The general linear model – Lecture 4
Interactions

Jan Vanhove
https://janhove.github.io

Ghent, 12–14 July 2023

Often, researchers aren't so much interested in how one predictor variable relates to some outcome. Rather, they're interested in how the relationship between one predictor and the outcome differs depending on another predictor. That is, they're interested in the interaction between two predictors.

Consider Figure 1, which shows four examples of what the joint effect of reading experience and word frequency on reading speed could look like. Note that in three out of four cases, the lines are not parallel to each other; in these cases, the effects of reading experience and word frequency on reading speed interact. In one case, the lines do run in parallel, and the effects of reading experience and word frequency on reading speed do not interact; that is, they are additive.

1 Interactions between two binary predictors

In Berthele (2012), future teachers were asked to rate the academic potential of a German-speaking boy based on a short recording in which he spoke French. About half of the future teachers were told that the boy's name was Luca (a typical Swiss name); the rest were told that the boy's name was Dragan (a name suggesting a Balkan migration background). Moreover, for about half of the participants, the recording contained code-switches from German; for about half, it didn't. Berthele (2012) wanted to find out how the purported name and the presence or absence of code-switches affect the teacher trainees' judgements of the boy's academic potential.

1.1 Data visualisation

Let's read in the data and plot them.

```
> library(tidyverse)
> library(here)
>
> d <- read_csv(here("data", "berthele2012.csv"))
>
> ggplot(d,
+        aes(x = Name,
+            y = Potential)) +
```

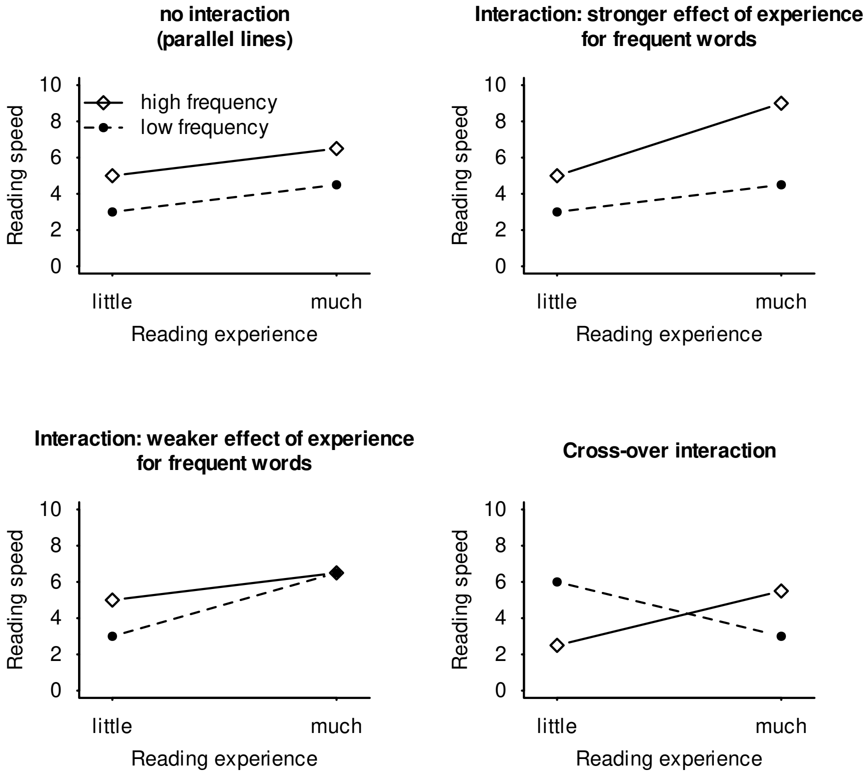

Figure 1: If the effects of reading experience and word frequency on reading speed interact, then the effect of reading experience on reading speed differs for different levels of word frequency. Or, equivalently, the effect of word frequency on reading speed differs for different levels of reading experience. This is reflected in the non-parallel lines. (The units on the y-axis in this example are arbitrary.)

