Introduction to structural equation modeling and mixed models in R

Day 9 – Part 1: SEM

Oksana Buzhdygan

oksana.buzh@fu-berlin.de

Day 9 – 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 9 – 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

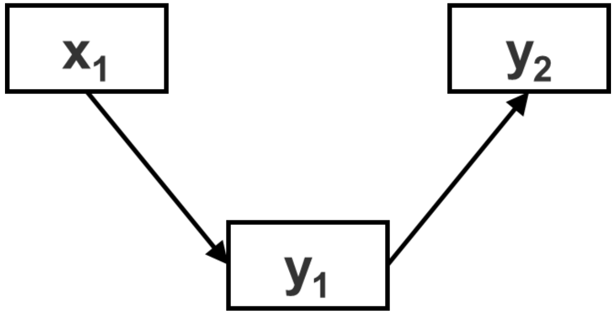
Global vs. Local Estimations

Two Paradigms for model estimation

Global Estimation
(*Covariance-Based Estimation*)

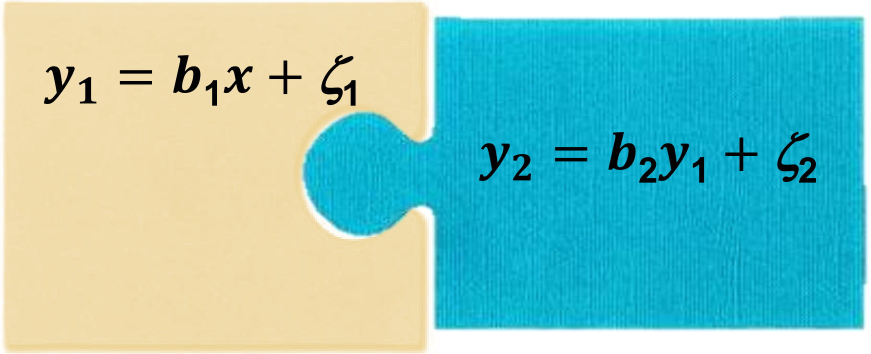
- reproduces a single variance-covariance matrix

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



Local Estimation
(*piecewiseSEM*)

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



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“)`

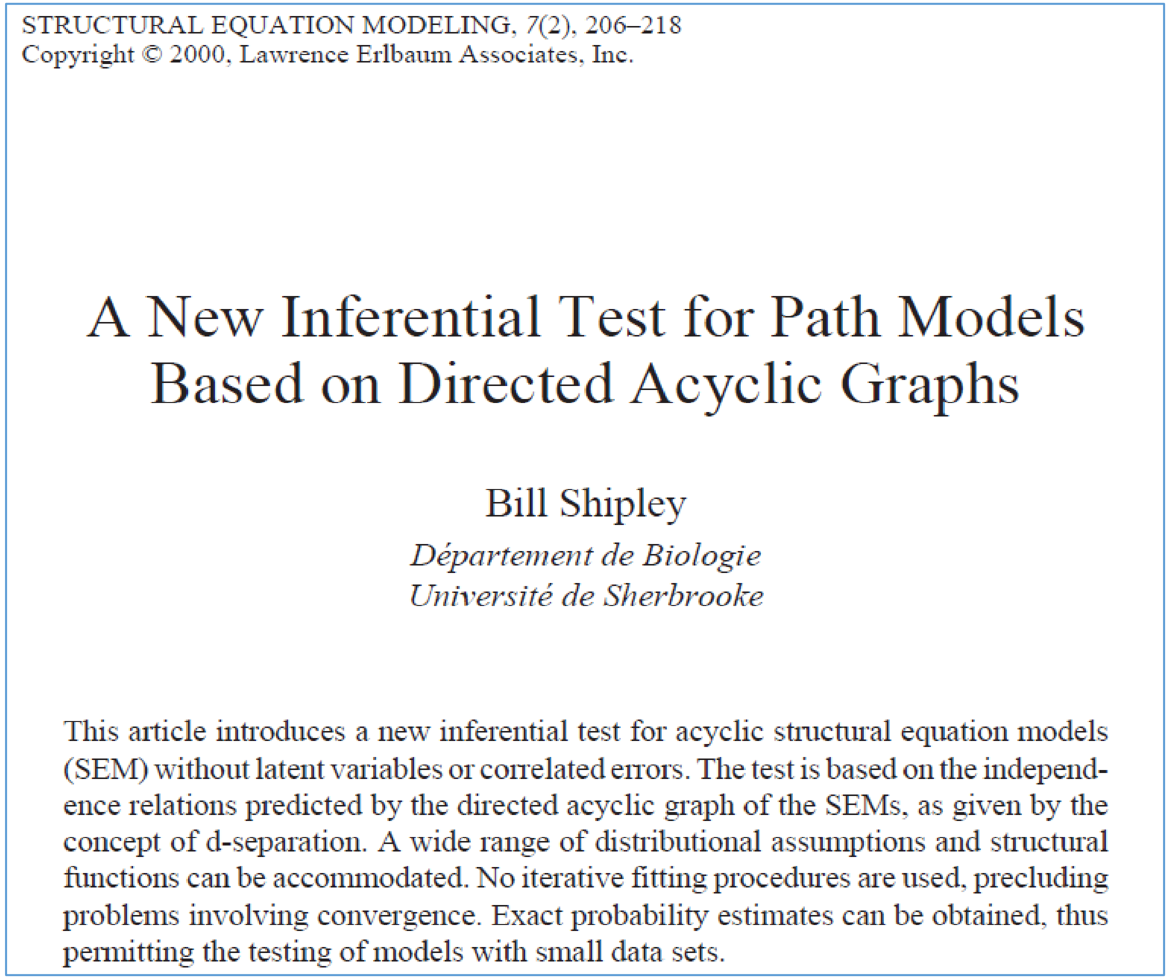
	Bootstapping: `# Always report results for 'robust' test statistics`
	`library(lavaan)`
	`sem(…, test="bollen.stine", se="bootstrap)`

