Median bias: application to the French lexicon project dataset

Guillaume A. Rousselet

2019-04-18

Contents

Quantify participant level sample bias due to differences in sample sizes.

```
# dependencies
library(ggplot2)
library(tibble)
library(cowplot)
library(HDInterval)
```

```
sessionInfo()
```

```
## R version 3.5.2 (2018-12-20)
## Platform: x86_64-apple-darwin15.6.0 (64-bit)
## Running under: macOS Mojave 10.14.4
##
## Matrix products: default
## BLAS: /Library/Frameworks/R.framework/Versions/3.5/Resources/lib/libRblas.0.dylib
## LAPACK: /Library/Frameworks/R.framework/Versions/3.5/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] HDInterval_0.2.0 cowplot_0.9.4    tibble_2.0.1     ggplot2_3.1.0
##
## loaded via a namespace (and not attached):
##  [1] Rcpp_1.0.0       knitr_1.21       magrittr_1.5     tidyselect_0.2.5
##  [5] munsell_0.5.0    colorspace_1.4-0 R6_2.4.0         rlang_0.3.1
##  [9] stringr_1.4.0    plyr_1.8.4       dplyr_0.8.0.1    tools_3.5.2
## [13] grid_3.5.2       gtable_0.2.0     xfun_0.4         withr_2.1.2
## [17] htmltools_0.3.6  assertthat_0.2.0 yaml_2.2.0       lazyeval_0.2.1
## [21] digest_0.6.18    crayon_1.3.4     purrr_0.3.0      glue_1.3.0
```

```
## [25] evaluate_0.13    rmarkdown_1.11   stringi_1.3.1    compiler_3.5.2
## [29] pillar_1.3.1     scales_1.0.0     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
```

Simulation

2,000 iterations.
Percentile boostrap bias correction using 200 bootstrap samples.
The Non-word condition (least skewed) has 200 trials and the Word condition varies from 10 to 200 trials in 10 trial increments.
For each participant and condition, the mean and the median across all available trials (~1000) are used as population values to compute bias.

```
set.seed(21)
p.list <- unique(flp$participant)
nP <- length(p.list) # 959 participants

# parameters
nsim <- 2000 # estimate bias using nsim resamples
nboot <- 200 # correct bias using nboot resamples
nmax <- 200
nvec <- seq(10, nmax, 10)

# declare matrices of results ================
# get population values to subtract from downsampled estimates,
# so that, on average, bias of the difference should be zero
pop.md.w <- vector(mode="numeric", nP)
pop.md.nw <- vector(mode="numeric", nP)
pop.m.w <- vector(mode="numeric", nP)
pop.m.nw <- vector(mode="numeric", nP)
sim.md.diff <- matrix(NA, nrow=nP, ncol=length(nvec))
sim.m.diff <- matrix(NA, nrow=nP, ncol=length(nvec))
sim.md.diff.md <- matrix(NA, nrow=nP, ncol=length(nvec))
sim.m.diff.md <- matrix(NA, nrow=nP, ncol=length(nvec))
