Environmental Data Science (2025), 4: 1–29
doi:10.1017/eds.2025.xx

CAMBRIDGE UNIVERSITY PRESS

APPLICATION PAPER

Coupling Uncertainty-Aware Flow Forecasts with Policy Optimisation in Water Management

Dami Akinniyi[1]

[1]Data Science Program, Government of Manitoba, Winnipeg, MB, Canada

Received: 31 January 2020; **Revised:** 01 May 2020; **Accepted:** 06 May 2020

Keywords: Calibrated Ensemble Forecasting; Spatiotemporal Modelling, Gradient Boosting Trees, Policy Synthesis, Evolutionary Optimisation

Abstract

This paper presents the 1st place entry to the Capgemini Invent Water Scarcity Hackathon, demonstrating an application that couples uncertainty-aware flow forecasting with adaptive policy optimization in water management. The forecasting framework integrates feature selection strategies with gradient boosting trees to identify and quantify the influence of dynamic and static streamflow drivers. By incorporating causal structure and mutual information regression for feature selection, the approach supports robust spatiotemporal generalization while accounting for uncertainty through group-bounded conformal calibration.

In parallel, policy functions were designed and optimized in a multi-objective setting to balance ecological and economic trade-offs under resource scarcity. The optimization framework explores the impact of quotas and incentive mechanisms on system outcomes, showing how prioritizing specific actors can intensify ecological stress and economic imbalance, while subsidies encourage cooperation but risk undermining resilience in highly stressed environments. By contrast, adaptive and well-calibrated policies promote ecofairness, ecological survival, and system stability under uncertainty.

This application illustrates how spatiotemporally generalizable, uncertainty-aware forecasts and adaptive policy design can together inform resilient strategies for sustainable water management.

Impact Statement

This application paper demonstrates a framework for uncertainty-aware water management under scarcity. Streamflow forecasts are generated using chained quantile regression, with k k-means clustering applied to group spatial locations for calibration and reduce predictive uncertainty. Separately, quota- and incentive-based policy functions are optimized in a multi-objective framework to balance ecological resilience and economic benefit. This framework illustrates how data-driven forecasting and policy optimization can support robust decision-making in real-world water management scenarios.

1. Flow Forecasting

The first section focuses on building models to forecast water flow at stations on river basins in France (Adour-Garonne, Rhône-Mediterranean) and Brazil (Doce River). It involved predicting streamflow four weeks ahead using a combination of historical streamflow, observed weather, and short-term weather forecasts.

Traditional hydrological models, while effective, are often region-specific and computationally intensive, limiting their scalability and transferability across different catchments (Beven, 2012; Singh, 1995). In contrast, machine learning approaches leverage rich datasets with lagged and contextual features to capture complex temporal and spatial patterns, providing flexibility and predictive power across diverse hydrological settings (Mosavi et al., 2018; Kratzert et al., 2019). My approach builds on this paradigm by using gradient boosting trees (Prokhorenkova et al., 2018) combined with chained multi-output regression (Spyromitros et al., 2016) to model the spatio-temporal nature of streamflow forecasting, explicitly accounting for dependencies across multiple future time steps.

The rest of this project details the data exploration, causal analysis to identify drivers of streamflow, feature engineering, and modeling strategies tailored to handle differing dynamics across locations. It concludes with an evaluation of model performance and reflections on lessons learned for building robust, interpretable streamflow forecasting models.

1.1. Related Work

Conformal quantile calibration provides finite-sample guarantees for prediction intervals ensuring that the true outcome falls within the interval with a specified probability (Romano et al., 2019). However, global calibration can be inefficient in heterogeneous data, producing overly wide or under-confident intervals (Angelopoulos and Bates, 2022). This motivates approaches that calibrate within homogeneous groups, improving both efficiency and interpretability of predictive uncertainty.

Clustered and Class-Conditional Conformal Prediction

Several recent works have explored calibration within homogeneous clusters to address heterogeneity. Sousa et al. (2022) introduced an enhancement to conformalized quantile regression by incorporating k-means clustering based on feature relevance. This approach applies conformal steps within each cluster, allowing for adaptive prediction intervals that better account for heteroscedasticity. Hort et al. (2024) proposed clustered conformal prediction, grouping data points with similar nonconformity scores for cluster-level calibration. Ding et al. (2023) extended the concept to class-conditional conformal prediction, clustering classes with similar conformal scores to provide accurate coverage in problems with many classes. These methods demonstrate that clustered calibration—based on spatial, feature, or class similarity—enhances both reliability and interpretability of predictive uncertainty in heterogeneous settings.

Modular Conformal Calibration (MCC)

Modular Conformal Calibration (MCC) provides a flexible framework for recalibrating probabilistic forecasts from any base model by decomposing the calibration into modular components (Marx et al., 2022). Each module targets a specific source of variability, such as temporal trends, spatial heterogeneity, or feature-dependent differences. By adjusting nonconformity scores independently in each module, MCC produces calibrated prediction intervals that are more accurate and interpretable than global calibration alone. Its generality makes it applicable to a wide range of models and datasets, allowing fine-grained control over the calibration process.

Adaptive Conformal Regression with Jackknife+ Rescaled Scores

Adaptive Conformal Regression methods, such as Jackknife+ with rescaled scores, adjust nonconformity scores based on the magnitude of predictions and local variability. This approach ensures that intervals are appropriately wide where uncertainty is high and narrower where the model is more confident, improving efficiency and coverage under heteroscedastic data (Deutsmann et al., 2023).

Building on these ideas, we applied relative conformal calibration (RCC) by expressing nonconformity scores relative to predicted values and calibrating them within spatially homogeneous groups

identified via k-means clustering. This approach produces locally scaled, interpretable, and robust prediction intervals, ensuring reliable uncertainty estimates for spatiotemporally heterogeneous streamflow forecasts.

1.2. Data Description

Dataset Overview

The dataset is also organized in three main splits - train, evaluation and mini challenge sets, to support both temporal and spatio-temporal generalization tasks. The training set is used to train the model on historical data. The evaluation and mini-challenge sets consists of unseen temporal and Spatiotemporal stations to evaluate the model's performance on the public leaderboard.

Dataset Split	Country	Coverage	Num_Stations	Temporal	Spatiotemporal
Training	France, Brazil	1990–2003	39	✓	
Evaluation	France, Brazil	2004–2009	52 - (13 new)	✓	✓
Mini-challenge	Brazil	2011–2015	2		✓

Table 1. *Dataset Structure and Coverage Summary*

This setup poses two main challenges: temporal robustness across seen stations and generalization to unseen ones. This project addresses these through rigorous feature analysis, thoughtful feature selection, and the incorporation of causal effects to support reliable and responsive forecasting.

Dataset Structure

To simulate differing hydrological conditions, the test data is comprised of a 4-week historical dataset (streamflow and weather), a 1-week weather forecast, and a 4-week streamflow prediction, all segmented to reflect different hydrological conditions

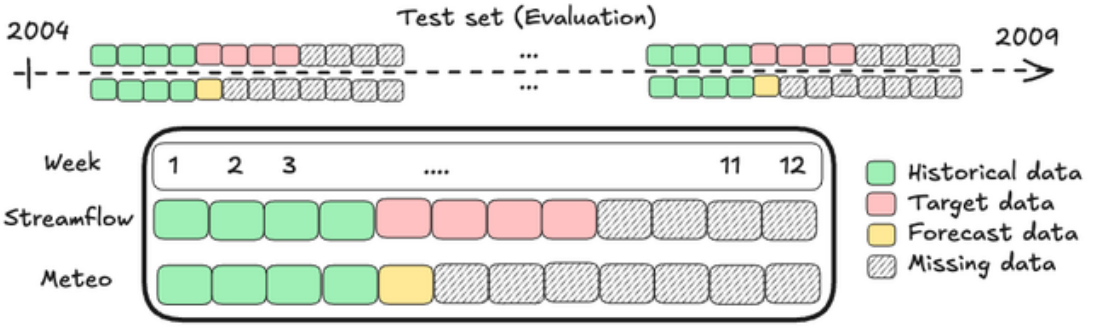


Figure 1. *Test Set Structure.*

Hydrographic Context

The stations in both locations (Brazil and France) are distributed within hydrographic areas that are defined hierarchically as *region, sector, sub-sector, and zone* (Figure 2). Regions and sectors vary widely in land area across the two countries, sub-sectors and zones exhibit greater spatial consistency, making them ideal for spatial feature aggregation.

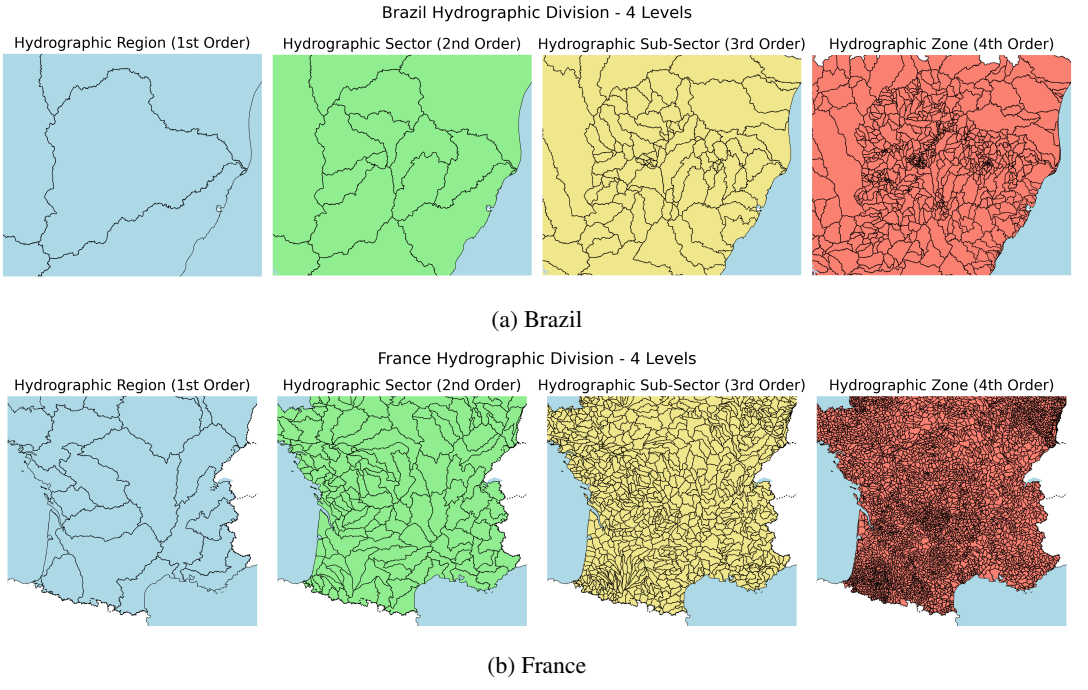


Figure 2. *Spatial delineation of hydrological areas in Brazil (top) and France (bottom), across four hierarchical geoscales: region, sector, sub-sector, and zone..*

