Scale Estimates from RUVASH

David Gerard

2016-06-02

Abstract

I plot the scale estimates from RUVASH under different alternative settings. The scale estimates seem to be invariant to the alternative type, which makes sense since I give it truly null genes. The scale estimates could have been different because (1) `num.sv` estimated the number of confounders differently based on the alternative type (but this is not the case) or (2) the alternative type affects the estimates of the coefficients of the confounders or the variances. But neither of these appears to have caused the scale estimates to be different.

Simulation Setup

I ran through 200 repetitions of generating data from GTEX muscle data under the following parameter conditions:

- $n \in \{10, 20, 40\}$,
- $p = 1000$.
- $\pi_0 \in \{0.5, 0.9\}$,
- The alternative distribution being either spiky, near-normal, flattop, skew, big-normal, or bimodal, where these are the same alternatives defined in Stephens (2016) and the following table. New alternatives are generated every iteration.

Scenario	Alternative Distribution
Spiky	$0.4N(0, 0.25^2) + 0.2N(0, 0.5^2) + 0.2N(0, 1^2), 0.2N(0, 2^2)$
Near Normal	$2/3N(0, 1^2) + 1/3N(0, 2^2)$
Flattop	$(1/7)N(-1.5, .5^2) + N(-1, .5^2) + N(-.5, .5^2) + N(0, .5^2) + N(0.5, .5^2) + N(1.0, .5^2) + N(1.5, .5^2)$
Skew	$(1/4)N(-2, 2^2) + (1/4)N(-1, 1.5^2) + (1/3)N(0, 1^2) + (1/6)N(1, 1^2)$
Big-normal	$N(0, 4^2)$
Bimodal	$0.5N(-2, 1^2) + 0.5N(2, 1^2)$

I extracted the most expressed p genes from the GTEX muscle data and n samples are chosen at random. Half of these samples are randomly given the "treatment" label 1, the other half given the "control" label 0. Of the p genes, $\pi_0 p$ were chosen to be non-null. Signal was added by a Poisson-thinning approach, where the log-2 fold change was sampled from one of five the alternative models above. That is

$$A_1, \ldots, A_{p/2} \sim f \tag{1}$$

$$B_i = 2^{A_i} \text{ for } i = 1, \ldots, p/2, \tag{2}$$

where f is from the table above. If $A_i > 0$ then we replace $Y_{[1:(n/2),i]}$ with $Binom(Y_{[j,i]}, 1/B_i)$ for $j = 1, \ldots, n/2$. If $A_i < 0$ then we replace $Y_{[(n/2+1):n,i]}$ with $Binom(Y_{[j,i]}, B_i)$ for $j = n/2 + 1, \ldots, n$.

I now describe the justification for this. Suppose that

$$Y_{ij} \sim Poisson(\lambda_j). \tag{3}$$

Let x_i be the indicator of treatment vs control for individual i. Let Ω be the set of non-null genes. Let Z be the new dataset derived via the steps above. That is

$$Z_{ij}|Y_{ij} = \begin{cases} Binom(Y_{ij}, 2^{A_j x_i}) & \text{if } A_j < 0 \text{ and } j \in \Omega \\ Binom(Y_{ij}, 2^{-A_j(1-x_i)}) & \text{if } A_j > 0 \text{ and } j \in \Omega \\ Y_{ij} & \text{if } j \notin \Omega. \end{cases} \tag{4}$$

Then

$$Z_{ij}|A_j, A_j < 0, j \in \Omega \sim Poisson(2^{A_j x_i}\lambda_j) \tag{5}$$
$$Z_{ij}|A_j, A_j > 0, j \in \Omega \sim Poisson(2^{-A_j(1-x_i)}\lambda_j), \tag{6}$$