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

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

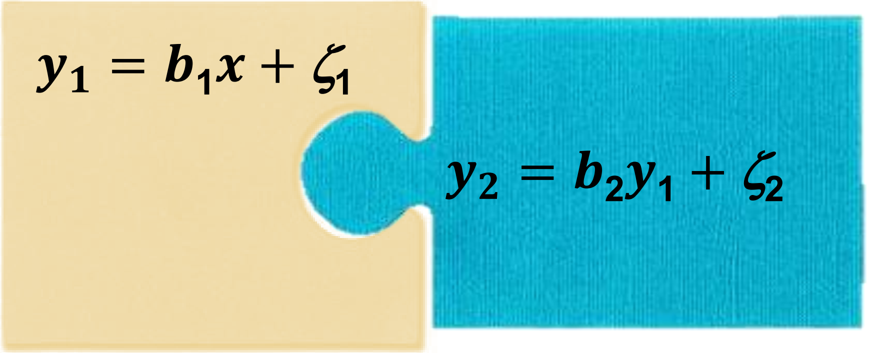
Bill Shipley
Département de Biologie
Université de Sherbrooke

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



Global vs. Local Estimations

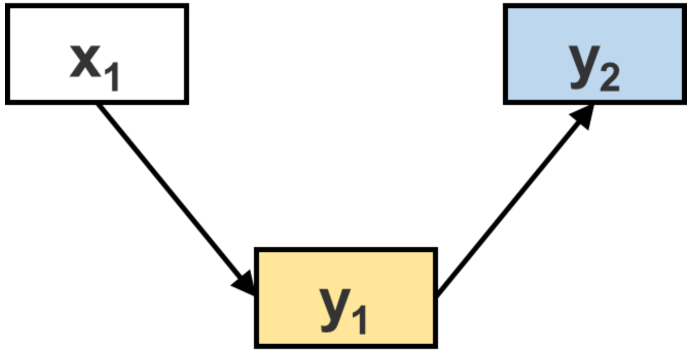
```
' y1~x1
  y2~y1 '
```

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

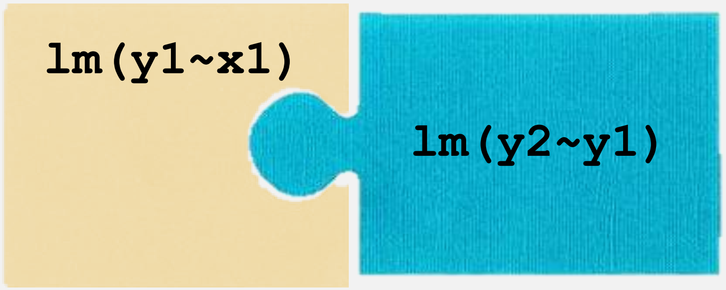
Global Estimation

(*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

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

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.

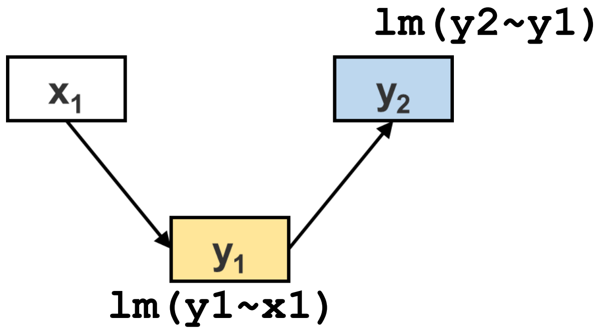
Moreover, as the author of the *piecewiseSEM* package, this format allows me to document newly-deployed functionality in the package, such as the addition of categorical variables, multigroup analysis and composite variables, new forms of coefficient standardization, and updates to model R^2s.

