Neural networks-based operational prototype for flash flood forecasting: application to Liane flash floods (France)

Dominique Bertin, Anne Johannet, Nathalie Gaffet, Frédéric Lenne

▶ To cite this version:

Neural networks-based operational prototype for flash flood forecasting: application to *Liane* flash floods (France)

Dominique Bertin[2], Anne Johannet[1a], Nathalie Gaffet[3], Frédéric Lenne[3]

[1] *Ecole des Mines d'Alès, 6 av de Clavières, 30319 Alès Cedex, France*
[2] *GEONOSIS, 650 chemin du Serre, 30140 St Jean du Pin, France*
[3] *DREAL Nord-Pas-de-Calais, Cellule prévision des crues, 44 rue de Tournai CS 40259, 59019 Lille Cedex, France*

Abstract. The *Liane* River is a small costal river, famous for its floods, which can affect the city of *Boulogne-sur-Mer*. Due to the complexity of land cover and hydrologic processes, a black-box non-linear modelling was chosen using neural networks. The multilayer perceptron model, known for its property of universal approximation is thus chosen. Four models were designed, each one for one forecasting horizon using rainfall forecasts: 24h, 12h, 6h, 3h. The desired output of the model is original: it represents the maximal value of the water level respectively 24h, 12h, 6h, 3h ahead. Working with best forecasts of rain (the observed ones during the event in the past), on the major flood of the database in test set, the model provides excellent forecasts. Nash criteria calculated for the four lead times are 0.98 (3h), 0.97 (6h), 0.91 (12h), 0.89 (24h). Designed models were thus estimated as efficient enough to be implemented in a specific tool devoted to real time operational use. The software tool is described hereafter: designed in Java, it presents a friendly interface allowing applying various scenarios of future rainfalls, and a graphical visualization of the predicted maximum water levels and their associated real time observed values.

1 Introduction

Flood forecasting in populated areas is a major challenge for early flood warning systems. For the watershed under consideration in the present paper, heterogeneity of geology and contrasted relief makes physical models difficult to calibrate. Therefore, machine-learning models based on past flood measurements in the same watershed are attractive alternatives.

After a presentation of the original flood vigilance signal investigated in the present paper, the *Liane* watershed whose floods are famous from more than one century, is described along with the database. In the subsequent section, after the presentation of neural network modelling for flood forecasting, and description of the model design methodology, with emphasis on variable and model selection by cross-validation, training and regularization, and independent testing, two candidate models for nonlinear dynamic process forecasting are presented: recurrent neural networks and feed-forward neural networks with time delays; a combination between both models seems more appropriate and is investigated in the present work.

The results are then described, and we show that satisfactory 24-hour ahead forecasts are feasible, thereby opening the way to issuing reliable population warnings in real time. In the final section the specific tool designed to efficiently help forecasters to manage decision in real-time is provided.

2 Strategy of warning of Artois-Picardie warning service

2.1 Level of vigilance

Europe is a temperate region, yet subject to water-related disasters causing casualties and material damages. Faced with this hazard, each country provides its own early warning system [1]. Facing the necessity to warn and protect the population, the French flood warning service (SCHAPI, *Service Central d'Hydrométéorologie et d'Appui à la Prévision des Inondations*) provides real-time *vigicrues* map feeding. The *Vigicrues* map displayed on the http://www.vigicrues.gouv.fr site offers four vigilance levels (http://www.developpement-durable.gouv.fr):

- Green: no particular vigilance required
- Yellow: risk of high or rapidly rising water not involving significant damage but requiring particular vigilance in the case of seasonal and/or outdoor activities
- Orange: a flood with considerable overflows liable to affect significantly daily life and security of people and property

---
[a] Corresponding author: anne.johannet@mines-ales.fr

- Red: major risk of flood directly and extensively threatening people and property.

For each area covered by the SCHAPI service, a Flood Information Rule is published. The Artois-Picardie region is monitored by the Artois-Picardie forecasting service; its Flood Information Rule [2] provides water-level values for the various levels of vigilance in each basin. This information will help assessing the relevance of forecasts with respect to practice of the local flood forecasting service.

The model performance will be assessed with respect to its ability to indicate to the forecaster the appropriate maximum level within the range of the forecast lead-time. However, the decision to broadcast a level of vigilance is made not only by monitoring the predicted discharge. Forecasters must also identify local issues that can be correlated with a specific period during the year (e.g. campsite filling rate, popular events), thus allowing them to modulate their decision criteria. In sum, the information provided by the level of vigilance forecasting will not be adequately thorough, prompting us to introduce other criteria that measure efficiency while evolving continuously.

2.2 Desired forecasts

In order to simplify the work of analysis and forecasting in real-time, forecasters of Artois-Picardie FFS (Flood Forecasting Service) use specific information: the maximum of water level at 3h, 6h, 12h and 24h lead-time. This information must be delivered by models. It can be calculated straightforwardly by models, or deduced from water level forecasting at each lead-time.

If one denotes as k the discrete present time (the instant of forecast); forecasting the water level at time $k+l_t$ (l_t is the lead time) consists in converting rainfall forecasts up to time $k+l_t$ in water level at time $k+l_t$. It can be pointed out that the information about the maximum of water level at time $k+l_t$, doesn't represent the physical behaviour of the basin, as it is shown in Fig. 1, because of the plateau that is observed: the plateau doesn't appear in water level or discharge measurements.

In the present study we focus on the design and utilization of a specific software tool for a river associated with important stakes in relation with flood hazards: the *Liane*, whose vigilance levels are provided in (Table 2).

3 Liane Basin

3.1 Presentation of the basin

The *Liane* is a coastal river of 35 km long situated in the North of France. The river begins at 101 m upper sea level at *Quesques*. Its outlet is the coastal city of *Boulogne-sur-Mer* (130,000 inhabitants in conurbation) that represents major stakes regarding flood hazards. The area of the basin is 244 km^2; it is principally composed of impervious soils except at the upper part where one can find limestone escarpments exceeding 200 m elevation.

The mean slope of the basin is 2,8%, and can reach 6% upstream (Figure 2) [2].

