**Multiple time series**

Our past work has focused primarily one variable for a time series. We examined ways to estimate dependence over time and use this dependence to construct statistical models. Now, we will extend this to multiple variables for a time series.

Cross-covariance function and Cross-correlation function

Measures like an ACF for one variable at a time will still be of interest. We can now also look at similar measures for two variables at a time. This can be especially helpful when one variable could be used to help predict the other.

Suppose there are two series denoted by xt and yt for t = 1, …, n. The cross-covariance function is

γxy(s,t) = Cov(xs, yt)

= E[(xs – μxs)(yt – μyt)]

= E(xsyt) – μxsμyt

where μxs = E(xs) and μyt = E(yt)

The cross-correlation function is



where s and t denote two time points and the x and y subscripts help denote the particular series.

Stationarity

Stationarity can also be examined when two time series are of interest. To have two series, xt and yt, be jointly stationary in a weak manner:

* + - Both time series must have constant mean
    - Both autocovariance functions must depend only on the lag difference
    - The cross-covariance must depend only on the lag difference.

If two series are jointly stationary, then the following notation can be used:

γxy(h) = E[(xt+h – μx)(yt – μy)]



Notes:

* γxy(h) = γyx(-h)
*  is not necessarily equal to  (usually will be different).
*  is not necessarily equal to  (usually will be different).

Example: This is an example from Shumway and Stoffer’s textbook.

Let xt = wt + wt-1 and yt = wt - wt-1 where wt ~ ind. (0,) for t = 1, …, n. This example was presented earlier when xt and yt were shown to be weakly stationary. Now, we are going to show xt and yt are weakly stationary in a joint manner.

γxy(s,t) = E[(xs - μxs)(yt - μyt)] = E[xsyt] because μxt = μyt = 0

Then γxy(s,t) = E[(ws + ws-1)(wt - wt-1)]

= E[wswt + ws-1wt - wswt-1 - ws-1wt-1]

If s = t, then

γxy(t,t) = E[wtwt + wt-1wt – wtwt-1 – wt-1wt-1]

= E[] – E[]

= Var(wt) + E[wt]2 – Var(wt-1) – E[wt-1]2

=  + 02 –  – 02

= 0

If s = t - 1, then

γxy(t-1,t) = E[wt-1wt + wt-2wt - wt-1wt-1 - wt-2wt-1]

= -E[]

= -

If s = t + 1, then

γxy(t+1,t) = E[wt+1wt + wtwt - wt+1wt-1 - wtwt-1]

= E[]

= 

If |s - t| > 1, then γxy(s,t) = 0.

Therefore, 

Thus, xt and yt are jointly stationary in a weak manner.

Example:  where wt ~ independent (0, ) for t = 1,…, n and |ϕ1|<1

This example was presented earlier when xt and wt were shown to be weakly stationary. Now, we are going to show xt and wt are jointly stationary.

Note that the sum of an infinite series is



The time series can be rewritten as



The cross-covariance function is

γxw(h) = E[(xt+h – μx,t+h)(wt – μwt)]

Remember that μwt = E(wt) = 0 and μx,t+h =E(xt+h) = 0. Then

γxw(h) = E[xt+hwt]

= E[()wt]

= 

Because the wt’s are independent with mean 0, we only need to be concerned about when the subscripts of the wt pairs match.

γxw(0)

After recording the video: I should have started with γxw(s,t) = E[(xs – μxs)(wt – μwt)] rather than γxw(h) = E[(xt+h – μx,t+h)(wt – μwt)] and then proceeded to looking at cases like s = t, s = t+1, s = t – 1, … like in the previous example. In the end, we do have joint stationarity so using γxw(h) leads to the correct result.



γxw(1)



γxw(-1)



In general,

γxw(h) =  for h ≥ 0 and γxw(h) = 0 for h < 0.

The h < 0 part should make intuitive sense because of the model. For example, xt-1 comes from wt-1, wt-2, …., but not from wt.

Therefore, xt and wt are jointly stationary.

After recording the video: A small adjustment to the sum below would be needed for . When h = 1 for , we would have y0 included, which is not observed. Thus, one could start the sum at 2 and go up to n to have the equality be true.

