Group estimation precision using the French lexicon project dataset

Guillaume A. Rousselet

2019-01-16

Contents

Quantify group level measurement precision / estimation accuracy. We sample participants with replacement, then we sample trials with replacement from each selected participant. For consistency, the same measure of central tendency is used at the two levels of analysis (means of means, medians of medians).

```
# dependencies
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 3.4.4
```

```
library(tibble)
```

```
## Warning: package 'tibble' was built under R version 3.4.3
```

```
library(cowplot)
```

```
## Warning: package 'cowplot' was built under R version 3.4.2
```

```
library(beepr)
```

```
## Warning: package 'beepr' was built under R version 3.4.4
```

```
sessionInfo()
```

```
## R version 3.4.0 (2017-04-21)
## Platform: x86_64-apple-darwin15.6.0 (64-bit)
## Running under: macOS  10.14.2
##
## Matrix products: default
## BLAS: /Library/Frameworks/R.framework/Versions/3.4/Resources/lib/libRblas.0.dylib
## LAPACK: /Library/Frameworks/R.framework/Versions/3.4/Resources/lib/libRlapack.dylib
##
## locale:
## [1] en_GB.UTF-8/en_GB.UTF-8/en_GB.UTF-8/C/en_GB.UTF-8/en_GB.UTF-8
```

```
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
## [1] beepr_1.3     cowplot_0.9.1 tibble_1.4.2  ggplot2_3.0.0
##
## loaded via a namespace (and not attached):
##  [1] Rcpp_0.12.19     rstudioapi_0.8   bindr_0.1.1      knitr_1.17
##  [5] magrittr_1.5     tidyselect_0.2.4 munsell_0.4.3    colorspace_1.3-2
##  [9] R6_2.3.0         rlang_0.2.2      stringr_1.2.0    plyr_1.8.4
## [13] dplyr_0.7.6      tools_3.4.0      grid_3.4.0       gtable_0.2.0
## [17] audio_0.1-5.1    withr_2.1.0      htmltools_0.3.6  assertthat_0.2.0
## [21] yaml_2.1.15      lazyeval_0.2.1   rprojroot_1.2    digest_0.6.12
## [25] crayon_1.3.4     bindrcpp_0.2.2   purrr_0.2.5      glue_1.3.0
## [29] evaluate_0.10.1  rmarkdown_1.8    stringi_1.1.6    compiler_3.4.0
## [33] pillar_1.3.0     scales_0.5.0     backports_1.1.1  pkgconfig_2.0.2
```

```
# get data - tibble = `flp`
load("./data/french_lexicon_project_rt_data.RData")
# columns =
#1 = participant
#2 = rt
#3 = acc = accuracy 0/1
#4 = condition = word/non-word
```

Population values

At the group level, the population median is defined as the median of all the pairwise differences between participants' medians. Similarly, the population mean is defined as the mean of all the pairwise differences between participants' means.

Population mean

```
# get data: mean RT for every participant
meanres <- tapply(flp$rt, list(flp$participant, flp$condition), mean)
# population mean of the differences between means
pop.diff.m <- mean(meanres[,2] - meanres[,1]) # Non-Word - Word
```

Population median

```
# # get data: median RT for every participant
# medres <- tapply(flp$rt, list(flp$participant, flp$condition), median)
# # population median of the differences between medians
# pop.diff.md <- median(medres[,2] - medres[,1]) # Non-Word - Word

load('./data/sim_bias_size_participants.RData')
diff <- sort(pop.md.nw - pop.md.w)
pop.diff.md <- diff[round(length(diff)*0.5)]
```

Simulation

10,000 iterations.
200 trials per condition. For each condition, the mean and the median across all available trials (~1000) for each participant and across participants are used as population values to estimate measurement precision within certain bounds.

Parameters

```
n <- 200 # sample size in each condition
preseq <- c(seq(5, 20, 5), 30, 40, 50) # precision bounds
Npre <- length(preseq)
ptseq <- c(seq(10, 100 ,10), 150, 200, 300) # number of participants
Npt <- length(ptseq)
Nsim <- 10000
p.list <- unique(flp$participant)
nP <- length(p.list) # 959 participants
```

Generate data

Same participants are used with increasing sample sizes, so that results can be directly compared across sample sizes.

