

MULTILEVEL AND HIERARCHICAL MODELS

Shrinkage, partial pooling, and mixed linear models

Diogo Melo
`damelo@ib.usp.br`

MULTILEVEL MODELS

- Multilevel models introduce structure to the parameter distribution
- Allows us to model several common processes
 - Repeated measures of the same individual
 - Block designs (some observations more similar than others)
 - Pseudo-replicates
 - Correlated observations
- Even in the absence of this type of structure, multilevel models can improve model performance

BASEBALL EXAMPLE

- Batting averages
- p is the probability that an attempted hit is successful.
- We can estimate the season long avg. of a player using a small sample in the beginning of the season.

ML estimate:

$$\hat{p} = N_{hits}/N_{attempts}$$

HYPER PARAMETERS

- We want to introduce some dependency between the different player's avgs.
- This allows the information in the joint distribution of all players to inform individual estimates.
- All estimates p_i come from the same distribution, and, therefore, depend on each other.

Data:

- y_i : number of hits
- n_i : number of attempts

Model:

$$y_i \sim \text{Binomial}(n_i, p_i)$$

$$p_i \sim \text{Beta}(\nu_1, \nu_2)$$

$$\nu_1, \nu_2 \sim \text{lognormal}(0, 1)$$

HYPER PARAMETERS

- We want to introduce some dependency between the different player's avgs.
- This allows the information in the joint distribution of all players to inform individual estimates.
- All estimates p_i come from the same distribution, and, therefore, depend on each other.

Data:

- y_i : number of hits
- n_i : number of attempts

Model:

$$y_i \sim \text{Binomial}(n_i, p_i)$$

$$p_i \sim \text{Beta}(\nu_1, \nu_2)$$

$$\nu_1, \nu_2$$

$$\sim \text{lognormal}(0, 1)$$

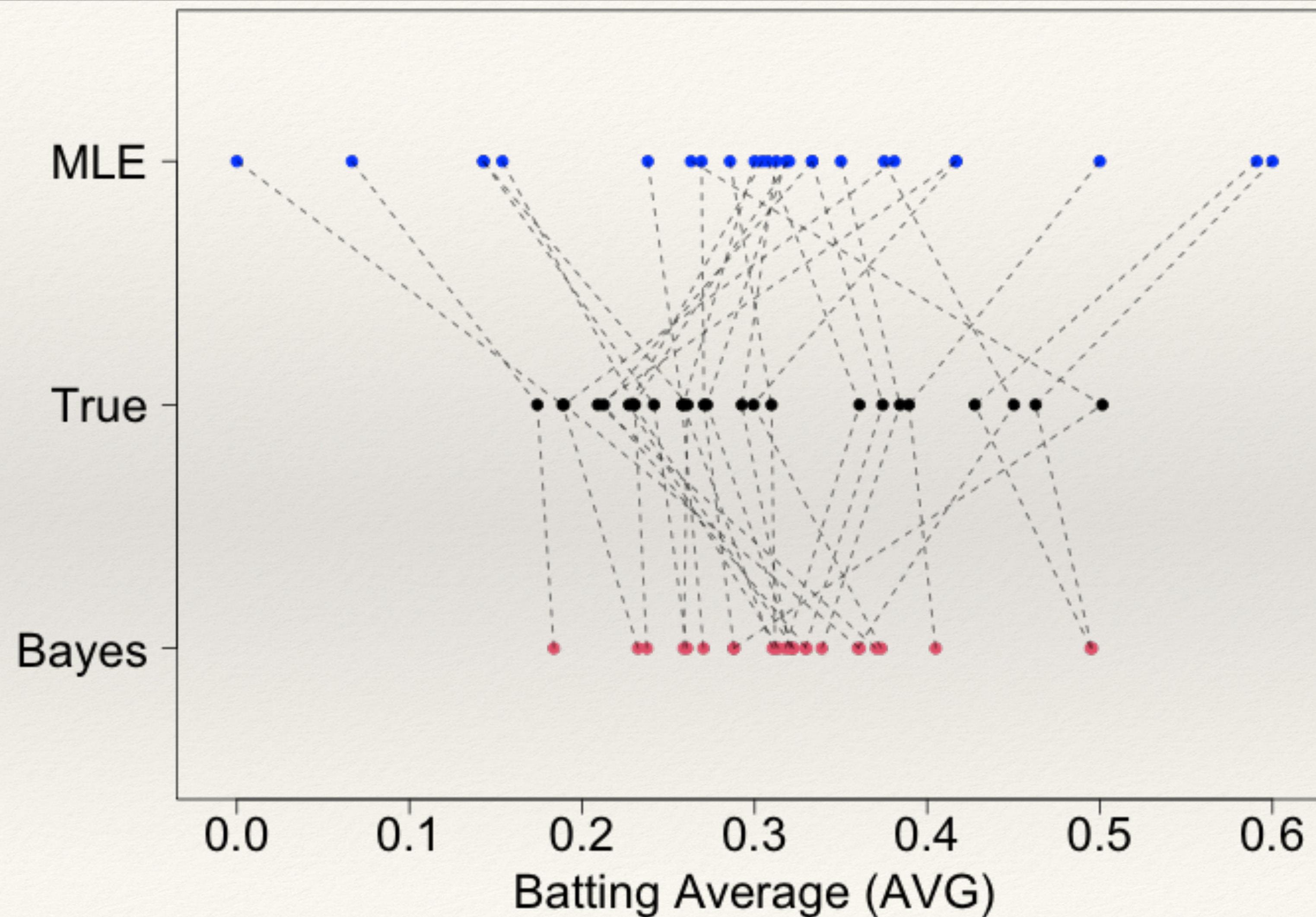
Hyper parameters

Hyper priors

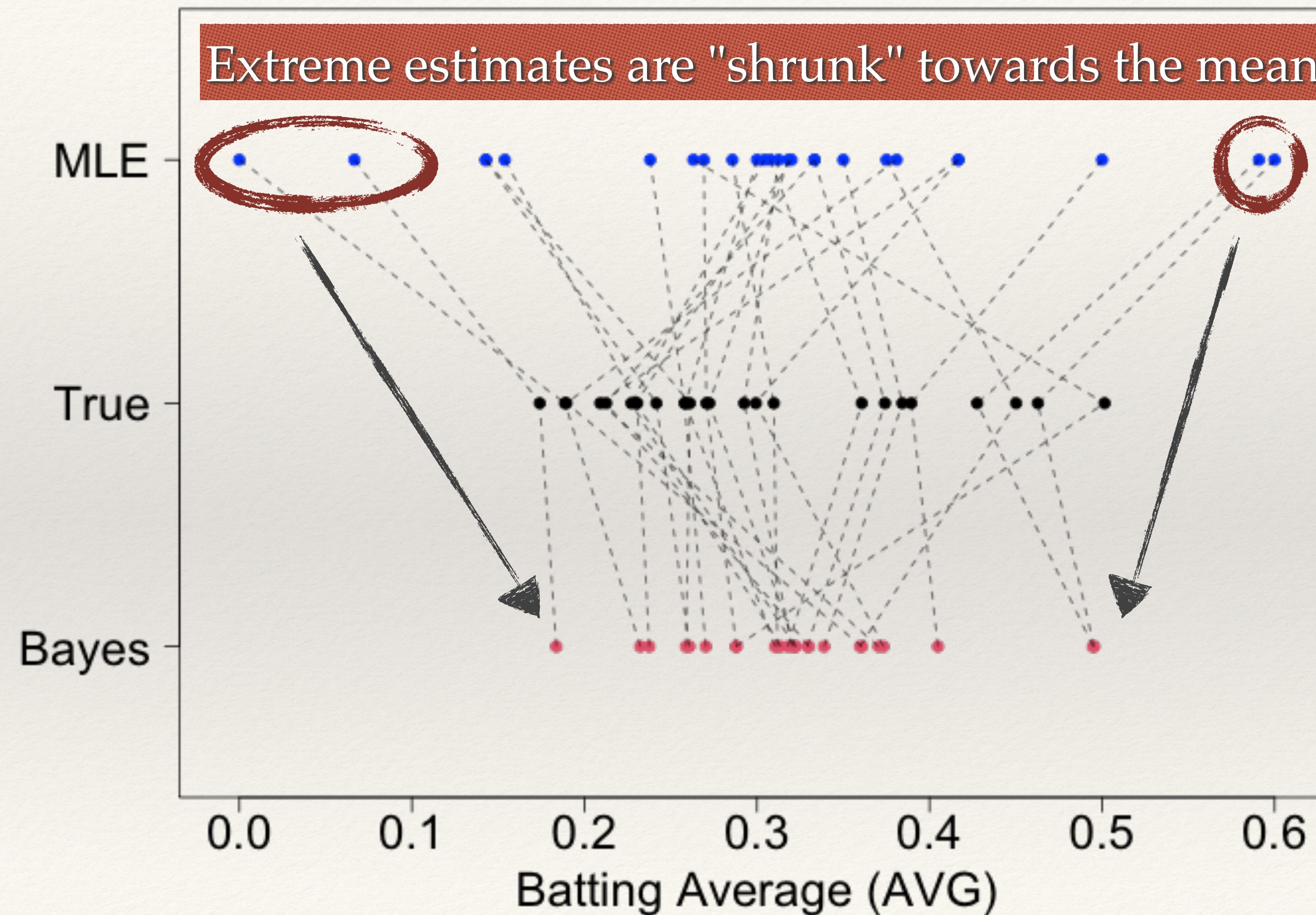
BASEBALL IN ULAM

```
m1 = ulam(alist(
    hits ~ binomial(atbats, avgs),
    avgs <p[player],
    p[player] ~ beta(nu0, nu1),
    nu0 ~ lognormal(0, 1),
    nu1 ~ lognormal(0, 1)
), data = list(hits = d[,2],
               atbats = d[,1],
               player = 1:Nplayers),
chains = 4, cores = 4, iter = 2000)
```

BATTING AVERAGES



BATTING AVERAGES

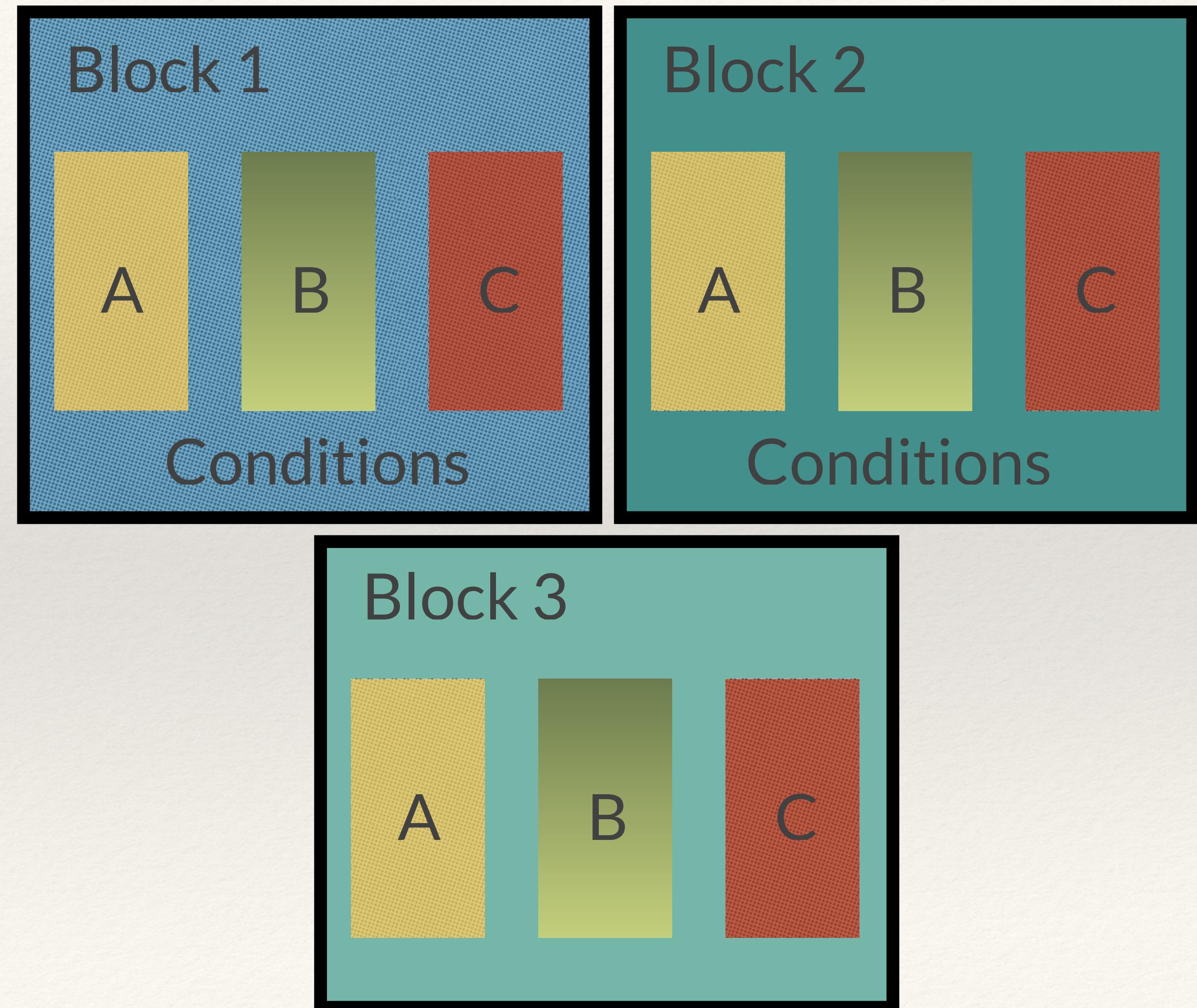


SHRINKAGE AND TYPES OF POOLING

1. **Complete pooling** Only one estimate for all players (underfitting)
3. **No pooling** Players are all independent, no information is shared (overfitting)
5. **Partial pooling** Players' estimates help inform each other. This means using an adaptive regularizing prior, as in the previous example.

BLOCKED EXPERIMENTAL DESIGNS

- Often in experiments there are natural blocks that exist non-intentionally.
- Even though these blocks were not created intentionally, they often still group experimental units that are more similar within blocks than between blocks due to unknown conditions.
- Analyzing results within each natural block makes it possible to separate treatment variability from variability due to the blocking factor that occurs unintentionally but systematically.



