

Lecture 11

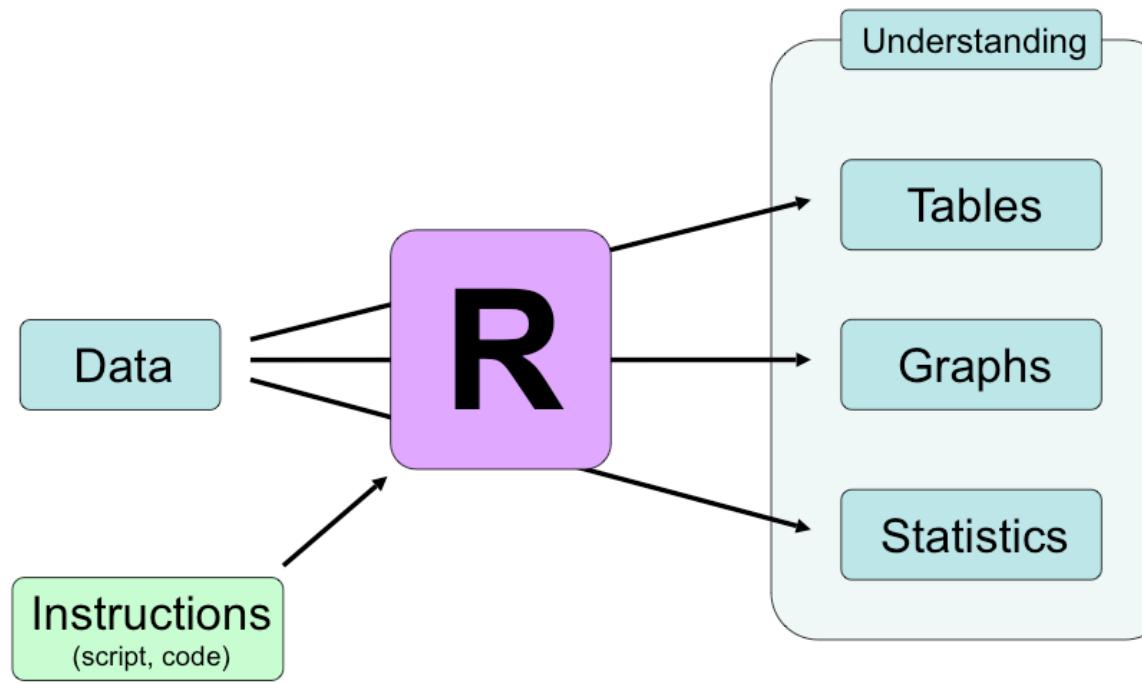
Mixed models
(repeated measures, random effects)

From the Learning Objectives

- Identify explanatory variables as better being included in models as *fixed* or as *random* effects.
- State what is a mixed model.
- Describe the type of biological / biomedical questions that required mixed models to analyse.
- Determine if a statistical test is probably pseudoreplicated.
- In R / RStudio, perform, check and interpret a simple linear mixed model.
- Do all of the same things as one does for a linear model.
- Say what is different about the model summary table produced by R / RStudio.

Lecture
BC material
IC exercise

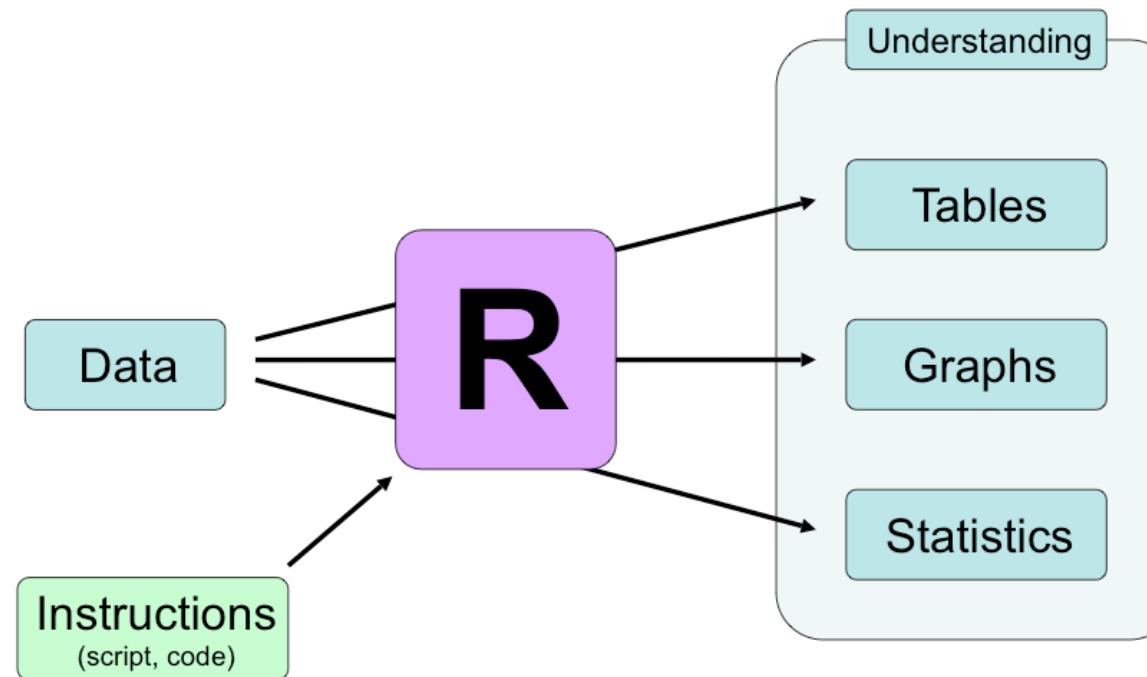
(And you will do more
of the learning yourself.
Less guidance.)



Mixed Models

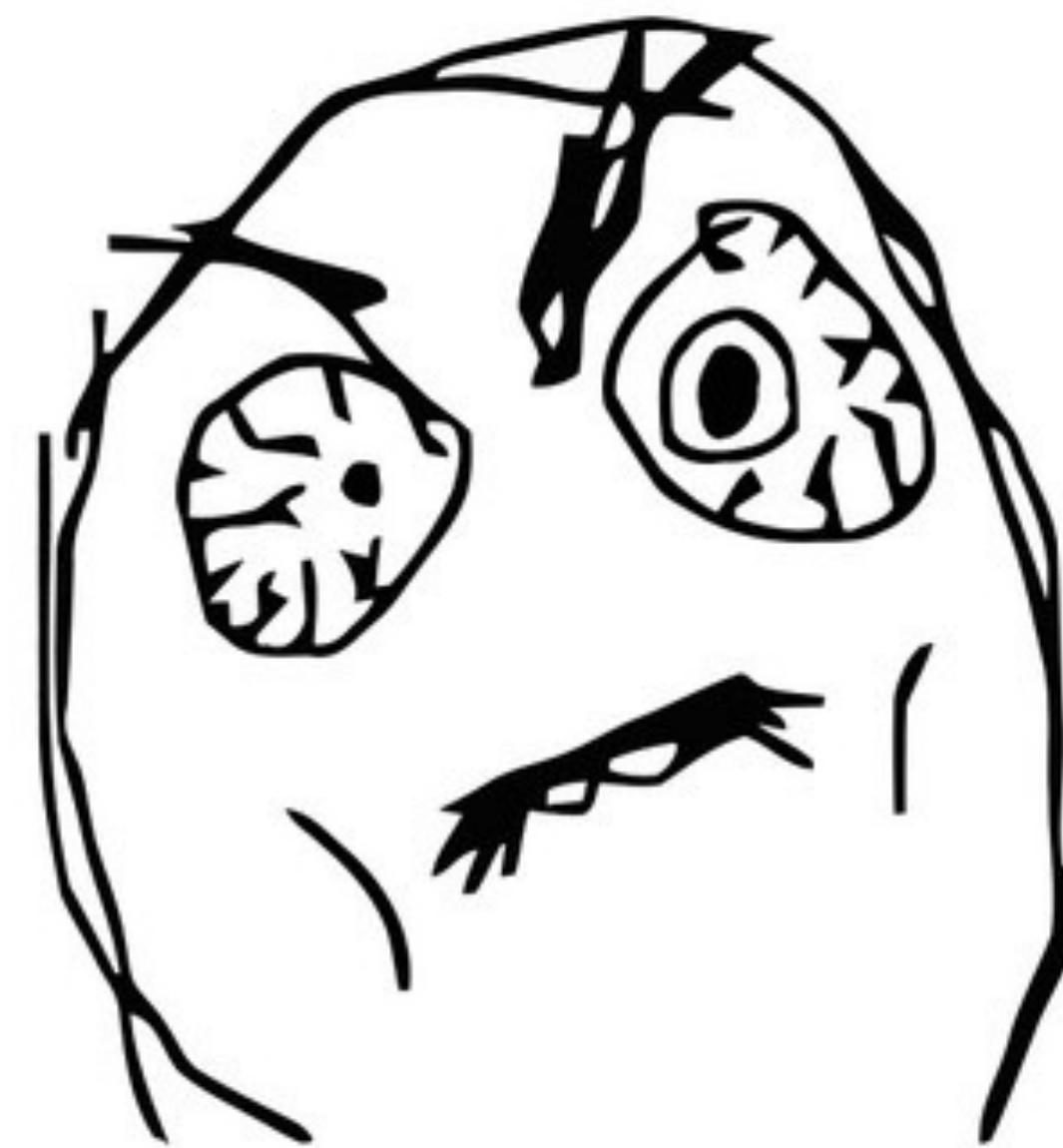
Powerful but dangerous





Mixed Models

Powerful but ~~dangerous~~ frustrating



Mixed models can be dangerous to your health



Warning on the r-mixed-model wiki:

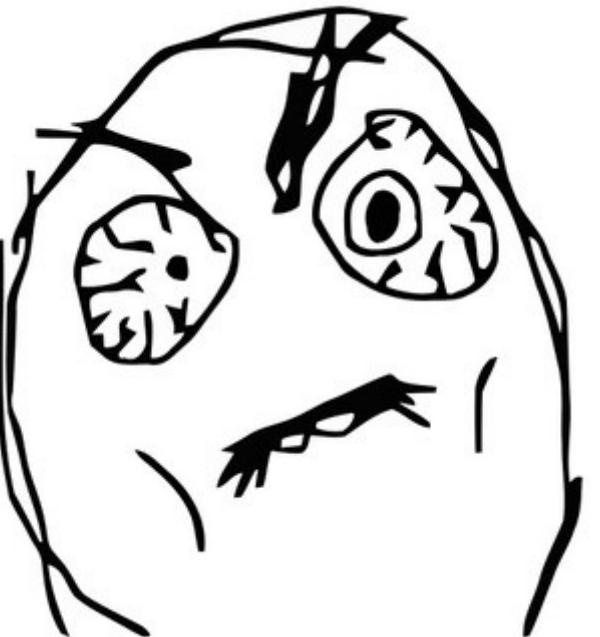
(G)LMMs are hard - harder than you may think based on what you may have learned in your second statistics class, which probably focused on picking the appropriate sums of squares terms and degrees of freedom for the numerator and denominator of an F test. ‘Modern’ mixed model approaches, which are much more powerful (they can handle complex designs, lack of balance, crossed random factors, some kinds of non-normally distributed responses, etc.), also require a new set of conceptual tools. **In order to use these tools you should have at least a general acquaintance with classical mixed-model experimental designs ...** If you are going to use generalized linear mixed models, you should understand generalized linear models... **All of the issues that arise with regular linear or generalized-linear modeling (e.g.: inadequacy of p-values alone for thorough statistical analysis; need to understand how models are parameterized; need to understand the principle of marginality and how interactions can be treated; dangers of overfitting, which are not mitigated by stepwise procedures; the non-existence of free lunches) also apply**, and can apply more severely, to mixed models.



lm
linear model
LM



lmer
linear mixed model
LMM



glm
generalised linear model
GLM



glmer
generalised linear mixed model
GLMM



Individual	Record	Gender	Reaction time
A	1	Female	410
A	2	Female	...
A	3	Female	...
B	1	Female	...
B	2	Female	...
B	3	Female	...
C	1	Female	...
C	2	Female	...
C	3	Female	...
D	1	Male	...
D	2	Male	...
D	3	Male	...
E	1	Male	...
E	2	Male	...
E	3	Male	...
F	1	Male	...
F	2	Male	...
F	3	Male	...

How much *independent* data?

How many *independent* data points?

Number of people?

Number of observations?

Somewhere in between?



Individual	Record	Gender	Reaction time
A	1	Female	410
A	2	Female	...
A	3	Female	...
B	1	Female	...
B	2	Female	...
B	3	Female	...
C	1	Female	...
C	2	Female	...
C	3	Female	...
D	1	Male	...
D	2	Male	...
D	3	Male	...
E	1	Male	...
E	2	Male	...
E	3	Male	...
F	1	Male	...
F	2	Male	...
F	3	Male	...

If we don't account for group... *pseudoreplication*

Assumes that repeated measures from an individual are just as independent as measures from different individuals

Degrees of freedom for error controlled by number of observations from each individual!

