The effect of calibration data length on the performance of conceptual versus data-driven hydrological models

Georgy Ayzel[a,*], Maik Heistermann[a]

[a] Institute for Environmental Sciences and Geography, University of Potsdam; Karl-Liebknecht-Str. 24-25, 14476 Potsdam, Germany

ARTICLE INFO

Keywords:
artificial neural networks
calibration
deep learning
rainfall-runoff modelling

ABSTRACT

We systematically explore the effect of calibration data length on the performance of a conceptual hydrological model, GR4H, in comparison to four Artificial Neural Network (ANN) architectures: Multi-Layer Perceptrons (MLP), Recurrent Neural Networks (RNN), Long Short-Term Memory Networks (LSTM) and Gated Recurrent Units (GRU) – of which the last two have just recently been introduced to the field of hydrology. We implemented our analysis for the Rimbaud River, Southern France, with 37 years of meteorological and discharge data. 17 years were reserved for independent validation; from the remaining 20 years, we sampled increasing amounts of data for model calibration, and found pronounced differences in model performance: GR4H, LSTM, and GRU were superior, converging to a similar performance with increasing calibration data length. While GR4H required less data to converge, LSTM and GRU caught up at a remarkable rate, considering their number of parameters. RNN appeared least robust; MLP, in turn, performed poorly for short calibration periods, but benefited more from additional calibration data. All ANN-based models exhibited a high performance loss from calibration to validation. These findings confirm the potential of modern deep-learning architectures in rainfall-runoff modelling, but also highlight the risk of extending calibration data at the cost of independent validation for these models.

1. Introduction

According to Mount et al. (2016), a hydrological model is a functional relationship in which the model output (y) is determined by its structure (f), inputs (x), parameters (p), and an error ε:

$$y = f(x, p) + \varepsilon \tag{1}$$

Following Mount et al. (2016), conceptual hydrological models put a strong emphasis on *structure* as a result of knowledge-driven model reduction, of how we parsimoniously formalize our concept of a hydrological system. In contrast, data-driven methods trust in the explanatory power of input data itself as they *"simultaneously 'discover' the model structure and optimize its parameters [and minimize] the role of a priori hypotheses"* (Mount et al., 2016). As a consequence, the number of parameters that need to be optimized (or *calibrated*) is relatively small for conceptual hydrological models (typically in the order of 10^0–10^2), and relatively high (in the order of 10^3–10^5) for data-driven models, such as Artificial Neural Networks (ANN).

Numerous studies have been devoted to different aspects of hydrological model calibration, such as the choice of the objective function Fowler et al. (2018), or the use of auxiliary information (Nijzink et al., 2018). Yet, the majority of hydrological models are still calibrated against discharge time series, using square error-based objective functions (Arsenault et al., 2018), and their validity requires examination on independent data (Refsgaard et al., 2005). To that end, split-sampling techniques have been suggested by various authors as a basis for calibration-validation experiments, as diagnostic tools to assess the reliability of models and the robustness of their parameterisation (e.g., Coron et al., 2012; Thirel et al., 2015; Motavita et al., 2019).

In the light of limited data availability for calibration and validation, such techniques also provide a means to study the effect of the amount (or length) of calibration data on the robustness and reliability of models. A fair number of studies investigated that effect for conceptual hydrological models (e.g., Sorooshian et al., 1983; Yapo et al., 1996; Boughton, 2007; Perrin et al., 2007; Li et al., 2010; Arsenault et al., 2018; Motavita et al., 2019). For data-driven ANN models, Anctil et al. (2004), Cigizoglu and Kisi (2005), and Toth and Brath (2007) investigated the effect of calibration data in the context of runoff forecasting, i.e. on the skill of runoff prediction at lead times of several days,

*Corresponding author

ayzel@uni-potsdam.de (G. Ayzel); heisterm@uni-potsdam.de (M. Heistermann)

ORCID(s):

using observed runoff and meteorological forcing as predictors. Rainfall-runoff modelling, however, aims to predict runoff from meteorological variables, only. That is a different problem, and we did not find any study to investigate the effect of calibration data length on data-driven rainfall-runoff models, least of all comparing that effect for conceptual hydrological models versus various data-driven models.

The objective of this study is to fill that gap: to investigate the effect of calibration data length on the ability of a conceptual hydrological model (GR4H) and four data-driven ANN models to predict runoff from rainfall – at an hourly resolution. Hence, we do not limit our analysis to a single ANN architecture. In fact, we include both "traditional" architectures such as Multi-Layer Perceptron (MLP) and Recurrent Neural Networks (RNN), as well as "modern" deep-learning architectures such as Long Short-Term Memory Networks (LSTM) and Gated Recurrent Units (GRU) which have just recently been introduced to the field of hydrological modelling (Marçais and de Dreuzy, 2017; Kratzert et al., 2018; Shen, 2018; Reichstein et al., 2019). Specifically, Kratzert et al. (2018, 2019) have identified LSTM as superior to traditional ANN architectures and a conceptual hydrological model. Therefore, we specifically examine to what extent that superiority could depend on the amount of data that is available for model calibration.

Altogether, this study aims to provide a broad perspective on the role of calibration data length, and to serve as a guideline for establishing corresponding calibration-validation experiments. We start with an outline of the experimental setup (Sect. 2.1) as well as the underlying data for the meteorological forcing and the observed discharge (Sect. 2.2); the different models are described in Sect. 2.3, followed by details on model calibration (Sect. 2.4). We then provide the results of our experiments in Sect. 3 and conclude with Sect. 4.

2. Data and Methods

2.1. Overview of the Experimental Design

To investigate the effect of calibration data length on model performance, we propose a three-step procedure (Figure 1): first, we split the entire dataset of meteorological forcings (namely, precipitation and potential evapotranspiration) and observed discharge (Sect. 2.2) into two mutually exclusive periods for model calibration (1968–1987, 20 years) and validation (1988–2004, 17 years). From the former, we select calibration sub-periods of specific lengths using a moving window approach. Second, we calibrate all models against observed discharge from the corresponding sub-periods in order to obtain a set of optimal model parameters. Third, and finally, we evaluate the performance of the corresponding parameter sets on the validation period.

For example, to investigate the model performance for only a single year of calibration, we begin to split the entire calibration period (1968–1987) into sub-periods which have the length of one calendar year using a moving window with the size of one year and the stride of one year, i.e. 1968, 1969, …, 1987 (20 sub-periods in total). Then, for each sub-period, we calibrate each model (GR4H, MLP, RNN, LSTM, and GRU, see Sect. 2.4). After that, we evaluate the performance on the validation period. In the end, we collect the performance measures for each pair "model – calibration sub-period" for both the calibration sub-period and the independent validation period. We iterate over calibration period lengths from one (20 calibration sub-periods) to 20 (one calibration sub-period) calendar years to obtain the final dataset for model performance diagnostics.

,

2.2. Study Area and Data

The Rimbaud River basin at Collobrières is located in the Maures highlands of southeastern France, close to the Mediterranean Sea coast, and drains an area of 1.4 km^2 (Figure 2). For a detailed description of the basin characteristics, such as elevation, vegetation, geology, climatology, and the hydrological regime, please see the Supplement material of Thirel et al. (2015).

,

Hourly precipitation data (P) were obtained from three different sources: two nearby rain gauges, and the SAFRAN reanalysis at a spatial resolution of 8-km (Vidal et al., 2010). In case the precipitation observation at the nearest rain gauge was missing at a specific time, we used the corresponding observation from the second gauge; in a situation when no observations from any of the gauges were available, we used the precipitation estimate from the nearest grid cell of the SAFRAN reanalysis. Rain gauge records as well as hourly discharge observed at the Rimbaud River basin at Collobrières were obtained from the RECOVER research laboratory at the Institut national de la recherche agronomique (INRAE, France). Hourly potential evapotranspiration data (PE) were obtained from the SAFRAN reanalysis which utilizes the Penman-Monteith method (Allen et al., 1998). The resulting data set covers the period

