

Mixed models introduction

O Rodriguez de Rivera Ortega, PhD
SE@K (Statistical Ecology @ Kent), University of Kent

10/11/2020

Example dataset: trees

- Data on 1000 trees from 10 sites.
- Trees per site: 4 - 392.

```
trees <- read.csv("~/Google Drive/Course/data/trees.csv")  
head(trees)
```

```
##   site   dbh height    sex dead  
## 1    4 29.68   36.1  male    0  
## 2    5 33.29   42.3  male    0  
## 3    2 28.03   41.9 female    0  
## 4    5 39.86   46.5 female    0  
## 5    1 47.94   43.9 female    0  
## 6    1 10.82   26.2  male    0
```

```
trees$site <- as.factor(trees$site)
```

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

A simple linear model

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

```
##  
## Call:  
## lm(formula = height ~ dbh, data = trees)  
##  
## Residuals:  
##      Min       1Q   Median       3Q      Max   
## -13.3270  -2.8978   0.1057   2.7924  12.9511   
##  
## Coefficients:  
##              Estimate Std. Error t value Pr(>|t|)      
## (Intercept)  19.33920    0.31064   62.26  <2e-16 ***  
## dbh          0.61570    0.01013   60.79  <2e-16 ***  
## ---  
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1  
##  
## Residual standard error: 4.093 on 998 degrees of freedom  
## Multiple R-squared:  0.7874, Adjusted R-squared:  0.7871
```

```
## F-statistic: 3695 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

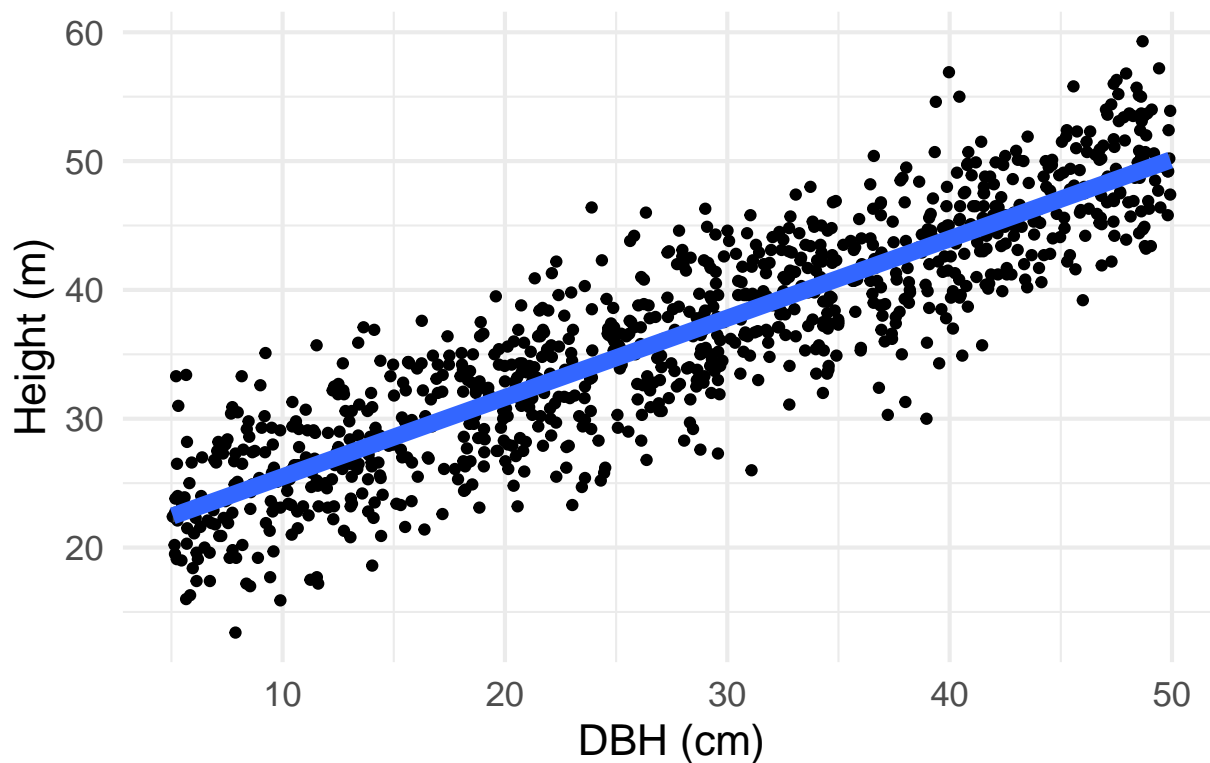
β : how much height increases with every unit increase of DBH

There is only one intercept

```
library(ggplot2)
ggplot(trees) +
  aes(dbh, height) +
  geom_point() +
  geom_smooth(method = "lm", size = 3) +
  labs(x = "DBH (cm)", y = "Height (m)", title = "Single intercept") +
  theme_minimal(base_size = 16)
```

