# Multilevel modelling

PSY9003

2020/03/24

Multilevel data

There are *many* situations in psychology where we have *nested* data.

Intervention studies are typically longitudinal - the same participants are tested multiple times on the same outcome measure.

Typical cognitive experiments show participants many repeats of similar trials.

| Fixation (500ms)	Visual cue (250ms)	Cue-target (1-1.25s)	Target (66ms)	Response
+	<-	+		???
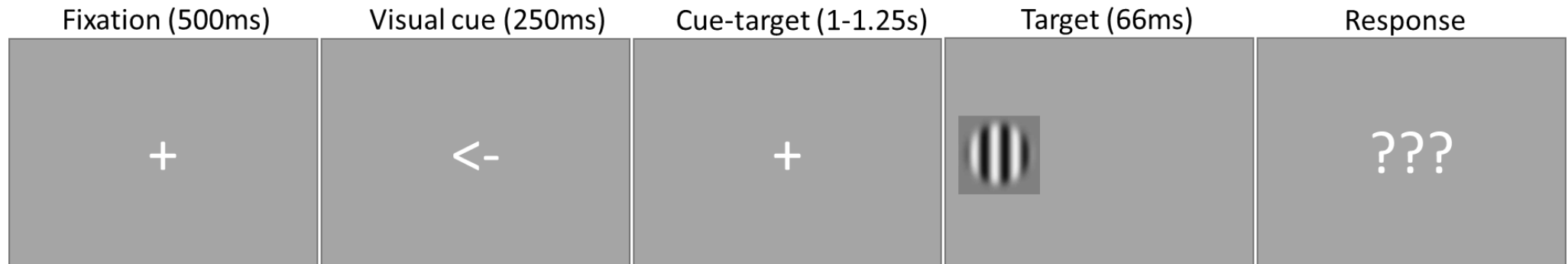
Multilevel data
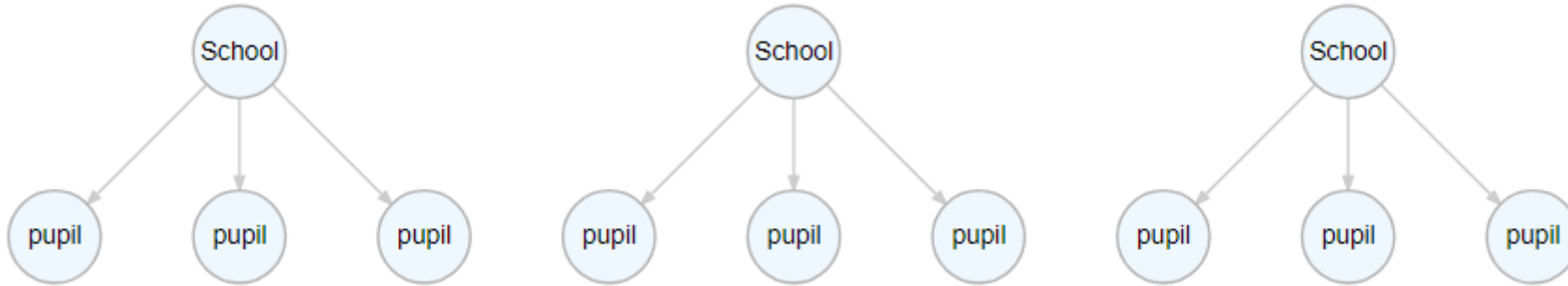
In this example, each pupil is a unit of observation.

But these pupils are not fully independent from each other - pupils who attend one school tend to be more similar to each other than they are to pupils who attend other schools.

Thus, *pupils* (Level 1) are nested in *schools* (Level 2).

Multilevel data
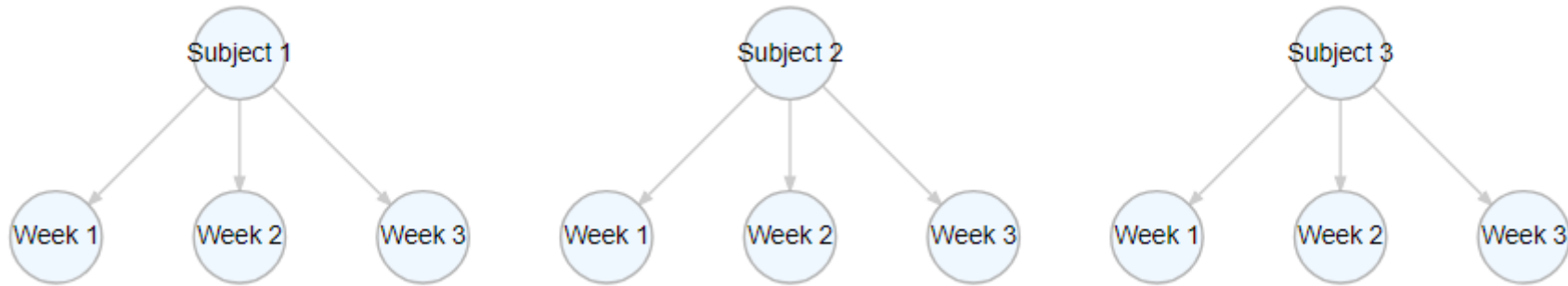
Other data may be *longitudinal*. For example, you may measure outcomes such as, for example, performance or attitudes on repeated occassions to see how they vary over time.

