, \mu_{\mathbf{x}j}, \Sigma_{\mathbf{x}j})}. \tag{5}$$

Sequential Gaussian Simulations

The idea of stochastic simulations is to develop a spatial Monte Carlo generator that will be able to generate many, and in some sense equally probable, realizations of a random function (in general, described by a joint probability density function).

Simulations differ from regression models as reconstruction of the histogram and of the spatial variability of original data takes precedence over local accuracy.

In the present study, SGS were applied. This method consists of generating values corresponding to given spatial locations, using a modelization of the spatial correlation (also called *variogram* model in Geostatistics) of a normally distributed known data set. The experimental variogram γ is first constructed using the formula:

$$\gamma(\mathbf{h}) = \frac{1}{2N(\mathbf{h})} \sum_{i=1}^{N(\mathbf{h})} (y(\mathbf{x}) - y(\mathbf{x} + \mathbf{h}))^2 \tag{6}$$

where $\mathbf{h}$ is the vector of the direction in which the correlation is measured, and $N(\mathbf{h})$ is the number of pairs of points $(\mathbf{x}_1, \mathbf{x}_2)$ such that $\overrightarrow{\mathbf{x}_1\mathbf{x}_2} = \mathbf{h}$. $\mathbf{h}$ usually has a user-defined tolerance in norm and direction. If its direction tolerance is 90°, the variogram is *omni-directional*. Given this experimental variogram, which is of course not a continuous function, we still need to model it using various continuous functions. One of the most commonly used is the spherical model, whose formula for a fixed direction of $\mathbf{h}$ is:

$$\hat{\gamma}(h) = \begin{cases} \frac{3h}{2a} - \frac{1}{2}\left(\frac{h}{a}\right)^3 & 0 \leq h \leq a \\ 1 & h > a \end{cases}$$

where h is the norm of the vector $\mathbf{h}$, and a is called the "range" of the variogram in the studied direction, i.e. the distance beyond which there is no more spatial correlation (in case of stationary data).

The variogram model is then used to compute the weights of a linear regression method called Kriging [2] (similar to Gaussian Processes [7]) which is the best linear unbiased estimator. It allows not only to estimate the value of new datum but also to compute the variance of this estimation.

Each simulated value is then generated from a normal distribution whose mean and variance are computed by applying Kriging on the neighboring (original *and* previously simulated) data points, based on the global variogram model.

METHODOLOGY

In the experiments presented in this paper, the data set is segmented into three parts. The first part is the training set, defined as

$$\mathcal{Y} = (\mathbf{x}_i, y_i), \quad \forall i = 1, \ldots, N \tag{7}$$

where $\mathbf{x}$ is the input vector (which represents the coordinates of the sample on a map), and y is the scalar output (studied value). The second part is the testing set, defined as

$$\mathcal{V} = (\mathbf{u}_i, v_i), \quad \forall i = 1, \ldots, M \tag{8}$$

where $\mathbf{u}$ is the input vector, and the output v is hidden to the models. The third part, which contains at least ten times more points than the training and testing sets, is called *reference* set. It will be used to compute the reference cumulative distribution of each point of the testing set.

The training set is used to tune the model's parameters and hyper-parameters as it will be explained for each method in the following subsections.

The tuned models are then used to build a cumulative distribution function for each point of the testing set. These cumulative distributions are then compared to those based on the reference set[2]. The quantitative performance of the models is evaluated on this last comparison.

GMM Experimental Protocol

In order to train a conditional GMM, one first need to select some hyper-parameters, such as the number of Gaussians, the relative variance lower bound in each dimension, and the Dirichlet prior on the weights of each Gaussians.

The initial position of the Gaussians must also be chosen. After several empirical experiments, we decided to initialize the GMM with one Gaussian per training point. The mean vector of each Gaussian was initially set to the position of the associated training point in the input space. Afterward,

[2]As the reference set is very large, a non-parametric method, detailed later, can be used to estimate reliably the distribution.

the GMM was trained using the Expectation-Maximization (EM) algorithm [3]. At the end of the training procedure, the Gaussians which were not contributing to the model (i.e. whose weights were close to 0) were removed.

The choice of the other hyper-parameters is done by k-fold cross-validation. Various criteria, measured on validation data, were tested to select efficiently the optimal set of hyper-parameters, such as the maximization of the likelihood, the lowest prediction error and the best reconstruction of centered moments. Finally, maximization of likelihood appeared to be the most efficient criterion.

