ENV 710

mixed models

roadmap

where we are

- recap

multivariate linear models

↓

interactions
centering/scaling explanatory
variables
random effects and mixed models

↓

general*ize*d linear models

1 – height & education

Do earnings increase with the height and education or interaction between height and education for a worker?

```
summary(c6)

Call:
lm(formula = earn ~ ht.c * ed.c, data = ht.dat)

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 23020.45     518.59  44.390  < 2e-16 ***
ht.c         1121.57     135.08   8.303 2.74e-16 ***
ed.c         2546.81     215.48  11.819  < 2e-16 ***
ht.c:ed.c     138.16      53.36   2.589  0.00974 **
---
```
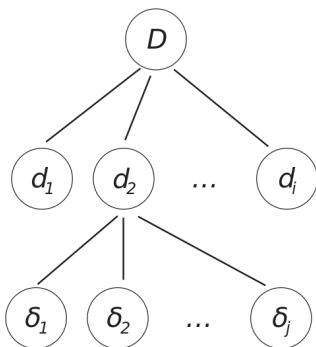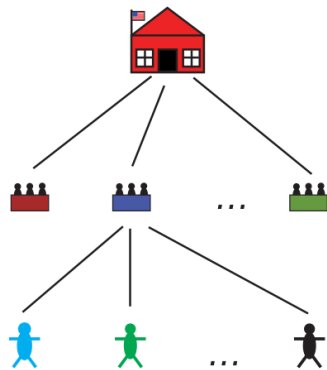
```
summary(c2)

Call:
lm(formula = earn ~ height * ed, data = ht.dat)


Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 38433.60   48668.60   0.790  0.42986
height       -744.22     726.70  -1.024  0.30599
ed          -6699.50    3581.35  -1.871  0.06164 .
height:ed     138.16      53.36   2.589  0.00974 **
---
```

mixed or multilevel models

mixed model: statistical model containing both fixed effects and random effects

accounts for (1) variation that is explained by the independent variables of interest – fixed effects, and (2) variation that is not explained by the independent variables of interest – random effects.

fixed effect factor if data has been collected from all levels of interest (e.g., gender: male/female, condition: easy/medium/hard)

random effect factor if the variable has a bunch of possible levels but you only sample a random collection (e.g., subjects, stimuli, classrooms) and though these samples will have some idiosyncrasies, you generally won't care about them, with the goal of generalizing to the broader population (e.g., all people, all scenarios, all classrooms)

fixed and random effects

am I interested in the variable (vs. just want to account for the correlation)?

- **yes** = fixed effect
- **no** = random effect

e.g., block effect
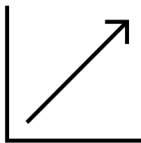
do I want an effect size for every level of the variable (vs. just want a single effect size)?

- **yes** = fixed effect
- **no** = random effect

e.g., 195 countries

do I want to make inference to an entire population (vs. interested in specific subjects)?

- **no** = fixed effect
- **yes** = random effect

e.g., random selection of tree species to study phenology

example

sleep study

The sleep study measures the average reaction time per day for subjects in a sleep deprivation study. On day 0-2 the subjects had their normal amount of sleep. Starting night 3, they were restricted to 3 hours of sleep per night. Observations represent the average reaction time (ms) on a series of tests given each day to each subject.

n = 18 subjects over 10 days (e.g., 10 measurements for each subject)

- run 4 models & discuss meaning of each model
- assess the variance explained by each model
- assess whether random effects for both the intercept (subjects) and slope (days) are necessary
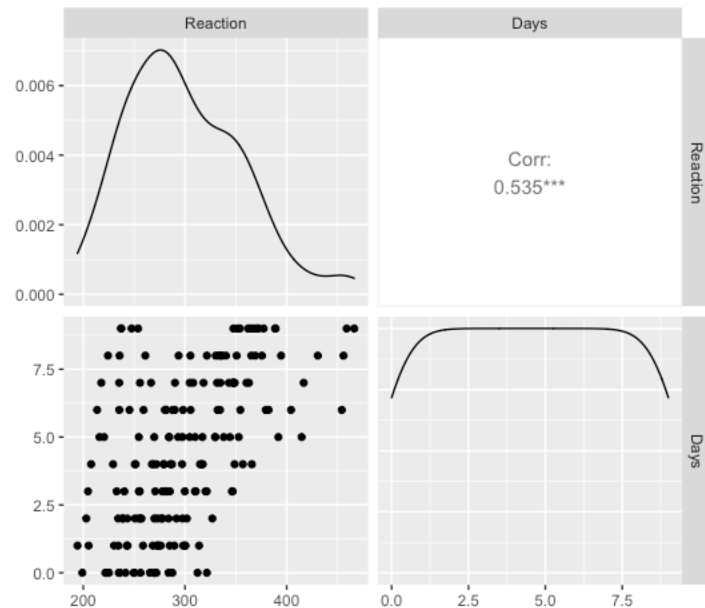
example

sleep study

```
require(lme4)
data(sleepstudy)
ss <- sleepstudy[sleepstudy$Days > 2,]
```

Run the "pooled" model – no effect of subject

```
mod1 <- lm(Reaction ~ Days, ss)
summary(mod1)

lm(formula = Reaction ~ Days, data = ss)

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  247.702     15.015   16.50  < 2e-16 ***
Days          11.063      2.374    4.66 8.03e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 '

Residual standard error: 53.3 on 124 degrees of freedom
Multiple R-squared:  0.149,      Adjusted R-squared:  0.1422
F-statistic: 21.71 on 1 and 124 DF,  p-value: 8.032e-06
```
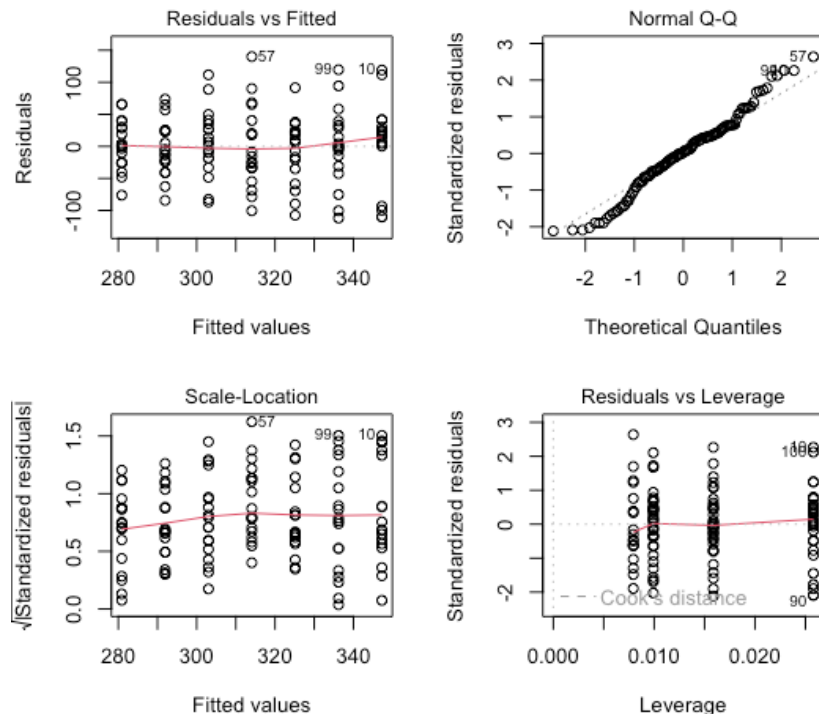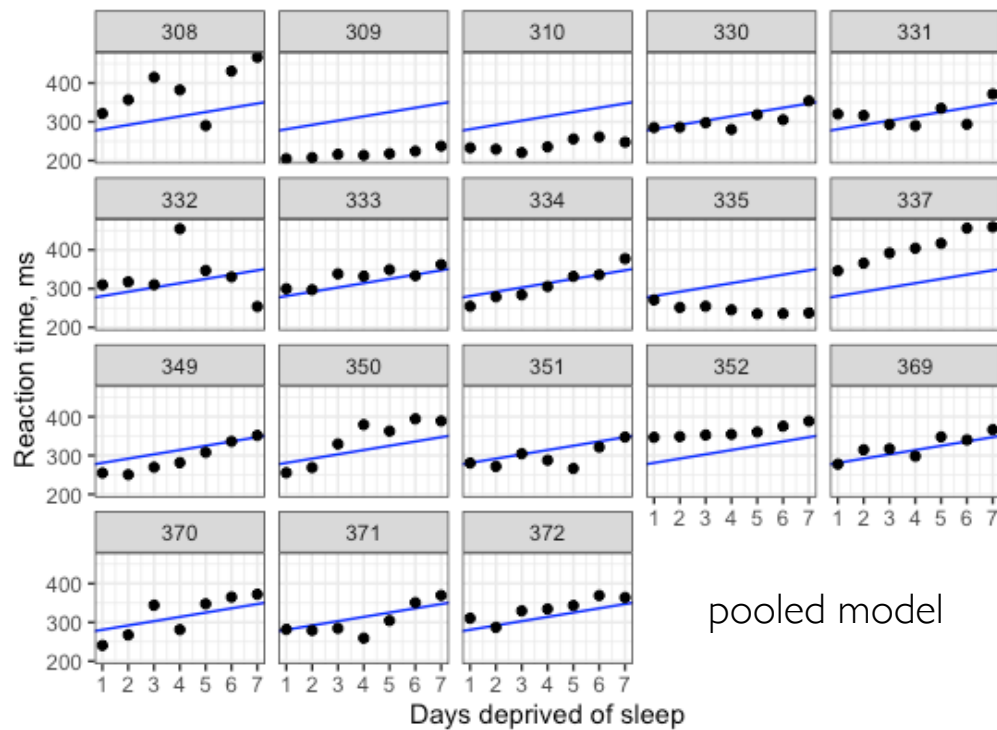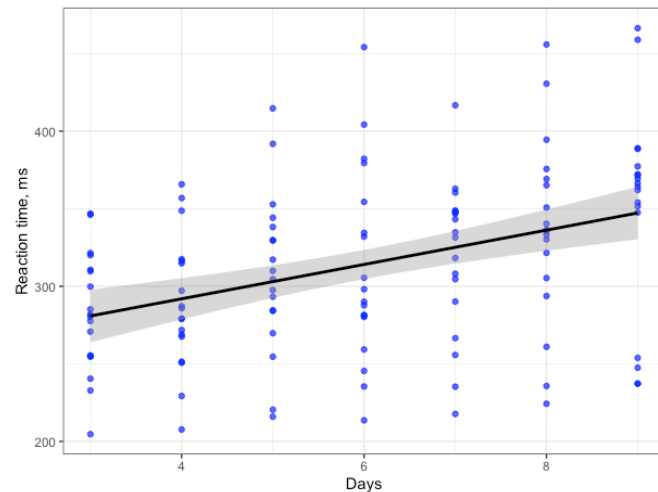
example

