Mixed-effects (a.k.a. multi-level, or hierarchical) models

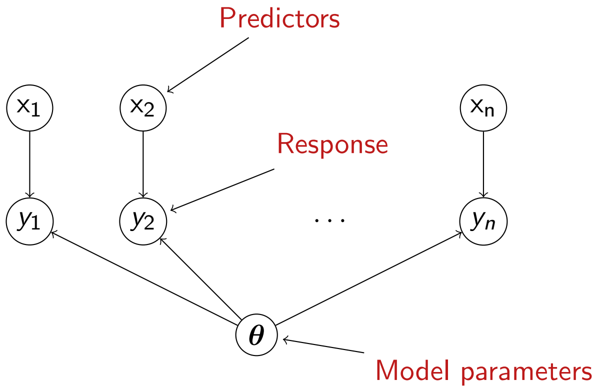
Roger Levy

Massachusetts Institute of Technology

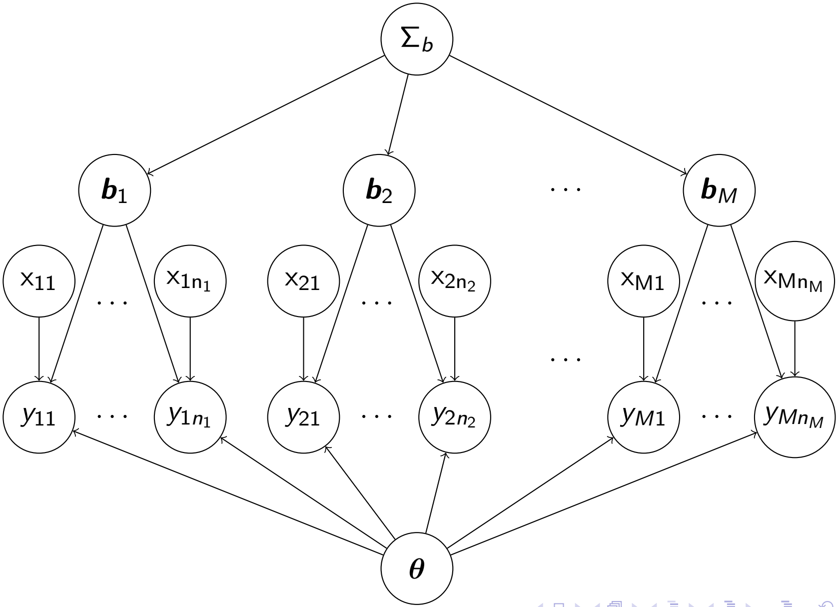
March 12, 2025

Mixed-effects/hierarchical GLMs

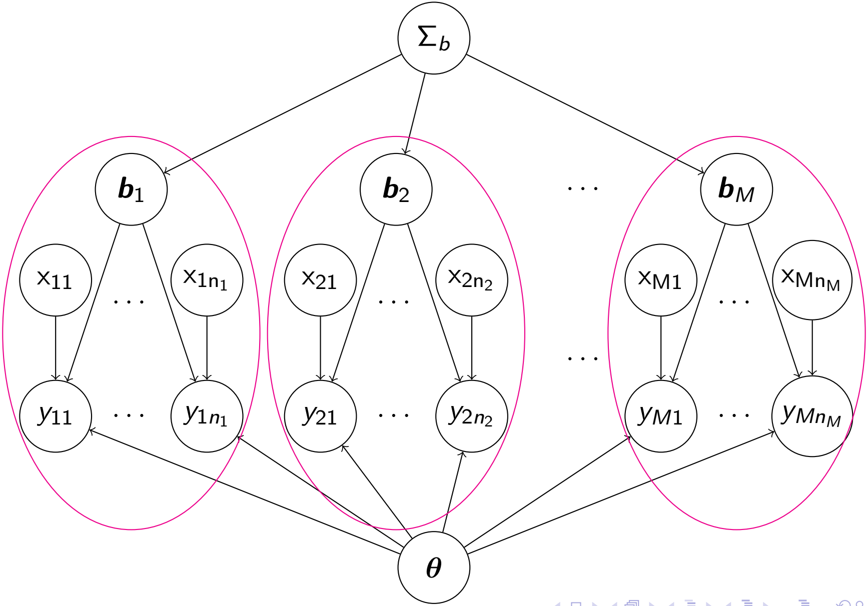
The non-hierarchical GLM picture:



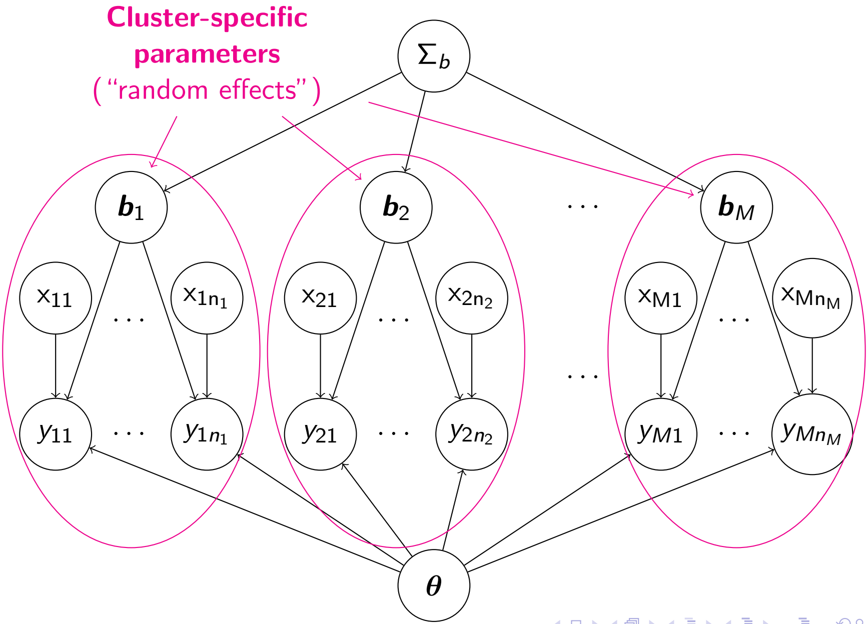
Mixed-effects/hierarchical GLMs



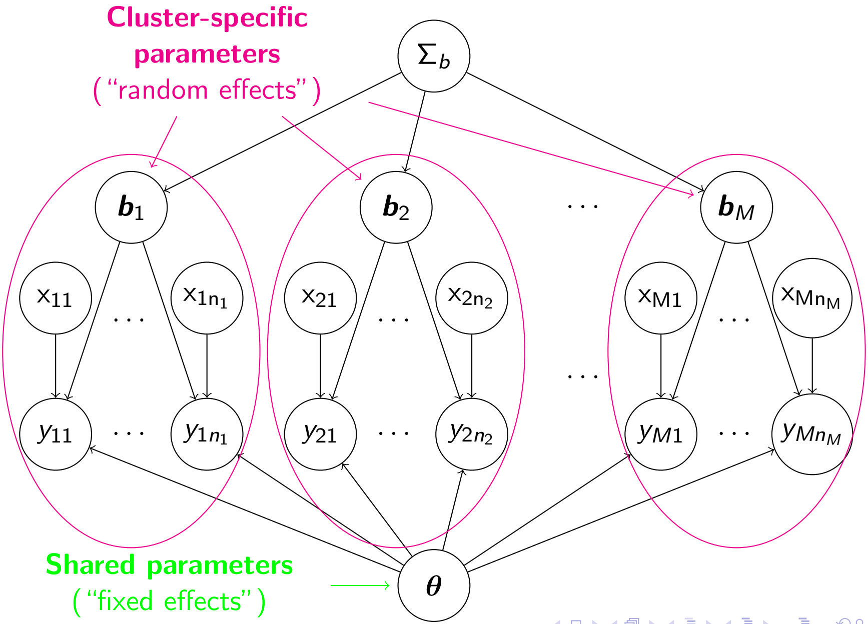
Mixed-effects/hierarchical GLMs



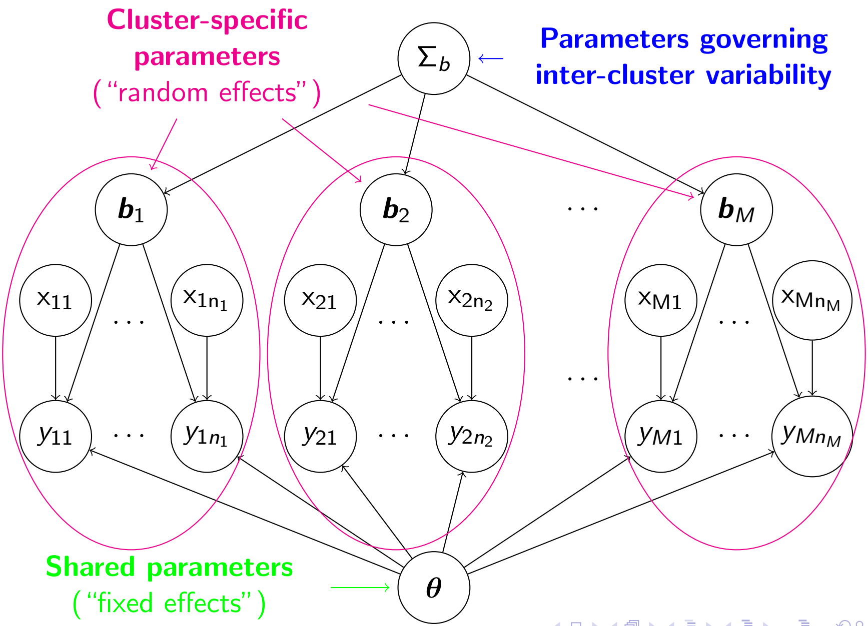
Mixed-effects/hierarchical GLMs



Mixed-effects/hierarchical GLMs



Mixed-effects/hierarchical GLMs



Multi-level Models I

An example of a multi-level model:

- Back to your lexical-decision experiment
  `tpozt` *Word or non-word?*
  `houze` *Word or non-word?*
- Non-words with different *neighborhood densities* should have different average decision time

Multi-level Models I

An example of a multi-level model:

- Back to your lexical-decision experiment
  `tpozt` *Word or non-word?*
  `houze` *Word or non-word?*
- Non-words with different *neighborhood densities* should have different average decision time
- Additionally, different participants in your study may also have:
  - different overall decision speeds
  - differing sensitivity to neighborhood density

Multi-level Models I

An example of a multi-level model:

- Back to your lexical-decision experiment
  `tpozt` *Word or non-word?*
  `houze` *Word or non-word?*
- Non-words with different *neighborhood densities* should have different average decision time
- Additionally, different participants in your study may also have:
  - different overall decision speeds
  - differing sensitivity to neighborhood density
- You want to draw inferences about all these things at the same time

Multi-level Models I: Model construction

- Once again we'll assume for simplicity that the number of word neighbors x has a linear effect on mean reading time, and that trial-level noise is normally distributed*

Multi-level Models I: Model construction

- Once again we'll assume for simplicity that the number of word neighbors x has a linear effect on mean reading time, and that trial-level noise is normally distributed*
- Random effects, starting simple: let each participant i have idiosyncratic differences in reading speed

$$RT_{ij} = \alpha + \beta x_{ij} + \overbrace{b_i}^{\sim N(0,\sigma_b)} + \overbrace{\epsilon_{ij}}^{\text{Noise}\sim N(0,\sigma_\epsilon)}$$

Multi-level Models I: Model construction

- Once again we'll assume for simplicity that the number of word neighbors x has a linear effect on mean reading time, and that trial-level noise is normally distributed*
- Random effects, starting simple: let each participant i have idiosyncratic differences in reading speed

$$RT_{ij} = \alpha + \beta x_{ij} + \overbrace{b_i}^{\sim N(0,\sigma_b)} + \overbrace{\epsilon_{ij}}^{\text{Noise}\sim N(0,\sigma_\epsilon)}$$

- In R, we'd write this relationship as

```
RT ~ 1 + x + (1 | participant)
```

Multi-level Models I: Model construction

- Once again we'll assume for simplicity that the number of word neighbors x has a linear effect on mean reading time, and that trial-level noise is normally distributed*
- Random effects, starting simple: let each participant i have idiosyncratic differences in reading speed

$$RT_{ij} = \alpha + \beta x_{ij} + \overbrace{b_i}^{\sim N(0,\sigma_b)} + \overbrace{\epsilon_{ij}}^{\text{Noise}\sim N(0,\sigma_\epsilon)}$$

- In R, we'd write this relationship as

```
RT ~ 1 + x + (1 | participant)
```

- Once again we can leave off the 1, and the noise term ϵ_{ij} is implicit

Multi-level Models I: Model construction

- Once again we'll assume for simplicity that the number of word neighbors x has a linear effect on mean reading time, and that trial-level noise is normally distributed*
- Random effects, starting simple: let each participant i have idiosyncratic differences in reading speed

$$RT_{ij} = \alpha + \beta x_{ij} + \overbrace{b_i}^{\sim N(0,\sigma_b)} + \overbrace{\epsilon_{ij}}^{\text{Noise}\sim N(0,\sigma_\epsilon)}$$

- In R, we'd write this relationship as

```
RT ~        x + (1 | participant)
```

- Once again we can leave off the 1, and the noise term ϵ_{ij} is implicit

Multi-level Models II: simulating data

$$\boxed{RT_{ij} = \alpha + \beta x_{ij} + \overbrace{b_i}^{\sim N(0,\sigma_b)} + \overbrace{\epsilon_{ij}}^{\text{Noise}\sim N(0,\sigma_\epsilon)}}$$

- One beauty of multi-level models is that you can simulate trial-level data
- This is invaluable for achieving deeper understanding of both your analysis and your data

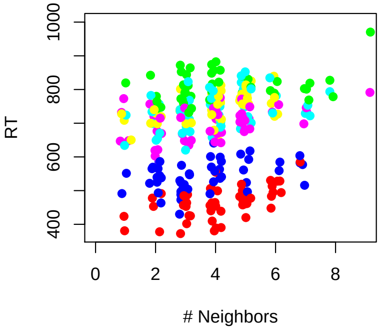
Multi-level Models II: simulating data

$$RT_{ij} = \alpha + \beta x_{ij} + \overbrace{b_i}^{\sim N(0,\sigma_b)} + \overbrace{\epsilon_{ij}}^{\text{Noise}\sim N(0,\sigma_\epsilon)}$$

- One beauty of multi-level models is that you can simulate trial-level data
- This is invaluable for achieving deeper understanding of both your analysis and your data

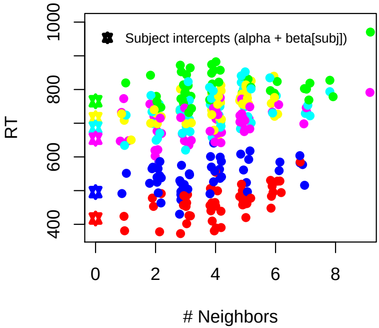
```
## simulate some data
> sigma.b <- 125       # inter-subject variation larger than
> sigma.e <- 40        # intra-subject, inter-trial variation
> alpha <- 500
> beta <- 12
> M <- 6                             # number of participants
> n <- 50                            # trials per participant
> b <- rnorm(M, 0, sigma.b)          # individual differences
> nneighbors <- rpois(M*n,3) + 1     # generate num. neighbors
> subj <- rep(1:M,n)
> RT <- alpha + beta * nneighbors + # simulate RTs!
   b[subj] + rnorm(M*n,0,sigma.e)   #
```

Multi-level Models III: simulating data



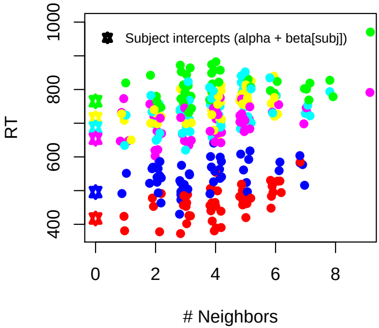
- Participant-level clustering is easily visible

Multi-level Models III: simulating data



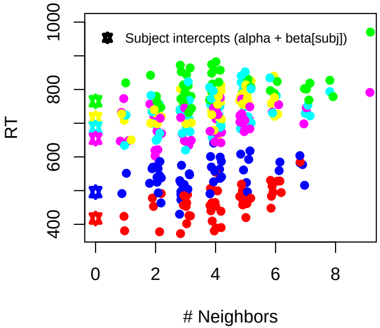
- Participant-level clustering is easily visible

Multi-level Models III: simulating data



- Participant-level clustering is easily visible
- This reflects the fact that inter-participant variation (125ms) is larger than inter-trial variation (40ms)

Multi-level Models III: simulating data



- Participant-level clustering is easily visible
- This reflects the fact that inter-participant variation (125ms) is larger than inter-trial variation (40ms)
- And the effects of neighborhood density are also visible

Statistical inference with multi-level models

$$\boxed{RT_{ij} = \alpha + \beta x_{ij} + \overbrace{b_i}^{\sim N(0,\sigma_b)} + \overbrace{\epsilon_{ij}}^{\text{Noise}\sim N(0,\sigma_\epsilon)}}$$

- Thus far, we've just defined a model and used it to generate data

Statistical inference with multi-level models

$$RT_{ij} = \alpha + \beta x_{ij} + \overbrace{b_i}^{\sim N(0,\sigma_b)} + \overbrace{\epsilon_{ij}}^{\text{Noise}\sim N(0,\sigma_\epsilon)}$$

- Thus far, we've just defined a model and used it to generate data
- We psycholinguists are usually in the opposite situation...

Statistical inference with multi-level models

$$RT_{ij} = \alpha + \beta x_{ij} + \overbrace{b_i}^{\sim N(0,\sigma_b)} + \overbrace{\epsilon_{ij}}^{\text{Noise} \sim N(0,\sigma_\epsilon)}$$

- Thus far, we've just defined a model and used it to generate data
- We psycholinguists are usually in the opposite situation...
- We *have* data and we need to infer a model
  - Specifically, the "fixed-effect" parameters α, β, and σ_ϵ, plus the parameter governing inter-subject variation, σ_b
  - e.g., hypothesis tests about effects of neighborhood density: can we reliably infer that β is $\{$non-zero, positive, $\ldots\}$?

