

Mixed / Multilevel models

Example dataset: trees

- Data on 1000 trees from 10 plots.

```
head(trees)
```

	plot	dbh	height	sex	dead	dbh.c
1	2	38.85	37.8	female	0	13.85
2	4	26.05	38.1	female	0	1.05
3	5	42.66	50.2	female	0	17.66
4	2	20.72	30.1	female	0	-4.28
5	4	21.83	34.0	female	0	-3.17
6	4	8.23	21.9	male	0	-16.77

Example dataset: trees

- ▶ Data on 1000 trees from 10 plots.
- ▶ Trees per plot: 4 - 392.

```
head(trees)
```

	plot	dbh	height	sex	dead	dbh.c
1	2	38.85	37.8	female	0	13.85
2	4	26.05	38.1	female	0	1.05
3	5	42.66	50.2	female	0	17.66
4	2	20.72	30.1	female	0	-4.28
5	4	21.83	34.0	female	0	-3.17
6	4	8.23	21.9	male	0	-16.77

Q: What's the relationship between tree diameter and height?

A simple linear model

```
lm.simple <- lm(height ~ dbh, data = trees)
```

Call:

```
lm(formula = height ~ dbh, data = trees)
```

Residuals:

Min	1Q	Median	3Q	Max
-13.7384	-4.7652	0.4759	4.2931	13.5282

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	13.18767	0.41476	31.80	<2e-16 ***
dbh	0.60967	0.01351	45.14	<2e-16 ***

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

Residual standard error: 5.549 on 998 degrees of freedom

Multiple R-squared: 0.6712, Adjusted R-squared: 0.6709

F-statistic: 2038 on 1 and 998 DF, p-value: < 2.2e-16

Remember our model structure

$$y_i \sim N(\mu_i, \sigma^2)$$

$$\mu_i = \alpha + \beta x_i$$

In this case:

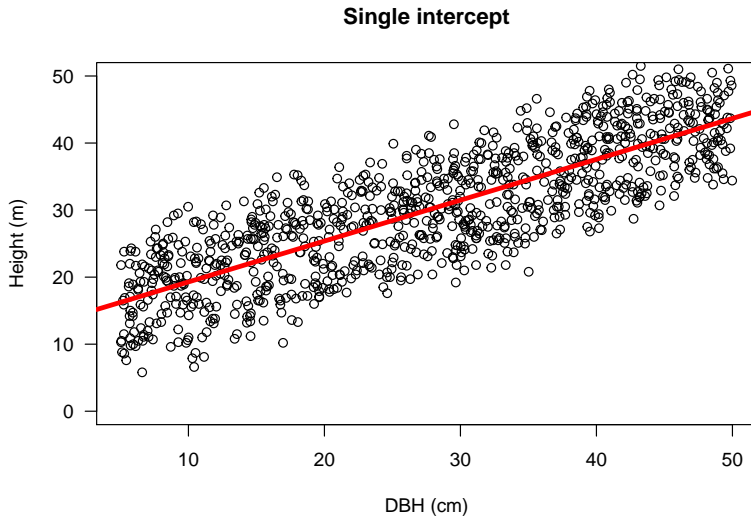
$$Height_i \sim N(\mu_i, \sigma^2)$$

$$\mu_i = \alpha + \beta DBH_i$$

α : expected height when $DBH = 0$

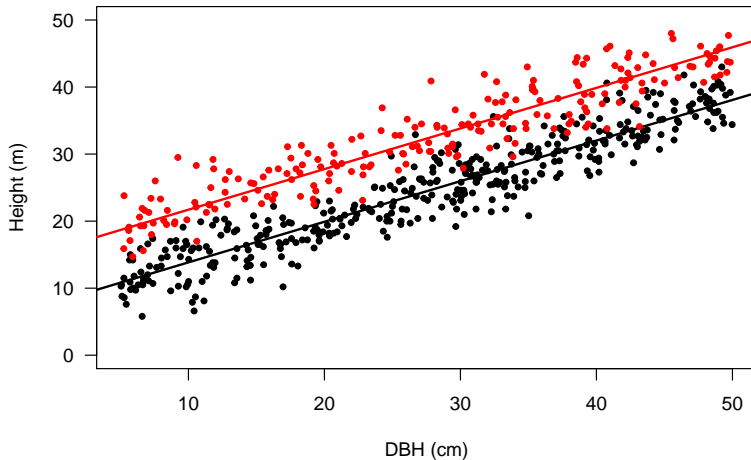
β : how much height increases with every unit increase of DBH

There is only one intercept



What if allometry varies among plots?