Bad



Individual	Record	Gender	Reaction time
A	1	Female	410
A	2	Female	...
A	3	Female	...
B	1	Female	...
B	2	Female	...
B	3	Female	...
C	1	Female	...
C	2	Female	...
C	3	Female	...
D	1	Male	...
D	2	Male	...
D	3	Male	...
E	1	Male	...
E	2	Male	...
E	3	Male	...
F	1	Male	...
F	2	Male	...
F	3	Male	...

So we cannot use each observation as an independent data point.



Individual	Record	Gender	Reaction time
A	1	Female	410
A	2	Female	...
A	3	Female	...
B	1	Female	...
B	2	Female	...
B	3	Female	...
C	1	Female	...
C	2	Female	...
C	3	Female	...
D	1	Male	...
D	2	Male	...
D	3	Male	...
E	1	Male	...
E	2	Male	...
E	3	Male	...
F	1	Male	...
F	2	Male	...
F	3	Male	...

Why not just average
to get
one value per individual?

Random

Fixed

Response

Individual	Record	Gender	Reaction time
A	1	Female	410
A	2	Female	average
A	3	Female	...
B	1	Female	...
B	2	Female	average
B	3	Female	...
C	1	Female	...
C	2	Female	average
C	3	Female	...
D	1	Male	...
D	2	Male	average
D	3	Male	...
E	1	Male	...
E	2	Male	average
E	3	Male	...
F	1	Male	...
F	2	Male	average
F	3	Male	...

Why might we not average to get one value per group?

(= why use mixed / multilevel / hierarchical models)

1. To adjust estimates for imbalanced sampling.
2. Which average?
3. Avoid false confidence.
4. To study variation.
5. To adjust estimates for repeat sampling (sharing information)
6. To keep information.
7. ...

1. To adjust estimates for imbalanced sampling

Individual	Record	Gender	Reaction time
A	1	Female	410
A	2	Female	...
A		Female	...
C		Female	...
C		Female	...
D		Male	...
D		Male	...
D		Male	...
E	1	Male	...
F	1	Male	...
F	2	Male	...

Treating every observation as independent gives individual A larger influence than individual C, for example

Analysing individual averages gives observations in individual C greater weight than those from individual A

2. Which average?

There are lots of averages:

- mean
- median
- mode



3. Avoid false confidence

We create one “observation” from multiple real observations.

Removing variation could easily lead to false confidence in our “observations”.



The image shows a Twitter profile for a user named "False confidence guy" (@Falseconfidence). The profile picture is a photo of a man with a mustache, wearing a black tank top, giving a thumbs-up. The background of the profile is blue. To the right of the profile picture is a "Follow" button with a person icon and the word "Follow". Below the profile picture, the username "False confidence guy" and the handle "@Falseconfidence" are displayed. A bio text reads "I can probably do that better than you". Underneath the bio, it says "1 FOLLOWING". At the bottom of the profile, there are three tabs: "TWEETS", "MEDIA", and "LIKES". A single tweet is visible, posted on "Mar 25, 2013". The tweet text is: "No one would talk to me at the party, My sleeveless underarmour shirt was probably way to intimidating #Biceps". Below the tweet are standard Twitter interaction icons: a reply arrow, a retweet arrow, a heart, and a message envelope.

4. To study variation

Obviously we need to keep the variation in order to study it.

If we want to study variation among individuals, we need multiple observations per individual.



5. Memory / sharing information

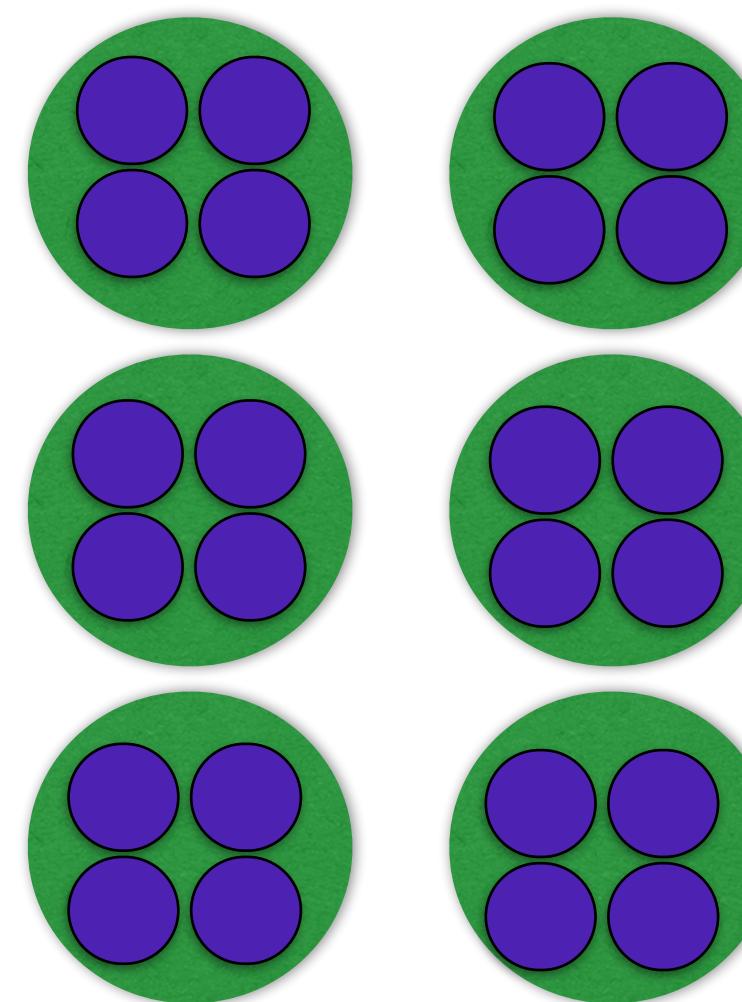
- **Single-level models:**
 - Every new cluster / group is a new world.
(individual, nest, pond, road, classroom)
 - No information passed among clusters.
- **Multilevel / hierarchical / mixed models:**
 - Remember and pool information
 - *Properties of clusters come from a population*
 - Inferred population defines pooling

6. Information is valuable

- **Don't discard it!**



Data are very often clustered



<u>Replicate (Intercept)</u>	<u>Pseudo-replicate</u>	<u>Pseudo²-replicate</u>
	Nests	Chicks
	Families/Dams/Sires	Offspring
	Genotypes	Individuals
	Fields/Plots	Quadrats
	Habitats	Fields/Plots
	CT Room	Jars/Cages
Your ‘Real’ N		Not the ‘Real’ N

Multiple measures
from each chick

We really need multilevel models!

Multilevel model

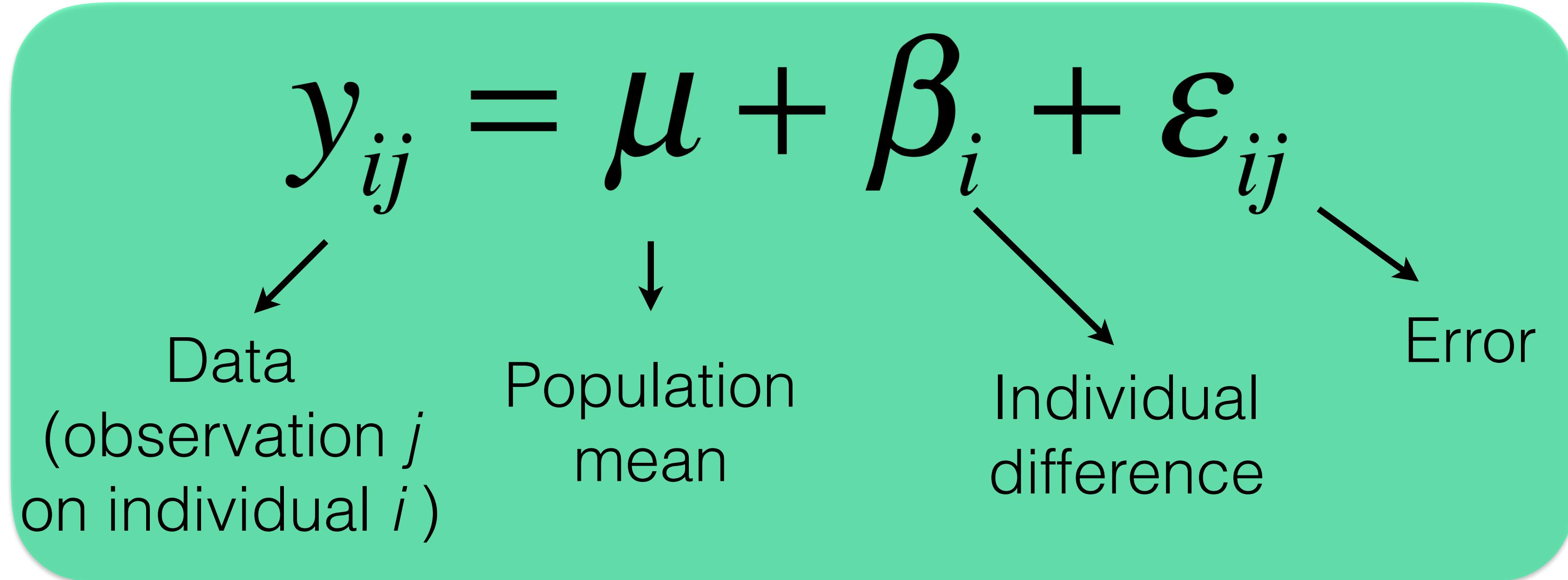
is largely synonymous with

mixed model

and

hierarchical model

Why are they called *multilevel models*?



$\beta_i \sim Norm(0, \sigma_i^2)$ This is what turns a linear
model into a
linear mixed model



If a fixed effect, B_i
would be a parameter.
Here, σ^2 is the
parameter

Fixed effects

Insecticide concentration
Temperature
Weight
Sex
Height
Spending
Herbivore
Brain size
Caste
Soil moisture
Landscape ruggedness

Groups / levels are
predetermined,
of direct interest,
repeatable

Random effects

Individual
Nest
Family
Incubator

Groups that are randomly sampled from a larger population of groups

Another way to think about it

Reaction times by **gender**,
with multiple measures per **individual**

*If we include
individual as a
fixed effect.*



```
lm(formula = y ~ sex + individual, data = dd)

Residuals:
    Min      1Q  Median      3Q     Max 
-105.588 -4.039 -0.181  1.385 93.292 

Coefficients: (1 not defined because of singularities)
              Estimate Std. Error t value Pr(>|t|)    
(Intercept)  2.24052   11.97911   0.187  0.852073  
sexmale      1.90260   16.07167   0.118  0.906041  
individualb  4.61440   14.50743   0.318  0.751202  
individualc 15.87175   14.02850   1.131  0.261035  
individuald 11.97688   15.15251   0.790  0.431455  
individuale 38.55930   16.07167   2.399  0.018591 *  
individualf 18.70002   17.49662   1.069  0.288160  
individualg 14.82729   14.02850   1.057  0.293499  
individualh 76.21909   14.50743   5.254  1.07e-06 *** 
individuali 66.81303   14.50743   4.605  1.41e-05 *** 
individualj 61.27430   16.07167   3.813  0.000258 *** 
individualk -1.02675   16.07167  -0.064  0.949209  
individuall  0.34023   15.46497   0.022  0.982499  
individualm  0.25280   15.01661   0.017  0.986608  
individualn -0.14553   16.07167  -0.009  0.992796  
individualo -0.86790   16.94103  -0.051  0.959261  
individualp  0.07921   18.29840   0.004  0.996556  
individualq -0.23622   15.01661  -0.016  0.987486  
individualr -0.23610   15.46497  -0.015  0.987854  
individuals -0.71004   15.46497  -0.046  0.963486  
individualt      NA        NA        NA        NA      
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 23.96 on 86 degrees of freedom
Multiple R-squared:  0.5749, Adjusted R-squared:  0.481 
F-statistic: 6.121 on 19 and 86 DF,  p-value: 1.728e-09
```

Another way to think about it

Reaction times by **gender**, with multiple measures per **individual**

If we include individual as a random effect.



```
Linear mixed model fit by REML [ 'lmerMod' ]
Formula: y ~ sex + (1 | individual)
Data: dd
```

```
REML criterion at convergence: 986.4
```

```
Scaled residuals:
```

Min	1Q	Median	3Q	Max
-4.0446	-0.2427	-0.0191	0.0497	4.3724

```
Random effects:
```

Groups	Name	Variance	Std.Dev.
individual	(Intercept)	289.8	17.02
	Residual	571.4	23.90

```
Number of obs: 106, groups: individual, 20
```

```
Fixed effects:
```

	Estimate	Std. Error	t value
(Intercept)	1.988	6.346	0.313
sexmale	32.982	8.974	3.675

```
Correlation of Fixed Effects:
```

(Intr)
sexmale -0.707

```
lm(formula = y ~ sex + individual, data = dd)
```

Residuals:

	Min	1Q	Median	3Q	Max
	-105.588	-4.039	-0.181	1.385	93.292

Coefficients: (1 not defined because of singularities)