sleep study

pooled model

example   sleep study

Run the "unpooled model" as mod2

```
mod2 <- lm(Reaction ~ Days + Subject, ss)
 summary(mod2)
```

```
lm(formula = Reaction ~ Days + Subject, data = ss)

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  313.945      13.703  22.912  < 2e-16 ***
Days          11.063       1.319   8.391 2.16e-13 ***
Subject309  -162.989      15.822 -10.301  < 2e-16 ***
Subject310  -139.995      15.822  -8.848 2.04e-14 ***
…
Subject371   -76.094      15.822  -4.809 4.98e-06 ***
Subject372   -46.271      15.822  -2.924 0.004214 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' '

Residual standard error: 29.6 on 107 degrees of freedom
Multiple R-squared:  0.7735,     Adjusted R-squared:  0.7354
F-statistic:  20.3 on 18 and 107 DF,  p-value: < 2.2e-16
```
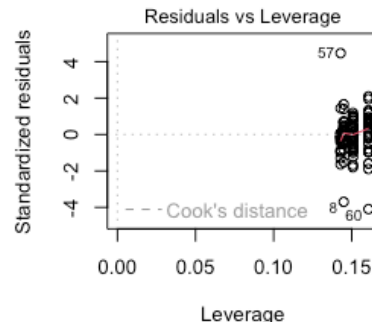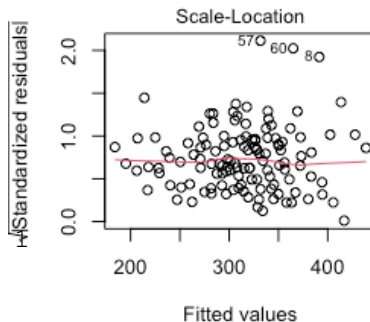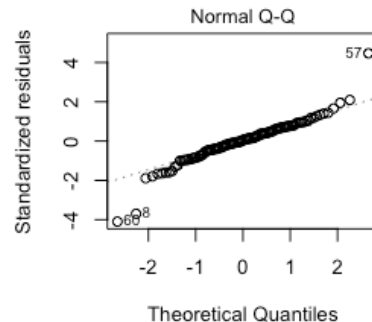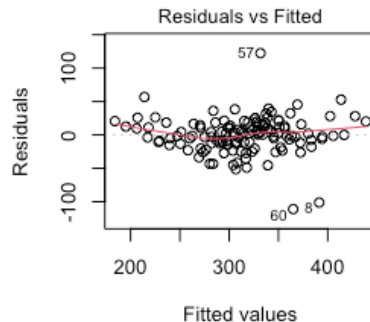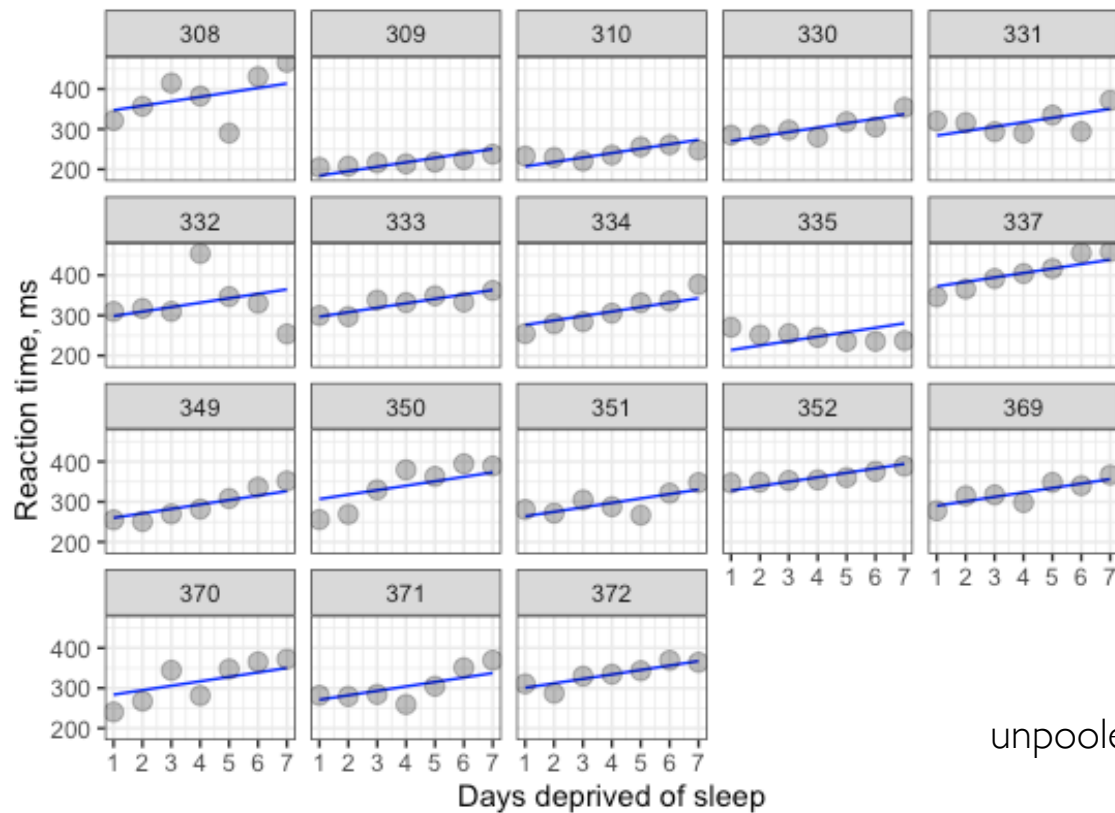
example   sleep study

unpooled model – subject treated as a factor

example   sleep study

Run the "unpooled model" as mod2a

```
mod2a <- lm(Reaction ~ Days * Subject, ss)
 summary(mod2a)
```

```
Call:
lm(formula = Reaction ~ Days * Subject, data = ss)

Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)        282.253     31.380   8.995 3.53e-14 ***
Days                16.345      4.962   3.294  0.00141 **
Subject370         -92.616     44.378  -2.087  0.03972 *
Subject371         -69.389     44.378  -1.564  0.12142
Subject372         -20.633     44.378  -0.465  0.64309
…
Days:Subject370      4.866      7.017   0.693  0.48980
Days:Subject371     -1.117      7.017  -0.159  0.87383
Days:Subject372     -4.273      7.017  -0.609  0.54409
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 26.25 on 90 degrees of freedom
Multiple R-squared:  0.8501,    Adjusted R-squared:  0.7919
F-statistic: 14.59 on 35 and 90 DF,  p-value: < 2.2e-16
```
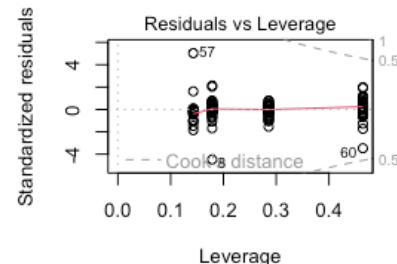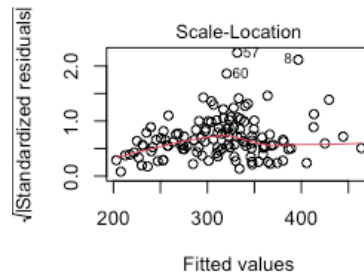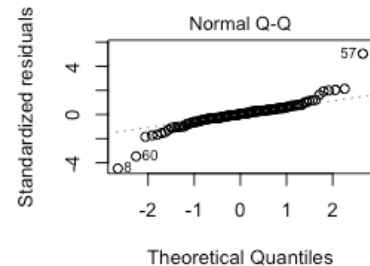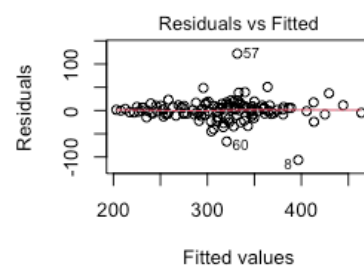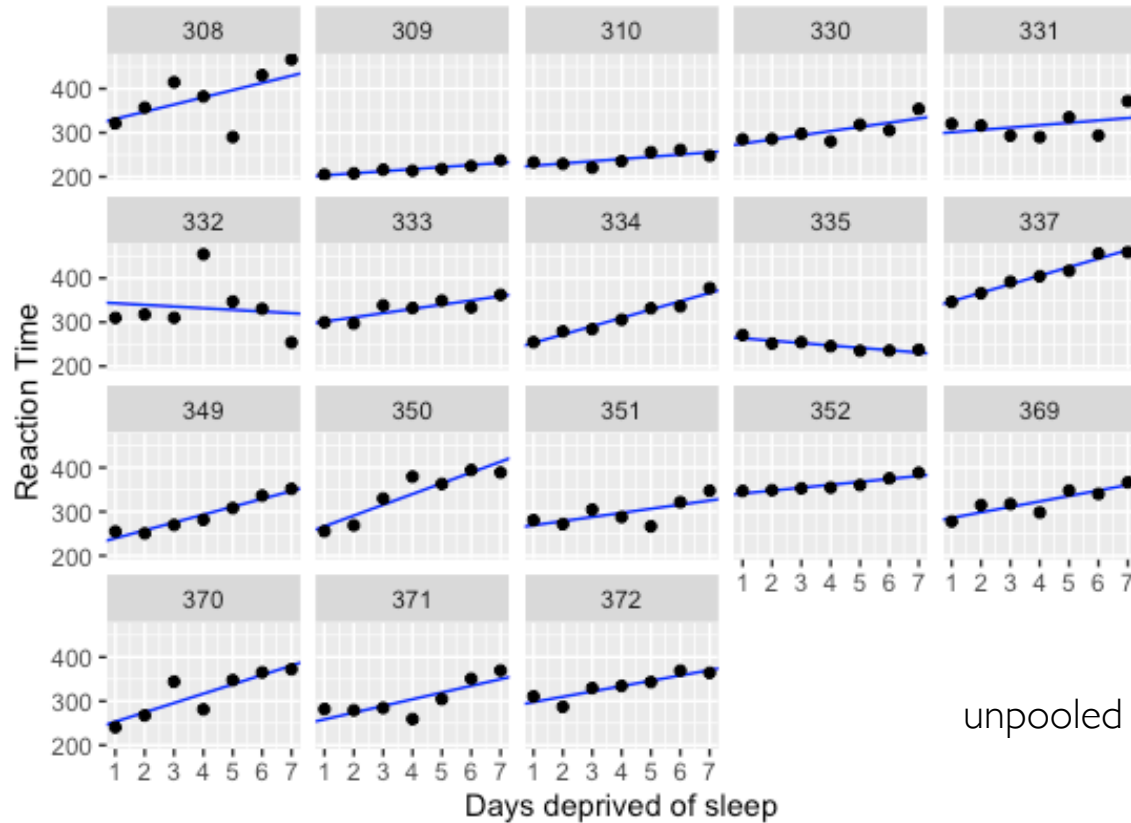
example

sleep study

unpooled model – interaction

example sleep study

Run the "varying intercept model" as mod3

```
mod3 <- lmer(Reaction ~ Days + (1|Subject), ss)
 summary(mod3)

Linear mixed model fit by REML. t-tests use Satterthwaite's method
lmerModLmerTest]
Formula: Reaction ~ Days + (1 | Subject)
   Data: ss

REML criterion at convergence: 1251.6

Random effects:
 Groups    Name         Variance Std.Dev.
 Subject  (Intercept) 2047.1    45.25
 Residual               876.2    29.60
Number of obs: 126, groups:  Subject, 18

Fixed effects:
            Estimate Std. Error      df t value Pr(>|t|)
(Intercept)  247.702     13.538  37.598  18.297  < 2e-16 ***
Days          11.063      1.319 107.000   8.391 2.16e-13 ***
---
```

