Power analysis on skew normal fitting for right tail

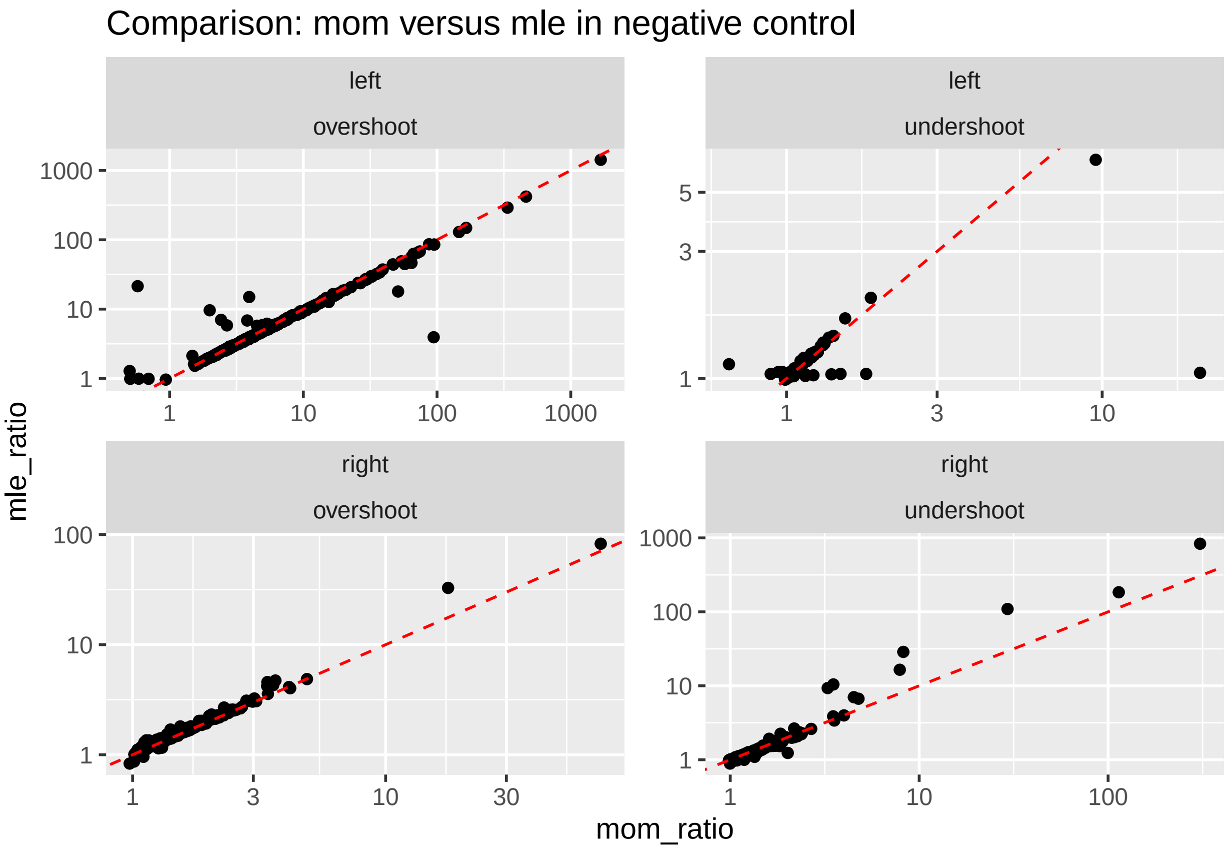
Some notation

In this report, define A = subsample fitted probability; B = subsample empirical tail probability; C = full sample fitted probability and D = full sample empirical tail probability. I consider the analysis for $A/D(D/A)$ and $A/B(B/A)$.

