What can we learn from 10,000 experiments?

Guillaume A. Rousselet

Contents

```r
rm(list=ls())
# dependencies
library(ggplot2)
library(tibble)
library(tidyr)
library(viridis)
source("akerd.txt")
source("HDIofMCMC.R")
```

Lexical decision data

Data from the French Lexicon Project. Click on "French Lexicon Project trial-level results with R scripts.zip". Data were saved using `getflprtdata.Rmd`.

Ferrand, L., New, B., Brysbaert, M., Keuleers, E., Bonin, P., Meot, A., Augustinova, M. & Pallier, C. (2010) The French Lexicon Project: lexical decision data for 38,840 French words and 38,840 pseudowords. Behav Res Methods, 42, 488-496.

```r
# get data - tibble = `flp`
load("./french_lexicon_project_rt_data.RData")
# columns =
#1 = participant
```

```
#2 = rt
#3 = acc = accuracy 0/1
#4 = condition = word/non-word
```

N = 967 participants.

Illustrate results for a few random participants

Word / non-word comparison:

```
set.seed(7)

# select participants
p.list <- unique(flp$participant)
sp <- p.list[sample(length(p.list), 3, replace=FALSE)]

for(iter in 1:length(sp)){
# make KDE
flp.w <- sort(flp$rt[flp$participant==sp[iter] & flp$condition=="word"])
flp.nw <- sort(flp$rt[flp$participant==sp[iter] & flp$condition=="non-word"])
a.flp.w <- akerd(flp.w, pyhat = TRUE, plotit = FALSE)
a.flp.nw <- akerd(flp.nw, pyhat = TRUE, plotit = FALSE)

# create data frame
df <- tibble(`x`=c(flp.w,flp.nw),
             `y`=c(a.flp.w,a.flp.nw),
             `Condition`=c(rep.int("Word",length(flp.w)),
                           rep.int("Non-word",length(flp.nw))))
# make plot
df$Condition <- as.character(df$Condition)
df$Condition <- factor(df$Condition, levels=unique(df$Condition))

# make plot
p <- ggplot(df, aes(x,y, group=Condition)) + theme_classic() +
  geom_line(aes(colour=Condition), size = 1.5) + # linetype=Condition,
  # scale_size_manual(values=c(1,0.5)) +
  # scale_linetype_manual(values=c("solid","solid")) +
  scale_color_manual(values=c("grey30", "#E69F00")) + #, "#56B4E9","black")) +
  scale_x_continuous(limits=c(0,2000), breaks=seq(0,2000,500), minor_breaks = waiver()) +
  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 = c(0.75,0.8),
        strip.text.y = element_text(size = 18, face = "bold", angle = 0)) +
        # legend.position = c(0.25,0.9)) +
  labs(x = "Reaction times", y = "Density") +
  ggtitle(paste0("Lexical decision: P",sp[iter]))
p
# save figure
```

```
ggsave(filename=paste0('./figure_flp_p',sp[iter],'.jpg'),width=10,height=7) #path=pathname
}
```

Illustrate group results

Number of trials per participant

```
# get data: median RT for every participant
ntrials <- tapply(flp$rt, list(flp$participant, flp$condition), length)
summary(ntrials)
```

```
##       word           non-word
##  Min.   : 996.0   Min.   : 997
##  1st Qu.: 999.0   1st Qu.:1000
##  Median : 999.0   Median :1001
##  Mean   : 999.5   Mean   :1001
##  3rd Qu.:1000.0   3rd Qu.:1001
##  Max.   :1000.0   Max.   :1001
```

Summary statistics for RT data

```
# get data: median RT for every participant
medres <- tapply(flp$rt, list(flp$participant, flp$condition), median)
summary(medres)
```

```
##       word           non-word
##  Min.   : 453.0   Min.   : 507.0
##  1st Qu.: 621.0   1st Qu.: 685.8
##  Median : 679.5   Median : 764.0
##  Mean   : 699.9   Mean   : 792.8
##  3rd Qu.: 763.8   3rd Qu.: 873.0
##  Max.   :1295.5   Max.   :1521.0
```

```
# get data: mean RT for every participant
meanres <- tapply(flp$rt, list(flp$participant, flp$condition), mean)
summary(meanres)
```

```
##       word           non-word
##  Min.   : 459.4   Min.   : 522.1
##  1st Qu.: 675.2   1st Qu.: 737.2
##  Median : 749.0   Median : 828.9
##  Mean   : 767.6   Mean   : 853.2
##  3rd Qu.: 847.0   3rd Qu.: 950.3
##  Max.   :1331.4   Max.   :1514.3
```

Save population results (all trials, all participants)

```
pop.m.w <- mean(meanres[,1])
pop.m.nw <- mean(meanres[,2])
pop.m.diff <- mean(meanres[,2] - meanres[,1])
pop.md.w <- median(medres[,1])
pop.md.nw <- median(medres[,2])
pop.md.diff <- median(medres[,2] - medres[,1])
```

