An evolutionary data mining approach on hydrological data with classifier juries

Wilfried Segretier*, Manuel Clergue*, Martine Collard*, Luis Izquierdo†
*Mathematics, Computer Sciences and their Application Laboratory (LAMIA)
University of French West Indies (UAG)
†Computer Sciences Department
University of La Habana

***Abstract*—In this paper, we present an evolutionary approach for extracting a model of flood prediction from hydrological data observed timely on water heights in a river watershed. Since this kind of data recorded by sensors on river basins is highly scarce and hopefully much unbalanced between cases of floods and non-floods, we have adopted the notion of aggregate variables which values are computed as aggregates on raw data. An evolutionary algorithm is involved to allow selecting the best sets - juries of classifiers- of such variables as predictive variables. Two real hydrological data sets are trained and they both show the efficiency of the method compared to traditional solutions for prediction.**

I. Introduction

River flooding produces big disasters all over the world with important damage and loss of lives. For instance, in 1931 the flood of the Yellow river in China and in 1993 the flood of the Mississipi river were highly devastating. More recently, in 2009 the flood of the Lezarde river, studied in this paper, produced destructive inundations and had tragic consequences in the island of La Martinique in the Caribbean area. Beyond those primary major effects, secondary results on transportation systems, disruption of gas and electric services or destruction of wildlife cause much concern to regional and environment agencies that are expecting accurate and reliable forecasting systems.

Floods like other geophysical signs like earthquakes, rainfalls and hurricanes are hardly predictable since they are governed by a large number of phenomena and they are non linear systems. They are depending to various degrees on water heights, flow discharge, rainfalls, ground topography, infiltration, snow and ice melt or sea tides. Among hydrological prediction systems, earliest models were deterministic physical models [1], [2] that use differential equations to represent the water runoff in the basins and are quite complex. They need a wide volume of spatial data and are known to provide prohibitive computational time. More recent approaches have explored the efficiency of artificial neural networks and stochastic models [3], [4] that do not consider physical transformation mechanisms in water flows but integrate the random aspect of these phenomena.

As argued by Damle et al. [3] flood prediction is a different issue from flow forecasting. Indeed it represents an even more challenging issue since events rather than values are predicted. To ensure accuracy, flood modeling needs detailed and uniformly distributed recorded data observed on different sensors along the basin. But *data scarcity* is known as a recurrent problem in hydrological studies. Hopefully non-flood states are much more frequent in watersheds but this situation produces very parsimonious relevant data such that the task is hard to extract reliable and stable models. Another characteristic is the specific profiles of watersheds governed by typical regional features such as snow melt in mountainous basins or hurricane phenomena in tropic regions for instance that command to design *customized solutions*. Current solutions are mostly fitted to specific basin.

We have investigated a flexible stochastic approach designed to address these issues. Our final objective is to propose a new model for this kind of natural systems that we can assume to be complex since factors like water height, flow, rainfall, saturation rate, slope rate or ground types that behave rather independently seem to infer collectively the flood phenomenon. The model flexibility is expected in order to obtain a solution able to be applied on different basins. One practical advantage with such a model will be on a long term to provide sufficient information for the optimization of limnimetric and rainfall sensor locations on the river basin. The complex system approach adopted consists in a first stage on applying data mining and optimization techniques in order to extract the most relevant knowledge on each factor implied. A second stage will be to simulate these individual models and merge them as multi-agents in order to observe possible emerging collective phenomena inducing similar floods as those observed.

In this paper, we focus on the first stage of the project to propose an extensible method in order to optimize the selection of features for predicting high limnimetric or discharge level overflow on one of the last sensors downstream a river bed. This sensor state may be considered as the prediction variable. We assume thus that a threshold overflow on this sensor - called *event sensor* - is equivalent to a flood occurrence. The challenge is not only to globally optimize the system predictive performances but more precisely and according to a decreasing priority:

- to ensure the first requirement that the FN (False Negative) rate has to be very low since this kind of error may have dramatic consequences,
- to extend as much as possible the flood anticipation time,
- not to neglect the FP rate to ensure system relevance.

We have conducted experiments on two sets of data from two very different contexts, a long and slow event watershed and a short and very fast event watershed. In both cases, height or discharge water levels were recorded by sensors located along the river course. Flood alarms are currently triggered when one sensor is recording a level reaching a predefined threshold. The current systems generally perform rather well to predict high values on the whole river with low rates of false positives (FP) and false negatives (FN). But the inherent phenomenon of flood in the river basin surroundings is not well managed. With same water levels observed on same spots at a same time before a flood, sensitive areas down below the river bed may be under water or not depending on other factors obviously. Another long-term issue to address is thus to study the whole mechanism that result in flooding all these areas. For now, in the current stage, we are looking for an explicit and comprehensible model that provides detailed knowledge on the flooding process.

This work was initiated and funded by the General Council of the island of *La Martinique* in French West Indies who is much concerned by river flood problems since strategic places in the island (main roads, airport, industrial areas) are threatened.

