MATH 8090: Spectral Analysis of Time Series II

Whitney Huang, Clemson University

11/9-11/2021

Contents

Spetral ANOVA example

```
x <- c(1, 2, 3, 2, 1)
x <- x - mean(x)
c1 <- cos(2 * pi * (1:5) * (1 / 5)); s1 <- sin(2 * pi * (1:5) * (1 / 5))
c2 <- cos(2 * pi * (1:5) * (2 / 5)); s2 <- sin(2 * pi * (1:5) * (2 / 5))
omega1 <- cbind(c1, s1); omega2 <- cbind(c2, s2)
anova(lm(x ~ omega1 + omega2))
```

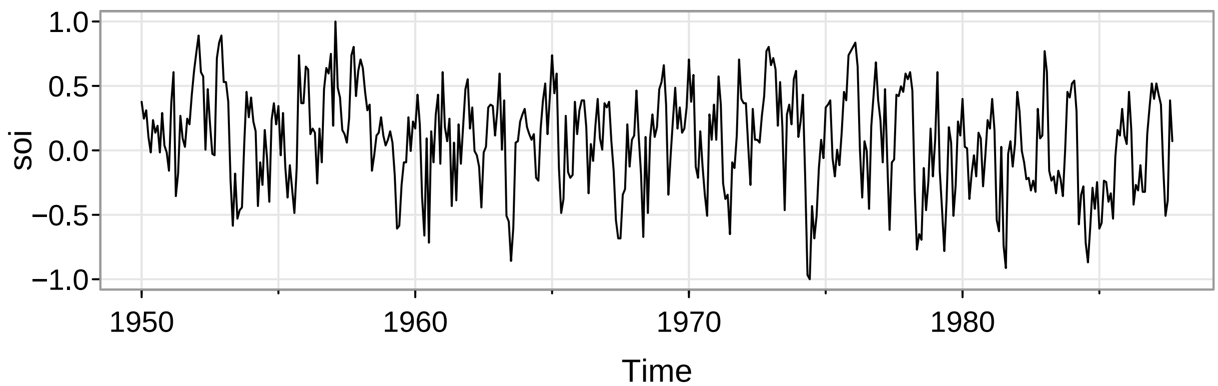
```
## Warning in anova.lm(lm(x ~ omega1 + omega2)): ANOVA F-tests on an essentially
## perfect fit are unreliable
```

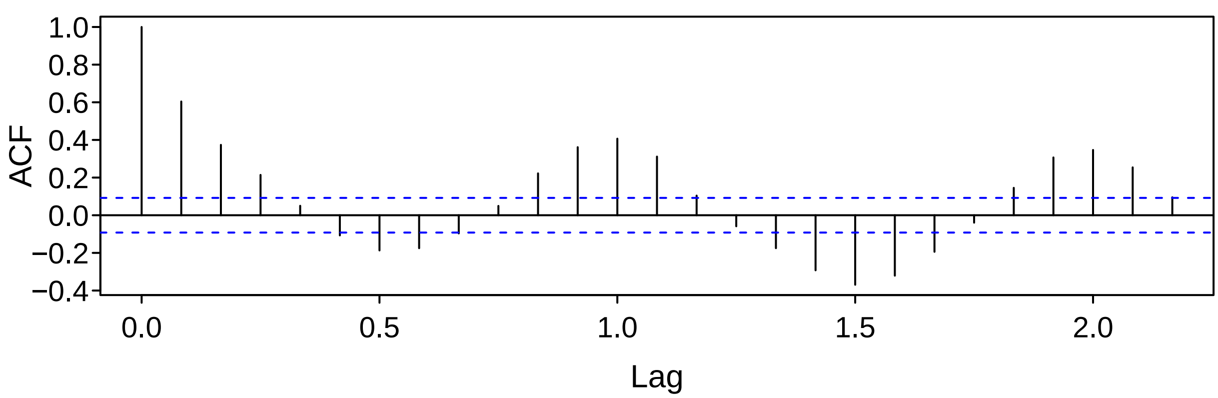
```
## Analysis of Variance Table
##
## Response: x
##           Df  Sum Sq Mean Sq F value Pr(>F)
## omega1     2 2.74164 1.37082     NaN    NaN
## omega2     2 0.05836 0.02918     NaN    NaN
## Residuals  0 0.00000     NaN
```

SOI example

Plot the time series and ACF

```r
library(astsa)
par(mgp = c(2.2, 1, 0), mar = c(3.5, 4, 0.8, 0.6), las = 1, mfrow = c(2, 1))
tsplot(soi)
acf(soi, main = "")
```



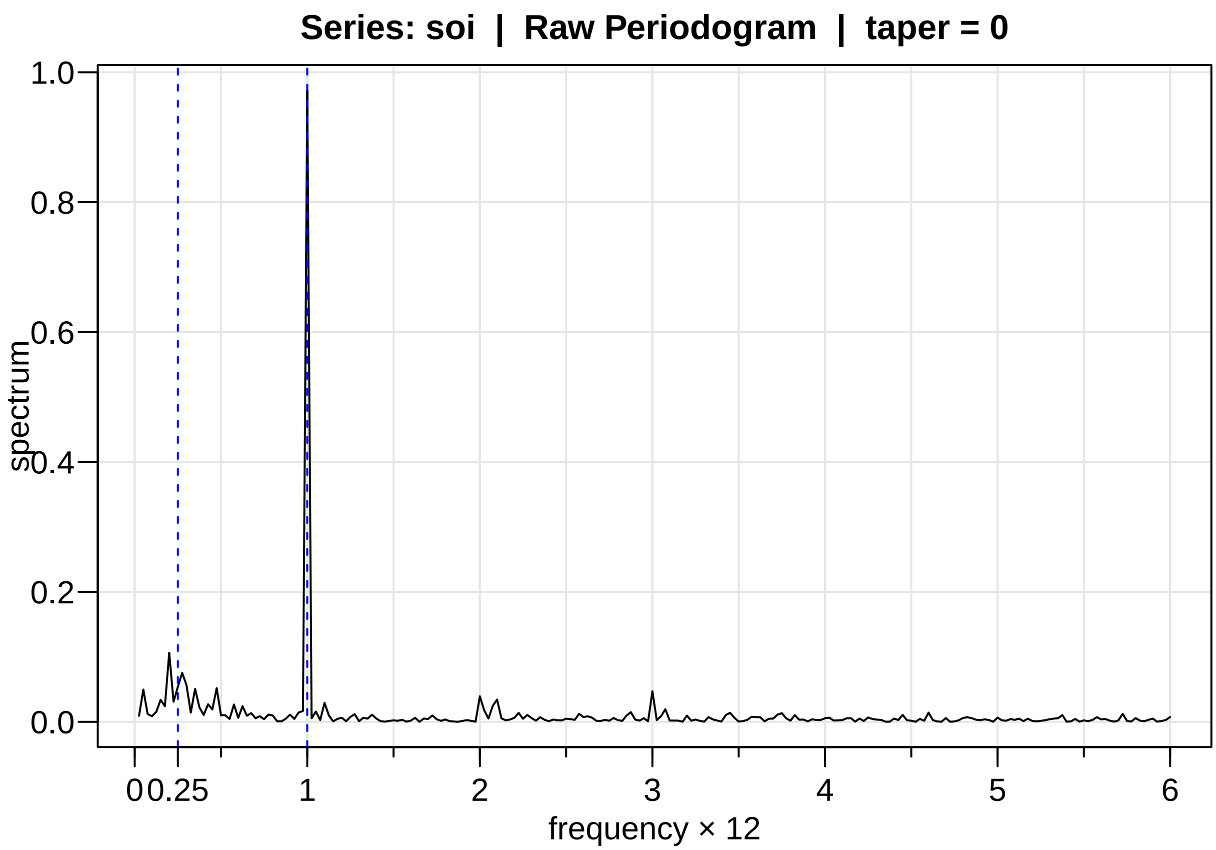


Raw periodogram

An approximate $100(1-\alpha)\%$ confidence interval for $f(\omega)$

$$\frac{2I(\omega_j)}{\chi^2_2(1-\alpha/2)} \le f(\omega) \le \frac{2I(\omega_j)}{\chi^2_2(\alpha/2)}$$

```r
par(mgp = c(2.2, 1, 0), mar = c(3.5, 4, 1.4, 0.6), las = 1)
soi.per <- mvspec(soi)
abline(v = c(1 / 4, 1), lty = 2, col = "blue")
axis(1, at = 0.25)
```




```r
U = qchisq(.025, 2)
L = qchisq(.975, 2)
# 4-year period
soi.per$details[10,]
```

```
## frequency    period  spectrum
##    0.2500    4.0000    0.0537
```

```r
c(2 * soi.per$spec[10] / L, 2 * soi.per$spec[10] / U)
```

```
## [1] 0.0145653 2.1222066
```

```r
# 1-year period
soi.per$details[40,]
```

```
## frequency    period  spectrum
##    1.0000    1.0000    0.9722
```

```r
c(2 * soi.per$spec[40] / L, 2 * soi.per$spec[40] / U)
```

