

# Stochastic Modeling of Multiple Streamflow Time Series in Colombian Based on Gaussian Processes

**Author:** Julián David Pastrana-Cortés

**Director:** Álvaro Angel Orozco-Gutiérrez

**Co-director:** David Augusto Cardenas-Peña

Automatics Research Group  
October 2, 2024



# Motivation

Understanding the implications of time series associated with hydrological variables, such as flow rates or reservoir levels, is essential for hydroelectric generation and the planning of other generation systems in Colombia.



(a) Irrigation



(b) Flood control



(c) Hydropower generation

## Challenges

Non-linearities, high stochasticity, and complex water resource patterns.

# The Importance of Hydrological Forecasting

Understanding hydrological processes has become increasingly critical in natural resource management. Suitable forecasting allows the anticipation capacity of extreme hydrological events such as droughts and heavy rainfall.



(a) Drought Condition



(b) Full Dam

# Problem Statement and Research Question

Model \ Challenge	Interpretability	Nonlinearity	Stochasticity	Long-term	Scalability	Output Constraint
Model						
Physically Driven [1, 2]	✓	✓	✗	✓	✗	✓
AR [3]	✓	✗	✓	✓	✓	✗
NN [4]	✗	✓	✗	✗	✓	✓
RNN [5]	✗	✓	✗	✗	✓	✓
LSTM [6, 7, 8]	✗	✓	✗	✓	✓	✓
GP [9, 10]	✓	✓	✓	✓	✗	✗

## Research Question

How to develop a joint probabilistic prediction model for multiple hydrological series associated with electricity generation, that describes the randomness of the forecast, is scalable, utilizes task correlations to improve performance, and incorporates output constraints to ensure feasible predictions?

# Objectives

## General Objective

Develop a stochastic forecasting model for making multiple simultaneous predictions of hydrological time series. This model will take advantage of cross-correlations among the outputs to improve performance, while maintaining scalability and output constraints.

## Specific Objectives

- Develop a model that allows the forecasting of hydrological time series, properly quantifying the **uncertainty** associated with each value within the prediction horizons.
- Design a **multi-output** forecasting methodology that captures and models cross-correlations between hydrological time series, to improve forecast accuracy within forecast horizons.
- Develop a multi-output prediction methodology that handles **data constraints** across reservoirs while maintaining high forecasting performance as measured by probabilistic metrics.

# Problem Setting

We model time series using observed vectors. At each time step  $n$ , the vector  $\mathbf{v}_n \in \mathbb{R}^D$  represents hydrological resources across  $D$  outputs. The input vector  $\mathbf{x}_n$  for the model is constructed from time  $n$  back to  $n - T + 1$ :

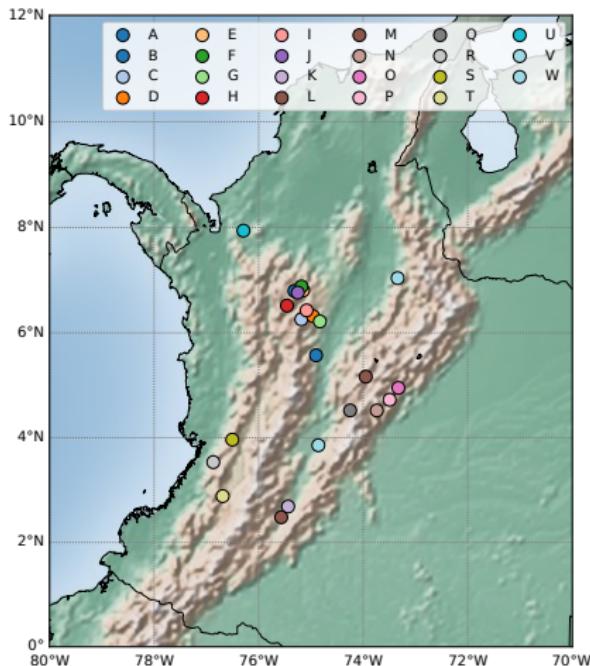
$$\mathbf{x}_n = \begin{bmatrix} \mathbf{v}_n \\ \mathbf{v}_{n-1} \\ \vdots \\ \mathbf{v}_{n-T+1} \end{bmatrix} \in \mathcal{X},$$

where  $T$  is the model order, and  $\mathcal{X} \subset \mathbb{R}^{DT}$  represents the input space. The target output vector  $\mathbf{y}_n$  is given as:

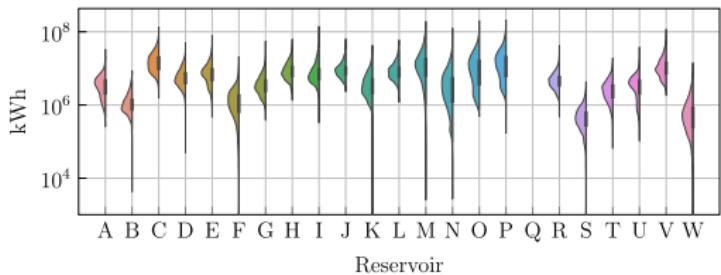
$$\mathbf{y}_n = \mathbf{v}_{n+H} \in \mathbb{R}^D,$$

where  $H$  is the prediction horizon. We build a dataset  $\mathcal{D} = \{\mathbf{x}_n, \mathbf{y}_n\}_{n=1}^N = \{\mathbf{X}, \mathbf{y}\}$ , comprising  $N$  input-output pairs.

# Reservoir Locations and Dataset Overview



The hydrological forecasting task utilizes daily streamflow data from  $D = 23$  Colombian reservoirs from January 1, 2010, to February 28, 2022.



Although volumetric measurements are recorded, they are reported in kilowatt-hours (kWh) by the hydroelectric power plants.

# Methodology

## Performance Metrics

- Mean Squared Error (MSE)
- Mean Standardized Log Loss (MSLL)
- Continuous Ranked Probability Score (CRPS)
- Negative Log Predictive Density (NLPD)

## Gaussian Process Models

- ① Start with a Sparse Variational GPs for scalable and stochastic regression.
- ② Extend to Multi-Output GPs, capturing dependencies across multiple outputs.
- ③ Introduce Chained Correlated GPs to handle non-Gaussian likelihoods.

