Introduction to structural equation modeling and mixed models in R

Day 7

Oksana Buzhdygan

oksana.buzh@fu-berlin.de

Day 7 – Part 1

Outline

- Categorical Variables in SEM

Categorical Variables in SEM

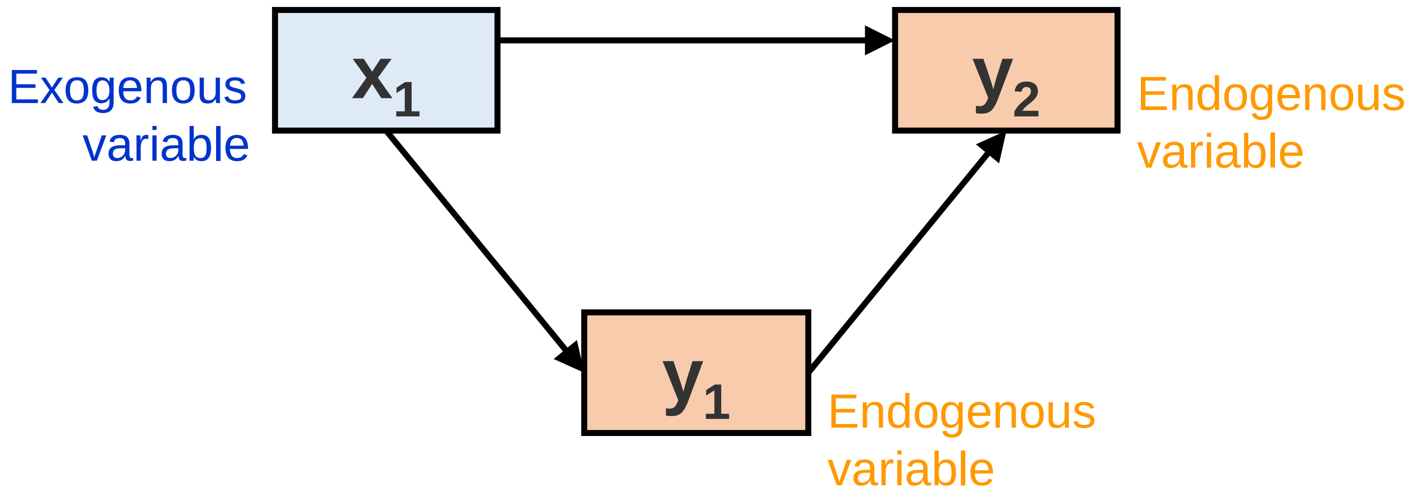
Categorical / discrete data

- binary (yes/no, failure/success, dead/alive, male/female),
- nominal (site 1, site 2, site 3)
- ordinal levels (small < medium < large; yang < middle < old).

Categorical Variables in SEM

Categorical / discrete data

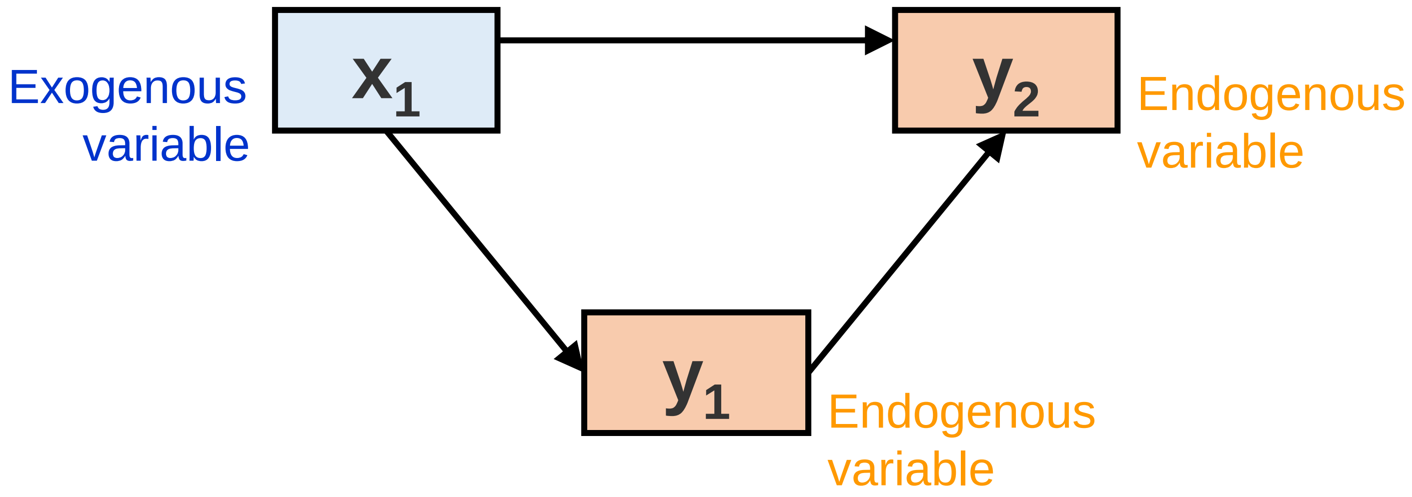
- binary (yes/no, failure/success, dead/alive, male/female),
- nominal (site 1, site 2, site 3)
- ordinal levels (small < medium < large; yang < middle < old).



Categorical Variables in SEM

Categorical / discrete data

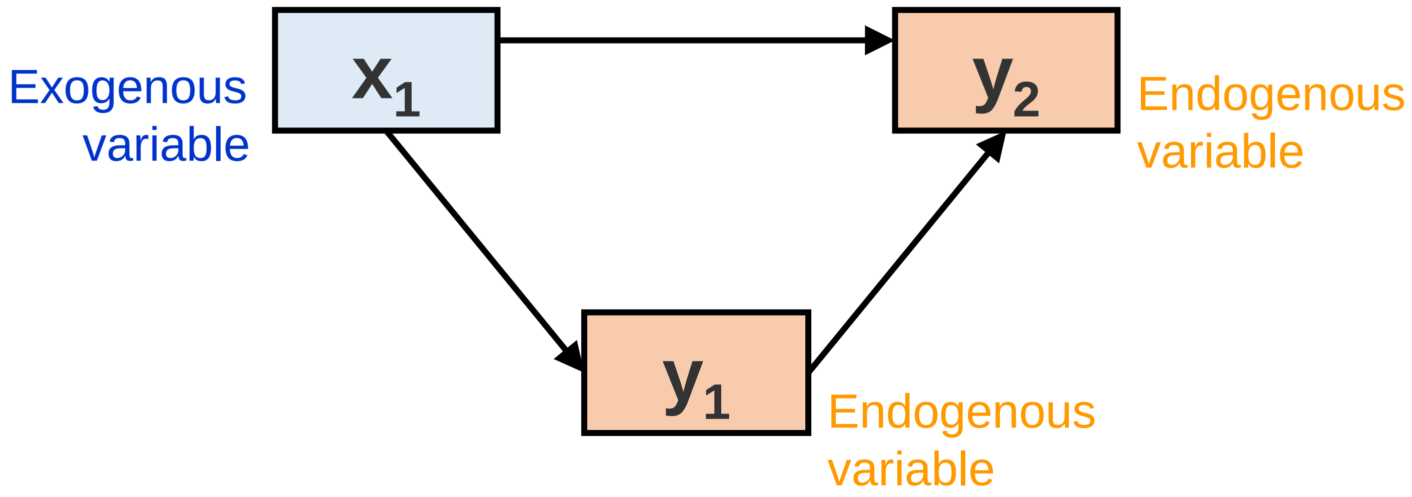
- binary (yes/no, failure/success, dead/alive, male/female),
- nominal (site 1, site 2, site 3)
- ordinal levels (small < medium < large; yang < middle < old).



Categorical Variables in SEM

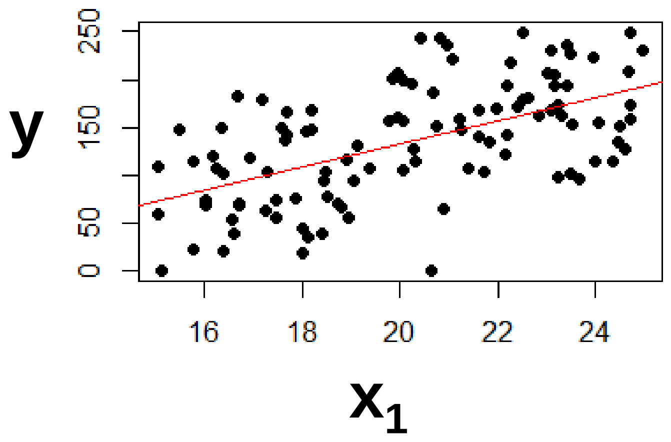
Categorical / discrete data

- binary (yes/no, failure/success, dead/alive, male/female),
- nominal (site 1, site 2, site 3)
- ordinal levels (small < medium < large; yang < middle < old).



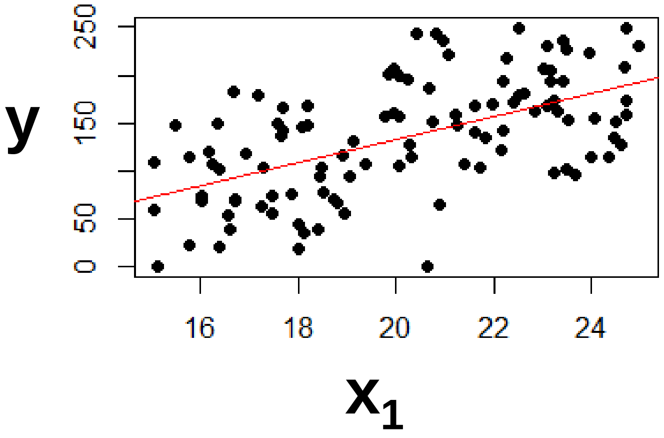
Categorical Variables in SEM

$$y = a + b_1 x_1 + \epsilon$$

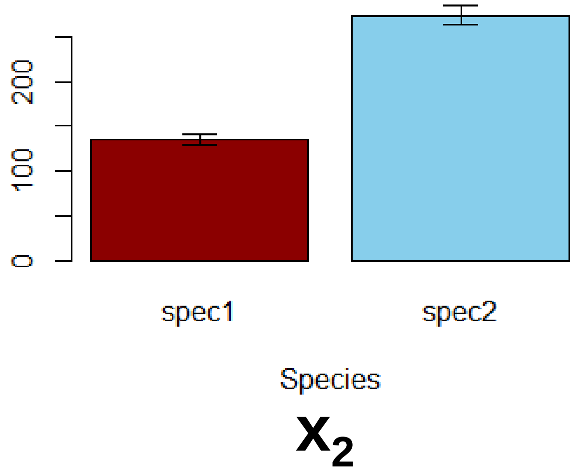


Categorical Variables in SEM

$$y = a + b_1x_1 + \epsilon$$



$$y = a + b_2x_2 + \epsilon$$

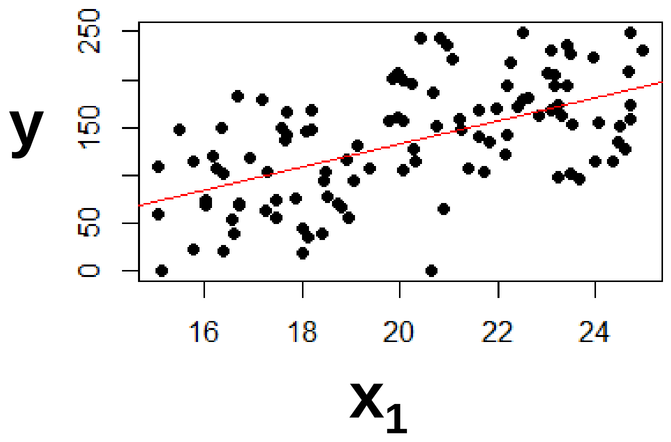


x_2

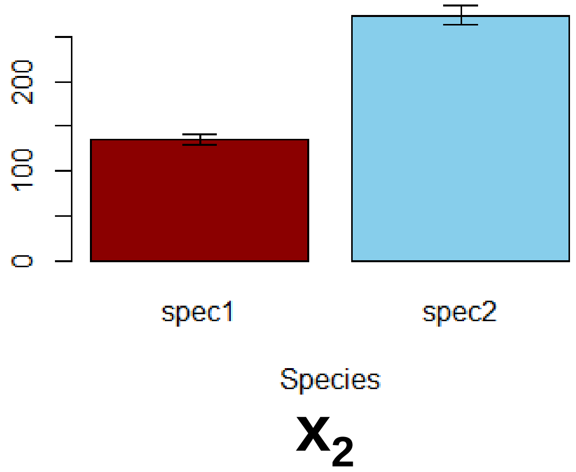
Species
spec1
spec1
spec2
spec1
spec2

Categorical Variables in SEM

$$y = a + b_1 x_1 + \epsilon$$



$$y = a + b_2 x_2 + \epsilon$$

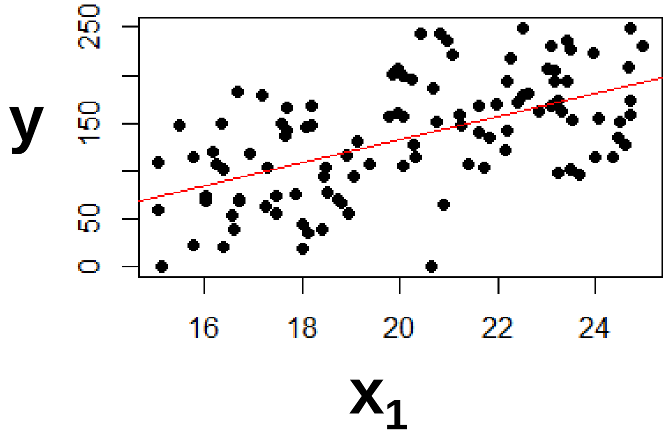


x_2

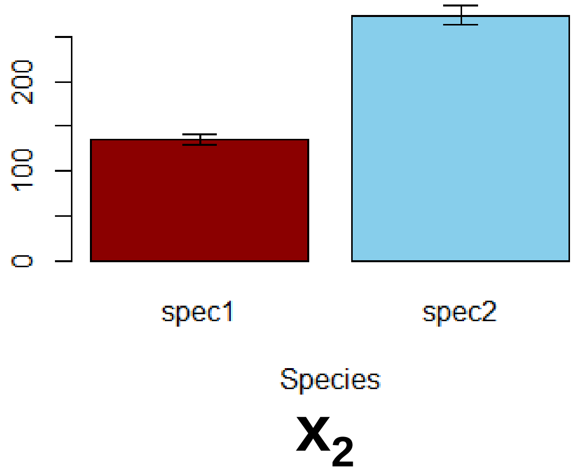
Species
spec1
spec1
spec2
spec1
spec2

Categorical Variables in SEM

$y = a + b_1 x_1 + \epsilon$



$y = a + b_2 x_2 + \epsilon$



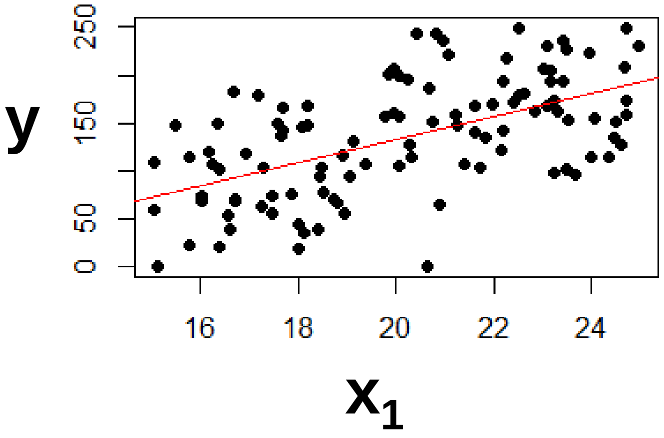
x_2

Species
spec1
spec1
spec2
spec1
spec2

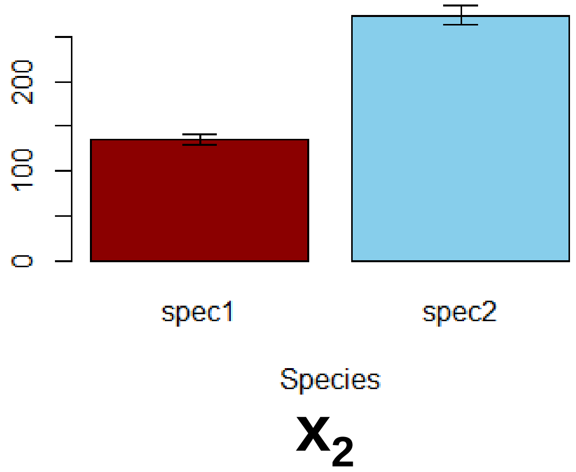
spec1	spec2
1	0
1	0
0	1
1	0
0	1

Categorical Variables in SEM

$$y = a + b_1 x_1 + \epsilon$$



$$y = a + b_2 x_2 + \epsilon$$



x_2

Species
spec1
spec1
spec2
spec1
spec2

spec1	spec2
1	0
1	0
0	1
1	0
0	1

Exogenous Categorical Variables

Approaches when we have Exogenous Categorical Variables:

1) for nominal, binary, or ordinal variables, create separate dummy variables for each factor levels (treat them as absent “0” or present “1”).
   - The key: for the factor with k levels use k-1 dummy variables (to avoid singularity)

2) for binary variables, set the values as 0 or 1 and model as numeric (yields a single coefficient).

3) for ordinal variables, set the values depending on the order of the factor, e.g., small = 1 < medium = 2 < large = 3, and then model as numeric (yields a single coefficient).

4) Use `piecewiseSEM`

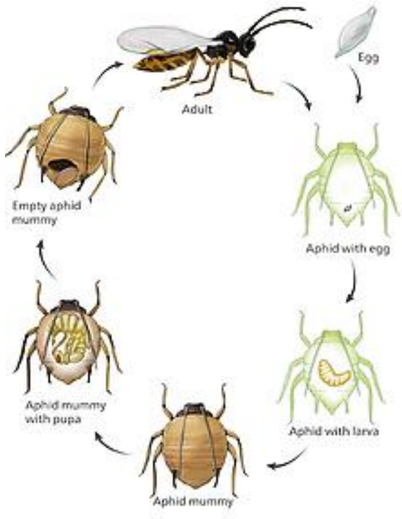
Example

Categorical Exogenous Variable

Biocontrol agents of crop-pests (aphids)

Lacewing larva

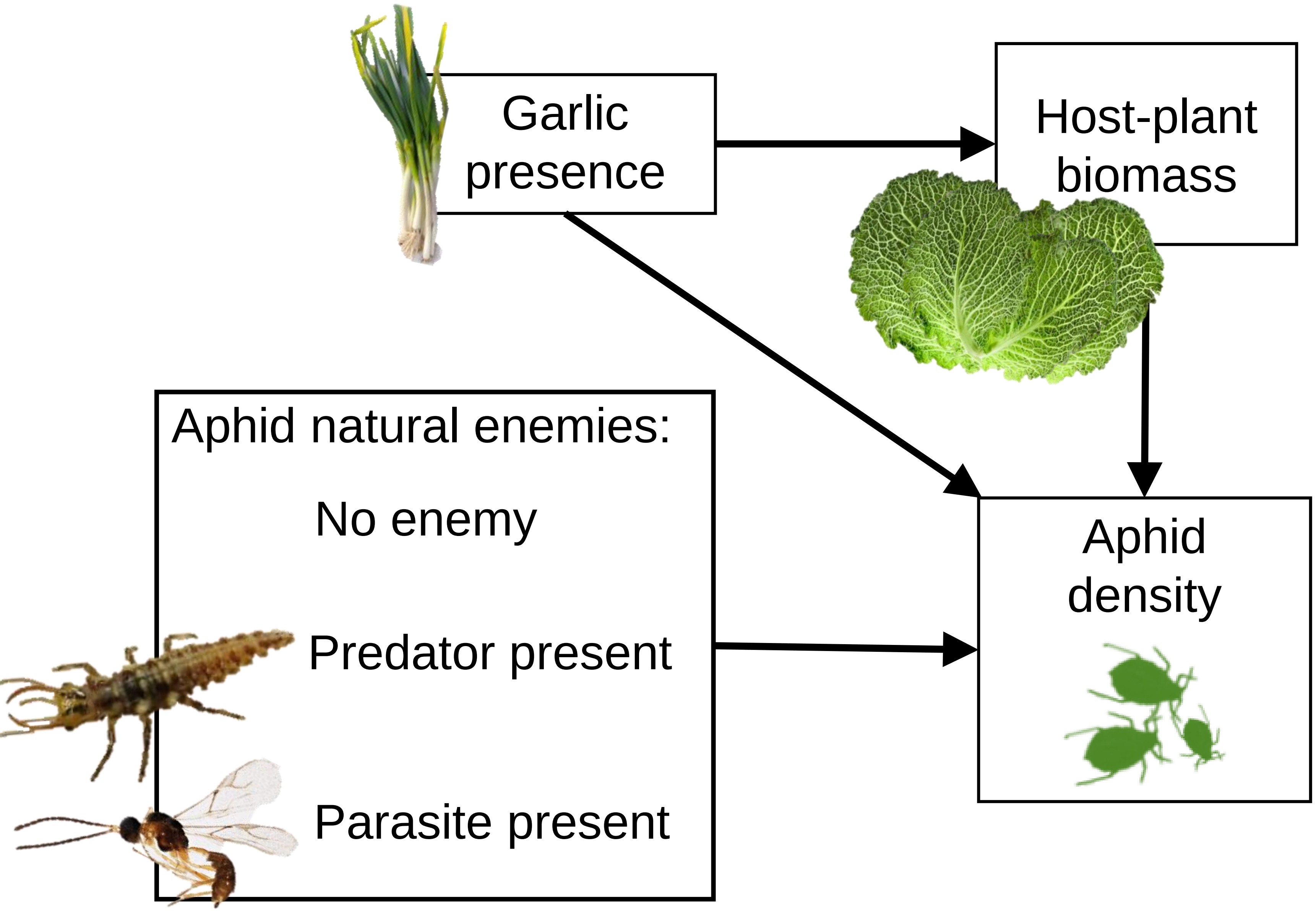
Parasitic wasp



Intercropping with repellent plants

Example

Categorical Exogenous Variable



150 experimental microcosms

Example

Categorical Exogenous Variable

```
# Read and check the data

aphid_data <- read_csv(“Data/Aphid_data.csv")

 > str(aphid_data)

spc_tbl_ [150 × 4] (S3: spec_tbl_df/tbl_df/tbl/data.frame)

 $ aphid     : num [1:150] 14.9 35.6 43.8 2.1 36.7 ...

 $ host_plant: num [1:150] 38.8 40.7 46.9 35.2 50.9 ...

 $ garlic_ef : chr [1:150] "present" "absent" "absent" "present" ...

 $ enemy     : chr [1:150] "predator" "predator" "no_enemy" "parasite" ..
```

binary variable

nominal variable

Example


```
# Create dummy variables-----
# convert "enemy" in 3 binary dummy variables
# and convert garlic_ef into 1 binary variable called garlic

aphid_data <- aphid_data %>%
   mutate(n = 1) %>%
   pivot_wider(names_from = enemy, values_from = n,
                          values_fill = list(n = 0)) %>% # convert "enemy"
   mutate(garlic = case_when(garlic_ef == "present" ~ 1, # convert " garlic_ef"
                             garlic_ef == "absent" ~ 0))
```

Example

Categorical Exogenous Variable

```
$ aphid      : num [1:150] 14.9 35.6 43.8 2.1 36.7 ...
$ host_plant: num [1:150] 38.8 40.7 46.9 35.2 50.9 ...
$ garlic_ef : chr [1:150] "present" "absent" "absent" "present" .
$ garlic     : num [1:150] 1 0 0 1 0 1 1 1 0 0 ...
$ predator   : num [1:150] 1 1 0 0 0 0 1 0 1 0 ...
$ no_enemy   : num [1:150] 0 0 1 0 1 1 0 0 0 0 ...
$ parasite   : num [1:150] 0 0 0 1 0 0 0 1 0 1 ...
```