Different intercept for each plot



Fitting a varying intercepts model with `lm`

```
lm(formula = height ~ factor(plot) + dbh, data = trees)
```

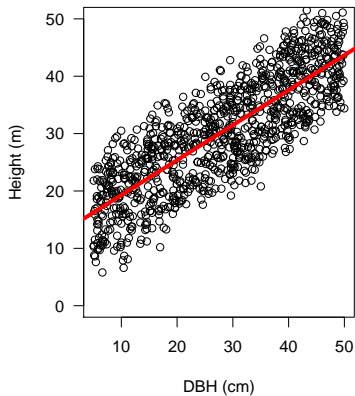
	coef.est	coef.se
(Intercept)	7.79	0.24
factor(plot)2	7.86	0.24
factor(plot)3	7.95	0.32
factor(plot)4	11.48	0.33
factor(plot)5	11.05	0.32
factor(plot)6	11.55	0.43
factor(plot)7	7.41	0.63
factor(plot)8	3.05	0.97
factor(plot)9	9.73	1.45
factor(plot)10	-0.14	0.92
dbh	0.61	0.01

```
n = 1000, k = 11
```

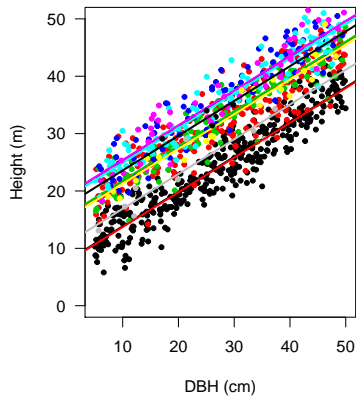
```
residual sd = 2.89, R-Squared = 0.91
```

Single vs varying intercept

Pooling all plots



Different intercept for each plot



Mixed models enable us to account for variability

► Varying intercepts

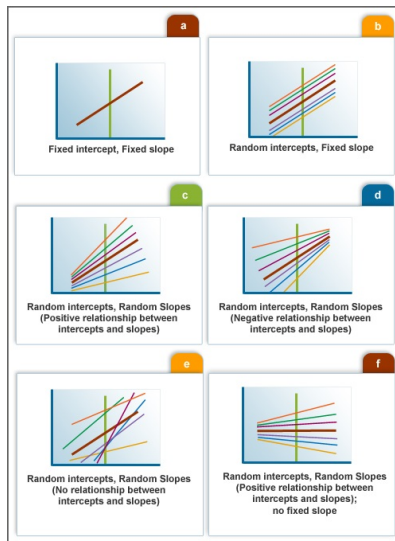


Figure 1:

Mixed models enable us to account for variability

- Varying intercepts
- Varying slopes

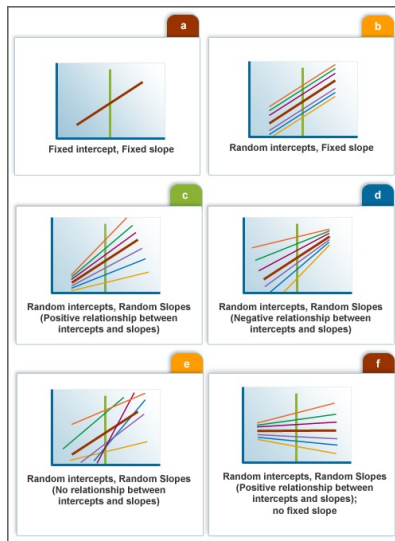


Figure 1:

Mixed model with varying intercepts

$$y_i = a_j + bx_i + \varepsilon_i$$

$$a_j \sim N(0, \tau^2)$$

$$\varepsilon_i \sim N(0, \sigma^2)$$

En nuestro ejemplo:

$$Height_i = plot_j + bDBH_i + \varepsilon_i$$

$$plot_j \sim N(0, \tau^2)$$

$$\varepsilon_i \sim N(0, \sigma^2)$$

Mixed models estimate varying parameters (intercepts and/or slopes) with pooling among levels (rather than considering them fully independent)

Hence there's gradient between

- ▶ **complete pooling**: Single overall intercept.

Hence there's gradient between

- ▶ **complete pooling:** Single overall intercept.
 - ▶ `lm (height ~ dbh)`

Hence there's gradient between

- ▶ **complete pooling**: Single overall intercept.
 - ▶ `lm (height ~ dbh)`
- ▶ **no pooling**: One *independent* intercept for each plot.

Hence there's gradient between

- ▶ **complete pooling:** Single overall intercept.
 - ▶ `lm (height ~ dbh)`
- ▶ **no pooling:** One *independent* intercept for each plot.
 - ▶ `lm (height ~ dbh + factor(plot))`

Hence there's gradient between

- ▶ **complete pooling**: Single overall intercept.
 - ▶ `lm (height ~ dbh)`
- ▶ **no pooling**: One *independent* intercept for each plot.
 - ▶ `lm (height ~ dbh + factor(plot))`
- ▶ **partial pooling**: Inter-related intercepts.

Hence there's gradient between

- ▶ **complete pooling:** Single overall intercept.
 - ▶ `lm (height ~ dbh)`
- ▶ **no pooling:** One *independent* intercept for each plot.
 - ▶ `lm (height ~ dbh + factor(plot))`
- ▶ **partial pooling:** Inter-related intercepts.
 - ▶ `lmer(height ~ dbh + (1 | plot))`

Random vs Fixed effects?

1. Fixed effects constant across individuals, random effects vary.

http://andrewgelman.com/2005/01/25/why_i_dont_use/

Random vs Fixed effects?

1. Fixed effects constant across individuals, random effects vary.
2. Effects are fixed if they are interesting in themselves; random if interest in the underlying population.

http://andrewgelman.com/2005/01/25/why_i_dont_use/

Random vs Fixed effects?