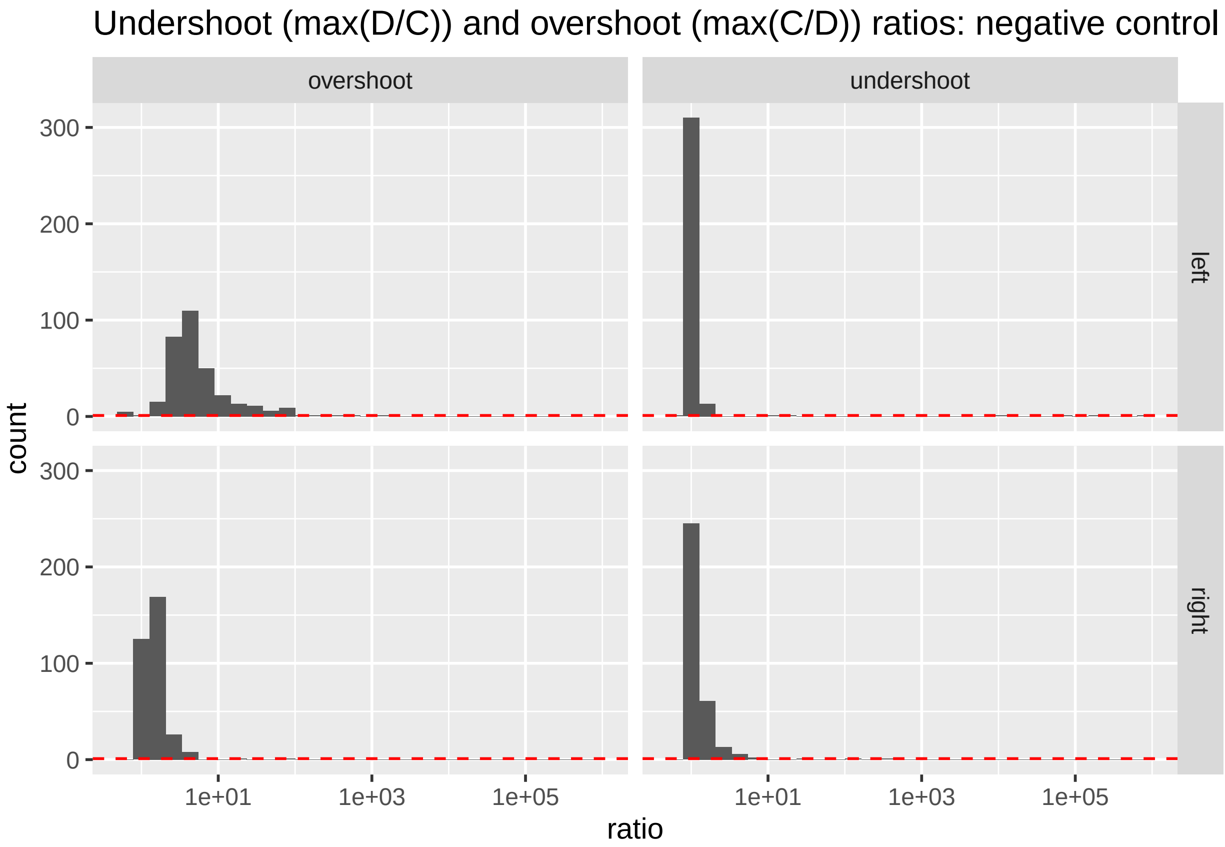
1.Load results for fitting comparison

```r
library(dplyr)
library(ggplot2)
library(tidyverse)
undershoot <- read_csv("undershoot_res_fit.csv")[,-1]
overshoot <- read_csv("overshoot_res_fit.csv")[,-1]
quantile_list <- seq(0.03, 0.97, length.out = 10)
no_sam <- round(exp(seq(log(1e3), log(5e4), length.out = 10)))
# rearrange the data frame
B <- 100
undershoot_df <- data.frame(id = rep(1:B, 10*10),
                            ratio_value = 0,
                            no_sam = 0,
                            ratio_quantile = 0)
overshoot_df <- data.frame(id = rep(1:B, 10*10),
                           ratio_value = 0,
                           no_sam = 0,
                           ratio_quantile = 0)

# i: quantile; j: no of sample
for (i in 1:10) {
  for (j in 1:10) {
    start <- (j - 1 + (i-1)*10)*B +1
    end <- (j + (i-1)*10)*B
    undershoot_df[start:end, 2] <- as.vector(undershoot[(((j-1)*B+1) :(j*B)), (i-1)*3+32])[[1]]
    undershoot_df[start:end, 3] <- rep(no_sam[j], B)
    undershoot_df[start:end, 4] <- rep(quantile_list[i], B)
    overshoot_df[start:end, 2] <- as.vector(overshoot[(((j-1)*B+1) :(j*B)), (i-1)*3+32])[[1]]
    overshoot_df[start:end, 3] <- rep(no_sam[j], B)
    overshoot_df[start:end, 4] <- rep(quantile_list[i], B)
