Introduction to structural equation modeling and mixed models in R

Day 8

Oksana Buzhdygan

oksana.buzh@fu-berlin.de

01.03.2023

Day 8 – Part 1

Outline

- Introduction to Local Estimation in SEM

  - ✓ Global vs. Local Estimations. Piecewise SEM
  - ✓ Assessing Model Fit
  - ✓ Model Comparison
  - ✓ Categorical Data in Piecewise SEM

Day 8 – Part 1

Outline

- Introduction to Local Estimation in SEM

✓ **Global vs. Local Estimations. Piecewise SEM**

✓ Assessing Model Fit

✓ Model Comparison

✓ Categorical Data in Piecewise SEM

Global vs. Local Estimations

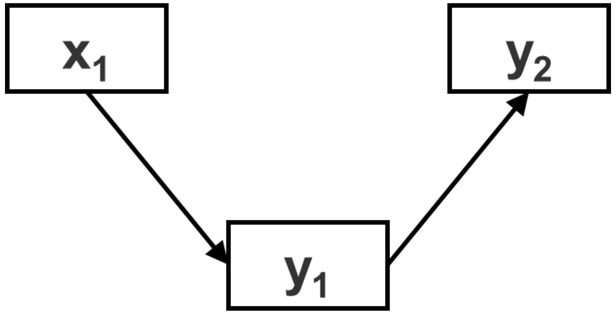
Two Paradigms for model estimation

Global Estimation

(*Covariance-Based Estimation*)

- reproduces a single variance-covariance matrix

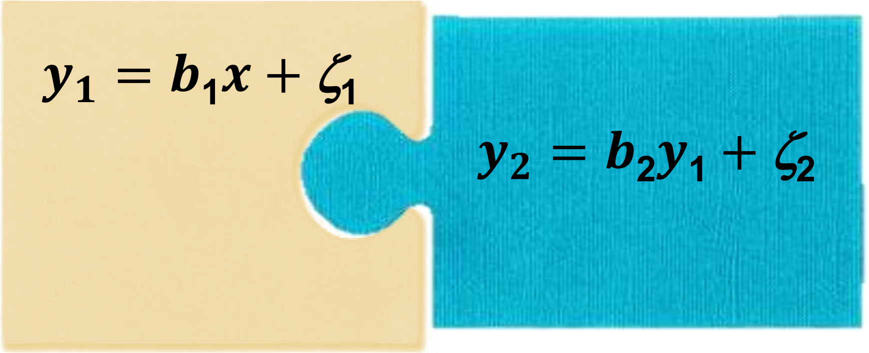
$$\left\{\begin{matrix} \sigma_{x_1} & & \\ \sigma_{x_1y_1} & \sigma_{y_1} & \\ \sigma_{x_1y_2} & \sigma_{y_1y_2} & \sigma_{y_2}\end{matrix}\right\}$$



Local Estimation

(*piecewiseSEM*)

- fit a model for each response
- strings together the inferences



$$y_1 = b_1x + \zeta_1$$

$$y_2 = b_2y_1 + \zeta_2$$

Protocol for violated assumptions of covariance-based SEM

Violated assumptions	Steps for Corrections
Non-normality of Residuals	Data transformation: e.g. *log*, *square root*
	Local estimation with GLM: package `piecewiseSEM`

Data are not multivariate normal	MLM estimation with robust SE & test statistic:
	`library(lavaan) # Always report results for 'robust' test statistics`
	`sem(…, estimator="MLM", se="robust‟`
	`#or test="Satorra-Bentler‟)`

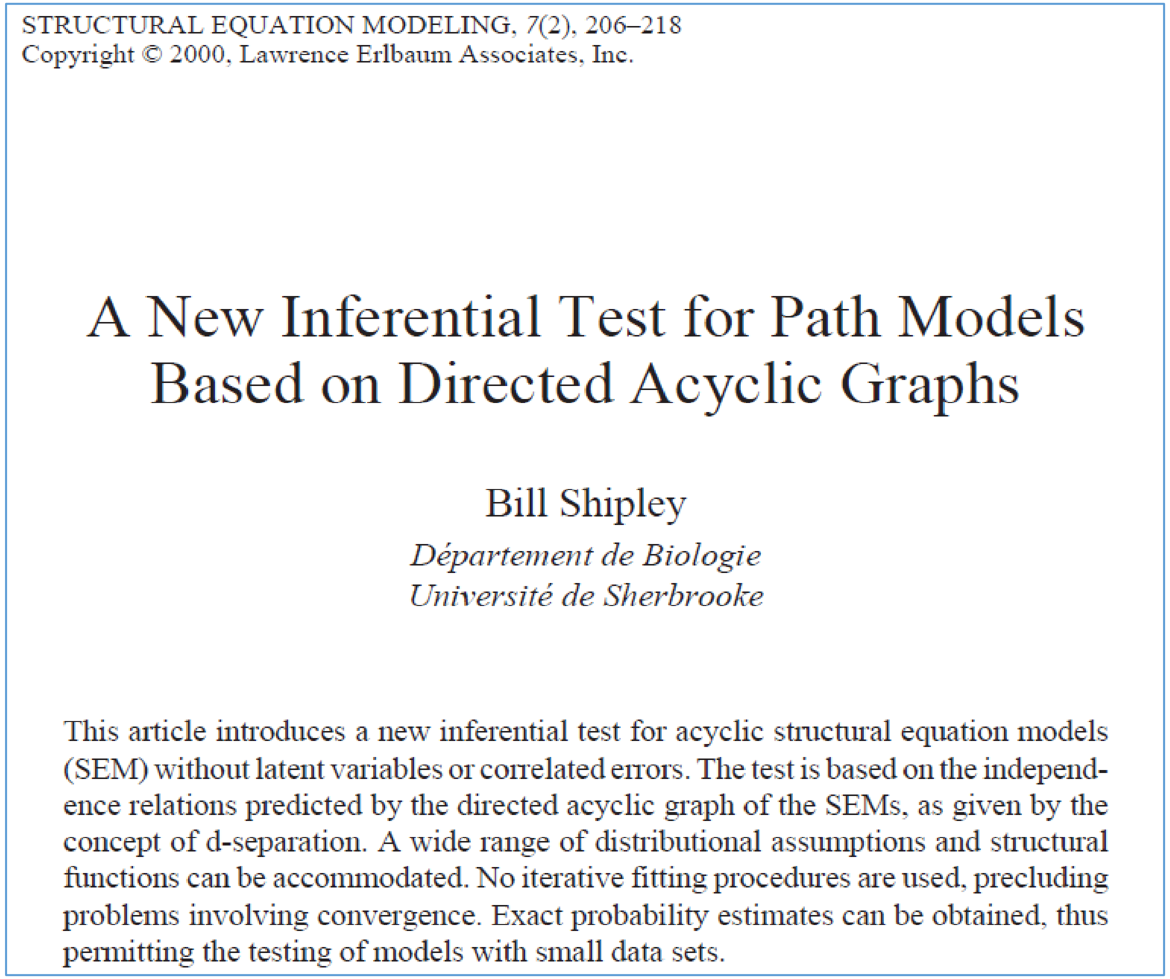
	Bootstapping: `# Always report results for 'robust' test statistics`
	`library(lavaan)`
	`sem(…, test="bollen.stine", se="bootstrap)`

Missing data	Full information maximum likelihood:
	`library(lavaan)`
	`sem(…, missing="fiml‟) #for normal data`
	`sem(…, missing="fiml‟, estimator="MLR")#for non-normal data`

Positive definite S matrix	Check for multicolinearity in each single regression model:
	`library(car)`
	`vif(m2) # vif ≤ 2 (no collinearity)`

Dependant samples (hierarchical)	Local estimation with LMM or GLMM: package `piecewiseSEM`
Not sufficient sample size	Local estimation: package `piecewiseSEM`

Global vs. Local Estimations

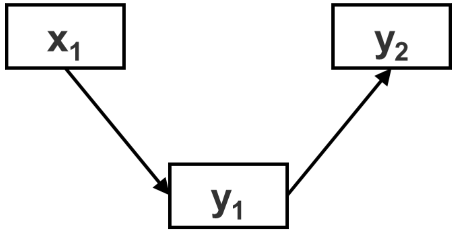


STRUCTURAL EQUATION MODELING, *7*(2), 206–218
Copyright © 2000, Lawrence Erlbaum Associates, Inc.

A New Inferential Test for Path Models Based on Directed Acyclic Graphs

Bill Shipley
Département de Biologie
Université de Sherbrooke

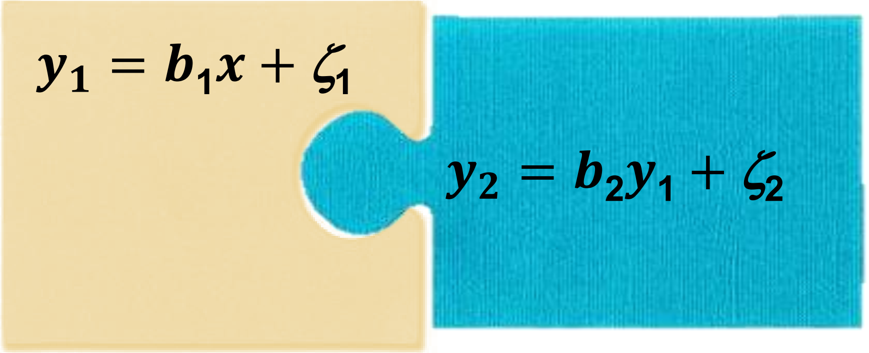
This article introduces a new inferential test for acyclic structural equation models (SEM) without latent variables or correlated errors. The test is based on the independence relations predicted by the directed acyclic graph of the SEMs, as given by the concept of d-separation. A wide range of distributional assumptions and structural functions can be accommodated. No iterative fitting procedures are used, precluding problems involving convergence. Exact probability estimates can be obtained, thus permitting the testing of models with small data sets.



Local Estimation

(*piecewiseSEM*)

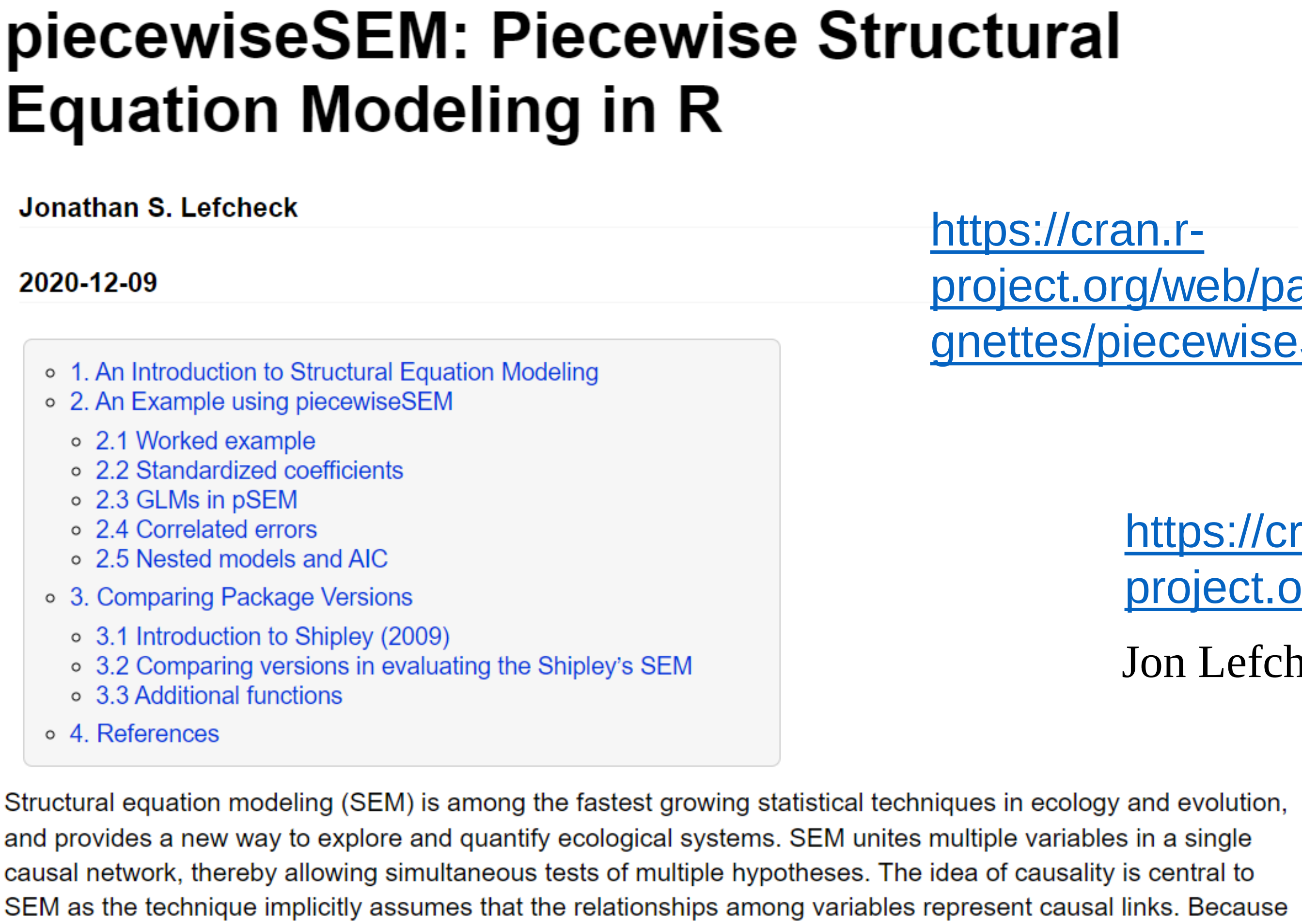
- fit a model for each response
- strings together the inferences



$$y_1 = b_1x + \zeta_1$$

$$y_2 = b_2y_1 + \zeta_2$$

Piecewise SEM



piecewiseSEM: Piecewise Structural Equation Modeling in R

Jonathan S. Lefcheck

2020-12-09

Structural equation modeling (SEM) is among the fastest growing statistical techniques in ecology and evolution, and provides a new way to explore and quantify ecological systems. SEM unites multiple variables in a single causal network, thereby allowing simultaneous tests of multiple hypotheses. The idea of causality is central to SEM as the technique implicitly assumes that the relationships among variables represent causal links. Because

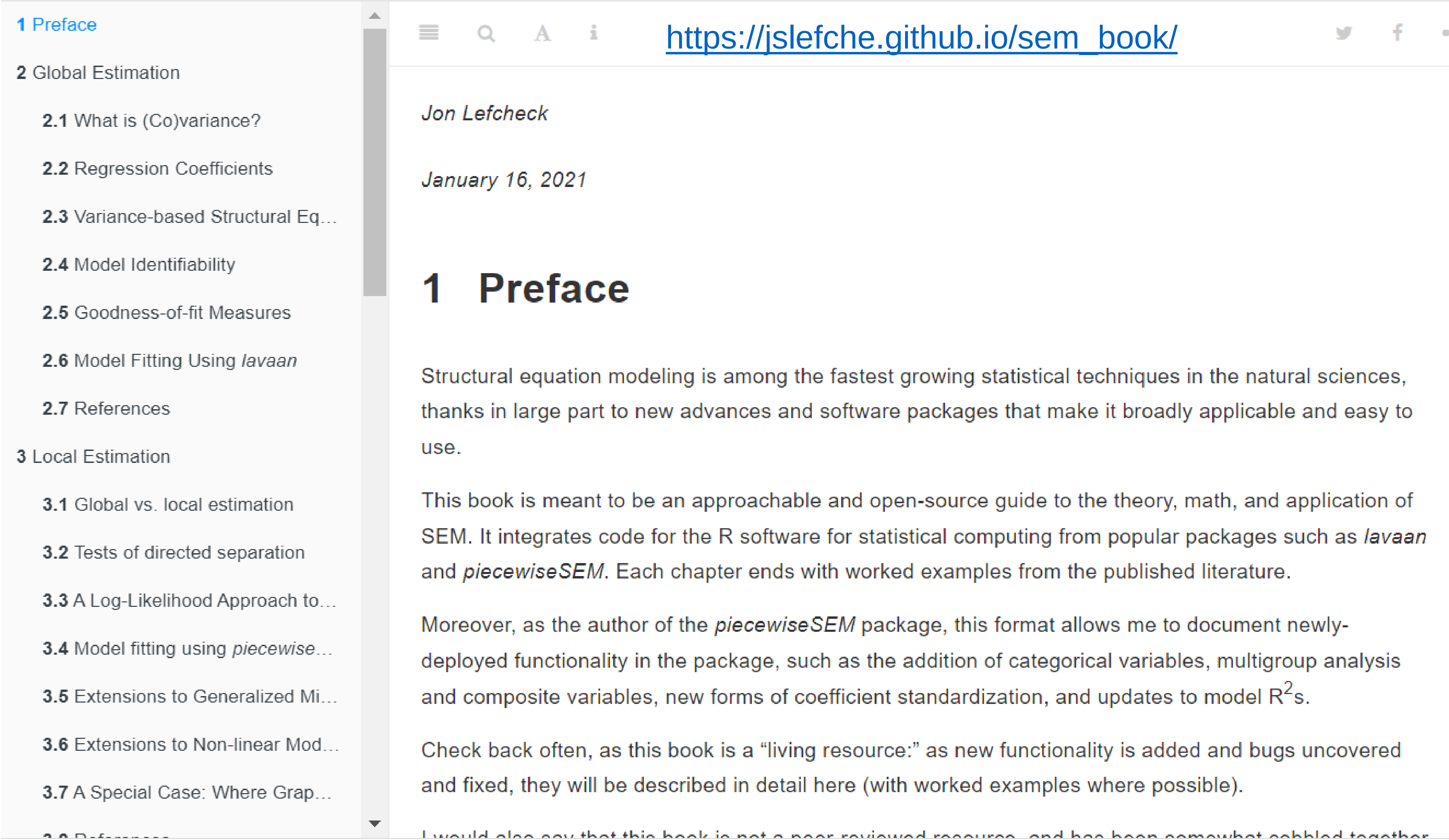
https://cran.r-project.org/web/packages/piecewiseSEM/vignettes/piecewiseSEM.html

https://cran.r-project.org/web/packages/piecewiseSEM

Jon Lefcheck [aut, cre], Jarrett Byrnes [aut], James Grace [aut]

Lefcheck, J.S. “piecewiseSEM: Piecewise structural equation modelling in r for ecology, evolution, and systematics.” Methods in Ecology and Evolution 7.5 (2016): 573-579.

Piecewise SEM



https://jslefche.github.io/sem_book/

Jon Lefcheck

January 16, 2021

1 Preface

Structural equation modeling is among the fastest growing statistical techniques in the natural sciences, thanks in large part to new advances and software packages that make it broadly applicable and easy to use.

This book is meant to be an approachable and open-source guide to the theory, math, and application of SEM. It integrates code for the R software for statistical computing from popular packages such as *lavaan* and *piecewiseSEM*. Each chapter ends with worked examples from the published literature.

Moreover, as the author of the *piecewiseSEM* package, this format allows me to document newly-deployed functionality in the package, such as the addition of categorical variables, multigroup analysis and composite variables, new forms of coefficient standardization, and updates to model R^{2}s.

Check back often, as this book is a “living resource:” as new functionality is added and bugs uncovered and fixed, they will be described in detail here (with worked examples where possible).

I would also say that this book is not a peer reviewed resource, and has been somewhat cobbled together

Global vs. Local Estimations

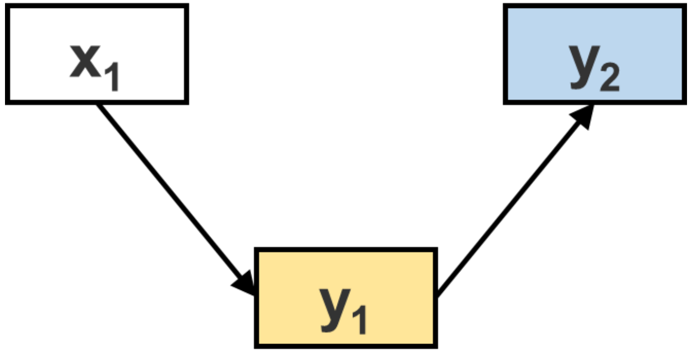
```
' y1~x1
  y2~y1 '
```

$$\left\{\begin{matrix} \sigma_{x_1} & & \\ \sigma_{x_1y_1} & \sigma_{y_1} & \\ \sigma_{x_1y_2} & \sigma_{y_1y_2} & \sigma_{y_2} \end{matrix}\right\}$$

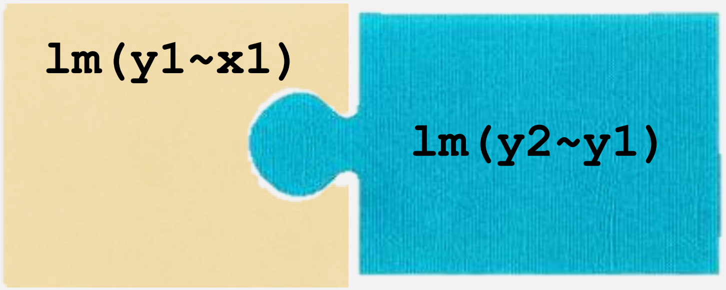
Global Estimation

(*Covariance-Based Estimation*)

- reproduces a single variance-covariance matrix



```
lm(y1~x1)
lm(y2~y1)
```



Local Estimation

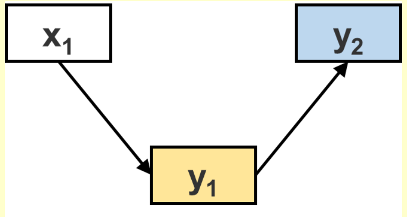
(*piecewiseSEM*)

- fit a model for each response
- strings together the inferences

Global vs. Local Estimations

$$\begin{Bmatrix} \sigma_{x_1} & & \\ \sigma_{x_1y_1} & \sigma_{y_1} & \\ \sigma_{x_1y_2} & \sigma_{y_1y_2} & \sigma_{y_2} \end{Bmatrix}$$

Global Estimation



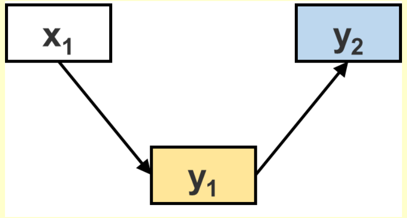
```
library(lavaan)


# Specify the model in lavaan
lav.mod <- 'y1 ~ x1
            y2 ~  y1’


# Fit the model
lav.fit <- sem(sem_mod1, data=data1)


# Extract results
summary(lav.fit, standardize=T, rsq=T,
fit.measures=TRUE)



# Plot the model
library(lavaanPlot)
lavaanPlot(model = lav.fit, coefs = TRUE, stand=TRUE,
stars = 'regress')
```

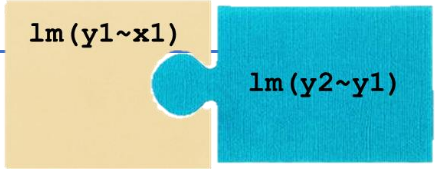
Global vs. Local Estimations

$$\left\{\begin{matrix} \sigma_{x_1} & & \\ \sigma_{x_1y_1} & \sigma_{y_1} & \\ \sigma_{x_1y_2} & \sigma_{y_1y_2} & \sigma_{y_2} \end{matrix}\right\}$$

Global Estimation

```
library(lavaan)

# Specify the model in lavaan
lav.mod <- 'y1 ~ x1
            y2 ~  y1'

# Fit the model
lav.fit <- sem(sem_mod1, data=data1)

# Extract results
summary(lav.fit, standardize=T, rsq=T,
fit.measures=TRUE)


# Plot the model
library(lavaanPlot)
lavaanPlot(model = lav.fit, coefs = TRUE, stand=TRUE,
stars = 'regress')
```





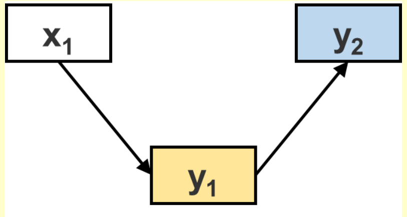
Local Estimation

```
library(piecewiseSEM)
```