The measurements each week are the main unit of observation, but they are nested within subjects.
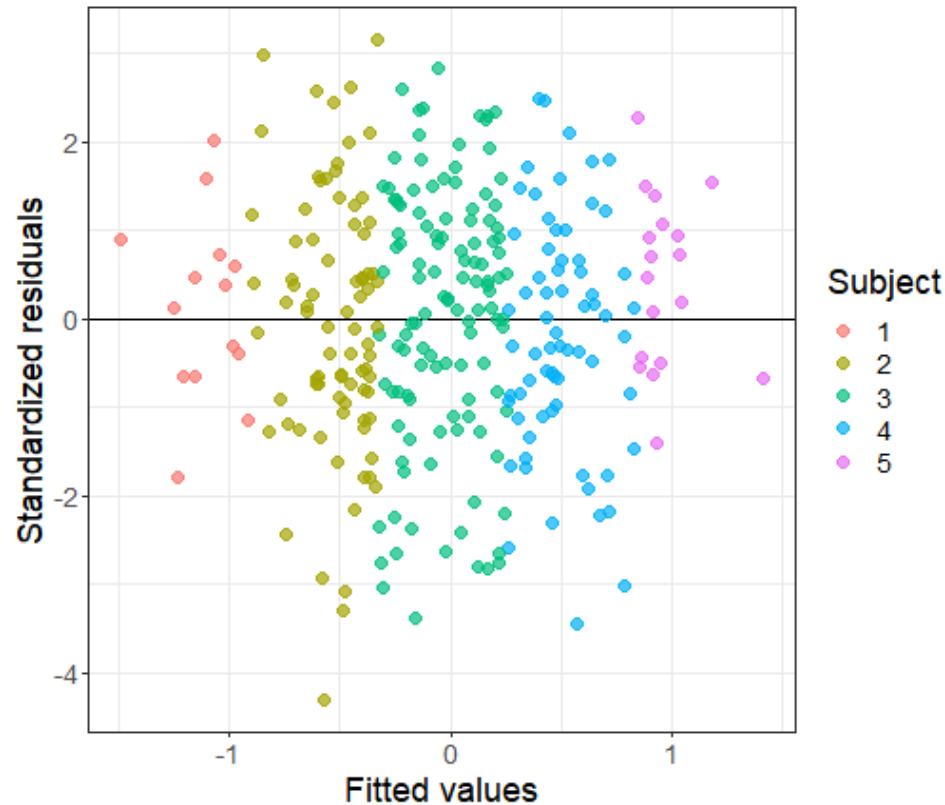
Independent identically distributed errors
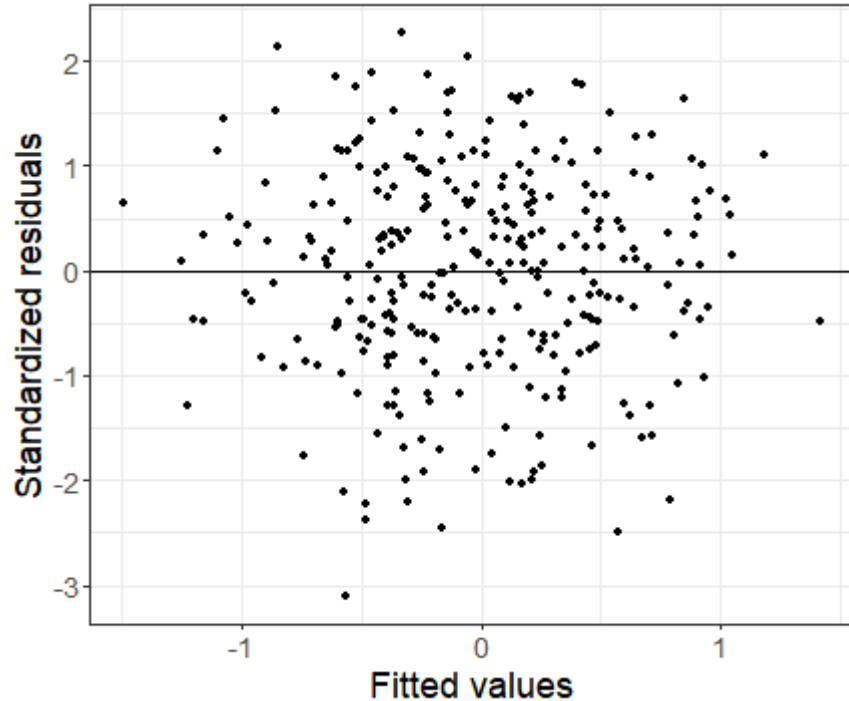
One of the key assumptions of many of the tests we've covered is that of **independent identically distributed** (*iid*) errors.

All of the points on this graph are from *independent sources*: each point is a score from one person, on one measure, at one point in time.

Clustered data

Data from nested designs like those we have just seen often have *clusters* of correlated observations.

Different people have different reaction speeds, or baseline attitudes; different schools have different teachers and different general environments.
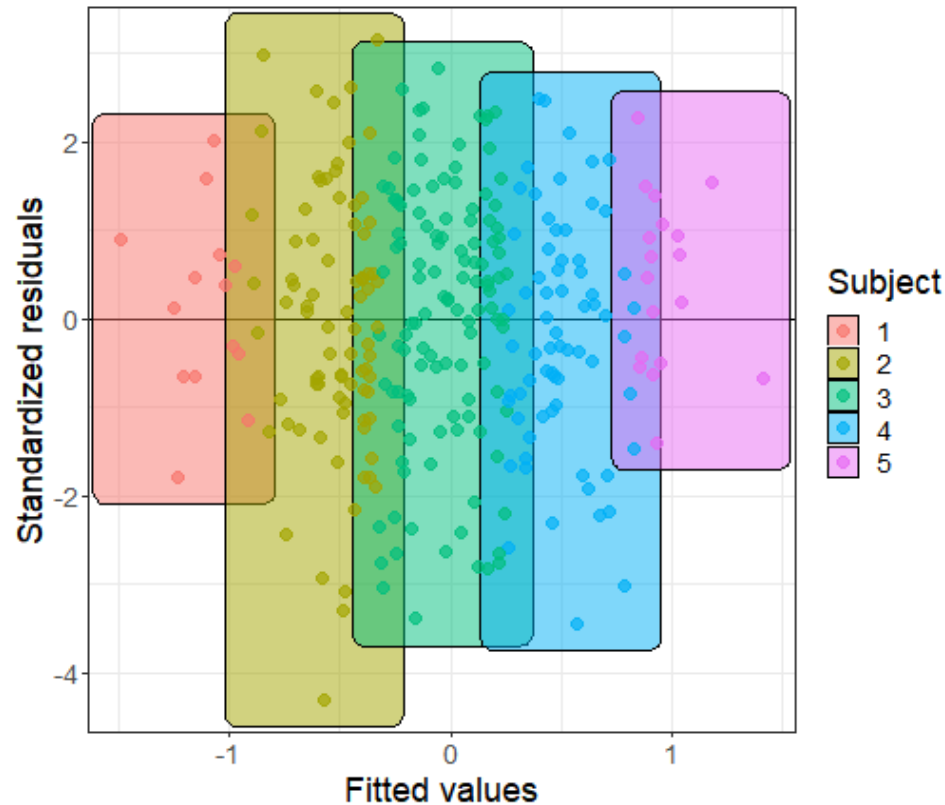
Clustered data

Data from nested designs like those we have just seen often have *clusters* of correlated observations.

