Author Response to Reviews of

How can regime characteristics of catchments help in training of local and regional LSTM based runoff models?

Reyhaneh Hashemi et al.
HESS, doi:10.5194/hess-2021-511

RC: Reviewer Comment AR: Author Response RV: Revision

Dear Editor, Dear Referees,

We would like to invite you to find in the present document a summary of the major changes we have made to the paper (§1), our response to the report provided by Editor Efrat Morin (§2), our point by point response to the review made by Referee John Quilty (§3), and our point by point response to Anonymous Referee #2's review (§4).

Kind regards,
Authors

1. SUMMARY OF THE MAJOR CHANGES

In the revised version, we have made the following major changes:

1. Sample — we have refined the initial sample by excluding:
   (a) the catchments with less than 30 years of full discharge record,
   (b) the influenced catchments with a degree of influence[1] greater than or equal to 0.1 ($d_i \geq 0.1$).

   The reduced sample that we have used for the revised manuscript has 361 catchments. We would also like to note that we do not have any more such an HP sample since the hyperparameter tuning approach has been fundamentally revised — according to Anonymous Referee #2's review.
2. Neural network architecture — we have dropped the S2 architecture of the old manuscript, which had 2 LSTM layers. In the revised manuscript, we have used the S1 architecture (from the old manuscript) for all LSTM experiments.
3. Learning rate — we have changed our learning rate to 0.0001. This is the learning rate that previous studies have reported (Kratzert et al., 2018; Lees et al., 2021).
4. Tuning hyperparameters — we have made the following changes:
   (a) in the revised manuscript, we have not tuned the "number of LSTM layers" and "batch size",

[1] The d_i variable is defined in lines 95-99 of the old manuscript.

(b) as suggested by Anonymous Referee #2, we have included the "number of hidden units" in the tuning hyperparameters along with lookback. Following the previous studies, we have also added "dropout rate" to the hyperparameters to be tuned.

In the revised manuscript, we have thus considered six variations of lookback — $30, 60, 90, 180, 365, 730$ [days] — and 3 variations for hidden unit size — $64, 128, 256$ — and 3 dropout rates — $0.0, 0.2, 0.4$.

5. Hyperparameter tuning approach — we have performed a full "catchment wise" hyperparameter tuning for LSTMs trained on individual catchments and a full "model wise" hyperparameter tuning for LSTMs trained on a group of catchments, as proposed by Anonymous Referee #2. We therefore performed 54 hyperparameter tunings for each of the LSTMs found in the revised manuscript.

6. Approach to selection of the best hyperparameter set — we have investigated a new approach for group trained LSTMs. (Please see §3.5 of the revised manuscript.)

7. Static attributes — following the recommendation from Anonymous Referee #2, we have included four new attributes — mean daily solid precipitation, mean daily solid precipitation, mean daily potential evapotranspiration, and median altitude — in the static inputs of group trained LSTMs.

8. Research questions of the paper are revisited. Please see Introduction of the revised manuscript.

9. The title of the paper is changed to "How can we benefit from regime information to make use of LSTM runoff models more effectively?"

2. EDITOR Efrat Morin

Dear authors
We had review reports from two reviewers. There are a few major concerns that were raised mainly by reviewer #2, mostly focused on hypertuning. I would like to emphasize two points:
- The hypertuning should include more important LSTM parameters, including the cell state dimension, the sequence length (lookback), and others.
- For a fair comparison, hypertuning has to be done separately for each catchment group (including the group of all catchments). The hyperparameters found for the single catchments do not necessarily work for the catchment groups. If you decide to submit a revised paper, please address the above and other comments of the reviewers.

AR: We would like to thank you for the provided report and your conclusion, with which we fully agree. In the revised manuscript, we have fully followed the approach suggested by Anonymous Referee #2 and have totally revised our hyperparameter tuning approach. We have included "hidden unit size" in our tuning hyperparameters. Please see §4.2.1 and §4.2.3 for further details.

3. REFEREE John Quilty

3.1. General comments

RC: This paper carefully studies long short-term memory networks (LSTM) for rainfall-runoff prediction, using a large-sample of catchments in France. The key focus is on exploring local and regional models as well as the impact of the 'lookback' period, an important hyper-parameter of LSTM, with respect to predictive performance and physical understanding of the model results. The authors include well-thought out experiments to identify the impact of the lookback period and cases where local and regional LSTM models are best suited. The authors also benchmark LSTM with GR4J, due to its useful ability to capture ground water exchanges with aquifers and/or between catchments.