Global vs. Local Estimations

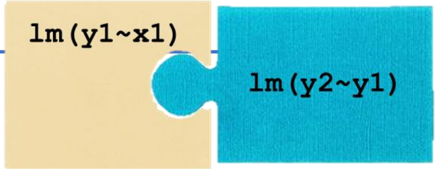
$$\begin{Bmatrix} \sigma_{x_1} & & \\ \sigma_{x_1 y_1} & \sigma_{y_1} & \\ \sigma_{x_1 y_2} & \sigma_{y_1 y_2} & \sigma_{y_2} \end{Bmatrix}$$

Global Estimation



```
library(lavaan)

# Specify the model in lavaan
lav.mod <- 'y1 ~ x1
            y2 ~  y1'

# Fit the model
lav.fit <- sem(sem_mod1, data=data1)

# Extract results
summary(lav.fit, standardize=T, rsq=T,
fit.measures=TRUE)


# Plot the model
library(lavaanPlot)
lavaanPlot(model = lav.fit, coefs = TRUE, stand=TRUE,
stars = 'regress')
```

Local Estimation



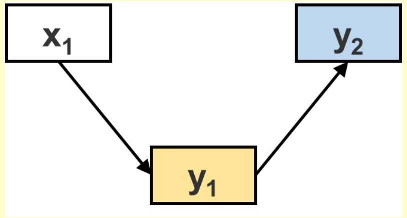
```
library(piecewiseSEM)

# Specify the model
m1 <- lm(y1 ~ x1, data = data1)
m2 <- lm(y2 ~ y1, data = data1)
```

Global vs. Local Estimations

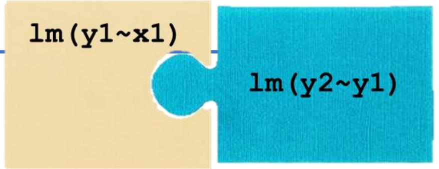
$$\left\{\begin{array}{lll} \sigma_{x_1} & & \\ \sigma_{x_1y_1} & \sigma_{y_1} & \\ \sigma_{x_1y_2} & \sigma_{y_1y_2} & \sigma_{y_2} \end{array}\right\}$$

Global Estimation

```
library(lavaan)


# Specify the model in lavaan
lav.mod <- 'y1 ~ x1
            y2 ~  y1’


# Fit the model
lav.fit <- sem(sem_mod1, data=data1)


# Extract results
summary(lav.fit, standardize=T, rsq=T,
fit.measures=TRUE)



# Plot the model
library(lavaanPlot)
lavaanPlot(model = lav.fit, coefs = TRUE, stand=TRUE,
stars = 'regress')
```



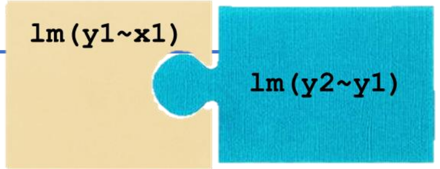
Local Estimation



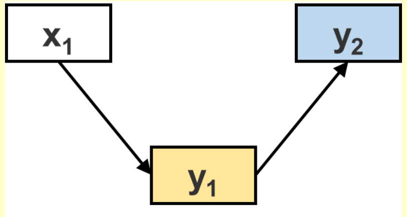
```
library(piecewiseSEM)


# Specify the model
m1 <- lm(y1 ~ x1, data = data1)
m2 <- lm(y2 ~ y1, data = data1)


# Fit the model
psem_mod <- psem(m1, m2)
```

Global vs. Local Estimations

$$\begin{Bmatrix} \sigma_{x_1} & & \\ \sigma_{x_1 y_1} & \sigma_{y_1} & \\ \sigma_{x_1 y_2} & \sigma_{y_1 y_2} & \sigma_{y_2} \end{Bmatrix}$$



Global Estimation



```
library(lavaan)


# Specify the model in lavaan
lav.mod <- 'y1 ~ x1
            y2 ~  y1’


# Fit the model
lav.fit <- sem(sem_mod1, data=data1)


# Extract results
summary(lav.fit, standardize=T, rsq=T,
fit.measures=TRUE)



# Plot the model
library(lavaanPlot)
lavaanPlot(model = lav.fit, coefs = TRUE, stand=TRUE,
stars = 'regress')
```

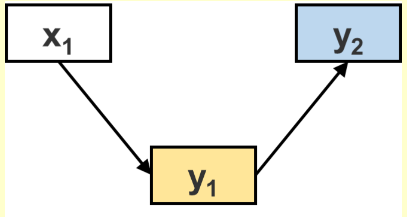
Local Estimation

```
library(piecewiseSEM)


# Specify the model
m1 <- lm(y1 ~ x1, data = data1)
m2 <- lm(y2 ~ y1, data = data1)


# Fit the model
psem_mod <- psem(m1, m2)
#or
psem_mod <- psem(
  lm(y1 ~ x1, data = data1),
  lm(y2 ~ y1, data = data1))
```

Global vs. Local Estimations

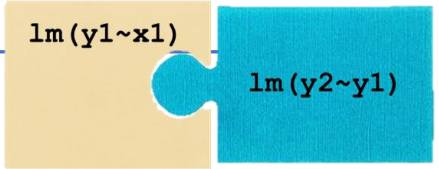
$$\begin{Bmatrix} \sigma_{x_1} & & \\ \sigma_{x_1 y_1} & \sigma_{y_1} & \\ \sigma_{x_1 y_2} & \sigma_{y_1 y_2} & \sigma_{y_2} \end{Bmatrix}$$

Global Estimation



```
library(lavaan)


# Specify the model in lavaan
lav.mod <- 'y1 ~ x1
            y2 ~  y1’


# Fit the model
lav.fit <- sem(sem_mod1, data=data1)


# Extract results
summary(lav.fit, standardize=T, rsq=T,
fit.measures=TRUE)



# Plot the model
library(lavaanPlot)
lavaanPlot(model = lav.fit, coefs = TRUE, stand=TRUE,
stars = 'regress')
```

Local Estimation



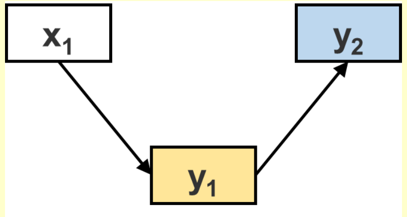
```
library(piecewiseSEM)


# Specify the model
m1 <- lm(y1 ~ x1, data = data1)
m2 <- lm(y2 ~ y1, data = data1)


# Fit the model
psem_mod <- psem(m1, m2)
#or
psem_mod <- psem(
  lm(y1 ~ x1, data = data1),
  lm(y2 ~ y1, data = data1))


# Extract results
summary(psem_mod)
```

Global vs. Local Estimations

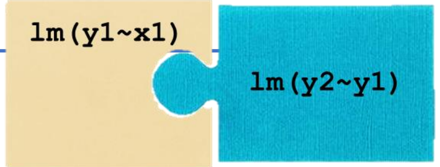
$$\left\{\begin{matrix} \sigma_{x_1} & & \\ \sigma_{x_1y_1} & \sigma_{y_1} & \\ \sigma_{x_1y_2} & \sigma_{y_1y_2} & \sigma_{y_2} \end{matrix}\right\}$$

Global Estimation

```
library(lavaan)


# Specify the model in lavaan
lav.mod <- 'y1 ~ x1
            y2 ~  y1’


# Fit the model
lav.fit <- sem(sem_mod1, data=data1)


# Extract results
summary(lav.fit, standardize=T, rsq=T,
fit.measures=TRUE)



# Plot the model
library(lavaanPlot)
lavaanPlot(model = lav.fit, coefs = TRUE, stand=TRUE,
stars = 'regress')
```

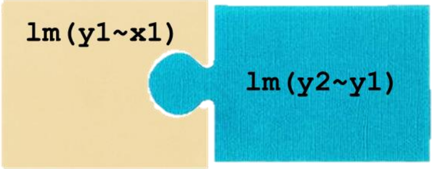


Local Estimation

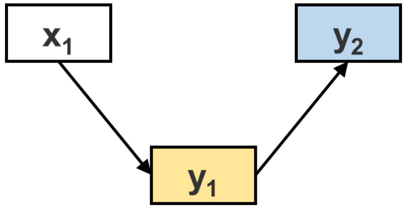


```
library(piecewiseSEM)


# Specify the model
m1 <- lm(y1 ~ x1, data = data1)
m2 <- lm(y2 ~ y1, data = data1)


# Fit the model
psem_mod <- psem(m1, m2)
#or
psem_mod <- psem(
  lm(y1 ~ x1, data = data1),
  lm(y2 ~ y1, data = data1))


# Extract results
summary(psem_mod)


# Plot the model
plot(psem_mod)
```

Piecewise SEM



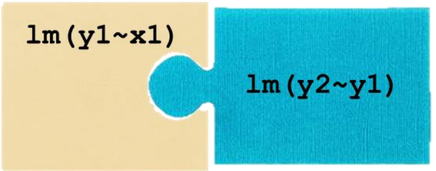
```
# additional syntax
library(piecewiseSEM)

   ~    # regressed
%~~%  # correlated errors
```

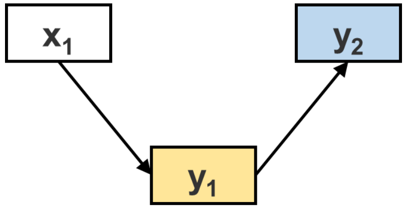


Local Estimation

```
library(piecewiseSEM)


# Specify the model
m1 <- lm(y1 ~ x1, data = data1)
m2 <- lm(y2 ~ y1, data = data1)


# Fit the model
psem_mod <- psem(m1, m2)
#or
psem_mod <- psem(
  lm(y1 ~ x1, data = data1),
  lm(y2 ~ y1, data = data1))


# Extract results
summary(psem_mod)



# Plot the model
plot(psem_mod)
```

Piecewise SEM





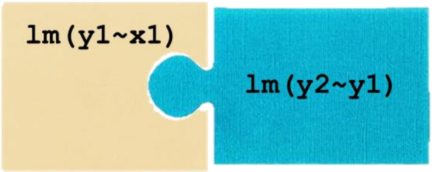
```
# additional syntax
library(piecewiseSEM)

   ~     # regressed
   ~~    # correlated
  %~~%   # correlated errors
```

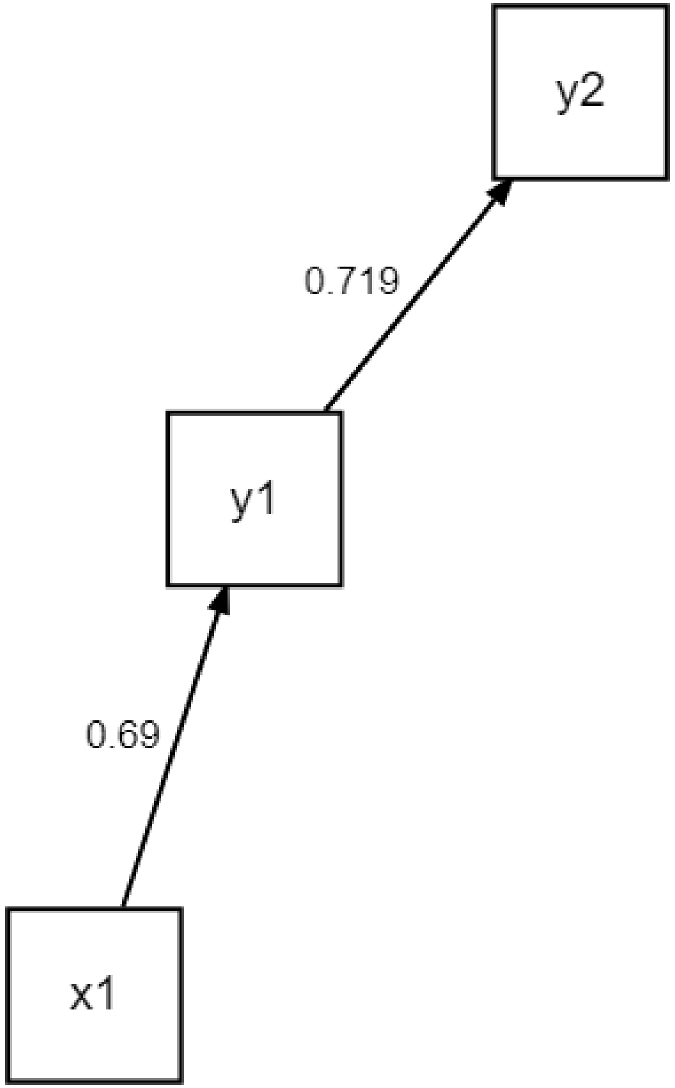
```
# development branch of piecewiseSEM from github

library(devtools)

install_github("jslefche/piecewiseSEM@devel",
                build_vignette = TRUE)
```

Local Estimation

```
library(piecewiseSEM)


# Specify the model
m1 <- lm(y1 ~ x1, data = data1)
m2 <- lm(y2 ~ y1, data = data1)


# Fit the model
psem_mod <- psem(m1, m2)
#or
psem_mod <- psem(
  lm(y1 ~ x1, data = data1),
  lm(y2 ~ y1, data = data1))


# Extract results
summary(psem_mod)


# Plot the model
plot(psem_mod)
```

Piecewise SEM

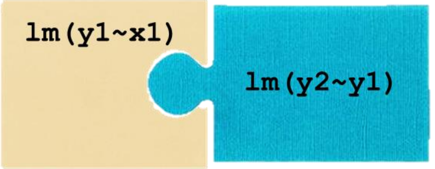


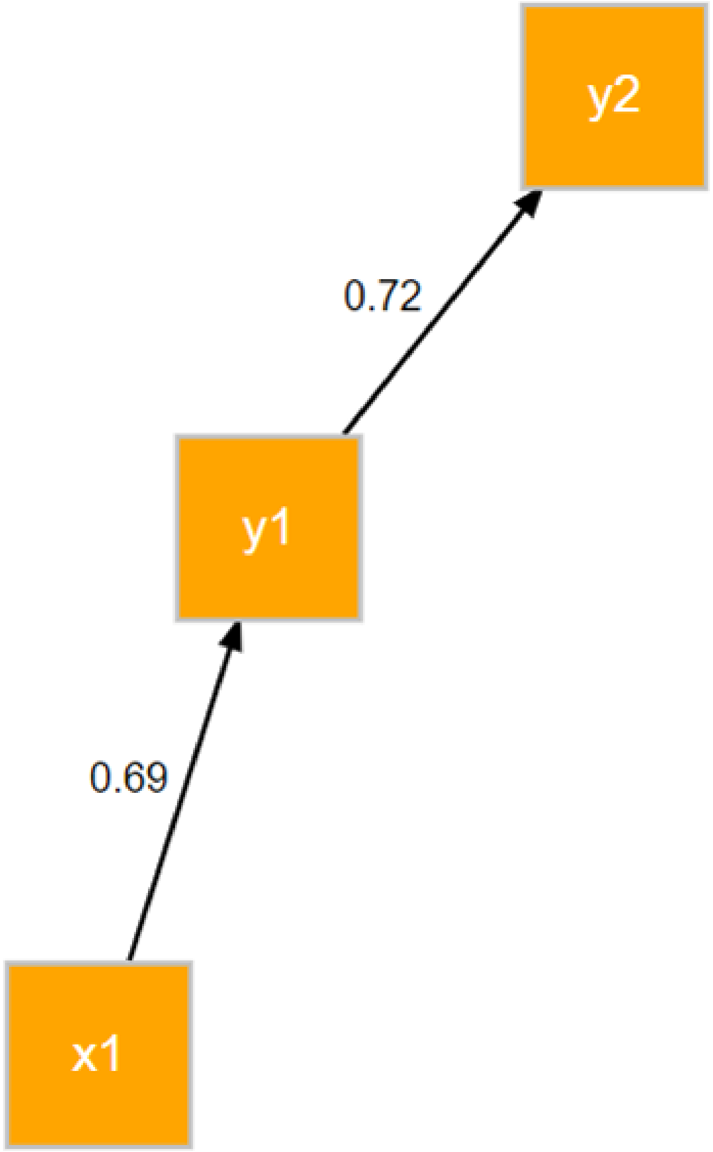


Local Estimation

```
library(piecewiseSEM)


# Specify the model
m1 <- lm(y1 ~ x1, data = data1)
m2 <- lm(y2 ~ y1, data = data1)


# Fit the model
psem_mod <- psem(m1, m2)



# Plot the model
plot(psem_mod)
```

Piecewise SEM

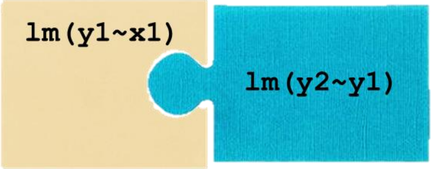




Local Estimation

```
library(piecewiseSEM)

# Specify the model
m1 <- lm(y1 ~ x1, data = data1)
m2 <- lm(y2 ~ y1, data = data1)

# Fit the model
psem_mod <- psem(m1, m2)


# Plot the model
plot(psem_mod)


# see
?plot.psem
```

Piecewise SEM



```
> psem_mod

Structural Equations of x :
lm: y1 ~ x1
lm: y2 ~ y1

Data:   y1          x1        y2
1 0.46353960 0.8461242 0.6299346
2 0.48084442 0.6626756 1.2847942
3 0.03044414 0.3068555 0.8495422
4 0.78902095 0.9327949 1.0768962
5 0.25329331 1.0490908 0.7902750
6 0.61409128 0.7134605 1.4559118
...with  94  more rows

[1] "class(psem)"
```

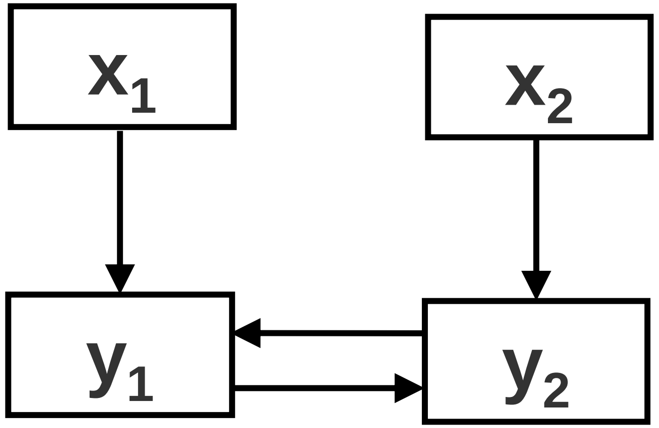
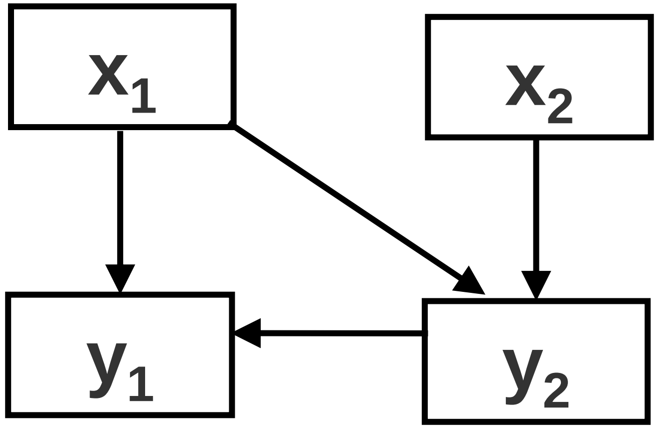
Local Estimation

```
library(piecewiseSEM)


# Specify the model
m1 <- lm(y1 ~ x1, data = data1)
m2 <- lm(y2 ~ y1, data = data1)


# Fit the model
psem_mod <- psem(m1, m2)



# Plot the model
plot(psem_mod)

# Call model
psem_mod
```

Global vs. Local Estimations

Properties:	Covariance-based SEM	Piecewise SEM
Estimation procedure	Single (global) variance-covariance matrix estimated	Variance-covariance matrices estimated separately for each endogenous variable
Solutions from the estimation	Simultaneous solution (computationally intensive)	Multiple solutions (modularized)
Data and residual distribution	Fit to normal distribution	Assumes constant variance an independence of errors for each regression equation. If violated, it icorporates various distributions (Poisson, Gamma, etc.)
Sample size	Minimum requirement $\boldsymbol{n} = \boldsymbol{p} \times 5$**,** where $\boldsymbol{n}$ sample size, $\boldsymbol{p}$ number of path coefficients	Only enough data is needed to be able to fit and estimate each individual regression
Independence of samples	Assumes independence of samples	Can model non-independence (blocked, temporal, spatial, etc.)
Latent variables	Latent & composite variables	No latent or composite variables (yet*)
Feedback-loops in a model	Non-recursive (cyclic) models are possible	Only for recursive (acyclic) models, i.e. no bidirectional relationships

Global vs. Local Estimations

Properties:	Covariance-based SEM	Piecewise SEM
	 **Non-recursive models** • with bidirectional feedbacks	 **Recursive models** • all causal effects are unidirectional
Feedback-loops in a model	Non-recursive (cyclic) models are possible	Only for recursive (acyclic) models, i.e. no bidirectional relationships

Day 8 – Part 1

Outline

- Introduction to Local Estimation in SEM

  - ✓ Global vs. Local Estimations. Piecewise SEM
  - ✓ **Assessing Model Fit**
  - ✓ Model Comparison
  - ✓ Categorical Data in Piecewise SEM

The concept of Goodness of Fit

Are we ignoring important links?

Links = Processes

When we are missing important paths:

- our parameter estimates may be incorrect
- our model is misspecified

Does the model fit the data?

=

Does the model represent the data well?

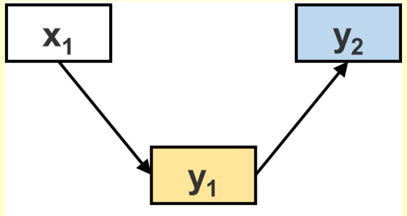
=

Are we missing important information?

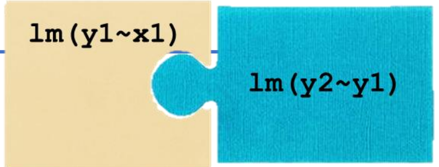
Assessing Model Fit

$$\left\{\begin{matrix} \sigma_{x_1} & & \\ \sigma_{x_1 y_1} & \sigma_{y_1} & \\ \sigma_{x_1 y_2} & \sigma_{y_1 y_2} & \sigma_{y_2} \end{matrix}\right\}$$

Global Estimation



```
library(lavaan)


# Specify the model in lavaan
lav.mod <- 'y1 ~ x1
            y2 ~  y1’
# Fit the model
lav.fit <- sem(sem_mod1, data=data1)


# Extract results
summary(lav.fit, standardize=T, rsq=T,
fit.measures=TRUE)

# or call the fit measures:


fitMeasures(lav.fit)
```



Local Estimation

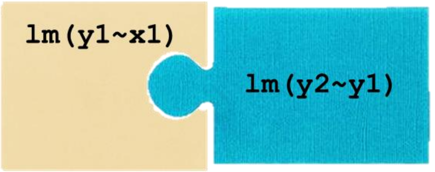
```
library(piecewiseSEM)


# Specify the model
m1 <- lm(y1 ~ x1, data = data1)
m2 <- lm(y2 ~ y1, data = data1)
# Fit the model
psem_mod <- psem(m1, m2)


# Extract results
summary(psem_mod)

# or call the fit measures:

# 1) Fisher’s C statistic

fisherC(psem_mod)

# 2) χ2 statistic

LLchisq(psem_mod)
```

Goodness of fit for Global Estimation lavaan

Measure	Name	Description	Cut-off for ‘good’ fit
χ^2	Model Chi-Square	Assess overall fit and the discrepancy between the observed and model-implied covariance matrices. Sensitive to sample size. H0: The model fits perfectly. (Present: χ^2, DF, p)	p-value > 0.05
RMSEA	Root Mean Square Error of Approximation	The square-root of the difference between the observed and model-implied covariance matrices. A parsimony-adjusted index. Values closer to 0 represent a good fit. RMSEA < 0.10 is generally ‘acceptable’ value. (Present: RMSEA, 90%CI, p_{RMSEA})	RMSEA < 0.08
CFI	Comparative Fit Index	Compares the fit of a model to the fit of a ‘null’ model (which estimates all variances but sets the covariances to 0). Low sensitivity to sample size.	CFI ≥ 0.90
SRMR	Standardized Root Mean Square Residual	The standardized difference between the observed and model-implied covariance matrices.	SRMR < 0.08

Principles and Practice of Structural Equation Modeling. Rex B. Kline. 2005.