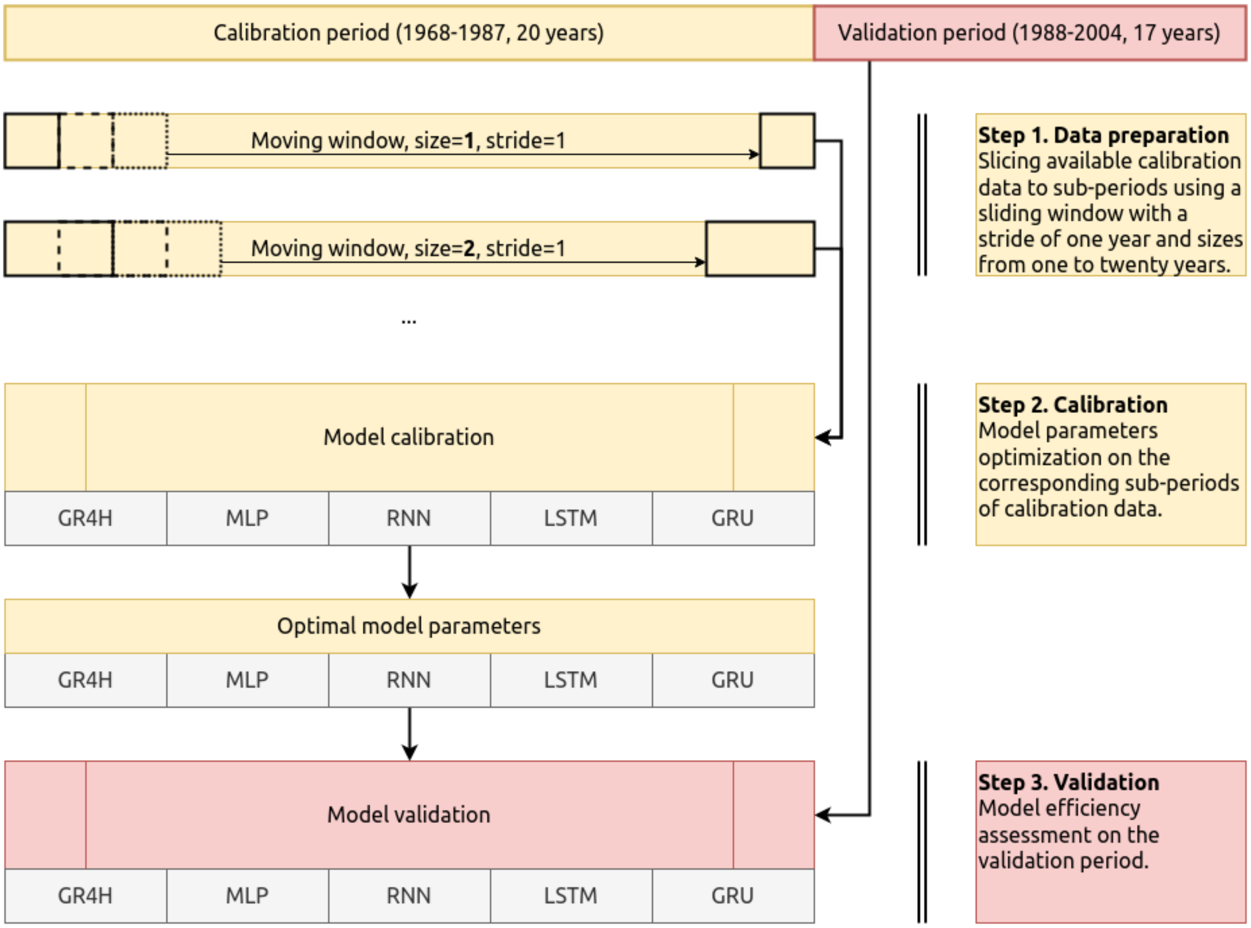
Calibration period (1968-1987, 20 years)
Validation period (1988-2004, 17 years)
Moving window, size=**1**, stride=1
Moving window, size=**2**, stride=1
...
Model calibration
GR4H
MLP
RNN
LSTM
GRU
Optimal model parameters
GR4H
MLP
RNN
LSTM
GRU
Model validation
GR4H
MLP
RNN
LSTM
GRU
Step 1. Data preparation Slicing available calibration data to sub-periods using a sliding window with a stride of one year and sizes from one to twenty years.
Step 2. Calibration Model parameters optimization on the corresponding sub-periods of calibration data.
Step 3. Validation Model efficiency assessment on the validation period.

Figure 1: Flowchart of the proposed workflow for the assessment of the effect of calibration data length on the performance of different runoff formation models.

from January 1, 1968, to December 31, 2004, has no missing values for the meteorological forcing (P and PE), and around 1% missing values for discharge.

2.3. Model description

2.3.1. GR4H Hydrological Model

GR4H is a process-based, lumped, conceptual hydrological model, which was developed at Irstea by Mathevet (2005) for runoff predictions at an hourly resolution (Figure 3).

,

GR4H has four free parameters (Table 1) and mainly mimics the structure of the daily GR4J model (Perrin et al., 2003): precipitation is partitioned in two components, the production store (S; parameter $X1$ describes its capacity), and effective rainfall. Water which percolates from the production store merges with the effective rainfall, and then moves to the outlet on two routes: 10% is routed via a single unit hydrograph (parameter $X4$ describes its time base in hours), and 90% is routed via a unit hydrograph and a nonlinear routing store (R; parameter $X3$ describes its capacity). Potential groundwater exchange with neighboring catchments (F) is taken into account by the empirical function $F = X2(\frac{R}{X3})^{3.5}$ (Perrin et al., 2003; Le Moine et al., 2008). Thus, four model parameters represent the specific properties of a basin, in which production and routing stores implicitly represent the basin memory. GR4H was implemented in many studies worldwide (van Esse et al., 2013; de Boer-Euser et al., 2017; Li et al., 2017) and showed a good efficiency in hourly runoff predictions for basins under different geographical conditions.

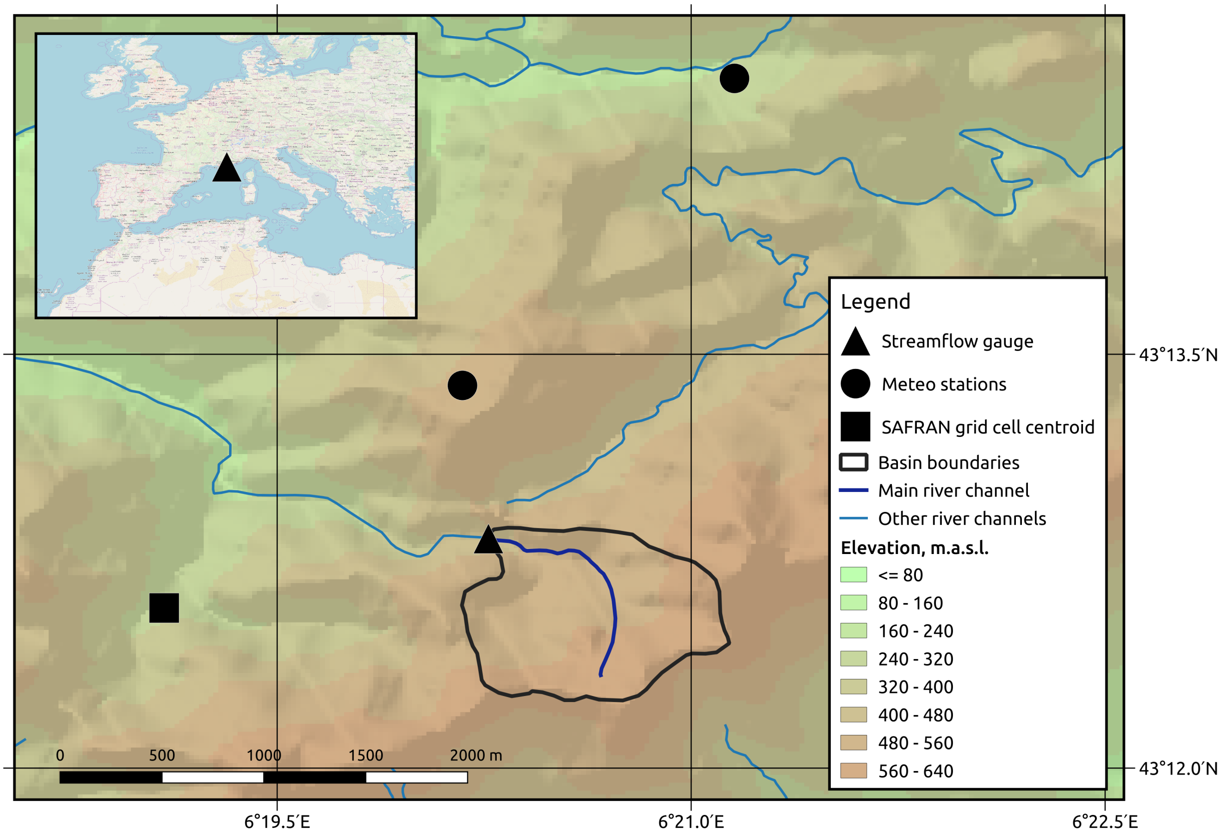

Figure 2: The Rimbaud River basin at Collobrières.

Table 1
List of the parameters of the GR4H model, their calibration ranges, and description.