The authors spend a considerable amount of effort on tying the performance of LSTM, locally and regionally, to a physical understanding of the results. Some examples include the comparison between local and regional LSTM models with GR4J in terms of a water balance exercise in §5.3 as well as the ability of LSTM to predict runoff in controlled catchments at a higher degree of accuracy than GR4J (in §5.4). This paper also presents findings (e.g., LSTM does not necessarily outperform simple conceptual rainfall-runoff models) that are counter to other recent studies on LSTM (Gauch et al., 2021; Kratzert et al., 2019; Lees et al., 2021); in all such cases, the authors take the time to carefully describe potential reasons for these differences. Overall, this paper is very strong and I could not find much to criticize. The methodology seems correct. The figures are very nice and easy to interpret and I did not find any of the content, tables, or figures to be superfluous.

I suspect this paper will be very useful to other researchers interested in exploiting the generality of machine learning for hydrological modelling and rainfall-runoff prediction, in particular. I think the paper only needs some very minor corrections and some additional brief explanations (as noted below). Afterwards, the paper could be published.

AR: We would like to thank you for reading our manuscript carefully and with interest and for your encouraging comments. We are very pleased to read that you find our paper useful. We would like to invite you to find our

point by point response[1] to your comments in the following subsections.

3.2. Specific comments

3.2.1 Line 32

RC: Does LSTM also help mitigate against exploding gradients? If so, this would be good to mention as well.

AR: Thank you for this relevant question. The answer is, yes. This is because vanishing and exploding gradients both result from the same mathematical challenge when optimizing neural networks (NNs) with very high non linearities, although the latter case (exploding gradients) is less frequent (Goodfellow et al., 2016). A full description of how LSTM overcomes both vanishing and exploding gradients is given in Hochreiter and Schmidhuber (1997). To put it briefly and in very simple words, in deeply nested NNs, such as (vanilla) RNNs when the length of processing sequence (T) becomes large, it happens that a factor — which in the problematic case is not close to an absolute value of 1 — gets multiplied by itself over and over — T times — due to the chain rule of the calculus. Therefore, the result will either exponentially shrink — if the factor is initially < 1 — or exponentially grow — if the factor is initially > 1 — and this is where the vanishing or exploding gradient issue arises. LSTM is designed to establish derivatives that neither vanish nor explode.

RV: Following the reviewer's suggestion, we have included this explanation in the revised manuscript (§3.1). We have also rewritten the whole section on LSTM's principles to make it more clear and tractable. We have removed unclear and confusing explanations and have given practical information on computation of different variables of the forward pass in an LSTM cell.

3.2.2 Figure 1

RC: It would be good to include a description of the acronyms HP and FR in the figure caption (since it is unclear what these acronyms represent).

AR: Thank you for this comment and we agree with you.

RV: Following your suggestion and the point made by the Anonymous Referee #2 on our naming strategy, we have revised all instances of acronym/letter based names — all such instances are replaced by descriptive names.

3.2.3 Grammatical corrections

RC: For the most part, the paper is well-written but there are a number of grammatical errors. I stopped correcting such errors around line 154. I recommend that a very carefully read through the paper be completed before re-submission.

AR: Thank you for spotting these grammatical errors. We agree with you about the section containing Line 154 and its surrounding. We found it also a bit stiff and wordy.

RV: We have thus totally rewritten this section to make it more concise, clear, and useful. We hope it reads well now in the revised manuscript (§3.1). We have also proofread the entire paper to check for any further errors.

[1] All line and figure mentions found in the title of the subsections of this section regard the old manuscript.

3.2.4 Line 165

RC: The sentence on line 165 can be moved to the last sentence of the same sub-section. A short sentence, 'The main equations used in LSTM are as follows (Ref, XXX):' can be used in it's place.

AR: Thank you for this suggestion.

RV: We have rewritten this subsection, as mentioned in the above RV.

