Deams
Dipartimento di
Scienze Economiche, Aziendali,
Matematiche e Statistiche "Bruno de Finetti"

UNIVERSITÀ
DEGLI STUDI DI TRIESTE

Advanced topics in statistical modeling

Multilevel/Hierarchical models and extensions

L. Egidi - DEAMS, Units (legidi@units.it)

ADSAI PhD

Indice

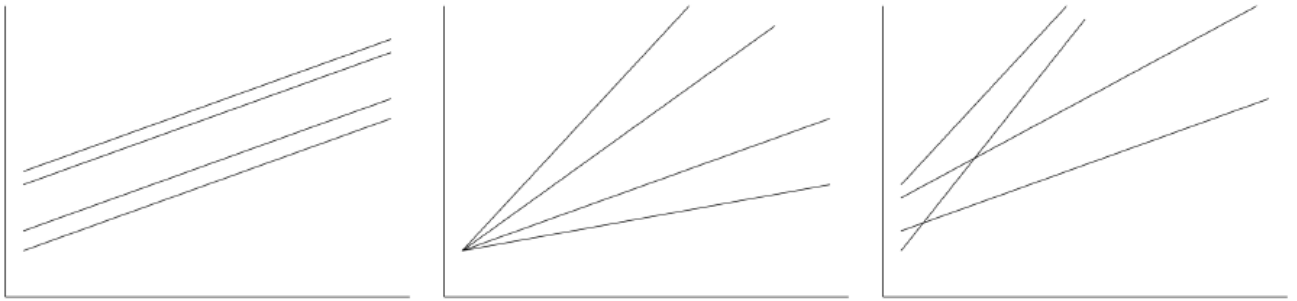
Multilevel structures

- **Hierarchical/Multilevel** models are extensions of regression in which data are structured in groups and coefficients can vary by group.
- Example of multilevel structures:
  - Simple grouped data—persons within cities—where some information is available on persons and some information is at the city level.
  - Repeated measurements.
  - Time-series cross sections.
  - Non-nested structures.

Indice

Varying-intercept and varying-slope models

- With grouped data, a regression that includes indicators for groups is called a *varying-intercept model* because it can be interpreted as a model with a different intercept within each group



(a) Varying intercept (b) Varying slope (c) Varying intercept and slope

Varying-intercept and varying-slope models (cont.)

- Model with one continuous predictor x and indicators for $J = 5$ groups. The model can be written as a regression with 6 predictors or, equivalently, as a regression with two predictors (x and the constant term), with the intercept varying by group (left figure panel):

$$y_i = \alpha_{j(i)} + \beta x_i + \epsilon_i, \quad \text{varying-intercept.}$$

- Another option (central panel) is to let the slope vary with constant intercept:

$$y_i = \alpha + \beta_{j(i)} x_i + \epsilon_i, \quad \text{varying-slope.}$$

Varying-intercept and varying-slope models (cont.)

- Finally, the right panel shows a model in which both the intercept and the slope vary by group:

$$y_i = \alpha_{j(i)} + \beta_{j(i)} x_i + \epsilon_i, \qquad \text{varying-intercept and slope.}$$

  The varying slopes are interactions between the continuous predictor x and the group indicators.
- It can be challenging to estimate all these α_j's and β_j's, especially when inputs are available at the group level.

Indice

Clustered data

- With multilevel modeling we need to go beyond the classical setup of a data vector y and a matrix of predictors X. Each level of the model can have its own matrix of predictors.
- Observational study from Gelman and Hill, (2006): effect of city-level policies on enforcing child support payments from unmarried fathers.
- The treatment is at the group (city) level, but the outcome is measured on individual families.
- To estimate the effect of child support enforcement policies, the key "treatment" predictor is a measure of enforcement policies, which is available at the city level.
- Aim: estimate the probability that the mother received informal support, given the city-level enforcement measure and other city- and individual-level predictors.

Clustered data (cont.)

ID	dad age	mom race	informal support	city ID	city name	enforce intensity	benefit level	city indicators 1	2	···	20
1	19	hisp	1	1	Oakland	0.52	1.01	1	0	···	0
2	27	black	0	1	Oakland	0.52	1.01	1	0	···	0
3	26	black	1	1	Oakland	0.52	1.01	1	0	···	0
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮		⋮
248	19	white	1	3	Baltimore	0.05	1.10	0	0	···	0
249	26	black	1	3	Baltimore	0.05	1.10	0	0	···	0
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮		⋮
1366	21	black	1	20	Norfolk	−0.11	1.08	0	0	···	1
1367	28	hisp	0	20	Norfolk	−0.11	1.08	0	0	···	1

Figure: Table 1: compact table for clustered data

Clustered data (cont.)

ID	dad age	mom race	informal support	city ID
1	19	hisp	1	1
2	27	black	0	1
3	26	black	1	1
⋮	⋮	⋮	⋮	⋮
248	19	white	1	3
249	26	black	1	3
⋮	⋮	⋮	⋮	⋮
1366	21	black	1	20
1367	28	hisp	0	20

city ID	city name	enforce-ment	benefit level
1	Oakland	0.52	1.01
2	Austin	0.00	0.75
3	Baltimore	−0.05	1.10
⋮	⋮	⋮	⋮
20	Norfolk	−0.11	1.08

Figure: Table 2: two data-matrices for clustered data

- First table: data for the analysis as it might be stored in a computer package, with information on each of the 1367 mothers surveyed.

Clustered data (cont.)

- Second table: to make use of the **multilevel structure** of the data, however, we need to construct two data matrices, one for each level of the model (city and mothers).
- Conceptually, the two-matrix, or multilevel, data structure has the advantage of clearly showing which information is available on individuals and which on cities.
- It also gives more flexibility in fitting models, allowing us to move beyond the classical regression framework.

Clustered data (cont.)

We briefly outline several possible ways of analyzing these data, as a motivation and lead-in to multilevel modeling.

- *Individual-level regression*: $\Pr(Y_i = 1) = \text{logit}^{-1}(X_i\beta)$ where X includes the constant term, the treatment (enforcement intensity), and the other predictors (father's age and indicators for mother's race at the individual level; and benefit level at the city level). X is thus constructed from the data matrix of Table 1.

  Problem: it ignores city-level variation beyond that explained by enforcement intensity and benefit level,which are the city-level predictors in the model.

- *Group-level regression on city averages*: perform a city-level analysis, with individual-level predictors included using their group-level averages. The outcome, y_j, would be the average total support among the respondents in city j, the enforcement indicator would be the treatment, and the other variables would also be included as predictors. Such a regression—in this case, with 20 data points—has the advantage that its errors are automatically at the city level.

  Problem: however, by aggregating, it removes the ability of individual predictors to predict individual outcomes.

Clustered data (cont.)

- *Individual-level regression with city indicators, followed by group-level regression of the estimated city effects*: two-steps analysis, first fitting a logistic regression to the individual data y given individual predictors (in this example, father's age and indicators for mother's race) along with indicators for the 20 cities. Then, the next step is to perform a linear regression at the city level, considering the estimated coefficients of the city indicators (in the individual model that was just fit) as the "data" y_j . This city-level regression has 20 data points and uses, as predictors, the city-level data (in this case, enforcement intensity and benefit level).

  Problem: can run into problems when sample sizes are small in particular groups, or when there are interactions between individual- and group-level predictors.

Clustered data (cont.)

Multilevel modeling is a more general approach that can include predictors at both levels at once.

- The multilevel model looks something like the two-step model we have described, except that both steps are fitted at once.
- Two components: a logistic regression with 1369 data points predicting the binary outcome given individual-level predictors and with an intercept that can vary by city, and a linear regression with 20 data points predicting the city intercepts from city-level predictors.

$$\Pr(Y_i = 1) = \text{logit}^{-1}(\alpha_{j(i)} + X_i\beta), \quad i = 1, \ldots, n,$$

where X is the matrix of individual-level predictors and $j(i)$ indexes the city where person i resides.

Clustered data (cont.)

The second part of the model—what makes it "multilevel"—is the regression of the city coefficients:

$$\alpha_j \sim \mathcal{N}(U_j\gamma, \sigma_\alpha^2), \quad j = 1, \ldots, 20,$$

where U is the matrix of city-level predictors, γ is the vector of coefficients for the city-level regression, and σ_α is the standard deviation of the unexplained group-level errors.

- The key is the group-level variation parameter σ_α, which is estimated from the data (along with α, β).
- The model for the α allows us to include all 20 of them in the model *without having to worry about collinearity*.

Indice

Repeated measurements

- Another kind of multilevel data structure involves repeated measurements on persons (or other units)—thus, measurements are clustered within persons, and predictors can be available at the measurement or person level.
- Suppose a dataset where some people who bought an insurance policy are every year asked either to renew or to interrupt the policy. We basically have as many repeated measurements for each person as many years that person is observed/asked.
- A naive multilevel logistic regression could then be similar to the previous model, with each α_j defined here in terms of the j-th ensured for which the i-th policy was observed.
- Here also, we can work with a more rectangular-structured data matrix (similarly as Table 1) or with two-data matrices: the choice is done in terms of users' convenience.

Indicator variables and fixed or random effects

- When including an input variable with J categories into a classical regression, standard practice is to choose one of the categories as a baseline and include indicators for the other $J-1$ categories (in the child enforcement example, one could set city 1 (Oakland) as the baseline and include indicators for the other 19. The coefficient for each city then represents its comparison to Oakland.)
- In a multilevel model it is unnecessary to do this arbitrary step of picking one of the levels as a baseline. For example, in the child support study, one would include indicators for all 20 cities in the model. In a classical regression these could not all be included because they would be collinear with the constant term, but in a multilevel model this is not a problem because they are themselves modeled by a group-level distribution.

Indicator variables and fixed or random effects (cont.)

- The varying coefficients (α_j's or β_j's) in a multilevel model are sometimes called **random effects**, a term that refers to the randomness in the probability model for the group-level coefficients.
- The term **fixed effects** is used in contrast to random effects—but not in a consistent way! Fixed effects are usually defined as varying coefficients that are not themselves modeled.
- As an interpretation issue, fixed effects are constant across individuals, and random effects vary.
- Varying slopes can be interpreted as *interactions* between an individual-level predictor and group indicators. As with classical regression models with interactions, the intercepts can often be more clearly interpreted if the continuous predictors are appropriately centered

Indice

Non-nested models

- So far we have considered the simplest hierarchical structure of individuals i in groups j. We briefly discuss now more complicated grouping structures.
- Example: a psychological experiment with two potentially interacting factors. We collect success rates data on pilots of flight simulators, with $n = 40$ data points corresponding to $J = 5$ treatment conditions and $K = 8$ different airports, as shown in the next figure (from G&H book, sect. 13.5):

Non-nested models (cont.)

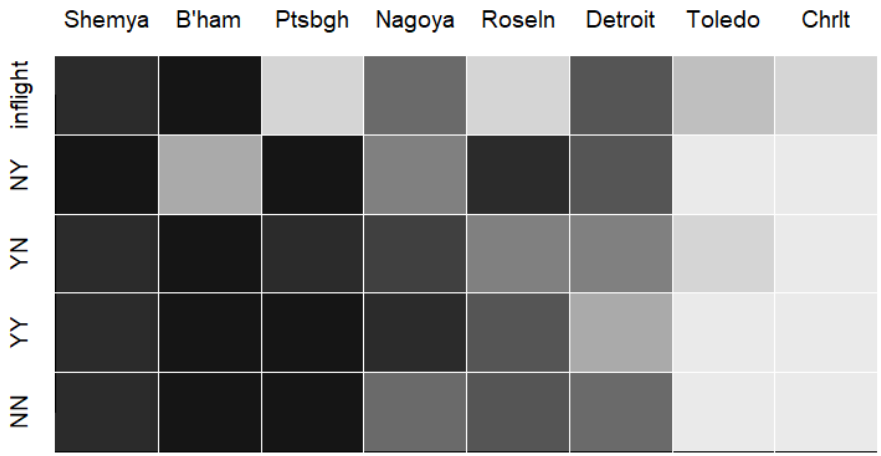


Figure 13.8 *Success rates of pilots training on a flight simulator with five different treatments and eight different airports. Shadings in the 40 cells i represent different success rates y_i, with black and white corresponding to 0 and 100%, respectively. For convenience in reading the display, the treatments and airports have each been sorted in increasing order of average success. These 40 data points have two groupings—treatments and airports—which are not nested.*

The data stored as a matrix and as an array are displayed in the next figure (always from G&H book):

Non-nested models (cont.)

Data in matrix form

airport	treatment conditions				
1	0.38	0.25	0.50	0.14	0.43
2	0.00	0.00	0.67	0.00	0.00
3	0.38	0.50	0.33	0.71	0.29
4	0.00	0.12	0.00	0.00	0.86
5	0.33	0.50	0.14	0.29	0.86
6	1.00	1.00	1.00	1.00	0.86
7	0.12	0.12	0.00	0.14	0.14
8	1.00	0.86	1.00	1.00	0.75

Data in vector form

y	j	k
0.38	1	1
0.00	1	2
0.38	1	3
0.00	1	4
0.33	1	5
1.00	1	6
0.12	1	7
1.00	1	8
0.25	2	1
...	...	...

Figure 13.9 *Data from Figure 13.8 displayed as an array* (y_{jk}) *and in our preferred notation as a vector* (y_i) *with group indicators* $j[i]$ *and* $k[i]$.

Non-nested models (cont.)

- The responses can be fit to a non-nested multilevel model of the form:

$$
\begin{aligned}
y_i &\sim \mathcal{N}(\mu + \gamma_{j(i)} + \delta_{k(i)}, \sigma_y^2), \ i = 1, \ldots, n \\
\gamma_j &\sim \mathcal{N}(0, \sigma_\gamma^2), \ j = 1, \ldots, J \\
\delta_k &\sim \mathcal{N}(0, \sigma_\delta^2), \ k = 1, \ldots, K,
\end{aligned} \tag{1}
$$

  where the parameters γ_j and δ_k represent treatment effects and airport effects. Their distributions are centered at zero because the regression model for y already has an intercept μ, and any nonzero mean for the γ and δ distributions could be folded into μ.

- When fit to the data in the figure, the estimated residual standard deviations at the individual, treatment and airport levels are $\hat{\sigma}_y = 0.23$, $\hat{\sigma}_\gamma = 0.04$ and $\hat{\sigma}_\delta = 0.32$. Thus, the variation among airports is huge—even larger than that among individual measurements—but the treatments vary almost not at all.

Non-nested models (cont.)

- Connection with **Analysis of Variance (ANOVA)**: as we know from classical statistics, ANOVA is typically used to learn the relative importance of different sources of variation in a dataset. In this example, how much of the variation in the data is explained by treatments, how much by airports, and how much remains after these factors have been included in a linear model? *If a multilevel model has already been fit, it can be summarized by the variation in each of its batches of coefficients*.
- In classical statistics, ANOVA refers either to a family of additive data decomposition, or to a method of testing the statistical significance of added predictors in a linear model. For the flight data simulator we can write:

$$y_i = \mu + \gamma_{j(i)} + \delta_{k(i)} + \epsilon_i, \tag{2}$$

and a classical two-way ANOVA can be obtained as follows:

Non-nested models (cont.)

```
> summary (aov (y ~ factor (treatment) + factor(airport)))

                  Df Sum Sq Mean Sq F value    Pr(>F)
factor(treatment)  4 0.0783  0.0196  0.3867    0.8163
factor(airport)    7 3.9437  0.5634 11.1299 1.187e-06 ***
Residuals         28 1.4173  0.0506
```

which indicates that the variation among treatments is not statistically significant. Let's see the sources of variation and degrees of freedom:

- 5 treatment effects minus 1 constraint = 4 degrees of freedom
- 8 airports effects minus 1 constraint = 7 df
- 40 residuals minus 12 constraints (1 mean, 4 treatment effects, 7 airport effects) = 28 df

- When comparing nested models, ANOVA is related to the classical test of the hypothesis that the smaller model is true, which is equivalent to the hypothesis that the additional predictors all have coefficients of zero when included in the larger model.

Non-nested models (cont.)

- When moving to multilevel modeling, the key idea we want to take from ANOVA is the estimation of the importance of different batches of predictors.
- A general solution to perform ANOVA here is to fit the model (2)—along with the random effects for γ, δ, and the error ϵ— and summarize the estimated variance components, $\hat{\sigma}_y, \hat{\sigma}_\gamma, \hat{\sigma}_\delta$.

Item-response and ideal-point models

- Usually applied to data with multilevel structure, typically non-nested, for example with measurements associated with persons and test items, or judges and cases.
- A standard model for success or failure in testing situations is the logistic item-response model, also called the Rasch model. Suppose J persons are given a test with K items, with $y_{jk} = 1$ if the response is correct. Then the logistic model can be written as:

$$\Pr(y_{jk} = 1) = \text{logit}^{-1}(\alpha_j - \beta_k), \tag{3}$$

with parameters:

  - α_j: the *ability* of person j,
  - β_k: the *difficulty* of item k.

Item-response and ideal-point models (cont.)

In general, not every person is given every item, so it is convenient to index the individual responses as $i = 1, \ldots, n$, with each response i associated with a person $j(i)$ and item $k(i)$. Thus model (3) becomes:

$$\Pr(y_i = 1) = \text{logit}^{-1}(\alpha_{j(i)} - \beta_{k(i)}). \tag{4}$$

- The model (4) is not identified, because a constant can be added to all the abilities α_j and all the difficulties β_k, and the predictions of the model will not change. From the standpoint of classical logistic regression, this nonidentifiability is a simple case of collinearity and can be resolved by constraining the estimated parameters in some way, for instance setting $\alpha_1 = \beta_1 = 0$, constraining the α_j's to sum to zero, or constraining the β_k's to sum to zero.

Item-response and ideal-point models (cont.)

- In a multilevel model, such constraints are unnecessary. The natural multilevel model for (4) assigns some normal distributions to the ability and the difficulty parameters:

$$\begin{aligned} \alpha_j &\sim \mathcal{N}(\mu_\alpha, \sigma_\alpha^2),\ j = 1, \ldots, J, \\ \beta_k &\sim \mathcal{N}(\mu_\beta, \sigma_\beta^2),\ k = 1, \ldots, K. \end{aligned}$$

  Now it is μ_α and μ_β that are not identified, because a constant can be added to each without changing the predictions. The simplest way to identify the multilevel model is set $\mu_\alpha = 0$, or to set $\mu_\beta = 0$ (but not both).

- *Ideal-point* modeling is an application of item-response models to a settimg where what is being easured is not ability of individuals and difficulty of items, but rather positions of individuals and items on some scale of values.

- Example: Supreme Court voting (G&H book, sect. 14.3).

Non-nested NB model of structure in social networks

- Understanding the structure of social networks, and the social processes that form them, is a central concern of sociology for both theoretical and practical reasons. Networks have been found to have important implications for social mobility, getting a job, the dynamics of fads and fashion, attitude formation, and the spread of infectious disease.
- Example (from book G&H, sect. 15.3): overdispersed Poisson regression model to learn about social structure. We fit the model to a random-sample survey of Americans who were asked, "How many X's do you know?" for a variety of characteristics X, defined by name (Michael, Christina, Nicole,...), occupation (postal worker, pilot, gun dealer,...), ethnicity (Native American), or experience (prisoner, auto accident victim,...).
- The original goals of the survey were (1) to estimate the distribution of individuals' network size, defined to be the number of acquaintances, in U.S. population and (2) to estimate the sizes of certain subpopulations, especially those that are hard to count using regular survey results.

Non-nested NB model of structure in social networks (cont.)

- Modeling setup: for respondent $i = 1, \ldots, 1370$ and subpopulations $k = 1, \ldots, 32$, we use the notation y_{ik} for the number of persons in group k known by person i.
- We evaluate three possible models:

$$\begin{aligned} \text{Erdos-Renyi model} &: \lambda_{ik} = ab_k \\ \text{null model} &: \lambda_{ik} = a_i b_k \\ \text{overdispersed model} &: \lambda_{ik} = a_i b_k g_{ik}. \end{aligned}$$

- *Null model*: in which individuals i have varying levels of gregariousness or popularity, so that the expected number of persons in group k known by person i will be proportional to this gregariousness parameter, which we label a_i. Departure from this model—patterns not simply explained by differing group sizes or individual popularities—can be viewed as evidence of structured social acquaintance networks.

Non-nested NB model of structure in social networks (cont.)

- *Overdispersion* in these data can arise if the relative propensity for knowing someone in prison, for example, varies from respondent to respondent. We can write this in the generalized linear model framework as:

$$y_{ik} \sim \text{Poisson}(e^{a_i + b_k + \gamma_{ik}}),$$

where each $\gamma_{ik} = \log(g_{ik}) \equiv 0$ in the null model. For each subpopulation k, we let the multiplicative factors $g_{ik} = e^{\gamma_{ik}}$ follow a Gamma distribution with a value of 1 for the mean and a value of $1/(\omega_k - 1)$ for the shape parameter. In this way:

$$y_{ik} \sim \text{NegBin}(e^{a_i + b_k}, \omega_k),$$

Costs and benefits of multilevel modeling

Before we go to the effort of learning multilevel modeling, it is helpful to briefly review what can be done with classical regression:

- Prediction for continuous or discrete outcomes,
- Fitting of nonlinear relations using transformations,
- Inclusion of categorical predictors using indicator variables,
- Modeling of interactions between inputs,
- Causal inference (under appropriate conditions).

Costs and benefits of multilevel modeling (cont.)

Motivations for moving to multilevel models:

- Accounting for individual- and group-level variation in estimating group-level regression coefficients.
- Modeling variation among individual-level regression coefficients. In classical regression, one can do this using indicator variables, but multilevel modeling is convenient when we want to model the variation of these coefficients across groups, make predictions for new groups, or account for group-level variation in the uncertainty for individual-level coefficients.
- Estimating regression coefficients for particular groups
- A potential drawback to multilevel modeling is the additional complexity of coefficients varying by group.

Costs and benefits of multilevel modeling (cont.)

- A multilevel model requires additional assumptions beyond those of classical regression—basically, each level of the model corresponds to its own regression with its own set of assumptions such as additivity, linearity, independence, equal variance, and normality.
- The usual alternative to multilevel modeling is classical regression—either ignoring group-level variation, or with varying coefficients that are estimated classically (and not themselves modeled)—or combinations of classical regressions.
- In various limiting cases, the classical and multilevel approaches coincide. When there is very little group-level variation, the multilevel model reduces to classical regression with no group indicators; conversely, when group-level coefficients vary greatly (compared to their standard errors of estimation), multilevel modeling reduces to classical regression with group indicators.

Costs and benefits of multilevel modeling (cont.)

- When the number of groups is small (less than five, say), there is typically not enough information to accurately estimate group-level variation. As a result, multilevel models in this setting typically gain little beyond classical varying-coefficient models.
- Computational softwares: `lme4`, `WinBUGS`, `JAGS`, Stan.

Indice

Motivations

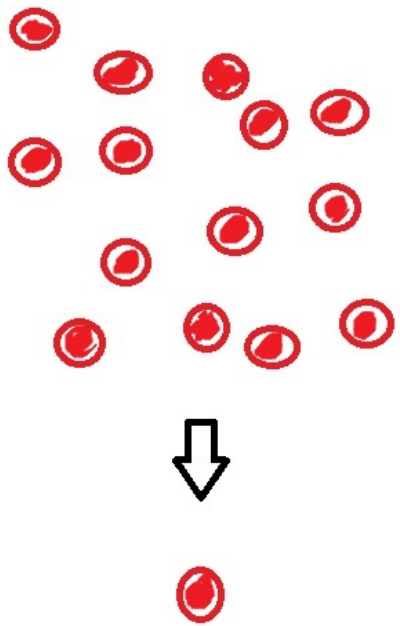
A common problem in applied statistics is modelling individuals/objects of a *population*.

Within this population, there may be some *subpopulations* sharing some common features. Thus, we should statistically acknowledge for this distinct groups' membership.

Multilevel/hierarchical models are extensions of regression models in which data are structured in groups and coefficients can vary by group. We start with simple grouped structures—such as people within cities, students within schools, etc—where some information is available on individuals and some information is at the group level.

Motivations

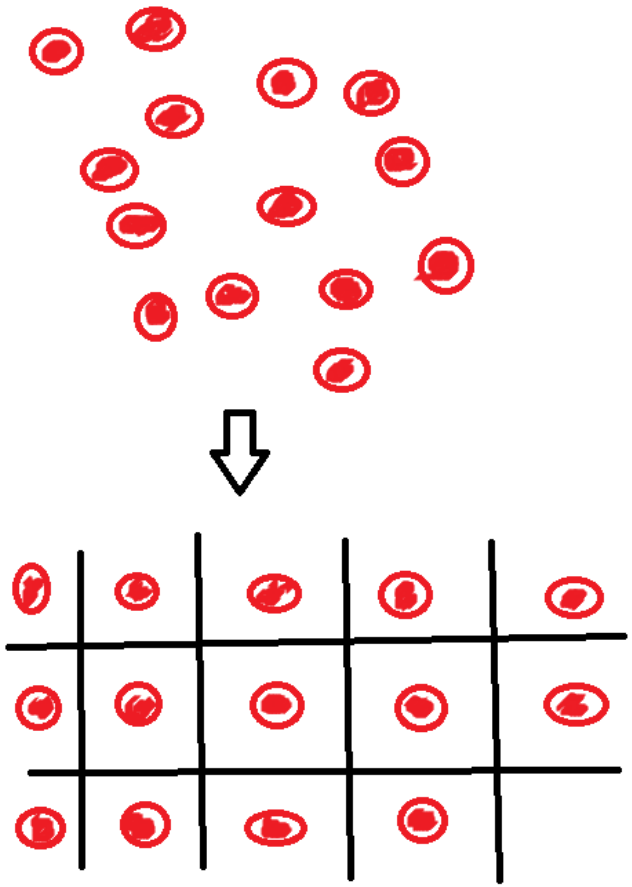
If we assume that every individual is equivalent then we can pool the data, but only at the expense of bias $\Leftrightarrow$ Complete pooling.



$$y_i \sim \mathcal{N}(\alpha + \beta x_i, \sigma^2)$$

Motivations

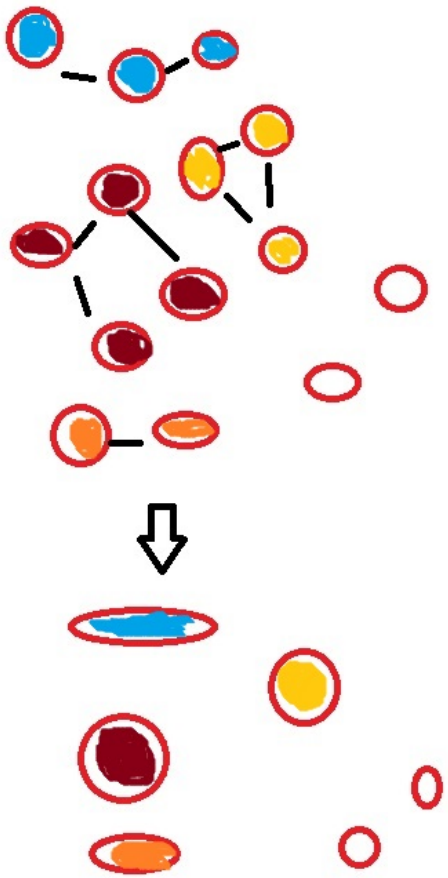
Conversely, modelling every individual separately avoids any bias, but then the data becomes very sparse and inferences weak $\Leftrightarrow$ No pooling.



$$y_i \sim \mathcal{N}(\alpha_i + \beta x_i, \sigma^2)$$

Motivations

A compromise between complete pooling and no pooling that could balance bias and variance would be ideal. Thus, hierarchical models allow for this:



$$y_{ij} \sim \mathcal{N}(\alpha_{j(i)} + \beta x_i, \sigma^2)$$

Motivations

The common feature of such models is that the observed units y_{ij} are indexed by the statistical unit i in group j (examples: *students within schools*, *players within teams*). In general, these observable outcomes are modelled conditionally on certain *not observable* parameters θ_j, viewed as drawn from a population distribution, which themselves are given a probabilistic (prior) distribution in terms of further parameters, known as *hyperparameters*.

