

Hypothesis test

Felix Schönbrodt

Tue Nov 29 15:37:33 2016

```
# plotting of results
library(ggplot2)
library(dplyr)
library(tidyr)
library(reshape2)

load("res.wide.red.boost.RData")
res.hyp <- res.wide.red %>% select(1:8, b0_estimate, b0_p.value, skewtest_p.value, 41:45) %>% filter(!method %in% c("pcurve", "pcurve.lack"))

# define critical p-value for each method
res.hyp$p.crit <- .05

# merge two p-value columns into one
res.hyp$p.value <- ifelse(!is.na(res.hyp$b0_p.value), res.hyp$b0_p.value, res.hyp$skewtest_p.value)
res.hyp <- res.hyp %>% select(-b0_p.value, -skewtest_p.value)

# compute rejection:
# Reject H0 if test is significant AND estimate in correct direction.
# In case of p-curve skewness tests, there is no estimate; estimate is set to NA there.
res.hyp$H0.reject <- (res.hyp$p.value < res.hyp$p.crit) & (is.na(res.hyp$b0_estimate) | res.hyp$b0_estimate > 0)

# Add combined hypothesis test: PETPEESE + 3PSM
PP3 <- res.hyp %>% filter(method %in% c("PETPEESE.lm", "3PSM")) %>% group_by(id, condition, k, delta, qrpEnv, selProp, tau, delta.label, k)