bc.md.diff <- matrix(NA, nrow=nP, ncol=length(nvec))

for(P in 1:nP){
  if(P %% 100 ==0){
  print(paste0("Participant ",P," out of ",nP,"..."))
  }

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

  # define population values -----------------
```

```
  pop.m.w[P] <- mean(flp.w)
  pop.m.nw[P] <- mean(flp.nw)
  # pop.m.diff[P] <- pop.m.nw[P] - pop.m.w[P]

  pop.md.w[P] <- sort(flp.w)[round(length(flp.w)*0.5)] # median(flp.w)
  pop.md.nw[P] <- sort(flp.nw)[round(length(flp.nw)*0.5)] # median(flp.nw)
  # pop.md.diff[P] <- pop.md.nw[P] - pop.md.w[P]

  # Non-word condition has nmax trials (least skewed)
  mc.nw <- matrix(sample(flp.nw, size=nmax*nsim, replace = TRUE), nrow=nsim)
  # centred distributions of measures of central tendency
  md.mc.nw <- apply(mc.nw, 1, median) - pop.md.nw[P]
  m.mc.nw <- apply(mc.nw, 1, mean) - pop.m.nw[P]

  # Bias correct non-word here
  tmp.bc.md.nw <- vector(mode="numeric", length=nsim)
  for(iter in 1:nsim){
    boot.md <- apply(matrix(sample(mc.nw[iter,], nmax*nboot, replace=TRUE),nrow=nboot),1,median)
    tmp.bc.md.nw[iter] <- 2*median(mc.nw[iter,]) - mean(boot.md) # BC
  }
  tmp.bc.md.nw <- tmp.bc.md.nw - pop.md.nw[P]

  # ========================================================================
  for(iter.n in 1:length(nvec)){ # Word condition: different sample sizes
    # Word
    mc.w <- matrix(sample(flp.w, size=nvec[iter.n]*nsim, replace = TRUE), nrow=nsim)

    md.mc.w <- apply(mc.w, 1, median) - pop.md.w[P]
    m.mc.w <- apply(mc.w, 1, mean) - pop.m.w[P]

    # Differences
    sim.md.diff[P, iter.n] <- mean(md.mc.nw - md.mc.w) # mean bias
    sim.m.diff[P, iter.n] <- mean(m.mc.nw - m.mc.w)

    sim.md.diff.md[P, iter.n] <- median(md.mc.nw - md.mc.w) # median bias
    sim.m.diff.md[P, iter.n] <- median(m.mc.nw - m.mc.w)

    # ======================================
    # Bias correction
    # Word -------------
    tmp.bc.md.w <- vector(mode="numeric", length=nsim)
    for(iter in 1:nsim){
       boot.md <- apply(matrix(sample(mc.w[iter,], nvec[iter.n]*nboot, replace=TRUE),nrow=nboot),1,media
     tmp.bc.md.w[iter] <- 2*median(mc.w[iter,]) - mean(boot.md) # BC
    }
    tmp.bc.md.w <- tmp.bc.md.w - pop.md.w[P]

    # Difference -------------
    bc.md.diff[P,iter.n] <- mean(tmp.bc.md.nw - tmp.bc.md.w) # mean of BC estimates
  } # sample sizes
} # participants
save(
  sim.m.diff,
```

```
  sim.md.diff,
  sim.m.diff.md,
  sim.md.diff.md,
  bc.md.diff,
  pop.md.w,
  pop.md.nw,
  pop.m.w,
  pop.m.nw,
  nvec,
  nboot,
  nsim,
  nP,
  file=('./data/sim_bias_size_participants.RData'))
```