Different people have different reaction speeds, or baseline attitudes; different schools have different teachers and different general environments.

The problem with nesting

sleepstudy

The *sleepstudy* dataset contains data from a sleep deprivation experiment.

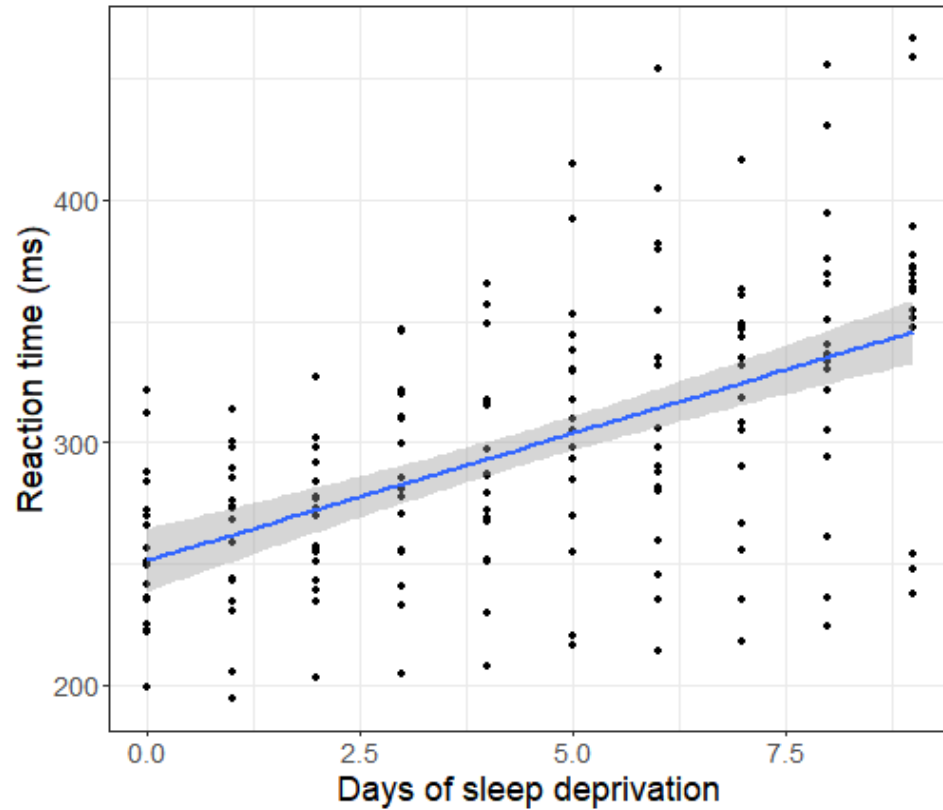Over the course of ten days, subjects were only allowed to sleep for 3 hours each night.

Each day their reaction times on a variety of cognitive tasks were recorded.

This is a *nested*, multilevel design.

Each observation - average RT on a given day - is nested within a *subject*.

sleepstudy

Suppose we were to ignore the clustering in the data.

We could simply fit a linear model to the whole dataset.

A basic linear model

```
basic_lm <- lm(Reaction ~ Days, data = sleepstudy)
summary(basic_lm)
```

```
##
## Call:
## lm(formula = Reaction ~ Days, data = sleepstudy)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -110.848  -27.483    1.546   26.142  139.953
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)   251.405      6.610   38.033  < 2e-16 ***
## Days           10.467      1.238    8.454 9.89e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 47.71 on 178 degrees of freedom
## Multiple R-squared:  0.2865,    Adjusted R-squared:  0.2825
## F-statistic: 71.46 on 1 and 178 DF,  p-value: 9.894e-15
```

A basic linear model

```
coef(basic_lm)
```

```
## (Intercept)        Days
##   251.40510    10.46729
```

The mean reaction time is 251 ms.

For each one unit increase in Days (i.e. for each extra day with disrupted sleep), predicted reaction times go up by around 10 ms.

A quick look at the residuals

```
plot_model(basic_lm, type = "diag")[[1]]
```

```
## `geom_smooth()` using formula 'y ~ x'
```
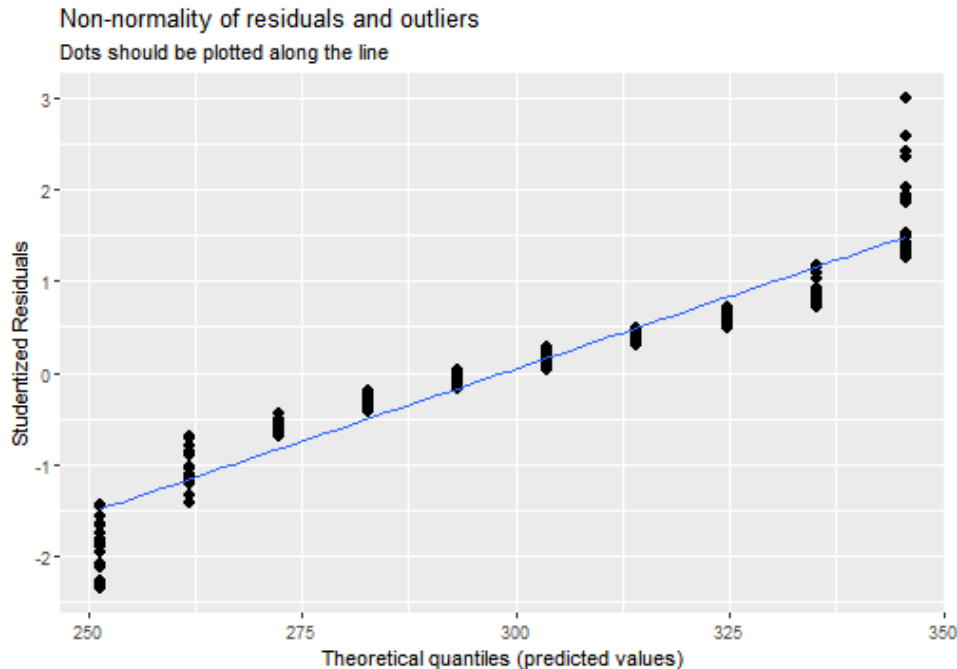
These residuals show a degree of patterning that suggests something is not being accounted for in our model.