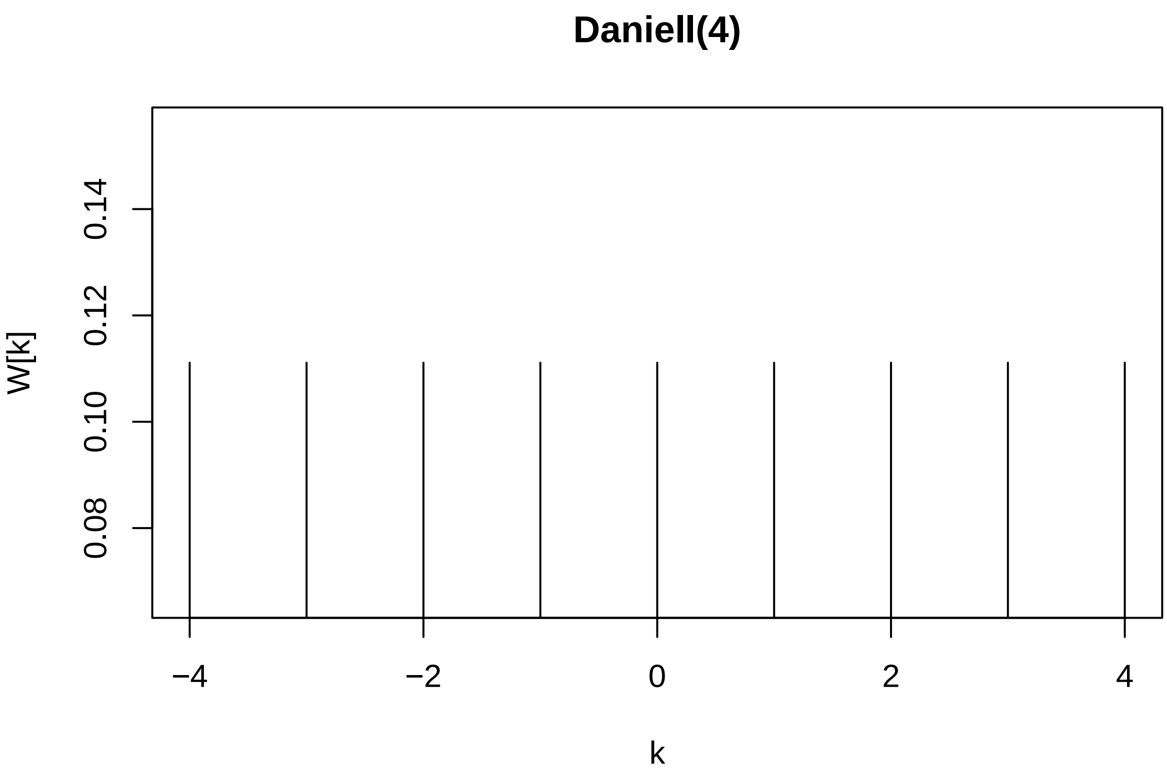
```
## [1]  0.2635573 38.4010800
```

Averaged periodogram

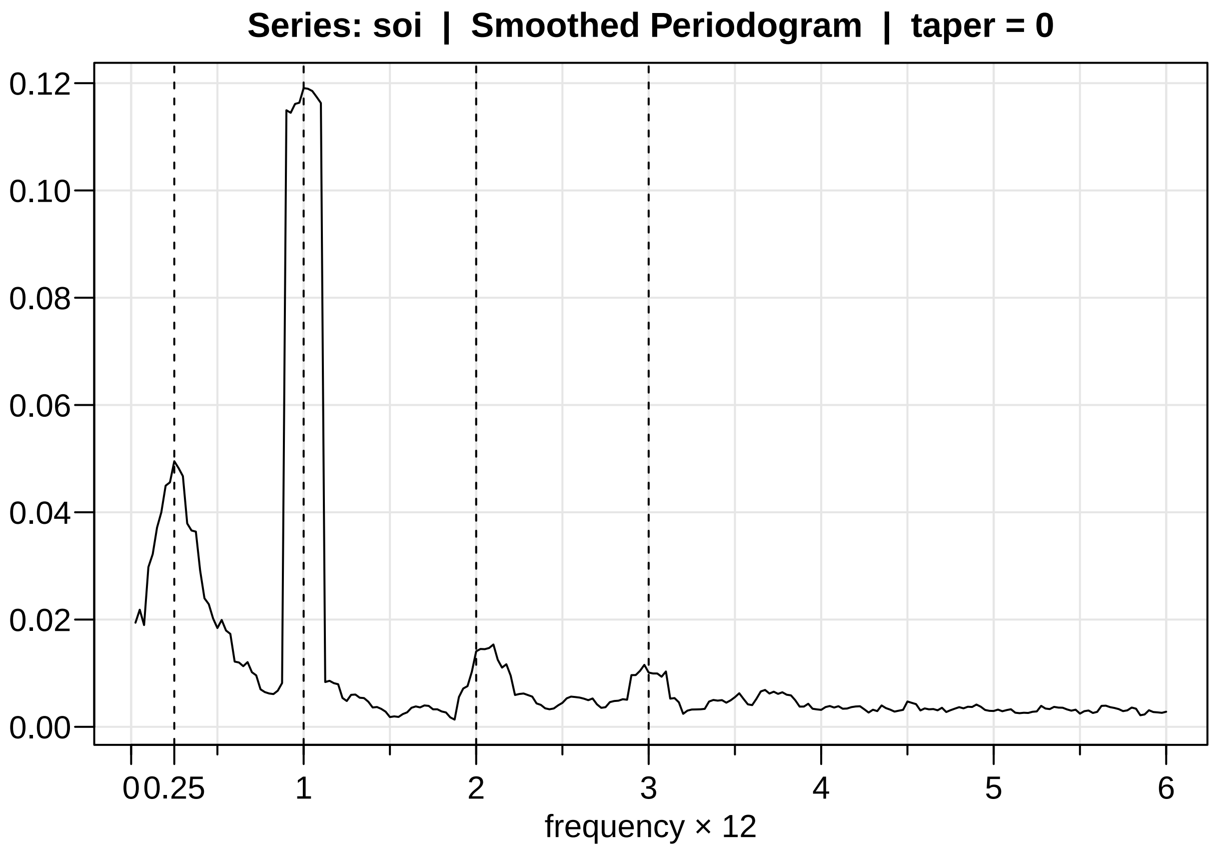
An approximate $100(1-\alpha)\%$ confidence interval for $f(\omega)$

$$\frac{2L\bar{f}(\omega)}{\chi^2_{2L}(1-\alpha/2)} \le f(\omega) \le \frac{2L\bar{f}(\omega)}{\chi^2_{2L}(\alpha/2)}$$

```r
plot(kernel("daniell", 4))
```




```r
par(mgp = c(3, 2, 0), mar = c(3.5, 4, 1.4, 0.6), las = 1)
soi.ave <- mvspec(soi, kernel('daniell', 4), ylab = "")
abline(v = c(.25, 1, 2, 3), lty = 2)
axis(1, at = 0.25)
```




```
soi.ave$bandwidth
```

```
## [1] 0.225
```

```
(df <- soi.ave$df)
```

```
## [1] 16.9875
```

```
(U <- qchisq(.025, df))
```

```
## [1] 7.555916
```

```
(L <- qchisq(.975, df))
```

```
## [1] 30.17425
```

```
soi.ave$spec[10]
```

```
## [1] 0.04952026
```

```
soi.ave$spec[40]
```

```
## [1] 0.11908
```

```r
# intervals
c(df * soi.ave$spec[10] / L, df * soi.ave$spec[10] / U)
```

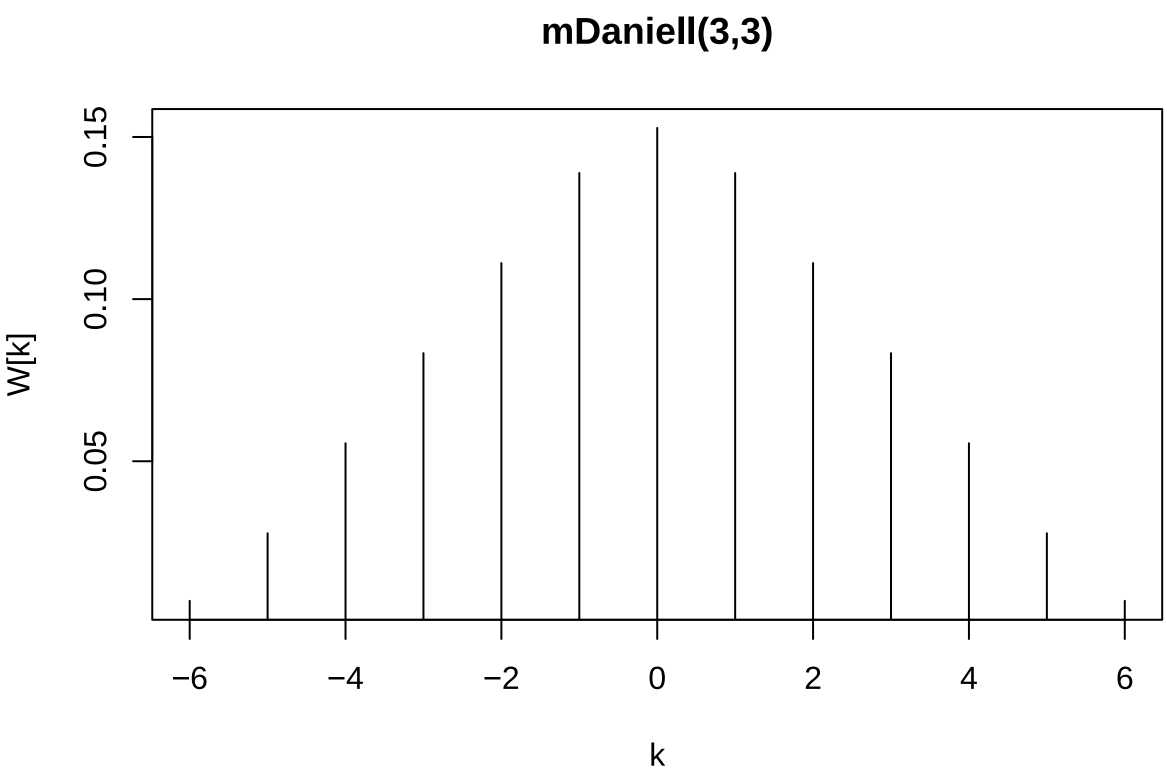
```
## [1] 0.02787891 0.11133335
```

```r
c(df * soi.ave$spec[40] / L, df * soi.ave$spec[40] / U)
```

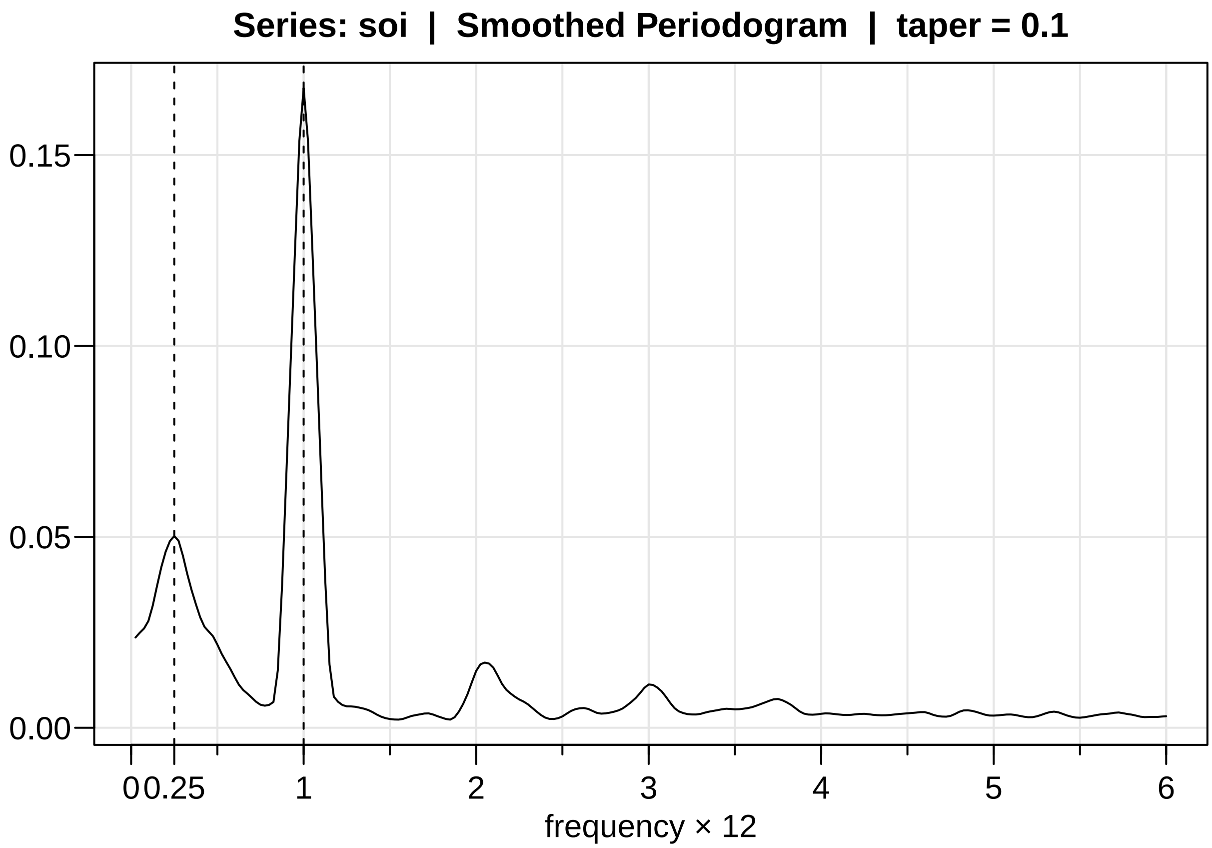
```
## [1] 0.06703963 0.26772011
```

Smoothed periodogram

```r
plot(kernel("modified.daniell", c(3, 3)))
```




```r
k <- kernel("modified.daniell", c(3, 3))

par(mgp = c(3, 2, 0), mar = c(3.5, 4, 1.4, 0.6), las = 1)
soi.smo <- mvspec(soi, kernel = k, taper = .1, ylab = "")
abline(v = c(.25, 1), lty = 2)
axis(1, at = 0.25)
```