ADDING BLOCKS USING MIXED MODELS

$$y_i \sim \text{Normal}(\mu_i, \sigma)$$

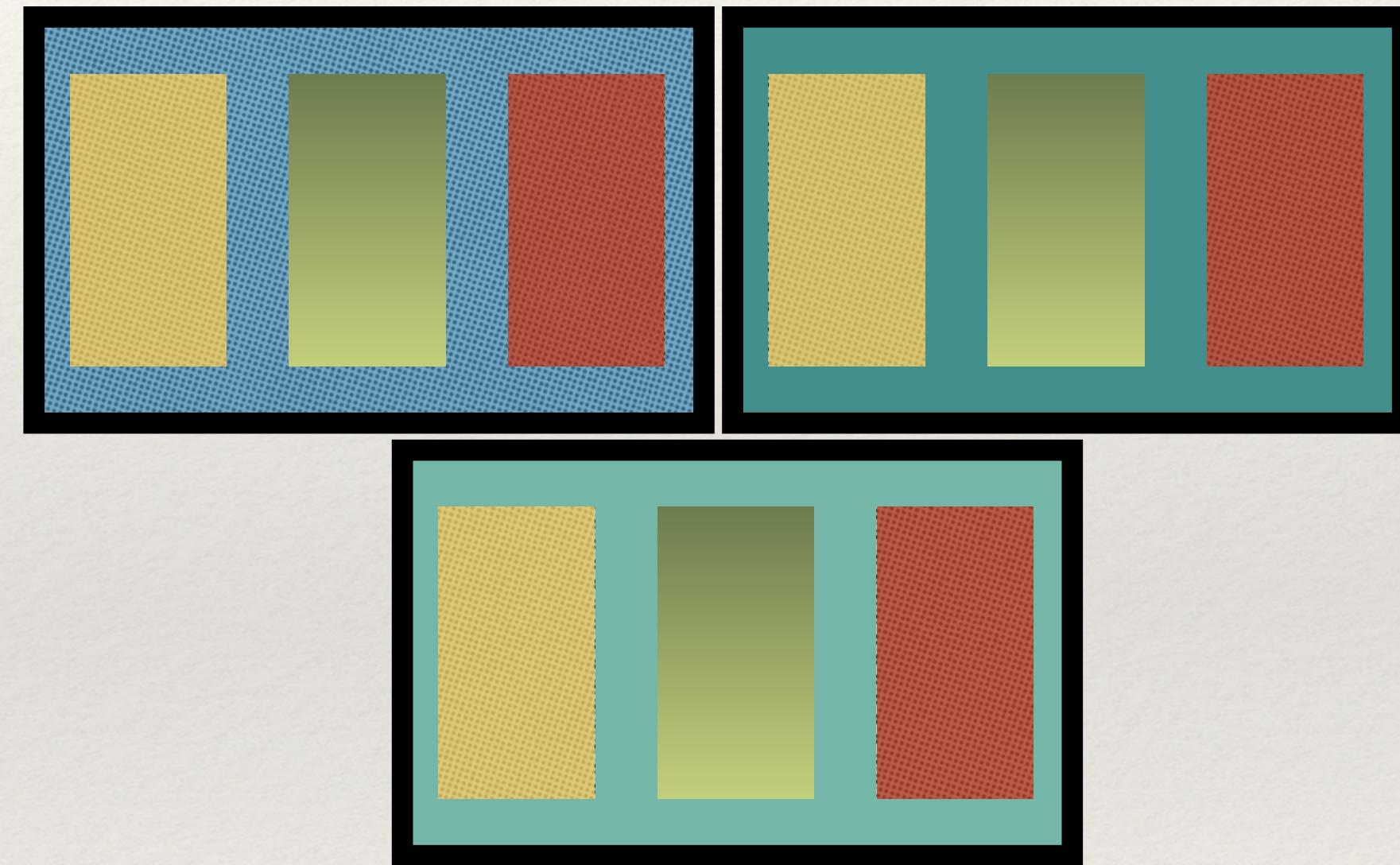
$$\mu_i = \alpha_0 + \alpha_{block[i]} + \beta x_i$$

$$\alpha_k \sim \text{Normal}(0, \sigma_\alpha), \text{ for } k \in \{1, \dots, N_{blocks}\}$$

$$\alpha_0 \sim \text{Normal}(0, 1)$$

$$\beta \sim \text{Normal}(0, 0.3)$$

$$\sigma, \sigma_{block} \sim \text{Exponential}(1)$$



ADDING BLOCKS USING MIXED MODELS

$$y_i \sim \text{Normal}(\mu_i, \sigma)$$

$$\mu_i = \alpha_0 + \alpha_{block[i]} + \beta x_i$$

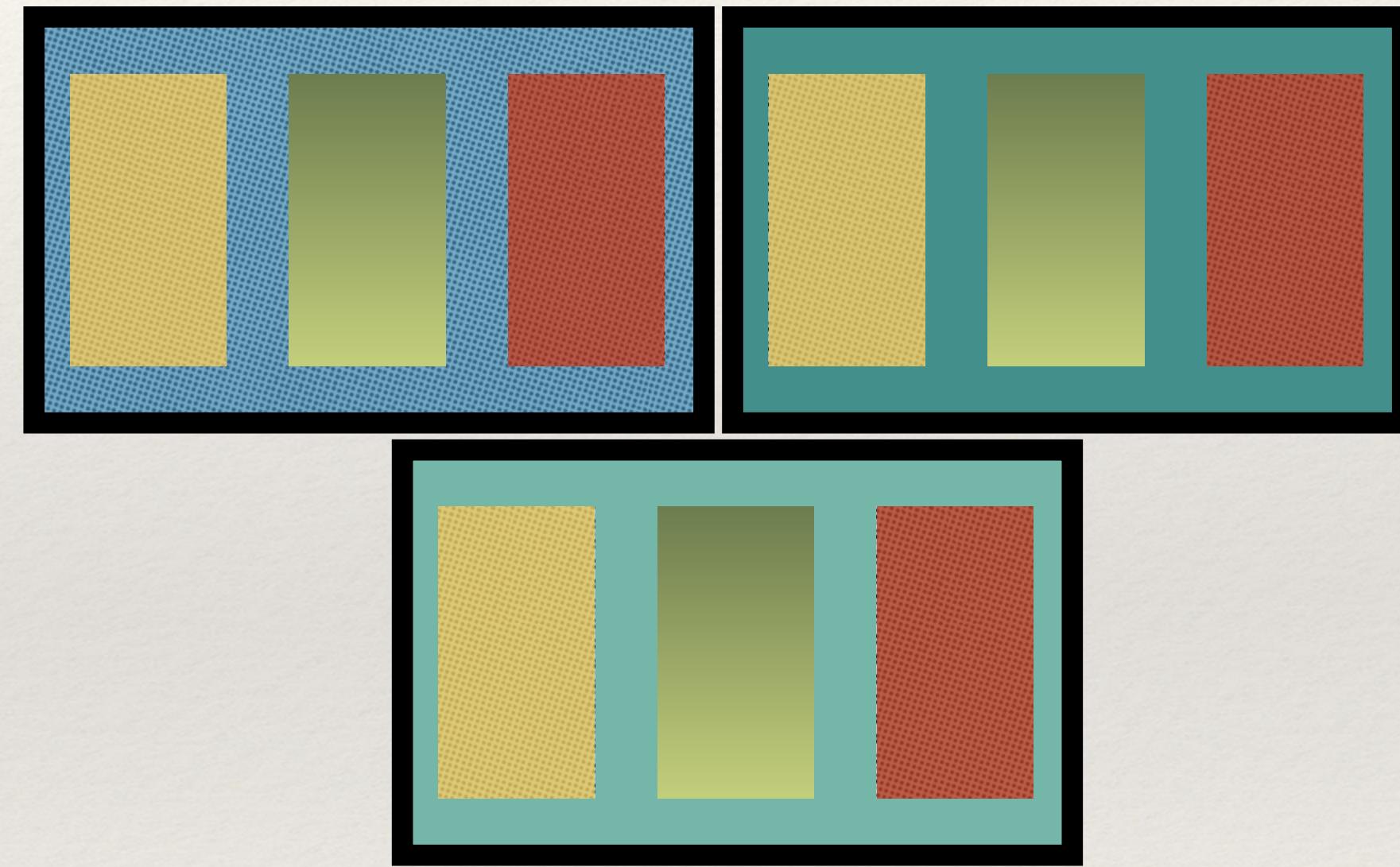
$$\alpha_k \sim \text{Normal}(0, \sigma_\alpha), \text{ for } k \text{ in } \{1, \dots, N_{blocks}\}$$

$$\alpha_0 \sim \text{Normal}(0, 1)$$

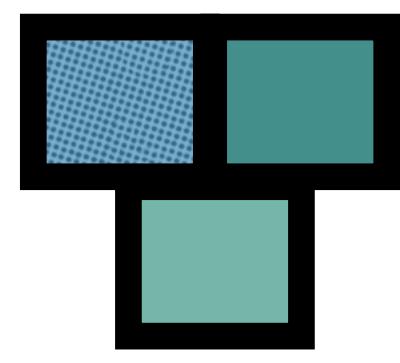
$$\beta \sim \text{Normal}(0, 0.3) \rightarrow$$



$$\sigma, \sigma_{block} \sim \text{Exponential}(1)$$



ADDING BLOCKS USING MIXED MODELS



$$y_i \sim \text{Normal}(\mu_i, \sigma)$$

$$\mu_i = \alpha_0 + \alpha_{block[i]} + \beta x_i$$

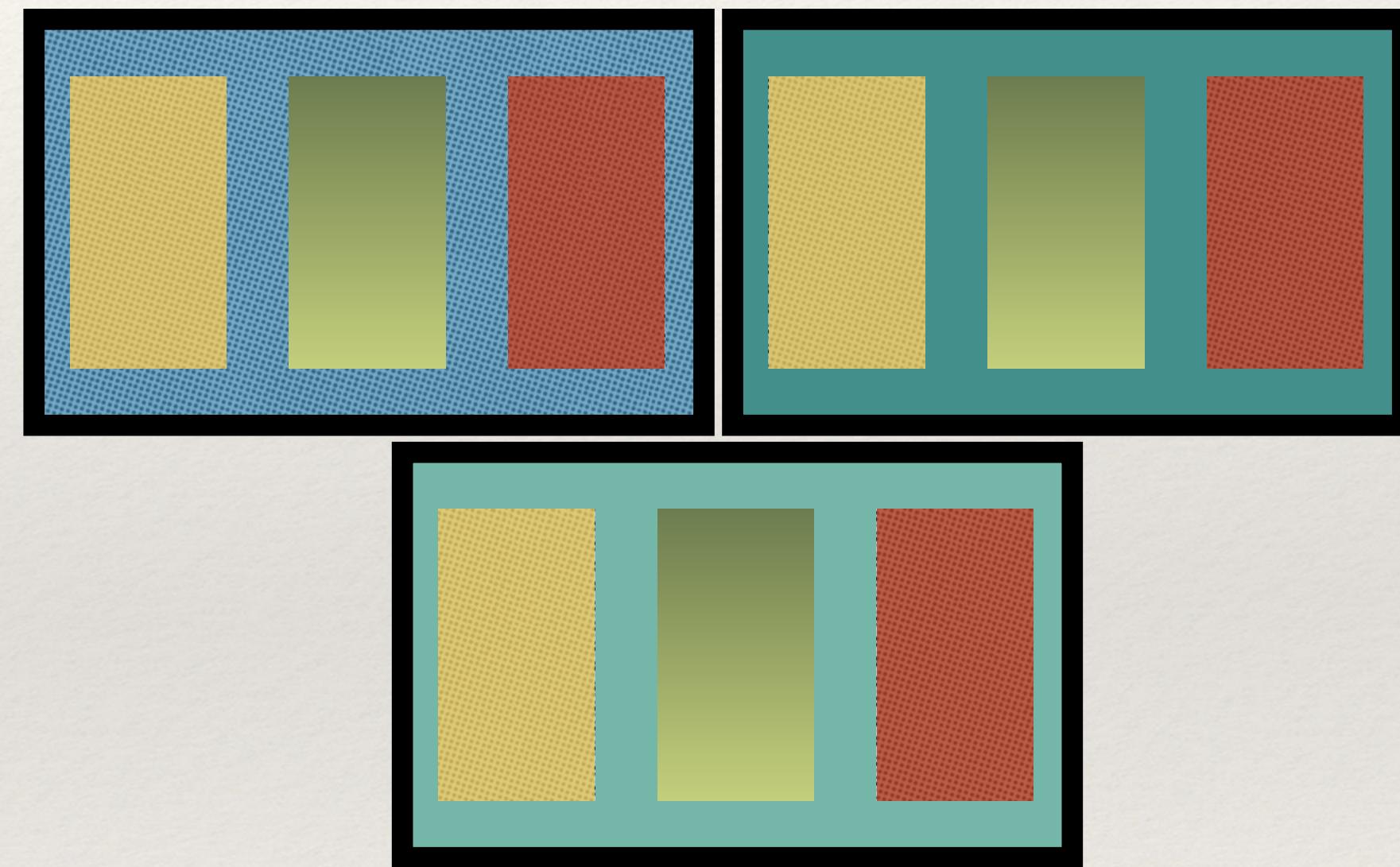
$$\alpha_k \sim \text{Normal}(0, \sigma_\alpha), \text{ for } k \text{ in } \{1, \dots, N_{blocks}\}$$

$$\alpha_0 \sim \text{Normal}(0, 1)$$

$$\beta \sim \text{Normal}(0, 0.3) \rightarrow$$



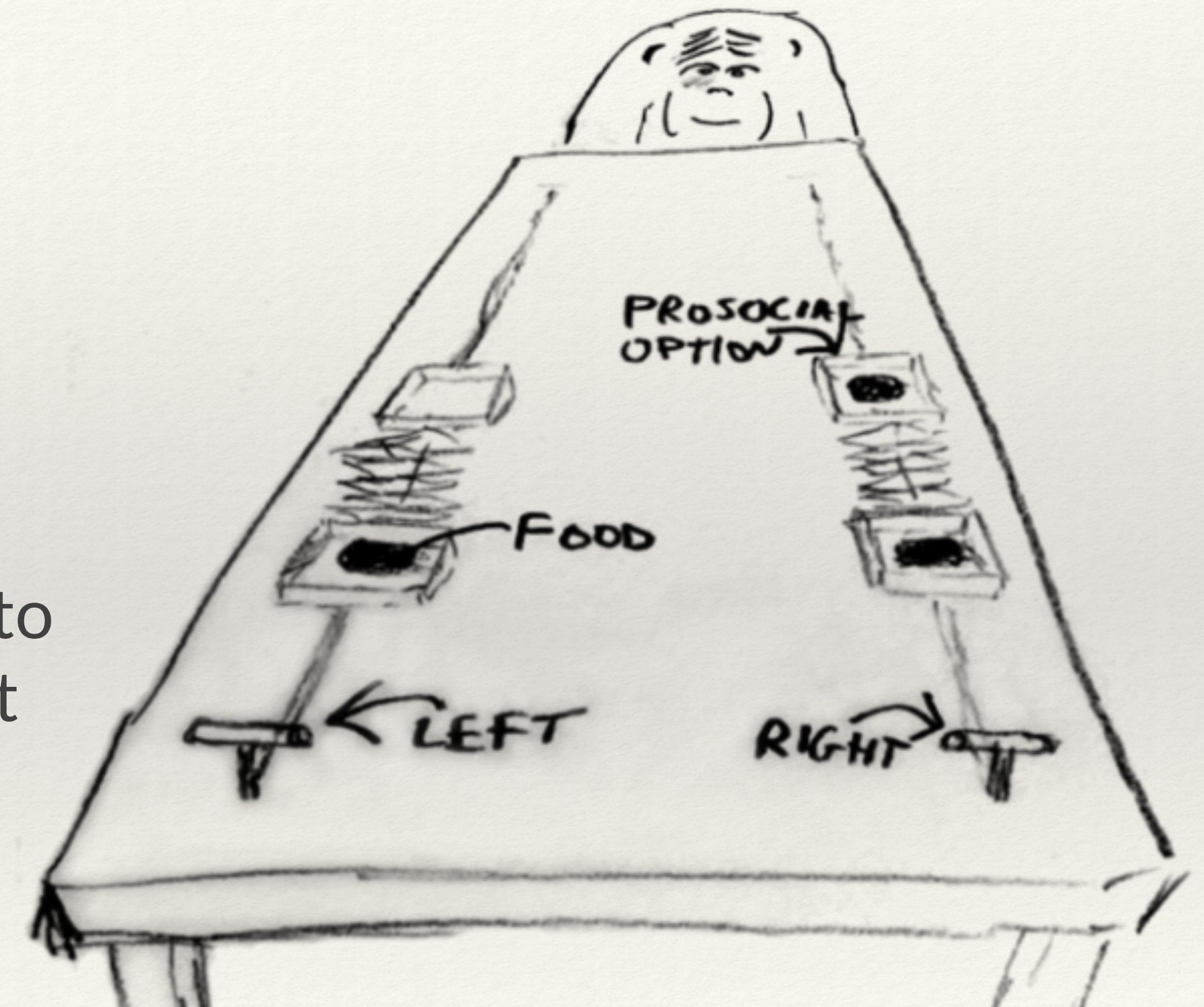
$$\sigma, \sigma_{block} \sim \text{Exponential}(1)$$



BIG EXAMPLE COMING UP!
QUESTIONS?