# Objective 1: Sparse Variational Gaussian Process

In a GP framework, the function  $f(\cdot)$  maps inputs  $x_n$  to outputs  $y_n$ . Adding i.i.d. Gaussian noise  $\epsilon$ , the model becomes:

$$y_n = f(x_n) + \epsilon$$

For test inputs  $X_*$ , the joint distribution of training outputs  $y$  and test outputs  $f_*$  is:

$$\begin{bmatrix} y \\ f_* \end{bmatrix} \sim \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} \mathbf{K}_y & \mathbf{K}_* \\ \mathbf{K}_*^\top & \mathbf{K}_{**} \end{bmatrix}\right)$$

The posterior distribution for test points is:

$$f_* | X_*, \mathcal{D} \sim \mathcal{N}(\mathbf{K}_*^\top \mathbf{K}_y^{-1} y, \mathbf{K}_{**} - \mathbf{K}_*^\top \mathbf{K}_y^{-1} \mathbf{K}_*)$$

- $\mathbf{K}_y = \mathbf{K} + \Sigma_\epsilon$ , where  $\mathbf{K} \in \mathbb{R}^{ND \times ND}$  is the covariance matrix for the train set and  $\Sigma_\epsilon$  contains task-wise noise.
- $\mathbf{K}_{**} \in \mathbb{R}^{N_* D \times N_* D}$  is the covariance matrix for the test set.
- $\mathbf{K}_* \in \mathbb{R}^{ND \times N_* D}$  represents the cross-covariance matrix between the training and test points.

# Variational Inference for Scalability

The prediction performance is influenced by the selected parameter set  $\theta$  and the matrix  $\Sigma_\epsilon$ . These parameters are determined by maximizing the marginal log-likelihood:

$$\{\theta_{\text{opt}}, \Sigma_{\epsilon \text{opt}}\} = \arg \max_{\theta, \Sigma_\epsilon} -\frac{1}{2}\mathbf{y}^\top \mathbf{K}_y^{-1} \mathbf{y} - \frac{1}{2} \ln |\mathbf{K}_y| - \frac{ND}{2} \ln 2\pi,$$

with complexity  $\mathcal{O}(N^3 D^3)$  due to the need to invert the matrix  $\mathbf{K}_y$ . For scalability, we introduce  $M \ll N$  inducing points  $Z$  with inducing variables  $q(\mathbf{u}) = \mathcal{N}(\boldsymbol{\mu}, \mathbf{S})$ , providing the following ELBO:

$$\mathcal{L} = \sum_{d=1}^D \sum_{n=1}^N \mathbb{E}_{q(f_d(\mathbf{x}_n))} \{ \ln p(y_{dn} \mid f_d(\mathbf{x}_n)) \} - \sum_{d=1}^D \text{KL}\{q(\mathbf{u}_d) \parallel p(\mathbf{u}_d)\},$$

where  $f_d(\mathbf{x}_n)$  represents the  $d$ -th latent function value at input  $\mathbf{x}_n$ , and  $y_{dn}$  is the corresponding observed value. This reduces the complexity to  $\mathcal{O}(NM^2D^3)$ . All parameters were tuned by Adam optimizer.

# Model Setup

The GP covariance is factorized into two kernels:  $k_{\mathcal{X}}$  for input correlations and  $k_D$  for task correlations:

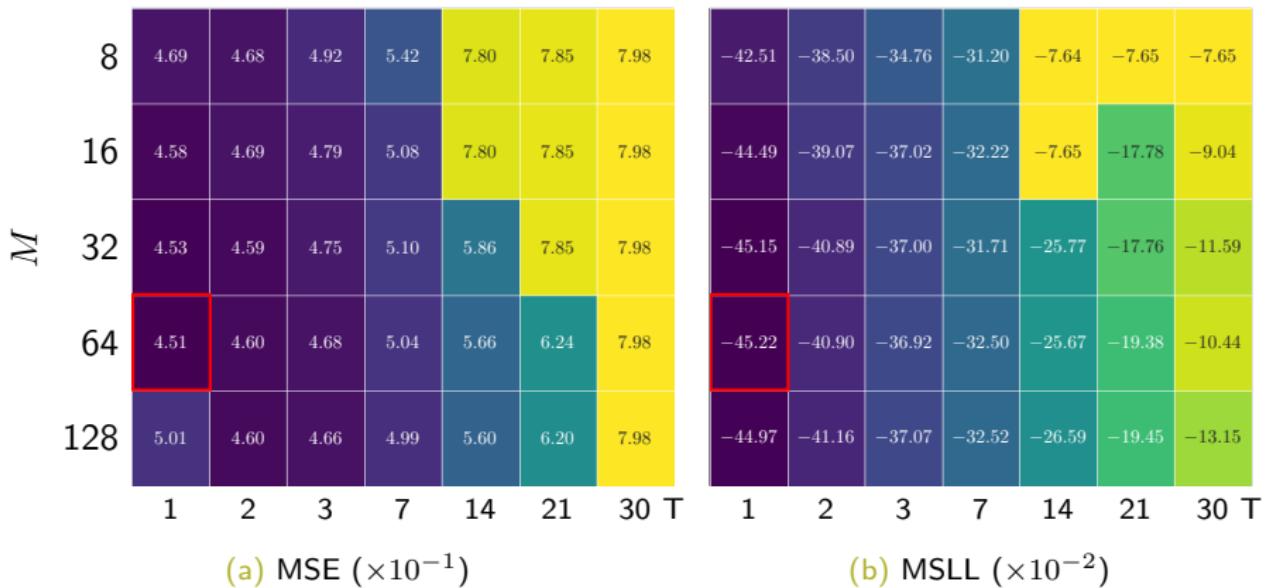
$$k((\mathbf{x}, d), (\mathbf{x}', d')) = k_{\mathcal{X}}(\mathbf{x}, \mathbf{x}' | \Theta_d) k_D(d, d' | \sigma_d),$$

with:

$$k_{\mathcal{X}}(\mathbf{x}, \mathbf{x}') = \exp\left(-\frac{1}{2}(\mathbf{x} - \mathbf{x}')^\top \Theta_d^{-2}(\mathbf{x} - \mathbf{x}')\right),$$
$$k_D(d, d') = \sigma_d^2 \delta_{d,d'},$$