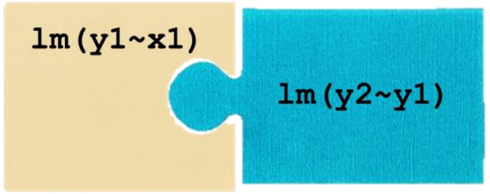
Check back often, as this book is a “living resource:” as new functionality is added and bugs uncovered and fixed, they will be described in detail here (with worked examples where possible).

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

Piecewise SEM

```r
# Install the development version of piecewiseSEM
devtools::install_github("jslefche/piecewiseSEM@devel")

library(piecewiseSEM)

# Check assumptions
m1 <- lm(y1 ~ x1, data = data1)
m2 <- lm(y2 ~ y1, data = data1)
plot(m1)
plot(m2)

# Model specification in piecewiseSEM
psem_mod1 <- psem(
  lm(y1 ~ x1, data = data1),
  lm(y2 ~ y1, data = data1)
)
# or
psem_mod1 <- psem(m1, m2)
```



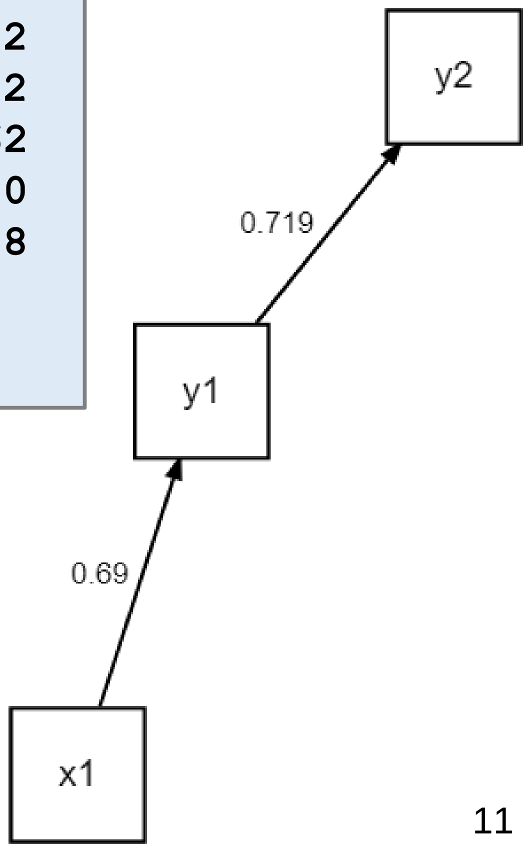



```r
# additional syntax
library(piecewiseSEM)

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

Piecewise SEM

```
# Install the development version of
piecewiseSEM
devtools::install_github("jslefche/piece
wiseSEM@devel")
library(piecewiseSEM)
# Check assumptions
m1 <- lm(y1 ~ x1, data = data1)
m2 <- lm(y2 ~ y1, data = data1)
plot(m1)
plot(m2)

# Model specification in piecewiseSEM
psem_mod1 <- psem(
  lm(y1 ~ x1, data = data1),
  lm(y2 ~ y1, data = data1)  )
# or
psem_mod1 <- psem(m1, m2)

# Examine the psem object
psem_mod1

# Plot the model
plot(psem_mod1)
```

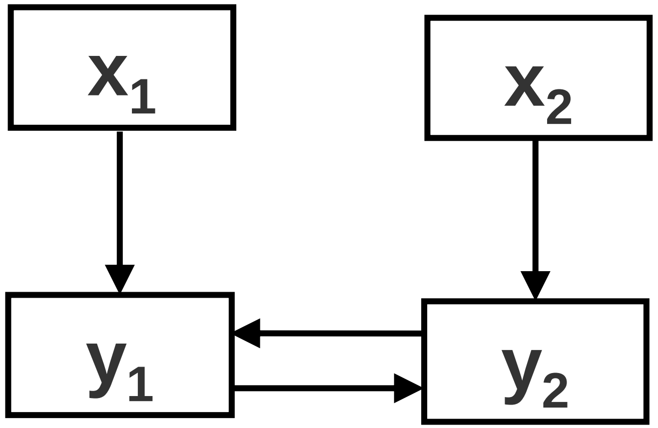
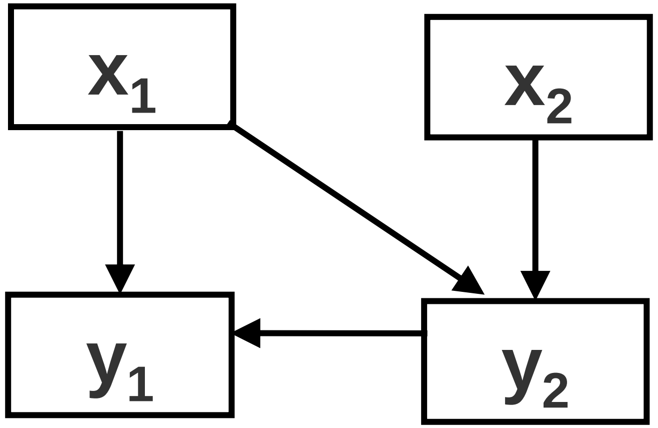
```
> psem_mod1

