State Space Model Analysis Using KFAS

Introduction

State space models allow representing a time series through a latent state variable that evolves over time and an observation equation that links the observed data to the state variable with the addition of noise.

In this document, the **KFAS** package is used to estimate state space models for the annual flow series of the Nile River (Nile).

Typically, some of the parameters in our state space models are unknown and must be estimated. If the assumption of Gaussian system disturbances is plausible, the Kalman filter makes it possible to compute the likelihood for any set of parameter values to be estimated. However, maximizing the (log-)Gaussian likelihood even when the data are not normally distributed still yields estimates whose asymptotic behavior is known (under regularity conditions).

State Space Form

```
library(KFAS)
modello1 <- SSModel(Nile~0+SSMtrend(1, NA), H = NA)
fit1 <- fitSSM(modello1, rep(log(var(diff(Nile))), 2))
fit1$optim.out$convergence
```

We estimate a random walk plus noise model, where:

- **Observation equation:**

$$y_t = \alpha_t + \varepsilon_t, \varepsilon_t \sim N(0, H)$$

- State equation (local level, order 1):

$$\alpha_{t+1} = \alpha_t + \eta_t, \eta_t \sim N(0, Q)$$

The model therefore has two variances to be estimated:

`SSMtrend(1, NA)` specifies a local level (random walk) with unknown variance (`NA`).

`H = NA` indicates that the variance of the observation noise is also unknown.

Maximum likelihood estimates

First method: logLik() function

```
# I create a function that estimates the variances of the LLT + WN model:

fit_RW_plus_WN <- function(model, pars.init = rep(log(var(diff(model$y))), 2))

objective <- function(pars) {
```

```
model$Q[1, 1, 1] <<- exp(pars[1])
model$H[1, 1, 1] <<- exp(pars[2])
-logLik(model) # I change the sign because optim() performs minimization
}

# We minimize the objective function (using the BFGS algorithm)
#and request the Hessian at the maximum point

optout <- optim(pars.init, objective, method = "BFGS", hessian = TRUE)

estim <- exp(optout$par)
#maximum likelihood estimation of the variances;

mI<- solve(optout$hessian)
#inverse of the estimated Fisher information;

mV <- diag(estim) %*% mI %*% diag(estim)
# delta method in action;

sterr <- sqrt(diag(mV))
#vector of standard errors;


# assign names to the parameters;
names(estim) <- names(sterr) <-
colnames(mV) <- rownames(mV) <- c("var(eta)", "var(eps)")

list(parameters = estim, stderr = sterr, cov = mV, logLik = -optout$value,opti
}


fit1 <- fit_RW_plus_WN(modello1)

# Estimation of the variances and their standard errors
cbind(param = fit1$parameters, stderr = fit1$stderr)

##                param   stderr
##  var(eta)   1469.163 1280.358
## var(eps)  15098.651 3145.560

# Value of the log-likelihood at its maximum
fit1$logLik
#[1] -632.5456


# Number of calls to the objective() function made by optim()
fit1$optim.out$counts
## function gradient
## 36 10

# Convergence code from optim(): if it is zero, the optimization was successfu
fit1$optim.out$convergence
#[1] 0
```

Second method: estimation via the `fitSSM()` function

```
updt <- function(pars, model) {
model$Q[1, 1, 1] <- exp(pars[1])
model$H[1, 1, 1] <- exp(pars[2])
model
}
fit3 <- fitSSM(modello1, rep(log(var(diff(Nile))), 2), updt)

# We check the convergence of the optimization;
fit3$optim.out$convergence
# [1] 0


# We look at the estimates in the matrices;
fit3$model$Q
## , , 1
##
## [,1]
## [1,] 1466.32
fit3$model$H
## , , 1
##
## [,1]
## [1,] 15098.18
```

The estimates are indeed almost identical to those obtained with the first method.
If we want to compute the standard errors of the estimates, we can use the formal argument `hessian = TRUE`, which will be passed to `optim().`

```
fit3 <- fitSSM(modello1, rep(log(var(diff(Nile))), 2), updt, hessian = TRUE)
V <- solve(fit3$optim.out$hessian)
GVG <- diag(exp(fit3$optim.out$par)) %*% V %*% diag(exp(fit3$optim.out$par))
stderr <- sqrt(diag(GVG))
cbind(param = exp(fit3$optim.out$par), stderr=stderr)

##         param    stderr
## [1,]  1466.32  1279.011
## [2,] 15098.18  3145.357
```

Filtering and smoothing

Once the unknown parameters of the state space model have been estimated, it is natural to use the model to forecast the time series, to perform inference on its components, and finally to diagnose the presence of shocks or structural changes in the series.