Statistical inference with multi-level models

$$RT_{ij} = \alpha + \beta x_{ij} + \overbrace{b_i}^{\sim N(0,\sigma_b)} + \overbrace{\epsilon_{ij}}^{\text{Noise}\sim N(0,\sigma_\epsilon)}$$

- Thus far, we've just defined a model and used it to generate data
- We psycholinguists are usually in the opposite situation...
- We *have* data and we need to infer a model
  - Specifically, the "fixed-effect" parameters α, β, and σ_ϵ, plus the parameter governing inter-subject variation, σ_b
  - e.g., hypothesis tests about effects of neighborhood density: can we reliably infer that β is {non-zero, positive, ...}?
- Fortunately, we can use the same principles as before to do this:
  - The principle of maximum likelihood
  - Or Bayesian inference

Fitting a multi-level model using maximum likelihood

$$RT_{ij} = \alpha + \beta x_{ij} + \overbrace{b_i}^{\sim N(0,\sigma_b)} + \overbrace{\epsilon_{ij}}^{\text{Noise}\sim N(0,\sigma_\epsilon)}$$

```
> m <- lmer(time ~ neighbors.centered +
    (1 | participant),dat,REML=F)
> print(m,corr=F)
[...]
Random effects:
 Groups      Name        Variance Std.Dev.
 participant (Intercept)  4924.9   70.177
 Residual                19240.5  138.710
Number of obs: 1760, groups: participant, 44

Fixed effects:
                   Estimate Std. Error t value
(Intercept)         583.787     11.082   52.68
neighbors.centered    8.986      1.278    7.03
```

Fitting a multi-level model using maximum likelihood

$$RT_{ij} = \alpha + \beta x_{ij} + \overbrace{b_i}^{\sim N(0,\sigma_b)} + \overbrace{\epsilon_{ij}}^{\text{Noise}\sim N(0,\sigma_\epsilon)}$$

```
> m <- lmer(time ~ neighbors.centered +
    (1 | participant),dat,REML=F)
> print(m,corr=F)
[...]
Random effects:
 Groups      Name        Variance Std.Dev.
 participant (Intercept)  4924.9   70.177
 Residual                19240.5  138.710
Number of obs: 1760, groups: participant, 44

Fixed effects:
              α̂        Estimate Std. Error t value
(Intercept)            583.787     11.082   52.68
neighbors.centered       8.986      1.278    7.03
```

Fitting a multi-level model using maximum likelihood

$$RT_{ij} = \alpha + \beta x_{ij} + \overbrace{b_i}^{\sim N(0,\sigma_b)} + \overbrace{\epsilon_{ij}}^{\text{Noise}\sim N(0,\sigma_\epsilon)}$$

```
> m <- lmer(time ~ neighbors.centered +
    (1 | participant),dat,REML=F)
> print(m,corr=F)
[...]
Random effects:
 Groups      Name        Variance Std.Dev.
 participant (Intercept)  4924.9   70.177
 Residual                19240.5  138.710
Number of obs: 1760, groups: participant, 44

Fixed effects:
                   Estimate Std. Error t value
(Intercept)         583.787     11.082   52.68
neighbors.centered    8.986      1.278    7.03
```

$\widehat{\alpha}$

$\widehat{\beta}$

Fitting a multi-level model using maximum likelihood

$$RT_{ij} = \alpha + \beta x_{ij} + \overbrace{b_i}^{\sim N(0,\sigma_b)} + \overbrace{\epsilon_{ij}}^{\text{Noise}\sim N(0,\sigma_\epsilon)}$$

```
> m <- lmer(time ~ neighbors.centered +
   (1 | participant),dat,REML=F)
> print(m,corr=F)
[...]
Random effects:
 Groups      Name        Variance Std.Dev.
 participant (Intercept)  4924.9   70.177
 Residual                19240.5  138.710
Number of obs: 1760, groups: participant, 44

Fixed effects:
                   Estimate Std. Error t value
(Intercept)         583.787     11.082   52.68
neighbors.centered    8.986      1.278    7.03
```

$\widehat{\sigma}_b$ $\widehat{\alpha}$ $\widehat{\beta}$

Fitting a multi-level model using maximum likelihood

$$RT_{ij} = \alpha + \beta x_{ij} + \overbrace{b_i}^{\sim N(0,\sigma_b)} + \overbrace{\epsilon_{ij}}^{\text{Noise}\sim N(0,\sigma_\epsilon)}$$

```
> m <- lmer(time ~ neighbors.centered +
   (1 | participant),dat,REML=F)
> print(m,corr=F)
[...]
Random effects:
 Groups      Name        Variance Std.Dev.
 participant (Intercept)  4924.9   70.177
 Residual                19240.5  138.710
Number of obs: 1760, groups: participant, 44

Fixed effects:
                   Estimate Std. Error t value
(Intercept)         583.787     11.082   52.68
neighbors.centered    8.986      1.278    7.03
```

$\widehat{\sigma}_b$

$\widehat{\sigma}_\epsilon$

$\widehat{\alpha}$

$\widehat{\beta}$

Interpreting parameter estimates

`Intercept`	583.79
`neighbors.centered`	8.99
$\widehat{\sigma}_b$	70.18
$\widehat{\sigma}_\epsilon$	138.7

Interpreting parameter estimates

`Intercept`	583.79
`neighbors.centered`	8.99
$\widehat{\sigma}_b$	70.18
$\widehat{\sigma}_\epsilon$	138.7

- The *fixed effects* are interpreted just as in a traditional single-level model:

Interpreting parameter estimates

`Intercept`	583.79
`neighbors.centered`	8.99
$\widehat{\sigma}_b$	70.18
$\widehat{\sigma}_\epsilon$	138.7

- The *fixed effects* are interpreted just as in a traditional single-level model:
  - The "average" RT for a non-word in this study is 583.79ms

Interpreting parameter estimates

`Intercept`	583.79
`neighbors.centered`	8.99
$\widehat{\sigma}_b$	70.18
$\widehat{\sigma}_\epsilon$	138.7

- The *fixed effects* are interpreted just as in a traditional single-level model:
  - The "average" RT for a non-word in this study is 583.79ms
  - Every extra neighbor increases "average" RT by 8.99ms

Interpreting parameter estimates

Intercept	583.79
neighbors.centered	8.99
$\widehat{\sigma}_b$	70.18
$\widehat{\sigma}_\epsilon$	138.7

- The *fixed effects* are interpreted just as in a traditional single-level model:
  - The "average" RT for a non-word in this study is 583.79ms
  - Every extra neighbor increases "average" RT by 8.99ms
- Inter-trial variability σ_ϵ also has the same interpretation

Interpreting parameter estimates

`Intercept`	583.79
`neighbors.centered`	8.99
$\widehat{\sigma}_b$	70.18
$\widehat{\sigma}_\epsilon$	138.7

- The *fixed effects* are interpreted just as in a traditional single-level model:
  - The "average" RT for a non-word in this study is 583.79ms
  - Every extra neighbor increases "average" RT by 8.99ms
- Inter-trial variability σ_ϵ also has the same interpretation
  - Inter-trial variability for a given participant is Gaussian, centered around the participant+word-specific mean with standard deviation 138.7ms

Interpreting parameter estimates

Intercept	583.79
neighbors.centered	8.99
$\widehat{\sigma}_b$	70.18
$\widehat{\sigma}_\epsilon$	138.7

- The *fixed effects* are interpreted just as in a traditional single-level model:
  - The “average” RT for a non-word in this study is 583.79ms
  - Every extra neighbor increases “average” RT by 8.99ms
- Inter-trial variability σ_ϵ also has the same interpretation
  - Inter-trial variability for a given participant is Gaussian, centered around the participant+word-specific mean with standard deviation 138.7ms
- Inter-participant variability σ_b is what's new:

Interpreting parameter estimates

`Intercept`	583.79
`neighbors.centered`	8.99
$\widehat{\sigma}_b$	70.18
$\widehat{\sigma}_\epsilon$	138.7

- The *fixed effects* are interpreted just as in a traditional single-level model:
  - The "average" RT for a non-word in this study is 583.79ms
  - Every extra neighbor increases "average" RT by 8.99ms
- Inter-trial variability σ_ϵ also has the same interpretation
  - Inter-trial variability for a given participant is Gaussian, centered around the participant+word-specific mean with standard deviation 138.7ms
- Inter-participant variability σ_b is what's new:
  - Variability in average RT in the population from which the participants were drawn has standard deviation 70.18ms

Inferences about cluster-level parameters

$$\boxed{RT_{ij} = \alpha + \beta x_{ij} + \overbrace{b_i}^{\sim N(0,\sigma_b)} + \overbrace{\epsilon_{ij}}^{\text{Noise}\sim N(0,\sigma_\epsilon)}}$$

- What about the participants' idiosyncracies themselves—the b_i?