1. Fixed effects constant across individuals, random effects vary.
2. Effects are fixed if they are interesting in themselves; random if interest in the underlying population.
3. Fixed when sample exhausts the population; random when the sample is small part of the population.

http://andrewgelman.com/2005/01/25/why_i_dont_use/

Random vs Fixed effects?

1. Fixed effects constant across individuals, random effects vary.
2. Effects are fixed if they are interesting in themselves; random if interest in the underlying population.
3. Fixed when sample exhausts the population; random when the sample is small part of the population.
4. Random effect if it's assumed to be a realized value of random variable.

http://andrewgelman.com/2005/01/25/why_i_dont_use/

Random vs Fixed effects?

1. Fixed effects constant across individuals, random effects vary.
2. Effects are fixed if they are interesting in themselves; random if interest in the underlying population.
3. Fixed when sample exhausts the population; random when the sample is small part of the population.
4. Random effect if it's assumed to be a realized value of random variable.
5. Fixed effects estimated using least squares or maximum likelihood; random effects estimated with shrinkage.

http://andrewgelman.com/2005/01/25/why_i_dont_use/

What is a random effect, really?

1. Varies by group

Random effects are estimated with partial pooling, while fixed effects are not (infinite variance).

What is a random effect, really?

1. Varies by group
2. Variation estimated with probability model

Random effects are estimated with partial pooling, while fixed effects are not (infinite variance).

Shrinkage improves parameter estimation

Especially for groups with low sample size

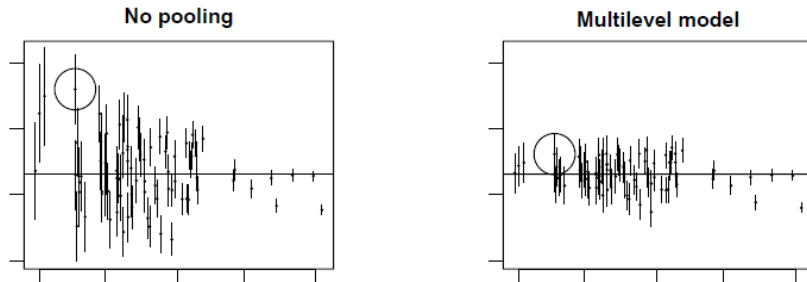


Figure 2:

From Gelman & Hill p. 253

Fitting mixed/multilevel models

```
library(lme4)
mixed <- lmer(height ~ dbh + (1|plot), data = trees)
```

```
lmer(formula = height ~ dbh + (1 | plot), data = trees)

               coef.est coef.se
(Intercept)  14.80      1.44
dbh           0.61      0.01
```

Error terms:

Groups	Name	Std.Dev.
plot	(Intercept)	4.45
Residual		2.89

```
number of obs: 1000, groups: plot, 10
AIC = 5015.6, DIC = 4996.4
deviance = 5002.0
```

Retrieve model coefficients

```
coef(mixed)
```

```
$plot
```

	(Intercept)	dbh
1	7.798373	0.6056549
2	15.647613	0.6056549
3	15.735397	0.6056549
4	19.253661	0.6056549
5	18.819467	0.6056549
6	19.306574	0.6056549
7	15.197908	0.6056549
8	11.016485	0.6056549
9	17.265448	0.6056549
10	7.940715	0.6056549

```
attr(,"class")
```

```
[1] "coef.mer"
```

Broom: model estimates in tidy form

```
library(broom)
tidy(mixed)
```

```
# A tibble: 4 x 5
```

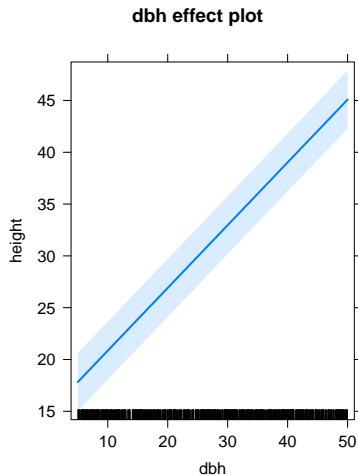
	term <chr>	estimate <dbl>	std.error <dbl>	statistic <dbl>	group <chr>
1	(Intercept)	14.8	1.44	10.3	fixed
2	dbh	0.606	0.00704	86.0	fixed
3	sd_(Intercept).plot	4.45	NA	NA	plot
4	sd_Observation.Residual	2.89	NA	NA	Residual

