Multiple time series

Given: two series Y and X.

Relationship between series?

Possible approaches:

- X deterministic: regress Y on X via generalized least squares: `arima.mle` in SPlus or `arima` in R. We have done this.

- X random; interested in impact of X on Y. Time series analogue of regression.

- X random; interested in joint behaviour. Multivariate ARIMA modelling.

Definition: Two processes X and Y are *jointly (strictly) stationary* if

$$\mathcal{L}(X_t, \ldots, X_{t+h}, Y_t, \ldots, Y_{t+h}) \\ = \mathcal{L}(X_0, \ldots, X_h, Y_0, \ldots, Y_h)$$

for all t and h.

Definition: X and Y are *jointly second order stationary* if each is second order stationary and also

$$C_{XY}(h) \equiv \text{Cov}(X_t, Y_{t+h}) = \text{Cov}(X_0, Y_h)$$

for all t and h.

Notice that negative values of h give, in general, different covariances than positive values of h.

Regression: estimate impulse response

Assume: $\mathrm{E}(X) = \mathrm{E}(Y) = 0$.

Example to illustrate range of models:

$$Y_t = aX_t + bX_{t-2} + cX_{t+4} + \nu_t$$

where ν is "noise", not necessarily white noise.

Crucial: ν independent of X.

General model:

$$Y_t = \sum_{-\infty}^{\infty} a_s X_{t-s} + \nu_t$$

Problem: estimate impulse response function a.

Alternative description

Find best predictor of Y from X?

Minimize

$$\mathsf{E}\left[\{Y_t - f_t(\ldots, X_{t-1}, X_t, \ldots)\}^2\right]$$

over all functions f_t.

Joint stationarity implies:

$$f_{t+1}(BX) = f_t(X)$$

For Gaussian mean 0 series:

$$f_t(X) = \sum a_s X_{t-s}$$

for some coefficients a_s

So: minimize

$$\mathsf{E}\left\{\left(Y_t - \sum_s a_s X_{t-s}\right)^2\right\}$$

Expand out:

$$\mathrm{E}(Y_t^2) - 2\sum a_s \mathrm{E}(Y_t X_{t-s}) + \sum a_r a_s \mathrm{E}(X_{t-r} X_{t-s})$$

which is

$$\mathrm{E}(Y_t^2) - 2\sum a_s C_{XY}(s) + \sum a_r a_s C_X(r-s)$$

Derivative wrt a_k:

$$-2C_{XY}(k) + 2\sum_s a_s C_X(k-s)$$

Set to 0 to find minimum:

$$C_{XY}(k) = \sum_s a_s C_X(k-s)$$

Solve?

With data?

Use spectral methods:

Definition: : convolution of two sequences a_t, b_t is

$$(a * b)_t = \sum_r a_r b_{t-r} = \sum b_r a_{t-r}$$

Fourier transform:

$$f_a(\omega) = \sum a_r e^{2\pi r \omega i}$$

$$\begin{aligned} f_{a*b}(\omega) &= \sum_{r=-\infty}^{\infty} (a * b)_r e^{2\pi\omega r i} \\ &= \sum_s \sum_r a_s b_{r-s} e^{2\pi\omega(r-s+s)i} \\ &= \left(\sum a_s e^{2\pi\omega s i}\right)\left(\sum b_r e^{2\pi\omega r i}\right) \\ &= f_a(\omega) f_b(\omega) \end{aligned}$$

So

$$f_{C_{XY}}(\omega) = f_a(\omega) f_{C_X}(\omega)$$

Cross spectrum

$$f_{XY}(\omega) = f_{C_{XY}}(\omega) = \sum C_{XY}(r) e^{2\pi r \omega i}$$

Property:

$$f_{XY}(\omega) = f_{YX}(-\omega) = \overline{f_{YX}(\omega)}$$

because

$$C_{XY}(h) = C_{YX}(-h)$$

Conclusion: *Frequency response function*, A, of filter a is:

$$\begin{aligned} A(\omega) &= \sum a_r e^{2\pi r \omega i} \\ &= f_{XY}(\omega)/f_X(\omega) \end{aligned}$$

Estimate A from estimates of f_{XY} and f_X.

Estimate f_X by smoothing periodogram.

Estimating the cross-spectrum

Use cross-periodogram:

$$\hat{f}_{XY}(\omega) = \hat{Y}(\omega)\overline{\hat{X}(\omega)}$$

where $\hat{Y}$ and $\hat{X}$ are discrete Fourier transforms.

Smooth to improve estimate.

Estimating the impulse response

Recall Fourier inversion

$$a_t = \int_0^1 A(\omega)e^{-2\pi t\omega i}d\omega$$

Replace A by estimate $\hat{A}$ to get $\hat{a}$.

Gain is sometimes used for $|A|$.

Write

$$A(\omega) = |A(\omega)|e^{i\theta(\omega)}$$

for some function θ with

$$-\pi < \theta(\omega) \le \pi$$

Call θ the *phase shift* of the filter.

MSE of optimal filter?

$$\mathrm{E}\left\{\left(Y_t - \sum_s a_s X_{t-s}\right)^2\right\}$$
$$= C_Y(0) - 2\sum a_s C_{XY}(s) + \sum a_r a_s C_X(r-s)$$

Recall Fourier inversion formulas like

$$C_Y(h) = \int f_Y(\omega) e^{-2\pi h\omega i} d\omega$$

Put $h = 0$ get

$$C_Y(0) = \int f_Y(\omega) d\omega$$

Notice

$$\sum a_s C_{XY}(s) = \sum a_s C_{YX}(0-s)$$

But

$$f_{a * C_{YX}} = f_a f_{C_{YX}} = f_a f_{YX}$$

so by Fourier inversion

$$\sum a_s C_{YX}(0-s) = \int f_a(\omega) f_{YX}(\omega) d\omega$$

Also

$$\sum a_r a_s C_X(r-s) = \sum_r a_r \int f_{a*C_X}(\omega) e^{-2\pi r\omega i} d\omega$$
$$= \int f_a(\omega) f_X(\omega) f_a(-\omega) d\omega$$

But f_a is just A and so

$$\sum a_r a_s C_X(r-s) = \int f_X(\omega) |A(\omega)|^2 d\omega$$

Assemble:

$$MSE = \int \{f_Y(\omega) - 2A(\omega) f_{YX}(\omega)$$
$$+ f_X(\omega)|A(\omega)|^2\} d\omega$$

Factor out f_Y, recall $A = f_{XY}/f_X$:

$$MSE = \int f_Y(\omega) \left\{ 1 - 2\frac{|f_{XY}(\omega)|^2}{f_X(\omega) f_Y(\omega)} \right.$$
$$\left. + \frac{|f_{XY}(\omega)|^2}{f_X(\omega) f_Y(\omega)} \right\} d\omega$$

Definition: Coherence between X and Y at frequency ω is

$$\gamma_{XY}(\omega) = \frac{|f_{XY}(\omega)|}{\sqrt{f_X(\omega) f_Y(\omega)}}$$

So

$$MSE = \int f_Y(\omega) \left\{ 1 - \gamma^2_{XY}(\omega) \right\} d\omega$$

Interpretation: γ is a correlation coefficient.