Estimation

The sample cross-covariance function is



Note that ; however,  is not necessarily equal to  and  is not necessarily equal to .

The sample cross-correlation function is



The sampling distribution for  is approximately normal with mean 0 and standard deviation of  if the sample size is large and at least one of the series is white noise.

For a hypothesis test, we can check if  is within the bounds of 0 or not. If it is not, then there is sufficient evidence to conclude that ρxy(h) ≠ 0.

Example: Simple CCF example

(simple\_CCF\_exampleV2.xlsx, simple\_CCF\_exampleV2.R)

The Excel file shows how some of the “by-hand” calculations of the cross-covariance function can be done. Below is part of the resulting spreadsheet.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  | h=0 | h=1 |
| t | xt | yt |  |  |  |  |
| 1 | 1 | 2 |  |  | 6.667 | 4.000 |
| 2 | 2 | 3 |  |  | 2.500 | 0.833 |
| 3 | 3 | 5 |  |  | -0.167 | 0.167 |
| 4 | 4 | 6 |  |  | 0.667 | 2.000 |
| 5 | 5 | 8 |  |  | 5.000 | 8.333 |
| 6 | 6 | 4 |  |  | -1.667 | No x7 |
| Mean | 3.5 | 5.5 |  | Sum: | 13.00 | 15.33 |

The estimated cross-covariance function is

.

Then





The cross-correlation function is .

Then





where  and  were found in R.

Below is the R code and output:

> x <- c(1, 2, 3, 4, 5, 6)

> y <- c(2, 3, 5, 6, 8, 4)

> gamma.x <- acf(x = x, type = "covariance", plot = FALSE)

> gamma.x

Autocovariances of series 'x', by lag

0 1 2 3 4 5

2.917 1.458 0.167 -0.792 -1.250 -1.042

> gamma.y <- acf(x = y, type = "covariance", plot = FALSE)

> gamma.y

Autocovariances of series 'y', by lag

0 1 2 3 4 5

3.889 1.093 -0.481 -1.556 -1.296 0.296

> # Covariance - Match with Excel file

> x.y.cov <- acf(x = cbind(x,y), type = "covariance")

> x.y.cov

Autocovariances of series ‘cbind(x, y)’, by lag

, , x

x y

2.9166667 ( 0) 2.1666667 ( 0)

1.4583333 ( 1) 0.6111111 (-1)

0.1666667 ( 2) -0.8055556 (-2)

-0.7916667 ( 3) -1.3333333 (-3)

-1.2500000 ( 4) -1.2222222 (-4)



γxy(h) for h ≤ 0

, , y

x y

2.1666667 ( 0) 3.8888889 ( 0)

2.5555556 ( 1) 1.0925926 ( 1)

0.7222222 ( 2) -0.4814815 ( 2)

-0.5000000 ( 3) -1.5555556 ( 3)

-1.3611111 ( 4) -1.2962963 ( 4)

γxy(h) for h ≥ 0

γyx(-h) for h ≥ 0



<PLOT EXCLUDED FROM OUTPUT>

The output labeling can be confusing. One way to remember what is being displayed is to always think of “x” as coming first with respect to our notation.

> # Correlation

> x.y.acf <- acf(x = cbind(x,y), type = "correlation")

> x.y.acf

Autocorrelations of series ‘cbind(x, y)’, by lag

, , x

x y

1.000 ( 0) 0.643 ( 0)

0.500 ( 1) 0.181 (-1)

0.057 ( 2) -0.239 (-2)

-0.271 ( 3) -0.396 (-3)

-0.429 ( 4) -0.363 (-4)

ρxy(h) for h ≤ 0



, , y

x y

0.643 ( 0) 1.000 ( 0)

0.759 ( 1) 0.281 ( 1)

0.214 ( 2) -0.124 ( 2)