The points clearly do not lie along the line, and seem to be more variable at the ends than over the full range of the x-axis.

sleepstudy
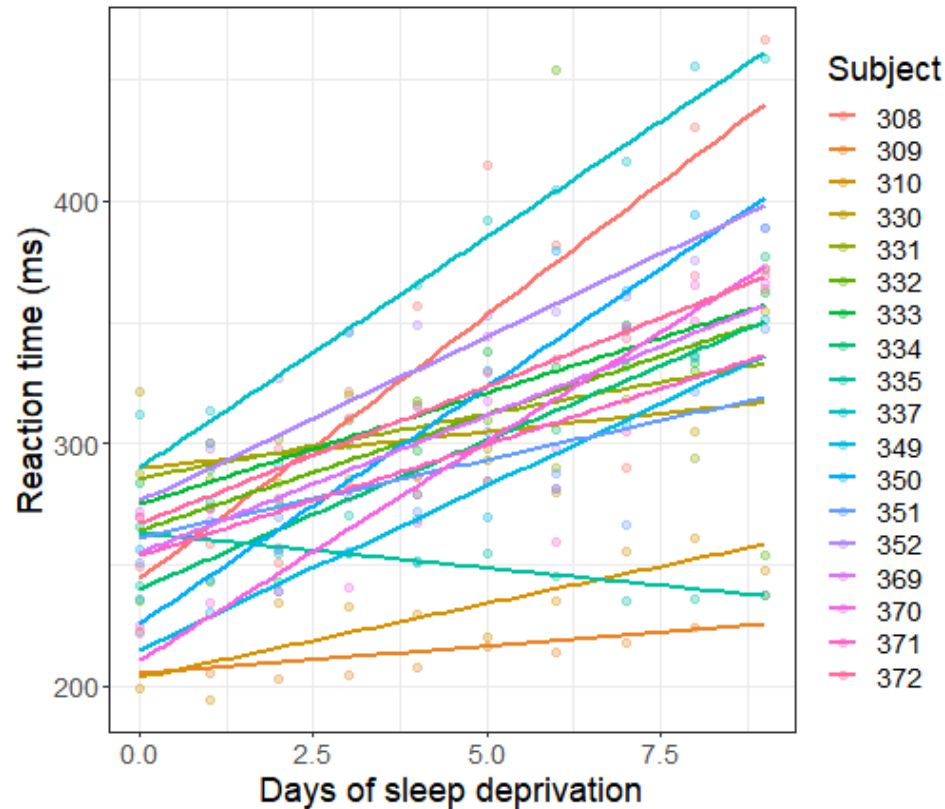
Our simple linear model ignores the fact that many of our observations are repeated measurements from each participant.

It assumes the effect is the same for everyone.

There are 18 participants in this study. Some of them are generally faster or slower than others; some of them show more effect of sleep deprivation than others.

Estimating multilevel models

Nesting in `sleepstudy`

In the `sleepstudy` data, each subject is tested on 10 consecutive days.

Thus, each subject contributes 10 reaction times.

Reaction times (level 1) are nested within *subjects* (level 2).

Our basic linear model ignores the nesting and fits a single model to all the data.

This is fine for telling us the *group-average* but doesn't tell us about the variability between subjects.

Multilevel models

Multilevel models allow us to account for the nested, correlated nature of the data, and explicitly model the variability between people.

You may also see them called:

- Hierarchical models
- Mixed-effects models
- Random-effects models
- Mixed models

Fixed and random effects

Fixed effects are the *population-average* effect: e.g. the *average* effect of days of sleep deprivation.

Random effects are those that vary across the *sampling units*.

They are *random* because the *sampling units* are randomly drawn from a wider *population*. e.g. the specific participants in an experiment are usually a random subset of all possible participants

Random intercept and random slopes

Random effects can be broken down into *random intercepts* and *random slopes*.

A model that models *random intercepts* models how *average* levels of performance for each *sampling unit* vary: e.g. the mean reaction time for each individual is modelled.

A model that models *random slopes* models how the effects of *predictors* varies across *sampling units*: e.g. how days of sleep deprivation affects performance is modelled across subjects.

Multilevel models using lme4

The most important library for fitting this type of model is `lme4`.

```
library(lme4)
```

A multilevel model can be fitted with the `lmer()` function.

lmer(DV ~ IV1 + IV2 + (IV1| random_factor), data = your_data)

Fixed *effects* are highlighted in blue.

Random *effects* are highlighted in red. Random effects are always *categorical*.

Random *slopes* are highlighted in green.

Random intercepts

Modelling random intercepts

Remember that in our basic model, the *intercept* represents the mean reaction time.

We can model the variability of the intercept better by including a *random effect* term - *(1 | Subject)*.

```r
int_only <-
  lmer(Reaction ~ Days + (1 | Subject),
       data = sleepstudy) # Random intercept
```

This model is a *random-intercept* model - it captures how mean reaction times vary across subjects.

