Reply to Referees' comments

Dear Editor and Referees,

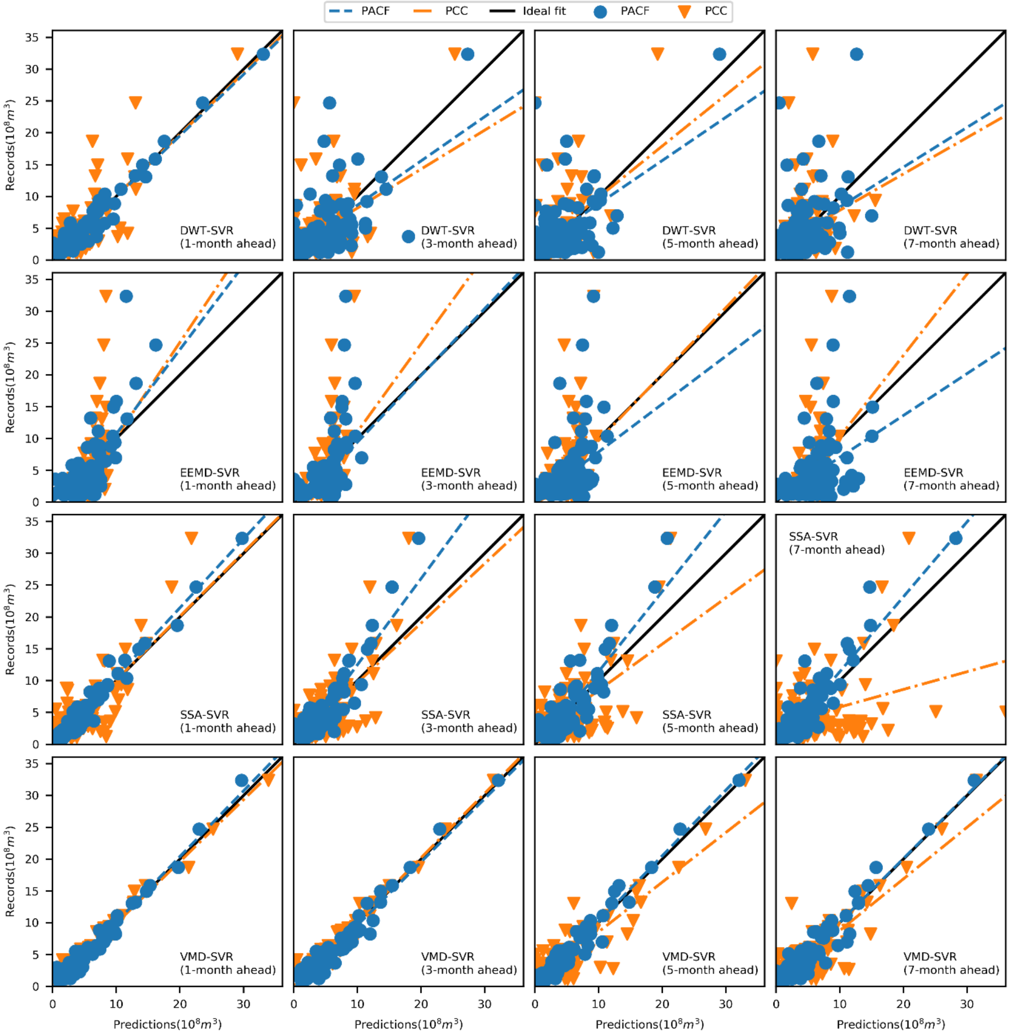
We thank the handling editor for handling the paper and the referees for their positive evaluation and for providing insightful and constructive comments, which have been a great help in improving the quality of our paper. We carefully revised the paper according to these comments and suggestions. The related parts of the paper have been rewritten and improved, and for your easy reading and evaluation, the changed parts are marked using BLUE COLORED text in the revised version.

We hope the revised version is to your satisfaction, and of course, we are more than happy to improve the paper again according to new comments and suggestions that might come.

Note that in some places of the manuscript, we have made improvements in addition to the referee comments. Please note that the line numbers in the referees' comments refer to the original version of the paper, while in our reply they refer to the revised version.

Main corrections:

1. *To make the manuscript concise and easy to read, the "Experiment 2: Performance evaluation of the TSDP models" and "Experiment 5: Evaluation of runoff forecasting for long leading times" in the original manuscript are combined as the "Experiment 4: Evaluation of the TSDP models for different lead times" in the revised manuscript (see Section 3.5.4).*
2. *In the original manuscript, the predictors for 3-, 5-, 7- and 9-month ahead runoff forecasting were determined by Pearson correlation coefficient (PCC). In the revised manuscript, the predictors of 3-, 5-, and 7-month ahead runoff forecasting were determined by partial autocorrelation coefficient function (PACF). Also, forecasting runoff 9 months ahead was removed. This is because determining the predictors by PACF has better forecasting performance than determining the predictors by PCC, which can be demonstrated by the following figure. Besides, forecasting runoff 3 months, 5 months, and 7 months ahead are enough to tell the difference of the forecasting models.*



3. *To make it easier for the reader to start reading the methodology, knowing where the research was developed, the "3.1 Study area and data observations" in the original manuscript were changed to "2 Monthly runoff data" in the revised manuscript.*
4. *We have added a flowchart to illustrate the optimization process of BOGP (see Figure 2).*
5. *We have added examples of the boundary effects to analyze the shift-copy variance and the data-addition sensitivity of VMD (see Figure 3).*
6. *We have computed kernel density estimates to evaluate the different error distribution of calibration and validation decompositions (see Figure 4).*
7. *We have computed absolute PCCs between validation sample predictors and prediction targets to indicate the importance of generating validation samples from appended decompositions (see Figure 5).*
8. *We have redrawn the block diagram of the TSDP framework to clarify the modeling process (see Figure 6).*
9. *We have added a block diagram of the comparative experimental setups to tell the difference of each experiment (see Figure 7).*
10. *We have added a block diagram of experiment 1 to denote the data partition of different forecasting schemes (see Figure 8).*
11. *We have added a table to describe the hyperparameter search ranges of ARIMA, SVR, BPNN, and LSTM(see Table 1).*
12. *We have evaluated the NSE of the BCMODWT-SVR scheme for different wavelet types and decomposition levels to find an optimal combination of wavelet and decomposition level (see Figure 12).*
13. *We have calculated the mutual information (MI) between each predictor and the predicted target to evaluate the feature importance for different lead times (see Figure 15).*
14. *For other corrections, please see the response details below.*

Referee 1 (John Quilty)

Comment by Referee:

ASSESSMENT:

This paper introduces a variational mode decomposition (VMD)-based support vector regression (SVR), i.e., VMD-SVR, model for multi-step ahead streamflow forecasting. The authors strive to address the 'boundary effects' problem common to many time series decomposition approaches that are typically coupled with data-driven models such as VMD, ensemble empirical mode decomposition (EEMD), singular spectrum analysis (SSA), wavelet transforms (WT), etc. outlined in earlier studies (Du et al., 2017; Maheswaran and Khosa, 2012; Quilty and Adamowski, 2018; Wang and Wu, 2016; Zhang et al., 2015). This is a worthwhile problem to address due to the growing interest in coupling these decomposition methods (VMD, EMD, SSA, WT, etc.) with data-driven models for hydrological forecasting and the vast majority of studies that overlook the impact of boundary effects on hydrological forecasting performance. Many of the just mentioned studies point out flaws in existing strategies for coupling decomposition methods with data-driven models and some go on to identify potential solutions.

In this paper, the authors put forth their own approach for addressing boundary effects. The authors claim that the main benefits of their proposed approach include that it "…can reduce the boundary effects, save the modelling time, and improve the prediction performance. This practical streamflow forecasting framework can be outlined as follows:

(1) Divide the entire streamflow data into training and validation sets and decompose each of these two sets separately into signal components. This procedure avoids using the validation information for training purposes.

(2) Combine the predictors of individual signal components into a final predictors, and select the original streamflow data as the prediction target in order to build only one optimized prediction model.

(3) Generate training and validation samples and divide the validation samples into development and testing samples. Mix and shuffle the training and the development samples to optimize the prediction model, and reduce the boundary effects."

Throughout the MAJOR COMMENTS section below, I raise several issues with how the authors' proposed approach actually satisfies these points. In my opinion, I think there is much clarification required on the authors' part to demonstrate that they adequately fulfill these points (in a way that is meaningful for operational forecasting problems, which the present study appears to be concerned with). In particular, the authors' methodology for how they decompose the time series using VMD (and other comparative approaches) and use it in training and validating their proposed VMD-SVR (and comparative) method(s) is not entirely clear. Out of all issues raised in this review, this point needs the most attention.

Nonetheless, I find that the paper is well-written, is properly structured, and is supported by appropriate figures. The references are sufficient. The analysis carried out in the paper is reasonable but the validity of the paper's results, in terms of how useful the results are to those concerned with developing operational forecasting models, largely depends on how the authors carried out decomposition of the time series (using VMD, DWT, etc.) and used it to develop the forecasting models.

If the authors can adequately address each of the comments/suggestions mentioned below, I would be happy to re-evaluate my current stance on the paper's suitability for being published in Hydrology and Earth System Sciences. In my opinion, the paper should not be published in its current form.

Answer by authors:

We thank you for the positive evaluation of our work and all the comments/suggestions on our manuscript. We have made improvements to the manuscript following your suggestions. Overall, the main corrections are:

1. *We have analyzed the VMD shift-variance and sensitivity of the addition of new data using the underlying data of our study (the monthly runoff at Huaxian station)(see Figure 3) and discussed how to reduce the influence of boundary effects caused by the VMD shift-variance and sensitivity of addition of new data (see Section 3.4).*
2. *We have compared the proposed TSDP framework with the WDDFF framework proposed by Quilty and Adamowski (2018). Additionally, we have compared no-decomposition ARIMA, SVR, BPNN, and LSTM with the TSDP and WDDFF framework in the revised manuscript.*

Please find more details about our changes below.

Comment by Referee:

MAJOR ITEMS:

In the Introduction, the authors mention how their "…proposed scheme can reduce the boundary effects, save the modelling time, and improve the prediction performance. This practical streamflow forecasting framework can be outlined as follows:

(1) Divide the entire streamflow data into training and validation sets and decompose each of these two sets separately into signal components. This procedure avoids using the validation information for training purposes.

(2) Combine the predictors of individual signal components into a final predictors, and select the original streamflow data as the prediction target in order to build only one optimized prediction model.

(3) Generate training and validation samples and divide the validation samples into development and testing samples. Mix and shuffle the training and the development samples to optimize the prediction model, and reduce the boundary effects."

Some comments on each of these points are given below:

Point 1

To decompose the validation data, the authors imply that they append one validation record at a time onto the calibration dataset (and any previous validation data) then perform VMD. The authors do this for each validation record, keeping the VMD components for each previous validation record static (this I my interpretation, the latter assertion was not specifically mentioned by the authors, at least that I could find). This appears to avoid boundary effects in the validation data due to the 'future data' issue; however, there are two major issues with this approach:

- The first issue is that VMD is sensitive to the addition of new data. I.e., by adding an additional data point to a time series and performing VMD creates inconsistencies between the intrinsic mode functions (IMFs) with the appended data and the IMFs prior to the appended data. Sometimes, these inconsistencies can be very large and tend to be largest at the edges of a time series, the most important time series observations in real-world forecasting applications (see example R script attached at the end of this review for the sensitivity of VMD to the additional of new data points).
- Because of this last point, the parameters of a model calibrated on initial IMFs generated by VMD that are then fed with updated IMFs based on the newly appended data may need to be updated to account for these newly introduced errors not seen during model calibration. This begs the question of whether each time the IMFs are updated whether the model should be updated too. Which goes against the authors' desire to implement a computationally-efficient forecasting method. Perhaps the authors may wish to consider a Kalman Filter to update their model parameters if they follow such an approach (as opposed to completely re-training the model). The Kalman Filter could be used to update the model parameters at each time step or at larger intervals.
- The second issue is that VMD is shift-variant, meaning that performing VMD on lagged versions of the same time series leads to distortions in the IMFs derived by the VMD at the same times. This further exacerbates the issue raised above in terms of calibrating and validating a data-driven model based on using time-lagged inputs that are decomposed via VMD (see example R script attached at the end of this review that demonstrates the shift-variance problem in VMD).

Answer by authors:

Thank you for providing the R script to test the shift-copy variance and the data-addition sensitivity of VMD. We have tested the shift-copy variance and data-addition sensitivity of VMD using a Matlab implementation which is derived from Dragomiretskiy and Zosso (2014). The results are shown in Figure 3. Note that the decomposition level (K), the quadratic penalty parameter (α), the noise tolerance (τ), and the convergence tolerance (ε) are the four parameters that influence the VMD decomposition performance (see the last paragraph of Section 3.1). The parameters α, τ, and ε remain static for decomposing the calibration set and appended sets. The parameter K was tuned based on the calibration set (see Section 4.4 and Figure 9 for how to tune K) and remain static for decomposing the appended sets (see Figure 6). The last decomposition of appended sets for each signal component is a validation decomposition.

We agree with you that VMD is shift-variant and sensitive to the addition of new data. We think the boundary effects cause these two issues. The boundary effects lead to large decomposition errors at the edges of a time series, but the rest decomposition errors are very small by adopting appropriate decomposition parameters, which can be proved by Figure 3. Fig. 3(b), (d), and (f) show that VMD has very small decomposition errors except for the boundary decompositions. Since the calibration set was concurrently decomposed and the validation set was sequentially appended to the calibration set and decomposed, the calibration decompositions (samples) are barely affected while the entire validation decompositions (samples) are affected by the *introduced decomposition errors. In other words, the calibration and validation samples have different error distribution (see Figure 4), which leads to the models calibrated on the calibration samples generalized poorly to validation samples.*

Since the calibration and validation samples can come from different distribution (see Machine learning yearning by Andrew Ng), we think it is not necessary to update model parameters with Kalman Filter because we have already dealt with the boundary effects with two different approaches. However, we do believe that updating model parameters with Kalman Filter is a new great idea to deal with boundary effects and we will research it in our subsequent experiments. One approach is to make the driving pattern (i.e., the relationship between predictors and prediction targets) in the validation sample as close as possible to that of the calibration sample. The other approach is to make the models assess validation error distribution during the calibration stage. In the first approach, the validation samples were generated from the decompositions of appended sets rather than validation decompositions because the driving pattern of boundary decomposition of appended sets is close to calibration decompositions (monotone increasing or decreasing, see minimap in Fig. 3a, c and e). In other words, the predictors selected from appended decomposition are more correlated to the predicted target (see Fig. 5). In the second approach, we mixed and shuffled the calibration and development (half validation) samples to build SVR models based on cross-validation. This is because the model can calibrate and validate on calibration and validation error distribution simultaneously. We have proved that these two approaches are worked by Experiment 1 (see Section 5.1 and Figure 16). The key ideas, potential reasons, and procedures of the TSDP framework have been clarified in the revised manuscript (see Section 3.4).

Comment by Referee:

Point 2

Although there are numerous studies that have considered forecasting each IMF (in EEMD, VMD, etc. based models) separately (and summing their constituent forecasts to obtain the final forecast), the authors should note that in the literature other studies have also built forecasts using, for instance, all wavelet-decomposed time series in a single forecasting model (Maheswaran and Khosa, 2013; Quilty and Adamowski, 2018). Many other examples of this approach can be found in the literature. It is suggested that the authors 'downplay' this feature of their framework as being something new or different.

Answer by authors:

Yes, you are right and we have downplayed "building a single forecasting model" as the new feature of the proposed TSDP framework and we also have added references for building a single forecasting model (see lines 289-291).

Comment by Referee:

Point 3

In terms of the mix and shuffle approach used to the training and validation data in the VMD-SVR models:

It is very difficult (for me) to see how taking all but the last 120 records of the red line in Figure 8 (b) (i.e., the development set) and randomly shuffling it with the red line from Figure 8 (a), (the training set) would lead to such a high performance on the last 120 records in Figure 8 (b) (i.e., the test set) as noted in Figure 9 and 11. Especially, when it appears that the training and combined development and test sets have completely different distributions (with the training set having a larger number of records).

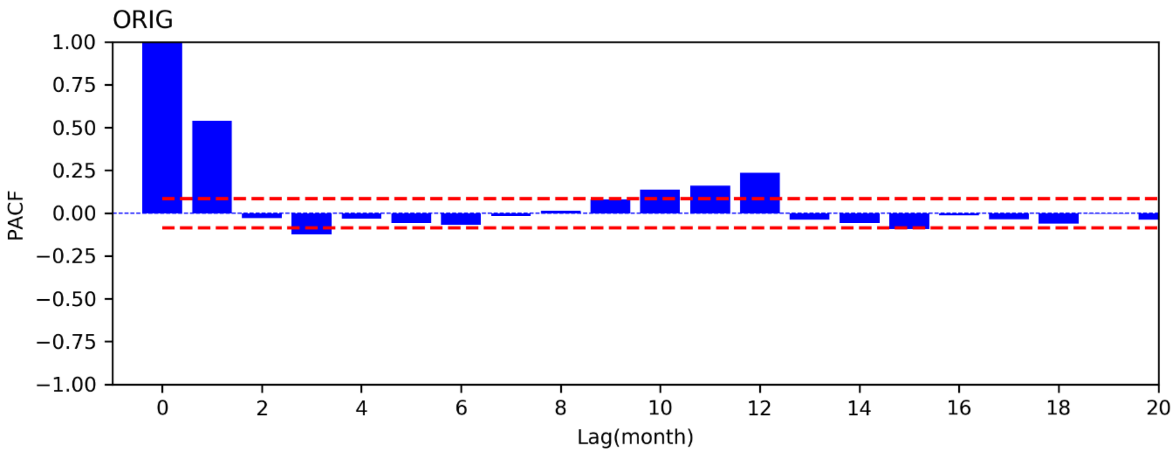
It seems like something is missing here. To have a mean NSE of 0.2 for standard SVR and a mean NSE of nearly 1 for SSA-SVR, VMD-SVR, and DWT-SVR (Figure 11) is a sign that something is potentially awry with the decomposition process and its division into training, development, and test sets. A modest increase in NSE would make sense (between SVR and VMD-SVR) but such a large discrepancy between the standard SVR and the coupled SVR approaches (SSA-SVR, VMD-SVR, and DWT-SVR) makes me think that important details of the decomposition of the time series and its partitioning into the different data sets is missing.

I think it is necessary for the authors (at the very least) to provide pseudo-code for how they decomposed the time series using EEMD, SSA, VMD, and DWT as well as how it was partitioned into training, development, and testing sets including how the mixed sampling approach was carried out. It would be ideal if the authors could provide the code that they used for these steps and (if possible) the time series used to develop the models. This would allow for the substantial difference in results (between SVR and VMD-SVR) to be validated.

Answer by authors:

Generating validation samples from the decompositions of appended sets (i.e., appended decompositions), which is performed before the mixing-and-shuffling step, helps a lot in improving the prediction performance (see Figure 16). This is because the predictors selected from appended decompositions are more correlated with the prediction targets than the predictors selected from validation decompositions (see Fig. 5). Of course, more development samples lead to better generalization performance but also lead to a greater workload. Set the number of development samples to 120 (accounts for about 22% of the calibration samples), is due to a balance between efficiency and accuracy. We have discussed why and how generating validation samples from appended decompositions and mixing and shuffling the calibration and development (half validation) samples improve the generalization ability of TSDP models (see Section 3.4).

In the original manuscript, the lag period of the SVR model is determined by PACF. For example, the lag equals to 1 for Huaxian station (as shown by the following figure). This is because the PACFs after the first lag are insignificant. This might lead to poor forecasting performance. We have tested that setting the lag to 12 leads to a better forecasting performance than lags from 1 to 11. Therefore, the lag period for SVR, BPNN, and LSTM is set to 12 in the revised manuscript. Additionally, we have clarified the decomposition process in Figure 6 and the data partition in Figure 8 and Section 2 (see lines 138-143).



We agree with you that providing code and data is helpful for the readers to validate our research results. The codes and data for this study were uploaded to GitHub (see https://github.com/zjy8006/MonthlyRunoffForecastByAutoReg.git) (see readme for how to reproduce our study) and we have also uploaded the updated code and data to Mendeley Data (http://dx.doi.org/10.17632/ybfvpgvvsj.4, Ganggang Zuo, 2020).

Comment by Referee:

Some other points include:

1. The authors seem to be familiar with the framework proposed by Quilty and Adamowski (2018) and note that these authors avoided boundary effects through their approach (Wavelet Data-Driven Forecasting Framework, WDDFF). Why did the authors not compare the WDDFF against their VMD approach? As it stands, each of the comparison methods (EEMD, SSA, and DWT) used in this study are all impacted by the boundary effect. At the end of this review, the Referee has included a MATLAB code for how the authors could obtain boundary-corrected wavelet and scaling coefficients through the maximal overlap discrete wavelet transform. If the authors use this script to decompose their input time series and include it in their SVR model, they can easily replicate the WDDFF from Quilty and Adamowski (2018).

Answer by authors:

We thank you for providing the Matlab implementation of boundary-corrected maximal overlap discrete wavelet transform (BCMODWT) to build WDDFF. We have compared the WDDFF framework with the TSDP framework. However, we think the WDDFF is not feasible for our case study. Because our underlying data does not include meteorological observation and we have to choose the explanatory variables from the monthly runoff. Additionally, our underlying data only contain 792 monthly runoffs, and remove the boundary-affected decompositions will lead to the sample size less than 792. In the revised manuscript, we have selected twelve *monthly runoff series lagging from one month to twelve months as explanatory variables and tested several wavelet functions (haar, db1, fk4, coif1, sym4, db5, coif2, and db10) and decomposition levels (1, 2, 3 and 4). The results are shown in Figure 12. However, the WDDFF framework based on BCMODWT and SVR, namely BCMODWT-SVR failed to predict our underlying data (see Fig.19 and Fig. 20). We have analyzed the potential reasons for these results in lines between 708 and 713.*

Comment by Referee:

2. Also, since the authors are following only a univariate time series forecasting problem without considering exogenous variables such as rainfall, evaporation, it would also seem plausible that they should compare their framework to simple time series methods such as ARIMA or even more appropriately, fractionally-differenced ARIMA (also known as Hurst-Kolmogorov processes, HKp) that are known to be suitable for forecasting time series with multiscale behaviour. The HKp method has been shown to be a useful method for monthly streamflow forecasting, with the potential to outperform common machine learning methods (k nearest neighbours, neural networks) (Koutsoyiannis et al., 2010).

Answer by authors:

We have added the ARIMA, BPNN, and LSTM models for predicting the underlying data of this study. The details of building these models are presented in Section 3.5.4 (see lines 393-398) and Table 1. The results are shown in Section 5.4, Figure 19, Figure 20, and discussed in Section 6 (see lines 714-716).

Comment by Referee:

3. Line 266-7: 'Therefore, only a few decomposition values of the training set are affected by the boundary effects.' Can the authors validate this claim (i.e., via a formula or through an experiment)? How can one determine which training records in the various VMD components include boundary effects?

Answer by authors:

That's a misstatement. We have confirmed that every decomposition of VMD will be affected by boundary effects through testing the shift-copy variance and the data-addition sensitivity of VMD (see Figure 3). What we try to explain with this sentence is that the decomposition errors except for the boundary decomposition errors can be ignored. This because these decomposition errors are very close to zero (see Fig.3b,d, and f). Since the calibration set was concurrently decomposed, the most calibration decomposition errors except for the boundary decomposition errors can be ignored. We have removed this sentence in the revised manuscript.

Comment by Referee:

4. VMD has many tuning parameters that, by the discussion in section 2.1, seems to greatly impact VMD performance. How then is VMD more user-friendly than the MODWT, which only requires the selection of a decomposition level and wavelet filter (although not trivial), for which there are only a finite number? From what I can tell, the parameters in VMD (aside from the selection of the number of IMFs) can take on an infinite number of values…

Answer by authors:

We agree with you that the VMD has more parameters that should be pre-assigned than MODWT. However, we think the VMD is more controllable than MODWT through the VMD parameters. There are four parameters, i.e., the decomposition level (K), the quadratic penalty parameter (α), the noise tolerance (τ), and the convergence tolerance (ε), mainly affect the decomposition performance of VMD. How these parameters affect the decomposition performance have been analyzed in Section 3.1 (see lines 164-170). As suggested by Zuo et al. (2020), the values of α, τ, and ε were set to 2000, 0, and 1e-9, respectively. Setting τ to 0 can remove noise in the original time series as much as possible and setting ε to 1e-9 can obtain more accurate decomposition results. Setting the α to 2000 tends to get small bandwidth, hence, avoid information redundancy and additional noise to be included in the decomposed signal components. As suggested by Xu et al. (2019), the decomposition results are very sensitive to the K. Therefore, we only have to tune K by observing the center-frequency aliasing (see Fig. 9) as suggested by Zuo et al. (2020). This can avoid mode mixing and extract more uncorrelated signal components with a low noise level (see Figure 13 and Figure 14). The VMD is more controllable than DWT or MODWT because we do not know how to select wavelet functions and decomposition levels to obtain uncorrelated signal components with a low noise level.

Comment by Referee:

5. I think Line 320 should be re-cast in light of the fact that selecting the right combination of VMD parameters is technically more computationally-intensive than for the DWT or MODWT.