Load results. No need to compute bias: each distribution was mean or median centred, so that the difference should on average be zero.

```
load('./data/sim_bias_size_participants.RData')
mmd <- apply(sim.md.diff, 2, mean)
bc.mmd <- apply(bc.md.diff, 2, mean)
mm <- apply(sim.m.diff, 2, mean)
# median bias
md.mmd <- apply(sim.md.diff.md, 2, mean)
md.mm <- apply(sim.m.diff.md, 2, mean)

# define x-ticks
n.seq <- seq(10, 200, 10)
n.seq2 <- c("10", "",  "30", "", "50", "", "70", "", "90", "", "110", "", "130", "",
            "150", "", "170", "", "190", "")
```

Illustrate bias

Bias as a function of sample size: all participants superimposed + group mean. Difference = non-word - word.

Median

Because the Word condition is overestimated (positive bias) with small sample sizes, the difference between Non-Word and Word is underestimated, it has a negative bias. The coloured thin lines show results for individual participants. The thick black line shows the mean across participants. For the smallest sample size, the average bias across participants is -10.9 ms. The variability across participants is large, with an HDI of [-17.1, -2.6] ms. For n=20, the average bias is -4.8 ms, for n=60 it is -1 ms.

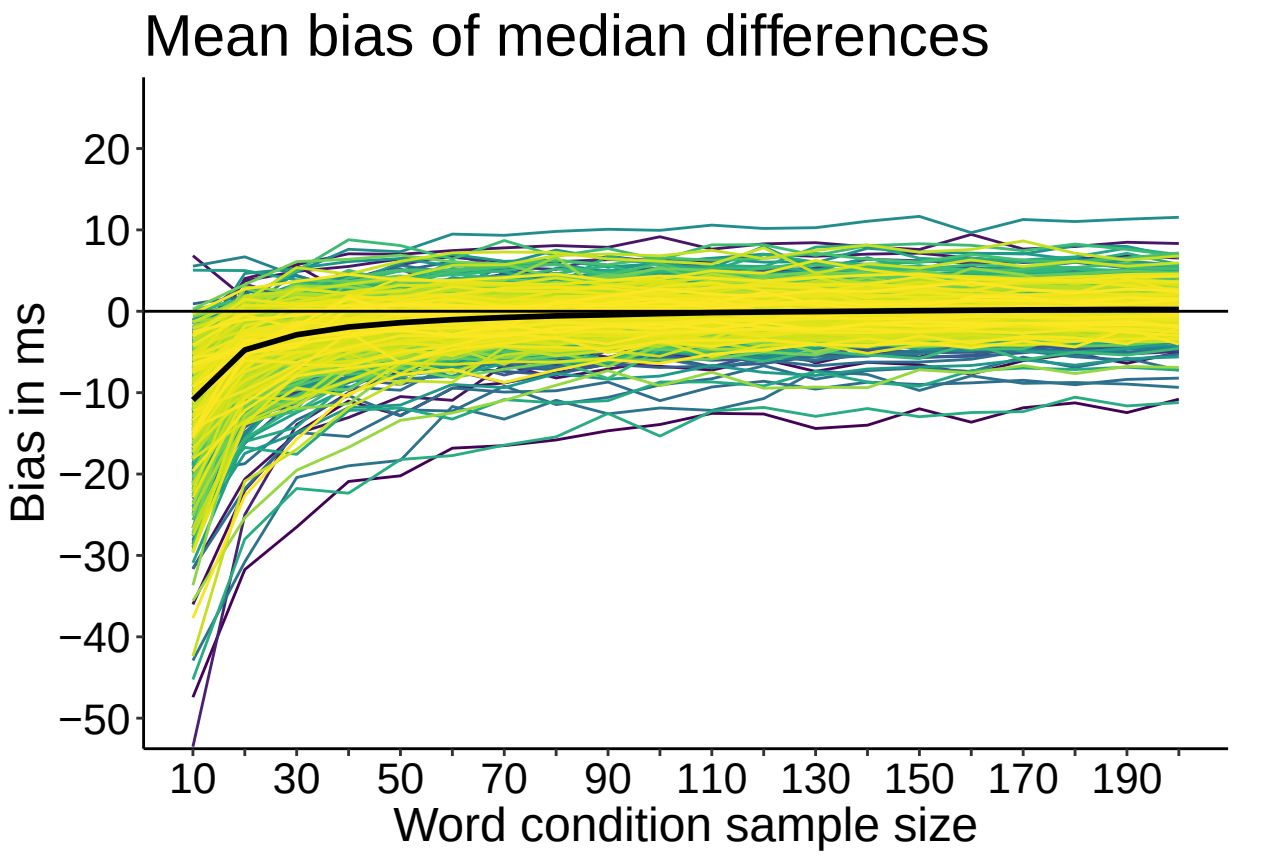