```
## `geom_smooth()` using formula 'y ~ x'
```

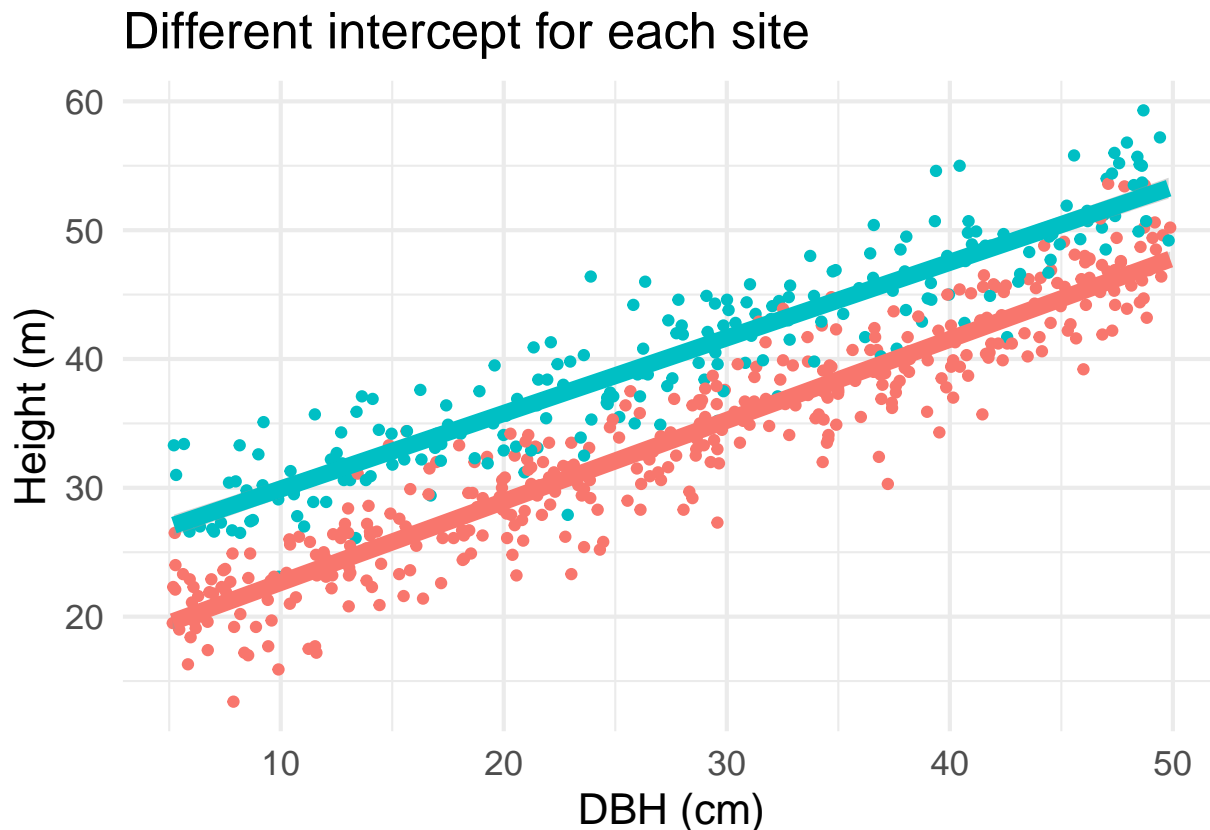
Single intercept



What if allometry varies among sites?

```
ggplot(subset(trees, site == 1 | site == 2)) +  
  aes(dbh, height, colour = site) +  
  geom_point() +  
  geom_smooth(method = "lm", size = 3) +  
  labs(x = "DBH (cm)", y = "Height (m)",  
       title = "Different intercept for each site") +  
  theme_minimal(base_size = 16) +  
  theme(legend.position = "none")
```

```
## `geom_smooth()` using formula 'y ~ x'
```



Fitting a varying intercepts model with lm

```
lm.interc <- lm(height ~ factor(site) + dbh, data = trees)  
summary(lm.interc)
```

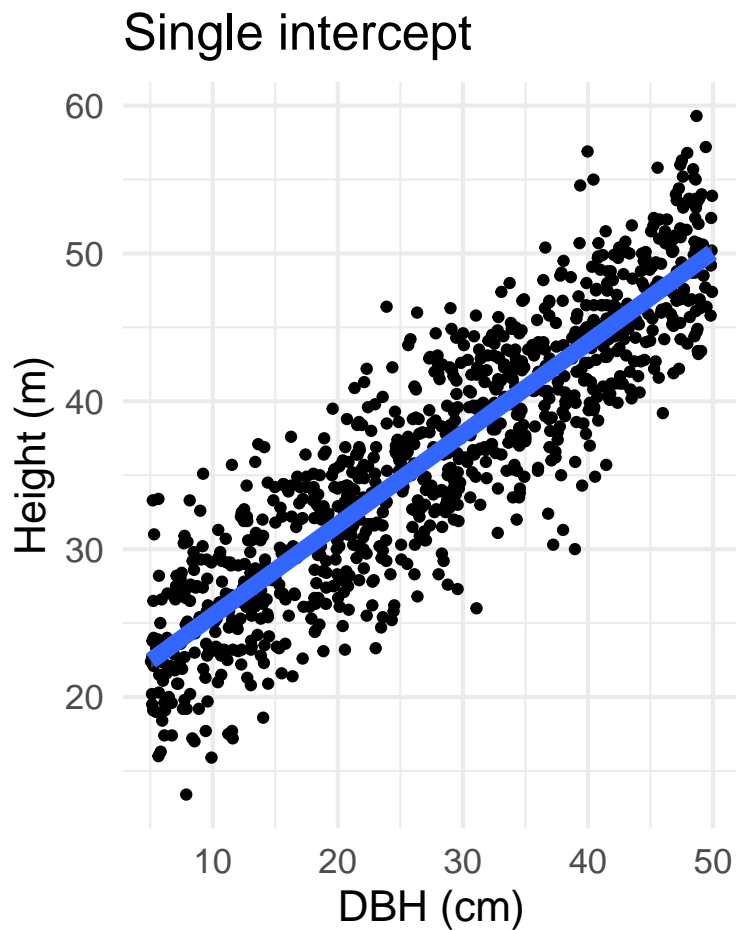
```
##  
## Call:  
## lm(formula = height ~ factor(site) + dbh, data = trees)  
##  
## Residuals:  
##      Min       1Q   Median       3Q      Max   
## -10.1130  -1.9885   0.0582   2.0314  11.3320   
##  
## Coefficients:  
##              Estimate Std. Error t value Pr(>|t|)      
## (Intercept)  16.699037   0.260565  64.088  < 2e-16 ***  
## factor(site)2   6.504303   0.256730  25.335  < 2e-16 ***  
## factor(site)3   4.357457   0.354181  12.303  < 2e-16 ***
```