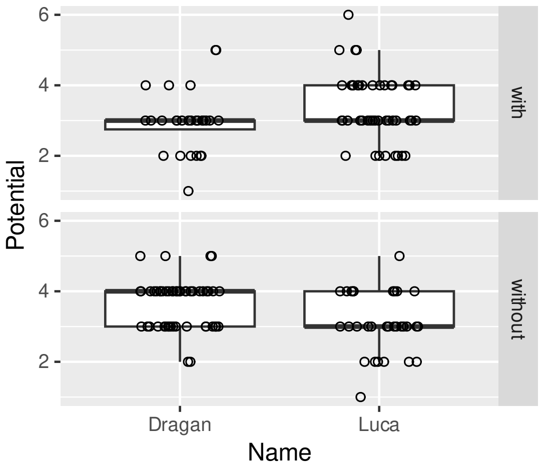

Figure 2: A first attempt at plotting the data. The patterns in the data aren't so clear because the data are too coarse for boxplots.

```
+   geom_boxplot(outlier.shape = NA) +
+   geom_point(shape = 1,
+              position = position_jitter(width = 0.2, height = 0)) +
+   facet_grid(rows = vars(CS))
```

While boxplots are a reasonable first choice, Figure 2 suggests that these data may be too coarse to plot in this way. Alternatively, we could compute the mean potential rating for each combination of predictor variables and plot these means. But then we wouldn't know how the data underlying these means are distributed; Figure 3.

```
> summary_berthele <- d |>
+   group_by(Name, CS) |>
+   summarise(n = n(),
+             MeanRating = mean(Potential),
+             StdRating = sd(Potential),
+             .groups = "drop")
> summary_berthele

# A tibble: 4 x 5
  Name    CS          n MeanRating StdRating
  <chr>   <chr>   <int>      <dbl>     <dbl>
1 Dragan  with       28       2.96     0.881
2 Dragan  without    51       3.63     0.692
3 Luca    with       44       3.36     0.942
4 Luca    without    32       3.09     0.856

> ggplot(summary_berthele,
+        aes(x = Name,
+            y = MeanRating,
+            linetype = CS,
+            group = CS)) +
```

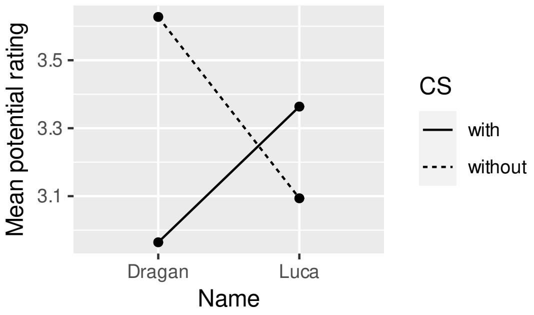

Figure 3: The trends in the data are clearer here, but we can't glean the distribution of the data from this plot.

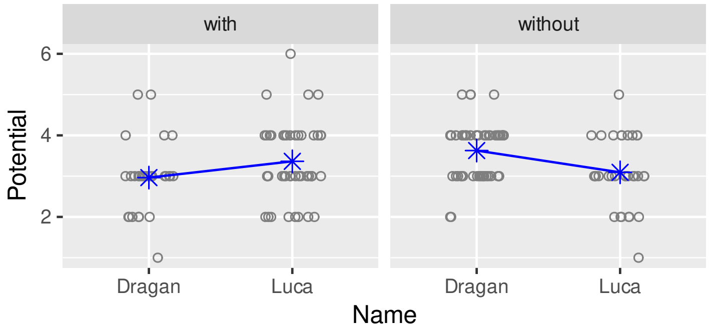

Figure 4: The best of both worlds.

```
+   geom_point() +
+   geom_line() +
+   ylab("Mean potential rating")
```

