Advanced Topics in SEM for Ecology & Evolutionary Biology

Jarrett E. K. Byrnes
UMass Boston

An Advanced Outline

1. Revisiting Sample Size
2. Revisiting Dsep in lavaan
3. Analysis of Nested Survey Data
4. Multilevel Generalized Piecewise SEM
5. Additional Spatial Techniques
6. Panel Models for Lagged Time Effects
7. Growth Curve Models & Time Series

Revisiting Sample Size

1. The further you are in a model from an exogenous data-generating, the weaker it's influence.

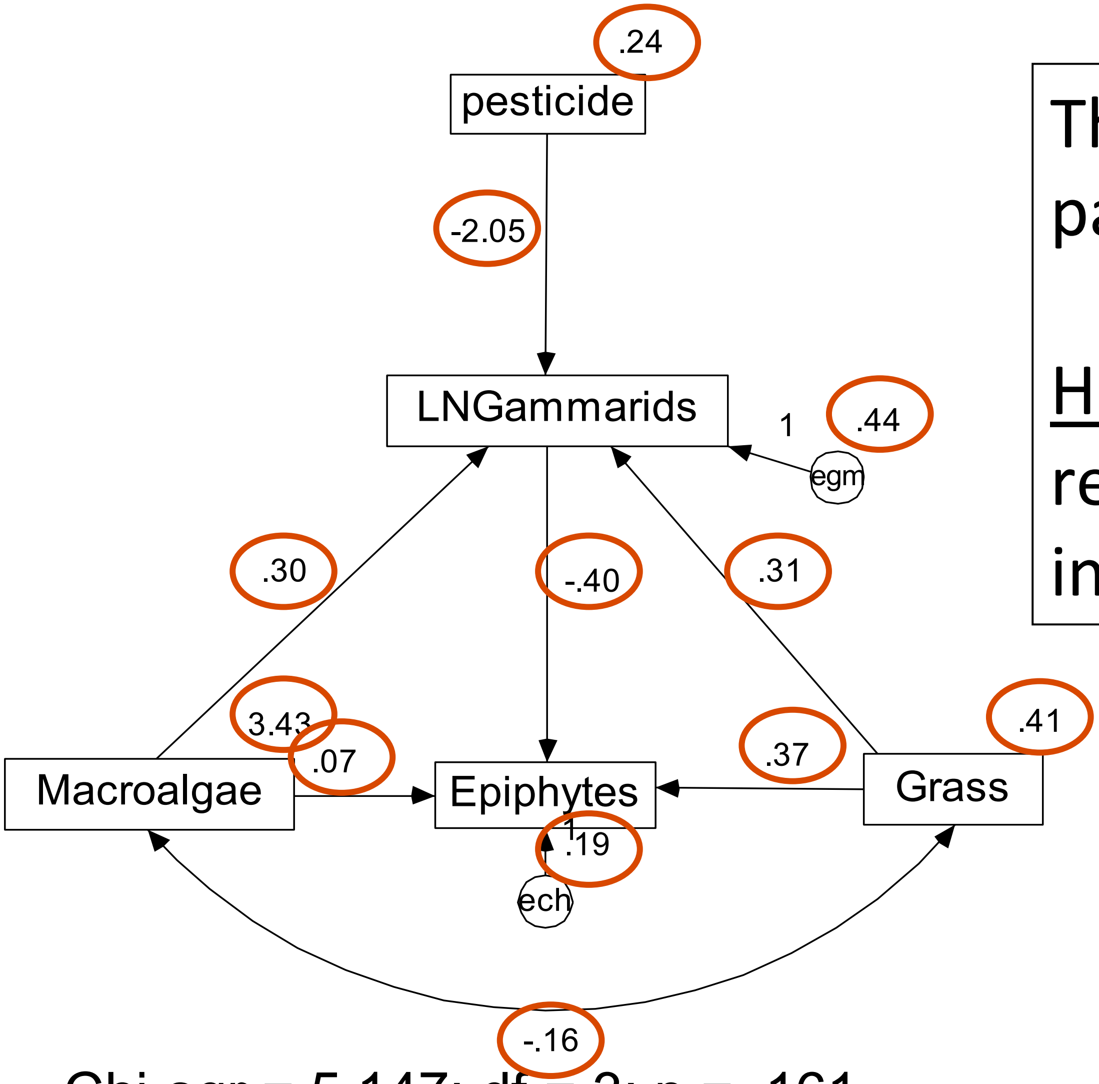
2. Our ability to detect the these tapering effect sizes is proportional to our <u>information</u> (especially sample size) and the <u>number of parameters being estimated</u>.

3. Our sample size sets an upper limit for the complexity of the model we can obtain.

4. <u>Rules of thumb for sample size</u> -- we hope to have at least 5 samples per estimated parameter and would prefer 20 samples per parameter.

5. Path coefficients add to our parameter list, not the variances

Number of Estimated Parameters

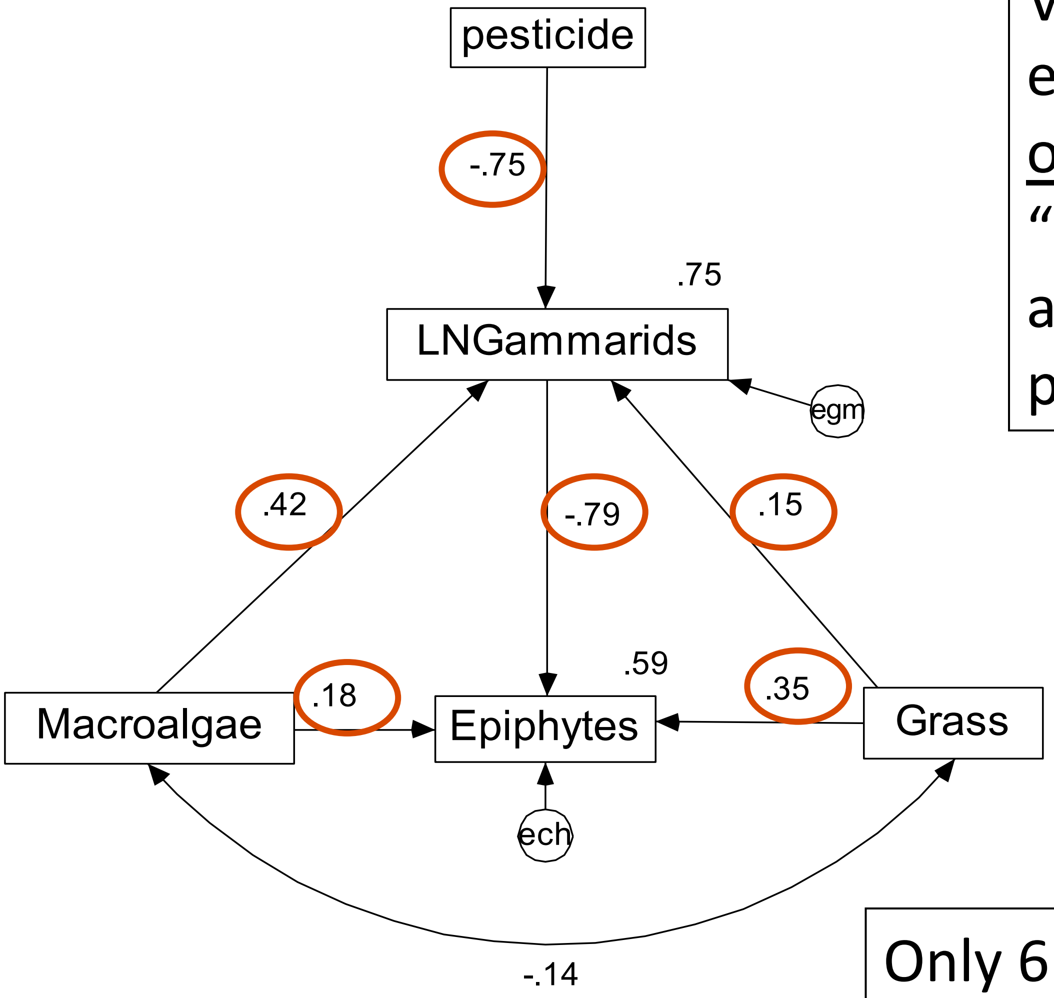


Chi-sqr = 5.147; df = 3; p = .161
AIC = 29.147; NPAR = 12

There are a total of 12 parameters shown.

However, only 6 of these require unique information…

Parameters Needing Unique Information



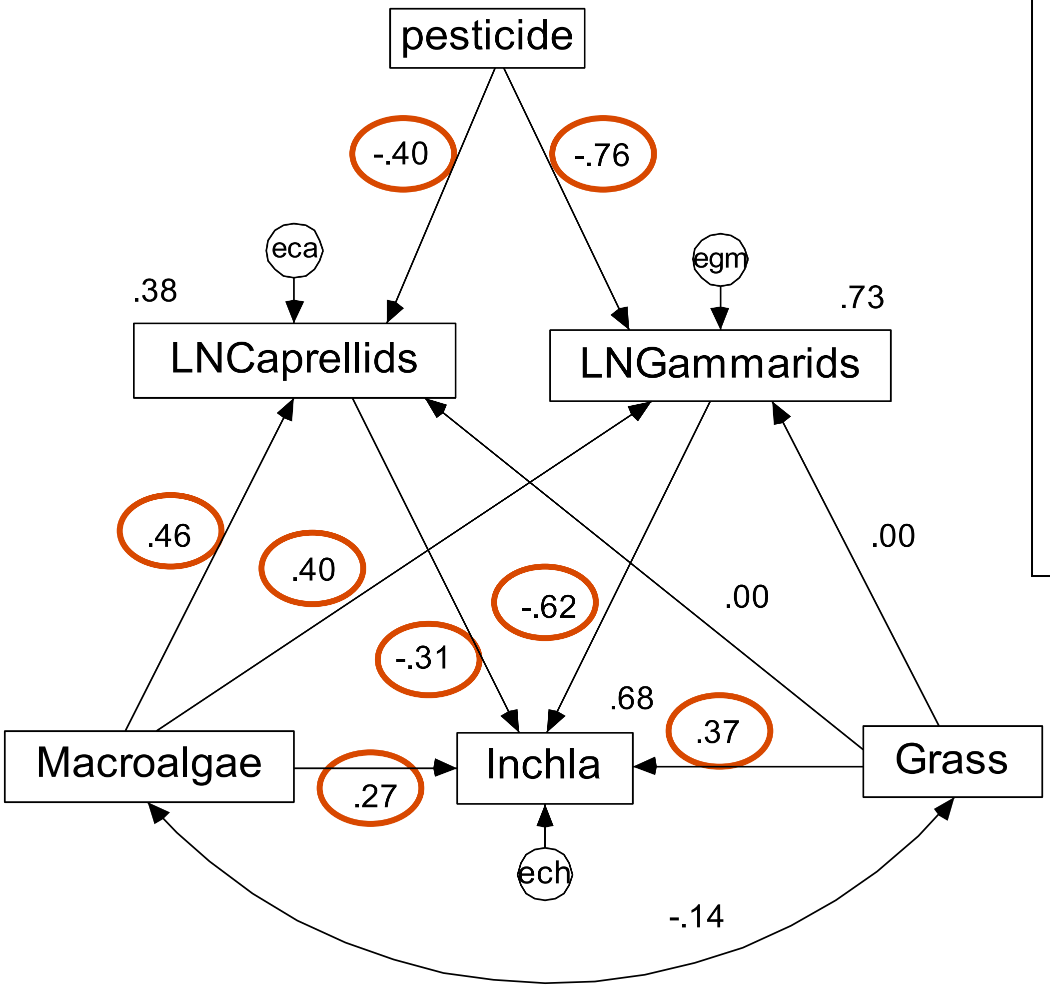
Chi-sqr = 5.147; df = 3; p = .161
AIC = 29.147; NPAR = 12

Variances & covariance of exogenous variables can be obtained from the data. For “pesticide”, “Macroalgae”, and “Grass", this yields 4 parameters.

Error variances (and R-sqrs) for endogenous variables are calculated from other parameters. This is 2 parameters.

Only 6 parameters require unique information.
Samples/parameters = 40/6 = 6.7.

Removing Unimportant Paths



chi-sqr = 9.545; df = 6; p = .145;
AIC = 39.545; NPAR = 15

For our more complex model, we would want to set non-contributing paths to zero to minimize estimated parameters.

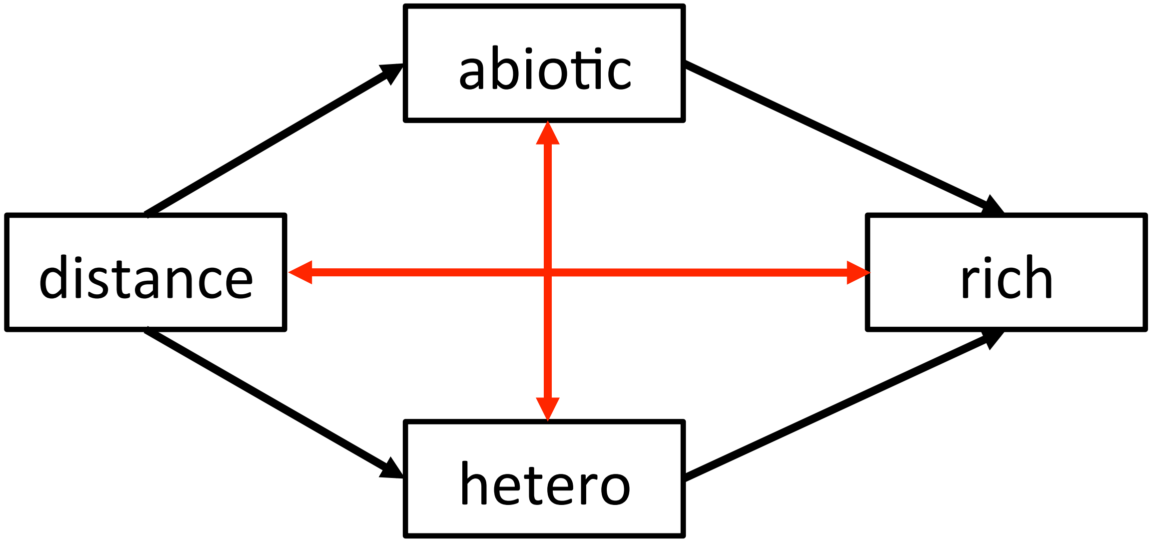
Here estimated parameters = 8, samples/parameters = 5.

If we can combine Caprellids and Gamarids, we could reduce parameters further.

An Advanced Outline

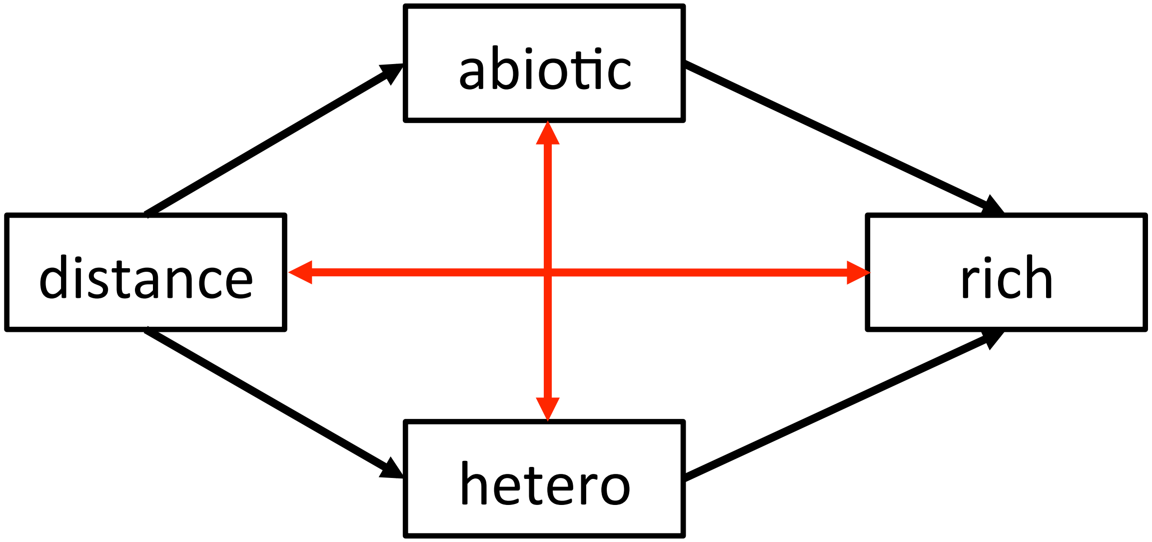
1. Revisiting Sample Size
2. Revisiting Dsep in lavaan
3. Analysis of Nested Survey Data
4. Multilevel Generalized Piecewise SEM
5. Additional Spatial Techniques
6. Panel Models for Lagged Time Effects
7. Growth Curve Models & Time Series

D-Separation & the χ^2



1. χ^2 gives you information regarding the discrepancy between your observed and predicted covariance matrix.
2. The test of D-Separation gives you information regarding whether you have missed key associations between variables.
3. We can test for D-Separation in recursive models without correlated error simply

D-Separation in lavaan

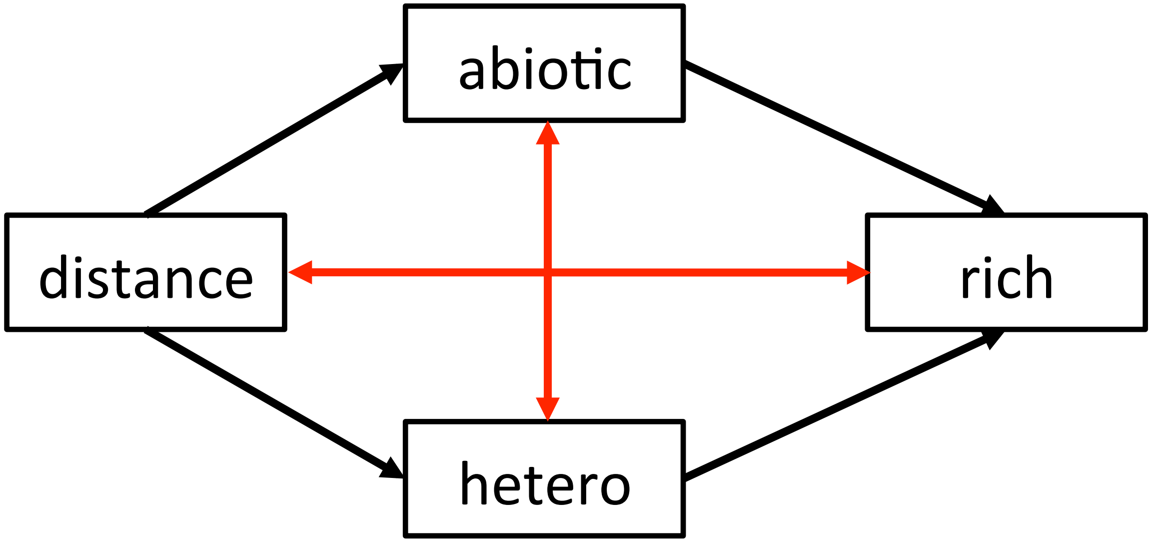


Two options

1) Feed model to DAG in ggm

```
#Full Mediation
distModel2 <- 'rich ~ abiotic  + hetero
              hetero ~ distance
              abiotic ~ distance'
```

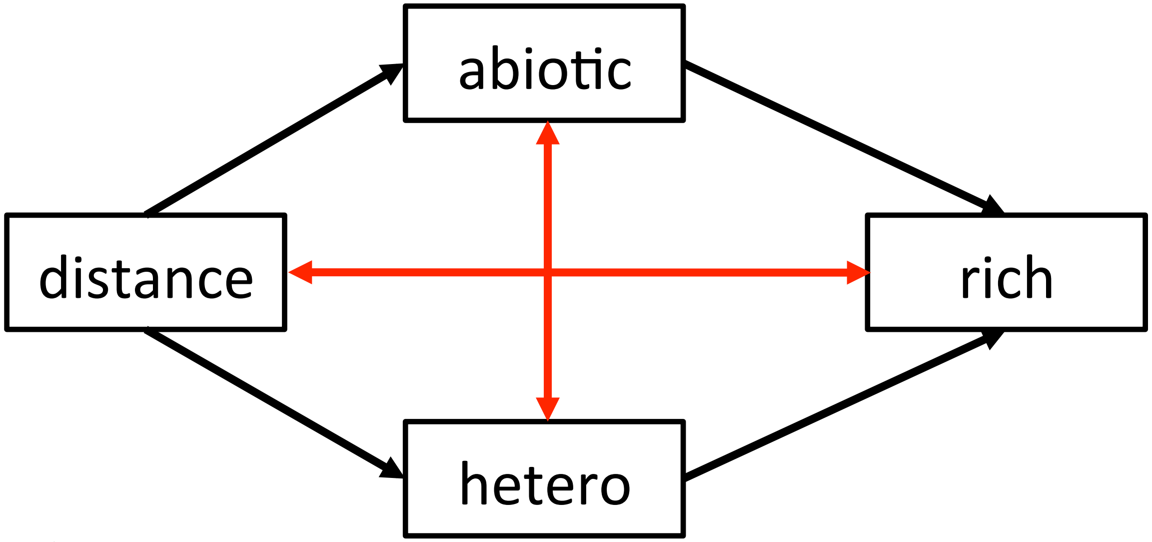
D-Separation in lavaan



2) Use script (and this will be in future lavaan versions)

```
> source("./dsepTest.R")
> dsepTest(distFit2)
$ctest
[1] 21.86173

$df
[1] 4

$pvalue
[1] 0.0002135289
```

D-Separation in lavaan


abiotic

distance

rich

hetero

```
> dsepTest(distFit2, showall=T)
$ctest
[1] 21.86173

$df
[1] 4

$pvalue
[1] 0.0002135289

$dsep
                   Pair   Conditioning         P.t.
distance  distance,rich hetero,abiotic 9.564005e-05
abiotic  abiotic,hetero       distance 1.871306e-01
```

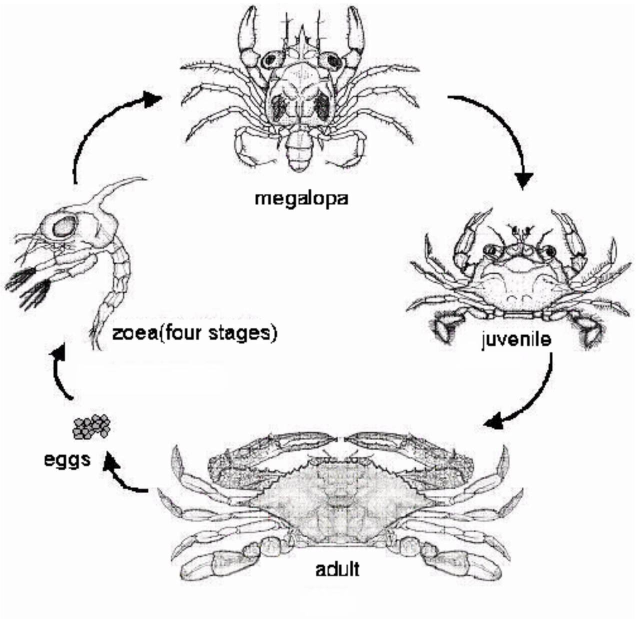
An Advanced Outline

1. Revisiting Sample Size
2. Revisiting Dsep in lavaan
3. Analysis of Nested Survey Data
4. Multilevel Generalized Piecewise SEM
5. Additional Spatial Techniques
6. Panel Models for Lagged Time Effects
7. Growth Curve Models & Time Series

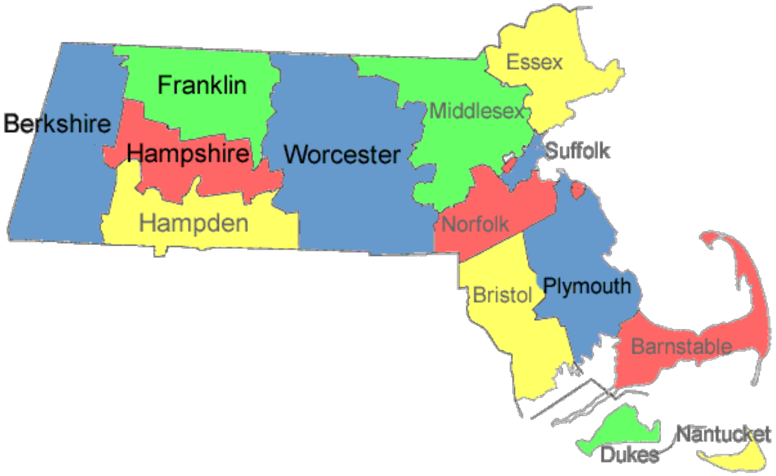
How might structure in surveys affect SEM results?