(0/1) - dummy variable for binary

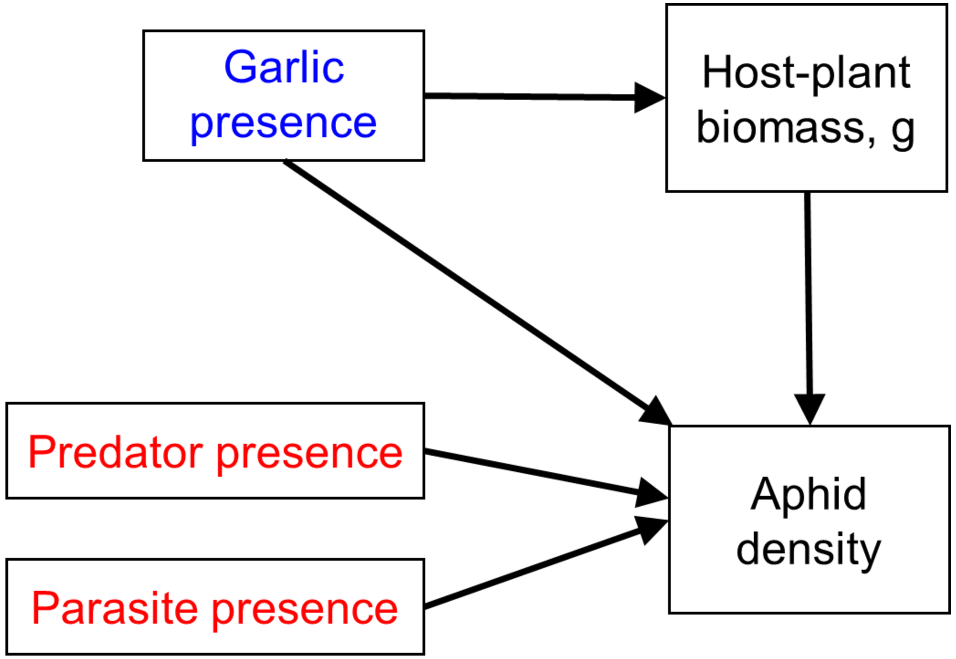
dummy variables created for each factor level from the nominal variable

Example

Categorical Exogenous Variable

```
# specify and fit the model in lavaan

sem_mod <- ' aphid ~ host_plant + garlic + predator + parasite

             host_plant ~ garlic

'
```

Only 2 out of 3 dummy variables are included



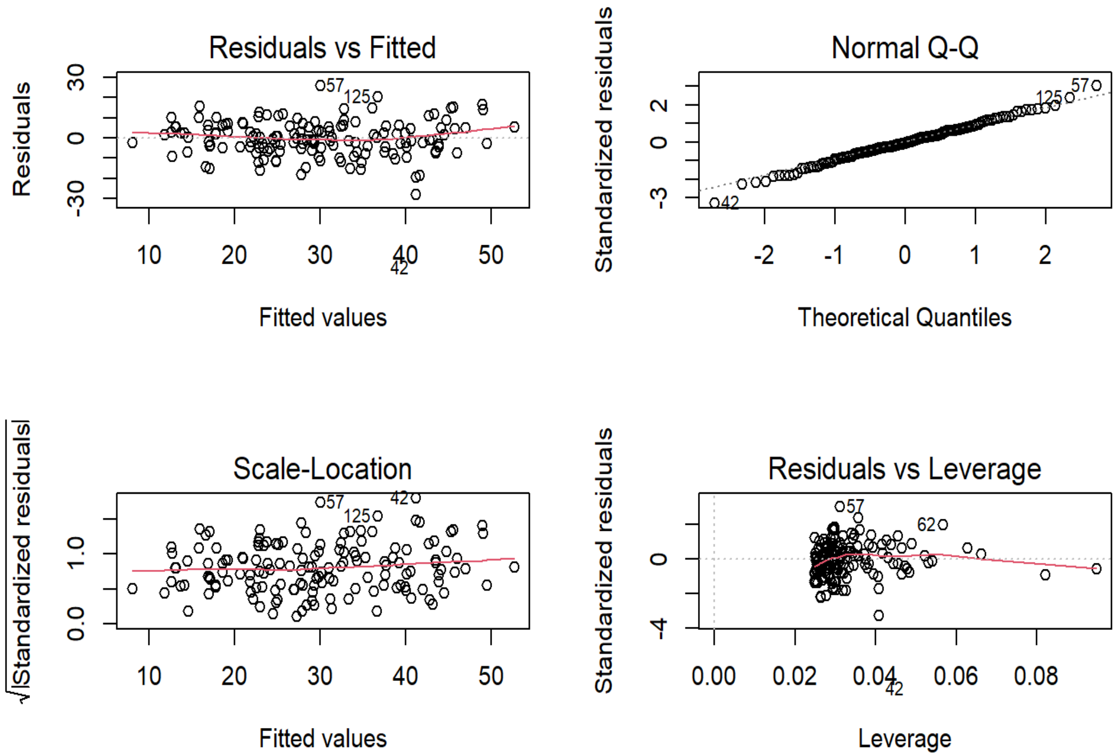
Example

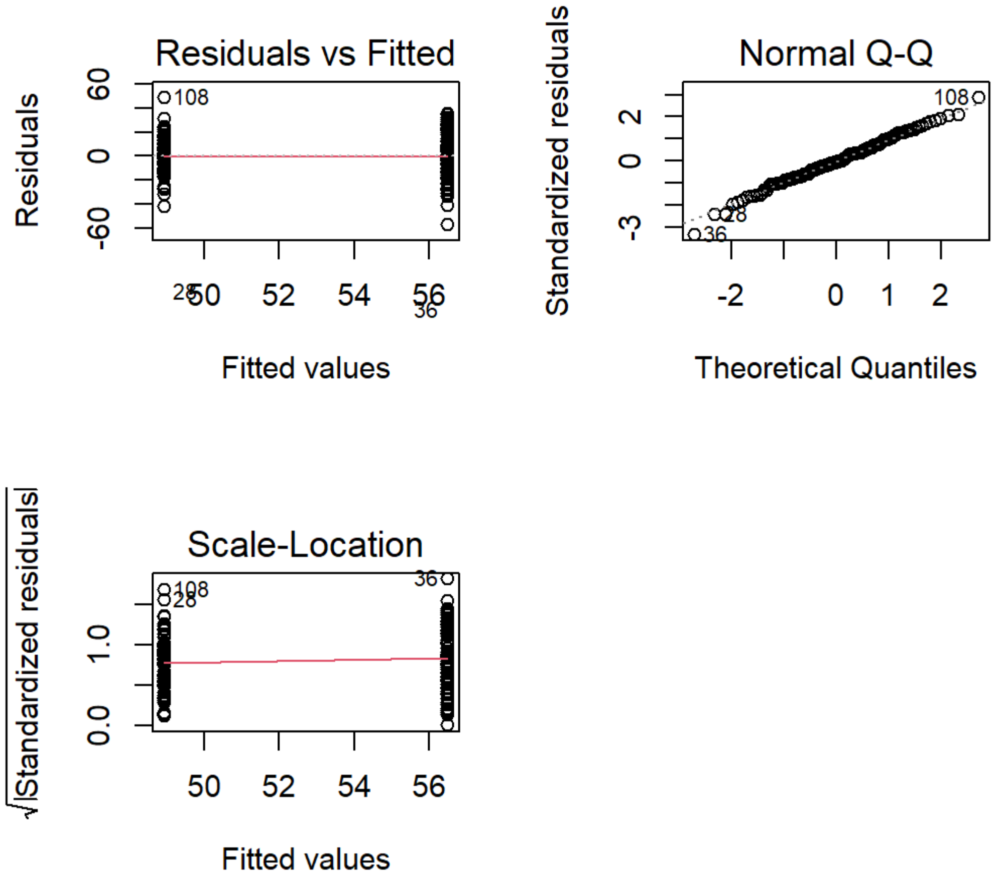

```
#Check the assumptions:
# Normality of residuals
mod1 <- lm(aphid ~ host_plant + garlic + predator + parasite, aphid_data)
car::vif(mod1) # check for correlation among predictors
>
host_plant     garlic   predator   parasite
  1.061508   1.066580   1.343933   1.355159

mod2 <- lm(host_plant ~ garlic, aphid_data)
```

Example

Categorical Exogenous Variable

```
par(mfrow=c(2,2))
plot(mod1)
plot(mod2)
par(mfrow=c(1,1))
```





Example

Categorical Exogenous Variable

```
# Normality of data
library(MVN)
mvn(aphid_data %>%
       select(-enemy_cat, -garlic_ef, -no_enemy),
       mvnTest="mardia", univariateTest="SW")
```

```
>
$multivariateNormality
             Test         Statistic              p value Result
1 Mardia Skewness  71.3789094435619 0.000273949726176333     NO
2 Mardia Kurtosis -3.9601958323228 7.48883249854781e-05     NO
3             MVN              <NA>                 <NA>     NO
```

Recap: 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")`

	Bootstrapping: `# 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`

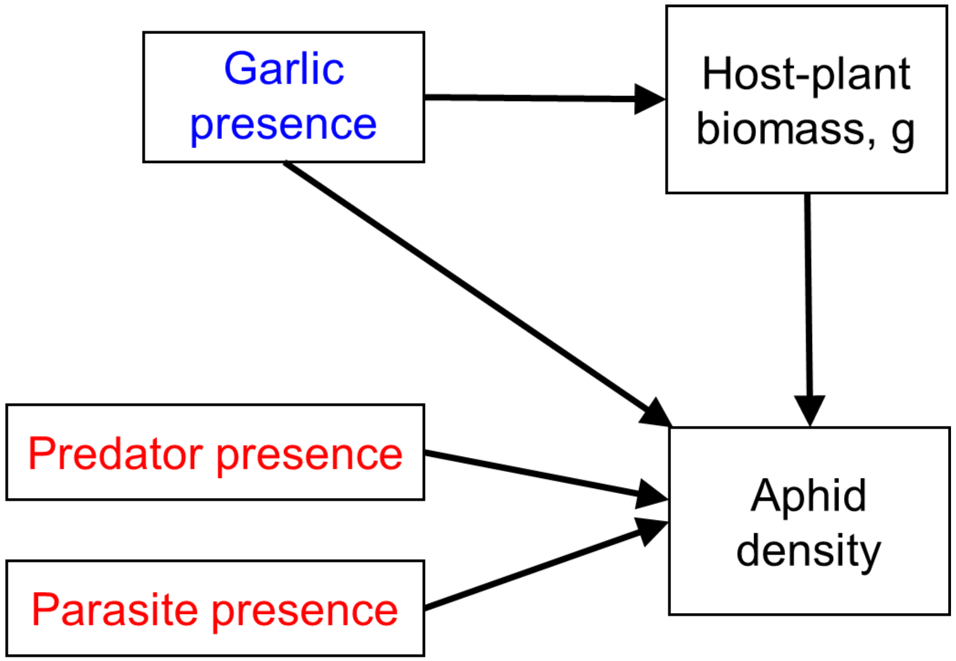
Positive definite S matrix	Check for multicolinearity in each single regression model:
	`library(car)`
	`vif(m2) # vif ≤ 2 (no collinearity)`

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

Example

Categorical Exogenous Variable

```
# specify and fit the model in lavaan

sem_mod <- ' aphid ~ host_plant + garlic + predator +
parasite

host_plant ~ garlic

'
```



Example

Categorical Exogenous Variable

```
# specify and fit the model in lavaan

sem_mod <- ' aphid ~ host_plant + garlic + predator +
parasite

host_plant ~ garlic

'

fit <- sem(sem_mod,
           test="Satorra-Bentler", data=aphid_data)



summary(fit, standardize = T, rsq=T, fit.measures=TRUE)
```

Example

Categorical Exogenous Variable

```
> summary(fit, standardize = T, rsq=T, fit.measures=TRUE)

Model Test User Model:
                                              Standard      Scaled
  Test Statistic                                 1.658       1.655
  Degrees of freedom                                 2           2
  P-value (Chi-square)                           0.436       0.437
  Scaling correction factor                                  1.002
    Satorra-Bentler correction
…
 Robust Comparative Fit Index (CFI)                          1.000
…
RMSEA                                     0.000            0.000
90 Percent confidence interval - upper    0.153            0.153
  P-value H_0: RMSEA <= 0.050             0.558            0.559
…
SRMR                                      0.025            0.025
```

How to present fit statistics?

χ^2 = 1.65, DF=2, n=150, p = 0.43

RMSEA=0, (CI = 0, 0.15) , p_{RMSEA}=0.55,

CFI=1.00;

SRMR=0.025

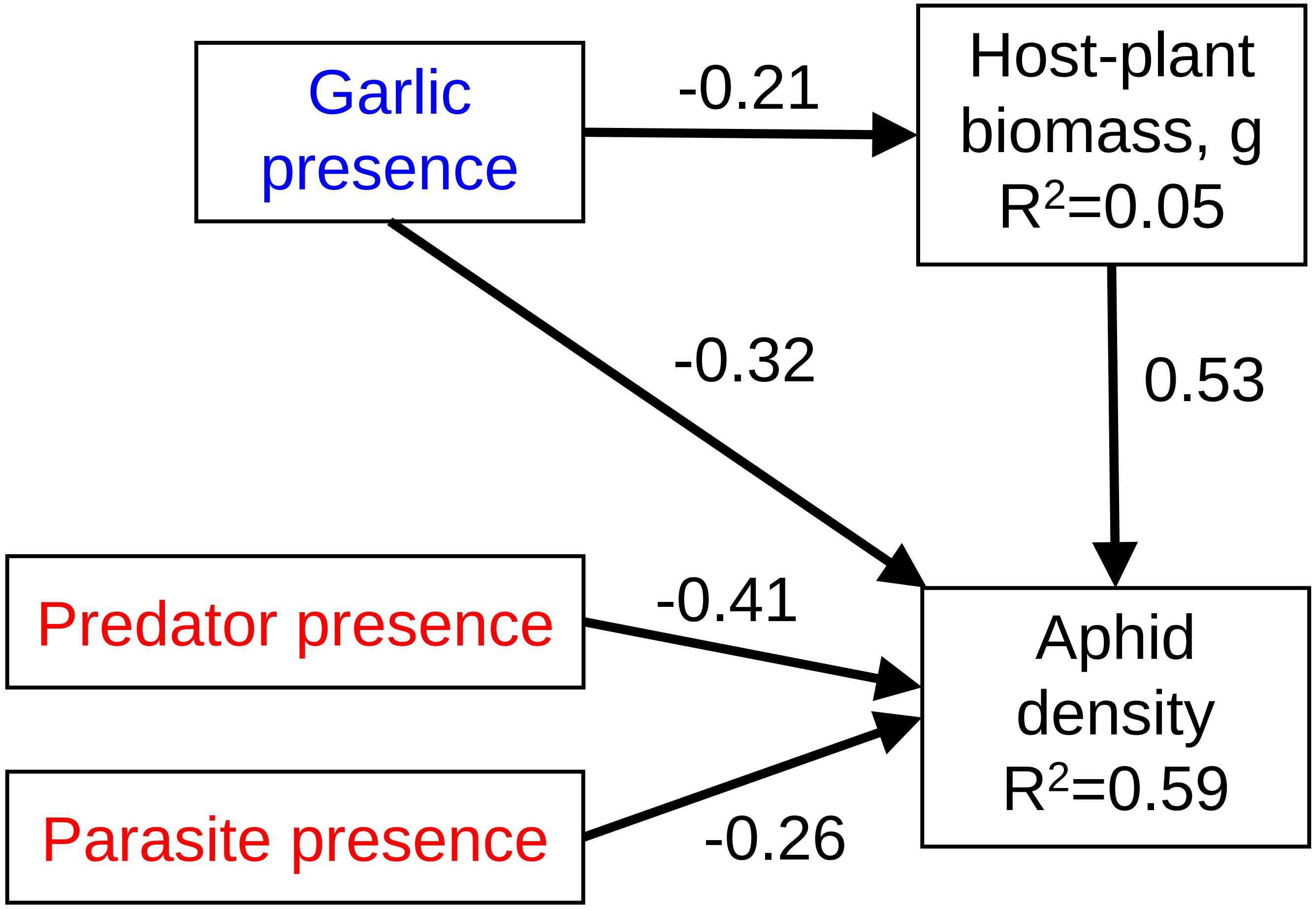
Example

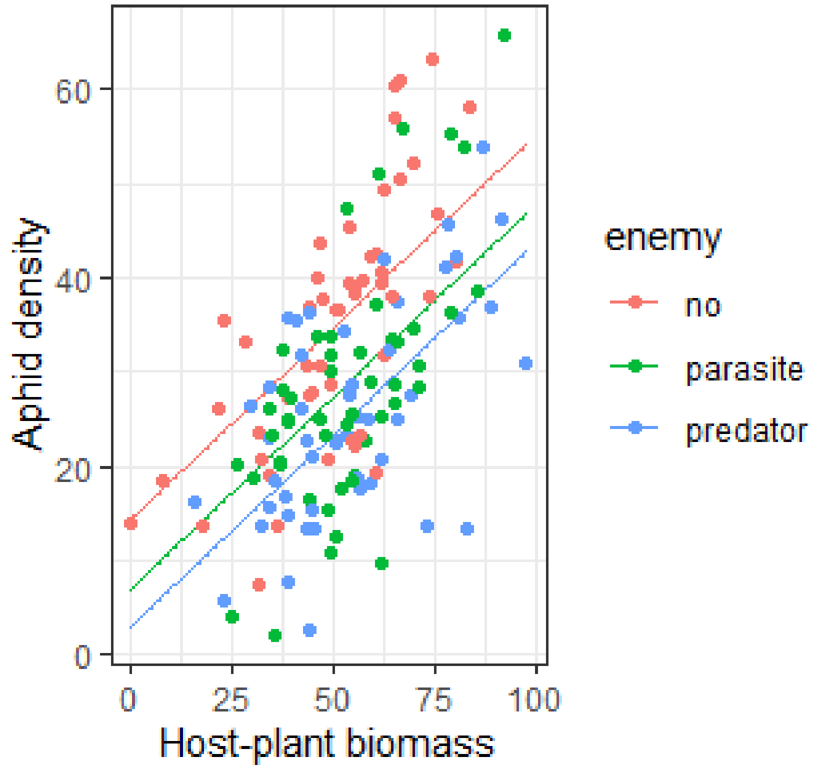
Categorical Exogenous Variable

```
> summary(fit, standardize = T, rsq=T, fit.measures=TRUE)
…
Regressions:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
  aphid ~
    host_plant        0.408    0.041    9.925    0.000    0.408    0.534
    garlic           -8.506    1.437   -5.921    0.000   -8.506   -0.321
    predator        -11.372    1.707   -6.663    0.000  -11.372   -0.405
    parasite         -7.375    1.712   -4.309    0.000   -7.375   -0.262
  host_plant ~
    garlic           -7.570    2.769   -2.734    0.006   -7.570   -0.218
Variances:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
   .aphid            72.753    8.401    8.660    0.000   72.753    0.414
   .host_plant      287.445   33.191    8.660    0.000  287.445    0.953
R-Square:
                   Estimate
    aphid             0.586
    host_plant        0.047
```

Example

Categorical Exogenous Variable



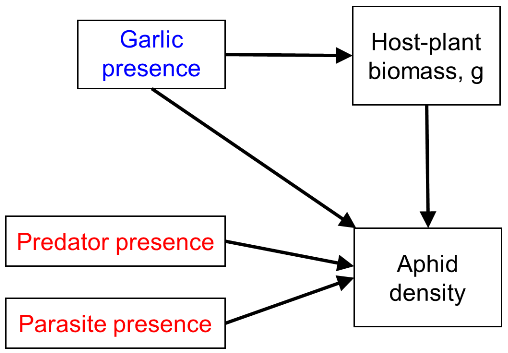


χ^2 = 1.65, DF=2, n=150, p = 0.43   RMSEA=0, (CI = 0, 0.15) , p_{RMSEA}=0.55, CFI=1.00; SRMR=0.025

Example

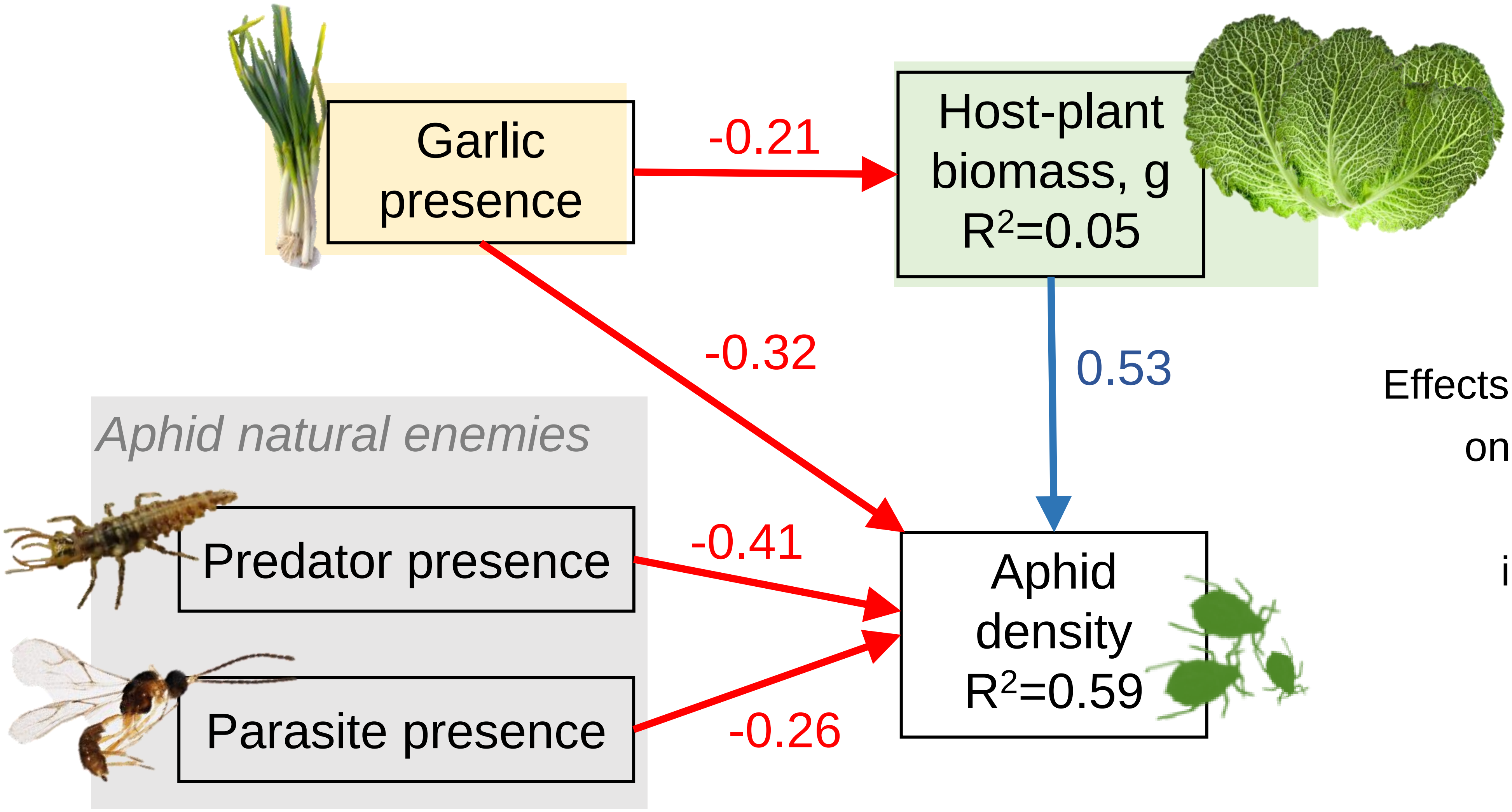
Categorical Exogenous Variable

```
# calculate indirect effects
sem_mod <- ' aphid ~ a1*host_plant + a2*garlic + predator + parasite
             host_plant ~ a3*garlic
                  # define indirect and total effect
                  direct := a2
                  indirect := a3*a1
                  total := direct + indirect
'
fit <- sem(sem_mod, data=aphid_data)
summary(fit, standardize = T, rsq = T, fit.measures=T)
>
Defined Parameters:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
    direct           -8.506    1.437   -5.921    0.000   -8.506   -0.321
    indirect         -3.086    1.171   -2.636    0.008   -3.086   -0.116
    total           -11.592    1.800   -6.439    0.000  -11.592   -0.437
```



Example

Categorical Exogenous Variable



Effects of **garlic presence** on **aphid density**:

direct: -0.32

indirect: -0.12

total: -0.44

χ^2 = 1.65, DF=2, n=150, p = 0.43 RMSEA=0, (CI = 0, 0.15) , p_{RMSEA}=0.55, CFI=1.00; SRMR=0.025

Endogenous Categorical Variables

Approaches when we have Endogenous Categorical Variables:

1) for binary and ordinal variables use the argument ‘ordered’ in ***lavaan*** with fitting function ‘sem’

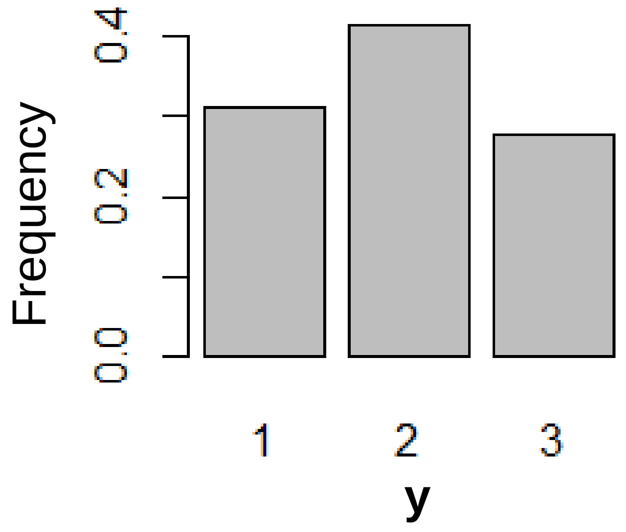
2) for nominal variables (i.e., levels are not ordered) use the factor levels to construct a composite variable.