SGS Experimental Protocol

SGS can only be used on normally distributed data. As a consequence, if this is not the case for original data, a Normal Score transformation[5] is needed. This transformation consists of the function NS : $F_{\mathcal{Y}} \to \mathcal{N}(0,1)$, where $F_{\mathcal{Y}}(y)$ is the cumulative distribution function of y in $\mathcal{Y}$. The crucial part is then to model the spatial correlation (i.e. the variogram). It can be difficult (or even impossible) to fit the variogram's shape with an appropriate model, depending on the studied phenomenon, the spatial repartition of data, the number of points, etc. In the present experiments, variograms were modeled using the classical spherical model presented before.

A simulation procedure starts by defining a random path visiting each location $\mathbf{u}$ of $\mathcal{V}$ once. Then, the simulated values are obtained by kriging of the neighboring training and previously simulated data. Afterward, they are back-transformed using $\mathrm{NS}^{-1} : \mathcal{N}(0,1) \to F_{\mathcal{Y}}$.

After a given number of simulations of the whole testing set (usually at least 100), the cumulative distribution at each point is estimated using a cumulative histogram.

Model Comparison Method

Comparing local PDF models is very difficult when there is only one realization of the studied phenomenon. In order to solve this problem, we used large data sets (thousands of samples) from which we only kept a small portion for training and testing (a typical training set in Geostatistics contains a few hundreds of samples). Remaining data were used to build a reference cumulative distribution function $C_{ref}(v|\mathbf{u})$ at every location $\mathbf{u}$ of the testing set.

$C_{ref}(v|\mathbf{u})$ is constructed by selecting the k nearest neighbors of each testing location, taken from the reference set and compute a cumulative histogram as it has been done for simulations. To define how many neighbors have to be taken, we simply divided the number of points in the reference set by the number of testing points. With such an approach, and providing that the testing locations are not clustered, one can consider that most of the k nearest neighbors of a testing point are only associated with this point. Of course, this cumulative histogram is only an approximation of the true CDF.

However it is the best approximation one can expect without any a priori knowledge of the data.

To measure the quality of a conditional PDF estimator, we proceed as follows:

- construct the conditional PDF estimator,
- estimate the cumulative $C_{model}(v|\mathbf{u})$ at every location $\mathbf{u}$,
- compute the D-Statistic, which is the greatest discrepancy between $C_{ref}(v|\mathbf{u})$ and $C_{model}(v|\mathbf{u})$ for each $\mathbf{u}$, as it is used in the Kolmogorov-Smirnof Test[4] to verify whether two distribution functions are different,
- compute the mean of the D-Statistics over the whole testing set.

This statistic is the main quantitative performance criterion that will be use in this paper.

EXPERIMENTS

Data description

Two data sets were used in this paper[3]. The first one is the digital elevation model of Switzerland and will be designed as SWRND (left of Figure 1). The second one is a subset of the previous one, and focuses on the mountains of the eastern part of Switzerland. It will be referenced as GRISONS (right of Figure 1).

In both cases, 3 subsets were generated: a training, a testing and a reference set. Table 1 gives the number of points inside each set.

Subsets	SWRND	GRISONS
Train	500	500
Test	1000	1000
Reference	93628	16032
Neighbors	100	20

Table 1: Number of data points inside the various subsets of SWRND and GRISONS. Train and Test were extracted randomly from the full data set, and Reference is the remaining data. The value "Neighbors" indicates how many neighbors were taken in the Reference set to construct the reference cumulative histogram of the Test points.

For numerical stability reasons, the coordinates have been linearly transformed so that they are all positive and smaller than 1. For the same reasons, altitudes are given in kilometers in order to have smaller values.

The main characteristic of the SWRND data set is that its histogram (left of Figure 2) shows clearly a multi-modal distribution of the altitude values.

[3]these data sets are available at http://www.idiap.ch/learning/data/swissdem.tar.gz.

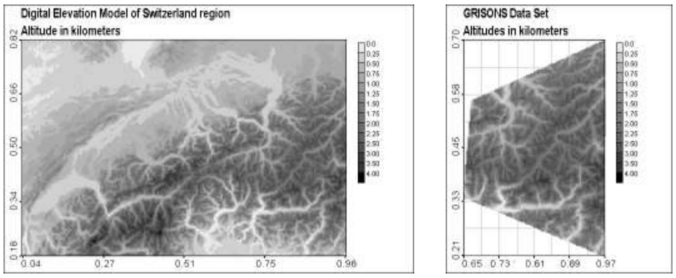

Figure 1: Presentation of the complete data sets SWRND (left) and GRISONS (right)

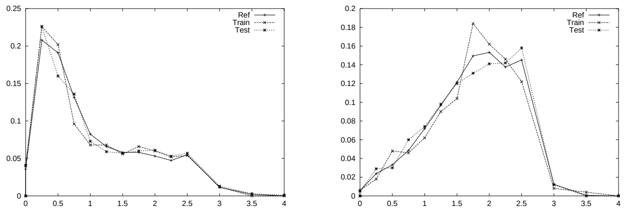

Figure 2: Statistical histograms of output values (altitudes in kilometers) for SWRND (left) and GRISONS (right). The plain curve is the reference set, the dash curve is the training set, and the dot curve is the testing set.