Note: suppose we use raw periodgrams to estimate cross-spectrum:

Then

$$\hat{\gamma}_{XY}(\omega) = \frac{|\hat{Y}(\omega)\overline{\hat{X}(\omega)}|}{|\hat{Y}(\omega)| \cdot |\hat{X}(\omega)|} = 1$$

Conclusion: must smooth to get useful estimates of γ.

A spectral analysis example

Chandler wobble: Measure direction earth's axis of ratation points – it wobbles.

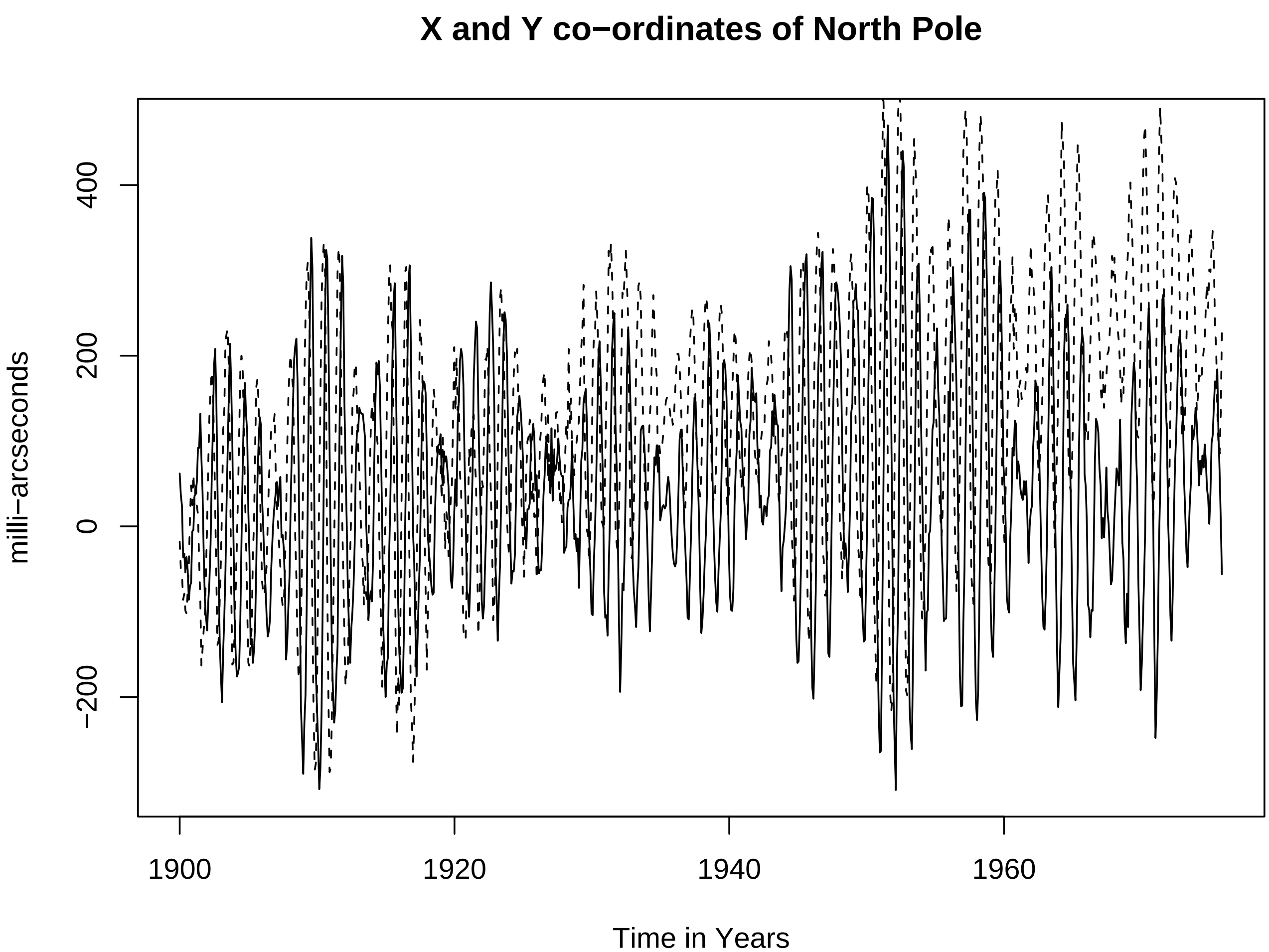
Data set 1: Monthly measurements of direction of North Pole.

Two co-ordinates: X and Y.

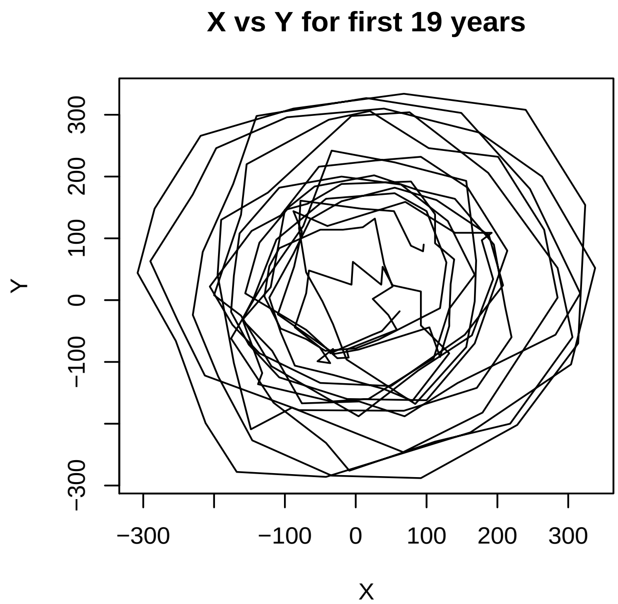
X direction toward Greenwich along Greenwich meridian.

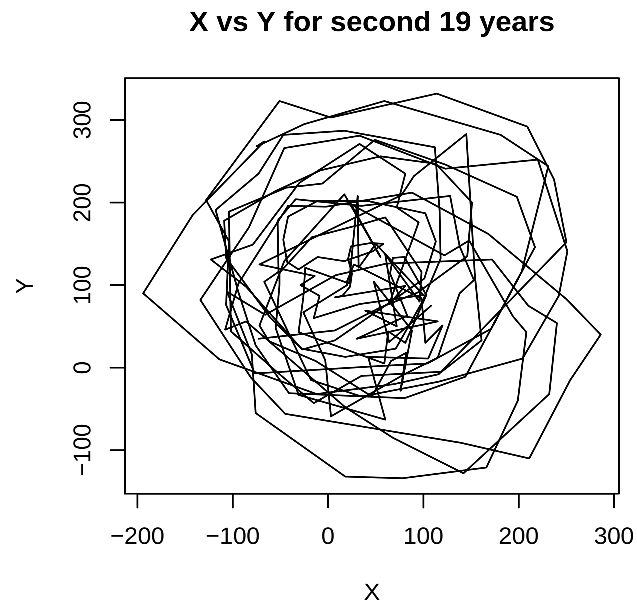
Study X and Y motions of pole together.

