Extensions to Local Estimation

Overview

1. Mixed effects models
2. Pseudo-R^2s
3. GLMM Example
4. GAM Example

1.1. Fixed vs. Random. Comparison

Fixed	Random
Interested in drawing inferences / making predictions	Not particularly interested in any particular value or level
Represent values from the entire 'universe' of interest	A (random) sample from a larger pool of potential values
Levels not interchangeable	Levels interchangeable (could swap / relabel levels without any change in meaning)
Directly manipulated	Introduces incidental error (e.g., between subjects, blocks, sites, etc.)
Few levels / worth sacrificing d.f. to fit model	Many levels / cannot sacrifice d.f. to fit model

1.1. Fixed vs. Random. From LM to LME

$$Y_i \sim X\beta + \epsilon$$

Grand mean effect

$$Y_i \sim X\beta + b_i + \epsilon$$

Group variation around grand mean

1.1. Fixed vs. Random. Why mixed models?

- More power than modeling the means of groups
- Reduces degrees of freedom necessary to fit model and estimate parameters (vs. modeling as a fixed effect)
- Accounts for uneven sampling within groups by using information across groups to inform the individual group means
- Can account for *non-independence* of observations by explicitly modeling their covariances (e.g., among sites, years, individuals, etc.)

1.1. Fixed vs. Random. Random structure

Different configurations of random structure:

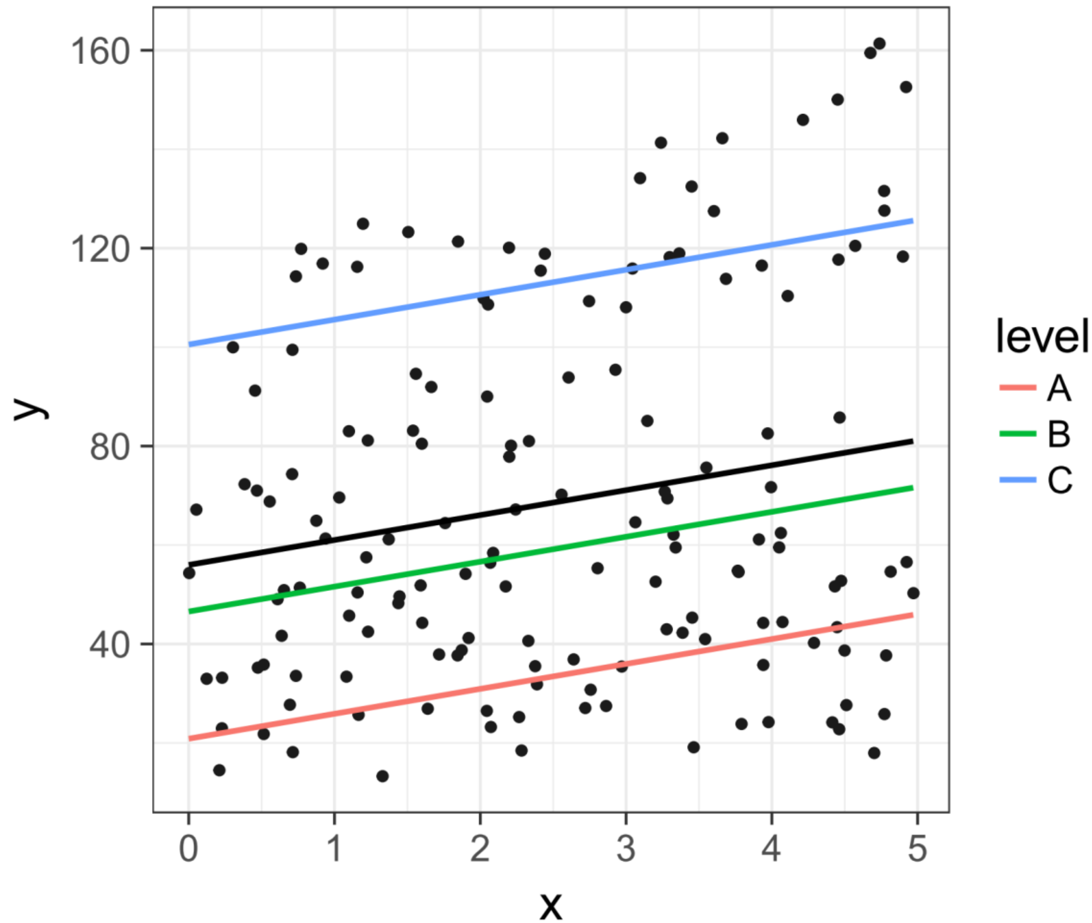
1. Varying intercept, fixed slope

2. Fixed intercept, varying slope

3. Varying intercept, varying slope

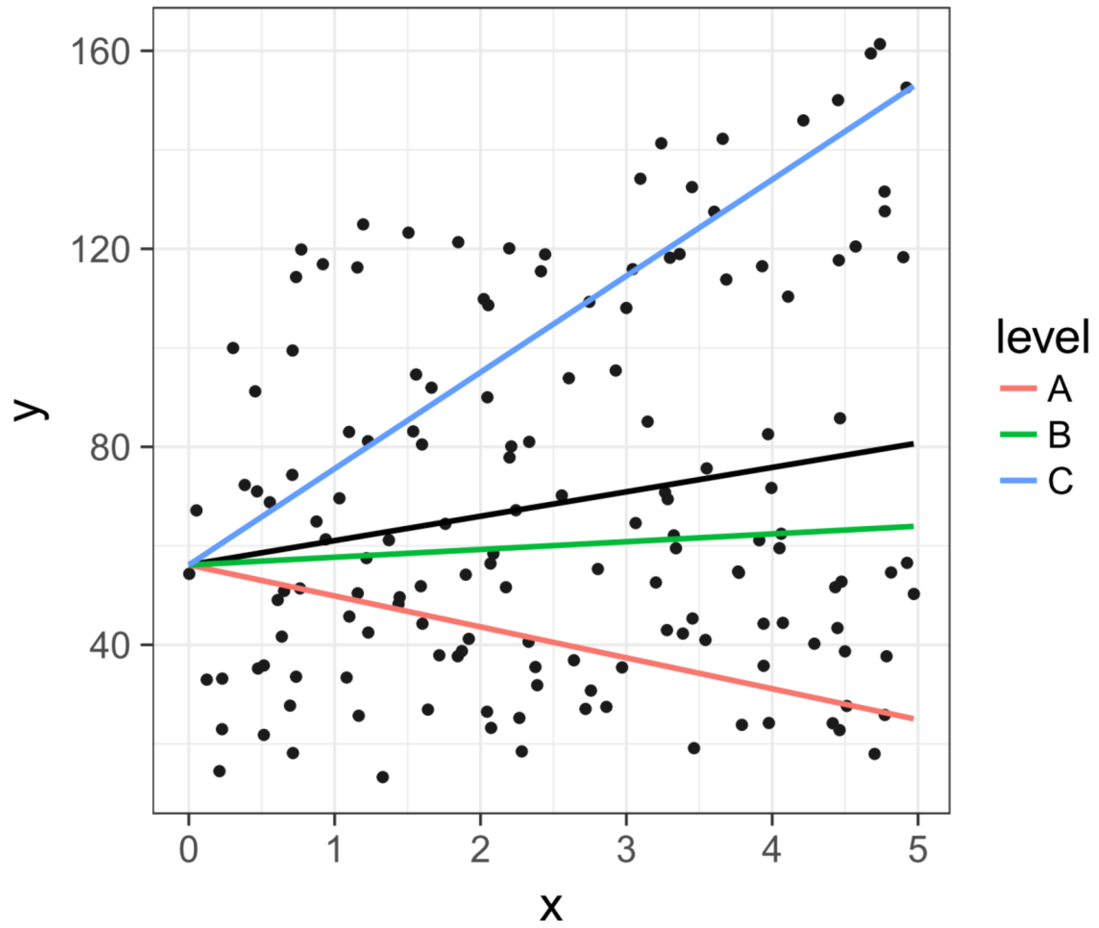
1.1. Fixed vs. Random. Varying intercept