- Parameter estimates & variation biased
  - Depends on structure of sampling
- Parameter estimates & variation inconsistent
  - Dependent on groups with largest range of variation and/or sample size
- Covariance matrix may also be biased
- Fit measures likely incorrect

Clusters and Strata in Surveys



- Strata: levels that divide a population into different classes
  - E.g. age classes of an organism, grades in high school, income level



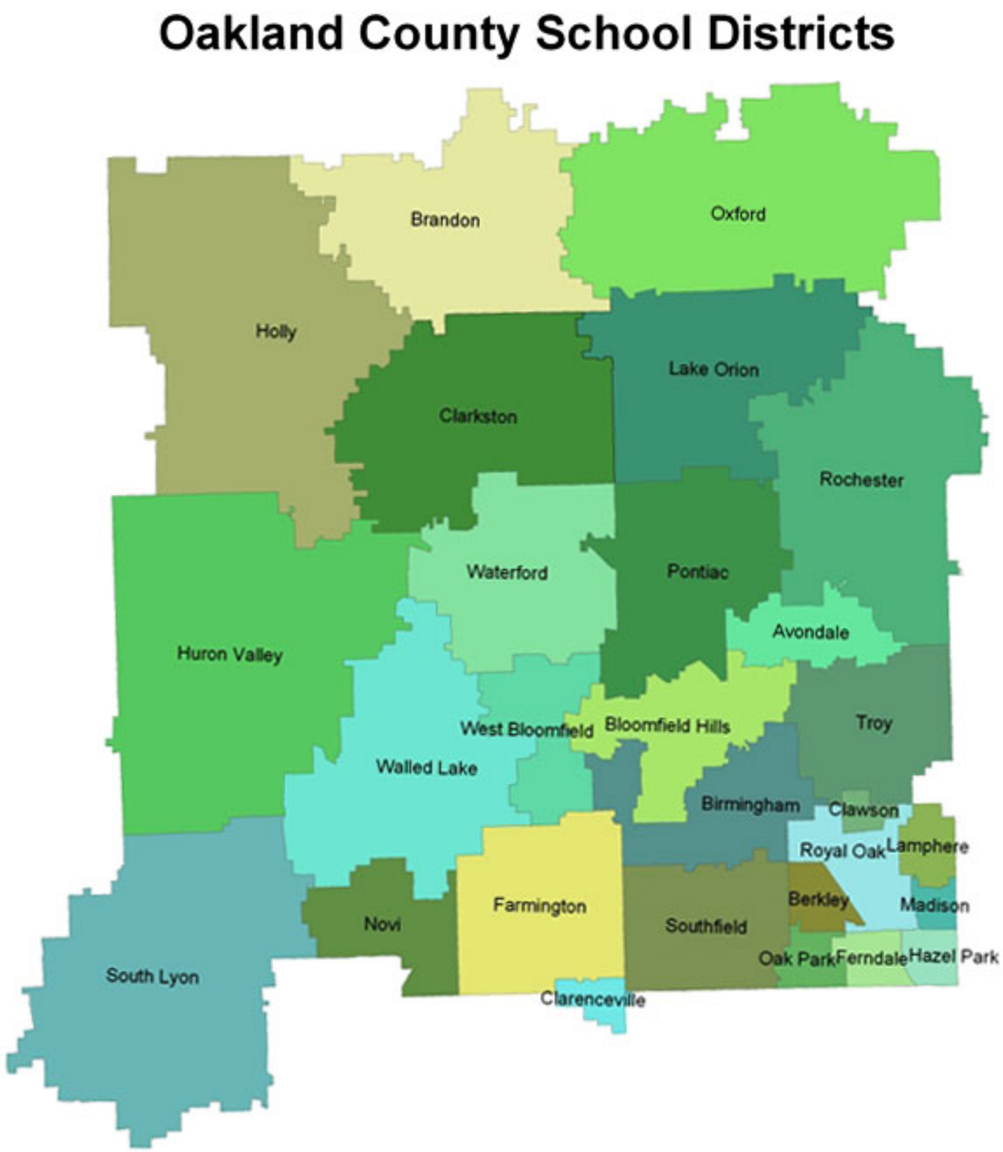
- Clusters
  - Blocks sampled at random
  - Not all blocks may be sampled
  - Can be levels of nesting

Example Survey

Sample of Algal Composition

- Islands randomly sampled – CLUSTER
- West/East side of island - STRATA

Example Survey



<u>Sample of High School Test Scores</u>

- District = CLUSTER
- No strata
- UNLESS: grade = STRATA

How do we Correct our Estimation?

1. Create a weight matrix based on the survey structure and then either

2. Correct error and $\chi 2$ of model fit using a corrected estimator or robust correction (e.g., S-B)

Or

3. Use weight matrix with WLS estimation

The lavaan.survey package

- Blends `lavaan` fits & survey structure

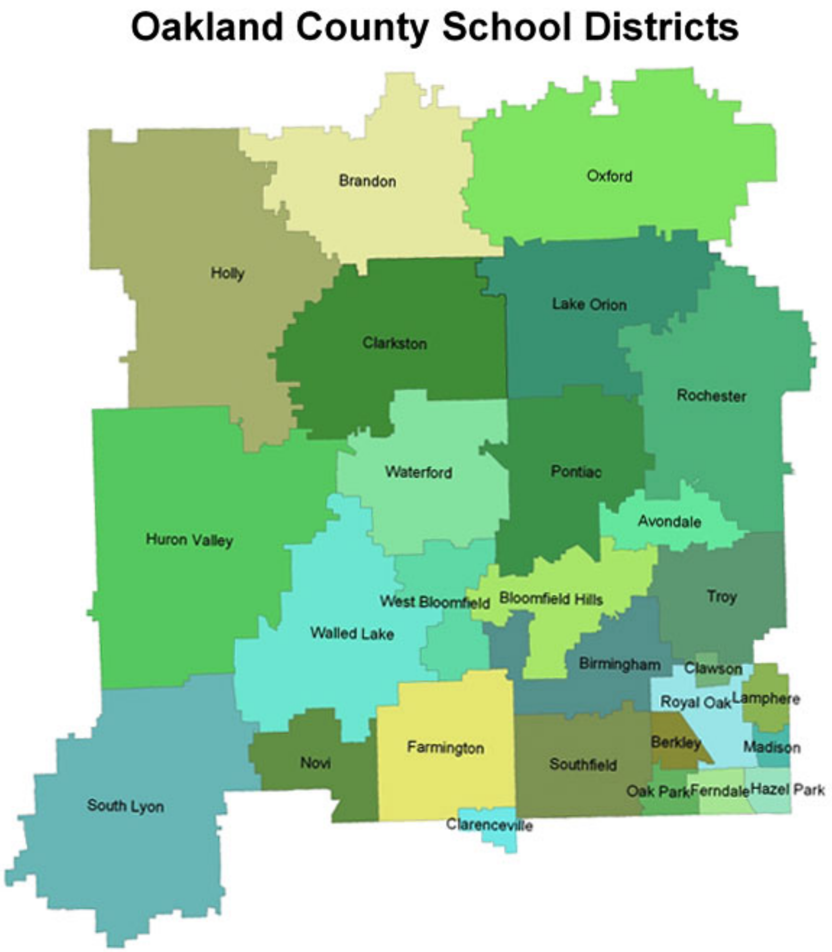
- Survey structure described using `survey` package

- Produces models with corrected errors and fit statistics

`Survey` Package

- Designed to provide robust estimation & SE under clustered, stratified surveys
- http://r-survey.r-forge.r-project.org/survey/index.html
- Complex Surveys: a guide to analysis using R by Thomas Lumley. 2010. Wiley & Sons. http://bit.ly/surveyByLumley
- Lumley, T. 2004. Analysis of Complex Survey Samples. J. Statistical Software **9**.

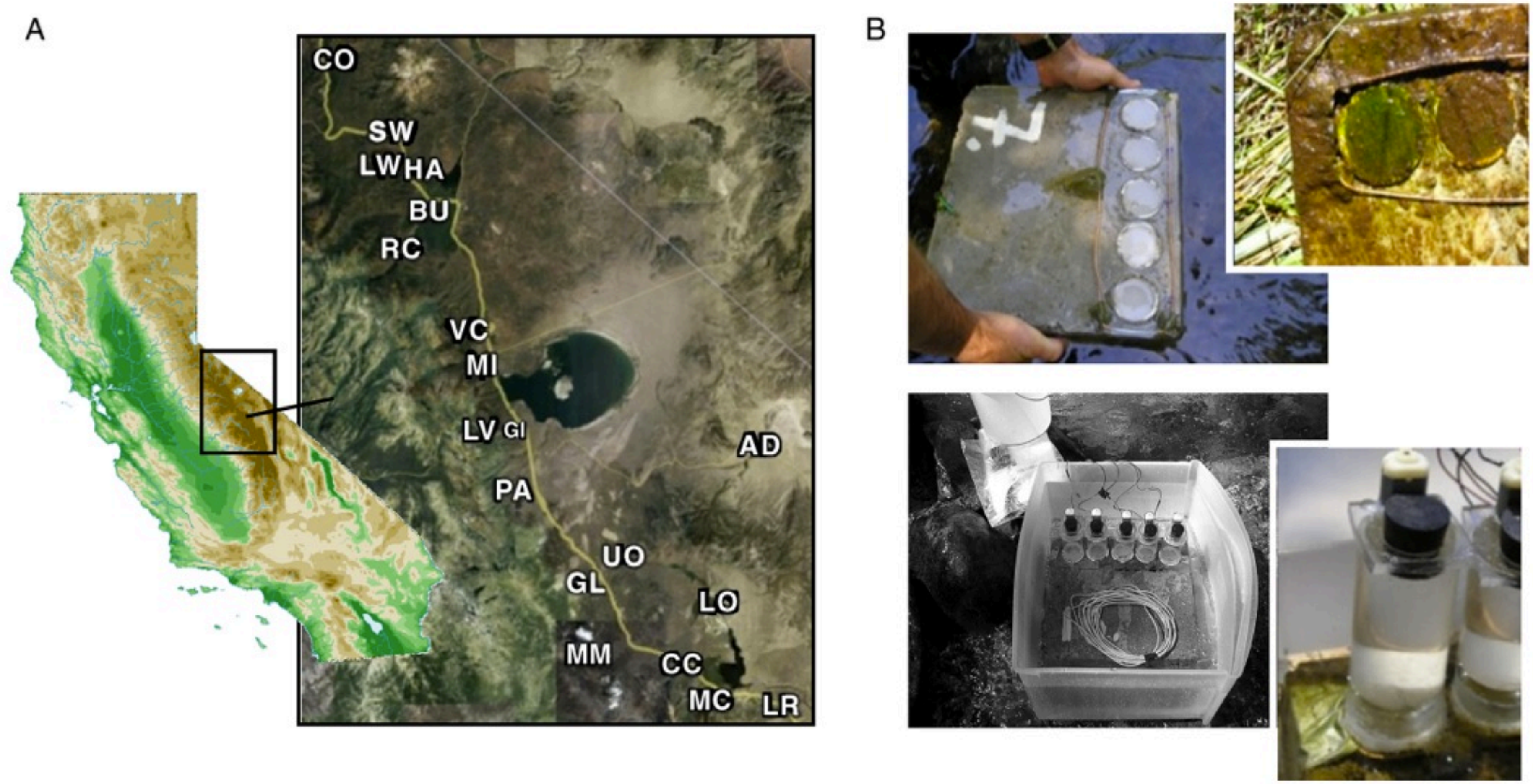
`svydesign` to Describe Survey



```
svydesign(
  ids =~ district,
  strata=grade,
  probs =~1, data=d)
```

probs means schools from each cluster have an equal probability of being sampled in the survey

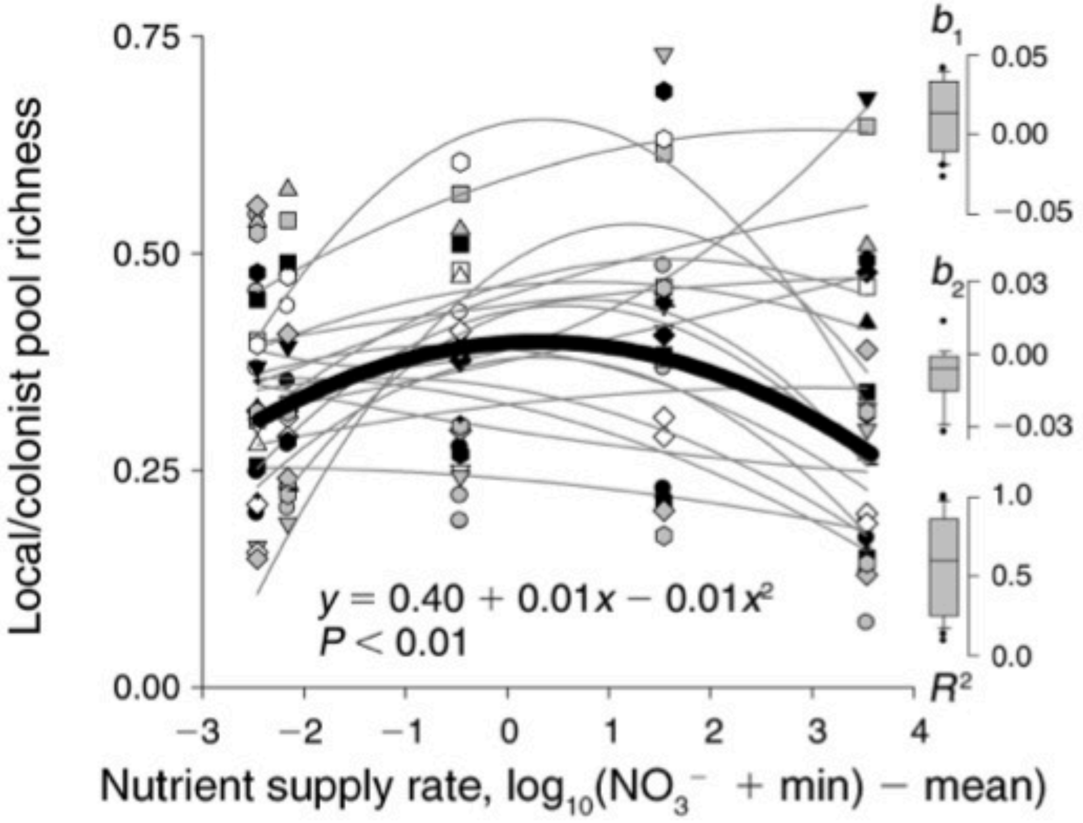
A Multi-Stream Experiment



Also wells grouped in blocks within a stream

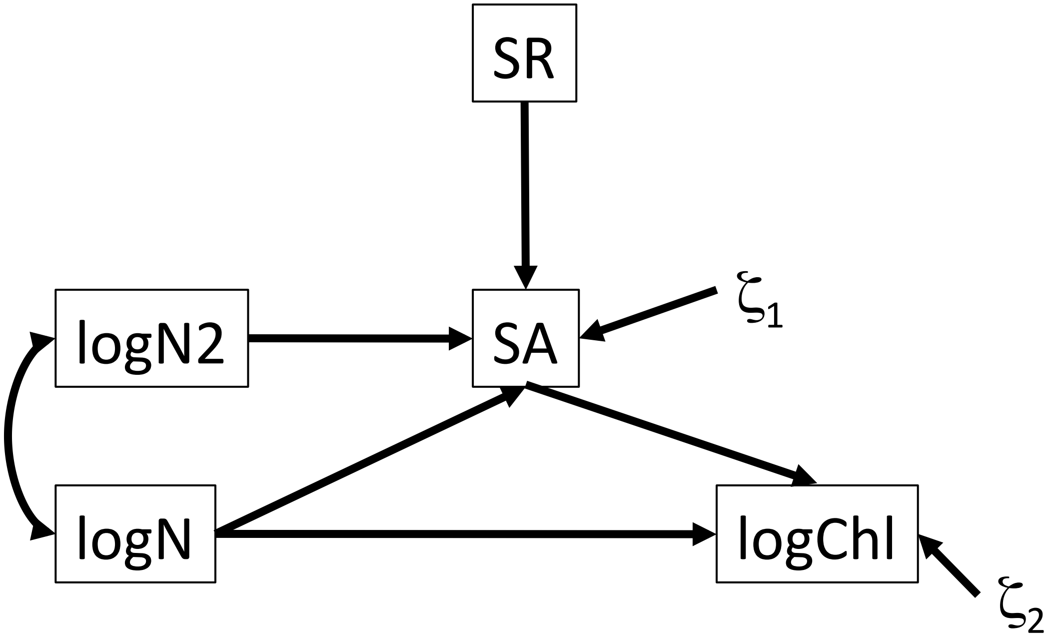
Cardinale et al 2009

Nonlinear Relationship Between Nutrient Addition and Richness



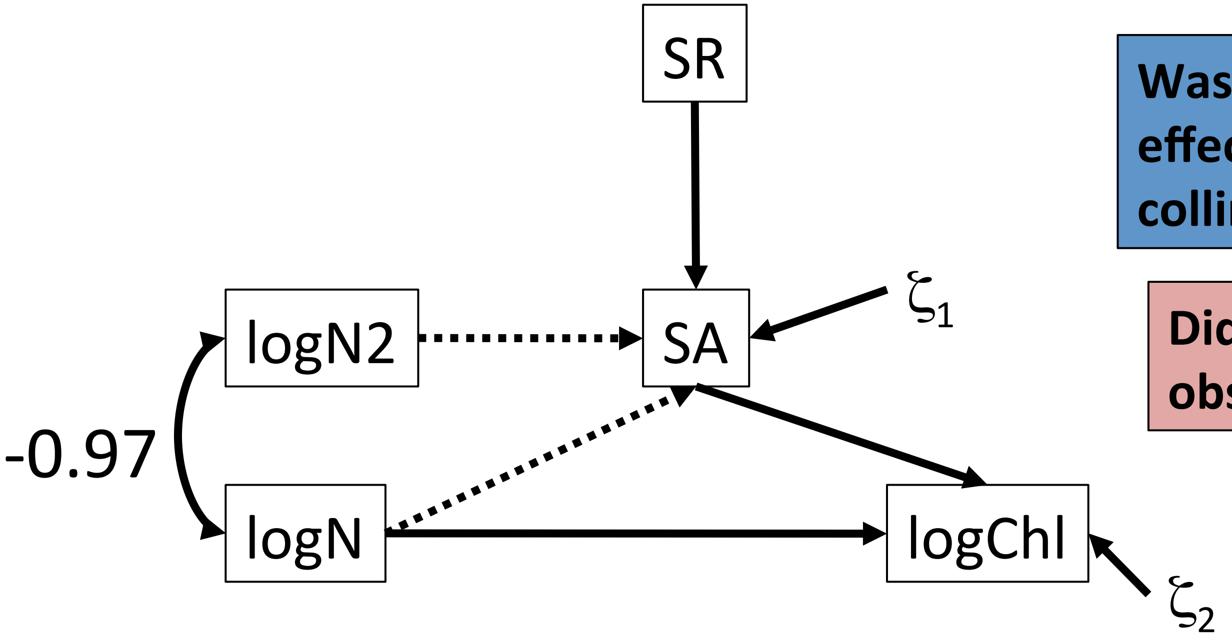
Cardinale et al 2009

Note that Treatment's Don't Covary with Regional Richness



```
cardModel<-'
  SA ~ logN + logNcen2 + SR
  logChl ~ SA + logN
  logN ~~ logNcen2’

cardFit <- sem(cardModel, data=cards, fixed.x=F,
  estimator=“MLM”)
```

But…no Nutrient Effect?



