

Pathologies in hierarchical models

FW 891

[Click here to view presentation online](#)


Christopher Cahill

16 October 2023



Quantitative Fisheries Center
MICHIGAN STATE UNIVERSITY

Purpose

- Introduce some background and theoretical concepts
- The Devil's funnel (spooky seaz'n )
- What to do about it
- Example
- Example #2 (Eight schools)

Background

- Many of the most exciting problems in applied statistical ecology involve intricate, high-dimensional models, and sparse data (at least relative to model complexity)

Background

- Many of the most exciting problems in applied statistical ecology involve intricate, high-dimensional models, and sparse data (at least relative to model complexity)
- In situations where the data alone cannot identify a model, significant prior information is required to draw valid inference

Background

- Many of the most exciting problems in applied statistical ecology involve intricate, high-dimensional models, and sparse data (at least relative to model complexity)
- In situations where the data alone cannot identify a model, significant prior information is required to draw valid inference
- Such prior information is not limited to an explicit prior distribution, but instead can be encoded in the model construction itself 🤖

A one level hierarchical model

$$\pi(\theta, \phi \mid \mathcal{D}) \propto \prod_{i=1}^n \pi(\mathcal{D}_i \mid \theta_i) \pi(\theta_i \mid \phi) \pi(\phi)$$

A one level hierarchical model

$$\pi(\theta, \phi \mid \mathcal{D}) \propto \prod_{i=1}^n \pi(\mathcal{D}_i \mid \theta_i) \pi(\theta_i \mid \phi) \pi(\phi)$$

- Hierarchical models are defined by the organization of a model's parameters into exchangeable groups, and the resulting conditional independencies between those groups

A one level hierarchical model

$$\pi(\theta, \phi \mid \mathcal{D}) \propto \prod_{i=1}^n \pi(\mathcal{D}_i \mid \theta_i) \pi(\theta_i \mid \phi) \pi(\phi)$$

- Hierarchical models are defined by the organization of a model's parameters into exchangeable groups, and the resulting conditional independencies between those groups
- Can also visualize this as a directed acyclic graph (DAG)