To obtain optimal estimates of the latent components, we use **Kalman smoothing**:

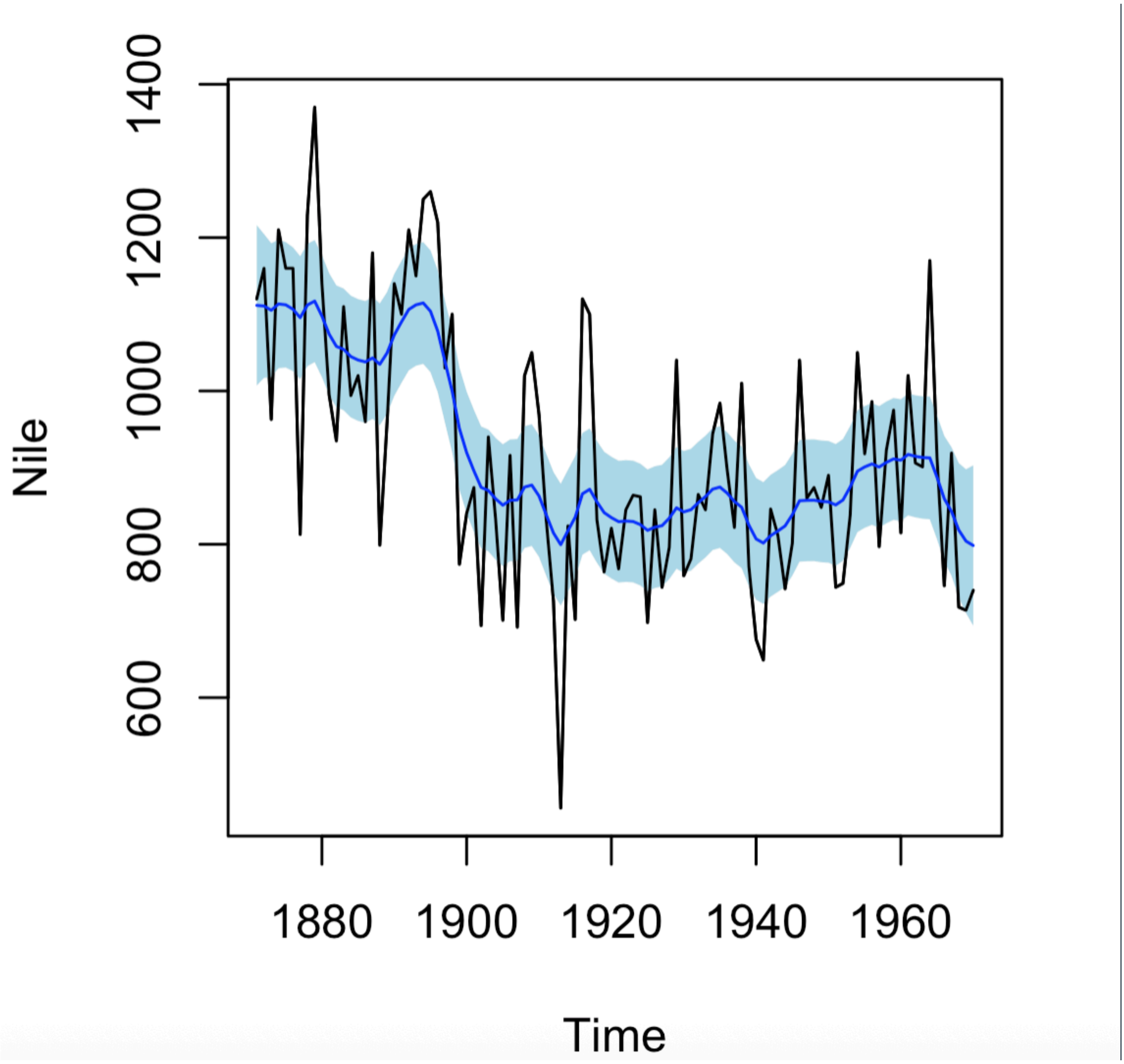
```
# I apply the state and disturbance smoother
smo1 <- KFS(fit1$model, smoothing = c("state", "disturbance"))
```

```
plot(Nile) #Plot of the Nile series

lines(smo1$alphahat[, 1], col = "blue")
# Add the smoothed level
```