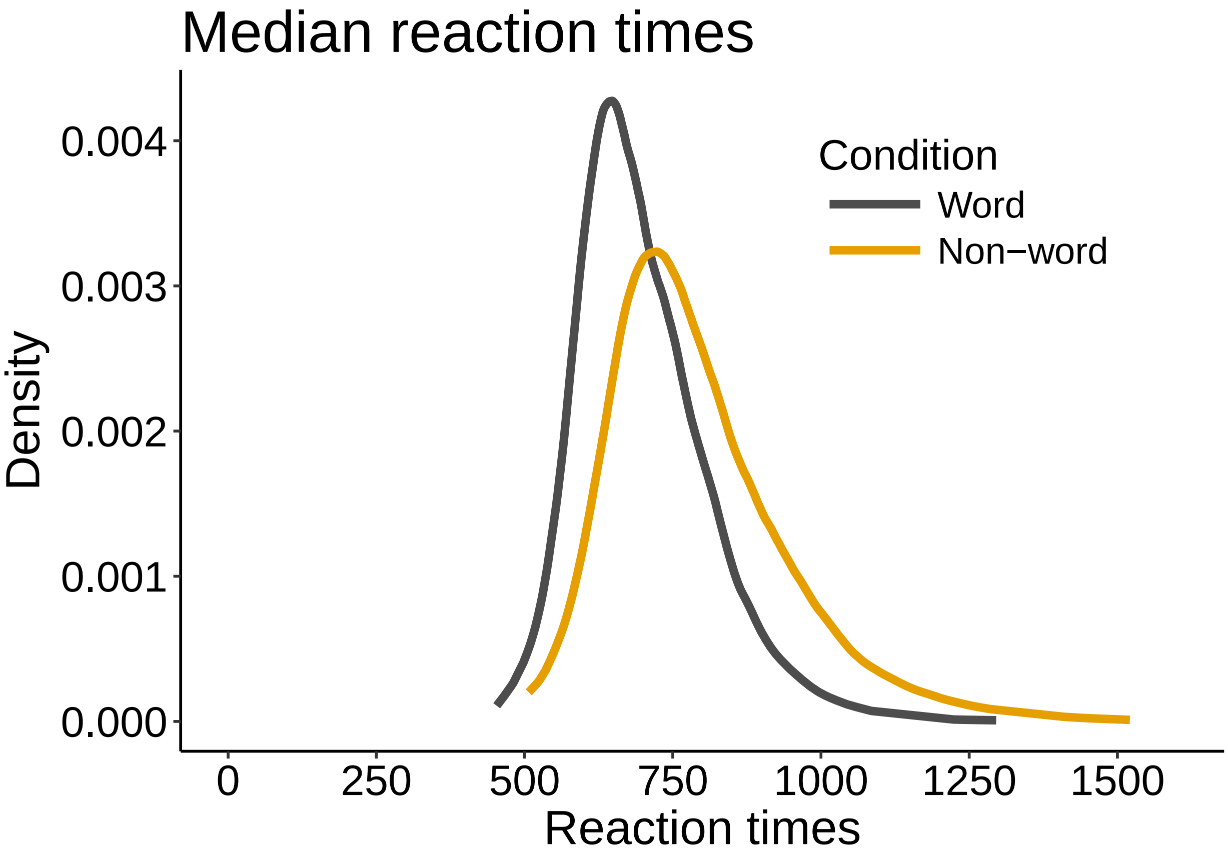
Word and non-word conditions

Distributions of median RT

```
# make KDE
flp.w <- sort(medres[,1])
flp.nw <- sort(medres[,2])
a.flp.w <- akerd(flp.w, pyhat = TRUE, plotit = FALSE)
a.flp.nw <- akerd(flp.nw, pyhat = TRUE, plotit = FALSE)

# create data frame
df <- tibble(`x`=c(flp.w,flp.nw),
             `y`=c(a.flp.w,a.flp.nw),
             `Condition`=c(rep.int("Word",length(flp.w)),
                           rep.int("Non-word",length(flp.nw))))
# make plot
df$Condition <- as.character(df$Condition)
df$Condition <- factor(df$Condition, levels=unique(df$Condition))

# make plot
p <- ggplot(df, aes(x,y, group=Condition)) + theme_classic() +
  geom_line(aes(colour=Condition), size = 1.5) + # linetype=Condition,
  # scale_size_manual(values=c(1,0.5)) +
  # scale_linetype_manual(values=c("solid","solid")) +
  scale_color_manual(values=c("grey30", "#E69F00")) + #, "#56B4E9","black")) + grey #999999
  scale_x_continuous(limits=c(0,1600), breaks=seq(0,1600,250), minor_breaks = waiver()) +
  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.key.width = unit(1.5,"cm"),
        legend.position = c(0.75,0.8),
        legend.title = element_text(size=16),
        legend.text = element_text(size = 14),
        strip.text.y = element_text(size = 18, face = "bold", angle = 0)) +
        # legend.position = c(0.25,0.9)) +
  labs(x = "Reaction times", y = "Density") +
  ggtitle("Median reaction times")
p
```

