The comments of the reviewer are written black, our answers in purple.

Anonymous Referee #1

The paper describes the use of deep learning rainfall-runoff models based on LongShort Term Memory networks for combining multiple forcing products and improve the model accuracy relative to using only individual input datasets. The approach is demonstrated over 531 basins in the CAMELS dataset. Overall, the approach is technically sound, the manuscript is very well written, and the general topic is interesting for HESS readership. However, there are a few of points that I would recommend to clarify before the paper is accepted for publications.

We want to thank Reviewer #1 for their sincere comments and suggestions. We made two major changes based on this review:

- The first was to add a new set of benchmarks related to how multiple forcings inputs are used in a traditional hydrological modeling situation.
- The second was to shorten the manuscript by moving a lot of the existing analysis to supplementary material and reorganizing the introduction to speak more clearly to the main point of the paper, which is leveraging multiple forcing products to help address challenges in existing methods.

1. The main contribution of the paper should be better contextualised with respect to the existing (and fast growing) literature on the topic. The current manuscript introduction is indeed relatively short (i.e. 30 lines) and only introduces the purpose of this study without illustrating other existing methods. while I found the idea of the proposed approach interesting, neither the use of deep learning hydrologic models or the idea of data fusion is completely new and, therefore, the paper will benefit from a critical analysis of existing methods and how the proposed model is advancing the state of the art. Moreover, I would recommend to better clarify the novel contribution of this paper wrt the sequence of previous publications by the same authors using LSTMs for rainfall-runoff models (I'm not saying this paper is not advancing the previous ones, but considering also the concerns related to the benchmarking discussed at point 2 I believe the authors should clearly demonstrate that the contribution of this paper is beyond the "minimum publication unit").

We agree with this assessment. The introduction in our original submission was short and missing a clear statement about why this manuscript is clearly advancing over previous publications. In the revised manuscript we include new introductory material that outlines challenges related to leveraging multiple inputs in traditional hydrology models as well as related literature. We expect that this, along with the added benchmarks related to these traditional methods (see answer to remark 2), will help clarify the new contribution presented in this manuscript.

2. The set up of the benchmarking analysis is not fully convincing as the authors are comparing their model accuracy against (A) models calibrated using a single product and (B) traditional hydrologic models from Kratzert et al. (2019b). While the first analysis is the core of the paper, I don't understand the reason for the second one for two main reasons: in Kratzert et al. (2019b) the authors have already demonstrated the superiority of LSTMs wrt standard hydrologic model; if the new models that combines multiple inputs outperform the LSTMs using a single forcing as shown in (A), it comes straight that the new models also perform better than standard hydrologic models. In addition, this second benchmarking might confuse some readers who may attribute the reported improvements to the combination of inputs, whereas they are mostly due to the model structure. Rather than the comparison with traditional hydrologic models (which cannot use multiple meteo forcing data as the LSTMs), I would suggest the paper will benefit much more from a benchmarking against other state-of-the-art data driven models.

We do understand why one would come to these conclusions. Nevertheless, we believe that the model comparison in Figure 4 is important, since it contextualizes and highlights the improvement we see due to using multiple inputs in a single LSTM. It gives a sense of how much this improvement really is (the multi-forcing LSTM almost - not quite - doubles the performance gap between LSTM-based models and traditional hydrological models).

We added this analysis to contextualize the results of our current manuscript to our previous studies, where we trained LSTMs just on a single forcing product. The purpose of the hydrological benchmark models is to highlight the improvement of the model performance over single-forcing LSTMs .

However, we agree with the reviewer that including a different set of benchmarks improves the manuscript. In the revised manuscript (uploaded on invitation by the editor) we benchmarked against arguably the most common method of using multiple forcing products in the context of traditional hydrological models, which is to train separate hydrological models for each forcing product, and to combine their outputs using ensembling techniques. We used the SAC-SMA + Snow-17 model, which is used for operational forecasting in the US and was also the model originally included in the CAMELS data set. To account for stochasticity in the optimization process, we calibrated multiple models per basin and forcing (similar to what was done in the original CAMELS paper by Addor et al.). The code and simulation outputs will be made available.

Regarding adding different state-of-the-art data driven models: We are not aware of any other data-driven modeling approach (something that is not based on LSTMs) that yields similar

performance for regional/continental modeling tasks (i.e. one model that predicts discharge everywhere) and can thus also be applied to forecasting (e.g., PUB,as was shown for the LSTMs in one of our previous publications).

3. Lastly, the paper is in my opinion a bit lengthy with 14 figures that make the narrative a bit scattered. I would then suggest to explore the option of selecting the main findings-figures worth to be discussed in the main paper (e.g. Fig. 6 and 7) and move some content to a supplementary material.

Thanks for the suggestion. We agree with this assessment and thus moved a lot of material from the original manuscript to supplementary sections.

The comments of the reviewer are written black, our answers in purple.

Anonymous Referee #2

The main point of the paper was that one can use multiple precipitation products in a single LSTM to improve streamflow model performance. The other analyses are secondary (and also problematic), without memorable take-home messages. One of the two conclusions in the abstract is "A sensitivity analysis showed that the LSTM learned to utilize different precipitation products in different ways in different basins and for simulating different parts of the hydrograph in individual basins", which does not seem to have said much. I think there is some value in this idea (although incremental) of fusing multiple forcing dataset using DL, but the effectiveness of DL has not been compared to other methods and is thus out of context.

Thank you for acknowledging the value of our work. The ability to learn nonlinear, nonstationary and spatially adaptive mixing strategies is a ‘holy grail’ of ensemble modeling - we’re not sure how one could characterize this particular finding as having “not said much”.

We do fully accept and agree with the first reviewer’s comment that this was not emphasized well enough against a backdrop of current approaches for using ensembles of inputs, and this has been rectified in the revised manuscript that we will upload conditional on invitation by the editor.

We changed the manuscript in three ways to emphasize this point more clearly: (1) Changing the introduction to highlight challenges and current strategies for using ensemble forcings, (2) adding more relevant benchmarks, and (3) moving a substantial portion of the hydrological analysis that is only tangentially related to this main point into supplementary material.

Also, stacking multiple data sources as inputs is a common practice among machine learning practitioners. The procedure itself is not the novelty, although I get it that in hydrology few might have used it. This might be a matter of personal opinion, so I ask the editor to weigh my opinion as what it is — an opinion, the paper appears too thin and too incremental for me to warrant a HESS contribution. This is based on my understanding of HESS as a premier outlet for hydrological science. When the authors published their first couple of papers, which they cited, it was novel. Now, LSTM seems to be widely used in hydrology and it is no longer novel, so the sole point become the use of multiple forcing datasets. My personal judgement is that this point alone lacks the punch needed for a HESS paper.

It seems that we disagree on the interpretation of the results on a fundamental level. We thus dedicate the next paragraphs to illustrate our point of view.

We do not make any claims about presenting a novel algorithm or any type of novel ML theory development (as a site note, we also never claimed that the LSTM is something novel in previous publications, in-fact it is almost 30 years old). To our knowledge we are the first to test the idea of using multiple forcing products as inputs in the context or large-scale, data-driven rainfall-runoff modeling, and the result of this very simple strategy was a relatively large improvement to simulation accuracy. As reviewer #1 implied, one can view this as an implicit form of input-fusing where the model learns the (nonlinear, heterogeneous) input combination/transformation itself.

These results are (1) novel in the context of hydrology, (2) currently the best large-sample daily streamflow simulation results ever published that we know of (that do not require data assimilation or auto-regression and could therefore be used in ungauged basins), and maybe more importantly (3) present a simple but effective solution for one of the classically 'hard' problems in hydrological modelling: how to combine information from multiple inputs in spatiotemporally heterogeneous ways. The fact that this is a simple (but effective) approach is a *strength*, not a *weakness*.

Beyond the main opinion, I raise some other major points below. It does occur to me many of the claims were rather casually made in this paper and need further validation. Many details were missing, and I would be worried some results are not stable.

1. The motivation seems problematic in logic. What is such a model used for? (i) Are you using it for climate change impact assessment? You are not going to have three forcing datasets which you can train the model with. None of the datasets will be available for future climate. Climate model outputs are not able to be used in supervised training like this, with daily streamflow as the target. (ii) Are you using it for flood forecasting? It does not seem like this model is optimally wired for forecasting, which in general taps into data assimilation. It is uncertain is significant value of multiple forcing would still exist in a setting with data assimilation. (iii) Are you using it for hydrologic budget analysis? With this setting, we don't even know how much rainfall has been applied in the model from a mass balance point of view. Hence, while the results may look nice, it may not have real-world use cases!

Daily streamflow simulation is one of the most common and impactful tasks in operational (surface) hydrology. This model is absolutely 'wired' for forecasting - we (Upstream Tech; company where the last author works) currently use a proprietary version of this model to produce operational forecasts at both the short term (10-day-out) and seasonal (multi-month) timescales using ensembles of weather forcing products. The setup tested here in hindcasting mode is a direct analogy of that operational model, but without proprietary products (like ECMWF weather forecasts, etc.). Our (last author's) company (Upstream Tech) currently sells the simulations made by a proprietary version of these models to public and private customers in environmental, hazard, and hydropower sectors.

These models work with several different forms of data assimilation (many of which we are running in the operational version of the models reported here), however data assimilation is not used in the majority of forecasting situations at the national or global scale (with any type of model) because data assimilation requires local streamflow observations, and the majority of forecast points in the US and in the world are ungauged. As an example, the US National Water Model has 2.X million forecast points and only ~18,000 of those are at gauge locations. Data assimilation is important, and is significantly easier with deep learning than with traditional hydrology models (although that is not the topic of this paper), but a forecasting model must work in ungauged locations as well.

2. I wish not to see hydrology becoming a computer-science competition where an incremental change in an experiment becomes a new paper. There should be either scientific advances or methodological innovations. The results might be publishable but as it reads now it does not look to be at HESS level.

There is no danger that hydrology will reduce to a computer science competition; and yet, this type of competition will definitely become a critical *part* of the future of hydrology as a discipline. Right now, the choice of model among hydrologists is guided more by lineage and affiliation than empirical evidence (Addor and Melsen, 2018). There is some argument to be made that once one masters a specific model to a high enough level a preference should be attached to it because it allows to solve problems faster and explore tasks in a deeper fashion (since it will not be necessary to learn everything from scratch). However, as of now, no universal principles exist that allow us to establish specific models for the given tasks and goals in hydrology. This means that we (hydrologists) are not doing the best job we could be doing to be an evidence-guided discipline. It is important that this kind of model competitions become *one aspect* of hydrological science. This does not mean that the discipline as a whole will be reduced to it - e.g., process hydrology will always remain an important part of what we do.

Simply put, hydrology is an applied science, and while the process-understanding component of the science is important, its purpose is ultimately to support societal applications (most of the time we are not exploring deep philosophical questions about the nature of the universe in hydrology). That said, empirical competition between models is a critical backbone of any discipline that cares about objectivity (see also the arguments made in Donoho, 2017). It is inevitable and important that part of the future of hydrological science will be machine learning, and to the extent that a journal chooses to ignore or de-emphasize this, that journal will position itself to not be a part of one of the most important emerging sub-branches of hydrology.

3. The authors claimed DL is better than traditional hydrologic models at using multiple sources of information. This is not proven. One can run an ensemble of simulations with multiple forcings, as authors said around line 270. Now, I do expect DL models to outperform, because previous papers have shown that. but perhaps the difference

between single-forcing and multiple-forcing will be similar to what is shown here. At least no hard evidence was provided.

Yes, we agree that this needs to be shown. This type of analysis and benchmark was added to the revision (see also our first answer to referee #1).

4. Does the author use only multiple precipitation data or all the forcing variables? If only multiple precipitation were used, the author should clearly state this and use “multiple precipitation” instead of “multiple meteorological/forcing” in the title and main texts. If multiple datasets were used for all forcing variables, why were the analyses only executed on the precipitation in section 3.2 and 3.3? How did you know the effects were not due to other variables? If only precipitation was used, why not use other variables?

In Line 62ff we state that we use all three forcing products as inputs, using all their 5 meteorological variables. We even say that two products (Maurer and NLDAS) do not include daily minimum and maximum temperature in the original CAMELS data set and therefore provide these variables with this publication (see L66 and data availability section). However, to make it as clear as possible, we included the following sentence in the revised manuscript:

“We used all five meteorological variables of all three data products as inputs for our model”.

We only looked at the influence of the three precipitation products because a) precipitation is arguably the most important variable in the process of rainfall-runoff modeling b) according to Behnke et al. (2016) there is little difference in all other meteorological variables in between these data products, c) we know from other research projects that precipitation has by far the most influence in LSTM-based rainfall-runoff models (see e.g. Frame et al., 2020), and d) nothing that we show or conclude implies that other variables are not important, but the point is to show that the LSTM learns to mix forcings in dynamically heterogeneous ways. We show this using the precipitation input.

It is trivial to perform similar analysis on all other variables (and we invite everyone to do this with the code we provide with our paper), but we believe that nothing would be gained from it and the paper would become even more bloated.

5. The benchmark scenarios with the hydrologic models are off topic. The comparison between the LSTM and these hydrologic models have been done in Kratzert et. al, 2019. Therefore, it’s quite obvious that the multi-forcing model in this study would further outperform hydrologic models. Given the main topic here is to show the effective synergy of multiple meteorological forcings using deep learning model, the fair benchmark would be the ensemble forcing simulations with hydrologic models.

Referee #1 had a similar impression (see Referee #1, comment 2). We do however believe that these benchmarks convey important information. The purpose of including them was to

contextualize the increase in performance between the LSTMs from Kratzert et al. (2019) and the models of this manuscript, compared to the performance of traditional hydrology models. This is shown in Figure 4, which directly compares (1) the spread between traditional models, (2) the gap between traditional models and LSTMs, and (3) the extra improvement due to multiple forcings. It puts the value of this approach into context of the previous step-change from deep learning.

We also added several multi-input ensemble benchmarks to the revision (which we will upload upon invitation by the editor), which - the reviewer is correct - were missing from the original manuscript (see also answer to reviewer #1 and to comment #3 above).

6. The explanation of the triple collocation method was not clearly provided in section 2.4.2. The meanings of α, β, e were not explained in equation 1, 2 and 3. How was Equation 4 derived from the equation 2 and 3? I was confused here. Did the author first fit a log-transformed linear model (equation 2 and 3) or directly use the covariances of time series to calculate the error variance as shown by equation 4?

If possible we would like to keep the description as concise as it currently is. It is generally not customary to re-derive each established method used in a paper, especially if this method is around 20 years old and has been applied many times in the field of hydrology. It is customary to show the main equations so that a reader can develop an intuition about what the analysis means and how it works. Readers interested in deriving the method can refer to the first-order references in the paper, which are/were referenced throughout Section 2.4.2.

7. The analysis of gradients is very unclear. In section 2.4.3 and 3.3, why did the author choose the “integrated gradients” instead of the simple gradients of inputs extracted from the network? At the time of simulation, only the actual variable value matters, not the whole range of x. And... how did you integrate it for different values in x? It wasn’t clear at all. If the author chose a more complex implementation, the comparison with the original one is needed. I believe most readers will have hard time understanding what the “gradients” really means as shown in section 3.3. For example, precipitation at previous hundreds of days (i.e. the length of training instance) can all contribute to the runoff prediction at the present time step T because of the memory characteristic of the LSTM model. Did the gradient at time step T refer to the gradient of the precipitation at the present time step or the sum/average gradients of all the previous hundreds of days? More clarifications should be given to help readers understand the gradient results here.

We believe that this is enough information to understand and interpret the result. Readers, interested in the method itself, do find the appropriate references throughout the manuscript.

Regarding why we choose integrated gradients instead of local gradients: Lines 159-164 of the original text explain it this way:

“All neural networks (like LSTMs) are differentiable almost everywhere by design. Therefore, a gradient-based contribution analysis seems natural. However, as discussed by Sundararajan et al. (2017), the naive solution of using local gradients is not a reliable measures of sensitivity, since gradients might be flat even if the model response is heavily influenced by a particular input data source (which is not by necessity a bad property, see for example Hochreiter and Schmidhuber, 1997a). This is especially true in neural networks, where activation functions often include step-changes over portions of the input space - e.g., the sigmoid and hyperbolic tangent activation functions used by LSTMs have close-to-zero gradients at both extremes (see also: Shrikumar et al., 2016; Sundararajan et al., 2017)”

What we tried to convey in this passage is that scholars and practitioners are aware that for this kind of analysis one should not use local gradients but e.g., something like the integrated gradient method. We believe that this is quite clearly stated in the above paragraph and linked to the related references.