Parameter	Calibration range	Description
X1	0.1, 1500	Maximum capacity of the production store, in mm
X2	-10, 10	Groundwater exchange coefficient, in mm
X3	0.1, 500	Maximum capacity of the routing store, in mm
X4	0.5, 24	Time base of unit hydrograph, in hours

2.3.2. Artificial Neural Networks (ANNs)

In this study, we consider various topologies of Artificial Neural Networks (ANN), all of which had been suggested for hydrological modelling. For each type of ANN, we employ a four-layer topology, which includes one input, one output, and two computational layers. The first layer (input) propagates input variables such as precipitation and potential evapotranspiration to the second layer. The second layer receives a vector of input variables and then performs computations depending on its type. Here, we use four different types of the second layer: dense (or fully-connected), recurrent, Long Short-Term Memory (LSTM), and Gated Recurrent Unit (GRU) layers. The second layer's output transmits to the third layer, which is always a dense (fully-connected) layer with one neuron and a linear activation function. The purpose of the third layer is to set up a connection between the second layer's multidimensional output and the output layer of the ANN, which consists of discharge as a single output variable.

In the following paragraphs, we will elaborate on the various topologies in more detail.

Multi-Layer Perceptron (MLP)

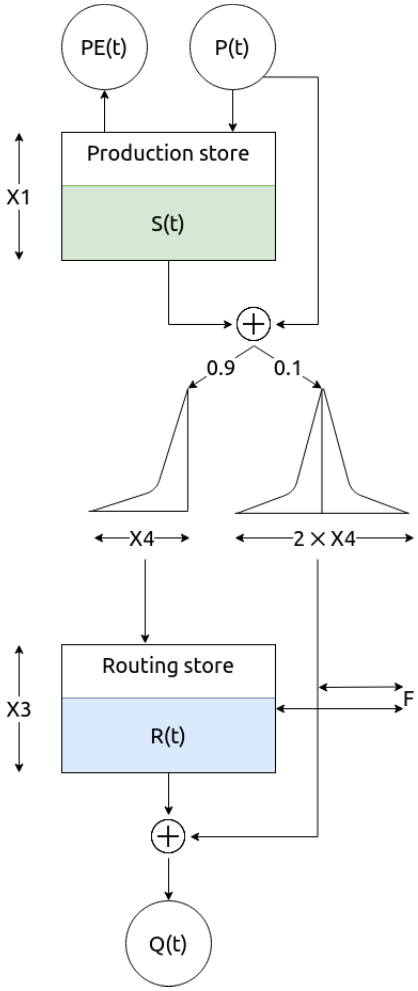

Figure 3: GR4H hydrological model structure.

In general, a multi-layer perceptron (MLP) may have many layers that represent a set of parallel processing units (neurons) (Anctil et al., 2004). However, the most common network topology in the field of hydrological modeling is a four-layer MLP (Figure 4) with one input layer, one dense (fully-connected) layer with a sigmoid activation function, one dense layer with a linear activation function, and one output layer (Maier and Dandy, 2000; Dawson and Wilby, 2001; Anctil et al., 2004; Maier et al., 2010).

The input layer of an MLP consists of the hourly observations of P and PE of the last n hours (t, $t-1$,..., $t-n$), while the output layer consists of the runoff observation at hour t. While the number of last hours n is a hyperparameter and could be a subject to numerical optimization, we fixed it to a value of n=720 to explicitly consider the dynamics of input variables during the preceding month. The first dense layer consists of 64 neurons with a sigmoid activation function, and the second dense layer consists of a single neuron with a linear activation function. Many research studies empirically proved that the respective MLP topology detects and captures the relevant patterns in the data performing a complex non-linear matching between the input and the output variables (Anctil et al., 2004; Cigizoglu and Kisi, 2005; Toth and Brath, 2007). Moreover, feeding the observations at different time steps to MLP might imply a memory effect similar to more complex ANN types such as RNN, LSTM, and GRU, where this effect is explicitly taken into account.

,

Recurrent Neural Network (RNN)

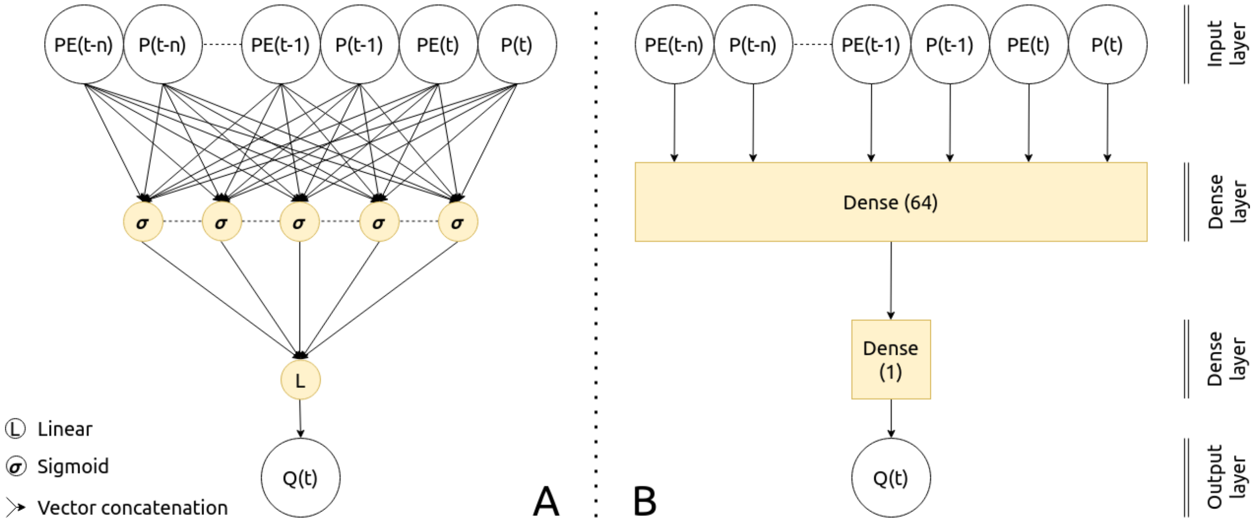

Figure 4: (A) The internal operations and activation functions in the MLP model, (B) a general representation of layers in the MLP model.

The main difference between RNN and MLP networks is that RNN utilize a recurrent layer as the first computational layer instead of the dense layer in MLP (Figure 5). While the first dense layer in an MLP has direct and simultaneous connections to each input variable (Figure 4), computations in a recurrent layer are performed sequentially from the first to the last time step of the input variables. For example, the RNN cell (Figure 5A) behaves as follows: first, both input variables of P and PE at time t are concatenated with a "hidden" state variable h which is received from the previous computation at time step $t-1$. The concatenated vector then propagates to the internal dense layer of the RNN cell with a tanh activation function. The output of the respective dense layer is the updated hidden state variable h (at time t). Thus, there is a sequential update of the hidden state variable while moving from the first ($t-n$) to the last (t) step of the input sequence. The last updated hidden state variable h (at time t) connects to the dense (third) layer that features a single neuron and a linear activation function, which is similar to MLP. Thus, the critical hyperparameter of RNN is the length of the hidden state variable, which also determines the number of neurons on the internal dense layer of the RNN cell. For this study, we set up this number to 64. In the first step ($t-n$), the hidden state variable at time $t-n-1$ is initialized as a vector of zeros.

RNN have been specifically designed to utilize the sequential order of input variables (Kratzert et al., 2018). Yet, in many hydrological studies they showed an equivalent or even lower efficiency in comparison to MLP (Hsu et al., 2002; Nagesh Kumar et al., 2004; Taver et al., 2015). Bengio et al. (1994) showed that RNN could hardly learn long-term patterns in data. That may have a critical effect on RNN efficiency in hydrological applications, as RNN might not fully capture the memory of basin-scale hydrological processes, such as groundwater or soil water dynamics (Kratzert et al., 2018).

,

Long Short-Term Memory Network (LSTM)

RNN and LSTM networks differ in the internal computations inside the RNN and LSTM cells, respectively (Figure 5A, Figure 6A). While an RNN has only one internal state variable – the hidden state variable h –, an LSTM has an additional cell state variable c, and also utilizes a more complex logic of internal computations by using three "gates", namely *forget*, *input*, and *output* (Hochreiter and Schmidhuber, 1997). Hence, the number of parameters in LSTM cells is four times higher than in RNN cells. The key hyperparameter of an LSTM is the same as for an RNN: the length of state variables, both for hidden and cell states. As with RNN, we set this parameter to a value of 64. Further details on LSTM networks in a hydrological context can be found in Kratzert et al. (2018).