This can also be observed on left of Figure 1 where the high altitudes of the Alps appear in dark, while the low altitudes of the Swiss Plateau appear in light grey. The GRISONS data set looks simpler as its histogram is unimodal (right of Figure 2). However it has a high local variability, not easy to extract from the global tendency of medium altitudes.

The cumulative histograms used for the experiments are constituted of 100 intervals lying between 0.0 and 5.0 km, in such a way that any altitude of the data sets is covered.

The SWRND data set

The D-Statistics, detailed in Table 2, shows that, on the SWRND data set, conditional GMM globally yields better estimates of local PDF than SGS. This was an expected result because of the multi-modal behavior of SWRND altitudes. The Normal Score back-transformation of data, which is necessary for SGS, can produce bad results when data are far from the normal distribution, and even more if they are multi-modal. This kind of problems doesn't occur with GMM, since no assumptions need to be done on data distribution, except the fact that data are supposed to be independently and identically distributed.

Figure 3 shows how GMM and SGS manage to reconstruct the local cumulative distribution at two locations taken randomly in the testing set. On the left, GMM's cumulative distribution fits almost perfectly the reference curve while SGS is completely missing the point. On the right, both methods give a good estimation of local PDF, and in this case, SGS seems a little bit better than conditional GMM.

SWRND D-Statistics	Mean	Min	Median	Max
Conditional GMM	0.350	0.051	0.332	0.835
SGS	0.531	0.000	0.510	1.000

Table 2: Results from D-Statistics calculation on SWRND testing set, between the reference cumulative distribution and the estimations from Sequential Gaussian Simulation (SGS) and Conditional Gaussians Mixture Model (GMM). The smaller the statistics, the better the model.

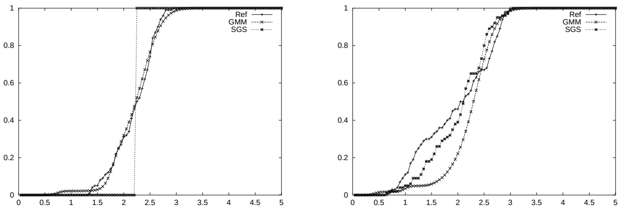

Figure 3: Comparison of the cumulative distributions of altitude at two different locations of the testing set. Plain lines are the references. Dash lines are the estimations from conditional GMM. Dots lines are the estimations from SGS.

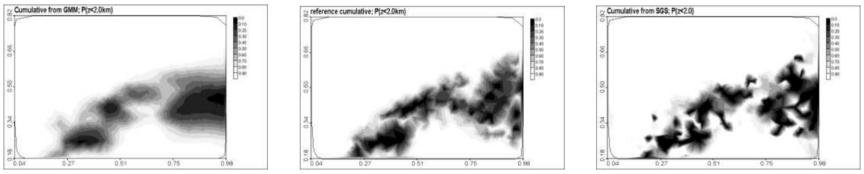

Figure 4: Risk Maps of the probability that altitude lies under 2.0 kilometers. The darker, the less probable. Conditional GMM risk map is on the left, reference is in the middle, and SGS risk map is on the right.

The risk maps of Figure 4 were constructed directly from the local estimation of the cumulative PDF of the testing set. Each map is a cut through these cumulative functions for $v < 2.0$ km given $\mathbf{u}$. A rapid comparison between the three models enlights the sharpness of SGS maps and the reference maps in front of the smoothness of GMM maps. The first reason is that the GMM model built for this data set contains “only” 95 Gaussians, while SGS is using a lot more points. The second reason is that GMM provides a continuous function while SGS and the reference don’t. As a consequence, there is a lot of discontinuities in SGS maps which make them appear sharp.

However, this “sharpness” of SGS is in fact mainly noise. Looking closer at the map on the right of Figure 4, SGS estimations don’t seem to be so close to the reference. GMM seems to be more efficient to keep the general structure, except in the Eastern part. One can finally see that GMM is generally also smoothing the distribution tales, as it seems to under-estimate the high probabilities and over-estimate the low probabilities.

The GRISONS data set

Table 3 shows that conditional GMM and SGS perform in a very similar way on the GRISONS data set. While for SWRND, D-Statistics performance of GMM was 33% better than SGS, it is now less than 4% better. On this data set, SGS is no longer perturbed by any multi-modal distribution, and thus, its performances are relatively better. On the other hand, GMM had a lot of difficulties to reproduce the variability of data: the optimal model found contains only 21 Gaussians, which is very few. However, it seems to be enough to perform efficiently regarding D-Statistics.