Hierarchical DAG

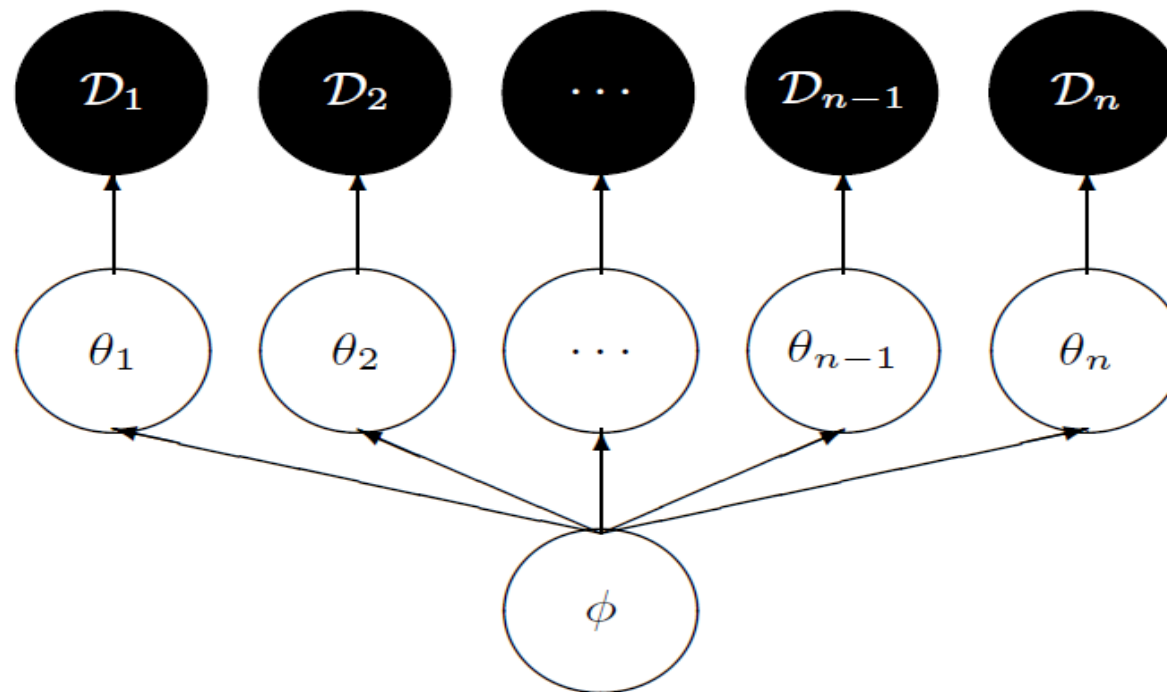


FIG. 1. In hierarchical models “local” parameters, θ , interact via a common dependency on “global” parameters, ϕ . The interactions allow the measured data, \mathcal{D} , to inform all of the θ instead of just their immediate parent. More general constructions repeat this structure, either over different sets of parameters or additional layers of hierarchy.

A one-level hierarchical model

$$y_i \sim N(\theta_i, \sigma_i^2)$$

$$\theta_i \sim N(\mu, \tau^2), \text{ for } i = 1, \dots, I$$

A one-level hierarchical model

$$y_i \sim N(\theta_i, \sigma_i^2)$$

$$\theta_i \sim N(\mu, \tau^2), \text{ for } i = 1, \dots, I$$

- In terms of the previous equations,
 $\mathcal{D} = (y_i, \sigma_i)$, $\phi = (\mu, \tau)$, and $\theta = (\theta_i)$

A one-level hierarchical model

$$y_i \sim N(\theta_i, \sigma_i^2)$$

$$\theta_i \sim N(\mu, \tau^2), \text{ for } i = 1, \dots, I$$

- In terms of the previous equations,
 $\mathcal{D} = (y_i, \sigma_i)$, $\phi = (\mu, \tau)$, and $\theta = (\theta_i)$
- Call any elements of ϕ *global* parameters

A one-level hierarchical model

$$y_i \sim N(\theta_i, \sigma_i^2)$$

$$\theta_i \sim N(\mu, \tau^2), \text{ for } i = 1, \dots, I$$

- In terms of the previous equations,
 $\mathcal{D} = (y_i, \sigma_i)$, $\phi = (\mu, \tau)$, and $\theta = (\theta_i)$
- Call any elements of ϕ *global* parameters
- Call any elements of θ *local* parameters

A one-level hierarchical model

$$y_i \sim N(\theta_i, \sigma_i^2)$$

$$\theta_i \sim N(\mu, \tau^2), \text{ for } i = 1, \dots, I$$

- In terms of the previous equations,
 $\mathcal{D} = (y_i, \sigma_i)$, $\phi = (\mu, \tau)$, and $\theta = (\theta_i)$
- Call any elements of ϕ *global* parameters
- Call any elements of θ *local* parameters
- However, recognize this nomenclature breaks down in situations with more levels

A key pathology

- Unfortunately, this one-level model exhibits some of the typical pathologies of hierarchical models
- Small changes in ϕ induce large changes in density
- When data are sparse, the density of these models looks like a “funnel”
 - Region of high density but low volume, and a region of low density but high volume
- However, the probability mass of these two regions is the same (or nearly so)
- Any algorithm must be able to manage the dramatic variations in curvature to fully map out the posterior

Naive model implementations

- Assuming a normal model with no data, a latent mean μ set at zero, and a lognormal prior on the variance

$$\tau^2 = e^{v^2}$$

Naive model implementations

- Assuming a normal model with no data, a latent mean μ set at zero, and a lognormal prior on the variance

$$\tau^2 = e^{v^2}$$

$$\pi(\theta_1, \dots, \theta_n, v) \propto \prod_{i=1}^n N\left(x_i \mid 0, \left(e^{-v/2}\right)^2\right) N(v \mid 0, 3^2)$$

Naive model implementations

- Assuming a normal model with no data, a latent mean μ set at zero, and a lognormal prior on the variance

$$\tau^2 = e^{v^2}$$

$$\pi(\theta_1, \dots, \theta_n, v) \propto \prod_{i=1}^n N\left(x_i \mid 0, \left(e^{-v/2}\right)^2\right) N(v \mid 0, 3^2)$$