- Estimates different intercept, same slope for all levels of the random effect (Good for block designs, repeated measures)



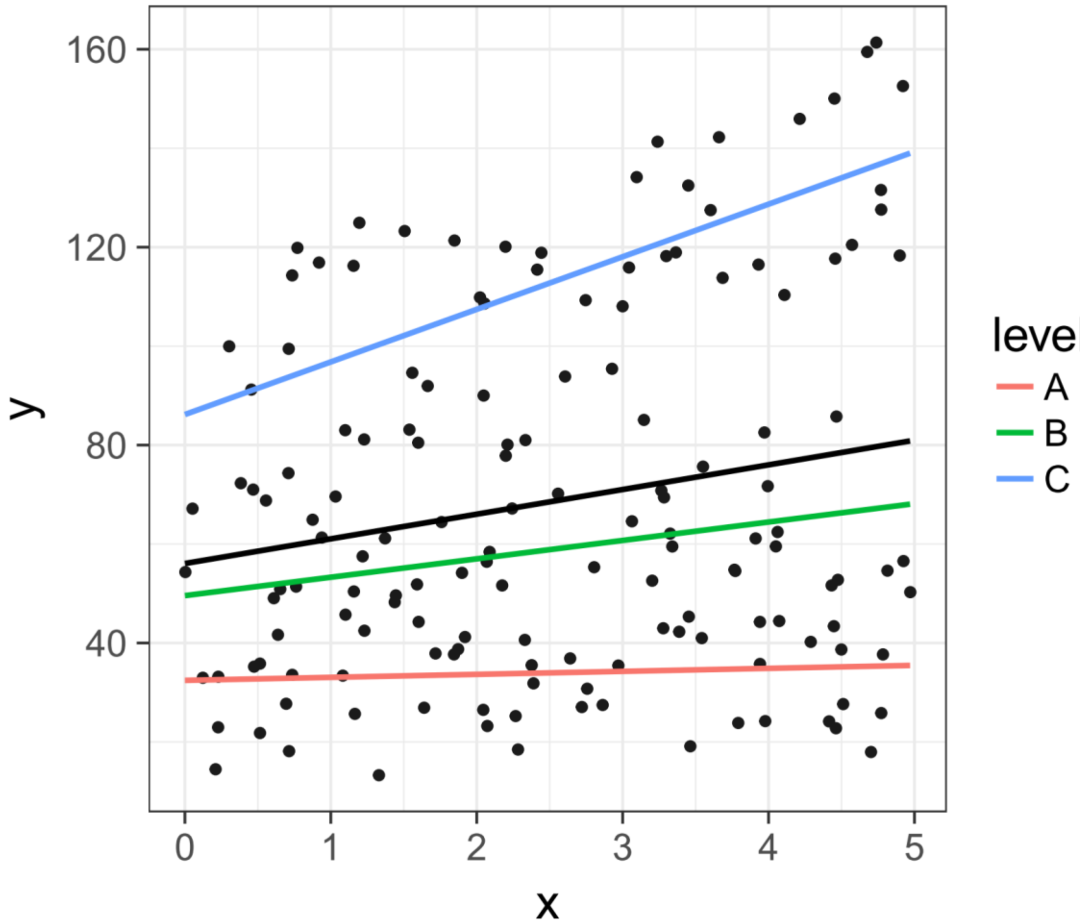
1.1. Fixed vs. Random. Varying slope

- Estimates different slope, same intercept for all levels



1.1. Fixed vs. Random. Varying intercept AND slope

- Estimates different slope, different intercept for all levels



1.1. Fixed vs. Random. Varying intercept AND slope

- Addresses multiple sources of non-independence of within and between levels, leading to lower Type I *and* Type II error
- Random slopes can be extracted and used in other analyses (lacks error)
- Computationally intensive, may lead to non-convergence

1.1. Fixed vs. Random. Nesting

- Hierarchical models represent nested random terms (e.g., site within region)

- Nesting further addresses non-independence by modeling correlations within *and* between levels of the hierarchy

- Good for stratified sampling designs (varying intercept) and split-plot designs (varying slope, varying intercept)

1.1. Fixed vs. Random. Random structures

<table>
<tr><td>(1|group)</td><td>random group intercept</td></tr>
<tr><td>(x|group) = (1+x|group)</td><td>random slope of x within group with correlated intercept</td></tr>
<tr><td>(0+x|group) = (-1+x|group)</td><td>random slope of x within group: no variation in intercept</td></tr>
<tr><td>(1|group) + (0+x|group)</td><td>uncorrelated random intercept and random slope within group</td></tr>
<tr><td>(1|site/block) = (1|site)+(1|site:block)</td><td>intercept varying among sites and among blocks within sites (nested random effects)</td></tr>
<tr><td>site+(1|site:block)</td><td>fixed effect of sites plus random variation in intercept among blocks within sites</td></tr>
<tr><td>(x|site/block) = (x|site)+(x|site:block) = (1 + x|site)+(1+x|site:block)</td><td>slope and intercept varying among sites and among blocks within sites</td></tr>
<tr><td>(x1|site)+(x2|block)</td><td>two different effects, varying at different levels</td></tr>
<tr><td>x*site+(x|site:block)</td><td>fixed effect variation of slope and intercept varying among sites and random variation of slope and intercept among blocks within sites</td></tr>
<tr><td>(1|group1)+(1|group2)</td><td>intercept varying among crossed random effects (e.g. site, year)</td></tr>
</table>

http://glmm.wikidot.com/faq

1.1. Fixed vs. Random. A warning

- Assumes fixed and random effects are *uncorrelated*
- If possible, fit random effects as fixed effects and compare parameter estimates of other predictors
- Need to ensure appropriate replication at *lowest* level of nested factors (5-6 levels, *minimum*) – otherwise, fit as fixed effects

1.1. Fixed vs. Random. Different distributions

- *lme4* can fit many kinds of different distributions using `glmer`

- Does not provide *P*-values (d.d.f uncertain, see: https://stat.ethz.ch/pipermail/r-help/2006-May/094769.html)

  - *piecewiseSEM* uses *pbkrtest* package which estimates d.d.f. using the Kenward-Rogers approximation (less finicky than *lmerTest*)
  - *piecewiseSEM* does this for you automatically using `coefs`

1.1. Fixed vs. Random. Different distributions

- *nlme* can only handle normal distributions
  - Ives (2015): "For testing the significance of regression coefficients, go ahead and log-transform count data"

- `glmmPQL` in the *MASS* package uses penalized quasi-likelihood to fit models, can incorporate many different distributions and their quasi- equivalents (e.g., quasi-Poisson)
  - Quasi-distributions estimate a separate term for how the variance scales with the mean, so ideal for over/under-dispersed data
  - Quasi-likelihood means no likelihood based statistics (e.g., AIC, LRT, etc.) for any models fit with `glmmPQL`

1.1. Fixed vs. Random. Testing significance

- No matter what reviewers insist, <u>you cannot test significance of random effects</u>

- If you want to assess significance, model them as fixed effects

- Alternatives:
  - Drop random effects and compare to mixed model using AIC/BIC
  - Examine variance components using `varcomp`
    - If they are sufficiently large relative to residual variance probably worth keeping them in
  - Compare conditional and marginal R^2s
  - Defend yourself philosophically: these are known sources of variation, why not account for them, even if they don’t contribute, better safe than sorry!