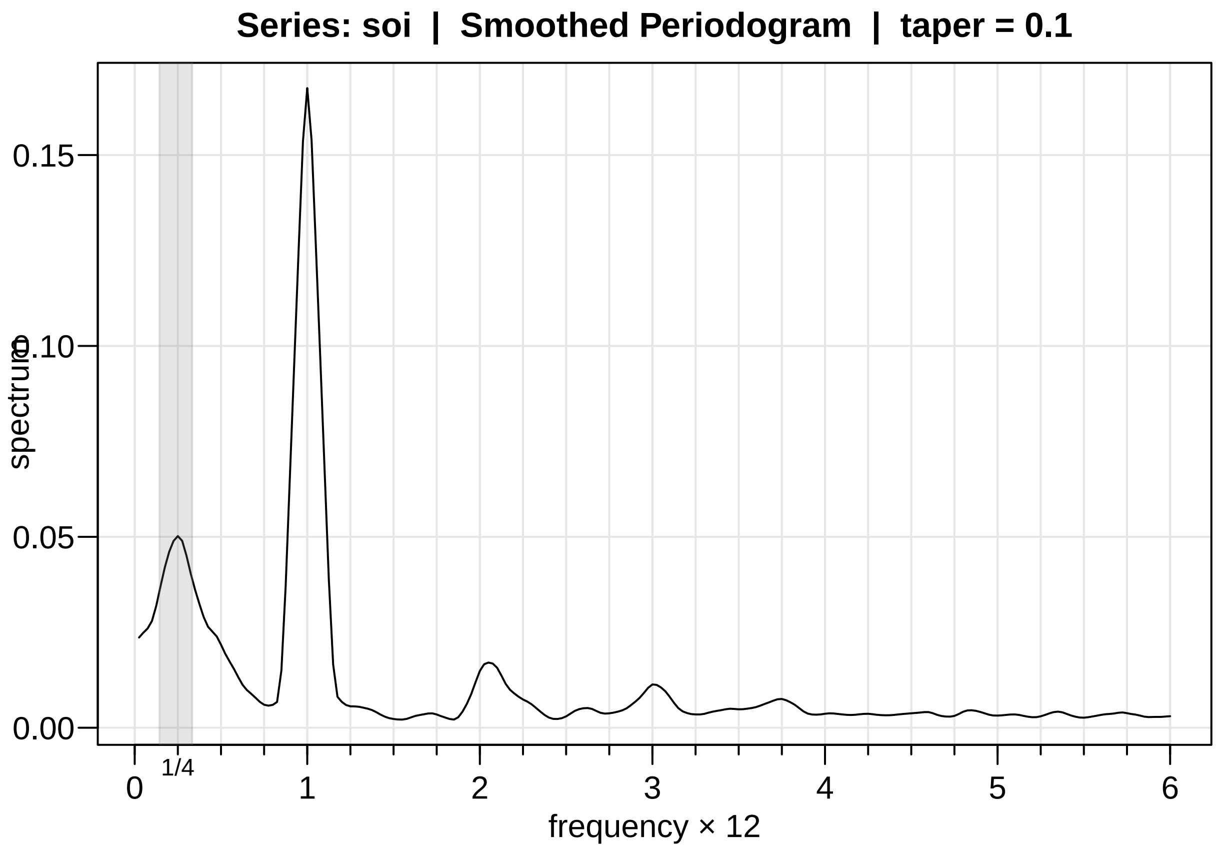

```
soi.smo$bandwidth
```

```
## [1] 0.2308103
```

```
(df <- soi.smo$df)
```

```
## [1] 17.42618
```

```
soi.smo <- mvspec(soi, spans = c(7, 7), taper = .1, nxm = 4)
rect(1/7, -1e5, 1/3, 1e5, density = NA, col = gray(.5, .2))
mtext("1/4", side = 1, line = 0, at = .25, cex = .75)
```




```
(U <- qchisq(.025, df))
```

```
## [1] 7.847084
```

```
(L <- qchisq(.975, df))
```

```
## [1] 30.76132
```

```
soi.smo$spec[10]
```

```
## [1] 0.05019866
```

```
soi.smo$spec[40]
```

```
## [1] 0.1675368
```

```
# intervals
c(df * soi.smo$spec[10] / L, df * soi.smo$spec[10] / U)
```

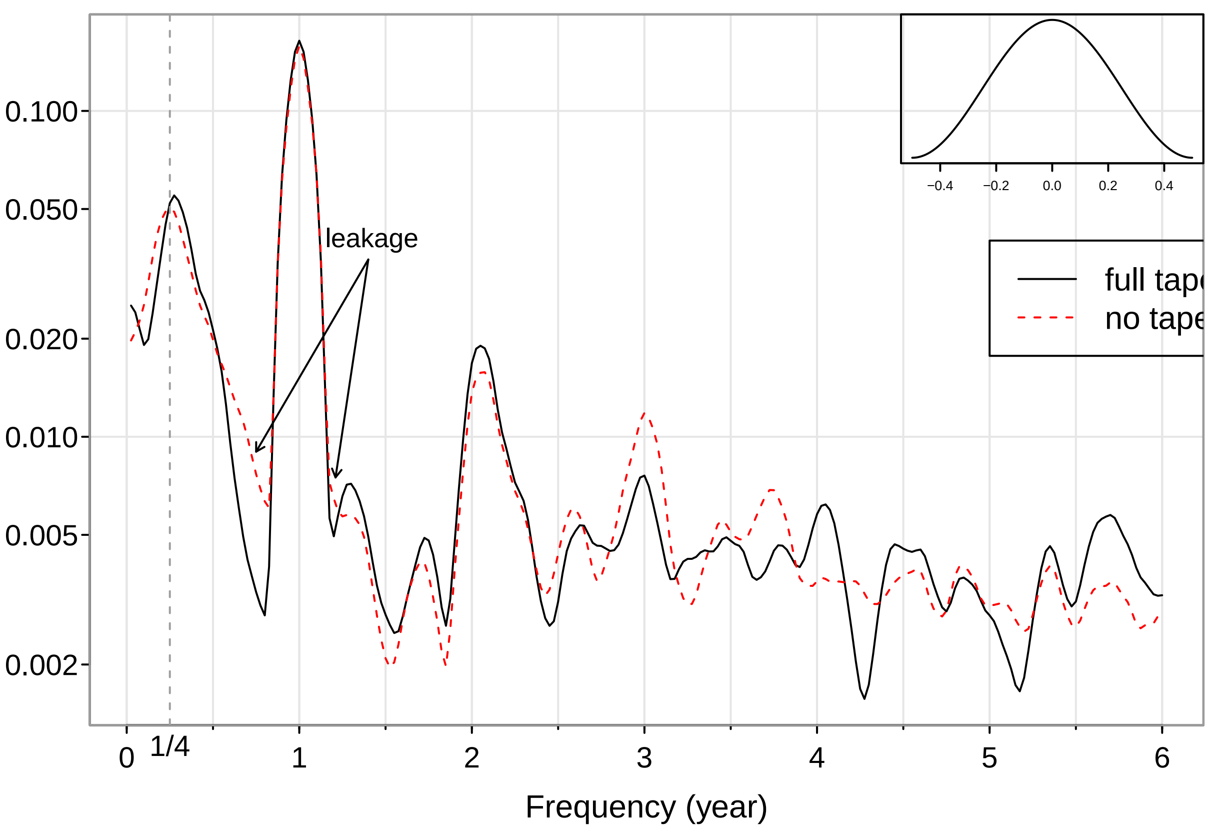
```
## [1] 0.02843736 0.11147718
```

```
c(df * soi.smo$spec[40] / L, df * soi.smo$spec[40] / U)
```

```
## [1] 0.09490899 0.37205231
```

SOI tapering

```
par(mgp = c(3, 2, 0), mar = c(3.5, 4, 1.4, 0.6), las = 1)
s0 <- mvspec(soi, spans = c(7, 7), plot = FALSE)
s50 <- mvspec(soi, spans = c(7, 7), taper = .5, plot = FALSE)
tsplot(s50$freq, s50$spec, log = "y", type = "l",
       ylab = "", xlab = "Frequency (year)")
lines(s0$freq, s0$spec, lty = 2, col = "red")
abline(v = .25, lty = 2, col = 8)
mtext('1/4', side = 1, line = 0, at = .25, cex = .9)
legend(5, .04, legend = c('full taper', 'no taper'), lty = 1:2, col = c("black", "red"))

text(1.42, 0.04, 'leakage', cex = .8)
arrows(1.4, .035, .75, .009, length = 0.05, angle = 30)
arrows(1.4, .035, 1.21, .0075, length = 0.05, angle = 30)
par(fig = c(.65, 1, .65, 1),  new = TRUE, cex = .5,
    mgp = c(0, -.1, 0), tcl = -.2)
taper <- function(x) {.5 * (1 + cos(2 * pi * x))}
x <- seq(from = -.5, to = .5, by = 0.001)
plot(x, taper(x), type = "l",  lty = 1,  yaxt = 'n', ann = FALSE)
```