PROSOCIALITY IN CHIMPS

- Prosociality refers to behaviors that are intended to benefit others.
- We can test for the presence of prosociality experimentally
- The set up allows for a subject to decide to give food to a partner or not, with no cost to himself.



DATA STRUCTURE

- Several measures per individual chimp (actor)
- Condition (social or non-social)
- Experimental blocks
- Prosocial option on the left (chimps are left- or right-handed)

Outcomes:

- Chose prosocial option
- Pulled left

DATA STRUCTURE

- Several measures per individual chimp (actor)

```
> library(rethinking)
> data(chimpanzees)
> head(chimpanzees)
```

	actor	recipient	condition	block	trial	prosoc_left	chose_prosoc	pulled_left	
1	1	NA		0	1	2	0	1	0
2	1	NA		0	1	4	0	0	1
3	1	NA		0	1	6	1	0	0
4	1	NA		0	1	8	0	1	0
5	1	NA		0	1	10	1	1	1
6	1	NA		0	1	12	1	1	1

- Pulled left

EXPERIMENTAL TREATMENTS

- (1) prosoc_left= 0 and condition= 0: Two food items on right and no partner.
- (2) prosoc_left= 1 and condition= 0: Two food items on left and no partner.
- (3) prosoc_left= 0 and condition= 1: Two food items on right and partner present.
- (4) prosoc_left= 1 and condition= 1: Two food items on left and partner present.

EXPERIMENTAL TREATMENTS

- 
- (1) prosoc_left= 0 and condition= 0: Two food items on right and no partner.
 - (2) prosoc_left= 1 and condition= 0: Two food items on left and no partner.
 - (3) prosoc_left= 0 and condition= 1: Two food items on right and partner present.
 - (4) prosoc_left= 1 and condition= 1: Two food items on left and partner present.

EXPERIMENTAL TREATMENTS



- (1) prosoc_left= 0 and condition= 0: Two food items on right and no partner.
- (2) prosoc_left= 1 and condition= 0: Two food items on left and no partner.
- (3) prosoc_left= 0 and condition= 1: Two food items on right and partner present.
- (4) prosoc_left= 1 and condition= 1: Two food items on left and partner present.

EXPERIMENTAL TREATMENTS

Relevant contrasts



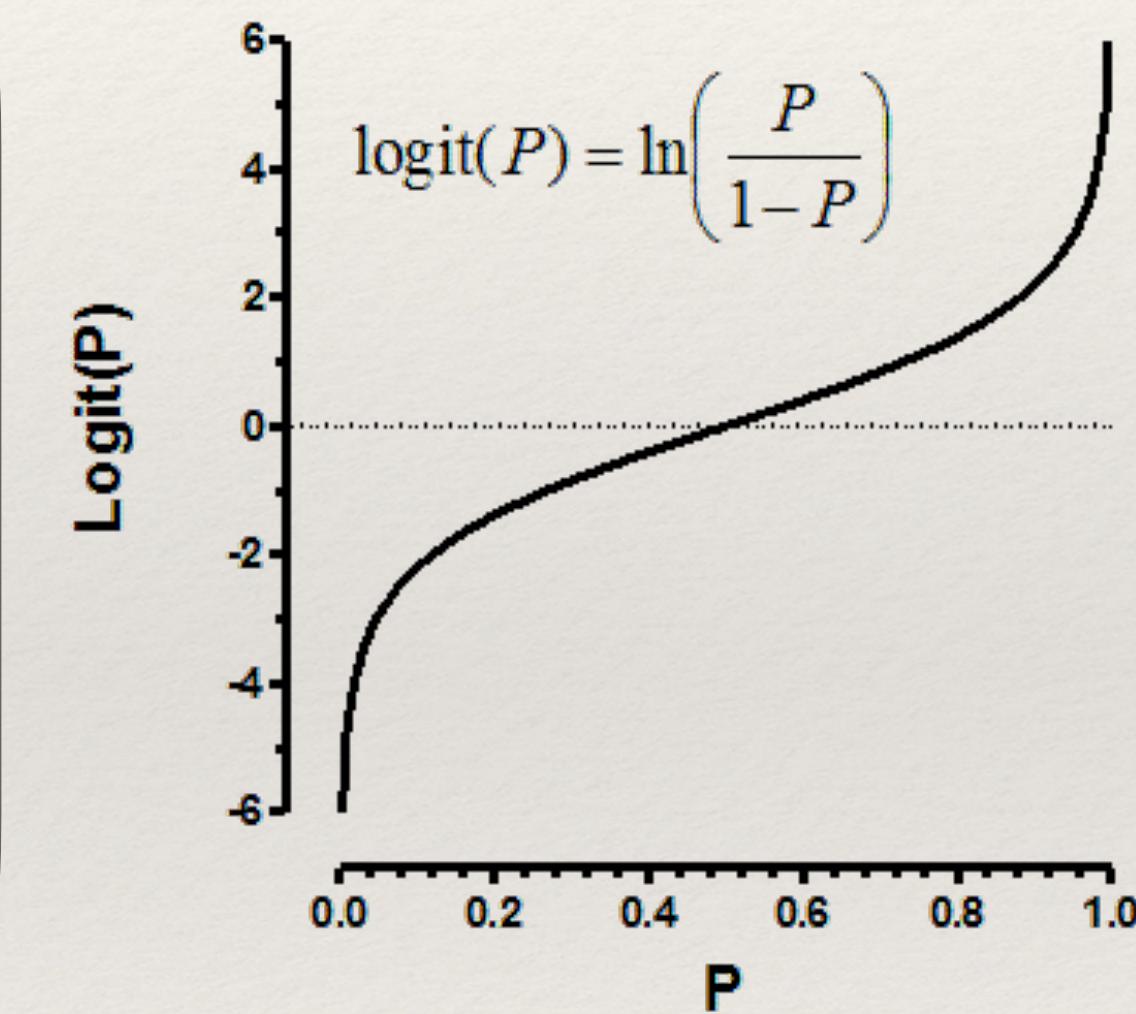
- (1) prosoc_left= 0 and condition= 0: Two food items on right and no partner.
- (2) prosoc_left= 1 and condition= 0: Two food items on left and no partner.
- (3) prosoc_left= 0 and condition= 1: Two food items on right and partner present.
- (4) prosoc_left= 1 and condition= 1: Two food items on left and partner present.

