Clim Dyn (2007) 28:307–324
DOI 10.1007/s00382-006-0184-y

Neural network based daily precipitation generator (NNGEN-P)

Jean-Philippe Boulanger · Fernando Martinez · Olga Penalba · Enrique Carlos Segura

Received: 13 June 2005 / Accepted: 9 August 2006 / Published online: 21 September 2006

Abstract Daily weather generators are used in many applications and risk analyses. The present paper explores the potential of neural network architectures to design daily weather generator models. Focusing this first paper on precipitation, we design a collection of neural networks (multi-layer perceptrons in the present case), which are trained so as to approximate the empirical cumulative distribution (CDF) function for the occurrence of wet and dry spells and for the precipitation amounts. This approach contributes to correct some of the biases of the usual two-step weather generator models. As compared to a rainfall occurrence Markov model, NNGEN-P represents fairly well the mean and standard deviation of the number of wet days per month, and it significantly improves the simulation of the longest dry and wet periods. Then, we compared NNGEN-P to three parametric distribution functions usually applied to fit rainfall cumulative distribution functions (Gamma, Weibull and double-exponential). A data set of 19 Argentine stations was used. Also, data corresponding to stations in the United States, in Europe and in the Tropics were included to confirm the results. One of the advantages of NNGEN-P is that it is non-parametric. Unlike other parametric function, which adapt to certain types of climate regimes, NNGEN-P is fully adaptive to the observed cumulative distribution functions, which, on some occasions, may present complex shapes. Ongoing works will soon produce an extended version of NNGEN to temperature and radiation.

J.-P. Boulanger (✉)
LODYC, UMR CNRS/IRD/UPMC,
Tour 45-55/Etage 4/Case 100, UPMC, 4 Place Jussieu,
75252 Paris Cedex 05, France
e-mail: jpb@lodyc.jussieu.fr

F. Martinez · E. C. Segura
Departamento de Computación, Facultad de Ciencias
Exactas y Naturales, University of Buenos Aires,
Buenos Aires, Argentina

Present Address:
J.-P. Boulanger · O. Penalba
Departamento de Ciencias de la Atmósfera y los Océanos,
Facultad de Ciencias Exactas y Naturales,
University of Buenos Aires, Buenos Aires, Argentina

1 Introduction

Weather risk management requires the analysis of large sets of climate observations with long history records, at least, on daily time scale. Moreover, the use of stochastic models simulating daily weather conditions with characteristics similar to past observations (Hutchinson 1986; Woolhiser 1992) makes it possible to make more accurate risk analysis than data analysis alone. Such stochastic models are valuable for many applications such as modeling crop growth, development and yields (Sharpley and Willians 1990; Hansen and Ines 2005), modeling hydrological processes and river flows (Siriwarden et al. 2002), pricing financial contracts known as weather derivatives (Caballero et al. 2001; Jewson 2004) and predicting impact of climate change (Wilks 1992; Semenov and Barrow 1997).

Abundant literature is dedicated to these stochastic models, and the reader is referred to Sirkanthan and McMahon (2001) for an exhaustive review of existing models. Briefly, Sirkanthan and McMahon (2001) de-

scribe four classes of daily rainfall stochastic models: two-part model (e.g. Richardson 1981), transition probability model (e.g. Allen et Haan 1975; Srikanthan and McMahon 1983, 1985), resampling model (Lall et al. 1996; Rajagopalan et al. 1996; Rajagopalan and Lall 1999) and ARMA (Auto Regressive Moving Average) model (e.g. Adamowski and Smith 1972; Hutchinson 1995). These models can be complemented by conditional rainfall models, which depend on weather patterns (Hay et al. 1991) or large-scale events such as the El Niño/Southern Oscillation (ENSO; see Woolhiser 1992) and/or by a multiple site approach (Wilson et al. 1992; Wilks 1998, 1999a; Charles et al. 1999).

In the present study, we will focus on the precipitation component of a two-part model (the second part deals with temperature and radiation; see Richardson 1981) designed for one station only. The precipitation component will be analyzed as a two-step class model, i.e. the model first simulates the rainfall occurrence (wet or dry day) and then the rainfall amount when it rains. Our Neural Network Based Daily Precipitation Generator (NNGEN-P) uses a collection of multi-layer perceptrons to approximate the empirical cumulative distribution (CDF) function for the occurrence of wet and dry spells and for the precipitation amounts for each month and each location. By means of these approximations and a uniform random generator, we simulate a random variable that follows the empirical PDF.

Different existing models and those we will consider for comparison to NNGEN-P are discussed in Sect. 3. In Sect. 4, we first consider the rainfall occurrence results based on 19 stations in Argentina. In Sect. 5, NNGEN-P rainfall simulation is compared to three parametric distribution models at 19 Argentine stations. The NNGEN-P performance in other regions of the world is also presented in Sect. 6. Section 7 shows that NNGEN-P is able to simulate fairly well extreme event probabilities as defined in Frich et al. (2002). Conclusions are presented in Sect. 8.

2 Data

2.1 Argentine data

Daily precipitation time series at 19 Argentine stations covering the 1959–2001 period are used in this paper. The Argentine stations analyzed are located north of 40°S and east of 67°W. The region under study is part of the La Plata basin, and forms the most important agricultural and hydrologic center of Argentina. Figure 1 shows the stations used and Table 1 gives their names, provinces, latitude, longitude and record length. The data used, provided by the respective National Weather Services, were processed to obtain consistent homogeneous databases. All stations selected have less than 10% of entire months missing for their period of record. Stations were subjected to statistical tests to check for artificial jumps, outliers and trends in the monthly series (Buishand 1982).

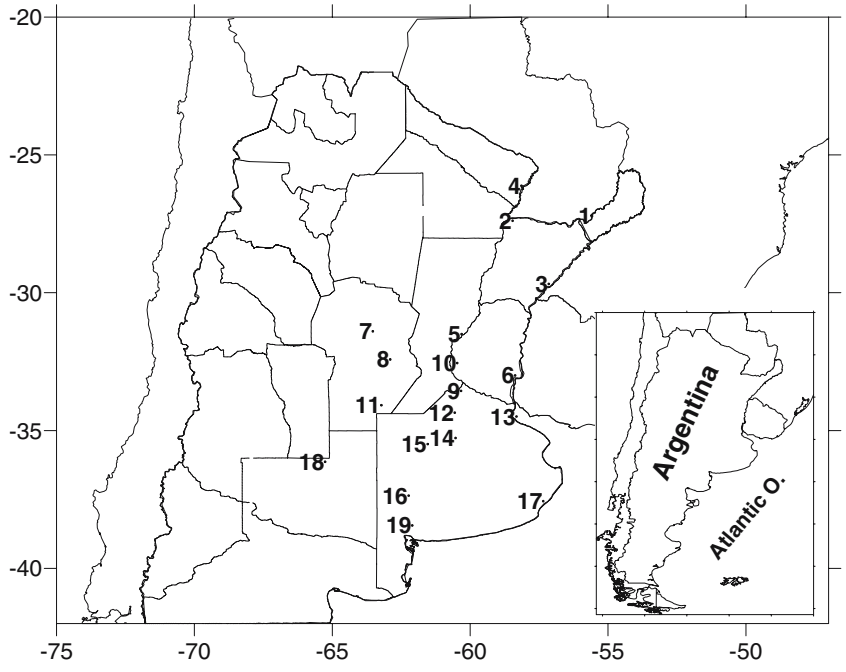

Fig. 1 Map of Argentina with the location of the 19 stations analyzed. Table 1 summarizes each station location

Table 1 List of Argentine stations

Stations	#	Provinces	Latitude (°S)	Longitude (°W)
Posadas	1	Misiones	27.22	55.58
Corrientes	2	Corrientes	27.39	58.46
Paso de Libres	3	Corrientes	29.68	57.15
Formosa	4	Formosa	26.12	58.14
Paraná	5	Entre Ríos	31.50	60.31
Gualeguaychú	6	Entre Ríos	33.00	58.37
Pilar	7	Córdoba	31.40	63.53
Marcos Juarez	8	Buenos Aires	32.42	62.90
Pergamino	9	Buenos Aires	33.56	60.33
Rosario	10	Santa Fe	32.55	60.47
Laboulaye	11	Córdoba	34.08	63.22
Junin	12	Buenos Aires	34.35	60.56
Ezeiza	13	Buenos Aires	34.49	58.32
Nueve de Julio	14	Buenos Aires	35.27	60.53
Pehuajo	15	Buenos Aires	35.49	61.54
Pigue Aero	16	Buenos Aires	37.36	62.23
Mar del Plata	17	Buenos Aires	37.56	57.35
Santa Rosa	18	La Pampa	36.13	65.26
Bahía Blanca	19	Buenos Aires	38.44	62.10

2.2 European data

In order to evaluate our method in different climates, NNGEN-P was applied to daily precipitation data available on the data server of the European Climate Assessment and Dataset (ECA&D, http://www.e-ca.knmi.nl). The ECA dataset contains daily resolution series of observations at meteorological stations throughout Europe and the Mediterranean. The series are quality controlled and individual data that are ''OK'', ''suspect'' or ''missing'' are flagged. Homogeneity testing has resulted in classification of series in ''useful'', ''doubtful'' or ''suspect''. These classes hold only for the particular time intervals for which the tests were applied. The use of the results of the homogeneity tests for selecting appropriate series is recommended. Note that the series have not been homogenized in the sense that values have not been changed. Analyzed stations are shown in Table 2.

2.3 Other data

In order to evaluate our method in different climates, NNGEN-P was applied to daily precipitation data as provided by the National Climate Data Center. We used Version 6 of dataset on line for January 1994–December 2004. Data are available via http://www.ncdc.noaa.gov. The on line data files begin in January 1994. About 8,000 stations' data are typically included in the dataset. We selected stations from United States and the Tropics. Tables 3 and 4 summarize all the analyzed stations.