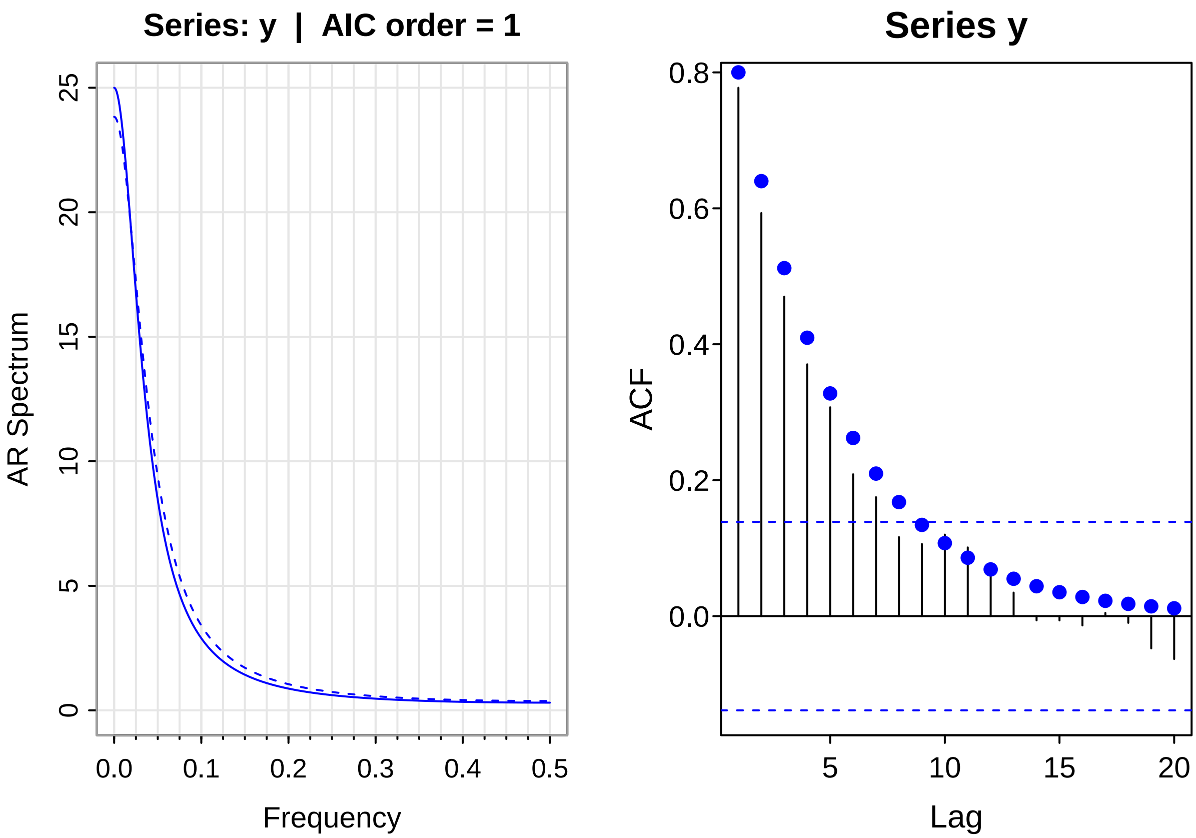
Parametric Spectral Estimation

Simulated examples

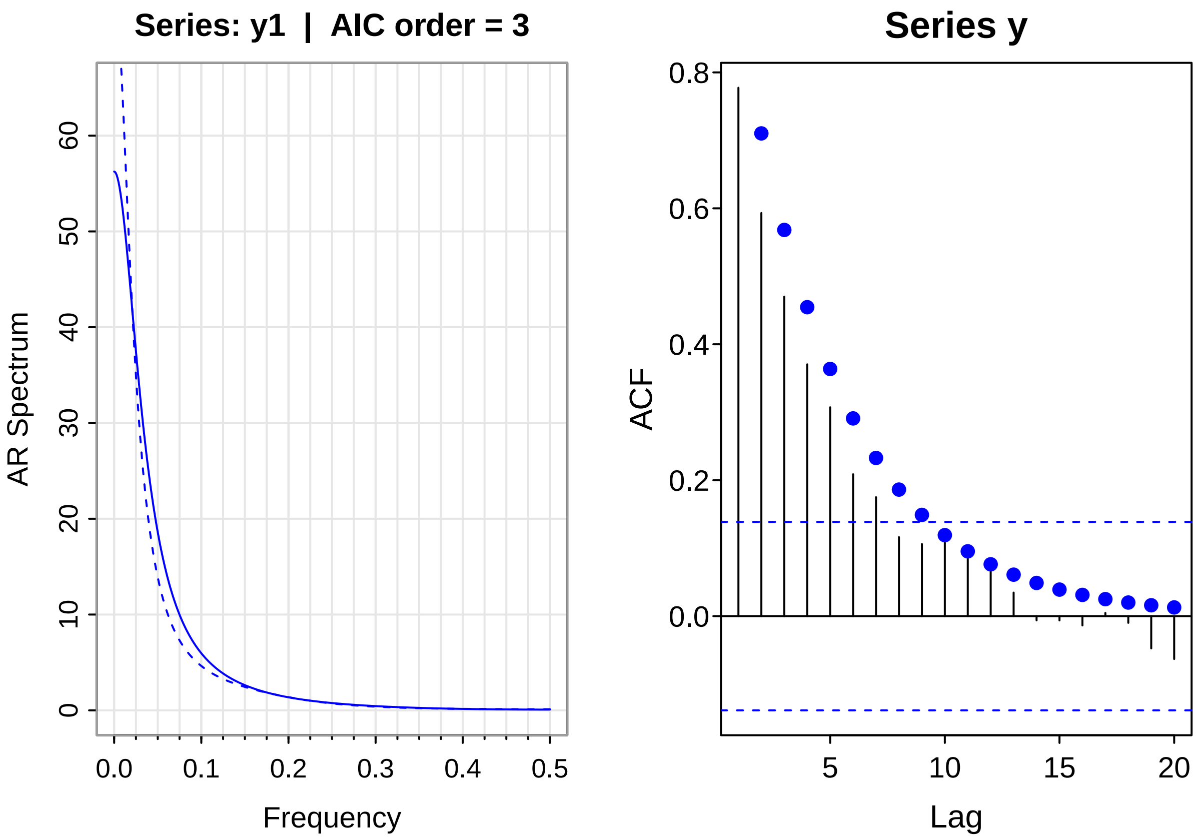
```r
library(TSA)
```

```
##
## Attaching package: 'TSA'

## The following objects are masked from 'package:stats':
##
##     acf, arima

## The following object is masked from 'package:utils':
##
##     tar
```

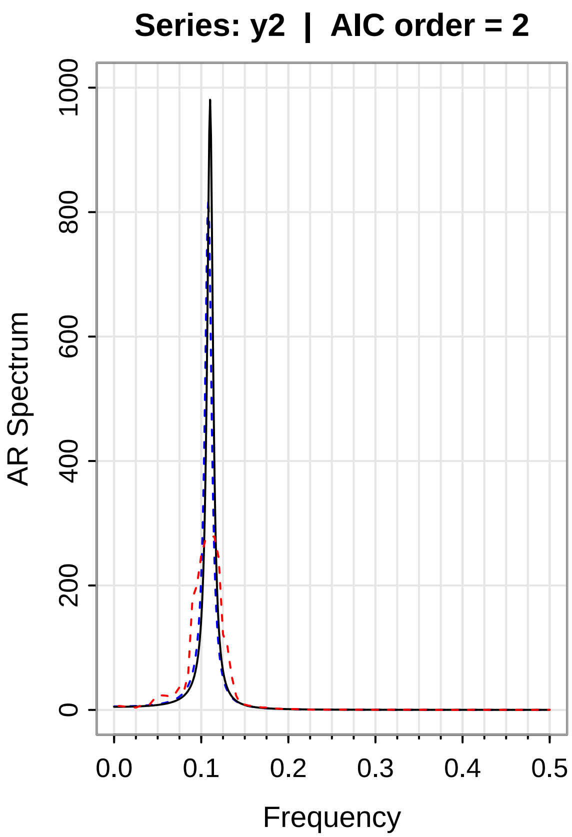
```r
set.seed(12345)
n = 200; phi = 0.8; theta = 0.5
phi1 = 1.5; phi2 = -.95
y <- arima.sim(model = list(ar = phi), n = n)
y1 <- arima.sim(model = list(ar = phi, ma = theta), n = n)
y2 <- arima.sim(model = list(ar = c(phi1, phi2)), n = n)
##AR(1)
par(las = 1, mar = c(4, 4, 2, 0.6), mgp = c(3, 1, 0), mfrow = c(1, 2))
spec <- spec.ic(y, detrend = F, col = "blue", lwd = 1, nxm = 4,
                ylim = c(0, 25), lty = 2)
freq <- spec[[2]][, 1]
lines(freq, ARMAspec(model = list(ar = phi), freq = freq,
plot = F)$spec, col = "blue")
acf(y, lag = 20)
acf_true <- ARMAacf(ar = phi, lag.max = 20)
points(1:20, acf_true[2:21], col = "blue", pch = 16)
```




```
##ARMA(1,1)
par(las = 1, mar = c(4, 4, 2, 0.6), mgp = c(3, 1, 0), mfrow = c(1, 2))
spec <- spec.ic(y1, detrend = F, col = "blue", lwd = 1, nxm = 4, ylim = c(0, 65), lty = 2)
freq <- spec[[2]][, 1]
lines(freq, ARMAspec(model = list(ar = phi, ma = theta), freq = freq,
plot = F)$spec, col = "blue")
acf(y, lag = 20)
acf_true <- ARMAacf(ar = phi, ma = theta, lag.max = 20)
points(1:20, acf_true[2:21], col = "blue", pch = 16)
```

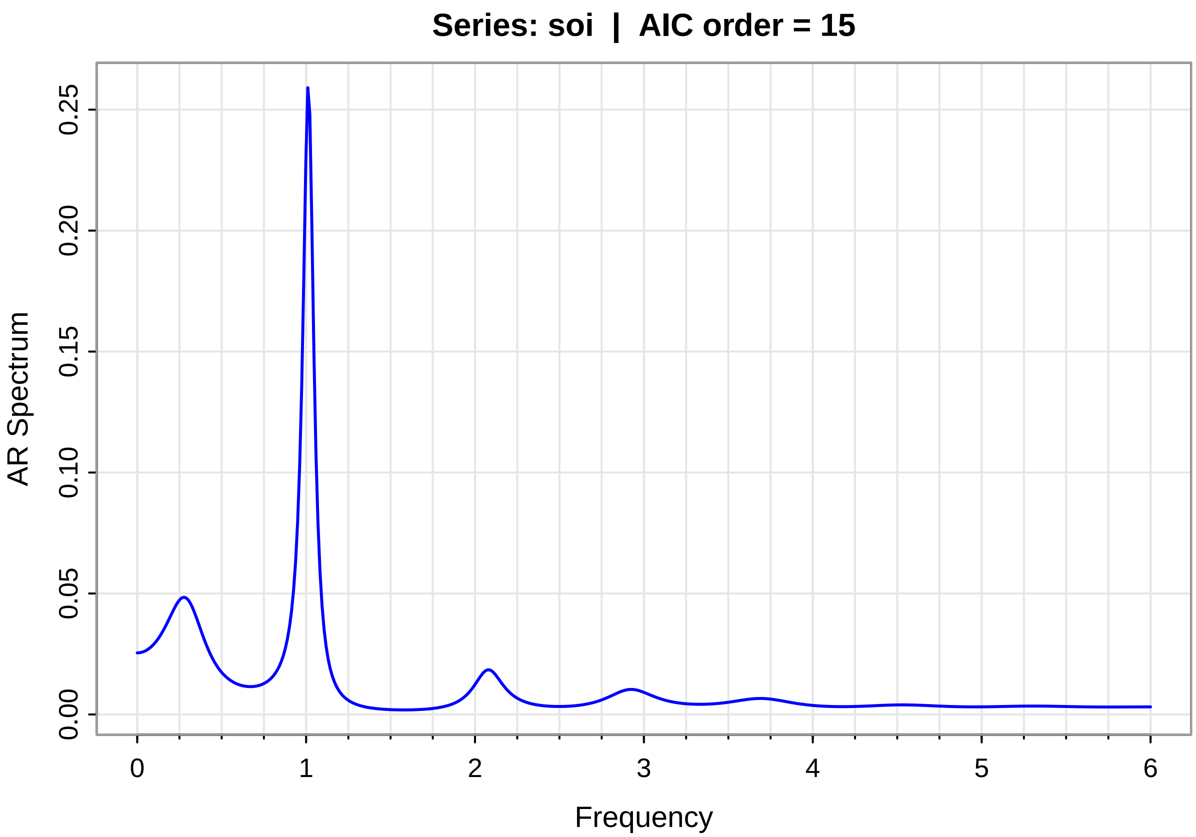



```r
##AR(2)
par(las = 1, mar = c(4, 4, 2, 0.6), mgp = c(3, 1, 0))
spec <- spec.ic(y2, detrend = F, col = "blue", lwd = 1, nxm = 4, ylim = c(0, 1000), lty = 2)
freq <- spec[[2]][, 1]
lines(freq, ARMAspec(model = list(ar = c(phi1, phi2)), freq = freq,
plot = F)$spec)

AR2_nonpar <- mvspec(y2, kernel('daniell', 3), plot = F)
lines(AR2_nonpar$freq, AR2_nonpar$spec, col = "red", lty = 2)
```

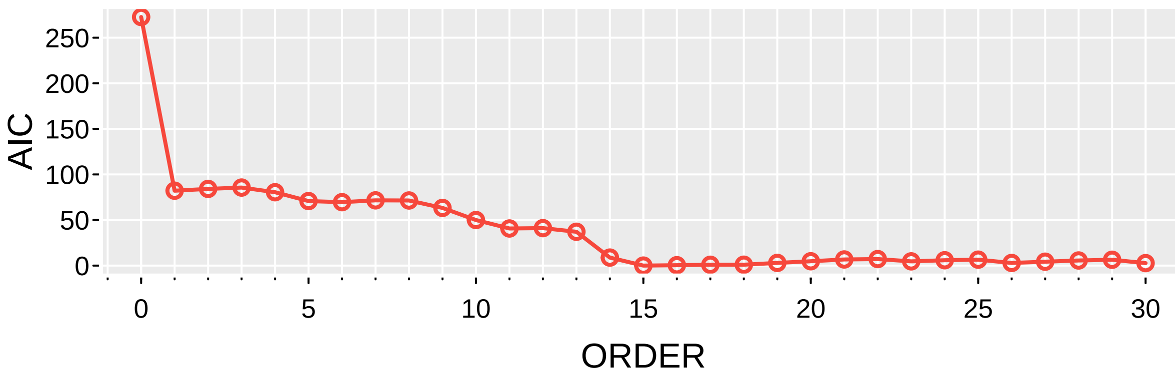


SOI Example

```
par(las = 1, mar = c(4, 4, 2, 0.6), mgp = c(3, 1, 0))
u <- spec.ic(soi, detrend = TRUE, col = "blue", lwd = 1.5, nxm = 4)
```