```
## factor(site)4    1.934650    0.356102    5.433 6.98e-08 ***
## factor(site)5    3.637432    0.339688   10.708 < 2e-16 ***
## factor(site)6    4.204511    0.421906    9.966 < 2e-16 ***
## factor(site)7   -0.176193    0.666772   -0.264  0.7916
## factor(site)8   -5.312648    0.893603   -5.945 3.82e-09 ***
## factor(site)9    5.437049    1.087766    4.998 6.84e-07 ***
## factor(site)10   2.263338    1.369986    1.652  0.0988 .
## dbh              0.617075    0.007574   81.473 < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.043 on 989 degrees of freedom
## Multiple R-squared:  0.8835, Adjusted R-squared:  0.8823
## F-statistic: 750 on 10 and 989 DF, p-value: < 2.2e-16
```

Single vs varying intercept

```
ggplot(trees) +
  aes(dbh, height) +
  geom_point() +
  geom_smooth(method = "lm", size = 3) +
  labs(x = "DBH (cm)", y = "Height (m)", title = "Single intercept") +
  theme_minimal(base_size = 16)
```

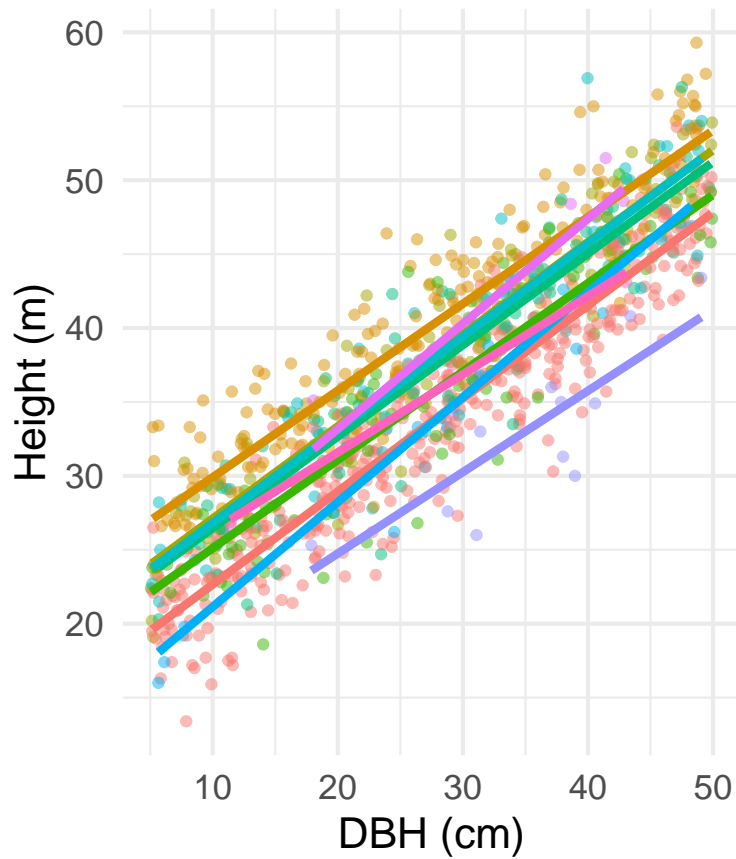
```
## `geom_smooth()` using formula 'y ~ x'
```



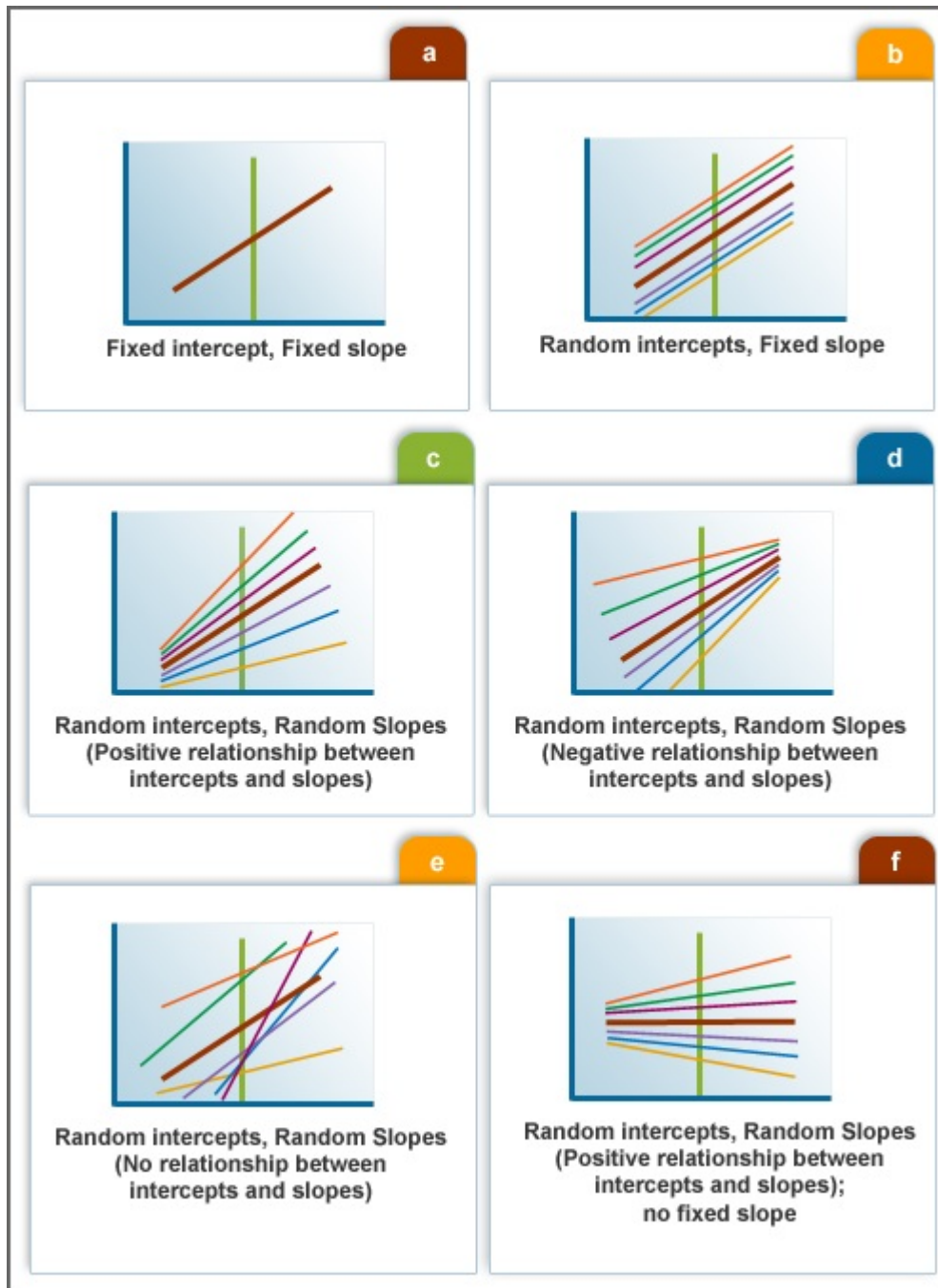
```
ggplot(trees) +
  aes(dbh, height, colour = site) +
  geom_point(alpha = 0.5) +
  geom_smooth(method = "lm", size = 1.5, se = FALSE) +
  labs(x = "DBH (cm)", y = "Height (m)",
       title = "Different intercept for each site") +
  theme_minimal(base_size = 16) +
  theme(legend.position = "none")
```

