Statistical Rethinking

2023

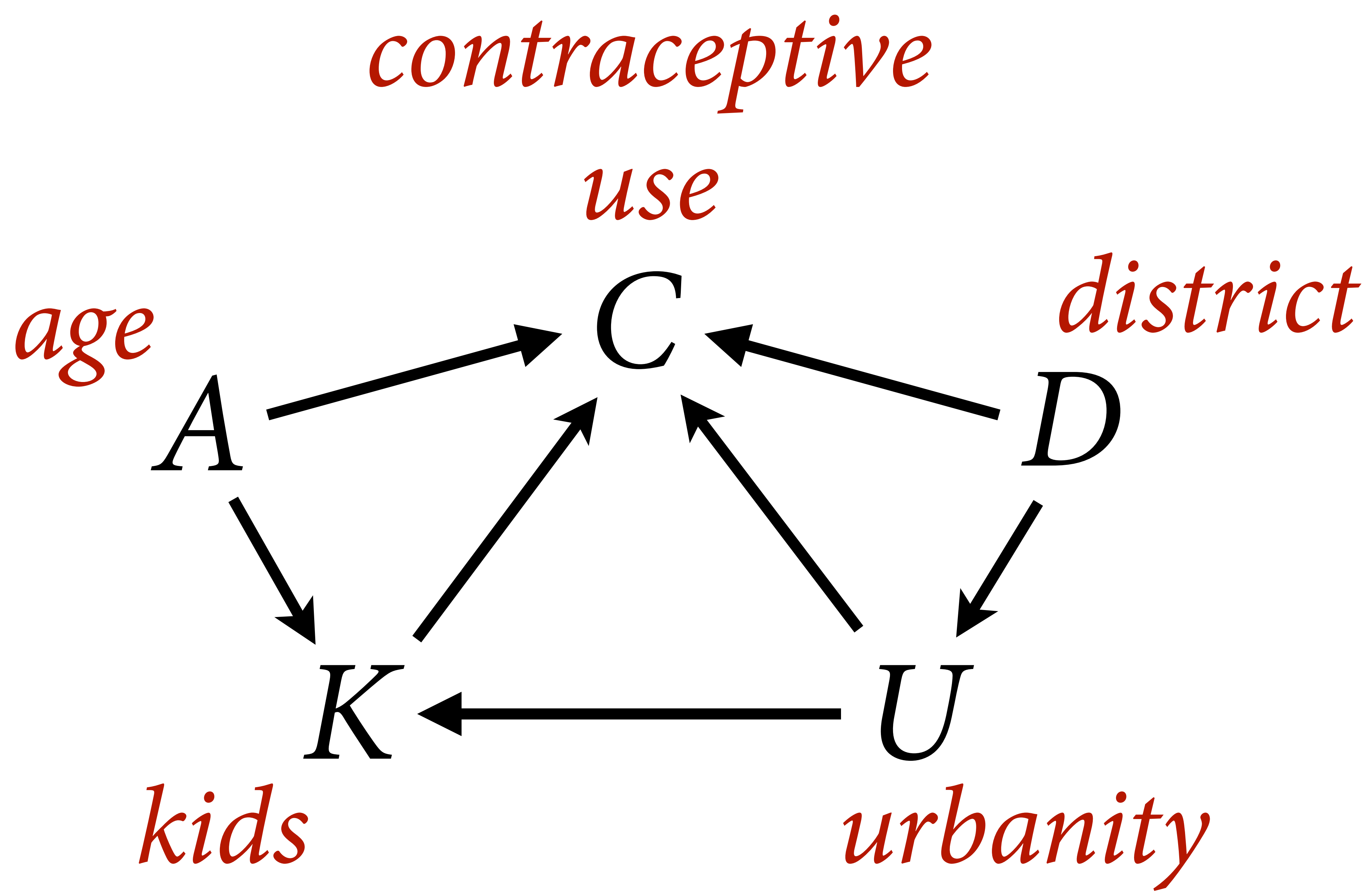
14. Correlated Features

Bangladesh 1989, Party Like It's

Estimand 1: C in each district

Estimand 2: Effect of U

Estimand 3: Effects of K and A

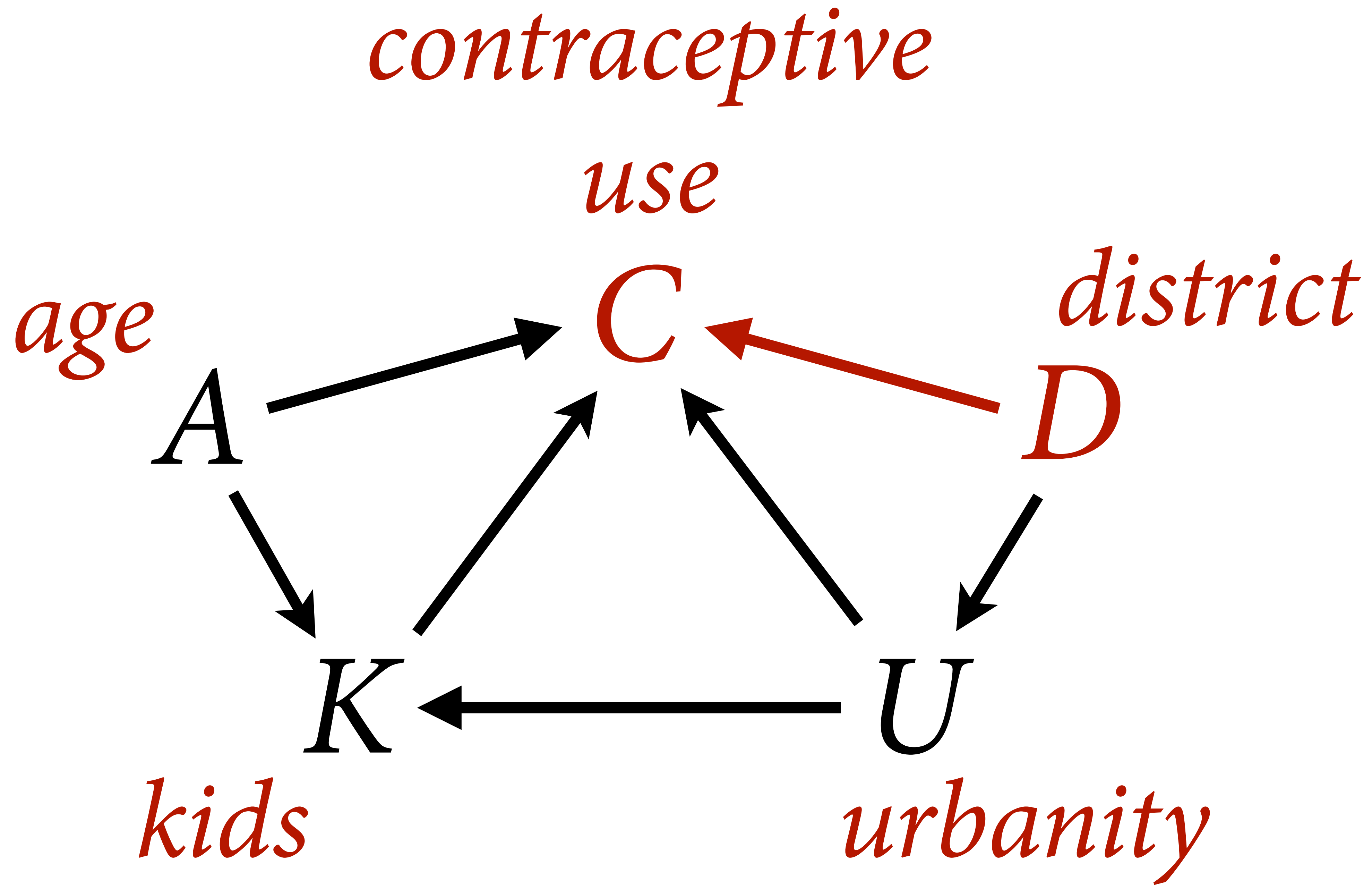


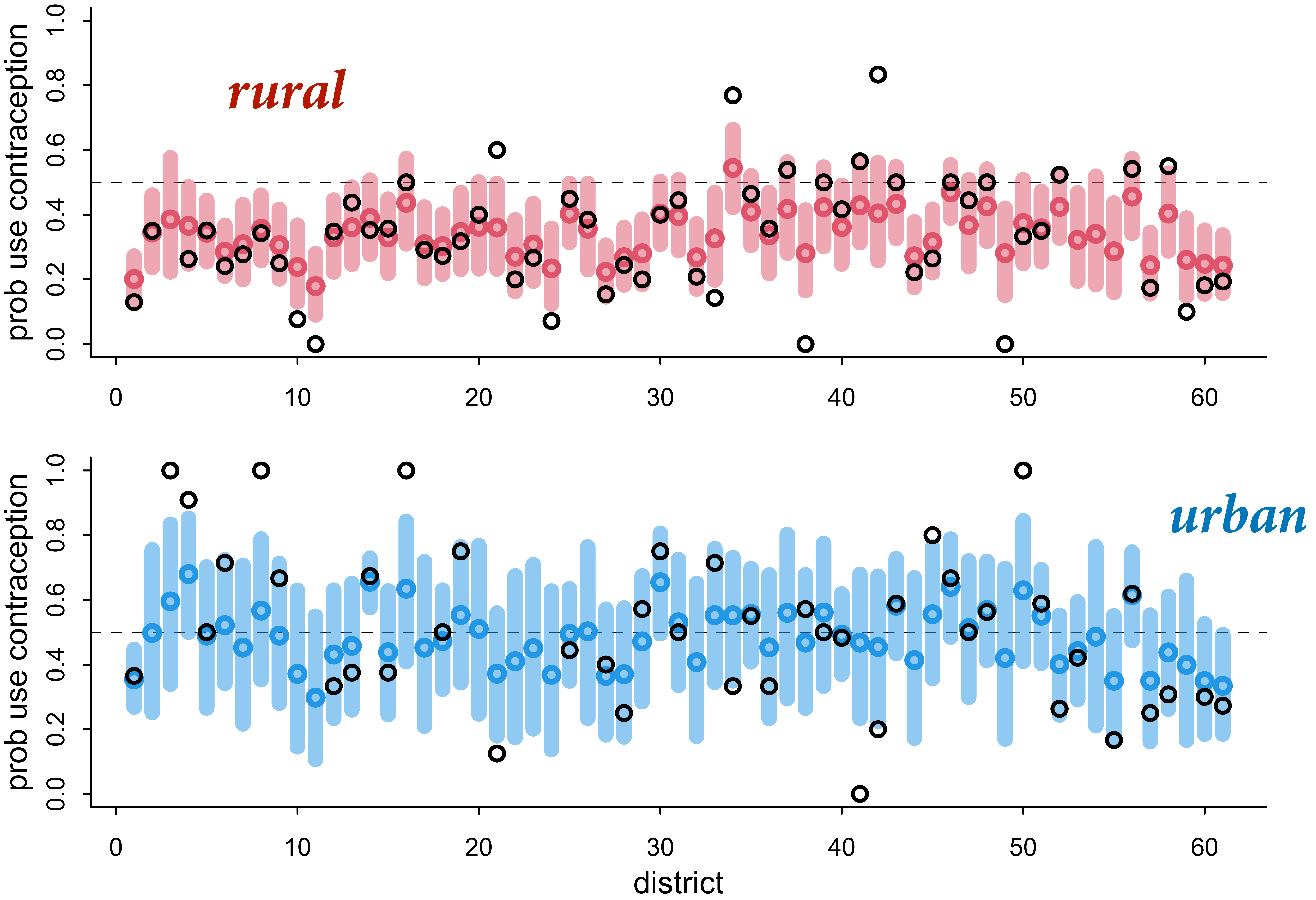
Bangladesh 1989, Party Like It's

Estimand 1: C in each district

Estimand 2: Effect of U

Estimand 3: Effects of K and A





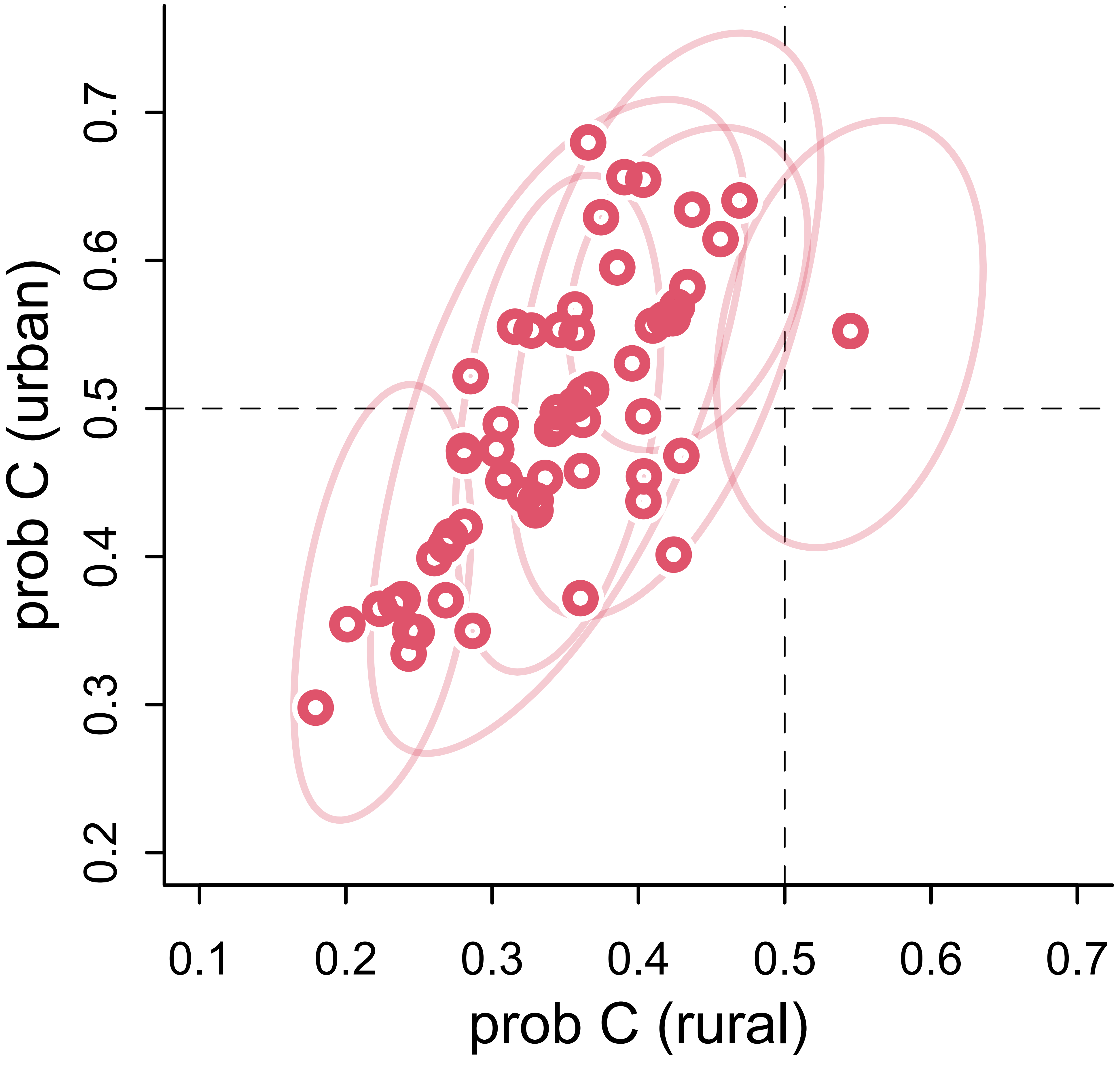
rural
urban
prob use contraception
district
0.0
0.2
0.4
0.6
0.8
1.0
0
10
20
30
40
50
60

Multilevel Adventures

Clusters: Kinds of groups in the data (districts)

Features: Aspects of the model (parameters) that vary by cluster (rural, urban)

There is useful information to transfer across features



Adding Correlated Features

One prior distribution for each cluster

One feature: One-dimensional distribution

$$\alpha_j \sim \text{Normal}(\bar{\alpha}, \sigma)$$

Two features: Two-D distribution

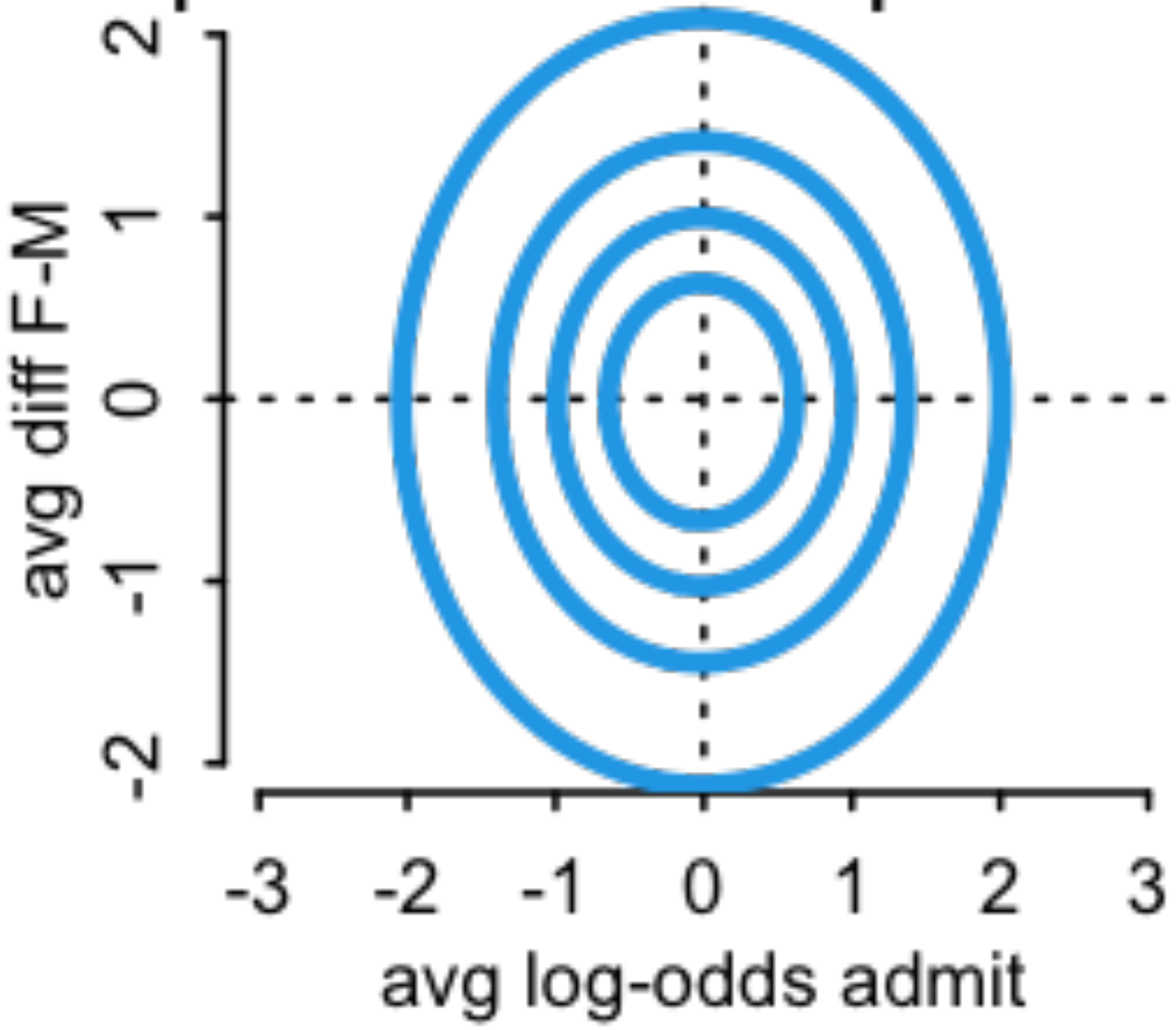
$$[\alpha_j, \beta_j] \sim \text{MVNormal}([\bar{\alpha}, \bar{\beta}], \Sigma)$$

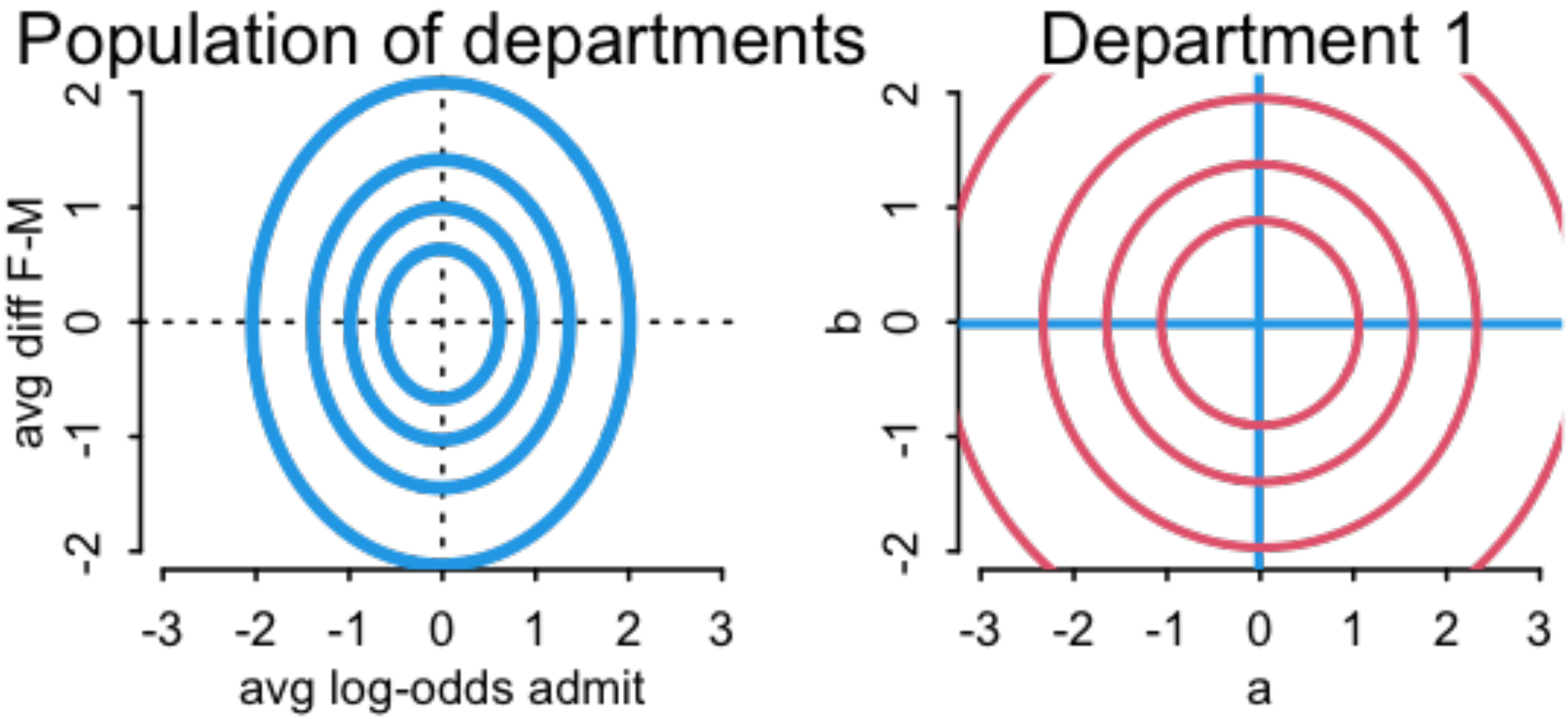
N features: N-dimensional distribution

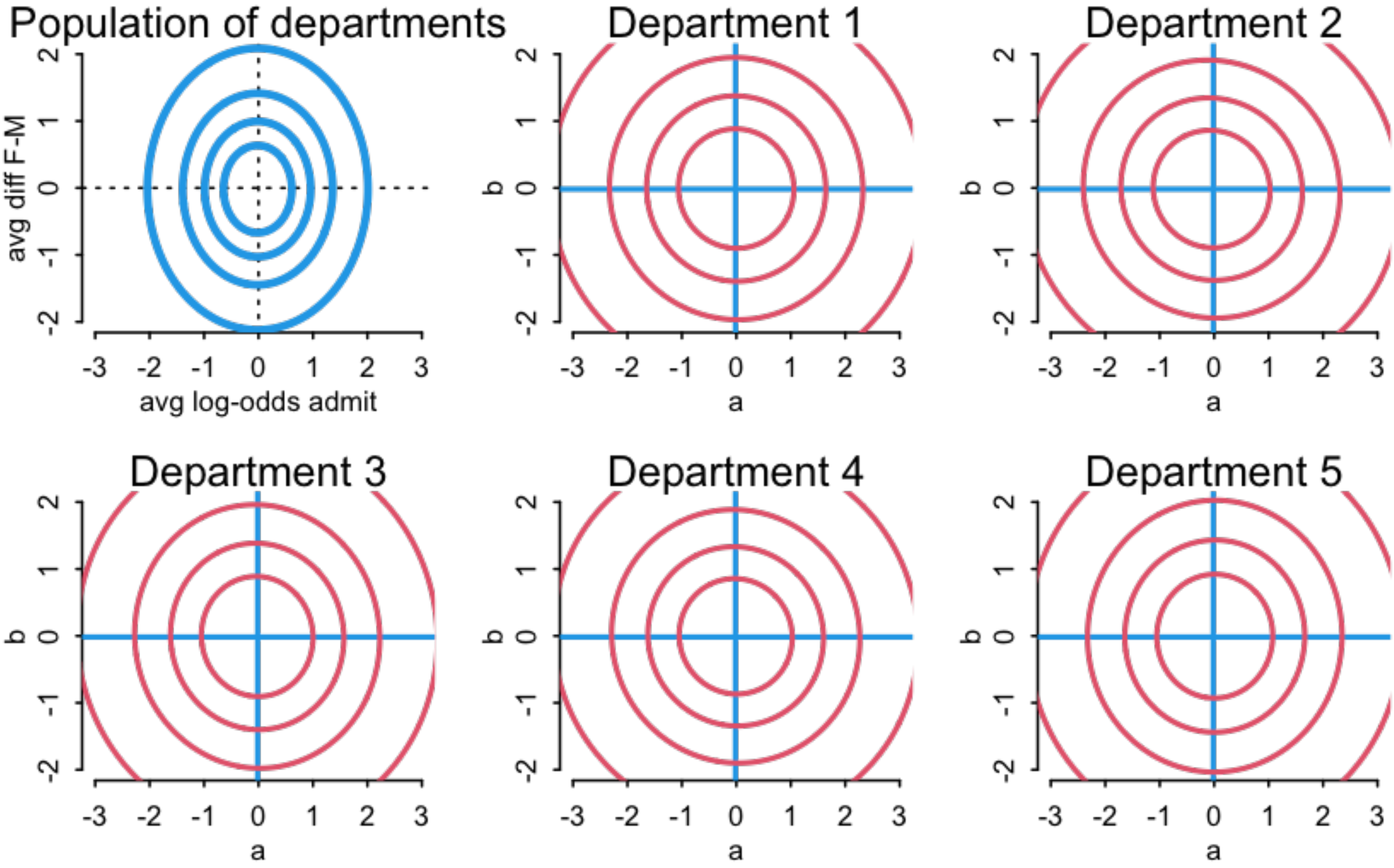
$$\alpha_{j,1..N} \sim \text{MVNormal}(A, \Sigma)$$