Table 2 List of European stations

Stations	#	Countries	Latitude (°N)	Longitude (°E)
Dublin Airport	1	Ireland	53.43	–6.25
Madrid/Barajas	2	Spain	40.45	–3.55
Zurich-Kloten	3	Switzerland	47.48	8.53
Lisboa/Gago Coutinh	4	Portugal	38.76	–9.13
Koebenhavn/Kastrup	5	Denmark	55.61	12.66
Helsinki-Vantaa	6	Finland	60.31	24.96
Paris-Aeroport Char	7	France	49.01	2.53
Roma Fiumicino	8	Italy	41.80	12.23
Amsterdam Ap Schiph	9	Holand	52.30	4.76
London Weather Cent	10	United Kingdom	51.51	–0.11
Budapest/Pestszentl	11	Hungary	47.43	19.18
Bruxelles National	12	Belgium	50.90	4.53
Heraklion (Airport)	13	Greece	35.33	25.18
Warszawa-Okecie	14	Poland	52.16	20.96
Innsbruck-Flughafen	15	Austria	47.26	11.35
Praha/Ruzyne	16	Czech Republic	50.10	14.28
Vilnius	17	Lithuania	54.63	25.28
Tallinn	18	Estonia	59.38	24.80
Riga	19	Latvia	56.96	24.05
Bratislava-Letisko	20	Slovakia	48.20	17.20
Ljubljana/Brnik	21	Slovenia	46.21	14.48
Luxembourg/Luxembou	22	Luxembourg	49.61	6.21
Larnaca Airport	23	Cyprus	34.88	33.63

3 The two-step stochastic model

3.1 Modeling wet/dry periods

As discussed in Sirkanthan and McMahon (2001) and in Wilks (1999b), the simulation of rainfall occurrence is modeled either by Markov chains or by alternating renewal processes. Markov chains simply relate the

Table 3 List of USA stations

Stations	#	States	Latitude (°N)	Longitude (°W)
Astoria	1	Oregon	46.15	123.88
Eugene	2	Oregon	44.12	123.22
Portland	3	Oregon	45.60	122.60
Salem	4	Oregon	44.90	123.00
Reno	5	Nevada	39.50	119.78
Minneapolis	6	Mineapolis	44.88	93.22
Rochester	7	Mineapolis	44.00	92.45
St. Cloud	8	Mineapolis	45.58	94.18
Tulsa OK	9	Oklahoma	36.20	95.90
Jackson MS	10	Mississippi	32.32	90.08
Meridian MS	11	Mississippi	32.33	88.75
Atlantic City NJ	12	New Jersey	39.45	74.58
Baltimore MD	13	Maryland	39.18	76.67
New York City NY	14	New York	40.65	73.78
Philadelphia PA	15	Pennsylvania	39.88	75.23
Williamsport PA	16	Pennsylvania	41.25	76.92

Table 4 List of tropical stations

Stations	#	Countries	Latitude (°N)	Longitude (°E)
Bogota/El Dorado	1	Colombia	4.70	–74.13
Iquitos	2	Peru	3.75	–73.25
Puerto Limon	3	Costa Rica	10.00	–83.05
Asuncion/Aeropuerto	4	Paraguay	–25.26	–57.63
Manila	5	Philippines	14.58	120.98
Abidjan	6	Liberia	5.25	–3.93
Pretoria	7	South Africa	–25.73	28.18

state (wet or dry) of the current day to the states of N preceding days. The number N of preceding days considered in the model defines the order of the Markov chain. Evaluating the optimal order of the chain is a complex issue, which has been addressed using either the Akaike information criterion (AIC; Akaike 1974) or the Bayesian information criterion (BIC; Schwartz 1978). Although the first-order Markov model is the most common (Katz 1977; Richardson 1981; Stern and Coe 1984; Wilks 1989, 1992), it may not produce synthetic time series with very long dry spells frequently enough (Buishand 1978; Racsko 1991). Therefore, although the first-order model is adequate for many locations, a second- or higher order model may be more appropriate at certain locations or certain periods of the year. Considering that a major concern in developing such a model is the parsimony principle, another class of models, called alternating renewal process, has been proposed. This class of models considers the daily rainfall data as a sequence of alternating wet and dry spells of varying lengths. Most of the time the lengths of the consecutive wet and dry spells are assumed to be independent, and their respective distributions are different. Various distribution models have been proposed to fit the wet and dry spell distributions. Although the alternating renewal process should be more accurate in simulating long dry spells (Rackso et al. 1991), Roldan and Woolhiser (1982), analyzing five US weather stations, found the first-order Markov process to outperform an alternating renewal process based on a truncated geometric distribution function for wet spells and a truncated negative binomial distribution function for dry spells. Moreover, as pointed out by Sirkanthan and McMahon (2001), a potential disadvantage of this method is that it may be more difficult to handle the seasonality in the rainfall occurrence process. Finally, an important concern when focusing on daily rainfall stochastic modeling is to ensure that synthetic series simulated means and variances are also consistent with observations on monthly and annual time scales.

3.1.1 *Reference model*

In the following, the reference model is the first-order Markov process. Although high-order Markov models or either alternating renewal process may outperform the first-order Markov model, this simple model is often used and therefore offers a good reference for validating our methodology. In brief, considering the transition probabilities from dry-to-wet or wet-to-wet conditions ($P\{X_{t-1} = 0|X_t = 1\} = p_{01}$ and $P\{X_{t-1} = 1|X_t = 1\} = p_{11}$, together with the complementary probabilities defined by $p_{00} = 1–p_{01}$ and $p_{10} = 1–p_{11}$), the stochastic simulation of state X_t results from applying the following algorithm where u_t is a random number of value between 0 and 1:

$$p_c = \begin{cases} p_{01} & \text{if } X_{t-1} = 0 \\ p_{11} & \text{if } X_{t-1} = 1 \end{cases}$$

$$X_t = \begin{cases} 1 & \text{if } u_t \le p_c \\ 0 & \text{otherwise} \end{cases}.$$

3.1.2 *Neural network model*

Our model is based on the alternating renewal process, and we will consider the consecutive wet and dry spells to be independent. Our approach is non-parametric and is based on a multi-layer perceptron neural network architecture (see Appendix A). In order to simulate wet/dry sequences, NNGEN-P aims at fitting two curves. One is the cumulative distribution function (CDF) of the number of days of dry sequences. The other is the CDF of the number of days of wet sequences. The MLP architecture necessary for fitting these curves is simple and represented in Fig. 14. The input layer has only one input neuron (probability of the CDF) and one output neuron (the amplitude of the CDF associated to the probability value). The number of neurons in the hidden layer is optimized by the method.

During the training phase, the number of observations used for the MLP to learn the CDF curve is equal to the number of observations with different values. An MLP is optimized for each month at each station. In the case of wet/dry spells, the number of observations is the maximum length of the observed wet or dry spell. While dry spells can be as along as various months at certain stations, wet spells are much shorter (5 days or more depending on the station location and the month of the year). Considering such a small number of observations (critical for the training

phase), the CDF is extended up to 60 days with values equal to one. While adding observations did not affect the observed CDF, it made possible for the MLP to smoothly fit the curve. This ability of NN models is commonly known as *generalization capacity*: the curve not only interpolates (in the manner of classical methods) but also approximates the distribution smoothly.

Given the problem and the simple architecture, the method relies in optimizing the number of neurons in the hidden layer and in avoiding overfitting. With a sufficient number of neurons, the MLP can reproduce exactly all the details of any curve. A way to avoid the overfitting problem is to penalize the MLP with bigger architectures. This is done using a Bayesian approach (see Appendix B). We found the optimal number of neurons in the hidden layer to vary according to the complexity of the shape to fit. However, the number of neurons is almost always equal to 3 when fitting the wet spell distribution, and, in more than 80% of the cases, this number is lower than 8 neurons when fitting the dry spell distribution, with a mean value around 7 neurons. Larger number of neurons (up to 14) are observed in very few cases when the shape of the distribution gets more complex.

3.1.3 Wet/dry spell generation

Once NNGEN-P is calibrated at a specific station, time series are generated as follows. Imposing a dry sequence as the initial condition, a random number between 0 and 1 is generated, this number is sent as an entry to the neural network, which output is the length of the dry spell. The same process is then used to generate the length of a wet spell, etc. When the date of the modeled time series falls into another month, the neural network of that month at this station is used.

3.2 Modeling the daily precipitation

3.2.1 Reference models

Among the numerous models used to simulate precipitation in two-step stochastic models, we will consider three for reference: the two-parameter Gamma distribution (Richardson 1981; Woolhiser and Roldan 1982; Wilks 1989, 1992), the two-parameter Weibull distribution (Geng et al. 1986; Selker and Haith 1990) and the Mixed Exponential Distribution (Woolhiser and Pegram 1979; Woolhiser and Roldan 1982, 1986; Foufoula-Georgiou and Lettenmaier 1987; Wilks 1998; Wilks 1999b). At each site and for each month, we will compare our neural network-based model and the three previously cited parametric distribution models.

The probability density function for each of these parametric models is as follows (where x is the precipitation amount):

- Two-parameter Gamma distribution: $f(x) = \frac{(x/\beta)^{\alpha-1}\exp[-x/\beta]}{\beta\Gamma(\alpha)}, x, \alpha, \beta > 0$
- Two-parameter Weibull distribution: $f(x) = ba^{-b}x^{b-1}\exp\left[-(x/a)^b\right], x, a, b > 0$
- Three-parameter Mixed Exponential distribution: $f(x) = (\alpha/\beta_1)\exp[-(x/\beta_1)] + (1-\alpha)/\beta_2\exp[-(x/\beta_2)]$, $x, \alpha, \beta_{1,2} > 0$

3.2.2 Neural network model

In the present case, the MLP architecture necessary for fitting the rainfall amount CDF is simple and represented in Fig. 14. The input layer has only one input neuron (CDF probability) and one output neuron (rainfall amplitude). The number of neurons in the hidden layer is optimized by the method described in Sect. 3.1.

During the training phase, the number of observations used for the MLP to learn the CDF is equal to the number of observations with different values. An MLP is optimized for each month at each station. The number of observations is usually in the order of 5–15

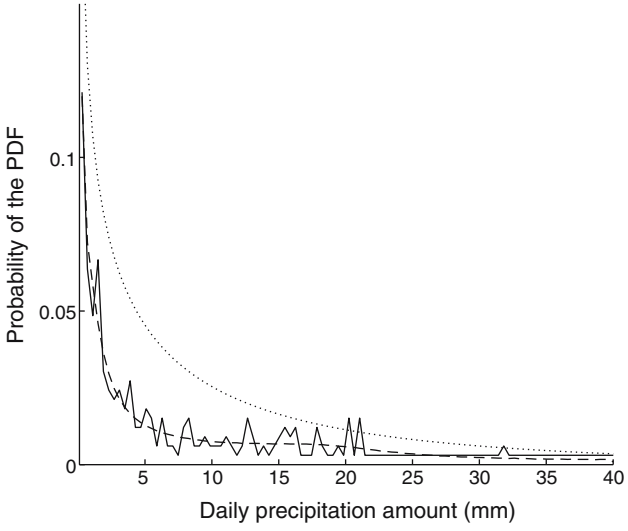

Fig. 2 Probability Distribution Function of the length of dry spell at Ezeiza in March. The *solid curve* is the observed PDF. The *dotted curve* is the first-order Markov model. The *dashed curve* is the NNGEN-P curve. The neural network optimized by the Bayesian approach makes possible to fit a smooth curve to the observed PDF and to avoid the risk of overfitting

per month, meaning that it can reach approximately 500 observations for 50 years of daily data. We found a minimum number of 10–15 years (around 50–100 observations) to be necessary to ensure a good fit. Here again, the ability of the NN to smoothly fit the curve (*generalization capacity*) is an advantage with respect to classical interpolation techniques.