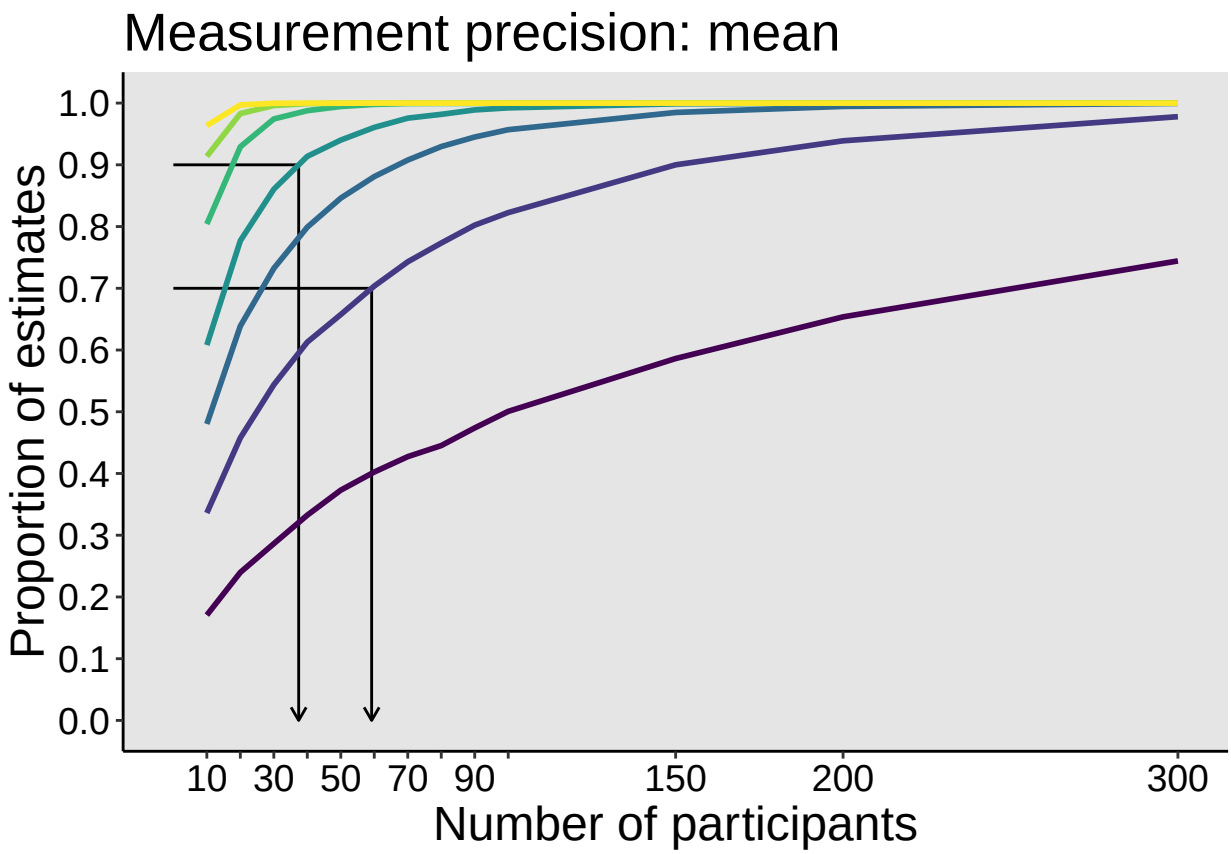
```
set.seed(666)

# declare result matrices
res.pre.m <- matrix(data = 0, nrow = Npre, ncol = Npt)
res.pre.md <- matrix(data = 0, nrow = Npre, ncol = Npt)

# sample participants with replacement once
# the same participants are used across sample sizes
all.bootsamples <- matrix(sample(nP, ptseq[Npt]*Nsim, replace = TRUE), ncol = Nsim)

# unique participants
todo <- unique(as.vector(bootsamples))
Ntodo <- length(todo)

gp.diff.m <- matrix(data = 0, nrow = ptseq[Npt], ncol = Nsim)
gp.diff.md <- matrix(data = 0, nrow = ptseq[Npt], ncol = Nsim)

for(P in 1:Ntodo){ # for each unique participant

  # occurences for that participant
  Nsamp <- sum(bootsamples==todo[P])

  # get data from one participant
  flp.w <- flp$rt[flp$participant==p.list[todo[P]] & flp$condition=="word"]
  flp.nw <- flp$rt[flp$participant==p.list[todo[P]] & flp$condition=="non-word"]

  # sample n trials with replacement x Nsamp times
  flp.w <- matrix(sample(flp.w, n*Nsamp, replace = TRUE), ncol = Nsamp)
  flp.nw <- matrix(sample(flp.nw, n*Nsamp, replace = TRUE), ncol = Nsamp)
```

```
  gp.diff.m[bootsamples==todo[P]] <- apply(flp.nw, 2, mean) - apply(flp.w, 2, mean)
  gp.diff.md[bootsamples==todo[P]] <- apply(flp.nw, 2, median) - apply(flp.w, 2, median)

}

beep(5)

for(iter.pt in 1:Npt){ # number of participants

  print(paste0("Sample size ",iter.pt," out of ",Npt,"..."))

    diff.m <- apply(gp.diff.m[1:ptseq[iter.pt],], 2, mean)
    diff.md <- apply(gp.diff.md[1:ptseq[iter.pt],], 2, median)

  # Probability of getting group estimate at most x ms from population value
  for(iter.p in 1:Npre){
    res.pre.m[iter.p, iter.pt] <- mean( abs(diff.m - pop.diff.m) <= (preseq[iter.p]) )
    res.pre.md[iter.p, iter.pt] <- mean( abs(diff.md - pop.diff.md) <= (preseq[iter.p]) )
  }

    beep(2)

} # number of participants

save(
  res.pre.m,
  res.pre.md,
  preseq, # precision bounds
  Npre,
  ptseq, # number of participants
  Npt,
  file=('./data/sim_precision.RData'))

beep(8)
```