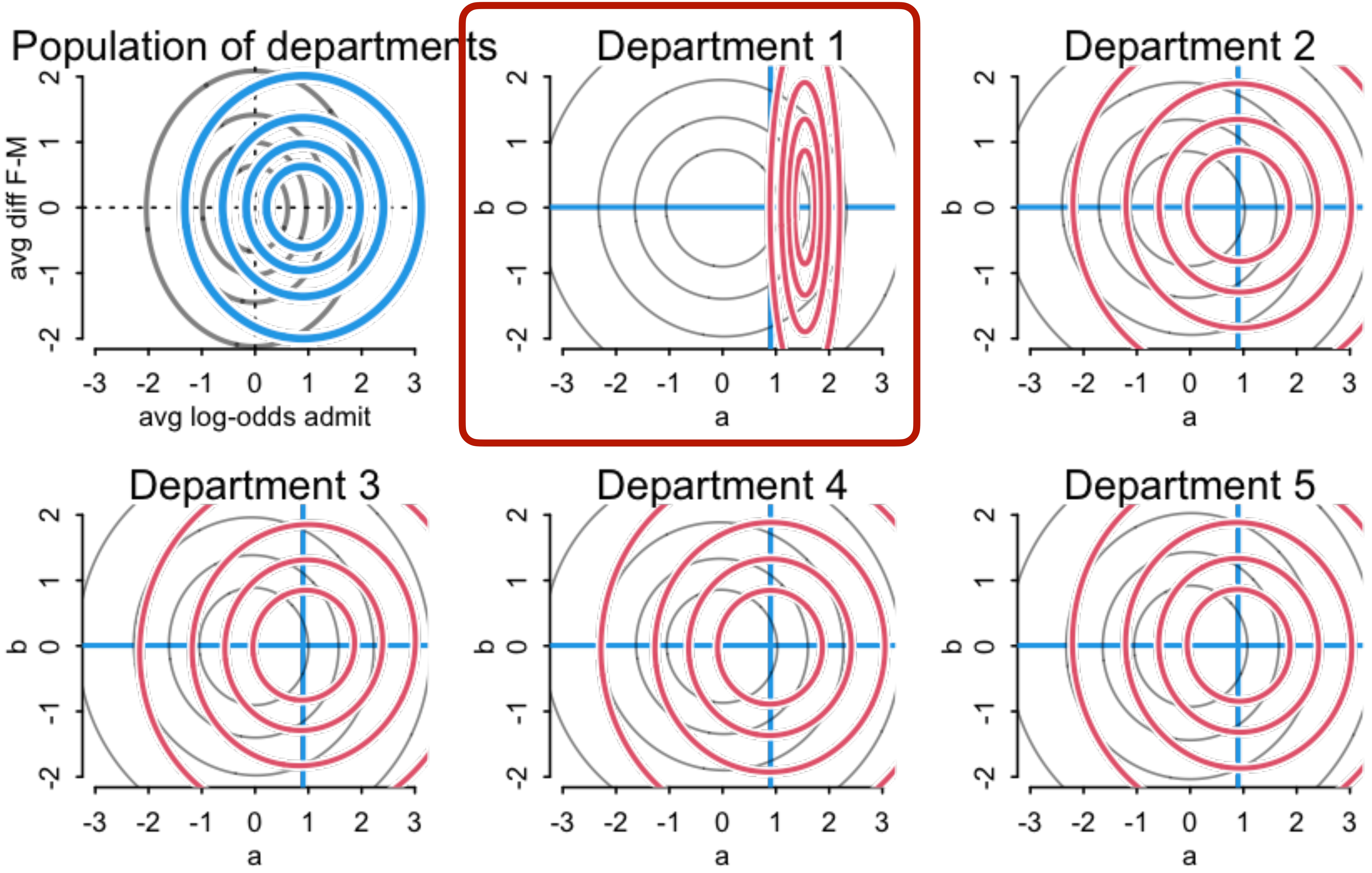
Hard part: Learning associations

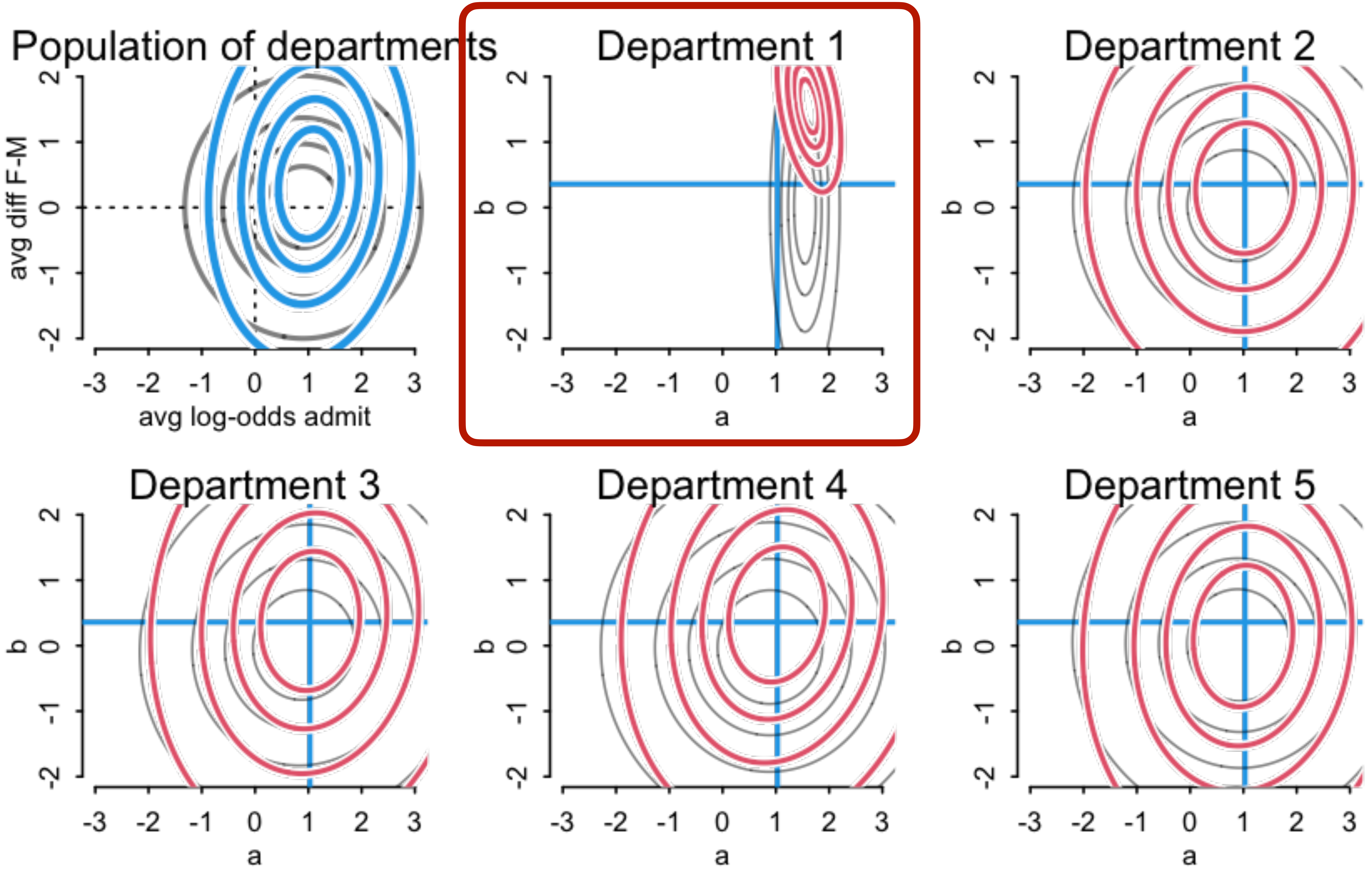
Population of departments



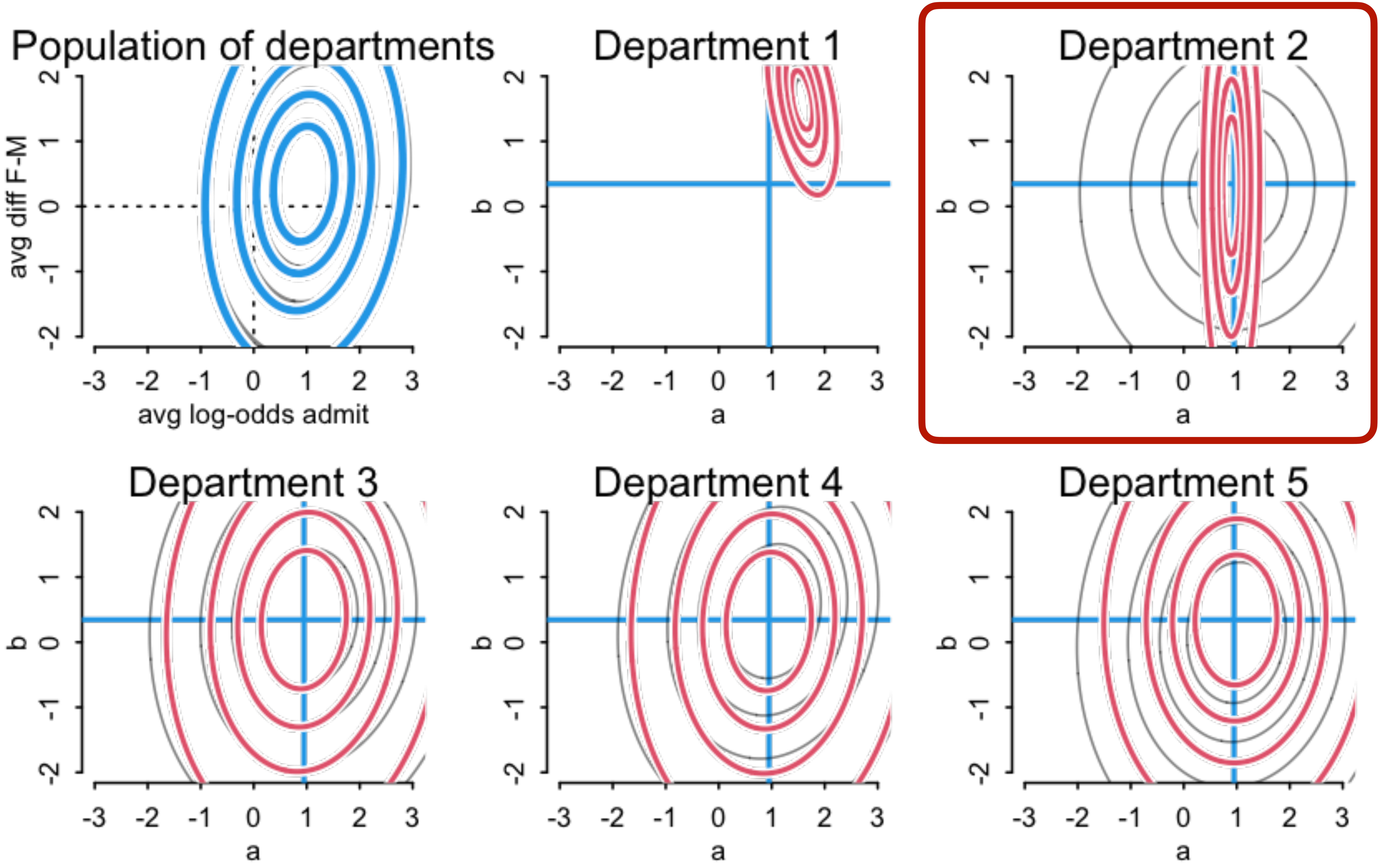

Population of departments
avg diff F-M
avg log-odds admit
-3
-2
-1
0
1
2
3
Department 1
b
a


Population of departments
avg diff F-M
avg log-odds admit
Department 1
Department 2
Department 3
Department 4
Department 5
a
b
-3 -2 -1 0 1 2 3
-2 -1 0 1 2

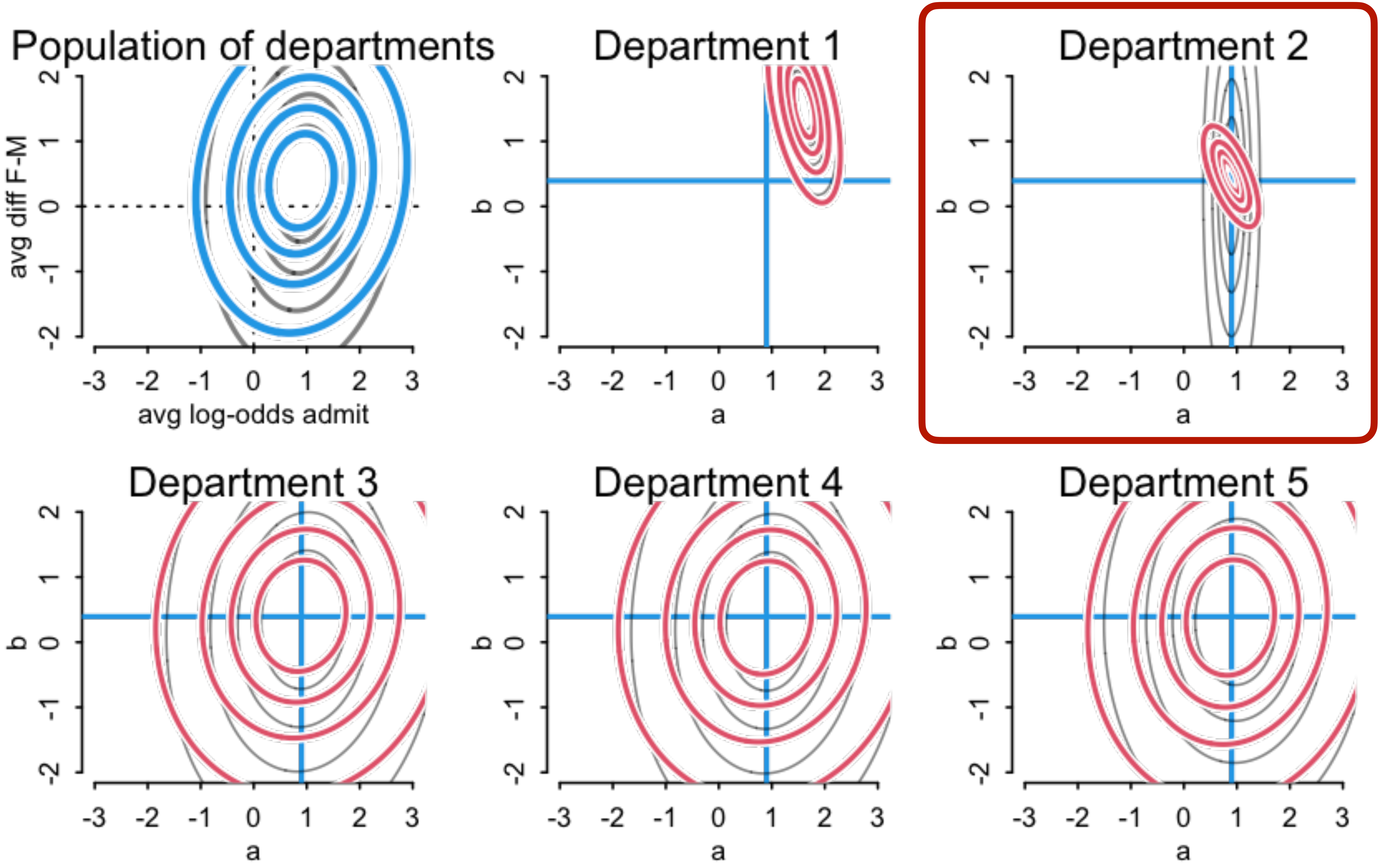

Population of departments
avg diff F-M
avg log-odds admit
Department 1
Department 2
Department 3
Department 4
Department 5
a
b
-3 -2 -1 0 1 2 3
-2 -1 0 1 2



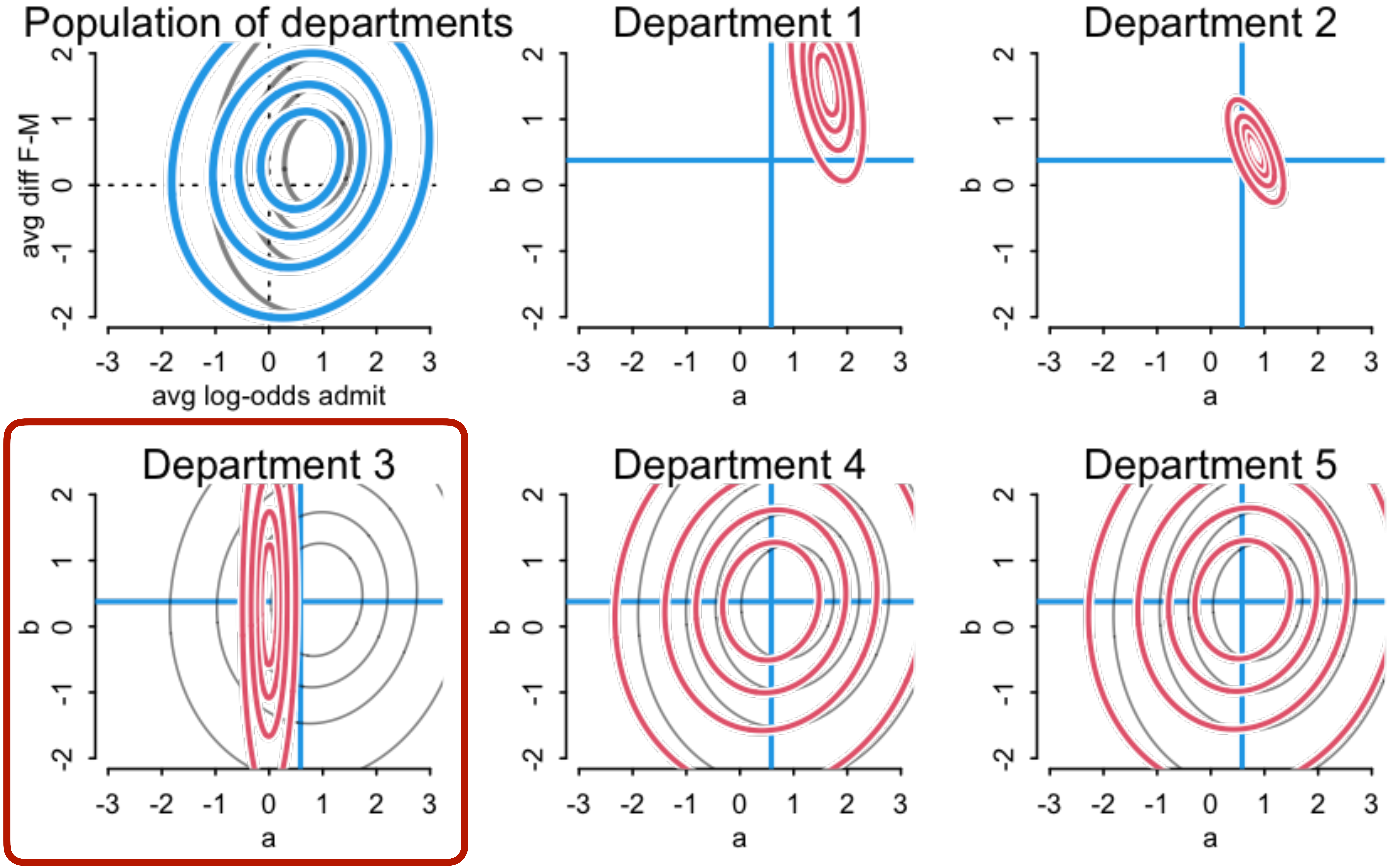
Population of departments
avg diff F-M
avg log-odds admit
Department 1
Department 2
Department 3
Department 4
Department 5
a
b
-3 -2 -1 0 1 2 3
-2 -1 0 1 2



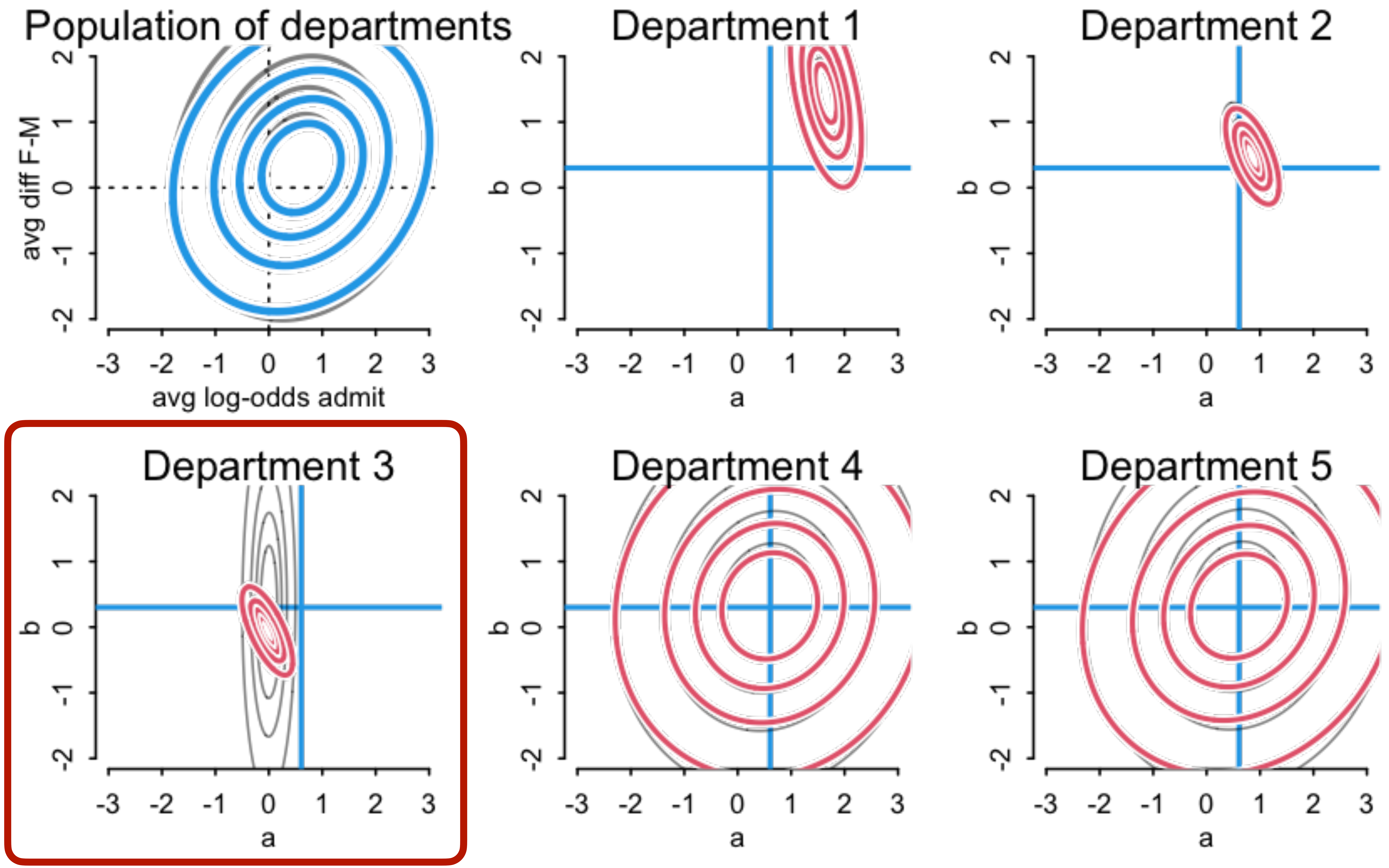
Population of departments
Department 1
Department 2
Department 3
Department 4
Department 5
avg diff F-M
avg log-odds admit
a
b



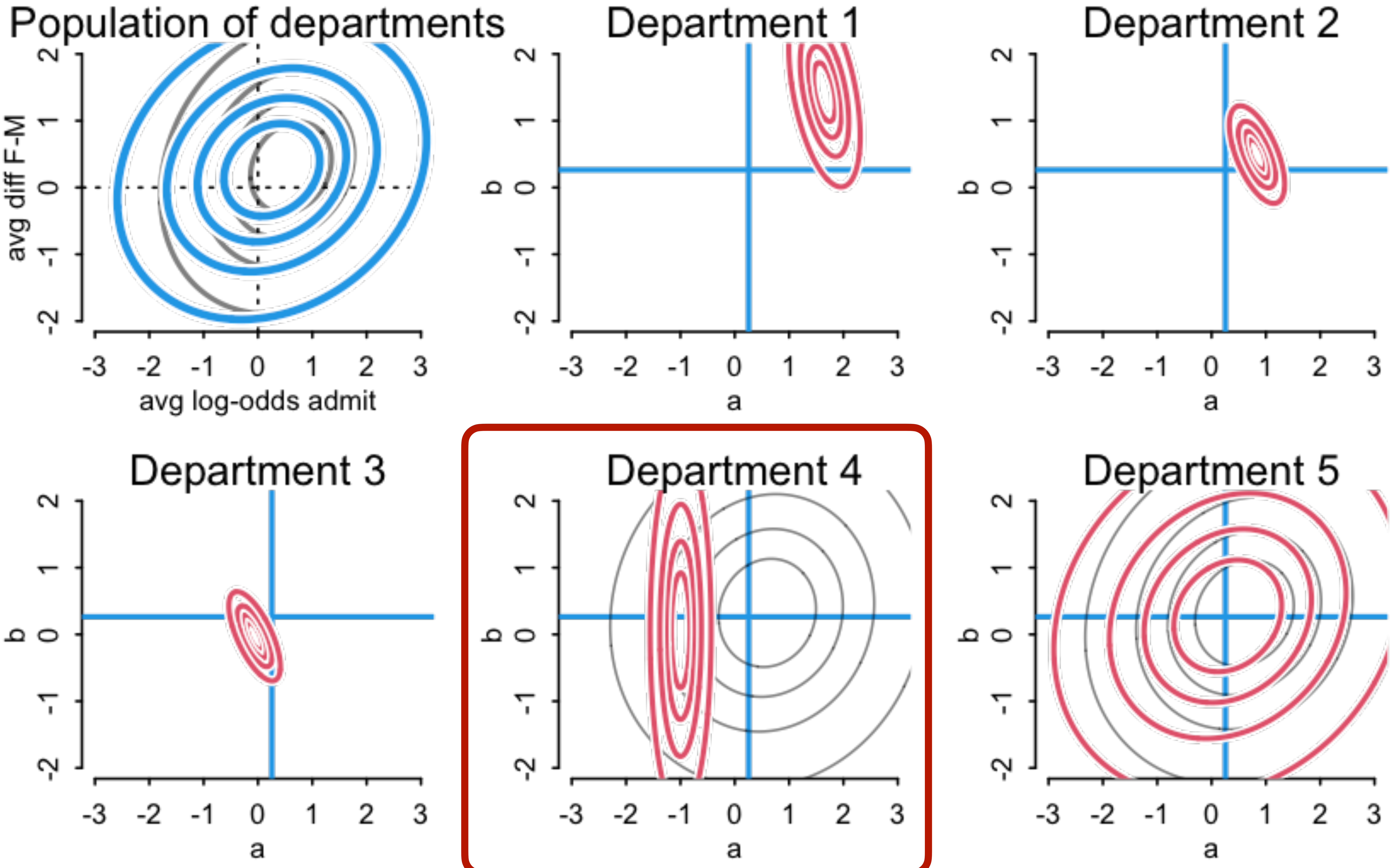
Population of departments
avg diff F-M
avg log-odds admit
Department 1
Department 2
Department 3
Department 4
Department 5
a
b
-3 -2 -1 0 1 2 3
-2 -1 0 1 2



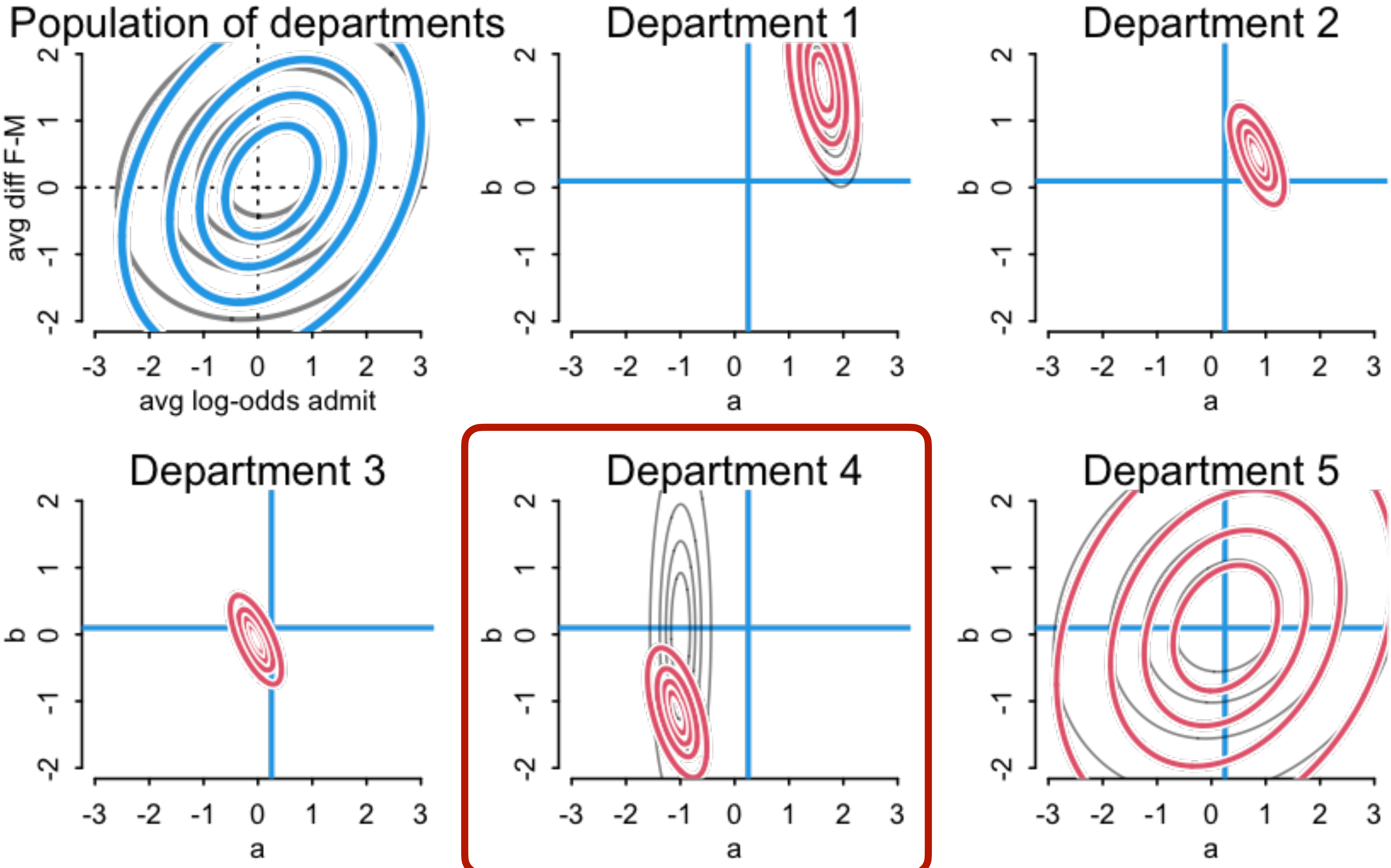
Population of departments
avg diff F-M
avg log-odds admit
Department 1
Department 2
Department 3
Department 4
Department 5
a
b
-3 -2 -1 0 1 2 3
-2 -1 0 1 2



