Performance of Different Machine Learning Algorithms in Rainfall Runoff Modelling Compared to a Conceptual Model

Master Thesis

Benjamin Meyer

## Institute of Geography

University of Bern

Supervisors

Prof. Dr. Bettina Schaefli

Dr. Pascal Horton

**Introduction**

Our physical and economic well-being depends on the availability of water. More than half of the SDGs are, in one way or another, directly related to water resources. Sustainable management of water resources is a necessity to ensure everyone's access to water in sufficient quality and quantity, without harming the environment. But as important as water is, it is also a potential source of danger. Flooding is just one example of how water can cause damage, either directly or indirectly. The basis for effective resource management is good data. Knowing where, when and how much water is available is fundamental. One of the key environmental variables for water resource management is runoff. While measurement is the foundation, it can only provide current and historical data. To gain insight into future conditions, some form of modelling is required. That's where Rainfall Runoff Models (RRM) come in.   
Countless models have been developed over the past decades for different purposes, each with trade-offs in terms of degree of realism, data requirements, computational power needed and ease of use (Beven, 2011). RRMs can be classified into three types: physics-based models that attempt to model the physical processes, conceptual models that are simplified representations of the real world, and finally data-driven models such as standard statistical models and, more recently, machine learning models (Sitterson et al., 2019). In this study, we take a closer look at two (three) machine learning algorithms (XGBoost, LSTM) and their capabilities in modelling runoff in Swiss catchments.   
While it has been shown that LSTMs can outperform conceptual models in RRM (Lees et al., 2021; Li et al., 2021; Kratzert et a., 2018), no study has yet been published on Swiss catchments. Catchments in Switzerland cover a wide range of defining characteristics, such as glaciated high mountain catchments, snow dominated alpine catchments, strongly karstified catchments and lowland catchments. This wide range of dominant factors poses a challenge to models, resulting in different model performance depending on the dominant process.

and to compare them with the process-oriented hydrological model "Precipitation-Runoff-Evapotranspiration HRU Model PREVAH" (Viviroli et al., 2009).

**Research Questions**

This study aims to answer the following research gaps mentioned above:

How good is the performance of the machine learning models compared to to the model conceptual model PREVAH using data from 1985 – 2016 (Muelchi et al. 2020).

How well do machine learning based models perform depending on the catchment characteristics.

Does the performance of the machine learning models increase with an extended data period (1981 – 2020)?

(How do the simulation results using the CH2018 data look compared to the result of Mülchi et al. (2020).)

**What is Machine Learning (ML)**

Machine learning, an umbrella term that encompasses a wide range of computational methods, has its roots in the mid-20th century. The term was first used in the 1950s to describe a new breed of algorithms that could “learn” from data and make predictions or decisions without being explicitly programmed to do so.

The wider use of machine learning in recent years is largely due to a sharp increase in computing power. This has made it possible to process large amounts of data and run complex algorithms much faster than previously possible. As a result, machine learning has found applications in a wide range of fields, from healthcare to finance, and is increasingly becoming an important tool for businesses and researchers.

An important aspect of machine learning is the concept of hyperparameters. Hyperparameters are parameters that are not learned from the data, but are set before learning begins. They control various aspects of the learning process, such as the complexity of the model or the speed at which it learns.

The process of hyperparameter optimisation is considered one of the most important steps in machine learning. Some argue that it's even more important than the type of model used. This is because different sets of hyperparameters can produce very different results when applied to the same model and data. Therefore, finding the optimal set of hyperparameters can significantly improve the performance of a machine learning model.

