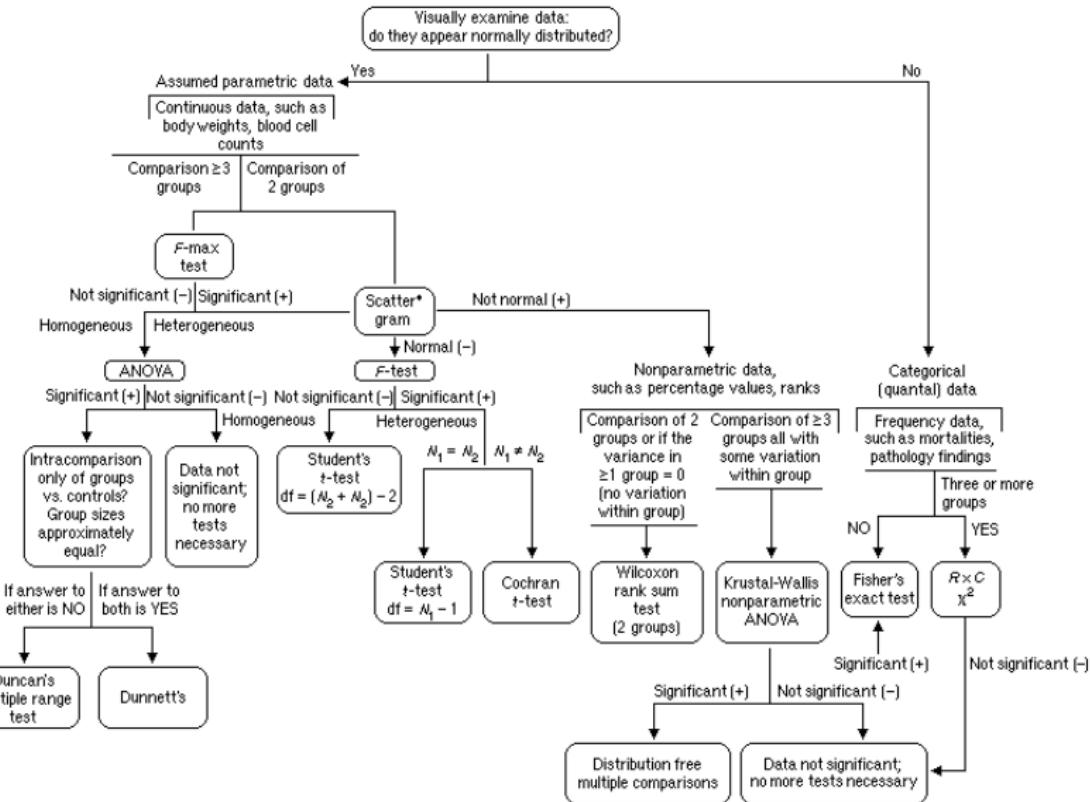


GLM as a unified framework for data analysis

Francisco Rodríguez-Sánchez

<https://frodriguezsanchez.net>

How I was taught statistics



So many questions

- Why should we really use analysis Y over Z?

So many questions

- Why should we really use analysis Y over Z?
- What if my data are **not Normal**?

So many questions

- Why should we really use analysis Y over Z?
- What if my data are **not Normal**?
- What if they are **not independent**?

So many questions

- Why should we really use analysis Y over Z?
- What if my data are **not Normal**?
- What if they are **not independent**?
- Why am I getting **different p-values** with different tests?

So many questions

- Why should we really use analysis Y over Z?
- What if my data are **not Normal**?
- What if they are **not independent**?
- Why am I getting **different p-values** with different tests?
- What even is a **p-value**?

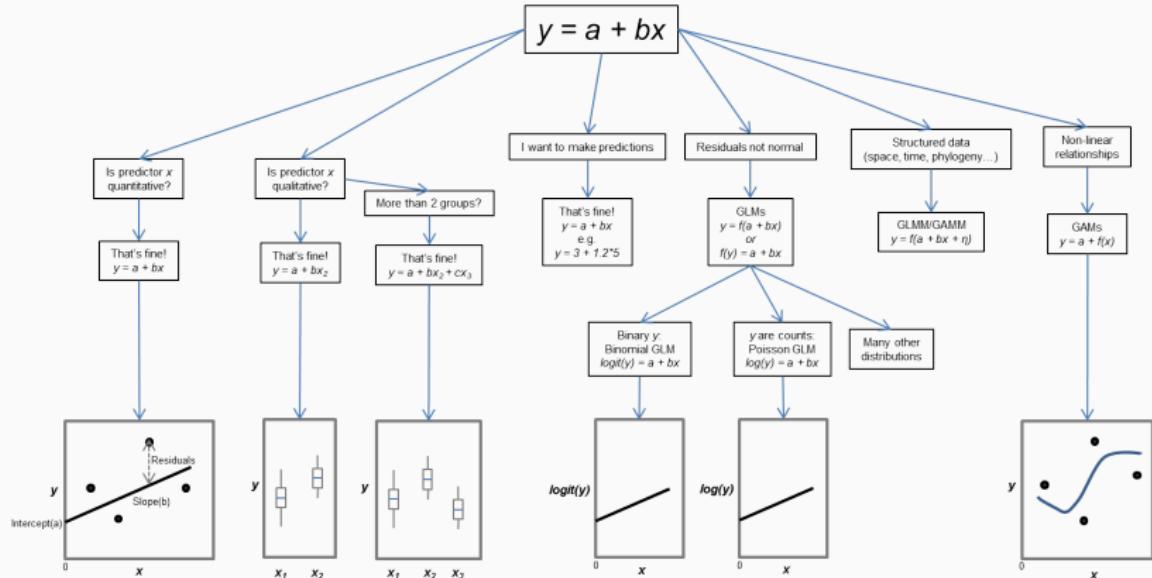
So many questions

- Why should we really use analysis Y over Z?
- What if my data are **not Normal**?
- What if they are **not independent**?
- Why am I getting **different p-values** with different tests?
- What even is a **p-value**?
- How can I take **different factors** into account?

So many questions

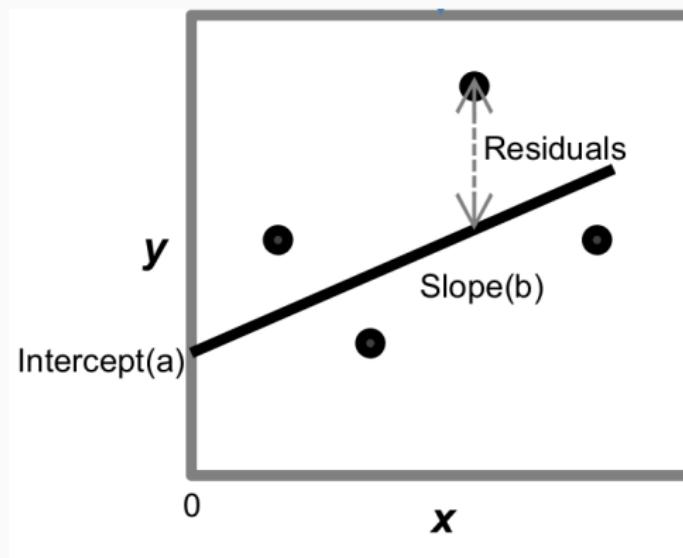
- Why should we really use analysis Y over Z?
- What if my data are **not Normal**?
- What if they are **not independent**?
- Why am I getting **different p-values** with different tests?
- What even is a **p-value**?
- How can I take **different factors** into account?
- Can I make **predictions**?

A unified framework



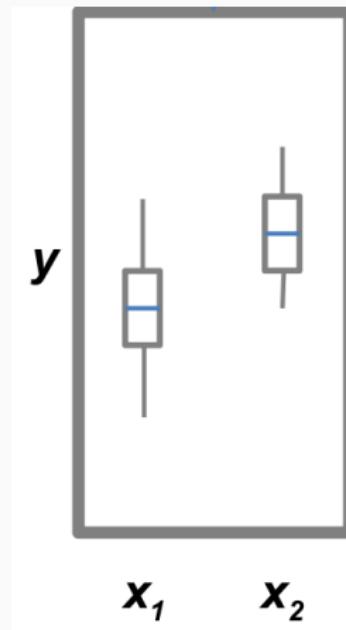
Linear regression

$$y = a + bx$$



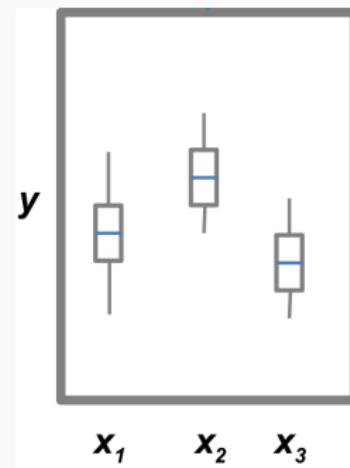
Is predictor X qualitative?

$$y = a + bx_2$$



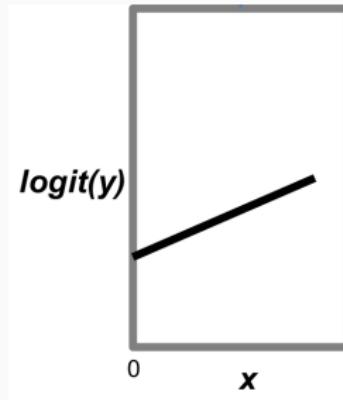
More than 2 groups?

$$y = a + bx_2 + cx_3$$



My data (residuals) are not Normal

$$y = f(a + bx)$$

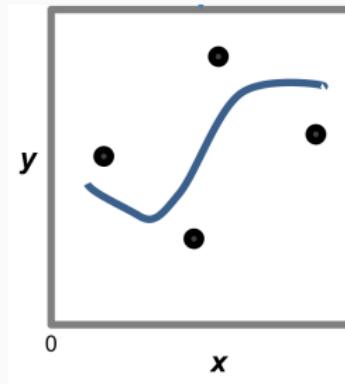


My data are structured (space, time, phylogeny)

$$y = f(a + bx + \eta)$$

Relationships are not linear

$$y = a + f(x)$$



t-tests

ANOVA

regression

.

are special cases of GLM

With GLM we can analyse
many different types of data
using many predictors
(quantitative & qualitative)

Unified, coherent framework for data analysis with many extensions:

- GLMM (mixed models): accomodate data structure & variation (space, time, phylogeny)

Unified, coherent framework for data analysis with many extensions:

- **GLMM** (mixed models): accomodate data structure & variation (space, time, phylogeny)
- **GAMM** (generalised additive models): non-linear relationships

Unified, coherent framework for data analysis with many extensions:

- **GLMM** (mixed models): accomodate data structure & variation (space, time, phylogeny)
- **GAMM** (generalised additive models): non-linear relationships
- **Model-based multivariate** statistics

Unified, coherent framework for data analysis with many extensions:

- **GLMM** (mixed models): accomodate data structure & variation (space, time, phylogeny)
- **GAMM** (generalised additive models): non-linear relationships
- **Model-based multivariate** statistics
- **Bayesian** modelling

The Generalised Linear Model (GLM) is a particularly reasonable vantage point on statistical analyses, as **many tests and procedures are special cases** of the GLM. The downside of that (and any other) vantage point is that **we first have to climb it**. There are the morass of unfamiliar terminology, the scree slopes of probability and the cliffs of distributions. **The vista, however, is magnificent.** From the GLM, t-test, ANOVA and regression neatly arrange themselves into regular patterns, and we can see the paths leading towards the horizon: to time series analyses, Bayesian statistics, spatial statistics and so forth.

Dormann 2020

Introduction to linear models

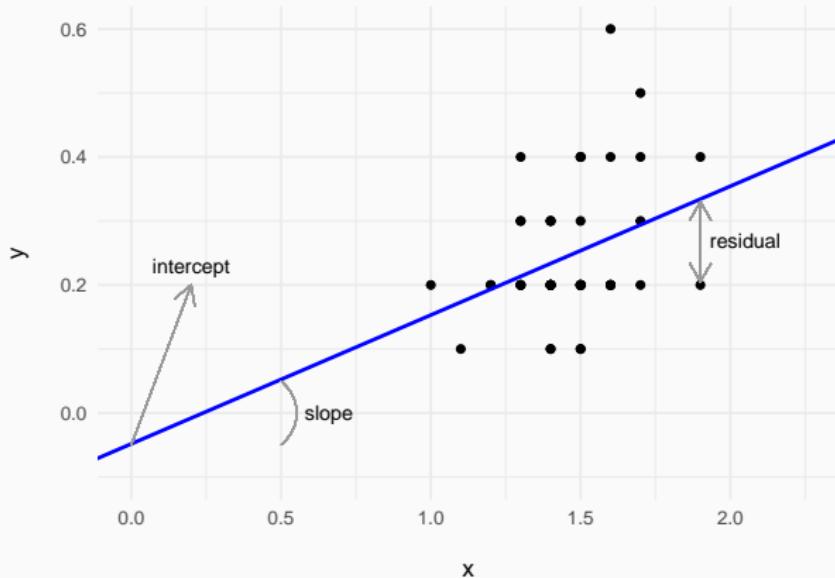
Francisco Rodríguez-Sánchez

<https://frodriguezsanchez.net>

Our unified regression framework (GLM)

$$y_i = a + bx_i + \varepsilon_i$$

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



Data

y = response variable
 x = predictor

Parameters

a = intercept
 b = slope
 σ = residual variation

ε = residuals

What's the intercept?

Expected value of y when predictors (x) = 0

If $x = 0$:

- $y = a + b*0$

What's the intercept?

Expected value of y when predictors (x) = 0

If $x = 0$:

- $y = a + b*0$
- $y = a$

What's the slope?

How much y increases (or decreases) when x increases in 1 unit

If we have model

$$y = 0.5 + 2 * x$$

- If $x = 10 \rightarrow y = 0.5 + 2 * 10 = 20.5$

If x increases 1 unit, y increases 2 units

What's the slope?

How much y increases (or decreases) when x increases in 1 unit

If we have model

$$y = 0.5 + 2 * x$$

- If $x = 10 \rightarrow y = 0.5 + 2 * 10 = 20.5$
- If $x = 11 \rightarrow y = 0.5 + 2 * 11 = 22.5$

If x increases 1 unit, y increases 2 units

Slopes can be negative

If we have model

$$y = 0.5 - 2 * x$$

- If $x = 10 \rightarrow y = 0.5 - 2 * 10 = -19.5$

If x increases 1 unit, y decreases 2 units

Slopes can be negative

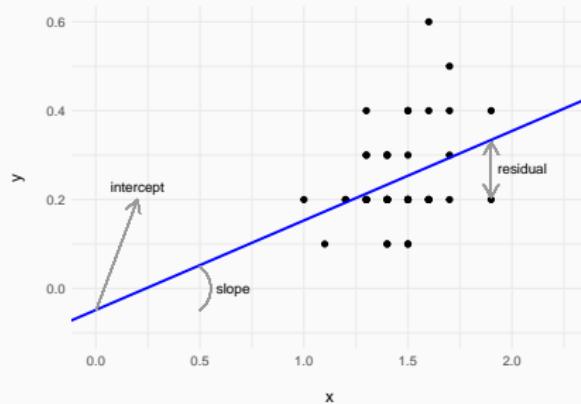
If we have model

$$y = 0.5 - 2 * x$$

- If $x = 10 \rightarrow y = 0.5 - 2 * 10 = -19.5$
- If $x = 11 \rightarrow y = 0.5 - 2 * 11 = -21.5$

If x increases 1 unit, y decreases 2 units

What are residuals?



How far points fall from the regression line

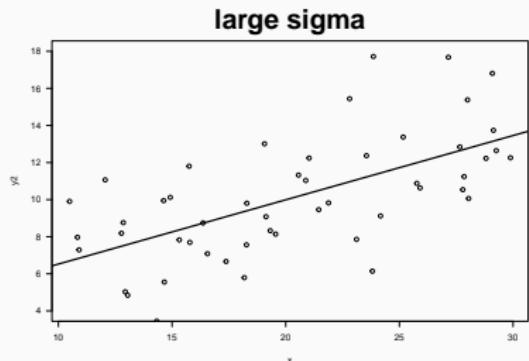
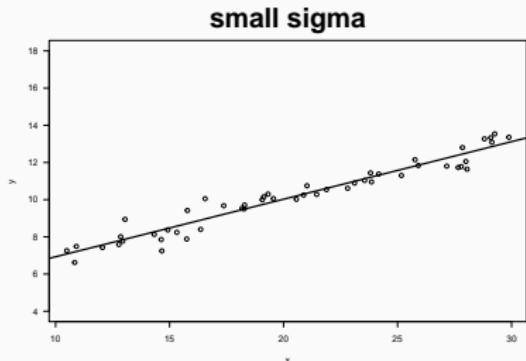
Difference between observed values and values predicted by model
(regression line)

If sigma is large, residuals are larger

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

If sigma is larger:

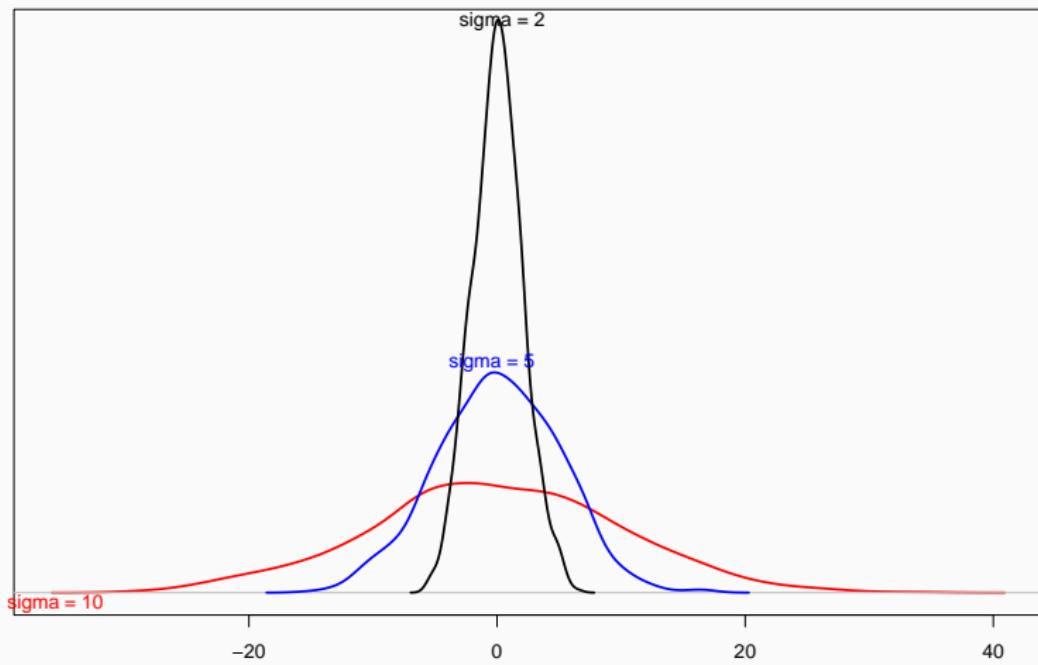
- points farther from regression line
- larger difference of observed - predicted values



Residual variation (sigma) is the Std. Dev. of residuals

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

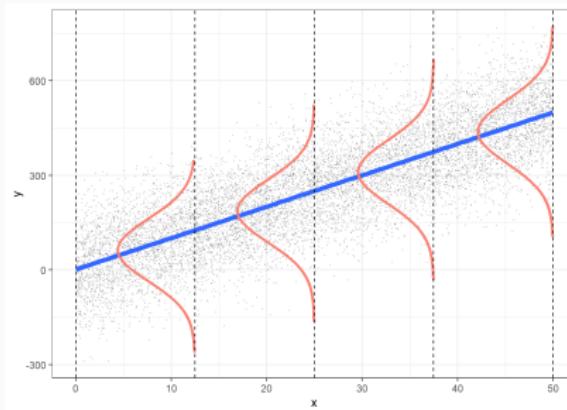
Distribution of residuals



In a general linear model we assume residuals are

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

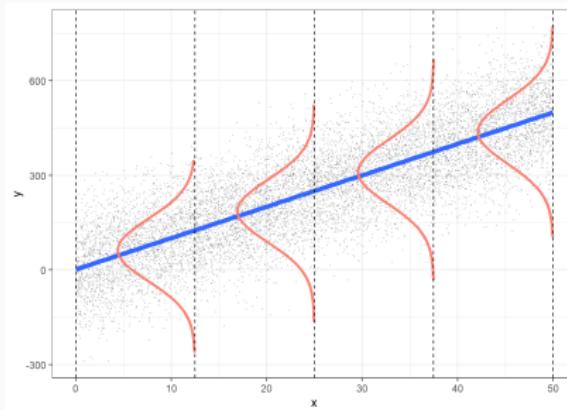
- Normal



In a general linear model we assume residuals are

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

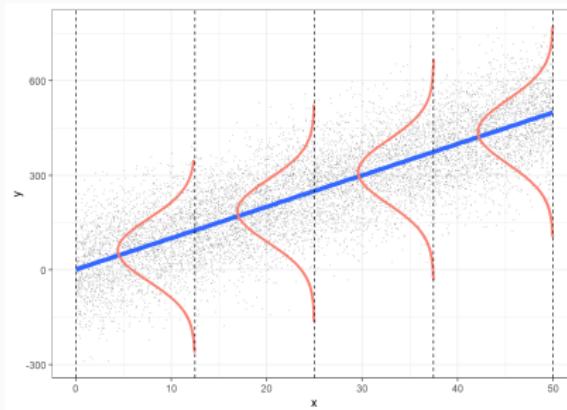
- Normal
- Centred on 0 (no bias)



In a general linear model we assume residuals are

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

- Normal
- Centred on 0 (no bias)
- Homogeneous variance (*homoscedasticity*)



Different ways to write same model

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

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

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

$$\mu_i = a + b x_i$$

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

Quiz

<https://pollev.com/franciscorod726>

Linear models

Francisco Rodríguez-Sánchez

<https://frodriguezsanchez.net>

Example dataset: forest trees

- Download [this dataset](#) (or the entire [zip file](#))

```
trees <- read.csv("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 |

Example dataset: forest trees

- Download [this dataset](#) (or the entire [zip file](#))
- Import:

```
trees <- read.csv("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 |

Questions

- What is the relationship between DBH and height?

Questions

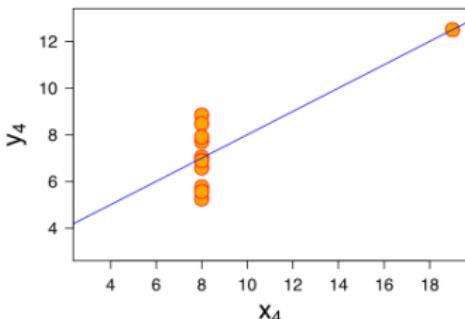
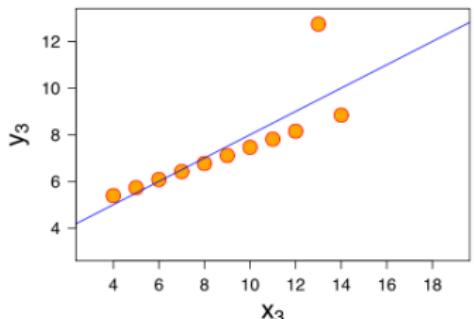
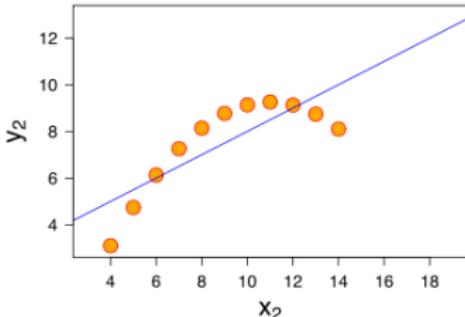
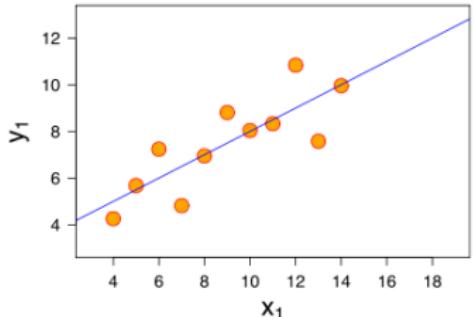
- What is the relationship between DBH and height?
- Do taller trees have bigger trunks?

Questions

- What is the relationship between DBH and height?
- Do taller trees have bigger trunks?
- Can we predict height from DBH? How well?

Always plot your data first!

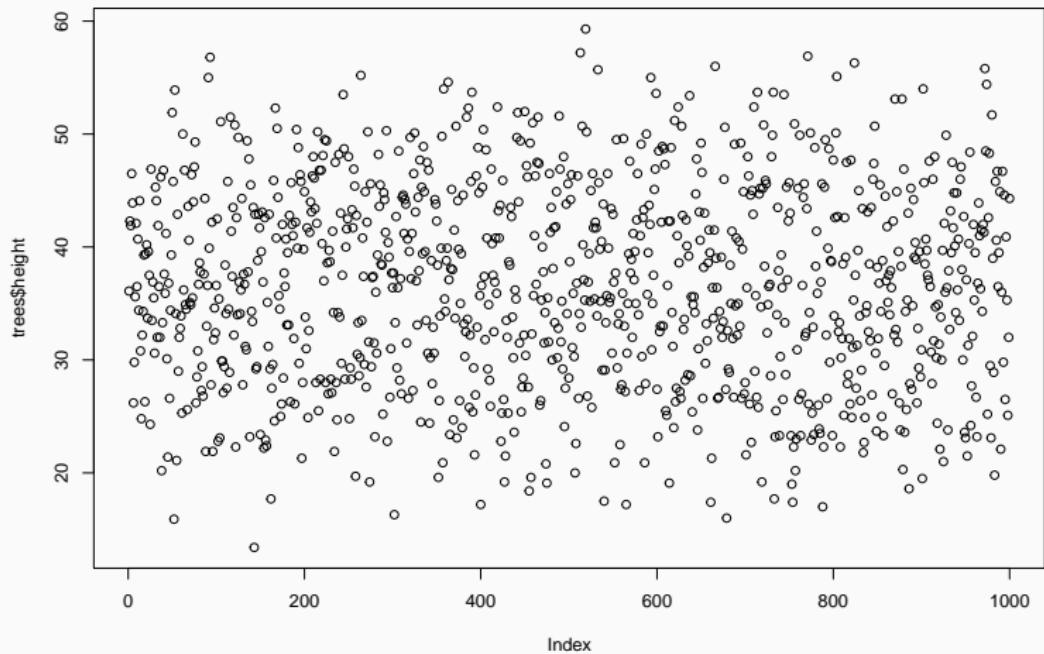
Always plot your data first!



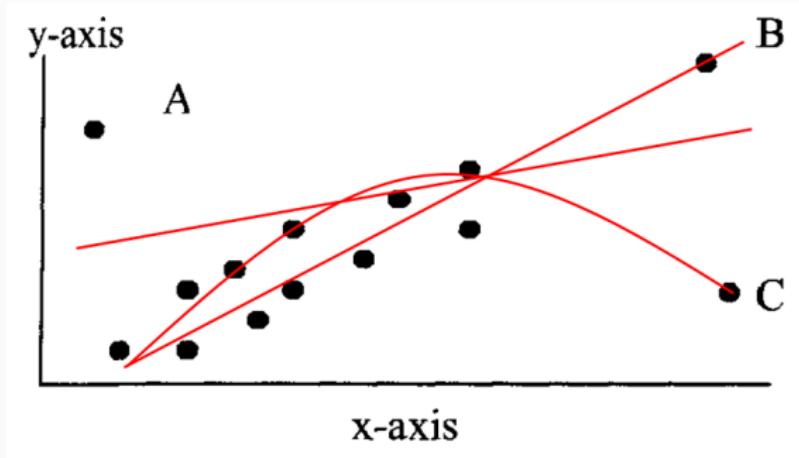
Exploratory Data Analysis (EDA)

Outliers

```
plot(trees$height)
```



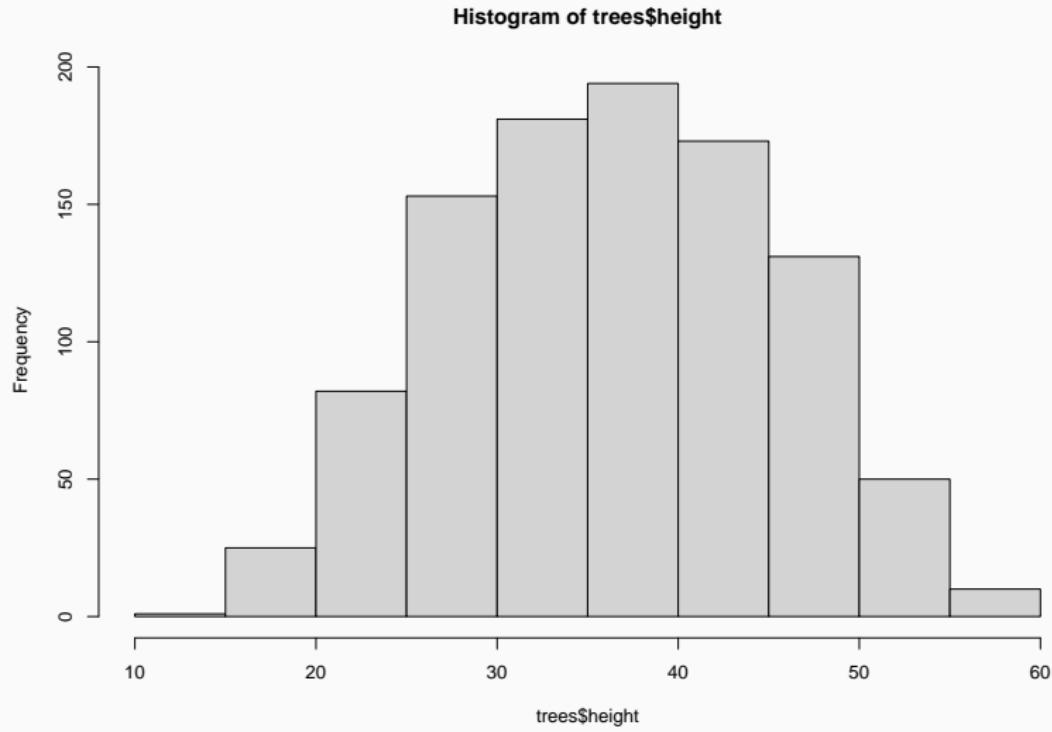
Outliers impact on regression



See <http://rpsychologist.com/d3/correlation/>

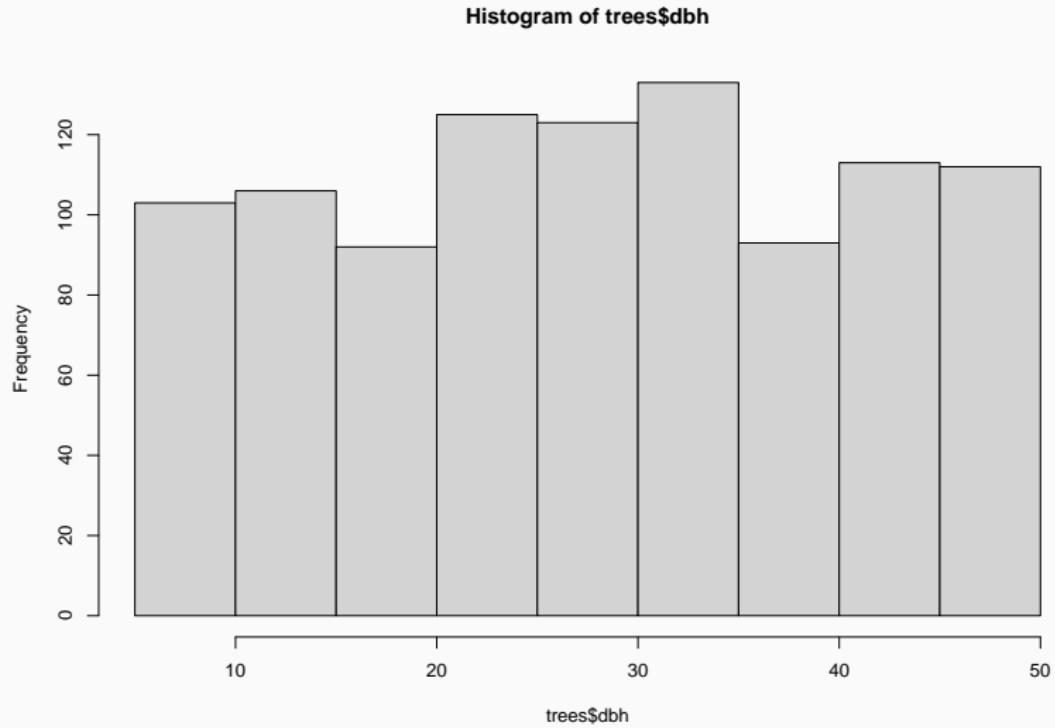
Histogram of response variable

```
hist(trees$height)
```



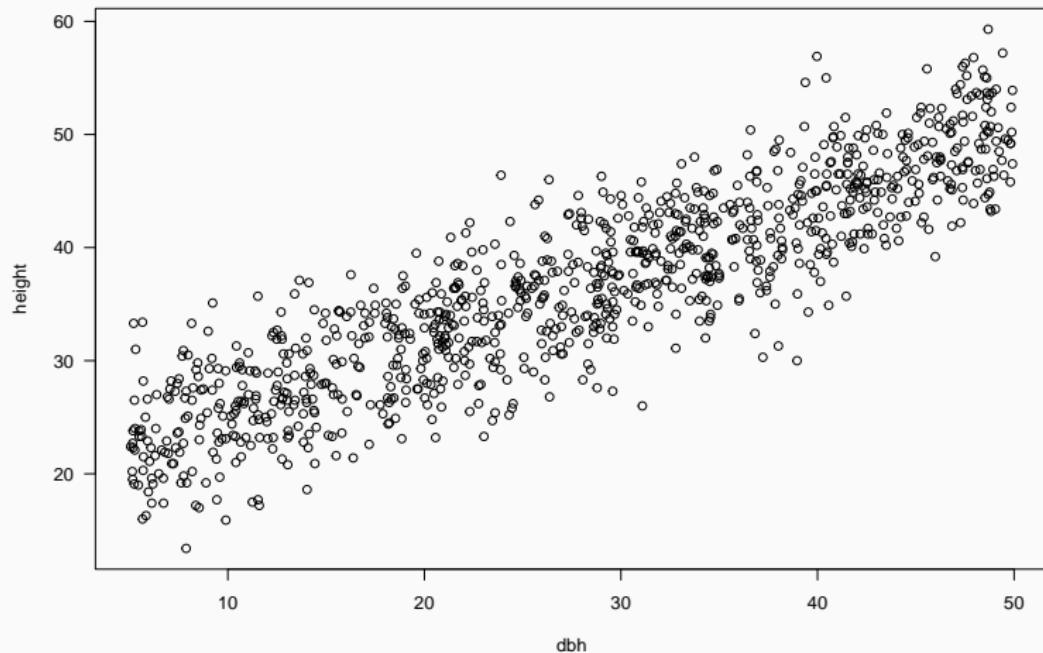
Histogram of predictor variable

```
hist(trees$dbh)
```



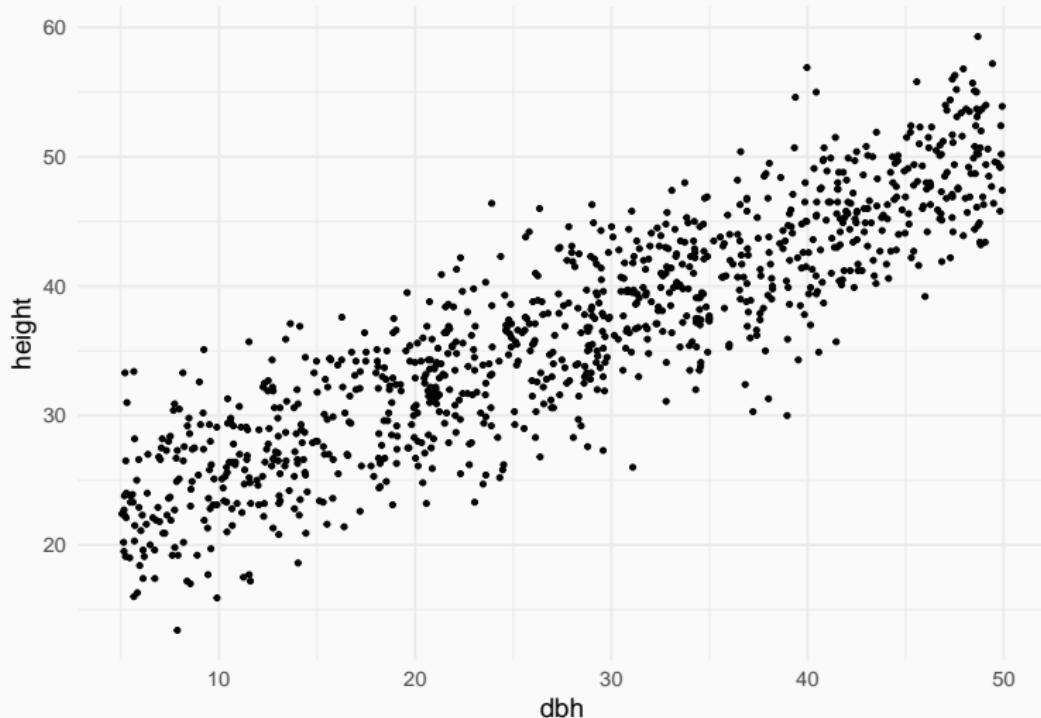
Scatterplot

```
plot(height ~ dbh, data = trees, las = 1)
```



Scatterplot

```
ggplot(trees) +  
  geom_point(aes(dbh, height))
```



Model fitting

Now fit model

Hint: `lm`

Now fit model

Hint: `lm`

```
m1 <- lm(height ~ dbh, data = trees)
```

which corresponds to

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

Package `equatiomatic` returns model structure

```
library("equatiomatic")
m1 <- lm(height ~ dbh, data = trees)
equatiomatic::extract_eq(m1)
```

$$\text{height} = \alpha + \beta_1(\text{dbh}) + \epsilon \quad (1)$$

```
equatiomatic::extract_eq(m1, use_coefs = TRUE)
```

$$\widehat{\text{height}} = 19.34 + 0.62(\text{dbh}) \quad (2)$$

Model interpretation

What does this mean?

```
summary(m1)
```

Call:

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

Residuals:

| Min | 1Q | Median | 3Q | Max |
|----------|---------|--------|--------|---------|
| -13.3270 | -2.8978 | 0.1057 | 2.7924 | 12.9511 |

Coefficients:

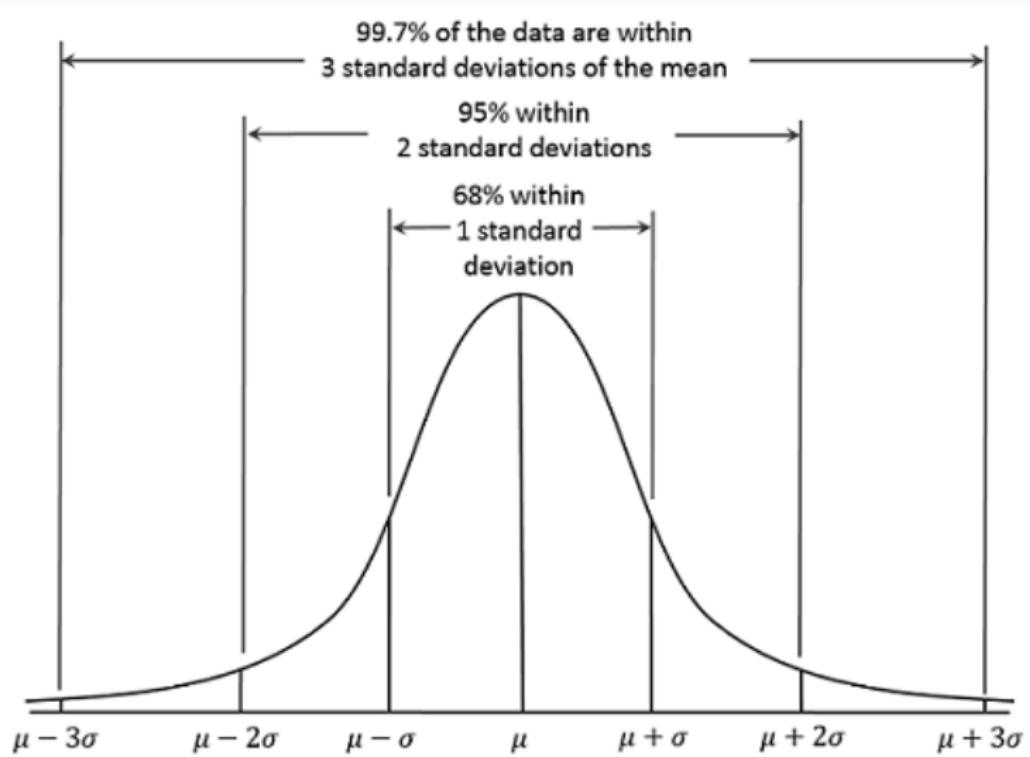
| | Estimate | Std. Error | t value | Pr(> t) | | | | | | | |
|----------------|----------|------------|---------|------------|------|-----|------|-----|-----|-----|---|
| (Intercept) | 19.33920 | 0.31064 | 62.26 | <2e-16 *** | | | | | | | |
| dbh | 0.61570 | 0.01013 | 60.79 | <2e-16 *** | | | | | | | |
| --- | | | | | | | | | | | |
| Signif. codes: | 0 | '***' | 0.001 | '**' | 0.01 | '*' | 0.05 | '.' | 0.1 | ' ' | 1 |

Residual standard error: 4.093 on 998 degrees of freedom

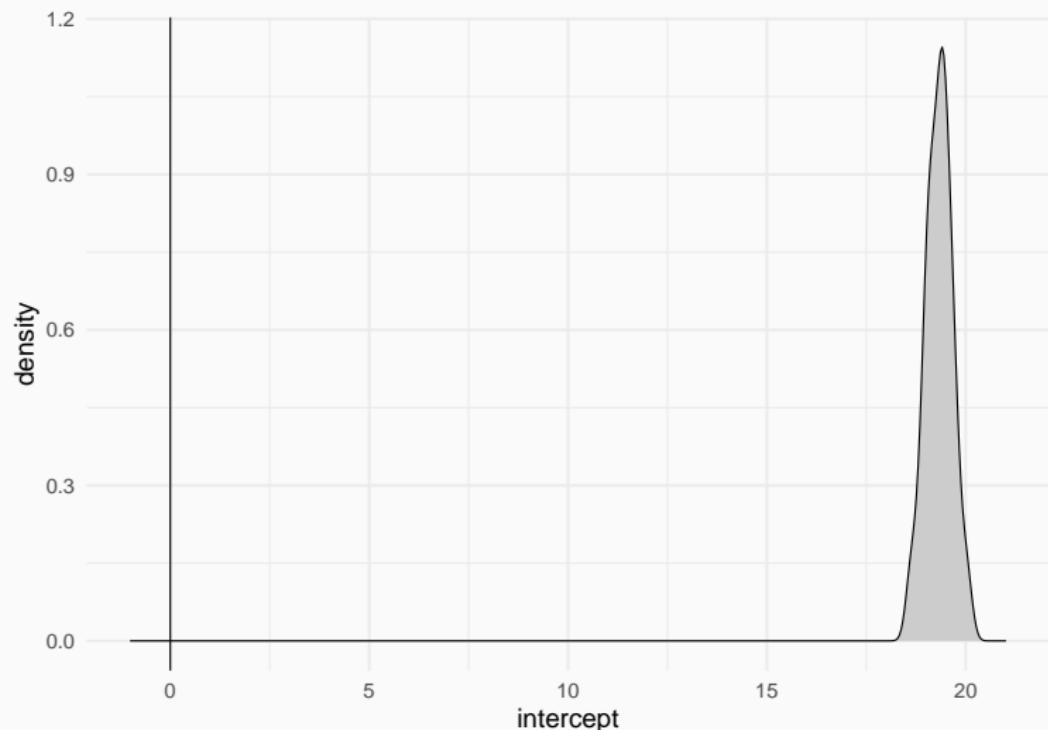
Multiple R-squared: 0.7874, Adjusted R-squared: 0.7871

F-statistic: 3695 on 1 and 998 DF, p-value: < 2.2e-16

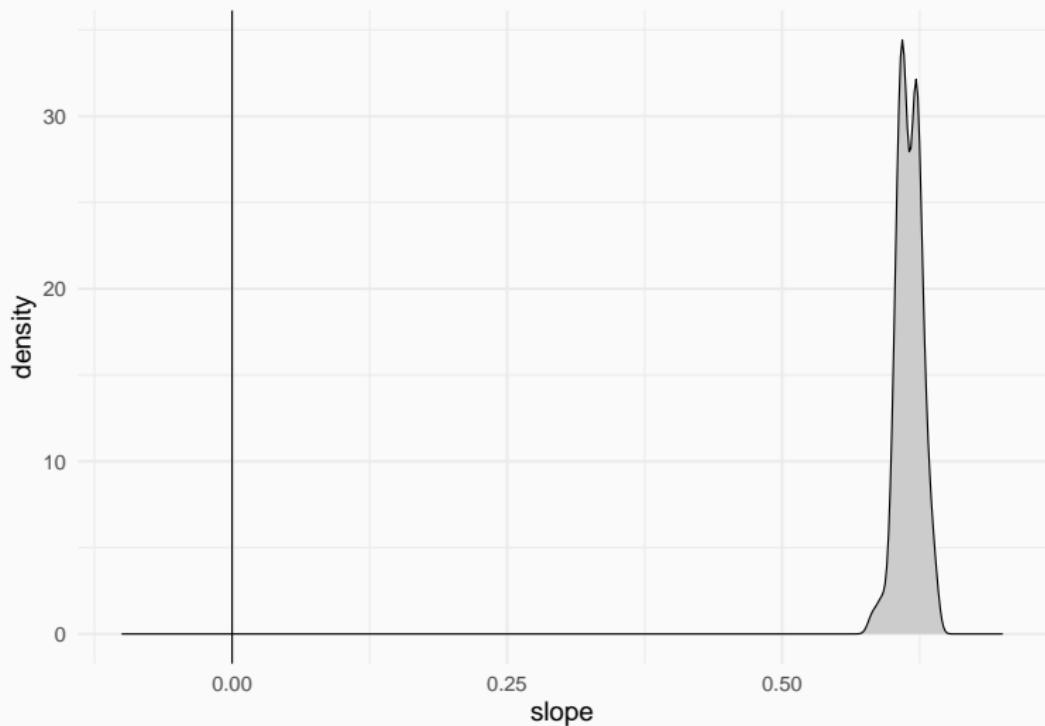
Remember that in a Normal distribution



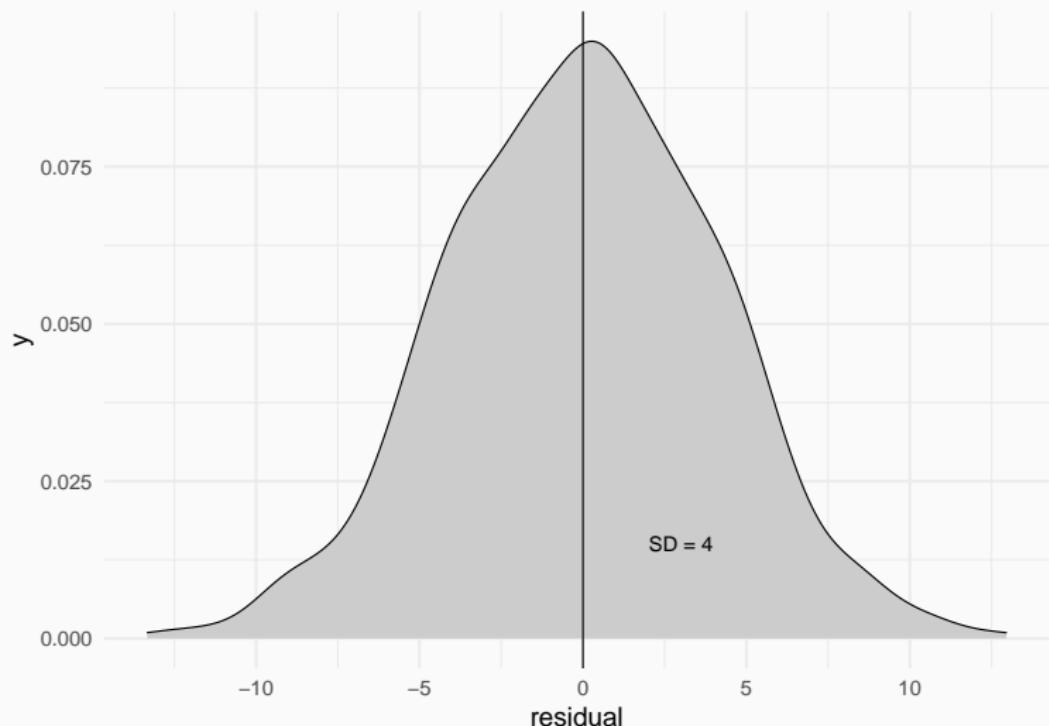
Estimated distribution of the intercept parameter



Estimated distribution of the slope parameter



Distribution of residuals



Degrees of freedom

$$DF = n - p$$

n = sample size

p = number of estimated parameters

R-squared

Proportion of 'explained' variance

$$R^2 = 1 - \frac{\text{Residual Variation}}{\text{Total Variation}}$$

Adjusted R-squared

Accounts for model complexity (number of parameters)

$$R_{adj}^2 = 1 - (1 - R^2) \frac{n-1}{n-p-1}$$

Quiz

<https://pollev.com/franciscorod726>

Retrieving model coefficients

```
coef(m1)
```

| | dbh |
|-------------|------------|
| (Intercept) | 19.3391968 |
| dbh | 0.6157036 |

Confidence intervals for parameters

```
confint(m1)
```

| | 2.5 % | 97.5 % |
|-------------|------------|-----------|
| (Intercept) | 18.7296053 | 19.948788 |
| dbh | 0.5958282 | 0.635579 |

Tidy up model coefficients with broom

```
library("broom")
tidy(m1)
```

```
# A tibble: 2 x 5
  term      estimate std.error statistic p.value
  <chr>      <dbl>     <dbl>      <dbl>     <dbl>
1 (Intercept) 19.3      0.311     62.3      0
2 dbh         0.616     0.0101    60.8      0
```

```
glance(m1)
```

```
# A tibble: 1 x 12
  r.squared adj.r.squared sigma statistic p.value    df logLik     AIC     BIC deviance df.residual
  <dbl>      <dbl>     <dbl>      <dbl>     <dbl> <dbl> <dbl> <dbl> <dbl>      <dbl>      <int>
1 0.787      0.787     4.09     3695.       0      1 -2827. 5660. 5675. 16716.      998
# ... with 1 more variable: nobs <int>, and abbreviated variable names
#   1: r.squared, 2: adj.r.squared, 3: statistic, 4: deviance, 5: df.residual
```

<https://broom.tidymodels.org/>

Retrieving model parameters with `parameters` package

```
library("parameters")
parameters(m1)
```

| Parameter | Coefficient | SE | 95% CI | t(998) | p |
|-------------|-------------|------|----------------|--------|--------|
| <hr/> | | | | | |
| (Intercept) | 19.34 | 0.31 | [18.73, 19.95] | 62.26 | < .001 |
| dbh | 0.62 | 0.01 | [0.60, 0.64] | 60.79 | < .001 |

<https://easystats.github.io/parameters/>

Understanding the fitted effects with `effects` package

```
library("effects")
summary(allEffects(m1))
```

model: height ~ dbh

dbh effect

dbh

5 20 30 40 50

22.41771 31.65327 37.81030 43.96734 50.12438

Lower 95 Percent Confidence Limits

dbh

5 20 30 40 50

21.89682 31.35487 37.55287 43.61733 49.61669

Upper 95 Percent Confidence Limits

dbh

5 20 30 40 50

22.93861 31.95167 38.06774 44.31735 50.63207

Communicating results

Avoid dichotomania of statistical significance

The image is a screenshot of a web page from the journal 'nature'. At the top, there is a red header bar with the word 'nature' and 'International journal of science' in white. To the left of 'nature' is a 'MENU' button with a dropdown arrow. To the right is a 'Subs' button. Below the header, the word 'EDITORIAL' is followed by a small dot and the date '20 MARCH 2019'. The main title of the article is 'It's time to talk about ditching statistical significance', displayed in a large, bold, black serif font. The background of the page is white.

It's time to talk about ditching statistical significance

- “Never conclude there is ‘no difference’ or ‘no association’ just because $p > 0.05$ or CI includes zero”

Avoid dichotomania of statistical significance

The image is a screenshot of a web page from the journal 'nature'. At the top, there is a red header bar with the word 'nature' and 'International journal of science' in white. To the left of 'nature' is a 'MENU' button with a dropdown arrow. To the right is a 'Subs' button. Below the header, the word 'EDITORIAL' is in bold, followed by a dot and the date '20 MARCH 2019'. The main title of the article is 'It's time to talk about ditching statistical significance', displayed in a large, bold, serif font. The background of the page is white.

It's time to talk about ditching statistical significance

- “Never conclude there is ‘no difference’ or ‘no association’ just because $p > 0.05$ or CI includes zero”
- Estimate and communicate effect sizes and their uncertainty

Avoid dichotomania of statistical significance

The image is a screenshot of a web page from the journal 'nature'. At the top, there is a red header bar with the word 'nature' and 'International journal of science' in white. To the left of 'nature' is a 'MENU' button with a dropdown arrow. To the right is a 'Subs' button. Below the header, the word 'EDITORIAL' is followed by a small dot and the date '20 MARCH 2019'. The main title of the article is 'It's time to talk about ditching statistical significance', displayed in a large, bold, black serif font. The background of the page is white.

It's time to talk about ditching statistical significance

- “Never conclude there is ‘no difference’ or ‘no association’ just because $p > 0.05$ or CI includes zero”
- Estimate and communicate effect sizes and their uncertainty
- <https://doi.org/10.1038/d41586-019-00857-9>

We found a **significant relationship** between DBH and Height ($p<0.05$).

We found a significant positive relationship between DBH and Height ($p<0.05$) ($b = 0.61$, $SE = 0.01$).

Models that describe themselves

```
library("report")
report(m1)
```

We fitted a linear model (estimated using OLS) to predict height with dbh (formula: height ~ dbh). The model explains a statistically significant and substantial proportion of variance ($R^2 = 0.79$, $F(1, 998) = 3695.40$, $p < .001$, adj. $R^2 = 0.79$). The model's intercept, corresponding to dbh = 0, is at 19.34 (95% CI [18.73, 19.95], $t(998) = 62.26$, $p < .001$). Within this model:

- The effect of dbh is statistically significant and positive ($\beta = 0.62$, 95% CI [0.60, 0.64], $t(998) = 60.79$, $p < .001$; Std. $\beta = 0.89$, 95% CI [0.86, 0.92])

Standardized parameters were obtained by fitting the model on a standardized version of the dataset. 95% Confidence Intervals (CIs) and p-values were computed using a Wald t-distribution approximation.

<https://easystats.github.io/report/>

Generating table with model results: `xtable`

```
library("xtable")
xtable(m1, digits = 2)
```

% latex table generated in R 4.2.1 by xtable 1.8-4 package % Sat Sep 17 21:12:52 2022

| | Estimate | Std. Error | t value | Pr(> t) |
|-------------|----------|------------|---------|----------|
| (Intercept) | 19.34 | 0.31 | 62.26 | 0.00 |
| dbh | 0.62 | 0.01 | 60.79 | 0.00 |

Generating table with model results: `texreg`

```
library("texreg")
texreg(m1, single.row = TRUE)
```

| Model 1 | |
|---------------------|-----------------|
| (Intercept) | 19.34 (0.31)*** |
| dbh | 0.62 (0.01)*** |
| R ² | 0.79 |
| Adj. R ² | 0.79 |
| Num. obs. | 1000 |

*** $p < 0.001$; ** $p < 0.01$; * $p < 0.05$

Table 1: Statistical models

Generating table with model results: `modelsummary`

```
library("modelsummary")
modelsummary(m1, output = "markdown")
```

| Model 1 | |
|-------------|-----------|
| (Intercept) | 19.339 |
| | (0.311) |
| dbh | 0.616 |
| | (0.010) |
| Num.Obs. | 1000 |
| R2 | 0.787 |
| R2 Adj. | 0.787 |
| AIC | 5660.3 |
| BIC | 5675.0 |
| Log.Lik. | -2827.125 |
| F | 3695.395 |
| RMSE | 4.09 |

Generating table with model results: `gtsummary`

```
library("gtsummary")
tbl_regression(m1, intercept = TRUE)
```

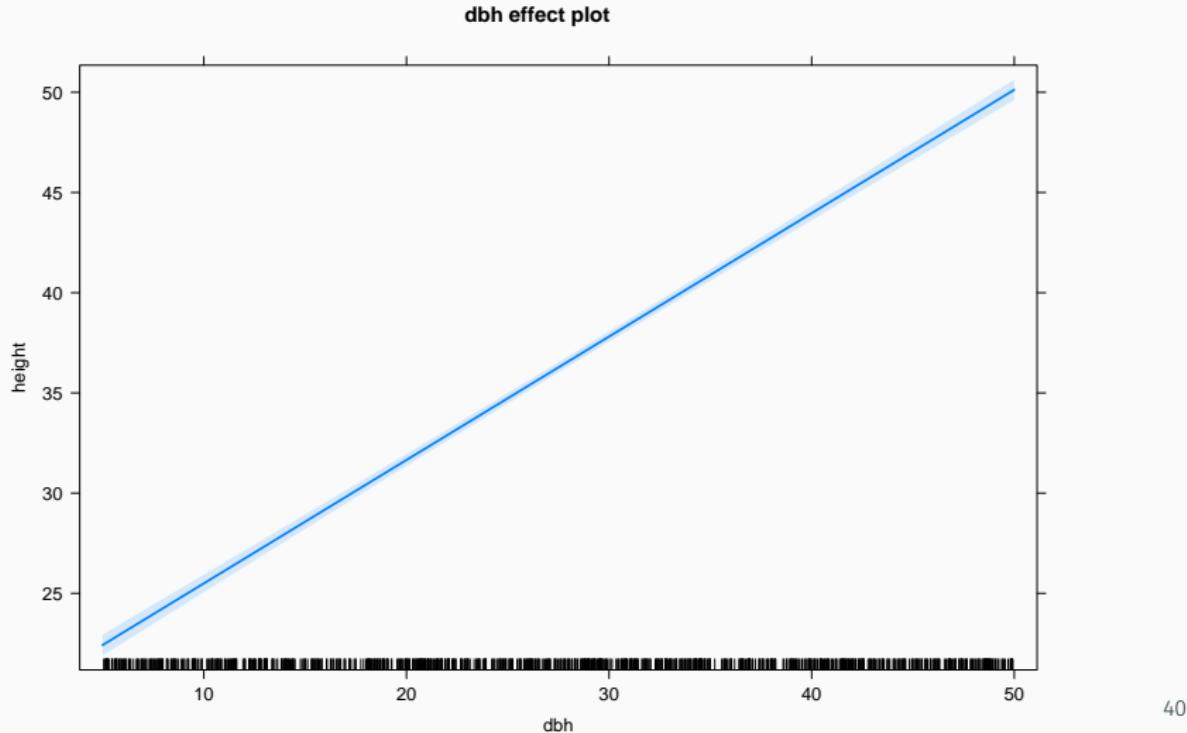
| **Characteristic** | **Beta** | **95% CI** | **p-value** |
|--------------------|----------|------------|-------------|
| (Intercept) | 19 | 19, 20 | <0.001 |
| dbh | 0.62 | 0.60, 0.64 | <0.001 |

<https://www.danieldsjoberg.com/gtsummary>

Visualising fitted model

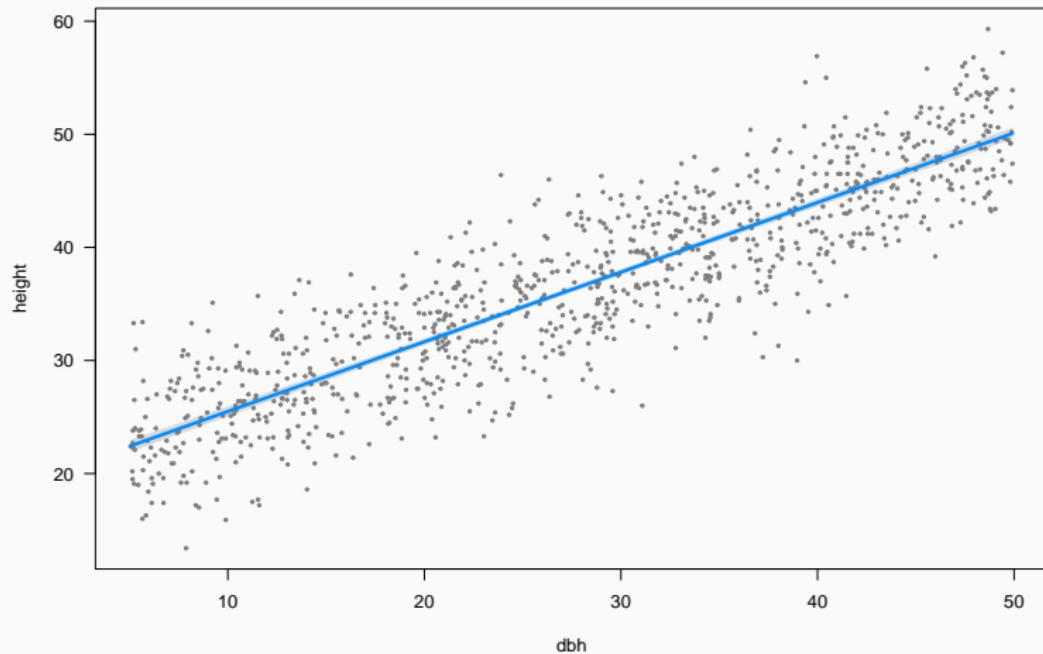
Plot model: effects package

```
library("effects")
plot(allEffects(m1))
```



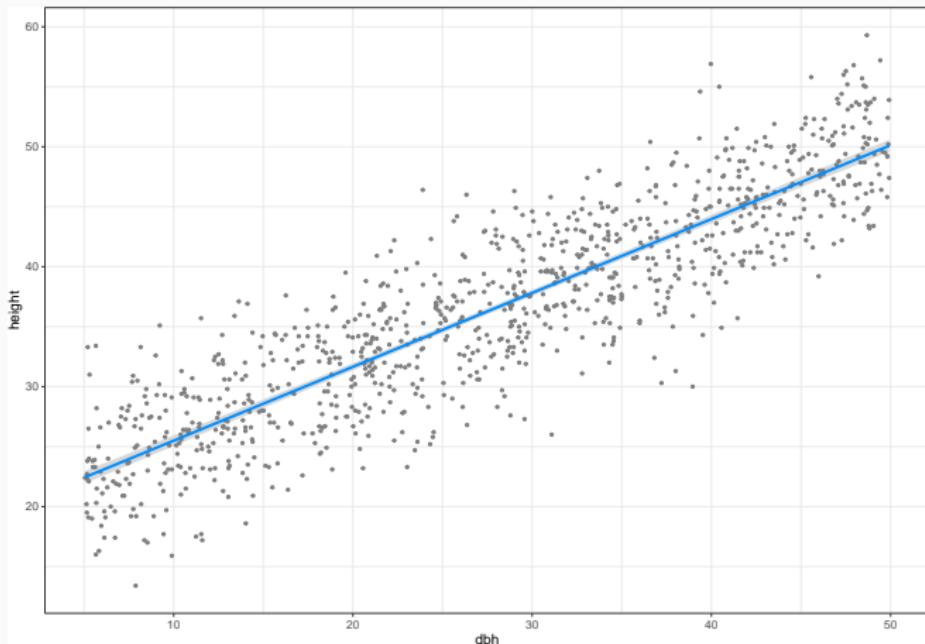
Plot model: visreg

```
library("visreg")
visreg(m1)
```



visreg can use ggplot2 too

```
visreg(m1, gg = TRUE) + theme_bw()
```

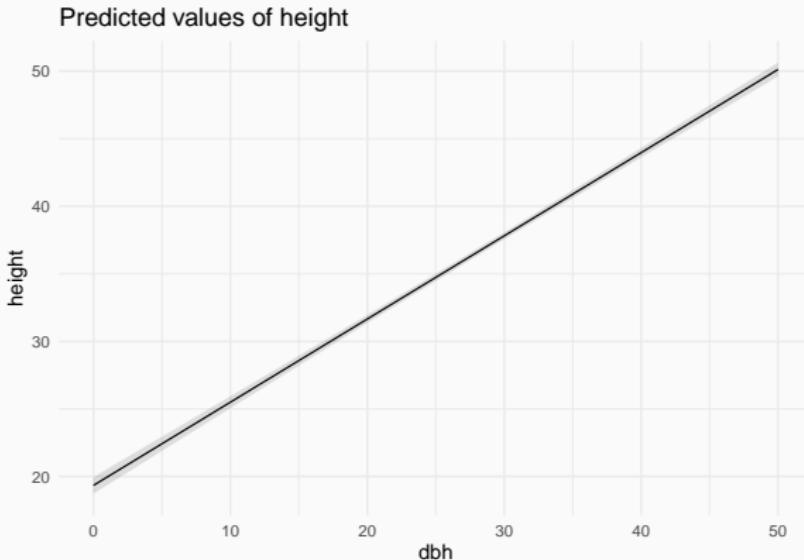


<https://pbreheny.github.io/visreg>

Plot model: sjPlot

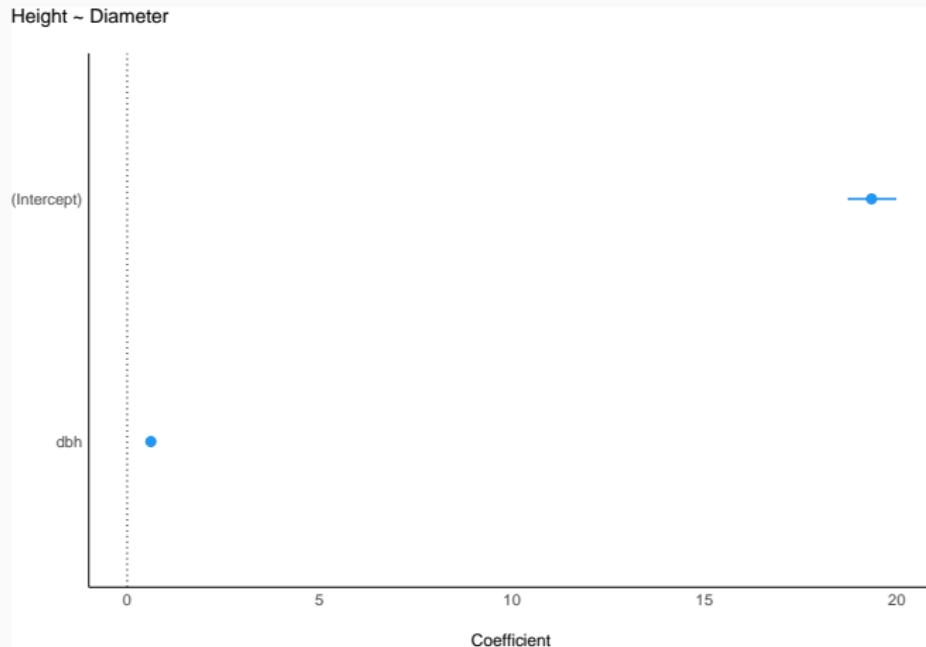
```
library("sjPlot")
plot_model(m1, type = "eff")
```