Answer by authors:

We agree with you that tuning the all the four parameters (the decomposition level (K), the secondary penalty parameter (α), the noise tolerance (τ), and the convergence tolerance (ε)) is computationally-intensive than DWT or MODWT. However, we can control decomposition results in terms of mode mixing and noise with the VMD parameters. We only need to tune the most sensitive parameter, i.e., the decomposition level, which has been demonstrated worked in our case study. Therefore, we think VMD is not more computationally-intensive than DWT or MODWT because the most VMD parameters do not need to be tuned. We have removed "Testing numerous combinations of ψ and L is quite laborious." in the revised manuscript.

Comment by Referee:

6. For Experiment 5, only odd numbered lead times were considered (3, 5, 7, and 9 months ahead). Why were even numbered lead times (2, 4, etc. months ahead) not considered?

Answer by authors:

We aim to evaluate the performance gap of the TSDP models for long lead times and the workload for evaluating both the odd- and even-numbered lead times is huge. Therefore, we think only evaluate the odd-numbered (or even-numbered) lead times is enough to tell the difference of models. We can also evaluate 1-, 2-, 3-, 4-month ahead forecasting models, however, the 1-, 3-, 5-, 7- and 9-month ahead forecasting models can tell the predication performance of much longer lead times. In fact, we have removed the 9-month ahead monthly runoff forecasting in our revised manuscript for the convenience of comparison and presentation of different forecasting models.

Comment by Referee:

7. Section 3.4: which open-source software was used for EEMD, SSA, VMD, and DWT?

Answer by authors:

We have clarified the open-source software of EEMD, SSA, VMD, and DWT in Section 4.3.

Comment by Referee:

8. Given that a Bayesian approach (BOGP) was used for SVR hyper-parameter optimization, could it not also be used to select the VMD-related parameters? One would think that you could use the BOGP to optimize both VMD and SVR parameters at once. If possible, I think it would be interesting for the authors to consider this. If it is not feasible, a short discussion on why it is not feasible would be interesting.

Answer by authors:

Bayesian optimization is a sequential design strategy for global optimization of black-box functions. We did not search VMD parameters using BOGP because it is hard to define an objective function for VMD (for SVR the objective function is mean square error). Besides, only the decomposition level is needed to be tuned in the case study, and determining the decomposition level by observing the center-frequency aliasing (see Figure 9) can avoid mode mixing to obtain more uncorrelated signal components (see Figure 13) with a low noise level (see Figure 14).

Comment by Referee:

9. Why was six-fold cross-validation selected for hyper-parameter optimization (why not 3, 5, or 10-fold cross-validation)?

Answer by authors:

The CV fold is a vital parameter that influences the forecasting performance of TSDP models. However, there is no theoretical method to determine the CV fold. The 10-fold CV and leave-one-out CV (LOOCV) are two frequently-used methods (Zhang and Yang, 2015; Jung, 2018). The research results of Zhang and Yang (2015) indicated the LOOCV has a better performance than a 10-fold or 5-fold CV. However, LOOCV is computationally expensive. Additionally, Hastie et al. (2009) empirically demonstrated that 5-fold CV sometimes has lower variance than LOOCV. Therefore, the selection of cross-validation folds needs to consider specific application scenarios. We used the 6-fold CV in the previous version of the manuscript because we referred to an SVR model example. We have changed the 6-fold CV to the frequently-used 10-fold CV scheme in the revised manuscript rather than LOOCV due to the limited computational resources. Additionally, the difference between the 6-fold CV and 10-fold CV is small in the case of this study. We have clarified why using 10-fold CV in lines 476 between 482.

Comment by Referee:

10. Normally one has to set a range for the different hyper-parameters in the BOGP approach. What range was set for the various SVR hyper-parameters? It would be good to include what guided your selection of these particular ranges.

Answer by authors:

In the revised manuscript, we have clarified the search range for the parameters of SVR in Table 1.

Comment by Referee:

11. Line 495: Figure 8 (a) – I find it hard to agree with the statement that the training data is ‘barely affected by the boundaries’. Between record 550 and 555 there is a difference between the red and blue lines of ~ 2 *108 m3! I think one can hardly dismiss this as being a small difference…I suggest acknowledging this rather large discrepancy as something significant.

In fact, this is one of the issues of VMD, EEMD, etc. They are not shift-invariant and are sensitive to the addition of new observations (see supporting R code at the end of this review). I suggest the authors discuss in detail these disadvantage of VMD, especially in relation to the MODWT, which does not suffer from these problems and which may also be used, in a mathematically sound manner, to decompose multiscale and/or non-stationary time series into sub-time series capturing their prominent features (which potentially includes trends, periodicities, transients, etc.). I think more effort needs to be devoted to clearly identifying the particular advantages of using the VMD-based approach in this study over the MODWT (which again, does not suffer from such issues).

Answer by authors:

That's a misstatement. We have removed this statement in the revised manuscript. We agree with you that the calibration data is affected by boundary effects. However, this statement means that the most decomposition errors except for the boundary decomposition errors caused by boundary effects can be ignored. Although the boundary decomposition errors are large, only a small number of calibration samples are affected by these boundary decomposition errors. We agree with you that VMD, EEMD, DWT, and SSA suffer form the shift-variance and sensitive to the addition of new data, which lead to decomposition errors. The boundary-corrected MODWT can avoid this problem. However, this study aims to propose a general solution to this problem by using different approaches. In other words, we think building practical forecasting models using VMD, EEMD, DWT and SSA without correcting and removing the boundary decompositions is worth trying. We have analyzed the particular advantages and disadvantages of using the VMD-based approach in this study in Section 6 (see lines 690-719).

Comment by Referee:

12. Figure 8(b) drives the above-mentioned point home much further… Comparing Figure 8 (a) and Figure 8 (b) it also appears to be the case that the validation data and training data come from distributions, too. It would seem logical that the forecasting model should be updated to account for this change through time (e.g., perhaps through a Kalman Filter)?

Answer by authors:

As shown in Fig. 3(f), only a small number of decompositions at the edges of the calibration set suffers from boundary effect. As shown in Fig.3(g), all the validation decompositions are suffering from boundary effects. Therefore, we think the calibration decompositions and validation decomposition have a different (error) distribution. Since the calibration and validation set can have different distribution (see Machine learning yearning by Andrew Ng). We think it is not necessary to update the model parameters because generating validation samples from appended decompositions and mixing and shuffling the calibration samples and development samples (half validation samples) help a lot to improve the generalization ability of the calibrated models (see Figure 16). Generating validation samples from appended decompositions obtain more correlated input predictors (see Figure 5) and mixing and shuffling the calibration and development samples can assess the validation error distribution during the calibration stage.

Comment by Referee:

13. Figure 10: it would make it much easier to read if the authors reduced the marker size for the different methods in the scatter plots. For the hydrograph plots, it would be good to zoom in on a particular section, perhaps concentrating on the largest peak event?

Answer by authors:

We have redrawn the scatter plots to make it easier to read, see Figure 19.

Comment by Referee:

14. Figure 18: Why was the standard SVR not included in this analysis? I think it should be included to show how much better the other approaches (DWT-SVR, VMD, SVR, etc.) are at longer lead times.

Answer by authors:

We have analyzed the standard ARIMA, SVR, BPNN, and LSTM model in the revised manuscript, see Figure 20 and Section 5.4.

Comment by Referee:

15. Line 755-756: ‘However, as far as we know, approaches of building a forecasting framework that is adapted to the boundary effect never be tried.’

    Are you sure? Quilty and Adamowski (2018) explored the boundary effect existing in popular wavelet-based decomposition methods (DWT, MODWT, etc.), then introduced a set of best practices that addresses these boundary conditions (completely) and implemented these best practices in a new forecasting framework tailored for real-world forecasting. I think one could say that their framework ‘adapted to the boundary effect’. In an earlier study by Maheswaran and Khosa (2012), the authors also discussed how to overcome some of the issues of the DWT by choosing a more appropriate wavelet decomposition method (à trous algorithm) that did not suffer from the same boundary conditions. I would also qualify their approach as ‘adapting to the boundary effect’. In my opinion, the texted quoted from Line 755-6 is not entirely true and should be revised.

Answer by authors:

We have clarified this sentence. What we want to emphasize by this sentence is that the approaches, which are adapted to the boundary effect without correcting and removing the boundary-affected decompositions and providing users with a high confidence level on the unused data, never be tried. See lines 723-725.

Comment by Referee:

MINOR ITEMS:

- There are numerous grammatical and spelling errors. I did not note all of these issues. I recommend that the authors carefully check the paper for grammatical and spelling issues (e.g., Line 673 '…increase and decrease patterns…' should read '…increasing and decreasing patterns…').

Answer by authors:

We have carefully revised the manuscript and check it for clarity and language. Additionally, our revised manuscript was edited for proper English language, grammar, punctuation, spelling, and overall style by one or more of the highly qualified native English speaking editors at Editideas. The editorial certificate is shown as follows.

辑思编译
EDITIDEAS

EDITORIAL CERTIFICATE

This document certifies that the manuscript listed below was edited for proper English language, grammar, punctuation, spelling, and overall style by one or more of the highly qualified native English speaking editors at Editideas.

Manuscript title:

Two-stage Variational Mode Decomposition and Support Vector Regression for Streamflow Forecasting

Authors:

Ganggang Zuo, Jungang Luo, Ni Wang, Yani Lian, Xinxin He

Date Issued:

2020-8-21

Certificate Number:

E-202004120011

Editing Certified · Editideas · Edit Ideas · Editing Certified

This document certifies that the manuscript listed above was edited for proper English language, grammar, punctuation, spelling, and overall style by one or more of the highly qualified native English speaking editors at Editideas. Neither the research content nor the authors' intentions were altered in any way during the editing process. Documents receiving this certification should be English-ready for publication; however, the author has the ability to accept or reject our suggestions and changes.

Editideas provides a range of editing, translation and manuscript services for researchers and publishers around the world. Our top-quality PhD editors are all native English speakers from famous institutions cross the U.S.. Our editors come from nearly every research field and possess the highest qualifications to edit research manuscripts written by non-native English speakers.

Comment by Referee:

- Line 709: ‘Orthometric’? I suggest trying to get your point across using different terms.

Answer by authors:

We have clarified this term. The ‘Orthometric’ term means uncorrelated signal components. See line 380, 387, 510, 675, 693.

Comment by Referee:

- Line 761: simulating or forecasting? This applies to the whole paragraph. Simulation and forecasting are generally regarded as different procedures in hydrology.

Answer by authors:

We have changed “simulating” to “forecasting” (see line 732) and rephrased this paragraph in the revised manuscript.

Comment by Referee:

- Line 764: It is not clear what is meant by ‘…predictor-runoff relationship and the decomposition-runoff relationship’.

Answer by authors:

We have clarified this sentence. We mean the relationship between input predictors and the output target, and the relationship between the original signal and decomposed signal components. See lines 665-666.

Comment by Referee:

- Line 765: I think you mean accuracy instead of reliability (the latter is generally measured using probabilistic performance metrics). The same comment also applies to the sentence two lines below.

Answer by authors:

You are right, we have clarified this in the revised manuscript. See line 15, 110, 321.

Comment by Referee:

- Line 773: ‘lead’ not ‘leading’.

Answer by authors:

We have revised this term, see line 17, 112, 304, 324, 389, 531, 595, 609, 618, 627, 628, 635, 737, and 742.

Comment by Referee:

- Line 782: I would rephrase point ‘c’. Perhaps mention something along lines ‘Although some overfitting of the VMD-SVR occurs, the model still provides accurate out-of-sample forecasts’.

Answer by authors:

Thanks. We have rephrased point ‘c’ (see lines 748-750).

Comment by Referee:

- Line 789-90: Such as…? It would be good to provide some ideas concerning how you think this can be realized.

Answer by authors:

Such as using interpretable models (e.g., decision trees) to analyze feature (predictor) importance, using partial dependence plots to observing the global or local convergence, visualizing the model structure and parameters to analyze how the model structure and parameters influence the prediction results, generating against data to test the model behavior.

Reference

Du, K., Zhao, Y., Lei, J., 2017. The incorrect usage of singular spectral analysis and discrete wavelet transform in hybrid models to predict hydrological time series. J. Hydrol. 552, 44–51. doi:10.1016/j.jhydrol.2017.06.019

Koutsoyiannis, D., Yao, H., Georgakakos, A., 2010. Medium-range flow prediction for the Nile: a comparison of stochastic and deterministic methods. Hydrol. Sci. J. 53, 142–164. doi:10.1623/hysj.53.1.142

Maheswaran, R., Khosa, R., 2013. Wavelets-based non-linear model for real-time daily flow forecasting in Krishna River. J. Hydroinformatics 15, 1022. doi:10.2166/hydro.2013.135

Maheswaran, R., Khosa, R., 2012. Comparative study of different wavelets for hydrologic forecasting. Comput. Geosci. 46, 284–295. doi:10.1016/j.cageo.2011.12.015

Quilty, J., Adamowski, J., 2018. Addressing the incorrect usage of wavelet-based hydrological and water resources forecasting models for real-world applications with best practices and a new forecasting framework. J. Hydrol. 563, 336–353. doi:https://doi.org/10.1016/j.jhydrol.2018.05.003

Wang, Y., Wu, L., 2016. On practical challenges of decomposition-based hybrid forecasting algorithms for wind speed and solar irradiation. Energy 112, 208–220. doi:10.1016/j.energy.2016.06.075

Zhang, X., Peng, Y., Zhang, C., Wang, B., 2015. Are hybrid models integrated with data preprocessing techniques suitable for monthly streamflow forecasting? Some experiment evidences. J. Hydrol. 530, 137–152. doi:10.1016/j.jhydrol.2015.09.047

Dragomiretskiy, K. and Zosso, D.: Variational Mode Decomposition, IEEE Trans. Signal Process., 62, 531–544, doi:10.1109/TSP.2013.2288675, 2014.

Zuo, G., Luo, J., Wang, N., Lian, Y., and He, X.: Decomposition ensemble model based on variational mode decomposition and long short-term memory for streamflow forecasting, Journal of Hydrology, 585, 124776, doi:10.1016/j.jhydrol.2020.124776, 2020.

Zhang, Y. and Yang, Y.: Cross-validation for selecting a model selection procedure, Journal of Econometrics, 187, 95–112, doi:10.1016/j.jeconom.2015.02.006, 2015.

Jung, Y.: Multiple predicting K -fold cross-validation for model selection, Journal of Nonparametric Statistics, 30, 197–215, doi:10.1080/10485252.2017.1404598, 2018.

Hastie, T., Friedman, J., and Tibshirani, R.: The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Second, Springer Series in Statistics, Springer-Verlag New York, New York, NY, Online-Ressource, 2009.

Referee 2

Comment by Referee:

General comments: Streamflow forecasting is important for water management and optimal allocation of water resources. This study aimed to improve the model performance of decomposition based forecasting methods. A two stage decomposition predication framework (TSDP) was proposed by the authors based on VMD and SVR, to avoid the influences of validation information on training. The effectiveness, efficiency and reliability of the TSDP framework and its VMD-SVR realization in terms of the boundary effect reduction, decomposition performance, prediction outcomes, time consumption, overfitting, and forecasting capability for long leading times were investigated. The final results on monthly runoff from three stations at the Wei River showed the superiority of the TSDP framework compared to benchmark models. It is found that the results are interesting for guiding proper use of decomposition-based forecasting methods in streamflow forecasting practice.

Answer by authors:

Thank you for your detailed evaluation of our manuscript. We have improved the manuscript according to your suggestions. Please find our response as to how each comment has been taken into consideration in the making revision below.

Comment by Referee:

Specific comments: 1) This study only focused on decomposition-based methods and aimed to solve one disadvantage existing in applying decomposition methods. Although this might be interesting for readers who use decomposition based methods, a wider scope including more streamflow forecasting techniques like ARIMA, BP, LSTM etc. can be more interesting. Even if a new technique is proposed (not the case in this manuscript), a companion with different types of techniques is often needed to support the application of the proposed technique.

Answer by authors:

We have analyzed the standard ARIMA, SVR, BPNN, and LSTM model in the revised manuscript, see Section 3.5.4 and Table 1 for experimental setups, see Figure 19, Figure 20, and Section 5.4 for experimental results and see lines 714-716 for discussion.

Comment by Referee:

2) Five experiments were designed for the assessment of different performance aspects including the reduction of the boundary effects, decomposition performance, predictability, time consumption, overfitting, and forecasting capabilities for long leading times. This might be interesting for readers. However, it is difficult to understand these experiments, since the complicated five-experiment design and presentation styles stopped the successful understanding and digestion of the results. I suggest the authors rewrite this part and add tables (for a comparison of five experiments) to help readers better understand the six different experiments and their differences.

Answer by authors:

To make the manuscript concise and easy to read, the "Experiment 2: Performance evaluation of the TSDP models" and "Experiment 5: Evaluation of runoff forecasting for long leading times" in the original manuscript are combined as the "Experiment 4: Evaluation of the TSDP models for different lead times" in the revised manuscript (see Section 3.5.4). We have rewritten this part (see Section 3.5) and added a block diagram of the comparative experimental setups (see Figure 7).

Comment by Referee:

3) Lines 66-67: when you mentioned the boundary effect for the first time in the manuscript, I expect an explanation of the 'boundary effect'.

Answer by authors:

We have explained the boundary effect the first time it is mentioned (see lines 62-64).

Comment by Referee:

4) VMD and SVR are well-known techniques. The authors can shorten the descriptions of these two techniques and focus on the new things the authors proposed.

Answer by authors:

We have shortened the descriptions of VMD and SVR in our revised manuscript (see Section 3.1 and Section 3.2).

Comment by Referee:

5) Line 81: change 'usage' to 'use'

Answer by authors:

We have rephrased this term in the revised manuscript (see lines 77-80).

Comment by Referee:

6) Line 259ïijŽ what is BOGP? Do you mean 'Bayesian optimization based on Gaussian processes'? How is BOGP used to optimize EEMD, SSA, DWT and SVR? Add some details.

Answer by authors:

Yes, the BOGP means Bayesian optimization based Gaussian processes. We have clarified this term in the nomenclature. The BOGP was only used to optimize SVR-, BPNN-, and LSTM-based models. The BOGP was not used to optimize the parameters of EEMD, SSA, DWT and VMD because (1) it is hard to define an objective function for the decomposition processes (the object function for SVR-, BPNN-, and LSTM-based models is mean square error) and (2) we can manually control the decomposition performance by setting specific parameters (e.g., set the noise tolerance of VMD to zero to obtain sub-signals with low noise level). We have added more details about how to use the BOGP to optimize the SVR-, BPNN-, and LSTM-based models in the revised manuscript (see Figure 2). Below we give a brief explanation of using BOGP to optimize SVR models.

In this study, the BOGP was used to obtain the optimized hyperparameters of SVR, i.e., the weight penalty(C), the error tolerance (ε), and the width control coefficient (σ). The BOGP algorithm for SVR model can be wrapped up as follows:

Setp 1. Input a set of mixed and shuffled samples, the object function(f),i.e., the loss function(the mean square error was used in this study), the convergence error (e.g., $E = 1e-6$) and the number of iterations (e.g., $N_C = 100$), and the hyperparameters search space (e.g., $C = [0.1,200]$, $\varepsilon = [1e-6,1]$, $\sigma = [1e-6,1]$).

Setp 2. Randomly sample a candidate (e.g., $\boldsymbol{x_0} = [C = 25, \varepsilon = 0.00001, \sigma = 0.26]$) based on the given search space and set the iteration index as $i = 1$.

Setp 3. Given the previous candidate, update the posterior expectation of f using the Gaussian process model (see Eq. 10).

Setp 4. Track the new candidate ($\boldsymbol{x_i}$) that maximize the expected improvement (EI) function (see Eq. 11), i.e., $\boldsymbol{x_i} = \arg\max EI(\boldsymbol{x})$.

Setp 5. Compute $f(\boldsymbol{x_i})$ based on the mixed and shuffled samples (including predictors and predicted targets) and set the iteration index to $i = i + 1$.

Setp 6. Repeat steps 3-5 until the convergence is achieved or the number of iterations is reached.

Setp 7. Output the last candidate as the optimal hyperparameters of the SVR model.

Comment by Referee:

7) Add a table for a clear comparison of five experiments

Answer by authors:

We have added a block diagram to compare the four experiments (see Figure 7). Since the previous experiments represent the baseline for the next ones, we did not add a table for the comparison of the experimental results of the four experiments.

Comment by Referee:

8) 4 ‘Experimental Results and Analysis’ should be ‘Experimental results’

Answer by authors:

We have changed ‘Experimental Results and Analysis’ to ‘Experimental Results’ (see Section 5).

Comment by Referee:

9) Line 354: Why 3,5,7,9?

Answer by authors:

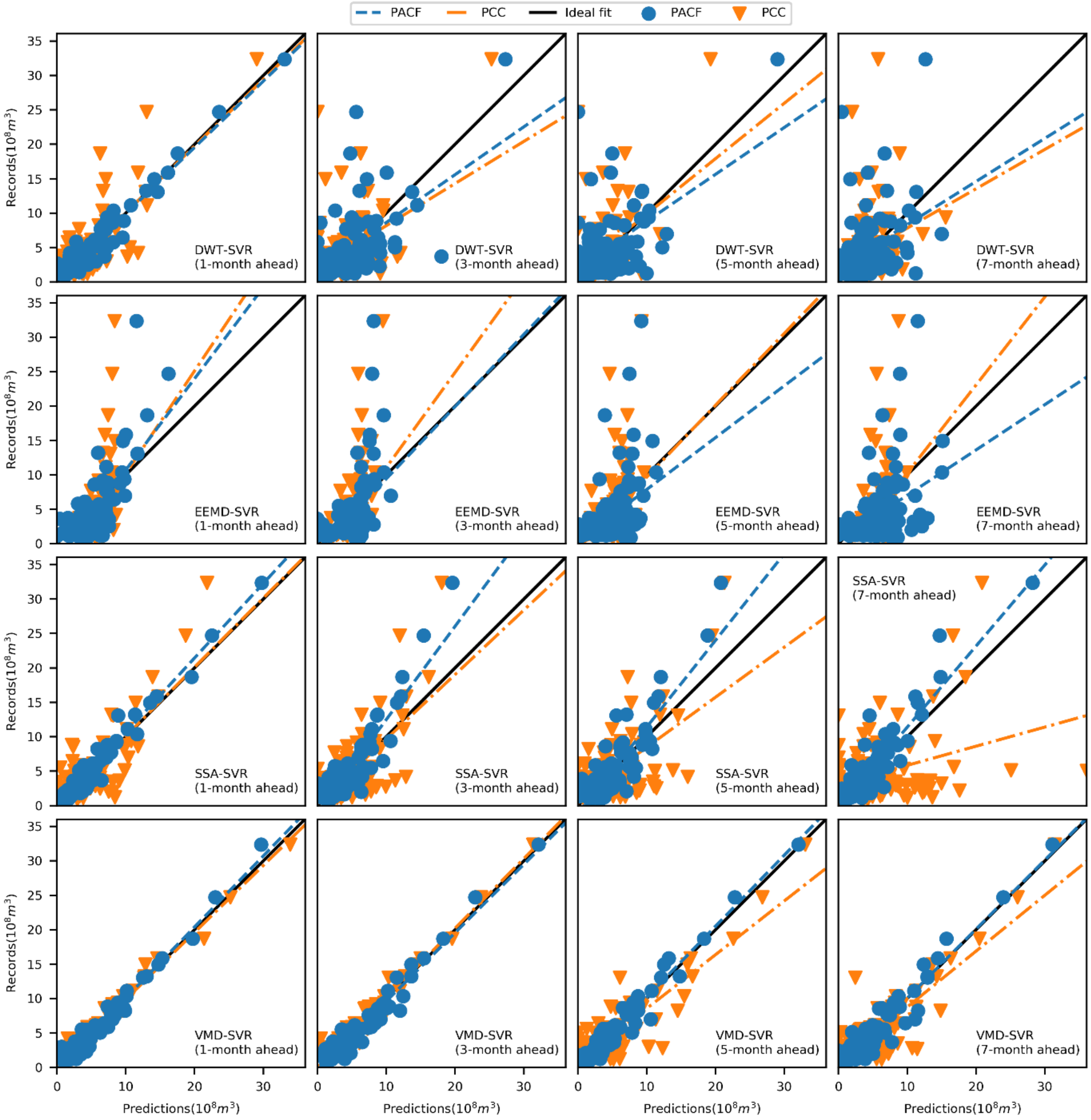
We aim to evaluate the performance gap of the TSDP models for long lead times and the workload for evaluating both the odd- and even-numbered lead times is huge. Therefore, we think only evaluate the odd-numbered (or even-numbered) lead times is enough to tell the difference of models. We can also evaluate 1-, 2-, 3-, 4-month ahead forecasting models, however, the 1-, 3-, 5-, 7- and 9-month ahead forecasting models can tell the predication performance of much longer lead times. In fact, we have removed the 9-month ahead monthly runoff forecasting in our revised manuscript for the convenience of comparison and presentation of different forecasting models.

Comment by Referee:

10) Line 356: What does that mean by ‘the 20-month lag’? Does that make sense for monthly forecast?