1.1. Fixed vs. Random. Testing significance

- R has the most infuriating error messages
- Can sometimes solve by switching to a different optimizer
    - `lmeControl(opt = “optim”)` usually works
- Reduce tolerance for convergence
    - `lmeControl(tol = 1e-4)`
- Respecify random structure
    - Optimizer constrained to have cov > 0, can sometimes get stuck bouncing around when random components are very close to 0
- https://stackexchange.com/

1.2. Pseudo-R^2s

1.2. Pseudo-R^2s. Omnibus test

- Fisher's C / χ^2 is the global fit statistic for local estimation but has many shortcomings:
  - Sensitive to the number of d-sep tests and the complexity of the model (harder to reject as the complexity increases)
  - Sensitive to the size of the dataset (e.g., high *n* leads to low *P*)
  - Fails symmetricity when dealing with unlinked non-normal intermediate variables
  - Cannot be computed for saturated models

1.2. Pseudo-R^2s. Local tests

- How do we infer the confidence in our SEM?
  - Examine standard errors of individual paths, qualitatively assess cumulative precision
  - Explore variance explained (i.e., R^2), qualitatively assess cumulative precision

1.2. Pseudo-R^2s. General linear regression

- Coefficient of determination (R^2) = proportion of variance in response explained by fixed effects
- For OLS regression, simply 1- the ratio of unexplained (error) variance (e.g., SS_{error}) over the total explained variance (e.g., SS_{total})
- Ranges (0, 1), independent of sample size
- Not good for model comparisons since R^2 monotonically increases with model complexity (go to AIC which is penalized for complexity)

1.2. Pseudo-R^2s. Generalized linear regression

- Likelihood estimation is not attempting to minimize variance but instead obtain parameters that maximize the likelihood of having observed the data

- In a likelihood framework, equivalent R^2 = 1- the ratio of the log-likelihood of the full model over the log-likelihood of the null (intercept-only) model

- Leads to identical R^2 as OLS for normal (Gaussian) distributions, not so for GLM – need to use likelihood-based pseudo-R^2 (e.g., McFadden, Nagelkerke)

1.2. Pseudo-R^2s. Generalized mixed models

- Becomes even worse for mixed models because variance is partitioned among levels of the random factor, so what is the error variance?

- Need a new formulation of R^2 :

  - Marginal R^2 = variance explained by fixed effects only

Fixed effects variance

$$R^2_{\mathrm{GLMM}(m)} = \frac{\sigma_f^2}{\sigma_f^2 + \sum_{l=1}^{u} \sigma_l^2 + \sigma_e^2 + \sigma_d^2}$$

Fixed effects variance

Random effects variance

Residual variance

Distribution-specific variance

1.2. Pseudo-R^2s. Generalized mixed models

- Conditional R^2 = variance explained by both the fixed and random effects

Fixed effects variance

Random effects variance

$$R^2_{\mathrm{GLMM}(c)} = \frac{\sigma_f^2 + \sum_{l=1}^{u} \sigma_l^2}{\sigma_f^2 + \sum_{l=1}^{u} \sigma_l^2 + \sigma_e^2 + \sigma_d^2}$$

Fixed effects variance

Residual variance

Random effects variance

Distribution-specific variance

1.2. Pseudo-R^2s. Generalized mixed models

- Comparison of marginal and conditional R^2 can lead to roundabout assessment of ‘significance’ of the random effects (e.g., if conditional R^2 is larger relative to marginal R^2)

- Best to report both and allow readers to determine how their magnitude affects the inferences

1.3. GLMM Example

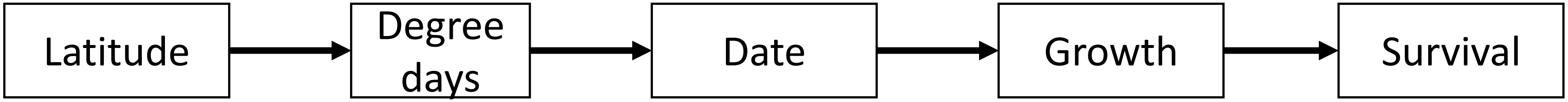
1.3. SEM Example. Shipley 2009

- Hypothetical dataset: predicting latitude effect on survival of a tree species
- Repeated measures on 5 subjects at 20 sites from 1970-2006
- Survival (0/1) influenced by phenology (degree days until bud break, Julian days until bud break), size (stem diameter growth)

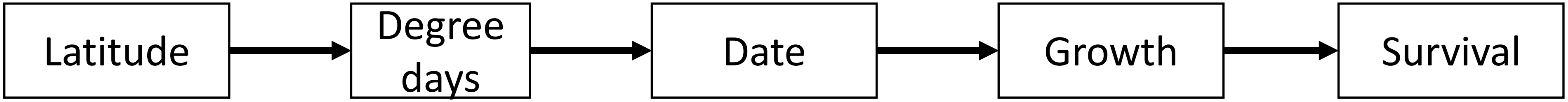
1.3. SEM Example. Shipley 2009

- Two distributions: normal, binary (survival)

- Random effects:
  - Site-only: latitude
  - Site and year: degree days, date
  - Site, year, and subject: diameter, survival

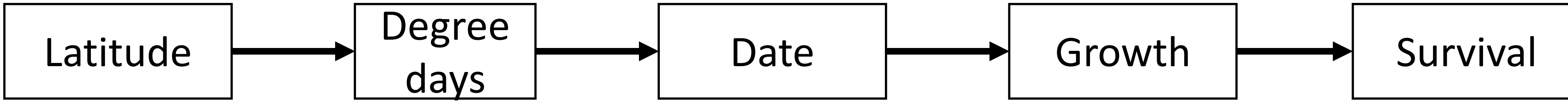


1.3. SEM Example. What is the basis set?



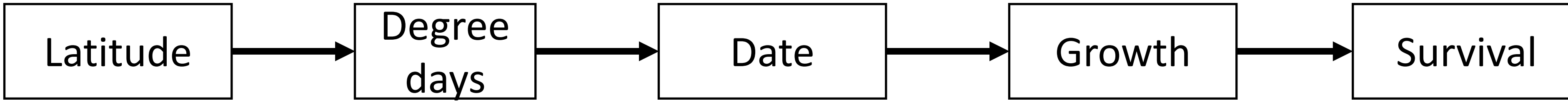
- Date $\perp$ Lat | (Degree days)
- Growth $\perp$ Lat | (Date)
- Survival $\perp$ Lat | (Growth)
- Growth $\perp$ Degree days | (Date, Lat)
- Survival $\perp$ Degree days | (Growth, Lat)
- Survival $\perp$ Date | (Growth, Degree days)

1.3. SEM Example. List of equations



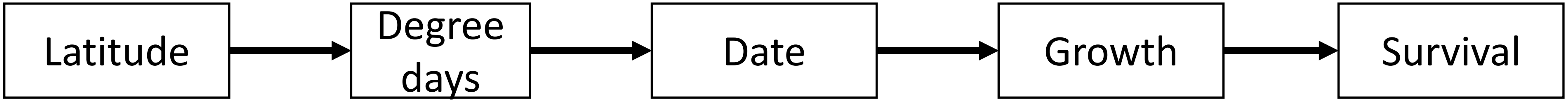

```
library(piecewiseSEM)
library(nlme)
library(lme4)

# Load data
data(shipley); shipley <- na.omit(shipley)

# Create list of structural equations
shipley.sem <- psem(
  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)
)
```

1.3. SEM Example. D-sep tests



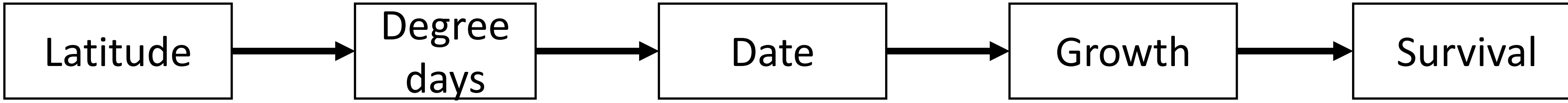

```
# Get summary
summary(shipley.sem)

Structural Equation Model of shipley.sem

Call:
  DD ~ lat
  Date ~ DD
  Growth ~ Date
  Live ~ Growth

    AIC
 21745.782

---
```