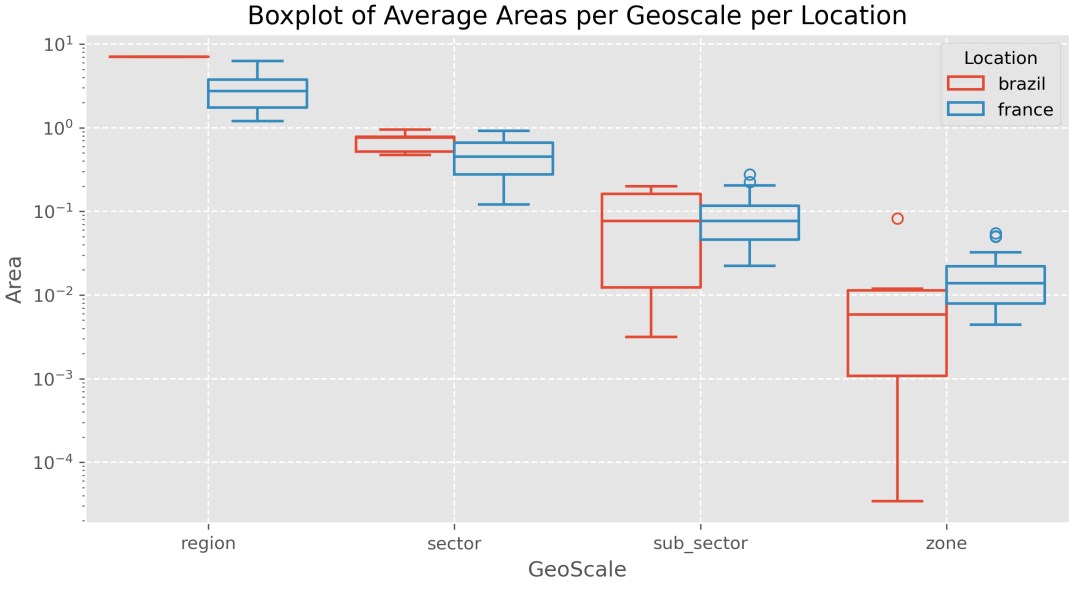


Figure 3. *Boxplot comparing areas in the four geoscales in Brazil and France. The maps illustrate spatial variation in size and structure across regions, sectors, sub-sectors, and zones, highlighting differences in spatial consistency between the two countries..*

1.2.1. Available Features

The dataset includes a rich set of static (non-temporal) and dynamic features that capture both spatio-temporal variability in hydrological conditions.

- **Static Features**:
  - Location features: Latitude, Longitude, Altitude
  - Hydrographic features: Region, Sector, Sub-sector, Zone, River Rank, Watershed Area
  - Soil composition
- **Dynamic Features**:
  - Historical Streamflow
  - Meteorological: Temperature (2m), total precipitation, evaporation, and soil moisture features

1.3. Methods and Approach

This project follows a structured but iterative process: data preprocessing, feature engineering, followed by data exploration and causal analysis and finally feature selection, modelling experiments and evaluation. This goal at all steps preceding the modelling phase is to uncover the most relevant features for predictive accuracy, while also ensuring that the model is interpretable and robust. The modelling phase focuses on iteratively testing different architectures and hyperparameters to find the best performing model.

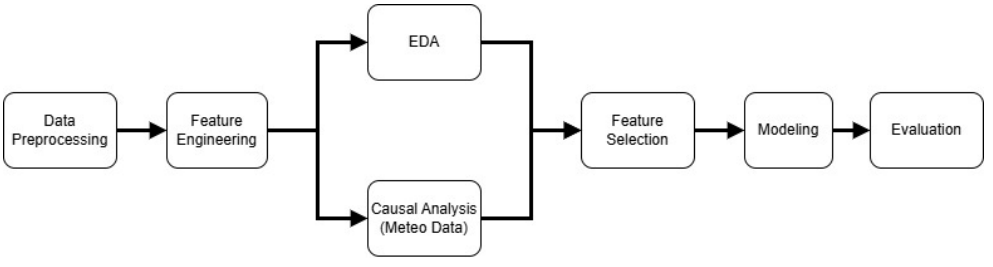


Figure 4. *Process Diagram.*

1.3.1. Data Preprocessing

Data preprocessing builds on the initial notebook provided in the Competition's Phase 1 GitHub repository which preprocesses soil and meteorological data for each station at the hydrological spatial scale. We extent this workflow to consolidate and structure information from multiple data sources for downstream modeling by:

- integrating local spatial aggregation scales - 50km and 100km buffer scales
- sampling directly from xarray files for soil and meteorological data. This skips the spatial interpolation step in the original notebook.
- modularizing and organizing all data preprocessing (for train, eval and mini-challenge) into executable scripts orchestrated by a Makefile pipeline for reproducibility and scalability.

1.3.2. Exploratory Data Analysis

We review the main feature groups to understand their distributions, variability, and potential predictive power.

- **Location Based Features**: Location features, (Latitude, Longitude, Altitude), are spatially diverse and capture distinct geographic patterns across regions (Figure 5). These features support location-aware modeling, and enhance generalization across spatial contexts. They provide a foundation for learning region-specific hydrological behavior, and are suitable proxies for such homogenous soil features (Arsenault et al., 2023).

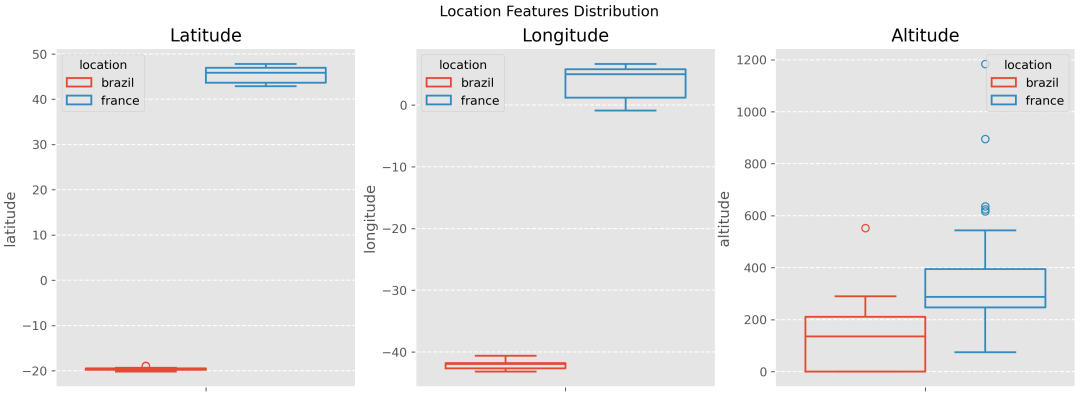
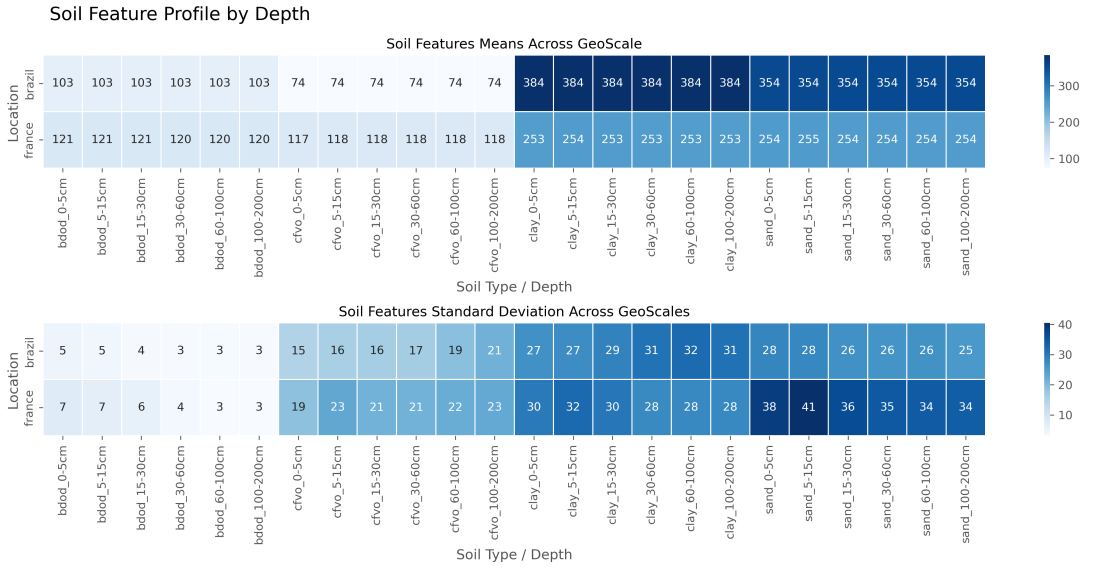


Figure 5. *Location-based features across the two countries..*

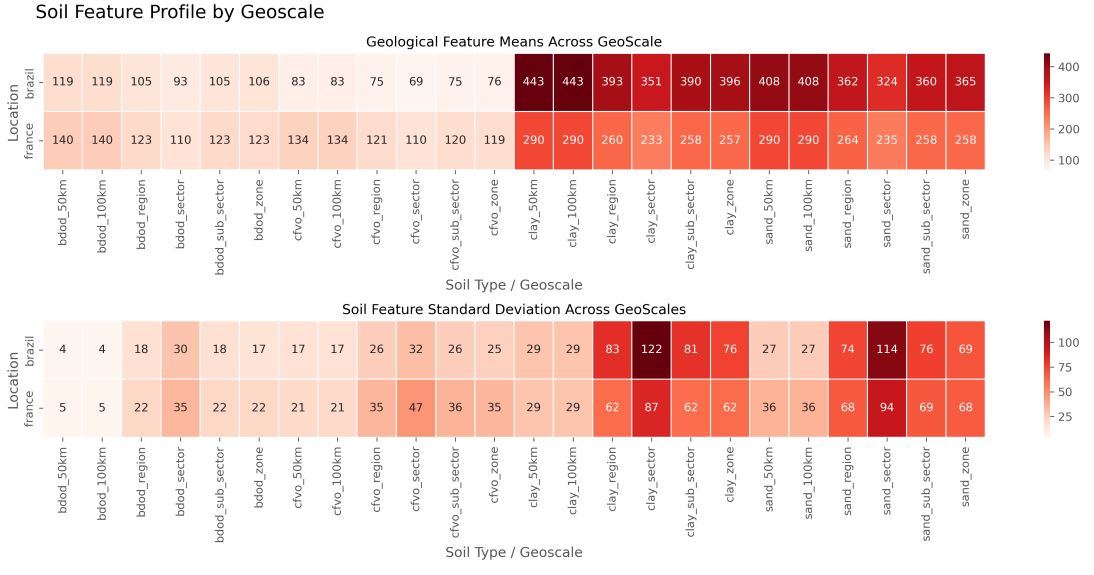
- **Soil Features**: Soil features are static, with no temporal variability, limiting usage to spatial modeling. This project builds on the soil aggregation methodology in the starter preprocessing notebook by extending soil aggregation to incorporate both hydrological scales and buffer zones around each station, enabling a multi-scale spatial understanding of soil properties. For each soil feature layer, both mean values and standard deviations were computed to assess variation across depth levels and geospatial units.
  The mean values within soil feature groups were generally consistent across depths and geoscales, indicating minimal variation in average soil characteristics. However, standard deviations showed greater variability, particularly in the upper soil layers and at the sector scale (Figure 6). These patterns were consistent across both regions, underscoring the need for careful feature selection to seperate discriminative features from noisy ones.