Visualising model: allEffects

```
model: height ~ dbh
```

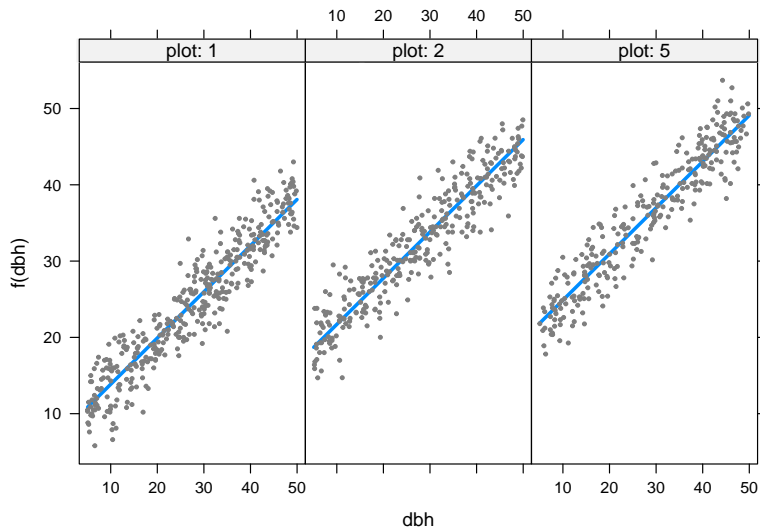
```
dbh effect
```

dbh	5	20	30	40	50
	17.82644	26.91126	32.96781	39.02436	45.08091



Visualising model: visreg

```
visreg(mixed, xvar = "dbh", by = "plot", re.form = ~(1|plot))
```