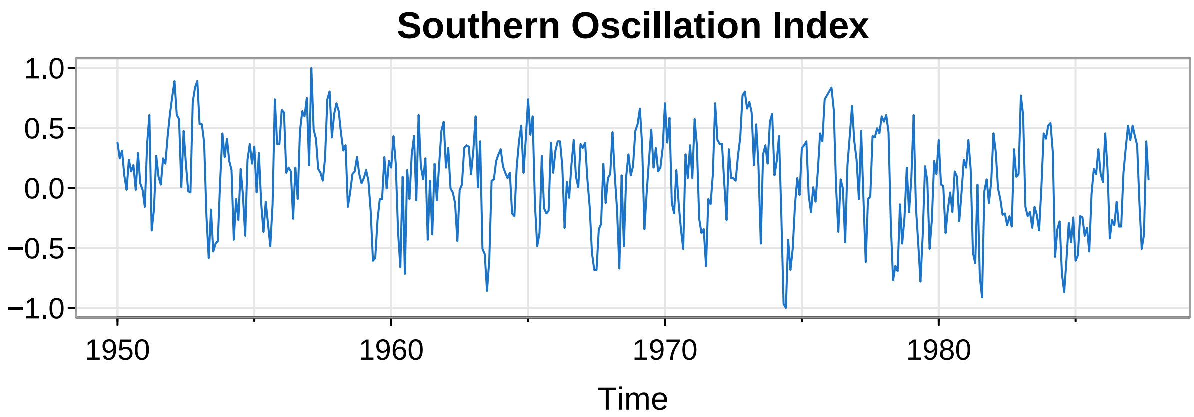

```r
# plot AIC and BIC
tsplot(0:30, u[[1]][, 2:3], type = 'o', col = 2:3,
       xlab = 'ORDER', nxm = 5, lwd = 2, gg = TRUE)
```

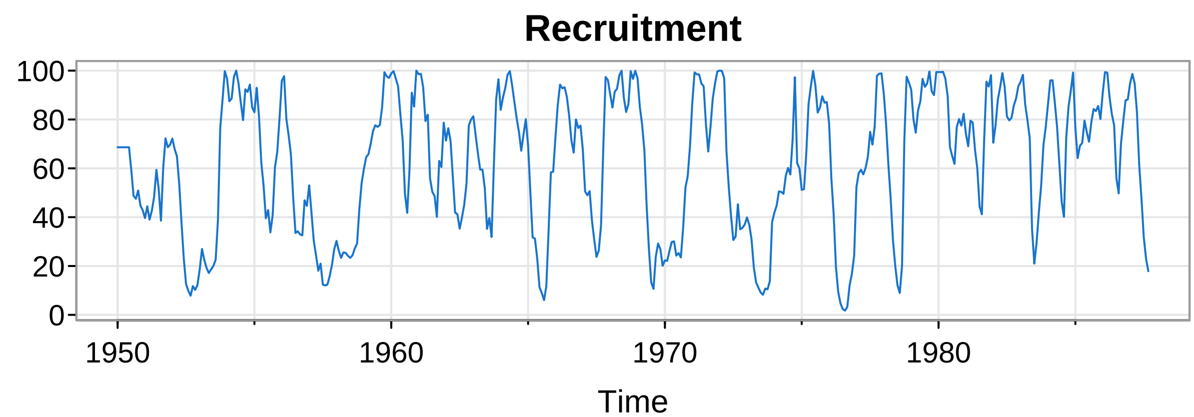


Lagged regression

SOI and Recruitment time series

```
par(mfrow = c(2, 1), las = 1)
tsplot(soi, col = 4, ylab = "", main = "Southern Oscillation Index")
tsplot(rec, col = 4, ylab = "", main = "Recruitment")
```

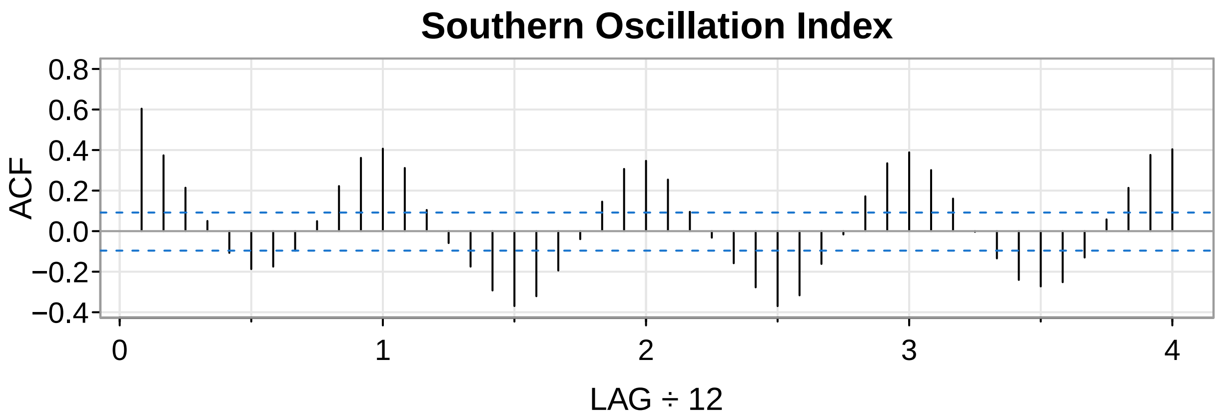


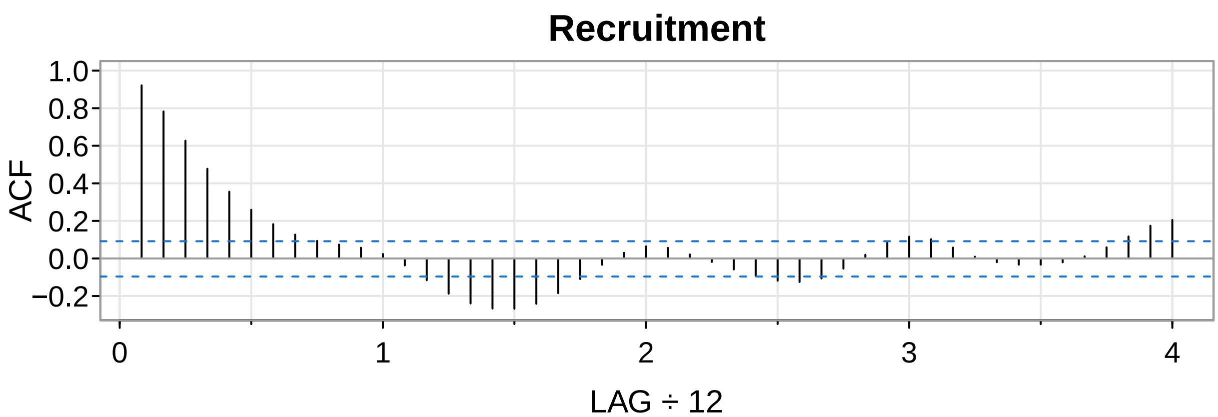



```r
acf1(soi, main = "Southern Oscillation Index")
```

```
##  [1]  0.60  0.37  0.21  0.05 -0.11 -0.19 -0.18 -0.10  0.05  0.22  0.36  0.41
## [13]  0.31  0.10 -0.06 -0.17 -0.29 -0.37 -0.32 -0.19 -0.04  0.15  0.31  0.35
## [25]  0.25  0.10 -0.03 -0.16 -0.28 -0.37 -0.32 -0.16 -0.02  0.17  0.33  0.39
## [37]  0.30  0.16  0.00 -0.13 -0.24 -0.27 -0.25 -0.13  0.06  0.21  0.38  0.40
```

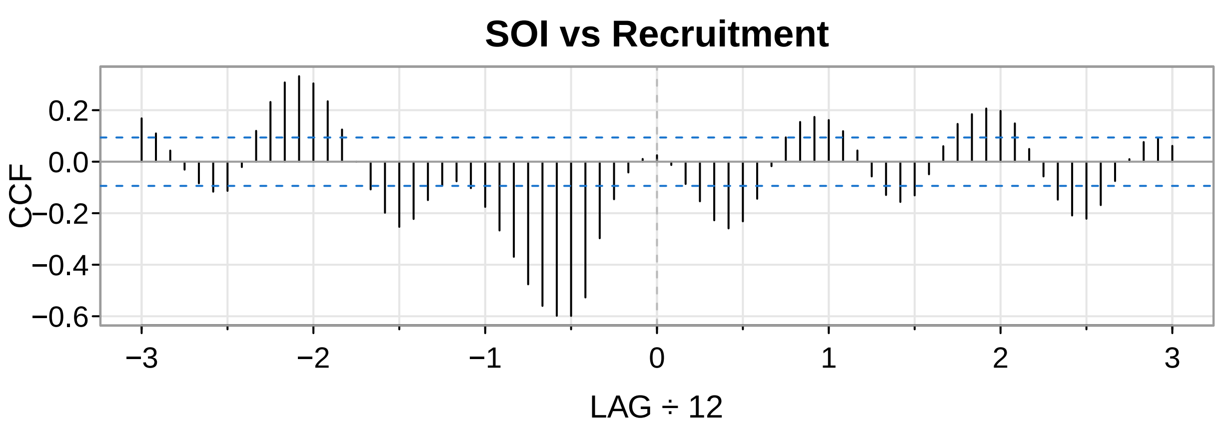
```r
acf1(rec, main = "Recruitment")
```





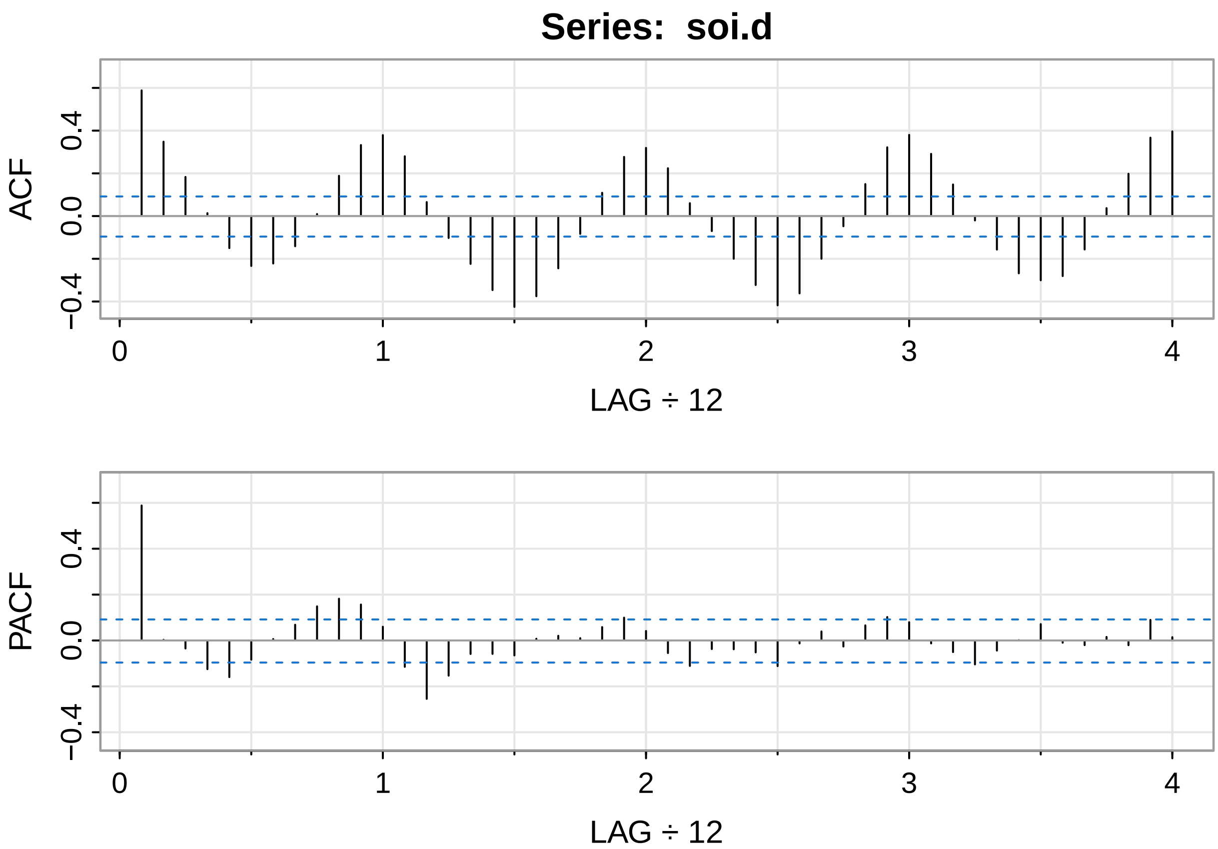

```
##  [1]  0.92  0.78  0.63  0.48  0.36  0.26  0.18  0.13  0.09  0.07  0.06  0.02
## [13] -0.04 -0.12 -0.19 -0.24 -0.27 -0.27 -0.24 -0.19 -0.11 -0.03  0.03  0.06
## [25]  0.06  0.02 -0.02 -0.06 -0.09 -0.12 -0.13 -0.11 -0.05  0.02  0.08  0.12
## [37]  0.10  0.06  0.01 -0.02 -0.03 -0.03 -0.02  0.01  0.06  0.12  0.17  0.20
```

```
ccf2(soi, rec, main = "SOI vs Recruitment", las = 1)
```