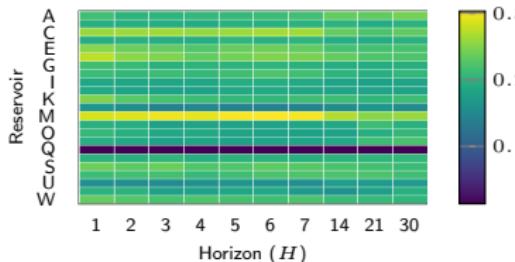
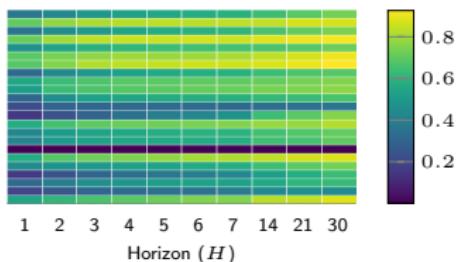
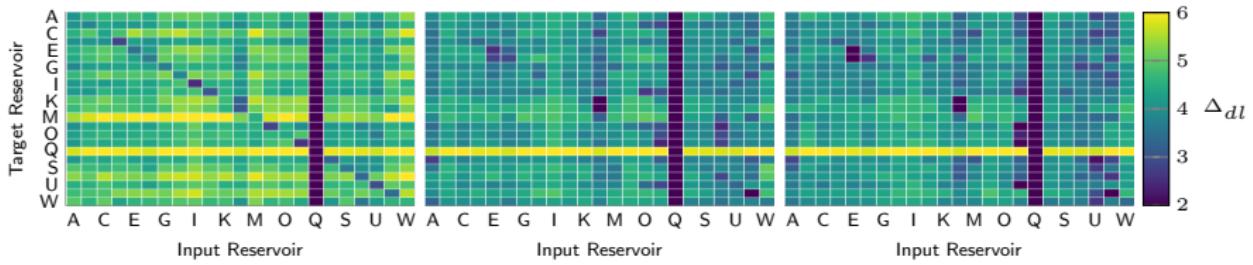
where  $\delta_{d,d'}$  is the Kronecker delta,  $\Theta_d = \text{diag}\{\Delta_{dl}\}_{l=1}^{DT}$  is the lengthscale matrix, and  $\sigma_d^2$  is the output scale. This reduces complexity to  $\mathcal{O}(NM^2D)$  by avoiding explicit task correlations.

# Grid search average values for tuning $T$ and $M$



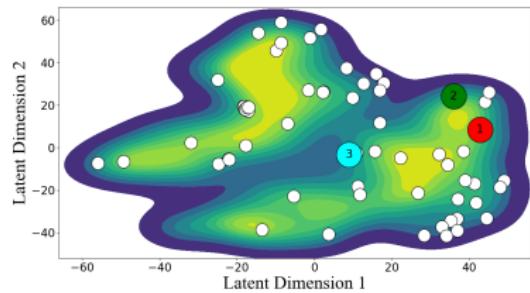
- The error increases with larger values of  $T$ .
- Smaller values of  $M$  are more sensitive to initialization.
- Larger values of  $M$  tend to overfit.
- The optimal parameters are  $M = 64$  and  $T = 1$ .

# Reservoir-Wise Kernel and Likelihood Parameters

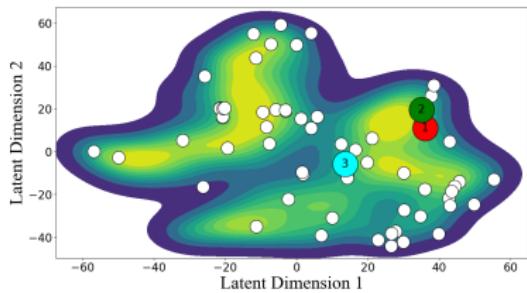
(a) Output scales  $\sigma_d^2$ (b) Noise variances  $\Sigma_\epsilon$ 

- For longer horizons, output scales decrease, while noise variances increase, reflecting weaker correlations and greater task complexity.
- As horizons extend, the relevance of main diagonal lengthscales diminishes, with off-diagonal lengthscales becoming more significant.

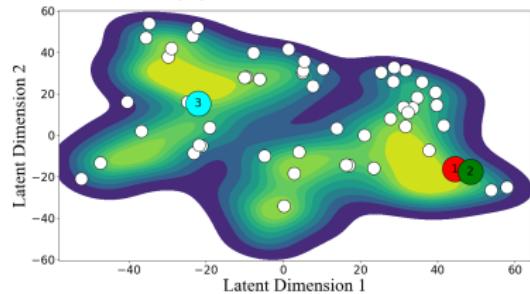
# SVGP t-SNE Latent Mapping Space



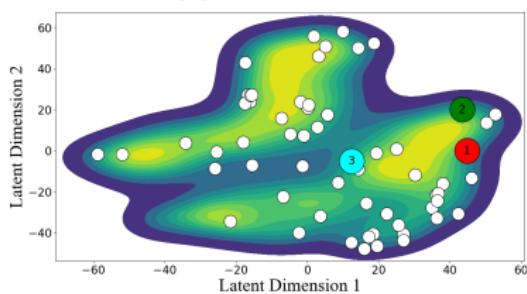
(a) Reservoir E



(b) Reservoir I



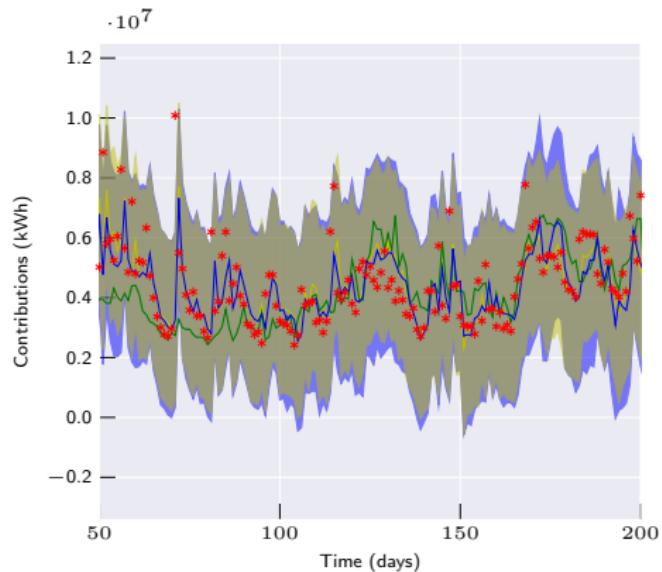
(c) Reservoir L



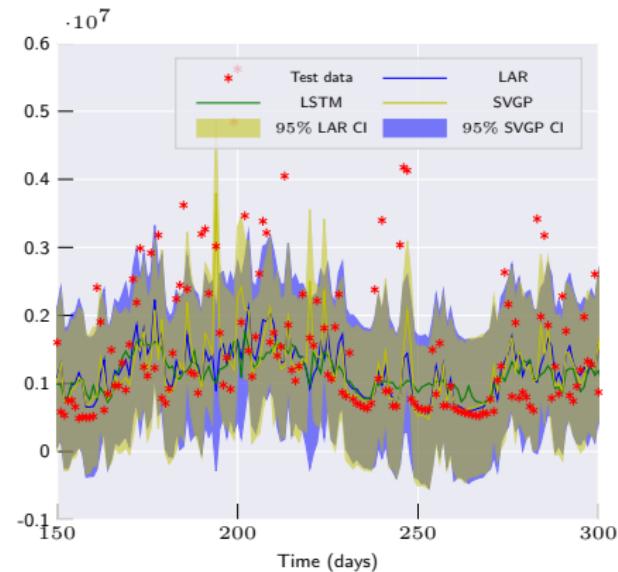
(d) Reservoir U

- Red and green points cluster near the data mode along the reservoir, capturing global information.
- The cyan point adjusts to capture task-specific details.