plot(mod3)
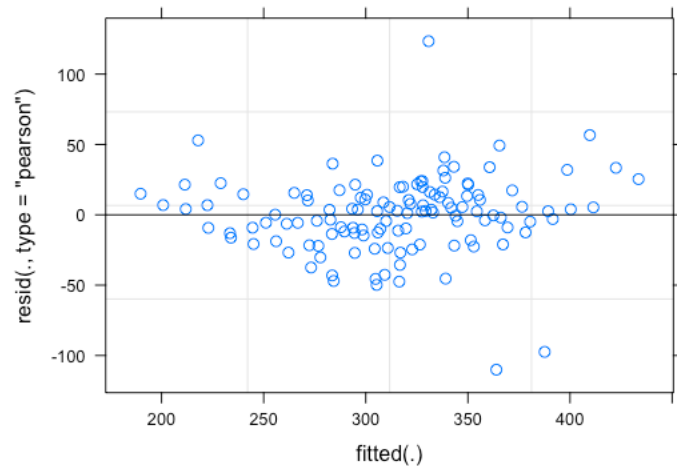
example

sleep study

```
require(lattice)
qqmath(mod3, id = 0.05)
```

```
plot(mod3, sqrt(abs(resid(.))) ~ fitted(.),
     type = c("p", "smooth"))
```

varying intercept model

example   sleep study

Reaction time, ms — Days deprived of sleep
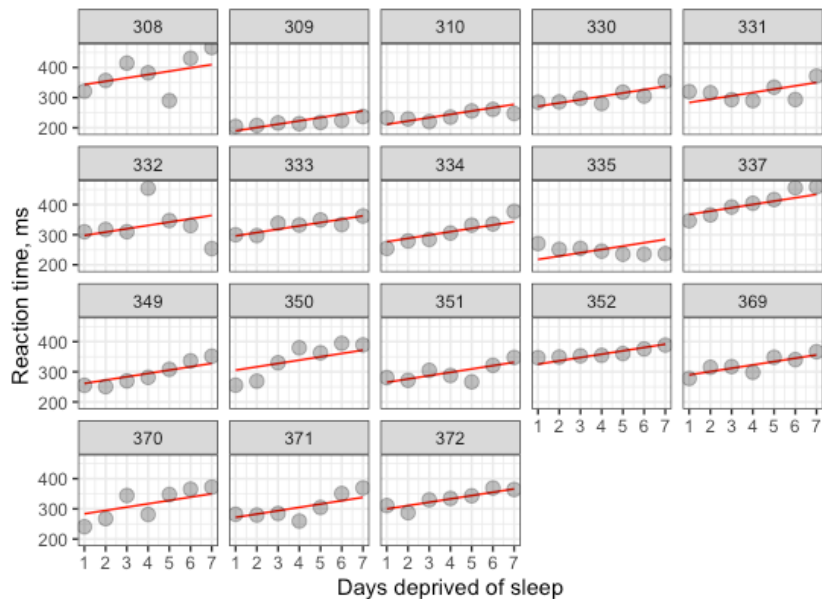
```
> re3 <- ranef(mod3)          > int <- fixef(mod3)[1] + re3$Subject

$Subject                              (Intercept)
    (Intercept)              308   310.1284
308   62.426297              309   156.5314
309  -91.170662              310   178.2000
310  -69.502118              330   237.9882
330   -9.713839              331   250.5780
331    2.875942              332   264.3711
332   16.668992              333   262.8942
333   15.192070              334   243.5842
334   -4.117917              335   184.7088
335  -62.993300              337   333.9777
337   86.275666              349   228.1046
349  -19.597461              350   272.1368
350   24.434740              351   231.8069
351  -15.895221              352   291.9799
352   44.277831              369   256.3416
369    8.639473              370   250.3623
370    2.660191              371   238.4195
371   -9.282580              372   266.5240
372   18.821896
```

example   sleep study

Run the "varying intercept/slope model" as mod4

```
mod4 <- lmer(Reaction ~ Days + (1 + Days|Subject), ss)
 summary(mod4)
```

```
Linear mixed model fit by REML. t-tests use Satterthwaite's method [
lmerModLmerTest]
Formula: Reaction ~ Days + (1 + Days | Subject)
   Data: ss

REML criterion at convergence: 1238.8

Random effects:
 Groups     Name         Variance Std.Dev. Corr
 Subject   (Intercept)  1473.60   38.387
           Days            42.03    6.483   -0.31
 Residual               689.25   26.254
Number of obs: 126, groups:  Subject, 18

Fixed effects:
            Estimate Std. Error      df t value Pr(>|t|)
(Intercept)  247.702     11.686  16.994   21.20 1.16e-13 ***
Days          11.063      1.924  16.995    5.75 2.36e-05 ***
---
```
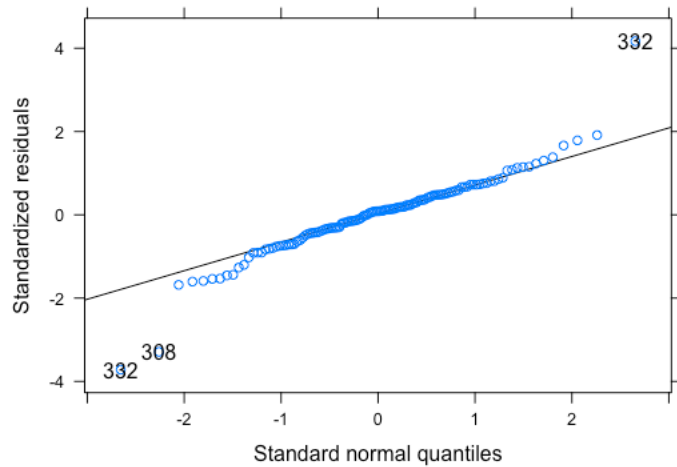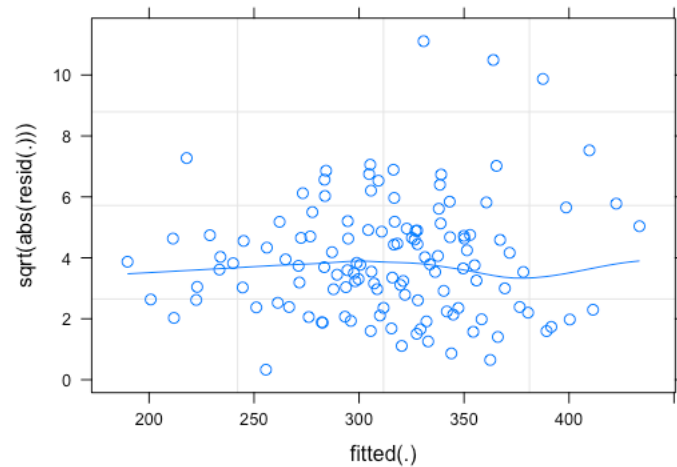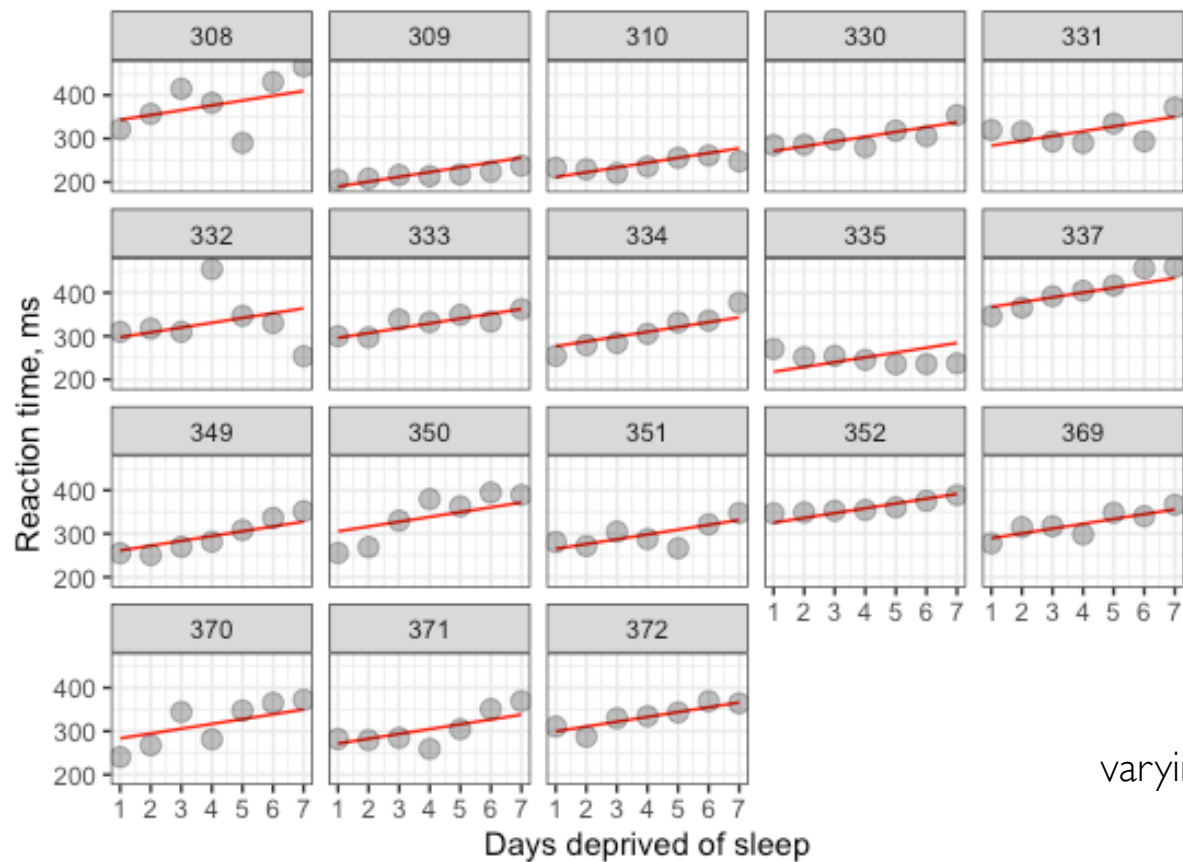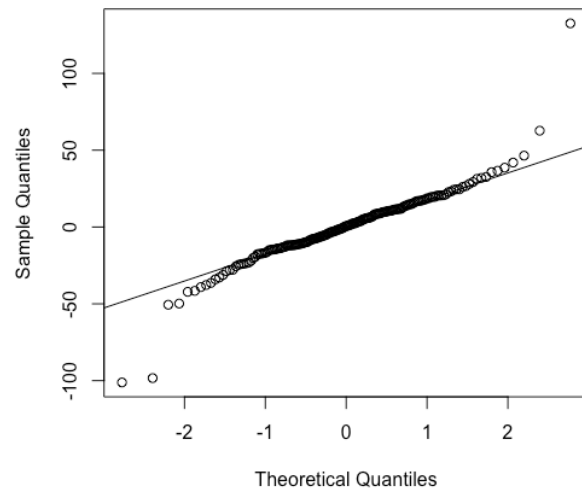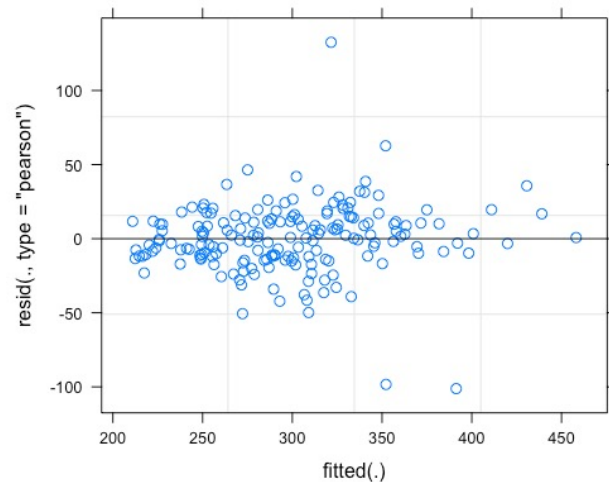
varying intercept &
slope model