**Methods and Data**

**Research area (Switzerland)**

The 6 representative catchments defined by Muelchi et al. 2020 (Rosegbach, Kander, Plessur, Emme, Venoge, Verzasca). They cover a wide range of catchment characteristics and are unregulated. All catchments were used by Mülchi et al. (2020) These 6 catchments had good model quality with the PREVAH model. In a second step, we also compare catchments where the PREVAH model performed poorly. This comparison should provide insight into how the ML models perform and where their strengths and weaknesses lie.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Catchment | MQ | Station elevation | Catchment area | Mean elevation | Glaciation |
| Rosegbach | 2.8 | 1770 | 66 | 2704 | 21.7 |
| Kander | 20 | 650 | 491 | 1854 | 5.1 |
| Plessur | 8.0 | 565 | 264 | 1868 | 0.0 |
| Emme |  |  |  |  | 0.0 |
| Venoge | 4.1 | 384 | 228 | 686 | 0.0 |
| Verzasca | 11 | 495 | 185 | 1651 | 0.0 |

**Data**

The following data are used in this study:

Shape-files of catchments

Discharge (BAFU)

Gridded mean temperature (TabsD; Frei, 2014; MeteoSwiss, 2019a)

Gridded daily precipitation sum (RhiresD; Frei and Schär, 1998; MeteoSwiss, 2019b)

(Gridded temperature and precipitation data of the dataset CH2018)

Gridded snow cover?

Gridded min/max Temperature

Gridded relative humidity (availability?)

**Data Preparation**

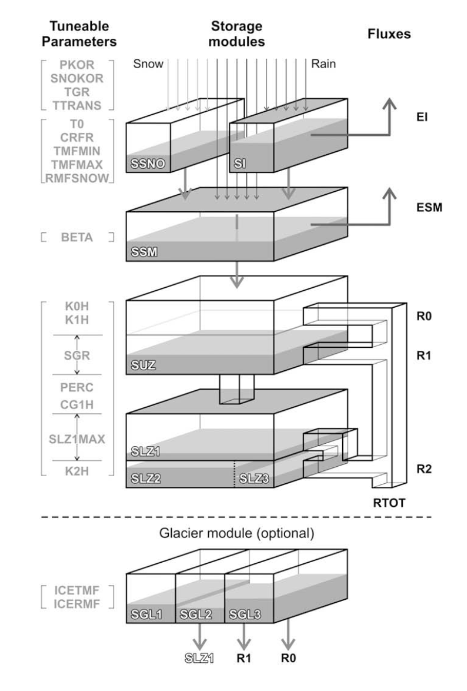
The daily mean temperature and daily precipitation sum was provided as gridded data. Using the shape files of the catchments the gridded data was aggregated to mean values.

**Methods**

Three different machine learning algorithms are considered and compared with PREVAH. The comparison is based on the Nash-Sutcliffe Efficiency (NSE) and the Kling-Gupta Efficiency (KGE). The above models are explained below. (For an in-depth video explanation visit: https://www.youtube.com/@statquest)

**PREVAH**

The model PREVAH which is used as a benchmark benchmark is a spacially explicit, process oriented model designed to model catchments with complex topography. The catchment is split into so called hydrological response units (HRU), surfaces with similar response. For each HRU a storage cascade is simulated and then combined to an area mean value. Figure ?? shows a storage cascade used it the model.



**XGBoost:**

XGBoost is a regression forest based gradient boosting algorithm that can handle large datasets very efficiently. It has seen some use in RRM, but is not as popular as other machine learning models.

The open source software library originated from a research project at the University of Washington and is developed by an active group of community members (https://github.com/dmlc/xgboost).

It's available in 6 programming languages and runs on all major platforms. The main advantages of XGBoost are its ease of use and it's resource efficiency. The low entry barrier opens it up to a wider range of users than more complicated deep learning models, and it's resource efficiency makes hyperparameter optimisation fast, even on large datasets. The main drawback of the algorithm is the limited value range of the predictions, which is restricted to the value range of the training set. This is a major limitation when predicting extreme events with low probability of occurrence, especially those that have not yet been observed.