,

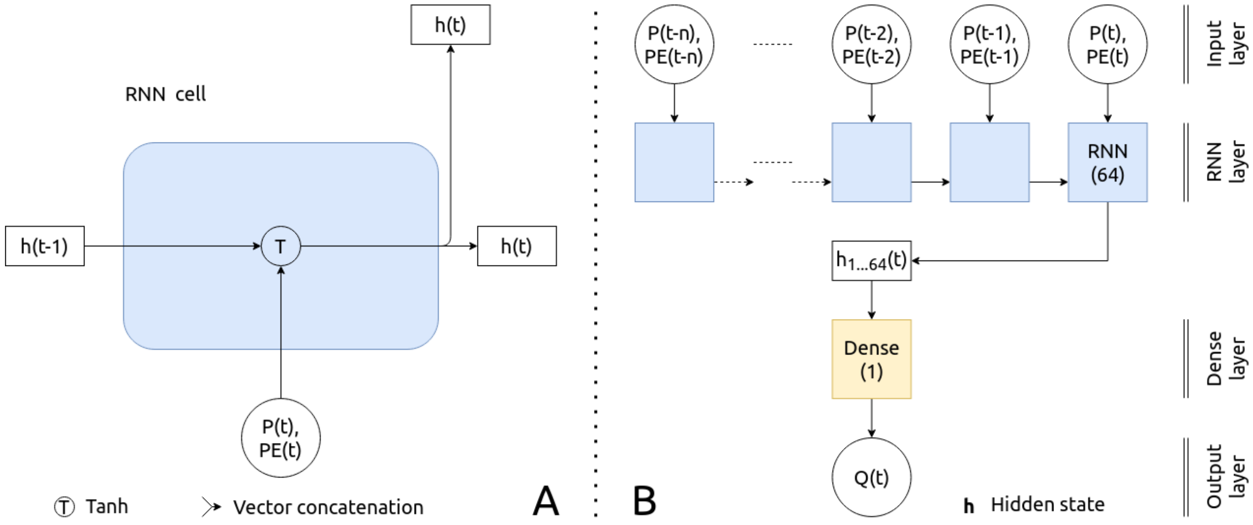

Figure 5: (A) The internal operations and activation functions of a RNN cell, (B) A general representation of the RNN model.

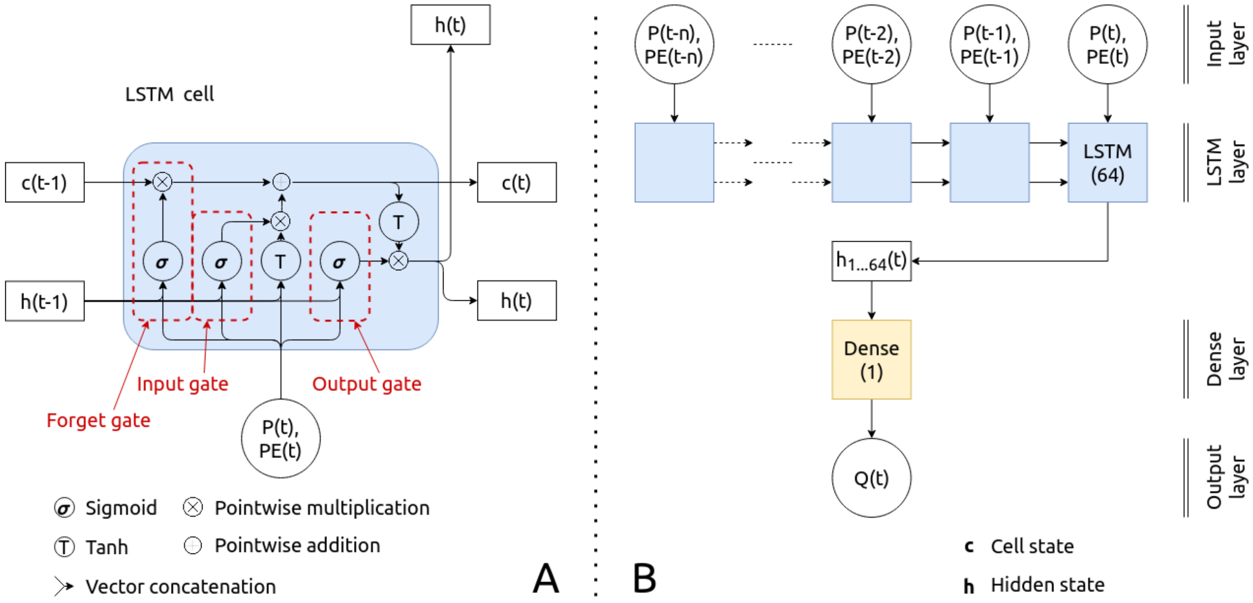

Figure 6: (A) The internal operations and activation functions of a LSTM cell, (B) A general representation of the LSTM model.

Gated Recurrent Unit Network (GRU)

The concept of GRU is similar to RNN and LSTM: computations are performed in a sequential order to account for temporal patterns in the data. GRU networks were proposed by Cho et al. (2014) with the main aim of simplifying LSTM. The authors proposed to merge the two state variables of LSTM (hidden and cell state) into one single hidden state variable h (similar to RNN), and introduced two computational "gates", namely *reset* and *update* (Figure 7A), instead of the three gates used in LSTM.

The proposed modifications have led to a reduction in the number of GRU cell parameters compared to LSTM cells. Still, the number of parameters is three times higher than that in RNN cells. Recently, Zhang et al. (2018, 2019) have shown that GRU has an efficiency comparable to LSTM. Thus, using GRU as an alternative to LSTM is promising due to a reduction in computational complexity. As with RNN and LSTM, the important hyperparameter of GRU is the length of state variable h, which we set to 64. Further details about GRU networks can be found in Zhang et al. (2018, 2019).

,

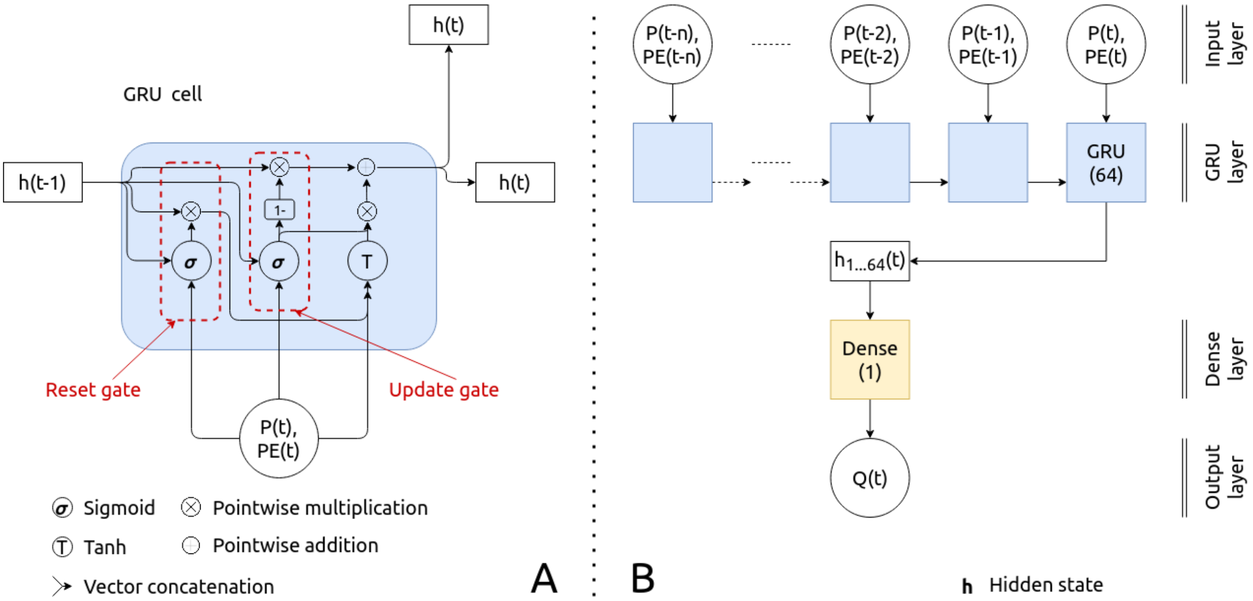

Figure 7: (A) The internal operations and activation functions of a GRU cell, (B) A general representation of the GRU model.

2.4. Model Calibration

The calibration procedure seeks for an optimal set of model parameter values that maximizes model efficiency for a given calibration period. However, the calibration procedure for conceptual hydrological models, such as GR4H, differs from that required for ANN-based models, such as MLP, RNN, LSTM, and GRU.

For conceptual hydrological models, an optimization (or calibration) algorithm typically samples a realization of model parameters from the parameters' hyperspace, based on which a runoff simulation is performed for the entire calibration period. Then, an objective function is calculated from the simulated and the observed runoff. This process is iterated until the objective function converges. In our study, the parameters of the GR4H model (Table 1) are calibrated using the differential evolution algorithm (Storn and Price, 1997) and the Nash-Sutcliffe efficiency (NSE) as an objective function (Nash and Sutcliffe, 1970). To initialize the GR4H model states, we use a warm-up period that is equal to the corresponding calibration period.

The high number of parameters in ANN-based models (Table 2) prevents the application of such standard calibration algorithms. Instead, backpropagation has become the standard for the efficient calibration of ANNs (LeCun et al., 2015), updating ANN model parameters through the calculation of the gradient of the objective function (for implementation details see, e.g., Goodfellow et al. (2016)).

Table 2
The number of parameters for different ANN-based models.