- **Climate Features**: Climate data is aggregated at both the provided hydrological unit scale and within surrounding buffer zones to capture localized spatial variation. Seasonal patterns are analyzed across locations, revealing consistent annual cycles across variables.
  Importantly, temperature(t2m) and evapotranspiration (evap) show opposing seasonal trends between northern and southern hemisphere sites due to their geographic positions. To address this, we account for hemispheric differences by enforcing location-specific constraints within the model, ensuring that the model interprets seasonal features within the appropriate spatial context. This prevents confusion from features that exhibit similar meanings but opposite temporal behaviors across regions.

- **Rivers and Watersheds**: Rivers are hierarchically ranked within each watershed, with lower values indicating higher-order rivers. Brazil uses ranks 3–13, while France uses 1–7; we align



(a) Soil Depth Profile



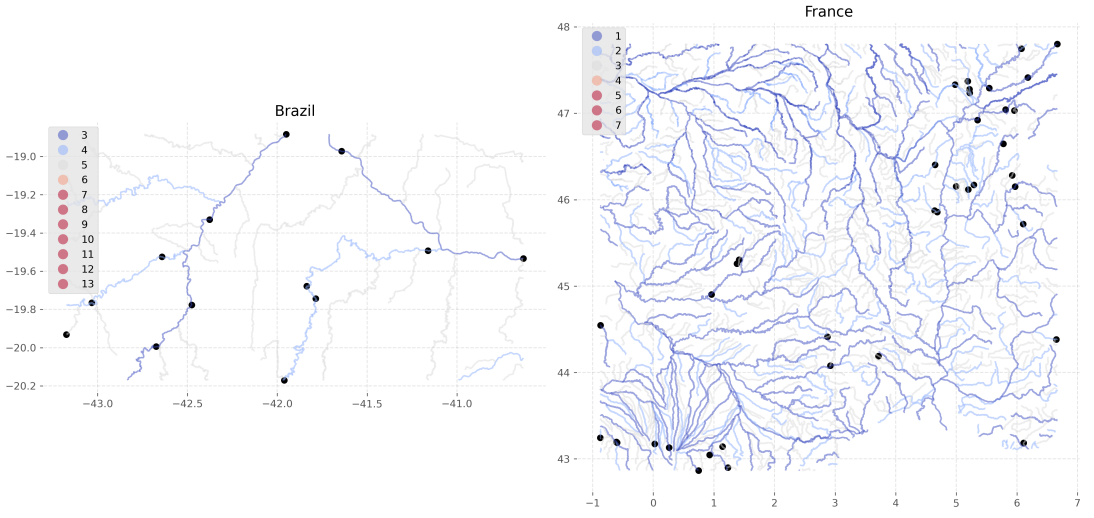
(b) Soil Geoscales Profile

Figure 6. *Top: Heatmap of soil feature means and standard deviations across depth per soil group. Bottom: Heatmap of soil feature means and standard deviations across geoscales per soil group. Fairly consistent means across depths and geoscales indicate minimal variaton in average soil characteristics.*

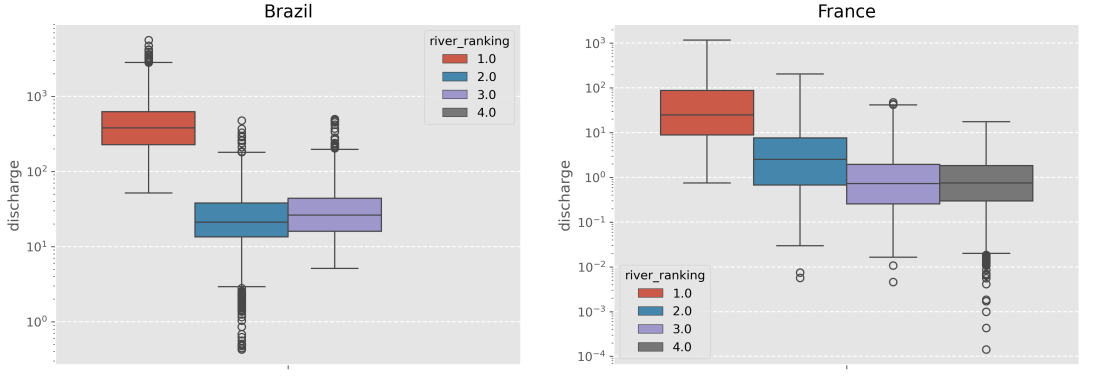
.

these by subtracting 2 from Brazil's ranks. Rank 1 rivers in both countries show the highest average discharge. This calibrated river ranking offers a valuable categorical spatial feature for modeling. Figure 7 shows river/station distribution and average discharge per river rank.

- **Water Flow Patterns**: Per-station analysis of water discharge reveals consistent flow patterns among stations located along the same river systems. Across these stations, discharge exhibits



(a) River / Stations Spatial Distribution



(b) Average Discharge per River Rank

Figure 7. *Top: Spatial distribution of top-ranked rivers in each location. Bottom: Average river discharge by rank, highlighting the dominance of higher-ranked rivers across countries.*

strong annual seasonality with little to no long-term trend. This behavior suggests empirical flow stationarity, indicating that historical discharge data remains representative of current conditions. To leverage this, we compute the annual empirical flow, the historical mean discharge for each week of the year, and include it as a model feature to capture expected seasonal behavior.

For stations not present in the training set, we estimate this feature by taking a weighted mean of the empirical flows from the two geographically nearest stations. This approach provides a reasonable proxy, allowing the model to generalize seasonal discharge patterns to ungauged locations.

- **Hydrographic Areas**: High missingness at finer hydrographic scales (*subsector and zone*) requires careful imputation, which we address using proximity-based filling from neighboring hydro regions.

1.3.3. Feature Engineering

Feature engineering plays a vital role in enhancing model performance by creating informative and predictive variables (Kuhn and Johnson, 2019). Techniques such as lag features, seasonal decomposition, and trend extraction are widely recognized for their effectiveness in time series forecasting (Kuhn and Johnson, 2019; Hyndman and Athanasopolous, 2018). In this work, we applied a range of techniques to capture spatial, temporal, and seasonal dynamics in the data

- **Lag Features:** Capture short-term dependencies (1, 2, 3, and 4 week lags) of key variables (e.g., climate or flow). 1, 2, 3 and 4 weeks.
- **Rolling Features:** Include rolling statistics (mean and variance) over time windows to capture temporal variability.
- **Seasonal Features:** Encode week-of-year using Gaussian basis functions (4 or 8 components) to represent cyclical seasonal patterns.
- **Temporal Features:** Include calendar-based indicators such as day, week, and month.
- **Categorical Features:** Metadata such as region, land use, and sensor type encoded as categorical variables.
- **Out-of-Fold (OOF) Features:** To avoid data leakage, certain features were computed using out-of-fold strategies during cross-validation. This includes:
  - *Empirical Flow:* Historical flow statistics aggregated using only training fold data.
  - *KMeans Clustering:* Cluster assignments based on spatial or environmental patterns, generated per fold.

Total number of features after engineering: *[insert total here]*

1.3.4. Feature Selection

We perform feature selection to improve model robustness and reduce overfitting by identifying the most relevant predictors from different feature groups:

- **Soil Features**: To reduce high dimensionality, we select a subset of informative and non-redundant soil variables using statistical and model-based methods. This streamlines the input space while retaining key spatial characteristics. We implement a two-step feature selection approach:

  1. **Mutual Information Regression (MIR) / Minimum Redundancy and Maximum Relevance (mRMR)**: We compute MIR scores in between each feature and the target variable. MIR, used in mRMR feature selection frameworks, is well-suited for identifying relevant soil features due to its ability to capture both linear and non-linear dependencies (Peng, 2005), which are common in hydrological and environmental data. We retain the top 10% of features with the highest MIR scores, ensuring that the most informative predictors are preserved.

  2. **Correlation Filtering:** Within the MIR-selected subset, we apply correlation filtering to remove redundant features. Feature pairs with a Pearson correlation coefficient greater than 0.9 are identified, and the feature with the lower MIR score (or lesser domain relevance) is removed. This reduces multicollinearity and improves model interpretability (Kuhn and Johnson, 2019).

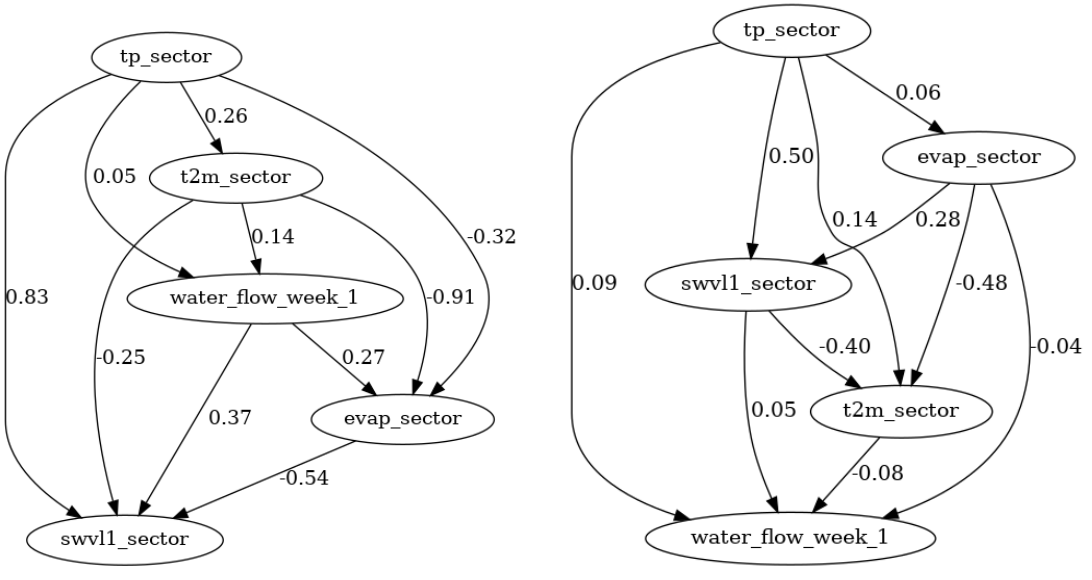
- **Climate Features**: We prioritize causally relevant climate variables cross different spatial scales to promote model robustness and generalizability. This ensures the model focuses on features with direct influence on hydrological responses. Similarly, we follow a multi-step approach:

    1. **Correlation Analysis:** We first analyze correlations at both local and global levels to identify general patterns and dependencies within the climate data.

    2. **Structured Causal Models:** To move beyond correlation and uncover causal effects, we apply structured causal models across spatial scales, including local buffer zones and national regions. We estimate Average Treatment Effects (ATE) for key climate variables, quantifying their direct impact on the target while adjusting for confounders. Directed Acyclic Graphs (DAGs) consistently identify temperature and precipitation as key drivers across regions, with soil water volume (SWVL) emerging as a consistent driver in France. These findings are supported by counterfactual analyses that visualize potential outcomes under different climate conditions.

    3. **Feature Importance:** Focusing on temperature, precipitation and soil moisture features, LightGBM is used to assess feature importance. Selecting the most predictive features within these groups at different scales helps the model better represent key climate drivers.