```
# Median
df <- tibble(`Participant` = factor(rep(seq(1:nP), length(nvec))),
             `Size` = rep(nvec, each=nP),
             `Bias` = as.vector(sim.md.diff))

df.sum <- tibble(`Size` = nvec,
                 `MMD`= mmd)

p <- ggplot(data=df.sum, aes(x=Size)) + theme_classic() +
```

```
  geom_line(data=df, aes(x=Size, y=Bias, colour = Participant)) +
  coord_cartesian(ylim = c(-50, 25)) +
  geom_line(data = df.sum, aes(y=MMD), colour="black", size=1) +
  geom_hline(yintercept = 0, colour="black") +
  scale_color_viridis_d() +
    theme(plot.title = element_text(size=22),
        axis.title.x = element_text(size = 18),
        axis.text = element_text(size = 16, colour = "black"),
        axis.title.y = element_text(size = 18),
        legend.text = element_text(size = 16),
        legend.title = element_text(size = 18),
        legend.key.width = unit(1.5,"cm"),
        legend.position = "none",#c(0.75,0.8),
        strip.text.y = element_text(size = 18, face = "bold", angle = 0)) +
    scale_x_continuous(breaks = n.seq, labels = n.seq2) +
    scale_y_continuous(breaks = seq(-50, 30, 10)) +
    labs(x = "Word condition sample size", y = "Bias in ms") +
  ggtitle(paste0("Mean bias of median differences"))
```

```
p
```