Model	# parameters
MLP	92289
RNN	4353
LSTM	17217
GRU	12929

For standard hydrological models, the objective function is calculated for the entire calibration period at each iteration of the calibration procedure. For ANN-based models, the objective function is calculated only for a subset of calibration data ("batch"). The number of samples in the batch, as well as the objective function, are hyperparameters of the calibration process. In this study, we use a batch of size 4096 together with the mean squared error as the objective function. Another hyperparameter of the ANN calibration is the number of calibration "epochs". The term epoch is similar to the number of iterations performed during the calibration procedure of hydrological models: it describes the period in which each sample in calibration data has been used for updating the ANN parameters. Here, we set up the maximum number of epochs to 1000, however, to reduce calibration time, we also utilize the early stopping technique to interrupt the calibration procedure while there is no improvement for model performance on the hold-out test period for five epochs. The respective hold-out test period is randomly assigned for every new calibration epoch and consists of 25% of the entire calibration period length. For the efficient calibration of ANN-based models, both meteorological variables and runoff have been scaled by subtracting the mean and dividing by the standard deviation (Maier and Dandy, 2001; Kratzert et al., 2018). We employ only the calibration period for the calculation of the mean and standard deviation, which were used for scaling. Further details on the ANN calibration procedure can be found in Kratzert et al. (2018). The experimental setup described above was run on a single Graphical Processing Unit (GPU, NVIDIA GTX TITAN X); the computations took a time of approximately two weeks.

The efficiency of runoff simulations was evaluated based on the widely-used Nash-Sutcliffe efficiency (NSE, Nash and Sutcliffe (1970); Equation 2). NSE is optimal with a value of 1 and positively oriented.

$$NSE = 1 - \frac{\sum_{\Omega}(Q_s - Q_o)^2}{\sum_{\Omega}(Q_o - \bar{Q}_o)^2} \tag{2}$$

where Q_s, Q_o, $\bar{Q}_o$ are simulated, observed and mean observed runoff; Ω is the period of evaluation.

3. Results and Discussion

Figure 8 illustrates the main results of our study regarding the effect of calibration data length on the performance of hourly runoff simulations by different models. The figure shows the distribution of NSE over the independent 17-year validation period (1988–2004) as a function of the number of calibration years *n*. Each of the solid lines represents the average NSE value over the available combinations of sub-periods of length *n* used for calibration. The shades represent the range between the lower and higher envelopes of the validation results, while the circles illustrate the results of each validation run.

,

Clearly, the performance of any model on the independent validation period tends to improve with an increasing length of the calibration period. This finding is expected, and confirms previous results obtained for different hydrological models (Sorooshian et al., 1983; Yapo et al., 1996; Anctil et al., 2004; Boughton, 2007; Perrin et al., 2007; Arsenault et al., 2018; Motavita et al., 2019). However, our study complements those results with various ANN-based architectures (MLP, RNN, LSTM, GRU).

More importantly, Figure 8 informs us about the number of calibration years needed to converge to a performance plateau on the validation period. GR4H, LSTM, and GRU models converge to the same maximum NSE value of 0.69. That value is significantly higher than the maximum average NSE values achieved by MLP (0.62) and RNN (0.63), respectively. Compared to LSTM and GRU, however, GR4H requires less calibration data for that convergence: GR4H reaches 95 percent of the maximum NSE of 0.69 with six years of calibration data; to achieve the same performance level, LSTM and GRU require seven and nine years, respectively. Given the large number of model parameters in LSTM and GRU, it is remarkable how quickly their performance increases with calibration data length, compared to GR4H with just four parameters. For MLP and RNN, the results are more ambiguous: the performance of the MLP model is particularly poor for less than 10 years of calibration data. With calibration data lengths of more than 10 years, the MLP catches up with the RNN, and finally outperforms it. In fact, the MLP model benefits the most from additional calibration data, both in relative and absolute terms. That is in contrast to the RNN model for which the effect of calibration data length is volatile and less pronounced: the RNN model performance is intermediate at short calibration periods, and does not benefit much from additional calibration data beyond a calibration period of ten years; model performance even drops with a calibration data length of more than 17 years. That behaviour confirms findings

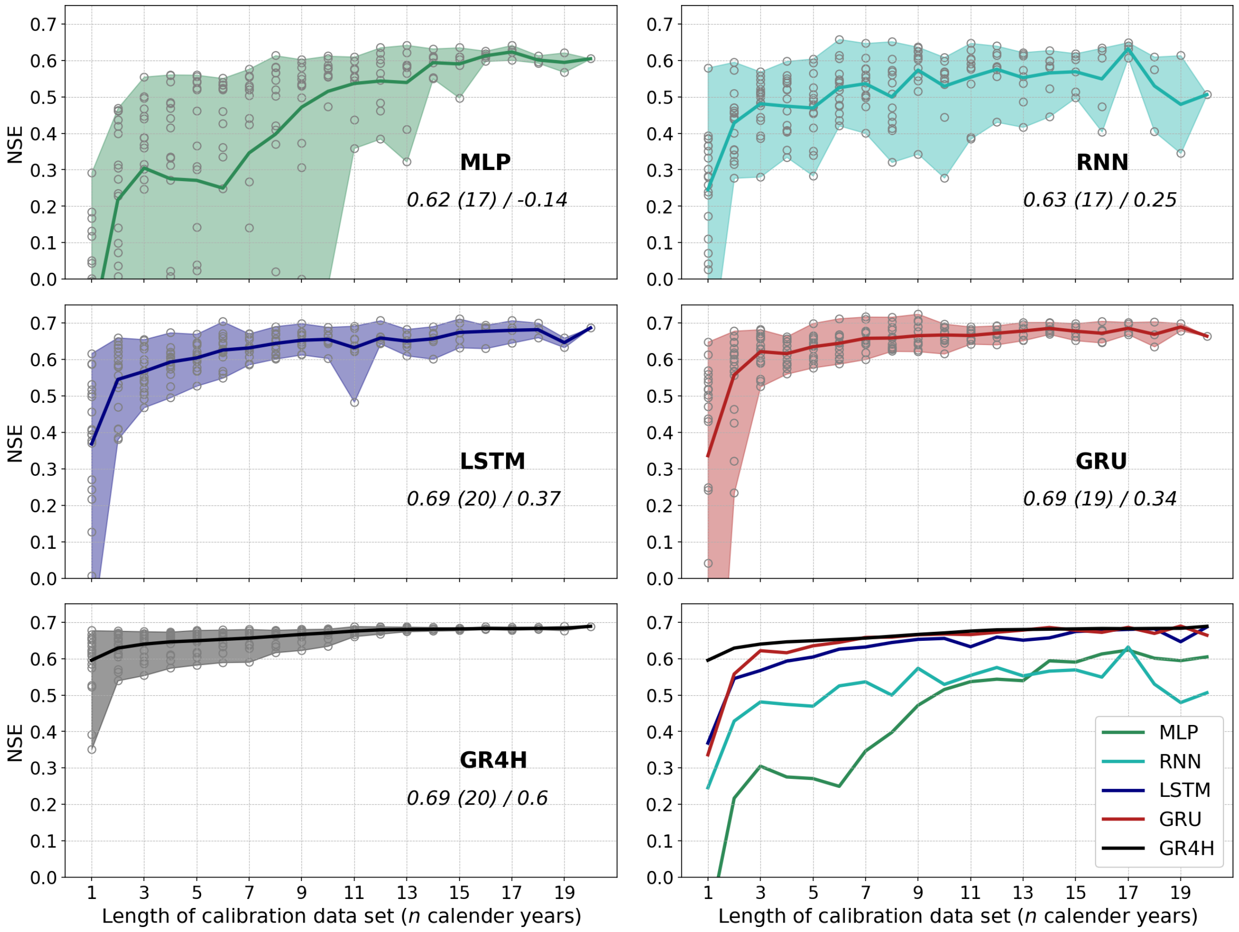

Figure 8: Validation-period NSE values with increasing number of contributing calibration years. Model names are represented in **bold**. The text in *italics* shows the highest mean NSE and the corresponding calibration data length (in parenthesis), as well as the mean NSE in case only one year is used for calibration (after the slash).

of Bengio et al. (1994) who explained the weak and unstable RNN performance as related to vanishing and exploding gradient problems, which make the calibration procedure vulnerable to non-convergence.