1.3. SEM Example. D-sep tests



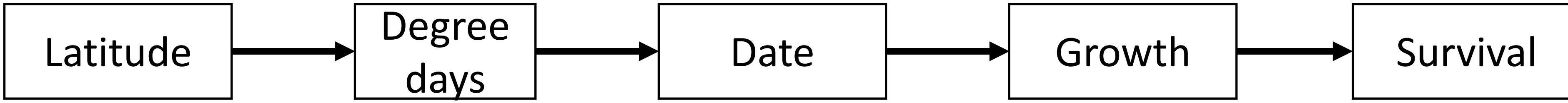

```
Tests of directed separation:

     Independ.Claim Test.Type   DF Crit.Value P.Value
   Date ~ lat + ...      coef   18    -0.0798  0.9373
 Growth ~ lat + ...      coef   18    -0.8929  0.3837
   Live ~ lat + ...      coef 1431     1.0280  0.3039
  Growth ~ DD + ...      coef 1329    -0.2967  0.7667
    Live ~ DD + ...      coef 1431     1.0046  0.3151
  Live ~ Date + ...      coef 1431    -1.5617  0.1184

--
Global goodness-of-fit:

Chi-Squared = NA with P-value = NA and on 6 degrees of freedom
Fisher's C = 11.536 with P-value = 0.484 and on 12 degrees of freedom

---

Warning message:
Check model convergence: log-likelihood estimates lead to negative Chi-squared!
```

1.3. SEM Example. D-sep tests

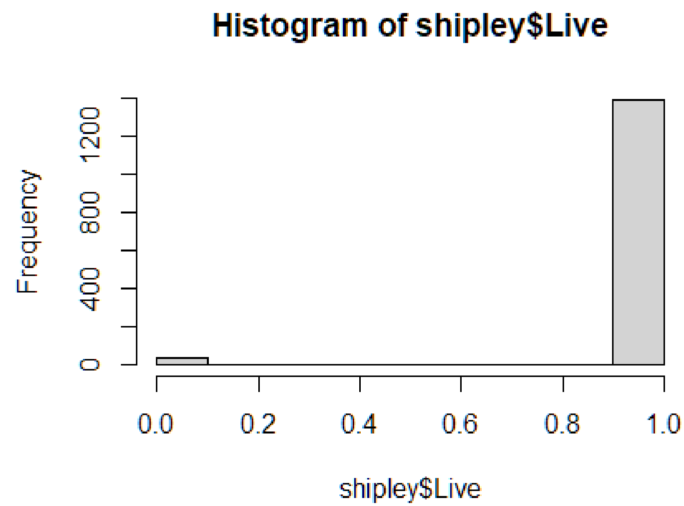



```
# Look at problematic model & variance components
Live.model <- glmer(Live ~ Growth + Date + DD + lat + (1|site) +
(1|tree), family = binomial(link = "logit"), data = shipley)
boundary (singular) fit: see ?isSingular

VarCorr(Live.model)
 Groups Name        Std.Dev.
 tree   (Intercept) 0
 site   (Intercept) 0
```

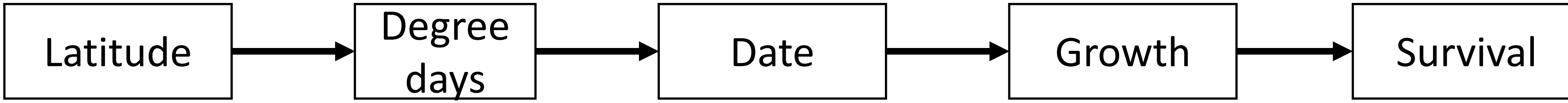
1.3. SEM Example. D-sep tests



- Re-specify random structure
- Still no positive χ^2 statistic ☹
- Consider other distributions (e.g., negative binomial)
- <u>Revert to d-sep test</u>

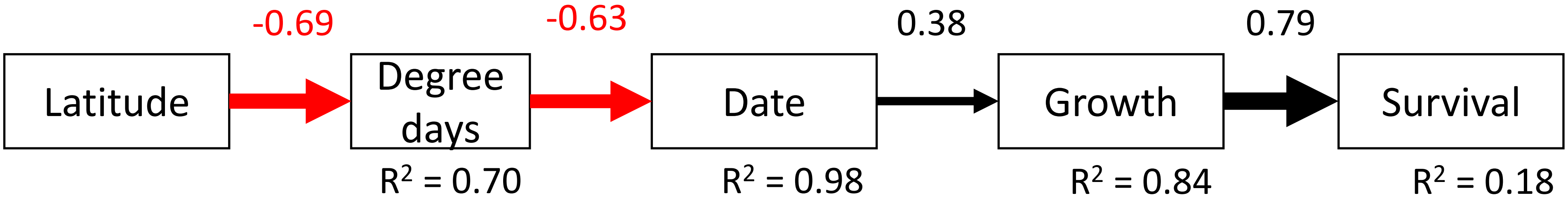


1.3. SEM Example. D-sep tests



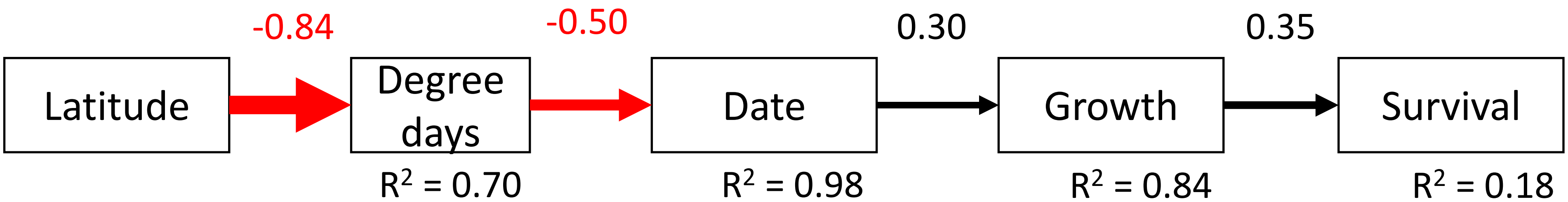

```
Coefficients:

  Response Predictor Estimate Std.Error   DF Crit.Value P.Value Std.Estimate
        DD       lat  -0.8355    0.1194   18    -6.9960       0      -0.6877 ***
      Date        DD  -0.4976    0.0049 1330  -100.8757       0      -0.6281 ***
    Growth      Date   0.3007    0.0266 1330    11.2.917       0        0.3824 ***
      Live    Growth   0.3479    0.0584 1431     5.9552       0       0.7866 ***

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

---
Individual R-squared:

  Response method Marginal Conditional
        DD   none     0.49        0.70
      Date   none     0.41        0.98
    Growth   none     0.11        0.84
      Live  delta     0.16        0.18
```

