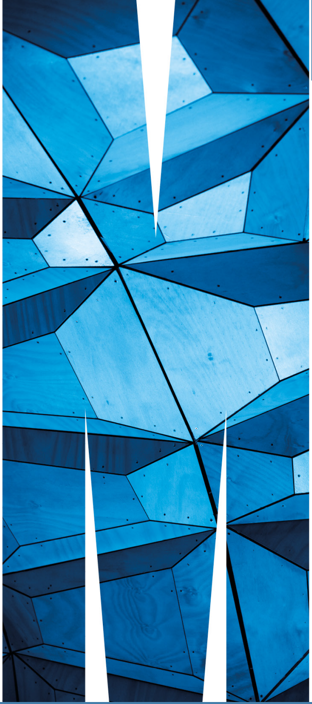
MONASH University

MONASH
BUSINESS
SCHOOL



Introduction to Multi-level Models using R

Professor Di Cook, Econometrics and Business Statistics

Workshop for the Institute for Safety, Compensation and Recovery Research

Outline

- Session 1: Basic models, fitting multiple separate models
- Session 2: Putting it together, using mixed effects models
- Session 3: Summarising and visualising models
- Session 4: Advanced modeling

Session 4

- **Advanced modeling**

Topics

- Cross-validation
- Lineups
- Non-linear mixed effects
- Categorical response variable
- Gapminder data, again

Cross-validation

A step up from leave-one-out diagnostics is cross-validation. Fit the model many times, and examine the variation in the estimates, and model summary statistics. Here is a simple example, for a basic linear model:

Sample within groups with given probability

```
random_group <- function(n, probs) {
  probs <- probs / sum(probs)
  g <- findInterval(seq(0, 1, length = n), c(0, cumsum(probs))
    rightmost.closed = TRUE)
  names(probs)[sample(g)]
}
```

Partition the data into samples, here training and test

```
partition <- function(df, n, probs) {
  replicate(n, split(df, random_group(nrow(df),
                                      probs)), FALSE) %>%
    transpose() %>%
    as_data_frame()
}
boot <- partition(mtcars, 100, c(training = 0.8, test = 0.2))
```

Function for computing the mean square error.

```
msd <- function(x, y) sqrt(mean((x - y) ^ 2))
```

Fit an lm to each sample of training data, prediction the test data, compute the mean square error for each.

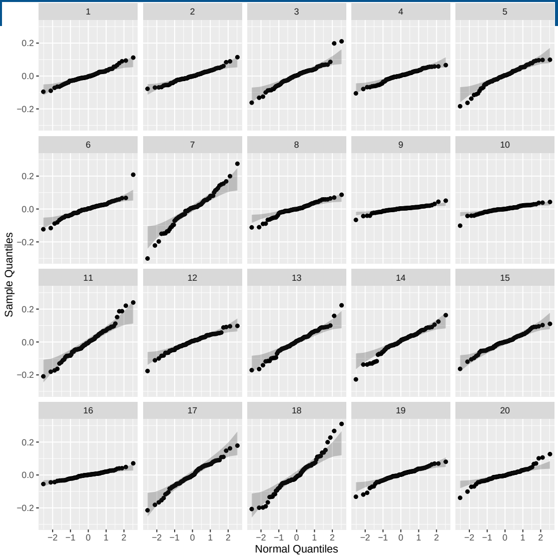
```
boot <- boot %>% mutate(
  # Fit the models
  models = map(training, ~ lm(mpg ~ wt, data = .)),
  # Make predictions on test data
  preds = map2(models, test, predict),
  diffs = map2(preds, test %>% map("mpg"), msd)
)

unlist(boot$diffs)
#   [1] 3.1 2.7 3.7 4.3 4.3 2.6 2.5 3.5 5.0 1.5 1.8 1.7 2.6 3.
#  [18] 2.6 3.8 3.5 2.6 3.3 3.5 5.5 3.4 3.3 3.9 3.7 2.9 2.8 3.
#  [35] 4.2 3.3 3.6 2.3 2.6 4.2 3.6 3.2 3.9 2.3 2.2 2.2 2.1 3.
#  [52] 3.6 2.2 3.9 3.4 3.7 3.8 3.6 1.7 4.8 3.4 2.7 4.0 4.0 3.
#  [69] 3.8 2.6 3.6 2.6 1.6 3.2 4.0 2.9 2.5 3.3 3.7 2.5 2.3 4.
#  [86] 2.2 3.7 3.5 3.1 4.1 3.4 1.9 4.8 4.0 3.2 2.5 4.3 3.4 2.
```

Lineups

- Have trouble reading normal probability plots and residual plots?
- Show the plots embedded in a page of plots of data simulated from the model fit, or from the theoretical distribution

```r
# Lineup of random intercepts
qplot(sample = X.Intercept., data = b0) %+%
    lineup(true = b0, sample = sim.b0, pos=5) +
    facet_wrap(~ .sample, ncol = 5) +
    geom_ribbon(aes(x = x, ymin = band.lower, ymax = band.uppe
    xlab("Normal Quantiles") + ylab("Sample Quantiles")
```

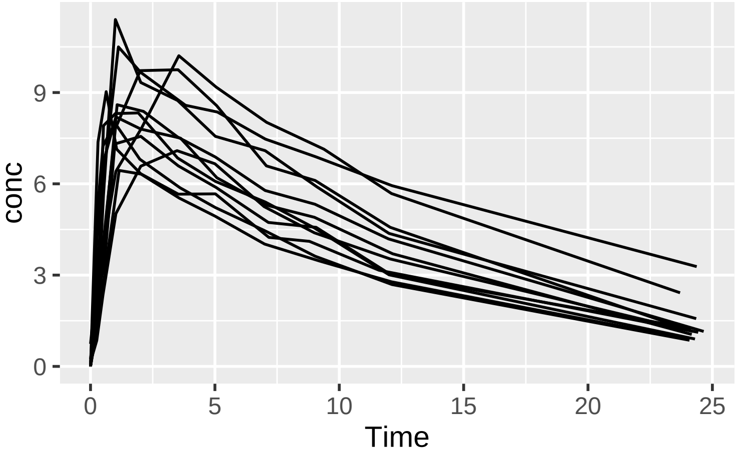

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
0.2
0.0
−0.2
Sample Quantiles
−2 −1 0 1 2
Normal Quantiles

- Ask “Which of these is the most different from the others?”"
- If it is the plot of the actual residuals then it would suggest that the model assumptions might not be satisfied, or the model hasn’t captured the structure in the data very well.
- Here the normal probability plot of the residuals is $\sqrt{3^2 + 4^2}$

Non-linear mixed effects models

The package `nlme` can fit nonlinear mixed effects models.

```
ggplot(Theoph, aes(x=Time, y=conc, group=Subject)) +
  geom_line()
```



Categorical response

We can use the generalized linear model framework for mixed effects models too:

```
library(lme4)
fit.mixed <- lmer(response ~ effect + (1+effect|id)
                , family="binomial"
                , data=dat)
```


Gapminder data

```
library("gapminder")
gap_lmer <- lmer(lifeExp ~ year + (1 | country),
                 data=gapminder)
gap_lmer_fit <- augment(gap_lmer)
```

Your turn

- Make a plot of the model
- Write down the equation corresponding to this model
- Refine the model, do a residual analysis, the influence and outlier detection, and find countries where the model doesn't fit well

Your turn

Discuss the relative merits of the mixed effects (multilevel) model vs individual linear model fits, using the gapminder to motivate the discussion.

Credits

Notes prepared by Di Cook, using material developed by Hadley Wickham, Heike Hofmann and Adam Loy.