Anonymous Referee #4

Received and published: 15 January 2017

Brief Summary/General Comments:

This manuscript develops an approach to analyze and interpret rain disdrometer data by using statistical and analytical tools common in other scientific communities, but relatively unknown within the hydrological and rain microphysical communities.
As a frequent user of disdrometric data, I find the analysis presented in this paper to be fascinating and potentially transformative to future work in hydrology and rain microphysics. It is clear the methods presented here are extremely novel within this community, and it appears that there is a lot of valuable information that can possibly be gained by conducting analyses in the ways described within the text.
However, as other referees have noted, the entire manuscript needs a thorough revision by native English speakers in the interest of readability. Since the analytical methods used here are expected to be new to most of the intended readership, it is vitally important that the exposition is clear, unambiguous, and complete enough so that others can follow in the footsteps of these investigators; the tone of these sections would be most useful to the reader if they were very tutorial in nature. In particular, sections 3 and 4 (which involved fundamentally new ideas for me) require more detail and clarity to allow a reader to reproduce and/or mimic the work with similar data-sets.
As it stands, I only have a vague understanding of the details of the process that was undertaken – but the results and methods are intriguing enough that I desperately wish the manuscript could be rewritten in such a way that a diligent reader could reproduce the results. The text is currently opaque enough that it is difficult for me to determine whether or not there are technical issues related to the work.

Responses to general comments

In section 3, we have added more detailed information and references concerning the variables selection. The figure 2 has been enlarged and made more readable. In addition, the following sentence will be added in the conclusion : " the Matlab code of the genetic algorithm is available on request from the authors".

The section 3 & 3.1 is replaced by this one :

Computer-assisted variable selection is important for several reasons. Indeed, the selection of a subset of variables in a high dimension space can improve the performance of the model or its statistical properties, but also provides more robust models and reduce their complexity. In practice it is generally impossible to try all possible combinations of variables and select the best ones because of the computational cost. Many approaches exist for variables selection (Guyon and Elisseeff, 2003)**.** Among then we choose to develop a model based on genetic algorithms to search for the best variables subset. Genetic algorithms (GAs) (Holland, 1975) are stochastic optimization algorithms based on the mechanics of natural selection and genetics described by Charles Darwin. In our study, a chromosome is defined as a subset of the 23 variables. A first generation composed of a population of 60 potential chromosomes is arbitrarily chosen. The performances of each chromosome (i.e. for each 60 corresponding subset of variables) is evaluated through a fitness function *f*. The fitness function is defined in such a way that the higher it is, the more the fitness function is able to represent the whole dataset (of dimension 23) with a minimal number of variables. Based on the performances of the 60 chromosomes we create a new generation of 60 chromosomes of potential solutions using classical evolutionary operators: selection, crossover and mutation. The performances of this new generation is then evaluated. This cycle is repeated until the stop criteria is asserted. The best chromosome of the last generation provides the optimal subset of variables.

3.1 Methodology

Let's define by x^k the chromosome number k : $x^k = (x_1, x_2, \ldots, x_{23})$. x^k is a binary vector in $\{0,1\}^{23}$ space such that each component has the following meaning:

$$\forall i \in \{1,\ldots,23\},\ \begin{cases} x_i = 1 & \textit{The variable number } i \textit{ in Tab. 2 column 1 is selected} \\ x_i = 0 & \textit{The variable number } i \textit{ in Tab. 2 column 1 is not selected} \end{cases} \quad (1)$$

The word "selected" in eq. 1 means that the corresponding variable will be used both in the training step described just below in "Step 2" and for performances evaluation. Otherwise if the corresponding variable is not selected it will be used only for performances evaluation.
As previously stated the fitness function allows to provide a measure of how well a minimal subset of variables can

represent the entire data space (in dimension 23). To do that the fitness function f is defined as follow :

$$f(x^k) = \frac{1}{nb(x^k)\ te(x^k)} \quad (2)$$

With

x^k : *chromosomes number k*

$nb(x^k)$: *number of selected variables in chromosome* x^k

$te(x^k)$: *topological error associated to chromosome* x^k

The aim is to minimise the number of selected variables *nb* and the topological error *te*. Consequently we seek to maximize the fitness function. The estimation of the topological error made from a Self-Organizing Maps (SOM) is somewhat complicated and requires some explanation. Self-Organizing Maps introduced by Teuvo Kohonen (Kohonen, 1982, 2001) is a popular clustering and visualization algorithm. SOM is a neural network algorithm that is based on unsupervised learning based on competitive learning (Kohonen, 1982, 2001; Vesanto and Alhoniemi, 2000). It may be considered has a nonlinear generalization that has many advantages over the conventinal features extraction methods such as Empirical Orthogonal Functions (EOF) or Principal Component analysis PCA (e.g., Liu et al., 2006). SOM applications are becoming increasingly useful in geosciences (e.g., Liu and Weisberg, 2011). As was written by Uriarte and Martín (2008) : "The SOM provides a non-linear, ordered, smooth mapping of high-dimensional input data manifolds onto the elements of a regular, low-dimensional array. The main characteristic of the projection provided by the algorithm is the preservation of neighbourhood relations; as far as possible, nearby data vectors in the input space are mapped onto neighbouring locations in the output space". This property allows to compute easily a topological error (see Uriarte and Martín, 2008, eq. 2). For each of the x^k chromosomes a Self-Organizing Maps $M(x^k)$ is learned on the training data set. Only the selected variables are used during the learning process. Finally for each Maps $M(x^k)$ the topological error $te(x^k)$ is then computed according to eq. 2 in Uriarte and Martín (2008). Section 4 provides additional information concerning Self-Organising Maps.