{merTools}
plotREsim(REsim(mod4))

```
> AIC(mod1, mod2, mod2a, mod3, mod4)
       df      AIC
mod1    3 1363.488
mod2   20 1230.703
mod2a  37 1212.670
mod3    4 1259.646
mod4    6 1250.799
```
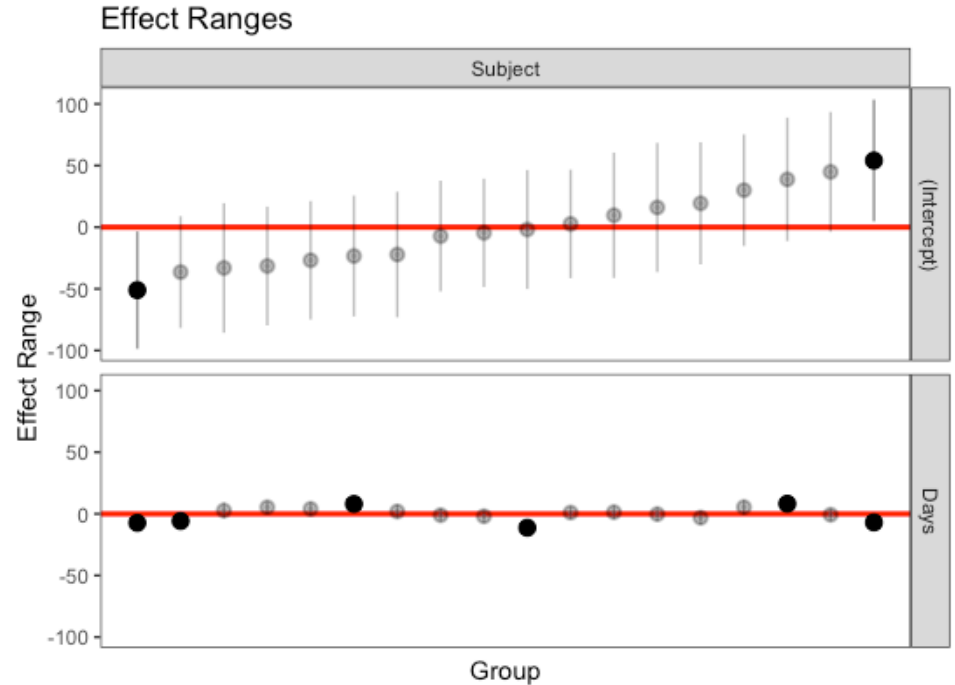
{MuMIn}
r.squaredGLMM(mod3)
```
 R2m       R2c
0.1444303 0.743562
```

r.squaredGLMM(mod4)
```
 R2m       R2c
0.1441022  0.7987363
```


Effect Ranges

example   sleep study

Run the "varying intercept/slope model" as mod4

```
{car}
outlierTest(mod4)
```

```
        rstudent      unadj p-value    Bonferroni p
57      5.147059      1.0576e-06       0.00013326
8      -4.316780      3.2938e-05       0.00415020
60     -4.059410      8.8413e-05       0.01114000
```
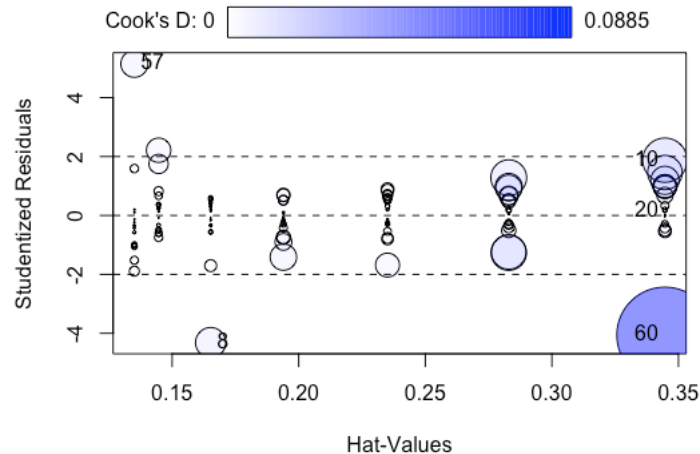
check out extreme residuals

```
sort(abs(resid(mod4)))
```

```
influencePlot(mod4)
    StudRes          Hat          CookD
8   -4.3167800   0.1652161   0.008768417
60  -4.0594101   0.3446554   0.088474558
80   0.9892771   0.3446554   0.005254465
10   1.8766544   0.3446554   0.018908677
57   5.1470591   0.1350741   0.007011464
```

example   sleep study

Remove extreme observations

```
ss1 <- ss[ss$resid4 > -80 & ss$resid4 < 50,]
mod5 <- lmer(Reaction ~ Days + (1 + Days|Subject), ss1)
```

```
Formula: Reaction ~ Days + (1 + Days | Subject)
   Data: ss1

REML criterion at convergence: 1121.2

Random effects:
 Groups    Name         Variance Std.Dev. Corr
 Subject   (Intercept) 1382.06   37.176
           Days          49.76    7.054   -0.30
 Residual               289.41   17.012
Number of obs: 122, groups:  Subject, 18

Fixed effects:
            Estimate Std. Error      df t value Pr(>|t|)
(Intercept)  242.559     10.013  17.005  24.224 1.27e-14 ***
Days          11.981      1.833  17.047   6.537 5.00e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```
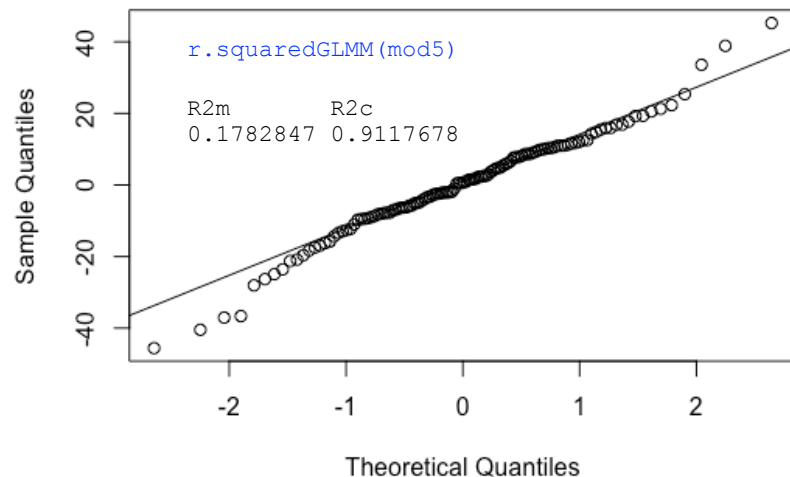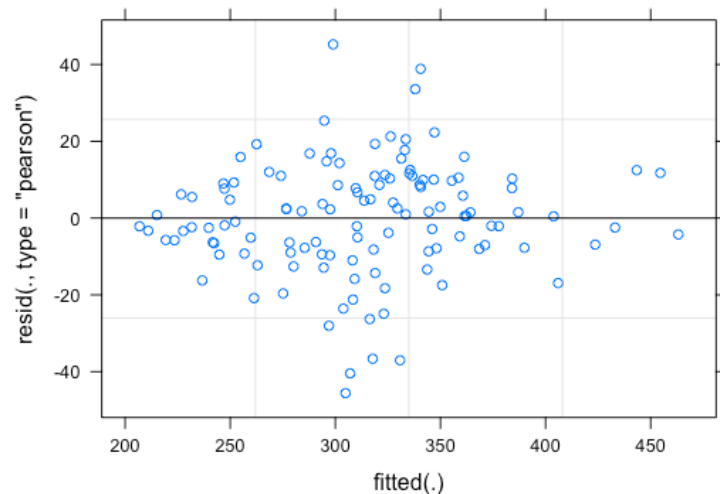

r.squaredGLMM(mod5)

```
R2m         R2c
0.1782847   0.9117678
```

a. pooled model
     `y ~ x`

b. no pooling model
     `y ~ x +` subject

c. varying intercept model
     `y ~ x + (1|subject)`

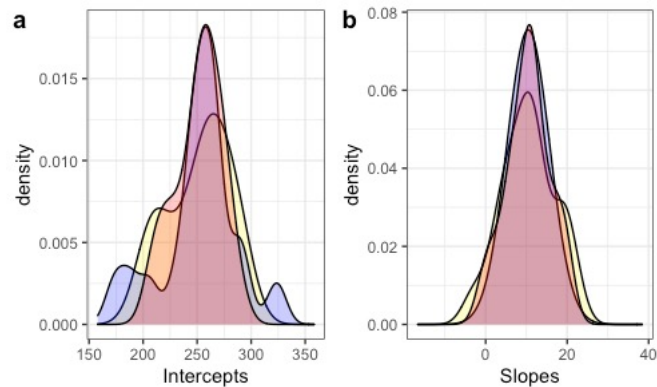d. varying intercept, varying slope model
     `y ~ x + (1 + x|subject)`
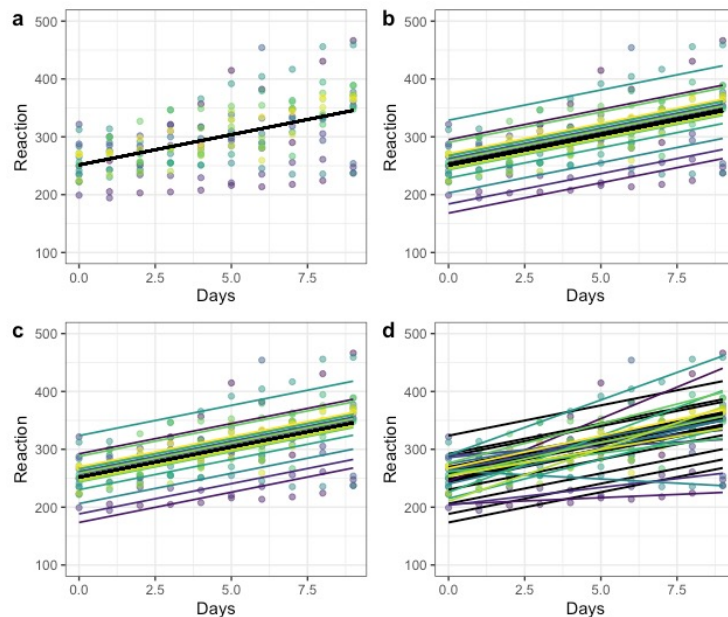
"Shrinkage"

a. intercept estimates

b. slope estimates

yellow = no pooling,
blue = varying intercept
red = varying intercept & slope