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)"
```



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

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?

Day 9 – 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

Assessing Model Fit

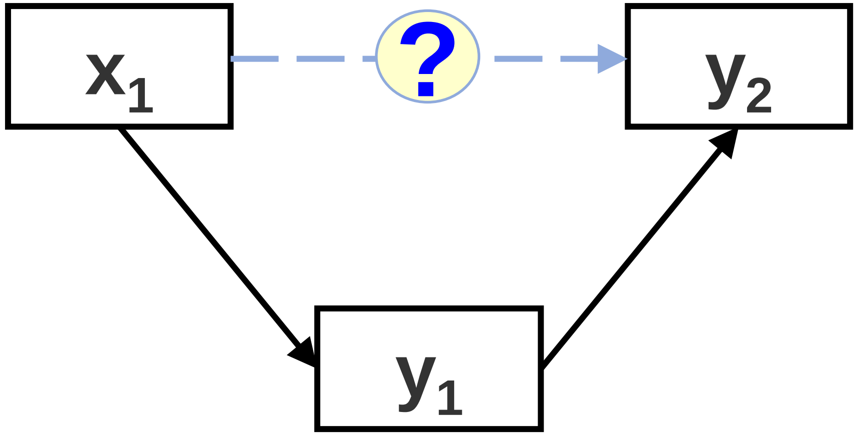
Tests of directed (d-)separation

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

Thus, we test the partial effect of x_1 on y_2 given 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.

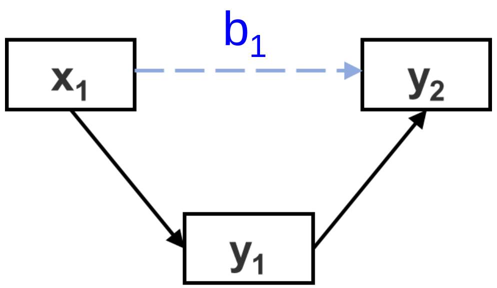
Assessing Model Fit

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

$y_2 = a + b_1 x_1 + b_2 y_1$

Assessing Model Fit

Fisher's *C* statistic

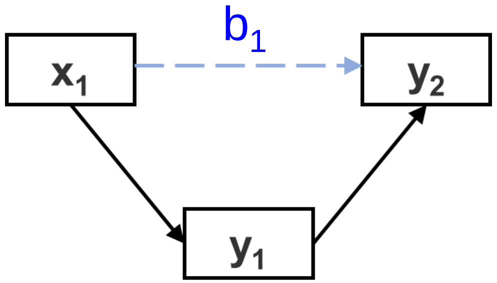
Tests of directed (d-)separation

(of statistical independence):

```
# Derive the basis set
basisSet(psem_mod1)
> $`1`
[1] "x1 | y2 ( y1 )"
```

Independence claim

$x_1|y_2(y_1)$



$y_2 = a + b_1x_1 + b_2y_1$

Assessing Model Fit

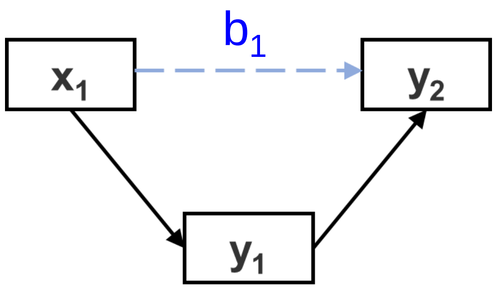
Fisher's *C* statistic

Tests of directed (d-)separation

(of statistical independence):

<u>Independence claim</u>

$x_1|y_2(y_1)$



$y_2 = a + b_1 x_1 + b_2 y_1$

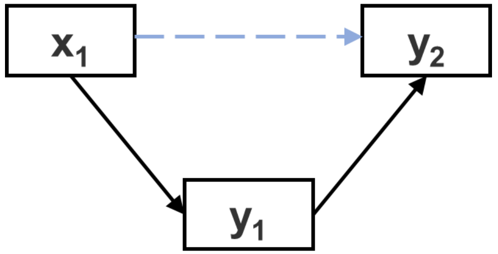
```
# Derive the basis set
basisSet(psem_mod1)
> $`1`
[1] "x1 | y2 ( y1 )"