INITIAL MODEL LIKELIHOOD

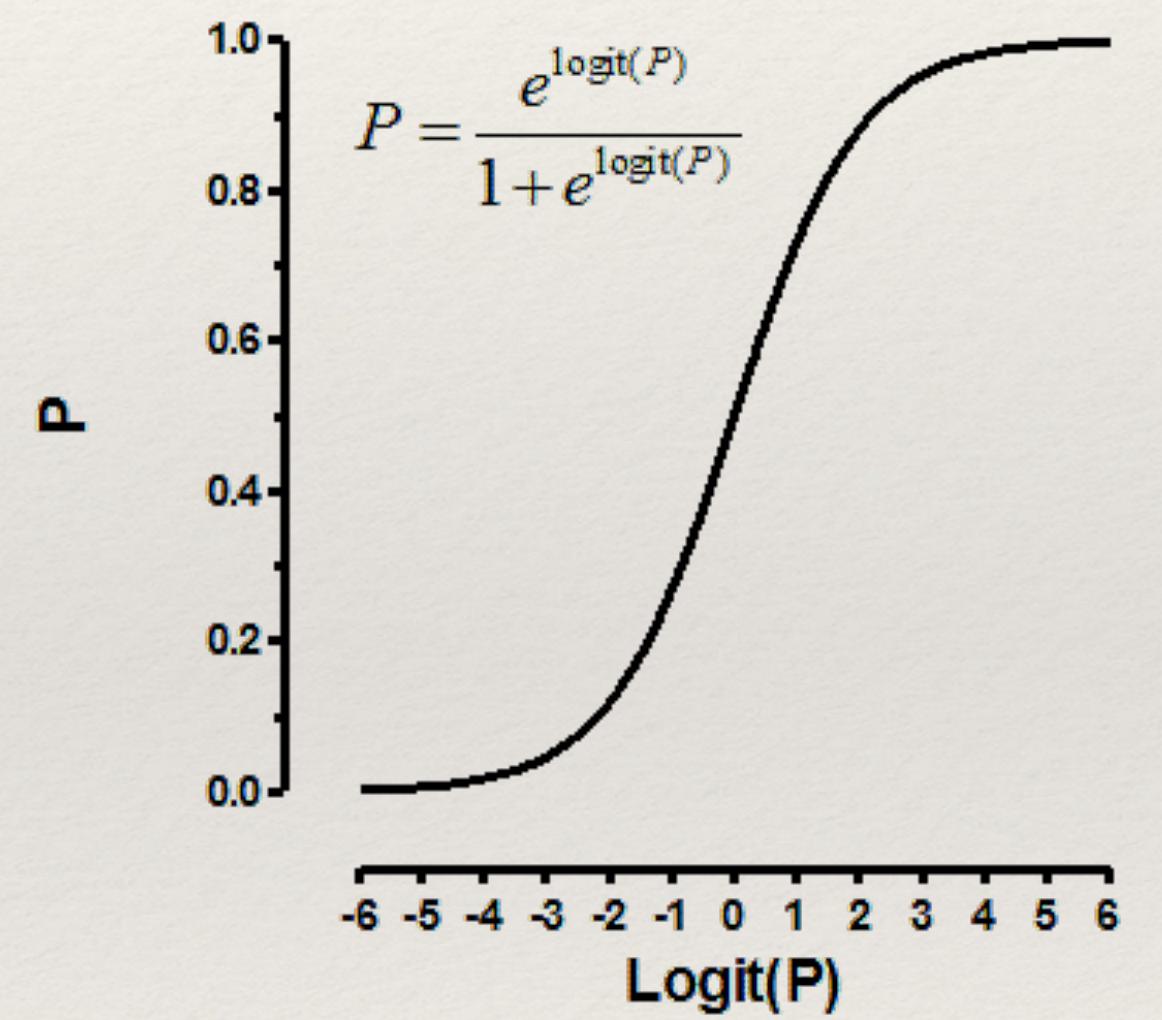
Outcome: $L_i = \text{pulled left}$

$$L_i \sim \text{Bernoulli}(p_i)$$

$$\text{logit}(p_i) = \alpha_{\text{actor}[i]} + \beta_{\text{treatment}[i]}$$



Logit(x) recap



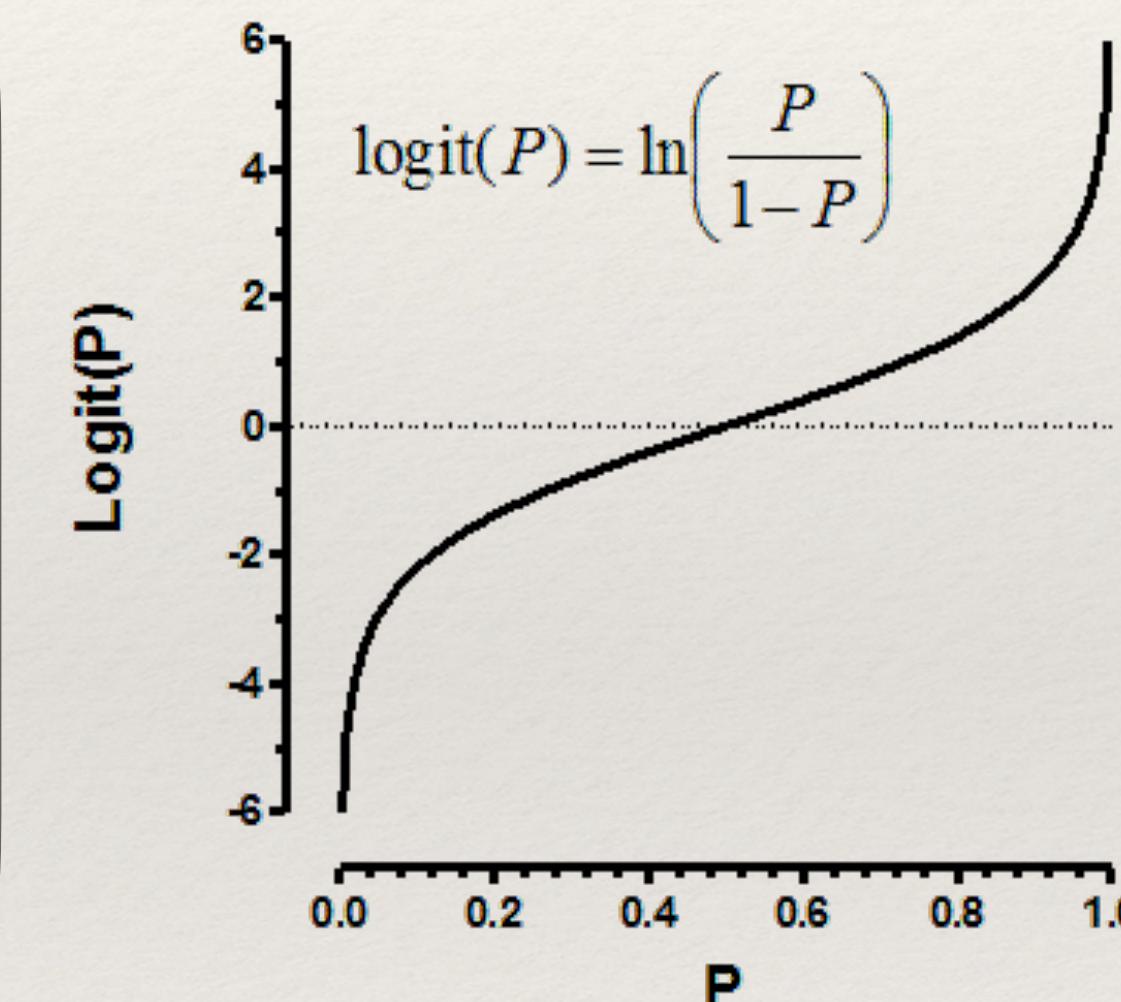
INITIAL MODEL LIKELIHOOD

Outcome: $L_i = \text{pulled left}$

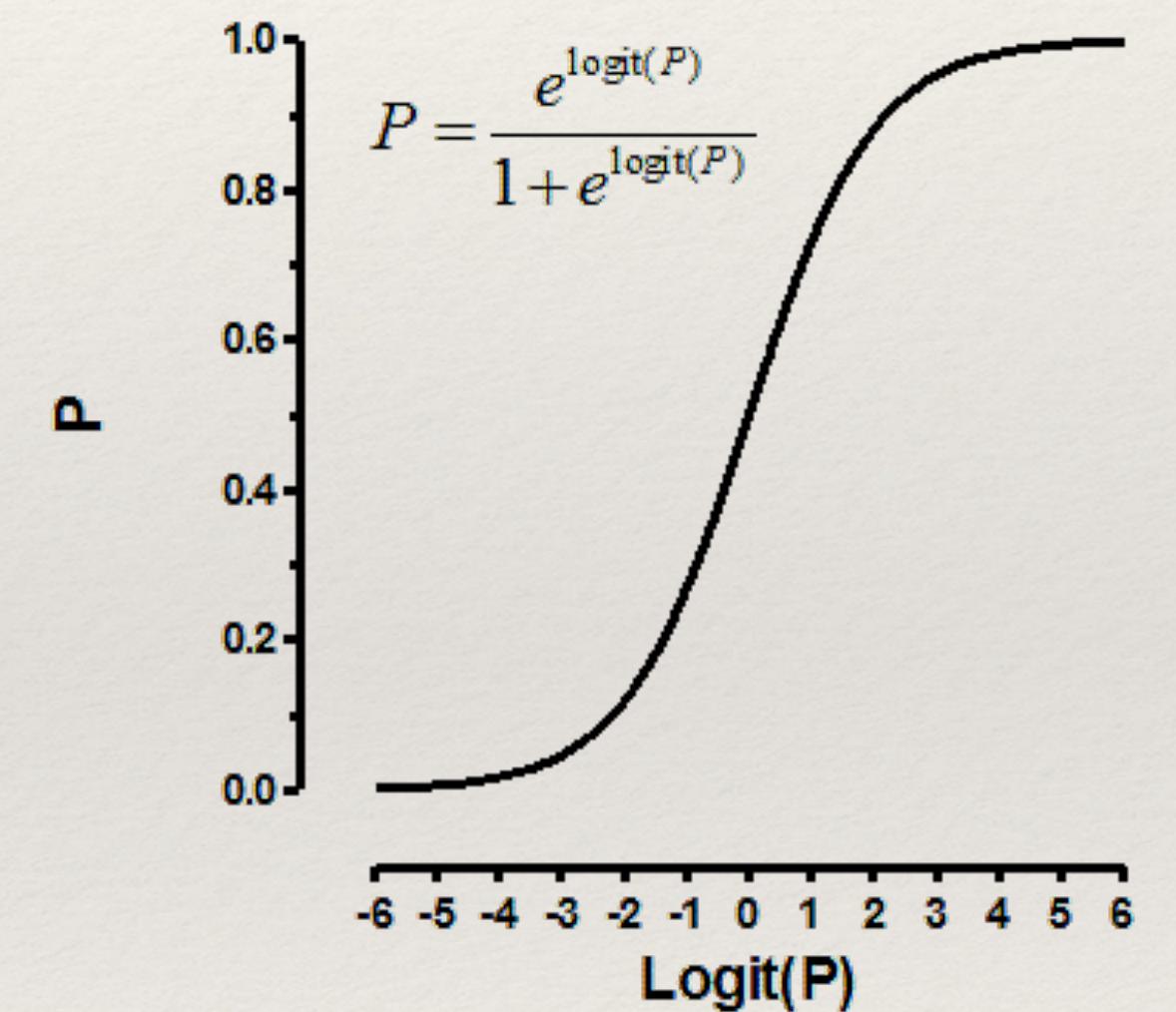
$$L_i \sim \text{Bernoulli}(p_i)$$

$$\text{logit}(p_i) = \alpha_{\text{actor}[i]} + \beta_{\text{treatment}[i]}$$

Accounts for the
actor handedness



Logit(x) recap



INITIAL MODEL LIKELIHOOD

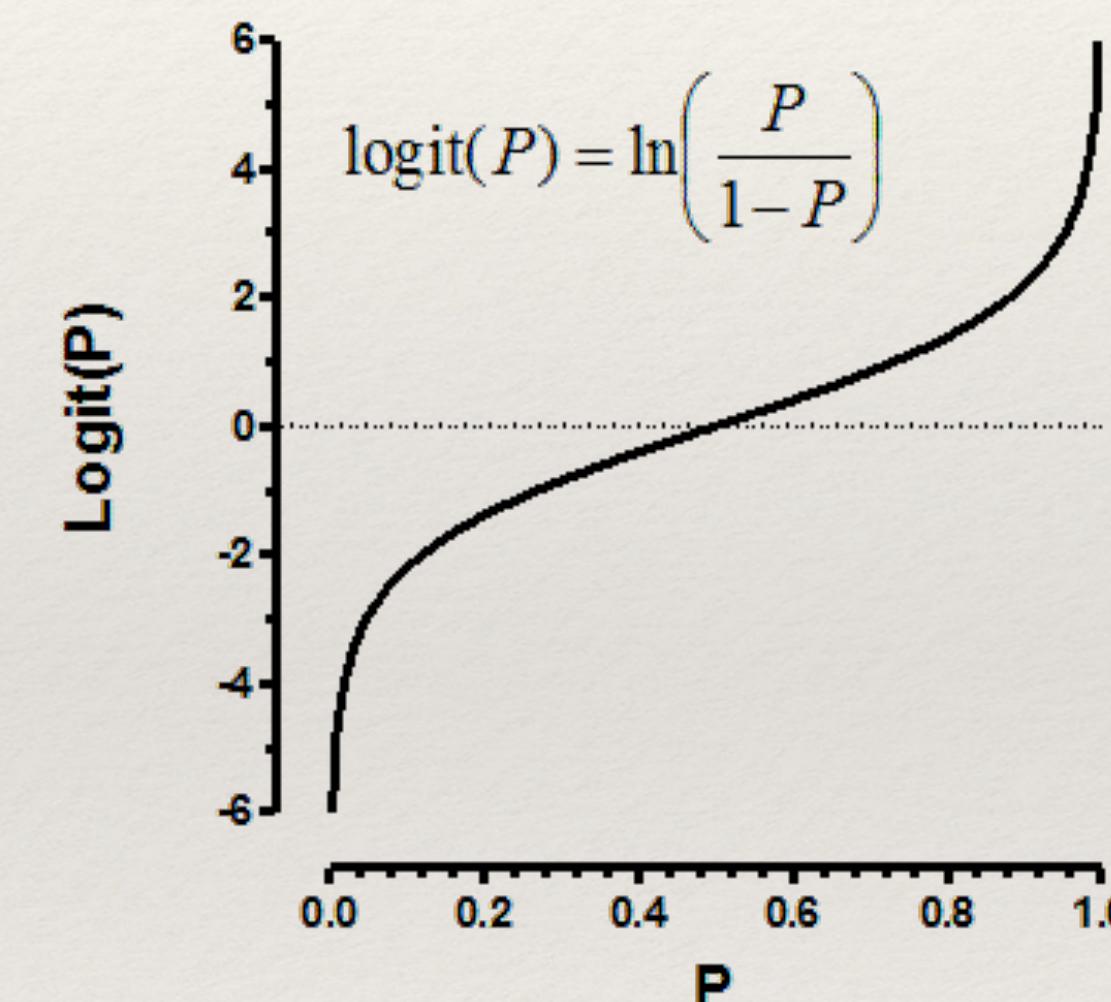
Outcome: $L_i = \text{pulled left}$

$$L_i \sim \text{Bernoulli}(p_i)$$

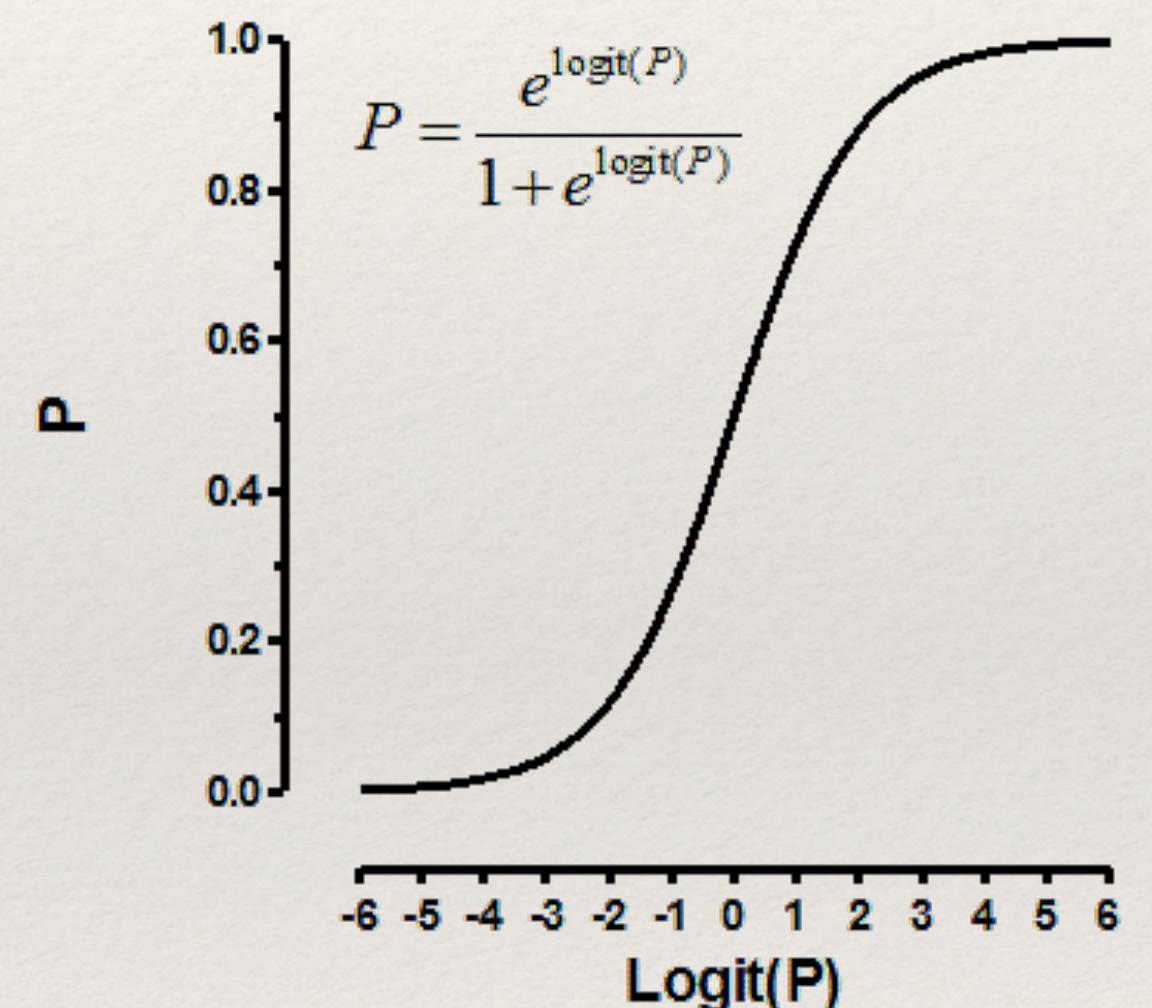
$$\text{logit}(p_i) = \alpha_{\text{actor}[i]} + \beta_{\text{treatment}[i]}$$

Accounts for the
actor handedness

Measures prosocial
treatment effect



Logit(x) recap



MODEL WITH BLOCK EFFECTS

Outcome: L_i = pulled left

$$L_i \sim \text{Bernoulli}(p_i)$$

$$\text{logit}(p_i) = \alpha_{\text{actor}[i]} + \gamma_{\text{block}[i]} + \beta_{\text{treatment}[i]}$$

MODEL WITH BLOCK EFFECTS

Outcome: L_i = pulled left

$$L_i \sim \text{Bernoulli}(p_i)$$

$$\text{logit}(p_i) = \alpha_{actor[i]} + \gamma_{block[i]} + \beta_{treatment[i]}$$

MODEL WITH BLOCK EFFECTS

Outcome: L_i = pulled left

$$L_i \sim \text{Bernoulli}(p_i)$$

$$\text{logit}(p_i) = \alpha_{\text{actor}[i]} + \gamma_{\text{block}[i]} + \beta_{\text{treatment}[i]}$$

Accounts systematic differences across blocks.

Next step is to add priors that introduce dependencies across coefficients.

ADDING REGULARIZING PRIORS FOR BLOCK

Outcome: L_i = pulled left

$$L_i \sim Bernoulli(p_i)$$

$$\text{logit}(p_i) = \alpha_{actor[i]} + \gamma_{block[i]} + \beta_{treatment[i]}$$

$$\gamma_j \sim Normal(0, \sigma_\gamma), \text{ for } j = 1..6$$

ADDING REGULARIZING PRIORS FOR BLOCK

Outcome: L_i = pulled left

$$L_i \sim Bernoulli(p_i)$$

$$\text{logit}(p_i) = \alpha_{actor[i]} + \gamma_{block[i]} + \beta_{treatment[i]}$$

$$\gamma_j \sim Normal(0, \sigma_\gamma), \text{ for } j = 1..6$$

ADDING REGULARIZING PRIORS FOR BLOCK

Outcome: L_i = pulled left

$$L_i \sim Bernoulli(p_i)$$

$$\text{logit}(p_i) = \alpha_{actor[i]} + \gamma_{block[i]} + \beta_{treatment[i]}$$

$$\gamma_j \sim Normal(0, \sigma_\gamma), \text{ for } j = 1..6$$



This estimates how similar the γ_j coefficients are to each other and changes the estimates accordingly

REGULARIZING PRIOR FOR THE ACTOR COEFFICIENT

Outcome: L_i = pulled left

$$L_i \sim Bernoulli(p_i)$$

$$\text{logit}(p_i) = \alpha_{actor[i]} + \gamma_{block[i]} + \beta_{treatment[i]}$$

$$\gamma_j \sim Normal(0, \sigma_\gamma), \text{ for } j = 1..6$$

$$\alpha_j \sim Normal(\alpha_0, \sigma_\alpha), \text{ for } j = 1..7$$

ALL THE MISSING PRIORS TO COMPLETE THE MODEL

$$L_i \sim Bernoulli(p_i)$$

$$\text{logit}(p_i) = \alpha_{\text{actor}[i]} + \gamma_{\text{block}[i]} + \beta_{\text{treatment}[i]}$$

$$\gamma_j \sim Normal(0, \sigma_\gamma), \text{ for } j = 1..6$$

$$\alpha_j \sim Normal(\alpha_0, \sigma_\alpha), \text{ for } j = 1..7$$

$$\beta_j \sim Normal(0, 0.5), \text{ for } j = 1..4$$

$$\alpha_0 \sim Normal(0, 1.5)$$

$$\sigma_\alpha, \sigma_\gamma \sim Exponential(1)$$

PROSOCIAL CHIMP MODEL CODE

```
## Model
m1 <- ulam(
  alist(
    pulled_left ~ binomial( 1 , p ) ,
    logit(p) <- a[actor] + g[block_id] + b[treatment] ,
    b[treatment] ~ normal( 0 , 0.5 ) ,
    ## regularizing multi level priors
    a[actor] ~ normal( a_0 , sigma_a ) ,
    g[block_id] ~ normal( 0 , sigma_g ) ,
    ## hyper-priors
    a_0 ~ normal( 0 , 1.5 ) ,
    sigma_a ~ exponential(1) ,
    sigma_g ~ exponential(1)
  ) , data=dat_list , chains=4 , cores=4 , log_lik=TRUE )
```