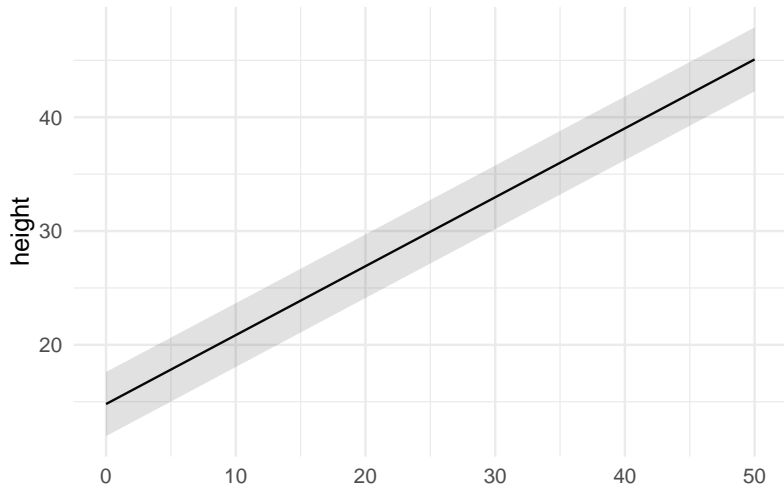


Visualising model: sjPlot

```
plot_model(mixed, type = "eff")
```

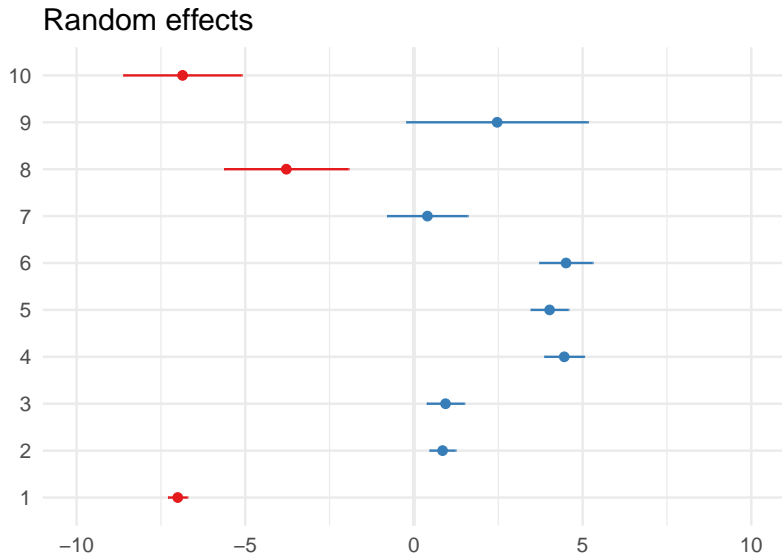
\$dbh

Predicted values of height

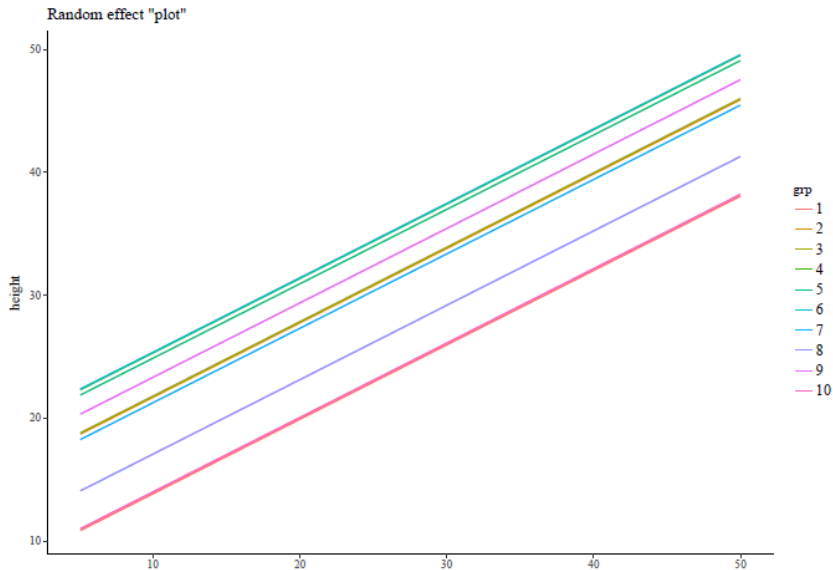


Visualising model: sjPlot

```
plot_model(mixed, type = "re")
```



Visualising model



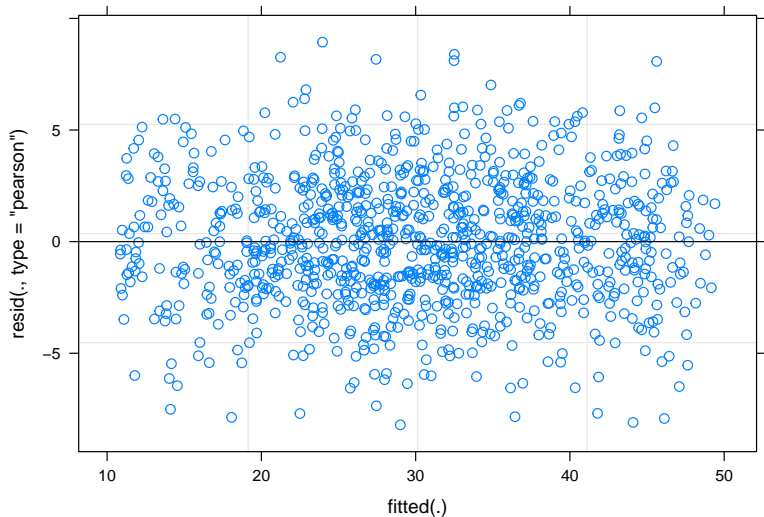
Using merTools to understand fitted model

```
library(merTools)
shinyMer(mixed)
```

```
## Plotting regression for individual forest plots
nplot <- 2
plot(trees$dbh[trees$plot==nplot], trees$height[trees$plot==nplot])
abline(a=coef(mixed)$plot[nplot, 1], b=coef(mixed)$plot[nplot, 2])
```

Checking residuals

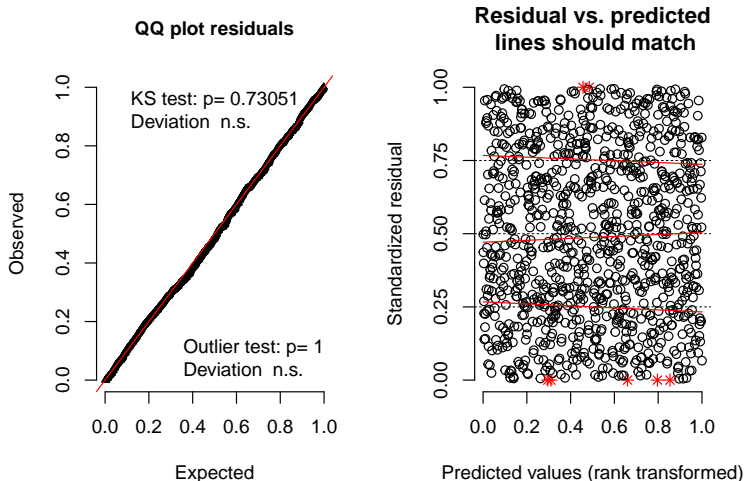
```
plot(mixed)
```



Checking residuals (DHARMA)

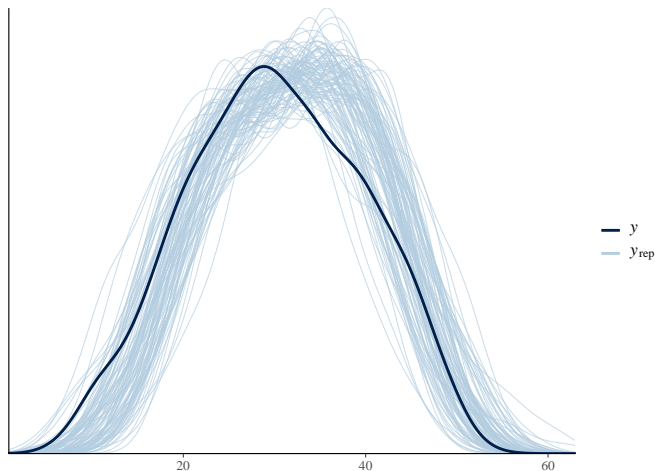
```
DHARMA::simulateResiduals(mixed, plot = TRUE, use.u = TRUE)
```

DHARMA scaled residual plots



Model checking with simulated data

```
library(bayesplot)
sims <- simulate(mixed, nsim = 100)
ppc_dens_overlay(trees$height, yrep = t(as.matrix(sims)))
```



R-squared for GLMMs

Many approaches! Somewhat polemic.

