# Nested vs Crossed random effects

## Advanced statistical methods and models in experimental design

Bartosz Maćkiewicz

# schools dataset

The example used for this class is fictional data where the interval scaled outcome variable "Extroversion" (`extro`) is predicted by fixed effects for the interval scaled predictor "Openness to new experiences" (`open`), the interval scaled predictor "Agreeableness (`agree`)", the interval scaled predictor "Social engagement" (`social`), and the nominal scaled predictor "Class" (`class`); as well as the random (nested) effect of Class within School (`school`).

The data contains 1200 cases evenly distributed among 24 nested groups (4 classes within 6 schools).

```
library(lme4)
library(lmerTest)
schools <- read.csv("schools.csv")
head(schools, 4)
##   X id    extro     open    agree    social class school
## 1 1  1 63.69356 43.43306 38.02668  75.05811     d     IV
## 2 2  2 69.48244 46.86979 31.48957  98.12560     a     VI
## 3 3  3 79.74006 32.27013 40.20866 116.33897     d     VI
## 4 4  4 62.96674 44.40790 30.50866  90.46888     c     IV
```

# Structure of effects

We know that “Openness”, “Agreeableness” and “Social engagement” will be modelled as fixed factors.

How to model the random effects od School and Class?

# Nested vs Crossed Random Effects (Harrison et al. 2018)

A common issue that causes confusion is this issue of specifying random effects as either **'crossed'** or **'nested'**.

In reality, the way you specify your random effects will be determined by your experimental or sampling design.

A simple example can illustrate the difference. Imagine a researcher was interested in understanding the factors affecting the clutch mass of a passerine bird. They have a study population spread across five separate woodlands, each containing 30 nest boxes. Every week during breeding they measure the foraging rate of females at feeders, and measure their subsequent clutch mass. Some females have multiple clutches in a season and contribute multiple data points. Here, female ID is said to be nested within woodland: each woodland contains multiple females unique to that woodland (that never move among woodlands).

# Nested vs Crossed Random Effects (Harrison et al. 2018)

The nested random effect controls for the fact that

(i) clutches from the same female are not independent, and

(ii) females from the same woodland may have clutch masses more similar to one another than to females from other woodlands

We can write the appropriate model model as:

```
Clutch Mass ~ Foraging Rate + (1|Woodland/Female ID)
```

# Nested vs Crossed Random Effects (Harrison et al. 2018)

Now imagine that this is a long-term study, and the researcher returns every year for five years to continue with measurements. Here it is appropriate fit year as a crossed random effect because every woodland appears multiple times in every year of the dataset, and females that survive from one year to the next will also appear in multiple years.

```
Clutch Mass ~ Foraging Rate + (1|Woodland/Female ID)+ (1|Year)
```

# Nested vs Crossed Random Effects (Harrison et al. 2018)

Understanding whether your experimental/sampling design calls for nested or crossed random effects is not always straightforward, but it can help to visualise experimental design by drawing it, or tabulating your observations by these grouping factors (e.g. with the `table` command in R) to identify how your data are distributed.

Always ensure that their levels of random effect grouping variables are uniquely labelled. For example, females are labelled $1 - n$ in each woodland, the model will try and pool variance for all females with the same code. Giving all females a unique code makes the nested structure of the data is implicit, and a model specified as

```
`Clutch Mass ~ Foraging Rate + (1| Woodland) + (1|FemaleID)
```

would be identical to the model above.