  H0.reject = H0.reject[1] & H0.reject[2] & !is.na(H0.reject[1]) & !is.na(H0.reject[2]),
  method="PP3",
  p.value = NA
)

res.hyp <- bind_rows(res.hyp, PP3)

## Warning in bind_rows_(x, .id): binding factor and character vector,
## coercing into character vector
```

```

save(res.hyp, file="res.hyp.boost.RData", compress="gzip")
#load(file="res.hyp.RData")

# Compute summary measures across replications
hyp.summ <- res.hyp %>% group_by(condition, k, k.label, delta, delta.label, grpEnv, grp.label, method) %>% dplyr::summarise(
  H0.reject = sum(H0.reject, na.rm=TRUE)/sum(!is.na(H0.reject)),
  n.simulations = n()
)

# loop plot for H0

# show partly loop style
library(gtools)

# new labels for meaningful order
#hyp.summ$grp.label <- factor(hyp.summ$grpEnv, labels=c("QRP = 0 (none)", "QRP = 1 (med)", "QRP = 2 (high)"))

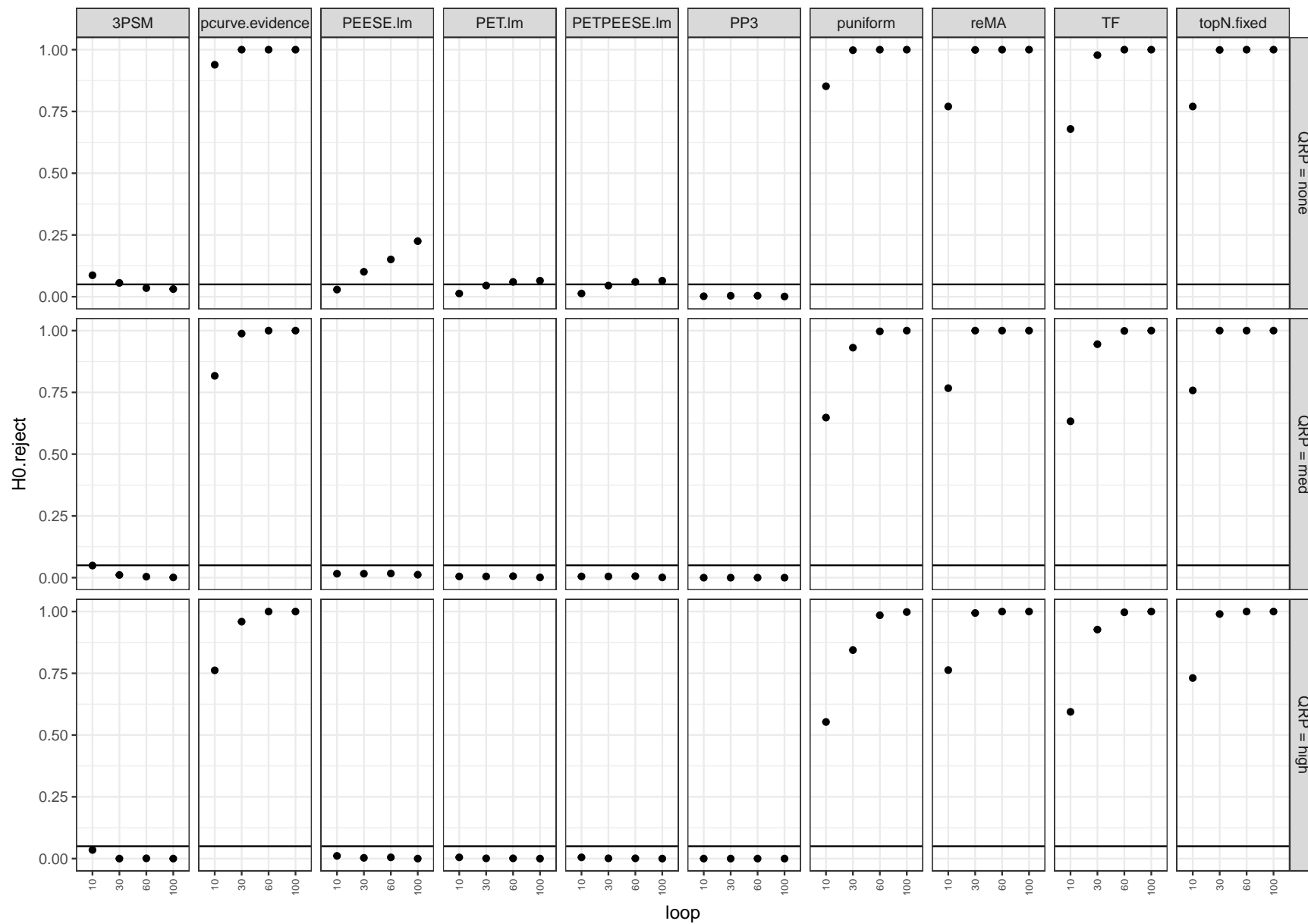
# order two variables into one loop
hyp.summ <- hyp.summ %>% mutate(loop = paste0(k))

# order loop factor alphabetically
hyp.summ$loop <- factor(hyp.summ$loop, levels = mixedsort(unique(hyp.summ$loop)))

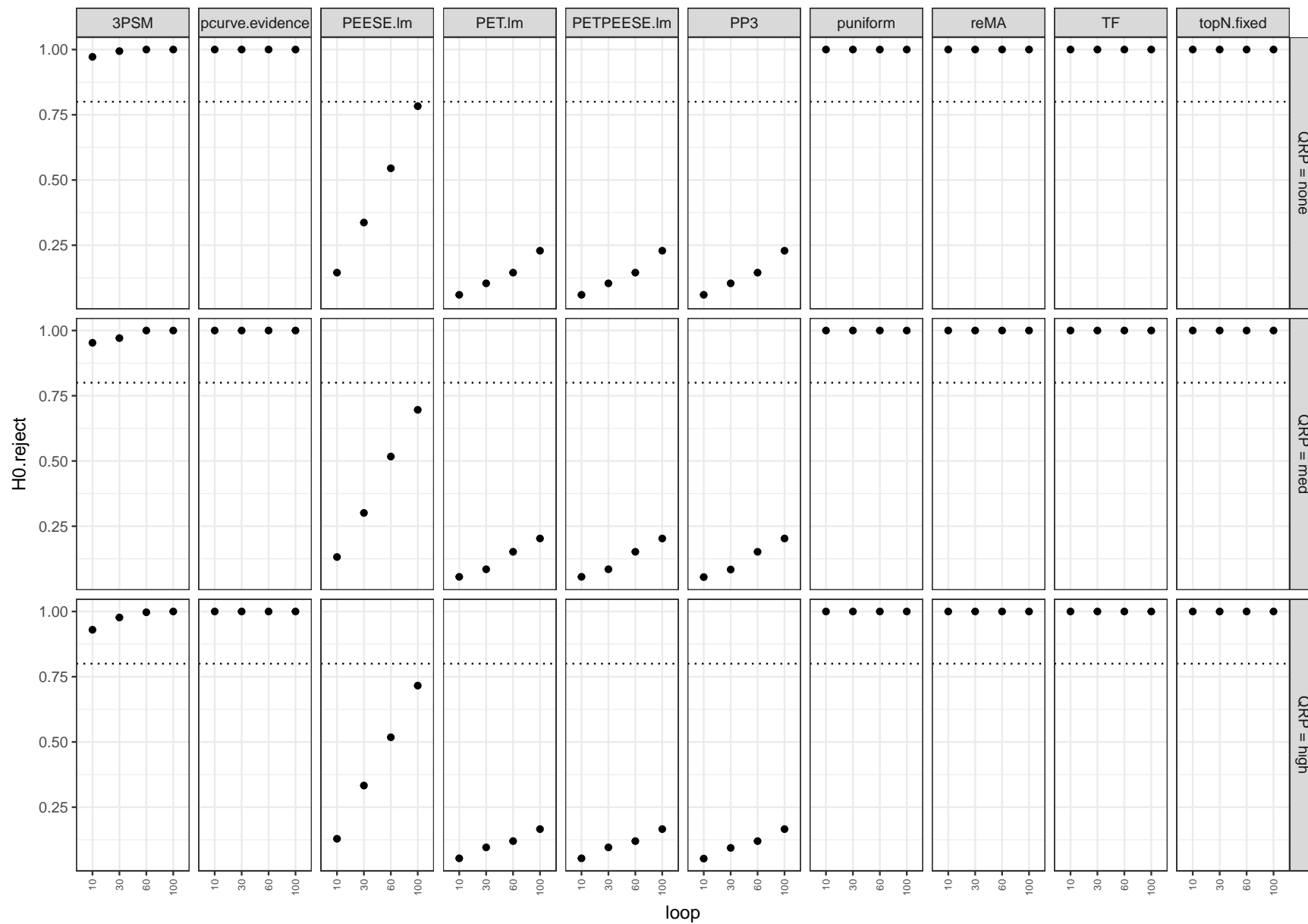
# H0