\$dbh



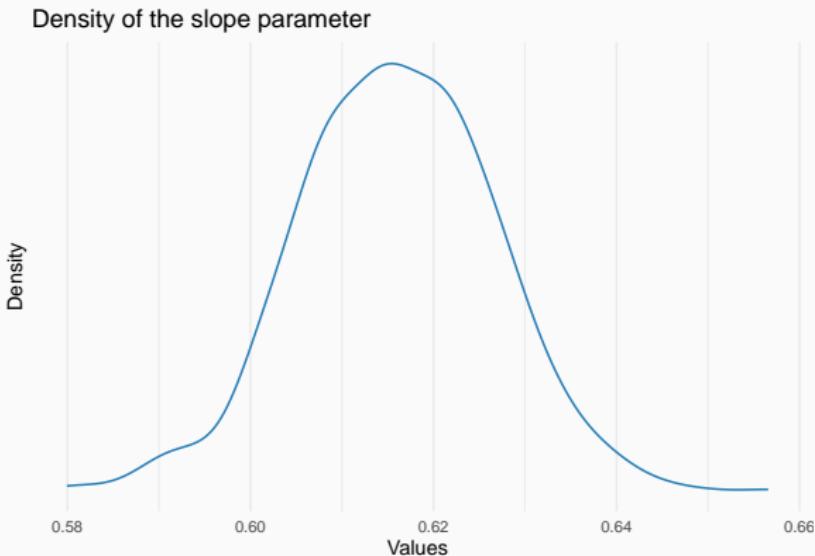
Plot model: see

```
library("see")
plot(parameters(m1), show_intercept = TRUE) +
  labs(title = "Height ~ Diameter")  # ggplot2
```



Plot parameters' estimated distribution: see

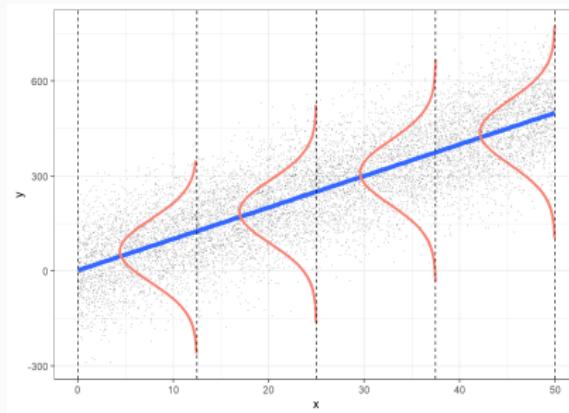
```
plot(simulate_parameters(m1)) +  
  labs(title = "Density of the slope parameter")
```



Model checking

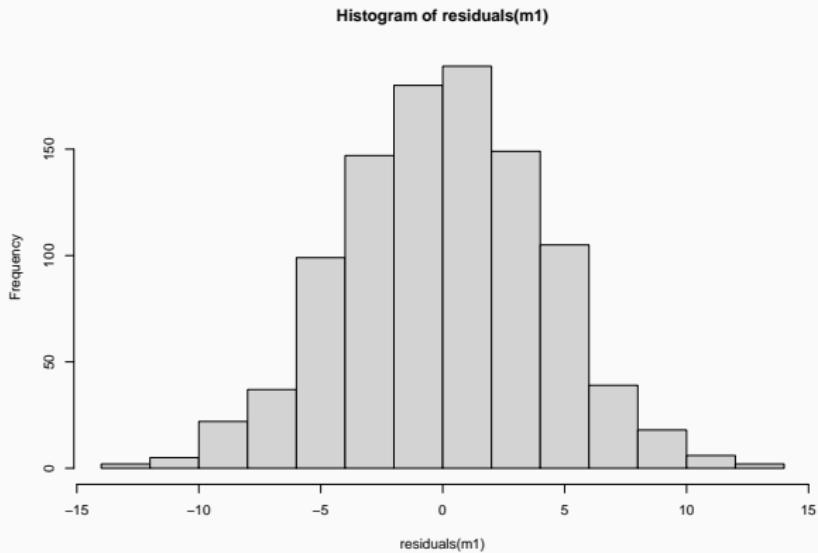
Linear model assumptions

- Linearity (transformations, GAM...)
- Residuals:
 - Independent
 - Equal variance
 - Normal
- Negligible measurement error in predictors



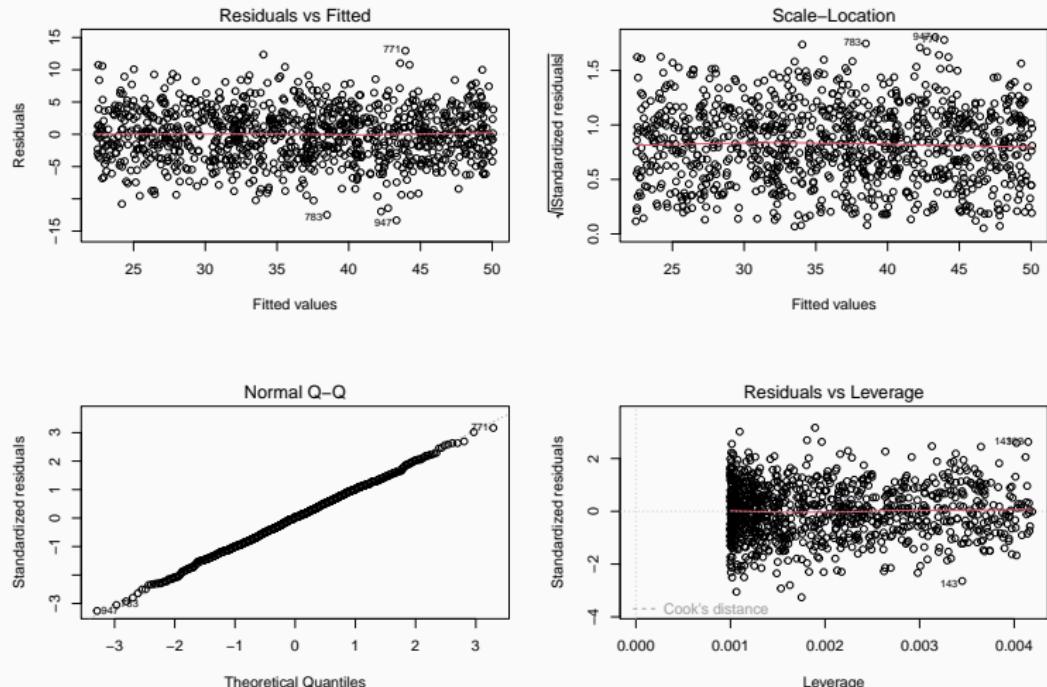
Are residuals normal?

```
hist(residuals(m1))
```



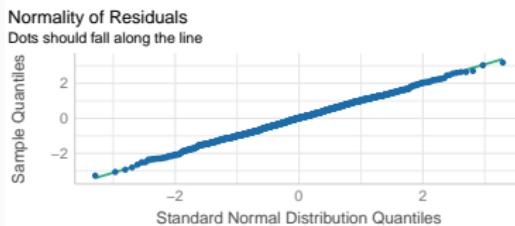
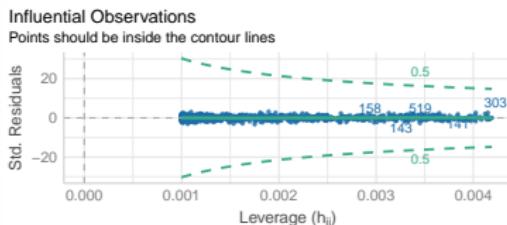
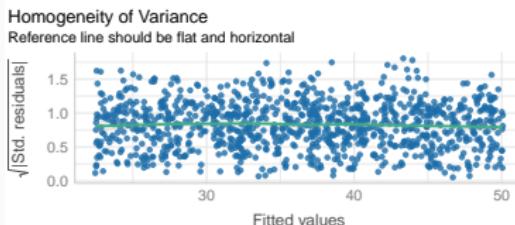
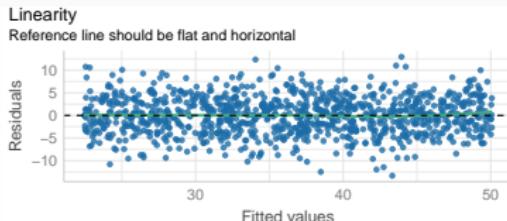
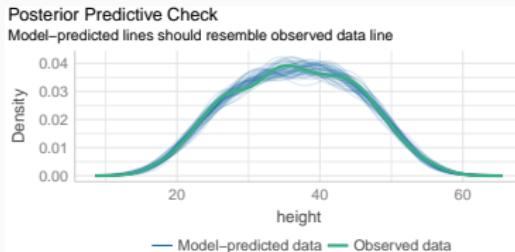
SD = 4.09

Model checking: `plot(model)`



Model checking with performance package

```
library("performance")
check_model(m1)
```



A dashboard to explore the full model

```
library("easystats")
model_dashboard(m1)
```

Using model for prediction

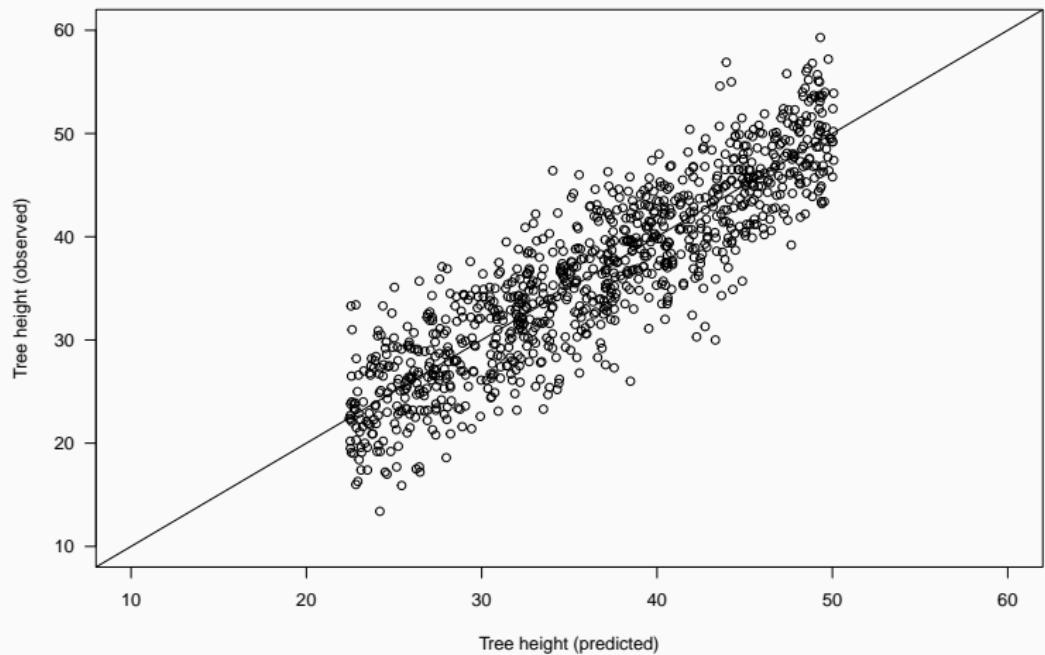
How good is the model in predicting tree height?

`fitted` gives expected value for each observation

```
trees$height.pred <- fitted(m1)
trees$resid <- residuals(m1)
head(trees)
```

| | site | dbh | height | sex | dead | height.pred | resid |
|---|------|-------|--------|--------|------|-------------|------------|
| 1 | 4 | 29.68 | 36.1 | male | 0 | 37.61328 | -1.5132797 |
| 2 | 5 | 33.29 | 42.3 | male | 0 | 39.83597 | 2.4640303 |
| 3 | 2 | 28.03 | 41.9 | female | 0 | 36.59737 | 5.3026313 |
| 4 | 5 | 39.86 | 46.5 | female | 0 | 43.88114 | 2.6188577 |
| 5 | 1 | 47.94 | 43.9 | female | 0 | 48.85603 | -4.9560274 |
| 6 | 1 | 10.82 | 26.2 | male | 0 | 26.00111 | 0.1988903 |

Calibration plot: Observed vs Predicted values



Making predictions for new data

Q: Expected tree height if DBH = 39 cm?

```
new.dbh <- data.frame(dbh = c(39))
predict(m1, new.dbh, se.fit = TRUE)
```

\$fit

1

43.35164

\$se.fit

[1] 0.1715514

\$df

[1] 998

\$residual.scale

[1] 4.092629

Confidence vs Prediction Intervals

Q: Expected tree height if DBH = 39 cm?

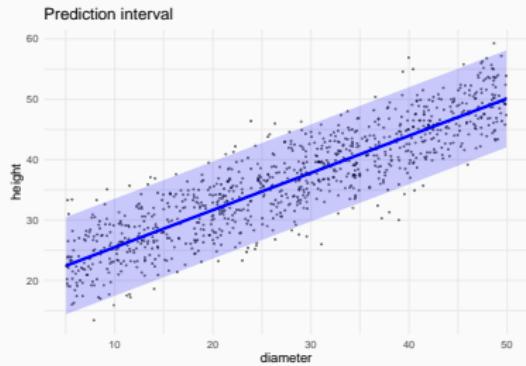
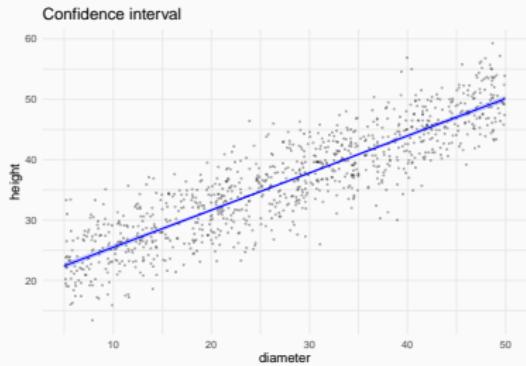
```
predict(m1, new.dbh, interval = "confidence")
```

| | fit | lwr | upr |
|---|----------|----------|----------|
| 1 | 43.35164 | 43.01499 | 43.68828 |

```
predict(m1, new.dbh, interval = "prediction")
```

| | fit | lwr | upr |
|---|----------|----------|----------|
| 1 | 43.35164 | 35.31344 | 51.38983 |

Confidence vs Prediction Intervals



Workflow

- Visualise data

Workflow

- Visualise data
- Understand fitted model (`summary`, `allEffects`...)

Workflow

- Visualise data
- Understand fitted model (`summary`, `allEffects`...)
- Visualise model (`plot(allEffects)`, `visreg`, `see`, `plot_model`...)

Workflow

- Visualise data
- Understand fitted model (`summary`, `allEffects`...)
- Visualise model (`plot(allEffects)`, `visreg`, `see`, `plot_model`...)
- Check model (`plot`, `check_model`, calibration plot...)

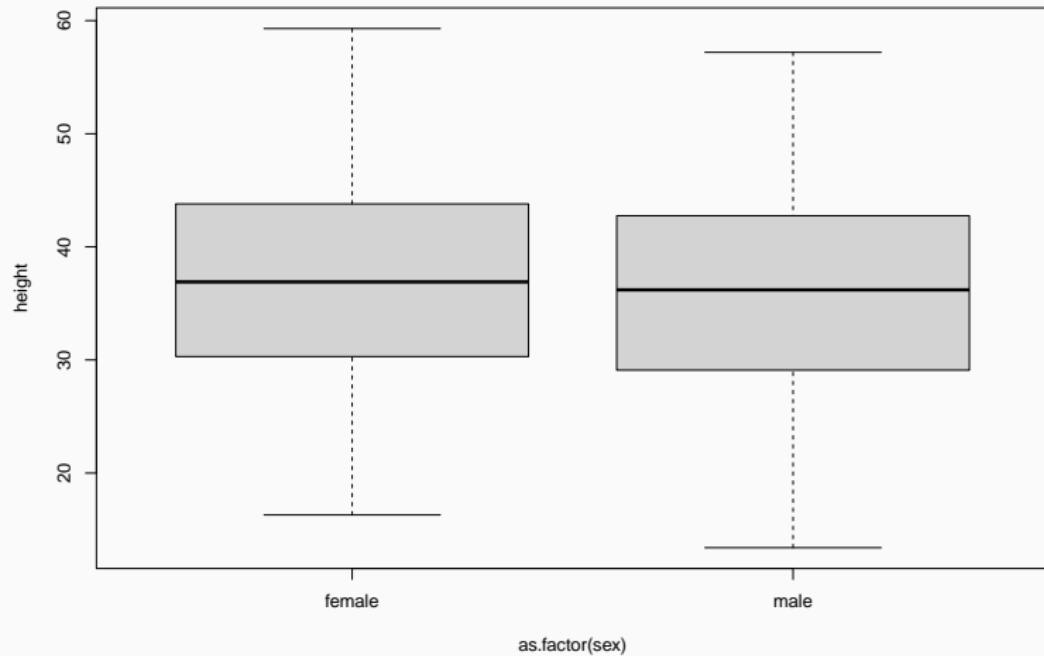
Workflow

- Visualise data
- Understand fitted model (`summary`, `allEffects`...)
- Visualise model (`plot(allEffects)`, `visreg`, `see`, `plot_model`...)
- Check model (`plot`, `check_model`, calibration plot...)
- Predict (`fitted`, `predict`)

Categorical predictors (factors)

Q: Does tree height vary with sex?

```
plot(height ~ as.factor(sex), data = trees)
```



Model height ~ sex

```
m2 <- lm(height ~ sex, data = trees)
```

Call:

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

Residuals:

| Min | 1Q | Median | 3Q | Max |
|----------|---------|---------|--------|---------|
| -22.6881 | -6.7881 | -0.0097 | 6.7261 | 22.3687 |

Coefficients:

| | Estimate | Std. Error | t value | Pr(> t) |
|-------------|----------|------------|---------|------------|
| (Intercept) | 36.9312 | 0.3981 | 92.778 | <2e-16 *** |
| sexmale | -0.8432 | 0.5607 | -1.504 | 0.133 |

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

Residual standard error: 8.865 on 998 degrees of freedom

Multiple R-squared: 0.002261, Adjusted R-squared: 0.001261

F-statistic: 2.261 on 1 and 998 DF, p-value: 0.133

Linear model with categorical predictors

```
m2 <- lm(height ~ sex, data = trees)
```

corresponds to

$$Height_i = a + b_{male} + \varepsilon_i$$

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

Model height ~ sex

```
m2 <- lm(height ~ sex, data = trees)
```

Call:

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

Residuals:

| Min | 1Q | Median | 3Q | Max |
|----------|---------|---------|--------|---------|
| -22.6881 | -6.7881 | -0.0097 | 6.7261 | 22.3687 |

Coefficients:

| | Estimate | Std. Error | t value | Pr(> t) | | | | | | | |
|----------------|----------|------------|---------|----------|------|-----|------|-----|-----|-----|---|
| (Intercept) | 36.9312 | 0.3981 | 92.778 | <2e-16 | *** | | | | | | |
| sexmale | -0.8432 | 0.5607 | -1.504 | 0.133 | | | | | | | |
| --- | | | | | | | | | | | |
| Signif. codes: | 0 | '***' | 0.001 | '**' | 0.01 | '*' | 0.05 | '.' | 0.1 | ' ' | 1 |

Quiz

<https://pollev.com/franciscorod726>

Let's read the model report...

```
report(m2)
```

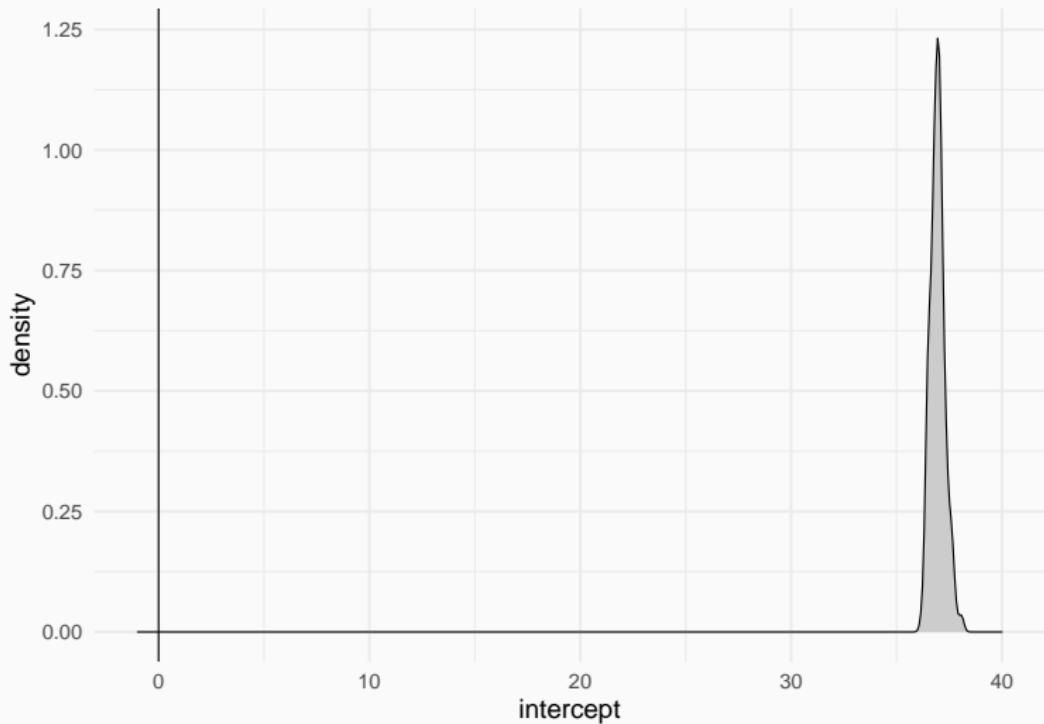
We fitted a linear model (estimated using OLS) to predict height with sex (formula: height ~ sex). The model explains a statistically not significant and very weak proportion of variance ($R^2 = 2.26e-03$, $F(1, 998) = 2.26$, $p = 0.133$, adj. $R^2 = 1.26e-03$). The model's intercept, corresponding to sex = female, is at 36.93 (95% CI [36.15, 37.71], $t(998) = 92.78$, $p < .001$). Within this model:

- The effect of sex [male] is statistically non-significant and negative (beta = -0.84, 95% CI [-1.94, 0.26], $t(998) = -1.50$, $p = 0.133$; Std. beta = -0.10, 95% CI [-0.22, 0.03])

Standardized parameters were obtained by fitting the model on a standardized version of the dataset. 95% Confidence Intervals (CIs) and p-values were computed using a Wald t-distribution approximation.

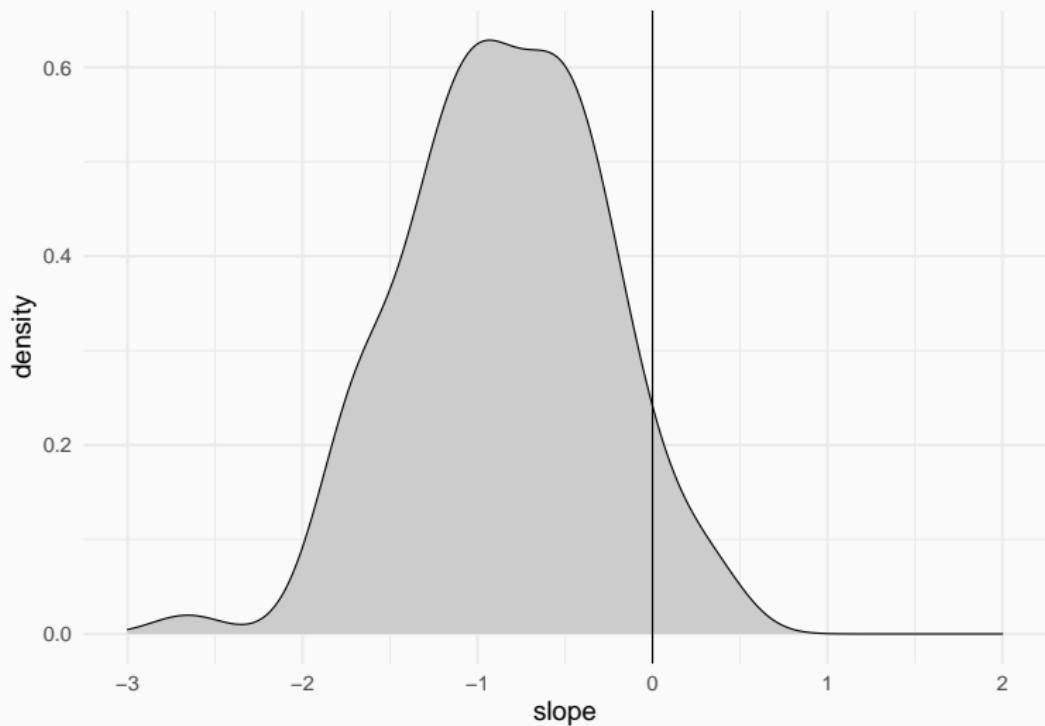
Estimated distribution of the intercept parameter

Intercept = Height of females



Estimated distribution of the *beta* parameter

beta = height difference of males vs females



Analysing differences among factor levels

```
library("modelbased")
estimate_means(m2)
```

Estimated Marginal Means

| sex | Mean | SE | 95% CI |
|--------|-------|------|----------------|
| ----- | | | |
| male | 36.09 | 0.39 | [35.31, 36.86] |
| female | 36.93 | 0.40 | [36.15, 37.71] |

Marginal means estimated at sex

Analysing differences among factor levels

```
estimate_contrasts(m2)
```

Marginal Contrasts Analysis

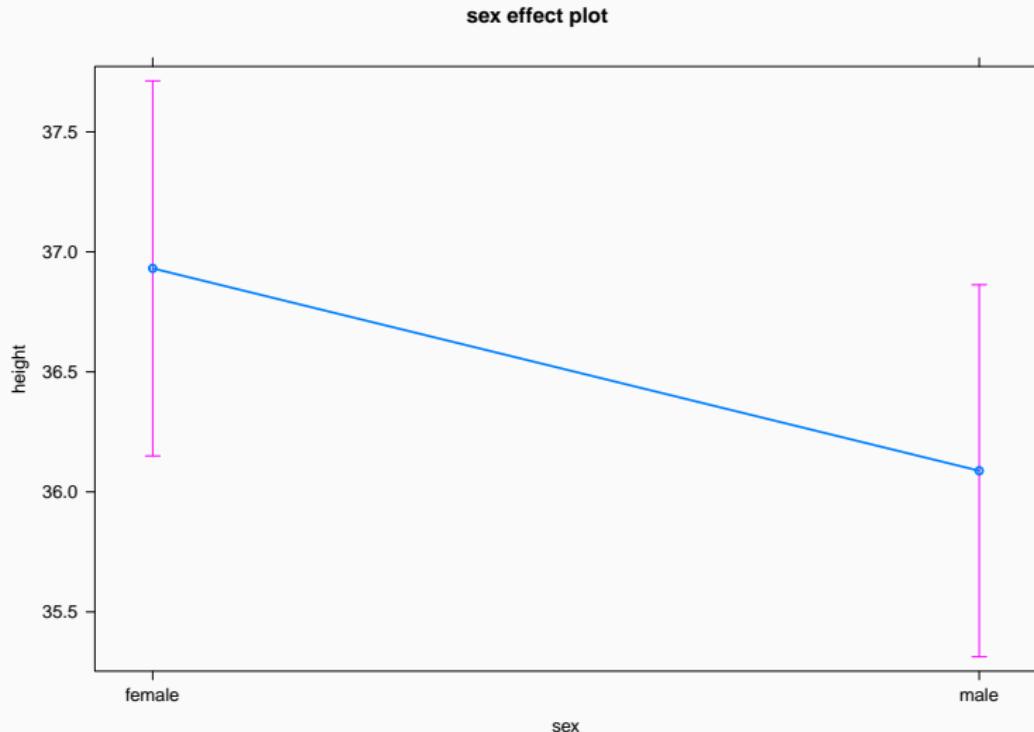
| Level1 | Level2 | Difference | 95% CI | SE | t(998) | p |
|--------|--------|------------|---------------|------|--------|-------|
| <hr/> | | | | | | |
| male | female | -0.84 | [-1.94, 0.26] | 0.56 | -1.50 | 0.133 |

Marginal contrasts estimated at sex

p-value adjustment method: Holm (1979)

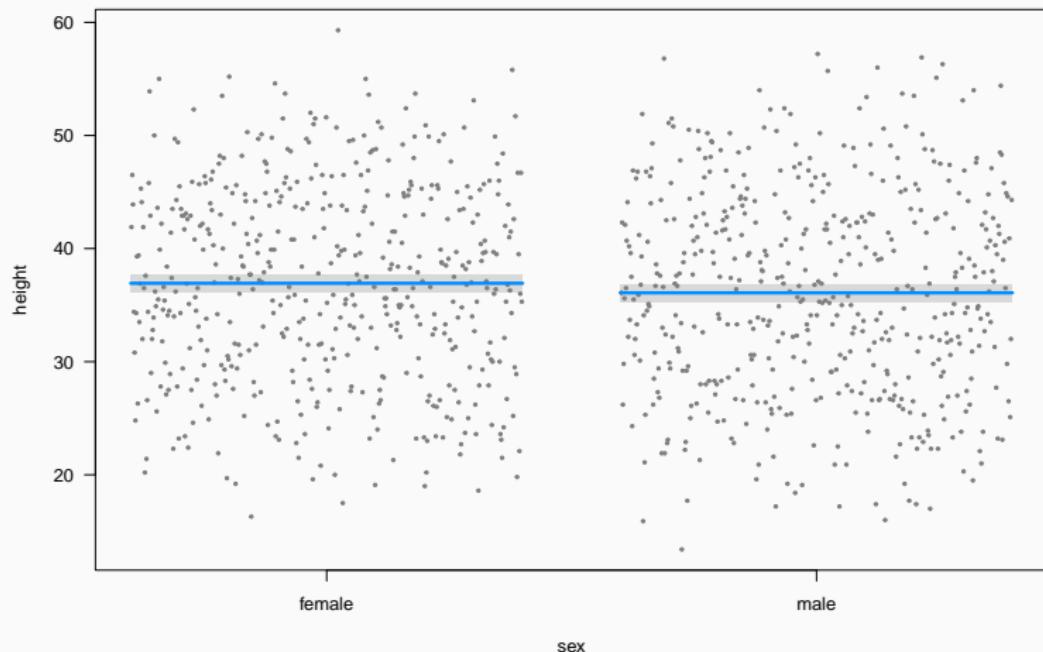
Plot

```
plot(allEffects(m2))
```



Plot (visreg)

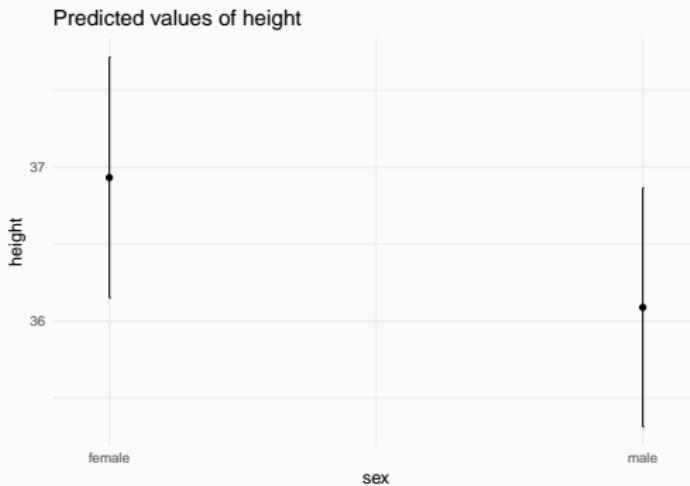
```
visreg(m2)
```



Plot model (sjPlot)

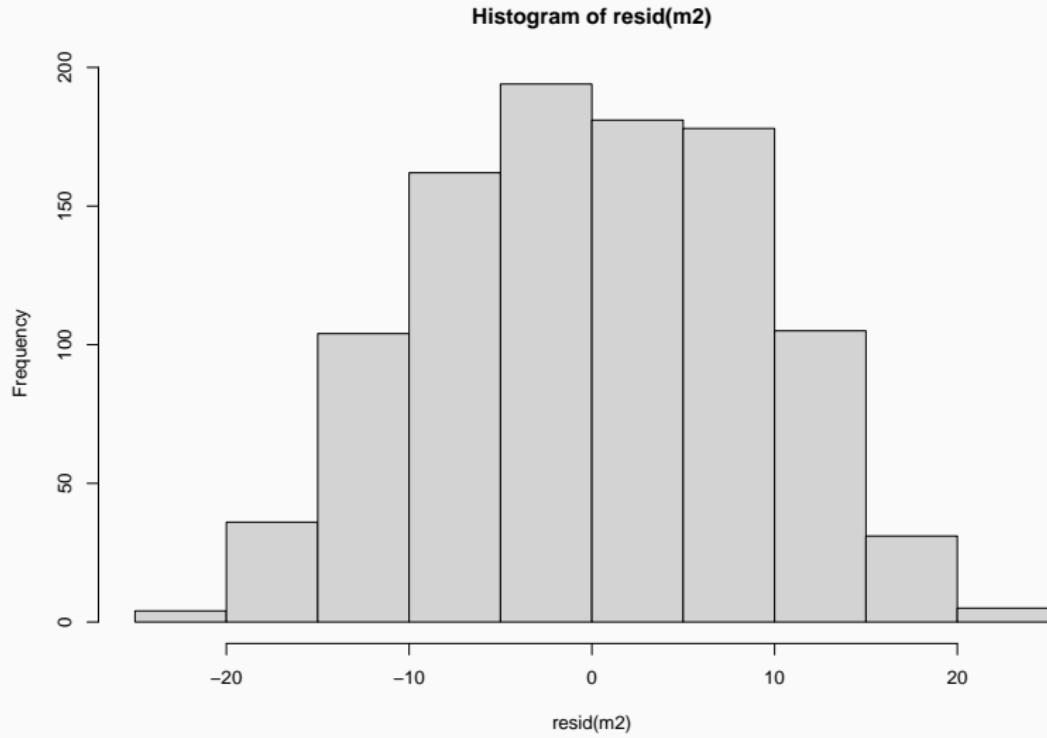
```
plot_model(m2, type = "eff")
```

\$sex

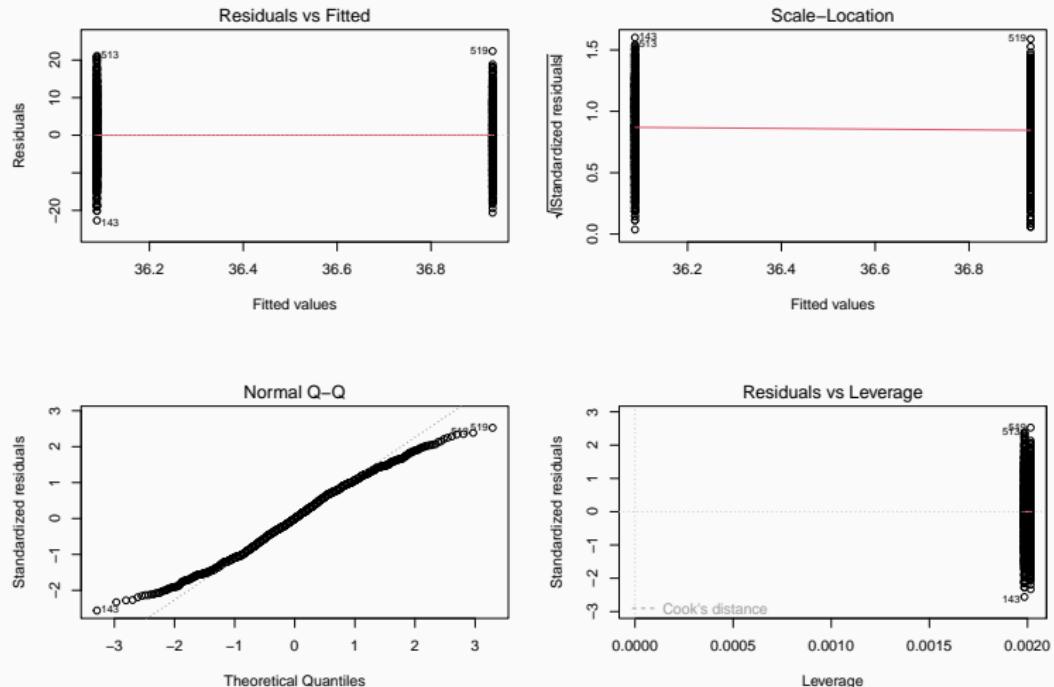


Model checking: residuals

```
hist(resid(m2))
```

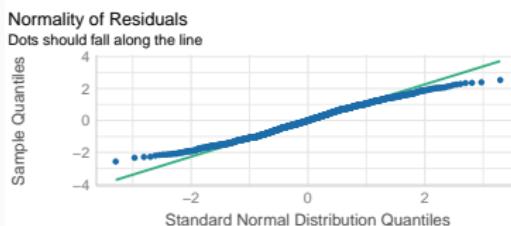
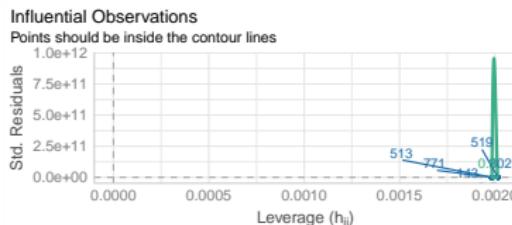
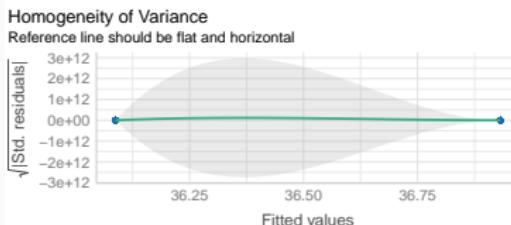
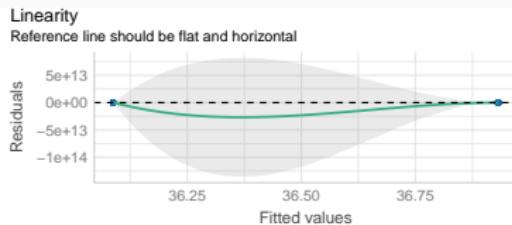
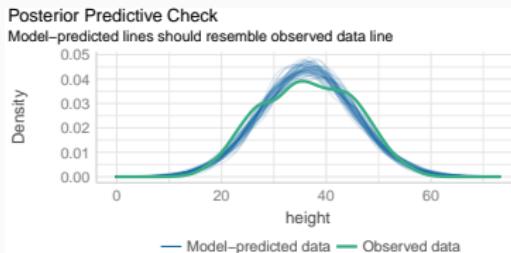


Model checking: residuals



Model checking

```
library("performance")
check_model(m2)
```



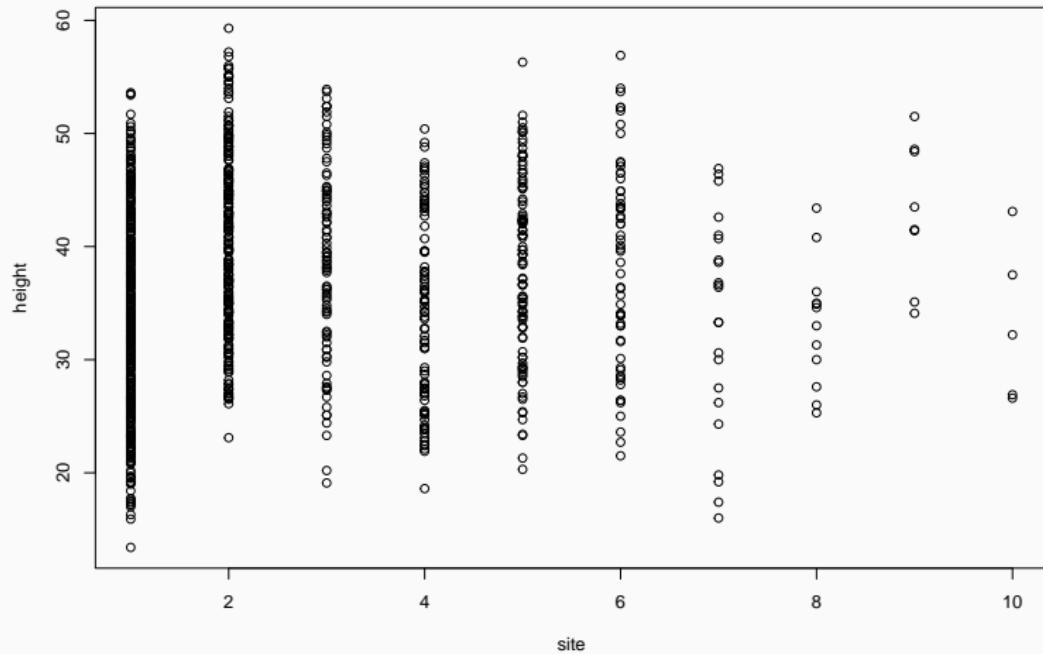
Model dashboard

```
model_dashboard(m2)
```

Q: Does height differ among field sites?

Plot data first

```
plot(height ~ site, data = trees)
```



Linear model with categorical predictors

```
m3 <- lm(height ~ site, data = trees)
```

$$y_i = a + b_{site2} + c_{site3} + d_{site4} + e_{site5} + \dots + \varepsilon_i$$
$$\varepsilon_i \sim N(0, \sigma^2)$$

Model Height ~ site

All right here?

```
m3 <- lm(height ~ site, data = trees)
```

Call:

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

Residuals:

| Min | 1Q | Median | 3Q | Max |
|----------|---------|--------|--------|---------|
| -22.4498 | -6.7049 | 0.0709 | 6.7537 | 23.0640 |

Coefficients:

| | Estimate | Std. Error | t value | Pr(> t) | | | | | | | |
|----------------|----------|------------|---------|-------------|------|-----|------|-----|-----|----|---|
| (Intercept) | 35.4636 | 0.4730 | 74.975 | < 2e-16 *** | | | | | | | |
| site | 0.3862 | 0.1413 | 2.733 | 0.00639 ** | | | | | | | |
| --- | | | | | | | | | | | |
| Signif. codes: | 0 | '***' | 0.001 | '**' | 0.01 | '*' | 0.05 | '.' | 0.1 | '' | 1 |

Residual standard error: 8.842 on 998 degrees of freedom

Multiple R-squared: 0.007429, Adjusted R-squared: 0.006435

F-statistic: 7.47 on 1 and 998 DF, p-value: 0.006385

Let's check model structure with `equatiomatic`

```
extract_eq(m3)
```

$$\text{height} = \alpha + \beta_1(\text{site}) + \epsilon \quad (3)$$

site is a factor!

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

Let's check model structure with `equatiomatic`

```
m3 <- lm(height ~ site, data = trees)  
extract_eq(m3)
```

$$\text{height} = \alpha + \beta_1(\text{site}_2) + \beta_2(\text{site}_3) + \beta_3(\text{site}_4) + \beta_4(\text{site}_5) + \beta_5(\text{site}_6) + \beta_6(\text{site}_7) + \beta_7(\text{site}_8) \quad (4)$$

Model Height ~ site

Call:

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

Residuals:

| Min | 1Q | Median | 3Q | Max |
|----------|---------|--------|--------|---------|
| -20.4416 | -6.9004 | 0.0379 | 6.3051 | 19.7584 |

Coefficients:

| | Estimate | Std. Error | t value | Pr(> t) |
|-------------|----------|------------|---------|--------------|
| (Intercept) | 33.8416 | 0.4266 | 79.329 | < 2e-16 *** |
| site2 | 6.3411 | 0.7126 | 8.899 | < 2e-16 *** |
| site3 | 4.9991 | 0.9828 | 5.086 | 4.36e-07 *** |
| site4 | 0.5329 | 0.9872 | 0.540 | 0.58949 |
| site5 | 4.3723 | 0.9425 | 4.639 | 3.97e-06 *** |
| site6 | 4.7601 | 1.1709 | 4.065 | 5.18e-05 *** |
| site7 | -0.7416 | 1.8506 | -0.401 | 0.68871 |
| site8 | -0.6832 | 2.4753 | -0.276 | 0.78258 |
| site9 | 9.1709 | 3.0165 | 3.040 | 0.00243 ** |
| site10 | -0.5816 | 3.8013 | -0.153 | 0.87843 |

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

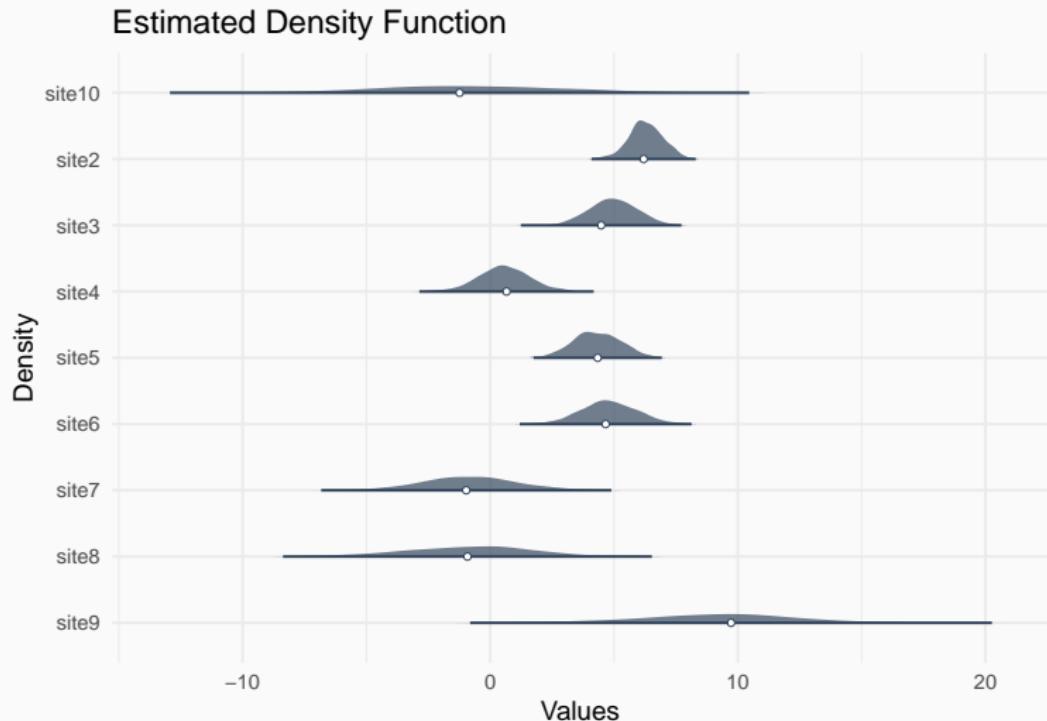
Residual standard error: 8.446 on 990 degrees of freedom

Multiple R-squared: 0.1016, Adjusted R-squared: 0.09344

F-statistic: 12.44 on 9 and 990 DF, p-value: < 2.2e-16

Estimated parameter distributions

```
plot(simulate_parameters(m3), stack = FALSE)
```



Analysing differences among factor levels

```
library("modelbased")
estimate_means(m3)
```

Estimated Marginal Means

| site | Mean | SE | 95% CI |
|-------|-------|------|----------------|
| <hr/> | | | |
| 1 | 33.84 | 0.43 | [33.00, 34.68] |
| 2 | 40.18 | 0.57 | [39.06, 41.30] |
| 3 | 38.84 | 0.89 | [37.10, 40.58] |
| 4 | 34.37 | 0.89 | [32.63, 36.12] |
| 5 | 38.21 | 0.84 | [36.56, 39.86] |
| 6 | 38.60 | 1.09 | [36.46, 40.74] |
| 7 | 33.10 | 1.80 | [29.57, 36.63] |
| 8 | 33.16 | 2.44 | [28.37, 37.94] |
| 9 | 43.01 | 2.99 | [37.15, 48.87] |
| 10 | 33.26 | 3.78 | [25.85, 40.67] |

Analysing differences among factor levels

For finer control see `emmeans` package

```
estimate_contrasts(m3)
```

Marginal Contrasts Analysis

| Level1 | Level2 | Difference | 95% CI | SE | t(990) | p |
|--------|--------|------------|-----------------|------|--------|--------|
| site1 | site10 | 0.58 | [-11.85, 13.01] | 3.80 | 0.15 | > .999 |
| site1 | site2 | -6.34 | [-8.67, -4.01] | 0.71 | -8.90 | < .001 |
| site1 | site3 | -5.00 | [-8.21, -1.78] | 0.98 | -5.09 | < .001 |
| site1 | site4 | -0.53 | [-3.76, 2.70] | 0.99 | -0.54 | > .999 |
| site1 | site5 | -4.37 | [-7.45, -1.29] | 0.94 | -4.64 | < .001 |
| site1 | site6 | -4.76 | [-8.59, -0.93] | 1.17 | -4.07 | 0.002 |
| site1 | site7 | 0.74 | [-5.31, 6.79] | 1.85 | 0.40 | > .999 |
| site1 | site8 | 0.68 | [-7.41, 8.78] | 2.48 | 0.28 | > .999 |
| site1 | site9 | -9.17 | [-19.04, 0.69] | 3.02 | -3.04 | 0.073 |
| site2 | site10 | 6.92 | [-5.57, 19.42] | 3.82 | 1.81 | 0.728 |
| site2 | site3 | 1.34 | [-2.10, 4.79] | 1.05 | 1.27 | 0.959 |
| site2 | site4 | 5.81 | [2.35, 9.27] | 1.06 | 5.49 | < .001 |
| site2 | site5 | 1.97 | [-1.35, 5.29] | 1.02 | 1.94 | 0.643 |
| site2 | site6 | 1.58 | [-2.44, 5.61] | 1.23 | 1.28 | 0.957 |
| site2 | site7 | 7.08 | [0.90, 13.26] | 1.89 | 3.75 | 0.007 |
| site2 | site8 | 7.02 | [-1.17, 15.21] | 2.50 | 2.81 | 0.136 |
| site2 | site9 | -2.83 | [-12.77, 7.11] | 3.04 | -0.93 | 0.995 |
| site3 | site10 | 5.58 | [-7.11, 18.27] | 3.88 | 1.44 | 0.915 |
| site3 | site4 | 4.47 | [0.36, 8.57] | 1.26 | 3.56 | 0.014 |
| site3 | site5 | 0.63 | [-3.37, 4.62] | 1.22 | 0.51 | > .999 |
| site3 | site6 | 0.24 | [-4.35, 4.83] | 1.40 | 0.17 | > .999 |
| site3 | site7 | 5.74 | [-0.82, 12.30] | 2.01 | 2.86 | 0.118 |
| site3 | site8 | 5.68 | [-2.80, 14.17] | 2.59 | 2.19 | 0.464 |
| site3 | site9 | -4.17 | [-14.36, 6.01] | 3.11 | -1.34 | 0.944 |
| site4 | site10 | 1.11 | [-11.58, 13.81] | 3.88 | 0.29 | > .999 |
| site4 | site5 | -3.84 | [-7.84, 0.16] | 1.22 | -3.14 | 0.055 |
| site4 | site6 | -4.23 | [-8.83, 0.38] | 1.41 | -3.00 | 0.081 |

Presenting model results

```
kable(xtable::xtable(m3), digits = 2)
```

| | Estimate | Std. Error | t value | Pr(> t) |
|-------------|----------|------------|---------|----------|
| (Intercept) | 33.84 | 0.43 | 79.33 | 0.00 |
| site2 | 6.34 | 0.71 | 8.90 | 0.00 |
| site3 | 5.00 | 0.98 | 5.09 | 0.00 |
| site4 | 0.53 | 0.99 | 0.54 | 0.59 |
| site5 | 4.37 | 0.94 | 4.64 | 0.00 |
| site6 | 4.76 | 1.17 | 4.07 | 0.00 |
| site7 | -0.74 | 1.85 | -0.40 | 0.69 |
| site8 | -0.68 | 2.48 | -0.28 | 0.78 |
| site9 | 9.17 | 3.02 | 3.04 | 0.00 |
| site10 | -0.58 | 3.80 | -0.15 | 0.88 |

Estimated tree heights for each site

```
summary(allEffects(m3))
```

model: height ~ site

site effect

site

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|----------|----------|----------|----------|----------|----------|----------|----------|---|
| 33.84158 | 40.18265 | 38.84066 | 34.37444 | 38.21386 | 38.60167 | 33.10000 | 33.15833 | |
| 9 | 10 | | | | | | | |
| 43.01250 | 33.26000 | | | | | | | |

Lower 95 Percent Confidence Limits

site

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|----------|----------|----------|----------|----------|----------|----------|----------|---|
| 33.00444 | 39.06264 | 37.10317 | 32.62733 | 36.56463 | 36.46190 | 29.56629 | 28.37367 | |
| 9 | 10 | | | | | | | |
| 37.15251 | 25.84764 | | | | | | | |

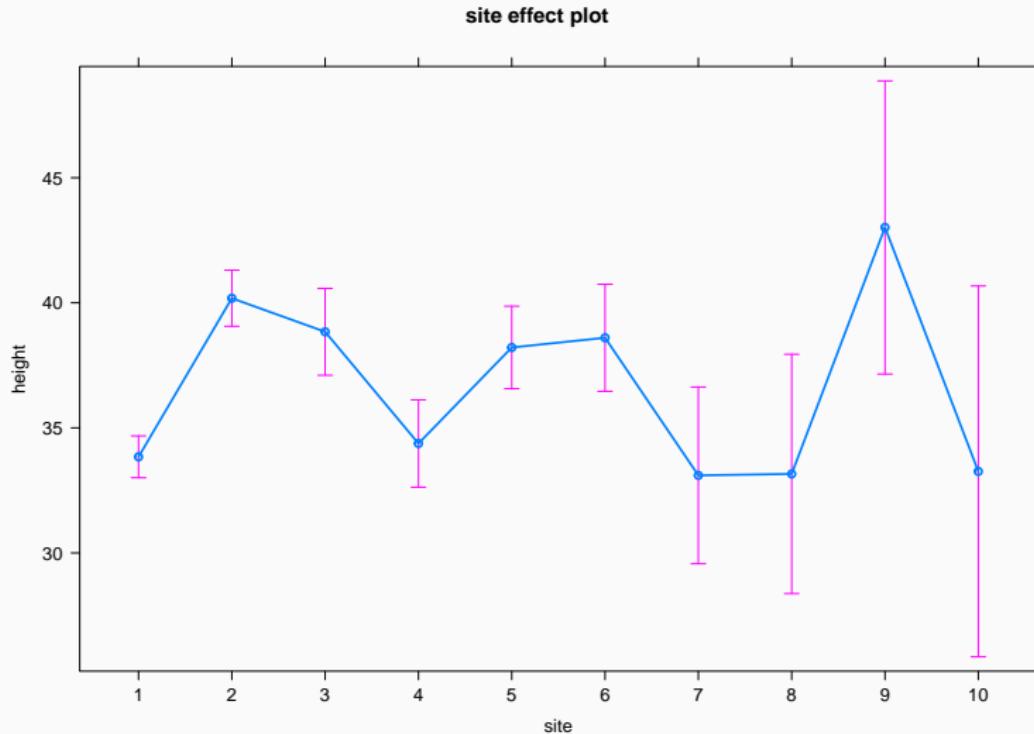
Upper 95 Percent Confidence Limits

site

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|----------|----------|----------|----------|----------|----------|----------|----------|---|
| 34.67872 | 41.30265 | 40.57814 | 36.12156 | 39.86309 | 40.74143 | 36.63371 | 37.94299 | |
| 9 | 10 | | | | | | | |
| 48.87249 | 40.67236 | | | | | | | |

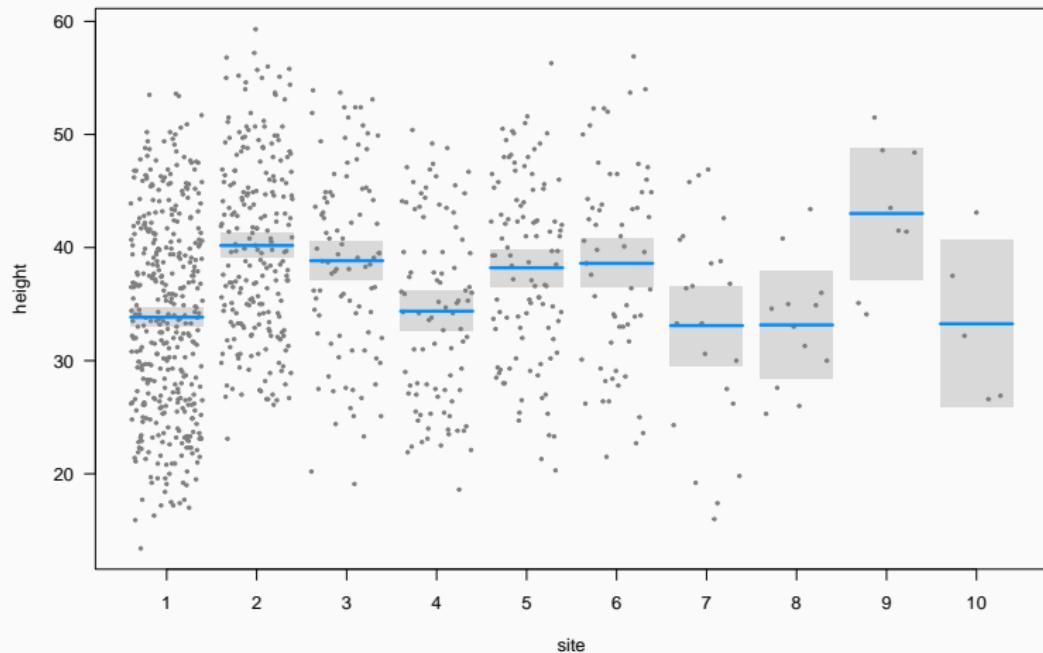
Plot

```
plot(allEffects(m3))
```



Plot (visreg)

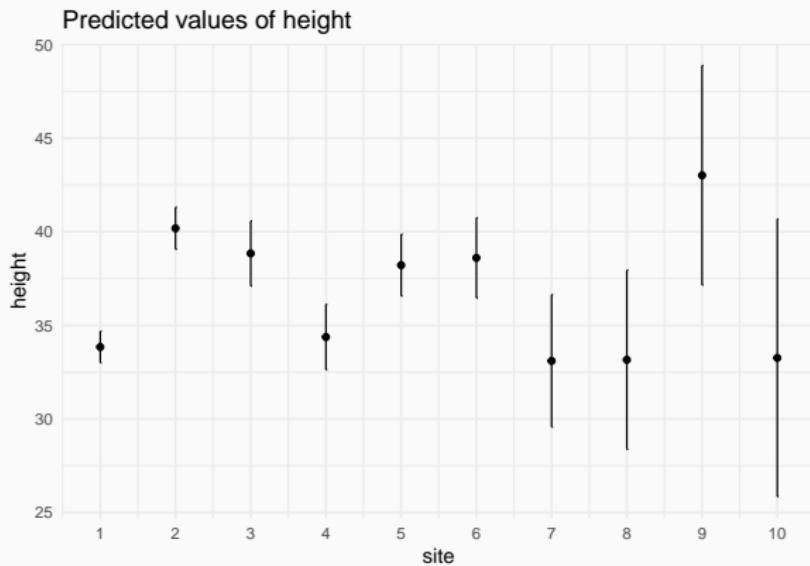
```
visreg(m3)
```



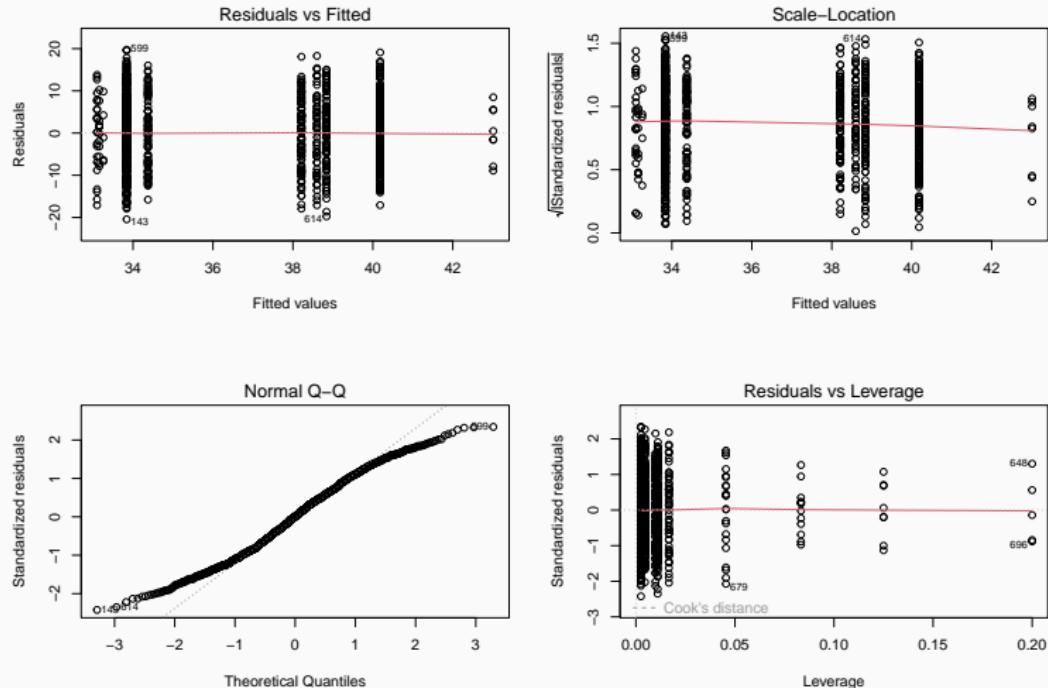
Plot model (sjPlot)

```
plot_model(m3, type = "eff")
```

`$site`

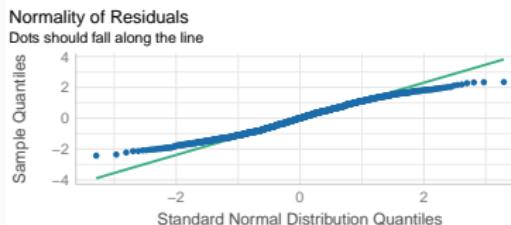
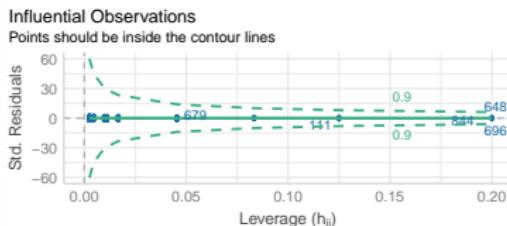
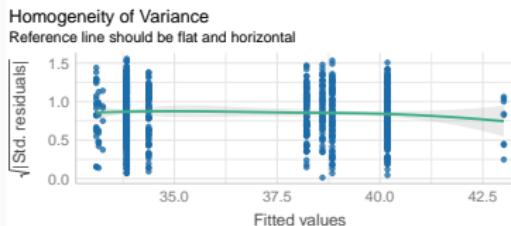
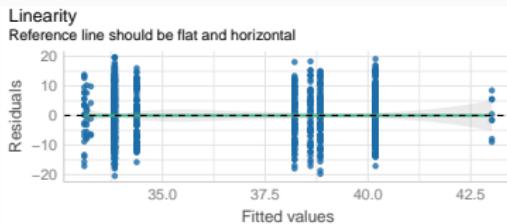
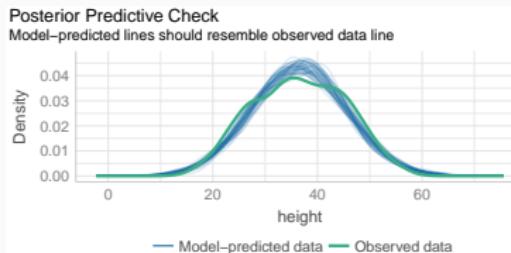