Answer by authors:

The '20-month lag' is the upper limit of lags for computing the Pearson correlation coefficient (PCC). The 20-month lags (i.e., 1-month lag, 2-month lag, ...) were used to compute the PCC, and the lags with higher absolute PCC were finally selected as input predictors. We set a 20-month lag as the upper limit is due to the maximum lags of Partial autocorrelation function(PACF) was also set to 20. Therefore, we can compare the prediction performance of models established using the input predictors determined by PCC and PACF. Note that in the revised manuscript, we chose PACF instead of PCC to determine the input predictors for 3-, 5- and 7-month ahead runoff forecasting. This is because PACF leads to better forecasting performance than PCC (see the following figure).



Comment by Referee:

11) Figure 2: if possible, put a map of China

Answer by authors:

We have added a map of China to illustrate the location of the Wei River (see Figure 1).

Comment by Referee:

12) I didn't really get how 'the mixing and shuffling step' works. If possible, please clarify.

Answer by authors:

The mixing-and-shuffling step first mixes (concatenates) the calibration samples(e.g., row indexes are 1,2,3,4,5) and development samples (i.e., half validation samples)(e.g., row index are 6,7,8,9) as a single set of samples and then randomly shuffles the mixed sample rows (e.g., original row indexes are 1,2,3,4,5,6,7,8,9,..., and the shuffled row indexes might be 6,3,9,1,5,2,7,4,8,...).

In fact, two crucial steps help to reduce the influences of the boundary effect. One is generating validation samples from appended decompositions and the other is mixing and shuffling the calibration and development samples. In the current manuscript, we only have discussed the later one. In the revised manuscript, we have added a discussion about the former one and clarify how these steps deal with the boundary effect (see Section 3.4).

The different error distribution of calibration and validation decompositions (see Figure 4), which is caused by the boundary effect, leads to the models calibrated on the calibration samples generalize poorly to the validation samples.

The aforementioned two steps are worked for reducing the influence of boundary effect because : (1) The relationship between input predictors selected from appended decompositions and output target is maintained by the decomposition algorithms. In other words, the predictors can be reconstructed to original monthly runoff values by the decomposition algorithm. However, due to the decomposition errors come from different sets of appended decompositions, the predictors selected from the validation decompositions cannot be reconstructed to original monthly runoff values. This leads to the absolute Pearson correlation coefficients (PCCs) between predictors and predicted targets of validation samples generated from appended decompositions are larger than that of validation samples generated from validation decompositions (see Figure 5); (2) Mixing and shuffling the calibration and development samples, and training the models based on the shuffled samples enable the models to calibrate and validate on the calibration and validation error distribution simultaneously. The TSDP models were established based on the mixed and shuffled samples using cross-validation (CV) strategy (e.g.,10-fold CV means the mixed and shuffled samples are divided into 10 sub-samples, of which each one will be used to calibrate and validate the TSDP models). Shuffling the mixed samples enables the validation samples to be *randomly distributed throughout the mixed samples, which means the sub-samples extracted from the mixed and shuffled samples can be used to calibrate and validate the calibration and validation error distribution simultaneously. Therefore, the mixing-and-shuffling step can improve the generalization ability.*

The results (see Figure 16) indicate that (1) generating validation samples from appended decompositions, (2) mixing and shuffling the calibration samples, and development samples improve the prediction performance compared with the scheme without these two steps.

Two-stage Variational Mode Decomposition and Support Vector Regression for Streamflow Forecasting

Ganggang Zuo, Jungang Luo, Ni Wang, Yani Lian, Xinxin He

State Key Laboratory of Eco-hydraulics in Northwest Arid Region, Xi’an University of Technology, Xi’an, Shaanxi 710048, China

Correspondence to: Jungang Luo (jgluo@xaut.edu.cn); Ni Wang (wangni@xaut.edu.cn)

Abstract. Streamflow forecasting is a crucial component in the management and control of water resources. Decomposition-based approaches have particularly demonstrated improved forecasting performance. However, direct decomposition of entire streamflow data with calibration and validation subsets is not practical for signal component prediction. This impracticality is due to the fact that the calibration process uses some validation information, that is not available in practical streamflow forecasting. Unfortunately, independent decomposition of calibration and validation sets lead to undesirable boundary effects and less accurate forecasting. To alleviate such boundary effects and improve the forecasting performance in basins lacking meteorological observations, we propose a two-stage decomposition prediction (TSDP) framework. We realize this framework using variational mode decomposition (VMD) and support vector regression (SVR), and refer to this realization as VMD-SVR. We demonstrate experimentally the effectiveness, efficiency and accuracy of the TSDP framework and its VMD-SVR realization in terms of the boundary effect reduction, computational cost, overfitting, in addition to decomposition and forecasting outcomes for different lead times. Specifically, four comparative experiments were conducted based on the ensemble empirical mode decomposition (EEMD), singular spectrum analysis (SSA), discrete wavelet transform (DWT), boundary-corrected maximal overlap discrete wavelet transform (BCMODWT), autoregressive integrated moving average (ARIMA), SVR, backpropagation neural network (BPNN) and long short-term memory (LSTM). The TSDP framework was also compared with the wavelet data-driven forecasting framework (WDDFF). Results of experiments on monthly runoff data collected from three stations at the Wei River show the superiority of the VMD-SVR model compared to benchmark models.

Nomenclature

TSDP	Two-stage decomposition prediction
TSDE	Three-stage decomposition ensemble
WDDFF	Wavelet data-driven forecasting framework
VMD	Variational mode decomposition
EEMD	Ensemble empirical mode decomposition
SSA	Singular spectrum analysis
DWT	Discrete wavelet transform
BCMODWT	Boundary-corrected maximal overlap discrete wavelet transform
PCA	Principal component analysis
SVR	Support vector regression
ARIMA	Autoregressive integrated moving average
BPNN	Backpropagation neural network
LSTM	Long short-term memory
ADF	Augmented Dickey Fuller
IMF	Intrinsic mode function
PACF	Partial autocorrelation coefficient
PCC	Pearson correlation coefficient
MI	Mutual information
MSE	Mean square error
NSE	Nash–Sutcliffe efficiency
NRMSE	Normalized root mean square error
PPTS	Peak percentage of threshold statistic
CV	Cross-validation
BOGP	Bayesian optimization based on Gaussian processes
GS	Grid search
VMD-SVR	A TSDP model based on VMD and SVR
EEMD-SVR	A TSDP model based on EEMD and SVR
SSA-SVR	A TSDP model based on SSA and SVR
DWT-SVR	A TSDP model based on DWT and SVR
BCMODWT-SVR	A WDDFF model based on BCMODWT and SVR.
VMD-SVR-A	A TSDE model based on VMD and SVR
EEMD-SVR-A	A TSDE model based on EEMD and SVR
SSA-SVR-A	A TSDE model based on SSA and SVR
DWT-SVR-A	A TSDE model based on DWT and SVR

1 Introduction

Reliable and accurate streamflow forecasting is of great significance for water resource management (Woldemeskel et al., 2018). The first attempts for streamflow prediction were based on precipitation measurements that date back to the 19th century

(Mulvaney, 1850; Todini, 2007). Since then, streamflow forecasting models have been progressively developed through the analysis of relevant physical processes and the incorporation of key hydrological terms into those models (Kratzert et al., 2018). The investigated hydrological terms include physical characteristics and boundary conditions of catchments, as well as spatial and temporal variabilities of hydrological processes (Kirchner, 2006; Paniconi and Putti, 2015). Also, physics-based models have been largely developed through harnessing high computational power and exploiting hydrometeorological and remote sensing data (Singh, 2018; Clark et al., 2015).

However, modeling hydrological processes with spatial and temporal variabilities at the catchment scale requires a lot of input meteorological data, information on boundary conditions and physical properties, as well as high-performance computational resources (Binley et al., 1991; Devia et al., 2015). Moreover, current physics-based models do not exhibit consistent performance on all scales and datasets because those models are constructed for small watersheds only (Kirchner, 2006; Beven, 1989; Grayson et al., 1992; Abbott et al., 1986). Therefore, physics-based models have been rarely used for practical streamflow forecasting (Kratzert et al., 2018). Alternatively, numerous studies have explored and developed data-driven models based on time-series analysis and machine learning (Wu et al., 2009).

In particular, streamflow prediction methods have been developed based on time-series models such as the Box-Jenkins (Castellano-Méndez et al., 2004), autoregressive (AR), moving average (MA), autoregressive moving average (ARMA) and autoregressive integrated moving average (ARIMA) models (Li et al., 2015; Mohammadi et al., 2006; Kisi, 2010; Valipour et al., 2013). However, the underlying linearity assumption of conventional time-series models makes them unsuitable for the forecasting of nonstationary, nonlinear, or variable streamflow patterns. Therefore, maximum likelihood (ML) models with nonlinear mapping capabilities have been introduced for streamflow forecasting. These models include decision trees (DT) (Erdal and Karakurt, 2013; Solomatine et al., 2008; Han et al., 2002), support vector regression (SVR) (Yu et al., 2006; Maity et al., 2010; Hosseini and Mahjouri, 2016), fuzzy inference systems (FIS) (Ashrafi et al., 2017; He et al., 2014; Yaseen et al., 2017) and artificial neural networks (ANN) (Kratzert et al., 2018; Nourani et al., 2009; Tiwari and Chatterjee, 2010; Rasouli et al., 2012).

Nevertheless, traditional ML models cannot always adequately forecast highly nonstationary, complex, nonlinear, or multiscale streamflow time-series data in catchments due to the lack of meteorological observations. To handle this inadequacy, signal processing algorithms have been applied to transform nonstationary time-series data into relatively stationary components, which can be analyzed more easily. These algorithms are most commonly based on flow decomposition, and they include wavelet analysis (WA) (Liu et al., 2014; Adamowski and Sun, 2010), empirical mode decomposition (EMD) (Huang et al., 2014; Meng et al., 2019), ensemble empirical mode decomposition (EEMD) (Bai et al., 2016; Zhao and Chen, 2015), singular spectrum analysis (SSA) (Zhang et al., 2015; Sivapragasam et al., 2001), seasonal-trend decomposition based on locally-estimated scatter-plot smoothing or LOESS (STL) (Luo et al., 2019) and variational mode decomposition (VMD) (He et al., 2019; Xie et al., 2019). These approaches have generally demonstrated improved streamflow forecasting.

However, the aforementioned decomposition-based methods don’t properly account for boundary effects on the decomposition results (Zhang et al., 2015). These boundary effects are effects that cause the boundary decompositions to be extrapolated. This extrapolation is carried out due to the unavailability of historical and future data points which serve as decomposition parameters (Zhang et al., 2015; Fang et al., 2019). In fact, each of these decomposition-based models firstly decompose the entire streamflow data and then divide the decomposition components into calibration and validation sets for streamflow prediction. This generally augments the calibration process with validation information, that is impractically available for realistic streamflow forecasting. Such validation information is useful in the reduction of the boundary effects, and is hence crucial for any operational streamflow forecasting algorithm. In order to avoid using this impractically-available validation information in calibration, streamflow time-series data must be first divided into calibration and validation sets, where each set is separately decomposed and the boundary effects are effectively reduced. Otherwise, the developed models would use some validation information in the calibration process, and hence would show unrealistically good forecasting performance.

Other relevant research contributions are those of Zhang et al. (2015), Du et al. (2017), Tan et al. (2018), Quilty and Adamowski (2018), and Fang et al. (2019) who recently pointed out and explicitly criticized the afore-mentioned impractical (and even incorrect) usage of signal processing techniques for streamflow data analysis. Zhang et al. (2015) evaluated and compared the outcomes of hindcast and forecast experiments (with and without validation information, respectively) for decomposition models based on WA, EMD, SSA, ARMA and ANN. The authors suggested that the decomposition-based models may not be suitable for practical streamflow forecasting. Du et al. (2017) demonstrated that the direct application of SSA and the discrete wavelet transform (DWT) to entire hydrological time-series data leads to incorrect outcomes. Tan et al. (2018) assessed the impracticality in streamflow forecasting with EEMD and ANN. Quilty and Adamowski (2018) addressed the pitfalls of using wavelet-based models for hydrological forecasting. Fang et al. (2019) demonstrated that EMD is not suitable for practical streamflow forecasting. In summary, these contributions have demonstrated that inadequate streamflow forecasting models often lead to practically unachievable performance.

Boundary effects still constitute a great challenge for practical streamflow forecasting. These effects can lead to shift variance for signal components, sensitivity to the addition of new data samples, and hence significant errors for decomposition-based models (see Section 3.4). Zhang et al. (2015) examined several extension methods, which can correct the boundary-affected decompositions, to reduce the boundary effects on decomposition outcomes. It was suggested that a properly-designed extension method can improve the forecasting performance. Quilty and Adamowski (2018) proposed a new wavelet-based data-driven forecasting framework (WDDFF), in which boundary-affected coefficients were removed by adopting either the stationary wavelet transform (SWT) algorithm (also known as “*algorithme à trous*”) or the maximal-overlap discrete wavelet transform (MODWT) algorithm. Tan et al. (2018) proposed an adaptive decomposition-based ensemble model to reduce boundary effects by adaptively adjusting the model parameters as new runoff data is added. These solutions demonstrated effective reduction of boundary effects.

In this context, we believe that a problem worthy of investigation is to reduce the influence of the boundary effects without altering or removing the boundary-effect decompositions, while providing high-confidence testing results on unseen data. To attain these goals, we designed a two-stage decomposition prediction (TSDP) framework, and proposed a TSDP realization based on VMD and SVR (where this realization is denoted by VMD-SVR). The proposed framework eliminates the need for validation information, reduces boundary effects, saves modeling time, avoids error accumulation, and improves the streamflow prediction performance. The key steps of this framework can be outlined as follows (see Section 3.4 for more details):

1. Divide the entire time series-data into a calibration set (which is then concurrently decomposed into time-series components), and a validation set (which is sequentially appended to the calibration set and decomposed).
2. Optimize and test a single data-driven forecasting model. For building a forecasting model, we use data samples that consist of input predictors (obtained by combining the predictors of different components of the signal decomposition), and output targets (selected from the original time series). The data samples can be divided into calibration samples (generated from the calibration-set decomposition), and validation samples (generated from the appended-set decomposition). The validation data samples are then divided into development samples (which are mixed and shuffled with the calibration samples to optimize the data-driven model), and testing samples (which are used to examine the confidence in the optimized data-driven model).

This paper aims to find a general solution for dealing with time-series decomposition errors caused by boundary effects. We designed four comparative experiments to demonstrate the effectiveness, efficiency, and accuracy of the designed TSDP framework and its VMD-SVR realization. Performance comparisons were made in terms of the reduction in boundary effects, computational cost, overfitting, as well as decomposition and forecasting outcomes for different lead times. In the first experiment, we demonstrate that the influence of boundary effects can be reduced through generating validation samples from appended-set decompositions, and then mixing and shuffling calibration and development samples. In the second experiment, we compare the performance of the TSDP framework with that of the three-stage decomposition ensemble (TSDE) framework, in which one optimized SVR model is built for each signal component. This comparison demonstrates that the designed TSDP framework saves the modeling time and might improves the prediction performance. In the third experiment, we demonstrate that combines the predictors of the individual signal components as the final predictors, barely overfits the TSDP models. For the fourth experiment, we compared the EEMD, SSA, DWT, VMD methods in the TSDP framework and the boundary-corrected maximal overlap discrete wavelet transform (BCMODWT) method in the WDDFF framework. Also, the decomposition-based models are compared to the no-decomposition ARIMA, SVR, BPNN and LSTM models. In order to evaluate the performance of the proposed model against the benchmark models, we used monthly runoff data collected at three stations which are located at the Wei River in China.

2 Monthly runoff data

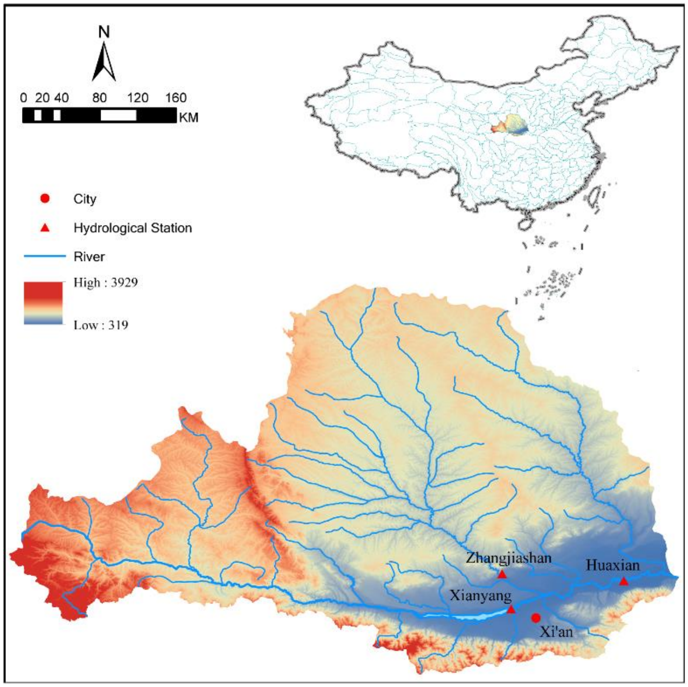


Figure 1: A geographical overview of the Wei River basin.

In this work, we use the monthly runoff data of the Wei River basin (Huang et al., 2014; He et al., 2019; He et al., 2020; Meng et al., 2019). The Wei River (see Fig. 1), the largest tributary of the Yellow River in China, lies between 33.68ºN-37.39ºN and 103.94ºE-110.03ºE and has a drainage area of 135,000 km^2 (Jiang et al., 2019). The Wei River has a total length of 818 km and originates from the Niaoshu Mountains in the Gansu province and flows east into the Yellow River (Gai et al., 2019). The associated catchment has a continental monsoon climate with an annual average precipitation of more than 550 mm. The precipitation of the flood season from June to September accounts for 60% of the annual total flow (Jiang et al., 2019). In the Guanzhong Plain, the Wei River serves as a key source of water for agricultural, industrial and domestic purposes (Yu et al., 2016). Therefore, robust monthly runoff prediction in this region plays a vital role in water resource allocation.

The historical monthly runoff records from January 1953 to December 2018 (792 records) at the Huaxian, Xianyang and Zhangjiashan stations (see Fig. 1) were used to evaluate the proposed model and the other state-of-the-art models. The records were collected from the Shaanxi Hydrological Information Center and the Water Resources Survey Bureau. The monthly runoff records were computed from the instantaneous values (in m^3/s) observed at 8 A.M. each day. The entire monthly runoff data was divided into calibration and validation sets. The calibration set covers the period from January 1953 to December 1998, and represents approximately 70% of the entire monthly runoff data. The validation set corresponds to the remaining period from January 1999 to December 2018. The validation set was further evenly divided into a development set (covering the period from January 1999 to December 2008) for selecting the optimal forecasting model, and a testing set (covering the period from January 2009 to December 2018) for validating the optimal model.

3 Methodologies

3.1 Variational mode decomposition

The variational mode decomposition (VMD) algorithm proposed by Dragomiretskiy and Zosso (2014) concurrently decomposes an input signal $f(t)$ into K intrinsic mode functions (IMFs).

The VMD process is mainly divided into two steps, namely (a) constructing a variational problem and (b) solving this problem. The constructed variational problem is expressed as follows:

$$\begin{cases} \min\limits_{\{u_k\}\{\omega_k\}} \left\{ \sum_k \left\| \partial_t \left[\left(\delta(t) + \frac{j}{\pi t} \right) * u_k(t) \right] e^{-j\omega_k t} \right\|_2^2 \right\}, \\ \qquad\qquad s.t. \quad \sum_k u_k(t) = f(t) \end{cases} \tag{1}$$

where $\{u_k\} = \{u_1, u_2, \cdots, u_k\}$ and $\{\omega_k\} = \{\omega_1, \omega_2, \cdots, \omega_k\}$ are shorthand notations for the set of modes and their center frequencies, respectively. The symbol t denotes time, $j^2 = -1$ is the square of the imaginary unit, * denotes the convolution operator, and δ is the Dirac delta function.

To solve this variational problem, a Lagrangian multiplier (λ) and a quadratic penalty term (α) are introduced to transform the constrained optimization problem (1) into an unconstrained problem. The augmented Lagrangian ℓ is defined as follows:

$$\ell(\{u_k\}, \{\omega_k\}, \lambda) := \alpha \sum_k \left\| \partial_t \left[\left(\delta(t) + \frac{j}{\pi t} \right) * u_k(t) \right] e^{-j\omega_k t} \right\|_2^2 + \| f(t) - \sum_k u_k(t) \|_2^2 + \langle \lambda(t), f(t) - \sum_k u_k(t) \rangle. \tag{2}$$

For the VMD method, the alternate direction method of multipliers (ADMM) is used to solve Eq. (2). The frequency-domain modes $u_k(\omega)$, the center frequencies ω_k and the Lagrangian multiplier λ are iteratively and respectively updated by

$$\hat{u}_k^{n+1}(\omega) = \frac{\hat{f}(\omega) - \sum_{i<k} \hat{u}_i^{n+1}(\omega) - \sum_{i>k} \hat{u}_i^{n}(\omega) + \frac{\hat{\lambda}^n(\omega)}{2}}{1 + 2\alpha(\omega - \omega_k)^2}, \tag{3}$$

$$\hat{\omega}_k^{n+1} = \frac{\int_0^\infty \omega \left| \hat{u}_k^{n+1}(\omega) \right|^2 d\omega}{\int_0^\infty \left| \hat{u}_k^{n+1}(\omega) \right|^2 d\omega}, \tag{4}$$

$$\hat{\lambda}^{n+1}(\omega) = \hat{\lambda}^n(\omega) + \tau\left(\hat{f}(\omega) - \sum_k \hat{u}_k^{n+1}(\omega) \right), \tag{5}$$

where n is the iteration counter, τ is the noise tolerance, while $\hat{u}_k^{n+1}(\omega)$, $\hat{f}(\omega)$, and $\hat{\lambda}^n(\omega)$ represent the Fourier transforms of $u_k^{n+1}(t)$, $f(t)$, and $\lambda^n(t)$, respectively.

The VMD performance is affected by the K, α, τ, and ε. A value of K that is too small may lead to poor IMF extraction from the input signal, whereas a too-large value of K may cause IMF information redundancy. A too-small value of α may lead to a large bandwidth, information redundancy, and additional noise for the IMFs. A too-large value of α may lead to a very small bandwidth and loss of some signal information. As shown in Eq. (5), the Lagrangian multiplier ensures optimal convergence when an appropriate value of $\tau > 0$ is used with a low-noise signal. The Lagrangian multiplier hinders the convergence when $\tau > 0$ is used with a highly noisy signal. This drawback can be avoided by setting τ to 0. However, it is not possible to reconstruct the input signal precisely if τ equals 0. Additionally, the value of ε affects the reconstruction error of the VMD.

3.2 Support Vector Regression

Support vector regression (SVR) was first proposed by Vapnik et al. (1997) for handling regression problems. The SVR mathematical principles are described here briefly.

For N pairs of samples $\{\boldsymbol{x_i}, y_i\}_{i=1}^{N}$, $\boldsymbol{x_i}$ and y_i denote the input variables and the desired output targets, respectively. Linear regression can be replaced by nonlinear regression, through the use of a nonlinear mapping function ϕ, as follows:

$$y_i \approx f(\boldsymbol{x_i}, \boldsymbol{w}) = \langle \boldsymbol{w}, \phi(\boldsymbol{x_i}) \rangle + b, \tag{6}$$

where $\boldsymbol{w}$ and b represent the regression weights and bias, respectively, and $\langle .,. \rangle$ is the inner product of two vectors. In the SVR framework, the error between y_i and $f(\boldsymbol{x_i}, \boldsymbol{w})$ is evaluated using the following ε-insensitive loss function:

$$|y_i - f(\boldsymbol{x_i}, \boldsymbol{w})|_\varepsilon = \begin{cases} 0, & if\ |y_i - f(\boldsymbol{x_i}, \boldsymbol{w})| < \varepsilon \\ |y_i - f(\boldsymbol{x_i}, \boldsymbol{w})| - \varepsilon, & otherwise \end{cases}. \tag{7}$$

Based on the $\boldsymbol{w}$ and b values, a regularized risk function R is defined as

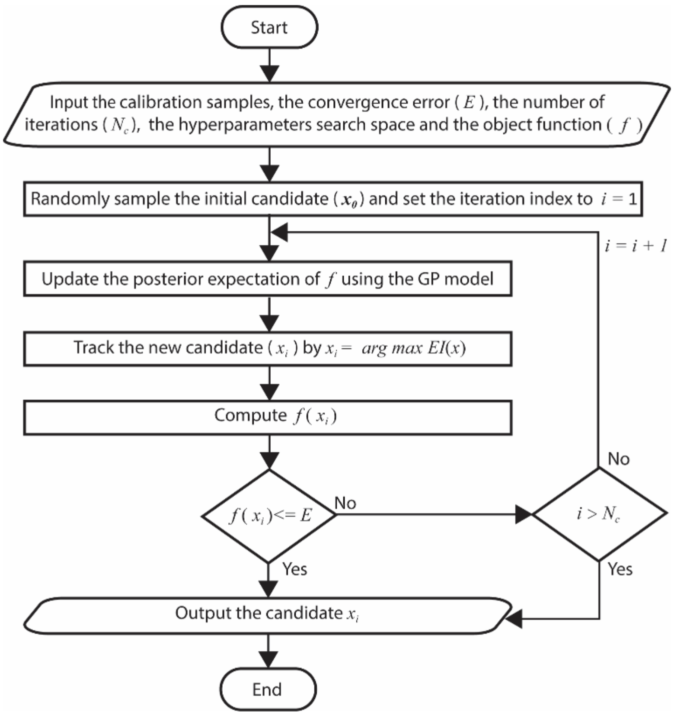
$$R = \frac{C}{N} \sum_{i=1}^{N} |y_i - f(\boldsymbol{x_i}, \boldsymbol{w})|_\varepsilon + \frac{1}{2} \|\boldsymbol{w}\|^2, \tag{8}$$

where the first term indicates the empirical risk based on the ε-insensitive loss function. The second term is a regularization term for penalizing the weight vector in order to limit the SVR model complexity. The parameter C is a weight penalty constant. To avoid high-dimensional nonlinear features $\phi(\boldsymbol{x})$, SVR uses a kernel trick that substitutes the inner product $\langle \phi(\boldsymbol{x}), \phi(\boldsymbol{x}') \rangle$ in the optimization algorithm with a kernel function, namely, $K(\boldsymbol{x}, \boldsymbol{x}')$. Some Lagrange multipliers, namely, α_i and β, are introduced to solve the constrained risk minimization problem. The Lagrange form of the regression function is

$$f(\boldsymbol{x}) = \sum_{i=1}^{N} \alpha_i K(\boldsymbol{x}, \boldsymbol{x}') + \beta. \tag{9}$$

The SVR model relies heavily on the kernel function and the hyperparameters. In this work, a radial basis function (RBF), namely, $K(\boldsymbol{x}, \boldsymbol{x}') = exp(-\|\boldsymbol{x} - \boldsymbol{x_i}\|^2 / 2\sigma^2)$, is used as the kernel function. The parameter σ is used to control the RBF width. In this study, the hyperparameters ε, C, and σ are tuned by Bayesian optimization.