Individual intercepts
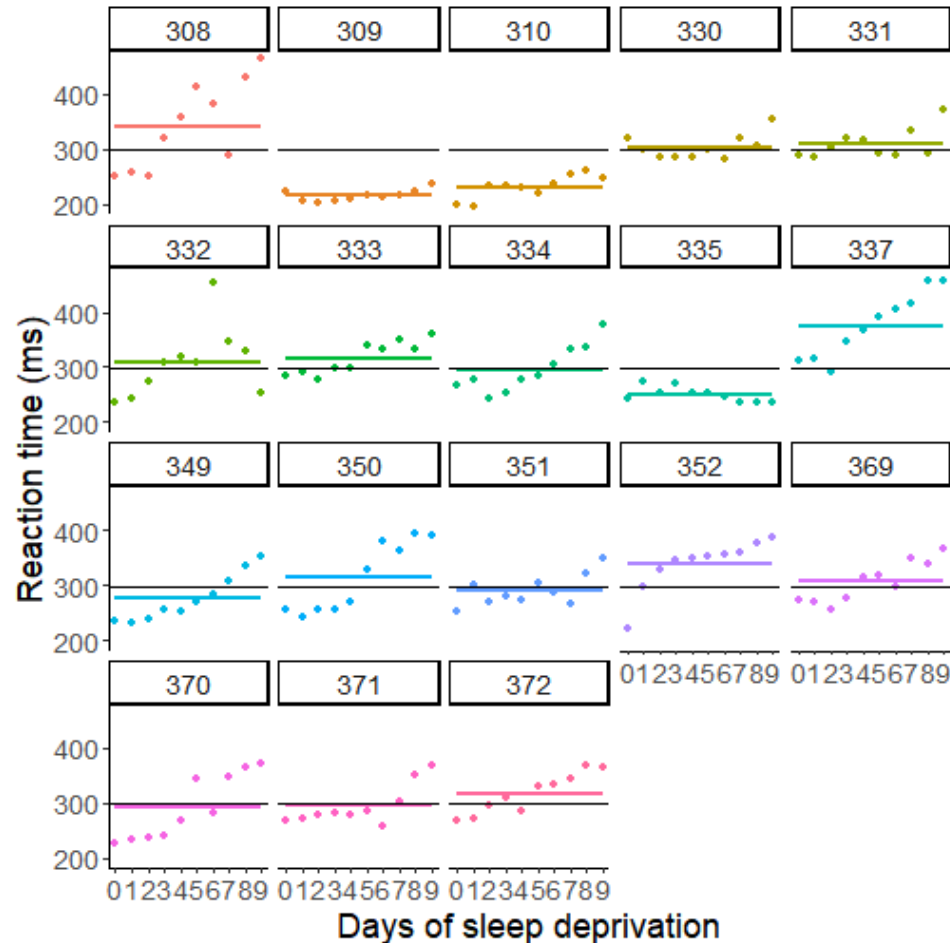
Here, the coloured lines represent the participants' *mean* reaction time, while the black lines represent the predicted mean reaction time of the linear model we fit to *all* the data.

A *random-intercept* model models that variability!

```
summary(int_only)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: Reaction ~ Days + (1 | Subject)
##    Data: sleepstudy
##
## REML criterion at convergence: 1786.5
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.2257 -0.5529  0.0109  0.5188  4.2506
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  Subject  (Intercept) 1378.2   37.12
##  Residual              960.5   30.99
## Number of obs: 180, groups:  Subject, 18
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept) 251.4051     9.7467   25.79
## Days         10.4673     0.8042   13.02
##
## Correlation of Fixed Effects:
##      (Intr)
## Days -0.371
```

Standard linear model

```
##               Estimate Std. Error   t value     Pr(>|t|)
## (Intercept) 251.40510   6.610154 38.033169 2.156888e-87
## Days         10.46729   1.238195  8.453663 9.894096e-15
```

Intercept only mixed-model

```
##               Estimate Std. Error  t value
## (Intercept) 251.40510  9.7467163 25.79383
## Days         10.46729  0.8042214 13.01543
```

The *standard errors* differ, which means the *t-values* differ.

The *intercept* variability increased, while the Days variability decreased!

Random effects

The *fixed* effects give us a measure of average performance and the overall effect of Days of sleep deprivation on RT.

```
fixef(int_only)
```

```
## (Intercept)        Days
##   251.40510    10.46729
```
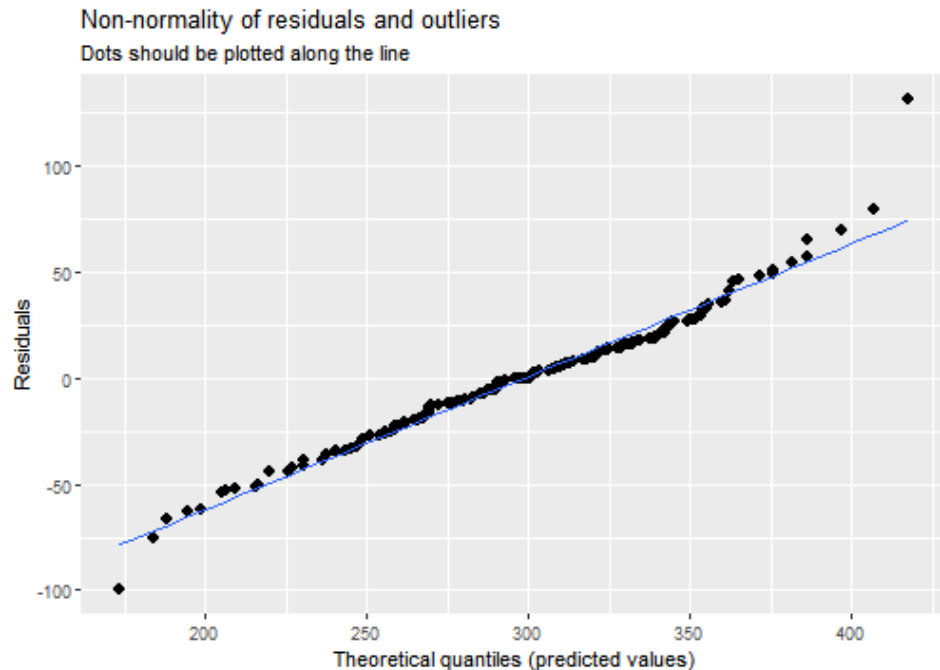
The *random* effects tell us how much variability there is *between-participants*. In this case, we only estimated participant-specific intercepts.

```
summary(int_only)$varcor
```

```
##  Groups   Name         Std.Dev.
##  Subject  (Intercept)  37.124
##  Residual              30.991
```

A quick look at the residuals

```
library(sjPlot)
plot_model(int_only, type = "diag")[[1]]
```

Non-normality of residuals and outliers

Dots should be plotted along the line

This model - the *random intercept* model - is *much much* better than our basic linear model.

