SOME MORE MODELLING

Dan Simpson

PROBIT REGRESSION AND BOUNDED VARIBLES

- Probit regression is cool!
- Instead of using the logistic link function, you use the inverse CDF of a Gaussian. eeeeek
- $y_i \sim \text{binomial}(1,p_i)$
- $p_i = \Phi(X\beta)$
- Here $\Phi(\cdot)$ is the CDF of a standard Gaussian.
- Let's think about how to do this in Stan!

PROBIT 2 WAYS

- Direct
- Latent variable
- (Multivariate extension https://mc-stan.org/docs/2_18/stan-users-guide/multivariate-outcomes.html)

LAST WEEK WE LOOKED AT HIERARCHICAL MODELS

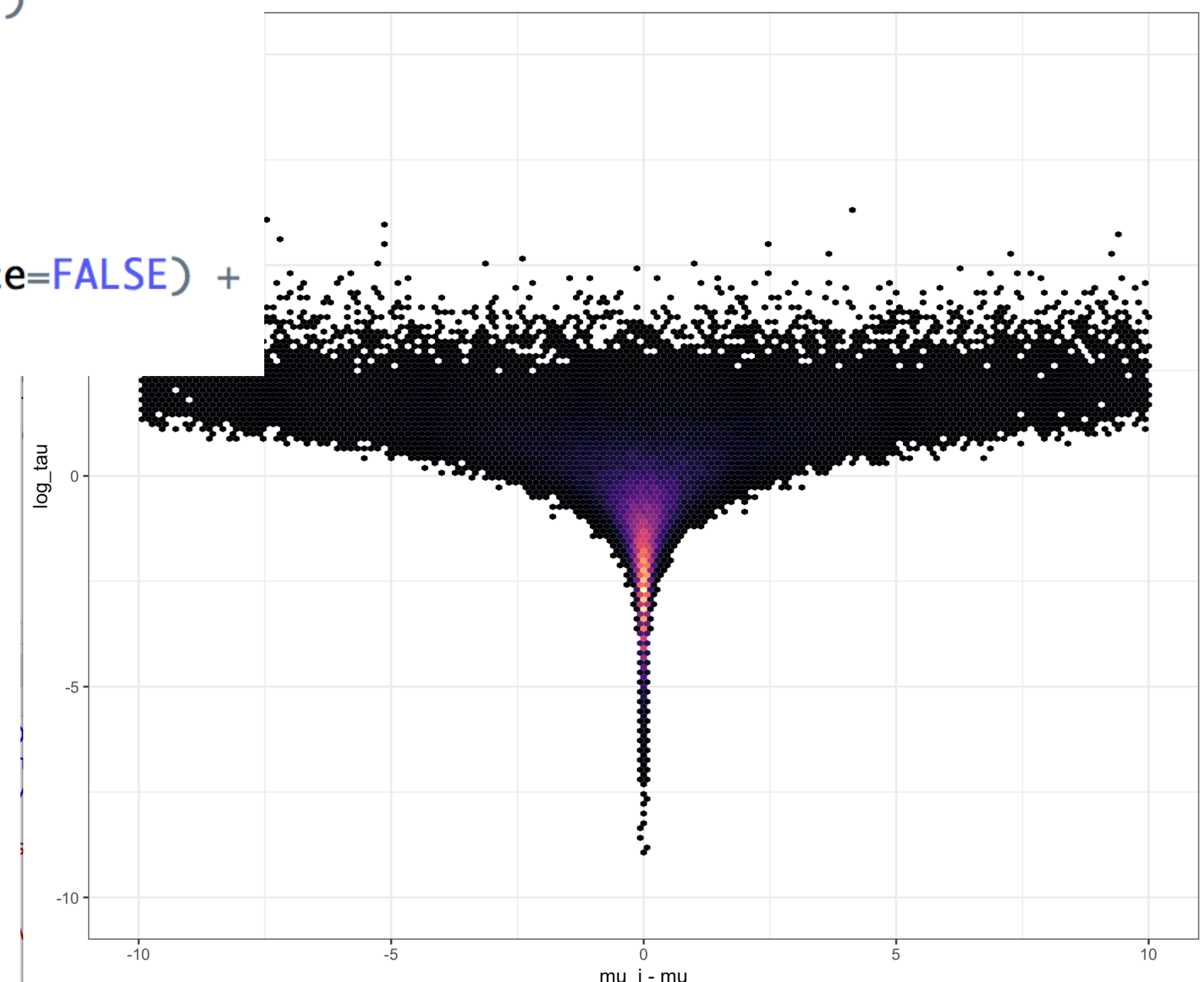
- Simplest example

$$
\begin{aligned}
&y_{ij} \mid \mu_j, \sigma \sim N(\mu_j, \sigma^2) \\
&\mu_j \mid \mu, \tau \sim N(\mu, \tau^2) \\
&\tau, \mu \sim N_+(\tau; 0,3) \cdot N(\mu; 0,3)
\end{aligned}
$$