Upstream, the watershed is covered by forests and grasslands. Downstream the river crosses more urbanized zones and ends up in the Channel at *Boulogne-sur-mer*. It is at its downstream part that risks regarding flooding are the more important.
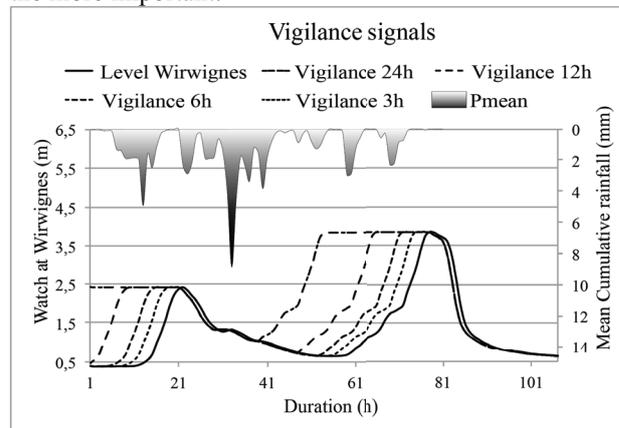

Figure 1. *Vigilance signals required by Flood Forecasting Service of Artois-Picardie at several lead-times: 3h, 6h, 12h, 24h.*
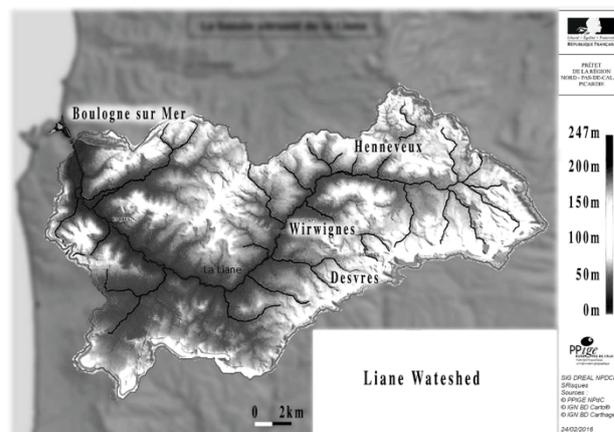

Figure 2. *Liane Watershed.*

3.2 Weather

Weather is oceanic with a mean temperature of 10°. Snow is marginal. The upper part of the watershed constitutes an anomaly considering mean precipitation because of the altitude gradient (rise of 220 m in 40 km long). Mean yearly rainfalls evolve from 750 mm near the cost to 1000 mm at the upper part of the basin. During summer the basin can receive important rainfalls during storms. Intensity can reach 30 mm/h.

Rainfalls are measured thanks to 3 rain gauges: *Desvres*, *Henneveux* and *Wirwignes*. *Wirwignes* is the outlet considered in this study; it is also a limnimetric station.

3.3 Database

Database used in this work includes hourly measurements of water level at *Wirwignes* and rainfalls at *Desvres*, *Henneveux* and *Wirwignes* from 1983 to present day. Model design was done with data up to 2012. The model is currently in operational working in FFS *Artois-*

Picardie. 4 major events can be identified in the database (Table 1). They all correspond to orange vigilance level. The more intense event is the event of November 2012. It reaches 4.36 m at *Wirwignes*. The red vigilance level is not currently defined. This event has been satisfactorily predicted, during the flood, as shown on the vigicrues map (Figure 3.).

Event	Water level Wirwignes	Vigilance level	Return period
28/10/1981	4,18 m	orange	$\simeq$ 10 years
01/11/1998	4,32 m	orange	$\simeq$ 12 years
21/11/2000	4,16 m	orange	$\simeq$ 10 years
02/11/2012	4,36 m	orange	$\simeq$ 12 years

Table 1. *Higher events of the database (30 years).*

3.4 Vigilance levels

Liane inundations are well known from several centuries, and Maurice Champion yet evokes this hazard in 1859 [3] specifically for low land inundation downstream of the basin. Due to numerous hydraulic settlements, downstream *Liane* river flows currently in artificial riverbed, but is yet subjected to frequent floods causing inundations. For this reason, and also because of the presence of canal lock at the outlet, the gauge station of *Wirwignes*, in the centre of the basin, is targeted for the vigilance definition because downstream water level would be difficult to manage and predict. Vigilance levels at *Wirwignes* gauge station are reported in Table 2. However, another gauge station downstream *Wirwignes* is working since 2012. It will help refine quantification of the risk regarding downstream stakes.

Water level *Wirwignes*	Vigilance level
Under 2.7	Green
3.1 - 3.9 m	Yellow
Over 4.1 m	Orange
Not defined	Red

Table 2. *Vigilance levels for the Liane at the station of Wirwignes.*
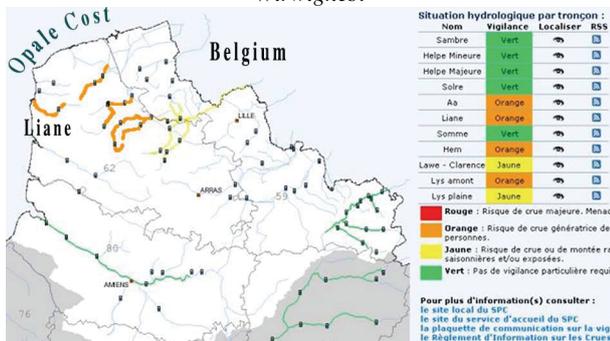

Figure 3. *Vigicrues map of the flood of the Liane (2 November 2012).*

3.5 Available models

Forecasts could be performed using several models [4]:

- two abacus, one from SPC, one from SCHAPI for 24h vigilance (too difficult to be applied in operational conditions),

- several ARMAX models at 2h lead-time, without forecast of rain, visualized by the SOPHIE platform,
- a set of GRP models, visualized through the SOPHIE platform, using future rainfalls. Several models can be run with various lead-times at the same instant.
- four neural networks models at following lead-times: 3h, 6h, 12h, 24h.

Synthetically, as explained in [4], the abacus used in real time can't take into account the past rainfalls, it is thus better at the beginning of the event. ARMAX models don't take into account future rainfalls, they can also be used for very short term forecast (2h).

In 2012 GRP was not used in real time, simulations were done after the flood.

Four models based on neural networks were shown efficient [5]. For this reason, the need of real-time efficiency during flood events and the update to a longer database a new design and the development of an *ad hoc* software tool was required. The new design of neural networks models and the description of the operational tool are the aim of this paper.

4 Neural networks for flood forecasting

4.1 General issue

Artificial neural networks are statistical black box models that use input-output measurements to identify nonlinear functions of a system [6]. Basics about neural modelling can be found in [7], only specific information, mandatory for a comprehensive presentation of this study will be provided hereafter. The chosen model is the multilayer perceptron because of its properties of universal approximation [8] and parsimony [9].
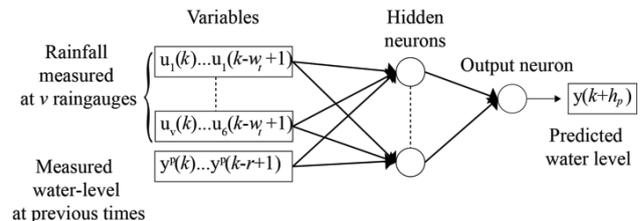

Figure 4. *Multilayer perceptron. Neurons are symbolized by circles and input variables by squares.*

The universal approximation is the capability to approximate any differentiable and continuous function with an arbitrary degree of accuracy. In this study, the multilayer perceptron is both a feed-forward and a recurrent model. The feed-forward model is widely used; it is a finite impulse response model. The recurrent part allows to better identifying the internal state of the basin [10, 11]. It corresponds to the infinite impulse response part of the model.