The Genetic Algorithm is based on the following five steps (Fig.2.):

Step 1- Initialization: (initial population)
Generate randomly, a population $\{x^k$, k=$1, \dots, 60\}$ of 60 chromosomes of dimension 23.
Step 2- Evaluation: For each of the x^k chromosomes a SOM $M(x^k)$ is learned. Only the selected variables are used for learning. Once training each Maps provides the topologic error $te(x^k)$ allowing to calculate their fitness score $f(x^k)$ by the mean of the fitness function.
Step 3- Select the best chromosome x^{Best} among the 60 chromosomes according to its fitness score previously computed on the test dataset. If x^{Best} remains the same during 50 generations then stop the procedure and select the relevant variables, i.e. the ones for which the corresponding components are equal to 1 in x^{Best}. Else go to step 4.
Step 4- Selection: Create a new population of 60 chromosomes from the current population by random sampling with replacement of chromosomes based on their probabilities calculated according to the formula:

$$\Pr(x^k) = \frac{f(x^k)}{\sum_{i=1}^{60} f(x^i)} \quad (3)$$

Step 5- Reproduction: Mutation and Crossover possibilities in the new population.
Mutation: It consists of modifying (or not) some components of the chromosomes. The probability of mutation is in general very low and is commonly set to $p = 10^{-7}$. In our case the number of necessaries generations to reach our objective is lower than a few hundred. Consequently in our case, the probability of a mutation is highly unlikely.
Crossover: First, we randomly draw $\frac{60}{2} = 30$ couples of chromosomes from our population. Then, for each couple (x^k, x^l) (called parents) one crossover point, noted I_c, is randomly drawn in the range [1, 23] using a discrete uniform law. Two new chromosomes ($x^{k'}$, $x^{l'}$) are created in the following way:

$$\begin{cases} x^{k'} = (x_1^k, x_2^k, \dots, x_{I_c}^k, x_{I_c+1}^l, x_{I_c+2}^l, \dots, x_{23}^l) \\ x^{l'} = (x_1^l, x_2^l, \dots, x_{I_c}^l, x_{I_c+1}^k, x_{I_c+2}^k, \dots, x_{23}^k) \end{cases} \quad (4)$$

So from two parents we generate two children allowing having a new generation with the same number of chromosomes. Finally go to step 2.

Figure 2: Diagram for the selection of variables based on a Genetic Algorithm associated with Kohonen Maps and a fitness function

In Section 4 : Some additional information are provided. The following sentences replace lines 11 to 27 p7

A SOM is a topological map composed of neurons. In our case, a neuron is a vector of dimension 23 containing the 23 variables defined previously in Tab. 2. Each neuron has 6 neighboring neurons. SOM is an unsupervised neural network trained by a competitive learning strategy that performs two tasks: vector quantization and vector projection. Different from K-means, SOM uses the neighborhood interaction set to learn the topological structure hidden in the data. In addition to the best matching referent vector (neuron), its neighbors on the map are updated, resulting in regions where neurons in the same neighborhood are very similar. It can be considered as an algorithm that maps a high dimensional data space to a two dimensional space called a map. A map can be used at the same time both to reduce the amount data by clustering and for projecting the data in a non-linearly way to a regular grid (the map grid).
In this study we used the toolbox developed by "the SOM Toolbox Team" available at the following address: http://www.cis.hut.fi/somtoolbox/. A SOM with 8×8 = 64 neurons is considered here. This choice corresponds to a compromise. A smaller map would not be able to distinguish fine details, whereas a large map would not make sense given the number of observations available.
After training by the GA algorithm described the previous section the obtained map $M(x^{Best})$ can be used to affect to any event the best matching referent vector (neuron) according to the 5 selected variables associated to the chromosome x^{Best}. Hence the obtained $M(x^{Best})$ map can be considered as an optimal representation of the initial data set.
Figure 3 shows the distances matrix. For each neuron the color indicates the mean distance between a neuron and its neighbors. The value at the center of each neuron represents the number of rain events of the training data set captured by the corresponding neuron. All neurons capture rain events and a bit more than half of them capture between 3 and 5 rain events which is close to the value that would be obtained ($234 / 64 \cong 4$) if the rain events were uniformly distributed on the map.

The following references were added :

Guyon, I. and André Elisseeff; An Introduction to Variable and Feature Selection (Kernel Machines Section)3(Mar):1157--1182, 2003.