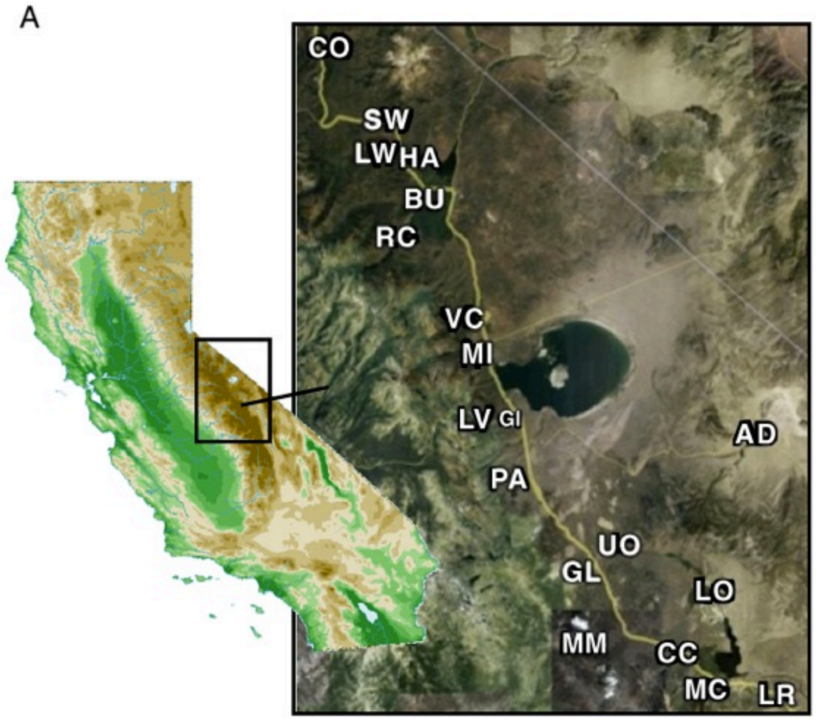
Was the lack of an effect all just due to collinearity?

Did clustering obscure the effect?

```
                    Estimate  Std.err  Z-value  P(>|z|)   Std.lv  Std.all
Regressions:
  SA ~
    logN              -2.996    2.960   -1.012    0.311   -2.996   -0.504
    logN2             -0.475    0.456   -1.041    0.298   -0.475   -0.506
    SR                 0.384    0.035   10.859    0.000    0.384    0.688
  logChl ~
    SA                 0.020    0.005    3.768    0.000    0.020    0.393
    logN               0.097    0.023    4.194    0.000    0.097    0.321
```

Cardinale et al 2009

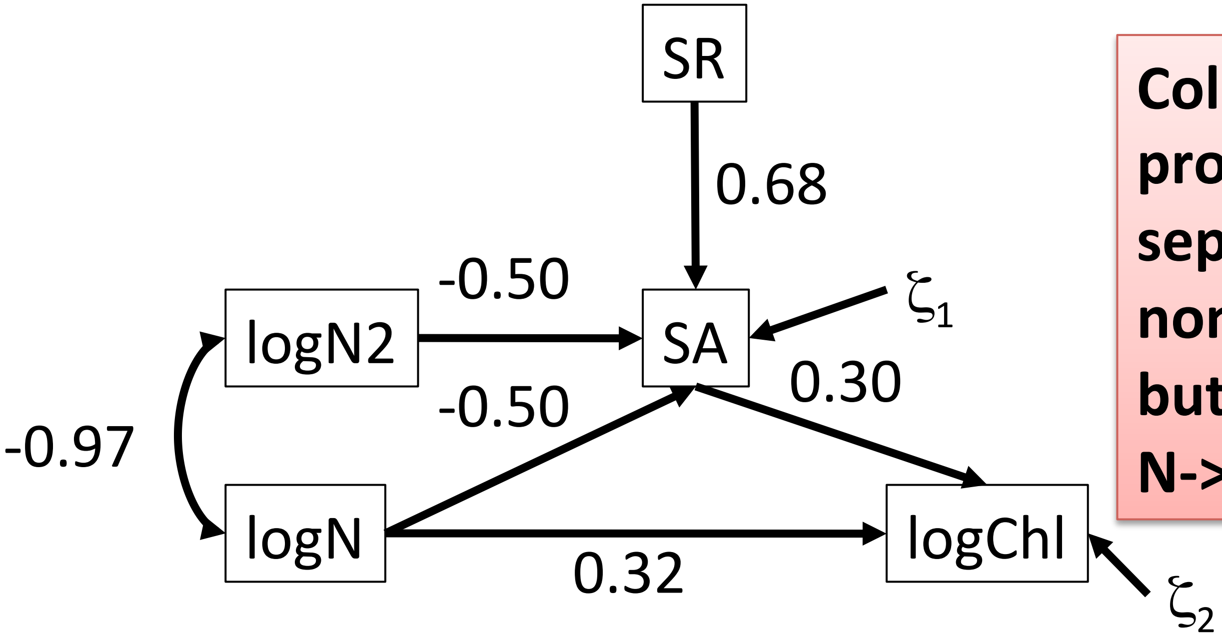
The Survey Design



```
cardSrvy <-
svydesign(
   ids=~Stream+Sub,
   probs=~1,
   data=cards)
```

Also wells grouped in blocks within a stream

But…no Nutrient Effect?

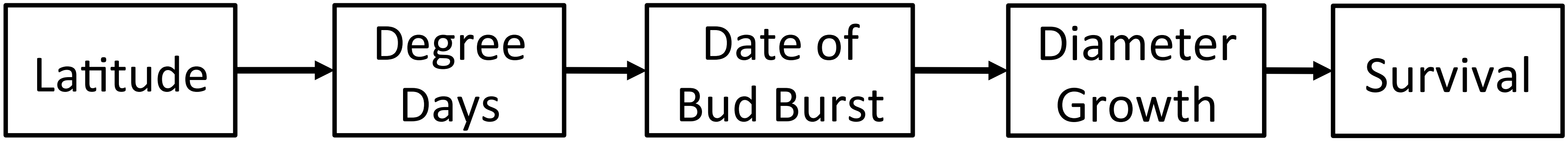


Colinearity still problematic in separating linear v. nonlinear effect – but, there is an N->SA effect!

```
fit.card.survey <-
  lavaan.survey(cardFit,
  cardSrvy, estimator = "MLM")
```

Cardinale et al 2009

Exercise: Shipley's Nested Data



Latitude

Degree Days

Date of Bud Burst

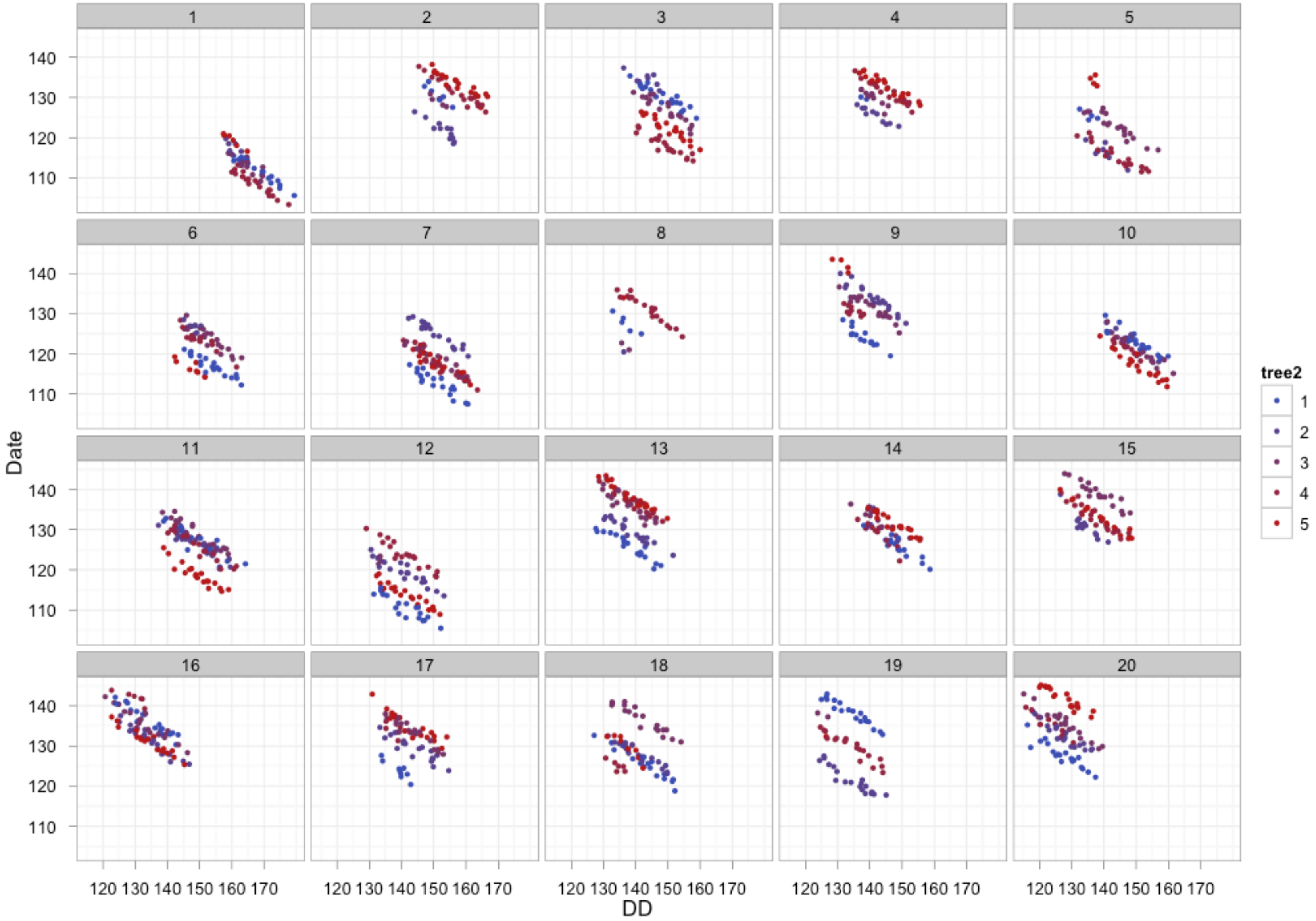
Diameter Growth

Survival

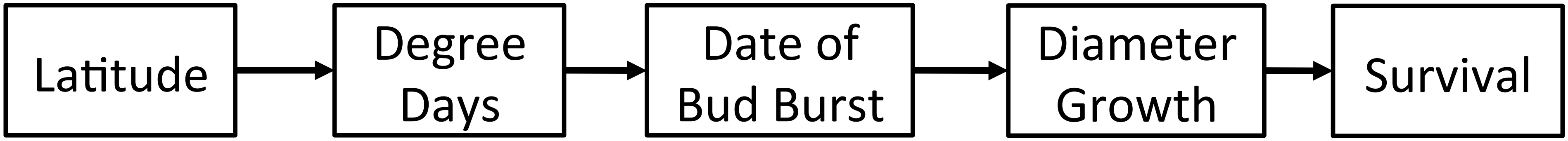
- Simulated data from a fit model
- 20 sites
- 5 trees measured per site

```
> Shipley<-read.table("./Shipley.dat")
> head(Shipley)
  site tree      lat year     Date       DD   Growth  Survival Live
1    1    1 40.38063 1970 115.4956 160.5703 61.36852 0.9996238    1
2    1    2 40.38063 1970 118.4959 158.9896 43.77182 0.8433521    1
3    1    3 40.38063 1970 115.8836 159.9262 44.74663 0.9441110    1
4    1    4 40.38063 1970 110.9889 161.1282 48.20004 0.9568525    1
5    1    5 40.38063 1970 120.9946 157.3778 50.02237 0.9759584    1
6    1    1 40.38063 1972 114.2315 160.6120 56.29615 0.9983398    1
```

Nested Structure in the Data



Exercise: Shipley’s Nested Data

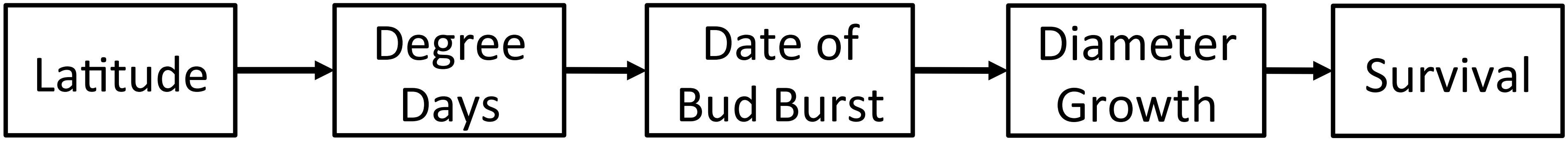


Latitude → Degree Days → Date of Bud Burst → Diameter Growth → Survival

- Simulated data from a fit model
- 20 sites
- 5 trees measured per site

```
> Shipley<-read.table("./Shipley.dat")
> head(Shipley)
  site tree      lat year     Date       DD   Growth  Survival Live
1    1    1 40.38063 1970 115.4956 160.5703 61.36852 0.9996238    1
2    1    2 40.38063 1970 118.4959 158.9896 43.77182 0.8433521    1
3    1    3 40.38063 1970 115.8836 159.9262 44.74663 0.9441110    1
4    1    4 40.38063 1970 110.9889 161.1282 48.20004 0.9568525    1
5    1    5 40.38063 1970 120.9946 157.3778 50.02237 0.9759584    1
6    1    1 40.38063 1972 114.2315 160.6120 56.29615 0.9983398    1
```

Exercise: Shipley's Nested Data


Latitude

Degree Days

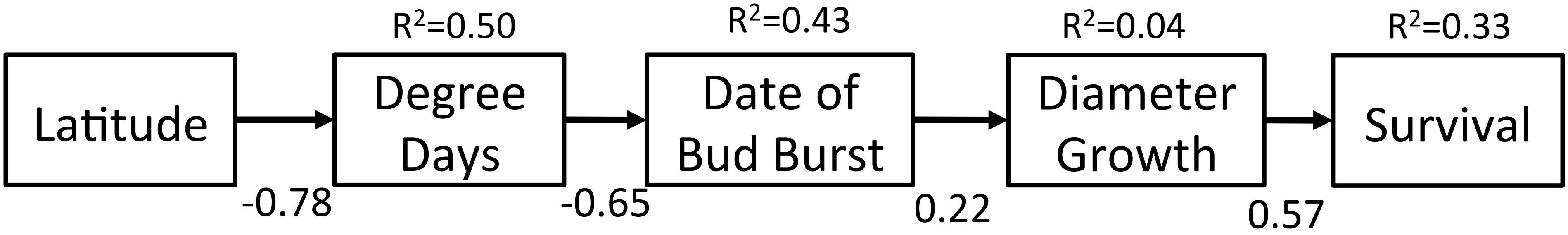
Date of Bud Burst

Diameter Growth

Survival

```
shipSurv <- svydesign(ids =~site + tree,

  probs=~1,

  data=Shipley)
```

Exercise: Shipley’s Nested Data



Latitude → Degree Days → Date of Bud Burst → Diameter Growth → Survival

Degree Days: R^2=0.50

Date of Bud Burst: R^2=0.43

Diameter Growth: R^2=0.04

Survival: R^2=0.33

-0.78

-0.65

0.22

0.57

```
shipFit <- sem(shipMod1, data=Shipley,
estimator=”MLM")
```

```
shipCorrect <- lavaan.survey(shipFit,
shipSurv)
```

An Advanced Outline

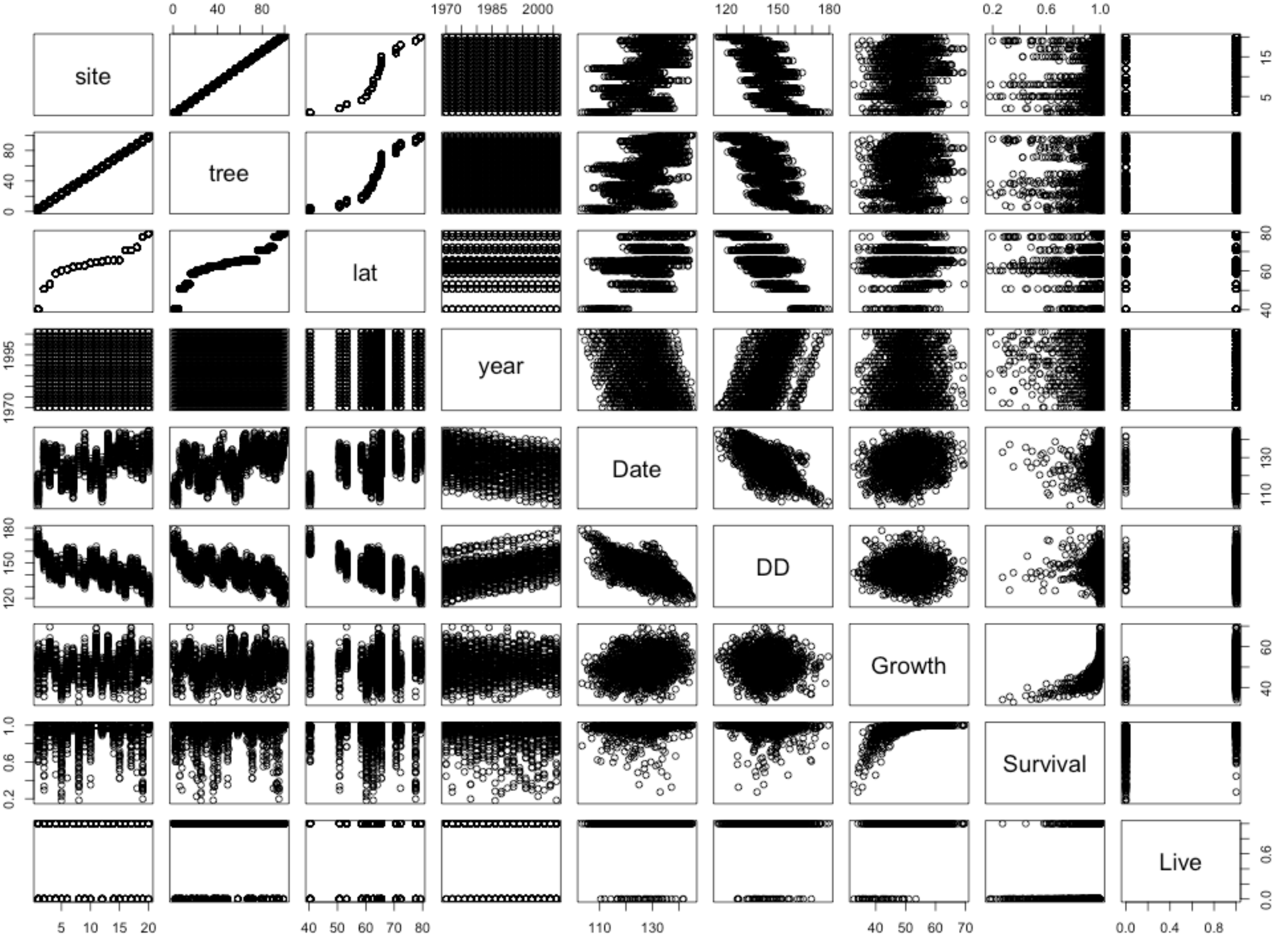
1. Revisiting Sample Size
2. Revisiting Dsep in lavaan
3. Analysis of Nested Survey Data
4. Multilevel Generalized Piecewise SEM
5. Additional Spatial Techniques
6. Panel Models for Lagged Time Effects
7. Growth Curve Models & Time Series