This process yields a compact, non-redundant, and highly predictive feature set that captures both direct and complex interactions with the target variable.



(a) Sector-level DAG for Brazil

(b) Sector-level DAG for France

Figure 8. *Learned Directed Acyclic Graphs (DAGs) at the sector level, showing causal relationships between climate variables and the target in Brazil and France..*

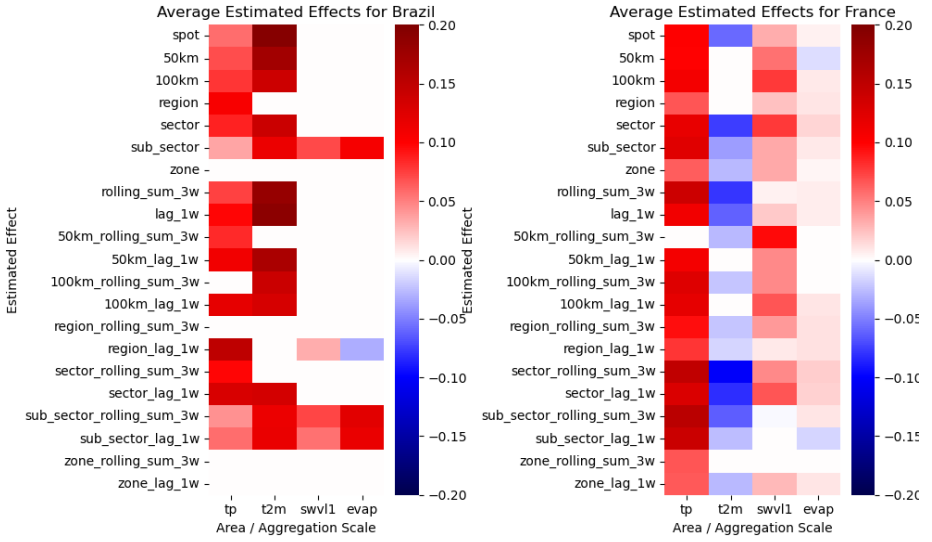


Figure 9. *Estimated Average Treatment Effects (ATE) of key climate variables across spatial scales. Strong causal influence is observed from temperature and precipitation..*

1.3.5. Modelling FrameWork

Gradient boosting methods have consistently demonstrated superior performance over other regression models in structured and environmental datasets (Shortridge et al., 2016; Kumar et al., 2023). The core model is a Gradient Boosting Trees (GBT) regressor trained in a chained forecasting setup, where predictions from previous timesteps (e.g., $\hat{y}_{t-1}, \hat{y}_{t-2}$) are chained into input features for predicting future values $\hat{y}_t$. This autoregressive structure enables multi-step forecasting while capturing temporal dependencies in the data.

To preserve temporal integrity, we use TimeSeriesSplit cross-validation. This ensures the model is always trained on past data and validated on future data. Each fold includes:

- Computation of out-of-fold (OOF) and out-of-year (OOY) empirical flow statistics to avoid leakage.
- Fold specific KMeans clustering to group stations into hydrologically and geographically similar clusters.
- Fold-specific model training.
- Validation residuals computed and conformal calibration factors computed per cluster and per week

At test time, predictions are aggregated by computing:

- The mean forecast.
- Quantiles (0.5, and 0.95) to characterize uncertainty.
- Qantile Conformal Calibration step is applied to improve the coverage and reliability of forecast intervals

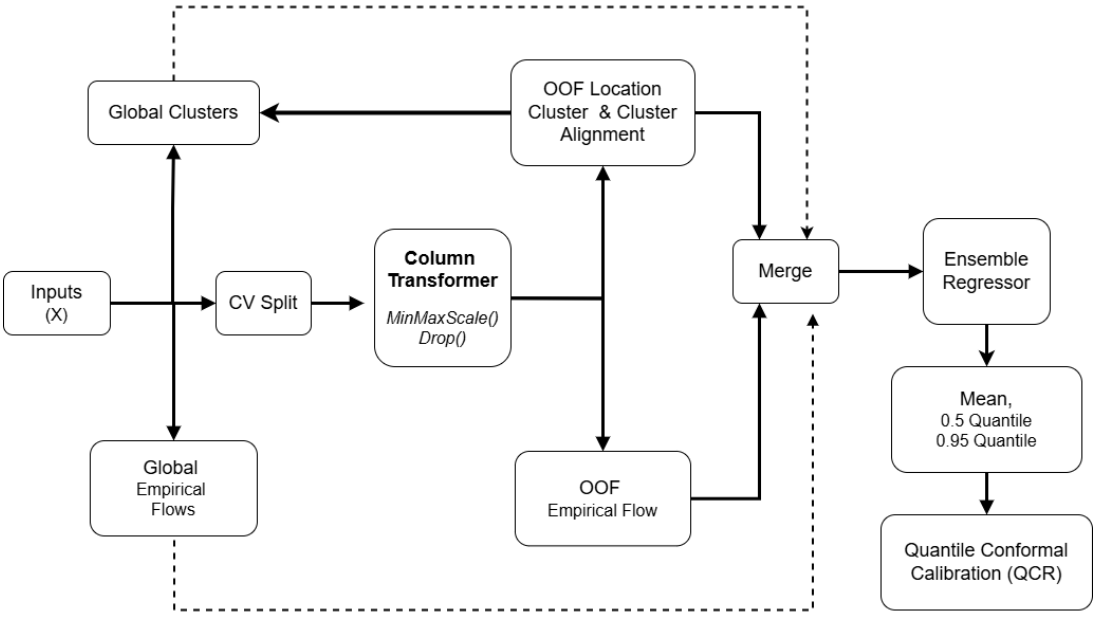


Figure 10. *Model Architecture.*

Algorithm 1 Chained Ensemble Forecasting

Input: Feature matrix X, target series y, number of folds C, number of ensemble models n, number of clusters k, conformal quantile level α
Output: Forecasts with calibrated quantile intervals $\hat{y}$
Compute global KMeans clusters on X_{train}
Compute global empirical flow features per station F
for each fold $c = 1, \ldots, C$ **do**
    Split data into $X_{\text{train}}^{(k)}, X_{\text{val}}^{(k)}$ using ContiguousTimeSeriesSplit
    Compute KMeans clusters on $X_{\text{train}}^{(k)}$ (per-fold clustering)
    Align fold cluster IDs to global cluster ID
    Compute OOF and OOY empirical flow features (excluding fold/year)
    MinMax scaling of features to [0, 1]
    for each ensemble model $i = 1, \ldots, n$ **do**
        Train chained GBT regressor on $X_{\text{train}}^{(k)}$
        Predict on $X_{\text{val}}^{(k)}$ and store $\hat{y}^{(c,i)}$
    end for
    Aggregate fold predictions: mean and quantiles
    Compute relative residuals $r^{(k)} = \frac{\hat{y}^{(k)} - y^{(k)}}{\hat{y}^{(k)} + \epsilon}$
    Compute $dt = Quantile_{1-\alpha}(r^{(k)})$ for each week w
    Store Conformal factors dt for each cluster c and week w
end for
Preprocess test data X_{test} using global clusters and empirical flow features
Predict on X_{test} using the trained ensemble models
Compute mean and quantiles of predictions $\hat{y}_{\text{test}}$
Smooth conformal factors using weekly temporal blending.
Calibrate quantile intervals using smoothed conformal factors.
Return ensemble forecasts with calibrated intervals

1.3.6. Modelling Component Details

ContiguousTimeSeriesSplit Cross Validation

This validation strategy builds upon the `TimeSeriesSplit` approach from SCIKIT-LEARN, where each fold's training set is a strict superset of the previous one, preserving the temporal ordering of the data. However, our method introduces two key modifications:

- **Contiguous Validation:** Instead of using a fixed-size validation block for each fold, we validate on all remaining data after the training split. This allows the model to be assessed on the full available future horizon, improving robustness and generalization assessment.

- **Minimum Training Size:** A constraint is imposed such that the first training fold must contain at least 50% of the dataset. This ensures sufficient historical context is available for the initial training phase, preventing unstable model estimates due to data sparsity.

The fold structure for a 5-fold setup with years ranging from 1990 to 2004 is defined as follows:

Fold	Training Years	Validation Years
1	1990–1998	1999–2004
2	1990–1999	2000–2004
3	1990–2000	2001–2004
4	1990–2001	2002–2004
5	1990–2002	2003–2004

Table 2. *Contiguous expanding time series cross-validation structure.*

This approach ensures that:

- Each model is trained only on past data relative to the validation set.
- The training window expands over time, simulating a real-world forecasting scenario.
- The evaluation is performed on all future data from the training endpoint, leading to more realistic performance metrics.

Data-Aware Global Clustering and Alignment

To group stations into hydrologically similar regimes, we apply KMeans clustering on the full training data using a fixed number of clusters k. This yields a global clustering structure that informs both model predictions and residual calibration.

To avoid data leakage in time series cross-validation, KMeans is recomputed independently within each training fold. Cluster centers vary between the global and fold-specific kMeans cluster ids, a well known label-switching issue in clustering. To ensure consistency of cluster assignments across cross-validation folds, we align fold-specific cluster ids with global cluster by solving an optimal assignment problem. Such alignment has been widely used in cluster ensembles and partition aggregation frameworks (Stephens, 2000; Strehl et al., 2003), with the optimization typically solved using the Hungarian algorithm (Kuhn, 1955).

The alignment minimizes the total euclidean distance between fold-specific cluster centers C^{fold} and global cluster centers C^{global}:

$$\min_{\pi \in \text{Perm}(k)} \sum_{i=1}^{k} \left\| C_i^{\text{fold}} - C_{\pi(i)}^{\text{global}} \right\|_2 \tag{1}$$

where π is a permutation of cluster indices. The aligned cluster IDs are then used to condition predictions and compute localized residual statistics.

Algorithm 2 Data-Aware Global Clustering and Alignment

Input: Training data X, number of clusters k, number of CV folds F
Fit global KMeans on X to obtain centers C^{global} and cluster IDs
for each fold $f = 1$ to F **do**
    Extract fold-specific training data $X^{(f)}$
    Fit KMeans on $X^{(f)}$ to get centers $C^{(f)}$
    Compute cost matrix $M_{ij} = \|C_i^{(f)} - C_j^{\text{global}}\|_2$
    Use Hungarian algorithm to solve:

$$\min_{\pi \in \text{Perm}(k)} \sum_{i=1}^{k} M_{i,\pi(i)}$$

    Map fold-specific cluster IDs using optimal assignment π
    Assign aligned cluster labels to $X^{(f)}$
end for
Output: Data with global-aligned cluster IDs for each fold

Empirical Flow Features

Empirical flow statistics are computed as the mean discharge per station and week, excluding the target year to prevent leakage. During training, global empirical flow is calculated and stored. After data splitting, empirical flow is recomputed using only the training fold data.

The inference function `_get_empirical_prior` assigns empirical flow values for each data point as follows:

1. If the station exists in the training priors, assign its empirical flow directly.
2. Otherwise, estimate empirical flow by weighted averaging of the two nearest neighbor stations:
    - Neighbors are selected based on river proximity, river ranking, and geographic location.
    - Euclidean distances on latitude and longitude are computed to find the nearest neighbors.
    - Weights are inverse-distance weighted to combine neighbors' empirical flows smoothly.