- We saw that there were some problems fitting this.
- It has to do with the joint prior $p(\mu_j, \mu, \tau)$
- (NB: It's ok to just look at one μ_j because *a priori* they are exchangeable)

ALWAYS SIMULATE!

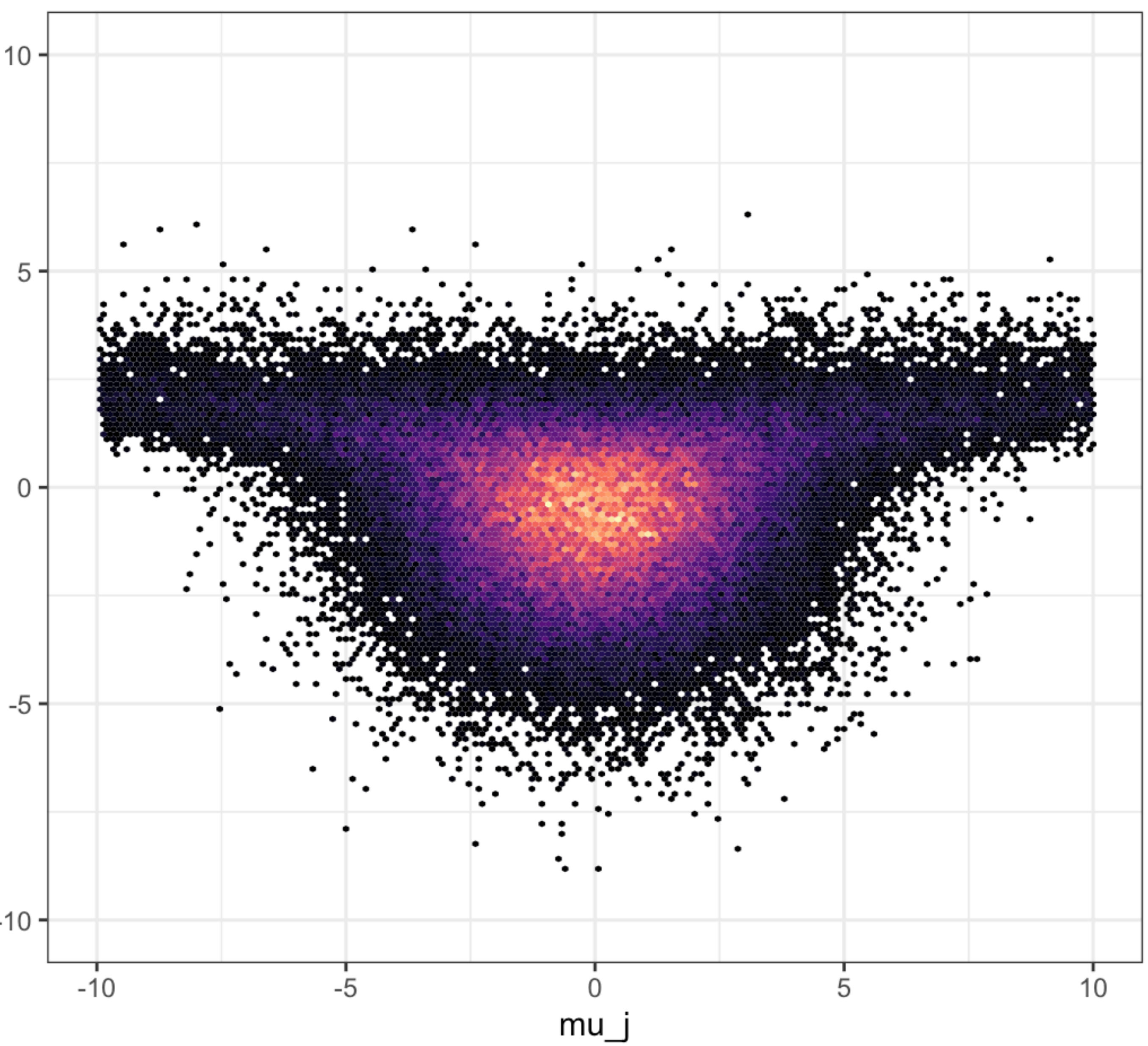
```r
library(tidyverse)
library(viridis)
N = 100000
m = 10
dat = tibble(log_tau = rnorm(N,0,2),
             mu = rnorm(N,0,2),
             mu_j = rnorm(N,mu,exp(log_tau)) )

dat %>% ggplot(aes(x=mu_j-mu,y=log_tau)) +
  ylim(-10,10) + xlim(-10,10) +
  geom_hex(bins = 150) +
  scale_fill_viridis(option = "magma",discrete=FALSE) +
  theme_bw()
```