Assessing Model Fit

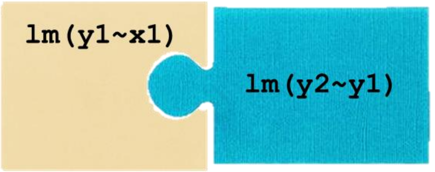


```
# both are in summary

> summary(psem_mod)
--
Global goodness-of-fit:

Chi-Squared = 1.064 with P-value = 0.302 and on 1 df
Fisher's C = 2.336 with P-value = 0.311 and on 2 df
```

Local Estimation

```
library(piecewiseSEM)


# Specify the model
m1 <- lm(y1 ~ x1, data = data1)
m2 <- lm(y2 ~ y1, data = data1)
# Fit the model
psem_mod <- psem(m1, m2)


# Extract results
summary(psem_mod)

# or call the fit measures:

# 1) Fisher’s C statistic

fisherC(psem_mod)

# 2) χ2 statistic

LLchisq(psem_mod)
```

Assessing Model Fit

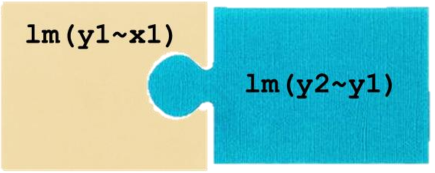


```
# both are in summary

> summary(psem_mod)
--
Global goodness-of-fit:

Chi-Squared = 1.064 with P-value = 0.302 and on 1 df
Fisher's C = 2.336 with P-value = 0.311 and on 2 df

# or

# 1) Fisher’s C statistic
> fisherC(psem_mod)
  Fisher.C df P.Value
1    2.336  2   0.311

# 2) χ2 statistic
> LLchisq(psem_mod)
  Chisq df P.Value
1 1.064  1   0.302
```

Local Estimation

```
library(piecewiseSEM)


# Specify the model
m1 <- lm(y1 ~ x1, data = data1)
m2 <- lm(y2 ~ y1, data = data1)
# Fit the model
psem_mod <- psem(m1, m2)


# Extract results
summary(psem_mod)

# or call the fit measures:

# 1) Fisher’s C statistic

fisherC(psem_mod)

# 2) χ2 statistic

LLchisq(psem_mod)
```

More functions:



Local Estimation

```
library(piecewiseSEM)
# Extract results
summary(psem_mod)

# see also coefficients

coefs(psem_mod, standardize = "none", intercepts = TRUE)

# get R square:

rsquared(psem_mod)
```

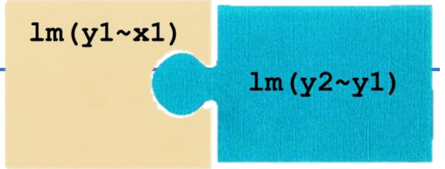
```
> coefs(psem_mod, standardize = "none", intercepts = TRUE)
  Response   Predictor Estimate Std.Error DF Crit.Value P.Value
1       y1 (Intercept)  -0.0423    0.0588 98    -0.7190  0.4738
2       y1          x1   0.5171    0.0548 98     9.4291  0.0000 ***
3       y2 (Intercept)   0.4210    0.0606 98     6.9489  0.0000 ***
4       y2          y1   1.1314    0.1104 98    10.2470  0.0000 ***

> rsquared(psem_mod)
   Response   family     link method R.squared
1       y1 gaussian identity   none 0.4756782
2       y2 gaussian identity   none 0.5172421
```

Missing links?

$$\left\{\begin{matrix} \sigma_{x_1} & & \\ \sigma_{x_1 y_1} & \sigma_{y_1} & \\ \sigma_{x_1 y_2} & \sigma_{y_1 y_2} & \sigma_{y_2} \end{matrix}\right\}$$

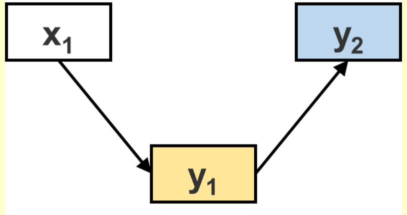
Global Estimation

```
library(lavaan)

# Request modification indices
summary(sem_fit1, standardize = T,
                   modindices=T)
# or
modificationIndices(sem_fit1)
```



Local Estimation

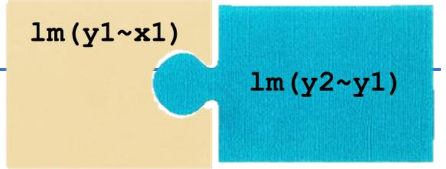
```
library(piecewiseSEM)
```



Missing links?

$$\left\{\begin{array}{lll} \sigma_{x_1} & & \\ \sigma_{x_1 y_1} & \sigma_{y_1} & \\ \sigma_{x_1 y_2} & \sigma_{y_1 y_2} & \sigma_{y_2} \end{array}\right\}$$

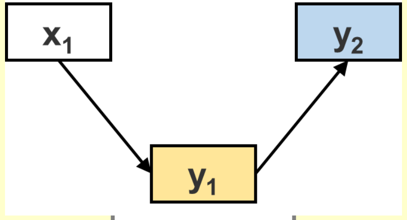
Global Estimation



Local Estimation

```
library(lavaan)


# Request modification indices
summary(sem_fit1, standardize = T,
                         modindices=T)
# or
modificationIndices(sem_fit1)



> modificationIndices(lav.fit,
standardized=F)
  lhs op rhs    mi    epc
6  y1  ~  y2 1.058  0.099
7  y2  ~  x1 1.058 -0.116
9  x1  ~  y2 1.058 -0.173
```

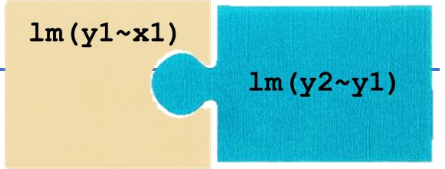
```
library(piecewiseSEM)
```



Missing links?

$$\begin{Bmatrix} \sigma_{x_1} & & \\ \sigma_{x_1y_1} & \sigma_{y_1} & \\ \sigma_{x_1y_2} & \sigma_{y_1y_2} & \sigma_{y_2} \end{Bmatrix}$$

Global Estimation

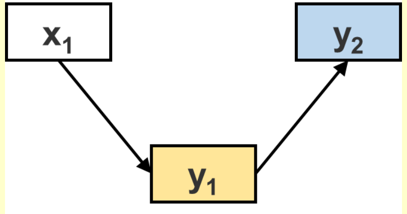


Local Estimation

```
library(lavaan)


# Request modification indices
summary(sem_fit1, standardize = T,
                    modindices=T)
# or
modificationIndices(sem_fit1)



> modificationIndices(lav.fit,
standardized=F)
  lhs op rhs    mi    epc
6   y1  ~  y2 1.058  0.099
7   y2  ~  x1 1.058 -0.116
9   x1  ~  y2 1.058 -0.173
```

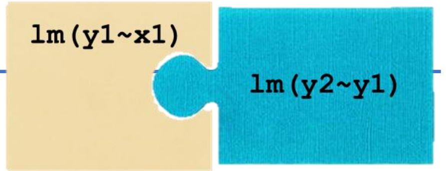
Drop in χ2 if link is added

```
library(piecewiseSEM)
```



Missing links?

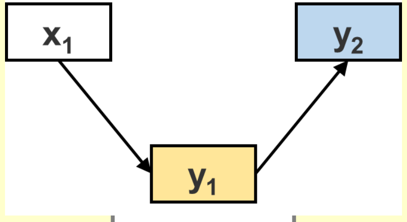
$$\begin{Bmatrix} \sigma_{x_1} & & \\ \sigma_{x_1y_1} & \sigma_{y_1} & \\ \sigma_{x_1y_2} & \sigma_{y_1y_2} & \sigma_{y_2} \end{Bmatrix}$$



Global Estimation

```
library(lavaan)


# Request modification indices
summary(sem_fit1, standardize = T,
                     modindices=T)
# or
modificationIndices(sem_fit1)



> modificationIndices(lav.fit,
standardized=F)
  lhs op rhs    mi    epc
6  y1  ~  y2 1.058  0.099
7  y2  ~  x1 1.058 -0.116
9  x1  ~  y2 1.058 -0.173
```

Drop in χ2 if link is added



Local Estimation

```
library(piecewiseSEM)

# Tests of directed separation
summary(psem_mod)







# or

dSep(psem_mod)
```

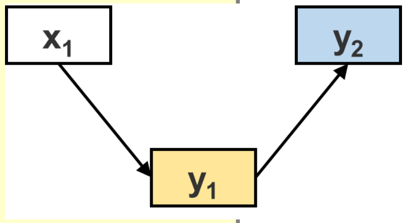
Missing links?

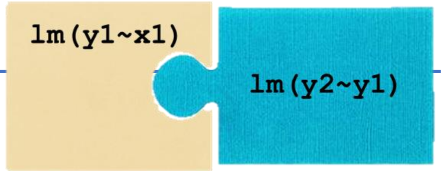
$$\left\{\begin{matrix} \sigma_{x_1} & & \\ \sigma_{x_1y_1} & \sigma_{y_1} & \\ \sigma_{x_1y_2} & \sigma_{y_1y_2} & \sigma_{y_2}\end{matrix}\right\}$$

Global Estimation

```
library(lavaan)


# Request modification indices
summary(sem_fit1, standardize = T,
                    modindices=T)
# or
modificationIndices(sem_fit1)



> modificationIndices(lav.fit,
standardized=F)
  lhs op rhs    mi    epc
6  y1  ~  y2 1.058  0.099
7  y2  ~  x1 1.058 -0.116
9  x1  ~  y2 1.058 -0.173
```

Drop in χ2 if link is added





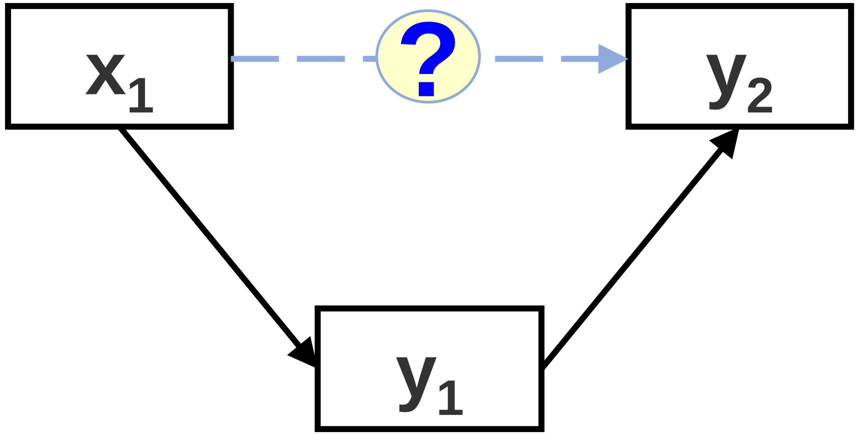
Local Estimation

```
library(piecewiseSEM)


# Tests of directed separation
summary(psem_mod)
>
---
Tests of directed separation:

   Independ.Claim Test.Type DF Crit.Value P.Value
    y2 ~ x1 + ...      coef 97    -1.0186  0.3109



# or

dSep(psem_mod)
>
Independ.Claim    Test.Type DF Crit.Value  P.Value
1  y2 ~ x1 + ...       coef 97  -1.018557 0.3109471
```

missing effect is not different from 0

Missing links?

Tests of directed (d-)separation



Independence claim

$x_1|y_2(y_1)$

x_1 and y_2 are independent conditioned on y_1

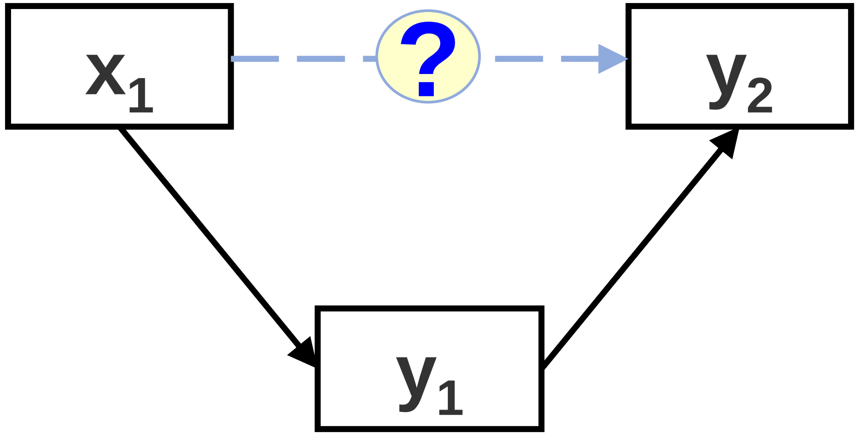
effect of x_1 on y_2 not other way around or both

Basis set is the minimum number of independence claims derived from a path diagram.

Missing links?

Tests of directed (d-)separation

```
# Derive the basis set
basisSet(psem_mod)
> $`1`
[1] "x1 | y2 ( y1 )"
```



Independence claim

$x_1|y_2(y_1)$

x_1 and y_2 are independent conditioned on y_1

effect of x_1 on y_2 not other way around or both

Basis set is the minimum number of independence claims derived from a path diagram.

Missing links?

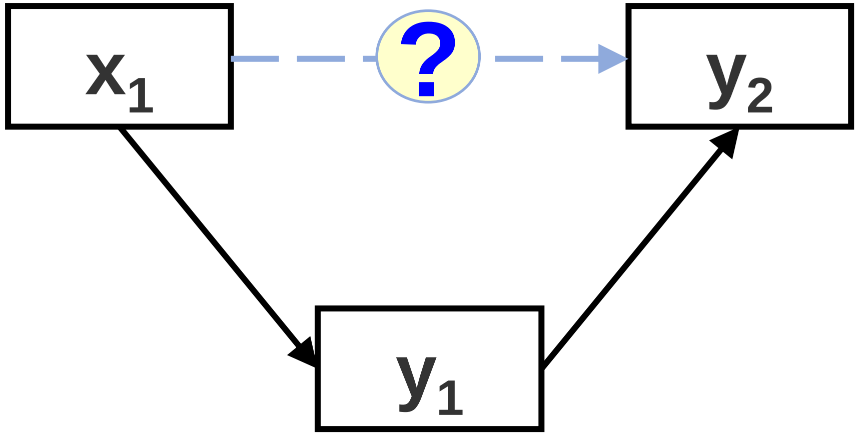
Tests of directed (d-)separation

Two variables are said to be *d-separated* if they are statistically **independent** conditional on their joint influences

no directed path connecting these two variables

accounting for contributions from other variables through indirect effects

H_0: partial effect of x_1 on y_2 is not different from 0



Independence claim

$x_1|y_2(y_1)$

x_1 and y_2 are independent conditioned on y_1

effect of x_1 on y_2 not other way around or both

Basis set is the minimum number of independence claims derived from a path diagram.

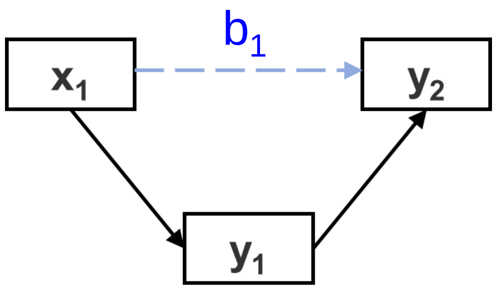
Missing links?

Tests of directed (d-)separation

(of statistical independence):

Independence claim

$x_1|y_2(y_1)$



Steps for each independence claim:

1. Fit sub-model including missing link
2. Extract *p*-value associated with that missing link
3. If $p < 0.05$ reject H_0 (that the missing effect is not different 0), suggesting model change

$y_2 = a + b_1 x_1 + b_2 y_1$

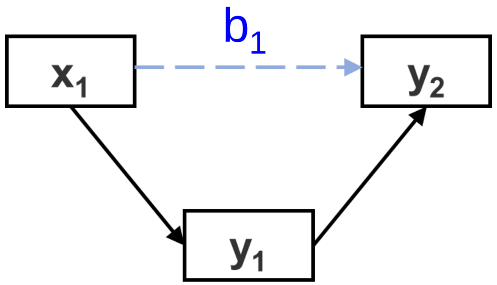
Missing links?

Tests of directed (d-)separation

(of statistical independence):

Independence claim

$x_1|y_2(y_1)$



```
# Tests of directed separation
dSep(psem_mod1)
>
 Independ.Claim Test.Type DF Crit.Value   P.Value
1  y2 ~ x1 + ...       coef 97  -1.018557 0.3109471


 # Manually calculated:
 summary(lm(y2 ~ y1+x1, data = data1))$coefficients[3, ]
 >
   Estimate  Std. Error    t value   Pr(>|t|)
 -0.1164159  0.1142950   -1.0185570  0.3109471
```

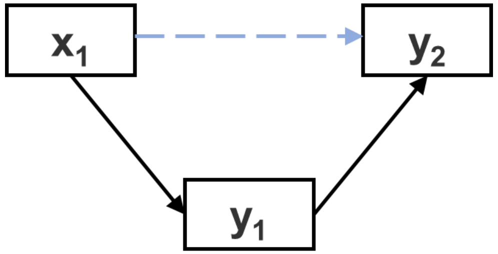
$$y_2 = a + b_1 x_1 + b_2 y_1$$

missing effect is not different from 0

Assessing Model Fit

Fisher’s *C* statistic

Independence claim $x_1|y_2(y_1)$



k is the number of independence claims in the basis set

$$C = -2\sum_{i=1}^{k} \ln(p_i)$$

p is the *p*-value from the d-separation test for each ***i***th claim

i is the ***i***th claim

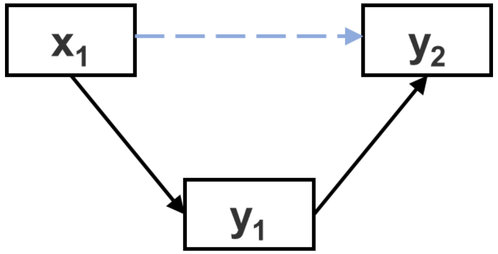
Model fit
Fisher’s ***C*** statistic

C is χ^2 distributed with $2k$ degrees of freedom
(allows obtaining a model-wide P-value)

Good fit: P > 0.05 (model is supported by the data)

Assessing Model Fit

Fisher’s *C* statistic



$$\boldsymbol{AIC = C + 2\mathrm{K}}$$

Fisher’s ***C***

K is the likelihood degrees of freedom

number of likelihood-estimated parameters)

$$\boldsymbol{AIC_c = C + 2\mathrm{K}\frac{n}{(n - \mathrm{K} - 1)}}$$

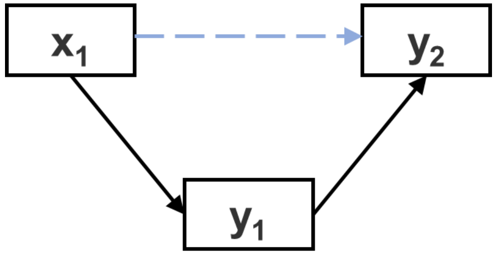
$\boldsymbol{AIC_c}$ is used when n/K < 40

$\boldsymbol{n}$ is sample size

Assessing Model Fit

Fisher’s *C* statistic

```
# Fisher’s C statistic
fisherC(psem_mod1)
>
    Fisher.C   df   P.Value
1   2.336       2   0.311

# Manually calculated:
C <- -2 * log(summary(lm(y2 ~ y1+x1, data = data1))$coefficients[3, 4])
> C
[1] 2.336265

1-pchisq(C, 2) # 2 DF (DF=2k; k - number of independence claims)
>
[1] 0.3109471


# AIC value based on the Fisher’s C statistic and the d-sep tests

AIC(psem_mod1, AIC.type = "dsep", aicc = TRUE)
   AIC     AICc     K   n
1  14.336  15.415   6   100
```



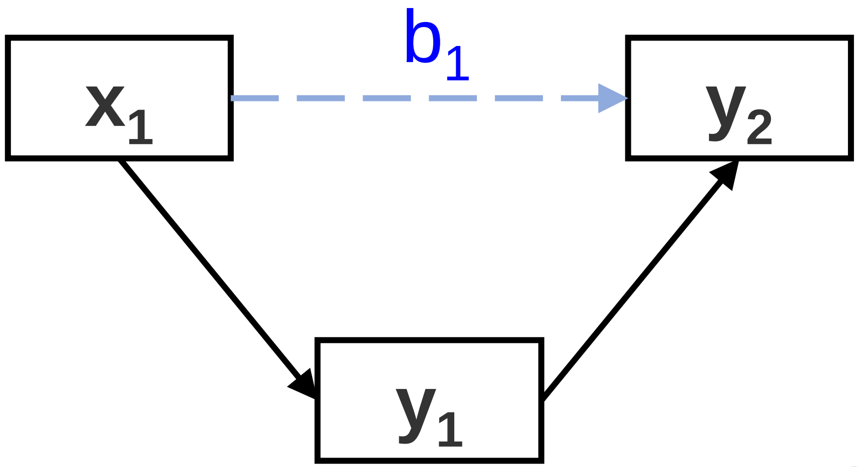
$$C = -2\sum_{i=1}^{k} \ln(p_i)$$

$$AIC = C + 2\text{K}$$

$$AIC_c = C + 2\text{K}\frac{n}{(n - \text{K} - 1)}$$

Assessing Model Fit

χ^2 statistic



$$y_2 = a + b_1 x_1 + b_2 y_1$$

Log-Likelihood test

$$y_2 = a + 0 * x_1 + b_2 y_1$$

Extended to the entire SEM

Log-Likelihood Approach

Compares the fitted sub-model to fully saturated sub-model

Assessing Model Fit

χ^2 statistic

$\boldsymbol{k}$ is the number of sub-models in the SEM model

Log-likelihood of our (nested) $\boldsymbol{i}^{\text{th}}$ sub-model $\mathbf{M}$

Log-likelihood of the fully saturated $\boldsymbol{i}^{\text{th}}$ sub-model $\mathbf{M}\boldsymbol{s}$

χ^2 of the SEM model

$$\boldsymbol{\chi^2 = -2\sum_{i=1}^{k}(\log(L_{M_i}) - \log(L_{Ms_i}))}$$

$\boldsymbol{i}$ is the $\boldsymbol{i}^{\text{th}}$ sub-model in the SEM model

$$\boldsymbol{AIC = \sum_{i=1}^{k} AIC(M_i)}$$

$\boldsymbol{AIC}$ of the SEM model

The same for the $\boldsymbol{AIC_c}$

Assessing Model Fit

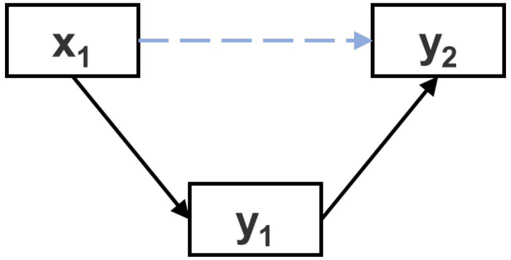
χ^2 statistic

```
# log-likelihood based χ2 statistic

LLchisq(psem_mod1)
>
   Chisq  df   P.Value
1  1.064   1   0.302
```

$$\chi^2 = -2\sum_{i=1}^{k}(\log(L_{M_i}) - \log(L_{Ms_i}))$$

```
# Manually calculated:
LL1 <- logLik(lm(y2 ~ y1, data=data1)) - logLik(lm(y2 ~ y1+x1, data=data1))
LL2 <- logLik(lm(y1 ~ x1, data=data1)) - logLik(lm(y1 ~ x1, data=data1))
#
ChiSq <- -2*sum(as.numeric(LL1), as.numeric(LL2))
> ChiSq
[1] 1.063866

DF <- 1 # one additional parameter estimated in the saturated model
1 - pchisq(ChiSq, DF)
> [1] 0.3023352

# AIC value based on log-likelihood
AIC(psem_mod1, aicc = TRUE)
>    AIC     AICc     K     n
1  -2.798   -2.298   6    100
```