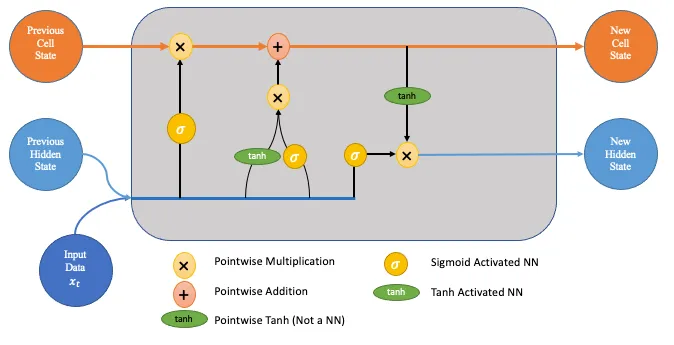
XGBoost has many hyperparameters that can be adjusted. The models here have been optimised for the following hyperparameters:

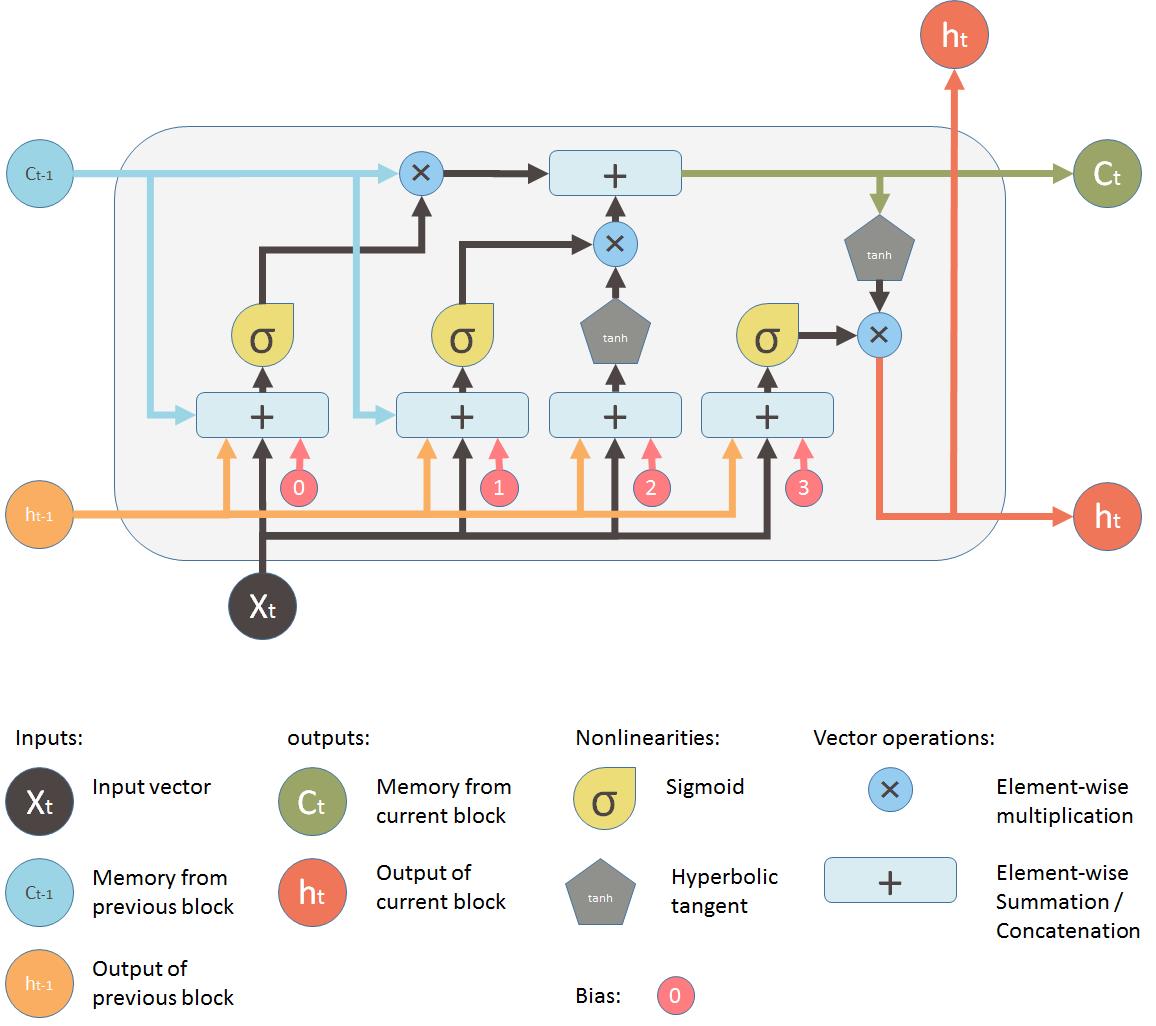
|  |  |  |
| --- | --- | --- |
| Hyperparameter | Description | Optimization range |
| eta | Eta is the learning rate and controls how much is learned per round. High values lead to fast overfitting | 0.001 – 0.25 |
| max depth | Maximum depth of a tree. Larger tree depths allow to learn more complex data structures and more different output values, but lead to overfitting. | 2 – 12 |
| Min child weight | Controls how much weight a branch must have to be further partitioned. Higher numbers lead to more conservative trees. | 1-50 |
| alpha | L1 regularisation on weights | 1 – 10 |
| lambda | L2 regularisation on weights | 1 – 10 |
| nrounds: | maximum number of boosting iterations | 1 – 200 |