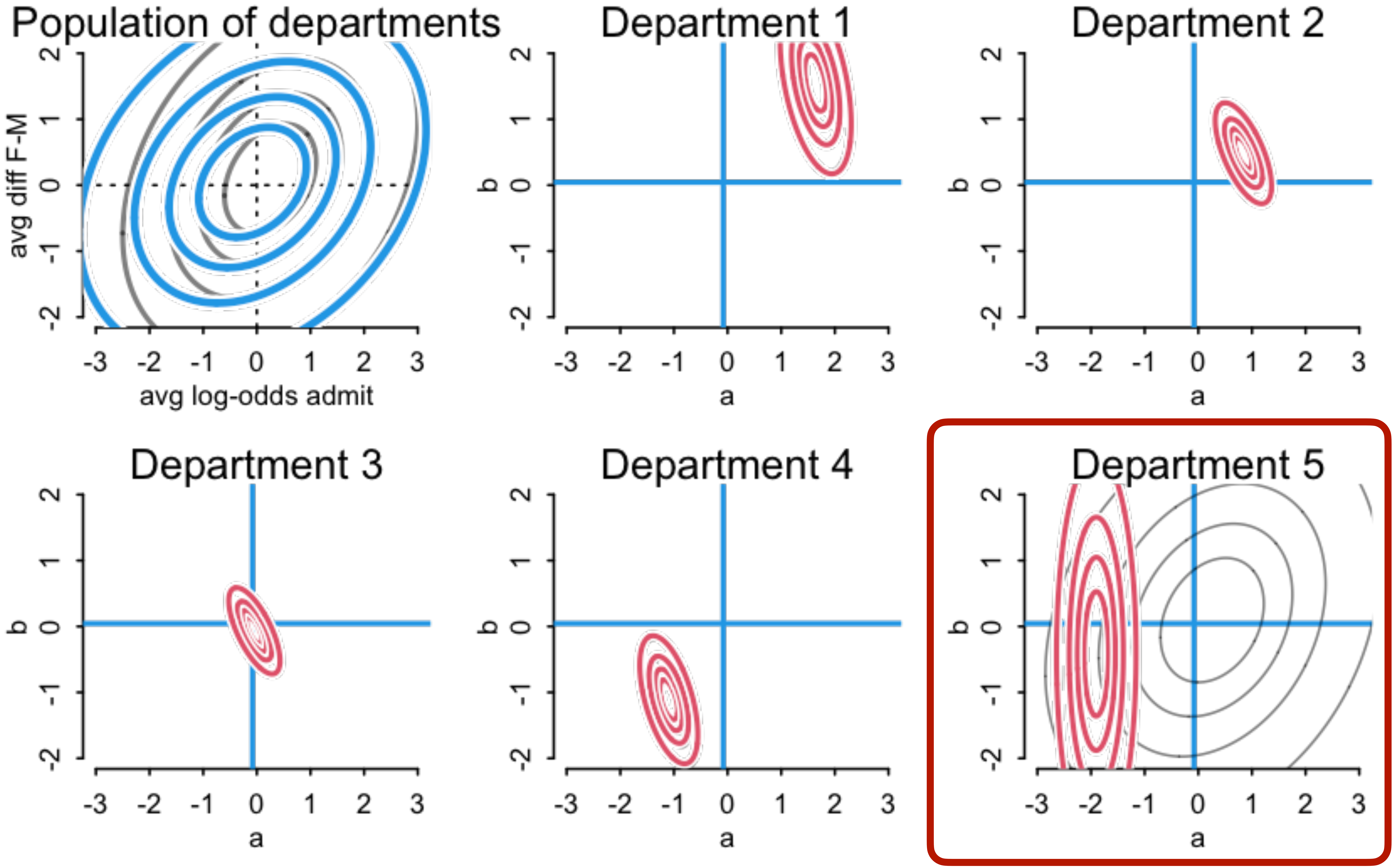
Population of departments
Department 1
Department 2
Department 3
Department 4
Department 5
avg diff F-M
avg log-odds admit
a
b

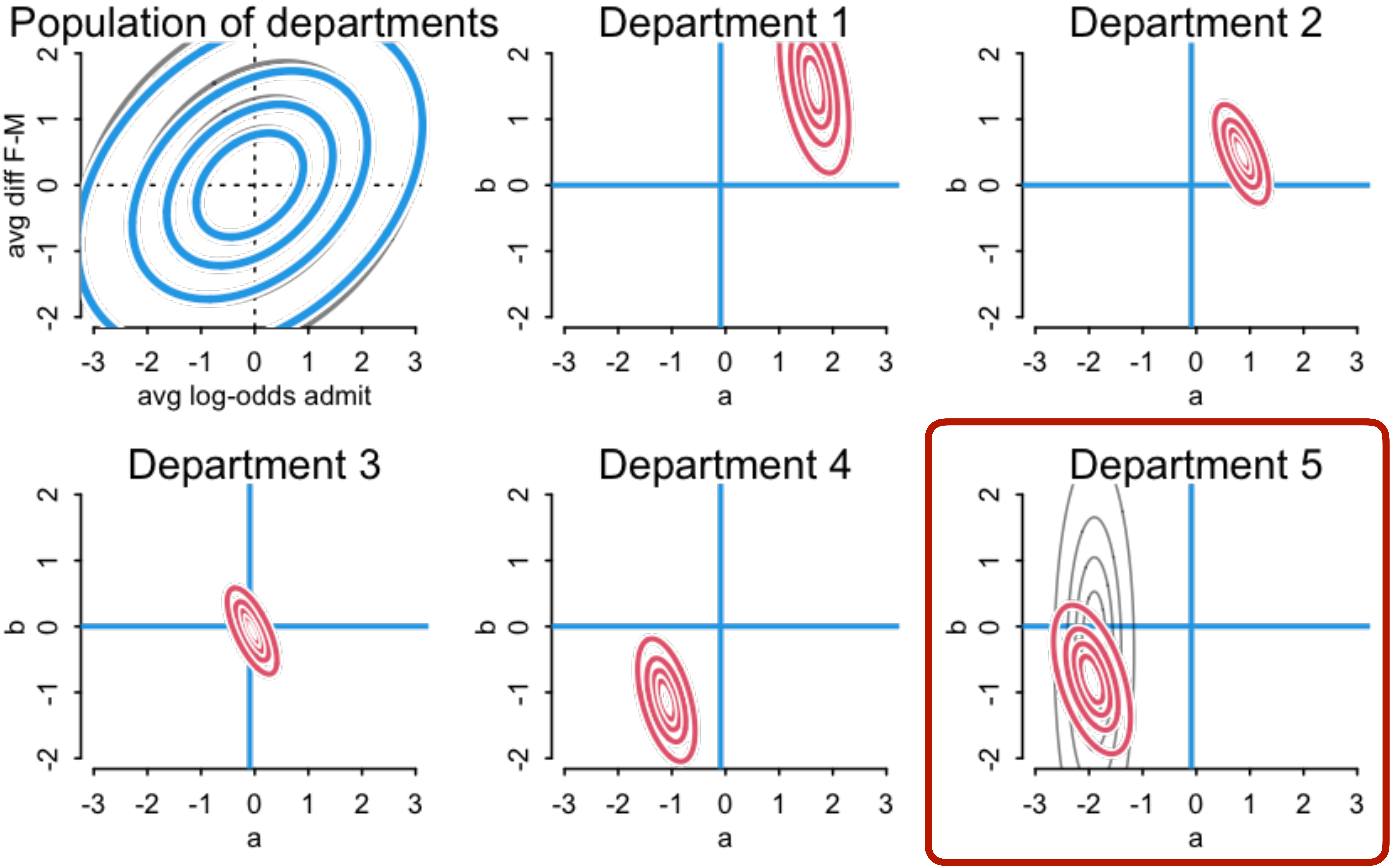


Population of departments
avg diff F-M
avg log-odds admit
Department 1
Department 2
Department 3
Department 4
Department 5
a
b
-3 -2 -1 0 1 2 3
-2 -1 0 1 2



Population of departments
avg diff F-M
avg log-odds admit
Department 1
Department 2
Department 3
Department 4
Department 5
a
b


Population of departments
avg diff F-M
avg log-odds admit
Department 1
Department 2
Department 3
Department 4
Department 5
a
b
-3
-2
-1
0
1
2
3

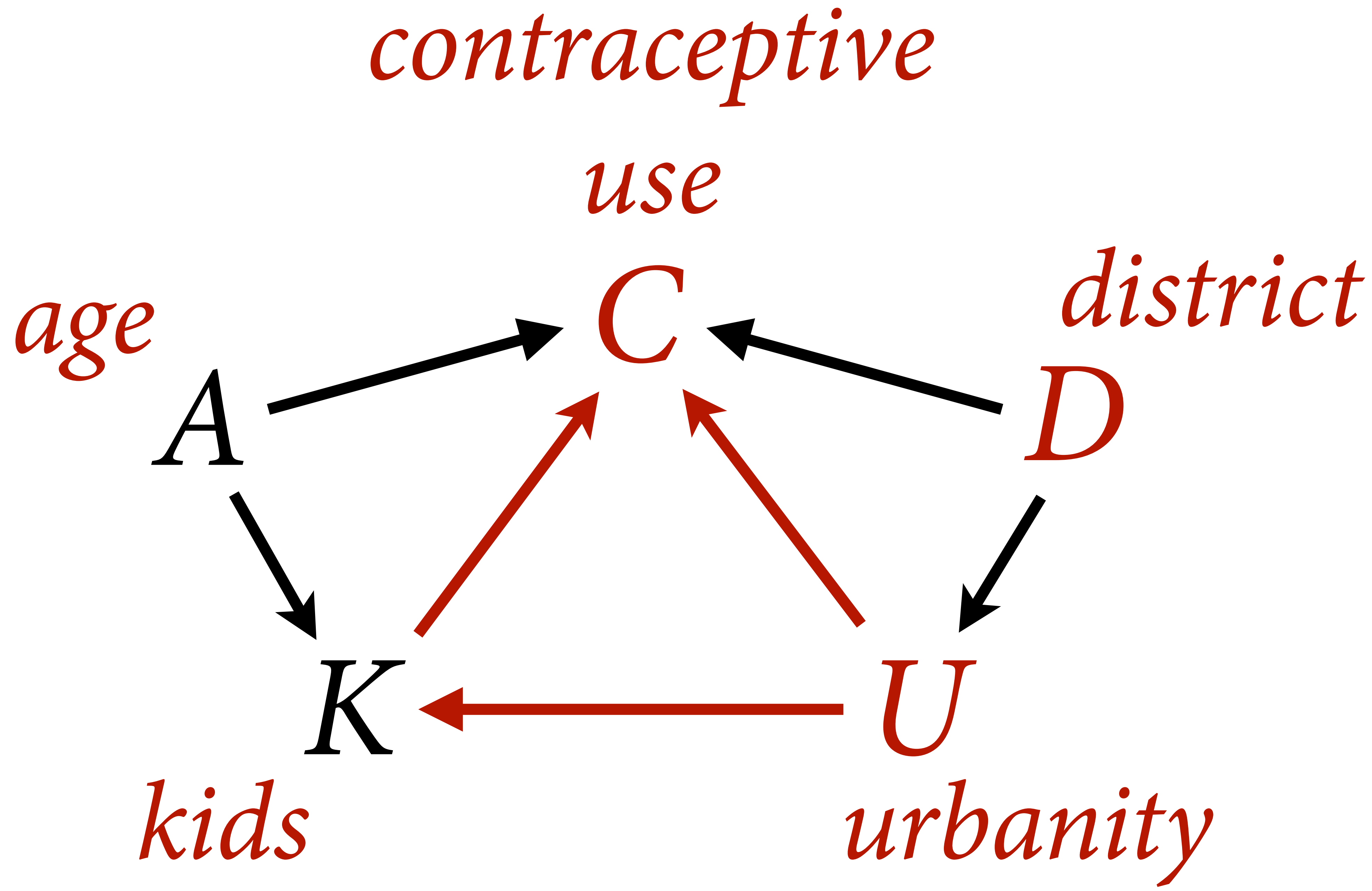

Population of departments
avg diff F-M
avg log-odds admit
Department 1
Department 2
Department 3
Department 4
Department 5
a
b
-3 -2 -1 0 1 2 3
-2 -1 0 1 2

Bangladesh 1989, Party Like It's

Estimand 1: C in each district

Estimand 2: Effect of U

Estimand 3: Effects of K and A



```
mCDUnc <- ulam(
    alist(
        C ~ bernoulli(p),
        logit(p) <- a[D] + b[D]*U,
        # define effects using other parameters
        save> vector[61]:a <<- abar + za*sigma,
        save> vector[61]:b <<- bbar + zb*tau,
        # z-scored effects
        vector[61]:za ~ normal(0,1),
        vector[61]:zb ~ normal(0,1),
        # ye olde hyper-priors
        c(abar,bbar) ~ normal(0,1),
        c(sigma,tau) ~ exponential(1)
    ) , data=dat , chains=4 , cores=4 )
```

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

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

$$\alpha_j = \bar{\alpha} + z_{\alpha,j} \times \sigma$$

$$\beta_j = \bar{\beta} + z_{\beta,j} \times \tau$$

$$z_{\alpha,j} \sim \text{Normal}(0,1)$$

$$z_{\beta,j} \sim \text{Normal}(0,1)$$

$$\bar{\alpha}, \bar{\beta} \sim \text{Normal}(0,1)$$

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

```r
dat <- list(
    C = d$use.contraception,
    D = as.integer(d$district),
    U = ifelse(d$urban==1,1,0) )

# total U
mCDU <- ulam(
    alist(
        C ~ bernoulli(p),
        logit(p) <- a[D] + b[D]*U,
        vector[61]:a ~ normal(abar,sigma),
        vector[61]:b ~ normal(bbar,tau),
        c(abar,bbar) ~ normal(0,1),
        c(sigma,tau) ~ exponential(1)
    ) , data=dat , chains=4 , cores=4 )
```

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

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

$$\alpha_j \sim \text{Normal}(\bar{\alpha}, \sigma)$$

$$\beta_j \sim \text{Normal}(\bar{\beta}, \tau)$$

$$\bar{\alpha}, \bar{\beta} \sim \text{Normal}(0,1)$$

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

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

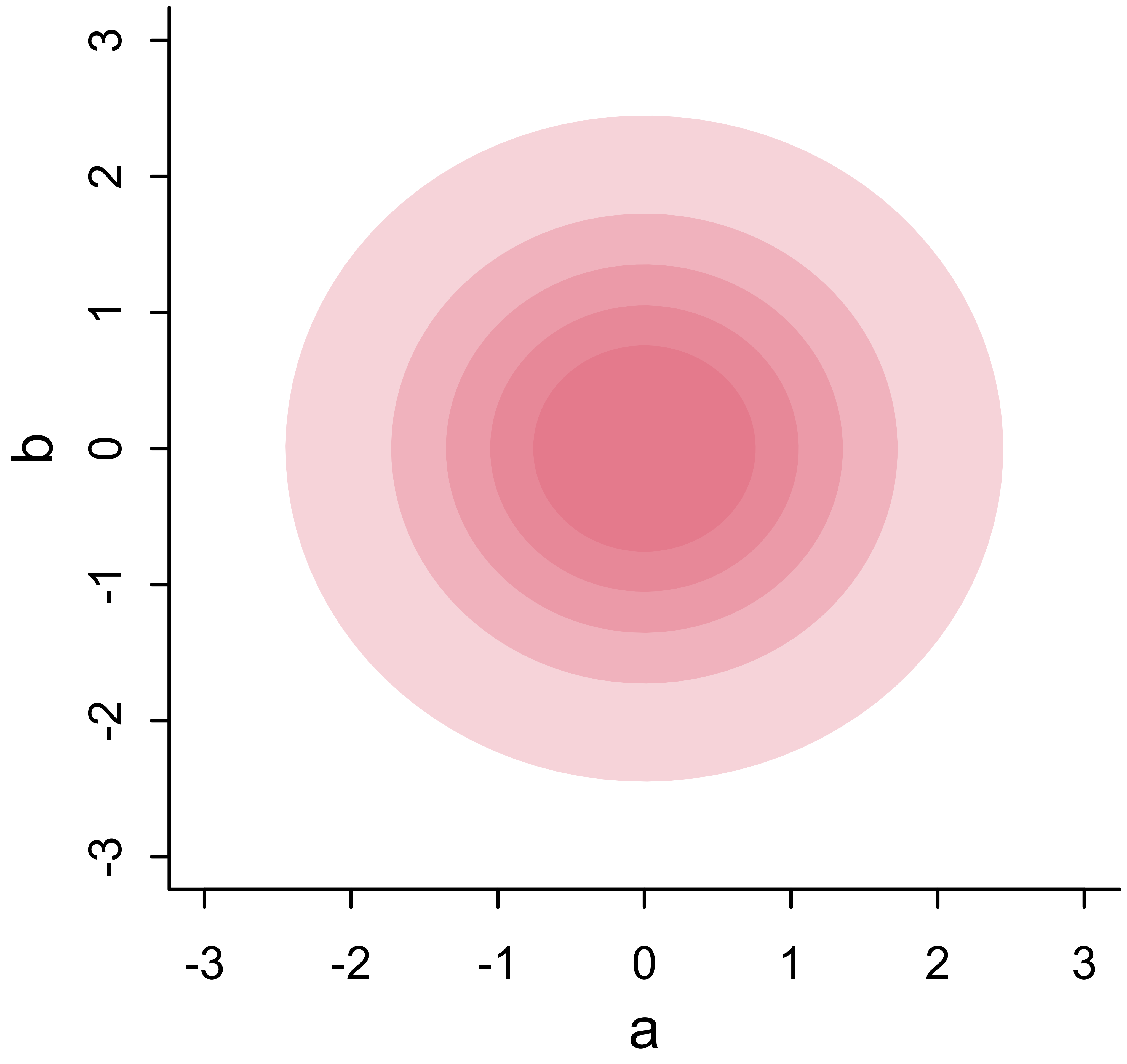
$$\text{logit}(p_i) = \alpha_{D[i]} + \beta_{D[i]} U_i$$

$$\alpha_j \sim \text{Normal}(\bar{\alpha}, \sigma)$$

$$\beta_j \sim \text{Normal}(\bar{\beta}, \tau)$$

$$\bar{\alpha}, \bar{\beta} \sim \text{Normal}(0,1)$$

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



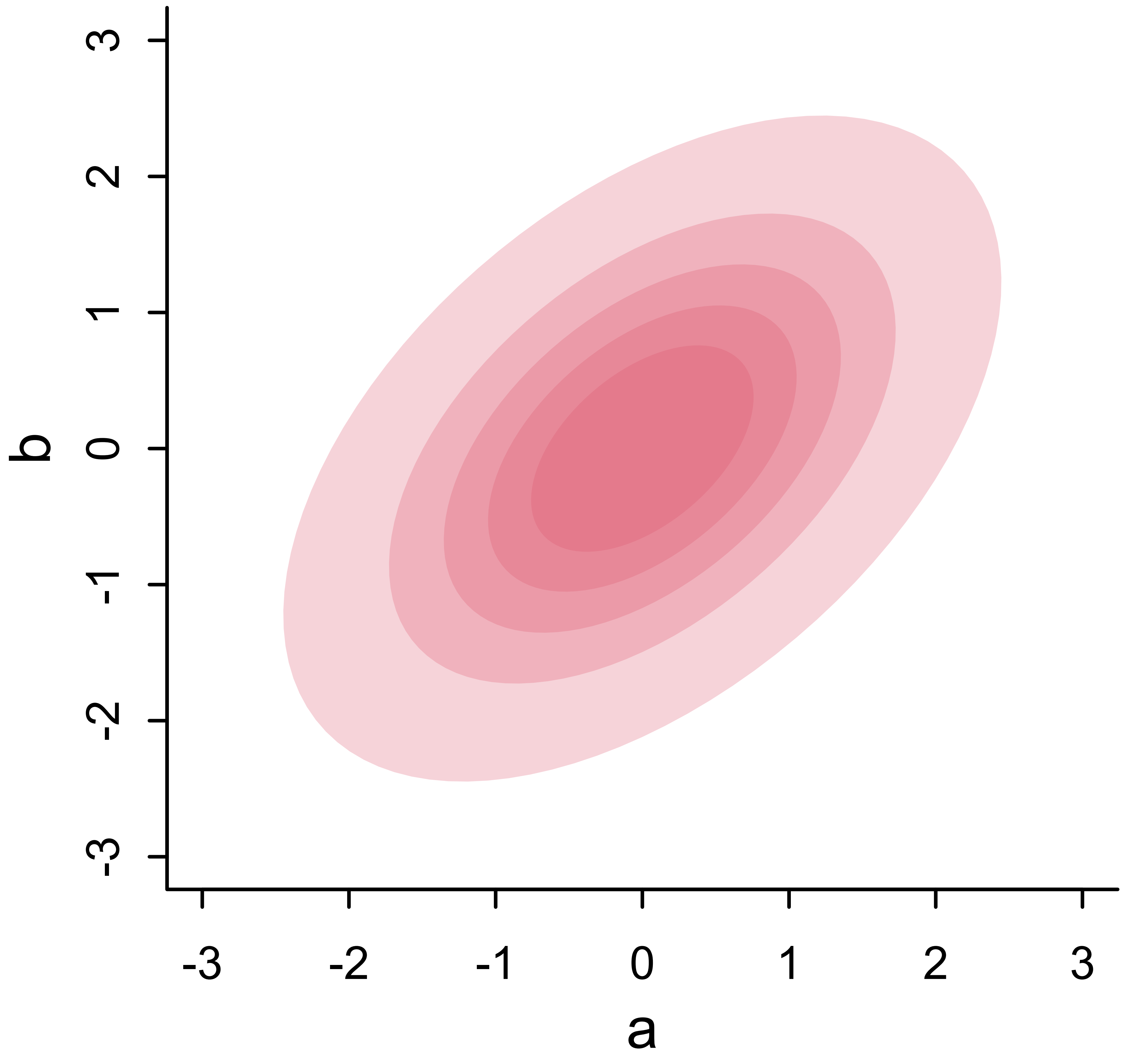
$$C_i \sim \mathrm{Bernoulli}(p_i)$$

$$\mathrm{logit}(p_i) = \alpha_{D[i]} + \beta_{D[i]} U_i$$

$$[\alpha_j, \beta_j] \sim \mathrm{MVNormal}([\bar{\alpha}, \bar{\beta}], \mathrm{R}, [\sigma, \tau])$$

$$\bar{\alpha}, \bar{\beta} \sim \mathrm{Normal}(0,1)$$

$$\sigma, \tau \sim \mathrm{Exponential}(1)$$



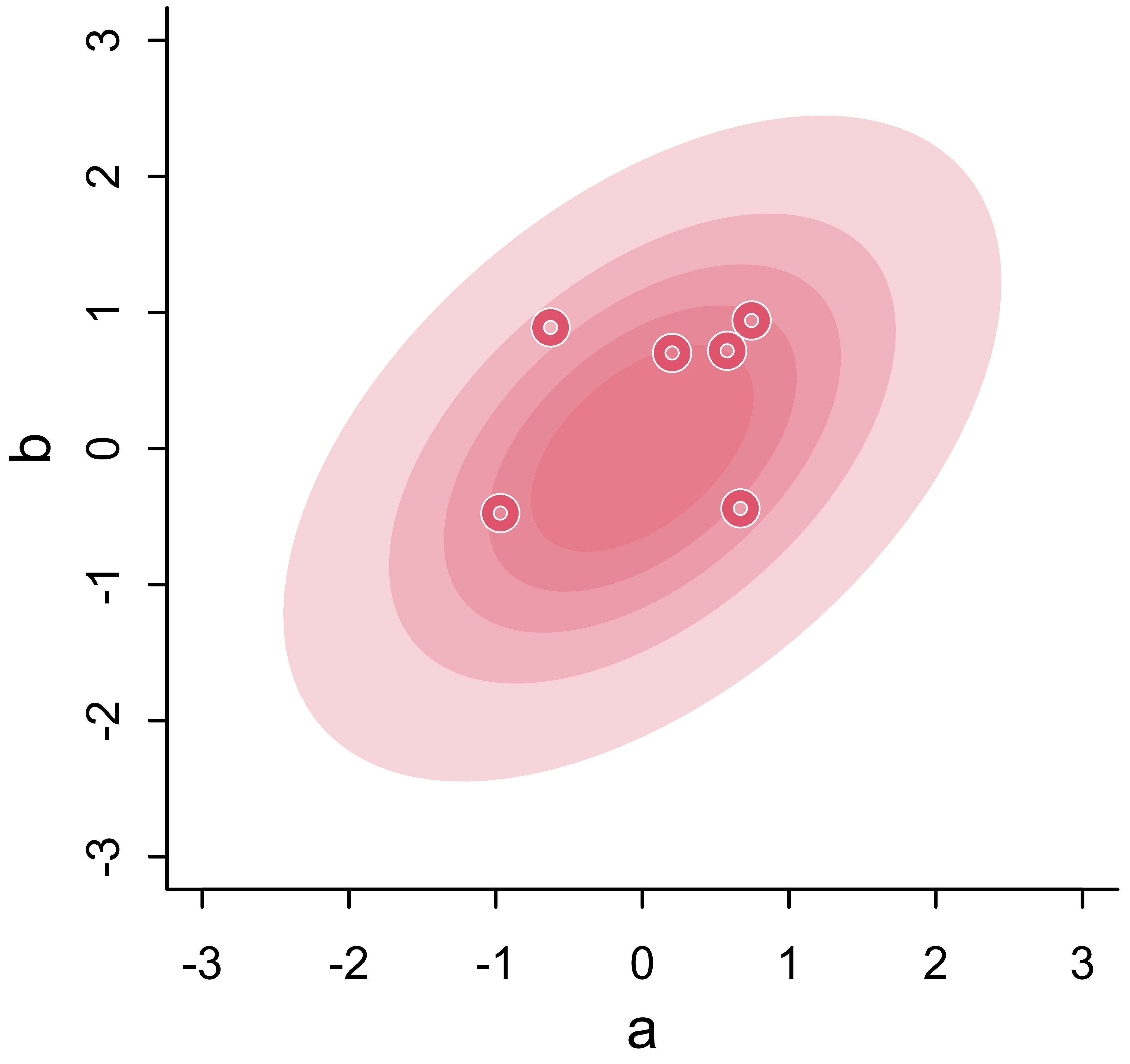
features for district j

correlation matrix

$$[\alpha_j, \beta_j] \sim \text{MVNormal}([\bar{\alpha}, \bar{\beta}], \text{R}, [\sigma, \tau])$$