3.3 Bayesian optimization based on Gaussian processes



Start
Input the calibration samples, the convergence error (E), the number of iterations (N_c), the hyperparameters search space and the object function (f)
Randomly sample the initial candidate (x_0) and set the iteration index to $i = 1$
$i = i + 1$
Update the posterior expectation of f using the GP model
Track the new candidate (x_i) by $x_i = arg\ max\ EI(x)$
Compute $f(x_i)$
No
$f(x_i)<= E$
No
$i > N_c$
Yes
Yes
Output the candidate x_i
End

Figure 2: A flowchart of the Bayesian optimization.

Bayesian optimization (BO) is a sequential model-based optimization (SMBO) approach typically used for global optimization of black-box objective functions, for which the true distribution is unknown or the evaluation is extremely expensive. For such objective functions, the BO algorithm sets a prior belief on the loss function in a learning model, sequentially refines this model by gathering function evaluations, and updates the Bayesian posterior (James et al., 2011; Shahriari et al., 2016).

To update the beliefs about the loss function and calculate the posterior expectation, a prior function is applied. Here, we assume that the real loss function distribution can be described by a Gaussian process (GP). Therefore, the loss function values $\{f(\boldsymbol{x_i})\}_{i=1}^{n}$ for an evaluation set $\{\boldsymbol{x_i}\}_{i=1}^{n}$ satisfy the multivariate Gaussian distribution over the function space

$$f_{1:n} \sim N(m(\boldsymbol{x_{1:n}}), \boldsymbol{K}), \tag{10}$$

where $m(\boldsymbol{x_{1:n}})$ is the GP mean function set and $\boldsymbol{K}$ is a kernel matrix given by the covariance function $K(\boldsymbol{x}, \boldsymbol{x}')$. An acquisition function is used to assess the utility of candidate points for finding the posterior distribution. In particular, the candidate point with the highest utility is selected as the candidate for the next evaluation of f. Many acquisition functions have been explored for Bayesian optimization. These functions include the expected improvement (EI), the upper confidence bounds (UCB), the probability of improvement, the Thompson sampling (TS), and the entropy search (ES). However, the EI function is the most commonly used among these functions (James et al., 2011; Shahriari et al., 2016). For the GP model, the expected improvement can be calculated as

$$EI(\boldsymbol{x}) = \begin{cases} [\mu(\boldsymbol{x}) - f(\widehat{\boldsymbol{x}})]\Phi(\boldsymbol{z}) + \sigma(\boldsymbol{x})\phi(\boldsymbol{z}) & if\ \sigma(\boldsymbol{x}) > 0 \\ 0 & if\ \sigma(\boldsymbol{x}) = 0 \end{cases}, \tag{11}$$

$$\boldsymbol{z} = \frac{\mu(\boldsymbol{x}) - f(\widehat{\boldsymbol{x}})}{\sigma(\boldsymbol{x})}, \tag{12}$$

where $f(\widehat{\boldsymbol{x}})$ is the current lowest loss value, and $\mu(\boldsymbol{x})$ is the expected loss value, while $\Phi(\boldsymbol{z})$ and $\phi(\boldsymbol{z})$ are the cumulative distribution function and the probability density function, respectively. Figure 2 shows a flowchart of the Bayesian optimization method based on Gaussian processes (BOGP).

3.4 The TSDP framework and the VMD-SVR realization

The boundary effects introduce errors into the construction of decomposition-based models. These errors arise from the extrapolation of the boundary decomposition components. In fact, this extrapolation is carried out due to the unavailability of historical and future data points which serve as decomposition parameters (Zhang et al., 2015; Fang et al., 2019). To find out the extent to which the boundary effects contribute to decomposition errors, we have evaluated the shift-copy variance and the data-addition sensitivity for each of the VMD, DWT, EEMD, and SSA methods. Given the monthly runoff data of the Huaxian station from January 1953 to November 2018, i.e., $\boldsymbol{x_0} = [q_1, q_2, \cdots, q_{791}]$, and a one-step-ahead (shift) copy of $\boldsymbol{x_0}$, i.e., $\boldsymbol{x_1} = [q_2, q_3, \cdots, q_{792}]$, assume the VMD method is applied to $\boldsymbol{x_0}$ and $\boldsymbol{x_1}$. Then, the $IMF_1(2{:}791)$ for the VMD of $\boldsymbol{x_0}$ should be maintained by $IMF_1(1{:}790)$ for the VMD of $\boldsymbol{x_1}$ since $\boldsymbol{x_0}(2{:}791)$ is maintained by $\boldsymbol{x_1}(1{:}790)$. The IMF_1 is the first decomposed signal component and "(2:791)" means the second data point to the 791st data point. However, the boundary decompositions of $\boldsymbol{x_0}(2{:}791)$ and $\boldsymbol{x_1}(1{:}790)$ are completely different (see Fig. 3a and b). Therefore, VMD is shift-variant. For the Huaxian station, given the monthly runoff data from January 1953 to November 2018, i.e., $\boldsymbol{x_{1-791}} = [q_1, q_2, \cdots, q_{791}]$ and the monthly runoff data from January 1953 to December 2018/12, i.e., $\boldsymbol{x_{1-792}} = [q_1, q_2, \cdots, q_{792}]$, the IMF_1 for the VMD of $\boldsymbol{x_{1-791}}$ should be maintained by the $IMF_1(1{:}791)$ for the VMD of $\boldsymbol{x_{1-792}}$, since $\boldsymbol{x_{1-791}}$ is maintained by $\boldsymbol{x_{1-792}}(1{:}791)$. However, the boundary decompositions of $\boldsymbol{x_{1-791}}$ and $\boldsymbol{x_{1-792}}(1{:}791)$ are completely different (see Fig. 3c and d). A similar result was obtained for the case in which several data points were appended to a given time series (see Fig. 3e and f). Therefore, VMD is also sensitive to the addition of new data. It can be demonstrated that the EEMD, DWT and SSA are also shift-variant and sensitive to addition of new data. The BCMODWT method developed by Quilty and Adamowski (2018) is shift-invariant, insensitive to the addition of new data, and also shows no decomposition errors. Thus we compared in this work the BCMODWT method of the WDDFF framework with the VMD, EEMD, SSA and DWT methods of the TSDP framework. The results in Fig. 3 collectively indicate that the concurrent decomposition errors are extremely small except for those of the boundary decompositions.

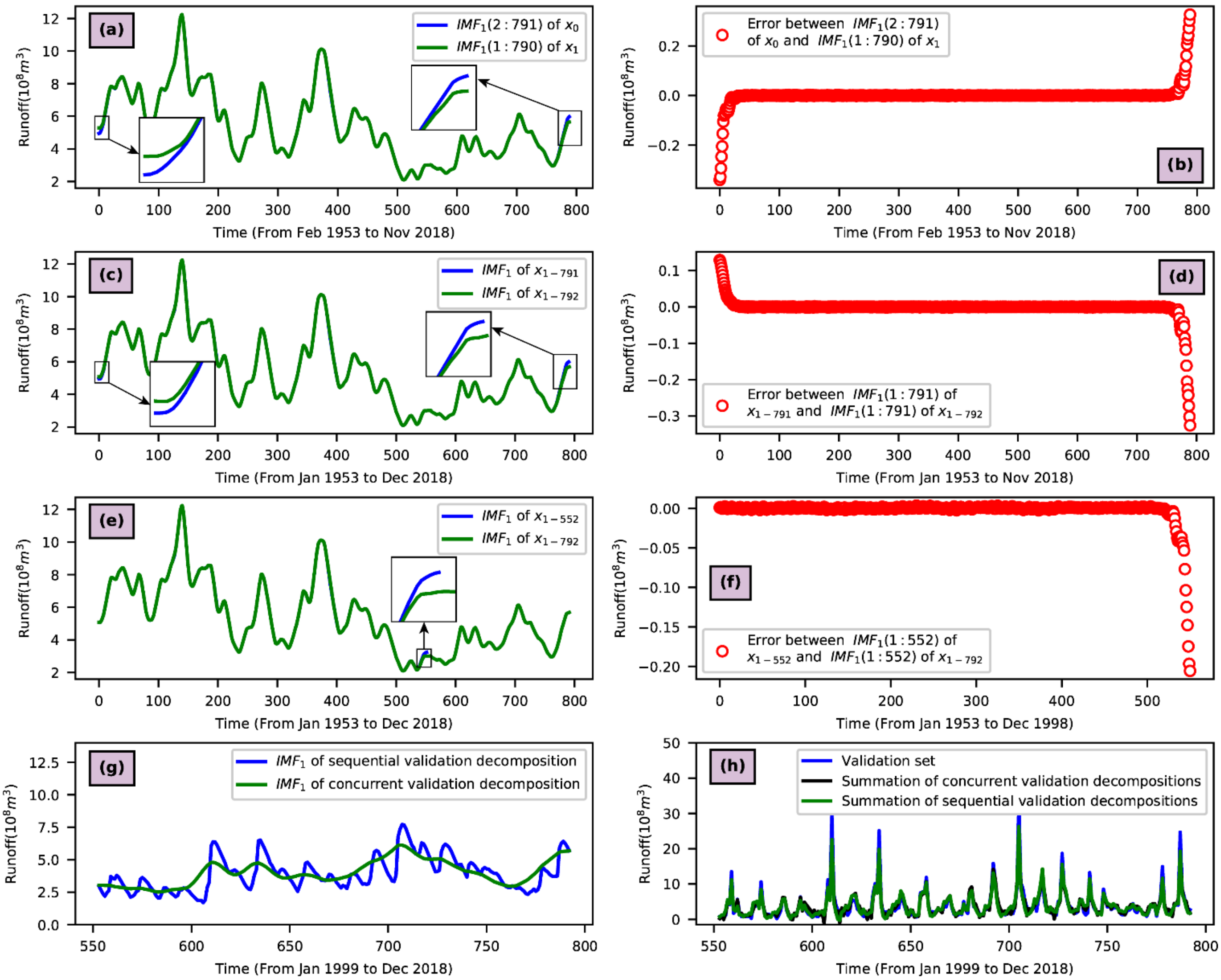


Figure 3: Examples of the boundary effects for the VMD method on monthly runoff data at the Huaxian station: (a and b) shift variance, (c and d) sensitivity to appending one data point, (e and f) sensitivity to appending several data points, (g) differences between sequential and concurrent validation-data decompositions, and (h) differences between the summation of the sequential and concurrent validation-data decompositions.

However, the boundary effects introduce small decomposition errors for the calibration set, but large such errors for the validation set. This is because the calibration set is concurrently decomposed whereas the validation set is sequentially appended to the calibration set and decomposed. Additionally, the last decompositions of an appended set are selected as the validation decompositions. Note that this procedure is followed for three reasons. (1) This procedure simulates practical forecasting scenarios in which a time series is observed and predicted incrementally. (2) The validation set should be decomposed on a sample-by-sample basis to avoid validation-data decomposition using future information. (3) The decomposition algorithms such as VMD, EEMD, SSA and DWT cannot decompose one validation data point each time (and might output the “not a number” data type). The decomposition errors of the calibration set could be ignored because only few of the boundary decompositions have relatively large errors (see Fig. 3f). Unfortunately, the decomposition errors of the validation set cannot be ignored because all decompositions of this set are selected from the boundary decompositions of the appended sets. In this context, large decomposition errors (corresponding to the differences between the blue and green lines in Fig. 3g) will be introduced to the model validation process. Figure 4 shows that the error distribution of the validation set has a larger scale than that of the calibration set. Thus, models calibrated on the calibration samples might generalize poorly on the validation samples due to the difference in error distribution between the calibration and validation decompositions.

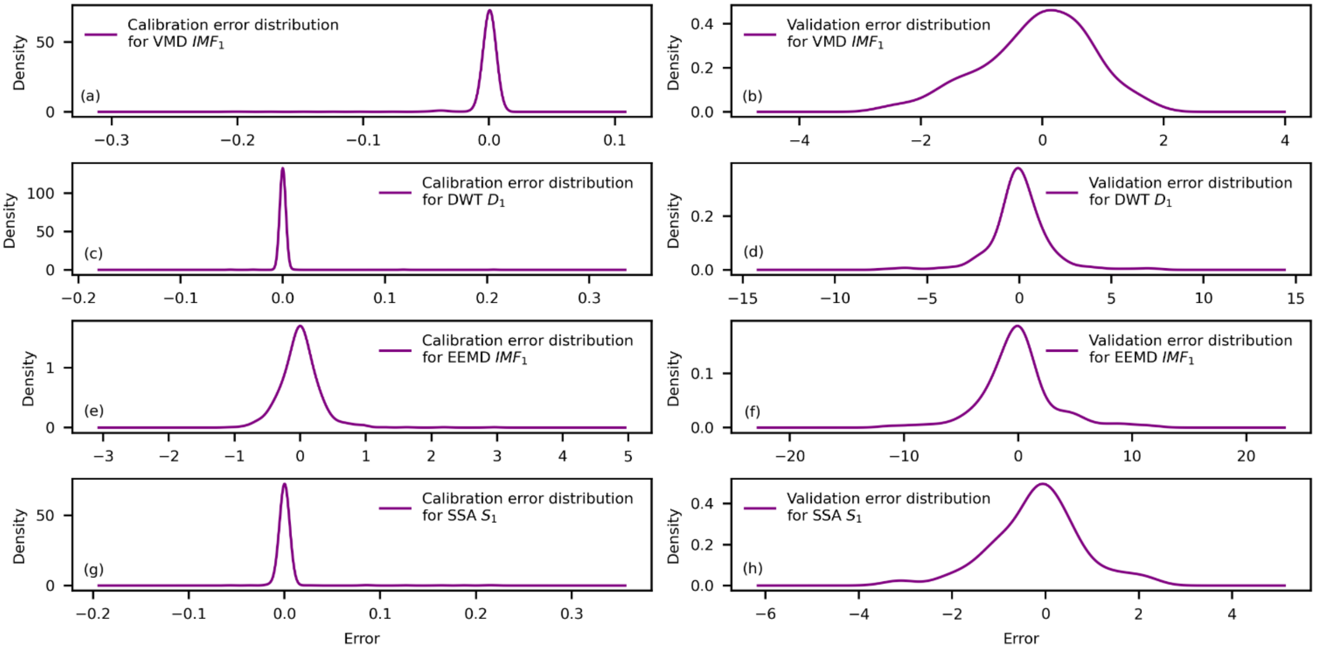


Figure 4: Density estimates with Gaussian-type kernels for the calibration and validation error distributions of the monthly runoff decompositions of the Huaxian station. The real decompositions are the joint decompositions of the entire monthly runoff for the period from January 1953 to December 2018.

Fortunately, the difference in error distribution between the calibration and validation decompositions can be handled without altering or removing the boundary decompositions. This is based on three key remarks: (1) the boundary-affected decompositions might contain some valuable information for building practical forecasting models, (2) the distribution of the validation samples can be different from that of the calibration samples (Ng, 2017), and (3) the validation decomposition errors can be eliminated by summing signal components into the original signal (see Fig. 3h). Note that the summation of the sequential validation decompositions of Fig. 3(h) cannot completely reconstruct the validation set. This is mainly caused by setting the VMD noise tolerance (τ) to 0 in this work (see Section 3.1) rather than the introduced validation decomposition errors. Therefore, the decomposition errors barely affect the prediction performance if the decomposition-based models are properly constructed to learn from the calibration set and generalize well to the validation set.

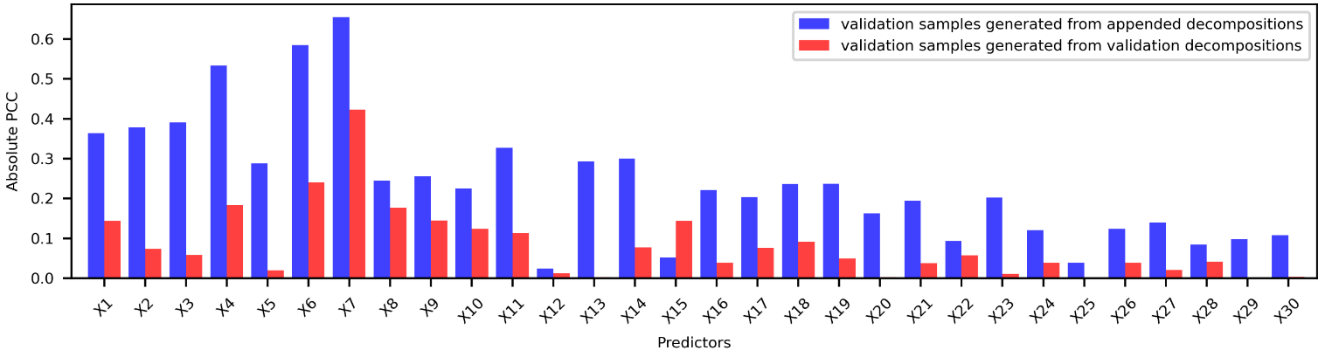


Figure 5: Absolute Pearson correlation coefficients (PCC) between predictors and predicted targets of validation samples generated from the VMD appended decompositions and validation decompositions. The samples were collected at the Huaxian station.

One way to deal with the influences of boundary effects is to generate validation samples using decompositions of appended sets, i.e., appended decompositions. The last sample generated from appended decompositions is selected as a validation sample since the predicted target of this sample belongs to the validation period. The advantage is that the predictors selected from appended decompositions are more correlated with the prediction targets than the predictors selected from validation decompositions (see Fig. 5). This is because the appended set is decomposed concurrently. However, the validation decompositions are reorganized from the decompositions of appended sets, which leads to the relationships between a decomposition and its lagging decompositions are changed a lot.

The other way to deal with boundary effects is to assess the validation error distribution during the calibration stage. A promising way to achieve this goal is to use the cross-validation (CV) based on the mixed and shuffled samples generated from the calibration and validation distributions. The key advantage is that the developed models are simultaneously calibrated and validated on these distributions. Additionally, enough validation samples should be allocated for testing the final optimized models in order to give users a high confidence level on unseen data. Therefore, the validation samples are further split into development samples for cross-validation and testing samples for testing the final optimized data-driven models.

Based on the aforementioned key remarks, the TSDP framework is designed as follows: *(i) Time series decomposition:* divide the entire time series (monthly runoff data in this work) into a calibration set (which is then concurrently decomposed) and a validation set (which is then sequentially appended to the calibration set and decomposed). *(ii) Time series prediction:* optimize and test a single prediction model using calibration and validation samples generated from the calibration and appended decompositions. For these samples, the optimal lag times (measured in hours, days, months, or years) of the decomposed signal components are combined as predictors, while the original signal samples are used as the desired prediction targets. This is the direct approach which has already been used by Maheswaran and Khosa (2013), Du et al. (2017) and Quilty and Adamowski (2018).

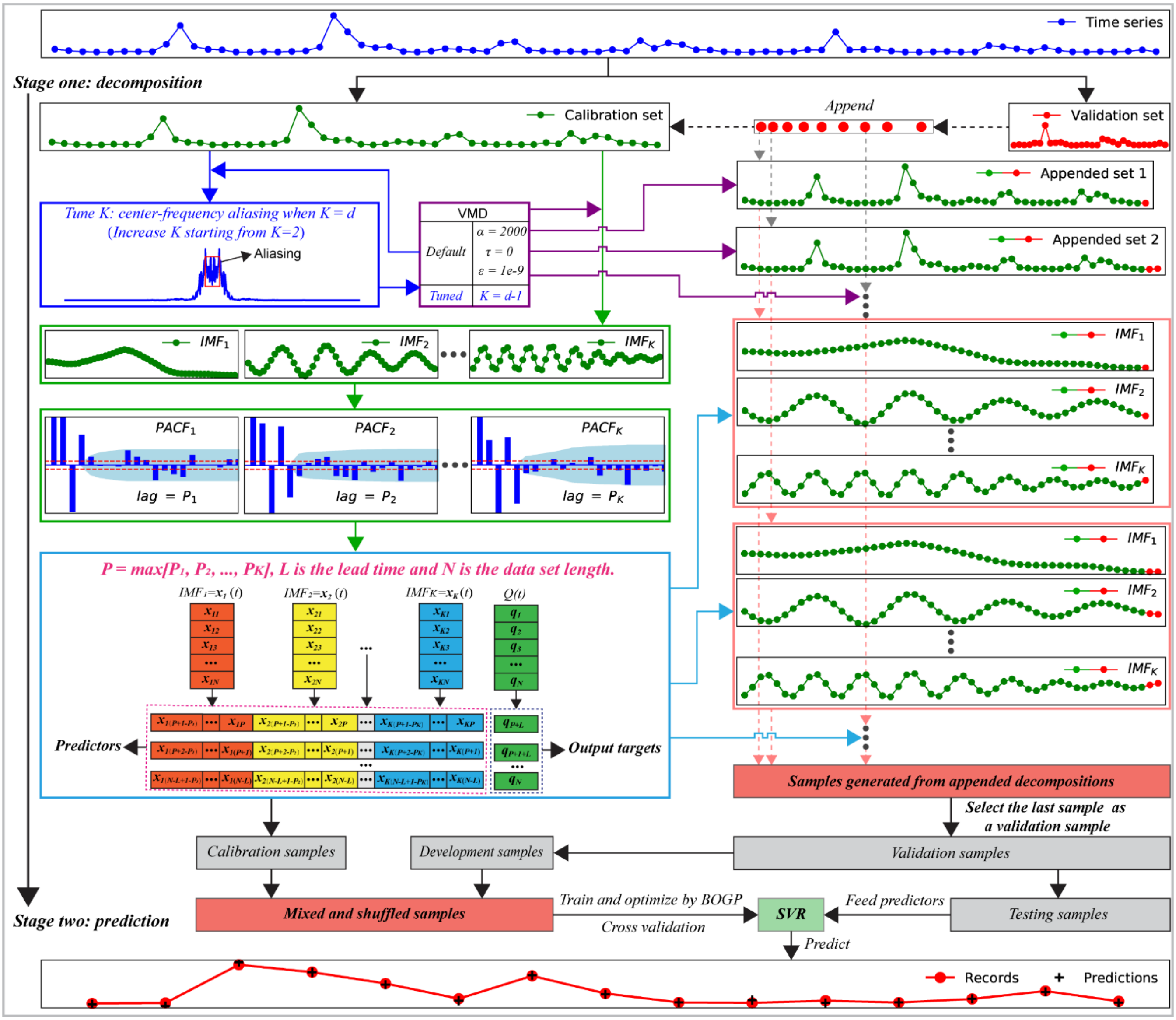

Time series
Stage one: decomposition
Calibration set
Append
Validation set
Appended set 1
Appended set 2
Tune K: center-frequency aliasing when K = d
(Increase K starting from K=2)
Aliasing
VMD
Default
α = 2000
τ = 0
ε = 1e-9
Tuned
K = d-1
IMF$_1$
IMF$_2$
IMF$_K$
PACF$_1$
PACF$_2$
PACF$_K$
lag = P$_1$
lag = P$_2$
lag = P$_K$
P = max[P$_1$, P$_2$, ..., P$_K$], L is the lead time and N is the data set length.
IMF$_1$=$x_1(t)$
IMF$_2$=$x_2(t)$
IMF$_K$=$x_K(t)$
Q(t)
x_{11} x_{12} x_{13} ••• x_{1N}
x_{21} x_{22} x_{23} ••• x_{2N}
x_{K1} x_{K2} x_{K3} ••• x_{KN}
q_1 q_2 q_3 ••• q_N
Predictors
Output targets
$x_{1(P+1-P_1)}$ ••• x_{1P} $x_{2(P+1-P_2)}$ ••• x_{2P} ••• $x_{K(P+1-P_K)}$ ••• x_{KP} q_{P+L}
$x_{1(P+2-P_1)}$ ••• $x_{1(P+1)}$ $x_{2(P+2-P_2)}$ ••• $x_{2(P+1)}$ ••• $x_{K(P+2-P_K)}$ ••• $x_{K(P+1)}$ q_{P+1+L}
$x_{1(N-L+1-P_1)}$ ••• $x_{1(N-L)}$ $x_{2(N-L+1-P_2)}$ ••• $x_{2(N-L)}$ ••• $x_{K(N-L+1-P_K)}$ ••• $x_{K(N-L)}$ q_N
Samples generated from appended decompositions
Select the last sample as a validation sample
Calibration samples
Development samples
Validation samples
Mixed and shuffled samples
Train and optimize by BOGP
Cross validation
SVR
Feed predictors
Testing samples
Predict
Stage two: prediction
Records
Predictions

Figure 6: A block diagram of the two-stage decomposition prediction (TSDP) framework with the VMD-SVR realization.