Precision results: mean

```
load('./data/sim_precision.RData')
df <- tibble(`Proportion` = as.vector(res.pre.m),
             `Precision` = rep(preseq, Npt),
             `N` = rep(ptseq, each = Npre))

df$Precision <- as.character(df$Precision)
df$Precision <- factor(df$Precision, levels=unique(df$Precision))

xlabels <- c("10","","30","","50","","70","","90","","150","200","300")

# data frame to plot segments ========================
# number of participants needed so that in 70% of experiments the group estimate is no more than preseq
tmp.pos <- approx(y=ptseq,x=res.pre.m[2,],xout=0.70)$y
df.seg1 <- tibble(x=0, xend=tmp.pos,
```

```
                  y=0.7, yend=0.7)
df.seg2 <- tibble(x=tmp.pos, xend=tmp.pos,
                  y=0.7, yend=0)
# number of participants needed so that in 90% of experiments the group estimate is no more than preseq
tmp.pos <- approx(y=ptseq,x=res.pre.m[4,],xout=0.90)$y
df.seg3 <- tibble(x=0, xend=tmp.pos,
                  y=0.9, yend=0.9)
df.seg4 <- tibble(x=tmp.pos, xend=tmp.pos,
                  y=0.9, yend=0)

# make plot
p <- ggplot(df, aes(x=N, y=Proportion)) + theme_classic() +
  # geom_abline(intercept=0.7, slope=0, colour="grey20") +
  geom_segment(data = df.seg1, aes(x=x, y=y, xend=xend, yend=yend)) +
  geom_segment(data = df.seg2, aes(x=x, y=y, xend=xend, yend=yend),
               arrow = arrow(length = unit(0.2, "cm"))) +
  geom_segment(data = df.seg3, aes(x=x, y=y, xend=xend, yend=yend)) +
  geom_segment(data = df.seg4, aes(x=x, y=y, xend=xend, yend=yend),
               arrow = arrow(length = unit(0.2, "cm"))) +
  geom_line(aes(colour = Precision), size = 1) +
  scale_color_viridis_d() +
  scale_x_continuous(breaks=ptseq, labels = xlabels) +
  scale_y_continuous(breaks=seq(0, 1, 0.1)) +
  coord_cartesian(ylim=c(0, 1)) +
  theme(plot.title = element_text(size=20),
        axis.title.x = element_text(size = 18),
        axis.text = element_text(size = 14, colour="black"),
        axis.title.y = element_text(size = 18),
        legend.key.width = unit(1.5,"cm"),
        legend.position = "none",#c(0.8, 0.3),
        legend.text=element_text(size = 16),
        legend.title=element_text(size = 18),
        panel.background = element_rect(fill="grey90")) +
  labs(x = "Number of participants", y = "Proportion of estimates") +
  guides(colour = guide_legend(override.aes = list(size=3), # make thicker legend lines
    title = "Precision \n(within +/- ms)")) + # change legend title
  ggtitle("Measurement precision: mean")
p
```




```
p.m <- p
```