feature means

standard deviations



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

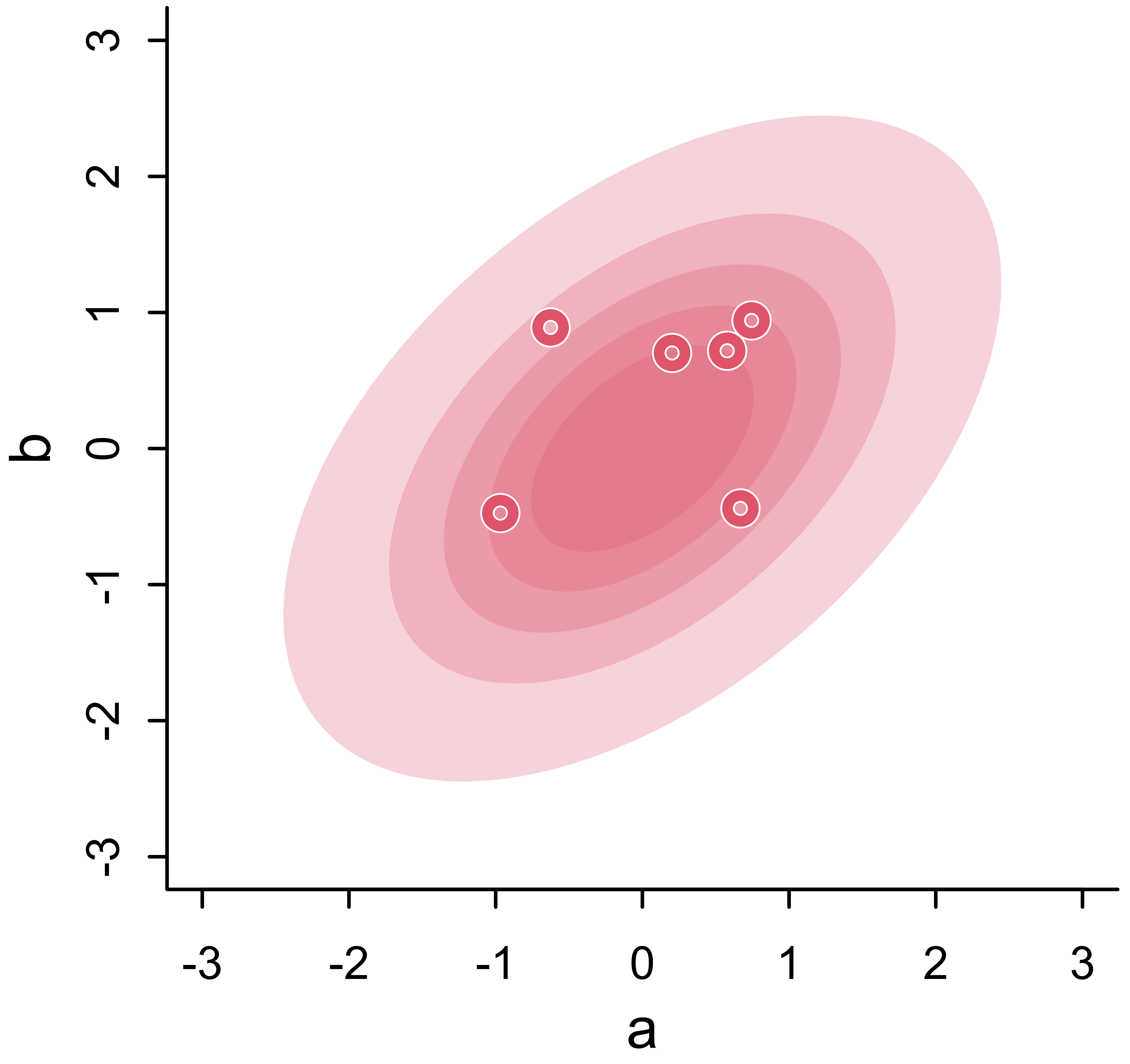
$$\text{logit}(p_i) = \alpha_{D[i]} + \beta_{D[i]} U_i$$

$$[\alpha_j, \beta_j] \sim \text{MVNormal}([\bar{\alpha}, \bar{\beta}], \text{R}, [\sigma, \tau])$$

$$\bar{\alpha}, \bar{\beta} \sim \text{Normal}(0,1)$$

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

$$\text{R} \sim ?$$



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

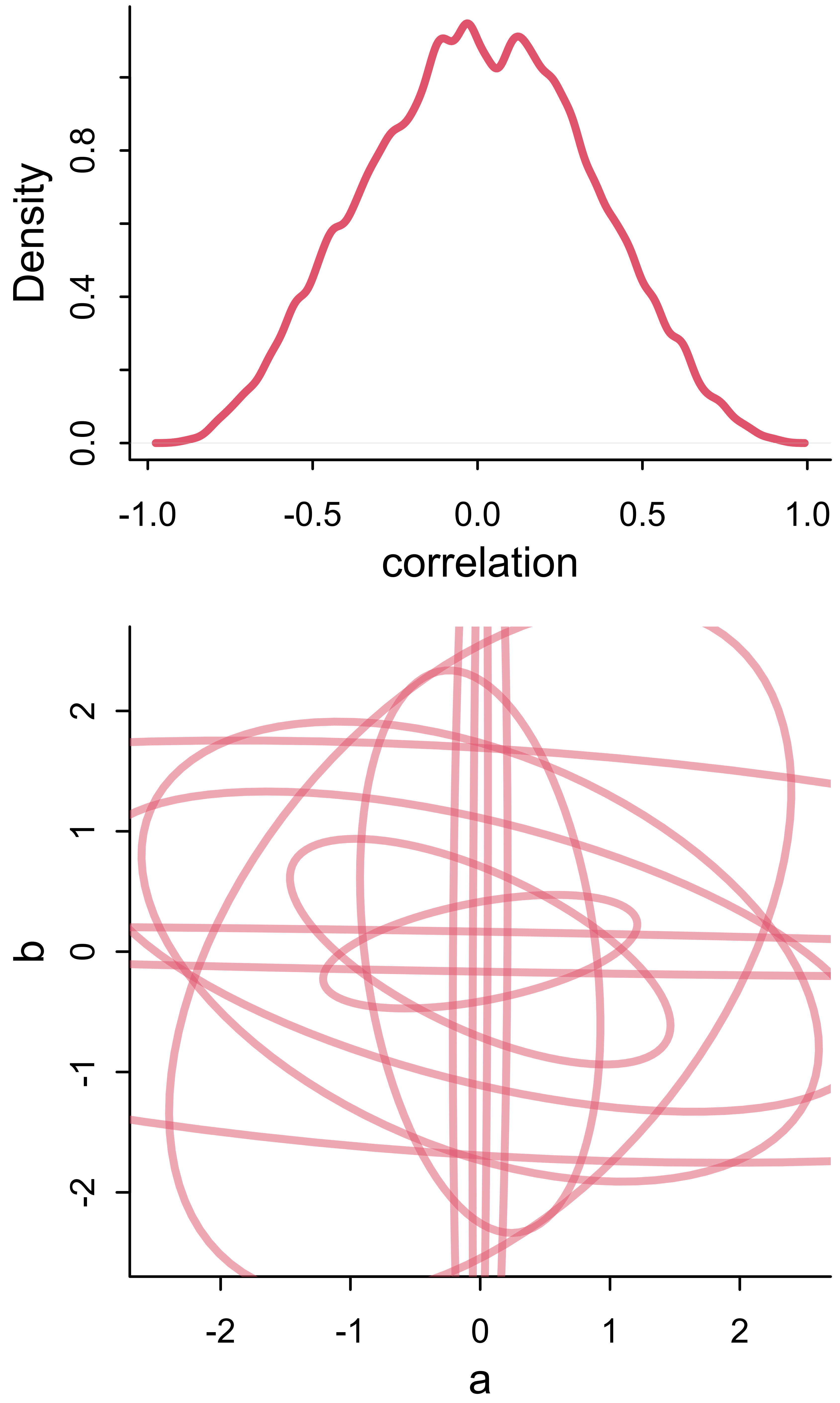
$$\text{logit}(p_i) = \alpha_{D[i]} + \beta_{D[i]} U_i$$

$$[\alpha_j, \beta_j] \sim \text{MVNormal}([\bar{\alpha}, \bar{\beta}], \text{R}, [\sigma, \tau])$$

$$\bar{\alpha}, \bar{\beta} \sim \text{Normal}(0,1)$$

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

$$\text{R} \sim \text{LKJCorr}(4)$$



PAUSE

```r
mCDUcov <- ulam(
    alist(
        C ~ bernoulli(p),
        logit(p) <- a[D] + b[D]*U,

        # define effects using other parameters
        transpars> vector[61]:a <<- v[,1],
        transpars> vector[61]:b <<- v[,2],

        # priors - centered correlated varying effects
        matrix[61,2]:v ~ multi_normal(abar,Rho,sigma),
        vector[2]:abar ~ normal(0,1),
        corr_matrix[2]:Rho ~ lkj_corr(4),
        vector[2]:sigma ~ exponential(1)
    ) , data=dat , chains=4 , cores=4 )
```

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

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

$$[\alpha_j, \beta_j] \sim \text{MVNormal}([\bar{\alpha}, \bar{\beta}], \text{R}, [\sigma, \tau])$$

$$\bar{\alpha}, \bar{\beta} \sim \text{Normal}(0,1)$$

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

$$\text{R} \sim \text{LKJCorr}(4)$$

```
mCDUcov <- ulam(
    alist(
        C ~ bernoulli(p),
        logit(p) <- a[D] + b[D]*U,

        # define effects using other parameters
        transpars> vector[61]:a <<- v[,1],
        transpars> vector[61]:b <<- v[,2],

        # priors - centered correlated varying effects
        matrix[61,2]:v ~ multi_normal(abar,Rho,sigma),
        vector[2]:abar ~ normal(0,1),
        corr_matrix[2]:Rho ~ lkj_corr(4),
        vector[2]:sigma ~ exponential(1)
    ) , data=dat , chains=4 , cores=4 )
```

61 districts, 2 features each

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

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

$$[\alpha_j, \beta_j] \sim \text{MVNormal}([\bar{\alpha}, \bar{\beta}], \text{R}, [\sigma, \tau])$$

$$\bar{\alpha}, \bar{\beta} \sim \text{Normal}(0,1)$$

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

$$\text{R} \sim \text{LKJCorr}(4)$$

```r
mCDUcov <- ulam(
    alist(
        C ~ bernoulli(p),
        logit(p) <- a[D] + b[D]*U,

        # define effects using other parameters
        transpars> vector[61]:a <<- v[,1],
        transpars> vector[61]:b <<- v[,2],

        # priors - centered correlated varying effects
        matrix[61,2]:v ~ multi_normal(abar,Rho,sigma),
        vector[2]:abar ~ normal(0,1),
        corr_matrix[2]:Rho ~ lkj_corr(4),
        vector[2]:sigma ~ exponential(1)
    ) , data=dat , chains=4 , cores=4 )
```

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

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

$$\boxed{[\alpha_j, \beta_j] \sim \text{MVNormal}([\bar{\alpha}, \bar{\beta}], \text{R}, [\sigma, \tau])}$$

$$\bar{\alpha}, \bar{\beta} \sim \text{Normal}(0,1)$$

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

$$\text{R} \sim \text{LKJCorr}(4)$$

```r
mCDUcov <- ulam(
    alist(
        C ~ bernoulli(p),
        logit(p) <- a[D] + b[D]*U,

        # define effects using other parameters
        transpars> vector[61]:a <<- v[,1],
        transpars> vector[61]:b <<- v[,2],

        # priors - centered correlated varying effects
        matrix[61,2]:v ~ multi_normal(abar,Rho,sigma),
        vector[2]:abar ~ normal(0,1),
        corr_matrix[2]:Rho ~ lkj_corr(4),
        vector[2]:sigma ~ exponential(1)
    ) , data=dat , chains=4 , cores=4 )
```

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

$$\boxed{\text{logit}(p_i) = \alpha_{D[i]} + \beta_{D[i]} U_i}$$

$$[\alpha_j, \beta_j] \sim \text{MVNormal}([\bar{\alpha}, \bar{\beta}], \text{R}, [\sigma, \tau])$$

$$\bar{\alpha}, \bar{\beta} \sim \text{Normal}(0,1)$$

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

$$\text{R} \sim \text{LKJCorr}(4)$$

```r
mCDUcov <- ulam(
    alist(
        C ~ bernoulli(p),
        logit(p) <- a[D] + b[D]*U,

        # define effects using other parameters
        transpars> vector[61]:a <<- v[,1],
        transpars> vector[61]:b <<- v[,2],

        # priors - centered correlated varying effects
        matrix[61,2]:v ~ multi_normal(abar,Rho,sigma),
        vector[2]:abar ~ normal(0,1),
        corr_matrix[2]:Rho ~ lkj_corr(4),
        vector[2]:sigma ~ exponential(1)
    ) , data=dat , chains=4 , cores=4 )
```

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

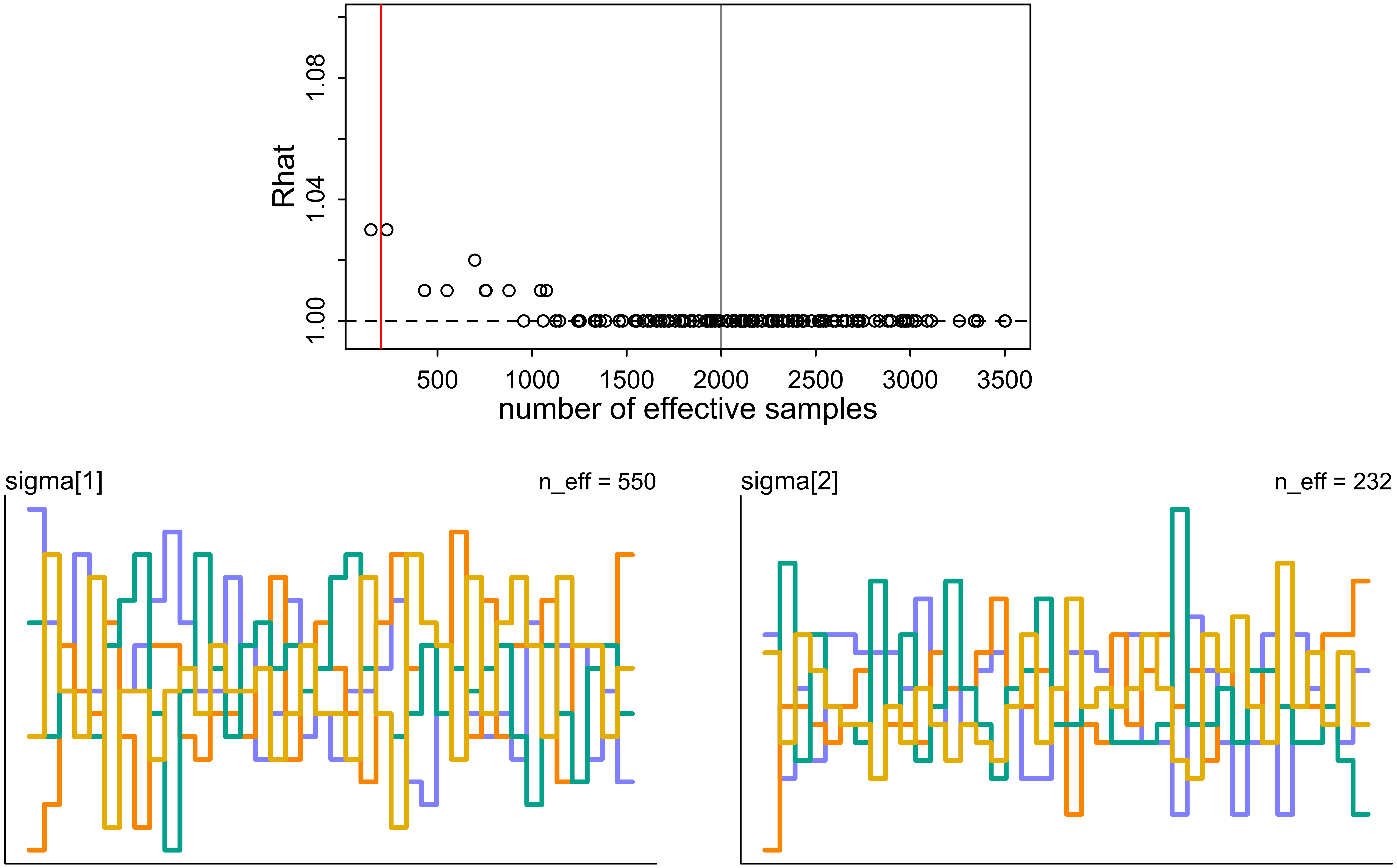
$$\text{logit}(p_i) = \alpha_{D[i]} + \beta_{D[i]} U_i$$

$$[\alpha_j, \beta_j] \sim \text{MVNormal}([\bar{\alpha}, \bar{\beta}], \text{R}, [\sigma, \tau])$$

$$\bar{\alpha}, \bar{\beta} \sim \text{Normal}(0,1)$$

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

$$\boxed{\text{R} \sim \text{LKJCorr}(4)}$$



Rhat
1.00
1.04
1.08
500
1000
1500
2000
2500
3000
3500
number of effective samples
sigma[1]
n_eff = 550
sigma[2]
n_eff = 232

Centered priors

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

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

$$[\alpha_j, \beta_j] \sim \text{MVNormal}([\bar{\alpha}, \bar{\beta}], \text{R}, [\sigma, \tau])$$

$$\bar{\alpha}, \bar{\beta} \sim \text{Normal}(0,1)$$

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

$$\text{R} \sim \text{LKJCorr}(4)$$

Centered priors

$$
\begin{aligned}
C_i &\sim \text{Bernoulli}(p_i) \\
\text{logit}(p_i) &= \alpha_{D[i]} + \beta_{D[i]} U_i \\
[\alpha_j, \beta_j] &\sim \text{MVNormal}([\bar{\alpha}, \bar{\beta}], \text{R}, [\sigma, \tau]) \\
\bar{\alpha}, \bar{\beta} &\sim \text{Normal}(0,1) \\
\sigma, \tau &\sim \text{Exponential}(1) \\
\text{R} &\sim \text{LKJCorr}(4)
\end{aligned}
$$

Non-centered priors

$$
\begin{aligned}
C_i &\sim \text{Bernoulli}(p_i) \\
\text{logit}(p_i) &= \alpha_{D[i]} + \beta_{D[i]} U_i \\
\alpha &= \bar{\alpha} + v_{\_,1} \\
\beta &= \bar{\beta} + v_{\_,2} \\
v &= \left(\text{diag}(\sigma)\mathbf{LZ}\right)^\intercal \\
\mathbf{Z}_{j,k} &\sim \text{Normal}(0,1) \\
\bar{\alpha}, \bar{\beta} &\sim \text{Normal}(0,1) \\
\sigma &\sim \text{Exponential}(1) \\
\text{R} &\sim \text{LKJCorr}(4)
\end{aligned}
$$

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

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

$$\alpha = \bar{\alpha} + v_{\_,1}$$

$$\beta = \bar{\beta} + v_{\_,2}$$

$$v = \left(\text{diag}(\sigma)\mathbf{L}\mathbf{Z}\right)^{\intercal}$$

$$\mathbf{Z}_{j,k} \sim \text{Normal}(0,1)$$

$$\bar{\alpha}, \bar{\beta} \sim \text{Normal}(0,1)$$

$$\sigma \sim \text{Exponential}(1)$$

$$\text{R} \sim \text{LKJCorr}(4)$$

```
# covariance - non-centered
mCDUcov_nc <- ulam(
    alist(
        C ~ bernoulli(p),
        logit(p) <- a[D] + b[D]*U,

        # define effects using other parameters
        # this is the non-centered Cholesky machine
        transpars> vector[61]:a <<- abar[1] + v[,1],
        transpars> vector[61]:b <<- abar[2] + v[,2],
        transpars> matrix[61,2]:v <-
            compose_noncentered( sigma , L_Rho , Z ),

        # priors - note that none have parameters inside them
        # that is what makes them non-centered
        matrix[2,61]:Z ~ normal( 0 , 1 ),
        vector[2]:abar ~ normal(0,1),
        cholesky_factor_corr[2]:L_Rho ~ lkj_corr_cholesky( 4 ),
        vector[2]:sigma ~ exponential(1),

        # convert Cholesky to Corr matrix
        gq> matrix[2,2]:Rho <<- Chol_to_Corr(L_Rho)
    ) , data=dat , chains=4 , cores=4 )
```

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

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

$$\alpha = \bar{\alpha} + v_{\_,1}$$

$$\beta = \bar{\beta} + v_{\_,2}$$

$$v = \left(\text{diag}(\sigma)\mathbf{LZ}\right)^{\intercal}$$

$$\mathbf{Z}_{j,k} \sim \text{Normal}(0,1)$$

$$\bar{\alpha}, \bar{\beta} \sim \text{Normal}(0,1)$$

$$\sigma \sim \text{Exponential}(1)$$

$$\text{R} \sim \text{LKJCorr}(4)$$

```
# covariance - non-centered
mCDUcov_nc <- ulam(
    alist(
        C ~ bernoulli(p),
        logit(p) <- a[D] + b[D]*U,

        # define effects using other parameters
        # this is the non-centered Cholesky machine
        transpars> vector[61]:a <<- abar[1] + v[,1],
        transpars> vector[61]:b <<- abar[2] + v[,2],
        transpars> matrix[61,2]:v <-
            compose_noncentered( sigma , L_Rho , Z ),

        # priors - note that none have parameters inside them
        # that is what makes them non-centered
        matrix[2,61]:Z ~ normal( 0 , 1 ),
        vector[2]:abar ~ normal(0,1),
        cholesky_factor_corr[2]:L_Rho ~ lkj_corr_cholesky( 4 ),
        vector[2]:sigma ~ exponential(1),

        # convert Cholesky to Corr matrix
        gq> matrix[2,2]:Rho <<- Chol_to_Corr(L_Rho)
    ) , data=dat , chains=4 , cores=4 )
```

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

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

$$\alpha = \bar{\alpha} + v_{\_,1}$$

$$\beta = \bar{\beta} + v_{\_,2}$$

$$v = \left(\text{diag}(\sigma)\mathbf{LZ}\right)^{\intercal}$$

$$\mathbf{Z}_{j,k} \sim \text{Normal}(0,1)$$

$$\bar{\alpha}, \bar{\beta} \sim \text{Normal}(0,1)$$

$$\sigma \sim \text{Exponential}(1)$$

$$\text{R} \sim \text{LKJCorr}(4)$$

```
# covariance - non-centered
mCDUcov_nc <- ulam(
    alist(
        C ~ bernoulli(p),
        logit(p) <- a[D] + b[D]*U,

        # define effects using other parameters
        # this is the non-centered Cholesky machine
        transpars> vector[61]:a <<- abar[1] + v[,1],
        transpars> vector[61]:b <<- abar[2] + v[,2],
        transpars> matrix[61,2]:v <-
            compose_noncentered( sigma , L_Rho , Z ),

        # priors - note that none have parameters inside them
        # that is what makes them non-centered
        matrix[2,61]:Z ~ normal( 0 , 1 ),
        vector[2]:abar ~ normal(0,1),
        cholesky_factor_corr[2]:L_Rho ~ lkj_corr_cholesky( 4 ),
        vector[2]:sigma ~ exponential(1),

        # convert Cholesky to Corr matrix
        gq> matrix[2,2]:Rho <<- Chol_to_Corr(L_Rho)
    ) , data=dat , chains=4 , cores=4 )
```

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

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

$$\alpha = \bar{\alpha} + v_{\_,1}$$

$$\beta = \bar{\beta} + v_{\_,2}$$

$$v = \left(\text{diag}(\sigma)\mathbf{LZ}\right)^{\intercal}$$

$$\mathbf{Z}_{j,k} \sim \text{Normal}(0,1)$$

$$\bar{\alpha}, \bar{\beta} \sim \text{Normal}(0,1)$$

$$\sigma \sim \text{Exponential}(1)$$

$$\text{R} \sim \text{LKJCorr}(4)$$

```
# covariance - non-centered
mCDUcov_nc <- ulam(
    alist(
        C ~ bernoulli(p),
        logit(p) <- a[D] + b[D]*U,

        # define effects using other parameters
        # this is the non-centered Cholesky machine
        transpars> vector[61]:a <<- abar[1] + v[,1],
        transpars> vector[61]:b <<- abar[2] + v[,2],
        transpars> matrix[61,2]:v <-
            compose_noncentered( sigma , L_Rho , Z ),

        # priors - note that none have parameters inside them
        # that is what makes them non-centered
        matrix[2,61]:Z ~ normal( 0 , 1 ),
        vector[2]:abar ~ normal(0,1),
        cholesky_factor_corr[2]:L_Rho ~ lkj_corr_cholesky( 4 ),
        vector[2]:sigma ~ exponential(1),

        # convert Cholesky to Corr matrix
        gq> matrix[2,2]:Rho <<- Chol_to_Corr(L_Rho)
    ) , data=dat , chains=4 , cores=4 )
```