We also try adding the 90% confidence bands of the smoother.

```
plot(Nile)
polygon(c(1871:1970, 1970:1871), c(smo1$alphahat[, 1]+qnorm(.95)*sqrt(smo1$V[1
col = "lightblue", border = FALSE)
lines(Nile)
lines(smo1$alphahat[, 1], col = "blue")
```

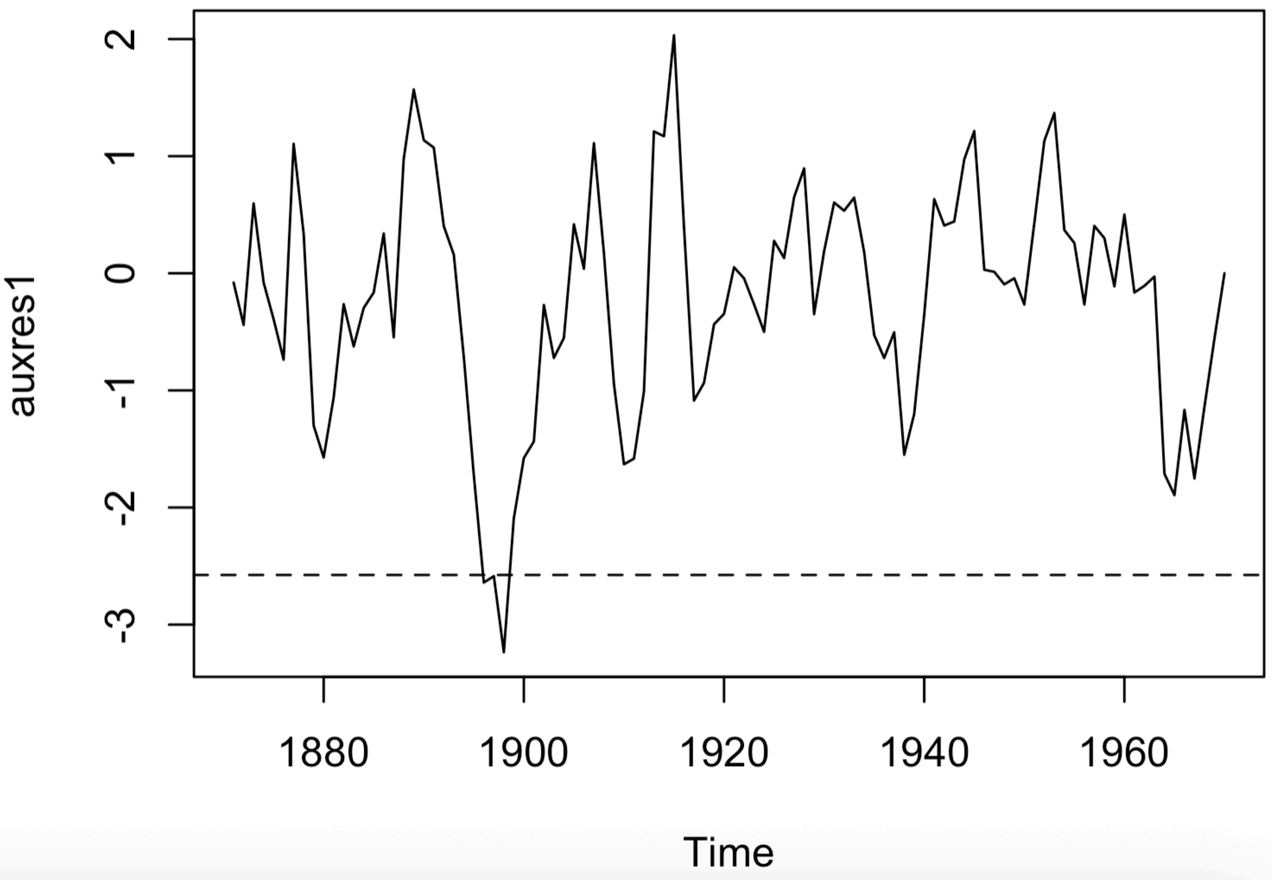


We use the standardized residuals to check for any extreme shocks in the disturbances of the random walk (these cause abrupt permanent level changes).

```
auxres1 <- rstandard(smo1, type = "state")
# Compute auxiliary residuals
```

```
plot(auxres1)
# Plot auxiliary residuals

abline(h = c(qnorm(0.005), qnorm(0.995)), lty = 2) # Bands at 1%
```



Sembra che verso la fine del secolo XIX vi sia un repentino salto verso il basso.

```
time(auxres1)[which.min(auxres1)]
## [1] 1898
```

It appears that towards the end of the 19th century there is a sudden downward jump.
The most extreme value is that of 1898, which appears in the time series in 1899.
Indeed, recall that KFAS uses the form $\mu_{t+1} = \mu_t + \eta_t$ so a shock at time t manifests in μ, and thus in y, at time t+1 .