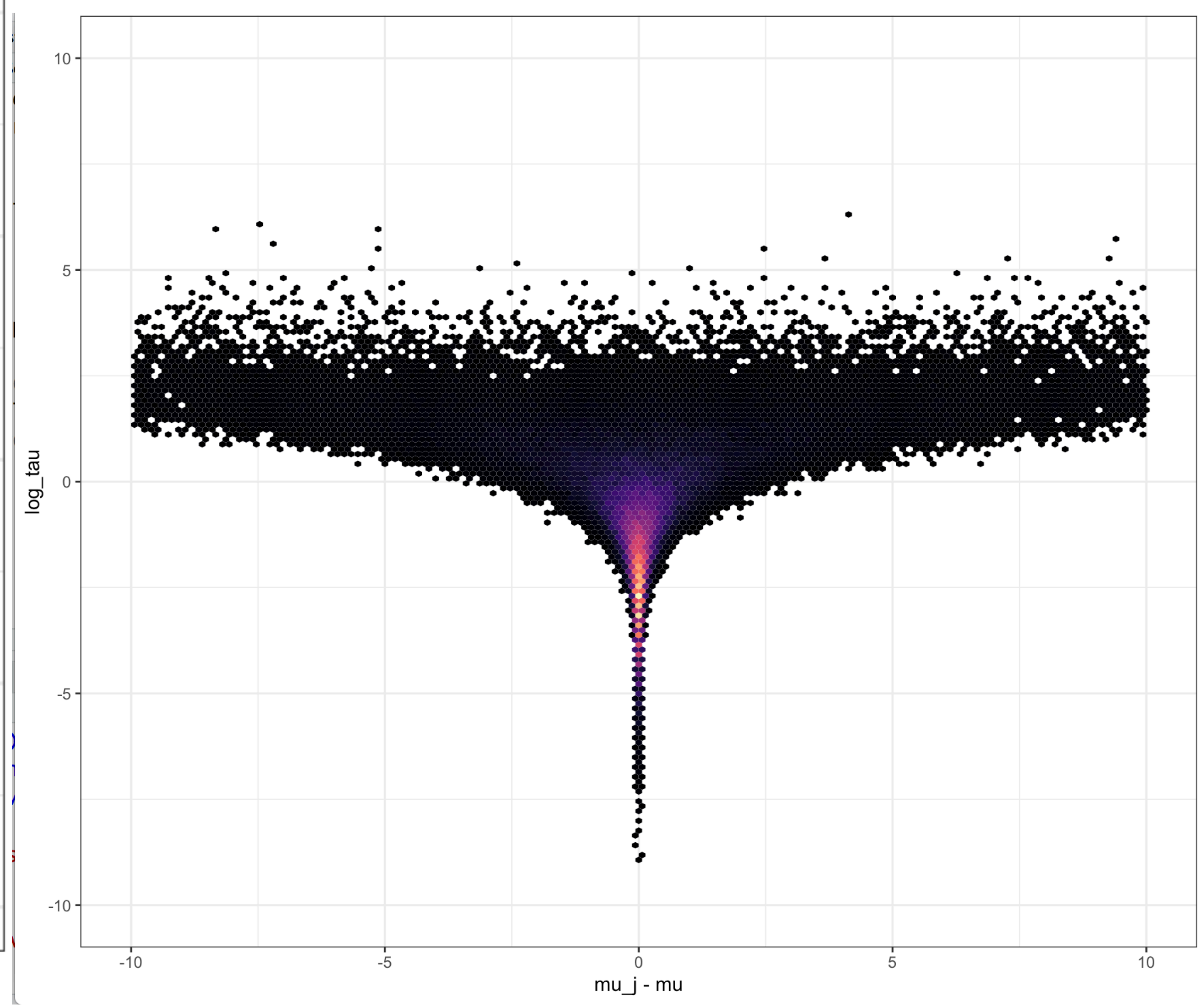


NOTE THE X-AXIS!