```

Under H0 – nominal Type I error? (for selProp = 60%)



Under H1 – Power? (for selProp = 60%, delta=0.5)



```

# Compute rejection ratios

RR <- hyp.summ %>% ungroup() %>%
  select(condition, k, delta, qrp.label, loop, method, H0.reject)

RR$TypeI <- RR$H0.reject
RR$TypeI[RR$delta!=0] <- NA

RR$TypeII <- 1-RR$H0.reject
RR$TypeII[RR$delta==0] <- NA
RR$Power <- 1-RR$TypeII

# compare delta==0 against delta==0.5
H1 <- 0.5
RR.H0 <- RR %>% filter(delta == 0) %>% select(condition, k, qrp.label, loop, method, TypeI)
RR.H1 <- RR %>% filter(delta == H1) %>% select(k, qrp.label, loop, method, Power)

glimpse(RR.H0)

## Observations: 120
## Variables: 6
## $ condition <int> 193, 193, 193, 193, 193, 193, 193, 193, 193, 193, 19...
## $ k <dbl> 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 30, 30, 30, ...
## $ qrp.label <fctr> QRP = none, QRP = none, QRP = none, QRP = none, QRP...
## $ loop <fctr> 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 30, 30, 30,...
## $ method <chr> "3PSM", "pcurve.evidence", "PEESE.lm", "PET.lm", "PE...
## $ TypeI <dbl> 0.087, 0.939, 0.029, 0.013, 0.013, 0.002, 0.852, 0.7...

glimpse(RR.H1)

## Observations: 120
## Variables: 5
## $ k <dbl> 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 30, 30, 30, ...
## $ qrp.label <fctr> QRP = none, QRP = none, QRP = none, QRP = none, QRP...
## $ loop <fctr> 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 30, 30, 30,...
## $ method <chr> "3PSM", "pcurve.evidence", "PEESE.lm", "PET.lm", "PE...
## $ Power <dbl> 0.972, 1.000, 0.145, 0.060, 0.060, 0.060, 1.000, 1.0...

RR.wide <- inner_join(RR.H0, RR.H1)

```

```
## Joining, by = c("k", "qrp.label", "loop", "method")
```

```
glimpse(RR.wide)
```

```
## Observations: 120
```

```
## Variables: 7
```

```
## $ condition <int> 193, 193, 193, 193, 193, 193, 193, 193, 193, 19...
```

```
## $ k          <dbl> 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 30, 30, 30, ...
```

```
## $ qrp.label  <fctr> QRP = none, QRP = none, QRP = none, QRP = none, QRP...
```

```
## $ loop       <fctr> 10, 10, 10, 10, 10, 10, 10, 10, 10, 10, 30, 30, 30, ...
```