Simple nonhierarchical models are usually inappropriate for hierarchical data: with few parameters, they generally cannot fit large datasets accurately.

Conversely, hierarchical models can have enough parameters to fit the data well, while using a population distribution.

The fundamental concept of exchangeability - 1

In order to formalize this approach we need to consider exchangeability.

Consider a set of experiments $j = 1, \ldots, J$, in which experiment j has data (vector) y_j and parameter vector θ_j, with likelihood $p(y_j|\theta_j)$. In the linear model, we have $\theta = (\alpha, \beta, \sigma^2)$

If no information-other than the data y-is available to distinguish any of the θ_j's from any of the others, and no ordering or grouping of the parameters can be made, one must assume symmetry among the parameters in their prior distribution.

The fundamental concept of exchangeability - 2

- This symmetry is represented probabilistically by exchangeability: the parameters $(\theta_1, \ldots, \theta_J)$ are exchangeable in their joint prior distribution if $\pi(\theta_1, \ldots, \theta_J)$ is invariant to permutations of the indexes $(1, \ldots, J)$.
- In practice, ignorance implies exchangeabilitiy. Consider the analogy to a roll of a dice: we should initially assign equal probabilities to all six outcomes, but if we study the measurements of the dice and weigh the dice carefully, we might eventually notice imperfections, which might make us favour one outcome over the others and thus eliminate the symmetry among the six outcomes.

The fundamental concept of exchangeability - 3

The simplest form of an *exchangeable distribution* has each of the parameters θ_j as an independent sample from a prior (or population) distribution governed by some unknown parameter vector ϕ; thus,

$$\pi(\theta|\phi) = \prod_{j=1}^{J} \pi(\theta_j|\phi). \tag{5}$$

In general, ϕ is unknown, so our distribution for θ must average over our uncertainty in ϕ:

$$\pi(\theta) = \int \left(\prod_{j=1}^{J} \pi(\theta_j|\phi) \right) \pi(\phi) d\phi. \tag{6}$$

The fundamental concept of exchangeability - 4

In such a way, the joint distribution for y and θ becomes:

$$p(\theta, y) = \prod_{i=1}^{n} p(y_{ij}|\theta_{j(i)})\pi(\theta_{j(i)}|\phi)\pi(\phi), \tag{7}$$

with the nested index $j(i)$ denoting the group membership of the i-th unit, whereas the joint posterior distribution for θ, ϕ is:

$$\pi(\theta, \phi|y) \propto \pi(\phi, \theta)p(y|\theta). \tag{8}$$

Careful! ϕ is usually not known. Thus, the joint prior distribution $\pi(\phi, \theta)$ may be factorized as

$$\pi(\phi, \theta) = \pi(\phi)\pi(\theta|\phi),$$

where $\pi(\phi)$ is the *hyperprior* distribution.

Hierarchical models: formalization

Often observations (and/or parameters) are not fully exchangeable, but are *partially* or *conditionally* exchangeable.

- If observations can be grouped, we may make hierarchical modelling, where each group has its own subgroup, but the group properties are unknown.
- If y_i has additional information x_i so that y_i are not exchangeable but (y_i, x_i) still are exchangeable, then we can make a joint model for (y_i, x_i) or a conditional model for $y_i|x_i$.

In general, the usual way to model exchangeability with covariates is through conditional independence:

$$\pi(\theta_1, \ldots, \theta_J | x_1, \ldots, x_J) = \int \left[\prod_{j=1}^{J} \pi(\theta_j | \phi, x_j) \right] \pi(\phi | x) d\phi$$

Hierarchical models: objections to exchangeability

- In virtually any statistical application, it is natural to object to exchangeability on the grounds that the units actually differ.
- That the units differ, implies that the θ_j's differ, but it might be perfectly acceptable to consider them as if drawn from a common distribution.
- As usual in regression, the valid concern is not about exchangeability, but about encoding relevant knowledge as explanatory variables where possible.

Hierarchical models: formalization

We may try to formalize a hierarchical model by acknowledging at least two levels:

- individual level: observed $y_{ij},\ i = 1, \ldots, n,\ j = 1, \ldots J$;

$$y_{ij} \sim p(y|\theta_j) \quad \text{likelihood}$$

- group level: unobserved θ_j, $j = 1, \ldots, J$, depending on an hyperparameter ϕ.

$$\theta_j \sim \pi(\theta|\phi) \quad \text{prior}$$

- heterogeneity level: unobserved ϕ

$$\phi \sim \pi(\phi) \quad \text{hyperprior}$$

Indice

Extending linear models

Hierarchical regression models are useful as soon as there are predictors at different levels of variation. Some examples may be:

- In studying scholastic achievement, we may have students within schools, with predictors both at the individual and at the group level.
- Data obtained by stratified or cluster sampling

We can think of a generalization of linear regression, where intercepts, and possibly slopes, are allowed to vary by group.

A batch of J coefficients is assigned a model, and this group-level model is estimated simultaneously with the data-level regression of y.

Extending linear models: radon data

Radon data

Suppose to measure radon emissions in more than 80000 houses throughout US. Our goal in analyzing these data is to estimate the distribution of radon levels in each of the approximately 3000 counties, so that homeowners could make decisions about measuring or remediating the radon in their houses.

The data are structured *hierarchically*: houses within counties. As a predictor, we have the floor on which th measurement is taken, either basement or first floor; radon comes from underground and can enter more easily when a house is built into the ground. We fit a model where y_i is the logarithm of the radon measurement in house i, and x is the floor variable (0 if basement, 1 if first floor).

Partial pooling with no predictors

Hierarchical (or multilevel) modelling is a compromise between two extremes: complete pooling, in which the group indicators are not included in the model, and no pooling, in which separate models are fit within each group. For such a reason, we may refer to hierarchical modellling as partial pooling.

We start our journey into hierarchical models with the simplest model ever for the radon data, a hierarchical linear model with no predictors:

$$\begin{aligned} y_{ij} &\sim \mathcal{N}(\alpha_{j(i)}, \sigma^2), \ i = 1, \ldots, n, \ \text{Individual level} \\ \alpha_j &\sim \mathcal{N}(\mu_\alpha, \tau^2), \ j = 1, \ldots, J, \qquad \text{Group level} \end{aligned} \tag{9}$$

where $\alpha_{j(i)} = 1, \ldots, J$ is the intercept for the i-th unit, belonging to the j-th group.

Partial pooling with no predictors

Consider the goal of estimating the distribution of radon levels of the houses within each of 85 counties in Minnesota. One estimate would be the average that completely pools data across all counties. This ignores variation among counties, however, so perhaps a better option would be simply to use the average log radon level in each county. Estimates $\pm$ standard errors are plotted against the number of observations in each county in the next plot, left panel.

A third option is hierarchical modelling: estimates $\pm$ standard errors are plotted against the number of observations for each county.

Partial pooling with no predictors

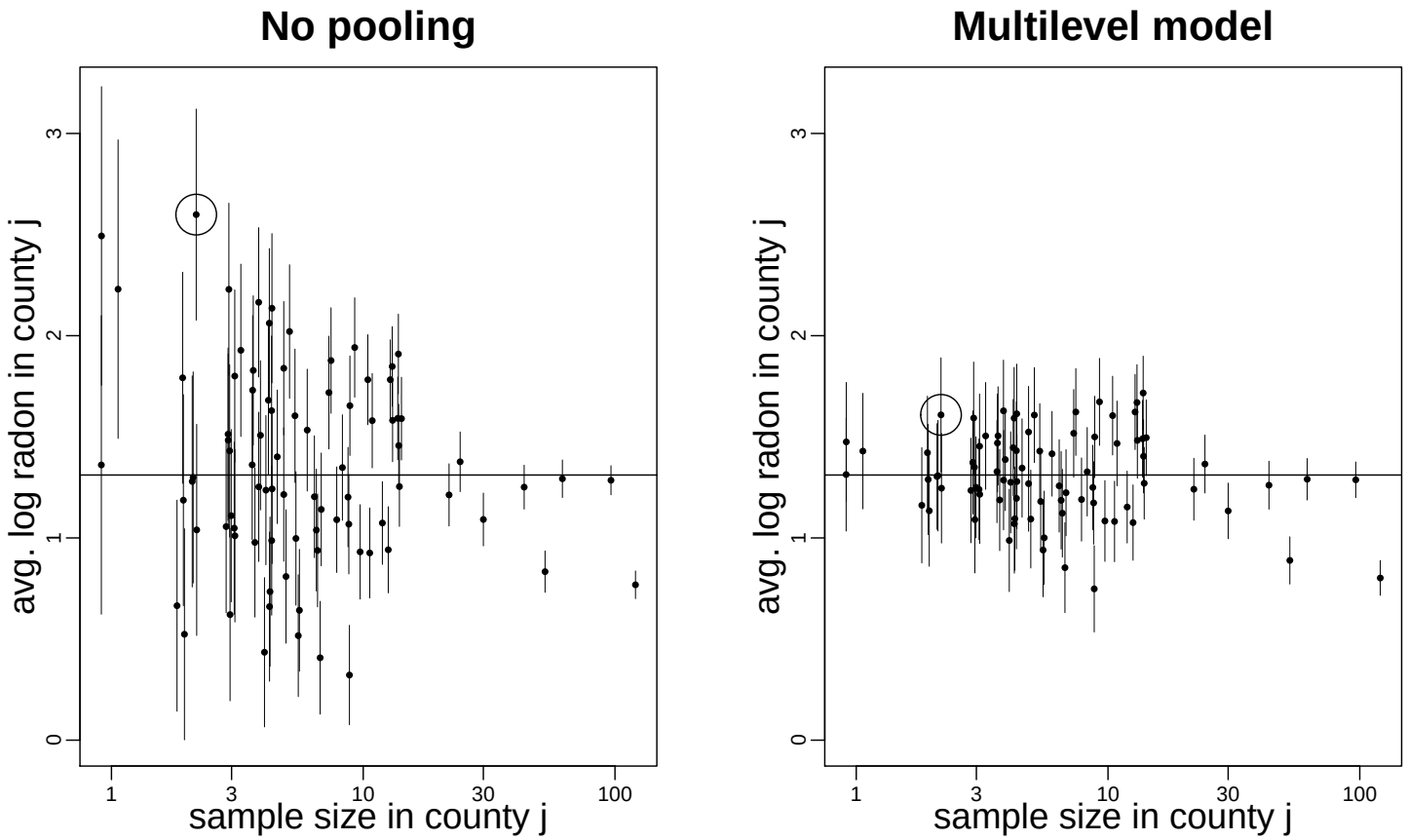


Figure: Estimates ± standard errors for the average log radon levels in Minnesota counties plotted versus the number of observations in the county.

Partial pooling with no predictors

- Whereas complete pooling ignores variation between counties, the no-pooling analysis overfits the data within each county.
- In no-pooling analysis, the counties with fewer measurements have more variable estimates and larger higher standard errors. It systematically causes us to think that certain counties are more extreme, just because they have smaller sample sizes!
- The hierarchical estimate for a given county j can be approximated as a weighted average:

$$\hat{\alpha}_j = \frac{\frac{n_j}{\sigma^2}\bar{y}_j + \frac{1}{\tau^2}\bar{y}_{\text{all}}}{\frac{n_j}{\sigma^2} + \frac{1}{\tau^2}} \qquad (10)$$

where n_j is the number of observations in the j-th county, $\bar{y}_j$ is the mean of the observations in the county (unpooled estimate), and $\bar{y}_{\text{all}}$ is the mean over all counties (completely pooled estimate).

Partial pooling with no predictors

The weighted average (10) reflects the relative amount of information available about the individual county, on one hand, and the average of all counties, on the other:

- Averages from counties with smaller sample sizes carry less information (n_j small), and the weighting pulls the multilevel estimates closer to the overall state average. If $n_j = 0$, $\hat{\alpha}_j = \bar{y}_{\text{all}}$, the overall average.
- Averages from counties with larger sample sizes carry more information. As $n_j \to \infty, \hat{\alpha}_j = \bar{y}_j$, the county average.
- When variation across counties is very small, the weighting pulls the multilevel estimates to the overall mean: as $\tau^2 \to 0$, $\hat{\alpha}_j = \bar{y}_{\text{all}}$.
- When variation across the counties is large, the weighting pulls the multilevel estimates to the county average: as $\tau^2 \to \infty, \hat{\alpha}_j = \bar{y}_j$.

Partial pooling with predictors

The same principle of finding a compromise between these two extremes applies for more general models. We consider now the individual-level predictor x, where $x_i = 1$ for the first floor and $x_i = 0$ for the basement.

Thus, the second model we consider is a *varying-intercept* model:

$$\begin{aligned} y_{ij} &\sim \mathcal{N}(\alpha_{j(i)} + \beta x_i, \sigma^2), \ i = 1, \ldots, n, \ \text{Individual level} \\ \alpha_j &\sim \mathcal{N}(\mu_\alpha, \tau^2), \ j = 1, \ldots, J, \qquad \text{Group level} \end{aligned} \tag{11}$$

To appreciate hierarchical modelling, we start plotting some estimates according to complete and no pooling.

Partial pooling with predictors

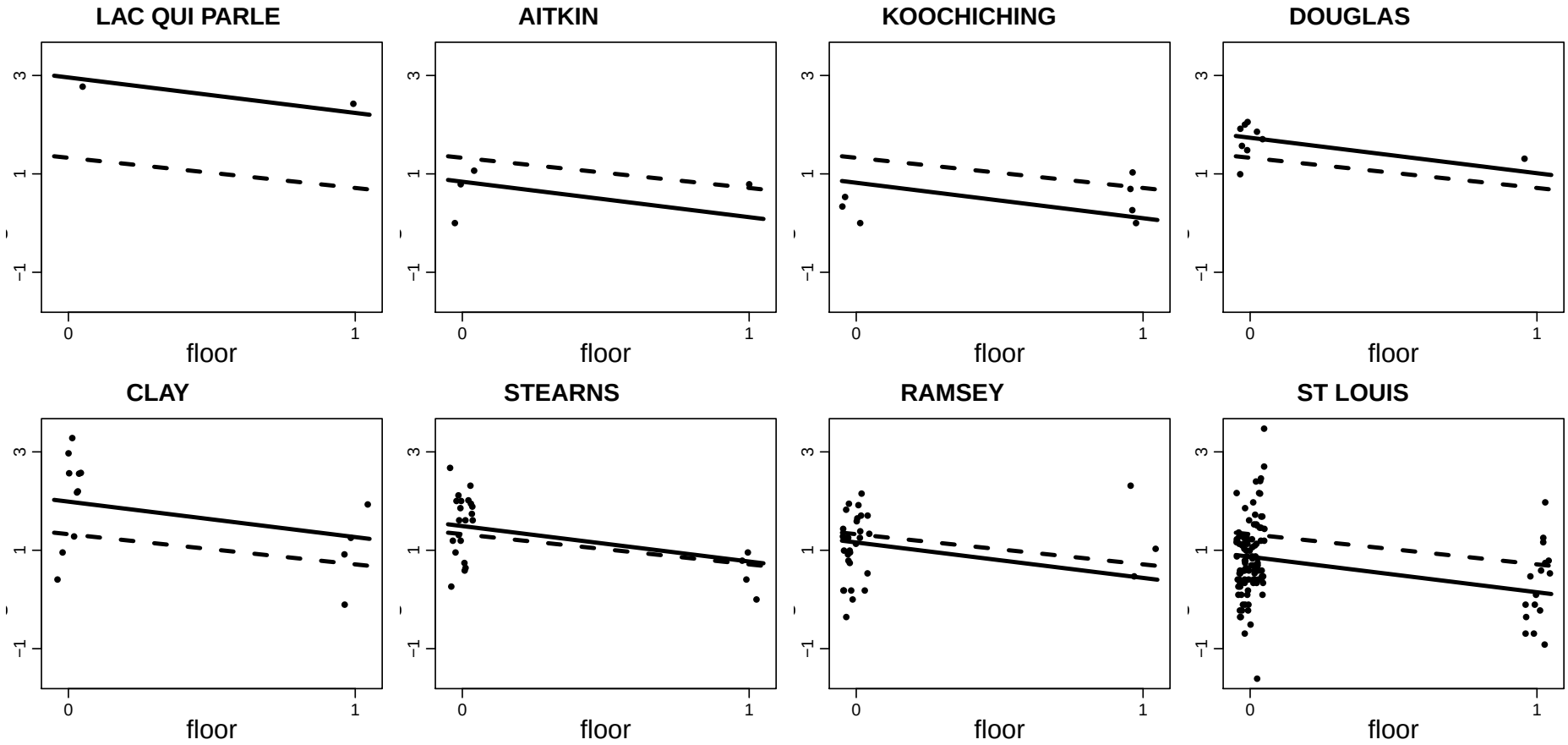


Figure: Complete pooling (dashed lines) and no pooling (solid lines) for 8 counties in Minnesota.

Partial pooling with predictors

Both these analysis have problems.

- The complete pooling analysis ignores any variation in average radon levels between counties.
- The no-pooling analysis has problems too, however, which we can see in Lac Qui Parle County, since the estimate is based on only two observations.

Let's fit now model (11) via the function `stan_lmer` of the `rstanarm` R package, and plot again the estimates.

Partial pooling with predictors

```
mlm.radon.pred <- stan_lmer(y ~ x+ (1|county))
print(mlm.radon.pred)
stan_lmer
 family:       gaussian [identity]
 formula:      y ~ x + (1 | county)
 observations: 919
------
            Median MAD_SD
(Intercept)  1.5    0.1
x           -0.7    0.1
```

Partial pooling with predictors

```
Error terms:
 Groups   Name        Std.Dev.
 county   (Intercept) 0.33
 Residual             0.76
Num. levels: county 85
```

We obtain the following posterior estimates for the two sources of variation: $\hat{\tau} = 0.33, \hat{\sigma} = 0.76$.

Partial pooling with predictors

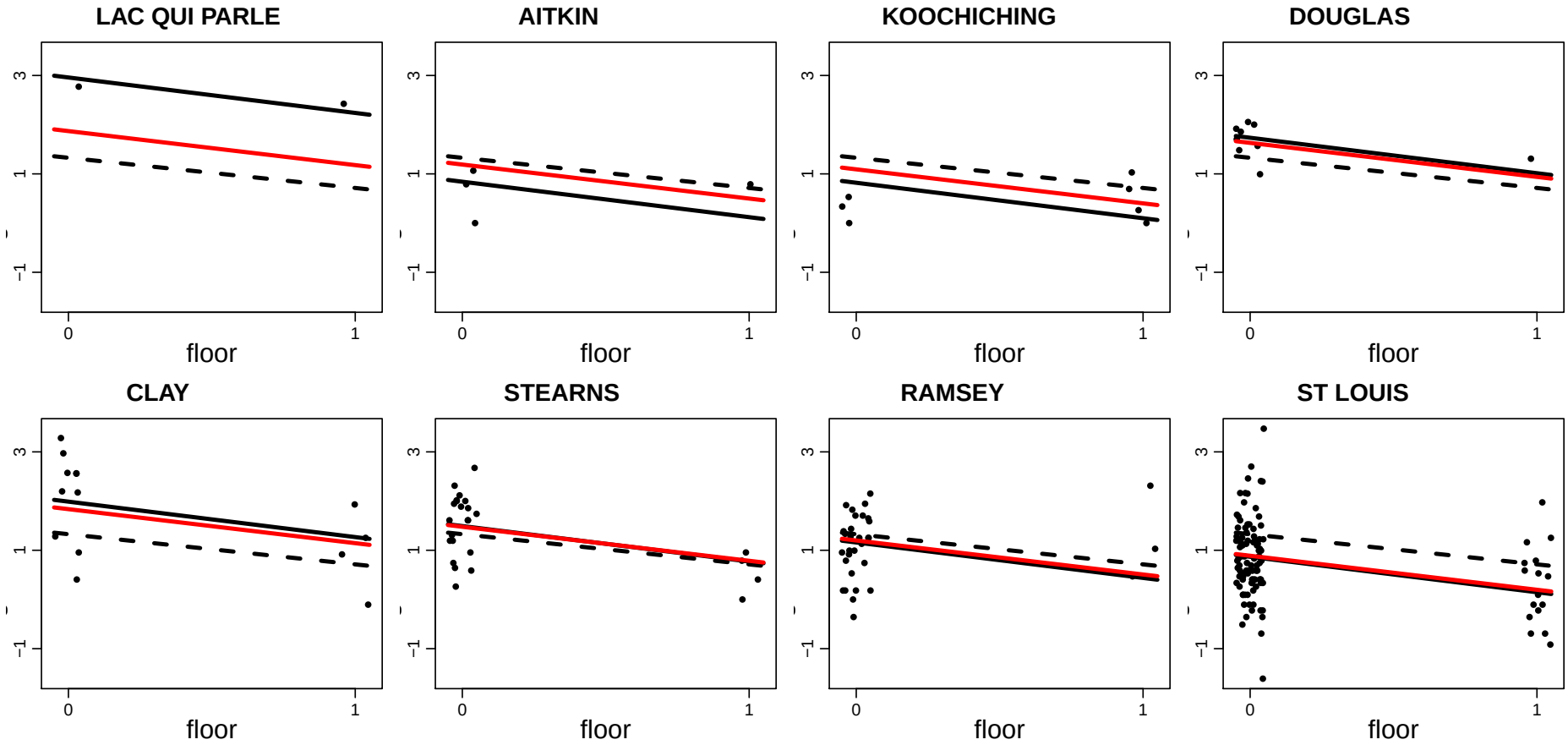


Figure: Complete pooling (dashed lines), no pooling (solid lines) and partial pooling (solid red lines).

Partial pooling with predictors

- The estimated line from the hierarchical model (11) in each county lies between the complete-pooling and no-pooling regression lines. There is strong pooling (solid red line closer to complete-pooling line) in counties with small sample sizes, and only weak pooling (solid red line close to no-pooling line) in counties containing many measurements.
- Classical regression models can be viewed as special cases of multilevel models. The limits $\tau \to 0$ (complete pooling) and $\tau \to \infty$ (no pooling) seem to be restrictive: given multilevel data, we can estimate τ, which acts as hyperparameter of a prior distribution on α.
- Note that the function `stan_lmer` works in the same way as the function `lmer` for classical inference. However, when the number of groups is small, it can be useful to switch to Bayesian inference, *to better account for uncertainty* in model fitting.

Partial pooling with predictors

We can generalize equation (10) as follows:

$$\hat{\alpha}_j \approx \frac{\frac{n_j}{\sigma^2}}{\frac{n_j}{\sigma^2} + \frac{1}{\tau_\alpha^2}}(\bar{y}_j - \beta\bar{x}_j) + \frac{\frac{1}{\tau_\alpha^2}}{\frac{n_j}{\sigma^2} + \frac{1}{\tau_\alpha^2}}\mu_\alpha, \tag{12}$$

a weighted average of the no-pooling estimate for its group $(\bar{y}_j - \beta\bar{x}_j)$ and the prior mean μ_α.

- Multilevel modeling partially pools the group-level parameters α_j toward their mean level, μ_α.
- There is more pooling when the group-level standard deviation τ is small.
- There is more smoothing for groups with fewer observations.

Partial pooling with predictors

We may disaggregate the information averaging over the counties, the *fixed* effects, and the county-level errors, the *random* effects, using the functions `fixef()` and `ranef()` of the `rstanarm` package:

```
fixef(mlm.radon.pred)
(Intercept)           x
  1.4623684  -0.6919822

ranef(mlm.radon.pred)
  $county
    (Intercept)
1  -0.264735142
2  -0.534511687
. . .
85 -0.073852110
```

The est. line for the first county is: $(1.46 - 0.26) - 0.69x = 1.20 - 0.69x$.

Eight schools example

We illustrate a normal model with a problem in which the hierarchical Bayesian analysis gives conclusions that differ in important respects from other methods.

Eight schools example (BDA, 5.5)

A study was performed for the Educational Testing Service to analyze the effects of special coaching programs on test scores in each of eight high-schools.

The outcome variable in each study was a score, varying between 200 and 800, with mean about 500 and standard deviation about 100. There is no prior reason to believe that any of the eight programs is more effective than any other.

As we'll see, the choice of the prior is of substantial importance here.