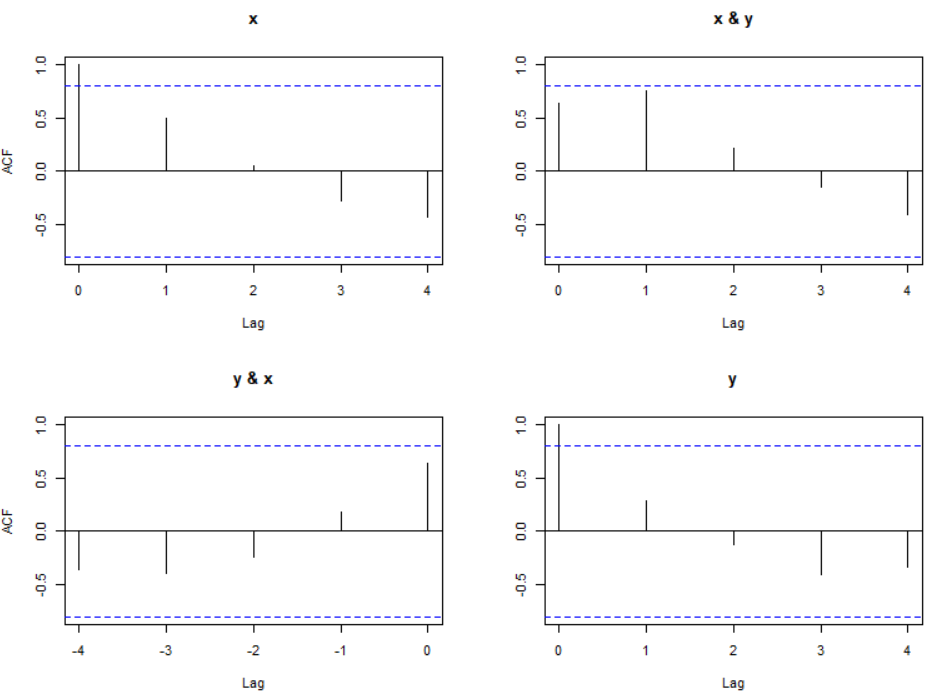
-0.148 ( 3) -0.400 ( 3)

-0.404 ( 4) -0.333 ( 4)

ρxy(h) for h ≥ 0

ρyx(-h) for h ≥ 0







The (1,2) and (2,1) plots are in the opposite order from what was given in the displayed output.

> x.y.ccf <- ccf(x = x, y = y, type = "correlation")

> x.y.ccf

Autocorrelations of series 'X', by lag

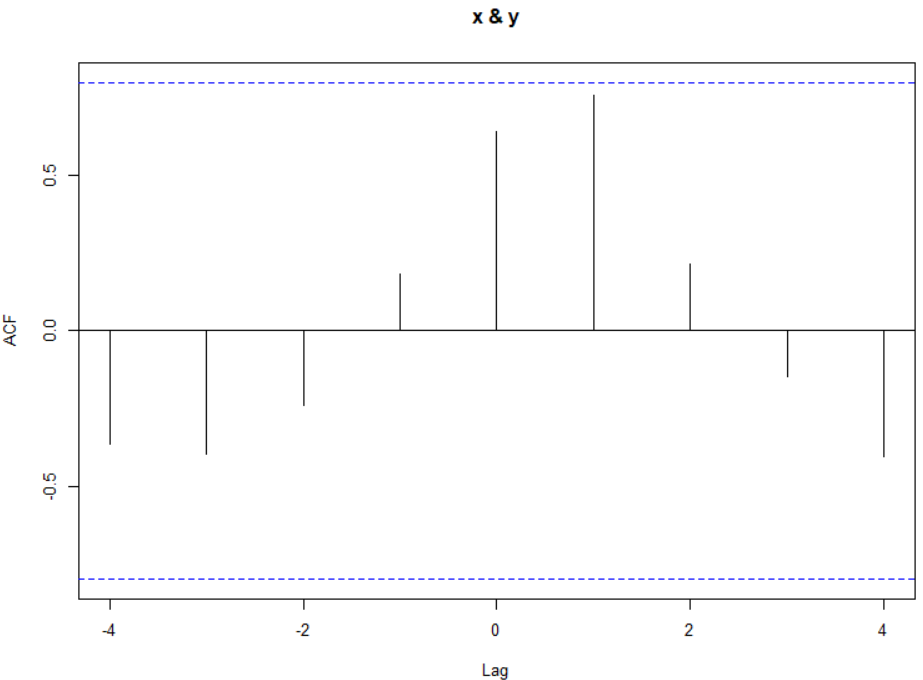
-4 -3 -2 -1 0 1 2 3

-0.363 -0.396 -0.239 0.181 0.643 0.759 0.214 -0.148

4

-0.404







Example: El Nino and fish population (ElNino.R)