Model checking: residuals



Model checking: residuals

```
check_model(m3)
```



Combining continuous and categorical predictors

Predicting tree height based on dbh and site

```
lm(height ~ site + dbh, data = trees)
```

corresponds to

$$y_i = a + b_{site2} + c_{site3} + d_{site4} + e_{site5} + \dots + k \cdot DBH_i + \varepsilon_i$$
$$\varepsilon_i \sim N(0, \sigma^2)$$

Predicting tree height based on dbh and site

Call:

```
lm(formula = height ~ 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 *** | | | | | | | |
| site2 | 6.504303 | 0.256730 | 25.335 | < 2e-16 *** | | | | | | | |
| site3 | 4.357457 | 0.354181 | 12.303 | < 2e-16 *** | | | | | | | |
| site4 | 1.934650 | 0.356102 | 5.433 | 6.98e-08 *** | | | | | | | |
| site5 | 3.637432 | 0.339688 | 10.708 | < 2e-16 *** | | | | | | | |
| site6 | 4.204511 | 0.421906 | 9.966 | < 2e-16 *** | | | | | | | |
| site7 | -0.176193 | 0.666772 | -0.264 | 0.7916 | | | | | | | |
| site8 | -5.312648 | 0.893603 | -5.945 | 3.82e-09 *** | | | | | | | |
| site9 | 5.437049 | 1.087766 | 4.998 | 6.84e-07 *** | | | | | | | |
| site10 | 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

Presenting model results

```
parameters(m4)
```

| Parameter | Coefficient | SE | 95% CI | t(989) | p |
|-------------|-------------|----------|----------------|--------|--------|
| <hr/> | | | | | |
| (Intercept) | 16.70 | 0.26 | [16.19, 17.21] | 64.09 | < .001 |
| site [2] | 6.50 | 0.26 | [6.00, 7.01] | 25.34 | < .001 |
| site [3] | 4.36 | 0.35 | [3.66, 5.05] | 12.30 | < .001 |
| site [4] | 1.93 | 0.36 | [1.24, 2.63] | 5.43 | < .001 |
| site [5] | 3.64 | 0.34 | [2.97, 4.30] | 10.71 | < .001 |
| site [6] | 4.20 | 0.42 | [3.38, 5.03] | 9.97 | < .001 |
| site [7] | -0.18 | 0.67 | [-1.48, 1.13] | -0.26 | 0.792 |
| site [8] | -5.31 | 0.89 | [-7.07, -3.56] | -5.95 | < .001 |
| site [9] | 5.44 | 1.09 | [3.30, 7.57] | 5.00 | < .001 |
| site [10] | 2.26 | 1.37 | [-0.43, 4.95] | 1.65 | 0.099 |
| dbh | 0.62 | 7.57e-03 | [0.60, 0.63] | 81.47 | < .001 |

Estimated tree heights for each site

```
summary(allEffects(m4))
```

model: height ~ site + dbh

site effect

site

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|----------|----------|----------|----------|----------|----------|----------|----------|---|
| 33.90437 | 40.40868 | 38.26183 | 35.83902 | 37.54181 | 38.10889 | 33.72818 | 28.59173 | |
| 9 | 10 | | | | | | | |
| 39.34142 | 36.16771 | | | | | | | |

Lower 95 Percent Confidence Limits

site

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|----------|----------|----------|----------|----------|----------|----------|----------|---|
| 33.60276 | 40.00512 | 37.63569 | 35.20858 | 36.94739 | 37.33787 | 32.45495 | 26.86438 | |
| 9 | 10 | | | | | | | |
| 37.22831 | 33.49623 | | | | | | | |

Upper 95 Percent Confidence Limits

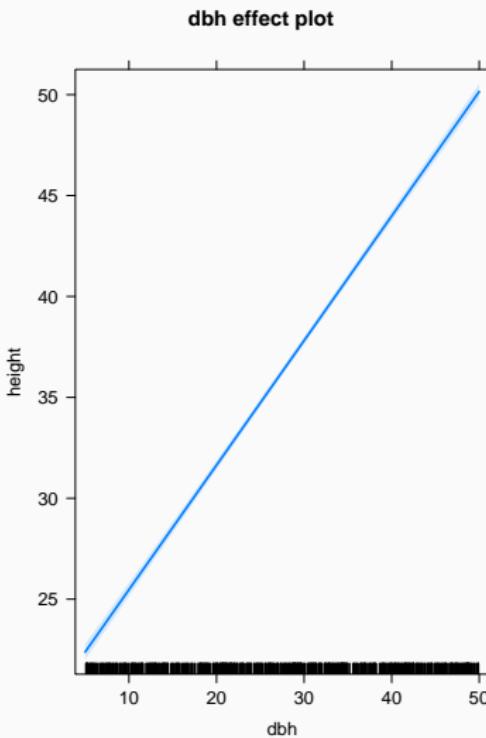
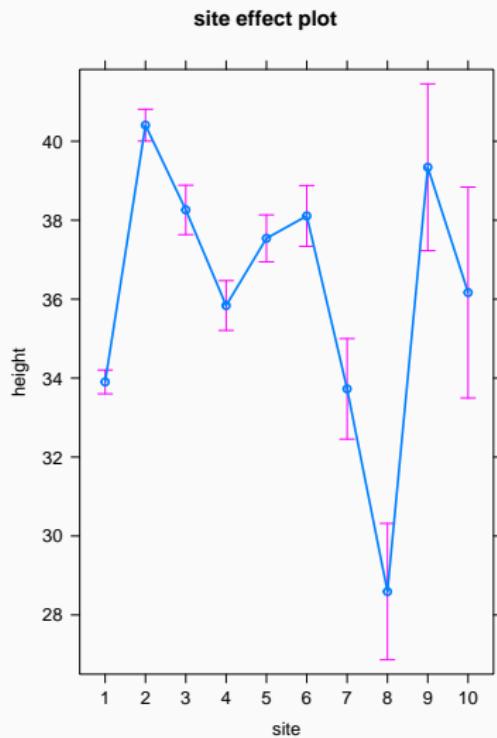
site

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|----------|----------|----------|----------|----------|----------|----------|----------|---|
| 34.20599 | 40.81223 | 38.88798 | 36.46947 | 38.13622 | 38.87990 | 35.00141 | 30.31907 | |
| 9 | 10 | | | | | | | |
| 41.45454 | 38.83919 | | | | | | | |

dbh effect

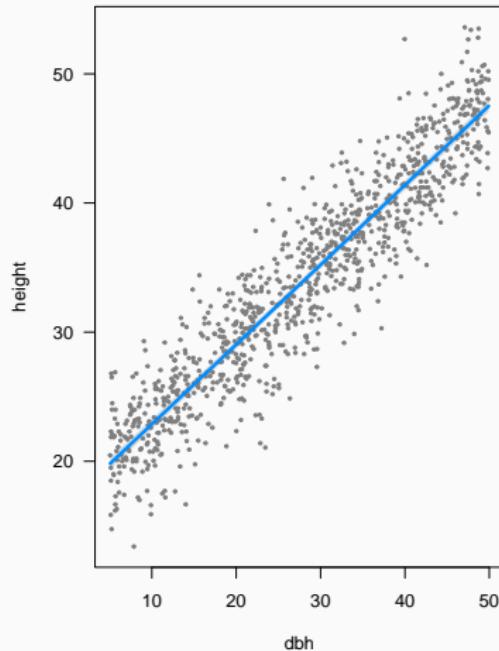
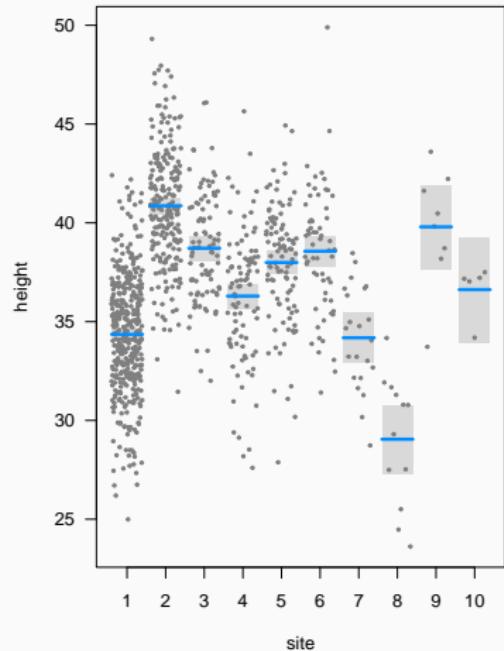
Plot

```
plot(allEffects(m4))
```



Plot (visreg)

```
visreg(m4)
```

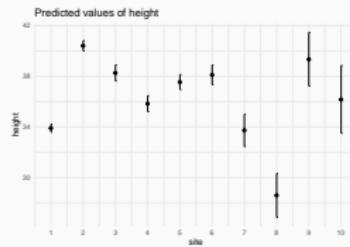


null device

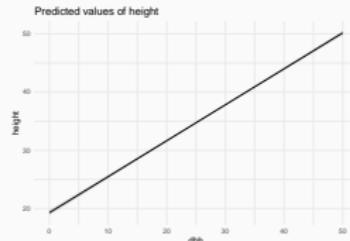
Plot model (sjPlot)

```
plot_model(m4, type = "eff")
```

`$site`

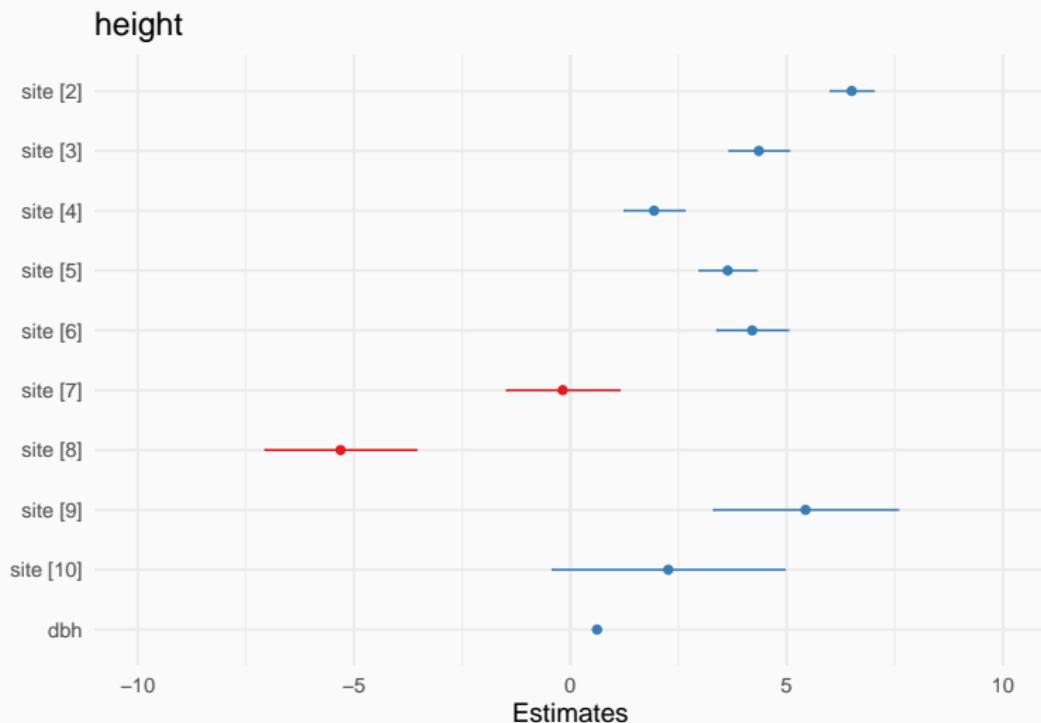


`$dbh`



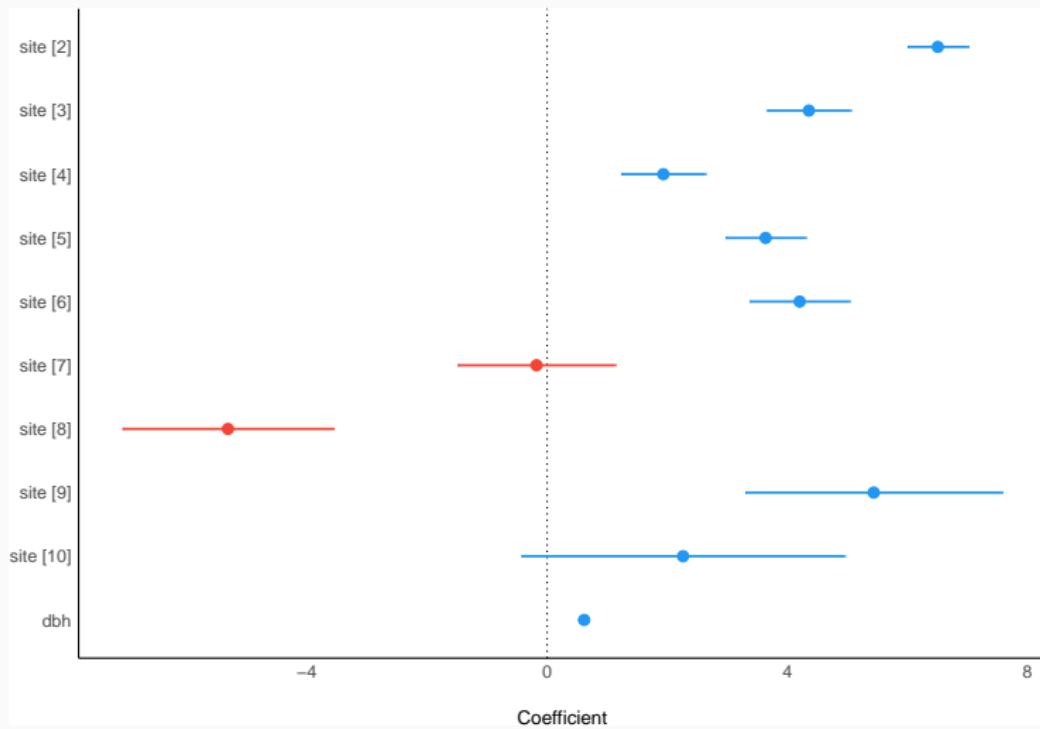
Plot model (sjPlot)

```
plot_model(m4, type = "est")
```

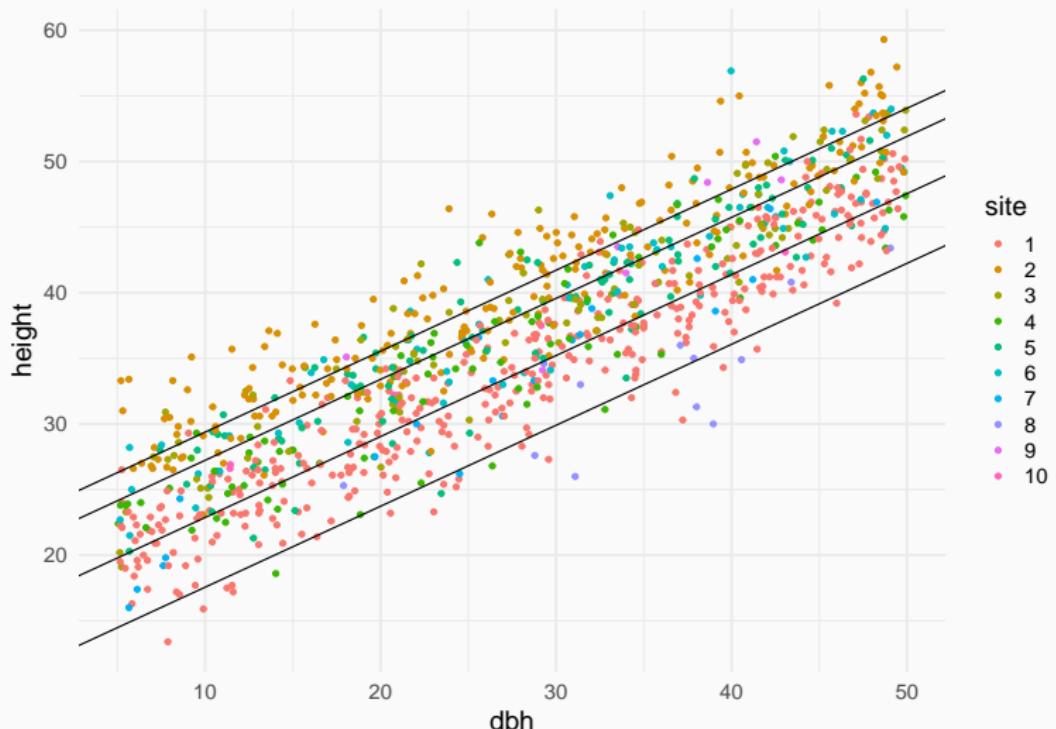


Plot model (see)

```
plot(parameters(m4))
```



We have fitted model w/ many intercepts and single slope



Slope is the same for all sites

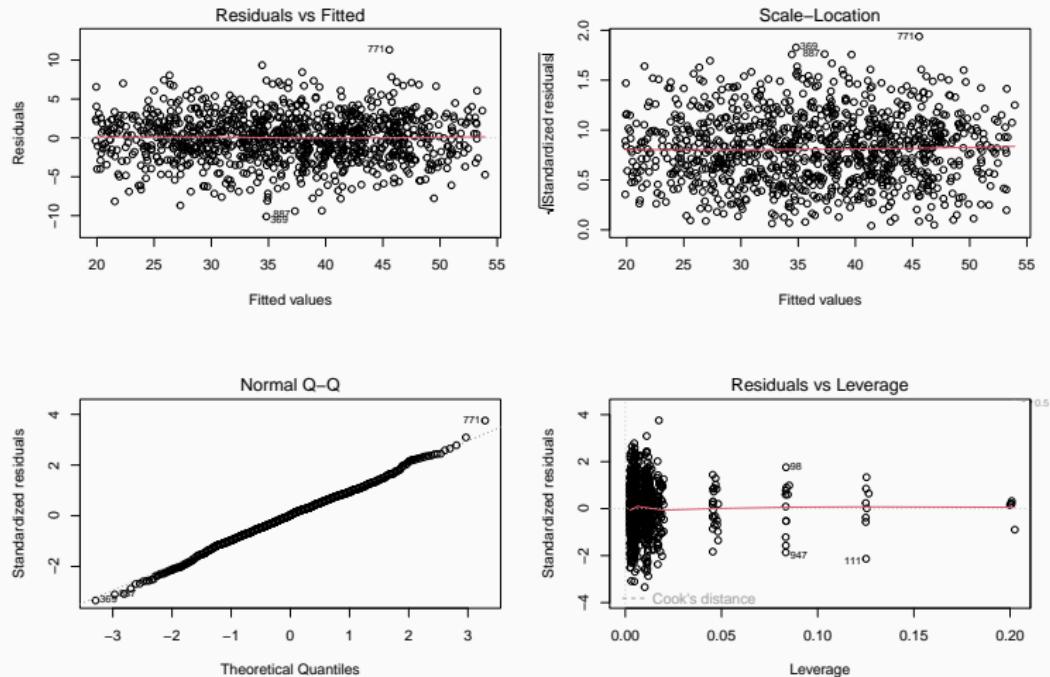
```
estimate_slopes(m4)
```

Estimated Marginal Effects

| Coefficient | SE | 95% CI | t(989) | p |
|-------------|----------|--------------|--------|--------|
| 0.62 | 7.57e-03 | [0.60, 0.63] | 81.47 | < .001 |

Marginal effects estimated for dbh

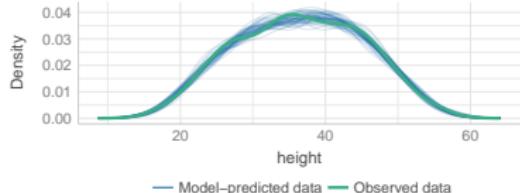
Model checking: residuals



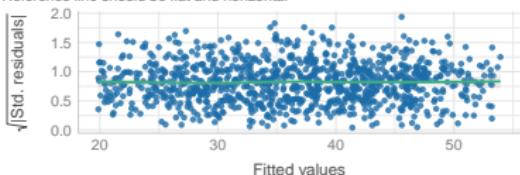
Model checking: residuals

```
check_model(m4)
```

Posterior Predictive Check
Model-predicted lines should resemble observed data line



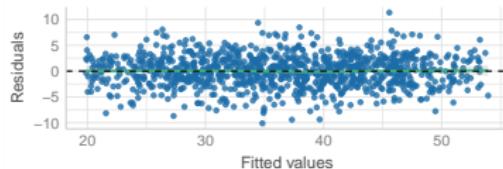
Homogeneity of Variance
Reference line should be flat and horizontal



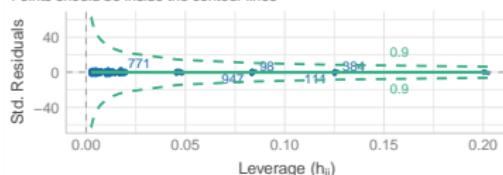
Collinearity
High collinearity (VIF) may inflate parameter uncertainty



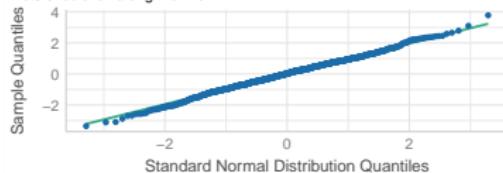
Linearity
Reference line should be flat and horizontal



Influential Observations
Points should be inside the contour lines



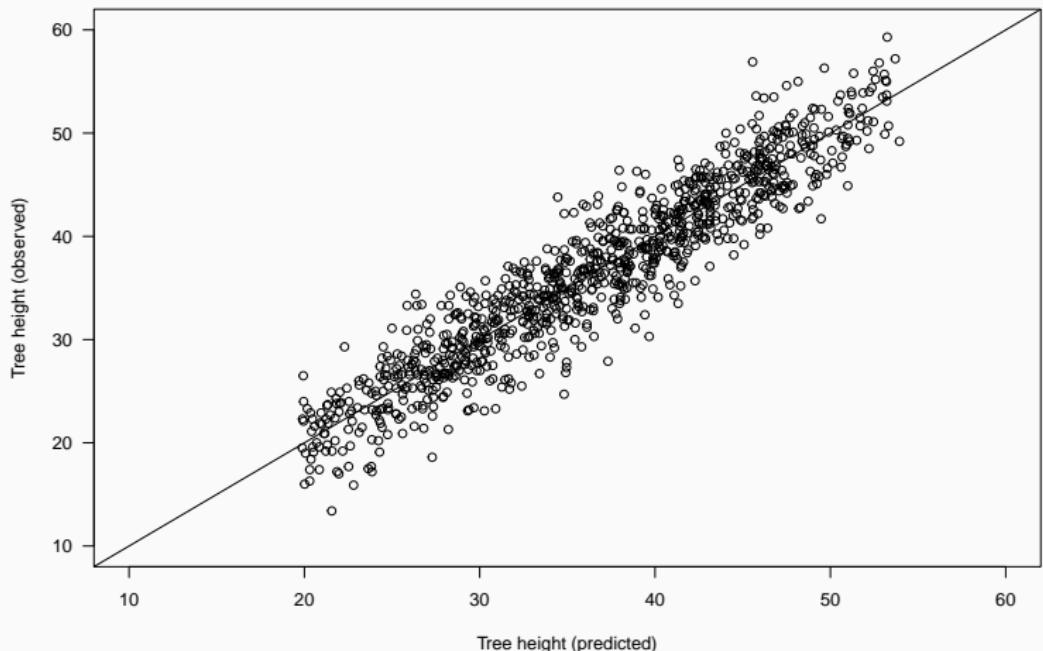
Normality of Residuals
Dots should fall along the line



● Low (< 5) ● Moderate (< 10) ● High (... 10)

How good is this model? Calibration plot

```
trees$height.pred <- fitted(m4)
plot(trees$height.pred, trees$height, xlab = "Tree height (predicted)",
      abline(a = 0, b = 1)
```

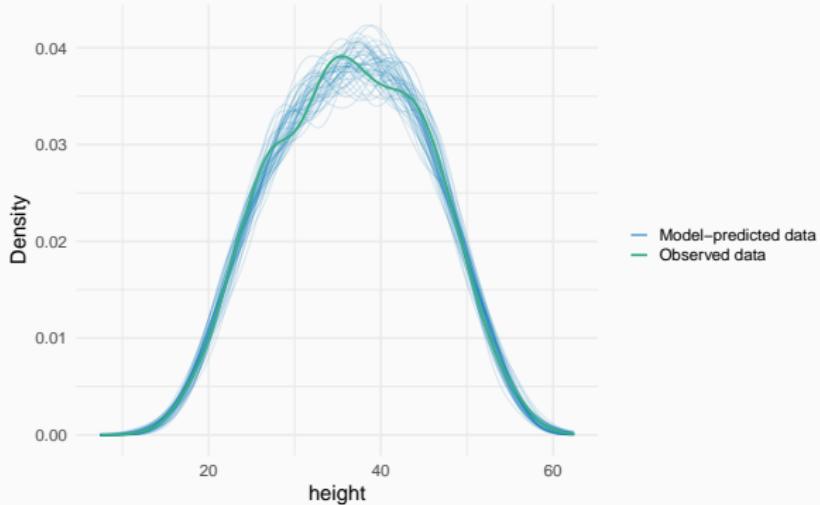


Posterior predictive checking

Simulating response data from fitted model (y_{rep})
and comparing with observed response (y)

```
performance::check_predictions(m4)
```

Posterior Predictive Check
Model-predicted lines should resemble observed data line



Using model for prediction

Expected height of 10-cm diameter tree in each site?

```
trees.10cm <- data.frame(site = as.factor(1:10),  
                           dbh = 10)  
trees.10cm
```

| | site | dbh |
|----|------|-----|
| 1 | 1 | 10 |
| 2 | 2 | 10 |
| 3 | 3 | 10 |
| 4 | 4 | 10 |
| 5 | 5 | 10 |
| 6 | 6 | 10 |
| 7 | 7 | 10 |
| 8 | 8 | 10 |
| 9 | 9 | 10 |
| 10 | 10 | 10 |

Using model for prediction

Confidence interval

```
predict(m4, newdata = trees.10cm, interval = "confidence")
```

| | fit | lwr | upr |
|----|----------|----------|----------|
| 1 | 22.86979 | 22.46878 | 23.27079 |
| 2 | 29.37409 | 28.89388 | 29.85430 |
| 3 | 27.22724 | 26.54160 | 27.91289 |
| 4 | 24.80444 | 24.13410 | 25.47477 |
| 5 | 26.50722 | 25.84952 | 27.16492 |
| 6 | 27.07430 | 26.25490 | 27.89370 |
| 7 | 22.69359 | 21.39601 | 23.99117 |
| 8 | 17.55714 | 15.79282 | 19.32146 |
| 9 | 28.30683 | 26.16606 | 30.44761 |
| 10 | 25.13312 | 22.45540 | 27.81085 |

Using model for prediction

Prediction interval (accounting for residual variance)

```
predict(m4, newdata = trees.10cm, interval = "prediction")
```

| | fit | lwr | upr |
|----|----------|----------|----------|
| 1 | 22.86979 | 16.88478 | 28.85480 |
| 2 | 29.37409 | 23.38325 | 35.36493 |
| 3 | 27.22724 | 21.21645 | 33.23804 |
| 4 | 24.80444 | 18.79537 | 30.81350 |
| 5 | 26.50722 | 20.49955 | 32.51489 |
| 6 | 27.07430 | 21.04678 | 33.10181 |
| 7 | 22.69359 | 16.58268 | 28.80451 |
| 8 | 17.55714 | 11.33039 | 23.78388 |
| 9 | 28.30683 | 21.96314 | 34.65053 |
| 10 | 25.13312 | 18.58868 | 31.67757 |

Using model for prediction

Prediction interval (99%)

```
predict(m4, newdata = trees.10cm, interval = "prediction",
       level = 0.99)
```

| | fit | lwr | upr |
|----|----------|-----------|----------|
| 1 | 22.86979 | 14.998587 | 30.74098 |
| 2 | 29.37409 | 21.495225 | 37.25295 |
| 3 | 27.22724 | 19.322133 | 35.13235 |
| 4 | 24.80444 | 16.901598 | 32.70727 |
| 5 | 26.50722 | 18.606216 | 34.40822 |
| 6 | 27.07430 | 19.147195 | 35.00140 |
| 7 | 22.69359 | 14.656813 | 30.73037 |
| 8 | 17.55714 | 9.368019 | 25.74626 |
| 9 | 28.30683 | 19.963913 | 36.64976 |
| 10 | 25.13312 | 16.526183 | 33.74007 |

Q: Does allometric relationship
between Height and Diameter
vary among sites?

Model with interactions

Call:

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

Residuals:

| Min | 1Q | Median | 3Q | Max |
|----------|---------|--------|--------|---------|
| -10.1017 | -1.9839 | 0.0645 | 2.0486 | 11.1789 |

Coefficients:

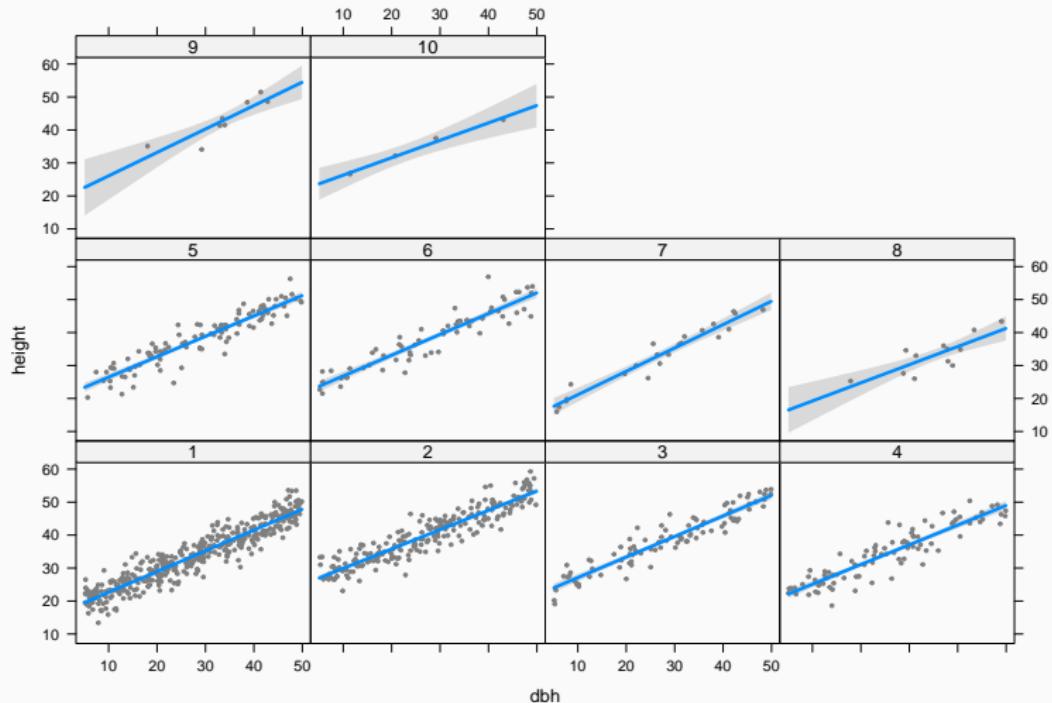
| | Estimate | Std. Error | t value | Pr(> t) | | | | | | | |
|----------------|-----------|------------|---------|--------------|------|-----|------|-----|-----|-----|---|
| (Intercept) | 16.359437 | 0.360054 | 45.436 | < 2e-16 *** | | | | | | | |
| site2 | 7.684781 | 0.609657 | 12.605 | < 2e-16 *** | | | | | | | |
| site3 | 4.518568 | 0.867008 | 5.212 | 2.28e-07 *** | | | | | | | |
| site4 | 2.769336 | 0.813259 | 3.405 | 0.000688 *** | | | | | | | |
| site5 | 3.917607 | 0.870983 | 4.498 | 7.68e-06 *** | | | | | | | |
| site6 | 4.155161 | 1.009379 | 4.117 | 4.17e-05 *** | | | | | | | |
| site7 | -2.306799 | 1.551303 | -1.487 | 0.137334 | | | | | | | |
| site8 | -2.616095 | 4.090671 | -0.640 | 0.522630 | | | | | | | |
| site9 | 2.621560 | 5.073794 | 0.517 | 0.605492 | | | | | | | |
| site10 | 4.662340 | 2.991072 | 1.559 | 0.119378 | | | | | | | |
| dbh | 0.629299 | 0.011722 | 53.685 | < 2e-16 *** | | | | | | | |
| site2:dbh | -0.042784 | 0.020033 | -2.136 | 0.032950 * | | | | | | | |
| site3:dbh | -0.006031 | 0.027640 | -0.218 | 0.827312 | | | | | | | |
| site4:dbh | -0.031633 | 0.028225 | -1.121 | 0.262677 | | | | | | | |
| site5:dbh | -0.010173 | 0.027887 | -0.365 | 0.715334 | | | | | | | |
| site6:dbh | 0.001337 | 0.032109 | 0.042 | 0.966797 | | | | | | | |
| site7:dbh | 0.079728 | 0.052056 | 1.532 | 0.125951 | | | | | | | |
| site8:dbh | -0.079027 | 0.113386 | -0.697 | 0.485984 | | | | | | | |
| site9:dbh | 0.081035 | 0.146649 | 0.553 | 0.580679 | | | | | | | |
| site10:dbh | -0.101107 | 0.114520 | -0.883 | 0.377522 | | | | | | | |
| --- | | | | | | | | | | | |
| Signif. codes: | 0 | '***' | 0.001 | '**' | 0.01 | '*' | 0.05 | '.' | 0.1 | ' ' | 1 |

Residual standard error: 3.041 on 980 degrees of freedom

Multiple R-squared: 0.8847 Adjusted R-squared: 0.8825

Does slope vary among sites?

```
visreg(m5, xvar = "dbh", by = "site")
```



Examining fitted model with {modelStudio}

```
library("modelStudio")
m5.explain <- DALEX::explain(m5, data = trees, y = trees$height)
modelStudio(m5.explain)
```

Extra exercises

- [paperplanes](#): How does flight distance differ with age, gender or paper type?

Extra exercises

- [paperplanes](#): How does flight distance differ with age, gender or paper type?
- [mammal sleep](#): Are sleep patterns related to diet?

Extra exercises

- [paperplanes](#): How does flight distance differ with age, gender or paper type?
- [mammal sleep](#): Are sleep patterns related to diet?
- [iris](#): Predict petal length ~ petal width and species

Extra exercises

- [paperplanes](#): How does flight distance differ with age, gender or paper type?
- [mammal sleep](#): Are sleep patterns related to diet?
- [iris](#): Predict petal length ~ petal width and species
- [Penguins data](#): Body mass ~ Flipper length, Bill length ~ Bill depth, differences across sites...

Extra exercises

- [paperplanes](#): How does flight distance differ with age, gender or paper type?
- [mammal sleep](#): Are sleep patterns related to diet?
- [iris](#): Predict petal length ~ petal width and species
- [Penguins data](#): Body mass ~ Flipper length, Bill length ~ Bill depth, differences across sites...
- [racing pigeons](#): is speed related to sex?

Variable and model selection

Francisco Rodríguez-Sánchez

<https://frodriguezsanchez.net>

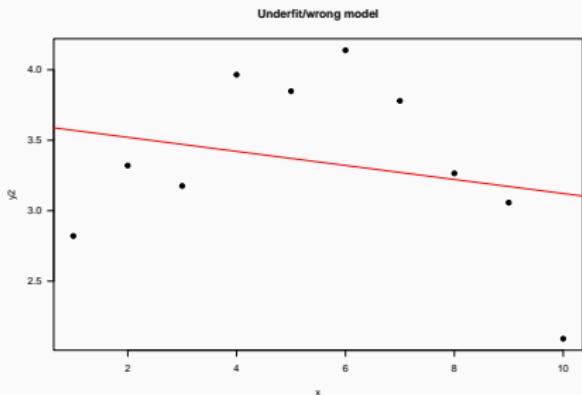
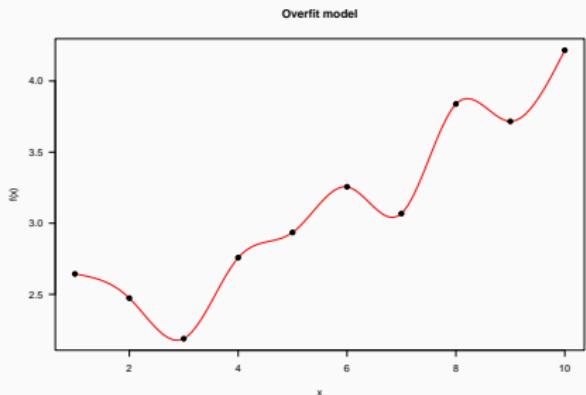
Overfitting and balanced model complexity

- On one hand, we want to **maximise fit**.

Overfitting and balanced model complexity

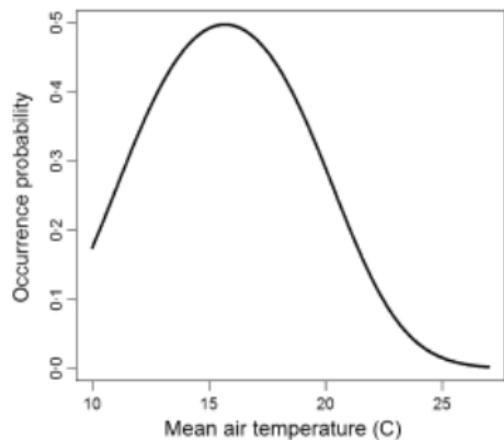
- On one hand, we want to **maximise fit**.
- On the other hand, we want to **avoid overfitting** and overly complex models.

Overfitting and balanced model complexity

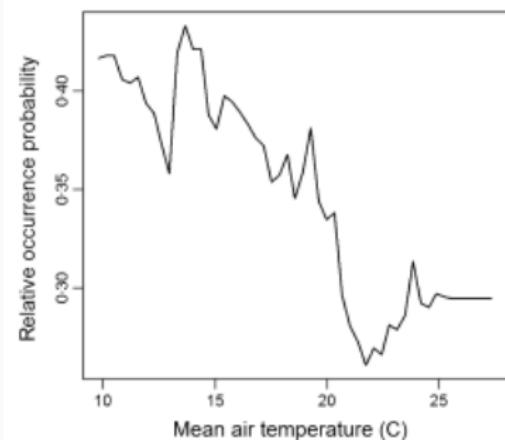


Overfitting and balanced model complexity

GLMM



Random forests



Wenger & Olden (2012)

Overfitted models will work badly on new data



Evaluating models' predictive accuracy

- Cross-validation (k-fold, leave one out...)

Evaluating models' predictive accuracy

- Cross-validation (k-fold, leave one out...)
- Information Criteria:

Evaluating models' predictive accuracy

- Cross-validation (k-fold, leave one out...)
- Information Criteria:
 - AIC

Evaluating models' predictive accuracy

- Cross-validation (k-fold, leave one out...)
- Information Criteria:
 - AIC
 - BIC

Evaluating models' predictive accuracy

- Cross-validation (k-fold, leave one out...)
- Information Criteria:
 - AIC
 - BIC
 - DIC

Evaluating models' predictive accuracy

- Cross-validation (k-fold, leave one out...)
- Information Criteria:
 - AIC
 - BIC
 - DIC
 - WAIC...

Evaluating models' predictive accuracy

- Cross-validation (k-fold, leave one out...)
- Information Criteria:
 - AIC
 - BIC
 - DIC
 - WAIC...
- All these methods have flaws!

AIC (Akaike Information Criteria)

$$AIC = -2 * \text{LogLikelihood} + 2K$$

- First term: **model fit**

AIC (Akaike Information Criteria)

$$AIC = -2 * \text{LogLikelihood} + 2K$$

- First term: **model fit**
- **K** = **number of parameters** (penalisation for model complexity)

AIC (Akaike Information Criteria)

$$AIC = -2 * \text{LogLikelihood} + 2K$$

- First term: **model fit**
- **K = number of parameters** (penalisation for model complexity)
- Lower is better

AIC (Akaike Information Criteria)

$$AIC = -2 * \text{LogLikelihood} + 2K$$

- First term: **model fit**
- **K = number of parameters** (penalisation for model complexity)
- Lower is better
- AIC biased towards complex models.

AIC (Akaike Information Criteria)

$$AIC = -2 * \text{LogLikelihood} + 2K$$

- First term: **model fit**
- **K = number of parameters** (penalisation for model complexity)
- Lower is better
- AIC biased towards complex models.
- AICc recommended with ‘small’ sample sizes ($n/p < 40$). But see [Richards 2005](#)

Problems of IC

- No information criteria is panacea: all have problems.

Problems of IC

- No information criteria is panacea: all have problems.
- They estimate *average* out-of-sample prediction error. But errors can differ substantially within dataset.

Problems of IC

- No information criteria is panacea: all have problems.
- They estimate *average* out-of-sample prediction error. But errors can differ substantially within dataset.
- Sometimes better models rank poorly (e.g. see [Gelman et al. 2013](#)). Combine with **thorough model checks**.

So which variables should enter
my model?

Choosing predictors

- Choose variables based on **background knowledge**, rather than throwing plenty of them in a fishing expedition.

Choosing predictors

- Choose variables based on **background knowledge**, rather than throwing plenty of them in a fishing expedition.
- Propose single global model or small set (< 10 - 20) of **reasonable** candidate models.

Choosing predictors

- Choose variables based on **background knowledge**, rather than throwing plenty of them in a fishing expedition.
- Propose single global model or small set (< 10 - 20) of **reasonable** candidate models.
- Number of variables **balanced with sample size** (e.g. at least 10 - 30 obs per param)

Choosing predictors

- Choose variables based on **background knowledge**, rather than throwing plenty of them in a fishing expedition.
- Propose single global model or small set (< 10 - 20) of **reasonable** candidate models.
- Number of variables **balanced with sample size** (e.g. at least 10 - 30 obs per param)
- Assess collinearity between predictors ([Dormann et al 2013](#))

Choosing predictors

- Choose variables based on **background knowledge**, rather than throwing plenty of them in a fishing expedition.
- Propose single global model or small set (< 10 - 20) of **reasonable** candidate models.
- Number of variables **balanced with sample size** (e.g. at least 10 - 30 obs per param)
- Assess collinearity between predictors ([Dormann et al 2013](#))
 - If $|r| > 0.5 - 0.7$, consider leaving one variable out, but keep it in mind when interpreting model results.

Choosing predictors

- Choose variables based on **background knowledge**, rather than throwing plenty of them in a fishing expedition.
- Propose single global model or small set (< 10 - 20) of **reasonable** candidate models.
- Number of variables **balanced with sample size** (e.g. at least 10 - 30 obs per param)
- Assess collinearity between predictors ([Dormann et al 2013](#))
 - If $|r| > 0.5 - 0.7$, consider leaving one variable out, but keep it in mind when interpreting model results.
 - Or combine 2 or more in a synthetic variable (e.g. water deficit ~ Temp + Precip).

Choosing predictors

- Choose variables based on **background knowledge**, rather than throwing plenty of them in a fishing expedition.
- Propose single global model or small set (< 10 - 20) of **reasonable** candidate models.
- Number of variables **balanced with sample size** (e.g. at least 10 - 30 obs per param)
- Assess collinearity between predictors ([Dormann et al 2013](#))
 - If $|r| > 0.5 - 0.7$, consider leaving one variable out, but keep it in mind when interpreting model results.
 - Or combine 2 or more in a synthetic variable (e.g. water deficit \sim Temp + Precip).
 - Many methods available, e.g. sequential, ridge regression...

Choosing predictors

- Choose variables based on **background knowledge**, rather than throwing plenty of them in a fishing expedition.
- Propose single global model or small set (< 10 - 20) of **reasonable** candidate models.
- Number of variables **balanced with sample size** (e.g. at least 10 - 30 obs per param)
- Assess collinearity between predictors ([Dormann et al 2013](#))
 - If $|r| > 0.5 - 0.7$, consider leaving one variable out, but keep it in mind when interpreting model results.
 - Or combine 2 or more in a synthetic variable (e.g. water deficit \sim Temp + Precip).
 - Many methods available, e.g. sequential, ridge regression...
 - Measurement error can seriously complicate things (Biggs et al 2009; Freckleton 2011)

Choosing predictors

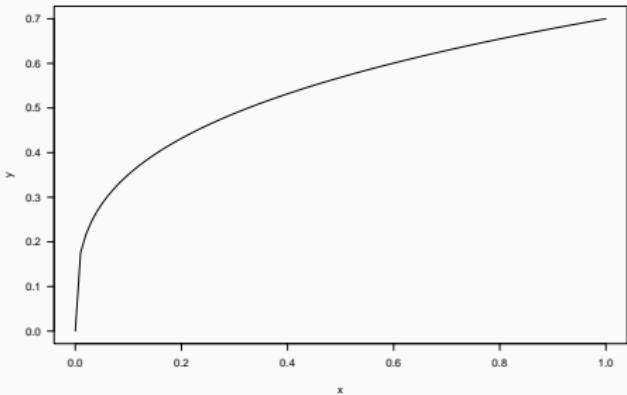
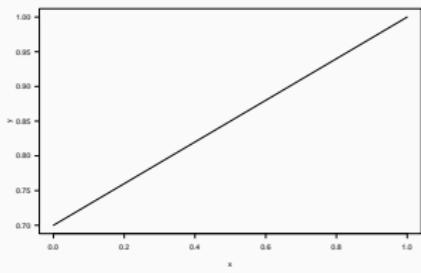
- Choose variables based on **background knowledge**, rather than throwing plenty of them in a fishing expedition.
- Propose single global model or small set (< 10 - 20) of **reasonable** candidate models.
- Number of variables **balanced with sample size** (e.g. at least 10 - 30 obs per param)
- Assess collinearity between predictors ([Dormann et al 2013](#))
 - If $|r| > 0.5 - 0.7$, consider leaving one variable out, but keep it in mind when interpreting model results.
 - Or combine 2 or more in a synthetic variable (e.g. water deficit \sim Temp + Precip).
 - Many methods available, e.g. sequential, ridge regression...
 - Measurement error can seriously complicate things (Biggs et al 2009; Freckleton 2011)
- For predictors with large effects, **consider interactions**.

Think about the shape of relationships

$$y \sim x + z$$

Really? Not everything has to be linear! Actually, it often is not.

Think about shape of relationship.



Removing predictors

Stepwise regression has many problems

- Whittingham et al. (2006) Why do we still use stepwise modelling in ecology and behaviour? *J. Animal Ecology*.

Stepwise regression has many problems

- Whittingham et al. (2006) Why do we still use stepwise modelling in ecology and behaviour? *J. Animal Ecology*.
- Mundry & Nunn (2009) Stepwise Model Fitting and Statistical Inference: Turning Noise into Signal Pollution. *Am Nat.*

Stepwise regression has many problems

- Whittingham et al. (2006) Why do we still use stepwise modelling in ecology and behaviour? *J. Animal Ecology*.
- Mundry & Nunn (2009) Stepwise Model Fitting and Statistical Inference: Turning Noise into Signal Pollution. *Am Nat.*
- This includes **stepAIC** (e.g. Dahlgren 2010; Burnham et al 2011; Hegyi & Garamszegi 2011).

Other common bad practices

- Testing bivariate relationships before building multivariable model

Heinze & Dunkler 2016

Other common bad practices

- Testing bivariate relationships before building multivariable model
- Removing non-significant predictors

Heinze & Dunkler 2016

Removing predictors?

- Always **keep 'core' predictors** (based on previous knowledge)

Heinze et al 2018

Removing predictors?

- Always **keep 'core' predictors** (based on previous knowledge)
- If ratio sample size/number of predictors is low (<10 EPP), avoid variable selection (too unstable)

Heinze et al 2018

Removing predictors?

- Always **keep 'core' predictors** (based on previous knowledge)
- If ratio sample size/number of predictors is low (<10 EPP), avoid variable selection (too unstable)
- If performing variable selection, always **assess stability** (bootstrap, etc)

[Heinze et al 2018](#)

Summary

1. Choose meaningful variables

Summary

1. Choose meaningful variables
 - Beware collinearity

Summary

1. Choose meaningful variables
 - Beware collinearity
 - Keep good n/p ratio

Summary

1. Choose meaningful variables
 - Beware collinearity
 - Keep good n/p ratio
2. Generate global model or (small) set of candidate models

Summary

1. Choose meaningful variables
 - Beware collinearity
 - Keep good n/p ratio
2. Generate global model or (small) set of candidate models
 - Avoid stepwise and all-subsets

Summary

1. Choose meaningful variables
 - Beware collinearity
 - Keep good n/p ratio
2. Generate global model or (small) set of candidate models
 - Avoid stepwise and all-subsets
 - Don't assume linear effects: think about appropriate functional relationships

Summary

1. Choose meaningful variables
 - Beware collinearity
 - Keep good n/p ratio
2. Generate global model or (small) set of candidate models
 - Avoid stepwise and all-subsets
 - Don't assume linear effects: think about appropriate functional relationships
 - Consider interactions for strong main effects

Summary

1. Choose meaningful variables
 - Beware collinearity
 - Keep good n/p ratio
2. Generate global model or (small) set of candidate models
 - Avoid stepwise and all-subsets
 - Don't assume linear effects: think about appropriate functional relationships
 - Consider interactions for strong main effects
3. If > 1 model have similar support, consider model averaging (or blending).

1. Choose meaningful variables
 - Beware collinearity
 - Keep good n/p ratio
2. Generate global model or (small) set of candidate models
 - Avoid stepwise and all-subsets
 - Don't assume linear effects: think about appropriate functional relationships
 - Consider interactions for strong main effects
3. If > 1 model have similar support, consider model averaging (or blending).
4. Always check fitted models thoroughly

1. Choose meaningful variables
 - Beware collinearity
 - Keep good n/p ratio
2. Generate global model or (small) set of candidate models
 - Avoid stepwise and all-subsets
 - Don't assume linear effects: think about appropriate functional relationships
 - Consider interactions for strong main effects
3. If > 1 model have similar support, consider model averaging (or blending).
4. Always check fitted models thoroughly
5. Always report effect sizes

Model comparison

Francisco Rodríguez-Sánchez

<https://frodriguezsanchez.net>

Trees dataset

```
trees <- read.csv("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 |

Four models

```
m1 <- lm(height ~ dbh, data = trees)
```

```
m2 <- lm(height ~ sex, data = trees)
```

```
m3 <- lm(height ~ site, data = trees)
```

```
m4 <- lm(height ~ site*dbh, data = trees)
```

Compare model performance

```
library("performance")
compare_performance(m1, m2, m3, m4)
```

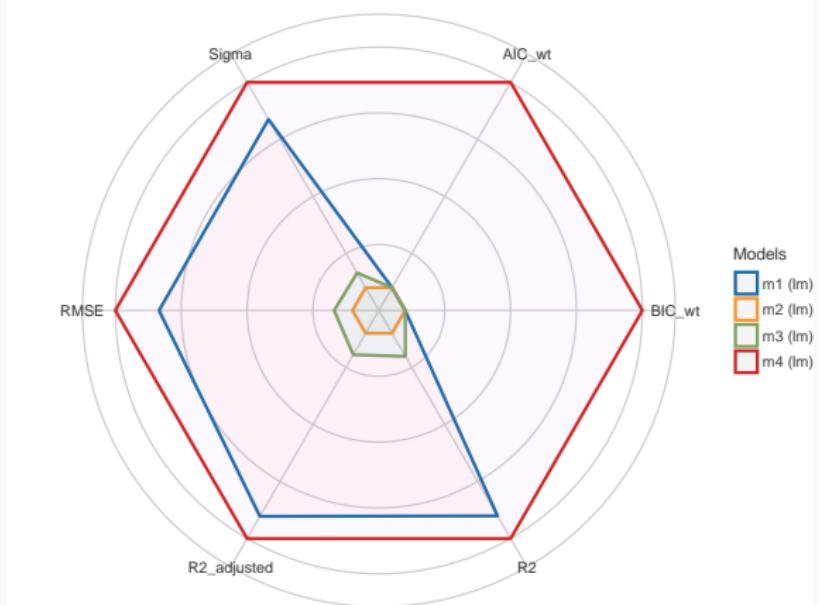
```
# Comparison of Model Performance Indices
```

| Name | Model | AIC | AIC weights | BIC | BIC weights | R2 | R2 (adj.) |
|-------|-------|----------|-------------|----------|-------------|-------|-----------|
| <hr/> | | | | | | | |
| m1 | lm | 5660.250 | 8.39e-126 | 5674.973 | 1.28e-106 | 0.787 | 0.787 |
| m2 | lm | 7206.145 | 0.00e+00 | 7220.868 | 0.00e+00 | 0.002 | 0.001 |
| m3 | lm | 7117.264 | 0.00e+00 | 7171.250 | 0.00e+00 | 0.102 | 0.093 |
| m4 | lm | 5084.253 | 1.00 | 5187.316 | 1.00 | 0.885 | 0.882 |

Compare model performance

```
library("see")
plot(compare_performance(m1, m2, m3, m4))
```

Comparison of Model Indices



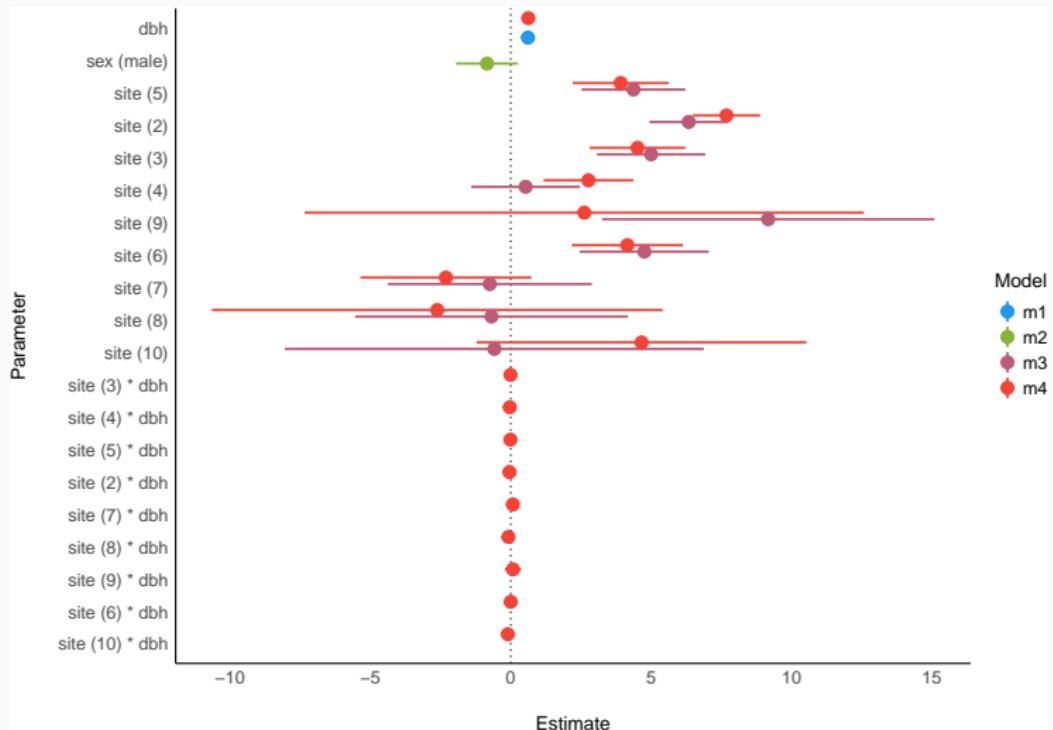
Compare parameters

```
library("parameters")
compare_parameters(m1, m2, m3, m4)
```

| Parameter | m1 | m2 | m3 | m4 |
|-----------------|----------------------|----------------------|----------------------|-------------------------|
| <hr/> | | | | |
| (Intercept) | 19.34 (18.73, 19.95) | 36.93 (36.15, 37.71) | 33.84 (33.00, 34.68) | 16.36 (15.65, 17.07) |
| dbh | 0.62 (0.60, 0.64) | | | 0.63 (0.61, 0.65) |
| sex (male) | | -0.84 (-1.94, 0.26) | | |
| site (5) | | | 4.37 (2.52, 6.22) | 3.92 (2.21, 5.63) |
| site (2) | | | 6.34 (4.94, 7.74) | 7.68 (6.49, 8.88) |
| site (3) | | | 5.00 (3.87, 6.93) | 4.52 (2.82, 6.22) |
| site (4) | | | 0.53 (-1.40, 2.47) | 2.77 (1.17, 4.37) |
| site (9) | | | 9.17 (3.25, 15.09) | 2.62 (-7.34, 12.58) |
| site (6) | | | 4.76 (2.46, 7.06) | 4.16 (2.17, 6.14) |
| site (7) | | | -0.74 (-4.37, 2.89) | -2.31 (-5.35, 0.74) |
| site (8) | | | -0.68 (-5.54, 4.17) | -2.62 (-10.64, 5.41) |
| site (10) | | | -0.58 (-8.04, 6.88) | 4.66 (-1.21, 10.53) |
| site (3) * dbh | | | | -6.03e-03 (-0.06, 0.05) |
| site (4) * dbh | | | | -0.03 (-0.09, 0.02) |
| site (5) * dbh | | | | -0.01 (-0.06, 0.04) |
| site (2) * dbh | | | | -0.04 (-0.08, 0.00) |
| site (7) * dbh | | | | 0.08 (-0.02, 0.18) |
| site (8) * dbh | | | | -0.08 (-0.30, 0.14) |
| site (9) * dbh | | | | 0.08 (-0.21, 0.37) |
| site (6) * dbh | | | | 1.34e-03 (-0.06, 0.06) |
| site (10) * dbh | | | | -0.10 (-0.33, 0.12) |
| <hr/> | | | | |
| Observations | 1000 | 1000 | 1000 | 1000 |

Compare parameters

```
library("parameters")
plot(compare_parameters(m1, m2, m3, m4))
```



Generalised Linear Models

Logistic regression

Francisco Rodríguez-Sánchez

<https://frodriguezsanchez.net>

Q: Survival of passengers on the Titanic ~ Class

Read `titanic_long.csv` dataset and fit linear model (survival ~ class).

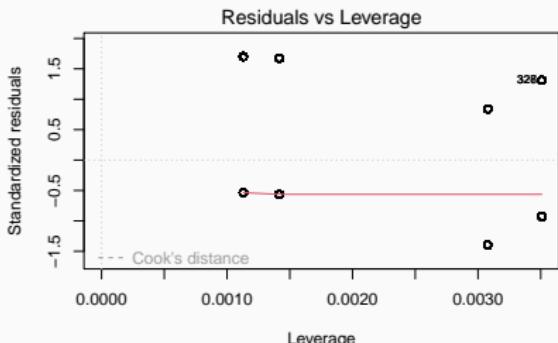
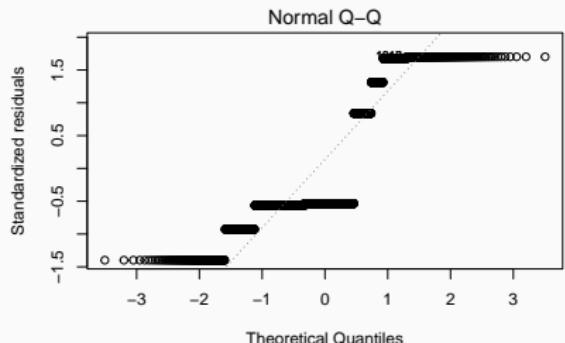
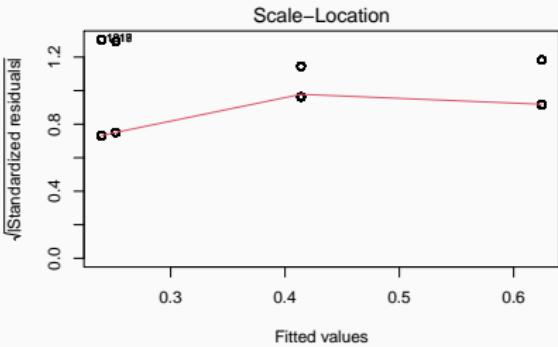
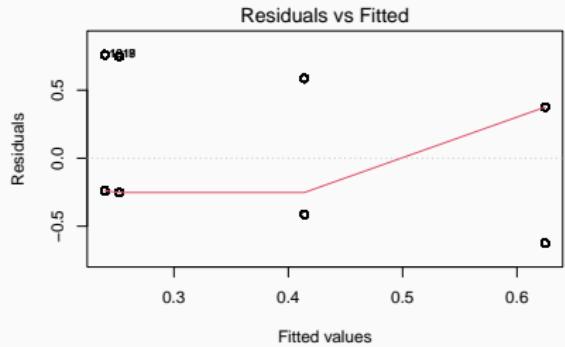
```
class    age   sex survived
1 first  adult male      1
2 first  adult male      1
3 first  adult male      1
4 first  adult male      1
5 first  adult male      1
6 first  adult male      1
```

Quiz: Did passenger class influence survival?

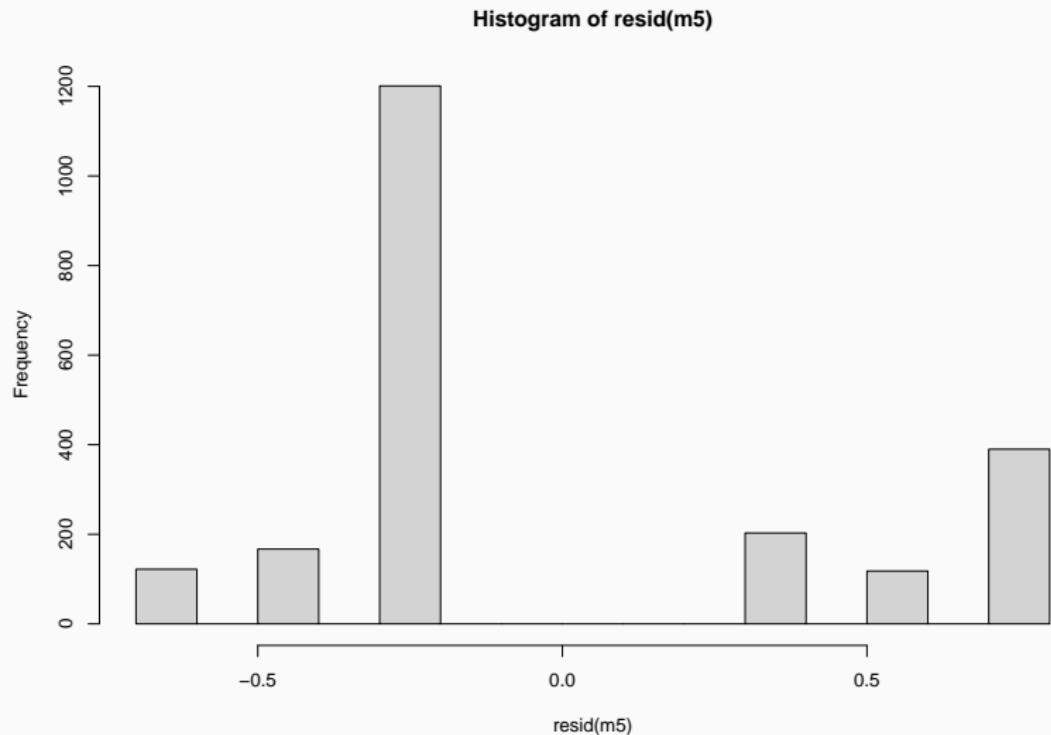
<https://pollev.com/franciscorod726>

Let's check linear model:

```
m5 <- lm(survived ~ class, data = titanic)
```



Weird residuals!



What if your residuals are clearly non-normal
or variance not constant (heteroscedasticity)?

Binary variables (0/1)

Counts (0, 1, 2, 3, ...)

Categories (“small”, “medium”, “large”...)

Generalised Linear Models to the rescue!

Generalised Linear Models

1. Response variable - distribution family

Generalised Linear Models

1. Response variable - distribution family
 - Bernouilli - Binomial

Generalised Linear Models

1. Response variable - distribution family

- Bernouilli - Binomial
- Poisson

Generalised Linear Models

1. Response variable - distribution family

- Bernouilli - Binomial
- Poisson
- Gamma

Generalised Linear Models

1. Response variable - distribution family

- Bernouilli - Binomial
- Poisson
- Gamma
- etc

Generalised Linear Models

1. Response variable - distribution family

- Bernouilli - Binomial
- Poisson
- Gamma
- etc

2. Predictors (continuous or categorical)

Generalised Linear Models

1. Response variable - distribution family

- Bernouilli - Binomial
- Poisson
- Gamma
- etc

2. Predictors (continuous or categorical)

3. Link function

Generalised Linear Models

1. Response variable - distribution family

- Bernouilli - Binomial
- Poisson
- Gamma
- etc

2. Predictors (continuous or categorical)

3. Link function

- Gaussian: identity

Generalised Linear Models

1. Response variable - distribution family

- Bernouilli - Binomial
- Poisson
- Gamma
- etc

2. Predictors (continuous or categorical)

3. Link function

- Gaussian: identity
- Binomial: logit, probit

Generalised Linear Models

1. Response variable - distribution family

- Bernouilli - Binomial
- Poisson
- Gamma
- etc

2. Predictors (continuous or categorical)

3. Link function

- Gaussian: identity
- Binomial: logit, probit
- Poisson: log...

Generalised Linear Models

1. Response variable - distribution family

- Bernouilli - Binomial
- Poisson
- Gamma
- etc

2. Predictors (continuous or categorical)

3. Link function

- Gaussian: identity
- Binomial: logit, probit
- Poisson: log...
- See [family](#).