and

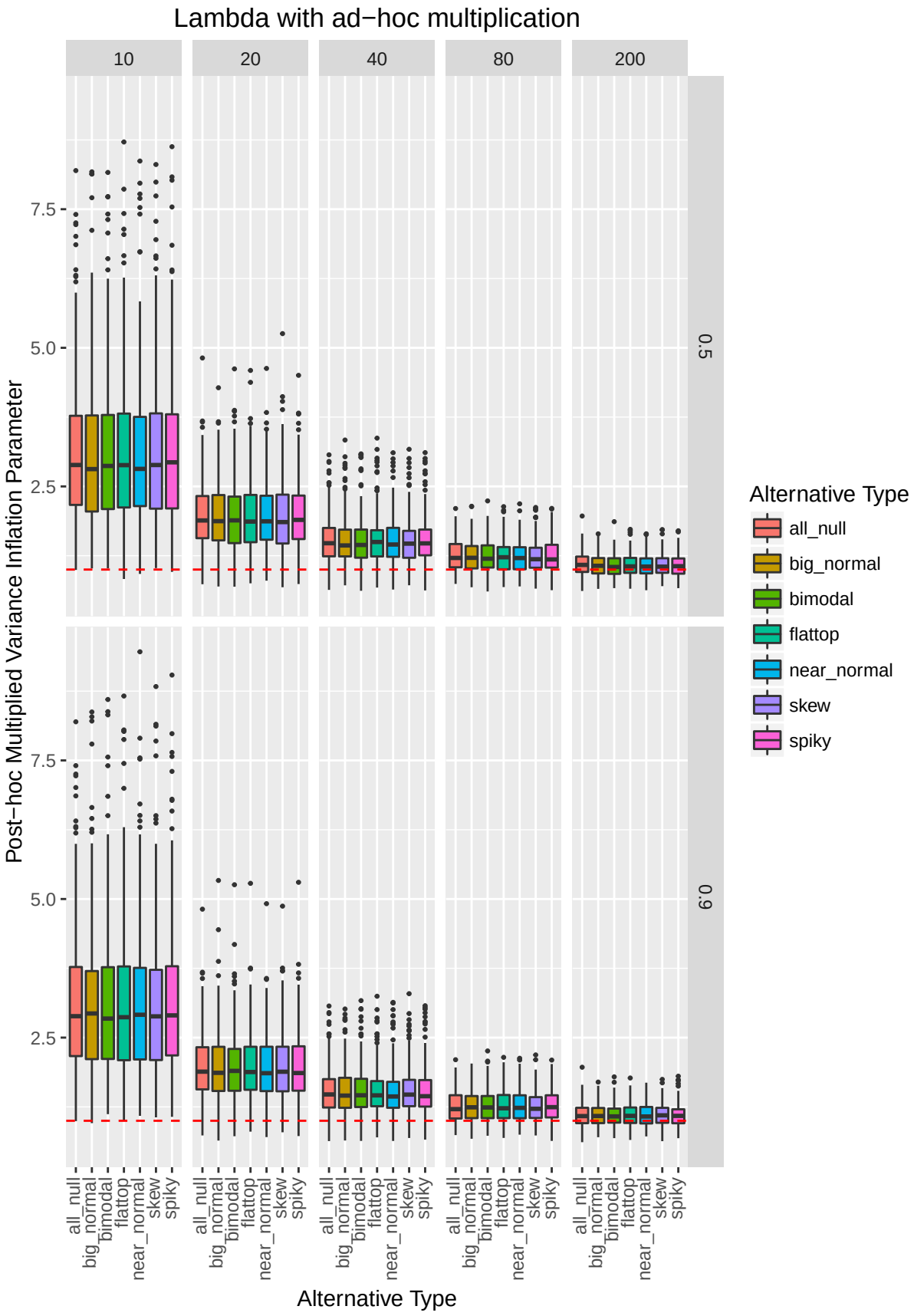
$$E[\log_2(Z_{ij}) - \log_2(Z_{kj})|A_j, A_j < 0, j \in \Omega] \approx A_j x_i - A_j x_k, \text{ and} \tag{7}$$
$$E[\log_2(Z_{ij}) - \log_2(Z_{kj})|A_j, A_j > 0, j \in \Omega] \approx -A_j(1-x_i) + A_j(1-x_k). \tag{8}$$

if individual i is in the treatment group and individual k is in the control group, then this just equals A_j. I treat the A_j's as the true coefficient values when calculating the MSE below.