Furthermore, there seems to be some misunderstandings on the applied method: The precipitation of the entire input sequence does influence the prediction. The integrated gradient signal we report for one time step t is not the gradient signal of only the inputs of time step t but the average integrated gradient signal over the entire input sequence (i.e. the previous 365 days), which is stated in L. 171:

“We calculated the integrated gradients of each daily streamflow estimate in each CAMELS basin during the 10-year test period with respect to precipitation inputs from the past 365 days (the look-back period of the LSTM). That is, on day $t = T$, we calculated $1095 = 3 * 365$ integrated gradient values related to the three precipitation products. The relative integrated gradient values quantify how the LSTM combines precipitation products over time, over space, and also as a function of lag or lead-time into the current streamflow prediction”

Additionally, in L. 233 in the results section, we explicitly re-iterate:

“To reiterate from above, the integrated gradient is a measure of input attribution, or sensitivity such that inputs with higher integrated gradients have a larger influence on model outputs. Integrated gradients shown in Fig. 11 were averaged over all timesteps in the test period, and also over all basins. This figure shows the sensitivity of streamflow at time $t = T$ to each of the three precipitation inputs at times $t = T − s$ where s is the lag value on the x-axis.”

We also don’t think that for the general reader, all details behind this method are important (similar to the triple collocation), as long as one understands what the output of this method is. This is explicitly stated in L. 233:

“To reiterate from above, the integrated gradient is a measure of input attribution, or sensitivity such that inputs with higher integrated gradients have a larger influence on model outputs”.

8. Following the above comments, the gradients of inputs w.r.t. the outputs can be quite unstable for deep learning models. Were the results shown in section 3.3 based on the

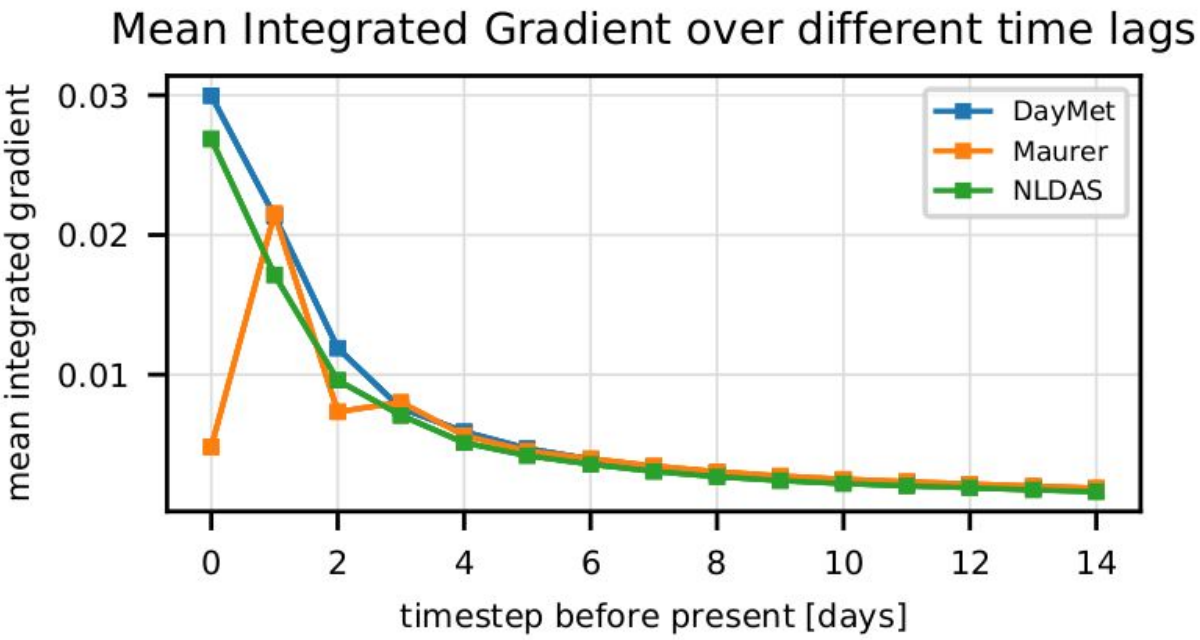
gradients of ensemble runs or single model? If single model was used here, the author should show the results of multiple ensemble members as well as standard deviations and demonstrate the robustness of their gradients.

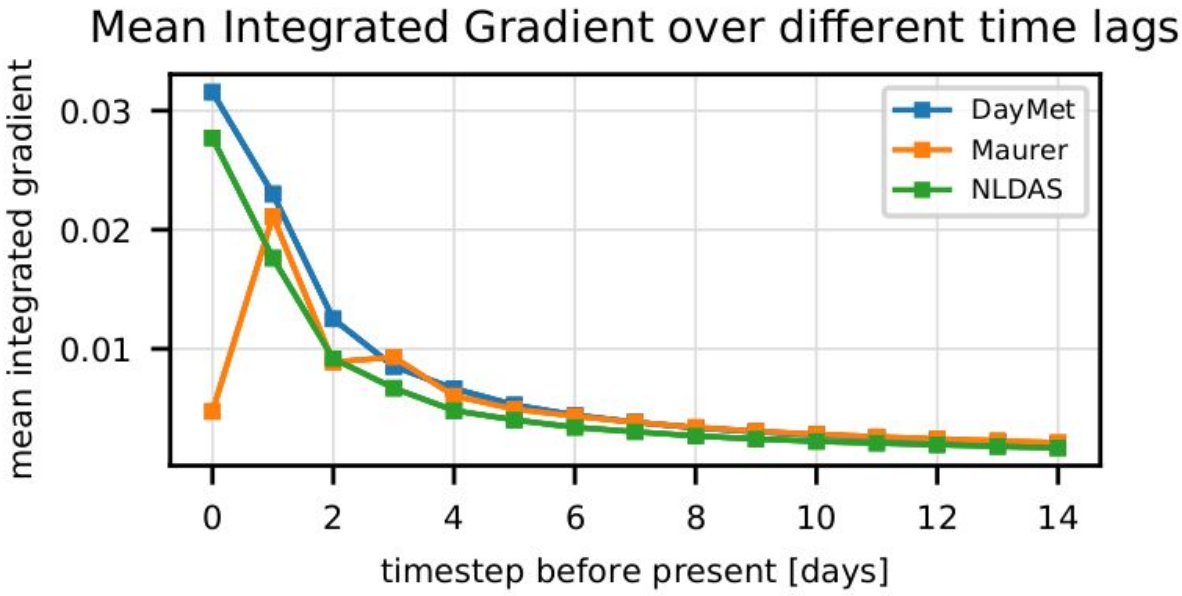
We had a lengthy discussion about this question, because it was not quite obvious to us what the reviewer meant. Generally, gradients are a perfectly robust concept (with gradients referring to the derivative of the loss/output w.r.t the network weights/inputs).

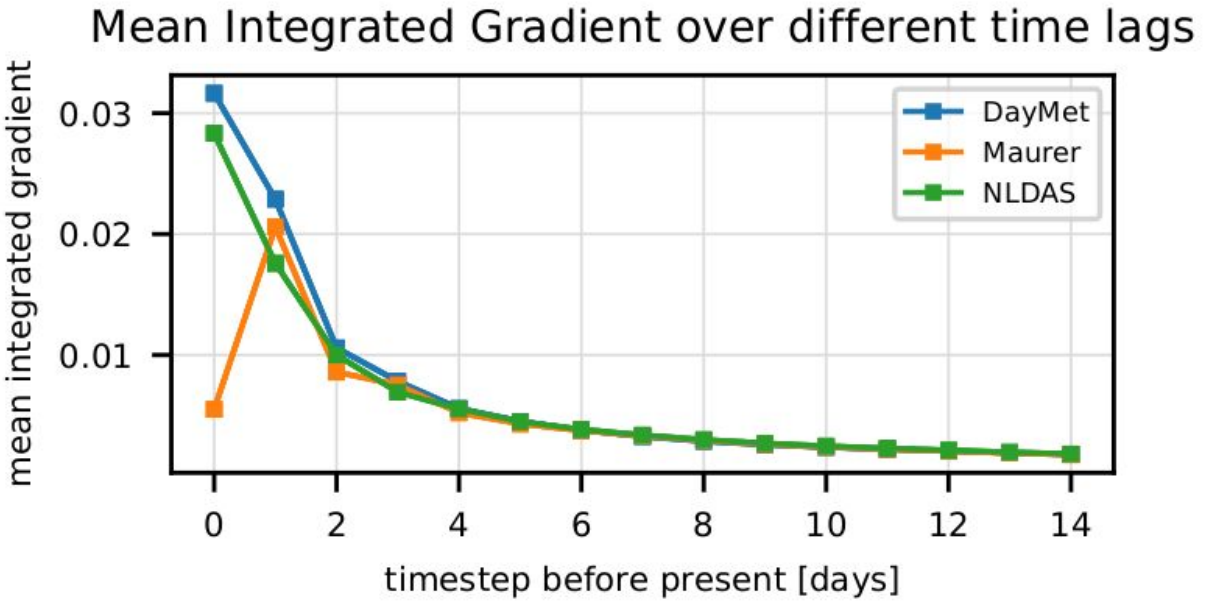
We think what the reviewer means is that deep learning models suffer a certain randomness due to different initialisations of the network parameters or stochasticity in the learning process (i.e. mini-batch sampling). Thus the question if the results are from a single model or not, because differently initialized models could learn different things. And, since we showed results from a single model (out of the 10 repetitions we trained, as reviewer #2 correctly points out), the learned behavior of the model could indeed be different (which, we believe, the statement of the reviewer hints at).

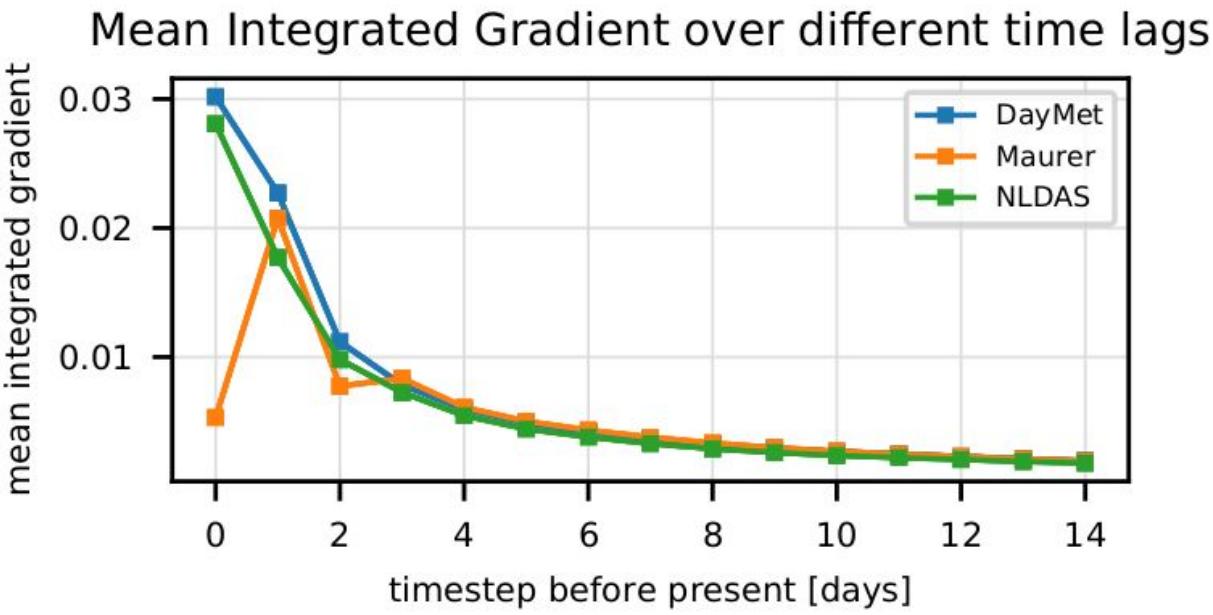
The reason for showing only a single model is that the variation between different models is negligible. That is, qualitatively all models make use of the three forcing products in a similar way. We will include a sentence to clarify this in the revised manuscript.
To validate this point, below are analogies of Figure 11 of the manuscript derived from 5 different models.

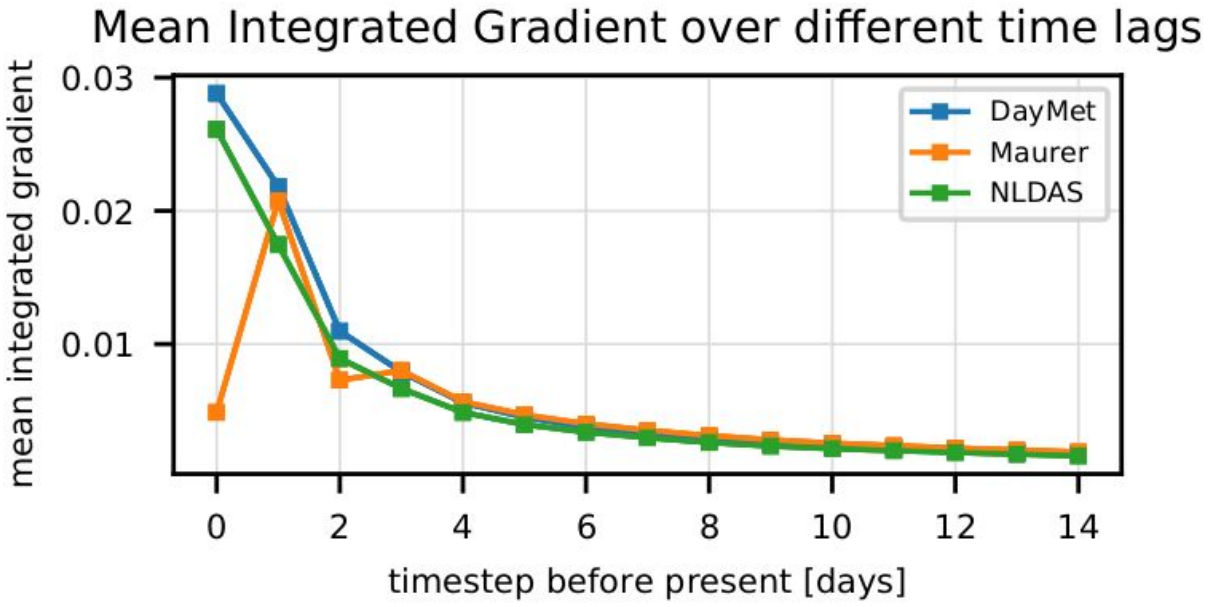
The reason for not including error bars or reporting standard deviations is, because the absolute value of the integrated gradient method is not of real importance (which makes it harder to quantitatively compare integrated gradient results of two models). It is rather a relative measure, showing which parts in the inputs are more (or less) important for the model prediction. And as the reviewer and editor can see for the figures below, the relative behavior between all models is practically identical (i.e. Maurer forcings are practically ignored at the last time step and only gain importance afterwards, which is the message we want to tell with this figure).


Mean Integrated Gradient over different time lags
mean integrated gradient
0.03
0.02
0.01
0
2
4
6
8
10
12
14
timestep before present [days]
DayMet
Maurer
NLDAS


Mean Integrated Gradient over different time lags
mean integrated gradient
0.03
0.02
0.01
0
2
4
6
8
10
12
14
timestep before present [days]
DayMet
Maurer
NLDAS


Mean Integrated Gradient over different time lags
mean integrated gradient
0.03
0.02
0.01
0
2
4
6
8
10
12
14
timestep before present [days]
DayMet
Maurer
NLDAS


Mean Integrated Gradient over different time lags
mean integrated gradient
0.03
0.02
0.01
0
2
4
6
8
10
12
14
timestep before present [days]
DayMet
Maurer
NLDAS



9. The analysis in section 3.2 is not convincing:
    a. In line 210-212, the author clearly stated that the model performed worse in basins with lower precipitation error, especially for NLDAS forcing. This is counter-intuitive, leading me to question the validity of the triple collocation method employed and what it really says. How can the author verify the validity of this method given this abnormal result? Although they have listed the locations of those abnormal basins, they did not give convincing and detailed explanations for this problem.

We believe that these “non-intuitive” results are explained quite thoroughly in the manuscript already. Several paragraphs immediately following the one that the reviewer references are devoted to it.

There are several factors that act together to cause this phenomenon, but the main one is that NLDAS has some anomalies in particular basins in the Rocky Mountains that dominate this effect (illustrated in Figures 7 and 8). Additionally, *“Triple collocation measures (dis)agreement between measurement sources, rather than error variances directly”* (Line 216), and it is not always the case that one forcing product disagreeing with others is actually error. Figure 9 then shows how this agreement/disagreement is correlated with total precip in one, but not the other two, of the products. This points to a systematic difference between daymet and the other two products, that is picked up by triple collocation. The point we will eventually make based on this analysis is that the LSTM can exploit this systematic difference.

As a point in this review process, we would like to point out that we dedicated two paragraphs and three figures (Figure 7, 8, and 9) to explaining these non-intuitive results that the reviewer has mentioned. It's a little difficult to know what more we could do.

Just to reiterate, the larger point that this is ultimately supporting is that the information in multiple forcing products is complex - with both redundant and synergistic characteristics, - and the LSTM learns how to leverage at least a lot of this complex information aggregation in ways that are difficult to predict a priori from detectable features of the inputs themselves. We would have preferred to do this analysis using information theory, which would have let us use more concrete terms and more direct quantification of redundant and synergistic information, but there simply is not enough data on a per-catchment basis for stable information theory results, so we had to use log-linear analogues (i.e., Triple Collocation).

b. What are the differences of "σ" and "total variance" in Figure 7 and 8? More explanations are needed here to avoid confusion.