The design details of the TSDP framework and its VMD-SVR realization are summarized as follows (see Fig. 6).

Step 1 Collect time-series data $Q(t)$ as the VMD-SVR input ($t = 1,2,\cdots N$, where N is the length of the time-series data).

Step 2 Divide the time-series data into calibration and validation sets (with 70% and 30% of the overall monthly runoff data, respectively, in this work).

Step 3 Concurrently extract K IMF signal components from the calibration set using the VMD scheme. For optimal selection of K, check whether the last extracted IMF component exhibits central-frequency aliasing.

Step 4 Sequentially append the validation data samples to the calibration set to generate appended sets. Decompose each appended set into K signal components using the VMD scheme.

Step 5 Plot the partial autocorrelation function (PACF) of each signal component for the calibration set in order to select the optimal lag period and hence generate modeling samples. The PACF lag count is set to 20. We assume that the predicted target of the k^{th} signal component is $x_k(t+L)$ (where L is the lead time which is measured in hours, days, months or years). If the PACF of the mth lag period lies outside the 95% confidence interval (i.e., $\left[-\frac{1.96}{\sqrt{n}},\frac{1.96}{\sqrt{n}}\right]$, where n is the signal component length) and is insignificant after the mth lag period, then the samples $x_k(t), x_k(t-1), \cdots, x_k(t+1-m)$ are selected as input predictors and m is selected as the optimal lag period for the k^{th} signal component.

Step 6 Combine the input predictors of each signal component to form the SVR predictors. Select the original time-series data sample after the maximum lag period ($Q(t+L)$) as the predicted target.

Step 7 Based on the input predictors and output targets obtained in Step 6, generate calibration samples using the calibration signal components. Also, generate appended samples using the appended signal components obtained in Step 4. Select the last sample of the appended samples as a validation sample. Divide the validation samples evenly into development and testing samples.

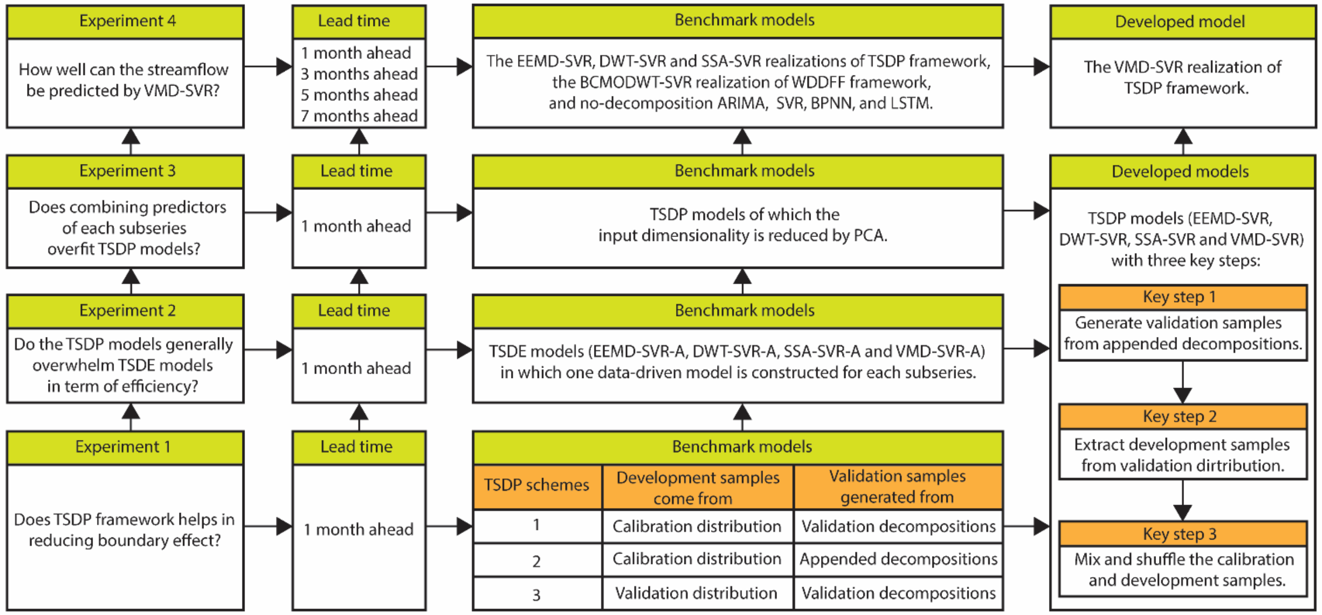
Step 8 Mix and shuffle the calibration and development samples. Train and optimize the SVR model using the shuffled samples and the BOGP algorithm. For testing, feed the test sample predictors into the optimized SVR model in order to predict time series samples and compare them against the original ones. The VMD-SVR output is the predicted samples for the test predictors.

Steps 1-4 represent the *decomposition* stage of the proposed framework while Steps 5-8 represent the *prediction* stage. Note that the VMD and SVR schemes can be respectively replaced by other decomposition and data-driven prediction models.

3.5 Comparative experimental setups

As shown in Fig. 7, we design four comparative experiments to evaluate the effectiveness, efficiency, and accuracy of the TSDP framework and its VMD-SVR realization. The evaluation is carried on in terms of the boundary effect reduction (see Ex. 1), computational cost (see Ex. 2), overfitting (see Ex. 3) as well as decomposition and forecasting capabilities for different lead times (see Ex. 4). The previous experiments represent the baseline for the next ones. We first give a brief review of the EEMD, SSA, DWT, and BCMODWT methods. Then, we explain each experiment in detail.

The EEMD method decomposes a time series into several IMFs and one residual (R) given the white noise amplitude (ε) and the number of ensemble members (M). In this work, we set M and ε to 100 and 0.2, respectively, as suggested by Wu and Huang (2009). The singular spectrum analysis (SSA) method decomposes a time series into independent trend, oscillation, and noise components ($\{S_1, \cdots, S_L\}$)). This decomposition is parameterized by the window length (W_L) and the number of groups (m). The SSA method has four main steps, namely embedding, singular value decomposition (SVD), grouping, and diagonal averaging. If one of the subseries is periodic, W_L can be set to the period of this subseries to enhance decomposition performance (Zhang et al., 2015). However, the grouping step can be ignored (i.e., we do not need to set m) if the value of W_L is small (e.g., $W_L \leq 20$) because grouping may hide information in the grouped subseries. In this work, W_L was set to 12 because we perform monthly runoff forecasting. The discrete wavelet transform (DWT) decomposes a time series into several detail series ($\{D_1, \cdots, D_L\}$) and one approximation series (A_L) given a discrete mother wavelet function (ψ) and a decomposition level (L). These parameters are typically selected experimentally. In this work, we set ψ to the *db10* as suggested by Seo et al. (2015). Also, we set L to $int[\log(N)]$ following Nourani et al. (2009). Given ψ and L, the BCMODWT method decomposes a given time-series into wavelets ($\{W_1, W_2, \cdots, W_L\}$) and scaling coefficients (V_L). The number of boundary-affected wavelets and scaling coefficients is given by $(2^L - 1)(J - 1) + 1$ (where J is the length of the given wavelet filter) (Quilty and Adamowski, 2018). These boundary-affected wavelets and scaling coefficients are finally removed by BCMODWT. In this work, several wavelet functions were evaluated including *haar*, *db1*, *fk4*, *coif1*, *sym4*, *db5*, *coif2* and *db10* (with wavelet filter lengths of 2, 2, 4, 6, 8, 10, 12, and 20, respectively). Since we have only 792 monthly runoff values and the BCMODWT method removes some wavelet and scaling coefficients, the maximum decomposition level was set to 4 (286 wavelets and scaling coefficients were removed for *db10*).


Experiment 4
How well can the streamflow be predicted by VMD-SVR?
Lead time
1 month ahead
3 months ahead
5 months ahead
7 months ahead
Benchmark models
The EEMD-SVR, DWT-SVR and SSA-SVR realizations of TSDP framework, the BCMODWT-SVR realization of WDDFF framework, and no-decomposition ARIMA, SVR, BPNN, and LSTM.
Developed model
The VMD-SVR realization of TSDP framework.
Experiment 3
Does combining predictors of each subseries overfit TSDP models?
Lead time
1 month ahead
Benchmark models
TSDP models of which the input dimensionality is reduced by PCA.
Developed models
TSDP models (EEMD-SVR, DWT-SVR, SSA-SVR and VMD-SVR) with three key steps:
Key step 1
Generate validation samples from appended decompositions.
Key step 2
Extract development samples from validation dirtribution.
Key step 3
Mix and shuffle the calibration and development samples.
Experiment 2
Do the TSDP models generally overwhelm TSDE models in term of efficiency?
Lead time
1 month ahead
Benchmark models
TSDE models (EEMD-SVR-A, DWT-SVR-A, SSA-SVR-A and VMD-SVR-A) in which one data-driven model is constructed for each subseries.
Experiment 1
Does TSDP framework helps in reducing boundary effect?
Lead time
1 month ahead
Benchmark models

TSDP schemes	Development samples come from	Validation samples generated from
1	Calibration distribution	Validation decompositions
2	Calibration distribution	Appended decompositions
3	Validation distribution	Validation decompositions

Figure 7: A block diagram of the comparative experimental setups.

3.5.1 Experiment 1: Evaluation of the boundary effect reduction

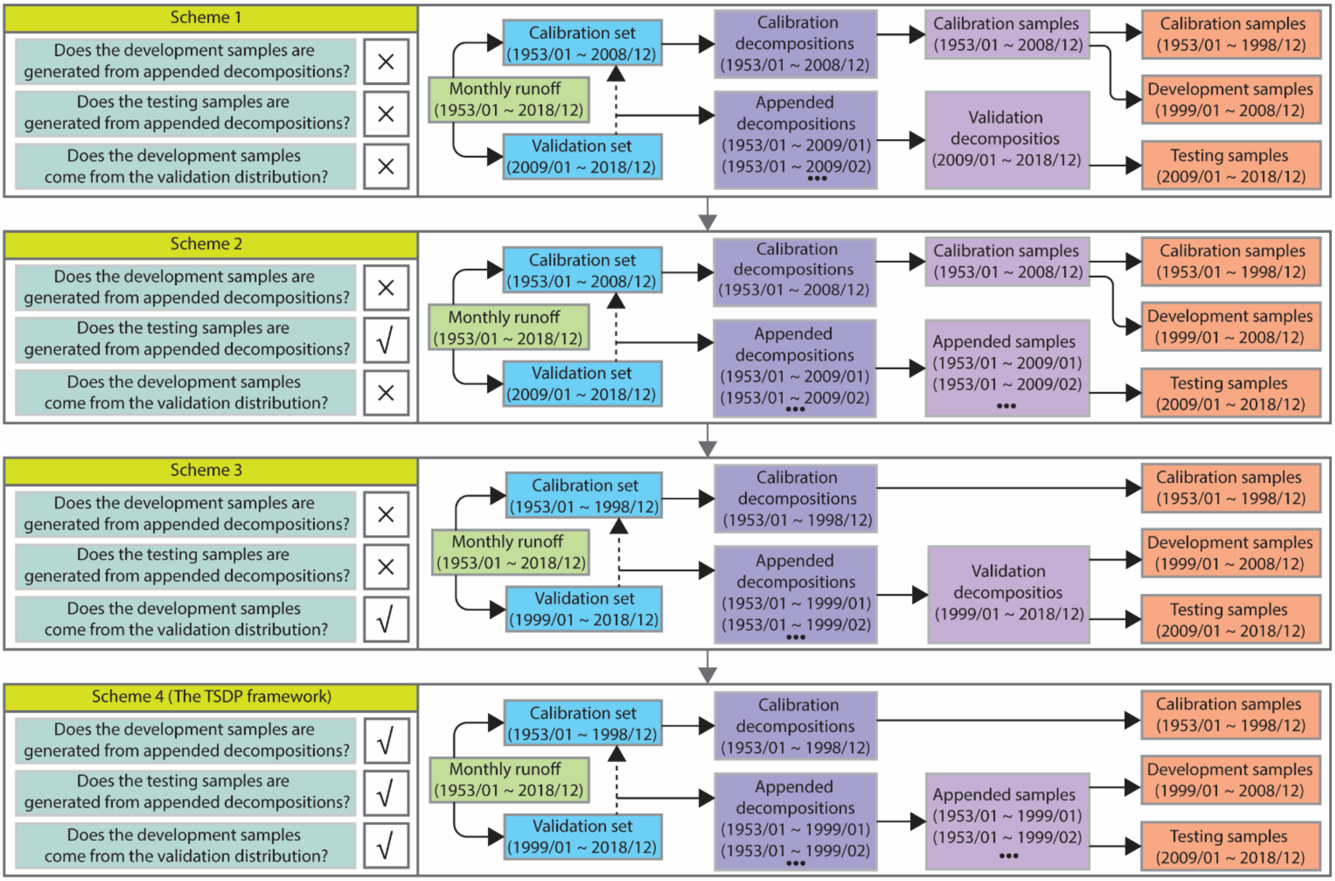

Scheme 1
Does the development samples are generated from appended decompositions?
×
Does the testing samples are generated from appended decompositions?
×
Does the development samples come from the validation distribution?
×
Monthly runoff (1953/01 ~ 2018/12)
Calibration set (1953/01 ~ 2008/12)
Validation set (2009/01 ~ 2018/12)
Calibration decompositions (1953/01 ~ 2008/12)
Appended decompositions (1953/01 ~ 2009/01) (1953/01 ~ 2009/02) •••
Calibration samples (1953/01 ~ 2008/12)
Validation decompositions (2009/01 ~ 2018/12)
Calibration samples (1953/01 ~ 1998/12)
Development samples (1999/01 ~ 2008/12)
Testing samples (2009/01 ~ 2018/12)
Scheme 2
Does the development samples are generated from appended decompositions?
×
Does the testing samples are generated from appended decompositions?
√
Does the development samples come from the validation distribution?
×
Monthly runoff (1953/01 ~ 2018/12)
Calibration set (1953/01 ~ 2008/12)
Validation set (2009/01 ~ 2018/12)
Calibration decompositions (1953/01 ~ 2008/12)
Appended decompositions (1953/01 ~ 2009/01) (1953/01 ~ 2009/02) •••
Calibration samples (1953/01 ~ 2008/12)
Appended samples (1953/01 ~ 2009/01) (1953/01 ~ 2009/02) •••
Calibration samples (1953/01 ~ 1998/12)
Development samples (1999/01 ~ 2008/12)
Testing samples (2009/01 ~ 2018/12)
Scheme 3
Does the development samples are generated from appended decompositions?
×
Does the testing samples are generated from appended decompositions?
×
Does the development samples come from the validation distribution?
√
Monthly runoff (1953/01 ~ 2018/12)
Calibration set (1953/01 ~ 1998/12)
Validation set (1999/01 ~ 2018/12)
Calibration decompositions (1953/01 ~ 1998/12)
Appended decompositions (1953/01 ~ 1999/01) (1953/01 ~ 1999/02) •••
Validation decompositions (1999/01 ~ 2018/12)
Calibration samples (1953/01 ~ 1998/12)
Development samples (1999/01 ~ 2008/12)
Testing samples (2009/01 ~ 2018/12)
Scheme 4 (The TSDP framework)
Does the development samples are generated from appended decompositions?
√
Does the testing samples are generated from appended decompositions?
√
Does the development samples come from the validation distribution?
√
Monthly runoff (1953/01 ~ 2018/12)
Calibration set (1953/01 ~ 1998/12)
Validation set (1999/01 ~ 2018/12)
Calibration decompositions (1953/01 ~ 1998/12)
Appended decompositions (1953/01 ~ 1999/01) (1953/01 ~ 1999/02) •••
Appended samples (1953/01 ~ 1999/01) (1953/01 ~ 1999/02) •••
Calibration samples (1953/01 ~ 1998/12)
Development samples (1999/01 ~ 2008/12)
Testing samples (2009/01 ~ 2018/12)

Figure 8: A block diagram for different methods of generating the training, development and test samples.

First, we show how boundary effects can be reduced through the generation of validation samples from appended decompositions, and then mixing and shuffling the calibration and development samples. As shown in Fig .8, we compare four TSDP schemes for one-month-ahead runoff forecasting in the first experiment. The development samples of the schemes 1 and 2 come from the calibration distribution whereas those of the schemes 3 and 4 come from the validation distribution. The testing samples of the schemes 1 and 3 are generated from validation decompositions whereas those of schemes 2 and 4 are generated from appended decompositions. The comparisons between the first and second TSDP schemes and between the third and fourth TSDP schemes are carried out to verify whether generating samples from appended decompositions reduces boundary effects. Moreover, the comparisons between the first and third TSDP schemes and between the second and fourth TSDP schemes are performed to check whether the mixing-and-shuffling step reduces boundary effects.

3.5.2 Experiment 2: Evaluation of TSDE models

In the second experiment, we compare the prediction performance and the computational cost of the TSDP and TSDE models for one-month-ahead runoff forecasting. Those models are implemented based on the EEMD, SSA, VMD, DWT and SVR methods. In particular, we investigate four combined schemes for TSDE models, namely EEMD-SVR-A ('A' means the ensemble approach is the *addition ensemble*), SSA-SVR-A, VMD-SVR-A, and DWT-SVR-A. The TSDE models include the extra ensemble stage compared to the TSDP models. The decomposition stages of the TSDP and TSDE models are identical. In the TSDE prediction stage, PACF is also used for selecting the predictors and the predicted target for each signal component. However, one optimized SVR model will be trained for each signal component. In the test phase, this model will be used for component prediction. The remaining prediction procedures are identical to those of the TSDP models. For testing in the ensemble stage, the prediction results of all signal components are fused to predict the streamflow data. Since the TSDE models build one SVR model for each signal component, the computational cost of each TSDE model is expected to be significantly higher than that of the corresponding TSDP model.

3.5.3 Experiment 3: Evaluation of the PCA-based dimensionality reduction

Our third experiment tests whether dimensionality reduction (i.e. reduction of the number of predictors) improves the prediction performance of the TSDP models. The TSDP models can reduce the modeling time and possibly improve the prediction performance compared with the TSDE models. However, combining the predictors of all signal components as the TSDP input predictors may lead to overfitting. This is because the TSDP predictors might be correlated and are typically much more than the TSDE ones. Therefore, it is necessary to test whether the reduction of the number of the TSDP predictors can help improve the prediction performance.

Principal component analysis (PCA) has been a key tool for addressing the overfitting problem of redundant predictors (Wangmeng Zuo et al., 2005; Musa, 2014). Therefore, PCA is used in this work to reduce the TSDP input predictors. This analysis uses an orthogonal transformation in order to transform the correlated predictors into a set of linearly uncorrelated predictors or principal components. For further details on PCA, see Jolliffe (2002). The main PCA parameter is the number of principal components, which indicates the number of predictors retained by the PCA procedure. The optimal number of predictors is found through grid search. We also estimate this number using the MLE method of Minka (2001). Since the number of predictors varies for different TSDP models, the (guessed) number of predictors is replaced by the (guessed) number of excluded predictors for convenience of comparison. In this paper, the number of excluded predictors ranges from 0 to 16. A value of 0 indicates that all predictors are retained (i.e. the dimensionality is not reduced), but the correlated predictors are transformed into uncorrelated ones. The PCA-based and no-PCA TSDP models for one-month-ahead runoff forecasting are finally compared.

3.5.4 Experiment 4: Evaluation of the TSDP models for different lead times

For the four experiments, we test the VMD decomposition performance by comparing the prediction outcomes of the VMD-SVR scheme with those of three other TSDP schemes which combine the EEMD, SSA and DWT methods, respectively, with SVR. Meanwhile, the TSDP models were compared with the BCMODWT-SVR realization of the WDDFF framework, which was proposed by Quilty and Adamowski (2018). Additionally, the no-decomposition ARIMA, SVR, BPNN, and LSTM models are compared with TSDP and WDDFF realizations. For each of these data-driven models, the associated hyperparameter settings or search ranges are shown in Table 1. Each hyperparameter is fine-tuned to minimize the mean-square error (MSE). The data-driven model with the lowest MSE is finally selected. The degree of differencing (d) of the ARIMA model is determined by the minimum differencing required to get a stationary time series from the original monthly runoff data. In our work, stationarity testing is performed by the augmented Dickey Fuller (ADF) test (Lopez, 1997).

Table 1 The hyperparameters, tuning strategies, and search ranges for the compared data-driven models.

Data-driven model	Tuning strategy	Hyperparameter	Search space
ARIMA	GS	Degree of differencing (d)	Determined by ADF test
		Autoregressive lags (p)	$[1, 20]$
		Moving-average lags (q)	$[1, 20]$
SVR	BOGP	Weight penalty (C)	$[0.1, 200]$
		Error tolerance (ε)	$[1e-6, 1]$
		Width control coefficient (σ)	$[1e-6, 1]$
BPNN&LSTM	BOGP	Batch size	256
		Optimizer	Adam
		Learning rate	$[1e-4, 1e-1]$
		Activation function	Relu
		Number of hidden layers	$[1, 2]$
		Number of hidden units	$[8, 32]$
		Dropout rate	$[0.1, 0.5]$

The single-hybrid method of the WDDFF framework has shown the best forecasting performance according to Quilty and Adamowski (2018). Therefore, in this work, the WDDFF models were built based on BCMODWT and SVR using the single-hybrid method. In the single-hybrid method, the explanatory variables are decomposed by BCMODWT. The decomposed signal components are selected jointly with the explanatory variables as input predictors. Since our work focuses on time-series forecasting using autoregressive patterns, the explanatory variables are extracted from historical time-series data. Twelve monthly runoff series lagging from one month to twelve months were selected as explanatory variables. Since these series have obvious inter-annual variations, they are also selected as the input predictors for the no-decomposition SVR, BPNN and LSTM models. The BCMODWT-SVR scheme was implemented as follows: (1) select the monthly runoff data ($Q(t+1)$, $Q(t+3)$, $Q(t+5)$, and $Q(t+7)$) as prediction targets and the twelve lagging monthly runoff series ($Q(t-11), Q(t-10), \cdots, Q(t-1), Q(t)$) as explanatory variables; (2) decompose each explanatory variable using the BCMODWT method; (3) combine the explanatory variables and the decomposed components to form the model predictors; (4) select the final input predictors of the BCMODWT-SVR scheme based on the mutual information (MI) criterion (Quilty et al., 2016) (5) train and optimize the SVR model based on the CV strategy and the calibration and development samples; (6) test the optimized BCMODWT-SVR scheme using the test samples.

4 Case study

4.1 Data normalization

To promote faster convergence of the BOGP algorithm, all predictors and prediction targets in this work were normalized to the [-1,1] range by the following equation:

$$\boldsymbol{y} = 2 \otimes \frac{\boldsymbol{x}-x_{min}}{x_{max}-x_{min}}, \tag{13}$$

where $\boldsymbol{x}$ and $\boldsymbol{y}$ are the raw and normalized vectors, respectively, while x_{max} and x_{min} are the maximum and minimum values of $\boldsymbol{x}$, respectively. Also, the multiplication and subtraction are element-wise operations. Note that the parameters x_{max} and x_{min} are computed based on the calibration samples. These parameters are also used to normalize the development and test samples in order to avoid using future information from the development and test phases, and enforce all samples to follow the calibration distribution.

4.2 Model evaluation criteria

For evaluating the forecasting performance, we employed four criteria, namely the Nash–Sutcliffe efficiency (NSE) (Nash and Sutcliffe, 1970), the normalized root-mean-square error (NRMSE), the peak percentage of threshold statistics (PPTS) (Bai et al., 2016; Stojković et al., 2017) and the time cost. The NSE, NRMSE, and PPTS criteria are respectively defined as follows:

$$NSE = 1 - \frac{\sum_{t=1}^{N}\left(x(t)-\hat{x}(t)\right)^2}{\sum_{t=1}^{N}\left(x(t)-\bar{x}(t)\right)^2}, \tag{14}$$

$$NRMSE = \frac{\sqrt{\sum_{t=1}^{N}\left(x(t)-\hat{x}(t)\right)^2/N}}{\sum_{t=1}^{N}x(t)/N}, \tag{15}$$

$$PPTS(\gamma)=\frac{100}{\gamma}\frac{1}{N}\sum_{t=1}^{G}\left|\frac{x(t)-\hat{x}(t)}{x(t)}\right|, \tag{16}$$

where N is the number of samples, and $x(t)$, $\bar{x}(t)$ and $\hat{x}(t)$ are the raw, average, and predicted data samples, respectively. The NSE evaluates the prediction performance of a hydrological model. Larger NSE values reflect more powerful forecasting models. The NRMSE criterion facilitates comparison between datasets or models at different scales. Lower NRMSE values indicate less residual variance. To calculate the PPTS criterion, raw data samples are arranged in descending order and the predicted data samples are arranged following the same order. The parameter γ denotes a threshold level that controls the percentage of the data samples selected from the beginning of the arranged data sequence. The parameter G is the number of values above this threshold level. For example, PPTS(5) means the top 5% flows, or the peak flows, which are evaluated by the PPTS criterion. Lower PPTS values indicate more accurate peak-flow predictions.