  }
}

# load the oracle ratio
mle_param_nc <- read_csv("figures/power_exploration/sknorm_tail_prob_500000_resamples_0.96_percentile/pa
mom_param_nc <- read_csv("figures/power_exploration/sknorm_tail_prob_500000_resamples_0.96_percentile/mo
mle_param_twosides <- t(mle_param_nc[,-1])
mom_param_twosides <- t(mom_param_nc[,-1])
mle_overshoot_ratio <- as.numeric(mle_param_twosides[, 6])
```

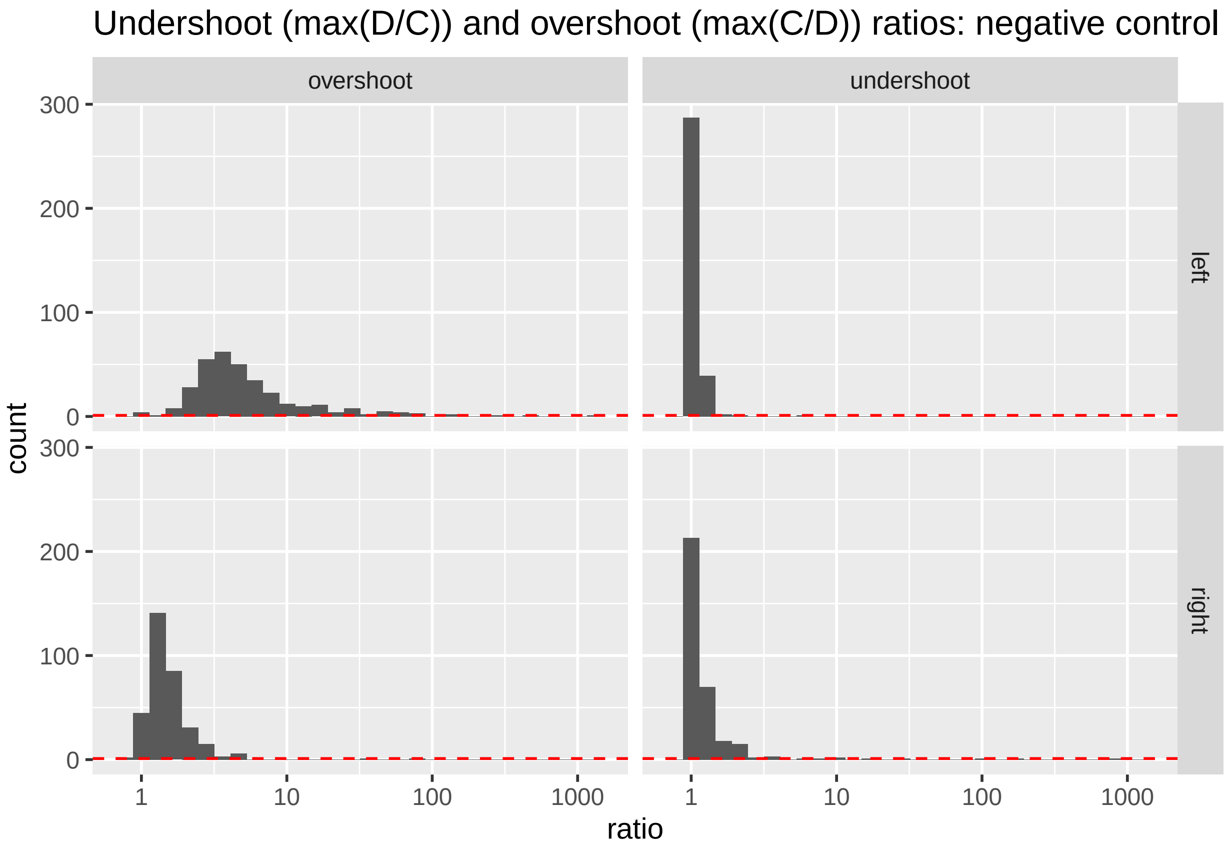
```r
mle_undershoot_ratio <- as.numeric(mle_param_twosides[, 7])
mom_overshoot_ratio <- as.numeric(mom_param_twosides[, 6])
mom_undershoot_ratio <- as.numeric(mom_param_twosides[, 7])
quantile_list <- seq(0.01, 0.99, length.out = 10)

# specify oracle value
n <- 5e5
overshoot_set <- data.frame(index = numeric(10), ratio = numeric(10))
undershoot_set <- data.frame(index = numeric(10), ratio = numeric(10))
tail_list <- (10:(0.04*n)) / n

# find distributions based on right tail
for (r in 1:10){
  # overshoot
  dist_right <- abs(mom_overshoot_ratio[331:660] - quantile(mom_overshoot_ratio[331:660], quantile_list
  overshoot_set[r, 1] <- which(dist_right == min(dist_right))
  overshoot_set[r, 2] <- mom_overshoot_ratio[which(dist_right == min(dist_right)) + 330]

  # undershoot right
  dist_right <- abs(mom_undershoot_ratio[331:660] - quantile(mom_undershoot_ratio[331:660], quantile_lis
