Model design in R: Class 2

Alexandre Cremers

August 13th, 2019

Today's class

- Introduce ordinary linear regression
  (formal definition and `R` implementation)
- Formalize yesterday's discussion
- Introduce mixed-effects models
- See why yesterday's problem has consequences beyond mere readability of our stats output.

A bit more technical: stop me whenever something isn't clear!

The ordinary linear model

Dependent variable: $\boldsymbol{y} = (y_1, \ldots y_n)$ (what we measure)
Independent variable(s): $X = (\boldsymbol{x_1}, \ldots, \boldsymbol{x_n})$ (predictors)

Linear model = linear relation between X and Y:

$$\begin{aligned} y_i &= \beta_1 x_{i1} + \cdots + \beta_k x_{ik} + \varepsilon_i \\ &= \boldsymbol{\beta} \cdot \boldsymbol{x_i} + \varepsilon_i \\ \boldsymbol{y} &= X\boldsymbol{\beta} + \boldsymbol{\varepsilon} \end{aligned}$$

Model fitting = finding the best possible $\boldsymbol{\beta}$
(the exact criteria do not matter for our discussion)

What is X?

- In principle, any set of values which may predict the value of the dependent variable for each data point.
- In a SemPrag experiment: usually the predictors are a small set of categorical variables (labels) for each data point.
- Considering all possible interactions between predictors, we need up to $K = \prod k_j$ parameters, where k_j is the number of levels for predictor j.
- Default `R` behavior:
  - 1 intercept,
  - $k_j - 1$ main effects for each predictor j,
  - the rest in interaction terms
- Let's see how this unfolds...

Unfolding R's default parameters

Cond: **target**, true, false

Order: **a-every**, every-a

```
Resp ~ Cond*Order1+Cond+Order+Cond:Order
```

Subj	Cond	Order	Resp	(inter)	Condtrue	Condfalse	Orderevery-a	Condtrue:Orderevery-a	Condtrue:Orderevery-a
1	target	every-a	1	1	0	0	1	0	0
1	true	a-every	1	1	1	0	0	0	0
1	target	a-every	0	1	0	0	0	0	0
1	false	every-a	0	1	0	1	1	0	1
2	true	every-a	1	1	1	0	1	1	0
⋮	⋮	⋮	⋮			X			

Our better parametrization

```
Resp ~ 0 + IS + True + False + (IS + True + False):Order
```

Subj	Cond	Order	Resp	IS	True	False	IS:Order	True:Order	False:Order
1	target	every-a	1	-1	1	0	-0.5	0.5	0
1	true	a-every	1	0	1	0	0	-0.5	0
1	target	a-every	0	1	0	1	-0.5	0	-0.5
1	false	every-a	0	0	0	1	0	0	0.5
2	true	every-a	1	0	1	0	0	0.5	0

X'

Relation between the two models

$$X' = XA \qquad \text{where } A = \begin{bmatrix} 1 & 0 & 1 & -0.5 & 0 & -0.5 \\ -1 & 1 & -1 & 0.5 & -0.5 & 0.5 \\ -1 & 0 & 0 & 0.5 & 0 & 0 \\ -2 & 1 & -1 & 0 & 0.5 & 0.5 \\ 2 & -1 & 1 & 0 & 0.5 & -0.5 \\ 2 & -1 & 1 & 0 & -0.5 & 0.5 \end{bmatrix}$$

So if our original model was:

$$\boldsymbol{y} = X\boldsymbol{\beta} + \boldsymbol{\varepsilon} = X'A^{-1}\boldsymbol{\beta} + \boldsymbol{\varepsilon}$$

We can deduce our new model parameters:

$$\boldsymbol{\beta'} = A^{-1}\boldsymbol{\beta}$$

More generally: We can use any re-parametrization $X'' = XB$ as long as B is *invertible*.

One step back: experiment design

This should guide our experiment design:

- Imagine we want to study a sentence with k potential readings $\varphi_1, \ldots \varphi_k$.
- For simplicity, let's imagine that these potential readings are ordered by entailment:

$$\varphi_k \rightarrow \varphi_{k-1} \rightarrow \cdots \rightarrow \varphi_1$$

- We can design $k+1$ conditions:
  - C_j validates readings φ_1 to φ_j and falsifies φ_{j+1} to φ_k.
  - ☞ C_0 and C_k are false and true controls, respectively.
  - With k parameters, the $k+1$ conditions can't be independent. We can add two parameters for error rates on controls.
- Many parametrizations are now possible...