This difference between total variance of a precipitation product and triple collocation error variance (sigma) is explained in Line 219 of the original manuscript, which the reviewer quotes in their next comment.

c. Line 218-219, "with higher total precipitation variance (not triple collocation error variance), indicating better performance in wetter catchments." This expression is not rigorous since the arid areas with frequent extreme events can also have large precipitation variance, such as Texas.

An early version of the manuscript had plots that showed the correlation between precipitation variance and total precipitation, but we removed these figures because some early feedback we got from an informal reviewer was that this was common knowledge in hydrology (precip variance is highly correlated with total precip). If the editor or reviewer thinks that we should include this figure again, we can do it, however our impression was that this is an additional figure (since, as both reviewers mentioned, the number of figures is already too high).

d. Line 220- 221, "This is not true for the other two models, where higher total variance is associated with a higher variance in model skill, indicating higher proportion of the variance due to measurement error". This is very hard to understand and needs further supporting evidence. I can only see that the variance spread is large for basins with high NSE performance for Maurer and NLDAS from Figure 9.

Since the reviewer seems to misunderstand the argument it could be that we compressed the message too much. Generally spoken, in any data product there are two basic sources of variance: variance of the true value and variance of the error. All we are saying here is that if increasing model skill is associated with increasing precip variance, then that increasing precip

variance is likely not due to measurement error. If increasing precip variance is not correlated with increasing model skill, then some of the precip variance is likely due to measurement error.

The point of this figure and analysis is to show that there is a systematic disagreement between the per-catchment error patterns in DayMet vs. the other two products, which explains the non-intuitive triple collocation results in the other two products where TC error (which is just a measure of agreement) is apparently not correlated with NSE improvements. This shows that while TC can 'see' systematic differences in the precip products, in this case it is likely that DayMet is better than the other two, and also carries more unique information than the other two.

We are trying to show uniqueness in the data sets without (unfortunately) actually being able to measure information directly (in the sense of mutual information). In the revised manuscript will add this explanation to the sentence that the reviewer quoted to make the message clearer.

e. The same problem as the point 1) in line 228 and Figure 10, the author can not draw a general conclusion and neglect those abnormal basins in Figure 7 and these basins behave differently than the conclusion, so it might be your conclusion that is wrong!

As we mentioned above, it seems like the reviewer seems to have missed or misunderstood our explanation of these "non-intuitive" results from Figure 7 (which is the bulk of the text in this subsection and is critical to understanding these results). As stated above, we will try to make this section even clearer in the revised manuscript to avoid further confusion.

10. Other comments:
    a. Why were only 447 of the 531 basins used for the benchmark with hydrologic models (section 2.4.1)? I went back to their 2019 paper and they used 571 there, where they benchmarked against other models. Furthermore, as we discussed earlier, there is no longer a point to benchmark against default traditional models. It has been done. I would welcome a benchmark with the ensemble forcing scenario, but that was not included.

In our 2019 Benchmarking paper, which we think the reviewer is referencing, ("Towards learning universal, regional, and local hydrological behaviors via machine learning applied to large-sample datasets"), we used the same basins as in this current paper. We used 531 to train our model and 447 to benchmark (same as here). As we explained in our earlier paper (e.g. first sentence of Section 3.2), and in this paper (Line 40), the reason is that not all of the hydrological benchmark models are available for all basins. Simulations from all benchmark models are available at only 447 basins. Again, we did not run our own benchmarks - this was a community contribution effort, which was done to avoid bias in the implementation of the benchmark models.

b. The paragraph starting at line 43 cited the statements in Behnke et al. (2016) and seemed quite incoherent here. It seems this paragraph should be better moved to the introduction part.

If the reviewer/editor does not insist we would like to keep the paragraph where it is. It would be unusual to move such a paragraph to the introduction. This is a specific statement about the specific data products we are using. It is not a general statement about hydrologic data, and does not serve as motivation or background for our project - it is a very specific characterization from a previous study of the specific data products used here, and thus belongs in the section describing the data.

c. Figure 7 and 6 can be combined since they tell similar stories. There have been so many figures in the paper which made the paper look redundant. Really with the actual content available in this paper 4 figures would have been adequate

We believe that it is better to keep them separated. Figure 7 and 6 emphasize different aspects of the story (cause for lack of TC/NSE correlation vs. cause (elevation) of NLDAS anomaly). 'Looking redundant' is not a concern for us and these figures would take up the same amount of space and contain exactly the same information if referenced by the same figure number as they do separately.

References

Addor, N., & Melsen, L. A. (2019). Legacy, rather than adequacy, drives the selection of hydrological models. *Water Resources Research*, 55, 378– 390. https://doi.org/10.1029/2018WR022958

David Donoho (2017) 50 Years of Data Science, Journal of Computational and Graphical Statistics, 26:4, 745-766, DOI: 10.1080/10618600.2017.1384734

Frame, J., Nearing, G., Kratzert, F., & Rahman, M. (2020, July 2). Post processing the U.S. National Water Model with a Long Short-Term Memory network. https://doi.org/10.31223/osf.io/4xhac

A note on leveraging synergy in multiple meteorological datasets with deep learning for rainfall-runoff modeling

Frederik Kratzert[1], Daniel Klotz[1], Sepp Hochreiter[1], and Grey S. Nearing[2]

[1]LIT AI Lab & Institute for Machine Learning, Johannes Kepler University Linz, Austria
[2]Department of Geological Sciences, University of Alabama, Tuscaloosa, AL United States

Correspondence: Frederik Kratzert (kratzert@ml.jku.at), Grey S. Nearing (gsnearing@ua.edu)

Abstract. A deep learning rainfall-runoff model can take multiple meteorological forcing products as inputs and learn to combine them in spatially and temporally dynamic ways. This is demonstrated using Long Short Term Memory networks (LSTMs) trained over basins in the continental US using the CAMELS data set. Using multiple precipitation products (NLDAS, Maurer, DayMet) in a single LSTM significantly improved simulation accuracy relative to using only individual precipitation products. A sensitivity analysis showed that the LSTM ~~learned to utilize different~~ combines precipitation products in different ways ~~in different basins and~~ depending on location and also in different ways for simulating different parts of the hydrograph~~in individual basins~~.

1 Introduction

~~There are many different meteorological products that a hydrologist might choose as forcing data , and no data product is perfect~~All meteorological forcing data available for hydrological modeling are subject to errors and uncertainty. While temperature estimates between different ~~forcing~~ data products are frequently similar, precipitation estimates are often subject to large disagreements (e.g., Behnke et al., 2016; Timmermans et al., 2019). The ~~appropriate choice of the input forcing data is an important step for every modelling task. To our knowledge it is so far not possible to dissect, which methodological choices lead to which disagreements in the data products (e.g., Beck et al., 2017; Newman et al., 2019); nor is it straightforward to estimate how these differences translate to model behavior (e.g., Yilmaz et al., 2005; Henn et al., 2018; Parkes et al., 2019). Thus, the choice of the *"right"* product, for a given modelling exercise, requires careful consideration.~~

~~Generally speaking, the~~ most accurate precipitation data ~~comes~~ generally come from in situ gauges, which ~~are~~ provide point-based measurements of rainfall events, which are complex spatial processes (although in certain cases, especially related to snow, modeled products might be better - e.g., Lundquist et al., 2019). ~~Today's~~ However, large-scale hydrological models ~~, however, require data fields~~ require spatial data (usually gridded), which are necessarily model-based products ~~,~~ resulting from a combination of spatial interpolation, and/or satellite retrieval algorithms~~. Each~~ , and sometimes process-based modeling. Every precipitation data product is based on different sets of assumptions that each potentially introduce different types of error and information loss. It is difficult to predict a priori how methodological choices in precipitation modeling or interpolation algorithms might lead to different types of disagreements in the resulting data products (e.g., Beck et al., 2017; Newman et al., 2019)

. As an example of the consequences of this difficulty, Behnke et al. (2016) showed that no existing gridded meteorological product is uniformly better than all others over the continental United States (CONUS).

In this context, we would like to point out that - depending of the goal of the modelling exercise - data-driven modelscan have an inherent advantage compared to traditional hydrological modeling techniques: *A single data-driven model can use multiple forcing products directly*. The models can The primary strategy for dealing with forcing uncertainty in hydrological modeling is to use ensembles of forcing products (e.g., Clark et al., 2016). These can be ensembles of opportunity or they can be drawn from probability distributions, and they can be combined either before (e.g., as precipitation) or after (e.g., as streamflow) being used in one or more hydrological models. In any case, it is generally not straightforward to predict how differences between different forcing products will translate into differences between hydrological model simulations (e.g., Yilmaz et al., 2005; Henn et al., 2018; Parkes et al., 2019), and given that data quality among different products varies over space and time, it is difficult to design ensembling strategies that maximize the information or value of forcing ensembles.

However, unlike conceptual or process-based hydrological models, machine learning (ML) or deep learning (DL) can use multiple precipitation (and other meteorological) data products simultaneously. This means that it is not necessary to design a priori strategies for combining input forcing data or for combining the outputs of hydrological models forced with different data products. In principle, such models could learn to exploit potential nonlinear synergies in different (imperfect) precipitation data sets(, or any other type of model input). In particular. Particularly, deep learning models as that are able to learn spatiotemporally heterogeneous behaviors, such as those used by Kratzert et al. (2019b, a) can take any number of different precipitation and other meteorological inputs at every timestep. Because the different input data sets are used simultaneously in a single nonparametric model, this has the potential to produce more accurate simulations by combining those inputs in spatiotemporally dynamic ways. The goal of this contribution is to test the strength of this hypothesis by assessing the modelshould be able to learn spatiotemporally dynamic 'effective mixing' s ability to learn complex and spatiotemporally variable interactions between different precipitation productsstrategies in the way that they leverage multiple input products in different locations and under different hydrological conditions. If successful, this could provide a simple and computationally efficient alternative to ensembling strategies currently used for hydrological modeling.

2 Methods

2.1 Data

This study uses the Catchment Attributes and Meteorological dataset for Large Sample Studies (CAMELS; Newman et al., 2014; Addor et al., 2017b). CAMELS contains basin-averaged daily meteorological forcing inputs derived from three different gridded data products for 671 basins across CONUS. The three forcing products are (i) DayMet (Thornton et al., 1997), (ii) Maurer (Maurer et al., 2002), and (iii) NLDAS (Xia et al., 2012), the former has 1 km x 1 km spatial resolution and the latter two have one-eighth degree spatial resolution. Although CAMELS includes 671 basins, to facilitate a direct comparison of results with previous studies we used only the subset of 531 basins that were originally chosen for model benchmarking by Newman

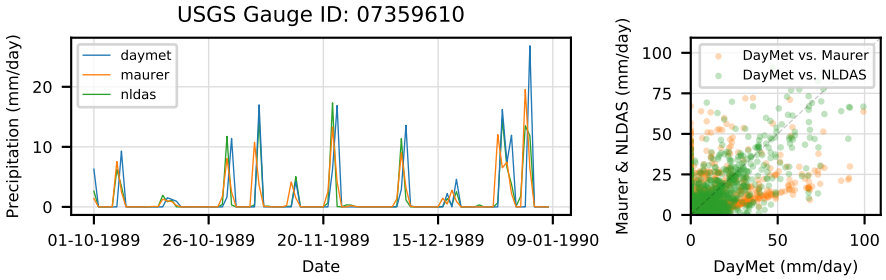


Figure 1. Illustration of the relationship between three CAMELS precipitation products at a randomly-selected basin (USGS ID: 07359610). The left-hand subplots show the first 100 days of precipitation data from all three products during the test period, and the right-hand subplot shows scatter between the three products over the full test period. The scatter shown in the right-hand subplot is the data uncertainty that we would like to mitigate~~by using multiple forcings simultaneously in a deep learning rainfall-runoff model~~. In this particular basin, there appears to be a 1-day shift between DayMet and Maurer, which is common in the CAMELS data set (this shift is apparent in 325 of the 531 basins; see Figure 2)

et al. (2017), who removed all basins with area greater than 2000 km^2, and also all basins where there was a discrepancy of more than 10% between different methods of calculating basin area. ~~These 531 basins were used for all experiments in this study except benchmarking against traditional hydrology models (see Sect. 2.5.1), because the benchmark models are only available at 447 of the 531 basins.~~

Behnke et al. (2016) conducted a detailed analysis of eight different precipitation and surface temperature (daily max/min) data products, including the three used by CAMELS. Those authors compared gridded precipitation and temperature values to station data using roughly 4000 weather stations across CONUS. Their findings were that *"no data set was 'best' everywhere and for all variables we analyzed"* and *"two products stood out in their overall tendency to be closest to (Maurer) and farthest from (NLDAS2) observed measurements."* Furthermore, they did not find a *"clear relationship between the resolution of gridded products and their agreement with observations, either for average conditions ... or extremes" and noted that the "high-resolution DayMet ... data sets had the largest nationwide mean biases in precipitation."*

Figure 1 gives an example of disagreement between precipitation products in CAMELS that we hope to capitalize on by training a model with multiple forcing inputs. This figure shows the noisy relationship between the three precipitation products in a randomly-selected basin (USGS ID: 07359610). ~~Deep Learning approaches can learn~~ The idea is that DL should be able to mitigate the type of noise shown in the scatter plot in the right-hand panel of Fig. 1~~and use the inherent information by using multiple forcing products simultaneously in a single model~~.

The left-hand subplot of Fig. 1 shows a time-shift between DayMet and Maurer precipitation in the same basin. This type of shift is common. Behnke et al. (2016), for example, reported that *"[b]ecause gridded products differ in how they define a calendar day (e.g., local time relative to Coordinated Universal Time), appropriate lag correlations were applied through ~~cross- correlation~~ cross-correlation analysis to account for the ~~several- hour~~ several-hour offset in daily station data."* We performed a lag-correlation analysis on the precipitation products in CAMELS and found a higher correlation between DayMet

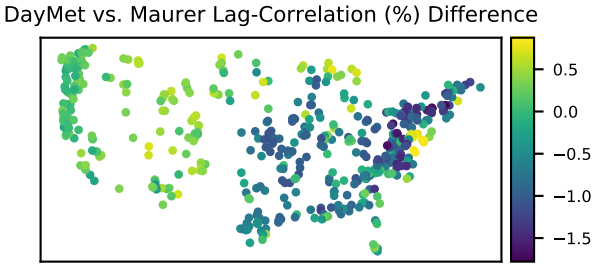


Figure 2. Spatial distributions of lagged vs. non-lagged correlations between DayMet and Maurer test-period precipitation. Positive values indicate that the 1-day lagged correlation is higher.

and Mauer when Mauer was lagged by one day in 325 (of 531) basins. Figure 2 shows the percent difference between lagged vs. non-lagged correlations between DayMet and Maurer.

Each of the forcing products in CAMELS includes daily precipitation (mm/d) and maximum and minimum daily temperature (°C), vapor pressure (Pa), and surface radiation (W/m^2). The original CAMELS data set hosted by the US National Center for Atmospheric Research (Newman et al., 2014) only contains daily mean temperatures for Maurer and NLDAS. CAMELS-relevant Maurer and NLDAS products with daily minimum and maximum temperatures are available from our HydroShare DOI (see data availability section). We used all five meteorological variables from all three data products as inputs into the models. In addition to the three daily forcing data sets from CAMELS, we used the same 27 catchment attributes as Kratzert et al. (2019a, b), which consist of ~~topological, climatic~~topography, climate, vegetation, and soil descriptors (Addor et al., 2017a). Prior to training any ~~model~~models, all input variables were normalized independently by subtracting the CONUS-wide mean and dividing by the CONUS-wide standard deviation.

2.2 Models

Long Short-Term Memory networks (LSTMs) are a type of recurrent neural network (Hochreiter, 1991; Hochreiter and Schmidhuber, 1997b; Gers et al., 1999). LSTMs ~~are a type of~~ have a state-space ~~model that function~~ that evolve through a set of input-state-output relationships. Gates, which are activated linear functions, control information flows from inputs and previous states to current state values (called an input gate), from current states to outputs (called an output gate), and also control the timescale of each element of the state vector (called a forget gate). States (~~usually~~ called cell states) accumulate and store information over time, much like the states of a dynamical ~~system~~systems model. Technical details of the LSTM ~~model~~ architecture have been described in several previous publications in hydrology journals, and we refer the reader to Kratzert et al. (2018) for a detailed explanation geared towards hydrologists.

2.3 ~~Experimental DesignTo conduct our analyses we trained an LSTM model using each~~ Benchmarks

Because all relevant benchmark models from previous studies (Kratzert et al., 2019b, see e.g.) were calibrated using only Maurer forcings, we produced a benchmark using the SAC-SMA model with multiple meteorological forcings. Following Newman et al. (2017), we calibrated SAC-SMA using the dynamically dimensioned search (DDS) algorithm (Tolson and Shoemaker, 2007 implemented in the Spotpy optimization library (Houska et al., 2019) using data from the training period in each basin. SAC-SMA was calibrated separately $n = 10$ times with $n = 10$ different random seeds in each basin for each of the three meteorological data products. This resulted in a total of 30 calibrated SAC-SMA models for each basin.