PROSOCIAL CHIMP MODEL CODE

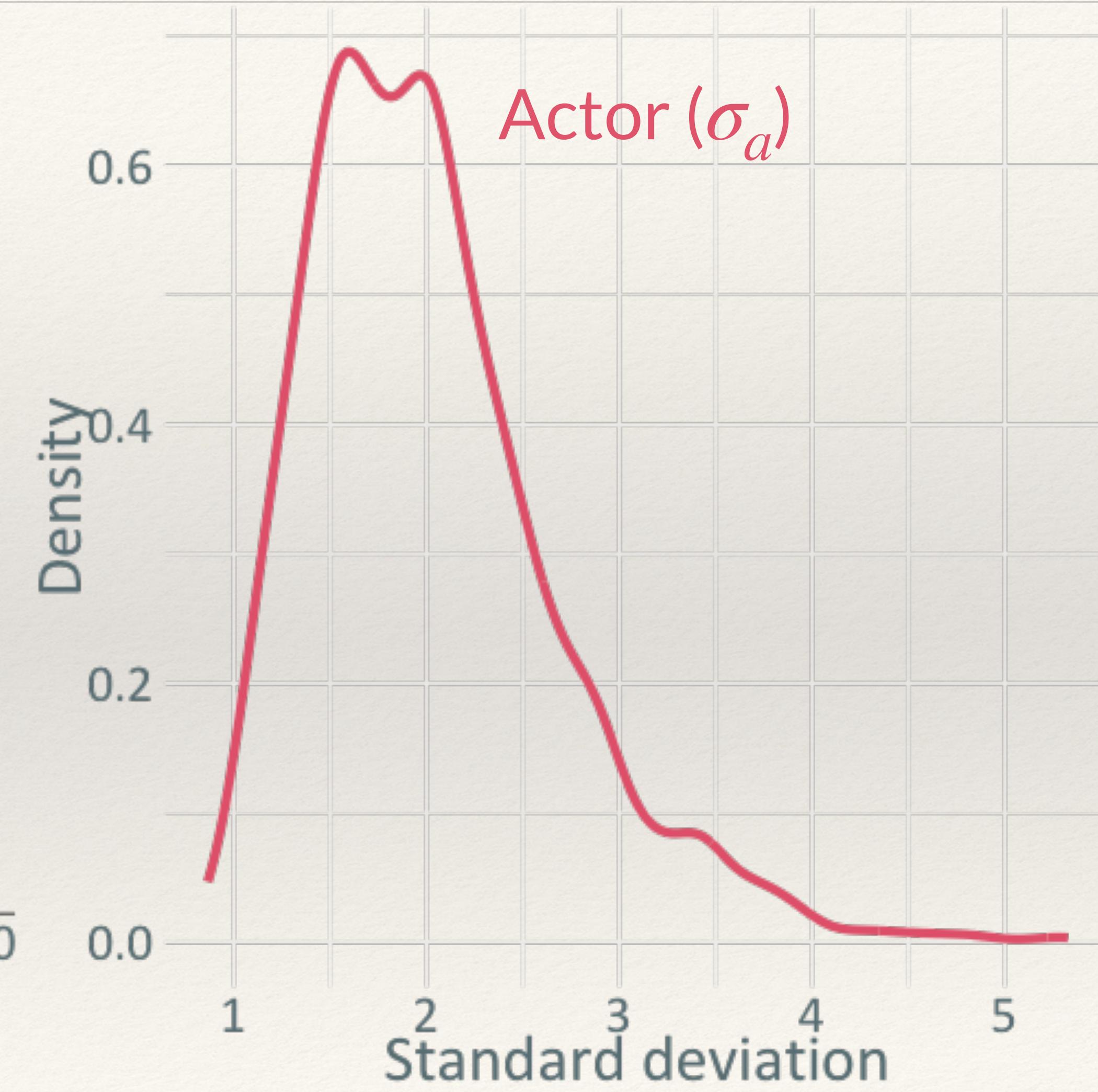
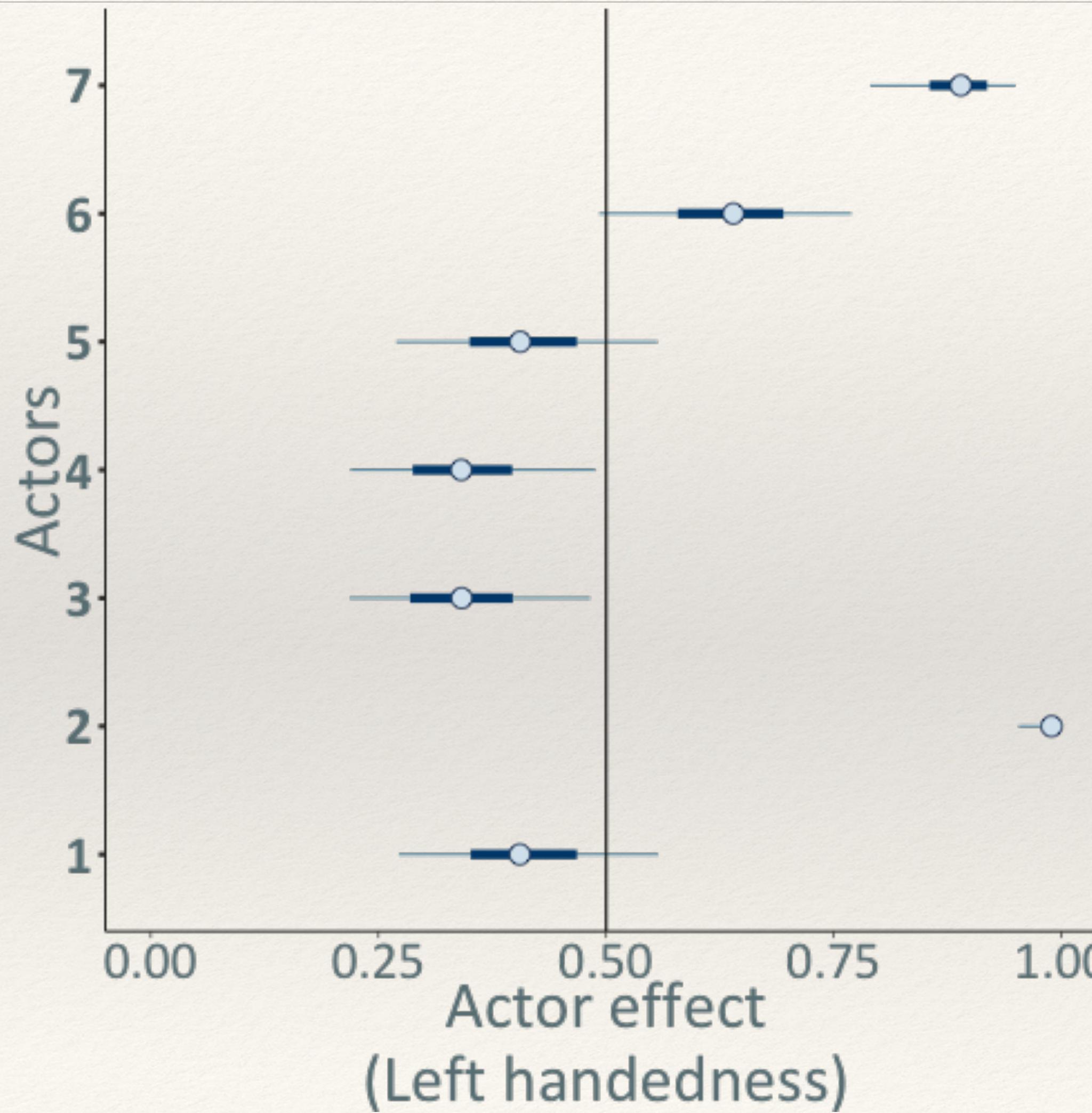
```
## Model          Exercise: add a regularizing prior to the treatment effects
m1 <- ulam(
  alist(
    pulled_left ~ binomial( 1 , p ) ,
    logit(p) <- a[actor] + g[block_id] + b[treatment] ,
    b[treatment] ~ normal( 0 , 0.5 ) ,
    ## regularizing multi level priors
    a[actor] ~ normal( a_0 , sigma_a ) ,
    g[block_id] ~ normal( 0 , sigma_g ) ,
    ## hyper-priors
    a_0 ~ normal( 0 , 1.5 ) ,
    sigma_a ~ exponential(1) ,
    sigma_g ~ exponential(1)
  ) , data=dat_list , chains=4 , cores=4 , log_lik=TRUE )
```

PROSOCIAL CHIMP MODEL CODE

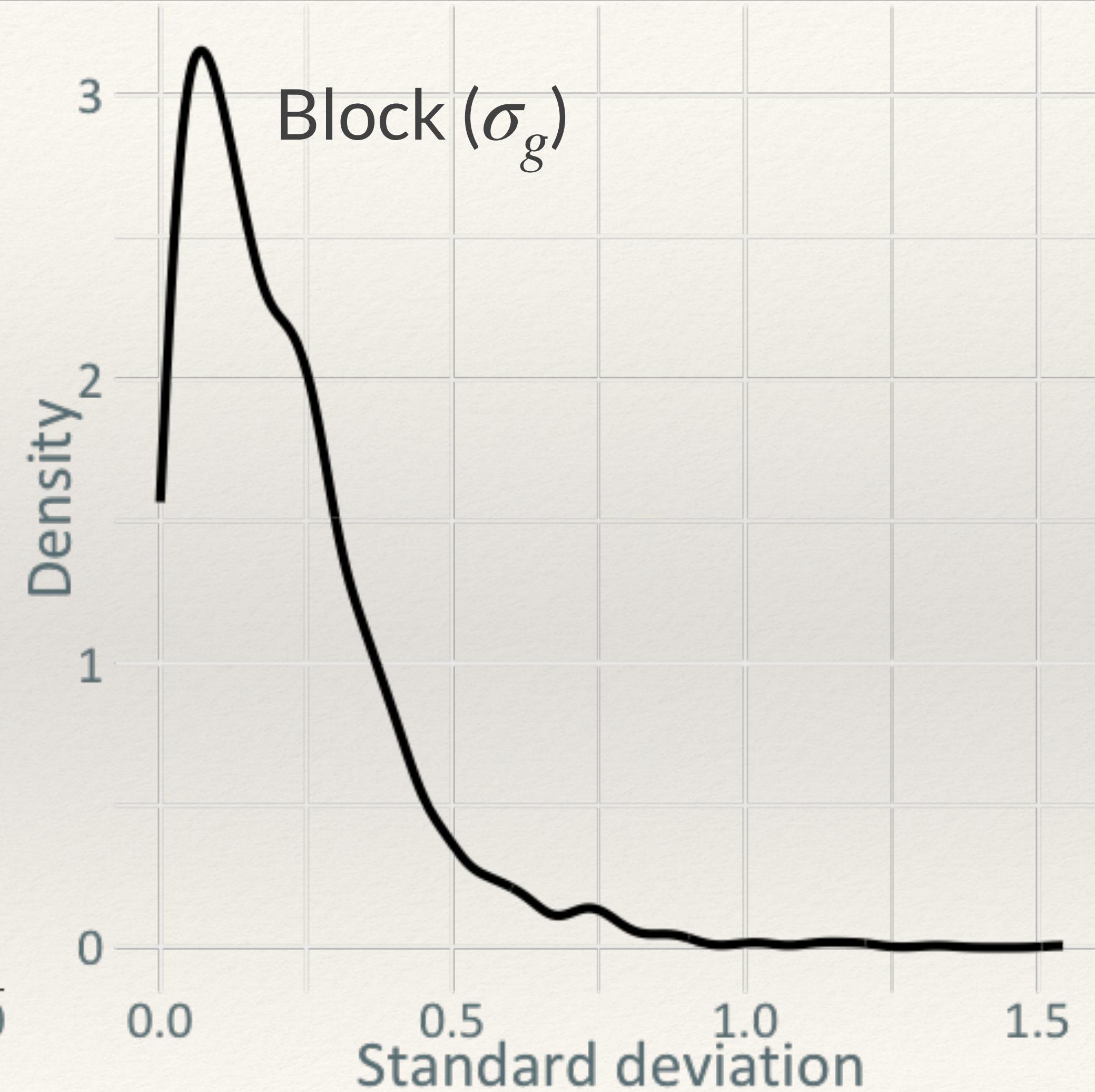
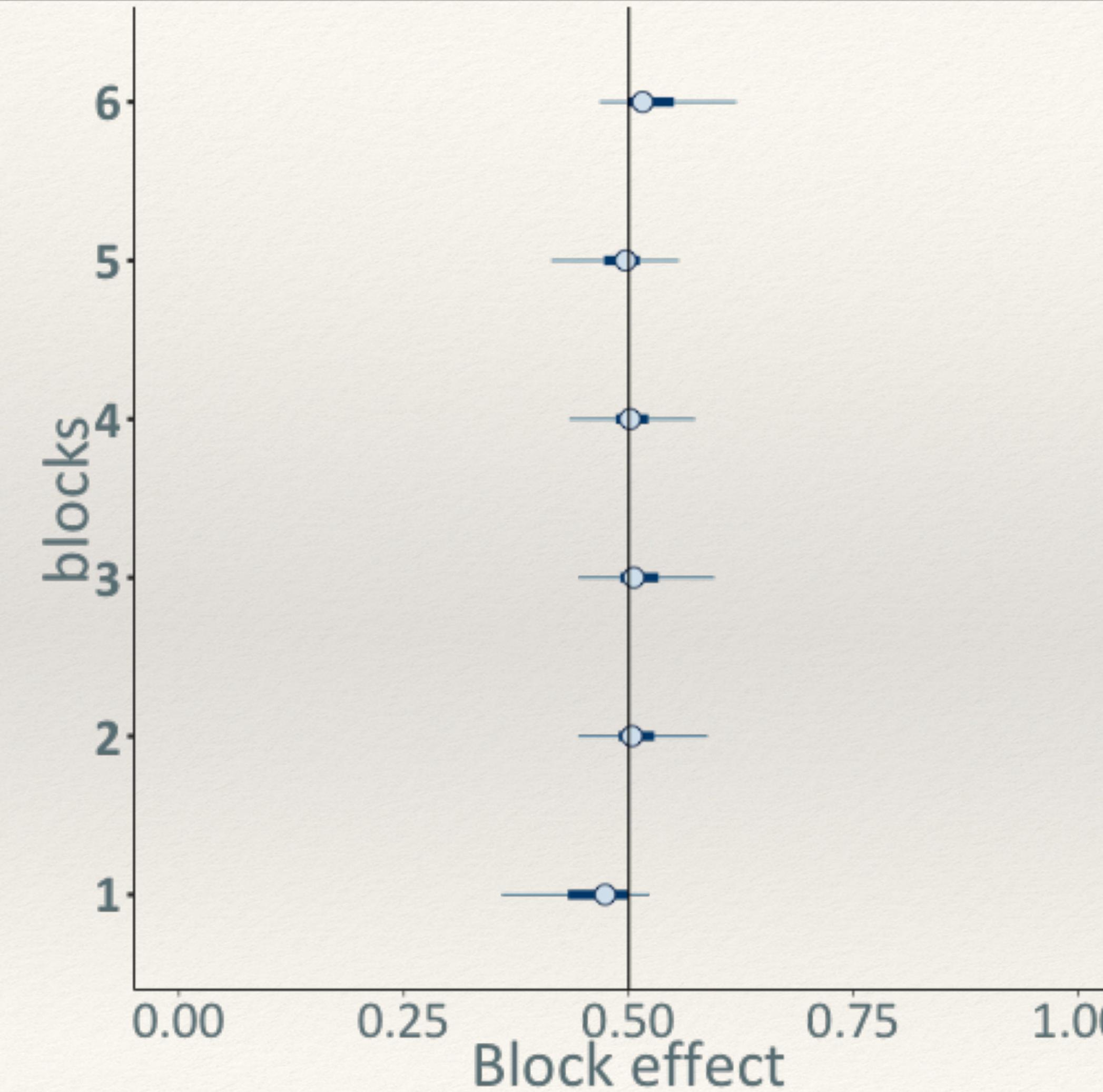
```
## Model          Exercise: add a regularizing prior to the treatment effects
m1 <- ulam(
  alist(
    pulled_left ~ binomial( 1 , p ) ,
    logit(p) <- a[actor] + g[block_id] + b[treatment] ,
    b[treatment] ~ normal( 0 , 0.5 ), ←
    ## regularizing multi level priors
    a[actor] ~ normal( a_0 , sigma_a ) ,
    g[block_id] ~ normal( 0 , sigma_g ) ,
    ## hyper-priors
    a_0 ~ normal( 0 , 1.5 ) ,
    sigma_a ~ exponential(1) ,
    sigma_g ~ exponential(1)
  ) , data=dat_list , chains=4 , cores=4 , log_lik=TRUE )
```