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	2.24052	11.97911	0.187	0.852073
sexmale	1.90260	16.07167	0.118	0.906041
individualb	4.61440	14.50743	0.318	0.751202
individualc	15.87175	14.02850	1.131	0.261035
individuald	11.97688	15.15251	0.790	0.431455
individuale	38.55930	16.07167	2.399	0.018591 *
individualf	18.70002	17.49662	1.069	0.288160
individualg	14.82729	14.02850	1.057	0.293499
individualh	76.21909	14.50743	5.254	1.07e-06 ***
individuali	66.81303	14.50743	4.605	1.41e-05 ***
individualj	61.27430	16.07167	3.813	0.000258 ***
individualk	-1.02675	16.07167	-0.064	0.949209
individuall	0.34023	15.46497	0.022	0.982499
individualm	0.25280	15.01661	0.017	0.986608
individualn	-0.14553	16.07167	-0.009	0.992796
individualo	-0.86790	16.94103	-0.051	0.959261
individualp	0.07921	18.29840	0.004	0.996556
individualq	-0.23622	15.01661	-0.016	0.987486
individualr	-0.23610	15.46497	-0.015	0.987854
individuals	-0.71004	15.46497	-0.046	0.963486
individualt	NA	NA	NA	NA

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 23.96 on 86 degrees of freedom

Multiple R-squared: 0.5749, Adjusted R-squared: 0.481

F-statistic: 6.121 on 19 and 86 DF, p-value: 1.728e-09

Linear mixed model fit by REML ['lmerMod']

Formula: y ~ sex + (1 | individual)

Data: dd

REML criterion at convergence: 986.4

Scaled residuals:

	Min	1Q	Median	3Q	Max
	-4.0446	-0.2427	-0.0191	0.0497	4.3724

Random effects:

Groups	Name	Variance	Std.Dev.
individual	(Intercept)	289.8	17.02
	Residual	571.4	23.90

Number of obs: 106, groups: individual, 20

Fixed effects:

	Estimate	Std. Error	t value
(Intercept)	1.988	6.346	0.313
sexmale	32.982	8.974	3.675

Correlation of Fixed Effects:

	(Intr)
sexmale	-0.707

Order of Things

1. **Plot the data**
2. Make a model (+check summary)
3. Check your assumptions
4. Do some “stats”

Model formula in lmer (in package lme4)

The fixed effects



You know how to do this

The random effects



The random effects bit is specified inside brackets (with one or more such components)

Read this as “the thing on the left (e.g. `ranef1`) varies by (|) the thing on the right (e.g. `groupvar`)”

So why are mixed models so hard then?

If we have M observations and a **grouping** variable with N levels then we have either

→ **M - N - 1**

OR

→ **M - 2** degrees of freedom “left over”

In truth, we’re somewhere in the middle of these two cases.

This is at the root of the problem for why significance tests of fixed effects are difficult when using mixed models

Three questions we can ask about mixed models when learning to use them

- 1.What are the “problems” with my data that they can accommodate?
- 2.What kind of scientific questions can they help me answer?
(I.e. the random effect might be interesting!)
- 3.How do the models relate to (and differ from) general(ised) linear models?

From the Learning Objectives

- Identify explanatory variables as better being included in models as *fixed* or as *random* effects.
- State what is a mixed model.
- Describe the type of biological / biomedical questions that required mixed models to analyse.
- Determine if a statistical test is probably pseudoreplicated.
- In R / RStudio, perform, check and interpret a simple linear mixed model.
- Do all of the same things as one does for a linear model.
- Say what is different about the model summary table produced by R / RStudio.

Lecture
BC material
IC exercise

(And you will do more
of the learning yourself.
Less guidance.)

End of lecture

Following slides may be useful BC material, but also are quite advanced.

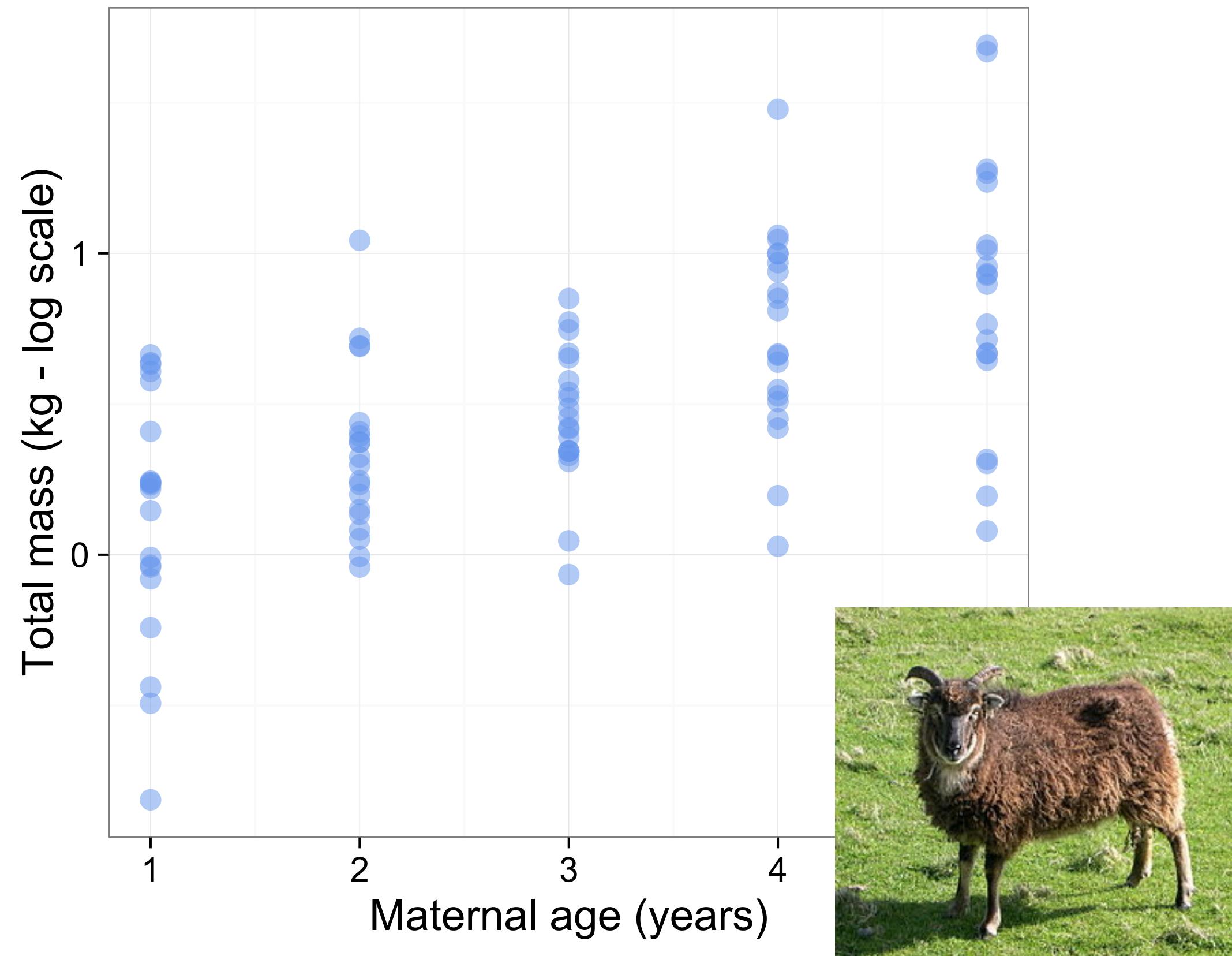
Great!
You know why one would use a
mixed / multilevel / hierarchical model
and what are
fixed effects and random effects

We'll now focus on a common situation that well dealt with by a mixed model:
repeated measures of individuals

But it ain't easy, hang on...

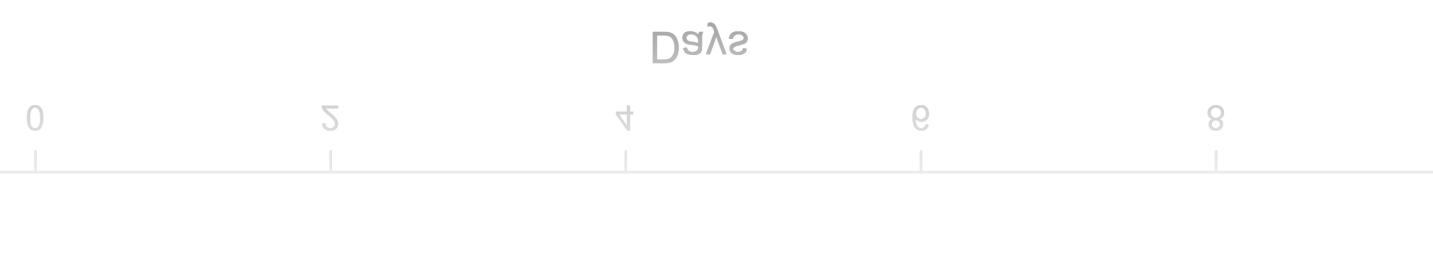
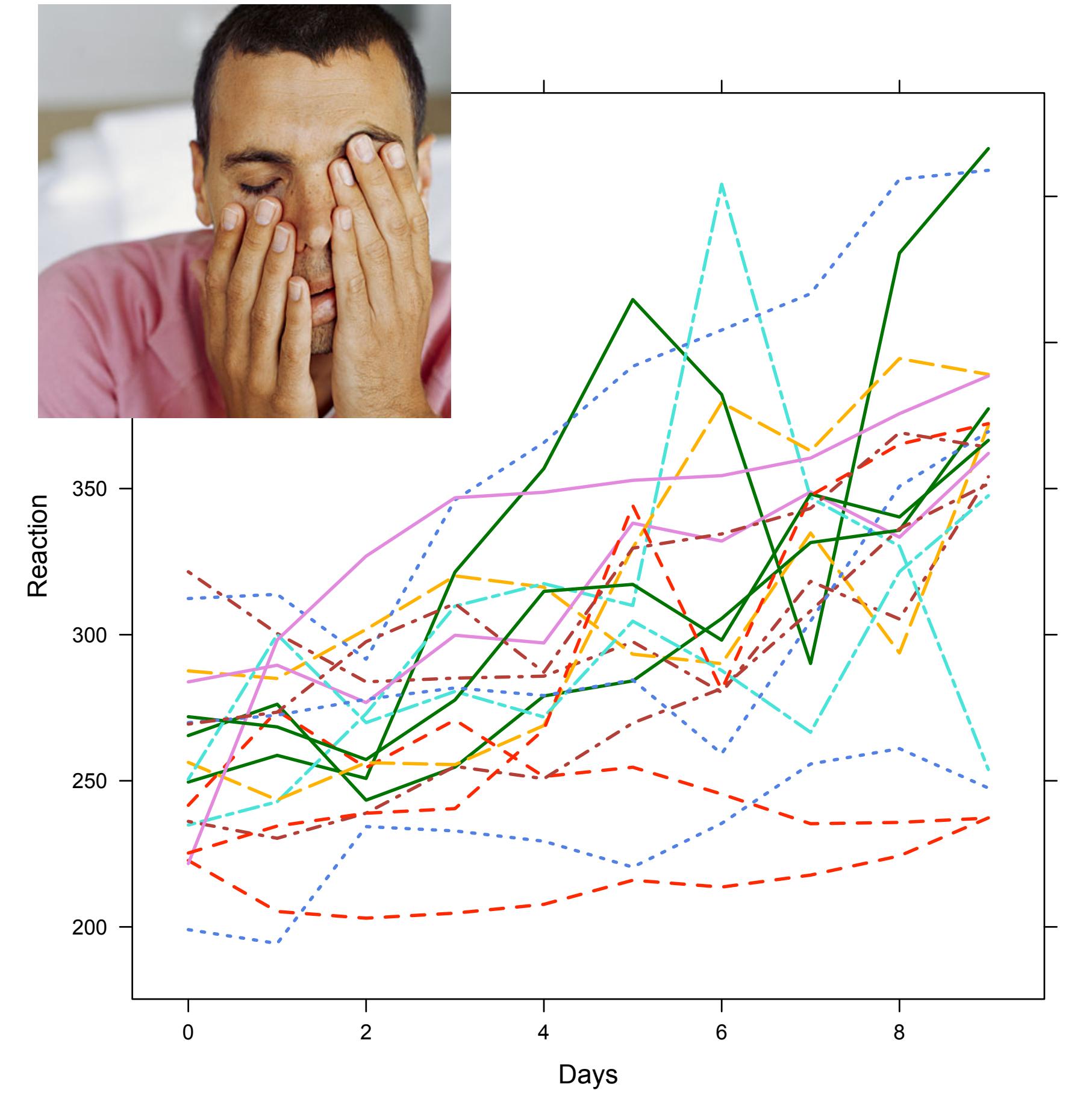