The optimization method is the same as the one described in detail in Sect. 3.1. The Bayesian approach leads to a selection of different numbers of neurons in the hidden layer according to the complexity of the CDF shapes to fit. We found the optimal number of neurons in the hidden layer to vary according to the complexity of the shape to fit. However, in more than 90% of the cases, this number is lower than 7 neurons, with a mean value between 5 and 6 neurons. Larger number of neurons (up to 14) are observed in very few cases when the shape of the distribution gets more complex. Figures 2 and 3 illustrates how the optimized neural network makes possible to fit the observed CDF curve and to avoid the overfitting problem.

3.2.3 Rainfall generation

Once NNGEN-P is calibrated at a specific station, time series are generated as follows. During each wet spell simulated during the first step, a random number between 0 and 1 is generated, this number is sent as an entry to the neural network, whose output is the amount of daily rainfall. The same process is repeated as many times as wet days in the modeled time series during the month. When the date of the modeled time series extends into another month, the neural network of the new month at this station is then used.

4 NNGEN results for rainfall occurrence

At each of the 19 Argentine stations and for each month of the year, we computed first the transition probability of the first-order Markov chain process. Then, we computed the observed probability for a wet or dry sequence starting in a given month for a length of N days. Each probability distribution was then processed independently. Each probability distribution was transformed into a cumulative density function. Two MLPs (one for wet sequences, the other for dry sequences) were trained over the entire period for each month and each station and optimized as explained in Sect. 3. The input vector is the cumulative probability to observe a wet (or dry) sequence of N days. The output vector is the sequence length (N days).

4.1 Comparison of model fits

Figure 4 displays the PP-plot of the observed dry spell cumulative distribution function (CDF) against both the Markov and NNGEN dry spell model CDFs at the 19 Argentine stations for all months of the year. The gap observed between the two extremes of the wet spell slope occurs because many stations have short wet spells (5–6 days). In such cases, the CDF presents a rather discrete shape as the gap probability of having 1–2 day spells (50–70%) and longer ones is large (gap of 10–20% in certain cases). The PP-plot makes it easier to compare observed and model distributions as the points should be close to a straight line with slope of one and intercept of zero. Moreover, we can compute both a mean error (sum of the model-data difference) and a quadratic error (sum of the model-data squared difference). The mean error tells us whether the model is biased, while the quadratic error is an indication of the dispersion of the fit respective to the straight line. Among the 228 Argentine station-months, the NNGEN-P model tends to be closer to the straight line than the Markov model. Indeed, the all-averaged Argentine station-months quadratic error is always smaller in both wet or dry sequences. However, the NNGEN-P mean error is found to be slightly larger. Similar results were found at all other stations in the United States of America, Europe and Tropics (not shown).

4.2 Comparison of model simulations

As pointed out by Wilks (1999), weather generator simulations of observed interannual variances are highly desirable and are therefore an interesting validation test. If one computes the variance of, say, the daily precipitation in January, this variance can be written as follows: $\mathrm{Var}(P) = E(Nwd)\sigma^2 + \mathrm{Var}(Nwd)\mu^2$, where Nwd is the number of wet days per month, $E()$ is the averaging operator, Var() is the variance operator and μ and σ are respectively the average and standard deviation of non-zero daily precipitation amounts of the selected month. In this section, and considering that both Markov and NNGEN-P simulates the monthly mean values of Nwd fairly well (not shown), evaluating the model skill in simulating the interannual variance of precipitation is therefore equivalent to evaluating the model skill in simulating the variance of the number of wet days.

Figure 5 displays the comparison between the standard deviation of observed number of wet days and simulated number of wet days (assuming a simulation as long as the observations; the results may slightly

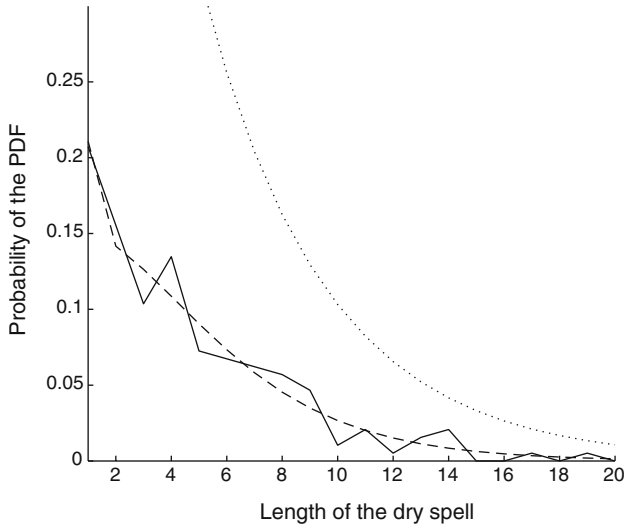

Fig. 3 Probability Distribution Function of the daily precipitation at Ezeiza in March. The *solid curve* is the observed PDF. The *dotted curve* is the Weibull model. The *dashed curve* is the NNGEN-P curve. The neural network optimized by the Bayesian approach makes possible to fit a smooth curve to the observed PDF and to avoid the risk of overfitting

vary depending on the simulation due to the stochastic nature of the weather generator). No significant differences between both the Markov and NNGEN-P models are observed.

However, if we compare the skill of both models in simulating longest dry and wet spells (Fig. 6); we observe that the Markov model shows a tendency to underestimate the longest observed dry periods and overestimate the longest wet periods. On the contrary, NNGEN-P has a much better skill in simulating both types of spells. It is remarkable that NNGEN-P never overestimates the longest dry spells, although it does not completely remove the underestimation bias in simulating the largest dry period. Here, two comments must be made. First, NNGEN-P fits fairly well the wet/dry spell CDFs. Therefore, due to its non-parametric nature, it adjusts smoothly to the observed CDFs, avoiding overfitting thanks to the Bayesian approach. As a consequence, NNGEN-P is unlikely to simulate spells much longer than the one observed. The Markov two-state probability, however, does not take into account the actual duration of a previous spell when computing whether the next day is dry or wet. Overall, in Argentina, where the probability of maintaining dry (resp. wet) conditions is close to 0.8 (resp. 0.4), and where according to the month and the station location the longest spells can be observed to be short (4–5 days) or long (15 days), the Markov model (due to its exponential shape) will tend to overestimate the short wet/dry spells, but to significantly underestimate the long wet/dry spells. Therefore, NNGEN-P should always improve the simulation of wet/dry spells. Second, the question remains of why NNGEN-P can underestimate the longest spells. The answer is in the stochastic nature of the weather generator. In one specific simulation of 40 years, the probability for NNGEN-P to simulate exactly all the same lengths of observed dry or wet spells are relatively small. The most interesting point is that the station/month, where, in Fig. 6, NNGEN-P underestimates the longest spells, change from one simulation to another confirming the stochastic explanation of the apparent bias detected by the non-zero mean error.

5 NNGEN results for rainfall amounts

5.1 Comparison of model fits

As shown in Fig. 7, the skill of the parametric functions in fitting the observed rainfall CDFs varies from one month to another or according to the station location i.e. the local climate regimes. In certain cases, we found that all the three parametric functions as well as NNGEN-P fit the data perfectly (Fig. 7a). In other cases, the differences may be very significant (Fig. 7b). Overall, unlike Gamma, Weibull or double-exponential functions, NNGEN-P did not depend either on the months, nor on the station location and always fitted the observed CDFs.

In order to demonstrate this result, we first plotted the PP-plot of the observed and modeled CDFs for Gamma, Weibull, Double Exponential and NNGEN-P (Fig. 8). The root-mean square of the error for each of the three parametric models is about ten times larger than the root-mean square error for NNGEN-P. In all cases, the absolute value of the mean error is high compared to NNGEN-P. The major strength of NNGEN-P, as already stated, is that it can fit any kind of function, and unlike the other three it is not constrained by a parametric-family shape. It can be seen in Fig. 8 that both Gamma and Weibull tend to overestimate the probability of the weak precipitation amplitudes and to underestimate the others with probabilities in the range 0.4–0.8. The Double Exponential function is actually much closer to the straight line, and it shows a much smaller error. However, while NNGEN-P shows more discrepancies for the smallest probabilities (inferior to 0.1), the fit gets better as the probability gets larger. This is coherent with the fact that NNGEN-P fits the high precipitation amount probability independently of the small amounts, contrary to the parametric functions. This result is further confirmed by Figs. 9 and

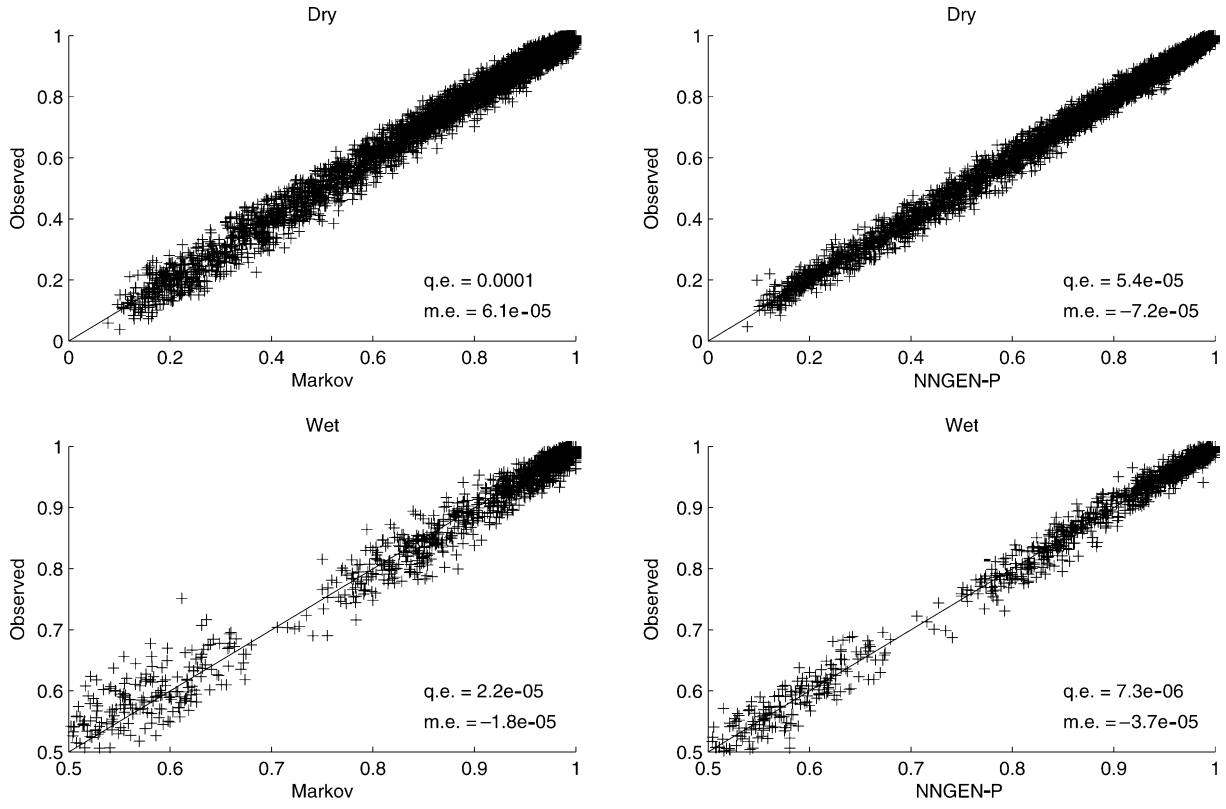

Fig. 4 PP-plot of the observed and simulated dry and wet spells. Simulated quantiles are obtained by either a two-state probability Markov model or NNGEN-P. As a reminder, a PP-plot is built as follows. To each probability of the observed CDF is associated a certain wet or dry sequence length. For each such wet or dry sequence length, we can associate a probability of the observed and CDFs. Such probabilities are then plotted as a scatter plot. Cumulative probabilities are in the range 0–1. The straight line crossing both axes at zero represent a perfect fit. For each plot, we computed a mean algebraic error and the root-mean square of the model-data error misfit