The modelling process

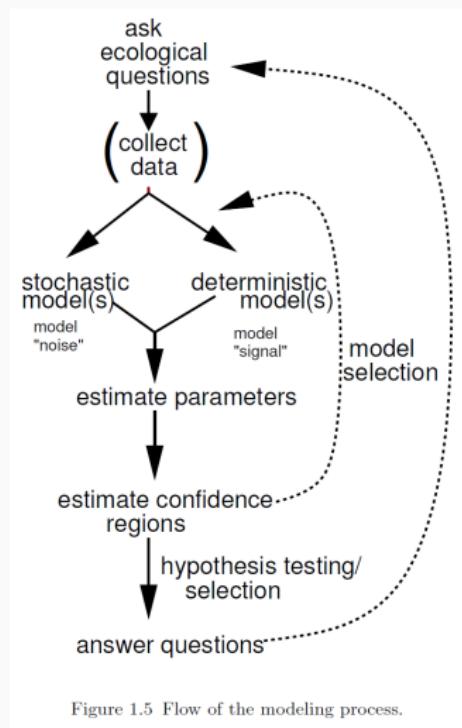


Figure 1.5 Flow of the modeling process.

Bernoulli - Binomial distribution (Logistic regression)

Response variable: **Yes/No** (e.g. survival, sex, presence/absence)

Canonical link function: **logit** (*log odds*), but others possible (see **family**)

$$\text{logit}(p) = \log \left(\frac{p}{1-p} \right)$$

Then

$$\begin{aligned}\text{logit}(P(\text{alive})) &= a + bx \\ P(\text{alive}) &= \text{invlogit}(a + bx) = \frac{e^{a+bx}}{1 + e^{a+bx}}\end{aligned}$$

Where is the variance?

In a Gaussian GLM

$$y \sim \text{Normal}(\mu, \sigma)$$

In a Binomial GLM

$$y \sim \text{Binomial}(n, p)$$

n = number of trials

p = probability of success

$$\text{Var}(y) = np(1 - p)$$

(maximum variance when p around 0.5)

Back to survival of Titanic
passengers

How many survived in each class?

```
table(titanic$class, titanic$survived)
```

| | 0 | 1 |
|--------|-----|-----|
| crew | 673 | 212 |
| first | 122 | 203 |
| second | 167 | 118 |
| third | 528 | 178 |

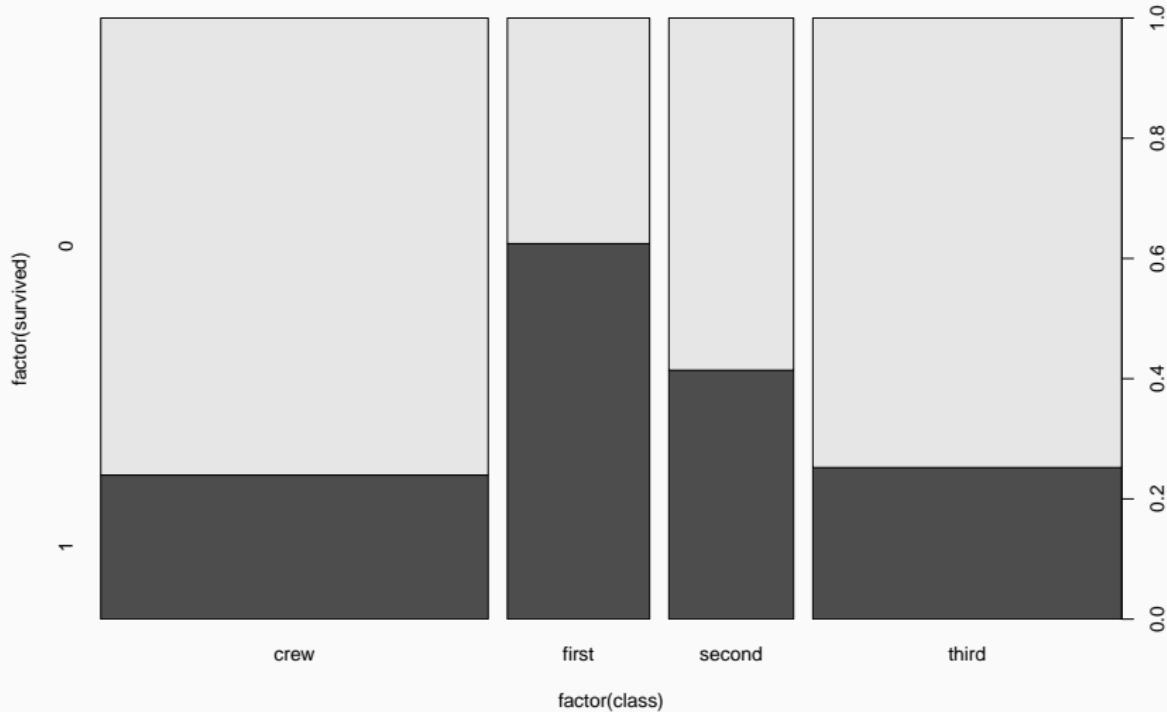
How many survived in each class? (dplyr)

```
titanic %>%  
  group_by(class, survived) %>%  
  summarise(count = n())
```

```
# A tibble: 8 x 3  
# Groups:   class [4]  
  class  survived  count  
  <chr>    <int> <int>  
1 crew        0    673  
2 crew        1    212  
3 first       0    122  
4 first       1    203  
5 second      0    167  
6 second      1    118  
7 third       0    528  
8 third       1    178
```

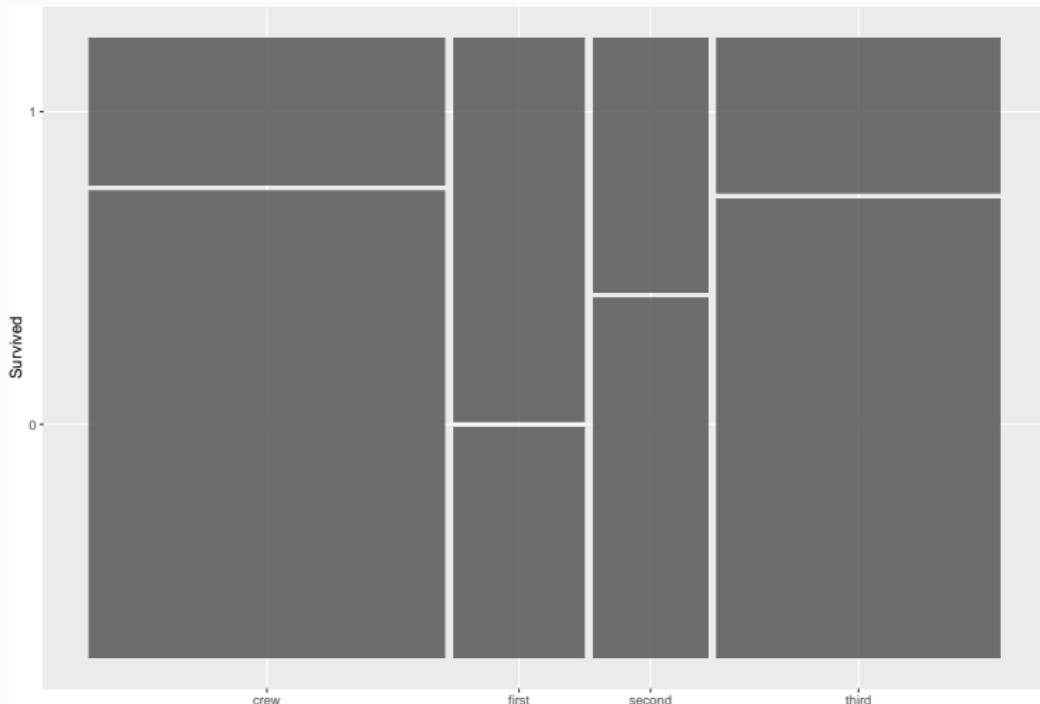
Data visualisation (mosaic plot)

```
plot(factor(survived) ~ factor(class), data = titanic)
```



Mosaic plots (ggplot2)

```
ggplot(titanic) +  
  geom_mosaic(aes(x = product(survived, class))) +  
  labs(x = "", y = "Survived")
```



Fitting GLMs in R: `glm`

```
tit.glm <- glm(survived ~ class,  
                 data = titanic,  
                 family = binomial)
```

which corresponds to

$$\text{logit}(P(\text{survival})_i) = a + b \cdot \text{class}_i$$

$$\text{logit}(P(\text{survival})_i) = a + b_{\text{first}} + c_{\text{second}} + d_{\text{third}}$$

Interpreting binomial GLM

```
tit.glm <- glm(survived ~ class, data = titanic, family = binomial)
```

Call:

```
glm(formula = survived ~ class, family = binomial, data = titanic)
```

Deviance Residuals:

| Min | 1Q | Median | 3Q | Max |
|---------|---------|---------|--------|--------|
| -1.3999 | -0.7623 | -0.7401 | 0.9702 | 1.6906 |

Coefficients:

| | Estimate | Std. Error | z value | Pr(> z) |
|--------------|----------|------------|---------|--------------|
| (Intercept) | -1.15516 | 0.07876 | -14.667 | < 2e-16 *** |
| classfirst | 1.66434 | 0.13902 | 11.972 | < 2e-16 *** |
| classesecond | 0.80785 | 0.14375 | 5.620 | 1.91e-08 *** |
| classthird | 0.06785 | 0.11711 | 0.579 | 0.562 |

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

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 2769.5 on 2200 degrees of freedom

Residual deviance: 2588.6 on 2197 degrees of freedom

AIC: 2596.6

Number of Fisher Scoring iterations: 4

Binomial GLM estimates are in **logit** scale!

We need to **back-transform** (apply *inverse logit*):

- Manually: `plogis`

Binomial GLM estimates are in **logit** scale!

We need to **back-transform** (apply *inverse logit*):

- Manually: `plogis`
- Automatically: `effects`, `modelbased`, etc.

Interpreting logistic regression output (effects pkg)

```
library("effects")
allEffects(tit.glm)
```

model: survived ~ class

```
class effect
class
    crew      first     second     third
0.2395480 0.6246154 0.4140351 0.2521246
```

Interpreting logistic regression output (effects pkg)

Including confidence intervals:

```
summary(allEffects(tit.glm))
```

```
model: survived ~ class

class effect
class
  crew      first     second     third
0.2395480 0.6246154 0.4140351 0.2521246

Lower 95 Percent Confidence Limits
class
  crew      first     second     third
0.2125668 0.5706887 0.3582390 0.2214588

Upper 95 Percent Confidence Limits
class
  crew      first     second     third
0.2687850 0.6756185 0.4721282 0.2854798
```

Interpreting logistic regression output (modelbased)

```
library("modelbased")
estimate_means(tit.glm)
```

Estimated Marginal Means

| class | Probability | SE | 95% CI |
|--------|-------------|------|--------------|
| <hr/> | | | |
| first | 0.62 | 0.03 | [0.57, 0.68] |
| second | 0.41 | 0.03 | [0.36, 0.47] |
| third | 0.25 | 0.02 | [0.22, 0.29] |
| crew | 0.24 | 0.01 | [0.21, 0.27] |

Marginal means estimated at class

Analysing differences among factor levels (class)

```
library("modelbased")
estimate_contrasts(tit.glm)
```

Marginal Contrasts Analysis

| Level1 | Level2 | Difference | 95% CI | SE | df | z |
|--------|--------|------------|---------------|------|-----|-------|
| <hr/> | | | | | | |
| first | crew | 1.66 | [1.30, 2.03] | 0.14 | Inf | 11.97 |
| first | second | 0.86 | [0.42, 1.29] | 0.17 | Inf | 5.16 |
| first | third | 1.60 | [1.22, 1.98] | 0.14 | Inf | 11.11 |
| second | crew | 0.81 | [0.43, 1.19] | 0.14 | Inf | 5.62 |
| second | third | 0.74 | [0.35, 1.13] | 0.15 | Inf | 4.99 |
| third | crew | 0.07 | [-0.24, 0.38] | 0.12 | Inf | 0.58 |

Marginal contrasts estimated at class
p-value adjustment method: Holm (1979)

Pseudo R-squared for GLMs

```
library("performance")
r2(tit.glm)
```

```
# R2 for Logistic Regression
Tjur's R2: 0.087
```

But there are caveats (e.g. see [here](#) and [here](#))

Presenting model results

```
kable(xtable::xtable(tit.glm), digits = 2)
```

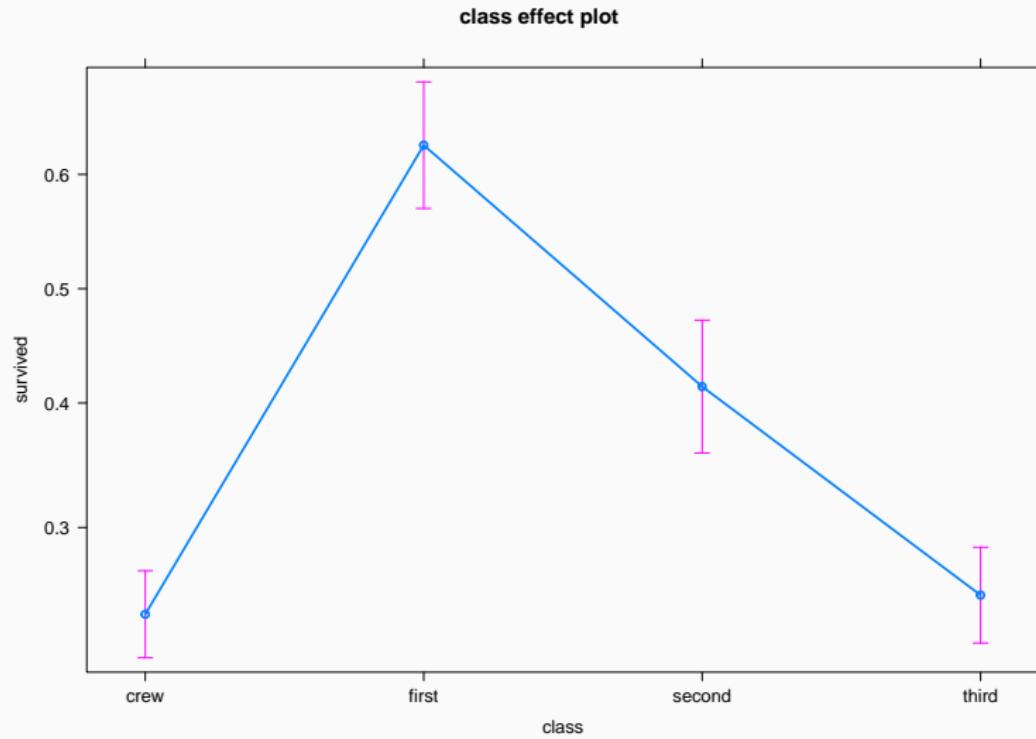
| | Estimate | Std. Error | z value | Pr(> z) |
|-------------|----------|------------|---------|----------|
| (Intercept) | -1.16 | 0.08 | -14.67 | 0.00 |
| classfirst | 1.66 | 0.14 | 11.97 | 0.00 |
| classsecond | 0.81 | 0.14 | 5.62 | 0.00 |
| classthird | 0.07 | 0.12 | 0.58 | 0.56 |

Presenting model results

```
library("modelsummary")
modelsummary(tit.glm)
```

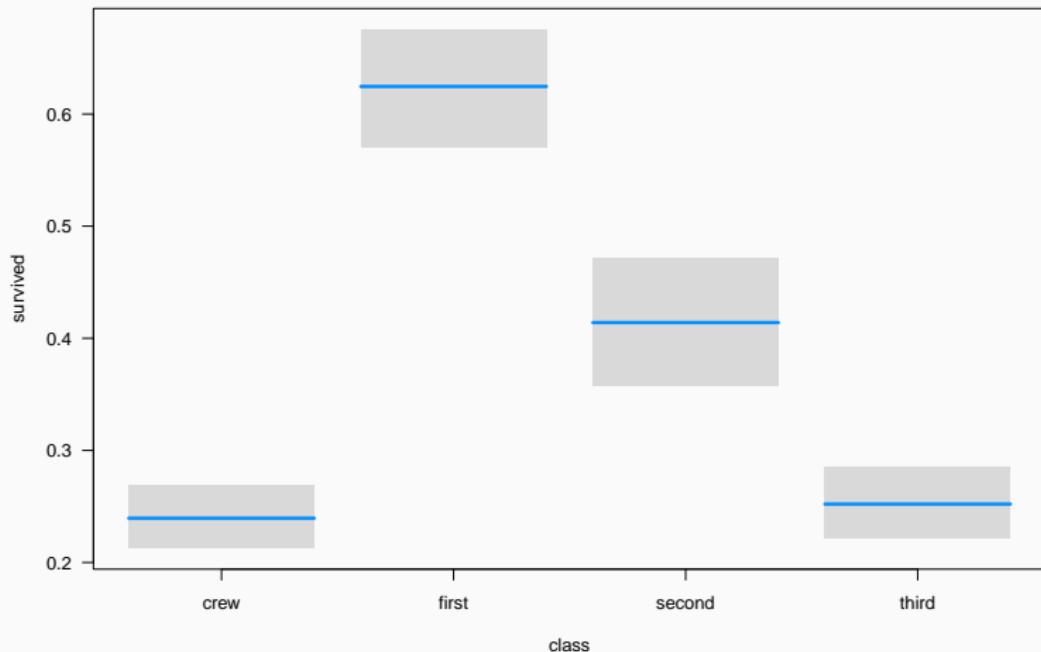
Visualising model: effects package

```
plot(allEffects(tit.glm))
```



Visualising model: visreg package

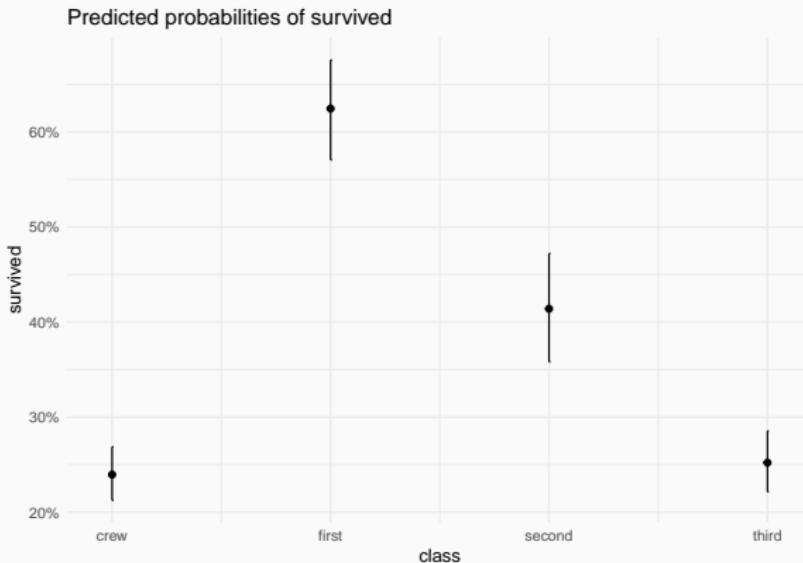
```
visreg(tit.glm, scale = "response", rug = FALSE)
```



Visualising model: sjPlot package

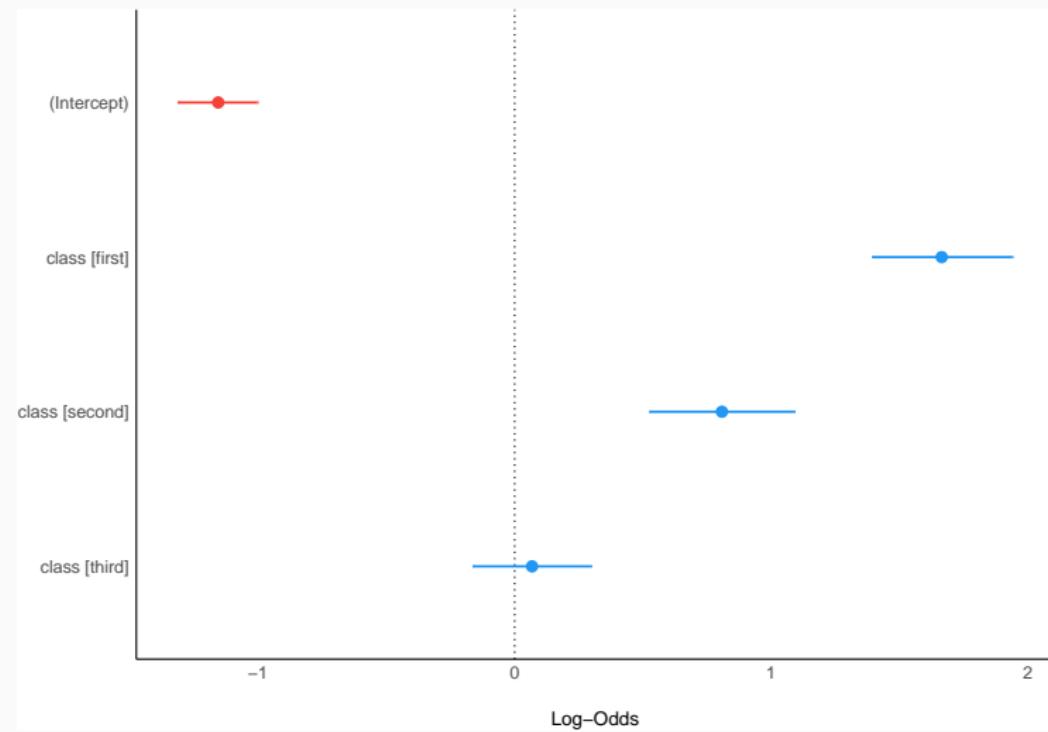
```
sjPlot::plot_model(tit.glm, type = "eff")
```

\$class



Visualising model: see package

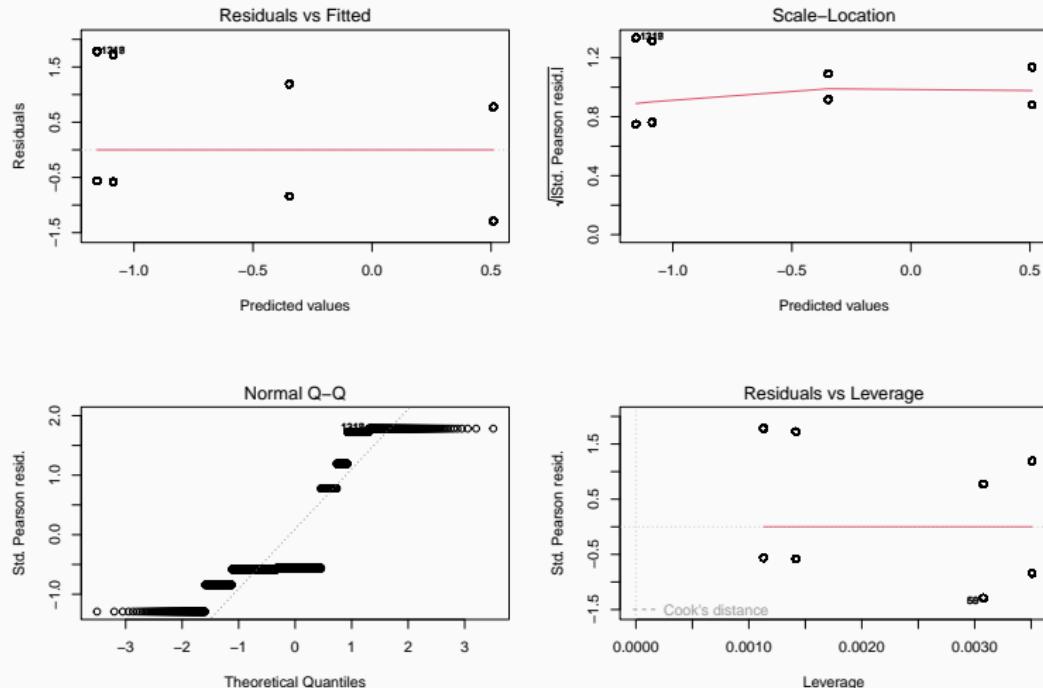
```
plot(parameters(tit.glm), show_intercept = TRUE)
```



Model checking

plot(model) not very useful with binomial GLM

```
plot(tit.glm)
```



```
null device
```

```
1
```

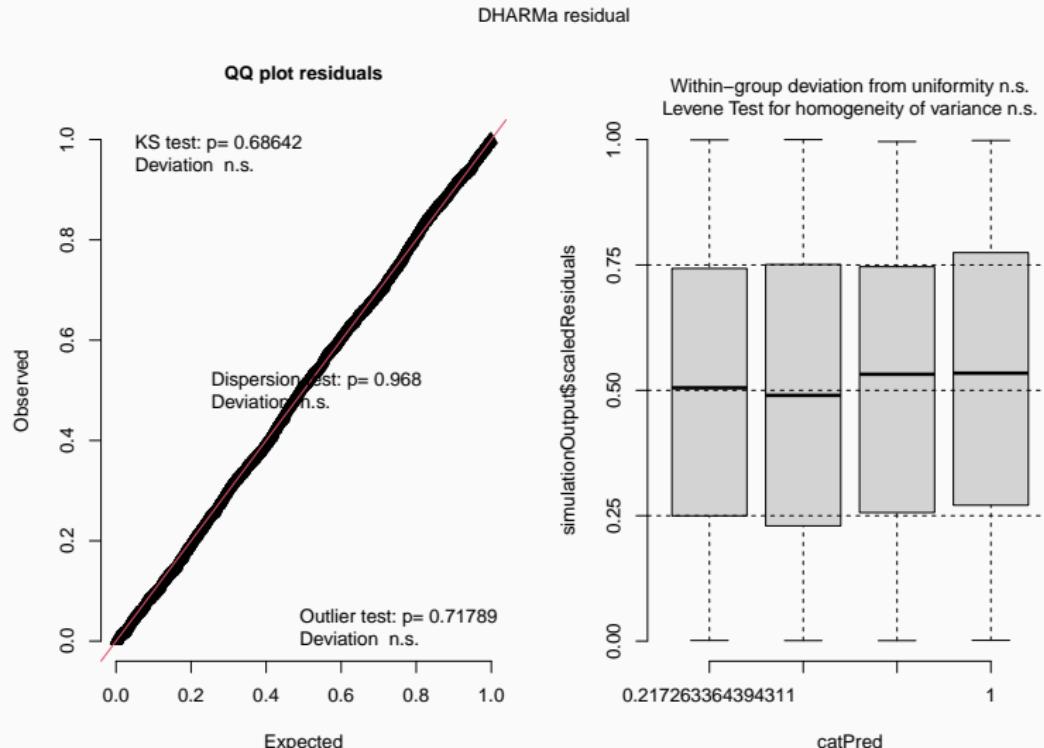
Binned residual plots for logistic regression

```
binned_residuals(tit.glm)
```

Ok: About 100% of the residuals are inside the error bounds.

Residual diagnostics with DHARMA

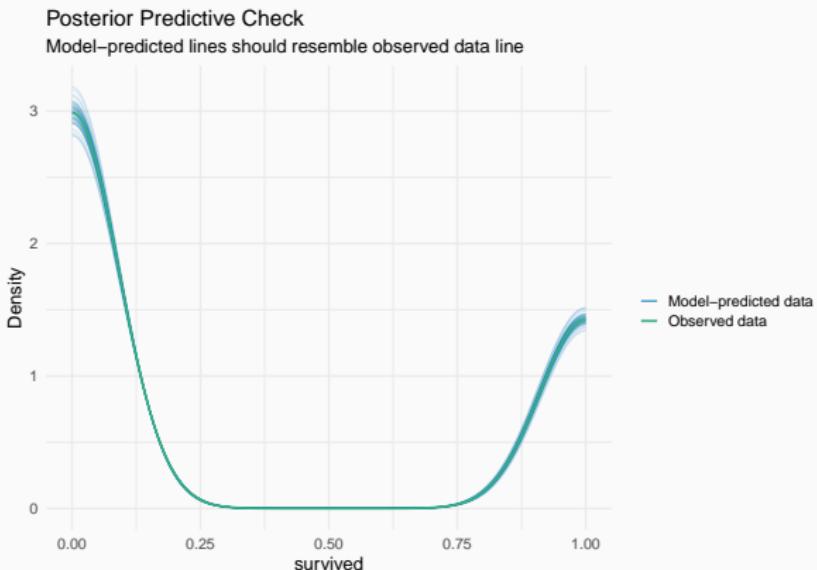
```
library("DHARMA")
simulateResiduals(tit.glm, plot = TRUE)
```



Posterior predictive checking

Simulate data from fitted model (y_{rep}) and compare with observed data (y)

```
check_predictions(tit.glm)
```

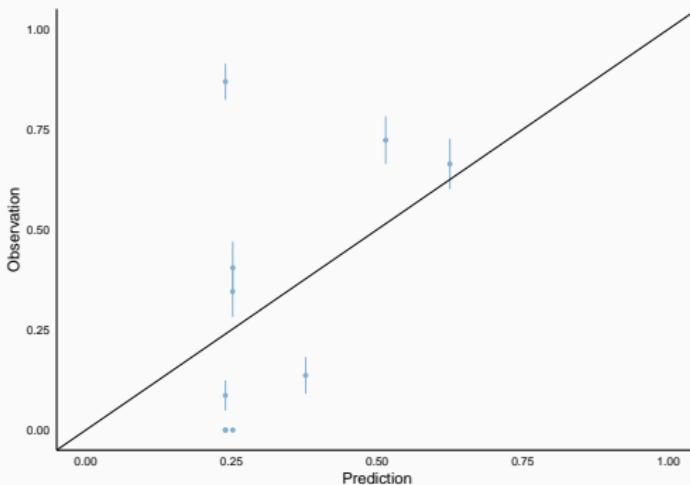


Calibration plot

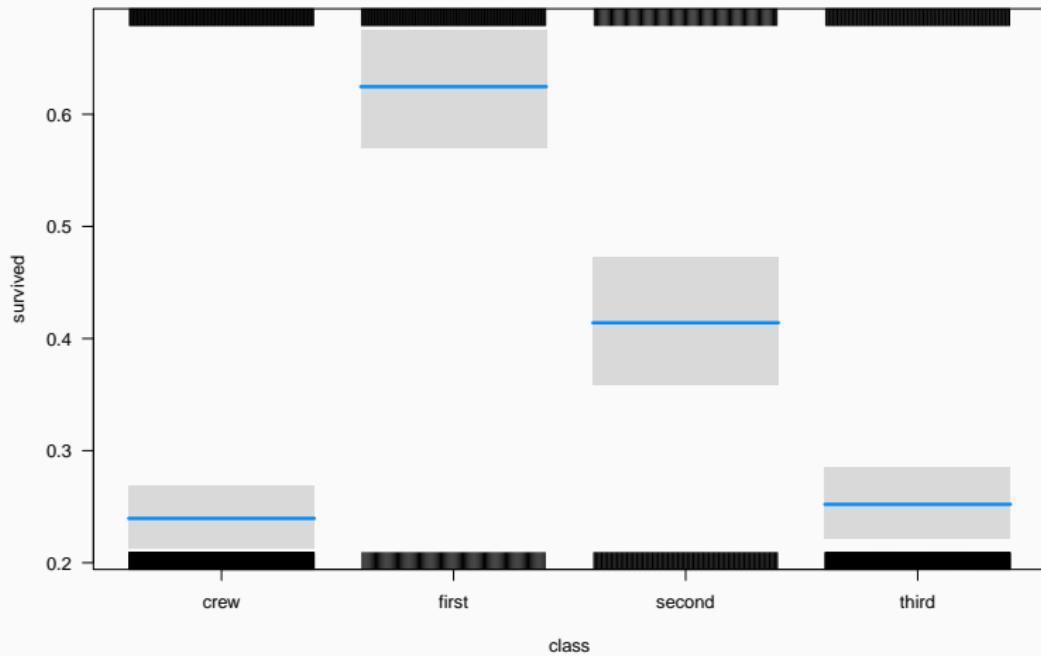
Compares predicted vs observed probabilities (grouped by quantiles)

```
library("predtools")
titanic$surv.pred <- predict(tit.glm, type = "response")
calibration_plot(data = titanic, obs = "survived", pred = "surv.pred",
                  x_lim = c(0,1), y_lim = c(0,1))
```

```
$calibration_plot
```



Passenger class was important, but lots of unexplained variation



The goal is not to test whether the model's assumptions are "true", because all models are false.

Rather, the goal is to assess exactly **how the model fails to describe the data**, as a path towards **model comprehension, revision, and improvement**.

Richard McElreath. *Statistical Rethinking*

Recapitulating

1. Visualise data

Recapitulating

1. Visualise data
2. Fit model: `glm`. Don't forget to specify `family`!

Recapitulating

1. Visualise data
2. Fit model: `glm`. Don't forget to specify `family`!
3. Examine model: `summary`

Recapitulating

1. Visualise data
2. Fit model: `glm`. Don't forget to specify `family`!
3. Examine model: `summary`
4. Back-transform parameters from *logit* into probability scale
(e.g. `allEffects`)

Recapitulating

1. Visualise data
2. Fit model: `glm`. Don't forget to specify `family`!
3. Examine model: `summary`
4. Back-transform parameters from *logit* into probability scale
(e.g. `allEffects`)
5. Plot model: `plot(allEffects(model))`, `visreg`, `plot_model`,
calibration plot...

Recapitulating

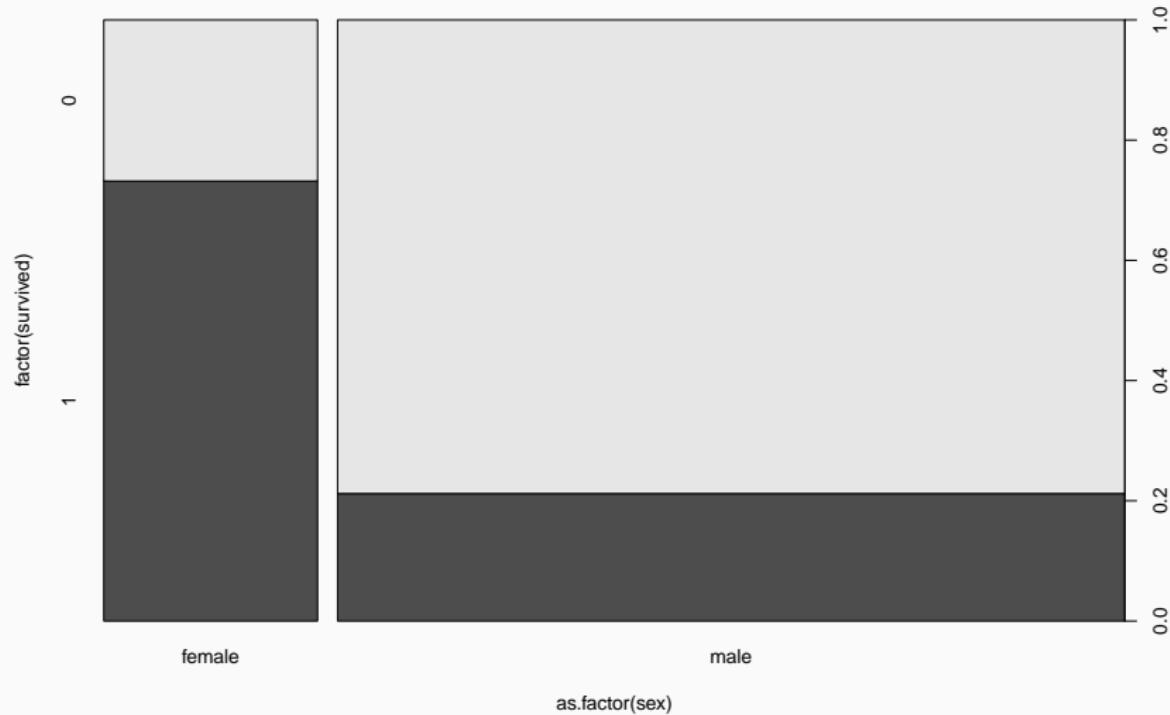
1. Visualise data
2. Fit model: `glm`. Don't forget to specify `family`!
3. Examine model: `summary`
4. Back-transform parameters from *logit* into probability scale
(e.g. `allEffects`)
5. Plot model: `plot(allEffects(model))`, `visreg`, `plot_model`,
calibration plot...
6. Examine residuals: `DHARMa::simulateResiduals`.

Q: Did men have higher survival
than women?

Quiz

<https://pollev.com/franciscorod726>

First, visualise data



Fit model

Call:

```
glm(formula = survived ~ sex, family = binomial, data = titanic)
```

Deviance Residuals:

| Min | 1Q | Median | 3Q | Max |
|---------|---------|---------|--------|--------|
| -1.6226 | -0.6903 | -0.6903 | 0.7901 | 1.7613 |

Coefficients:

| | Estimate | Std. Error | z value | Pr(> z) | | |
|----------------|----------|------------|----------|------------|---------|---|
| (Intercept) | 1.0044 | 0.1041 | 9.645 | <2e-16 *** | | |
| sexmale | -2.3172 | 0.1196 | -19.376 | <2e-16 *** | | |
| --- | | | | | | |
| Signif. codes: | 0 '***' | 0.001 '**' | 0.01 '*' | 0.05 '.' | 0.1 ' ' | 1 |

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 2769.5 on 2200 degrees of freedom
Residual deviance: 2335.0 on 2199 degrees of freedom
AIC: 2339

Number of Fisher Scoring iterations: 4

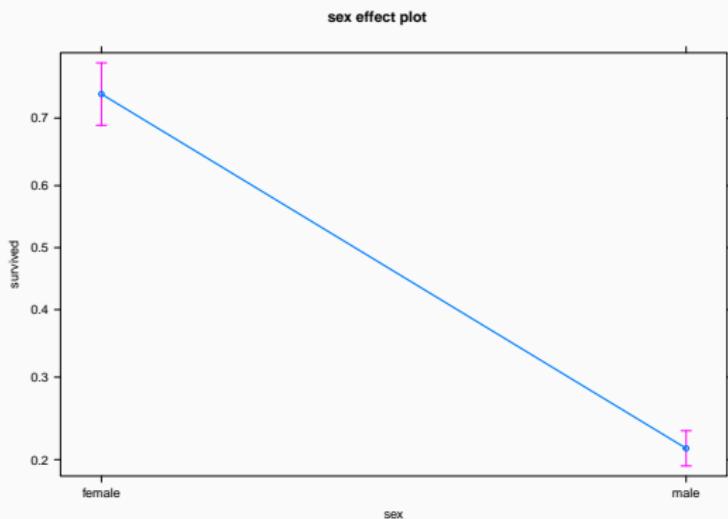
Model interpretation

```
model: survived ~ sex
```

```
sex effect
```

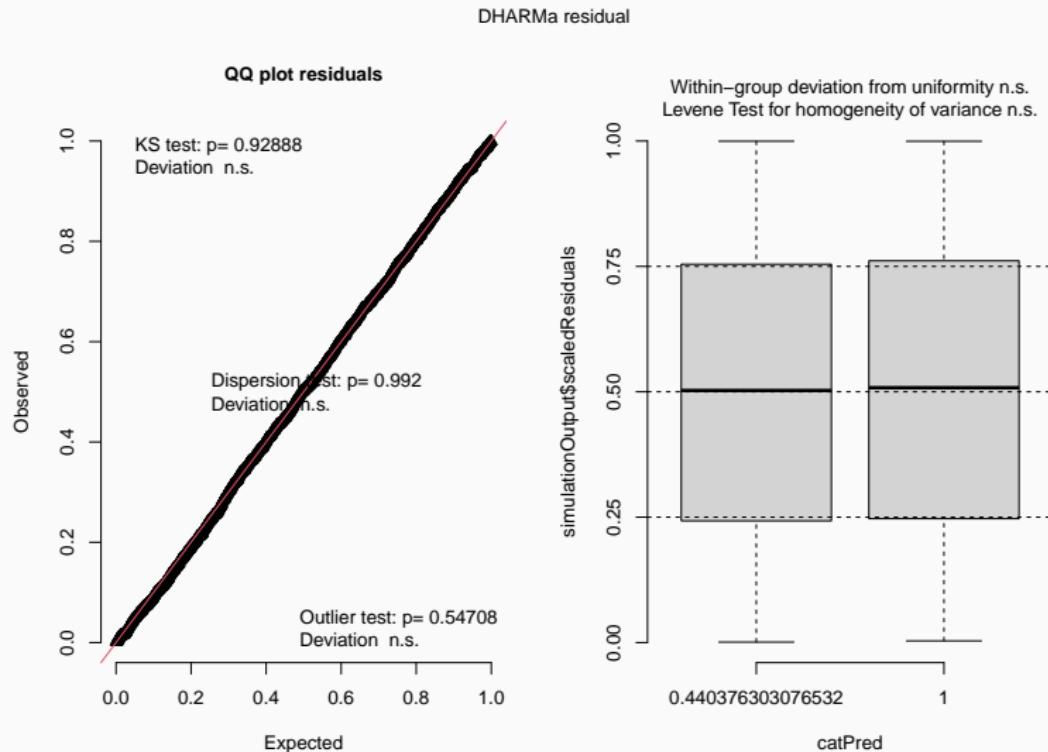
```
sex
```

| female | male |
|-----------|-----------|
| 0.7319149 | 0.2120162 |



Model checking

```
simulateResiduals(tit.sex, plot = TRUE)
```



Q: Did women have higher survival because they travelled more in first class?

Did women have higher survival because they travelled more in first class?



Let's look at the data

```
table(titanic$class, titanic$survived, titanic$sex)
```

```
, , = female
```

| | 0 | 1 |
|--------|-----|-----|
| crew | 3 | 20 |
| first | 4 | 141 |
| second | 13 | 93 |
| third | 106 | 90 |

```
, , = male
```

| | 0 | 1 |
|--------|-----|-----|
| crew | 670 | 192 |
| first | 118 | 62 |
| second | 154 | 25 |
| third | 422 | 88 |

Quiz

<https://pollev.com/franciscorod726>

Fit additive model with both factors

Call:

```
glm(formula = survived ~ class + sex, family = binomial, data = titanic)
```

Deviance Residuals:

| Min | 1Q | Median | 3Q | Max |
|---------|---------|---------|--------|--------|
| -2.0915 | -0.7149 | -0.5012 | 0.7297 | 2.0673 |

Coefficients:

| | Estimate | Std. Error | z value | Pr(> z) |
|-------------|----------|------------|---------|--------------|
| (Intercept) | 1.18740 | 0.15747 | 7.541 | 4.68e-14 *** |
| classfirst | 0.88081 | 0.15697 | 5.611 | 2.01e-08 *** |
| classsecond | -0.07178 | 0.17093 | -0.420 | 0.675 |
| classthird | -0.77742 | 0.14231 | -5.463 | 4.69e-08 *** |
| sexmale | -2.42133 | 0.13909 | -17.408 | < 2e-16 *** |

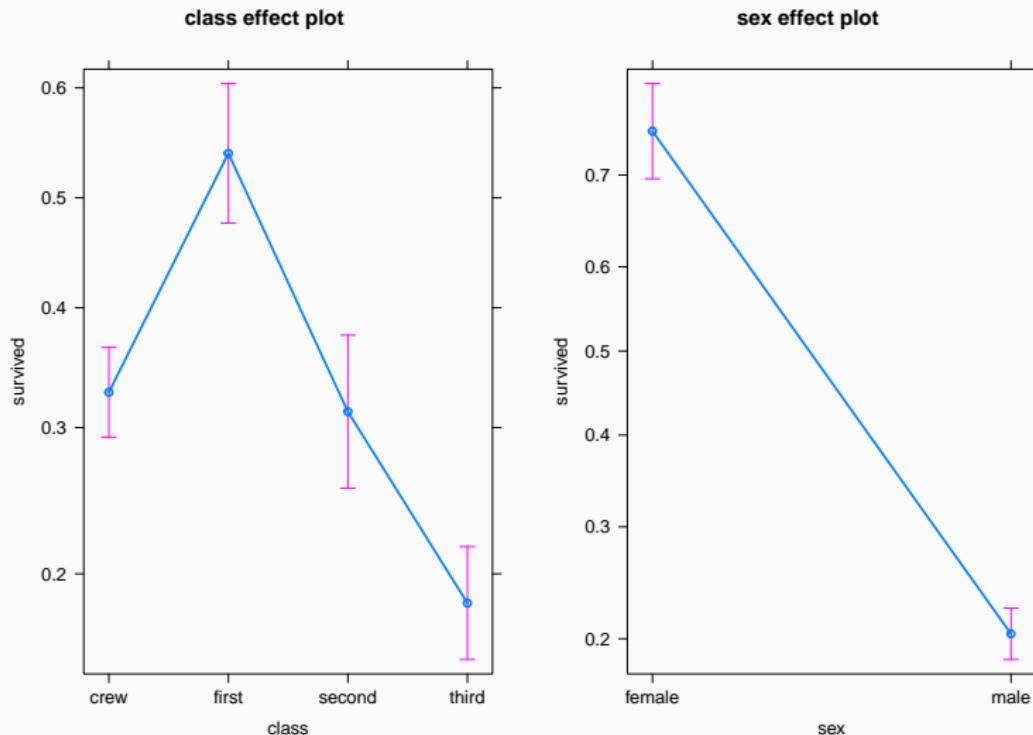
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 2769.5 on 2200 degrees of freedom

Residual deviance: 2228.9 on 2196 degrees of freedom

Plot additive model



Fit model with the interaction of both factors

Call:

```
glm(formula = survived ~ class * sex, family = binomial, data = titanic)
```

Deviance Residuals:

| Min | 1Q | Median | 3Q | Max |
|---------|---------|---------|--------|--------|
| -2.6797 | -0.7099 | -0.6155 | 0.5115 | 1.9842 |

Coefficients:

| | Estimate | Std. Error | z value | Pr(> z) |
|---------------------|----------|------------|---------|--------------|
| (Intercept) | 1.89712 | 0.61914 | 3.064 | 0.00218 ** |
| classfirst | 1.66535 | 0.80026 | 2.081 | 0.03743 * |
| classsecond | 0.07053 | 0.68630 | 0.103 | 0.91815 |
| classthird | -2.06075 | 0.63551 | -3.243 | 0.00118 ** |
| sexmale | -3.14690 | 0.62453 | -5.039 | 4.68e-07 *** |
| classfirst:sexmale | -1.05911 | 0.81959 | -1.292 | 0.19627 |
| classsecond:sexmale | -0.63882 | 0.72402 | -0.882 | 0.37760 |
| classthird:sexmale | 1.74286 | 0.65139 | 2.676 | 0.00746 ** |
| --- | | | | |

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

(Dispersion parameter for binomial family taken to be 1)

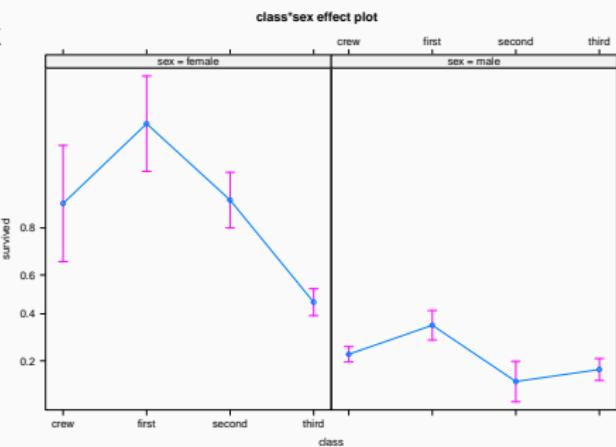
Women had higher survival than men, even within the same class

model: survived ~ class * sex

class*sex effect

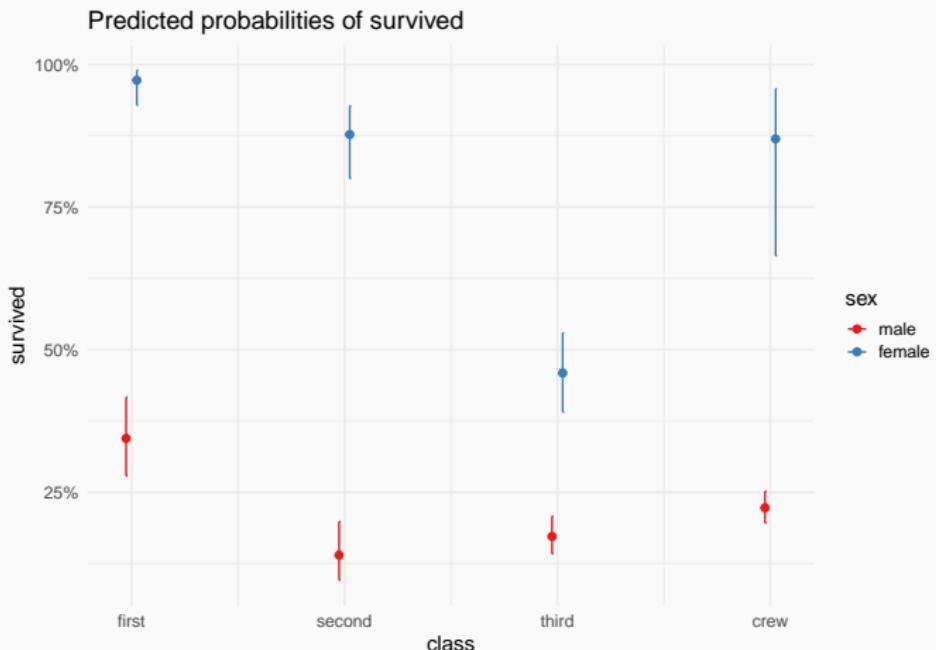
sex

| class | female | male |
|--------|-----------|-----------|
| crew | 0.8695652 | 0.2227378 |
| first | 0.9724138 | 0.3444444 |
| second | 0.8773585 | 0.1396648 |
| third | 0.4591837 | 0.1725490 |



Visualising model (sjPlot)

```
plot_model(tit.sex.class.int, type = "int")
```



Comparing models

```
library("performance")
compare_performance(tit.sex.class.add, tit.sex.class.int)
```

Comparison of Model Performance Indices

| Name | Model | AIC | AIC weights | BIC | BIC weights | Tjur' |
|-------------------|-------|----------|-------------|----------|-------------|-------|
| <hr/> | | | | | | |
| tit.sex.class.add | glm | 2238.913 | 1.41e-13 | 2267.396 | 7.25e-10 | 0 |
| tit.sex.class.int | glm | 2179.733 | 1.000 | 2225.306 | 1.000 | 0 |

Comparing parameters

```
compare_parameters(tit.sex.class.add, tit.sex.class.int)
```

| Parameter | tit.sex.class.add | tit.sex.class.int |
|-----------------------------|----------------------|----------------------|
| (Intercept) | 1.19 (0.88, 1.50) | 1.90 (0.68, 3.11) |
| class (first) | 0.88 (0.57, 1.19) | 1.67 (0.10, 3.23) |
| class (second) | -0.07 (-0.41, 0.26) | 0.07 (-1.27, 1.42) |
| class (third) | -0.78 (-1.06, -0.50) | -2.06 (-3.31, -0.82) |
| sex (male) | -2.42 (-2.69, -2.15) | -3.15 (-4.37, -1.92) |
| class (first) * sex (male) | | -1.06 (-2.67, 0.55) |
| class (second) * sex (male) | | -0.64 (-2.06, 0.78) |
| class (third) * sex (male) | | 1.74 (0.47, 3.02) |
| Observations | 2201 | 2201 |

Extra exercises:

Is survival related to age?

Are age effects dependent on sex?

Logistic regression for proportion data

Read Titanic data in different format

Read `titanic_prop.csv` data.

| | X | Class | Sex | Age | No | Yes |
|---|---|-------|--------|-------|-----|-----|
| 1 | 1 | 1st | Female | Adult | 4 | 140 |
| 2 | 2 | 1st | Female | Child | 0 | 1 |
| 3 | 3 | 1st | Male | Adult | 118 | 57 |
| 4 | 4 | 1st | Male | Child | 0 | 5 |
| 5 | 5 | 2nd | Female | Adult | 13 | 80 |
| 6 | 6 | 2nd | Female | Child | 0 | 13 |

These are the same data, but summarized (see `Freq` variable).

Use `cbind(n.success, n.failures)` as response

```
prop.glm <- glm(cbind(Yes, No) ~ Class, data = tit.prop, family = binomial)
```

Call:

```
glm(formula = cbind(Yes, No) ~ Class, family = binomial, data = tit.prop)
```

Deviance Residuals:

| Min | 1Q | Median | 3Q | Max |
|---------|---------|--------|--------|---------|
| -9.6404 | -0.2915 | 1.5698 | 5.0366 | 10.1516 |

Coefficients:

| | Estimate | Std. Error | z value | Pr(> z) | | |
|----------------|----------|------------|----------|--------------|---------|---|
| (Intercept) | 0.5092 | 0.1146 | 4.445 | 8.79e-06 *** | | |
| Class2nd | -0.8565 | 0.1661 | -5.157 | 2.51e-07 *** | | |
| Class3rd | -1.5965 | 0.1436 | -11.114 | < 2e-16 *** | | |
| ClassCrew | -1.6643 | 0.1390 | -11.972 | < 2e-16 *** | | |
| --- | | | | | | |
| Signif. codes: | 0 '***' | 0.001 '**' | 0.01 '*' | 0.05 '.' | 0.1 ' ' | 1 |

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 671.96 on 13 degrees of freedom

Effects

```
model: cbind(Yes, No) ~ Class
```

Class effect

Class

| 1st | 2nd | 3rd | Crew |
|-----|-----|-----|------|
|-----|-----|-----|------|

| | | | |
|-----------|-----------|-----------|-----------|
| 0.6246154 | 0.4140351 | 0.2521246 | 0.2395480 |
|-----------|-----------|-----------|-----------|

Compare with former model based on binary data:

```
model: survived ~ class
```

class effect

class

| crew | first | second | third |
|------|-------|--------|-------|
|------|-------|--------|-------|

| | | | |
|-----------|-----------|-----------|-----------|
| 0.2395480 | 0.6246154 | 0.4140351 | 0.2521246 |
|-----------|-----------|-----------|-----------|

Logistic regression with continuous predictors

Example dataset: [GDP and infant mortality](#)

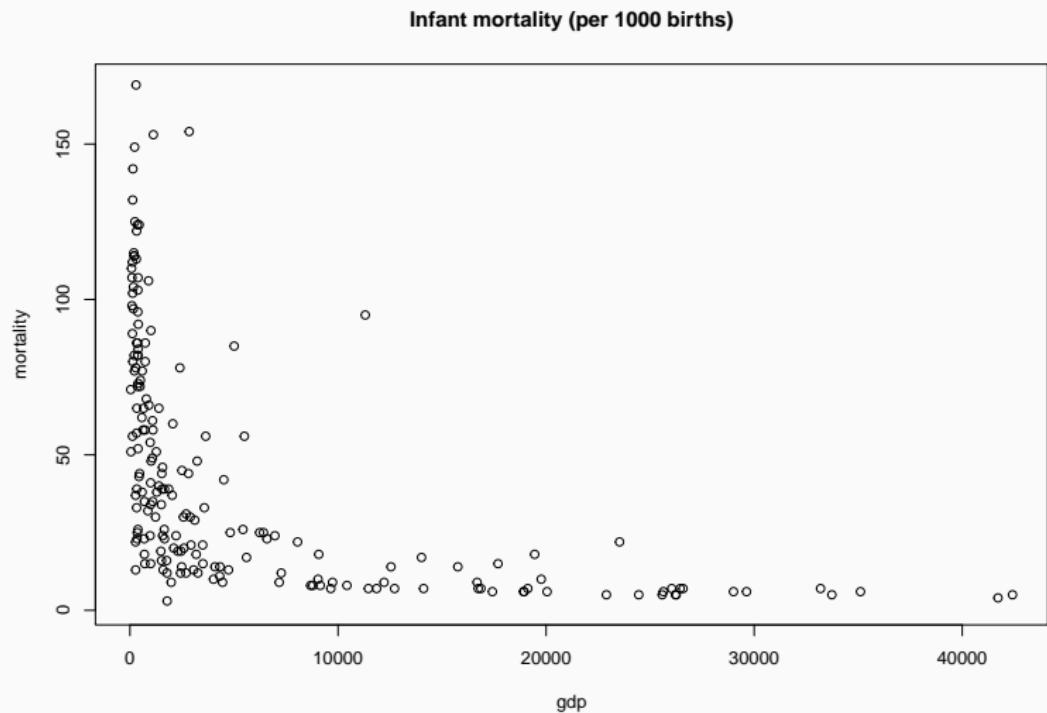
Read `UN_GDP_infantmortality.csv`.

```
country          mortality          gdp
Length:207      Min.   : 2.00      Min.   : 36
Class :character 1st Qu.: 12.00    1st Qu.: 442
Mode  :character Median : 30.00    Median : 1779
                  Mean   : 43.48    Mean   : 6262
                  3rd Qu.: 66.00    3rd Qu.: 7272
                  Max.   :169.00    Max.   :42416
                  NA's   :6          NA's   :10
```

Q: Is infant mortality related to GDP?

<https://pollev.com/franciscorod726>

Visualising data



Fit model

```
gdp.glm <- glm(cbind(mortality, 1000 - mortality) ~ gdp,  
                 data = gdp, family = binomial)
```

Call:

```
glm(formula = cbind(mortality, 1000 - mortality) ~ gdp, family = binomial,  
     data = gdp)
```

Deviance Residuals:

| Min | 1Q | Median | 3Q | Max |
|---------|---------|---------|--------|---------|
| -9.2230 | -3.5163 | -0.5697 | 2.4284 | 13.5849 |

Coefficients:

| | Estimate | Std. Error | z value | Pr(> z) | | | | | | | |
|----------------|------------|------------|---------|------------|------|-----|------|----|-----|-----|---|
| (Intercept) | -2.657e+00 | 1.311e-02 | -202.76 | <2e-16 *** | | | | | | | |
| gdp | -1.279e-04 | 3.458e-06 | -36.98 | <2e-16 *** | | | | | | | |
| --- | | | | | | | | | | | |
| Signif. codes: | 0 | '***' | 0.001 | '**' | 0.01 | '*' | 0.05 | .' | 0.1 | ' ' | 1 |

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 6430.2 on 192 degrees of freedom

Effects

```
allEffects(gdp.glm)
```

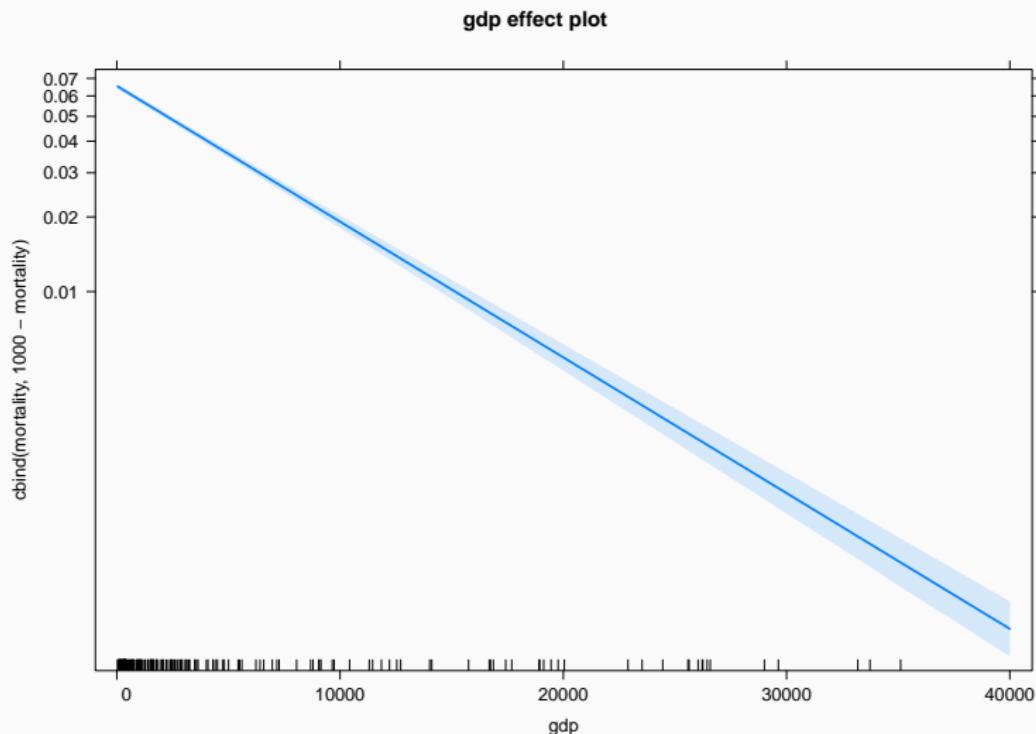
```
model: cbind(mortality, 1000 - mortality) ~ gdp
```

```
gdp effect
```

```
gdp
```

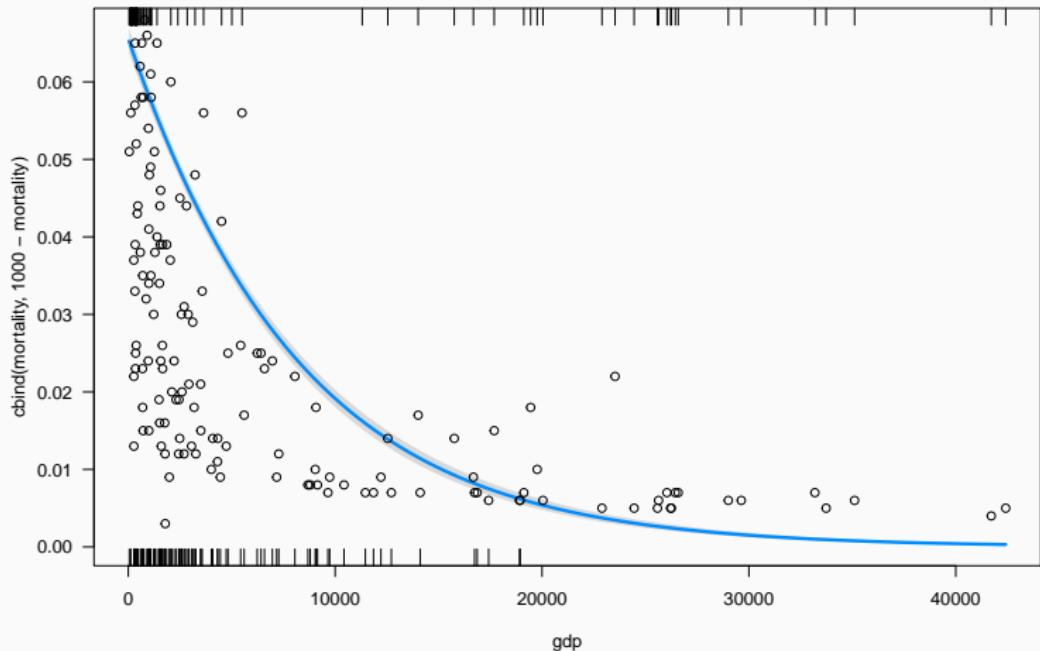
| 40 | 10000 | 20000 | 30000 | 40000 |
|--------------|--------------|--------------|--------------|--------------|
| 0.0652177296 | 0.0191438829 | 0.0054028095 | 0.0015096074 | 0.0004206154 |

Effects plot



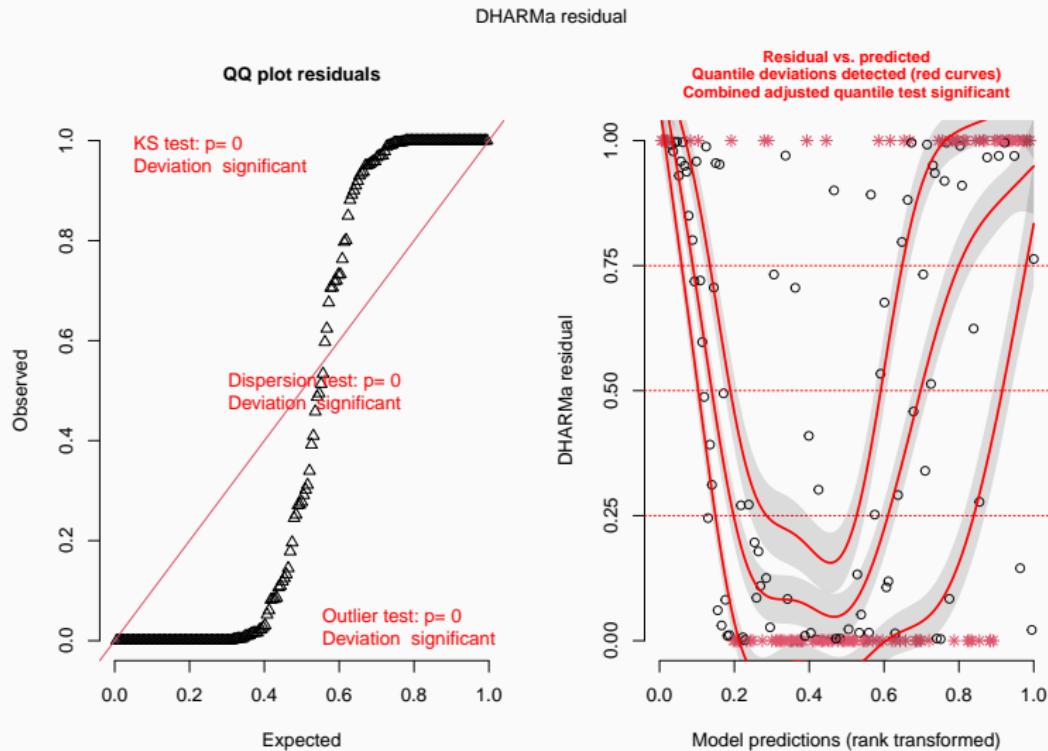
Plot model using visreg:

```
visreg(gdp.glm, scale = "response")
points(mortality/1000 ~ gdp, data = gdp)
```



Residuals diagnostics with DHARMA

```
simulateResiduals(gdp.glm, plot = TRUE)
```



Overdispersion

Overdispersion:

more variation in the data than assumed by statistical model

$$\text{Var}(y) = np(1 - p)$$

Testing for overdispersion (DHARMa)

```
simres <- simulateResiduals(gdp.glm, refit = TRUE)
testDispersion(simres, plot = FALSE)
```

DHARMa nonparametric dispersion test via mean deviance residual
vs. simulated-refitted

```
data: simres
dispersion = 21, p-value < 2.2e-16
alternative hypothesis: two.sided
```

`quasibinomial` allows us to model overdispersed binomial data

Overdispersion in logistic regression with proportion data

```
gdp.overdisp <- glm(cbind(mortality, 1000 - mortality) ~ gdp,  
                     data = gdp, family = quasibinomial)
```

Call:

```
glm(formula = cbind(mortality, 1000 - mortality) ~ gdp, family = quasibinomial,  
     data = gdp)
```

Deviance Residuals:

| Min | 1Q | Median | 3Q | Max |
|---------|---------|---------|--------|---------|
| -9.2230 | -3.5163 | -0.5697 | 2.4284 | 13.5849 |

Coefficients:

| | Estimate | Std. Error | t value | Pr(> t) | | | | | | | |
|----------------|------------|------------|---------|--------------|------|-----|------|-----|-----|----|---|
| (Intercept) | -2.657e+00 | 5.977e-02 | -44.465 | < 2e-16 *** | | | | | | | |
| gdp | -1.279e-04 | 1.577e-05 | -8.111 | 5.96e-14 *** | | | | | | | |
| --- | | | | | | | | | | | |
| Signif. codes: | 0 | '***' | 0.001 | '**' | 0.01 | '*' | 0.05 | '.' | 0.1 | '' | 1 |

(Dispersion parameter for quasibinomial family taken to be 20.7947)

Null deviance: 6430.2 on 192 degrees of freedom

Mean estimates do not change after accounting for overdispersion

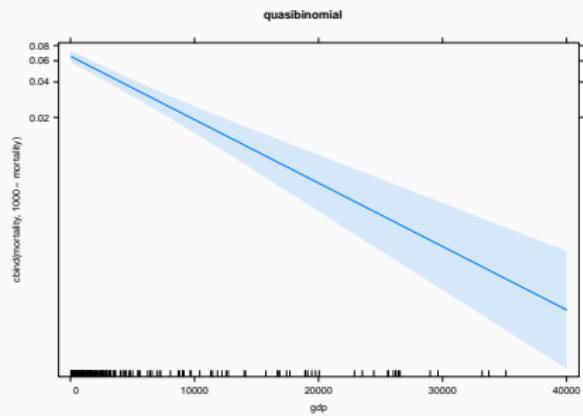
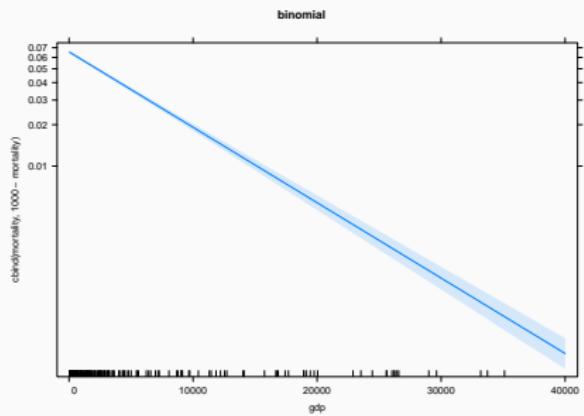
```
coef(gdp.overdisp)
```

```
(Intercept)          gdp  
-2.6574663734 -0.0001278976
```

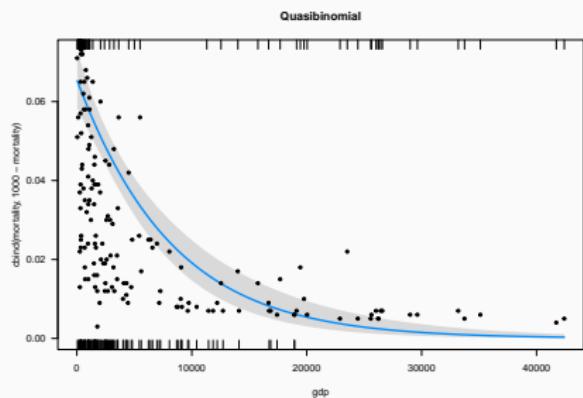
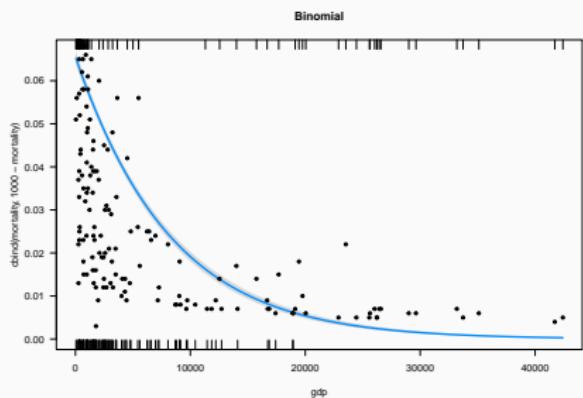
```
coef(gdp.glm)
```

```
(Intercept)          gdp  
-2.6574663734 -0.0001278976
```

But standard errors (uncertainty) do!



