1: Meta-analysis

Emma James

1 Introduction

> **Learning objectives**
>
> As a reminder, the learning objectives for this week are as follows:
>
> 1. Explain the purpose and value of meta-analytic approaches
> 2. Describe key stages of running a meta-analysis
> 3. Compare different ways to weight and pool effect sizes, and implement them using the *metafor* package
> 4. Assess and mitigate the impact of publication bias
> 5. Produce and interpret standard visualisations of meta-analytic results

This week's lecture introduced you to *meta-analysis,* a computational technique for combining results across several studies. We walked through the key steps involved in gathering data systematically, and outlined different ways that these can be combined in the statistical model.

In this practical session, we will focus primarily on what we do *after* the data has been gathered. How can we pool effect sizes according to our analytical choices, consider evidence for publication bias, and communicate our findings to a wider audience? We will start by using a ready loadable psychology dataset from the *metadat* package, then progress to a dataset curated here in York.

---

2 Setting up

2.1 Packages

Today's practical will use the following packages. You will need to install each one, before loading them at the top of your script.

```
library(tidyverse)  # data wrangling tools
library(metadat)    # meta-analytic datasets
library(metafor)    # core meta-analysis tools
```

3 Part 1: Core example

3.1 Introduction to the dataset

We will start by walking through the key analytic steps with a dataset that has already been curated. We will run an analysis to assess evidence in favour of the "Red-Romance Hypothesis" (Elliot & Niesta, 2008): a theory grounded in evolutionary and cultural psychology that the colour red enhances perceptions of attractiveness. The dataset was curated by Lehmann et al (2018), and released through the `metadat` package that we have loaded above.

Start by loading it into your environment. Inspect the dataframe that you have loaded, and use the help files to understand the variables.

```
# Load  metadat dataset into environment
rr_dat <- dat.lehmann2018

# Inspect dataset information
help(dat.lehmann2018)
```

You should be able to see that the dataset consists of a single effect size (`yi`) per row, with various other columns to code information about the study it was collected from.

Let's explore some data characteristics:

- Use `nrow()` to count the number of effect sizes (rows) in the dataset.
- You can count the number of different studies using `length(unique(rr_dat$Full_Citation)`.
- Use the `range()` or `min()`/`max()` functions to assess the range of sample sizes (`Total.SampleSize`) in the studies.
- Count the number of studies with male versus female participants using `table(rr_dat$Gender)`.
- How many studies were preregistered? How many used within versus between subject designs?

```
# Number of effect sizes
nrow(rr_dat)

# Number of different studies
length(unique(rr_dat$Full_Citation))

# Range of sample sizes
range(rr_dat$Total.SampleSize)

# Number of male vs female
table(rr_dat$Gender)

# Number preregistered vs not
table(rr_dat$Preregistered)

# Number of within versus between subject designs
table(rr_dat$Design)
```

Lehmann et al's meta-analysis considered the evidence separately for 'males rating females' versus 'females rating males'. Let's start by focusing on 'males rating females'. Filter the dataset accordingly, and then use the `forest()` function to plot the effect sizes.

```
# Select only effect sizes corresponding to male participants
rr_m_ppts <- filter(rr_dat, Gender == "Males")

# Plot
forest(rr_m_ppts$yi, rr_m_ppts$vi, slab = rr_m_ppts$Short_Title)
```

We have provided the `forest()` function with the list of standardised effect sizes, sampling variances, and list of study labels. How many studies look as if they show significant effects (confidence intervals that do not cross zero)? Are they all in the same direction?

3.2 Computing pooled effect sizes

In the lecture, you learned about fixed and random effects meta-analyses. Let's have a go at running both.

3.2.1 Fixed effects meta-analysis

In a fixed effects analysis, we assume that all studies are measuring the same underlying true effect size, and calculate the average effect size across the included studies. The `metafor` package terms this an "equal effects" model, as a better descriptive name. Use the `rma()` function to run this as follows:

```
# Fixed effects model
rr_fixed_m <- rma(yi = yi, vi = vi,
                  method = "FE", data = rr_m_ppts)
```

We've specified the variables with our effect size estimates and sampling variance, which have conveniently been named the same already by the authors. We've then specified the "FE" method for fixed effects, and the dataset to use.

We can use the `summary()` function on the model object to inspect the results.

```
summary(rr_fixed_m)
```

The output tells us that we ran a Fixed-Effects model with $k = 45$ effect sizes. The pooled estimate for the standardised mean difference is ~0.15, showing a small but significant effect of the colour red on men's attractiveness ratings of women. However, it also tells us that there is substantial heterogeneity in study outcomes, indicating that the studies might be capturing varied effect sizes due to potential moderators or different study designs.