A classical approach could be to take natural variables that represent limnimetric or discharge values observed upstream for predicting a high level on this given sensor downstream. And indeed this kind of predictive models may apparently perform well as we show in further sections. But they obviously cannot be considered as relevant and sound solutions since they are learnt on punctual and sparse data. We show in this paper how to define much more relevant variables that integrate levels observed on a time period. Thus we consider variables, called *aggregate variables*, that represent aggregate values on a time period - called *aggregation period* - rather than punctual values. These variables are defined by a set of parameters that give more flexibility allowing to consider different intervals of observation and prediction. The approach is extensible to other numeric records like rainfalls data or sea tides.

In this work, we have trained available data that represent real levels recorded by sensors all along two non similar river courses. To address the issue of data scarcity, we have applied an optimization technique in order to select the best *aggregate variables* that represent aggregates of source raw values. An evolutionary algorithm has been employed to search for the best sets (or juries) of predictive variables among the wide space of potential solutions. The experimental results we obtained show good performances by comparison to standard predictive solutions. They demonstrate the predictive power of aggregate variables grouped in juries of classifiers.

The paper is organized in five sections. Section II describes source data, the data pre-processing method and a first simple exploratory analysis based on traditional predictive algorithms as a reference. Section III introduces the principles of the evolutionary approach based on classifiers-that implement the concept of aggregate variables- and juries of classifiers. In Section IV we discuss experimental results obtained and in Section V we conclude and present future works

II. Source data

The two data sets involved in this study are:

- a set of limnimetric levels (gage heights) recorded on several spots along the *Lezarde* river in the island of *La Martinique*, French West Indies by the departmental flood alert system (SDAC),
- a set of discharge values observed at multiple spots of the *Missouri, Grand river* and its affluents, from the U.S. Geological Survey (USGS) National Water Information System [1].

In the following we call them *SDAC* data and *USGS* data sets. In a data-mining process, a particularly important step is the data pre-processing as it allows to clean and transform the raw data that are frequently noisy, may be unreliable and contain missing values and are generally in a inadequate format for learning algorithms, in order to generate good training and test set candidates.

In this section, we first describe the source data and explain how they were pre-processed, then we present as a reference for further comparisons, the performances obtained by traditional classification algorithms on simple (i.e. non aggregate) variables.

A. Raw data

The SDAC source data, after format standardization, consists of eight files corresponding to eight sensors distributed along the river basin as showed by figure 1. The average distance between sensors is about five kilometers. Each one registers the water level in millimeters (mm) at its location every 6 minutes. The data files contain these periodical measurements for the period January 2006 - August 2010.

Among the very large amount of data available on the USGS National Water Information System for each state of the USA, we picked-up a set of data corresponding to measures recorded by seven stations located along the course of the *Grand River* or some of its affluents such as *Thompson River*, *Missouri* as illustrated by figure 2. These measures represent discharge values (ft^3/s) and are generally recorded every 15 minutes for a time period ranging from October 1994 to September 2010. As we will see in the following, there might be sometimes rather large time ranges during which data are unavailable for

[1] http://waterdata.usgs.gov/nwis

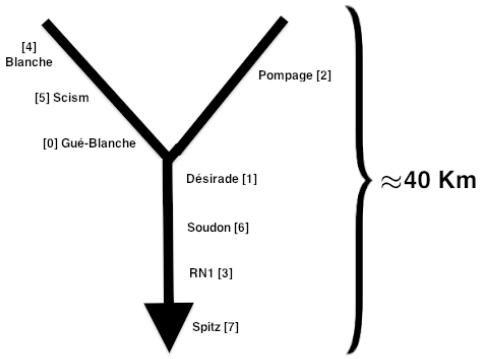

Fig. 1. Schematical view of *La Lezarde* sensors layout

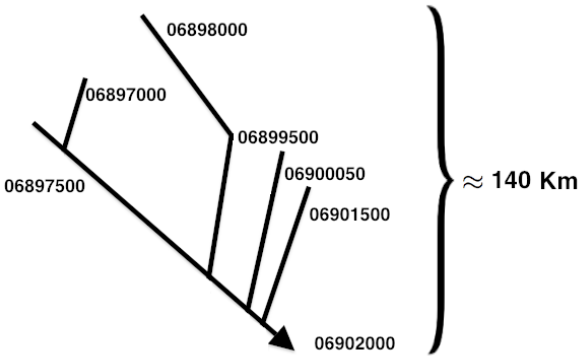

Fig. 2. Schematical view of *Grand River* sensors layout

one station.
The USGS dataset is much larger than the SDAC and its topological, meteorological and hydrological characteristics are significantly different. We have selected these two examples that are quite representative of real watersheds in order to train our approach on two very different samples to check its generalization ability.