This is an example from Shumway and Stoffer’s textbook involving two variables:

* Southern Oscillation Index (SOI; xt) – Measurement of air pressure in central Pacific Ocean (helps to determine if El Nino effect is present)
* Recruitment (yt) – Number of new fish

There are 453 months of observations. The data are available in the soi and rec objects of the authors’ package.

> library(package = "astsa")

> options(width = 60)

> class(soi)

[1] "ts"

> class(rec)

[1] "ts"

> window(x = soi, start = 1960, end = 1960.99)

Jan Feb Mar Apr May Jun Jul

1960 0.169 0.432 0.202 -0.366 -0.661 0.093 -0.716

Aug Sep Oct Nov Dec

1960 0.148 -0.093 0.279 0.432 -0.104

> window(x = rec, start = 1980, end = 1981)

Jan Feb Mar Apr May Jun

1980 99.46000 99.37000 99.51999 96.64000 89.55000 68.67000

1981 74.11000

Jul Aug Sep Oct Nov Dec

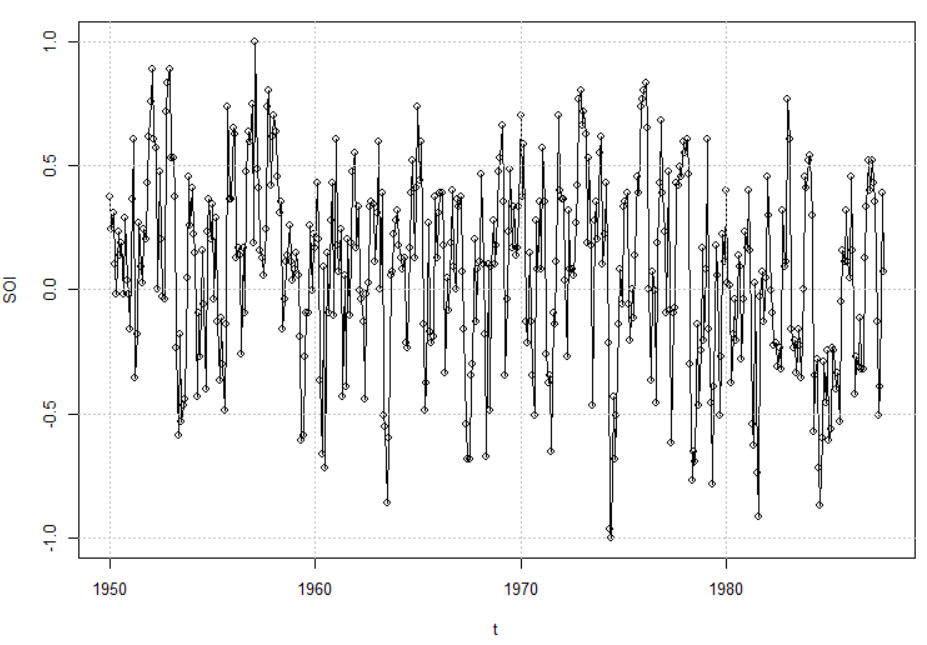
1980 65.02000 61.82000 76.92000 80.17000 77.48001 82.34000

1981

> plot(x = soi, ylab = "SOI", xlab = "t", type = "o")