3.2.5 Lines 205-206

RC: Is this sort of standardization the most appropriate for LSTM? Since sigmoid and tanh activation functions are used, should not the data be scaled to [0,1] or [-1,1] as these ranges match the output ranges of the (previously mentioned) activation functions? Perhaps others have adopted the form of standardization adopted here, if so, can the authors indicate this?

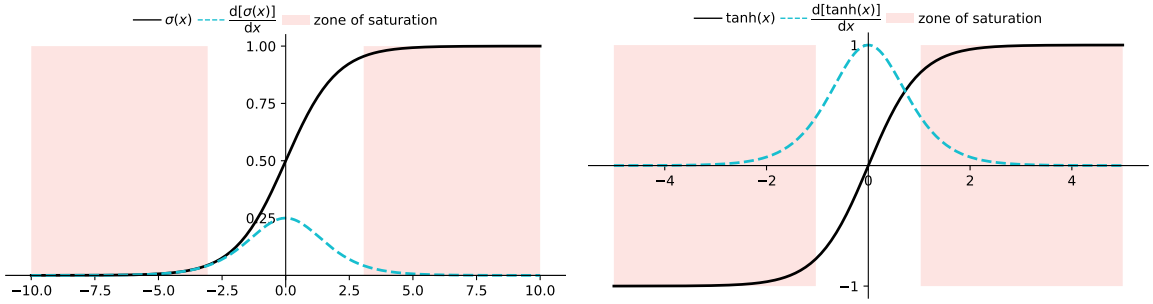
AR: Thank you for this interesting question. LeCun et al. (2012) explain that why centering the input data around 0 and scaling them by the standard deviation is typically a good idea and usually makes gradient descent converge faster. Besides, we could not in principle benefit from a [0, 1] scaling since the temperature feature might include negative values.
The interesting point of standardization comes when we investigate how the derivative of different activation functions changes with respect to the range of input data. Please note that here we are talking about the activation function for the hidden layers and not the last layer, which is given by the type of the problem — for instance, Softmax for multi class classification, Sigmoid for binary classification, and Identity for regression. Looking at Fig. 1 (of the present document), it turns out that the Sigmoid ($\sigma(x)$) and $\tanh$ functions suffer from a problem — their derivative gets saturated very quickly. By the term "saturation", we mean that their derivative approaches very quickly to zero indicating that weights can not get updated effectively at these points thus the NN can not learn effectively. We observe this problem almost everywhere except in the small region in the middle centered around 0 where the derivative is the most dynamic. Therefore, having the inputs centered around 0 with a variance of 1 would also help fall in the useful area of these functions. Please note that even in their dynamic region the derivatives are small and could bring about the vanishing gradient problem in NNs with high non linearities.
Now, you might ask why not simply using an activation function that does not have vanishing gradients, for instance ReLu? The answer is that the ReLu activation function proved to be typically a more appropriate default choice — if we are allowed to use it. For LSTM, there is a specific reason for which we need to stick with the Sigmoid function in gates, despite the mentioned disadvantages. Indeed, the it plays a "gate" role — a function granting us a value between 0 and 1 — and it is not possible to replace it by ReLu or any other activation functions not having this output range.

RV: We have included a summary of this answer in §3.2 of the revised manuscript, where we have also referred to Kratzert et al. (2018), who standardized their data using the mean and standard deviation of the training data.

3.2.6 Adam algorithm

RC: What were the hyper-parameters (alpha, beta_1, beta_2) set to in the Adam algorithm?

AR: Except for learning rate (α) that we have set to 0.0001 in the revised manuscript, we have kept all other arguments, including β_1 and β_2 (L^1 and L^2 norms), at their default values in Keras (Chollet et al., 2015):



(a) $\sigma(x)$ vs. $\frac{\mathrm{d}[\sigma(x)]}{\mathrm{d}(x)}$ (b) $\tanh(x)$ vs. $\frac{\mathrm{d}[\tanh(x)]}{\mathrm{d}(x)}$

Figure 1: Sensitivity of the derivative of the Sigmoid and $\tanh$ functions to the range of input data.

```
tf.keras.optimizers.Adam(
    learning_rate =0.0001,
    beta_1=0.9,
    beta_2=0.999,
    epsilon=1e-07,
    amsgrad=False,
    name="Adam",
    )
```

RV: In the revised manuscript, we have included a phrase indicating this setting (§3.5).

