# Day 2 exercise

## James Thorson

## 2025-05-20

### Arrow-and-lag interface for simultaneous and lagged effects

Agenda section E: Exercise Goal:

1. Understand arrow-and-lag notation

2. Fit simultaneous and lagged effects in a time-series model

Structure:

1. Students are to self-organize into groups of 2-3 people, and work in a group on step-1 for 20 minutes.

2. For Step-0, we will walk through code as a group;

3. For later steps, the instructor will then share code for that step on github, and groups will continue with step-2 for 20 minutes, then repeating for step-3.

### Step 0: Try fitting a linear model (20 minutes)

Try simulating data from a linear model:

$$x_i \sim \text{Normal}(2, \sigma_X^2) \quad y_i \sim \text{Normal}(1 + 0.5, \sigma_Y^2)$$

Where:

- $i \in 1, 2, \ldots, 100$, $x$ and $y$ are vectors with length 100

- $\sigma_X^2 = 1$ is the variance in the predictor variable $x$ and $\sigma_Y^2 = 1$ is the residual variance in the response variable $y$

- $E(x_i) = 2$ and $E(y_i) = 1 + 0.5x_i$

- $x_i$ Normal$(2, \sigma_X^2)$ indicates that $x_i$ is simulated from a normal distribution, which can be done in R using `rnorm( n = 100, mean = 2, sd = 1)`

```
# Simulate x and y
x = rnorm( n = 100, mean = 2, sd = 1 )
y = rnorm( n = 100, mean = 1 + 0.5*x, sd = 1 )
```

Then try fitting in dsem:

```r
#install.packages("dsem")
library(dsem)

# Define arrow-and-lag notation
time_term = "
# predictor -> response, lag, parameter_name, start_value
x -> y, 0, slope, 1
x <-> x, 0, sd_x, 1
y <-> y, 0, sd_y, 1
"

# format data into ts object
tsdata = ts(data.frame(x=x, y=y))

# fit in dsem
dsem_fit = dsem(
  tsdata = tsdata,
  sem = time_term
)
```

```
## List of estimated fixed and random effects:

##   Coefficient_name Number_of_coefficients   Type
## 1           beta_z                      3  Fixed
## 2             mu_j                      2 Random


## Running nlminb_loop #1


## Running newton_loop #1
```

You can then compare this with a linear model

```r
lm_fit = lm( y ~ 1 + x )

#
summary(lm_fit)
```

```
##
## Call:
## lm(formula = y ~ 1 + x)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -3.10006 -0.71600 -0.00444  0.64479  2.13122
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   0.5648     0.2354   2.400   0.0183 *
## x             0.7092     0.1126   6.297 8.59e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
## 
## Residual standard error: 1.041 on 98 degrees of freedom
## Multiple R-squared:  0.2881, Adjusted R-squared:  0.2808
## F-statistic: 39.66 on 1 and 98 DF,  p-value: 8.593e-09
```

```r
summary(dsem_fit)
```

```
##       path lag  name start parameter first second direction  Estimate   Std_Error
## 1  x -> y   0 slope     1         1     x      y         1 0.7092111 0.11204855
## 2 x <-> x   0  sd_x     1         2     x      x         2 0.9289072 0.06601439
## 3 y <-> y   0  sd_y     1         3     y      y         2 1.0356098 0.07359743
##     z_value       p_value
## 1  6.329499 2.459591e-10
## 2 14.071284 5.702457e-45
## 3 14.071277 5.703036e-45
```

## Step 1: Simulate data (20 minutes)

Please write an R script that simulates abundance at age $n_{(a,t)}$ for ages $a \in 1, 2, \ldots, 10$ and years $t \in 1, 2, \ldots, 40$ with the following simplified dynamics:

$$n_{a,t} = \begin{cases} \mu_R \times e^{\sigma_1 \epsilon_{a,t} - ma} & \text{if } a = 1 \,\&\, t = 1 \\ n_{a-1,t-1} \times e^{\sigma_2 \epsilon_{a,t} - m} & \text{otherwise} \end{cases}$$

Where process errors $\epsilon_{a,t}$ Normal$(0, \sigma^2)$, median age-0 recruitment $\mu_R = 10^9$ [individuals], recruitment variation $\sigma_1 = 0.6$ [dimensionless], demographic variation $\sigma_2 = 0.1$ [dimensionless], and natural mortality $m = 0.4$ [per year].

Now imagine that we can only sample abundance-at-age from a sampling gear with logistic selectivity:

$$s_a = \frac{1}{1 + e^{-\theta_2(a - \theta_1)}}$$

Where age at 50% selection $\theta_1 = 3$ [years] and the logit-slope $\theta_2 = 1$ [year^(-1)]. We then observe:

$$n^*_{a,t} = n_{a,t} s_a$$

## Step 2: Fit model using dsem (20 minutes)

Write out the arrow-and-lag notation whereby $n_{a,t}$ is caused by abundance $n_{a-1,t-1}$ the preceding age and year. This will presumably involve several one-headed arrows:

```r
X -> Y, 1, parameter
```

Where X is the predictor, Y the response, 1 represents a time-lag, and **parameter** is a parameter name (you can use the same name multiple times to force a single parameter value to be estimated and shared across lines)

Next, modify this arrow-and-lag notation to represent the assumption that exogenous variation at age1 (representing recruitment deviations) will be different than subsequent ages (representing demographic variation). This will presumably involve several two-headed arrows:

```
X <-> X, 0, parameter
```

where these always have a lag of zero.

## Step 3: Retrospective skill testing (20 minutes)

Finally, conduct a "leave-future-out" crossvalidation by dropping the final five years of simulated data. To do so, modify `tsdata` by replacing values (measurements) with NAs (representing missing data). This will ensure that the model still predicts those missing values.