Inferences about cluster-level parameters

$$RT_{ij} = \alpha + \beta x_{ij} + \overbrace{b_i}^{\sim N(0,\sigma_b)} + \overbrace{\epsilon_{ij}}^{\text{Noise}\sim N(0,\sigma_\epsilon)}$$

- What about the participants' idiosyncracies themselves—the b_i?
- We can also draw inferences about these—once again, a common estimate of them is known as the BLUP

Inferences about cluster-level parameters

$$\boxed{RT_{ij} = \alpha + \beta x_{ij} + \overbrace{b_i}^{\sim N(0,\sigma_b)} + \overbrace{\epsilon_{ij}}^{\text{Noise}\sim N(0,\sigma_\epsilon)}}$$

- What about the participants' idiosyncracies themselves—the b_i?
- We can also draw inferences about these—once again, a common estimate of them is known as the BLUP
- To understand these: committing to fixed-effect and random-effect parameter estimates determines a conditional probability distribution on participant-specific effects:

$$P(b_i|\widehat{\alpha}, \widehat{\beta}, \widehat{\sigma}_b, \widehat{\sigma}_\epsilon)$$

Inferences about cluster-level parameters

$$RT_{ij} = \alpha + \beta x_{ij} + \overbrace{b_i}^{\sim N(0,\sigma_b)} + \overbrace{\epsilon_{ij}}^{\text{Noise}\sim N(0,\sigma_\epsilon)}$$

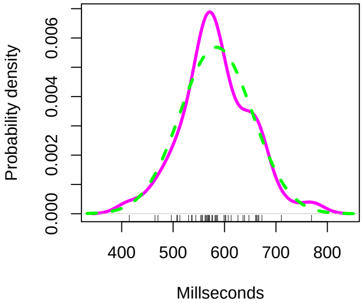
- What about the participants' idiosyncracies themselves—the b_i?
- We can also draw inferences about these—once again, a common estimate of them is known as the BLUP
- To understand these: committing to fixed-effect and random-effect parameter estimates determines a conditional probability distribution on participant-specific effects:

$$P(b_i | \widehat{\alpha}, \widehat{\beta}, \widehat{\sigma}_b, \widehat{\sigma}_\epsilon)$$

- The BLUPS are the conditional modes of b_i—the choices that maximize the above probability

Inferences about cluster-level parameters II

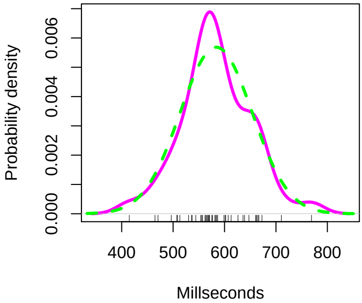
- The BLUP participant-specific “average” RTs for this dataset are black lines on the base of this graph



  - The solid line is a guess at their distribution

Inferences about cluster-level parameters II

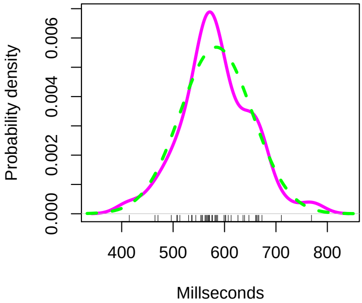
- The BLUP participant-specific “average” RTs for this dataset are black lines on the base of this graph



  - The solid line is a guess at their distribution
  - The dotted line is the distribution predicted by the model for the population from which the participants are drawn

Inferences about cluster-level parameters II

- The BLUP participant-specific "average" RTs for this dataset are black lines on the base of this graph



- The solid line is a guess at their distribution
- The dotted line is the distribution predicted by the model for the population from which the participants are drawn
- Reasonably close correspondence

Inference about cluster-level parameters III

- Participants may also have idiosyncratic sensitivities to *neighborhood density*

Inference about cluster-level parameters III

- Participants may also have idiosyncratic sensitivities to *neighborhood density*
- Incorporate by adding cluster-level slopes into the model:

$$RT_{ij} = \alpha + \beta x_{ij} + \overbrace{b_{1i} + b_{2i}}^{\sim N(0,\Sigma_b)} x_{ij} + \overbrace{\epsilon_{ij}}^{\text{Noise}\sim N(0,\sigma_\epsilon)}$$

Inference about cluster-level parameters III

- Participants may also have idiosyncratic sensitivities to *neighborhood density*
- Incorporate by adding cluster-level slopes into the model:

$$RT_{ij} = \alpha + \beta x_{ij} + \overbrace{b_{1i} + b_{2i}}^{\sim N(0,\Sigma_b)} x_{ij} + \overbrace{\epsilon_{ij}}^{\text{Noise}\sim N(0,\sigma_\epsilon)}$$

- In `R` (once again we can omit the `1`'s):
  `RT ∼ 1 + x + (1 + x | participant)`

Inference about cluster-level parameters III

- Participants may also have idiosyncratic sensitivities to *neighborhood density*
- Incorporate by adding cluster-level slopes into the model:

$$RT_{ij} = \alpha + \beta x_{ij} + \overbrace{b_{1i} + b_{2i}}^{\sim N(0,\Sigma_b)} x_{ij} + \overbrace{\epsilon_{ij}}^{\text{Noise}\sim N(0,\sigma_\epsilon)}$$

- In R (once again we can omit the 1's):
  `RT ~ 1 + x + (1 + x | participant)`

```
> lmer(RT ~ neighbors.centered +
    (neighbors.centered | participant), dat,REML=F)
[...]
Random effects:
 Groups      Name               Variance  Std.Dev. Corr
 participant (Intercept)         4928.625  70.2042
             neighbors.centered    19.421   4.4069 -0.307
 Residual                       19107.143 138.2286
```

Inference about cluster-level parameters III

- Participants may also have idiosyncratic sensitivities to *neighborhood density*
- Incorporate by adding cluster-level slopes into the model:

$$RT_{ij} = \alpha + \beta x_{ij} + \overbrace{b_{1i} + b_{2i}}^{\sim N(0,\Sigma_b)} x_{ij} + \overbrace{\epsilon_{ij}}^{\text{Noise}\sim N(0,\sigma_\epsilon)}$$

- In R (once again we can omit the 1's):
  `RT ~ 1 + x + (1 + x | participant)`

```
> lmer(RT ~ neighbors.centered +
    (neighbors.centered | participant), dat,REML=F)
[...]
Random effects:
 Groups      Name               Variance  Std.Dev. Corr
 participant (Intercept)         4928.625  70.2042
             neighbors.centered    19.421   4.4069 -0.307
 Residual                       19107.143 138.2286
```

These three numbers jointly characterize $\widehat{\Sigma}_b$

Inferences about cluster-level parameters IV

- Let's talk a little more about cluster-level slopes

$$RT_{ij} = \alpha + \beta x_{ij} + \overbrace{b_{1i} + b_{2i}}^{\sim N(0,\Sigma_b)} x_{ij} + \overbrace{\epsilon_{ij}}^{\text{Noise}\sim N(0,\sigma_\epsilon)}$$

Inferences about cluster-level parameters IV

- Let's talk a little more about cluster-level slopes

$$RT_{ij} = \alpha + \beta x_{ij} + \overbrace{b_{1i} + b_{2i}}^{\sim N(0,\Sigma_b)} x_{ij} + \overbrace{\epsilon_{ij}}^{\text{Noise}\sim N(0,\sigma_\epsilon)}$$

- We've said that participant-specific idiosyncracies are MULTIVARIATE NORMALLY DISTRIBUTED around the origin with covariance matrix Σ_b

```
Random effects:
 Groups      Name                  Variance  Std.Dev. Corr
 participant (Intercept)            4928.625  70.2042
             neighbors.centered       19.421   4.4069 -0.307
```

Inferences about cluster-level parameters IV

- Let's talk a little more about cluster-level slopes

$$RT_{ij} = \alpha + \beta x_{ij} + \overbrace{b_{1i} + b_{2i}}^{\sim N(0,\Sigma_b)} x_{ij} + \overbrace{\epsilon_{ij}}^{\text{Noise}\sim N(0,\sigma_\epsilon)}$$

- We've said that participant-specific idiosyncracies are MULTIVARIATE NORMALLY DISTRIBUTED around the origin with covariance matrix Σ_b

```
Random effects:
 Groups      Name                   Variance  Std.Dev. Corr
 participant (Intercept)             4928.625  70.2042
             neighbors.centered        19.421   4.4069 -0.307
```

- The results of the `lmer()` fit are saying that the maximum-likelihood estimate of the covariance matrix Σ_b governing participant-specific variability is

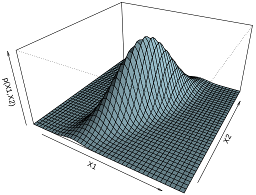
$$\widehat{\Sigma_b} = \begin{pmatrix} 70.20 & -0.3097 \\ -0.3097 & 4.41 \end{pmatrix}$$

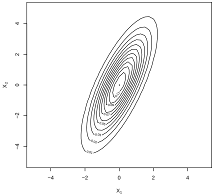
Inference about cluster-level parameters V