Intermezzo

Concrete example with 3 interpretations ordered by entailment:

(1) Mary knows who played at the soccer match

a. For each player, Mary knows that they played

b. For each player, Mary knows that they played and she doesn't falsely believe that anyone else played

c. For each player, Mary knows that they played and she knows that no one else played

☞ 4 possible experimental conditions (Cremers&Chemla 2016)

Possible parametrizations for ordered readings

R default: treatment coding, usually with C_0 (or C_k) as baseline:

$$\begin{array}{c} C_0 \\ C_1 \\ \vdots \\ \vdots \\ C_k \end{array} \begin{bmatrix} 1 & 0 & 0 & \dots & 0 \\ 1 & 1 & 0 & \dots & 0 \\ \vdots & 0 & \ddots & (0) & \vdots \\ \vdots & \vdots & (0) & \ddots & \vdots \\ 1 & 0 & 0 & \dots & 1 \end{bmatrix}$$

Possible parametrizations for ordered readings

Alternative 1: encoding readings φ_1 to φ_k plus false baseline.

$$\begin{array}{c} C_0 \\ C_1 \\ \vdots \\ \vdots \\ C_k \end{array} \begin{bmatrix} 1 & 0 & \dots & \dots & 0 \\ 1 & 1 & 0 & \dots & 0 \\ \vdots & \vdots & & \ddots & (0) & \vdots \\ \vdots & \vdots & (1) & & \ddots & \vdots \\ 1 & \dots & \dots & \dots & 1 \end{bmatrix}$$

Possible issue: the error rate on C_k is unlikely to vary much from sentence to sentence (more on that later).

Possible parametrizations for ordered readings

Alternative 2: encode true and false baselines, plus readings $\varphi_1 \dots \varphi_{k-1}$

$$
\begin{array}{c}
C_0 \\ C_1 \\ \vdots \\ \vdots \\ C_{k-1} \\ C_k
\end{array}
\begin{bmatrix}
1 & 0 & \dots & \dots & 0 & 0 \\
1 & 1 & 0 & \dots & 0 & 0 \\
\vdots & \vdots & & \ddots & \vdots & 0 \\
\vdots & \vdots & (1) & & \ddots & \vdots & 0 \\
1 & \dots & \dots & \dots & 1 & 0 \\
0/1 & \dots 0 & \dots & \dots & 0 & 1
\end{bmatrix}
$$

NB: choosing 0 at the bottom left means that the last predictor represents the absolute response rate on true controls. Choosing 1 instead let us represent the differences between true and false controls instead.

Interim conclusion

- At this point—every thing else being equal—these different models are equivalent (they have the exact same residuals).
- The only difference is that the parameters are easier to interpret than R's default parameters.
- However, the transformation may lead to different results in more sophisticated models...

Models used in SemPrag diverge from ordinary linear models on **two** dimensions.

Generalized linear models

First difference: Models used for SemPrag are rarely linear.

- Binary responses (True/False) cannot vary linearly or even continuously.
- Logistic regression: predict the *probability* of a true response.
- Same thing for other SemPrag measures:
  - Likert scales: ordinal models,
  - Continuous scales: should be treated with beta regression.

Ordinary linear model: $\longrightarrow$ Generalized linear model:

$$y \sim \mathcal{N}(X\beta) \qquad E[\boldsymbol{y}] = g^{-1}(X\boldsymbol{\beta})$$

$$(\boldsymbol{y} = X\boldsymbol{\beta} + \boldsymbol{\varepsilon})$$

The difference between ordinary and generalized linear models is mostly orthogonal to our discussion.

However: Ordinal models, as implemented in `R` package `ordinal`, always come with an intercept (will play a role tomorrow).