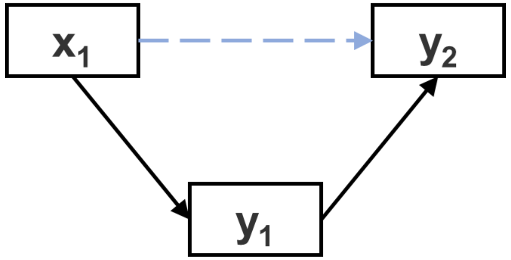
$$AIC = \sum_{i=1}^{k} AIC(M_i)$$



Assessing Model Fit

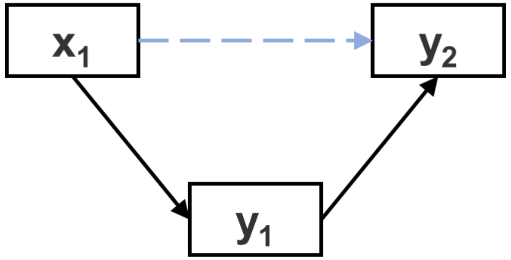
χ^2 statistic

```
# log-likelihood based χ2 statistic piecewiseSEM
LLchisq(psem_mod1)
>
   Chisq  df   P.Value
1  1.064   1   0.302


 # The same as from  lavaan
 library(lavaan)
 sem_mod1 <- '
             y1 ~ x1
             y2 ~ y1
 '
 sem_fit1 <- sem(sem_mod1, data = data1)
 fit <- lavInspect(sem_fit1, "fit")
 fit["chisq"]; fit["pvalue"]

 >
  chisq
 1.063866
  pvalue
 0.3023352
```



Assessing Model Fit

```
summary(psem_mod1)

>
Call:
  y1 ~ x1
  y2 ~ y1

    AIC
 -2.798
---
Tests of directed separation:

  Independ.Claim     Test.Type  DF    Crit.Value   P.Value
   y2 ~ x1 + ...      coef      97    -1.0186      0.3109
--
Global goodness-of-fit:

Chi-Squared = 1.064 with P-value = 0.302 and on 1 degrees of freedom
Fisher's C = 2.336 with P-value = 0.311 and on 2 degrees of freedom
---
Coefficients:

  Response Predictor Estimate Std.Error DF Crit.Value  P.Value   Std.Estimate
        y1        x1   0.5171    0.0548 98     9.4291        0       0.6897 ***
        y2        y1   1.1314    0.1104 98    10.2470        0       0.7192 ***

  Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05
---
Individual R-squared:

  Response method R.squared
        y1   none      0.48
        y2   none      0.52
```



Day 8 – Part 1

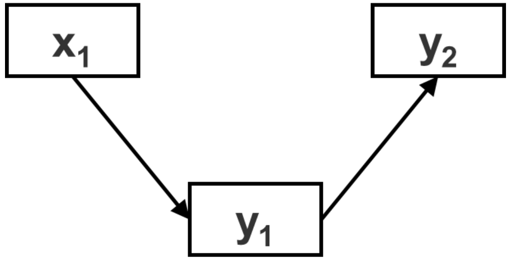
Outline

- Introduction to Local Estimation in SEM

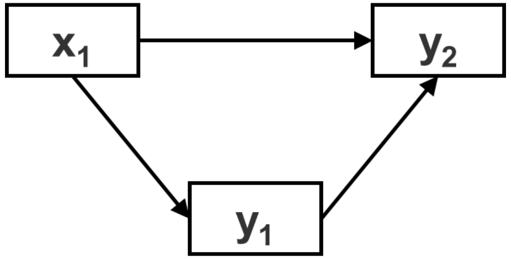
  - ✓ Global vs. Local Estimations. Piecewise SEM
  - ✓ Assessing Model Fit
  - ✓ **Model Comparison**
  - ✓ Categorical Data in Piecewise SEM

Model Comparison

```
# Model comparison
psem_mod2 <- psem(
  lm(y1 ~ x1, data = data1),
  lm(y2 ~ y1+x1, data = data1))

# χ2-difference test
anova(psem_mod2, psem_mod) # arrange correct order manually
>Chi-square Difference Test
              Df    AIC K   n Chisq   Chisq.diff   Df.diff P.value
 psem_mod2     0 -1.862 7 100 0.000
 vs psem_mod  1 -2.798 6 100 1.064      1.064        1      0.3023

# AIC comparison
aic <- AIC(psem_mod, psem_mod2)
>    AIC   K   n
1   -2.798 6   100
2   -1.862 7   100
d_aic <- aic[1] - min(aic[1])
>
 AIC
1 0.000
2 0.936
```



Mod. 1



Mod. 2

Day 8 Task 1

California, USA.

Photos credit: USFS, and Jon Keeley, USGS

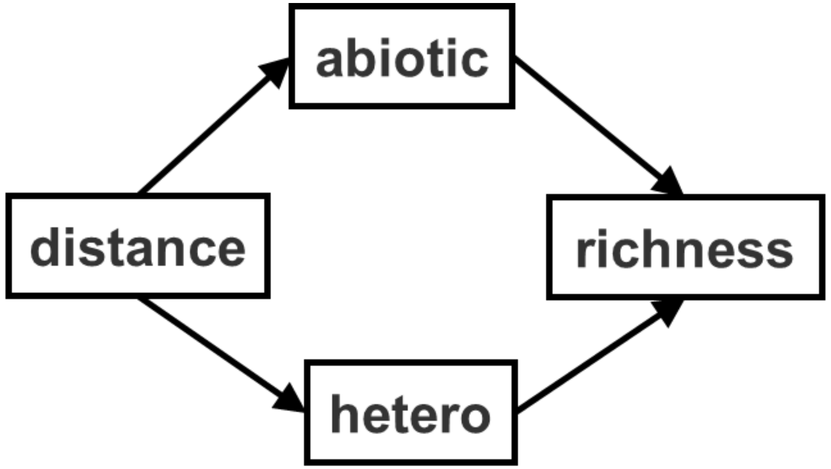
doi.org/10.1186/s42408-019-0041-0

doi.org/10.1071/WF07049

Postfire recovery of plant communities in California shrublands

- Vegetation species richness "**`richness`**"
- Local abiotic conditions (aspect, soils) "**`abiotic`**"
- Spatial heterogenity "**`hetero`**"
- Distance from coast "**`distance`**"

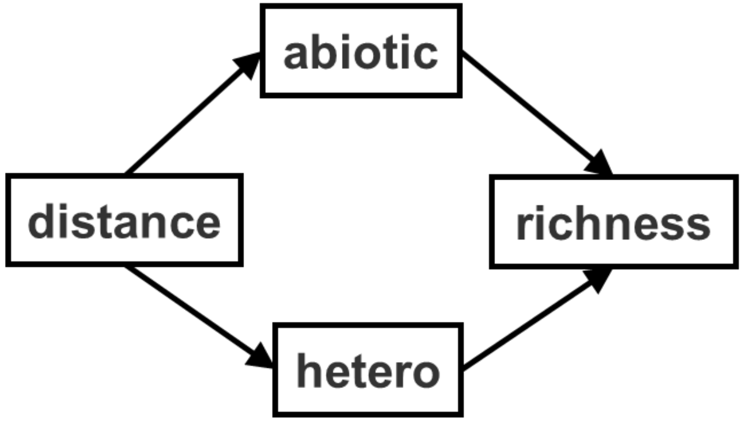
```
# Keeley data
library(piecewiseSEM)
data(keeley)
```



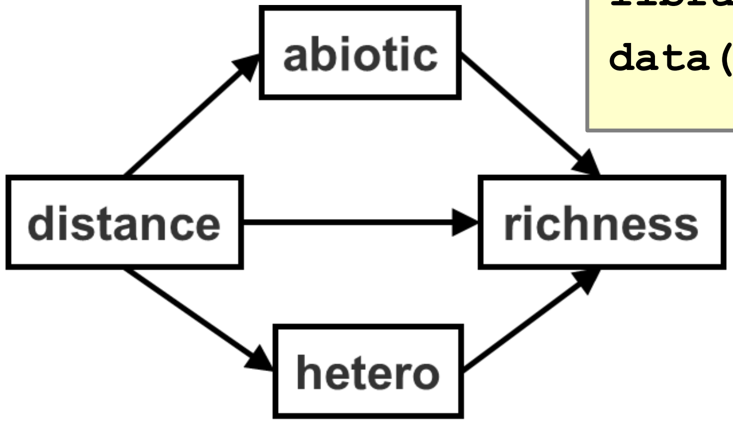
Data: Grace, J.B. and Keeley, J.E. 2006. Ecological Applications 16:503-514

Day 8 Task 1

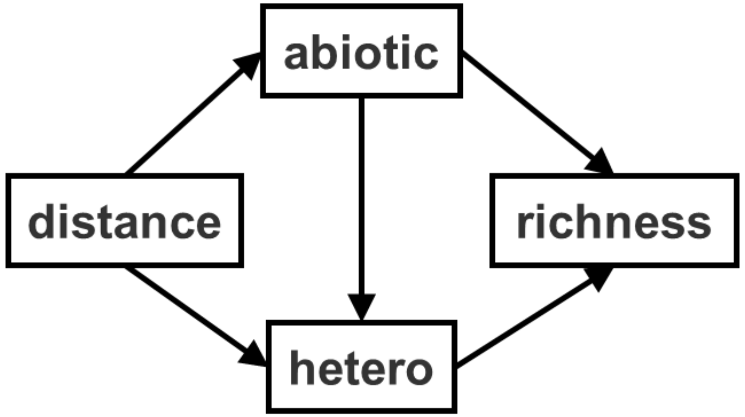
```
# Keeley data
library(piecewiseSEM)
data(keeley)
```



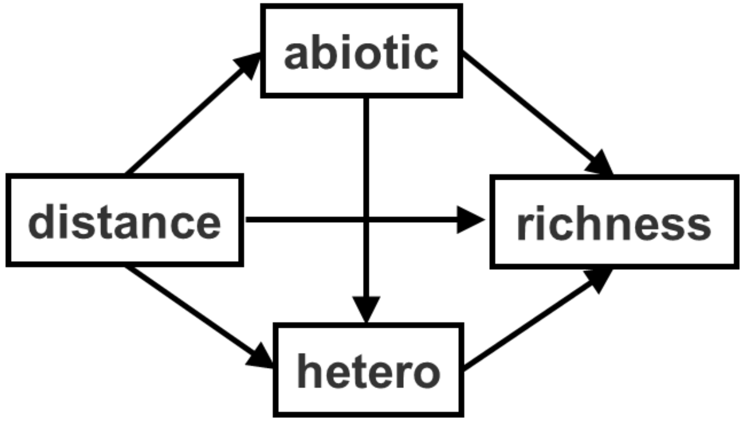
mod1



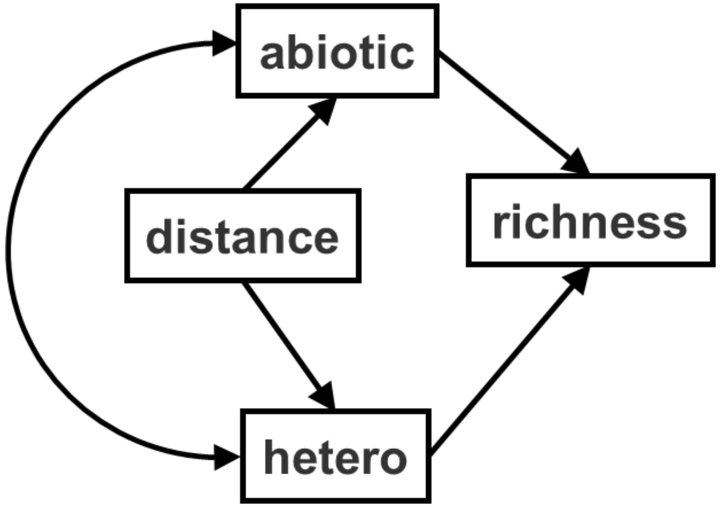
mod2



mod3



mod4



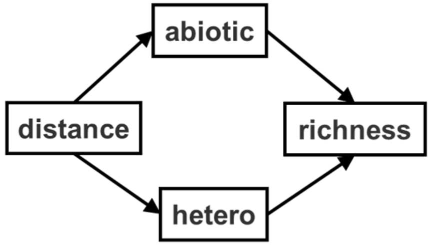
mod5

Day 8 Task 1

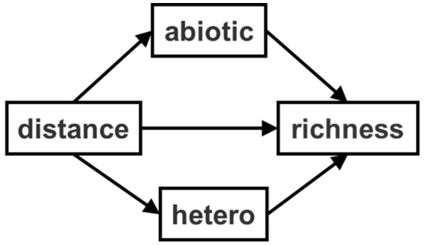
1. Specify each model in piecewiseSEM

2. Fit each model using data `data(keeley)`

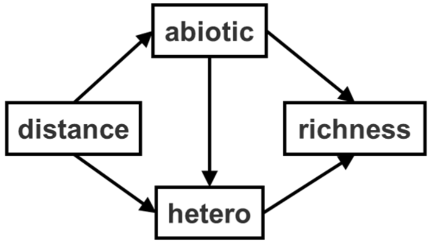
4. Compare the models using AIC or AICc and select the best model



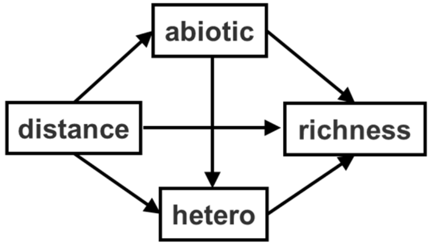
mod1



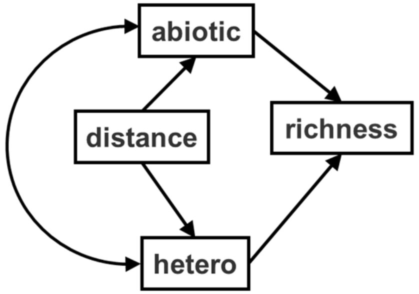
mod2



mod3



mod4



mod5

Day 8 – Part 1

Outline

- Introduction to Local Estimation in SEM

  - ✓ Global vs. Local Estimations. Piecewise SEM
  - ✓ Assessing Model Fit
  - ✓ Model Comparison
  - ✓ **Categorical Data in Piecewise SEM**

Categorical Data in Piecewise SEM

Categorical Variables	Exogenous Categorical Variables	Endogenous Categorical Variables

Binary variables
yes/no;
presence/absence;
failure/success;
dead/alive;
male/female
	1. Treat as numeric: set the values as 0 or 1 and model as numeric (yields a single path coefficient).	Endogenous categorical variables are not implemented in **piecewiseSEM**.
		Treat binary and ordinal variables as numerical (follow step 1 shown for 'Endogenous Categorical Variables')

	2. Create separate dummy variables for each factor levels with values 0, 1 each. **Rule**: for the factor with k levels use k-1 dummy variables (to avoid singularity).	
	3. Use as categorical variable (Marginal Means approach)	
Ordinal variables: small < medium < large; yang < middle < old	1. Treat as numeric: set the values depending on the order of the factor, e.g., small = 1 < medium = 2 < large = 3, and then model as numeric.	
	2. Create separate dummy variables for each factor levels with values 0, 1 each. **Rule**: for the factor with k levels use k-1 dummy variables (to avoid singularity).	
	3. Use as categorical variable (Marginal Means approach)	
Nominal variables study sites ;countries; sampling campaigns	Use as categorical variable (Marginal Means approach)	Nominal endogenous categorical variables are not implemented in **piecewiseSEM**

Exogenous Categorical Variables as Marginal Means

Marginal means are the expected average value of one predictor given the other co-variables in the model.

```
data3 <- read.csv("Data/SEMdata2.csv")
str(data3)

model1 <- lm(y ~ Group, data3)

summary(model1)
>  Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  1.05644    0.05863  18.018   <2e-16 *** # the mean of y in group “A”
GroupB      -0.09223    0.08292  -1.112    0.269 # effect of “B” on y in the absence of “A”.

# marginal mean for A - expected average value of y in group “A”(no other covariates in mod 1)
predict(model1, data.frame(Group = "A"))
>
1.056437
```

Exogenous Categorical Variables as Marginal Means

the marginal mean is evaluated while holding the covariate x at its mean value.

```
model2 <- lm(y ~ x+Group, data3)
# the marginal mean:
predict(model2, data.frame(Group = "A", x = mean(data3$x))
> 1.062983
predict(model2, data.frame(Group = “B", x = mean(data3$x))
> 0.9576651
```

Exogenous Categorical Variables as Marginal Means

the marginal mean is evaluated while holding the covariate x at its mean value.

```
model2 <- lm(y ~ x+Group, data3)
# the marginal mean:
predict(model2, data.frame(Group = "A", x = mean(data3$x))
> 1.062983
predict(model2, data.frame(Group = “B", x = mean(data3$x))
> 0.9576651
```

Exogenous Categorical Variables as Marginal Means

the marginal mean is evaluated while holding the covariate x at its mean value.

```
model2 <- lm(y ~ x+Group, data3)
# the marginal mean:
predict(model2, data.frame(Group = "A", x = mean(data3$x))
> 1.062983
predict(model2, data.frame(Group = “B", x = mean(data3$x))
> 0.9576651


library(emmeans)
emmeans(model2, specs = "Group")
>
Group emmean     SE df lower.CL upper.CL
 A      1.063 0.0419 97    0.980     1.15
 B      0.958 0.0419 97    0.875     1.04
```

specs is the variable or list of variables whose means are to be estimated

Exogenous Categorical Variables as Marginal Means

the marginal mean is evaluated while holding the covariate x at its mean value.

```
model2 <- lm(y ~ x+Group, data3)
# the marginal mean:
predict(model2, data.frame(Group = "A", x = mean(data3$x))
> 1.062983
predict(model2, data.frame(Group = “B", x = mean(data3$x))
> 0.9576651
```

specs is the variable or list of variables whose means are to be estimated

```
library(emmeans)
emmeans(model2, specs = "Group")
>
Group  emmean     SE df lower.CL upper.CL
 A      1.063 0.0419 97    0.980     1.15
 B      0.958 0.0419 97    0.875     1.04
```

pairwise Tukey tests

```
emmeans(model2, list(pairwise ~ Group))
 > ...
 1      estimate      SE  df   t.ratio   p.value
 A - B     0.105  0.0592  97   1.778     0.0786
```

Exogenous Categorical Variables as Marginal Means

```
library(piecewiseSEM)
psem_model <- psem(model2)
coefs(psem_model)
 >
Response Predictor Estimate Std.Error DF Crit.Value P.Value Std.Estimate
1        y         x   1.0529     0.108 97     9.7511  0.0000       0.6994 ***
2        y     Group        -         -  1     3.1604  0.0786            -
3        y Group = B   0.9577    0.0419 97    22.8640  0.0000            - ***
4        y Group = A    1.063    0.0419 97    25.3784  0.0000            - ***


summary(psem_model)
```

The significance test for the effect of **Group**

marginal means

pairwise Tukey tests

Provides a measure of whether the path between the exogenous categorical variable and the response is significant as well as parameters for each level in the form of the model-estimated marginal means.

Categorical Data in Piecewise SEM

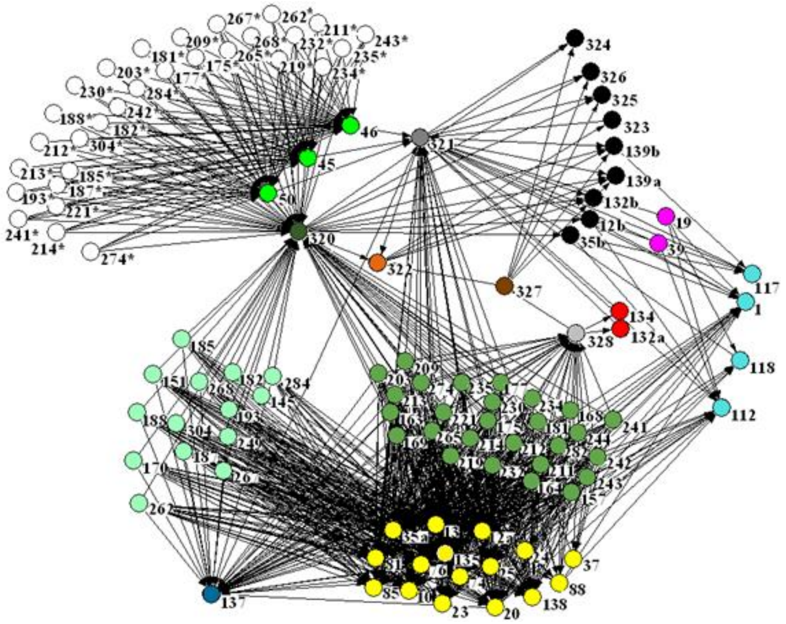
Categorical Variables	**Exogenous Categorical Variables**	**Endogenous Categorical Variables**

Binary variables
yes/no;
presence/absence;
failure/success;
dead/alive;
male/female
	1. Treat as numeric: set the values as 0 or 1 and model as numeric (yields a single path coefficient).	Endogenous categorical variables are not implemented in **`piecewiseSEM`**.
		Treat binary and ordinal variables as numerical (follow step 1 shown for 'Endogenous Categorical Variables')

	2. Create separate dummy variables for each factor levels with values 0, 1 each. **Rule**: for the factor with k levels use k-1 dummy variables (to avoid singularity).	
	3. Use as categorical variable (Marginal Means approach)	
Ordinal variables: small < medium < large; yang < middle < old	1. Treat as numeric: set the values depending on the order of the factor, e.g., small = 1 < medium = 2 < large = 3, and then model as numeric.	
	2. Create separate dummy variables for each factor levels with values 0, 1 each. **Rule**: for the factor with k levels use k-1 dummy variables (to avoid singularity).	
	3. Use as categorical variable (Marginal Means approach)	
Nominal variables study sites ;countries; sampling campaigns	Use as categorical variable (Marginal Means approach)	Nominal endogenous categorical variables are not implemented in **`piecewiseSEM`**

Day 8 Task 2

Effects of land use on arthropod food webs in grasslands



Food web length (1,2,3)

Net sampling of arthropods in grasslands

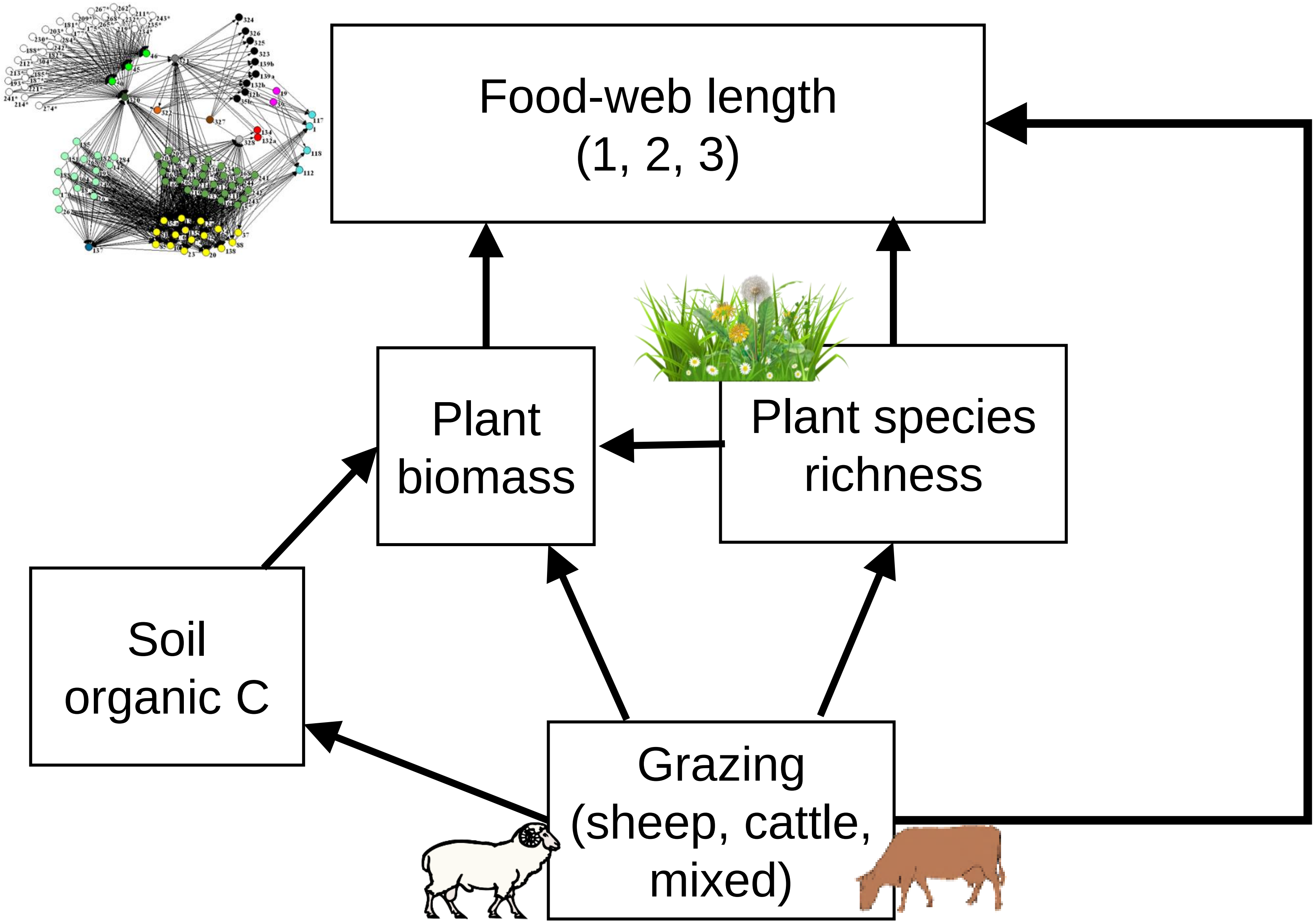
235 grasslands

Grazing type
(“sheep”, “cattle”, or “mixed grazing”)

```
read.csv("Data/Food_web_data_2.csv")

'data.frame': 235 obs. of  5 variables:
 $ Gr_type    : chr  "sheep" "sheep" "sheep" ...
 $ soil_C     : num  1.336 1.631 1.577 ...
 $ plant_sr   : num  5.07 28.39 24.52 ...
 $ plant_biom: num  185 207 224 238 203 ...
 $ FW.length  : int  1 1 1 1 1 1 1 1 1 1 ...
```

Day 8 Task 2

Effects of land use on food webs in grasslands



Food-web length
(1, 2, 3)

Plant biomass

Plant species richness

Soil organic C

Grazing
(sheep, cattle, mixed)

Day 8 Task 2

Effects of land use on food webs in grasslands

`Gr_type` (grazing type) is your exogenous nominal categorical variables. Test **`Gr_type`** (as a part of the SEM model on fig. 1) using marginal means in ***`piecewiseSEM`***.