Eight schools

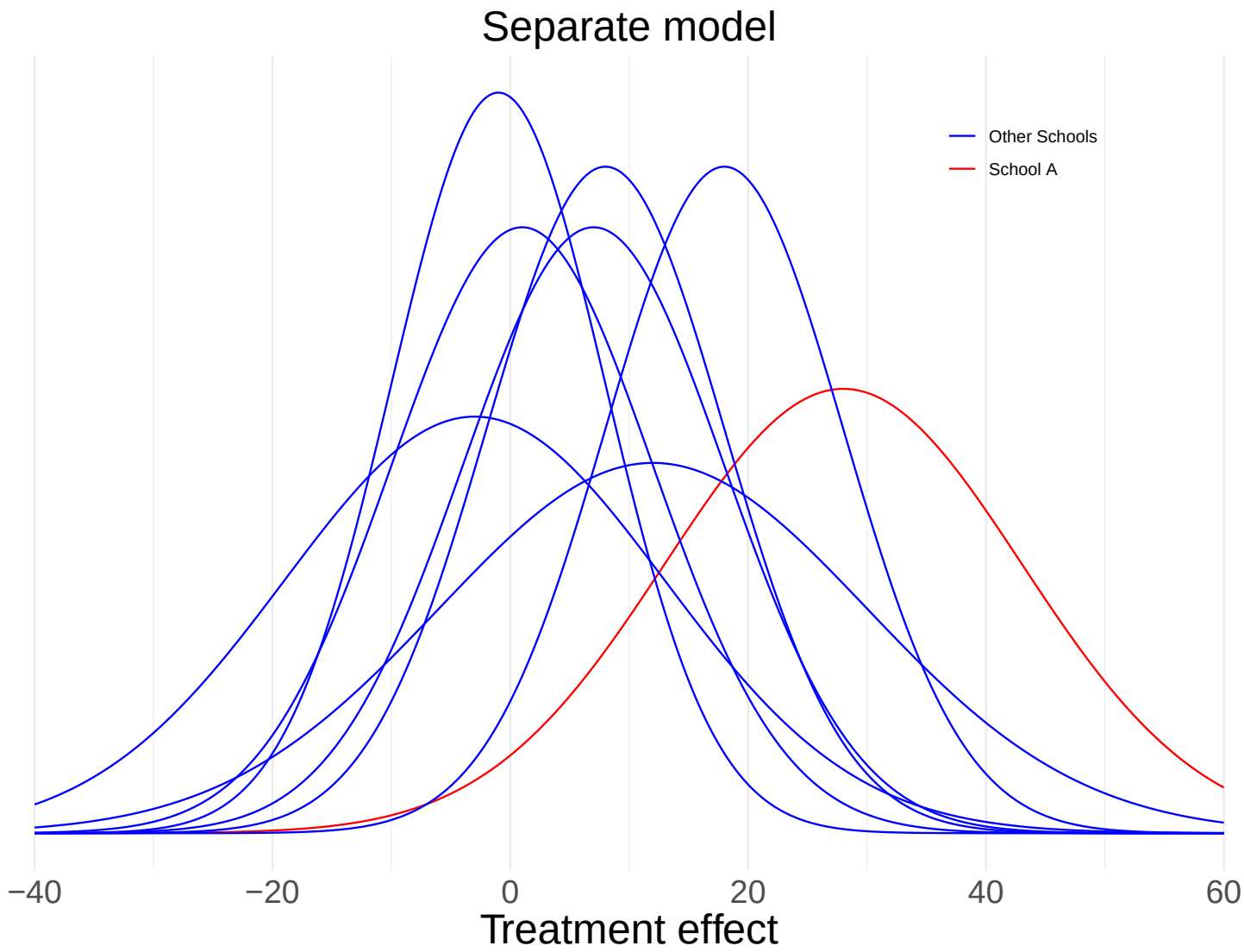
We denote with y_{ij} the result of the i-th test in the j-th school. We assume the following model:

$$\begin{aligned} y_{ij} &\sim \mathcal{N}(\theta_j, \sigma_y^2) \\ \theta_j &\sim \mathcal{N}(\mu, \tau^2) \end{aligned} \tag{13}$$

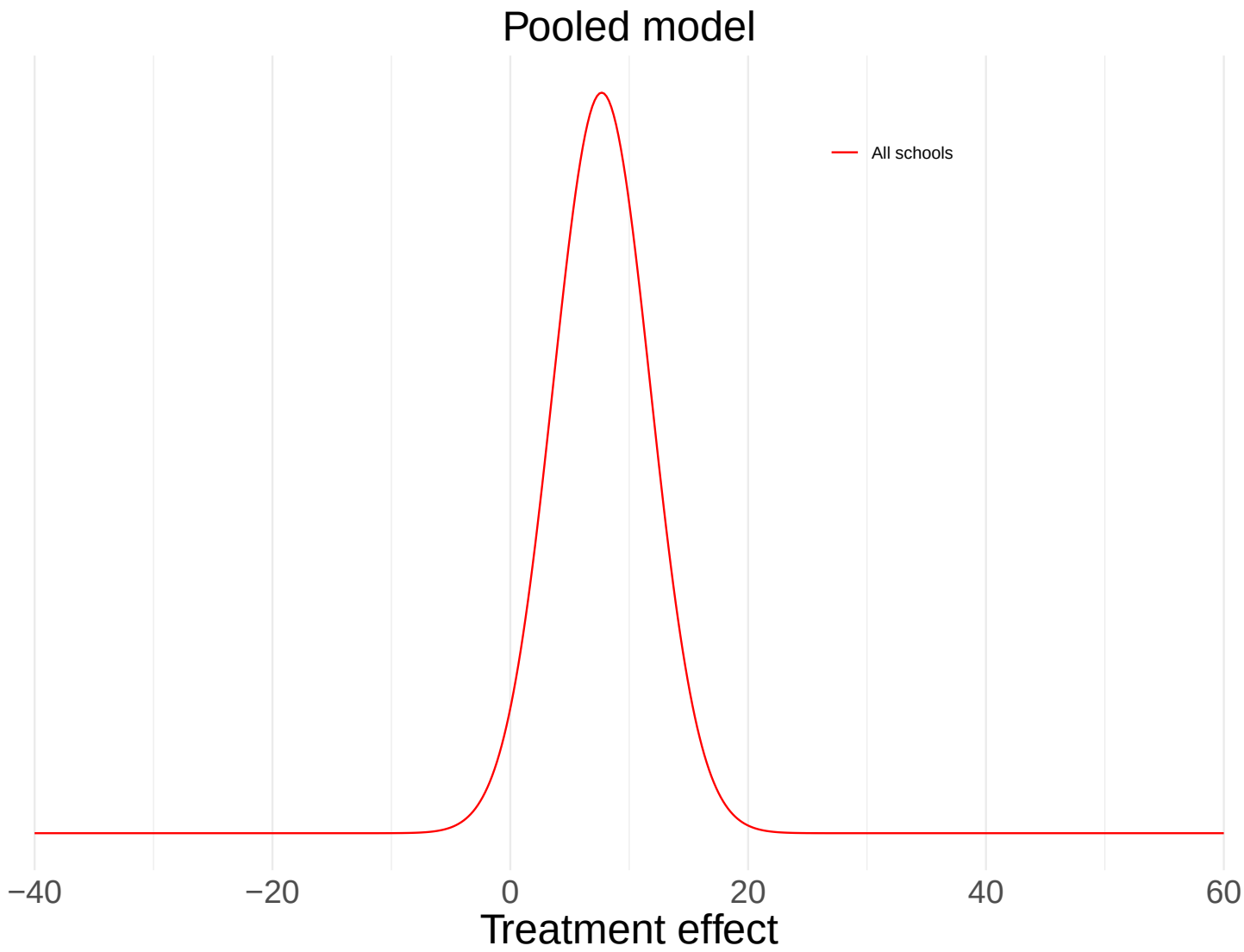
Do some schools perform better/worse according to these coaching effects?
We will make three distinct analysis: separate analysis, pooled analysis and hierarchical modelling.

Actually, for each school we have the estimated coaching effects y_j, $y = (28, 8, -3, 7, -1, 1, 18, 12)$, and a measure of standard deviation for them, $s = (15, 10, 16, 11, 9, 11, 10, 18)$.

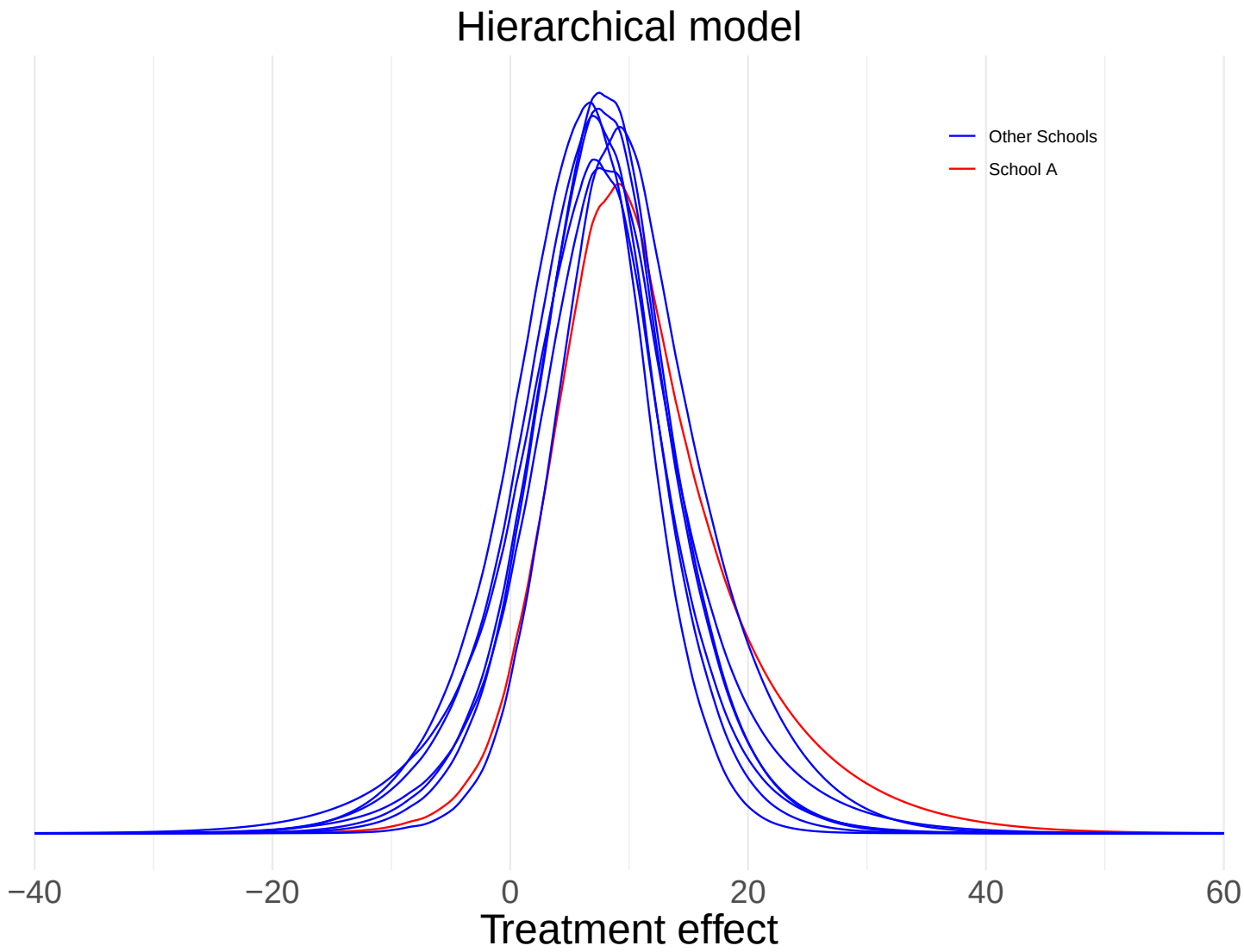
Eight schools: separate analysis



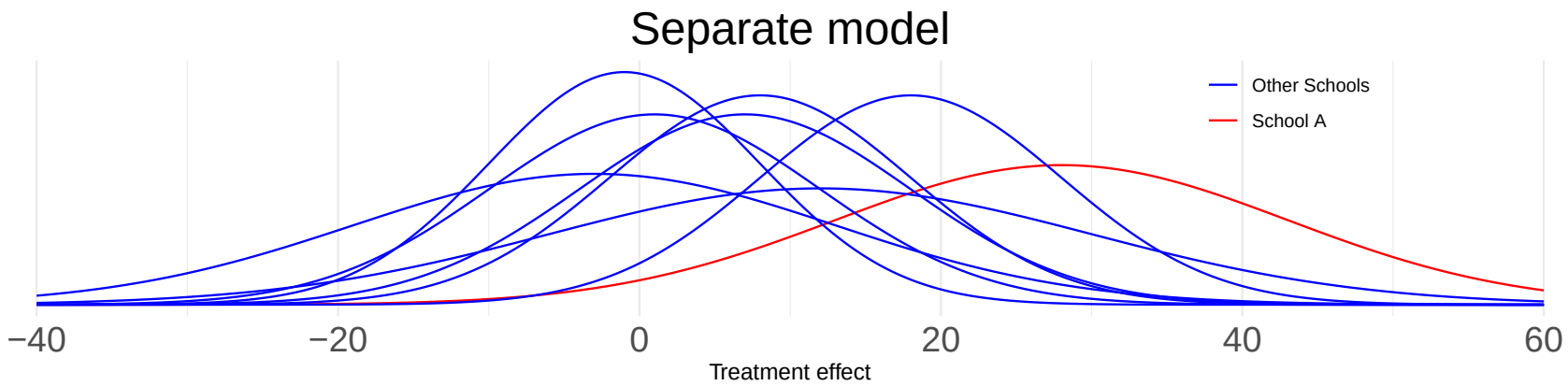
Eight schools: pooled analysis

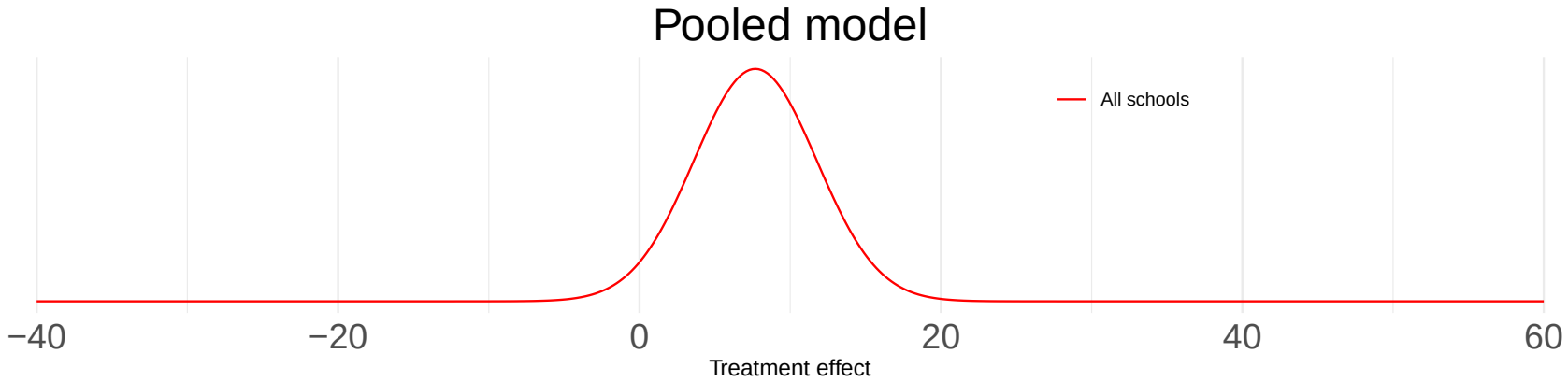


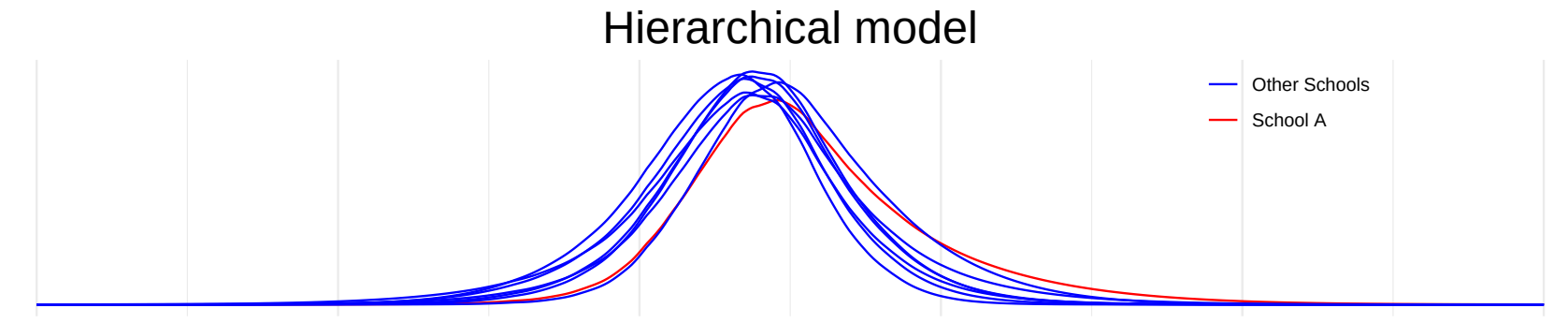
Eight schools: hierarchical model



Eight schools: three models







Eight schools: three models

Comments:

- Separate analysis: the standard errors of these estimated effects make very difficult to distinguish between any of the experiments...treating each experiment separately and applying the simple normal analysis in each yields 95% posterior intervals that all overlap substantially.
- Pooled-analysis: under the hypothesis that all experiments have the same effect and produce independent estimates of this common effect, we could treat y as eight normally distributed observations with known variances. The pooled estimate is 7.7, and the posterior variance is 16.6.

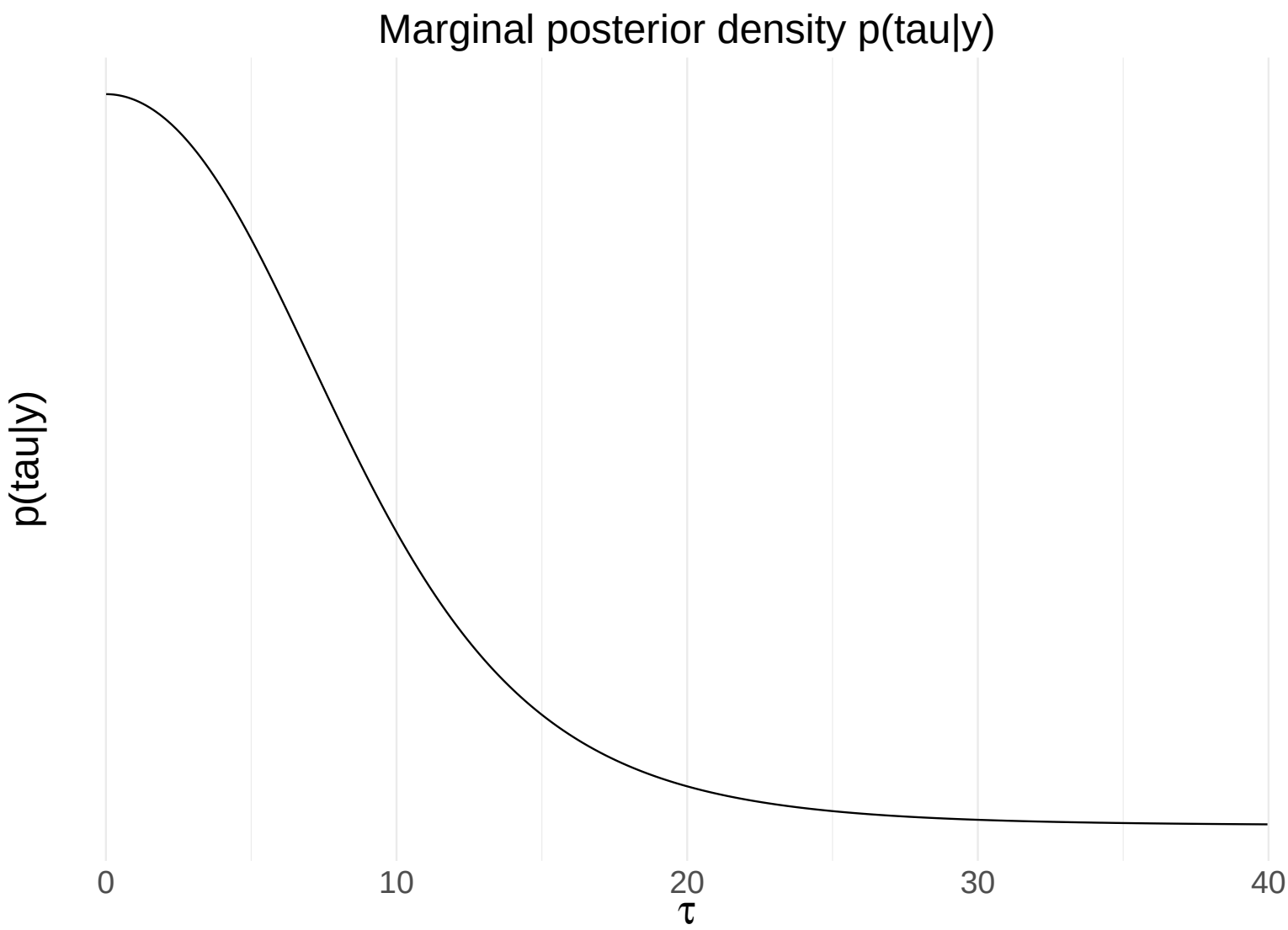
However, both the extreme analysis have difficulties.

Eight schools: three models

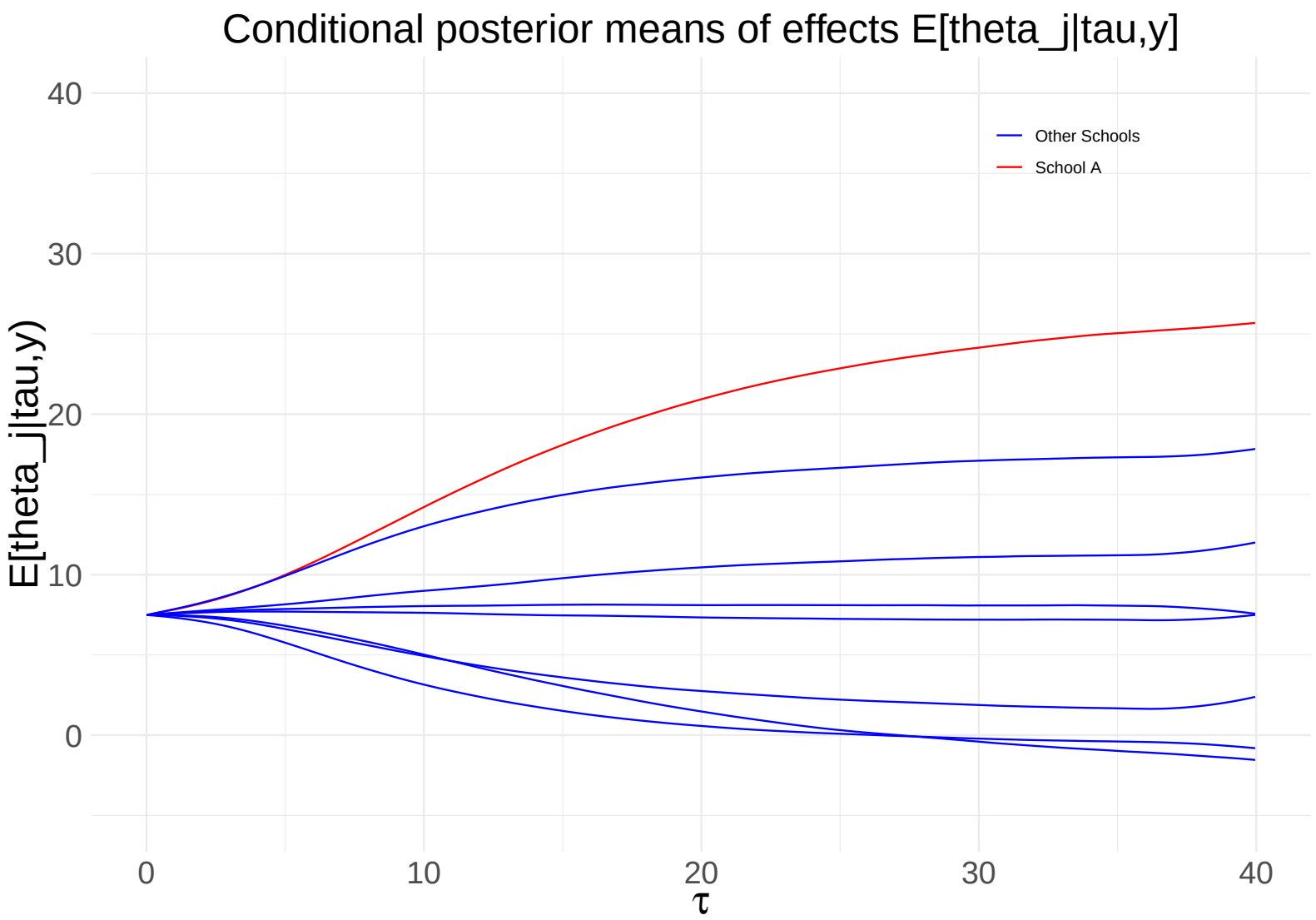
Other comments:

- Consider school A. The effect in school A is estimated as 28.4 with a standard error of 14.9 under the separate analysis, versus a pooled estimate of 7.7 with a standard error of 4.1. Mmh...should I flip a coin?
- We would like a compromise that combines information from all the eight experiments without assuming all the θ_j to be equal. The Bayesian analysis under the hierarchical model provides exactly that.
- As we may see from the third plot, the posterior distribution of $\theta_1, \ldots, \theta_8$ results to be closer to the complete analysis. Let's see now some other posterior analysis.

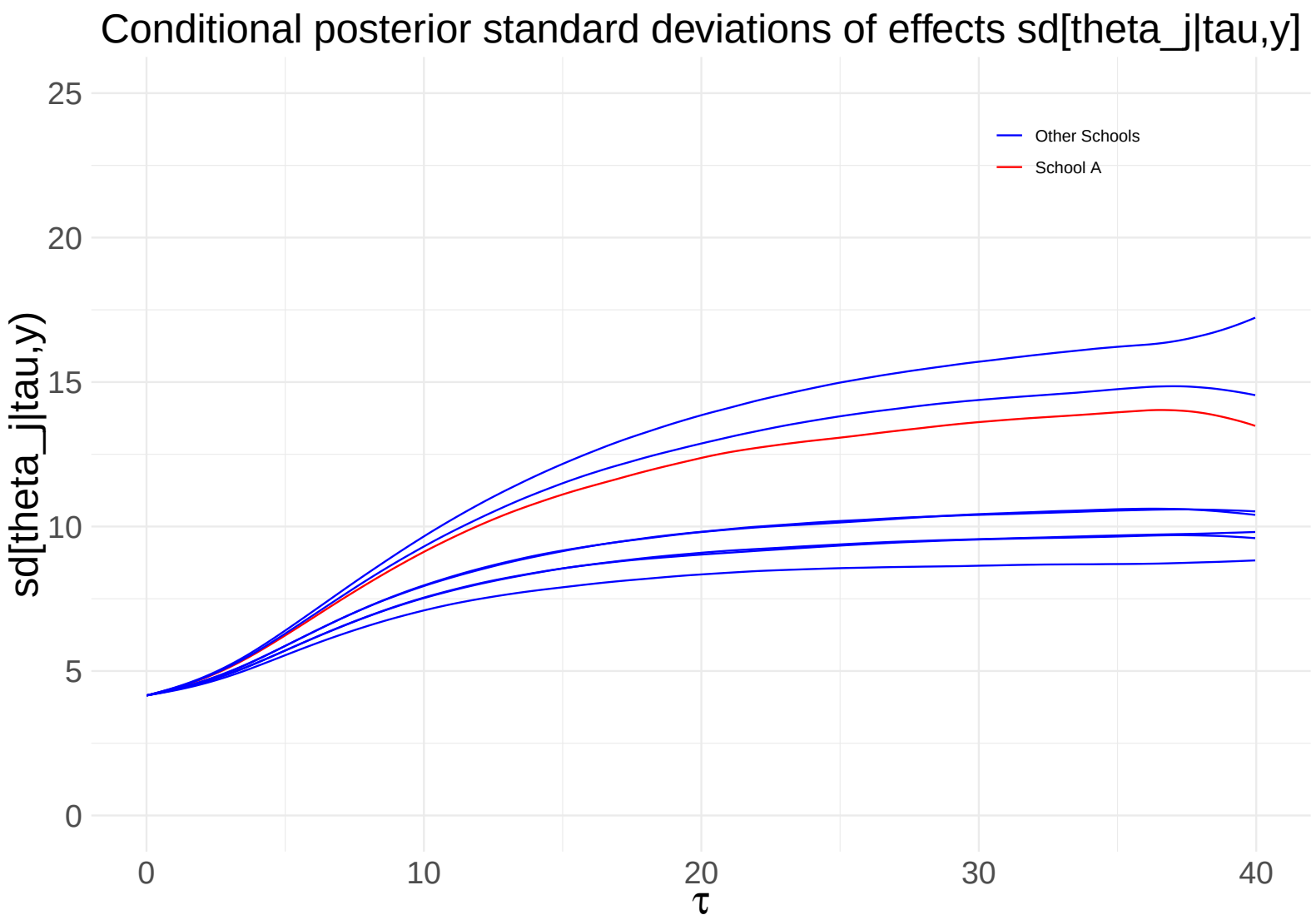
Eight schools: posterior summaries for hierarchical model



Eight schools: posterior summaries for hierarchical model



Eight schools: posterior summaries for hierarchical model



Eight schools: posterior summaries for hierarchical model

- In the plot for the marginal posterior $\pi(\tau|y)$, $\tau = 0$ is the most likely value (no variation in θ, complete pooling).
- Conditional posterior means $\mathrm{E}(\theta_j|\tau, y)$ are displayed as functions of τ: for most of the likely values of τ, the estimated effects are relatively close together: as τ becomes larger (more variability among schools), the estimates approach the separate analysis results.
- Conditional standard deviations $\mathrm{sd}(\theta_j|\tau, y)$ become larger as τ increases.