D-Separation in Piecewise models beyond linear regression

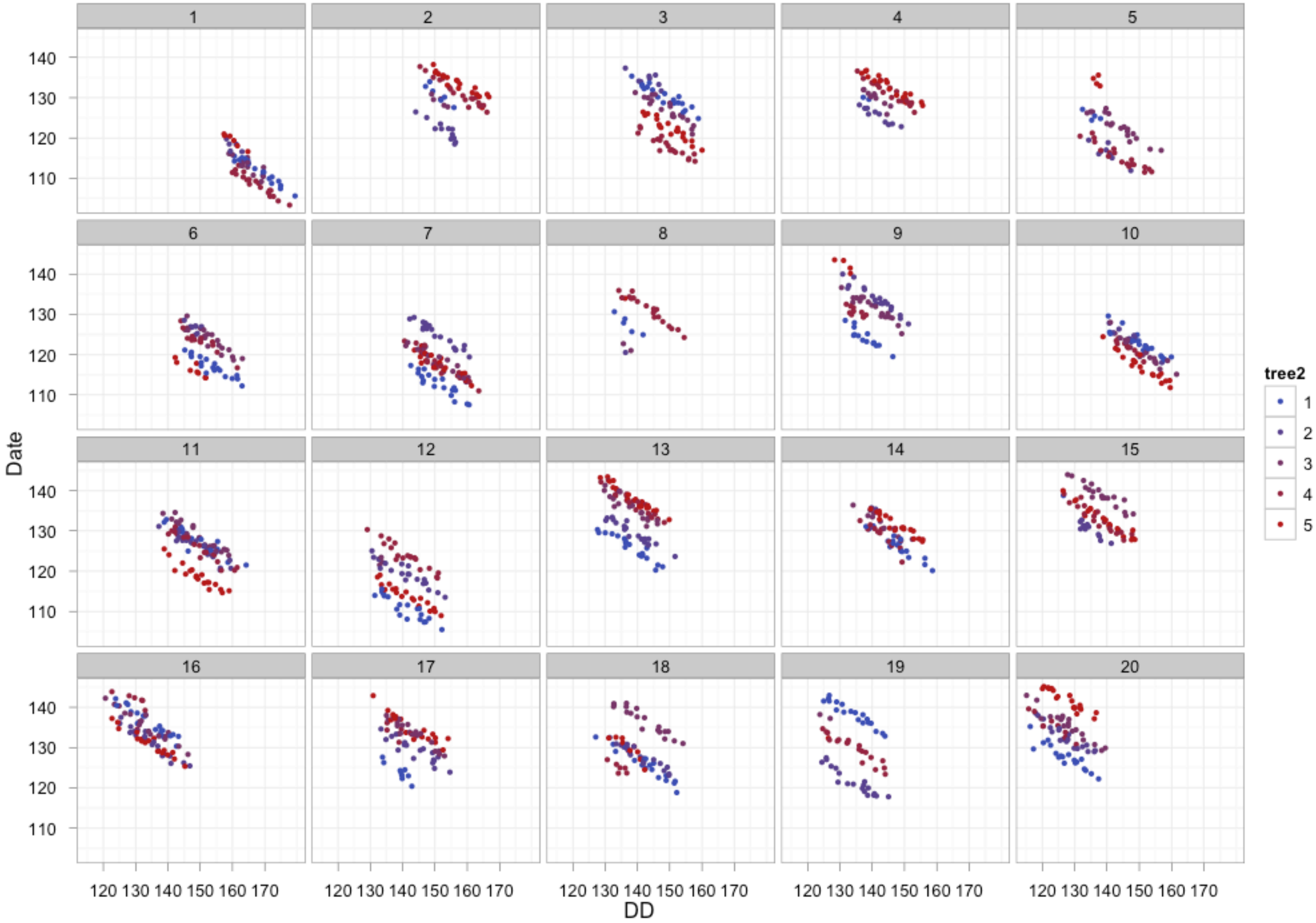
1. We have models that deal with
   1. Hierarchical/nested data (mixed models)
   2. Nonlinear relationships
   3. Non-normal error distributions (glms)
2. The test of the effect of a variable in one of those models serves the same purpose as a partial correlation test in a linear model
3. These p-values can be used for tests of D-Separation

Shipley, B. (2009). Confirmatory path analysis in a generalized multilevel context. *Ecology, 90, 363–368.*

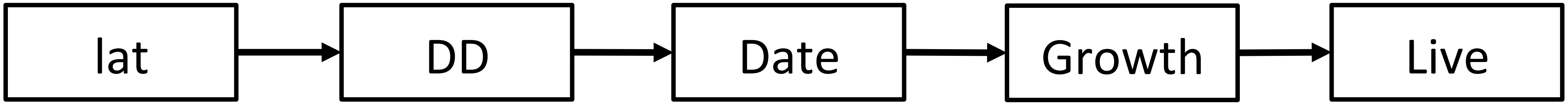
The Simulated Data



Nested Structure in the Data



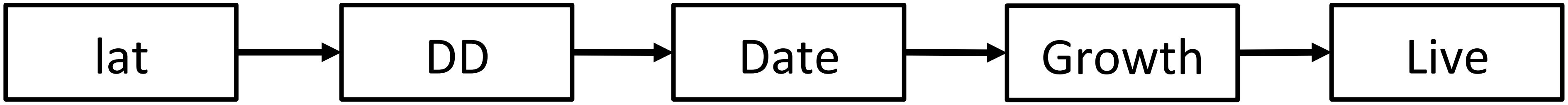
Piecewise Hierarchical Model Fitting

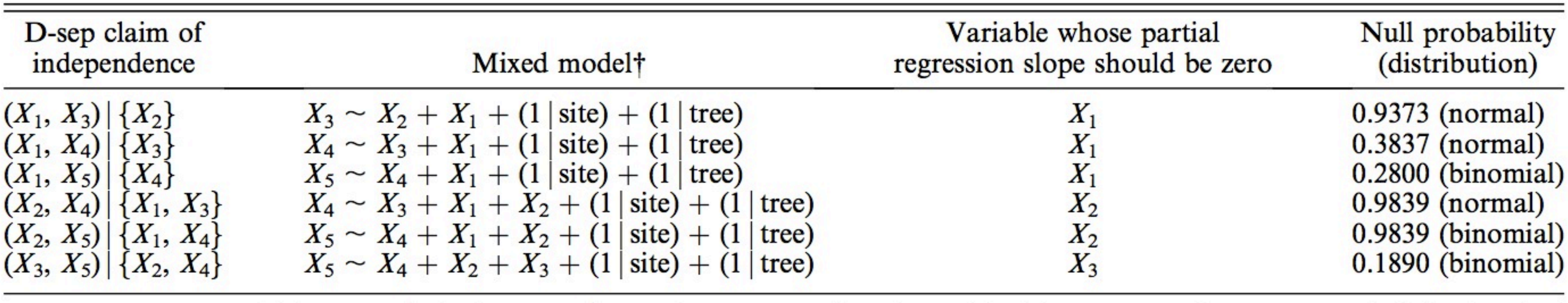


lat → DD → Date → Growth → Live

```
#e.g., for DD -> lat
Shipley<-read.table("./Shipley.dat")
library(nlme)

#model with random intercept
#tree nested in site
Date_dd<-lme(Date~DD,data=Shipley,
  random=~1|site/tree,na.action=na.omit)
```

The Basis Set Needs to Accommodate the Nested Structure

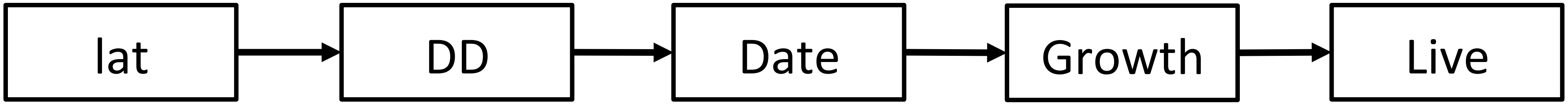




D-sep claim of independence	Mixed model†	Variable whose partial regression slope should be zero	Null probability (distribution)
$(X_1, X_3) \mid \{X_2\}$	$X_3 \sim X_2 + X_1 + (1 \mid \text{site}) + (1 \mid \text{tree})$	X_1	0.9373 (normal)
$(X_1, X_4) \mid \{X_3\}$	$X_4 \sim X_3 + X_1 + (1 \mid \text{site}) + (1 \mid \text{tree})$	X_1	0.3837 (normal)
$(X_1, X_5) \mid \{X_4\}$	$X_5 \sim X_4 + X_1 + (1 \mid \text{site}) + (1 \mid \text{tree})$	X_1	0.2800 (binomial)
$(X_2, X_4) \mid \{X_1, X_3\}$	$X_4 \sim X_3 + X_1 + X_2 + (1 \mid \text{site}) + (1 \mid \text{tree})$	X_2	0.9839 (normal)
$(X_2, X_5) \mid \{X_1, X_4\}$	$X_5 \sim X_4 + X_1 + X_2 + (1 \mid \text{site}) + (1 \mid \text{tree})$	X_2	0.9839 (binomial)
$(X_3, X_5) \mid \{X_2, X_4\}$	$X_5 \sim X_4 + X_2 + X_3 + (1 \mid \text{site}) + (1 \mid \text{tree})$	X_3	0.1890 (binomial)

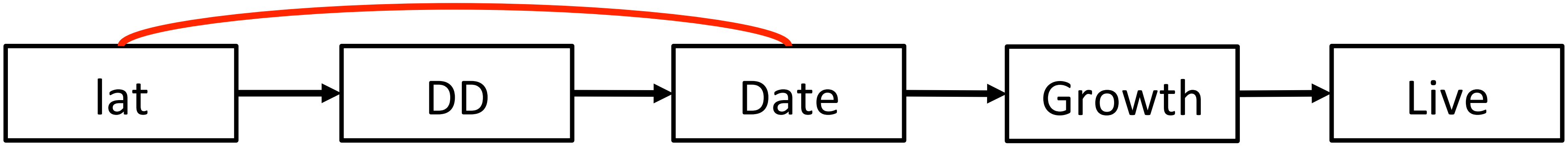
To calculate the partial regression slope, use hierarchical models

Evaluate Independence Claims with Hierarchical Models


lat
DD
Date
Growth
Live

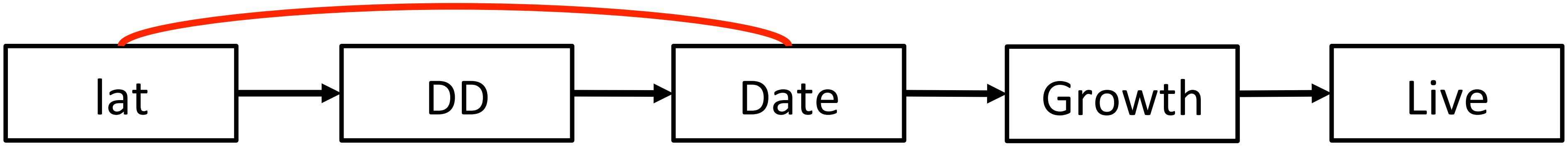
```
#Independence claim: (Date,lat)|{DD}

fit1<-lme(Date~DD+lat,data=Shipley,
  random=~1|site/tree,na.action=na.omit)

summary(fit1)$tTable
```

Evaluate Independence Claims with Hierarchical Models



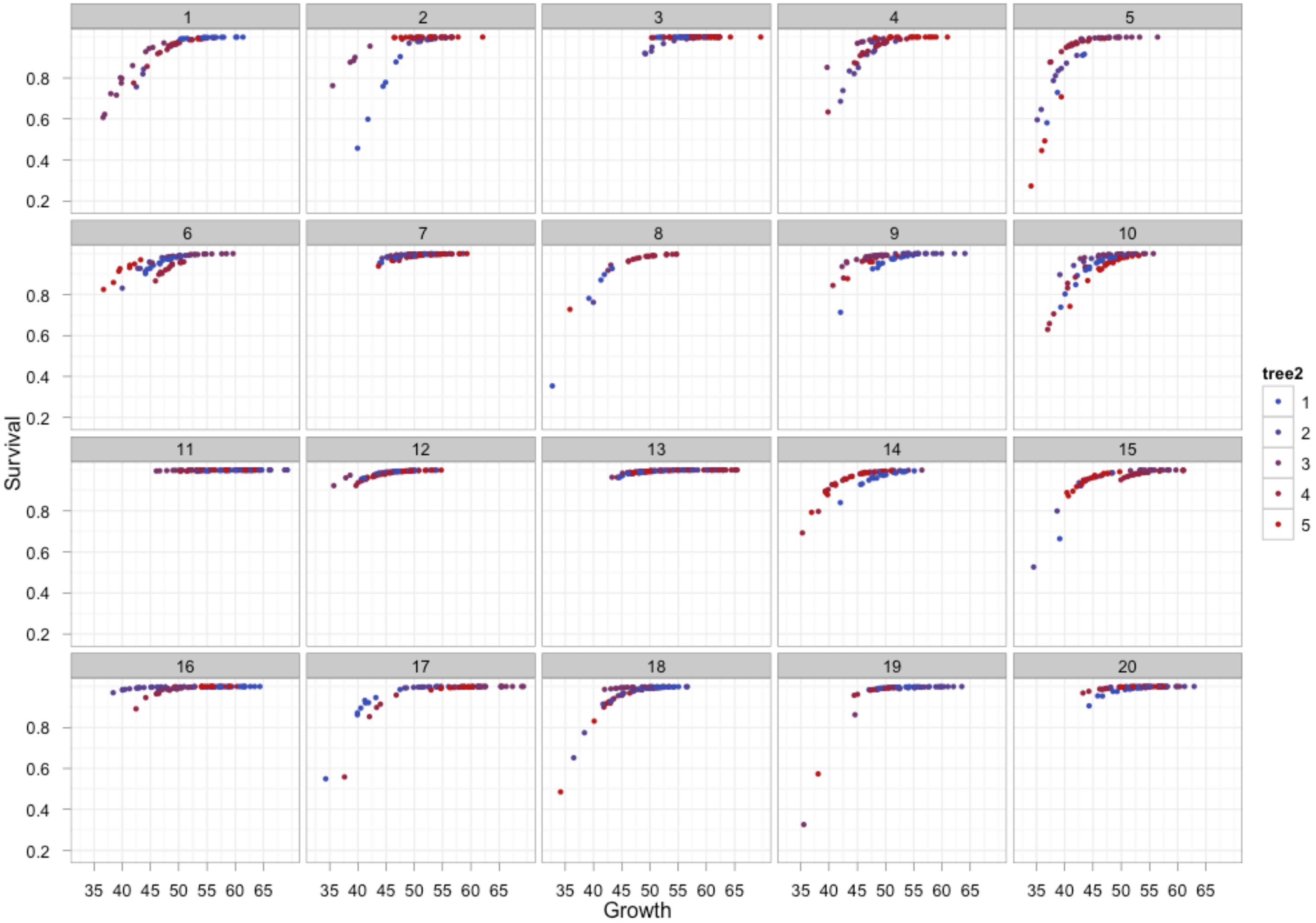
```
#Independence claim: (Date,lat)|{DD}

fit1<-lme(Date~DD+lat,data=Shipley,
  random=~1|site/tree,na.action=na.omit)
```

Evaluate Independence Claims with Hierarchical Models



```
> summary(fit1)$tTable
                   Value   Std.Error   DF      t-value       p-value
(Intercept) 198.915223483 7.337099813 1330   27.11087876 3.185667e-129
DD           -0.497660383 0.004936809 1330 -100.80608521  0.000000e+00
lat          -0.009051378 0.113476607   18   -0.07976426  9.373049e-01
```

We Have Nonlinear Relationships with Non-Normal Distributions



Use generalized linear models – e.g., logit curve with a binomial error

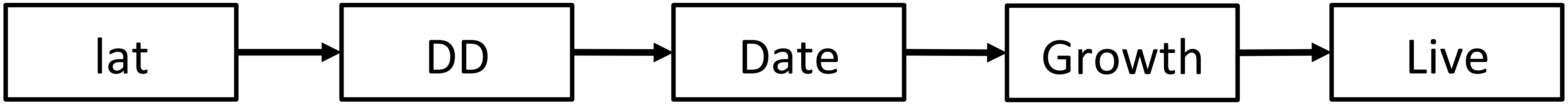
Evaluate Independence Claims with GLMMs

```
###need lme4 for the glmms
library(lme4)

#Independence claim with glmm (Live,lat)|{Growth}
fit4<-lmer(Live~Growth+lat+(1|site)+(1|tree),
        data=Shipley, na.action=na.omit,
        family=binomial(link="logit"))
```

Evaluate Independence Claims with GLMMs

```
> summary(fit4)@coefs
                Estimate Std. Error   z value     Pr(>|z|)
(Intercept) -14.43837636 2.65394004 -5.440355 5.317446e-08
Growth        0.35530576 0.04554481  7.801235 6.130440e-15
lat           0.03051257 0.02819180  1.082321 2.791099e-01
```

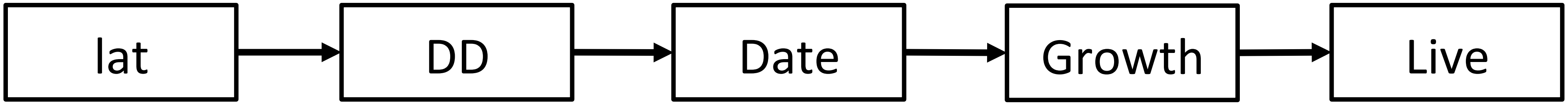
Putting it All Together in Shipley's Test



lat → DD → Date → Growth → Live

```
#note, since we're logging things
#we can use log(a)+log(b) = log(a*b)

> fisherC <- -2* log(9.373049e-01 * 3.836896e-01 *
                     7.667083e-01 * 2.791099e-01 *
                     3.159286e-01 * 1.519170e-01)
> fisherC
[1] 11.20225

>1-pchisq(fisherC, 2*6)

[1] 0.5116698
```

AIC and D-Sep



lat → DD → Date → Growth → Live

$$\boxed{\mathbf{AIC = C + 2K}}$$

Why? Shipley has proven that:

$$-2\ \ln(L(\text{model} \mid \text{data})) = -2 \sum \ln(p) = \text{Fisher's } C$$

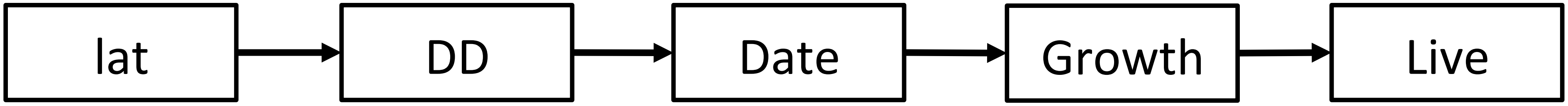
Shipley, B. In Press. The AIC model selection method applied to path analytic models compared using a d-separation tests. *Ecology*.

AIC and D-Sep

lat → DD → Date → Growth → Live

```
#each piece except last has 5 parameters - slope,
# intercept, variance, and random variance for
#slope & intercept, so, K=5*4-1

> fisherC + 2*(5*4-1)
[1] 49.20225
```

Using `piecewiseSEM` for GLMMs



lat

DD

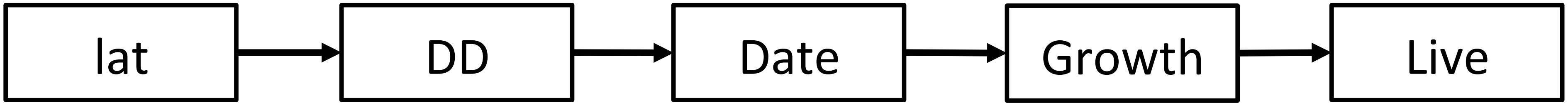
Date

Growth

Live

```
library(piecewiseSEM)

shipley2009.modlist = list(
 lme(DD~lat, random = ~1|site/tree, na.action = na.omit,
      data = Shipley),

 lme(Date~DD, random = ~1|site/tree, na.action = na.omit,
      data = Shipley),

 lme(Growth~Date, random = ~1|site/tree, na.action = na.omit,
      data = Shipley),

 glmer(Live~Growth+(1|site)+(1|tree),
        family=binomial(link = "logit"), data = Shipley) )
```

Using `piecewiseSEM` for GLMMs



lat

DD

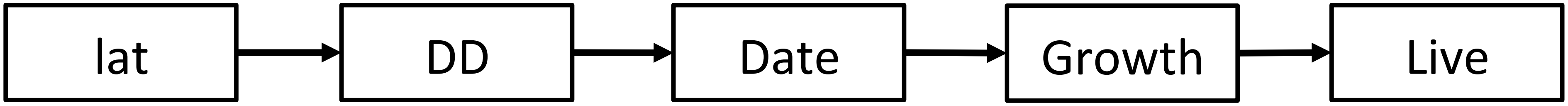
Date

Growth

Live

```
> sem.fit(shipley2009.modlist, data=Shipley)

$missing.paths
   missing.path estimate std.error   DF crit.value p.value
1    Date <- lat   -0.009     0.113   18     -0.080   0.937
2 Growth <- lat   -0.099     0.111   18     -0.893   0.384
3    Live <- lat    0.030     0.030   NA      1.028   0.304
4   Growth <- DD   -0.011     0.036 1329     -0.297   0.767
5     Live <- DD    0.027     0.027   NA      1.004   0.315
6   Live <- Date   -0.047     0.030   NA     -1.562   0.118
```

Using `piecewiseSEM` for GLMMs



lat

DD

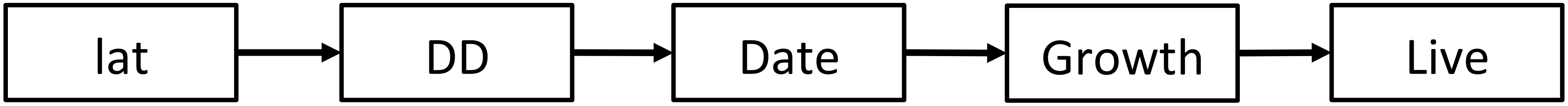
Date

Growth

Live

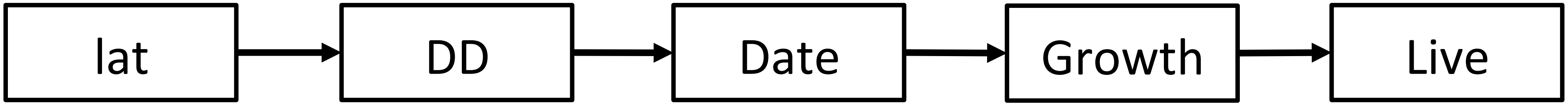
```
> sem.fit(shipley2009.modlist, data=Shipley)
...


$Fisher.C
Fisher.C         k         P
  11.54    12.000    0.484

...
```

AIC from `piecewiseSEM`



```
> sem.fit(shipley2009.modlist, data=Shipley)
...


$AIC
    AIC   AICc  K    n
1 49.54 50.079 19 1431

...
```

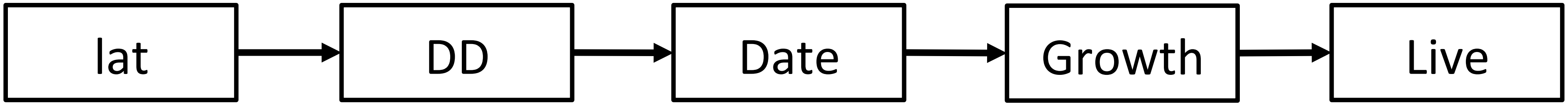
What do we gain from random effects in `piecewiseSEM`?



```
#Want to compare to a lavaan fit?
    > sem.lavaan(shipley2009.modlist, data=Shipley)

    lavaan (0.5-20) converged normally after  27 iterations

                                                      Used       Total
      Number of observations                          1431        1900

      Estimator                                         ML
      Minimum Function Test Statistic               38.433
      Degrees of freedom                                 6
      P-value (Chi-square)                           0.000
```