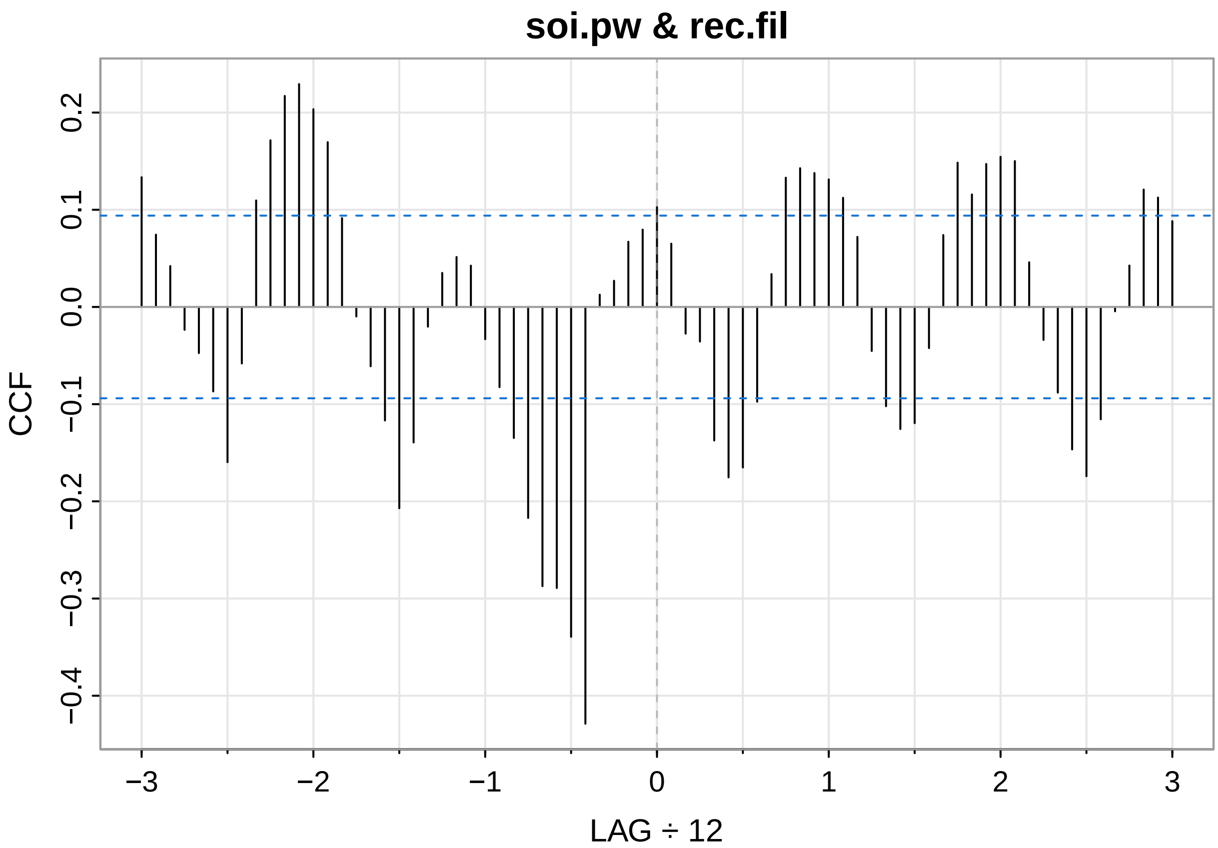
Transfer function modeling

```r
soi.d <- resid(lm(soi ~ time(soi), na.action = NULL))
acf2(soi.d)
```




```
##      [,1] [,2]  [,3]  [,4]  [,5]  [,6]  [,7]  [,8] [,9] [,10] [,11] [,12] [,13]
## ACF  0.59 0.35  0.18  0.01 -0.15 -0.23 -0.22 -0.14 0.01  0.19  0.33  0.38  0.28
## PACF 0.59 0.00 -0.03 -0.12 -0.16 -0.08  0.01  0.07 0.15  0.18  0.16  0.06 -0.11
##      [,14] [,15] [,16] [,17] [,18] [,19] [,20] [,21] [,22] [,23] [,24] [,25]
## ACF   0.07 -0.10 -0.22 -0.35 -0.43 -0.38 -0.24 -0.08  0.11  0.28  0.32  0.22
## PACF -0.25 -0.15 -0.06 -0.06 -0.07  0.01  0.02  0.01  0.06  0.10  0.04 -0.05
##      [,26] [,27] [,28] [,29] [,30] [,31] [,32] [,33] [,34] [,35] [,36] [,37]
## ACF   0.06 -0.07 -0.20 -0.32 -0.42 -0.36 -0.20 -0.05  0.15  0.32  0.38  0.29
## PACF -0.11 -0.04 -0.04 -0.05 -0.11 -0.01  0.04 -0.03  0.07  0.10  0.08 -0.01
##      [,38] [,39] [,40] [,41] [,42] [,43] [,44] [,45] [,46] [,47] [,48]
## ACF   0.15 -0.02 -0.16 -0.27 -0.30 -0.28 -0.16  0.04  0.20  0.37  0.40
## PACF -0.05 -0.10 -0.04  0.00  0.07 -0.01 -0.02  0.02 -0.02  0.09  0.01
```

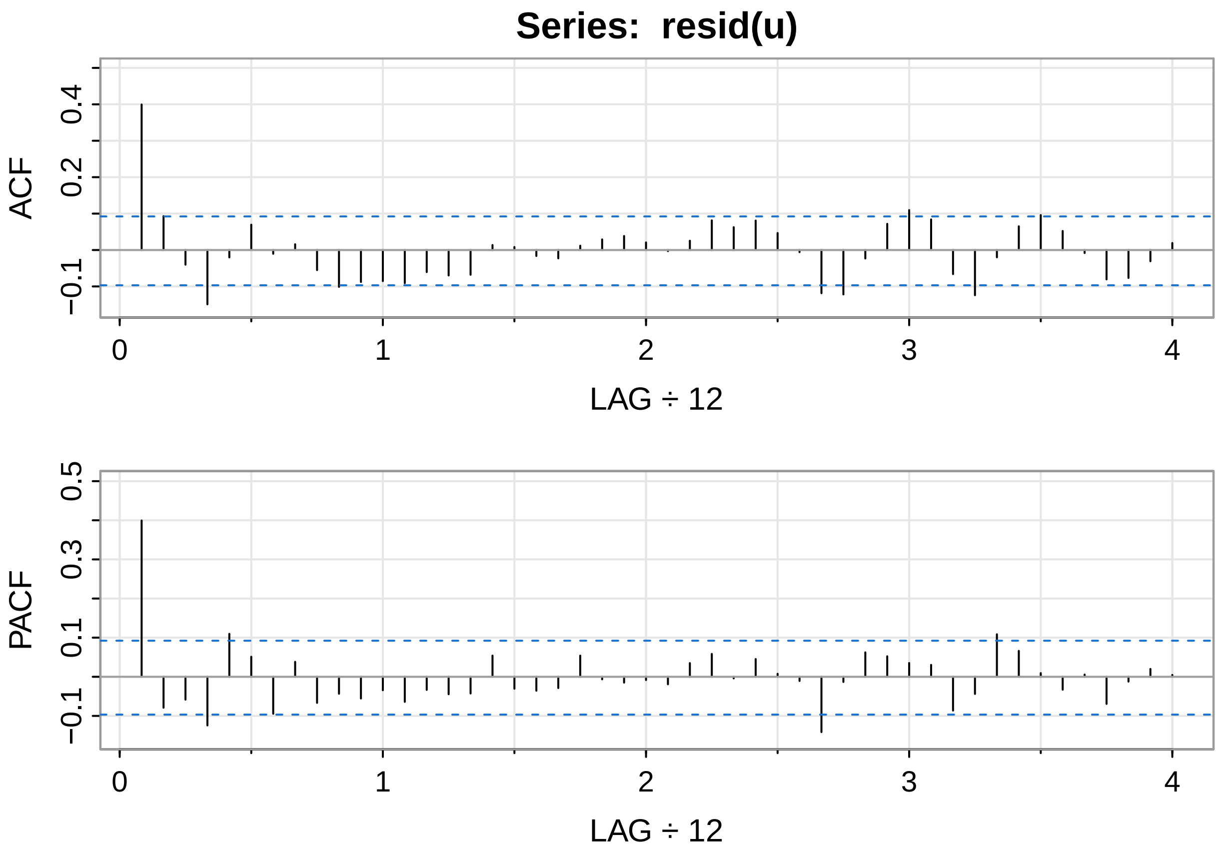
```r
fit <- arima(soi.d, order = c(1, 0, 0))
ar1 <- as.numeric(coef(fit)[1])
soi.pw <- resid(fit)
rec.fil <- filter(rec, filter = c(1, -ar1), sides = 1)
ccf2(soi.pw, rec.fil)
```




```r
fish <- ts.intersect(rec, RL1 = lag(rec, -1), SL5 = lag(soi.d, -5))
(u <- lm(fish[, 1] ~ fish[, 2:3], na.action = NULL))
```

```
##
## Call:
## lm(formula = fish[, 1] ~ fish[, 2:3], na.action = NULL)
##
## Coefficients:
##    (Intercept)  fish[, 2:3]RL1  fish[, 2:3]SL5
##         8.8971          0.8556        -20.3771
```

```r
acf2(resid(u))
```

