Experiment: Koksilah River

Caden Hewlett

2025-07-14

Introduction

This document is a preview and doesn't include everything that would normally be in a performance report.

In this experiment, I take the 'new' GARMA DKR object described in the previous reports and fit it to the Koksilah River watershed.

As usual, we allow $\mathbf{x}_t$ to be the rainfall and $\mathbf{y}_t$ to be the stream flow. We further allow $\mathbf{z}_t$ to be the PET rather than streamflow to avoid fitting negative values (which are impermissible under Gamma likelihood.)

For simplicity, I fit a single model with $(k, p, q) = (2, 2, 2)$.

We load the data below:

```
data <- DKR::koksilah_PET
train <- data[data$hydr_year < 30,]
configs <- list(
  k=2,p=2,q=2
)
xt <- train$rain
zt <- train$PET
yt <- train$gauge
```

The model is fitted as follows

```
source(here::here("Gamma AR", "GARMA_DKR", "garma_dkr_main.R"))
# get data
set.seed(1928)
garma_fit <- fit_DKR(
  xt = xt,
  zt = zt,
  yt = yt,
  configs = configs,
  report = TRUE
)
saveRDS(garma_fit, here::here("Gamma AR", "koksilah_fit.RData"))
```

Which we have fitted ahead of time and load below:

```
mod <- readRDS(here::here("Gamma AR", "koksilah_fit.RData"))
```

1. In-Sample (Training) Evaluation Metrics

The table below presents performance metrics for each model fitted during training. These include log-likelihood, normalized Root Mean-Squared Error (NRMSE), RMSE, R^2, Nash-Sutcliffe Efficiency (NSE), and Kling-Gupta Efficiency (KGE).

Table 1: Summary of In-Sample Training Metrics

loglik	RMSE	NRMSE	R2	NSE	KGE
-4528.54	0.91	0.25	0.98	0.98	0.96

2. Estimated Model Parameters

We have the estimated kernel parameters:

Table 2: Estimated Kernel Parameters

Kernel	$\hat{\beta}_0$	$\hat{\beta}_1$	$\hat{\delta}$	$\hat{\sigma}$
1	1.19	-0.32	1.52	13.39
2	0.34	-0.10	2.29	1.79

And the rest of the parameters:

Table 3: Estimated ARMA and Shape Parameters

$\hat{\phi}_1$	$\hat{\phi}_2$	$\hat{\theta}_1$	$\hat{\theta}_2$	$\hat{\alpha}$
0.3495	0.5606	0.75	0.0974	8.5608

3. Composition of μ_t

Here, we examine exactly how the Kernel Regression component $\tilde{y}_t$ and the ARMA error component τ_t contribute to a fitted μ_t in a one-step-ahead forecast style.