μ_j vs τ

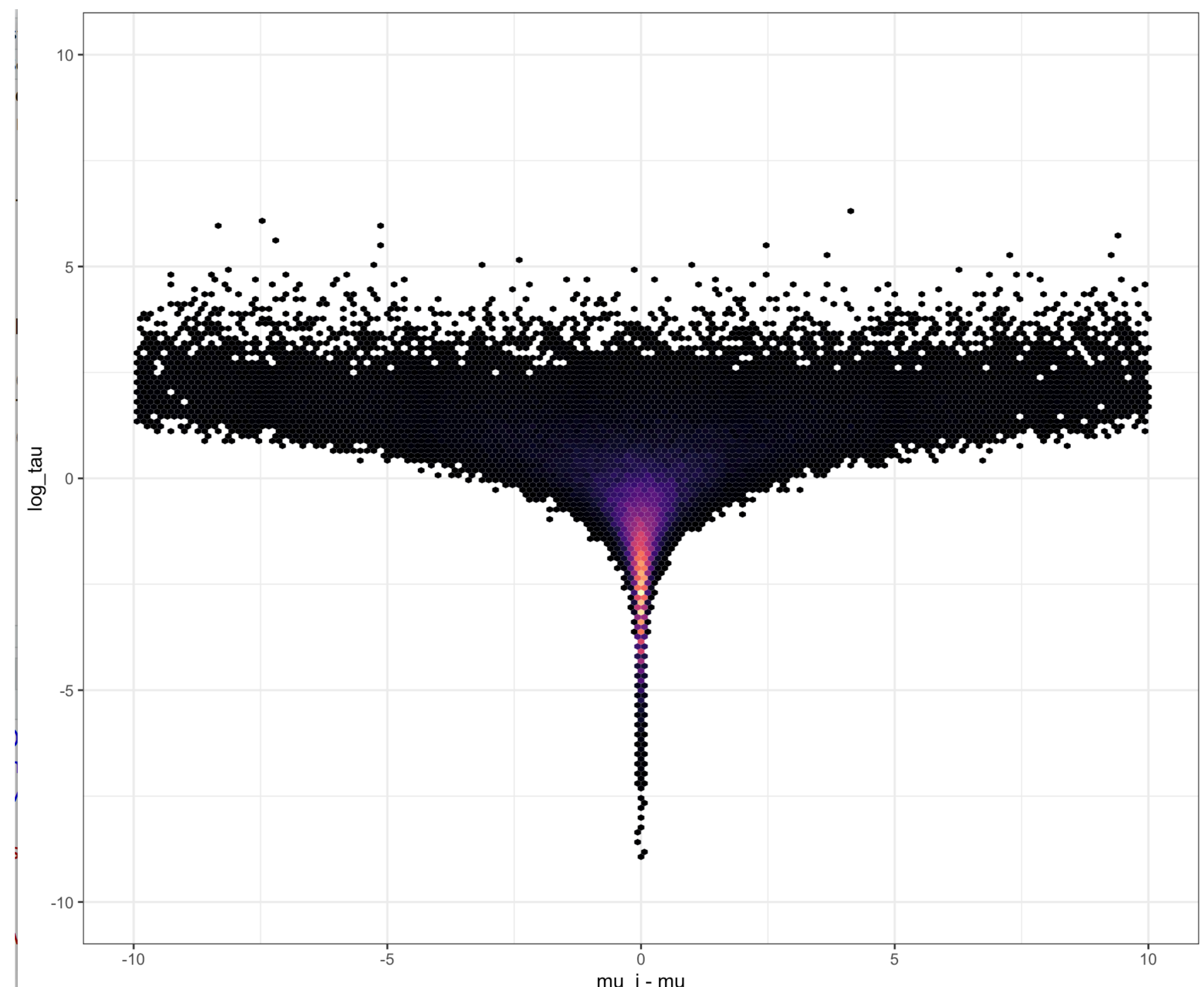
$(\mu_j - \mu)$ vs τ





THAT CUSP BEHAVIOUR CAN BE BAD

- It's not that the pinch happens
- It's that there's a lot of prior mass in the funnel
- So it might be important!
- So we fix it with reparameterization