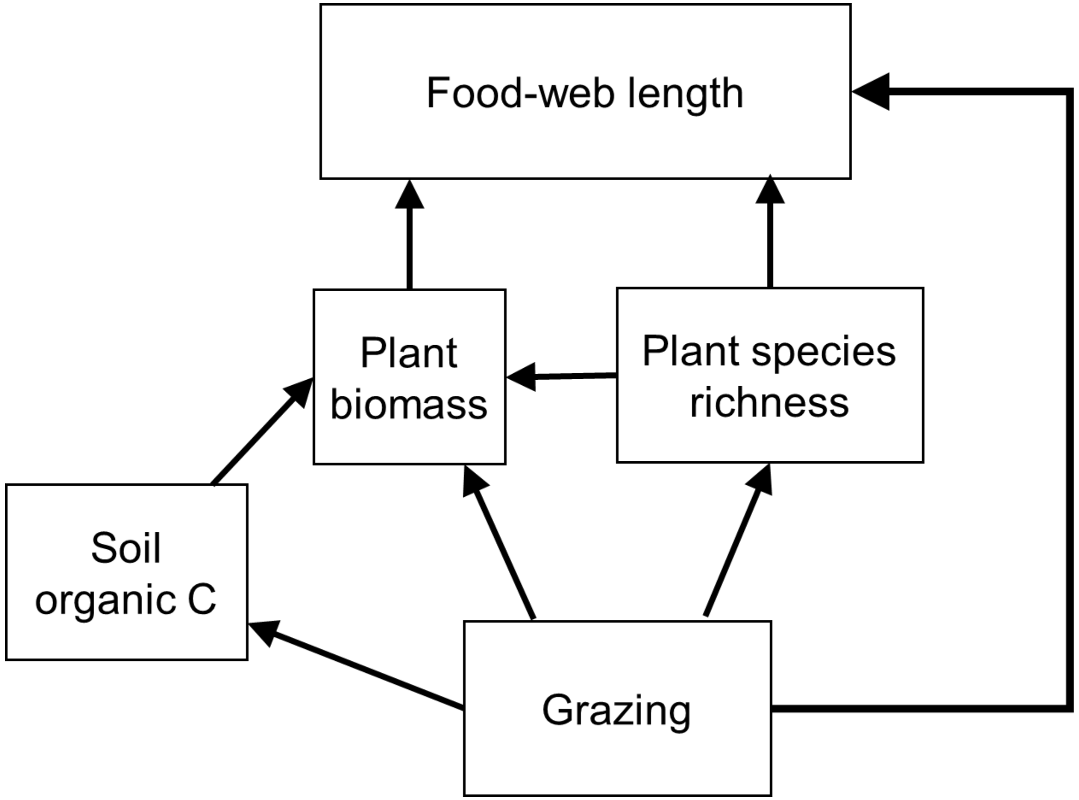


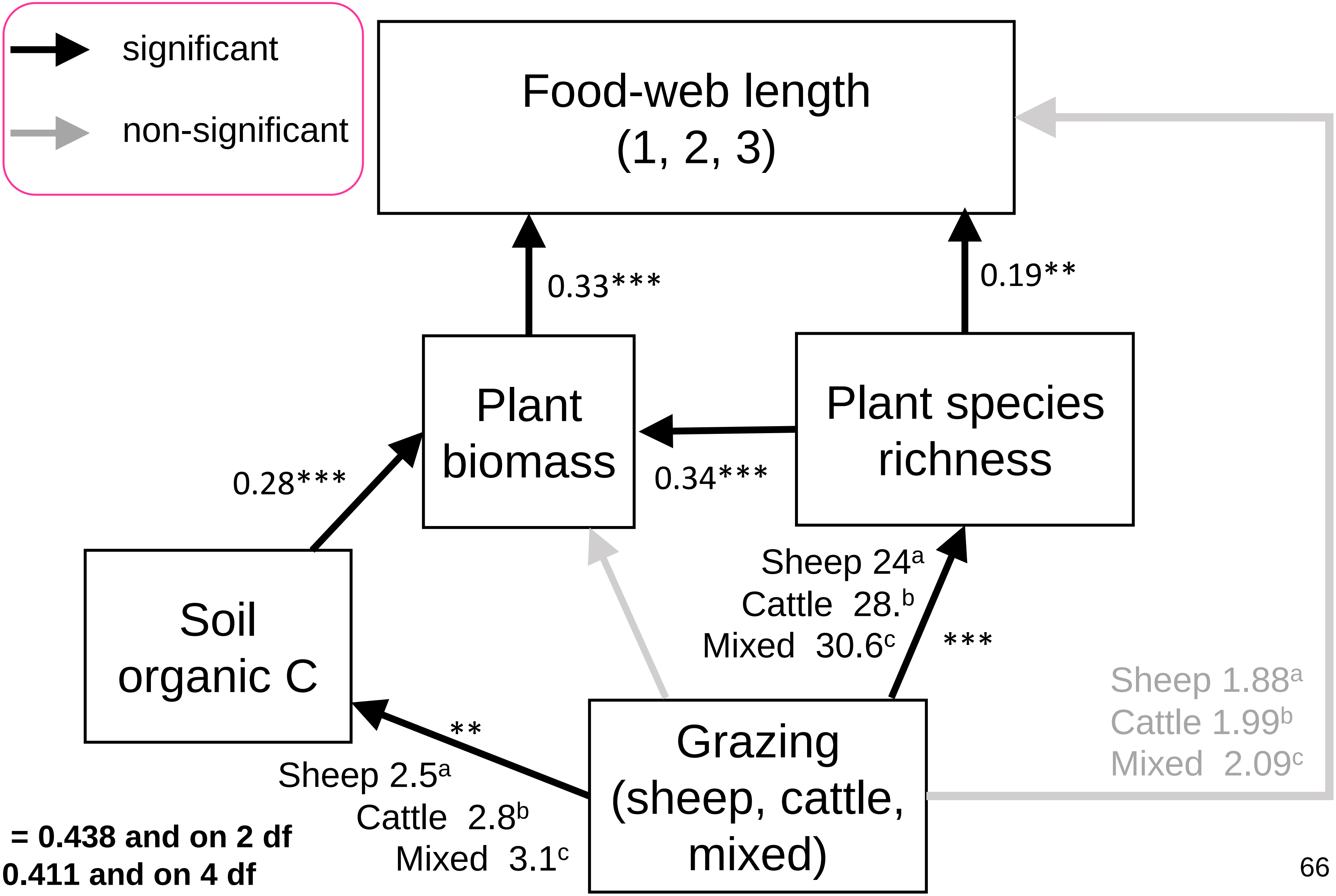
fig. 1

Day 8 Task 2

Solution

Results:

- Direct effect of grazing on food-web length is not significant.
- effects of grazing on food-web length significantly differed among grazing types: length was the largest in mixed grazing and the lowest in sheep grazing (see marginal means).
- Grazing affects food-web length indirectly by altering plant diversity and soil C.



Chi-Squared = 1.653 with P-value = 0.438 and on 2 df
Fisher's C = 3.965 with P-value = 0.411 and on 4 df

Day 8 – Part 2

Outline

- Interactions in SEM
  - ✓ Single Interactions in sub-models
  - ✓ Multigroup Analysis (model-wide interactions)

Day 8 – Part 2

Outline

- Interactions in SEM

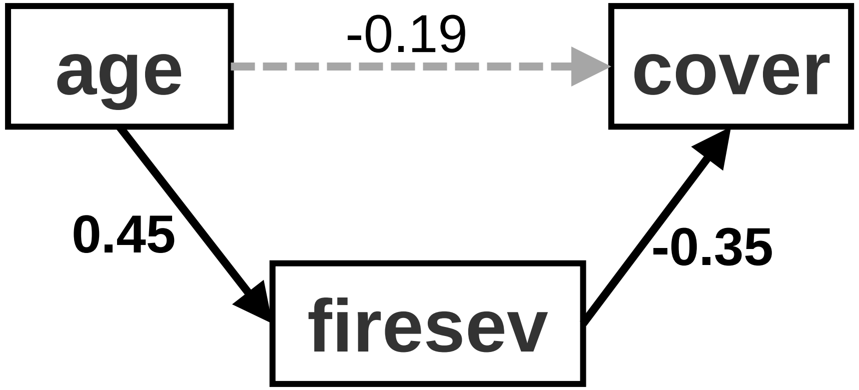
  ✓ **Single Interactions in sub-models**

  ✓ Multigroup Analysis (model-wide interactions)

Interactions

- reproduces a single variance-covariance matrix

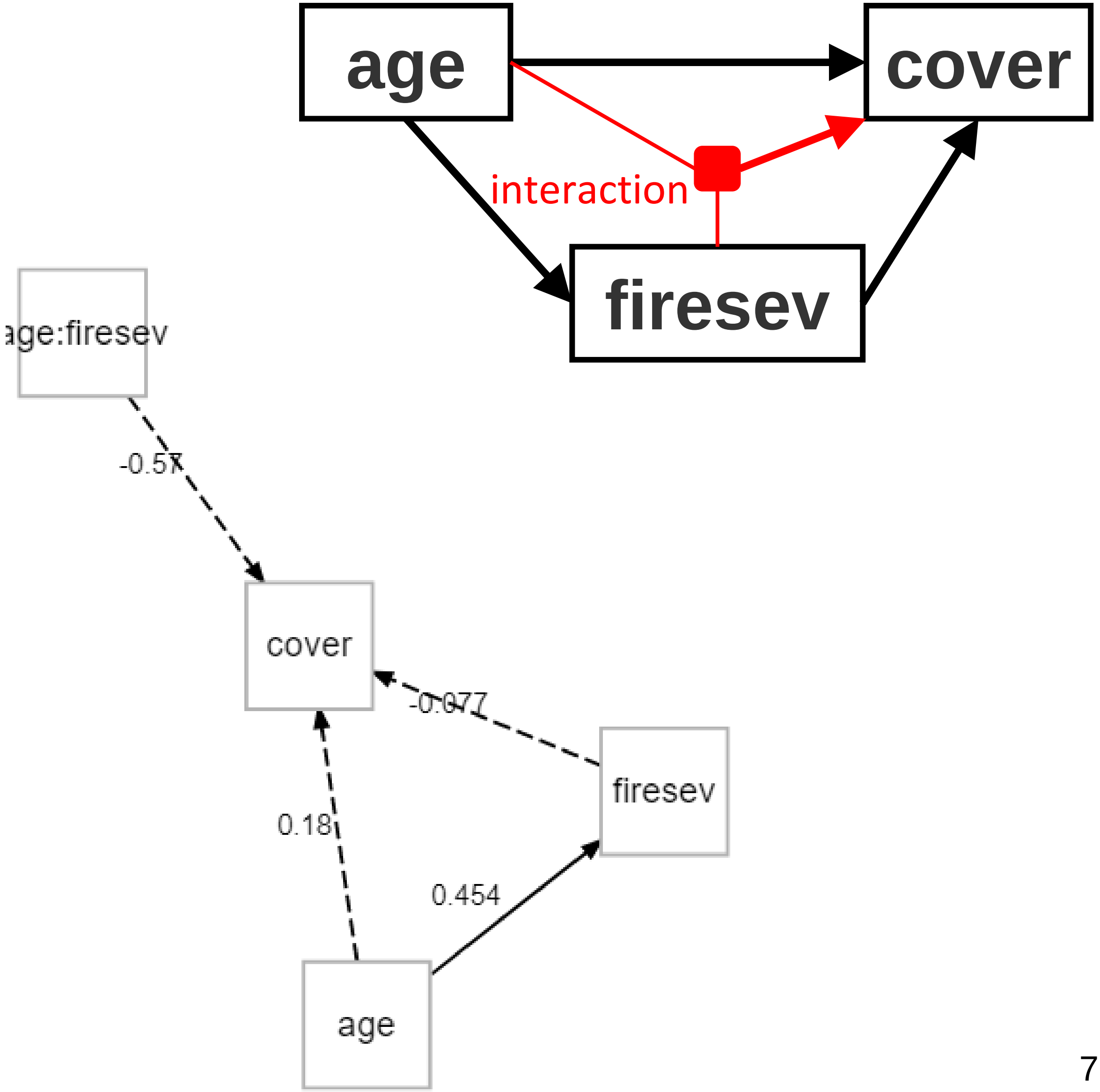
```
# Keeley data
library(piecewiseSEM)
data(keeley)
```



Data: Grace, J.B. and Keeley, J.E. 2006. A structural equation model analysis of postfire plant diversity in California shrublands. Ecological Applications 16:503-514

Interactions

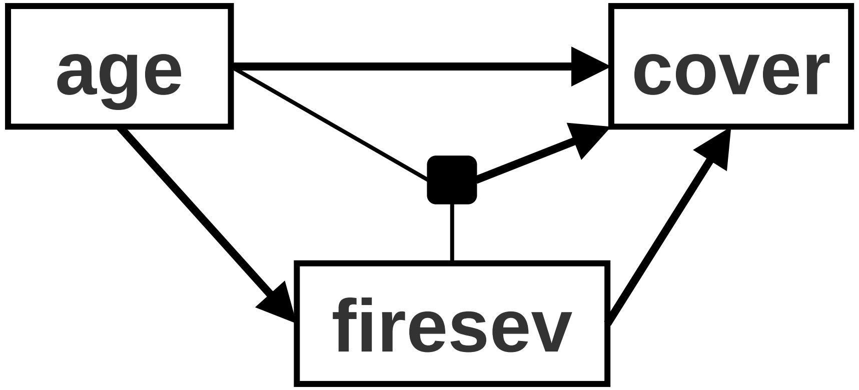
```
library(piecewiseSEM)

data(keeley)

psem_m1 <- psem(
  lm(cover ~ age*firesev, data = keeley),
  lm(firesev ~ age, data = keeley)
)

plot(psem_m1)
```



Does the effect of fire severity on plant cover depends on plant age?

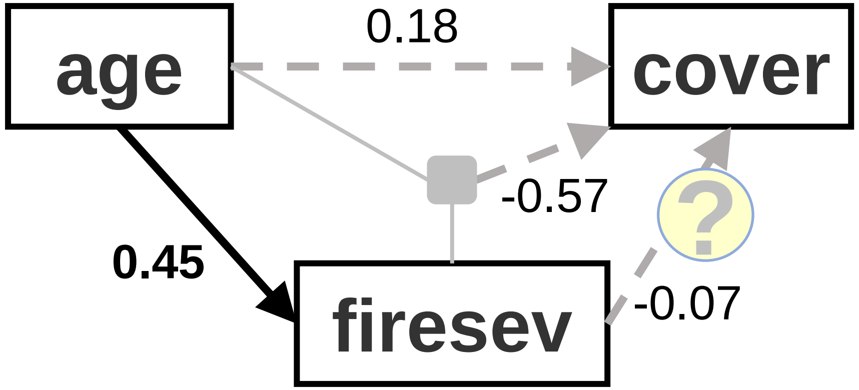
Interactions

```
psem_m1 <- psem(
  lm(cover ~ age*firesev, data = keeley),
  lm(firesev ~ age, data = keeley))

fisherC(psem_m1)# This is saturated model (no DF)
LLchisq(psem_m1)

> Fisher.C df P.Value
1  NA   0      NA
> Chisq df P.Value
1     0  0        1
```



Interactions

```
psem_m1 <- psem(
  lm(cover ~ age*firesev, data = keeley),
  lm(firesev ~ age, data = keeley))

fisherC(psem_m1)# This is saturated model (no DF)
LLchisq(psem_m1)

> Fisher.C df P.Value
1  NA   0       NA
> Chisq df P.Value
1      0  0         1
```



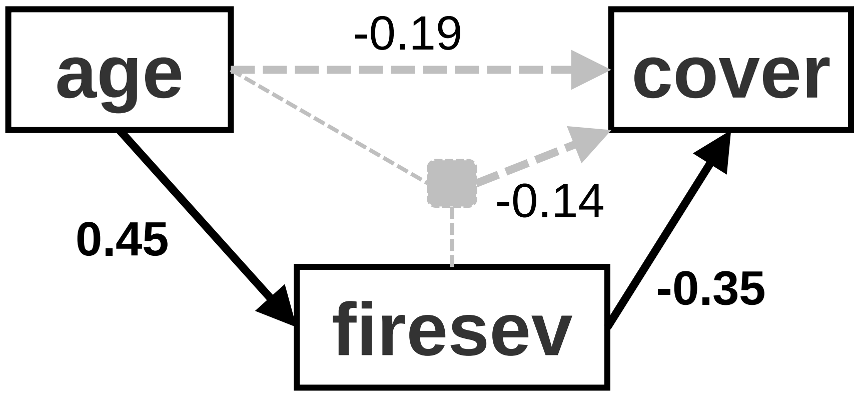
```
coefs(psem_m1)
>
 Response   Predictor Estimate Std.Error DF Crit.Value P.Value Std.Estimate
1    cover         age   0.0045    0.0067 86     0.6786  0.4992       0.1800
2    cover     firesev  -0.0149    0.0398 86    -0.3729  0.7101      -0.0774
3    cover age:firesev  -0.0021    0.0014 86    -1.5263  0.1306      -0.5700
4  firesev         age   0.0597    0.0125 88     4.7781  0.0000       0.4539 ***


 with(keeley, cor(age, age*firesev))
 >
 [1] 0.8687952  # we have colinearity
```

Centering data helps remove collinearities.

Interactions

Centering data helps remove collinearities.

```
# Centering for Interactions

data2 <- data.frame(age_c=scale(keeley$age, scale=FALSE),
                    firesev_c=scale(keeley$firesev, scale=FALSE),
                      cover=keeley$cover)



with(data2, cor(age_c, age_c*firesev_c))
>
[1] -0.06792114
```

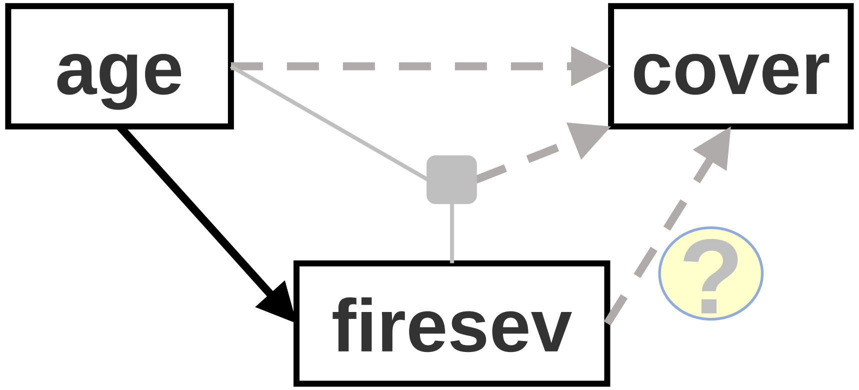


```
psem_m2 <- psem(
  lm(cover ~ age_c*firesev_c, data = data2),
  lm(firesev_c ~ age_c, data = data2))



> coefs(psem_m2)
   Response         Predictor Estimate Std.Error DF Crit.Value P.Value Std.Estimate
1     cover             age_c  -0.0050    0.0027 86    -1.8810  0.0634      -0.1985
2     cover         firesev_c  -0.0684    0.0203 86    -3.3752  0.0011      -0.3561  **
3     cover age_c:firesev_c  -0.0021    0.0014 86    -1.5263  0.1306      -0.1438
4 firesev_c             age_c   0.0597    0.0125 88     4.7781  0.0000       0.4539 ***
```

lavaan

Interactions

```
# Interactions using lavaan

library(lavaan)
sem_m1 <-'
  firesev ~ age
  cover ~ firesev + age + firesev:age
'
sem_fit1 <- sem(sem_m1, data=keeley)
```



```
summary(sem_fit2, standardize = T)

Regressions:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
  firesev ~
    age               0.060    0.012    4.832    0.000    0.060    0.454
  cover ~
    firesev          -0.015    0.020   -0.751    0.453   -0.015   -0.079
    age               0.005    0.005    0.941    0.347    0.005    0.183
    firesev:age      -0.002    0.001   -3.073    0.002   -0.002   -0.580
```

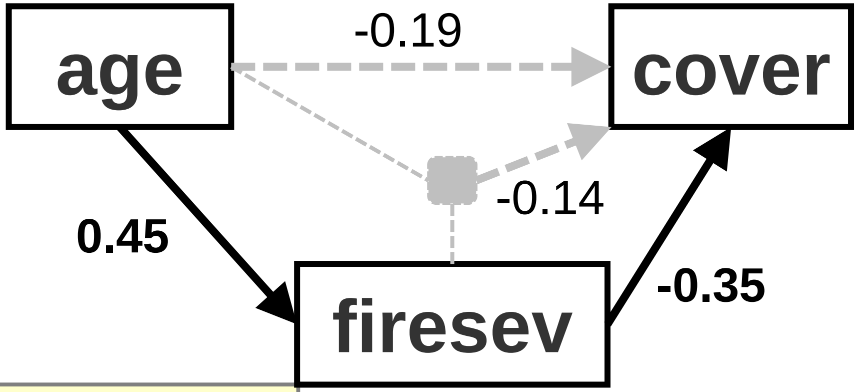
Interactions

lavaan

```
# use centered data

sem_m2<-'
  firesev_c ~ age_c
  cover ~ firesev_c + age_c + firesev_c:age_c
'
sem_fit2 <- sem(sem_m2, data=data2)
```

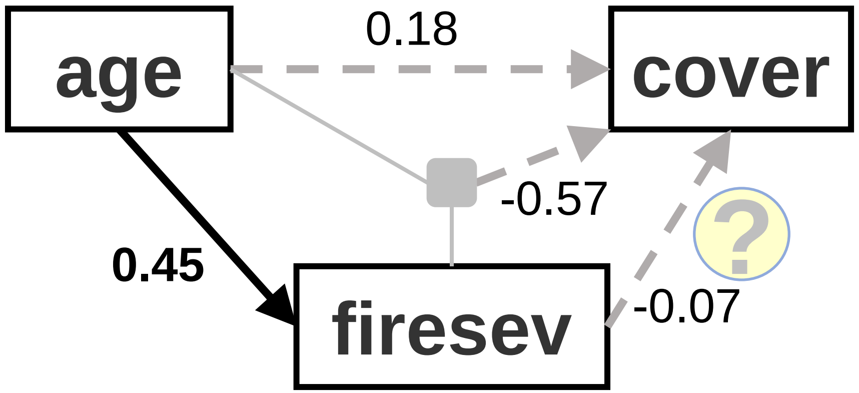
```
summary(sem_fit2, standardize = T)

Regressions:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
  firesev_c ~
    age_c             0.060    0.012    4.832    0.000    0.060    0.454
  cover ~
    firesev_c        -0.068    0.020   -3.455    0.001   -0.068   -0.356
    age_c            -0.005    0.003   -1.923    0.055   -0.005   -0.198
    firesev_c:ag_c   -0.002    0.001   -1.562    0.118   -0.002   -0.144
```

piecewiseSEM



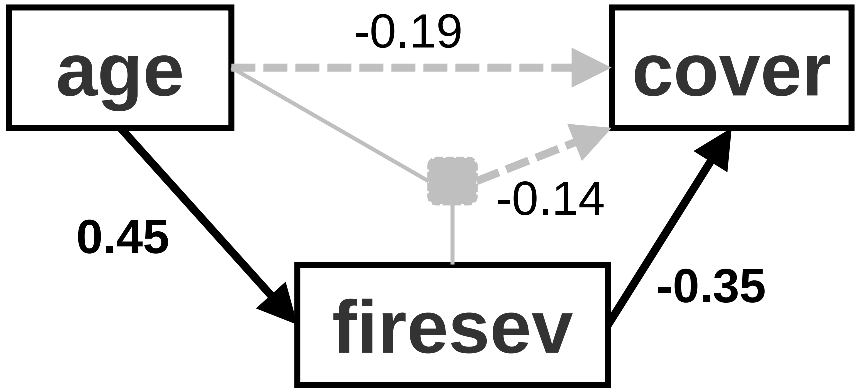
Interactions

Uncentered



Additive paths estimate the effect of one variable in the absence of the other.