Reproductive effort in Soay sheep

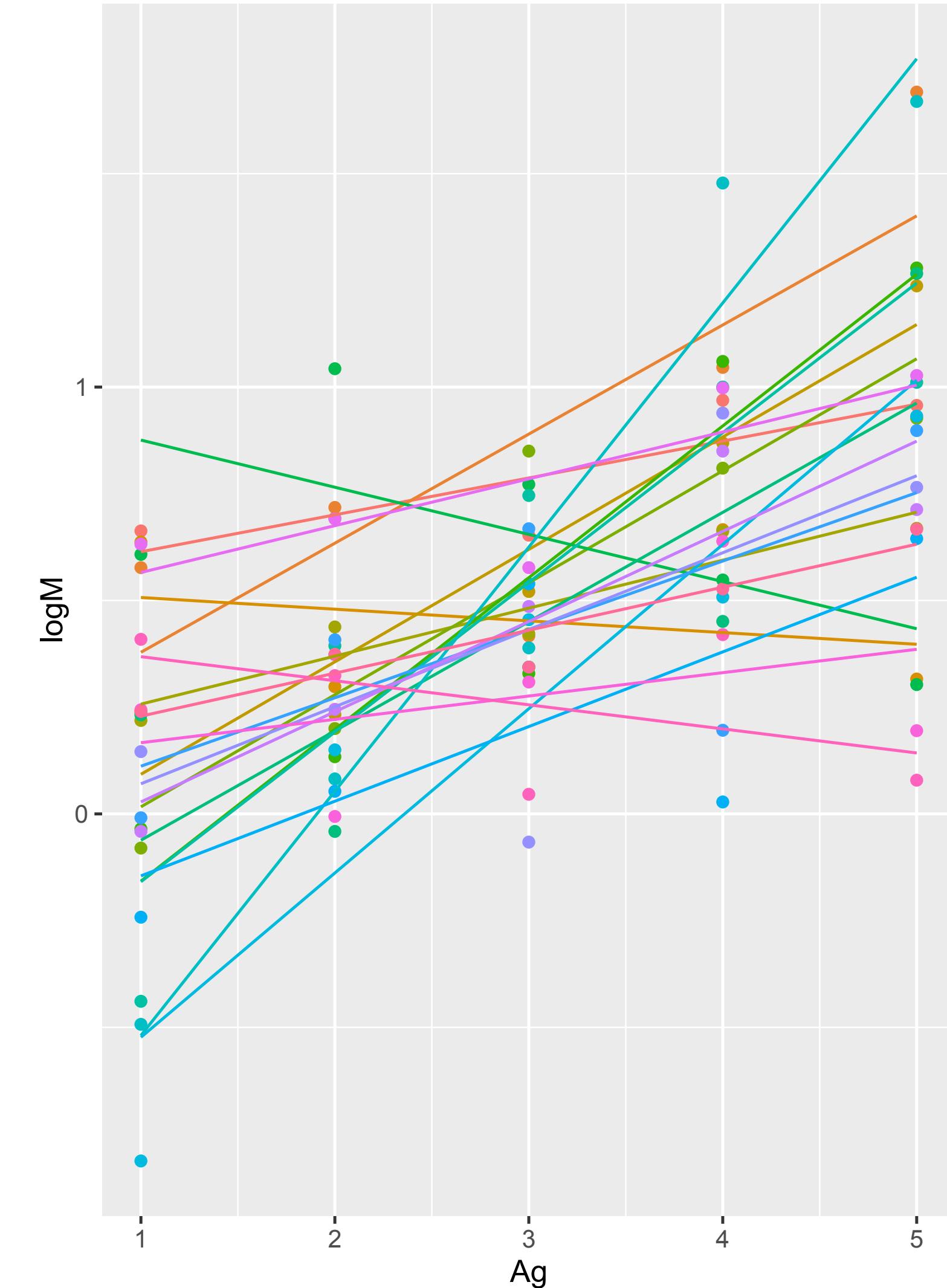
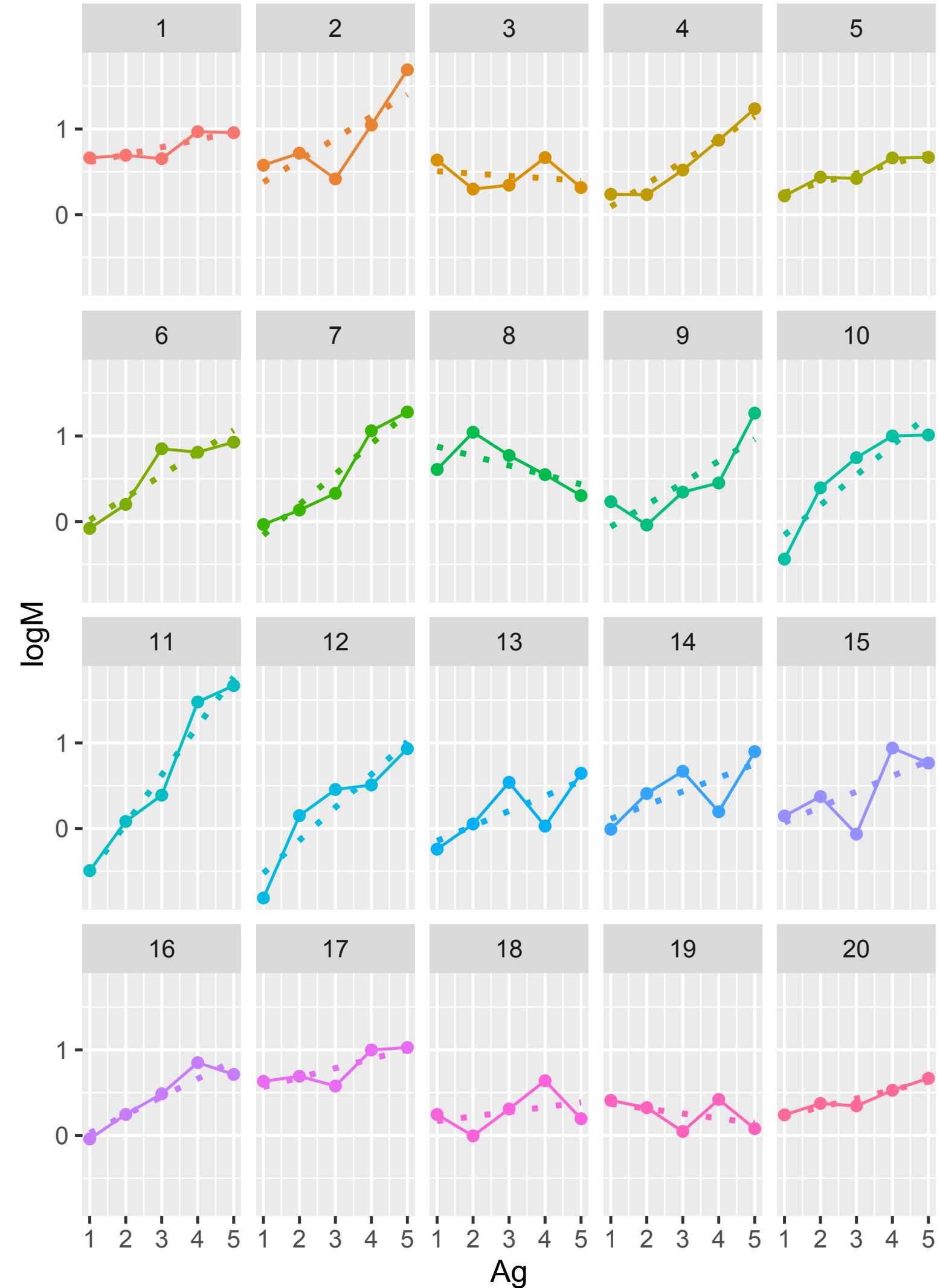


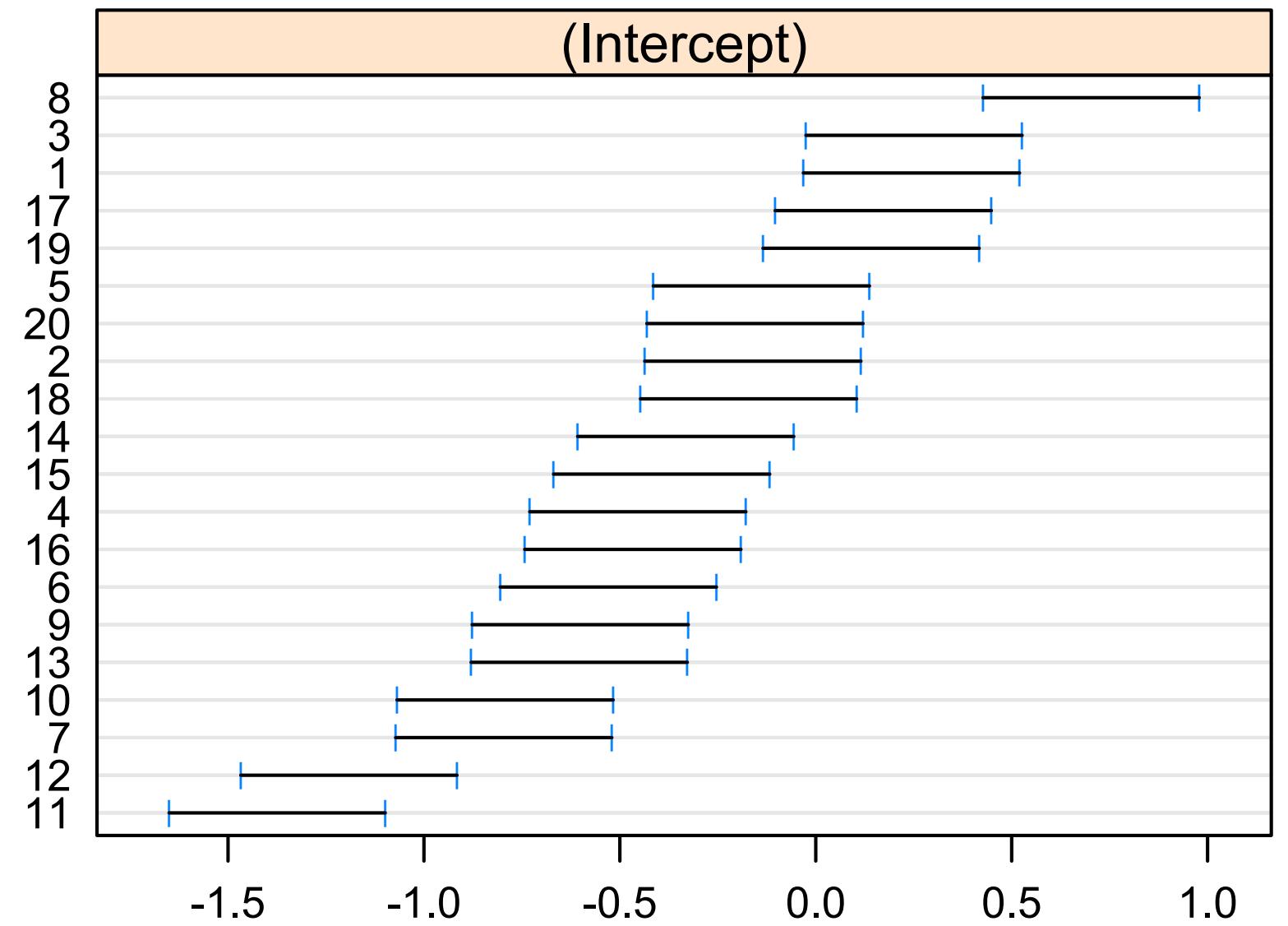
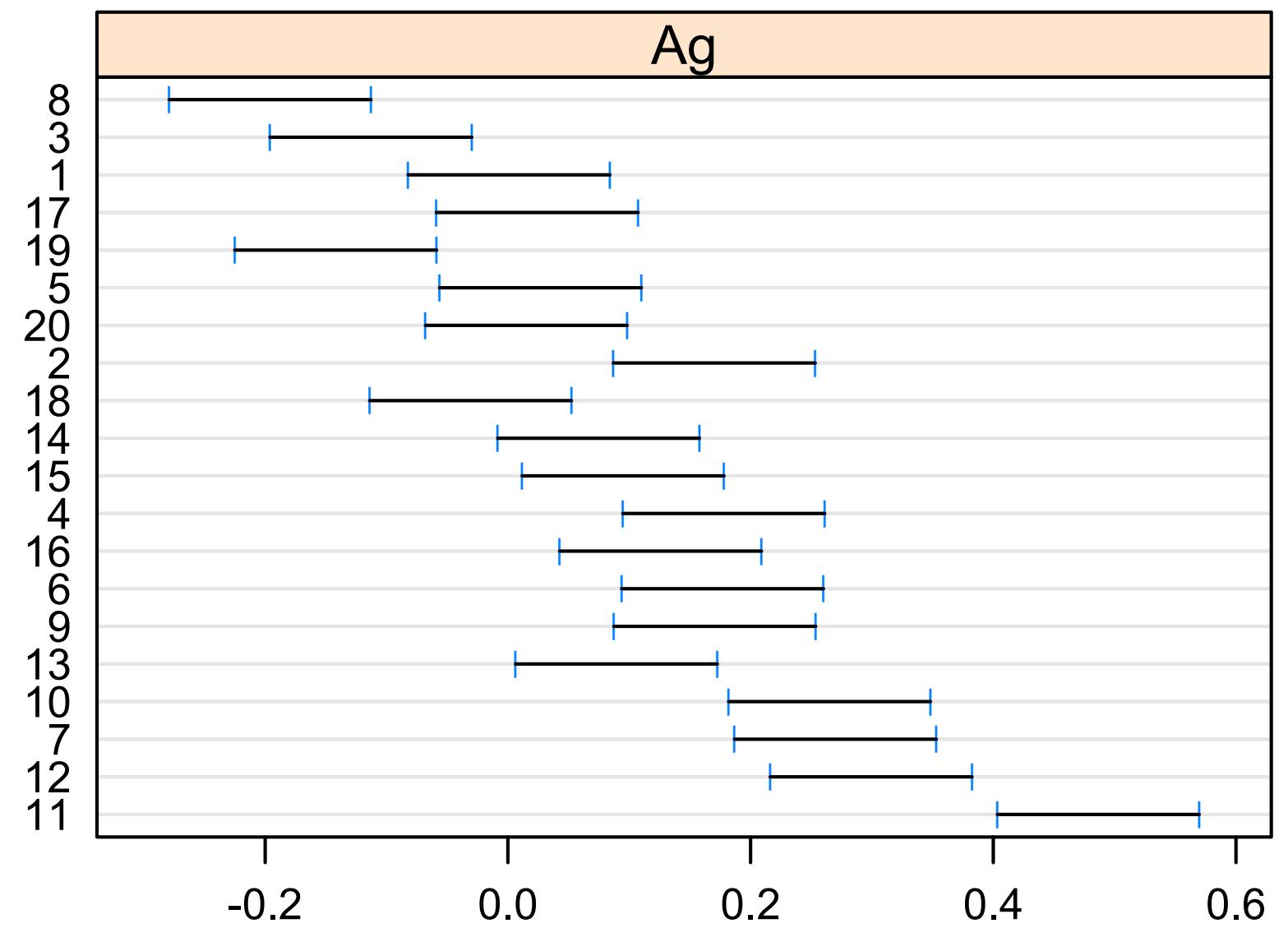
soay_mass_data.csv

Sleep deprivation and reaction times



Non-independence....