  undershoot_set[r, 1] <- which(dist_right == min(dist_right))
  undershoot_set[r, 2] <- mom_undershoot_ratio[which(dist_right == min(dist_right)) + 330]
}




# accuracy using 5e5 mom estimate
mle_groundtruth <- data.frame(ratio = c(mle_overshoot_ratio, mle_undershoot_ratio),
                              type = c(rep("overshoot", 660), rep("undershoot", 660)),
                              tail = c(rep(c(rep("left", 330),rep("right", 330)) , 2)),
                              id = c(rep(1:330, 2)))

mom_groundtruth <- data.frame(ratio = c(mom_overshoot_ratio, mom_undershoot_ratio),
                              type = c(rep("overshoot", 660), rep("undershoot", 660)),
                              tail = c(rep(c(rep("left", 330),rep("right", 330)) , 2)),
                              id = c(rep(1:330, 2)))

groundtruth_comp <- data.frame(mom_ratio = c(mom_overshoot_ratio, mom_undershoot_ratio),
                               mle_ratio = c(mle_overshoot_ratio, mle_undershoot_ratio),
                               type = c(rep("overshoot", 660), rep("undershoot", 660)),
                               tail = c(rep(c(rep("left", 330),rep("right", 330)) , 2)),
                               id = c(rep(1:330, 2)))

groundtruth_comp |>