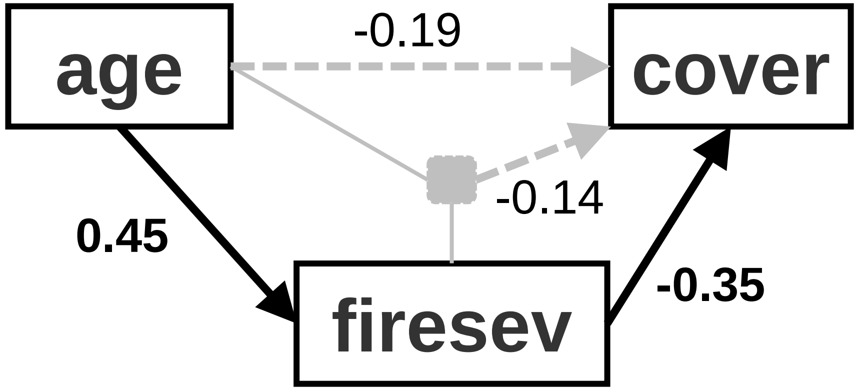
Centered



Additive paths estimate the effect of one variable at the average level of the other

Fire severity has no effect when age = 0 versus fire severity has an effect at the mean level of age.

Interactions



Result for interaction: the effect of fire severity on plant cover does not differ with plant age.

Fire severity is expected to have the same magnitude of effect regardless of what is the age of plant.

Day 8 Task 3

- **Topic**: Consequences of grassland abandonment
- **Background:** 94 semi-natural grasslands were sampled along elevation gradient in the Carpathians. Study grasslands differed in the land-use type. Tree and shrub encroachment was measured as the rate of tree and shrub cover change per year.
- **Tasks:** Fit the hypothetical model, test if there are messing links. Test if and how the effects of land use on plant species richness differ among the types of land use. Interpret the model results.
- **Dataset:** `dataset4_grassland_abandonment.csv`
  - Variables:
    - `LU_type` … Land use type (abandoned, low-intensity mowing, low-intensity grazing)
    - `Elev`… elevation
    - `Encroch_rate` … woody encroachment rate (per year),
    - `Soil_C` … soil organic carbon content
    - `SR` … number of species of herbaceous plants

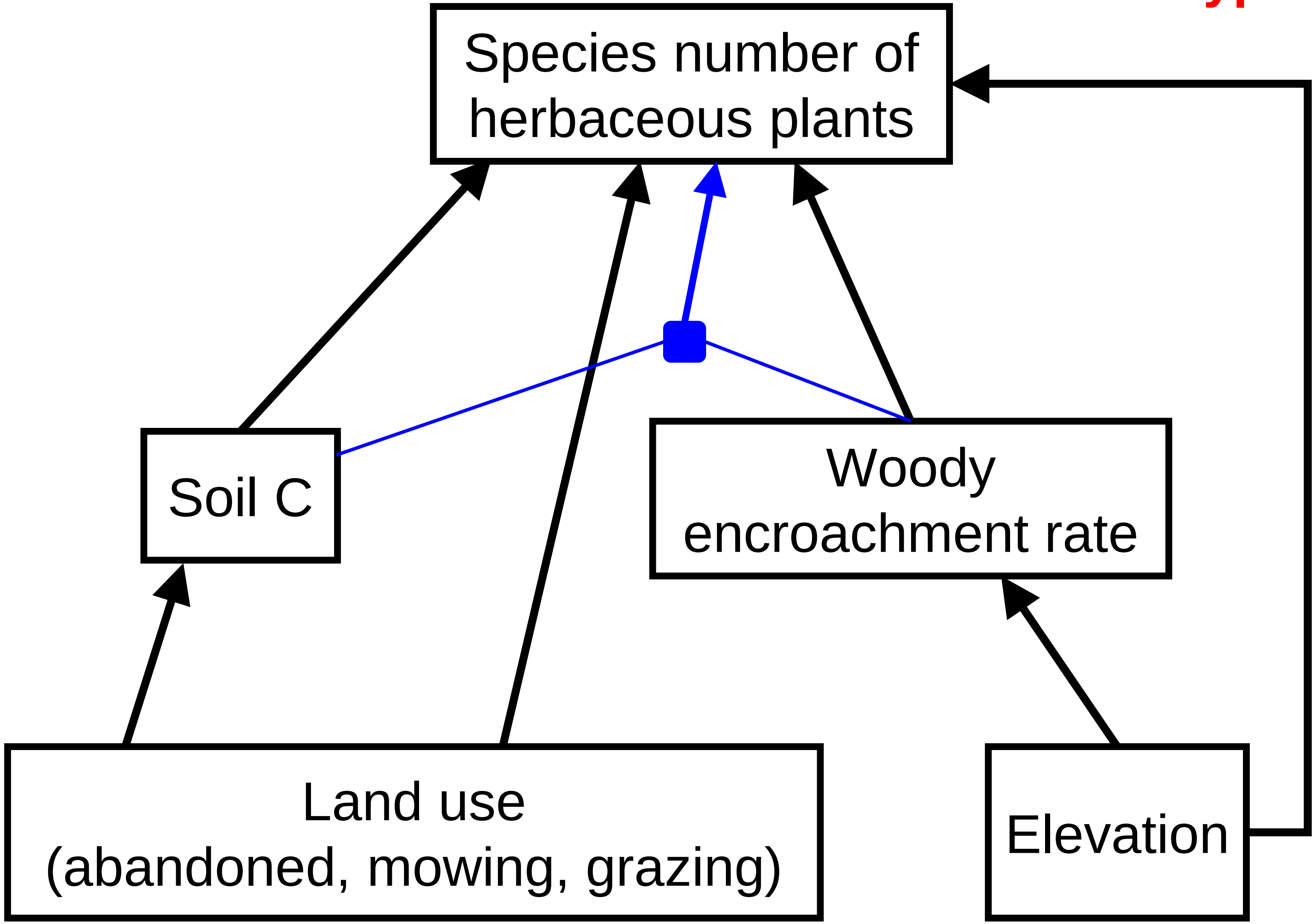
July 5, 2006

July 15, 2016

Tree and shrub encroachment in seminatural grasslands in the Carpathians

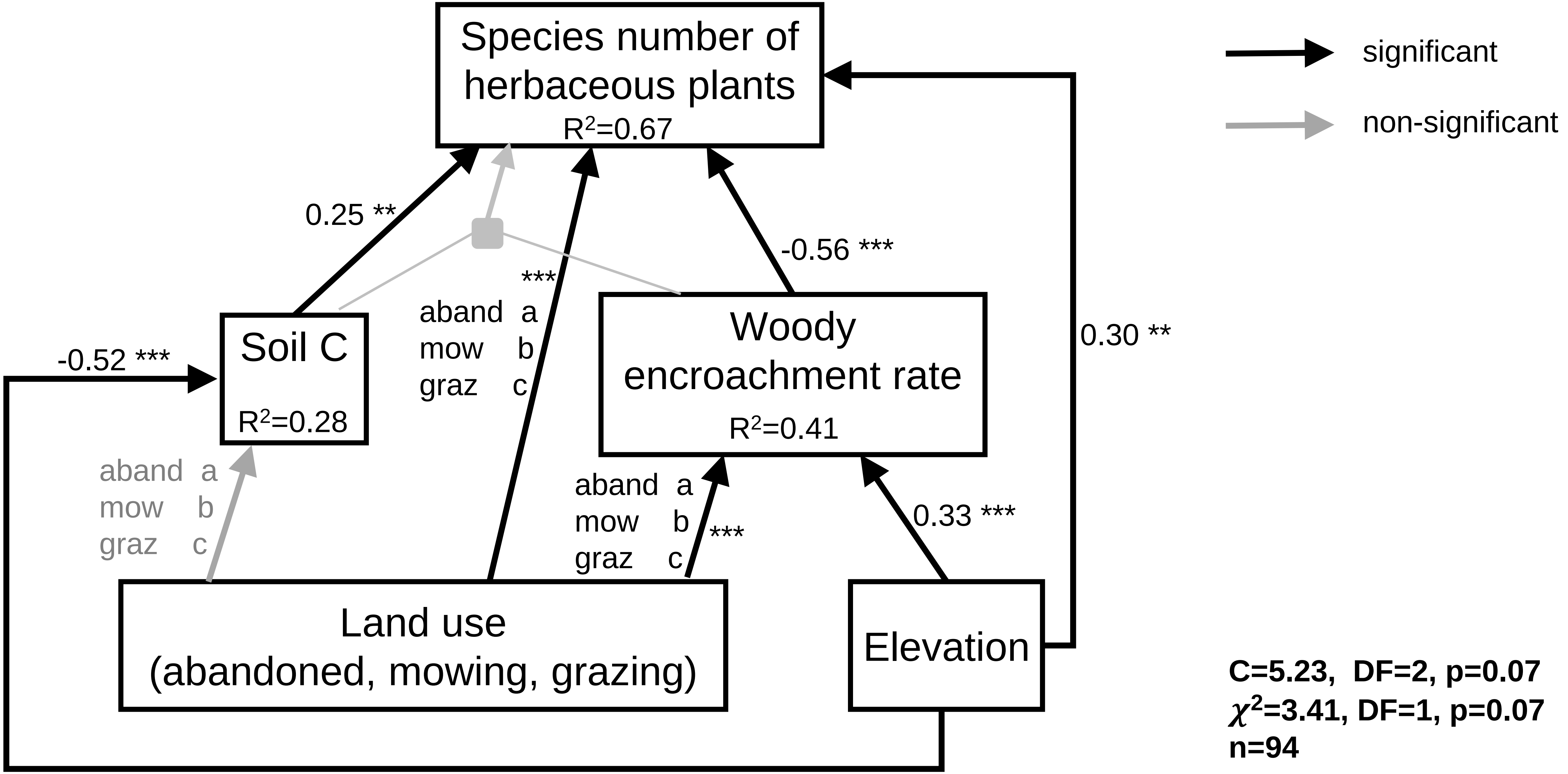
Day 8 Task 3

Hypothetical SEM model


Species number of herbaceous plants
Soil C
Woody encroachment rate
Land use (abandoned, mowing, grazing)
Elevation

Day 8 Task 3

Solution



C=5.23, DF=2, p=0.07
χ^2=3.41, DF=1, p=0.07
n=94

Day 8 – Part 3

Outline

- Interactions in SEM

  ✓ Single Interactions in sub-models

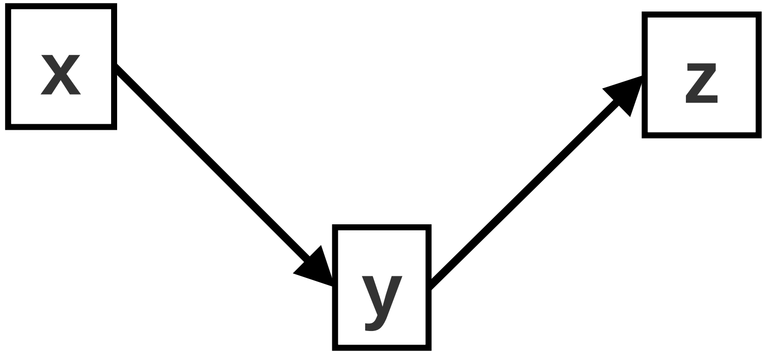
  ✓ **Multigroup Analysis (model-wide interactions)**

Multigroup Analysis

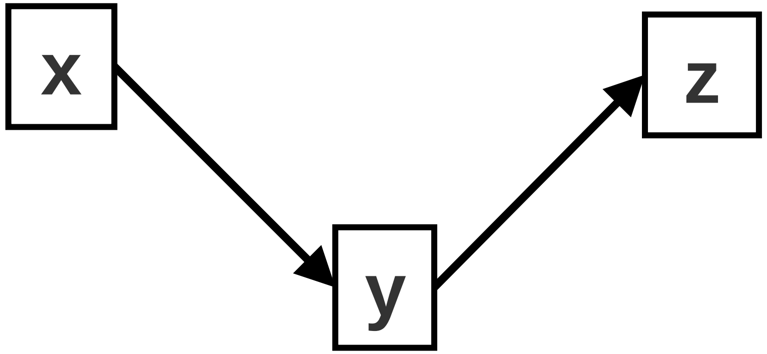
A **multigroup** model is a “model-wide” interaction:

- Instead of focusing on a single response, the interaction is applied across the entire SEM model;
- It asks if not just one but *all* coefficients are the same or different across groups;
- It identifies which paths have the same effect across groups and which paths vary depending on the group.

Group A



Group B



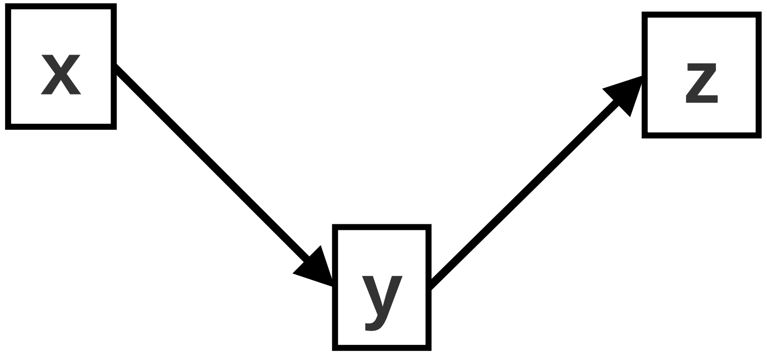
Global vs. Local Estimations

Properties:	**Covariance-based SEM**	**Piecewise SEM**
Interactions (test whether effect of one variable vary with another variable)	Allows testing interactions	Allows testing interactions
Multigroup Analysis (test whether some or all effects in the SEM vary by the grouping variable)	back-and-forth manual process of relaxing and constraining paths in ***lavaan***	***piecewiseSEM*** tests constraints and automatically selects the best output for the data

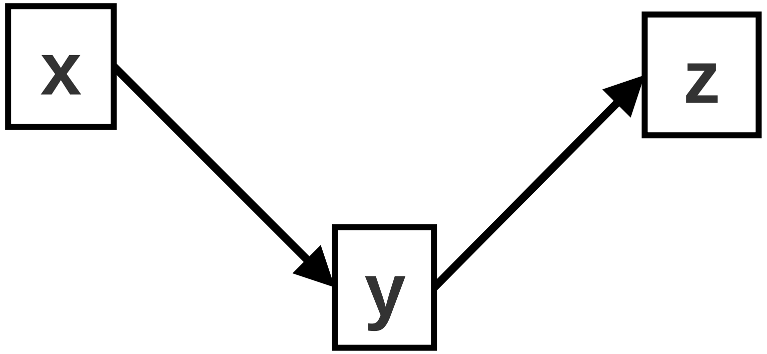
Multigroup Analysis

```
data3 <- read.csv("Data/SEMdata2.csv")

anova(lm(y ~ x * Group, data3))
>
Response: y
          Df Sum Sq Mean Sq F value    Pr(>F)
x          1 8.2740  8.2740 97.9518 2.475e-16 ***
Group      1 0.2772  0.2772  3.2811   0.07321 .
x:Group    1 0.3974  0.3974  4.7051   0.03254 *
Residuals 96 8.1091  0.0845


anova(lm(z ~ y * Group, data3))
>
Response: z
          Df  Sum Sq Mean Sq  F value Pr(>F)
y          1 15.8899 15.8899 176.3764 <2e-16 ***
Group      1  0.0366  0.0366   0.4066 0.5252
y:Group    1  0.1271  0.1271   1.4107 0.2379
Residuals 96  8.6487  0.0901
```

Group A



Group B



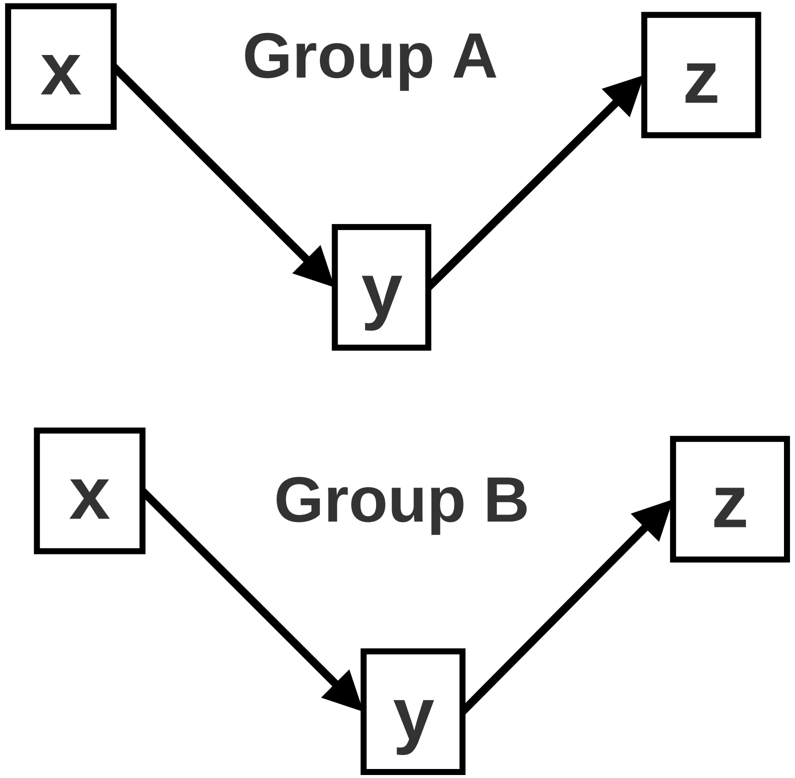
Multigroup Analysis

```
library(piecewiseSEM)

psem_model <- psem(
  lm(y ~ x, data3),
  lm(z ~ y, data3))

multigroup(psem_model, group = "Group")
```

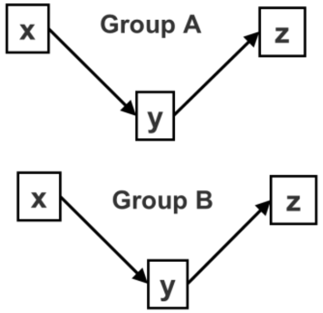
Interaction variable

```
>
Groups = Group [ A, B ]
---
Global goodness-of-fit:

  Fisher's C = 0.301 with P-value = 0.86 and on 2 degrees
of freedom
---
Model-wide Interactions:

  Response Predictor Test.Stat DF P.Value
         y   x:Group       8.3  1  0.0325 *
         z   y:Group      15.5  1  0.2379

 y -> z constrained to the global model
---
```

estimates the effect of y on z using the entire dataset and report that single constrained coefficient across all groups



```
>
...
Group [A] coefficients:

  Response Predictor Estimate Std.Error DF Crit.Value P.Value Std.Estimate
         y         x   0.7712    0.1662 48     4.6387       0       0.5563 ***
         z         y   0.9652    0.0726 98    13.2931       0       0.6895 *** c

Group [B] coefficients:

  Response Predictor Estimate Std.Error DF Crit.Value P.Value Std.Estimate
         y         x   1.2404    0.1379 48     8.9963       0       0.7923 ***
         z         y   0.9652    0.0726 98    13.2931       0       0.8914 *** c

Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05    c = constrained
```



Day 8 Task 4

The Effects of Grazing on Finnish Coastal Meadows

Photo: Jorma Pessa

```
# meadow data
library(piecewiseSEM)
data(meadows)

> str(meadows)
'data.frame': 354 obs. of  4 variables:
 $ grazed: int  1 1 1 1 1 1 1 1 1 1 ...
 $ mass  : num  461.6 704.8 1278.8 76.4 ...
 $ elev  : num  19.2 18.8 12.5 38.2 37.4 ...
 $ rich  : int  6 2 1 6 14 12 6 10 10 17 ...
```

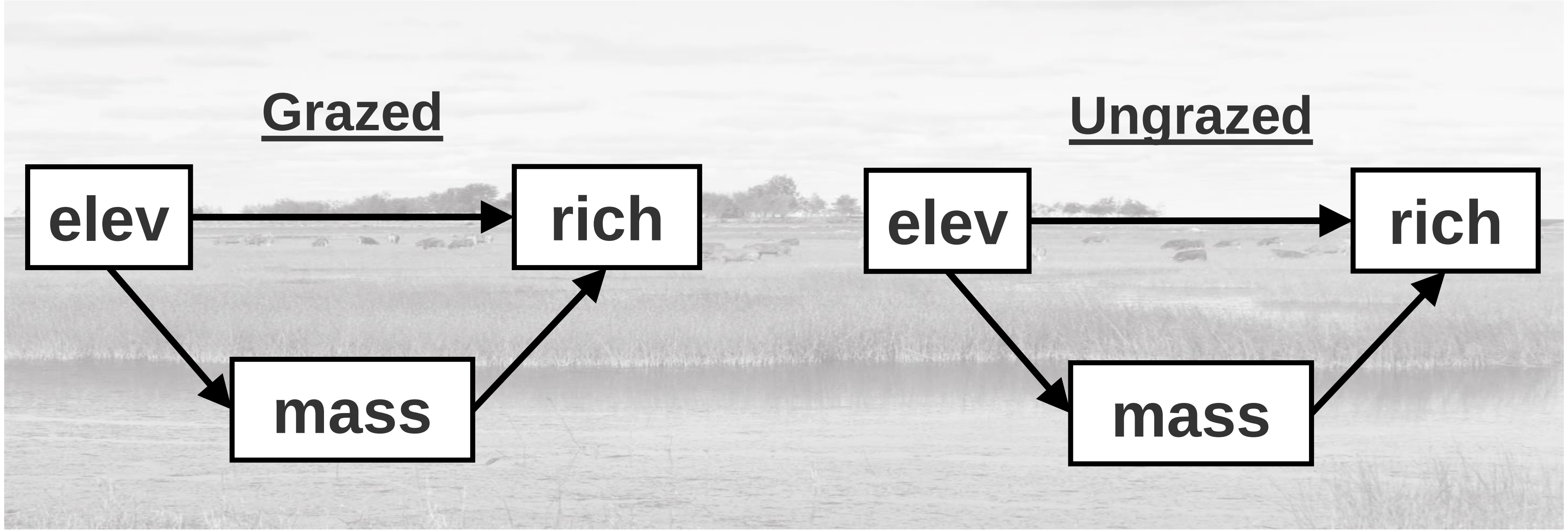
grazed is our grouping variable

Data:

Jutila, H. (1997) Vascular plant species richness in grazed and ungrazed coastal meadows, SW Finland. - Ann. Bot. Fenn. 34:245-263.