10 which displays the QQ-Plot of the precipitation amounts. The root-mean square error for each of the three parametric models is about ten times larger than the root-mean square error for NNGEN-P. The major strength of NNGEN-P, as already stated, is that it can fit any kind of function, and is not constrained, like the other three, by a family shape. In some cases, Gamma, Weibull or Double exponential functions were found to get close to NNGEN-P, but in general, their parametric shapes prevent them from fitting all the observed rainfall CDFs. Errors tend to get larger as the precipitation amplitude increases. It is worth noting that both Gamma and Double Exponential may underestimate the large amplitudes, while the Weibull function tends to overestimate them. NNGEN-P does not have any of these biases and offers a very good fit to the observed CDFs.

5.2 Comparison of model simulations

As pointed earlier, an interesting validation test is to compare the model skill in simulating the interannual rainfall standard deviation for each month at each station. We performed the analysis on simulations (Table 5) as long as observations (1959–2001). Four simulations use the Markov model to compute the wet/dry day occurrence, and the other four use the NNGEN-P component. Finally, for each case, we simulate precipitation amounts using either the Gamma, Weibull, Double Exponential and NNGEN-P models. First, it is apparent that there is no significant difference between Markov or NNGEN-P to simulate the wet/dry day condition. This result is coherent with results shown in Sect. 4 where we found that the major difference occurred in the skill in simulating the longest dry/wet spells. Second, focusing on the simulations using NNGEN-P for the wet/dry conditions, both the Gamma and Double Exponential functions tend to underestimate the interannual standard deviation (strong positive bias, in consequence the AVO is bigger). The Weibull function underestimates the interannual standard deviation of weak precipitation variability and overestimates the amplitude of the interannual variability when it is large.

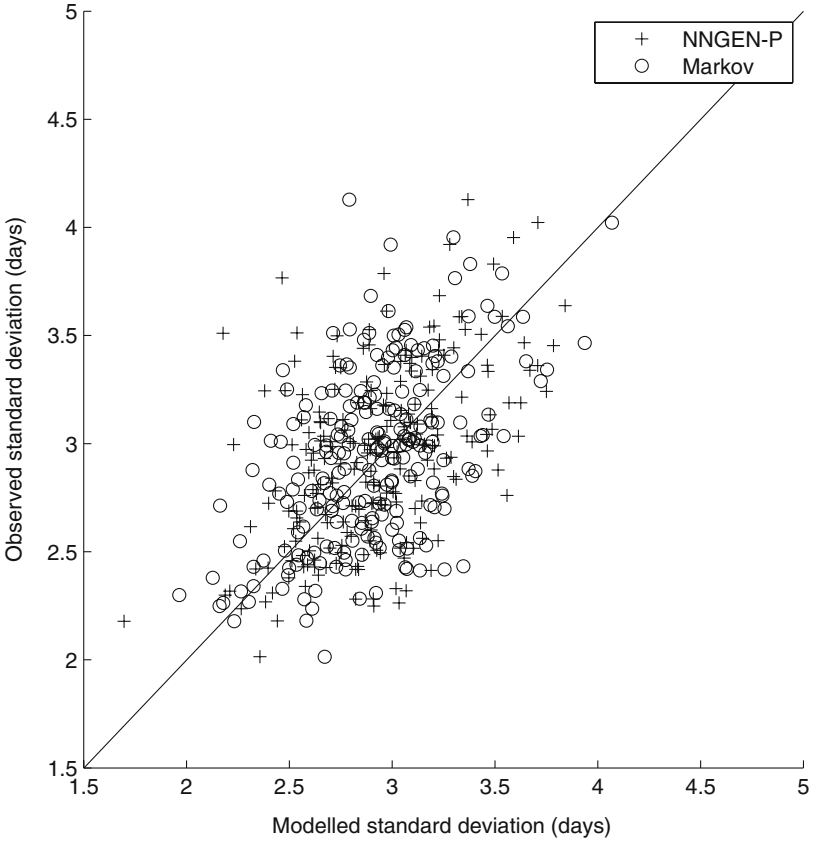

Fig. 5 Standard deviation of the number of wet days per month at each station (228 points) for both Markov (*circles*) and NNGEN-P (*crosses*). The Average Variance Overdispersion (AVO) is calculated as (Observed variance/modeled variance–1)*100%. The AVO value is 2% for Markov and 1.5% for NNGEN-P

This contrary behavior explains a small mean error (compensation of errors). However, the error remains very similar to the one of the other two distributions. NNGEN-P however has a fairly good fit between observed and simulated interannual standard deviation. NNGEN-P is slightly biased toward an underestimation of the interannual variability and a quadratic error which is much smaller than in the other three parametric functions.

A final word on model validation: we compared, as in Wilks (1999), the largest simulated amplitude with the largest observed amplitude. Taking the largest amplitude (associated to a probability of 1) to be infinity for each of the three parametric functions, the largest value is the one associated to a probability $(N–1/3)/(N + 1/3)$ where N is the number of observations used to construct the observed CDFs. Figure 11 confirms the results found previously. Both Gamma and Double Exponential functions strongly underestimate the maximum observed amplitude (strong positive bias). The Weibull function does not show such a bias, but displays a relatively large scatter on both sides of the straight line. Finally, NNGEN-P largely coincides with observations confirming the high potential that neural network architectures provide for the design of weather generators.

6 Validation against stations in different regions of the world

As stated above, the major strength of NNGEN is that it is fully adaptable to any kind of cumulative distribution function. As such, its skill in simulating daily precipitation should not depend on the station location as it is the case for each of the three parametric functions analyzed in the present paper. To confirm that point, we analyzed daily rainfall stations from different regions of the globe (United States of America, Europe and the Tropics). We found as in Argentina NNGEN-P to reproduce better the longest dry/wet spells (not shown). Moreover, it simulates well the precipitation amount distribution and to illustrate this point, we present the QQ-plots at stations in the United States (Fig. 11), in Europe (Fig. 12) and in the Tropics (Fig. 13) for the three parametric functions (Gamma, Weibull and Double Exponential) and NNGEN. As in Argentina, NNGEN-P presents a very good fit with observations characterized by a very small mean and quadratic error.

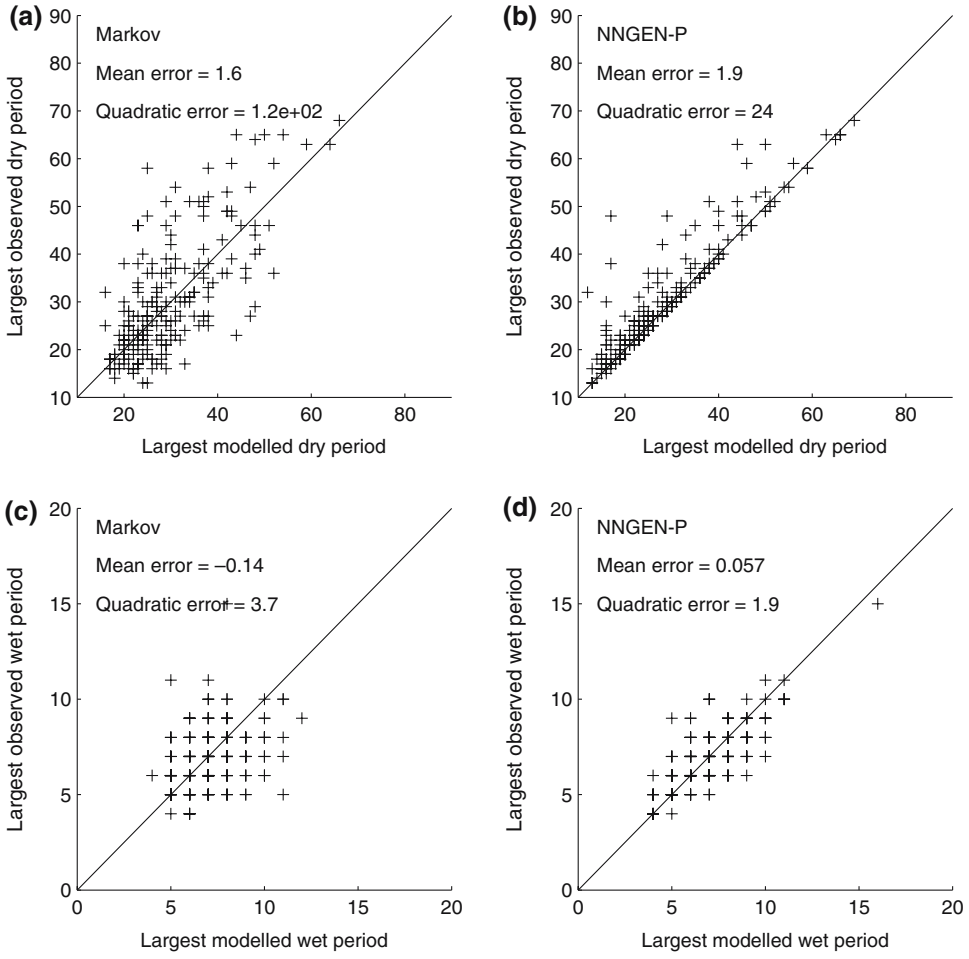

Fig. 6 Observed longest dry (resp. wet) period as a function of the modeled longest dry (resp. wet) period for the Markov model (**a**–**c**) and NNGEN-P (**b**–**d**)

7 Simulation of extreme event probabilities

Another important weather generator validation test is the evaluation of its capacity to represent extreme event probabilities. Considering that climate series are usually relatively too short to study extreme event changes, the use of a well-calibrated weather generator can potentially compensate for this by running long simulations. This is also important if weather generators are calibrated to future climate change conditions as they can provide a much more efficient tool to study extreme event changes than direct climate model outputs. In the present section, we decided to compare the skills of NNGEN-P and the three other parametric functions in simulating extreme precipitation probabilities. Thus, following Frich et al. (2002), we computed the five extreme event indices related to rainfall. These indices are:

- R10 is the number of days with precipitation ≥10 mm/day (units are in days).
- CDD is the maximum number of consecutive dry days defined as precipitation lower than 1 mm (units are in days).
- R5d is the total of the largest 5 days of precipitation (units are in 0.1 mm).
- SDII is the simple daily intensity index fined as the annual total divided by the number of days with precipitation larger than 1 mm/day (units are in 0.1 mm/day).
- R95T is the fraction of annual total precipitation due to events exceeding the 1961–1990 95th percentile (units are in %).