Endogenous Categorical Variables

- Normal distribution means continuous data

- Ordinal data can not be assumed normal

  Solution: to use the threshold models

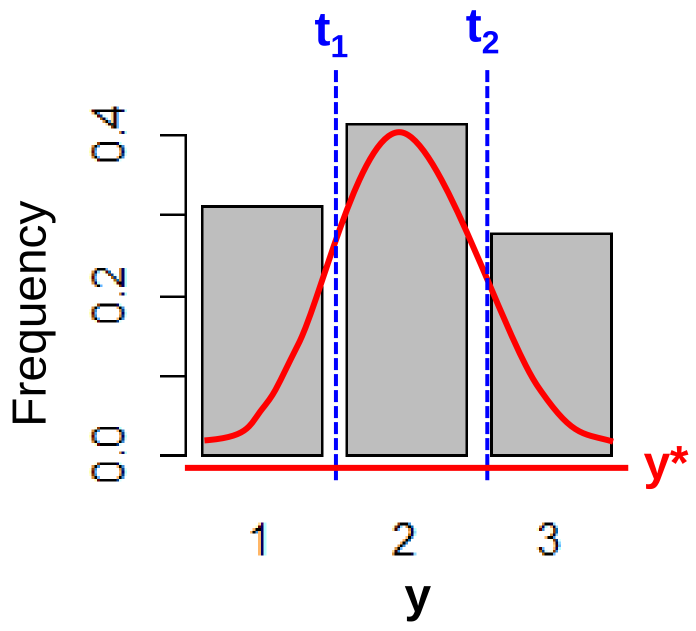


Endogenous Categorical Variables

- Normal distribution means continuous data

- Ordinal data can not be assumed normal

  Solution: to use the threshold models

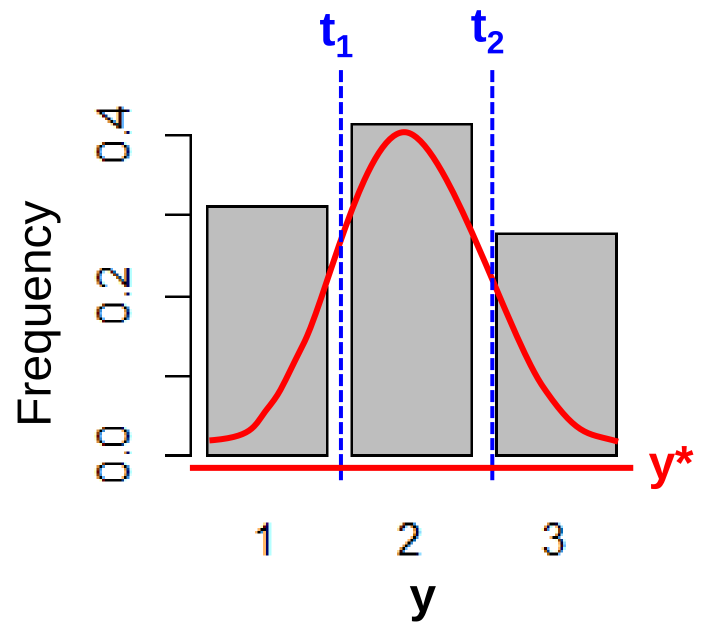


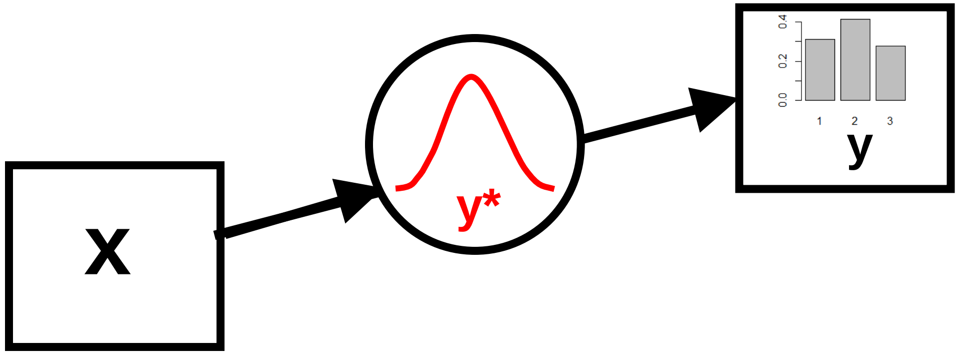
Threshold, for instance t1 is a match beetwing the probability of y=1 and actual percent that the observed data =1.

Endogenous Categorical Variables

- Normal distribution means continuous data
- Ordinal data can not be assumed normal

  Solution: to use the threshold models

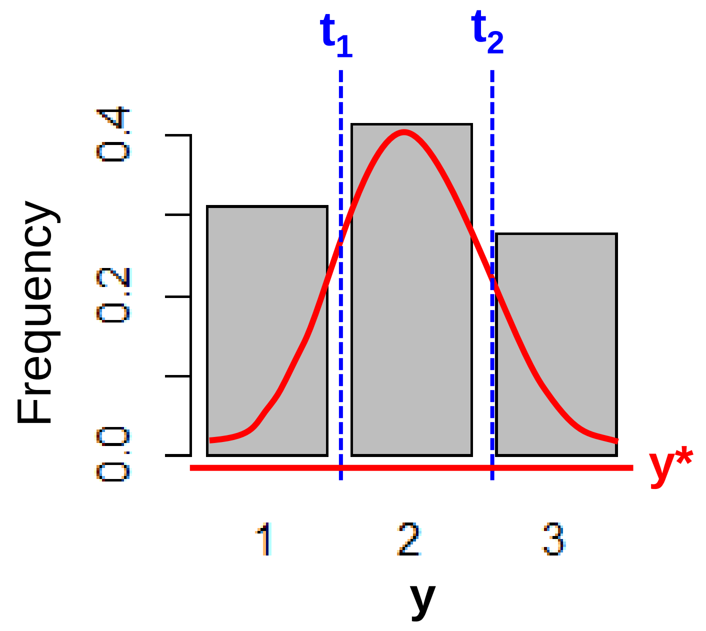




Endogenous Categorical Variables

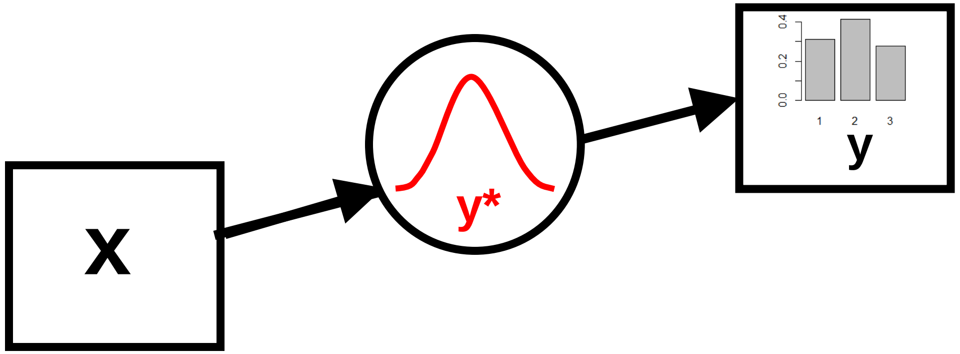
- Normal distribution means continuous data
- Ordinal data can not be assumed normal

Solution: to use the threshold models



Estimation not via ML but via
(diagonally) weighted least squares (D)WLS

$$F_{WLS} = (\boldsymbol{s} - \boldsymbol{\sigma})^{\intercal} \mathbf{W}^{-1} (\boldsymbol{s} - \boldsymbol{\sigma})$$



Example

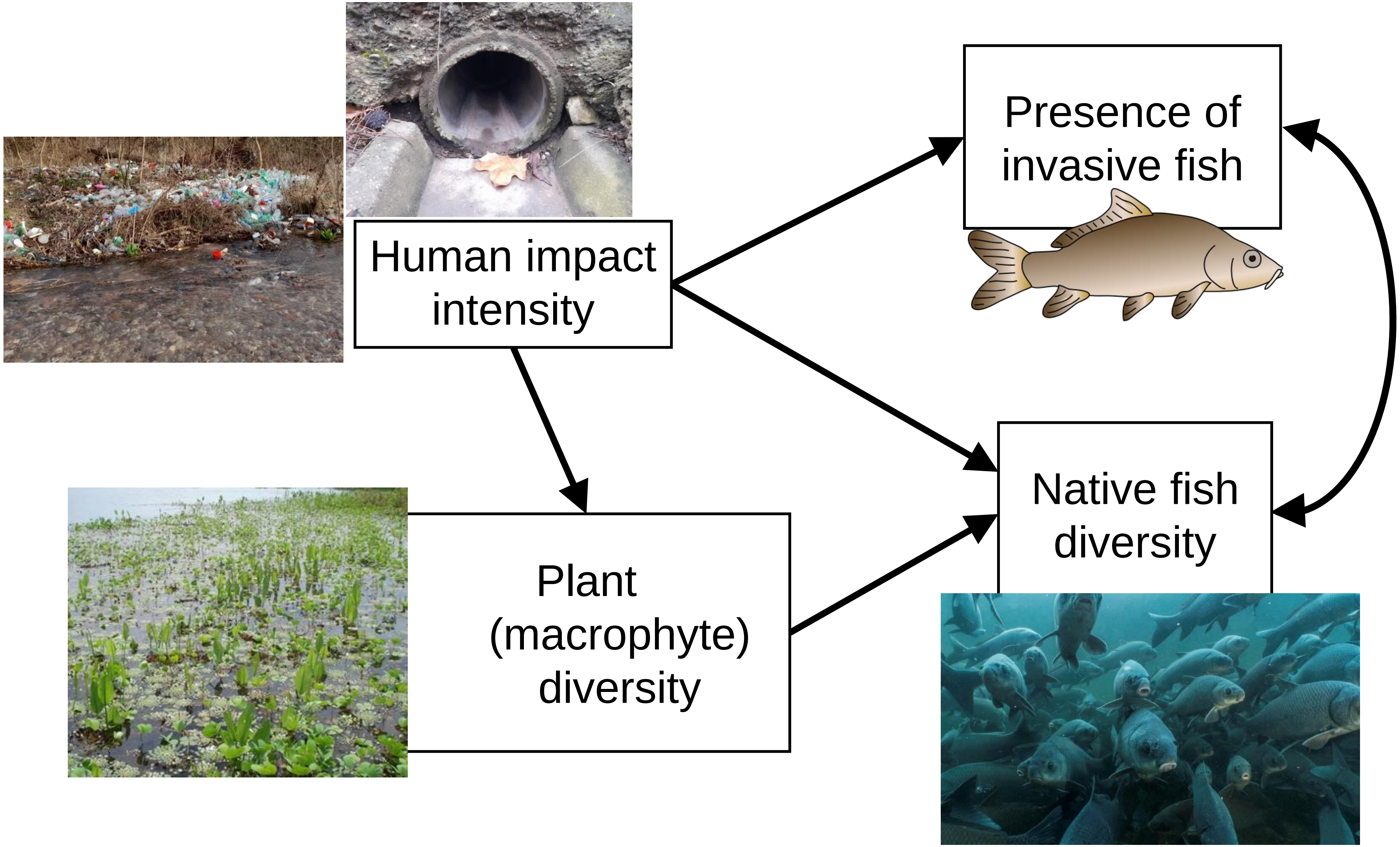
Categorical Endogenous Variable

Human activities affect fish communities in ponds

120 ponds

Example

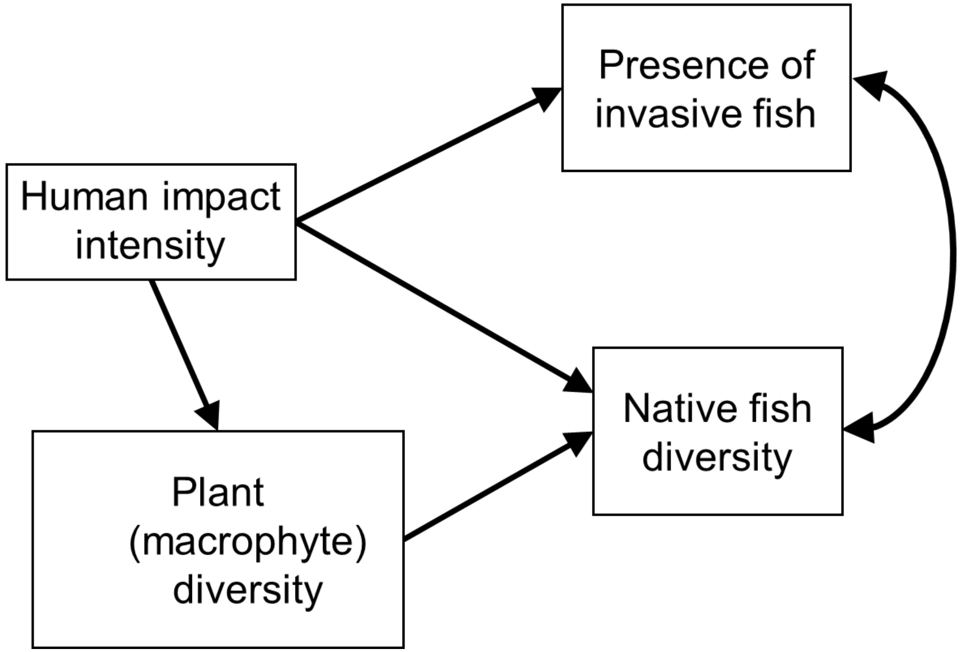
Categorical Endogenous Variable



Example

Categorical Endogenous Variable

```
# Read and check the data
fish_data <- read_csv("Data/Fish_data.csv")
str(fish_data)

sem_mod2 <- ' inv_fish ~  HII
                native_fish ~ plant_div + HII
                plant_div ~ HII
                native_fish ~~ inv_fish
'
fit2 <- sem(sem_mod2, data=fish_data,
                      ordered = c("inv_fish"))

summary(fit2, standardize = T, rsq = T)
```



Example


```
# Read and check the data

Estimator                                         DWLS
  Optimization method                             NLMINB
  Number of model parameters                          10

  Number of observations                             120

Model Test User Model:
                                              Standard      Scaled

  Test Statistic                                 0.022       0.022
  Degrees of freedom                                 1           1
  P-value (Chi-square)                           0.882       0.882
  Scaling correction factor                                  1.000
  Shift parameter                                            0.000
       simple second-order correction

Parameter Estimates:

  Standard errors                           Robust.sem
```

Example

Categorical Endogenous Variable

```
# Read and check the data
…                                              Standard      Scaled
Comparative Fit Index (CFI)                     1.000         1.000
  Tucker-Lewis Index (TLI)                      1.085         1.098

  Robust Comparative Fit Index (CFI)                             NA
  Robust Tucker-Lewis Index (TLI)                                NA
…
RMSEA                                            0.000        0.000
  90 Percent confidence interval - lower         0.000        0.000
  90 Percent confidence interval - upper         0.121        0.121
  P-value H_0: RMSEA <= 0.050                    0.898        0.898
  P-value H_0: RMSEA >= 0.080                    0.081        0.081

  Robust RMSEA                                                   NA
…
SRMR                                             0.007        0.007
```

robust RMSA and other fit measures are not calculated in DWLS

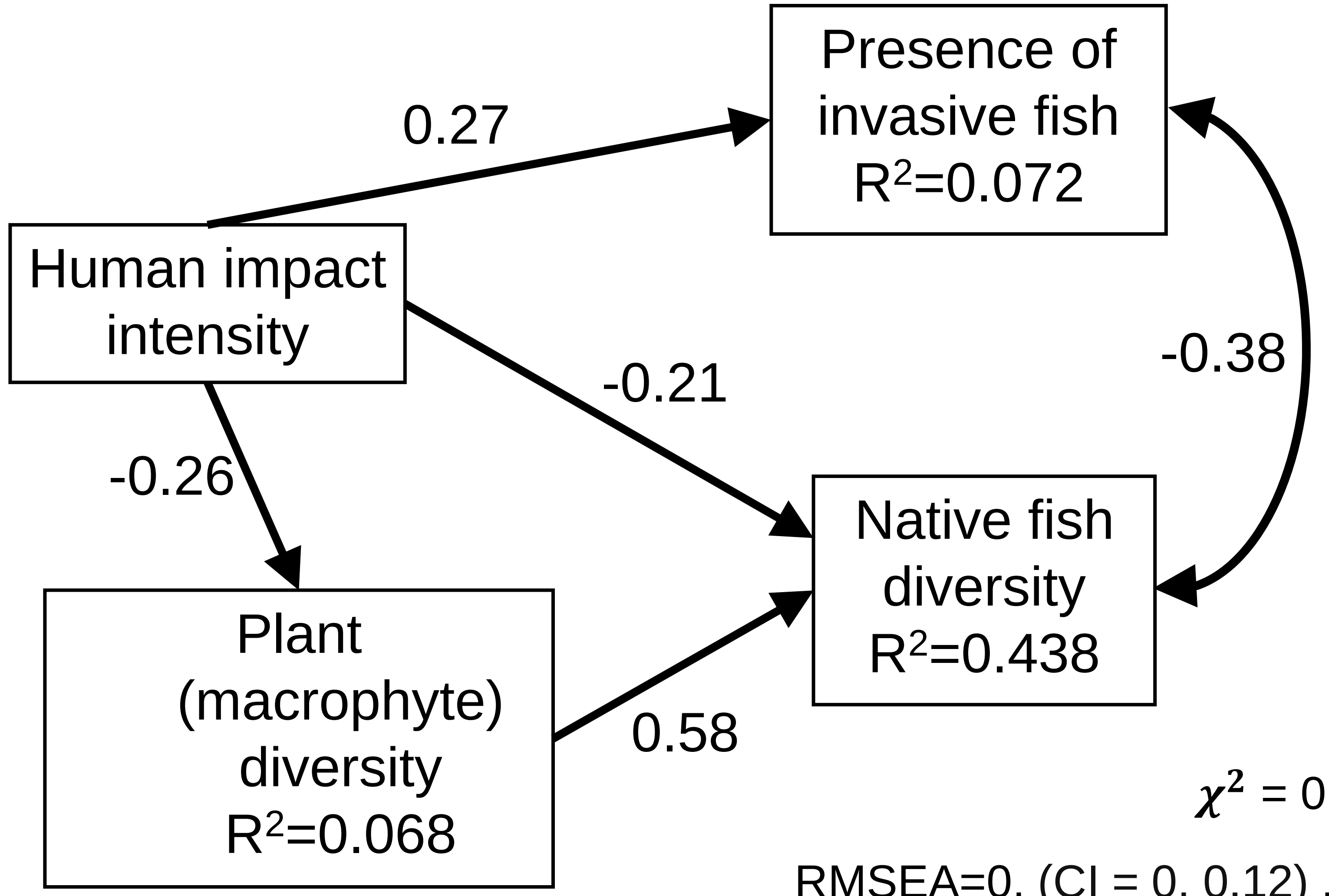
Use standard measures

Example


```
Regressions:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
  inv_fish ~
    HII               0.308    0.128    2.411    0.016    0.308    0.268
  native_fish ~
    plant_div         0.475    0.059    7.994    0.000    0.475    0.576
    HII              -1.186    0.424   -2.797    0.005   -1.186   -0.210
  plant_div ~
    HII              -1.785    0.695   -2.569    0.010   -1.785   -0.261

Covariances:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
 .inv_fish ~~
   .native_fish      -1.466    0.572   -2.561    0.010   -1.466   -0.383

Thresholds:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
    inv_fish|t1       0.567    0.288    1.969    0.049    0.567    0.546

R-Square:
                   Estimate
    inv_fish          0.072
    native_fish       0.438
    plant_div         0.068
```

Example

Categorical Exogenous Variable



χ^2 = 0.022, DF=1, n=120, p = 0.88

RMSEA=0, (CI = 0, 0.12) , p_{RMSEA}=0.89, CFI=1.00; SRMR=0.007

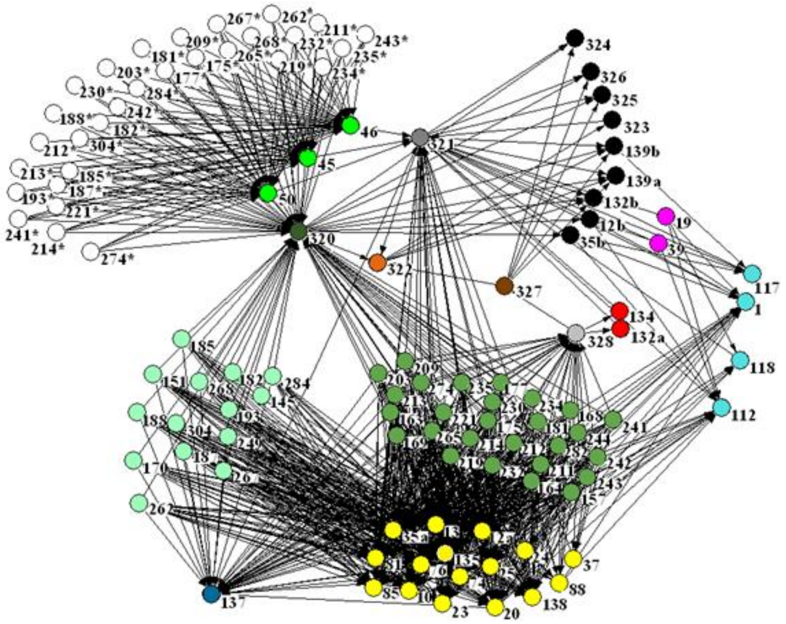
Protocol for treating categorical variables in SEM

<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. Set the values as 0 or 1 and model as numeric (yields a single path coefficient).</td><td rowspan="6"><code>library(lavaan)</code>
<code>sem(…,ordered=c(“categ_varibl”))</code>
• Take care that the levels of your variable have the correct order (e.g. small < medium < large)
• DWMS estimator is used, which corrects for non-normal data and for ordered data.
• Report 'robust' test statistics for χ^2. But report ‘scaled’ RMSA, CFI, SRMR; as no clear suggestions exist regarding the application of these fit indices for non-ML estimators.