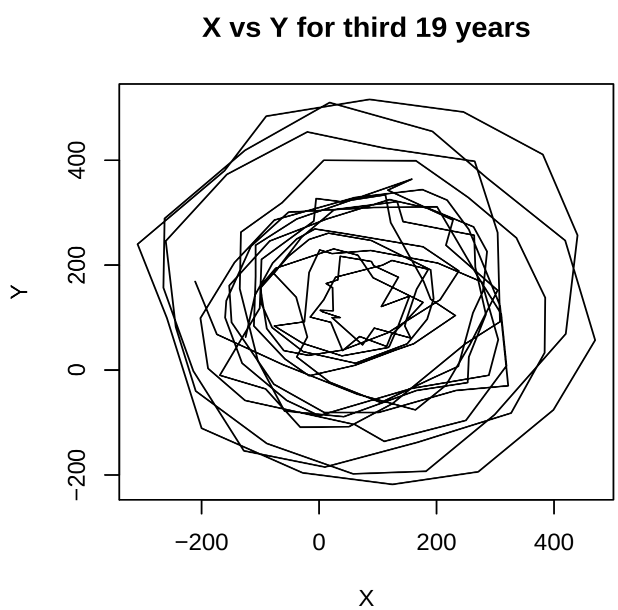
Plot of X and Y together:

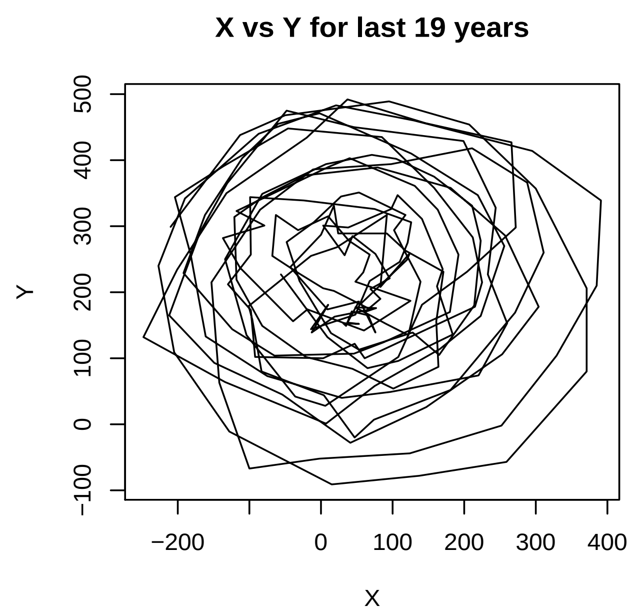


Parametric plots of X and Y:

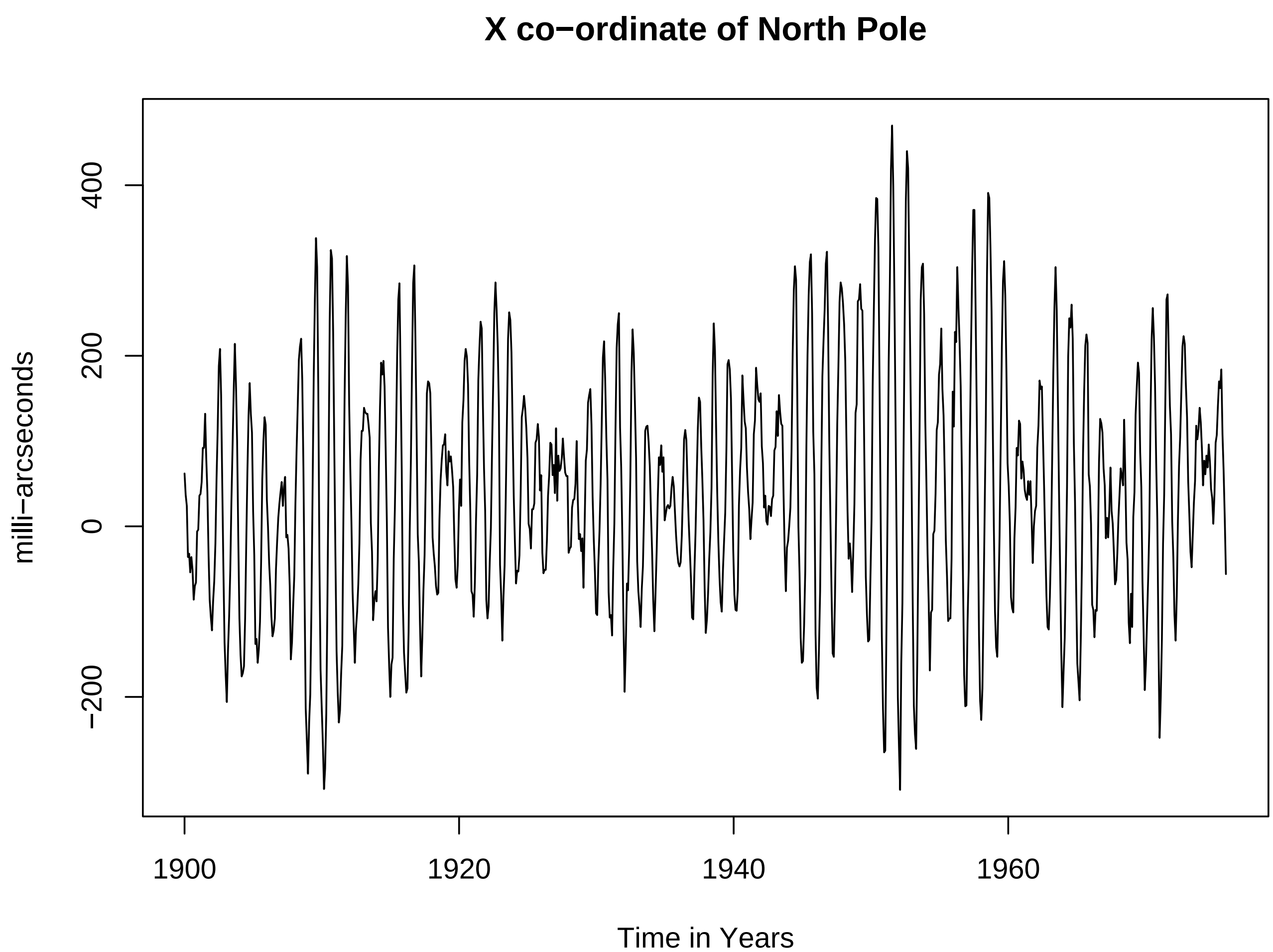






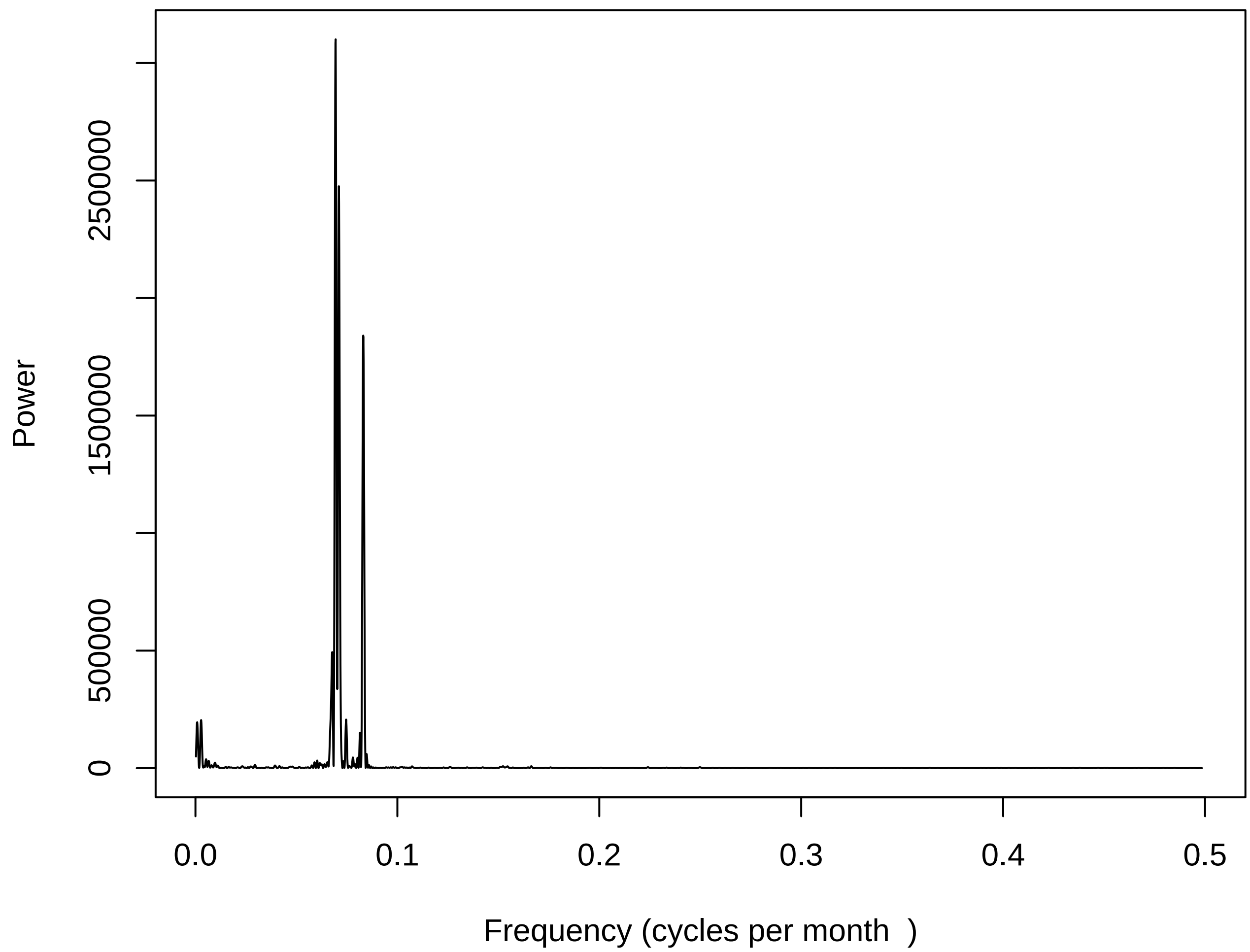


Re-examine plot of X:

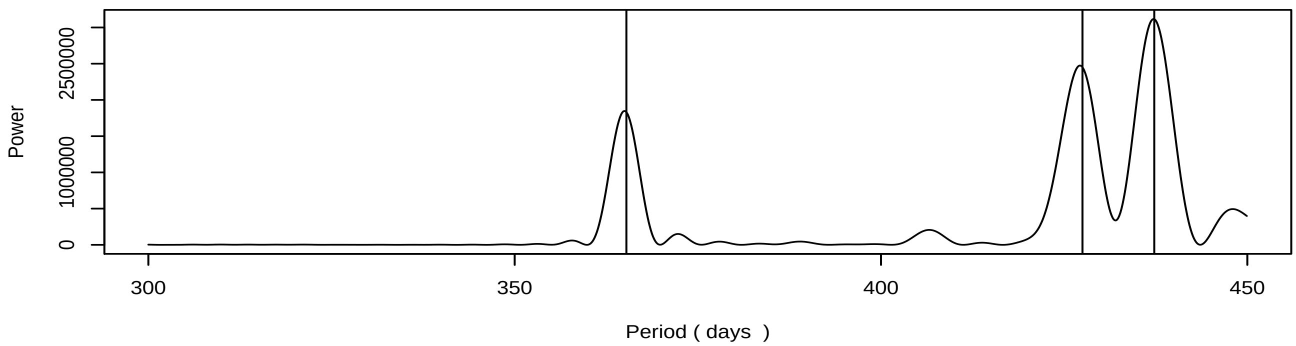


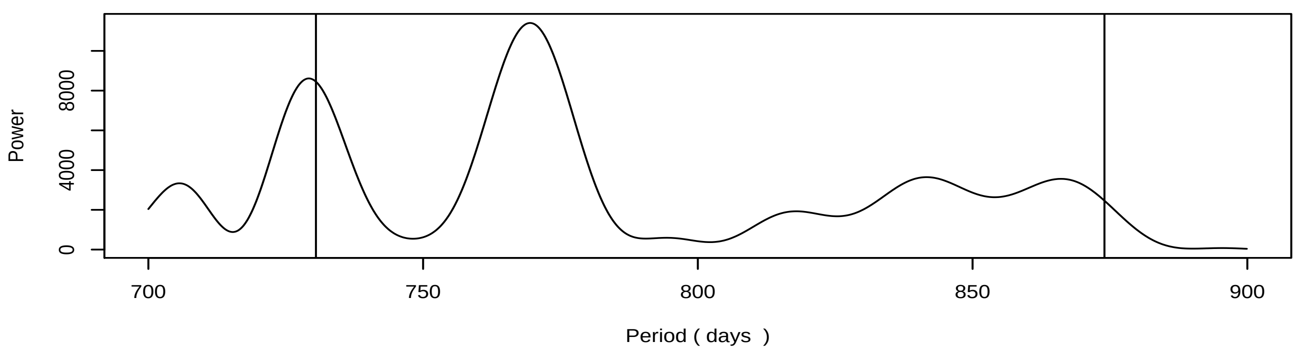
Clear cycles, what else, what periods?

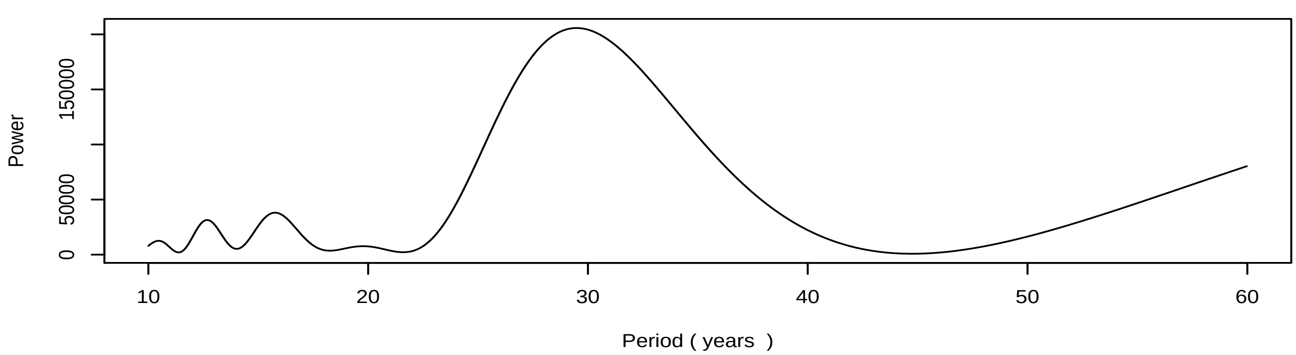
Look at spectrum of X:



Notice lots of spikes – interpret by plotting against periods over various ranges.