> grid(col = "gray", lty = "dotted") # Puts grid lines in

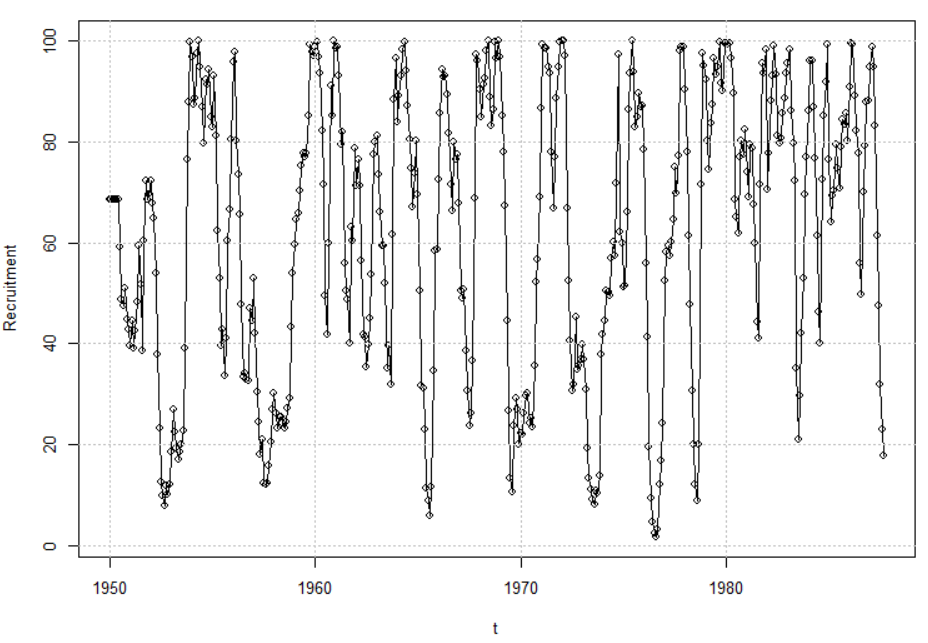
correct location



> plot(x = rec, ylab = "Recruitment", xlab = "t", type =

"o")

> grid(col = "gray", lty = "dotted")



> soi.rec.acf <- acf(x = cbind(soi,rec), type =

"correlation", lag.max = 50)

> soi.rec.acf

Autocorrelations of series 'cbind(soi, rec)', by lag

, , soi

soi rec

1.000 ( 0.0000) 0.025 ( 0.0000)

0.604 ( 0.0833) 0.011 (-0.0833)

0.374 ( 0.1667) -0.042 (-0.1667)

0.214 ( 0.2500) -0.146 (-0.2500)

0.050 ( 0.3333) -0.297 (-0.3333)

-0.107 ( 0.4167) -0.527 (-0.4167)



ρxy(h) for h ≤ 0

<OUTPUT EDITED>

, , rec

soi rec

0.025 ( 0.0000) 1.000 ( 0.0000)

-0.013 ( 0.0833) 0.922 ( 0.0833)

-0.086 ( 0.1667) 0.783 ( 0.1667)

-0.154 ( 0.2500) 0.627 ( 0.2500)

-0.228 ( 0.3333) 0.477 ( 0.3333)

-0.259 ( 0.4167) 0.355 ( 0.4167)