But while the issue of instability is most obvious for the RNN model, it is also apparent for the other ANN-based models examined in this study. With the term instability, we refer to the variability of model performance on the validation period: the variability between calibration sub-periods of the same length, but also the variability in-between sub-periods of *different* lengths. That variability – or instability – is also illustrated by Figure 8: the grey circles represent all NSE values obtained on the validation period, together with the colored shades which represent the full *range* of NSE values for a specific calibration data length. For all models (except RNN) that range generally tends to decrease with calibration data length, i.e. the shades tend to narrow. That is, of course, partly a result of the fact that the overlap between calibration sub-periods increases with sub-period length – with a calibration data length of 20 years, there is only one single calibration sub-period, hence the range of NSE values necessarily becomes zero. For that reason, the absolute widths of the shades should be interpreted with care. Instead, we should focus on the relative differences of NSE variability between the models. In that regard, we find that GR4H exhibits, for any calibration data length, the narrowest range of NSE values in comparison to any of the ANN-based models. Furthermore, the range of NSE values of the GR4H model is strictly decreasing with calibration data length – which is not the case for any of the ANN-based models. Altogether, calibration is less stable for all examined ANN-based models, in comparison to GR4H. That fundamental property of the ANN-based models is most likely a result of the large number of parameters in combination with an optimization procedure that is prone to converge to local optima. Still, the stability of the calibration tends to increase with calibration data length for all ANN-based models except RNN. In particular, the GRU model and – to a lesser extent – the LSTM model exhibit a remarkable gain in stability over calibration period lengths of one to three years (and, less pronounced, from three to eleven or twelve years, respectively). For the MLP model, such a stabilisation only occurs after about 15 years, but on a lower average performance level. As already pointed out, the RNN model does not show clear signs of a stabilisation with increasing calibration period length.

Figure 9 complements this discussion on the robustness of the calibration: it shows the relative loss of model performance (in terms of NSE) from calibration to validation, again depending on the length of the calibration period. On average, each calibrated model examined in our study significantly loses performance when being transferred from calibration to validation periods. That loss quantifies what literature refers to as "overfitting", meaning that the model overemphasizes specific relationships in the data that are not representative for other periods. As expected, the tendency to overfitting decreases with increasing calibration data length, and the GR4H model, though not immune, is least affected by overfitting. The MLP model is most affected by overfitting, but also benefits, in relative terms, most from increasing calibration data length. The GRU and LSTM models exhibit intermediate levels of overfitting which is consistent with the width of the shaded NSE ranges in Figure 8. The results of the RNN model have to be interpreted with care as the relative losses refer to a generally lower average performance level, and are thus not directly comparable to the loss curves that represent the GRU and LSTM models.

,

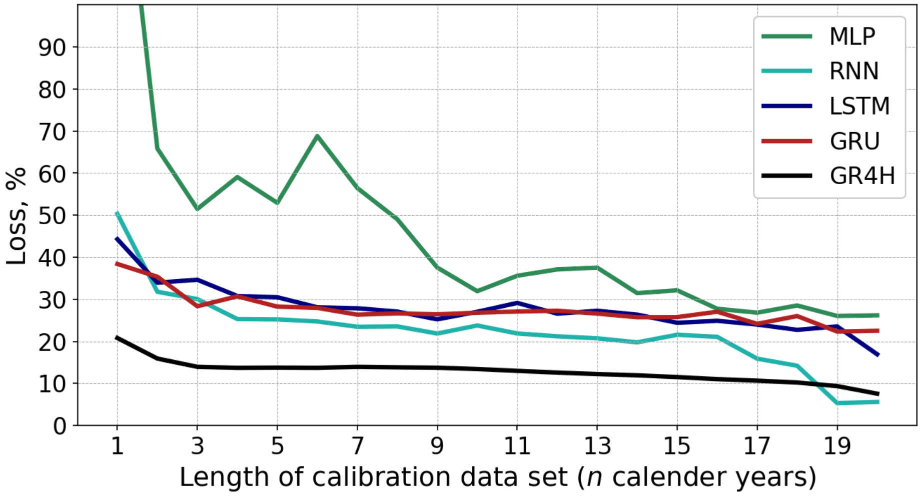

Figure 9: The relative loss of model performance (in terms of NSE) from calibration to validation, depending on the length of calibration period.

4. Conclusions

In this study, we examined the effects of calibration data length on the performance of hourly runoff simulations by different models: the lumped hydrological model GR4H, and four ANN-based models, namely MLP, RNN, LSTM, and GRU. For each model, the calibration was carried out with gradually increasing amounts of data – from one to 20 calendar years – utilizing a sliding window approach. Then, the model performance was evaluated on an independent validation period of 17 years.

Our results confirm the intuitive expectation that a longer calibration period improves model performance on the validation period. Yet, we provide new insights into how different model types and ANN architectures can benefit from additional calibration data, both in terms of their average performance, and in terms of the robustness of model calibration results.

With increasing calibration data length, the conceptual hydrological model GR4H as well as the GRU and LSTM models converge to the same maximum average performance level, an NSE of 0.69 (the term "average" refers to the average of validation performance values that correspond to calibration sub-periods of a specific length). For that convergence, the GR4H model requires less calibration data than GRU and LSTM models: it reaches 95 percent of its maximum average performance with a calibration data length of six years (GRU: seven years, LSTM: nine years). That finding confirms the expectation that the parsimonious conceptual model GR4H, with just four parameters, can be more efficiently calibrated on limited amounts of data, as its structure results from a process of continuous model development and reduction that is based on physical reason, expert knowledge and repeated application in various environments. But while GR4H requires less calibration data to converge to its maximum performance, LSTM and GRU models catch up at a remarkable rate, considering the number of parameters (12,929 for GRU, 17,217 for LSTM). The efficiency, though, at which ANN-based models learn patterns from limited amounts of data does not just depend on the number of parameters, but on its internal design. That becomes obvious from the performance of the RNN model which did not manage to take advantage of increasing calibration data lengths in the same way as its competitors, although it has far less parameters (4,353) than GRU and LSTM. The MLP model with more than 90,000 parameters did not achieve the same maximum average performance as GRU and LSTM, either, but it still showed a pronounced effect of calibration data length.

Apart from the average performance for specific calibration data lengths, we also compared the variability of validation performance between calibration sub-periods of the *same* length as well as in-between sub-periods of *different* lengths. We interpret that variability as a measure of instability, or inversely, robustness of the calibration. Except for the RNN model, the calibration of all models becomes more robust with increasing calibration data length, and GR4H is clearly more robust than all its competitors. Yet again, LSTM and particularly GRU catch up quickly and become, already after three years of calibration data length, much more robust than RNN and MLP. The latter achieves a comparable level of robustness only when more than 15 years of calibration data are used.

Altogether, we found that the conceptual hydrological model GR4H outperforms ANN-based model architectures for very brief calibration data lengths of a few years, both with regard to average performance and robustness of calibration. Yet, deep-learning architectures such as LSTM and – even more – GRU achieve, with very little additional calibration data, a comparable performance level, although they do not fully catch up with GR4H regarding the robustness of calibration. That is consistent with findings of Kratzert et al. (2018) for daily runoff simulations. In that light, neither RNN and nor MLP appear as promising alternatives, although MLP is at least able to capitalize on much longer calibration data lengths.

At this point, we should recall that the present study only examines the performance of a limited set of models in just one study catchment. It should rather be understood as a case study to demonstrate the fundamental effects of calibration data length on selected model architectures. Most certainly, the analyzed ANN architectures could be improved further: by optimizing the hyperparameters (e.g., the number of layers and their type in terms of fully-connected or recurrent), the length of the input sequence, or the length of hidden (RNN, LSTM, GRU) or cell state (LSTM) variables; by the incorporation of advanced regularization techniques; and, most importantly, by the inclusion of additional input variables. In our study, we limited the input variables to the same variables required by GR4H, and order to ensure comparability. We are aware, though, that this could be considered as "unfair" towards ANN-based architectures, as one of their strengths is to flexibly incorporate additional input variables. In the Rimbaud River catchment, snow-related processes do not play a role, but in other catchments affected by snow, additional variables related to temperature and the radiation balance might have a strong effect. Furthermore, for larger catchments, the effects of runoff concentration in the landscape might become more important in relation to the effects of local runoff generation.

We have also highlighted the vulnerability of ANN-based architectures to overfitting: in comparison to a conceptual hydrological model, the loss of performance from calibration to validation is high. This is to be expected with models that have thousands and thousands of parameters. Still, we consider it important to emphasize this fact since hydrological modelers, again and again, tend to boost the length of the calibration period at the cost of an independent validation. And while we could show that even a parsimonious conceptual model such as GR4H is not entirely immune to overfitting, the risk that emanates from dropping the validation step for ANN-based models becomes just unacceptable. Ironically, the incentive to boost the length of the calibration period is particularly high for such models, as also shown in our results. That is why calibration/validation experiments such as the ones presented in this study are particularly helpful in scrutinizing the length of the calibration period for a specific model. To that end, we provide the corresponding software code along with the present paper, as a basis for prospective research (see `https://github.com/hydrogo/KALI`).

Competing interests

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Funding

G.A. was financially supported by Geo.X, the Research Network for Geosciences in Berlin and Potsdam.

Acknowledgements

We thank Dr. Guillaume Thirel (INRAE, France) for his help in obtaining the meteorological and discharge gauge data.

Data Availability

Discharge data for Rimbaud River as well as rain gauge records for the area Real Collobrières is available upon request from the RECOVER research laboratory at INRAE (`https://www6.paca.inrae.fr/recover`). The data for the SAFRAN reanalysis are available upon request from Meteo France.