Power
0
1000000
2500000
300
350
400
450
Period (days)


Power
0
4000
8000
700
750
800
850
900
Period (days)


Power
0
50000
150000
10
20
30
40
50
60
Period (years)

Peak at 365.25 days: external forcing due to seasons. No surprise.

Secondary peak at $730.5 = 2 \times 365.25$. Overtone due to imperfect annual sinusoid.

Peaks at 427, 433 days, roughly: Chandler wobble.

Peak at 30 Years?

Compare amplitudes – smaller in middle plot.

Modelling possibilities:

1) Fit multiviate Autogregression:

Let

$$\mathbf{Z}_t = \left[\begin{array}{c} X_t \\ Y_t \end{array} \right]$$

and fit models like

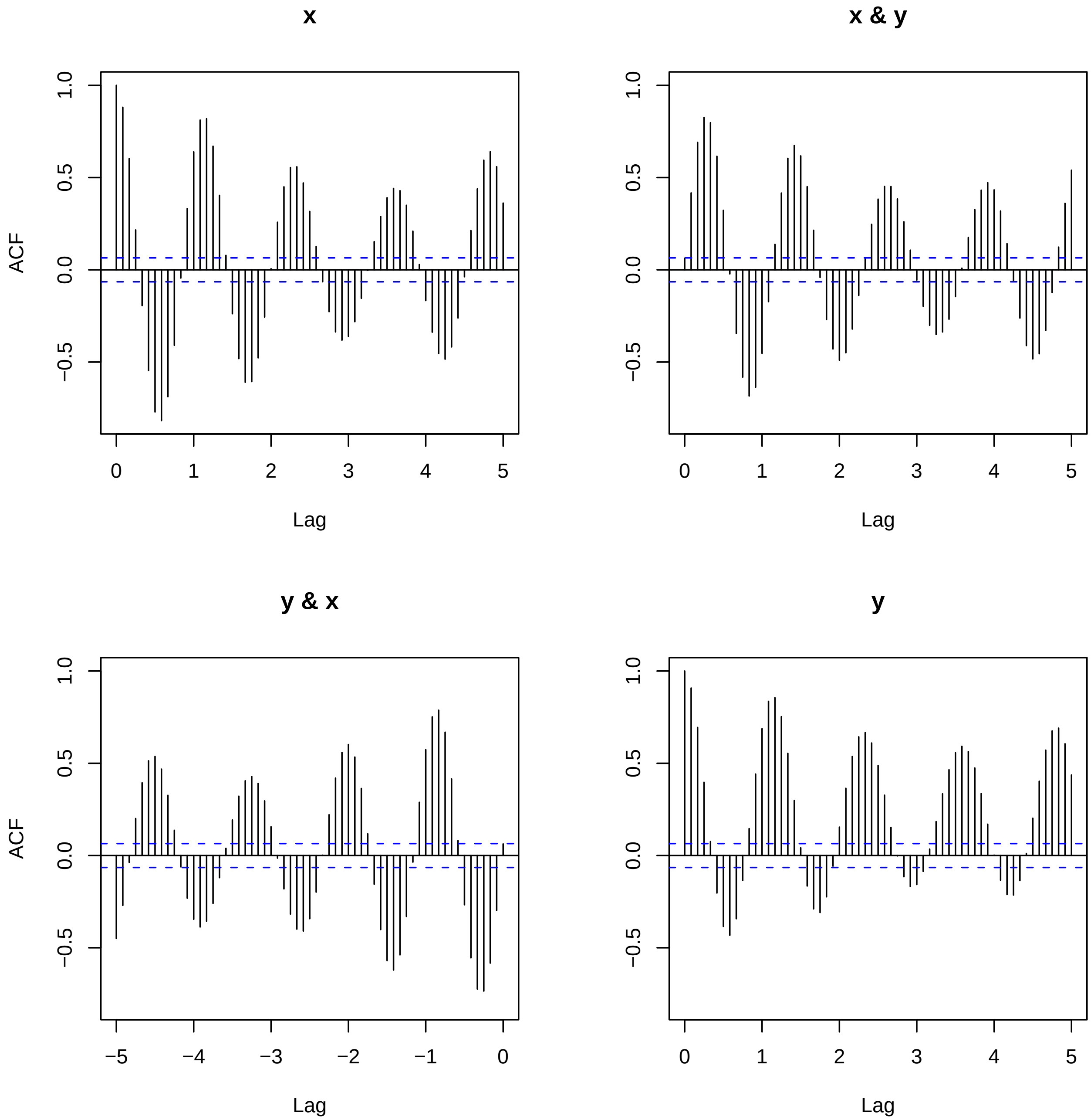
$$\mathbf{Z_t} - \mu = \sum_{j=1}^{p} \mathbf{A}_s \left(\mathbf{Z}_{t-s} - \mu \right) + \epsilon_t$$

where now $\mu, \mathbf{Z}_t$ and ϵ_t are vectors and each $\mathbf{A}_s$ is a matrix. General theory exists.

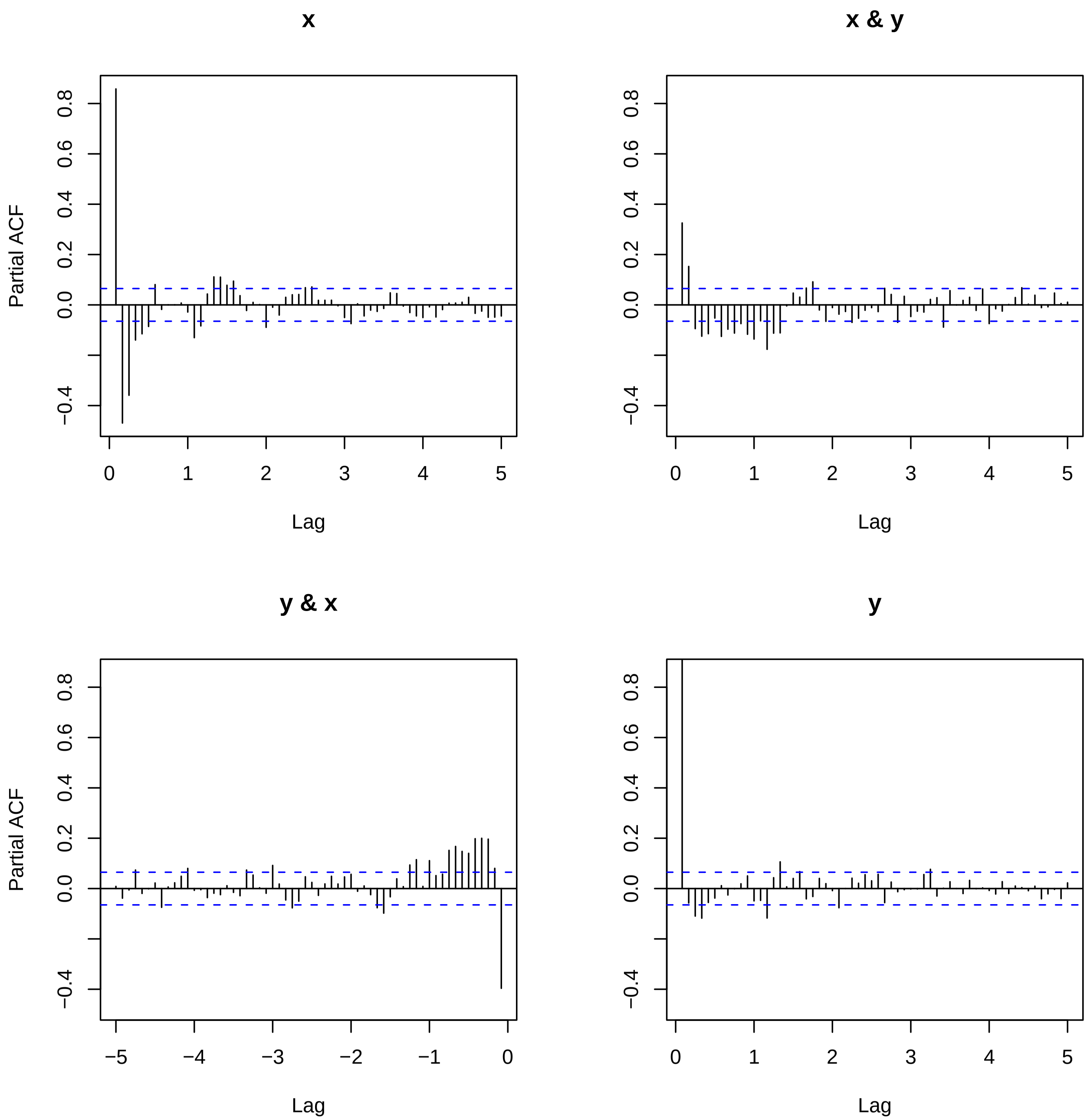
Fitted to Chandler in `R` picks AR(21). (Warning, not full maximum likelihood.)