# One-day-ahead Models Forecasting



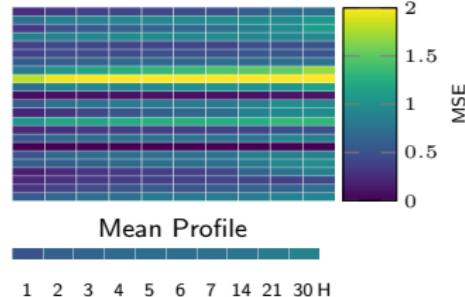
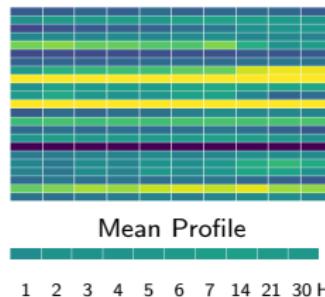
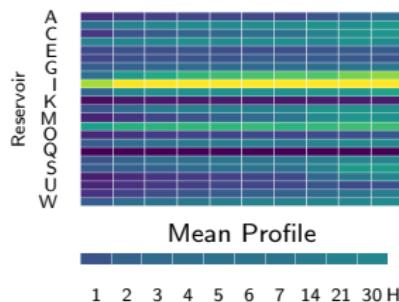
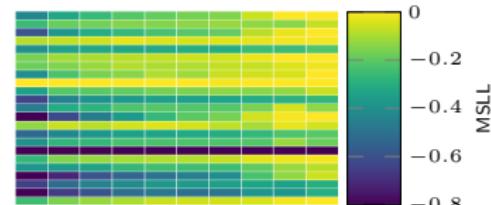
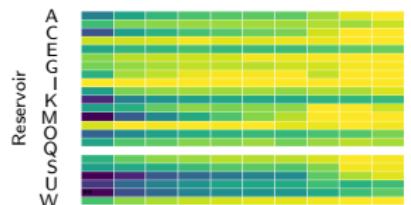
(a) Reservoir A



(b) Reservoir I

The SVGP closely follows the time-series data and captures its stochastic nature through the predictive distribution.

# Metrics achieved by LAR, LSTM, and SVGP



(a) LAR

(b) LSTM

(c) SVGP

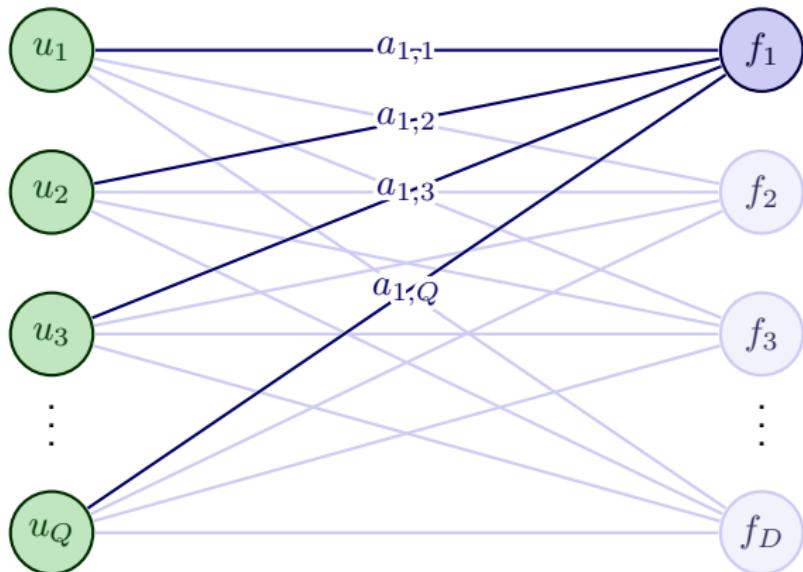
- Longer horizons yield larger errors across all models.
- LAR and SVGP models significantly outperform LSTM models in all scenarios.
- SVGP models demonstrate lower errors and a slower increase in error with longer horizons compared to LAR models.

# Performance Assessment

H	MSE			MSLL		CRPS	
	LAR	LSTM	SVGP	LAR	SVGP	LAR	SVGP
1	0.51	0.96	0.52 *	-0.39	-0.53	0.34	0.32
2	0.63	1.01	<b>0.61</b> *	-0.27	<b>-0.42</b>	0.39	<b>0.36</b>
3	0.68	1.02	<b>0.65</b> *	-0.22	<b>-0.38</b>	0.41	<b>0.38</b>
4	0.72	1.03	<b>0.69</b> *	-0.18	<b>-0.34</b>	0.42	<b>0.39</b>
5	0.74	1.06	<b>0.71</b> *	-0.17	<b>-0.33</b>	0.43	<b>0.40</b>
6	0.76	1.07	<b>0.72</b> *	-0.16	<b>-0.32</b>	0.44	<b>0.40</b>
7	0.76	1.11	<b>0.74</b> *	-0.15	<b>-0.31</b>	0.44	<b>0.41</b>
14	0.83	1.12	<b>0.79</b> *	-0.10	<b>-0.27</b>	0.46	<b>0.43</b>
21	0.88	1.08	<b>0.83</b> *	-0.07	<b>-0.23</b>	0.48	<b>0.45</b>
30	0.91	1.06	<b>0.88</b> *	-0.05	<b>-0.20</b>	0.49	<b>0.46</b>
Grand Average	0.74	1.05	<b>0.71</b> *	-0.18	<b>-0.33</b>	0.43	<b>0.40</b>

- Bold and asterisk indicate a  $p$ -value  $p < 1\%$  (LAR vs. SVGP, LSTM vs. SVGP).
- SVGP outperforms all models, except for LAR at  $H = 1$ , where linear dependence is stronger.
- As the horizon increases, SVGP effectively captures complex input-output relations, significantly outperforming the other models.

# Objective 2: Multi-Output Gaussian Processes



**Independent Process (IGP)**

$$u_q(\mathbf{x}) \sim \mathcal{GP}(0, k_q(\mathbf{x}, \mathbf{x}'))$$

**Latent Process (LMCGP)**

$$f_d(\mathbf{x}) = \sum_{q=1}^Q a_{d,q} u_q(\mathbf{x})$$

# Variational Inference, ELBO, and Predictive Distribution

We extend variational inference to include the independent set, utilizing the inducing variables  $u_q$  derived from independent processes. The ELBO is given by:

$$\mathcal{L} = \sum_{d=1}^D \sum_{n=1}^N \mathbb{E}_{q(f_{dn})} \{ \log p(y_{dn} | f_{dn}) \} - \sum_{\mathbf{q}=1}^Q \text{KL}\{q(\mathbf{u}_q) \| p(\mathbf{u}_q)\}$$

The posterior over test points  $X_*$ ,  $p(\mathbf{f}_* | \mathbf{y})$ , is given by:

$$p(\mathbf{f}_* | \mathbf{y}) \approx q(\mathbf{f}_*) = \int p(\mathbf{f}_* | \mathbf{u}) q(\mathbf{u}) d\mathbf{u}$$

Gaussian noise  $\sigma_{N_d}^2$  is added to obtain the predictive distribution. The complexity is now  $\mathcal{O}(DNQM^2)$ .