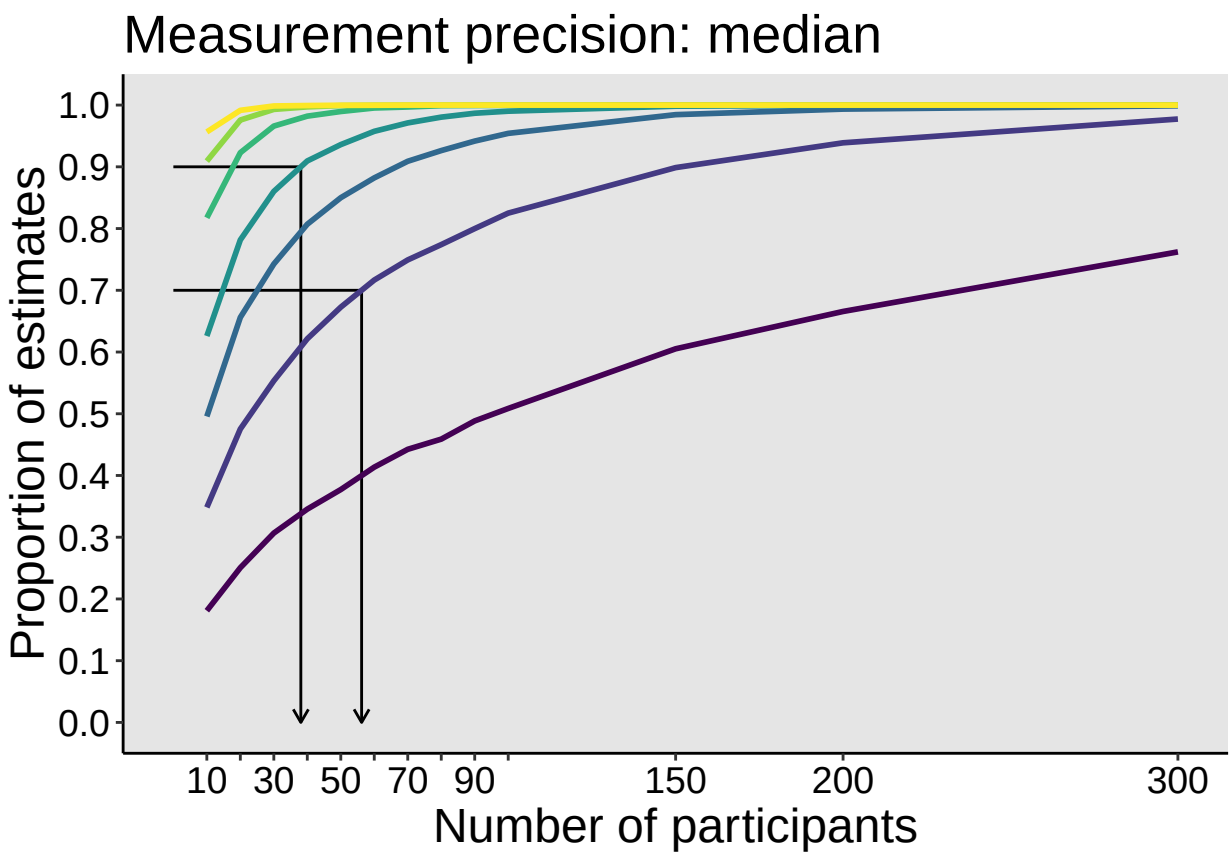
So that in 70% of experiments the group estimate is no more than 10 ms from the population value, we need to test at least 59 participants.

So that in 90% of experiments the group estimate is no more than 20 ms from the population value, we need to test at least 37 participants.