Eight schools: discussion

Comments:

- The general conclusion from these posterior summaries is that an effect as large as 28.4 points (school A) in any school is unlikely. For the likely values of τ, the estimates in all schools are substantially less than 28 points.
- To sum up, the Bayesian analysis of this example not only allows straightforward inferences about many parameters, but provides posterior inferences that account for the partial pooling as well as the uncertainty in the hyperparameters.
- We have still to investigate the role of the prior for the population standard deviation τ.

Eight schools: priors for τ^2

As we have already seen in other situations, assigning a prior may have a substantial effect on the final posterior inferences.

In this example, τ^2 governs the extent of variation between the schools: which are some suitable priors?

We review three choices:

$$\tau \sim \text{Uniform}(0, 100) \tag{14}$$

$$\tau^2 \sim \text{InvGamma}(0.01, 0.01) \tag{15}$$

$$\tau \sim \text{HalfCauchy}(0, 2.5) \tag{16}$$

Eight schools: priors for τ^2

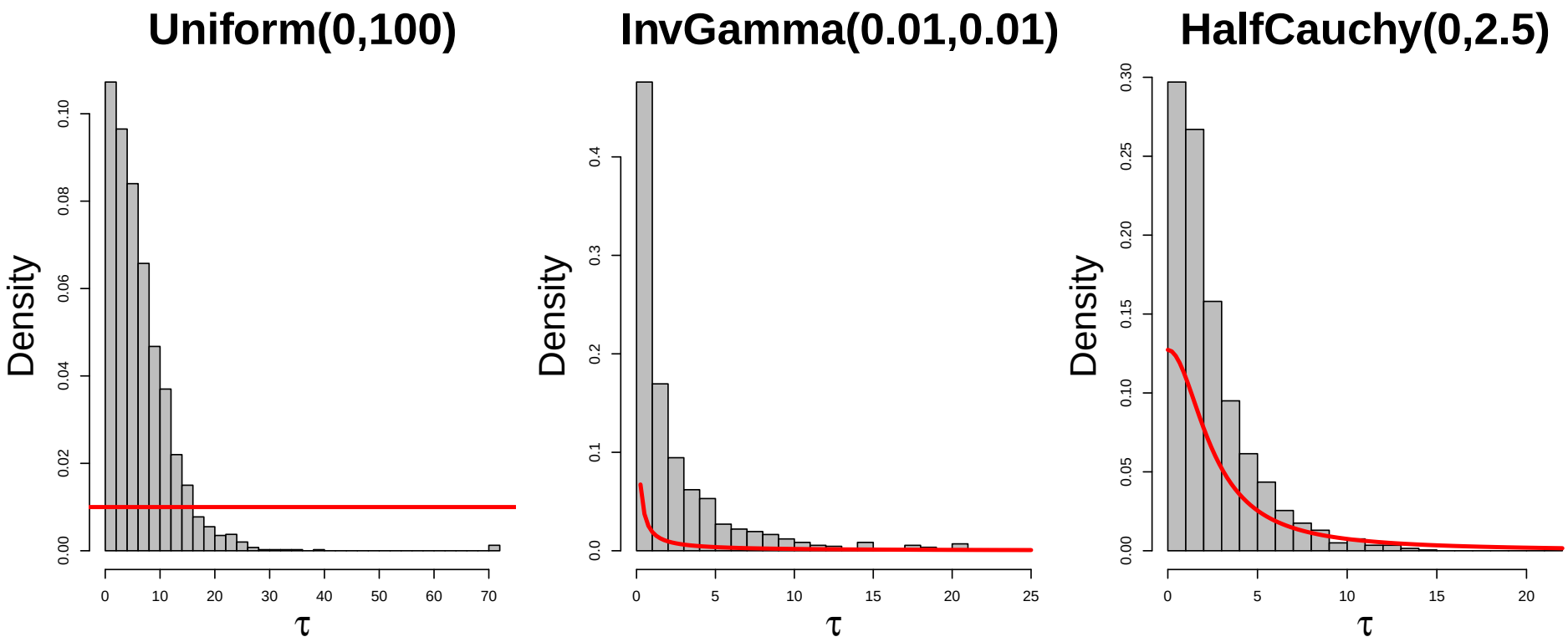


Figure: Marginal posterior (histograms) vs priors (solid red lines)

Eight schools: priors for τ^2

- Uniform The data show support for a range of values below $\tau = 20$, with a slight tail after that, reflecting the possibility of larger values, which are difficult to rule out given that the number of groups J is only 8 (that is, not much more than the $J = 3$ required to ensure a proper posterior density with finite mass in the right tail)
- Inverse gamma This prior distribution is sharply peaked near zero and further distorts posterior inferences, with the problem arising because the marginal likelihood for τ^2 remains high near zero. Moreover, the posterior is quite sensitive to the choices of the hyperparameters (try!)
- Half Cauchy less likely to dominate the inferences

Eight schools: priors for τ^2

Comments:

- The InvGamma prior is not at all noninformative for this problem since the resulting posterior distribution remains highly sensitive to the choice of the hyperparameters.
- The Uniform prior distribution seems fine for the 8-school analysis, but problems arise if the number of groups J is much smaller, in which case the data supply little information about the group-level variance, and a noninformative prior distribution can lead to a posterior distribution that is improper or is proper but unrealistically broad.

Indice

Hierarchical logistic regression

1988 US polls

We choose a single outcome—the probability that a respondent prefers the Republican candidate Bush against the democrat Dukakis for president—as estimated by a logistic regression model from a set of seven CBS News polls conducted during the week before the 1988 presidential election.

We introduce multilevel logistic regression including two individual 0-1 predictors—`female` and `black`—and the 51 states:

$$\begin{aligned} \Pr(y_i = 1) =& \text{logit}^{-1}(\alpha_{j(i)} + \beta^{\text{female}}\text{female}_i + \beta^{\text{black}}\text{black}_i), \quad i = 1, \ldots, n \\ \alpha_j \sim& \mathcal{N}(\mu_\alpha, \tau^2_{\text{state}}), \; j = 1, \ldots, 51 \end{aligned} \tag{17}$$

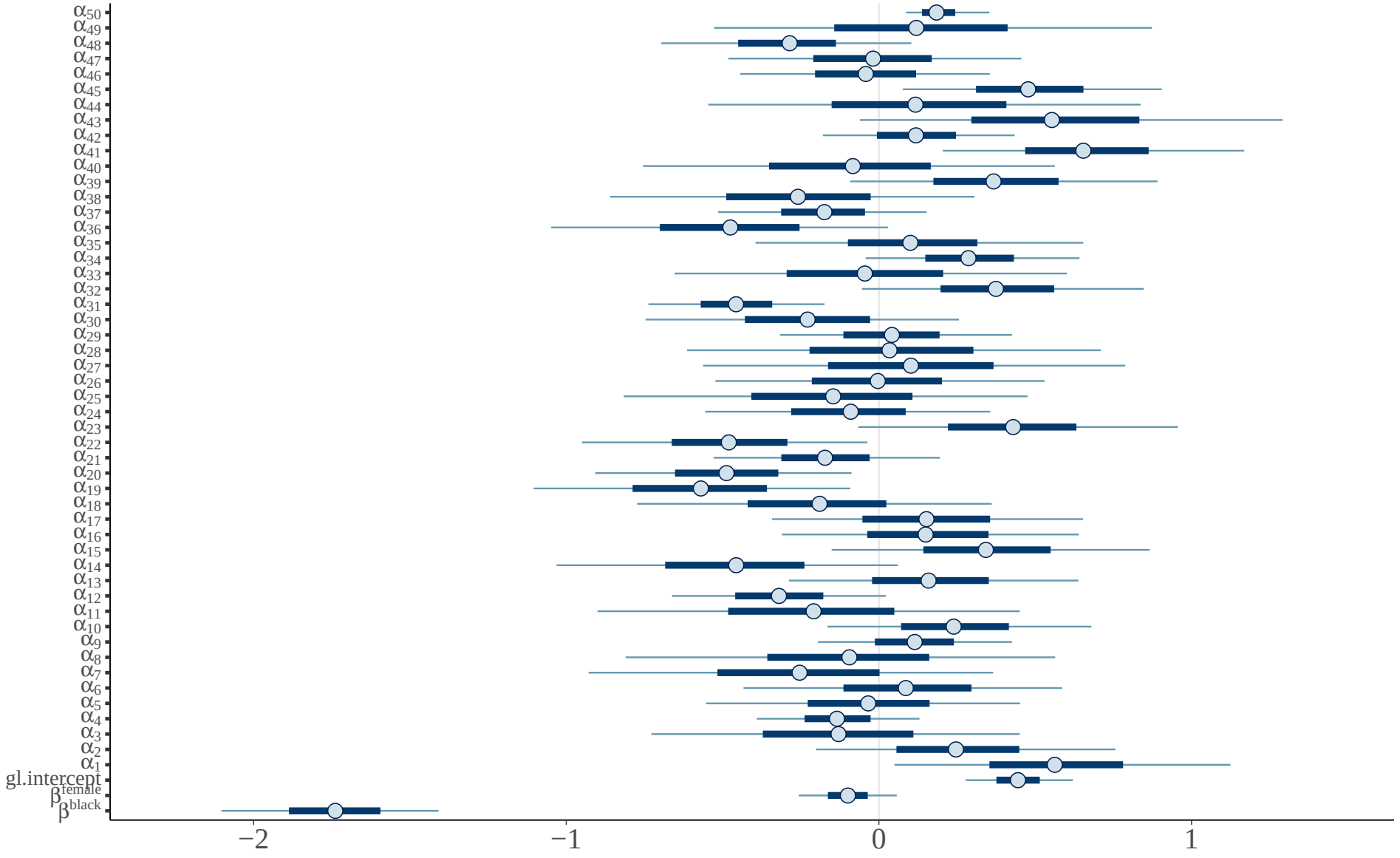
where $j(i)$ is the state index.

1988 US polls. Varying-intercept model

```
stan_glmer
 family:       binomial [logit]
 formula:      y ~ black + female + (1 | state)
 observations: 2015
------
            Median MAD_SD
(Intercept)  0.4    0.1
black       -1.7    0.2
female      -0.1    0.1

Error terms:
 Groups Name        Std.Dev.
 state  (Intercept) 0.45
Num. levels: state 49
```

The state variation is estimated at $\hat{\tau}_{\text{state}} = 0.45$.

1988 US polls. Varying-intercept model



1988 US polls. Varying-intercept model

Parameters' interpretation

- The coefficient β^{black} reports a posterior estimate of -1.7: `black` is a categorical variable (coded as 1 for black people, 0 otherwise). A difference of 1 unit in this predictor has a linear effect of -1.7 on the logit probability of supporting Bush. In terms of odds ratios, being black gives an odds ratio of $\exp(-1.7) \approx 0.18$, causing a decrease in the odds of approximately 0.82 (82%).
- The coefficient β^{female} is estimated at -0.1. `female` is a categorical predictor (1 for women, 0 otherwise). Being a woman has an effect of -0.1 on the logit probability of supporting Bush. OR interpretation: $\exp(-0.1) \approx 0.9$, decrease in the odds of approx. 10%.

Be aware: understanding and interpreting model estimates is the first step! Ask, ask, ask yourself whether your estimates make sense...

Hierarchical logistic regression: 1988 US polls

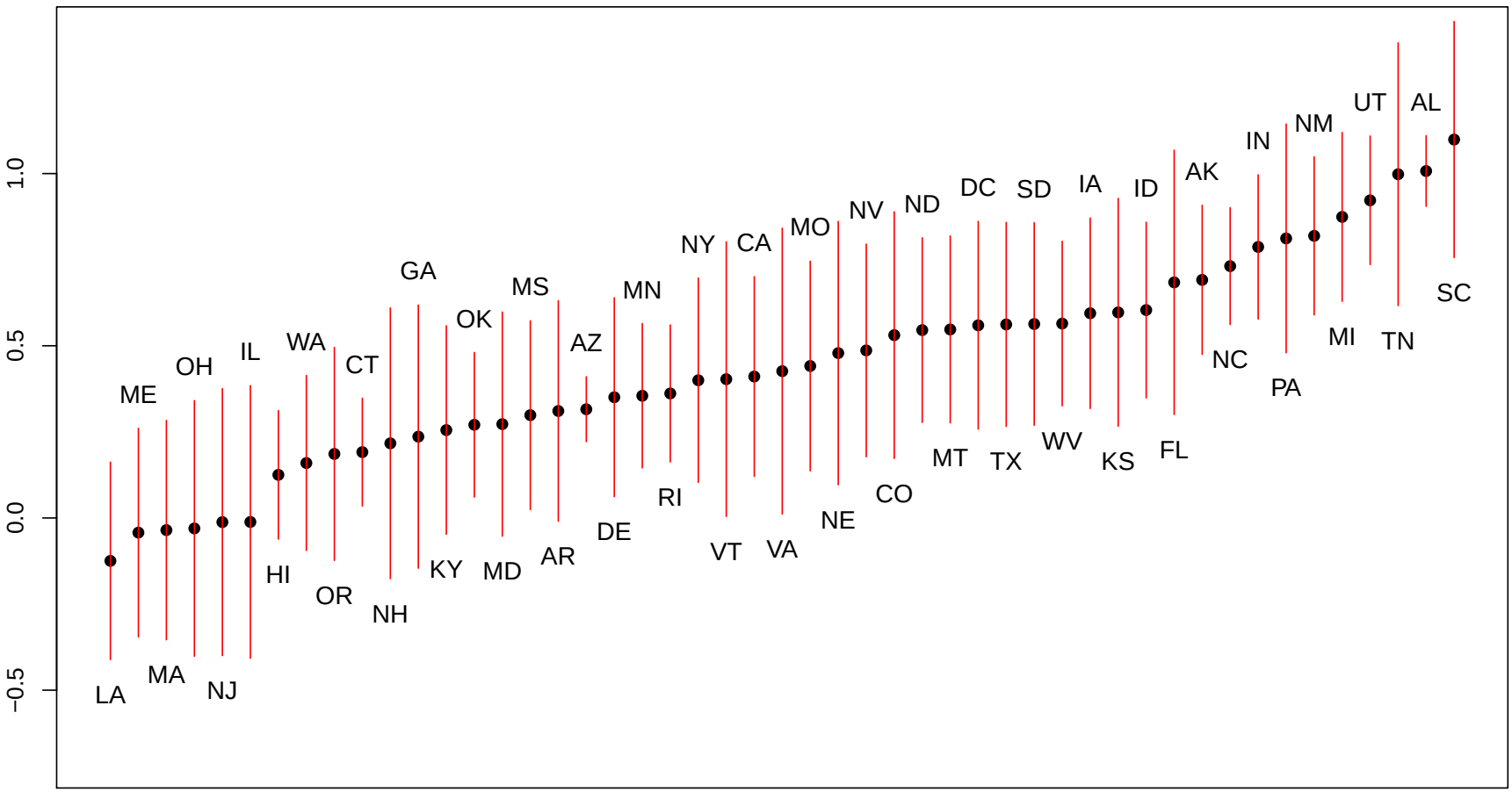
Many issues arise when you fit a model:

- Interpret your results. Do they make sense?
- Produce some plots for your estimates.
- Check your model. Is your model plausible, according to the data that you have? To be continued...
- Augment your model, if necessary: predictors, random effects,etc.
- Compare your model with other competing models. Is your model better than the others? Use AIC, DICC, LOIIC...To be continued...
- Use your model to make predictions.

Being a modeller represents a compromise between a mathematician and an artist. You can tremble between these two extremes.

Hierarchical logistic regression: 1988 US polls

Random effects α for the states: post. means $\pm$ s.e.



States

1988 US polls. Varying-intercept and slope

We could ask ourself: is also the slope for the female varying in some states? Maybe, the women Bush preference for Bush in Alabama is rather different than the same support in New Jersey...

We propose a second model, a *varying-intercept and slope model*:

$$\begin{aligned}\Pr(y_i = 1) =& \text{logit}^{-1}(\alpha_{j(i)} + \beta^{\text{female}}_{j(i)}\text{female}_i + \beta^{\text{black}}\text{black}_i), \quad i = 1, \ldots, n \\ \begin{pmatrix}\alpha_j \\ \beta_j\end{pmatrix} \sim& \mathcal{N}\left(\begin{pmatrix}\mu_\alpha \\ \mu_\beta\end{pmatrix}, \begin{pmatrix}\tau^2_\alpha & \rho\tau_\alpha\tau_\beta \\ \rho\tau_\alpha\tau_\beta & \tau^2_\beta\end{pmatrix}\right), \quad j = 1, \ldots, 51,\end{aligned} \tag{18}$$

where τ^2_α and τ^2_β are the variances for the intercepts and the slopes, repsectively, and ρ is the correlation coefficients between α and β.

1988 US polls. Varying-intercept and slope

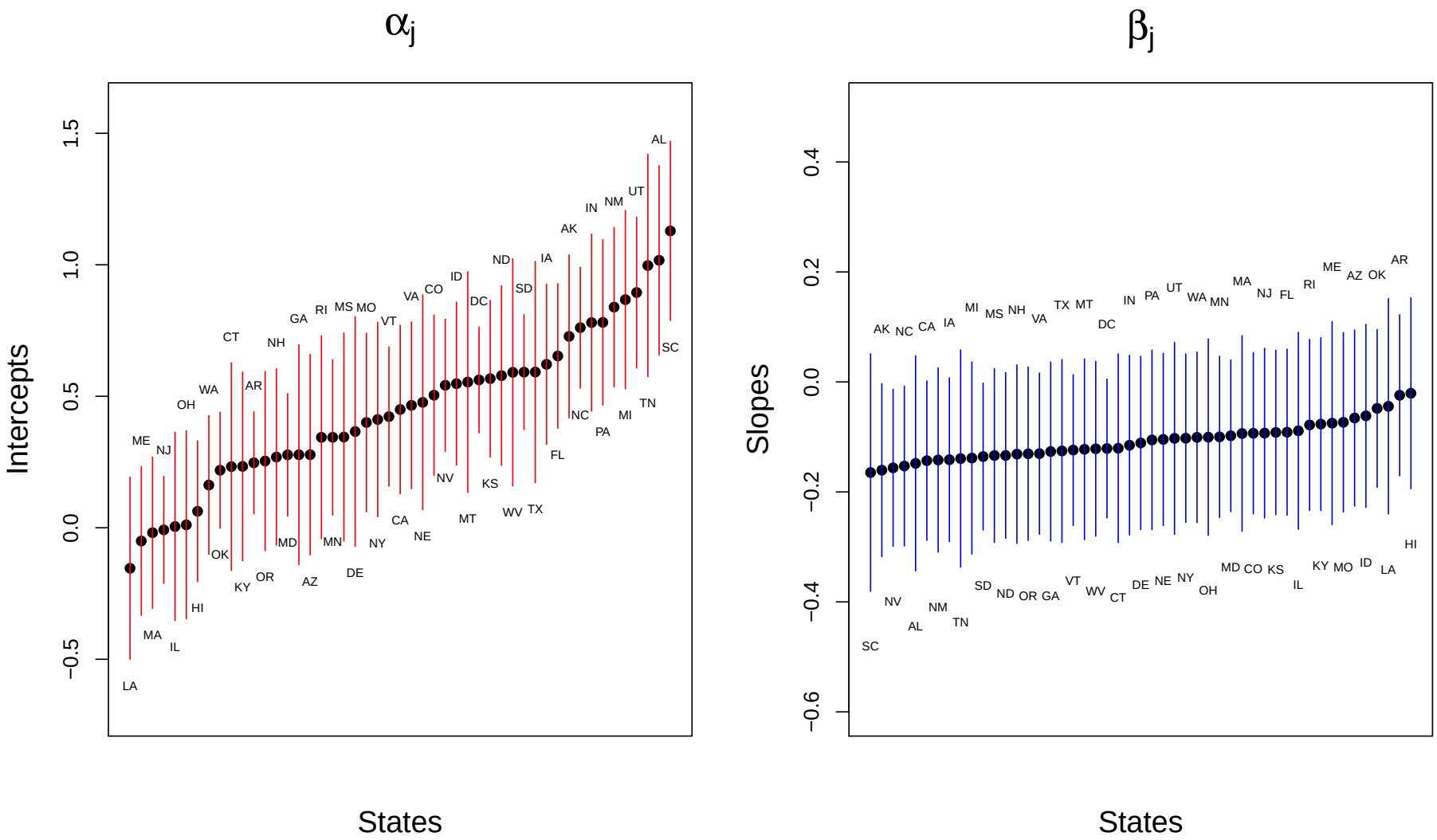
```
stan_glmer
 family:       binomial [logit]
 formula:      y ~ black + female + (1 + female | state)
 observations: 2015
------
            Median MAD_SD
(Intercept)  0.5    0.1
black       -1.7    0.2
female      -0.1    0.1

Error terms:
 Groups Name        Std.Dev. Corr
 state  (Intercept) 0.47
        female      0.23     -0.40
```

1988 US polls. Varying-intercept and slope

Parameters' interpretation:

- $\hat{\tau}_\alpha = 0.47$, the variation between the β^{female}, $\hat{\tau}_\beta$, is 0.23, whereas $\hat{\rho} = -0.4$. Thus, there is negative correlation between the states' effects and the female effects.
- Other parameters are almost unchanged with respect to the varying-intercept model.

1988 US polls. Varying-intercept and slope



Model comparison

We should start assessing the goodness of fit of our models. In Bayesian inference, the main tools to compare models are the penalized likelihood criteria: AIC, DIC, BIC,...

We consider here also an extension of AIC based on cross validation, LOOIC, available via the `loo` package.

The meaning is the same: the lower is the value of one among these criteria, and the better is the model fit.

Model comparison

```
lpd1 <- log_lik(M1.rstanarm)
loo1 <- loo(lpd1)
lpd2 <- log_lik(M2.rstanarm)
loo2 <- loo(lpd2)
c(loo1$looic, loo2$looic)
```

```
[1] 2649.373 2651.668
```

The varying-intercept and slope model does not improve over the fit of the varying intercept model. The simpler the better!

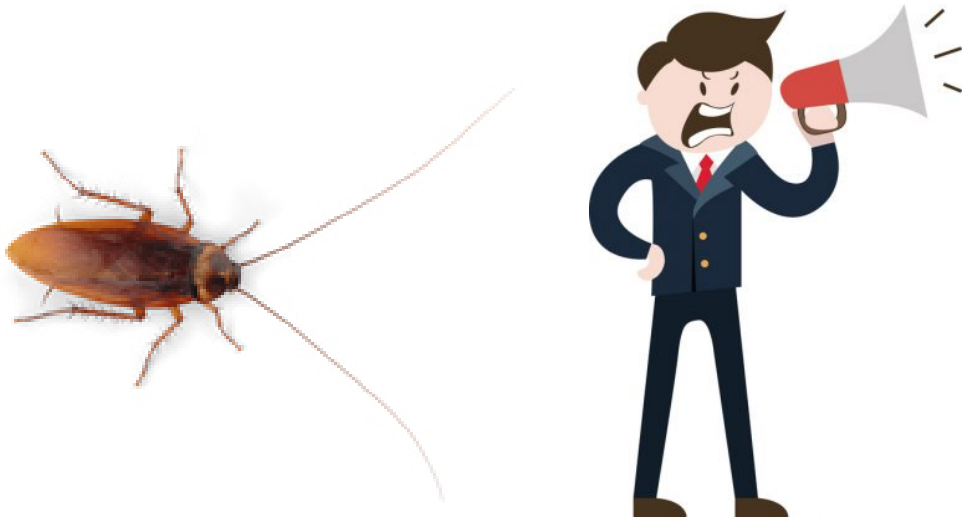
We could try to extend our model and, eventually, increase the goodness of fit (to be continued).

Indice

Discrete data regression: cockroaches data

Cockroaches data

A company that owns many residential buildings throughout New York City tells that they are concerned about the number of cockroach complaints that they receive from their 10 buildings. They provide you some data collected in an entire year for each of the buildings and ask you to build a model for predicting the number of complaints over the next months.



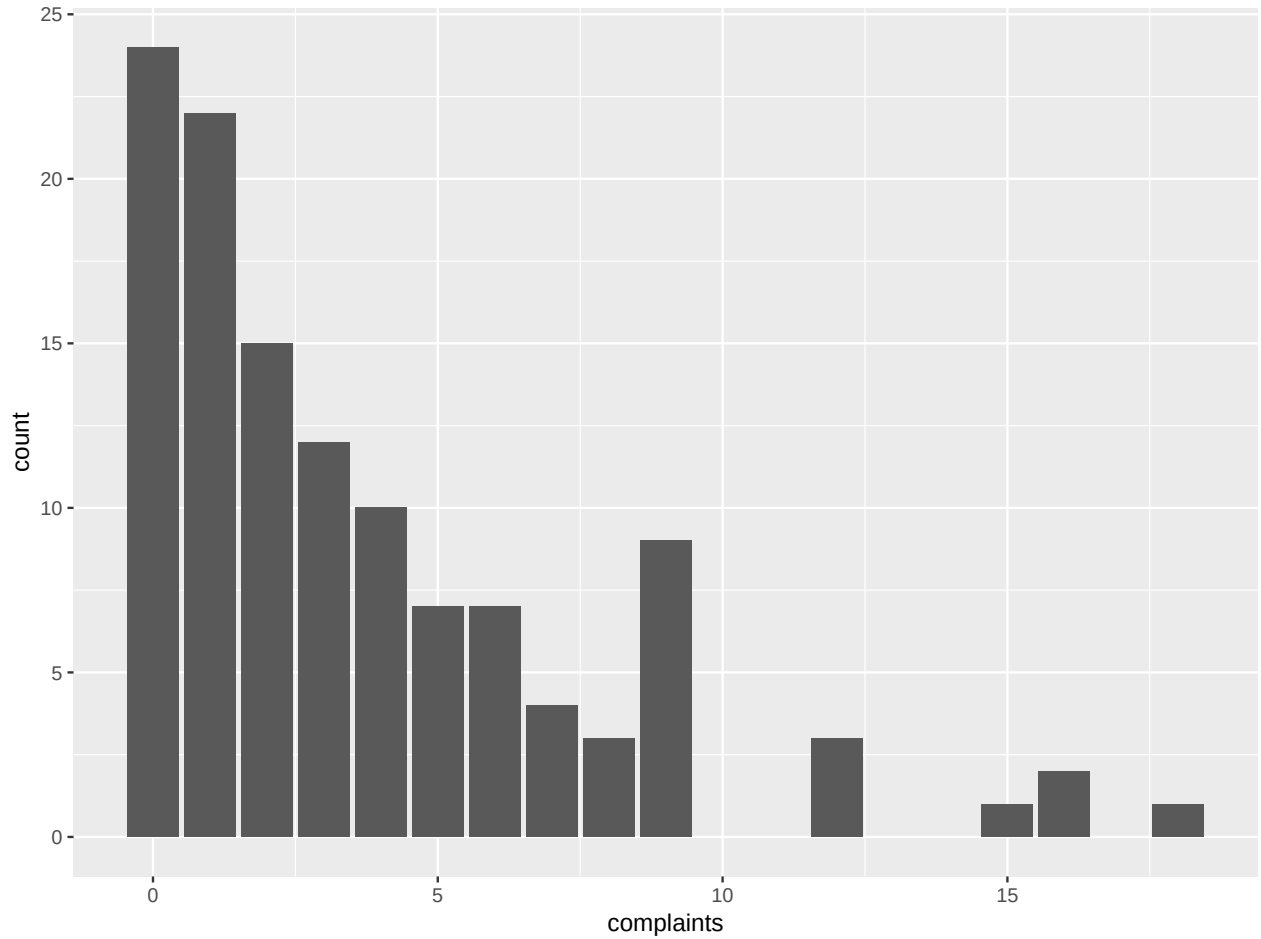
Discrete data regression: cockroaches data

We have access to the following fields (`pest_data.RDS`):

- `complaints`: Number of complaints per building in the current month
- `traps`: The number of traps used per month per building
- `live_in_super`: An indicator for whether the building has a live-in super
- `age_of_building`: The age of the building
- `total_sq_foot`: The total square footage of the building
- `average_tenant_age`: The average age of the tenants per building
- `monthly_average_rent`: The average monthly rent per building
- `floors`: The number of floors per building

Discrete data regression: cockroaches data

Let's make some plots of the raw data, such as the distribution of the complaints:



Poisson regression: cockroaches data

A common way of modeling this sort of skewed, single bounded count data is as a Poisson random variable. For simplicity, we will start assuming:

- ungrouped data, with no building distinction
- no time-trend structures

We use the number bait stations placed in the building, denoted below as `traps`, as explanatory variable. This model assumes that the mean and variance of the outcome variable `complaints` (number of complaints) is the same. For the i-th complaint, $i = 1, \ldots, n$, we have

$$\begin{aligned} \text{complaints}_i &\sim \text{Poisson}(\lambda_i) \\ \lambda_i &= \exp(\eta_i) \\ \eta_i &= \alpha + \beta\, \text{traps}_i \end{aligned}$$

Hierarchical Poisson regression

We can extend Poisson models encoding hierarchical structure. Consider again the cockroach regression, and consider now to include as many intercepts as buildings. Thus, for each complaint i we have:

$$\begin{aligned} \text{complaints}_{ib} &\sim \text{Poisson}(\lambda_{ib}) \\ \lambda_{ib} &= \exp(\eta_{ib}) \\ \eta_{ib} &= \alpha_{b(i)} + \beta \, \text{traps}_i + \beta_{\text{super}} \, \text{super}_i + \text{log\_sq\_foot}_i \\ \alpha_b &\sim \mathcal{N}(\mu, \tau_\alpha^2), \end{aligned} \tag{19}$$

where $b(i)$ is the nested index for the building where the i-th complaint is registered.