  filter(mom_ratio / mle_ratio <= 100) |>
  ggplot(aes_string(x = "mom_ratio", y = "mle_ratio")) +
  facet_wrap(tail~type, scales = "free") +
  geom_point() +
  scale_y_log10() +
  scale_x_log10() +
  geom_abline(linetype = "dashed", color = "red") +
  labs(title = "Comparison: mom versus mle in negative control")
```



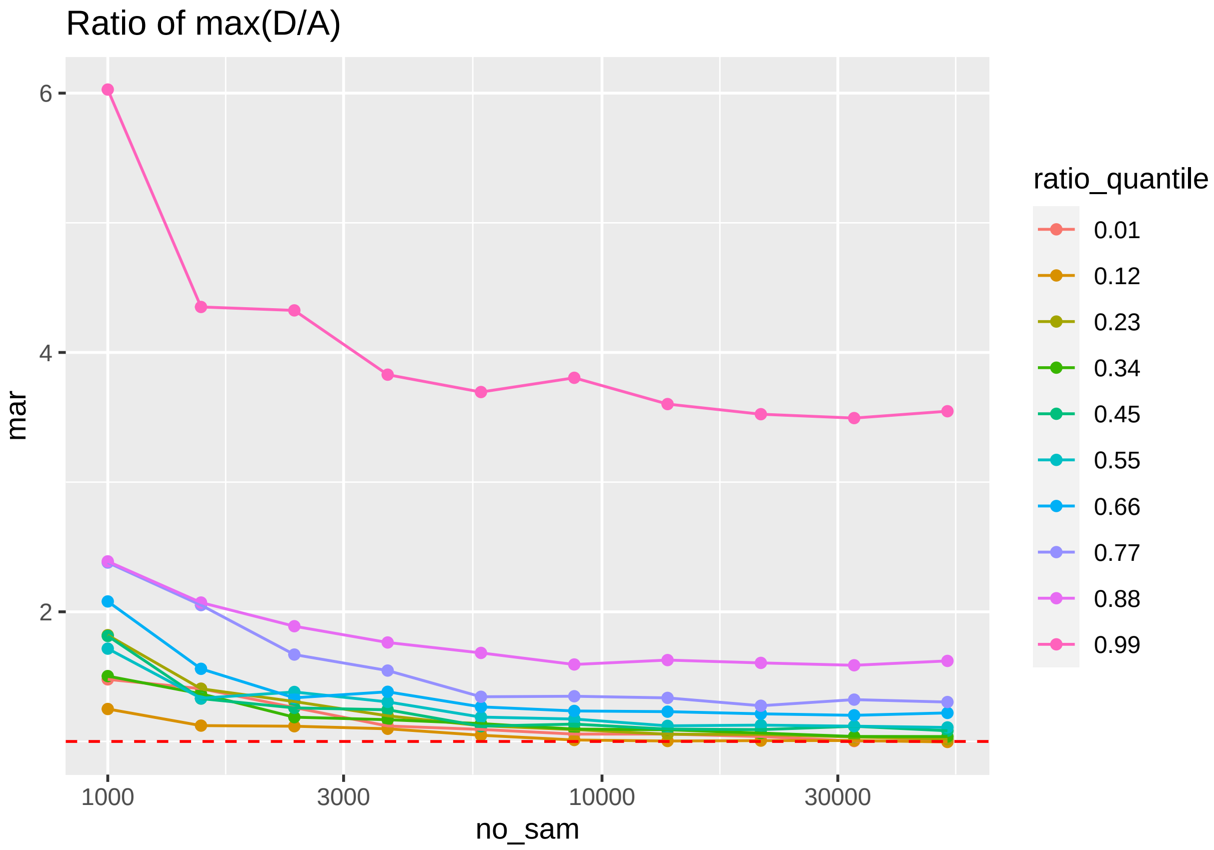
```
mom_groundtruth |>
  ggplot(aes_string(x = "ratio")) +
  facet_grid(tail~type) +
  geom_histogram() +
  scale_x_log10() +
  geom_hline(yintercept = 1, linetype = "dashed", colour = "red") +
  labs(title = "Undershoot (max(D/C)) and overshoot (max(C/D)) ratios: negative control")
```



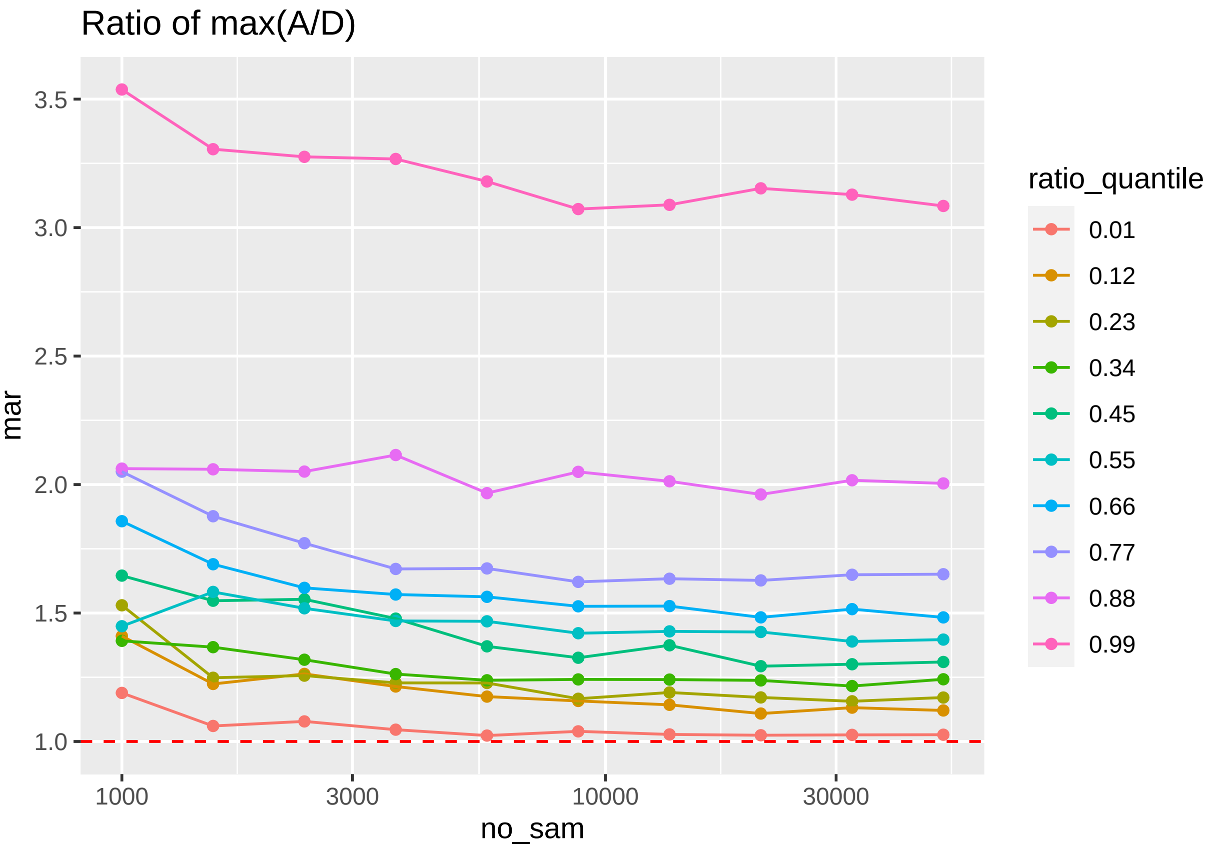
```
mle_groundtruth |>
  ggplot(aes_string(x = "ratio")) +
  facet_grid(tail~type) +
  geom_histogram() +
  scale_x_log10() +
  geom_hline(yintercept = 1, linetype = "dashed", colour = "red") +
  labs(title = "Undershoot (max(D/C)) and overshoot (max(C/D)) ratios: negative control")
```



```r
# accuracy matrix for undershoot matrix
undershoot_acc <- matrix(abs(undershoot_df$ratio_value), 100, 100)
undershoot_ame <- data.frame(mar = apply(undershoot_acc, 2, mean),
                             no_sam = rep(no_sam, 10),
                             ratio_quantile = as.character(round(rep(quantile_list, each = 10), 2)))