We can modify the model by introducing a regressor to represent a jump in the level at time t=1899 .

```
# Creation of a step (dummy) variable
step <- Nile
step[] <- 0
window(step, start = 1899) <- 1
# Creation of the state space form and estimation
modello2 <- SSModel(Nile~0+step+SSMtrend(1, NA), H = NA)
fit2 <- fitSSM(modello2, rep(log(var(diff(Nile))), 2))
cat("Convergence code =",fit2$optim.out$convergence)
```

```
## Convergence code = 0
round(exp(fit2$optim.out$par), 3)
## [1] 0.001 16302.228
```

The variance of the random walk disturbance is now zero, so the level remains constant.
It appears that the abrupt level change in 1899 absorbs all movements of the level component.
The remaining variance is due to observation noise.

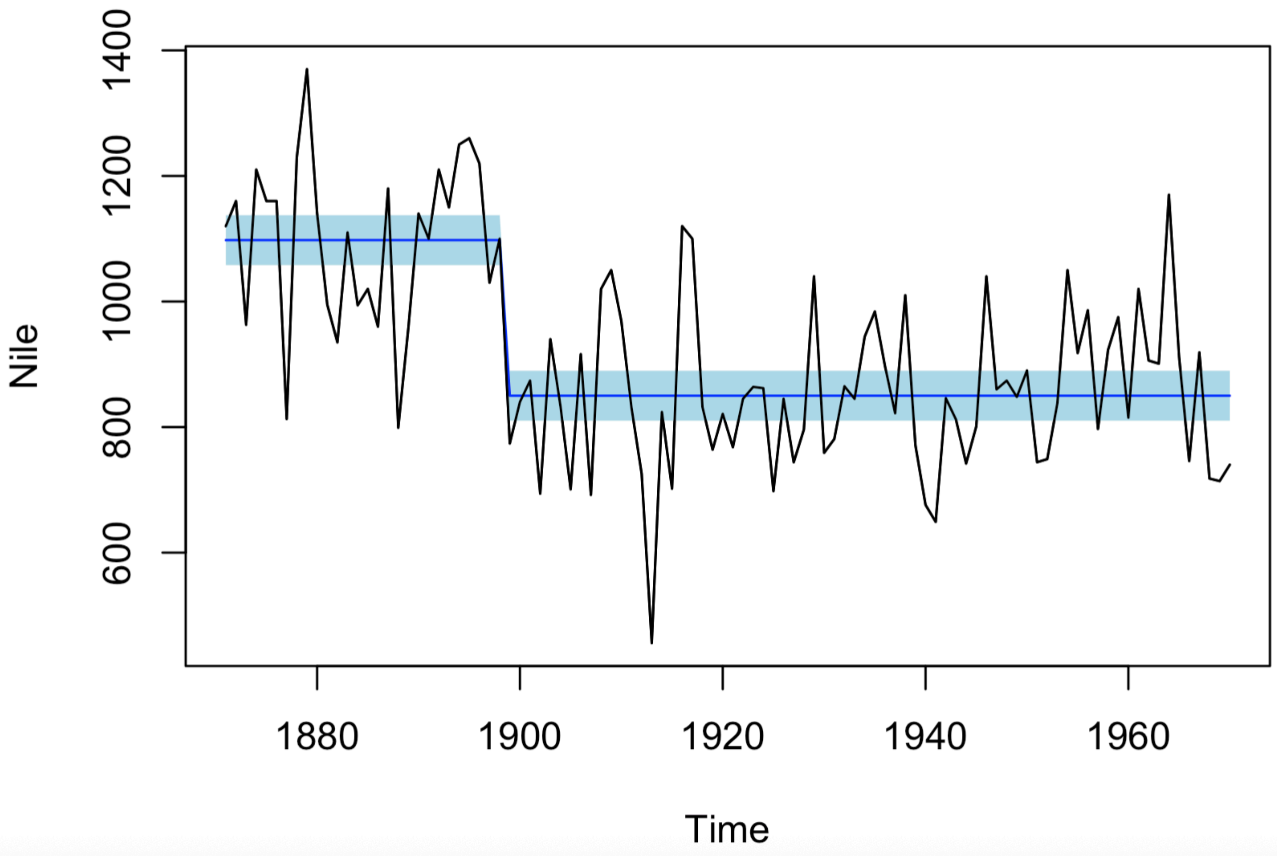
We now test, using a t-statistic, whether the level change in 1899 is statistically zero.

```
smo2 <- KFS(fit2$model, smoothing = c("state", "disturbance", "signal","mean")
ttest <- smo2$alphahat[length(Nile), 1] / sqrt(smo2$V[1, 1, length(Nile)])
pvalue <- pnorm(-abs(ttest))*2
round(rbind(tstat=ttest, pvalue=pvalue), 4)

## step
## tstat -8.7132
## pvalue 0.0000
```