4.3 Open-source software and hardware environments

In this work, we utilize multiple open-source software tools. We use Pandas (McKinney, 2010) and Numpy (Stéfan et al., 2011) to perform data preprocessing and management, Scikit-Learn (Pedregosa et al., 2011) to create SVR models for forecasting monthly runoff data and perform PCA-based dimensionality reduction, Tensorflow (Abadi et al., 2016) to build BPNN and LSTM models, Keras-tuner to tune BPNN and LSTM, Scikit-Optimize (Tim et al., 2018) to tune the SVR models, and Matplotlib (Hunter, 2007) to draw the figures. The MATLAB implementations of the EEMD and VMD methods are derived from Wu and Huang (2009) and Dragomiretskiy and Zosso (2014), respectively. The Python-based SSA implementation is adapted from Jordan D'Arcy (2018). The DWT and ARIMA methods were performed based on the MATLAB built-in "Wavelet Analyzer Toolbox" and "Econometric Modeler Toolbox", respectively. As well, Dr. John Quilty of McGill University, Canada, provided the MATLAB implementation of the BCMODWT method. All models were developed and the computational cost of each model was computed based on a 2.50-GHz Intel Core i7-4710MQ CPU with a 32.0 GB of RAM.

4.4 Modeling stages

The VMD-SVR model for one-month-ahead runoff forecasting of the Huaxian station is employed as an example to illustrate the modeling stages of the TSDP, TSDE, WDDFF, and no-decomposition models. As stated in Section 3.1, the decomposition level (K), the quadratic penalty parameter (α), the noise tolerance (τ) and the convergence tolerance (ε) are the four parameters that influence the VMD decomposition performance. In particular, this performance is very sensitive to K (Xu et al., 2019). As suggested by Zuo et al. (2020), the values of α, τ, and ε were set to 2000, 0, and 1e-9, respectively. The optimal K value was determined by checking whether the last IMF had central-frequency aliasing (as represented by the red rectangle area in Fig. 9). Specifically, we increase K starting from $K = 2$ with a step size of 1. If the center-frequency aliasing of the last IMF is first observed when $K = L$, then the optimal K is set to $L - 1$. As shown in Fig. 9, the optimal decomposition level for the Huaxian station is K = 8.

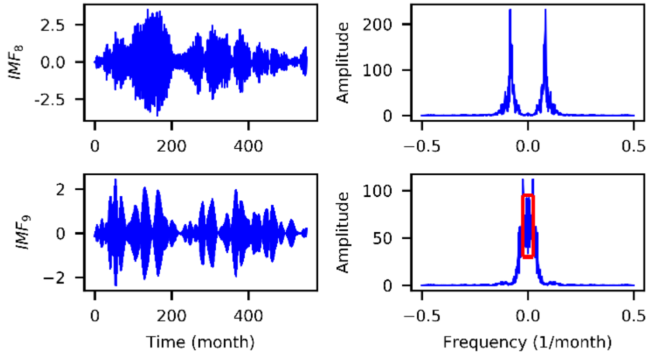


Figure 9: Center-frequency aliasing for the last signal component of time-series data collected from the Huaxian station.

According to the procedure of Section 3.4, PACF is used to determine the optimal predictors for the VMD-SVR scheme. For the time-series data of the Huaxian station, the first VMD IMF component is used as an example of tracking the optimal input predictors from PACF. Figure 10 shows that the PACF of the third lag month exceeds the boundary of the 95% confidence interval (illustrated by the red dashed line) and is insignificant after the third lag month. Thus $x_1(t)$, $x_1(t-1)$ and $x_1(t-2)$ are selected as the optimal input predictors for IMF₁. In such a manner, the input predictors of all signal components are combined together to form the VMD-SVR predictors. Then, the original monthly runoff data, i.e., $Q(t+1)$, is selected as the predicted target.

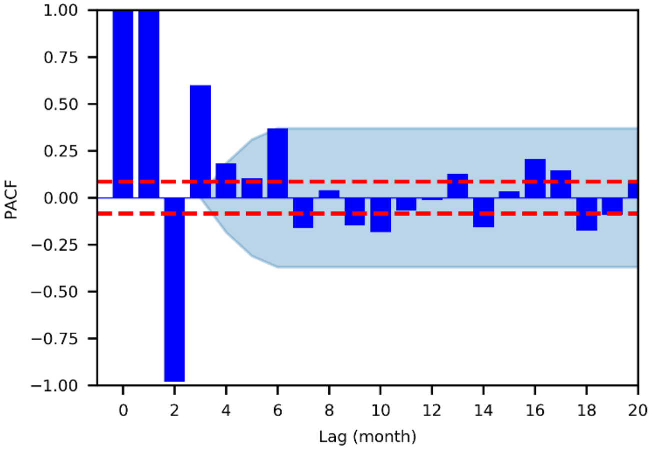


Figure 10: PACF of the first VMD signal component for the time-series data collected from the Huaxian station.

As described in Section 3.2, the VMD-SVR model performance can be optimized by tuning the SVR hyperparameters, namely the weight penalty (C), the error tolerance (ε), and the width control coefficient (σ). To tune these hyperparameters (C, ε, and σ), the maximum number of BOGP iterations was set to 100. The search space of SVR parameters is shown in **Table 1**. Moreover, the CV fold number is a vital parameter that influences the TSDP model performance. In fact, the 10-fold CV and leave-one-out CV (LOOCV) are two frequently-used schemes (Zhang and Yang, 2015; Jung, 2018). Zhang and Yang (2015) show that the LOOCV scheme has a better performance than a 10-fold or a 5-fold CV scheme. However, LOOCV is computationally expensive. Additionally, Hastie et al. (2009) empirically demonstrated that 5-fold CV sometimes has lower variance than LOOCV. Therefore, the selection of the number of CV folds should be made while taking the specific application scenario into consideration. In this work, a 10-fold CV scheme was used for tuning the SVR hyperparameters due to the limited computational resources. We ran the BOGP procedure ten times to reduce the impact of random sampling, and the parameters associated with the lowest MSE on development samples were selected. As shown in Fig. 11 for the time-series data of the Huaxian station, the pairwise partial dependence of the SVR hyperparameters shows that the tuned parameters ($C = 18.97$, $\varepsilon = 1e - 6$ and $\sigma = 0.22$) are globally optimized. This analysis indicates that the BOGP procedure provides reasonable results.

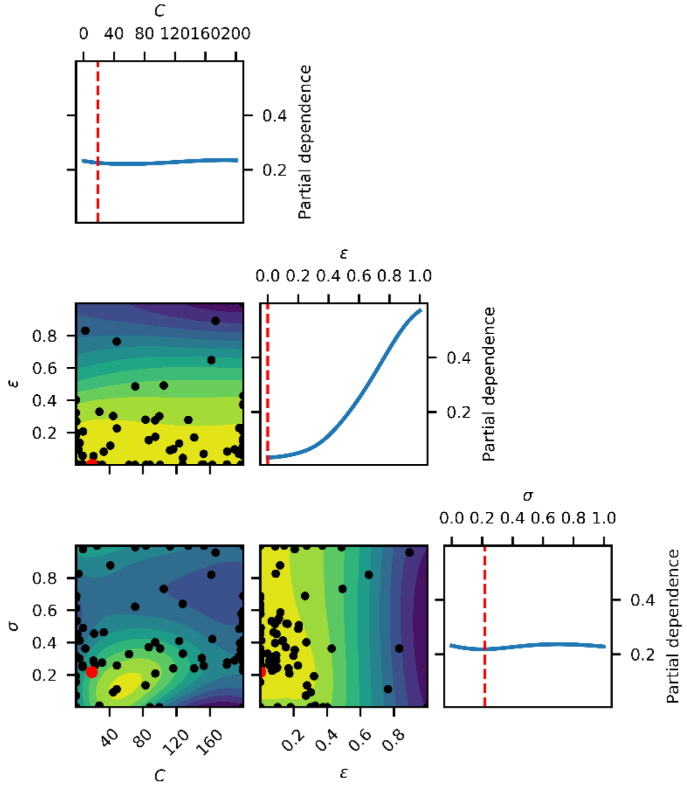


Figure 11: Pairwise partial dependence plot of the MSE objective function for the VMD-SVR scheme based on time-series data of the Huaxian station.

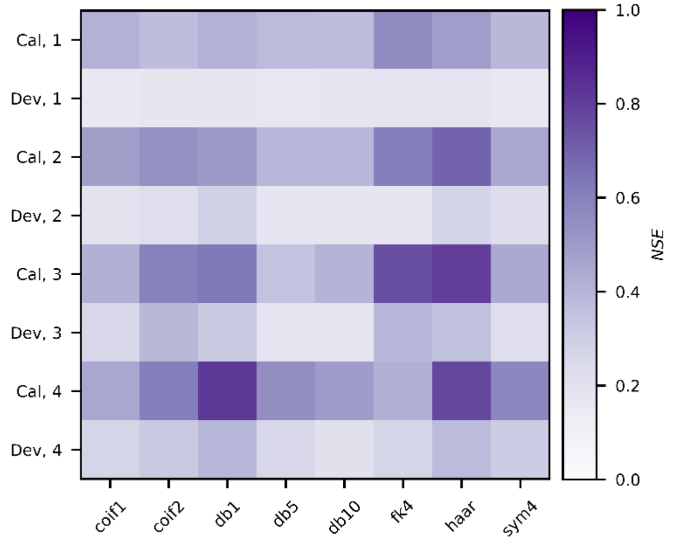


Figure 12: NSE of the BCMODWT-SVR scheme for different wavelet types and decomposition levels. The horizontal axis represents the wavelet types while the two vertical axes respectively represent the decomposition level and the modeling stage (e.g., "Cal, 1" and "Dev, 1" respectively represent the calibration and development stages with a decomposition level of 1).

As stated in Section 3.5.4, the input predictors of the BCMODWT-SVR scheme were generated from the explanatory variables and further filtered by the MI criterion. The input predictors with a MI value larger than 0.1 were retained to train the BCMODWT-SVR scheme. This choice was made since the number of predictors is close to 0 if the MI value is larger than 0.2. Figure 12 shows the NSE values of the BCMODWT-SVR scheme for different wavelets and decomposition levels at the calibration-and-development stage. The *db1* wavelet with a decomposition level of 4 lead to higher calibration and development NSE compared to other combinations of wavelet types and decomposition levels. Therefore, the wavelet type and decomposition level of the BCMODWT-SVR models were set to *db1* and 4, respectively.

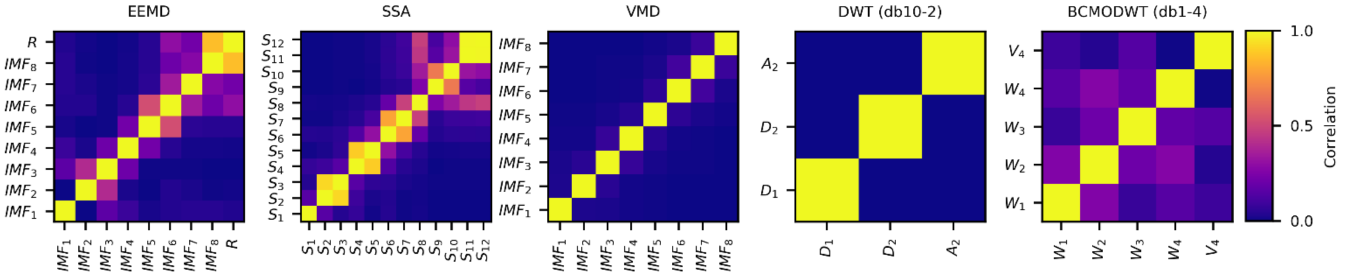


Figure 13: Absolute Pearson correlation coefficients of signal components obtained by different decomposition methods for the time-series data of the Huaxian station.

In this section, we compare the performance of the decomposition algorithms through the analysis of the absolute PCC between each signal component and the original monthly runoff data (see Fig. 13), the frequency spectrum of each signal component (see Fig. 14), and the MI between each predictor and the prediction target (see Fig. 15). The absolute PCCs for only the first explanatory variables of the BCMODWT method are presented in Fig. 13. This figure shows that the coefficients of the EEMD, SSA, and BCMODWT methods are much larger than 0, indicating that most of the signal components of these methods are highly correlated and redundant. The coefficients of the VMD and DWT signal components are less than 0.1 and 0.001, respectively. This indicates that these components are highly uncorrelated. Similar results were obtained for the time-series data of the Xianyang and Zhangjiashan stations. In general, these findings demonstrate that the SVR models established based on the BCMODWT, EEMD, SSA signal components might poorly forecast original monthly runoff data. On the contrary, SVR models based on the DWT and VMD signal components have great potential to accurately forecast monthly runoff data.

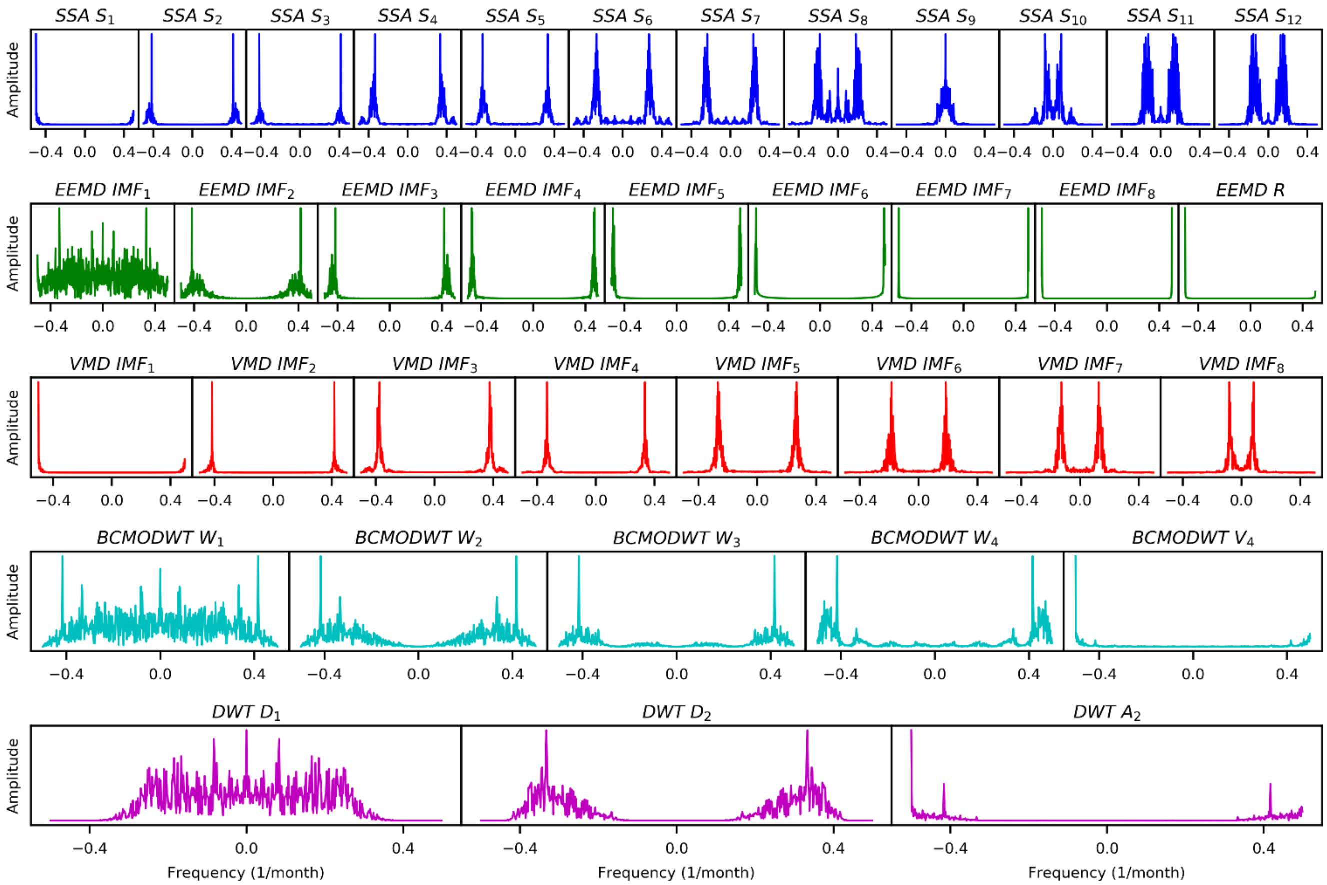


Figure 14: Frequency spectra of the signal components for the time-series data of the Huaxian station. The spectra are shown for the SSA, EEMD, VMD, BCMODWT, and DWT decomposition methods.

Figure 14 shows that the VMD components have a very low noise level around the main frequency. The EEMD *IMF_1* has a large noise level over the entire frequency domain while the EEMD *IMF_2* has large noise levels around the main frequency.

The noise level of SSA S_6-S_{12} around the main frequency is larger than that of the VMD signal components. The DWT D_1 has a large noise level for low frequencies and DWT D_2 has a large noise level around the main frequency. The BCMODWT W_1 has a large noise level over the entire frequency domain and BCMODWT W_2 has a large noise level around the main frequency. These results indicate that (1) the VMD scheme is much more robust to noise than the EEMD, SSA, DWT and BCMODWT schemes, (2) the main components of other schemes (e.g. W_1 and W_2 of BCMODWT, IMF_1 and IMF_2 of EEMD, D_1 and D_2 of DWT, and S_6-S_{12} of SSA) might lead to poor forecasting performance. Similar results were obtained for the time-series data of the Xianyang and Zhangjiashan stations.

Figure 15(d) and (e) show that the DWT and BCMODWT predictors for the one-month-ahead runoff forecast have higher MI values than that for 3-, 5-, and 7-month-ahead forecasts. Figure 15(a)-(c) show that the MI values of the VMD, SSA, and EEMD predictors for the 1-, 3-, 5- and 7-month-ahead runoff forecasts are very close. This indicates that the prediction performance of the DWT-SVR and BCMODWT-SVR schemes for the one-month-ahead runoff forecast may be much better than that for the 3-, 5- and 7-month-ahead runoff forecasts. Also, the results indicate that the prediction performance of the VMD-SVR, SSA-SVR, and EEMD-SVR schemes for all four lead times may not significantly vary. Overall, the findings obtained from Fig. 13-15 show that VMD has the best decomposition performance and a great potential to achieve a good prediction performance.

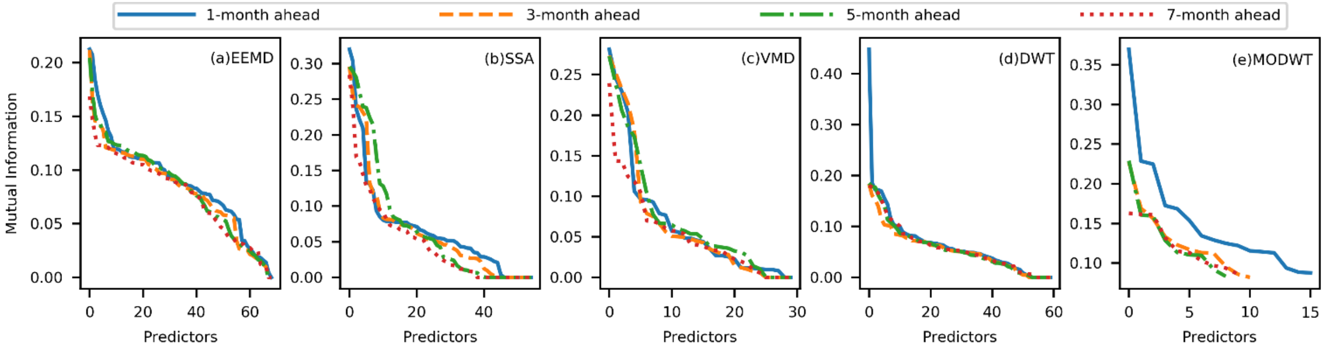


Figure 15: Mutual information between each predictor and the predicted target for the time-series data of the Huaxian station.

5 Experimental Results

5.1 Reduction of boundary effects in the TSDP models

An experimental comparison of TSDP models established with and without the appended decompositions and the mixing-and-shuffling step is illustrated in Fig. 16. Figure 16(a) shows that the calibration and development NSE values of the scheme 1 are very close but larger than the test NSE value. This indicates that the optimized model based on samples generated without the appended decompositions and the mixing-and-shuffling step approximates the calibration distribution reasonably well, though this model poorly generalizes to the test distribution. Figure 16(b) shows that the NSE interquartile range decreased substantially compared to the test NSE of the scheme 1. Also, the NSE mean value increased considerably except for the EEMD-SVR scheme. This demonstrates the importance of generating test samples from appended decompositions in order to improve the prediction performance on the test samples. As well, Fig. 16(c) shows that the NSE interquartile range increased substantially compared to the NSE of the scheme 1, while the NSE mean decreased considerably. This demonstrates that the mixing-and-shuffling step does not improve the generalization performance if the validation samples are not generated from appended decompositions. Moreover, Fig. 16(d) shows that the NSE interquartile range decreased substantially in comparison with the NSE of the scheme 3, while the NSE mean increased considerably. These results also demonstrate the importance of generating validation samples from appended decompositions in order to improve the TSDP generalization capability. Figure 16(d) also shows that the NSE interquartile range decreased substantially compared with the test NSE of the scheme 2, while the NSE mean increased considerably. This demonstrates the importance of the mixing-and-shuffling step in improving the prediction performance on test samples under the condition that the validation samples are generated from appended decompositions. Similar results were obtained for the NRMSE and PPTS criteria. In general, generating validation samples from appended decompositions, and also mixing and shuffling the calibration and development samples help a lot with boosting prediction performance. Nevertheless, generating samples from appended decompositions is more important than the mixing-and-shuffling step for reducing the boundary effect consequences.

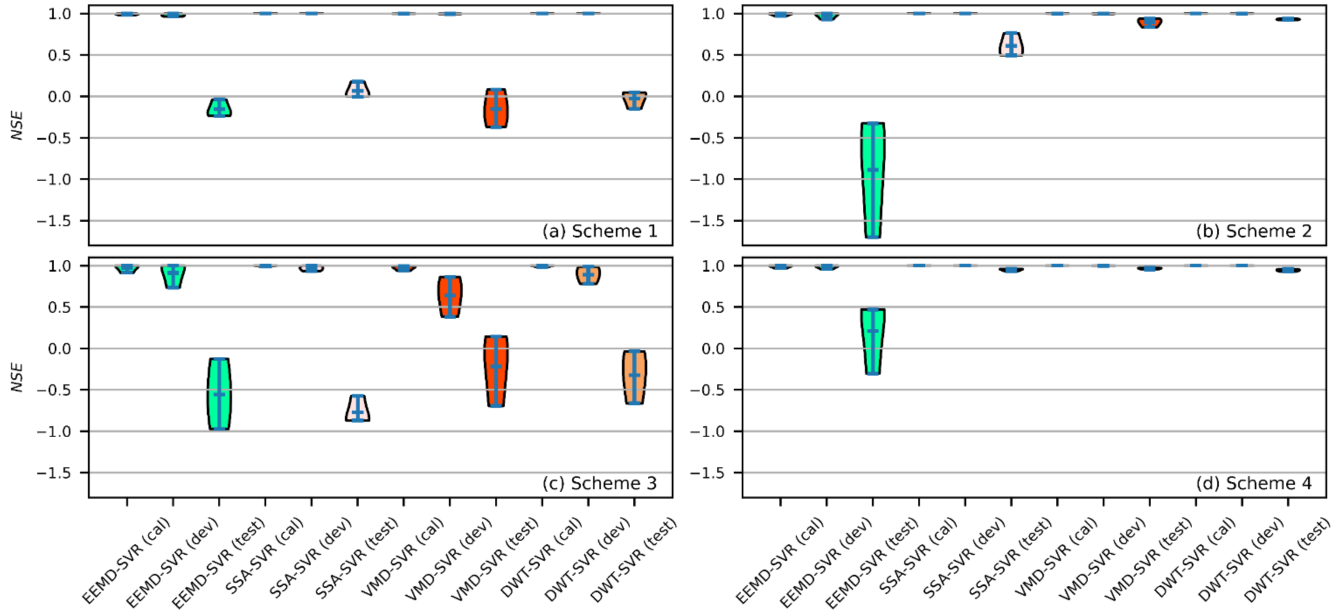


Figure 16: Violin plots of the NSE criterion for TSDP models and one-month-ahead runoff forecasting (See Fig. 8 for the details of each scheme).