Taking participants into account

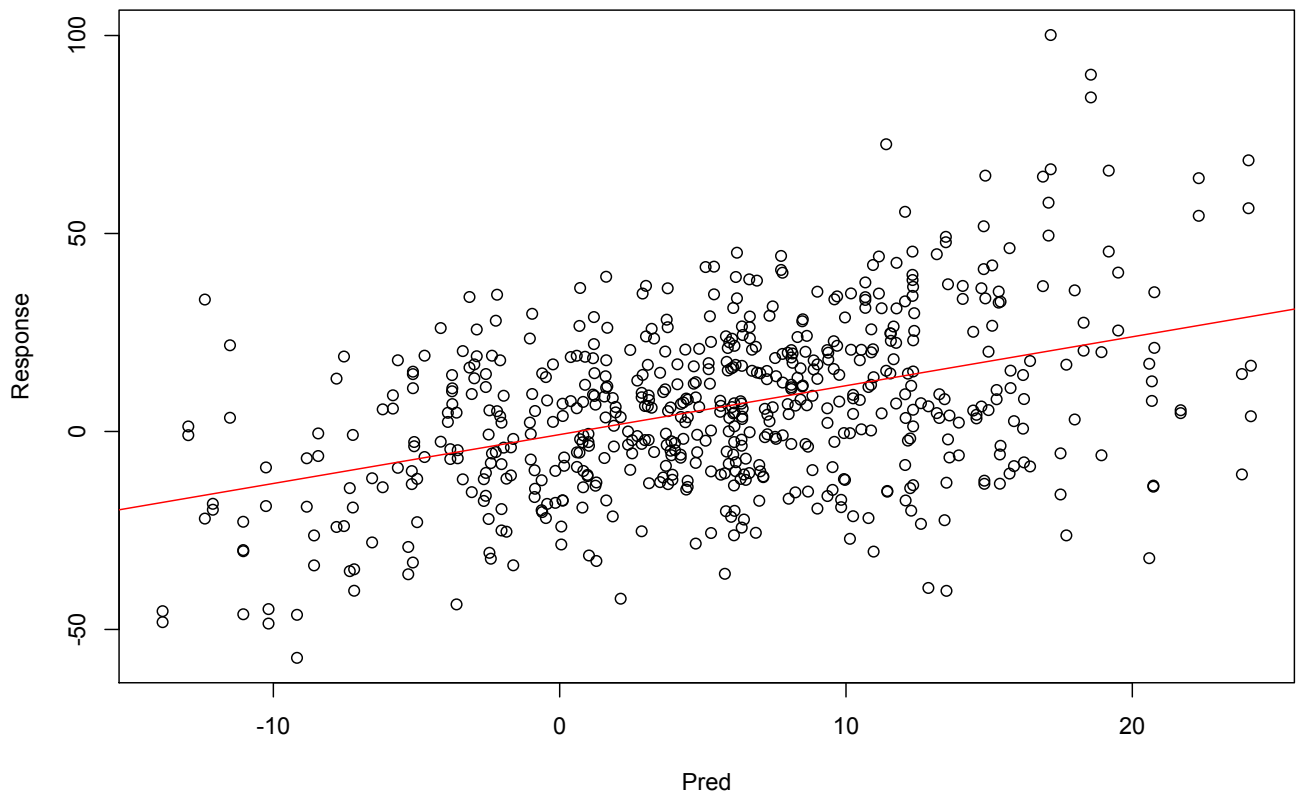
Second difference: Independence of data points.

- In a typical experiment, many participants each give us many data points (repeated measures design).
- Data points coming from a single participants are not independent.
- ~~Solution 1: average across a participant's responses to get a single data point.~~
  Loss of statistical power and ill-behaved dependent variable (e.g., rates are neither binary nor normally distributed).
- Solution 2: mixed-effects models.

Mixed-effects linear models

Back to our basic linear model:

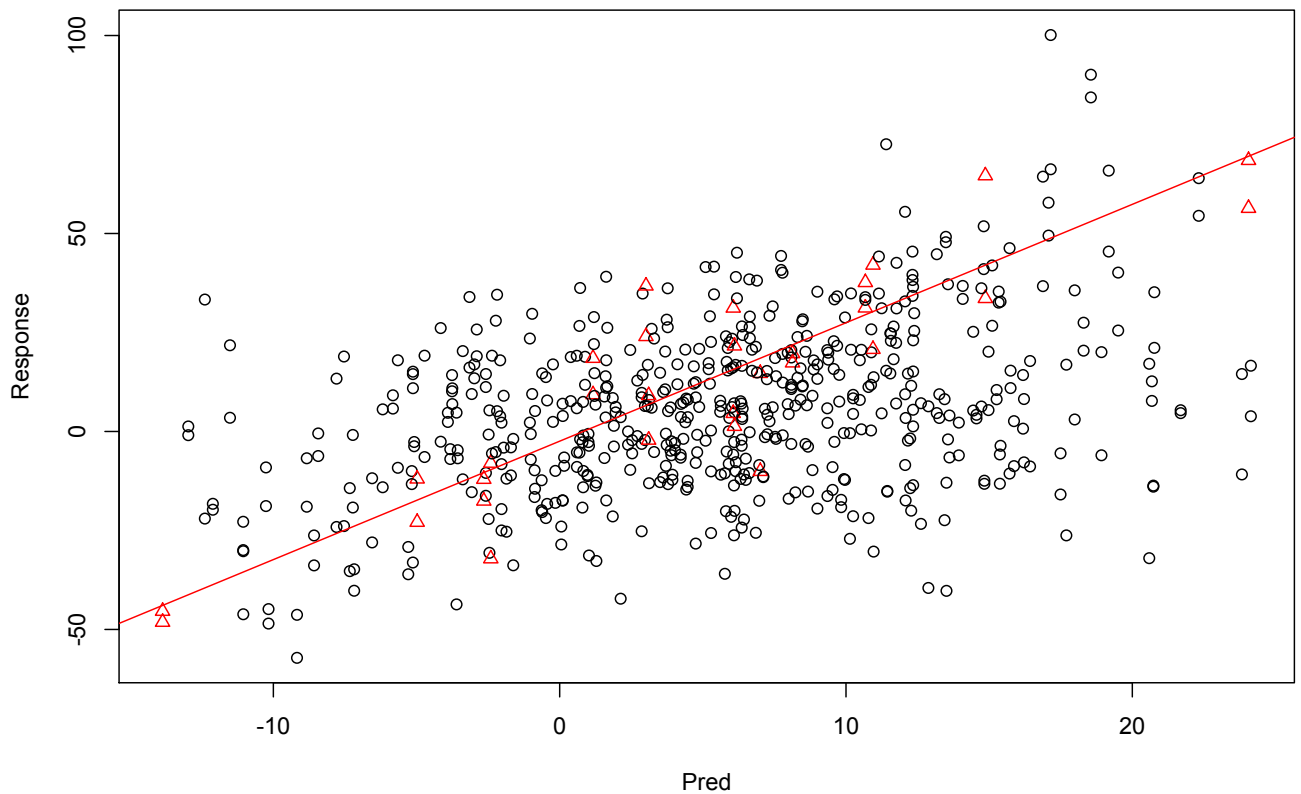
$$\boldsymbol{y} = X\boldsymbol{\beta} + \boldsymbol{\varepsilon}$$



Mixed-effects linear models

We could run a different model for each participants:

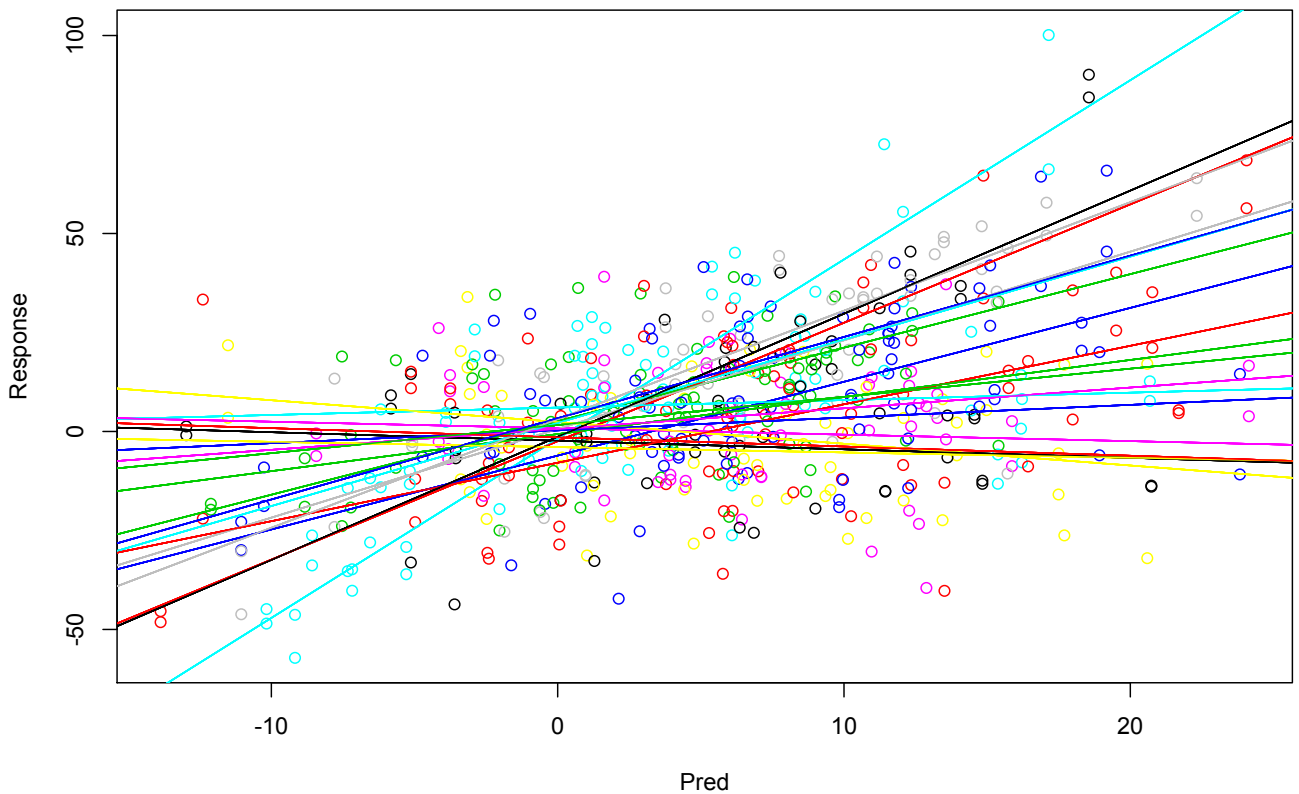
$$\boldsymbol{y} = \beta_0^s \mathbf{1} + \beta_1^s \boldsymbol{x} + \boldsymbol{\varepsilon}$$



Mixed-effects linear models

We could run a different model for each participants:

$$\boldsymbol{y} = \beta_0^s \mathbf{1} + \beta_1^s \boldsymbol{x} + \boldsymbol{\varepsilon}$$



Mixed-effects linear models

$$y_{i,s} = (\beta_0 + u_s) + (\beta_1 + v_s)x_i + \varepsilon_i$$

β_0, β_1: fixed effects
$\boldsymbol{u}, \boldsymbol{v}$: random effects
☞ $\boldsymbol{u}, \boldsymbol{v}$ are not the actual parameters of the model, but represent random samples from a distribution. What we evaluate are the parameters of that distribution.