3.2.2 Random effects meta-analysis

The heterogeneity indicates that a random effects model might be more appropriate— i.e., one that assumes that the samples in the meta-analysis are a random sample from a larger population of studies that could plausibly have been/will be conducted. This makes sense when we consider the study descriptions: they use a range of different outcome measures, and often include methodological variation within the same publications (as described in the study titles).

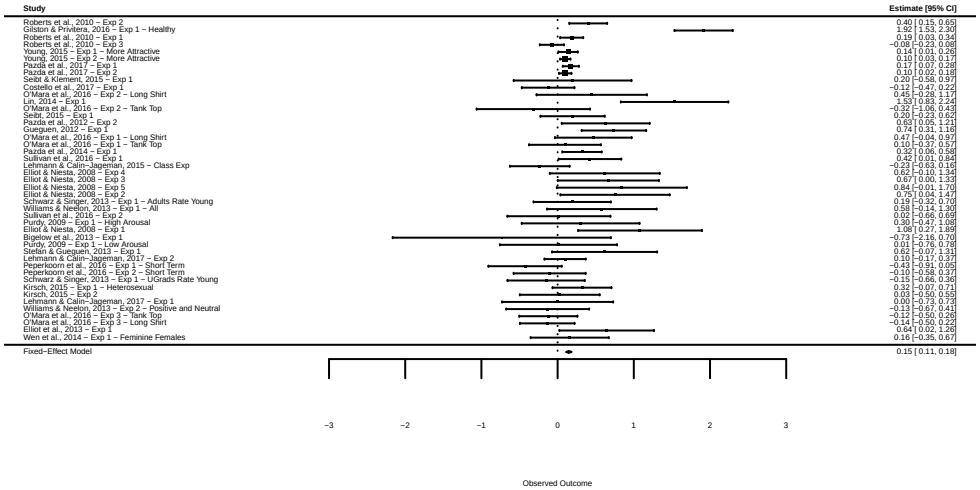
We can run this by simply changing the method in the meta-analysis function. The package provides us with several different options that have different ways of estimating the heterogeneity between studies (feel free to read more about these!), but for now we will stick with the default option.

```
# Random effects model
rr_random_m <- rma(yi = yi, vi = vi,
                   method = "REML", data = rr_m_ppts)
```

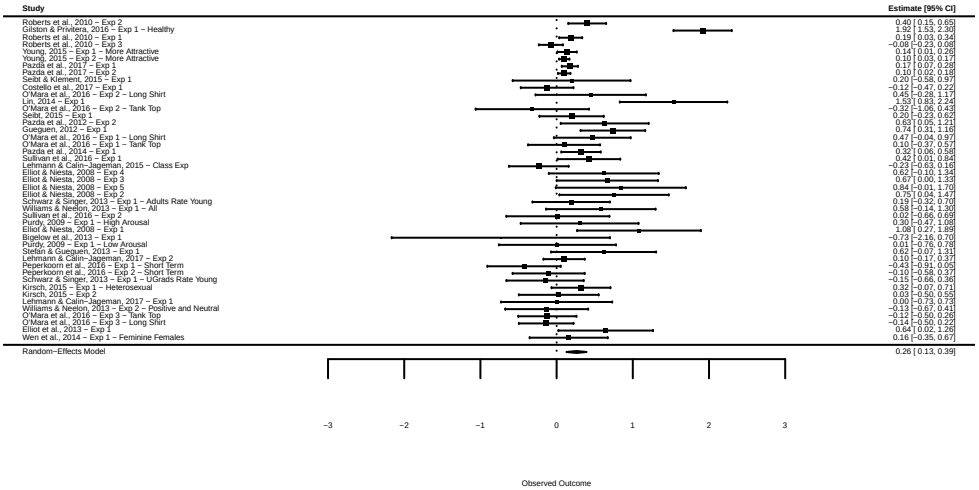
Use the `summary()` function as above to inspect the model output. How has the pooled estimate for the standardised mean difference changed?

We can also inspect the results by including the pooled estimates on our forest plot. The `forest()` function can also take the meta-analysis model objects directly (rather than the separate lists of effect sizes and variances). Try this now with each of your models.

```
forest(rr_fixed_m, slab = rr_m_ppts$Short_Title)
```




```
forest(rr_random_m, slab = rr_m_ppts$Short_Title)
```



3.3 Weighting studies

We saw when we loaded the dataset that the variation in sample size was very large. We should consider this in our analysis: a very large study likely provides a better estimate of the true effect size than a very small study.