# Nested vs Crossed Random Effects (Schielzeth & Nakagawa 2012)

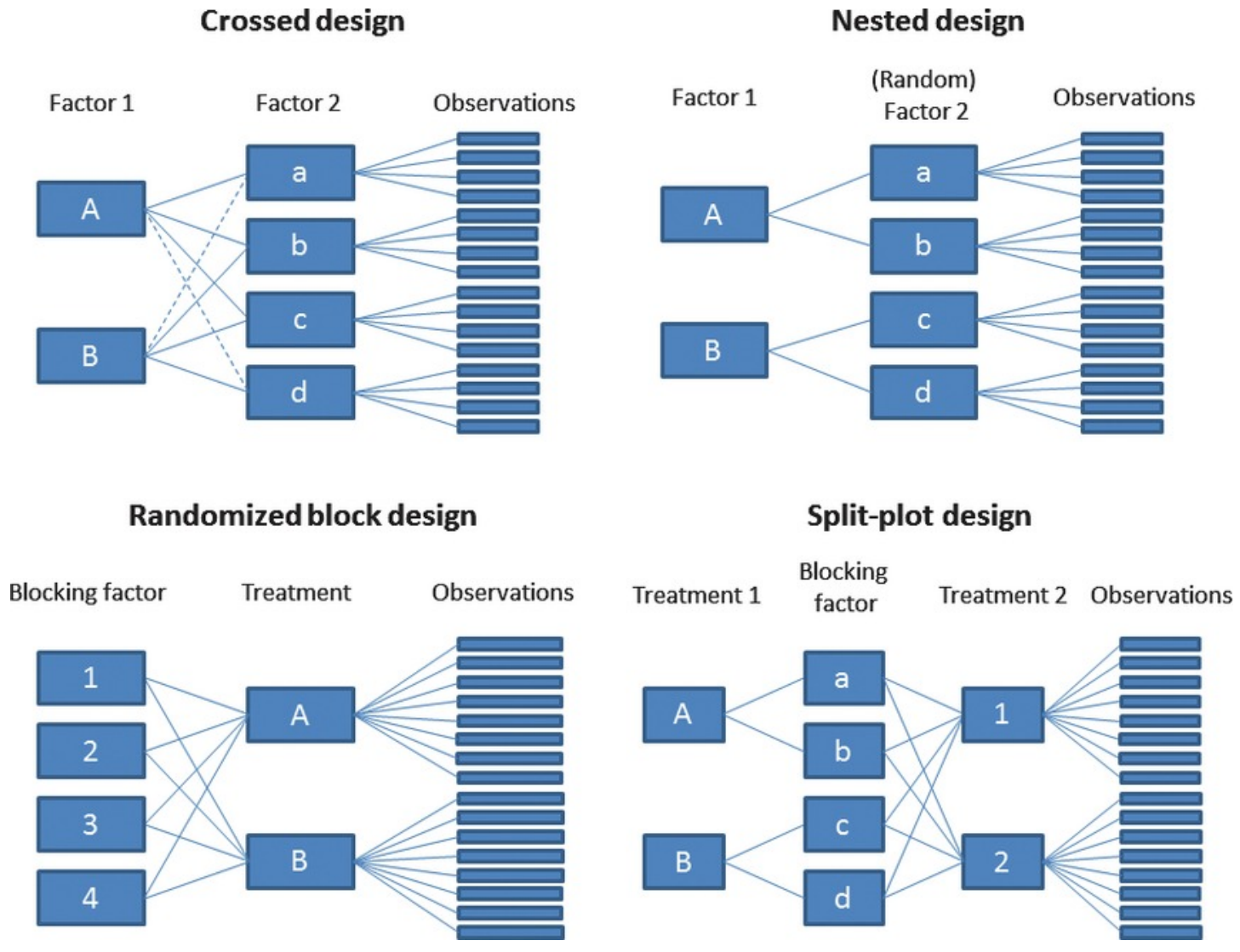


Figure 1: Common experimental designs

# Crossed Random Effects

```
fit_crossed <- lmer(formula = extro ~ # Extroversion
                        open + # fixed effect of Openness
                        agree + # fixed effect of Agreeableness
                        social + # fixed effect of Social engagement
                        (1|school) + # random intercept for each school
                        (1|class), # random intercept for each class
                    data = schools)
summary(fit_crossed)
```

Is this model appropriate given the structure of the data?

# Nested Random Effect

```
fit_nested <- lmer(formula = extro ~ # Extroversion
                       open + # fixed effect of Openness
                       agree + # fixed effect of Agreeableness
                       social + # fixed effect of Social engagement
                       (1|school/class),
                       # random intercept for each class WITHIN the school
                     data = schools)
summary(fit_nested)
```

This is a shorthand for:

```
fit_nested_short <- lmer(formula = extro ~
                               open +
                               agree +
                               social +
                               (1|school) + # random intercept for each school
                               (1|class:school),
                               # and for each class WITHIN school
                     data = schools)
summary(fit_nested_short)
```

# Modeling the data using LME models: this is really the end

# Random slopes? (Harrison et al. 2018)

Often, researchers fit random intercepts to control for non-independence among measurements of a statistical group (e.g. birds within a woodland), but force variables to have a common slope across all experimental units. However, there is growing evidence that researchers should be fitting random slopes as standard practice in (G)LMMs. Random slope models allow the coefficient of a predictor to vary based on clustering/non-independence in the data.