```
## `geom_smooth()` using formula 'y ~ x'
```

Different intercept for each si



Mixed models enable us to account for variability



- Varying intercepts
- Varying slopes

Mixed model with varying intercepts

$$y_i = a + \alpha_j + b \cdot x_i + \varepsilon_i$$

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

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

En nuestro ejemplo:

$$\begin{aligned} \text{Height}_i &= a + \text{site}_j + b \cdot \text{DBH}_i + \varepsilon_i \\ \text{site}_j &\sim N(0, \tau^2) \\ \varepsilon_i &\sim N(0, \sigma^2) \end{aligned}$$

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.
 - `lm (height ~ dbh)`
- **no pooling:** One *independent* intercept for each site.
 - `lm (height ~ dbh + site)`
- **partial pooling:** Inter-related intercepts.
 - `lmer(height ~ dbh + (1 | site))`

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.

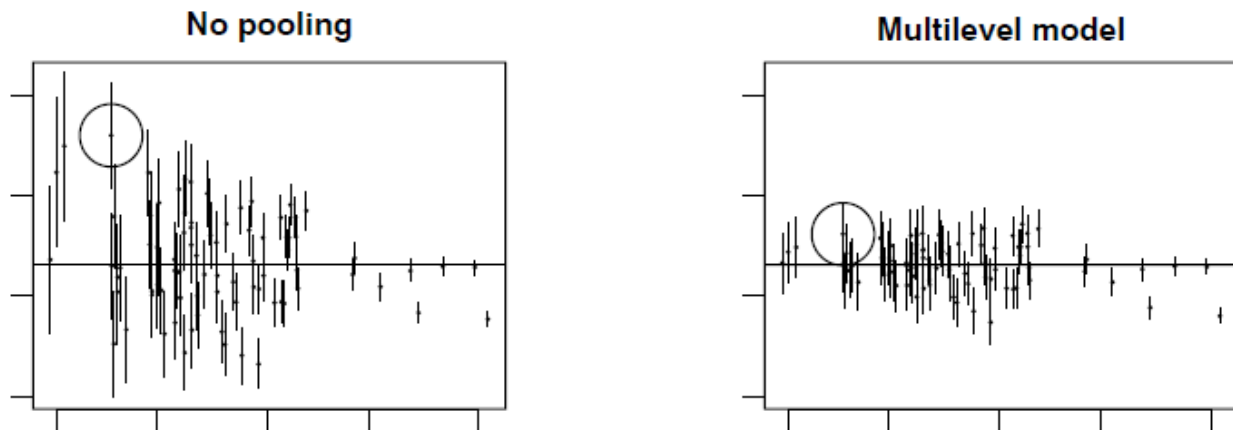
What is a random effect, really?

- Varies by group
- 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



From Gelman & Hill p. 253

Fitting mixed/multilevel models

```
library(lme4)

## Loading required package: Matrix
mixed <- lmer(height ~ dbh + (1|site), data = trees)
summary(mixed)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: height ~ dbh + (1 | site)
## Data: trees
##
## REML criterion at convergence: 5108.3
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -3.3199 -0.6607  0.0227  0.6716  3.7328
##
## Random effects:
## Groups Name Variance Std.Dev.
## site (Intercept) 11.195 3.346
## Residual 9.261 3.043
## Number of obs: 1000, groups: site, 10
##
## Fixed effects:
## Estimate Std. Error t value
## (Intercept) 19.011468 1.100444 17.28
## dbh 0.616927 0.007572 81.47
##
## Correlation of Fixed Effects:
## (Intr)
## dbh -0.197
```

Retrieve model coefficients