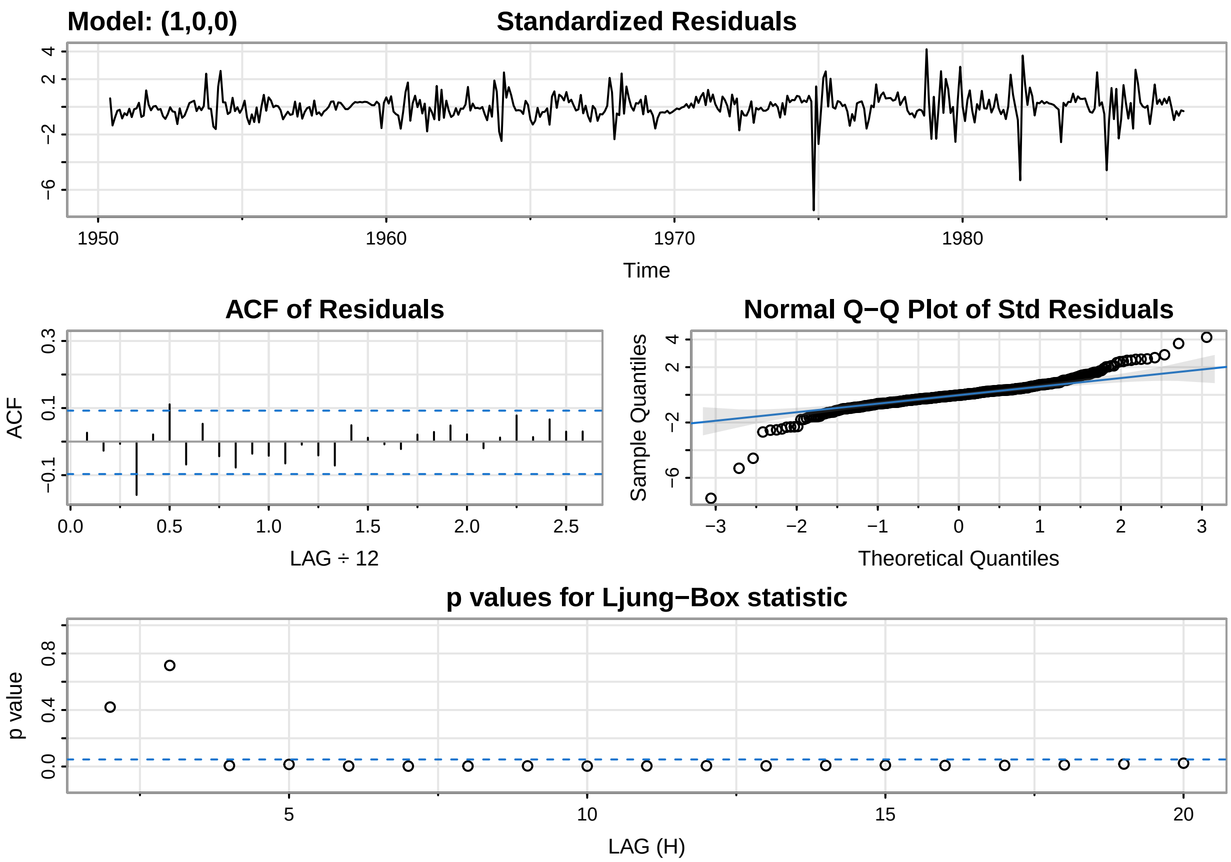



```
##       [,1]  [,2]  [,3]  [,4]  [,5] [,6]  [,7] [,8]  [,9] [,10] [,11] [,12] [,13]
## ACF   0.4  0.09 -0.04 -0.15 -0.02 0.07 -0.01 0.02 -0.06 -0.10 -0.09 -0.09 -0.09
## PACF  0.4 -0.08 -0.06 -0.12  0.11 0.05 -0.09 0.04 -0.07 -0.04 -0.06 -0.03 -0.06
##       [,14] [,15] [,16] [,17] [,18] [,19] [,20] [,21] [,22] [,23] [,24] [,25]
## ACF  -0.06 -0.07 -0.07  0.01  0.01 -0.02 -0.02  0.01  0.03  0.04  0.02  0.00
## PACF -0.03 -0.04 -0.04  0.05 -0.03 -0.04 -0.03  0.05 -0.01 -0.01 -0.01 -0.02
##       [,26] [,27] [,28] [,29] [,30] [,31] [,32] [,33] [,34] [,35] [,36] [,37]
## ACF   0.03  0.08  0.06  0.08  0.05 -0.01 -0.12 -0.12 -0.02  0.07  0.11  0.08
## PACF  0.03  0.06  0.00  0.05  0.01 -0.01 -0.14 -0.01  0.06  0.05  0.04  0.03
##       [,38] [,39] [,40] [,41] [,42] [,43] [,44] [,45] [,46] [,47] [,48]
## ACF  -0.07 -0.12 -0.02  0.07  0.10  0.05 -0.01 -0.08 -0.08 -0.03  0.02
## PACF -0.09 -0.04  0.11  0.07  0.01 -0.03  0.01 -0.07 -0.01  0.02  0.00
```

```r
(arx <- sarima(fish[, 1], 1, 0, 0, xreg = fish[, 2:3]))
```

```
## initial  value 2.050589
## iter   2 value 1.963560
## iter   3 value 1.962035
## iter   4 value 1.956727
## iter   5 value 1.956486
## iter   6 value 1.956230
## iter   7 value 1.956056
## iter   8 value 1.956027
## iter   9 value 1.956024
## iter  10 value 1.956024
## iter  10 value 1.956024
```

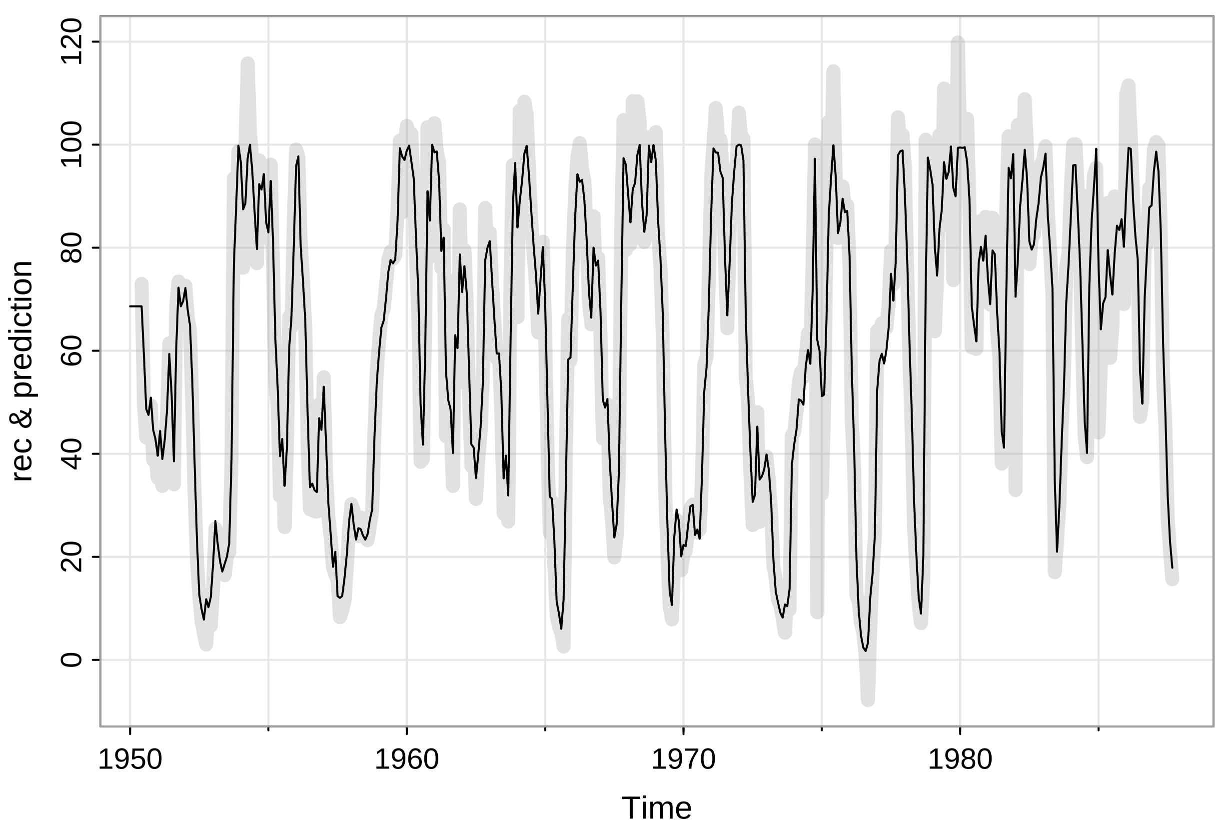
```
## final  value 1.956024
## converged
## initial  value 1.955587
## iter   2 value 1.955586
## iter   3 value 1.955585
## iter   4 value 1.955584
## iter   5 value 1.955584
## iter   6 value 1.955584
## iter   7 value 1.955584
## iter   8 value 1.955584
## iter   8 value 1.955584
## iter   8 value 1.955584
## final  value 1.955584
## converged
```




```
## $fit
##
## Call:
## stats::arima(x = xdata, order = c(p, d, q), seasonal = list(order = c(P, D,
##     Q), period = S), xreg = xreg, transform.pars = trans, fixed = fixed, optim.control = list(trace =
##     REPORT = 1, reltol = tol))
##
## Coefficients:
##          ar1  intercept     RL1      SL5
##       0.4487    12.3323  0.8005  -21.0307
## s.e.  0.0503     1.5746  0.0234    1.0915
```

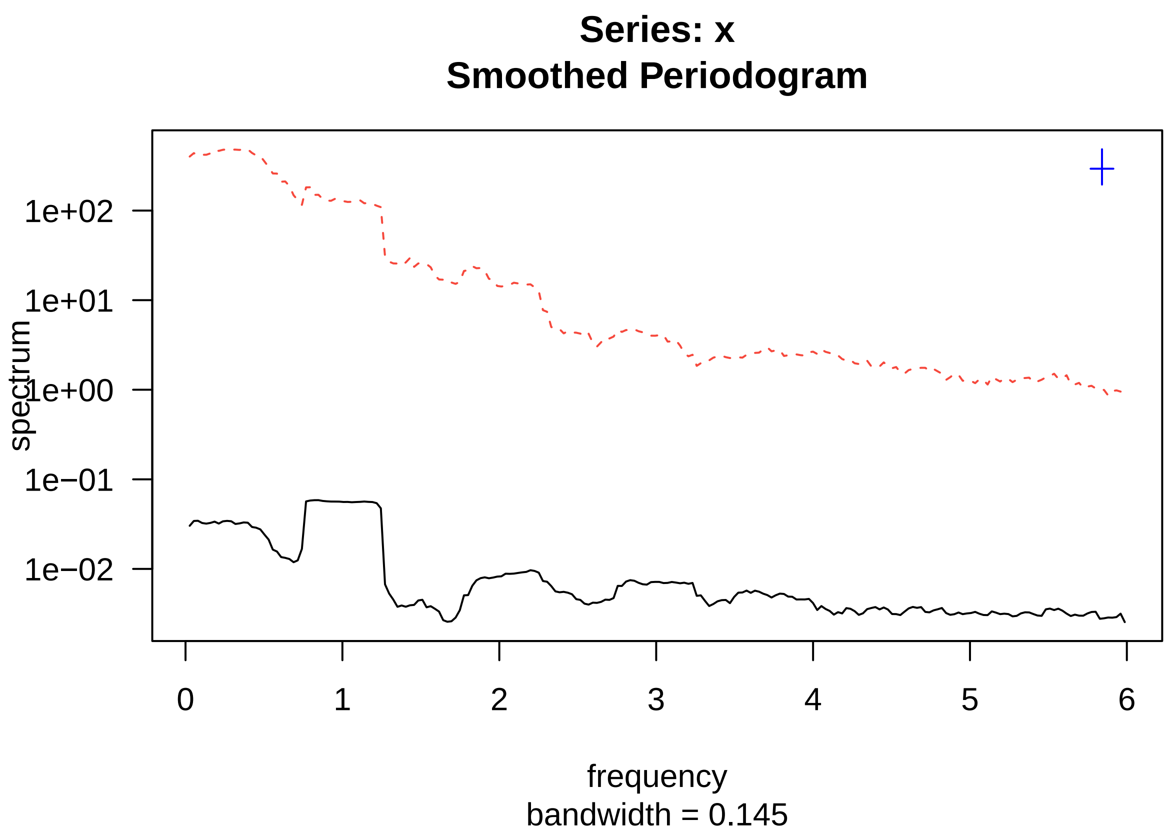
```
##
## sigma^2 estimated as 49.93:  log likelihood = -1511.79,  aic = 3033.57
##
## $degrees_of_freedom
## [1] 444
##
## $ttable
##           Estimate     SE  t.value p.value
## ar1         0.4487 0.0503   8.9183       0
## intercept  12.3323 1.5746   7.8321       0
## RL1         0.8005 0.0234  34.2778       0
## SL5       -21.0307 1.0915 -19.2674       0
##
## $AIC
## [1] 6.771366
##
## $AICc
## [1] 6.771567
##
## $BIC
## [1] 6.817178
```

```
pred <- rec + resid(arx$fit)
tsplot(pred, col = astsa.col(8, .3), lwd = 7, ylab = 'rec & prediction')
lines(rec)
```