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

Median with bias correction

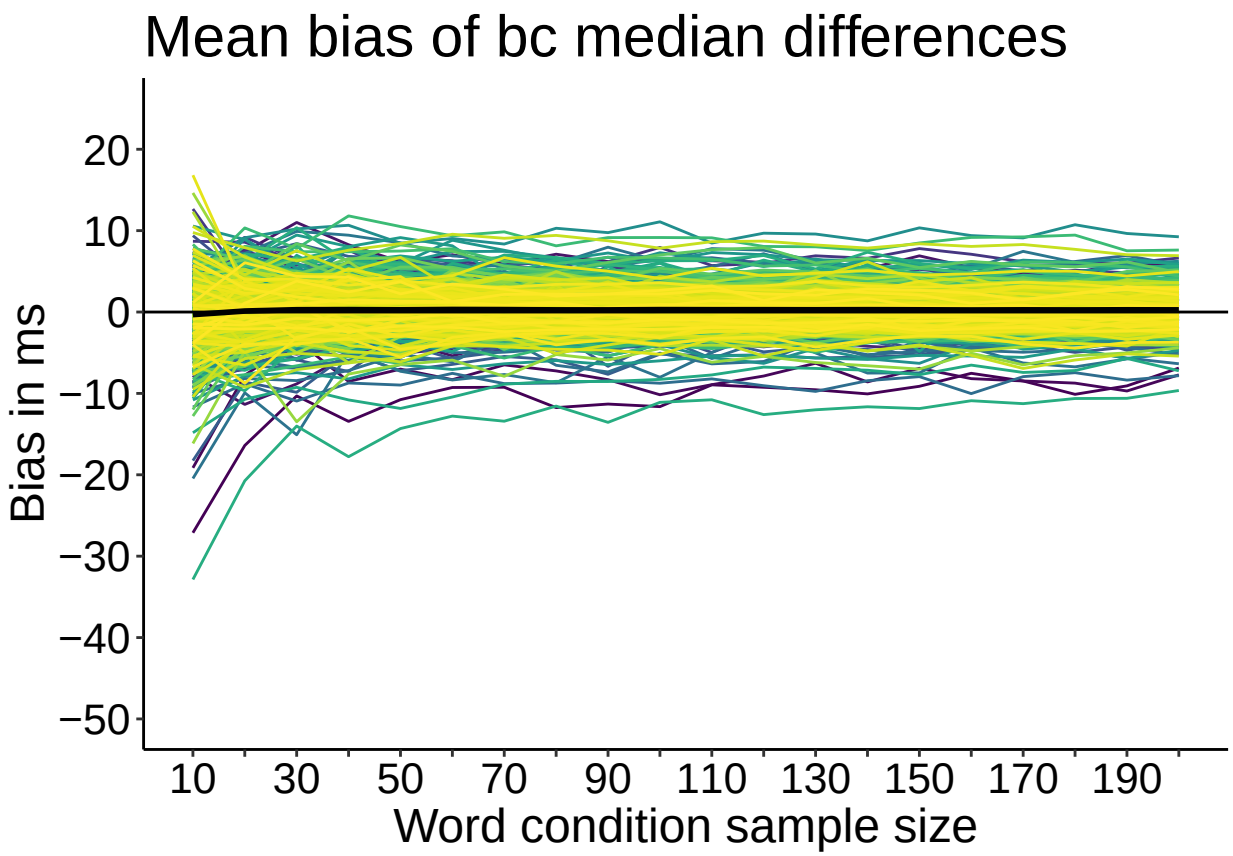
For the smallest sample size, the average bias across participants is 0 ms.

```r
# Median with bias correction
df <- tibble(`Participant` = factor(rep(seq(1:nP), length(nvec))),
             `Size` = rep(nvec, each=nP),
             `Bias` = as.vector(bc.md.diff))

df.sum <- tibble(`Size` = nvec,
                 `MMD`= bc.mmd)

p <- ggplot(data=df.sum, aes(x=Size)) + theme_classic() +
  geom_line(data=df, aes(x=Size, y=Bias, colour = Participant)) +
  coord_cartesian(ylim = c(-50, 25)) +
  geom_line(data = df.sum, aes(y=MMD), colour="black", size=1) +
  geom_hline(yintercept = 0, colour="black") +
    scale_color_viridis_d() +
    theme(plot.title = element_text(size=22),
        axis.title.x = element_text(size = 18),
        axis.text = element_text(size = 16, colour = "black"),
        axis.title.y = element_text(size = 18),
        legend.text = element_text(size = 16),
        legend.title = element_text(size = 18),
        legend.key.width = unit(1.5,"cm"),
        legend.position = "none",#c(0.75,0.8),
        strip.text.y = element_text(size = 18, face = "bold", angle = 0)) +
        scale_x_continuous(breaks = n.seq, labels = n.seq2) +
        scale_y_continuous(breaks = seq(-50, 30, 10)) +
  labs(x = "Word condition sample size", y = "Bias in ms") +
  ggtitle(paste0("Mean bias of bc median differences"))

p
```