Precision results: median

```
load('./data/sim_precision.RData')
df <- tibble(`Proportion` = as.vector(res.pre.md),
             `Precision` = rep(preseq, Npt),
             `N` = rep(ptseq, each = Npre))

df$Precision <- as.character(df$Precision)
df$Precision <- factor(df$Precision, levels=unique(df$Precision))

# data frame to plot segments ========================
# number of participants needed so that in 70% of experiments the group estimate is no more than preseq
tmp.pos <- approx(y=ptseq,x=res.pre.md[2,],xout=0.70)$y
df.seg1 <- tibble(x=0, xend=tmp.pos,
                  y=0.7, yend=0.7)
df.seg2 <- tibble(x=tmp.pos, xend=tmp.pos,
                  y=0.7, yend=0)
# number of participants needed so that in 90% of experiments the group estimate is no more than preseq
```

```r
tmp.pos <- approx(y=ptseq,x=res.pre.md[4,],xout=0.90)$y
df.seg3 <- tibble(x=0, xend=tmp.pos,
                  y=0.9, yend=0.9)
df.seg4 <- tibble(x=tmp.pos, xend=tmp.pos,
                  y=0.9, yend=0)

# make plot
p <- ggplot(df, aes(x=N, y=Proportion)) + theme_classic() +
  # geom_abline(intercept=0.7, slope=0, colour="grey20") +
  geom_segment(data = df.seg1, aes(x=x, y=y, xend=xend, yend=yend)) +
  geom_segment(data = df.seg2, aes(x=x, y=y, xend=xend, yend=yend),
               arrow = arrow(length = unit(0.2, "cm"))) +
  geom_segment(data = df.seg3, aes(x=x, y=y, xend=xend, yend=yend)) +
  geom_segment(data = df.seg4, aes(x=x, y=y, xend=xend, yend=yend),
               arrow = arrow(length = unit(0.2, "cm"))) +
  geom_line(aes(colour = Precision), size = 1) +
  scale_color_viridis_d() +
  scale_x_continuous(breaks=ptseq, labels = xlabels) +
  scale_y_continuous(breaks=seq(0, 1, 0.1)) +
  coord_cartesian(ylim=c(0, 1)) +
  theme(plot.title = element_text(size=20),
        axis.title.x = element_text(size = 18),
        axis.text = element_text(size = 14, colour="black"),
        axis.title.y = element_text(size = 18),
        legend.key.width = unit(1.5,"cm"),
        legend.position = "none",
        legend.text=element_text(size = 16),
        legend.title=element_text(size = 18),
        panel.background = element_rect(fill="grey90")) +
  labs(x = "Number of participants", y = "Proportion of estimates") +
  guides(colour = guide_legend(override.aes = list(size=3), # make thicker legend lines
    title = "Precision \n(within +/- ms)")) + # change legend title
  ggtitle("Measurement precision: median")
p
```




```
p.md <- p
```