PROSOCIAL CHIMP MODEL RESULTS

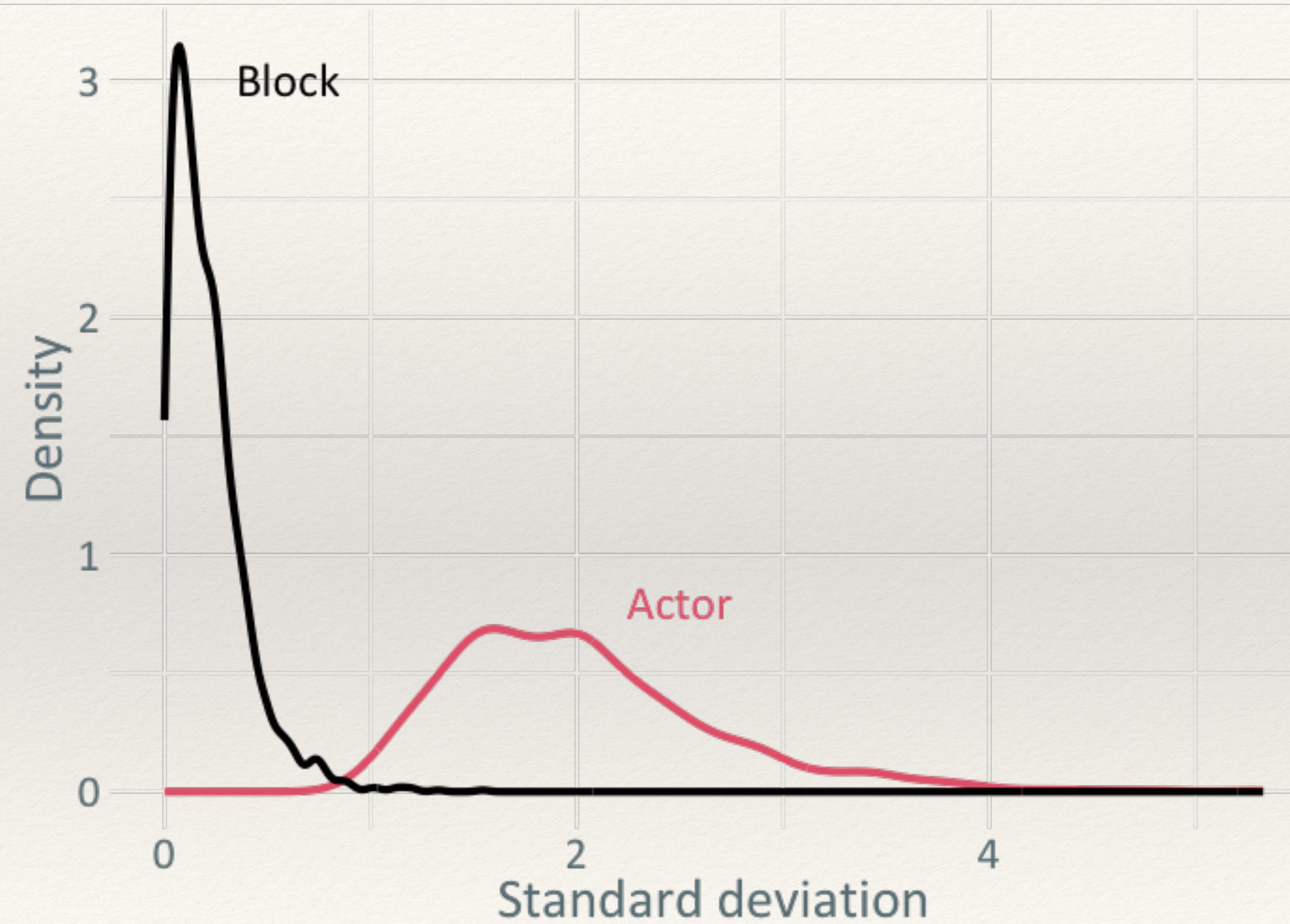
ACTOR ESTIMATES



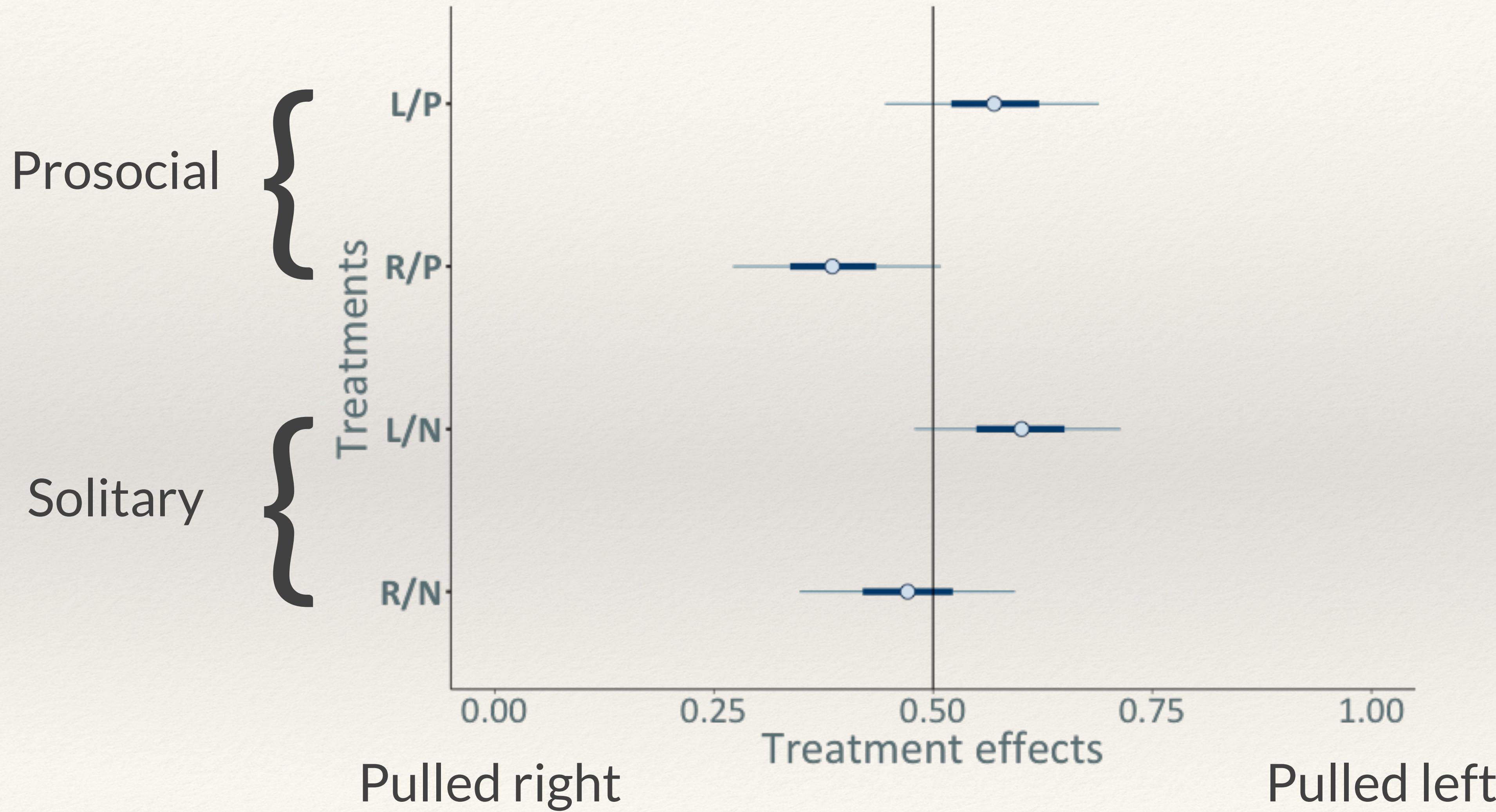
BLOCK ESTIMATES



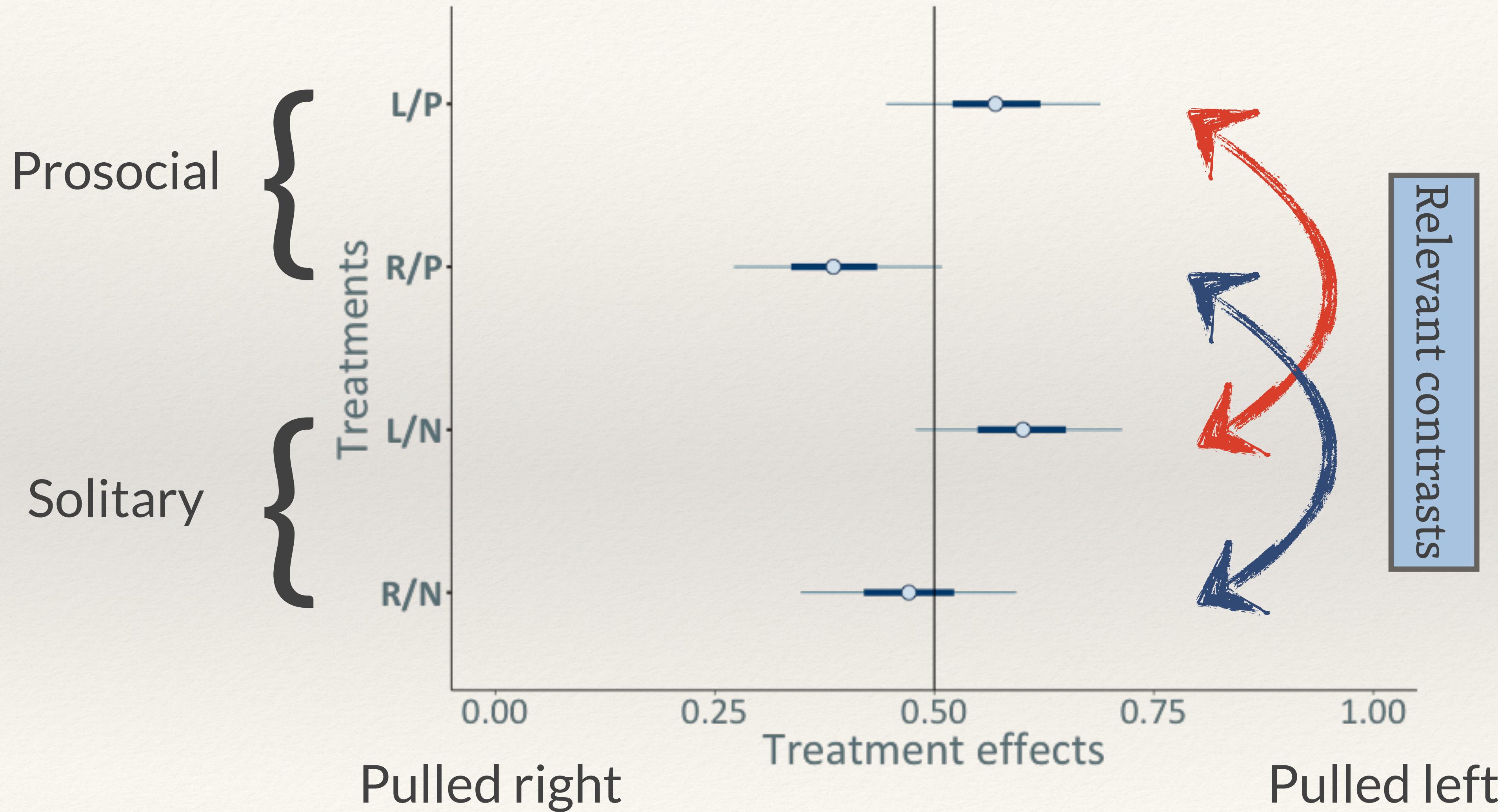
RANDOM EFFECTS STANDARD DEVIATIONS



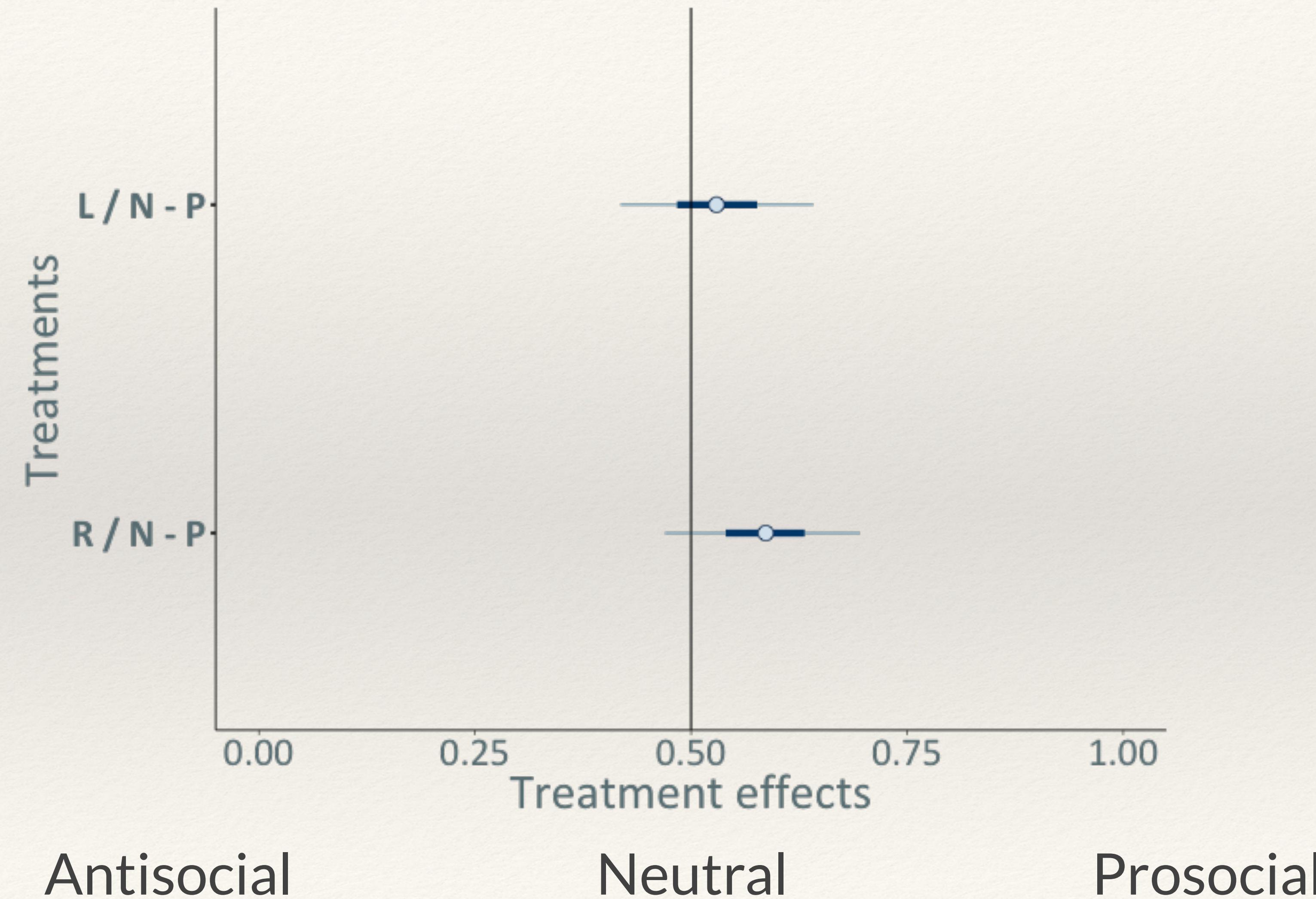
PROSOCIAL OPTION TREATMENT EFFECTS



PROSOCIAL OPTION TREATMENT EFFECTS



TREATMENT CONTRASTS



SOME TECHNICAL DETAILS

CENTERED POOLED EFFECT

The most common way to represent a normal distribution is with the usual mean and sd parameters:

$$x_j \sim N(\mu, \sigma)$$

This is equivalent to:

$$\tilde{x}_j \sim N(0, 1)$$

$$x_j = \mu + \tilde{x}_j \sigma$$

Centered coefficient

$$\text{logit}(p_i) = \alpha_{\text{actor}[i]}$$

$$\alpha_j \sim \text{Normal}(\alpha_0, \sigma_\alpha), \text{ for } j = 1..7$$

$$\alpha_0 \sim \text{Normal}(0, 1.5)$$

$$\sigma_\alpha \sim \text{Exponential}(1)$$

NON-CENTERED POOLED EFFECT

Centered coefficient

$$\text{logit}(p_i) = \alpha_{\text{actor}[i]}$$

$$\alpha_j \sim \text{Normal}(\alpha_0, \sigma_\alpha), \text{ for } j = 1..7$$

$$\alpha_0 \sim \text{Normal}(0, 1.5)$$

$$\sigma_\alpha \sim \text{Exponential}(1)$$

$$x_j \sim N(\mu, \sigma)$$

Non-Centered coefficient

$$\text{logit}(p_i) = \alpha_0 + \tilde{\alpha}_{\text{actor}[i]} * \sigma_\alpha$$