Nakagawa & Schielzeth propose **marginal** (considering fixed effects only) and **conditional** R^2 (including random effects too):

```
library(MuMIn)
r.squaredGLMM(mixed)
```

	R2m	R2c
[1,]	0.687565	0.9076325

Growing the hierarchy: adding plot-level
predictors

Model with group-level predictors

We had:

$$y_i = a_j + bx_i + \varepsilon_i$$

$$a_j \sim N(0, \tau^2)$$

$$\varepsilon_i \sim N(0, \sigma^2)$$

Now

$$y_i = a_j + bx_i + \varepsilon_i$$

$$a_j \sim N(\mu_j, \tau^2)$$

$$\mu_j = \gamma + \delta \cdot \text{predictor}_j$$

$$\varepsilon_i \sim N(0, \sigma^2)$$

Merging trees and plot data

```
plotdata <- read.csv("data-raw/plotdata.csv")
trees.full <- merge(trees, plotdata, by = "plot")
head(trees.full)
```

	plot	dbh	height	sex	dead	dbh.c	temp
1	1	28.63	22.1	female	0	3.63	15.1
2	1	44.71	39.0	female	0	19.71	15.1
3	1	28.31	29.0	female	0	3.31	15.1
4	1	19.33	19.1	male	0	-5.67	15.1
5	1	9.25	12.2	female	0	-15.75	15.1
6	1	30.02	23.1	female	0	5.02	15.1

Centre continuous variables

Plot temperatures referred as deviations from 15°C

```
trees.full$temp.c <- trees.full$temp - 15
```

Fit multilevel model

```
group.pred <- lmer(height ~ dbh + (1 | plot) + temp.c, data = tr  
arm::display(group.pred)
```

```
lmer(formula = height ~ dbh + (1 | plot) + temp.c, data = trees.
```

	coef.est	coef.se
(Intercept)	11.79	1.75
dbh	0.61	0.01
temp.c	1.07	0.46

Error terms:

Groups	Name	Std.Dev.
plot	(Intercept)	3.61
Residual		2.89

number of obs: 1000, groups: plot, 10

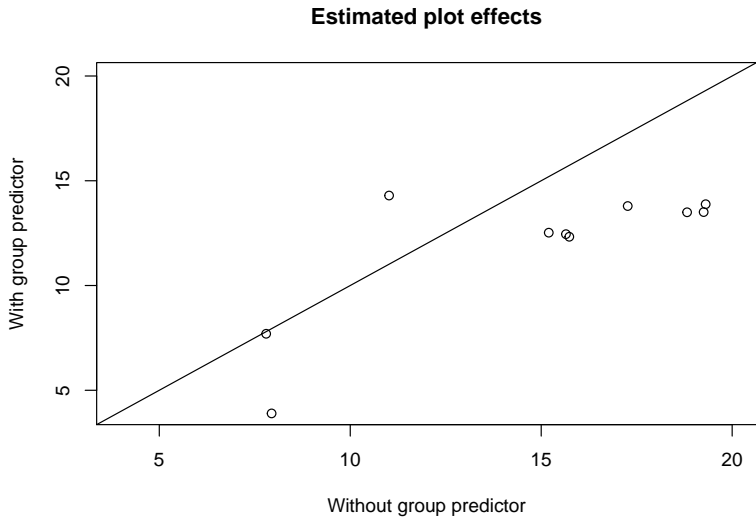
AIC = 5012.8, DIC = 4991

deviance = 4996.9

Examine model with merTools

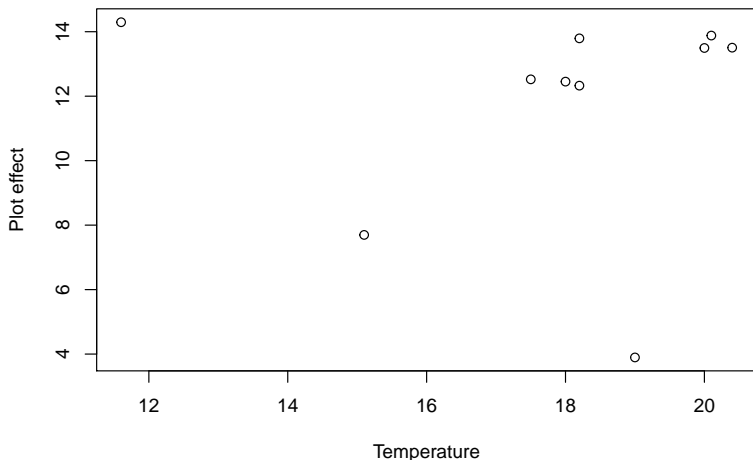
```
shinyMer(group.pred)
```

Comparing plot effects with and without group predictor



Are plot effects related to temperature?

```
plot(plotdata$temp, coef(group.pred)$plot[,1],  
     xlab = "Temperature", ylab = "Plot effect")
```



Varying intercepts and slopes

Varying intercepts and slopes

- ▶ There is overall difference in height among plots (different intercepts)

```
mixed.slopes <- lmer(height ~ dbh + (1 + dbh | plot), data=trees)
```

Varying intercepts and slopes

- ▶ There is overall difference in height among plots (different intercepts)
- ▶ AND

```
mixed.slopes <- lmer(height ~ dbh + (1 + dbh | plot), data=trees)
```

Varying intercepts and slopes

- ▶ There is overall difference in height among plots (different intercepts)
- ▶ AND
- ▶ Relationship between DBH and Height varies among plots (different slopes)

```
mixed.slopes <- lmer(height ~ dbh + (1 + dbh | plot), data=trees)
```

Varying intercepts and slopes

```
lmer(formula = height ~ dbh + (1 + dbh | plot), data = trees)
```

	coef.est	coef.se
(Intercept)	14.82	1.30
dbh	0.61	0.01

Error terms:

Groups	Name	Std.Dev.	Corr
plot	(Intercept)	3.98	
	dbh	0.01	-0.29
Residual		2.89	

number of obs: 1000, groups: plot, 10

AIC = 5018.9, DIC = 4995.6

deviance = 5001.3

Varying intercepts and slopes

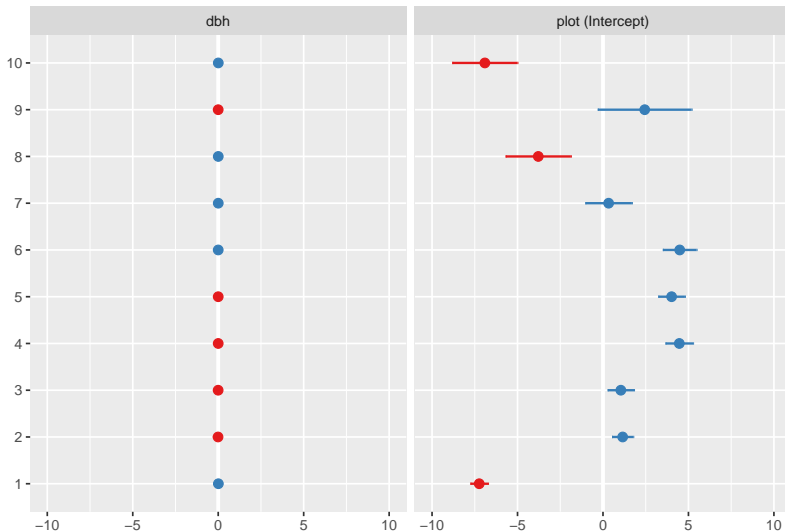
```
$plot
      (Intercept)      dbh
1      7.577721 0.6136944
2     15.969067 0.5941570
3     15.862676 0.6010346
4     19.282902 0.6043557
5     18.832500 0.6049803
6     19.307373 0.6051534
7     15.148412 0.6071041
8     11.032456 0.6072821
9     17.259351 0.6032577
10     7.908138 0.6092241
```

```
attr(,"class")
[1] "coef.mer"
```

Visualising model: sjPlot

```
plot_model(mixed.slopes, type = "re")
```

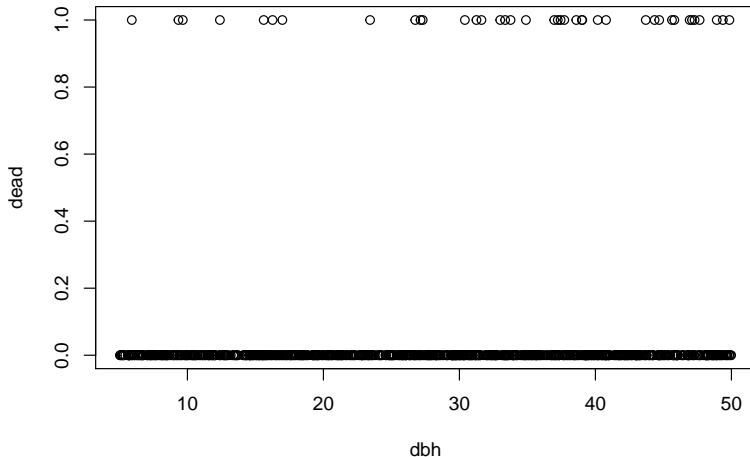
Random effects



Multilevel logistic regression

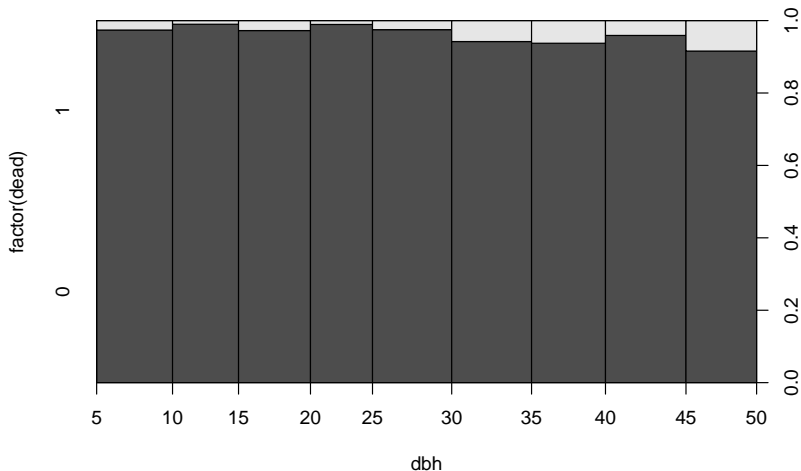
Q: Relationship between tree size and mortality

```
plot(dead ~ dbh, data = trees)
```



Q: Relationship between tree size and mortality

```
plot(factor(dead) ~ dbh, data = trees)
```



Fit simple logistic regression

```
simple.logis <- glm(dead ~ dbh, data = trees, family=binomial)
```

Call:

```
glm(formula = dead ~ dbh, family = binomial, data = trees)
```

Deviance Residuals:

Min	1Q	Median	3Q	Max
-0.4121	-0.3287	-0.2624	-0.2048	2.9127

Coefficients:

	Estimate	Std. Error	z value	Pr(> z)
(Intercept)	-4.46945	0.49445	-9.039	< 2e-16 ***
dbh	0.04094	0.01380	2.967	0.00301 **