3.3.1 Under the hood: inverse variance

One of the most common ways to do this is via inverse-variance, which the `rma()` function does by default. If you inspect your `rr_m_ppts`data frame, you will see that the smaller samples generally have the highest variance (`vi`). You might be able to see this more clearly via a quick scatterplot:

```
# Inspect relationship between
plot(rr_m_ppts$Total.SampleSize, rr_m_ppts$vi)

# Now plot inverse variance instead
plot(rr_m_ppts$Total.SampleSize, 1/rr_m_ppts$vi)
```

To use this variance in the analysis, the model uses the *inverse* (`1/variance`), so that smaller samples have the smallest weighting. The function does this under the hood, but you can see what this looks like by computing it for yourself. Recreate your plot using `1/rr_m_ppts$vi` as your y variable.

You should now see a steady positive trend for the majority of studies, with higher inverse variance for large studies. A few studies buck this trend, with little variance in their estimates despite small-medium sample sizes. Can you identify them in the dataset?

3.3.2 Changing weights in the meta-analysis

Because weighting by inverse variance is a very common and sensible approach, the `rma()` meta-analysis function does this by default. This means you've already weighted your samples in your above models!

To compare, try fitting two new models (fixed, random) and include the argument `weighted = FALSE`. Use a different object name to store the model output, so that you can inspect the differences in output when you call `summary()`.

Alternatively, you can try weighting by sample size alone, by setting `weights = N`.

3.4 Inspecting for publication bias

Finally, let's consider whether this analysis might be affected by publication bias. Let's use the random effects model given it would be the more appropriate choice here, and create a funnel plot:

```
funnel(rr_random_m)
```

Take a look at this linked guide for interpreting the funnel plots. What do you conclude?

You can also test for asymmetry using Egger's regression test. Is there statistical evidence for assymmetry?

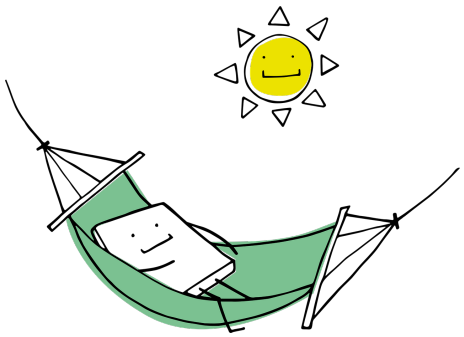
```
regtest(rr_random_m)
```

For comparison, inspect the evidence for publication bias using female participants instead. You'll need to subset the data, run a new random effects model, and then inspect for asymmetry. What do you conclude about publication bias in this area of research?

```
# Select only effect sizes corresponding to female participants
rr_f_ppts <- filter(rr_dat, Gender == "Females")
rr_random_f <- rma(yi = yi, vi = vi, data = rr_f_ppts)
funnel(rr_random_f)
regtest(rr_random_f)
```

Take a break!

If you haven't already taken a break today, we suggest you so before you start the next section.



---

4 Part 2: Apply your knowledge

4.1 Introduction to the dataset

Let's jump in with a new dataset! This time we will use a recently published meta-analysis that examined the relationship between procedural learning and reading ability (Oliveira et al., 2023).

The Procedural Deficit Hypothesis proposes that reading and language disorders stem from a difficulty in the procedural learning system - i.e., the system that underpins the learning of probabilistic knowledge (Ullman et al., 2020). Procedural learning has often been measured using Serial Reaction Time (SRT) tasks: participants respond to a repeating sequence of stimuli, and their improvements in speed over several blocks are measured as an index of learning. However, evidence that SRT performance is associated with reading ability is mixed, with some studies showing a strong association and others finding nothing at all. Many studies in this area suffer from small sample sizes, making it hard to draw sound conclusions.

This meta-analysis was conducted by a former PhD student at York, Dr Catia Oliveira, together with Professor Lisa Henderson and Dr Emma Hayiou-Thomas. We will walk through one new step of the process together in extracting effect sizes and calculating pooled variances. Then you will apply your new meta-analysis skills.

4.2 Updating the dataset

Oliveira et al. made their protocol and analyses openly available on the Open Science Framework. This means that we can directly download the data into R as follows.

```
# Download original dataset
srt_dat <- read.csv("https://osf.io/yj9bs/?action=download")
```

Let's focus just on the relationship between SRT and reading, for simplicity, and stick to word/nonword reading tasks. We're going to add a new study to the dataset, so for the sake of this practical we will also strip back the dataset to a few key variables (otherwise adding the new dataset will take you forever!).

```r
srt_dat2 <- srt_dat %>%
  filter(Component == "Reading") %>%
  filter(Task == "Word Reading" | Task == "Nonword Reading") %>%
  select(Study, N, Age, Task, Group, Cor)
```

Use your R skills to answer the following questions about the dataset:

- How many effect sizes are there?
- How many unique studies do they come from?
- What is the range of average ages?
- How many samples are from the different participant groups?

```r
# Number of effect sizes
nrow(srt_dat2)

# Number of unique studies
length(unique(srt_dat2$Study))

# Range of ages
range(srt_dat2$Age)

# Effect sizes from different participant groups
table(srt_dat2$Group)
```