```
ch1 <- lmer(weight ~ Diet + (1|Chick), data = ChickWeight)

{emmeans}
> emmeans(ch1, list(pairwise ~ Diet), adjust = "tukey")
```

```
$`emmeans of Diet`
 Diet emmean   SE   df lower.CL upper.CL
 1       102 6.04 47.0     89.5      114
 2       123 8.26 44.0    106.0      139
 3       143 8.26 44.0    126.3      160
 4       135 8.31 44.8    118.3      152

Degrees-of-freedom method: kenward-roger
Confidence level used: 0.95

$`pairwise differences of Diet`
 1     estimate   SE   df t.ratio p.value
 1 - 2   -20.98 10.2 45.0  -2.049  0.1856
 1 - 3   -41.32 10.2 45.0  -4.035  0.0012
 1 - 4   -33.40 10.3 45.5  -3.252  0.0112
 2 - 3   -20.33 11.7 44.0  -1.740  0.3159
 2 - 4   -12.42 11.7 44.4  -1.060  0.7152
 3 - 4     7.91 11.7 44.4   0.675  0.9059

Degrees-of-freedom method: kenward-roger
P value adjustment: tukey method for comparing a family of 4 estimates
```

```
> data(ChickWeight)
> head(ChickWeight)
Grouped Data: weight ~ Time |
Chick
  weight Time Chick Diet
1     42    0     1    1
2     51    2     1    1
3     59    4     1    1
4     64    6     1    1
5     76    8     1    1
6     93   10     1    1
```

example   chick weight

```
ch1 <- lmer(weight ~ Diet + (1|Chick), data = ChickWeight)


{multcomp}
> summary(glht(ch1, linfct = mcp(Diet = "Tukey")), test = adjusted("holm"))
```

```
              Simultaneous Tests for General Linear Hypotheses

Multiple Comparisons of Means: Tukey Contrasts


Fit: lmer(formula = weight ~ Diet + (1 | Chick), data = ChickWeight)

Linear Hypotheses:
            Estimate Std. Error z value Pr(>|z|)
2 - 1 == 0    20.983     10.236   2.050 0.161521
3 - 1 == 0    41.316     10.236   4.036 0.000326 ***
4 - 1 == 0    33.404     10.270   3.253 0.005718 **
3 - 2 == 0    20.333     11.688   1.740 0.245709
4 - 2 == 0    12.421     11.717   1.060 0.578201
4 - 3 == 0    -7.912     11.717  -0.675 0.578201
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
(Adjusted p values reported -- holm method)
```

```
> head(ChickWeight)
Grouped Data: weight ~ Time |
Chick
  weight Time Chick Diet
1     42    0     1    1
2     51    2     1    1
3     59    4     1    1
4     64    6     1    1
5     76    8     1    1
6     93   10     1    1
```

```
ch1 <- lmer(weight ~ Diet + (1|Chick), data = ChickWeight)


{multcomp}
> summary(glht(ch1, linfct = mcp(Diet = "Tukey")), test = adjusted("holm"))


                Simultaneous Tests for General Linear Hypotheses


Multiple Comparisons of Means: Tukey Contrasts


Fit: lmer(formula = weight ~ Diet + (1 | Chick), data = ChickWeight)


Linear Hypotheses:
            Estimate Std. Error z value Pr(>|z|)
2 - 1 == 0   20.983     10.236   2.050 0.161521
3 - 1 == 0   41.316     10.236   4.036 0.000326 ***
4 - 1 == 0   33.404     10.270   3.253 0.005718 **
3 - 2 == 0   20.333     11.688   1.740 0.245709
4 - 2 == 0   12.421     11.717   1.060 0.578201
4 - 3 == 0   -7.912     11.717  -0.675 0.578201
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
(Adjusted p values reported -- holm method)
```

```
> head(ChickWeight)
Grouped Data: weight ~ Time |
Chick
  weight Time Chick Diet
1     42    0     1    1
2     51    2     1    1
3     59    4     1    1
4     64    6     1    1
5     76    8     1    1
6     93   10     1    1
```

tree mass

The sleep study examines the relationship between tree biomass (kg) and its height (m), dbh (mm), and disturbance (logged+hunted, logged, none).
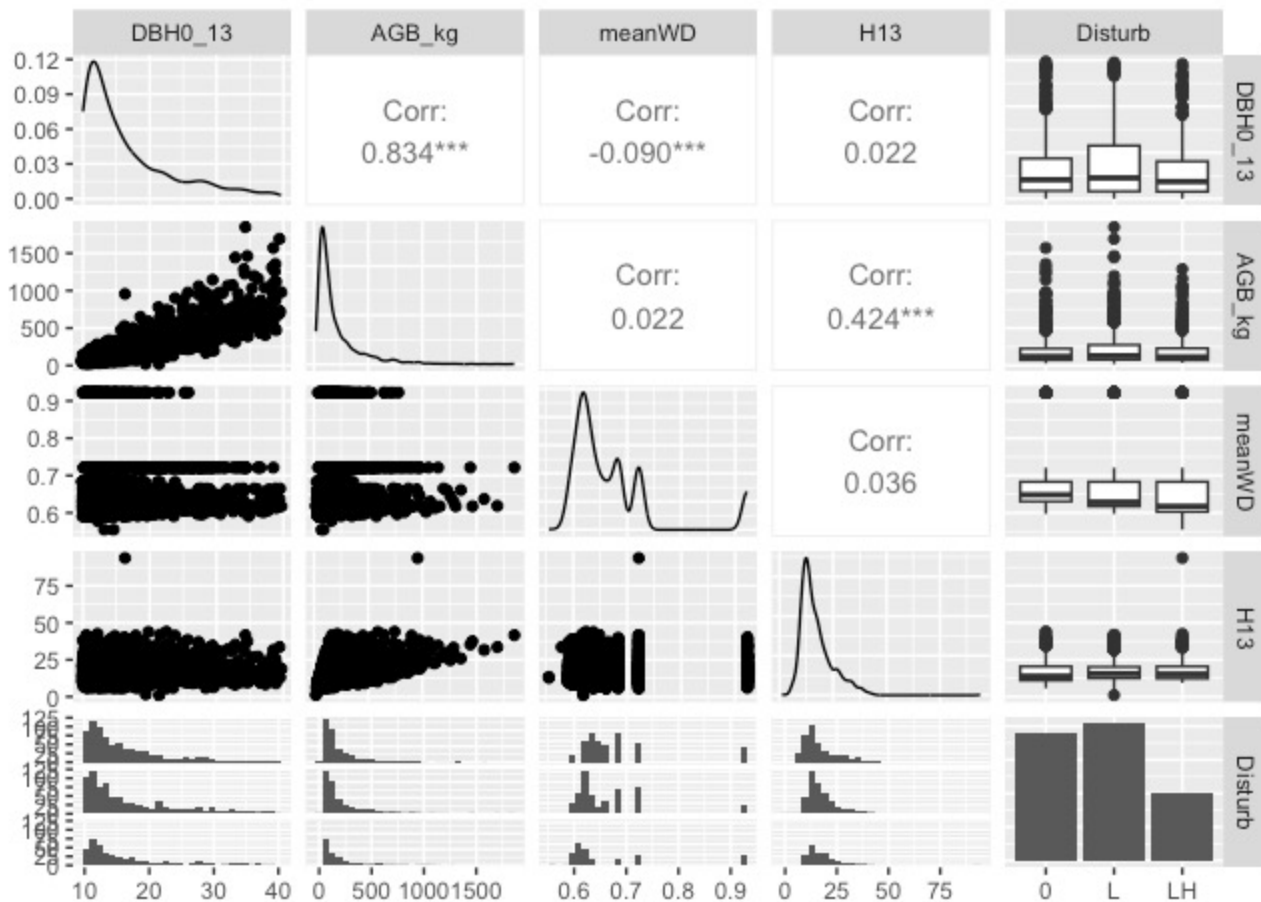
There are 1980 trees from 7 species and 30 plots.

- what type of model? why?
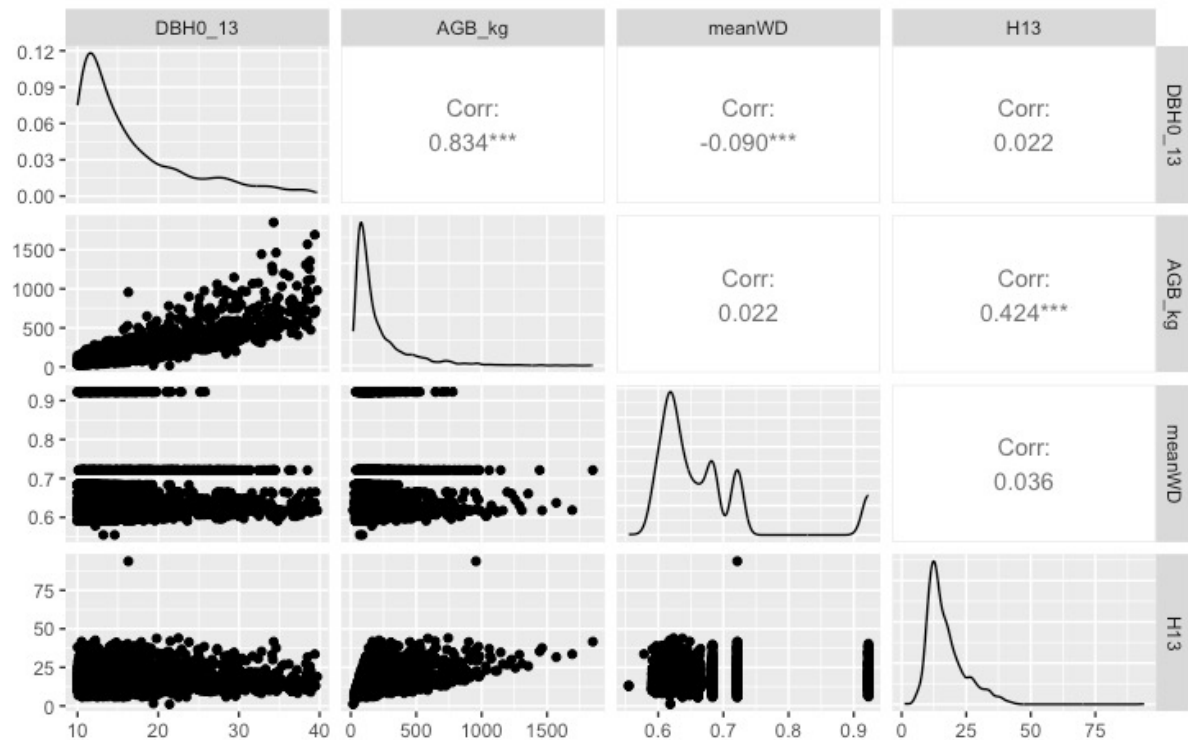- model structure?

example   tree mass