This method provides stable, leakage-free empirical priors for both seen and unseen stations during training and inference.

Relative Quantile Conformal Calibration

To address heteroscedasticity and the varying magnitudes of streamflow across regions and seasons, we propose **Relative Quantile Conformal Calibration (RQCC)** — an extension of conformal quantile calibration that adjusts prediction intervals using residuals scaled relative to the target value.

In standard quantile conformal prediction, coverage is achieved by computing residuals between predicted quantile bounds and observed values. For a model that outputs lower and upper quantile predictions $\hat{y}_t^{\text{lower}}$ and $\hat{y}_t^{\text{upper}}$, the standard conformal residual is:

$$r_t = \min\left(\hat{y}_t^{\text{upper}} - y_t,\ y_t - \hat{y}_t^{\text{lower}}\right) \tag{2}$$

Algorithm 3 Empirical Flow Feature Extraction

Require: Dataframe X with columns `week, year, station_code`, empirical flow priors `priors`, station metadata `stations_gdf`
Ensure: Dataframe X augmented with empirical flow prior `empirical_flow`

Assert columns `week, year, station_code` exist in X
Copy X to avoid modifying input
for all unique stations s in X **do**
    if s in columns of `priors` **then**
        Extract empirical flow data for station s
    else
        Find neighbors for s by:

- Filtering stations with the same river and similar river ranking and location
- Calculating Euclidean distance based on latitude and longitude
- Selecting two nearest neighbors

        Compute inverse distance weights for neighbors
        Compute weighted average of neighbors' empirical flow by week
    end if
    Append station empirical flow data to results
end for
Merge empirical flow results into X by `station_code` and `week`
return augmented X

Using these residuals, the calibrated prediction interval is:

$$\left[\hat{y}_t^{\text{lower}} - q_{1-\alpha},\ \hat{y}_t^{\text{upper}} + q_{1-\alpha}\right] \tag{3}$$

where $q_{1-\alpha}$ is the $(1-\alpha)$-quantile of residuals from a held-out calibration set.
However, in real-world hydrological applications, streamflow values can vary drastically between basins — from small ephemeral streams to major rivers. Applying absolute residuals in such cases results in mis-scaled intervals: overly wide in high-flow regimes and too narrow in low-flow areas. This undermines the interpretability and sharpness of calibrated intervals.

To normalize across such variation in target magnitudes, we define **relative residuals**:

$$r_t^{\text{rel}} = \frac{\min\left(\hat{y}_t^{\text{upper}} - y_t,\ y_t - \hat{y}_t^{\text{lower}}\right)}{y_t + \epsilon} \tag{4}$$

where $\epsilon > 0$ is a small constant (e.g., 10^{-6}) added for numerical stability.
Residuals are then grouped by **cluster** c (e.g., hydrological regime) and **timestep** t (e.g., season), and the conformal quantile is computed per group:

$$\delta_{c,t} = \text{Quantile}_{1-\alpha}\left(\left\{r_t^{\text{rel}}\right\}_{i \in \text{cluster } c}\right) \tag{5}$$

Finally, these relative quantile factors $\delta_{c,t}$ are used to expand the predicted interval proportionally:

$$\left[\hat{y}_t^{\text{lower}} - \delta_{c,t} \cdot (y_t + \epsilon),\ \hat{y}_t^{\text{upper}} + \delta_{c,t} \cdot (y_t + \epsilon)\right] \tag{6}$$

This formulation maintains correct coverage while adapting to local flow characteristics, producing interpretable and consistent intervals across space and time.

1.4. Experiments and Results

1.4.1. Experimental Setup

We evaluate two variants of our model to assess the impact of soil features:

- **Model A:** CatBoost model with the full feature set, including soil features.
- **Model B:** CatBoost model excluding all soil-related features.

Both models are trained using CatBoost with a quantile regression objective (quantile $\alpha = 0.1$). We employ a 5-fold contiguous cross-validation scheme with fold expansion. The splits are made by year to preserve temporal order and prevent data leakage. Model performance is evaluated using:

- Negative Log-Likelihood (NLL)
- Interval Coverage (calibration of predicted quantiles)

This setup allows comparison of predictive accuracy, uncertainty quantification, and calibration between models with and without soil features.

Model	**OOF NLL**	**Leaderboard Score**		
		Coverage	**Temporal Split**	**SpatioTemporal Split**
A - *With Soil Features*	**1.568**	**0.904**	**2.63**	3.18
B - *Without Soil Features*	1.572	0.900	2.65	**3.05**

Table 3. *Performance Metrics Comparison of Models With and Without Soil Features*

1.4.2. Explainability

In this study, we focus on feature importance analyses aggregated across folds and weeks to interpret the model's global behavior and temporal variation. While advanced explainability methods like SHAP values provide detailed local and global insights, they were not used here due to computational constraints and the complexity introduced by our ensemble model with multiple folds and repetitions.
Instead, we rely on aggregated feature importance scores from the feature selection phase to capture variable influence on model predictions. Further explainability analyses, including residual diagnostics and local interpretability methods, remain valuable avenues for future work to enhance understanding of model performance and error characteristics.

Global Feature Importance

Feature importances are computed using the average feature importances across all models in the ensemble.

Top Feature Importances per Fold and Week

Feature importances are computed for each fold separately, allowing us to observe how feature relevance varies across different training and validation sets. The top 10 features per fold are summarized in Table 4, and Table 5 showing both their importance values and ranks within each fold and week respectively.
The tables show that water flow lag features consistently rank among the top predictors across folds and weeks, indicating their strong temporal influence on streamflow. Empirical flow also remains a key feature, reflecting its importance as a hydrological prior. Other features such as rolling statistics and regional climate variables vary in importance depending on the fold or week, suggesting that local hydrological conditions and seasonal patterns significantly affect model predictions.

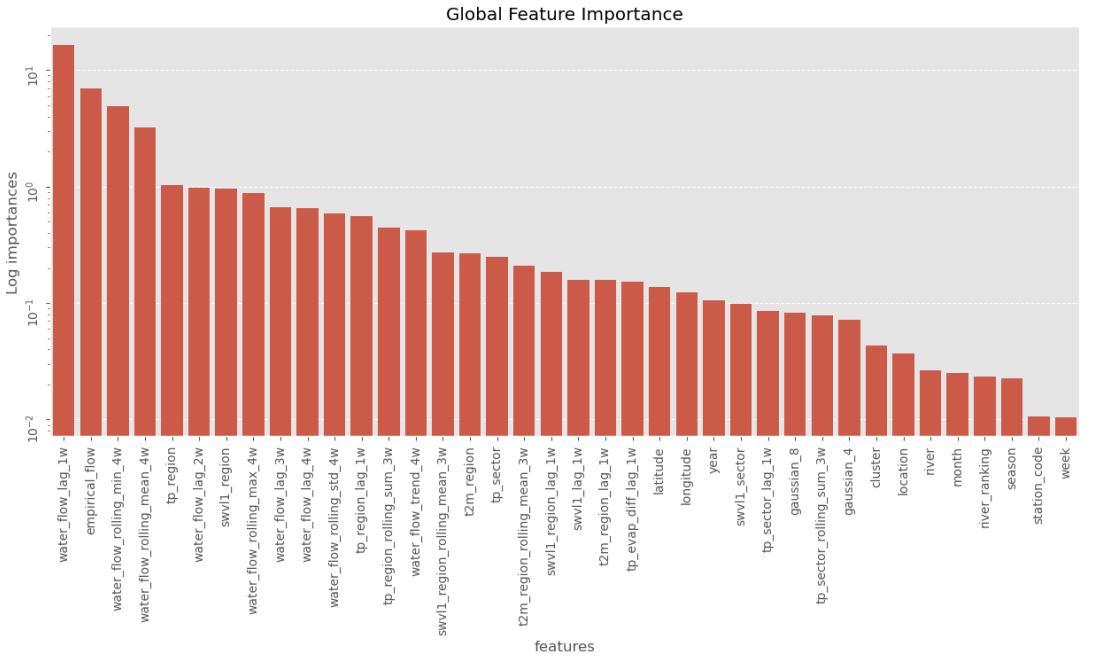


Figure 11. *Global Feature Importance for the Model with Soil Features. The top 10 features are shown, with empirical flow and water flow lag features dominating the importance scores..*

Feature	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
water_flow_lag_1w	15.29 (1)	16.73 (1)	16.76 (1)	16.51 (1)	16.55 (1)
empirical_flow	6.75 (2)	7.06 (2)	7.39 (2)	5.97 (2)	7.46 (2)
water_flow_rolling_min_4w	6.12 (3)	4.98 (3)	5.44 (3)	4.18 (4)	3.85 (3)
water_flow_rolling_mean_4w	3.66 (4)	3.36 (4)	2.78 (5)	3.70 (3)	2.56 (4)
tp_region	1.02 (7)	1.03 (5)	0.96 (6)	1.08 (6)	1.06 (6)
swvl1_region	0.90 (8)	0.99 (6)	0.92 (7)	1.17 (5)	0.80 (7)
water_flow_lag_2w	1.12 (6)	0.86 (9)	0.92 (8)	0.88 (7)	1.15 (5)
water_flow_lag_3w	0.75 (9)	0.88 (7)	0.71 (10)	0.77 (8)	0.50 (9)
water_flow_lag_4w	0.64 (10)	0.64 (10)	0.75 (9)	0.77 (9)	0.62 (8)
water_flow_rolling_max_4w	1.62 (5)	0.64 (11)	1.02 (5)	0.72 (10)	-
water_flow_rolling_std_4w	-	0.86 (8)	-	0.68 (11)	-
tp_region_lag_1w	-	-	-	-	0.50 (10)

Table 4. *Top feature importances (values) and ranks (in parentheses) per fold. "-" indicate feature not in top 10 for corresponding fold.*

1.4.3. Robustness

Given the model's complexity and reliance on diverse features, we assess robustness across three dimensions: out-of-fold consistency, spatiotemporal generalization, and interval reliability.

Out-of-Fold Validation

We evaluate the model's performance across five validation folds to ensure stable behavior across different training splits. Table 6 reports the Mean Absolute Error (MAE) for each week.

The model maintains consistent performance across folds, with lower errors in later folds—likely due to more historical data becoming available (see Table 6). Errors increase with forecasting horizon, reflecting compounding uncertainty and growing temporal distance from observed lags.