- This hierarchical structure induces large correlations between v and each θ_i

Visualizing the pathology 🤩

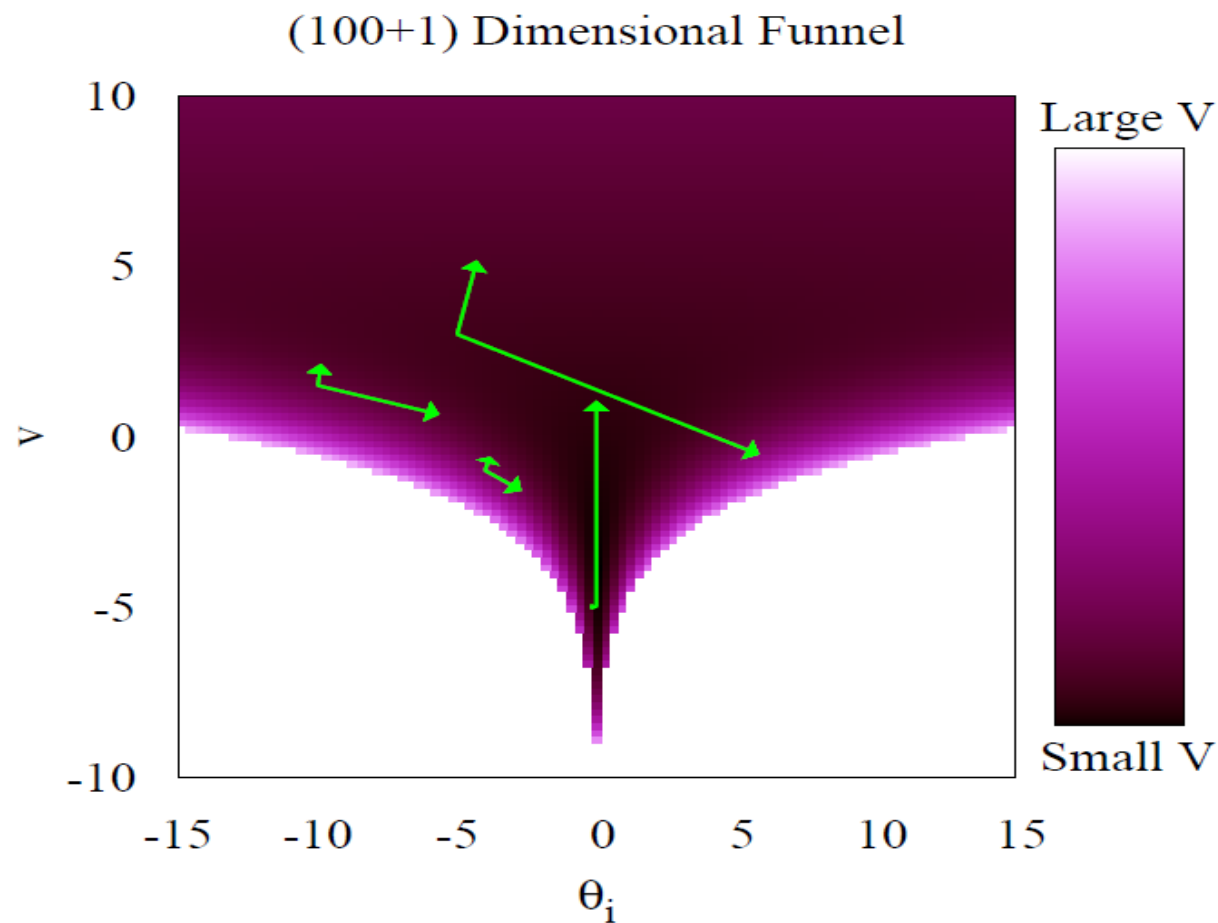


FIG. 2. Typical of hierarchical models, the curvature of the funnel distribution varies strongly with the parameters, taxing most algorithms and limiting their ultimate performance.

Some things worth noting

- There is position dependence in the correlation structure, i.e., correlation changes depending on where you are located in the posterior

Some things worth noting

- There is position dependence in the correlation structure, i.e., correlation changes depending on where you are located in the posterior
- No global correction, like rotating or rescaling will solve this problem!