To check our SAC-SMA calibrations, we compared the performance of our Maurer calibrations against SAC-SMA model from the benchmark data set calibrated by Newman et al. (2017). We used the (paired) Wilcoxon test to test for significance in any difference between the average per-basin performance scores from our $n = 10$ different SAC-SMA calibrations with Maurer forcings vs. the SAC-SMA calibrations with Maurer forcings from Newman et al. (2017). The p-value of this test was $p \approx 0.9$, meaning no significant difference.

Results reported in Section 3 used a simple average of these 30 SAC-SMA ensembles in each basin, which is what we found to be the most accurate overall. We also tested (not reported) a Bayesian model averaging strategy with basin-specific likelihood weights chosen according to relative training-performance of the SAC-SMA ensemble members using Gaussian likelihoods with a wide range of variance parameters. We were not able to achieve a overall higher performance in the test period using an ensembling method more sophisticated than equal-weighted averaging. There are possibilities to potentially improve on this benchmark (e.g., Duan et al., 2007; Madadgar and Moradkhani, 2014), however as will be shown in Section 3, the difference between ensemble averaging and the multi-input LSTMs is large and we would be surprised if any ensembling strategy could account for this difference.

2.4 Experimental Design

We trained $n = 10$ LSTMs using (1)all of the three forcing products together, ~~separate LSTM models~~ (2) for each pairwise combination of forcing products (DayMet & Maurer, DayMet & NLDAS, and Maurer & NLDAS), and ~~separate LSTMs~~ (3) separately for all three forcing products individually.

For each of these seven input configurations, we trained an ensemble of $n = 10$ different LSTMs with different randomly initialized weights. We report the statistics from averaging the simulated hydrographs from each of these 10-member ensembles (single model results are provided in ~~the Appendix~~ Appendix B). Ensembles are used to account for randomness inherent in the training procedure. The importance of using ensembles for this purpose was demonstrated by Kratzert et al. (2019b). Notice that ensembles are used here to mitigate a different type of uncertainty than when using ensembles for combining forcing products. In this case, the model learns how to (dynamically) combine forcing products, and ensembles are used for the same reason as proposed by Newman et al. (2017): to account for randomness in the calibration/training.

~~We used the same time periods for model training and testing as by Kratzert et al. (2019b) - this allows us to directly compare results of this study with the full set of benchmark hydrology models used by that previous study.~~ The training period was from 1 October 1999 to 30 September 2008 (9 years of training data for each catchment) and the test period was 1 October 1989 to 30 September 1999 (10 years of test data for each catchment).

A single LSTM was trained on the combined training period of all 531 basins. Similar to previous studies (Kratzert et al., 2019b, a), we used LSTMs with 256 memory cells and a dropout rate of 0.4 (40%) in the fully connected layer that derives network predictions (streamflow) from LSTM output. All models were trained with a mini-batch size of 256 for 30 epochs using the Adam optimizer (Kingma and Ba, 2014) with an initial learning rate of 1e-3, reduced to 5e-4 after 20 epochs and further reduced to 1e-4 after 25 epochs. All inputs were standardized to have zero mean and unit variance over all 531 catchments collectively. During model evaluation, negative predictions in the original value space were clipped to zero, i.e. no negative discharges. The loss function was the basin-averaged Nash-Sutcliffe Efficiency (NSE), see Kratzert et al. (2019b).

2.5 Analysis

We examined the experiments described above with ~~three types of analysis~~two types of analyses. The goal is to provide ~~different~~ illustrations of how the LSTM ~~leveraged~~ leverages multiple forcing products in spatiotemporally dynamic ways.

- **Analysis 1 - Feature Ablation**: An *ablation study* removes parts of the network to gain a better understanding of the model. We adopted this procedure by removing the different meteorological forcing products in a step-wise fashion and ~~comparing the individual results by using multiple~~ subsequently comparing results using several performance metrics and hydrologic signatures (see Table 1). To provide context~~we also compare them against a family of conceptual and process-based hydrological models~~ , we also benchmarked the LSTMs against ensembles of SAC-SMA models (see Section 2.3).

- **Analysis 2 - ~~Precipitation Uncertainty~~Sensitivity & Contribution**~~: We used triple collocation to estimate spatially-varying error characteristics of the three precipitation forcing products, and assessed relationships between these error statistics with the performance of both single- and multiple-forcing LSTMs. These experiments help us understand where we can expect value from using multiple forcing products in a single model.~~

- ~~**Analysis 3 - Sensitivity & Contribution**~~: We performed an input attribution analysis of the trained LSTM models to quantify how the ~~LSTM learned to~~ trained LSTMs leverage different forcing products in ~~spatiotemporally dynamic ways~~different places and under different hydrologic conditions.

In addition, we performed an analysis that correlates estimated uncertainty in different precipitation products with LSTM performance to help understand in what sense the LSTM is using different precipitation data to mitigate data uncertainty directly. This analysis is presented in Appendix E.

2.5.1 Analysis 1: Feature Ablation

All LSTM ensembles were trained using a squared-error loss function (the ~~basin-averaged NSE~~ average of the basin-specific NSE values), however we are interested to know how the models simulate different aspects of the hydrograph. As such, we report a collection of hydrologically-relevant performance metrics outlined in Table 1. These statistics include the standard time-average performance metrics (e.g., NSE, KGE), as well as comparisons between observed and simulated hydrologic

Table 1. Description of the performance metrics (top part) and signatures (bottom part) considered in this study. For each signature, we derived a metric by computing the Pearson correlation between the signature of the observed flow and the signature of the simulated flow over all basins. Description of the signatures taken from Addor et al. (2018)

Metric/Signature	Description	Reference
NSE	Nash-Sutcliff efficiency	Eq. 3 in Nash and Sutcliffe (1970)
KGE	Kling-Gupta efficiency	Eq. 9 in Gupta et al. (2009)
Pearson rt	Pearson correlation between observed and simulated flow	
α-NSE	Ratio of standard deviations of observed and simulated flow	From Eq. 4 in Gupta et al. (2009)
β-NSE	Ratio of the means of observed and simulated flow	From Eq. 10 in Gupta et al. (2009)
FHV	Top 2% peak flow bias	Eq. A3 in Yilmaz et al. (2008)
FLV	Bottom 30% low flow bias	Eq. A4 in Yilmaz et al. (2008)
FMS	Bias of the slope of the flow duration curve between the 20% and 80% percentile	Eq. A2 Yilmaz et al. (2008)
Peak-Timing	Mean peak time lag (in days) between observed and simulated peaks	See Appendix D
Baseflow index	Ratio of mean daily baseflow to mean daily discharge	Ladson et al. (2013)
HFD mean	Mean half-flow date (date on which the cumulative discharge since October first reaches half of the annual discharge)	Court (1962)
High flow dur.	Average duration of high-flow events (number of consecutive days >9 times the median daily flow)	Clausen and Biggs (2000), Table 2 in Westerberg and McMillan (2015)
High flow freq.	Frequency of high-flow days (>9 times the median daily flow)	Clausen and Biggs (2000), Table 2 in Westerberg and McMillan (2015)
Low flow dur.	Average duration of low-flow events (number of consecutive days <0.2 times the mean daily flow)	Olden and Poff (2003), Table 2 in Westerberg and McMillan (2015)
Low flow freq.	Frequency of low-flow days (<0.2 times the mean daily flow)	Olden and Poff (2003), Table 2 in Westerberg and McMillan (2015)
Q5	5% Flow quantile (low flow)	
Q95	95% Flow quantile (high flow)	
Q mean	Mean daily discharge	
Runoff ratio	Runoff ratio (ratio of mean daily discharge to mean daily precipitation, using DayMet precipitation)	Eq. 2 in Sawicz et al. (2011)
Slope FDC	Slope of the flow duration curve (between the log-transformed 33rd and 66th streamflow percentiles	Eq. 3 in Sawicz et al. (2011)
Stream elasticity	Streamflow precipitation elasticity (sensitivity of streamflow to changes in precipitation at the annual time scale, using DayMet precipitation)	Eq. 7 in Sankarasubramanian et al. (2001)
Zero flow freq.	Frequency of days with zero discharge.	

signatures. The hydrologic signatures we report are the same ones used by Addor et al. (2018). For each hydrologic signature, we ~~compute~~ computed the Pearson correlation between the ~~signature derived from the observed discharge and derived from the simulated discharge of~~ signatures derived from observed discharge vs. from simulated discharge in each basin. Correlation metrics were calculated on simulated vs. observed signatures in all basins.

~~To provide a baseline for comparison, LSTM ensembles were benchmarked against the same family of hydrological models used for benchmarking by Kratzert et al. (2019b). These models are: (i) SAC-SMA (Burnash et al., 1973; Burnash, 1995) coupled with the Snow-17 snow routine (Anderson, 1973), hereafter referred to as SAC-SMA, (ii) VIC (Liang et al., 1994), (iii) FUSE (Clark et al., 2008; Henn et al., 2008) (three different model structures, 900, 902, 904), (iv) HBV (Seibert and Vis, 2012) , and (v) mHM (Samaniego et al., 2010; Kumar et al., 2013). Some of these models were calibrated to individual basins and others were regionally calibrated. All of the benchmarks used Maurer forcings and all were calibrated and validated on the same time periods used in this study. In order to avoid any potential or implicit bias, we did not run any of our own benchmark models~~

~~- all models were solicited originally by Kratzert et al. (2019b) from different groups with experience running each individual model. The whole family of benchmark model runs is only available in 447 of the 531 CAMELS catchments, chosen by Newman et al. (2017). Thus, while we use the set of 531 catchments for all other parts of this study, we only considered 447 catchments for benchmarking against traditional hydrology models.~~

2.5.2 Analysis 2: ~~Precipitation Uncertainty~~Sensitivity & Contribution

~~The objective of the second analysis is to demonstrate that the multiple-forcing model learns to leverage patterns in forcing data error structures. Our approach was to relate error characteristics of the different precipitation products with model performance, and with performance improvements due to using multiple forcing products. We used triple collocation to estimate error characteristics of the different forcing products. Triple collocation is a statistical technique to estimate error variances of three or more noisy measurement sources without knowing the true values of the measured quantities (Stoffelen, 1998; Scipal et al., 2010) . Its major assumptions are that the error models are linear and independent between sources; in particular, that all (three or more) measurement sources are each a combination of a scaled value of the true variable plus some additive random noise:~~

$$M_{i,t} = \alpha_i T_t + \varepsilon_{i,t},$$

~~where M_* are measurement values (i.e. here the modeled precipitation values), subscript i represents the source (DayMet, Maurer, NLDAS), and subscript t represents the timestep in the test period (1 October 1989 to 30 September 1999); T_* is the unobserved true value of total precipitation in a given catchment on a given day; ε_* are i.i.d. measurement errors from any distribution.~~

~~The linearity assumption is not appropriate for precipitation data, which are typically assumed to have multiplicative error. Following Alemohammad et al. (2015), we assumed a multiplicative error model for all three precipitation source, and converted these to linear error models by working with the log-transformed precipitation data:~~

$$M_{i,t} = \alpha_i T_t^{\beta_i} + e^{\varepsilon_{i,t}}$$

$$\ln(M_{i,t}) = \alpha_i + \beta_i T_t + \varepsilon_{i,t}.$$

~~Standard triple collocation is then applied, so that estimates of the error variances for each source are:~~

$$\sigma_i = C_{i,i} - \frac{C_{i,j} C_{i,k}}{C_{j,k}},$$

~~for all i, j, k, where $C_{i,j}$ is the covariance between the time series of source i and source j; σ_i is the variance of the distribution that each i.i.d. $\varepsilon_{i,t}$ is drawn from.~~

~~Additionally, extended triple collocation (McColl et al., 2014) allows us to derive the correlation coefficients between measurement sources and truth as:~~

$$\rho_i = \frac{C_{i,j} C_{i,k}}{C_{i,i} C_{j,k}}.$$

~~This triple collocation analysis was applied separately in each of the 531 CAMELS catchments to obtain basin-specific estimates of the error variances, σ_i, and truth-correlations, ρ_i, for each of the three precipitation products. Albeit the assumption that the forcing products have independent error structures (i.e. $\varepsilon_{i,t} \perp\!\!\!\perp \varepsilon_{j,t}$) is not met in our case we expect the results to be robust enough for the purpose at hand.~~

2.5.3 ~~Analysis 3: Sensitivity & Contribution~~

All neural networks (like LSTMs) are differentiable ~~*almost everywhere*~~ almost everywhere by design. Therefore, a gradient-based input contribution analysis seems natural. However, as discussed by Sundararajan et al. (2017), the naive solution of using local gradients ~~is not a~~ does not provide reliable measures of sensitivity, since gradients might be flat even if the model response is heavily influenced by a particular input data source (which is not by necessity a bad property, see for example Hochreiter and Schmidhuber, 1997a). This is especially true in neural networks, where activation functions often include step-changes over portions of the input space - e.g., the sigmoid and hyperbolic tangent activation functions used by LSTMs have close-to-zero gradients at both extremes (see also: Shrikumar et al., 2016; Sundararajan et al., 2017).

Sundararajan et al. (2017) proposed a method of input attribution for neural networks which accounts for this ~~described~~ lack of local sensitivity: this method is called *integrated gradients*. Integrated gradients are a path integral of the gradients from some baseline input value x', to the actual value of the input, x:

$$\text{IntegratedGrads}_i^{\text{approx}}(\boldsymbol{x}) := \frac{\mathbf{x}_i - \mathbf{x'}_i}{m} \sum_{k=1}^{m} \frac{\partial F\left(\tilde{\mathbf{x}}\right)}{\partial \tilde{\mathbf{x}}_i}\bigg|_{\tilde{\mathbf{x}}=\mathbf{x'}+\frac{k}{m}(\mathbf{x}-\mathbf{x'})} . \tag{1}$$

We used a value of zero precipitation everywhere as the baseline for calculating integrated gradients with respect to the three different precipitation forcings (DayMet, Maurer, NLDAS). We calculated the integrated gradients of each daily streamflow estimate in each CAMELS basin during the 10-year test period with respect to precipitation inputs from the past 365 days (the look-back period of the LSTM). That is, on day $t = T$, we calculated $1095 = 3 * 365$ integrated gradient values related to the three precipitation products. The relative integrated gradient values quantify how the LSTM combines precipitation products over time, over space, and also as a function of lag or lead-time into the current streamflow prediction. In theory, one has to take into account the effect of "*explaining away*", when analysing the decision process in models (Pearl, 1988; Wellman and Henrion, 1993). However, we assume that if evaluated over hundreds of basins and thousands of time steps, this effect is largely averaged out and therefore the analysis provides an indication of the actual information used by the model.

3 Results & Discussion

3.1 Results: Analysis 1 - Feature Ablation

The feature ablation analysis compared NSE values over 10-year test periods from the CAMELS basins for the seven distinct input combinations. As shown in Fig. 3, the three-forcing ~~model~~ LSTM ensemble had a median NSE value of 0.82 for the ~~447 basins, which were available for all benchmarking models~~531 basins. The three-forcing model outperformed all two-

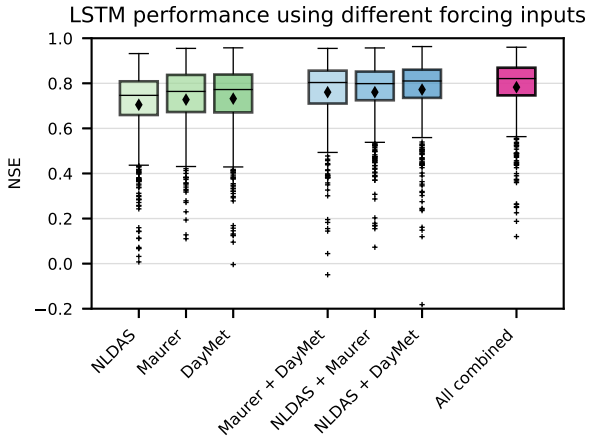


Figure 3. Test-period comparison between single-forcing and multiple-forcing LSTM ensembles ($n = 10$) over ~~447~~ 531 CAMELS basins. All differences are statistically significant ($\alpha = 0.001$), with the exceptions of "DayMet" vs. "Maurer" ($p \approx 0.08$) and "NLDAS + Maurer" vs. "Maurer + DayMet" ($p \approx 0.4$)

forcing models. Similarly, all two-forcing models outperformed all single-forcing models (all improvements were statistically significant at ~~$\alpha = 0.001$~~$\alpha = 0.05$, using the Wilcoxon test). The best single-forcing LSTM had a median NSE of ~~0.76~~0.77. This indicates that the LSTM was able to leverage unique information in the precipitation signals (this is not an unusual finding in the context of machine learning, see for example: Sutton, 2019). We also note that the single-forcing LSTM with Maurer inputs outperformed the single-forcing NLDAS model, which agrees with the results of Behnke et al. (2016) who showed that Maurer precipitation was generally more accurate than NLDAS precipitation.