Further reading

Further reading

- Chapter 15 and 16 from *Bayesian Data Analysis*, A.Gelman et al.
- Chapter 11, 12, 13, 14, 15, 16 from *Data Analysis using Regression and Multilevel/Hierarchical models*, A. Gelman and J. Hill.

Indice

Motivations

Once we have accomplished the first two steps of a Bayesian analysis—constructing a probability model and computing the posterior distribution of all estimands—we should not ignore the relatively easy step to assessing the fit of the model to the data and to our substantive knowledge.

It is worth to remind that we use the term *model* to encompass the sampling distribution, the prior distribution, any hierarchical structure, and issues such as which explanatory variables have been included in a regression.

Motivations

It is not correct to ask 'Is our model true or false?', since probability models in most data analysis will not be perfectly true.

The more relevant question is 'Do the model's deficiencies have a noticeable effect on the substantive inferences?'. Remember the George E.P. Box quote:

All models are wrong, but some are useful.

How to judge when assumptions of convenience can be made safely is a central task of Bayesian sensitivity analysis. Failures in the model lead to practical problems by creating false inferences about estimands of interest.

The external validation paradigm

We can check a model by external validation using the model to make predictions about future/hypothetical data, and then collecting those data and comparing to their predictions.

Bayesian analysis use *posterior predictive checking* to check the joint posterior predictive distribution of future data given the data at hand, $p(\tilde{y}|y)$.

The idea is the following: if the model fits, then replicated data under the model should look similar to observed data. To put in another way, the observed data should look *plausible* under the posterior predictive distribution.

Posterior predictive checking

The basic technique for checking the fit of a model is to draw simulated values from the joint posterior predictive distribution of replicated data and compare these samples to the observed data. Any systematic differences between the simulation and the data indicate potential failings of the model.

We define y^{rep} as the replicated data that *could have been observed*. We distinguish between y^{rep} and $\tilde{y}$:

- $\tilde{y}$ is any future observable value or vector of observable quantities (out-of-sample replication)
- y^{rep} is specifically a replication just like y (in-sample replication)

Posterior predictive checking

The posterior predictive distribution of y^{rep} given the current state of knowledge is:

$$p(y^{\text{rep}}|y) = \int p(y^{\text{rep}}|\theta)\pi(\theta|y)d\theta. \tag{20}$$

We measure the *discrepancy* between model and data by defining some test quantities $T(y, \theta)$, the aspects of the data we wish to check. T is a scalar summary of parameters and data that is used to compare data to predictive simulations.

Test (or discrepancy) quantities play the role in Bayesian model checking that test statistics play in classical testing.

Classical p-value vs Bayesian p-value

Lack of fit of the data with respect to the ppd can be measured by the *tail-area* probability, or p-value, of the test quantity, and computed using posterior simulations of (θ, y^{rep}). We define the p-value mathematically, first for classical inference.

The classical p-value for the test statistic $T(y)$ is

$$p_C = \Pr(T(y^{\text{rep}}) \geq T(y)|\theta), \qquad (21)$$

where the probability is taken over the distribution of y^{rep} with θ fixed.

Classical p-value vs Bayesian p-value

The Bayesian p-value is defined as the probability that the replicated data could be more extreme than the observed data, as measured by the test quantity:

$$p_B = \Pr(T(y^{\text{rep}}, \theta) \geq T(y, \theta)|y), \qquad (22)$$

where the probability is taken over the posterior distribution of θ and the ppd of y^{rep}:

$$p_B = \int \int |T(y^{\text{rep}}, \theta) \geq T(y, \theta)| p(y^{\text{rep}}|\theta)\pi(\theta|y) dy^{\text{rep}} d\theta = \int p_C \pi(\theta|y) d\theta,$$

where $|\cdot|$ denotes the indicator function. Thus the Bayesian p-value is an average of the classical p-value over θ.

Bayesian p-values in practice

In practice, we usually compute the ppd (20) using *simulation*. Specifically, this happens with a two-steps procedure:

- Suppose to have S simulations $\theta^{(s)}$, $s = 1, \ldots, S$ from the posterior distribution.
- We generate S draws $y^{\text{rep}(s)}$ from $p(y^{\text{rep}}|\theta^{(s)})$.
- We compute now $T(y^{\text{rep}}, \theta)$: the estimated Bayesian p-value for (22) is the proportion of these S simulations for which the test quantity equals or exceeds its realized values; that is, for which $T(y^{\text{rep}(s)}, \theta) \geq T(y, \theta)$.

Bayesian p-values in practice

Thus, we almost never have a closed form for (20). What we do, is performing something similar to Monte Carlo simulation, and approximating the integral in (20) by the sum over the S draws:

$$\sum_{s=1}^{S} p(y^{\text{rep}(s)}|\theta^{(s)})\pi(\theta^{(s)}|y). \tag{23}$$

The resulting estimation of (22) is then equal to:

$$\frac{1}{S}\sum_{s=1}^{S} |T(y^{\text{rep}(s)}, \theta^{(s)}) \geq T(y, \theta^{(s)})|. \tag{24}$$

Eight schools: model checking

Consider again the eight schools example about the effects of special coaching programs on test scores in each of eight high-schools:

$$y_{ij} \sim \mathcal{N}(\theta_j, \sigma_y^2)$$
$$\theta_j \sim \mathcal{N}(\mu, \tau^2)$$

The example is based on many assumptions:

1. normality of the estimates y_j given θ_j and σ_j, where the σ_j are assumed known;
2. exchangeability of the prior distribution of the θ_j's;
3. normality of the prior distribution of each θ_j given μ and τ.

The exchangeability assumption means that we will let the data tell us about the relative ordering and similarity of effects in the eight schools.

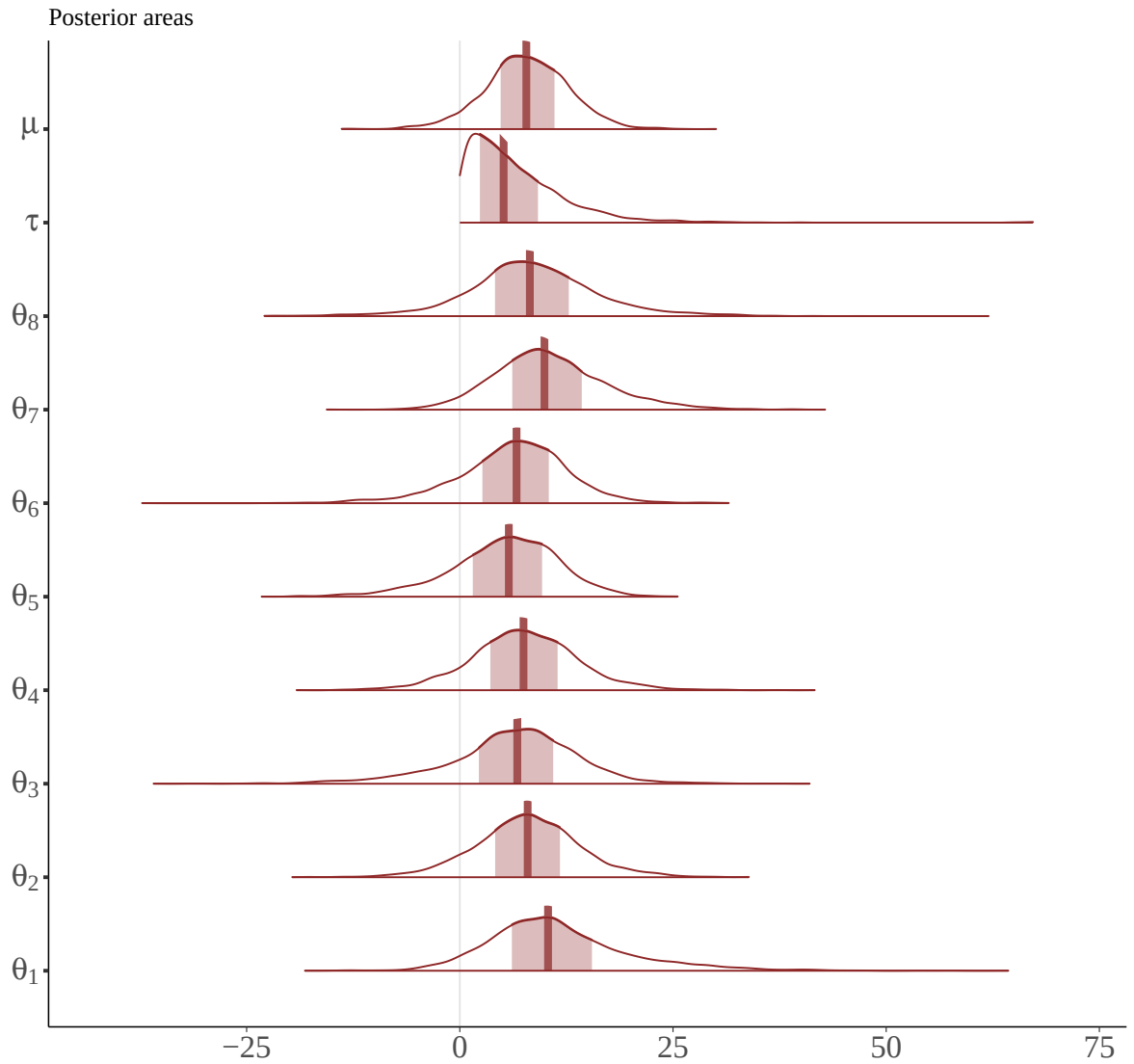
Eight schools: Stan model

```
data {
  int<lower=0> J; // number of schools
  real y[J]; // estimated treatment effects
  real<lower=0> sigma[J]; // s.e. of effect estimates
}
parameters {
  real mu;
  real<lower=0> tau;
  real eta[J];
}
transformed parameters {
  real theta[J];
  for (j in 1:J)
    theta[j] = mu + tau * eta[j];
}
```

Eight schools: Stan model (cont.)

```
model {
  target += normal_lpdf(eta | 0, 1);
  target += normal_lpdf(y | theta, sigma);
}
generated quantities {
  real y_rep[J];
  for (j in 1:J)
     y_rep[j] = normal_rng(theta[j], sigma[j]);
}
```

Eight schools: estimation



Replications

We simulate the ppd of a hypothetical replication of the experiment. In Stan, we do this by the cycled instruction:

```
y_rep[j] = normal_rng(theta[j], sigma[j]);
```

We have now S draws for the replicated vector $y^{\text{rep}} = (y_1^{\text{rep}}, \ldots, y_8^{\text{rep}})$. We should now visualize this distribution over the S draws and detect eventual deficiencies of the model.

We will perform many kinds of pp checks. The main tool here is visualization. All the plots are obtained with the `bayesplot` package.

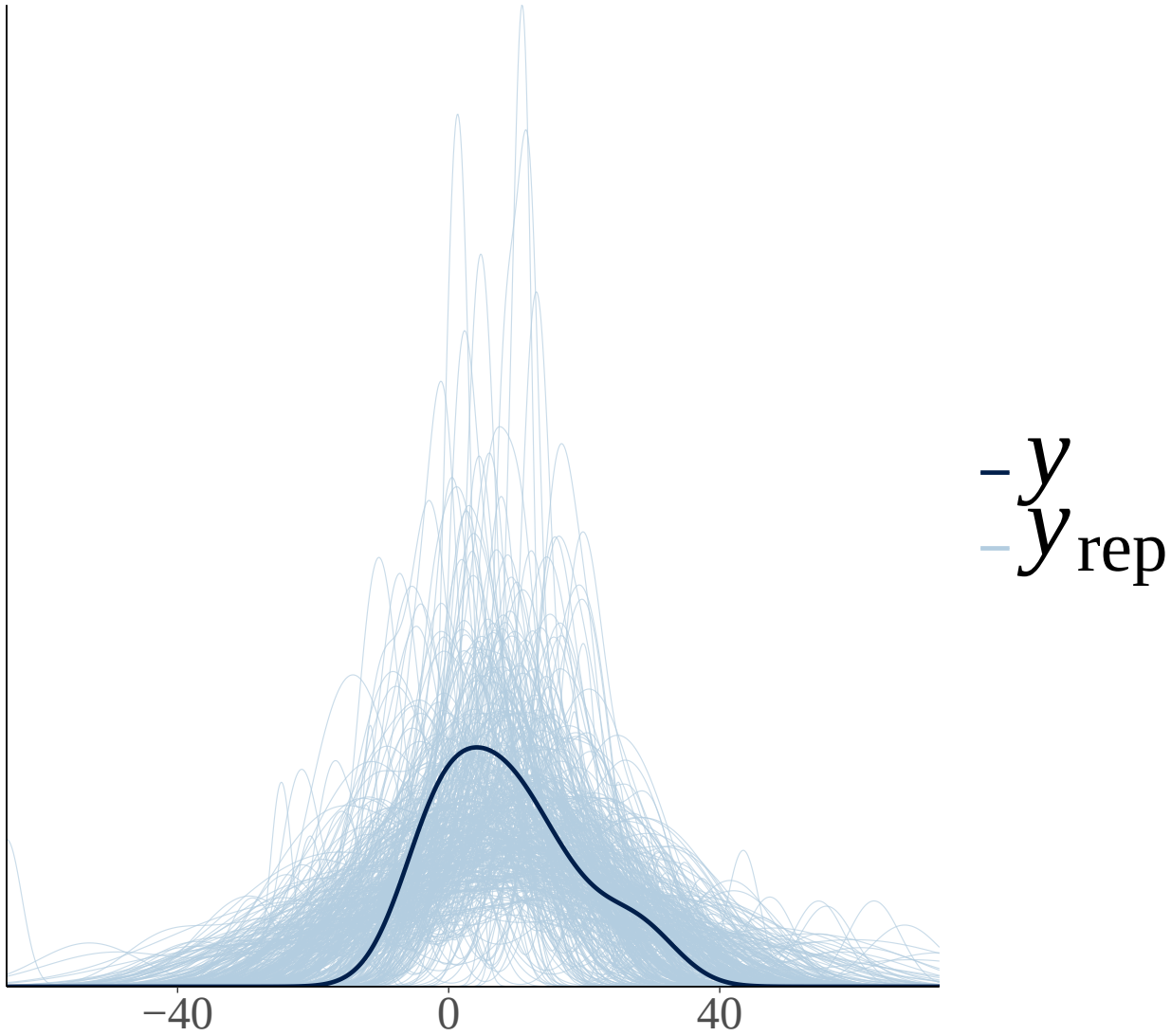
Graphical posterior predictive checks

The basic idea of graphical model checking is to *display the data alongside simulated data* from the fitted model, and to look for systematic discrepancies between real and simulated data. Essentially, we may recognize three kinds of graphical display:

- direct display of all the data
- display of data summaries or parameter inferences
- graphs of residuals or other measures of discrepancy between model and data.

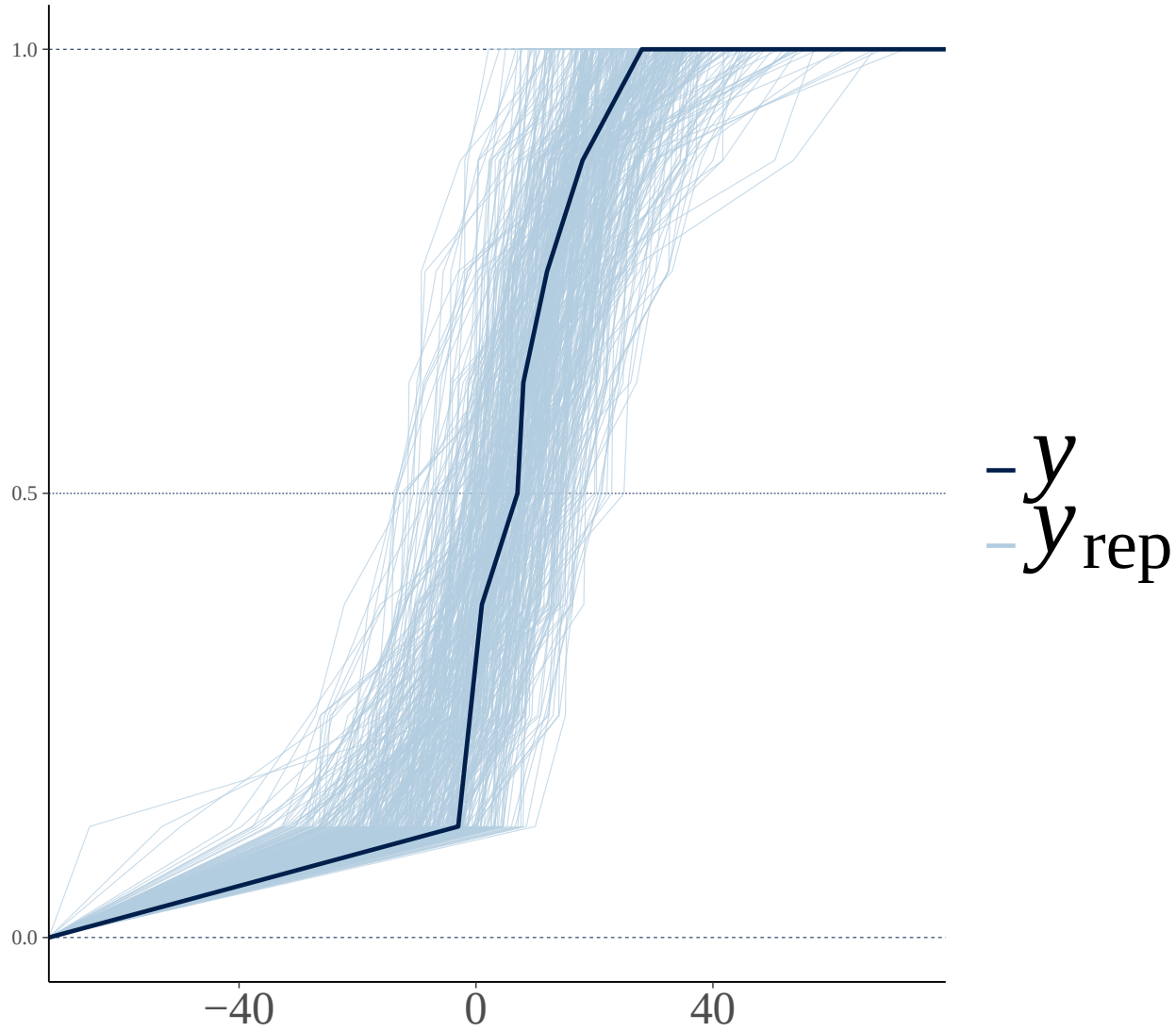
Check 1: distribution of replicated data vs real data

```
ppc_dens_overlay(y,y_rep)
```



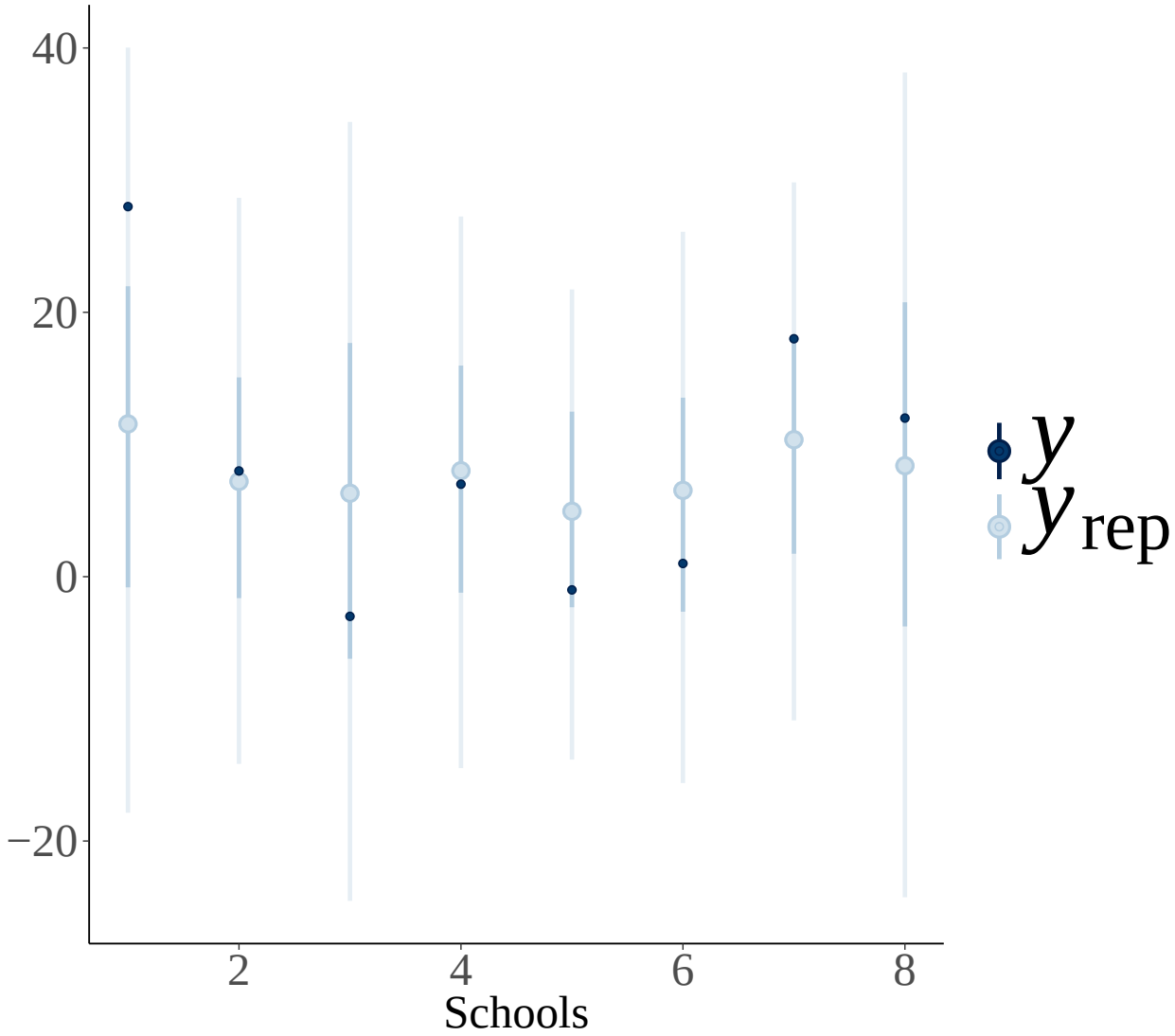
Check 2: empirical distribution function

```
ppc_ecdf_overlay(y,y_rep)
```



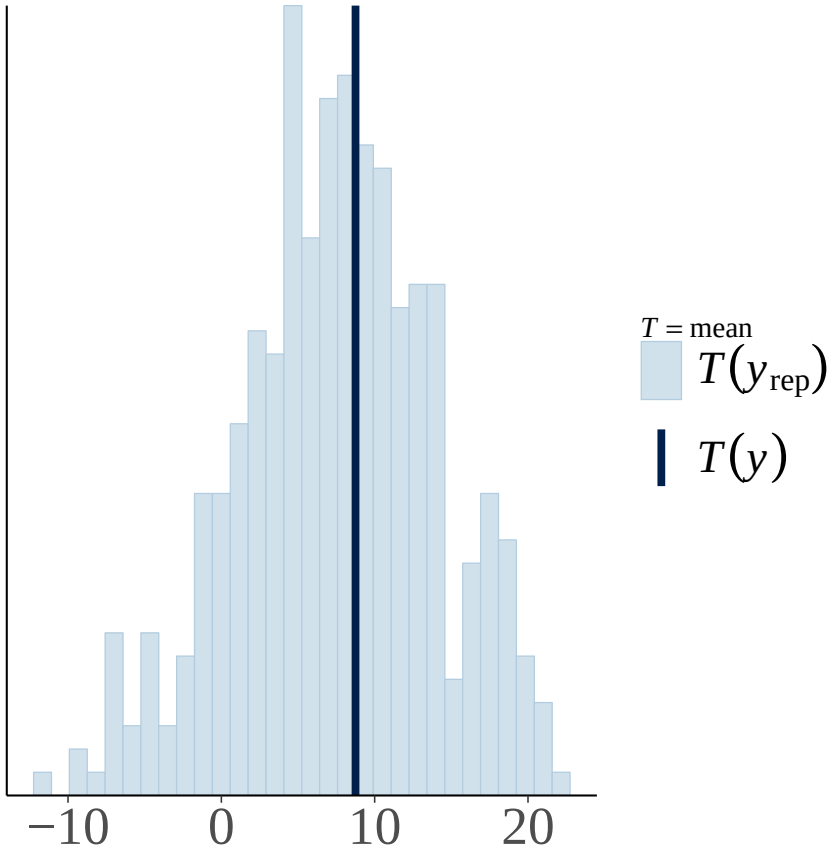
Check 3: predictive intervals vs observed values

```
ppc_intervals(y,y_rep)
```