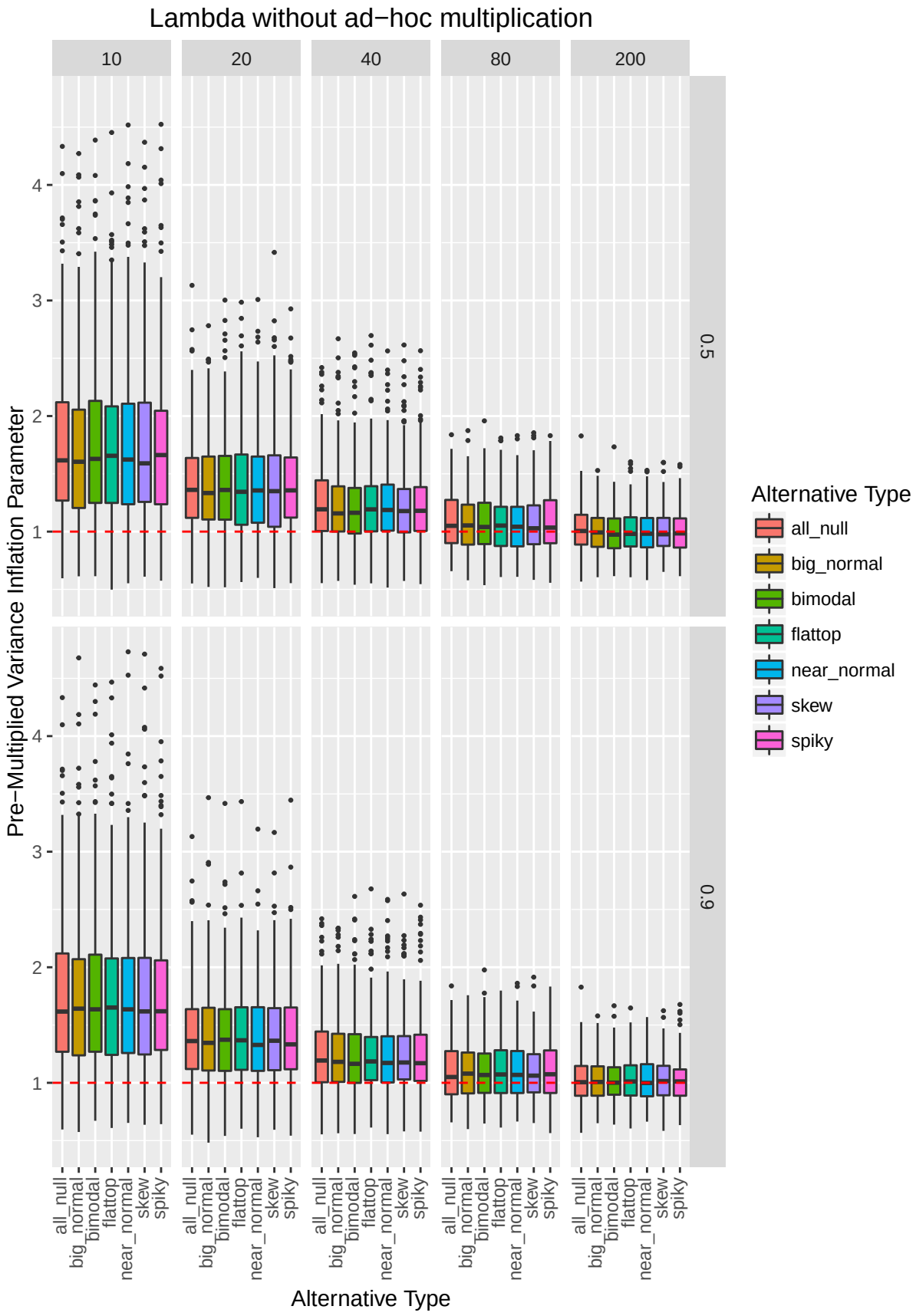
Plots

```
library(ggplot2)
load("scale_muscle_ruvash.Rd")
load("numsv_muscle_ruvash.Rd")
par_vals <- read.csv("par_vals.csv")
par_vals$scale <- sapply(ruvash_lim_scale, c)
par_vals$numsv <- sapply(ruvash_lim_numsv, c)
par_vals$posthoc_mult <- (par_vals$Nsamp * 2) / (par_vals$Nsamp * 2 - 2 - par_vals$numsv)
par_vals$premult_lambda <- par_vals$scale / par_vals$posthoc_mult
par_vals$Nsamp <- par_vals$Nsamp * 2

alt_type_seq <- unique(par_vals$alt_type)

plot_df <- par_vals[par_vals$alt_type != "all_null", ]
all_null1 <- par_vals[par_vals$alt_type == "all_null", ]
all_null1$nullpi <- 0.5
all_null2 <- par_vals[par_vals$alt_type == "all_null", ]
all_null2$nullpi <- 0.9
plot_df <- rbind(plot_df, all_null1, all_null2)
```