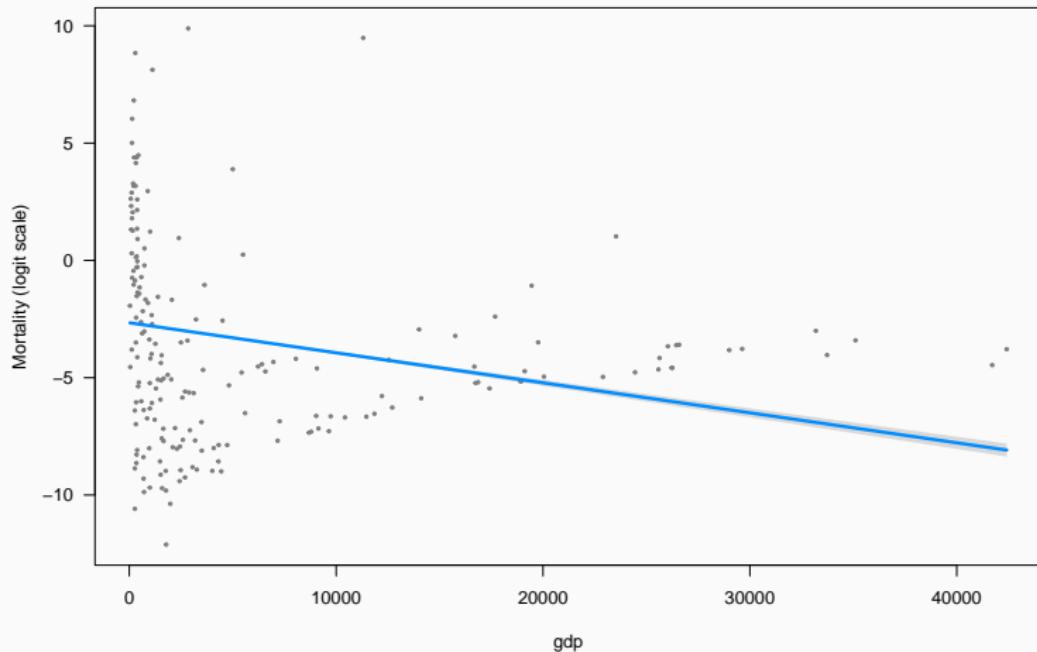
Plot model and data



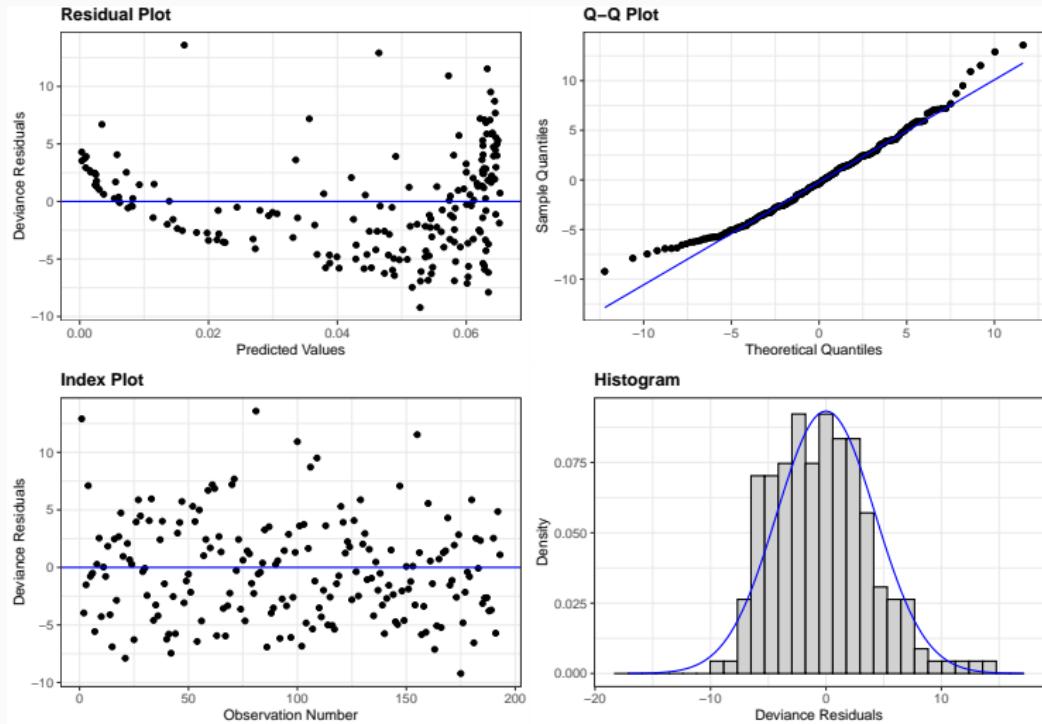
Think about the shape of
relationships

Think about the shape of relationships

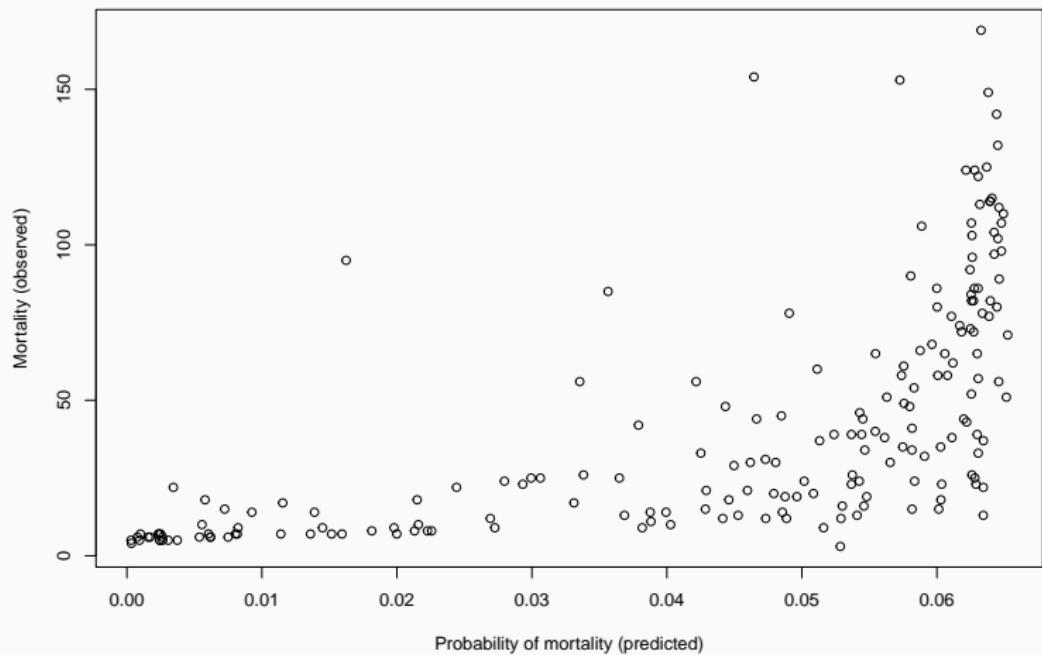
Not everything has to be linear...



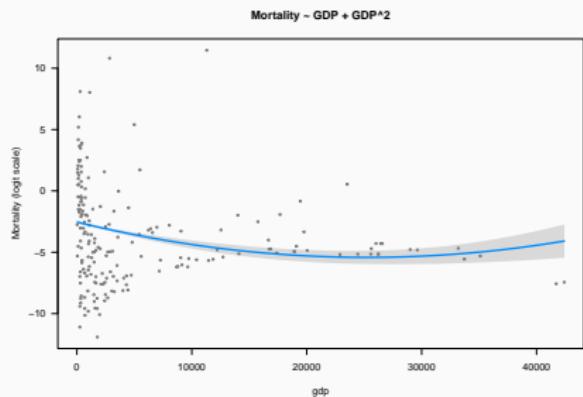
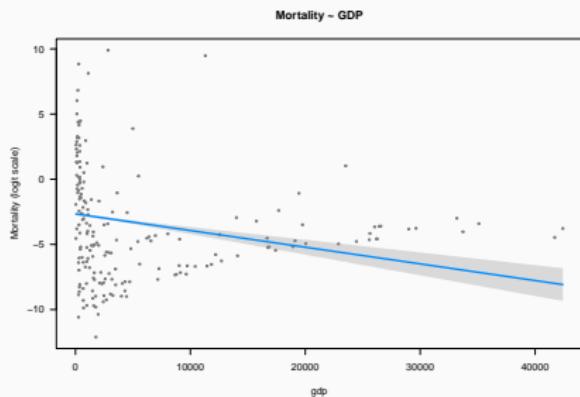
Residuals show non-linear pattern



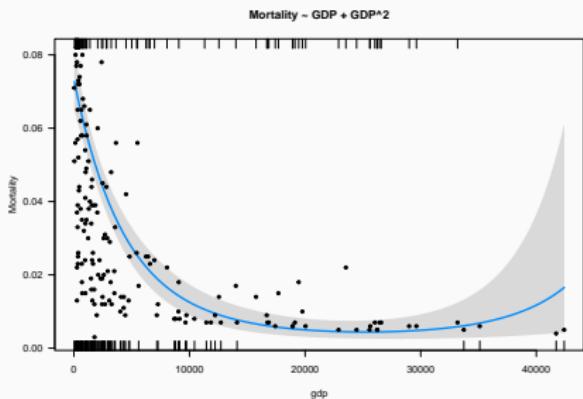
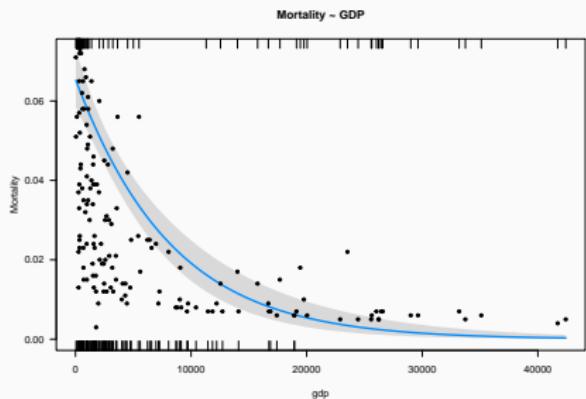
Calibration plot shows non-linear pattern



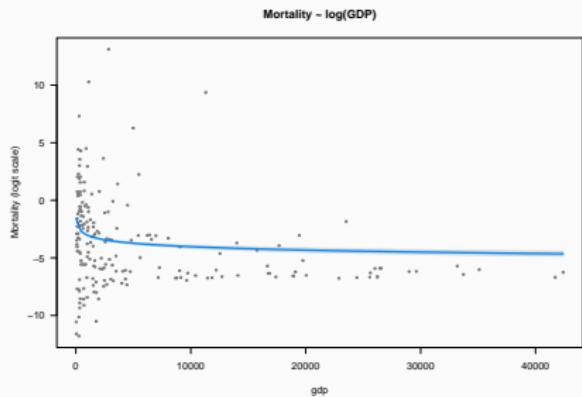
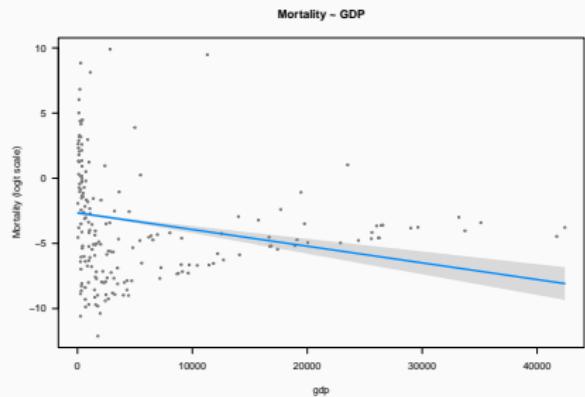
Trying polynomial predictor (GDP + GDP²)



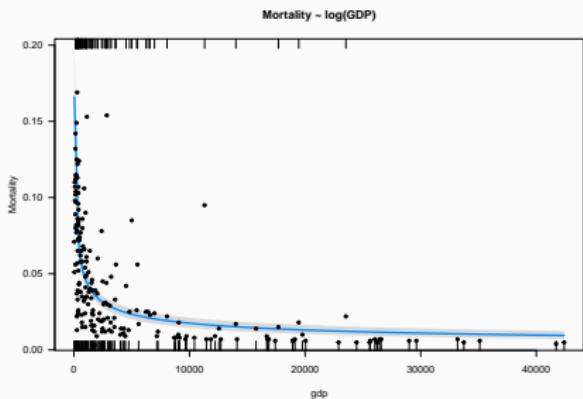
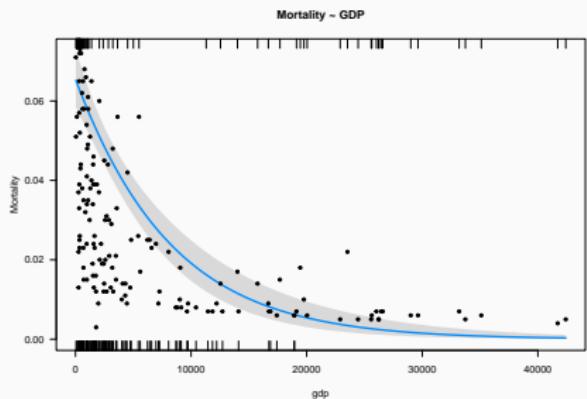
Think about the shape of relationships



Trying $\log(\text{GDP})$



Trying $\log(\text{GDP})$



More examples

- `seedset.csv`: Comparing seed set among plants (Data from [Harder et al. 2011](#))

More examples

- `seedset.csv`: Comparing seed set among plants (Data from [Harder et al. 2011](#))
- `moth.csv`: Probability of moth predation on trunk trees depending on morph (light/dark) and distance to Liverpool ([Bishop 1972](#))

More examples

- `seedset.csv`: Comparing seed set among plants (Data from [Harder et al. 2011](#))
- `moth.csv`: Probability of moth predation on trunk trees depending on morph (light/dark) and distance to Liverpool ([Bishop 1972](#))
- `soccer.csv`: Probability of scoring penalty depending on goalkeeper's team being ahead, behind or tied ([Roskes et al 2011](#))

Probability of scoring penalty

Data on penalty shots

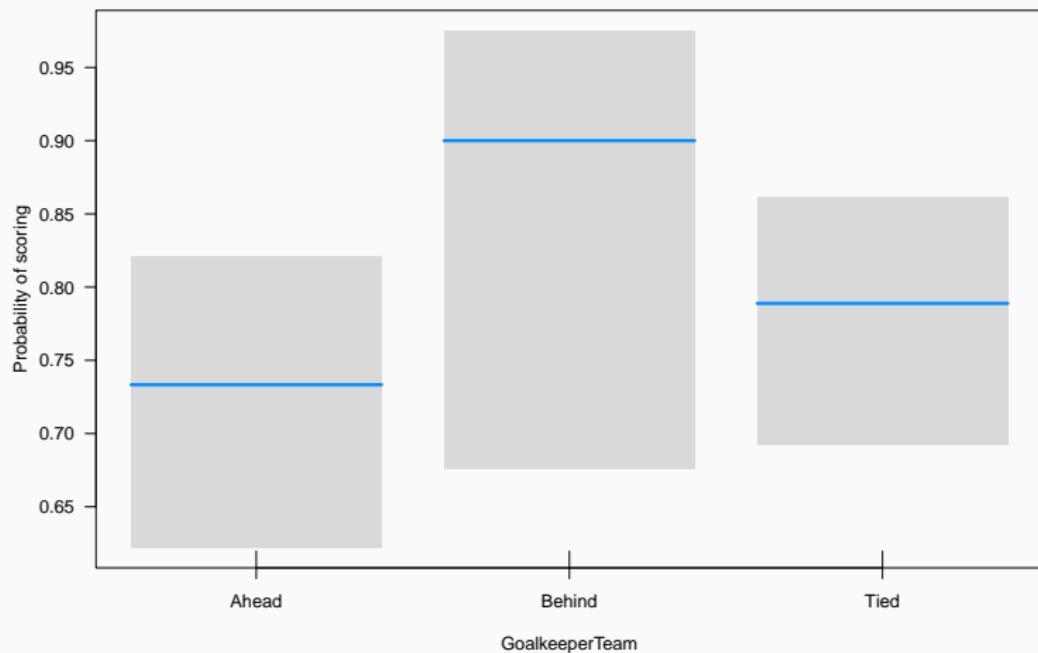
```
soccer <- read.csv("data/soccer.csv")  
soccer
```

| | GoalkeeperTeam | Nshots | Scored |
|---|----------------|--------|--------|
| 1 | Behind | 20 | 18 |
| 2 | Tied | 90 | 71 |
| 3 | Ahead | 75 | 55 |

Does probability of scoring penalty depends on match situation?

<https://pollev.com/franciscorod726>

Probability of scoring depending on match situation



Seed set among plants

Seed set among plants



Seed set among plants

```
# A tibble: 6 x 6
  species    plant  pcmass fertilized  seeds  ovulecnt
  <chr>      <dbl>   <dbl>      <dbl>   <dbl>      <dbl>
1 ferruginea 2     0          70      52      330
2 ferruginea 2     0.2        321     188      461
3 ferruginea 2     0.485      351     278      435
4 ferruginea 2     0.737      386     301      430
5 ferruginea 2     1          367     342      419
6 ferruginea 3     0          185     39       470
```

Questions:

<https://pollev.com/franciscorod726>

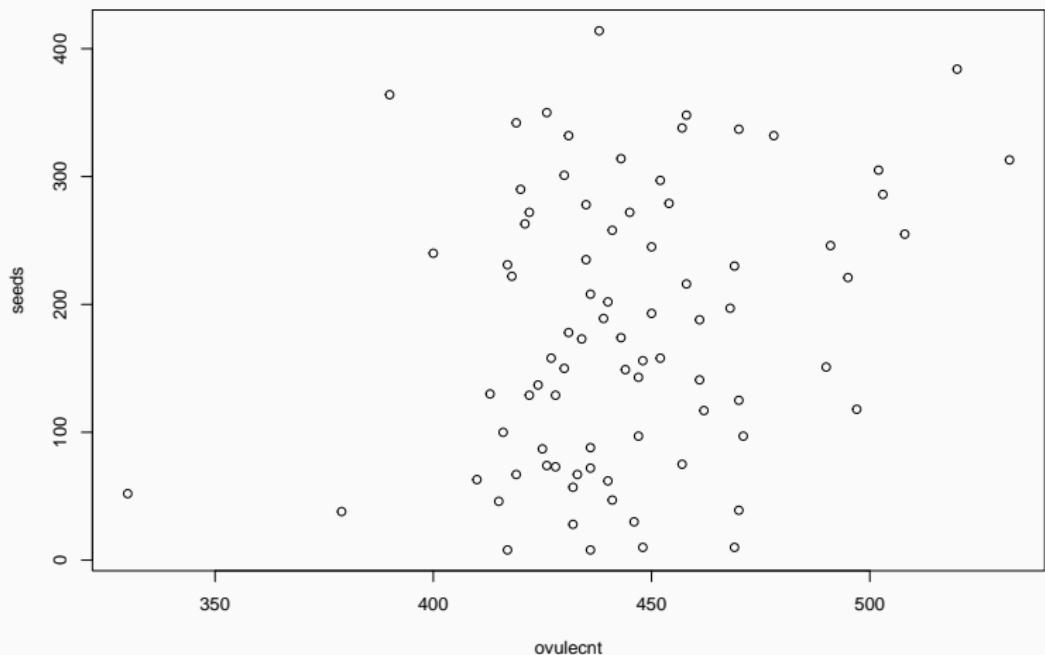
- Is seed set related to proportion of outcross pollen (pcmass)?

Questions:

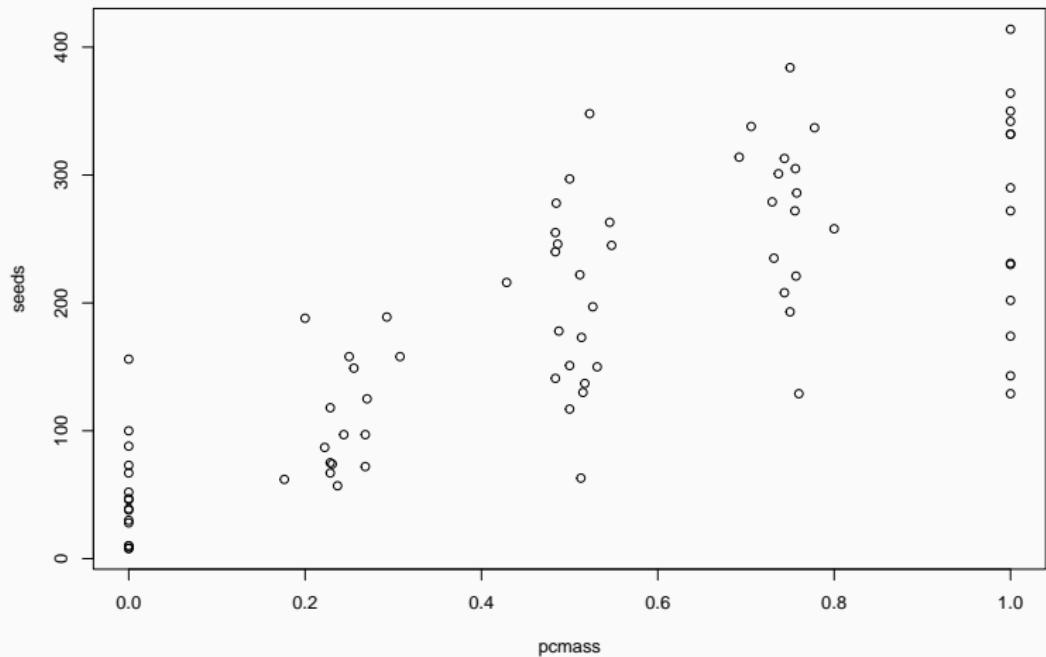
<https://pollev.com/franciscorod726>

- Is seed set related to proportion of outcross pollen (pcmass)?
- Which plant had lower seed set?

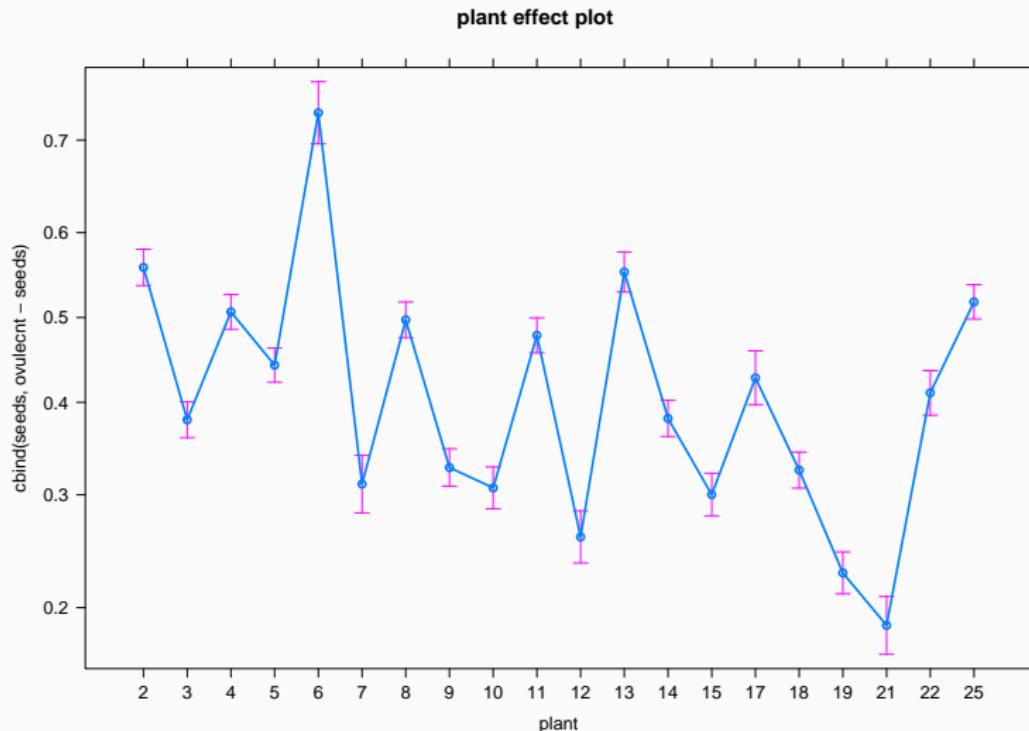
Number of seeds vs Number of ovules



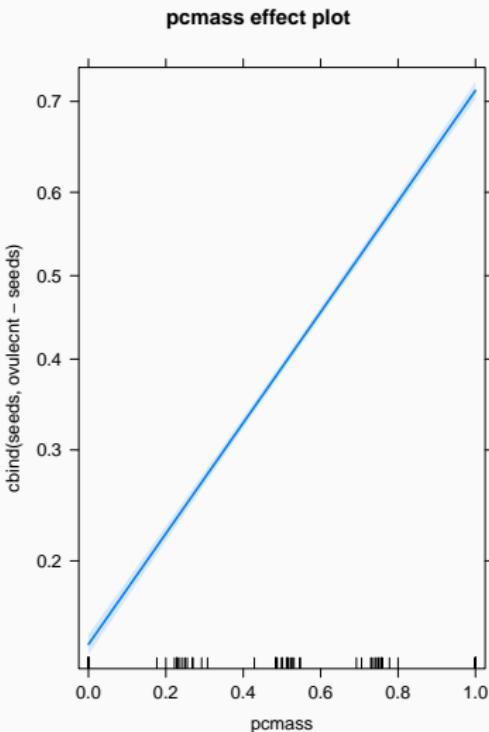
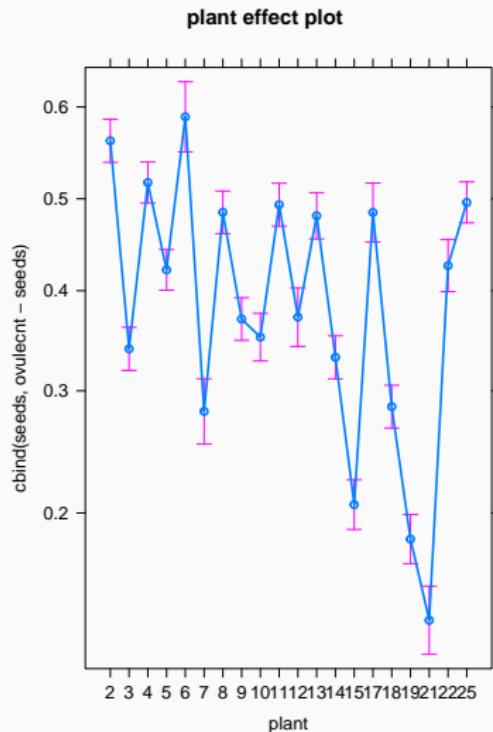
Number of seeds vs Proportion outcross pollen



Seed set across plants



Seed set ~ outcross pollen



GLM for count data: Poisson regression

Francisco Rodríguez-Sánchez

<https://frodriguezsanchez.net>

Types of response variable

- Gaussian: `lm`

Types of response variable

- Gaussian: `lm`
- Binary: `glm` (family `binomial` / `quasibinomial`)

Types of response variable

- Gaussian: `lm`
- Binary: `glm (family binomial / quasibinomial)`
- Counts: `glm (family poisson / quasipoisson)`

Poisson regression

- Response variable: Counts (0, 1, 2, 3...) - discrete
- Link function: **log**

Then

$$\log(N) = a + bx$$

$$N = e^{a+bx}$$

Example dataset: Seedling counts in quadrats

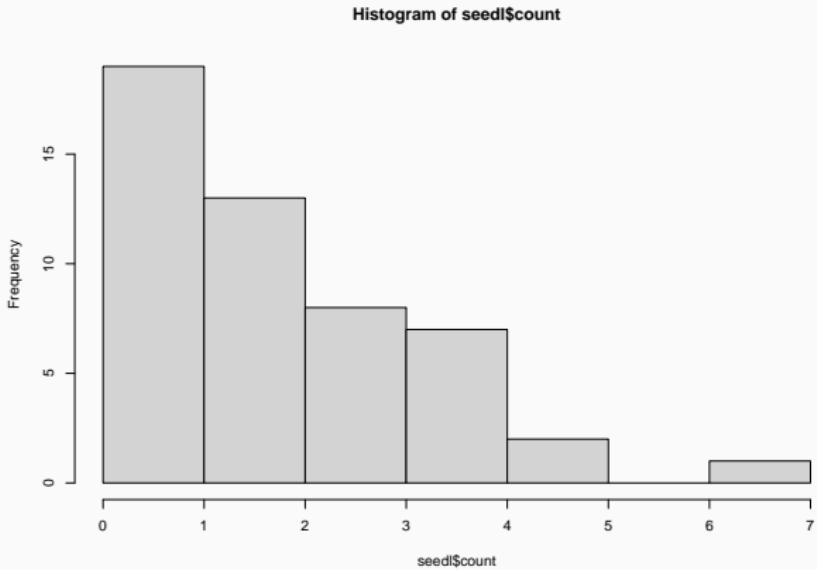
```
seedl <- read.csv("data/seedlings.csv")
```

| sample | count | light | area |
|---------------|--------------|----------------|--------------|
| Min. : 1.00 | Min. :0.00 | Min. : 2.571 | Min. :0.25 |
| 1st Qu.:13.25 | 1st Qu.:1.00 | 1st Qu.:26.879 | 1st Qu.:0.25 |
| Median :25.50 | Median :2.00 | Median :47.493 | Median :0.50 |
| Mean :25.50 | Mean :2.14 | Mean :47.959 | Mean :0.62 |
| 3rd Qu.:37.75 | 3rd Qu.:3.00 | 3rd Qu.:67.522 | 3rd Qu.:1.00 |
| Max. :50.00 | Max. :7.00 | Max. :99.135 | Max. :1.00 |

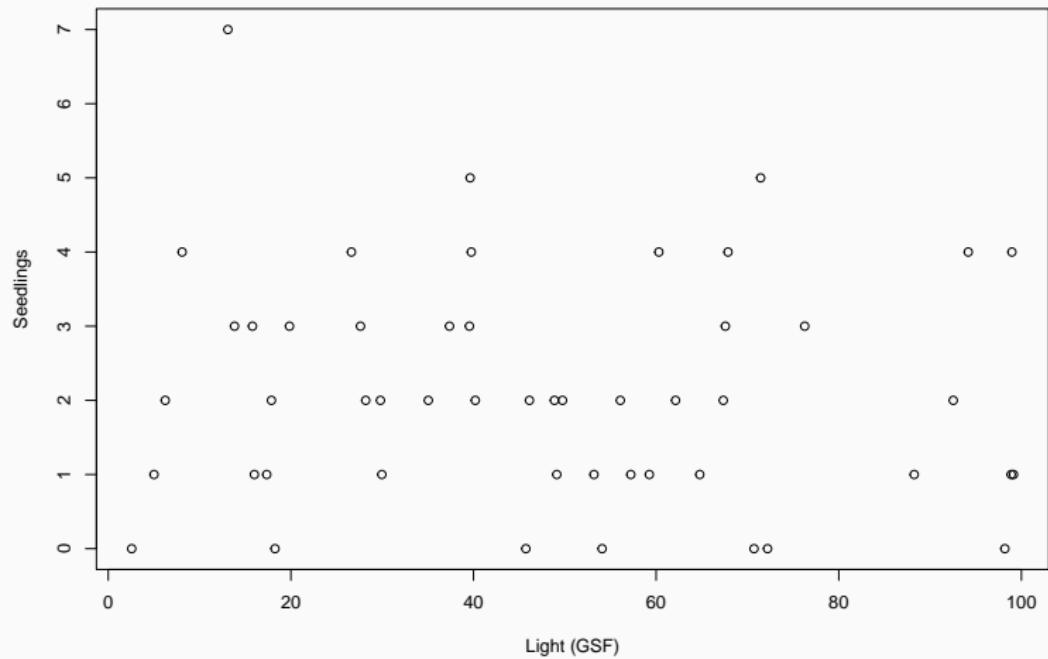
Exploring the data

```
table(seed1$count)
```

| | | | | | | |
|---|----|----|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 7 |
| 7 | 12 | 13 | 8 | 7 | 2 | 1 |



Relationship between Nseedlings and light?



Poisson regression

```
seedl.glm <- glm(count ~ light,  
                   data = seedl,  
                   family = poisson)
```

which corresponds to

```
equatiomatic::extract_eq(seedl.glm)
```

$$\log(E(\text{count})) = \alpha + \beta_1(\text{light}) \quad (1)$$

Interpreting Poisson GLM

Call:

```
glm(formula = count ~ light, family = poisson, data = seedl)
```

Deviance Residuals:

| Min | 1Q | Median | 3Q | Max |
|---------|---------|---------|--------|--------|
| -2.1906 | -0.8466 | -0.1110 | 0.5220 | 2.4577 |

Coefficients:

| | Estimate | Std. Error | z value | Pr(> z) |
|-------------|-----------|------------|---------|--------------|
| (Intercept) | 0.881805 | 0.188892 | 4.668 | 3.04e-06 *** |
| light | -0.002576 | 0.003528 | -0.730 | 0.465 |
| --- | | | | |

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

(Dispersion parameter for poisson family taken to be 1)

Null deviance: 63.029 on 49 degrees of freedom

Residual deviance: 62.492 on 48 degrees of freedom

AIC: 182.03

Number of Fisher Scoring iterations: 5

Parameter estimates are in log scale!

Parameter estimates (log scale):

```
coef(seedl.glm)[1]
```

(Intercept)

0.881805

We need to back-transform: apply the inverse of the logarithm

```
exp(coef(seedl.glm)[1])
```

(Intercept)

2.415255

Using effects package

```
allEffects(seed1.glm)
```

```
model: count ~ light
```

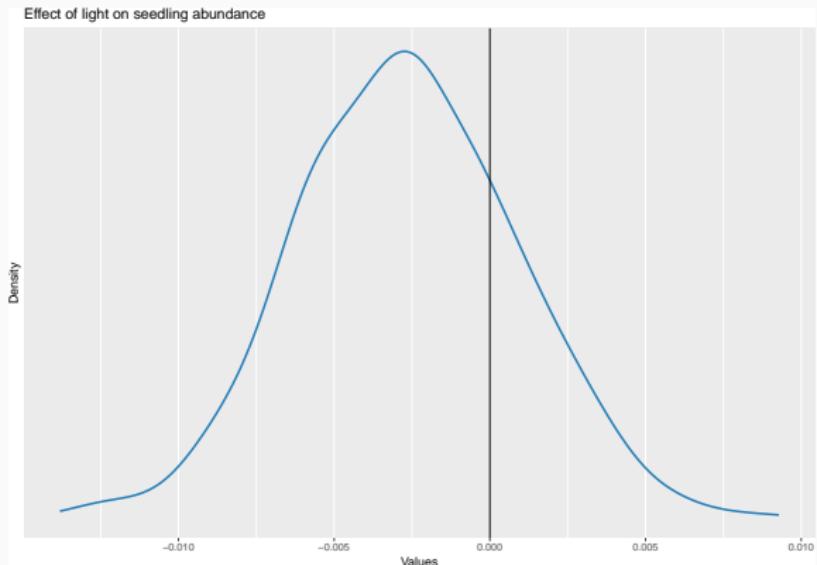
```
light effect  
light
```

| | 3 | 30 | 50 | 70 | 100 |
|--|---|----|----|----|-----|
|--|---|----|----|----|-----|

| | | | | | |
|--|----------|----------|----------|----------|----------|
| | 2.396665 | 2.235657 | 2.123408 | 2.016794 | 1.866826 |
|--|----------|----------|----------|----------|----------|

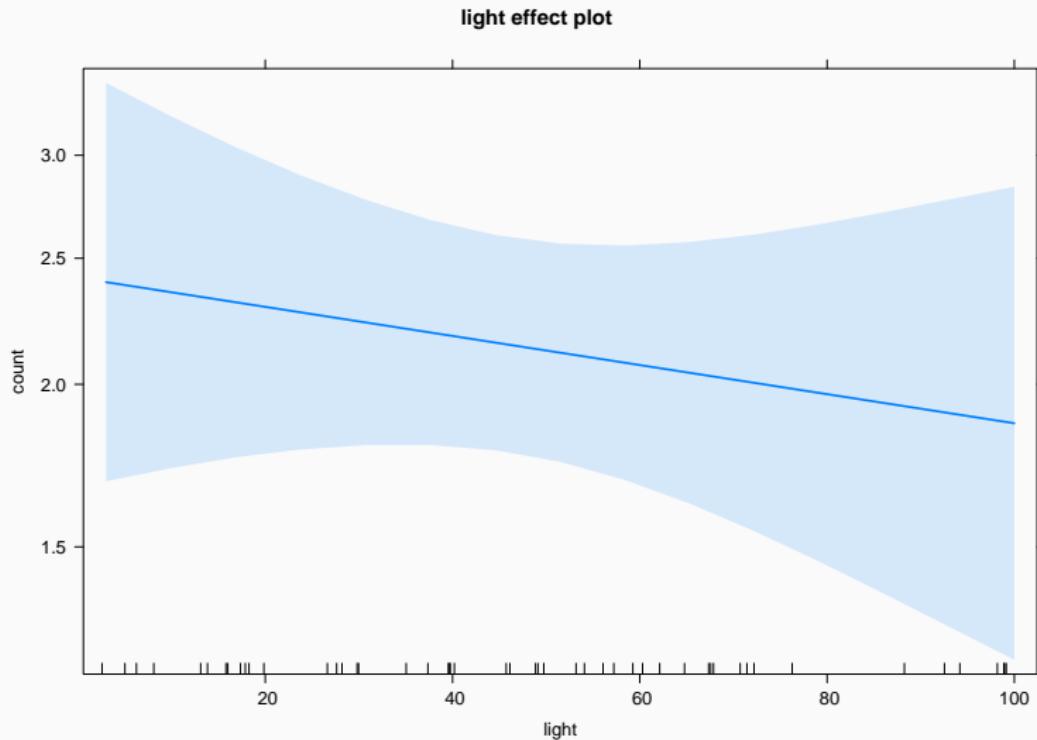
Estimated distribution of the slope parameter

```
library("parameters")
plot(simulate_parameters(seedl.glm)) +
  geom_vline(xintercept = 0) +
  ggtitle("Effect of light on seedling abundance")
```



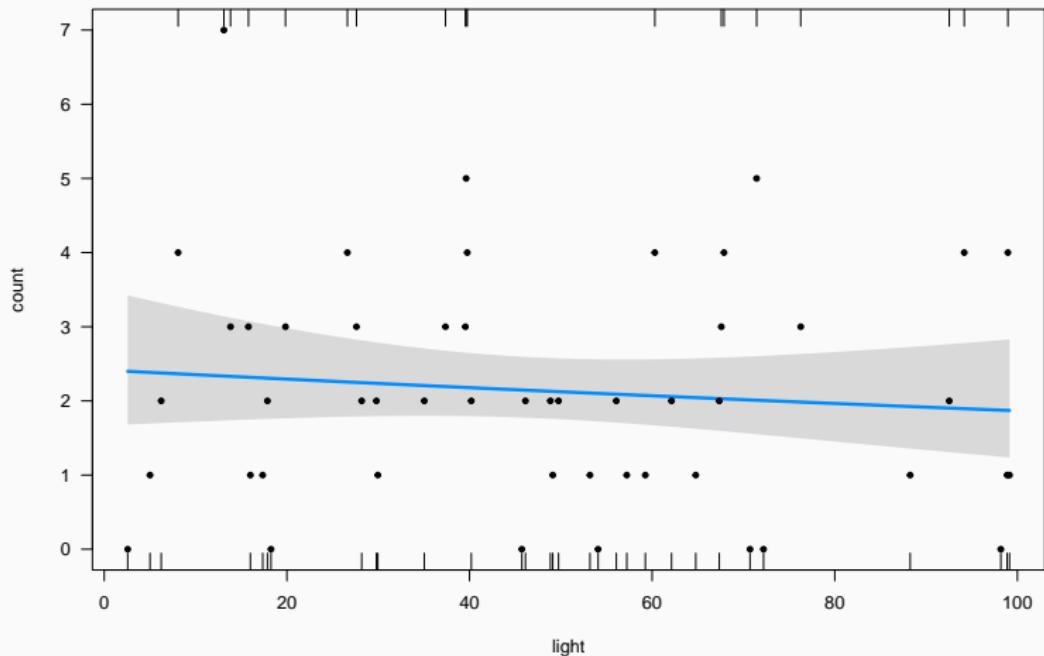
So what's the relationship between Nseedlings and light?

```
plot(allEffects(seedl.glm))
```



Using visreg

```
visreg(seedl.glm, scale = "response", ylim = c(0, 7))  
points(count ~ light, data = seedl, pch = 20)
```



Low R-squared

```
library("performance")
r2(seedl.glm)
```

```
# R2 for Generalized Linear Regression
Nagelkerke's R2: 0.015
```

Describing the model results

```
library("report")
report(seed1.glm)
```

We fitted a poisson model (estimated using ML) to predict count with light (formula: count ~ light). The model's explanatory power is very weak (Nagelkerke's R² = 0.01). The model's intercept, corresponding to light = 0, is at 0.88 (95% CI [0.50, 1.24], p < .001). Within this model:

- The effect of light is statistically non-significant and negative (beta = -2.58e-03, 95% CI [-9.57e-03, 4.28e-03], p = 0.465; Std. beta = -0.07, 95% CI [-0.27, 0.12])

Standardized parameters were obtained by fitting the model on a standardized version of the dataset. 95% Confidence Intervals (CIs) and p-values were computed using a Wald z-distribution approximation.

Model checking

Assumptions of Poisson regression

- Linearity (log response ~ predictors)

Assumptions of Poisson regression

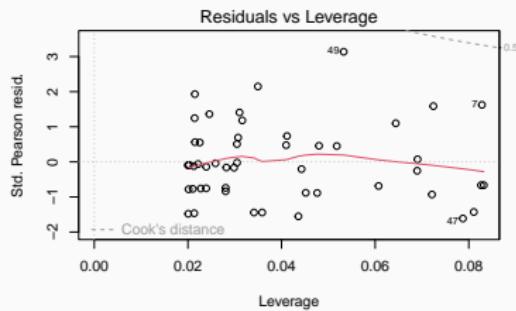
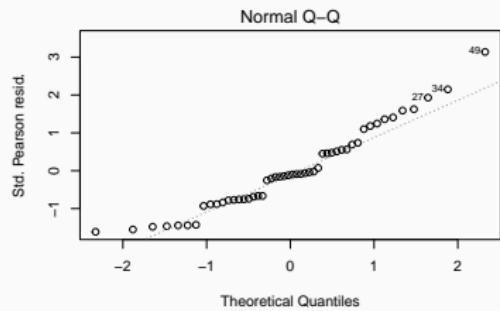
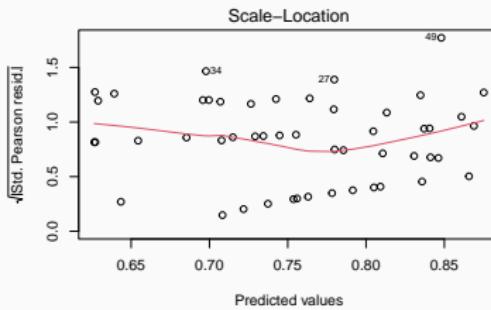
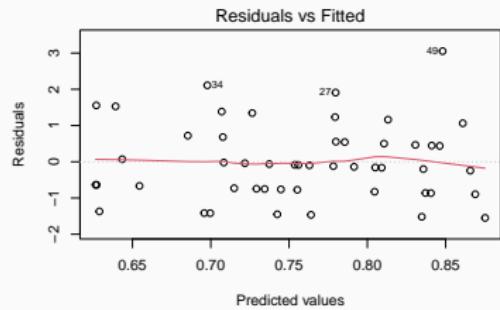
- Linearity (\log response \sim predictors)
- Observations are independent

Assumptions of Poisson regression

- Linearity (log response ~ predictors)
- Observations are independent
- Mean = Variance

Checking Poisson GLM

```
plot(seedl.glm)
```

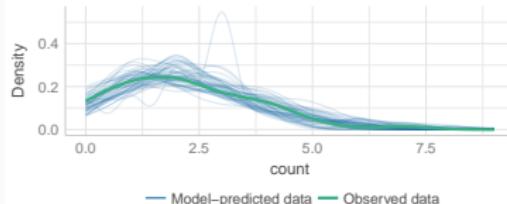


null device

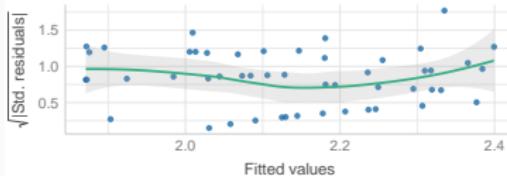
Checking Poisson GLM

```
check_model(seed1.glm)
```

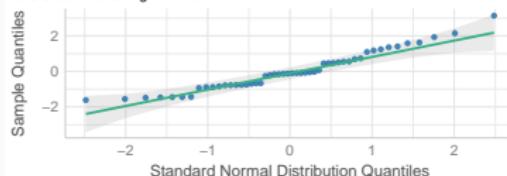
Posterior Predictive Check
Model-predicted lines should resemble observed data line



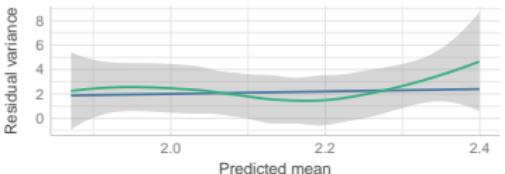
Homogeneity of Variance
Reference line should be flat and horizontal



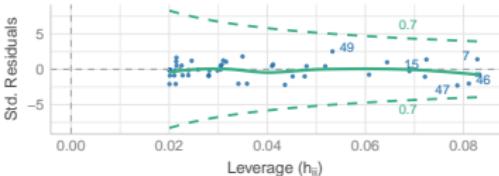
Normality of Residuals
Dots should fall along the line



Overdispersion and zero-inflation
Observed residual variance (green) should follow predicted residual variance (blue)

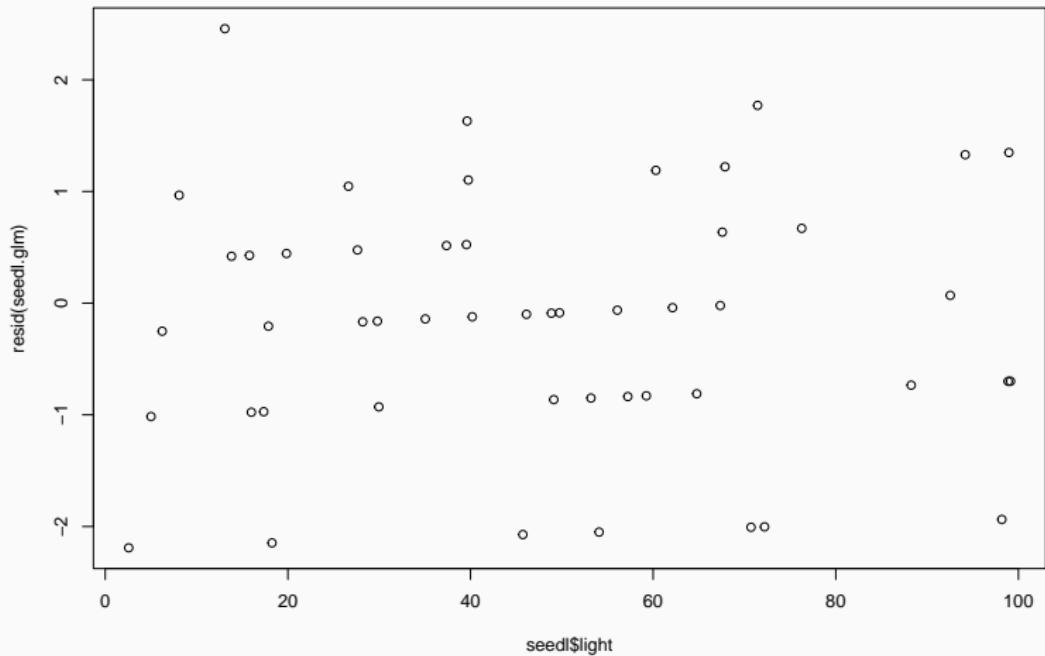


Influential Observations
Points should be inside the contour lines



Is there pattern of residuals along predictor?

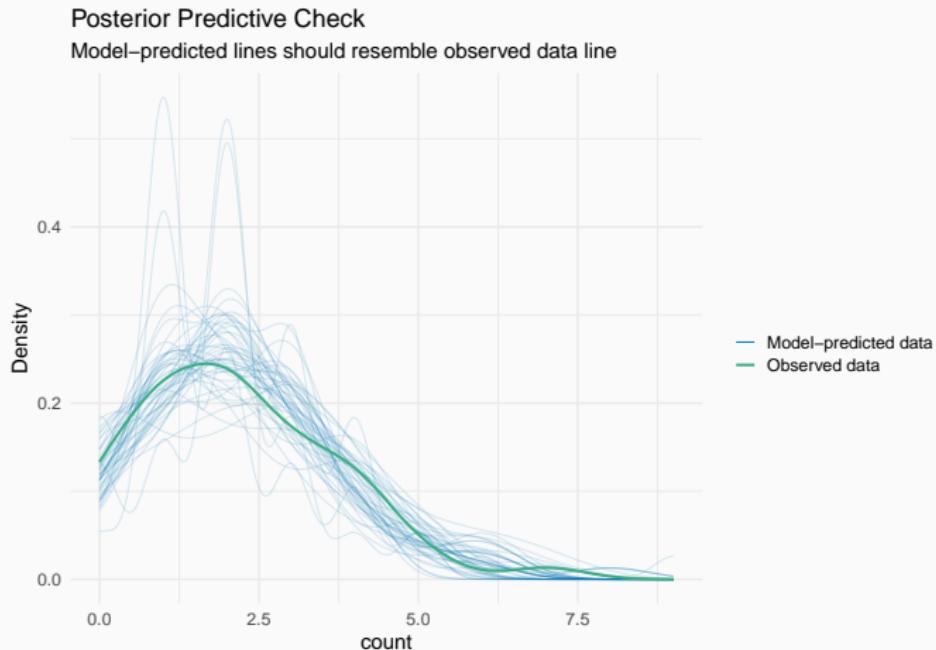
```
plot(seedl$light, resid(seedl.glm))
```



Posterior predictive checking

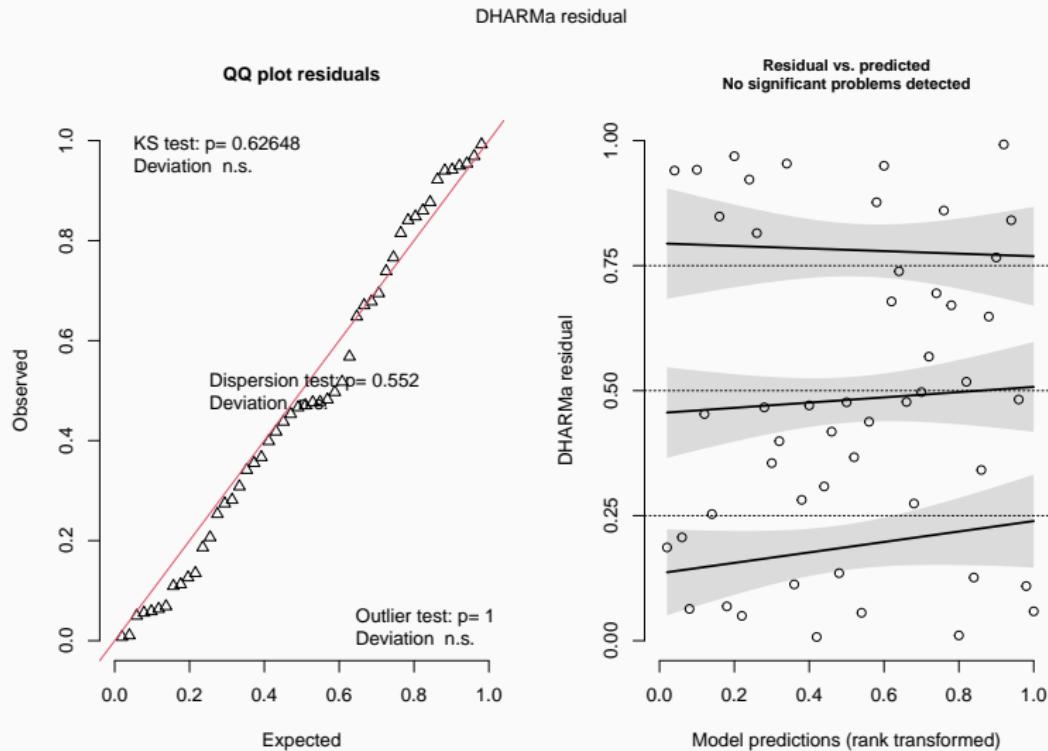
Simulate data from fitted model (y_{rep}) and compare with observed data (y)

```
check_predictions(seed1.glm)
```



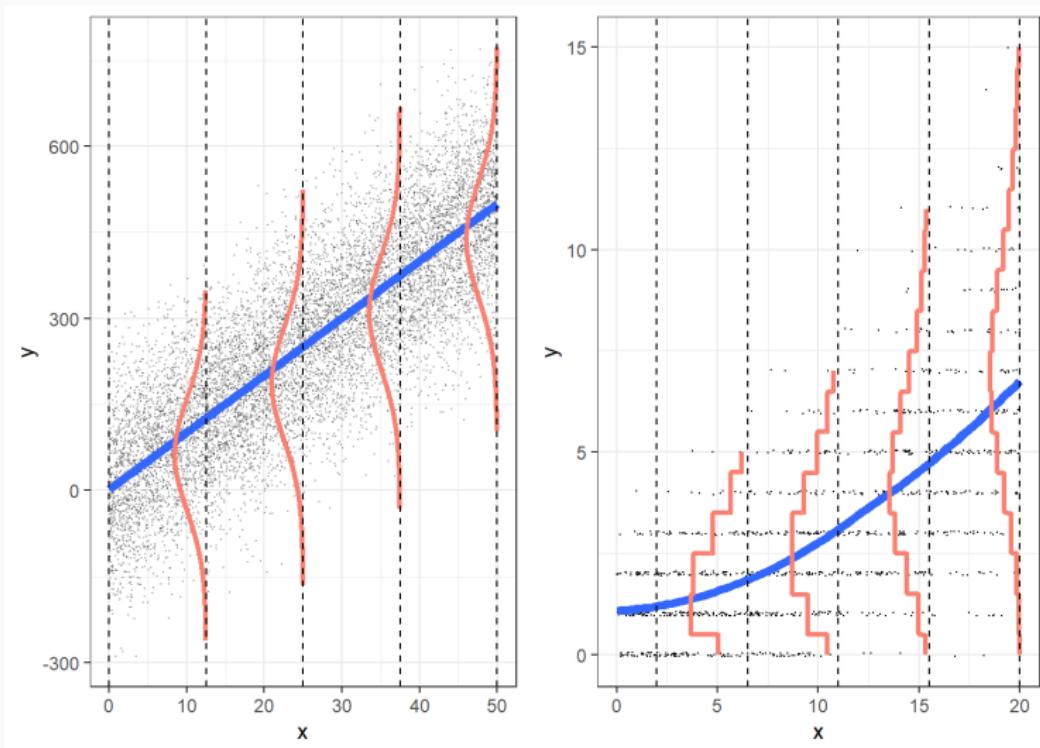
Residuals diagnostics with DHARMA

```
simulateResiduals(seed1.glm, plot = TRUE)
```



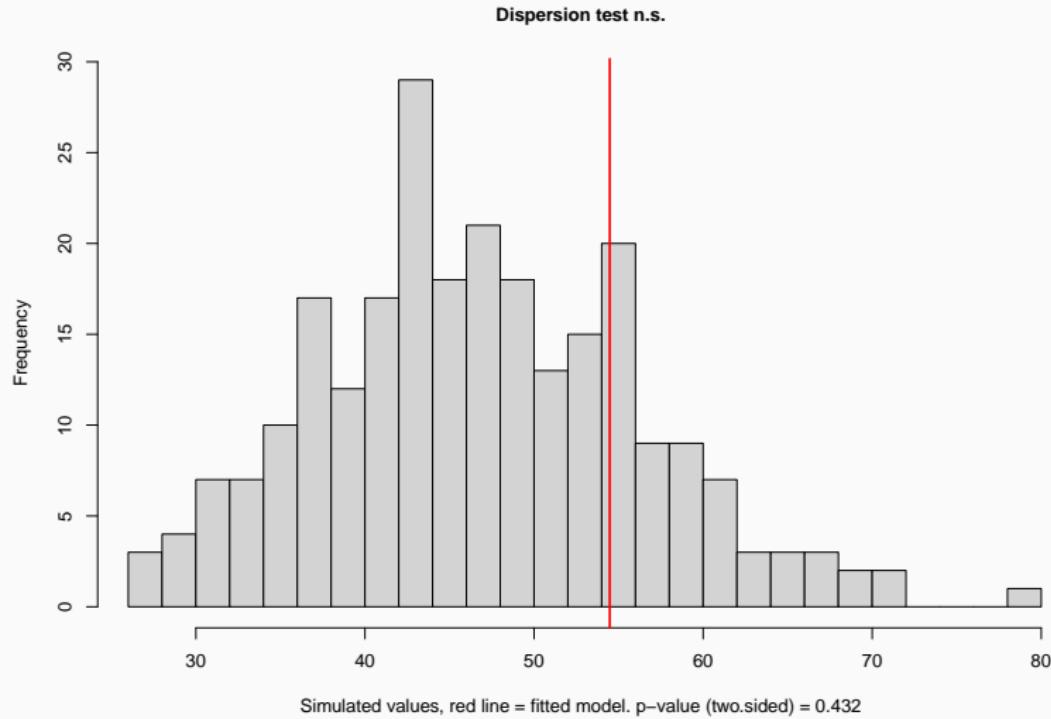
Overdispersion

Poisson GLM assumes mean = variance



Always check overdispersion with count data

```
simres <- simulateResiduals(seed1.glm, refit = TRUE)  
testDispersion(simres)
```



Accounting for overdispersion in count data

- Use family quasipoisson

Accounting for overdispersion in count data

- Use family `quasipoisson`
- Use negative binomial distribution (`MASS::glm.nb`)

Accounting for overdispersion in count data

- Use family `quasipoisson`
- Use negative binomial distribution (`MASS::glm.nb`)
- Include observation-level random effect (e.g. see [Harrison 2014](#))

Accounting for overdispersion with family quasipoisson

Call:

```
glm(formula = count ~ light, family = quasipoisson, data = seed1)
```

Deviance Residuals:

| Min | 1Q | Median | 3Q | Max |
|---------|---------|---------|--------|--------|
| -2.1906 | -0.8466 | -0.1110 | 0.5220 | 2.4577 |

Coefficients:

| | Estimate | Std. Error | t value | Pr(> t) | | |
|----------------|-----------|------------|----------|--------------|---------|---|
| (Intercept) | 0.881805 | 0.201230 | 4.382 | 6.37e-05 *** | | |
| light | -0.002576 | 0.003758 | -0.685 | 0.496 | | |
| --- | | | | | | |
| Signif. codes: | 0 '***' | 0.001 '**' | 0.01 '*' | 0.05 '.' | 0.1 ' ' | 1 |

(Dispersion parameter for quasipoisson family taken to be 1.134907)

Null deviance: 63.029 on 49 degrees of freedom
Residual deviance: 62.492 on 48 degrees of freedom
AIC: NA

Number of Fisher Scoring iterations: 5

Mean estimates do not change after accounting for overdispersion

```
allEffects(seedl.overdisp)
```

```
model: count ~ light

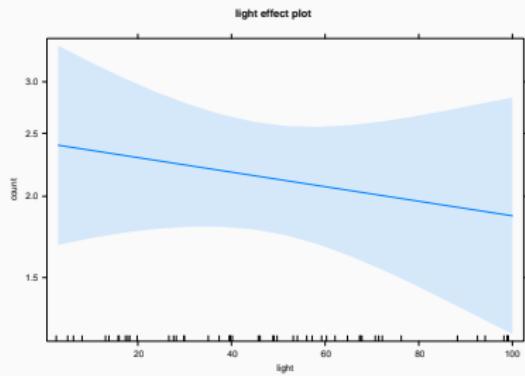
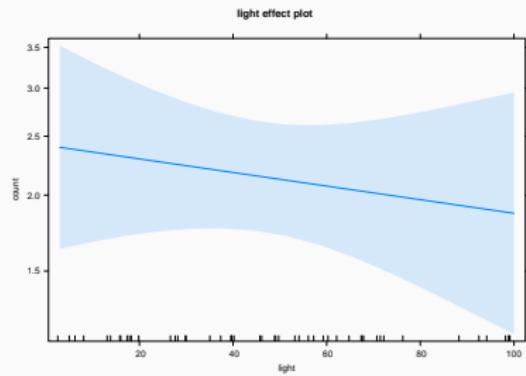
light effect
light
      3       30       50       70       100
2.396665 2.235657 2.123408 2.016794 1.866826
```

```
allEffects(seedl.glm)
```

```
model: count ~ light

light effect
light
      3       30       50       70       100
2.396665 2.235657 2.123408 2.016794 1.866826
```

But standard errors may change



Accounting for overdispersion using negative binomial

```
library("MASS")
seedl.nb <- glm.nb(count ~ light, data = seedl)
```

Call:

```
glm.nb(formula = count ~ light, data = seedl, init.theta = 22.23419419,
       link = log)
```

Deviance Residuals:

| Min | 1Q | Median | 3Q | Max |
|---------|---------|---------|--------|--------|
| -2.1349 | -0.8162 | -0.1061 | 0.4954 | 2.2814 |

Coefficients:

| | Estimate | Std. Error | z value | Pr(> z) |
|-------------|-----------|------------|---------|-------------|
| (Intercept) | 0.881996 | 0.198213 | 4.450 | 8.6e-06 *** |
| light | -0.002580 | 0.003691 | -0.699 | 0.485 |

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

(Dispersion parameter for Negative Binomial(22.2342) family taken to be 1)

Null deviance: 58.247 on 49 degrees of freedom

Comparing Poisson and Negative Binomial

```
compare_models(seedl.glm, seedl.nb)
```

| Parameter | | seedl.glm | | seedl.nb |
|--------------|--|-------------------------|--|-------------------------|
| (Intercept) | | 0.88 (0.51, 1.25) | | 0.88 (0.49, 1.27) |
| light | | -2.58e-03 (-0.01, 0.00) | | -2.58e-03 (-0.01, 0.00) |
| Observations | | 50 | | 50 |

```
compare_performance(seedl.glm, seedl.nb)
```

Comparison of Model Performance Indices

| Name | | Model | | AIC | | AIC weights | | BIC | | BIC weights | | Nagelkerke's R2 | | RMSE | |
|-----------|--|--------|--|---------|--|-------------|--|---------|--|-------------|--|-----------------|--|-------|--|
| seedl.glm | | glm | | 182.034 | | 0.710 | | 185.858 | | 0.864 | | 0.015 | | 1.529 | |
| seedl.nb | | negbin | | 183.827 | | 0.290 | | 189.563 | | 0.136 | | 0.014 | | 1.529 | |

What if survey plots have
different area?