- Visualizing some multivariate normal distributions:

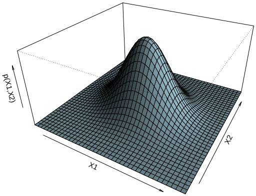
Covariance matrix | Perspective plot | Contour plot

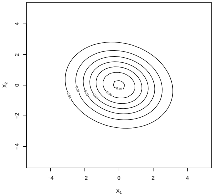
$$\Sigma_b = \begin{pmatrix} 1 & 0.75 \\ 0.75 & 4 \end{pmatrix}$$





$$\Sigma_b = \begin{pmatrix} 2.5 & -0.13 \\ -0.13 & 2 \end{pmatrix}$$

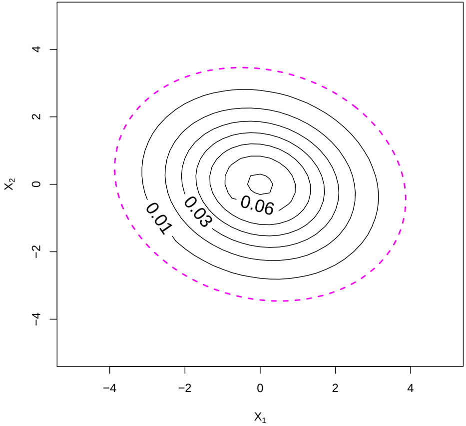




Inference about cluster-level parameters VI

- In 2D we often visually summarize a multivariate normal distribution with a CHARACTERISTIC ELLIPSE

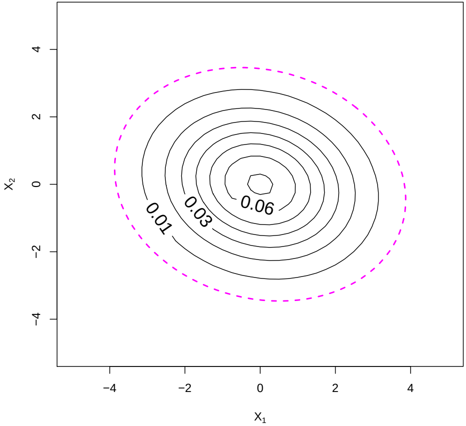
$$\Sigma_b = \begin{pmatrix} 1 & 0.75 \\ 0.75 & 4 \end{pmatrix}$$



Inference about cluster-level parameters VI

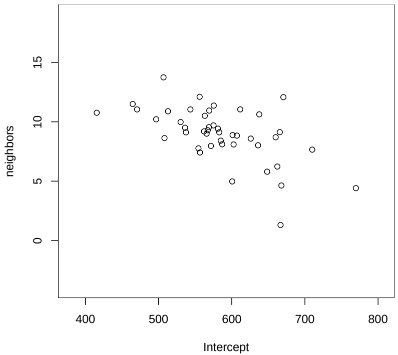
- In 2D we often visually summarize a multivariate normal distribution with a CHARACTERISTIC ELLIPSE

$$\Sigma_b = \begin{pmatrix} 1 & 0.75 \\ 0.75 & 4 \end{pmatrix}$$

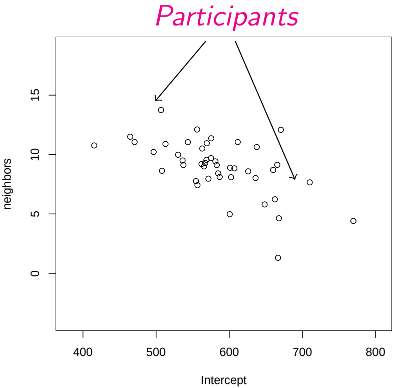


- This ellipse contains a certain proportion (here & conventionally, 95%) of the probability mass for the distribution in question

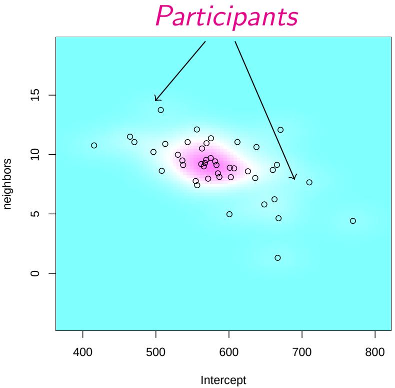
Inference about cluster-level parameters VII



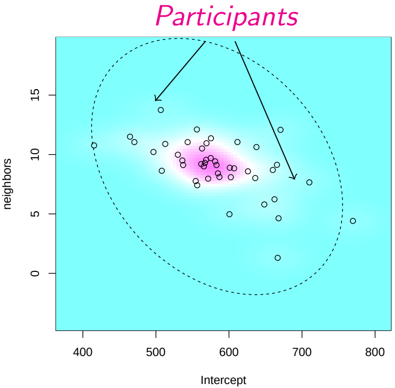
Inference about cluster-level parameters VII



Inference about cluster-level parameters VII

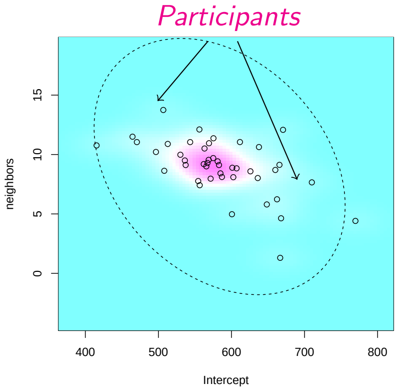


Inference about cluster-level parameters VII



- Correlation visible in participant-specific BLUPs

Inference about cluster-level parameters VII



- Correlation visible in participant-specific BLUPs
- Participants who were faster overall also tend to be more affected by neighborhood density

$$\widehat{\Sigma_b} = \begin{pmatrix} 70.20 & -0.3097 \\ -0.3097 & 4.41 \end{pmatrix}$$

Bayesian inference for multilevel models

$$P(\{\beta_i\}, \sigma_b, \sigma_\epsilon | Y) = \frac{\overbrace{P(Y|\{\beta_i\}, \sigma_b, \sigma_\epsilon)}^{\text{Likelihood}} \overbrace{P(\{\beta_i\}, \sigma_b, \sigma_\epsilon)}^{\text{Prior}}}{P(Y)}$$

- We can also use Bayes' rule to draw inferences about fixed effects

Bayesian inference for multilevel models

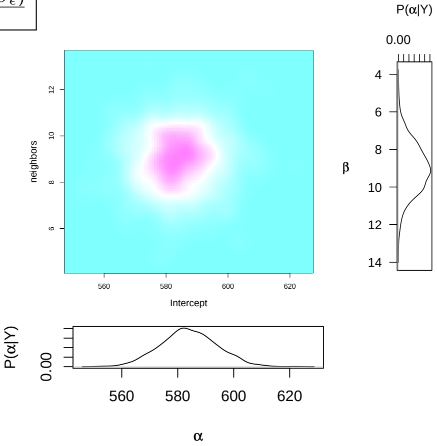
$$P(\{\beta_i\}, \sigma_b, \sigma_\epsilon | Y) = \frac{\overbrace{P(Y|\{\beta_i\}, \sigma_b, \sigma_\epsilon)}^{\text{Likelihood}} \overbrace{P(\{\beta_i\}, \sigma_b, \sigma_\epsilon)}^{\text{Prior}}}{P(Y)}$$

- We can also use Bayes' rule to draw inferences about fixed effects
- Computationally more challenging than with single-level regression; Markov-chain Monte Carlo (MCMC) sampling techniques allow us to approximate it

Bayesian inference for multilevel models

$$P(\{\beta_i\}, \sigma_b, \sigma_\epsilon | Y) = \frac{\overbrace{P(Y|\{\beta_i\}, \sigma_b, \sigma_\epsilon)}^{\text{Likelihood}} \overbrace{P(\{\beta_i\}, \sigma_b, \sigma_\epsilon)}^{\text{Prior}}}{P(Y)}$$

- We can also use Bayes' rule to draw inferences about fixed effects
- Computationally more challenging than with single-level regression; Markov-chain Monte Carlo (MCMC) sampling techniques allow us to approximate it



What random effects structure to use in drawing inferences about fixed effects?

- We looked at models that both *included* and *didn't include* random slopes for the #-neighbors effect in this dataset

What random effects structure to use in drawing inferences about fixed effects?

- We looked at models that both *included* and *didn't include* random slopes for the #-neighbors effect in this dataset
- Which one is the right model?

What random effects structure to use in drawing inferences about fixed effects?

- We looked at models that both *included* and *didn't include* random slopes for the #-neighbors effect in this dataset
- Which one is the right model?
- This is a very general problem with no one solution, but I'll describe what I think are good answers *for the situation where one ultimately wants to make inferences about the importance of a "fixed-effect" (shared) model parameter*

What random effects structure to use in drawing inferences about fixed effects?

- We looked at models that both *included* and *didn't include* random slopes for the #-neighbors effect in this dataset
- Which one is the right model?
- This is a very general problem with no one solution, but I'll describe what I think are good answers *for the situation where one ultimately wants to make inferences about the importance of a "fixed-effect" (shared) model parameter*
- There are two situations:

What random effects structure to use in drawing inferences about fixed effects?