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

Median: median bias

For the smallest sample size, the average bias across participants is -1.9 ms. For n=20, the bias is -0.3 ms.

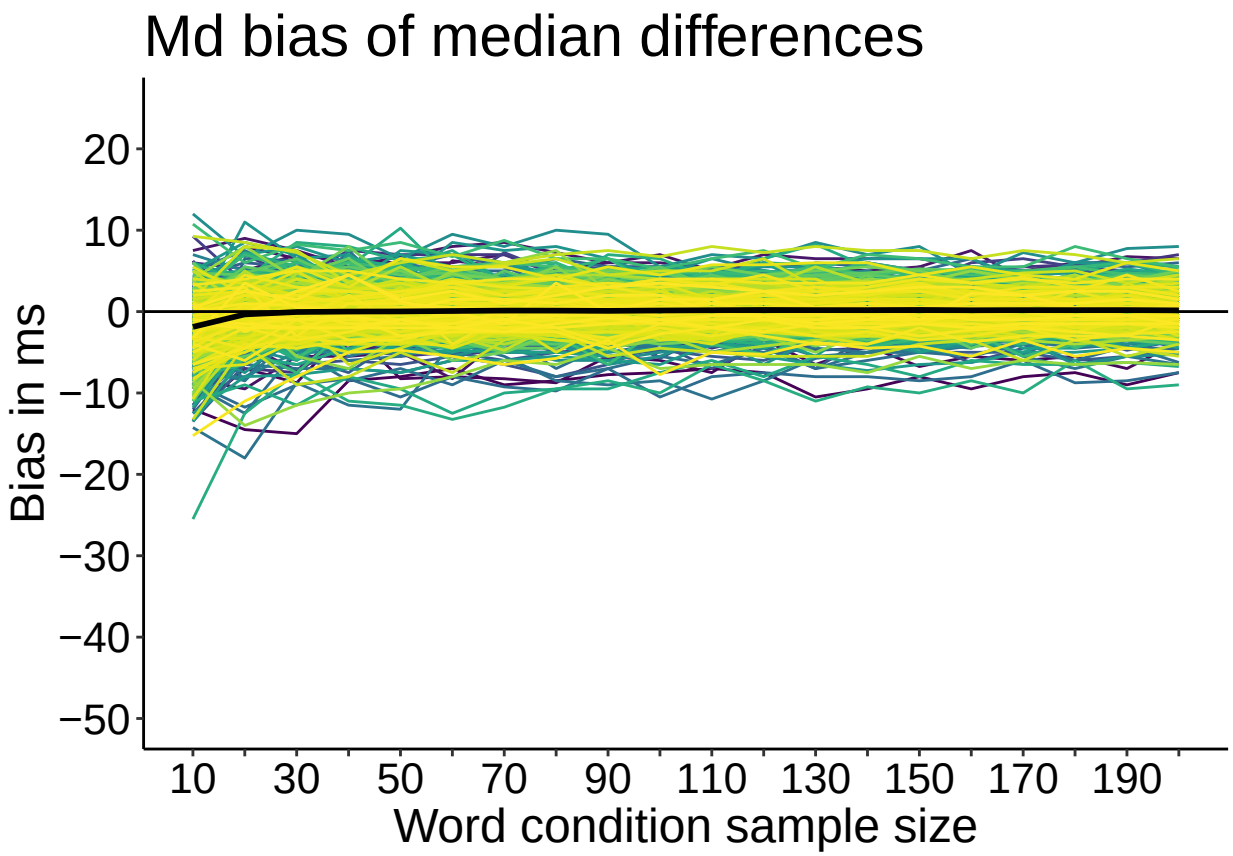
```
# Median: median bias
df <- tibble(`Participant` = factor(rep(seq(1:nP), length(nvec))),
             `Size` = rep(nvec, each=nP),
             `Bias` = as.vector(sim.md.diff.md))

df.sum <- tibble(`Size` = nvec,
                 `MMD`= md.mmd)

p <- ggplot(data=df.sum, aes(x=Size)) + theme_classic() +
  geom_line(data=df, aes(x=Size, y=Bias, colour = Participant)) +
  coord_cartesian(ylim = c(-50, 25)) +
  geom_line(data = df.sum, aes(y=MMD), colour="black", size=1) +
  geom_hline(yintercept = 0, colour="black") +
    scale_color_viridis_d() +
    theme(plot.title = element_text(size=22),
        axis.title.x = element_text(size = 18),
        axis.text = element_text(size = 16, colour = "black"),
        axis.title.y = element_text(size = 18),
        legend.text = element_text(size = 16),
        legend.title = element_text(size = 18),
```

```
        legend.key.width = unit(1.5,"cm"),
        legend.position = "none",#c(0.75,0.8),
        strip.text.y = element_text(size = 18, face = "bold", angle = 0)) +
    scale_x_continuous(breaks = n.seq, labels = n.seq2) +
    scale_y_continuous(breaks = seq(-50, 30, 10)) +
    labs(x = "Word condition sample size", y = "Bias in ms") +
  ggtitle(paste0("Md bias of median differences"))

p
```