Computer Code Availability

We support this paper by making the entire computational workflow available online in the corresponding repository on Github (`https://github.com/hydrogo/KALI`, accessed 20.03.2020). The code is written in the Python 3 programming language (`https://www.python.org/`, accessed 20.03.2020) and is based on open-source software libraries, namely *Numpy* (Oliphant, 2006), *Pandas* (McKinney et al., 2010), *Scipy* (Virtanen et al., 2020), *Numba* (Lam et al., 2015), *Tensorflow* (Abadi et al., 2015), and *Keras* (Chollet et al., 2015). We use Jupyter notebooks (`https://jupyter.org`, accessed 20.03.2020) for code development and interactive analysis of obtained results, as well as the *Matplotlib* library (Hunter, 2007) for plotting.

CRediT authorship contribution statement

Georgy Ayzel: ran the experiments, performed the analysis, and wrote the manuscript.. **Maik Heistermann:** co-authored the manuscript..

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G.S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., Zheng, X., 2015. TensorFlow: Large-scale machine learning on heterogeneous systems. URL: `https://www.tensorflow.org/`. software available from tensorflow.org.

Allen, R.G., Pereira, L.S., Raes, D., Smith, M., 1998. Crop evapotranspiration – guidelines for computing crop water requirements. fao irrigation and drainage paper 56. FAO, Rome, Italy 300, D05109. Http://www.fao.org/3/x0490e/x0490e00.htm.

Anctil, F., Perrin, C., Andréassian, V., 2004. Impact of the length of observed records on the performance of ann and of conceptual parsimonious rainfall-runoff forecasting models. Environmental Modelling & Software 19, 357 – 368. URL: `http://www.sciencedirect.com/science/article/pii/S136481520300135X`, doi:`https://doi.org/10.1016/S1364-8152(03)00135-X`.

Arsenault, R., Brissette, F., Martel, J.L., 2018. The hazards of split-sample validation in hydrological model calibration. Journal of Hydrology 566, 346 – 362. URL: `http://www.sciencedirect.com/science/article/pii/S0022169418307145`, doi:`https://doi.org/10.1016/j.jhydrol.2018.09.027`.

Bengio, Y., Simard, P., Frasconi, P., 1994. Learning long-term dependencies with gradient descent is difficult. IEEE Transactions on Neural Networks 5, 157–166. doi:`10.1109/72.279181`.

de Boer-Euser, T., Bouaziz, L., De Niel, J., Brauer, C., Dewals, B., Drogue, G., Fenicia, F., Grelier, B., Nossent, J., Pereira, F., Savenije, H., Thirel, G., Willems, P., 2017. Looking beyond general metrics for model comparison – lessons from an international model intercomparison study. Hydrology and Earth System Sciences 21, 423–440. URL: `https://www.hydrol-earth-syst-sci.net/21/423/2017/`, doi:`10.5194/hess-21-423-2017`.

Boughton, W., 2007. Effect of data length on rainfall–runoff modelling. Environmental Modelling & Software 22, 406 – 413. URL: http://www.sciencedirect.com/science/article/pii/S1364815206000077, doi:https://doi.org/10.1016/j.envsoft.2006.01.001. special section: Advanced Technology for Environmental Modelling.

Cho, K., van Merrienboer, B., Gülçehre, Ç., Bougares, F., Schwenk, H., Bengio, Y., 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. URL: http://arxiv.org/abs/1406.1078.

Chollet, F., et al., 2015. Keras. https://keras.io.

Cigizoglu, H.K., Kisi, O., 2005. Flow prediction by three back propagation techniques using k-fold partitioning of neural network training data. Hydrology Research 36, 49–64. URL: https://doi.org/10.2166/nh.2005.0005, doi:10.2166/nh.2005.0005, arXiv:https://iwaponline.com/hr/article-pdf/36/1/49/364590/49.pdf.

Coron, L., Andréassian, V., Perrin, C., Lerat, J., Vaze, J., Bourqui, M., Hendrickx, F., 2012. Crash testing hydrological models in contrasted climate conditions: An experiment on 216 australian catchments. Water Resources Research 48. URL: https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2011WR011721, doi:10.1029/2011WR011721, arXiv:https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1029/2011WR011721.

Dawson, C.W., Wilby, R.L., 2001. Hydrological modelling using artificial neural networks. Progress in Physical Geography: Earth and Environment 25, 80–108. URL: https://doi.org/10.1177/030913330102500104, doi:10.1177/030913330102500104, arXiv:https://doi.org/10.1177/030913330102500104.

van Esse, W.R., Perrin, C., Booij, M.J., Augustijn, D.C.M., Fenicia, F., Kavetski, D., Lobligeois, F., 2013. The influence of conceptual model structure on model performance: a comparative study for 237 french catchments. Hydrology and Earth System Sciences 17, 4227–4239. URL: https://www.hydrol-earth-syst-sci.net/17/4227/2013/, doi:10.5194/hess-17-4227-2013.

Fowler, K., Peel, M., Western, A., Zhang, L., 2018. Improved rainfall-runoff calibration for drying climate: Choice of objective function. Water Resources Research 54, 3392–3408. URL: https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2017WR022466, doi:10.1029/2017WR022466, arXiv:https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1029/2017WR022466.

Goodfellow, I., Bengio, Y., Courville, A., 2016. Deep learning. MIT press. http://www.deeplearningbook.org.

Hochreiter, S., Schmidhuber, J., 1997. Long short-term memory. Neural Computation 9, 1735–1780. URL: https://doi.org/10.1162/neco.1997.9.8.1735, doi:10.1162/neco.1997.9.8.1735, arXiv:https://doi.org/10.1162/neco.1997.9.8.1735.

Hsu, K.l., Gupta, H.V., Gao, X., Sorooshian, S., Imam, B., 2002. Self-organizing linear output map (solo): An artificial neural network suitable for hydrologic modeling and analysis. Water Resources Research 38, 1–17. URL: https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2001WR000795, doi:10.1029/2001WR000795, arXiv:https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1029/2001WR000795.

Hunter, J.D., 2007. Matplotlib: A 2d graphics environment. Computing in Science Engineering 9, 90–95. doi:10.1109/MCSE.2007.55.

Kratzert, F., Klotz, D., Brenner, C., Schulz, K., Herrnegger, M., 2018. Rainfall–runoff modelling using long short-term memory (lstm) networks. Hydrology and Earth System Sciences 22, 6005–6022. URL: https://www.hydrol-earth-syst-sci.net/22/6005/2018/, doi:10.5194/hess-22-6005-2018.

Kratzert, F., Klotz, D., Shalev, G., Klambauer, G., Hochreiter, S., Nearing, G., 2019. Towards learning universal, regional, and local hydrological behaviors via machine learning applied to large-sample datasets. Hydrology and Earth System Sciences 23, 5089–5110. URL: https://www.hydrol-earth-syst-sci.net/23/5089/2019/, doi:10.5194/hess-23-5089-2019.

Lam, S.K., Pitrou, A., Seibert, S., 2015. Numba: A llvm-based python jit compiler, in: Proceedings of the Second Workshop on the LLVM Compiler Infrastructure in HPC, Association for Computing Machinery, New York, NY, USA. URL: https://doi.org/10.1145/2833157.2833162, doi:10.1145/2833157.2833162.

Le Moine, N., Andréassian, V., Mathevet, T., 2008. Confronting surface- and groundwater balances on the La Rochefoucauld-Touvre karstic system (Charente, France). Water Resources Research 44. URL: https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2007WR005984, doi:10.1029/2007WR005984, arXiv:https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1029/2007WR005984.

LeCun, Y., Bengio, Y., Hinton, G., 2015. Deep learning. Nature 521, 436–444. URL: http://www.nature.com/articles/nature14539, doi:10.1038/nature14539.

Li, C.z., Wang, H., Liu, J., Yan, D.h., Yu, F.l., Zhang, L., 2010. Effect of calibration data series length on performance and optimal parameters of hydrological model. Water Science and Engineering 3, 378–393. URL: https://www.sciencedirect.com/science/article/pii/S1674237015301289, doi:10.3882/J.ISSN.1674-2370.2010.04.002.

Li, M., Wang, Q., Robertson, D.E., Bennett, J.C., 2017. Improved error modelling for streamflow forecasting at hourly time steps by splitting hydrographs into rising and falling limbs. Journal of Hydrology 555, 586 – 599. URL: http://www.sciencedirect.com/science/article/pii/S0022169417307345, doi:https://doi.org/10.1016/j.jhydrol.2017.10.057.

Maier, H., Dandy, G., 2001. Neural network based modelling of environmental variables: A systematic approach. Mathematical and Computer Modelling 33, 669 – 682. URL: http://www.sciencedirect.com/science/article/pii/S0895717700002715, doi:https://doi.org/10.1016/S0895-7177(00)00271-5.