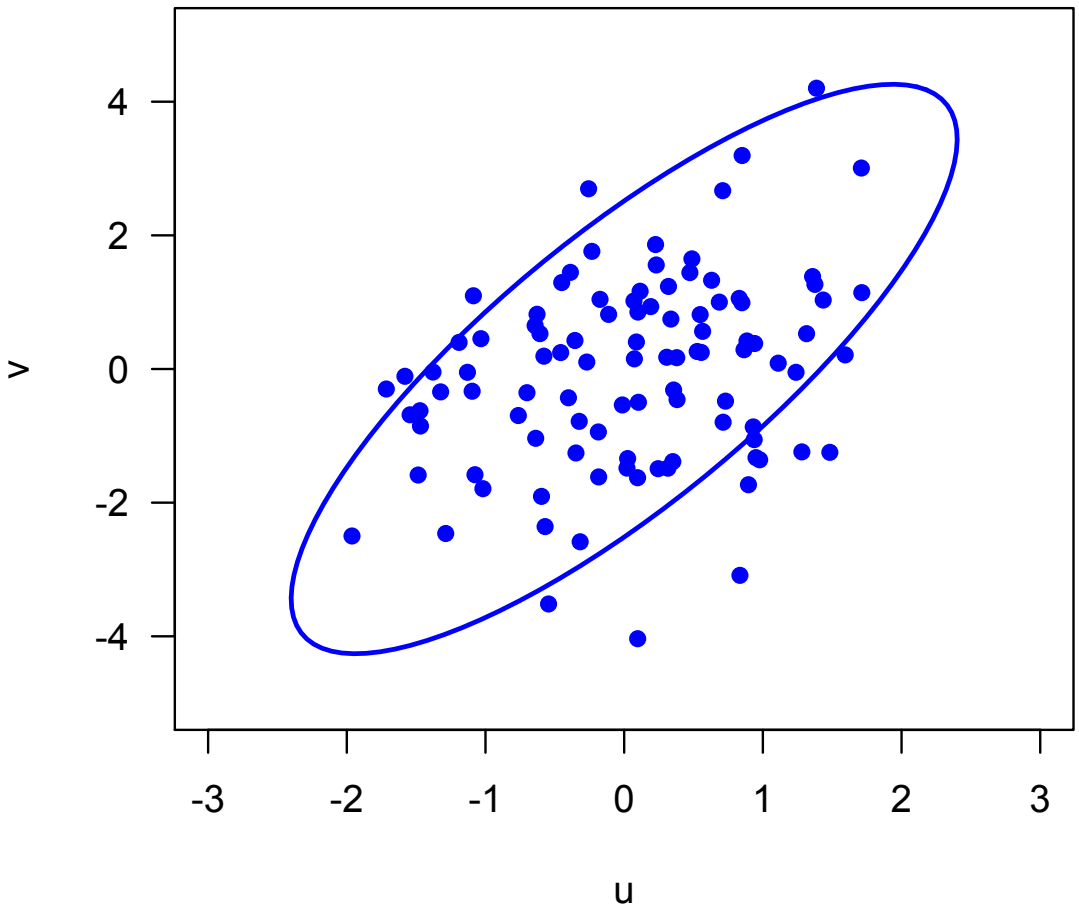
In `R`: mixed-effects models usually fitted with package `lme4`:

```
lmer(Response ~ 1 + Pred + (1+Pred|Subject))
```

This assumes $\begin{bmatrix} \boldsymbol{u} \\ \boldsymbol{v} \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} \sigma_1^2 & \rho\sigma_1\sigma_2 \\ \rho\sigma_1\sigma_2 & \sigma_2^2 \end{bmatrix} \right)$

Random effects distribution

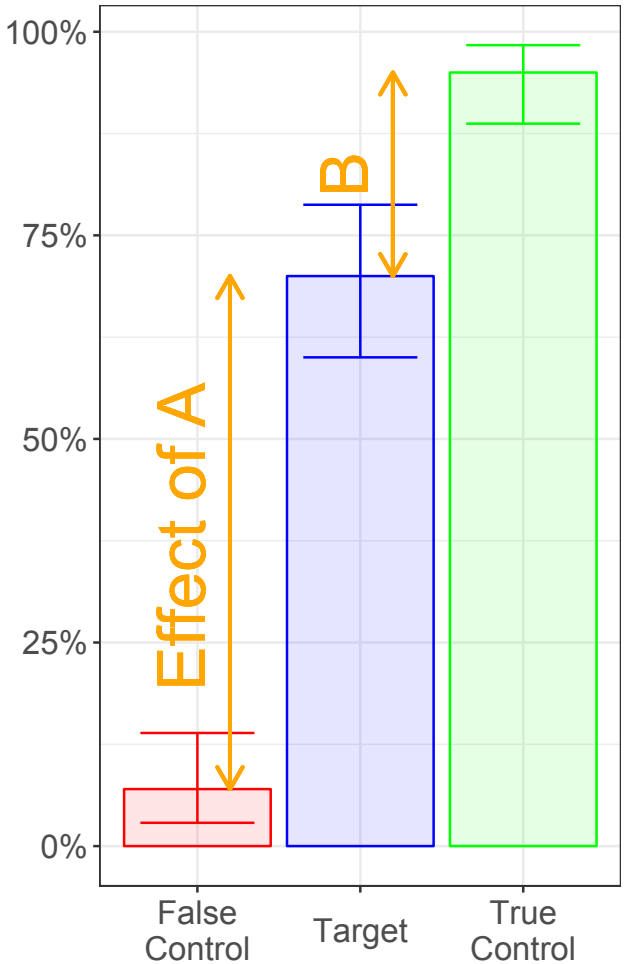
$$\begin{bmatrix} \boldsymbol{u} \\ \boldsymbol{v} \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} \sigma_1^2 & \rho\sigma_1\sigma_2 \\ \rho\sigma_1\sigma_2 & \sigma_2^2 \end{bmatrix} \right)$$



Some technical difficulties

- With n fixed effects, we get $\frac{n(n+1)}{2}$ parameters for the random effects distribution.
- Most psycholinguistics experiments do not have enough data to fit so many parameters and including them can lead to uninterpretable models (Bates et al. 2015).
- A solution is to get rid of unnecessary correlations, which is a bit complicated but can be done in a principled way.
- Problem: for truth-value judgments tasks analyzed in the traditional way, correlations are very often important!