undershoot_ame |>
  ggplot(aes_string(x = "no_sam", y = "mar", colour = "ratio_quantile")) +
  scale_x_log10() +
  geom_point() +
  geom_line() +
  geom_hline(yintercept = 1, linetype = "dashed", colour = "red") +
  labs(title = "Ratio of max(D/A)")
```



```
# accuracy matrix for overshoot matrix
overshoot_acc <- matrix(abs(overshoot_df$ratio_value), 100, 100)
overshoot_ame <- data.frame(mar = apply(overshoot_acc, 2, mean),
                            no_sam = rep(no_sam, 10),
                            ratio_quantile = as.character(round(rep(quantile_list, each = 10), 2)))
overshoot_ame |>
  ggplot(aes_string(x = "no_sam", y = "mar", colour = "ratio_quantile")) +
  scale_x_log10() +
  geom_point() +
  geom_line() +
  geom_hline(yintercept = 1, linetype = "dashed", colour = "red") +
  labs(title = "Ratio of max(A/D)")
```



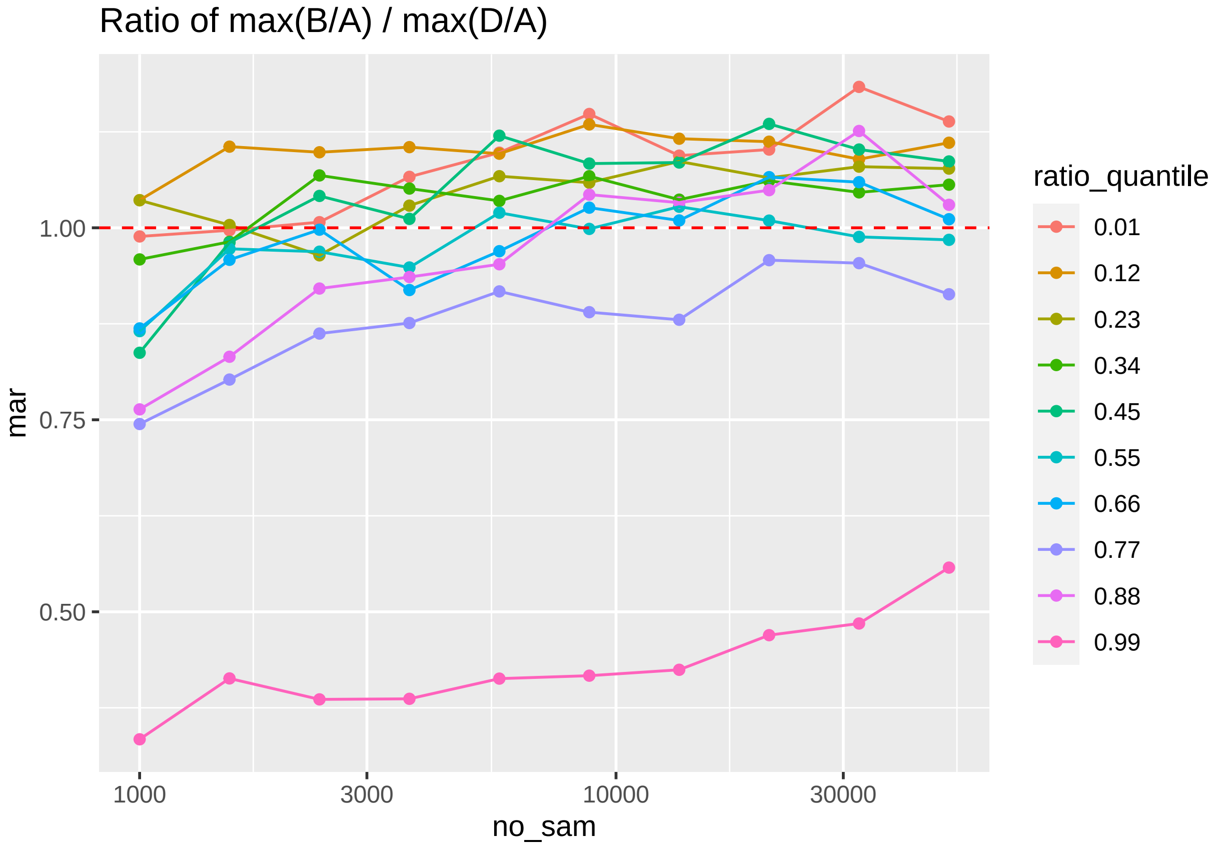
```r
# store D/A (A/D)
gt_overshoot_curve <- overshoot_df$ratio_value
gt_undershoot_curve <- undershoot_df$ratio_value
```

Here we clearly see the ratio $\max[A/D](max[D/A])$ approaches $\max[C/D](\max[D/C])$ very fast. This indicates the error decrease rather fast in estimating the tail probability at least in average sense. In the next section, we mainly consider the changes for the ratio $\max[A/B](max[B/A])$ and how does this approach $\max[C/D](\max[D/C])$.