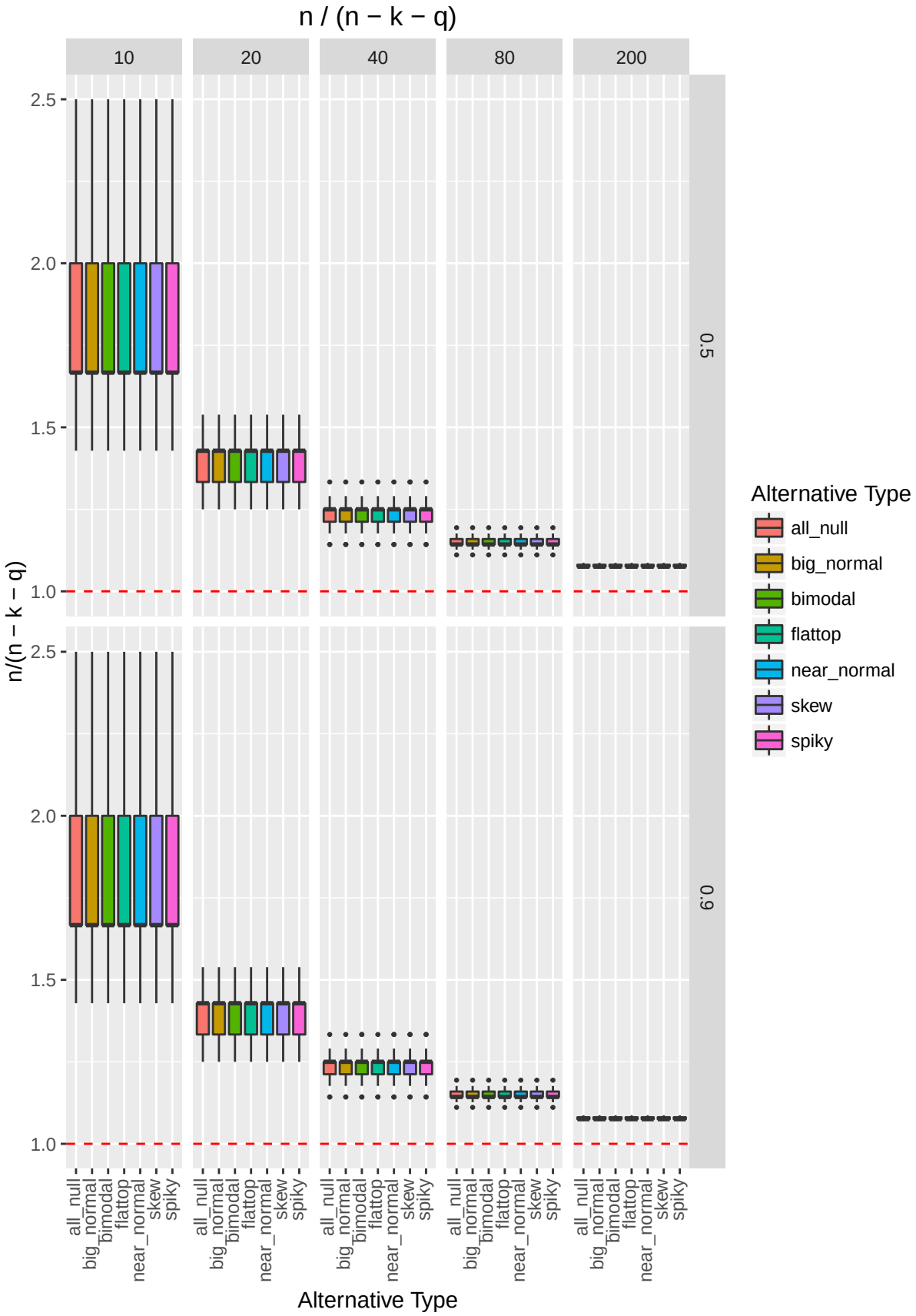
```r
ggplot(data = plot_df,
       mapping = aes(x = alt_type, y = scale, fill = alt_type)) +
    facet_grid(nullpi ~ Nsamp) +
    geom_boxplot(outlier.size = 0.5) +
    geom_hline(yintercept = 1, col = 2, lty = 2) +
    theme(axis.text.x = element_text(angle = 90, hjust = 1, vjust = 0.3)) +
    xlab("Alternative Type") +
    ylab("Post-hoc Multiplied Variance Inflation Parameter") +
    guides(fill = guide_legend(title="Alternative Type")) +
    ggtitle("Lambda with ad-hoc multiplication")
```