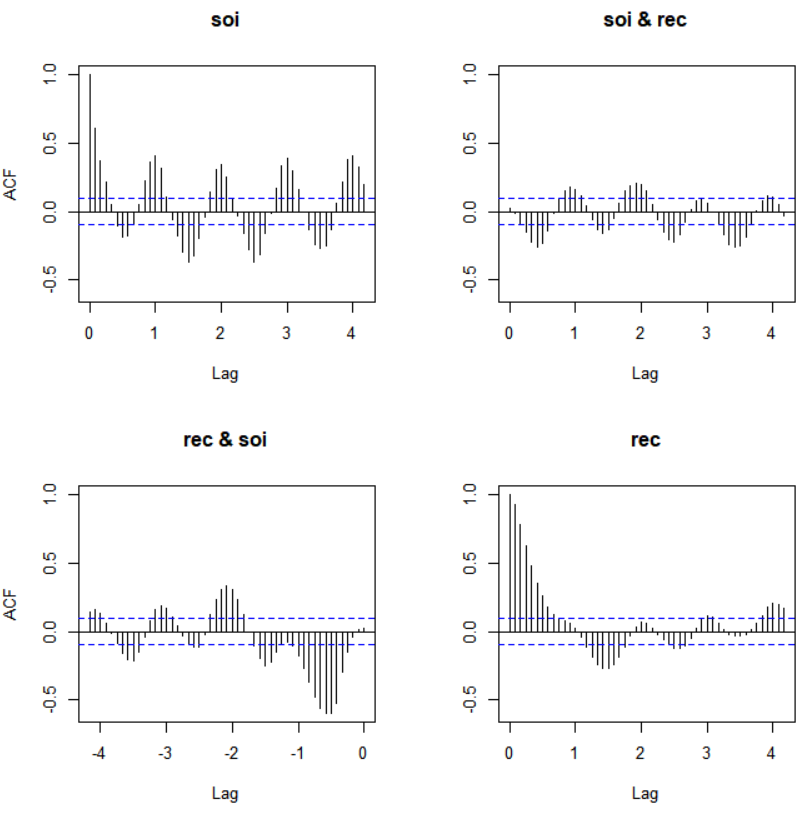


ρxy(h) for h ≥ 0

ρyx(-h) for h ≥ 0



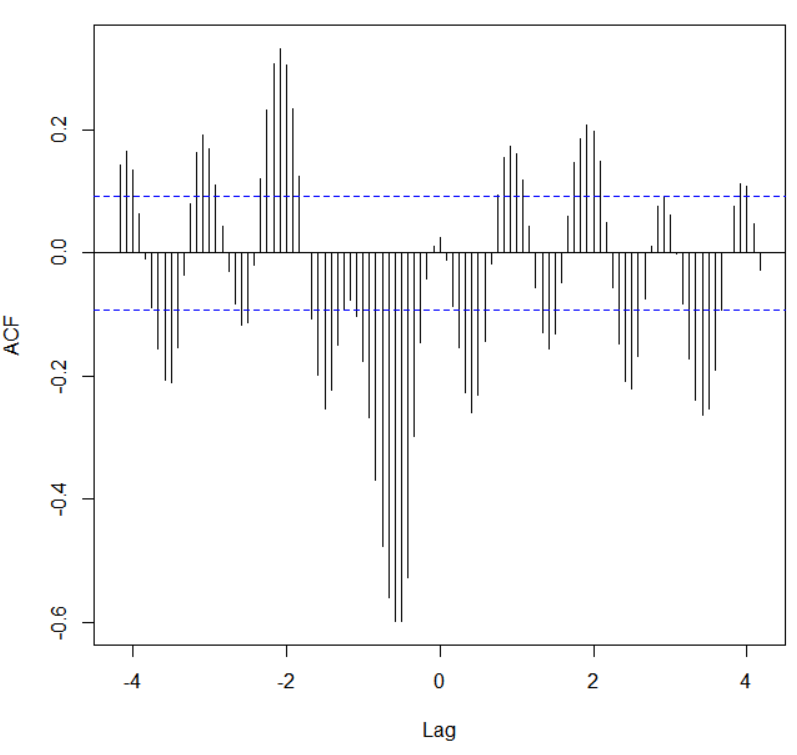
<OUTPUT EDITED>





The tick marks on the x-axis represent years. There are still 50 lags on the x-axis.

> ccf(x = soi, y = rec, type = "correlation", lag = 50)





Examine the strength of association for the two series individually and together.

**Vector time series**

Because many different time series often occur at the same time, it is useful to consider a vector of time series data.

Let  be a vector time series. Note that this could also be represented as a transpose:

**x**t = (xt1, xt2, …, xtp)′

A vector is represented as a bold letter. Note that xt1 represents the first time series variable at time t, …, xtp represents the pth time series variable at time t.

For the jointly stationary case,

* **μ** = E(**x**t) where **μ** = (μt1, μt2, …, μtp)′ is the mean vector
* **Γ**(h)= E[(**x**t+h - **μ**)(**x**t - **μ**)′] is the autocovariance matrix

The autocovariance matrix is similar to the covariance matrix discussed in other statistics courses. Elements of this matrix are





where γij(h) = E[(xt+h,i - μi)(xtj - μj)]

Notes:

* For example, γ12(h) = Cov(xt+h,1, xt,2)
* Remember that a covariance matrix is symmetric
* **Γ**(-h) = **Γ**(h)′ since γij(h) = γji(-h)

Sample autocovariance matrix:



where 

Note that 

We are indexing a series of random variables by one value: time. A series of random variables can also be indexed by more than only one index. For example, this can happen in spatial statistics.