# Model Setup

## Covariance Function (LMCGP)

The LMCGP model uses a squared exponential kernel:

$$k_q(\mathbf{x}, \mathbf{x}' | \Theta_q) = \exp\left(-\frac{1}{2}(\mathbf{x} - \mathbf{x}')^\top \Theta_q^{-2}(\mathbf{x} - \mathbf{x}')\right)$$

Here,  $\Theta_q$  is the lengthscale matrix, and  $a_{d,q}$  work as outputscales.

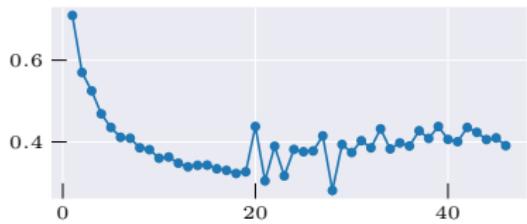
## Optimization and Model Variants

Strong dependencies between parameters may cause poor local minima [11]. We address this by combining Natural Gradient (NG) to optimize variational parameters, and Adam for the rest [12].

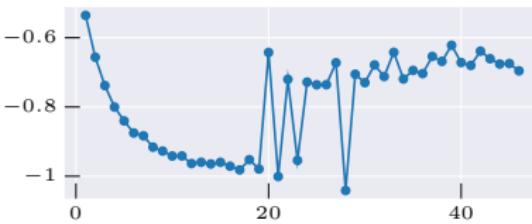
### Variants:

- IGP: Independent GP (Adam).
- IGP+: Independent GP (Adam+NG).
- LMCGP: Correlated GP (Adam+NG)

# Performance metrics vs Q



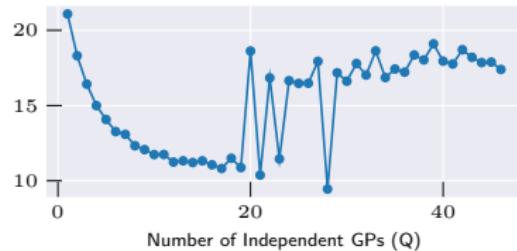
(a) MSE



(b) MSLL



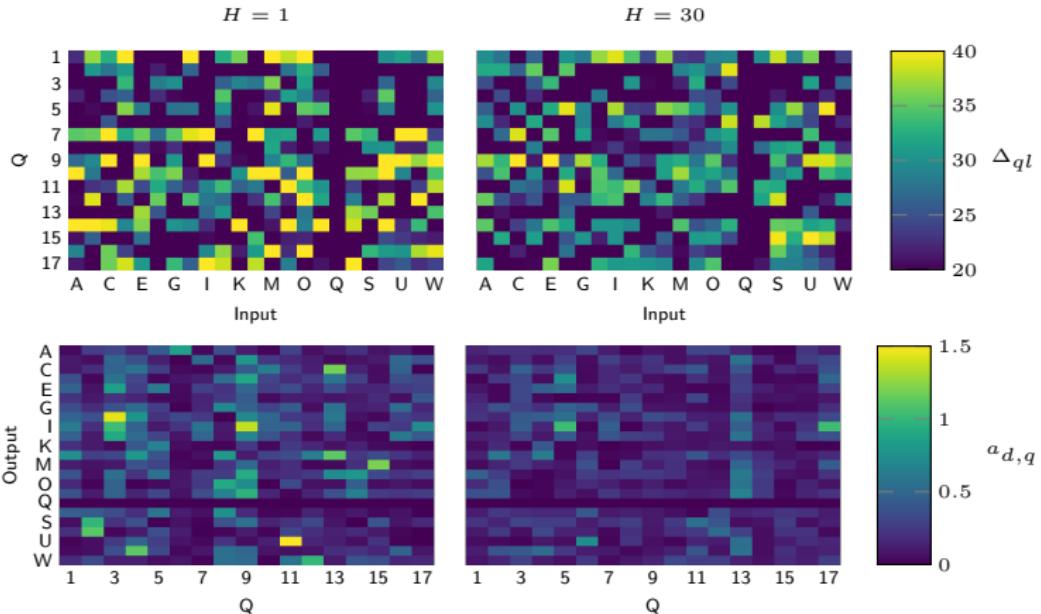
(c) CRPS



(d) NLPD

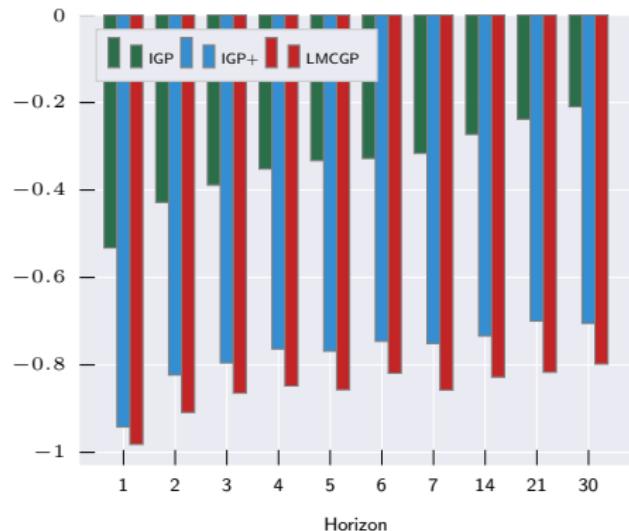
Beyond  $Q = 19$ , adding more independent GPs becomes counterproductive due to handling a more complex model. We select  $Q = 17$  as the proper parameter.

# Reservoir-wise Lengthscale and $a_{d,q}$

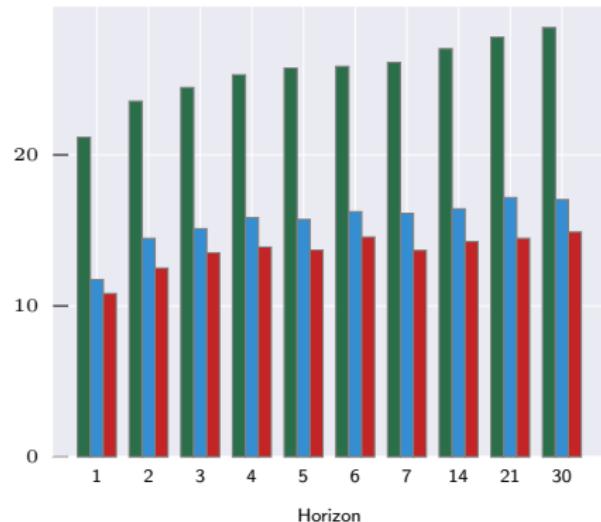


- As the horizon increases, lengthscale values become less selective.
- All independent GPs incorporate more features due to the extended time gap.
- The  $a_{d,q}$  coefficients decrease, indicating weaker individual feature contributions to each output.