5.2 Performance gap between the TSDP and TSDE models

The performance gap between the TSDP and TSDE models is illustrated in Fig. 17. Figure 17(a), (b) and (c) show that the mean NSE for the DWT-SVR-A scheme is larger than that of the DWT-SVR one, while the mean NRMSE and PPTS values of DWT-SVR-A are smaller than that of DWT-SVR. The DWT-SVR-A scheme also has smaller NSE, NRMSE and PPTS interquartile ranges than those of the DWT-SVR one. This indicates that the DWT-SVR scheme does not improve prediction performance in comparison to the DWT-SVR-A one. Similar results and conclusions were obtained for the EEMD-SVR and EEMD-SVR-A schemes. Figure 17 (a), (b) and (c) also show that the mean NSE of the VMD-SVR scheme is larger than that of the VMD-SVR-A one, while the mean NRMSE and PPTS values of VMD-SVR are smaller than those of VMD-SVR-A. The NSE, NRMSE and PPTS interquartile ranges of VMD-SVR are smaller than those of VMD-SVR-A. This shows that VMD-SVR improves prediction performance compared with VMD-SVR-A. Similar results and conclusions were obtained for SSA-SVR and SSA-SVR-A. Figure 17(d) shows that the computational cost of the TSDE models is much larger than that of the TSDP models, and that the computational cost of the TSDE models is positively correlated to the decomposition level. Overall, these findings demonstrated that the TSDP models do not always improve the prediction performance but are generally of smaller computational cost compared to the TSDE models.

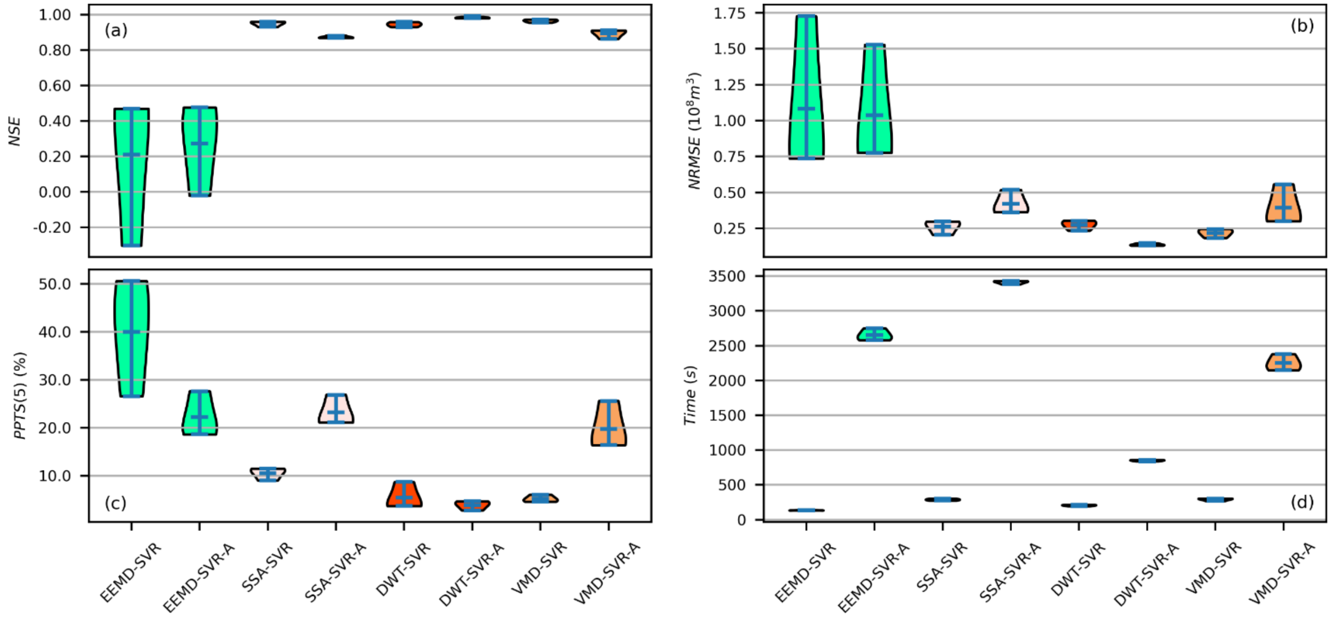


Figure 17: Violin plots of the evaluation criteria for one-month-ahead runoff forecasting during the test phase of the TSDP and TSDE models.

5.3 Effect of dimensionality reduction on the TSDP models

The violin plots of NSE values for different (guessed) numbers of excluded predictors and all three data collection stations are illustrated in Fig. 18. Figure 18(a) and (b) show that dimensionality reduction generally reduces the NSE scores of the EEMD-

SVR and SSA-SVR schemes. This indicates that dimensionality reduction causes these schemes to lose some valuable information. Figure 18(c) shows that the NSE scores of the DWT-SVR scheme are slightly larger than the mean NSE without PCA. Figure 18(d) shows that the NSE scores of the VMD-SVR scheme are slightly larger than the mean NSE without PCA when the number of excluded predictors is less than 8. The NSE score generally decreased as the number of excluded predictors is increased from 0 to 16. These results demonstrate that the DWT-SVR and VMD-SVR schemes have overfitting to some extent, and the predictors of these schemes are slightly linearly correlated. Figure 18 shows that the associated NSE scores of the guessed number of excluded predictors for the EEMD-SVR and SSA-SVR schemes are smaller than the mean NSE score without PCA. On the contrary, the corresponding NSE scores for the DWT-SVR and VMD-SVR schemes are slightly larger than the mean NSE score without PCA This indicates that the guessed number of principal components obtained by the MLE method reduces the prediction performance of the EEMD-SVR and SSA-SVR schemes but slightly improves the performance for the DWT-SVR and VMD-SVR schemes. In fact, we chose not to perform the dimensionality reduction on the subsequent TSDP models to avoid the risk of information loss.

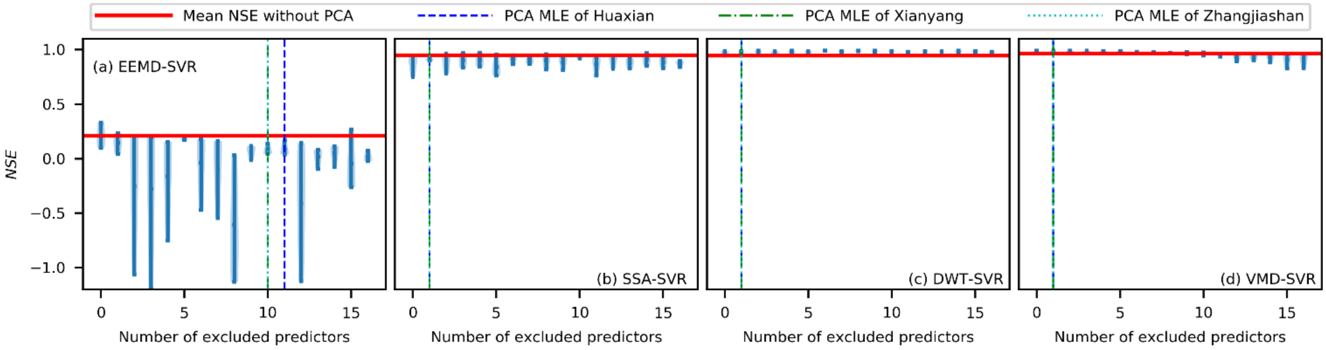


Figure 18: Violin plots of the NSE values for different numbers of excluded components and one-month-ahead runoff forecasting.

5.4 Performance of the TSDP models for different lead times

Figure 19 shows that the correlation values of the VMD-SVR scheme for 1-, 3-, 5- and 7-month-ahead runoff forecasting are concentrated around the ideal fit, with a small angle between the ideal and linear fits. This indicates that the raw measurements and the VMD-SVR predictions have a high degree of agreement. Also, the DWT-SVR correlation values are concentrated around the ideal fit with a small angle between the ideal and linear fits for forecasting runoff data one month ahead (see Fig. 19a). However, the correlation values are dispersed around the ideal fit with a large angle between the ideal and linear fits for forecasting runoff 3, 5, and 7 months ahead (see Fig. 19b, c and d). This indicates that the DWT-SVR model has good prediction performance for forecasting runoff data one month ahead but not for 3, 5 and 7 months ahead. While similar results can be observed for SSA-SVR, the correlation values of this scheme are less concentrated for forecasting runoff one month ahead and more concentrated for forecasting runoff 3, 5 and 7 months ahead in comparison to the DWT-SVR correlation values. This demonstrates that DWT-SVR is better than SSA-SVR in 1-month-ahead prediction but worse in 3-, 5-, and 7-month-ahead prediction. Figure 19 also shows that the correlation values of the EEMD-SVR and BCMODWT-SVR schemes are less concentrated than those of the VMD-SVR, DWT-SVR and SSA-SVR schemes for forecasting runoff one month ahead, and also less concentrated than those of the VMD-SVR and SSA-SVR schemes for forecasting runoff 3, 5 and 7 months ahead. This demonstrates that the EEMD and BCMODWT methods have poor prediction performance for all lead times. As shown in Fig. 19(a), the correlation values of the EEMD-SVR model are more concentrated than those of the ARIMA, SVR, BPNN and LSTM models, and the angle between the ideal and linear fits of BCMODWT-SVR is larger than that of the ARIMA, SVR, BPNN and LSTM models. This indicates that the decomposition of the original monthly runoff data cannot always help improve the prediction performance. As shown in Fig. 19, similar results were obtained for the time-series data of the Xianyang and Zhangjiashan stations.

Quantitative evaluation results are presented in Fig. 20. Compared with the SVR, BPNN and LSTM models, the ARIMA models have larger mean NSE, and smaller mean NRMSE and mean PPTS. This indicates that the ARIMA models have better prediction performance than the SVR, BPNN and LSTM ones. The VMD-SVR scheme is the only scheme with a mean NSE exceeding 0.8 for all three stations and four lead times. This NSE value is often taken as a threshold value for reasonably well-performing models (Newman et al., 2015). This result indicates that the measurements are reasonably matched by the VMD-SVR predictions. Compared with the no-decomposition ARIMA model for forecasting runoff data one month ahead, the mean NSE values of VMD-SVR for forecasting runoff data 1, 3, 5 and 7 months ahead are respectively increased by 139%, 135%, 134%, and 132%. For the SSA-SVR scheme, the corresponding increases are 136%, 103%, 101% and 104%, respectively. For the DWT-SVR scheme, the mean NSE values respectively increased by 134%, 2%, -71% and -93%. For the EEMD-SVR scheme, the respective decrements are -48%, -55%, -88% and -125%. For BCMODWT-SVR, the respective changes are -51%, -90%, -79% and -84%. These findings indicate that (1) VMD-SVR and SSA-SVR play a positive role while EEMD-SVR and BCMODWT-SVR play a negative role in improving the prediction performance of decomposition-based models for all lead times; (2) DWT-SVR has a positive impact on the prediction performance for forecasting runoff 1 and 3 months ahead but a negative impact on the prediction performance for forecasting runoff 5 and 7 months ahead; (3) as the lead time increased, the VMD-SVR prediction performance slightly decreased, the SSA-SVR and BCMODWT-SVR prediction performance slowly decreased, while the prediction performance of DWT-SVR and EEMD-SVR dramatically decreased. Indeed, the overall performance is ranked from the highest to the lowest as follows: VMD-SVR>SSA-SVR>DWT-SVR>EEMD-SVR≈BCMODWT-SVR. Additionally, the VMD-SVR scheme generally has a smaller interquartile range and a good generalization capability for different watersheds. Similar results were obtained for the NRMSE and PPTS criteria (as shown in Fig. 20b and c). Overall, the results obtained from Fig. 19 and 20 demonstrate that the proposed VMD-SVR scheme has the best prediction performance as well as satisfactory generalization capabilities for different data collection stations and lead times. The results also show that the BCMODWT-SVR scheme may not be feasible for our case study.

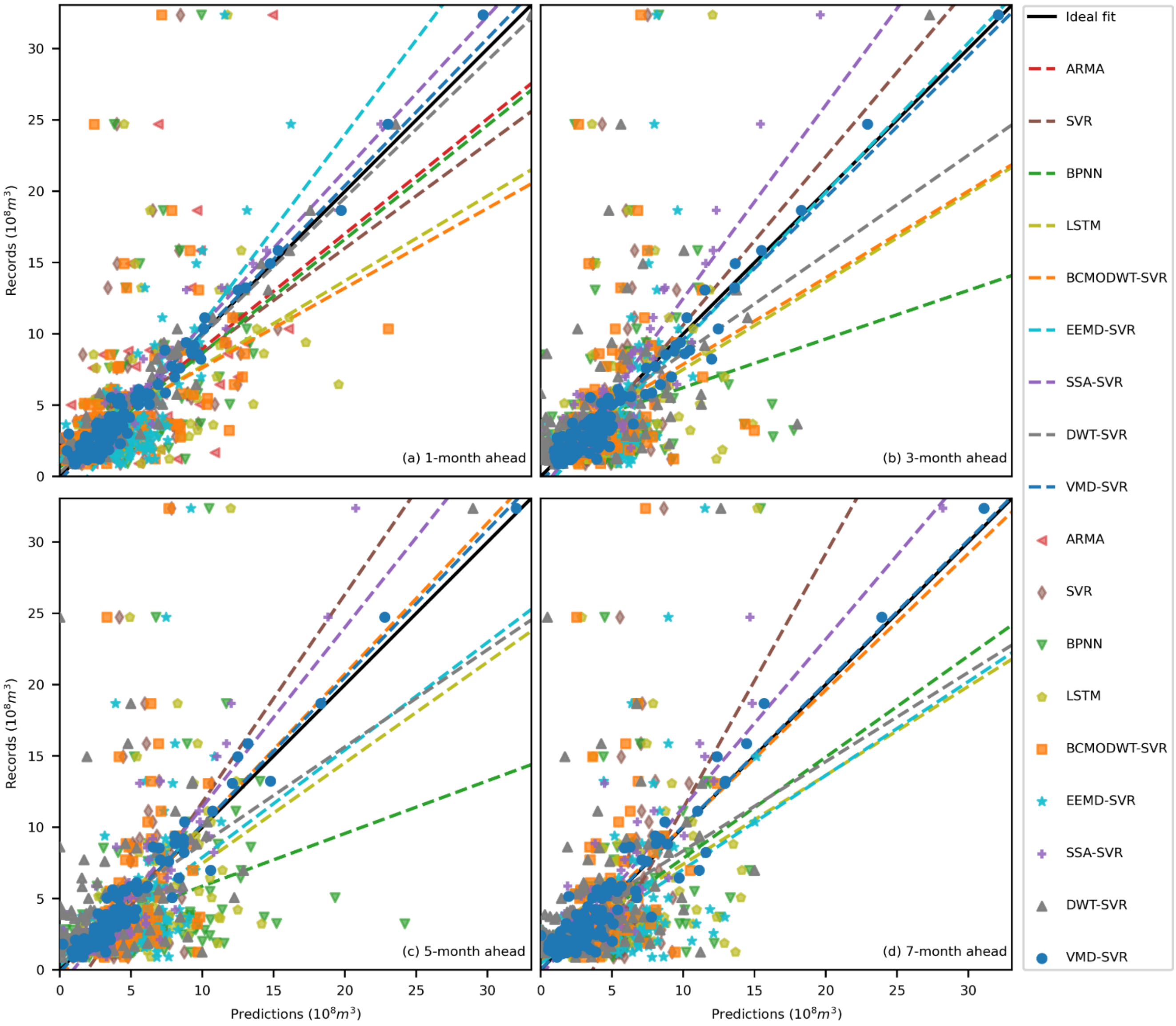


Figure 19: Scatter plots of the TSDP and benchmark models during the test phase for forecasting runoff (a)1 month ahead, (b) 3 months ahead, (c) 5 months ahead and (d) 7 months ahead at the Huaxian station.

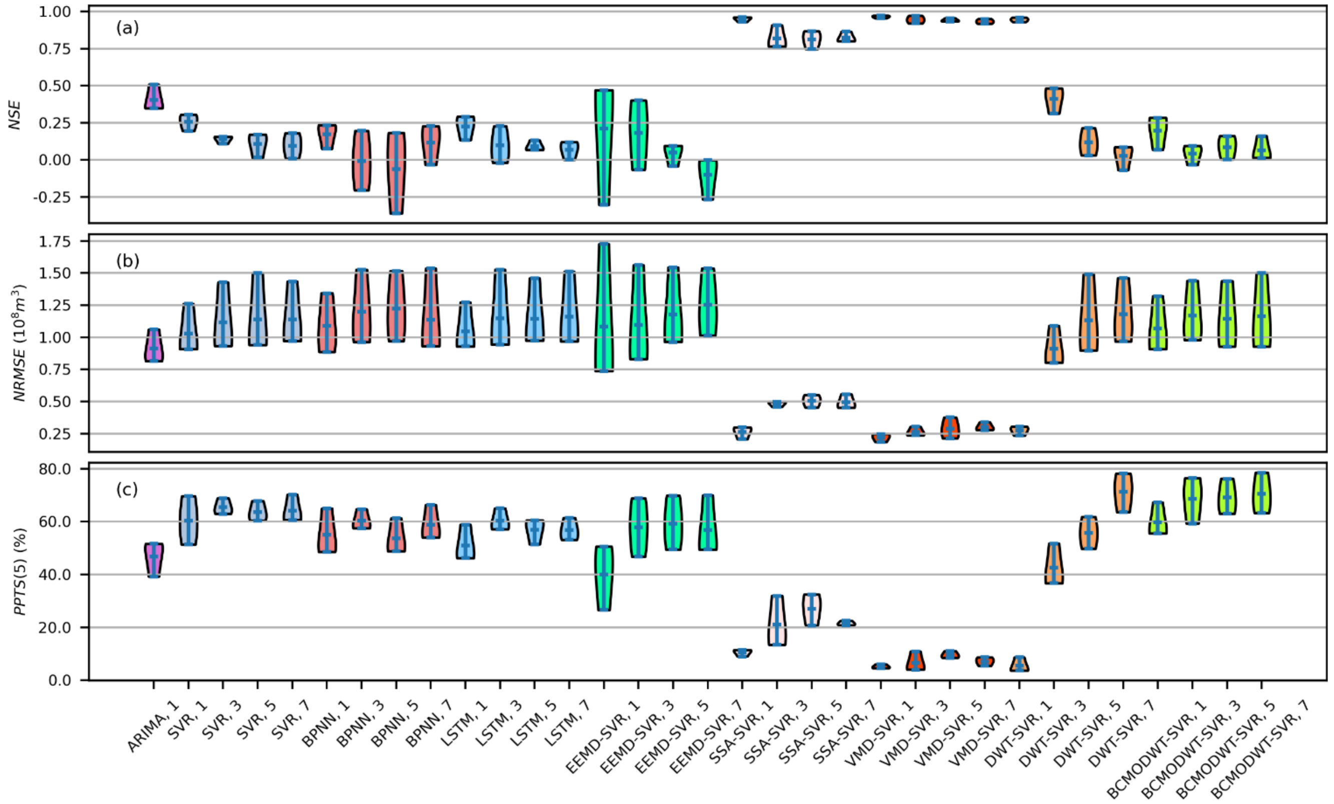


Figure 20: Violin plots of the evaluation criteria during testing for the TSDP and benchmark models (the horizontal axes represent the model and lead time, e.g., "VMD-SVR, 1" represents the VMD-SVR model for 1-month-ahead runoff forecasting).

6 Discussion

As we can see from the experimental results of Section 5, the designed TSDP framework and its VMD-SVR realization attain the aforementioned desirable goals (see Section 1). We now discuss why and how the TSDP framework and its VMD-SVR realization are superior to other decomposition-based streamflow forecasting frameworks and models.

The results in Section 5.1 show that generating samples from appended decompositions, as well as mixing and shuffling the calibration and development samples improve the prediction performance on test samples (see Fig. 16). The calibration and the validation samples have quite different error distributions due to boundary effects (see Fig. 4). The predictors of validation samples generated from appended decompositions are more correlated to the predicted targets than the predictors of validation samples directly generated from validation decompositions (see Fig. 5). Therefore, generating validation samples from appended decompositions helps the TSDP framework improve its generalization capability. Mixing and shuffling the calibration and development samples and training SVR model based on a CV strategy using the mixed and shuffled samples enable the assessment of the validation distribution during calibration with no test information. In other words, the SVR models can be calibrated and validated on the calibration and validation distributions simultaneously. Therefore, the mixing-and-shuffling step helps the TSDP framework enhance its generalization capability. Nevertheless, this step does not help much if the validation samples are generated from validation decompositions (see Fig.16). This because the relationship between predictors and prediction targets presented in validation samples is changed a lot compared to that of calibration samples. However, one may sequentially append the calibration set to the first streamflow data samples and decompose the appended sets to force the calibration and validation samples to follow approximately the same distribution. We refrain from doing this (and strongly advise against it) because the modeling process will become quite laborious and large decomposition errors will be also introduced into the calibration samples.

The experimental outcomes of Section 5.2 indicate that the TSDP framework saves modeling time and sometimes improves the prediction performance compared to the TSDE framework (see Fig. 17). This improvement can be ascribed to the fact that the TSDP models avoid the error accumulation problem and also simulate the relationship between signal components and the original monthly runoff data as well as the relationship between the predictors and the predicted target. This simulation improves the prediction performance because the summation of the signal components (summation is the ensemble strategy used by TSDE) obtained by some decomposition algorithms cannot precisely reconstruct the original monthly runoff data (e.g., VMD in this work, see Fig. 3h). However, the TSDP framework accounts for the noise when the predictors are fused. Therefore, the TSDP framework might be outperformed by the TSDE framework if some signal components have a large noise level. The DWT-SVR and EEMD-SVR schemes do not improve the performance considerably compared with the DWT-SVR-A and EEMD-SVR-A schemes (see Fig.17) since the respective main decomposition components (i.e., DWT D_1 and D_2, and EEMD IMF_1 and IMF_2) have large noise levels (see Fig.14). However, compared with the EEMD-SVR and EEMD-SVR-A schemes, the performance gap between DWT-SVR and DWT-SVR-A is quite small (see Fig.17) because the DWT has fewer signal components which are more uncorrelated than the EEMD signal components (see Fig 13). Overall, we still suggest using the DWT-SVR scheme rather than the DWT-SVR-A one to predict runoff one month ahead and save modeling time.

The results of Section 5.3 indicate that combining the predictors of the individual signal components causes overfitting in the VMD-SVR and DWT-SVR schemes but does not overfit the EEMD-SVR and SSA-SVR schemes at all (see Fig. 18). The reason is that the predictors and prediction targets come from the same source (the monthly runoff data in this work) and the TSDP models simulate the relationship inside the original monthly runoff data rather than the relationship between the precipitation, evaporation, temperature, and monthly runoff data. Therefore, the TSDP models focus on simulating the relationship between historical and future monthly runoff data rather than fitting noise (random sampling error). Although the predictors of the VMD-SVR and DWT-SVR schemes are slightly correlated, the prediction performance of these schemes for one-month-ahead forecasting is already good enough and the dimensionality reduction improves the prediction performance a little bit. Therefore, we suggest predicting the original streamflow directly based on the proposed TSDP framework (see Section 3.4 and Fig. 6) without dimensionality reduction in the autoregression cases. However, Noori et al. (2011) have demonstrated that, compared with the no-PCA SVR model, PCA enhances considerably the prediction performance for the monthly runoff with rainfall, temperature, solar radiation, and discharge. Therefore, performing PCA on the TSDP framework is necessary if the predictors come from different sources.

The experimental outcomes of Section 5.4 indicate that the VMD-SVR scheme has the best performance (see Fig. 19 and 20). This validates the guess we made in Section 4.4. This performance improvement is due to the fact that the VMD signal components are barely correlated (see Fig. 13) and have a low noise level (see Fig. 14). Determining the VMD decomposition level by observing the center-frequency aliasing (see Fig. 9) helps avoid mode mixing, and hence leads to uncorrelated signal components. Setting the VMD noise tolerance (τ) to 0 removes some noise components inside the original monthly runoff data (see Section 3.1), and thus allows signal components with a low noise level. Although setting the noise tolerance to 0 does not enable the summation of the VMD signal components for the original streamflow reconstruction (see Fig. 3h), the TSDP framework perfectly solves this problem by building a single SVR model to predict the original streamflow instead of summing the predictions of all signal components. Also, results from Section 5.4 show that DWT-SVR exhibits prediction performance that is better than that of SSA-SVR for one-month-ahead runoff forecasting but worse than that of SSA-SVR for 3-, 5- and 7-month-ahead runoff forecasting (see Fig. 19 and 20). Once again, this result verifies the guess we gave in Section 4.4. This is because, in comparison with SSA, the DWT predictors for one-month-ahead runoff forecasting have higher MI than those for 3-, 5-, and 7-month-ahead runoff forecasting (see Fig. 15). The SSA-SVR scheme shows prediction performance that is inferior to that of VMD-SVR, but shows better prediction performance compared to other models. These outcomes are due to the fact that the SSA signal components are correlated (see Fig. 13) and have a larger noise level than VMD but a lower noise level than that of the EEMD, DWT, BCMODWT signal components (see Fig. 14). The EEMD-SVR poor prediction performance (see Fig.19 and 20) is because of the EEMD limitations such as sensitivity to noise and sampling (Dragomiretskiy and Zosso, 2014). These limitations lead to large-noise EEMD components IMF_1 and IMF_2 (see Fig. 14) with component correlation, redundancy, and chaotically represented trend, period and noise terms (see Fig. 13). The BCMODWT-SVR scheme failed to provide reasonable forecasting performance due to: (1) the limited sample size (only 792 data points in the original monthly runoff data), of which the wavelet and scaling coefficients are further removed by the BCMODWT method, (2) the limited information explained by the explanatory variables of the original monthly runoff, where the PACF is very small after the first lag month, (3) the correlated BCMODWT signal components (see Fig. 13), and (4) the large-noise BCMODWT components W_1 and W_2 (see Fig. 14). Therefore, the WDDFF realization, i.e., BCMODWT-SVR, may not be feasible for our problem. Additionally, the ARIMA models have better performance than the SVR, BPNN and LSTM models but worse performance than the VMD-SVR and SSA-SVR models. This performance is likely because the ARIMA models automatically determine the p and q in the range [1,20] to find more useful historical information for explaining the monthly runoff data. However, signal components with different frequencies extracted from VMD and SSA explain more information inside the original monthly runoff data. Overall, the VMD method is more robust to sampling and noise, and is therefore recommended for performing monthly runoff forecasting in autoregressive scenarios.