Some things worth noting

- There is position dependence in the correlation structure, i.e., correlation changes depending on where you are located in the posterior
- No global correction, like rotating or rescaling will solve this problem!
- Often manifests as a divergent transition in Stan, as HMC cannot accurately explore the posterior

How can we fix this problem?

- Remember that the prior information we include in an analysis is not only limited to the choice of an explicit prior distribution

How can we fix this problem?

- Remember that the prior information we include in an analysis is not only limited to the choice of an explicit prior distribution
- The dependence between layers in our model can actually be broken up by reparameterizing the existing parameters into a so-called “non-centered” parameterization
 - Think about the DAG

How can we fix this problem?

- Remember that the prior information we include in an analysis is not only limited to the choice of an explicit prior distribution
- The dependence between layers in our model can actually be broken up by reparameterizing the existing parameters into a so-called “non-centered” parameterization
 - Think about the DAG
- Non-centered parameterizations factor certain dependencies into deterministic transformations between the layers, leaving the actively sampled variables uncorrelated

Centered vs. non-centered model maths

Centered model:

$$\begin{aligned} y_i &\sim N(\theta_i, \sigma_i^2) \\ \theta_i &\sim N(\mu, \tau^2), \end{aligned}$$

Non-centered analog:

Centered vs. non-centered model maths

Centered model:

$$\begin{aligned}y_i &\sim N(\theta_i, \sigma_i^2) \\ \theta_i &\sim N(\mu, \tau^2),\end{aligned}$$

Non-centered analog:

$$\begin{aligned}y_i &\sim N(\vartheta_i \tau + \mu, \sigma_i^2) \\ \vartheta_i &\sim N(0, 1).\end{aligned}$$

Centered vs. non-centered model maths

Centered model:

$$\begin{aligned}y_i &\sim N(\theta_i, \sigma_i^2) \\ \theta_i &\sim N(\mu, \tau^2),\end{aligned}$$

Non-centered analog:

$$\begin{aligned}y_i &\sim N(\vartheta_i \tau + \mu, \sigma_i^2) \\ \vartheta_i &\sim N(0, 1).\end{aligned}$$

Key point: NCP shifts correlation from the latent parameters to data

Centered vs. non-centered model DAG

4

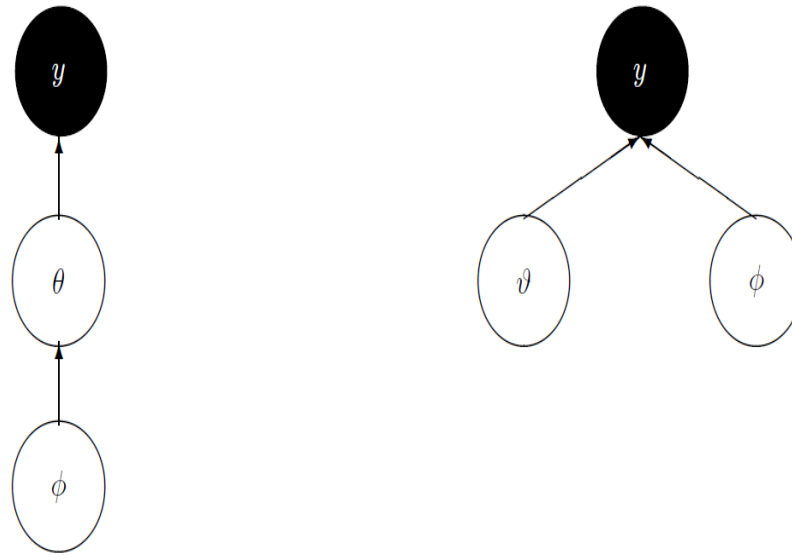


FIG. 4. In one-level hierarchical models with global parameters, ϕ , local parameters, θ , and measured data y , correlations between parameters can be mediated by different parameterizations of the model. Non-centered parameterizations exchange a direct dependence between ϕ and θ for a dependence between ϕ and y ; the reparameterized ϑ and ϕ become independent conditioned on the data. When the data are weak these non-centered parameterizations yield simpler posterior geometries.

When does NCP help?