The points now lie almost entirely along the line.

This indicates a good correspondence between the model predictions and the actual data!

A quick look at the residuals

```
plot_model(int_only, type = "diag")[[4]]
```


Homoscedasticity (constant variance of residuals)
Amount and distance of points scattered above/below line is equal or randomly spread
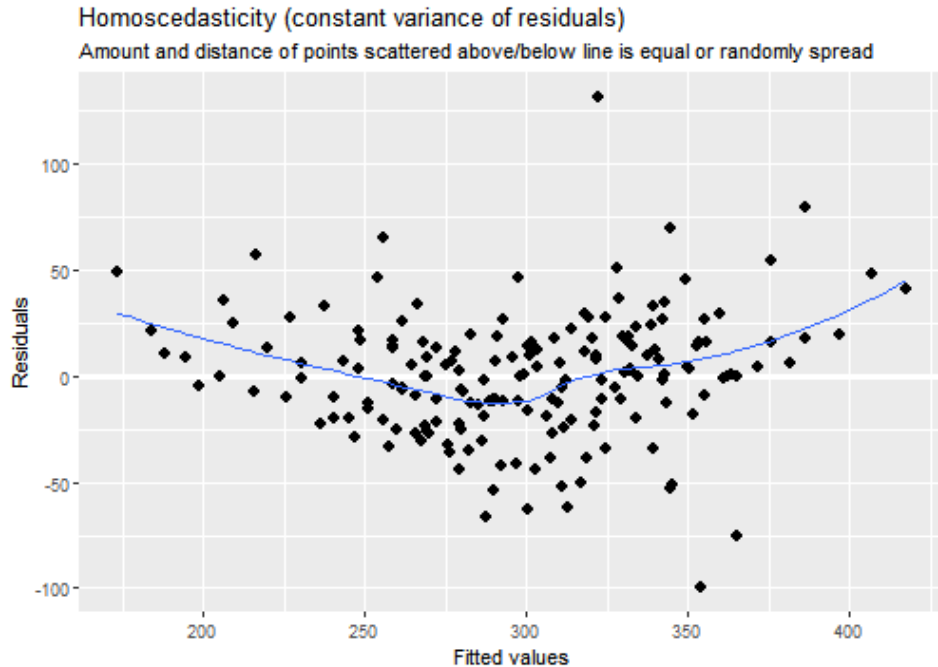
But this plot shows evidence of heteroskedasticity - non-constant variance.

The dots seem to *curve* somewhat.

This suggests there is still something not quite right in our model.

First Exercise

`10:00`

Random slopes

Individual slopes
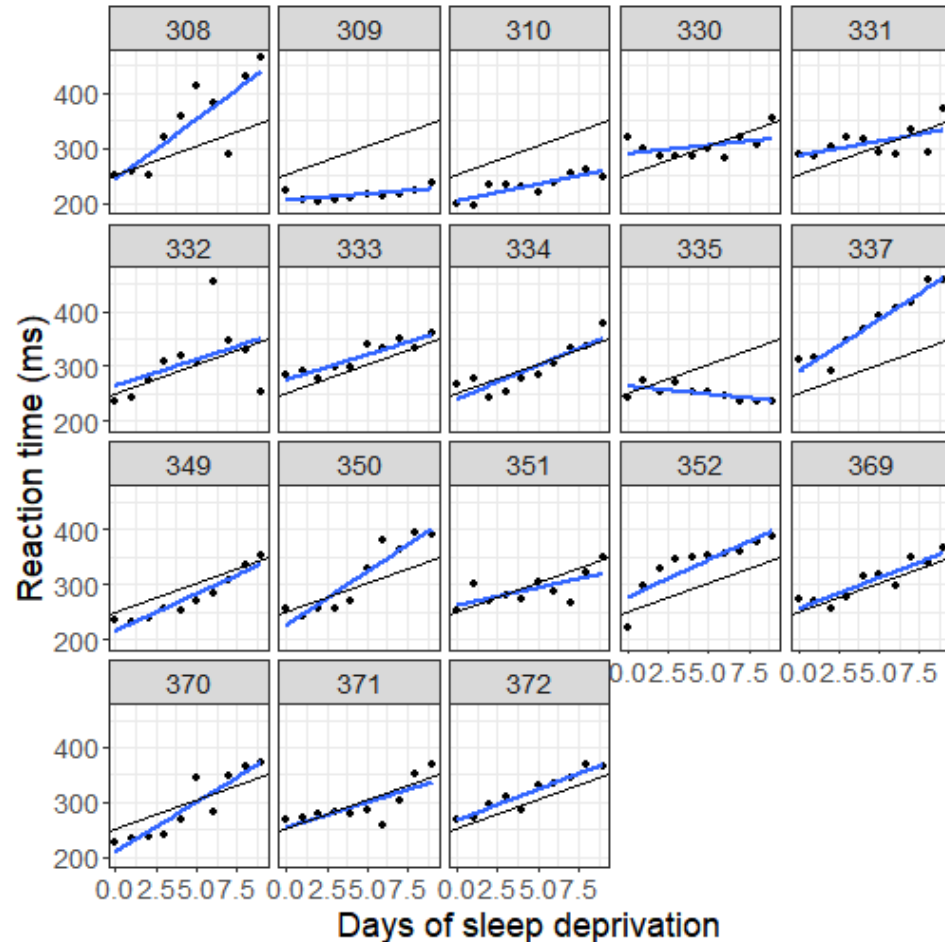
This plot now show individual plots for each participant with the individual effect of Days added.

The general trend is consistent, but it's clear that some participants have stronger effects than others.

And it looks a little like people who are generally fast responders show *less* effect of Days of sleep deprivation.

Modelling random slopes

We can model how much the effect of `Days` varies between participants by adding *random slopes* to our model - `(Days | Subject)`.

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

Note that `Days` now appears twice.

The first time models the *population-average* effect of `Days`.