Designing a multilayer perceptron consists mainly of selecting input variables and the number of hidden neurons. This determines the number of parameters mechanically; model complexity increases with the number of parameters. The general equation of the predictor calculated by the feed-forward multilayer perceptron is the following:

$$y^{k+l_t} = g_{NN}\left(y_p^k, ..., y_p^{k-r+1}, \mathbf{u}^k, ..., \mathbf{u}^{k-w+1}, \mathbf{C}\right) \quad (1)$$

where the estimated value of the output at the discrete time $k+l_t$ is y^{k+l_t}, the observed value of this variable at the current time is y_p^k (at present), the input vector is $\mathbf{u}^k$, the nonlinear function implemented by the neural network is g_{NN}; w and r are the widths of windows used to apply the input time-series, they are linked to the length of the vectors of input variables $\mathbf{u}$ and $\mathbf{y_p}$; $\mathbf{C}$ is the matrix of parameters of the model, also called "weights".

It is also possible to use the recurrent model, which makes forecasts thanks to its own estimation of the water level:

$$y^{k+l_t} = g_{NN}\left(y^k, ..., y^{k-r+1}, \mathbf{u}^k, ..., \mathbf{u}^{k-w+1}, \mathbf{C}\right) \quad (2)$$

with the same notations than previously.

One can note that the only difference with the previous equation (1) is that the calculation of the model depends only on its previous outputs, and doesn't necessitate observations anymore. The feed-forward model is well known to be generally the most efficient model; nevertheless, it was shown by [11] that the recurrent model generally better represents the dynamic of the system.

As statistical models, neural networks are designed in relation to a database. This database is usually divided into three sets: a training set, a stop set, and a test set. The training set is used to calculate parameters through a training procedure that minimizes the mean quadratic error calculated on output neurons. The training is stopped by the stop set (usually called validation set, cf. Sect. 4.2), and model quality is estimated by the third part of the database: the test set, which is separate from the training and stopping sets. The model's ability to be efficient on the test set is called *generalisation*. However, it was shown that the training error is not an efficient estimator of the generalisation error: the efficiency of the training algorithm makes the model specific to the training set. This specialisation of the neural network on the training set is called *overfitting*. Overfitting is exacerbated by large errors and uncertainties in field measurements: the model learns the specific realization of noise in the training set and could thus be unable to generalize. This major issue of neural network modelling is called bias-variance trade-off [12]. Usually regularization methods are used to avoid overfitting; to this end, two regularisation methods were used in this study.

4.2 Regularisation methods

In the context of this study, the goal of regularisation methods is to minimize output variance. To this end, cross-validation [13, 14, 15] was used to empirically select input variables and the number of hidden neurons. Cross-validation thus minimizes model complexity and therefore output variance [16].

Another regularization method is commonly employed: early-stopping [17]. This method stops training before overtraining occurs. A dedicated set,

called a stop-set, is considered separately from the database. During training, the chosen quality criteria is calculated on both the training set and the stop set. When it is observed that the quality criteria always improves on training set but worsens on the stop set, which means that the model begins to not be able to generalise, training is stopped. Early stopping is thus a method to prevent the model to train too much.

Working on flash floods of the *Lez* hydrosystem (Southern France) [15] concludes that early stopping used in conjunction with cross-validation was efficient. We thus adopt this way to prevent overfitting.

In the current study, parameters are iteratively calculated using the Levenberg-Marquardt algorithm [18].

It is well known also that model performance depends strongly on the parameters initialisation before training. To define a reliable simulation independent from the initialisation, [19] proposed to establish an ensemble of 50 models trained from different initialisations. The output is calculated at each time step by the median of the 50 outputs. It was shown that, applied to the database of the *Liane*, cross-validation and early stopping worked well; it was thus unnecessary to build an ensemble model.

4.3 Complexity selection

In order to get the best of the model in generalization and taking into consideration the bias-variance dilemma, the complexity, which is mechanically linked with the number of parameters of the model, must be rigorously adjusted. It consists in selecting the set of variables and the number of hidden neurons providing the best results in validation phase. This selection can be done thanks to cross-validation, in a rigorous and systematic method, as extensively explained in [14,15].

After the model design, final performances must be assessed on a dataset independent of the training and stopping sets: the test set. Contrarily to the statement, sometimes found in the literature, stating that neural networks models "are prisoners of their training set"; neural networks rigorously designed can generalize satisfactorily to events or behaviour out of the range of the training or stop set [21].

4.4 Uncertainties drawing

Because of the important noise and uncertainties present in hydrologic data, it is specifically important to take into account the estimation of uncertainties in the result of modelling and forecasting. This estimation will be required in the future version of the vigicrues map called vigicrues 2. Usually one has to distinguish uncertainties coming from data and coming from modelling. Artigue et al. in [11] showed for Mediterranean flash-floods that the neural network model doesn't exacerbated uncertainties due to rainfalls: the noise artificially added in the rainfalls variables is transferred to the forecast without amplification, nor attenuation. Regarding the uncertainties linked to the model, another approach was proposed in [21]. It is possible to build an ensemble model composed of, for example 100 models, differing only by the parameters

initialization before training. During real-time forecast the ensemble is run and at each time step the max and min of the forecast is chosen. Such max and min values determine an envelope that visualizes the uncertainty of modelling. In the present study, 100 models were taken and the extreme values max and min were removed from the envelope, so only the second value from the top and the bottom were taken into account. This allows thus to visualize if the forecast has a great dispersion depending on the model selection. Envelopes can be seen on Fig. 9)

4.5 Quality criteria

Several criteria can be used to quantify the quality of the model. The Nash criterion [22] is often used in the field of hydrology; it corresponds to the R² determination coefficient, i.e.:

$$R^2 = 1 - \frac{\sum_{k=1}^{s}(y^{k+l_t} - y_p^{k+l_t})^2}{\sum_{k=1}^{s}(y_p^{k+l_t} - \overline{y_p^{k+l_t}})^2} \qquad (3)$$

with the same notation than previously (section 4.1), s is the number of observed couples, respectively simulated values targeted by the simulation, $\overline{y_p^{k+l_t}}$ is the average observed value on the n-sized sample. In this definition we took into account the lead-time l_t because the l_t-1 first values of the considered set are not taken into account (too early to be predicted).