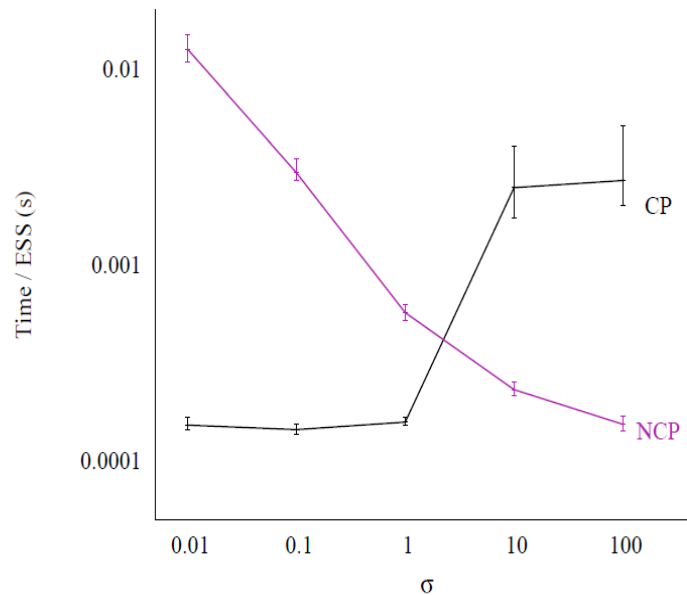


FIG. 8. Depending on the common variance, σ^2 , from which the data were generated, the performance of a 10-dimensional one-way normal model (2) varies drastically between centered (CP) and non-centered (NCP) parameterizations of the latent parameters, θ_i . As the variance increases and the data become effectively more sparse, the non-centered parameterization yields the most efficient inference and the disparity in performance increases with the dimensionality of the model.

Example



Example

- consider the one-way normal model with 800 latent θ_i
- constant measurement error $\sigma_i = \sigma = 10$
- latent parameters are $\mu = 8, \tau = 3$
- θ_i and y_i sampled randomly

Example

- consider the one-way normal model with 800 latent θ_i
- constant measurement error $\sigma_i = \sigma = 10$
- latent parameters are $\mu = 8, \tau = 3$
- θ_i and y_i sampled randomly

Add weakly informative priors to this generative likelihood

$$\pi(\mu) = N(0, 5^2)$$

$$\pi(\tau) = \text{Half-Cauchy}(0, 2.5).$$

Example, centered vs. noncentered

The centered parameterization of this model can be written as

$$\begin{aligned} y_i &\sim N(\theta_i, \sigma_i^2) \\ \theta_i &\sim N(\mu, \tau^2), \text{ for } i = 1, \dots, 800 \end{aligned}$$

Example, centered vs. noncentered

The centered parameterization of this model can be written as

$$\begin{aligned} y_i &\sim N(\theta_i, \sigma_i^2) \\ \theta_i &\sim N(\mu, \tau^2), \text{ for } i = 1, \dots, 800 \end{aligned}$$

and it should have inferior performance relative to the noncentered model:

$$\begin{aligned} y_i &\sim N(\tau\vartheta_i + \mu, \sigma_i^2) \\ \vartheta_i &\sim N(0, 1), \text{ for } i = 1, \dots, 800 \end{aligned}$$

Using Stan to simulate fake data

```
1 transformed data {
2   real mu;
3   real<lower=0> tau;
4   real alpha;
5   int N;
6   mu = 8;
7   tau = 3;
8   alpha = 10;
9   N = 800;
10 }
11 generated quantities {
12   real mu_print;
13   real tau_print;
14   vector[N] theta;
15   vector[N] sigma;
16   vector[N] y;
17   mu_print = mu;
18   tau_print = tau;
19   for (i in 1:N) {
20     theta[i] = normal_rng(mu, tau);
21     sigma[i] = alpha;
22     y[i] = normal_rng(theta[i], sigma[i]);
23   }
24 }
```

Calling that from R

```
1 library("cmdstanr")
2
3 one_level <- cmdstan_model("src/sim_one_level.stan")
4
5 # simulate data
6 sim <- one_level$sample(
7   fixed_param = T, # look here
8   iter_warmup = 0, iter_sampling = 1,
9   chains = 1, seed = 1
10 )
```

Running MCMC with 1 chain...

Chain 1 Iteration: 1 / 1 [100%] (Sampling)

Chain 1 finished in 0.0 seconds.