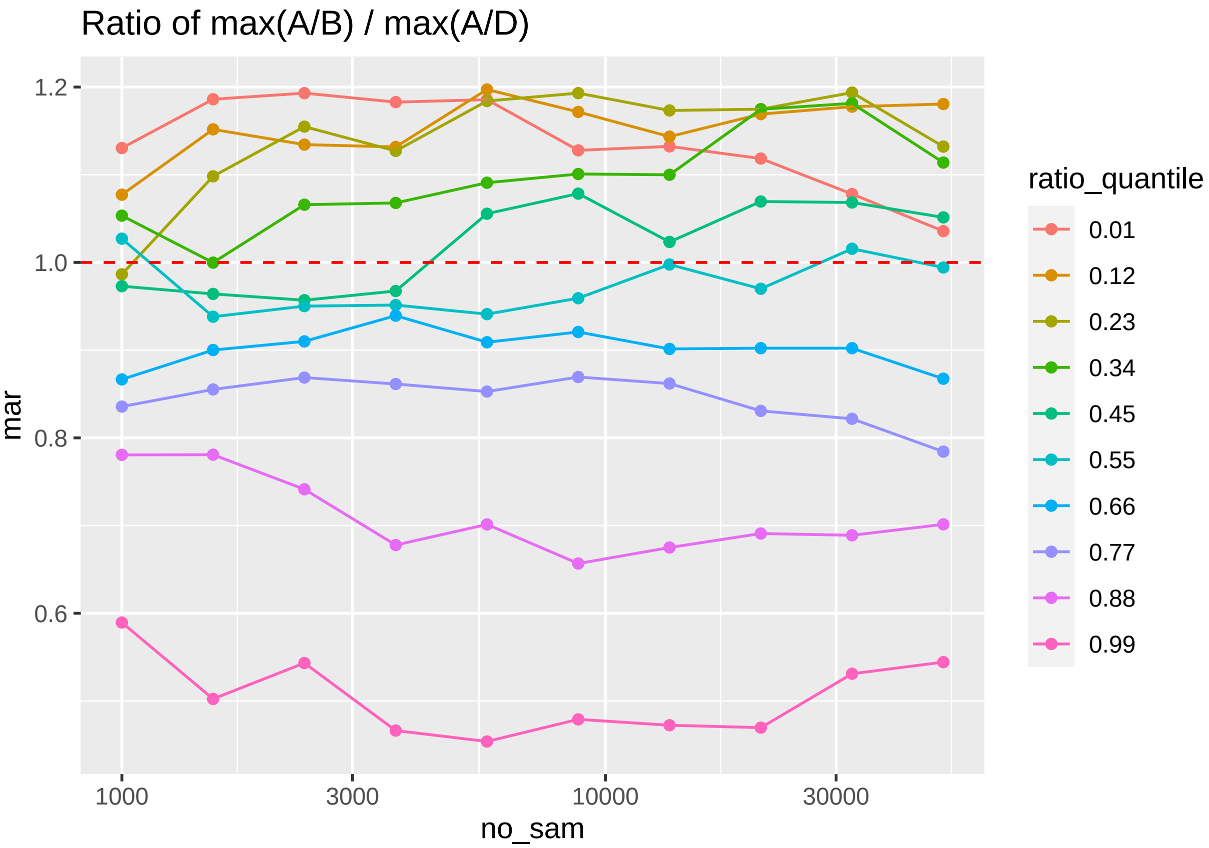
1.Load results for power comparison

```r
undershoot <- read_csv("undershoot_res_power.csv")[,-1]
overshoot <- read_csv("overshoot_res_power.csv")[,-1]
quantile_list <- seq(0.01, 0.99, length.out = 10)
no_sam <- round(exp(seq(log(1e3), log(5e4), length.out = 10)))
# rearrange the data frame
B <- 100
undershoot_df <- data.frame(id = rep(1:B, 10*10),
                            ratio_value = 0,
                            no_sam = 0,
                            ratio_quantile = 0)
overshoot_df <- data.frame(id = rep(1:B, 10*10),
                           ratio_value = 0,
                           no_sam = 0,
                           ratio_quantile = 0)

# i: quantile; j: no of sample
```

```r
for (i in 1:10) {
  for (j in 1:10) {
    start <- (j - 1 + (i-1)*10)*B +1
    end <- (j + (i-1)*10)*B
    undershoot_df[start:end, 2] <- as.vector(undershoot[(((j-1)*B+1) :(j*B)), (i-1)*3+32])[[1]]
    undershoot_df[start:end, 3] <- rep(no_sam[j], B)
    undershoot_df[start:end, 4] <- rep(quantile_list[i], B)
    overshoot_df[start:end, 2] <- as.vector(overshoot[(((j-1)*B+1) :(j*B)), (i-1)*3+32])[[1]]
    overshoot_df[start:end, 3] <- rep(no_sam[j], B)
    overshoot_df[start:end, 4] <- rep(quantile_list[i], B)