To put these results into context, Fig. 4 compares all LSTMs against ~~the~~ benchmark hydrology models~~in the 447 basins where simulations of all benchmark models were available~~, which are all ensembles of SAC-SMA models that were calibrated for each of the three different forcings. All LSTM models were better than all corresponding benchmark models through the entire CDF curve. ~~As a point of reference, the difference in the median NSE between the best-performing~~ The following points can be seen in Fig 4. First, the SAC-SMA sees a large improvement from using two-forcing products ensembles - this improvement was larger than the corresponding improvement in the LSTMs. However, adding calbrated SAC-SMA models from a third data product did not increase performance by much (see e.g. Fig. 4a, where the NLDAS + DayMet ensemble CDF overlaps most of the time with the three forcing ensemble). In contrast, CDFs of the LSTM results show a constant improvement from one- to two-forcing models, and from two- to three-forcing models.

Second, the difference between the worst single-forcing ~~LSTM (DayMet) and the best-performing traditional hydrology model (HBV)was 0.09, while using all three CAMELS forcings increased that improvement over traditional models by another 0.055 (61%) . The total improvement in the median NSE of the multi-forcing LSTM over the best-performing hydrology model was 0.143 (21%).~~ ensemble and the three-forcing ensemble is larger for the LSTM (ΔNSE=0.074) than for the SAC-SMA

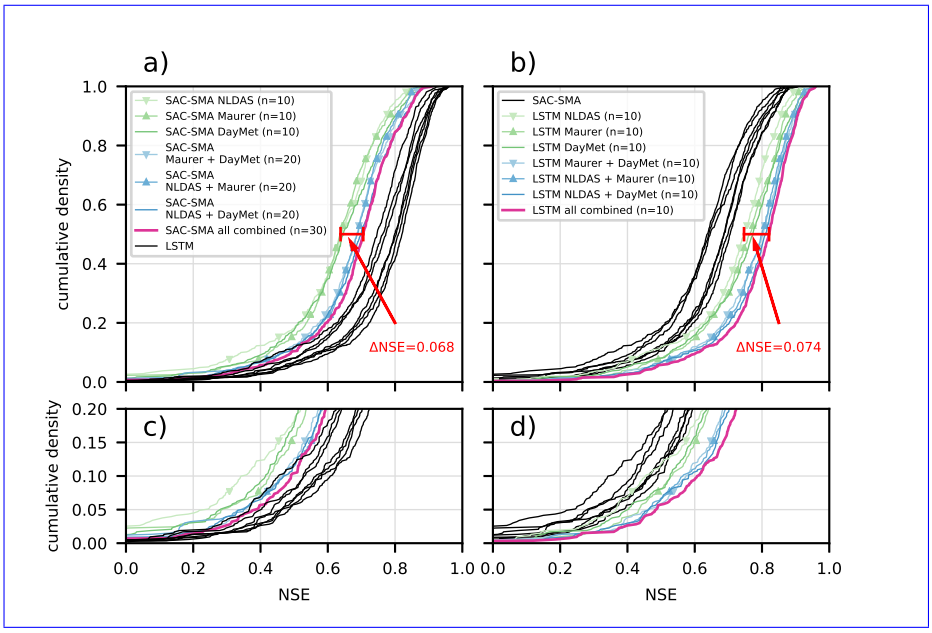


Figure 4. Empirical cumulative density function of the NSE performance over the ~~447 commonly modelled~~ 531 basins of ~~all LSTMs~~ different SAC-SMA ensembles (a and ~~benchmark models~~c) and different LSTM ensembles (b and d). Top row shows the entire range of the cumulative density function, while the bottom row shows the lower range in more detail. The red indicator lines mark the median NSE difference between the ~~best hydrological benchmark model (~~worst single forcing ensemble ~~of 100 calibrated HBV models)~~ and the ~~LSTM~~ multi-forcing ensemble of ~~our previous publication (Kratzert et al., 2019b), as well as~~ the ~~current best~~LSTM and SAC-SMA, ~~if trained with all forcing products combined~~respectively.

(ΔNSE = 0.068). This difference could arise from the fact that the LSTM is better able to handle the data shift of the Maurer forcings that occurs in some of the basins (see. Section 3.3), while this is impossible for the SAC-SMA ensemble.

Third, the worst-performing single-forcing LSTM ensemble (i.e., with NLDAS forcings) was significantly better ($p < 1e-13$) than the whole $n = 30$ SAC-SMA ensemble, which uses all three forcing products (i.e., the best SAC-SMA result that we found). In fact, even the average single LSTM (not the full $n = 10$ ensemble) trained with NLDAS forcings is as good as the $n = 30$ SAC-SMA ensemble (see Appendix B for non-ensemble LSTM performances), and the average single LSTM (not the ensemble) trained with Maurer or DayMet forcings was significantly better ($p < 1e-8$) than the $n = 30$ SAC-SMA ensemble.

Fourth, the ranking of the forcing products is not as clear for the SAC-SMA ensembles as it was the LSTM ensembles (there is more separation in the LSTM single-forcing CDFs than the SAC-SMA singel-forcing CDFs). However, qualitatively, the same ranking is visible, i.e., that DayMet models are better than NLDAS or Maurer, and that NLDAS + DayMet produce the best two-forcing results.

Table 2. Values of the benchmarking metrics from Table 1. Bold indicates the best ~~value per metric or signature~~model ($\alpha < 0.05$). Multiple bold numbers per row ~~mean that there is~~ indicate no ~~statistical~~ significant difference~~to the best performing model in the given metric/signature at ($\alpha = 0.001$)~~.

toprule	LSTM all forcing								SAC-SMA ~~VIC VIC mHM mHM HBV HBV~~								
~~Metric (all)~~	~~(basin)~~	~~(CONUS)~~	~~(basin)~~	~~(CONUS)~~	~~(lower)~~	~~(upper)~~	~~(900)~~	~~(902~~ensemble (n=10)	~~(904~~ensemble (n=30								
NSE[i] (median)	~~**0.82~~**0.821								~~0.60~~	~~0.55~~	~~0.31~~	~~0.67~~	~~0.53~~	~~0.42~~	~~0.68~~	~~0.6~~	
NSE[i] (mean)	~~**0.79~~**0.783								~~0.56~~	~~0.52~~	~~0.17~~	~~0.63~~	~~0.44~~	~~0.24~~	~~0.63~~	~~0.5~~	
KGE[ii]	~~**0.81~~**0.801								~~0.63~~	~~0.59~~	~~0.26~~	~~0.69~~	~~0.47~~	~~0.39~~	~~0.68~~	~~0.6~~	
Pearson r[iii]	~~**0.92~~**0.915								~~0.79~~	~~0.76~~	~~0.65~~	~~0.83~~	~~0.79~~	~~0.71~~	~~0.83~~	~~0.8~~	
α-NSE[iv]	~~**0.87~~**0.861								~~0.78~~	~~0.73~~	~~0.46~~	~~0.81~~	~~0.59~~	~~0.58~~	~~0.79~~	~~0.8~~	
β-NSE[v]	~~-0.03~~	~~-0.07~~	~~**-0.02**~~	~~-0.07~~	~~-0.04~~	~~-0.04~~	~~-0.02~~	~~**-0.01**~~	~~-0.03~~	~~-0.05~~	-0.028				~~-0.07~~	**0.024**	
FHV[vi]	~~**-13.32~~**-13.818								~~-20.36~~	~~-28.14~~	~~-56.48~~	~~-18.64~~	~~-40.18~~	~~-41.86~~	~~-18.49~~		
FLV[vii]	~~40.81~~	~~37.42~~	~~-74.77~~	~~18.87~~	~~11.43~~	~~36.80~~	~~23.88~~	~~18.34~~	~~**-10.54~~**41.277						~~-68.22~~	~~-67.60~~	**49.641**
FMS[viii]	~~-8.15~~	~~-14.36~~	~~**-6.56~~**-8.087						~~-27.99~~	~~-7.22~~	~~-30.35~~	~~-15.94~~	~~-24.94~~	~~**-5.09**~~			
Peak-Timing[ix]	~~**0.36~~**0.370								~~0.81~~	~~0.69~~	~~0.92~~	~~0.69~~	~~0.75~~	~~1.21~~	~~0.63~~	~~0.5~~	

[i]: *Nash-Sutcliffe efficiency:* $(-\infty, 1]$, *values closer to one are desirable.*
[ii]: *Kling-Gupta efficiency:* $(-\infty, 1]$, *values closer to one are desirable.*
[iii]: *Pearson correlation:* $[-1, 1]$, *values closer to one are desirable.*
[vi]: *α-NSE decomposition:* $(0, \infty)$, *values close to one are desirable.*
[v]: *β-NSE decomposition:* $(-\infty, \infty)$, *values close to zero are desirable.*
[vi]: *Top 2 % peak flow bias:* $(-\infty, \infty)$, *values close to zero are desirable.*
[vii]: *30 % low flow bias:* $(-\infty, \infty)$, *values close to zero are desirable.*
[viii]: *Bias of FDC midsegment slope:* $(-\infty, \infty)$, *values close to zero are desirable.*
[ix]: *Lag of peak timing:* $(-\infty, \infty)$, *values close to zero are desirable.*

Table 2 and Table 3 give ~~the~~ benchmarking results from all metrics and signatures in Table 1. The three-forcing LSTM ~~out-performs all benchmark models against~~ significantly out-performed the three-forcing SAC-SMA ensemble in all metrics except β-NSE decomposition, ~~the bias of the slope of the flow duration curve (FMS) and the bias of the low flows (FLV~~where the SAC-SMA ensemble was better, and FLV where the difference was not significant (see Tab. 2). The three-forcing LSTM also ~~out-performs all benchmark models against all hydrologic signatures except~~ significantly out-performed the three forcing SAC-SMA ensemble in all signatures (see Tab. 3), except the HFD mean and the ~~ones related to low-flows (frequency of zero flows and frequency and duration of flows below 20% of basin-average). We therefore note~~ Q95, where the difference was not significant. Note that the LSTM ~~approach~~ - while generally providing the best ~~available model~~ model overall - ~~still~~ has approximation difficulties towards the extreme lower-end of the runoff distribution (low flow duration, low flow frequency, and zero flow frequency).

~~Looking at~~

Table 3. Values of the correlation coefficients (over ~~447~~ 531 basins) of the simulated vs. observed hydrological signatures from Table 1. Bold indicates ~~that the model is not statistically different than~~ the best ~~performing~~ model ~~in a given metric~~($\alpha < 0.05$). Multiple bold numbers per row indicate no significant difference.

toprule ~~Signature (all)~~	LSTM all forcing ~~(basin)~~ ~~(CONUS)~~ ~~(basin)~~ ~~(CONUS)~~ ~~(lower bound)~~ ~~(upper bound)~~ ~~(900)~~ ~~(902~~ensemble (n=10)	SAC-SMA ~~VIC~~ ~~VIC mHM~~ ~~mHM~~ ~~(904~~ens
Baseflow index	~~**0.92**~~**0.93**	~~0.84~~ ~~0.75~~ ~~0.29~~ ~~0.78~~ ~~0.~~
HFD mean	**0.98**	~~0.92~~ ~~0.92~~ ~~0.87~~ ~~0.95~~ ~~0.91~~
High flow dur.	~~**0.88**~~**0.84**	0.72 ~~0.60~~ ~~0.51~~ ~~0.71~~ ~~0.~~
High flow freq.	~~**0.85**~~**0.81**	~~0.67~~ ~~0.66~~ ~~0.43~~ ~~0.63~~ ~~0.52~~
Low flow dur.	~~0.48~~ ~~0.30~~ ~~0.31~~ ~~0.23~~ ~~0.39~~ ~~0.38~~ ~~0.40~~ ~~**0.55**~~**0.50**	~~0.32~~ ~~0.~~
Low flow freq.	~~0.79~~ ~~0.75~~ **0.79**	0.63 ~~0.26~~ ~~0.61~~ ~~0.33~~
Q5	**0.96**	~~0.81~~ ~~0.74~~ ~~0.42~~ ~~0.81~~ ~~0.64~~
Q95	**0.99**	~~0.98~~ ~~0.97~~ ~~0.90~~ ~~0.98~~ ~~0.93~~
Q mean	**1.00**	~~0.98~~ ~~0.98~~ ~~0.95~~ ~~0.98~~ ~~0.9~~
Runoff ratio	**0.99**	~~0.96~~ ~~0.94~~ ~~0.83~~ ~~0.94~~ ~~0.83~~
Slope FDC	~~**0.67**~~**0.65**	~~0.61~~ 0.62 ~~0.44~~ ~~0.49~~ ~~0.~~
Stream elasticity	~~**0.75**~~**0.72**	~~0.66~~ ~~0.51~~ ~~0.31~~ ~~0.66~~ ~~0.57~~
Zero flow freq.	~~0.02~~ ~~**0.46**~~**0.03**	~~0.36~~ ~~0.10~~ ~~-0.00~~ ~~NaN~~

Figure 5 shows the spatial distribution of the performance differences between the best single-forcing model and the three-forcing model in all basins~~used for model training (i.e., 531 basins instead of the 447 basins used for the benchmarking described above), it is evident that the~~. The three-forcing LSTM outperformed the single forcing ~~models almost everywhere (Fig. 5)~~LSTMs almost everywhere. Individual exceptions where ~~"~~"*less is more*" do~~however~~, however, exist (e.g.~~,~~ Southern California). Concretely, the three-forcing model was better than the best single forcing model in 66% of the basins (351 of 531) and had a higher NSE than the individual single-forcing LSTMs in over 80% of the basins

3.2 Results: Analysis 2 - ~~Precipitation Uncertainty~~Sensitivity & Contribution

~~DayMet typically produces lower NSE values in basins where triple collocation reported that the DayMet precipitation error variances are high. This is what we would expect: low skill in basins with high precipitation error; however we did not see similar patterns with the other two precipitation products (see Fig. E1, where the triple collocation error variances and truth-correlation are plotted against the NSE scores of the single-source models). In fact, the NLDAS LSTM tends to perform worse in basins with lower precipitation error (as estimated by triple collocation).~~

~~Triple collocation error variances and truth-correlations plotted against NSE scores of the single-forcing LSTM models. DayMet typically produces lower NSE values in basins where triple collocation reports that the precipitation error variances~~

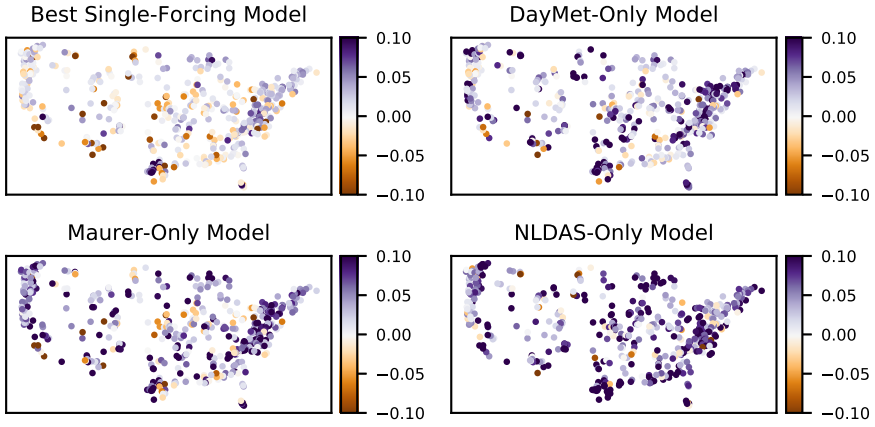


Figure 5. Spatial distribution of the NSE differences between the three-forcing LSTM relative to the best single-forcing model in each basin (top-left subplot), and relative to each single-forcing model (other three subplots). The three-forcing LSTM was better than the best single-forcing model in 351 of 531 basins (66%) and was better than each single-forcing model in: 443 (83%; DayMet), 456 (86%; Maurer), and 472 (89%; NLDAS) basins.