3.2.7 Equation 17

RC: what does epsilon represent?

AR: Thank you for noticing this — we had forgotten to indicate that Kratzert et al. (2019) added this term (ϵ) to the denominator in the equation of NSE^* so that the loss function would not explode when s was very close to 0 (catchments with very small discharge variance).

RV: We have updated the text to include this explanation (§3.5).

3.3. Technical corrections

RC: Lines 16, 35, 57, 61, 73, 135, 148-150, 154, Figure 6

AR: Thank you for spotting these errors.

RV: The manuscript is (almost) rewritten. We tried to avoid such errors in the new manuscript and hope that such errors have not accrued in the new version.

4. ANONYMOUS REFEREE #2

AR: We would like to thank you, Anonymous Referee #2, for your review and constructive thoughts, which have greatly improved our paper. You were critical of the design of our experiments and hyperparameter tuning approach and had identified technical issues warranting a major revision of these two aspects. We agree with you. We have made all the revisions you had suggested. We have updated our point by point response to your individual comments according to these revisions and would like to invite you to find it below.

4.1. Summary of review

RC: This paper addresses two research questions related to the use of LSTMs for rainfall-runoff modeling: (1) Does appropriate sequence length depend on hydrological regime, and (2) should LSTM training be done on hydrologically similar basins?
To state my opinion up front, I have run similar experiments (unpublished) and found results that are qualitatively different than what are reported here. There are several technical issues in this paper (overall, the methodology is not appropriate for testing the stated hypotheses), and it might be worth addressing those before we look carefully at the results.

AR: We thank the reviewer for their interest in performing similar experiments. We would be happy to engage in an ongoing dialogue with the reviewer about the details of their experiments since without further information, in particular, on the hydro-geo-climatic context of their data, it would be hard to provide a definite explanation. Indeed, such discrepancies could be investigated at different levels. At the highest level, we would conjecture that the reason lies in definition of the homogeneity component. We believe that the hydrological similarity rule — i.e., the regime classification — is a crucial question. Given the term "similar experiments", we could think of the following two cases.

Case 1 The exact same classification is applied to a sample in a non French context. In this case, we would be afraid that the exact same rule would not be immediately applicable to other climatic contexts. The following elaboration on our classification approach aims to underline how it is intensely context dependent — in terms of number of classes, criteria, and thresholds.

As a property of the French context, we knew in advance that there existed five main regime patterns, which we named Uniform, Mediterranean, Oceanic, Nival Pluvial, and Nival. Therefore, any catchment in our sample could be classified in one of the five categories. Using the fact sheets available at https://webgr.inrae.fr/activites/base-de-donnees/, we tried to identify hydro-geo-climatic signals that could reflect different features of all five patterns. The decision feature(s) — based on which we could distinguish one pattern from the others — was (were) not the same for all regimes. For instance, we were observing that the minimum temperature attribute alone was able to detect the Nival pattern. While, in catchments with known water ground effects, it was certainly a criterion on discharge that was doing this. In the same spirit of a decision tree algorithm, but at a human level, we concluded that the mean annual discharge, total precipitation, and temperature signals would be the most useful signals to exploit to identify the distinctive attribute(s) in each class. After a number of trial and errors on different properties of these signals, such as the number, magnitude, and time of occurrence of their global/local peaks, we identified our classification attributes — IQ, IP, and $T_{\min}$ — and their thresholds. Therefore, inferring the similarity rule for any other climatic context warrants a redefinition of different elements in this analysis.

Case 2 Another classification (concluded at an AI or a human level) is applied to data belonging to a non French context. In this case, getting different results would not be surprising — since a different rule

involves different decision attributes and criteria. The question in this case would be rather the extent to which we could compare the results obtained from two different classifications.

To conclude on this point, we believe that such cross study comparisons warrant caution, since the imposed similarity rule will not be identical between the studies. That is why we would like to emphasis and acknowledge that research questions established on a subjective component, such as regime classification in our paper, will always make the corresponding conclusions subject to that component.

RC: My overall recommendation is to revise the experiment as suggested in one of the comments below. The experimental design that is appropriate to test the (two) hypotheses outlined here is very simple (but somewhat computationally expensive). If the authors were to find similar results using a more appropriate experiment, this would be an interesting study.