2) Transfer function modelling. Try to predict say Y from X. (Done as an example – not well motivated scientifically.)

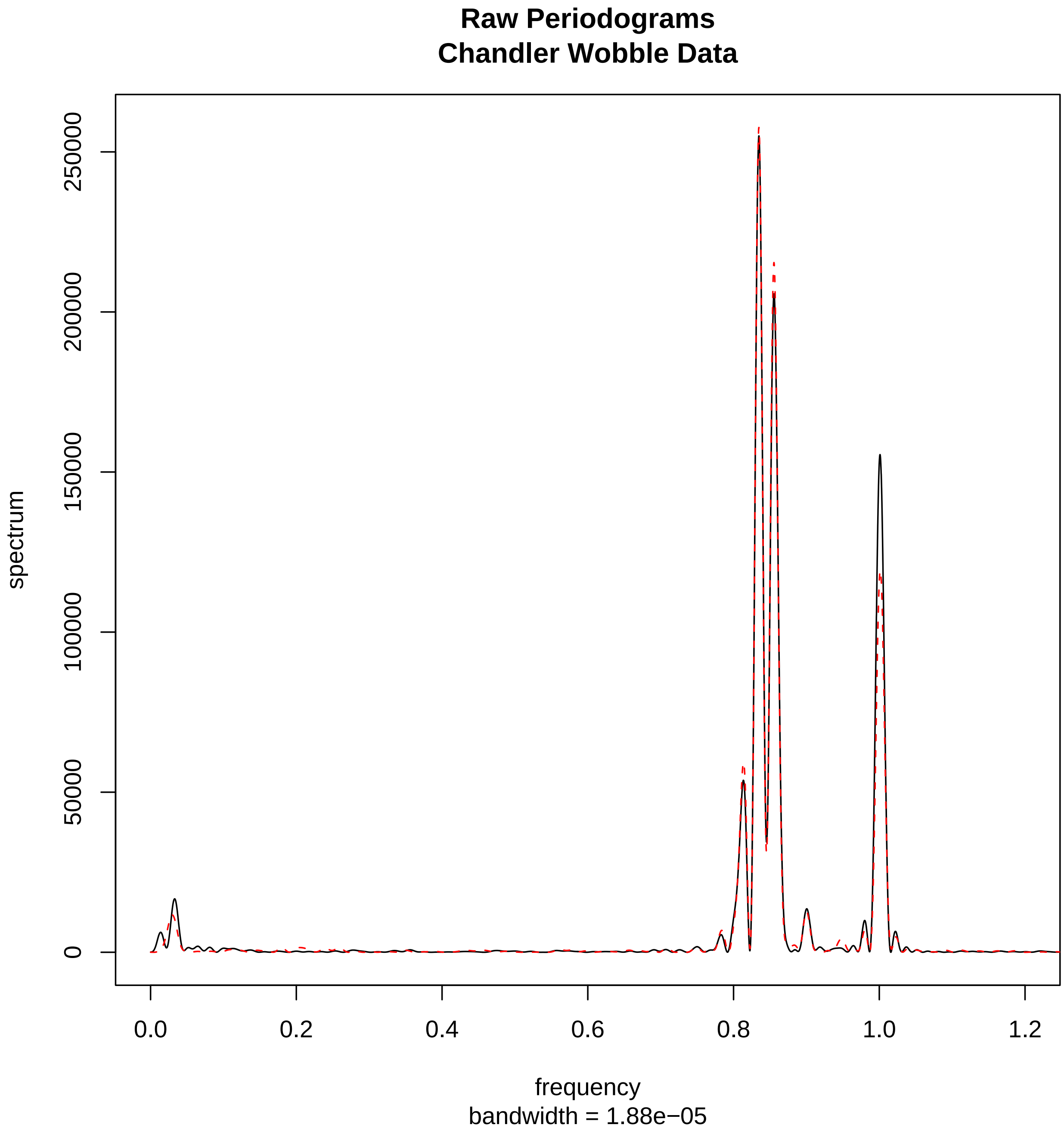
Plot the ACF:



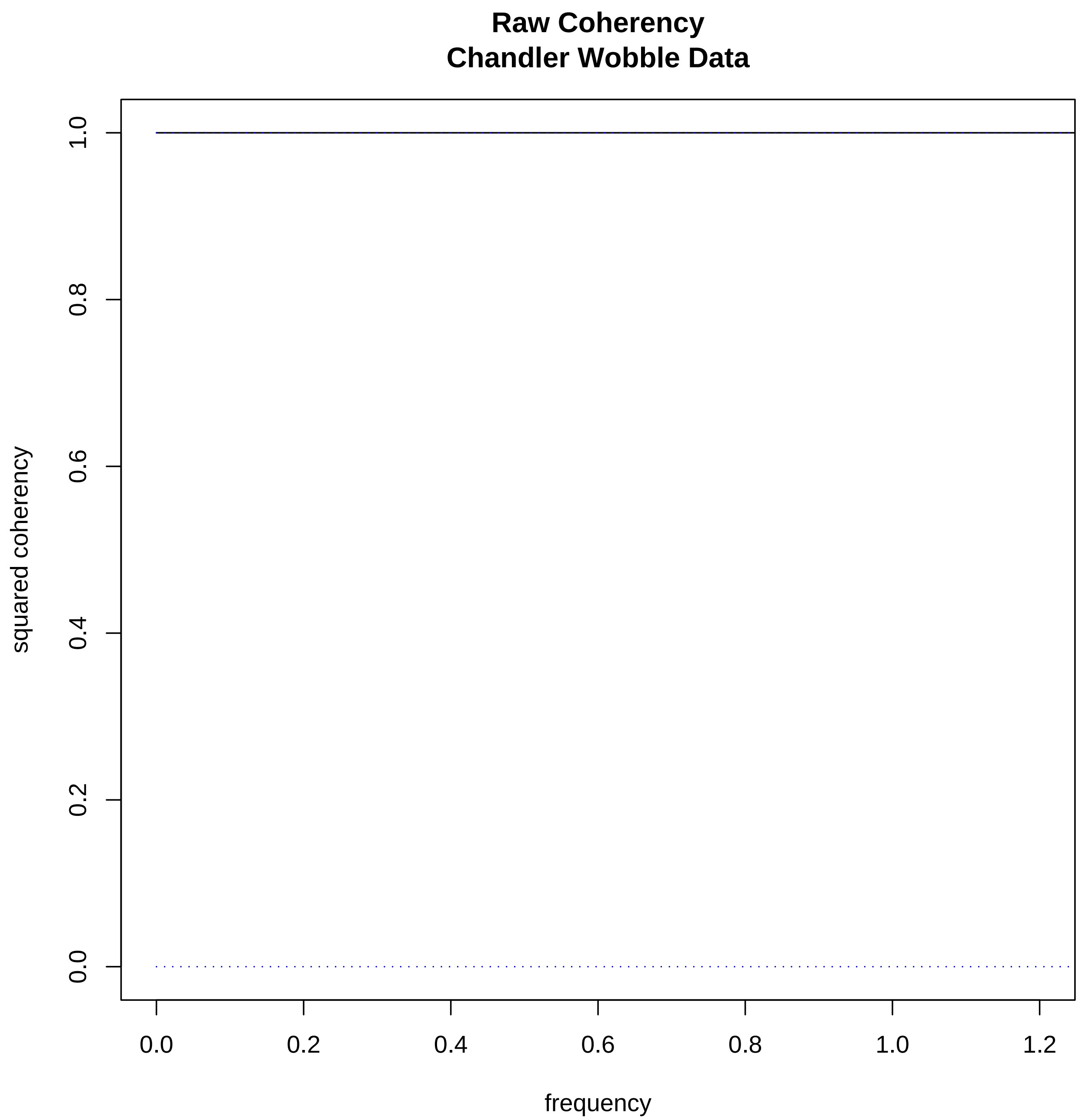
And the PACF



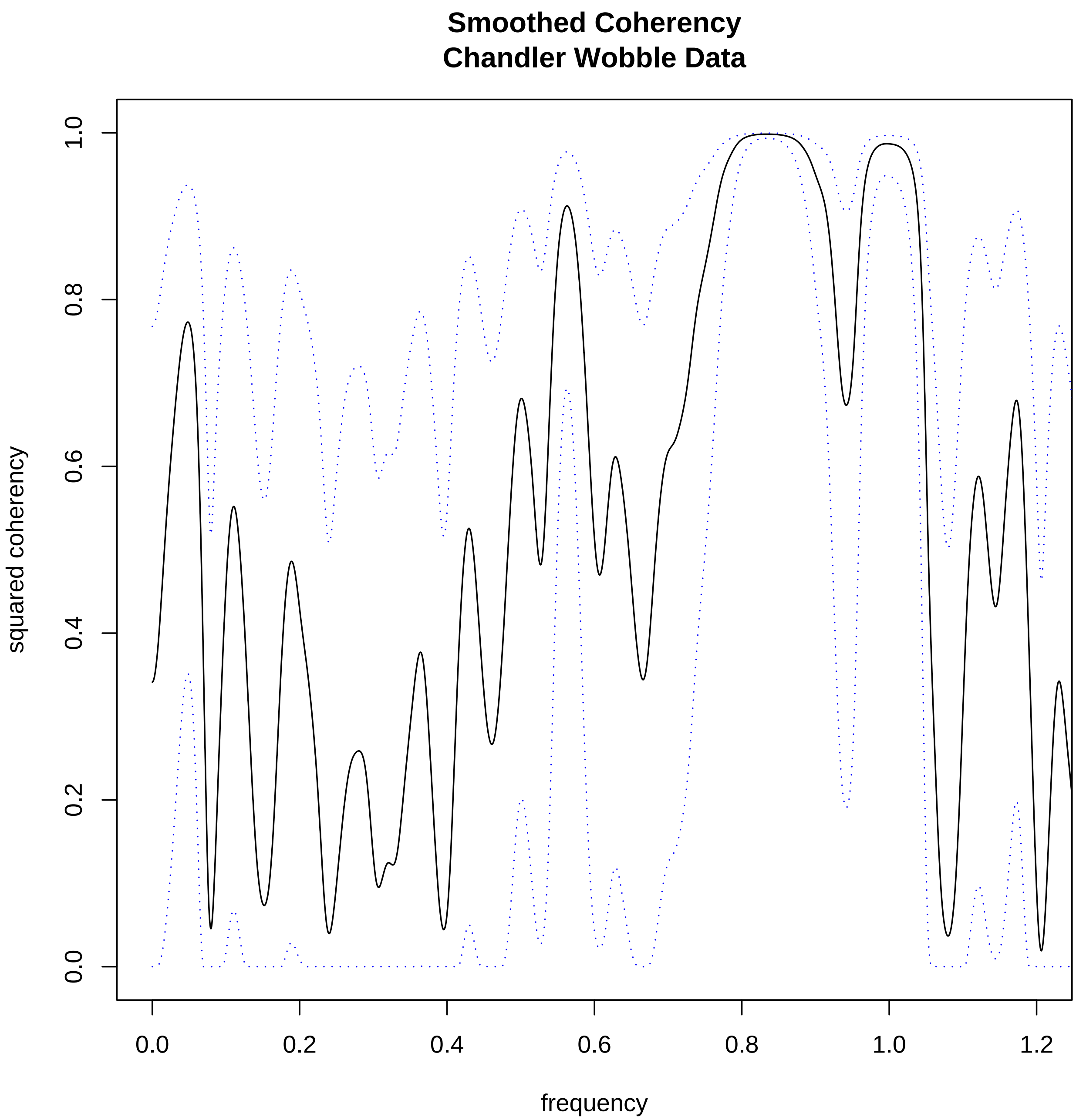
Two periodograms superimposed



Unsmoothed coherency



Smoothed coherency



```
plot(Chandler)
acf(Chandler,lag.max=60)
acf(Chandler,lag.max=60,type="partial")
ar.fit <- ar(Chandler)
plot(ar.fit$resid)
spectrum(Chandler,
    main="Raw Periodograms\nChandler Wobble Data",
            log="no",pad=200,xlim=c(0,1.2))
spectrum(Chandler,
    main="Raw Coherency\nChandler Wobble Data",
    log="no",pad=200,plot.type="coherency",xlim=c(0,1.2))
spectrum(Chandler,
    main="Raw Phase\nChandler Wobble Data",
    log="no",pad=200,plot.type="phase",xlim=c(0,1.2))
spectrum(Chandler,
    main="Smoothed Coherency\nChandler Wobble Data",
    log="no",pad=200,plot.type="coherency",xlim=c(0,1.2),
    spans=c(500,500,500))
spectrum(Chandler,
    main="Smoothed Phase\nChandler Wobble Data",
    log="no",pad=200,plot.type="phase",xlim=c(0,1.2),
    spans=c(500,500,500))
```