~~are high, whereas NLDAS produces lower NSE values in basins where triple collocation reports that the error variances are low. There is no apparent pattern in the Maurer data.~~

~~Figure E2 is an adapted version of Fig. E1 that highlights a few high-skill, high-variance NLDAS basins in blue. These basins correspond to a cluster of basins in the Rocky Mountains (Fig. E3) where NLDAS has low correlation with the other two products but still yields high-skill LSTM simulations.~~

~~As in Fig. E1 the triple collocation error variances and truth-correlations are plotted against NSE scores of the single-forcing LSTM models. The coloring shows the anomalous NLDAS basins in blue and all others in red. For these basins NLDAS has low correlation with the other two products but still yields high-skill simulations.~~

~~Spatial distribution of anomalous NLDAS basins shown in Fig. E2 (left) compared with elevation of the CAMELS basins (right).~~

~~Triple collocation measures (dis)agreement between measurement sources, rather than error variances directly. Figure E4 plots model performance against the individual variances of the precipitation products in each basin. This figure shows that the single-forcing DayMet LSTM tends to perform better in catchments with higher total precipitation variance (not triple collocation error variance), indicating better performance in wetter catchments. This is not true for the other two models, where higher total variance is associated with a higher variance in model skill, indicating higher proportion of the variance due to measurement error.~~

~~Performance of single-input models relative to the total variance of log-precipitation in each basin. The DayMet model tends to perform better in wetter basins (as the total DayMet variance increases), but the other two products have poor performing basins in catchments with high precipitation variance.~~

~~To analyse the synergy due to using all forcings in a single LSTM we transposed the NSE *improvements* in each basin (due to using all three forcing products in the same LSTM) with the log-determinant of the covariance matrix of all three (standardized, log-transformed) precipitation products (Fig. E5). The log-determinant is a proxy for the joint entropy of the three (standardized, log-transformed) products, and increases when there is larger disagreement between the three data sets. Unlike in Fig. E4, the variances in Fig. E5 were calculated after removing the mean and overall variance of each log-transformed precipitation product so that the log-determinant of the covariance is not affected by the overall magnitude of precipitation in each catchment (i.e., does not increase in wetter catchments). With the exception of the anomalous NLDAS basins, Fig. E5 shows that the three-forcing model offers improvements with respect to the single-forcing models when there is larger disagreement between the three data sets.~~

~~Fractional increase in NSE from the three-forcing model relative to the single-forcing models plotted against the log-determinant of the covariance matrix of all three (standardized, log-transformed) precipitation products. With the exception of the anomalous NLDAS basins (blue markers), the three-forcing model offers improvements with respect to the single-forcing models when there is larger disagreement between the three data sets. The three-forcing model learned to leverage synergy in these three precipitation products.~~

3.3 ~~Results: Analysis 3 - Sensitivity & Contribution~~

Figure 6 shows the time- and basin-averaged integrated gradient of ~~the~~ one of the $n = 10$ multi-forcing ~~LSTM~~ LSTMs as a function of lead time. To reiterate from above, the integrated gradient is a measure of input attribution, or sensitivity, such that inputs with higher integrated gradients have a larger influence on model outputs. Integrated gradients shown in Fig. 6 were averaged over all ~~timesteps~~ time steps in the test period, and also over all basins. This figure shows the sensitivity of streamflow at time $t = T$ to each of the three precipitation inputs at times $t = T - s$ where s is the lag value on the x-axis. The main takeaways from this high-level illustration of ~~the~~ input sensitivities are: (1) that the sensitivity of current streamflow to precipitation decays with lead time (i.e., time before present) and (2) that the multi-forcing model has learned to ignore the Maurer input at the present ~~timestep~~time step. The reason for the latter is the time shift in the Maurer product illustrated in Fig. 2.

Figure 6 shows results from only one of $n = 10$ model repetitions, however we performed an integrated gradient analysis on all $n = 10$ multi-input LSTMs (not shown), and the results were qualitatively similar. It is difficult to show all results on the same figure because the values are relative, so integrated gradients between two different models often have different absolute scales - the results presented for a single model in Fig. 6 are representative.

The multi-forcing ~~LSTM~~ LSTMs learned to combine the different precipitation products in spatiotemporally variable ways. Fig. 6 demonstrates the overall behavior of the multi-forcing LSTM. It is, however a highly condensed aggregate of a highly non-linear system. As such, a lot of specific information is lost ~~- as is always the case when nonlinearities are aggregated~~in that figure.

~~Therefore, Fig. 7 details the overall model behavior (through the lense of integrated gradients)~~ Figure 7 shows integrated gradients by basin, and up to a lead time of $s = 3$ days prior to present. The model largely ignores Maurer precipitation at

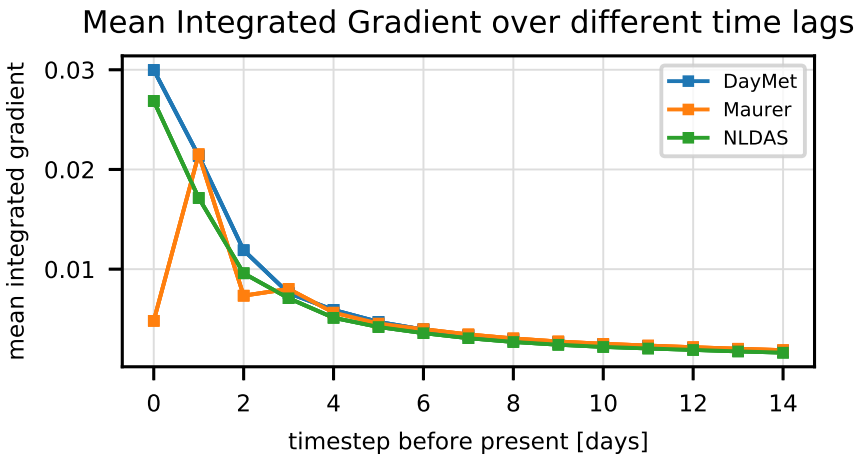


Figure 6. Time- and basin-averaged integrated gradients as a function of lag time (days before current streamflow prediction) of the three precipitation products. Because of the time shift shown in Fig. 2, the model has learned to ignore the Maurer input at the current ~~timestep~~time step.

the current ~~timestep~~ time step in most basins - as ~~is already~~ was apparent in Fig. 6, but the ratio of the contributions of each product ~~varies between basin~~(averaged over the whole test-period hydrograph) varies between basins. Figure 7 shows relative contributions of each precipitation product, but it is important to note that the overall importance of precipitation also varies between basin.

~~Similarly, Fig.~~ Figure 8 shows the spatial ~~extend~~ distribution of the most sensitive ~~contribution over all time steps (left-subplot)~~ precipitation contribution (averaged over the whole hydrograph in each basin) in the left-subplot, and the the overall sensitivity to all three ~~products combined (~~precipitation products combined in the right-subplot~~), which~~ . The latter (total sensitivity to precipitation relative to all other inputs) is highly correlated with the total (or average) precipitation in the basin. ~~. That is, they display the sum of the integrated gradients over time, lag, and product. From the right-subplot it becomes evident that the precipitation has a larger contribution to the sensitivity of streamflow predictions in wetter basins. Figure 8 also shows the product with the highest overall contribution in each basin.~~

It is possible to break the spatial relationship down even further. ~~Concretely, we did examine at the~~ The spatial distribution of the highest-ranked product as a function of the lag time for rising and falling limits ~~. We can then see that~~ is shown in Fig. 9. This figure shows some of the nuance in how the multi-forcing LSTM ~~learns~~ learned to combine the different ~~products in very nuanced ways,~~ precipitation products - by distinguishing between different memory timescales in different basins for different hydrological conditions (~~Fig.9~~i.e., rising and falling limbs of the hydrograph).

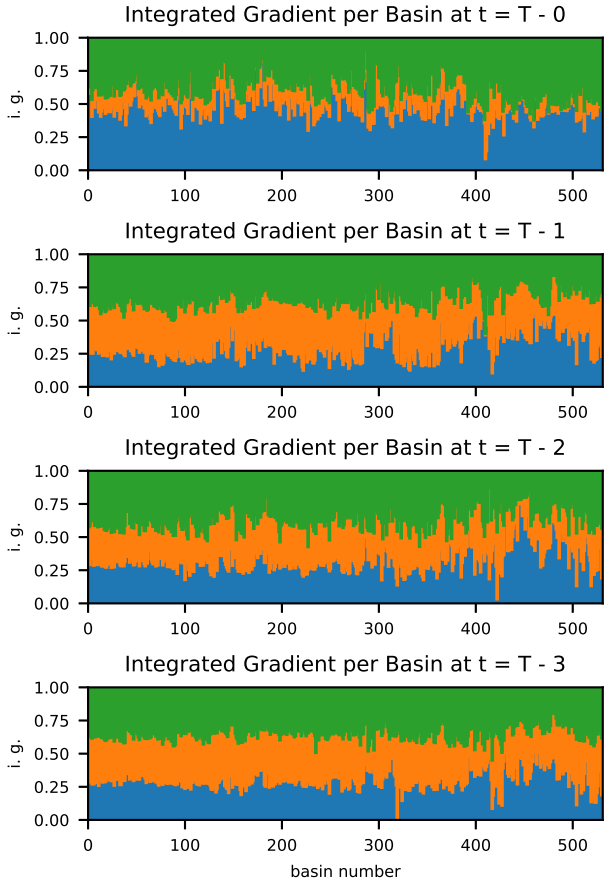


Figure 7. Expansion of Fig. 6 by individual basins, truncated at a lag of $s = 3$. The multi-forcing LSTM combined the precipitation products in different ways in different basins. DayMet is generally more important in high-number basins, located in the Pacific Northwest

4 Conclusions

The purpose of this paper is to show ~~how~~ that LSTMs can leverage different precipitation products in spatiotemporally dynamic ways to improve streamflow simulations. ~~The~~ These experiments show that there exist systematic and location- and ~~time- specific~~ time-specific differences between different precipitation products that can be learned and leveraged by deep learning. As might be expected, the LSTMs tested here tended to improve hydrological simulations more when there were larger disagreement between different precipitation estimates in a given basin (see Appendix E).

It is worth comparing these findings with classical conceptual and process-based hydrological models that treat precipitation estimate as an unique input. Current best-practice for using multiple precipitation products is to run an ensemble of hydrological models, such that each forcing data set is treated independently. Deep learning models ~~not only~~ have the ability to use a larger number and variety of inputs than classical hydrology models~~. As a matter of fact, deep learning~~ , and in fact, DL models do

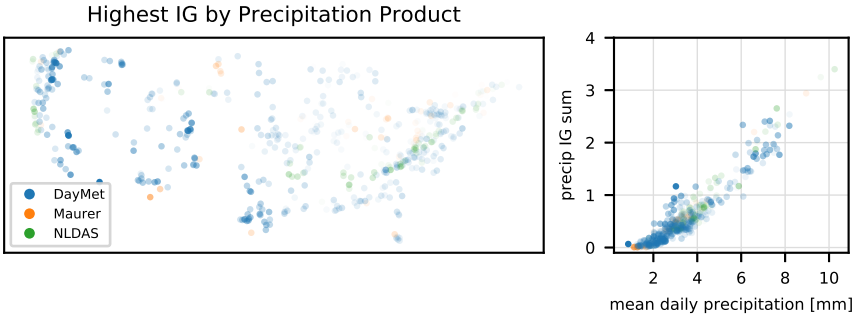


Figure 8. The forcing product with highest overall contribution (sensitivity) in each basin (left-hand subplot) - averaged over prediction time step and lag. The alpha value (opacity) of each dot on this map is a relative measure of the fraction of the total integrated gradients of all three precipitation products (summed over time, lag, and product) due to the highest-contributing product. The right-hand subplot shows that the total integrated gradient summed over all three precipitation products is highly correlated with total precipitation in the basin.

not need inputs that represent any given hydrological variable or process, and therefore have the potential to use less highly processed input data like remote sensing brightness temperatures, etc. Future work might focus on building runoff models that take as inputs the raw measurements that were used to create standard precipitation data products.

Deep learning provides possibilities not only for improving the quality of regional (Kratzert et al., 2019b) and even ungauged (Kratzert et al., 2019a) simulations, but also potentially for replacing large portions of ensemble-based strategies for uncertainty quantification (e.g. Clark et al., 2016) with multi-input models. There are many ways to deal with the uncertainty in traditional hydrological modeling workflows~~. Arguably~~, but almost certainly, the most common approach is to use ensembles~~(e.g., Clark et al., 2016)~~. Ensembles can be ~~either~~ opportunistic - i.e., from a set of pre-existing models or data products - or constructed - i.e., sampled from a probability distribution~~- (Clark et al., 2016)~~, but in either case the idea is to use variability to represent lack of perfect information. Clark et al. (2016) advocated for using ensembles as 'hydrologic storylines', which would avoid the problem of sparsity of sampling any explicit or implied probability distributions. No matter how ensembles are used, however, with conceptual and process-based hydrology models, each model takes one precipitation estimate (time series) as input. Multi-input ~~deep learning has~~ DL models have the potential to provide a fundamentally ~~alternative method for assessing~~ different alternative for modeling under this kind of uncertainty, since DL models can learn how to combine different inputs in ways that leverage - in nonlinear ways - all data available to the full simulation task. Future work ~~should additionally~~ could focus on producing predictive probabilities with multi-input deep learning models.

5 Code availability

The code to reproduce all LSTM results and figures will be made available at https://github.com/kratzert/multiple_forcing. Code for running and optimizing SAC-SMA is available from the 'multi-inputs' branch at the following repository: https://github.com/Upstream-Tech/SACSMA-SNOW17.git

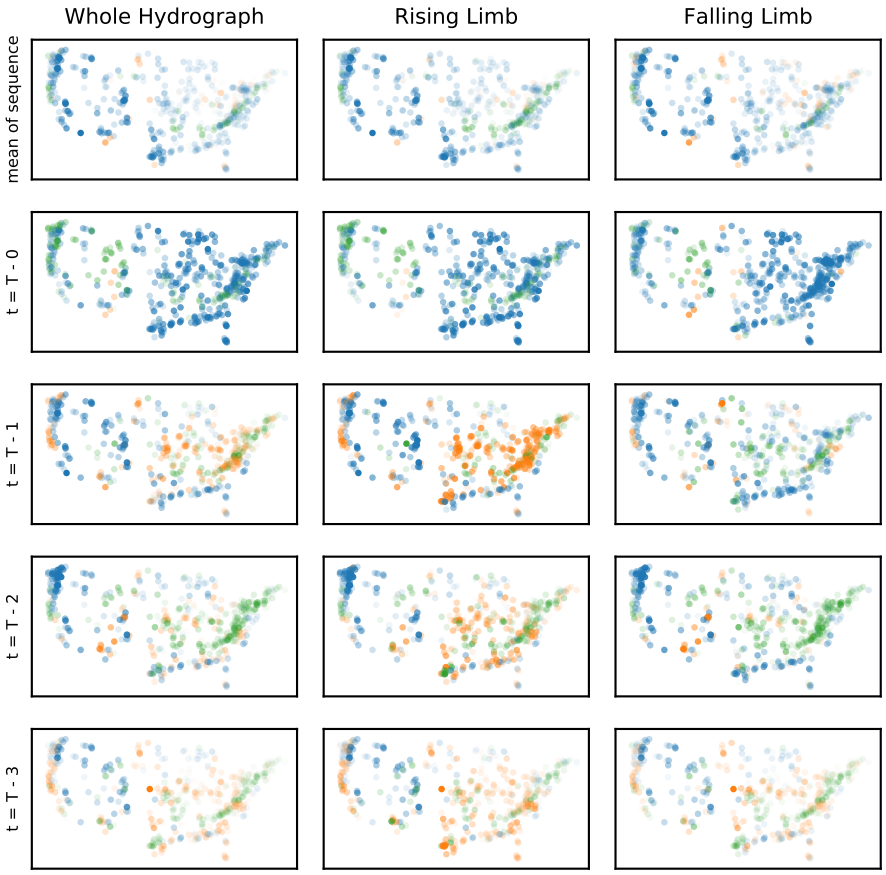


Figure 9. Spatial distribution of highest-ranked precipitation products at specific lags (different rows) over the whole hydrograph (left-hand column), and the rising- and falling-limbs of the hydrograph (center and right-hand columns, respectively), where blue circles denote DayMet, orange circles denote Maurer and green circles denote NLDAS. The take-away from this figure is that the multi-forcing LSTM learns to combine the different products in different ways for different memory timescales in different basins and under different hydrological conditions. The alpha value (opacity) of each dot is a relative measure of the fraction of the total integrated gradients of all three precipitation products due to the highest-contributing product.

6 Data availability

The validation periods of all benchmark models used in this study are available at https://doi.org/10.4211/hs.474ecc37e7db45baa425cdb4fc1b61e1. The extended Maurer forcings, including daily minimum and maximum temperature, are available at https://doi.org/10.4211/hs.17c896843cf940339c3c3496d0c1c077. The extended NLDAS forcings, including daily minimum and maximum temperature, are available at https://www.hydroshare.org/resource/0a68bfd7ddf642a8be9041d60f40868c/.

Table B1. Average single LSTM performance over a variety of metrics. The average single model performances is computed as the mean of the metric of the the $n = 10$ model repetitions.