Hierarchical structure & non-normality necessary for fit

Variation Explained


lat

DD

Date

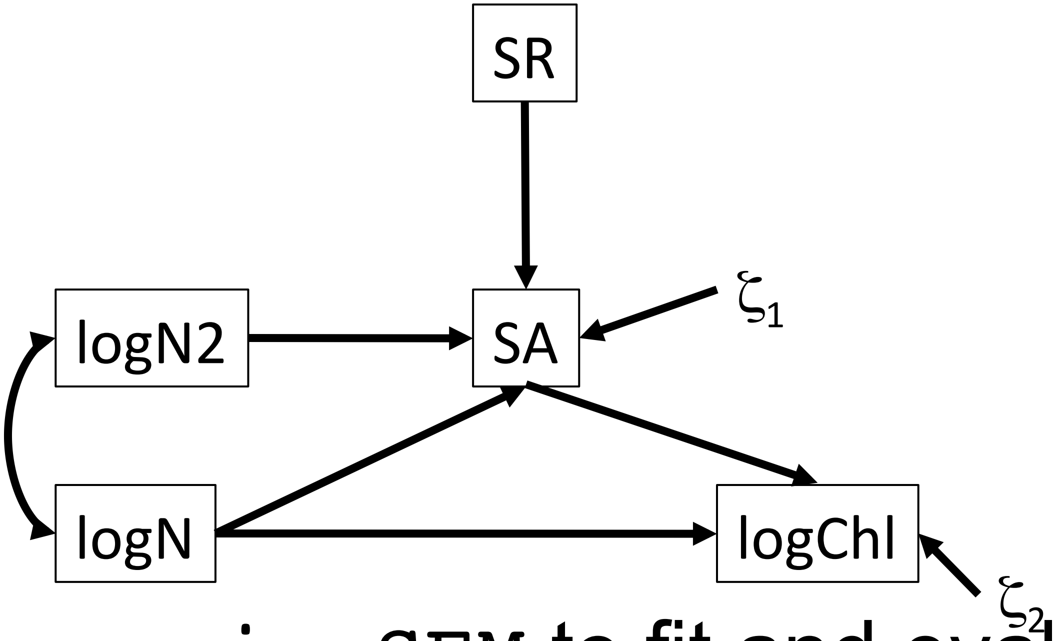
Growth

Live

- Mixed Models have two kinds of R^2
  - **Marginal**: R^2 due to fixed effects only
  - **Conditional**: R^2 due to fixed & random effects

```
> sem.model.fits(shipley2009.modlist)
     Class   Family     Link  Marginal Conditional       AIC
1      lme gaussian identity 0.4864825   0.6990231 9166.9738
2      lme gaussian identity 0.4095855   0.9838829 4694.9821
3      lme gaussian identity 0.1079098   0.8366353 7611.3338
4 glmerMod binomial    logit 0.5589201   0.6291994  261.0824
```

Exercise: Multilevel Model with Richness Model



- Use `piecewiseSEM` to fit and evaluate this model
- Use the squared term (may need to center)
- Stream and Sub nested within Stream are your random effects

Final Thoughts on Piecewise Fits

- You can use anything: generalizes linear models, mixed models, generalized least squares fits with temporal or spatial autocorrelation built-in
- Bayesian methods also provide flexible frameworks for piecewise models, but cannot calculate omnibus fit tests

An Advanced Outline

1. Revisiting Sample Size
2. Revisiting Dsep in lavaan
3. Analysis of Nested Survey Data
4. Multilevel Generalized Piecewise SEM
5. Additional Spatial Techniques
6. Panel Models for Lagged Time Effects
7. Growth Curve Models & Time Series

Spatial Effects

There are two key issues regarding space:

(1) Are their things to learn about the other factors that could explain variations in the data that vary spatially?

(2) Do we have nonindependence in our residuals?

<u>Recent reference on the subject</u>:
Hawkins, BA (2011) Eight (and a half) deadly sins of spatial analysis. Journal of Biogeography. doi:10.1111/j.1365-2699.2011.02637.x

Spatial References

Reference where mechanistic questions have been asked:

Grace JB and Guntenspergen, GR (1999) The effects of landscape position on plant species density: Evidence of past environmental effects in a coastal wetland. Ecoscience Vol. 6, pp. 381-391.

(Distance from mouth of river and edge of shore served as proxies for past storm-driven saltwater intrusions.)

Mancera et al. (2005) Fine-scale spatial variation in plant species richness and its relationship to environmental conditions in coastal marshlands. Plant Ecology 178:39-50.

(Showed fine-scale matching of plant to abiotic conditions in severe environments. No evidence of mass effects.)

Spatial References

<u>Reference where autocorrelation has been adjusted for in SEM studies:</u>

Harrison, S and Grace, JB (2007) Biogeographic affinity contributes to our understanding of productivity-richness relationships at regional and local scales. American Naturalist. 170:S5-S15.

Degrees of freedom and sample size adjusted using Moran's I.

Adjusting for Spatial Autocorrelation

1. Is there spatial autocorrelation?
   - Calculate Moran's I on residuals

2. If yes, it may bias the SE of parameters
   - Calculate new effective sample size
   - Recalculate parameter SE from VCOV matrix

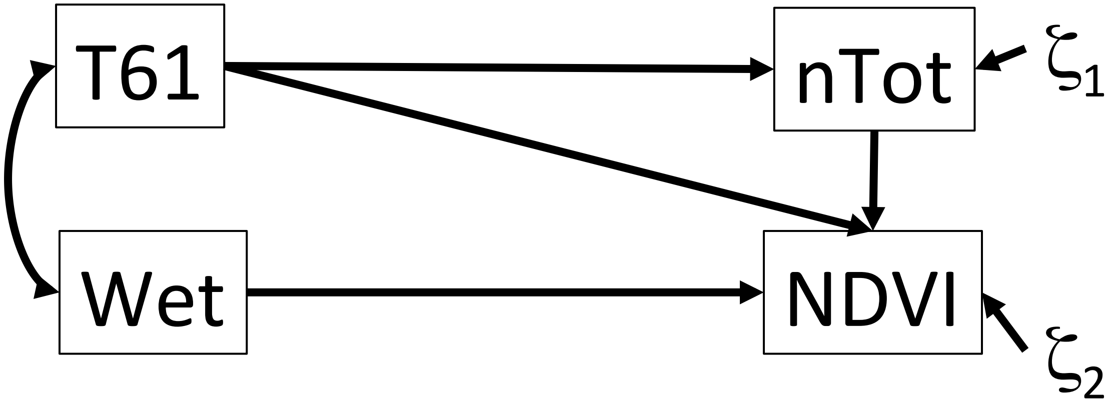
3. Correct by recalculating SE and Z-tests
   - By hand…for now

Example: NDVI in a Boreal System

Data Contains Spatial Information

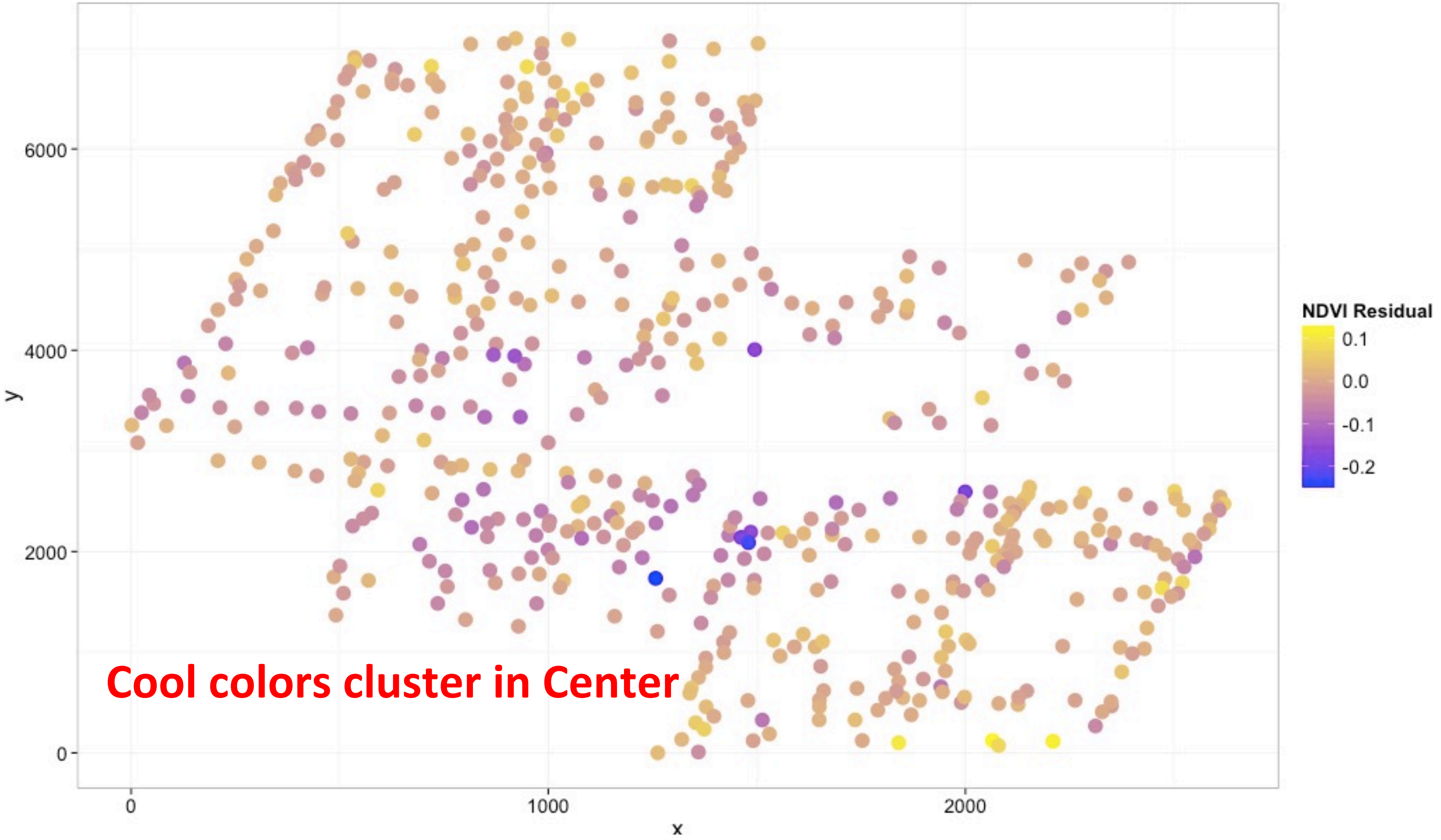
```
> boreal <- read.table("./Boreality.txt", header=T)

> head(boreal)
  point       x       y Oxalis   boreal nBor nTot       Grn     NDVI     T61
1     1 2109.70 2093.52      0 15.38462    2   13 0.0597027 0.480180 296.367
2     2 2190.18 2105.71      1 19.04762    4   21 0.0514881 0.483990 296.367
3     3 2064.48 2052.77      1 20.00000    6   30 0.0509510 0.489213 296.367
4     4 2277.34 2103.42      0 15.38462    2   13 0.0521183 0.473226 296.367
5     5 2347.91 2074.81      0  0.00000    0   13 0.0422267 0.405898 296.785
6     6 2437.21 2086.95      0 16.66667    1    6 0.0417779 0.424769 296.367
        Wet nBroSq
1 -0.0264378      4
2 -0.0234048     16
3 -0.0189264     36
4 -0.0280431      4
5 -0.0292287      0
6 -0.0229209      1
```

Model of NDVI in a Boreal System



```
borModel <- '
  NDVI ~ nTot + T61 + Wet
  nTot ~ T61’

borFit <- sem(borModel, data=boreal,
meanstructure=T)
```

Residuals Might be Spatially Correlated



Cool colors cluster in Center

```
borRes <- as.data.frame(residuals(borFit, "casewise"))
```

Calculating a Distance Matrix

1. Distance matrices tell us how close points are in space
    - ape library calculates matrix and Moran's I

```
library(ape)
distMat <- as.matrix(dist(
               cbind(boreal$x, boreal$y)))
```

2. We take the inverse, as closer points have greater similarity
    - The diagonal is 0, as there is no correlation within a point

```
distsInv <- 1/distMat
diag(distsInv) <- 0
```

Moran’s I using Residuals

```
> mi.ndvi <- Moran.I(borRes$NDVI, distsInv)
> mi.ndvi
$observed
[1] 0.08265236

$expected
[1] -0.001879699

$sd
[1] 0.003985846

$p.value
[1] 0
```

Data is more spatially correlated than expected – need a correction

Adjusting the Effective Sample Size

- Estimate of effective sample size (Fortin & Dale 2005, p. 223, Equation 5.15):

$$\boxed{n' = \frac{n^2}{\sum_{i=1}^{n} \sum_{j=1}^{n} Cor(x_i, x_j)}}$$

- For first-order autocorrelation *ρ* and large *n*:

$$\boxed{n' \cong n \cdot \frac{1-\rho}{1+\rho}}$$

Where did that SE Come From?

SE's calculated as the square root of the variance-covariance matrix of parameters

```
> vcov(borFit)
           NDVI~nT   NDVI~T6   NDVI~W    nT~T61    NDVI~~    nTt~~T    NDVI~1
NDVI~nTot      0.000
NDVI~T61       0.000     0.000
NDVI~Wet       0.000     0.000     0.017
nTot~T61       0.000     0.000     0.000     0.298
NDVI~~NDVI     0.000     0.000     0.000     0.000     0.000
nTot~~nTot     0.000     0.000     0.000     0.000     0.000    47.113
NDVI~1         0.000    -0.002     0.027     0.000     0.000     0.000     0.448
nTot~1         0.000     0.000     0.000   -88.329     0.000     0.000     0.000
```

Given that Var(x) = sum[(x-x')^2]/n, all we need to do is change n!

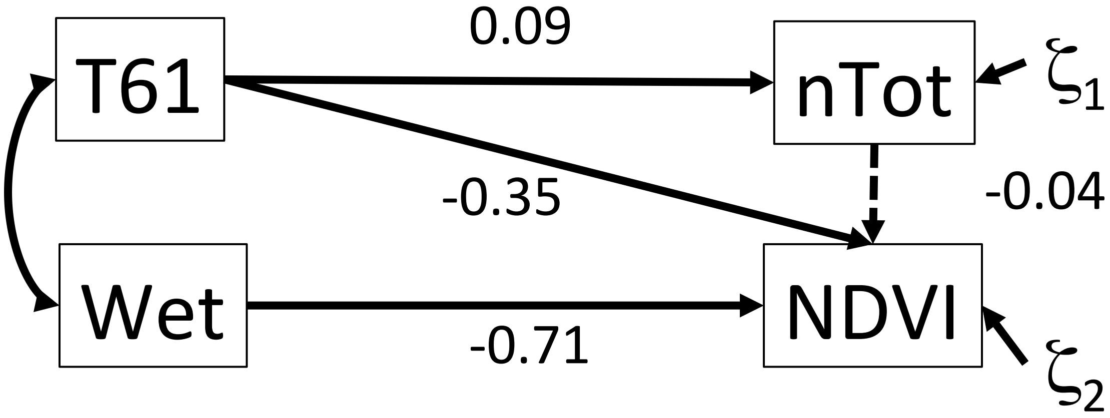
Calculating a New Standard Error

```
#New SE
ndvi.var <- diag(vcov(borFit))[1:3]

ndvi.se <- sqrt(ndvi.err*nrow(boreal)/n.ndvi)

ndvi.se

 NDVI~nTot     NDVI~T61     NDVI~Wet
0.0001848868 0.0024493462 0.1436405689
```

Compare to former SE – new SE is wider

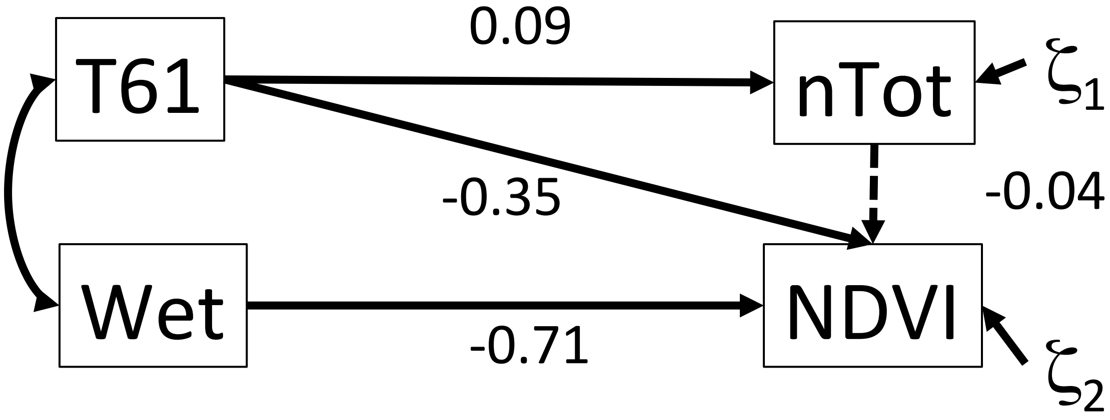
```
> sqrt(diag(vcov(borFit)))[1:3]
   NDVI~nTot     NDVI~T61     NDVI~Wet
0.0001701878 0.0022546163 0.1322207383
```

Z-Tests Show Not All Paths Still ≠ 0



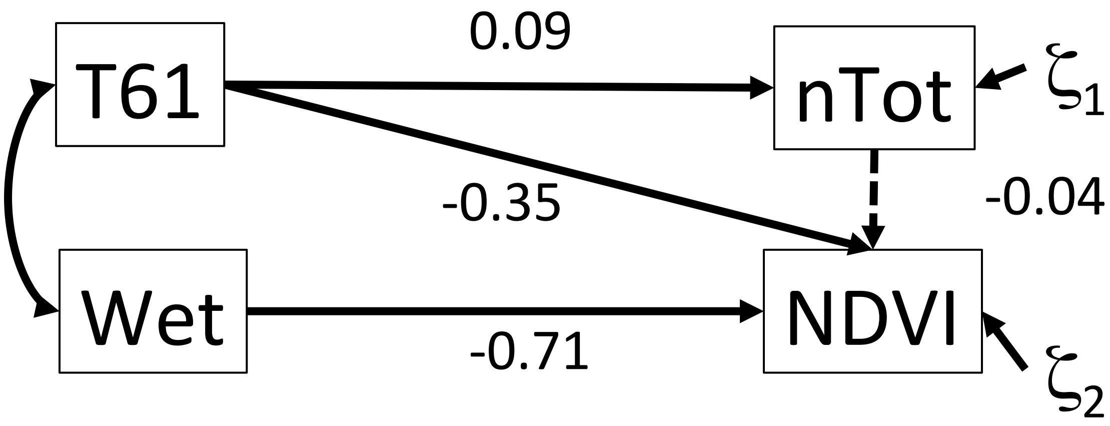
```
> z <- coef(borFit)[1:3]/ndvi.se

> 2*pnorm(abs(z), lower.tail=F)

   NDVI~nTot       NDVI~T61       NDVI~Wet
5.366259e-02  1.517587e-47 3.404230e-194
```

Or…do it the easy way



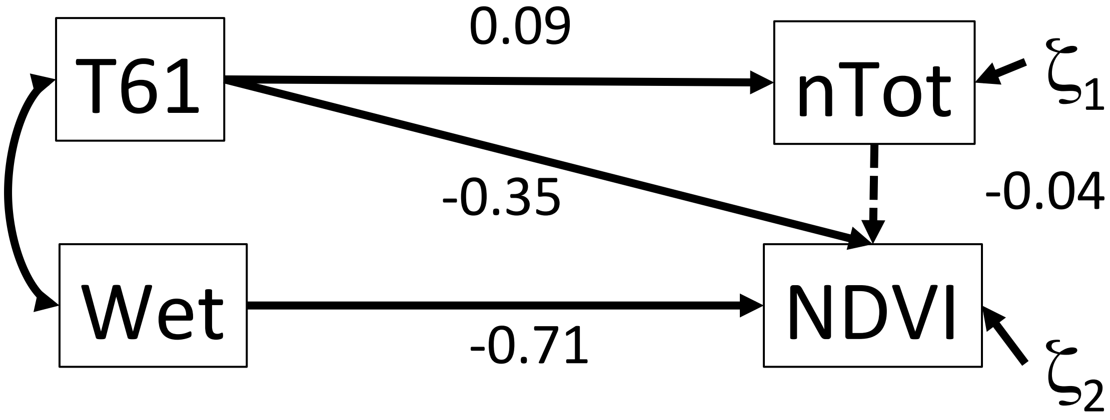
```
> library(semTools)

> spatialCorrect(borFit, boreal$x, boreal
$y)
```

Or…do it the easy way



```
$Morans_I
$Morans_I$NDVI
    observed     expected          sd p.value    n.eff
1 0.08265236 -0.001879699 0.003985846       0 451.6189

$Morans_I$nTot
    observed     expected          sd p.value    n.eff
1 0.03853411 -0.001879699 0.003998414       0 493.4468

...
```