The indices were computed each year both for the observations and the models. Then we computed the CDF of each index both for observations and models at each station. The quadratic error and mean error for each extreme event index are shown in Table 6.

R10 We can observe that the Gamma distribution is biased often overestimating the number of days with precipitation of more than 10 mm/day. The other three models are similar although the Weibull model is slightly better both in mean and quadratic error.

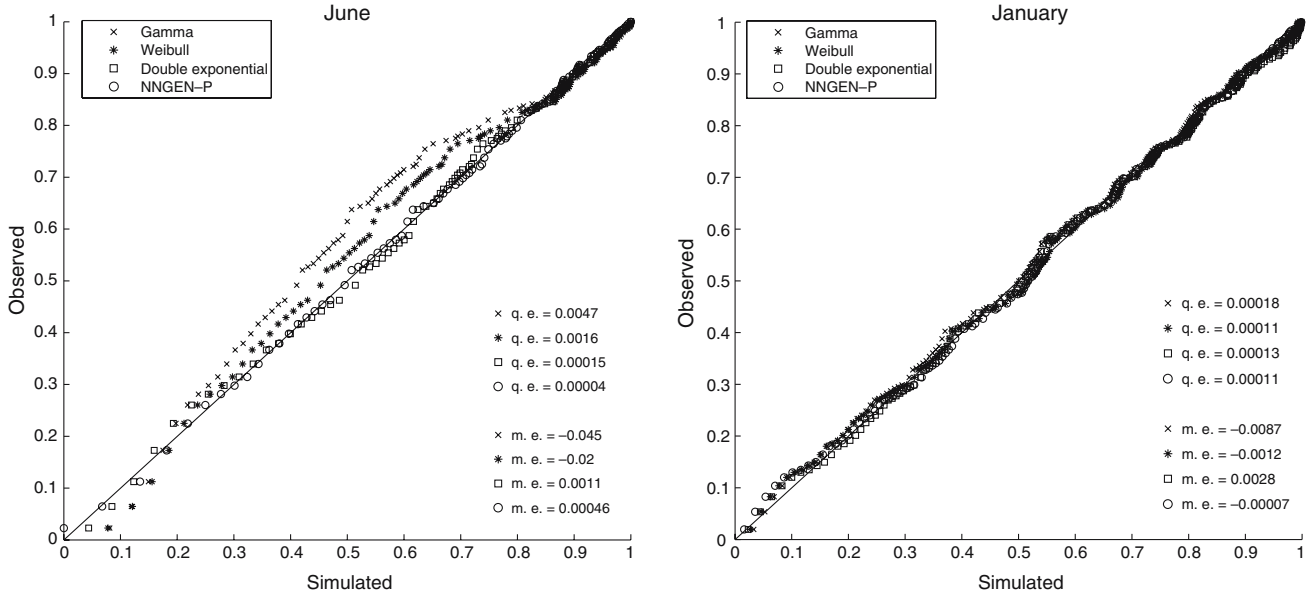

Fig. 7 PP-plots of the 4 model fits to observed rainfall CDFs in January in Ezeiza (**a**) and July in Bahia Blanca (**b**)

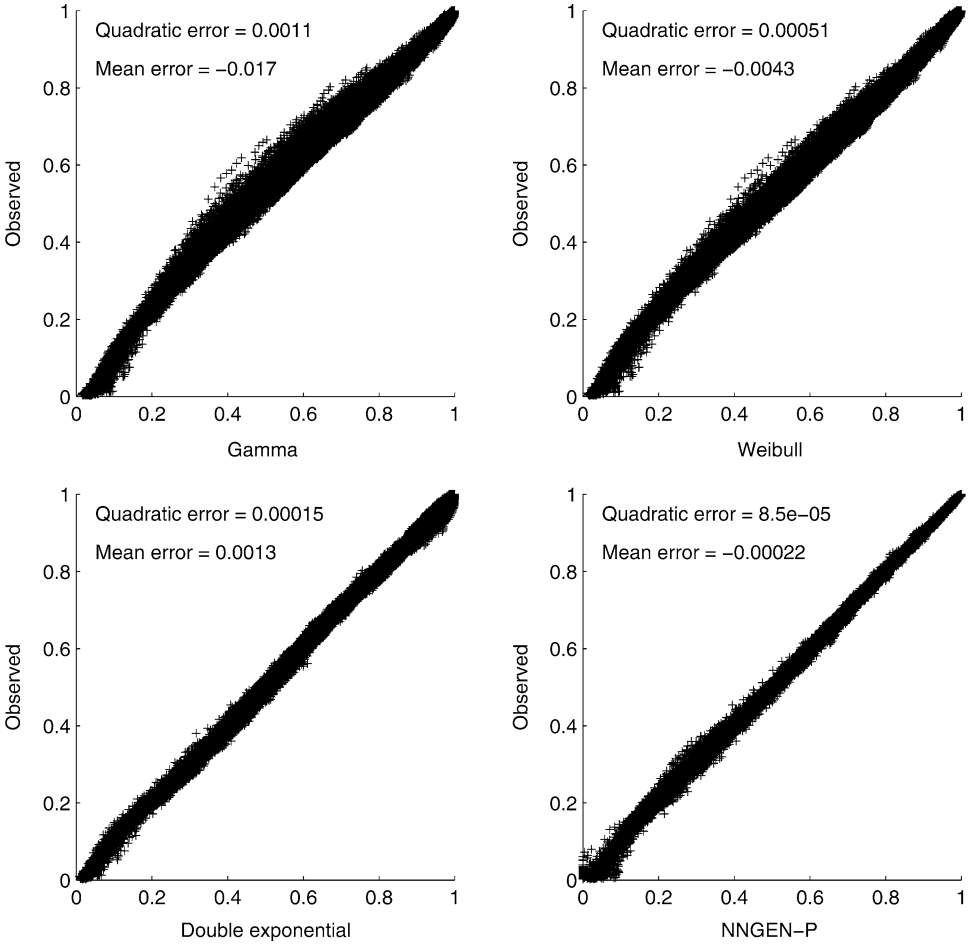

Fig. 8 PP-plot of the observed and modeled rainfall amount CDFs. Simulated quintiles are performed by the Gamma (**a**), Weibull (**b**), Double Exponential (**c**) and NNGEN-P (**d**) models. The mean error and the root-mean square of each model-data misfit are indicated

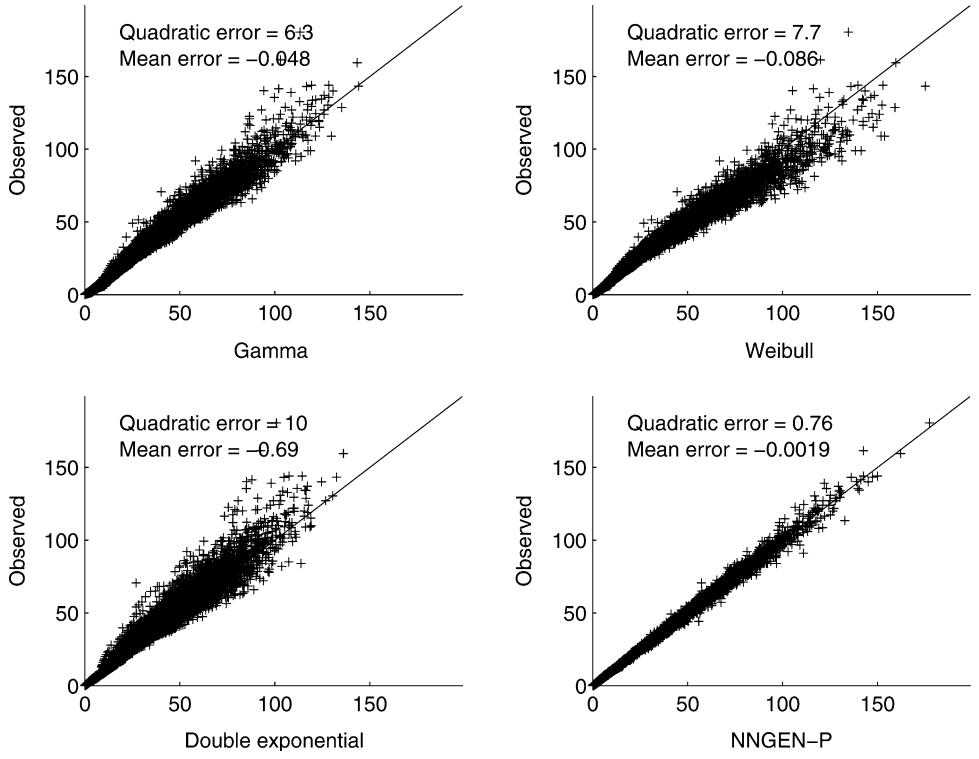

Fig. 9 QQ-plot of the observed and simulated rainfall amount CDFs. Simulated quintiles are performed by the Gamma (**a**), Weibull (**b**), Double Exponential (**c**) and NNGEN-P (**d**) models. The mean error and the root-mean square of each model-data misfit are indicated. As a reminder, a QQ-plot is built as follows. To each probability of the CDF (both observations and simulations have the same number of points, and thus the same bins of the CDF), we can associate a rainfall amount for observed or simulated series. The two series are then plotted as a scatter plot. Daily rainfall is in the range 0.1–200 mm. The straight line crossing both axes at zero represent a perfect fit. For each plot, we computed a mean algebraic error and the root-mean square of the model-data error misfit

CDD Considering that the dry/wet day occurrence model is the same for all four simulations (we use NNGEN-P), the differences between the four models can only result from their skill in representing the adequate probability for precipitation lower than 1 mm/day. Here again, we obtain poorer results with the Gamma function. The Weibull model tends a little to underestimate such long dry periods. Both NNGEN-P and Double Exponential have the same quadratic error, with a slightly weaker bias from the latter.