1.3. SEM Example. Populate final model



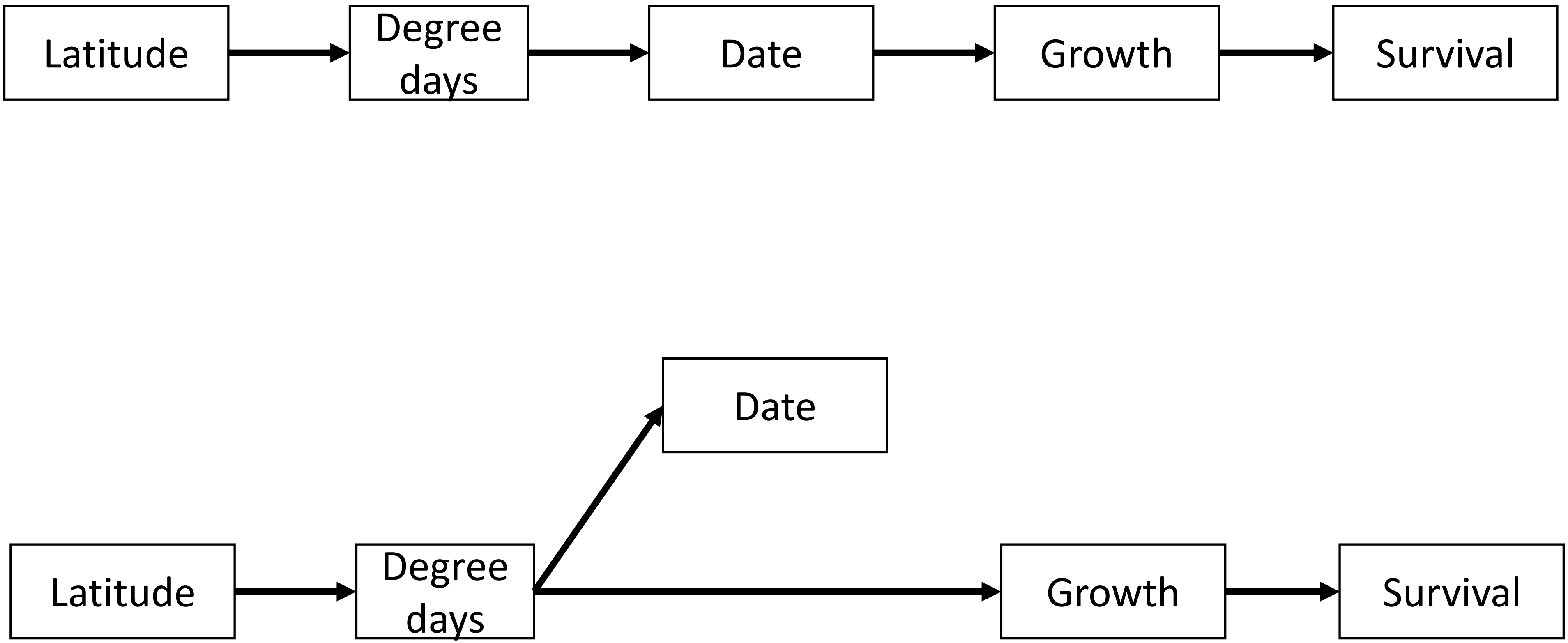

```
Coefficients:

  Response Predictor Estimate Std.Error   DF Crit.Value P.Value Std.Estimate
        DD       lat  -0.8355    0.1194   18    -6.9960       0      -0.6877 ***
      Date        DD  -0.4976    0.0049 1330  -100.8757       0      -0.6281 ***
    Growth      Date   0.3007    0.0266 1330    11.2.917        0       0.3824 ***
      Live    Growth   0.3479    0.0584 1431     5.9552       0       0.7866 ***

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

---
Individual R-squared:

  Response method Marginal Conditional
        DD   none     0.49        0.70
      Date   none     0.41        0.98
    Growth   none     0.11        0.84
      Live  delta     0.16        0.18
```

1.3. SEM Example. Refit using *lavaan*



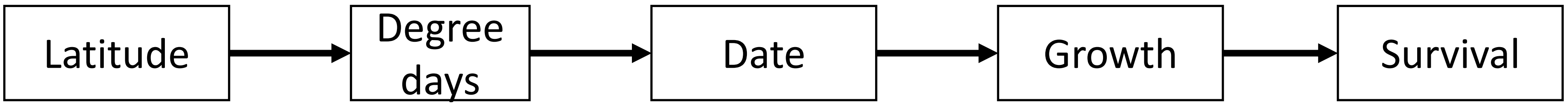

```
...

  Estimator                                         ML
  Model Fit Test Statistic                      38.433
  Degrees of freedom                                 6
  P-value (Chi-square)                           0.000

...

Regressions:
                   Estimate  Std.Err  z-value  P(>|z|)
  DD ~
    lat              -0.860    0.023  -37.923    0.000
  Date ~
    DD               -0.517    0.016  -32.525    0.000
  Growth ~
    Date              0.173    0.020    8.508    0.000
  Live ~
    Growth            0.006    0.001    9.854    0.000
```