B. Data pre-processing

Missing data due to sensors malfunctions are the more common irregularity that we found in these two data sources. We observed an amount of missing values ranging from 12.2% to 56.8% (31.5% on average), depending on the sensor considered, in the SDAC case, and from 16.6% to 76.6% (49% on average) in the USGS case. These missing records may correspond to short (minutes, hours) and long (weeks, months) time periods. When the period length was short enough (less than six times the default interval), we applied a linear interpolation technique to generate the missing values. We have indeed observed linear behaviours on such periods. The i^{th} missing value d_i ($1 \leq i \leq 5$) between d_0 and d_{n+1} was computed as $d_i = d_0 + i \times R$, with $R = \frac{d_{n+1} - d_0}{n}$ and n the number of contiguous missing values. After this step, the average amount of missing data decreased to 30.2% in SDAC data and to 30% in USGS data.

An even more important issue was the presence of erroneous data. By means of data visualization we have found some abnormal behaviors, such as sudden level risings for only one station while the others remain normal. Since unreliable data can easily lead to erroneous conclusions, we have manually eliminated at least the more obviously wrong fragments for each sensor.

In the following, we call *event sensor* - *target station* -, the sensor - or station - on which we want to predict flood or non-flood events and *measure sensor*, a sensor which data are used for prediction. Several elements may influence the choice of the event sensor, such as the inundation risks of the surrounding region, the number (and quality) of both event types available to generate efficient train and test datasets or the amount of upriver available data. In this paper, experimental results have been obtained with *Soudon* as the target sensor on SDAC data and *06902000* as target station on USGS data, with the possibility of a future application of the same methodology to other downriver spots.

In order to label the data for supervised classification, we have compared each record to the alert threshold set up for the target station in the current system: when a value overtakes the threshold, we label it as the beginning of a Flood (F) event. As we are interested only in the conditions that precede the very beginning of the floods, we have discarded threshold overflows that represent the continuation rather than the beginning of an event. As it may happen that during a flood event, water levels vary several times over and under the alert threshold, we have defined a safe threshold under which we are sure that the event ended. We consider as a Non-flood (N) event any value under the alert threshold with the only constraint of being far enough from a positive case, i.e. at least during a period P before its beginning or after its end.

Table I shows the amount of threshold overflow events found in selected SDAC and USGS stations. One can see the scarceness and unbalanced distribution of positive events: 10 to 26 compared to the 250000 points (on average) in each data file for the four and a half year period studied in the first case and 6 to 95 for 400000 (on average) points on sixteen years in the second case. Consequently, the large majority of the data represents non-flood events, so that we had to find a way to reduce this very high class imbalance. Since rivers are often quiet, if we randomly pick-up non-flood events among the whole available data, most of them will have low measures which will not help to discriminate high non-flood events from flood events. This is the reason why we chose to apply a uniform sampling technique which consists in splitting the complete range of possible values (under the flood threshold in this case) in equal intervals and selecting uniformly samples in each interval.

C. Exploratory analysis

In order to get a first estimation of which kind of performances could be obtained from these data with simple

TABLE I
THRESHOLDS AND OVERFLOWS OF DOWNRIVER STATIONS

Period	Station	Threshold	Overflows
2006-2010	Soudon	2100 mm	26
	Spitz	2700 mm	10
	Pont RN1	2900 mm	8
1994-2010	06902000	22000 ft^3/s	95
	06897500	24100 ft^3/s	42
	06899500	39000 ft^3	6

approaches, we have generated simple (non aggregate) datasets containing on each line the values of each measure sensors taken δt minutes before a flood (F) or non-flood (N) event occurring on time T on the event sensor. Since there is a time shift δt of *earliness* between times when the prediction values and the event are observed, the event sensor may be also used as a measure sensor. When a measure sensor value is not available, it is replaced by "?". Table II shows an extract of such a dataset. The first column corresponds to the measure sensors values whereas the last one represents the class to predict (F or N).

TABLE II
SAMPLE OF A SIMPLE DATASET: HEIGHT LEVELS ON MEASURE SENSORS RECORDED δt MINUTES BEFORE A FLOOD OR NON-FLOOD EVENT ON AN EVENT SENSOR

Sensor	1	2	...	n	class
Height levels	257	699	...	1373	F
	?	582	...	350	N
			...		
	373	?	...	1244	F
	678	903	...	1618	F

We applied different standard classification algorithms available on the Weka[2] data-mining platform, such as C4.5 [5], Best First Tree (BFT) [6], Functional Tree (FT) [7], Naïve Bayes (NB) [8] or Multilayer Perceptron (MLP) (neural network), on our generated datasets. Tables III, IV and V shows average performances obtained by these algorithms for different δt earliness values on SDAC and USGS data. The first column gives the δt *earliness* value, the second column gives the name of the algorithm and the last column gives five rates: False Positive (FP), False Negative (FN), True Positive (TP), True Negative (TN) and weighted average accuracy $\left(Acc = \frac{|Flood|*TP+|NonFlood|*TN}{|Flood|+|NonFlood|}\right)$ rates. Since the number of flood events available in SDAC data is too low, we provide performances of the different methods on the training set only. It was indeed unreliable to split the original dataset in separate sets for training and test phases. However, for the sake of completeness, we also give performances obtained by 10-fold cross validation on the training set. In the USGS case, we have randomly separated the datapoints into a learning and a test dataset. Thus the performance values given in this case correspond to those obtained on the test dataset.