R5d NNGEN-P represents relatively well the inter-annual distribution of the total precipitation of the five most rainy days. The quadratic error is smaller than the other models and the mean error is close to zero. This result confirms the skill of NNGEN-P in fitting both the low amplitude and large amplitude precipitation amounts.

SDII All models have skill in representing the mean precipitation on rainy days. This was probable as all the models tend to simulate the mean climatological rainfall (not shown) at each station.

R95T NNGEN-P represents better the interannual distribution of the fraction of annual total precipitation due to events exceeding the 1961–1990 95th percentile. This result is coherent with the result found for R5d. Here, NNGEN-P has barely any bias and its quadratic error is much weaker than all other models.

To conclude, the analysis of five extreme precipitation events suggested by Frich et al. (2002) and used for IPCC (Intergovernmental Panel on Climate Change) model output analysis shows that NNGEN-P has more skill in representing these five indices than any other parametric model. This gives confidence in the analysis of the evolution of such indices simulated by NNGEN-P once calibrated to 21st century climate conditions.

8 Conclusion/discussion

The present study describes a new generation of weather generators based on neural network architecture. After presenting the methodology and the optimization of the selected network architectures, it

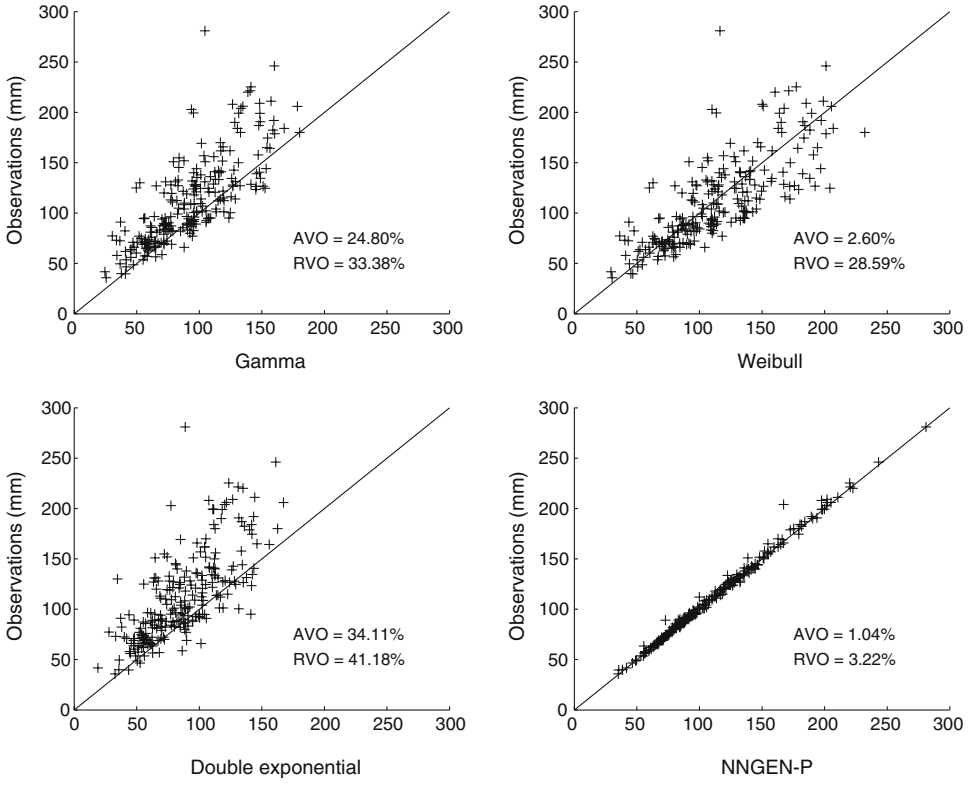

Fig. 10 Relationships between the largest observed daily precipitation amounts (*vertical*) and the corresponding model derived extremes for the four models (Gamma, Weibull, Double Exponential and NNGEN-P)

Table 5 Monthly standard deviation

Ocurrence model	Amount model	AVO (%)	RVO (%)
Markov	Gamma	13.25	13.19
Markov	Weibull	3.07	13.88
Markov	Double exponential	16.62	18.04
Markov	NNGEN-P	4.89	7.14
NNGEN-P	Gamma	13.48	13.97
NNGEN-P	Weibull	3.25	14.66
NNGEN-P	Double exponential	16.95	19.55
NNGEN-P	NNGEN-P	5.13	7.63

In this table, we can see the Average Variance Overdispersion (AVO) and the Standard Deviation Variance Overdispersion (RVO) relative to the capacity of each model in reproduce the interannual rainfall standard deviation for each month at each station

presents the application of the weather generator for daily precipitation called NNGEN-P to a large set of stations located in Argentina, as well as in the United States, Europe and the Tropics.

Like other classical daily rainfall generators, NNGEN-P separates the simulation of the wet/day nature of each day from the simulation of the rainfall during a wet day. Consequently, its evaluation and comparison to other models is divided in two steps.

First, NNGEN-P has been designed to simulate the length of the following dry or wet spell. When compared to a classical two-state Markov model, we found the NNGEN-P results similar in mean number of wet days per month and interannual variation of number of wet days for each month of the year. However, the major improvement of NNGEN-P is to simulate with great accuracy the longest observed wet/dry spells. This feature is important when using weather generators to force crop models. Indeed, an underestimation or overestimation of the risks of dry/wet spells can strongly affect plant growth and therefore its simulated yields.

Second, NNGEN-P has been used to simulate rainfall during wet days. We compared NNGEN-P results to three parametric models usually applied to fit rainfall distribution (Gamma, Weibull and Double Exponential). We found that each of these parametric functions may actually give good results during certain months at certain stations, but overall are unable to offer a good fit to any kind of climate regime. This is simply due to the parametric nature of these functions, which defines the shape of the family of each of these functions, whatever the parameters used. On the contrary, the non-parametric adaptive nature of NNGEN-P allows it to fit any kind of distributions and simulate it with a much higher accuracy. It is important to note here that the procedure to optimize NNGEN-P avoids the overfitting problem. NNGEN-P simulated CDFs

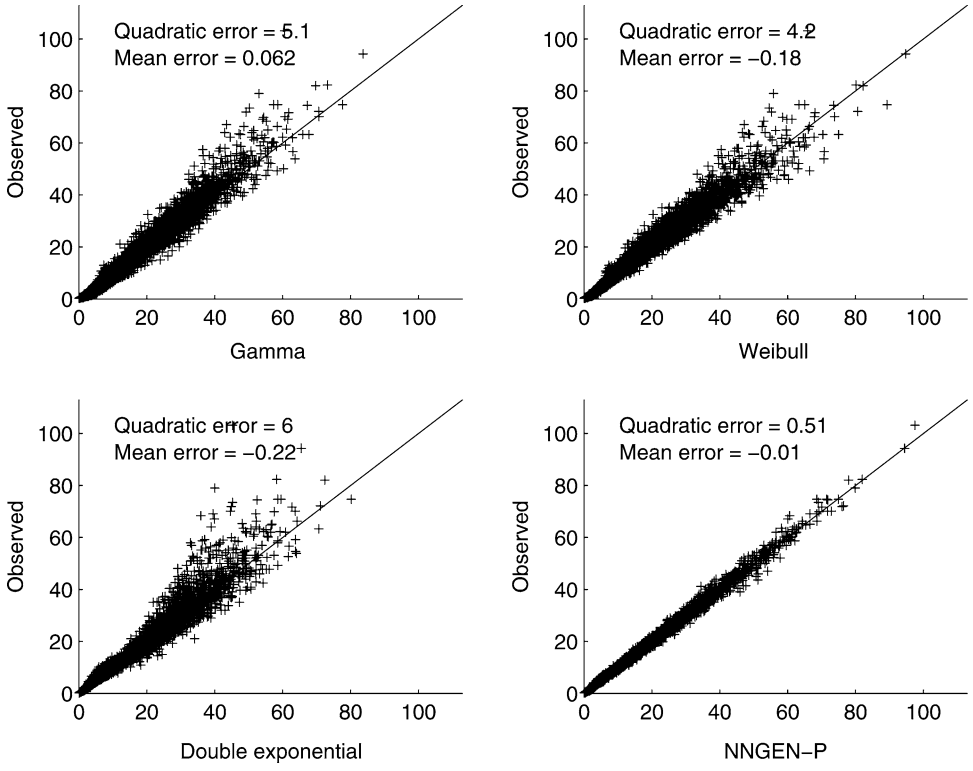

Fig. 11 Same as Fig. 9 but for stations located in the United States of America (see Table 3)

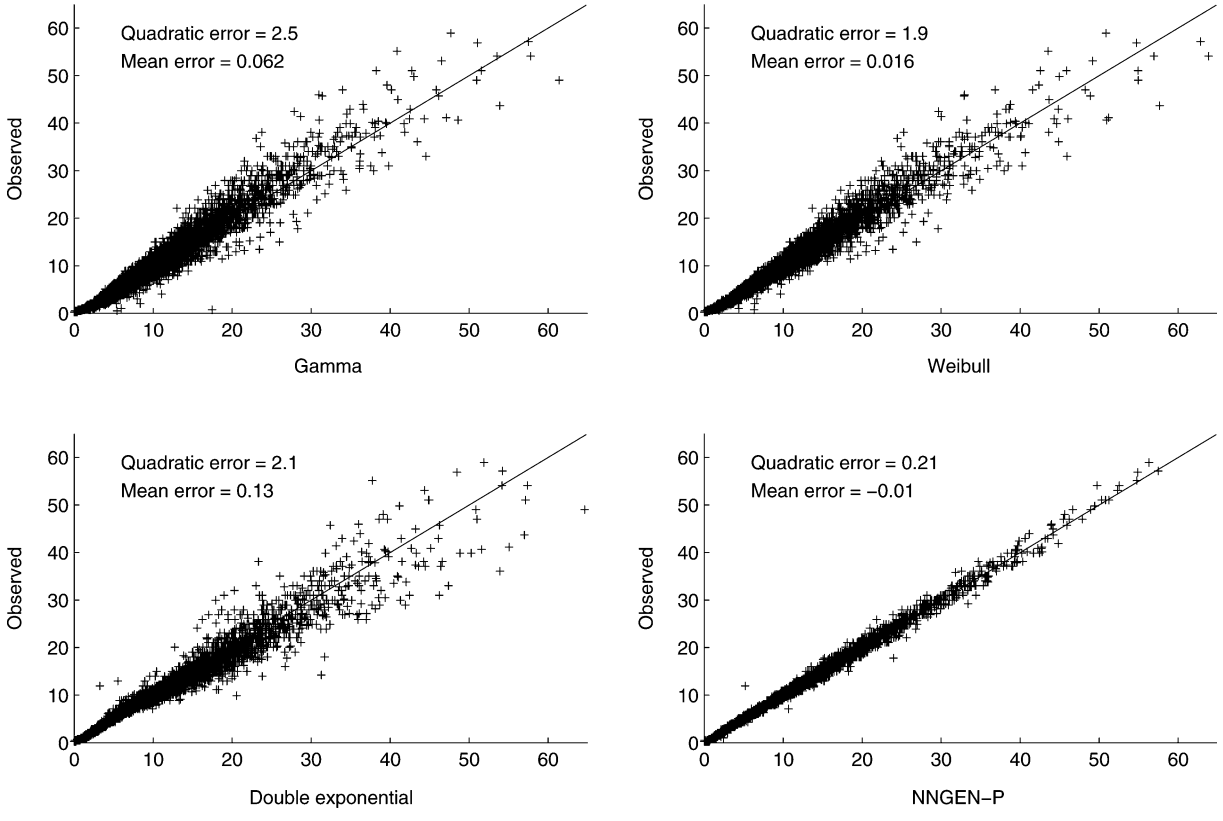

Fig. 12 Same as Fig. 9 but for stations located in Europe (see Table 2)