# Bar plots comparing IGP, IGP+, and LMCGP



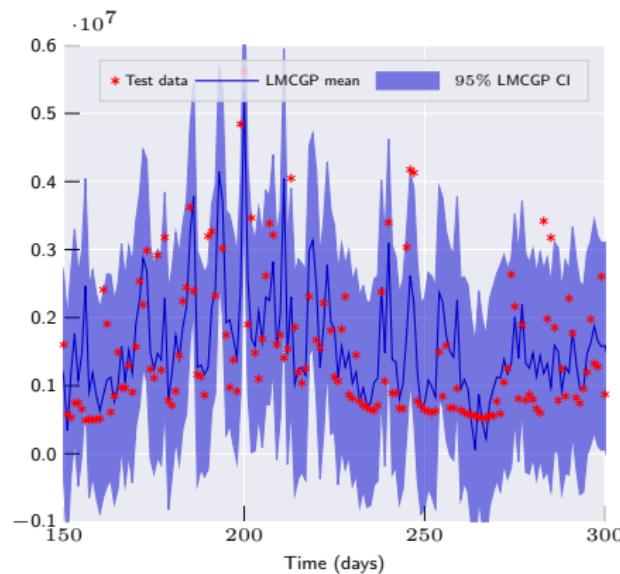
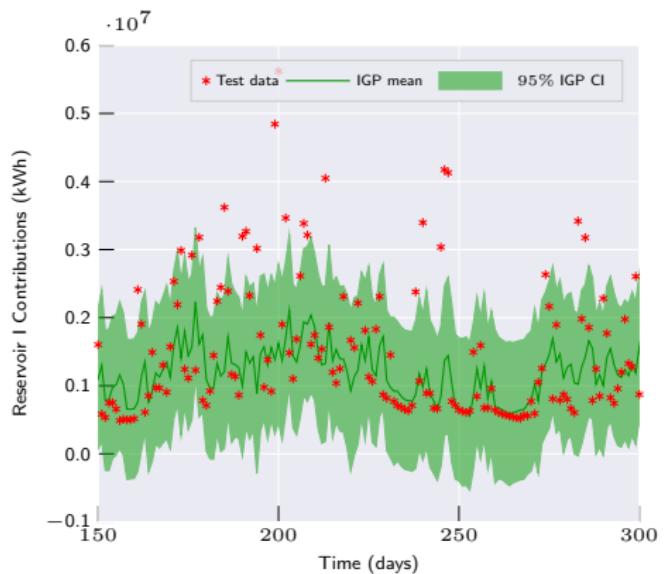
(a) MSLL



(b) NLPD

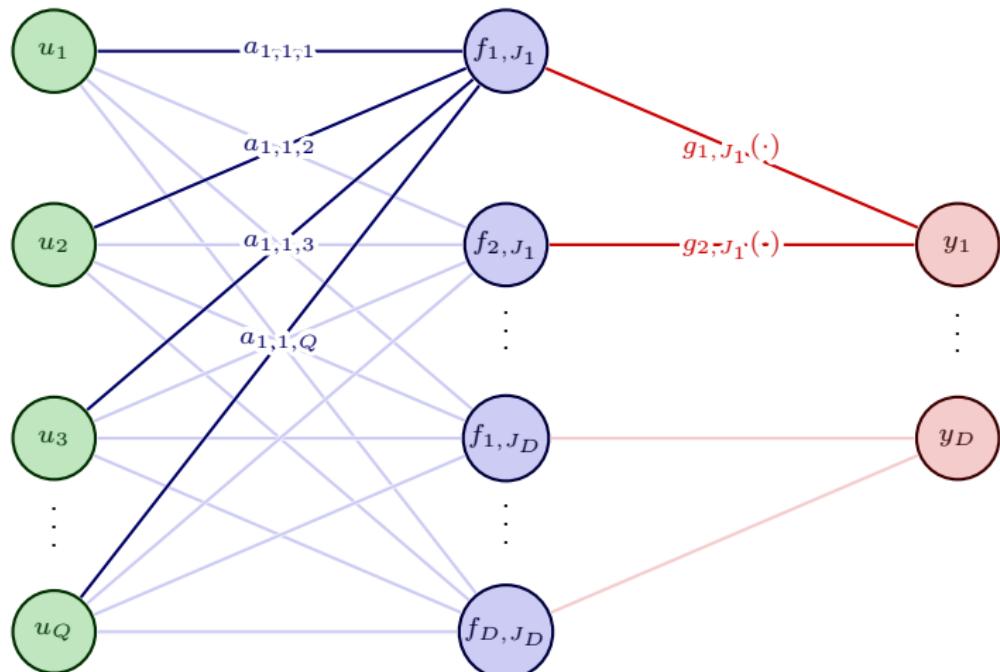
- The Adam+NG optimizer significantly boosts performance.
- LMCGP shows the most improvement, especially for larger horizons.

# One-day-ahead Models Forecasting



- The LMCGP model more accurately captures the peaks in the data.
- This is attributable to its ability to model complex behaviors through the incorporation of output correlations.

# Objective 3: Chained Correlated Gaussian Processes



**Independent Process (IGP)**  
 $u_q(\mathbf{x}) \sim \mathcal{GP}(0, k_q(\mathbf{x}, \mathbf{x}'))$

**Latent Process (LMCGP)**

$$f_{d,j}(\mathbf{x}) = \sum_{q=1}^Q a_{d,j,q} u_q(\mathbf{x})$$

**Likelihood (Chd GP)**

$$\mathbf{y} | \mathbf{f} \sim \prod_{d=1}^D p(\theta_{d,1}, \dots, \theta_{d,J_d})$$

$$\theta_{d,j} = g_{d,j}(f_{d,j})$$

# Variational Inference, ELBO and Predictive Distribution

We extent our variational inference, providing the following ELBO:

$$\begin{aligned} \mathcal{L} = & \sum_{d=1}^D \sum_{n=1}^N \mathbb{E}_{q(f_{d,1,n}), \dots, q(f_{d,J_d,n})} \{ \log p(y_{d,n} | f_{d,1,n}, \dots, f_{d,J_d,n}) \} \\ & - \sum_{q=1}^Q \text{KL} \{ q(\mathbf{u}_q) \| p(\mathbf{u}_q) \} \end{aligned}$$

The approximated posterior over test points is given by:

$$p(\mathbf{f}_* | \mathbf{y}) \approx q(\mathbf{f}_*) = \int p(\mathbf{f}_* | \mathbf{u}) q(\mathbf{u}) d\mathbf{u}$$

And the predictive distribution for a new output  $\mathbf{y}_*$ :

$$p(\mathbf{y}_* | \mathbf{y}) \approx \int p(\mathbf{y}_* | \mathbf{f}_*) q(\mathbf{f}_*) d\mathbf{f}_*,$$

The expectation values can be approximated via Monte Carlo methods. The complexity is now  $\mathcal{O}(JNQM^2)$  with  $J = \sum_d^D J_d$ .