[2]http://www.cs.waikato.ac.nz/ml/weka/

We can observe that in both cases, the true negative (TN) rates stay rather high (more than 90% on average) whatever the δt period is, whereas the true positive rates (TP) are more sensible to this period. Globally, we can see that the more δt earliness value increases, the more the TP rates decreases (from 84.2% to 0% in SDAC case and from 74.2% to 19.4% in USGS case). Since we kept an imbalance between the two classes (about 5 to 10 Non-floods for a Flood) to reflect the reality, the weighted average accuracy (Acc) stays rather high, but the most important objective to reduce FN (or optimize TP) is far from being reached even for low earliness values.

TABLE III
LEARNING PERFORMANCES (%) OF CLASSIFICATION ALGORITHMS ON SDAC DATA.

δt	Algorithm	Performances				
		FP	FN	TP	TN	Acc
60	C4.5	0	100	0	100	90.7
	FT	2.2	36.8	63.2	97.8	94.6
	BFT	0	15.8	84.2	100	98.5
	MLP	2.2	36.8	63.2	97.8	94.6
	NB	13.5	36.8	63.2	86.5	84.3
120	C4.5	0.5	68.4	31.6	99.6	93.1
	FT	1.1	73.7	26.3	98.9	92.2
	BFT	0	100	0	100	90.7
	MLP	0	73.7	26.3	100	93.1
	NB	14.6	52.6	47.4	85.4	81.9

TABLE IV
10 CROSS FOLD VALIDATION PERFORMANCES (%) OF CLASSIFICATION ALGORITHMS ON SDAC DATA.

δt	Algorithm	Performances				
		FP	FN	TP	TN	Acc
60	C4.5	1.5	80.8	19.2	98.5	91.3
	FT	2.3	50	50	97.7	93.4
	BFT	2.3	61.5	38.5	97.7	92.4
	MLP	4.2	69.2	30.8	95.8	90
	NB	11.4	26.9	73.1	88.6	87.2
120	C4.5	1.1	80.8	19.2	98.9	91.7
	FT	2.3	73.1	26.9	97.7	91.3
	BFT	3.4	76.9	23.1	96.6	90
	MLP	1.5	73.1	26.9	98.5	92
	NB	11.8	53.8	46.2	88.2	84.4

III. EVOLUTIONARY APPROACH

The basic principle we will follow for flood prediction using limnimetric or discharge data is to consider the water levels during a time interval and to try to predict whether an overflow is going to take place some minutes (or hours) later in a station's proximity. From a data mining perspective, the problem is about classifying a data vector, or actually a set of data vectors, as precursor of a flood event or not. In order to do so, it's necessary to find the regularities among flood previous conditions, that differentiate them from non alarming conditions.

TABLE V
VALIDATION PERFORMANCES (%) OF CLASSIFICATION ALGORITHMS ON USGS DATA.

δt	Algorithm	Performances				
		FP	FN	TP	TN	Acc
60	C4.5	2.8	33.3	66.7	97.2	91.9
	FT	2.8	30	70	97.2	92.4
	BFT	5.6	26.7	73.3	94.4	90.7
	MLP	2.8	30	70	97.2	92.4
	NB	2.8	26.7	73.3	97.2	93
180	C4.5	1.4	35.5	64.5	98.6	92.5
	FT	0.7	35.5	64.5	99.3	93.1
	BFT	1.4	35.5	64.5	98.6	92.5
	MLP	4.9	35.5	64.5	95.1	89.6
	NB	1.4	25.8	74.2	98.6	94.2
360	C4.5	2.8	45.2	54.8	97.2	89.6
	FT	2.1	29	71	97.9	93.1
	BFT	2.8	51.6	61.3	48.4	97.2
	MLP	9.2	38.7	61.3	90.8	85.5
	NB	2.1	29	71	97.9	93.1
720	C4.5	4.9	77.4	22.6	95.1	82.1
	FT	1.4	61.3	38.7	98.6	87.9
	BFT	2.8	80.6	19.4	97.2	83.2
	MLP	8.5	58.1	41.9	91.5	82.7
	NB	4.9	51.6	48.4	95.1	86.7

A. Simple classifiers

Our first and simpler classification method is based on an aggregate variable analysis. This approach aims to characterize the data for a given time interval by applying a mathematical operation to the records of just one station. For our case, the mathematical operation, called aggregation function, can be one of the functions "maximum", "minimum", "average", "standard deviation" and "slope" (the latter refers to the slope of a linear regression line). A simple aggregated variable-based classifier or "simple classifier" will assign a boolean (alarming/normal) classification to a data vector according to the comparison between the aggregation function result and a fixed threshold.