This criterion must be close to one, which means that the predicted water-level is close to the observed water-level. A 0 value represents an average discharge equivalent forecasting whereas a negative value indicates that the forecasting provided is even worse than the simple average of the observed value during the event. Generally speaking, for flash floods purposes, a Nash criterion value greater than 0.8 is considered satisfactory. However, especially when using a feed-forward model, a risk for the model to provide a naive forecasting (when the model provides the same value at the forecast lead time as the one observed at the instant of forecasting) exists. That kind of result generally induces, for short lead times, a good value of the Nash criterion whereas the model does not bring any information. In order to assess the forecasting provided compared to the naive forecasting, the persistence criterion [23] has been defined. Usually the persistency criterion must be used to assess rigorously the forecast performances. Nevertheless in the case of *Liane* water-level forecasting, as the desired signal is not the water-level, but directly the vigilance signal (the maximum of water level l_t time steps ahead), the persistency has no interest.

In this case it is possible to use a simpler index which focuses on both value and synchronization of the peak: SP_{PD} as Synchronous Percentage of the Peak Discharge [11]. If the instant of the maximum of peak discharge is denoted as t_{peak}, SP_{PD} is computed as:

$$SP_{PD} = \frac{y^{k_{peak}}}{y_p^{k_{peak}}}, \qquad (4)$$

with the same notations as before.

SP_{PD} is expressed in percentage, if the forecast underestimates the maximum of the flood, the SPPD is lower to 100%. In the contrary case it is superior to 100%.

5 Results

5.1 Model design

In order to design a continuous model (not based on event modelling) and based on previous presentation of neural network modelling, we choose to use a model taking profit of both advantages of recurrent and feed-forward multilayer perceptrons. This model is shown in Fig. 5. Using simultaneously recurrent and feed-forward models is rarely done. The model receives as variables: (*i*) rainfall from the 3 rain gauges of *Desvres*, *Henneveux* and *Wirwignes*, (*ii*) a gaussian estimation of evapotranspiration, (*iii*) recurrent (previous estimated water level), and (*iv*) present measurements of water level at *Wirwignes*. We choose to use a rough estimation of evapotranspiration (gauss curve) as it was shown by [24, 25, 26] that this signal has significant efficiency for reservoir models as well as neural networks models.
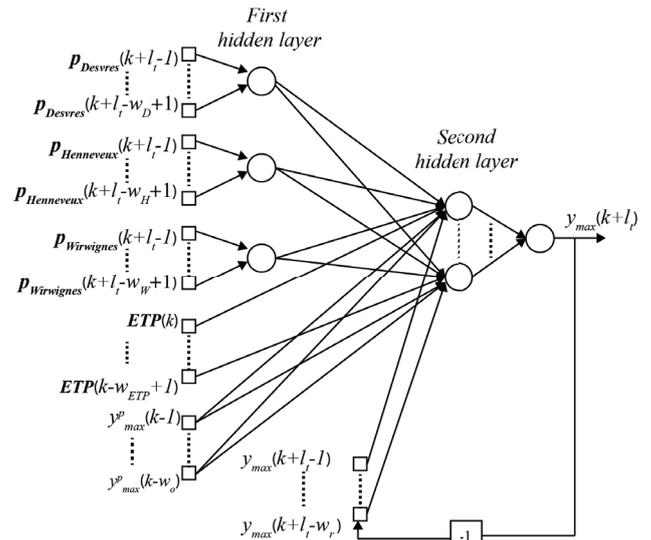

Figure 5. *MLP inspired model for Liane water level forecasting.*

Moreover the used architecture was not exactly a multilayer perceptron (MLP). Indeed, because of the modelling time step (1 hour), and the long lead-time (24h) the rainfall window widths must be long, increasing thus mechanically the number of parameters. As an uncontrolled rise of the number of parameters should worsen the ability to generalise (remember bias-variance dilemma in section 4.1), it was necessary to constrain the number of parameters to be as low as possible. To this end we introduced a linear neuron connected to rainfall data in order to diminish the number of connections

between the input layer and the second hidden layer of the Fig. 5. These supplementary neurons are represented in Fig.5 in the first hidden layer.
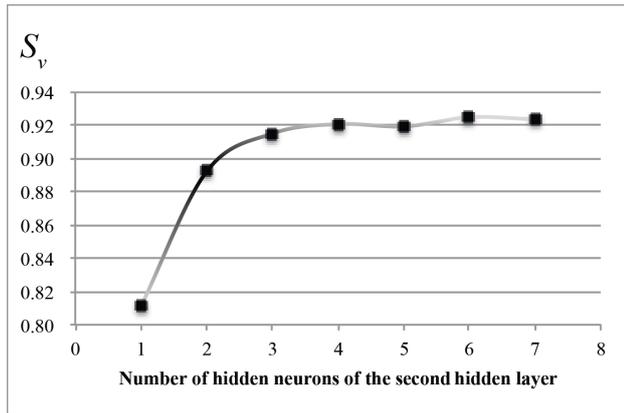

Figure 6. *Evolution of the cross-validation score versus the number of neurons of the second hidden layer. The best value maximizes the Nash-based score forecasting. In this case, the best $h_n=6$.*

As presented in Sect. 4.3, model selection is done using cross-validation and early stopping. During cross-validation we choose to use a cross validation score based on Nash criterion. The score (S_v) is simply the average of each score calculated for each validation set. Each validation set comprises 1 year. For database of 19 years, we have thus 17 validation sets of 1 year each, 1 stop set of 1 year and 1 test set equal to the year 2012 that comprises the most intense event of the database. Selected values of various window-widths and hidden neurons numbers are provided in Table 3.

One can note that the complexity of the model is moderate (small number of hidden neurons).

To make the model assessment more reliable on the most intense events of 2012, model selection was done without this year (blind assessment).