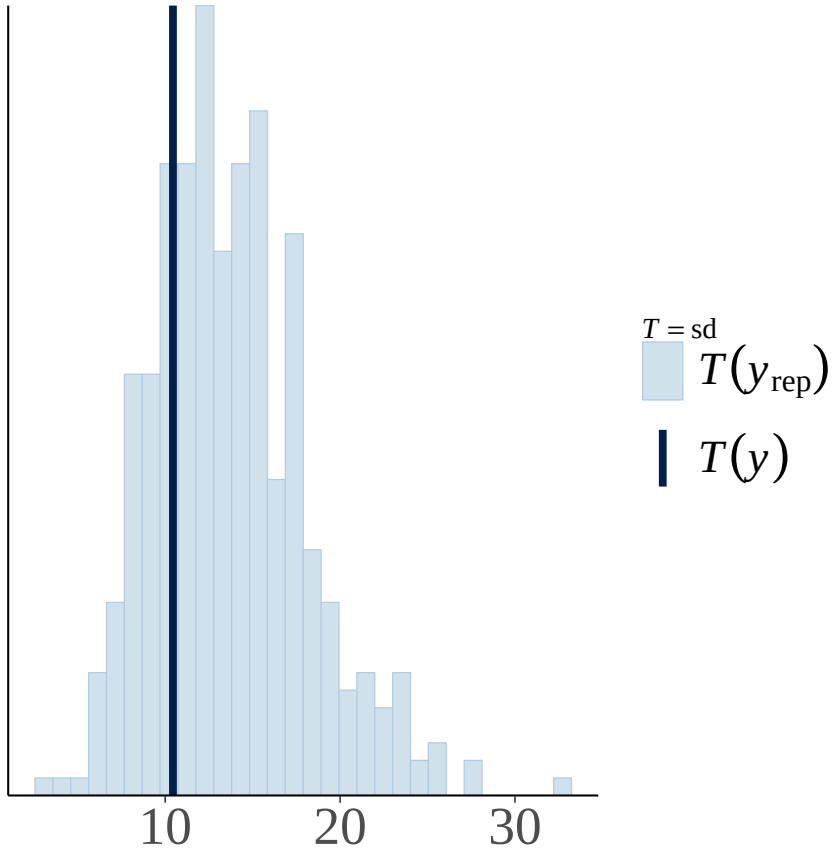


Check 4: statistics

```
ppc_stat(y,y_rep)
```



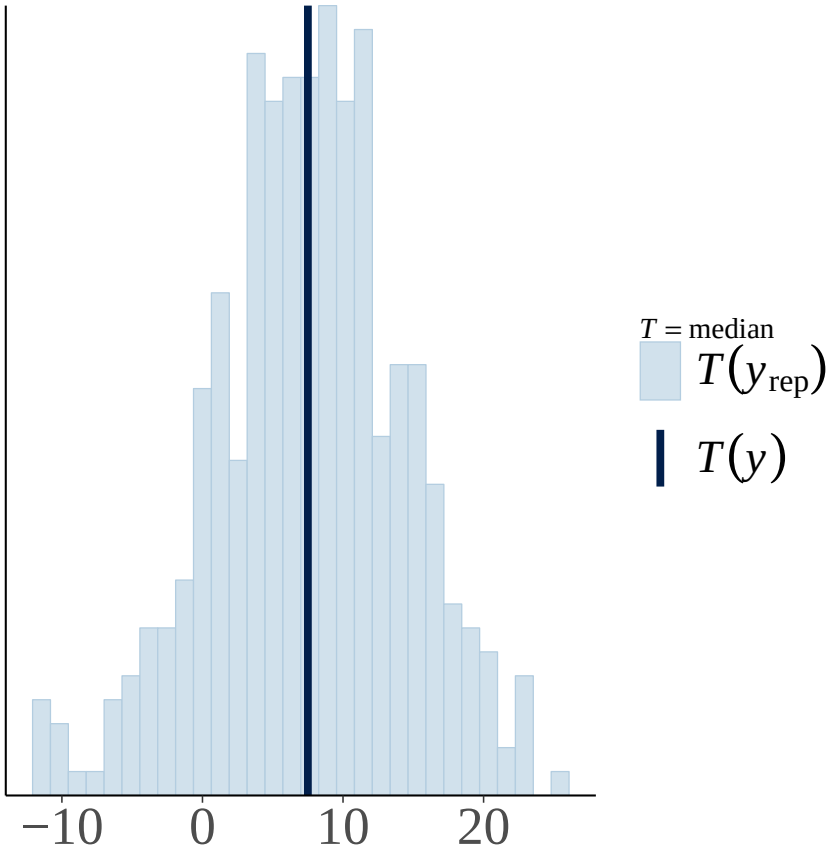
(a) $T(y) = \bar{y}$



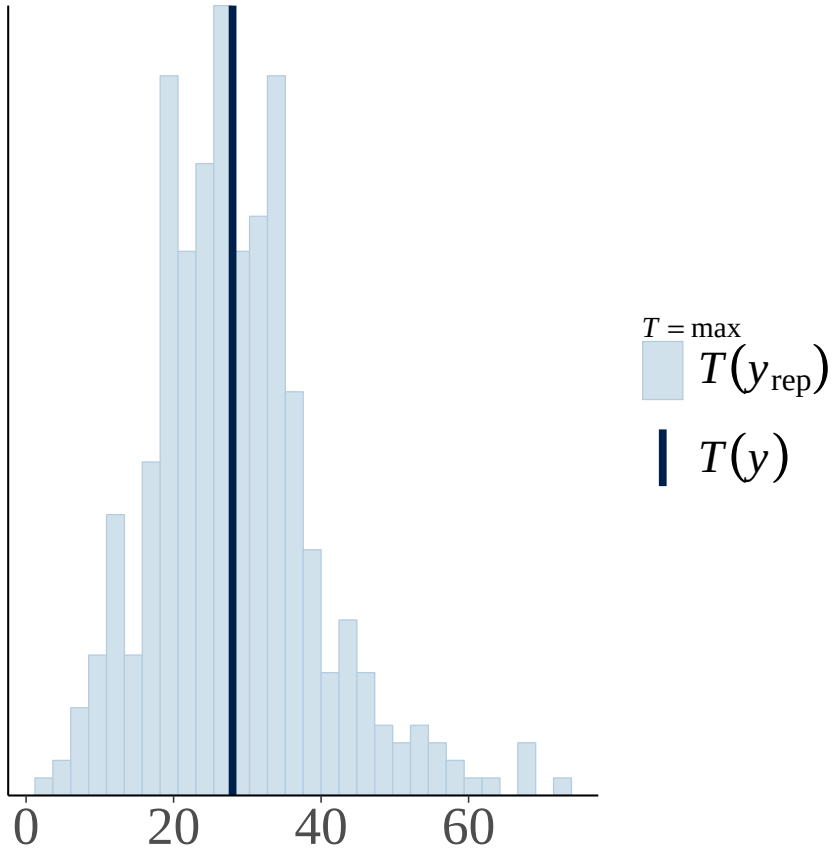
(b) $T(y) = \mathrm{sd}(y)$

Check 4: statistics

```
ppc_stat(y,y_rep)
```



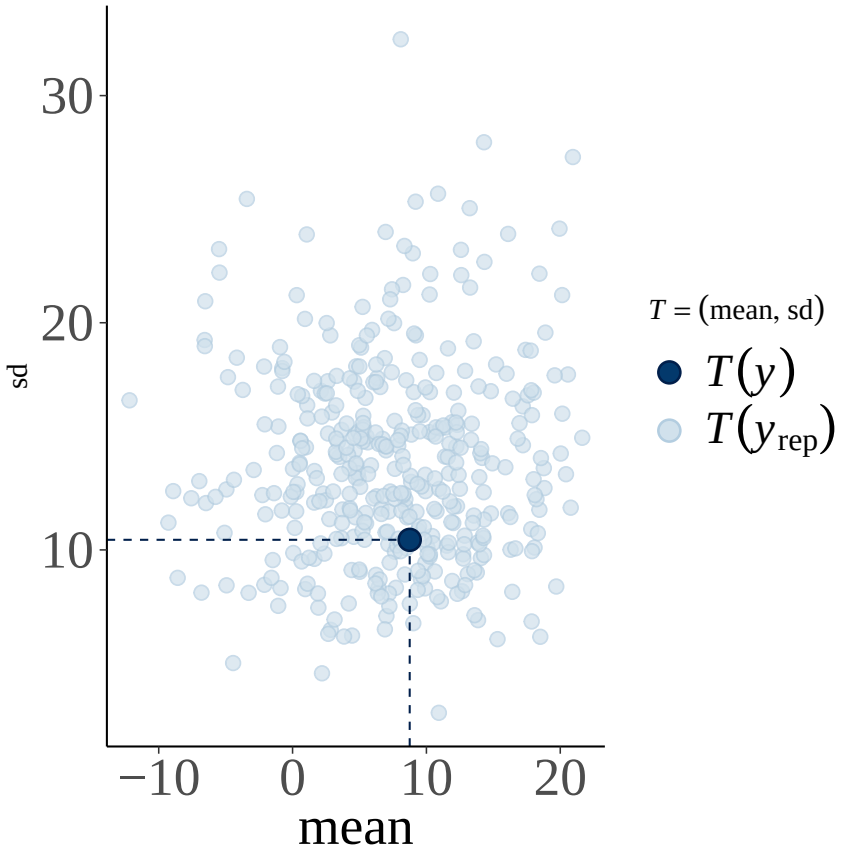
(c) $T(y) = \text{Me}(y)$



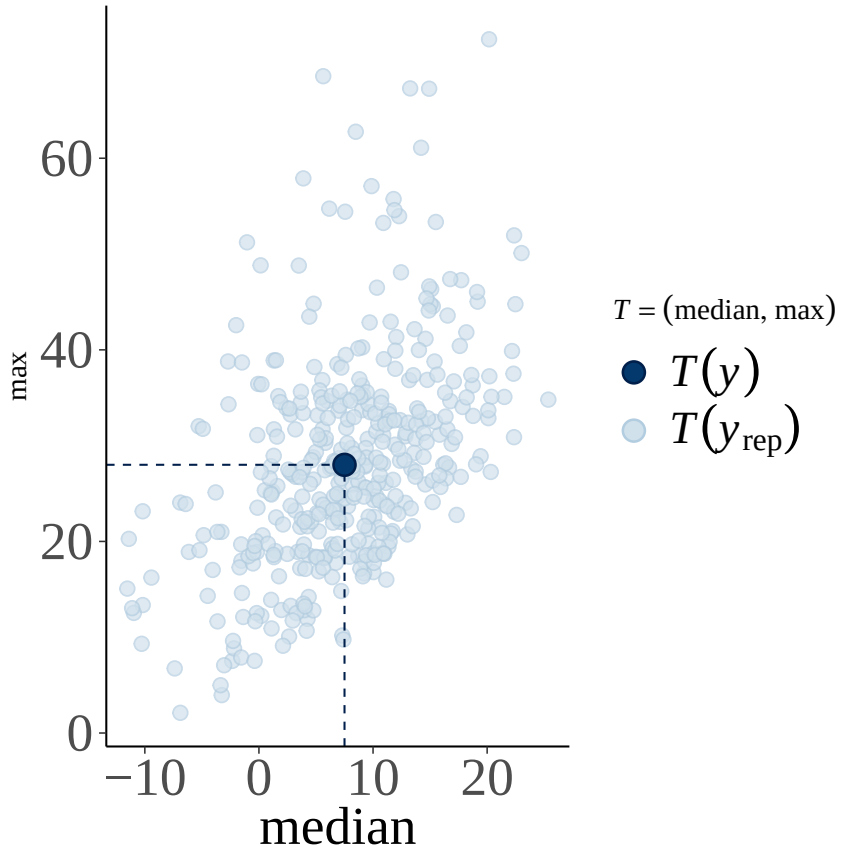
(d) $T(y) = \max(y)$

Check 5: bivariate statistics

```
ppc_stat_2d(y,y_rep)
```



(e) Mean and sd



(f) Median and max

Comments for the pp checks

- The graphical summaries suggest that the model generates predicted results similar to the observed data in the study. Observed test statistics fall within their replicated distributions (Check 4), and the distribution of the data is coherent with the replicated ones (Check 1 and 2).
- As a further measure of discrepancy, we may compute the estimated Bayesian p-value (22) from check 4: in each of the four considered statistics, $p_B \approx 0.5$. Remember that a model is suspect if p_B is close to 0 or 1. If a p-value is close to 0 or 1, it is not so important exactly how extreme it is!

Pest control example

Remind the simple Poisson regression for the cockroaches:

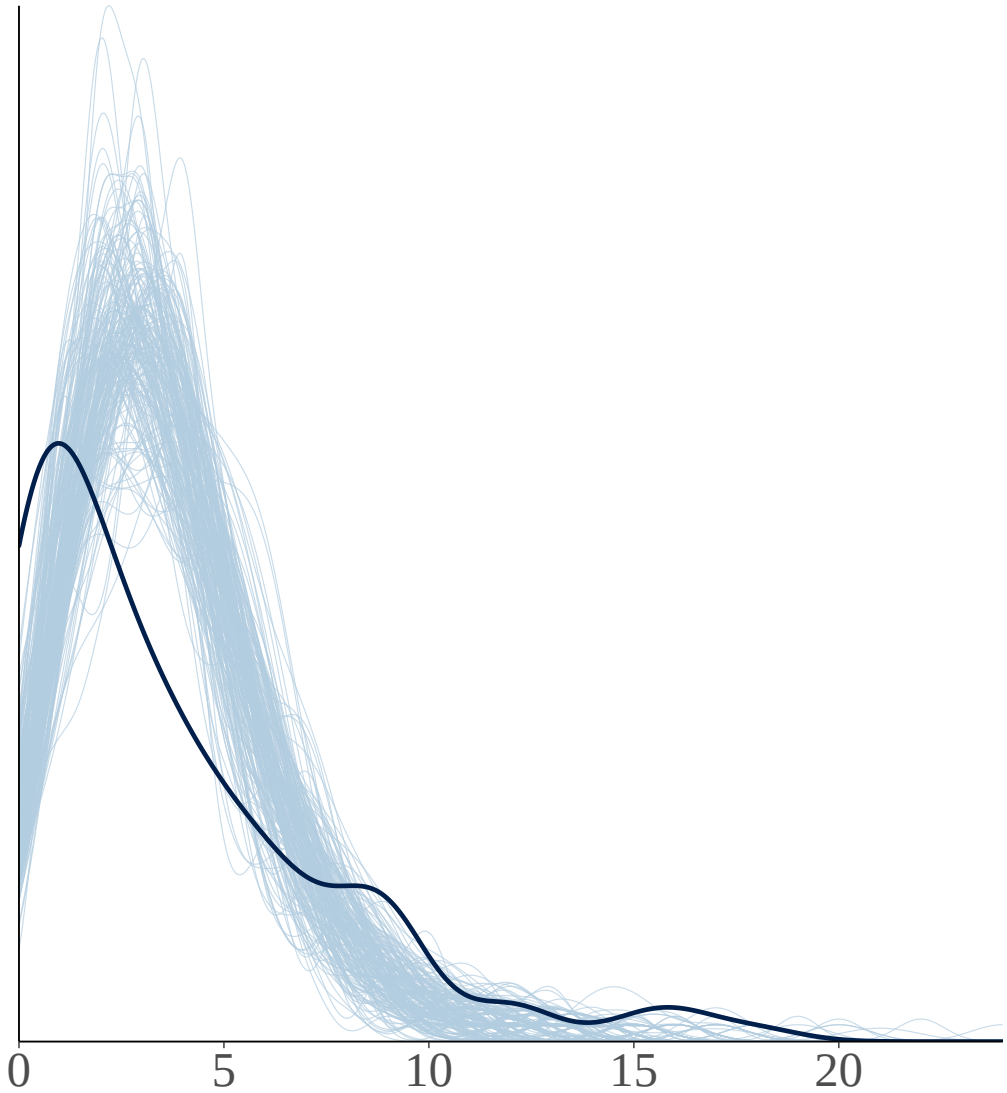
$$\begin{aligned} \text{complaints}_i &\sim \text{Poisson}(\lambda_i) \\ \lambda_i &= \exp(\eta_i) \\ \eta_i &= \alpha + \beta \, \text{traps}_i \end{aligned}$$

We fit the model in Stan and we obtain the following posterior estimates (R output):

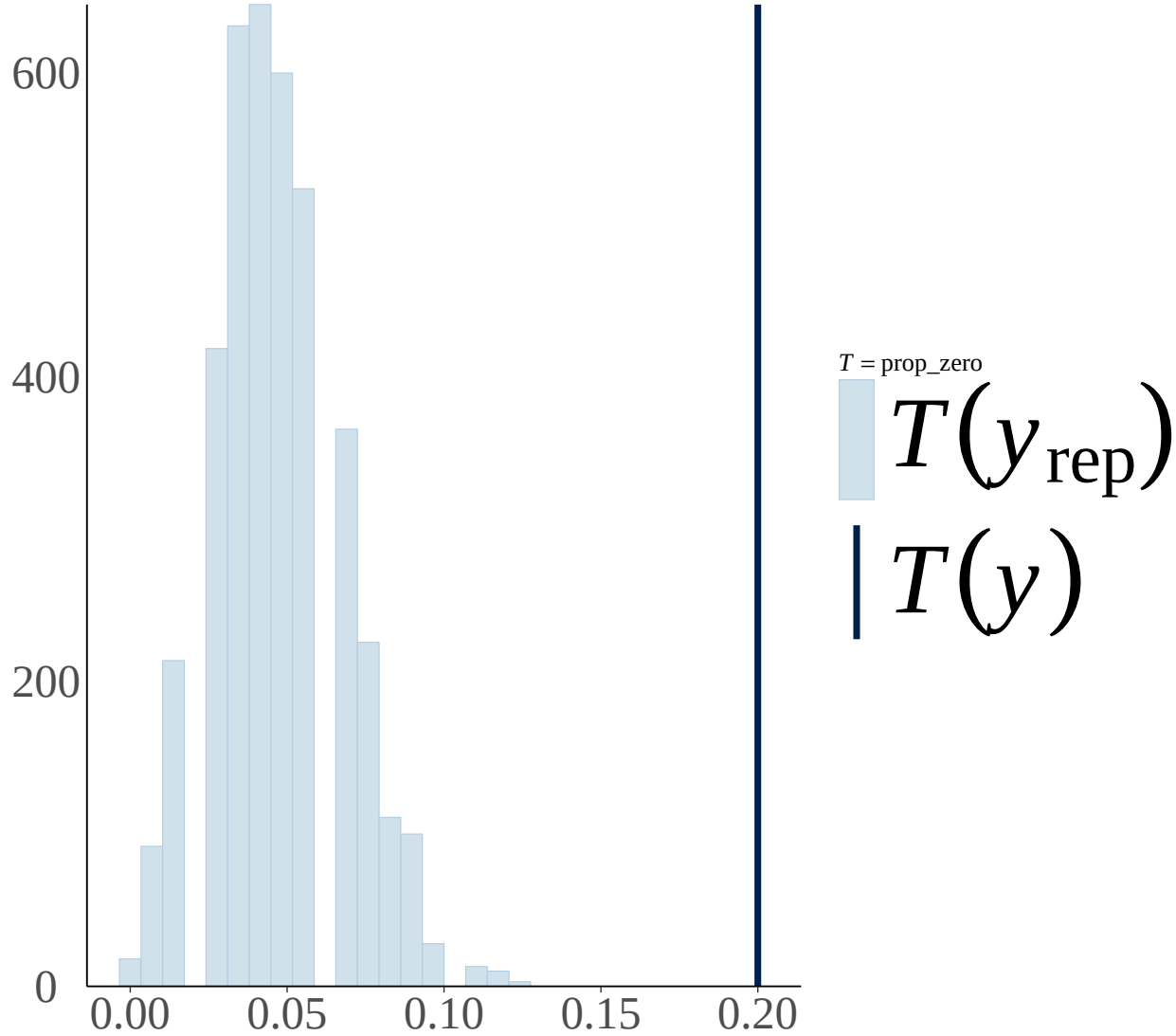
	mean	sd	2.5%	25%	50%	75%	97.5%	n_eff	Rhat
alpha	2.58	0.15	2.28	2.48	2.58	2.69	2.88	979	1
beta	-0.19	0.02	-0.24	-0.21	-0.19	-0.18	-0.15	997	1

Pest control: pp check. Densities

We check the model via some simulated data:



Pest control: pp check. Proportion of zeros

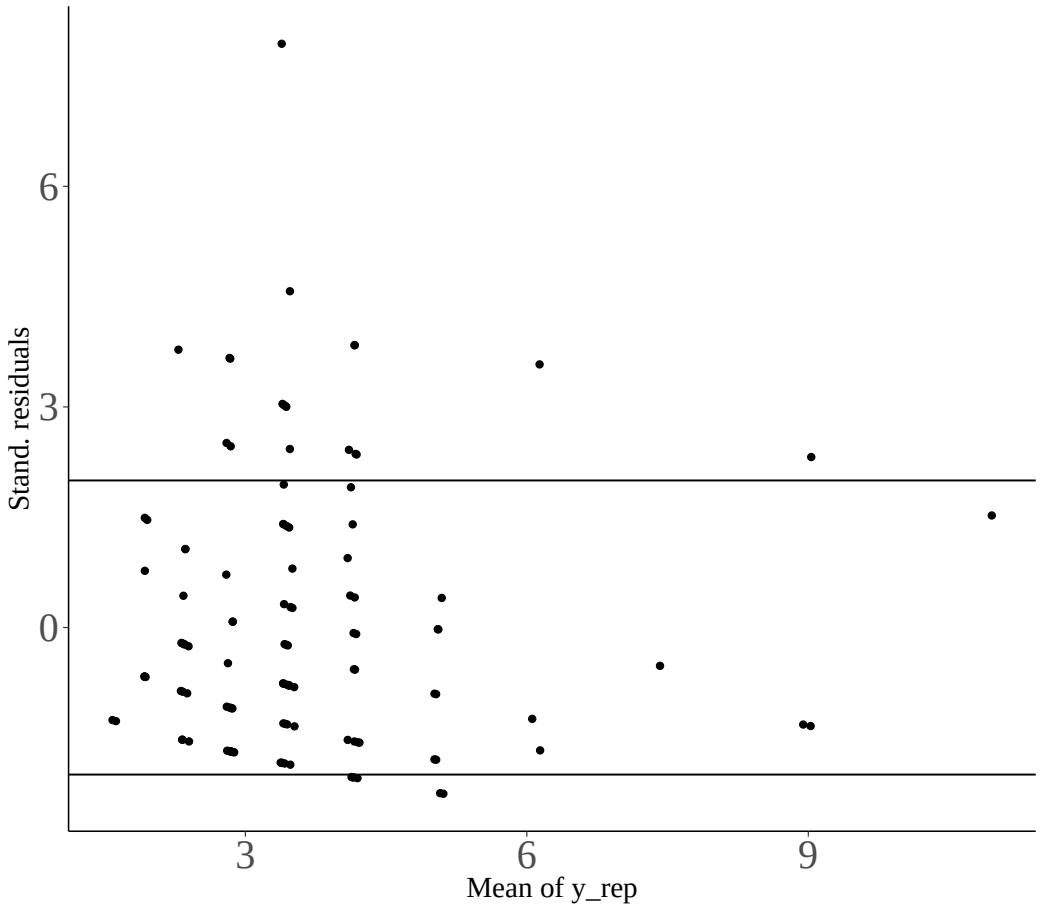


Pest control: pp check.

Comments:

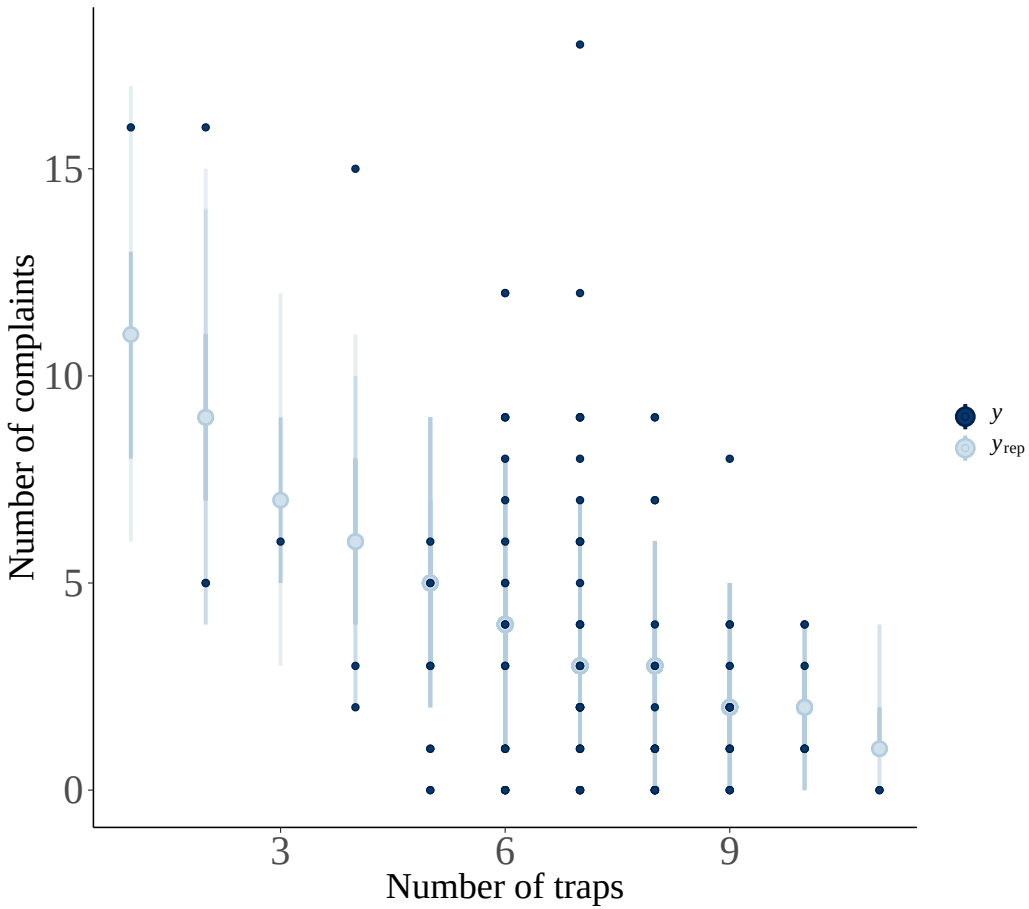
- We immediately realize that replicated distributions are far from the observed data distribution, and that the proportion of zero assumed by the Poisson model is quite underestimated...It is clear that the model does not capture this feature of the data well at all.
- Maybe the Poisson distribution distribution is not suited in this case...let's still explore the standardised residuals of the observed vs predicted number of complaints.
- We can also view how the predicted number of complaints varies with the number of traps.

Pest control: pp check. Residuals



It looks as though we have more positive residuals than negative $\Rightarrow$ the model tends to underestimate the number of complaints.

Pest control: pp check. Predictive intervals



We can see that the model does not seem to fully capture the data.

Strategies when a pp check fails

What to do if a pp check fails? There is not a unique answer. However, some tips may be the following ones:

- extend the model: augment the predictors, include eventual hierarchies
- change the sampling distribution
- change the priors
- transform your data, for instance using logarithm scale.

In what follows, we do not include further predictors, but we will work on the choice of the sampling distribution and, finally, we will consider further hierarchies.