Or…do it the easy way

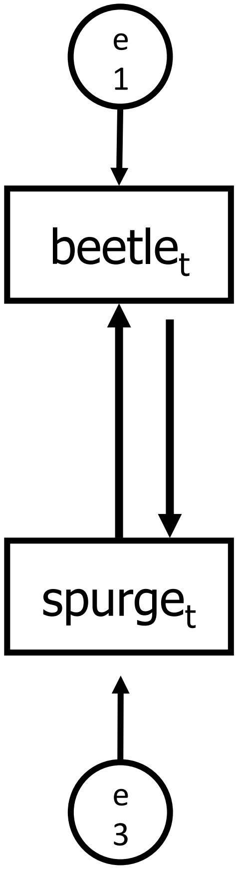


```
...
$parameters
$parameters$NDVI
            Parameter     Estimate    n.eff      Std.err   Z-value      P(>|z|)
NDVI~nTot   NDVI~nTot -0.0003567484 451.6189 0.0001848868  -1.92955  5.366259e-02
NDVI~T61     NDVI~T61 -0.0354776273 451.6189 0.0024493462 -14.48453  1.517587e-47
NDVI~Wet     NDVI~Wet -4.2700526589 451.6189 0.1436405689 -29.72734 3.404230e-194
NDVI~~NDVI NDVI~~NDVI  0.0017298286 451.6189 0.0001151150  15.02696  4.889505e-51
NDVI~1         NDVI~1 10.8696158663 451.6189 0.7268790958  14.95382  1.470754e-50

$parameters$nTot
            Parameter   Estimate    n.eff     Std.err   Z-value      P(>|z|)
nTot~T61     nTot~T61   1.170661 493.4468   0.5674087  2.063171 3.909634e-02
nTot~~nTot nTot~~nTot 112.051871 493.4468   7.1336853 15.707431 1.345204e-55
nTot~1         nTot~1 -322.936937 493.4468 168.1495917 -1.920534 5.479054e-02
```

An Advanced Outline

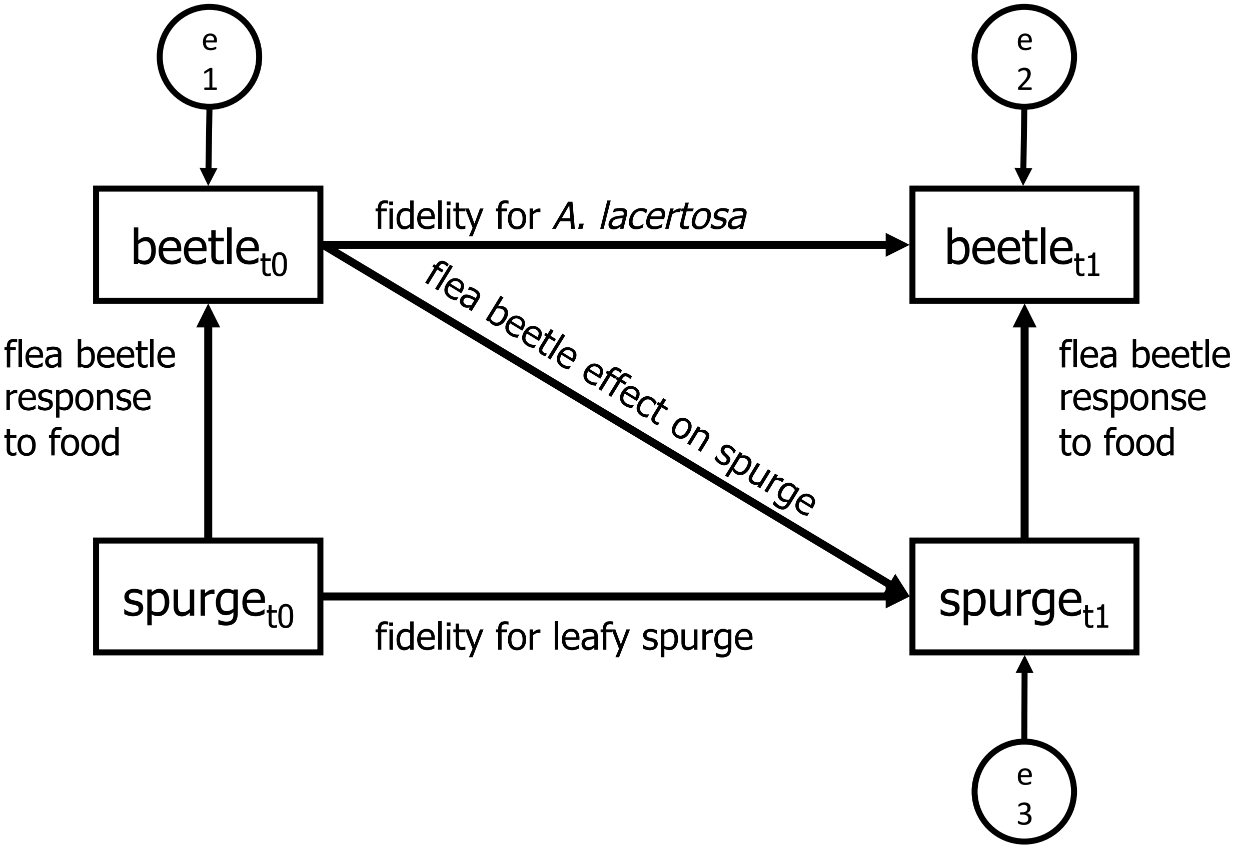
1. Revisiting Sample Size
2. Revisiting Dsep in lavaan
3. Analysis of Nested Survey Data
4. Multilevel Generalized Piecewise SEM
5. Additional Spatial Techniques
6. Panel Models for Lagged Time Effects
7. Growth Curve Models & Time Series

Problem of Non-Recursive Models



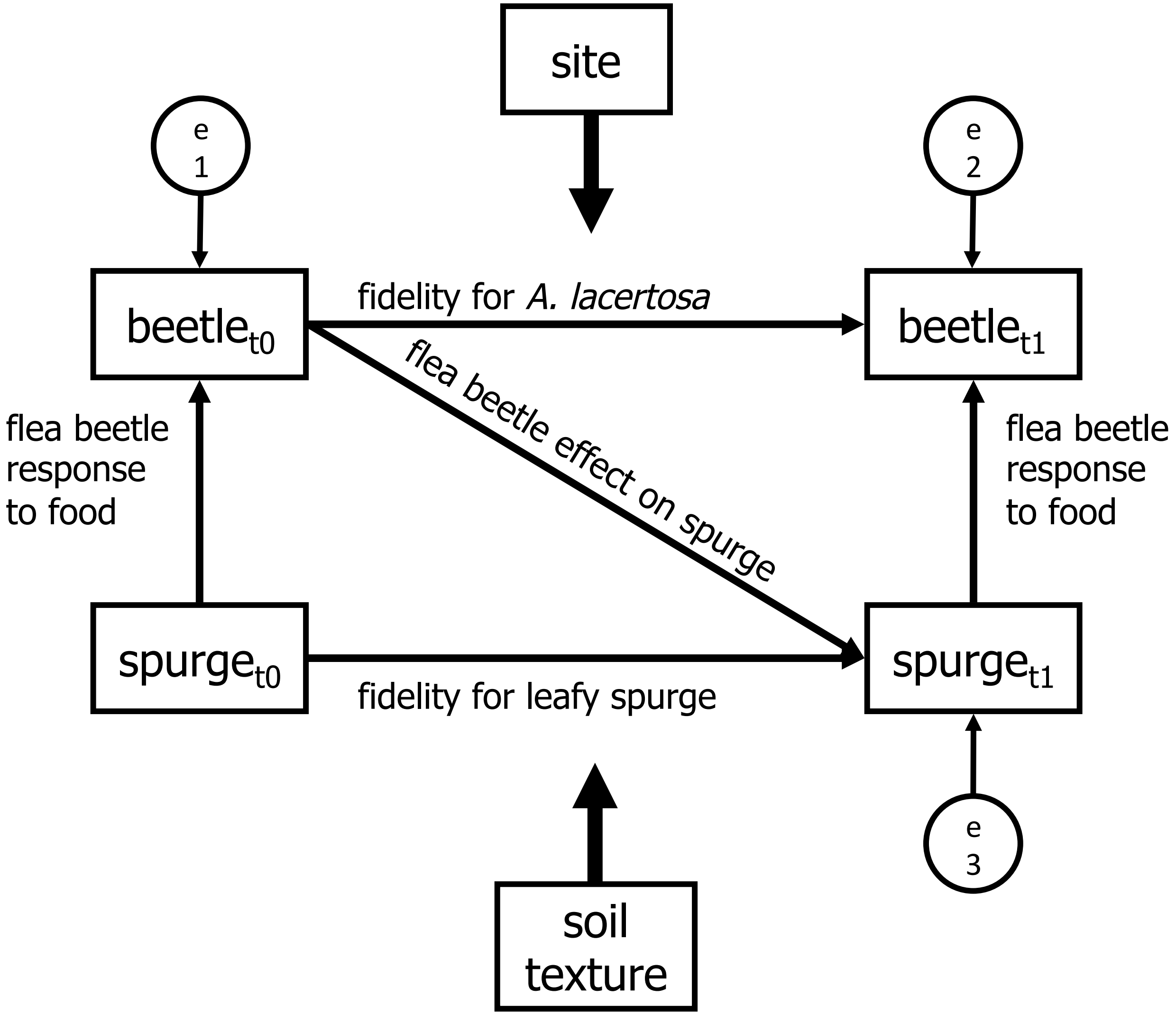
Larson, DL and Grace, JB (2004) Temporal Dynamics of Leafy Spurge (Euphorbia esula) and Two Species of Flea Beetles (Aphthona spp.) Used as Biological Control Agents. Biological Control 29:207–214.

Longitudinal Studies – Time-Step (Panel) Model for Lagged Effects



Larson, DL and Grace, JB (2004) Temporal Dynamics of Leafy Spurge (Euphorbia esula) and Two Species of Flea Beetles (Aphthona spp.) Used as Biological Control Agents. Biological Control 29:207–214.

Time-independent dynamics in a Panel Model

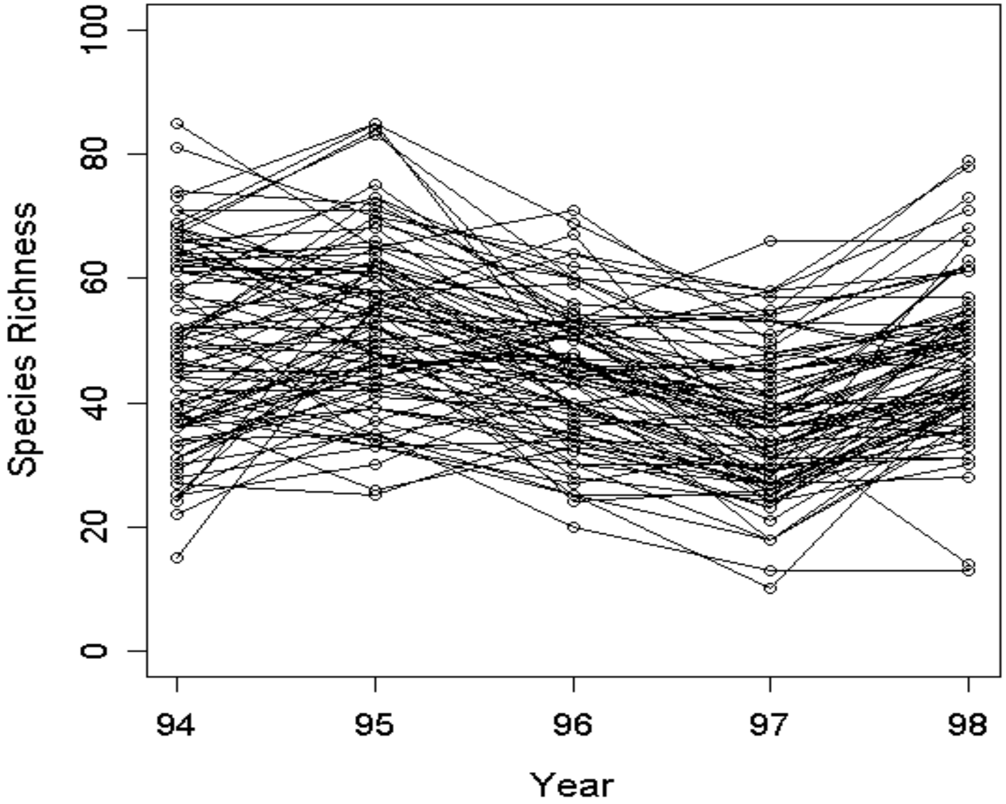


Larson, D.L., Grace, J.B., and Larson, J.L. 2008. Long-term dynamics of leafy spurge (*Euphorbia esula*) and its biocontrol agent, the flea beetle *Aphthona lacertosa*. *Biological Control* 47:250-256.

An Advanced Outline

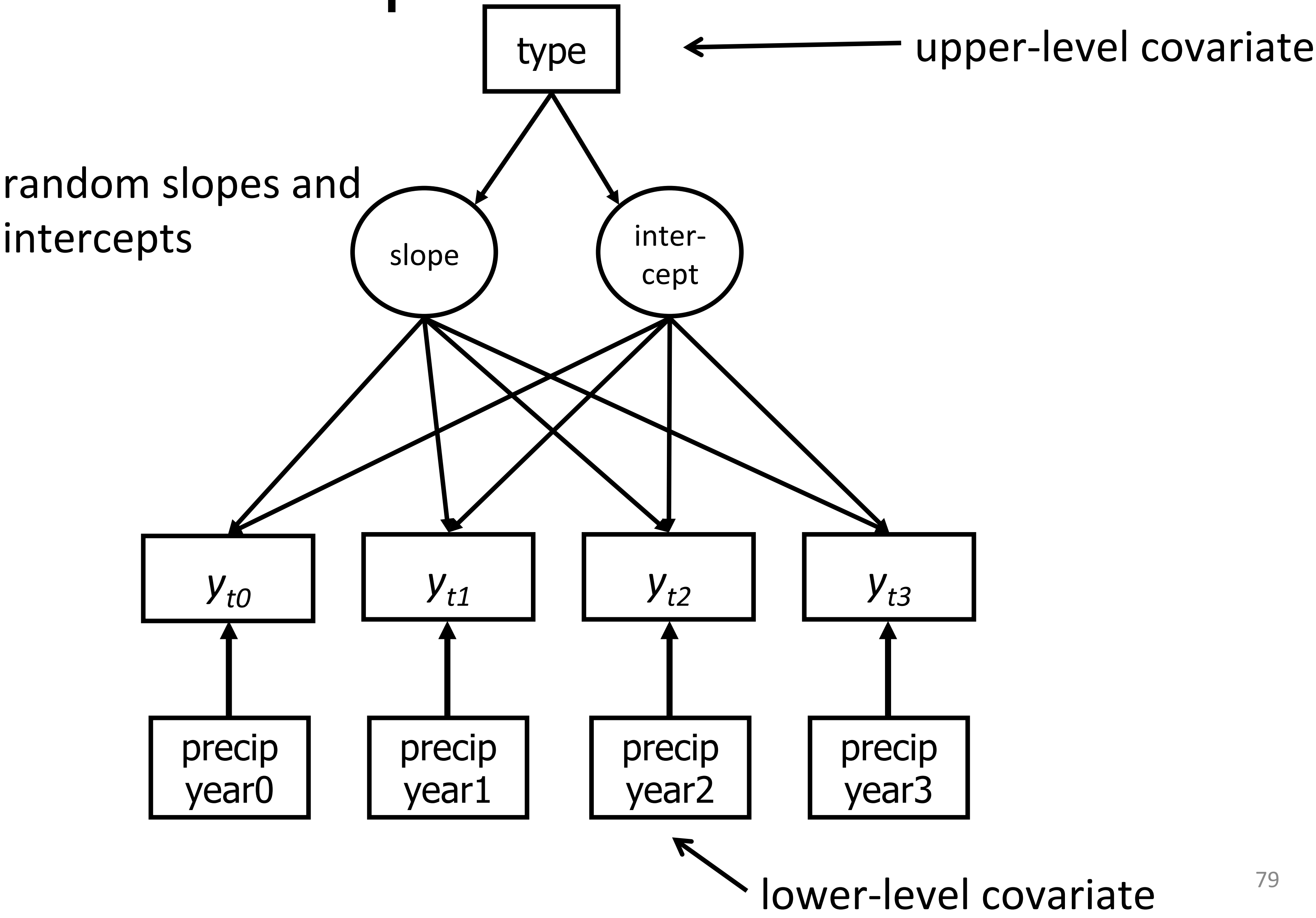
1. Revisiting Sample Size
2. Revisiting Dsep in lavaan
3. Multilevel Generalized Piecewise SEM
4. Additional Spatial Techniques
5. Panel Models for Lagged Time Effects
6. Growth Curve Models & Time Series

Latent Trajectory Models for Timeseries & Repeated Measures



Grace, J.B., Keeley, J., Johnson, D., and Bollen, K.A. 2012. Structural equation modeling and the analysis of long-term monitoring data. In: Gitzen, R.A., Millspaugh, J.J., Cooper, A.B., and Licht, D.S. Design and Analysis of Long-Term Ecological Monitoring Studies. Cambridge University Press.

Latent Trajectory Models for Repeated Measures



type

upper-level covariate

random slopes and intercepts

slope

inter-cept

y_{t0}

y_{t1}

y_{t2}

y_{t3}

precip year0

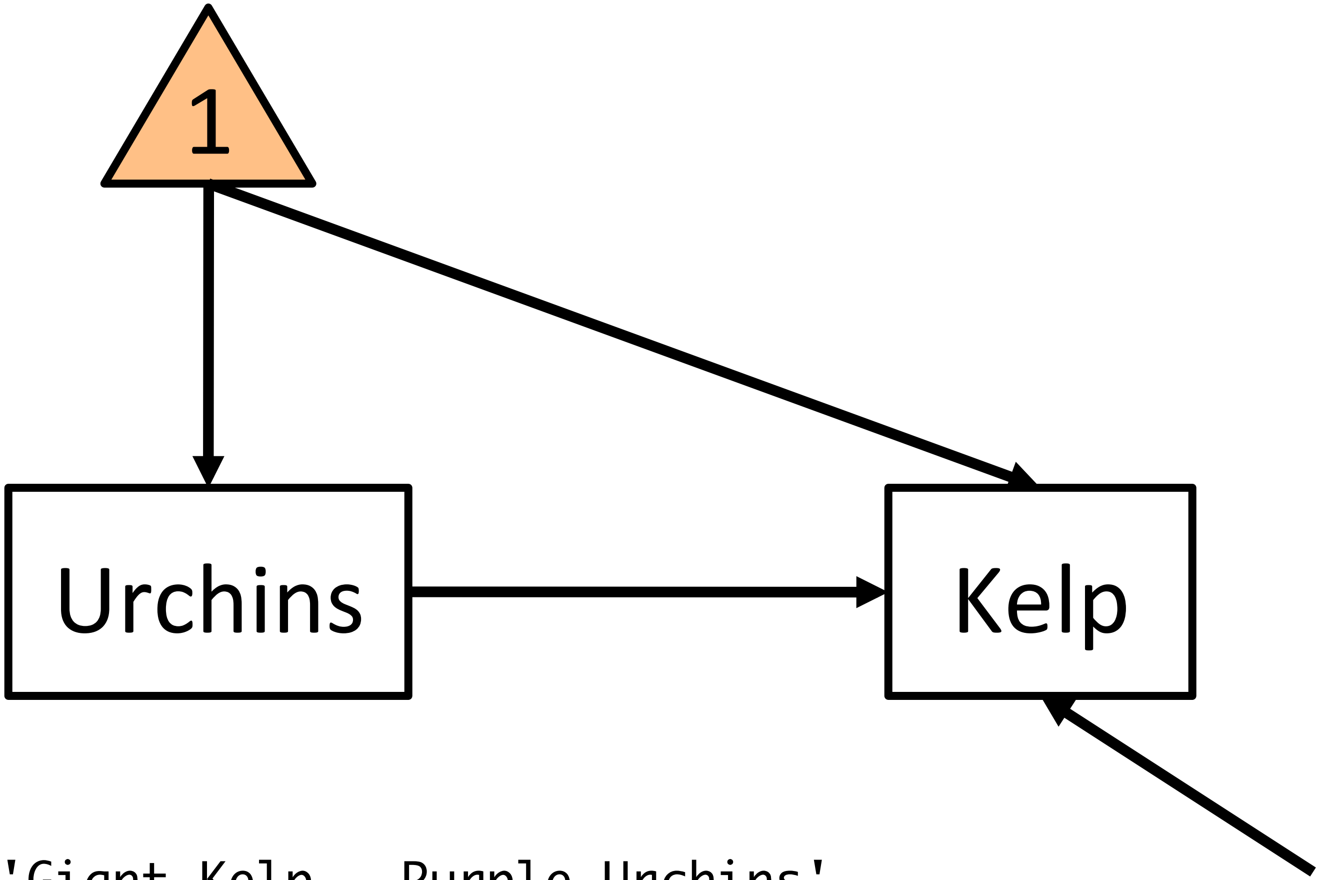
precip year1

precip year2

precip year3

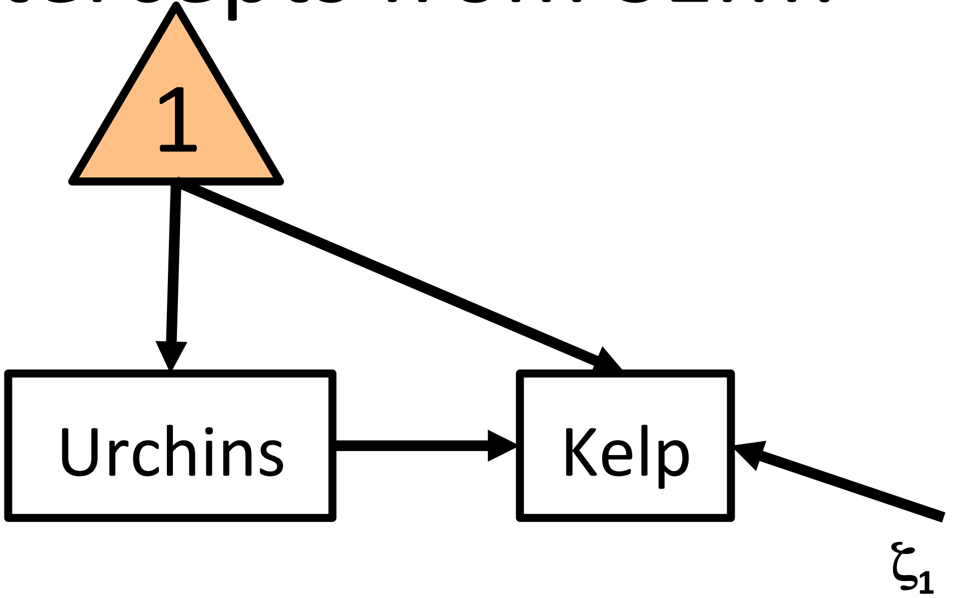
lower-level covariate

Means Structures: Acquiring Intercepts from SEM!