```
ggpairs(ss1, columns = c("DBH0_13",
"AGB_kg", "meanWD", "H13", "Disturb"))
```

example   tree mass

```
ggpairs(ss1, columns = c("DBH0_13",
"AGB_kg", "meanWD", "H13"))
```

example   tree mass

```
m2 <- lmer(AGB_kg ~ H13 + DBH0_13 + Disturb + (1|spp) + (1|Plot), data = ss1)



Linear mixed model fit by REML. t-tests use Satterthwaite's method [lmerModLmerTest]
Formula: AGB_kg ~ H13 + DBH0_13 + Disturb + (1 | spp) + (1 | Plot)
   Data: ss1

REML criterion at convergence: 22805.6

Random effects:
 Groups    Name         Variance  Std.Dev.
 Plot      (Intercept)   36.73     6.06
 spp       (Intercept)  547.04    23.39
 Residual              5711.30    75.57
Number of obs: 1984, groups:  Plot, 30; spp, 7

Fixed effects:
             Estimate Std. Error       df t value Pr(>|t|)
(Intercept) -424.3423    11.3617  14.4426 -37.349 8.93e-16 ***
H13           11.4182     0.2328 1921.3592  49.057  < 2e-16 ***
DBH0_13       26.0274     0.2957 1897.0509  88.012  < 2e-16 ***
DisturbL       1.7596     4.8046  24.5533   0.366    0.717
DisturbLH      4.1783     5.7263  35.0128   0.730    0.470
------
```

example   tree mass

```
performance::check_model(m2,
check = c("reqq", "normality", "linearity"))
```
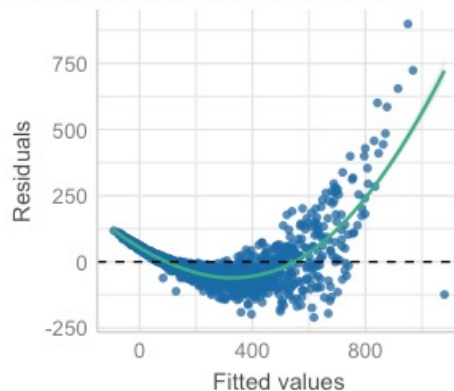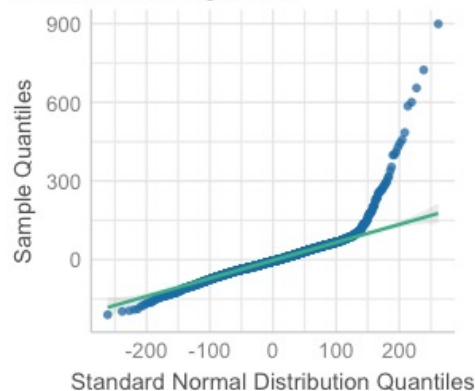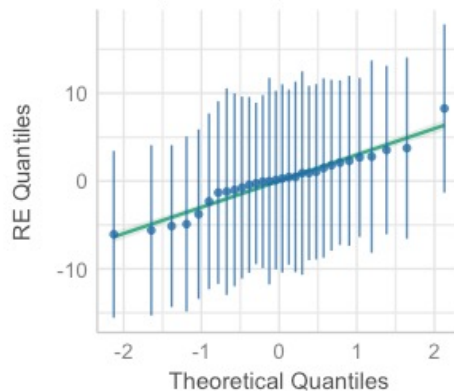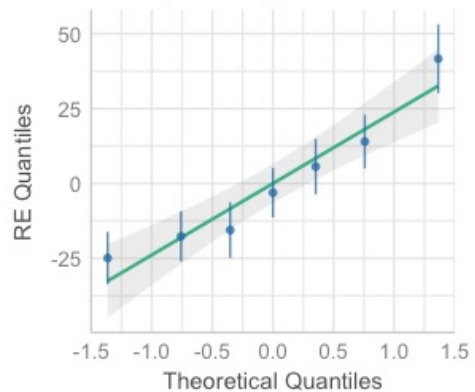
```
m3 <- lmer(AGB_kg ~ H13 + DBH0_13 + Disturb +  (1 + Plot|spp), data = ss1)
boundary (singular) fit: see help('isSingular')
```

the parameters are on the boundary of the feasible parameter space: variances of one or more linear combinations of effects are (close to) zero

when a random effect variance is estimated very near zero and (very loosely) the data is not sufficiently informative to drag the estimate away from the zero starting value

```
 m4 <- lmer(log(AGB_kg) ~ H13 + DBH0_13 + meanWD + Disturb + (1|spp) + (1|Plot), data = ss1)


Linear mixed model fit by REML. t-tests use Satterthwaite's method [lmerModLmerTest]
Formula: log(AGB_kg) ~ H13 + DBH0_13 + Disturb + (1 | spp) + (1 | Plot)
   Data: ss1

REML criterion at convergence: -1529.5

Random effects:
 Groups    Name         Variance Std.Dev.
 Plot      (Intercept) 0.001714 0.0414
 spp       (Intercept) 0.022916 0.1514
 Residual              0.025483 0.1596
Number of obs: 1984, groups:  Plot, 30; spp, 7

Fixed effects:
              Estimate Std. Error        df t value Pr(>|t|)
(Intercept)  2.459e+00  6.053e-02  7.413e+00  40.632 5.55e-10 ***
H13          5.040e-02  4.979e-04  1.972e+03 101.216  < 2e-16 ***
DBH0_13      9.925e-02  6.316e-04  1.966e+03 157.159  < 2e-16 ***
DisturbL     1.051e-02  2.043e-02  2.559e+01   0.514    0.611
DisturbLH   -2.347e-03  2.236e-02  3.099e+01  -0.105    0.917
---------
```
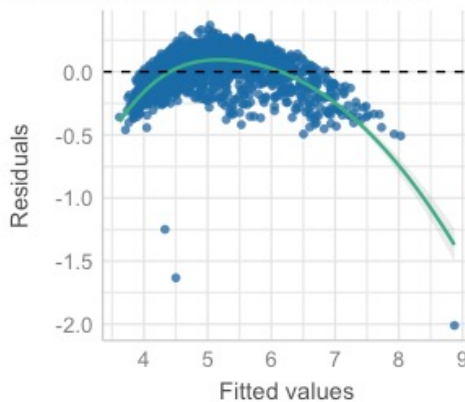
```
performance::check_model(m4,
check = c("reqq", "qq", "linearity"))
```
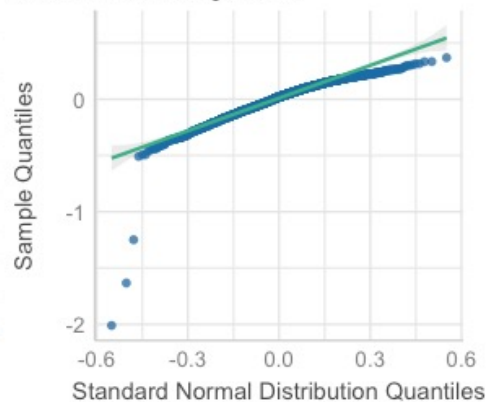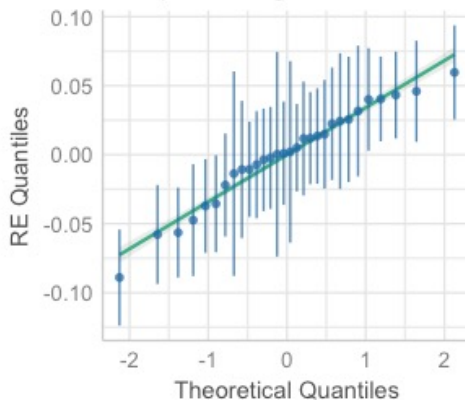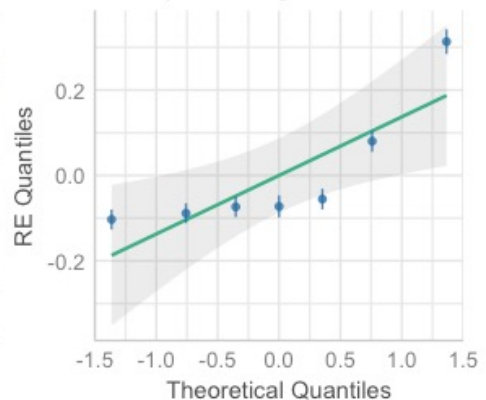
example   tree mass

```
m5 <- lmer(log(AGB_kg) ~ H13 + log(DBH0_13) + Disturb + (1|spp) + (1|Plot), data = ss1)
```

```
Linear mixed model fit by REML. t-tests use Satterthwaite's method [lmerModLmerTest
]
Formula: log(AGB_kg) ~ H13 + DBH0_13 + Disturb + (1 | spp) + (1 | Plot)
   Data: ss1

REML criterion at convergence: -1529.5

Random effects:
 Groups    Name        Variance  Std.Dev.
 Plot      (Intercept) 0.001714  0.0414
 spp       (Intercept) 0.022916  0.1514
 Residual              0.025483  0.1596
Number of obs: 1984, groups:  Plot, 30; spp, 7

Fixed effects:
              Estimate Std. Error        df t value Pr(>|t|)
(Intercept)  2.459e+00  6.053e-02  7.413e+00  40.632 5.55e-10 ***
H13          5.040e-02  4.979e-04  1.972e+03 101.216  < 2e-16 ***
DBH0_13      9.925e-02  6.316e-04  1.966e+03 157.159  < 2e-16 ***
DisturbL     1.051e-02  2.043e-02  2.559e+01   0.514    0.611
DisturbLH   -2.347e-03  2.236e-02  3.099e+01  -0.105    0.917
---------
```
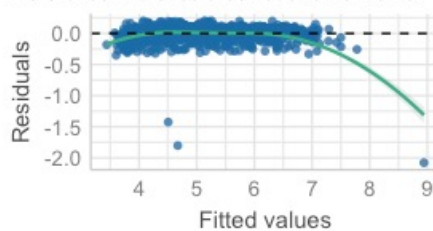
```
performance::check_model(m5,
check = c("vif", "reqq", "qq", "linearity"))
```
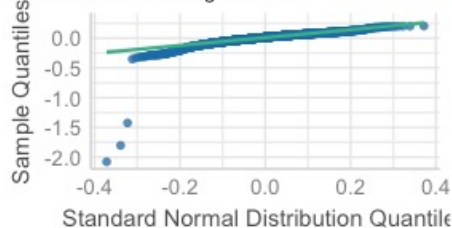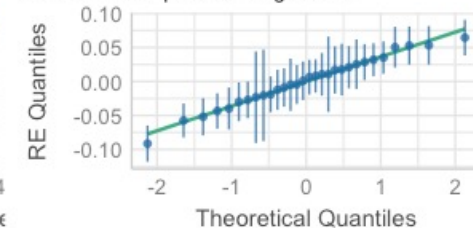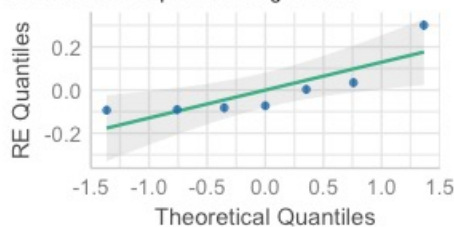
example   tree mass