Shall we *standardise* counts dividing by sampling plot area?

Model would be: count/area ~ light

| | sample | count | light | area |
|---|--------|-------|----------|------|
| 1 | 1 | 0 | 70.71854 | 0.50 |
| 2 | 2 | 1 | 88.26021 | 0.25 |
| 3 | 3 | 2 | 67.35133 | 0.50 |
| 4 | 4 | 3 | 67.57850 | 1.00 |
| 5 | 5 | 4 | 26.63098 | 0.25 |
| 6 | 6 | 3 | 15.79433 | 1.00 |

Avoid regression of ratios

J. R. Statist. Soc. A (1993)
156, Part 3, pp. 379–392

Spurious Correlation and the Fallacy of the Ratio Standard Revisited

By RICHARD A. KRONMAL†

<https://doi.org/10.2307/2983064>

Use offset to account for variable sampling effort

```
seedl.offset <- glm(count ~ light,  
                      offset = log(area),  
                      data = seedl,  
                      family = poisson)
```

Note estimates now referred to area units!

Call:

```
glm(formula = count ~ light, family = poisson, data = seedl,  
    offset = log(area))
```

Deviance Residuals:

| Min | 1Q | Median | 3Q | Max |
|---------|---------|--------|--------|--------|
| -2.9918 | -1.0142 | 0.1673 | 0.8401 | 3.8230 |

Coefficients:

| | Estimate | Std. Error | z value | Pr(> z) |
|-------------|-----------|------------|---------|------------|
| (Intercept) | 1.513185 | 0.183245 | 8.258 | <2e-16 *** |
| light | -0.005674 | 0.003384 | -1.677 | 0.0936 . |

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

(Dispersion parameter for poisson family taken to be 1)

Note estimates now referred to area units!

```
exp(coef(seedl.offset)[1])
```

(Intercept)

4.541173

Prediction

Predicting number of seedlings given light

```
new.lights <- data.frame(light = c(10, 90))
predict(seedl.glm, newdata = new.lights, type = "response", se.fit
```

```
$fit
```

```
1 2
```

```
2.353841 1.915533
```

```
$se.fit
```

```
1 2
```

```
0.3756992 0.3502446
```

```
$residual.scale
```

```
[1] 1
```

Poisson GLM: more examples

- Infant mortality ~ GDP

Poisson GLM: more examples

- Infant mortality ~ GDP
- Number of cones consumed by squirrels ([data](#))

Poisson GLM: more examples

- Infant mortality ~ GDP
- Number of cones consumed by squirrels ([data](#))
- Elephant matings ([Poole 1989](#))

Modelling zero-inflated (and overdispersed) count data

Francisco Rodríguez-Sánchez

<https://frodriguezsanchez.net>

How many eggs in nests?



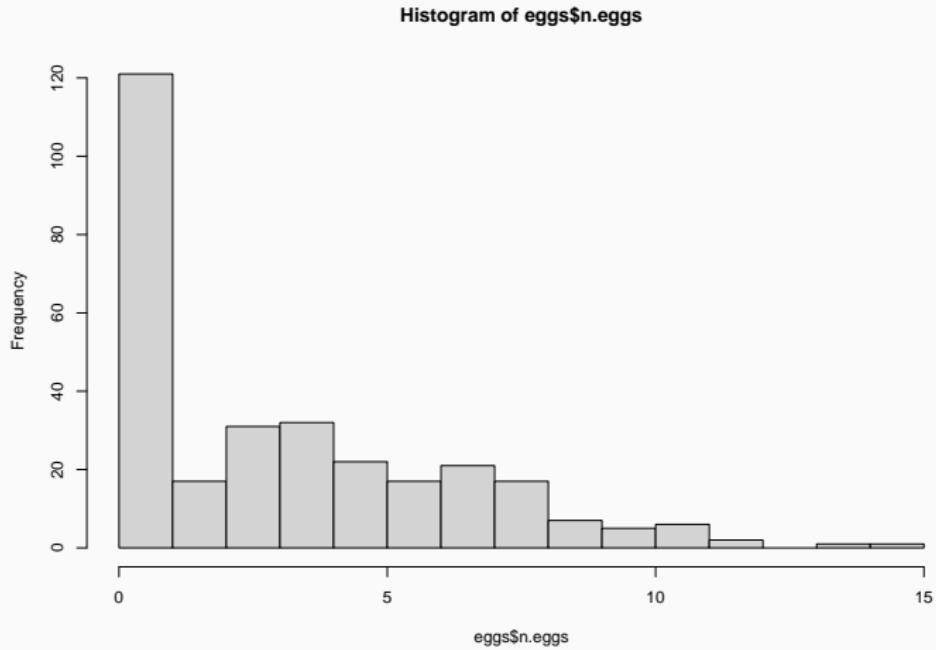
```
eggs <- read.csv("data/eggs.csv")
```

| diameter | old | n.eggs |
|----------|-----|--------|
| 14 | no | 4 |
| 8 | yes | 0 |
| 7 | yes | 0 |

diameter: nest diameter (cm)

old: does nest look old/abandoned?

How many eggs in nests?



Many zeros does not mean you need a zero-inflated model!

Check model afterwards

How many eggs in nests?

- Nests may be occupied or not

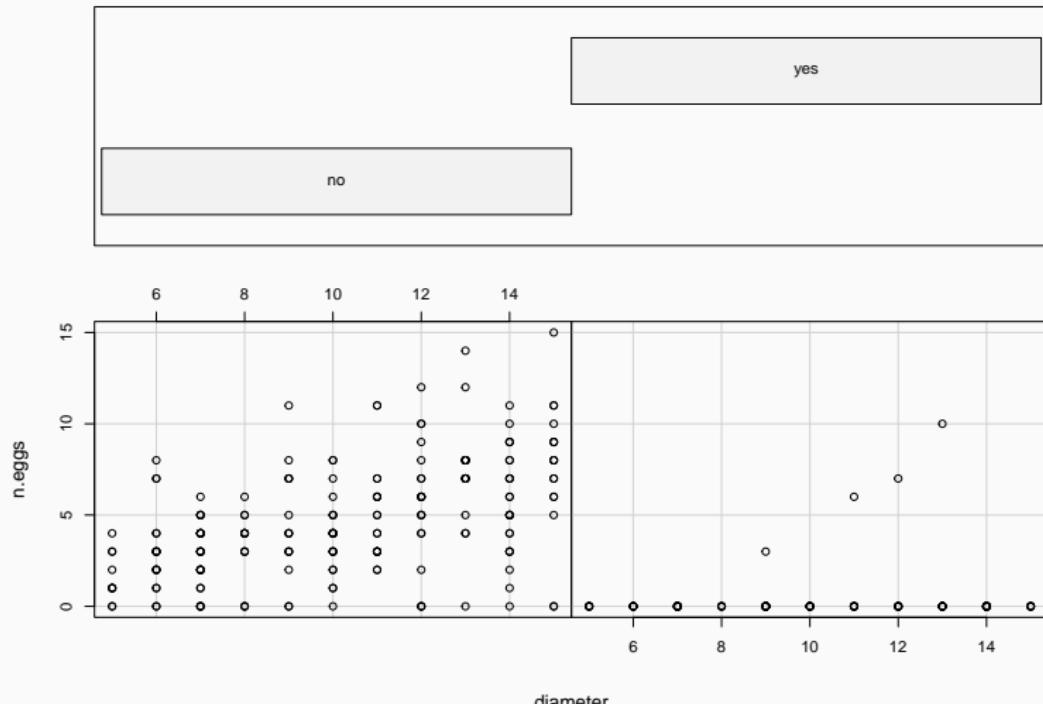
How many eggs in nests?

- Nests may be occupied or not
- Occupied nests may not have eggs (too soon, predation, etc)

Number of eggs ~ nest diameter * old appearance

```
coplot(n.eggs ~ diameter | old, data = eggs)
```

Given : old



Trying Poisson GLM

```
eggs.poi <- glm(n.eggs ~ old * diameter,  
                  data = eggs,  
                  family = poisson)
```

Trying Poisson GLM

Call:

```
glm(formula = n.eggs ~ old * diameter, family = poisson, data = eggs)
```

Deviance Residuals:

| Min | 1Q | Median | 3Q | Max |
|---------|---------|---------|--------|--------|
| -3.8905 | -0.8784 | -0.4514 | 0.3892 | 6.6795 |

Coefficients:

| | Estimate | Std. Error | z value | Pr(> z) |
|-----------------|----------|------------|---------|--------------|
| (Intercept) | 0.30773 | 0.12883 | 2.389 | 0.0169 * |
| oldyes | -3.78879 | 0.92230 | -4.108 | 3.99e-05 *** |
| diameter | 0.11441 | 0.01105 | 10.354 | < 2e-16 *** |
| oldyes:diameter | 0.08513 | 0.07634 | 1.115 | 0.2648 |
| --- | | | | |

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

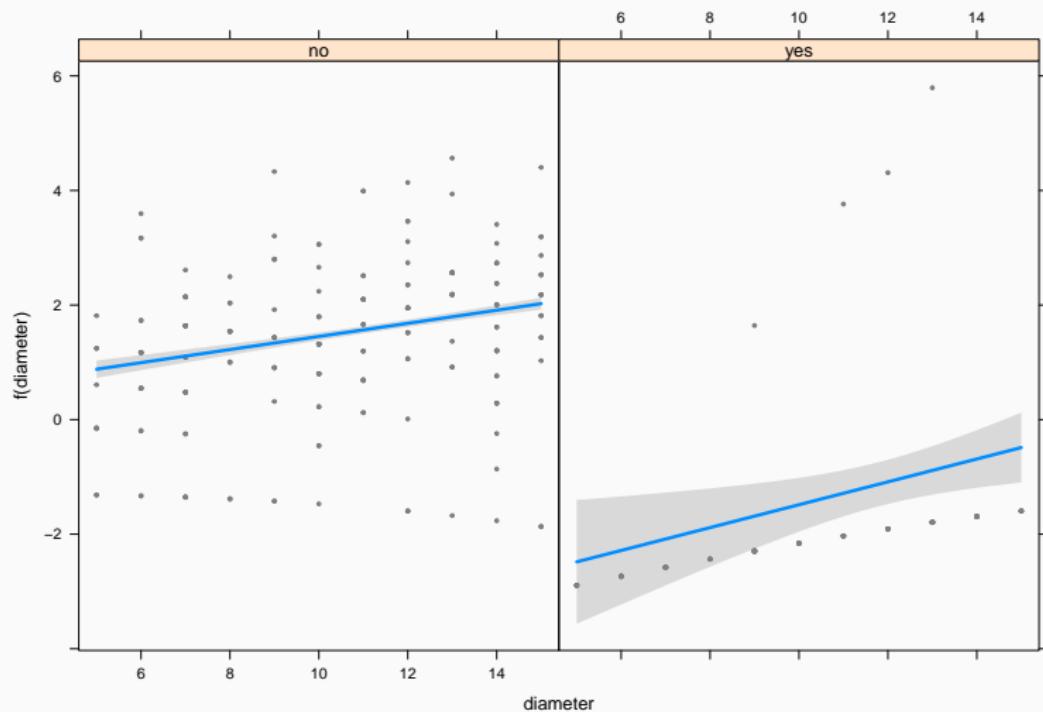
(Dispersion parameter for poisson family taken to be 1)

Null deviance: 1184.57 on 299 degrees of freedom

Residual deviance: 526.97 on 296 degrees of freedom

AIC: 1176.7

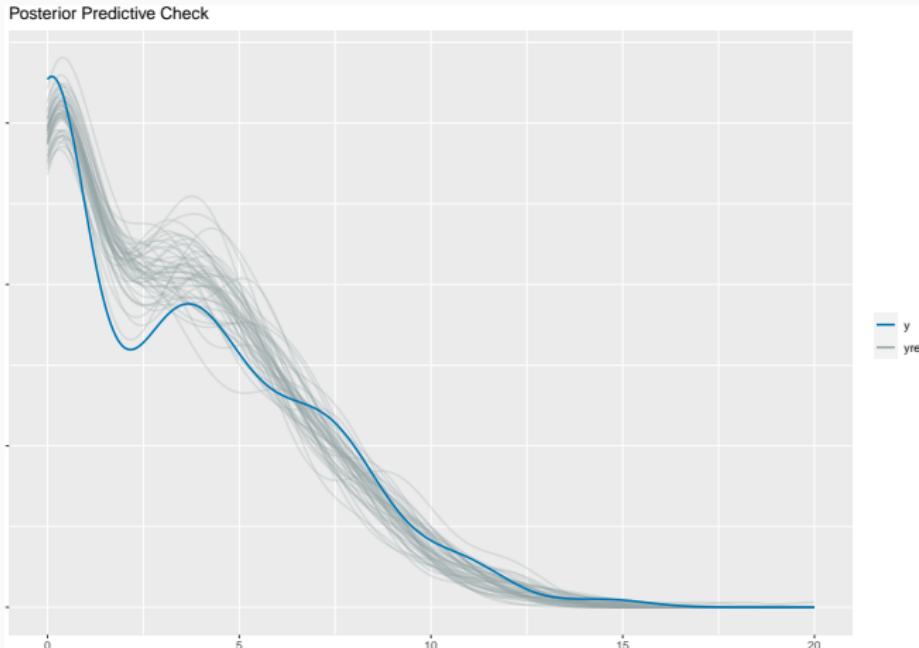
Visualising the fitted Poisson GLM



Checking Poisson GLM

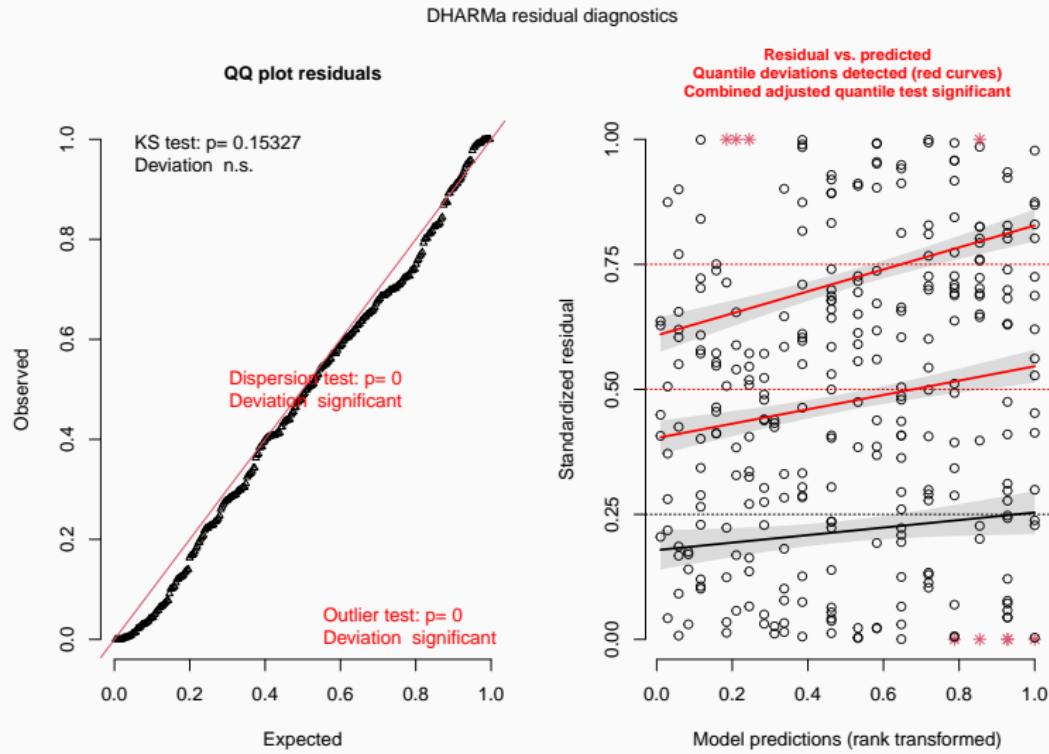
Simulate data from fitted model (y_{rep}) and compare with observed data (y)

```
library("performance")
pp_check(eggs.poi)
```



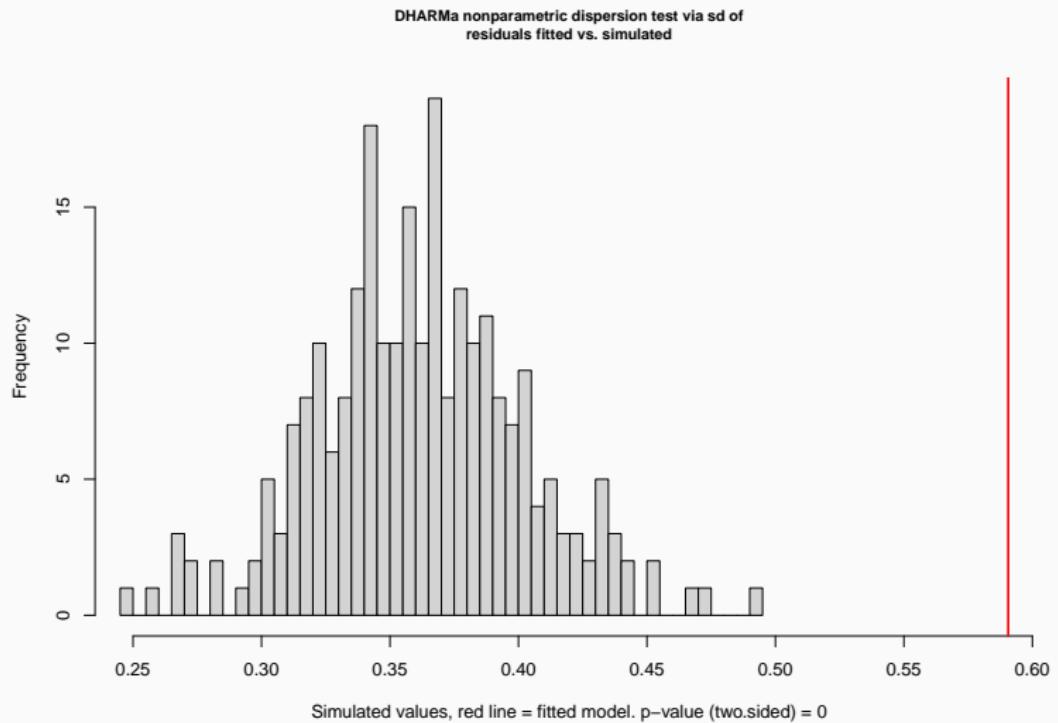
Checking Poisson GLM with DHARMA

```
library("DHARMA")
eggs.poi.res <- simulateResiduals(eggs.poi, plot = TRUE)
```



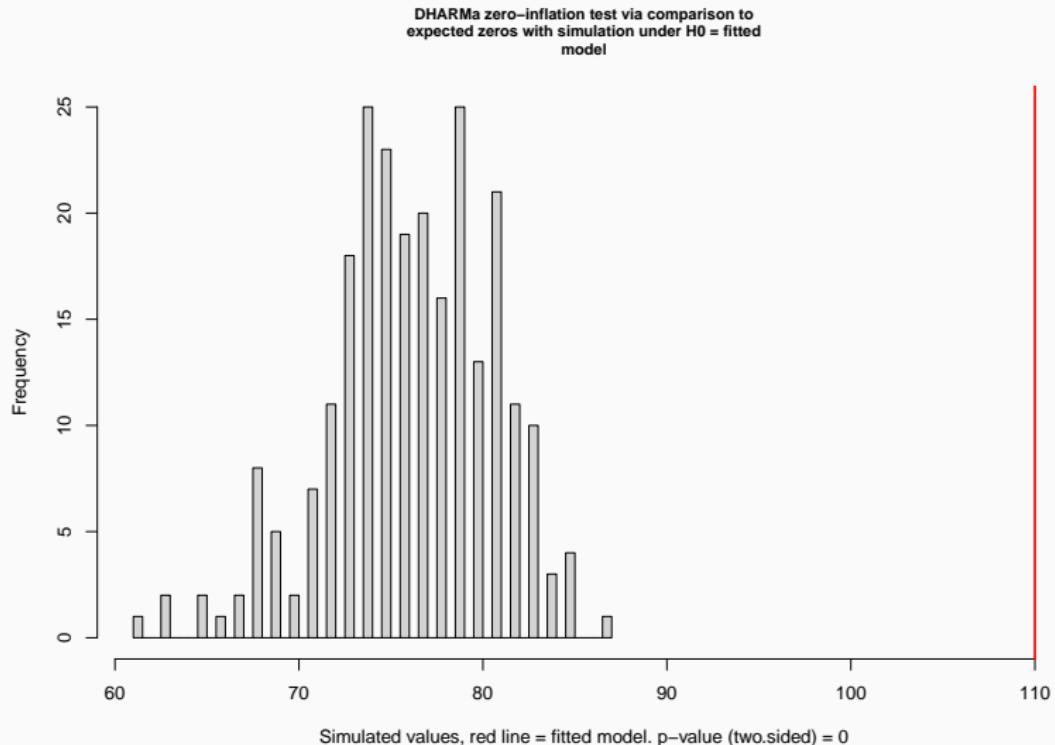
Checking overdispersion

```
testDispersion(eggs.poi.res)
```



Checking zero inflation

```
testZeroInflation(eggs.poi.res)
```



Accounting for zero-inflation

Zero-inflated Poisson/Negative Binomial

Mixture model:

1. Model probability of 0 (Binomial)

Zero-inflated Poisson/Negative Binomial

Mixture model:

1. Model probability of 0 (Binomial)
2. Model counts (including 0) (Poisson/Negative Binomial)

Modelling egg number as Zero-Inflated Poisson (ZIP)

Nests may be occupied or not:

$\text{Probability nest not occupied} \sim \text{old}$ (Binomial)

For occupied nests:

$\text{Number of eggs} \sim \text{Nest diameter}$ (Poisson)

```
library("glmmTMB")
eggs.zip <- glmmTMB(n.eggs ~ diameter,
                      family = "poisson",
                      ziformula = ~ old,
                      data = eggs)
```

Modelling egg number as Zero-Inflated Poisson

```
Family: poisson ( log )
Formula:         n.eggs ~ diameter
Zero inflation: ~old
Data: eggs

AIC      BIC      logLik deviance df.resid
993.8    1008.6    -492.9     985.8      296
```

Conditional model:

| | Estimate | Std. Error | z value | Pr(> z) |
|-------------|----------|------------|---------|-------------|
| (Intercept) | 0.41622 | 0.13619 | 3.056 | 0.00224 ** |
| diameter | 0.11248 | 0.01155 | 9.737 | < 2e-16 *** |

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

Zero-inflation model:

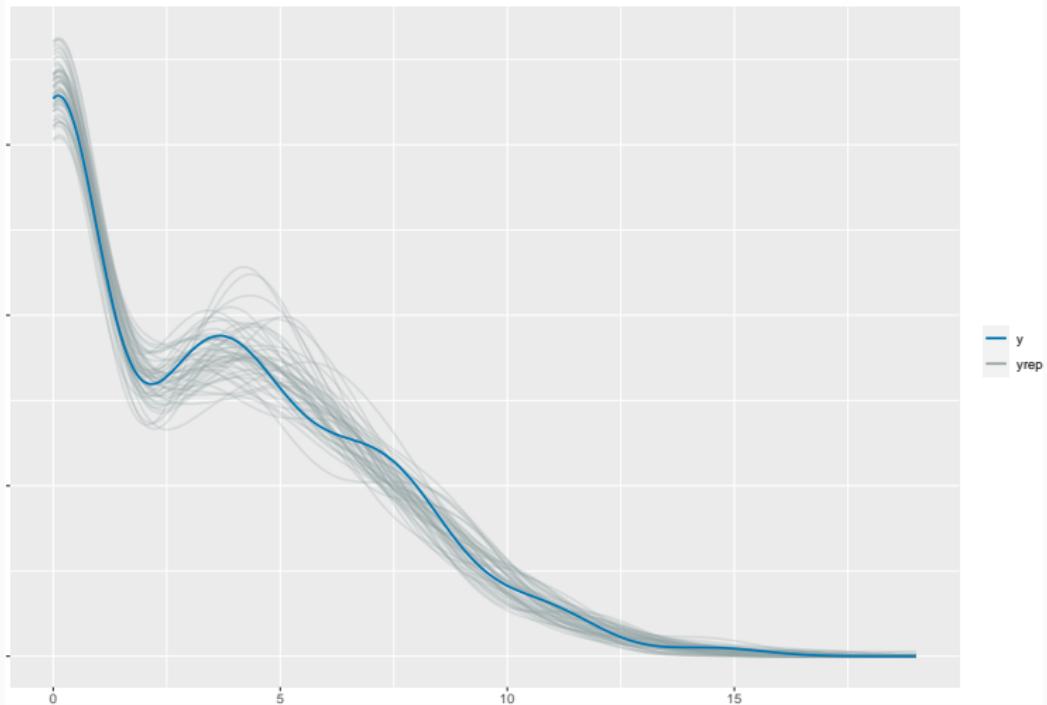
| | Estimate | Std. Error | z value | Pr(> z) |
|-------------|----------|------------|---------|------------|
| (Intercept) | -2.4054 | 0.2803 | -8.582 | <2e-16 *** |
| oldyes | 5.4897 | 0.5830 | 9.416 | <2e-16 *** |

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

Checking ZIP model

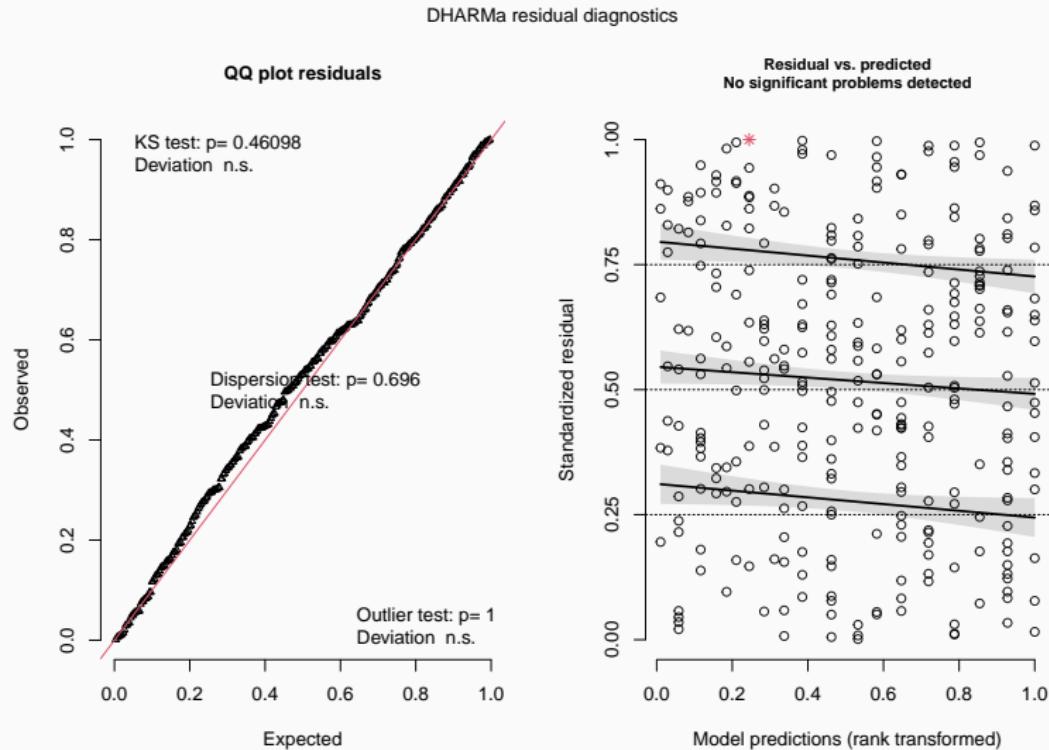
```
pp_check(eggs.zip)
```

Posterior Predictive Check



Checking ZIP model with DHARMA

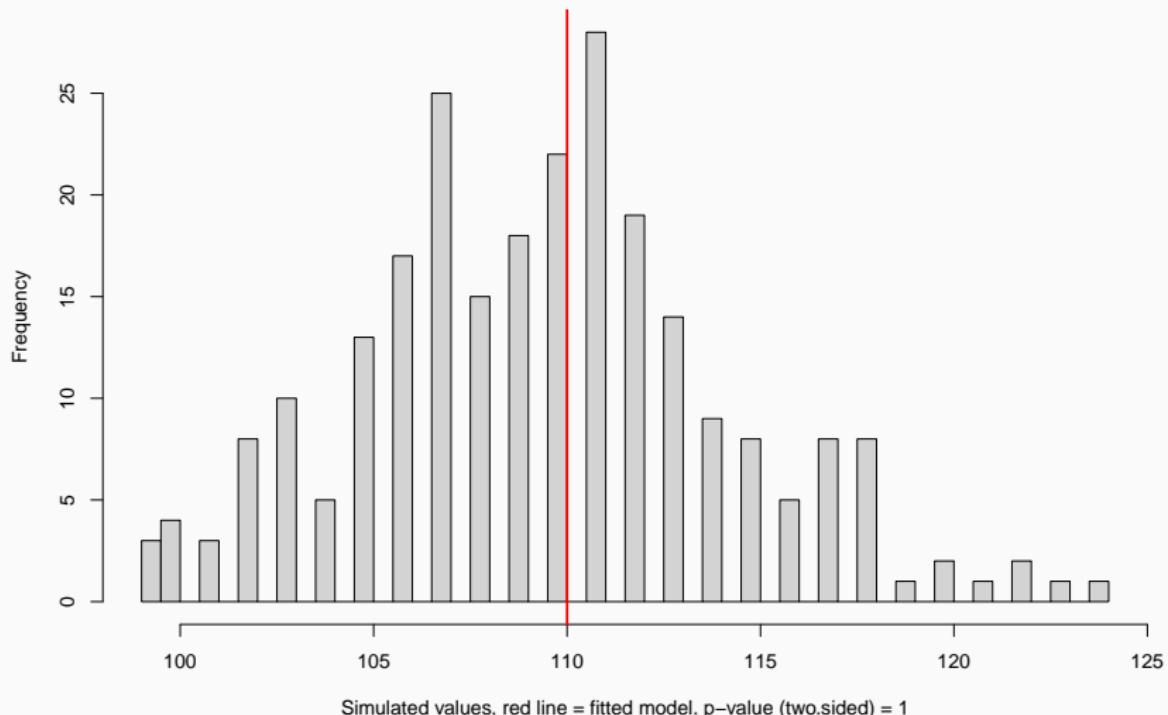
```
eggs.zip.res <- simulateResiduals(eggs.zip, plot = TRUE)
```



Checking ZIP model with DHARMA

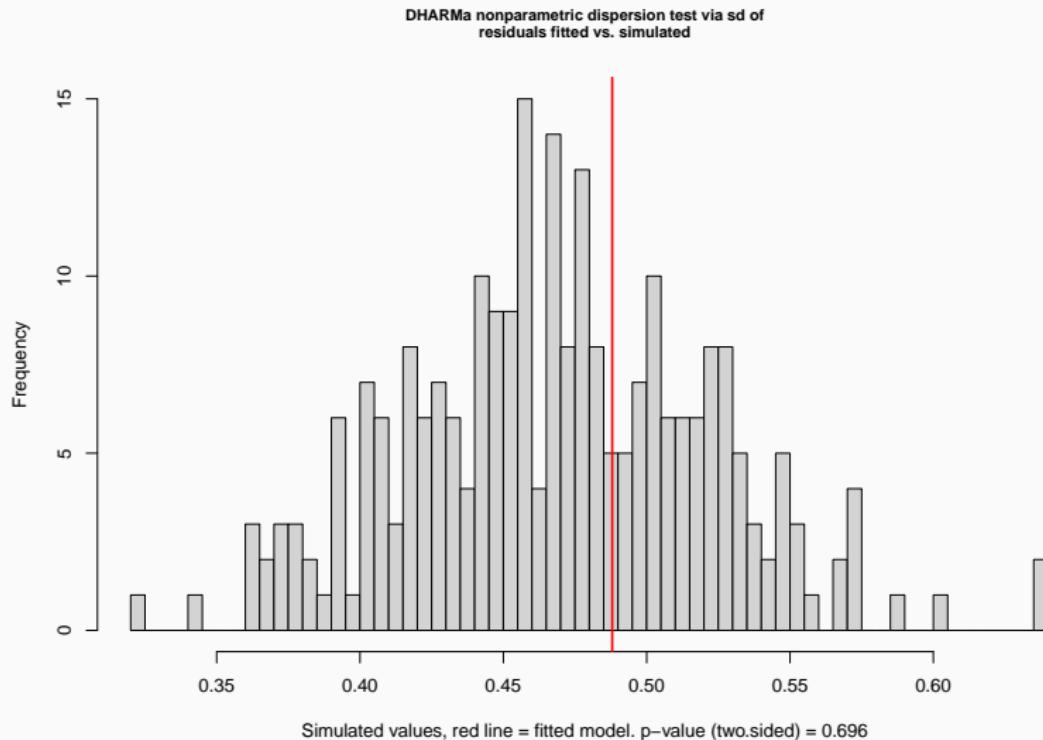
```
testZeroInflation(eggs.zip.res)
```

DHARMA zero-inflation test via comparison to
expected zeros with simulation under H_0 = fitted
model



Checking ZIP model with DHARMA

```
testDispersion(eggs.zip.res)
```



Modelling egg number as Zero-Inflated Negative Binomial (ZINB)

(If there were overdispersion with Poisson)

```
eggs.zinb <- glmmTMB(n.eggs ~ diameter,  
                      family = "nbinom2",  
                      ziformula = ~ old,  
                      data = eggs)
```

Modelling egg number as ZINB

```
Family: nbinom2 ( log )
Formula: n.eggs ~ diameter
Zero inflation: ~old
Data: eggs

      AIC      BIC      logLik deviance df.resid
995.7  1014.2   -492.8    985.7      295
```

Dispersion parameter for nbinom2 family (): 143

Conditional model:

| | Estimate | Std. Error | z value | Pr(> z) |
|-------------|----------|------------|---------|-------------|
| (Intercept) | 0.4118 | 0.1389 | 2.964 | 0.00304 ** |
| diameter | 0.1128 | 0.0118 | 9.561 | < 2e-16 *** |

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

Zero-inflation model:

| | Estimate | Std. Error | z value | Pr(> z) |
|-------------|----------|------------|---------|------------|
| (Intercept) | -2.4160 | 0.2846 | -8.489 | <2e-16 *** |
| oldyes | 5.4995 | 0.5850 | 9.401 | <2e-16 *** |

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

Comparing models

```
library("parameters")
compare_models(eggs.poi, eggs.zip, eggs.zinb)
```

| Parameter | eggs.poi | eggs.zip | eggs.zinb |
|----------------------|----------------------|-------------------|-------------------|
| <hr/> | | | |
| (Intercept) | 0.31 (0.06, 0.56) | 0.42 (0.15, 0.68) | 0.41 (0.14, 0.68) |
| diameter | 0.11 (0.09, 0.14) | 0.11 (0.09, 0.14) | 0.11 (0.09, 0.14) |
| old (yes) | -3.79 (-5.60, -1.98) | | |
| old (yes) * diameter | 0.09 (-0.06, 0.23) | | |
| <hr/> | | | |
| Observations | 300 | 300 | 300 |

Comparing models

```
library("performance")
compare_performance(eggs.poi, eggs.zip, eggs.zinb)
```

Comparison of Model Performance Indices

| Name | Model | AIC | BIC | RMSE | Sigma | Score_log | Score_sph |
|-----------|---------|----------|----------|-------|---------|-----------|-----------|
| <hr/> | | | | | | | |
| eggs.poi | glm | 1176.701 | 1191.516 | 2.324 | 1.334 | -1.948 | |
| eggs.zip | glmmTMB | 993.790 | 1008.605 | 2.324 | 1.000 | -1.643 | |
| eggs.zinb | glmmTMB | 995.666 | 1014.185 | 2.324 | 143.279 | | |

Accounting for zero-inflation with hurdle models

Tracking measles outbreak

Counting number of hives/person

Many people not sick (0 hives)

Those sick, have many hives (>1)



ZIP/ZINB vs Hurdle models

ZIP/ZINB:

1. Binomial model: probability of zero

Hurdle:

ZIP/ZINB vs Hurdle models

ZIP/ZINB:

1. Binomial model: probability of zero
2. Count model (Poisson/NegBin) includes zero

Hurdle:

ZIP/ZINB vs Hurdle models

ZIP/ZINB:

1. Binomial model: probability of zero
2. Count model (Poisson/NegBin) includes zero

Hurdle:

1. Binomial model: probability of non-zero

ZIP/ZINB vs Hurdle models

ZIP/ZINB:

1. Binomial model: probability of zero
2. Count model (Poisson/NegBin) includes zero

Hurdle:

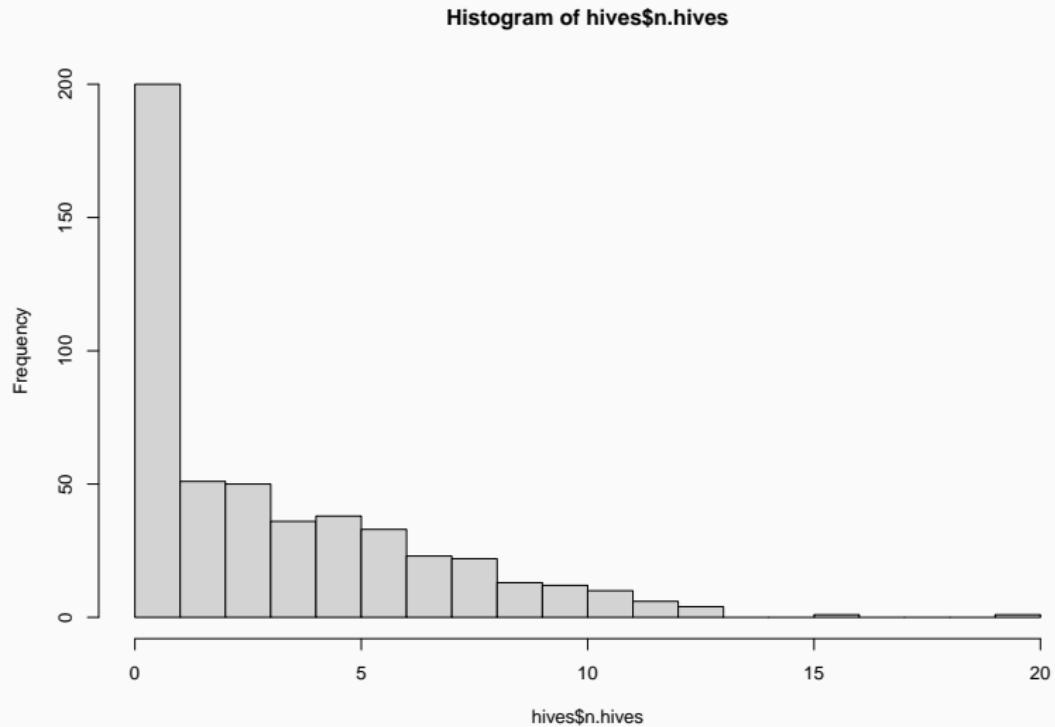
1. Binomial model: probability of non-zero
2. Count model truncated at 1

How many hives per skin area?

```
hives <- read.csv("data/hives.csv")
```

| age | vaccinated | area.cm2 | n.hives |
|--------------|----------------|----------------|----------------|
| Min. : 1.0 | Min. : 0.000 | Min. : 5.000 | Min. : 0.000 |
| 1st Qu.:23.0 | 1st Qu.: 0.000 | 1st Qu.: 6.000 | 1st Qu.: 0.000 |
| Median :45.0 | Median : 1.000 | Median : 8.000 | Median : 2.000 |
| Mean : 44.7 | Mean : 0.648 | Mean : 7.482 | Mean : 3.256 |
| 3rd Qu.:65.0 | 3rd Qu.: 1.000 | 3rd Qu.: 9.000 | 3rd Qu.: 5.250 |
| Max. :90.0 | Max. : 1.000 | Max. :10.000 | Max. :20.000 |

Many people with 0 hives

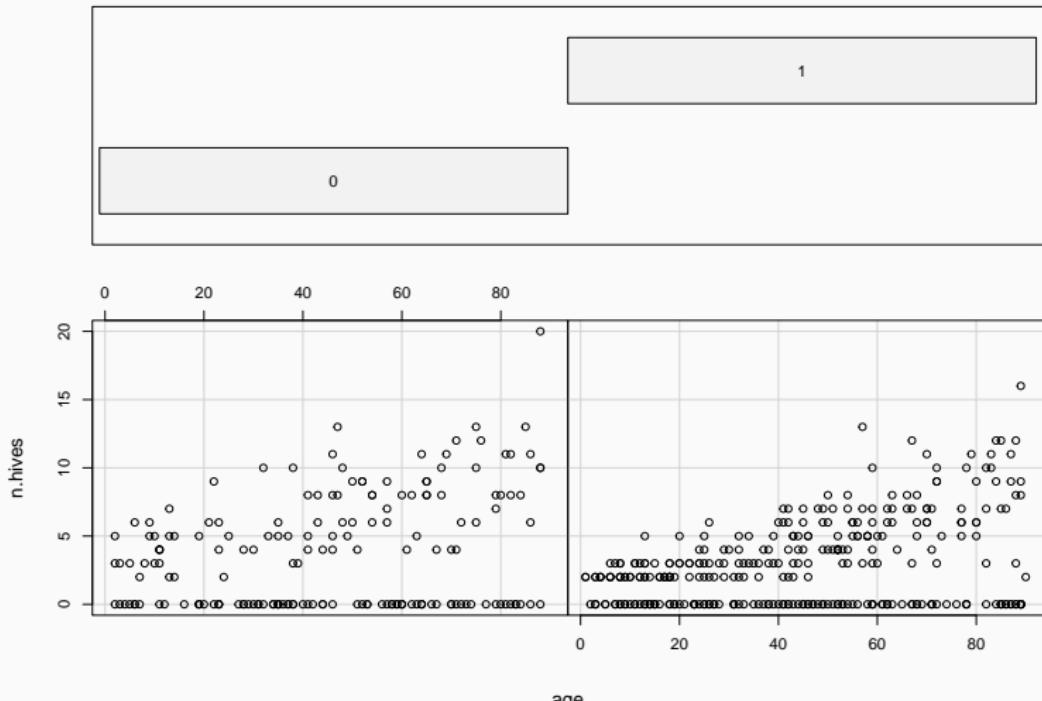


(that does not mean we need zero-inflated model!)

Number of hives ~ age * vaccinated

```
coplot(n.hives ~ age | as.factor(vaccinated), data = hives)
```

Given : as.factor(vaccinated)



Trying Poisson GLM

```
hives.poi <- glm(n.hives ~ vaccinated * age,  
                  offset = log(area.cm2),  
                  data = hives,  
                  family = poisson)
```

Trying Poisson GLM

Call:

```
glm(formula = n.hives ~ vaccinated * age, family = poisson, data = hives,  
    offset = log(area.cm2))
```

Deviance Residuals:

| Min | 1Q | Median | 3Q | Max |
|---------|---------|--------|--------|--------|
| -4.0081 | -2.2235 | 0.1396 | 1.2155 | 4.2198 |

Coefficients:

| | Estimate | Std. Error | z value | Pr(> z) |
|----------------|-----------|------------|---------|-------------|
| (Intercept) | -1.363696 | 0.095097 | -14.340 | < 2e-16 *** |
| vaccinated | -0.334184 | 0.122887 | -2.719 | 0.00654 ** |
| age | 0.013626 | 0.001623 | 8.395 | < 2e-16 *** |
| vaccinated:age | 0.002034 | 0.002075 | 0.980 | 0.32708 |

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

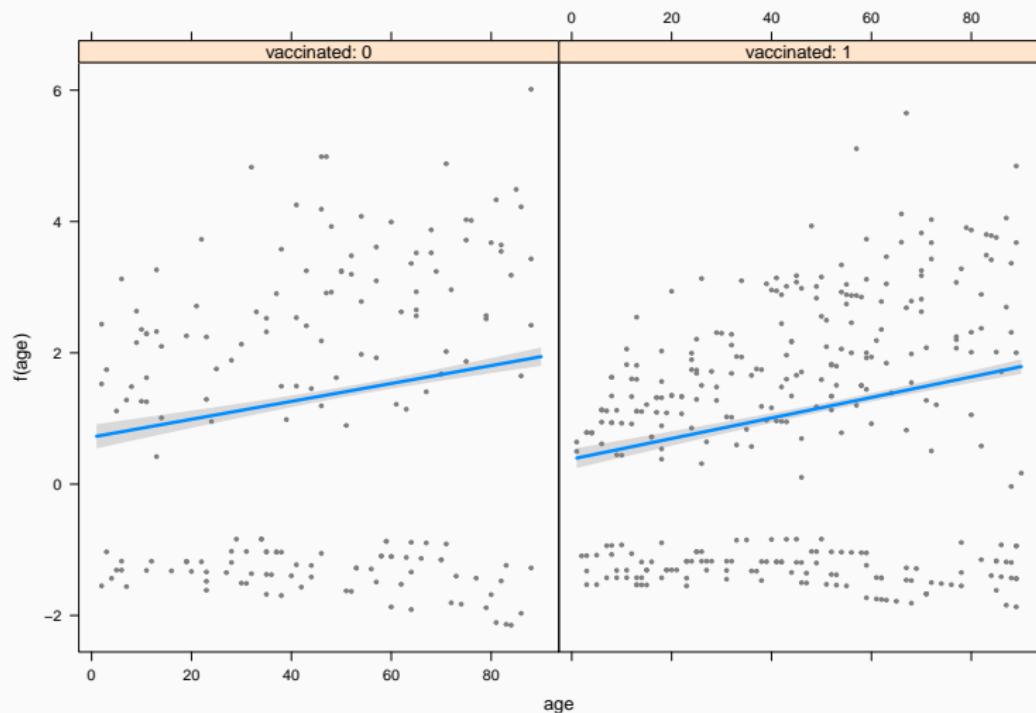
(Dispersion parameter for poisson family taken to be 1)

Null deviance: 2137.0 on 499 degrees of freedom

Residual deviance: 1891.7 on 496 degrees of freedom

AIC: 2925.6

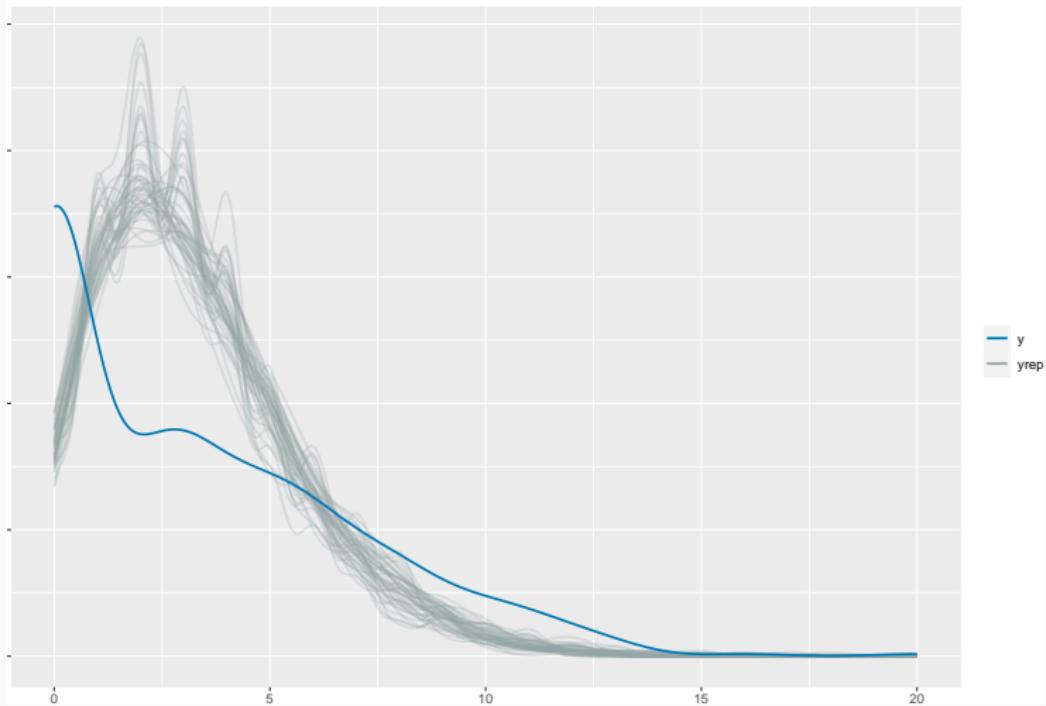
Visualising fitted Poisson GLM



Checking Poisson GLM

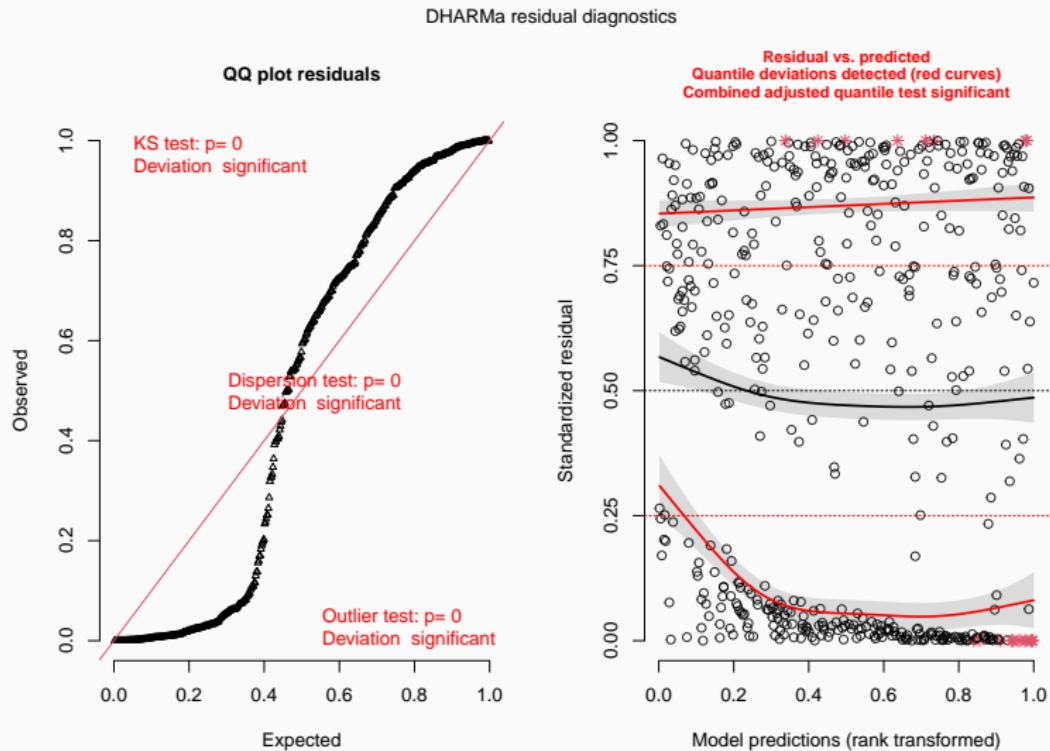
```
pp_check(hives.poi)
```

Posterior Predictive Check



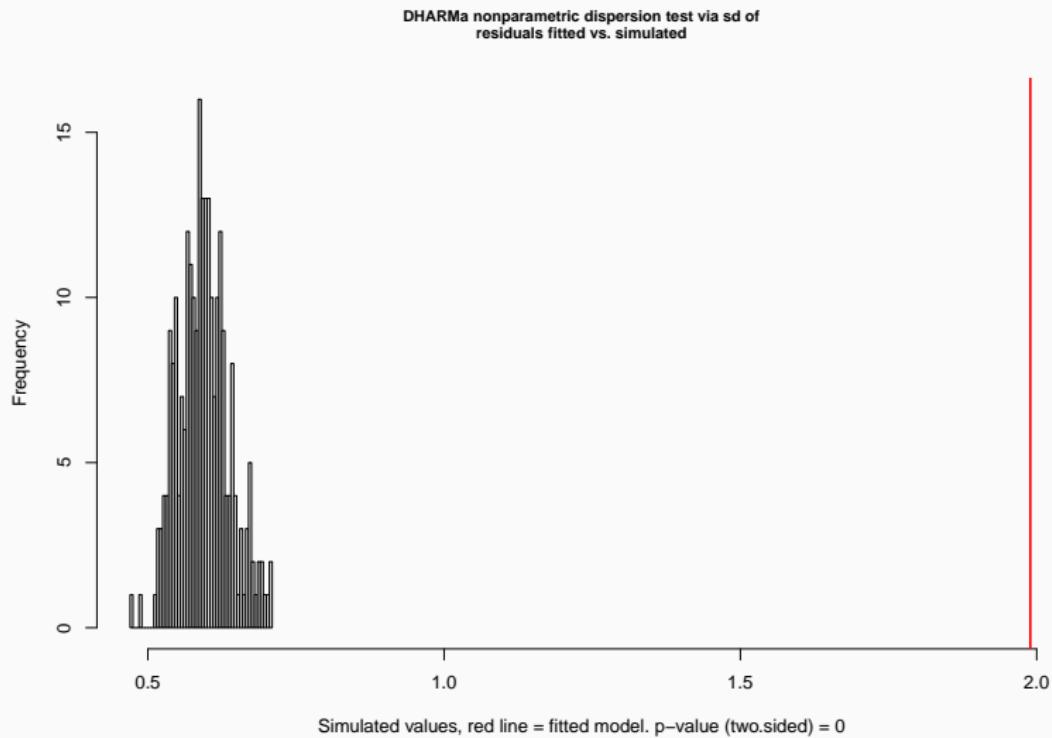
Checking Poisson GLM

```
hives.poi.res <- simulateResiduals(hives.poi, plot = TRUE)
```



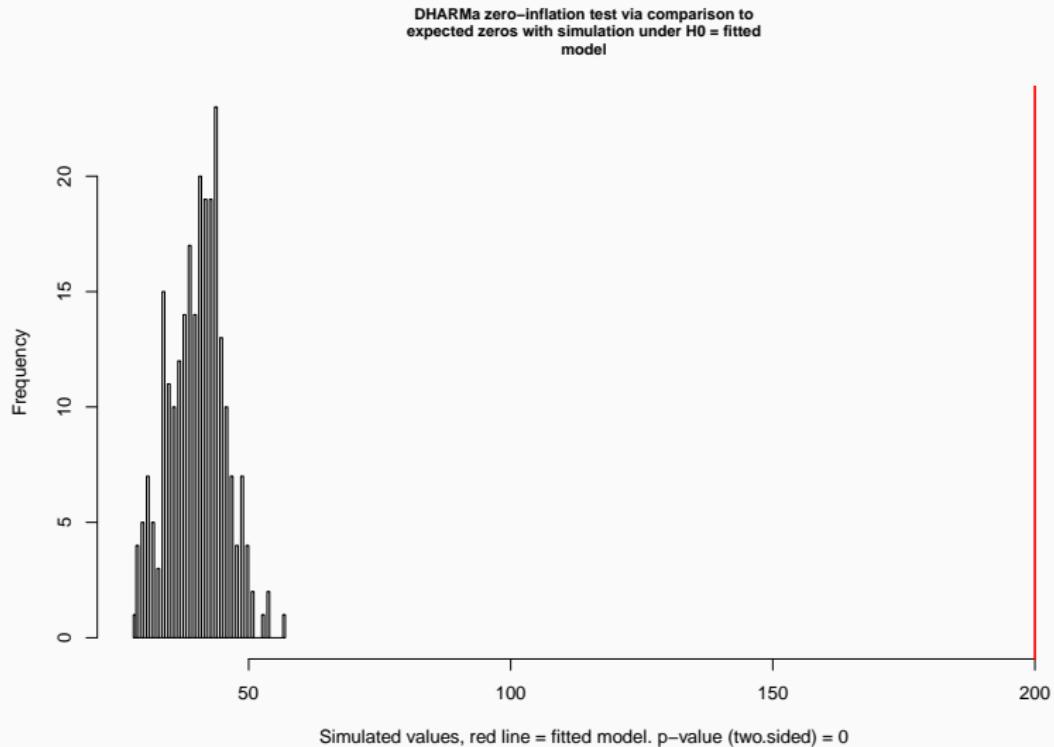
Checking overdispersion

```
testDispersion(hives.poi.res)
```



Checking zero inflation

```
testZeroInflation(hives.poi.res)
```



Accounting for zero-inflation with hurdle model

```
hives.hur <- glmmTMB(n.hives ~ vaccinated + age,  
                      family = truncated_poisson,  
                      ziformula = ~ 1,  
                      offset = log(area.cm2),  
                      data = hives)
```

Accounting for zero-inflation with hurdle model

```
Family: truncated_poisson ( log )
Formula: n.hives ~ vaccinated + age
Zero inflation: ~1
Data: hives
Offset: log(area.cm2)
```

| AIC | BIC | logLik | deviance | df.resid |
|--------|--------|--------|----------|----------|
| 1932.1 | 1949.0 | -962.1 | 1924.1 | 496 |

Conditional model:

| | Estimate | Std. Error | z value | Pr(> z) | | | | | | | |
|----------------|-----------|------------|---------|--------------|------|-----|------|-----|-----|-----|---|
| (Intercept) | -0.853885 | 0.070755 | -12.068 | < 2e-16 *** | | | | | | | |
| vaccinated | -0.365664 | 0.051532 | -7.096 | 1.29e-12 *** | | | | | | | |
| age | 0.014860 | 0.001065 | 13.955 | < 2e-16 *** | | | | | | | |
| --- | | | | | | | | | | | |
| Signif. codes: | 0 | '***' | 0.001 | '**' | 0.01 | '*' | 0.05 | '.' | 0.1 | ' ' | 1 |

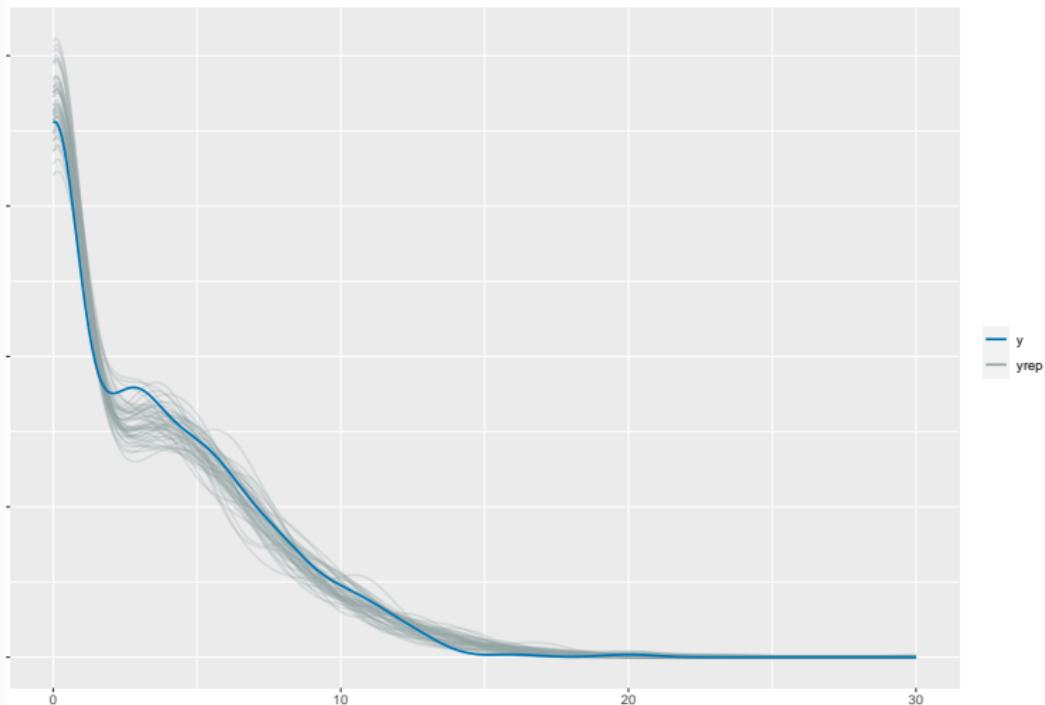
Zero-inflation model:

| | Estimate | Std. Error | z value | Pr(> z) | | | | | | | |
|----------------|----------|------------|---------|--------------|------|-----|------|-----|-----|-----|---|
| (Intercept) | -0.40547 | 0.09129 | -4.442 | 8.93e-06 *** | | | | | | | |
| --- | | | | | | | | | | | |
| Signif. codes: | 0 | '***' | 0.001 | '**' | 0.01 | '*' | 0.05 | '.' | 0.1 | ' ' | 1 |