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

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 329.51 on 999 degrees of freedom

Logistic regression with *independent* plot effects

```
logis2 <- glm(dead ~ dbh + factor(plot), data = trees, family=bi
```

Call:

```
glm(formula = dead ~ dbh + factor(plot), family = binomial, data
```

Deviance Residuals:

Min	1Q	Median	3Q	Max
-0.5923	-0.3198	-0.2549	-0.1940	2.8902

Coefficients:

	Estimate	Std. Error	z value	Pr(> z)	
(Intercept)	-4.40106	0.52997	-8.304	<2e-16	***
dbh	0.04060	0.01386	2.929	0.0034	**
factor(plot)2	-0.59168	0.52132	-1.135	0.2564	
factor(plot)3	0.54576	0.47094	1.159	0.2465	
factor(plot)4	0.05507	0.57434	0.096	0.9236	
factor(plot)5	-0.38312	0.64222	-0.597	0.5508	
factor(plot)6	-0.08426	0.76908	-0.110	0.9128	
factor(plot)7	0.03126	1.06064	0.029	0.9765	

Fit multilevel logistic regression

```
mixed.logis <- glmer(dead ~ dbh + (1|plot), data=trees, family =
```

```
glmer(formula = dead ~ dbh + (1 | plot), data = trees, family =
```

	coef.est	coef.se
(Intercept)	-4.47	0.49
dbh	0.04	0.01

Error terms:

Groups	Name	Std.Dev.
plot	(Intercept)	0.00
Residual		1.00

number of obs: 1000, groups: plot, 10

AIC = 325.9, DIC = 319.9

deviance = 319.9

Retrieve model coefficients

```
coef(mixed.logis)
```

```
$plot
```

	(Intercept)	dbh
1	-4.469446	0.04093806
2	-4.469446	0.04093806
3	-4.469446	0.04093806
4	-4.469446	0.04093806
5	-4.469446	0.04093806
6	-4.469446	0.04093806
7	-4.469446	0.04093806
8	-4.469446	0.04093806
9	-4.469446	0.04093806
10	-4.469446	0.04093806

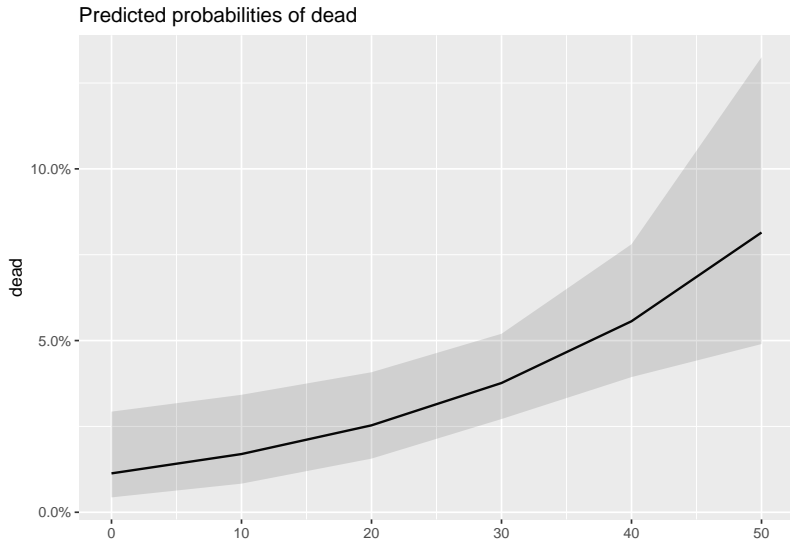
```
attr(,"class")
```

```
[1] "coef.mer"
```

Visualising model: sjPlot

```
plot_model(mixed.logis, type = "eff", show.ci = TRUE)
```

\$dbh



Advantages of multilevel models

- ▶ Perfect for structured data (space-time)

Advantages of multilevel models

- ▶ Perfect for structured data (space-time)
- ▶ Predictors enter at the appropriate level

Advantages of multilevel models

- ▶ Perfect for structured data (space-time)
- ▶ Predictors enter at the appropriate level
- ▶ Accommodate variation in treatment effects

Advantages of multilevel models

- ▶ Perfect for structured data (space-time)
- ▶ Predictors enter at the appropriate level
- ▶ Accommodate variation in treatment effects
- ▶ More efficient inference of regression parameters

Advantages of multilevel models

- ▶ Perfect for structured data (space-time)
- ▶ Predictors enter at the appropriate level
- ▶ Accommodate variation in treatment effects
- ▶ More efficient inference of regression parameters
- ▶ Using all the data to perform inferences for groups with small sample size

Formula syntax for different models

$y \sim x + (1 \mid \text{group})$ # varying intercepts

$y \sim x + (1 + x \mid \text{group})$ # varying intercepts and slopes

$y \sim x + (1 \mid \text{group/subgroup})$ # nested

$y \sim x + (1 \mid \text{group1}) + (1 \mid \text{group2})$ # varying intercepts, crossed

$y \sim x + (1 + x \mid \text{group1}) + (1 + x \mid \text{group2})$ # varying intercepts and slopes, crossed

More examples

- ▶ sleepstudy