# 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
```

missing effect is not different from 0

Assessing Model Fit

Fisher's *C* statistic

<u>Independence claim</u> $x_1|y_2(y_1)$



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

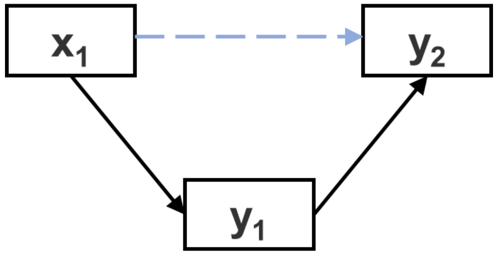
Model fit
Fisher's ***C*** statistic

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***

$\mathbf{K}$ is the likelihood degrees of freedom

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

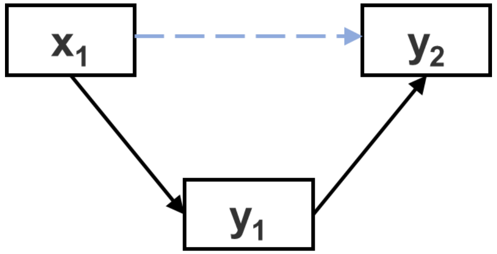
$\boldsymbol{AIC_c}$ is used when n/K < 40

$\boldsymbol{n}$ 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
```



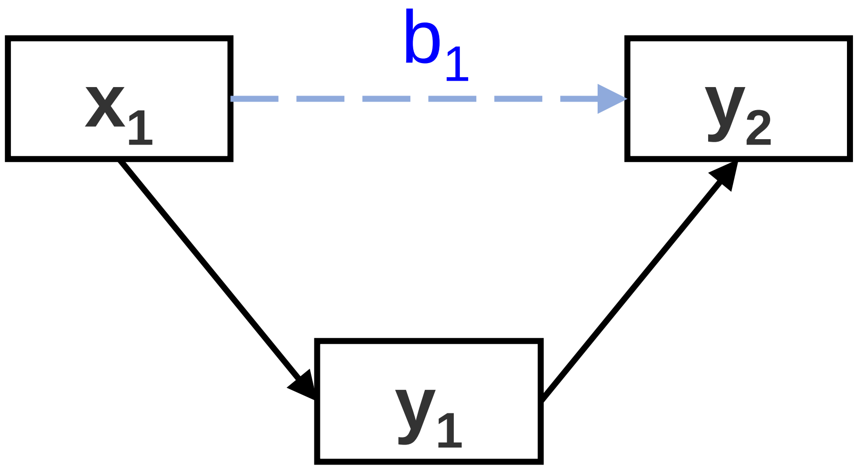
$$C = -2\sum_{i=1}^{k} \ln(p_i)$$

$$AIC = C + 2K$$

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

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

$$\chi^2 = -2\sum_{i=1}^{k}\left(\log(\mathbf{L}_{\mathbf{M}_i}) - \log(\mathbf{L}_{\mathbf{M}s_i})\right)$$

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

$$AIC = \sum_{i=1}^{k} AIC(\mathbf{M}_i)$$

AIC of the SEM model

The same for the AIC_c

Assessing Model Fit

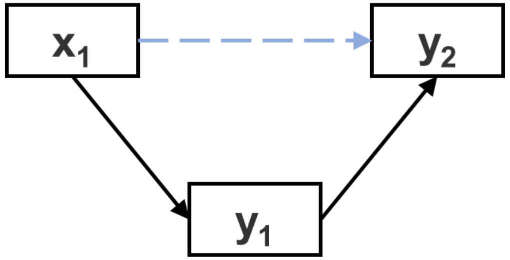
χ^2 statistic

```
# log-likelihood based χ2 statistic

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

$$\chi^2 = -2\sum_{i=1}^{k}\left(\log(\mathrm{L}_{\mathrm{M}_i}) - \log(\mathrm{L}_{\mathrm{Ms}_i})\right)$$