In order to illustrate the way of selecting variables and hidden neuron numbers, we represent in Fig. 6 the evolution of the cross-validation score during the selection of the number of hidden neurons of the second layer of hidden neurons, for the lead-time of 12 hours.
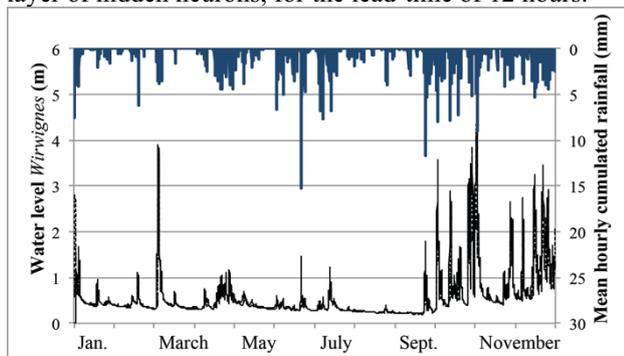

Figure 7. *Forecast for the year 2012 at the station of Wirwignes. Lead-time 24h. The observed vigilance signal is in solid line and the predicted one in dotted line. Both are quite superposed.*

It can be noticed that the highest value of the cross validation score is 0.923 for 6 hidden neurons. This value

was thus chosen, and was indicated in the Table 3 synthetizing the selected architectures, for the 4 lead-times.

l_t	w_D	w_H	w_W	w_{ETP}	w_o	w_r	h_n
3h	11	8	11	2	1	6	4
6h	2	12	12	2	1	7	4
12h	4	4	3	1	1	4	6
24h	5	5	1	2	1	2	7

Table 3. *Result of the complexity selection for each lead-time (l_t). The role of each hyper-parameter can be found on Fig. 5. For example w_D is the length of the sliding window of Desvres rainfalls; h_n is the number of neurons of the second hidden layer.*

5.2 Validation

The validation of the model can be done in considering cross-validation scores. Indeed these scores measure the efficiency of the model in situation of validation: neither in training, neither in stopping situation. It can be seen on Fig. 6 that these scores are very good for the selection of h_n, that occurs at the end of the process.

5.3 Efficiency on 2012 year

Let us remember that the test set is the 2012-year: the year including the highest event of the database. This allows to both evaluate the quality of forecasts, and to verify that the neural network model is able to generalize the learnt behaviour on the most intense event.

This can be verified visually by drawing the hydrograph of the year 2012 for the 24 h lead-time (Fig. 7). One can see that both curves are quite superposed. These satisfying forecasts are attested by R^2 scores presented in Table 4. It appears that they exceed 0.89 for all lead-times, yearly or at the level of event, allowing thus to predict the good vigilance level, 24h in advance.

Nash criterion	3h	6h	12h	24h
Year	0.99	0.98	0.96	0.94
Event	0.98	0.97	0.91	0.89

Table 4. *Nash scores on the test set.*

In order to evaluate more accurately the quality of the forecasts, hydrographs focused on the event of 2 Nov. 2012 are given in Fig. 8.

5.4 Available models

Comparison with other models applied to *Liane* basin by SPC Artois-Picardie is not straightforward because of their differences in use. Set of GRP models were run and provided forecasts for several lag-times using future observed rains (as well as neural networks model). In the Return of Experience document of the 2012 floods [4], several values of *SPPD* (see eq. 4) can be calculated and are provided in Table 5. It is not possible to compare R^2 scores because NN models and abacus provide vigilance signals while SOPHIE and GRP provide hydrographs. The *SPPD* score makes comparisons possible as it takes

into account only the forecast of the peak amplitude. This information is available for all models. Only SOPHIE, abacus and an old version of NN models were available in real-time during the flood, but the return of experience integrated forecasts provided by GRP. In the present paper we add also forecasts of the new NN model implemented in the LianePlayer©.

Lead-time/SP$_{PD}$	GRP	Neural networks	SOPHIE	Abacus
2h	86%	-	94%	-
3h	-	93%	-	107%
5h	70%	-	-	-
6h	-	99%	-	126%
7h	77%	-	-	-
9h	65%	-	-	-
12h	-	83%	-	96%
15h	69%	-	-	-
21h	74%	-	-	-
24h	-	89%	-	103%
26h	74%	-	-	-
33h	69%	-	-	-

Table 5. *SP$_{PD}$ scores on the test set (event of 2-3 November 2012).*

For long-term vigilance (24h) abacus works as well as NN model. Roughly the neural network models works better for smaller lead-times; this is not the case of the abacus which has bad forecast at 6h lead-time. Regarding GRP, it doesn't provide good estimation of the peak with a good synchronization for this event. Looking at hydrographs in [4], one can note that the forecasts peak occurs 5h in advance compared to the real peak. The max value of the peak is thus better, leading to peaks between 70% to 86% of the observed peak.

For this specific major event, the neural network model seems thus to provide useful forecasts.

5.5 Discussions

The event of 2 November 2012 is the major event of the database; it follows the event of 30 October that was the second greater event of the database. This specific configuration induced difficulties for estimating the soil moisture between both events; indeed the HU2 index is not available at the good temporal resolution. Also, as the flood event lasted a long time with several rain events, abacus was not considered as a reliable model; nevertheless one can note that it was very accurate at the
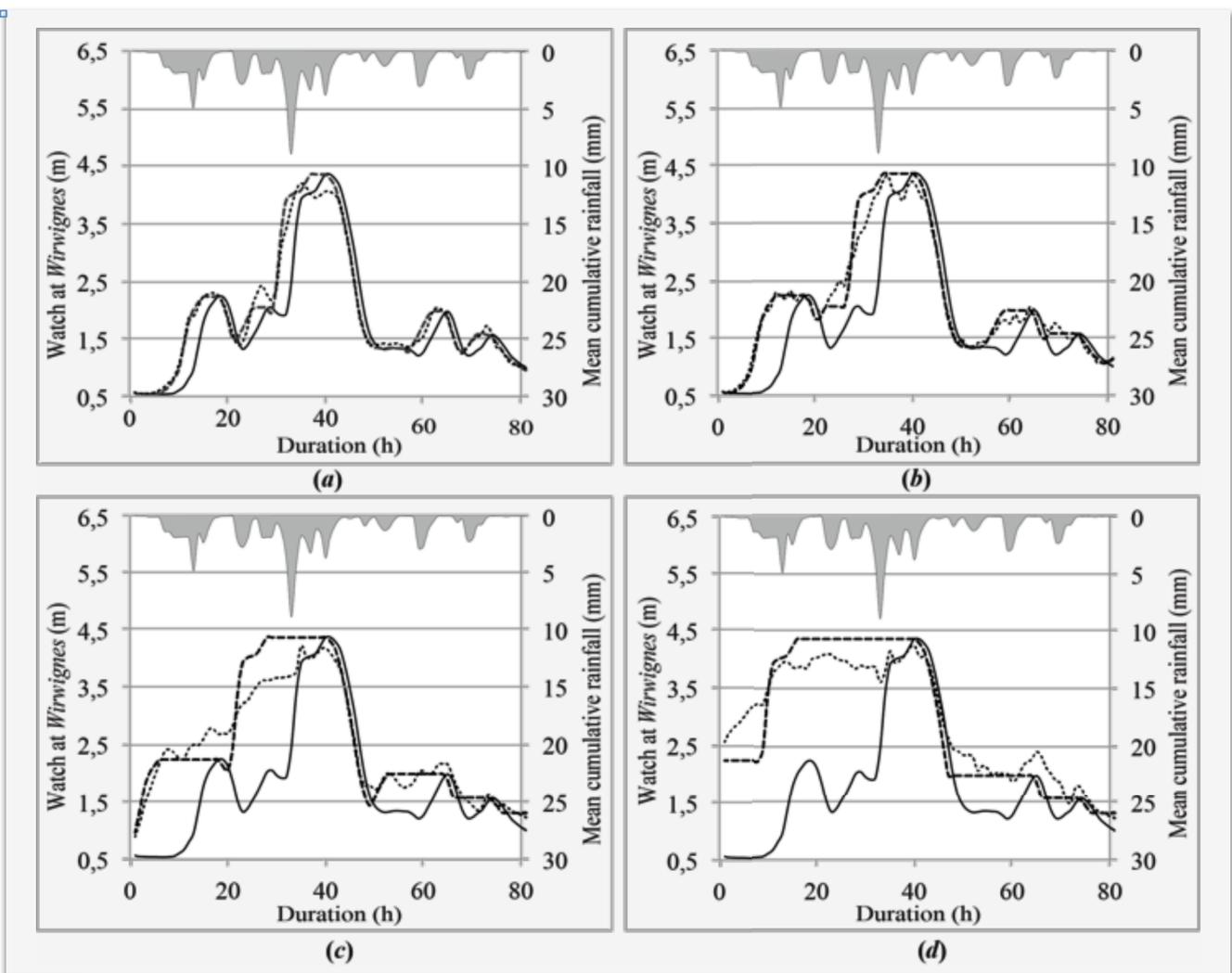