1.3. SEM Example. Compare these models

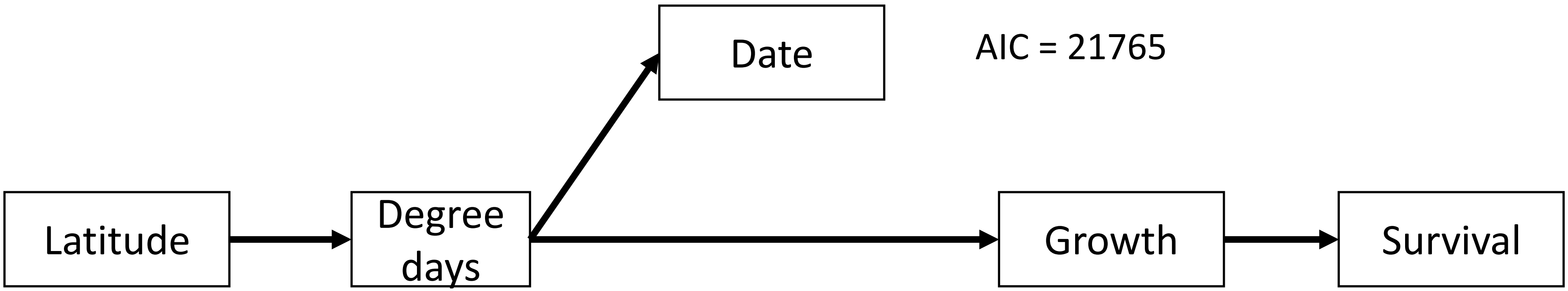


1.3. SEM Example. Compare these models

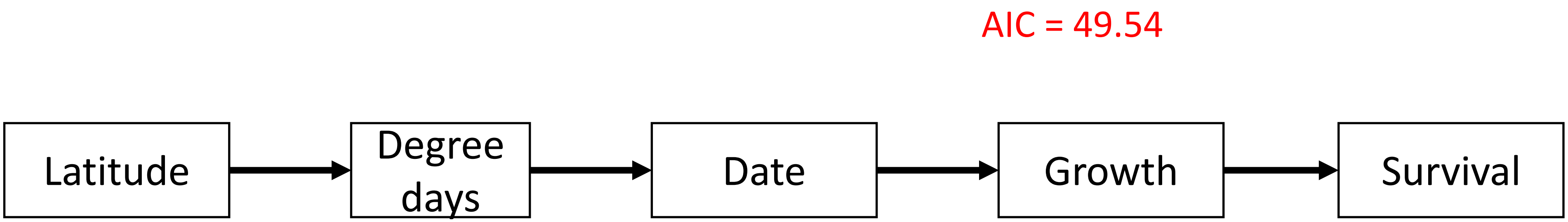
AIC = 21745

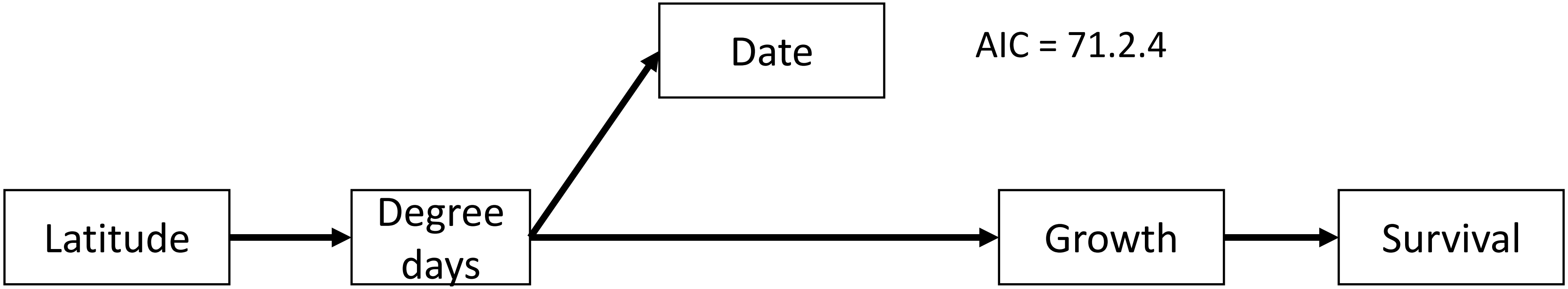


AIC = 21765

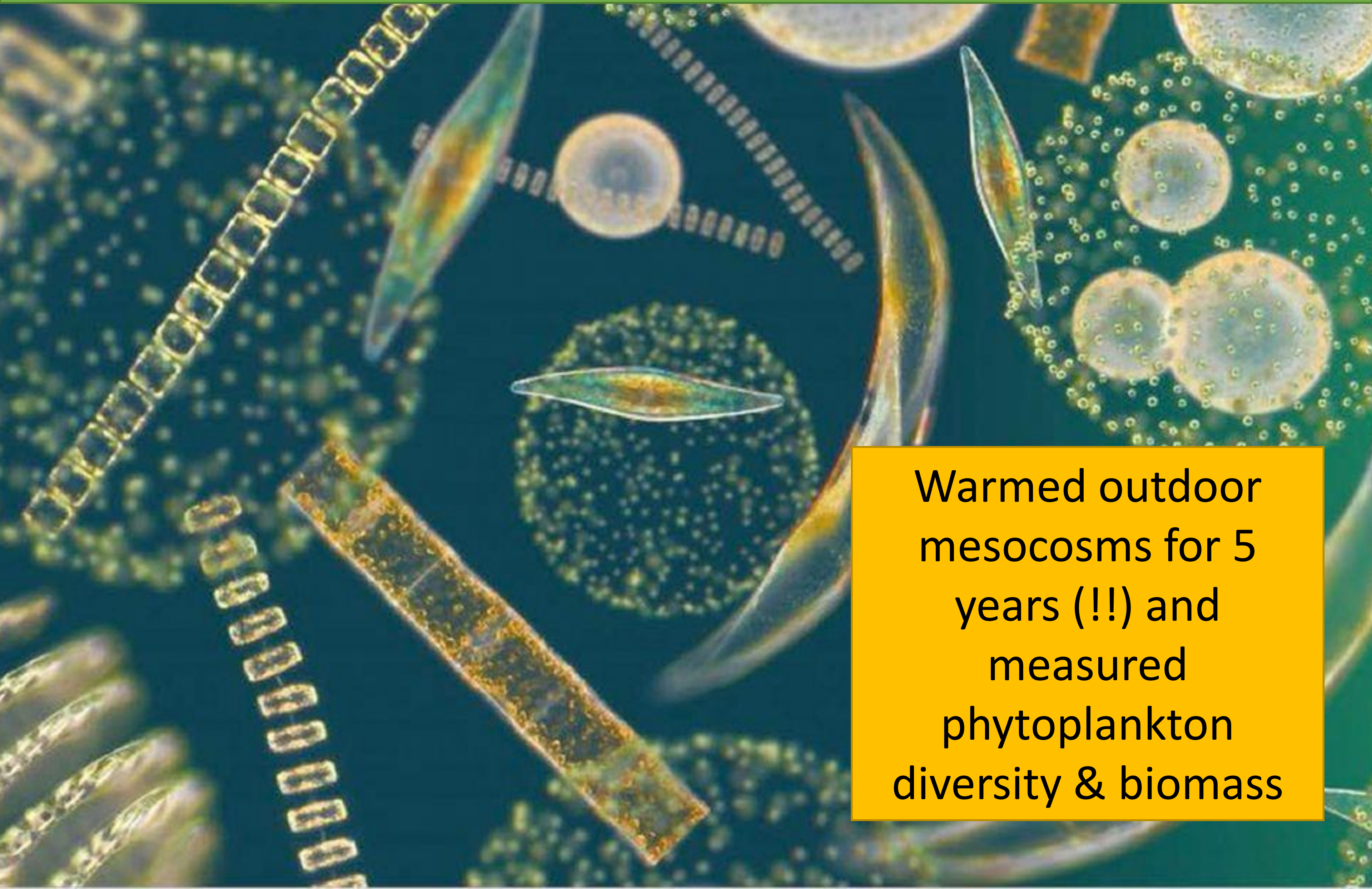


1.3. SEM Example. Compare these models (d-sep)



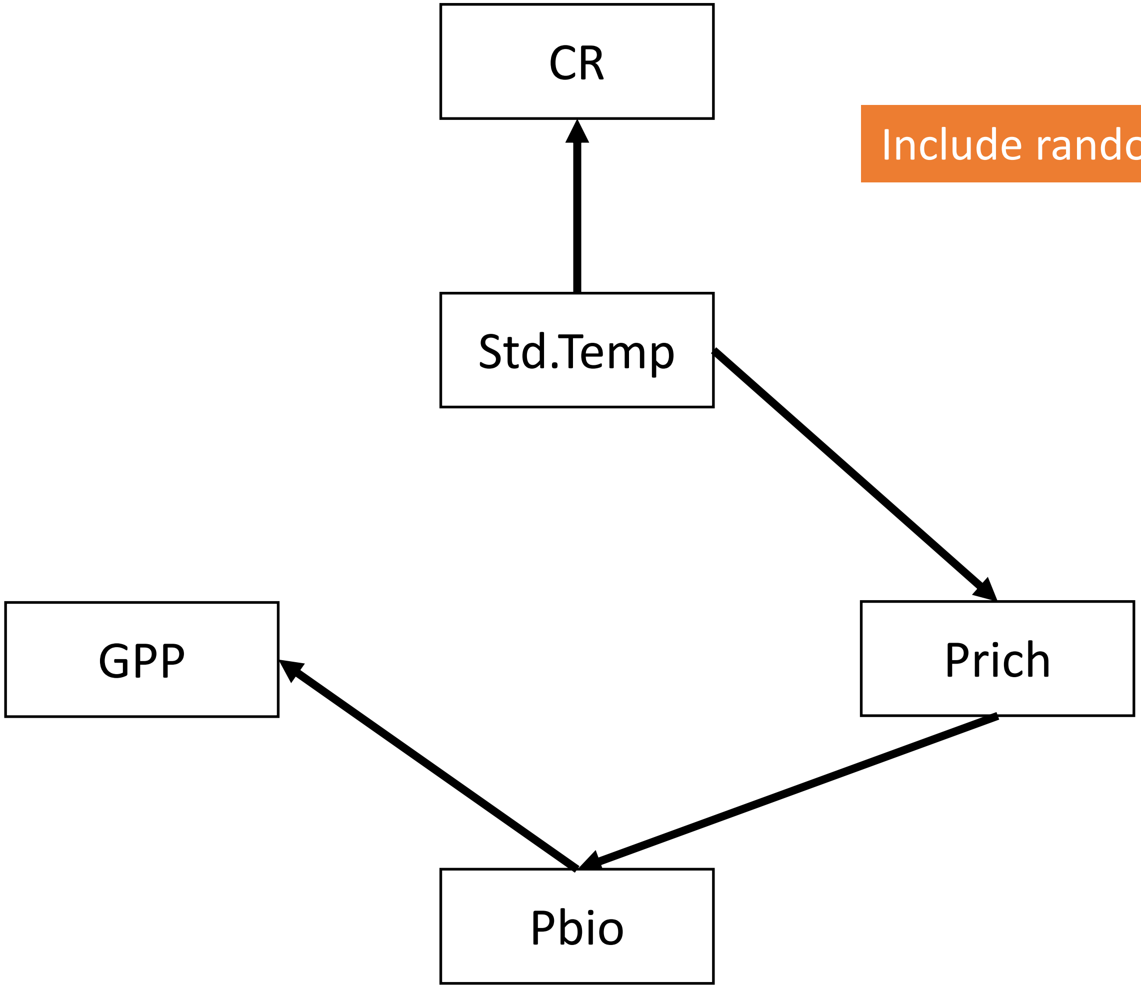


Yvon-Durocher et al (2015): Experimental warming on phytoplankton diversity and biomass