Luckily, we can have the best of both worlds. With the following commands, we plot the raw data as in Figure 2 and then add the mean trends to them; Figure 4.

```
> ggplot(d,
+        aes(x = Name,
+            y = Potential)) +
+   geom_point(shape = 1, colour = "grey50",
+              position = position_jitter(width = 0.2, height = 0)) +
+   geom_point(shape = 8, size = 3, colour = "blue",
+              data = summary_berthele,     # Data from different dataframe
+              aes(x = Name, y = MeanRating)) +
+   geom_line(colour = "blue",
+             data = summary_berthele,      # Data from different dataframe
+             aes(x = Name, y = MeanRating, group = CS)) +
+   facet_grid(cols = vars(CS))
```

Try out several plots. For group comparisons, boxplots are often a good choice, but it's often possible to improve on them. Don't hesitate to try out alternative plots.

Incidentally, it can take some time to find a good way to visualise your data. For the last graph, I had to tinker with the colour of the data points as we as with the colour, shape and size of the means. But ultimately, all of this is time well spent.

1.2 Model

We now turn our attention to the matter of modelling these data in the general linear model. Note that the graphs suggest that the purported name and the presence or absence of code-switches interact: If code-switches are present, 'Dragan's' academic potential is rated worse than 'Luca's', but if they are absent, the order is flipped. We want our model to capture this interplay between the two predictors. To that end, we will need four β parameter.

- β_0, the intercept, captures the baseline of the ratings.
- β_1 captures the difference between the cells depending on the purported name (Dragan vs. Luca).
- β_2 captures the difference between the cells depending on the presence or absence of code-switches.
- β_3 adjusts β_1 and β_2: How much larger or smaller is the difference between Dragan and Luca if there are code-switches compared to when there are no code-switches? Or, equivalently, how much larger or smaller is the difference between the presence and absence of code-switches depending on whether the purported name is Dragan or Luca?

The model equation is as follows:

$$y_i = \beta_0 + \beta_1 \cdot x_{1,i} + \beta_2 \cdot x_{2,i} + \beta_3 \cdot (x_{1,i} \cdot x_{2,i}) + \varepsilon_i.$$

- $x_{1,i}$ indicates whether the i-th participant was told that the boy's name was Dragan (1) or Luca (0).
- $x_{2,i}$ indicates whether the i-th participant rated a recording with (1) or without (0) code-switches.
- Consequently, β_0 represents the average rating by participants who've purportedly heard Luca (0) talk without code-switches (0).

The term $(x_{1,i} \cdot x_{2,i})$ may surprise you, but it does the job: The interaction term is a new variable that is the pointwise product of the two predictor variables. For the four cells in the present design, this new variable takes on two values:

- Luca (0) without Codeswitches (0): $x_{1,i} \cdot x_{2,i} = 0 \cdot 0 = 0$.
- Luca (0) with Codeswitches (1): $x_{1,i} \cdot x_{2,i} = 0 \cdot 1 = 0$.
- Dragan (1) without Codeswitches (0): $x_{1,i} \cdot x_{2,i} = 1 \cdot 0 = 0$.
- Dragan (1) with Codeswitches (1): $x_{1,i} \cdot x_{2,i} = 1 \cdot 1 = 1$.

Let's compute the dummy variables and their product:

```
> d <- d |>
+   mutate(
+     Dragan = ifelse(Name == "Dragan", 1, 0),
+     WithCS = ifelse(CS == "with", 1, 0),
+     DraganWithCS = Dragan * WithCS
+   )
```

Now fit a linear model with all these dummy variables:

```
> potential.lm <- lm(Potential ~ Dragan + WithCS + DraganWithCS, data = d)
> potential.lm


Call:
lm(formula = Potential ~ Dragan + WithCS + DraganWithCS, data = d)

Coefficients:
 (Intercept)        Dragan        WithCS  DraganWithCS
       3.094         0.534         0.270        -0.933
```