The correlation issue in SemPrag

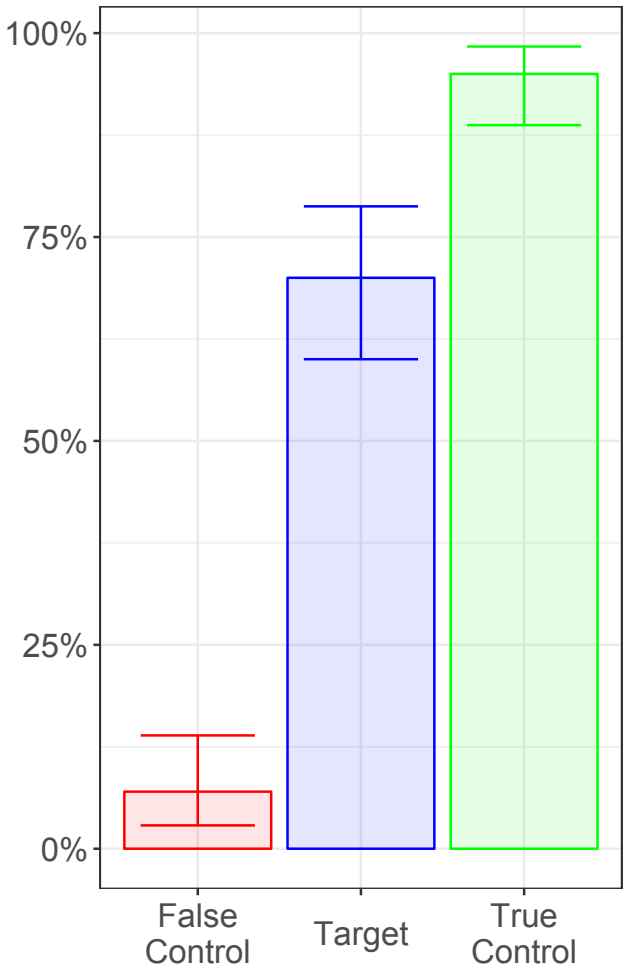


Analyzed with treatment coding with 'Target' as baseline.

```
Answer~1+A+B + (1+A+B|Subject)
```

Variation on the ambiguous target more important than on controls, and probably independent.
☞ strong correlations between all random effects, we're stuck with 6 parameters.

The correlation issue in SemPrag



Alternative parametrization:

	False	Weak	True
FalseCtrl	1	0	0
Target	1	1	0
TrueCtrl	0	0	1

```
Answer~0+False+Weak+True
```

☞ RE for True/False likely correlate
☞ RE for Weak likely independent

Defining a custom random effect structure

Fixed effects: `0+False+Weak+True`
Goal:

- Keep correlation between the RE for False and True
- Treat Weak as independent

Formally:

$$\begin{bmatrix} \boldsymbol{u}_t \\ \boldsymbol{u}_f \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \sigma_t^2 & \rho\sigma_t\sigma_f \\ \rho\sigma_t\sigma_f & \sigma_f^2 \end{bmatrix}\right)$$

$$\boldsymbol{u}_w \sim \mathcal{N}\left(0, \sigma_w\right)$$

Implementation in `R/lme4`:

```
Answer~  0+False+Weak+True +
        (0+False+True|Subject) +
        (0+Weak|Subject)
```

Conclusion

- Mixed-effects models are the standard for psycholinguistics.
- However, mixed-effects models run into problems with complex RE structures, *especially* logit mixed-effects model on SemPrag data with controls.
- Main issue: with a default parametrization, can't avoid complex correlations between the different RE.
- By using parameters of theoretical significance, we reduce these arbitrary correlations and instead have principled way for simplifying the RE structure.
- Alternatively, see Bates et al (2015) for an empirically-driven approach to RE structure simplification.