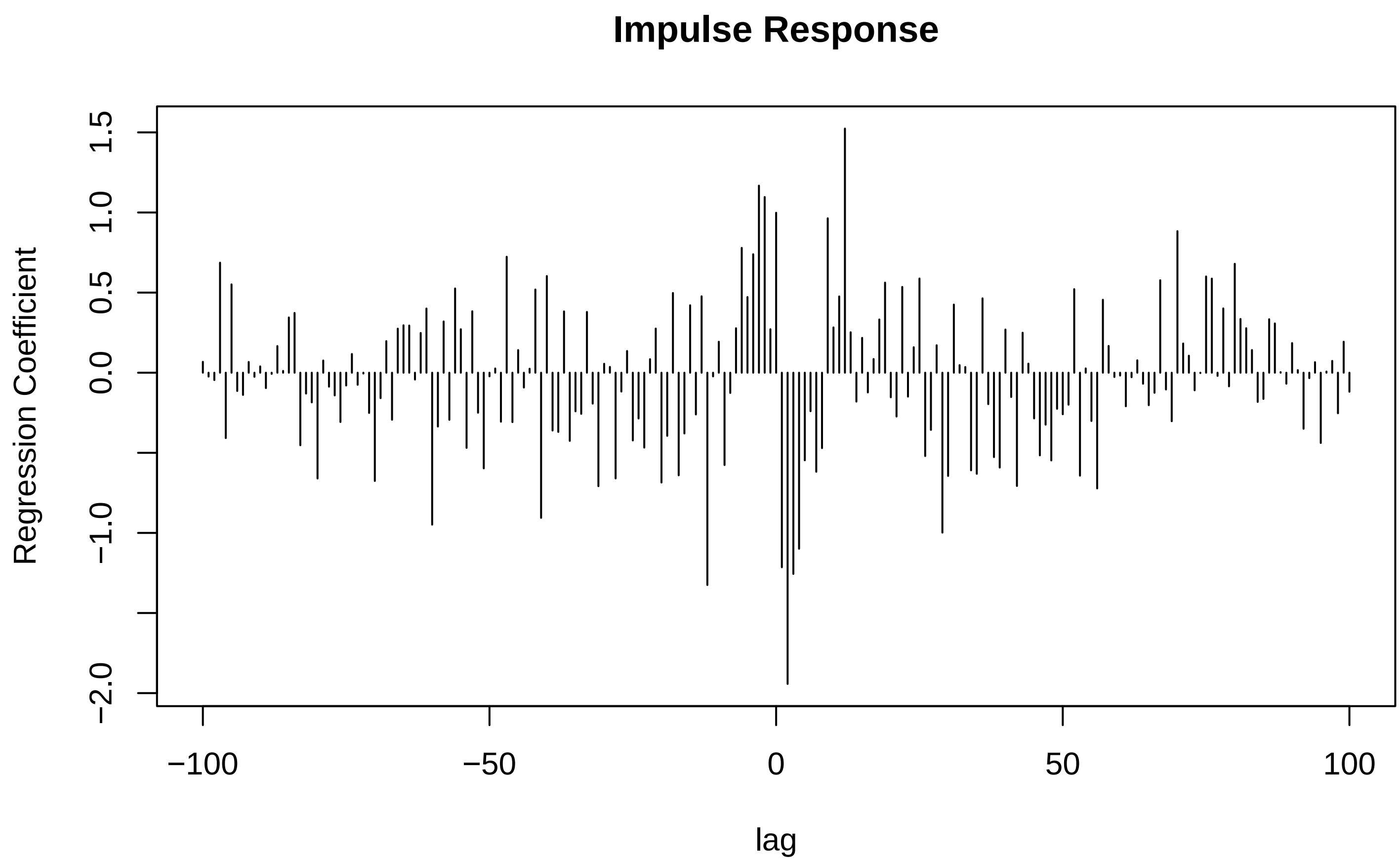
```
plot(Chandler)
spec = spectrum(Chandler,plot=FALSE,
            log="no",pad=200,spans=c(500,500,500))
cross.mod = sqrt(spec$coh*spec$spec[,1]*spec$spec[,2])
delf = mean(diff(spec$freq))/12
a <- rep(0,101)
for( i in 1:101){
 a[i] = 2*sum(cross.mod*cos(spec$phase-2*pi*(i-1)
                 *spec$freq/12)/spec$spec[,1])*delf
}
b <- rep(0,101)
for( i in 1:101){
 b[i] = 2*sum(cross.mod*cos(spec$phase+2*pi*(i-1)
                 *spec$freq/12)/spec$spec[,1])*delf
}
```

```
> ar.fit
$ar
, , 1                   , , 2                  , , 3
        x       y       x       y                x       y
x  0.7649 0.06698  x  0.05891 0.1308 x -0.134613 0.04038
y -0.2006 0.75669  y -0.10236 0.1029 y -0.004118 0.02718

, , 4                           , , 5                , , 6
    x         y                x         y                 x        y
x 0.014903  0.002543 x -0.01575 -0.039453 x -0.11804  0.06597
y 0.008151 -0.027783 y  0.03842 -0.008386 y  0.01383 -0.03717

, , 7                        , , 8                  , , 9
         x        y         x        y         x         y
x 0.082192 -0.04957 x -0.01096 -0.03097 x 0.01807 -0.051767
y 0.003160  0.05850 y  0.03817 -0.00669 y 0.10135 -0.002142

, , 10                        , , 11                   , , 12
          x         y         x        y        x        y
x  0.018404 -0.005875 x  0.0599131 -0.02887 x 0.1332 -0.09216
y -0.002739  0.010477 y -0.0003785  0.12447 y 0.1458  0.02205

, , 13                    , , 14                    , , 15
         x       y         x        y         x         y
x  0.03053 0.08509 x -0.052735 -0.05536 x -0.02703  0.004704
y -0.07984 0.07215 y  0.006195 -0.13834 y  0.05474 -0.033375

, , 16                    , , 17                    , , 18
         x        y      x        y          x        y
x  0.02273 -0.08411 x 0.03268 -0.02253 x -0.02145  0.02733
y -0.04357  0.08900 y 0.04345 -0.04299 y  0.04306 -0.02300

, , 19                  , , 20                    ,, 21
         x        y        x        y          x        y
x  0.03961 -0.02551 x  0.03864 -0.00362 x -0.02233  0.09161
y -0.02263  0.09121 y -0.05178 -0.01938 y -0.02435 -0.03170
```

Estimates of residual variance covariance and correlation:

```
$var.pred
       x      y
x 856.05 -15.84
y -15.84 837.45
> sqrt(diag(ar.fit$var.pred))
       x        y
29.25832 28.93881
> ar.fit$var.pred[1,2]/prod(sqrt(diag(ar.fit$var.pred)))
[1] -0.01870605
```

Finally, the estimated impulse response function:



Not enough smoothing – but too much for spikes.

So: not a good fitting technique for this sort of data.