We can use the estimated coefficients to reconstruct the cell means we computed earlier when drawing the graphs. In the following sums, rounding errors were corrected:

- Not Dragan (so Luca), without code-switches:

$$\widehat{y} = 3.09 + (0.53 \cdot 0) + (0.27 \cdot 0) + (-0.93 \cdot 0) = 3.09.$$

- Dragan, without code-switches:

$$\widehat{y} = 3.09 + (0.53 \cdot 1) + (0.27 \cdot 0) + (-0.93 \cdot 0) = 3.63.$$

- Not Dragan (so Luca), with code-switches:

$$\widehat{y} = 3.09 + (0.53 \cdot 0) + (0.27 \cdot 1) + (-0.93 \cdot 0) = 3.36.$$

- Dragan, with code-switches:

$$\widehat{y} = 3.09 + (0.53 \cdot 1) + (0.27 \cdot 1) + (-0.93 \cdot 1) = 2.96.$$

Note, incidentally, that we are using **treatment coding** here. As a result, the estimate of 0.53 for `Dragan` *only* pertains to the recording without code-switches (the level coded as 0). In order to obtain difference between the estimated conditional means between Luca and Dragan when the recording contais code-switches, you need to include the interaction terms: $0.53 - 0.93 = -0.40$. Similarly, the estimate of 0.27 for `WithCS` only pertains to ratings of 'Luca' (the level coded as 0). In order to obtain the difference between the estimated conditional means between recordings with vs. without code-switches for Dragan, you again need to include the interaction term: $0.27 - 0.93 = -0.66$.

In the first exercise for this lecture, you will learn to interpret the estimated parameters for a model that uses a different coding scheme. See Schad et al. (2020) and my blog post on recoding predictors for more details.

1.3 Uncertainty estimates

We can obtain estimated standard errors as well as confidence intervals just like before by using the `summary()` and `confint()` commands. Again, these computations are based on the assumption that the residuals are i.i.d. normal, but you can always use bootstrapping to check if a different set of assumptions leads to the same conclusion.

```
> summary(potential.lm)$coefficients

             Estimate Std. Error t value   Pr(>|t|)
(Intercept)   3.09375    0.14796 20.9090 1.9477e-46
Dragan        0.53370    0.18876  2.8274 5.3289e-03
WithCS        0.26989    0.19446  1.3879 1.6722e-01
DraganWithCS -0.93305    0.27672 -3.3719 9.4836e-04

> confint(potential.lm, level = 0.90)

                   5 %     95 %
(Intercept)   2.848871  3.33863
Dragan        0.221305  0.84610
WithCS       -0.051948  0.59172
DraganWithCS -1.391020 -0.47508
```

Because we used treatment coding here, the estimates for `Dragan` and `WithCS` aren't too relevant. The study's main result is that the presence of code-switches is some 0.93 ± 0.28 points more detrimental to ratings of 'Dragan's' academic potential than it is to ratings of 'Luca's' academic potential. Equivalently, the *absence* of code-switches is some 0.93 ± 0.28 more beneficial to ratings of 'Dragan's' academic potential than it is to ratings of 'Luca's' academic potential. (Interaction effects can often be framed in several equivalent ways.)

2 Interactions between a binary and a continuous predictor

Sometimes, researchers want to find out if the relationship between a continuous predictor and the outcome differs between groups. This type of research question, too, can be addressed in a general linear model. The idea is the same as in the previous section: Dummy-code the group variable, compute its pointwise product with the continuous predictor, and feed the dummy-coded group variable, the continuous predictor, and their pointwise product into the model.

For reasons of time, we won't run through an example of such an analysis. But you should be aware of a common analytical strategy that does *not* work. This doomed strategy is to fit several models in order to gauge the relationship between the continuous predictor and the outcome separately for each group, and to conclude that if this relationship is statistically significant in one group but not in the other, there must be an interaction between the groups and the continuous predictor. Gelman & Stern (2006) and Nieuwenhuis et al. (2011) explain why this is a terrible idea.