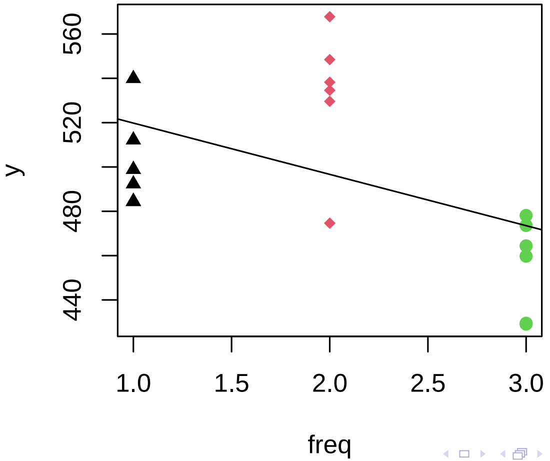
- We looked at models that both *included* and *didn't include* random slopes for the #-neighbors effect in this dataset
- Which one is the right model?
- This is a very general problem with no one solution, but I'll describe what I think are good answers *for the situation where one ultimately wants to make inferences about the importance of a "fixed-effect" (shared) model parameter*
- There are two situations:
    1. When the (average) value that a fixed effect takes *varies across clusters*

What random effects structure to use in drawing inferences about fixed effects?

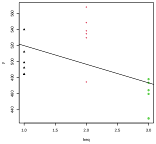
- We looked at models that both *included* and *didn't include* random slopes for the #-neighbors effect in this dataset
- Which one is the right model?
- This is a very general problem with no one solution, but I'll describe what I think are good answers *for the situation where one ultimately wants to make inferences about the importance of a "fixed-effect" (shared) model parameter*
- There are two situations:
  1. When the (average) value that a fixed effect takes *varies across clusters*
  2. When the value that a fixed effect takes *varies within some or all clusters*

Predictors varying between clusters

Hypothetical relationship observed for three words:

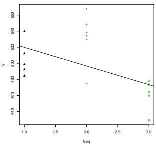


Predictors varying between clusters



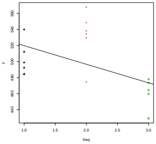
- If we were to ignore the potential cross-cluster variability here, it would look like we have good evidence for a word frequency effect

Predictors varying between clusters



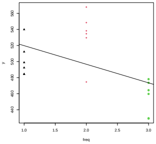
- If we were to ignore the potential cross-cluster variability here, it would look like we have good evidence for a word frequency effect
- But we have measurements for only three words!

Predictors varying between clusters



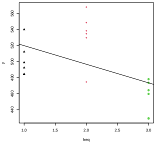
- If we were to ignore the potential cross-cluster variability here, it would look like we have good evidence for a word frequency effect
- But we have measurements for only three words!
- Suppose that there were no effect of word frequency, but words themselves varied idiosyncratically in their ease of recognition

Predictors varying between clusters



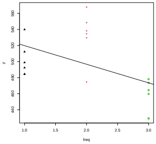
- If we were to ignore the potential cross-cluster variability here, it would look like we have good evidence for a word frequency effect
- But we have measurements for only three words!
- Suppose that there were no effect of word frequency, but words themselves varied idiosyncratically in their ease of recognition
- But the probability that the observed means would have this monotonicity would still be $\frac{1}{6}$

Predictors varying between clusters



- If we were to ignore the potential cross-cluster variability here, it would look like we have good evidence for a word frequency effect
- But we have measurements for only three words!
- Suppose that there were no effect of word frequency, but words themselves varied idiosyncratically in their ease of recognition
- But the probability that the observed means would have this monotonicity would still be $\frac{1}{6}$
- To address this issue we need a random intercept

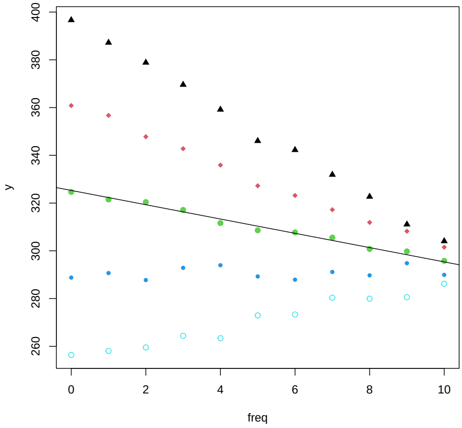
Predictors varying between clusters



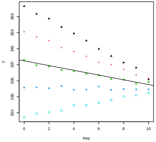
- If we were to ignore the potential cross-cluster variability here, it would look like we have good evidence for a word frequency effect
- But we have measurements for only three words!
- Suppose that there were no effect of word frequency, but words themselves varied idiosyncratically in their ease of recognition
- But the probability that the observed means would have this monotonicity would still be $\frac{1}{6}$
- To address this issue we need a random intercept
- Our model will wind up answering the question of whether there is a systematic trend across words for frequency sensitivity, *above and beyond idiosyncratic variation among*

Predictors varying within clusters

Hypothetical frequency-based responses for five different individual participants:



Predictors varying within clusters I



- It looks like we have good evidence for frequency-sensitivity of the response

Predictors varying within clusters II

- Classic question: *above and beyond idiosyncratic sensitivities of different individuals to context-driven predictability*, are predictable words in general named faster than unpredictable words?
- In mixed-effects models, this implies a need for a random by-speaker slope in our null-hypothesis model
- Inferences about the fixed effect will wind up meaning, *is there a systematic effect of word frequency, above and beyond idiosyncratic speaker-specific sensitivities to word frequency?*

The nonwords experiment

- The Bicknell et al. (2010) experiment had many different participants and many different nonwords

```
response ∼ X + (1 | Word) + (1 + X | Participant)
```

The nonwords experiment

- The Bicknell et al. (2010) experiment had many different participants and many different nonwords
- Each nonword has only one different number of neighbors, of course

```
response ~ X + (1 | Word) + (1 + X | Participant)
```

The nonwords experiment

- The Bicknell et al. (2010) experiment had many different participants and many different nonwords
- Each nonword has only one different number of neighbors, of course
- Each participant is exposed to nonwords with many different numbers of neighbors

$$\boxed{\texttt{response} \sim \texttt{X + (1 | Word) + (1 + X | Participant)}}$$

The nonwords experiment

- The Bicknell et al. (2010) experiment had many different participants and many different nonwords
- Each nonword has only one different number of neighbors, of course
- Each participant is exposed to nonwords with many different numbers of neighbors
- Hence, variation in neighborhood density is between-words but within-participant

$$\boxed{\texttt{response} \sim \texttt{X + (1 | Word) + (1 + X | Participant)}}$$

The nonwords experiment

- The Bicknell et al. (2010) experiment had many different participants and many different nonwords
- Each nonword has only one different number of neighbors, of course
- Each participant is exposed to nonwords with many different numbers of neighbors
- Hence, variation in neighborhood density is between-words but within-participant
- In the formula syntax of `R`'s `lme4` package:

```
response ~ X + (1 | Word) + (1 + X | Participant)
```

Hypothesis testing for LMEMs

- Exactly as for GLMs, the variance-covariance matrix of the fixed-effects covariance matrix contains a lot of information about confidence level in the parameters

Hypothesis testing for LMEMs

- Exactly as for GLMs, the variance-covariance matrix of the fixed-effects covariance matrix contains a lot of information about confidence level in the parameters
- It can be used to determine a t-statistic for each parameter

Hypothesis testing for LMEMs

- Exactly as for GLMs, the variance-covariance matrix of the fixed-effects covariance matrix contains a lot of information about confidence level in the parameters
- It can be used to determine a t-statistic for each parameter
- As with GLMs (but not as with LMs), the properties of this statistic are *asymptotic*—it is asymptotically normal

Hypothesis testing for LMEMs

- Exactly as for GLMs, the variance-covariance matrix of the fixed-effects covariance matrix contains a lot of information about confidence level in the parameters
- It can be used to determine a t-statistic for each parameter
- As with GLMs (but not as with LMs), the properties of this statistic are *asymptotic*—it is asymptotically normal
- Likewise, the likelihood-ratio test can be used to compare models differing in fixed effects structure alone

Hypothesis testing for LMEMs

- Exactly as for GLMs, the variance-covariance matrix of the fixed-effects covariance matrix contains a lot of information about confidence level in the parameters
- It can be used to determine a t-statistic for each parameter
- As with GLMs (but not as with LMs), the properties of this statistic are *asymptotic*—it is asymptotically normal
- Likewise, the likelihood-ratio test can be used to compare models differing in fixed effects structure alone
- It's slightly anticonservative, but not too bad in practice