# Model Setup

We again make use of squared exponential kernel to construct the covariance function and Adam + NG framework to train the models.

## Gaussian Likelihood: ChdGP Normal

$$p(\mathbf{y} \mid \mathbf{f}) = \prod_{d=1}^D \mathcal{N}(y_d \mid g_{d,1}(f_{d,1}), g_{d,2}(f_{d,2}))$$

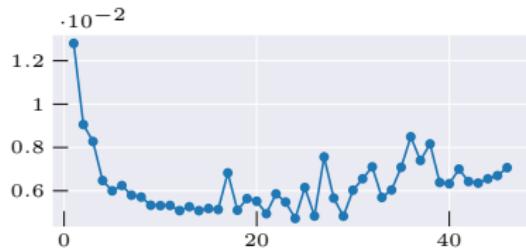
In this formulation,  $g_{d,1}(\cdot) = \cdot$ , while  $g_{d,2}(\cdot) = \ln(\exp(\cdot) + 1)$ .

## Gamma Likelihood: ChdGP Gamma

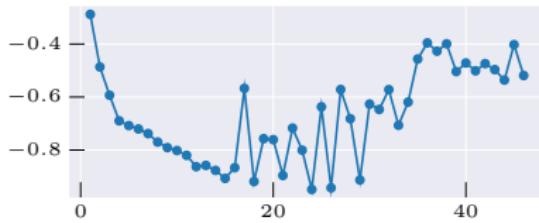
$$p(\mathbf{y} \mid \mathbf{f}) = \prod_{d=1}^D \text{Gamma}(y_d \mid g_{d,1}(f_{d,1}), g_{d,2}(f_{d,2}))$$

In this formulation  $g_{d,1}(\cdot) = g_{d,2}(\cdot) = \ln(\exp(\cdot) + 1)$ .

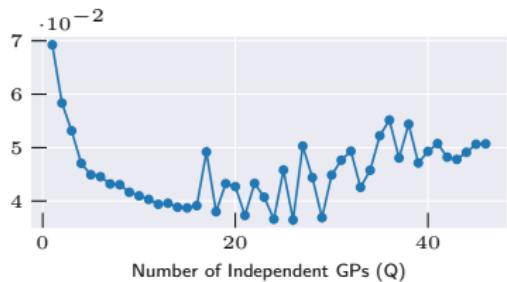
# Performance metrics vs Q



(a) MSE

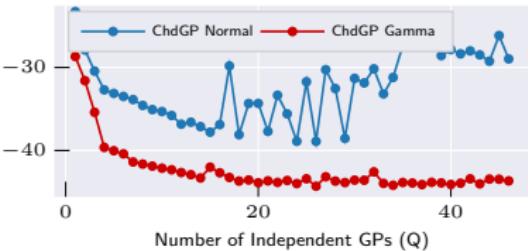


(b) MSLL



Number of Independent GPs (Q)

(c) CRPS

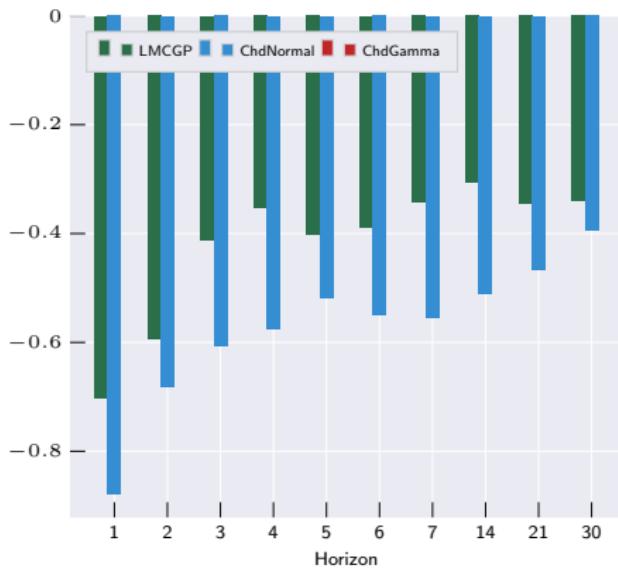


Number of Independent GPs (Q)

(d) NLPD

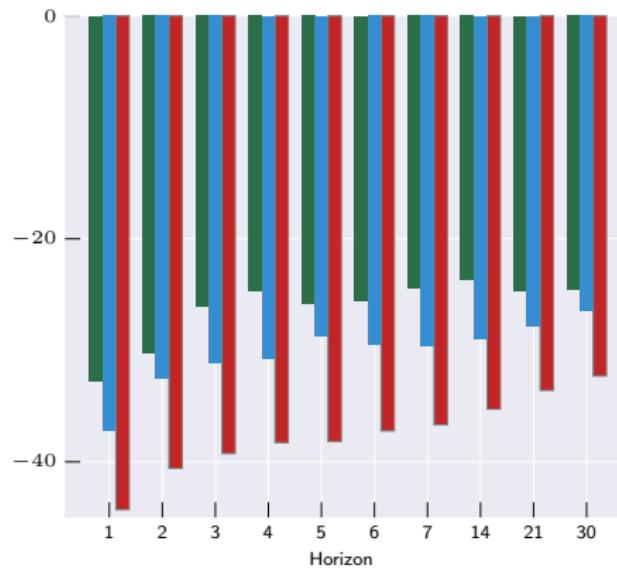
- The Gamma likelihood provides greater stability by offering a more precise description of the data.
- We select  $Q = 15$  for the Normal likelihood and  $Q = 26$  for the Gamma likelihood.