RV: We provide the details of our revision later in §4.2.3 where the reviewer details their suggested design of experiments.

4.2. Comments

4.2.1 Hyperparameter tuning

RC: Hyperparameter tuning was done on LSTMs trained on individual basins. LSTMs trained on individual basins behave fundamentally differently than LSTMs trained on multiple basins, which means that lessons learned from hypertuning on individual basins do not translate to multiple-basin models.

RV: We have totally revised our hyperparameter tuning approach following the methodology suggested by the reviewer. We have defined 54 hyperparameter tunings, where 54 reflects the number of all possible combinations of the three considered hyperparameters — LSTM sequence length with 6 variations, hidden unit number with 3 variations, and dropout rate with 3 variations; that is $6 \times 3 \times 3$. For "each and every one" of the paper's LSTMs — either trained on individual or a group of catchments — we have performed these 54 tunings. Please see §3.2 and §3.4 of the revised manuscript.

RC: Additionally, 15 catchments is not enough for robust hypertuning – we would need to perform hyperparameter tuning on the full (evaluation) dataset (although see a later comment – the experimental design needs to be changed fundamentally). Also, notice that the only portions of the "hypertuning" that were actually used for the other experiments in this paper were (1) discarding the S2 model architecture, and (2) batch size.

RV: Please see the above RV on our new hyperparameter tuning approach.

4.2.2 Number of hidden units in the LSTM layer

RC: There is strong relationship between the dimension of the cell state and the sequence length, and also between the cell state dimension and the ability of the model to generalize (Kratzert et al. (2019) shows how the model uses the cell state to map catchment similarity). This parameter was not included in the hyperparameter tuning, and it was also not considered in the experimental design. 64 cell states is smaller than used by most of the previously published work. The hypotheses that are tested here are about the ability of the model to generalize and about memory timescales, both of which are directly controlled by the cell state (more cell states means more ability to have different memory timescales for different hydrological regimes).

AR: We thank the reviewer for this relevant and constructive comment regarding the number of hidden units in the LSTM layer.

RV: In addition to 64, we have included two larger hidden unit sizes — 128 and 256 — in our hyperparameter tuning.

4.2.3 Design of experiments

RC: It would be interesting (and useful) to know whether there is value in clustering catchments prior to training models, and if so whether we could find correlations between different hyperparameters (e.g., sequence length, cell state dimension) and hydrological regime (the former is a more interesting question than the latter, in my opinion). The way to test this is simple – you separately (and fully) hypertune each model. For example, if you want to test the clustering strategy described in lines 120-125, you would hypertune models separately for each catchment group (considering all of the important LSTM hyperparameters), and as a benchmark you would hypertune a model for all of the catchments combined. Then the results would be directly comparable. After that, you could look at whether there was any relationship between hydrological regime and the "optimal" (hypertuning is never actually optimal) sequence length for that cluster. If you really wanted to train single-basin models (which I suggest you should not do), then these need to be separately (and fully) hypertuned for each basin.

AR: We appreciate the detailed description of the suggested design of experiments (DOE) and we acknowledge that it is thorough. We recognize that the reviewer identifies the following limitations for our old DOE with respect to our research questions:

Limitation 1 We had proposed a hypothesis: the effective size of lookback and regime of catchments are correlated. In order for the hypothesis to be valid, all lookbacks should have been tested for all regimes as well as the entire sample. In the old manuscript, this had been done only in local training and not for regional LSTMs.

Limitation 2 The second hypothesis was that training less but hydrologically homogeneous catchments would be more effective than training more but hydrologically heterogeneous catchments. In our old regional experiments, we had used the lookbacks that were concluded at the local scale. Therefore, we might have not taken the most effective lookback for regional models when answering to the second research question.

Limitation 3 In the reviewer's opinion, the number of hidden units should have been varied along with the lookback size.

RV: We have revised our DOE as follows taking into account the reviewer's suggestions and the three identified limitations:

Revision 1 For all group trained LSTMs, we have tested all and the exact same lookbacks that have been tested for LSTMs trained on individual catchments. This revision has addressed Limitation 1 and 2.