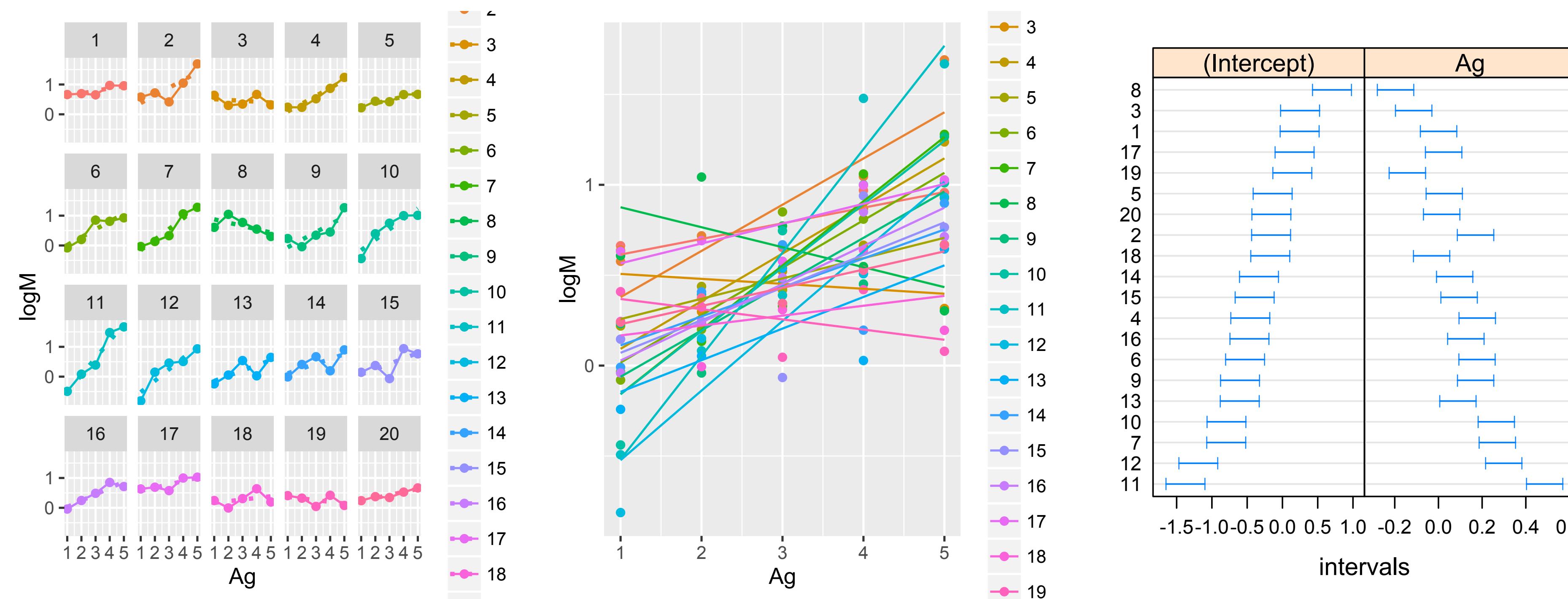


Fit a separate model for each individual and look at the intercepts and slopes

What does this plot suggest?

Make plots that let you see....

- The variation among Id in INTERCEPTS
- The variation among Id in SLOPES
- The CORRELATION between Intercepts and Slopes



Order of Things

1. Plot the data
2. Make a model (+check summary)
3. Check your assumptions
4. Do some “stats”

Many standard mixed models can be fit in lmer

- Single random effect (e.g. repeated measure)

$\dots + (1 | \text{genotype})$

- Correlated random effects

$\dots + (1 + \text{sex} | \text{year})$ *** an interaction*

- Random regression

$\dots + (1 + \text{age} | \text{individual})$

- Nested random effects

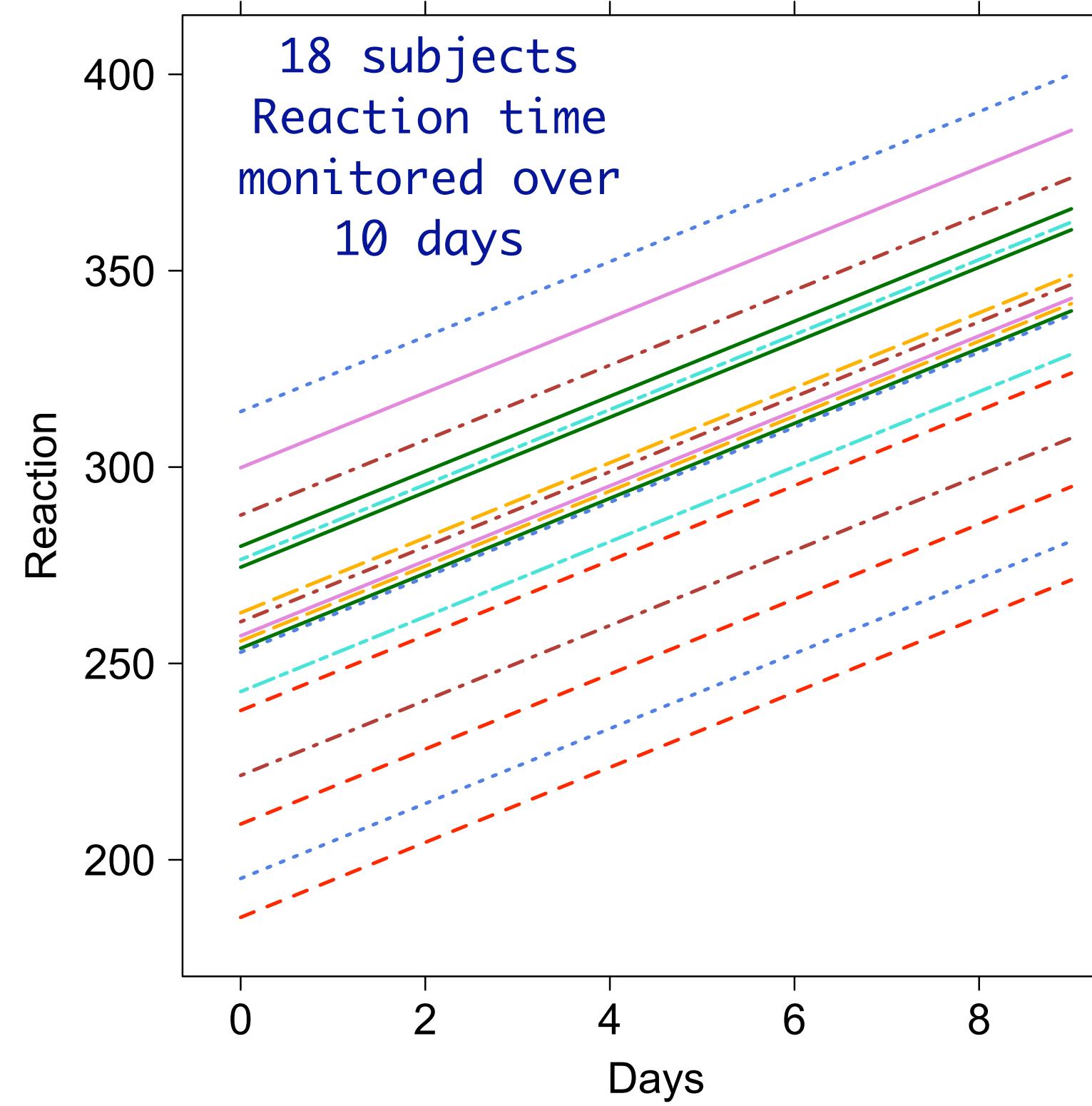
$\dots + (1 | \text{field/plot})$ *** see later slide*

- Crossed random effects

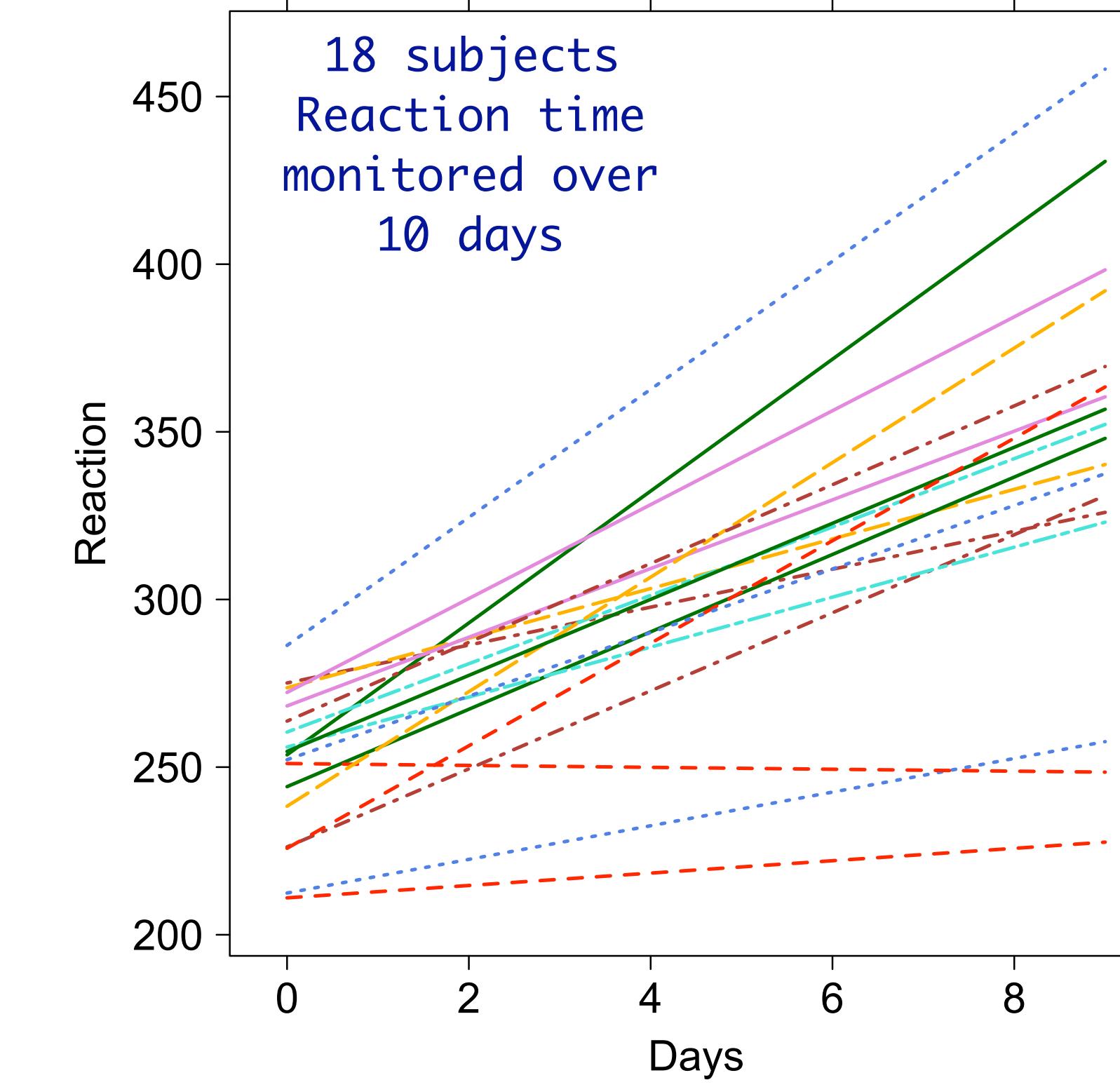
$\dots + (1 | \text{year}) + (1 | \text{sheep})$

Random intercepts vs. “Random regression”