<code>library(piecewiseSEM)</code>
Endogenous categorical variables are not implemented in <code>piecewiseSEM</code>. 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 package <code>piecewiseSEM</code></td></tr>
<tr><td rowspan="3">Ordinal variables:
small < medium < large;
yang < middle < old</td><td>1. 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 package <code>piecewiseSEM</code></td></tr>
<tr><td rowspan="2">Nominal variables
study sites
(e.g., site 1, site 2, site 3);
countries;
sampling campaigns</td><td>1. 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><td>Use the factor levels to construct a composite variable.</td></tr>
<tr><td>2. Use package <code>piecewiseSEM</code></td><td>Nominal endogenous categorical variables are not implemented in <code>piecewiseSEM</code></td></tr>
</table>

Day 7 Task 1

Effects of land use on arthropod food webs in grasslands



Food webs

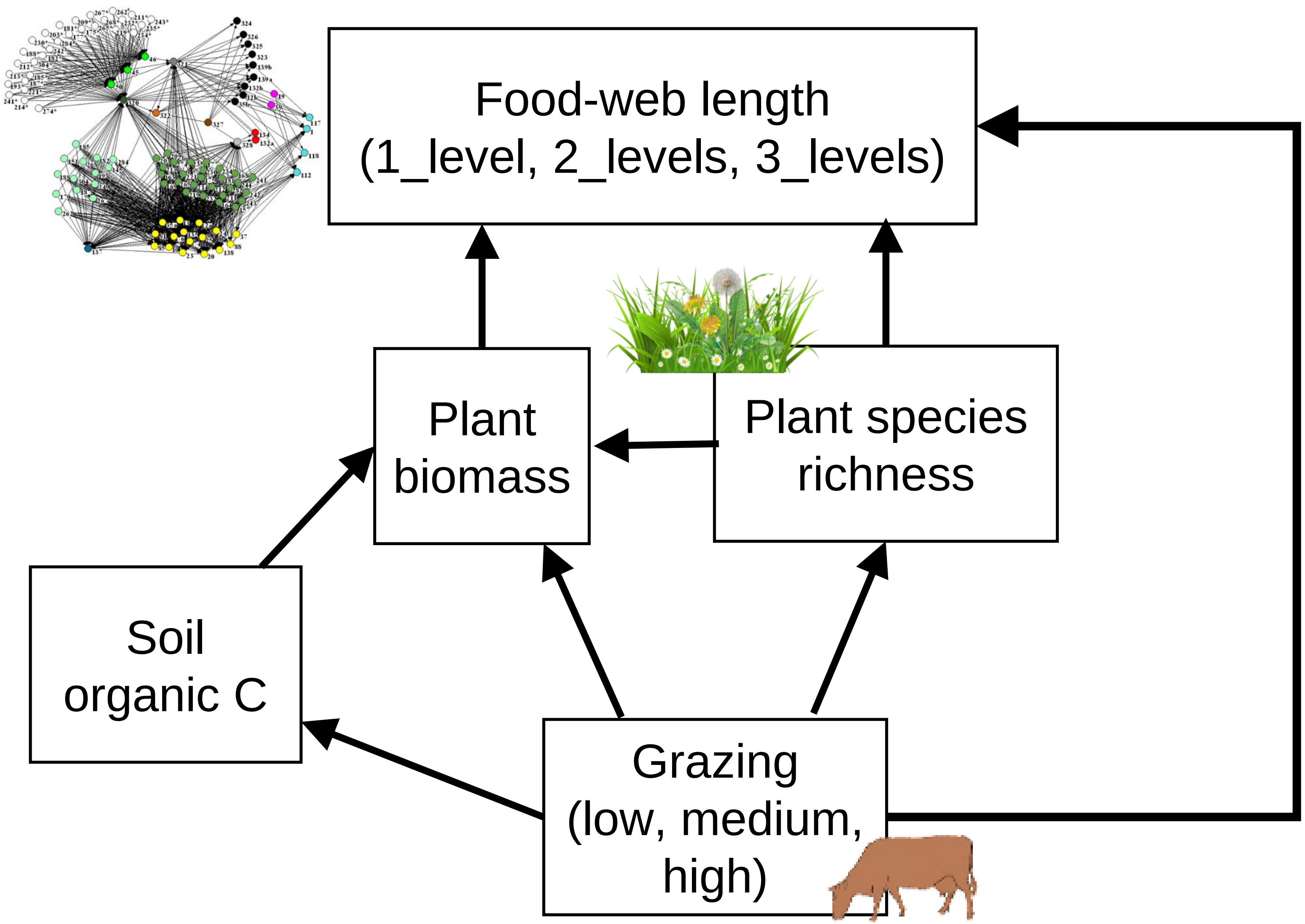
235 grasslands

Food-web length
"1 level": only herbivores and decomposers,
"2 levels": carnivores present in addition to level 1,
"3 levels": omnivores present in addition to level 1 and level 2.

Grazing intensity
("low", "medium", or "high")

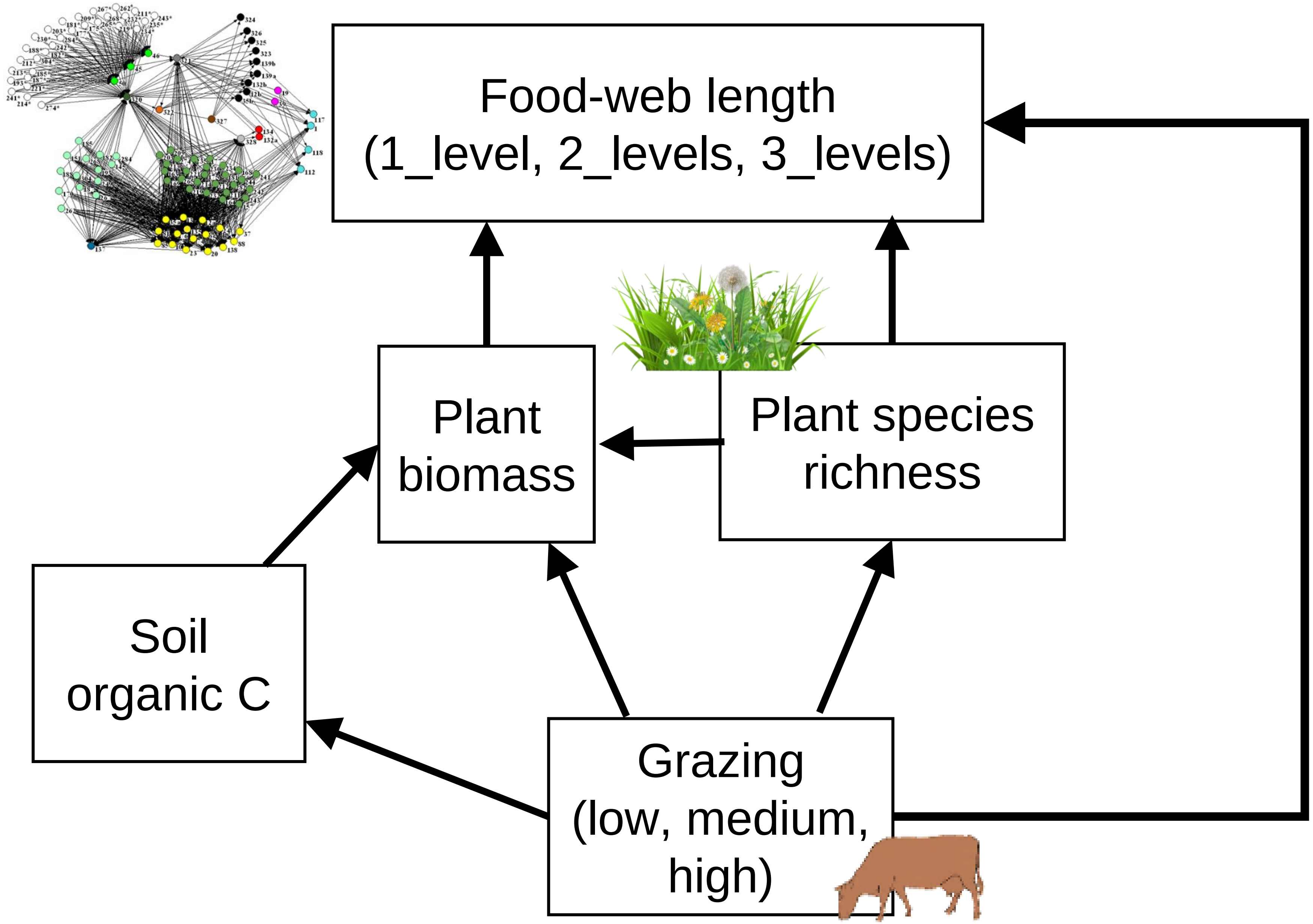
Day 7 Task 1

Effects of land use on food webs in grasslands



Day 7 Task 1

Effects of land use on food webs in grasslands



Day 7 Task 1

Effects of land use on food webs in grasslands

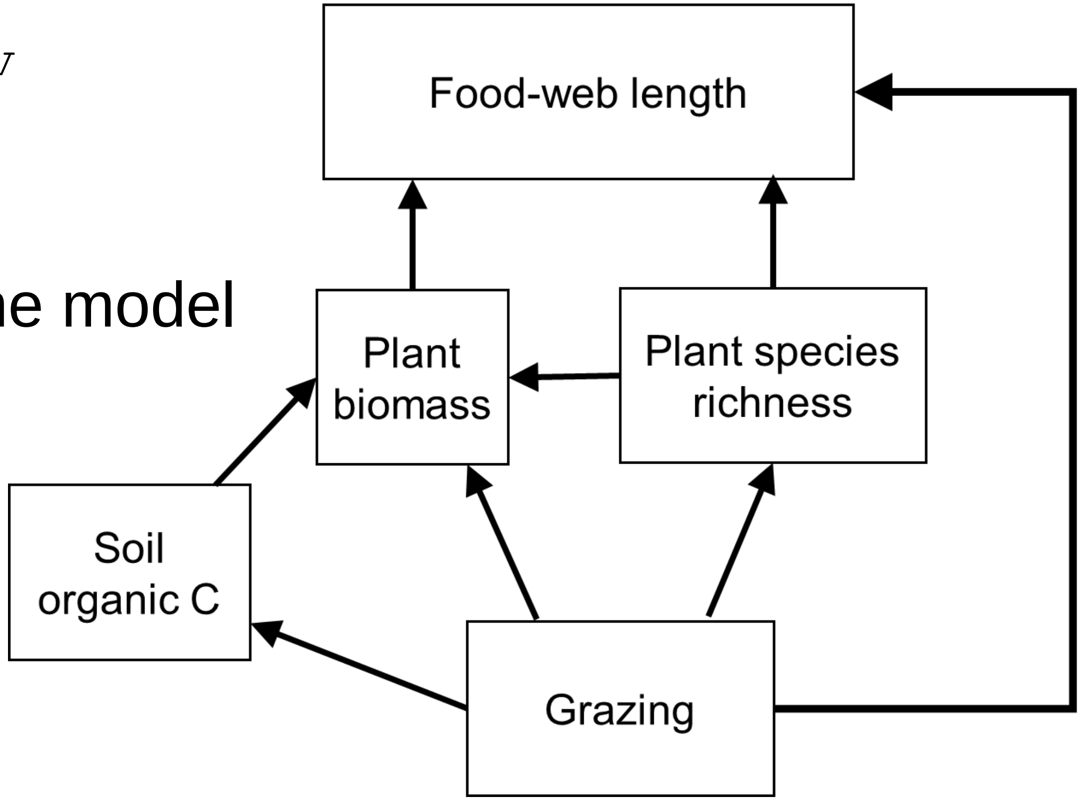
1. Specify the following model in lavaan
   - For this, if needed, recode the categorical variables in a way appropriate for the analysis

3. Fit the model using data `Food-web_data.csv`

4. Get the fit indices

5. Fill in Standardized Coeficients and R^2 for the model

6. Think about how to interpret the results



Day 7 Task 1

Effects of land use on food webs in grasslands

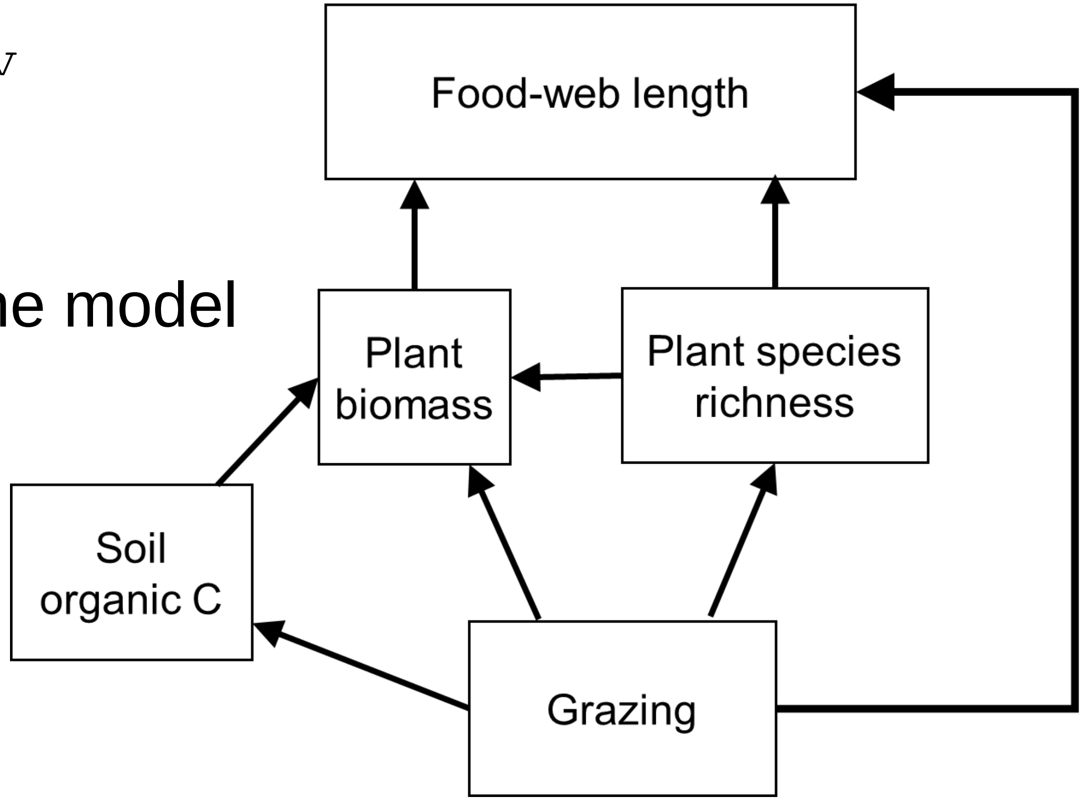
1. Specify the following model in lavaan
   - For this, <u>if needed</u>, recode the categorical variables in a way appropriate for the analysis

3. Fit the model using data `Food-web_data.csv`

4. Get the fit indices

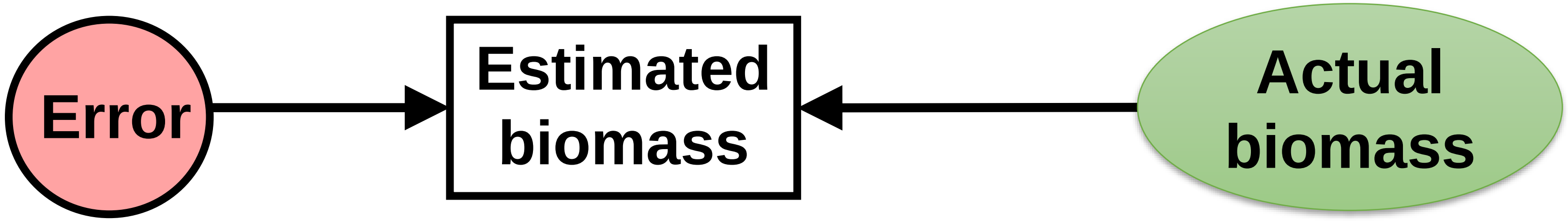
5. Fill in Standardized Coeficients and R^2 for the model

6. Think about how to interpret the results



Day 7 – Part 2 Outline

- Latent Variables in SEM



Outline

Latent Variables in SEM

- What are Latent Variables? Why to use them?
- Multi-indicator Latent Variables
- Fitting Latent Variables

  (Confirmatory Factor Analysis)

Outline

Latent Variables in SEM

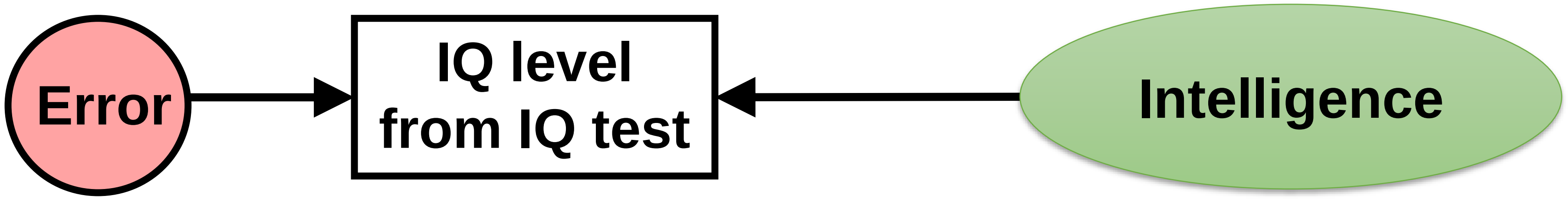
- **What are Latent Variables? Why to use them?**
- Multi-indicator Latent Variables
- Fitting Latent Variables

  (Confirmatory Factor Analysis)

What is Latent Variable?

Latent – hypothetical, hidden

- a variable that is **unmeasured**, but is **hypothesized to exist**
- scientific concept that is **not directly observed**, but is hypothetical **construct**
- can be **approximated using observable indicators**

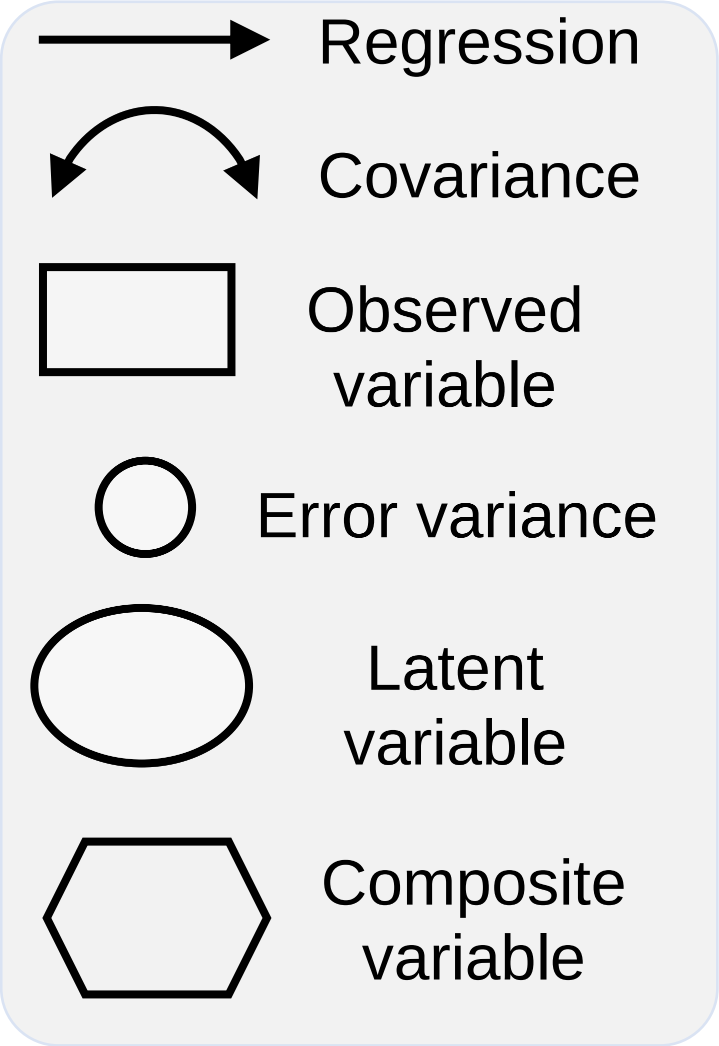


What is Latent Variable?

Specification operators in ‘lavaan’

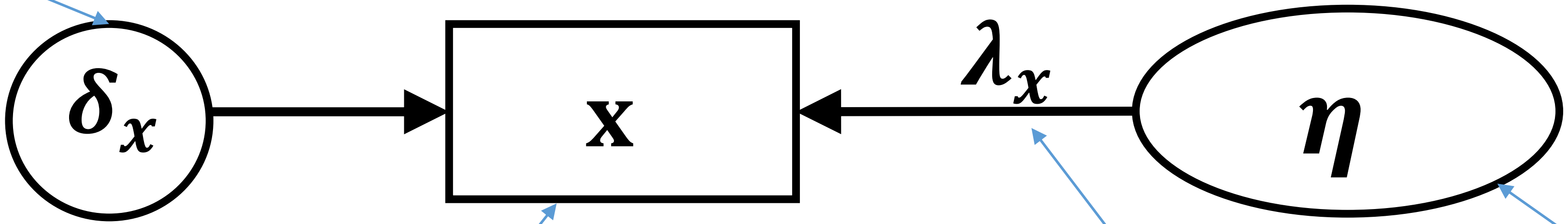
formula type	**operator**	**meaning**
Regression	~	“regressed on”
Correlation	~~	"correlated with"
Intercept	~ 1	"estimates intercept"
Latent variable	=~	"is measured by"
Composite	<~	"is caused by"

Path Diagram Notations:



What is Latent Variable?

The error in the measurement of **x** b $\boldsymbol{\eta}$



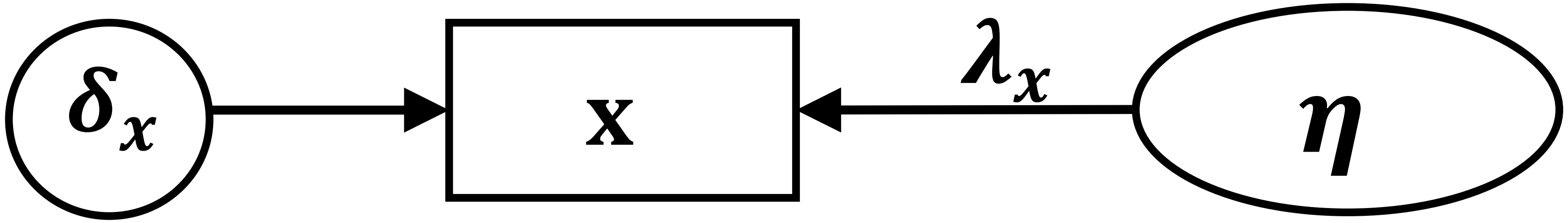
Observed variable
“**manifest indicator**”

“ **factors**” or “**latent traits**”

The relationship between a latent variable and its observed indicator

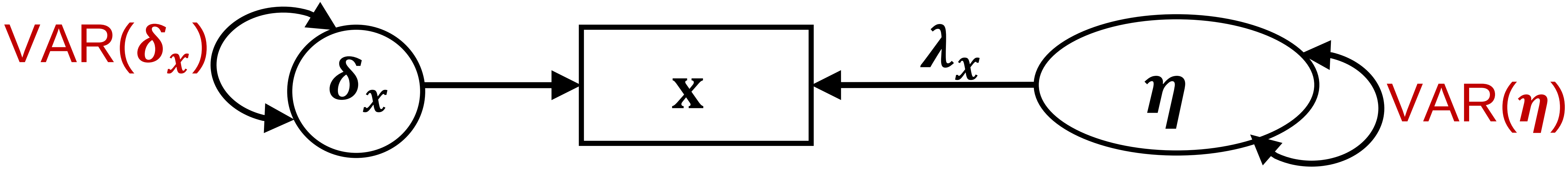
Latent variable

What is Latent Variable?



$$\mathbf{x} = \lambda_x \eta + \delta_x$$

What is Latent Variable?