The parameters that define a simple classifier are: Station; Aggregation function; Aggregation interval; Earliness; Threshold; Comparison sense ($\leq$, $\geq$). Exhaustive exploration of all possible value combinations for the classifier parameters isn't possible due to the obvious combinatorial explosion: 7 or 8 (according to the data set) possible stations, 5 implemented aggregation functions, numerous conceivable aggregation interval lengths, nearly any imaginable comparison threshold...

An evolutionary algorithm was used to search for good parameter combinations for simple classifiers. Indeed, evolutionary algorithms, and metaheuristics in general, are particularly adapted to this kind of optimization problems, mixing discrete and continuous variables. When applying evolutionary algorithms to solve a problem, one needs to design three elements: the coding of individuals, the fitness function and the genetic operators.

We adopt a natural coding for the individuals, i.e., we directly handle classifier parameters by genetic variables. Some of them are boolean or discrete variables with a reduced domain, other are integer variable with a larger domain, and one of them is a real parameter. Table VI lists all the variables and their types.

TABLE VI
SIMPLE CLASSIFIER PARAMETER MUTATION PROBABILITIES

Parameter	Type	Mut. rate	Domain
Threshold	real	0.15	real
Station	discrete	0.005	number of stations
Comparison sense	boolean	0.005	0,1
Agg. function	integer	0.05	[30,720]
Earliness	integer	0.025	[δt, δt+360]
Agg. Interval	integer	0.05	number of agg. functions

As mutation operator, we have implemented one that modifies each characteristic of a classifier independently and with different probabilities. Some changes tend to modify radically the behavior of a classifier, such as an aggregation function or station replacement. For example, the set of intervals for which the slope of the levels at Spitz is greater than 5 has nothing to do with that of the intervals with an average level greater than 5 at Spitz. As a second and extreme example, in the case of the comparison sense, its modification totally inverts the classification. However, slight variations can be made to other parameters as the comparison threshold or the aggregation interval producing a much more gradual change in classifier's behavior. In order to make more frequent the mutations that affect more slightly the classifier, we have established the mutation probabilities showed in table VI for each one of its parameters. Each variable mutates according to its type. Boolean and discrete variables mutate uniformly in their domain. Integer and real variables mutate with a normal law (discretized for integer) centered on the current value of the variable with a fixed standard deviation, 50 for the real variable and domain length divided by 6 for the integer ones.

As a consequence, mutations are not going to be too frequent and most of the time they will affect only one or two of the least sensible characteristics of the individuals, thus avoiding too chaotic movements in the search space. The expected mutation probability of a precise individual in a generation is 0.26.

Our first impression was that a variable-wise crossover would not be efficient, and some preliminary tests comfort us in this impression.

The fitness of an individual is a linear combination of the classification correctness on a learning set composed of event (positive) cases and non event (negative) cases. For each row in the learning set, there are 5 possible responses:

- true positive (TP): the point is an event and is correctly classified
- true negative (TN): the point is a non event and is correctly classified
- false positive (FP): the point is an event and is incorrectly classified
- false negative (FN): the point is a non event and is incorrectly classified
- unclassified (U): data are unavailable so the classifier can not compute its response

TABLE VII
COEFFICIENTS USED FOR FITNESS COMPUTATION

Coefficient	Value
TP	10
TN	1
FP	-2
FN	-15
U	-2

Note that if data are missing for more than a half of the aggregation period for a case, we decide that the classifier responds the last one, i.e. unclassified. Thus, the fitness (f) of a classifier is the linear combination of the number of responses of each type (n_X), weighted by coefficients (c_X) that allow to set their relative importance according to the objective :

$$f = n_{TP}*c_{TP}+n_{TN}*c_{TN}+n_{TP}*c_{FP}+n_{TP}*c_{FN}+n_U*c_U$$

The coefficients values are given in table VII.

The evolutionary algorithm used elitist replacement, which grants that every generation's best individual will be at least as good as the best of the previous one.

Experiments exhibit poor results and low performances, for two main reasons. First, as we shown in section II, some sensors may be unavailable for varying period of time, from short ones to longer ones. If the simple classifier follows a sensor that becomes unavailable, it looses its only source of information, and become unable to compute a prediction, whatever its accuracy during normal period. Second, floods may occur from only one of the affluents of the river, especially in Martinique, where rainfalls are very localized. Listening only to only one station, as simple classifiers do, prevents to know what happens on other affluent. This leads us to consider jury of classifiers instead of single classifier.

B. Classifier juries

A simple classifier characterizes a data interval taking into account the information of only one station. It is hence very sensible to the errors or the absence of data for its particular station: even if there is high quality data for an interval from most of the sensors, the noise or lack of data in a specific station can easily lead to erroneous classification. The combination of several simple classifiers comes in as a resource for improving the classification robustness.