A NEW CUT. A NEW COLOR

- Fix it with a new parameterization

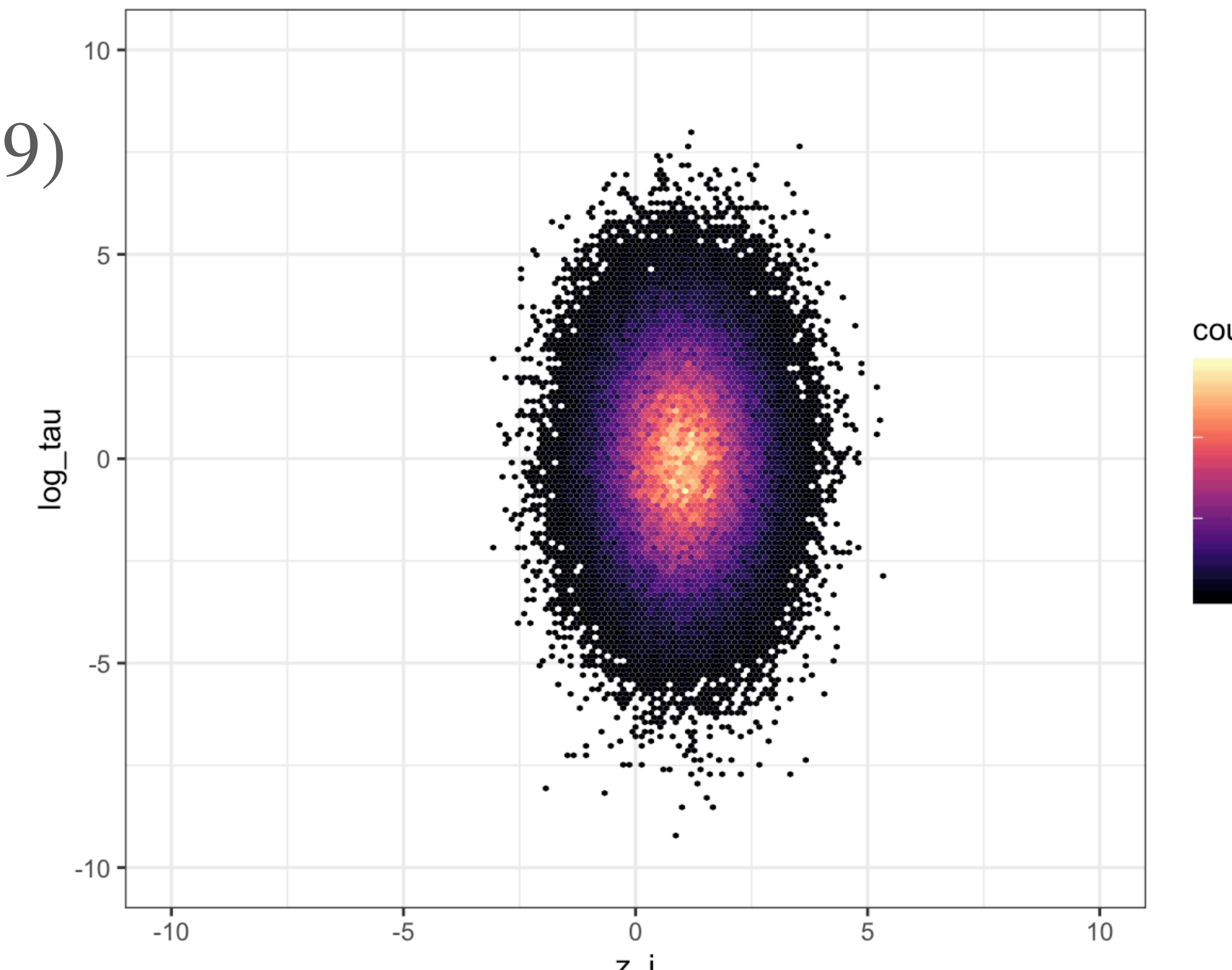
  $y_{ij} \mid \mu_j, \sigma \sim N(\mu_j, \sigma^2)$

  $\mu_j = \mu + \tau z_j$

  $z_j \sim N(0,1)$

  $\tau, \mu \sim N_+(\tau; 0,9) \cdot N(\mu; 0,9)$

- Goodbye strong prior dependence



BUT MAYBE IT MAKES THE POSTERIOR WORSE

```
sigma2_j = 0.000001
dat2 = tibble( log_tau = rnorm(N,0,800*sqrt(sigma2_j)),
         mu_j = rnorm(N,0,
                  sqrt(1/(1+exp(log_tau*2)/sigma2_j))),
         z_j = mu_j*exp(-log_tau) )
dat2 %>% ggplot(aes(x=mu_j,y=log_tau)) +
  geom_hex(bins = 150) +
  scale_fill_viridis(option = "magma",discrete=FALSE) +
  theme_bw()
```