The jump is significant at any usual level.
We now overlay the plot of the Nile series with the sum of the level and coefficient * step.

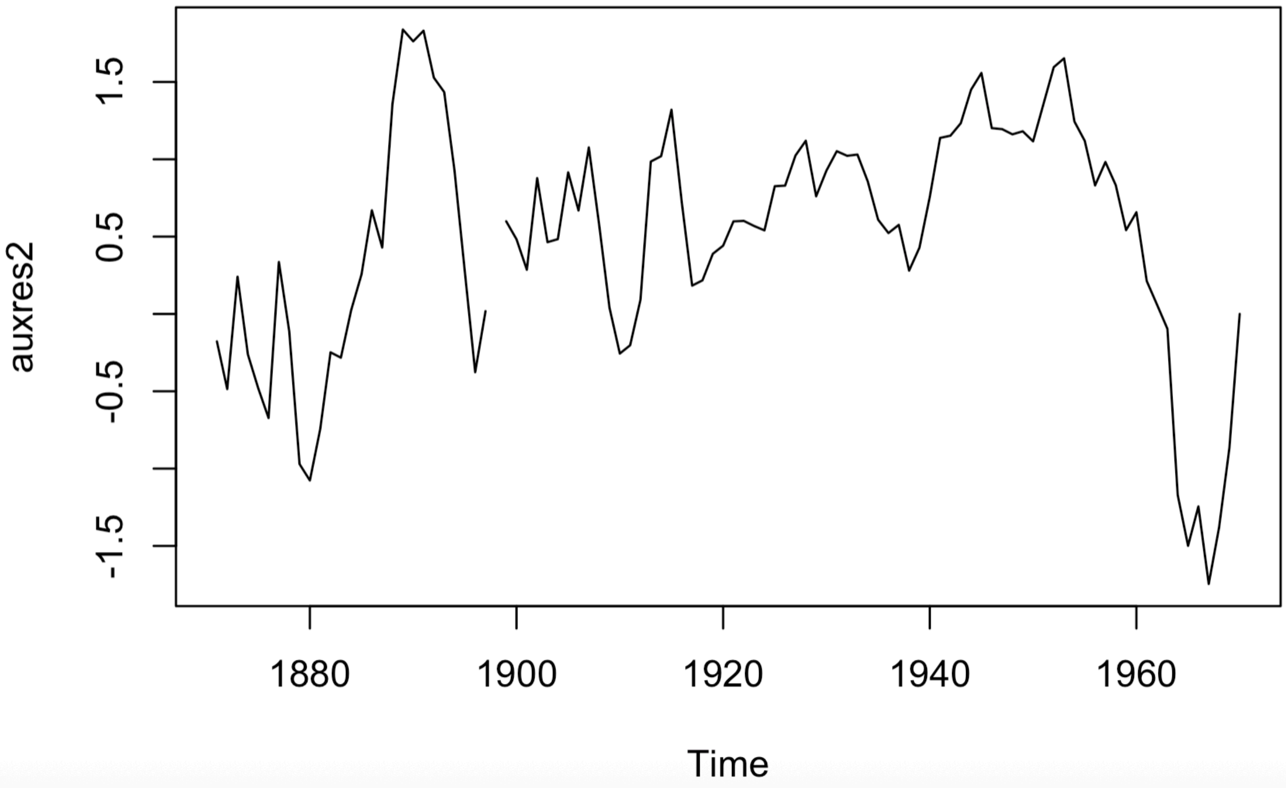
```
# I create the variable coeff_step * step + level
lvl_jump <- smo2$alphahat[, "step"]*step + smo2$alphahat[, "level"]
plot(Nile)
polygon(c(1871:1970, 1970:1871), c(lvl_jump + qnorm(.95)*sqrt(smo2$V[2, 2, ]),
col = "lightblue", border = FALSE)
lines(lvl_jump, col = "blue")
lines(Nile)
```