Hypothesis testing for LMEMs

- Exactly as for GLMs, the variance-covariance matrix of the fixed-effects covariance matrix contains a lot of information about confidence level in the parameters
- It can be used to determine a t-statistic for each parameter
- As with GLMs (but not as with LMs), the properties of this statistic are *asymptotic*—it is asymptotically normal
- Likewise, the likelihood-ratio test can be used to compare models differing in fixed effects structure alone
- It's slightly anticonservative, but not too bad in practice
- Finally, models differing in random effects structure alone can *in principle* be compared with likelihood-ratio tests

Hypothesis testing for LMEMs

- Exactly as for GLMs, the variance-covariance matrix of the fixed-effects covariance matrix contains a lot of information about confidence level in the parameters
- It can be used to determine a t-statistic for each parameter
- As with GLMs (but not as with LMs), the properties of this statistic are *asymptotic*—it is asymptotically normal
- Likewise, the likelihood-ratio test can be used to compare models differing in fixed effects structure alone
- It's slightly anticonservative, but not too bad in practice
- Finally, models differing in random effects structure alone can *in principle* be compared with likelihood-ratio tests
    - However, these results can be either conservative or anti-conservative, so take them with a grain of salt

Results for the nonword-recognition experiment

```
dat$X <- dat$neighbors
m2 <- lmer(time ~ X + (1 + X | participant) + (1|target), dat,REML=F)

## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl =
control$checkConv, :  Model failed to converge with max|grad| =
0.0021026 (tol = 0.002, component 1)

print(m2,corr=F)

## Linear mixed model fit by maximum likelihood  ['lmerMod']
## Formula: time ~ X + (1 + X | participant) + (1 | target)
##    Data: dat
##       AIC       BIC    logLik  deviance  df.resid
##  22451.93  22490.24 -11218.96  22437.93      1753
## Random effects:
##  Groups      Name        Std.Dev. Corr
##  participant (Intercept)  76.140
##              X             4.803  -0.46
##  target      (Intercept)  25.485
##  Residual                135.763
## Number of obs: 1760, groups:  participant, 44; target, 40
## Fixed Effects:
```

Principles of random-effects specification I

- There has been disagreement/unclarity regarding how to specify random-effects structure for one's model

Principles of random-effects specification I

- There has been disagreement/unclarity regarding how to specify random-effects structure for one's model
  - Random intercepts are enough?

Principles of random-effects specification I

- There has been disagreement/unclarity regarding how to specify random-effects structure for one's model
  - Random intercepts are enough?
  - Start with random intercepts and then use model selection?

Principles of random-effects specification I

- There has been disagreement/unclarity regarding how to specify random-effects structure for one's model
  - Random intercepts are enough?
  - Start with random intercepts and then use model selection?
  - Maximal random effect structure, backing off to random intercepts if there are convergence problems?

Principles of random-effects specification I

- There has been disagreement/unclarity regarding how to specify random-effects structure for one's model
  - Random intercepts are enough?
  - Start with random intercepts and then use model selection?
  - Maximal random effect structure, backing off to random intercepts if there are convergence problems?
- In Barr et al. (2013) we have taken a strong but, we believe, traditional stand (really following Clark, 1973):

  Random-effect structure should be maximal with respect to the theoretically critical questions you are posing of your data.

Principles of random-effects specification I

- There has been disagreement/unclarity regarding how to specify random-effects structure for one's model
  - Random intercepts are enough?
  - Start with random intercepts and then use model selection?
  - Maximal random effect structure, backing off to random intercepts if there are convergence problems?
- In Barr et al. (2013) we have taken a strong but, we believe, traditional stand (really following Clark, 1973):

  Random-effect structure should be maximal with respect to the theoretically critical questions you are posing of your data.
- **This position has been widely** (though not universally) **accepted by the field, and we continue to advocate for it**

Principles of random-effects specification II

- For traditional, balanced designs with a small number of theoretically critical predictor, this means:

Principles of random-effects specification II

- For traditional, balanced designs with a small number of theoretically critical predictor, this means:
  - For every theoretically critical fixed-effect term in your model that varies *between* clusters (e.g., subjects or items), include a random intercept for that clustering

Principles of random-effects specification II

- For traditional, balanced designs with a small number of theoretically critical predictor, this means:
  - For every theoretically critical fixed-effect term in your model that varies *between* clusters (e.g., subjects or items), include a random intercept for that clustering
  - For every theoretically critical fixed-effect term in your model that varies *within* clusters, include a random slope for that clustering

A controlled experiment I

- Rohde et al. (2011) used self-paced reading to assess the real-time deployment of discourse knowledge in syntactic ambiguity resolution

A controlled experiment I

- Rohde et al. (2011) used self-paced reading to assess the real-time deployment of discourse knowledge in syntactic ambiguity resolution
- Sample item:
  John babysat the children of the musician who. . .

A controlled experiment I

- Rohde et al. (2011) used self-paced reading to assess the real-time deployment of discourse knowledge in syntactic ambiguity resolution
- Sample item:
  John babysat the children of the musician who…
  - …*was* generally arrogant and rude.

A controlled experiment I

- Rohde et al. (2011) used self-paced reading to assess the real-time deployment of discourse knowledge in syntactic ambiguity resolution
- Sample item:
  John babysat the children of the musician who…
  - …*was* generally arrogant and rude.
  - …*were* generally arrogant and rude.

A controlled experiment I

- Rohde et al. (2011) used self-paced reading to assess the real-time deployment of discourse knowledge in syntactic ambiguity resolution
- Sample item:
  John babysat the children of the musician who...
  - ...*was* generally arrogant and rude.
  - ...*were* generally arrogant and rude.
- Sample item in implicit causality condition:
  John *detested* the children of the musician who...

A controlled experiment I

- Rohde et al. (2011) used self-paced reading to assess the real-time deployment of discourse knowledge in syntactic ambiguity resolution
- Sample item:
  John babysat the children of the musician who. . .
  - . . . *was* generally arrogant and rude.
  - . . . *were* generally arrogant and rude.
- Sample item in implicit causality condition:
  John *detested* the children of the musician who. . .
  - . . . *was* generally arrogant and rude.

A controlled experiment I

- Rohde et al. (2011) used self-paced reading to assess the real-time deployment of discourse knowledge in syntactic ambiguity resolution
- Sample item:
  John babysat the children of the musician who...
  - ...*was* generally arrogant and rude.
  - ...*were* generally arrogant and rude.
- Sample item in implicit causality condition:
  John *detested* the children of the musician who...
  - ...*was* generally arrogant and rude.
  - ...*were* generally arrogant and rude.

A controlled experiment I

- Rohde et al. (2011) used self-paced reading to assess the real-time deployment of discourse knowledge in syntactic ambiguity resolution
- Sample item:
  John babysat the children of the musician who...
  - ... *was* generally arrogant and rude.
  - ... *were* generally arrogant and rude.
- Sample item in implicit causality condition:
  John *detested* the children of the musician who...
  - ... *was* generally arrogant and rude.
  - ... *were* generally arrogant and rude.
- The question of theoretical interest for our data is whether the processing penalty induced by disambiguation of the RC attachment would show up immediately (before potentially biasing semantic content of the RC shows up).

A controlled experiment II

- In self-paced reading, many kinds of word properties show up primarily in reading times *one or more words downstream* ("spillover" effects)

A controlled experiment II

- In self-paced reading, many kinds of word properties show up primarily in reading times *one or more words downstream* ("spillover" effects)
- Thus we focus on statistical analysis of the word immediately after disambiguation:

  John $\overbrace{\textit{babysat/detected}}^{\text{V}}$ *the children of the musician who* $\overbrace{\textit{was/were}}^{\text{B}}$ *generally arrogant and rude*

A controlled experiment II

- In self-paced reading, many kinds of word properties show up primarily in reading times *one or more words downstream* ("spillover" effects)
- Thus we focus on statistical analysis of the word immediately after disambiguation:

  John $\overbrace{\textit{babysat/detested}}^{\text{V}}$ *the children of the musician who* $\overbrace{\textit{was/were}}^{\text{B}}$ *generally arrogant and rude*

- We'll abbreviate the type of verb (implicit causality or not) the **V** factor and the RC's attachment level (high or low) the **A** factor

A controlled experiment II

- In self-paced reading, many kinds of word properties show up primarily in reading times *one or more words downstream* ("spillover" effects)
- Thus we focus on statistical analysis of the word immediately after disambiguation:

  $$\textit{John}\ \overbrace{\textit{babysat/detested}}^{\text{V}}\ \textit{the children of the musician who}\ \overbrace{\textit{was/were}}^{\text{B}}\ \textit{generally arrogant and rude}$$

- We'll abbreviate the type of verb (implicit causality or not) the **V** factor and the RC's attachment level (high or low) the **A** factor
- These factors are crossed in the experiment, and both within-subject. We **sum-code** the two factors to give the main effects the desired interpretation.