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

Mean

For the smallest sample size, the average bias across participants is -0.1 ms.

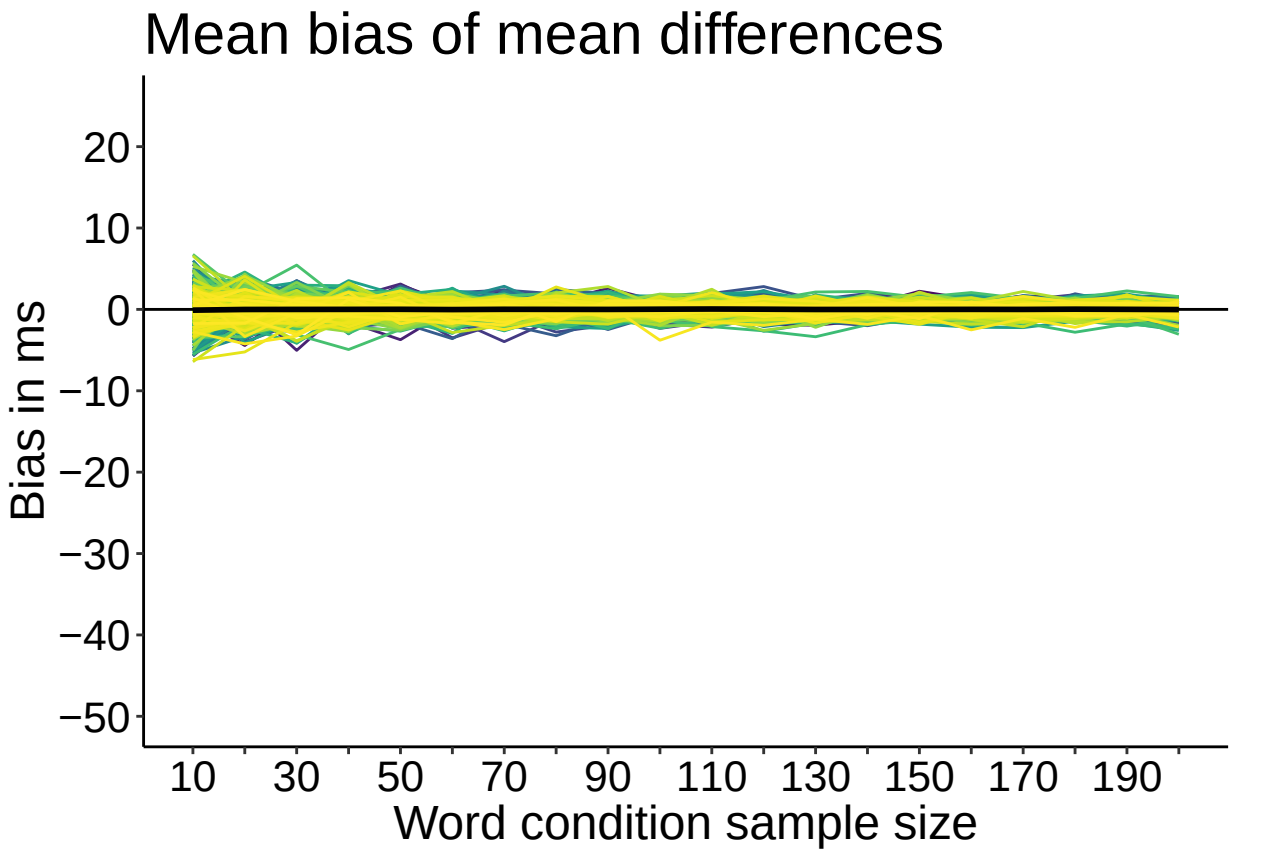
```
# Mean
df <- tibble(`Participant` = factor(rep(seq(1:nP), length(nvec))),
             `Size` = rep(nvec, each=nP),
             `Bias` = as.vector(sim.m.diff))

df.sum <- tibble(`Size` = nvec,
                 `MMD`= mm)

p <- ggplot(data=df.sum, aes(x=Size)) + theme_classic() +
  geom_line(data=df, aes(x=Size, y=Bias, colour = Participant)) +
```

```
  coord_cartesian(ylim = c(-50, 25)) +
  geom_line(data = df.sum, aes(y=MMD), colour="black", size=1) +
  geom_hline(yintercept = 0, colour="black") +
    scale_color_viridis_d() +
    theme(plot.title = element_text(size=22),
        axis.title.x = element_text(size = 18),
        axis.text = element_text(size = 16, colour = "black"),
        axis.title.y = element_text(size = 18),
        legend.text = element_text(size = 16),
        legend.title = element_text(size = 18),
        legend.key.width = unit(1.5,"cm"),
        legend.position = "none",#c(0.75,0.8),
        strip.text.y = element_text(size = 18, face = "bold", angle = 0)) +
    scale_x_continuous(breaks = n.seq, labels = n.seq2) +
    scale_y_continuous(breaks = seq(-50, 30, 10)) +
    labs(x = "Word condition sample size", y = "Bias in ms") +
  ggtitle(paste0("Mean bias of mean differences"))

p
```