Feature	Week 1	Week 2	Week 3	Week 4
water_flow_lag_1w	15.84 (1)	17.22 (1)	17.05 (1)	15.35 (1)
empirical_flow	6.91 (2)	7.02 (2)	6.32 (2)	7.45 (2)
water_flow_rolling_min_4w	5.86 (3)	5.34 (3)	4.21 (3)	4.24 (3)
water_flow_rolling_mean_4w	3.57 (4)	3.17 (4)	3.28 (4)	2.83 (4)
tp_region	1.05 (7)	0.96 (5)	1.02 (6)	1.1 (5)
swvl1_region	0.95 (8)	0.93 (6)	1.13 (5)	0.81 (7)
water_flow_rolling_max_4w	1.34 (5)	0.91 (7)	0.81 (8)	-
water_flow_lag_2w	1.11 (6)	0.88 (8)	0.9 (7)	1.05 (6)
water_flow_lag_4w	-	-	0.81 (9)	0.6 (8)
tp_region_rolling_sum_3w	-	-	-	0.52 (9)
water_flow_rolling_std_4w	0.93 (9)	-	-	0.52 (10)
water_flow_lag_3w	0.82 (10)	0.71 (9)	0.65 (10)	-
tp_region_lag_1w	-	0.62 (10)	-	-

Table 5. *Top feature importances (values) and ranks (in parentheses) per week. "-" indicates feature not in top 10 for corresponding week.*

Fold	Week 1	Week 2	Week 3	Week 4
fold_1	14.300	17.446	24.149	27.747
fold_2	14.387	17.384	24.414	27.469
fold_3	14.008	17.205	24.654	27.470
fold_4	14.125	17.065	25.264	28.311
fold_5	12.005	16.657	23.105	22.378

Table 6. *Mean Absolute Error (MAE) per fold and week on the out-of-fold validation set.*

Spatiotemporal Generalization

To evaluate the model's ability to generalize across spatial and temporal domains, we introduce a held-out spatiotemporal validation split. This includes four stations from France and two from Brazil, excluded entirely from training. Table 7 shows MAE across weeks on this combined set.

Fold	Week 1	Week 2	Week 3	Week 4
fold_1	14.677	18.508	26.391	29.732
fold_2	14.584	18.778	26.164	30.165
fold_3	14.893	18.533	26.309	29.712
fold_4	15.025	19.117	27.309	30.653
fold_5	13.129	17.930	24.705	23.950

Table 7. *Aggregated spatiotemporal generalization performance on held-out stations and temporal split.*

As shown in Table 7, performance degrades slightly on the spatiotemporal set, though it remains within acceptable bounds. To provide fairer comparisons across stations, we report station-normalized MAE in Table 8.

Split	Week 1	Week 2	Week 3	Week 4
Temporal_split	**0.287**	**0.464**	0.569	**0.605**
Spatio_temporal_split	0.348	0.485	**0.564**	**0.605**

Table 8. *Station-normalized MAE across weeks for temporal and spatiotemporal splits.*

The normalized MAE in Table 8 shows slightly better results on the temporal split, but the gap is narrow—highlighting the model's generalization ability even on unseen spatial regions.

Prediction Interval Width

We also assess uncertainty calibration via the prediction interval width, normalized per station. Results are presented in Table 9.

Split	Week 1	Week 2	Week 3	Week 4
Temporal_split	1.178	1.671	1.889	1.969
Spatio_temporal_split	1.188	1.651	1.746	1.763

Table 9. *Station-normalized prediction interval width across weeks for temporal and spatiotemporal splits.*

As shown in Table 9, prediction intervals are slightly tighter for spatiotemporal splits, which may lead to undercoverage and elevated NLL scores. This suggests that while the model is confident, it may be underestimating uncertainty on unseen stations.

Discussion on Generalization and Uncertainty

The observed gap between out-of-fold and leaderboard scores highlights key opportunities for improvement. Strong out-of-fold performance (Table 6) provides a solid foundation, while the results on unseen stations (Tables 8, 9) emphasize areas for better spatial generalization and uncertainty calibration.

Enhancements such as stronger regularization, increased training diversity, and refined spatiotemporal validation can help bridge this gap. Overall, the model demonstrates strong robustness with promising potential for deployment across varied hydrological settings.

1.4.4. Frugality Analysis

The final model is an ensemble composed of 15 base chained regression learners per fold, trained across K cross-validation splits, resulting in a total of $15 \times K \times 4$ models. This structure enhances predictive performance and stability but introduces additional computational and memory overhead.

Data Usage and Efficiency

The model uses moderate-sized datasets with internally generated features such as out-of-fold (OOF) predictions and empirical flow estimates. While scalable, the generation of OOF features adds to the overall training complexity. No external datasets are required beyond the primary climate and site-level inputs.

Training Runtime and Inference Efficiency

Training time increases with the number of base learners but can be parallelized across folds and ensemble members. Reducing the folds and/or number of models per fold substantially decreases latency and memory usage, making the model more deployable in resource-constrained environments.

Table 10. *Model Configuration Trade-offs*

Configuration	OOF NLL	Models	Train (m)	Inference (ms)	Size (MB)
Full Ensemble (15/fold)	1.57	300	20	2820	~78
Reduced Ensemble (10/fold)	1.58	200	14	2260	~58.4
Single Fold (15)	1.588	60	8	-	~27.18

2. Part 2 - Policy Design and Optimizattion

The second phase of the competition focused on the design and optimization of policies to govern the allocation of shared water resources in a game theory simulation environment.

Effective water policy must balance ecological protection, equitable access, and economic stability, reflecting the broader principles of sustainable water management that call for joint consideration of social, economic and environmental needs (Wang, 2025). Our approach emphasizes fairness—allocating water by priority—while ensuring sustainability for downstream ecosystems and resilience under uncertainty.We iteratively test the simulation environment, manually adjusting key parameters (e.g., demand and penalty multipliers) to understand how policy rules affect economic and ecological outcomes. This process aligns with simulation-based optimization approaches in water resource research, which are often used to explore policy trade-offs under uncertainty (**?**Tang et al., 2021; Fu et al., 2020).

We observed the incentive function shapes drive cooperation and economic stability, while quota allocations and actor priorities impact ecological health. Based on these insights, we design adaptive policies that respond to environmental signals, guiding agents toward sustainable behavior. Our framework includes:

- Simulation Environment Parallelization
- Quota Function: A priority-aware allocation mechanism with configurable thresholds.
- Incentive Function: A fine/subsidy structure with both stepwise and smooth variants.
- Multiobjective Optimization : A Pareto-front approach using Optuna to discover trade-offs between economic efficiency and ecological viability.
- Policy Regulator Framework: A python class that encapsulates the policy design and optimization process to yield simulation-ready policy functions.

This framework allows us to explore how adaptive, minimalistic rules can guide cooperation and robustness in a complex, uncertain environment.

2.1. Simulation Environment Parallelization

At the heart of the policy evaluation workflow is the *multi_scenario_analysis* function, responsible for executing simulations across a range of behavioral and environmental configurations. This function plays a central role in assessing the robustness and generalizability of policy designs, ensuring that any proposed strategy is evaluated across diverse and uncertain scenarios.

The original function is effective in exploring these scenarios sequentially. The parallelization of the *multi_scenario_analysis* function is a crucial step in optimizing the policy evaluation process. By leveraging Python's concurrency capabilities, we can execute multiple scenarios simultaneously, significantly reducing the time required for each evaluation cycle. This is particularly important in the context of policy optimization, where rapid feedback on policy performance is essential for iterative refinement and decision-making.

We also enhance the function's flexibility to allow specifying custom scenario subsets—enabling targeted evaluation and more efficient use of compute resources during optimization. These enhancements integrate seamlessly with optuna's multi-objective optimization framework, allowing for large-scale policy search.

Together, these improvements reduce the runtime of a 700 turns, 10 iterations simulation from approximately 3 hours to less than 20 minutes per full evaluation cycle.

2.2. Quota Function Design

In water-stressed systems, resource allocation must carefully balance fairness, actor resilience, and ecological resilience — ensuring that critical needs are met without compromising the long-term health of the environment. This becomes especially important during crisis conditions, where water scarcity intensifies and allocation decisions carry systemic consequences.

To address this challenge, we propose a quota policy function for allocating water across multiple actors with varying demands and priorities.

2.3. Policy Mechanism

The function is guided by three key parameters:

- **rate** (r): The baseline proportion of each actor's demand that is fulfilled during non-crisis periods. This controls total water use and supports ecosustainability by preventing excessive extraction even under normal conditions.
- **priority_effect** (p_e): A tuning parameter that controls how much high-priority actors are favored during crises. When priority effect is 0, non-low-priority actors share water equally (relative to demand). Higher values increasingly concentrate allocation toward the highest-priority actors. Low-priority actors receive no water during alert and crises, based on prior analysis showing they maintain high buffer-to-demand ratios and can self-sustain through temporary shortages.
- **crisis_rate** (r_c): The overall reduction in water availability during crisis or alert conditions, defining what percentage of each actor's demand (before applying priority) is considered for allocation. This simulates external constraints and enforces ecosustainability by capping water use according to environmental thresholds.

2.4. Mathematical Formulation

Let:

- $\mathbf{D} = (D_1, D_2, \ldots, D_n)$: vector of average water pumped per actor.
- $\mathbf{P} = (P_1, P_2, \ldots, P_n)$: vector of actor priorities.
- crisis_level$(c) \in \mathbb{R}$: severity of crisis (non-crisis if ≤ -1).
- scale$(s) = (\text{MAD_score}(\mathbf{D}))^{-0.25}$: Mean Absolute Deviation of avg_pump to identify high demand actors.
- $r \in [1, \infty)$
- $r_c \in [0, \infty)$
- $p_e \geq 0$
- $\mathbb{I}_{\{P_i>0\}}$: indicator function (1 if $P_i > 0$, else 0)

Quota per actor q_i is defined as:

$$q_i = \begin{cases} D_i \cdot r, & \text{if crisis\_level} \leq -1 \\ \dfrac{A_i}{\sum_{j=1}^{n} A_j} \cdot W, & \text{otherwise} \end{cases}$$

where allocation weights A during crisis is

$$A = D_i \cdot \exp\left(P_i \cdot c \cdot p_e \cdot s\right) \cdot \mathbb{I}_{\{P_i>0\}}$$

and the total available water W during crisis is:

$$W = \frac{r_c}{\max(1, c)} \cdot \sum_{k=1}^{n} D_k \cdot \mathbb{I}_{\{P_k > 0\}}$$

3. Incentives Policy Design

This incentives policy promotes **equitable and sustainable water use** by influencing actor behavior through **penalties for overuse** and **rewards for conservation**. It is designed to balance three core objectives:

- **Fairness:** Incentives are scaled relative to each actor's quota and demand, ensuring proportionate responses.
- **Equity:** The policy accounts for actor priority, offering more leniency to high-priority users while maintaining system-wide consistency.
- **Ecosustainability:** Incentive strength increases with crisis severity, encouraging conservation and system resilience under stress.

3.0.1. Policy Mechanism

Core Mechanism

The policy calculates an *incentive rate* based on the difference between actual water use and the actor's quota (the *overuse*). This rate is scaled by:

- **policy_rate (r_p):** Governs the overall intensity of incentives, controlling how strongly behavior is encouraged or discouraged.
- **equity_spread (eqs):** Adjusts the baseline leniency by influencing how much the actor's priority affects the scaling of incentives. A higher equity spread means more leniency towards higher-priority actors.
- **crisis_sensitivity (cs):** Dictates how sharply the incentive system reacts to crisis severity. When crisis levels increase, incentives become more severe, compelling actors to reduce consumption.

Subsidy and Fine Asymmetry

The incentive function is **asymmetric** between penalties and subsidies:

- The **subsidy_fine_ratio (γ)** controls the relative steepness of rewards (subsidies) versus penalties (fines). This allows the policy to fine-tune whether conservation is rewarded more gently or harshly compared to overuse penalties.
- Both penalties and subsidies are bounded by maximum caps - **max_fine_weight (F_{max})** and **max_subvention_weight (S_{max})**. This prevents excessively harsh fines or unreasonably large subsidies.