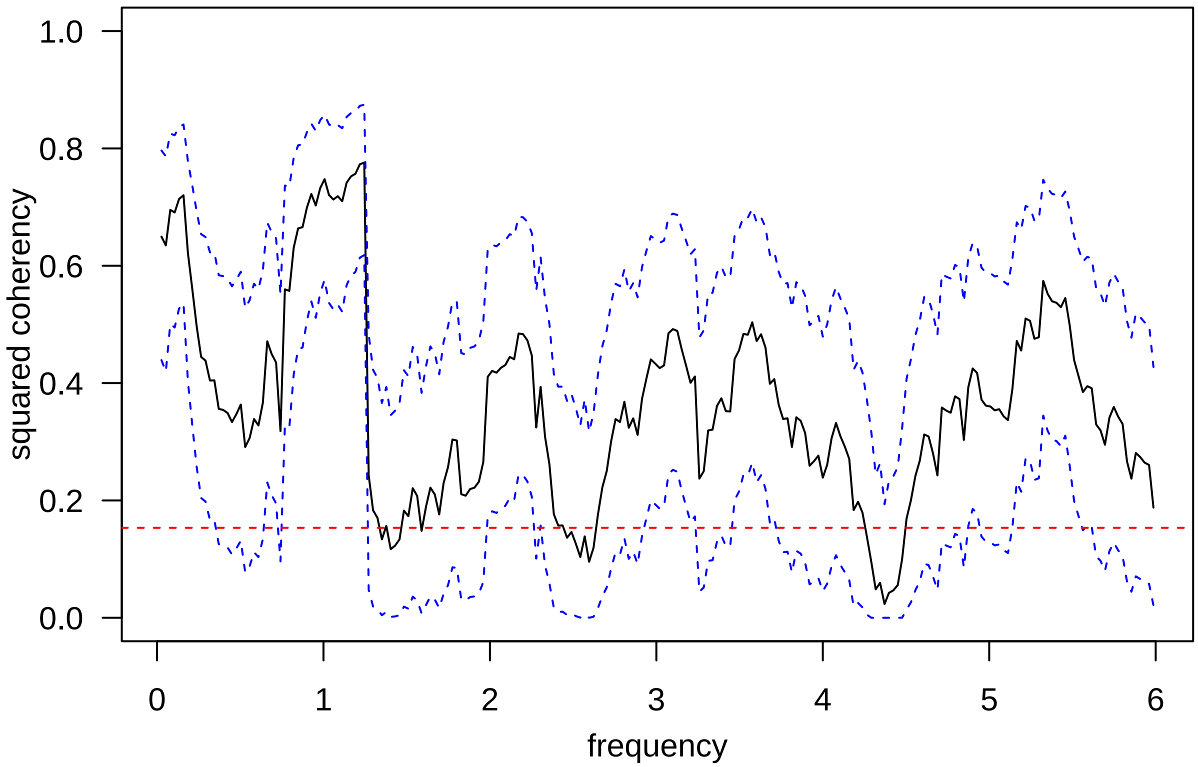


Estimating cross-spectrum

```
par(las = 1)
s = spectrum(cbind(soi, rec), kernel("daniell", 9), taper = 0, fast = FALSE)
```



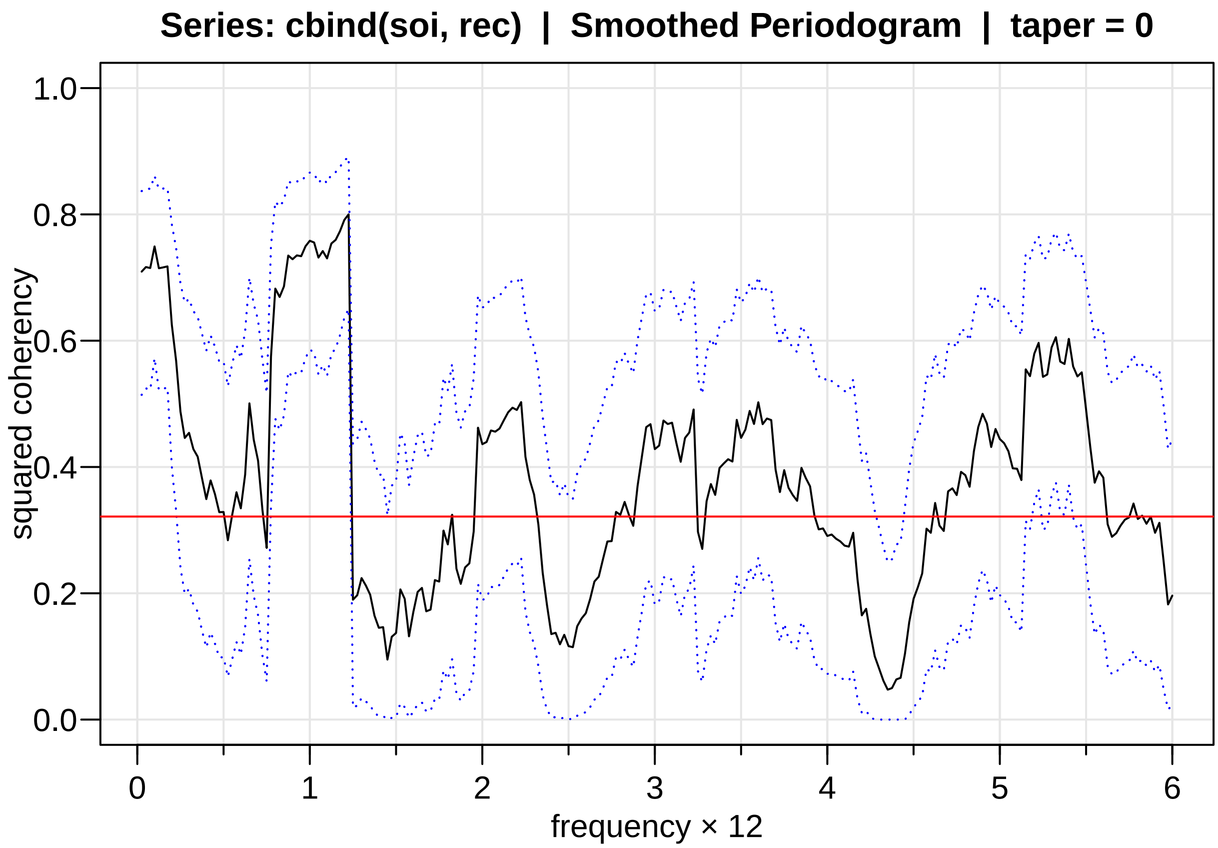

```
par(las = 1, mar = c(4, 4, 2, 0.6), mgp = c(2.2, 1, 0))
plot(s, plot.type = "coh", ci.lty = 2, main = "")
f = qf(.95, 2, s$df - 2);
abline(h = f / ((s$df - 2) / 2 + f), col = "red", lty = 2)
```




```r
sr <- mvspec(cbind(soi, rec), kernel("daniell", 9), plot.type = "coh")
sr$df
```

```
## [1] 35.8625
```

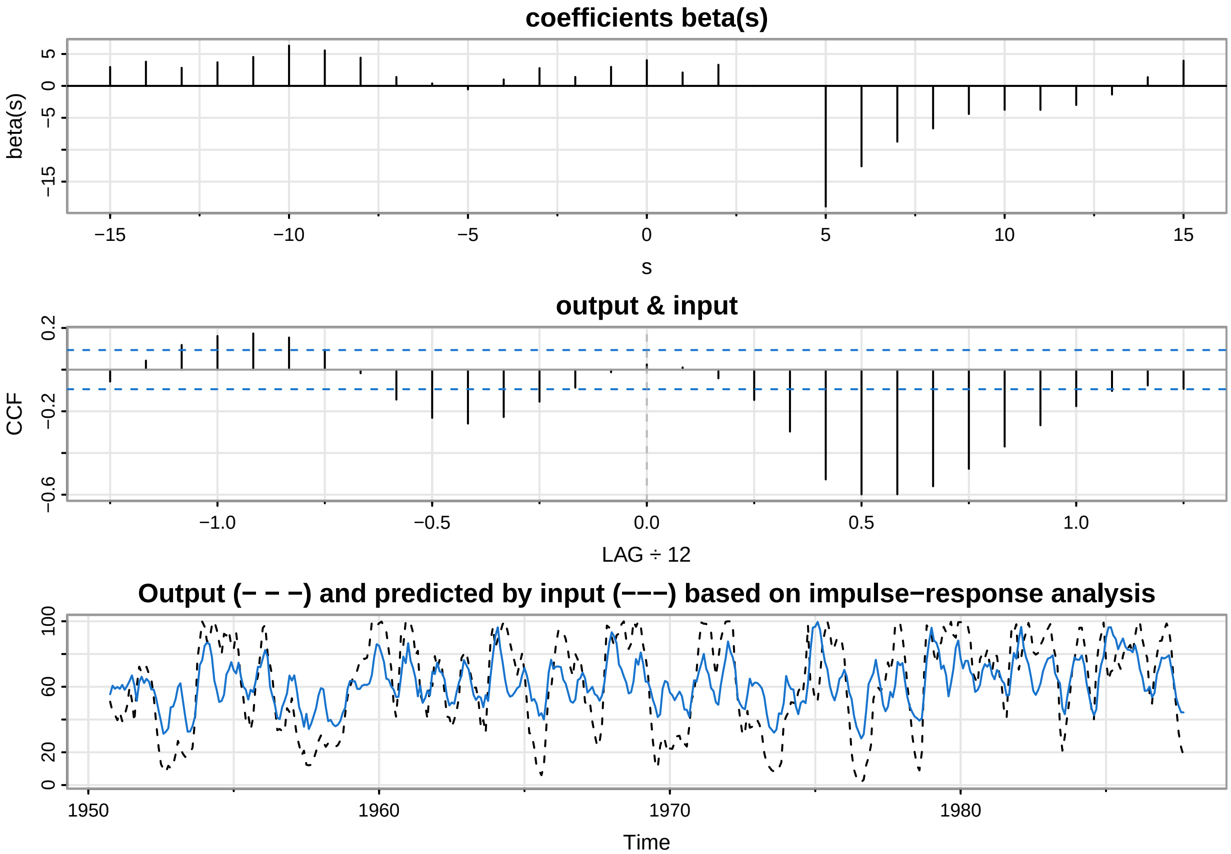
```r
f = qf(.999, 2, sr$df - 2)
C = f / (18 + f)
abline(h = C, col = "red")
```



Lagged Regression in frequency domain

```r
LagReg_SOI2REC <- LagReg(soi, rec, L = 15, M = 32, threshold = 6)
```

```
## INPUT: soi OUTPUT: rec   L = 15    M = 32
##
## The coefficients beta(0), beta(1), beta(2) ... beta(M/2-1) are
##
## 4.03743 2.103372 3.31812 0.01247538 0.005194443 -18.90914 -12.60978 -8.746491
## -6.670373 -4.404543 -3.748336 -3.760936 -2.991477 -1.355261 1.375379 3.955252
##
##
##
## The coefficients beta(0), beta(-1), beta(-2) ... beta(-M/2+1) are
##
## 4.03743 2.987159 1.409949 2.788212 1.017324 -0.5528797 0.402843 1.389537
## 4.426287 5.563582 6.315986 4.540402 3.703423 2.840445 3.798354 2.974338

## The positive lags, at which the coefficients are large
## in absolute value, and the coefficients themselves, are:
##      lag s    beta(s)
## [1,]     5 -18.909140
## [2,]     6 -12.609781
## [3,]     7  -8.746491
## [4,]     8  -6.670373
```




```
##
## The prediction equation is
## rec(t) = alpha + sum_s[ beta(s)*soi(t-s) ], where alpha = 66.01941
## MSE = 411.5948
```

The fitted model is

$$y_t = 66.02 - 18.91x_{t-5} - 12.61x_{t-6} - 8.75x_{t-7} - 6.67x_{t-8} + w_t.$$