Maier, H.R., Dandy, G.C., 2000. Neural networks for the prediction and forecasting of water resources variables: a review of modelling issues and applications. Environmental Modelling & Software 15, 101 – 124. URL: http://www.sciencedirect.com/science/article/pii/S1364815299000079, doi:https://doi.org/10.1016/S1364-8152(99)00007-9.

Maier, H.R., Jain, A., Dandy, G.C., Sudheer, K., 2010. Methods used for the development of neural networks for the prediction of water resource variables in river systems: Current status and future directions. Environmental Modelling & Software 25, 891 – 909. URL: http://www.sciencedirect.com/science/article/pii/S1364815210000411, doi:https://doi.org/10.1016/j.envsoft.2010.02.003.

Marçais, J., de Dreuzy, J.R., 2017. Prospective interest of deep learning for hydrological inference. Groundwater 55, 688–692. URL: https://ngwa.onlinelibrary.wiley.com/doi/abs/10.1111/gwat.12557, doi:10.1111/gwat.12557, arXiv:https://ngwa.onlinelibrary.wiley.com/doi/pdf/10.1111/gwat.12557.

Mathevet, T., 2005. Which rainfall-runoff model at the hourly time-step? empirical development and intercomparison of rainfall runoff model on a large sample of watersheds. Ph. D. thesis, Cemagref, École nationale du génie rural, des eaux et des forêts Univ.

McKinney, W., et al., 2010. Data structures for statistical computing in python, in: Proceedings of the 9th Python in Science Conference, Austin, TX. pp. 51–56.

Motavita, D., Chow, R., Guthke, A., Nowak, W., 2019. The comprehensive differential split-sample test: A stress-test for hydrological model robustness under climate variability. Journal of Hydrology 573, 501 – 515. URL: http://www.sciencedirect.com/science/article/pii/S0022169419302835, doi:https://doi.org/10.1016/j.jhydrol.2019.03.054.

Mount, N., Maier, H., Toth, E., Elshorbagy, A., Solomatine, D., Chang, F.J., Abrahart, R., 2016. Data-driven modelling approaches for socio-hydrology: opportunities and challenges within the panta rhei science plan. Hydrological Sciences Journal 61, 1192–1208. URL: https://doi.org/10.1080/02626667.2016.1159683, doi:10.1080/02626667.2016.1159683, arXiv:https://doi.org/10.1080/02626667.2016.1159683.

Nagesh Kumar, D., Srinivasa Raju, K., Sathish, T., 2004. River Flow Forecasting using Recurrent Neural Networks. Water Resources Management 18, 143–161. URL: https://doi.org/10.1023/B:WARM.0000024727.94701.12, doi:10.1023/B:WARM.0000024727.94701.12.

Nash, J., Sutcliffe, J., 1970. River flow forecasting through conceptual models part i — a discussion of principles. Journal of Hydrology 10, 282 – 290. URL: http://www.sciencedirect.com/science/article/pii/0022169470902556, doi:https://doi.org/10.1016/0022-1694(70)90255-6.

Nijzink, R.C., Almeida, S., Pechlivanidis, I.G., Capell, R., Gustafssons, D., Arheimer, B., Parajka, J., Freer, J., Han, D., Wagener, T., van Nooijen, R.R.P., Savenije, H.H.G., Hrachowitz, M., 2018. Constraining conceptual hydrological models with multiple information sources. Water Resources Research 54, 8332–8362. URL: https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2017WR021895, doi:10.1029/2017WR021895, arXiv:https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1029/2017WR021895.

Oliphant, T.E., 2006. A guide to NumPy. volume 1. Trelgol Publishing USA.

Perrin, C., Michel, C., Andréassian, V., 2003. Improvement of a parsimonious model for streamflow simulation. Journal of Hydrology 279, 275 – 289. URL: http://www.sciencedirect.com/science/article/pii/S0022169403002257, doi:https://doi.org/10.1016/S0022-1694(03)00225-7.

Perrin, C., Oudin, L., Andreassian, V., Rojas-serna, C., Michel, C., Mathevet, T., 2007. Impact of limited streamflow data on the efficiency and the parameters of rainfall—runoff models. Hydrological Sciences Journal 52, 131–151. URL: http://www.tandfonline.com/doi/abs/10.1623/hysj.52.1.131, doi:10.1623/hysj.52.1.131.

Refsgaard, J.C., Henriksen, H.J., Harrar, W.G., Scholten, H., Kassahun, A., 2005. Quality assurance in model based water management – review of existing practice and outline of new approaches. Environmental Modelling & Software 20, 1201 – 1215. URL: http://www.sciencedirect.com/science/article/pii/S1364815204002087, doi:https://doi.org/10.1016/j.envsoft.2004.07.006.

Reichstein, M., Camps-Valls, G., Stevens, B., Jung, M., Denzler, J., Carvalhais, N., Prabhat, 2019. Deep learning and process understanding for data-driven Earth system science. Nature 566, 195–204. URL: https://doi.org/10.1038/s41586-019-0912-1, doi:10.1038/s41586-019-0912-1.

Shen, C., 2018. A transdisciplinary review of deep learning research and its relevance for water resources scientists. Water Resources Research 54, 8558–8593. URL: https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2018WR022643, doi:10.1029/2018WR022643, arXiv:https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1029/2018WR022643.

Sorooshian, S., Gupta, V.K., Fulton, J.L., 1983. Evaluation of maximum likelihood parameter estimation techniques for conceptual rainfall-runoff models: Influence of calibration data variability and length on model credibility. Water Resources Research 19, 251–259. URL: https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/WR019i001p00251, doi:10.1029/WR019i001p00251, arXiv:https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1029/WR019i001p00251.

Storn, R., Price, K., 1997. Differential Evolution – A Simple and Efficient Heuristic for global Optimization over Continuous Spaces. Journal of Global Optimization 11, 341–359. URL: http://link.springer.com/10.1023/A:1008202821328http://dx.doi.org/10.1023/A:1008202821328, doi:10.1023/A:1008202821328.

Taver, V., Johannet, A., Borrell-Estupina, V., Pistre, S., 2015. Feed-forward vs recurrent neural network models for non-stationarity modelling using data assimilation and adaptivity. Hydrological Sciences Journal 60, 1242–1265. URL: https://doi.org/10.1080/02626667.2014.967696, doi:10.1080/02626667.2014.967696, arXiv:https://doi.org/10.1080/02626667.2014.967696.

Thirel, G., Andréassian, V., Perrin, C., Audouy, J.N., Berthet, L., Edwards, P., Folton, N., Furusho, C., Kuentz, A., Lerat, J., Lindström, G., Martin, E., Mathevet, T., Merz, R., Parajka, J., Ruelland, D., Vaze, J., 2015. Hydrology under change: an evaluation protocol to investigate how hydrological models deal with changing catchments. Hydrological Sciences Journal 60, 1184–1199. URL: https://doi.org/10.1080/02626667.2014.967248, doi:10.1080/02626667.2014.967248, arXiv:https://doi.org/10.1080/02626667.2014.967248.

Toth, E., Brath, A., 2007. Multistep ahead streamflow forecasting: Role of calibration data in conceptual and neural network modeling. Water Resources Research 43. URL: https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2006WR005383, doi:10.1029/2006WR005383, arXiv:https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1029/2006WR005383.

Vidal, J.P., Martin, E., Franchistéguy, L., Baillon, M., Soubeyroux, J.M., 2010. A 50-year high-resolution atmospheric reanalysis over france with the safran system. International Journal of Climatology 30, 1627–1644. URL: https://rmets.onlinelibrary.wiley.com/doi/abs/10.1002/joc.2003, doi:10.1002/joc.2003, arXiv:https://rmets.onlinelibrary.wiley.com/doi/pdf/10.1002/joc.2003.

Virtanen, P., Gommers, R., Oliphant, T.E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., et al., 2020. Scipy 1.0: fundamental algorithms for scientific computing in python. Nature methods , 1–12doi:10.1038/s41592-019-0686-2.

Yapo, P.O., Gupta, H.V., Sorooshian, S., 1996. Automatic calibration of conceptual rainfall-runoff models: sensitivity to calibration data. Journal of Hydrology 181, 23 – 48. URL: http://www.sciencedirect.com/science/article/pii/0022169495029184, doi:https://doi.org/10.1016/0022-1694(95)02918-4.

Zhang, D., Lindholm, G., Ratnaweera, H., 2018. Use long short-term memory to enhance internet of things for combined sewer overflow monitoring. Journal of Hydrology 556, 409 – 418. URL: http://www.sciencedirect.com/science/article/pii/S0022169417307722, doi:https://doi.org/10.1016/j.jhydrol.2017.11.018.

Zhang, D., Peng, Q., Lin, J., Wang, D., Liu, X., Zhuang, J., 2019. Simulating reservoir operation using a recurrent neural network algorithm. Water 11. URL: https://www.mdpi.com/2073-4441/11/4/865, doi:10.3390/w11040865.