3 More complex interactions

It's possible to fit interactions between two continuous predictors; see the blog entry *Interactions between continuous variables*. It's also possible to fit interactions between three or more predictors. However, it can be difficult to make sense of three-way, four-way, etc. interactions, and I don't have any datasets that call for such an analysis.

4 A word of warning to the cognitive scientists

The mere fact that a statistical model suggests that two predictor variables interact in their effect on some outcome variable does *not* imply that these predictor variable interact in their effect on the *construct* that this outcome variable represents. This is particularly important to appreciate if the interaction in question is not a cross-over interaction, that is, if the relative order between two groups or conditions switches depending on the other predictor. For instance, if we observe a non-cross-over interaction between reading experience and word frequency on reading speed, this does not imply that reading experience and word frequency have non-additive effects on the cognitive construct that reading speed represents, viz., cognitive effort. See Wagenmakers et al. (2012) for further explanation.

5 Exercises

1. (Paper and pencil.) Consider the following fictitious experiment and analysis. Eighty participants are randomly assigned to the four cells of a two-by-two design, each cell corresponding to one of the combinations of two binary predictor variables (Variable 1: A vs. B, Variabe 2: X vs. Y). Then, their performance on some task is measured, yielding a continuous outcome variable.

   For the analysis, the analyst uses sum-coding: `Var1` reads $+0.5$ if the participant was assigned to a B-cell and -0.5 if they were assigned to an A-cell; `Var2` reads $+0.5$ if the participant was assigned to a Y-cell and -0.5 if they were assigned to an X-cell. The `Interaction` term is the pointwise product of `Var1` and `Var2`. The estimated parameter coefficients are as follows:

   ```
   (Intercept)        Var1        Var2 Interaction
       9.55076    -0.39801    -3.88412    -6.50224
   ```

   (a) Compute the mean outcome value for each of the four cells.

   (b) Explain what the estimated `(Intercept)` coefficient represents.

   (c) Explain what the estimated `Var1` and `Var2` coefficient represent.

   (d) Explain what the estimated `Interaction` coefficient represents.

2. (With R.) The main finding in Berthele (2012) was the interaction effect of -0.93 ± 0.28 points (90% CI: $[-1.40, -0.47]$). Double-check this estimated standard error and the 90% confidence interval using a semi-parametric bootstrap that does not assume that the residuals are homoskedastic.

   Tip: You can group the data by two variables by using `group_by(variable1, variable2)`.

References

Berthele, Raphael. 2012. The influence of code-mixing and speaker information on perception and assessment of foreign language proficiency: An experimental study. *International Journal of Bilingualism* 16(4). 453–466. doi:10.1177/1367006911429514.

Gelman, Andrew & Hal Stern. 2006. The difference between "significant" and "not significant" is not itself statistically significant. *The American Statistician* 60(4). 328–331. doi:10.1198/000313006X152649.

Nieuwenhuis, Sander, Birte U. Forstmann & Eric-Jan Wagenmakers. 2011. Erroneous analyses of interactions in neuroscience: A problem of significance. *Nature Neuroscience* 14. 1105–1107. doi:10.1038/nn.2886.

Schad, Daniel J., Shravan Vasishth, Sven Hohenstein & Reinhold Kliegl. 2020. How to capitalize on a priori contrasts in linear (mixed) models: A tutorial. *Journal of Memory and Language* 110. doi:10.1016/j.jml.2019.104038.

Wagenmakers, Eric-Jan, Angelos-Miltiadis Krypotos, Amy H. Criss & Geoff Iverson. 2012. On the interpretation of removable interactions: A survey of the field 33 years after Loftus. *Memory & Cognition* 40(2). 145–160. doi:10.3758/s13421-011-0158-0.