Checking hurdle model

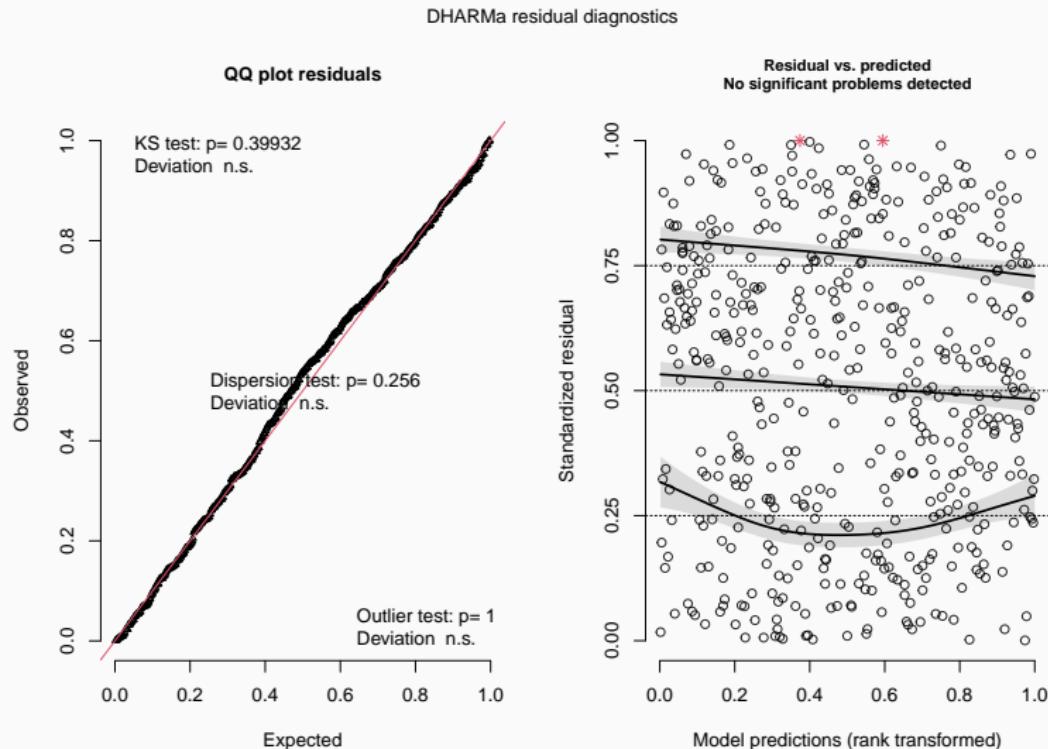
```
pp_check(hives.hur)
```

Posterior Predictive Check



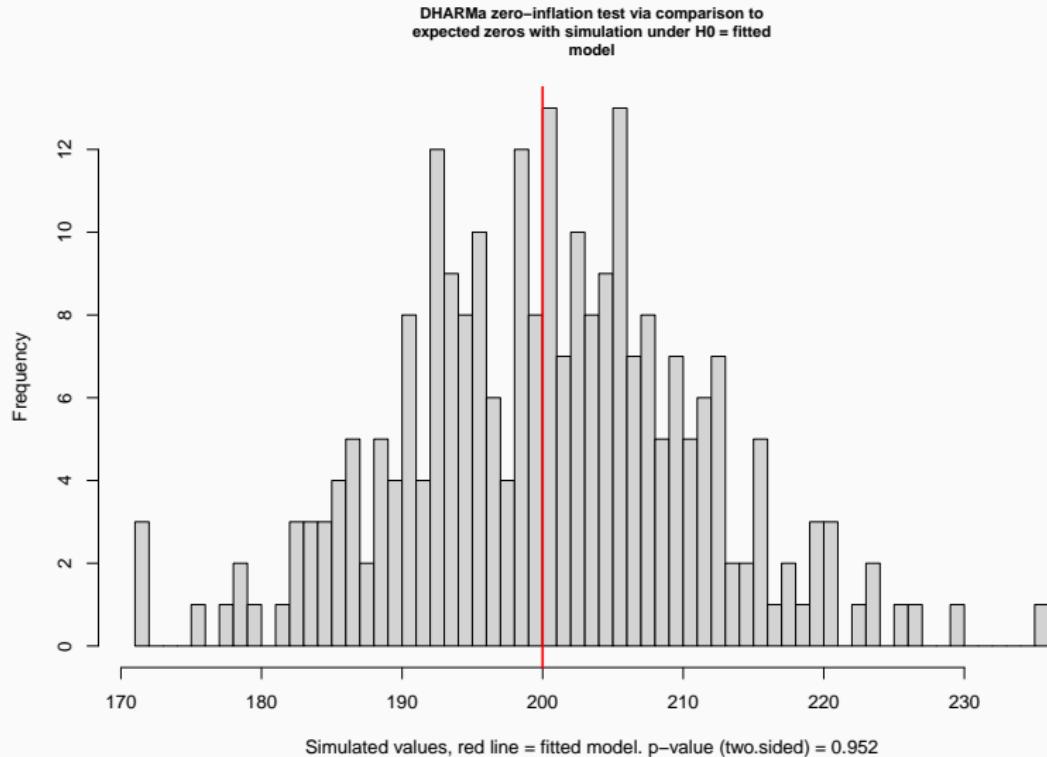
Checking hurdle model with DHARMA

```
hives.hur.res <- simulateResiduals(hives.hur, plot = TRUE)
```



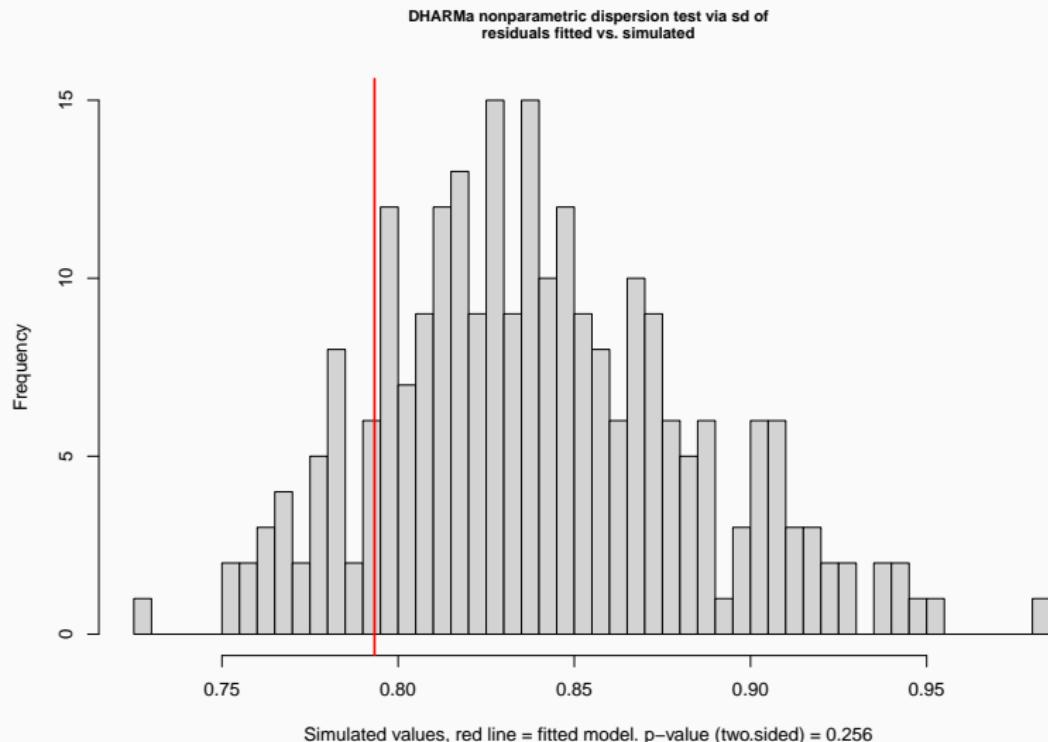
Checking zero inflation

```
testZeroInflation(hives.hur.res)
```



Checking overdispersion

```
testDispersion(hives.hur.res)
```



Comparing models

```
compare_models(hives.poi, hives.hur)
```

| Parameter | | hives.poi | | hives.hur |
|------------------|--|------------------------|--|----------------------|
| ----- | | | | |
| (Intercept) | | -1.36 (-1.55, -1.18) | | -0.85 (-0.99, -0.72) |
| vaccinated | | -0.33 (-0.58, -0.09) | | -0.37 (-0.47, -0.26) |
| age | | 0.01 (0.01, 0.02) | | 0.01 (0.01, 0.02) |
| vaccinated * age | | 2.03e-03 (0.00, 0.01) | | |
| ----- | | | | |
| Observations | | 500 | | 500 |

Comparing models

```
compare_performance(hives.poi, hives.hur)
```

```
# Comparison of Model Performance Indices
```

| Name | Model | AIC | BIC | RMSE | Sigma | Score_log | Score_spher |
|-----------|---------|----------|----------|-------|-------|-----------|-------------|
| <hr/> | | | | | | | |
| hives.poi | glm | 2925.603 | 2942.462 | 3.299 | 1.953 | -2.918 | 0 |
| hives.hur | glmmTMB | 1932.124 | 1948.982 | 3.313 | 1.000 | -2.262 | 0 |

Mixed / Multilevel Models

Francisco Rodríguez-Sánchez

<https://frodriguezsanchez.net>

Example dataset: trees

- Data on 1000 trees from 10 sites.

```
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 |

Example dataset: trees

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

```
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 |

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 |
|----------|---------|--------|--------|---------|
| -13.3270 | -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 |

Remember our model structure

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

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

In this case:

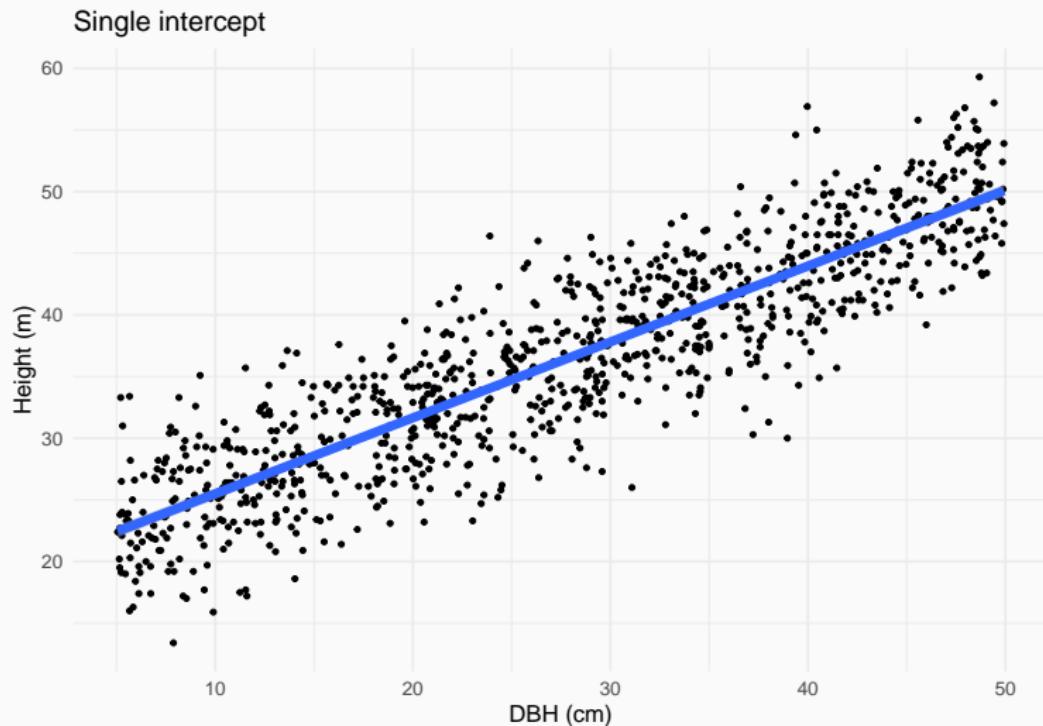
$$Height_i \sim N(\mu_i, \sigma^2)$$

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

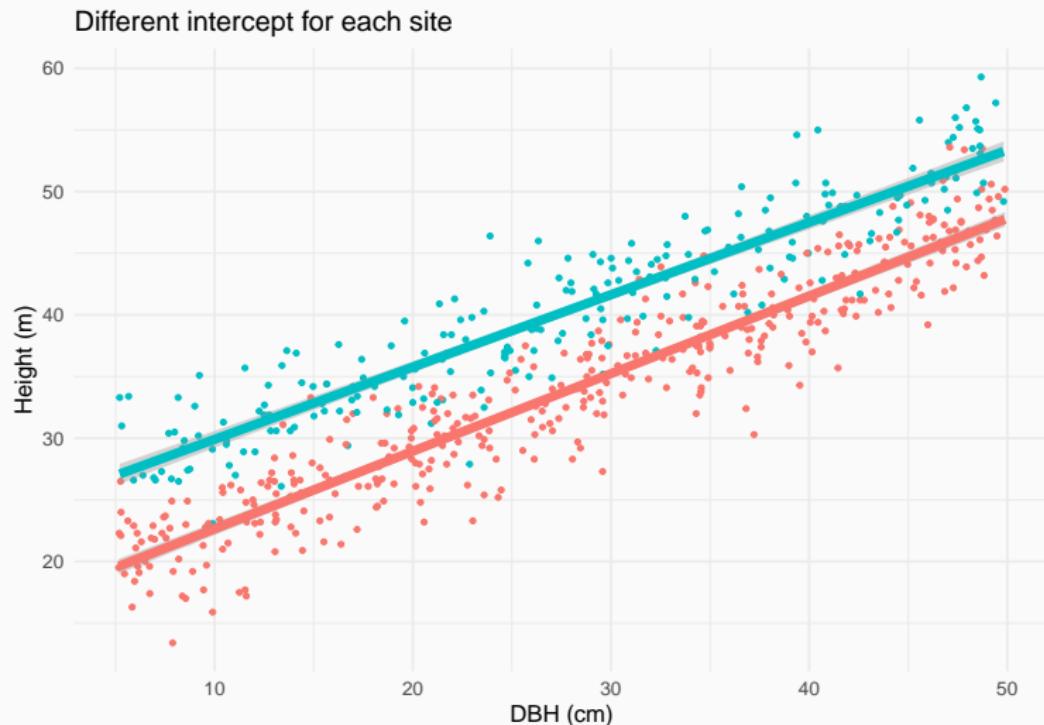
α : expected height when DBH = 0

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

There is only one intercept



What if allometry varies among sites?



Fitting a varying intercepts model with lm

Call:

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

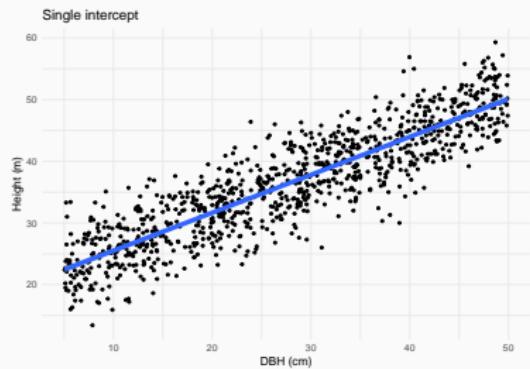
Residuals:

| Min | 1Q | Median | 3Q | Max |
|----------|---------|--------|--------|---------|
| -10.1130 | -1.9885 | 0.0582 | 2.0314 | 11.3320 |

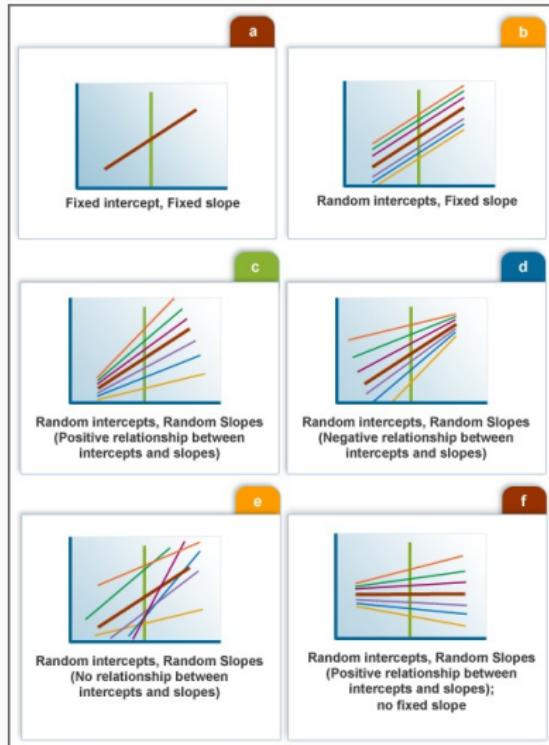
Coefficients:

| | 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 *** |
| --- | | | | |

Single vs varying intercept



Mixed models enable us to account for variability



Mixed model with varying intercepts

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

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

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

In our example:

$$Height_i = a + site_j + b \cdot DBH_i + \varepsilon_i$$

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

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

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 site.

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)`

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.

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.

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?

- Varies by group

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

What is a random effect, really?

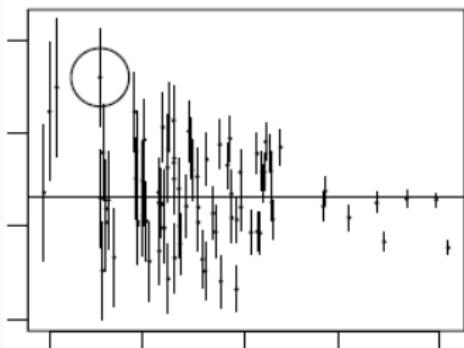
- Varies by group
- Variation estimated with **probability model**

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

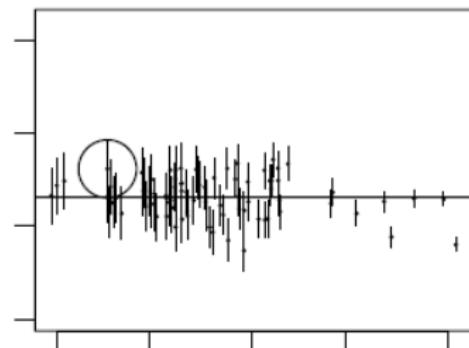
Shrinkage improves parameter estimation

Especially for groups with low sample size

No pooling



Multilevel model



From Gelman & Hill p. 253

Fitting mixed/multilevel models

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

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)

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"
```

Broom: model estimates in tidy form

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

```
# A tibble: 4 x 6  
  effect  group    term          estimate std.error statistic  
  <chr>   <chr>    <chr>        <dbl>     <dbl>      <dbl>  
1 fixed    <NA>    (Intercept)    19.0      1.10      17.3  
2 fixed    <NA>      dbh        0.617     0.00757    81.5  
3 ran_pars site    sd__(Intercept)  3.35      NA        NA  
4 ran_pars Residual sd__Observation 3.04      NA        NA
```

See also [broom.mixed](#)

Visualising model: `allEffects`

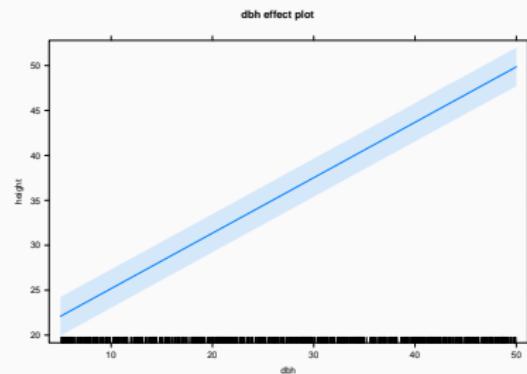
```
allEffects(mixed)
```

```
model: height ~ dbh
```

dbh effect

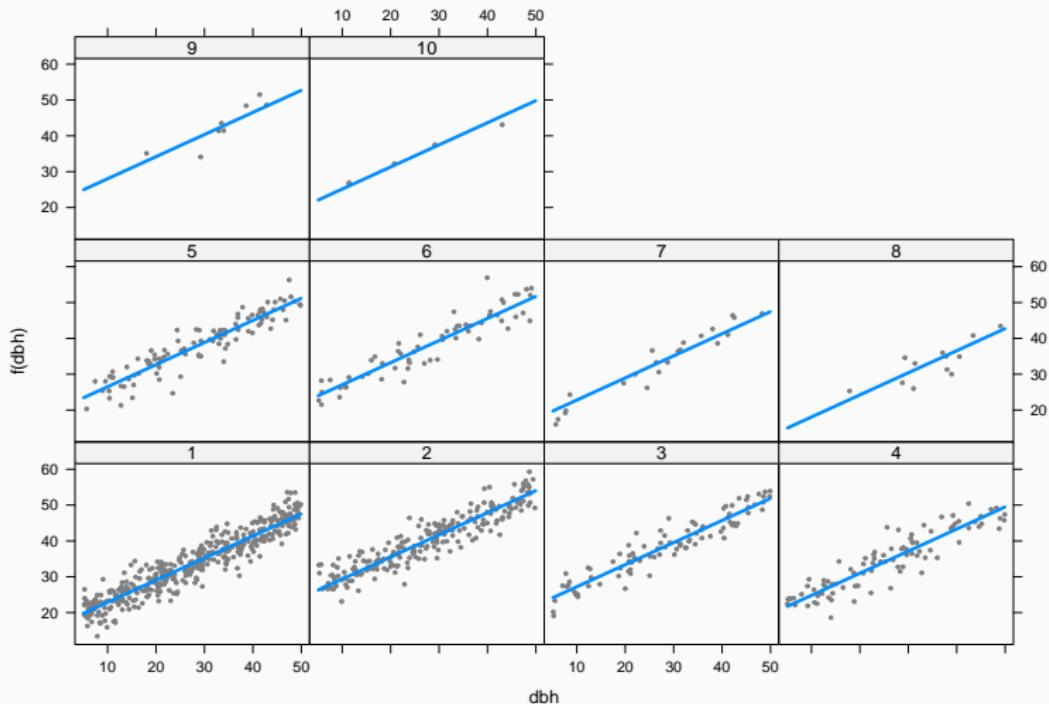
dbh

| | | | | |
|----------|----------|----------|----------|----------|
| 5 | 20 | 30 | 40 | 50 |
| 22.09610 | 31.35001 | 37.51928 | 43.68855 | 49.85782 |



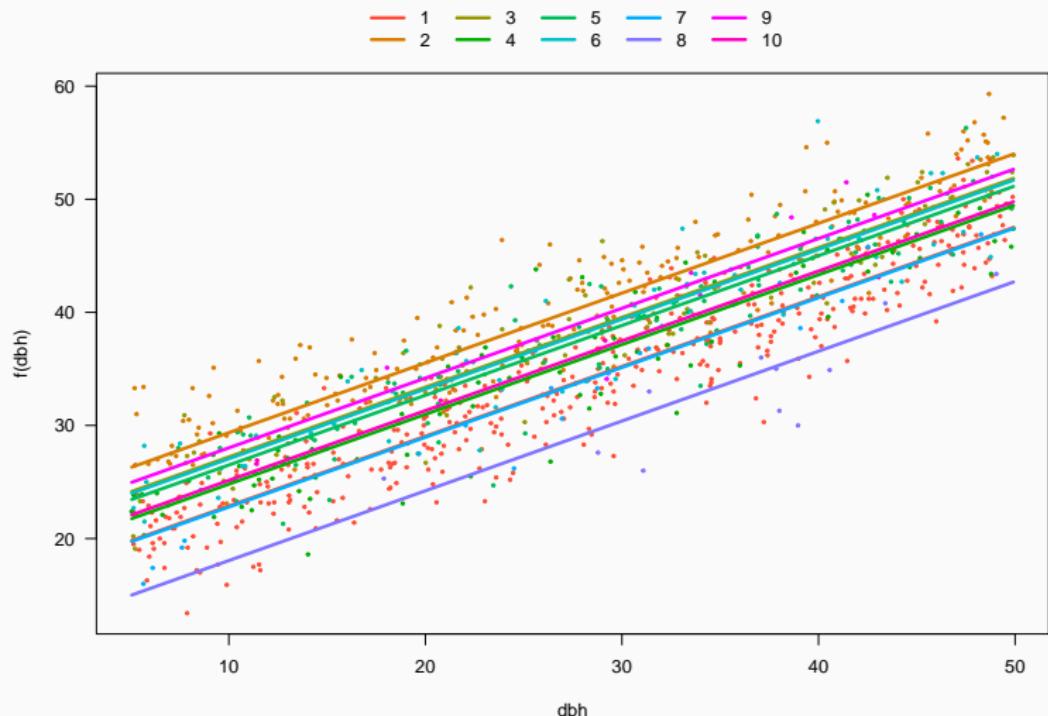
Visualising model: visreg

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



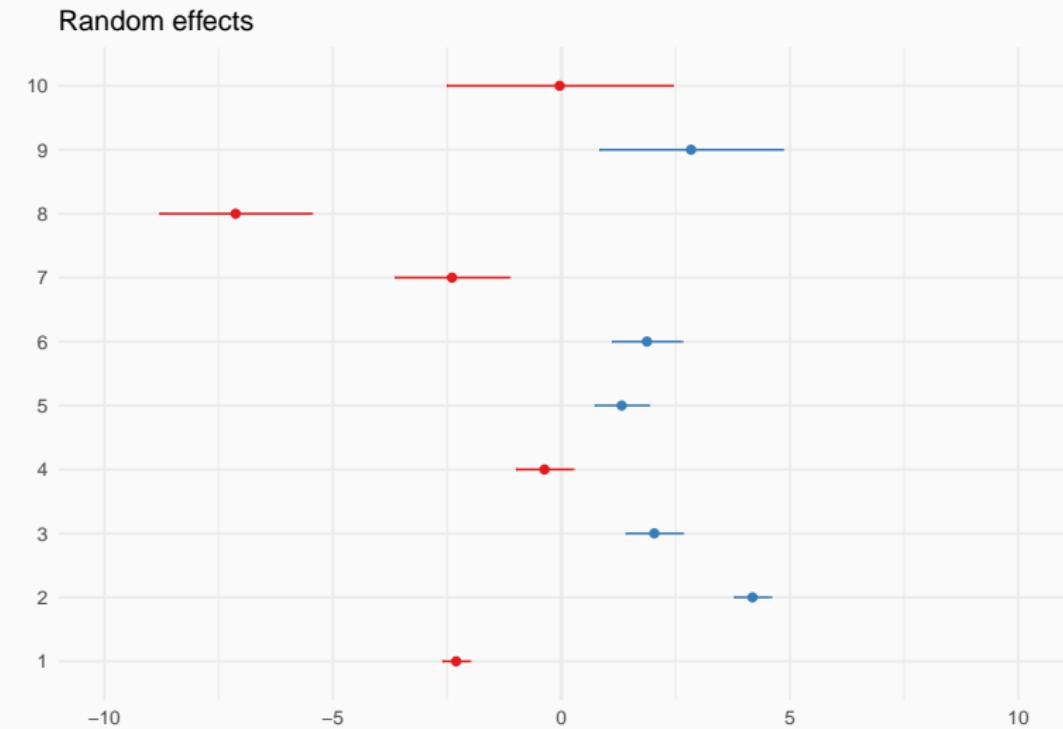
Visualising model

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



Visualising model: sjPlot

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

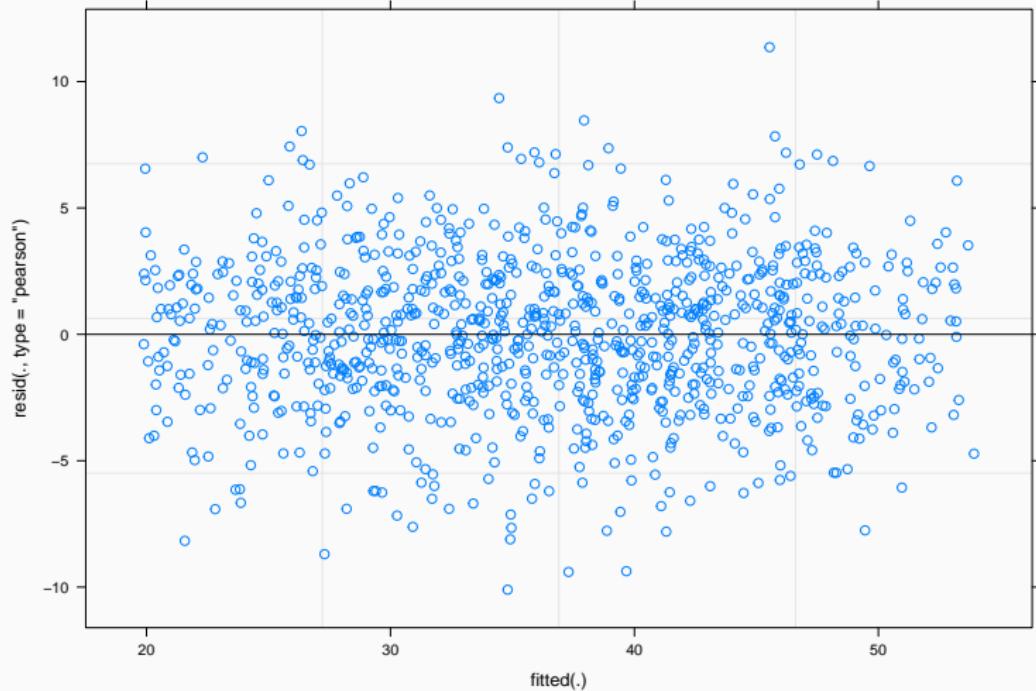


Using merTools to understand fitted model

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

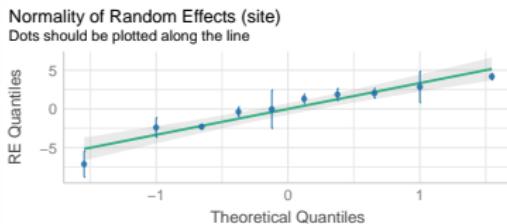
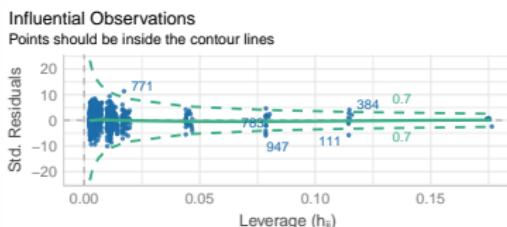
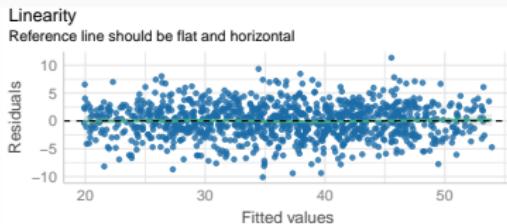
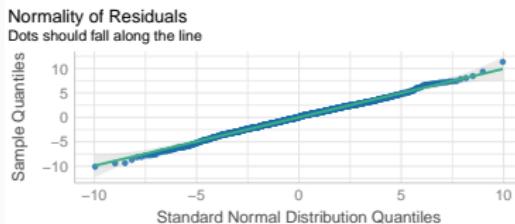
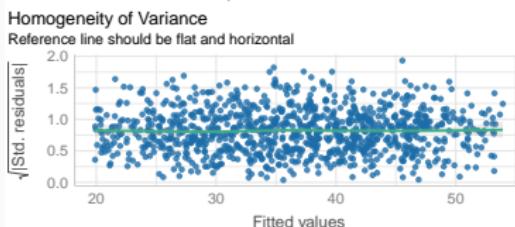
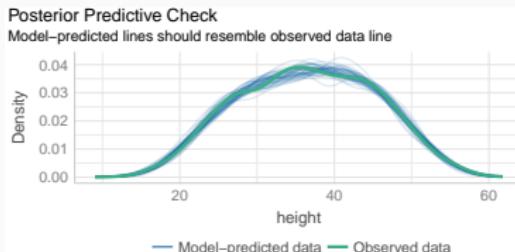
Checking residuals

```
plot(mixed)
```



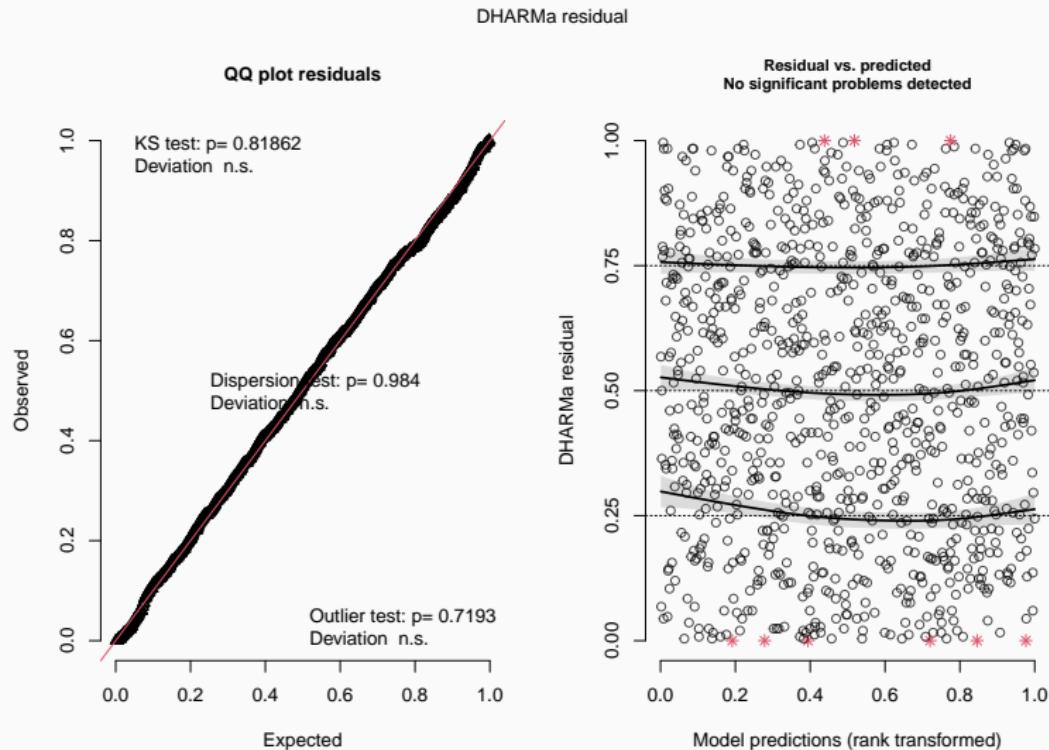
Checking residuals

```
library("performance")
check_model(mixed)
```



Checking residuals (DHARMA)

```
DHARMA::simulateResiduals(mixed, plot = TRUE, re.form = NULL)
```

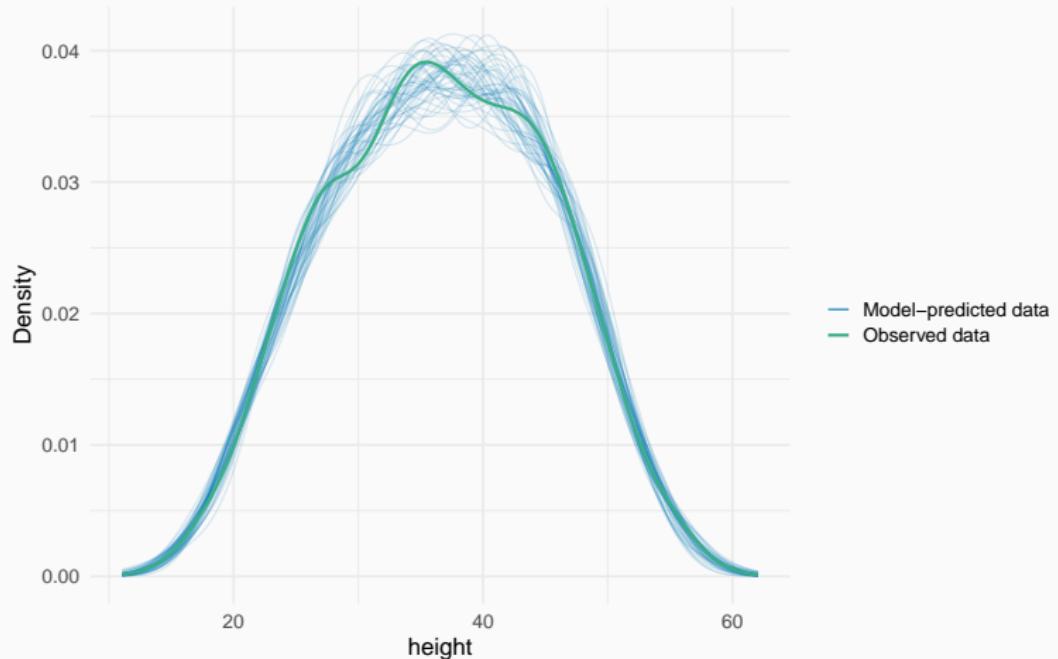


Model checking with simulated data

```
check_predictions(mixed)
```

Posterior Predictive Check

Model-predicted lines should resemble observed data line



R-squared for GLMMs

Many approaches! Somewhat polemic (e.g. see [this](#)).

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

```
r2(mixed)
```

```
# R2 for Mixed Models
```

Conditional R2: 0.888

Marginal R2: 0.753

Growing the hierarchy: adding site-level predictors

Model with group-level predictors

We had:

$$\begin{aligned}y_i &= a + \alpha_j + b \cdot x_i + \varepsilon_i \\ \alpha_j &\sim N(0, \tau^2) \\ \varepsilon_i &\sim N(0, \sigma^2)\end{aligned}$$

Now

$$\begin{aligned}y_i &= a + \alpha_j + b \cdot x_i + \varepsilon_i \\ \alpha_j &\sim N(\mu_j, \tau^2) \\ \mu_j &= \delta \cdot Predictor_j \\ \varepsilon_i &\sim N(0, \sigma^2)\end{aligned}$$

Are height differences among sites related to temperature?

$$Height_i = site_j + b \cdot DBH_i + \varepsilon_i$$

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

$$\mu_j = a + \delta \cdot Temperature_j$$

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

Are height differences among sites related to temperature?

```
sitedata <- read.csv("data/sitedata.csv")  
sitedata
```

| | site | temp |
|----|------|------|
| 1 | 1 | 15.1 |
| 2 | 2 | 22.0 |
| 3 | 3 | 20.1 |
| 4 | 4 | 20.4 |
| 5 | 5 | 20.0 |
| 6 | 6 | 20.1 |
| 7 | 7 | 17.5 |
| 8 | 8 | 14.6 |
| 9 | 9 | 19.2 |
| 10 | 10 | 16.0 |

Merging trees and site data

```
trees.full <- merge(trees, sitedata, by = "site")
head(trees.full)
```

| | site | dbh | height | sex | dead | temp |
|---|------|-------|--------|--------|------|------|
| 1 | 1 | 21.05 | 32.2 | male | 0 | 15.1 |
| 2 | 1 | 46.63 | 45.9 | female | 0 | 15.1 |
| 3 | 1 | 43.86 | 45.5 | male | 0 | 15.1 |
| 4 | 1 | 29.03 | 35.5 | male | 0 | 15.1 |
| 5 | 1 | 6.02 | 21.1 | male | 0 | 15.1 |
| 6 | 1 | 40.82 | 38.7 | male | 0 | 15.1 |

Fit multilevel model

```
group.pred <- lmer(height ~ dbh + (1 | site) + temp, data = trees.full)
```

Linear mixed model fit by REML ['lmerMod']

Formula: height ~ dbh + (1 | site) + temp

Data: trees.full

REML criterion at convergence: 5098.2

Scaled residuals:

| Min | 1Q | Median | 3Q | Max |
|---------|---------|--------|--------|--------|
| -3.3247 | -0.6517 | 0.0192 | 0.6663 | 3.7268 |

Random effects:

| Groups | Name | Variance | Std.Dev. |
|--------|-------------|----------|----------|
| site | (Intercept) | 3.158 | 1.777 |
| | Residual | 9.266 | 3.044 |

Number of obs: 1000, groups: site, 10

Fixed effects:

| | Estimate | Std. Error | t value |
|-------------|-----------|------------|---------|
| (Intercept) | -1.730910 | 4.671330 | -0.371 |
| dbh | 0.616894 | 0.007571 | 81.484 |
| temp | 1.115104 | 0.248000 | 4.496 |

Correlation of Fixed Effects:

| (Intr) | dbh |
|--------|-----|
|--------|-----|

Centre (and scale) continuous variables

```
mean(sitedata$temp)
```

```
[1] 18.5
```

```
trees.full$temp.c <- trees.full$temp - 18
```

Temperatures now referred as deviations from 18 °C (close to average)

Fit multilevel model

```
group.pred <- lmer(height ~ dbh + (1 | site) + temp.c, data = trees.full)
```

Linear mixed model fit by REML ['lmerMod']
Formula: height ~ dbh + (1 | site) + temp.c
Data: trees.full

REML criterion at convergence: 5098.2

Scaled residuals:

| Min | 1Q | Median | 3Q | Max |
|---------|---------|--------|--------|--------|
| -3.3247 | -0.6517 | 0.0192 | 0.6663 | 3.7268 |

Random effects:

| Groups | Name | Variance | Std.Dev. |
|--------|-------------|----------|----------|
| site | (Intercept) | 3.158 | 1.777 |
| | Residual | 9.266 | 3.044 |

Number of obs: 1000, groups: site, 10

Fixed effects:

| | Estimate | Std. Error | t value |
|-------------|-----------|------------|---------|
| (Intercept) | 18.340954 | 0.655054 | 27.999 |
| dbh | 0.616894 | 0.007571 | 81.484 |
| temp.c | 1.115104 | 0.248000 | 4.496 |

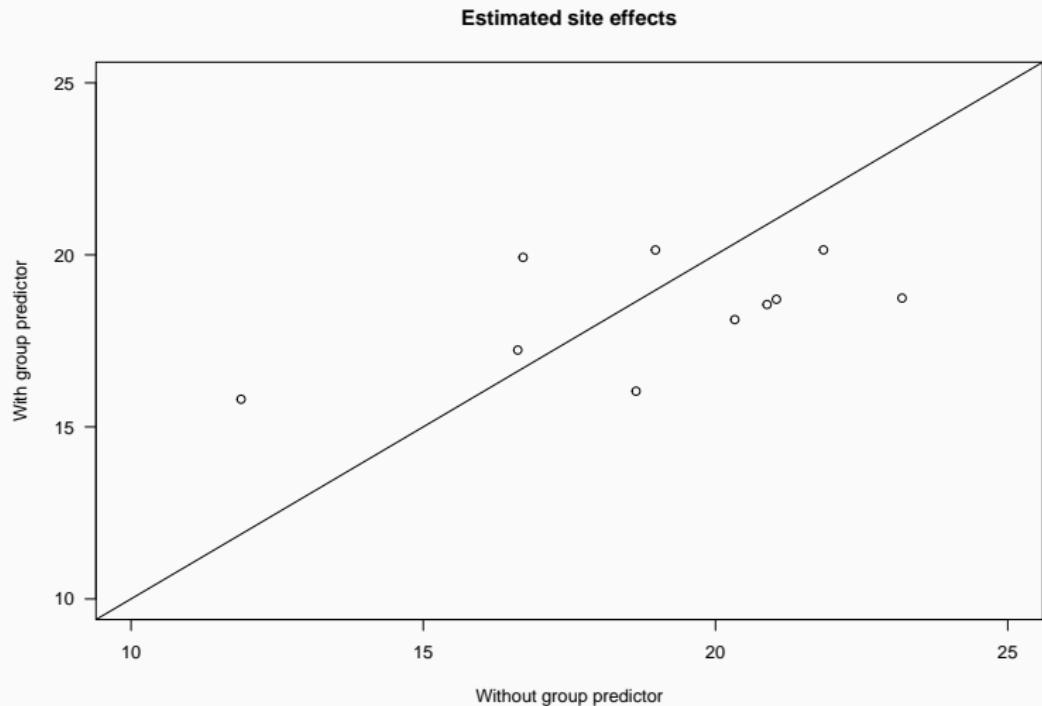
Correlation of Fixed Effects:

| (Intr) | dbh |
|--------|-----|
| 1.000 | |

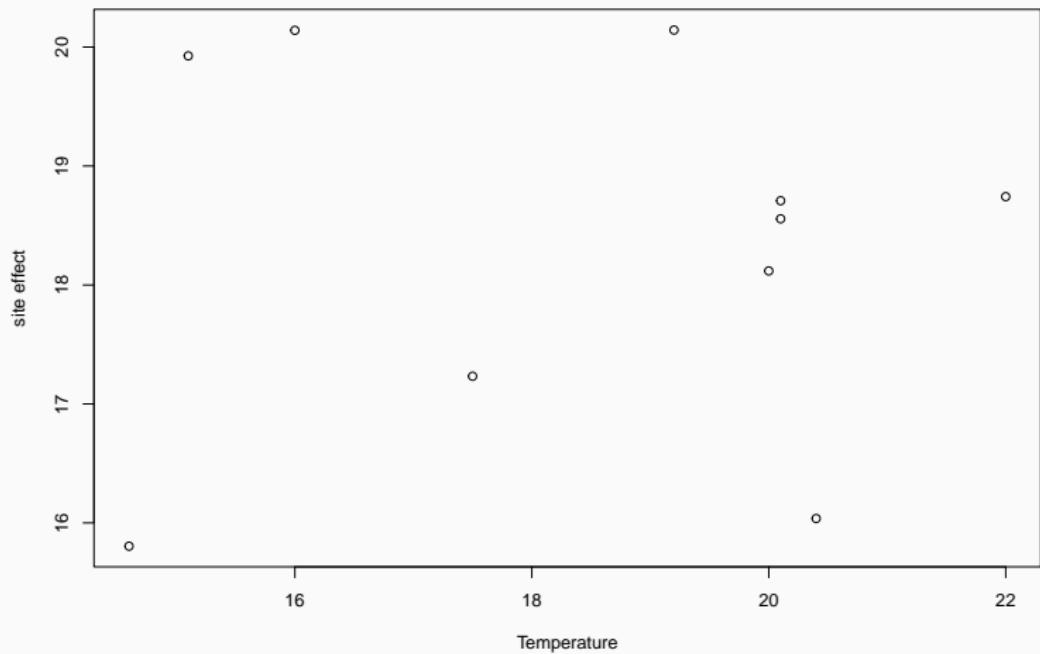
Examine model with merTools

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

Comparing site effects with and without group predictor



Are site effects related to temperature?



Varying intercepts and slopes

Varying intercepts and slopes

There is overall difference in height among sites (different intercepts)

AND

Relationship between DBH and Height varies among sites (different slopes)

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

Varying intercepts and slopes

```
Linear mixed model fit by REML ['lmerMod']  
Formula: height ~ dbh + (1 + dbh | site)  
Data: trees
```

REML criterion at convergence: 5105.1

Scaled residuals:

| Min | 1Q | Median | 3Q | Max |
|---------|---------|--------|--------|--------|
| -3.3342 | -0.6599 | 0.0375 | 0.6916 | 3.7756 |

Random effects:

| Groups | Name | Variance | Std.Dev. | Corr |
|----------|-------------|-----------|----------|-------|
| site | (Intercept) | 1.566e+01 | 3.95671 | |
| | dbh | 3.087e-04 | 0.01757 | -1.00 |
| Residual | | 9.226e+00 | 3.03744 | |

Number of obs: 1000, groups: site, 10

Fixed effects:

| | Estimate | Std. Error | t value |
|-------------|----------|------------|---------|
| (Intercept) | 18.95272 | 1.29190 | 14.67 |
| dbh | 0.61837 | 0.00946 | 65.37 |

Varying intercepts and slopes

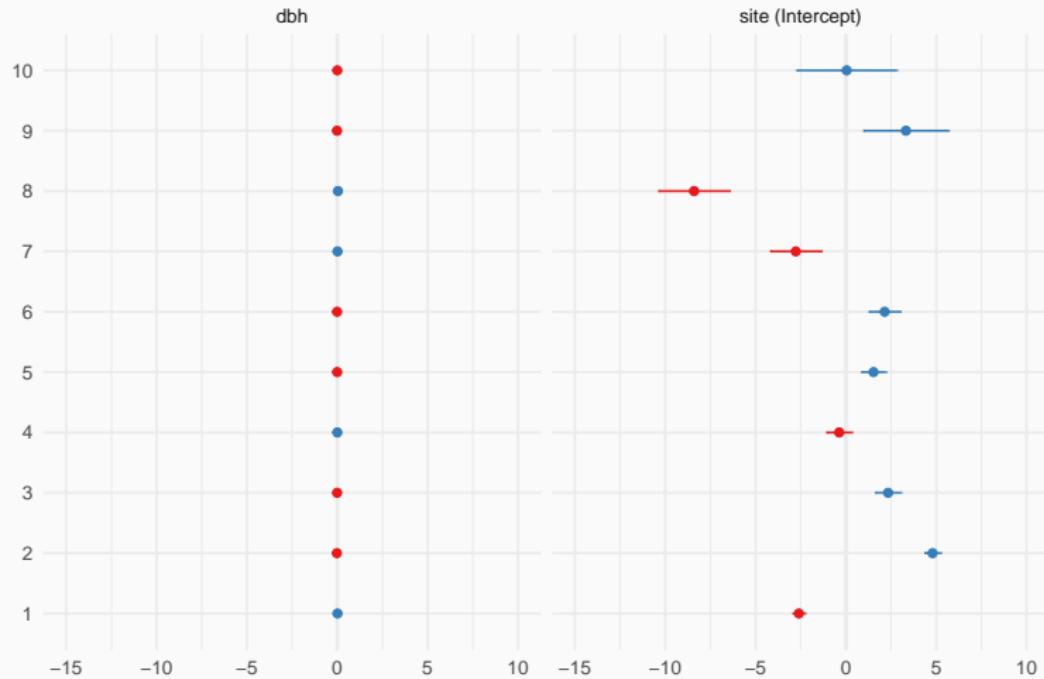
```
$site
  (Intercept)      dbh
1  16.34655  0.6299443
2  23.74733  0.5970814
3  21.28802  0.6080019
4  18.57844  0.6200337
5  20.47961  0.6115916
6  21.09608  0.6088542
7  16.17675  0.6306983
8  10.54681  0.6556978
9  22.27301  0.6036281
10 18.99463  0.6181856

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

Visualising model: sjPlot

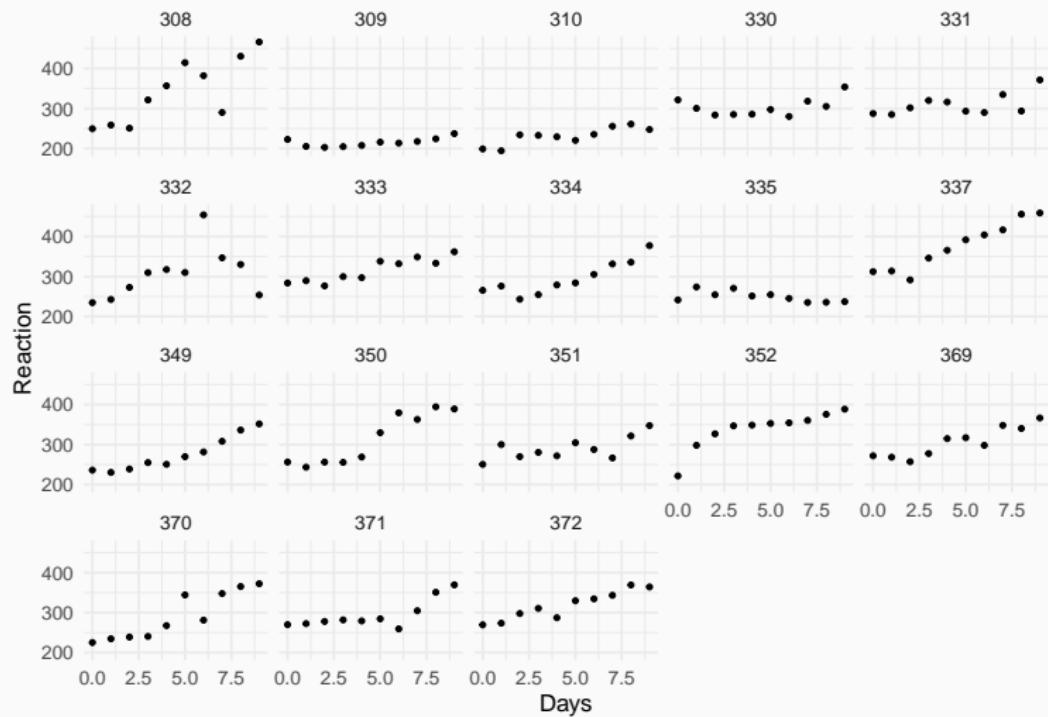
```
plot_model(mixed.slopes, type = "re")
```

Random effects



More examples

sleepstudy (repeated measures)



Varying intercepts and slopes (lme4)

```
sleep <- lmer(Reaction ~ Days + (1+Days|Subject), data = sleepstudy)
```

Linear mixed model fit by REML ['lmerMod']
Formula: Reaction ~ Days + (1 + Days | Subject)
Data: sleepstudy

REML criterion at convergence: 1743.6

Scaled residuals:

| Min | 1Q | Median | 3Q | Max |
|---------|---------|--------|--------|--------|
| -3.9536 | -0.4634 | 0.0231 | 0.4634 | 5.1793 |

Random effects:

| Groups | Name | Variance | Std.Dev. | Corr |
|---------|-------------|----------|----------|------|
| Subject | (Intercept) | 612.10 | 24.741 | |
| | Days | 35.07 | 5.922 | 0.07 |
| | Residual | 654.94 | 25.592 | |

Number of obs: 180, groups: Subject, 18

Fixed effects:

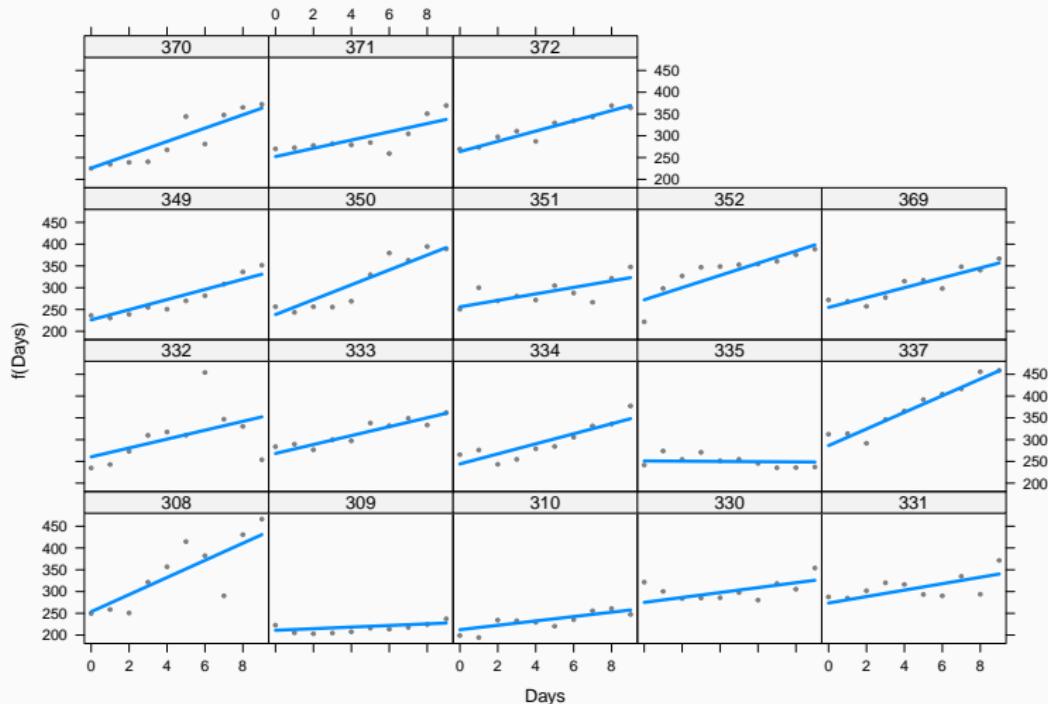
| | Estimate | Std. Error | t value |
|-------------|----------|------------|---------|
| (Intercept) | 251.405 | 6.825 | 36.838 |
| Days | 10.467 | 1.546 | 6.771 |

Correlation of Fixed Effects:

(Intr)

Varying intercepts and slopes (lme4)

```
visreg(sleep, xvar = "Days", by = "Subject", re.form = NULL)
```



Fitting multilevel models (GAMM) with mgcv

```
sgamm <- mgcv:::gam(Reaction ~ s(Days, Subject, k = 3, bs = "fs"),
                     data = sleepstudy, method = "REML")
```

Family: gaussian

Link function: identity

Formula:

Reaction ~ s(Days, Subject, k = 3, bs = "fs")

Parametric coefficients:

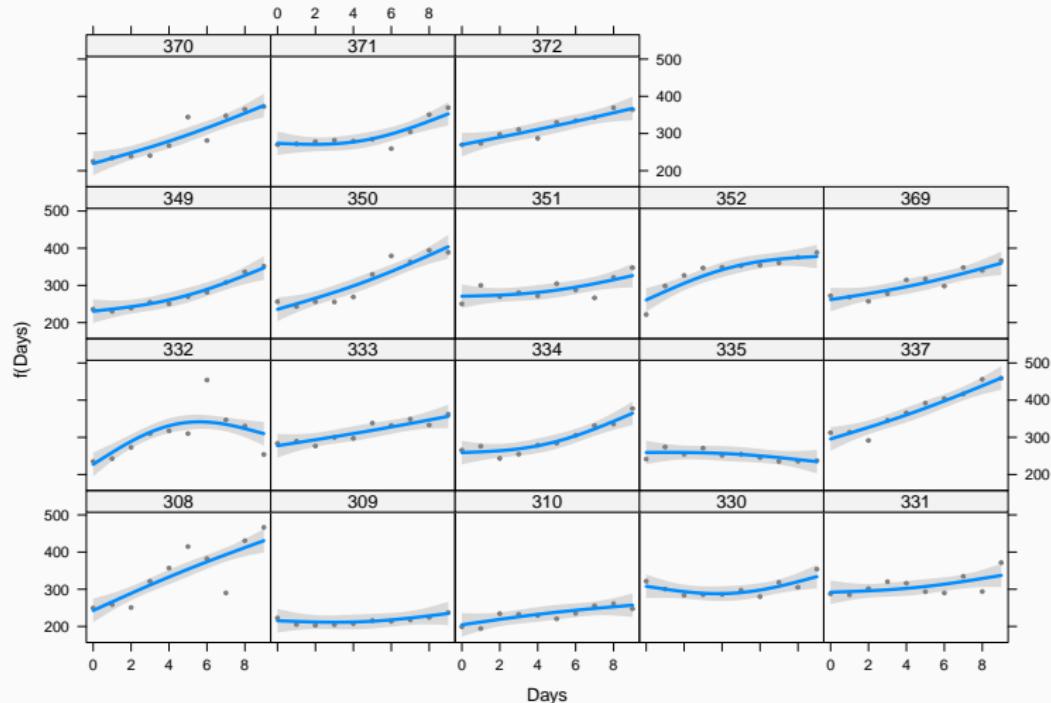
| | Estimate | Std. Error | t value | Pr(> t) | | | | | | | |
|----------------|----------|------------|---------|----------|------|-----|------|-----|-----|-----|---|
| (Intercept) | 298.51 | 9.05 | 32.98 | <2e-16 | *** | | | | | | |
| --- | | | | | | | | | | | |
| Signif. codes: | 0 | '***' | 0.001 | '**' | 0.01 | '*' | 0.05 | '.' | 0.1 | ' ' | 1 |

Approximate significance of smooth terms:

| | edf | Ref.df | F | p-value | |
|-----------------|-------|--------|-------|---------|-----|
| s(Days,Subject) | 45.67 | 53 | 17.11 | <2e-16 | *** |

Fitting multilevel models (GAMM) with mgcv

```
visreg(sgamm, xvar = "Days", by = "Subject")
```



Hierarchical generalized additive models: an introduction with `mgcv`

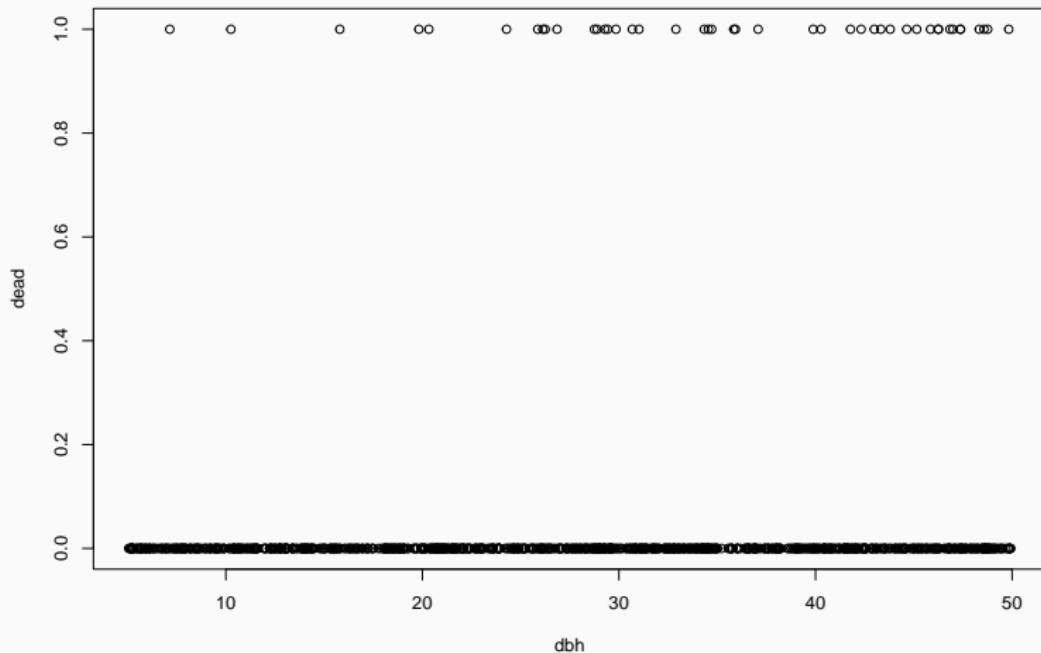
Eric J Pedersen ^{Corresp.} ^{1,2}, David L. Miller ^{3,4}, Gavin L. Simpson ⁵, Noam Ross ⁶

<https://doi.org/10.7287/peerj.preprints.27320v1>

Multilevel logistic regression

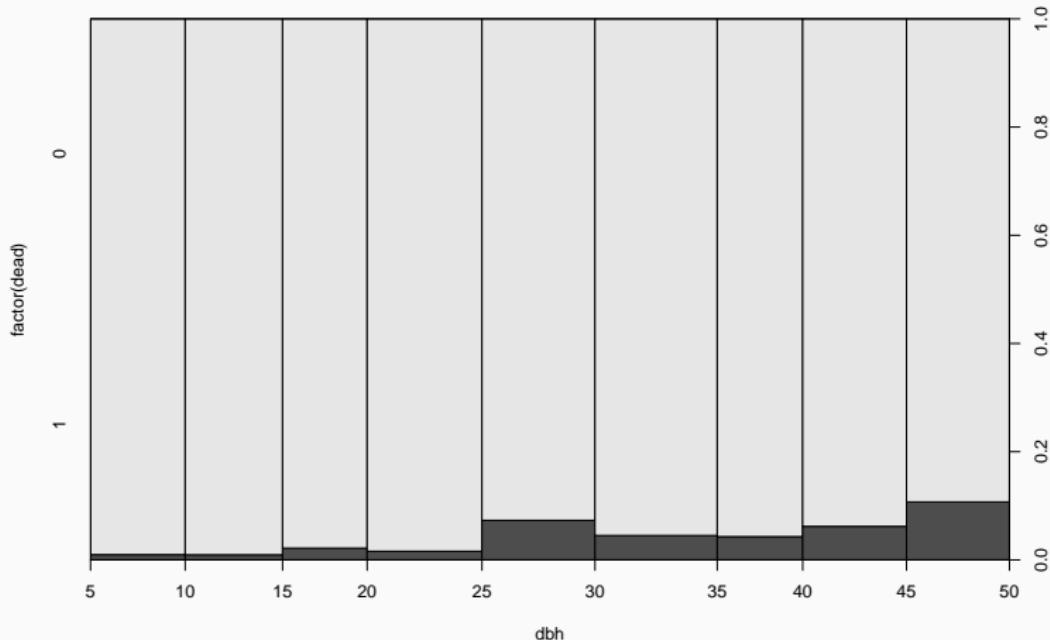
Q: Relationship between tree size and mortality

```
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 |
|---------|---------|---------|---------|--------|
| -0.4805 | -0.3520 | -0.2647 | -0.1928 | 2.9690 |

Coefficients:

| | Estimate | Std. Error | z value | Pr(> z) |
|-------------|----------|------------|---------|--------------|
| (Intercept) | -4.77874 | 0.50902 | -9.388 | < 2e-16 *** |
| dbh | 0.05365 | 0.01377 | 3.895 | 9.82e-05 *** |

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

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 360.91 on 999 degrees of freedom
Residual deviance: 343.69 on 998 degrees of freedom
AIC: 347.69

Number of Fisher Scoring iterations: 6

Logistic regression with *independent* site effects

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

Call:

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

Deviance Residuals:

| Min | 1Q | Median | 3Q | Max |
|---------|---------|---------|---------|--------|
| -0.6359 | -0.3449 | -0.2561 | -0.1852 | 2.9763 |

Coefficients:

| | Estimate | Std. Error | z value | Pr(> z) |
|----------------|-----------|------------|----------|------------|
| (Intercept) | -4.80123 | 0.54985 | -8.732 | <2e-16 *** |
| dbh | 0.05371 | 0.01381 | 3.889 | 0.0001 *** |
| factor(site)2 | -0.29692 | 0.46073 | -0.644 | 0.5193 |
| factor(site)3 | 0.21275 | 0.52799 | 0.403 | 0.6870 |
| factor(site)4 | 0.39841 | 0.53025 | 0.751 | 0.4524 |
| factor(site)5 | -0.42557 | 0.64018 | -0.665 | 0.5062 |
| factor(site)6 | 0.66861 | 0.53656 | 1.246 | 0.2127 |
| factor(site)7 | 0.11862 | 1.06211 | 0.112 | 0.9111 |
| factor(site)8 | 0.43899 | 1.08058 | 0.406 | 0.6846 |
| factor(site)9 | -13.63389 | 840.90382 | -0.016 | 0.9871 |
| factor(site)10 | -13.17148 | 1042.21823 | -0.013 | 0.9899 |
| --- | | | | |
| Signif. codes: | 0 '***' | 0.001 '**' | 0.01 '*' | 0.05 '.' |
| | 0.1 ' ' | 1 | | |

Fit multilevel logistic regression

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