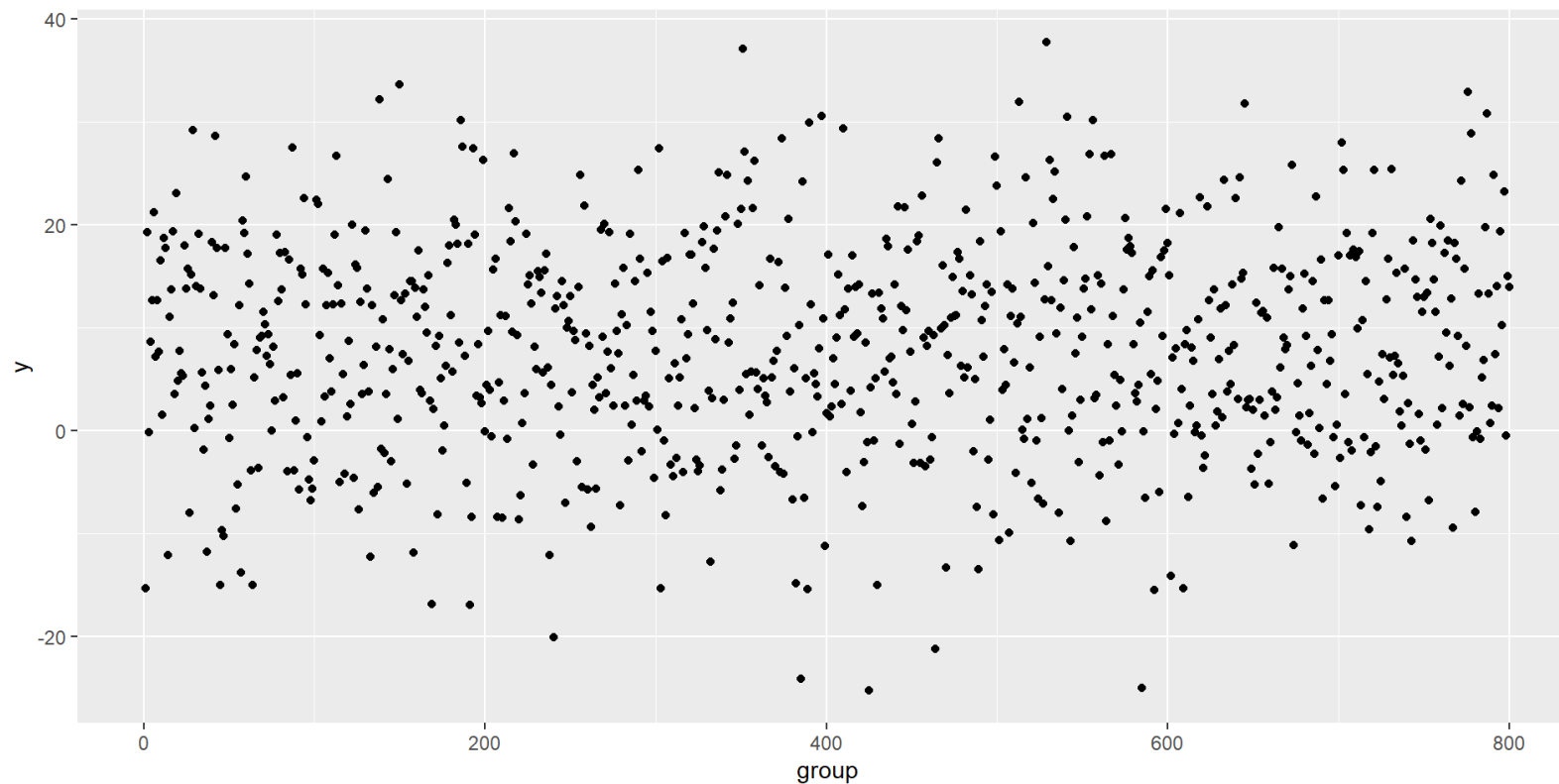
- Note the `fixed_param` and `iters`

Extract the relevant quantities

```
1 # extract it
2 y <- as.vector(sim$draws("y", format = "draws_matrix"))
3 sigma <- as.vector(sim$draws("sigma", format = "draws_matrix"))
4 theta <- as.vector(sim$draws("theta", format = "draws_matrix"))
5 mu <- as.vector(sim$draws("mu_print", format = "draws_matrix"))
6 tau <- as.vector(sim$draws("tau_print", format = "draws_matrix"))
```