Only the intercept
varies by subject

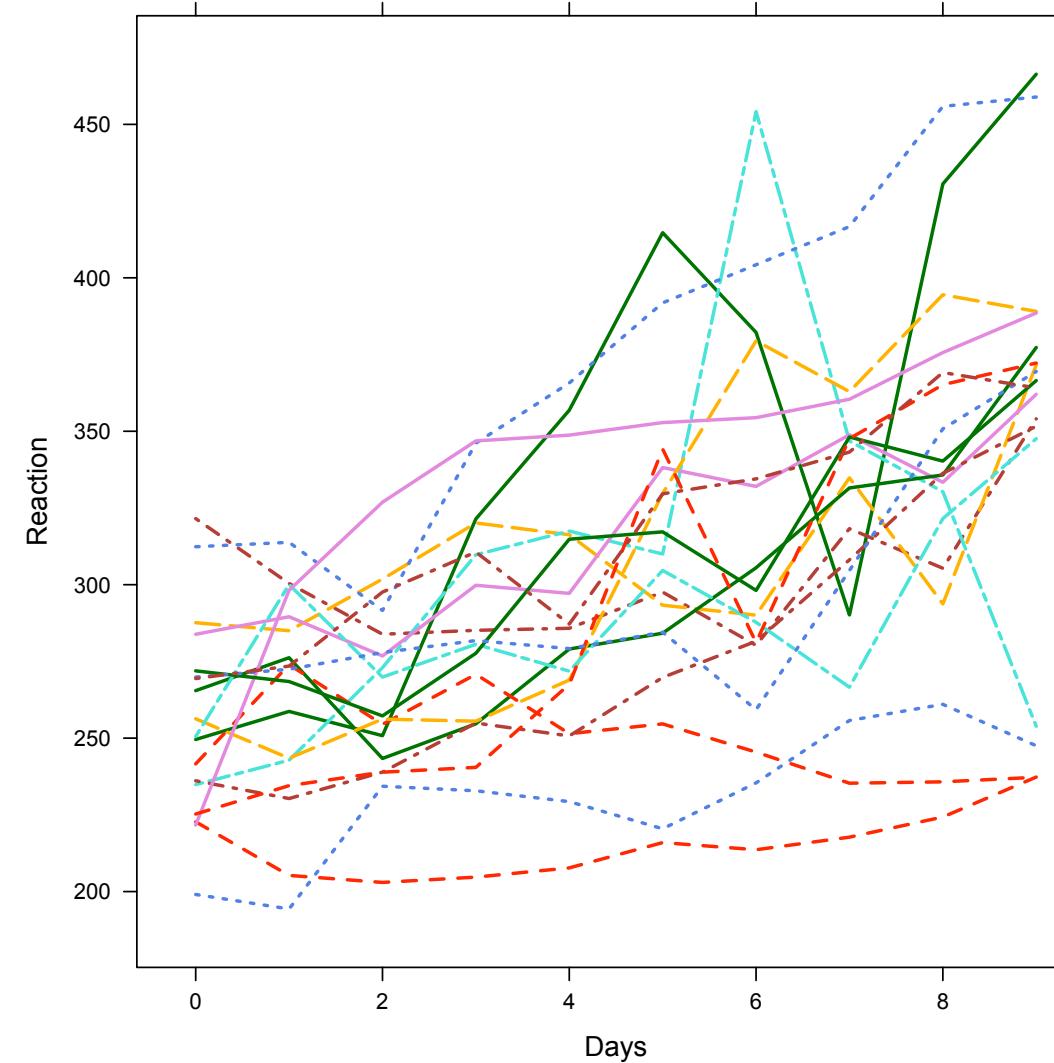


Intercept and slope
= random regression



A simple example bundled in lmer

18 subjects reaction
times monitored over 10
days



What could we test?

1. Each subject has different inherent response (intercept) and no response through time (0 slope) → $\text{Reaction} \sim 1 + (1 \mid \text{subject})$
2. Each subject has different inherent reaction, but same response through time → $\text{Reaction} \sim \text{Days} + (1 \mid \text{Subject})$
3. Each subject has different inherent reaction and response through time → $\text{Reaction} \sim \text{Days} + (\text{Days} \mid \text{Subjects})$
4. Each subject has a different reaction and response but these are uncorrelated → $\text{Reaction} \sim \text{Days} + (1 \mid \text{Subjects}) + (0 + \text{Days} \mid \text{Subjects})$

The lmer syntax

Order of Things

1. Plot the data
2. Make a model (+check summary)
3. Check your assumptions
4. Do some “stats”

Make the full model

```
offMassMod1 <- lmer(logM ~ Ag + (1 + Ag | Id),  
data=offMassData)
```



The Intercept & Slope
Vary with Mum

Results

```
> offMassMod1
Linear mixed model fit by REML [ 'lmerMod' ]
Formula: logM ~ Ag + (1 + Ag | Id)
Data: offMassData
```

REML criterion at convergence: 54

Scaled residuals:

Min	1Q	Median	3Q	Max
-2.24353	-0.50273	0.04592	0.64857	1.81782

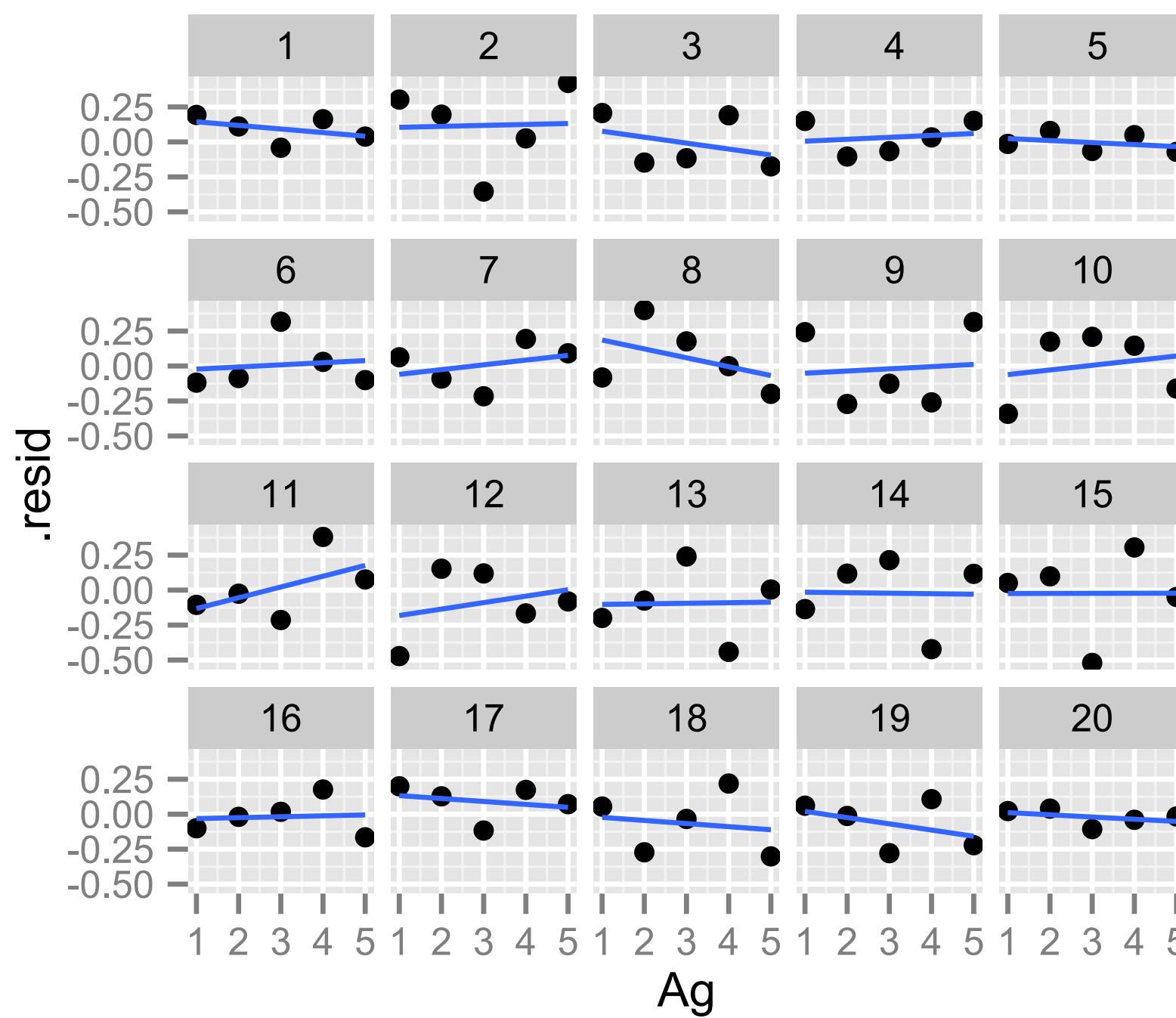
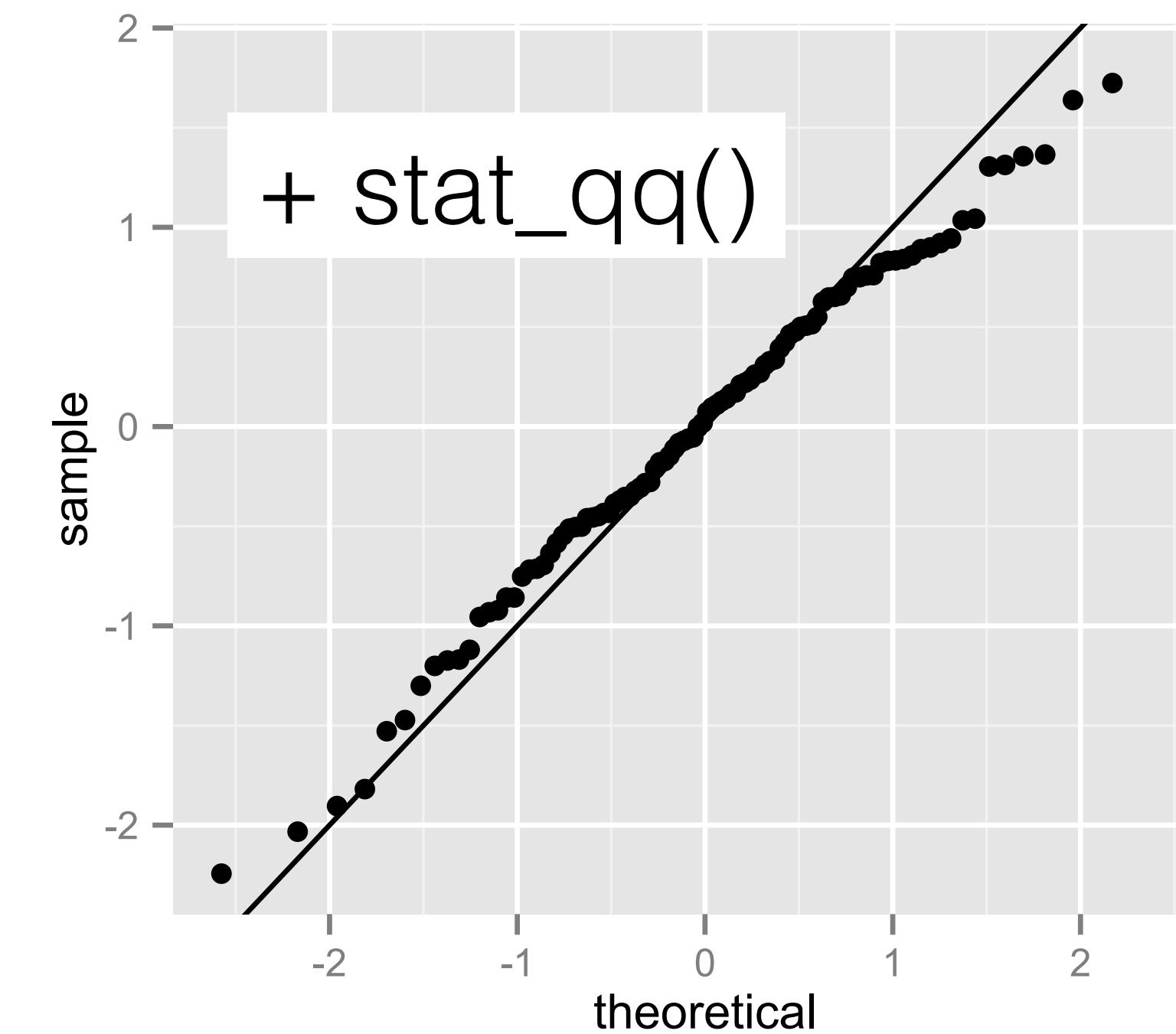
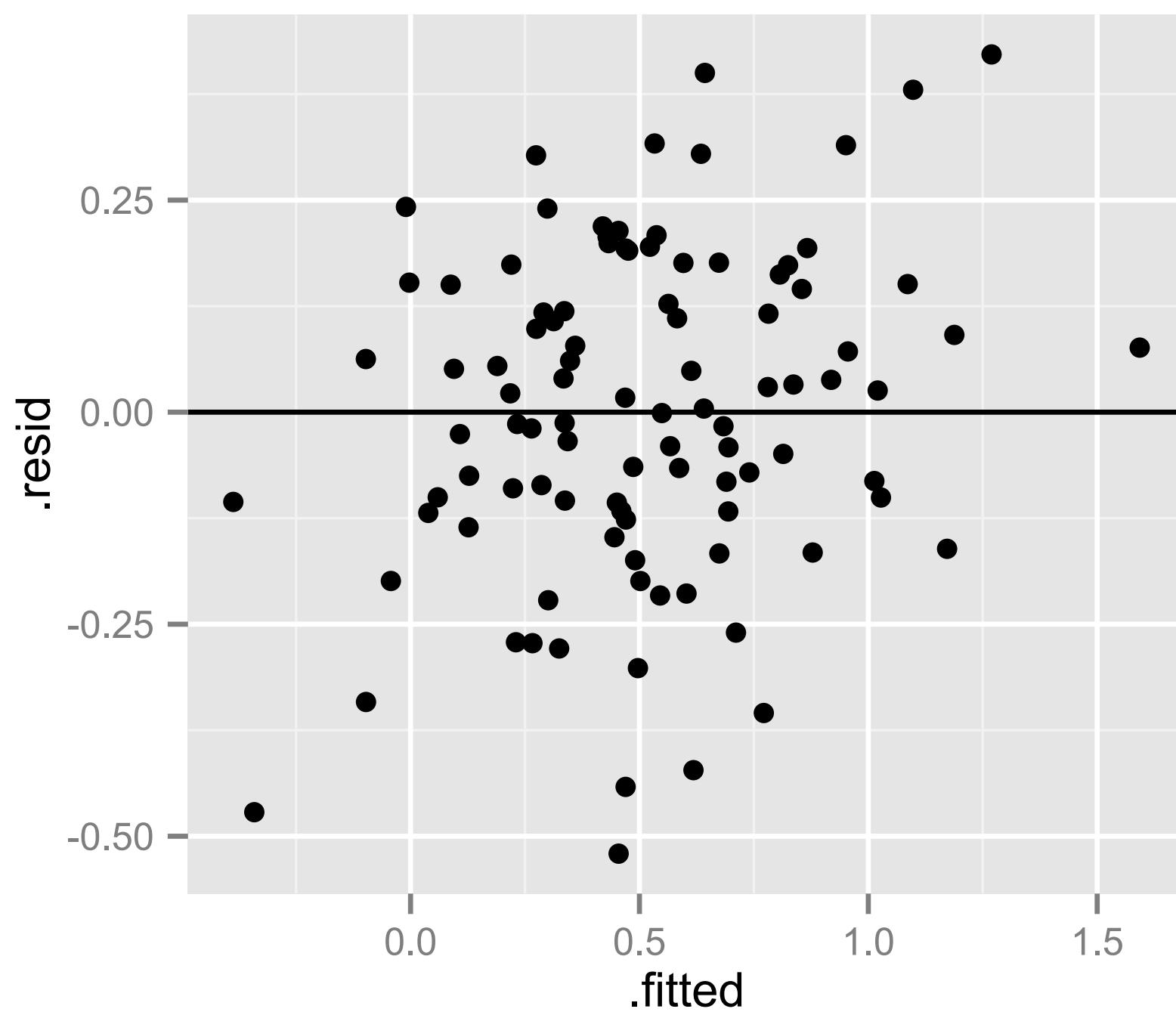
Random effects:

Groups	Name	Variance	Std.Dev.	Corr
Id	(Intercept)	0.19449	0.4410	
	Ag	0.02121	0.1456	-0.94
Residual		0.05383	0.2320	

Number of obs: 100, groups: Id, 20

Fixed effects:

	Estimate	Std. Error	t value
(Intercept)	-0.04897	0.11263	-0.435
Ag	0.18483	0.03646	5.069



See if you can
make these with
ggplot...

Mixed model diagnostics

Take home message: it is up to you to decide what to check when working with mixed models

lme4: Useful functions

Fitted values of ‘fixed’ component

```
> fitted(modelObject)
```

Residuals

```
> resid(modelObject)
```

Random effects (‘predictors’)

```
> ranef(modelObject)
```

Coefficients (fixed + random)

```
> coef(modelObject)
```

Plot Many aspects of the model

```
> fortify(modelObject)
```

All the functions you can use on a fitted model object

```
> ?merMod # <- also look at the ‘See Also’ section in this  
help file
```

Order of Things

1. Plot the data
2. Make a model (+check summary)
3. Check your assumptions
4. **Do some “stats”**

R Help

< > Print

Q pvalues

pvalues {lme4}

R Documentation

Getting p-values for fitted models

Description

One of the most frequently asked questions about `lme4` is "how do I calculate p-values for estimated parameters?" Previous versions of `lme4` provided the `mcmcSamp` function, which efficiently generated a Markov chain Monte Carlo sample from the posterior distribution of the parameters, assuming flat (scaled likelihood) priors. Due to difficulty in constructing a version of `mcmcSamp` that was reliable even in cases where the estimated random effect variances were near zero (e.g. <https://stat.ethz.ch/pipermail/r-sig-mixed-models/2009q4/003115.html>), `mcmcSamp` has been withdrawn (or more precisely, not updated to work with `lme4` versions $\geq 1.0.0$).

Many users, including users of the `aovlmer.fnc` function from the `languageR` package which relies on `mcmcSamp`, will be deeply disappointed by this lacuna. Users who need p-values have a variety of options:

- likelihood ratio tests via `anova` (MC,+)
- profile confidence intervals via `profile.merMod` and `confint.merMod` (CI,+)
- parametric bootstrap confidence intervals and model comparisons via `bootMer` (or `PBmodcomp` in the `pbkrtest` package) (MC/CI,*,+)
- for random effects, simulation tests via the `RLRsim` package (MC,*)
- for fixed effects, F tests via Kenward-Roger approximation using `KRmodcomp` from the `pbkrtest` package (MC)
- `car:::Anova` and `lmerTest::anova` provide wrappers for `pbkrtest::lmerTest::anova` also provides t tests via the Satterthwaite approximation (P,*)

In the list above, the methods marked MC provide explicit model comparisons; CI denotes confidence intervals; and P denotes parameter-level or sequential tests of all effects in a model. The starred (*) suggestions provide finite-size corrections (important when the number of groups is < 50); those marked (+) support GLMMs as well as LMMs.

When all else fails, don't forget to keep p-values in perspective: <http://www.phdcomics.com/comics/archive.php?comicid=905>

[Package `lme4` version 1.1-0 [Index](#)]

How do we generate p-values?

Three options:

1. Likelihood ratio tests (*fixed* or *random*)
2. Parametric bootstrap (*fixed* or *random*)
3. Adjusted F-tests (*fixed* only)

[Or... just construct confidence intervals]

One last bit of theory (sort of)

REML (Restricted Maximum Likelihood)

versus

ML (Maximum Likelihood)

What's wrong
with this?

e.g. estimating a variance:

$$\sum_i \frac{(x_i - \bar{x})^2}{n-1}$$

versus

$$\sum_i \frac{(x_i - \bar{x})^2}{n}$$

REML

ML



One last bit of theory (sort of)

REML (Restricted Maximum Likelihood)

versus

ML (Maximum Likelihood)

Biased!

e.g. estimating a variance:

$$\sum_i \frac{(x_i - \bar{x})^2}{n-1}$$

versus

$$\sum_i \frac{(x_i - \bar{x})^2}{n}$$

REML

ML



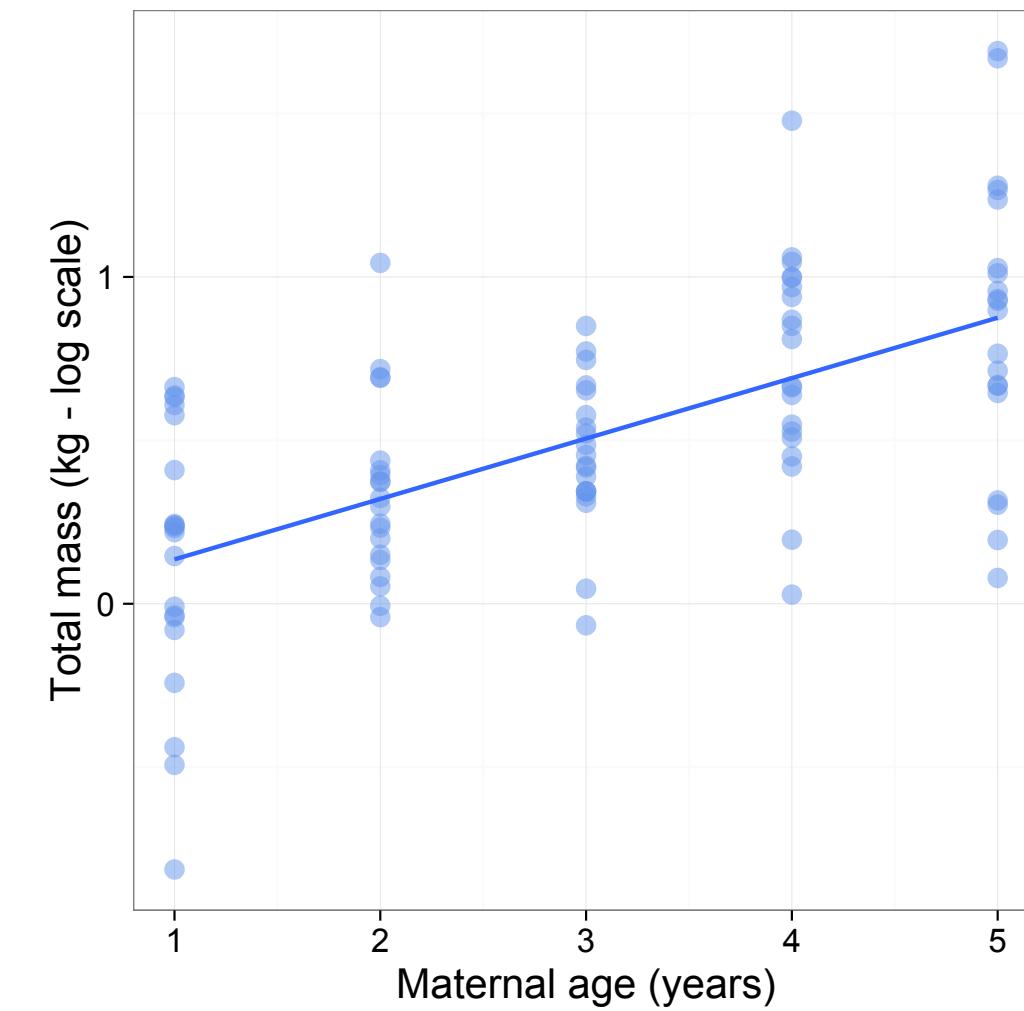
Likelihood ratio tests — Fixed effects

Is the fixed age ('Ag') effect significant? You already know how to do this....

1. Fit the “full” model
2. Fit the “reduced” model
3. Compare using the ...?.... function

Exercise: Check whether the fixed “age” effect on Soay litter mass is significantly different from

Testing fixed effects



```
## Is the fixed effect of age significant?
```

```
offMassMod1 <- lmer(logM ~ 1 + Ag + (1 + Ag | Id), offMassData)
offMassMod2 <- lmer(logM ~ 1 +           (1 + Ag | Id), offMassData)
```

```
anova(offMassMod1, offMassMod2) # yes
```



Using a Likelihood Ratio Test
Defaults to ML based test
Fixed Effects Comparison

Likelihood ratio tests — Random effects

The first question you should ask yourself is
should I really be examining the significance
of a random effect?

- ‘sacrificial pseudoreplication’
- do you care about the random effect?

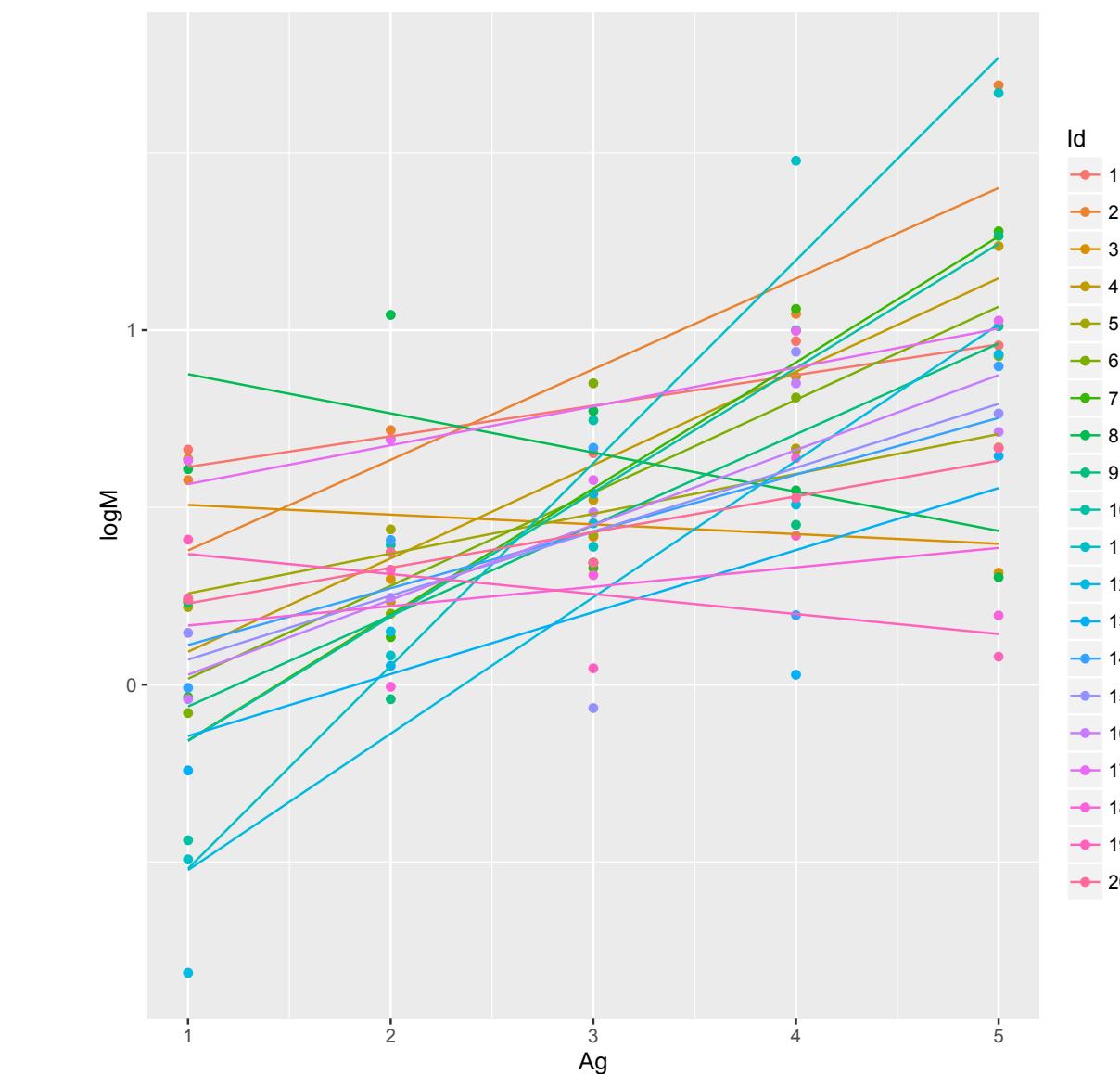
Likelihood ratio tests — Random effects

You already know how to do this (again)....