```
coef(mixed)

## $site
## (Intercept) dbh
## 1 16.70800 0.6169271
## 2 23.19162 0.6169271
## 3 21.04229 0.6169271
```



```
## 4      18.64086 0.6169271
## 5      20.32995 0.6169271
## 6      20.88200 0.6169271
## 7      16.61686 0.6169271
## 8      11.88302 0.6169271
## 9      21.84779 0.6169271
## 10     18.97228 0.6169271
##
## attr(,"class")
## [1] "coef.mer"
```

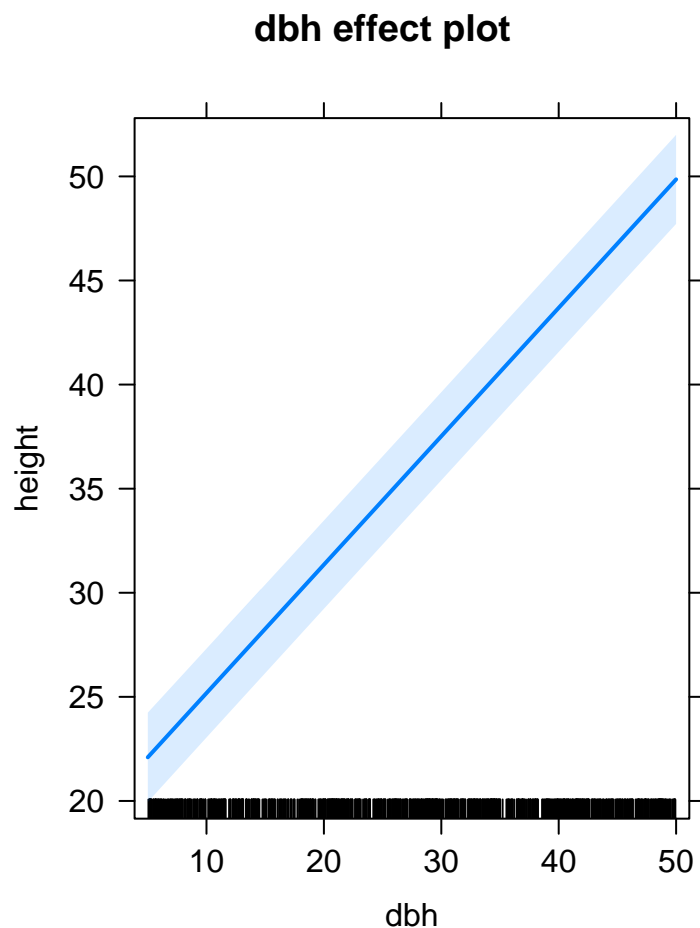
Visualising model: allEffects

```
library(effects)
```

```
## Loading required package: carData
## lattice theme set by effectsTheme()
## See ?effectsTheme for details.
```

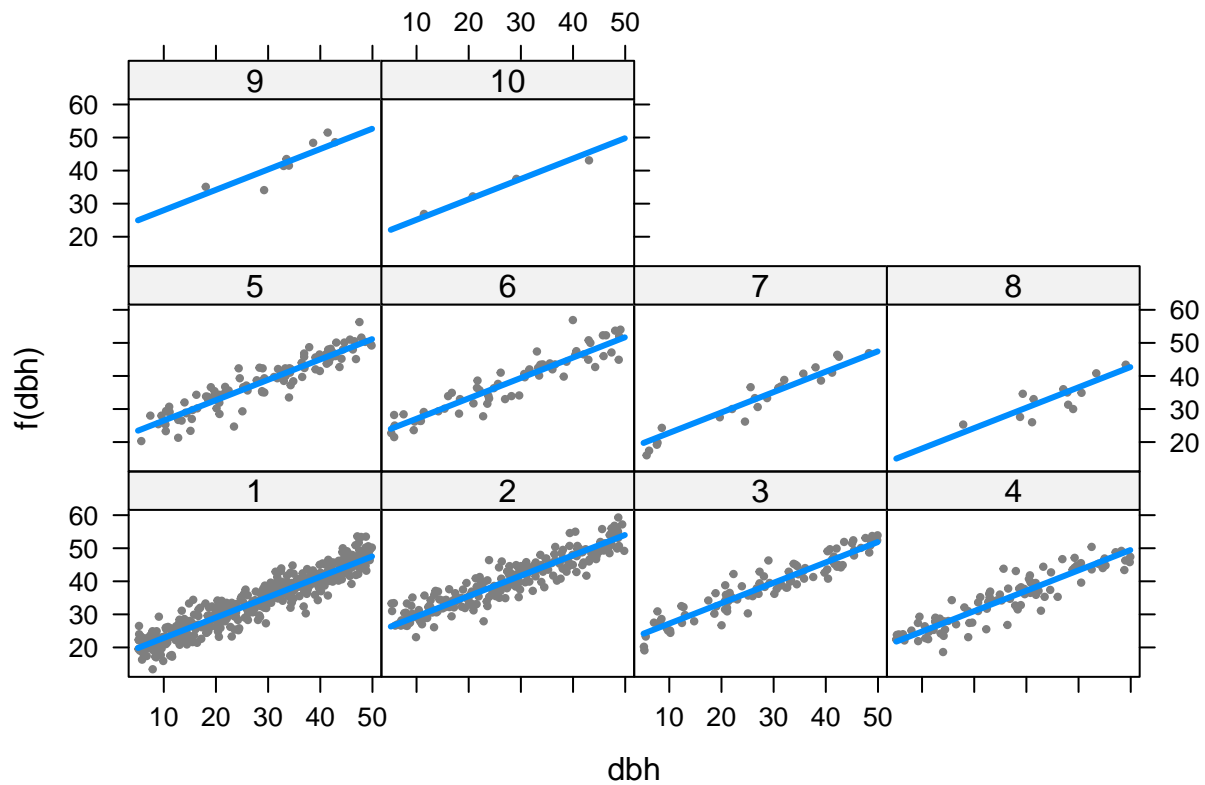
```
allEffects(mixed)
```