Pest example. Negative binomial model

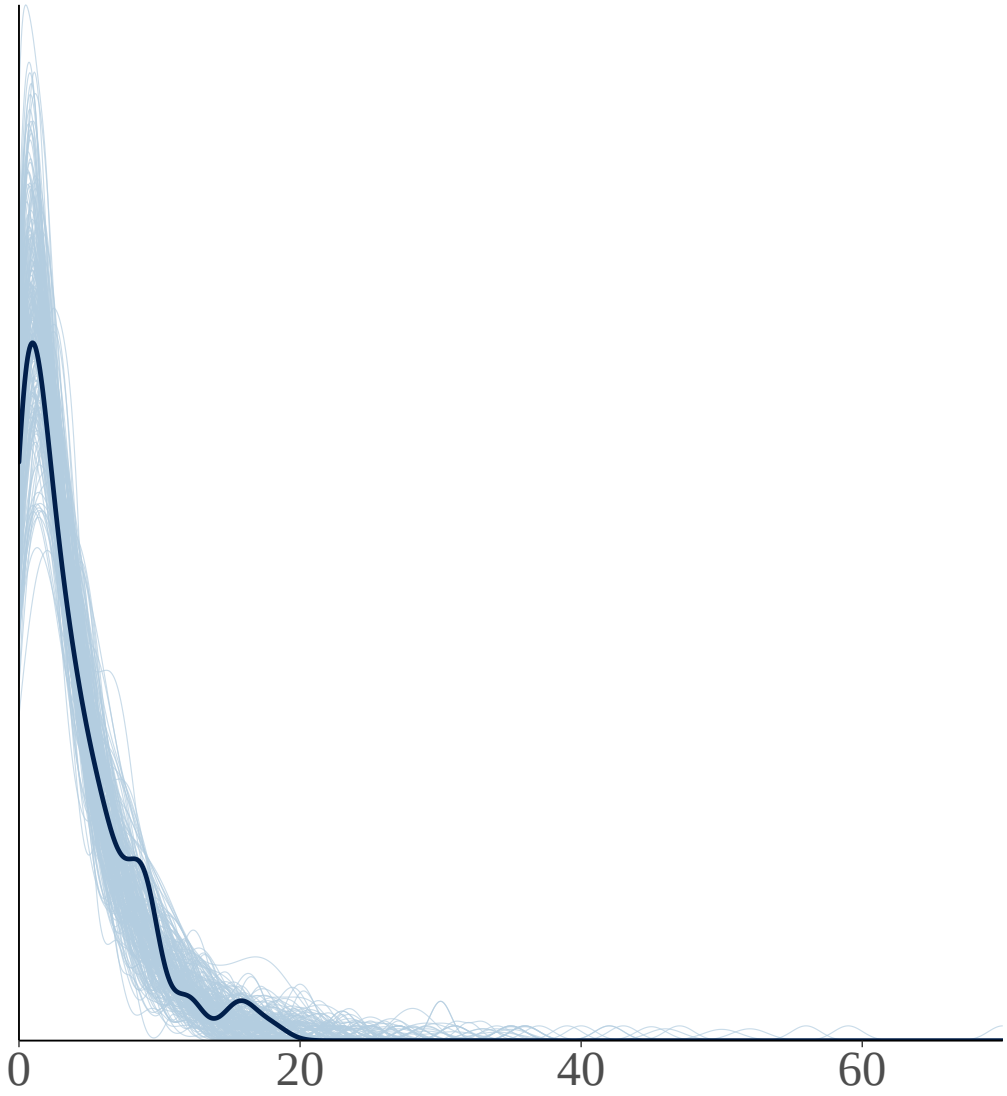
As already seen, negative binomial distribution may capture the *overdispersion* in the data with the parameter ϕ:

$$\begin{aligned} \text{complaints}_i &\sim \text{Neg-Binomial}(\lambda_i, \phi) \\ \lambda_i &= \exp(\eta_i) \\ \eta_i &= \alpha + \beta \, \text{traps}_i \end{aligned}$$

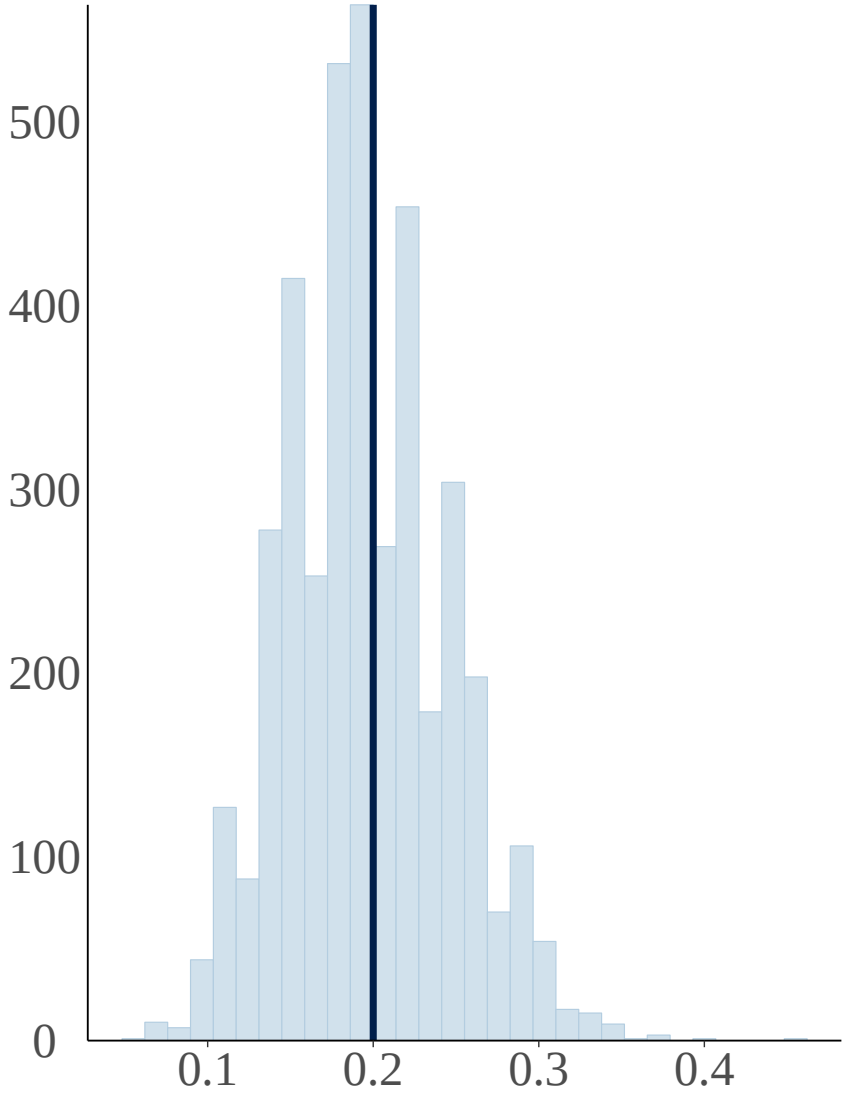
We fit also the negative-binomial model in Stan:

	mean	sd	2.5%	25%	50%	75%	97.5%	n_eff	Rhat
alpha	2.49	0.34	1.81	2.26	2.49	2.73	3.16	1177	1
beta	-0.18	0.05	-0.27	-0.21	-0.18	-0.15	-0.09	1167	1

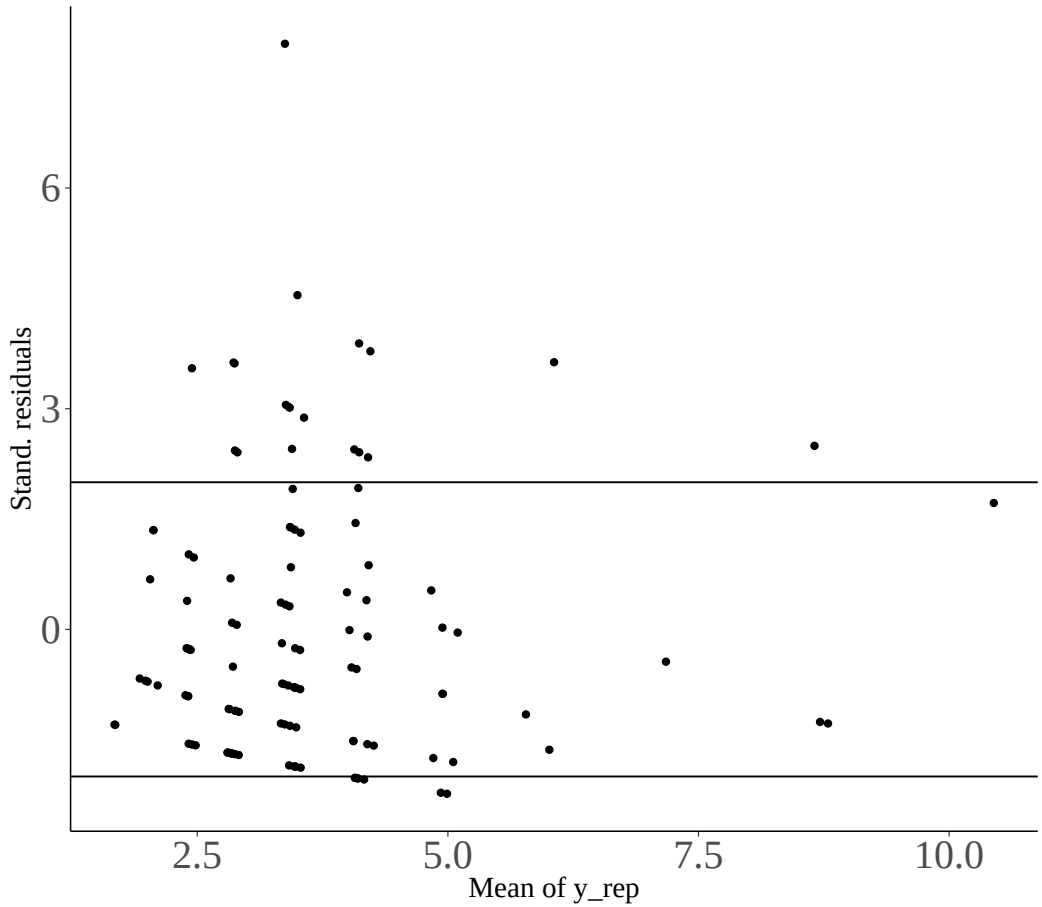
Pest control, NB model. PP check: densities



Pest control, NB model: pp check. Proportion of zeros



Pest control, NB model: pp check. Residuals



It looks as though we have more positive residuals than negative $\Rightarrow$ the model tends to underestimate the number of complaints.

Comments for pp check, NB model

- It appears that our model now captures both the number of small counts better as well as the tails. The negative binomial model does a better job in capturing the number of zeros.
- However, we still have some very large standardized residuals. This might be because we are currently ignoring that the data are clustered by buildings, and that the probability of roach issue may vary substantially across buildings.

Pest control: Hierarchical modelling

Let's add a hierarchical intercept parameter, α_b at the building level to our model. Thus, for the i-th complaint in the b-th building we have:

$$\begin{aligned}
\text{complaints}_{ib} &\sim \text{Neg-Binomial}(\lambda_{ib}, \phi) \\
\lambda_{ib} &= \exp\left(\eta_{ib}\right) \\
\eta_{ib} &= \alpha_{b(i)} + \beta\,\text{traps}_i + \beta_{\text{super}}\,\text{super}_i + \text{log_sq_foot}_i \\
\alpha_b &\sim \mathcal{N}(\mu, \sigma_\alpha^2)
\end{aligned}$$

One of our predictors varies only by building, so we can rewrite the above model more efficiently like so:

$$\begin{aligned}
\eta_{ib} &= \alpha_{b(i)} + \beta\,\text{traps}_i + \text{log_sq_foot}_i \\
\alpha_b &\sim \mathcal{N}(\mu + \beta_{\text{super}}\,\text{super}_i, \sigma_\alpha^2)
\end{aligned}$$

Pest control: Hierarchical modelling

We have more information at the building level as well, like the average age of the residents, the average age of the buildings, and the average per-apartment monthly rent so we can add that data into a matrix called `building_data`, which will have one row per building and four columns:

- `live_in_super`: An indicator for whether the building has a live-in super
- `age_of_building`: The age of the building
- `average_tenant_age`: The average age of the tenants per building
- `monthly_average_rent`: The average monthly rent per building

We'll write the Stan model like:

$$\begin{aligned} \eta_{ib} &= \alpha_{b(i)} + \beta \, \text{traps}_i + \text{log\_sq\_foot}_i \\ \alpha_b &\sim \mathcal{N}(\mu + \texttt{building\_data}\,\zeta, \sigma_\alpha^2) \end{aligned} \tag{25}$$

Model fit in Stan

We fit the model in Stan, at the end we obtain these warnings:

```
Warning messages:
1: There were 915 divergent transitions after warmup.
Increasing adapt_delta above 0.8 may help.
```

What happened? We get a bunch of warnings from Stan about divergent transitions, which is an indication that there may be regions of the posterior that have not been explored by the Markov chains. We will return to this issue later...

In this example we will see that we have divergent transitions because we need to reparametrize our model - i.e., we will retain the overall structure of the model, but transform some of the parameters so that it is easier for Stan to sample from the parameter space.

Model fit in Stan

	mean	sd	2.5%	25%	50%	75%	97.5%	n_eff	Rhat
sigma_alpha	0.25	0.17	0.05	0.13	0.22	0.34	0.69	182	1.03
beta	-0.23	0.06	-0.35	-0.27	-0.22	-0.19	-0.11	715	1.00
mu	1.25	0.42	0.43	0.98	1.22	1.53	2.12	849	1.00
phi	1.54	0.36	0.99	1.29	1.49	1.75	2.38	302	1.01
alpha[1]	1.28	0.54	0.21	0.95	1.24	1.62	2.37	1007	1.00
alpha[2]	1.23	0.52	0.21	0.91	1.20	1.56	2.31	914	1.00
alpha[3]	1.39	0.49	0.51	1.05	1.38	1.71	2.41	397	1.01
alpha[4]	1.43	0.48	0.53	1.09	1.39	1.75	2.42	561	1.00
alpha[5]	1.07	0.42	0.25	0.76	1.08	1.33	1.94	880	1.01
alpha[6]	1.16	0.48	0.22	0.86	1.16	1.45	2.16	914	1.00
alpha[7]	1.43	0.52	0.49	1.07	1.39	1.77	2.51	434	1.01
alpha[8]	1.27	0.42	0.45	1.00	1.29	1.52	2.12	1156	1.00
alpha[9]	1.40	0.55	0.29	1.05	1.41	1.74	2.51	1077	1.00
alpha[10]	0.86	0.37	0.17	0.60	0.85	1.11	1.62	644	1.01

Model fit in Stan

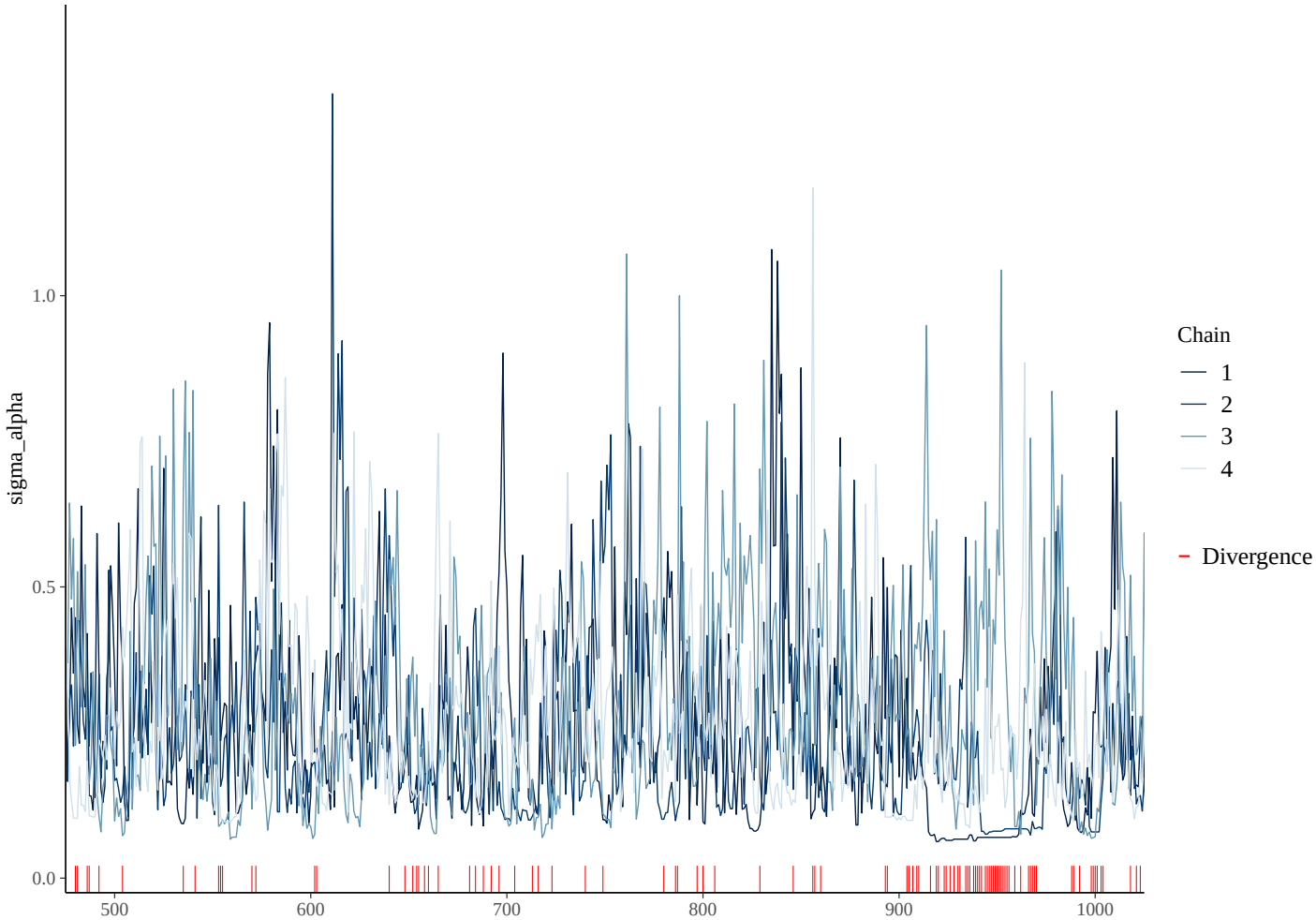
Before we go through exactly how to do this reparameterization, we will first go through what indicates that this is something that reparameterization will resolve. We will go through:

1. Examining the fitted parameter values, including the effective sample size
2. Traceplots and scatterplots that reveal particular patterns in locations of the divergences.

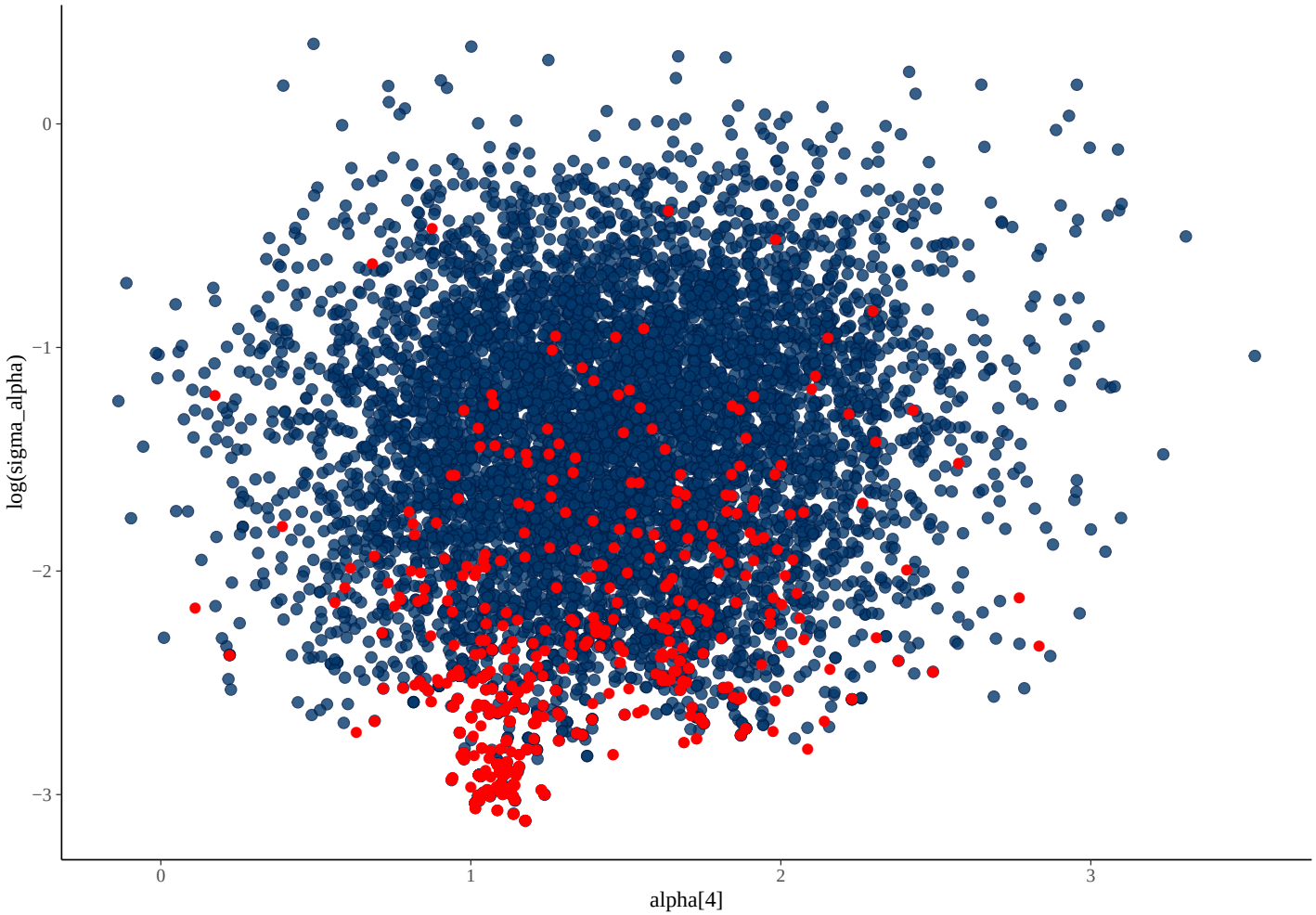
The effective samples are quite low for many of the parameters relative to the total number of samples. This alone isn't indicative of the need to reparameterize, but it indicates that we should look further at the trace plots and pairs plots.

Model fit in Stan

First let's look at the traceplots to see if the divergent transitions form a pattern.

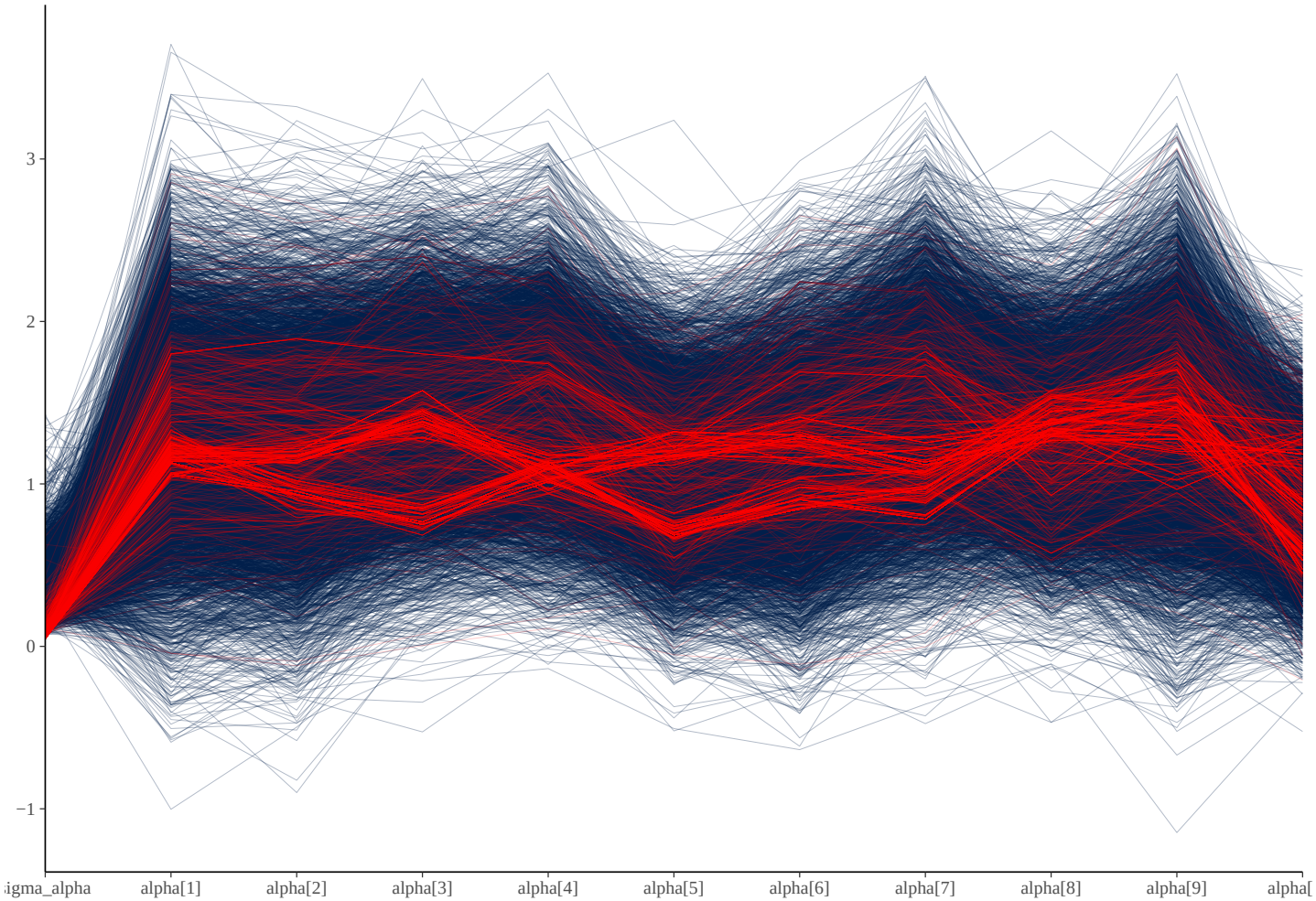


Model fit in Stan



Model fit in Stan

Another way to look at the divergences is via a parallel coordinates plot:



Model fit in Stan

Comments:

- Looks as if the divergent parameters, the little red bars underneath the traceplots correspond to samples where the sampler gets stuck at one parameter value for σ_α.
- What we have in the scatterplot, is a cloud-like shape, with most of the divergences clustering towards the bottom. We'll see a bit later that we actually want this to look more like a funnel than a cloud, but the divergences are indicating that the sampler can't explore the narrowing neck of the funnel.
- From the parallel plot, again, we see evidence that our problems concentrate when σ_α is small.