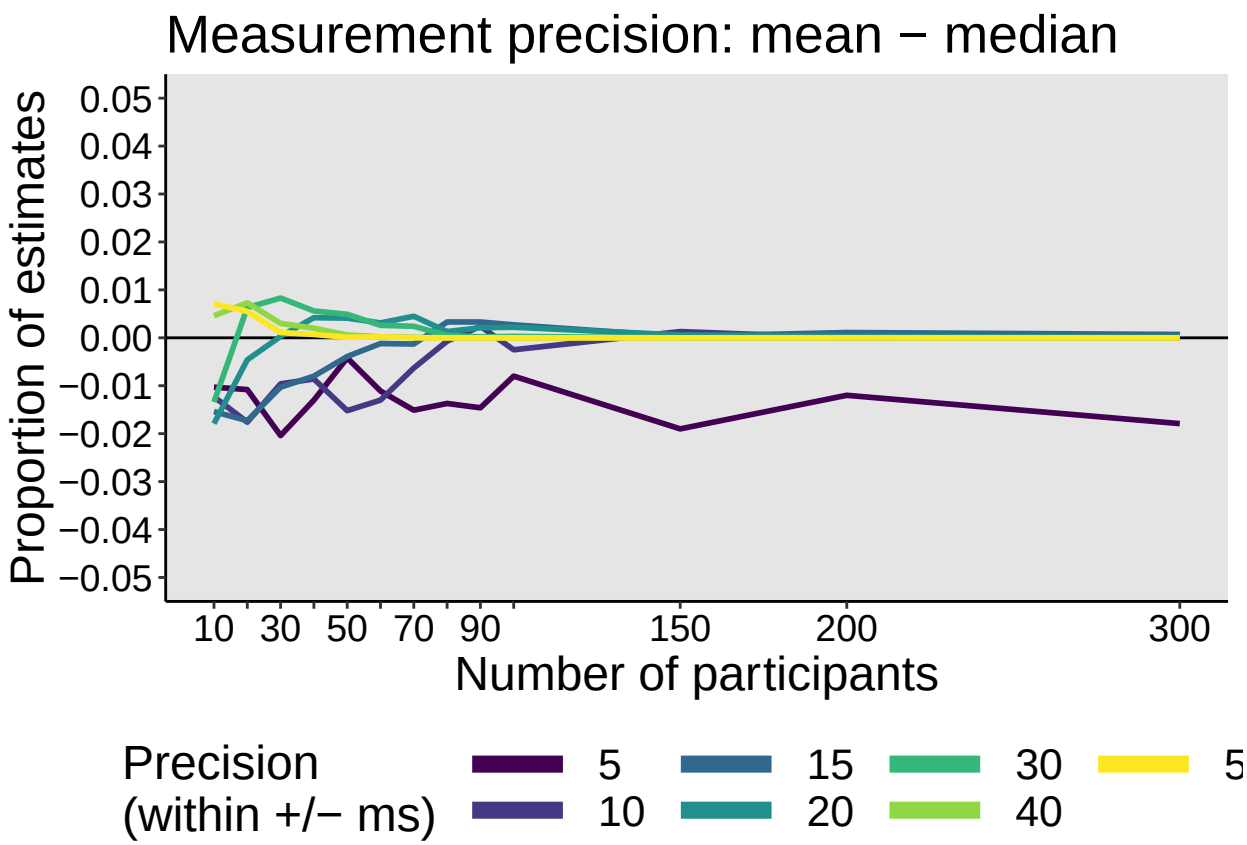
So that in 70% of experiments the group estimate is no more than 10 ms from the population value, we need to test at least 56 participants.

So that in 90% of experiments the group estimate is no more than 20 ms from the population value, we need to test at least 38 participants.

Precision results: mean - median

```
load('./data/sim_precision.RData')
df <- tibble(`Proportion` = as.vector(res.pre.m - res.pre.md),
             `Precision` = rep(preseq, Npt),
             `N` = rep(ptseq, each = Npre))

df$Precision <- as.character(df$Precision)
df$Precision <- factor(df$Precision, levels=unique(df$Precision))

# make plot
p <- ggplot(df, aes(x=N, y=Proportion)) + theme_classic() +
  geom_abline(intercept = 0, slope = 0) +
  geom_line(aes(colour = Precision), size = 1) +
  scale_color_viridis_d() +
  scale_x_continuous(breaks=ptseq, labels = xlabels) +
  scale_y_continuous(breaks=seq(-0.1, 0.1, 0.01)) +
  coord_cartesian(ylim=c(-0.05, 0.05)) +
```

```
  theme(plot.title = element_text(size=20),
        axis.title.x = element_text(size = 18),
        axis.text = element_text(size = 14, colour="black"),
        axis.title.y = element_text(size = 18),
        legend.key.width = unit(1.5,"cm"),
        legend.position = "bottom",
        legend.text=element_text(size = 16),
        legend.title=element_text(size = 18),
        panel.background = element_rect(fill="grey90")) +
  labs(x = "Number of participants", y = "Proportion of estimates") +
  guides(colour = guide_legend(override.aes = list(size=3), # make thicker legend lines
    title = "Precision \n(within +/- ms)")) + # change legend title
  ggtitle("Measurement precision: mean - median")
p
```




```
p.diff <- p
```

Summary figure

```
# combine panels into one figure
cowplot::plot_grid(p.m, p.md, p.diff,
                           labels = c("A", "B", "C"),
                           ncol = 1,
                           nrow = 3,
                           # rel_widths = c(1, 1, 1),
                           label_size = 20,
```

```
                          hjust = -1.5,
                          scale=.95,
                          align = "v")
# save figure
ggsave(filename='./figures/figure_flp_sim_precision.pdf',width=8,height=15)
```