Figure 8. *Forecasts of the flood of 3 November 2012: (a) 3h lead time; (b) 6h lead time; (c) 12h lead time; (d) 24h lead time. Measured water level (solid line), ideal forecast (dashed lines), forecast (dotted line).*

beginning of the event, before the rain increase. Logically it fails for the 6h lead-time due to the rise of water in conjunction with a new rain impulse.

Regarding GRP, forecasts were not good in amplitude and had an important advance (generally 5 hours), this could be due to the calibration of the model that was done
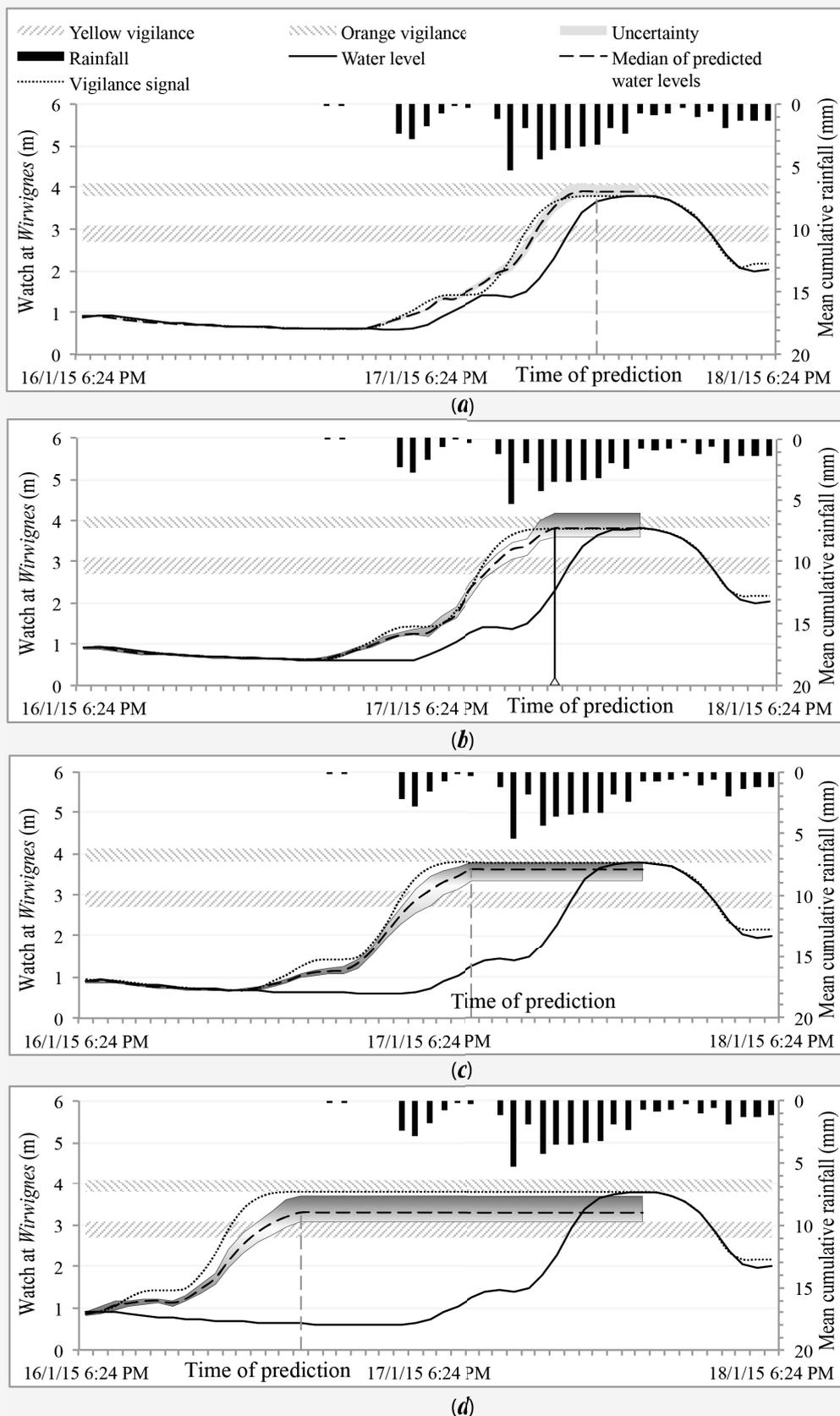

Figure 9. Forecasts of the flood of 16-17 January 2015: (*a*) 3h lead time; (*b*) 6h lead time; (*c*) 12h lead time; (*d*) 24h lead time. Estimated uncertainties due to the model are shown in grey-degraded envelope.

on 1997 without recent important events.

Neural networks models provided reliable forecasts as shown in Fig. 8 and Table 5. For this reason an operational tool was investigated in order to calculate and visualize vigilance signals and their associated uncertainties.