```
# 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
```

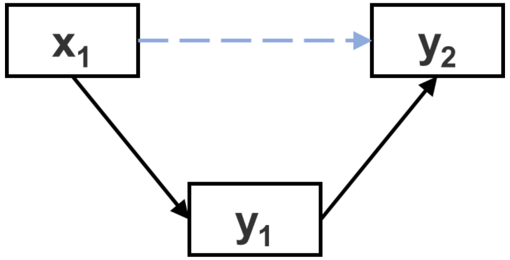
$$AIC = \sum_{i=1}^{k} AIC(\mathrm{M}_i)$$



Assessing Model Fit

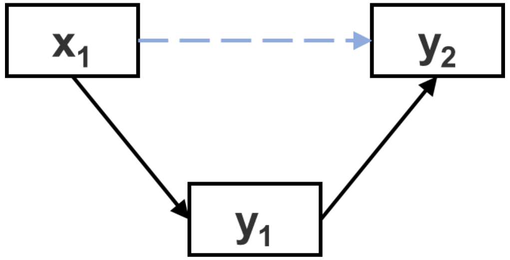
χ^2 statistic

```
# 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 9 – Part 1

Outline

- Introduction to Local Estimation in SEM

  ✓ Global vs. Local Estimations. Piecewise SEM

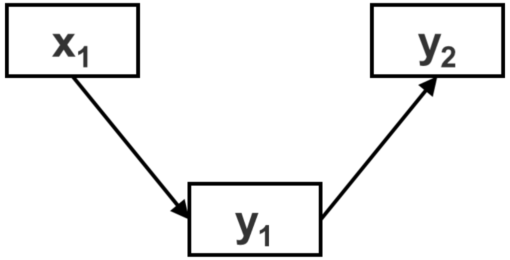
  ✓ Assessing Model Fit

  ✓ **Model Comparison**

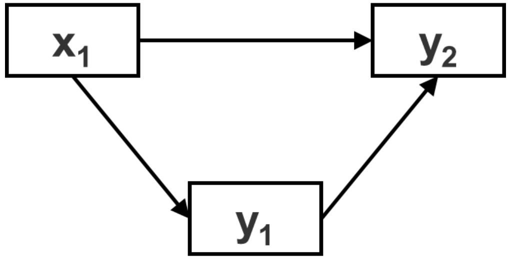
  ✓ Categorical Data in Piecewise SEM

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_mod1) # arrange correct order manually
>Chi-square Difference Test
      Df AIC.AIC AIC.K AIC.n Chisq Chisq.diff Df.diff P.value
Mod.2 0  -1.862     7   100 0.000
Mod.1 1  -2.798     6   100 1.064      1.064       1  0.3023

# AIC comparison
aic <- AIC(psem_mod1, 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

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

<table>
<tr><th>Categorical Variables</th><th>Exogenous Categorical Variables</th><th>Endogenous Categorical Variables</th></tr>
<tr><td rowspan="3">Binary variables
yes/no;
presence/absence;
failure/success;
dead/alive;
male/female</td><td>1. Treat as numeric: set the values as 0 or 1 and model as numeric (yields a single path coefficient).</td><td rowspan="6">Endogenous categorical variables are not implemented in piecewiseSEM.
Treat binary and ordinal variables as numerical (follow step 1 shown for ‘Endogenous Categorical Variables’)</td></tr>
<tr><td>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).</td></tr>
<tr><td>3. Use as categorical variable (Marginal Means approach)</td></tr>
<tr><td rowspan="3">Ordinal variables:
small < medium < large;
yang < middle < old</td><td>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.</td></tr>
<tr><td>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).</td></tr>
<tr><td>3. Use as categorical variable (Marginal Means approach)</td></tr>
<tr><td>Nominal variables
study sites ;countries;
sampling campaigns</td><td>Use as categorical variable (Marginal Means approach)</td><td>Nominal endogenous categorical variables are not implemented in piecewiseSEM</td></tr>
</table>

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

```
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
```