Lambda with ad−hoc multiplication
10
20
40
80
200
0.5
0.9
Post−hoc Multiplied Variance Inflation Parameter
7.5
5.0
2.5
all_null
big_normal
bimodal
flattop
near_normal
skew
spiky
Alternative Type

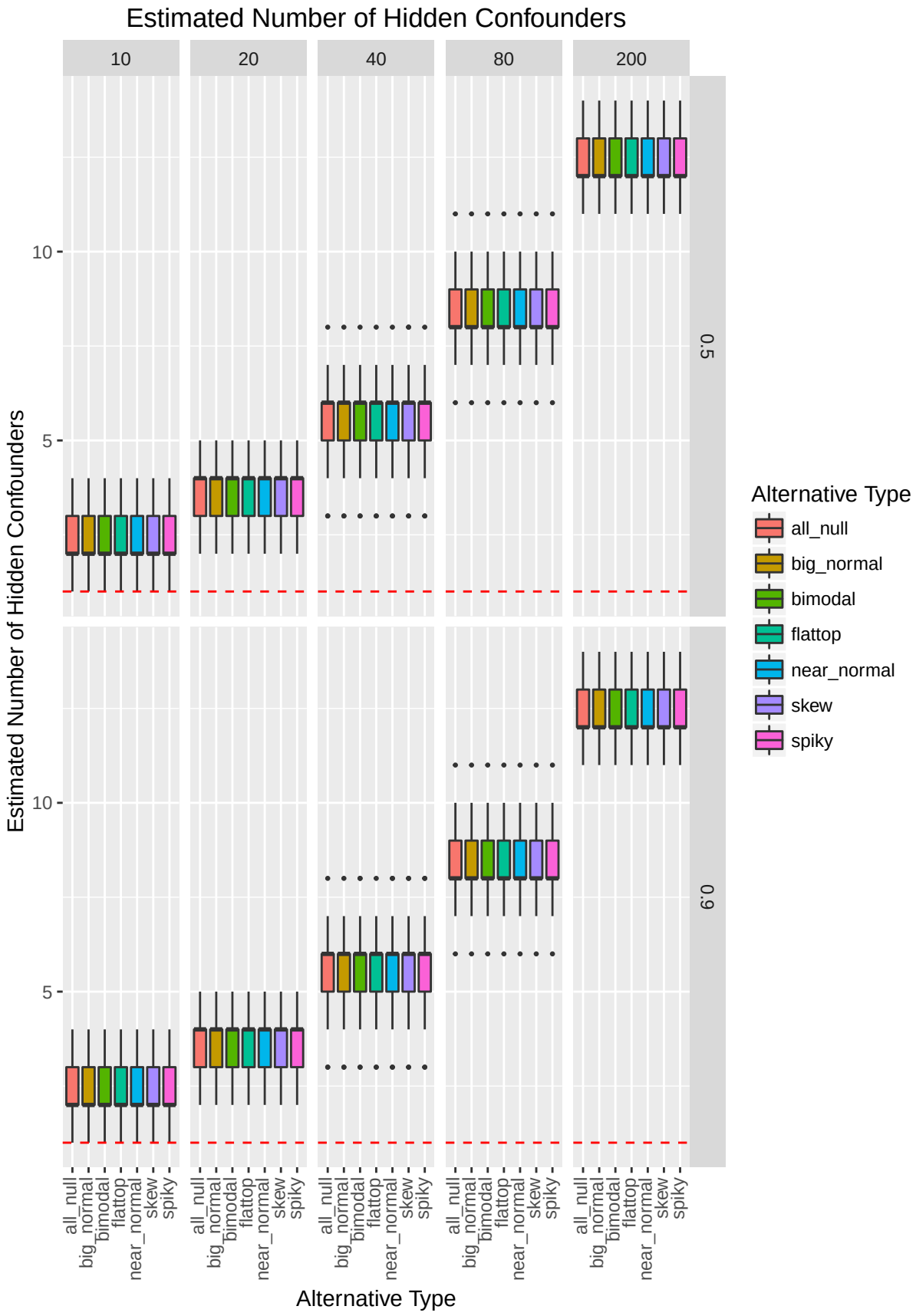
```r
ggplot(data = plot_df,
       mapping = aes(x = alt_type, y = premult_lambda, fill = alt_type)) +
    facet_grid(nullpi ~ Nsamp) +
    geom_boxplot(outlier.size = 0.5) +
    geom_hline(yintercept = 1, col = 2, lty = 2) +
    theme(axis.text.x = element_text(angle = 90, hjust = 1, vjust = 0.3)) +
    xlab("Alternative Type") +
    ylab("Pre-Multiplied Variance Inflation Parameter") +
    guides(fill = guide_legend(title="Alternative Type")) +
    ggtitle("Lambda without ad-hoc multiplication")
```