Crisis Memory and Dynamic Prioritization

An additional important feature is the policy's ability to **incorporate historical crisis information** to modulate responses dynamically:

- The policy tracks whether a *crisis is ongoing*, based on both current water availability and consumption patterns.
- A parameter α balances how much weight to give to *past crisis conditions* versus the *current crisis status*.

- This memory effect allows the system to **anticipate future water shortages** and **adjust crisis severity accordingly**. As a result, if the crisis persists over multiple iterations, the crisis level can escalate, increasing the strictness of incentives.

3.0.2. Mathematical Formulation

Let:

- $\mathbf{D} = (D_1, D_2, \ldots, D_n)$: vector of average water pumped per actor.
- $\mathbf{P} = (P_1, P_2, \ldots, P_n)$: vector of actor priorities.
- $\mathbf{C} \in \mathbb{R}$: vector of severity of crisis (non-crisis if ≤ -1).
- $\mathbf{A} = (A_1, A_2, \ldots, A_n)$: vector of water pumped by each actor.
- $\mathbf{W} = (W_1, W_2, \ldots, W_t)$: vector of water flows.
- $\mathbf{Q} = (Q_1, Q_2, \ldots, Q_n)$: vector of quotas assigned to each actor.
- $\mathbf{I} = (I_1, I_2, \ldots, I_n)$: The average income of the actor.
- $r_p \in (0, \infty)$: The base rate of the incentive.
- $eqs \in (0, \infty)$: The equity spread factor.
- $cs \in (0, \infty)$: The crisis sensitivity factor.
- $\alpha in[0, 1]$: The weight given to the current crisis versus the past crisis
- $\gamma \in (0, \infty)$: The ratio of subsidy to fine.
- $F_{max} \in (0, \infty)$: The maximum weight for fines.
- $S_{max} \in (0, \infty)$: The maximum weight for subsidies.
- n: Number of actors in the system.

We define the effective $crisis\_level$ as a weighted average::

$$crisis\_level = \alpha * next\_crisis\_level + (1 - \alpha) * C[-1]$$

where $C[-1]$ is the crisis level from $EcologyManager$
and $\alpha \in [0, 1]$ controls the weight given to the current crisis versus the past crisis.
and $next\_crisis\_level$ is equivalent to $EcologyManager.compute\_crisis(water\_flow, avg\_pump)$

Modify $crisis\_level$ to account for the continuing crisis:

$$crisis\_level = \begin{cases} crisis\_level + C[-2] & \text{if } crisis\_level \geq 1 \quad or \quad C[-2] \geq 1 \\ crisis\_level & otherwise \end{cases}$$

We define:

- $c = crisis\_level + 1$
- $policy\_regularization(\tau) = eqs \times \exp(-cs \times c)$.
- $priority\_adjustment(p_i) = \exp(2 - P_i)$

$policy\_regularization(\tau)$ controls balance between fairness and equity, and how that balance shifts as crisis severity increases. Penalty rates shift from relative overuse to absolute overuse as crisis level increases.
Let the overuse rate be:

$$u_i = W_i - Q_i$$

The raw incentive rate before asymmetry and capping is:

$$r_p^i = r_p \times \frac{u_i}{Q_i + \tau + p_i}$$

The incentive rate is asymmetric for subsidies (underuse) and fines (overuse):

$$r_p^i = \begin{cases} r_p^i \times \gamma, & \text{if } r_p^i < 0 \quad \text{(subsidy)} \\ r_p^i, & \text{if } r_p^i \geq 0 \quad \text{(fine)} \end{cases}$$

The incentive rate is bounded by caps to ensure fairness and predictability:

$$r_p^i = \min\left(\max(r_p^i, S_{max}), F_{max}\right)$$

The actual monetary incentive applied to actor R_i is:

$$R_i = I_i \times r_p^i$$

3.1. Policy Optimization

Policy optimization is treated as a multi-objective optimization problem, focused on identifying optimal trade-offs between ecological sustainability, economic performance, and actor satisfaction.

This process involves adjusting both quota parameters and incentive parameters to discover the best-performing policies within a well-defined search space. The optimization is implemented using a Pareto optimization approach within an evolutionary algorithm framework. The search process is powered by the off-the-shelf Optuna library, which efficiently explores the parameter space and constructs the Pareto front — the set of non-dominated solutions representing the best trade-offs among objectives.

3.1.1. Implementation via PolicyRegulator Class

The policy optimization logic is encapsulated in the $PolicyRegulator$ class, which orchestrates the following sequence:

- **Define Function Arguments**: Configure the range and type of policy parameters (quota and incentive-related) to explore during optimization.
- **Run Fast Simulation**: Use lightweight simulations to quickly evaluate large numbers of candidate policies under varied conditions.
- **Objective Functions**: Simultaneously optimize ecological, economic, and satisfaction outcomes.

  $$\min\ I_{ecol}, \quad \max\ I_{econ}, \quad 1 - S_{priority} \leq 0$$

  where: I_{ecol} is the ecological impact, I_{econ} is the economic impact, and $S_{priority}$ is the satisfaction of actors.
- **Find Pareto Front**: Identify the set of policies that offer the best possible trade-offs between the three objectives.
- **Select Best Policy**: Use gradient-based gain functions to rank policies within the Pareto front.

3.1.2. Optimization Results

We applied the Optuna-enabled NSGA-II evolutionary algorithm for multi-objective optimization, executing 60 trials with an initial population size of 15 while keeping all other parameters at their default settings. Convergence was observed around the 35th trial, after which improvements in the objective space plateaued.

Notably, the final run of the optimization yielded a Pareto front consisting of a single non-dominated solution. This outcome is atypical, as previous runs generally produced multiple non-dominated solutions. Despite this, the resulting solution was found to be robust and well-aligned with the objectives, clearly illustrating the trade-offs inherent in the problem formulation. The emergence of a singular dominant solution suggests that, under the given parameterization, a strong compromise exists between competing objectives, and further diversification may require broader hyperparameter exploration or relaxed constraint settings.

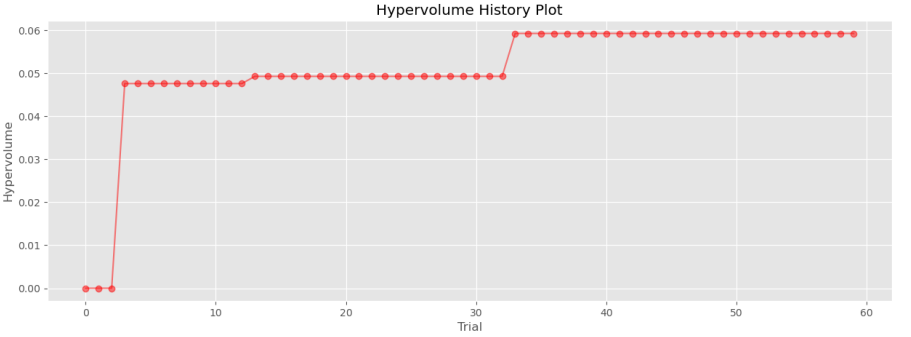


Figure 12. *Hypervolume history over optimization trials. The plot illustrates the convergence behavior of the NSGA-II algorithm, with hypervolume improving steadily and stabilizing around trial 35. This indicates a progressive exploration of the objective space and eventual convergence toward a robust Pareto-optimal solution..*

3.2. Observations

1. **Satisfaction–Ecology Tradeoff**
   A strong negative correlation ($r = -1.0$) was observed between actor satisfaction and ecological impact, indicating a near-perfect inverse relationship. Policies that maximize user satisfaction, particularly for high-priority actors, consistently result in elevated ecological stress, highlighting the fundamental tension between user demands and environmental preservation.

   - Increasing water availability or allocation flexibility during scarcity events consistently improved satisfaction across actor types but led to higher ecological degradation.
   - Emphasizing high-priority actor satisfaction skewed allocations, improving individual-level outcomes but intensifying overall environmental stress.
   - Policies that constrained resource distribution in order to preserve ecological thresholds yielded the lowest satisfaction levels but were the most effective in limiting ecological impact.
   - **Figure 14** shows the hyperparameter importance plot. Priority weighting and crisis water thresholds emerge as the most influential parameters for this tradeoff, underscoring their central role in balancing satisfaction against ecological outcomes.

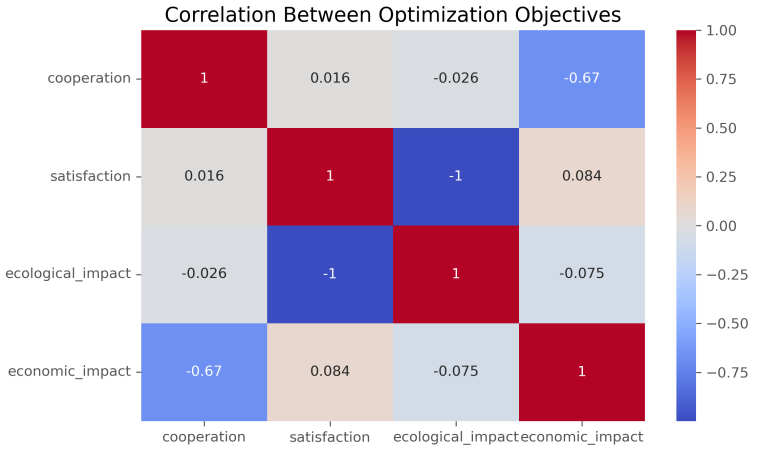


Figure 13. *Correlation matrix of optimization objectives. The plot highlights strong inverse relationships between key objectives, particularly between actor satisfaction and ecological impact ($r = -1.0$), and between cooperation and economic outcome ($r = -0.7$). These correlations underscore the inherent trade-offs in the multi-objective optimization landscape..*

2. **Cooperation–Economy Link via Subvention**
   A moderately strong negative correlation ($r = -0.7$) was observed between economic outcomes and cooperation, particularly under subvention-heavy policy regimes. While subventions enhance individual actor incentives and promote cooperative behavior, they may also reduce systemic economic efficiency by encouraging resource redistribution over true productivity gains.

   - Subventions increased cooperation by improving short-term actor returns, but they did not significantly raise total system economic output.
   - Higher levels of cooperation under subsidy regimes often led to economic fragility and dependency, reducing long-term adaptability.
   - Economic benefits were concentrated among actors best positioned to leverage subventions, raising concerns about equity and structural bias.
   - **Figure 15** illustrates that subvention-related hyperparameters are dominant drivers of cooperation. However, their marginal contribution to net economic outcomes highlights a potential misalignment in incentive structures.

3. **Diffusion of Cooperation Under Dissatisfaction**: When satisfaction is achieved, cooperation levels tend to stabilize around a compact distribution—typically centered near 0.5—indicating a balanced and cohesive system. In contrast, low satisfaction scenarios produce more diffuse and variable cooperation, reflecting fragmentation that can destabilize system dynamics.

   - Medium-priority, high-demand actors tend to be less cooperative and more competitive during balanced cooperation-competition regimes. This arises because they occupy an intermediate position with limited buffer capacity and lower prioritization during scarcity.
   - This dynamic is especially pronounced in the most ecologically sound solutions, where increased competition among high-demand actors plays a key role in system resilience.