```
## model: height ~ dbh
##
## dbh effect
## dbh
##      5      20      30      40      50
## 22.09610 31.35001 37.51928 43.68855 49.85782
plot(allEffects(mixed))
```



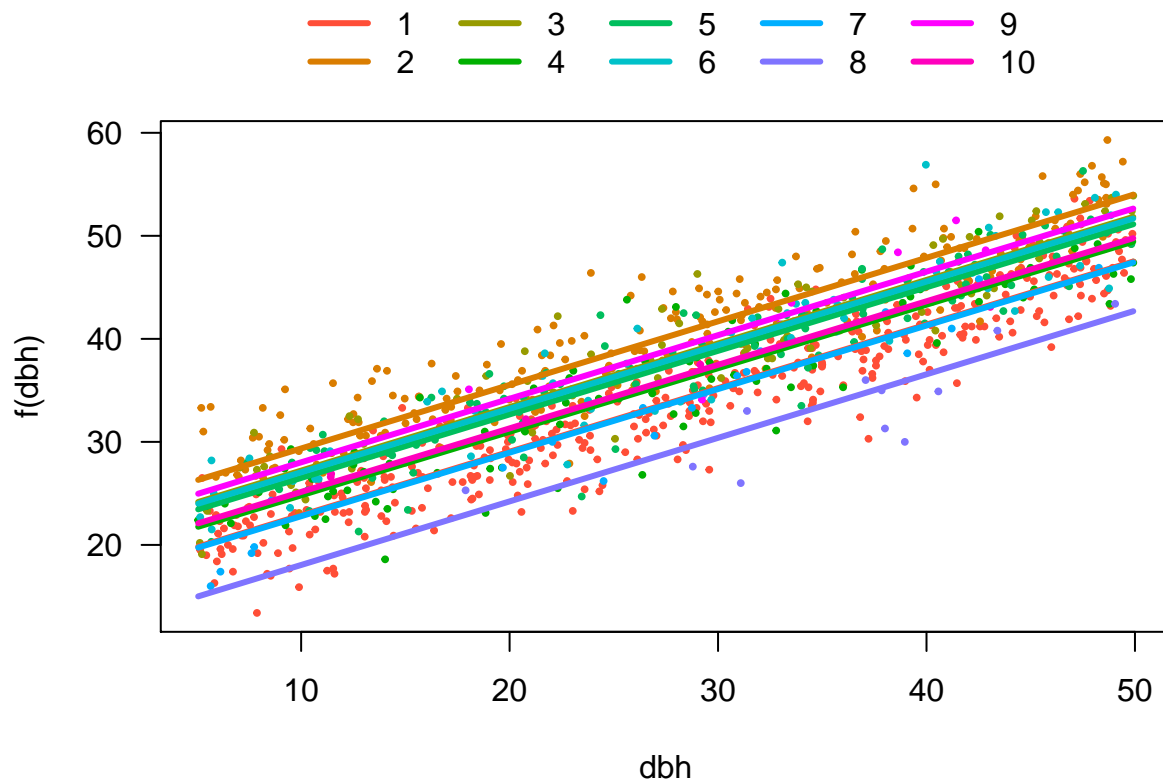
Visualising model: visreg

```
library(visreg)
visreg(mixed, xvar = "dbh", by = "site", re.form = NULL)
```



Visualising model

```
visreg(mixed, xvar = "dbh", by = "site", re.form = NULL, overlay = TRUE)
```



Visualising model: sjPlot

```
library(ggplot2)
library(sjPlot)
```

```
## Learn more about sjPlot with 'browseVignettes("sjPlot")'.
```

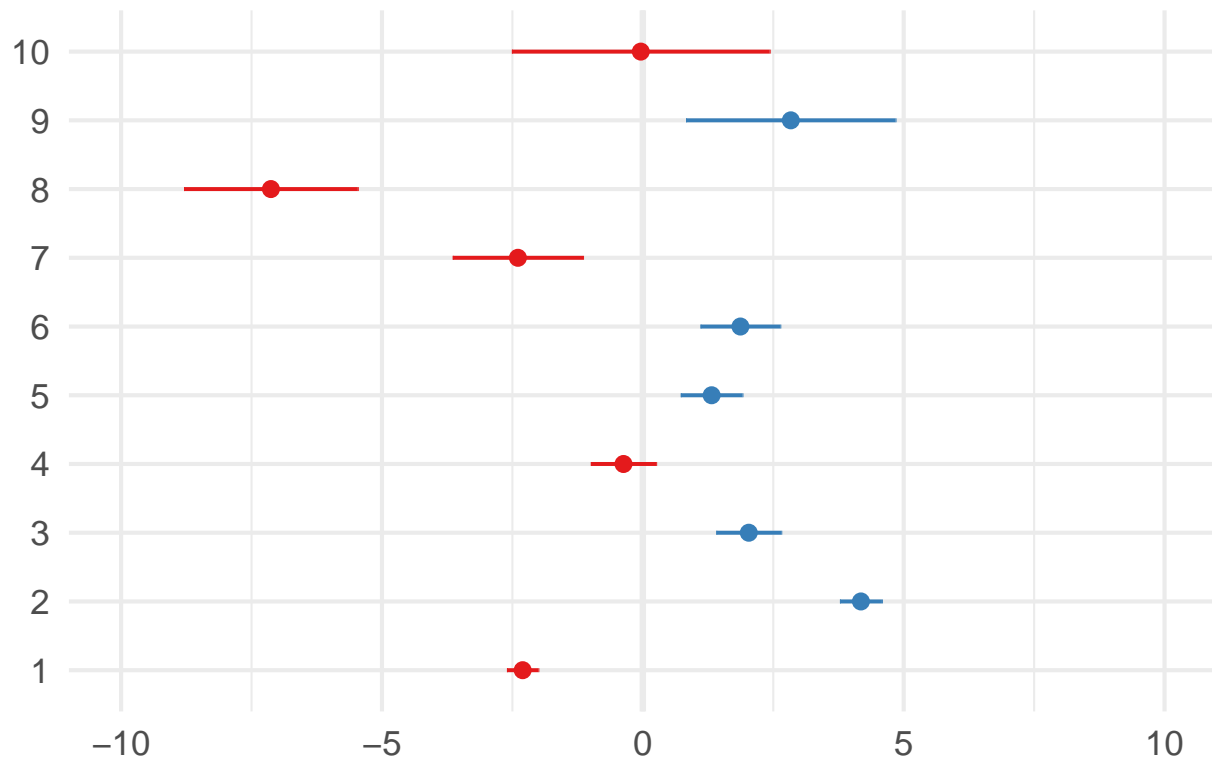
```
theme_set(theme_minimal(base_size = 16))
```