6 Operational tool design and implementation

6.1 Specifications

Specifications of the prototype were:

- Visualize several curves: (*i*) observed water level, up to the instant when the forecast is done, (*ii*) the maximum of water level l_t hours before, (*iii*) envelope of uncertainty associated to each forecast, vigilance levels yellow and orange, (*iv*) rainfalls, (*v*) the current instant when the forecaster uses the software.
- Be able to try several scenarii of future rainfalls by feeding manually the software. These future rainfalls are visualized continuously with the actual ones.

6.2 Implementation

The software LianePlayer© was implemented in Java in order to be able to be run on any platform. It follows the specifications, and the interface is shown in Fig. 10 thanks to a screen shot. One can distinguish several areas:

Zones 4 and 5 allow visualizing outputs of several models (each having a different lead-time) and the associated uncertainty. The way to visualize uncertainty is illustrated in Fig. 9, on the event of January 2015 during real-time utilization. Several scenarii can be compared (loaded from a file or entered with the user interface). Models can be provided in the RNFPro format (the software used to design models) or in a specific text format (java like formula). Each run configuration includes several models, each one for a different horizon.

In the screenshot (Fig.10) one can see in 1), the observed inputs loaded from a file and also updatable by the user. When they come from a file, missing data are replaced when possible by the average value from the other rainfall stations or interpolated for the water level. These modified values are highlighted in the table and a tooltip indicates which operation has been done. As well user modified values are highlighted the same way with different colours. In 2), the table shows the values computed by the models. In 3), it is possible to setup a rainfall scenario for which the outputs are automatically computed and displayed in the table and the charts. In (4) and (5), charts display outputs at a given time that can be chosen with arrow keys or a table selection. It shows also the vigilance levels and can display vertical bars when missing values are replaced.

This tool is used in real time from the beginning of the year 2016, and is tested in less convivial form from the beginning of 2015.
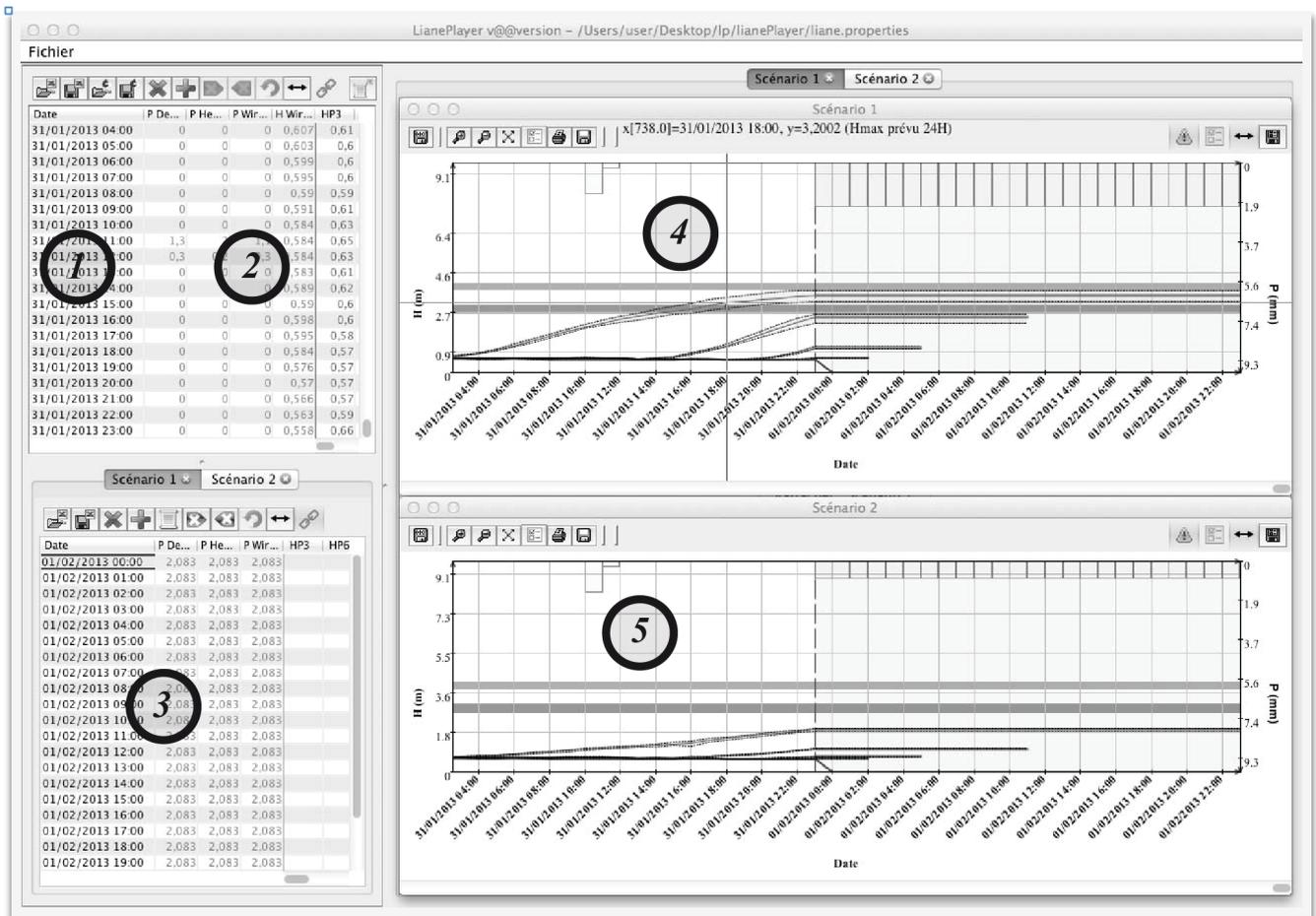

Figure 10. *Screenshot of the LianePlayer real-time software tool.*

7 Conclusion

Forecasting floods in populated areas is an important and difficult task. However, financial losses related to floods have made their study and forecasting a very challenging concern. For this reason research is active regarding flash floods, especially for Mediterranean regions, and the lack of knowledge on the processes operating during these catastrophic events persuaded SCHAPI to investigate black-box models as neural networks. In this paper, same argument incited to study flood forecasting of *Liane*, a small river of the North of France, but known for its important floods. To do this, we have evaluated one original type of model based on machine learning: a neural network combining both recurrent and feed-forward models. The first category is seldom used in hydrology; it represents models using previous estimated water level values, while the second belongs to feed-forward models using previous observed water level values. The model was assessed, in four versions corresponding to four forecast lead-times and in comparison with three kind of other models: abacus, GRP and ARMAX models. The forecast signal was not water level but a specific vigilance signal based on forecast water level. It appeared then that models provided very good forecasts up to the response time assuming that future rainfalls were as good as observed rainfalls.

A rigorous variable selection process and an accurate application of regularisation methods (early stopping, cross-validation) have highlighted one more time the ability of neural networks to model nonlinear recurrent systems such as rapid basins. Their parsimony is highly valued in the context of flood forecasting, as characterised by a poorly known hydrological context.