Lambda without ad−hoc multiplication
10
20
40
80
200
0.5
0.9
Pre−Multiplied Variance Inflation Parameter
1
2
3
4
all_null
big_normal
bimodal
flattop
near_normal
skew
spiky
Alternative Type

```r
ggplot(data = plot_df,
       mapping = aes(x = alt_type, y = posthoc_mult, fill = alt_type)) +
    facet_grid(nullpi ~ Nsamp) +
    geom_boxplot(outlier.size = 0.5) +
    geom_hline(yintercept = 1, col = 2, lty = 2) +
    theme(axis.text.x = element_text(angle = 90, hjust = 1, vjust = 0.3)) +
    xlab("Alternative Type") +
    ylab("n/(n - k - q)") +
    guides(fill = guide_legend(title="Alternative Type")) +
    ggtitle("n / (n - k - q)")
```


n / (n − k − q)
10
20
40
80
200
0.5
0.9
2.5
2.0
1.5
1.0
n/(n − k − q)
all_null
big_normal
bimodal
flattop
near_normal
skew
spiky
Alternative Type

```
ggplot(data = plot_df,
       mapping = aes(x = alt_type, y = numsv, fill = alt_type)) +
    facet_grid(nullpi ~ Nsamp) +
    geom_boxplot(outlier.size = 0.5) +
    geom_hline(yintercept = 1, col = 2, lty = 2) +
    theme(axis.text.x = element_text(angle = 90, hjust = 1, vjust = 0.3)) +
    xlab("Alternative Type") +
    ylab("Estimated Number of Hidden Confounders") +
    guides(fill = guide_legend(title="Alternative Type")) +
    ggtitle("Estimated Number of Hidden Confounders")
```


Estimated Number of Hidden Confounders
10
20
40
80
200
0.5
0.9
10
5
Estimated Number of Hidden Confounders
all_null
big_normal
bimodal
flattop
near_normal
skew
spiky
Alternative Type

References

Stephens, Matthew. 2016. "False Discovery Rates: A New Deal." *BioRxiv*. Cold Spring Harbor Labs Journals, 038216.