```
ss1$resid <- resid(m5)
ss1[ss1$resid < -1, ]
```

```
Plot DBH0_13    meanWD    AGB13.Mg        H13       genusCorr       sppCorr
496     10    16.3 0.7213065 0.95674746 93.843623        Strombosia nigropunctata
2164    14    21.4 0.6185869 0.01760324  1.058761 Greenwayodendron    suaveolens
2361    14    19.4 0.6185869 0.02178308  1.602595 Greenwayodendron    suaveolens
        familyAPG    AGB_kg                                spp Disturb     resid
496     Olacaceae 956.74746    Strombosia nigropunctata         LH -2.077444
2164 Annonaceae   17.60324 Greenwayodendron suaveolens          L -1.803027
2361 Annonaceae   21.78308 Greenwayodendron suaveolens          L -1.424433
```

```
ss2 <- ss1[ss1$resid > -1,]

m6 <- lmer(log(AGB_kg) ~ H13 + log(DBH0_13) +
             Disturb + (1|spp) + (1|Plot), data = ss2)
```

```
> summary(m6)
Linear mixed model fit by REML. t-tests use Satterthwaite's method [lmerModLmerTest
]
Formula: log(AGB_kg) ~ H13 + log(DBH0_13) + Disturb + (1 | spp) + (1 |      Plot)
   Data: ss2

REML criterion at convergence: -4194.3

Random effects:
 Groups    Name        Variance Std.Dev.
 Plot      (Intercept) 0.001731 0.04160
 spp       (Intercept) 0.019695 0.14034
 Residual              0.006501 0.08063
Number of obs: 1981, groups:  Plot, 30; spp, 7

Fixed effects:
              Estimate Std. Error        df t value Pr(>|t|)
(Intercept)  -1.295e+00  5.750e-02  8.243e+00 -22.518 1.08e-08 ***
H13           5.121e-02  2.598e-04  1.955e+03 197.081  < 2e-16 ***
log(DBH0_13)  1.961e+00  6.199e-03  1.951e+03 316.409  < 2e-16 ***
DisturbL      3.095e-02  1.912e-02  2.451e+01   1.619    0.118
DisturbLH     1.447e-02  2.000e-02  2.767e+01   0.724    0.475
---
```

```
MuMIn::r.squaredGLMM(m6)
        R2m        R2c
[1,] 0.9581771 0.9902642
```

example   tree mass

```
> summary(m6)

Fixed effects:
              Estimate Std. Error        df t value Pr(>|t|)
(Intercept)  -1.295e+00  5.750e-02  8.243e+00 -22.518 1.08e-08 ***
H13           5.121e-02  2.598e-04  1.955e+03 197.081  < 2e-16 ***
log(DBH0_13)  1.961e+00  6.199e-03  1.951e+03 316.409  < 2e-16 ***
DisturbL      3.095e-02  1.912e-02  2.451e+01   1.619    0.118
DisturbLH     1.447e-02  2.000e-02  2.767e+01   0.724    0.475
---
```

```
> cfs <- fixef(m6)
> (exp(cfs[2]) - 1) * 100

H13
5.254165
```

Only DV is log-transformed: Exponentiate the coefficient, subtract one from this number, and multiply by 100.

For every one-unit increase in H13, AGB increases by about 5.3%.

```
> summary(m6)

Fixed effects:
                 Estimate Std. Error         df t value Pr(>|t|)
(Intercept)    -1.295e+00  5.750e-02  8.243e+00 -22.518 1.08e-08 ***
H13             5.121e-02  2.598e-04  1.955e+03 197.081  < 2e-16 ***
log(DBH0_13)    1.961e+00  6.199e-03  1.951e+03 316.409  < 2e-16 ***
DisturbL        3.095e-02  1.912e-02  2.451e+01   1.619    0.118
DisturbLH       1.447e-02  2.000e-02  2.767e+01   0.724    0.475
---
```

For 5% percent increase,:

```
> (1.05^1.96-1)*100
[1]  10.03505
```

AGB increases by about 10%

Both DV & IV log-transformed: Interpret the coefficient as the percent increase in the dependent variable for every 1% increase in the independent variable.

For every 1% increase in DBH, AGB increases by about 1.96%.

Only independent/predictor variable(s) is log-transformed. Divide the coefficient by 100. This tells us that a 1% increase in the independent variable increases (or decreases) the dependent variable by (coefficient/100) units.

example   tree mass

```
> summary(m6)

Fixed effects:
               Estimate Std. Error        df t value Pr(>|t|)
(Intercept)  -1.295e+00  5.750e-02  8.243e+00 -22.518 1.08e-08 ***
H13           5.121e-02  2.598e-04  1.955e+03 197.081  < 2e-16 ***
log(DBH0_13)  1.961e+00  6.199e-03  1.951e+03 316.409  < 2e-16 ***
DisturbL      3.095e-02  1.912e-02  2.451e+01   1.619    0.118
DisturbLH     1.447e-02  2.000e-02  2.767e+01   0.724    0.475
---
```

Only independent/predictor variable(s) is log-transformed. Divide the coefficient by 100. This tells us that a 1% increase in the independent variable increases (or decreases) the dependent variable by (coefficient/100) units. Example: the coefficient is 0.198, then 0.198/100 = 0.00198. For every 1% increase in the independent variable, our dependent variable increases by about 0.002. For x percent increase, multiply the coefficient by log(1.x). Example: For every 10% increase in the independent variable, our dependent variable increases by about 0.198 * log(1.10) = 0.02.
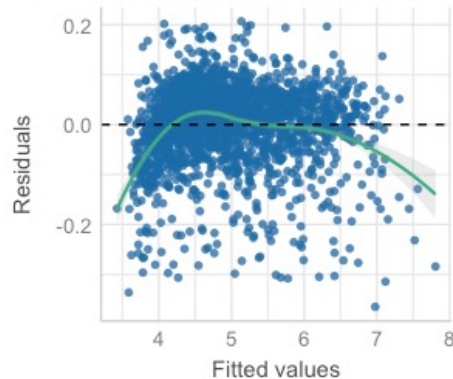
this situation not shown in this model!

```
performance::check_model(m6,
check = c("reqq", "qq", "linearity"))
```
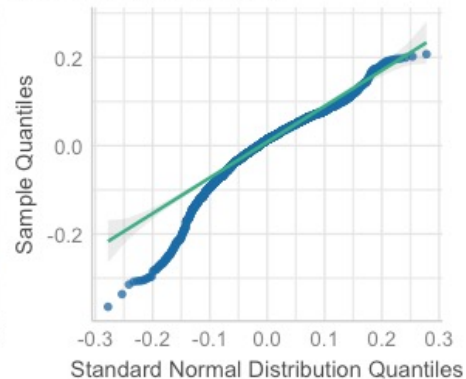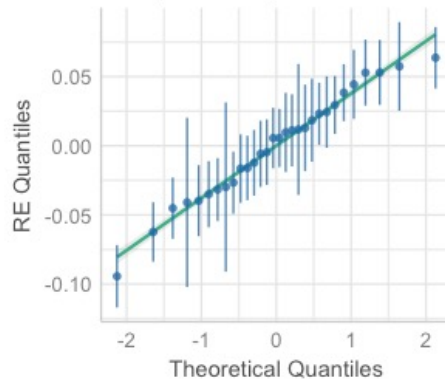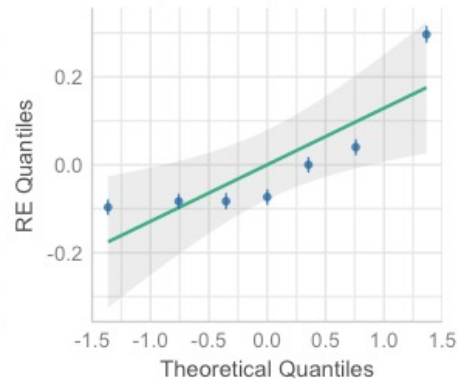
```
> summary(m6)

Random effects:
 Groups     Name          Variance Std.Dev.
 Plot       (Intercept) 0.001731 0.04160
 spp        (Intercept) 0.019695 0.14034
 Residual               0.006501 0.08063

Fixed effects:
                Estimate Std. Error         df t value Pr(>|t|)
(Intercept)  -1.295e+00   5.750e-02  8.243e+00 -22.518 1.08e-08 ***
H13           5.121e-02   2.598e-04  1.955e+03 197.081  < 2e-16 ***
log(DBH0_13)  1.961e+00   6.199e-03  1.951e+03 316.409  < 2e-16 ***
DisturbL      3.095e-02   1.912e-02  2.451e+01   1.619    0.118
DisturbLH     1.447e-02   2.000e-02  2.767e+01   0.724    0.475
---
```

Our model accounted for about 99% of the variance in AGB with main effects accounting for 95.8% of variance. Height (effect = 0.051, t = 197.1, $p < 0.001$) and tree diameter (log-effect = 1.96, t = 316.4, $p < 0.001$) strongly affect aboveground biomass. Tree plots (SD = 0.04) and species (SD = 0.14) accounted for a relatively small proportion of the variance in AGB. For every increase of 1-m in height, AGB increases by about 5.3%; whereas for every 1% increase in tree diameter, AGB increases by about 1.96%.

```
ss2$Hz <- scale(ss2$H13, scale = T)
ss2$Dz <- scale(ss2$DBH0_13, scale = T)
 m7 <- lmer(log(AGB_kg) ~ Hz + log(Dz) +
              Disturb + (1|spp) + (1|Plot), data = ss2)


> summary(m7)
Linear mixed model fit by REML. t-tests use Satterthwaite's method
['lmerModLmerTest']
Formula: log(AGB_kg) ~ Hz + log(Dz) + Disturb + (1 | spp) + (1 | Plot)
   Data: ss2

Random effects:
 Groups    Name        Variance Std.Dev.
 Plot      (Intercept) 0.001103 0.0332
 spp       (Intercept) 0.020161 0.1420
 Residual              0.065633 0.2562
Number of obs: 696, groups:  Plot, 28; spp, 7