```
#sjp.lmer(mixed, type = "ri.slope")
```

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

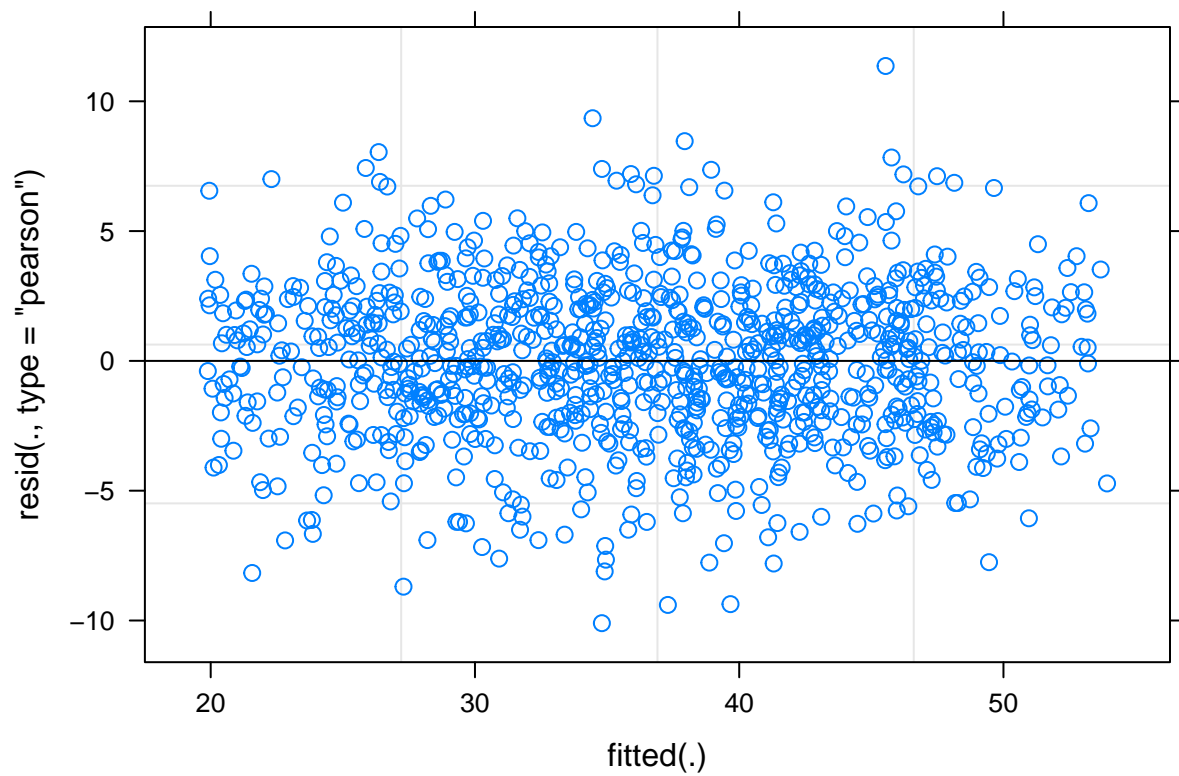
```
sjPlot::plot_model(mixed, type = "re")
```

Random effects



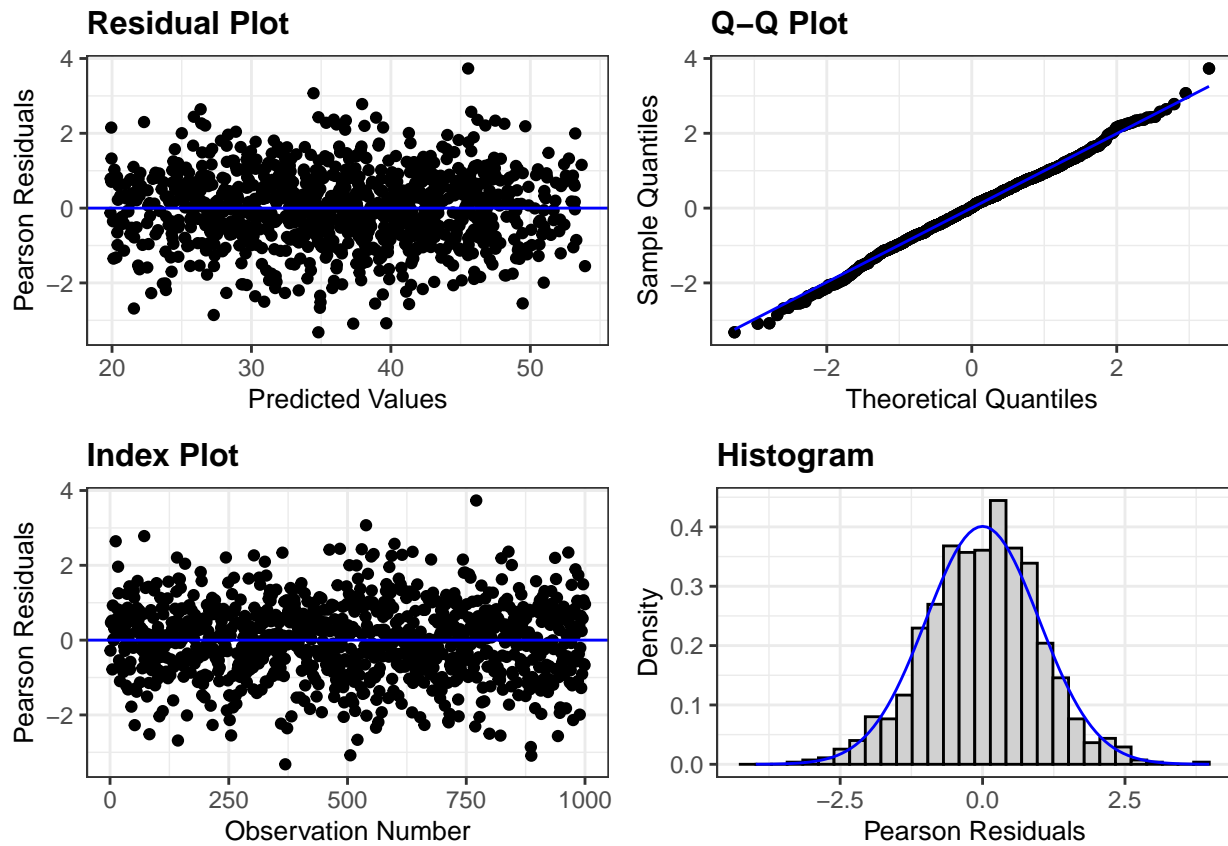
Checking residuals