  }
}


# load the oracle ratio
undershoot_acc <- matrix(abs(undershoot_df$ratio_value / gt_undershoot_curve), 100, 100)
undershoot_ame <- data.frame(mar = apply(undershoot_acc, 2, mean),
                             no_sam = rep(no_sam, 10),
                             ratio_quantile = as.character(round(rep(quantile_list, each = 10), 2)))
undershoot_ame |>
  ggplot(aes_string(x = "no_sam", y = "mar", colour = "ratio_quantile")) +
  scale_x_log10() +
  geom_point() +
  geom_line() +
  geom_hline(yintercept = 1, linetype = "dashed", colour = "red") +
  labs(title = "Ratio of max(B/A) / max(D/A)")
```



```r
# accuracy matrix for overshoot matrix
overshoot_acc <- matrix(abs(overshoot_df$ratio_value / gt_overshoot_curve), 100, 100)
overshoot_ame <- data.frame(mar = apply(overshoot_acc, 2, mean),
                            no_sam = rep(no_sam, 10),
                            ratio_quantile = as.character(round(rep(quantile_list, each = 10), 2)))
overshoot_ame |>
  ggplot(aes_string(x = "no_sam", y = "mar", colour = "ratio_quantile")) +
  scale_x_log10() +
  geom_point() +
  geom_line() +
  geom_hline(yintercept = 1, linetype = "dashed", colour = "red") +
  labs(title = "Ratio of max(A/B) / max(A/D)")
```


Ratio of max(A/B) / max(A/D)
mar
no_sam
ratio_quantile