The second time models the *individual* effect of `Days`.

```
summary(random_slope)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: Reaction ~ Days + (Days | Subject)
##    Data: sleepstudy
##
## REML criterion at convergence: 1743.6
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.9536 -0.4634  0.0231  0.4633  5.1793
##
## Random effects:
##  Groups   Name        Variance Std.Dev. Corr
##  Subject  (Intercept) 611.90   24.737
##           Days         35.08    5.923   0.07
##  Residual             654.94   25.592
## Number of obs: 180, groups:  Subject, 18
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)  251.405      6.824  36.843
## Days          10.467      1.546   6.771
##
## Correlation of Fixed Effects:
##      (Intr)
```

Basic linear model

```
##              Estimate Std. Error   t value      Pr(>|t|)
## (Intercept) 251.40510   6.610154 38.033169 2.156888e-87
## Days         10.46729   1.238195  8.453663 9.894096e-15
```

Intercept only mixed-model

```
##              Estimate Std. Error  t value
## (Intercept) 251.40510  9.7467163 25.79383
## Days         10.46729  0.8042214 13.01543
```

Random slopes and intercepts mixed-model

```
##              Estimate Std. Error   t value
## (Intercept) 251.40510   6.823773 36.842535
## Days         10.46729   1.545958  6.770744
```

Intercept only mixed-model

```
##  Groups    Name        Std.Dev.
##  Subject   (Intercept) 37.124
##  Residual              30.991
```

Random slopes and intercepts mixed-model

```
##  Groups    Name        Std.Dev. Corr
##  Subject   (Intercept) 24.7366
##            Days          5.9229  0.066
##  Residual              25.5918
```

Model comparisons

Is this model an improvement? Use anova() to check!

```
anova(int_only, random_slope)
```

```
## refitting model(s) with ML (instead of REML)

## Data: sleepstudy
## Models:
## int_only: Reaction ~ Days + (1 | Subject)
## random_slope: Reaction ~ Days + (Days | Subject)
##              Df    AIC    BIC  logLik deviance  Chisq Chi Df Pr(>Chisq)
## int_only      4 1802.1 1814.8 -897.04   1794.1
## random_slope  6 1763.9 1783.1 -875.97   1751.9 42.139      2  7.072e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

(it's significant, so yes!)

A quick look at the residuals

```
plot_model(random_slope, type = "diag")[[4]]
```

Homoscedasticity (constant variance of residuals)
Amount and distance of points scattered above/below line is equal or randomly spread
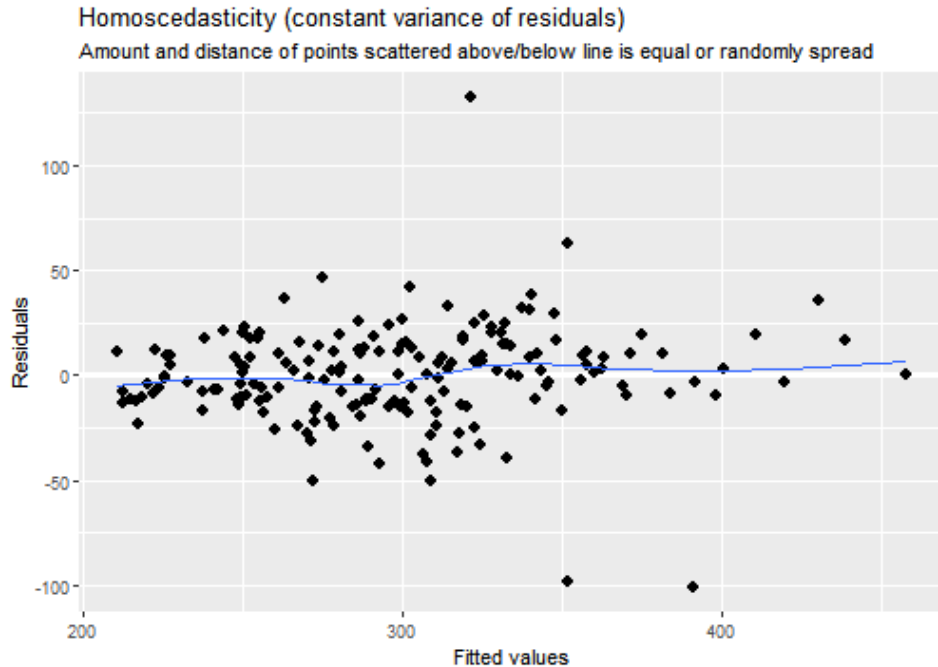
These residuals are the best of all so far.

A few points look suspiciously like outliers, but overall, there's little to suggest any particular problems with this model!

```
tab_model(random_slope)
```

Predictors	Reaction Estimates	Reaction CI	Reaction p
(Intercept)	251.41	238.03 – 264.78	**<0.001**
Days	10.47	7.44 – 13.50	**<0.001**

Random Effects

σ^2	654.94
τ_{00} Subject	611.90
τ_{11} Subject.Days	35.08
ρ_{01} Subject	0.07
ICC	0.72
N Subject	18
Observations	180
Marginal R^2 / Conditional R^2	0.279 / 0.799

Second Exercise

10:00

Multiple random effects

The "language as fixed-effect" fallacy

A common circumstance in psychological research is that we have more than one random effect.

For example, in language experiments, subject often need to read a many different words; these may be words from different categories, or vary in other ways.

These words themselves are random samples, but many researchers treat them as being *fixed*.

Clark, 1973

The `politeness` study

Winter and Grawunder (2012) looked at the relationship between vocal pitch and the level of politeness of a sentence.

Participants were asked to imagine how they would respond to a variety of scenarios when talking politely or informally.