The *policy_optimization* notebook illustrates the difference in cooperation patterns between satisfied and unsatisfied regimes. The and *single_scenario_∗* demonstrate different single scenario outcomes for cooperation under satisfaction and dissatisfaction simulations.

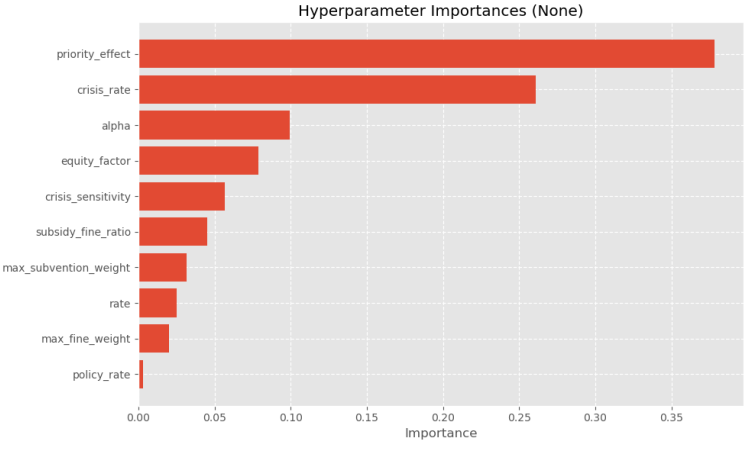


Figure 14. *Hyperparameter importance for the Ecological Objectives. Priority weights and water availability thresholds are key drivers of the ecological health..*

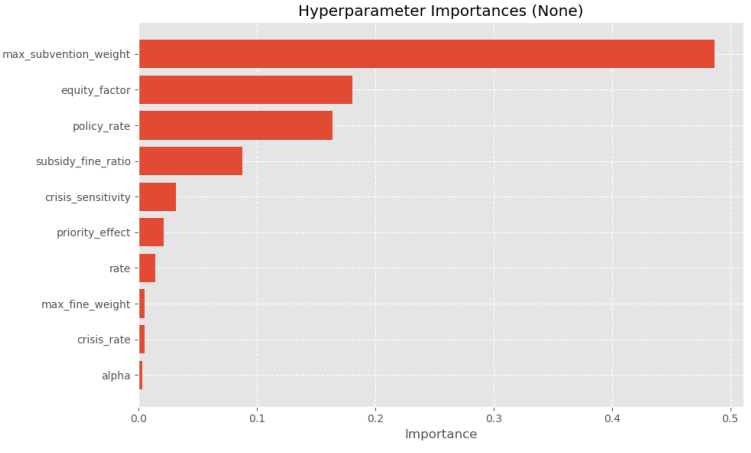


Figure 15. *Hyperparameter importance for cooperation and economic outcomes. Subvention levels and cooperation thresholds strongly influence cooperation, while their economic effects are more diffuse..*

3.2.1. Post-Optimization Evaluation and Manual Selection

Following the optimization, the final run of the NSGA-II algorithm yielded a single non-dominated solution on the Pareto front. Due to the lack of multiple trade-off candidates, we manually selected and evaluated additional promising trials—specifically Trial 33, Trial 46, and a configuration derived from the best parameters of the `pol_regulator` policy.

These parameter sets were tested using the `multi_scenario_analysis` notebook to assess their performance across multiple objectives. To further refine outcomes, we manually adjusted some parameters to improve simulation results, focusing on satisfaction, economic impact, and ecological impact.

Table 11 presents a comparison of the evaluated parameter sets. Trial 33 demonstrated the most favorable balance, achieving absolute satisfaction while maintaining acceptable ecological and economic trade-offs. Based on this analysis, Trial 33 was selected as the final solution for submission.

Parameter Set	Satisfaction	Violations	Economic Impact	Ecological Impact
Trial 33 (*best*)	**1.000**	0	**0.885**	**0.935**
Trial 46	**1.000**	0	0.840	0.941

Table 11. *Comparison of parameter sets based on key optimization objectives*

4. Conclusion and Future Work

This work demonstrates a robust framework combining uncertainty-aware streamflow forecasting with multi-objective policy optimization, showing strong performance across spatiotemporal domains. Opportunities remain to improve efficiency and uncertainty handling. Future efforts will focus on optimizing model complexity for faster and more memory-efficient inference, enhancing uncertainty calibration to provide trustworthy prediction intervals across varied hydrological conditions, and developing transfer learning strategies to leverage knowledge from data-rich basins in underrepresented regions. Integrating real-time adaptation through online learning will further improve responsiveness to evolving data streams and environmental changes.

The policy optimization revealed fundamental trade-offs between actor satisfaction, ecological impact, cooperation, and economic outcomes. For example, a near-perfect negative correlation between satisfaction and ecological health underscores the challenge of balancing human needs with environmental sustainability. Cooperation improves with higher subvention levels but exhibits a negative correlation with economic performance, suggesting diminishing returns and potential long-term fragility. Behavioral analysis indicates that cooperation tends to concentrate around moderate levels when satisfaction is high, while dissatisfaction leads to fragmented patterns, highlighting system instability. Detailed single-scenario analyses validate these insights, reinforcing the importance of nuanced policy designs that balance competing objectives and leverage evolutionary optimization to identify effective solutions and meaningful system behaviors.

Competing Interests. None

Ethical Standards. The research meets all ethical guidelines, including adherence to the legal requirements of the study country.

Author Contributions. Conceptualization: D.A.; Methodology: D.A; Data curation: D.A. Data visualisation: D.A. Writing original draft: D.A; All authors approved the final submitted draft.

References

Beven, KJ (2012) *Rainfall-Runoff Modelling: The Primer. Chichester: Wiley-Blackwell.*

Singh VP (1995) Computer Models of Watershed Hydrology, *Water Resources Publications*

Mosavi A, Ozturk P, and Chau KW (2018) Flood prediction using machine learning models: A review, *Water 10*(11), 1536.

Kratzert F, Klotz D, Brenner C, Schulz K, and Herrnegger M (2019) Neural Hydrology: Interpreting LSTMs in Hydrology, *Hydrology and Earth System Sciences 23* 375—-397.

Prokhorenkova L, Gusev G, Vorobev A, Dorogush A, and Gulin A (2018) CatBoost: unbiased boosting with categorical features, *Advances in Neural Information Processing Systems 31*.

Spyromitros-Xioufis E, Tsoumakas G, Groves W, and Vlahavas I (2016) Multi-target regression via input space expansion: Treating targets as inputs, *European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases* 105—120 Springer

Romano Y, Patterson E, and Candes EJ (2019) Conformalized quantile regression, *Advances in Neural Information Processing Systems 32*.

Angelopoulos A and Bates S (2022) A gentle introduction to conformal prediction and distribution-free uncertainty quantification, *arXiv e-prints Art. no. arXiv:2107.07511v6, 2022. doi:10.48550/arXiv.2107.07511*

Sousa M, Tomé AM, and Moreira J (2022) Improved conformalized quantile regression, *arXiv e-prints Art. no. arXiv:2207.02808, 2022. doi:10.48550/arXiv.2207.02808*

Hjort A, Williams JP, and Pensar J (2024) Clustered Conformal Prediction for the Housing Market, *Proceedings of the Thirteenth Symposium on Conformal and Probabilistic Prediction with Applications*, *Proceedings of Machine Learning Research PMLR* 230:366-386, Available from https://proceedings.mlr.press/v230/hjort24a.html.

Ding T, Angelopoulos AN, Bates S, Jordan M, and Tibshirani R (2023) Class-Conditional Conformal Prediction with Many Classes, *Thirty-seventh Conference on Neural Information Processing Systems*, Available from https://openreview.net/forum?id=mYz6ApeU4J.

Marx C, Zhao S, Neiswanger W, and Ermon S (2022). Modular Conformal Calibration, *Proceedings of the 39th International Conference on Machine Learning (ICML)* Available at: https://arxiv.org/abs/2206.11468

Deutschmann N, Rigotti M, and Rodríguez Martínez M (2023) Adaptive Conformal Regression with Jackknife+ Rescaled Scores, *arXiv preprint arXiv:2305.19901*, Available at: https://arxiv.org/abs/2305.19901

Arsenault R, Martel J-L, Brunet F, Brissette F, and Mai J (2023) Continuous streamflow Prediction in Ungauged Basins: Long Short-Term Memory in Neural Networks Clearly Outperform Traditional Hydrological Models, *Hydrology and Earth System Sciences*, *27*(1), 139–157, *https://hess.copernicus.org/articles/27/139/2023/hess-27-139-2023.pdf*

Kuhn M and Johnson K (2019). *Feature Engineering and Selection: A Practical Approach for Predictive Models*. *Chapman and Hall/CRC*.

Hyndman R and Athanasopolous G (2018). *Forecasting: Principles and Practice*. *OTexts*.

Peng H (2005)Feature Selection Based on Mutual Information: Criteria of Max-Dependency, Max-Relevance, and Min-Redundancy. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *27*(8):1226-1238, *doi:10.1109/TPAMI.2005.159*

Shortridge JE, Guikema SD, Zaitchik BF. (2016) Machine Learning Methods for Empirical Streamflow Simulation: A comparison of Model Accuracy. *Hydrological Processes*, 30(3): 366-376. DOI: 10.1002/hyp.10573

Kumar V, Kedam N, Sharma KV, Mehta DJ, and Caloiero T (2023) Advanced Machine Learning Techniques to Improve Hydrological Prediction: A Comparative Analysis of Streamflow Prediction Models, *Water*, *15*(14):2572, *https://doi.org/10.3390/w15142572*

Stephens M (2000) Dealing with Label Switching in Mixture Models, *Journal of the Royal Statistical Society: Series B*, *62*(4): 795-809.

Strehl A and Ghosh J (2003) Cluster Ensembles - A Knowledge Reuse Framework for Combining Multiple Partitions, *Journal of Machine Learning Research*, *3*: 583-617.

Kuhn HW (1955) The Hungarian Method for the Assignment Problem, *Naval Research Logistics Quarterly*, *2*(1-2): 83-97.

Wang X (2025) Sustainable Water Management: Balancing Social, Economic, and Environmental Needs. *Journal of Lifestyle and SDG Reviews*, *5*(5) e06606 *https://doi.org/10.47172/2965-730X.SDGsReview.v5.n05.pe06606*

tian2019] **Tian J, Guo S, Liu D, Pan Z, and Hong X** (2019) A Fair Approach for Multi-Objective Water Resources Allocation, *Water Resources Management*, *33*, 3633-3653 *https://doi.org/10.1007/s11269-019-02323-5*

Tang X, He Y, Qi P, Chang Z, Jiang M and Dai Z (2021) A New Multi-Objective Optimization Model of Water Resources Considering Fairness and Water Shortage Risk, *Water*, *13*, 2648, *https://doi.org/10.3390/w13192648*

Fu Q, Li L, Li M, Li T, Liu D, and Cui S (2018) A Simulation-Based Linear Fractional Programming Model for Adaptable Water Allocation Planning in the Main Stream of the Songhua River Basin, China, *Water*, *10*(5), 627, *https://doi.org/10.3390/w10050627*