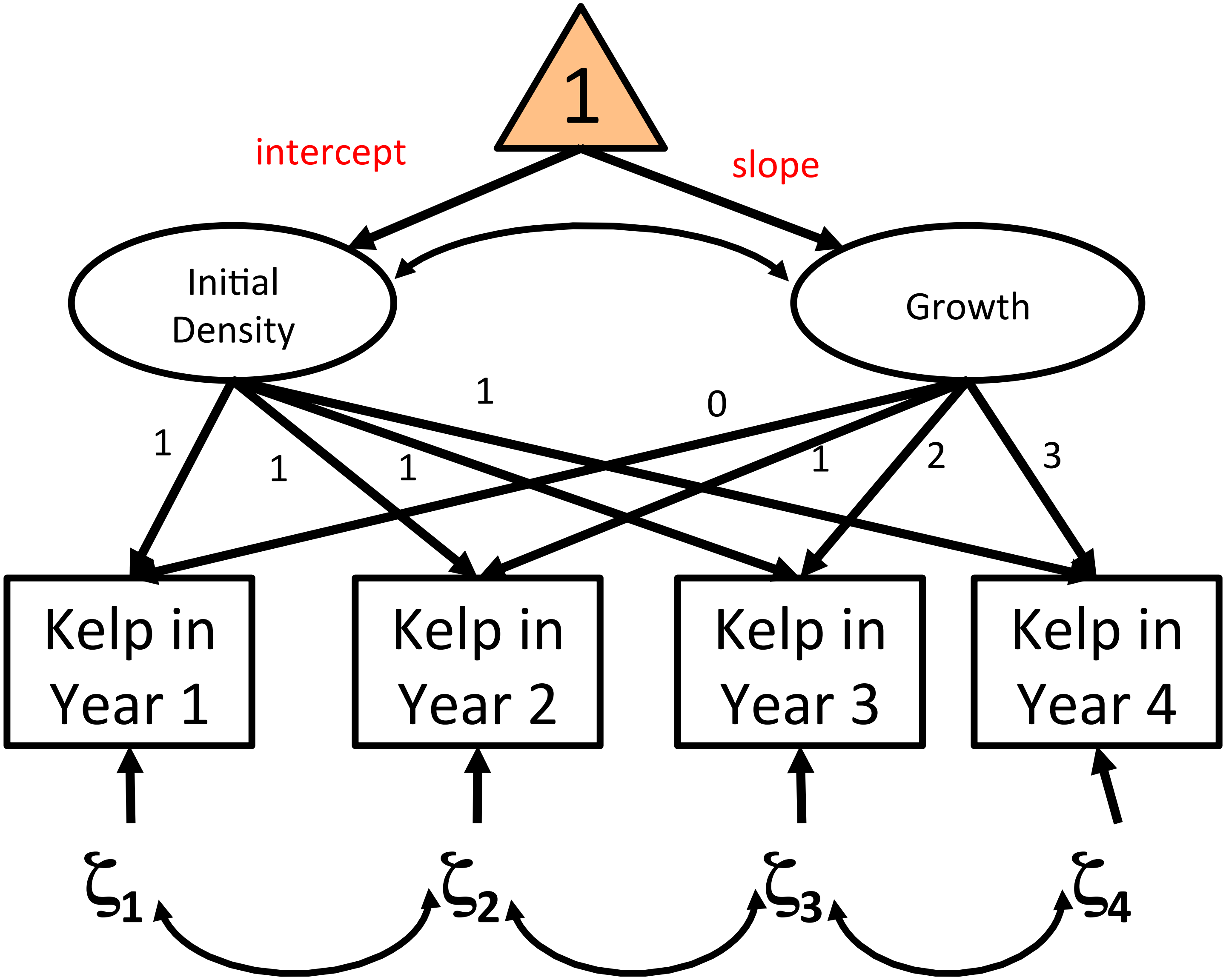
```
meanMod<-'Giant.Kelp ~ Purple.Urchins'
meanFit <- sem(meanMod, data=kfm, meanstructure=T)
```

Means Structures: Acquiring Intercepts from SEM!

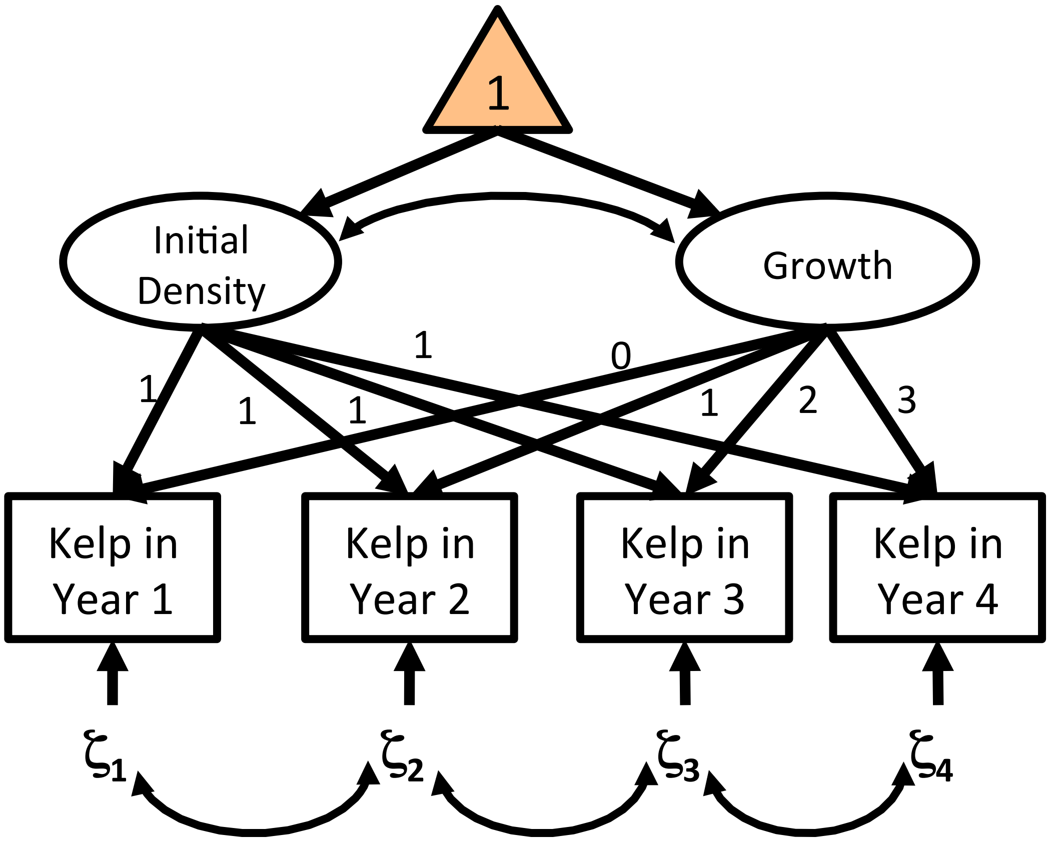


```
        Estimate  Std.err  Z-value  P(>|z|)
Regressions:
  Giant.Kelp ~
    Purple.Urchin    -0.366    0.029  -12.397    0.000

Intercepts:
    Giant.Kelp        1.590    0.076   20.791    0.000

Variances:
    Giant.Kelp        0.579    0.045   12.961    0.000
```

Latent Variable Growth Model

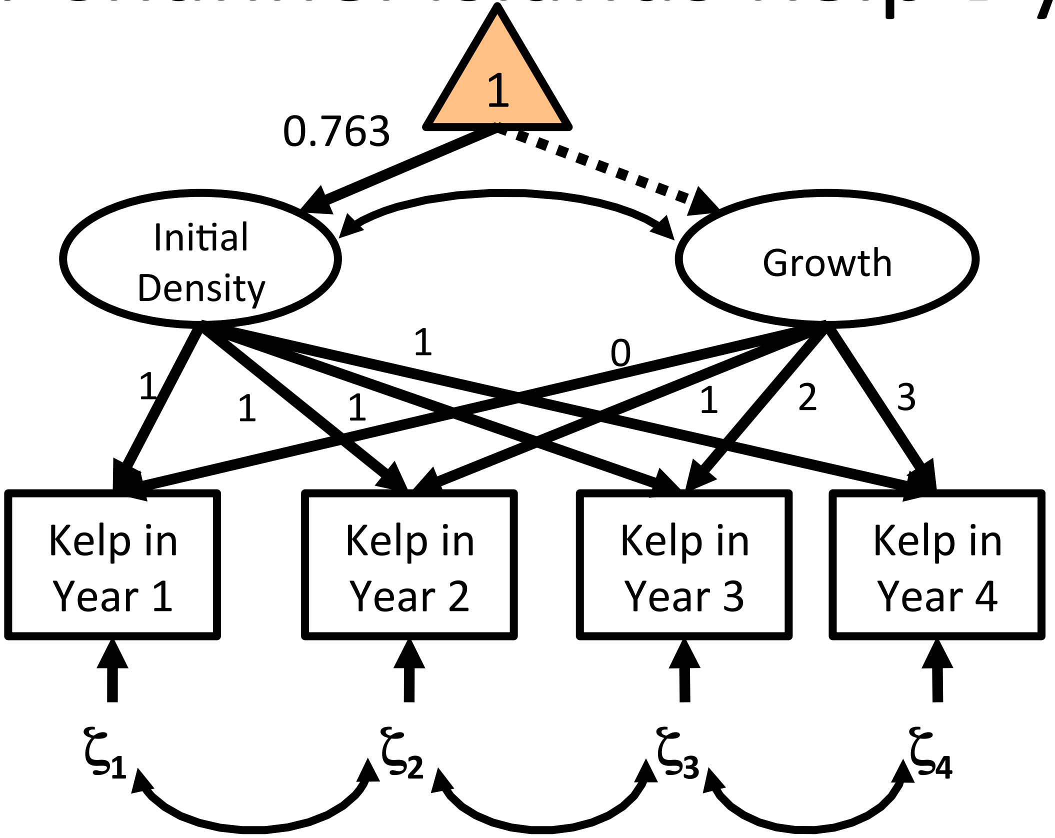


Example: Channel Islands Kelp Dynamics



```
gMod<-'
   Initial =~  1*KelpT1  + 1*KelpT2 + 1*KelpT3 + 1*KelpT4
   Growth =~ 0*KelpT1 + 1*KelpT2 + 2*KelpT3 + 3*KelpT4
   '

gFit<-growth(gMod, data=kelpTseries)
```

Example: Channel Islands Kelp Dynamics



R^2=0.5-0.67

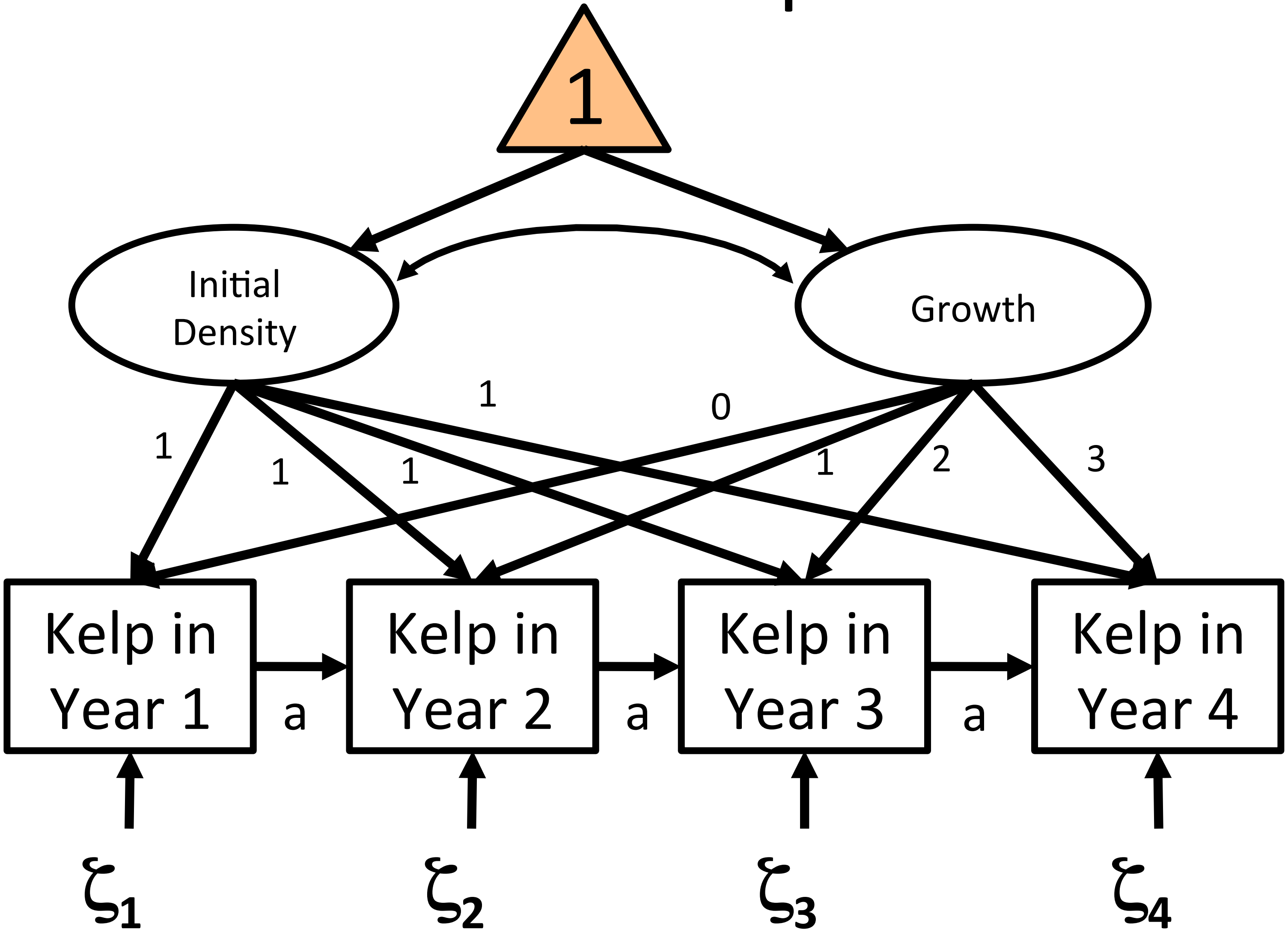
Conclusions:

At minimum, no linear trajectory.

At most, kelp densities stay constant with some small variation

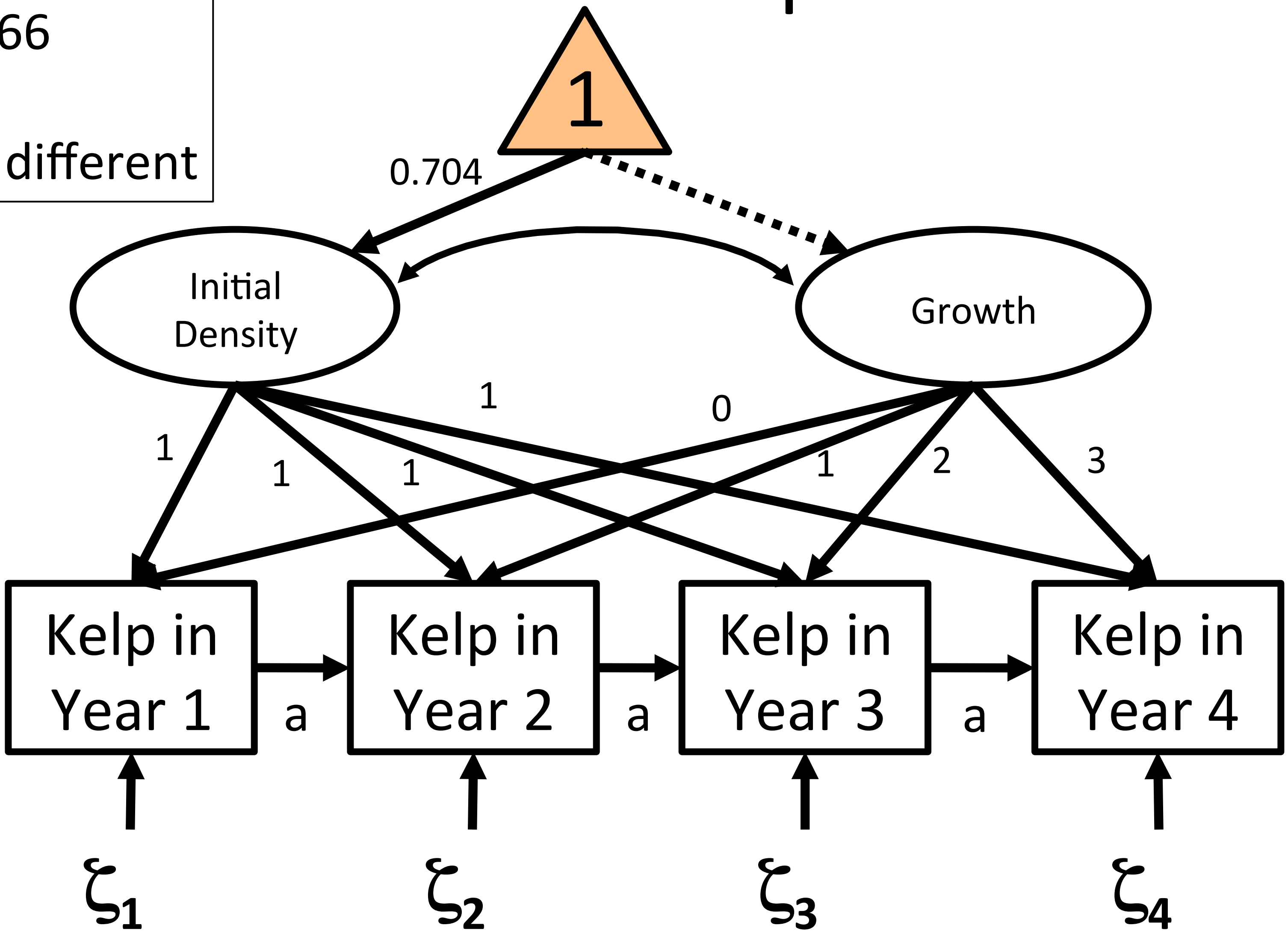
```
                   Estimate  Std.err  Z-value  P(>|z|)
Intercepts:
    KelpT1            0.000
    KelpT2            0.000
    KelpT3            0.000
    KelpT4            0.000
    Initial           0.763    0.096    7.976    0.000
    Growth            0.027    0.032    0.837    0.403
```