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

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

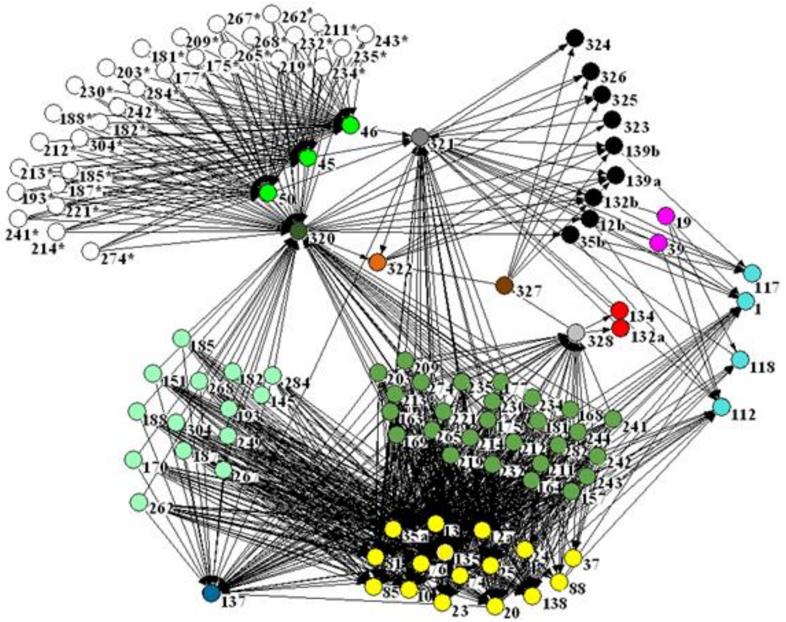
Provides a measure of whether the path between the exogenous categorical variable and the response is significant as well as parameters for each level in the form of the model-estimated marginal means.

Categorical Data in Piecewise SEM

<table>
<tr><th>Categorical Variables</th><th>Exogenous Categorical Variables</th><th>Endogenous Categorical Variables</th></tr>
<tr><td rowspan="3">Binary variables
yes/no;
presence/absence;
failure/success;
dead/alive;
male/female</td><td>1. Treat as numeric: set the values as 0 or 1 and model as numeric (yields a single path coefficient).</td><td rowspan="6">Endogenous categorical variables are not implemented in piecewiseSEM.
Treat binary and ordinal variables as numerical (follow step 1 shown for ‘Endogenous Categorical Variables’)</td></tr>
<tr><td>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).</td></tr>
<tr><td>3. Use as categorical variable (Marginal Means approach)</td></tr>
<tr><td rowspan="3">Ordinal variables:
small < medium < large;
yang < middle < old</td><td>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.</td></tr>
<tr><td>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).</td></tr>
<tr><td>3. Use as categorical variable (Marginal Means approach)</td></tr>
<tr><td>Nominal variables
study sites ;countries;
sampling campaigns</td><td>Use as categorical variable (Marginal Means approach)</td><td>Nominal endogenous categorical variables are not implemented in piecewiseSEM</td></tr>
</table>

Day 9 Task 1

Effects of land use on arthropod food webs in grasslands



Food web length (1,2,3)

```
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 ...
```