The optimization was done using a Bayesian optimization process to find the optimal parameter set.

**Long short-term memory (LSTM):**

The LSTM is a recurrent neural network (RNN) that solves the exploding/vanishing gradient problem. When using long sequences, traditional RNNs face the problem that the long term gradient tends to zero or infinity. LSTMs are designed to classify, process and predict data based on time series and have been used in speech recognition, translation, robot control and time series prediction. As it was designed to process time series type data, it has naturally been used extensively in connection with RRM (ca. 7300 results on google scholar) and has been shown to give good results. The disadvantage of LSTMs is the computational power required to train the model, which makes hyper-parameter optimisation more difficult.





LSTMs have many hyperparameters that can be adjusted. The models here have been optimised for the following hyperparameters:

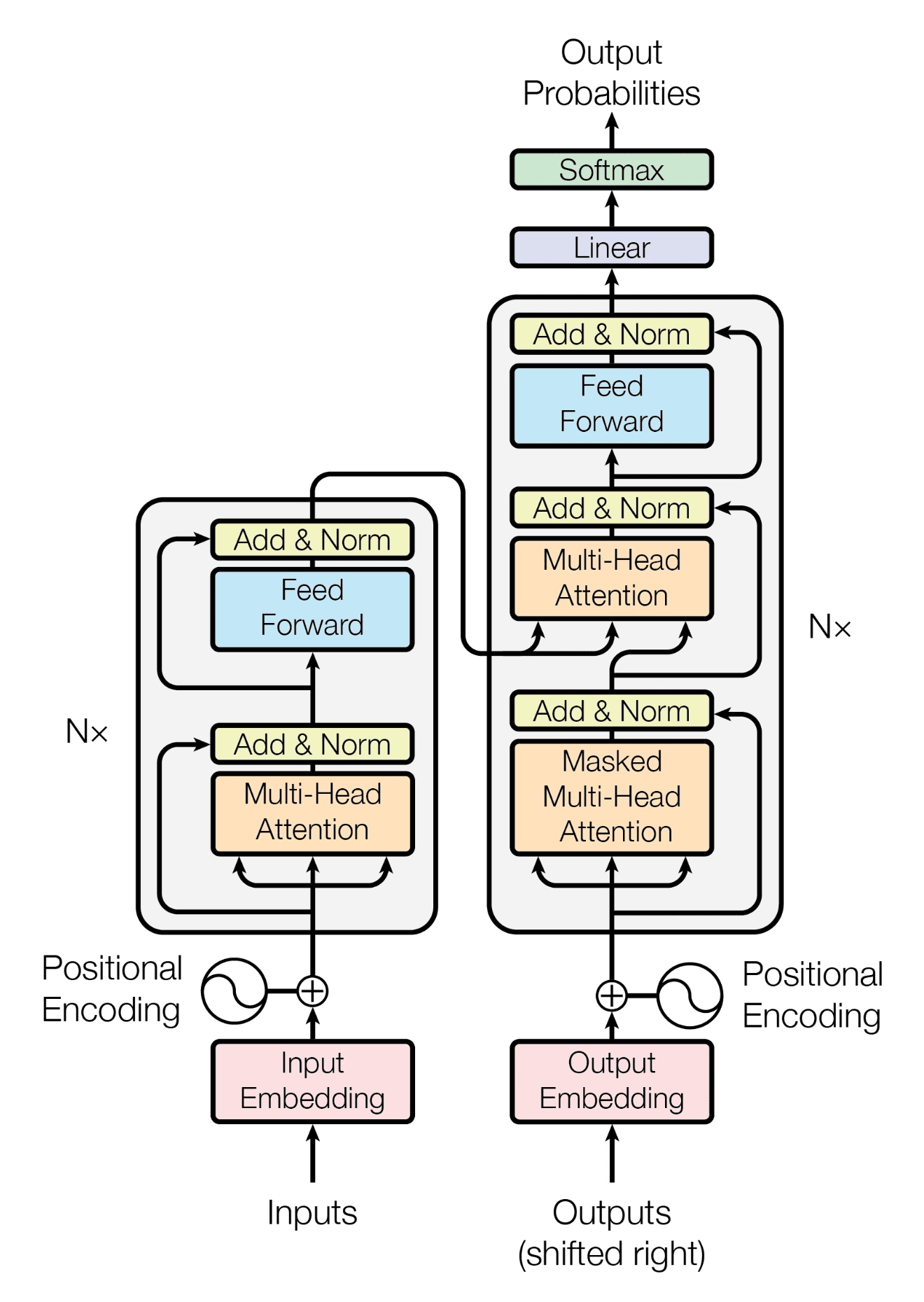
|  |  |  |
| --- | --- | --- |
| Hyperparameter | Description | Optimization range |
| nlayers | Number of layers in the neural network | 1 – 5 |
| nnodes | Number of nodes in each layer | 5 – 300 |
| batch size | Step size after which the weights are updated within an epoch | 5 – 150 |
| epoche | Number of times the dataset is used to train the neural network | 1 – 100 |
| step size | Number of time steps of the input data | 5 – 200 |