In summary, the major contribution of this work is the development of a new feasible and accurate approach for dealing with boundary effects in streamflow time-series analysis. Previous approaches handled the boundary effects by removing or correcting the boundary-affected decompositions (Quilty and Adamowski, 2018; Zhang et al., 2015) or adjusting the model parameters as new data is added (Tan et al., 2018). However, to the best of our knowledge, no approaches have been successfully applied in building a forecasting framework that can adapt to boundary effects without removing or correcting boundary-affected decompositions, while providing users with a high confidence level on unseen data. Indeed, this work focuses on exploiting rather than correcting or eliminating boundary-affected decompositions, in order to develop an effective, efficient, and accurate decomposition-based forecasting framework. Note that we do not need a lot of prior experience with signal processing algorithms or mathematical methods for correcting boundary deviations. We just enforce the models to assess the validation distribution during the calibration phase, and ensure proper handling of the validation decomposition errors. Overall, this operational streamflow forecasting framework is quite simple and easy to implement.

7 Conclusions

This work investigated the potential of the proposed TSDP framework and its VMD-SVR realization for forecasting runoff data in basins lacking meteorological observations. The TSDP decomposition stage extracts hidden information of the original data and avoids using validation information that is not available in practical forecasting applications. The TSDP prediction stage reduces boundary effects, saves modeling time, avoids error accumulation, and possibly improves prediction performance. With four experiments, we explored the reduction in boundary effects, computational cost, overfitting, as well as decomposition and forecasting outcomes for different lead times. We demonstrated that the TSDP framework with its VMD-SVR realization can simulate monthly runoff data with competitive performance outcomes compared to reference models. With the first experiment, we evaluated the reduction of the boundary effects in the TSDP framework. In the second experiment, we assessed the performance gap between the TSDP and TSDE models. For the third experiment, we empirically tested overfitting in TSDP models. Additionally, we evaluated the prediction performance of the TSDP models for different lead times in the fourth and last experiment.

In summary, the major conclusions of this work are as follows:

a. Generating validation samples with appended decompositions, as well as mixing and shuffling the calibration and development samples, can significantly reduce the ramifications of boundary effects.

b. The TSDP framework saves modeling time and sometimes improves the prediction performance compared to the TSDE framework.

c. Combining the predictors of all signal components as the ultimate predictors does not overfit the EEMD-SVR and SSA-SVR models and barely overfits the VMD-SVR and DWT-SVR models. Although some overfitting of the VMD-SVR and DWT-SVR occurs, these models still provide accurate out-of-sample forecasts.

d. The VMD-SVR scheme with NSE scores clearly exceeding 0.8 possesses the best forecasting performance for all forecasting scenarios. The BCMODWT-SVR scheme may not be feasible for autoregressive monthly runoff data modeling.

The boundary effects represent a potential barrier for practical streamflow forecasting. We do believe that generating samples from appended decompositions, in addition to mixing and shuffling the calibration and development samples, are promising ways to reduce the influences of boundary effects and improve the prediction performance on monthly runoff future test samples. Ultimately, however, the black-box nature of the TSDP framework and the VMD-SVR model (or any data-driven model) is a justifiable barrier of making decisions in water resource management using the prediction results. Further research is needed on the VMD-SVR result interpretability.

Code/Data availability. The data and codes that support the findings of this study are available on https://github.com/zjy8006/MonthlyRunoffForecastByAutoReg (last access: 24 August 2020, http://dx.doi.org/10.17632/ybfvpgvvsj.4, Ganggang Zuo, 2020). The remaining data are available upon request.

Author Contributions. Ganggang Zuo and Ni Wang designed all the experiments. Yani Lian and Xinxin He collected and preprocessed the data. Ganggang Zuo and Yani Lian conducted all the experiments and analyzed the results. Ganggang Zuo wrote the first draft of the manuscript with contributions from Yani Lian and Xinxin He. Ni Wang and Jungang Luo supervised the study and edited the manuscript.

Conflicts of Interest. Declarations of interest: none.

Acknowledgments. We would like to thank Dimitri Solomatine for handling this paper, John Quilty and one anonymous referee for providing insightful and constructive comments, which have been a great help in improving the quality of this paper. We also wish to thank John Quilty for providing the MatLab implementation of BCMODWT and the R script for testing VMD shift-variance and sensitivity of the addition of new data.

Financial support. This work was supported by the National Natural Science Foundation of China (Grant Nos. 51679186, 51679188, 51979221 and 51709222), the National Key R&D Program of China (Grant No. 2016YFC0401409), and the Research Fund of the State Key Laboratory of Eco-hydraulics in Northwest Arid Region, Xi’an University of Technology (Grant No. 2019KJCXTD-5).

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mane, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viegas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., and Zheng, X.: TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems, 2016.

Abbott, M. B., Bathurst, J. C., Cunge, J. A., O'Connell, P. E., and Rasmussen, J.: An introduction to the European Hydrological System — Systeme Hydrologique Europeen, “SHE”, 1: History and philosophy of a physically-based, distributed modelling system, Journal of Hydrology, 87, 45–59, doi:10.1016/0022-1694(86)90114-9, 1986.

Adamowski, J. and Sun, K.: Development of a coupled wavelet transform and neural network method for flow forecasting of non-perennial rivers in semi-arid watersheds, Journal of Hydrology, 390, 85–91, doi:10.1016/j.jhydrol.2010.06.033, 2010.

Ashrafi, M., Chua, L. H. C., Quek, C., and Qin, X.: A fully-online Neuro-Fuzzy model for flow forecasting in basins with limited data, Journal of Hydrology, 545, 424–435, doi:10.1016/j.jhydrol.2016.11.057, 2017.

Bai, Y., Chen, Z., Xie, J., and Li, C.: Daily reservoir inflow forecasting using multiscale deep feature learning with hybrid models, Journal of Hydrology, 532, 193–206, doi:10.1016/j.jhydrol.2015.11.011, 2016.

Beven, K.: Changing ideas in hydrology — The case of physically-based models, Journal of Hydrology, 105, 157–172, doi:10.1016/0022-1694(89)90101-7, 1989.

Binley, A. M., Beven, K. J., Calver, A., and Watts, L. G.: Changing responses in hydrology: Assessing the uncertainty in physically based model predictions, Water Resour. Res., 27, 1253–1261, doi:10.1029/91WR00130, 1991.

Castellano-Méndez, M., González-Manteiga, W., Febrero-Bande, M., Manuel Prada-Sánchez, J., and Lozano-Calderón, R.: Modelling of the monthly and daily behaviour of the runoff of the Xallas river using Box–Jenkins and neural networks methods, Journal of Hydrology, 296, 38–58, doi:10.1016/j.jhydrol.2004.03.011, 2004.

Clark, M. P., Nijssen, B., Lundquist, J. D., Kavetski, D., Rupp, D. E., Woods, R. A., Freer, J. E., Gutmann, E. D., Wood, A. W., Brekke, L. D., Arnold, J. R., Gochis, D. J., and Rasmussen, R. M.: A unified approach for process-based hydrologic modeling: 1. Modeling concept, Water Resour. Res., 51, 2498–2514, doi:10.1002/2015WR017198, 2015.

Devia, G. K., Ganasri, B. P., and Dwarakish, G. S.: A Review on Hydrological Models, Aquatic Procedia, 4, 1001–1007, doi:10.1016/j.aqpro.2015.02.126, 2015.

Dragomiretskiy, K. and Zosso, D.: Variational Mode Decomposition, IEEE Trans. Signal Process., 62, 531–544, doi:10.1109/TSP.2013.2288675, 2014.

Du, K., Zhao, Y., and Lei, J.: The incorrect usage of singular spectral analysis and discrete wavelet transform in hybrid models to predict hydrological time series, Journal of Hydrology, 552, 44–51, doi:10.1016/j.jhydrol.2017.06.019, 2017.

Erdal, H. I. and Karakurt, O.: Advancing monthly streamflow prediction accuracy of CART models using ensemble learning paradigms, Journal of Hydrology, 477, 119–128, doi:10.1016/j.jhydrol.2012.11.015, 2013.

Fang, W., Huang, S., Ren, K., Huang, Q., Huang, G., Cheng, G., and Li, K.: Examining the applicability of different sampling techniques in the development of decomposition-based streamflow forecasting models, Journal of Hydrology, 568, 534–550, doi:10.1016/j.jhydrol.2018.11.020, 2019.

Gai, L., Nunes, J. P., Baartman, J. E.M., Zhang, H., Wang, F., Roo, A. de, Ritsema, C. J., and Geissen, V.: Assessing the impact of human interventions on floods and low flows in the Wei River Basin in China using the LISFLOOD model, Science of The Total Environment, 653, 1077–1094, doi:10.1016/j.scitotenv.2018.10.379, 2019.

Grayson, R. B., Moore, I. D., and McMahon, T. A.: Physically based hydrologic modeling: 2. Is the concept realistic?, Water Resour. Res., 28, 2659–2666, doi:10.1029/92WR01259, 1992.

Han, D., Cluckie, I. D., Karbassioun, D., Lawry, J., and Krauskopf, B.: River Flow Modelling Using Fuzzy Decision Trees, Water Resources Management, 16, 431–445, doi:10.1023/A:1022251422280, 2002.

Hastie, T., Friedman, J., and Tibshirani, R.: The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Second, Springer Series in Statistics, Springer-Verlag New York, New York, NY, Online-Ressource, 2009.

He, X., Luo, J., Li, P., Zuo, G., and Xie, J.: A Hybrid Model Based on Variational Mode Decomposition and Gradient Boosting Regression Tree for Monthly Runoff Forecasting, Water Resour Manage, 34, 865–884, doi:10.1007/s11269-020-02483-x, 2020.

He, X., Luo, J., Zuo, G., and Xie, J.: Daily Runoff Forecasting Using a Hybrid Model Based on Variational Mode Decomposition and Deep Neural Networks, Water Resour Manage, 33, 1571–1590, doi:10.1007/s11269-019-2183-x, 2019.

He, Z., Wen, X., Liu, H., and Du, J.: A comparative study of artificial neural network, adaptive neuro fuzzy inference system and support vector machine for forecasting river flow in the semiarid mountain region, Journal of Hydrology, 509, 379–386, doi:10.1016/j.jhydrol.2013.11.054, 2014.

Hosseini, S. M. and Mahjouri, N.: Integrating Support Vector Regression and a geomorphologic Artificial Neural Network for daily rainfall-runoff modeling, Applied Soft Computing, 38, 329–345, doi:10.1016/j.asoc.2015.09.049, 2016.

Huang, S., Chang, J., Huang, Q., and Chen, Y.: Monthly streamflow prediction using modified EMD-based support vector machine, Journal of Hydrology, 511, 764–775, doi:10.1016/j.jhydrol.2014.01.062, 2014.

Hunter, J. D.: Matplotlib: A 2D Graphics Environment, Computing in Science & Engineering, 9, 90–95, doi:10.1109/MCSE.2007.55, 2007.

James, B., Bardenet, R., Bengio, Y., and Balázs Kégl (Eds.): Algorithms for Hyper-Parameter Optimization, 2011.

Jiang, R., Wang, Y., Xie, J., Zhao, Y., Li, F., and Wang, X.: Assessment of extreme precipitation events and their teleconnections to El Niño Southern Oscillation, a case study in the Wei River Basin of China, Atmospheric Research, 218, 372–384, doi:10.1016/j.atmosres.2018.12.015, 2019.

Jolliffe, I. T.: Principal Component Analysis, Second Edition, Springer Series in Statistics, Springer-Verlag New York Inc, New York, NY, 2002.

Jordan D'Arcy: Introducing SSA for Time Series Decomposition, Kaggle, 4/29/2018, https://www.kaggle.com/jdarcy/introducing-ssa-for-time-series-decomposition, last access: 28 April 2020.966Z, 2018.

Jung, Y.: Multiple predicting K -fold cross-validation for model selection, Journal of Nonparametric Statistics, 30, 197–215, doi:10.1080/10485252.2017.1404598, 2018.

Kirchner, J. W.: Getting the right answers for the right reasons: Linking measurements, analyses, and models to advance the science of hydrology, Water Resour. Res., 42, 2465, doi:10.1029/2005WR004362, 2006.

Kisi, O.: Wavelet regression model for short-term streamflow forecasting, Journal of Hydrology, 389, 344–353, doi:10.1016/j.jhydrol.2010.06.013, 2010.

Kratzert, F., Klotz, D., Brenner, C., Schulz, K., and Herrnegger, M.: Rainfall–runoff modelling using Long Short-Term Memory (LSTM) networks, Hydrol. Earth Syst. Sci., 22, 6005–6022, doi:10.5194/hess-22-6005-2018, 2018.

Li, M., Wang, Q. J., Bennett, J. C., and Robertson, D. E.: A strategy to overcome adverse effects of autoregressive updating of streamflow forecasts, Hydrol. Earth Syst. Sci., 19, 1–15, doi:10.5194/hess-19-1-2015, 2015.

Liu, Z., Zhou, P., Chen, G., and Guo, L.: Evaluating a coupled discrete wavelet transform and support vector regression for daily and monthly streamflow forecasting, Journal of Hydrology, 519, 2822–2831, doi:10.1016/j.jhydrol.2014.06.050, 2014.

Lopez, J.H.: The power of the ADF test, Economics Letters, 57, 5–10, doi:10.1016/S0165-1765(97)81872-1, 1997.

Luo, X., Yuan, X., Zhu, S., Xu, Z., Meng, L., and Peng, J.: A hybrid support vector regression framework for streamflow forecast, Journal of Hydrology, 568, 184–193, doi:10.1016/j.jhydrol.2018.10.064, 2019.

Maheswaran, R. and Khosa, R.: Wavelets-based non-linear model for real-time daily flow forecasting in Krishna River, Journal of Hydroinformatics, 15, 1022–1041, doi:10.2166/hydro.2013.135, 2013.

Maity, R., Bhagwat, P. P., and Bhatnagar, A.: Potential of support vector regression for prediction of monthly streamflow using endogenous property, Hydrol. Process., 24, 917–923, doi:10.1002/hyp.7535, 2010.

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

Meng, E., Huang, S., Huang, Q., Fang, W., Wu, L., and Wang, L.: A robust method for non-stationary streamflow prediction based on improved EMD-SVM model, Journal of Hydrology, 568, 462–478, doi:10.1016/j.jhydrol.2018.11.015, 2019.

Minka, T. P.: Automatic Choice of Dimensionality for PCA, Advances in Neural Information Processing Systems, 598–604, 2001.

Mohammadi, K., Eslami, H. R., and Kahawita, R.: Parameter estimation of an ARMA model for river flow forecasting using goal programming, Journal of Hydrology, 331, 293–299, doi:10.1016/j.jhydrol.2006.05.017, 2006.

Mulvaney, T. J.: On the use of self-registering rain and flood gauges in making observations of the relations of rainfall and of flood discharges in a given catchment, Proceedings Institution of Civil Engineers, 4, 18–31, 1850.

Musa, A. B.: A comparison of ℓ1-regularizion, PCA, KPCA and ICA for dimensionality reduction in logistic regression, Int. J. Mach. Learn. & Cyber., 5, 861–873, doi:10.1007/s13042-013-0171-7, 2014.

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, doi:10.1016/0022-1694(70)90255-6, 1970.

Newman, A. J., Clark, M. P., Sampson, K., Wood, A., Hay, L. E., Bock, A., Viger, R. J., Blodgett, D., Brekke, L., Arnold, J. R., Hopson, T., and Duan, Q.: Development of a large-sample watershed-scale hydrometeorological data set for the contiguous USA: data set characteristics and assessment of regional variability in hydrologic model performance, Hydrol. Earth Syst. Sci., 19, 209–223, doi:10.5194/hess-19-209-2015, 2015.

Ng, A.: Machine learning yearning, URL: http://www. mlyearning. org/(96), 2017.

Noori, R., Karbassi, A. R., Moghaddamnia, A., Han, D., Zokaei-Ashtiani, M. H., Farokhnia, A., and Gousheh, M. G.: Assessment of input variables determination on the SVM model performance using PCA, Gamma test, and forward selection techniques for monthly stream flow prediction, Journal of Hydrology, 401, 177–189, doi:10.1016/j.jhydrol.2011.02.021, 2011.

Nourani, V., Komasi, M., and Mano, A.: A Multivariate ANN-Wavelet Approach for Rainfall–Runoff Modeling, Water Resources Management, 23, 2877, doi:10.1007/s11269-009-9414-5, 2009.

Paniconi, C. and Putti, M.: Physically based modeling in catchment hydrology at 50: Survey and outlook, Water Resour. Res., 51, 7090–7129, doi:10.1002/2015WR017780, 2015.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, É.: Scikit-learn: Machine Learning in Python, Journal of Machine Learning Research, 12, 2825–2830, 2011.

Quilty, J. and Adamowski, J.: Addressing the incorrect usage of wavelet-based hydrological and water resources forecasting models for real-world applications with best practices and a new forecasting framework, Journal of Hydrology, 563, 336–353, doi:10.1016/j.jhydrol.2018.05.003, 2018.

Quilty, J., Adamowski, J., Khalil, B., and Rathinasamy, M.: Bootstrap rank-ordered conditional mutual information (broCMI): A nonlinear input variable selection method for water resources modeling, Water Resour. Res., 52, 2299–2326, doi:10.1002/2015WR016959, 2016.

Rasouli, K., Hsieh, W. W., and Cannon, A. J.: Daily streamflow forecasting by machine learning methods with weather and climate inputs, Journal of Hydrology, 414-415, 284–293, doi:10.1016/j.jhydrol.2011.10.039, 2012.

Seo, Y., Kim, S., Kisi, O., and Singh, V. P.: Daily water level forecasting using wavelet decomposition and artificial intelligence techniques, Journal of Hydrology, 520, 224–243, doi:10.1016/j.jhydrol.2014.11.050, 2015.

Shahriari, B., Swersky, K., Wang, Z., Adams, R. P., and Freitas, N. de: Taking the Human Out of the Loop: A Review of Bayesian Optimization, Proc. IEEE, 104, 148–175, doi:10.1109/JPROC.2015.2494218, 2016.

Singh, V. P.: Hydrologic modeling: progress and future directions, Geosci. Lett., 5, 1145, doi:10.1186/s40562-018-0113-z, 2018.

Sivapragasam, C., Liong, S.-Y., and Pasha, M. F. K.: Rainfall and runoff forecasting with SSA–SVM approach, Journal of Hydroinformatics, 3, 141–152, doi:10.2166/hydro.2001.0014, 2001.

Solomatine, D. P., Maskey, M., and Shrestha, D. L.: Instance-based learning compared to other data-driven methods in hydrological forecasting, Hydrol. Process., 22, 275–287, doi:10.1002/hyp.6592, 2008.

Stéfan, v. d. W., Colbert, S. C., and Varoquaux, G.: The NumPy Array: A Structure for Efficient Numerical Computation: A Structure for Efficient Numerical Computation, Comput. Sci. Eng., 13, 22–30, doi:10.1109/MCSE.2011.37, 2011.

Stojković, M., Kostić, S., Plavšić, J., and Prohaska, S.: A joint stochastic-deterministic approach for long-term and short-term modelling of monthly flow rates, Journal of Hydrology, 544, 555–566, doi:10.1016/j.jhydrol.2016.11.025, 2017.

Tan, Q.-F., Lei, X.-H., Wang, X., Wang, H., Wen, X., Ji, Y., and Kang, A.-Q.: An adaptive middle and long-term runoff forecast model using EEMD-ANN hybrid approach, Journal of Hydrology, 567, 767–780, doi:10.1016/j.jhydrol.2018.01.015, 2018.

Tim, H., MechCoder, Gilles, L., Iaroslav, S., fcharras, Zé Vinícius, cmmalone, Christopher, S., nel215, Nuno, C., Todd, Y., Stefano, C., Thomas, F., rene-rex, Kejia, (K.) S., Justus, S., carlosdanielcsantos, Hvass-Labs, Mikhail, P., SoManyUsernamesTaken, Fred, C., Loïc, E., Lilian, B., Mehdi, C., Karlson, P., Fabian, L., Christophe, C., Anna, G., Andreas, M., and Alexander, F.: Scikit-Optimize/Scikit-Optimize: V0.5.2, Zenodo, 2018.

Tiwari, M. K. and Chatterjee, C.: Development of an accurate and reliable hourly flood forecasting model using wavelet–bootstrap–ANN (WBANN) hybrid approach, Journal of Hydrology, 394, 458–470, doi:10.1016/j.jhydrol.2010.10.001, 2010.

Todini, E.: Hydrological catchment modelling: past, present and future, Hydrol. Earth Syst. Sci., 11, 468–482, doi:10.5194/hess-11-468-2007, 2007.

Valipour, M., Banihabib, M. E., and Behbahani, S. M. R.: Comparison of the ARMA, ARIMA, and the autoregressive artificial neural network models in forecasting the monthly inflow of Dez dam reservoir, Journal of Hydrology, 476, 433–441, doi:10.1016/j.jhydrol.2012.11.017, 2013.

Vapnik, V., Golowich, S. E., and Smola, A. J.: Support Vector Method for Function Approximation, Regression Estimation and Signal Processing, Advances in Neural Information Processing Systems, 281–287, 1997.

Wangmeng Zuo, Kuanquan Wang, and D. Zhang: Bi-directional PCA with assembled matrix distance metric, in: IEEE International Conference on Image Processing 2005, IEEE International Conference on Image Processing 2005, 2, II-958, 2005.

Woldemeskel, F., McInerney, D., Lerat, J., Thyer, M., Kavetski, D., Shin, D., Tuteja, N., and Kuczera, G.: Evaluating post-processing approaches for monthly and seasonal streamflow forecasts, Hydrol. Earth Syst. Sci., 22, 6257–6278, doi:10.5194/hess-22-6257-2018, 2018.

Wu, C. L., Chau, K. W., and Li, Y. S.: Predicting monthly streamflow using data-driven models coupled with data-preprocessing techniques, Water Resour. Res., 45, 1331, doi:10.1029/2007WR006737, 2009.

Wu, Z. and Huang, N. E.: Ensemble Empirical Mode Decomposition: a Noise-Assisted Data Analysis Method, Adv. Adapt. Data Anal., 01, 1–41, doi:10.1142/S1793536909000047, 2009.

Xie, T., Zhang, G., Hou, J., Xie, J., Lv, M., and Liu, F.: Hybrid forecasting model for non-stationary daily runoff series: A case study in the Han River Basin, China, Journal of Hydrology, 577, 123915, doi:10.1016/j.jhydrol.2019.123915, 2019.

Xu, B., Zhou, F., Li, H., Yan, B., and Liu, Y.: Early fault feature extraction of bearings based on Teager energy operator and optimal VMD, ISA Transactions, 86, 249–265, doi:10.1016/j.isatra.2018.11.010, 2019.

Yaseen, Z. M., Ebtehaj, I., Bonakdari, H., Deo, R. C., Danandeh Mehr, A., Mohtar, W. H. M. W., Diop, L., El-Shafie, A., and Singh, V. P.: Novel approach for streamflow forecasting using a hybrid ANFIS-FFA model, Journal of Hydrology, 554, 263–276, doi:10.1016/j.jhydrol.2017.09.007, 2017.

Yu, P.-S., Chen, S.-T., and Chang, I.-F.: Support vector regression for real-time flood stage forecasting, Journal of Hydrology, 328, 704–716, doi:10.1016/j.jhydrol.2006.01.021, 2006.

Yu, S., Xu, Z., Wu, W., and Zuo, D.: Effect of land use types on stream water quality under seasonal variation and topographic characteristics in the Wei River basin, China, Ecological Indicators, 60, 202–212, doi:10.1016/j.ecolind.2015.06.029, 2016.

Zhang, X., Peng, Y., Zhang, C., and Wang, B.: Are hybrid models integrated with data preprocessing techniques suitable for monthly streamflow forecasting? Some experiment evidences, Journal of Hydrology, 530, 137–152, doi:10.1016/j.jhydrol.2015.09.047, 2015.

Zhang, Y. and Yang, Y.: Cross-validation for selecting a model selection procedure, Journal of Econometrics, 187, 95–112, doi:10.1016/j.jeconom.2015.02.006, 2015.

Zhao, X.-h. and Chen, X.: Auto Regressive and Ensemble Empirical Mode Decomposition Hybrid Model for Annual Runoff Forecasting, Water Resources Management, 29, 2913–2926, doi:10.1007/s11269-015-0977-z, 2015.

Zuo, G., Luo, J., Wang, N., Lian, Y., and He, X.: Decomposition ensemble model based on variational mode decomposition and long short-term memory for streamflow forecasting, Journal of Hydrology, 585, 124776, doi:10.1016/j.jhydrol.2020.124776, 2020.