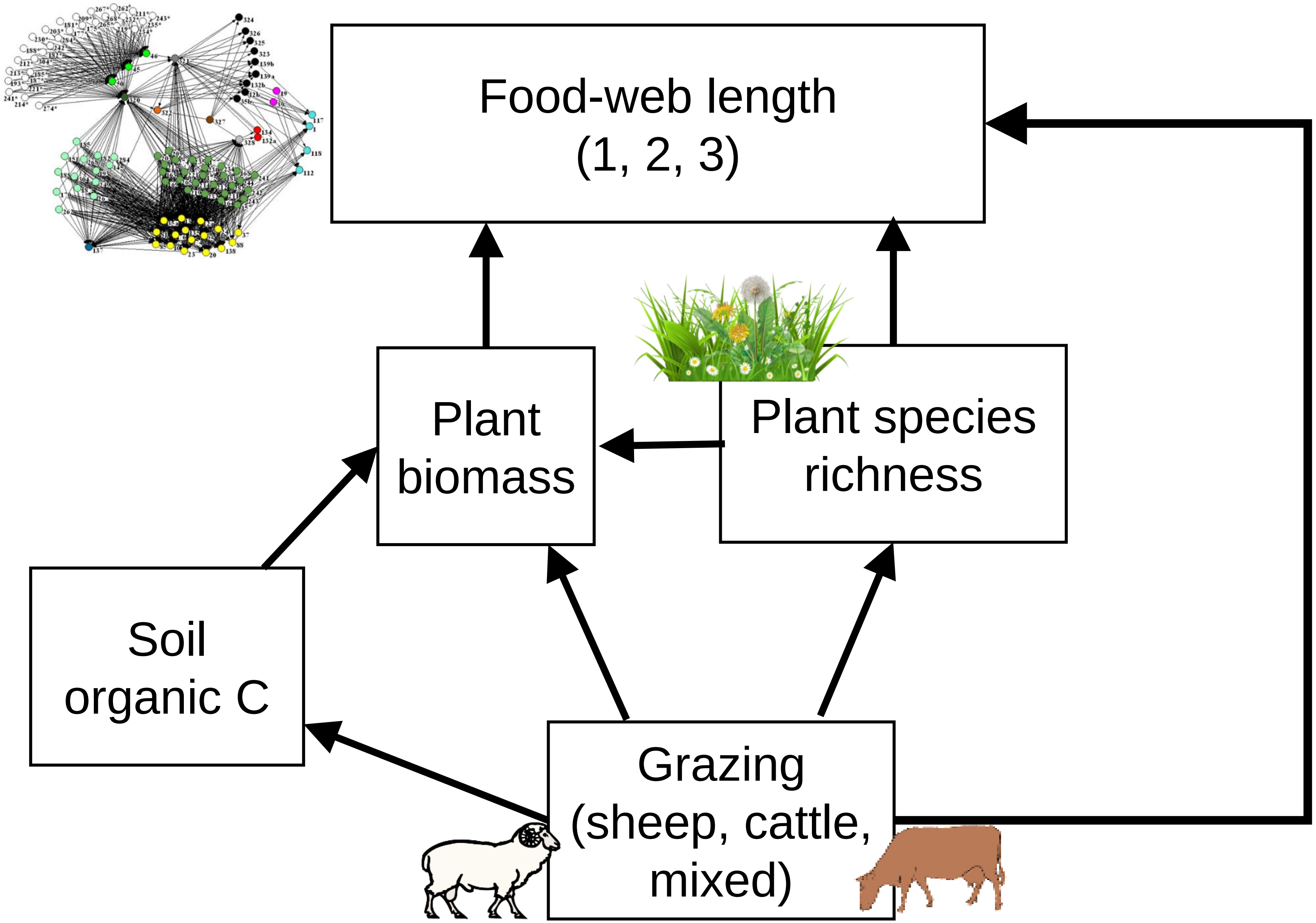
235 grasslands

Grazing type

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

Day 9 Task 1

Effects of land use on food webs in grasslands



Day 9 Task 1

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`***.

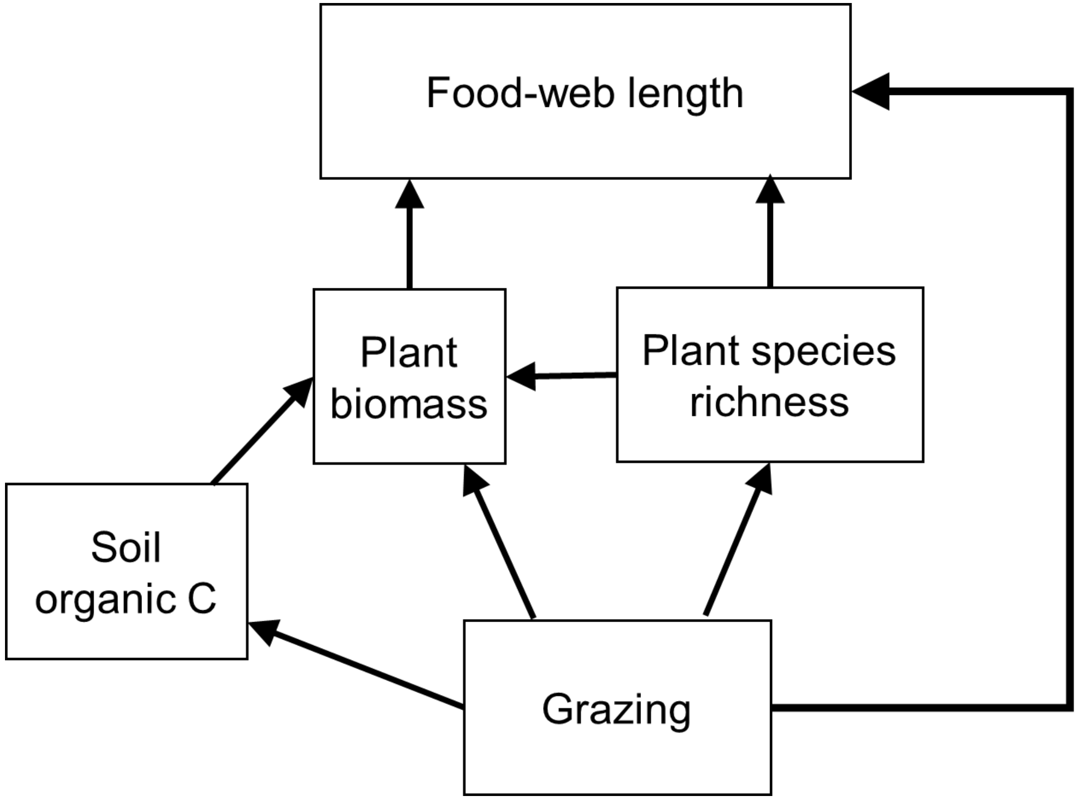


fig. 1