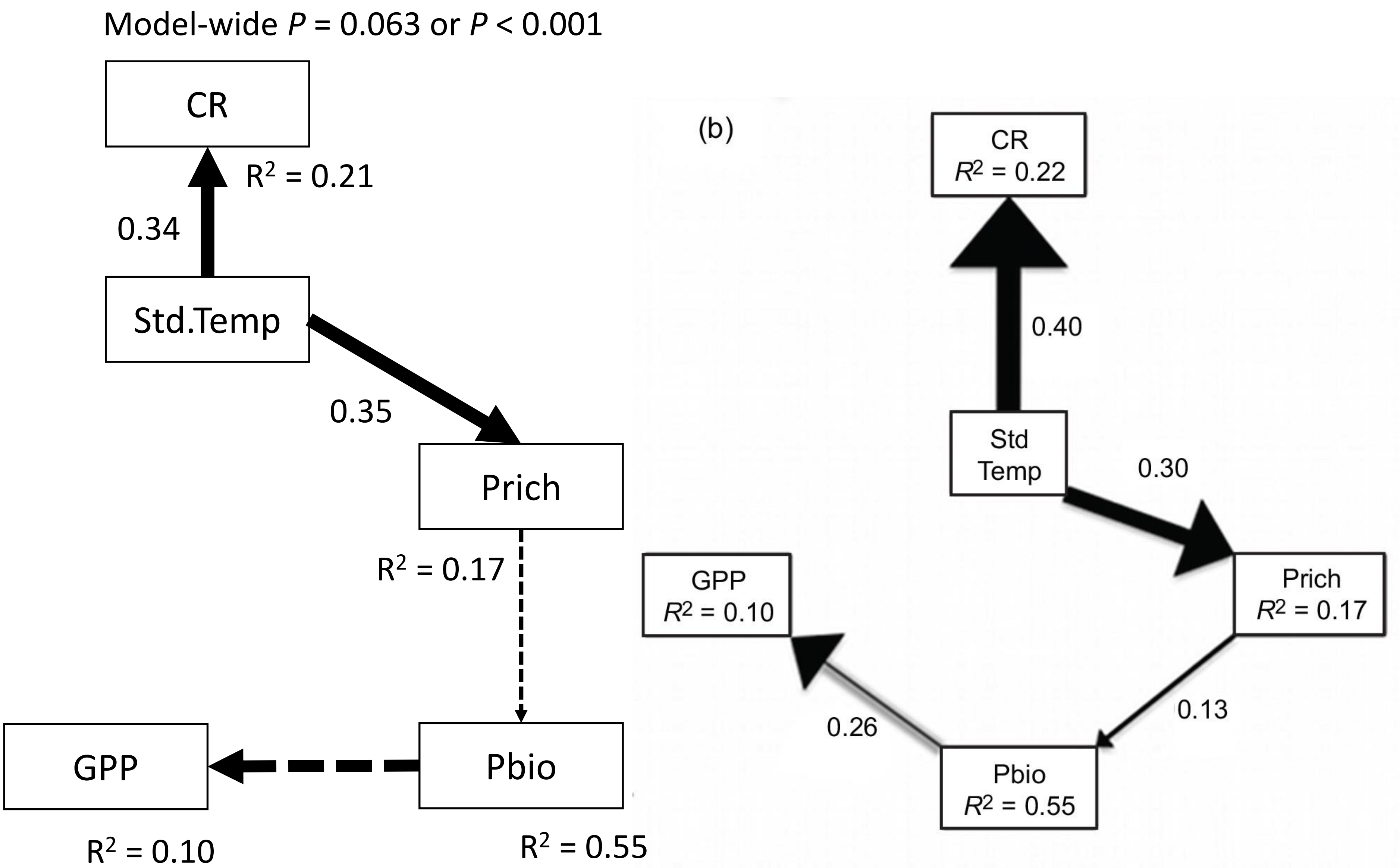
Warmed outdoor mesocosms for 5 years (!!) and measured phytoplankton diversity & biomass

ACTIVITY. Fit Durocher dataset



Include random effect of Pond.ID!

1.3. SEM Example. Your turn...



1.3. SEM Example. Your turn…

- Try removing incomplete cases first: `complete.cases`
  - What is their mistake here?

- Methods state: "with multiple measurements of variables made seasonally, nested within replicate mesocosms," but then, "a path model as a set of hierarchical linear mixed effects models, each of which included hypothesized relationships between a response variable and a set of predictors as fixed effects and mesocosm ID as a random effect on the intercept."
  - Play with the random structure?

- What about by treatment (Ambient vs. Heated)?

- Can anyone reproduce this result? Is it time to write a response?

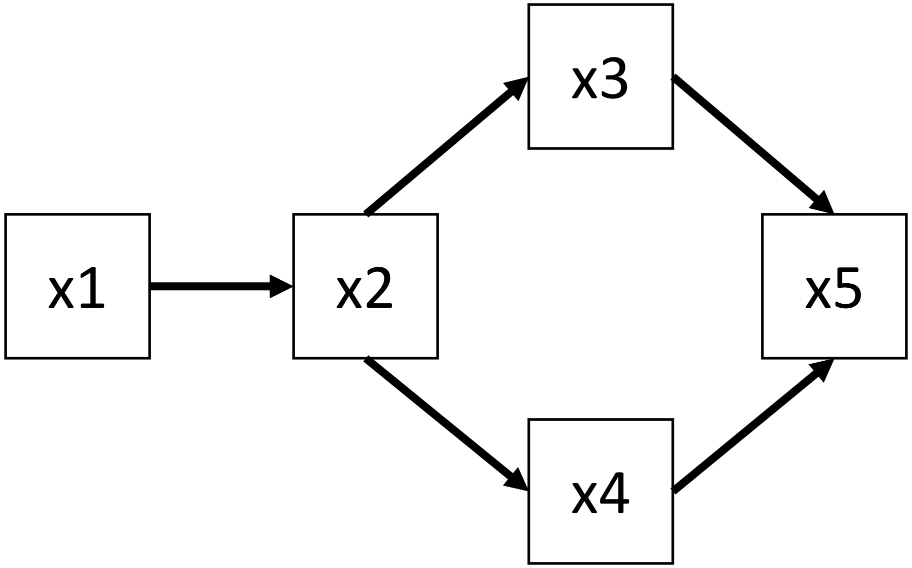
1.4. GAM Example

1.4. Generate example data

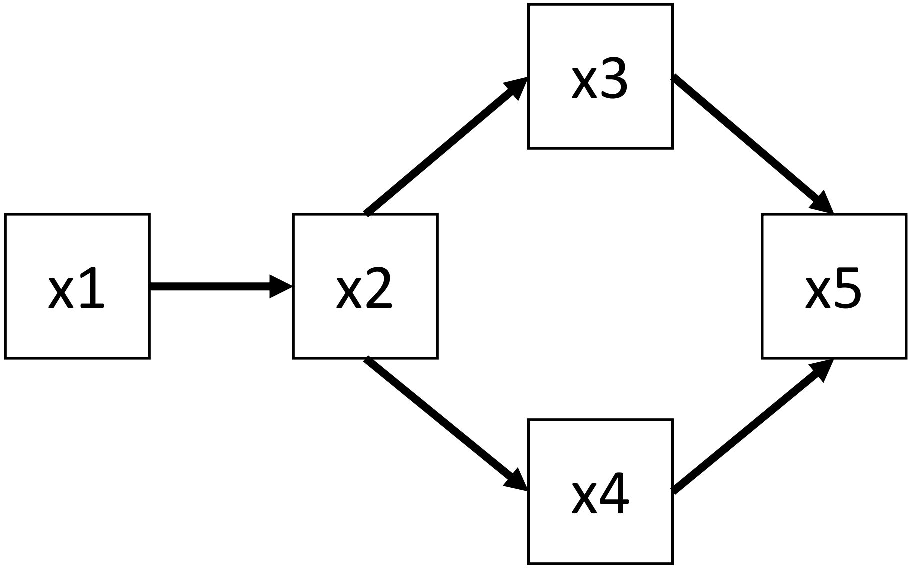
- Example data from appendix of Shipley and Douma using a mix of non-normal and non-linear variables

```
# Generate data from paper
set.seed(100)
n <- 100
x1 <- rchisq(n, 7)
mu2 <- 10*x1/(5 + x1)
x2 <- rnorm(n, mu2, 1)
x2[x2 <= 0] <- 0.1
x3 <- rpois(n, lambda = (0.5*x2))
x4 <- rpois(n, lambda = (0.5*x2))
p.x5 <- exp(-0.5*x3 + 0.5*x4)/(1 + exp(-0.5*x3 + 0.5*x4))
x5 <- rbinom(n, size = 1, prob = p.x5)
dat2 <- data.frame(x1 = x1, x2 = x2, x3 = x3, x4 = x4, x5 = x5)
```