This meta-analysis included data that were published up until November 2020, but many studies on this topic have been published since then. Let's add this study published by van Witteloostuijn et al. (2021), which adds 2 new correlation effect sizes between SRT and word/nonword reading, for each of two groups (typically developing children, children with developmental dyslexia). Use Appendix 1 of the paper to replace the NAs below with the relevant numbers, and run the code to add 4 new rows to the dataset.

```r
srt_dat3 <- srt_dat2 %>%

  add_row(Study = "van Witteloostuijn et al. (2021)",
          N = NA,
          Age = 9.66,
          Task = "Word Reading",
          Group = "TD",
          Cor = NA) %>%

    add_row(Study = "van Witteloostuijn et al. (2021)",
          N = NA,
          Age = 9.66,
          Task = "Nonword Reading",
          Group = "TD",
          Cor = NA) %>%

    add_row(Study = "van Witteloostuijn et al. (2021)",
```

```
          N = NA,
          Age = 9.83,
          Task = "Word Reading",
          Group = "DD",
          Cor = NA) %>%

    add_row(Study = "van Witteloostuijn et al. (2021)",
          N = NA,
          Age = 9.83,
          Task = "Nonword Reading",
          Group = "DD",
          Cor = NA)
```

4.3 Compute sampling variance

In the dataset above, the effect sizes and sampling variances had already been calculated for us. In the current dataset, we have extracted a consistent measure of effect size (the correlation coefficient representing the relationship between SRT task performance and reading ability), but still need to compute the sampling variance around the estimates to capture uncertainty. The *metafor* package provides a handy function for doing this: simply replace the NAs in the below code.

```
srt_dat3 <- escalc(measure = "COR",   # specify effect size measure
                   ri = NA,           # specify correlation variable
                   ni = NA,           # specify sample size variable
                   data = srt_dat3)   # dataset
```

You should now see two new columns nicely formatted for the meta analysis:

- **yi***:* your effect size, which in this case is identical to the one in the Cor column as we haven't asked it to do any kind of transformation
- **vi:** your sampling variances, which have now been computed based on the correlation coefficient and sample size

Now you're ready to go!

4.4 Conduct your meta-analysis

You can now finish conducting your meta-analysis. Conduct the following steps, and write notes to justify your decisions along the way:

- Decide on the appropriate model and weighting strategy
- Run the model and interpret the output
- Create a forest plot
- Evaluate evidence for publication bias

```r
# Random effects model
srt_mod <- rma(yi = yi, vi = vi,
               method = "REML", data = srt_dat3)
summary(srt_mod)

# Create forest plot
forest(srt_mod, slab = srt_dat3$Study)

# Examine evidence for publication bias
funnel(srt_mod)
regtest(srt_mod)
```

5 Challenge yourself!

Often we are interested in how effect sizes vary across different conditions. In this case, the researchers were interested in whether the relationship between procedural learning (SRT) and reading ability might differ depending on the participant groups tested.

If you finish the practical in good time or want to push your skills further beyond the end of today's session, see if you can test whether participant `Group` is a moderator in your analysis. You can look at the help files or use Google to find out how to include this extra predictor.

```r
# mod_mod <- rma(yi = yi, vi = vi, mods = Group,
#                method = "REML", data = srt_dat3)
# summary(srt_mod)
```

6 Summary

6.1 Recap

This week, we've seen how meta-analyses can be used to synthesise evidence across a range of different studies to answer a research question. The starting point of any meta-analysis is a systematic literature search—although this would have been far too large a task for today's practical! From there, we've learned to extract effect sizes, calculate sampling variances, and pool them according to principled decisions about how they should be weighted and combined in the model. We saw how to inspect for possible publication bias, and create gold-standard visualisations of the results. Well done!

We ran a basic version of these models here, but there is much more you can do to correct for small sample sizes and adjust for the fact that some estimates come from the same study (i.e., there were different correlation coefficients for different measures of reading in the same sample). If you are interested in these issues, you can access the full paper here or download the full dataset and analysis scripts.

6.2 Further learning

Make sure that you have completed the core reading for this week, and consider reading further resources to consolidate your understanding. As you'll have seen today, the code for actually running analyses is quite sparse(!), but it's important that you read around the decisions that need to be made along the way.

If you come to do meta-analysis in the future or want further details on the different steps that we have covered, then there is an excellent textbook available for free online: Harrer et al. (2021). *Doing Meta-Analysis with R: A Hands-On Guide*.

! Feedback!

These practical sessions are new this year. Please take a moment to rate the difficulty and length of these practical activities via this survey, and let us know of any issues you encountered or aspects you didn't understand (positive feedback is also welcome!). You will need to be logged in with your university account to respond, but your submission will be anonymous.
For more general questions about the practical, please post them on the VLE Discussion Board.