In our bird example above, we might fit a random slope for the effect of foraging rate on clutch mass given each individual bird ID. That is, the magnitude of the effect foraging rate on resultant clutch mass differs among birds.

# Random slopes? (Harrison et al. 2018)

Schielzeth & Forstmeier (2009) found that including random slopes controls Type I error rate (yields more accurate $p$ values), but also gives more power to detect among individual variation. Barr et al. (2013) suggest that researchers should fit the **maximal random effects structure possible** for the data. That is, if there are four predictors under consideration, all four should be allowed to have random slopes. However, we believe this is unrealistic because random slope models require large numbers of data to estimate variances and covariances accurately (Bates et al., 2015a).

# Model complexity?

Guidelines for the ideal ratio of data points ($n$) to estimated parameters ($k$) vary widely (see Forstmeier & Schielzeth, 2011). Crawley (2013) suggests a minimum $n/k$ of 3, though we argue this is very low and that an $n/k$ of 10 is more conservative. A 'simple' model containing a three-way interaction between continuous predictors, all that interaction's daughter terms, and a single random intercept needs to estimate eight parameters, so requires a dataset of a minimum $n$ of 80 using this rule.

# Fit and performance?

The strength of the Nakagawa & Schielzeth (2013) method for GLMMs is that it returns two complementary $R^2$ values: the marginal $R^2$ encompassing variance explained by only the fixed effects, and the conditional $R^2$ comprising variance explained by both fixed and random effects i.e. the variance explained by the whole model (Nakagawa & Schielzeth, 2013). Ideally, both should be reported in publications as they provide different information; which one is more 'useful' may depend on the rationale for specifying random effects in the first instance.

Here is how you can compute both types of $R^2$ using R:

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

# Popularity data

On the pupil (`pupil`) level, we have the outcome variable

- popularity (`popularity`), measured by a self-rating scale that ranges from 0 (very unpopular) to 10 (very popular).

We have two explanatory variables on the pupil level:

- pupil gender (`sex`) (`0`=boy, `1`=girl),
- pupil extraversion (`extrav`), measured on a selfrating scale ranging from 1–10,

and one class level explanatory variable

- teacher experience (`texp`), in years, ranging from 2–25.

There are data on 2000 pupils in 100 classes, so the average class size is 20 pupils.

# Popularity data

```
popular <- read.csv("popular.csv")
popular <- popular %>% select(pupil, class, extrav, sex, te
head(popular) # we have a look at the first 6 observations
```

```
##   pupil class extrav sex texp popular
## 1     1     1      5   1   24     6.3
## 2     2     1      7   0   24     4.9
## 3     3     1      4   1   24     5.3
## 4     4     1      3   1   24     4.7
## 5     5     1      5   1   24     6.0
## 6     6     1      4   0   24     4.7
```

# Popularity data: structure of random effects

Variables?

# Popularity data: intecept only model

```
fit <- lmer(popular ~
              1 +
              (1|class),
            data = popular)
summary(fit)
```

# Popularity data: first level predictors

```
fit <- lmer(popular ~
              extrav +
              sex +
              (1|class),
            data = popular)
summary(fit)
```

# Popularity data: first and second level predictors

```
fit <- lmer(popular ~
                extrav +
                sex +
                texp +
                (1|class),
            data = popular)
summary(fit)
```

# Popularity data: first and second level predictors with random slopes (1)

```
fit <- lmer(popular ~
                extrav +
                sex +
                texp +
                (sex + extrav|class), data = popular)
summary(fit)
```

```
ranova(fit)
```

# Popularity data: first and second level predictors with random slopes (2)

```
fit <- lmer(popular ~
              extrav +
              sex +
              texp +
              (extrav|class),
            data = popular)
summary(fit)
```

# Popularity data: first and second level predictors with random slopes and cross-level interaction

```
fit <- lmer(popular ~
              extrav * texp +
              sex +
              (extrav|class), data = popular)
summary(fit)
```

```
r.squaredGLMM(fit)
```