The optimization was done using a Bayesian optimization process to find the optimal parameter set. In addition the loss function “mse” and optimizer “adam” were used.

**Transformer:**

Although the self-attention mechanism in deep learning models has mainly been used in natural language processing (ChatGPT, BERT), it has recently started to emerge in other fields. However, its use in rainfall runoff modelling is still limited due to its recent development. One of the advantages of this mechanism is that it can develop different temporal relationships on its own, making it a state of the art algorithm for other sequential data such as speech, video and audio. However, multivariate modelling does not perform as well as univariate modelling. This is because it struggles to link the attention mechanism between the different input variables.

Open questions: computational costs? Regional fitting?



**Calibration and Validation**

Where available, the models were calibrated and validated with data from 1985 to 2016. This period was used in the Muelchi et al. 2020 study. In addition, the models were calibrated and validated over a longer time period (1981 – 2020) to see if the models improved with more data. Two out of every three years were used to calibrate the models. Every third year was used for validation. This pattern minimised the influence of random and non-random trends on model calibration (particularly noticeable in catchments with high mean elevation). Muelchi et al. (2020) used Nash-Sutcliffe efficiency and Kling-Gupta efficiency to analyse model performance. Both performance indicators measure model bias, where 1 describes a perfect model. The Kling-Gupta efficiency focuses more on the correct variability and correlation of the runoff.

**Results**

**Discussion**

**Conclusion**

**Links:**

Peel, M. C. and McMahon, T. A.: Historical development of rainfall-runoff modeling, Wiley Interdisciplinary Reviews: Water, 7, e1471, <https://doi.org/10.1002/wat2.1471>, 2020. [a](https://hess.copernicus.org/articles/25/5517/2021/" \l "xref_paren.20)

Addor, N. and Melsen, L.: Legacy, rather than adequacy, drives the selection of hydrological models, Water Resour. Res., 55, 378–390, 2019. [a](https://hess.copernicus.org/articles/25/5517/2021/" \l "xref_paren.29)