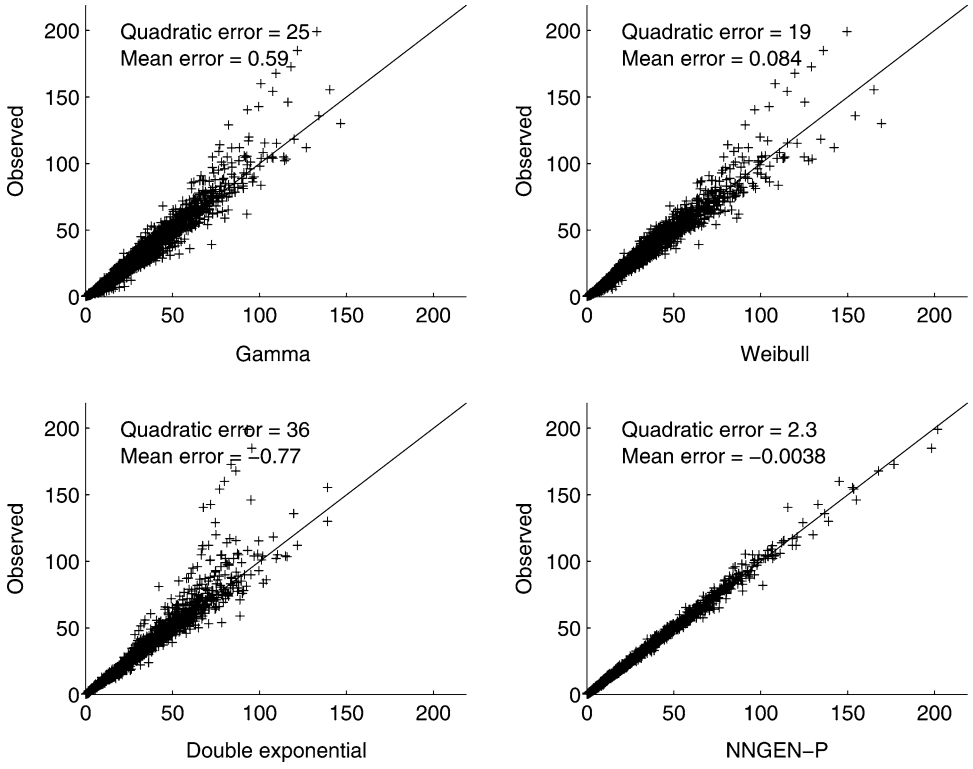

Fig. 13 Same as Fig. 9 but for stations located in the Tropics (see Table 4)

Table 6 Extreme Indices

Model	R10		CDD		R5d		SDII		R95T	
	q.e.	m.e.	q.e.	m.e.	q.e.	m.e.	q.e.	m.e.	q.e.	m.e.
Gamma	9.1	–2.2	43	3.2	6	1.1	0.36	–3.1	24	2.8
Weibull	4.1	–0.13	36	2.7	4.7	–0.068	0.34	–1.6	18	–1.5
Double exponential	5.4	–0.87	27	1.4	8.3	1.6	0.5	11	30	3.6
NNGEN-P	4.4	–0.25	27	1.7	4.7	0.84	0.32	0.074	9.3	0.19

In this table, we can see the quadratic error (q.e.) and mean error (m.e.) for each model and an each extreme index

are a smoothed version of the observed CDFs. The major strength of NNGEN-P is that the non-linear adaptive nature of the neural network allows to fit complex CDF shapes such as are often observed during austral winter.

Finally, when comparing the skill of NNGEN-P and the other three models (Gamma, Weibull and Double Exponential) in simulating five extreme precipitation event indices defined by Frich et al. (2002) and used to evaluate IPCC model outputs, we found NNGEN-P to be the most consistent model for all five indices. NNGEN-P has much larger skills in simulating the indices associated to extreme rainfall amounts in particular. This gives us confidence in analyzing the simulation of such extreme indices using NNGEN-P when calibrated to 21st century climate conditions. This will be done in a forthcoming paper.

Finally, the results described in the present paper encourage us to develop a full weather generator including temperature and radiation, the validation results of which will be presented in another paper.

Acknowledgments We wish to thank the Institut de Recherche pour le Développement (IRD), the Institut Pierre-Simon Laplace (IPSL), the Centre National de la Recherche Scientifique (CNRS; Programme ATIP-2002) for their financial support crucial in the development of the authors' collaboration. We are also grateful to the European Commission for funding the CLARIS Project (Project 001454) in whose framework part of the present study was undertaken. We are wish to thank the University of Buenos Aires and the Department of Atmosphere and Ocean Sciences for welcoming Jean-Philippe Boulanger. Special thanks are addressed to the Ecole Normale Supérieure de Lyon and to Elie Desmond with whom this work was initiated. Fruitful discussions with Sylvie Thiria and Carlos Mejia were helpful during the training phase of Jean-Philippe Boulanger in the use of neural networks. Finally, we are thankful to

Ian Nabney and Chris Bishop for sharing freely the Netlab Software source codes. Special thanks are addressed to Santiago Meira from INTA Pergamino for providing the Pergamino daily rainfall time series.

9 Appendix A: the multi-layer perceptron

The multi-layer perceptron (MLP) is probably the most widely used architecture for practical applications of neural networks (Nabney 2002). From a computational point of view, the MLP can be described as a set of functions applied to different elements (neurons) using relatively simple arithmetic formulae, and a series of methods to optimize these functions based on a set of data. In the present study, we will only focus on a two-layer network architecture (Fig. 14). Its simplest element is called a neuron and is connected to all the neurons in the upper layer (either the hidden layer if the neuron belongs to the input layer or the output layer if the neuron belongs to the hidden layer). Each neuron has a value, and each connection is associated to a weight.

As shown in Fig. 14, in the MLP case we considered, the neurons are organized in layers: an input layer (the values of all the input neurons except the bias are specified by the user), a hidden layer and an output layer. Each neuron in one layer is connected to all the neurons in the next layer. More specifically, in the present case, the MLP architecture has one input neuron (I), H neurons in the hidden layer (value to be estimated by the method) and one output neuron (O). The first layer of the network forms H linear combinations of the input vector to give the following set of intermediate activation variables: $h_j^{(1)} = w_j^{(1)} I + b_j^{(1)}$ $j = 1,...,H$ where $b_j^{(1)}$ corresponds to the bias of the input layer. Then, each activation variable is transformed by a non-linear activation function, which in most cases (including ours), is the hyperbolic tangent function (tanh):$v_j = \tanh(h_j^{(1)})$ $j = 1,...,H$. Finally, the v_j are transformed to give a second set of activation variables associated to the neurons in the output layer: $O^{(2)} = \sum_{j=1}^{H} w_j^{(2)} v_j + b^{(2)}$ where $b^{(2)}$ corresponds to the bias of the hidden layer.

The weights and biases are initialized by random selection from a zero mean, unit variance isotropic Gaussian where the variance is scaled by the fan-in of the hidden or output units as appropriate. During the training phase, the neural network compares its outputs to the correct answers (a set of observations used as output vector), and it adjusts its weights in order to minimize an error function. In our case, the weights and biases are optimized by back-propagation using the scaled conjugate gradient method.

Such an architecture is capable of universal approximation and given a sufficiently large number of data, the MLP can model any smooth function. Finally, the interested reader can find an exhaustive description of the MLP network, its architecture, initialization and training methods in Nabney (2002). Our study made use of the Netlab software (Nabney 2002).

10 Appendix B: Bayesian approach to optimize the MLP architecture

When optimizing a model to the data, it is usual to consider the model as a function such as: $y = f(x, w) + \epsilon$, where y are the observations, x the inputs, f the model, w the parameters to optimize (or the weights in our case) and ϵ the remaining error (model-data misfit). The more complex the model to fit (i.e. the number of parameters), the smaller the error, with the usual drawback of overfitting the data by fitting both the ''true'' data and its noise. Such an overfit is usually detected due to a very poor performance of the model on unseen data (data not included in the training phase). Therefore, optimizing the model parameters through minimizing the residual ϵ may actually lead to a poor model performance. One way to avoid this problem is to consider also the errors on the model parameters. The use of a Bayesian approach is very helpful to deal with this difficulty. Although two kinds of Bayesian approaches have been demonstrated to be effective (Laplace approximation and Monte Carlo techniques), in the following we will only consider the first one. Nabney (2002) offers an exhaustive discussion on this subject. And, for the reader to understand our approach, we believe important to present a summary.