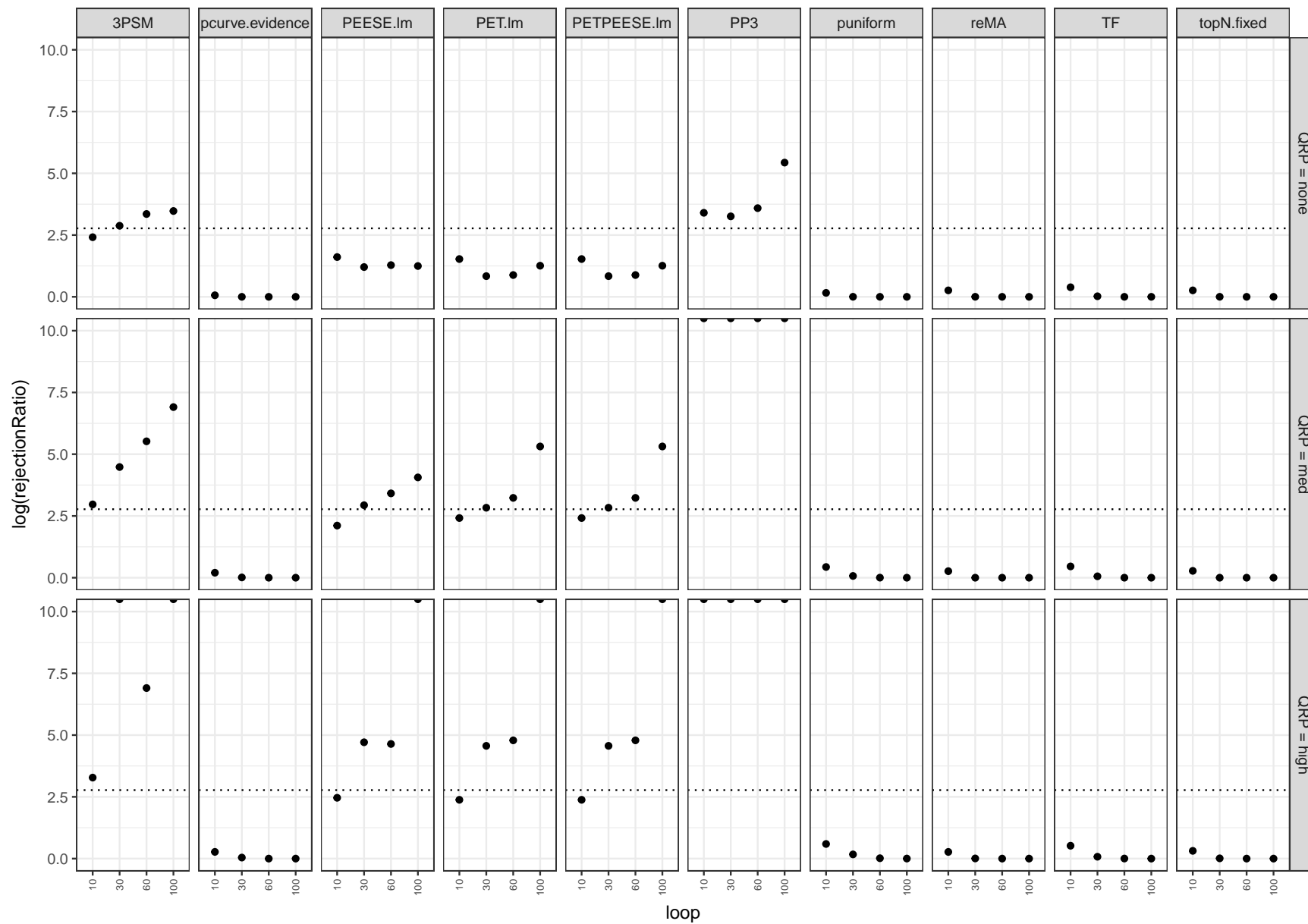
```
## $ method     <chr> "3PSM", "pcurve.evidence", "PEESE.lm", "PET.lm", "PE...
```

```
## $ TypeI       <dbl> 0.087, 0.939, 0.029, 0.013, 0.013, 0.002, 0.852, 0.7...
```

```
## $ Power       <dbl> 0.972, 1.000, 0.145, 0.060, 0.060, 0.060, 1.000, 1.0...
```

```
RR.wide$rejectionRatio <- RR.wide$Power/RR.wide$TypeI
```

Rejection Ratio (for selProp = 60%, H0 against delta=0.5)



```

# Rate of significant results in the wrong direction

wrongSig <- res.wide.red %>% select(1:8, b0_estimate, b0_p.value) %>%
  filter(!method %in% c("pcurve.evidence", "pcurve.lack"), delta < 0.5, !is.na(b0_estimate), !is.na(b0_p.value)) %>%
  group_by(condition, k, delta, qrpEnv, method) %>%
  summarise(
    wrongSig = sum((b0_estimate < 0) & (b0_p.value < .05)) / n()
  ) %>%
  mutate(loop = paste0(k))

# order loop factor alphabetically
wrongSig$loop <- factor(wrongSig$loop, levels = mixedsort(unique(wrongSig$loop)))

```


Percentage of significant estimates in the *wrong* direction