```
politeness <- read_csv("data/politeness_data.csv")
head(politeness)
```

```
## # A tibble: 6 x 5
##   subject gender scenario attitude frequency
##   <chr>   <chr>     <dbl> <chr>        <dbl>
## 1 F1      F             1 pol           213.
## 2 F1      F             1 inf           204.
## 3 F1      F             2 pol           285.
## 4 F1      F             2 inf           260.
## 5 F1      F             3 pol           204.
## 6 F1      F             3 inf           287.
```

The `politeness` study

In the `politeness` study, there are *two* distinct groupings:

1) Subjects repeat the same task (imaging a scenario) over and over again

2) Individual scenarios are repeated by different subjects

Thus there are *two* possible sources of correlated data - we'd expect responses to particular scenarios to be fairly consistent across subjects, and responses by individual subjects to be fairly consistent across items,

Variability between subjects

```
boxplot(frequency ~ subject, data = politene
```
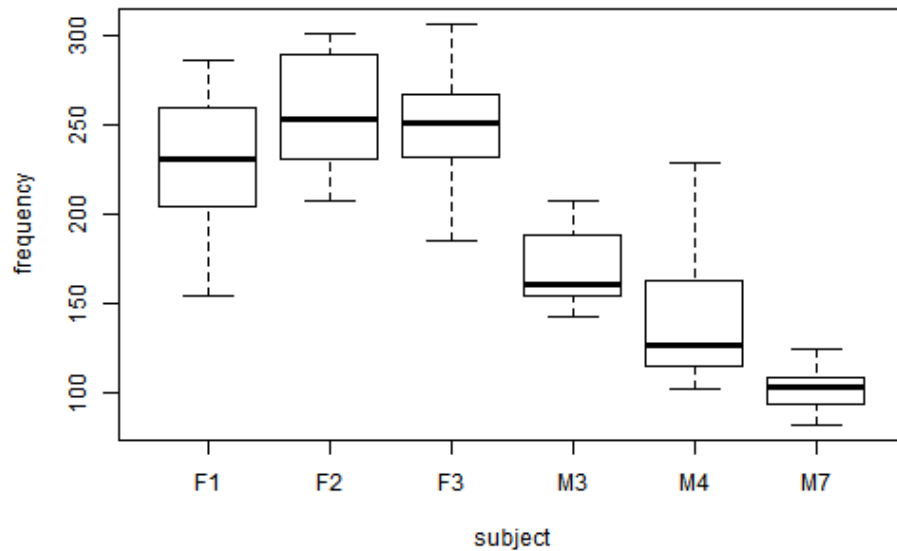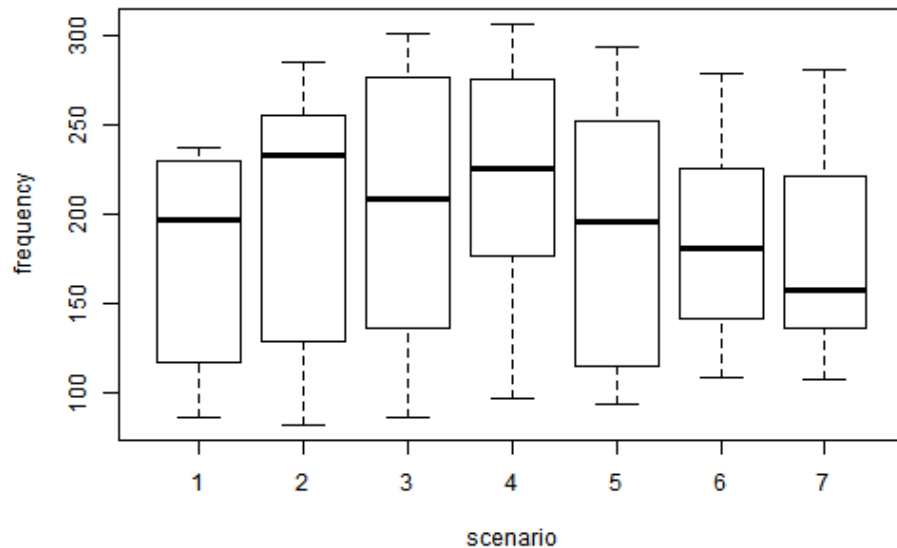

Individual participants vary in their baseline vocal frequency.

Male participants typically have lower frequency voices than female participants.

Variability between scenarios

```
boxplot(frequency ~ scenario, data = politen
```


There seems to be some variability across scenarios, but less consistent than across participants.

But scenario 7 seems consistently lower than scenario 4, for example.

Multiple random effects

We can model *both* of these sources of variability simultaneously by adding *multiple random effects*.

```
full_mod <- lmer(frequency ~ attitude + (1|subject) + (1|scenario), data = politeness)
```

Whereas before we only added `(1|subject)`, here we also add `(1|scenario)`.

This models separate intercepts for each subject and each scenario, allowing for, for example, high-pitched individuals or scenarios that typically elicit low-pitched responses.

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: frequency ~ attitude + (1 | subject) + (1 | scenario)
##    Data: politeness
##
## REML criterion at convergence: 793.5
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.2006 -0.5817 -0.0639  0.5625  3.4385
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  scenario (Intercept)  219     14.80
##  subject  (Intercept) 4015     63.36
##  Residual              646     25.42
## Number of obs: 83, groups:  scenario, 7; subject, 6
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept)  202.588     26.754   7.572
## attitudepol  -19.695      5.585  -3.527
##
## Correlation of Fixed Effects:
##             (Intr)
## attitudepol -0.103
```

```
tab_model(full_mod)
```

	frequency		
Predictors	*Estimates*	*CI*	*p*
(Intercept)	202.59	150.15 – 255.02	**<0.001**
attitude [pol]	-19.69	-30.64 – -8.75	**<0.001**
Random Effects			
σ^2	646.02		
$\tau_{00 \text{ scenario}}$	218.98		
$\tau_{00 \text{ subject}}$	4014.57		
ICC	0.87		
N $_{\text{subject}}$	6		
N $_{\text{scenario}}$	7		
Observations	83		
Marginal R^2 / Conditional R^2	0.020 / 0.870		

Final Exercise

10:00

Generalized linear mixed effects models

As discussed last week, there are many types of data for which a linear model is *inappropriate.*

Fortunately, we can fit **generalized linear mixed effects models** too!

glmer(DV ~ IV1 + IV2 + (IV1 | random_factor), family = binomial(), data = your_data)

Additional reading

Complete vs Partial vs no pooling

An introduction to mixed models

Keep it Maximal

Generalizing over encounters: statistical and theoretical considerations