The plot clearly shows the regime change from 1899 onwards.

Let’s now examine the auxiliary residuals of the level in this model.

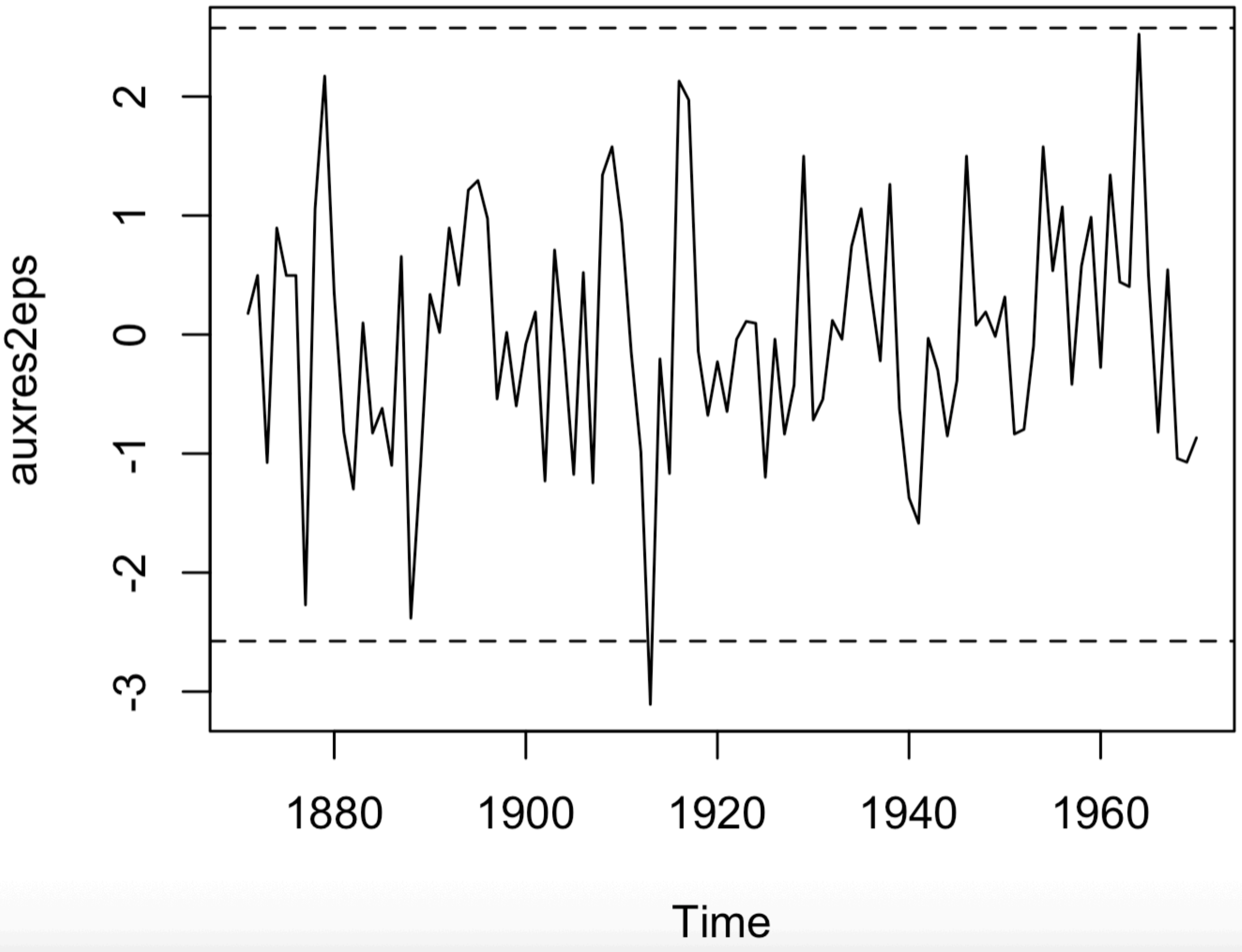
```
auxres2 <- rstandard(smo2, type = "state")
plot(auxres2)
abline(h = qnorm(c(0.005, 0.995)), lty = 2)
```