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

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

$$\alpha = \bar{\alpha} + v_{\_,1}$$

$$\beta = \bar{\beta} + v_{\_,2}$$

$$v = \left(\text{diag}(\sigma)\mathbf{L}\mathbf{Z}\right)^\intercal$$

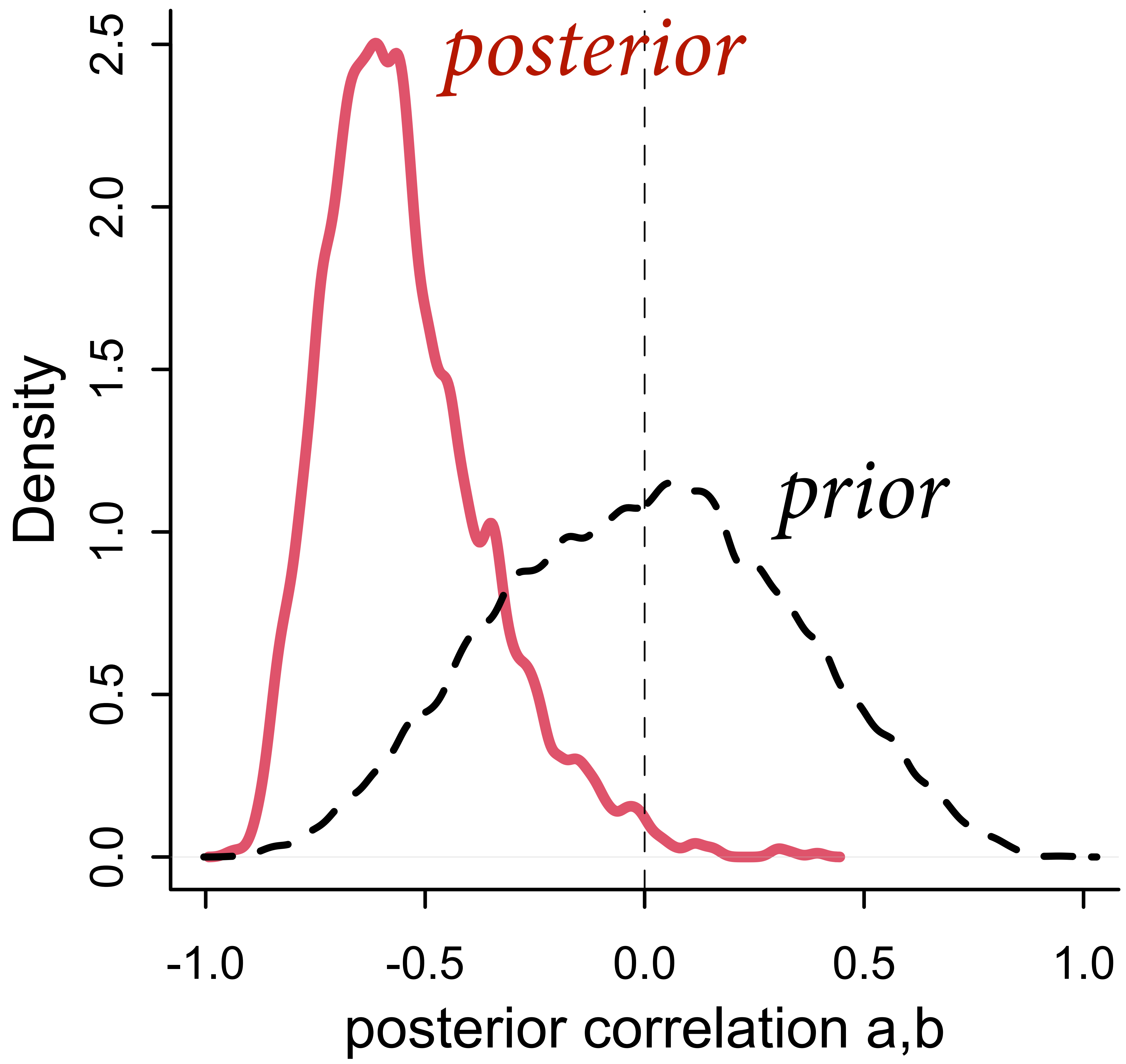
$$\mathbf{Z}_{j,k} \sim \text{Normal}(0,1)$$

$$\bar{\alpha}, \bar{\beta} \sim \text{Normal}(0,1)$$

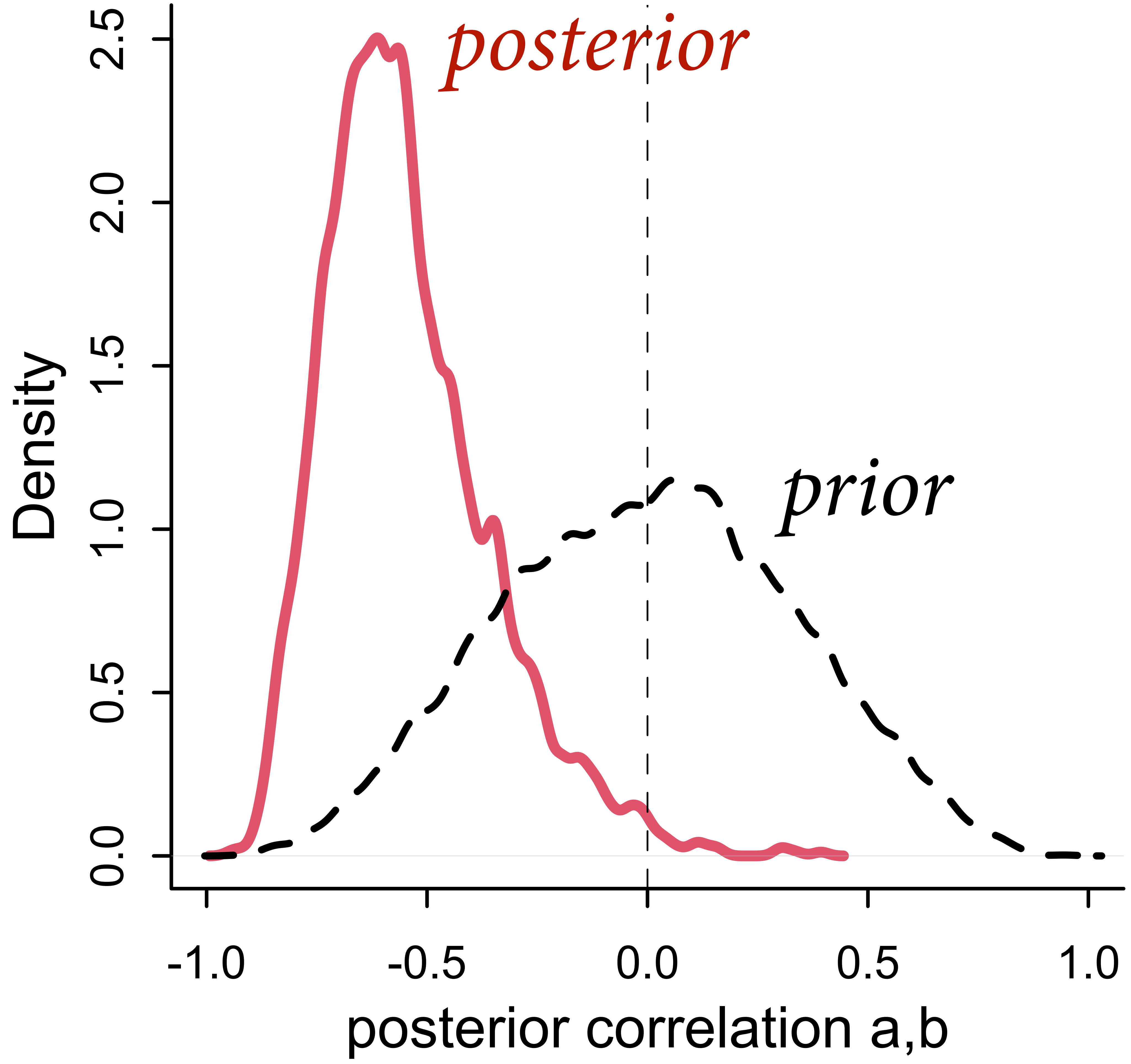
$$\sigma \sim \text{Exponential}(1)$$

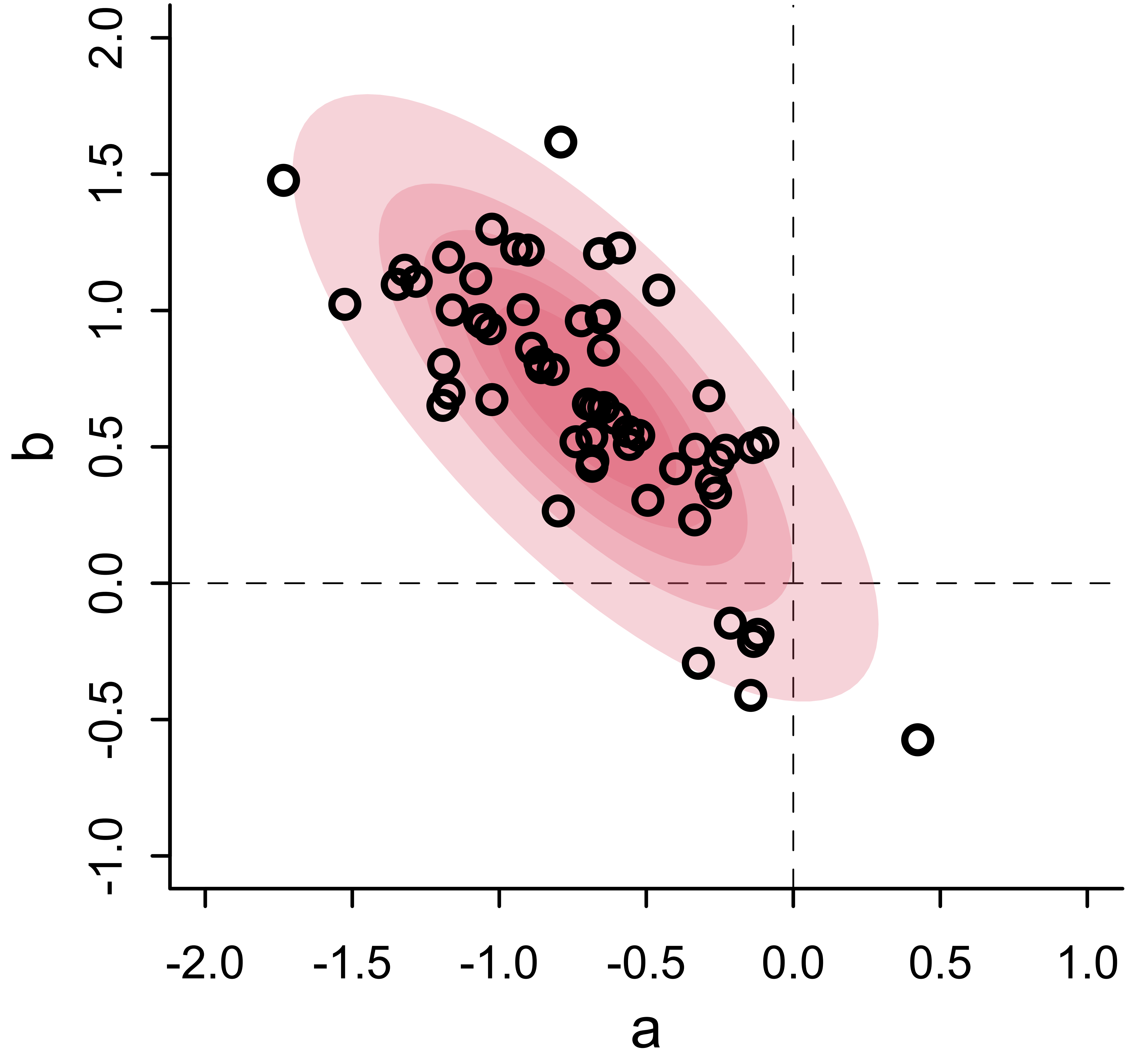
$$\boxed{\text{R} \sim \text{LKJCorr}(4)}$$

Correlated features

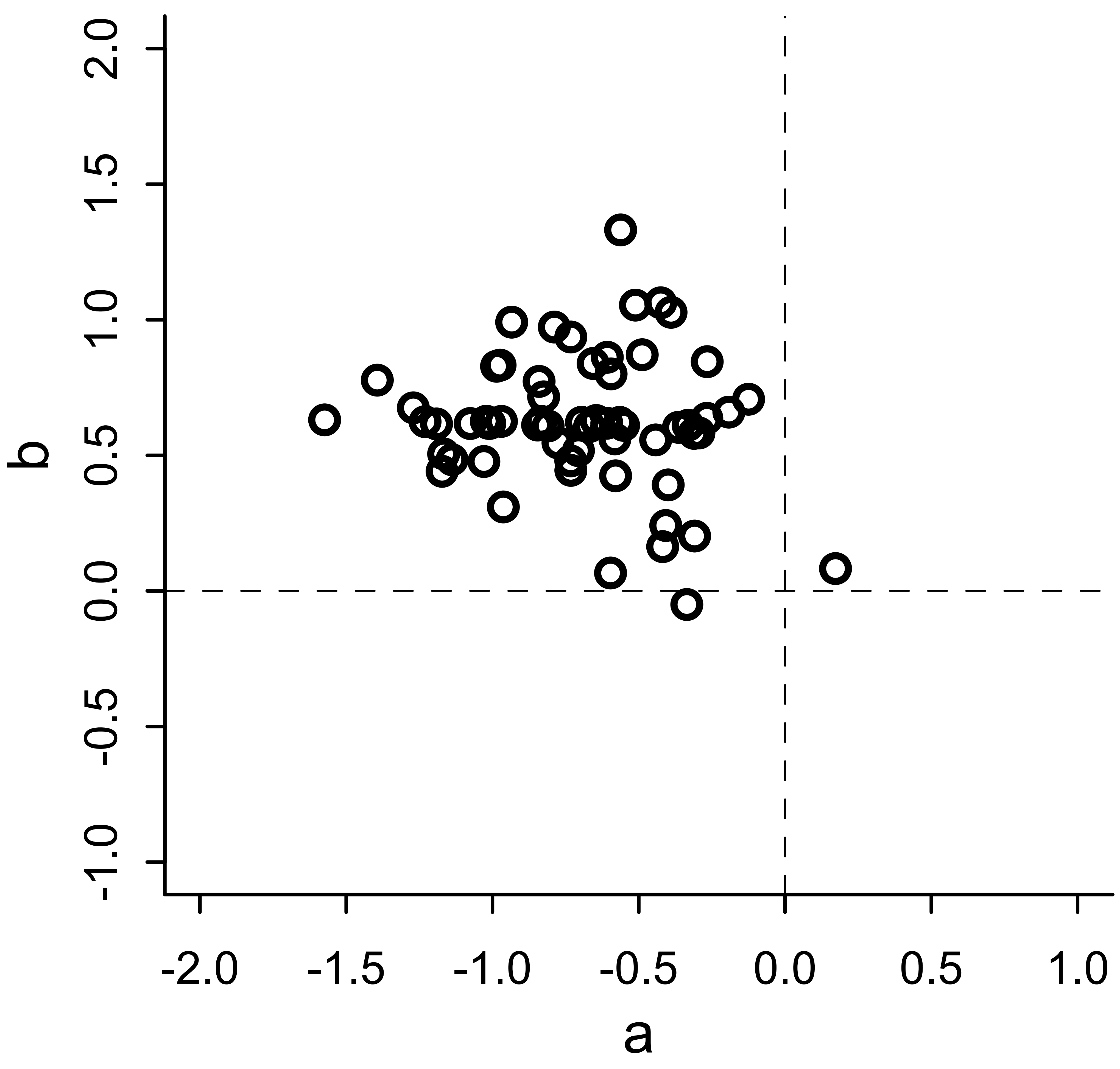


Correlated features

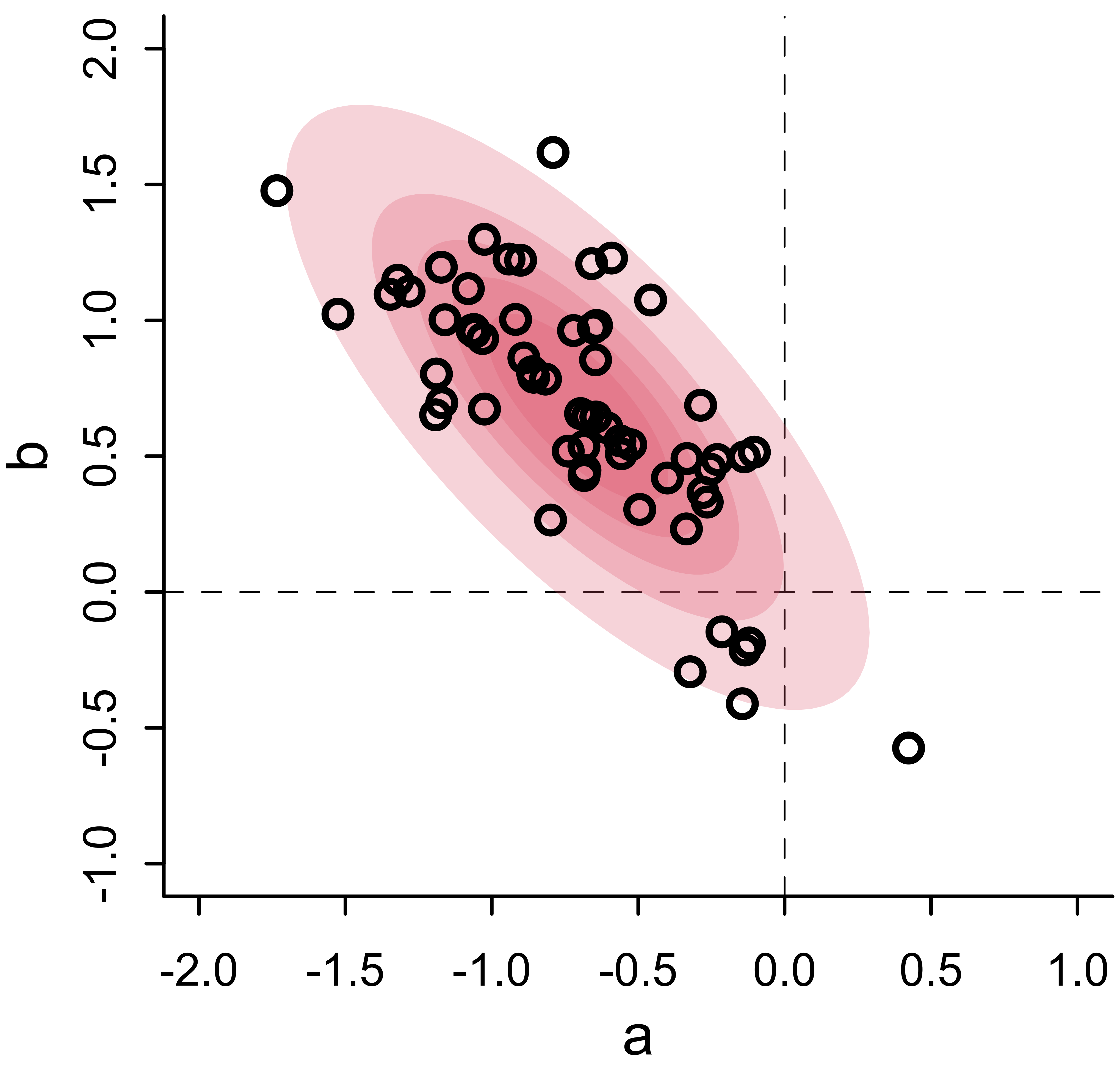


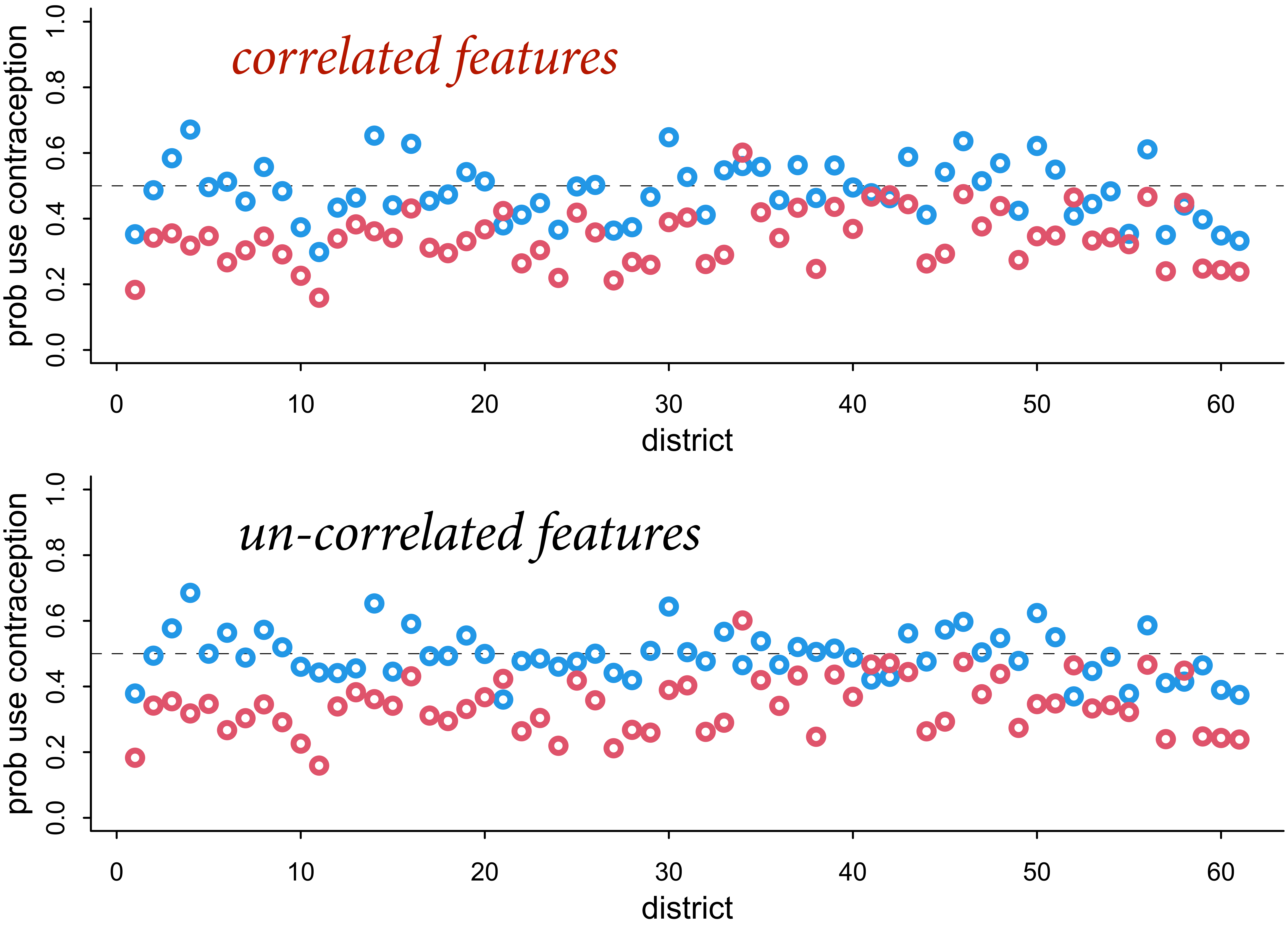


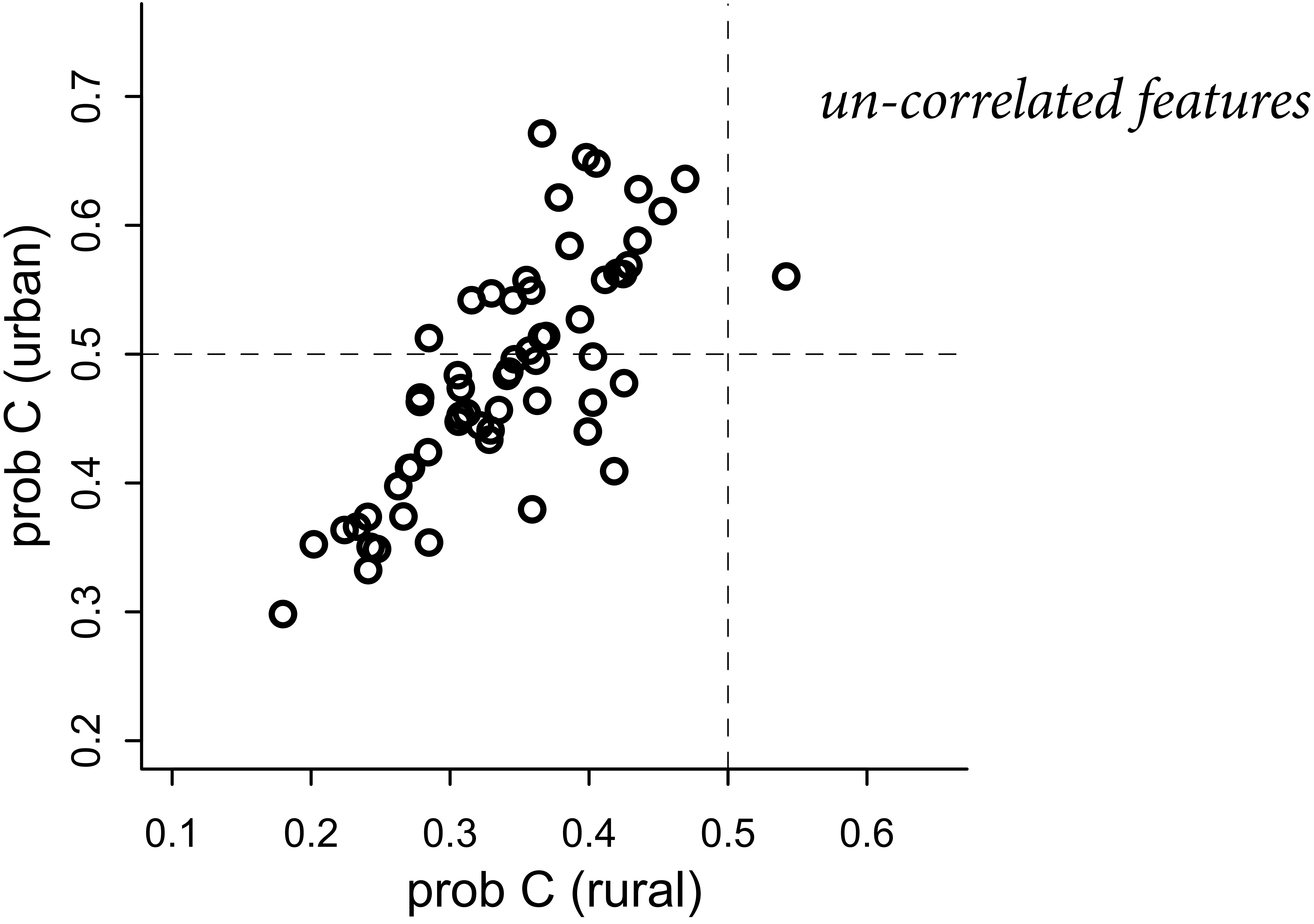
un-correlated features

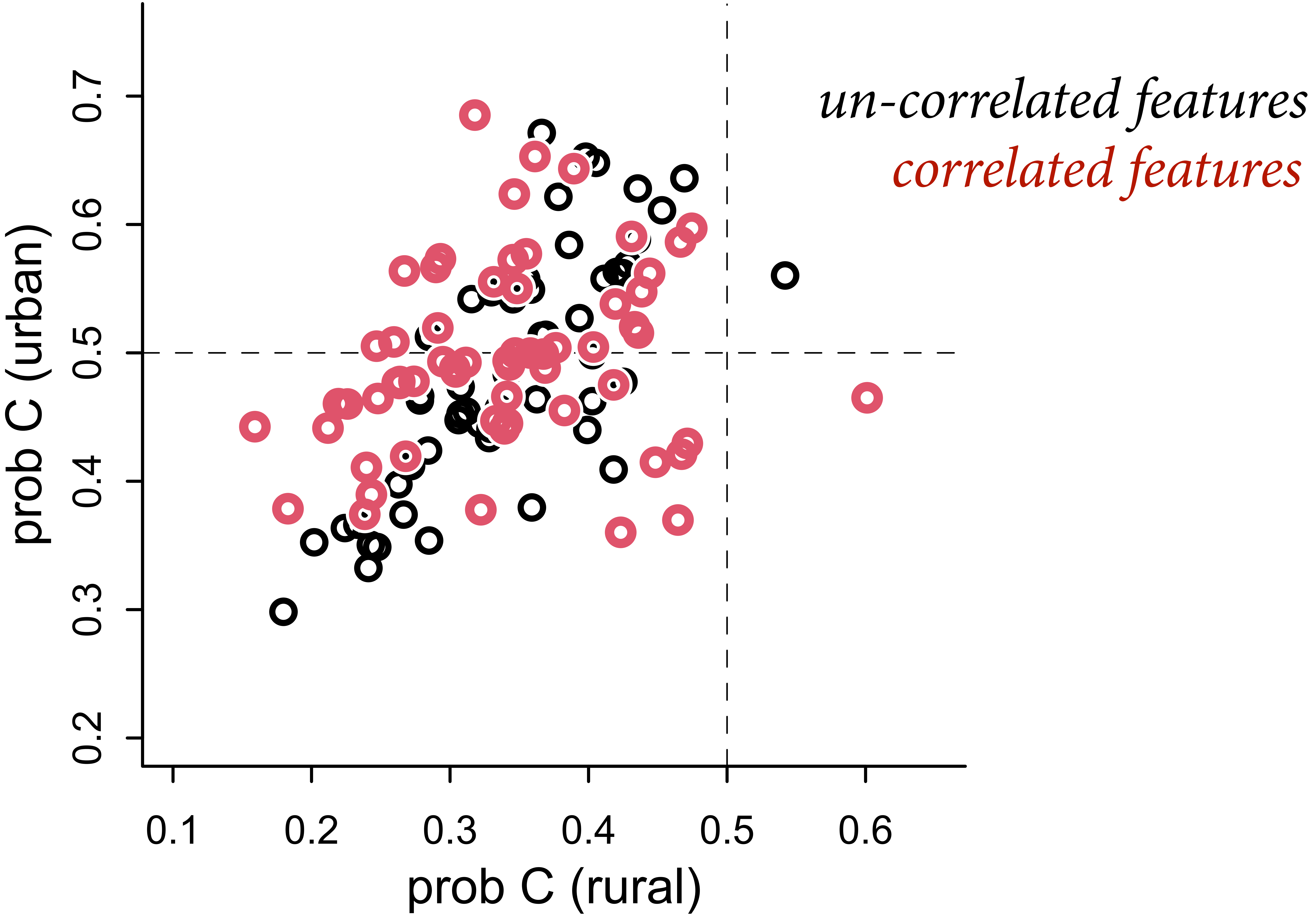


correlated features




correlated features
prob use contraception
district
un-correlated features
prob use contraception
district
0.0
0.2
0.4
0.6
0.8
1.0
0
10
20
30
40
50
60


un-correlated features
prob C (urban)
prob C (rural)
0.2
0.3
0.4
0.5
0.6
0.7
0.1
0.2
0.3
0.4
0.5
0.6