	NLDAS	Maurer	DayMet	Maurer + DayMet	NLDAS + Maurer	NLDAS + DayMet	All combined
NSE^{i} (median)	0.72	0.73	0.74	0.77	0.77	0.79	0.80
	±0.003	±0.003	±0.002	±0.003	±0.004	±0.002	±0.001
NSE^{i} (mean)	0.68	0.70	0.70	0.73	0.74	0.75	0.76
	±0.003	±0.006	±0.002	±0.003	±0.002	±0.002	±0.002
KGE^{ii} (median)	0.74	0.76	0.76	0.79	0.78	0.79	0.80
	±0.006	±0.005	±0.003	±0.005	±0.008	±0.005	±0.004
Pearson r^{iii} (median)	0.86	0.87	0.88	0.89	0.89	0.90	0.90
	±0.002	±0.002	±0.002	±0.001	± 0.001	±0.001	±0.001
α-NSE^{vi} (median)	0.83	0.86	0.86	0.88	0.85	0.87	0.88
	±0.010	±0.011	±0.008	±0.007	±0.007	±0.005	±0.008
β-NSE^{v} (median)	-0.03	-0.03	-0.03	-0.03	-0.03	-0.03	-0.02
	±0.005	±0.004	±0.004	± 0.004	± 0.004	±0.002	±0.004
FHV^{vi} (median)	-17.28	-13.89	-15.00	-12.52	-14.20	-13.15	-11.91
	±0.904	±1.217	±0.504	± 0.791	± 0.881	±0.450	±0.549
FLV^{vii} (median)	-0.88	2.83	0.05	-4.02	0.86	-1.54	2.57
	± 7.637	±5.403	±6.056	± 6.825	±5.499	±6.955	±4.072
FMS^{viii} (median)	-9.44	-7.31	-5.96	-5.60	-7.55	-6.93	-6.69
	±1.293	±1.500	± 1.234	±1.241	±1.358	± 0.911	± 1.678
Peak-Timing^{ix} (median)	0.46	0.49	0.46	0.44	0.42	0.41	0.41
	± 0.010	±0.009	±0.008	±0.007	± 0.007	± 0.009	± 0.015

^{i}: *Nash-Sutcliffe efficiency:* $(-\infty, 1]$*, values closer to one are desirable.*

^{ii}: *Kling-Gupta efficiency:* $(-\infty, 1]$*, values closer to one are desirable.*

^{iii}: *Pearson correlation:* $[-1, 1]$*, values closer to one are desirable.*

^{vi}: *α-NSE decomposition:* $(0, \infty)$*, values close to one are desirable.*

^{v}: *β-NSE decomposition:* $(-\infty, \infty)$*, values close to zero are desirable.*

^{vi}: *Top 2 % peak flow bias:* $(-\infty, \infty)$*, values close to zero are desirable.*

^{vii}: *30 % low flow bias:* $(-\infty, \infty)$*, values close to zero are desirable.*

^{viii}: *Bias of FDC midsegment slope:* $(-\infty, \infty)$*, values close to zero are desirable.*

^{ix}: *Lag of peak timing:* $(-\infty, \infty)$*, values close to zero are desirable.*

Appendix B: Average LSTM single model performance

Table C1. Average single LSTM performance across a range of different hydrological signatures. The derived metric for each signature is the Pearson correlation between the signature derived from the observed discharge vs. the signature derived from the simulated discharge. The average single model performances is then reported as the mean value of the the $n = 10$ model repetitions.

	NLDAS	Maurer	DayMet	Maurer + DayMet	NLDAS + Maurer	NLDAS + DayMet	All combined
Baseflow index	0.93	0.92	0.93	0.94	0.93	0.93	0.92
	±0.014	±0.018	±0.011	±0.005	±0.013	±0.009	±0.018
HFD mean	0.95	0.97	0.97	0.97	0.97	0.97	0.97
	±0.004	±0.003	±0.002	±0.002	±0.003	±0.003	± 0.004
High flow dur.	0.82	0.85	0.83	0.86	0.85	0.85	0.85
	±0.027	±0.014	±0.010	±0.014	±0.014	±0.008	± 0.014
High flow freq.	0.82	0.82	0.82	0.82	0.81	0.81	0.79
	±0.013	±0.014	±0.016	±0.016	±0.040	± 0.032	±0.037
Low flow dur.	0.44	0.42	0.46	0.47	0.43	0.46	0.45
	±0.033	±0.027	±0.025	±0.035	±0.018	±0.015	±0.039
Low flow freq.	0.83	0.82	0.84	0.86	0.82	0.84	0.83
	±0.020	±0.044	±0.028	±0.022	±0.027	±0.021	±0.043
Q5	0.95	0.95	0.96	0.96	0.95	0.96	0.96
	±0.005	±0.006	±0.003	±0.003	± 0.005	±0.005	±0.003
Q95	0.99	0.99	0.98	0.99	0.99	0.99	0.99
	±0.001	±0.001	±0.001	±0.001	± 0.000	±0.001	±0.000
Q mean	0.99	1.00	0.99	0.99	1.00	0.99	1.00
	±0.001	±0.000	±0.001	±0.000	±0.000	±0.000	±0.000
Runoff ratio	0.98	0.98	0.98	0.98	0.98	0.98	0.99
	±0.002	± 0.001	± 0.001	± 0.001	± 0.001	±0.001	± 0.001
Slope FDC	0.62	0.63	0.59	0.56	0.59	0.59	0.57
	±0.095	±0.053	± 0.093	± 0.053	± 0.061	±0.091	±0.096
Stream elasticity	0.61	0.69	0.70	0.70	0.68	0.69	0.71
	±0.015	±0.024	±0.017	±0.018	±0.025	±0.032	±0.021
Zero flow freq.	0.30	0.42	0.27	0.33	0.33	0.31	0.28
	±0.101	± 0.097	± 0.088	± 0.080	±0.067	± 0.086	±0.085

Appendix D: Peak flow timing

To evaluate the model performance on the peak timing we used the following procedure: First, we determined peaks in the observed runoff time series by locality search. That is, potential peaks are defined as local maxima. To reduce the number of peaks and filter out noise, the next step was an iterative process where, by pairwise comparison, only the maximum peak is kept until all peaks have at least a distance of 100 time steps to each other. The procedure is implemented in SciPy's find_peak function (Virtanen et al., 2020) and is used in the current work.

Second, we iterated over all peaks and searched for the corresponding peak in the simulated discharge time series. The simulated peak is defined as the highest discharge value inside of a window of ± 3 days around the observed peak. And, the peak timing error is the offset between the observed peak and the simulated peak. The resulting metric is the average offset over all peaks.

Appendix E: Analysis of precipitation uncertainty

The goal of this supplementary analysis was to understand the relationship between precipitation uncertainty and improvements to streamflow simulations due to using multiple forcing data sets. Because we don't have access to 'true' precipitation values in each catchment, we used triple collocation to estimate precipitation uncertainty. Triple collocation is a statistical technique to estimate error variances of three or more noisy measurement sources without knowing the true values of the measured quantities (Stoffelen, 1998; Scipal et al., 2010). Its major assumption is that the error models are linear and independent between sources; in particular, that all (three or more) measurement sources are each a combination of a scaled value of the 'true' variable plus additive random noise:

$$M_{i,t} = \alpha_i T_t + \varepsilon_{i,t}, \tag{E1}$$

where M_* are measurement values (i.e. here the modeled precipitation values), subscript i represents the source (DayMet, Maurer, NLDAS), and subscript t represents the time step in the test period (1 October 1989 to 30 September 1999); T_* is the unobserved true value of total precipitation in a given catchment on a given day; ε_* are i.i.d. measurement errors from any distribution.

The linearity assumption is not appropriate for precipitation data, which are typically assumed to have multiplicative error. Following Alemohammad et al. (2015), we assumed a multiplicative error model for all three precipitation source, and converted these to linear error models by working with the log-transformed precipitation data:

$$M_{i,t} = \alpha_i T_t^{\beta_i} + e^{\varepsilon_{i,t}} \tag{E2}$$

$$\ln(M_{i,t}) = \alpha_i + \beta_i T_t + \varepsilon_{i,t}. \tag{E3}$$

Standard triple collocation is then applied, so that estimates of the error variances for each source are:

$$\sigma_i = C_{i,i} - \frac{C_{i,j} C_{i,k}}{C_{j,k}}, \tag{E4}$$

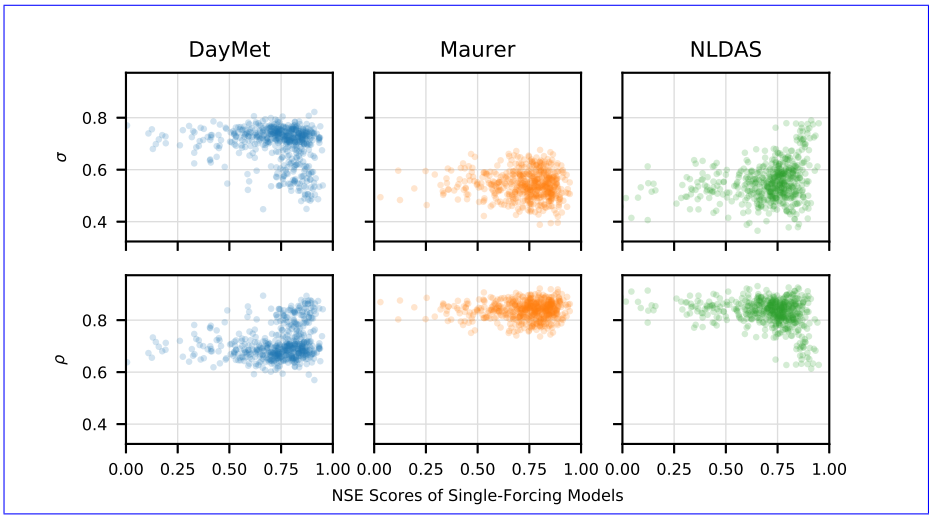


Figure E1. Triple collocation error variances and truth-correlations plotted against NSE scores of the single-forcing LSTM models. DayMet typically produces lower NSE values in basins where triple collocation reports that the precipitation error variances are high, whereas NLDAS produces lower NSE values in basins where triple collocation reports that the error variances are low. There is no apparent pattern in the Maurer data.

for all i, j, k, where $C_{i,j}$ is the covariance between the time series of source i and source j; σ_i is the variance of the distribution that each i.i.d. $\varepsilon_{i,t}$ is drawn from.

Additionally, extended triple collocation (McColl et al., 2014) allows us to derive the correlation coefficients between measurement sources and truth as:

$$\rho_i = \frac{C_{i,j}C_{i,k}}{C_{i,i}C_{j,k}}. \tag{E5}$$

This triple collocation analysis was applied separately in each of the 531 CAMELS catchments to obtain basin-specific estimates of the error variances, σ_i, and truth-correlations, ρ_i, for each of the three precipitation products. Albeit, the assumption that the forcing products have independent error structures (i.e. $\varepsilon_{i,t} \perp\!\!\!\perp \varepsilon_{j,t}$) is not met in our case we expect the results to be robust enough for the purpose at hand.

DayMet typically produced lower NSE values in basins where triple collocation reported that the DayMet precipitation error variances were high. This is what we would expect: low model skill in basins with high precipitation error. However, we did not see similar patterns with the other two precipitation products - see Fig. E1, where the triple collocation error variances and truth-correlation are plotted against the NSE scores of the single-source models. In fact, the NLDAS LSTM tended to perform worse in basins with lower precipitation error (as estimated by triple collocation).

One reason for this is shown in Figure E2, which is an adapted version of Fig. E1 that highlights a few high-skill, high-triple-collocation-v NLDAS basins in blue. These basins correspond to a cluster of basins in the Rocky Mountains (Fig. E3) where NLDAS has low correlation with the other two products but still yields high-skill LSTM simulations. What is happening here is that triple

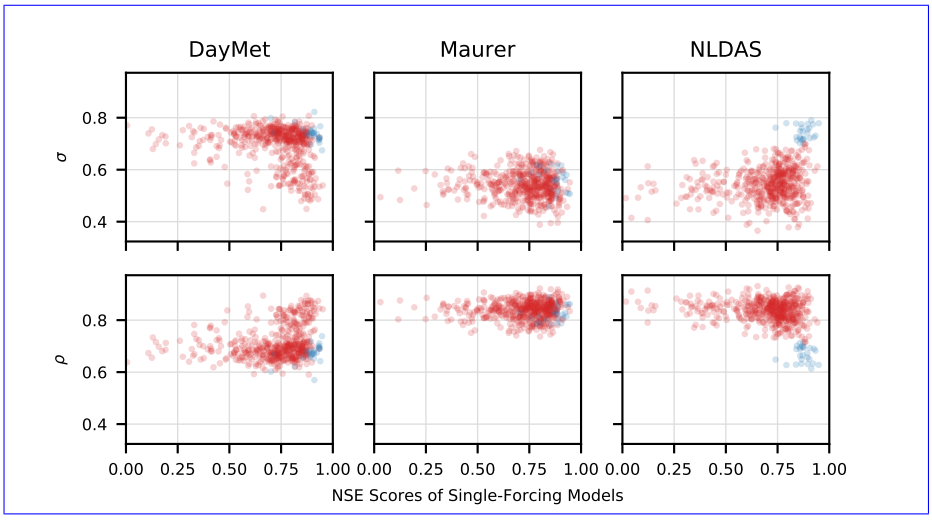


Figure E2. As in Fig. E1 the triple collocation error variances and truth-correlations are plotted against NSE scores of the single-forcing LSTM models. The coloring shows the anomalous NLDAS basins in blue and all others in red. For these basins NLDAS has low correlation with the other two products but still yields high-skill simulations.

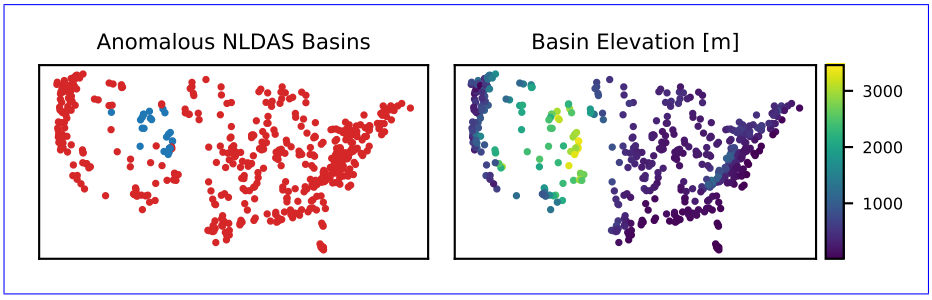


Figure E3. Spatial distribution of anomalous NLDAS basins shown in Fig. E2 (left) compared with elevation of the CAMELS basins (right).

collocation measures (dis)agreement between measurement sources, rather than error variances directly. Thus, the results in Figure E1 that appear to show NLDAS forcing models tending to perform well in basins with high precipitation error is driven in part by the fact that there are a few basins in the Rockies where NLDAS disagrees with, but is generally better than, the other two products. What Figure E1 is really showing is disagreement between precipitation estimates, and it is not necessarily the case that if one precipitation product disagrees with the others then this product contains more error. The LSTM is able to learn and account for this type of situation - it is not simply learning to trust one product over the others, and it is not simply learning to do something resembling a 'majority vote' in each basin.

Figure E4 plots model performance against the individual variances of the precipitation products in each basin. This figure shows that the single-forcing DayMet LSTM tended to perform better in catchments with higher total precipitation variance

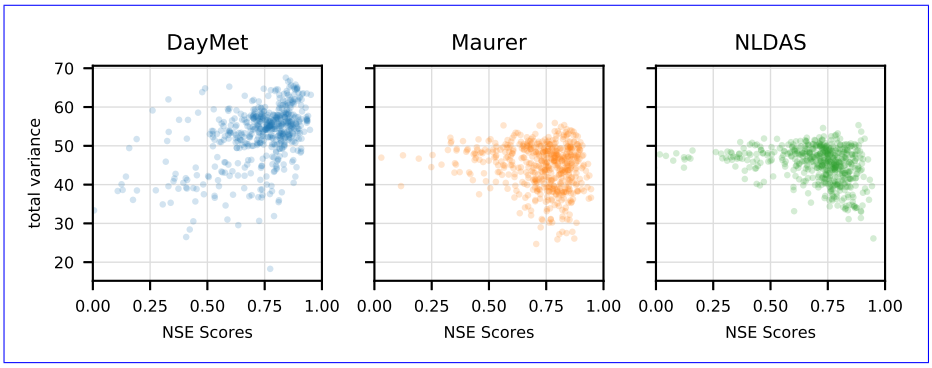


Figure E4. Performance of single-input models relative to the total variance of log-precipitation in each basin. The DayMet model tends to perform better in wetter basins (as the total DayMet variance increases), but the other two products have poor performing basins in catchments with high precipitation variance.

(not triple collocation error variance). This is again not true for the other two models, where higher total variance was associated with a higher variance in model skill, indicating higher proportion of the total variance is likely due to measurement error.

To analyse the synergy due to using all forcings in a single LSTM we transposed the NSE *improvements* in each basin (due to using all three forcing products in the same LSTM) with the log-determinant of the covariance matrix of all three (standardized, log-transformed) precipitation products (Fig. E5). The log-determinant is a proxy for the joint entropy of the three (standardized, log-transformed) products, and increases when there is larger disagreement between the three data sets. Unlike in Fig. E4, the variances in Fig. E5 were calculated after removing the mean and overall variance of each log-transformed precipitation product so that the log-determinant of the covariance is not affected by the overall magnitude of precipitation in each catchment (i.e., does not increase in wetter catchments). With the exception of the anomalous NLDAS basins, Fig. E5 shows that the three-forcing model offered improvements with respect to the single-forcing models when there was larger disagreement between the three data sets. This indicates that there is value in diversity among precipitation data sets, and that the LSTM can exploit this diversity.

