Week 2: Logistic Multilevel Modelling

- Last semester you learned about using logistic regression for binary outcomes
- A quick review: when there's a single binary outcome, it's intuitive that logistic regression is the right option. A common mistake is that this is still true if you have a bunch of binary trials and aggregate the results.
    - For example, if you're talking about accuracy or fixation proportion over a set of trials, 85% correct or 30% target fixation sound like continuous variables but they're not.
    - You can tell they're not because (1) they are bounded – proportions are never less than 0.0 or higher than 1.0, and (2) they have a very specific non-uniform variance pattern
- This can have real consequences for analysis. Here's an example study of a treatment effect – which group benefitted the most from treatment?
    - If you do a linear regression, it looks like the biggest benefits were in the moderate and severe groups, and the mild group had a smaller benefit.
        - But notice that the mild group is close to the ceiling. Linear regression doesn't know that, but logistic regression does and knows that variance is smaller near the ends.
    - In a logistic regression, the benefit for the mild and moderate group is approximately the same, and the severe group is the one showing a substantially larger benefit of treatment
- Ok, so logistic regression is a model of the binomial process that generated your binary data
    - To fit that model, it's not enough to know that overall accuracy was, say, 90%, you need to specify whether that was 9 out of 10 trials or 90 out of 100
    - You can specify this as a vector of individual 1's and 0's, in a more compact way as the number (or count) of 1's and 0's
    - The outcome that is being modelled is log-odds, also known as the logit, here is the formula for it, which you can see is different from the formula for proportions
    - When the probability is 50%, the log odds is 0. When the probability is less than 50, the log odds is negative; when the probability is higher, the log odds is positive. So log odds are not bounded at 0 and 1 the way proportions are
    - But there's a bit of a problem when proportions are exactly 0 or 1: the logit is undefined for those values (Inf), which makes it hard for logistic models to fit data with such extreme values. This is something to keep in mind if you're trying to model very rare outcomes – a different kind of model might be required.
- Let's look at an example: a word learning study comparing control participants and participants with aphasia
- There's a learning phase, then a test, then a follow-up 1 week later
- Let's just look at the test data for now – the test that was immediately after training and the follow-up 1 week later
    - We can ask whether the patients were less successful than controls were at learning these new words? (Lower test performance)
    - Did recall decrease from immediate test to 1-week follow-up?
    - Was retention (recall decrease) different for the two groups?
    - One might be tempted to use a 2x2 ANOVA (2 groups x 2 test phases)

        - Take a moment to think about what would be right about that approach and what would be wrong
    - What it gets right: group as a between-participant variable, Phase as a within-participant variable (MLM is a more flexible version of repeated measures ANOVA)
        - Phase-by-group interaction to test group differences in retention
    - What it gets wrong: ANOVA would treat Accuracy as a continuous linear variable, but it is an aggregated binary variable
- The good news is that fitting a logistic multilevel model is very similar to fitting a linear MLM, there are just three differences
        - You need to use the glmer() function instead of lmer() -- *generalised* linear mixed effects regression
        - Your outcome variable needs to be either a binary vector of 1's and 0's or two columns with paired counts of 1's and 0's – that's what we'll use here
        - And you need to add "family=binomial" as an option so glmer() knows which generalised linear model you're using
        - Here's what the model code looks like:
            - Use "cbind" (column bind) to make the outcome pair of columns. These values are each participants number of correct responses and number of errors in each test phase.
            - The fixed effects are test Phase and participant Group (the asterisk is a compact notation meaning both main effects and the interaction between them)
            - The random effects are by-participant random intercepts (implicit) and slopes – that means random by-participant variability in performance (intercept) and retention (slope)
            - The data – only the test phase data are included
            - And the family option
    - Note: logistic MLMs are slower to fit and are prone to convergence problems
        - This may require simplifying random effect structures (we'll talk more about that later in module)
        - Don't panic if you get convergence warnings or this singular fit message. These are not errors, the algorithm did produce a model, but it is telling you to carefully check your model and be cautious about interpreting the estimated parameters
- Looks like there is a significant effect of Phase -- a decrease in performance from the immediate phase to the 1-week follow-up
        - A significant effect of group – patients perform worse than controls
        - And no interaction
        - Note that these parameter estimates correspond to *simple* effects, not *main* effects: that test phase parameter is estimate for the control group and the group effect is estimated for the follow-up test phase
- When interpreting parameter estimates for categorical variables, it is important to keep in mind how the contrasts are coded. In R, the default is "treatment" coding, which produces simple effects as I just described.
        - This can be very sensible for some situations, like a treatment study: you get estimates for the control group (for example, is there natural recovery or a placebo effect or something like that) and baseline difference between control and