$$\tilde{\alpha}_j \sim \text{Normal}(0, 1), \text{ for } j = 1..7$$

$$\alpha_0 \sim \text{Normal}(0, 1.5)$$

$$\sigma_\alpha \sim \text{Exponential}(1)$$

$$\tilde{x}_j \sim N(0, 1)$$

$$x_j = \mu + \tilde{x}_j \sigma$$

NON-CENTERED POOLED EFFECT

Centered coefficient

$$\text{logit}(p_i) = \alpha_{\text{actor}[i]}$$

$$\alpha_j \sim \text{Normal}(\alpha_0, \sigma_\alpha), \text{ for } j = 1..7$$

$$\alpha_0 \sim \text{Normal}(0, 1.5)$$

$$\sigma_\alpha \sim \text{Exponential}(1)$$

$$x_j \sim N(\mu, \sigma)$$

Non-Centered coefficient

$$\text{logit}(p_i) = \alpha_0 + \tilde{\alpha}_{\text{actor}[i]} * \sigma_\alpha$$

$$\tilde{\alpha}_j \sim \text{Normal}(0, 1), \text{ for } j = 1..7$$

$$\alpha_0 \sim \text{Normal}(0, 1.5)$$

$$\sigma_\alpha \sim \text{Exponential}(1)$$

$$\tilde{x}_j \sim N(0, 1)$$

$$x_j = \mu + \tilde{x}_j \sigma$$

NON-CENTERED POOLED EFFECT

Centered coefficient

$$\text{logit}(p_i) = \alpha_{\text{actor}[i]}$$

$$\alpha_j \sim \text{Normal}(\alpha_0, \sigma_\alpha), \text{ for } j = 1..7$$

$$\alpha_0 \sim \text{Normal}(0, 1.5)$$

$$\sigma_\alpha \sim \text{Exponential}(1)$$

$$x_j \sim N(\mu, \sigma)$$

Non-Centered coefficient

$$\text{logit}(p_i) = \alpha_0 + \tilde{\alpha}_{\text{actor}[i]} * \sigma_\alpha$$

$$\tilde{\alpha}_j \sim \text{Normal}(0, 1), \text{ for } j = 1..7$$

$$\alpha_0 \sim \text{Normal}(0, 1.5)$$

$$\sigma_\alpha \sim \text{Exponential}(1)$$

$$\tilde{x}_j \sim N(0, 1)$$

$$x_j = \mu + \tilde{x}_j \sigma$$

NON-CENTERED POOLED EFFECT

Centered coefficient

$$\text{logit}(p_i) = \alpha_{\text{actor}[i]} + \alpha_j \sim \text{Normal}(\alpha_0, \sigma_\alpha), \text{ for } j = 1..7$$

$$\alpha_0 \sim \text{Normal}(0, 1.5)$$

$$\sigma_\alpha \sim \text{Exponential}(1)$$

$$x_j \sim N(\mu, \sigma)$$

Non-Centered coefficient

$$\text{logit}(p_i) = \alpha_0 + \tilde{\alpha}_{\text{actor}[i]} * \sigma_\alpha$$

$$\tilde{\alpha}_j \sim \text{Normal}(0, 1), \text{ for } j = 1..7$$

$$\alpha_0 \sim \text{Normal}(0, 1.5)$$

$$\sigma_\alpha \sim \text{Exponential}(1)$$

$$\tilde{x}_j \sim N(0, 1)$$

$$x_j = \mu + \tilde{x}_j \sigma$$

NON-CENTERED POOLED EFFECT

Centered coefficient

$$\text{logit}(p_i) = \alpha_{\text{actor}[i]} + \alpha_j \sim \text{Normal}(\alpha_0, \sigma_\alpha), \text{ for } j = 1..7$$

$$\alpha_0 \sim \text{Normal}(0, 1.5)$$

$$\sigma_\alpha \sim \text{Exponential}(1)$$

$$x_j \sim N(\mu, \sigma)$$

Non-Centered coefficient

$$\text{logit}(p_i) = \alpha_0 + \tilde{\alpha}_{\text{actor}[i]} * \sigma_\alpha$$

$$\tilde{\alpha}_j \sim \text{Normal}(0, 1), \text{ for } j = 1..7$$

$$\alpha_0 \sim \text{Normal}(0, 1.5)$$

$$\sigma_\alpha \sim \text{Exponential}(1)$$

$$\tilde{x}_j \sim N(0, 1)$$

$$x_j = \mu + \tilde{x}_j \sigma$$

NON-CENTERED CHIMP MODEL

```
m1nc <- ulam(  
  alist(  
    pulled_left ~ binomial( 1 , p ) ,  
    logit(p) <- a_0 + z[actor]*sigma_a +  
      x[block_id]*sigma_g +  
      b[treatment] ,  
    b[treatment] ~ normal( 0 , 0.5 ) ,  
    z[actor] ~ normal( 0 , 1 ) ,  
    x[block_id] ~ normal( 0 , 1 ) ,  
    a_0 ~ normal( 0 , 1.5 ) ,  
    sigma_a ~ exponential(1) ,  
    sigma_g ~ exponential(1) ,  
    gq> vector[actor]:a <<- a_0 + z*sigma_a , # actor intercepts  
    gq> vector[block_id]:g <<- x*sigma_g          # block intercepts  
  ) , data=dat_list , chains=4 , cores=4 )
```

OTHER MIXED MODEL TYPES

RANDOM SLOPES MODEL

$$y_i \sim Normal(\mu_i, \sigma)$$

$$\mu_i = \alpha_0 + \alpha_{block[i]} + (\beta_0 + \beta_{block[i]})x_i$$

$$\alpha_k \sim Normal(0, \sigma_\alpha), \text{ for } k \in \{1, \dots, N_{blocks}\}$$

$$\beta_k \sim Normal(0, \sigma_\beta), \text{ for } k \in \{1, \dots, N_{blocks}\}$$

$$\alpha_0, \beta_0 \sim Normal(0, 1)$$

$$\beta \sim Normal(0, 0.3)$$

$$\sigma, \sigma_{block} \sim Exponential(1)$$

RANDOM SLOPES MODEL

$$y_i \sim \text{Normal}(\mu_i, \sigma)$$

$$\mu_i = \alpha_0 + \alpha_{block[i]} + (\beta_0 + \beta_{block[i]})x_i$$

$$\alpha_k \sim \text{Normal}(0, \sigma_\alpha), \text{ for } k \in \{1, \dots, N_{blocks}\}$$

$$\beta_k \sim \text{Normal}(0, \sigma_\beta), \text{ for } k \in \{1, \dots, N_{blocks}\}$$

$$\alpha_0, \beta_0 \sim \text{Normal}(0, 1)$$

$$\beta \sim \text{Normal}(0, 0.3)$$

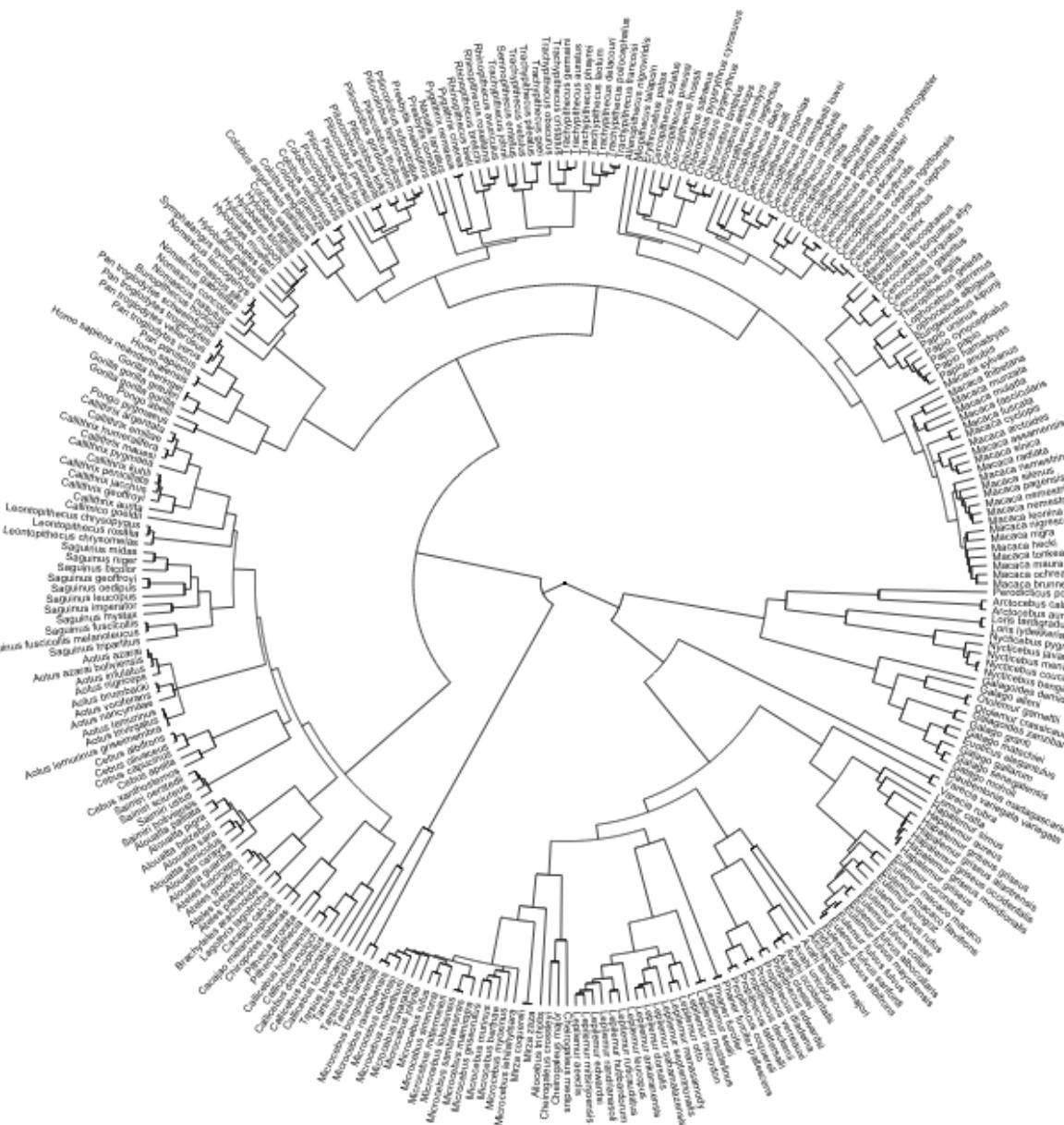
$$\sigma, \sigma_{block} \sim \text{Exponential}(1)$$

Allows the slope associated with the x predictor to vary across blocks, but maintains a dependency across blocks