Grace, J.B. and Jutila, H. (1999) The relationship between species density and community biomass in grazed and ungrazed coastal meadows. Oikos, 85:398-408.

Hypothetical model



Task:

Perform the multigroup analysis for this hypothetical SEM model using *piecewiseSEM*

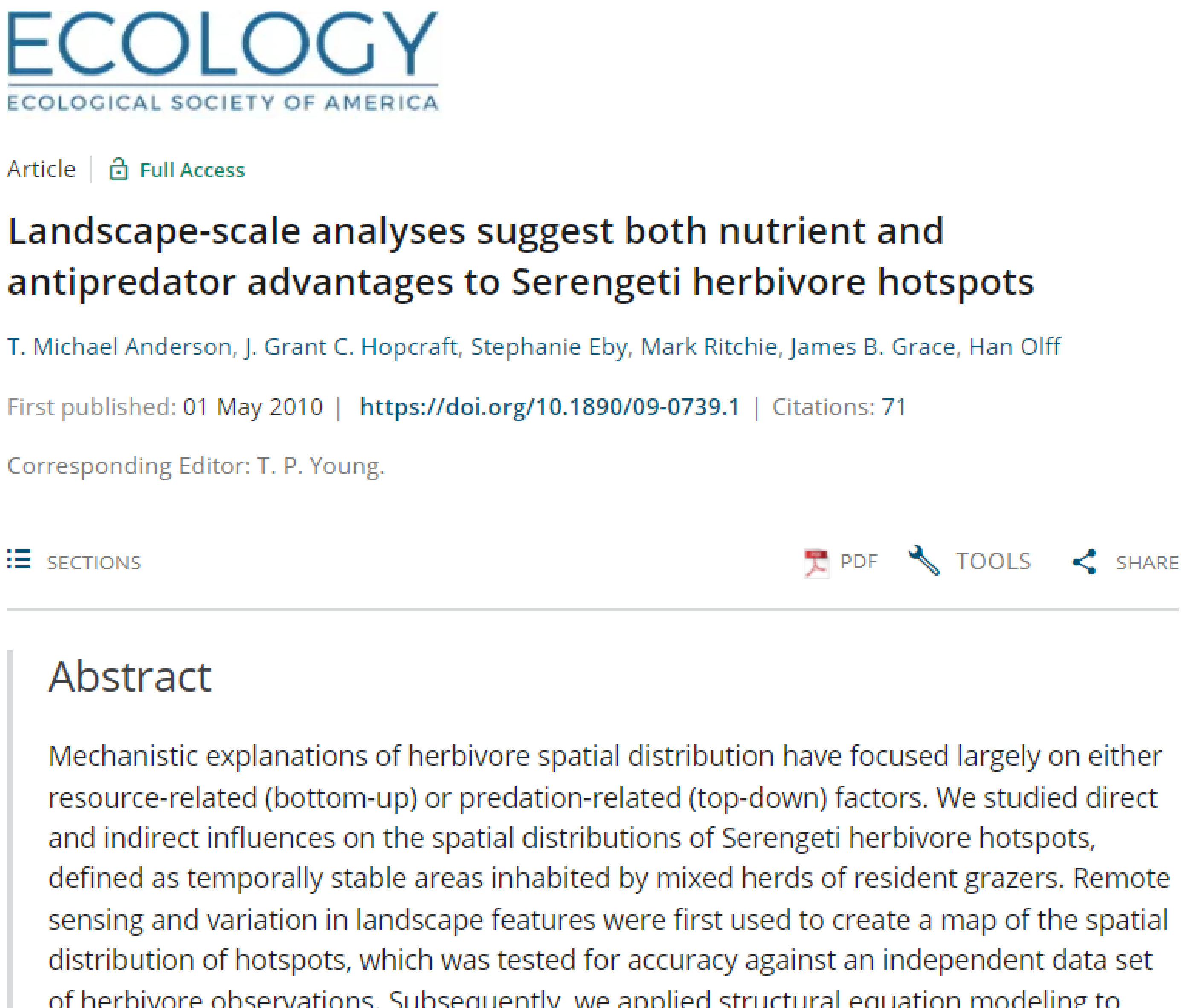
Day 8 – Part 3

Outline

- Extensions to GLM, LMM, and GLMM

GLM in SEM

Example



ECOLOGY
ECOLOGICAL SOCIETY OF AMERICA

Article | Full Access

Landscape-scale analyses suggest both nutrient and antipredator advantages to Serengeti herbivore hotspots

T. Michael Anderson, J. Grant C. Hopcraft, Stephanie Eby, Mark Ritchie, James B. Grace, Han Olff

First published: 01 May 2010 | https://doi.org/10.1890/09-0739.1 | Citations: 71

Corresponding Editor: T. P. Young.

Abstract

Mechanistic explanations of herbivore spatial distribution have focused largely on either resource-related (bottom-up) or predation-related (top-down) factors. We studied direct and indirect influences on the spatial distributions of Serengeti herbivore hotspots, defined as temporally stable areas inhabited by mixed herds of resident grazers. Remote sensing and variation in landscape features were first used to create a map of the spatial distribution of hotspots, which was tested for accuracy against an independent data set of herbivore observations. Subsequently, we applied structural equation modeling to

GLM in SEM

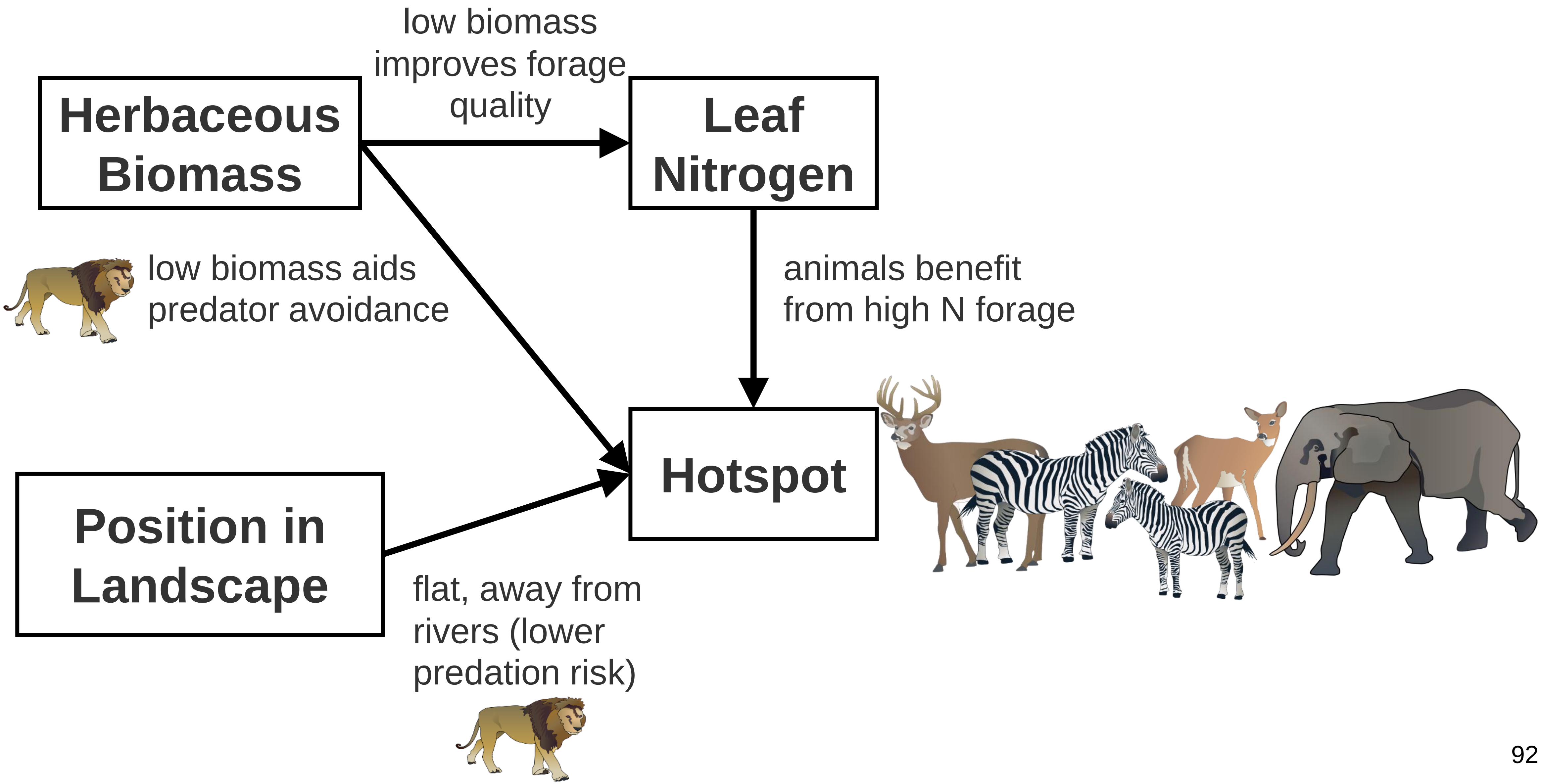
Example

wikipedia.org

Serengeti you think of…

Serengeti you may not think of…

GLM in SEM



Herbaceous Biomass

low biomass improves forage quality

Leaf Nitrogen

low biomass aids predator avoidance

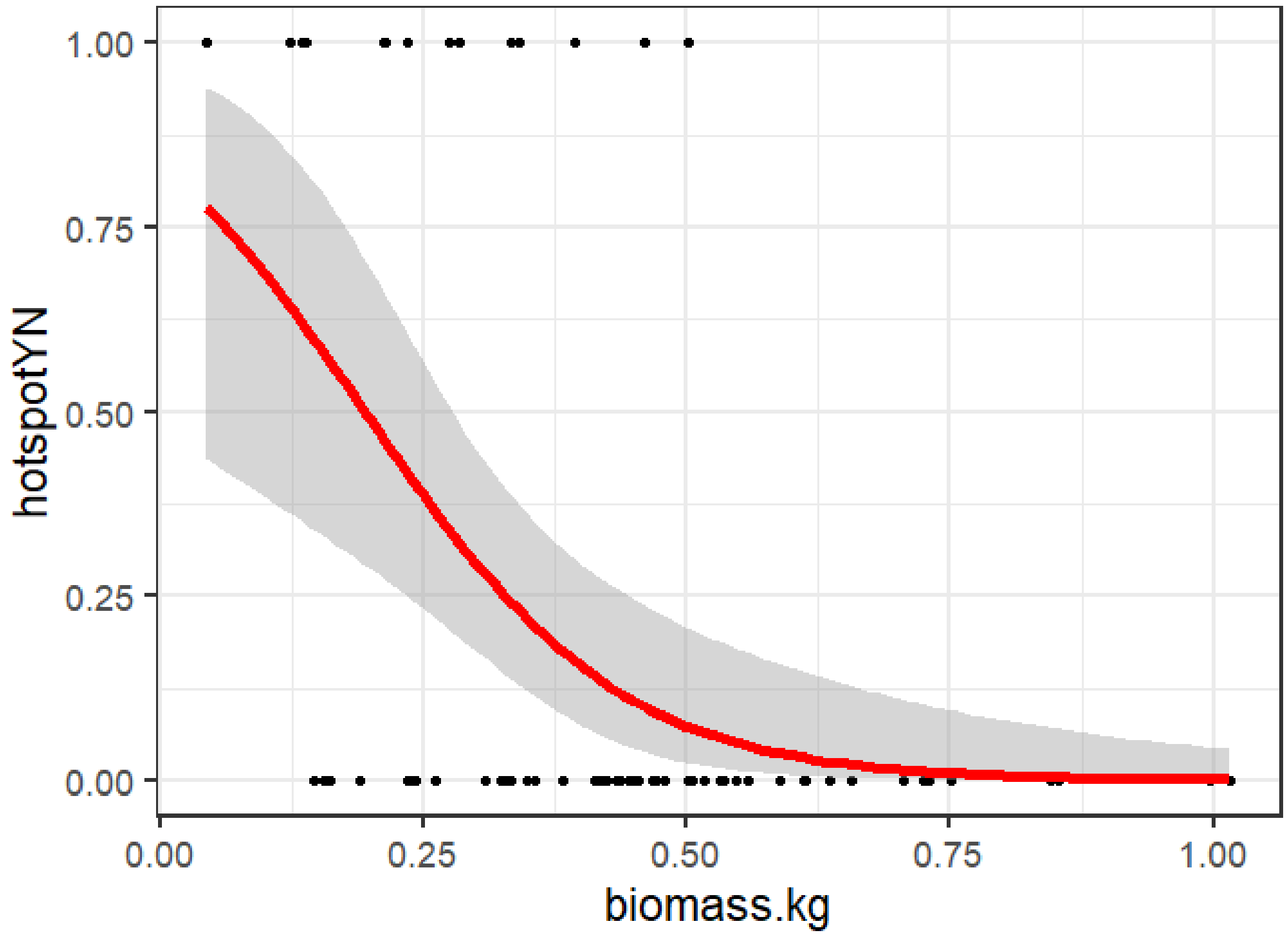
animals benefit from high N forage

Hotspot

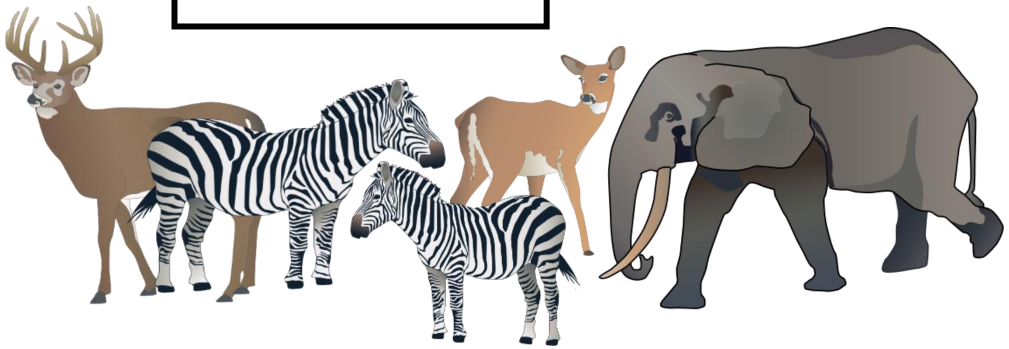
Position in Landscape

flat, away from rivers (lower predation risk)

GLM in SEM

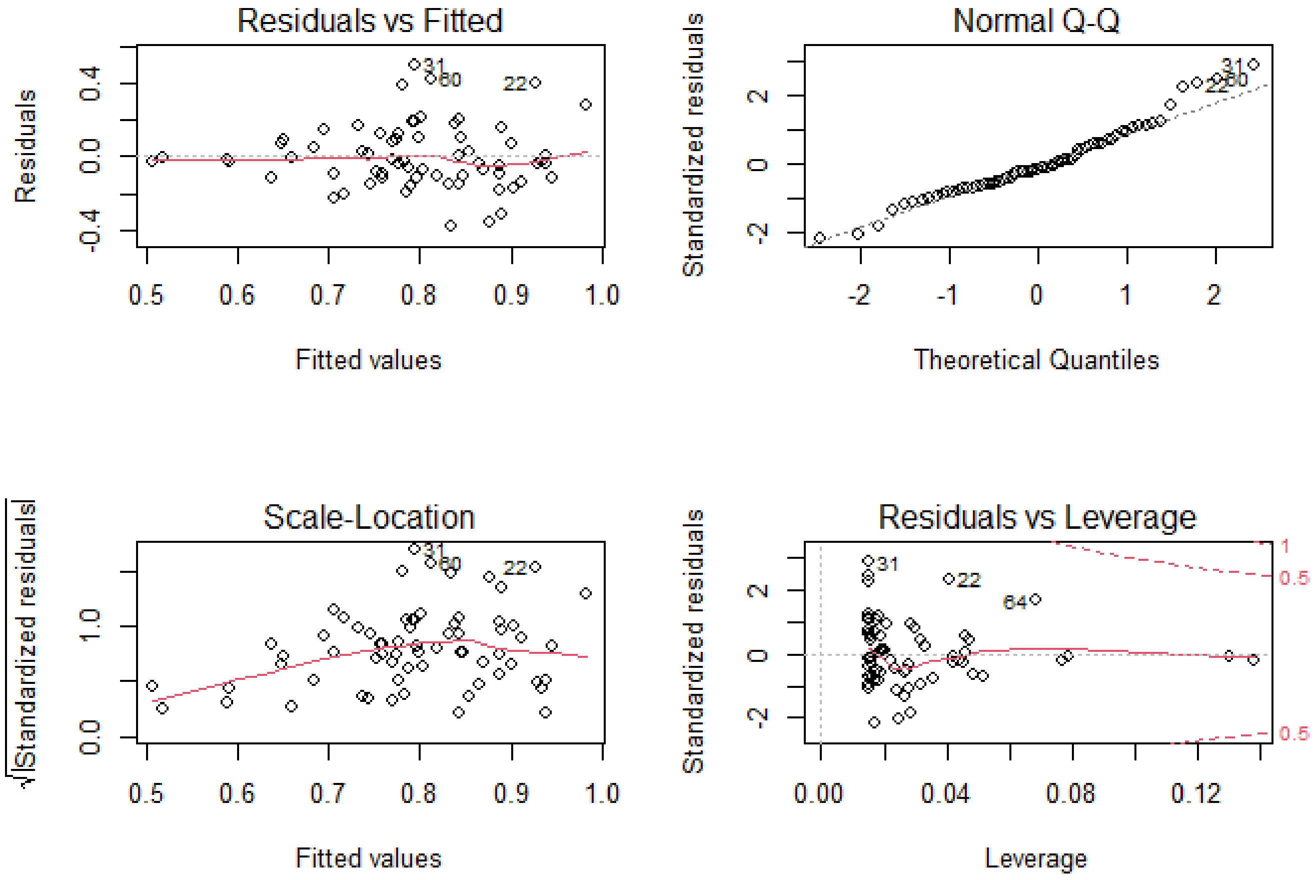


Hotspot



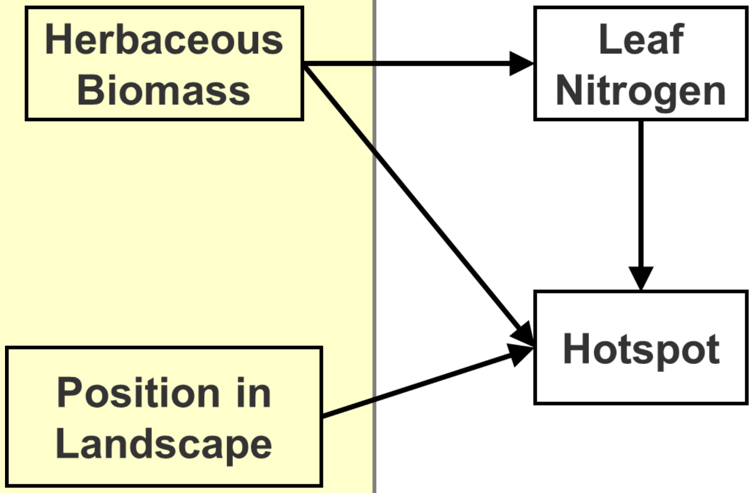
GLM in SEM

```
anderson <- read_csv("Data/Anderson.csv")
str(anderson)
# check assumptions
m1 <- lm(leafN ~ biomass.kg, anderson)
plot(m1)

m2 <- glm(hotspotYN ~ leafN + biomass.kg +
             landscape, family = "binomial",
             data=anderson)
summary(m2) # check overdispersion

anderson.sem <- psem( m1,m2)
summary(anderson.sem)
```



GLM in SEM

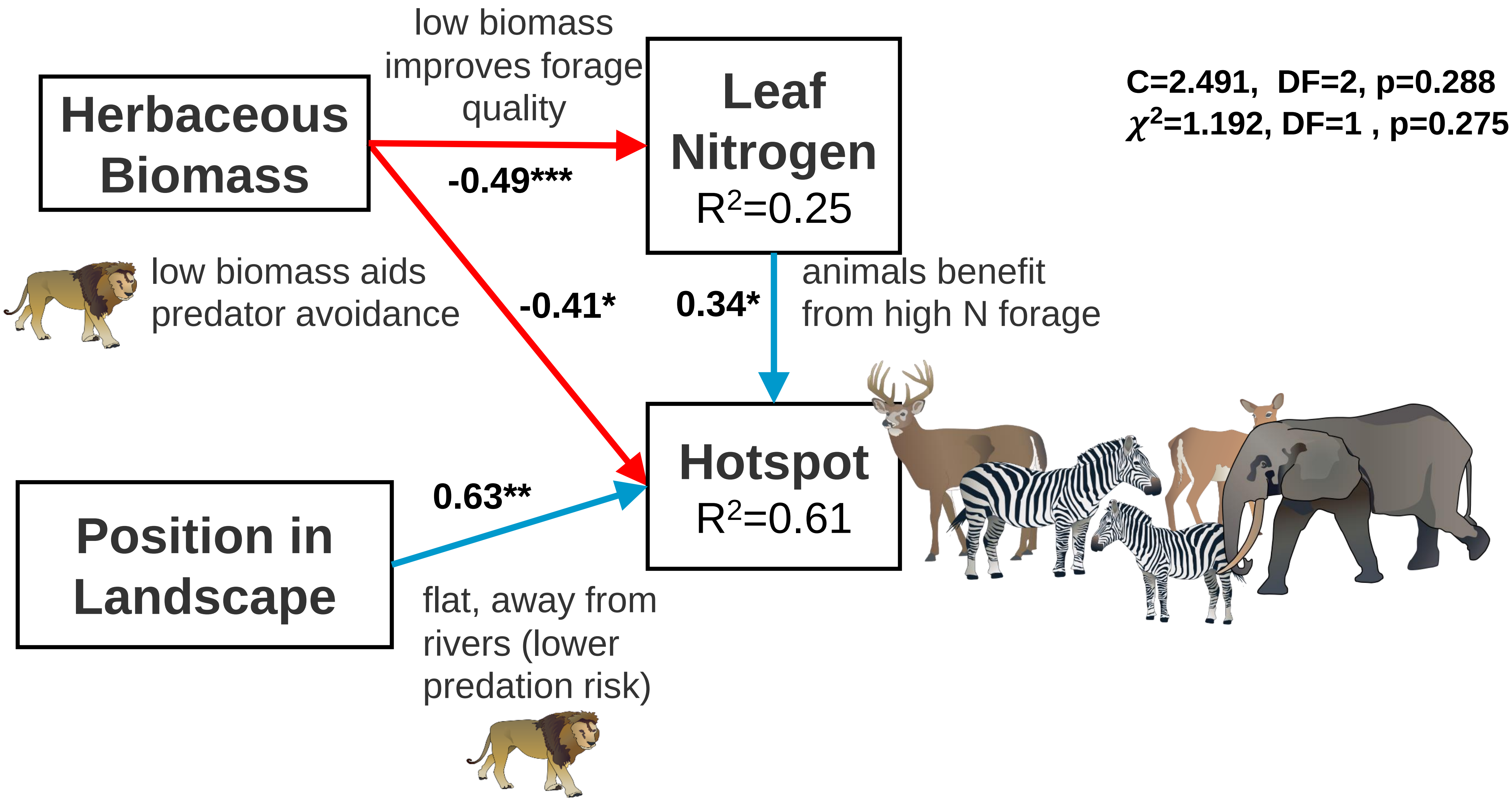
```
> summary(anderson.sem)

Structural Equation Model of anderson.sem
Call:
  leafN ~ biomass.kg
  hotspotYN ~ leafN + biomass.kg + landscape

    AIC
 4.617

---
Tests of directed separation:

           Independ.Claim Test.Type DF Crit.Value P.Value
  leafN ~ landscape + ...      coef 64    -1.0718  0.2878
--
Global goodness-of-fit:

Chi-Squared = 1.192 with P-value = 0.275 and on 1 degrees of freedom
Fisher's C = 2.491 with P-value = 0.288 and on 2 degrees of freedom
```