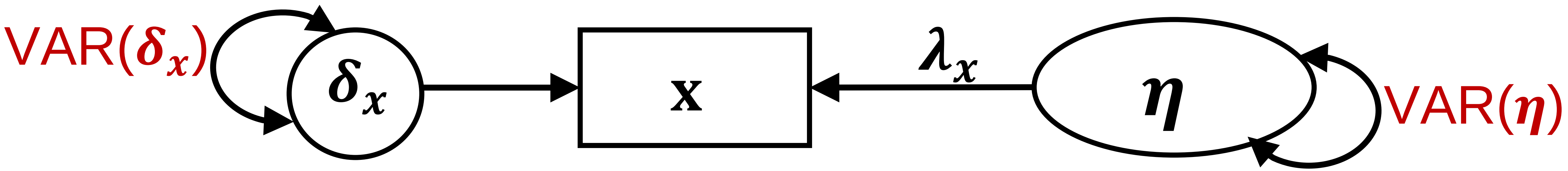


$$\mathbf{x} = \boldsymbol{\lambda_x \eta} + \boldsymbol{\delta_x}$$

$$\boldsymbol{\eta} \sim \mathrm{N}(0, \mathrm{SD}(\boldsymbol{\eta}))$$

$$\boldsymbol{\delta} \sim \mathrm{N}(0, \mathrm{SD}(\boldsymbol{\delta}))$$

What is Latent Variable?



$$\mathbf{x} = \boldsymbol{\lambda_x \eta} + \boldsymbol{\delta_x}$$

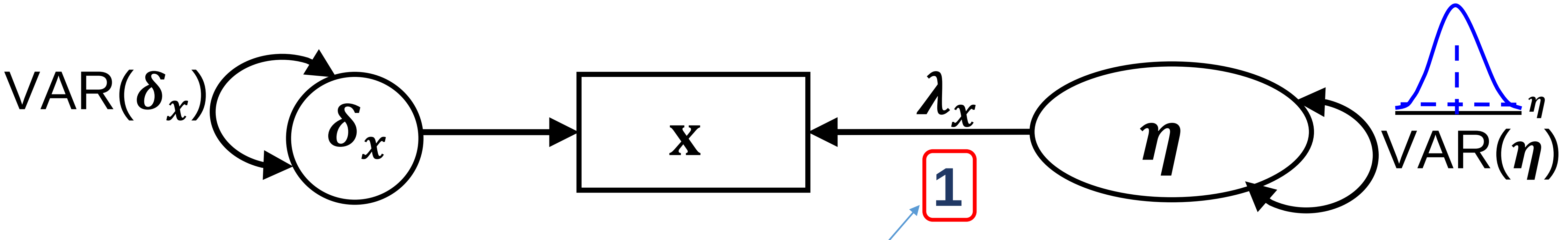
$$\boldsymbol{\eta} \sim \mathrm{N}(0, \mathrm{SD}(\boldsymbol{\eta}))$$
$$\boldsymbol{\delta} \sim \mathrm{N}(0, \mathrm{SD}(\boldsymbol{\delta}))$$

$$\mathrm{VAR}(\mathrm{x}) = \boldsymbol{\lambda_x}^2 \mathrm{VAR}(\boldsymbol{\eta}) + \mathrm{VAR}(\boldsymbol{\delta})$$

How much variance does the LV explain?

$$\frac{\boldsymbol{\lambda_x}^2 \mathrm{VAR}(\boldsymbol{\eta})}{\boldsymbol{\lambda_x}^2 \mathrm{VAR}(\boldsymbol{\eta}) + \mathrm{VAR}(\boldsymbol{\delta})}$$

What is Latent Variable?



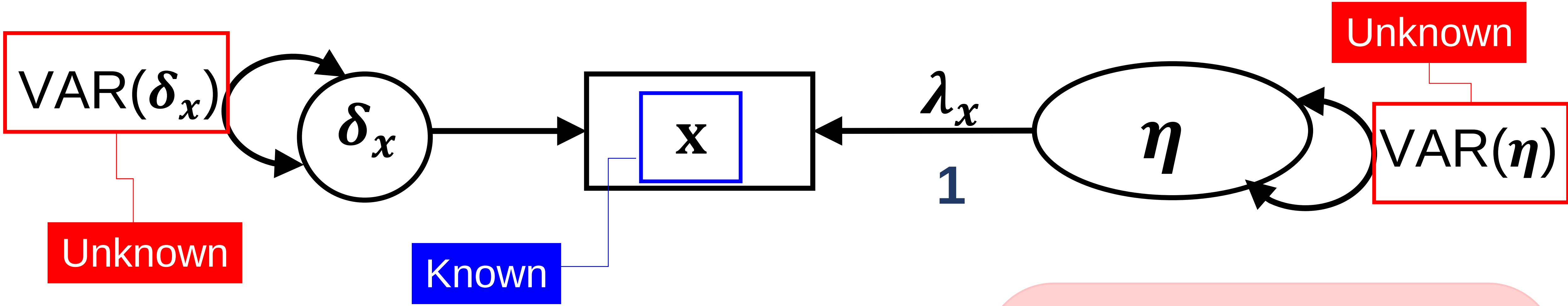
Raw scale coefficient: matches observed (co)variances to parameters $VAR(\delta)$ and $VAR(\eta)$

We explain the data well if:

$VAR(x) = VAR(\eta) + VAR(\delta)$

- What is the scale/unit of our LV?
  It needs to be defined to get the regression weights.

What is Latent Variable?



$$t \le t_{max}$$

$$t_{max} = \frac{s(s+1)}{2} = 1$$

$s = 1$ known

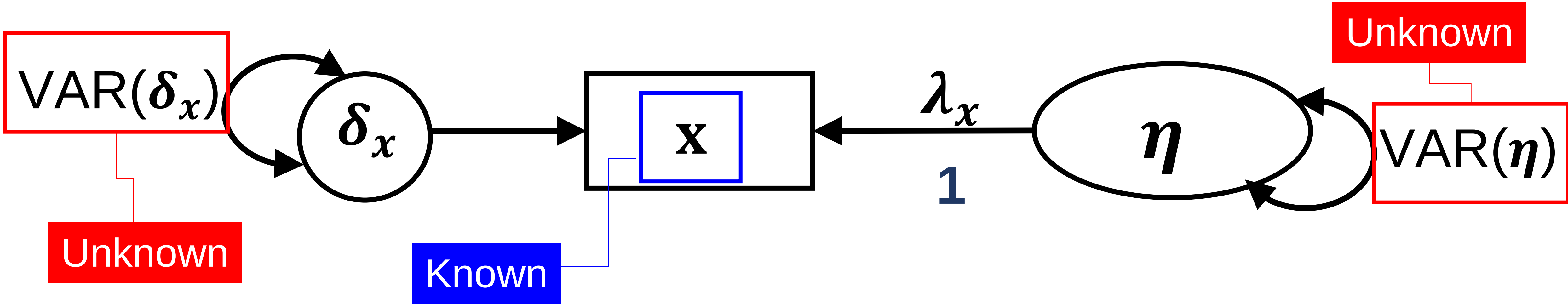
$t = 2$ unknowns

$$\mathbf{DF} = t_{max} - t = -1$$

- Model is not identified

What is Latent Variable?

New



Rules for LV models:

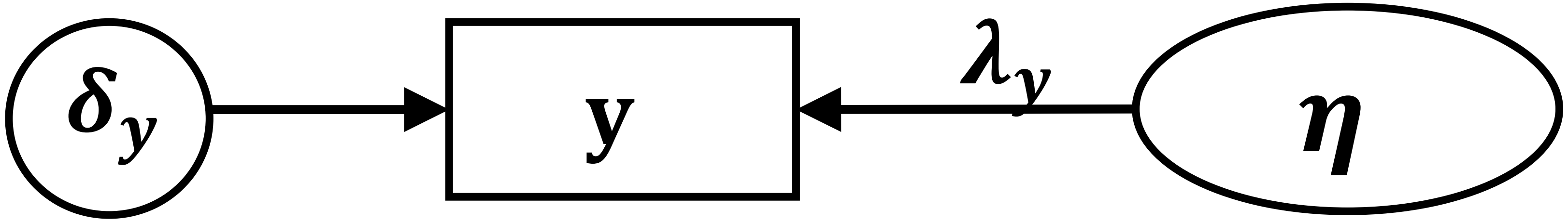
- Scaling of LV
- Non-negative DF

We need at least:

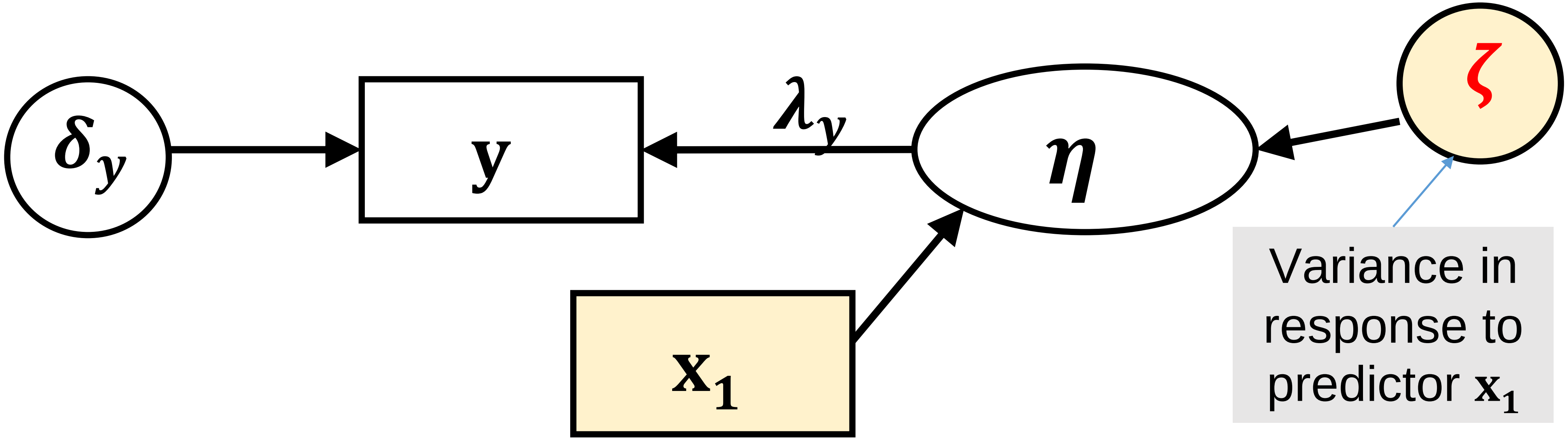
- 3 indicators for a single LV
- 2 indicators per LV for models with multiple (correlated) LVs

What is Latent Variable?

Latent **Exogenous** Variable



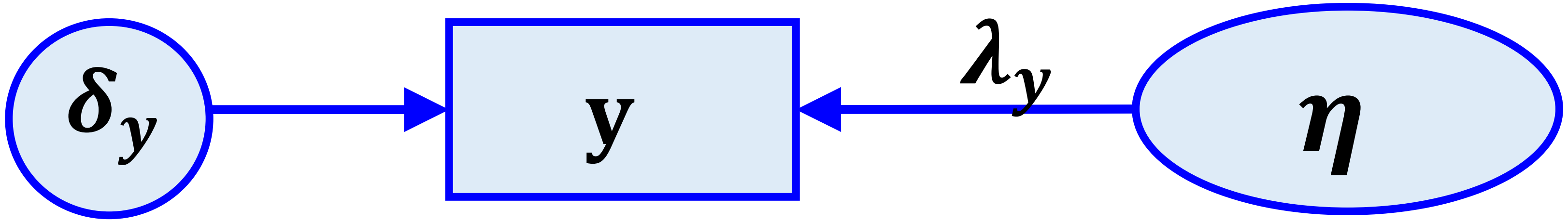
Latent **Endogenous** Variable

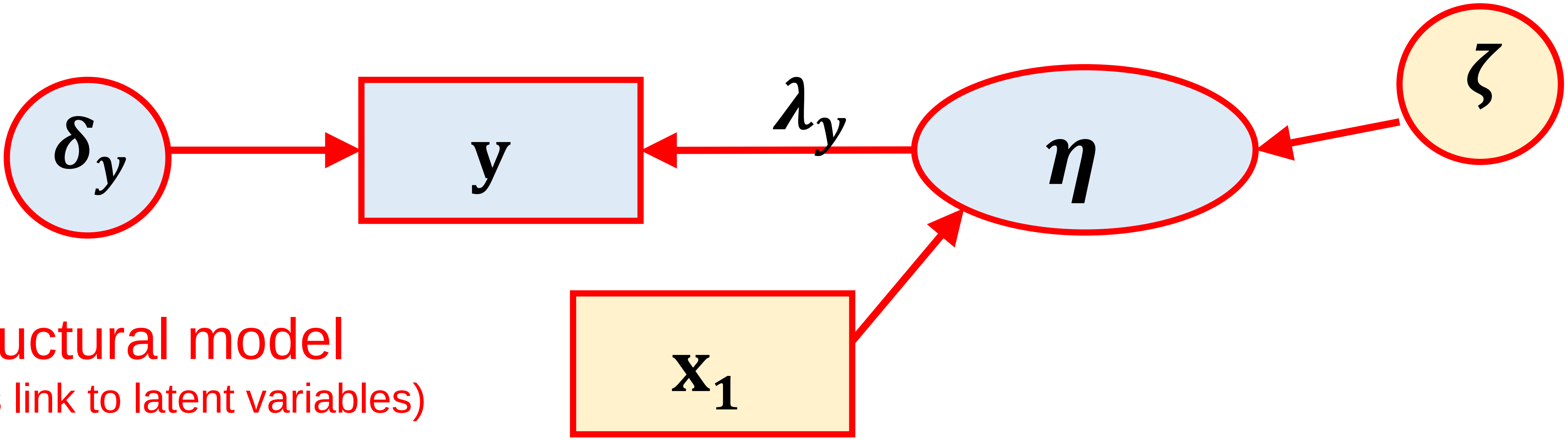


What is Latent Variable?

Measurement model

(solely relates indicators to latent variables)

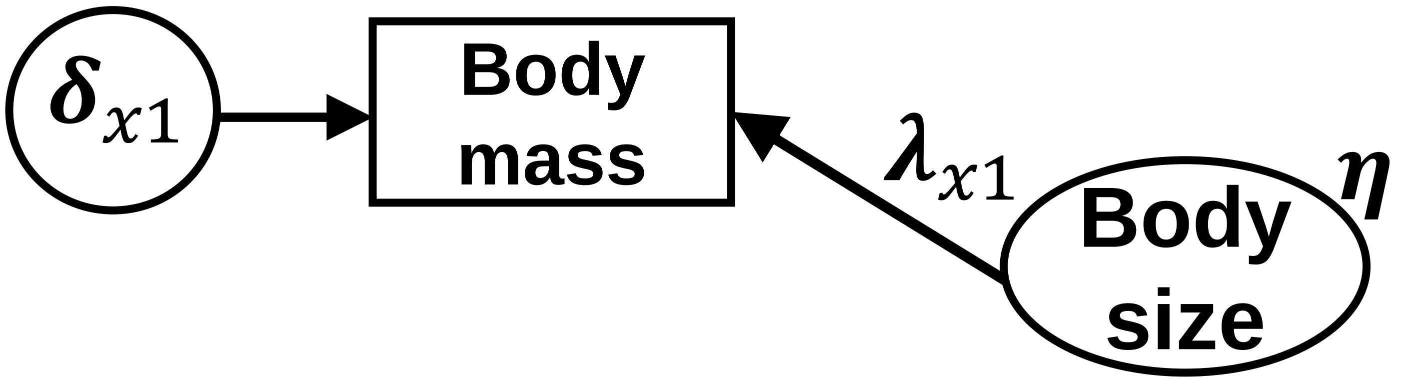




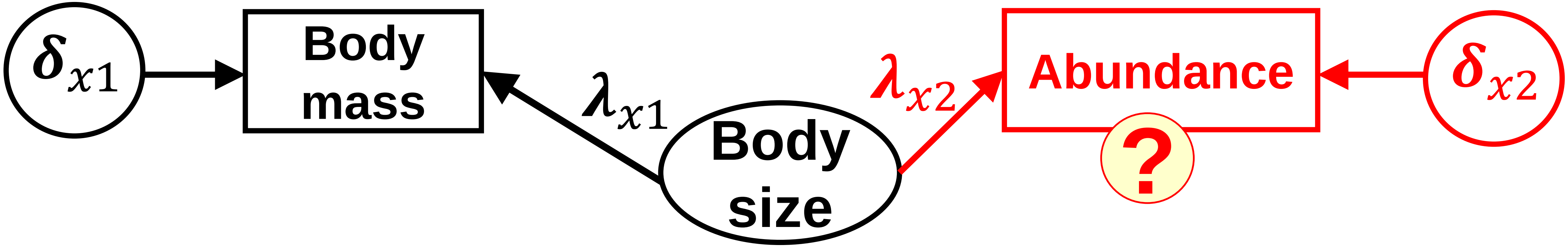
Structural model

(has link to latent variables)

Latent Variables



Latent Variables



- Be sure that the latent variable reflects the actual properties captured by the indicator variables!

Why use Latent Variables?

- Allows estimating complex and **multifaceted concepts**
- **Reduces random error** in construct (latent variable)

random error in dependent variables
→ less precisely measured estimates

random error in independent variables
→ underestimated regression coefficients

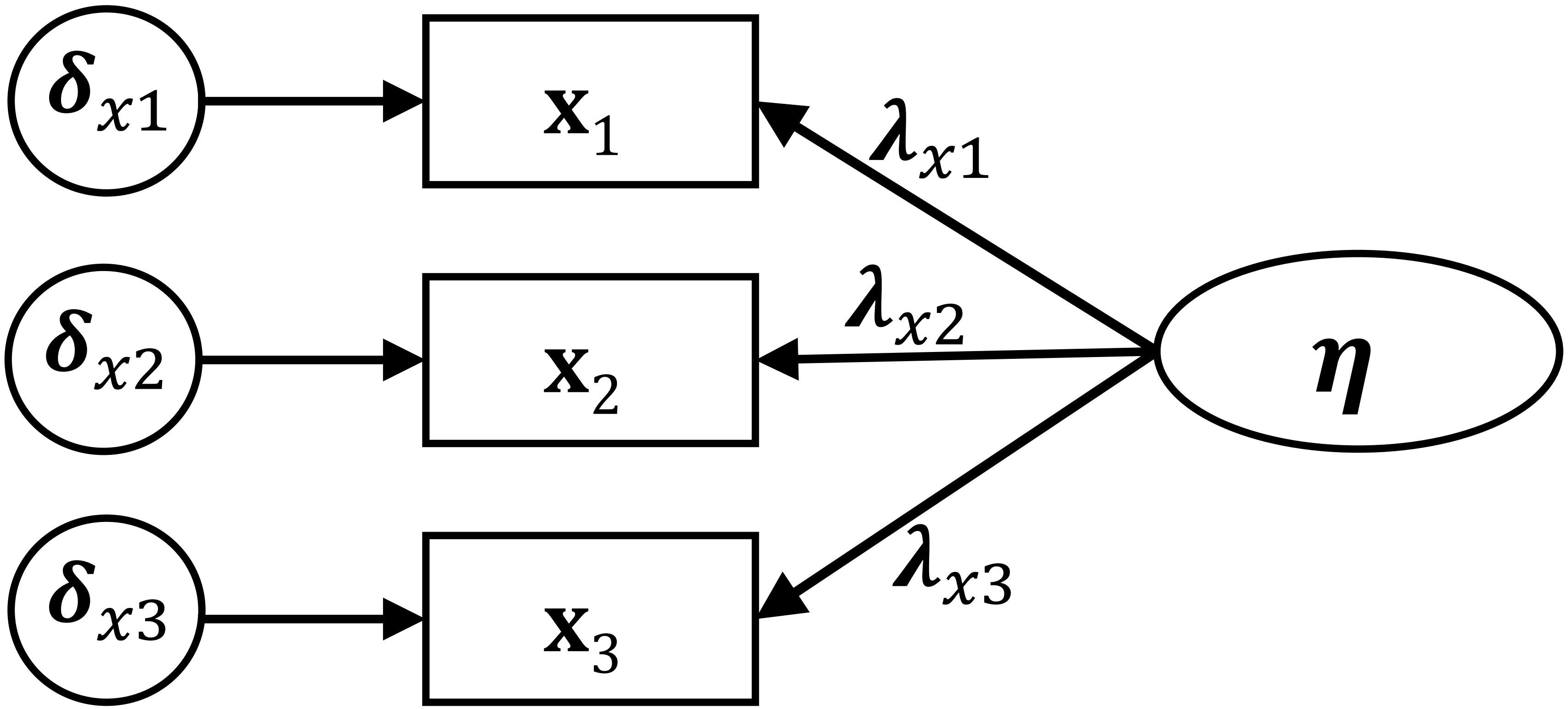
Outline

Latent Variables in SEM

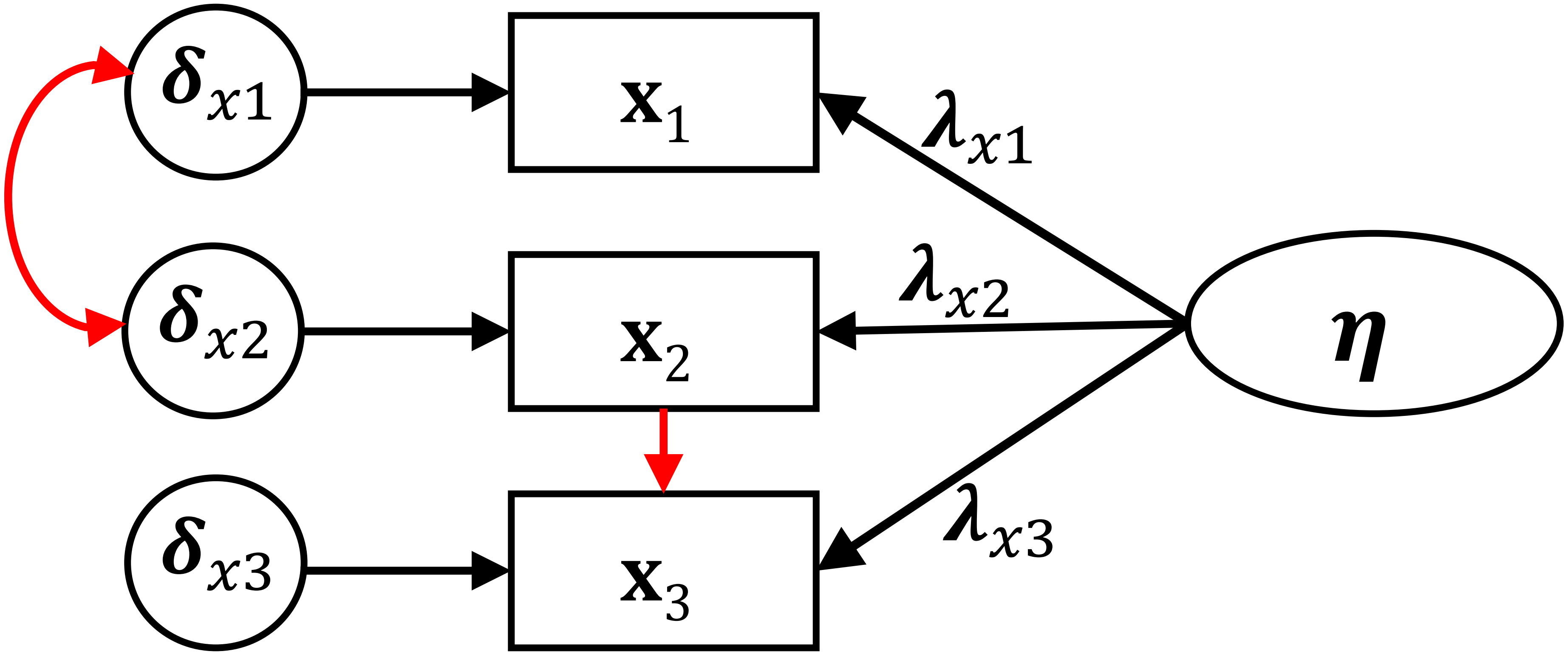
- What are Latent Variables? Why to use them?
- **Multi-indicator Latent Variables**
- Fitting Latent Variables

  (Confirmatory Factor Analysis)