Revision 2 In doing Revision 1, we have tested 3 different hidden units ≥ 64 for all local and regional LSTMs to address Limitation 3.

4.2.4 Interest of local models

RC: I wonder why we are training local models. There is no situation where we would ever use a model trained on a single catchment for any real-world purpose. Additionally, the behavior of the LSTM is fundamentally and qualitatively different when trained on one catchment vs. many, which means that we cannot learn anything general or useful from locally trained models. If there was a specific hypothesis that we wanted to test that required training local models, then this might make sense, but I do not believe this is the case here – we could ask the question about appropriate sequence length on hydrologically grouped models, and asking the question this way would give us a more useful answer. Just a note: Kratzert et al. (in all papers after their 2018 paper) trained single-basin models only to make the point that this is not an appropriate thing to do.

AR: We agree with the reviewer and share their interest in using universal regional models. We also agree with the reviewer on the point that hyperparameters of local models are not optimal for regional LSTMs. That is why in the revised manuscript we have performed a full hyperparameter tuning for all models, as suggested by the reviewer. In the revised manuscript, we have kept the local LSTM to be able to study our first research question at the local scale as well. We have been also interested in benchmarking the LSTM against the GR4J model. We believe that having a local baseline would also help the reader to better interpret and understand different aspects of the regional results that we present, such as the performance gain in the passage from local to regional trainings.

4.3. Minor comments

4.3.1 S2 Architecture

RC: The S2 architecture (stacked LSTMs) is interesting, but not related to either of the hypotheses of the study. What was the motivation for testing this and how does it relate to the questions that were motivated in the introduction? I'm not saying to remove it, just give us some reasoning or motivation. Also, when the "complexity" of this model is discussed, you might give us the number of free parameters so that we can get a sense of what the differences are.

AR: Thank you for this interesting inquiry. The original intent behind studying the S2 architecture in the old manuscript was to act on classical instructions that are given for training any Deep Learning models: prevent underfitting. We wanted to see if vertically stacking LSTMs could immediately bring a better performance thanks to a better hierarchical learning — in the same spirit of operating successive convolutions in a Convolutional Neural Network, if we could "metaphorically" think that what LSTM looks for in time is comparable to what a CNN detects in space. But, we did not observe any instant improvement. There were thus two speculations: either, we still underfit a lot, or, the stacked setup would not basically help. Finding a definite answer to the question of "still underfitting" required an unmanageable amount of work — stacking more LSTMs and increasing hidden units until we overfit and repeating all local and regional experiments at every step — and after all such effort, there was still the risk that "the stacked setup would not basically help". We therefore chose to assume that it was the second hypothesis that held true — stacking LSTMs vertically would not bring significant performance improvement. Thus, we ruled out the S2 structure.

RV: We have excluded this architecture from the revised manuscript.

4.3.2 Purpose of validation set (Line 192)

RC: ***The validation set is intended to be used for finding the best weights and biases during training and control overfitting.*** **I think this is just a typo. Validation data is used to help find the best *hyperparameters* and control overfitting (it is explicitly *not* used to help tune weights and biases, except through early stopping).**

RV: In the revised manuscript, for all local and regional trainings, we have used the validation data to select the best hyperparameter set as well as in the Early Stopping algorithm.

4.3.3 Catchments with very short train data (Line 201)

RC: ***What remains constitutes the train period (P1) the length of which varies between 1 year to 40 years in the FR sample.*** **It is a little concerning to have different sized training data records per catchment, especially if some catchments only have 1 year of training data. This is *especially* problematic if we are looking at differences between what data is required to train in different types of catchments.**

AR: In the old manuscript all catchments with less than 10 years of training data had been excluded from the analyses at a very early stage.

RV: In the revised manuscript, we have excluded all catchments with less than 10 years of training data from the study.

4.3.4 Training duration (Line 180)

RC: In line 180 is reads like you are doing sequence-to-one prediction, however in line 259 you say that you are using a patience of 50 epochs with a maximum of 500 epochs. Typically you only need this many epochs if you are doing sequence-to-sequence training. Regardless, the number of epochs used by previous studies was in the range of 20-50. Have you found that more epochs help (we looked at this carefully in previous studies), or is there something else about your model that is different from previously published work?