treatment conditions (did your randomisation work properly), then the interaction gives you the differential effect of treatment
    - But in many of our studies, we want "main" effects. To get those, you need to use sum coding and you can specify that in the model code
        - These parameter estimates now correspond to the main effects of test Phase (across both groups) and Group (across both test phases)
- Plotting the model fits from logistic models can be a bit tricky, but conveniently the fitted() function returns proportions, so you can do something like this – the violins show the distributions of the observed data and the points show the model-estimated means
- I mentioned earlier that the computational demands of logistic models may require simplifying random effects. There are two key issues regarding getting the random effects structure right
        - When you have within-subject variables, omitting their random effect tends to inflate false positive rates for the corresponding fixed effect. So if you want to make inferences about a particular (within-subject) fixed effect, you'll want to make sure you include the corresponding random effect
    - However, random effects require lots of data to estimate so it's easy to over-parameterise a model with too many random effects. When you have convergence problems, there may be other fixed effect estimates that (nearly) as good as the ones you got, which is a problem if you want to make inferences based on the specific estimates you got.
    - So, as a general strategy, I recommend starting with a maximal random effect structure.
        - If it converges well, great.
        - If you have convergence problems, check the model summary random effect variance-covariance matrix and look for values that are very small or unrealistic – these are good candidates for removing to simplify the structure
        - You can also compare the fixed effect estimates (and SE) under different random effect structures – ideally, they should stay about the same. If they are substantially different, then you've got a fragile model and you should be *very* cautious about interpreting the results (and maybe consider a different statistical approach).
        - We'll keep revisiting issues related to random effect structures each week, from slightly different angles and with increasing sophistication

Week 2: Longitudinal Data Analysis using Multilevel Modelling

- Longitudinal data are a natural application domain for MLM
    - Longitudinal measurements are *nested* within subjects (by definition)
    - Longitudinal measurements are related by a continuous variable, spacing can be uneven across participants, and data can be missing -- these are problems rmANOVA
    - Trajectories of longitudinal change can be nonlinear (we'll get to that next week)
  - We've already seen some examples of this:
    - the weight maintenance data from the Week 1 lab
    - the visual search example wasn't longitudinal, but the idea was the same
- Let's consider another example where we can think about longitudinal issues some more
    - These are Public Health England data on various mental health indicators. For this example, let's focus on county-level percentage of adults who are physically active at recommended levels
- The indicator ID for percentage of physically active adults is 90275
- Here's a plot of the data by region of England from 2012 to 2015
    - We can ask whether the baseline rates differ and whether the slopes of the change differed during this time window
- To answer the first question, we need to check what we mean by "baseline"
    - The intercept coefficient could answer this question
  - But those will be estimated at Year=0 and the question is not about whether adults were physically active the year Jesus was born.
    - We need to adjust the time variable so that 2012 corresponds to time 0 (and we can select just the physical activity values while we're at it)
    - Now Time is a variable just like Year, but going from 0 to 3 instead of 2012 to 2015
- Now we can fit the models
    - The base model just has an overall effect of time, and by-county random intercepts and slopes
    - Then we can add baseline differences between regions
    - And the full model will have an effect of time, differences between regions, and region-by-time interaction (that is, slope differences between regions)
    - Notice that, as usual, the random effects remain the same across all models, we're only changing the fixed effects
- The model comparison reveals that there were significant baseline (2012) differences between regions in terms of % of physically active adults, but not in the slope of change over the next 3 years
    - When doing the model comparisons, the degrees of freedom are a good quick check: df=8 here, which makes sense because there are 9 regions. So when we add a region effect (or the region-by-time interaction), one region becomes the reference and the model estimates 8 additional coefficients for each of the other regions
- We can plot the model-estimated trajectories using the effects package, like we've done with previous examples
- Another option is to plot the parameter estimates themselves. I think this approach gets used more in epidemiology and biostats than in psychology, but it can be quite useful
    - Side note: it took me a while to work out the data wrangling for pulling the estimates from the model and setting them up to make this plot
    - I spend at least half of my analysis time doing data wrangling and it's often closer to 80%. That's not a major part of this course, but it will be once you're analysing your own data.