Fixed effects:
             Estimate Std. Error        df t value Pr(>|t|)
(Intercept) 5.857e+00  5.808e-02 6.963e+00 100.830  2.8e-12 ***
Hz          3.625e-01  9.801e-03 6.775e+02  36.988  < 2e-16 ***
log(Dz)     2.925e-01  8.149e-03 6.854e+02  35.892  < 2e-16 ***
DisturbL    6.205e-02  2.728e-02 2.146e+01   2.275   0.0333 *
DisturbLH   2.668e-02  3.309e-02 3.425e+01   0.806   0.4256
---
```
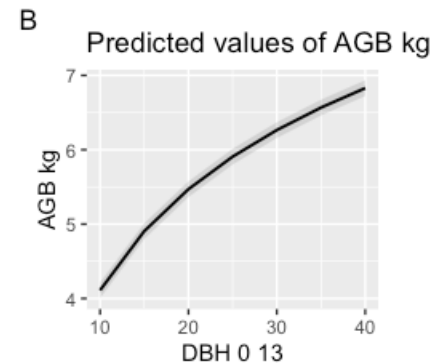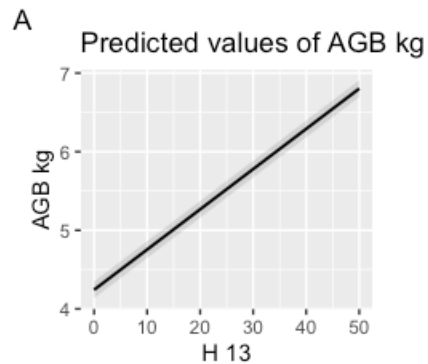
```
require(sjPlot)
re_gg <- plot_model(m6, type = "re", sort.est = TRUE)
 re_gg[1]
 re_gg[2]
```
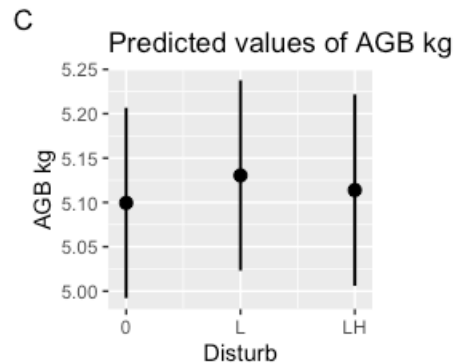

Random effects

Random effects
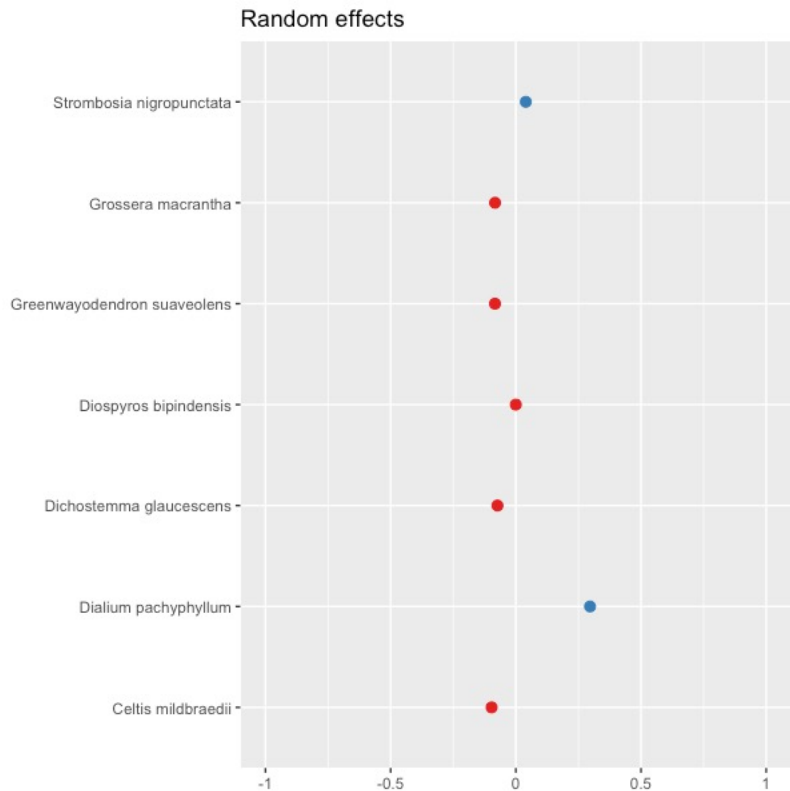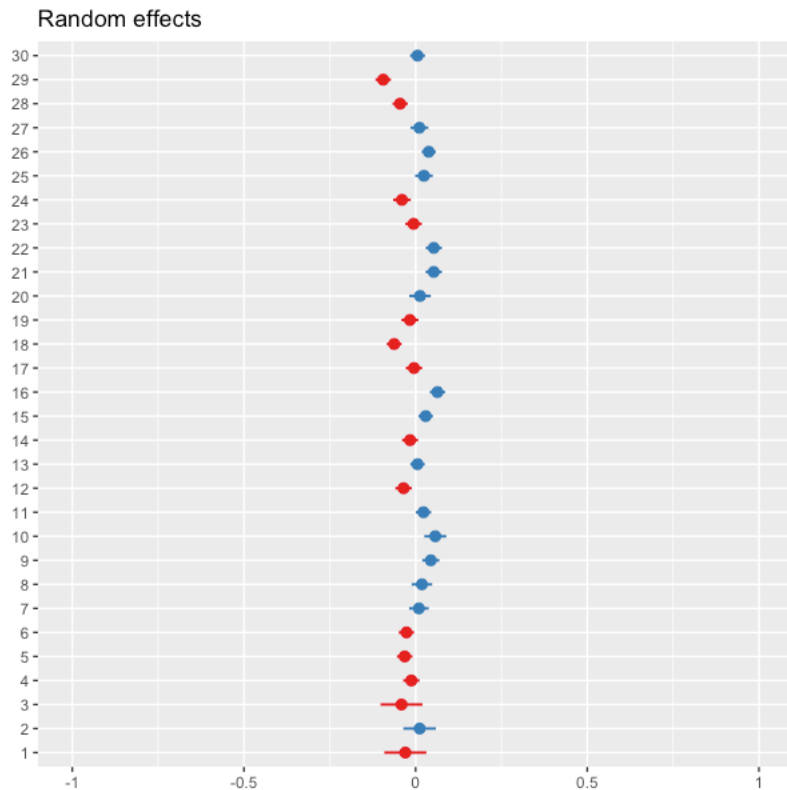
example   tree mass


```
eff_gg <- plot_model(m6, type = "eff", sort.est = TRUE)
plot_grid(eff_gg)
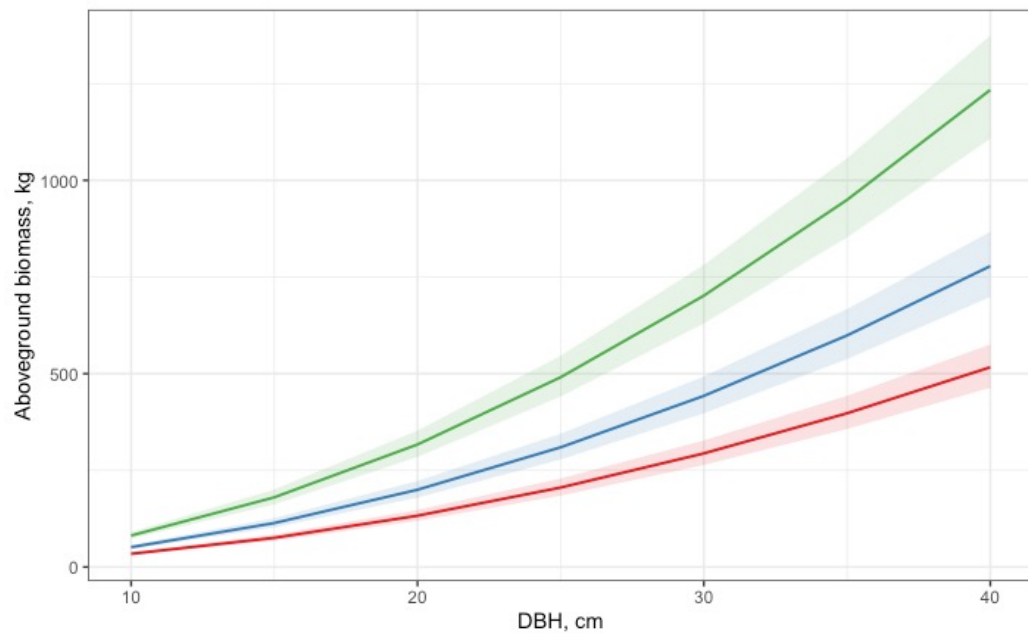```

Predicted values of AGB kg
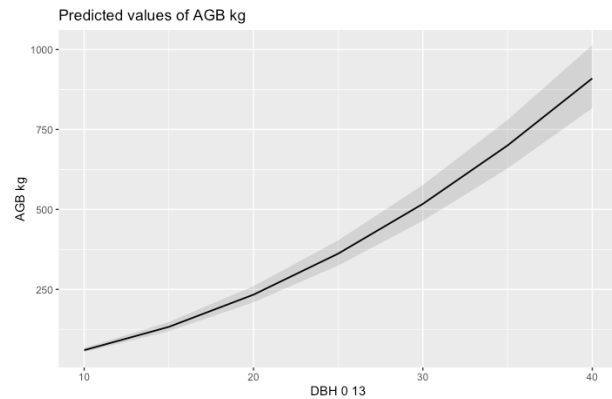
```
plot_model(m6, type = "pred", terms = "DBH0_13")
```
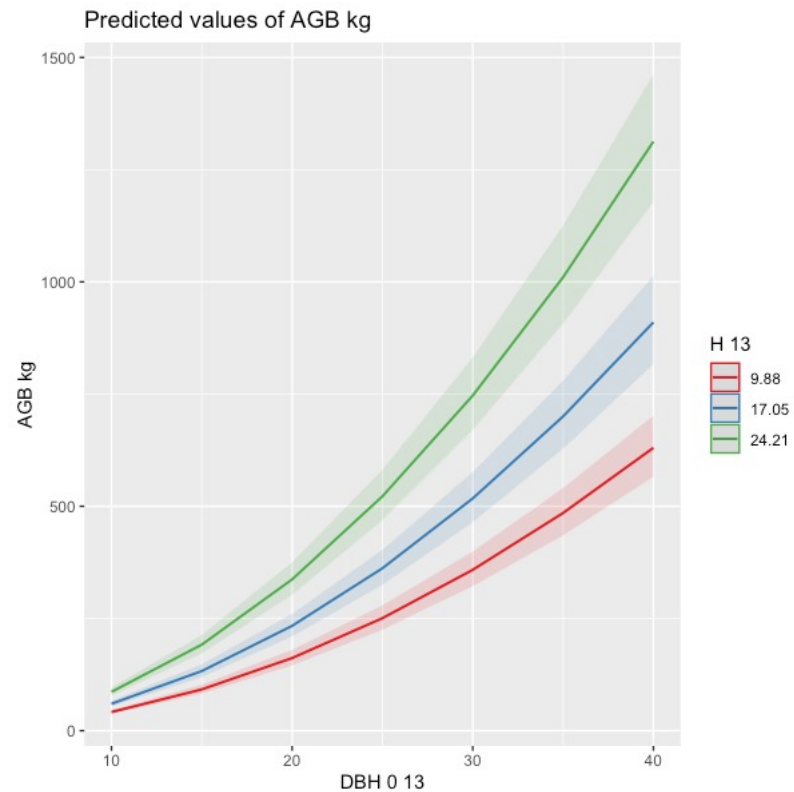
```
plot_model(m6, type = "pred", terms = c("DBH0_13", "H13"))
```

```
plot_model(m6, type = "pred", terms = c("DBH0_13", "H13 [6, 14, 23]"),
           title = "") +
  labs(y = "Aboveground biomass, kg", x = "DBH, cm") +
  theme_bw() + theme(legend.position = "none")
```

Questions?