AR: Thank you for this interesting inquiry. One reason that we opted for the early stopping algorithm was that it would not impose to all catchments/simulations the same predefined non traversable training duration. Instead, it allows the model to continue to learn as long as its performance (on the validation set) is improving.

The so large numbers that we considered for these two parameters were meant to provide a free boundary for training duration. This would give the model the freedom to learn as long as it needed (unknown to us) without being stopped too early — due to either a too little patience or a too small maximum number of epochs.

This feature was advantageous to us since our data set was new and had not been used in any of the previous studies, notably that it included catchments with very long time series — sometimes up to 40 years for the training data.

4.3.5 Static attributes (Line 291)

RC: This is a pretty small list of catchment attributes. Given that catchment attributes are available globally (e.g., HydroAtlas), and this will directly influence the generalizability of a model, why did we use such a limited set of attributes here?

AR: Thank you for this relevant question. We agree with the reviewer. In the old manuscript, we thought that not considering plenty of static attributes would be concerning if the model was to apply to ungauged catchments, which was not the case in our paper. We therefore took the classical static descriptors often used in previous regionalization studies in the French context (Oudin et al., 2008).

RV: In the revised manuscript, we have included four more attributes — mean daily solid precipitation, mean daily solid precipitation, mean daily potential evapotranspiration, and median altitude — in the static inputs of group trained LSTMs, giving in total 10 static features.

4.3.6 Naming strategy

RC: In general, naming experiments with non-descriptive names like R1, R2. P1, etc. makes the paper more difficult to read than is necessary. This means that the reader must always refer back to the text in order to understand each figure. This can be solved simply by naming each of the models/experiments/datasets with descriptive names.

AR: We fully agree with the reviewer and thank them for this constructive suggestion.

RV: We have replaced all such instances by descriptive names as suggested by the reviewer. We have, for instance, replaced P1, P2, and P3 by training, validation, and test. We have chose SINGLE, REGIONAL, and HYBRID names for our LSTMs.

REFERENCES

F. Chollet et al. Keras, 2015. URL https://github.com/fchollet/keras.

M. Gauch, J. Mai, and J. Lin. The proper care and feeding of camels: How limited training data affects streamflow prediction. *Environmental Modelling Software*, 135:104926, 2021. ISSN 1364-8152. 10.5194/hess-2021-511https://doi.org/10.1016/j.envsoft.2020.104926. URL https://www.sciencedirect.com/science/article/pii/S136481522030983X.

I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. MIT Press, 2016.

S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

F. Kratzert, D. Klotz, C. Brenner, K. Schulz, and M. Herrnegger. Rainfall–runoff modelling using long short-term memory (lstm) networks. *Hydrology and Earth System Sciences*, 22(11):6005–6022, 2018. 10.5194/hess-2021-51110.5194/hess-22-6005-2018. URL https://hess.copernicus.org/articles/22/6005/2018/.

F. Kratzert, D. Klotz, G. Shalev, G. Klambauer, S. Hochreiter, and G. Nearing. Towards learning universal, regional, and local hydrological behaviors via machine learning applied to large-sample datasets. *Hydrology and Earth System Sciences*, 23(12):5089–5110, 2019. 10.5194/hess-2021-51110.5194/hess-23-5089-2019. URL https://hess.copernicus.org/articles/23/5089/2019/.

Y.-A. LeCun, L. Bottou, G.-B. Orr, and K.-R. Müller. Efficient backprop. In *Neural networks: Tricks of the trade*, pages 9–48. Springer, 2012.

T. Lees, M. Buechel, B. Anderson, L. Slater, S. Reece, G. Coxon, and S. J. Dadson. Benchmarking data-driven rainfall–runoff models in great britain: a comparison of long short-term memory (lstm)-based models with four lumped conceptual models. *Hydrology and Earth System Sciences*, 25(10):5517–5534, 2021. 10.5194/hess-2021-51110.5194/hess-25-5517-2021. URL https://hess.copernicus.org/articles/25/5517/2021/.

L. Oudin, V. Andréassian, C. Perrin, C. Michel, and N. Le Moine. Spatial proximity, physical similarity, regression and ungaged catchments: A comparison of regionalization approaches based on 913 french catchments. *Water Resources Research*, 44(3), 2008.