Hoedt, P.-J., Kratzert, F., Klotz, D., Halmich, C., Holzleitner, M., Nearing, G. S., Hochreiter, S., and Klambauer, G.: MC-LSTM: MassConserving LSTM, in: Proceedings of the 38th International Conference on Machine Learning, vol. 139 of Proceedings of Machine Learning Research, edited by: Meila, M. and Zhang, T., 4275–4286, PMLR, available at: <http://proceedings.mlr.press/v139/hoedt21a.html>(last access: 1 October 2021), 2021. [a](https://hess.copernicus.org/articles/25/5517/2021/" \l "xref_paren.24)

[Aktuelle Daten der Nationalen Daueruntersuchung der Fliessgewässer (NADUF) (admin.ch)](https://www.bafu.admin.ch/bafu/de/home/themen/wasser/zustand/daten/aktuelle-messwerte-der-gewaesser/aktuelle-daten-der-nationalen-daueruntersuchung-der-fliessgewaes.html)

[Application of a conceptual runoff model in different physiographic regions of Switzerland: Hydrological Sciences Journal: Vol 37, No 3 (tandfonline.com)](https://www.tandfonline.com/doi/abs/10.1080/02626669209492583)

[IHACRES, GR4J and MISD-based multi conceptual-machine learning approach for rainfall-runoff modeling | Scientific Reports (nature.com)](https://www.nature.com/articles/s41598-022-16215-1)

https://link.springer.com/article/10.1007/s00506-021-00767-z

https://tubaf.qucosa.de/landing-page/?tx\_dlf[id]=https%3A%2F%2Ftubaf.qucosa.de%2Fapi%2Fqucosa%253A22570%2Fmets

<https://en.wikipedia.org/wiki/Timeline_of_machine_learning>

https://xgboost.ai/about

Beven, K. J.: Rainfall-runoff modelling: the primer, John Wiley & Sons, 2011. [a](https://hess.copernicus.org/articles/25/5517/2021/" \l "xref_paren.7), [b](https://hess.copernicus.org/articles/25/5517/2021/" \l "xref_paren.8), [c](https://hess.copernicus.org/articles/25/5517/2021/" \l "xref_paren.26)

Gupta, H. V., Kling, H., Yilmaz, K. K., and Martinez, G. F.: Decomposition of the mean squared error and NSE performance criteria: Implications for improving hydrological modelling, J. Hydrol., 377, 80–91, https://doi.org/10.1016/j.jhydrol.2009.08.003, 2009.

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, https://doi.org/10.5194/hess-22-6005-2018, 2018.

Lees, T., Buechel, M., Anderson, B., Slater, L., Reece, S., Coxon, G., and Dadson, S. J.: Benchmarking data-driven rainfall–runoff models in Great Britain: a comparison of long short-term memory (LSTM)-based models with four lumped conceptual models, Hydrol. Earth Syst. Sci., 25, 5517–5534, https://doi.org/10.5194/hess-25-5517-2021, 2021.

Li, W., Kiaghadi, A. & Dawson, C. High temporal resolution rainfall–runoff modeling using long-short-term-memory (LSTM) networks. *Neural Comput & Applic* 33, 1261–1278 (2021). https://doi.org/10.1007/s00521-020-05010-6

<https://boris.unibe.ch/148715/1/Hydro_CH2018_finalreport.pdf>

Muelchi R., Roessler O., Schwanbeck J., Weingartner R., Martius O. 2020. Neue  
hydrologische Szenarien für die Schweiz. Im Auftrag des Bandesamts für Umwelt  
(BAFU), Bern, Schweiz, 51 S., doi: 10.7892/boris.148715.

Nash, J. E. and Sutcliffe, J. V.: River flow forecasting through conceptual models part I – A discussion of principles, J. Hydrol., 10, 282–290, https://doi.org/10.1016/0022-1694(70)90255-6, 1970.

Sitterson, Jan; Knightes, Chris; Parmar, Rajbir; Wolfe, Kurt; Avant, Brian; and Muche, Muluken, "An Overview of Rainfall-Runoff Model Types" (2018). International Congress on Environmental Modelling and Software. 41. https://scholarsarchive.byu.edu/iemssconference/2018/Stream-C/41

**Apendix**