As exhibited in the literature, the feed-forward model is very efficient. Applied to the *Liane* basin, it yields efficient forecasts on major events, tested up to a 24 h horizon.

Thanks to these results a software tool dedicated to predict vigilance signal receiving several scenarii of future rainfalls was designed and implemented. Its friendly interface will help forecaster to manage efficiently measured and artificial data as well as the various responses of the model and of the real river. It will be extended to several other basins of the FFS Artois-Picardie.

8 Acknowledgement

9 References

1. Alfieri, L., Salamon, P., Pappenberger, F., Wetterhall, F. and Thielen, J. (2012). Operational early warning systems for water-related hazards in Europe, Environmental Science & Policy, pp. 35-49.
2. http://www.vigicrues.gouv.fr/ftp/RIC/RIC_SPC_AP_2014.pdf
3. Champion, M. (1859). Les inondations en France du VIème siècle à nos jours, Eds Dalmont et Dunod.
4. Lenne, F. (2013). Crues de la Liane, Hem, Aa, Lys amont et plaine de la Lys du 29 octobre au 5 novembre 2012. Retour d'expérience SPC-SCHAPI 89.
5. Azahaf, O.-S. (2007). Création de réseaux de neurones pour la prévision des crues. Stage de M2 Mathématiques Appliquées de l'Université des Sciences et Technologie de Lille.
6. Abrahart, R. J. and See, L. M. (2007). Neural network modelling of non-linear hydrological relationships, *Hydrology and Earth System Sciences*, **11(5)**, pp. 1563–1579, doi:10.5194/hess-11-1563-2007.
7. Dreyfus, G. (2005). Neural Networks: Methodology and Applications, Softcover reprint of hardcover 1st ed. 2005 edition. Springer, Berlin; New York.
8. Hornik, K., Stinchcombe, M., and White, H. (1989). Multilayer Feedforward Networks Are Universal Approximators, *Neural Networks* **2**, pp. 359-366.
9. Barron, A.R. (1993). Universal approximation bounds for superpositions of a sigmoidal function. *IEEE Transactions on Information Theory* IT-**39**, pp. 930-945.
10. Nerrand, O., Roussel-Ragot, P., Personnaz, L., Dreyfus, G. and Marcos, S. (1993). Neural networks and nonlinear adaptive filtering: unifying concepts and new algorithms. *Neural Computation*, **5(2)**, pp. 165–199.
11. Artigue, G., Johannet, A., Borrell, V. and Pistre, S. (2012). Flash flood forecasting in poorly gauged basins using neural networks: case study of the Gardon de Mialet basin (southern France), *Nat Hazards Earth Syst Sci,* **12(11)**, pp. 3307–3324, doi:10.5194/nhess-12-3307-2012.
12. Geman, S., Bienenstock, E. and Doursat, R., (1992). Neural networks and the bias/variance dilemma. *Neural Computation*, **4 (1)**, pp. 1-58.
13. Stone, M. (1974). Cross-validatory choice and assessment of statistical forecasting. *Journal of the Royal Statistical Society*, **B36**, pp. 111-147.
14. Kong-A-Siou, L., Johannet, A., Borrell, V. and Pistre, S. (2011). Complexity selection of a neural network model for karst flood forecasting: The case of the Lez Basin (southern France). *Journal of Hydrology.* **403(3–4)**, pp. 367–380.
15. Kong-A-Siou, L., Johannet, A., Valérie, B. E. and Pistre, S. (2012). Optimization of the generalization capability for rainfall–runoff modeling by neural networks: the case of the Lez aquifer (southern France). *Environmental Earth Sciences,* **65(8)**, pp. 2365–2375, doi:10.1007/s12665-011-1450-9.
16. Schoups, G., Van de Giesen, N. C. and Savenije, H. G. (2008). Model complexity control for hydrologic prediction, *Water Resource Research*, 44(12), W00B03, doi:10.1029/2008WR006836,.
17. Sjöberg, J. and Ljung, L. (1994). Overtraining, regularization and searching for minimum in neural

networks, Preprint IFAC Symposium on Adaptive Systems in Control and Signal Processing.

18. Hagan, M. T. and Menhaj, M. B. (1994). Training feedforward networks with the Marquardt algorithm, *IEEE Trans. Neural Netw.*, **5(6)**, pp. 989–993, doi:10.1109/72.329697.

19. Darras, T., Johannet, A., Vayssade, B., Kong-A-Siou, L. and Pistre, S. (2014). Influence of the Initialization of Multilayer Perceptron for Flash Floods Forecasting: How Designing a Robust Model, Granada, ITISE Conference, Ruiz, I. R., and Garcia, G. R., Eds, pp. 687-698.

20. Toukourou, M., Johannet, A., Dreyfus, G. and Ayral P.-A. (2011). Rainfall-runoff modeling of flash floods in the absence of rainfall forecasts: the case of "Cévenol flash floods". *Journal of Applied Intelligence*, **35, 2,** pp. 1078-189.

21. Kong-A-Siou, L., Johannet, A., Estupina, V. and Pistre, S. (2015). Neural networks for karst groundwater management: case of the Lez spring (Southern France). *Environmental Earth Sciences*, **74(12)** pp. 7617-7632.

22. Nash, J.E. and Sutcliffe, J.V. (1970). River flow forecasting through conceptual models part I – A discussion of principles. *Journal of hydrology,* **10(3)**, 282-290.

23. Kitadinis, P.K. and Bras, R. (1980). Real time forecasting with a conceptual hydrologic model, applications and results, Water Resources Research, **16**, n°6, pp. 1034-1044,

24. Oudin, L., Michel, C. and Anctil, F. (2005). Which potential evapotranspiration input for a lumped rainfall-runoff model? Part 1—Can rainfall-runoff models effectively handle detailed potential evapotranspiration inputs? *Journal of Hydrology* **303**, pp. 275–289.

25. Oudin, L., Hervieu, F., Michel, C., Perrin, C., Andréassian, V., Anctil, F. and Loumagne, C. (2005). Which potential evapotranspiration input for a lumped rainfall–runoff model? Part 2—Towards a simple and efficient potential evapotranspiration model for rainfall–runoff modelling. *Journal of Hydrology* **303**, pp. 290–306.

26. Kong-A-Siou, L., Fleury, P., Johannet, A., Borrell Estupina, V., Pistre, S. and Dörfliger, N. (2014). Performance and complementarity of two systemic models (reservoir and neural networks) used to simulate spring discharge and piezometry for a karst aquifer. *Journal of Hydrology*, **519(D)**, pp. 3178-3192.