Look at it (duh)

```
1 library(tidyverse)
2 my_data <- data.frame(y, group = 1:length(y))
3 my_data %>%
4   ggplot(aes(x = group, y = y)) +
5   geom_point()
```



The centered parameterization in code

```
1 data {  
2   int<lower=0> J;  
3   array[J] real y;  
4   array[J] real sigma;  
5 }  
6 parameters {  
7   real mu;  
8   real<lower=0> tau;  
9   array[J] real theta;  
10 }  
11 model {  
12   mu ~ normal(0, 5);  
13   tau ~ cauchy(0, 2.5);  
14   theta ~ normal(mu, tau);  
15   y ~ normal(theta, sigma);  
16 }
```


The noncentered parameterization in code

```
1 data {
2   int<lower=0> J;
3   array[J] real y;
4   array[J] real sigma;
5 }
6 parameters {
7   real mu;
8   real<lower=0> tau;
9   array[J] real var_theta;
10 }
11 transformed parameters {
12   array[J] real theta;
13   for (j in 1:J) theta[j] = tau * var_theta[j] + mu;
14 }
15 model {
16   mu ~ normal(0, 5);
17   tau ~ cauchy(0, 2.5);
18   var_theta ~ normal(0, 1);
19   y ~ normal(theta, sigma);
20 }
```

Running things from R

```
1 # Centered estimation model:
2 stan_data <- list(
3   J = length(y), y = y, sigma = sigma
4 )
5 one_level_cp <- cmdstan_model("src/one_level_cp.stan")
6
7 fit_cp <- one_level_cp$sample(
8   data = stan_data,
9   iter_warmup = 1000, iter_sampling = 1000,
10  chains = 4, parallel_chains = 4,
11  seed = 13, refresh = 0, adapt_delta = 0.99
12 )
```

Running MCMC with 4 parallel chains...

Chain 1 finished in 7.4 seconds.

Chain 4 finished in 8.2 seconds.

Chain 3 finished in 9.5 seconds.

Chain 2 finished in 13.8 seconds.

All 4 chains finished successfully.

Mean chain execution time: 9.7 seconds.

Total execution time: 14.0 seconds.

Checking diagnostics

```
1 fit_cp$cmdstan_diagnose()
```

```
Processing csv files: C:/Users/Chris/AppData/Local/Temp/RtmpmEiqtE/one_level_cp-  
202310152224-1-6bae41.csv, C:/Users/Chris/AppData/Local/Temp/RtmpmEiqtE/one_level_cp-  
202310152224-2-6bae41.csv, C:/Users/Chris/AppData/Local/Temp/RtmpmEiqtE/one_level_cp-  
202310152224-3-6bae41.csv, C:/Users/Chris/AppData/Local/Temp/RtmpmEiqtE/one_level_cp-  
202310152224-4-6bae41.csv
```

```
Checking sampler transitions treedepth.  
Treedepth satisfactory for all transitions.
```

```
Checking sampler transitions for divergences.  
No divergent transitions found.
```

```
Checking E-BFMI - sampler transitions HMC potential energy.  
The E-BFMI, 0.00, is below the nominal threshold of 0.30 which suggests that HMC may have  
trouble exploring the target distribution.  
If possible, try to reparameterize the model.
```

```
The following parameters had fewer than 0.001 effective draws per transition:
```

Running things from R

```
1 # noncentered estimation model:
2 one_level_ncp <- cmdstan_model("src/one_level_ncp.stan")
3
4 fit_ncp <- one_level_ncp$sample(
5   data = stan_data,
6   iter_warmup = 1000, iter_sampling = 1000,
7   chains = 4, parallel_chains = 4,
8   seed = 13, refresh = 0, adapt_delta = 0.99
9 )
```

Running MCMC with 4 parallel chains...

Chain 2 finished in 9.9 seconds.

Chain 4 finished in 10.4 seconds.