Model fit in Stan: non-centered parametrization

CENTERED

$$\eta_{ib} = \alpha_{b(i)} + \beta\, x_i$$
$$\alpha_b \sim \mathcal{N}(\mu + \zeta\, z_b, \sigma_\alpha^2)$$

NON-CENTERED

$$\eta_{ib} = \alpha_{b(i)} + \beta\, x_i$$
$$\alpha_b = \mu + \zeta\, z_b + \sigma_\alpha \tilde{\alpha}_b$$
$$\tilde{\alpha}_b \sim \mathcal{N}(0, 1)$$

We should use the non-centered parameterization for α_b. We define a vector of auxiliary variables in the parameters block, `alpha_raw` that is given a $\mathcal{N}(0, 1)$ prior in the model block. We then make `alpha` a transformed parameter. We can reparameterize the random intercept α_b, which is distributed:

$$\alpha_b \sim \mathcal{N}(\mu + \texttt{building\_data}\,\zeta, \sigma_\alpha^2)$$

Model fit in Stan: non-centered parametrization

In the `transformed parameters` block we define now:

```
transformed parameters {
  vector[J] alpha;
  alpha = mu + building_data * zeta + sigma_alpha * alpha_raw;
}
```

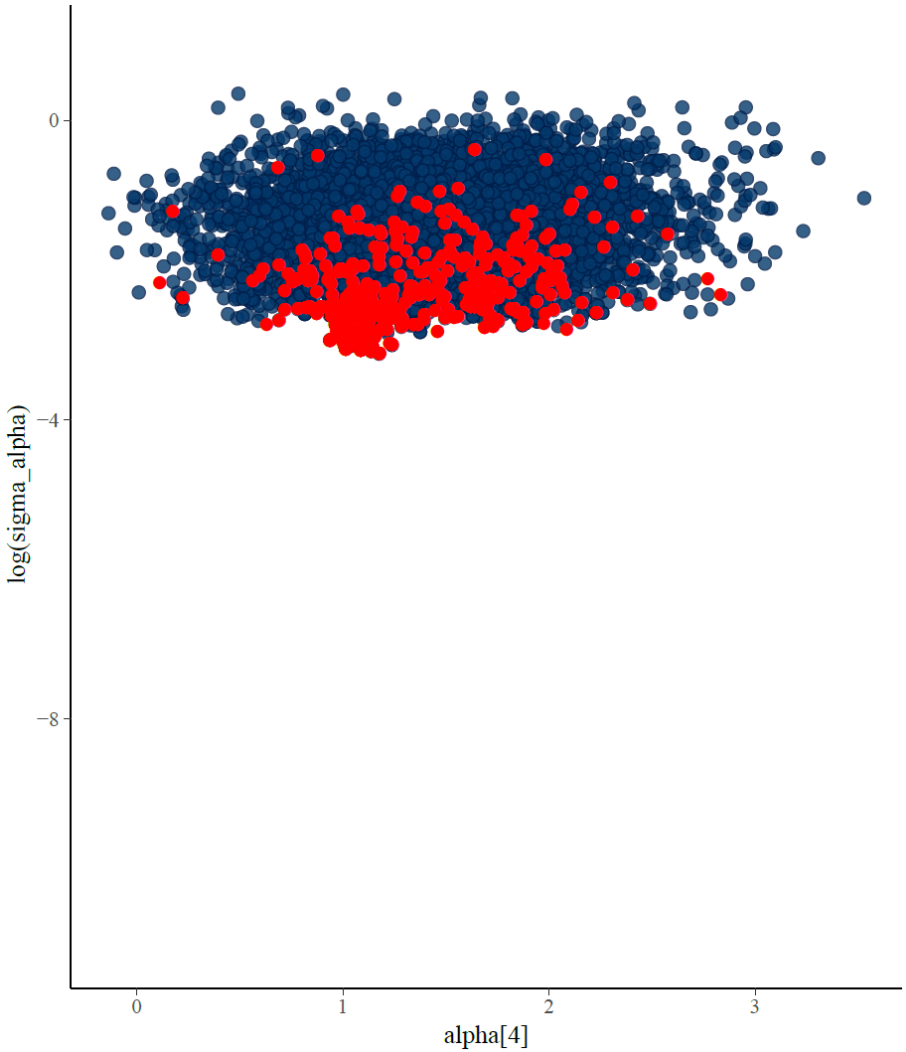
This gives `alpha` a $\mathcal{N}(\mu + \texttt{building\_data}\,\zeta, \sigma_\alpha^2)$ distribution, but it decouples the dependence of the density of each element of `alpha` from `sigma_alpha` (σ_α).

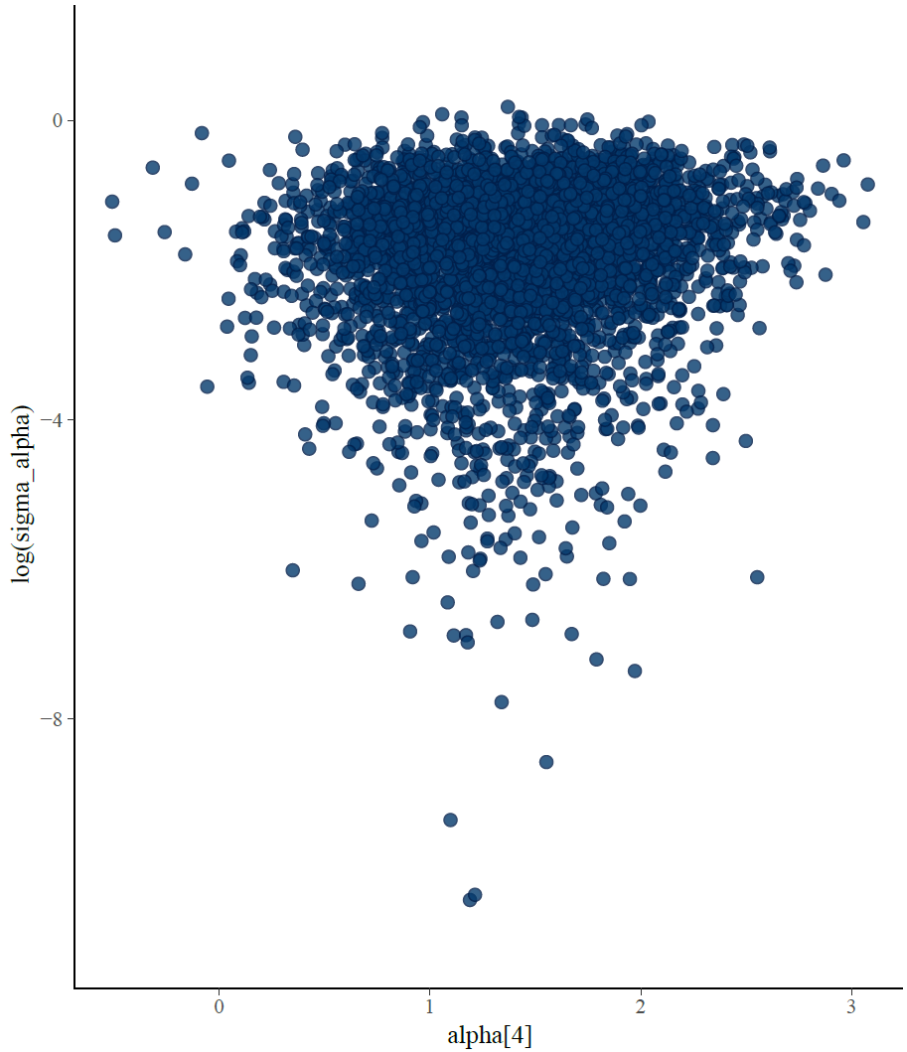
We fit this new model version in Stan. We will examine the effective sample size of the fitted model to see whether we've fixed the problem with our reparameterization.

Model fit in Stan: non-centered parametrization

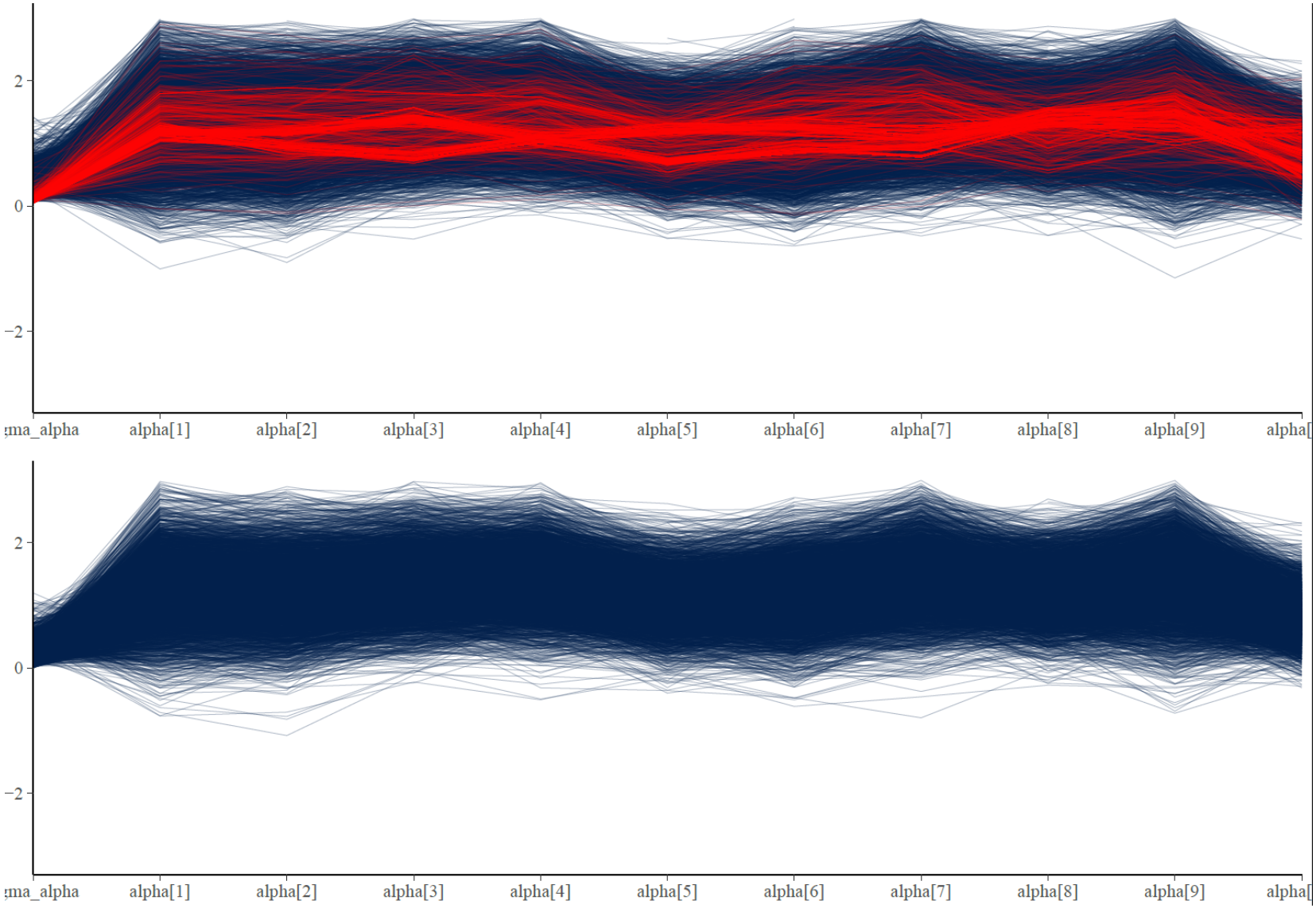
	mean	sd	2.5%	25%	50%	75%	97.5%	n_eff	Rhat
sigma_alpha	0.23	0.17	0.01	0.10	0.20	0.32	0.63	1447	1
beta	-0.23	0.06	-0.35	-0.27	-0.23	-0.19	-0.11	2649	1
mu	1.25	0.44	0.40	0.95	1.24	1.54	2.12	2555	1
phi	1.58	0.34	1.03	1.34	1.54	1.77	2.35	4256	1
alpha[1]	1.27	0.56	0.15	0.90	1.27	1.64	2.37	2566	1
alpha[2]	1.21	0.53	0.19	0.86	1.21	1.56	2.28	2551	1
alpha[3]	1.38	0.49	0.42	1.05	1.38	1.71	2.38	2672	1
alpha[4]	1.42	0.49	0.46	1.08	1.42	1.74	2.39	2783	1
alpha[5]	1.08	0.42	0.26	0.81	1.07	1.34	1.92	3162	1
alpha[6]	1.17	0.49	0.22	0.85	1.17	1.49	2.12	2502	1
alpha[7]	1.45	0.52	0.42	1.10	1.44	1.79	2.49	2996	1
alpha[8]	1.23	0.43	0.40	0.94	1.23	1.52	2.10	3481	1
alpha[9]	1.41	0.58	0.25	1.03	1.42	1.80	2.51	2780	1
alpha[10]	0.86	0.37	0.17	0.61	0.85	1.11	1.60	3417	1

Model fit in Stan: centered vs non-centered parametrization





Model fit in Stan: centered vs non-centered parametrization

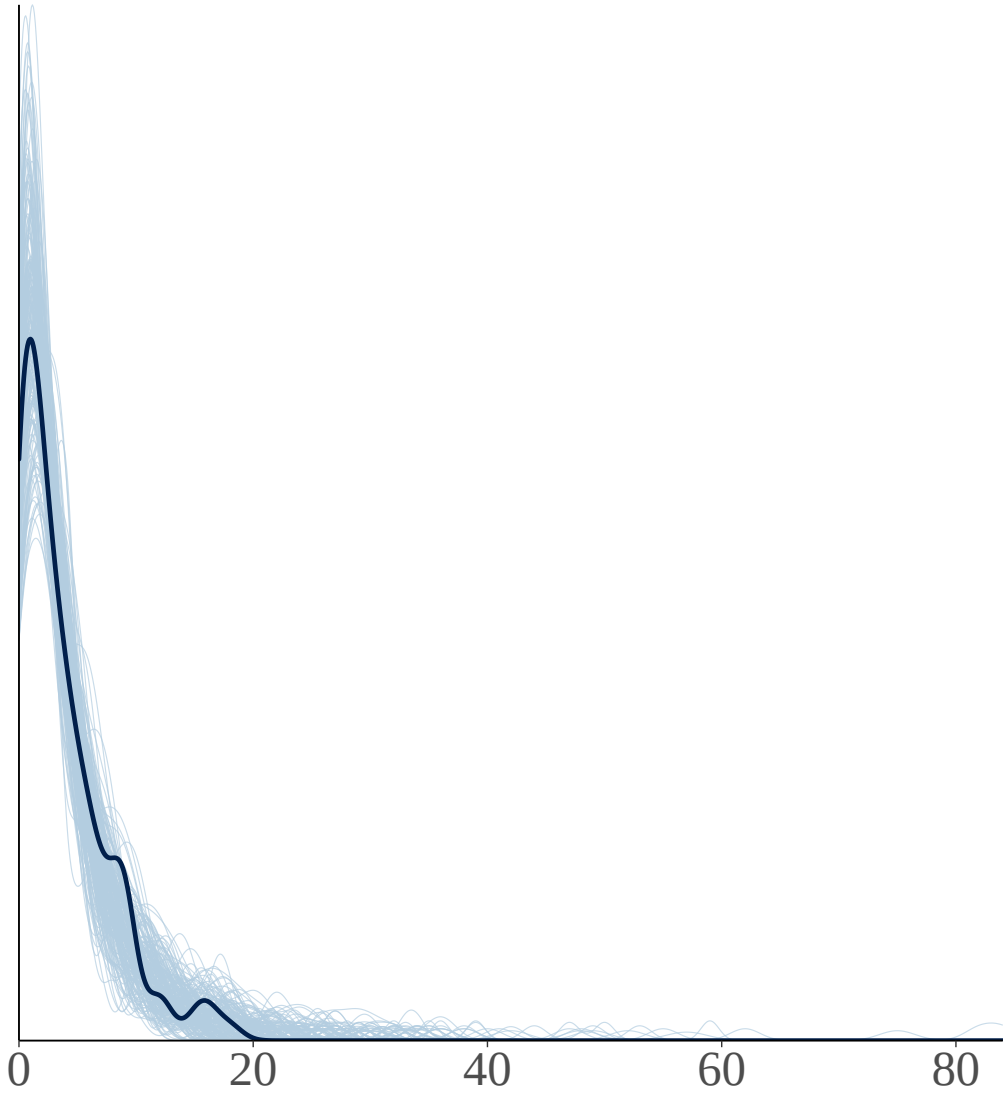


Model fit in Stan: centered vs non-centered parametrization

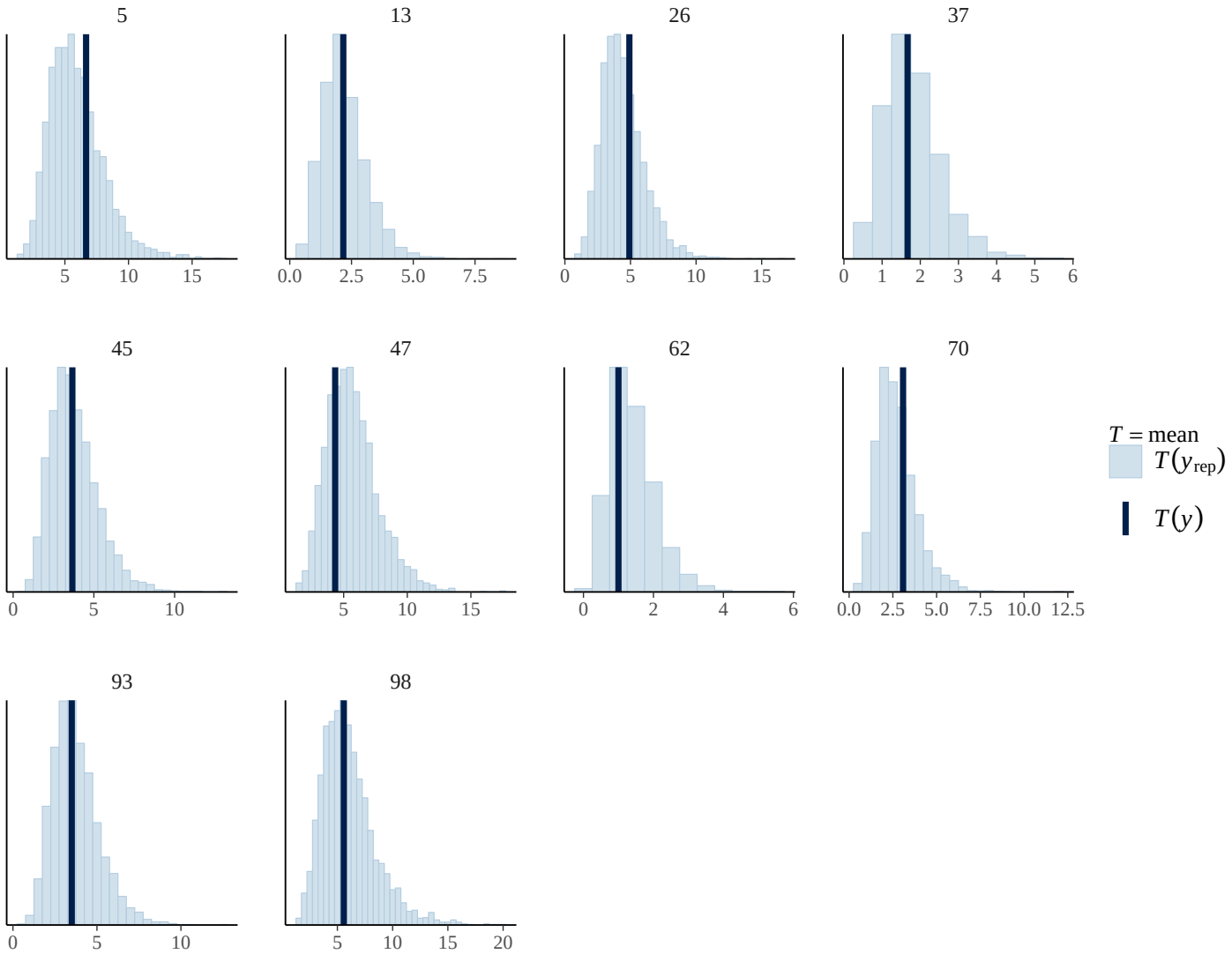
Comments:

- This has improved the effective sample sizes of α.
- No more divergent transitions!

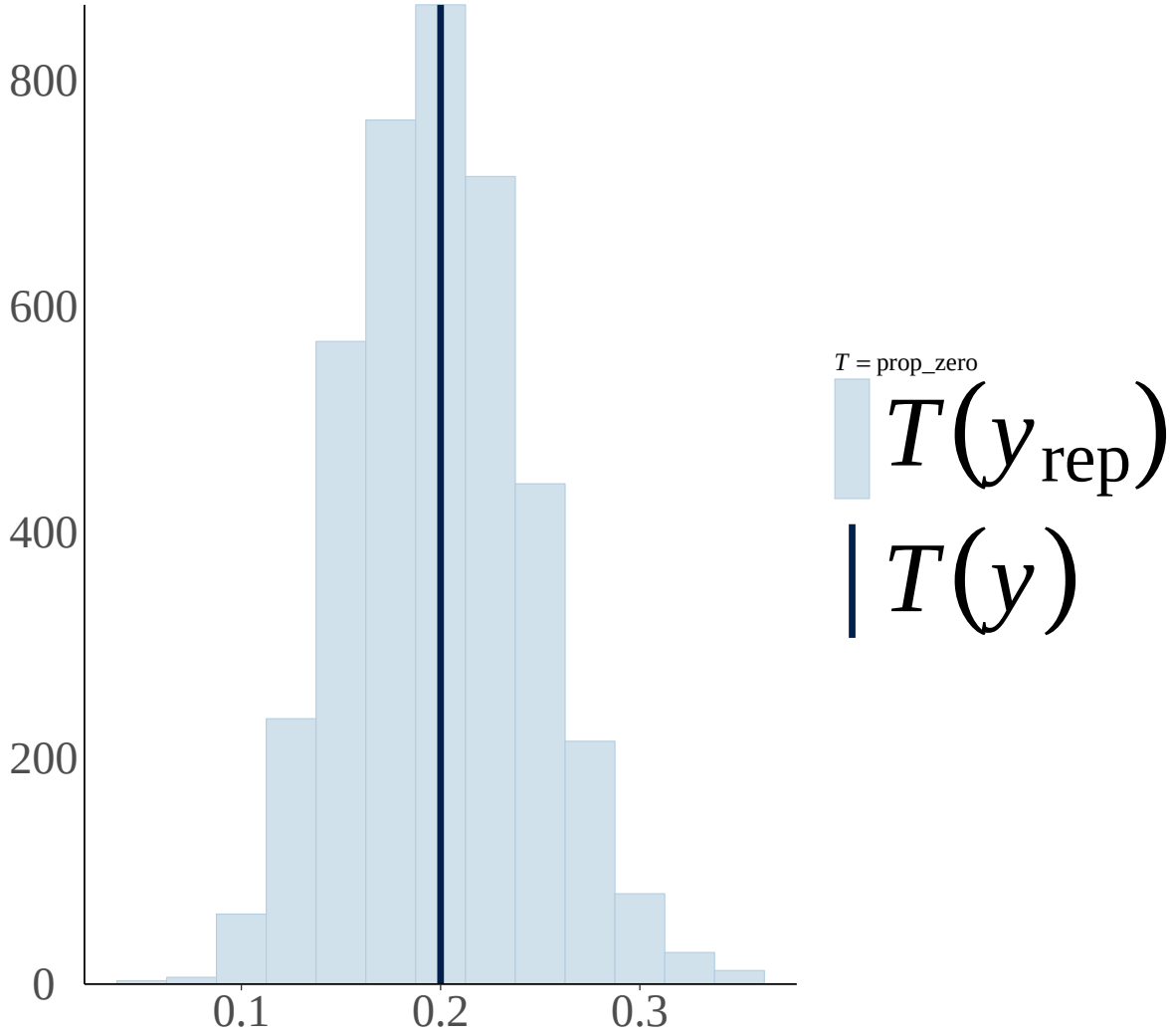
Pest model, hierarchical NB ncp model. PP check: densities



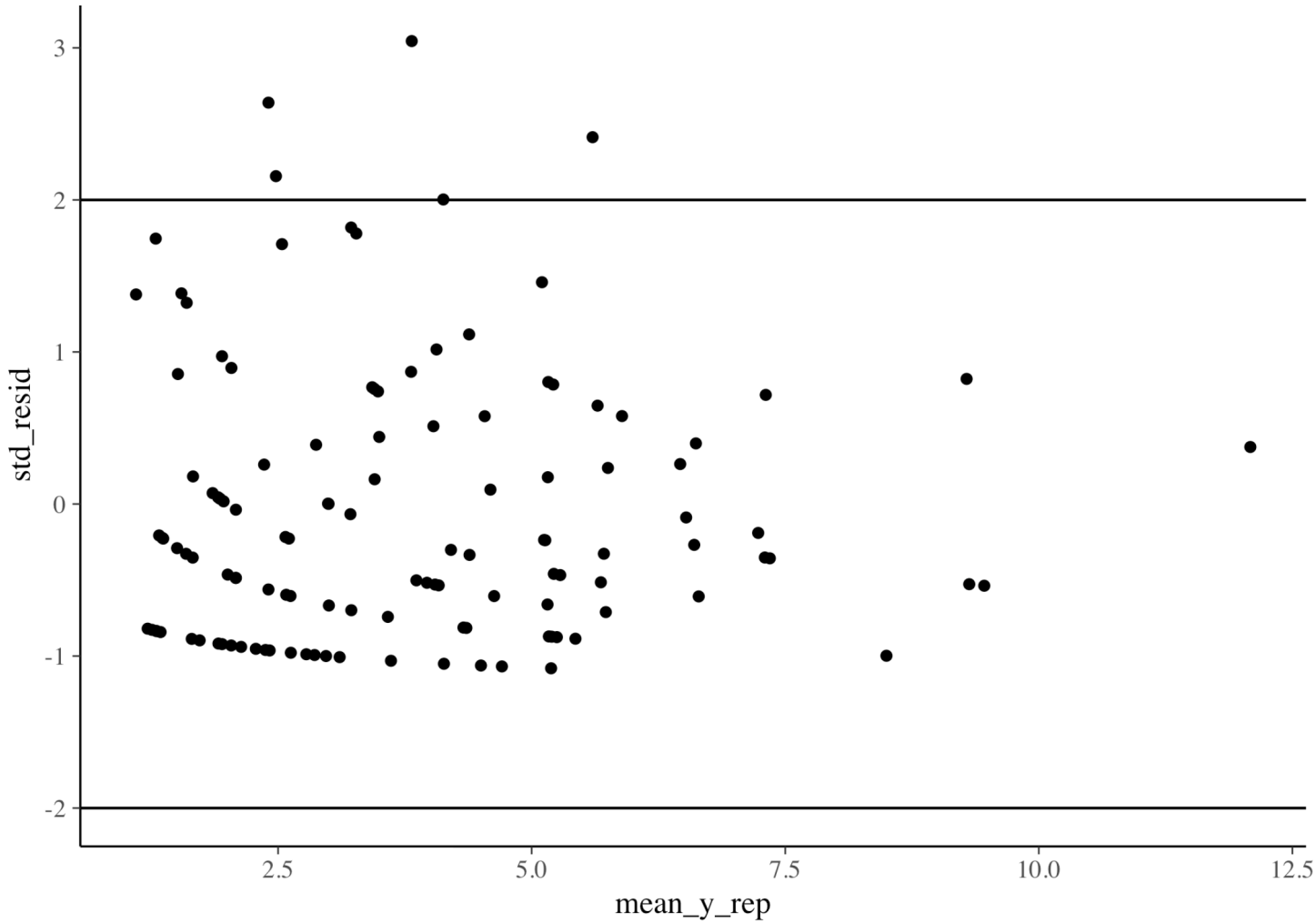
Pest model, hierarchical NB ncp model. PP check: statistics



Pest model, hier NB ncp model. PP check: prop. of zeros



Pest control, hier NB ncp model: pp check. Residuals



Better!

Further readings

Further reading:

- Chapter 6 from *BDA*, A. Gelman et al. (model checking)
- Chapter 20 from the Stan Users Guide (reparametrization)