MORE COMPLEX STRUCTURING OF THE DEPENDENCY ACROSS COEFFICIENTS

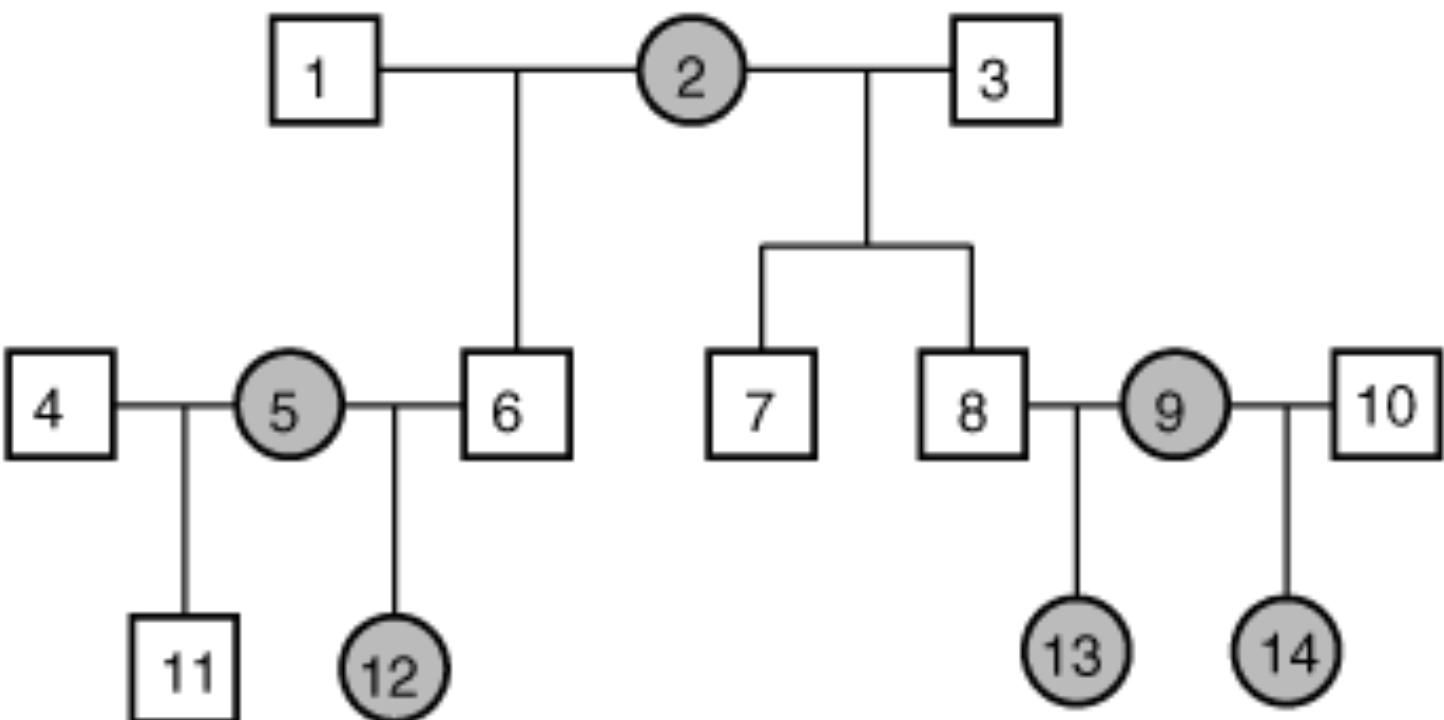
Phylogenetic models

Evolutionary relatedness



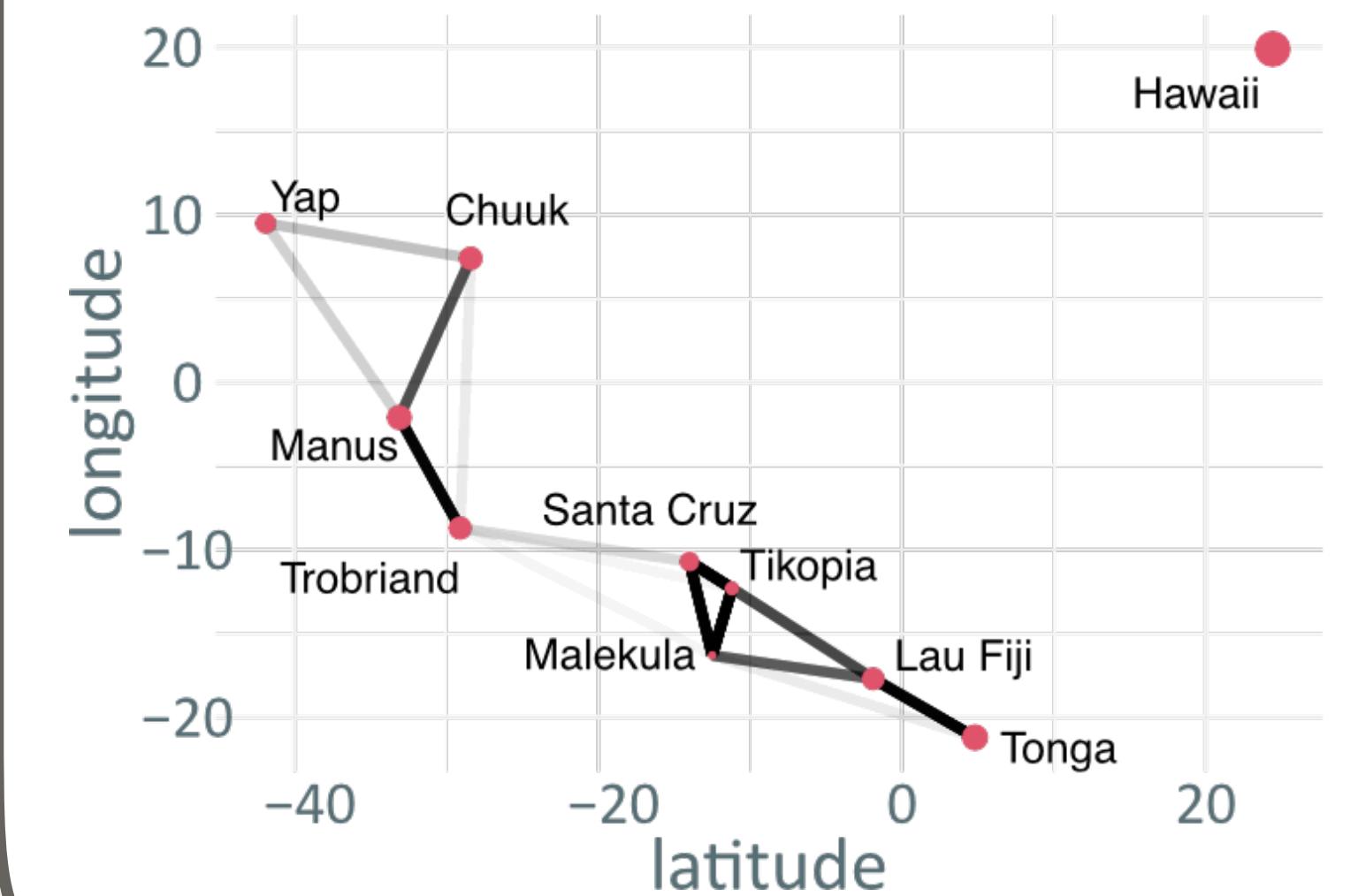
Animal models

Genetic relatedness

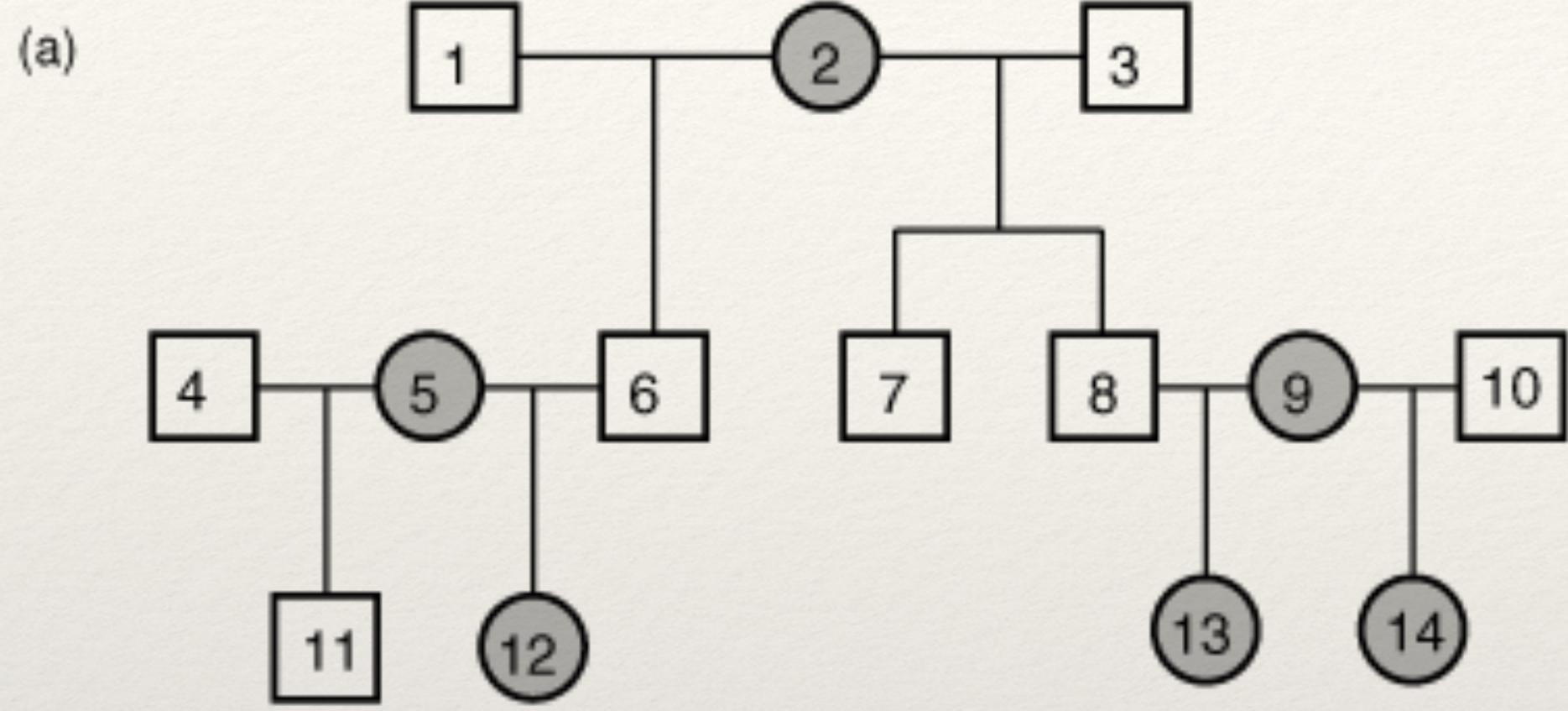


Spacial auto-correlation

Spacial proximity



EXAMPLE: ANIMAL MODEL



(b)

$\mathbf{A} =$

1	0	0	0	0	0.5	0	0	0	0	0.25	0	0	0
0	1	0	0	0	0.5	0.5	0.5	0	0	0	0.25	0.25	0
0	0	1	0	0	0	0.5	0.5	0	0	0	0	0.25	0
0	0	0	1	0	0	0	0	0	0	0.5	0	0	0
0	0	0	0	1	0	0	0	0	0	0.5	0.5	0	0
0.5	0.5	0	0	0	1	0.25	0.25	0	0	0	0.5	0.125	0
0	0.5	0.5	0	0	0.25	1	0.5	0	0	0	0.125	0.25	0
0	0.5	0.5	0	0	0.25	0.5	1	0	0	0	0.125	0.5	0
0	0	0	0	0	0	0	0	1	0	0	0	0.5	0.5
0	0	0	0	0	0	0	0	0	1	0	0	0	0.5
0	0	0	0.5	0.5	0	0	0	0	0	1	0.25	0	0
0.25	0.25	0	0	0.5	0.5	0.125	0.125	0	0	0.25	1	0.0625	0
0	0.25	0.25	0	0	0.125	0.25	0.5	0.5	0	0	0.0625	1	0.25
0	0	0	0	0	0	0	0	0.5	0.5	0	0	0.25	1

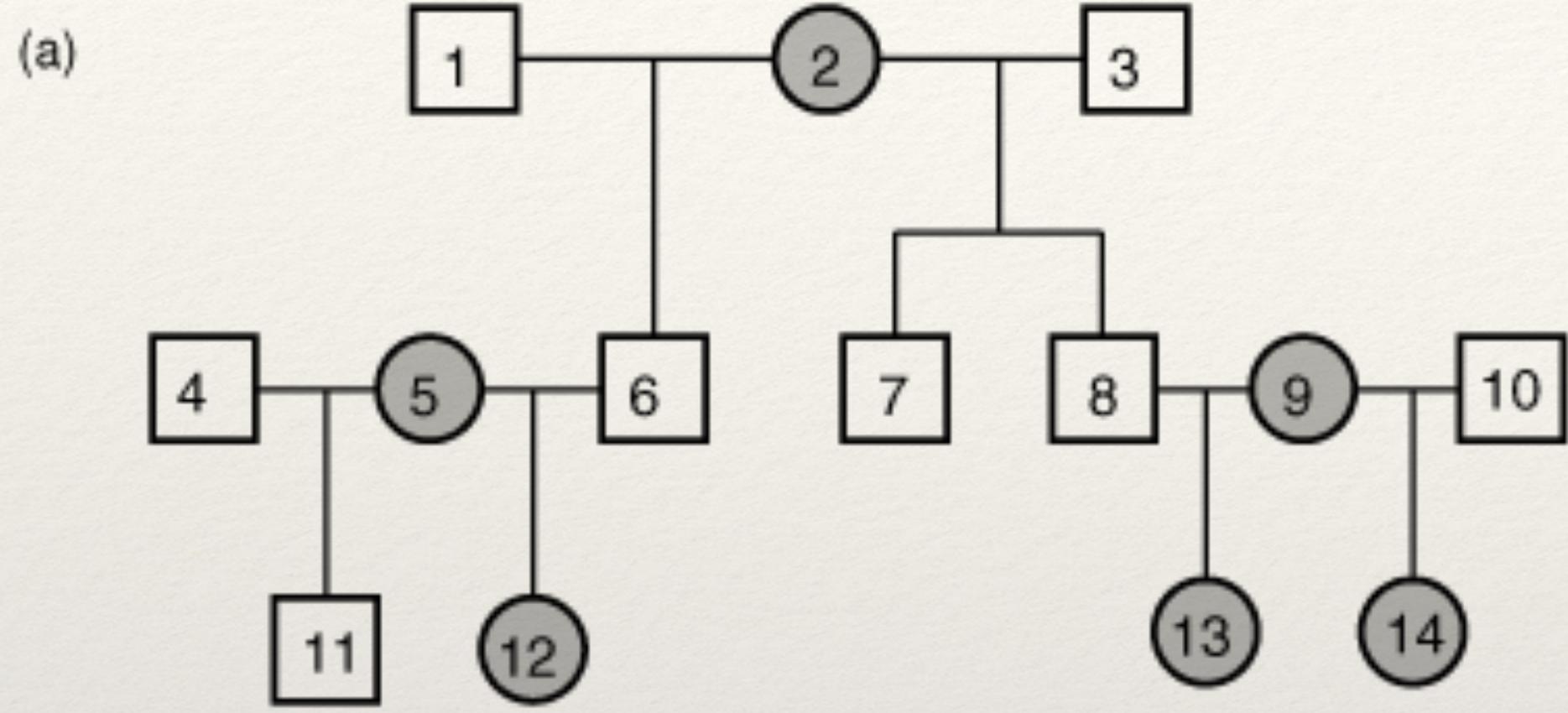
$$y_i \sim \text{Normal}(\mu_i, \sigma_R)$$

$$\mu_i = \alpha_0 + \alpha_i$$

$$\alpha \sim \text{MVNormal}(0, \sigma_G \times A)$$

$$\sigma_G, \sigma_R \sim \text{Exponential}(1)$$

EXAMPLE: ANIMAL MODEL



(b)

$\mathbf{A} =$

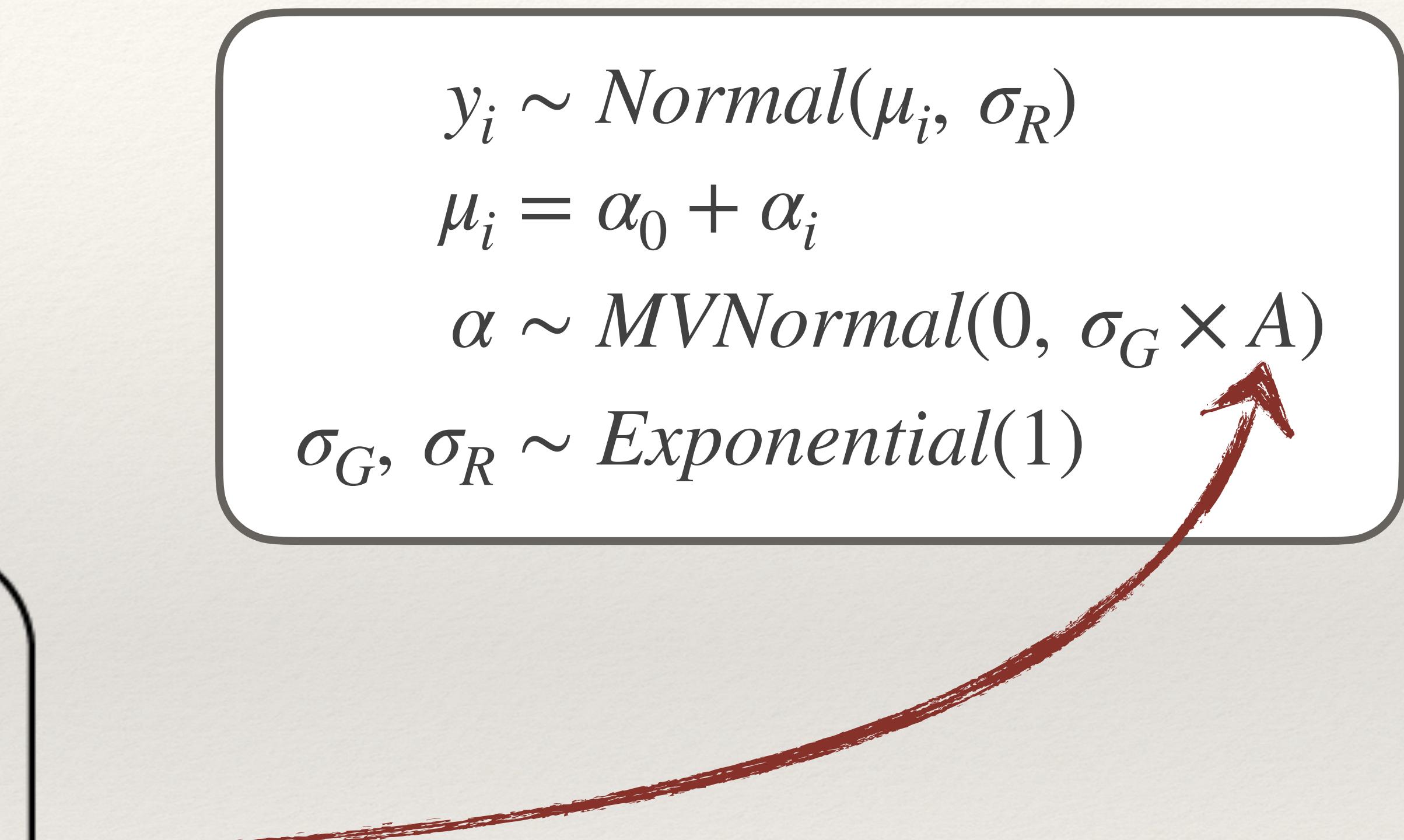
1	0	0	0	0	0.5	0	0	0	0	0.25	0	0
0	1	0	0	0	0.5	0.5	0.5	0	0	0.25	0.25	0
0	0	1	0	0	0	0.5	0.5	0	0	0	0.25	0
0	0	0	1	0	0	0	0	0	0.5	0	0	0
0	0	0	0	1	0	0	0	0	0.5	0.5	0	0
0.5	0.5	0	0	0	1	0.25	0.25	0	0	0.5	0.125	0
0	0.5	0.5	0	0	0.25	1	0.5	0	0	0.125	0.25	0
0	0.5	0.5	0	0	0.25	0.5	1	0	0	0.125	0.5	0
0	0	0	0	0	0	0	0	1	0	0	0.5	0.5
0	0	0	0	0	0	0	0	0	1	0	0	0.5
0	0	0	0.5	0.5	0	0	0	0	1	0.25	0	0
0.25	0.25	0	0	0.5	0.5	0.125	0.125	0	0	0.25	1	0.0625
0	0.25	0.25	0	0	0.125	0.25	0.5	0.5	0	0.0625	1	0.25
0	0	0	0	0	0	0	0	0.5	0.5	0	0.25	1

$$y_i \sim \text{Normal}(\mu_i, \sigma_R)$$

$$\mu_i = \alpha_0 + \alpha_i$$

$$\alpha \sim \text{MVNormal}(0, \sigma_G \times A)$$

$$\sigma_G, \sigma_R \sim \text{Exponential}(1)$$



Same thing for the phylogeny or
the spacial auto-correlation