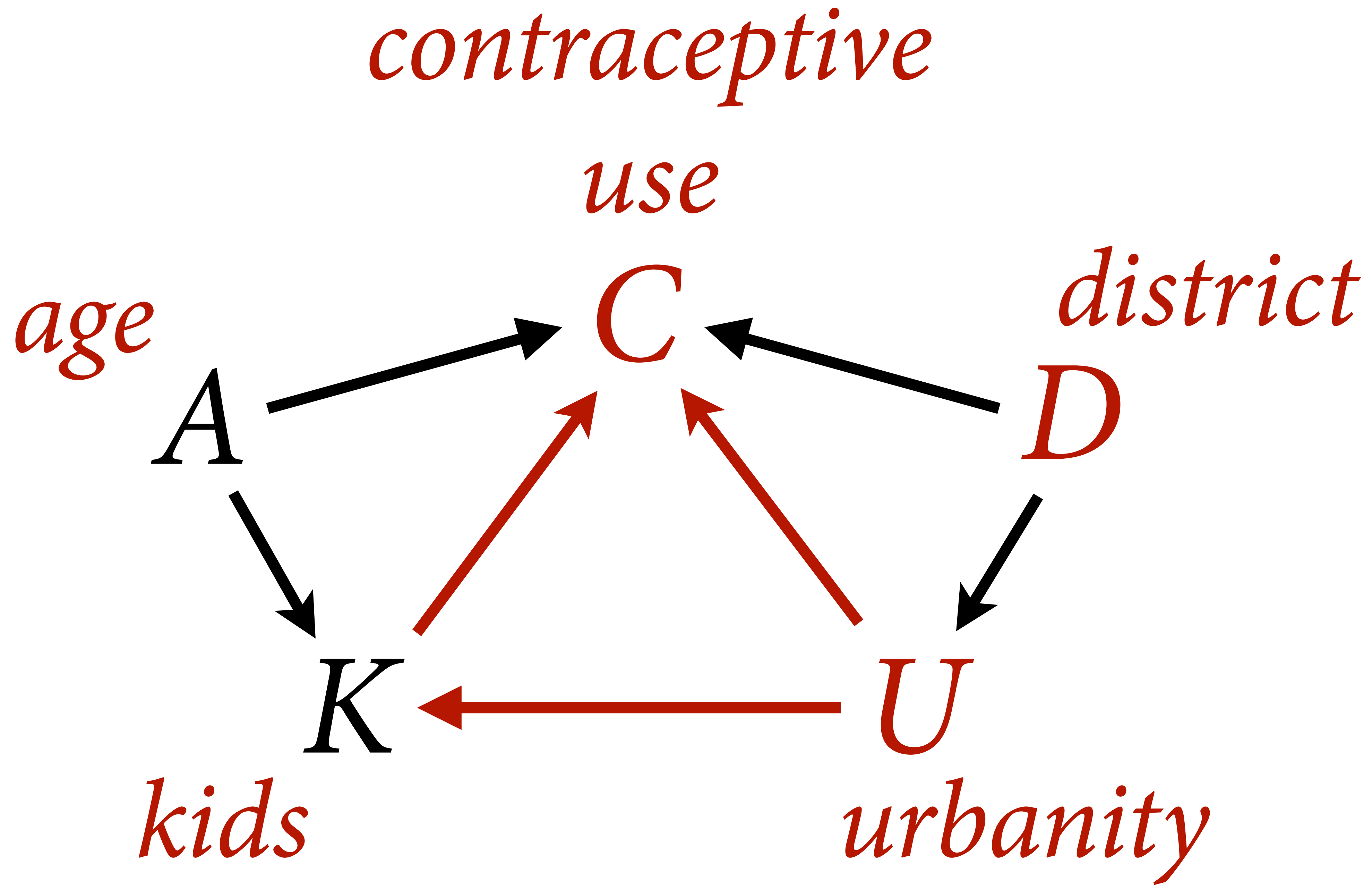
un-correlated features
correlated features
prob C (urban)
prob C (rural)
0.2
0.3
0.4
0.5
0.6
0.7
0.1
0.2
0.3
0.4
0.5
0.6

Bangladesh 1989, Party Like It's

Estimand 1: C in each district

Estimand 2: Effect of U

Estimand 3: Effects of K and A

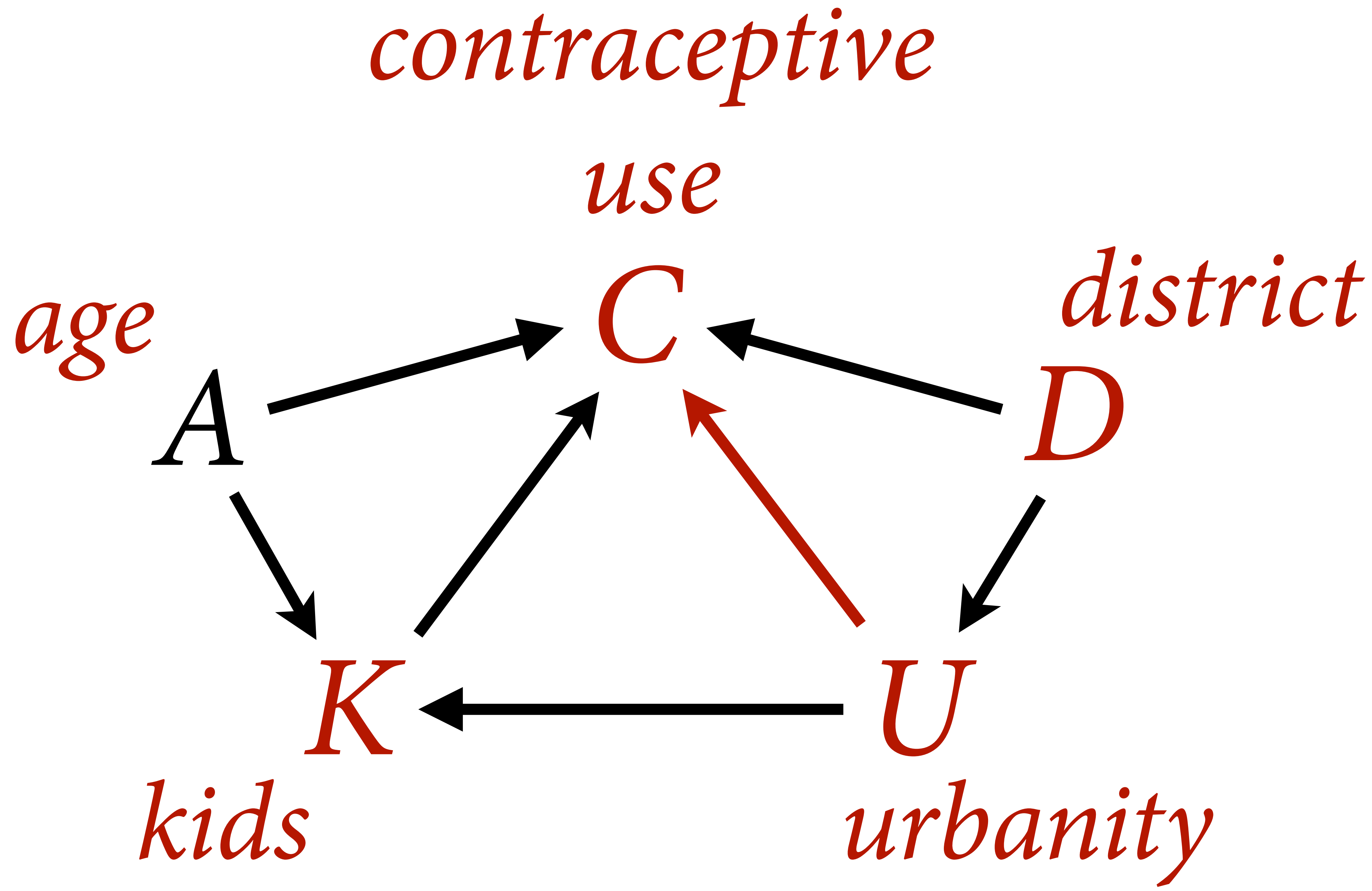


Bangladesh 1989, Party Like It's

Estimand 1: C in each district

Estimand 2: Effect of U

Estimand 3: Effects of K and A

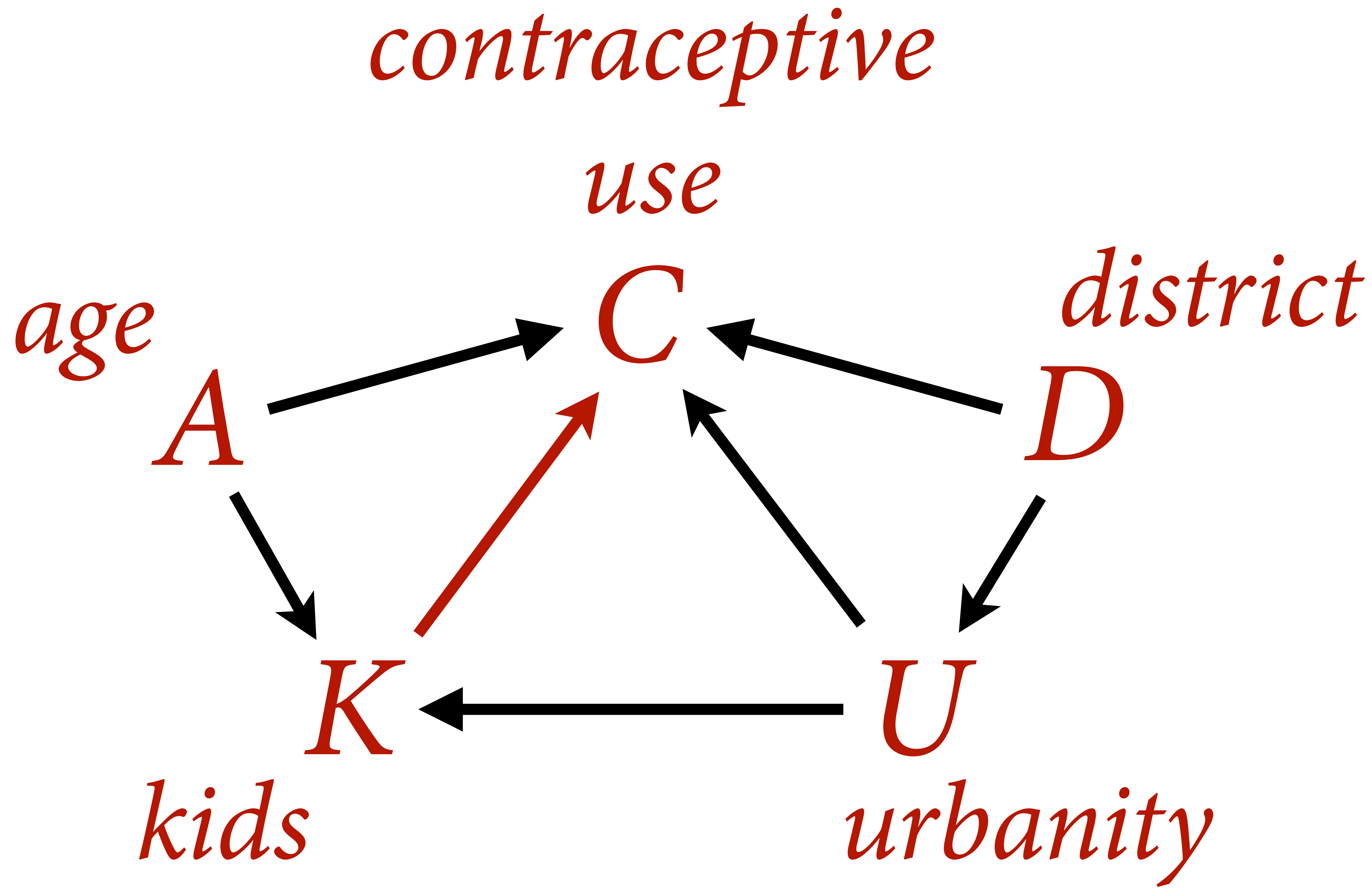


Bangladesh 1989, Party Like It's

Estimand 1: C in each district

Estimand 2: Effect of U

Estimand 3: Effects of K and A



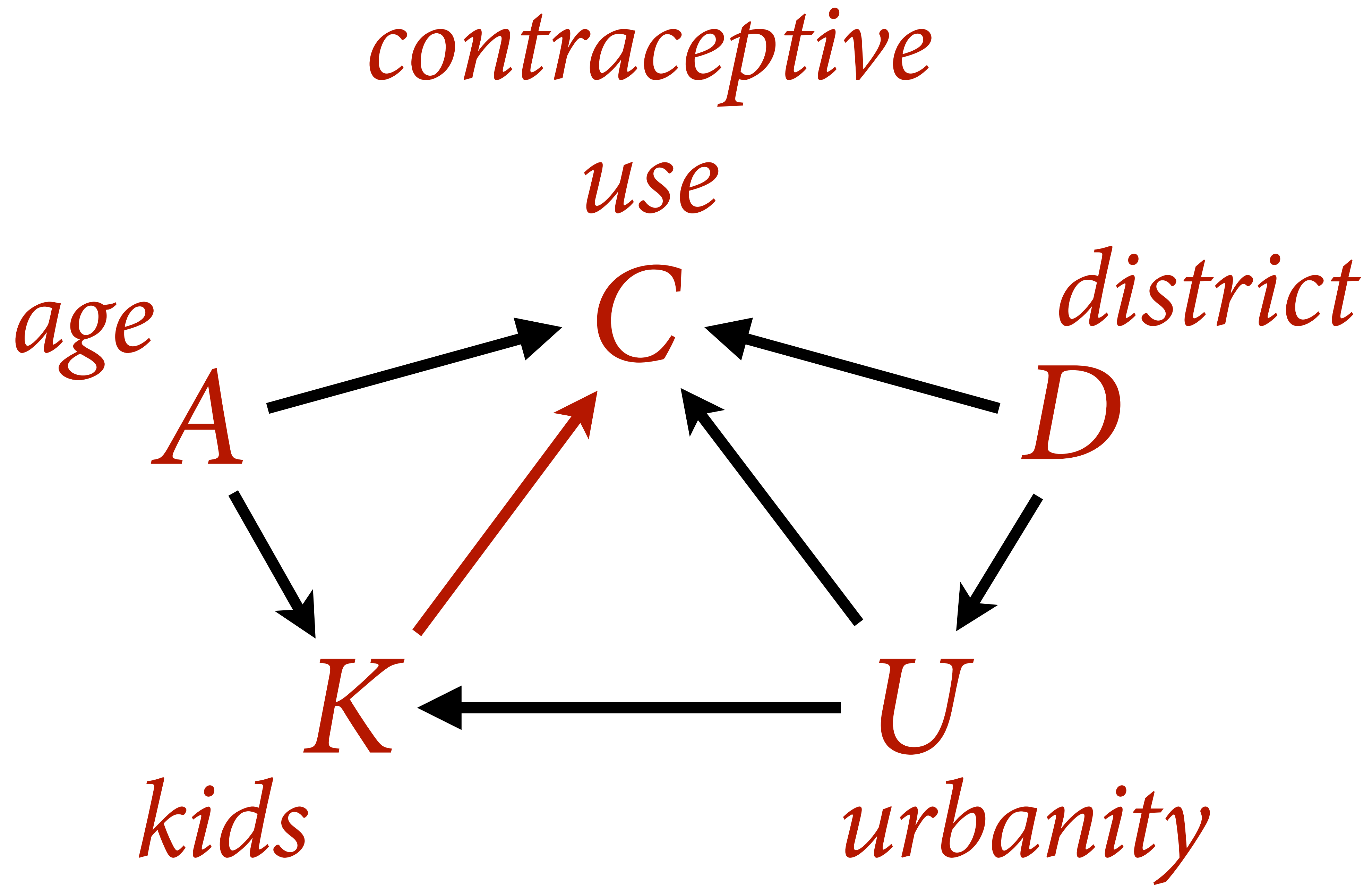
Bangladesh 1989, Party Like It's

Estimand 1: C in each district

Estimand 2: Effect of U

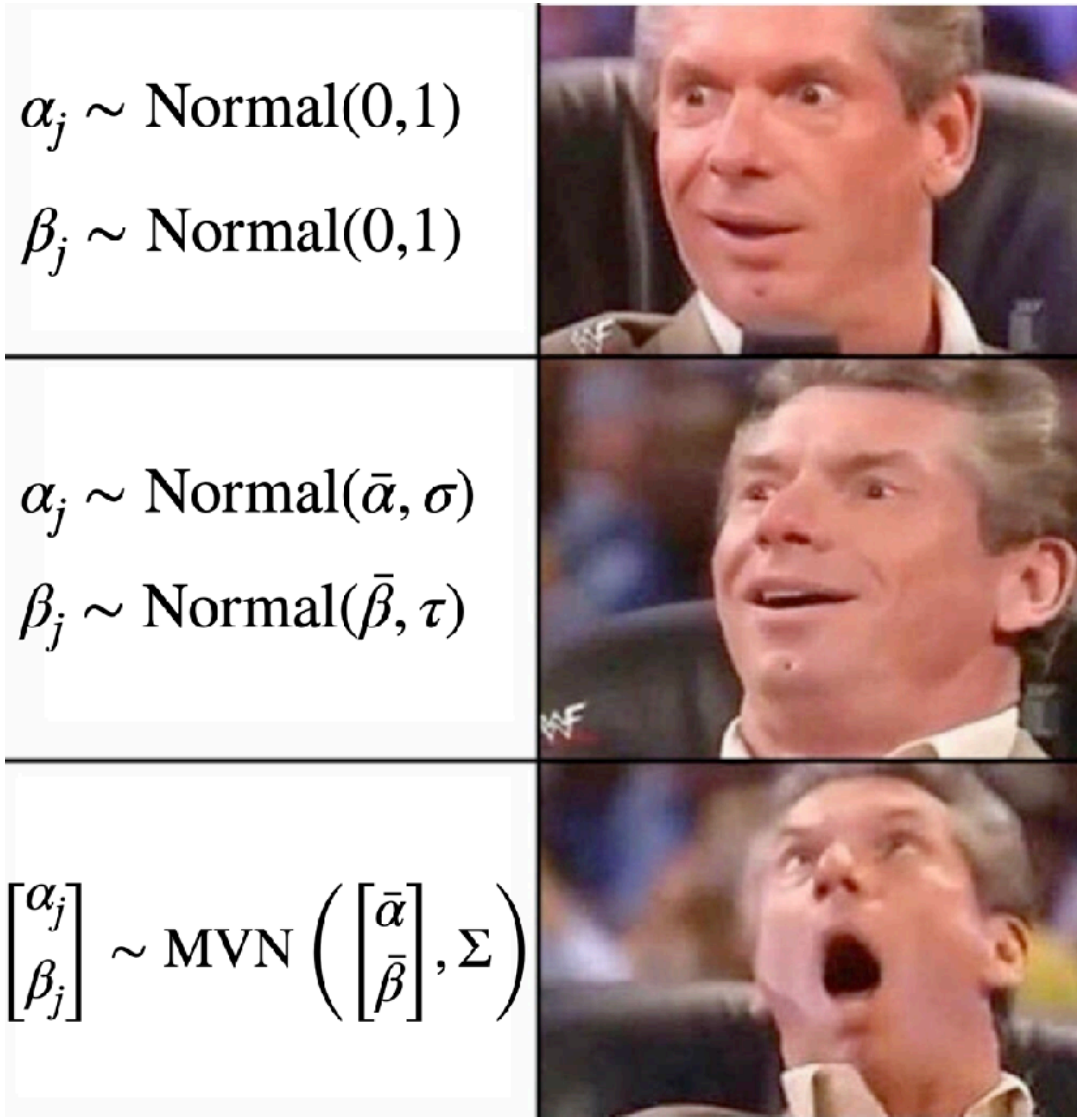
Estimand 3: Effects of K and A

Synthetic data & testing



Correlated Varying Effects

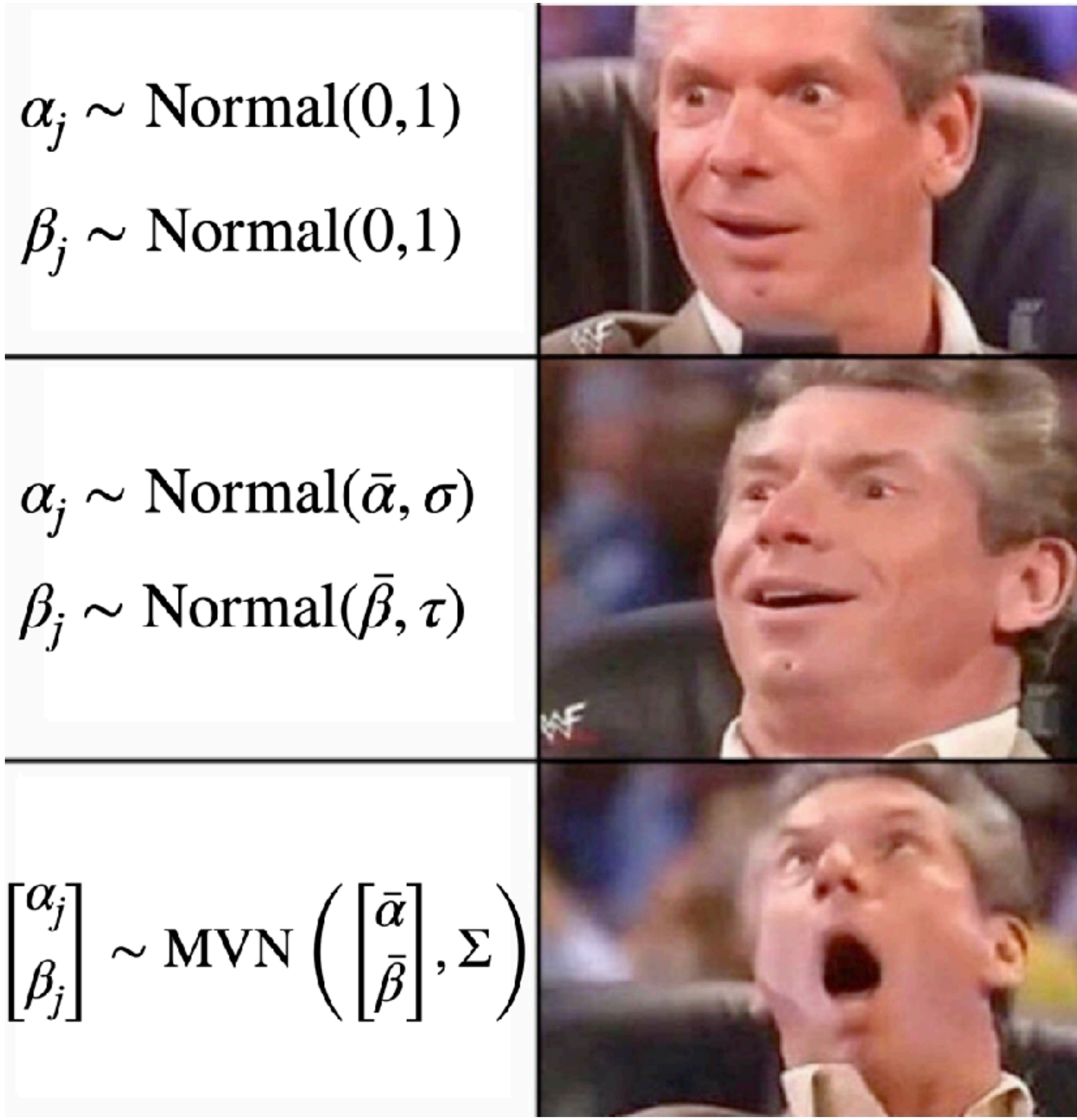
Priors that learn correlation structure:
(1) partial pooling across features
(2) learn correlations



Correlated Varying Effects

Priors that learn correlation structure:
(1) partial pooling across features
(2) learn correlations

Varying effects can be correlated even if the prior doesn't learn the correlations

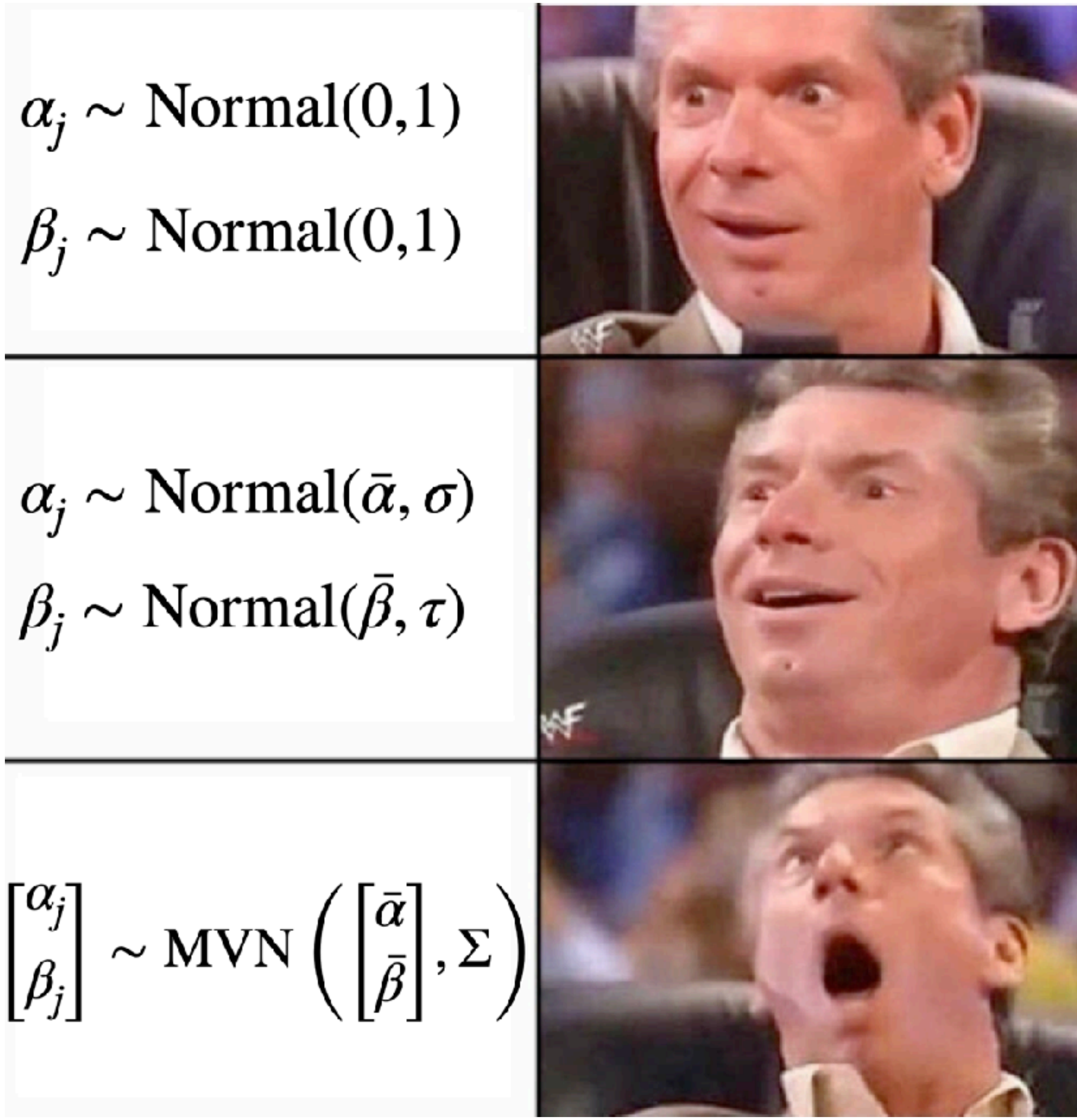


Correlated Varying Effects

Priors that learn correlation structure:
(1) partial pooling across features
(2) learn correlations

Varying effects can be correlated even if the prior doesn't learn the correlations