```
plot(mixed)
```



Checking residuals

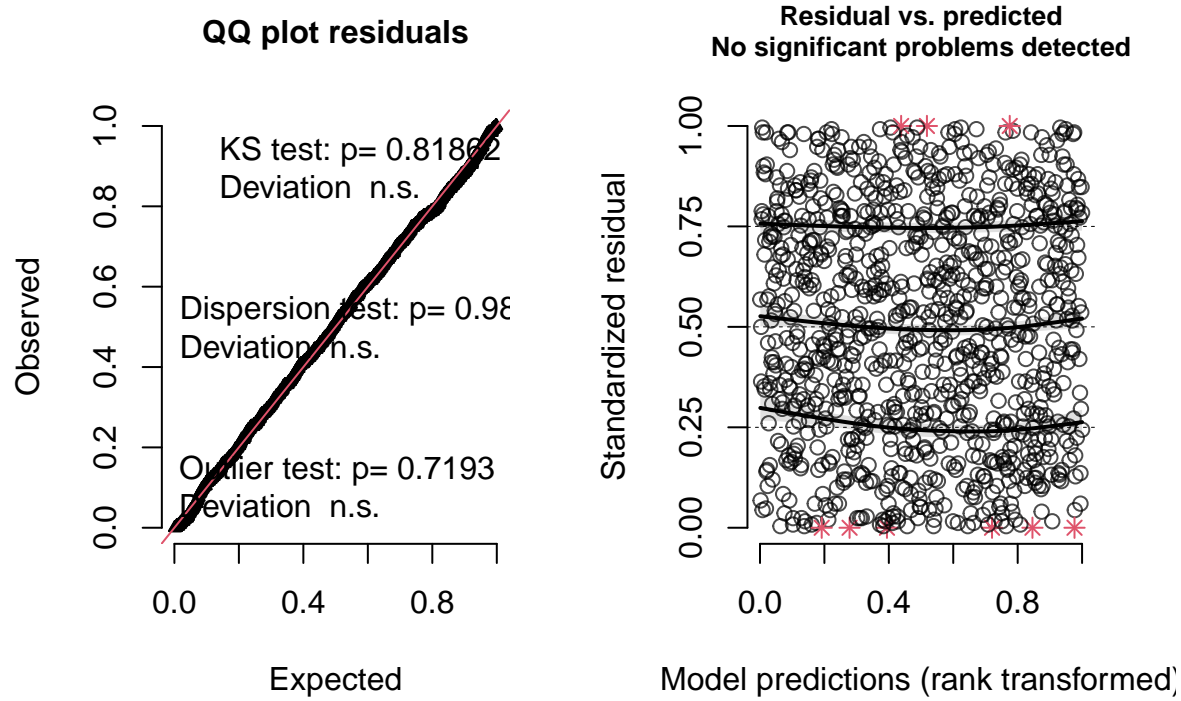
```
ggResidpanel::resid_panel(mixed)
```



Checking residuals (DHARMA)

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

DHARMA residual diagnostics



```
## Object of Class DHARMA with simulated residuals based on 250 simulations with refit = FALSE . See ?DHARMA::simulateResiduals
##
## Scaled residual values: 0.444 0.68 0.724 0.712 0.216 0.824 0.488 0.508 0.616 0.736 0.704 0.992 0.672 0.708 0.624 0.784 0.98 0
```