E. Arsuaga Uriarte, and F. Díaz Martín, Topology Preservation in SOM, World Academy of Science, Engineering and Technology. International Journal of Computer, Electrical, Automation, Control and Information Engineering Vol:2, No:9, 2008

Kohonen, T. (2001). Self-Organizing Maps. Springer-Verlag, ISBN 3-540-67921-9, New York, Berlin, Heidelberg

Vesanto, J. & Alhoniemi, E. (2000). Clustering of the self-organizing map. IEEE Transactions on Neural Networks, Vol. 11, 586–600, ISSN 1045-9227

Liu, Y.; Weisberg, R. H. & Mooers, C. N. K. (2006). Performance evaluation of the self- organizing map for feature extraction, Journal of Geophysical Research, Vol. 111, C05018, doi:10.1029/2005JC003117, ISSN 0148-0227

Liu, Y.,and R.H. Weisberg (2011) A review of self-organizing map applications in meteorology and oceanography. In: Self-Organizing Maps-Applications and Novel Algorithm Design, 253-272.

Specific Comments:

In addition to the presentation concerns outlined above (and by the other reviewers), I do have some basic scientific questions to add to the discussion.

1. As noted by other reviewers, I am curious as to how much the choice of a 30-minute MIT interval used to define a rain event effects the results. Similarly, how much does the minimum detection threshold influence the results?
Our choice was made based on the literature. Even if it cannot be called a sensitivity study, we also run some test to see if the conclusion drawn in the lite nevertheless literature were coherent with our data set.
We have not really investigated this point and consequently we are not really in position to answer to this question. A greater MIT value will aggregate rain periods together and consequently will modify some parameters like the event duration De or the Rain event Depth Rd. Some others parameters can remain unchanged or not, this is the case for example for parameters Pci (which is sensitive to the variability of the rain rate). We expect that a higher MIT may aggregate events of a different type, whereas a shorter MIT tends to increase the number of events while they belong to the same group of rain cells.

The chosen threshold (0.1 mm/h) allows to detect very light rain while avoiding false detections due to dust or insects. This threshold can play an important role concerning the rain support but has very little influence on whole of the others paramaters used in this study

2. Given that this method is new to some readers, could it briefly be described what happens with this method if it is

employed with non-transformed data? The normalization method used to make each of the variables pseudo-normal sounds quite practical, but I always wonder what biases this can introduce in fundamental non-linear processes like rainfall. What would happen if such a transformation is not utilized?
We used 7 transformations. Among the seven (see Table 3) 6 transformations are nonlinear. There are simply used to contract/dilate the data space to guarantee a better spreading of the map among the data. Without any transformation the resulting map would be less representative of the data.
Yes there are quite practical, properly speaking, there is no statistical or probabilistic framework in this article (No estimator, no confidence interval or no whatsoever are involved.)
The nonlinear transformations are intended to avoid any biais.

3. The instrument you utilized had a 1-minute integration time. There are other detectors with sub-second integration times, and one can always coarsen data. What influence does integration time have on the method proposed? Does one find the same basic results (for the entire process if the full analysis is done with a different integration time-scale? This is not merely of academic interest here, since scale-matching is of vital importance within the hydrological community where instruments with very coarse resolution (e.g. radar) are often "ground-truthed" with point-detector measurements that can have sub-millisecond resolutions.

From a practical point of view we could simply aggregate the data if they are at a resolution finer than 1 minute. Otherwise :
Almost the 23 parameters are derived from rain rate parameter. Hence the vectors representing an events will be different from the ones at a 1-minute resolution. We expect to find the same basic results conditioned on the loss of information due to a coarser resolution. For resolution closed to 1 minute the expected results should be close.
It is one of the perspectives of this work. It would give access to more exhaustive (as well in space and in time) data sets. We are working on this aspect. It has to be noticed that the nature of the measurements is different. (For example, a tipping bucket rain gauge is measuring the time to fill a given volume. It has some important implications for stratiform rain events.)
Since the resolution is less good it will definitely impact the results. It will strongly impact some parameters such as the Rmax and the PC parameters. Since Rmax and PC_1 are two of the five parameters selected by the algorithm it will impact the results. This is nevertheless a study work in itself.

4. As noted by other reviewers, the conclusions seem a bit of an over-reach for a study done using one type of instrument, with one type of MIT choice, one minimum detection threshhold, at one location, with one integration time. I believe that if this approach was applied by a number of investigators at different locations using data from a number of different instruments and a variety of parameter choices it is possible that a very powerful result could be attained. I would love to work on such a study, but – due to the quality of written English in this paper – I can't follow the method closely enough to do so at this time.

I would like to petition the authors to give this paper a substantial re-write and have native English speakers edit it. In the interest of readability, I would encourage figure captions to be more descriptive. Finally, I would love to have access to either code or other materials necessary to replicate the work (so, as other reviewers have pointed out, unspecified details like the choices made in te(xˆk) are known to a practitioner).
If these changes could be implemented, I would promise the authors that not only will this paper be read, but some of us will use the results in their own work – myself included.

We agree. A professional translator will edit the manuscript. Additional information is now provided in sections 3 and 4. References are added to help to replicate the work.