Growth Models and Autoregressive Relationship

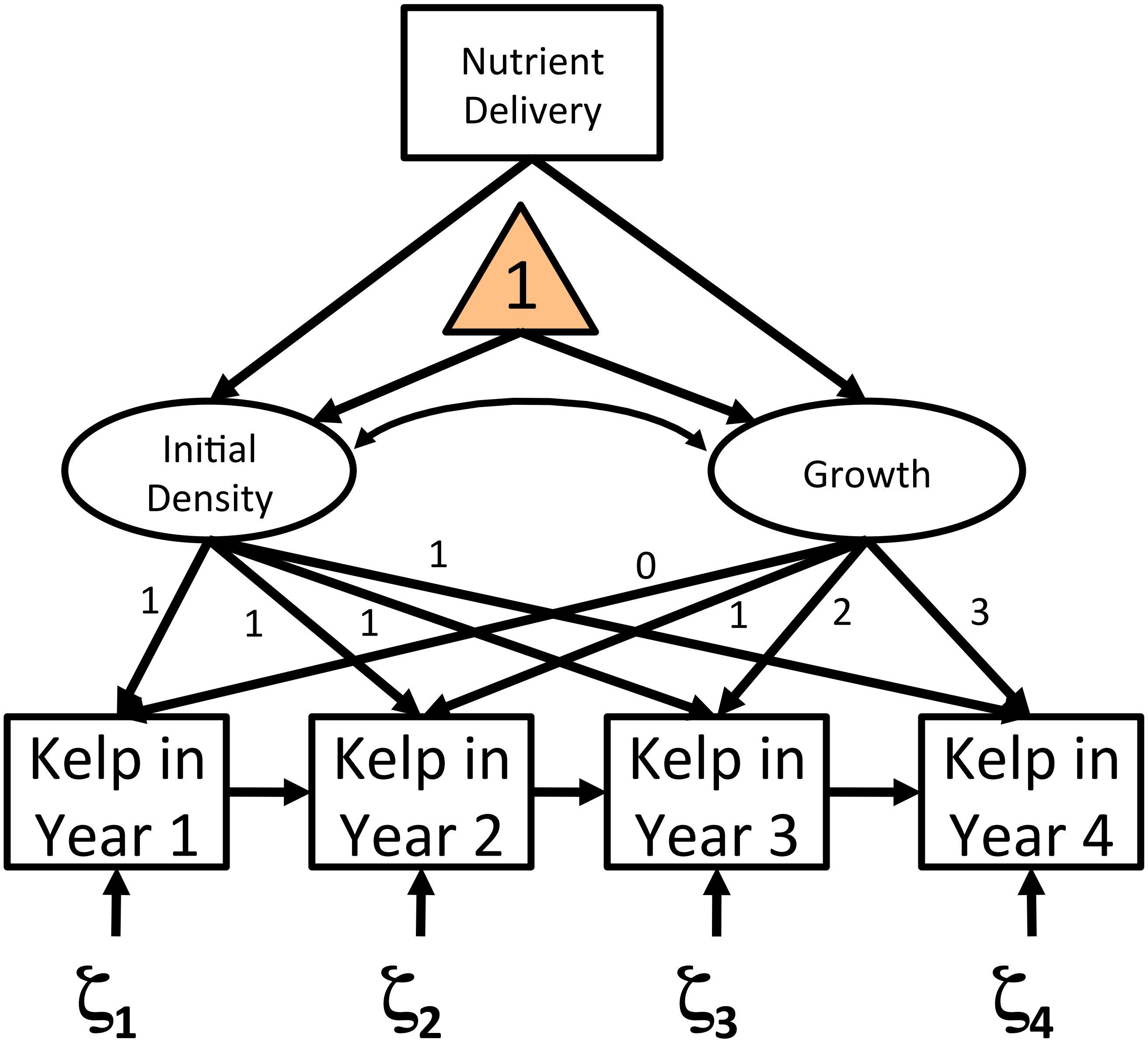


Growth Models and Autoregressive Relationship

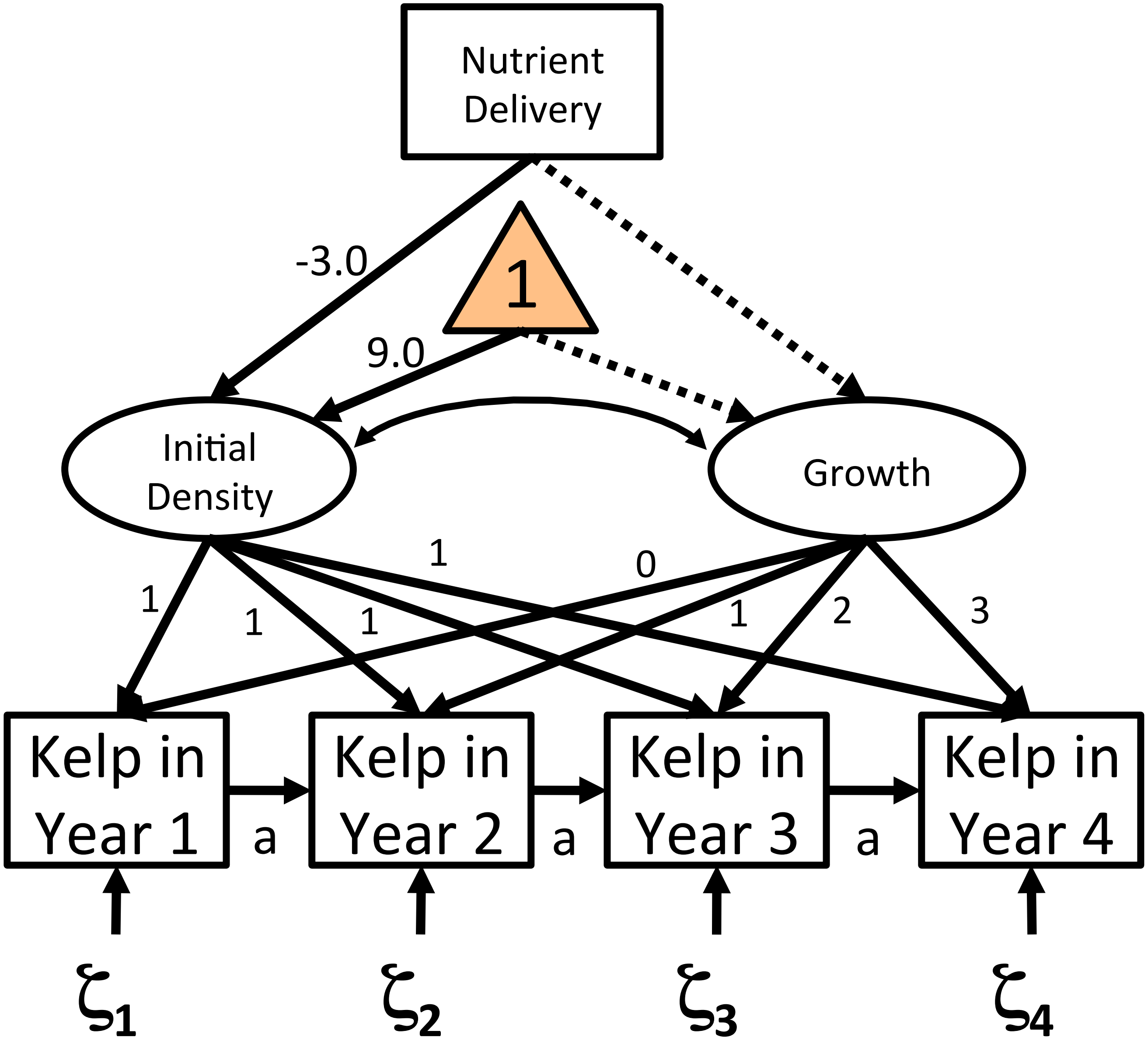
- a = 0.266
- Fit not different



Other Processes Affect Growth Curves



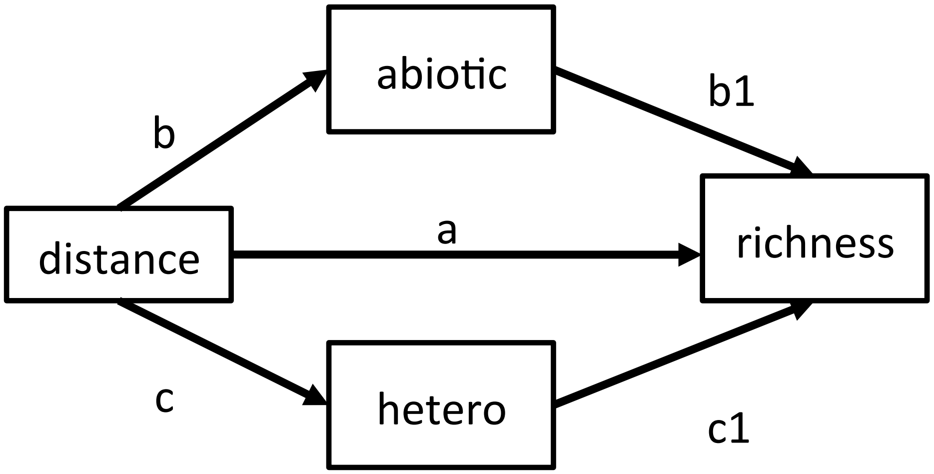
Other Processes Affect Growth Curves



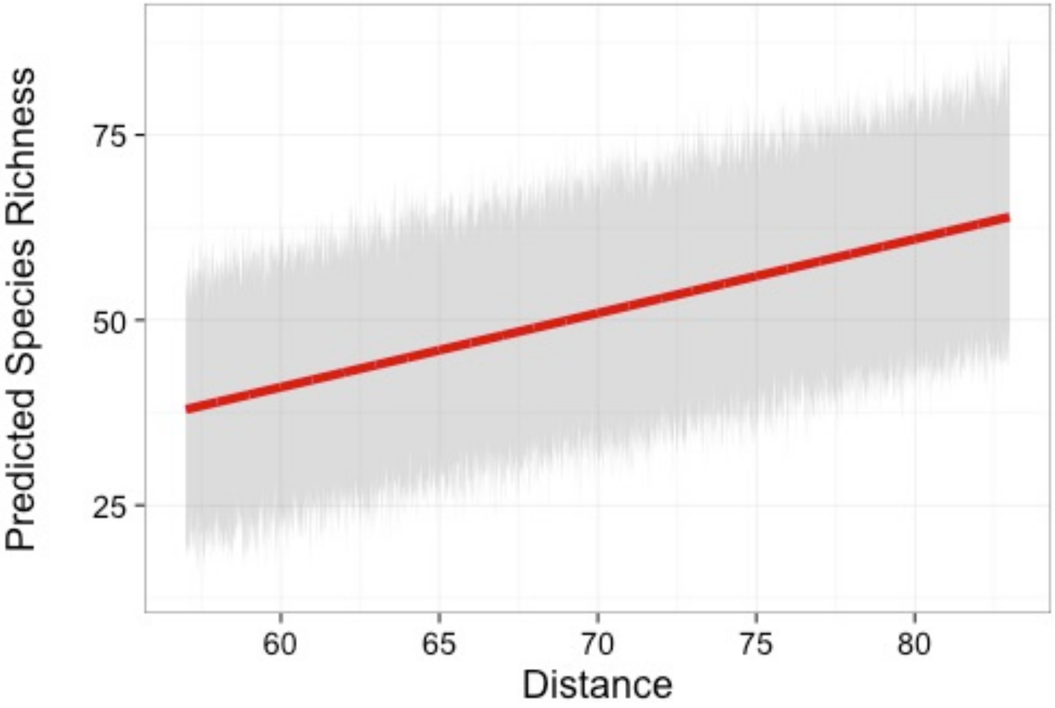
- a = 0.304

Final Comments on Advanced Topics

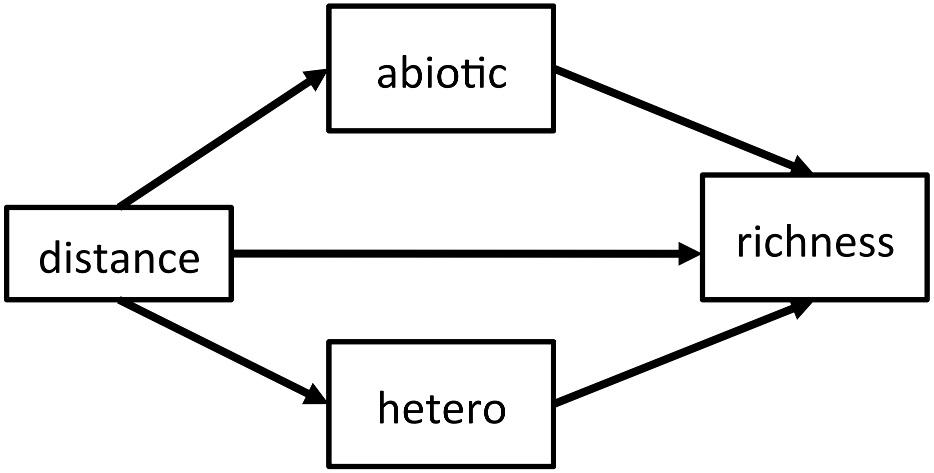
1. Often, our concern for spatial and temporal effects is due to our deep ecological fear of pseudoreplication.
2. If you can account for the drivers that create spatial or temporal blocks, you gain information.
3. Many cases are more easily dealt with in a piecewise approach.
4. But, many special cases have techniques in the literature that YOU can now use!



Prediction in SEM Using `lavaan`



If I have new Distance values, what is richness?



abiotic

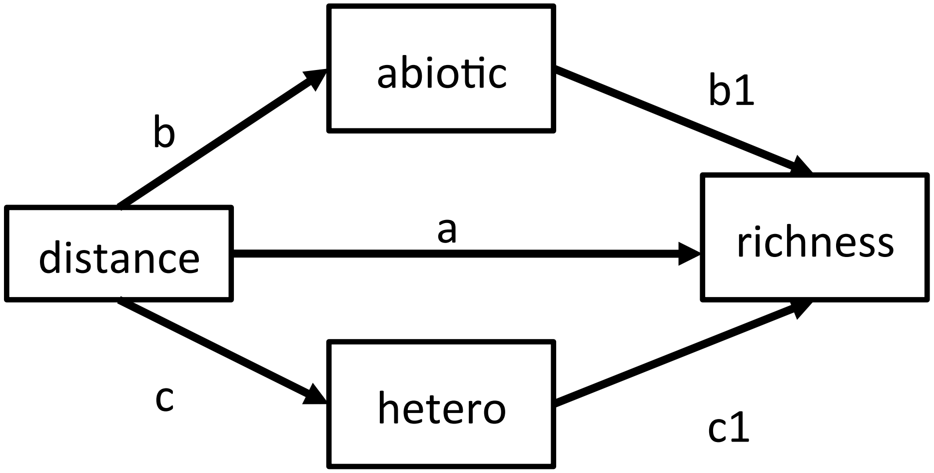
distance

richness

hetero

- Predict in lavaan merely gives factor loadings
- To get predicted values, we need to think about the underlying structural equations!

If I have new Distance values, what is richness?



Richness ~ a*distance + b1*b*distance + c1*c*distance

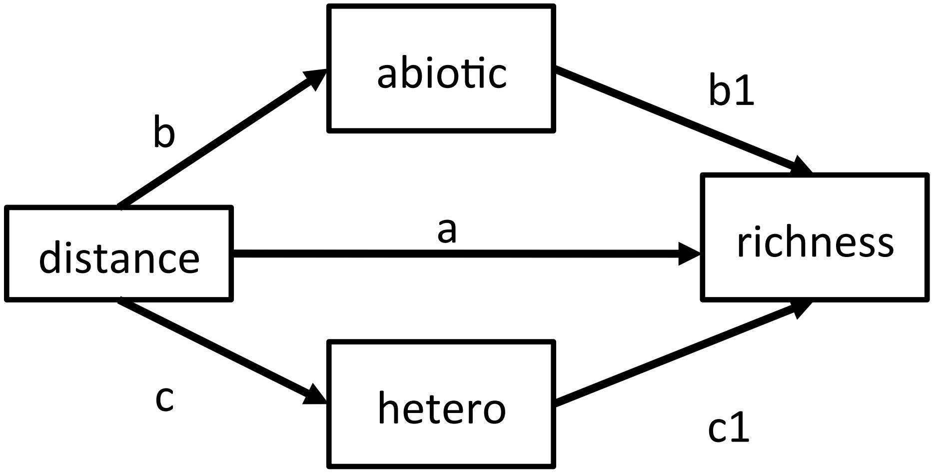
But what about intercepts?

Intercept = ia + ib*b1 + ic*c1

Estimating Direct and Indirect Effects with Named Coefficients

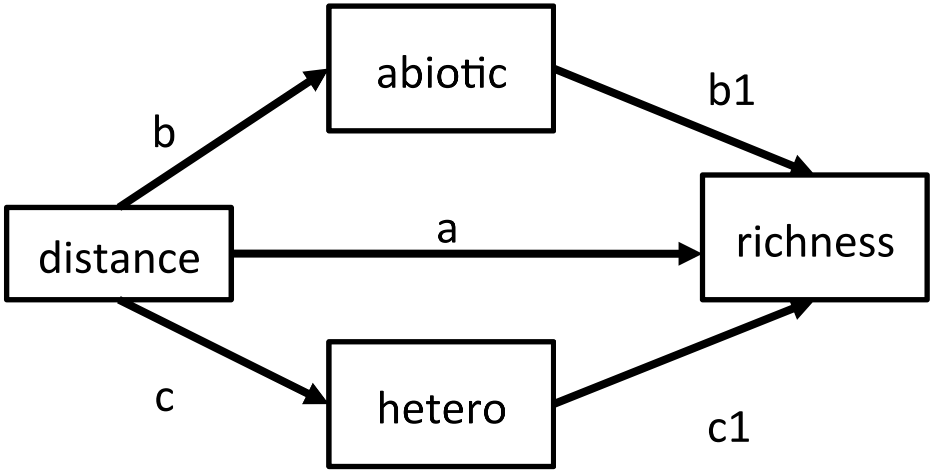
```
totDistModel <- '
rich ~ a*distance + b*abiotic + c*hetero
abiotic ~ b1*distance
hetero ~ c1*distance

rich ~ ia*1
abiotic ~ ib*1
hetero ~ ic*1

direct:=  a
indirect:= b1*b + c1*c
total:= b1*b + c1*c + a
totalIntercept := i1 + ib*b1 + ic*c1
'
```

Estimates of Compounded Effects



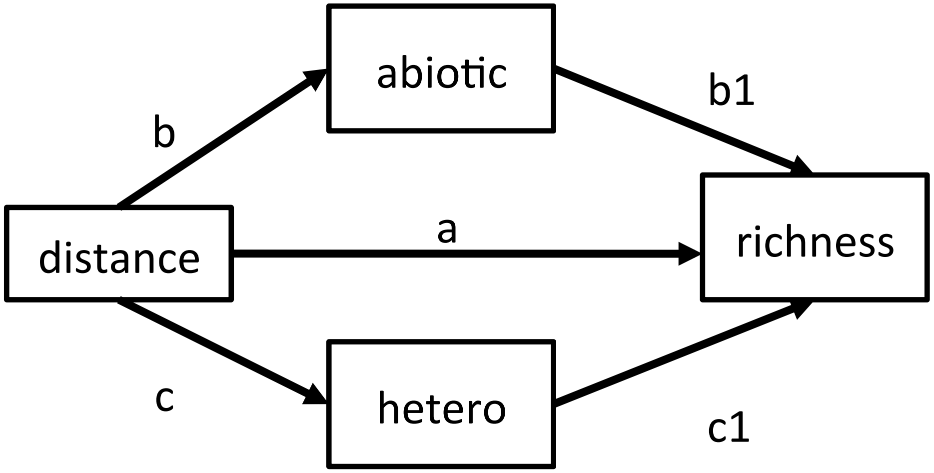
```
Defined Parameters:
                   Estimate  Std.Err  Z-value  P(>|z|)
    direct            0.640    0.156    4.117    0.000
    indirect          0.360    0.103    3.488    0.000
    total             1.000    0.145    6.918    0.000
    totalIntercept  -19.070    9.683   -1.969    0.049
```

Predictions with those Coefficients!



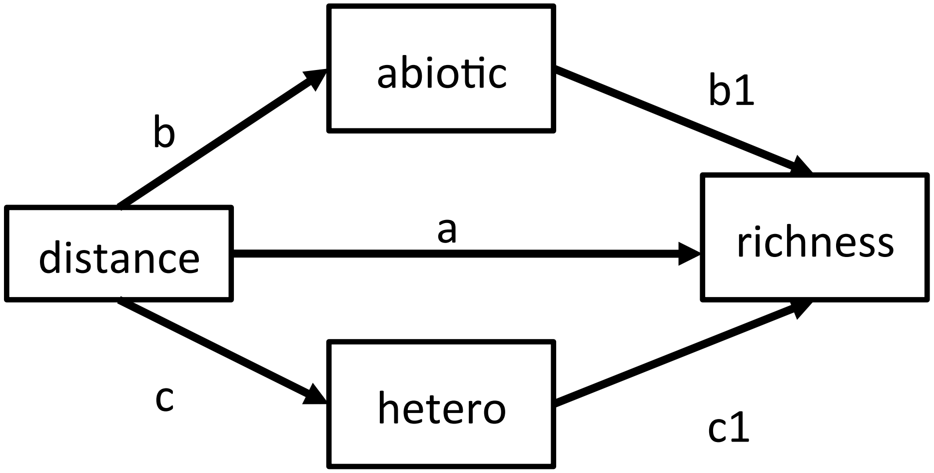
```
> #Get new predictions
> newDistance <- 57

> #naieve prediction
> newDistance*1 + -19.070

[1] 37.93
```

You have Compound Parameter Uncertainty



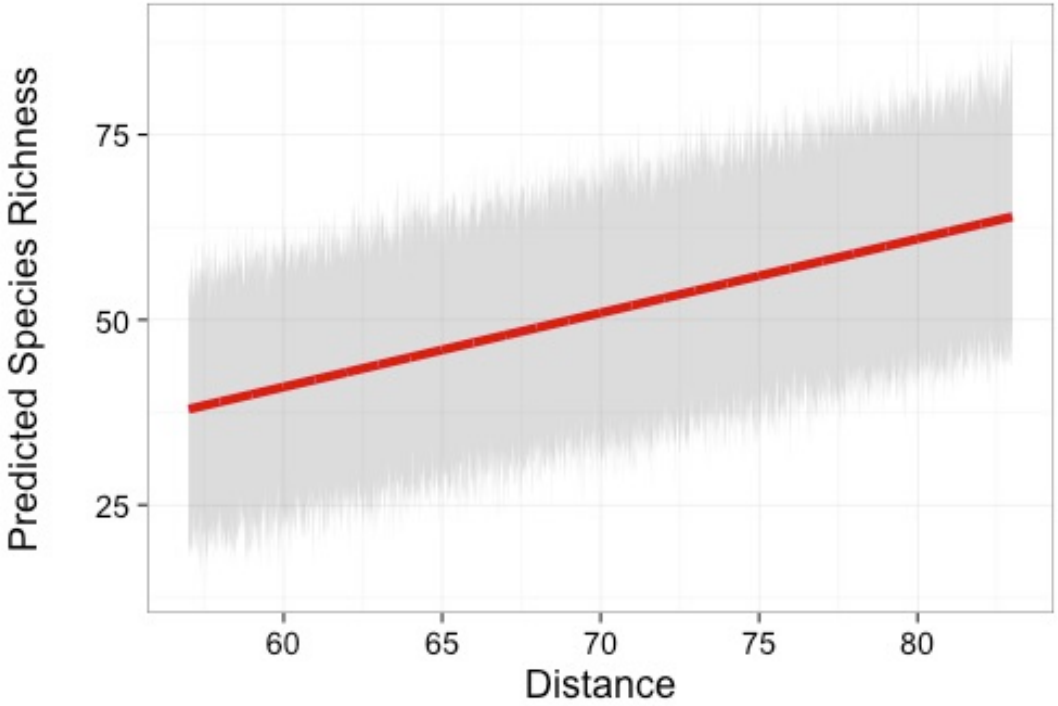
```
Defined Parameters:
                   Estimate  Std.Err  Z-value  P(>|z|)
    direct            0.640    0.156    4.117    0.000
    indirect          0.360    0.103    3.488    0.000
    total             1.000    0.145    6.918    0.000
    totalIntercept  -19.070    9.683   -1.969    0.049
```

You have Compound Parameter Uncertainty



```
> predDist <- rnorm(100, 1, 0.145)*newDistance +
   rnorm(100, -19.070, 9.683)


> quantile(predDist)
      0%       25%       50%       75%      100% 
10.58009 31.15239 39.55937 46.08567 66.19564
```

Creating Ranges of Predictions



```
new_data <- data.frame(distance = seq(57,83,.01))
new_data$predicted_richness <- new_data$distance*1 + -19.070

new_data <- cbind(new_data,
     t( sapply(new_data[,1], function(x)
        quantile( rnorm(100, 1, 0.145)*x +
                      rnorm(100, -19.070, 9.683),
                  probs=c(0.025, 0.975)))
      ))
```