Chain 1 finished in 10.7 seconds.

Chain 3 finished in 15.9 seconds.

All 4 chains finished successfully.

Mean chain execution time: 11.7 seconds.

Total execution time: 16.0 seconds.

Checking diagnostics

```
1 fit_ncp$cmdstan_diagnose()
```

```
Processing csv files: C:/Users/Chris/AppData/Local/Temp/RtmpmEiqtE/one_level_ncp-  
202310152224-1-0e1b62.csv, C:/Users/Chris/AppData/Local/Temp/RtmpmEiqtE/one_level_ncp-  
202310152224-2-0e1b62.csv, C:/Users/Chris/AppData/Local/Temp/RtmpmEiqtE/one_level_ncp-  
202310152224-3-0e1b62.csv, C:/Users/Chris/AppData/Local/Temp/RtmpmEiqtE/one_level_ncp-  
202310152224-4-0e1b62.csv
```

```
Checking sampler transitions treedepth.  
Treedepth satisfactory for all transitions.
```

```
Checking sampler transitions for divergences.  
No divergent transitions found.
```

```
Checking E-BFMI - sampler transitions HMC potential energy.  
E-BFMI satisfactory.
```

```
Effective sample size satisfactory.
```

```
Split R-hat values satisfactory all parameters.
```

Comparing the two models

```
1 fit_cp$summary(c("mu", "tau"))
```

```
# A tibble: 2 × 10
```

	variable	mean	median	sd	mad	q5	q95	rhat	ess_bulk	ess_tail
	<chr>	<num>	<num>	<num>	<num>	<num>	<num>	<num>	<num>	<num>
1	mu	7.79	7.76	0.373	0.413	7.25	8.45	1.19	15.0	152.
2	tau	1.82	1.94	1.09	1.31	0.251	3.45	2.02	5.50	12.2

```
1 fit_ncp$summary(c("mu", "tau"))
```

```
# A tibble: 2 × 10
```

	variable	mean	median	sd	mad	q5	q95	rhat	ess_bulk	ess_tail
	<chr>	<num>	<num>	<num>	<num>	<num>	<num>	<num>	<num>	<num>
1	mu	7.88	7.87	0.355	0.340	7.28	8.47	1.00	6132.	2937.
2	tau	1.68	1.65	0.994	1.16	0.177	3.34	1.01	479.	1147.

First example wrap up

- The centered parameterization throws low-EBFMI warnings, occasional divergent transition warnings, and maximum treedepth reached warnings
- NCP increases efficiency (measured as ESS / run time)

Eight schools in class demo

Centered model:

$$y_j \sim \text{Normal}(\theta_j, \sigma_j), \quad j = 1, \dots, J$$

$$\theta_j \sim \text{Normal}(\mu, \tau), \quad j = 1, \dots, J$$

$$\mu \sim \text{Normal}(0, 10)$$

$$\tau \sim \text{half - Cauchy}(0, 10)$$

Non-centered analog:

$$\theta_j = \mu + \tau\eta_j, \quad j = 1, \dots, J$$

$$\eta_j \sim N(0, 1), \quad j = 1, \dots, J.$$

Key thing to note about Eight Schools

NCP vs. CP

- NCP replaces the vector θ with a vector η of i.i.d. standard normal parameters and then constructs θ deterministically from η by scaling by τ and shifting by μ

Key thing to note about Eight Schools

NCP vs. CP

- NCP replaces the vector θ with a vector η of i.i.d. standard normal parameters and then constructs θ deterministically from η by scaling by τ and shifting by μ
- To the code!

References

- Betancourt, M. and Girolami, M. 2013. Hamiltonian Monte Carlo for hierarchical models. <https://arxiv.org/abs/1312.0906>
- Gelman, A., Carlin, J. B., Stern, H. S., Dunson, D. B., Vehtari, A., and Rubin, D. B. 2013. Bayesian Data Analysis. Chapman & Hall/CRC Press, London, third edition.
- Rubin, D. B. 1981. Estimation in Parallel Randomized Experiments. *Journal of Educational and Behavioral Statistics*. 6:377–401.
- Stan Development Team. 2023. Stan Modeling Language Users Guide and Reference Manual. <https://mc-stan.org/users/documentation/>