1.4. Fit this SEM using `lm` and get GoF

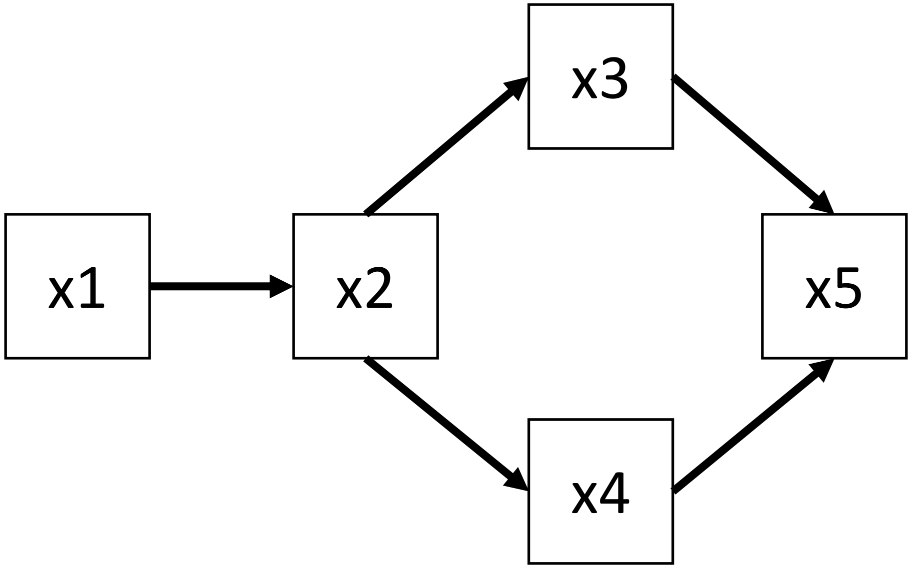


1.4. Fit this SEM using `lm` and get GoF



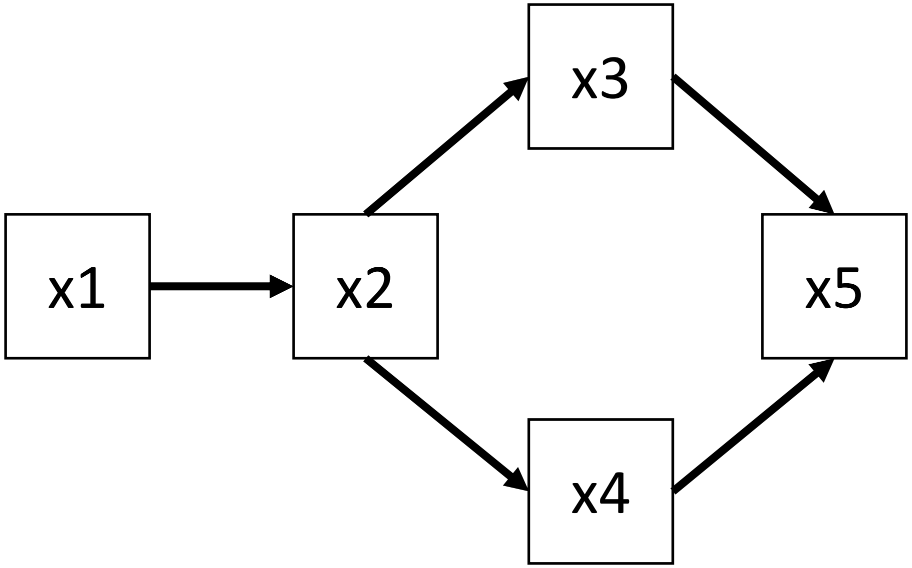

```
LLchisq(shipley_psem2)
  Chisq df P.Value
1 4.143  5   0.529
```

1.4. Fit using GAM and GLM



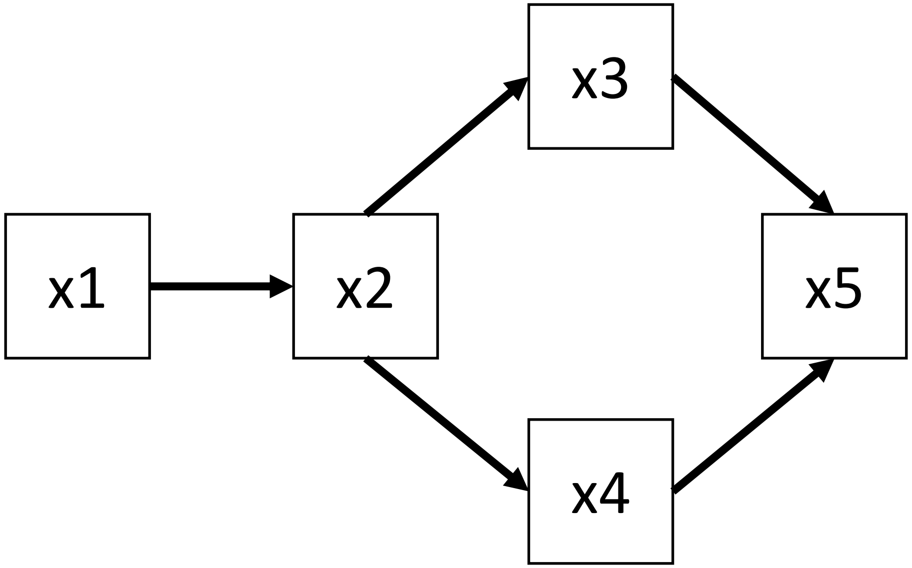

```r
shipley_psem3 <- psem(
  gam(x2 ~ s(x1), data = dat2, family = gaussian),
  glm(x3 ~ x2, data = dat2, family = poisson),
  gam(x4 ~ x2, data = dat2, family = poisson),
  glm(x5 ~ x3 + x4, data = dat2, family = binomial)
)
```

1.4. Fit using GAM and GLM




```
# Get goodness-of-fit
LLchisq(shipley_psem2)

  Chisq df P.Value
1 4.143  5   0.529
```

1.4. Fit using GAM and GLM




```
# Compare linear and non-linear models
AIC(shipley_psem2, shipley_psem3)

      AIC       K   n
1 1240.20 13.000 100
2 1190.75 11.563 100
```

1.4. Truly Non-Linear Implementations

- Possible to compare models with the same typology but different ML fitting functions and forms (or nested models)

- Do not get coefficients returned by `coefs` because smoothed terms are non-linear functions

- How to present this path diagram???

1.4. Truly Non-Linear Implementations

- Piecewise SEM can be extended to many different model types: as long as you can get a *P*-value or compute a log-likelihood, you can estimate fit
  - Matrix regression (Barnes et al. 2016)
  - Spatially-explicit models