GLM in SEM

```
> ...
---
Coefficients:

   Response  Predictor Estimate Std.Error DF Crit.Value P.Value Std.Estimate    
      leafN biomass.kg  -0.4880    0.1050 65    -4.6486  0.0000      -0.4995 ***
  hotspotYN      leafN   6.6867    2.7818 63     2.4037  0.0162       0.3399   *
  hotspotYN biomass.kg  -7.7838    3.5694 63    -2.1807  0.0292      -0.4050   *
  hotspotYN  landscape   1.3600    0.4955 63     2.7449  0.0061       0.6332  **

  Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05

---
Individual R-squared:

   Response     method R.squared
      leafN       none      0.25
  hotspotYN nagelkerke      0.61
```

GLM in SEM



Herbaceous Biomass

Leaf Nitrogen
$R^2=0.25$

Hotspot
$R^2=0.61$

Position in Landscape

low biomass improves forage quality

-0.49***

low biomass aids predator avoidance

-0.41*

0.34*

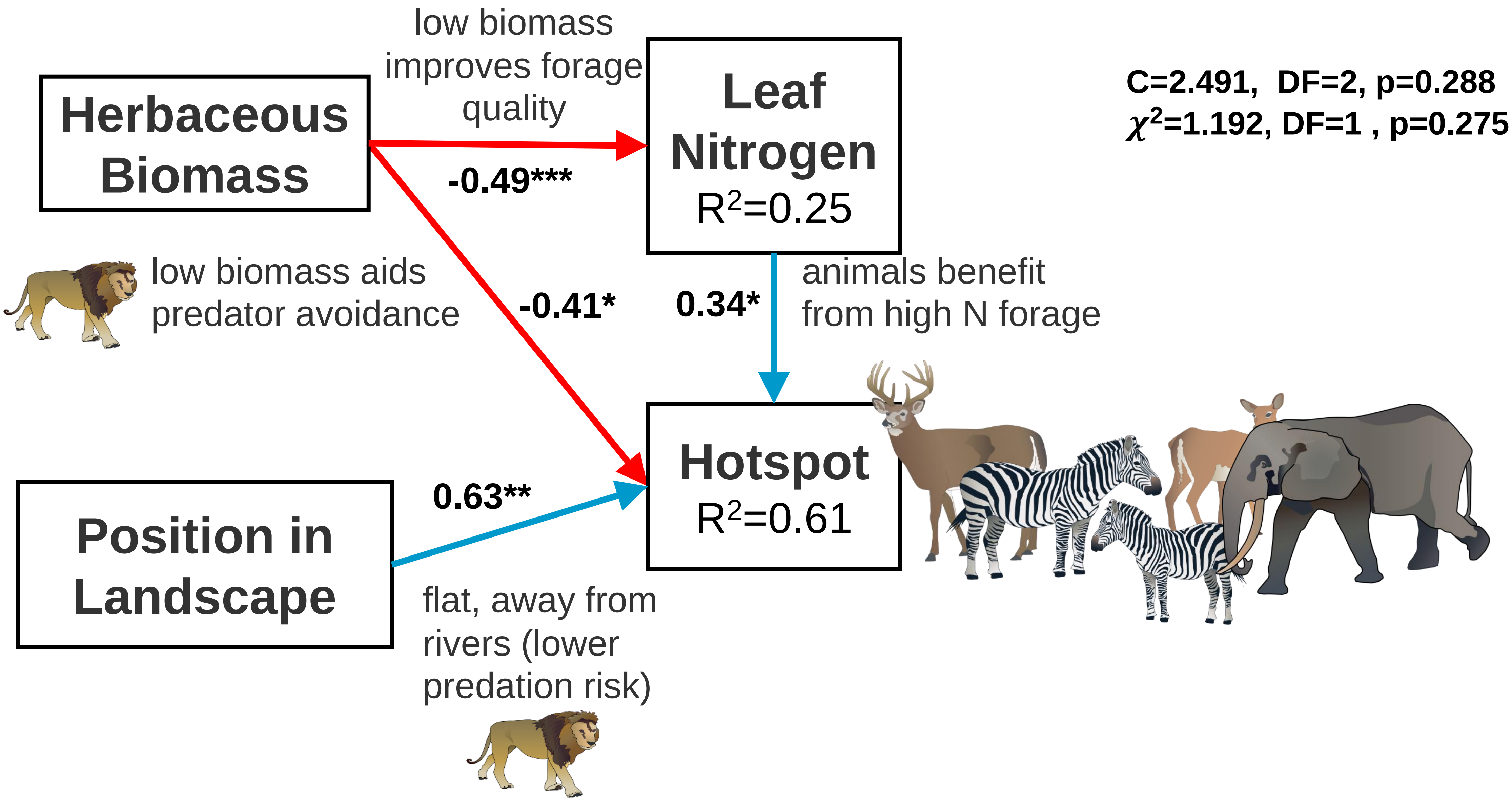
animals benefit from high N forage

0.63**

flat, away from rivers (lower predation risk)

C=2.491, DF=2, p=0.288
χ^2=1.192, DF=1 , p=0.275

GLM in SEM



Herbaceous Biomass

low biomass improves forage quality

-0.49***

Leaf Nitrogen
R^2=0.25

C=2.491, DF=2, p=0.288
χ^2=1.192, DF=1 , p=0.275

low biomass aids predator avoidance

-0.41*

0.34*

animals benefit from high N forage

Hotspot
R^2=0.61

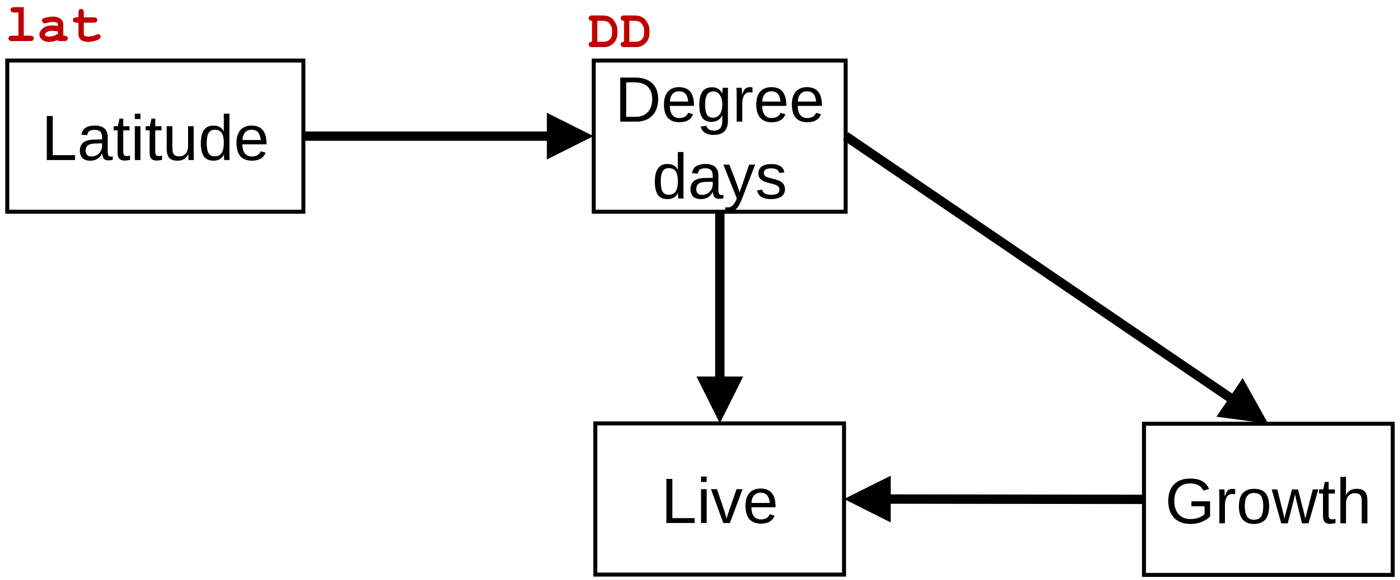
Position in Landscape

0.63**

flat, away from rivers (lower predation risk)

Day 8 Task 5

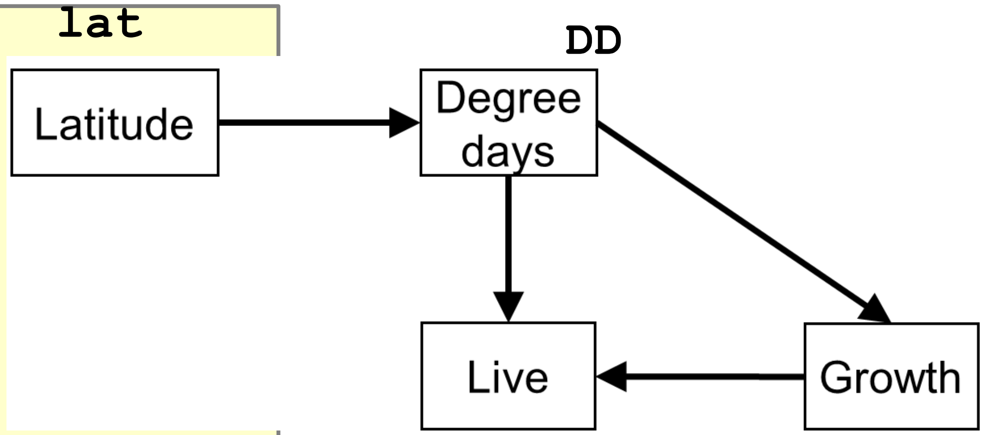
```
# Shipley data
library(piecewiseSEM)
data(shipley)
```



- Dataset: predicting latitude (**lat**) effect on survival of tree species survival - **Live** (0/1), which is influenced by phenology (**DD** - degree days until bud break, i.e. Julian days until bud break) and size (stem diameter **growth**)

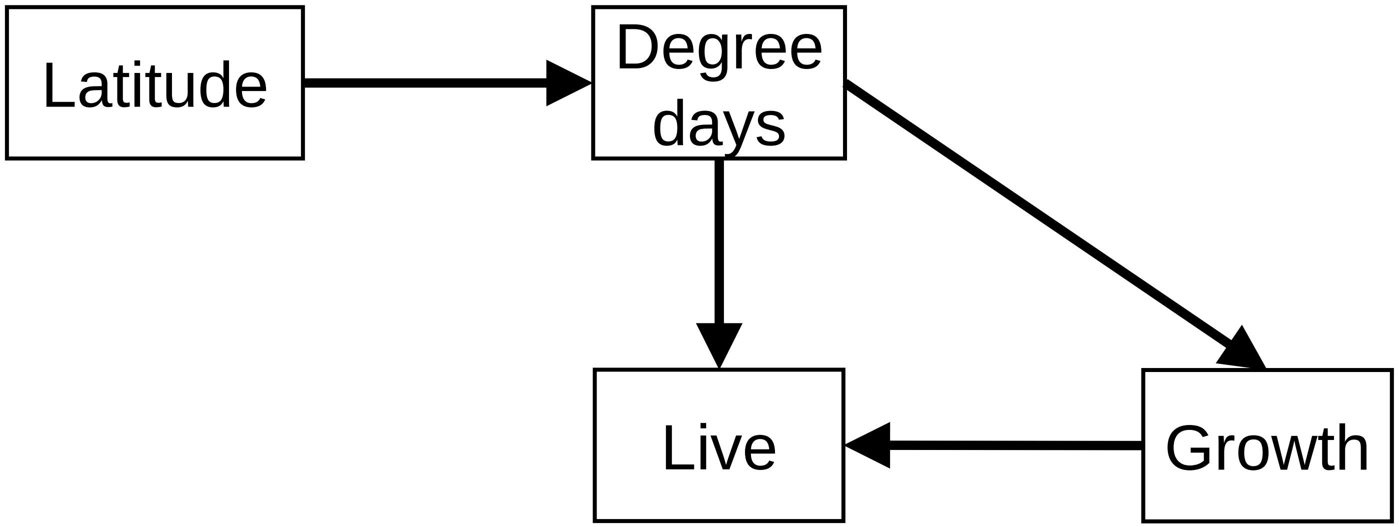
Day 8 Task 5

```
# Shipley data
library(piecewiseSEM)
data(shipley)

> str(shipley)
'data.frame': 1900 obs. of  9 variables:
 $ site     : int  1 1 1 1 1 1 1 1 1 1 ...
 $ tree     : int  1 2 3 4 5 1 2 3 4 5 ...
 $ lat      : num  40.4 40.4 40.4 40.4 40.4 ...
 $ year     : int  1970 1970 1970 1970 1970 1972 1972 1972 1972
1972 ...
 $ Date     : num  115 118 116 111 121 ...
 $ DD       : num  161 159 160 161 157 ...
 $ Growth   : num  61.4 43.8 44.7 48.2 50 ...
 $ Survival: num  1 0.843 0.944 0.957 0.976 ...
 $ Live     : int  1 1 1 1 1 1 1 1 1 1 ...
```



Day 8 Task 6

```
# Shipley data
library(piecewiseSEM)
data(shipley)
```

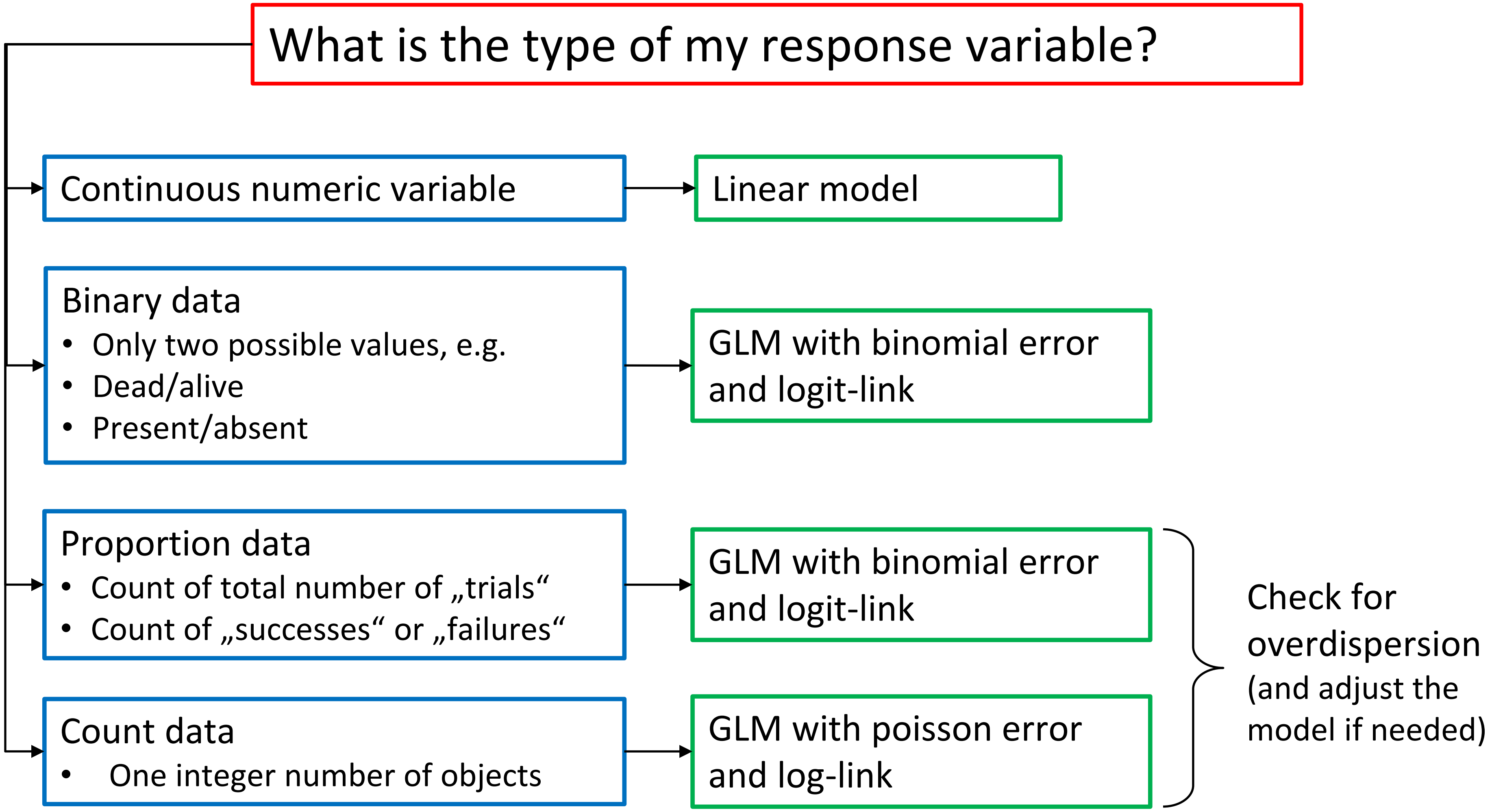


- Dataset: predicting latitude (**lat**) effect on survival of tree species survival - **Live** (0/1), which is influenced by phenology (**DD** - degree days until bud break, i.e. Julian days until bud break) and size (stem diameter **growth**)

Measures of “Live” and “growth” were the repeated measures on 5 trees at 20 sites. “DD” was the repeated measures on sites.

Given this information repeat the analysis with mixed effect models

Recap Select the appropriate model type



What is the type of my response variable?

Continuous numeric variable

Linear model

Binary data
- Only two possible values, e.g.
- Dead/alive
- Present/absent

GLM with binomial error and logit-link

Proportion data
- Count of total number of „trials“
- Count of „successes“ or „failures“

GLM with binomial error and logit-link

Count data
- One integer number of objects

GLM with poisson error and log-link

Check for overdispersion (and adjust the model if needed)

Recap

GLMs

	Linear models LM	**Generalized linear models GLM**

Model fitting	`lm()`	`glm(..., family = “poisson”/`
		`family = “binomial”`

Checking model assumptions	`plot()`	`Overdispersion`
Looking at coefficients	`summary()`	`summary()`

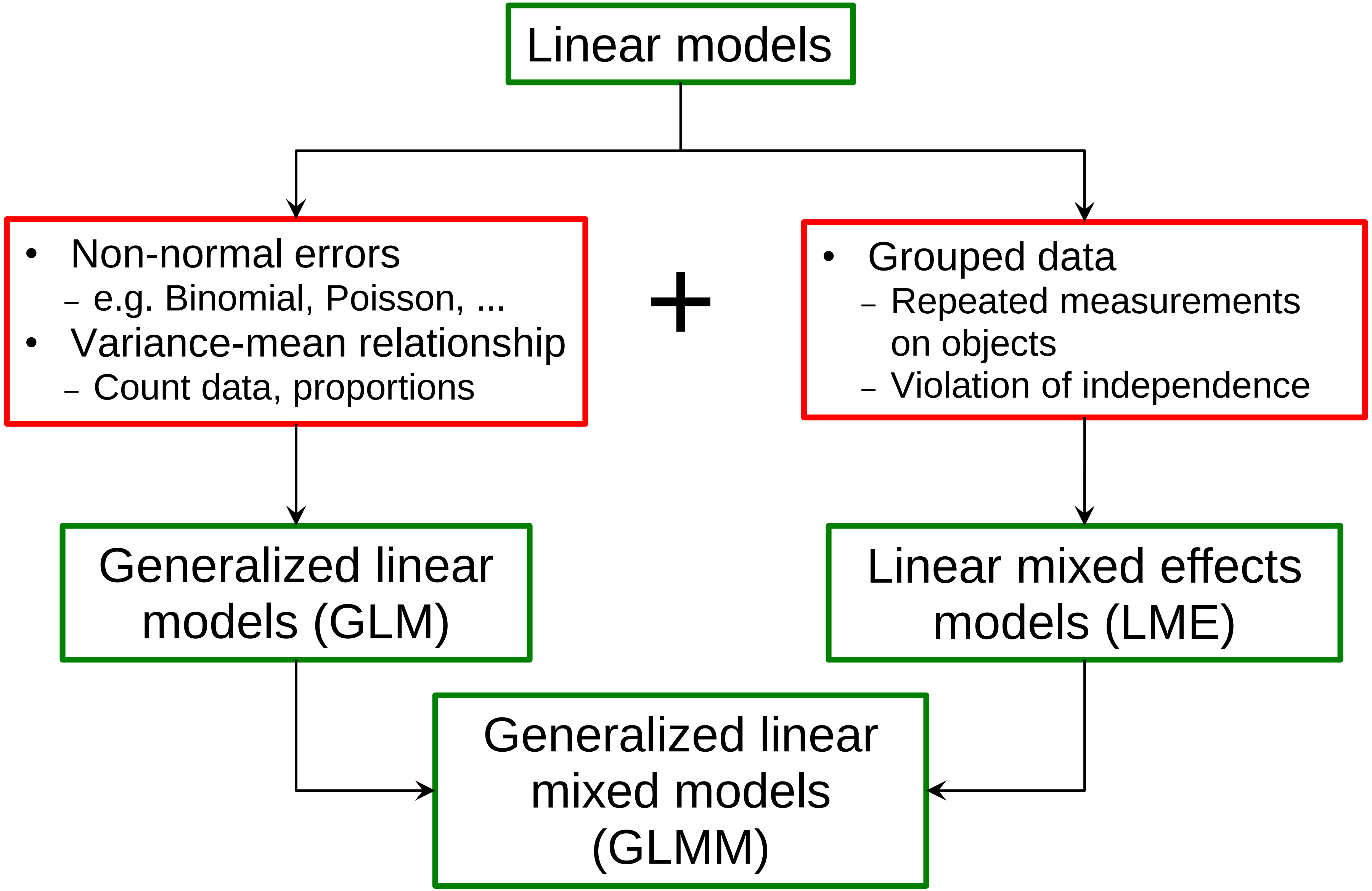
Model predictions	`predict()`	`predict(…, type = “response” /`
		`type = “link”)`

Hypothesis testing & model selection	`drop1(…, test = “F”)`
	`AIC()`
		`drop1(…, test = “Chi”)`
		`AIC()`

- Overdispersion:
  - Residual deviance ⇔ residual degrees of freedom (df)
- Residual deviance / df > 1.5 → overdispersion
- Residual deviance / df < 0.6 → underdispersion
  - (very rare)

Recap

G(L)MMs



Linear models
- Non-normal errors
  - e.g. Binomial, Poisson, ...
- Variance-mean relationship
  - Count data, proportions
+
- Grouped data
  - Repeated measurements on objects
  - Violation of independence
Generalized linear models (GLM)
Linear mixed effects models (LME)
Generalized linear mixed models (GLMM)

Recap

LMM vs. GLMM in R

Package `lme4`

	Linear mixed models – LMM	Generalized linear mixed models GLMM

Model fitting	`lmer()`	`glmer(…, family = “poisson”/`
		`family = “binomial”)`

Checking model assumptions	`plot()`	`plot()`
Looking at coefficients	`summary()`	`summary()`

Model predictions with random effects	`predict(…, re.form = NULL)`	`predict(…, re.form = NULL,`
		`type = “response”)`

Model predictions fixed effects only	`predict(…, re.form = NA)`	`predict(…, re.form = NA,`
		`type = “response”)`

Recap

LMM vs. GLMM in R

Packages `lme4` and `lmerTest`

	Linear mixed models – LMM	Generalized linear mixed models GLMM
Testing random effects	`ranova()` from package `lmerTest`	`anova()`

Testing fixed effects	`drop1()`
	`(drop1.lmerModLmerTest)`
		`drop1()`
		`(drop1.merMod)`

SSQ / df > 1.5 → indication of overdispersion