Multi-indicator Latent Variables

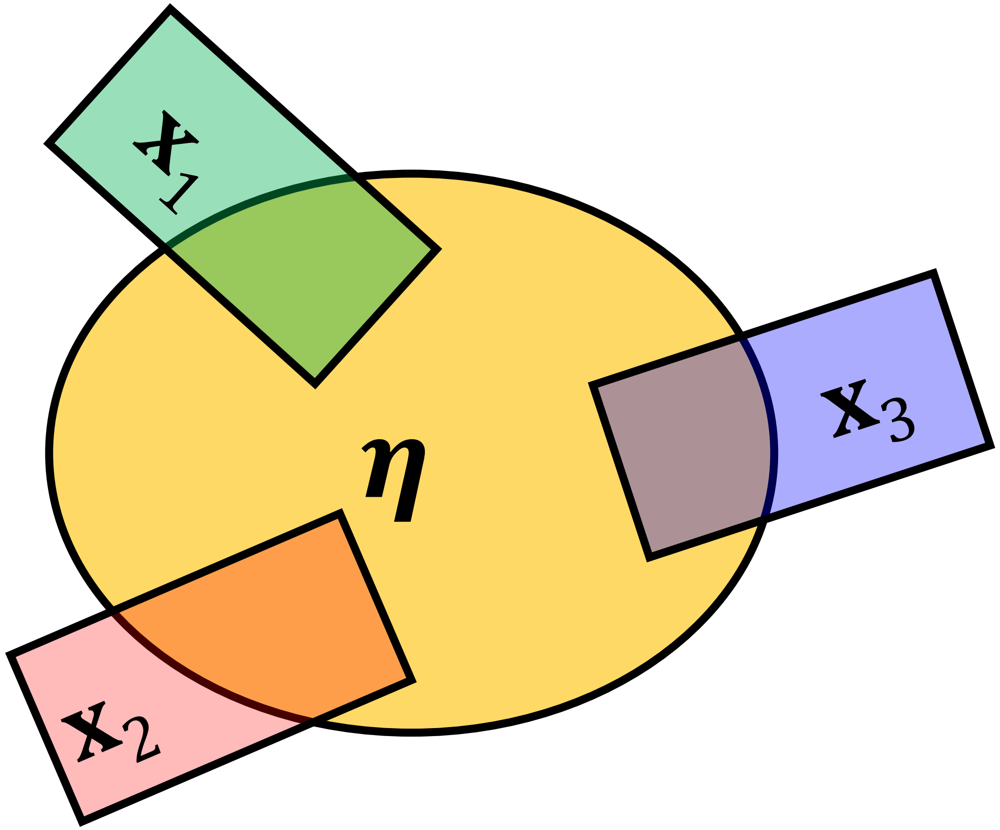


Multi-indicator Latent Variables



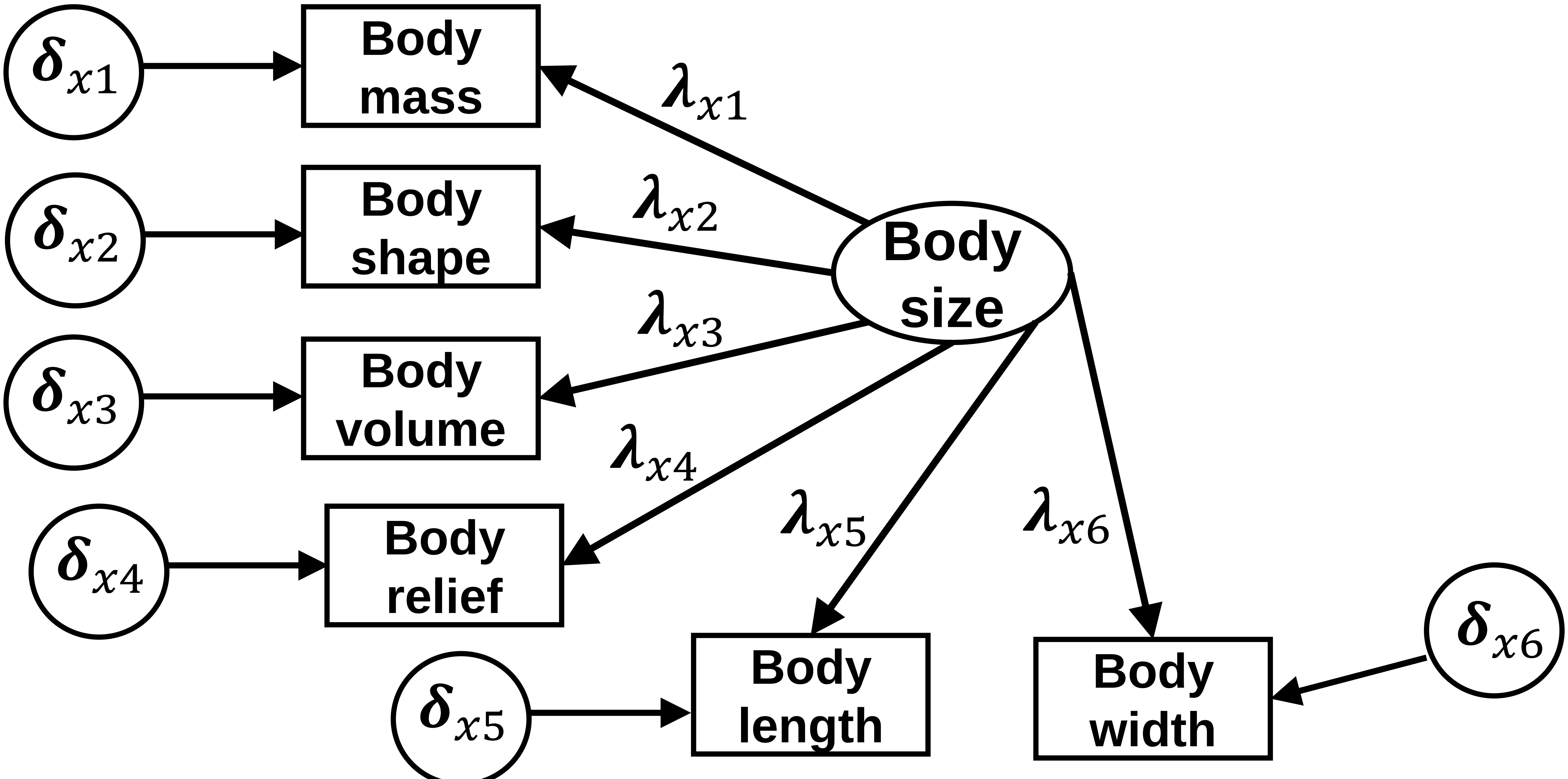
- Indicators may have causal links
- Indicators may covary for other reasons

Multi-indicator Latent Variables



Latent variable $\boldsymbol{\eta}$ represents shared information of observed indicators $\mathbf{x}$

Multi-indicator Latent Variables

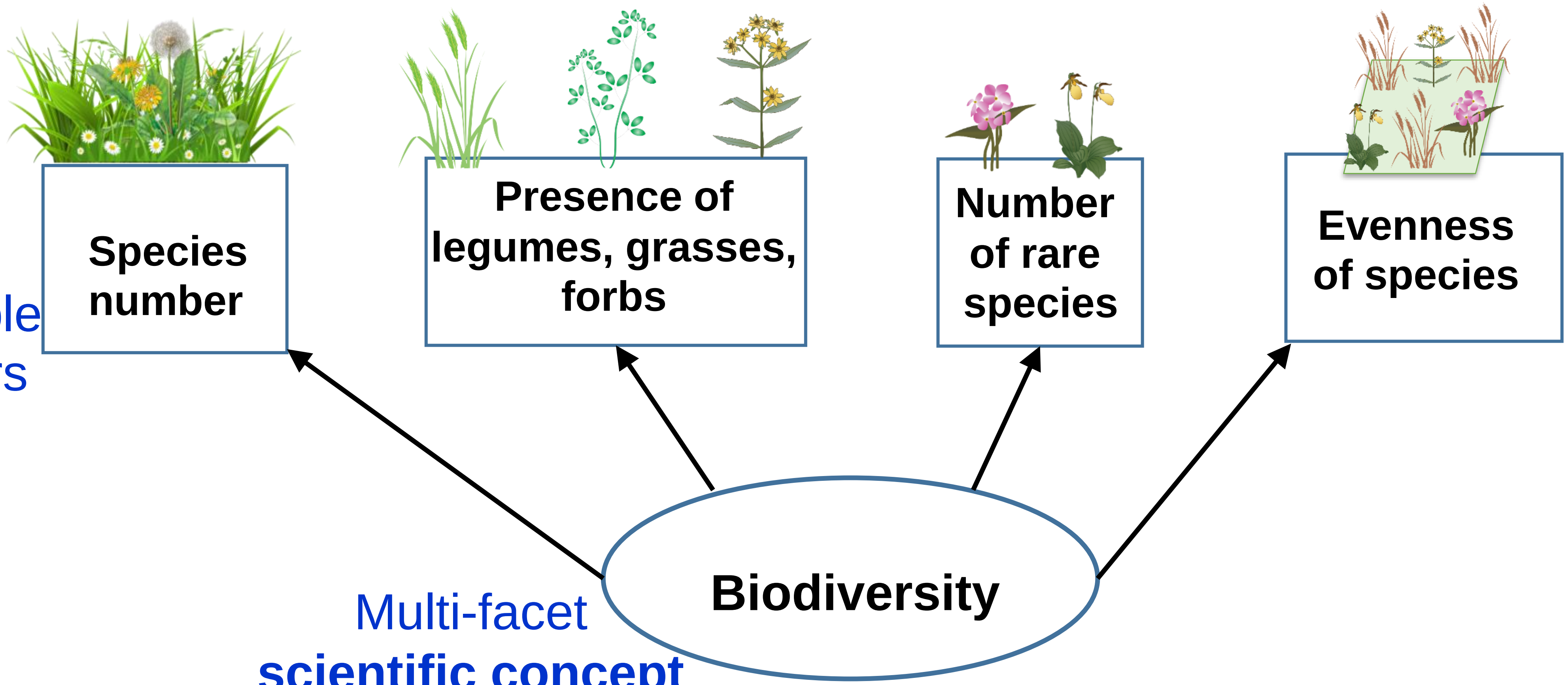


What is Latent Variable?

Example

As Theoretical Constructs

Observable indicators

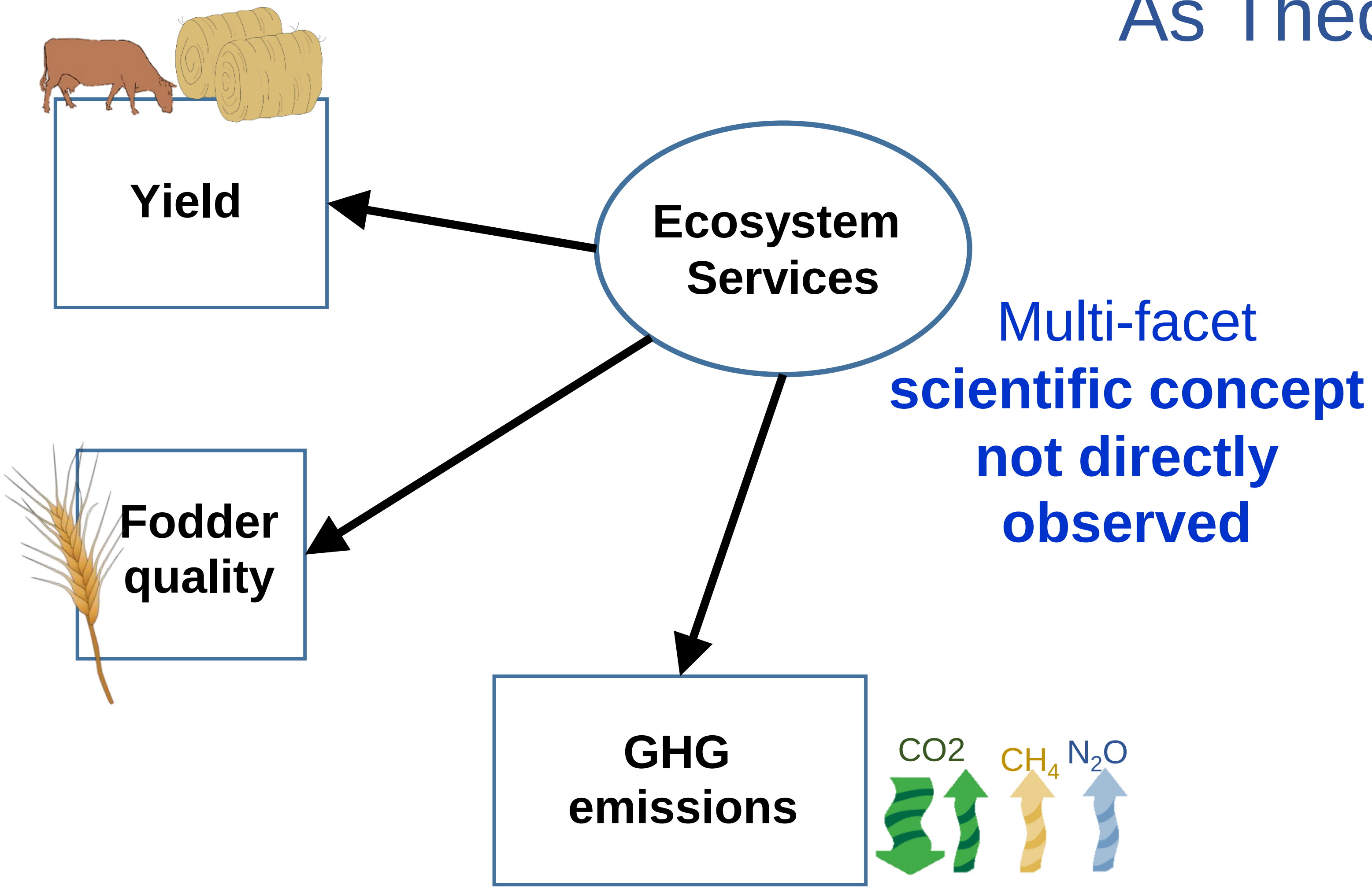


Multi-facet **scientific concept not directly observed**

What is Latent Variable?

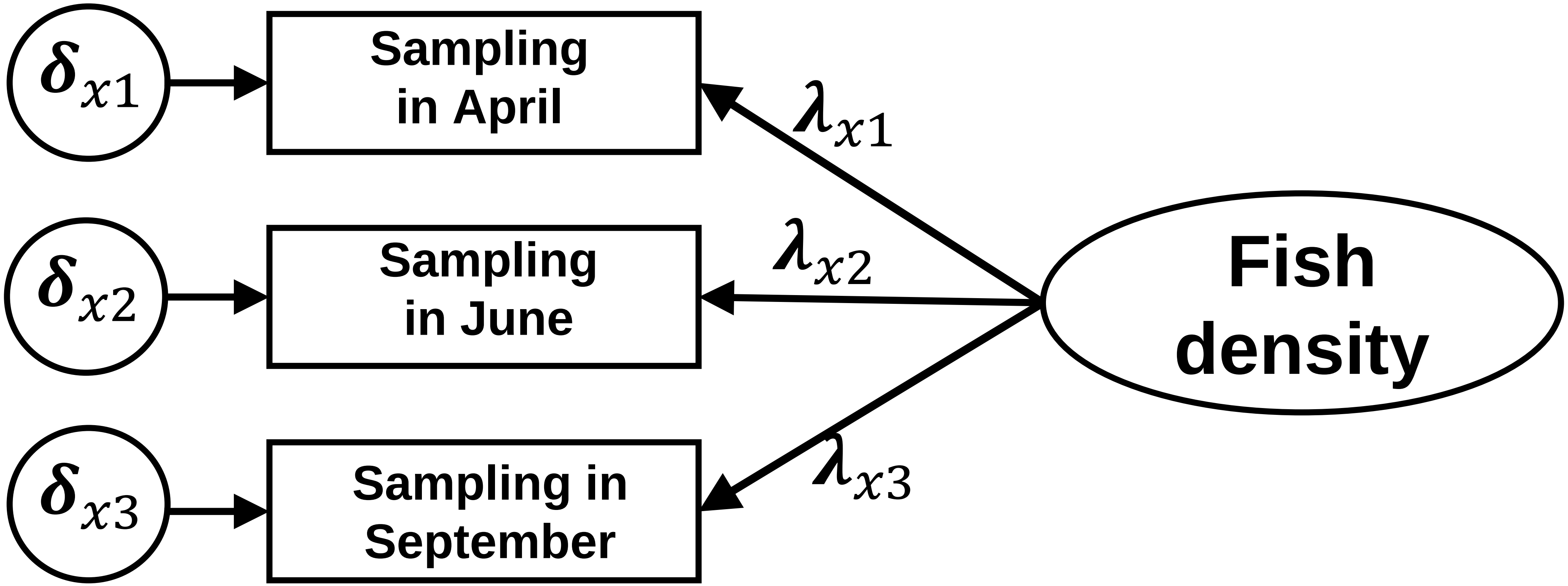
Example

As Theoretical Constructs



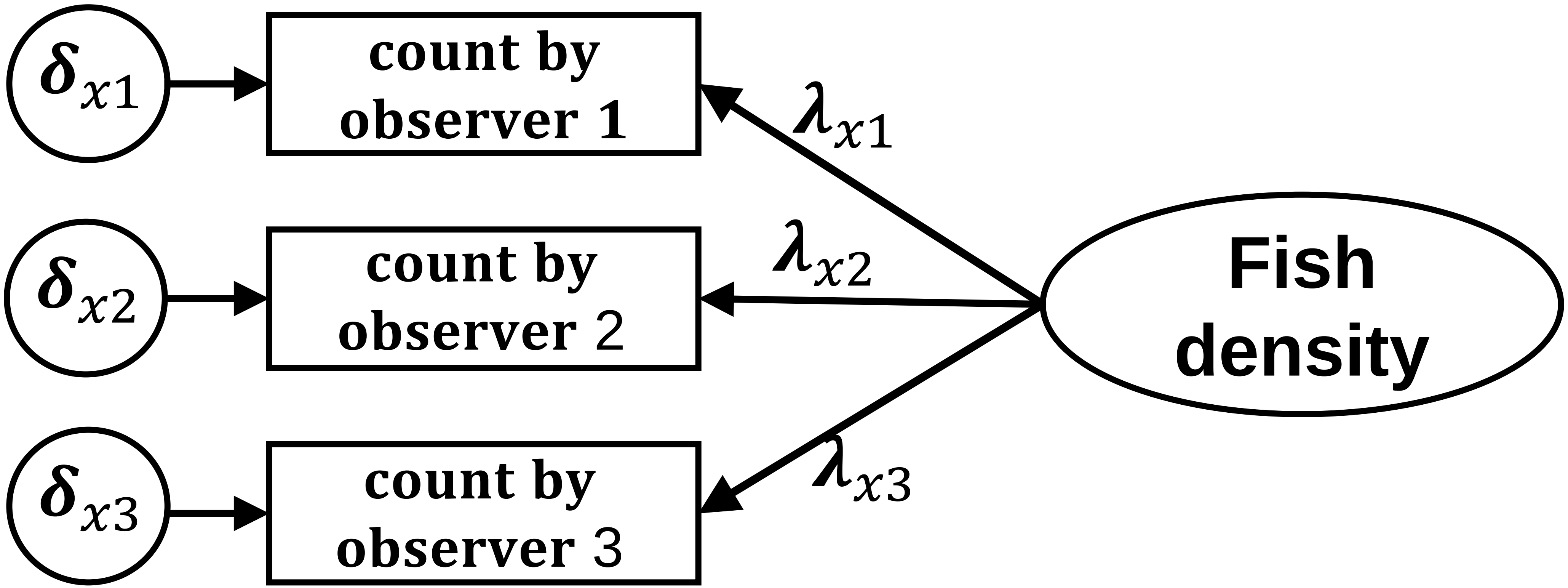
Multi-indicator Latent Variables

Repeated Measurements



Multi-indicator Latent Variables

Multi-sampling



Why use Latent Variables with Multiple Indicators?

- Allows estimating complex and multifaceted concepts
- Reduces random error in construct (latent variable)
- Better accuracy in measurement of relationships due to shared variation between observed indicators.

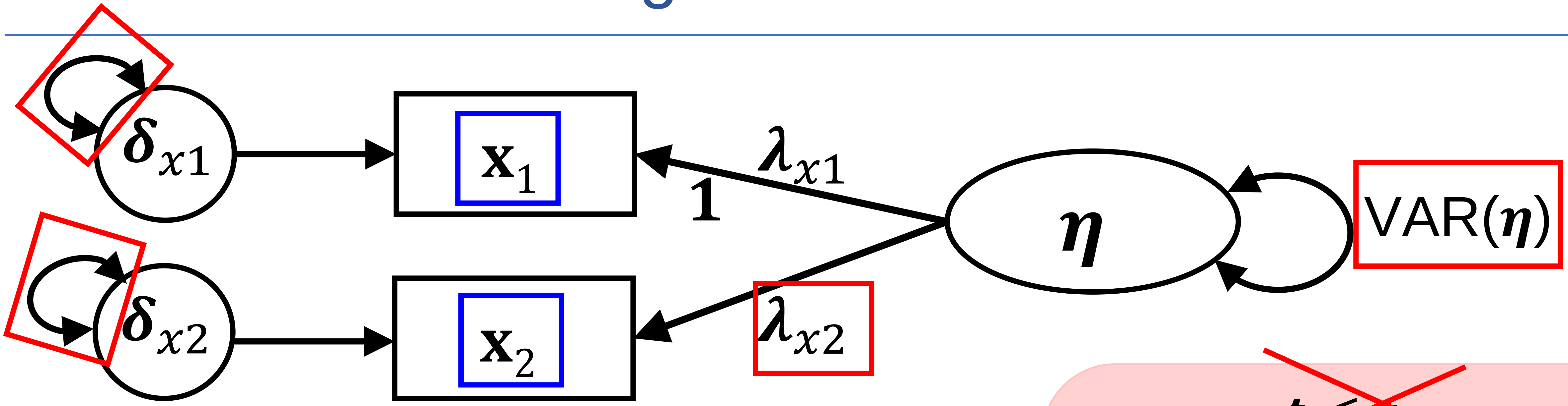
Outline

Latent Variables in SEM

- What are Latent Variables? Why to use them?
- Multi-indicator Latent Variables
- **Fitting Latent Variables**

  (Confirmatory Factor Analysis)

Fitting Latent Variables



Rules for LV models:

- Scaling of LV
- Non-negative DF

$$\mathbf{DF = -1}$$

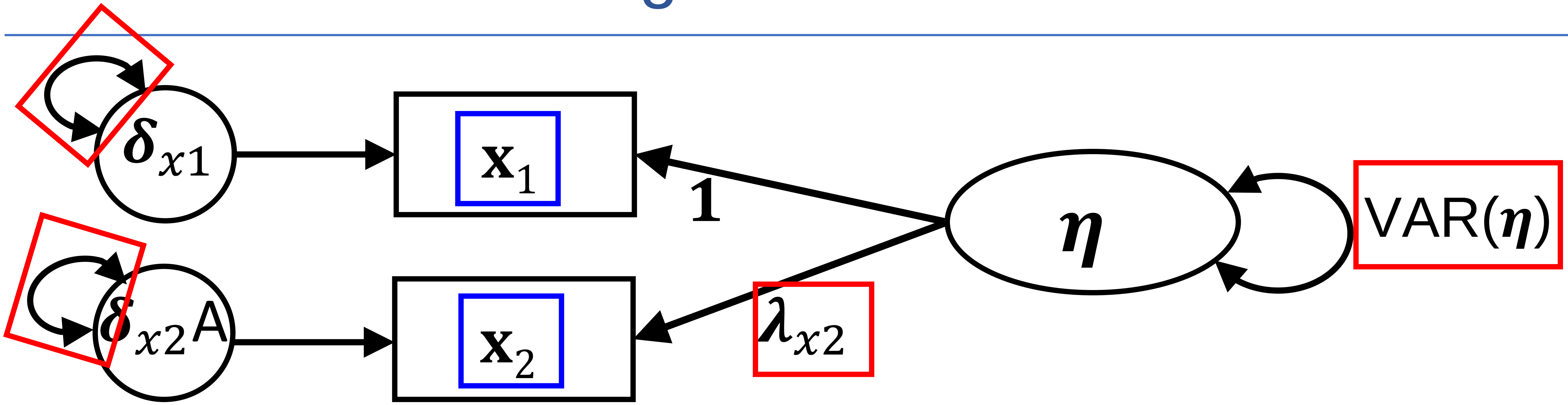
$$\cancel{t \leq t_{max}}$$

$$t_{max} = \frac{s(s+1)}{2} = 3$$

$s = 2$ knowns

$t = 4$ unknowns

Fitting Latent Variables



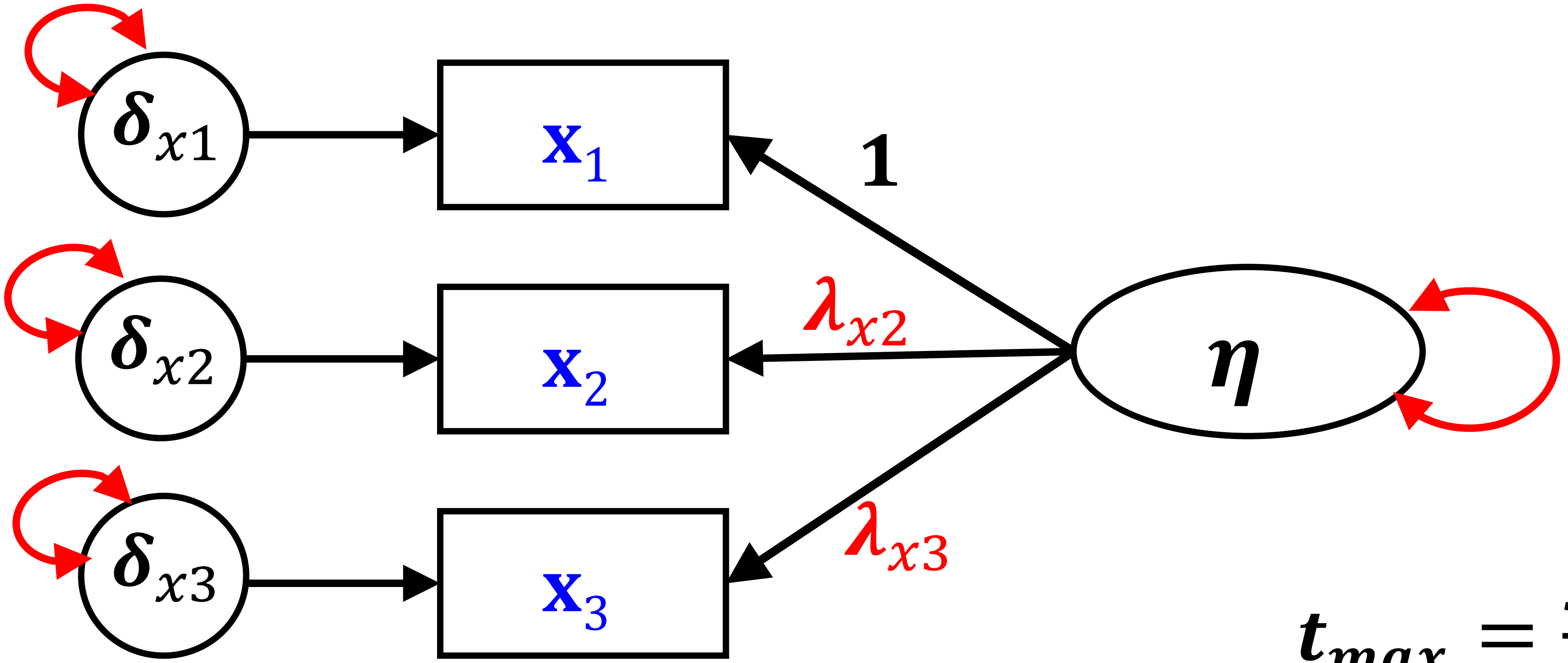
Rules for LV models:

- Scaling of LV
- Non-negative DF

We need at least:

- 3 indicators for a single LV
- 2 indicators per LV for models with multiple (correlated) LVs

Fitting Latent Variables



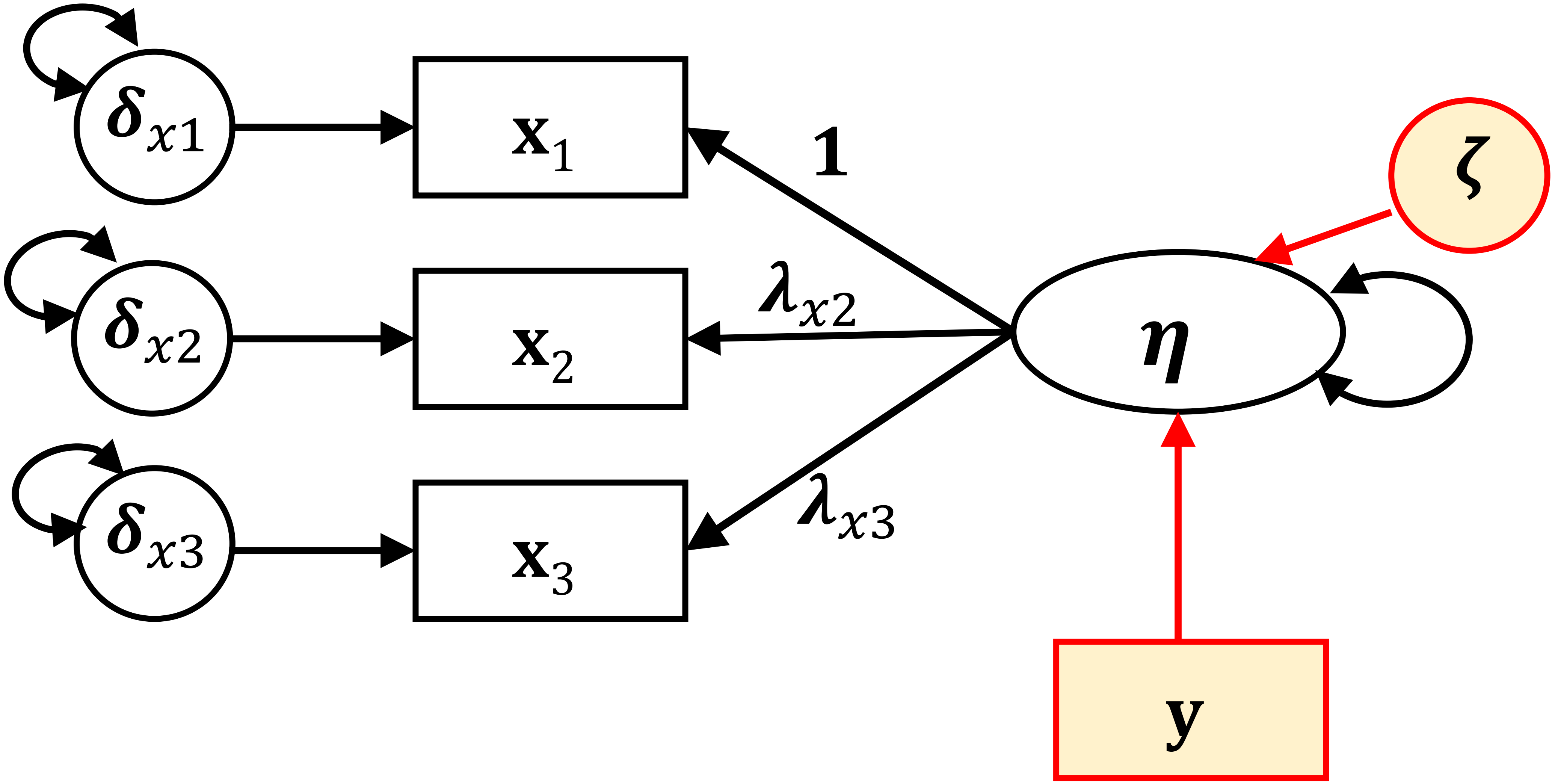
$$t_{max} = \frac{s(s+1)}{2} = 6$$

$s = 3$ knowns

$t = 6$ unknowns

$$\mathbf{DF} = t_{max} - t = \mathbf{0}$$

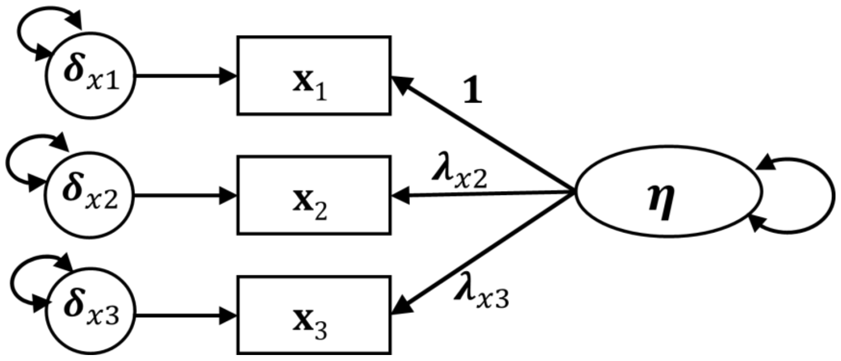
Fitting Latent Variables



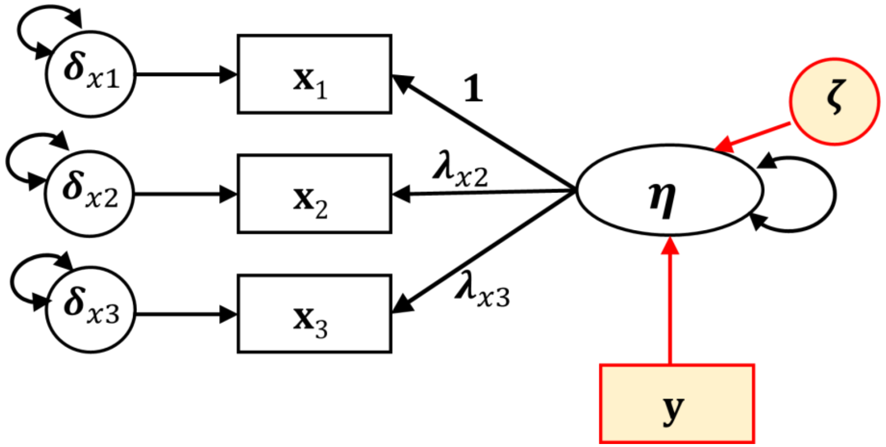
Fitting Latent Variables

1) Evaluate the latent relationships among variables (**Confirmatory Factor Analysis**).

- Do our indicators make a Good Latent Variable?



2) Use Latent Variables as a Response or a Predictor



Confirmatory Factor Analysis

Population-based ecological restoration

Aim: understand the performance of transplanted plants as a function of their dissimilarity to local conditions

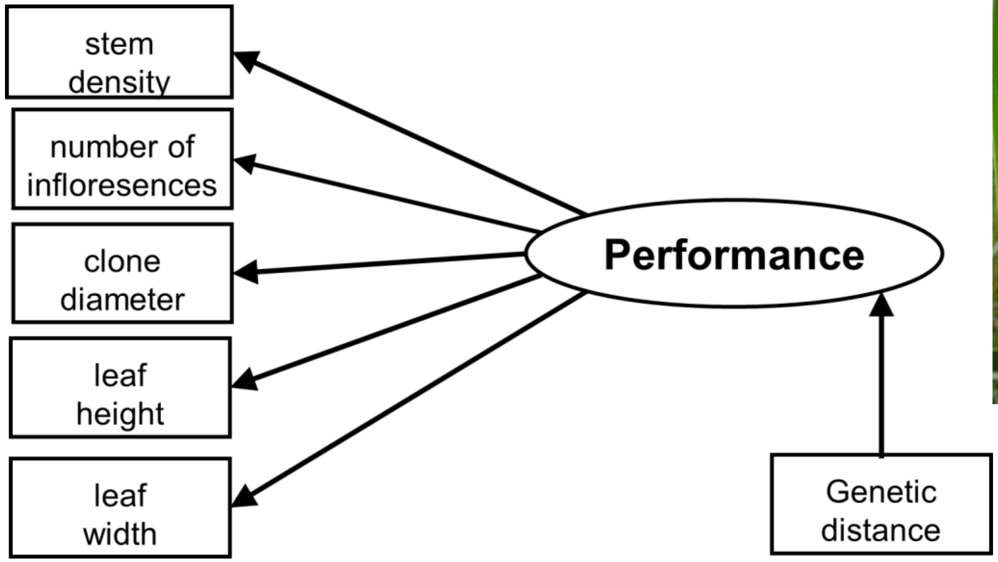

```
# Read and check the data

travis <- read.csv(" Travis_data.csv")
```

Travis, S. E., & Grace, J. B. (2010). Predicting performance for ecological restoration: a case study using Spartina alterniflora. Ecological Applications, 20(1), 192-204.

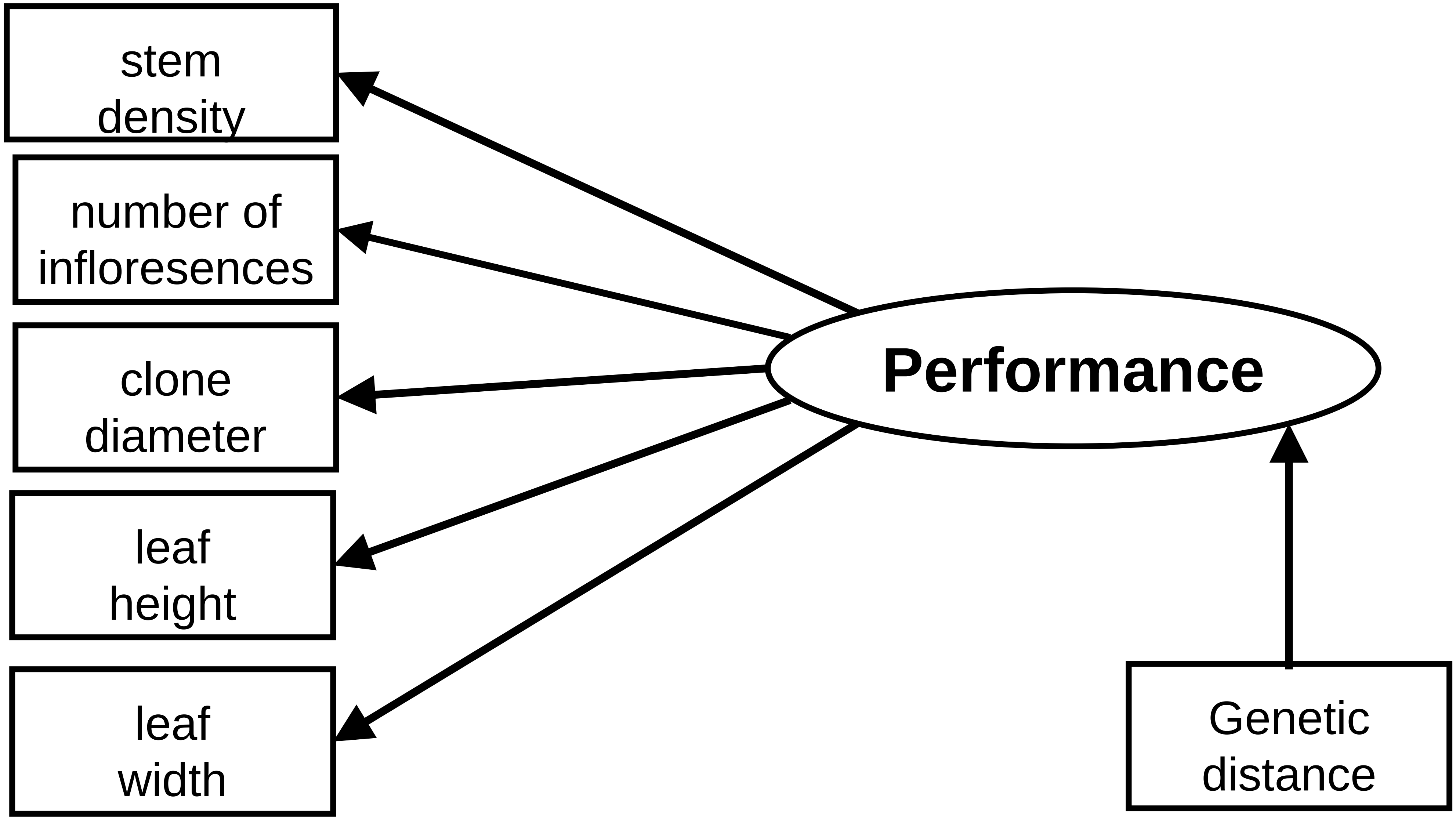
Confirmatory Factor Analysis

Exercise

- Collected individuals of the salt marsh plant *Spartina alterniflora* eight clones each from 23 populations
- Transplanted individuals and measured their performance relative to local populations.
- Performance was approximated with stem density, the number of infloresences, clone diameter, leaf height, and leaf width



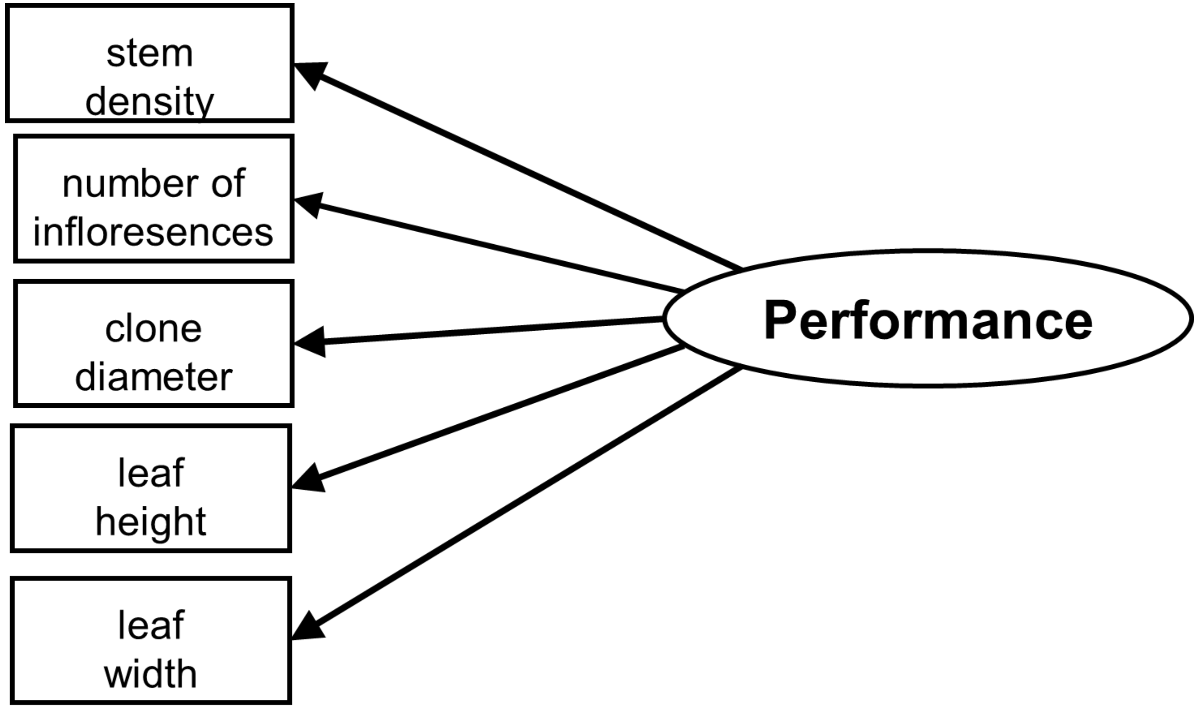
Confirmatory Factor Analysis



Confirmatory Factor Analysis

1) Evaluate the latent relationships among variables (**Confirmatory Factor Analysis**).

- Do our indicators make a Good Latent Variable?

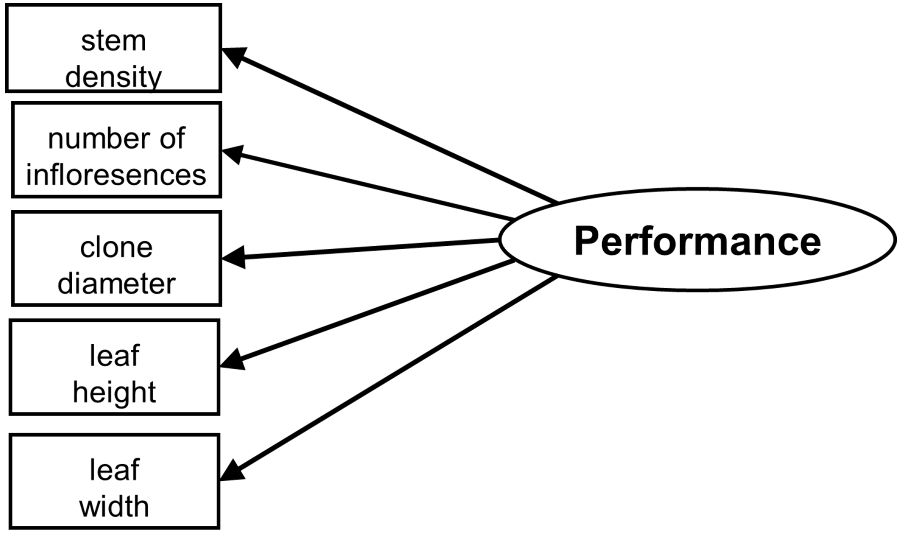


A first step is to analyze the “measurement model” using CFA.