Ethical obligation to do our best

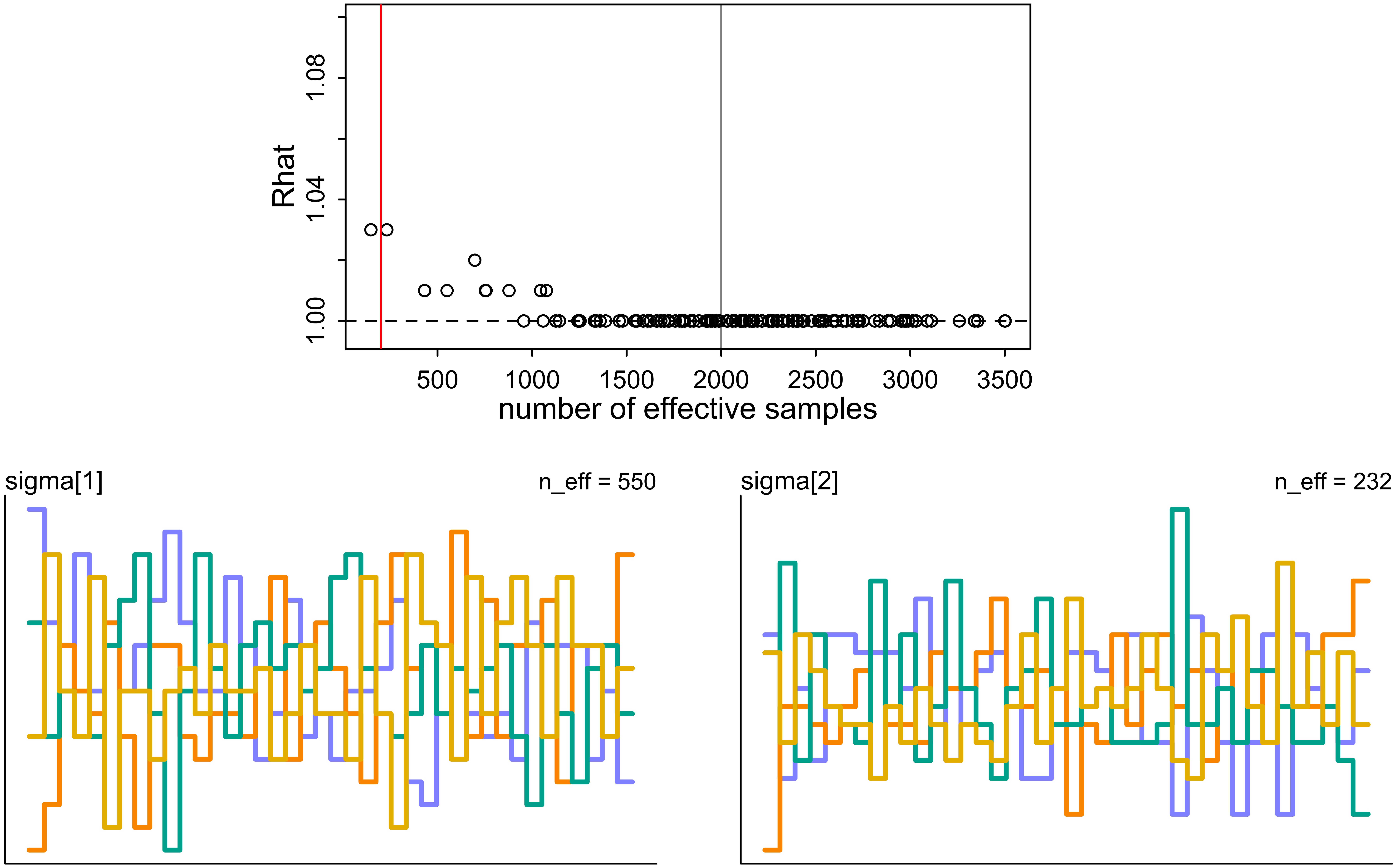


Course Schedule

Week 1	Bayesian inference	Chapters 1, 2, 3
Week 2	Linear models & Causal Inference	Chapter 4
Week 3	Causes, Confounds & Colliders	Chapters 5 & 6
Week 4	Overfitting / MCMC	Chapters 7, 8, 9
Week 5	Generalized Linear Models	Chapters 10, 11
Week 6	Ordered categories & Multilevel models	Chapters 12 & 13
Week 7	More Multilevel models	Chapters 13 & 14
Week 8	Multilevel Tactics & Gaussian Processes	Chapter 14
Week 9	Measurement & Missingness	Chapter 15
Week 10	Generalized Linear Madness	Chapter 16

`https://github.com/rmcelreath/stat_rethinking_2023`

BONUS


Rhat
1.00
1.04
1.08
500
1000
1500
2000
2500
3000
3500
number of effective samples
sigma[1]
n_eff = 550
sigma[2]
n_eff = 232

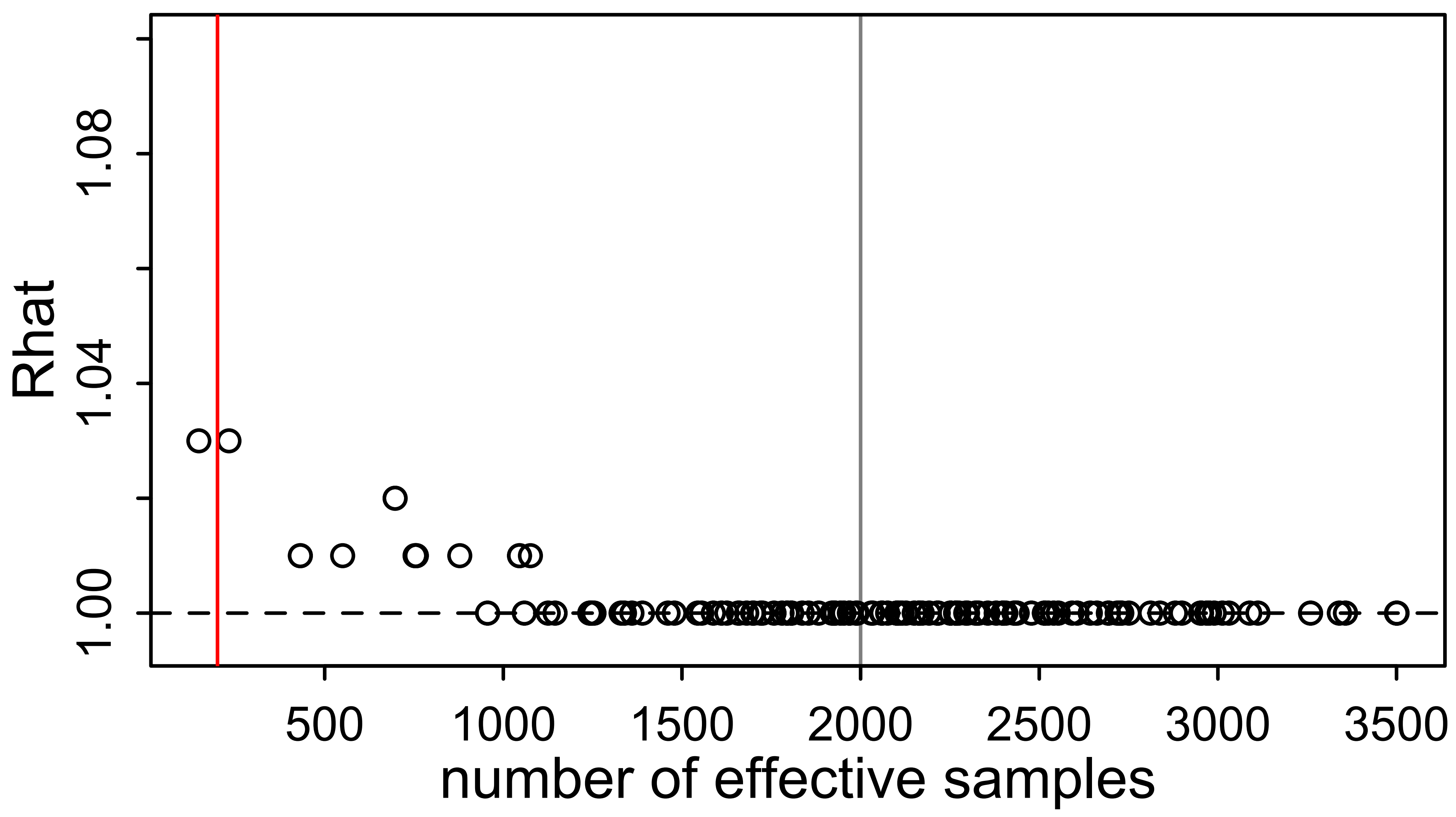
Inconvenient Posteriors

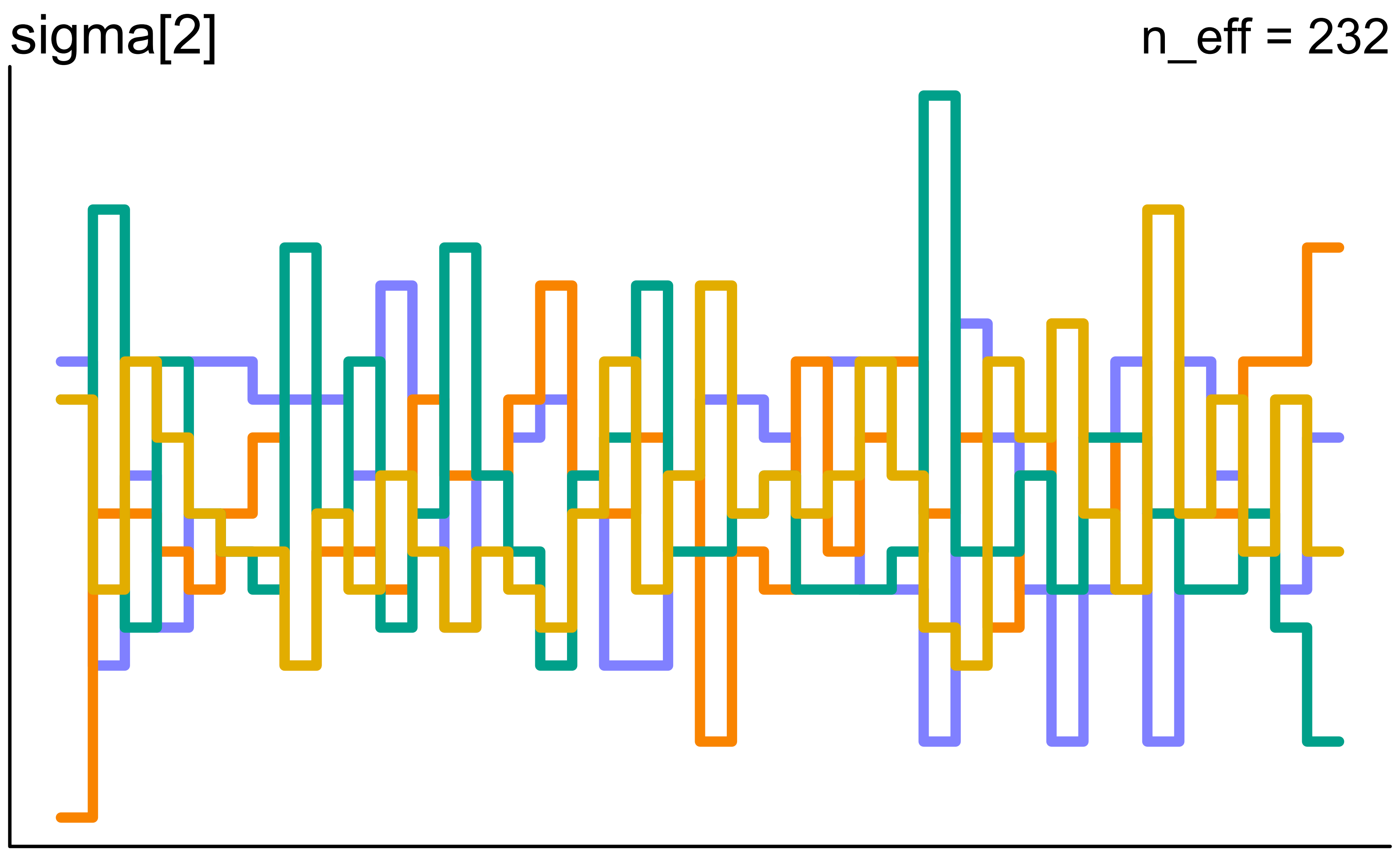
Inefficient MCMC can be caused by steep curvature

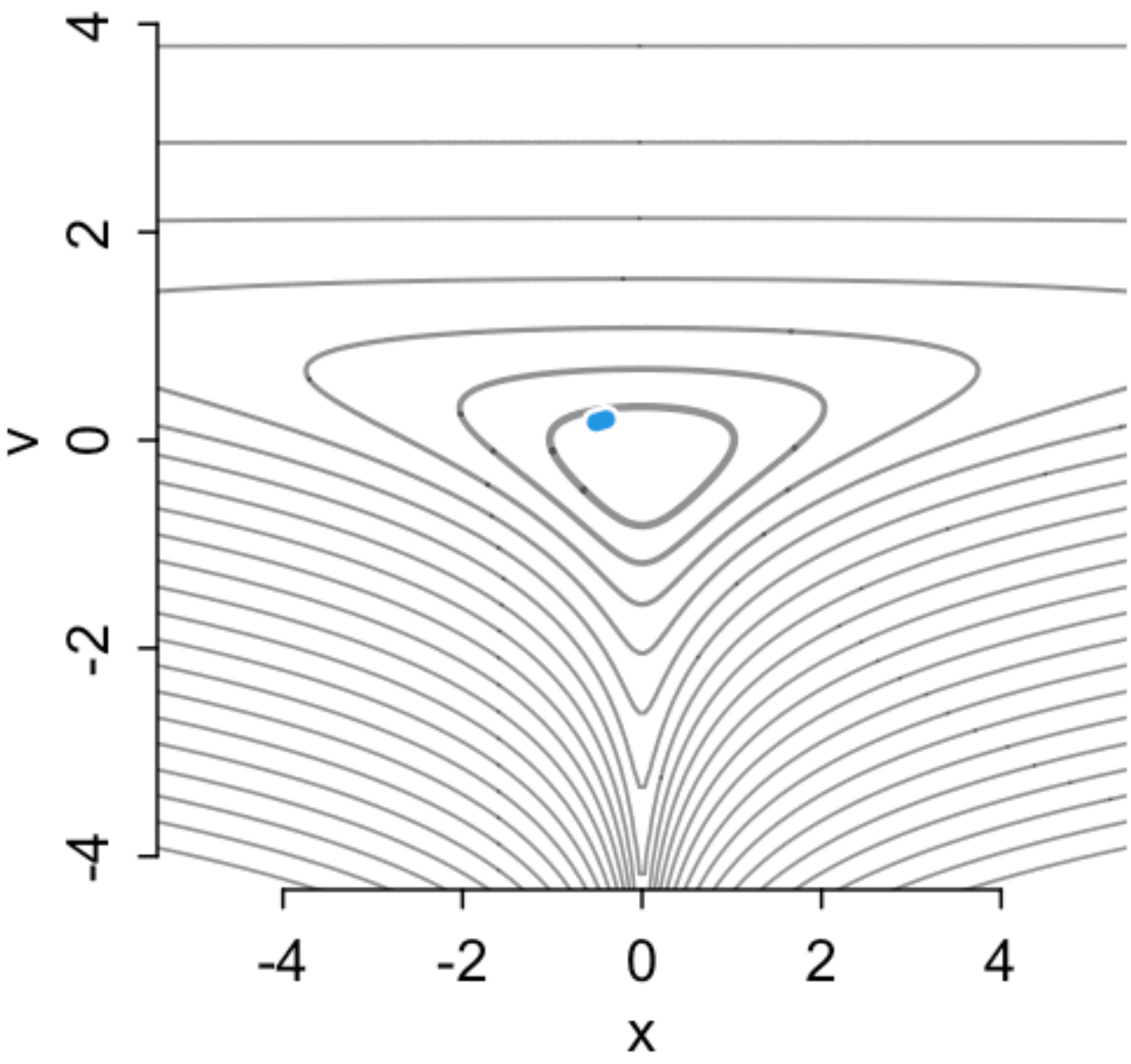
Hamiltonian simulation has trouble exploring surface

“Divergent transitions”

Transforming priors can help

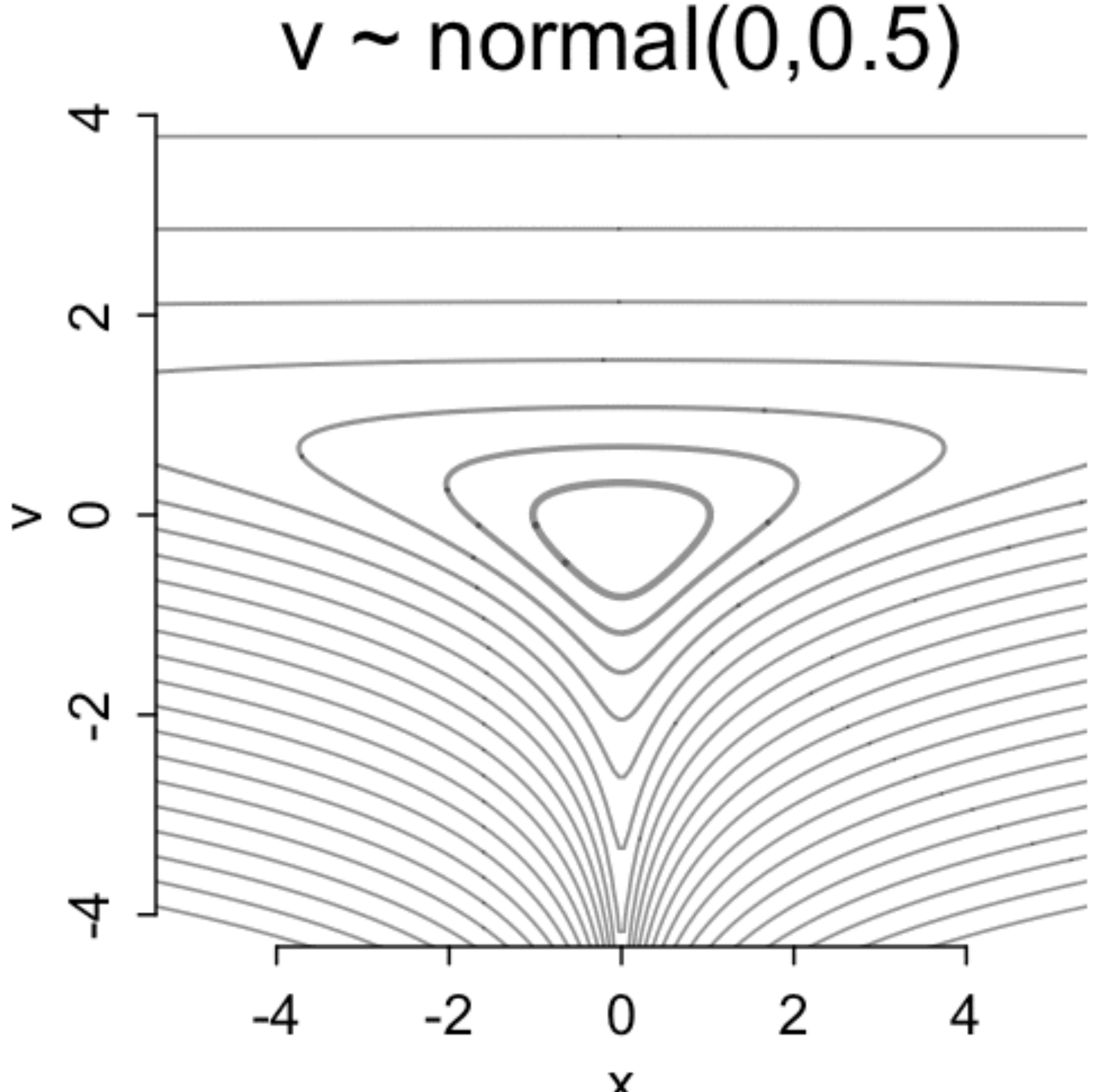




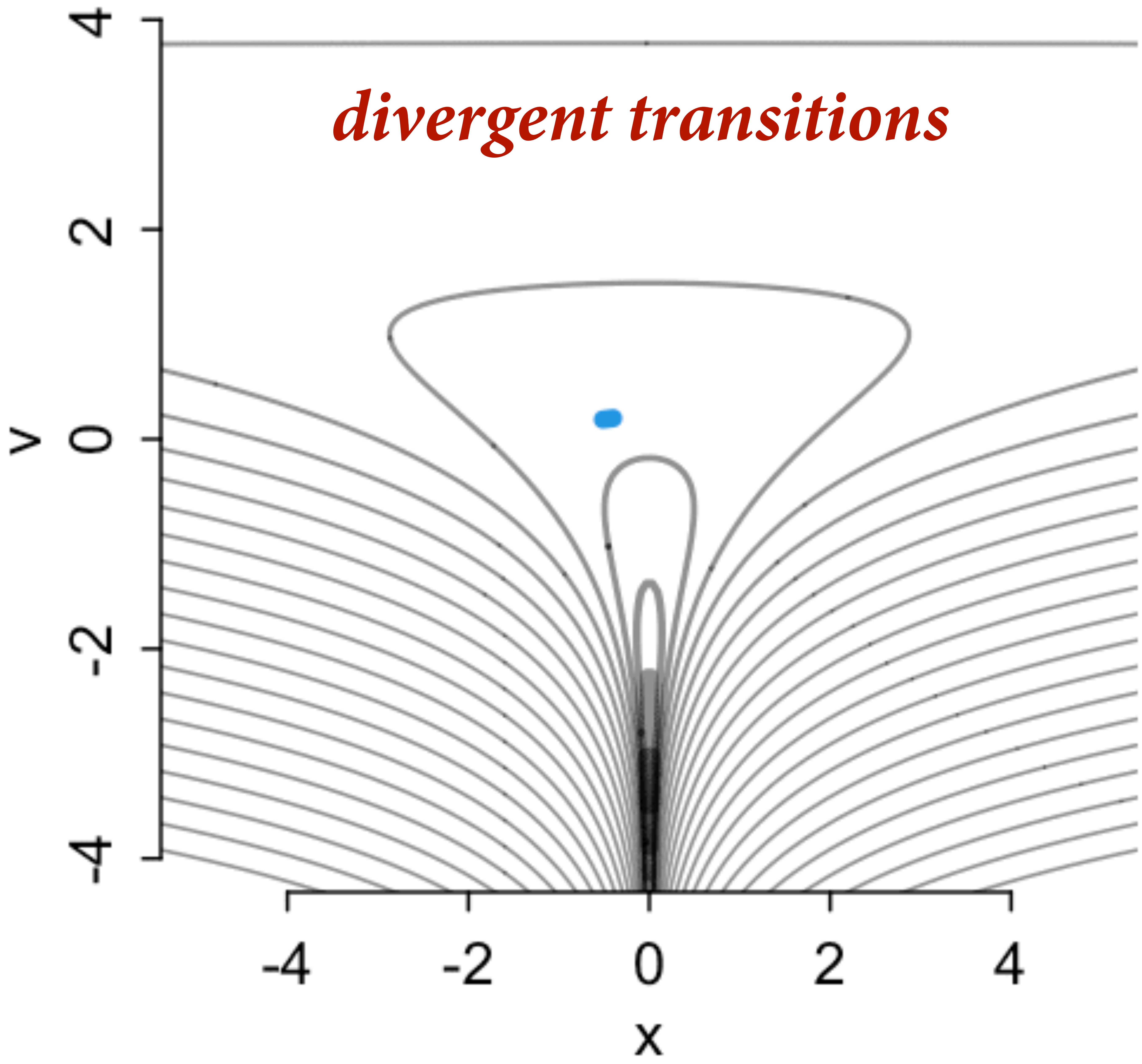


$$v \sim \text{Normal}(0, 0.5)$$

$$x \sim \text{Normal}(0, \exp(v))$$



$$v \sim \mathrm{Normal}(0,\_)$$
$$x \sim \mathrm{Normal}(0, \exp(v))$$



$$v \sim \text{Normal}(0,3)$$
$$x \sim \text{Normal}(0, \exp(v))$$

Divergent transitions

Why? Same step size not optimal everywhere

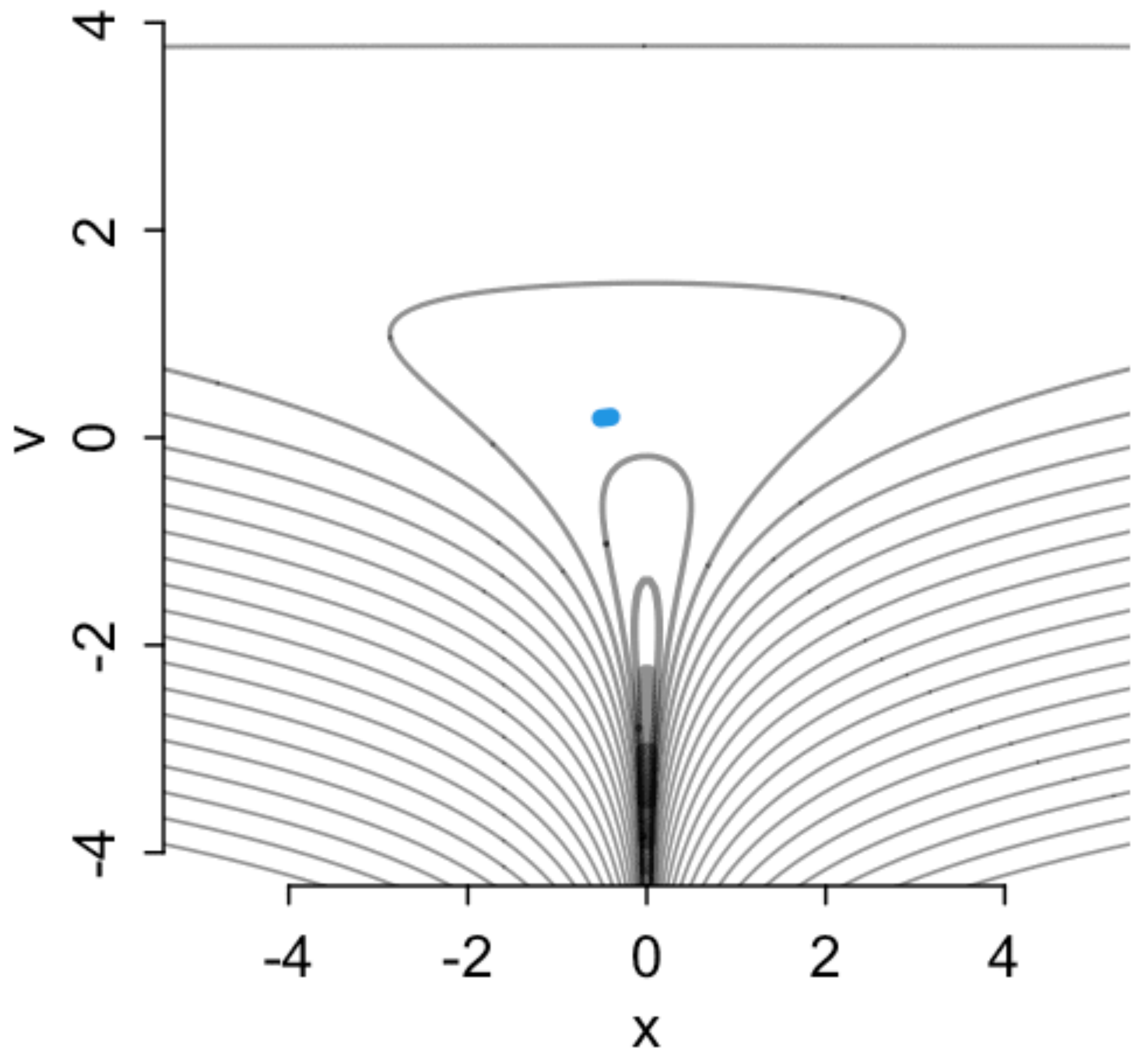
High curvature = simulation cannot follow surface

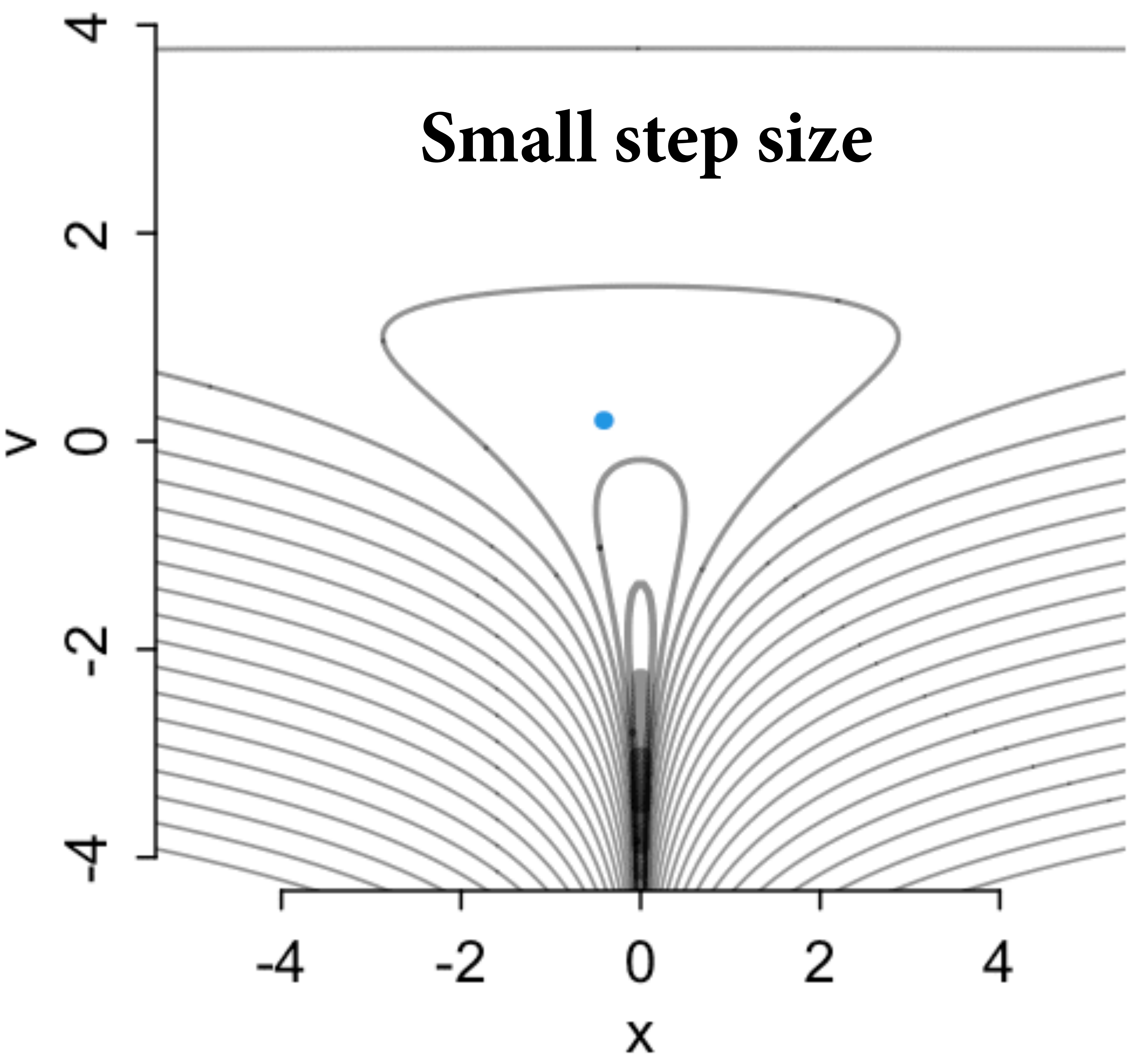
What can we do?