If the causes of different sensor malfunction are independent enough, the probability of simultaneous failure of several stations should be smaller than the individual failure probabilities. Thus, we can hope that from a set of classifiers not necessarily based on the same station information, there is often an important fraction of them that is not affected by sporadic data faults. Following that idea, we chose as simple classifier combination method the principle of a jury. In order to make the prediction for a moment, each simple classifier of a jury performs its individual classification of precedent data (with respect to its station, its aggregation interval, etc) and the jury's final classification is decided based on majority rule.

Evaluating the quality of a jury of classifiers is rather similar to the simple classifier case: each simple classifier classifies the elements of a reference set and the jury fitness is calculated from the number of positive/negative answers.

TABLE VIII
SIMPLE CLASSIFIER PARAMETER MUTATION PROBABILITIES WHEN IN JURIES

Parameter	Mutation probability
Threshold	0.15 / 8 = 0.01875
Station	0.005 / 8 = 0.000625
Comparison sense	0.005 / 8 = 0.000625
Aggregation function	0.05 / 8 = 0.00625
Earliness	0.025 / 8 = 0.003125
Aggregation Interval	0.05 / 8 = 0.00625

TABLE IX
NUMBER OF EVENT CASES AND NON-EVENT CASES OF THE TRAIN SET FOR SDAC AND USGS, WITH THE MAX FITNESS.

Data	event cases	non event cases	max fitness
SDAC	19	185	375
USGS	64	335	975

The implemented evolutionary algorithm for the optimal juries construction has a population of juries instead of simple classifiers. Each individual of the population is a vector of simple classifiers. The amount of classifiers per jury is a parameter that doesn't vary along the evolution process. The juries of the initial population are made of totally random simple classifiers. The mutation operator implemented for this case introduces variation to a jury by executing on each of its members the mutation logic used in the simple classifier case. In order not to change many of the members of a jury in a single mutation operation, the classifier parameters mutation probability have been reduced as shown in table VIII.

The probability that a particular 30 members jury mutates in a generation is 0.67.

Crossover is classifier-wise, i.e., it recombines classifiers between two juries, in the 1-point crossover way, and it doesn't recombine internal parameters of classifiers.

Elitist replacement was also used for the classifier jury evolution case.

IV. EXPERIMENTAL RESULTS

Every experience presented in this section was run 50 times with a different random number generator seed in order to get a valid estimator of the behavior of the algorithm. Unless stated differently, the size of the population is set to 100, the crossover rate is 0.5 and the evolution goes for 5000 generations.

The optimal fitness values are given in table IX.

A. Simple classifiers

As expected, simple classifiers have poor results, on both rivers. For instance, with the earliness of 60 minutes, on USGS data, the average fitness over 50 runs is 603.4, which is far below 975. It is the same for SDAC data where simple classifiers fitness reach, on average 178.48, to be compared with the max fitness value, 375. If the number of generations is increased, there are only few fitness improvements, as it can be observed on figure 3. Analysis of the details shows that there

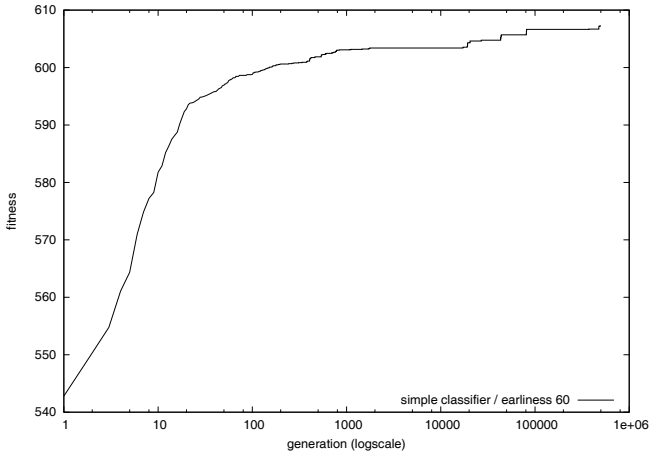

Fig. 3. Evolution of the average fitness over 50 runs on the USGS data with min earliness set to 60.

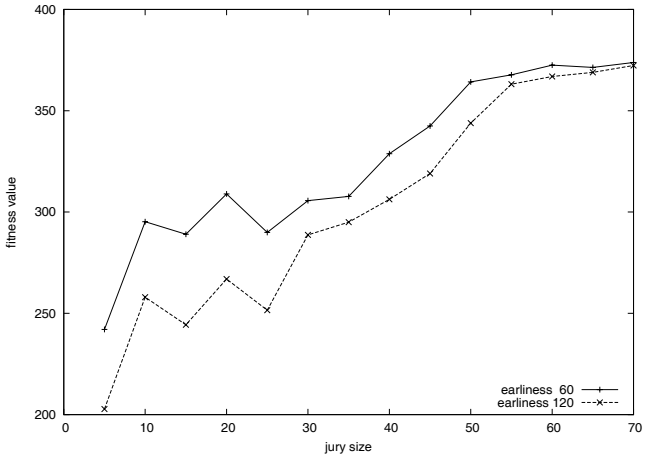

Fig. 4. Fitness value vs. jury size with different min earliness for SDAC data.

are on average 32.66 (about 8%) cases that are unclassified on the train set. This proportion of unclassified cases rises to 10% on the validation set. This is an explanation of the low results of the simple classifier model.