The similarity between GMM and SGS performances is also visible on Figure 5. For both sample locations, the cumulative from GMM and the one from SGS are very close to each other.

GRISONS D-Statistics	Mean	Min	Median	Max
Conditional GMM	0.385	0.081	0.360	0.965
SGS	0.400	0.080	0.370	1.000

Table 3: Results from the D-Statistics calculation on GRISONS testing set, between the reference cumulative distribution and the estimation from Sequential Gaussian Simulation (SGS) and Conditional Gaussians Mixture Model (GMM). The smaller the values, the better the model.

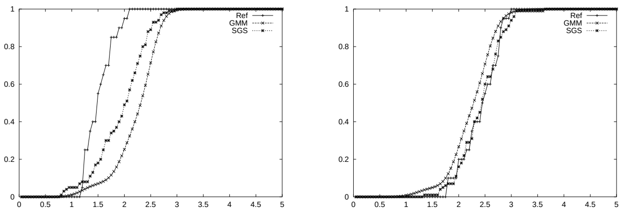

Figure 5: Comparison of the cumulative distributions of altitude at two different locations of the testing set. Plain lines are the references. Dash lines are the estimations from conditional GMM. Dot lines are the estimations from SGS.

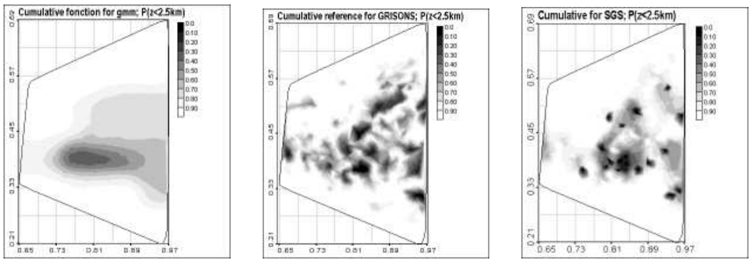

Figure 6: Risk Maps of the probability that altitude lies under 2.5 kilometers. The darker, the less probable. Conditional GMM risk map is on the left, reference is in the middle, and SGS risk map is on the right.

The smoothing tendency of GMM pointed out with SWRND data set becomes obvious when comparing the various risk maps of Figure 6. GMM did not manage to reproduce the complexity of the GRISONS data set and the optimal model generated was a very "simple" one. SGS appeared to reproduce this complexity, but in fact, results are more noisy than sharp, and out of the general tendencies (also found by GMM) it does not perform very well. It is interesting to notice that GMM and SGS are performing similarly in terms of D-Statistics but in a completely different way.

CONCLUSION

Conditional Gaussians Mixture Models proved to be efficient to estimate local probability density function in order to draw risk maps. When compared to the classical method used in this field, it appeared to be at least as efficient in terms of D-Statistics, and even better when the distribution of data is multimodal. A strong advantage of conditional GMM over SGS is that it needs less expert knowledge and less hypotheses on data distribution. It also gives a real function of the conditional probability of a variable at any location of the studied area, and can easily handle joint distributions of multiple output variables.

ACKNOWLEDGEMENTS

This work was supported by Swiss National Science Foundation (CARTANN project: FN 20-63859.00).

The GMM experiments were conducted using the Torch Machine Learning Library[4]. The SGS experiments were performed using the Geostat Office software.

REFERENCES

[1] C. Bishop, **Neural Networks for Pattern Recognition**, Clarendon Press, Oxford, 1995.

[2] P. Chauvet, **Processing Data with a Spatial Support: Geostatistics and its Methods**, Paris: ENSMP, 1993.

[3] A. Dempster, N. Laird and D. Rubin, "Maximum-Likelihood from Incomplete Data via the EM Algorithm." **Journal of Royal Statistical Society**, vol. B, 1977.

[4] A. Kolmogorov, "Sulla determinazione empirica di una leggi di distribuzione," **G. Inst. Ital. Attuari**, vol. 4, 1933, translated in English in *Breakthroughs in Statistics*, by Kotz and Johnson (editors), Springer-Verlag, 1992.

[5] C. Lantuéjoul, **Geostatistical Simulation**, Berlin: Springer-Verlag, 2002.

[6] R. Neuneier, F. Hergert, W. Finnoff and D. Ormoneit, "Estimation of Conditional Densities: a Comparison of Neural Network Approaches," **International Conference on Artificial Neural Networks**, vol. 1, pp. 689–692, 1994.

[7] C. K. Williams and C. E. Rasmussen, "Gaussian Processes for Regression," **Advances in Neural Information Processing Systems**, vol. 8, pp. 514–520, 1996.

[4] `http://www.torch.ch`