Exercise: Check whether the correlation between random intercept and age slope is significantly different from zero.

Hint: Need to use the REML fits. Check the help for `anova.merMod` and read about the `refit` argument.

Testing “random” terms



```
## Is the among-individual age-slope variation significant?
```

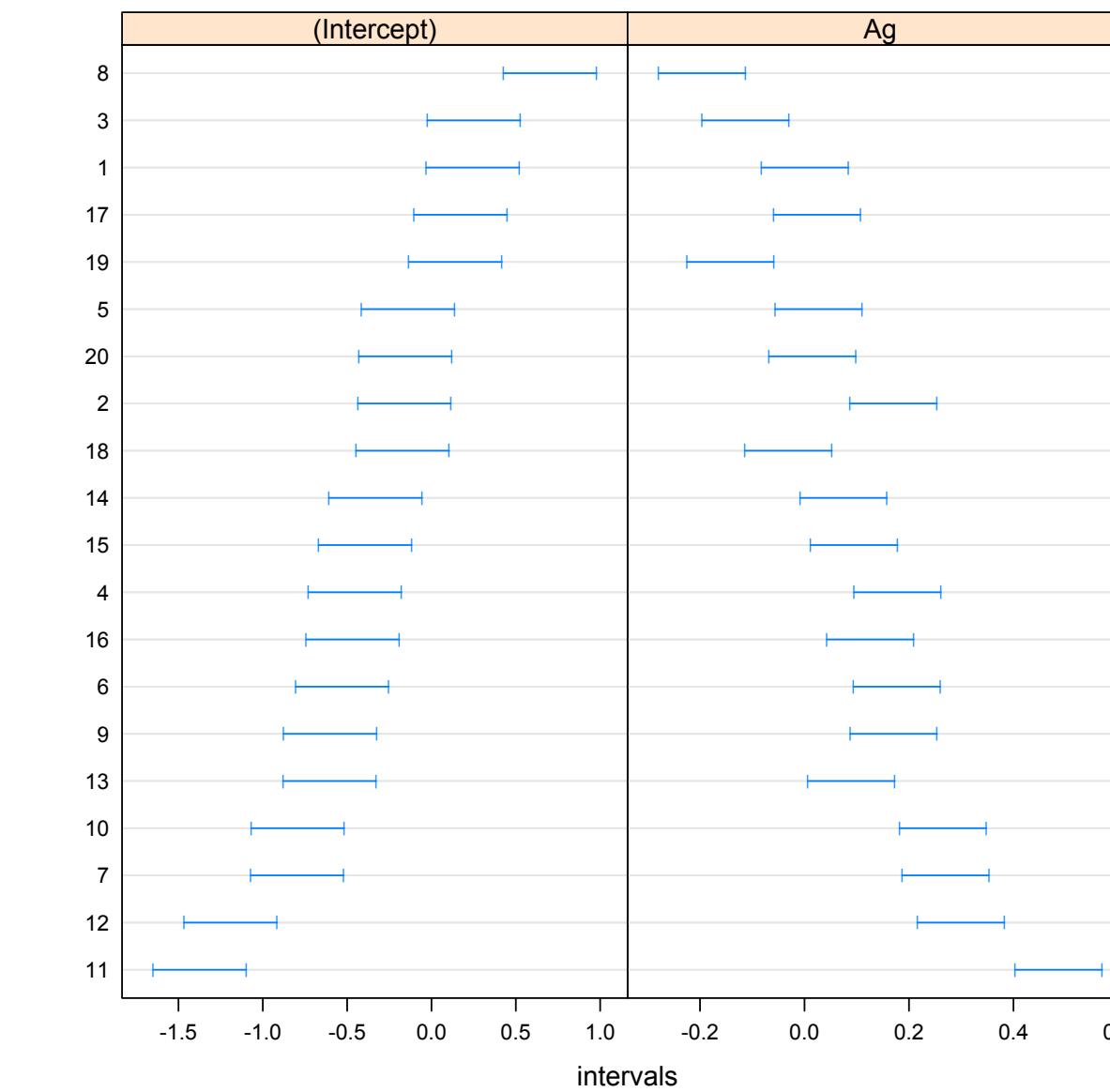
```
offMassMod1 <- lmer(logM ~ 1 + Ag + (1 + Ag | Id), offMassData)
offMassMod3 <- lmer(logM ~ 1 + Ag + (1 | Id), offMassData)
```

```
anova(offMassMod1, offMassMod4, refit = FALSE) # yes
```



PREVENT ML, ‘cause Random Test

Testing “random” terms



```
## Is the correlation term significant?
```

```
offMassMod1 <- lmer(logM ~ 1 + Ag + (1 + Ag | Id), offMassData)
offMassMod4 <- lmer(logM ~ 1 + Ag + (1 + Ag || Id), offMassData)
```

```
anova(offMassMod1, offMassMod4, refit = FALSE) # yes
```



PREVENT ML, ‘cause Random Test

Bootstrapped p-values— Fixed effects

- Much better for small or medium sized studies (does not rely on large sample assumptions), i.e. ameliorates the anti-conservative behaviour of LRTs
- Implemented by the `PBmodcomp` function in the `pbkrtest` package
- <http://www.jstatsoft.org/v59/i09/paper>

Bootstrapped p-values— Fixed effects

Exercise: Use the `PBmodcomp` function in the `pbkrtest` package to assess whether the fixed “age” effect on Soay litter mass is significantly different from zero

Hint: You need to install the package and then check the help file first.

Results

```
> library(pbkrtest)
> PBmodcomp(offMassMod1, offMassMod2, nsim = 1000)
```

Output:

```
Parametric bootstrap test; time: 46.29 sec; samples: 1000 extremes: 0;
large : logM ~ 1 + Ag + (1 + Ag | Id)
small : logM ~ 1 + (1 + Ag | Id)
      stat df   p.value
LRT    17.098  1 3.549e-05 ***
PBtest 17.098        0.000999 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

A final remark about p-values

- Life is much easier if you can design an experiment with “nice” properties, i.e. one that is balanced in some way.
- Then you can ditch `lmer` and just work with exact F tests (standard ANOVA, via the `aov` function in R).
- Consult a statistician first though unless phrases like “error strata” mean anything to you.

R Help

< > Print

Help Search

confint.merMod {lme4}

R Documentation

Description

Compute confidence intervals on the parameters of an lme4 fit

Usage

```
## S3 method for class 'merMod'
confint(object, parm, level = 0.95,
        method = c("profile", "Wald", "boot"), zeta,
        nsim = 500, boot.type = "perc", quiet = FALSE,
        oldNames = TRUE, ...)
```

Arguments

- object a fitted [ng]lmer model
- parm parameters (specified by integer position)
- level confidence level
- method for computing confidence intervals
- zeta likelihood cutoff (if not specified, computed from `level`: "profile" only)
- nsim number of simulations for parametric bootstrap intervals
- boot.type bootstrap confidence interval type
- quiet (logical) suppress messages about computationally intensive profiling?
- oldNames (logical) use old-style names for `method="profile"`? (See `signames` argument to [profile.merMod](#))
- ... additional parameters to be passed to [profile.merMod](#) or [bootMer](#)

Details

Depending on the method specified, this function will compute confidence intervals by ("profile") computing profile and finding the appropriate cutoffs based on the likelihood ratio test; ("Wald") approximate the confidence intervals for fixed-effect parameters only based on the estimated local curvature of the likelihood surface; ("boot") perform parametric bootstrapping with confidence intervals computed from the bootstrap distribution according to `boot.type`.

Value

... Confidence intervals

... Confidence Intervals

Alternative to p-values...

R Help

< > Print

Help Search

pvalues {lme4}

R Documentation

Description

Getting p-values for fitted models

One of the most frequently asked questions about `lme4` is "how do I calculate p-values for estimated parameters?" Previous versions of `lme4` provided the `mcmcSamp` function, which efficiently generated a Markov chain Monte Carlo sample from the posterior distribution of the parameters, assuming flat (scaled likelihood) priors. Due to difficulty in constructing a version of `mcmcSamp` that was reliable even in cases where the estimated random effect variances were near zero (e.g. <https://stat.ethz.ch/pipermail/r-sig-mixed-models/2009q4/003115.html>), `mcmcSamp` has been withdrawn (or more precisely, not updated to work with `lme4` versions $\geq 1.0.0$).

Many users, including users of the `aovLmer.fnc` function from the `languageR` package which relies on `mcmcSamp`, will be deeply disappointed by this lacuna. Users who need p-values have a variety of options:

- likelihood ratio tests via `anova` (MC,+)
- profile confidence intervals via [profile.merMod](#) and [confint.merMod](#) (CI,+)
- parametric bootstrap confidence intervals and model comparisons via [bootMer](#) (or `PBmodcomp` in the `pbkrtest` package) (MC/CI,*,+)
- for random effects, simulation tests via the `RLRsim` package (MC,*)
- for fixed effects, F tests via Kenward-Roger approximation using `KRmodcomp` from the `pbkrtest` package (MC)
- `car::Anova` and `lmerTest::anova` provide wrappers for `pbkrtest`; `lmerTest::anova` also provides t tests via the Satterthwaite approximation (P,*)

In the list above, the methods marked MC provide explicit model comparisons; CI denotes confidence intervals; and P denotes parameter-level or sequential tests of all effects in a model. The starred (*) suggestions provide finite-size corrections (important when the number of groups is < 50); those marked (+) support GLMMs as well as LMMs.

When all else fails, don't forget to keep p-values in perspective: <http://www.phdcomics.com/comics/archive.php?comicid=905>

[Package `lme4` version 1.1-0 [Index](#)]

Confidence Intervals

There are multiple ways to calculate confidence intervals for mixed models. The `confint` function in `lme4` implements these.

Exercise: Check the help for the `confint` function (`?confint.merMod`), then use this to construct “Wald” and “profile” confidence intervals for the litter mass model

```
> confint(offMassMod1, method="Wald")
              2.5 %    97.5 %
.sig01          NA        NA
.sig02          NA        NA
.sig03          NA        NA
.sigma          NA        NA
(Intercept) -0.2697198 0.1717798
Ag            0.1133662 0.2562938
```

```
> confint(offMassMod1, method="profile")
Computing profile confidence intervals ...
              2.5 %    97.5 %
.sig01      0.27322374 0.6499094
.sig02     -0.98314565 -0.8078543
.sig03      0.09348089 0.2123072
.sigma      0.19597269 0.2806440
(Intercept) -0.27488741 0.1769474
Ag          0.11169327 0.2579667
```

Can You Identify
What is What?

ANNOTATE!!!

```
> offMassMod1
```

Random effects:

Groups	Name	Variance	Std.Dev.	Corr
Id	(Intercept)	0.19449	0.4410	
	Ag	0.02121	0.1456	-0.94
Residual		0.05383	0.2320	

Number of obs: 100, groups: Id, 20

Fixed effects:

	Estimate	Std. Error	t value
(Intercept)	-0.04897	0.11263	-0.435
Ag	0.18483	0.03646	5.069

> confint(offMassMod1, method='profile')

	2.5 %	97.5 %
.sig01	0.27322374	0.6499094
.sig02	-0.98314565	-0.8078543
.sig03	0.09348089	0.2123072
.sigma	0.19597269	0.2806440
(Intercept)	-0.27488741	0.1769474
Ag	0.11169327	0.2579667