Author contributions. FK had the idea for the training LSTMs on multiple forcing products. FK, DK, and GN designed all the experiments. FK trained the models and evaluated the results. GN did the triple collocation analysis, as well as the integrated gradients analysis. GN supervised the manuscript from the hydrological perspective and SH from the machine-learning perspective. GN and SH share the responsibility for the last authorship in the respective fields. All the authors worked on the manuscript.

Competing interests. The authors declare that they have no conflict of interest.

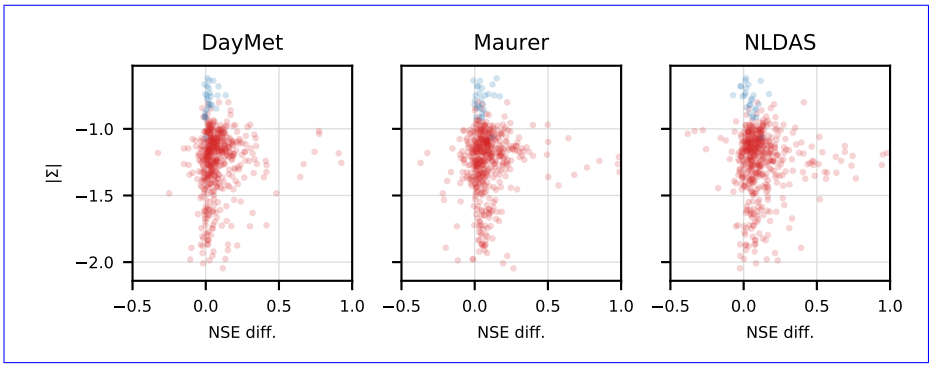


Figure E5. Fractional increase in NSE from the three-forcing model relative to the single-forcing models plotted against the log-determinant of the covariance matrix of all three (standardized, log-transformed) precipitation products. With the exception of the anomalous NLDAS basins (blue markers), the three-forcing model offers improvements with respect to the single-forcing models when there is larger disagreement between the three data sets. The three-forcing model learned to leverage synergy in these three precipitation products.

Acknowledgements. Authors from the Johannes Kepler University acknowledge support by Bosch, ZF, Google (Faculty Research Award), the NVIDIA Corporation with the GPU donations, LIT (grant no. LIT-2017-3-YOU-003) and FWF (grant no. P 28660-N31). Grey Nearing acknowledges support from the NASA Advanced Information Systems Technology program (award ID 80NSSC17K0541).

The project relies heavily on open source software. All programming was done in Python version 3.7 (van Rossum, 1995) and associated libraries including: Numpy (Van Der Walt et al., 2011), Pandas (McKinney, 2010), PyTorch (Paszke et al., 2017), SciPy (Virtanen et al., 2020), Matplotlib (Hunter, 2007)~~and xarray (Hoyer and Hamman, 2017)~~, xarray (Hoyer and Hamman, 2017), and Spotpy (Houska et al., 2019).

References

Addor, N., Newman, A. J., Mizukami, N., and Clark, M. P.: The CAMELS data set: Catchment attributes and meteorology for large-sample studies, Hydrology and Earth System Sciences, 21, 5293–5313, 2017a.

Addor, N., Newman, A. J., Mizukami, N., and Clark, M. P.: Catchment attributes for large-sample studies, Boulder, CO: UCAR/NCAR, https://doi.org/https://doi.org/10.5065/D6G73C3Q, 2017b.

Addor, N., Nearing, G., Prieto, C., Newman, A. J., Le Vine, N., and Clark, M. P.: A Ranking of Hydrological Signatures Based on Their Predictability in Space, Water Resources Research, 54, 8792–8812, https://doi.org/10.1029/2018WR022606, 2018.

Alemohammad, S. H., McColl, K. A., Konings, A. G., Entekhabi, D., and Stoffelen, A.: Characterization of precipitation product errors across the United States using multiplicative triple collocation, Hydrology and Earth System Sciences, 19, 3489–3503, https://doi.org/10.5194/hess-19-3489-2015, https://www.hydrol-earth-syst-sci.net/19/3489/2015/, 2015.

Anderson, E. A.: National Weather Service river forecast system: Snow accumulation and ablation model, NOAA Tech. Memo. NWS HYDRO-17, 87 pp., 1973.

Beck, H. E., Vergopolan, N., Pan, M., Levizzani, V., van Dijk, A. I. J. M., Weedon, G. P., Brocca, L., Pappenberger, F., Huffman, G. J., and Wood, E. F.: Global-scale evaluation of 22 precipitation datasets using gauge observations and hydrological modeling, Hydrology and Earth System Sciences, 21, 6201–6217, https://doi.org/10.5194/hess-21-6201-2017, https://www.hydrol-earth-syst-sci.net/21/6201/2017/, 2017.

Behnke, R., Vavrus, S., Allstadt, A., Albright, T., Thogmartin, W. E., and Radeloff, V. C.: Evaluation of downscaled, gridded climate data for the conterminous United States, Ecological applications, 26, 1338–1351, 2016.

Burnash, R.: The NWS river forecast system-catchment modeling, Computer models of watershed hydrology, 188, 311–366, 1995.

Burnash, R. J., Ferral, R. L., and McGuire, R. A.: A generalized streamflow simulation system, conceptual modeling for digital computers, Joint Federal and State River Forecast Center,U.S. National Weather Service, and California Departmentof Water Resources Tech. Rep., 204 pp., 1973.

Clark, M. P., Slater, A. G., Rupp, D. E., Woods, R. A., Vrugt, J. A., Gupta, H. V., Wagener, T., and Hay, L. E.: Framework for Understanding Structural Errors (FUSE): A modular framework to diagnose differences between hydrological models, Water Resour. Res., 44, 2008.

Clark, M. P., Wilby, R. L., Gutmann, E. D., Vano, J. A., Gangopadhyay, S., Wood, A. W., Fowler, H. J., Prudhomme, C., Arnold, J. R., and Brekke, L. D.: Characterizing uncertainty of the hydrologic impacts of climate change, Current Climate Change Reports, 2, 55–64, 2016.

Clausen, B. and Biggs, B.: Flow variables for ecological studies in temperate streams: groupings based on covariance, Journal of Hydrology, 237, 184–197, https://doi.org/10.1016/S0022-1694(00)00306-1, 2000.

Court, A.: Measures of streamflow timing, Journal of Geophysical Research, 67, 4335–4339, https://doi.org/10.1029/JZ067i011p04335, 1962.

Duan, Q., Ajami, N. K., Gao, X., and Sorooshian, S.: Multi-model ensemble hydrologic prediction using Bayesian model averaging, Advances in Water Resources, 30, 1371–1386, 2007.

Gers, F. A., Schmidhuber, J., and Cummins, F.: Learning to forget: Continual prediction with LSTM, 1999.

Gupta, H. V., Kling, H., Yilmaz, K. K., and Martinez, G. F.: Decomposition of the mean squared error and NSE performance criteria: Implications for improving hydrological modelling, Journal of hydrology, 377, 80–91, 2009.

Henn, B., Clark, M. P., Kavetski, D., and Lundquist, J. D.: Estimating mountain basin-mean precipitation fromstreamflow using Bayesian inference, Water Resour. Res., 51, 2008.

Henn, B., Newman, A. J., Livneh, B., Daly, C., and Lundquist, J. D.: An assessment of differences in gridded precipitation datasets in complex terrain, Journal of hydrology, 556, 1205–1219, 2018.

Hochreiter, S.: Untersuchungen zu dynamischen neuronalen Netzen, Diploma, Technische Universität München, 91, 1991.

Hochreiter, S. and Schmidhuber, J.: Flat minima, Neural Computation, 9, 1–42, 1997a.

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

Houska, T., Kraft, P., Chamorro-Chavez, A., and Breuer, L.: SPOTPY: A Python library for the calibration, sensitivity-and uncertainty analysis of Earth System Models., in: Geophysical Research Abstracts, vol. 21, 2019.

Hoyer, S. and Hamman, J.: xarray: N-D labeled arrays and datasets in Python, Journal of Open Research Software, 5, https://doi.org/10.5334/jors.148, http://doi.org/10.5334/jors.148, 2017.

Hunter, J. D.: Matplotlib: A 2D graphics environment, Computing In Science & Engineering, 9, 90–95, 2007.

Kingma, D. P. and Ba, J.: Adam: A method for stochastic optimization, arXiv preprint arXiv:1412.6980, 2014.

Kratzert, F., Klotz, D., Brenner, C., Schulz, K., and Herrnegger, M.: Rainfall–runoff modelling using Long Short-Term Memory (LSTM) networks, Hydrology and Earth System Sciences, 22, 6005–6022, 2018.

Kratzert, F., Klotz, D., Herrnegger, M., Sampson, A. K., Hochreiter, S., and Nearing, G. S.: Toward Improved Predictions in Ungauged Basins: Exploiting the Power of Machine Learning, Water Resources Research, 2019a.

Kratzert, F., Klotz, D., Shalev, G., Klambauer, G., Hochreiter, S., and Nearing, G.: Towards learning universal, regional, and local hydrological behaviors via machine learning applied to large-sample datasets., Hydrology & Earth System Sciences, 23, 2019b.

Kumar, R., Samaniego, L., and Attinger, S.: Implications of distributed hydrologic model parameterization on water fluxes at multiple scales and locations, Water Resources Research, 49, 360–379, 2013.

Ladson, A., Brown, R., Neal, B., and Nathan, R.: A standard approach to baseflow separation using the Lyne and Hollick filter, Australian Journal of Water Resources, 17, https://doi.org/10.7158/W12-028.2013.17.1, 2013.

Liang, X., Lettenmaier, D. P., Wood, E. F., and Burges, S. J.: A simple hydrologically based model of land surface water and energy fluxes for general circulation models, Journal of Geophysical Research: Atmospheres, 99, 14 415–14 428, 1994.

Lundquist, J., Hughes, M., Gutmann, E., and Kapnick, S.: Our skill in modeling mountain rain and snow is bypassing the skill of our observational networks, Bulletin of the American Meteorological Society, 2019.

Madadgar, S. and Moradkhani, H.: Improved B ayesian multimodeling: Integration of copulas and B ayesian model averaging, Water Resources Research, 50, 9586–9603, 2014.

Maurer, E. P., Wood, A., Adam, J., Lettenmaier, D. P., and Nijssen, B.: A long-term hydrologically based dataset of land surface fluxes and states for the conterminous United States, Journal of climate, 15, 3237–3251, 2002.

McColl, K. A., Vogelzang, J., Konings, A. G., Entekhabi, D., Piles, M., and Stoffelen, A.: Extended triple collocation: Estimating errors and correlation coefficients with respect to an unknown target: EXTENDED TRIPLE COLLOCATION, Geophysical Research Letters, 41, 6229–6236, https://doi.org/10.1002/2014GL061322, http://doi.wiley.com/10.1002/2014GL061322, 2014.

McKinney, W.: Data Structures for Statistical Computing in Python, Proceedings of the 9th Python in Science Conference, 1697900, 51–56, 2010.

Nash, J. E. and Sutcliffe, J. V.: River flow forecasting through conceptual models part I—A discussion of principles, Journal of hydrology, 10, 282–290, 1970.

Newman, A., Sampson, K., Clark, M., Bock, A., Viger, R., and Blodgett, D.: A large-sample watershed-scale hydrometeorological dataset for the contiguous USA, Boulder, CO: UCAR/NCAR, https://doi.org/https://dx.doi.org/10.5065/D6MW2F4D, 2014.

Newman, A. J., Mizukami, N., Clark, M. P., Wood, A. W., Nijssen, B., and Nearing, G.: Benchmarking of a physically based hydrologic model, Journal of Hydrometeorology, 18, 2215–2225, 2017.

Newman, A. J., Clark, M. P., Longman, R. J., and Giambelluca, T. W.: Methodological intercomparisons of station-based gridded meteorological products: Utility, limitations, and paths forward, Journal of Hydrometeorology, 20, 531–547, 2019.

Olden, J. D. and Poff, N. L.: Redundancy and the choice of hydrologic indices for characterizing streamflow regimes, River Research and Applications, 19, 101–121, https://doi.org/10.1002/rra.700, 2003.

Parkes, B., Higginbottom, T. P., Hufkens, K., Ceballos, F., Kramer, B., and Foster, T.: Weather dataset choice introduces uncertainty to estimates of crop yield responses to climate variability and change, Environmental Research Letters, 14, 124 089, 2019.

Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., and Lerer, A.: Automatic differentiation in PyTorch, 2017.

Pearl, J.: Embracing causality in default reasoning, Artificial Intelligence, 35, 259–271, 1988.

Samaniego, L., Kumar, R., and Attinger, S.: Multiscale parameter regionalization of a grid-based hydrologic model at the mesoscale, Water Resources Research, 46, 2010.

Sankarasubramanian, A., Vogel, R. M., and Limbrunner, J. F.: Climate elasticity of streamflow in the United States, Water Resources Research, 37, 1771–1781, https://doi.org/10.1029/2000WR900330, 2001.

Sawicz, K., Wagener, T., Sivapalan, M., Troch, P. A., and Carrillo, G.: Catchment classification: empirical analysis of hydrologic similarity based on catchment function in the eastern USA, Hydrology and Earth System Sciences, 15, 2895–2911, https://doi.org/10.5194/hess-15-2895-2011, 2011.

Scipal, K., Dorigo, W., and deJeu, R.: Triple collocation—A new tool to determine the error structure of global soil moisture products, in: 2010 IEEE International Geoscience and Remote Sensing Symposium, pp. 4426–4429, IEEE, 2010.

Seibert, J. and Vis, M. J. P.: Teaching hydrological modeling with a user-friendly catchment-runoff-model software package, Hydrology and Earth System Sciences, 16, 3315–3325, 2012.

Shrikumar, A., Greenside, P., Shcherbina, A., and Kundaje, A.: Not just a black box: Learning important features through propagating activation differences, arXiv preprint arXiv:1605.01713, 2016.

Stoffelen, A.: Toward the true near-surface wind speed: Error modeling and calibration using triple collocation, Journal of geophysical research: oceans, 103, 7755–7766, 1998.

Sundararajan, M., Taly, A., and Yan, Q.: Axiomatic attribution for deep networks, in: Proceedings of the 34th International Conference on Machine Learning-Volume 70, pp. 3319–3328, JMLR. org, 2017.

Sutton, R.: The bitter lesson, Incomplete Ideas (blog), March, 13, 2019.

Thornton, P. E., Running, S. W., White, M. A., et al.: Generating surfaces of daily meteorological variables over large regions of complex terrain, Journal of hydrology, 190, 214–251, 1997.

Timmermans, B., Wehner, M., Cooley, D., O'Brien, T., and Krishnan, H.: An evaluation of the consistency of extremes in gridded precipitation data sets, Climate dynamics, 52, 6651–6670, 2019.

Tolson, B. A. and Shoemaker, C. A.: Dynamically dimensioned search algorithm for computationally efficient watershed model calibration, Water Resources Research, 43, 2007.

Van Der Walt, S., Colbert, S. C., and Varoquaux, G.: The NumPy array: A structure for efficient numerical computation, Computing in Science and Engineering, 13, 22–30, 2011.

van Rossum, G.: Python tutorial, Technical Report CS-R9526, Tech. rep., Centrum voor Wiskunde en Informatica (CWI), Amsterdam, 1995.

Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., van der Walt, S. J., Brett, M., Wilson, J., Jarrod Millman, K., Mayorov, N., Nelson, A. R. J., Jones, E., Kern, R., Larson, E., Carey, C., Polat, İ., Feng, Y., Moore, E. W., Vand erPlas, J., Laxalde, D., Perktold, J., Cimrman, R., Henriksen, I., Quintero, E. A., Harris, C. R., Archibald, A. M., Ribeiro, A. H., Pedregosa, F., van Mulbregt, P., and Contributors, S. . .: SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python, Nature Methods, 17, 261–272, https://doi.org/https://doi.org/10.1038/s41592-019-0686-2, 2020.

Wellman, M. P. and Henrion, M.: Explaining'explaining away', IEEE Transactions on Pattern Analysis and Machine Intelligence, 15, 287–292, 1993.

Westerberg, I. K. and McMillan, H. K.: Uncertainty in hydrological signatures, Hydrology and Earth System Sciences, 19, 3951–3968, https://doi.org/10.5194/hess-19-3951-2015, 2015.

Xia, Y., Mitchell, K., Ek, M., Sheffield, J., Cosgrove, B., Wood, E., Luo, L., Alonge, C., Wei, H., Meng, J., et al.: Continental-scale water and energy flux analysis and validation for the North American Land Data Assimilation System project phase 2 (NLDAS-2): 1. Intercomparison and application of model products, Journal of Geophysical Research: Atmospheres, 117, 2012.

Yilmaz, K. K., Hogue, T. S., Hsu, K.-L., Sorooshian, S., Gupta, H. V., and Wagener, T.: Intercomparison of rain gauge, radar, and satellite-based precipitation estimates with emphasis on hydrologic forecasting, Journal of Hydrometeorology, 6, 497–517, 2005.

Yilmaz, K. K., Gupta, H. V., and Wagener, T.: A process-based diagnostic approach to model evaluation: Application to the NWS distributed hydrologic model, Water Resources Research, 44, 2008.