```
Generalized linear mixed model fit by maximum likelihood (Laplace
Approximation) [glmerMod]
Family: binomial ( logit )
Formula: dead ~ dbh + (1 | site)
Data: trees
```

| AIC | BIC | logLik | deviance | df.resid |
|-------|-------|--------|----------|----------|
| 349.7 | 364.4 | -171.8 | 343.7 | 997 |

Scaled residuals:

| Min | 1Q | Median | 3Q | Max |
|---------|---------|---------|---------|--------|
| -0.3498 | -0.2528 | -0.1888 | -0.1370 | 9.0031 |

Random effects:

| Groups | Name | Variance | Std.Dev. |
|--------|-------------|----------|----------|
| site | (Intercept) | 0 | 0 |

Number of obs: 1000, groups: site, 10

Fixed effects:

| | Estimate | Std. Error | z value | Pr(> z) |
|-------------|----------|------------|---------|--------------|
| (Intercept) | -4.77874 | 0.50904 | -9.388 | < 2e-16 *** |
| dbh | 0.05365 | 0.01377 | 3.895 | 9.83e-05 *** |
| --- | | | | |

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

Retrieve model coefficients

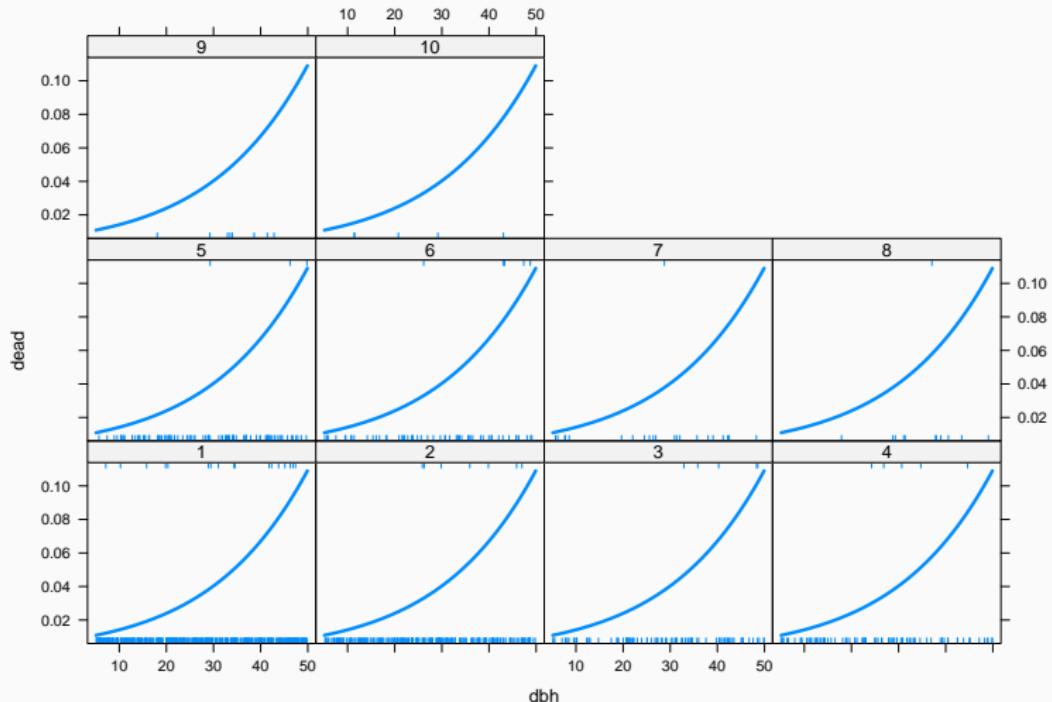
```
coef(mixed.logis)
```

```
$site
  (Intercept)      dbh
1 -4.778744 0.05364989
2 -4.778744 0.05364989
3 -4.778744 0.05364989
4 -4.778744 0.05364989
5 -4.778744 0.05364989
6 -4.778744 0.05364989
7 -4.778744 0.05364989
8 -4.778744 0.05364989
9 -4.778744 0.05364989
10 -4.778744 0.05364989
```

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

Visualising model: visreg

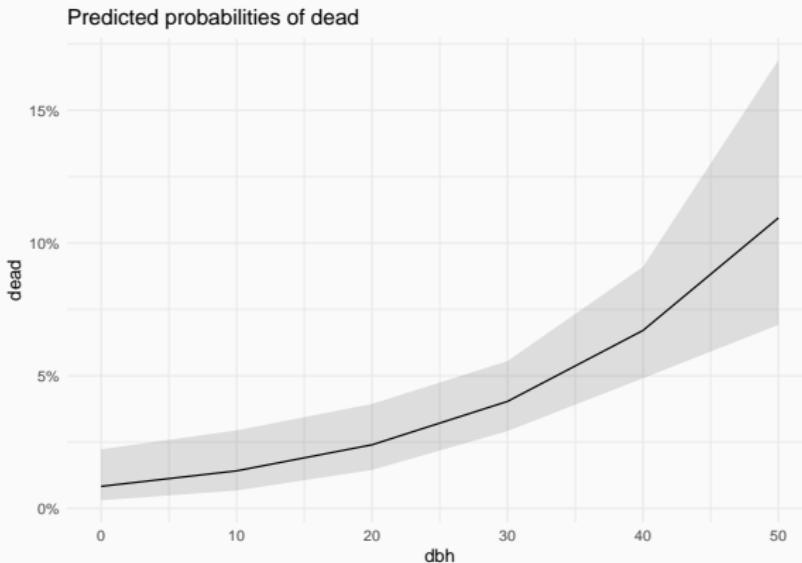
```
visreg(mixed.logis, xvar = "dbh", by = "site", scale = "response")
```



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

- Varying intercepts

Formula syntax for different models

- Varying intercepts
 - $y \sim x + (1 | group)$

Formula syntax for different models

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

Formula syntax for different models

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

Formula syntax for different models

- Varying intercepts
 - $y \sim x + (1 \mid \text{group})$
- Varying intercepts and slopes
 - $y \sim x + (1 + x \mid \text{group})$
- Varying intercepts, 2 groups (crossed)

Formula syntax for different models

- Varying intercepts
 - $y \sim x + (1 \mid \text{group})$
- Varying intercepts and slopes
 - $y \sim x + (1 + x \mid \text{group})$
- Varying intercepts, 2 groups (crossed)
 - $y \sim x + (1 \mid \text{group1}) + (1 \mid \text{group2})$

Formula syntax for different models

- Varying intercepts
 - $y \sim x + (1 \mid \text{group})$
- Varying intercepts and slopes
 - $y \sim x + (1 + x \mid \text{group})$
- Varying intercepts, 2 groups (crossed)
 - $y \sim x + (1 \mid \text{group1}) + (1 \mid \text{group2})$
- Varying intercepts, 2 groups (nested)

Formula syntax for different models

- Varying intercepts
 - $y \sim x + (1 \mid \text{group})$
- Varying intercepts and slopes
 - $y \sim x + (1 + x \mid \text{group})$
- Varying intercepts, 2 groups (crossed)
 - $y \sim x + (1 \mid \text{group1}) + (1 \mid \text{group2})$
- Varying intercepts, 2 groups (nested)
 - $y \sim x + (1 \mid \text{group/subgroup})$

Formula syntax for different models

- Varying intercepts
 - $y \sim x + (1 \mid \text{group})$
- Varying intercepts and slopes
 - $y \sim x + (1 + x \mid \text{group})$
- Varying intercepts, 2 groups (crossed)
 - $y \sim x + (1 \mid \text{group1}) + (1 \mid \text{group2})$
- Varying intercepts, 2 groups (nested)
 - $y \sim x + (1 \mid \text{group/subgroup})$
 - This is **equivalent** to $y \sim x + (1 \mid \text{group1}) + (1 \mid \text{group2})$ with distinct labelling of group levels.

Formula syntax for different models

- Varying intercepts
 - $y \sim x + (1 \mid \text{group})$
- Varying intercepts and slopes
 - $y \sim x + (1 + x \mid \text{group})$
- Varying intercepts, 2 groups (crossed)
 - $y \sim x + (1 \mid \text{group1}) + (1 \mid \text{group2})$
- Varying intercepts, 2 groups (nested)
 - $y \sim x + (1 \mid \text{group/subgroup})$
 - This is **equivalent** to $y \sim x + (1 \mid \text{group1}) + (1 \mid \text{group2})$ with distinct labelling of group levels.
- Varying intercepts and slopes, 2 groups (crossed)

Formula syntax for different models

- Varying intercepts
 - $y \sim x + (1 \mid \text{group})$
- Varying intercepts and slopes
 - $y \sim x + (1 + x \mid \text{group})$
- Varying intercepts, 2 groups (crossed)
 - $y \sim x + (1 \mid \text{group1}) + (1 \mid \text{group2})$
- Varying intercepts, 2 groups (nested)
 - $y \sim x + (1 \mid \text{group/subgroup})$
 - This is **equivalent** to $y \sim x + (1 \mid \text{group1}) + (1 \mid \text{group2})$ with distinct labelling of group levels.
- Varying intercepts and slopes, 2 groups (crossed)
 - $y \sim x + (1 + x \mid \text{group1}) + (1 + x \mid \text{group2})$

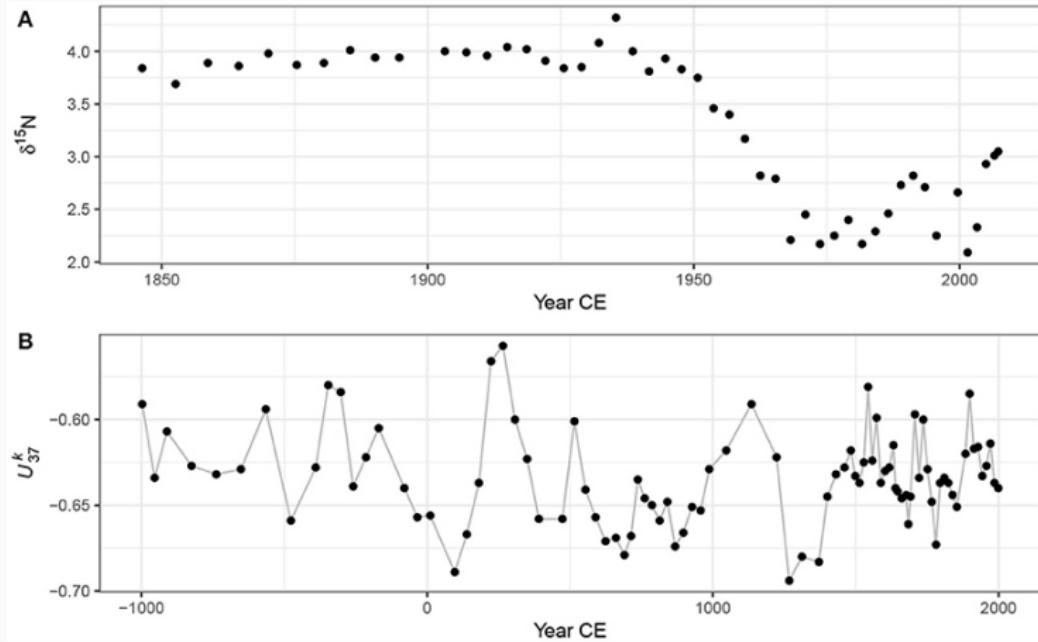
<https://bbolker.github.io/mixedmodels-misc/glmmFAQ.html>

Generalised Additive Models

Francisco Rodríguez-Sánchez

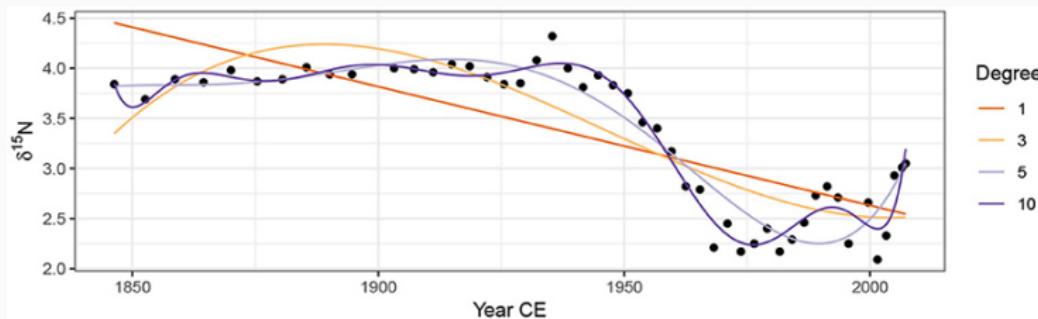
<https://frodriguezsanchez.net>

How do we model these time series?



Simpson 2018

How do we model these time series?



Simpson 2018

GAMs allow us to model non-linear relationships using smooths

Generalised Linear Model (GLM):

$$y = a + bx$$

Generalised Additive Model (GAM):

$$y = a + s(x)$$

Modelling non-linear time series with GAM

```
isotopes <- readRDS("data/isotope.rds")
```

| | Depth | d13C | TotalC | d15N | TotalN | DryWeight | Year |
|---|-------|--------|---------|------|--------|-----------|----------|
| 1 | 0.2 | -27.57 | 806.49 | 3.05 | 64.21 | 8.2 | 2007.254 |
| 2 | 0.4 | -27.67 | 949.33 | 3.01 | 73.26 | 7.6 | 2006.510 |
| 3 | 0.8 | -27.63 | 1305.52 | 2.93 | 93.25 | 11.6 | 2004.941 |
| 4 | 1.2 | -27.62 | 1136.04 | 2.33 | 86.09 | 9.6 | 2003.269 |
| 5 | 1.6 | -27.48 | 1028.27 | 2.09 | 93.80 | 10.9 | 2001.496 |
| 6 | 2.0 | -27.39 | 809.91 | 2.66 | 79.98 | 9.9 | 1999.626 |

Modelling non-linear time series with GAM

```
library("mgcv")
m <- gam(d15N ~ s(Year, k = 15), data = isotopes, method = "REML")
```

Family: gaussian

Link function: identity

Formula:

d15N ~ s(Year, k = 15)

Parametric coefficients:

| | Estimate | Std. Error | t value | Pr(> t) |
|--|----------|------------|---------|----------|
|--|----------|------------|---------|----------|

| | | | | |
|-------------|---------|---------|-------|------------|
| (Intercept) | 3.30958 | 0.02622 | 126.2 | <2e-16 *** |
|-------------|---------|---------|-------|------------|

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

Approximate significance of smooth terms:

| edf | Ref.df | F | p-value |
|-----|--------|---|---------|
|-----|--------|---|---------|

| | | | | |
|---------|-------|-------|-------|------------|
| s(Year) | 9.282 | 11.07 | 61.33 | <2e-16 *** |
|---------|-------|-------|-------|------------|

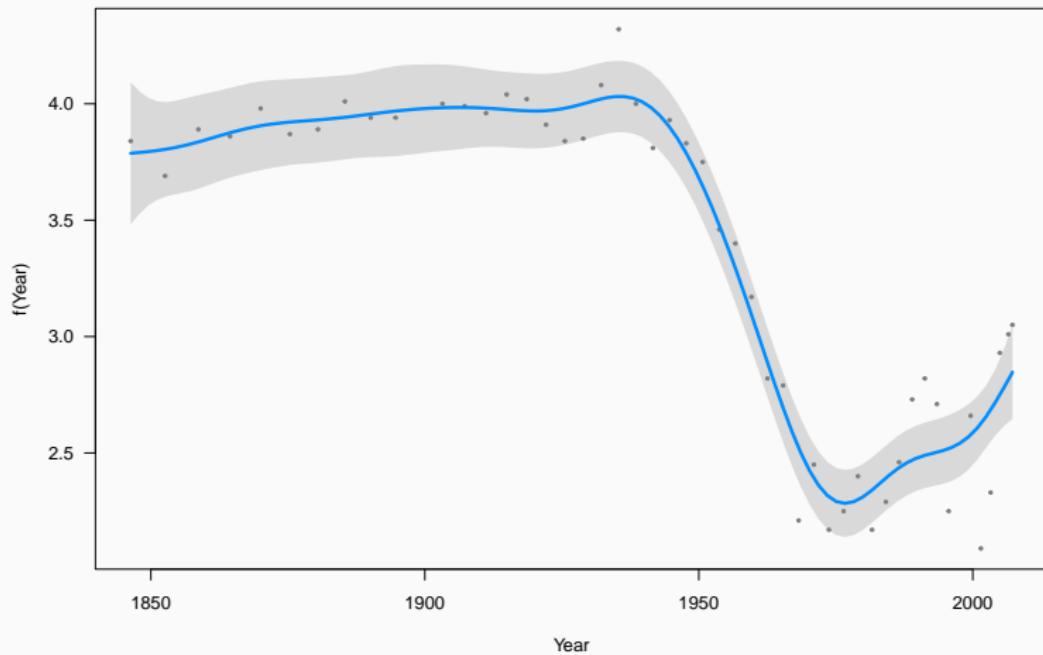
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

R-sq.(adj) = 0.935 Deviance explained = 94.8%

-REML = 3.9734 Scale est. = 0.03299 n = 48

Visualising fitted GAM

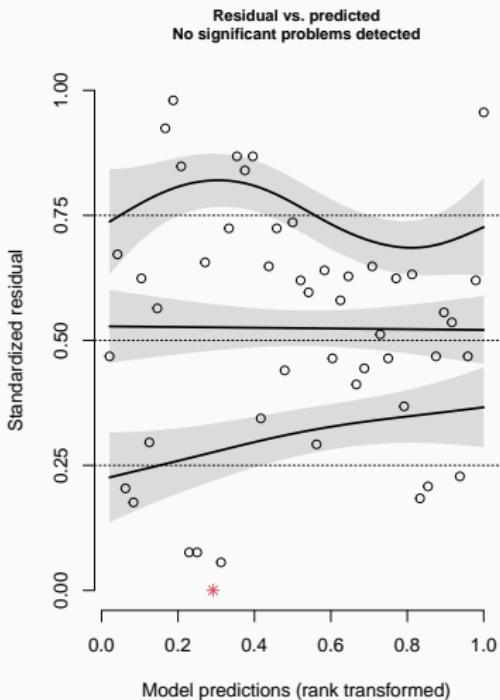
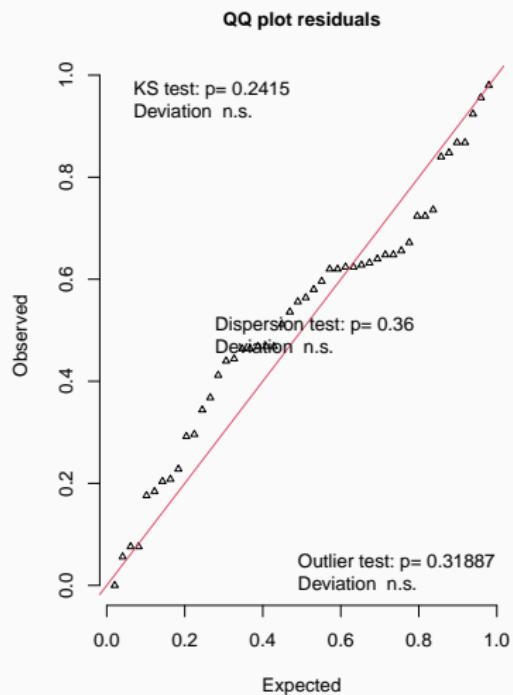
```
visreg(m)
```



Checking fitted GAM

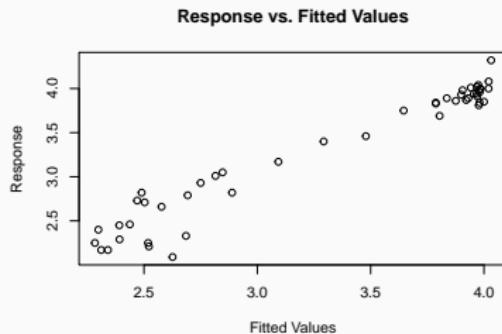
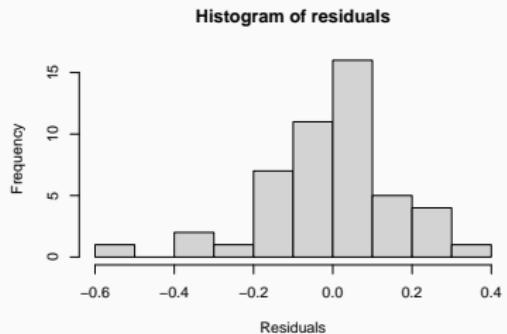
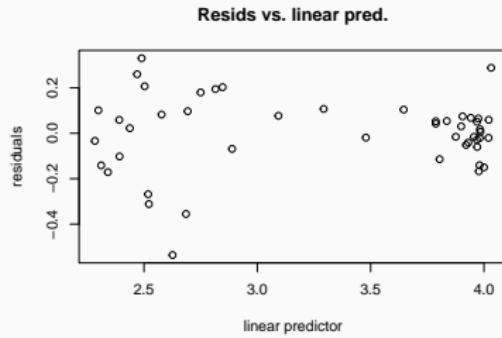
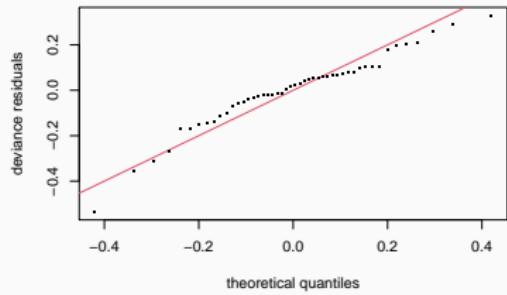
```
library("DHARMa")
simulateResiduals(m, plot = TRUE)
```

DHARMa residual diagnostics



Checking fitted GAM

```
gam.check(m)
```



Method: REML

Optimizer: outer newton

Including temporal autocorrelation

```
mod <- gamm(d15N ~ s(Year, k = 15), data = isotopes,
             correlation = corCAR1(form = ~ Year), method = "REML")
```

Family: gaussian

Link function: identity

Formula:

d15N ~ s(Year, k = 15)

Parametric coefficients:

| | Estimate | Std. Error | t value | Pr(> t) |
|-------------|----------|------------|---------|------------|
| (Intercept) | 3.30909 | 0.03489 | 94.84 | <2e-16 *** |

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

Approximate significance of smooth terms:

| | edf | Ref.df | F | p-value |
|---------|-------|--------|-------|------------|
| s(Year) | 7.954 | 7.954 | 47.44 | <2e-16 *** |

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

R-sq.(adj) = 0.929

Modelling infant mortality

Modelling infant mortality

```
mort <- read.csv("data/UN_GDP_infantmortality.csv")
```

| | country | infant.mortality | gdp |
|---|----------------|------------------|------|
| 1 | Afghanistan | 154 | 2848 |
| 2 | Albania | 32 | 863 |
| 3 | Algeria | 44 | 1531 |
| 4 | American.Samoa | 11 | NA |
| 5 | Andorra | NA | NA |
| 6 | Angola | 124 | 355 |

Modelling infant mortality with a GLM

```
library("MASS")
mort.glm <- glm.nb(infant.mortality ~ gdp, data = mort)
```

Call:

```
glm.nb(formula = infant.mortality ~ gdp, data = mort, init.theta = 2.460991808,
       link = log)
```

Deviance Residuals:

| Min | 1Q | Median | 3Q | Max |
|---------|---------|---------|--------|--------|
| -2.8024 | -1.0447 | -0.3650 | 0.5232 | 2.9116 |

Coefficients:

| | Estimate | Std. Error | z value | Pr(> z) |
|-------------|------------|------------|---------|------------|
| (Intercept) | 4.072e+00 | 5.727e-02 | 71.11 | <2e-16 *** |
| gdp | -8.675e-05 | 6.221e-06 | -13.95 | <2e-16 *** |

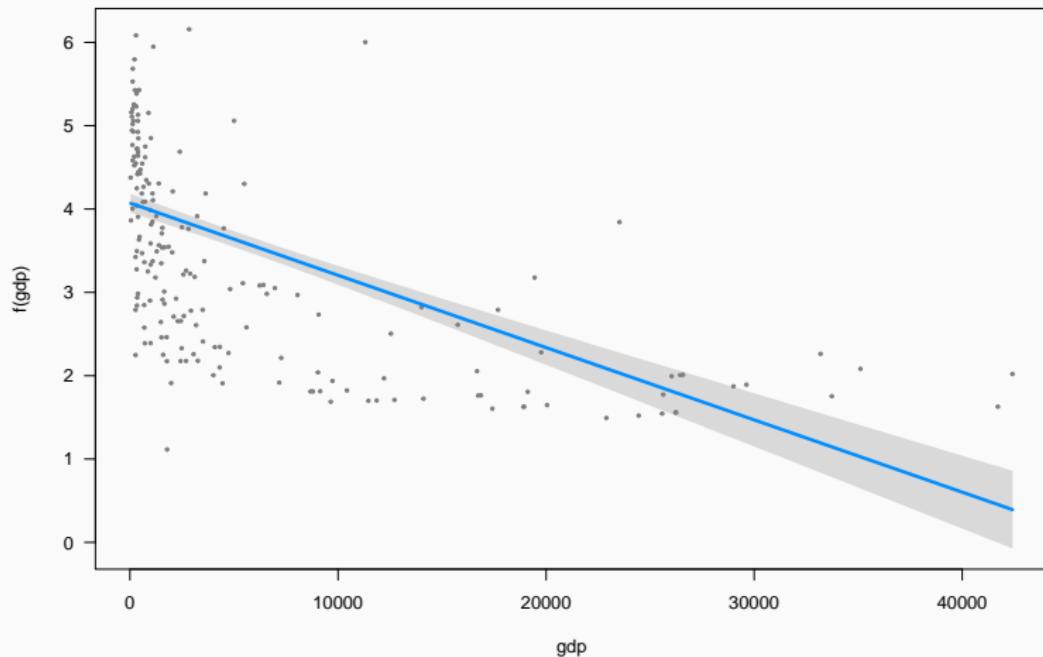
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for Negative Binomial(2.461) family taken to be 1)

Null deviance: 385.83 on 192 degrees of freedom

Residual deviance: 202.51 on 191 degrees of freedom

Modelling infant mortality with a GLM



Modelling infant mortality with a GLM (log.gdp)

```
mort$log.gdp <- log(mort$gdp)
mort.glm.log <- glm.nb(infant.mortality ~ log.gdp, data = mort)
```

Call:

```
glm.nb(formula = infant.mortality ~ log.gdp, data = mort, init.theta = 3.119314453,
       link = log)
```

Deviance Residuals:

| Min | 1Q | Median | 3Q | Max |
|---------|---------|---------|--------|--------|
| -2.7517 | -0.8692 | -0.3575 | 0.3090 | 4.5063 |

Coefficients:

| | Estimate | Std. Error | z value | Pr(> z) |
|-------------|----------|------------|---------|------------|
| (Intercept) | 7.07818 | 0.20045 | 35.31 | <2e-16 *** |
| log.gdp | -0.47238 | 0.02647 | -17.85 | <2e-16 *** |
| --- | | | | |

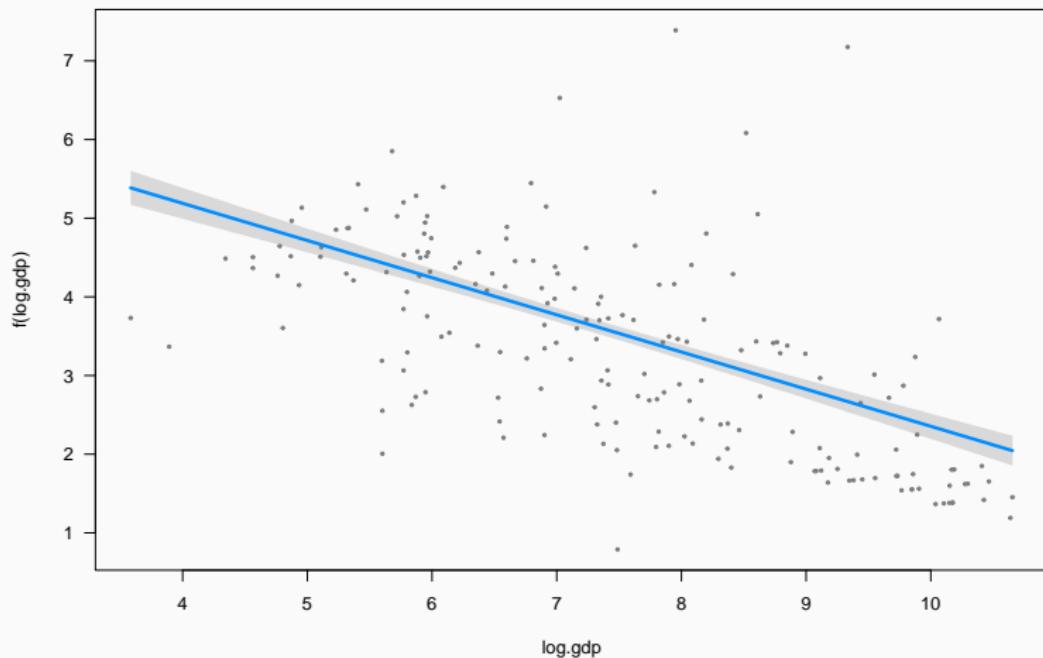
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for Negative Binomial(3.1193) family taken to be 1)

Null deviance: 478.54 on 192 degrees of freedom

Residual deviance: 198.03 on 191 degrees of freedom

Modelling infant mortality with a GLM (log.gdp)



Modelling infant mortality with a GAM

```
library("mgcv")
mort.gam <- gam(infant.mortality ~ s(log.gdp), family = nb, data = mort)
```

Family: gaussian

Link function: identity

Formula:

d15N ~ s(Year, k = 15)

Parametric coefficients:

| | Estimate | Std. Error | t value | Pr(> t) |
|-------------|----------|------------|---------|------------|
| (Intercept) | 3.30958 | 0.02622 | 126.2 | <2e-16 *** |

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

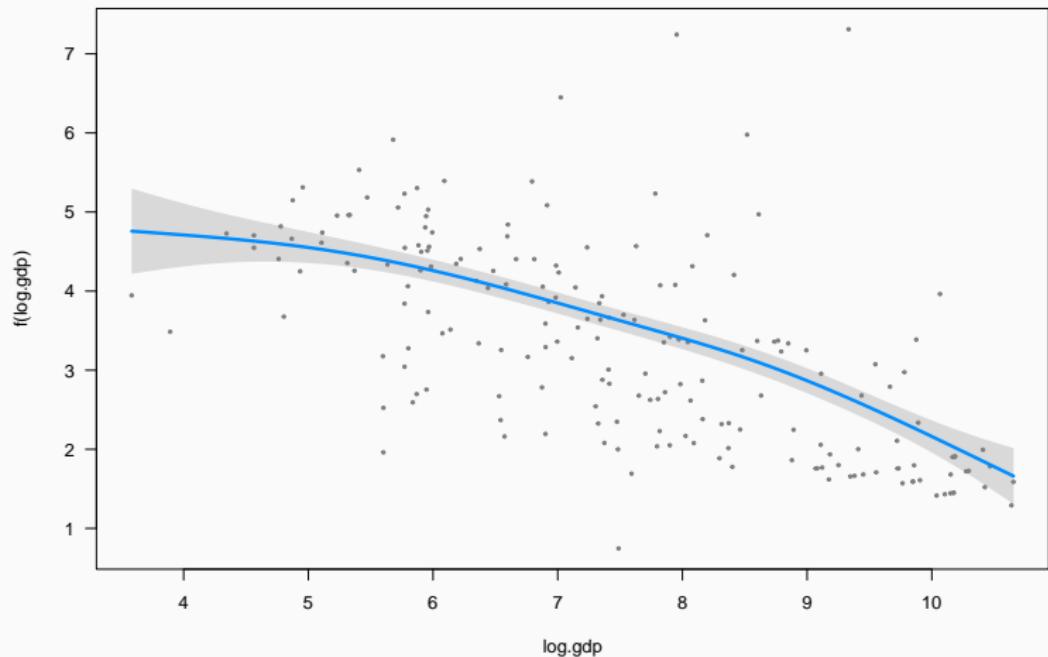
Approximate significance of smooth terms:

| | edf | Ref.df | F | p-value |
|---------|-------|--------|-------|------------|
| s(Year) | 9.282 | 11.07 | 61.33 | <2e-16 *** |

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

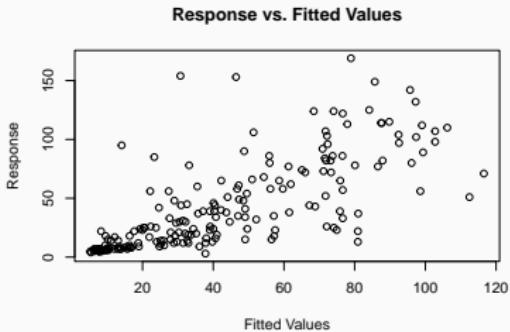
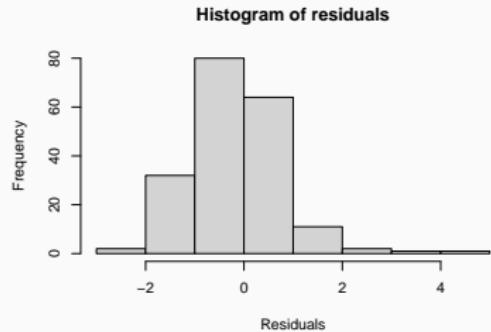
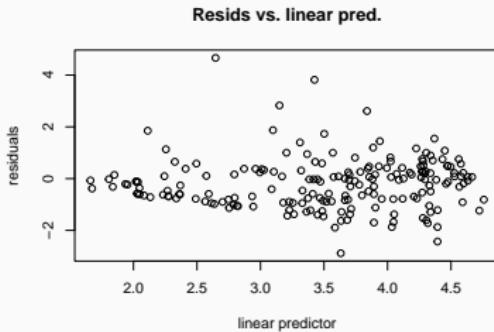
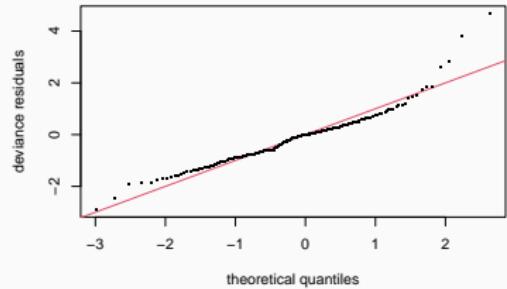
R-sq.(adj) = 0.935 Deviance explained = 94.8%

Modelling infant mortality with a GAM



Checking GAM

```
gam.check(mort.gam)
```



Method: REML

Optimizer: outer newton

Comparing models

```
library("performance")
compare_performance(mort.glm, mort.glm.log, mort.gam)
```

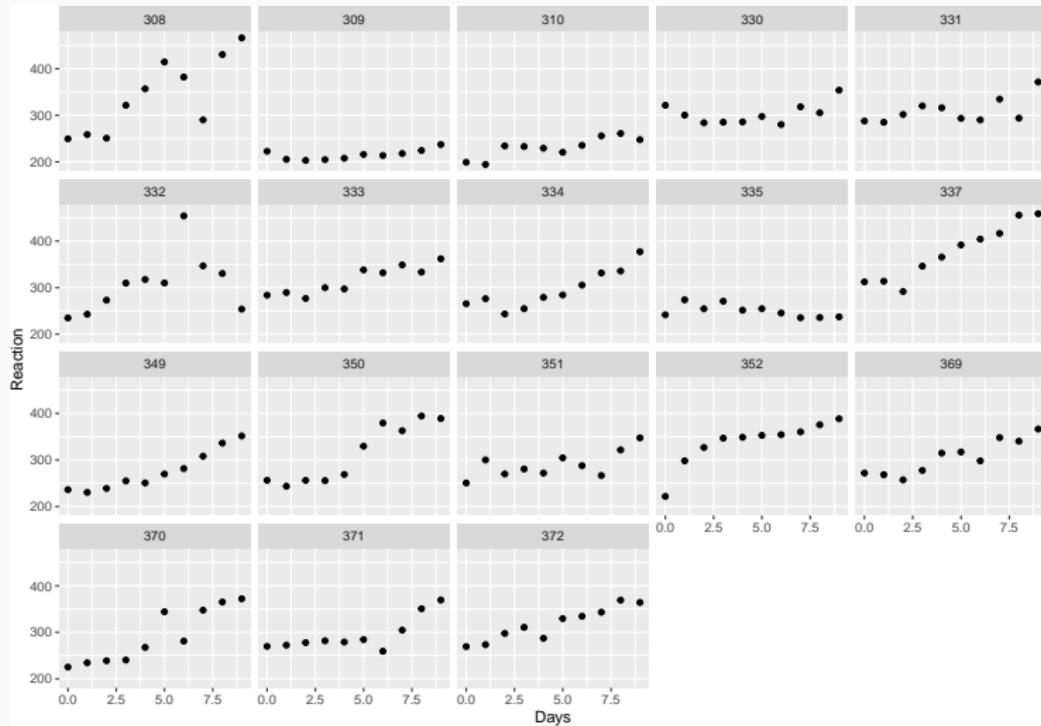
```
# Comparison of Model Performance Indices
```

| Name | Model | AIC | BIC | RMSE | Sigma | Score |
|--------------|--------|----------|----------|--------|-------|-------|
| <hr/> | | | | | | |
| mort.glm | negbin | 1714.957 | 1724.745 | 31.089 | 1.030 | |
| mort.glm.log | negbin | 1667.750 | 1677.538 | 30.034 | 1.018 | |
| mort.gam | gam | 1661.141 | 1680.512 | 26.249 | 1.027 | |

Generalised Additive Mixed Models (GAMM)

Reaction time with sleep deprivation

```
library("lme4")
data("sleepstudy")
```



Modelling reaction time with sleep deprivation (GAMM)

```
sgamm <- gam(Reaction ~ s(Days, Subject, k = 3, bs = "fs"),
              data = sleepstudy, method = "REML")
```

Family: gaussian

Link function: identity

Formula:

```
Reaction ~ s(Days, Subject, k = 3, bs = "fs")
```

Parametric coefficients:

| | Estimate | Std. Error | t value | Pr(> t) |
|--|----------|------------|---------|----------|
|--|----------|------------|---------|----------|

```
(Intercept) 295.22      10.49   28.15  <2e-16 ***
```

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

Approximate significance of smooth terms:

| | edf | Ref.df | F | p-value |
|--|-----|--------|---|---------|
|--|-----|--------|---|---------|

```
s(Days,Subject) 42.2      53 16.05  <2e-16 ***
```

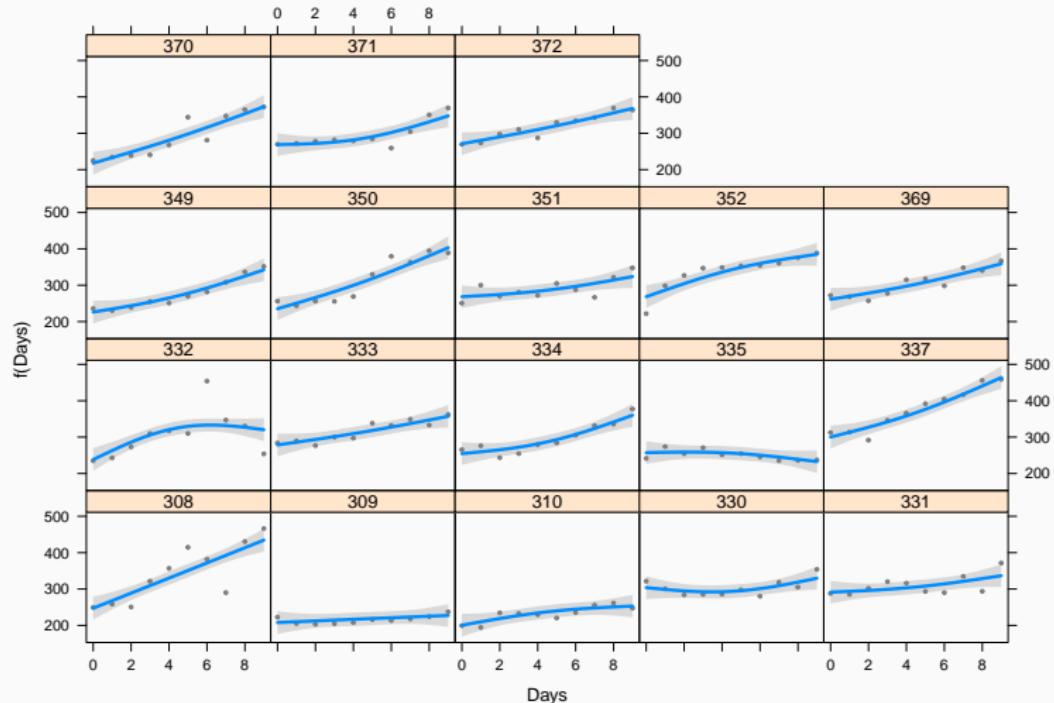
```
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

R-sq.(adj) = 0.826 Deviance explained = 86.7%

-REML = 886.71 Scale est. = 551.61 n = 180

Modelling reaction time with sleep deprivation (GAMM)

```
visreg(sgamm, xvar = "Days", by = "Subject")
```



An introduction to Bayesian modelling with brms and Stan

Francisco Rodríguez-Sánchez

<https://frodriguezsanchez.net>

Our dataset: tree heights and DBH

- One species

```
trees <- read.csv("data/trees.csv")
```

| | site | dbh | height |
|----------|------|---------------|---------------|
| Min. : | 1.0 | Min. : 5.06 | Min. :13.40 |
| 1st Qu.: | 1.0 | 1st Qu.:17.69 | 1st Qu.:29.68 |
| Median : | 2.0 | Median :28.62 | Median :36.55 |
| Mean : | 2.7 | Mean :27.88 | Mean :36.51 |
| 3rd Qu.: | 4.0 | 3rd Qu.:38.97 | 3rd Qu.:43.33 |
| Max. : | 10.0 | Max. :49.92 | Max. :59.30 |

Our dataset: tree heights and DBH

- One species
- 10 plots

```
trees <- read.csv("data/trees.csv")
```

| | site | dbh | height |
|----------|------|---------------|---------------|
| Min. : | 1.0 | Min. : 5.06 | Min. :13.40 |
| 1st Qu.: | 1.0 | 1st Qu.:17.69 | 1st Qu.:29.68 |
| Median : | 2.0 | Median :28.62 | Median :36.55 |
| Mean : | 2.7 | Mean :27.88 | Mean :36.51 |
| 3rd Qu.: | 4.0 | 3rd Qu.:38.97 | 3rd Qu.:43.33 |
| Max. : | 10.0 | Max. :49.92 | Max. :59.30 |

Our dataset: tree heights and DBH

- One species
- 10 plots
- 1000 trees

```
trees <- read.csv("data/trees.csv")
```

| | site | dbh | height |
|----------|------|---------------|---------------|
| Min. : | 1.0 | Min. : 5.06 | Min. :13.40 |
| 1st Qu.: | 1.0 | 1st Qu.:17.69 | 1st Qu.:29.68 |
| Median : | 2.0 | Median :28.62 | Median :36.55 |
| Mean : | 2.7 | Mean :27.88 | Mean :36.51 |
| 3rd Qu.: | 4.0 | 3rd Qu.:38.97 | 3rd Qu.:43.33 |
| Max. : | 10.0 | Max. :49.92 | Max. :59.30 |

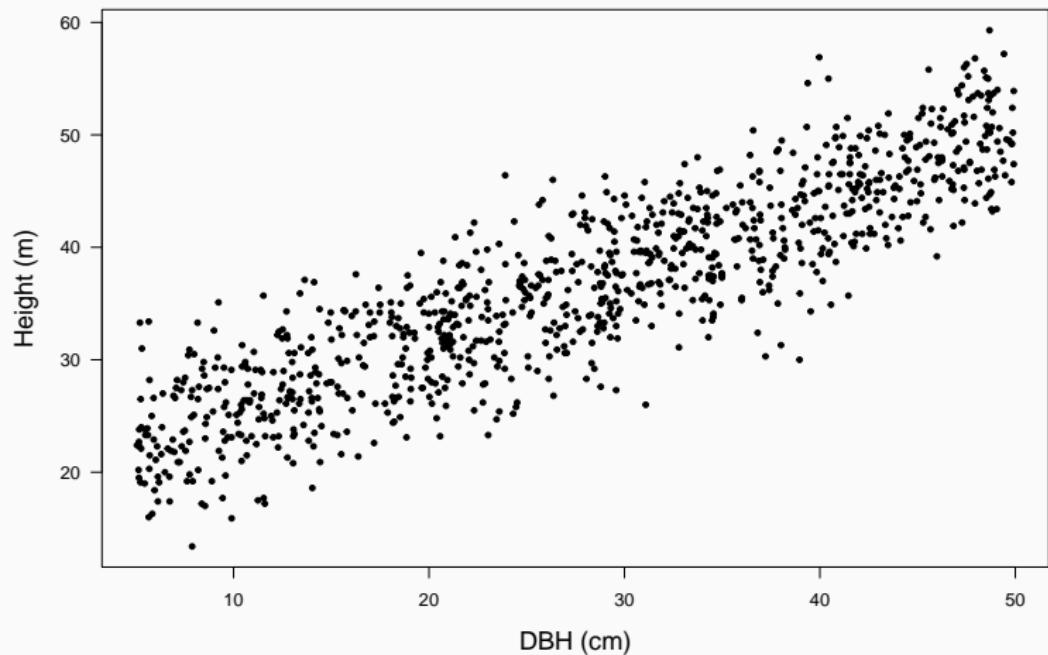
Our dataset: tree heights and DBH

- One species
- 10 plots
- 1000 trees
- Number of trees per plot ranging from 4 to 392

```
trees <- read.csv("data/trees.csv")
```

| | site | dbh | height |
|----------|------|---------------|---------------|
| Min. : | 1.0 | Min. : 5.06 | Min. :13.40 |
| 1st Qu.: | 1.0 | 1st Qu.:17.69 | 1st Qu.:29.68 |
| Median : | 2.0 | Median :28.62 | Median :36.55 |
| Mean : | 2.7 | Mean :27.88 | Mean :36.51 |
| 3rd Qu.: | 4.0 | 3rd Qu.:38.97 | 3rd Qu.:43.33 |
| Max. : | 10.0 | Max. :49.92 | Max. :59.30 |

What's the relationship between DBH and height?



First step: linear regression (lm)

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

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

Center continuous variables

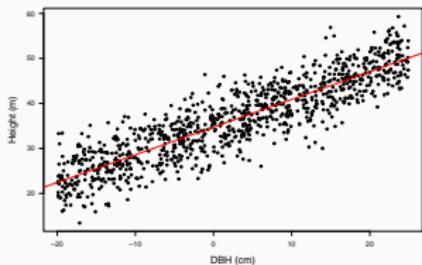
```
summary(trees$dbh)
```

| Min. | 1st Qu. | Median | Mean | 3rd Qu. | Max. |
|------|---------|--------|-------|---------|-------|
| 5.06 | 17.69 | 28.62 | 27.88 | 38.97 | 49.92 |

```
trees$dbh.c <- trees$dbh - 25
```

So, all parameters will be referred to a 25 cm DBH tree.

Linear regression with centred DBH



```
lm(formula = height ~ dbh.c, data = trees)
            coef.est  coef.se
(Intercept) 34.73      0.13
dbh.c        0.62      0.01
---
n = 1000, k = 2
residual sd = 4.09, R-Squared = 0.79
```

Let's make it Bayesian

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 = 25 cm

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

Defining model formula

```
library("brms")  
  
height.formu <- brmsformula(height ~ dbh.c)
```

We must define **prior distributions** for every parameter

brms default priors

```
get_prior(height.formu, data = trees)
```

```
  prior     class  coef group resp dpar npar bound
  (flat)      b
  (flat)      b dbh.c
student_t(3, 36.5, 10.2) Intercept
  student_t(3, 0, 10.2)    sigma
  source
  default
(vectorized)
  default
  default
```

Choosing priors

Avoid 'non-informative' priors

Use *weakly informative* (e.g. relatively wide Normal or t-student distributions)

or *strongly informative* priors based on previous knowledge and common sense.

Some tips for setting priors:

- <https://github.com/stan-dev/stan/wiki/Prior-Choice-Recommendations>

Run **prior predictive checks** (just priors, no data)

Choosing priors

Avoid 'non-informative' priors

Use *weakly informative* (e.g. relatively wide Normal or t-student distributions)

or *strongly informative* priors based on previous knowledge and common sense.

Some tips for setting priors:

- <https://github.com/stan-dev/stan/wiki/Prior-Choice-Recommendations>
- http://www.mrc-bsu.cam.ac.uk/wp-content/uploads/bugsbook_chapter5.pdf

Run **prior predictive checks** (just priors, no data)

Choosing priors

Avoid 'non-informative' priors

Use *weakly informative* (e.g. relatively wide Normal or t-student distributions)

or *strongly informative* priors based on previous knowledge and common sense.

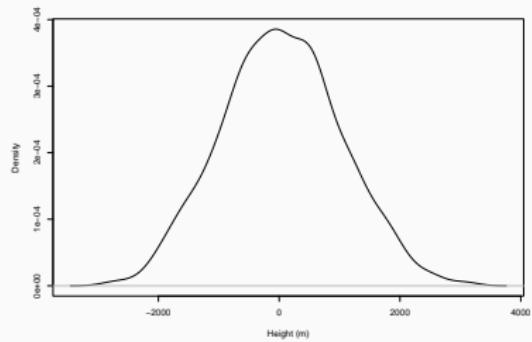
Some tips for setting priors:

- <https://github.com/stan-dev/stan/wiki/Prior-Choice-Recommendations>
- http://www.mrc-bsu.cam.ac.uk/wp-content/uploads/bugsbook_chapter5.pdf
- <https://doi.org/10.1111/oik.05985>

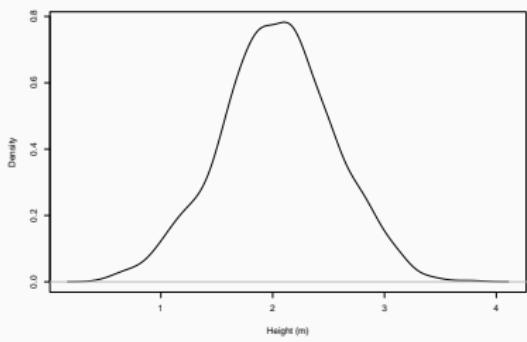
Run **prior predictive checks** (just priors, no data)

Example: estimating people height across countries

Unreasonable prior



Reasonable prior



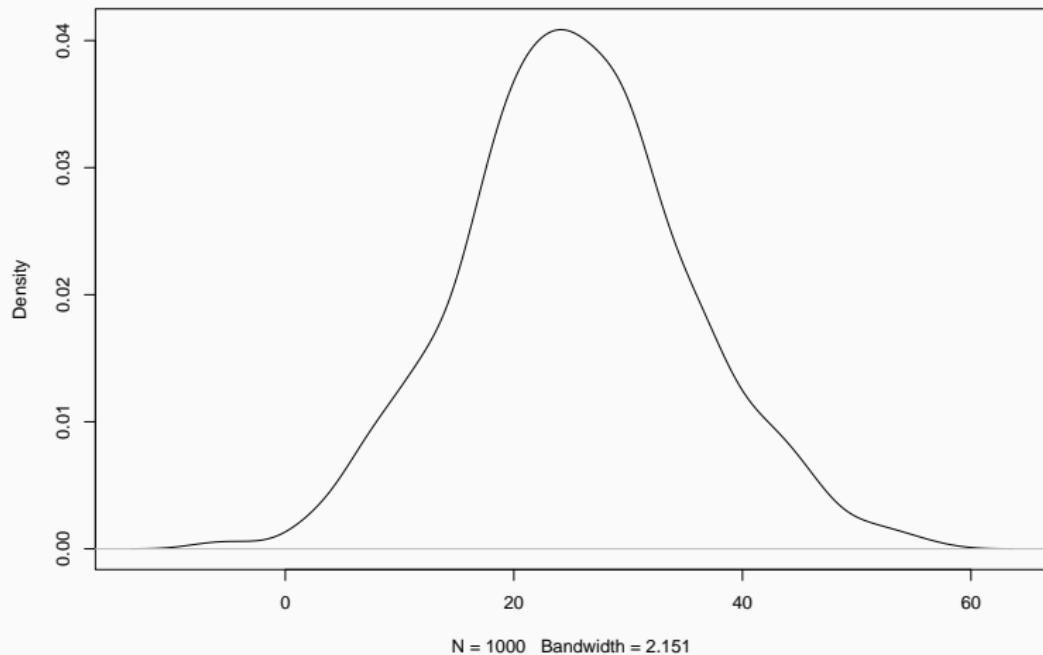
Defining priors for our trees example

```
priors <- c(  
  set_prior("normal(30, 10)", class = "Intercept"),  
  set_prior("normal(0.5, 0.4)", class = "b"),  
  set_prior("normal(0, 5)", class = "sigma")  
)
```

Prior for intercept (average height of 25-cm diameter tree)

```
plot(density(rnorm(1000, 25, 10)))
```

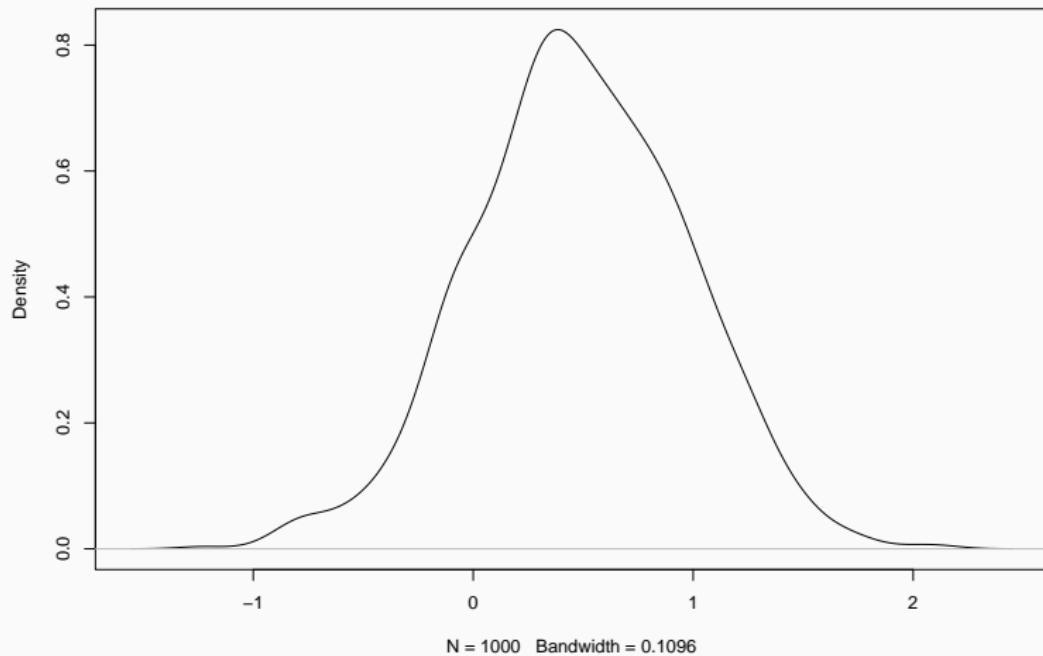
```
density.default(x = rnorm(1000, 25, 10))
```



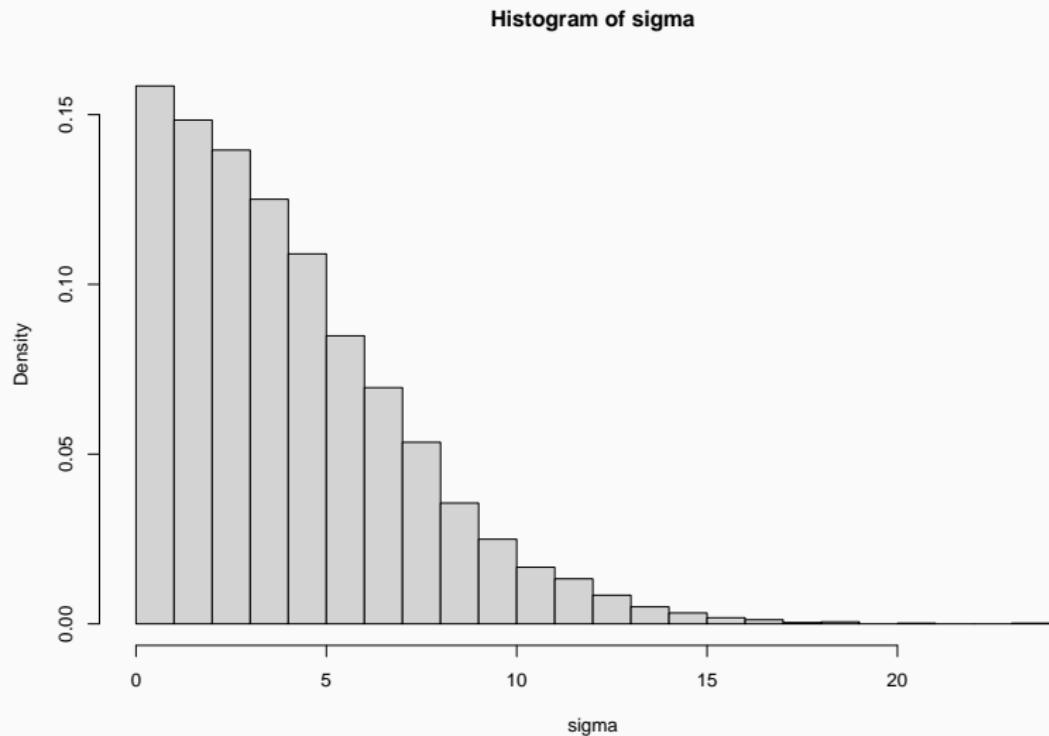
Prior for slope

```
plot(density(rnorm(1000, 0.5, 0.5)))
```

density.default(x = rnorm(1000, 0.5, 0.5))



Prior for sigma (residual sd)

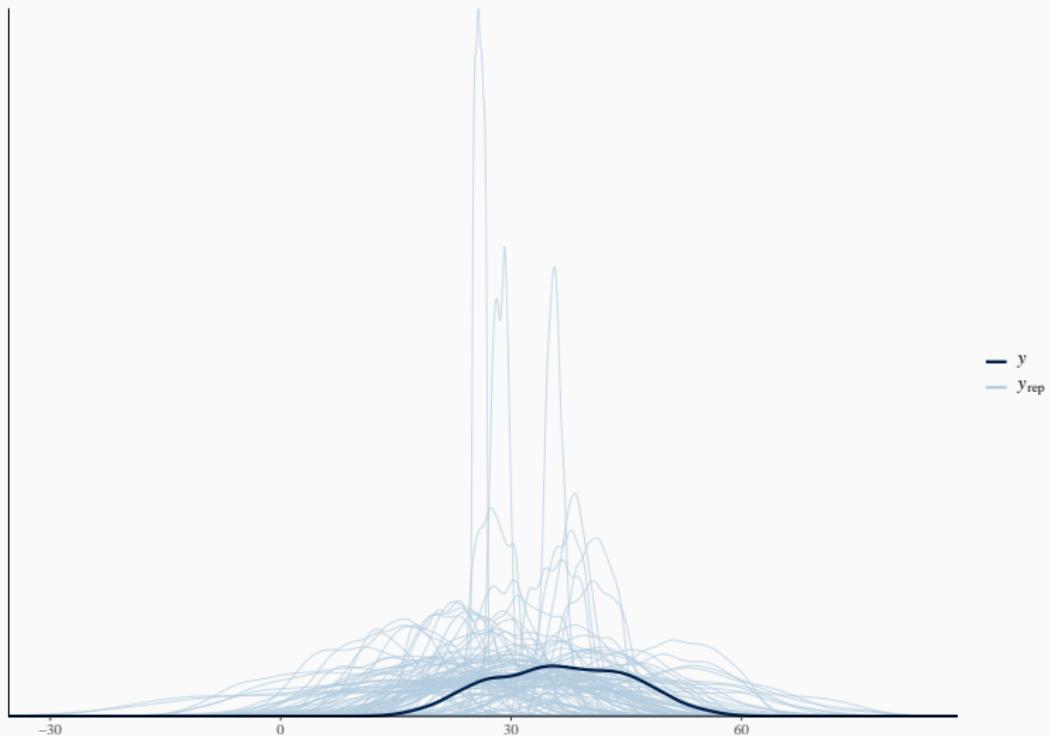


Prior predictive check

```
height.mod <- brm(height.formu,  
                     data = trees,  
                     prior = priors,  
                     sample_prior = "only")
```

Prior predictive check

```
pp_check(height.mod, ndraws = 100)
```



Fit model (now with data)

```
height.mod <- brm(height.formu,  
                     data = trees,  
                     prior = priors)
```

Model summary

```
summary(height.mod)
```

```
Family: gaussian
Links: mu = identity; sigma = identity
Formula: height ~ dbh.c
Data: trees (Number of observations: 1000)
Draws: 4 chains, each with iter = 2000; warmup = 1000; thin = 1;
       total post-warmup draws = 4000
```

Population-Level Effects:

| | Estimate | Est.Error | l-95% CI | u-95% CI | Rhat | Bulk_ESS | Tail_ESS |
|-----------|----------|-----------|----------|----------|------|----------|----------|
| Intercept | 34.73 | 0.13 | 34.47 | 34.98 | 1.00 | 3620 | 2251 |
| dbh.c | 0.62 | 0.01 | 0.60 | 0.64 | 1.00 | 4523 | 3133 |

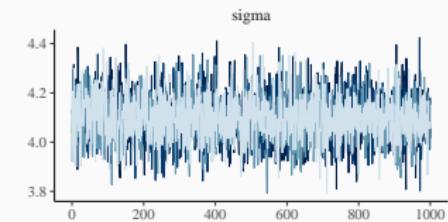
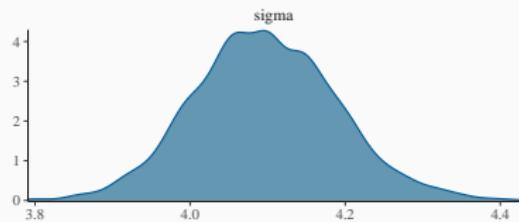
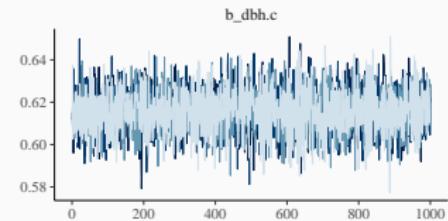
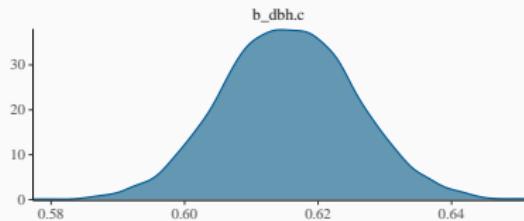
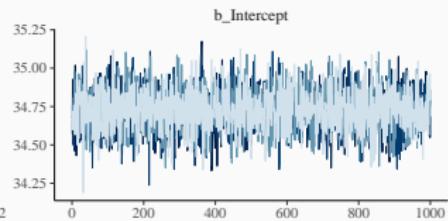
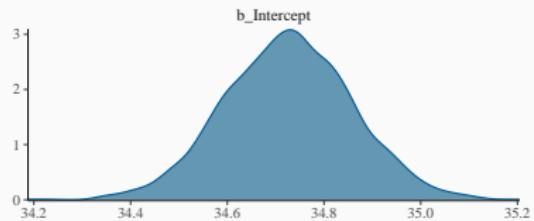
Family Specific Parameters:

| | Estimate | Est.Error | l-95% CI | u-95% CI | Rhat | Bulk_ESS | Tail_ESS |
|-------|----------|-----------|----------|----------|------|----------|----------|
| sigma | 4.10 | 0.09 | 3.92 | 4.29 | 1.00 | 4176 | 2913 |

Draws were sampled using `sampling(NUTS)`. For each parameter, Bulk_ESS and Tail_ESS are effective sample size measures, and Rhat is the potential scale reduction factor on split chains (at convergence, Rhat = 1).

Model visualisation

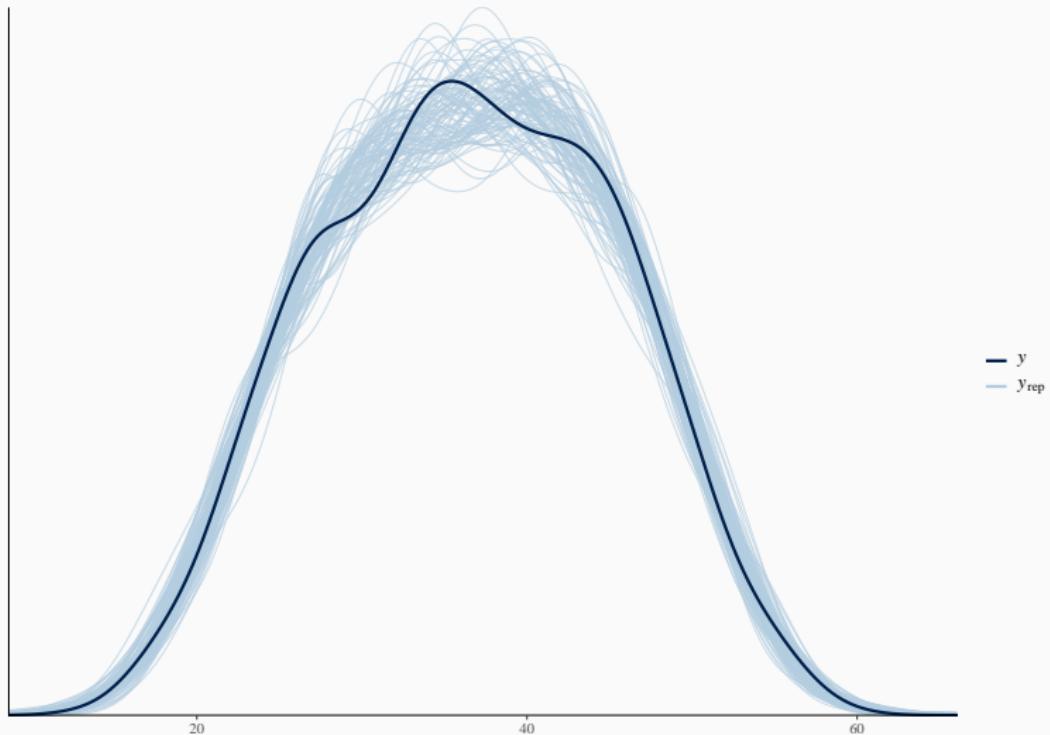
```
plot(height.mod)
```



Chain
— 1
— 2
— 3
— 4

Posterior predictive checking

```
pp_check(height.mod, ndraws = 100)
```



Interactive model exploration

```
library("shinystan")
launch_shinystan(height.mod)
```

Exercise

height ~ sex