(1) use a smaller step size

(2) reparameterize!

$$v \sim \text{Normal}(0,3)$$
$$x \sim \text{Normal}(0, \exp(v))$$





$v \sim \mathrm{Normal}(0,3)$

$x \sim \mathrm{Normal}(0, \exp(v))$

Small step size helps, but makes exploration slow

“Centered”

$$v \sim \text{Normal}(0,3)$$

$$x \sim \text{Normal}(0, \exp(v))$$

"Centered"

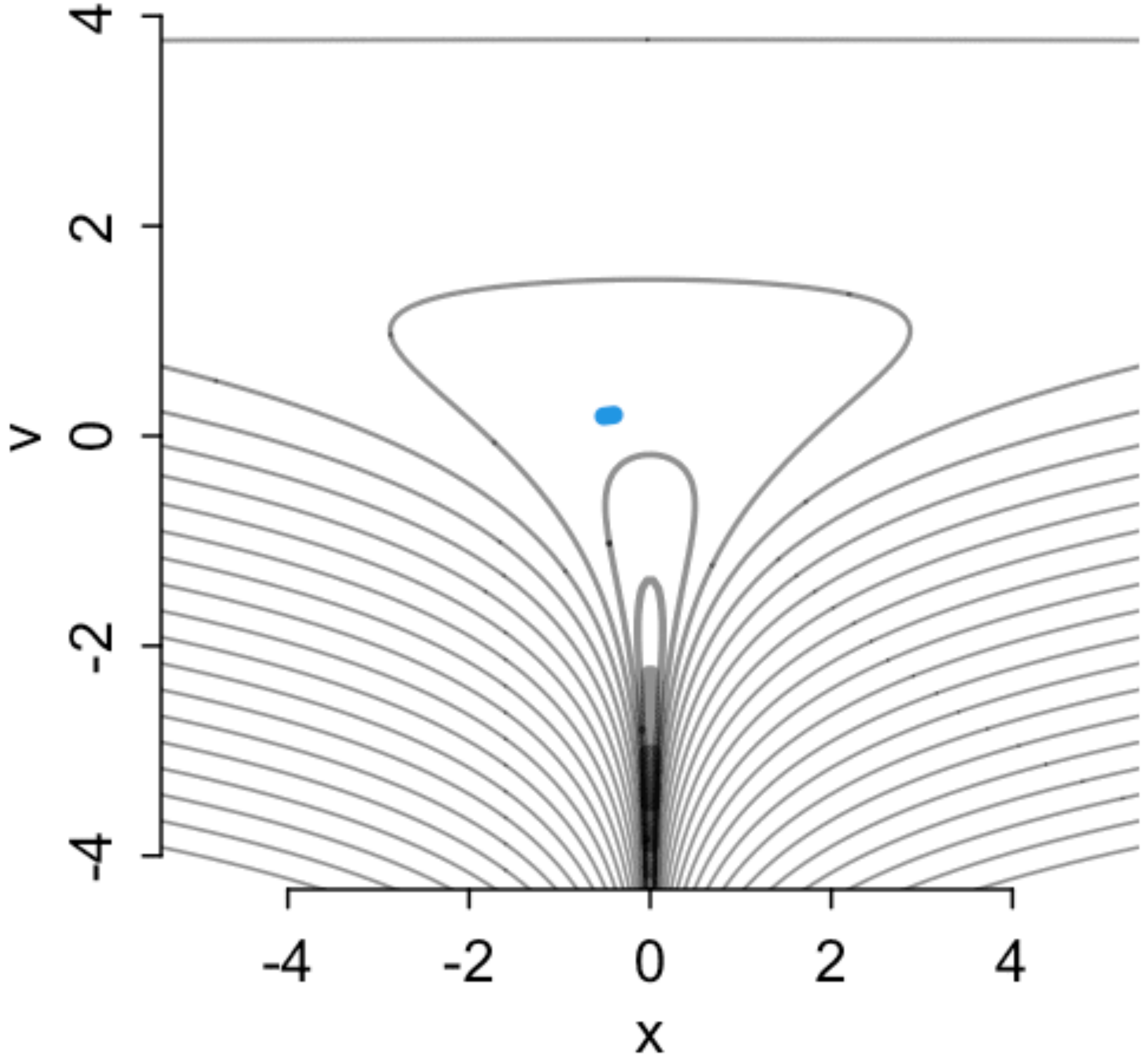
$$v \sim \text{Normal}(0,3)$$
$$x \sim \text{Normal}(0, \exp(v))$$

"Non-centered"

$$v \sim \text{Normal}(0,3)$$
$$z \sim \text{Normal}(0,1)$$
$$x = z \exp(v)$$

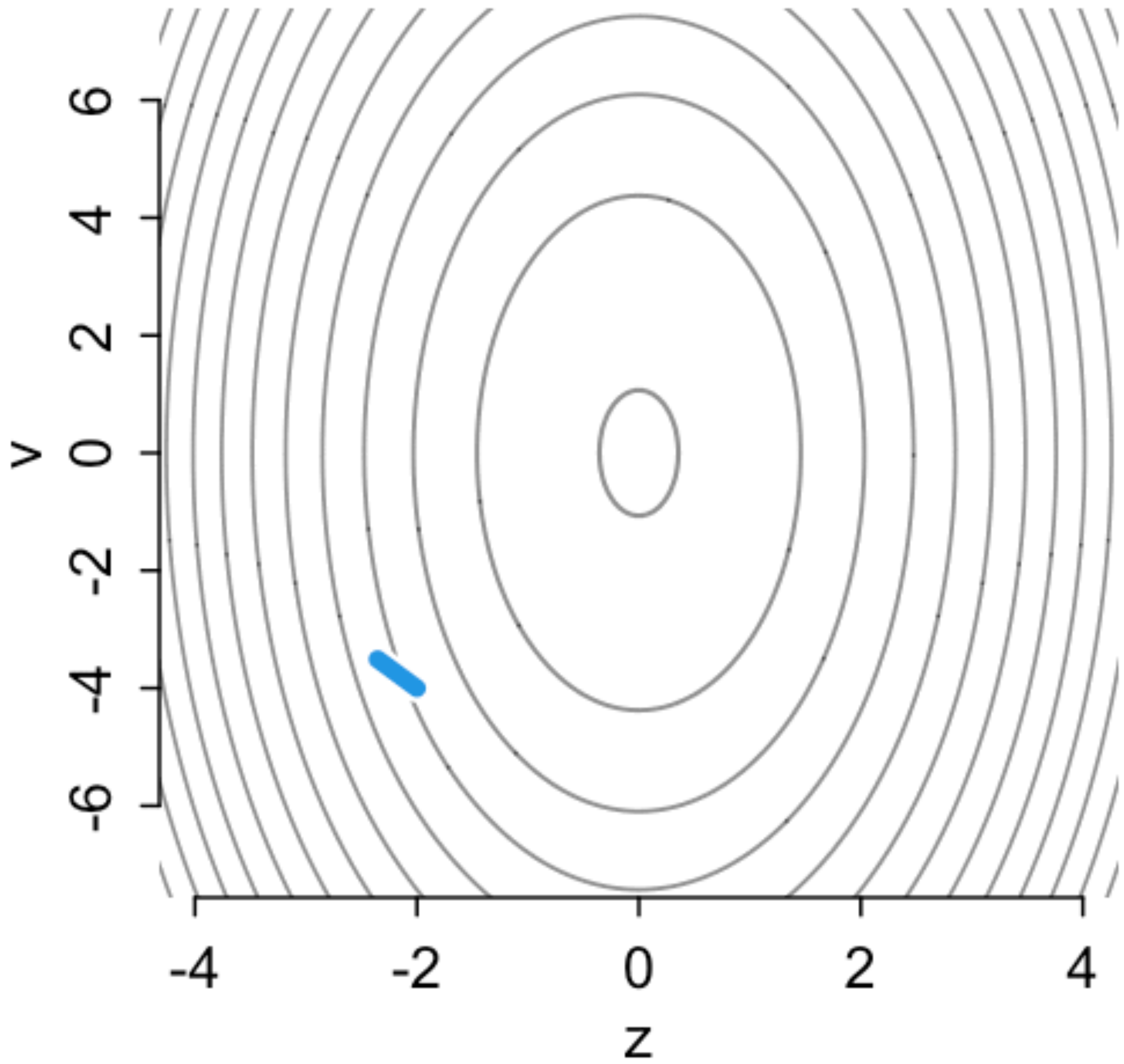
"Centered"

$$v \sim \text{Normal}(0,3)$$
$$x \sim \text{Normal}(0, \exp(v))$$



"Non-centered"

$$v \sim \text{Normal}(0,3)$$
$$z \sim \text{Normal}(0,1)$$
$$x = z \exp(v)$$

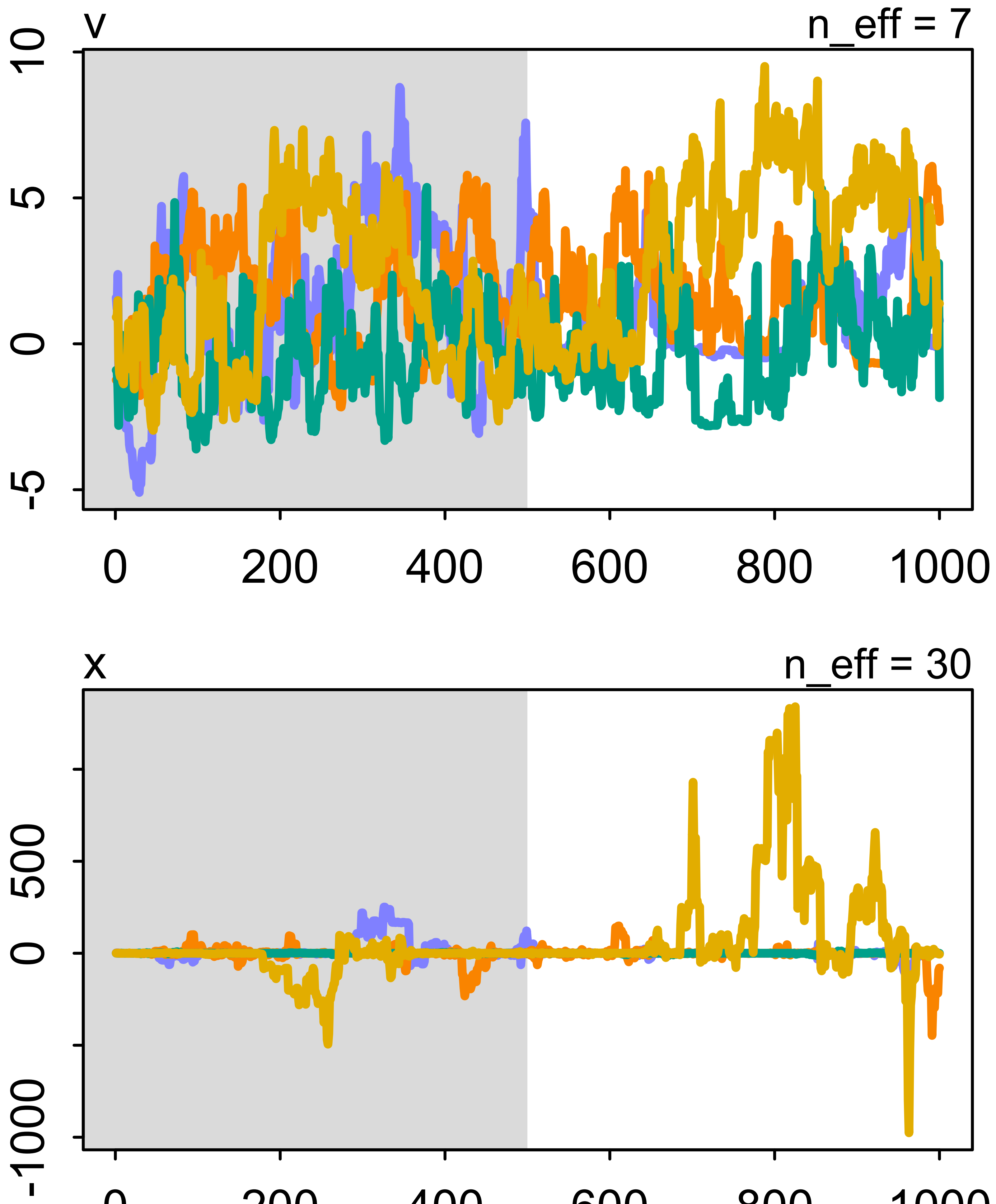


```
m13.7 <- ulam(
    alist(
        v ~ normal(0,3),
        x ~ normal(0,exp(v))
    ), data=list(N=1) , chains=4 )

m13.7nc <- ulam(
    alist(
        v ~ normal(0,3),
        z ~ normal(0,1),
        gq> real[1]:x <<- z*exp(v)
    ), data=list(N=1) , chains=4 )
```

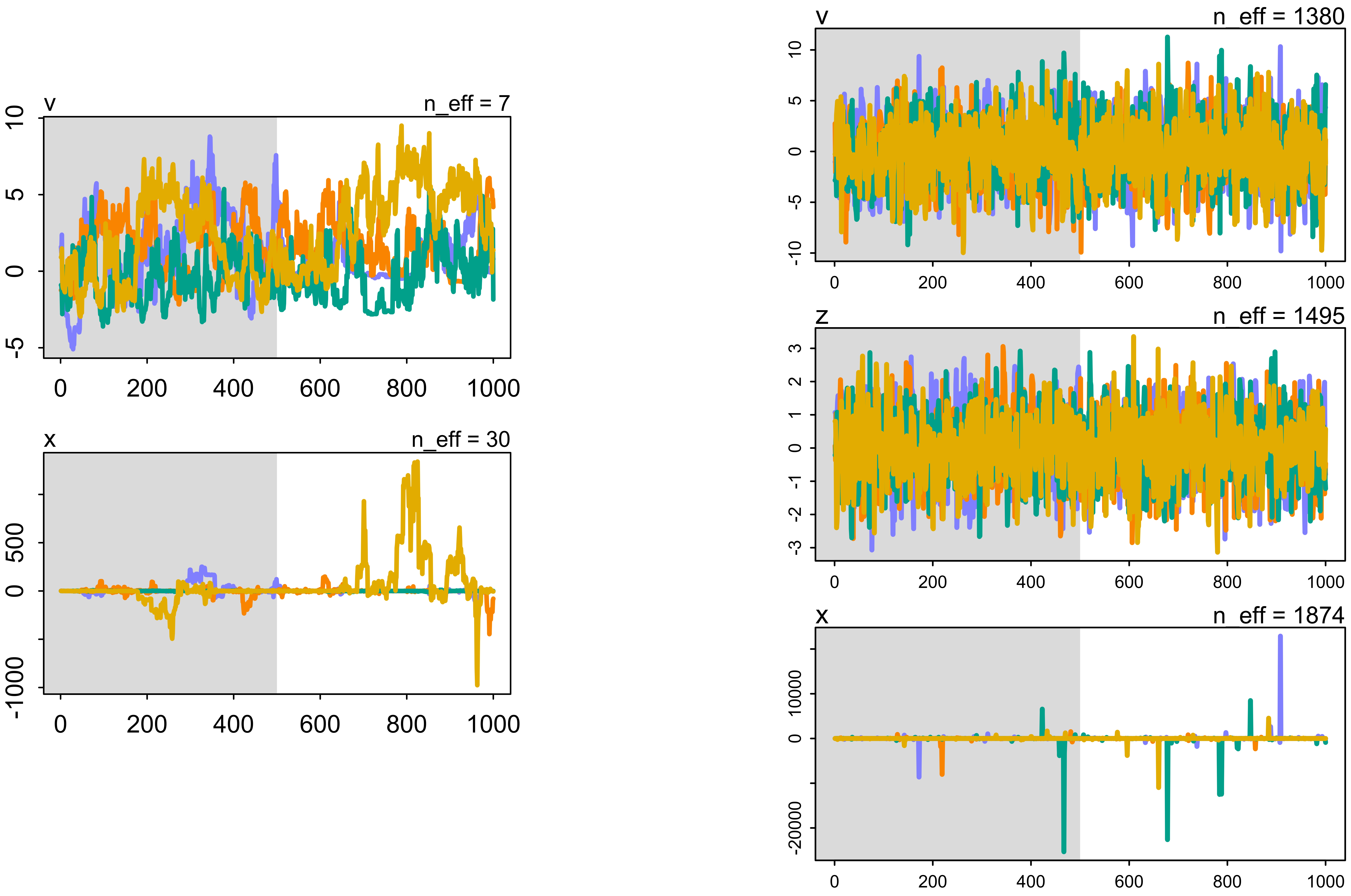
```
Warning: 112 of 2000 (6.0%) transitions ended with a divergence.
> precis( m13.7 )
   mean     sd   5.5%  94.5% n_eff Rhat4
v  1.41   2.37  -1.84   5.93     7  1.46
x 35.93 168.42 -21.15 258.86    30  1.19
```

```
m13.7 <- ulam(
    alist(
        v ~ normal(0,3),
        x ~ normal(0,exp(v))
    ), data=list(N=1) , chains=4 )

m13.7nc <- ulam(
    alist(
        v ~ normal(0,3),
        z ~ normal(0,1),
        gq> real[1]:x <<- z*exp(v)
    ), data=list(N=1) , chains=4 )
```

```
> precis( m13.7nc )
    mean     sd   5.5% 94.5% n_eff Rhat4
v  -0.04   3.12  -5.17  4.84  1380     1
z  -0.01   0.96  -1.60  1.51  1495     1
x -19.34 899.98 -30.81 24.86  1874     1
```



v
n_eff = 7
10
5
0
-5
0
200
400
600
800
1000
x
n_eff = 30
500
0
-1000
0
200
400
600
800
1000


v
n_eff = 7
x
n_eff = 30
v
n_eff = 1380
z
n_eff = 1495
x
n_eff = 1874

Centered priors

$$C_i \sim \text{Bernoulli}(p_i)$$
$$\text{logit}(p_i) = \alpha_{D[i]} + \beta_{D[i]} U_i$$
$$[\alpha_j, \beta_j] \sim \text{MVNormal}([\bar{\alpha}, \bar{\beta}], \text{R}, [\sigma, \tau])$$
$$\bar{\alpha}, \bar{\beta} \sim \text{Normal}(0,1)$$
$$\sigma, \tau \sim \text{Exponential}(1)$$
$$\text{R} \sim \text{LKJCorr}(4)$$

Non-centered priors

$$C_i \sim \text{Bernoulli}(p_i)$$
$$\text{logit}(p_i) = \alpha_{D[i]} + \beta_{D[i]} U_i$$
$$\alpha = \bar{\alpha} + v_{\_,1}$$
$$\beta = \bar{\beta} + v_{\_,2}$$
$$v = \left(\text{diag}(\sigma)\mathbf{LZ}\right)^{\intercal}$$
$$\mathbf{Z}_{j,k} \sim \text{Normal}(0,1)$$
$$\bar{\alpha}, \bar{\beta} \sim \text{Normal}(0,1)$$
$$\sigma \sim \text{Exponential}(1)$$
$$\text{R} \sim \text{LKJCorr}(4)$$

$$
\begin{aligned}
C_i &\sim \text{Bernoulli}(p_i) \\
\text{logit}(p_i) &= \alpha_{D[i]} + \beta_{D[i]} U_i \\
\alpha &= \bar{\alpha} + v_{\_,1} \\
\beta &= \bar{\beta} + v_{\_,2} \\
v &= \left(\text{diag}(\sigma)\mathbf{LZ}\right)^{\intercal} \\
\mathbf{Z}_{j,k} &\sim \text{Normal}(0,1) \\
\bar{\alpha}, \bar{\beta} &\sim \text{Normal}(0,1) \\
\sigma &\sim \text{Exponential}(1) \\
\text{R} &\sim \text{LKJCorr}(4)
\end{aligned}
$$

$$
v = \left(\begin{bmatrix} \sigma_1 & 0 \\ 0 & \sigma_2 \end{bmatrix} \mathbf{LZ} \right)^{\intercal}
$$

Matrix v is back on non-standard, correlated scale a 61-by-2 matrix

$$v = \left(\begin{bmatrix} \sigma_1 & 0 \\ 0 & \sigma_2 \end{bmatrix} \mathbf{LZ} \right)^{\mathsf{T}}$$

$$\underset{\text{a 61-by-2 matrix}}{v} = \left(\overbrace{\begin{bmatrix} \sigma_1 & 0 \\ 0 & \sigma_2 \end{bmatrix}}^{\text{diagonal matrix of standard deviations}} \mathbf{LZ} \right)^{\intercal}$$

diagonal matrix of standard deviations

a 61-by-2 matrix

$$v = \left(\begin{bmatrix} \sigma_1 & 0 \\ 0 & \sigma_2 \end{bmatrix} \mathbf{LZ} \right)^{\intercal}$$

Cholesky factor of correlation matrix across features

diagonal matrix of standard deviations

a 61-by-2 matrix

$$v = \left(\begin{bmatrix} \sigma_1 & 0 \\ 0 & \sigma_2 \end{bmatrix} \mathbf{LZ} \right)^{\mathsf{T}}$$

2-by-61 matrix of feature z-scores

Cholesky factor of correlation matrix across features

diagonal matrix of standard deviations

a 61-by-2 matrix

$$v = \left(\begin{bmatrix} \sigma_1 & 0 \\ 0 & \sigma_2 \end{bmatrix} \mathbf{L}\mathbf{Z} \right)^{\intercal}$$

transpose! flips rows and columns

2-by-61 matrix of feature z-scores

Cholesky factor of correlation matrix across features

I. — NOTICES SCIENTIFIQUES

Commandant BENOIT[1].

NOTE SUR UNE MÉTHODE DE RÉSOLUTION DES ÉQUATIONS NORMALES PROVENANT DE L'APPLICATION DE LA MÉTHODE DES MOINDRES CARRÉS A UN SYSTÈME D'ÉQUATIONS LINÉAIRES EN NOMBRE INFÉRIEUR A CELUI DES INCONNUES. — APPLICATION DE LA MÉTHODE A LA RÉSOLUTION D'UN SYSTÈME DEFINI D'ÉQUATIONS LINÉAIRES.

(Procédé du Commandant Cholesky[2].)

Le Commandant d'Artillerie Cholesky, du Service géographique de l'Armée, tué pendant la grande guerre, a imaginé, au cours de recherches sur la compensation des réseaux géodésiques, un procédé très ingénieux de résolution des équations dites *normales*, obtenues par application de la méthode des moindres carrés à des équations linéaires en nombre inférieur à celui des inconnues. Il en a conclu une méthode générale de résolution des équations linéaires.

Nous suivrons, pour la démonstration de cette méthode, la progression même qui a servi au Commandant Cholesky pour l'imaginer.

André-Louis Cholesky (1875–1918)

I. — NOTICES SCIENTIFIQUES

Commandant BENOIT[1].

NOTE SUR UNE MÉTHODE DE RÉSOLUTION DES ÉQUATIONS NORMALES PROVENANT DE L'APPLICATION DE LA MÉTHODE DES MOINDRES CARRÉS A UN SYSTÈME D'ÉQUATIONS LINÉAIRES EN NOMBRE INFÉRIEUR A CELUI DES INCONNUES. — APPLICATION DE LA MÉTHODE A LA RÉSOLUTION D'UN SYSTÈME DEFINI D'ÉQUATIONS LINÉAIRES.

(Procédé du Commandant Cholesky[2].)

Le Commandant d'Artillerie Cholesky, du Service géographique de l'Armée, tué pendant la grande guerre, a imaginé, au cours de recherches sur la compensation des réseaux géodésiques, un procédé très ingénieux de résolution des équations dites *normales*, obtenues par application de la méthode des moindres carrés à des équations linéaires en nombre inférieur à celui des inconnues. Il en a conclu une méthode générale de résolution des équations linéaires.

André-Louis Cholesky (1875–1918)

I. — NOTICES SCIENTIFIQUES

Commandant BENOIT[1].

Note on a method for solving normal equations resulting from the application of the method of least squares to a system of linear equations of smaller number than the unknowns.

D'ÉQUATIONS LINÉAIRES.

(Procédé du Commandant CHOLESKY[2].)

The artillery commander Cholesky, of the Geographical Service of the army, killed during the Great War, imagined, during research on the compensation of the geodesic networks, a very ingenious process of solving the equations known as normal, obtained by application of the method of the least squares to linear equations in lower number than that of the unknowns.

résolution des équations linéaires.

André-Louis Cholesky (1875–1918)

ques, un procédé très ingénieux de résolution des équations dites *normales*, obtenues par application de la méthode des

```r
# define 2D Gaussian with correlation 0.6
N <- 1e4
sigma1 <- 2
sigma2 <- 0.5
rho <- 0.6

# independent z-scores
z1 <- rnorm( N )
z2 <- rnorm( N )

# use Cholesky to blend in correlation
a1 <- z1 * sigma1
a2 <- ( rho*z1 + sqrt( 1-rho^2 )*z2 )*sigma2
```

ques, un procédé très ingénieux de résolution des équations dites *normales*, obtenues par application de la méthode des

```
# define 2D Gaussian with correlation 0.6
N <- 1e4
sigma1 <- 2
sigma2 <- 0.5
rho <- 0.6

# independent z-scores
z1 <- rnorm( N )
z2 <- rnorm( N )

# use Cholesky to blend in correlation
a1 <- z1 * sigma1
a2 <- ( rho*z1 + sqrt( 1-rho^2 )*z2 )*sigma2
```

ques, un procédé très ingénieux de résolution des équations dites *normales*, obtenues par application de la méthode des

```r
# define 2D Gaussian with correlation 0.6
N <- 1e4
sigma1 <- 2
sigma2 <- 0.5
rho <- 0.6

# independent z-scores
z1 <- rnorm( N )
z2 <- rnorm( N )

# use Cholesky to blend in correlation
a1 <- z1 * sigma1
a2 <- ( rho*z1 + sqrt( 1-rho^2 )*z2 )*sigma2
```

```
> cor(z1,z2)
[1] -0.0005542644
> cor(a1,a2)
[1] 0.5999334
> sd(a1)
[1] 1.997036
> sd(a2)
[1] 0.4989456
```

$$v = \left(\begin{bmatrix} \sigma_1 & 0 \\ 0 & \sigma_2 \end{bmatrix} \mathbf{LZ} \right)^\intercal$$

Transformed Priors

Both centered and non-centered better in different contexts

Centered: Lots of data in each cluster (data probability dominant)

Non-centered: Many clusters, sparse evidence (prior dominant)