Confirmatory Factor Analysis

Exercise

```
# Read and check the data
travis <- read_csv("Travis_data.csv")
str(travis)
# correlations
cor(travis[, 4:8])
```

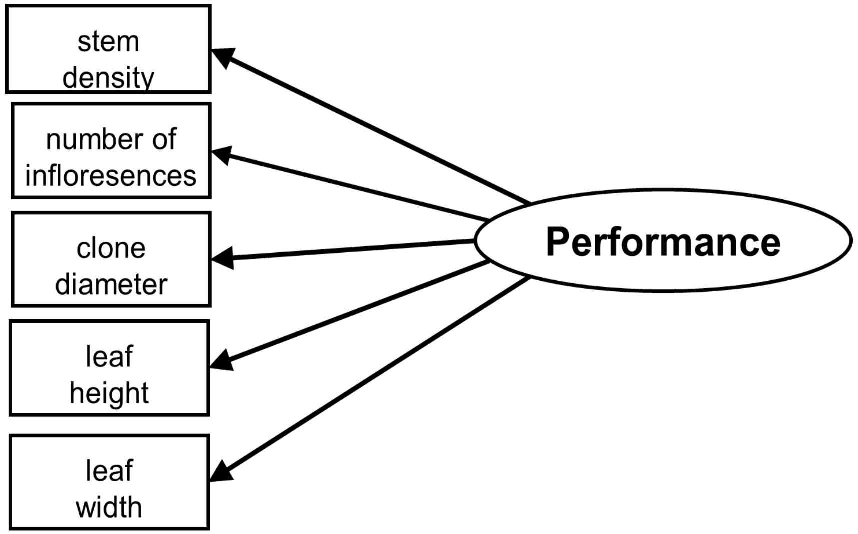



```
> round(cor(travis[, 4:8]),2)
          stems infls clonediam leafht leafwdth
stems      1.00  0.83      0.93   0.73     0.65
infls      0.83  1.00      0.81   0.69     0.60
clonediam  0.93  0.81      1.00   0.77     0.73
leafht     0.73  0.69      0.77   1.00     0.97
leafwdth   0.65  0.60      0.73   0.97     1.00
```

Confirmatory Factor Analysis

Exercise

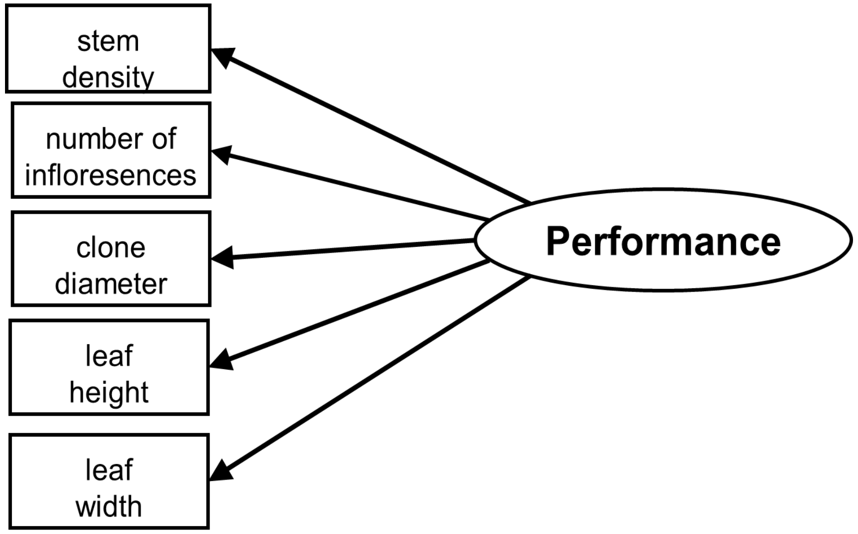
```
# specify the model
cfa_mod <- `
performance =~ stems + infls + clonediam + leafht + leafwdth
`
# fit the model
cfa_fit <- sem(cfa_mod, travis)
```




```
Warning message:
In lav_object_post_check(object) :
  lavaan WARNING: some estimated ov variances are negative
```

Confirmatory Factor Analysis

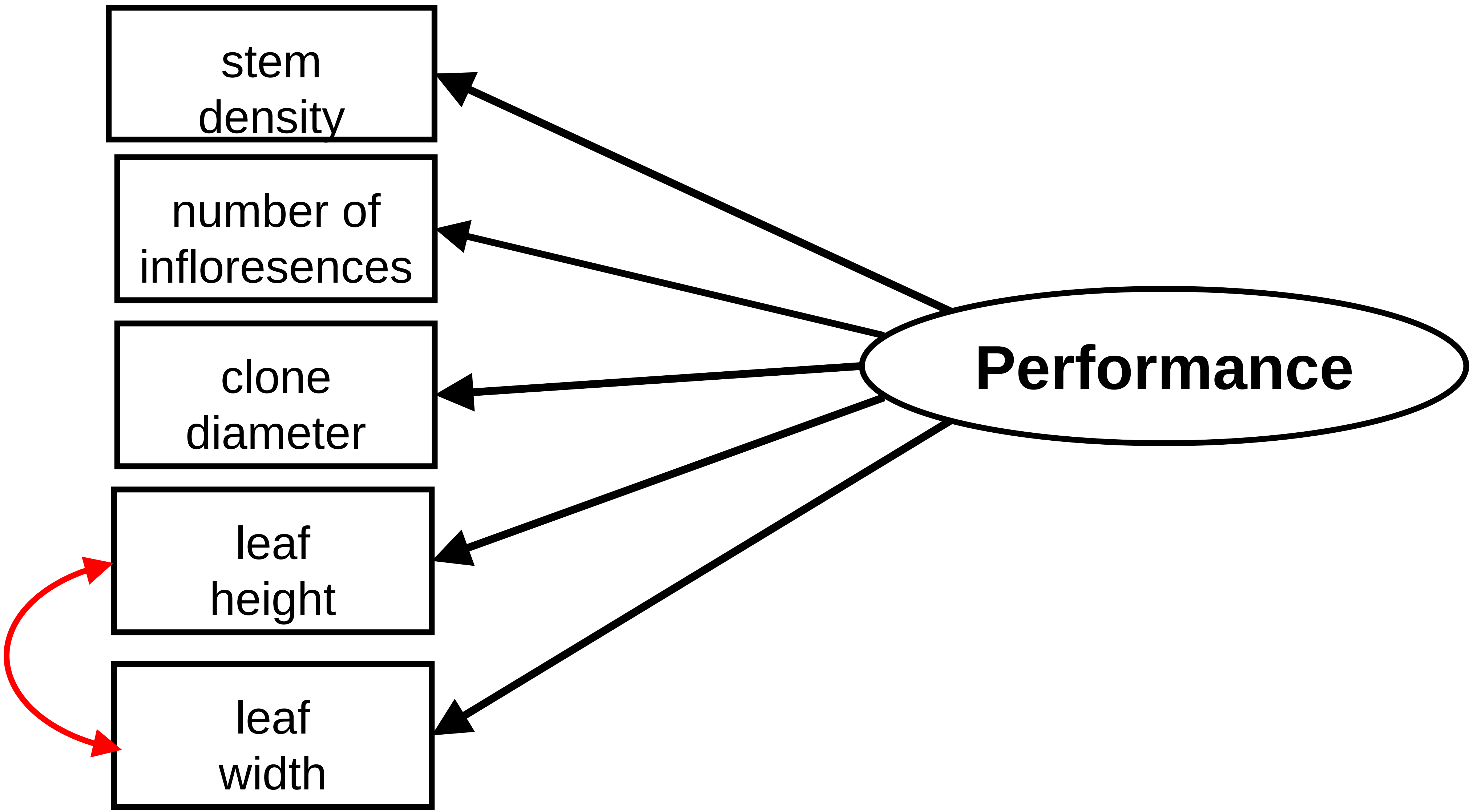

```
> summary(cfa_fit)

lavaan 0.6-9 ended normally after 82 iterations

  Estimator                                         ML
  Optimization method                           NLMINB
  Number of model parameters                        10

  Number of observations                            23
Model Test User Model:

  Test statistic                                51.106
  Degrees of freedom                                 5
  P-value (Chi-square)                           0.000
```



Confirmatory Factor Analysis

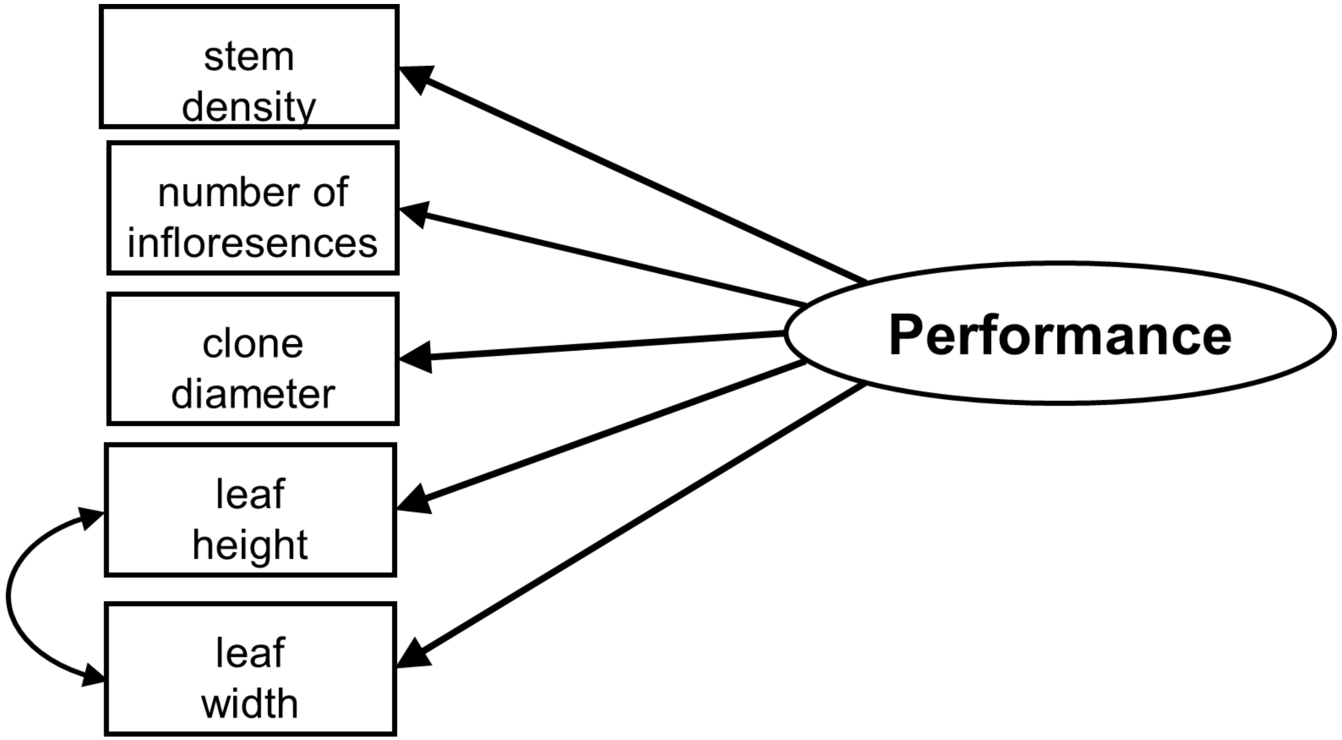
```
> modindices(cfa_fit)
         lhs op       rhs     mi     epc sepc.lv sepc.all sepc.nox
12     stems ~~     infls 10.470  11.784  11.784    0.677    0.677
13     stems ~~ clonediam 17.152 112.521 112.521    0.871    0.871
14     stems ~~    leafht  0.693  -7.889  -7.889   -0.517   -0.517
15     stems ~~  leafwdth  2.214  -1.836  -1.836   -0.346   -0.346
16     infls ~~ clonediam  8.773  11.092  11.092    0.621    0.621
17     infls ~~    leafht  0.062  -0.312  -0.312   -0.148   -0.148
18     infls ~~  leafwdth  2.906  -0.281  -0.281   -0.383   -0.383
19 clonediam ~~    leafht  4.028 -21.233 -21.233   -1.357   -1.357
20 clonediam ~~  leafwdth  0.037  -0.261  -0.261   -0.048   -0.048
21    leafht ~~  leafwdth 37.862  17.177  17.177   26.752   26.752
```

Confirmatory Factor Analysis



Confirmatory Factor Analysis


```
cfa_mod2 <- '
performance =~ stems + infls + clonediam + leafht + leafwdth
leafht ~~ leafwdth
'
cfa_fit2 <- sem(cfa_mod2, travis)
summary(cfa_fit2)
```



Confirmatory Factor Analysis


```
Estimator                                         ML
  Optimization method                         NLMINB
  Number of model parameters                      11

  Number of observations                          23

Model Test User Model:

  Test statistic                               7.410
  Degrees of freedom                               4
  P-value (Chi-square)                         0.116
```

Confirmatory Factor Analysis

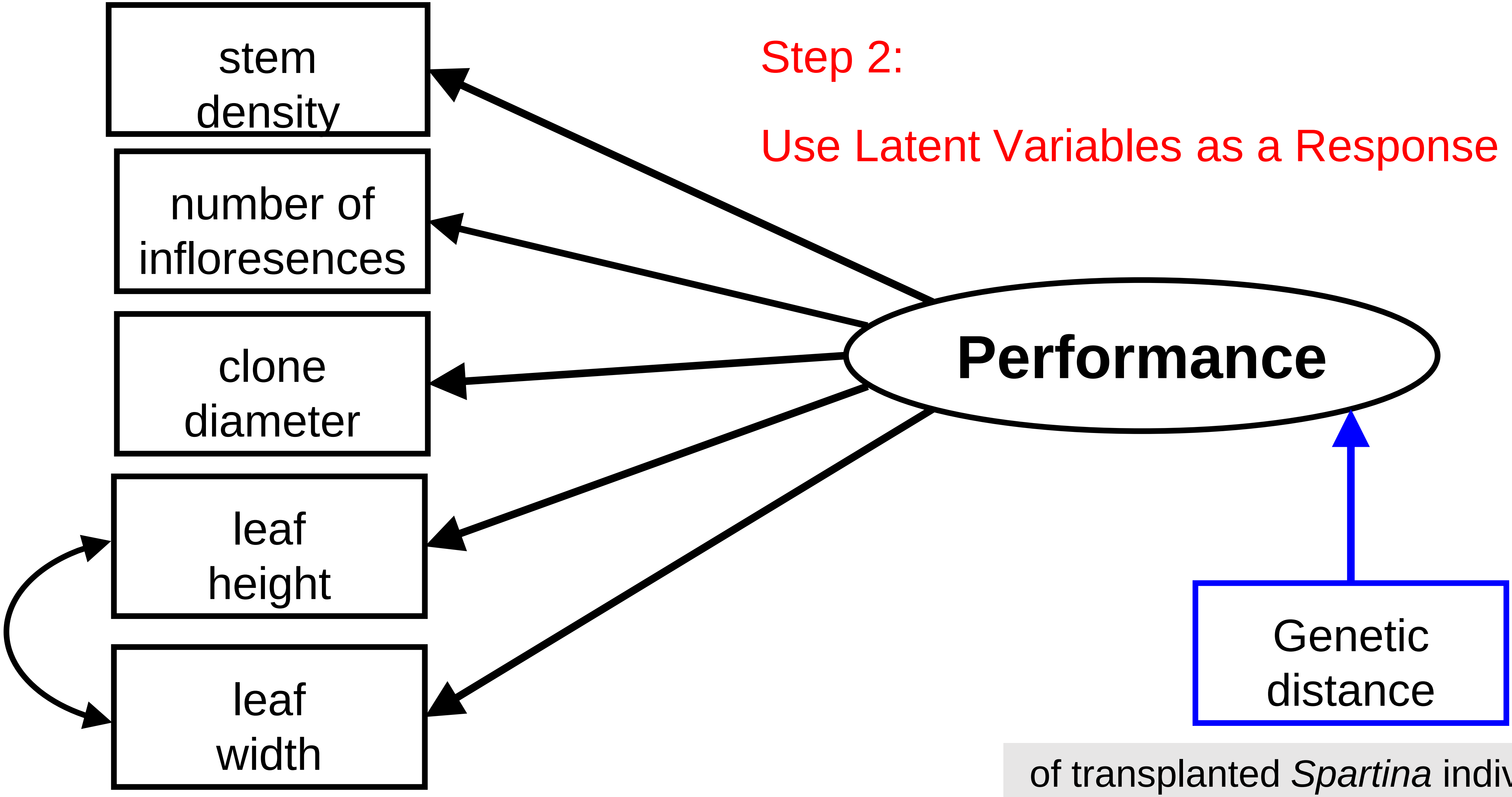

```
...

Latent Variables:
                   Estimate  Std.Err  z-value  P(>|z|)
  performance =~
    stems              1.000
    infls              0.117    0.016    7.173    0.000
    clonediam          1.086    0.096   11.319    0.000
    leafht             0.697    0.127    5.509    0.000
    leafwdth           0.082    0.018    4.529    0.000

Covariances:
                   Estimate  Std.Err  z-value  P(>|z|)
 .leafht ~~
   .leafwdth          10.831    3.432    3.156    0.002
```

CFA as a part of structural model

Exercise

Step 2:

Use Latent Variables as a Response or a Predictor



of transplanted *Spartina* individuals from the local population

CFA as a part of structural model

Exercise

```
SEM_latent_mod <- `
            # latent
performance =~ stems + infls + clonediam + leafht + leafwdth

            # structural paths
performance ~ geneticdist

             # correlated errors
leafht ~~ leafwdth
'

SEM_latent_fit <- sem(SEM_latent_mod , travis)

summary(SEM_latent_fit, standardize = T, rsq = T, fit.measures=T)
```

CFA as a part of structural model

Exercise

```
Estimator                                         ML
  Optimization method                         NLMINB
  Number of model parameters                      12

  Number of observations                          23

Model Test User Model:

  Test statistic                              12.237
  Degrees of freedom                               8
  P-value (Chi-square)                         0.141
```

CFA as a part of structural model

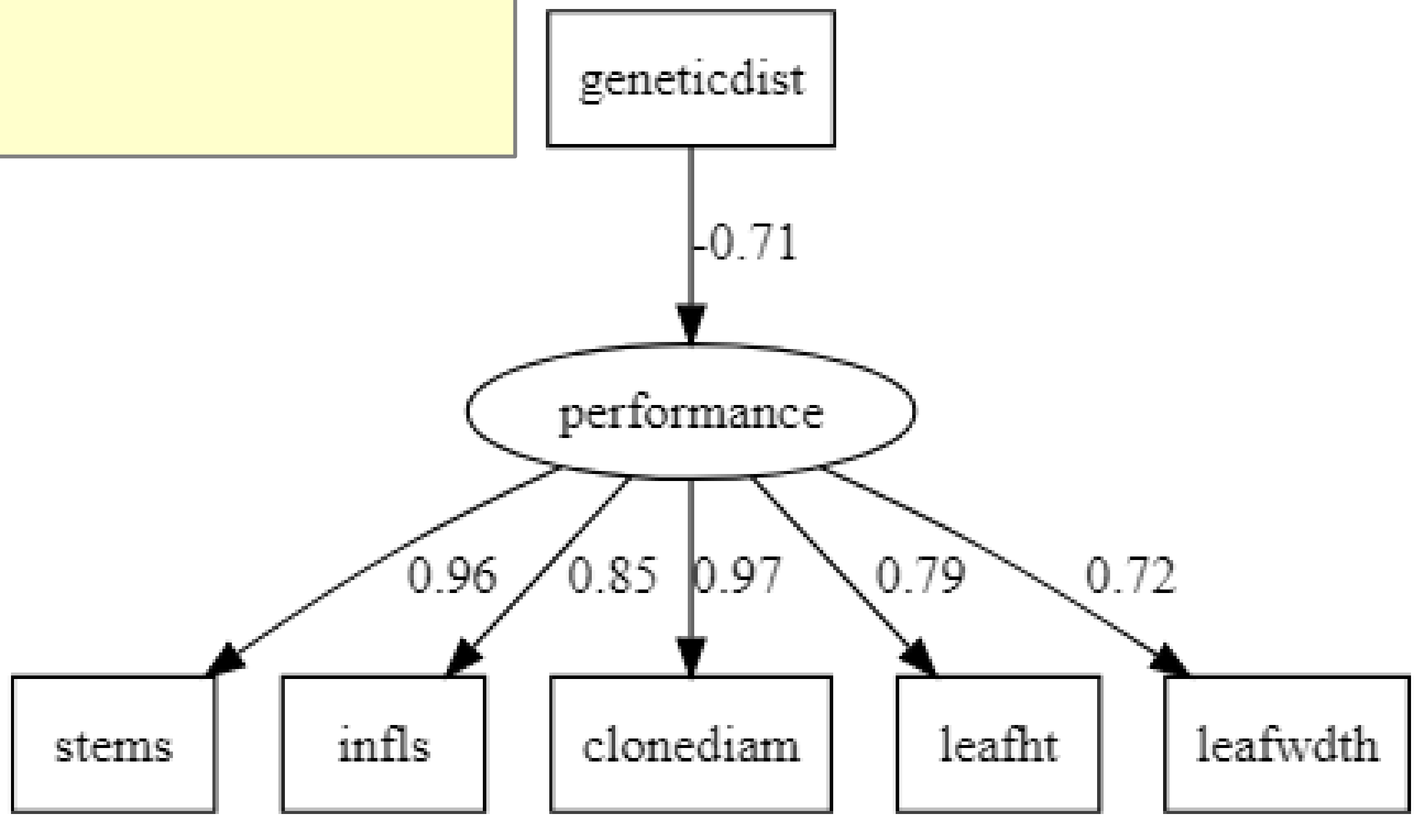
Exercise

```
Latent Variables:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
  performance =~
    stems             1.000                              15.555    0.962
    infls             0.117    0.017    6.929    0.000    1.822    0.853
    clonediam         1.106    0.096   11.508    0.000   17.199    0.969
    leafht            0.711    0.127    5.601    0.000   11.066    0.785
    leafwdth          0.084    0.018    4.650    0.000    1.308    0.718

Regressions:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
  performance ~
    geneticdist     -51.673   11.365   -4.547    0.000   -3.322   -0.708

Covariances:
                   Estimate  Std.Err  z-value  P(>|z|)   Std.lv  Std.all
 .leafht ~~
   .leafwdth         10.416    3.312    3.145    0.002   10.416    0.940
```

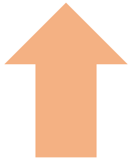
CFA as a part of structural model

Exercise

```
library(lavaanPlot)
lavaanPlot(model = SEM_latent_fit,
           coefs = TRUE, stand=TRUE,
           # graph_options = list(layout = "circo"),
           # stars = 'regress', # shows stars for regr coef
           digits = 2)
```



Day 7 Task 2



Human
Impact Intensity

Macroinvertebrate body size

Body size traits

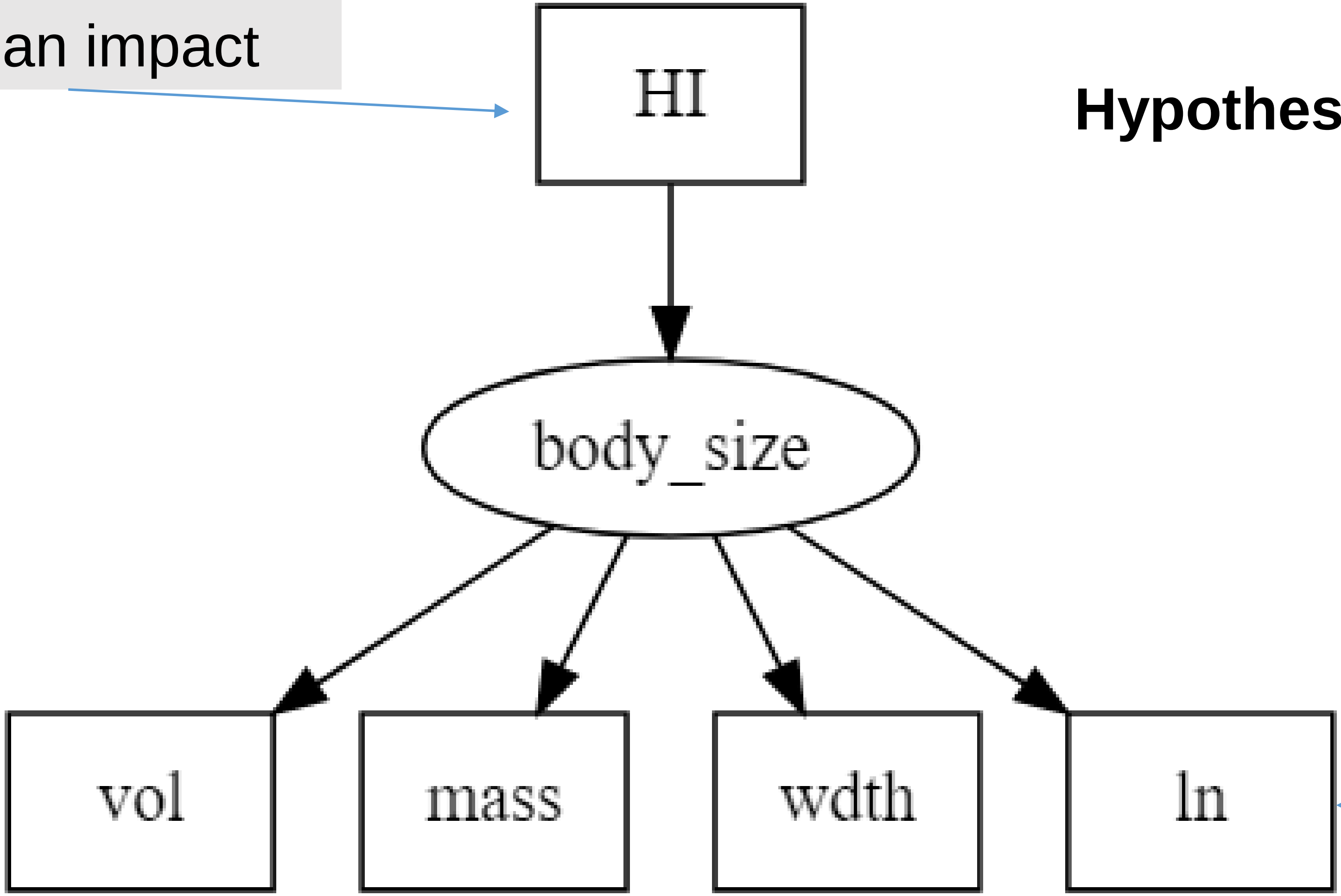
Body mass
Body volume
Body length
Body width

```
# Read and check the data

read.csv(" Bodysize_data.csv")
```

Day 5 Task 2

Human impact

Hypothesized model



Body volume
Body mass
Body width
Body length

Day 5 Task 2

1. Perform the confirmatory factor analysis for the latent variable “body size”
2. Use the results from step 1 and perform the SEM by adding human impact variable

3. Fill in Standardized Coeficients and R^2 for the model, add the fit indices

4. Think about how to interpret the results