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

Mean: median bias

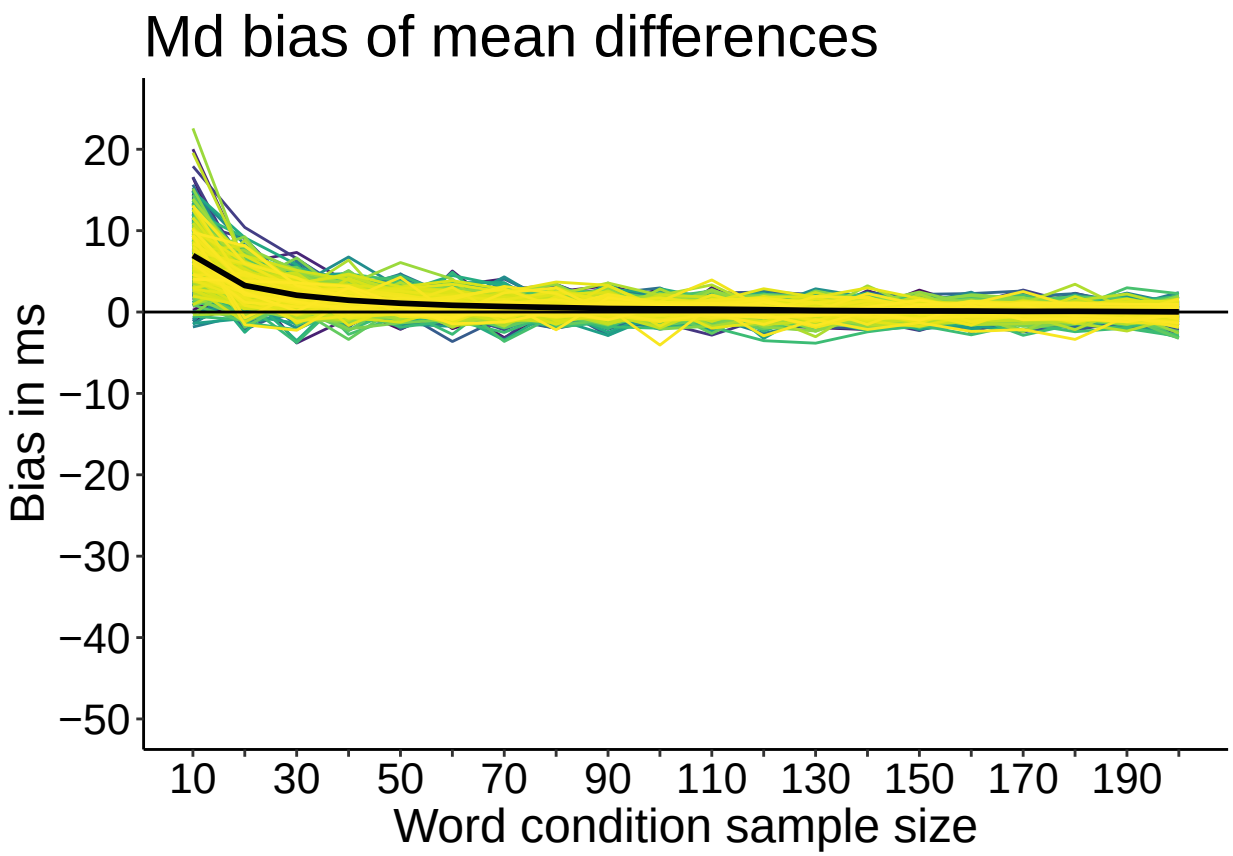
For the smallest sample size, the average bias across participants is 6.9 ms.

```r
# Mean: median bias
df <- tibble(`Participant` = factor(rep(seq(1:nP), length(nvec))),
             `Size` = rep(nvec, each=nP),
             `Bias` = as.vector(sim.m.diff.md))

df.sum <- tibble(`Size` = nvec,
                 `MMD`= md.mm)

p <- ggplot(data=df.sum, aes(x=Size)) + theme_classic() +
  geom_line(data=df, aes(x=Size, y=Bias, colour = Participant)) +
  coord_cartesian(ylim = c(-50, 25)) +
  geom_line(data = df.sum, aes(y=MMD), colour="black", size=1) +
  geom_hline(yintercept = 0, colour="black") +
    scale_color_viridis_d() +
    theme(plot.title = element_text(size=22),
        axis.title.x = element_text(size = 18),
        axis.text = element_text(size = 16, colour = "black"),
        axis.title.y = element_text(size = 18),
        legend.text = element_text(size = 16),
        legend.title = element_text(size = 18),
        legend.key.width = unit(1.5,"cm"),
        legend.position = "none",#c(0.75,0.8),
        strip.text.y = element_text(size = 18, face = "bold", angle = 0)) +
    scale_x_continuous(breaks = n.seq, labels = n.seq2) +
    scale_y_continuous(breaks = seq(-50, 30, 10)) +
    labs(x = "Word condition sample size", y = "Bias in ms") +
  ggtitle(paste0("Md bias of mean differences"))

p
```




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

Summary figure

```
# combine panels into one figure
cowplot::plot_grid(p.md, p.m, p.md.bc, NULL, p.md.md, p.m.md,
                           labels = c("A", "D", "B", "", "C", "E"),
                           ncol = 2,
                           nrow = 3,
                           # rel_widths = c(1, 1, 1),
                           label_size = 20,
                           hjust = -1.5,
                           scale=.95,
                           align = "h")
# save figure
ggsave(filename='./figures/figure_flp_sim.pdf',width=12,height=15)
```