There are now no disturbances exceeding the threshold values we set (−2.58, 2.58), i.e., values in the 1% tails.
The step-change model captures the level dynamics well.

Using the standardized observation error residuals, we can also identify possible additive outliers.

```
auxres2eps <- rstandard(smo2, type = "pearson")
plot(auxres2eps)
abline(h = qnorm(c(0.005, 0.995)), lty = 2)
```



There is only one value that exceeds the 99% confidence bands.
However, since there are 100 auxiliary residuals, this result is expected (under the null hypothesis that the auxiliary residuals are standard normal, on average one value out of 100 will exceed the 99% confidence bands).

Conclusions

The analysis carried out on the annual flow data of the Nile using state space models made it possible to capture important aspects of the series' dynamics and to improve the understanding of its latent components.
In particular, through estimation with the KFAS package and the application of the Kalman filter and smoother, the following conclusions were drawn:

1. Adequacy of the local level model with observation noise

The initial model, a random walk with additive white noise, proved capable of representing the time series in a general sense, separating a latent level component that evolves over time from an observation noise component.
The variance estimates obtained via maximum likelihood reflect strong variability both in the level dynamics and in the observation noise.

However, the analysis of the standardized state residuals revealed a significant anomaly: a sudden and marked shock at the end of the 19th century (around 1899) that the basic model could not

explain gradually.
This results in an extreme residual beyond the significance thresholds, indicating a structural change.

2. Identification and modeling of the structural change in 1899

To account for this evidence, a dummy variable (step) was introduced to trigger a permanent change in the series level starting in 1899.
This approach allows us to explicitly model a regime change: a sharp jump in the level that is no longer interpreted as a simple temporary shock but as a stable modification of the system.

The inclusion of this regressor resulted in a more parsimonious and interpretable model, where the variance of the random walk disturbance becomes negligible — indicating that the level dynamics are essentially constant after the jump — and the remaining variability is mainly attributable to observation noise.

3. Statistical significance and practical implications

The t-test on the coefficient associated with the jump confirmed the statistical significance of the level change, with a p-value that is virtually zero.
This indicates with high statistical confidence that the 1899 event caused a structural break in the Nile’s regime, presumably linked to long-term natural, climatic, or anthropogenic phenomena.

This result highlights the importance of including components in models that can capture exceptional events and regime changes, which are essential for an accurate representation and forecasting of the series.

4. Diagnostic validation and quality of the improved model

The analysis of standardized auxiliary residuals, both state and observation, showed that in the step-change model:

- No extreme state residuals are observed, suggesting that all significant level variations have been adequately modeled.

- Observation residuals are consistent with the hypothesis of normal white noise, with only a few outliers expected given the sample size.

This final diagnostic confirms that the model with a structural jump is well-calibrated and better suited to represent the dynamics of the time series.