# LMCGP vs ChdGP across horizons



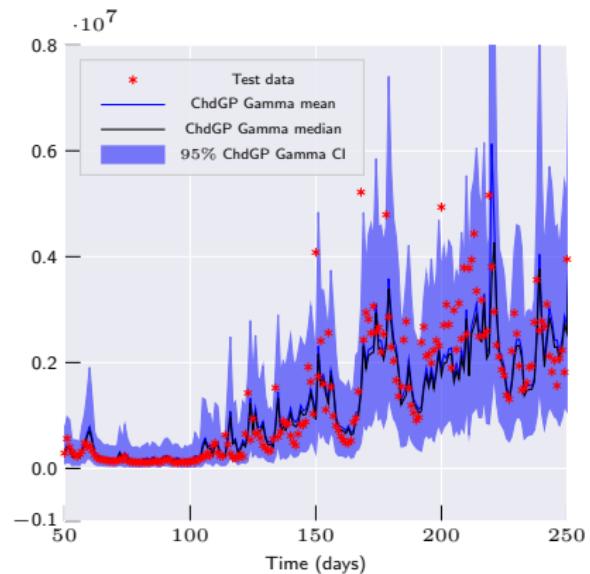
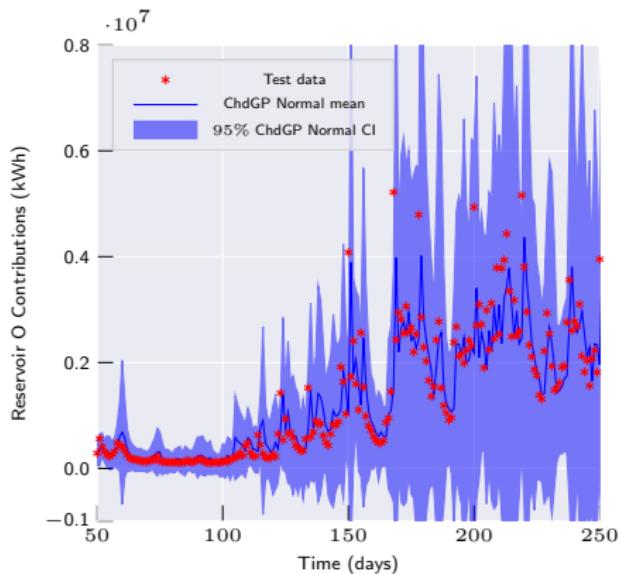
(a) MSLL

- The ChdGP model consistently outperforms the LMC GP model in all scenarios.
- The Gamma likelihood offers a superior explanation of the data, leading to enhanced performance.



(b) NLPD

# One-day-ahead Models Forecasting



- The predictive variance dynamically adapts to the complexity of uncertainty.
- For the ChdGP Gamma setting, the predictive distribution does not accommodate negative values.
- The presence of peaks results in a discrepancy between the mean and median.

# Conclusions

- The SVGP model demonstrated superior performance over state-of-the-art methods like LSTM and linear autoregression. It effectively captured nonlinear relationships and provided **uncertainty** estimates, outperforming alternative models across various forecasting horizons.
- By linearly combining independent GPs, we developed the LMCGP, enhancing prediction accuracy through shared task-specific information into a **multi-output** setting. To improve optimization, we introduced a combined Adam + NG framework, resulting in more stable solutions.
- Finally, to ensure positiveness **data constrain** streamflow predictions, we introduced the ChdGP with a Gamma likelihood. This model showed improved stability and accuracy, particularly in modeling peak occurrences, making it a suitable approach for hydrological forecasting.

# Products

## Article Published (Q1)



Article

### Scalable and Interpretable Forecasting of Hydrological Time Series Based on Variational Gaussian Processes

## Proceeding Published



Proceeding Paper

### Multi-Output Variational Gaussian Process for Daily Forecasting of Hydrological Resources <sup>†</sup>

## Registered Software *hydrogpower*



MINISTERIO DEL INTERIOR  
DIRECCION NACIONAL DE DERECHO DE AUTOR  
UNIDAD ADMINISTRATIVA ESPECIAL  
OFICINA DE REGISTRO

#### CERTIFICADO DE REGISTRO DE SOPORTE LOGICO - SOFTWARE

Libro - Tomo - Partida

13-98-419

Fecha Registro

29-may.-2024

# Future Work

- ① Integrate ChdGP-based streamflow forecasting models into the scheduling algorithms of thermoelectric power plants to optimize operational efficiency.
- ② Enhance the robustness of ChdGP forecasting models to handle missing data, ensuring reliable performance across diverse scenarios.
- ③ Combining ChdGP with Deep Gaussian Processes to improve predictive accuracy and model flexibility.

# Acknowledgments

- Gratitude to the Maestría en Ingeniería Eléctrica, graduate program of the Universidad Tecnológica de Pereira, for their support.
- Special thanks to the Minciencias project: “Desarrollo de una herramienta para la planeación a largo plazo de la operación del sistema de transporte de gas natural de Colombia”—código de registro 69982—part of the "CONVOCATORIA DE PROYECTOS CONECTANDO CONOCIMIENTO 2019 852-2019".
- Acknowledgment to the Automatics Research Group of the Universidad Tecnológica de Pereira for their invaluable support in the development of the project outcomes.

# References



Zaher Mundher Yaseen, Mohammed Falah Allawi, Ali A. Yousif, Othman Jaafar, Firdaus Mohamad Hamzah, and Ahmed El-Shafie.

Non-tuned machine learning approach for hydrological time series forecasting.

*Neural Computing and Applications*, 30(5):1479–1491, 2018.



Mahsa H. Kashani, Mohammad Ali Ghorbani, Yagob Dinپashoh, and Sedaghat Shahmorad.

Integration of volterra model with artificial neural networks for rainfall-runoff simulation in forested catchment of northern iran.

*Journal of Hydrology*, 540:340–354, 2016.



Muhammed Sit, Bekir Z. Demiray, Zhongrun Xiang, Gregory J. Ewing, Yusuf Sermet, and Ibrahim Demir.

A comprehensive review of deep learning applications in hydrology and water resources.

*Water Science and Technology*, 82(12):2635–2670, 08 2020.



Alexander Ley, Helge Bormann, and Markus Casper.

Intercomparing lstm and rnn to a conceptual hydrological model for a low-land river with a focus on the flow duration curve.

*Water*, 15(3), 2023.



Xiujie Wang, Yanpeng Wang, Peixian Yuan, Ling Wang, and Dongling Cheng.

An adaptive daily runoff forecast model using vmd-lstm-psو hybrid approach.

*Hydrological Sciences Journal*, 66(9):1488–1502, 2021.



Zheng Wang and Yuansheng Lou.

Hydrological time series forecast model based on wavelet de-noising and arima-lstm.

In *2019 IEEE 3rd Information Technology, Networking, Electronic and Automation Control Conference (ITNEC)*, pages 1697–1701, 2019.



Bibhuti Bhusan Sahoo, Ramakar Jha, Anshuman Singh, and Deepak Kumar.

Long short-term memory (lstm) recurrent neural network for low-flow hydrological time series forecasting.

*Acta Geophysica*, 67(5):1471–1481, 2019.



Yiran Li and Juan Yang.

Hydrological time series prediction model based on attention-lstm neural network.

In *Proceedings of the 2019 2nd International Conference on Machine Learning and Machine Intelligence, MLMI '19*, page 21–25, New York, NY, USA, 2019. Association for Computing Machinery.