B. Classifier juries

On figures 4 and 5, we plot the fitness of the jury according to its size, for different earliness value. As expected, the higher the earliness is, the lower are the performances.

Examining evaluation detail for the USGS data with an earliness set to 360, for jury size superior to 10 there are no more unclassified cases on the learning set, and very few for the validation set (the proportion is 0 for size above 30).

Table X shows the validation performances obtained on USGS data with several jury sizes. Except for the second column, it follows the same schema than tables III, IV and V. For each δt value, we can see that the true positive rates are significantly higher than in our first approach. For example, with an earliness of 720, values are ranging from 54.5 to 71.3 (22.6 to 48.4 for simple variable classification). The true negatives rates are a little bit lower. However, as said before, our priority is to reduce the false negative rates (maximizing TP) since the non prediction of a real flood event have more disastrous consequences than the prediction of a false flood event.

Figure 6 shows the true negative and true positive rates on validation set for USGS data with an earliness of 360. One can observe that when the jury size increases the true negative rate also increases but the true positive rate decreases. For the learning set these rates, are for the size 70 very close to 100%. Further investigations are needed to understand this phenomena.

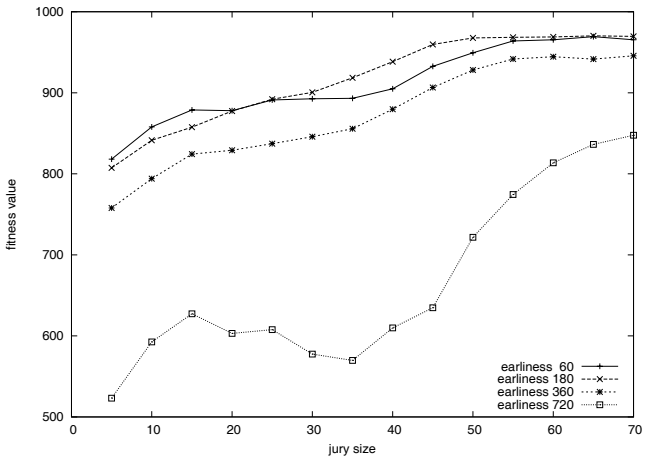

Fig. 5. Fitness value vs. jury size with different min earliness for USGS data.

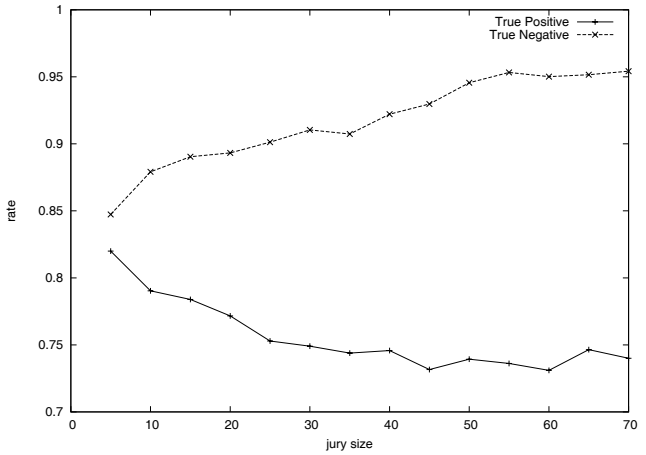

Fig. 6. True positive and true negative rate vs. jury size on the test set for the Grand-river with a min earliness set to 360 minutes.

V. Conclusion

In this paper, we have address the flood prediction issue according to a stochastic approach and by means of aggregate

TABLE X
PERFORMANCES (%) OF CLASSIFIER JURIES OF DIFFERENT SIZES, ON USGS DATA.

δt	Jury size	Test				
		FP	FN	TP	TN	Acc
60	5	12.1	8.3	91.7	87.9	88.6
	20	6.8	13.7	86.3	93.2	92
	35	5.6	17.4	82.6	94.4	92.3
	55	3.3	18.9	81.1	96.7	93.9
	70	3.3	18.5	81.5	96.7	94
180	5	13.8	12.5	87.5	86.2	86.4
	20	7.6	22.3	77.7	92.4	89.8
	35	5.8	24.9	75.1	94.2	90.8
	55	3.1	21.7	78.3	96.9	93.6
	70	3.2	19.7	80.3	96.8	93.8
360	5	15.3	18	82	84.7	84.2
	20	10.7	22.8	77.2	89.3	87.1
	35	9.3	25.6	74.4	90.7	87.8
	55	4.7	26.4	73.6	95.3	91.4
	70	4.6	26	74	95.4	91.6
720	5	29.3	28.7	71.3	70.7	70.8
	20	19	38.8	61.2	81	77.5
	35	14.6	45	55	85.4	79.9
	55	9.2	45.5	54.5	90.8	84.3
	70	11.5	41.5	58.5	88.5	83.1

variables. The objective is to be able to classify with accuracy and reliability, current situations of an hydrological system in two classes, flood alert or quiet situation, from past states of the system, given that watersheds may be very different from one to another, but are all sharing the same sparseness of target events. Our method is characterized by three parameters:

- the concept of *aggregate variables*, i.e. variables which values are aggregate over a period of time, instead of being punctual ;
- simple linear classifiers teamed in *juries* ;
- an *evolutionary algorithm* to design best juries.

We have shown that the first point allows to smooth irregularities due to uncertainty and missing data. It provides also some flexibility in time, an advantage which is much valuable since the relation between values read on a sensor and the trigger of an alert is stochastic by nature.
The second point is critical, particularly in flood prediction, characterized by the scarcity of the data due to unavailability of some sensors over large period of time. Combination of aggregate variables in a jury is thus more efficient to cover the different dimensions of the flooding system than standard models based on a reduced set of variables. Moreover, the jury model brings together simplicity and expressiveness. The complexity of the juries space, implied by the number of parameters to set and their different types (discrete or continuous) justifies the choice of the last point.

We have demonstrated that our method shows both adaptivity and robustness. Indeed the two rivers on which we made predictions have different profiles. The first river with very fast and short flood events. The second river undergoes slower and longer ones. Moreover, data are of different kinds, limnimetric or discharge values. In both cases, we succeed in finding good predictor, despite the scarcity of the data.

From the data mining point of view, future work on a data set enriched by new recorded values, should allow to better understand the mechanisms that produce such differences between learning and test performances, and to reduce them. Our experiments have shown that this difference could be important, either with our stochastic method or with classical classification methods. Obviously, true positive rate, true negative rate and earliness are antagonist objectives. It would be interesting to investigate multi-objective techniques to optimize them concurrently.

From the artificial evolution point of view, we plan to work on understanding the relation between performances and the jury size. Fitness landscape analysis will certainly highlight this question. Our method lays between parametrized models setting, where the objective is to find the parameters of a fixed model, and free models discovery, as in genetic programing, where the structure of the model has to be searched along with its parameters. The former are simpler and the latter are more adaptive. One track to explore will be toward genetic programming if we allow variable jury sizes in the population. In this case, the system can adapt its expressiveness to the problem under consideration. However, genetic programming issues such as bloat will certainly arise.

ACKNOWLEDGMENT

This work was partly funded by the General Council of La Martinique. The evolutionary experiments was run on the ORCA cluster of the Centre Commun de Calcul Intensif (C3I) of the French West Indies University, using the ParadisEO framework[3]. We also thank the Claude Emmanuel Blandin Foundation and the Regional Council of La Guadeloupe for their financial support.

REFERENCES

[1] B. E. Vieux, Z. Cui, and A. Gaur, "Evaluation of a physics-based distributed hydrologic model for flood forecasting," *Journal of Hydrology*, vol. 298, no. 1-4, pp. 155–177, 2004. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S0022169404002434

[2] Z. Liu, M. L. V. Martina, and E. Todini, "Flood forecasting using a fully distributed model: application of the topkapi model to the upper xixian catchment," *Hydrology and Earth System Sciences*, vol. 9, no. 4, pp. 347–364, 2005. [Online]. Available: http://www.hydrol-earth-syst-sci.net/9/347/2005/

[3] C. Damle and A. Yalcin, "Flood prediction using Time Series Data Mining," *Journal of Hydrology*, vol. 333, no. 2-4, pp. 305–316, Feb. 2007. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S0022169406004331

[4] C. L. Wu and K. W. Chau, "A flood forecasting neural network model with genetic algorithm," *International Journal of Environment and Pollution*, vol. 28, no. 3/4, p. 261, 2006. [Online]. Available: http://www.inderscience.com/link.php?id=11211

[5] J. R. Quinlan, *C4.5: Programs for Machine Learning*. Morgan Kaufmann, 1993.

[6] H. Shi, "Best-first decision tree learning," University of Waikato, Tech. Rep., 2007.

[7] J. Gama, "Functional trees," *Machine Learning*, vol. 55, no. 3, pp. 219–250, 2004.

[8] I. Rish, "An empirical study of the naive bayes classifier," *IJCAI 2001 Workshop on Empirical Methods in Artificial Intelligence*, vol. 3, no. 22, p. 4146, 2001. [Online]. Available: http://www.cc.gatech.edu/home/isbell/classes/reading/papers/Rish.pdf

[3] http://paradiseo.gforge.inria.fr/