A controlled experiment III

A controlled experiment III

- Results of a maximal LME fit:

```
## Linear mixed model fit by maximum likelihood  ['lmerMod']
## Formula: rt ~ V * A + (V * A | subj) + (V * A | item)
##    Data: d
##
##      AIC      BIC   logLik deviance df.resid
##  12527.4  12648.0  -6238.7  12477.4      894
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.0464 -0.5641 -0.1500  0.2521  6.0545
##
## Random effects:
##  Groups   Name        Variance Std.Dev. Corr
##  subj     (Intercept) 16769.20 129.496
##           V             476.13  21.820  -0.82
##           A              24.39   4.939  -0.94  0.96
##           V:A         12433.81 111.507  -0.48  0.90  0.75
##  item     (Intercept)  1520.66  38.996
##           V            2055.65  45.339  -0.80
##           A            1663.59  40.787   0.06 -0.64
##           V:A          5531.05  74.371   0.04  0.56 -1.00
##  Residual             38497.55 196.208
## Number of obs: 919, groups:  subj, 55; item, 20
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept) 470.4938    20.5949  22.845
## V           -33.7621    16.8373  -2.005
## A            -0.1967    15.9802  -0.012
## V:A         -85.0056    34.4913  -2.465
## optimizer (nloptwrap) convergence code: 0 (OK)
## boundary (singular) fit: see help('isSingular')
```

A controlled experiment III

- Likelihood-ratio-based hypothesis testing for a fixed effect:

```
## Error:  bad 'data':  object 'd' not found
```

```
rt.lmer.null <- lmer(rt ~ V + A + ( V*A | subj)  + ( V*A | item),
  data=d,REML=F)
```

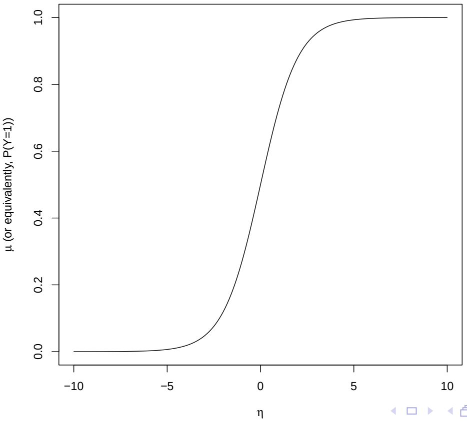
```
print(anova(rt.lmer.full,rt.lmer.null))

## Error in h(simpleError(msg, call)):  error in evaluating the
argument 'x' in selecting a method for function 'print':  object
'rt.lmer.null' not found
```

```
##
## Attaching package:  'bayesm'
## The following object is masked from
'package:brms':
##
##     rdirichlet
```

Mixed logit models

Recall the inverse logit function that we used for logistic regression:

$$\mu = \frac{e^{\eta}}{1 + e^{\eta}}$$



Mixed logit models

- A generalized linear mixed model (GLMM) works exactly the same as an LME model; the cluster-level variables contribute to the linear predictor

Mixed logit models

- A generalized linear mixed model (GLMM) works exactly the same as an LME model; the cluster-level variables contribute to the linear predictor
- A mixed logit model thus has the logit link function:

$$\eta = \log \frac{\mu}{1 - \mu}$$

Mixed logit models

- A generalized linear mixed model (GLMM) works exactly the same as an LME model; the cluster-level variables contribute to the linear predictor
- A mixed logit model thus has the logit link function:

$$\eta = \log \frac{\mu}{1-\mu}$$

- Bernoulli noise distribution around predicted mean μ:

$$P(Y = y|\mu) = \begin{cases} \mu & y = 1 \\ 1-\mu & y = 0 \\ 0 & \text{otherwise} \end{cases}$$

Mixed logit models

- A generalized linear mixed model (GLMM) works exactly the same as an LME model; the cluster-level variables contribute to the linear predictor
- A mixed logit model thus has the logit link function:

$$\eta = \log \frac{\mu}{1-\mu}$$

- Bernoulli noise distribution around predicted mean μ:

$$P(Y=y|\mu) = \begin{cases} \mu & y=1 \\ 1-\mu & y=0 \\ 0 & \text{otherwise} \end{cases}$$

- And linear predictor

$$\eta = \mathrm{X}\boldsymbol{\beta} + \mathrm{Zb}$$

where b is multivariate-normal distributed:

$$\mathrm{b} \sim N(0, \Sigma_{\mathrm{b}})$$

References I

Barr, D. J., Levy, R., Scheepers, C., and Tily, H. J. (2013). Random effects structure for confirmatory hypothesis testing: Keep it maximal. *Journal of Memory and Language*, 68(3):255–278.

Bicknell, K., Elman, J. L., Hare, M., McRae, K., and Kutas, M. (2010). Effects of event knowledge in processing verbal arguments. *Journal of Memory and Language*, 63:489–505.

Clark, H. H. (1973). The language-as-fixed-effect fallacy: A critique of language statistics in psychological research. *Journal of Verbal Learning and Verbal Behavior*, 12:335–359.

Rohde, H., Levy, R., and Kehler, A. (2011). Anticipating explanations in relative clause processing. *Cognition*, 118(3):339–358.

A note on p-values and philosophy of science

- Frequentist hypothesis testing means the Neyman-Pearson paradigm, with an asymmetry between null (H_0) and alternative (H_1) hypotheses

A note on p-values and philosophy of science

- Frequentist hypothesis testing means the Neyman-Pearson paradigm, with an asymmetry between null (H_0) and alternative (H_1) hypotheses
- A p-value from a dataset D is how unlikely a given dataset was to be produced under H_0

A note on p-values and philosophy of science

- Frequentist hypothesis testing means the Neyman-Pearson paradigm, with an asymmetry between null (H_0) and alternative (H_1) hypotheses
- A p-value from a dataset D is how unlikely a given dataset was to be produced under H_0
- Note that so-called "p_{MCMC}" is NOT a p-value in the Neyman-Pearson sense!

A note on p-values and philosophy of science

- Frequentist hypothesis testing means the Neyman-Pearson paradigm, with an asymmetry between null (H_0) and alternative (H_1) hypotheses
- A p-value from a dataset D is how unlikely a given dataset was to be produced under H_0
- Note that so-called "p_{MCMC}" is NOT a p-value in the Neyman-Pearson sense!
- Weakness, both in practice and in principle: the alternative hypothesis is never actually used (except indirectly in determining optimal acceptance and rejection regions)

A note on p-values and philosophy of science

- Alternative: Bayesian hypothesis testing, which is symmetric:

$$\frac{P(H_0|D)}{P(H_1|D)} = \frac{P(D|H_0)}{P(D|H_1)} \frac{P(H_0)}{P(H_1)}$$

A note on p-values and philosophy of science

- Alternative: Bayesian hypothesis testing, which is symmetric:

$$\frac{P(H_0|D)}{P(H_1|D)} = \frac{P(D|H_0)}{P(D|H_1)} \frac{P(H_0)}{P(H_1)}$$

- I am fundamentally Bayesian in my philosophy of science

A note on p-values and philosophy of science

- Alternative: Bayesian hypothesis testing, which is symmetric:

$$\frac{P(H_0|D)}{P(H_1|D)} = \frac{P(D|H_0)}{P(D|H_1)} \frac{P(H_0)}{P(H_1)}$$

- I am fundamentally Bayesian in my philosophy of science
- But, weakness in practice: your likelihoods $P(D|H_0)$ and $P(D|H_1)$ can depend on fine details of your assumptions about H_0 and H_1

A note on p-values and philosophy of science

- Alternative: Bayesian hypothesis testing, which is symmetric:

$$\frac{P(H_0|D)}{P(H_1|D)} = \frac{P(D|H_0)}{P(D|H_1)} \frac{P(H_0)}{P(H_1)}$$

- I am fundamentally Bayesian in my philosophy of science
- But, weakness in practice: your likelihoods $P(D|H_0)$ and $P(D|H_1)$ can depend on fine details of your assumptions about H_0 and H_1
- I do not trust you to assess these likelihoods neutrally! (Nor should you trust me)

A note on p-values and philosophy of science

- Alternative: Bayesian hypothesis testing, which is symmetric:

$$\frac{P(H_0|D)}{P(H_1|D)} = \frac{P(D|H_0)}{P(D|H_1)} \frac{P(H_0)}{P(H_1)}$$

- I am fundamentally Bayesian in my philosophy of science
- But, weakness in practice: your likelihoods $P(D|H_0)$ and $P(D|H_1)$ can depend on fine details of your assumptions about H_0 and H_1
- I do not trust you to assess these likelihoods neutrally! (Nor should you trust me)
- So for me, the p-value of your experiment serves as a rough indicator of how small $P(D|H_0)$ may be

A note on p-values and philosophy of science

- Alternative: Bayesian hypothesis testing, which is symmetric:

$$\frac{P(H_0|D)}{P(H_1|D)} = \frac{P(D|H_0)}{P(D|H_1)}\frac{P(H_0)}{P(H_1)}$$

- I am fundamentally Bayesian in my philosophy of science
- But, weakness in practice: your likelihoods $P(D|H_0)$ and $P(D|H_1)$ can depend on fine details of your assumptions about H_0 and H_1
- I do not trust you to assess these likelihoods neutrally! (Nor should you trust me)
- So for me, the p-value of your experiment serves as a rough indicator of how small $P(D|H_0)$ may be
- Technically, such a measure doesn't need to be a true Neyman-Pearson p-value (p_{MCMC} falls into this category)