First of all, following the same notations as in Nabney (2002), let's consider two models M1 and M2 (in our case two MLPs which only differ by the number of neurons in the hidden layer and with M2 having more neurons than M1). Using Bayesian theorem, the posterior probability or likelihood for each model is: $p(M_i|D) = \frac{p(D|M_i)p(M_i)}{p(D)}$. Without any a priori reason to prefer any of the two models, the models should actually be compared considering probability $p(D|M_i)$, which can be written (MacKay 1992) as $p(D|M_i) = \int p(D|w, M_i)p(w|M_i))\mathrm{d}w$. Considering that for either model, there exists a best choice of parameters $\hat{w}_i$ for which the probability is strongly peaked, then the previous equation can actually be simplified: $p(D|M_i) \approx p(D|\hat{w}_i, M_i)p(\hat{w}_i|M_i)\Delta\hat{w}_i^{\text{posterior}}$ where the

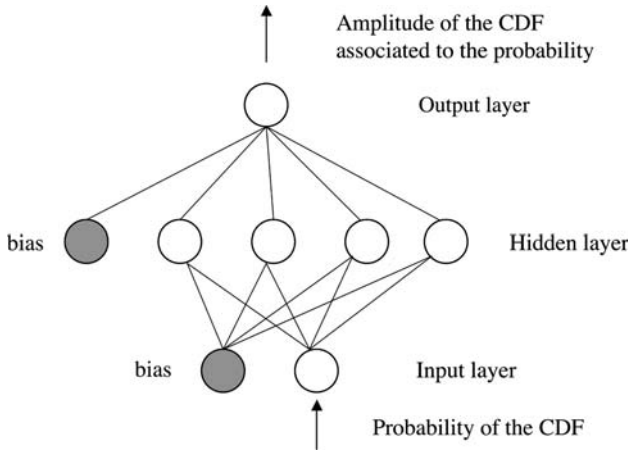

Fig. 14 Schematic representation a two-layer MLP as used in this study. In the input layer, one neuron represents the probability value of the CDF under study (wet spell, dry spell, rainfall amount). The number of neurons in the hidden layer is optimized by the method. In the output layer, one neuron represents the amplitude of the CDF associated to the input probability (length of the wet spell, length of the dry spell or amount of rainfall). The units or neurons called bias are units not connected to a lower layer and whose value is always equal to –1. They actually represent the threshold value of the next upper layer

last term represents the volume (in the space of the parameters) where the probability is uniform. Assuming that the prior probability $p(\hat{w}_i|M_i)$ has been initialized so that it is uniform over a certain volume of the prior parameters, we can rewrite the previous equation as: $p(D|M_i) \approx p(D|\hat{w}_i, M_i)(\Delta\hat{w}_i^{\text{posterior}}/\Delta\hat{w}_i^{\text{prior}})$. The new equation is the product of two terms evolving in opposite directions as the complexity of the model increases. The first term on the right-hand side increases (i.e. the model-data misfit decreases) as the model complexity increases. The second term is always lower than 1 and is approximately exponential with parameters (Nabney 2002), which penalizes the most complex models. In conclusion, taking into account the weight uncertainty should reduce the overfitting problem. We will now explain how this can be done.

For a given number of units in the hidden layer, an optimum set of weights and biases can be calculated using the maximum likelihood to fit a model to data. In this case, the optimum set of parameters (weights and biases) is the one, which is most likely to have generated the observations. A Bayesian approach (or quasi-Bayesian approach due to difficulties in using Bayesian inference caused by the non-linear nature of the neural networks) may be valuable to infer these two classes of errors: model-data misfit and parameter uncertainty.

According to Bayesian theorem, for a given MLP architecture, the density of the parameters (noted w) for a given dataset (D) is given by:

$$p(w|D) = \frac{p(D|w)p(w)}{p(D)}.$$

In a first step, let's only consider the terms depending on the weights. The negative log likelihood is given by E = -Log($p(w|D)$) = -Log($p(D|w)$)-Log($p(w)$).

The likelihood $p(D|w)$ represents the model-data fit error, which can be modeled by a Gaussian function such as:

$$p(D|w) = \left(\frac{\beta}{2\pi}\right)^{N/2} \exp\left(-\frac{\beta}{2}\sum_{n=1}^{N}\{f(x_n, w) - y_n\}^2\right) = \left(\frac{\beta}{2\pi}\right)^{N/2} \exp\left(-\frac{\beta}{2}E_D\right)$$

where β represents the inverse variance of the model-data fit error.

The requirement for small weights (i.e. avoiding the overfitting) suggests a Gaussian distribution for the weights of the form:

$$p(w) = \left(\frac{\alpha}{2\pi}\right)^{W/2} \exp\left(-\frac{\alpha}{2}\sum_{i=1}^{W} w_i^2\right) = \left(\frac{\alpha}{2\pi}\right)^{W/2} \exp\left(-\frac{\alpha}{2}E_W\right)$$

where α represents the inverse variance of the weight distribution. α and β are known as hyperparameters. Therefore, to compare different MLP architectures, we need first to optimize the MLP weights, biases and hyperparameters for any given architecture. Such an optimization can be reached using the evidence procedure, which is an iterative algorithm. Here again, we refer the reader to Nabney (2002). Briefly, if we consider a model to be determined (for any given architecture) by its two hyperparameters, we can write (as previously) that two models may be compared through their respectively maximized evidence $p(D|\alpha,\beta)$, log evidence of which can be written as:

$$\ln p(D|\alpha,\beta) = -\alpha E_w - \beta E_D - \frac{1}{2}\ln|A| + \frac{W}{2}\ln\alpha + \frac{N}{2}\ln\beta - \frac{W+N}{2}\ln(2\pi)$$

where A is the Hessian matrix of the total error function (function of α and β).

Based on the previous equation, the evidence procedure is used to optimize the weights and hyperparameters for any given architecture, and the model optimized log evidence is calculated. We then compare the computed log evidence for different architectures,

and we finally chose the smallest architecture giving the minimum negative log evidence.

Applying this Bayesian approach, different numbers of neurons in the hidden layer was found according to the complexity of the CDF shapes to fit. We found this number to range from 3 to 14 neurons, with larger numbers mainly found when fitting the dry spell CDF (the longest wet spells do not exceed 15 days and are often in the range of 5–6 days, while dry spells can be as long as a few months).

References

Adamowsky K, Smith AF (1972) Stochastic generation of rainfall. J Hydraul Eng ASCE 98:1935–1945

Akaike A (1974) New look at the statistical model identification. IEEE Trans Automatic Control 19:716–723

Allen DM, Haan CT (1975) Stochastic simulation of daily rainfall. Res. Rep. No. 82, Water Resource Inst., University Kentucky, Lexington,Kentucky, USA

Buishand TA (1982) Some methods for testing the homogeneity of rainfall records. J Hydrol

Buishand TA (1978) Some remarks on the use of daily rainfall models. J Hydrol 36:295–308

Caballero R, Jewson S, Brix A (2001) Long memory in surface air temperature: detection, modeling and application to weather derivative calculation, Clim Res

Charles SP, Bates BC, Hughes JP (1999) A spatiotemporal model for downscaling precipitation occurrence and amounts. J Geophys Res 104:31657–31669

Foufoula-Georgiou E, Lettenmaier DP (1987) A Markov renewal model for rainfall occurrences. Water Resour Res 23:875–884

Frich P, Alexander LV, Della-Marta P, Gleason B, Haylock M, Klein Tank AMG, Peterson T (2002) Observed coherent changes in climatic extremes during the second half of the twentieth century. Clim Res 19:193–212

Geng S, Penning De Vries FWT, Supit I (1986) A simple method for generating daily rainfall data. Agric For Meteorol 36:363–376

Hansen JW, Ines AVM (2005) Stochastic disaggregation of monthly rainfall data for crop simulation studies. Agric For Meteorol

Hay LE, McCabe GJ Jr, Wolock DM, Ayres MA (1991) Simulation of precipitation by weather type analysis. Water Resour Res 27:493–501

Hutchinson MF (1986) Methods of generation of weather sequences. In: Bunting AH (ed) Agric Env CAB. International, Wallingford, pp 149–157

Hutchinson MF (1995) Stochastic space-time weather models from ground-based data. Agric For Meteorol 73:237–265

Jewson S (2004) Weather derivative pricing and the potential accuracy of daily temperature modeling. http://ssrn.com/abstract = 535122

Katz RW (1977) Precipitation as a chain-dependent process. J Appl Meteor 16:671–676

Lall U, Rajagopalan B, Traboton DG (1996) A non-parametric wet/dry spell model for resampling daily precipitation. J Clim 11:591–601

MacKay DJC (1992) Bayesian interpolation. Neural Comput 4:415–447

Nabney IT (2002) Netlab. Algorithms for pattern recognition, advances in pattern recogniction. Springer, Berlin Heidelberg New York, pp 420

Racsko P, Szeidl L, Semenov M (1991) A serial approach to local statistic weather models. Ecol Modeling 57:27–41

Rajagopalan B, Lall U (1999) A k-nearest-neighbor simulator for daily precipitation and other weather variables. Water Resour Res 35:3089–3101

Rajagopalan B, Lall U, Tarboton DG (1996) A nonhomogeneous Markov model for daily precipitation simulation. J Hydrol Eng-ASCE 1:33–40

Richardson CW (1981) Stochastic simulation of daily precipitation, temperature, and solar radiation. Water Resour Res 17:182–190

Roldan J, Woolhiser DA (1982) Stochatic daily precipitation models. 1. A comparison of occurrence processes. Water Resour Res 18:1451–1459

Schwartz G (1978) Estimating the dimension of a model. Ann Statistics 6:641–644

Selker JS, Haith DA (1990) Development and testing of single-parameter precipitation distributions. Water Resour Res

Semenov MA, Barrow EM (1997) Use of stochastic weather generator in the development of climate change scenarios. Clim Change 35:397–414

Sharpley AR, Willians JR (1990) EPIC an Erosion/Productivity Impact Calculator: 2. User Manual, U.S. Department of Agriculture, ARS Tech Bull No 1768

Siriwarden et al. 2002

Srikanthan R, McMahon TA (1983) Stochastic simulation of daily rainfall for Australian stations. Trans ASAE 26:754–759, 766

Srikanthan R, McMahon (1985) Stochastic generation of rainfall and evaporation data. Australian Water Resources Council, Tech. Paper No. 84, AGPS, Canberra, 301pp

Srikanthan R, McMahon TA (2001) Stochastic generation of annual, monthly and daily climate data: a review. Hydrol Earth Syst Sci 5:653–670

Stern RD, Coe R (1984) A model fitting analysis of daily rainfall data. J Roy Stat Soc A147:1–34

Wilks DS (1989) Conditioning stochastic daily precipitation models on total monthly precipitation. Water Resour Res 25:1429–1439

Wilks DS (1992) Adapting stochastic weather generation algorithms for climate change studies. Clim Change 22:67–84

Wilks DS (1998) Multisite generalization of a daily stochastic precipitation generation model. J Hydrol 210:178–191

Wilks DS (1999a) Simultaneous stochastic simulation of daily precipitation, temperature and solar radiation at multiple sites in complex terrain. Agric For Meteor 96:85–101

Wilks DS (1999b) Interannual variability and extreme-value characteristics of several stochastic daily precipitation models. Agric For Meteor 93:153–169

Wilson LL, Lettenmaier DP, Skyllinggstad E (1992) A hierarchic stochastic model of large-scale atmospheric circulation patterns and multiple station daily precipitation. J Geophys Res D3:2791–2809

Woolhiser DA, Roldan J (1982) Stochastic daily precipitation models. 2. A comparison of distribution amounts. Water Resour Res 18:1461–1468

Woolhiser DA, Pegram GS (1979) Maximum likelihood estimation of Fourier coefficients to describe seasonal variation of parameters in stochastics daily precipitation models. J Appl Meteor 18:34–42

Woolhiser DA, Roldan J (1986) Seasonal and regional variability of parameters for stochastic daily precipitation models. Water Resour Res 22:965–978

Woolhiser DA (1992) Modeling daily precipitation: B. Progress and problems. In: Walden AT, Guttorp P (eds) Statistics in the Env. And Earth Sciences, John Wiley, New York, pp 71–89