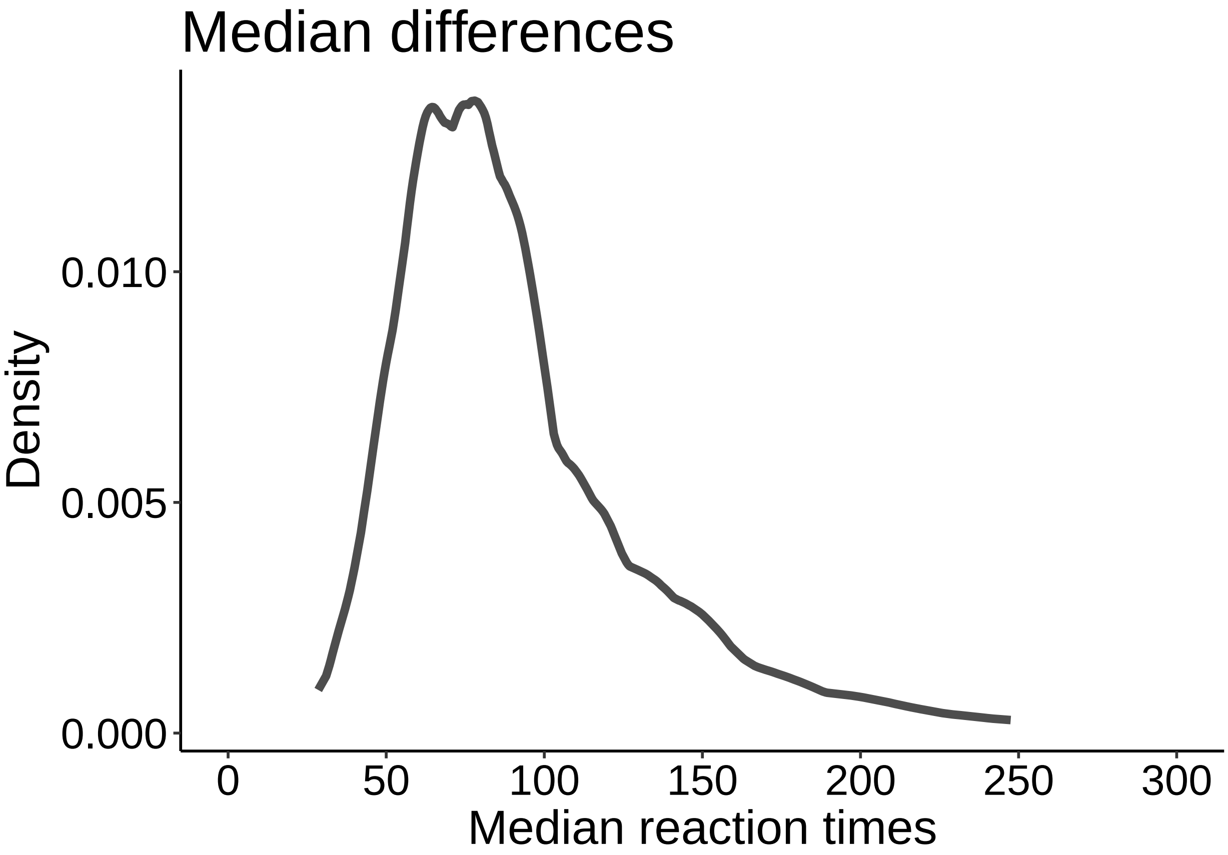


```r
# save figure
ggsave(filename='./figure_all_p_medians.jpg',width=10,height=7) #path=pathname
```

Distribution of differences: non-word - word

```r
# make KDE
diff <- sort(flp.nw - flp.w)
a.diff <- akerd(diff, pyhat = TRUE, plotit = FALSE)

# create data frame
df <- tibble(`x`=diff,
             `y`=a.diff)
# make plot
p <- ggplot(df, aes(x,y)) + theme_classic() +
  geom_line(colour="grey30", size = 1.5) + # linetype=Condition,
  scale_x_continuous(limits=c(0,300), breaks=seq(0,300,50), minor_breaks = waiver()) +
  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.key.width = unit(1.5,"cm"),
        legend.position = c(0.75,0.8),
        legend.title = element_text(size=16),
        legend.text = element_text(size = 14),
        strip.text.y = element_text(size = 18, face = "bold", angle = 0)) +
        # legend.position = c(0.25,0.9)) +
  labs(x = "Median reaction times", y = "Density") +
  ggtitle("Median differences")
p
```



```
# save figure
ggsave(filename='./figure_all_p_median_diff.jpg',width=10,height=7) #path=pathname
```

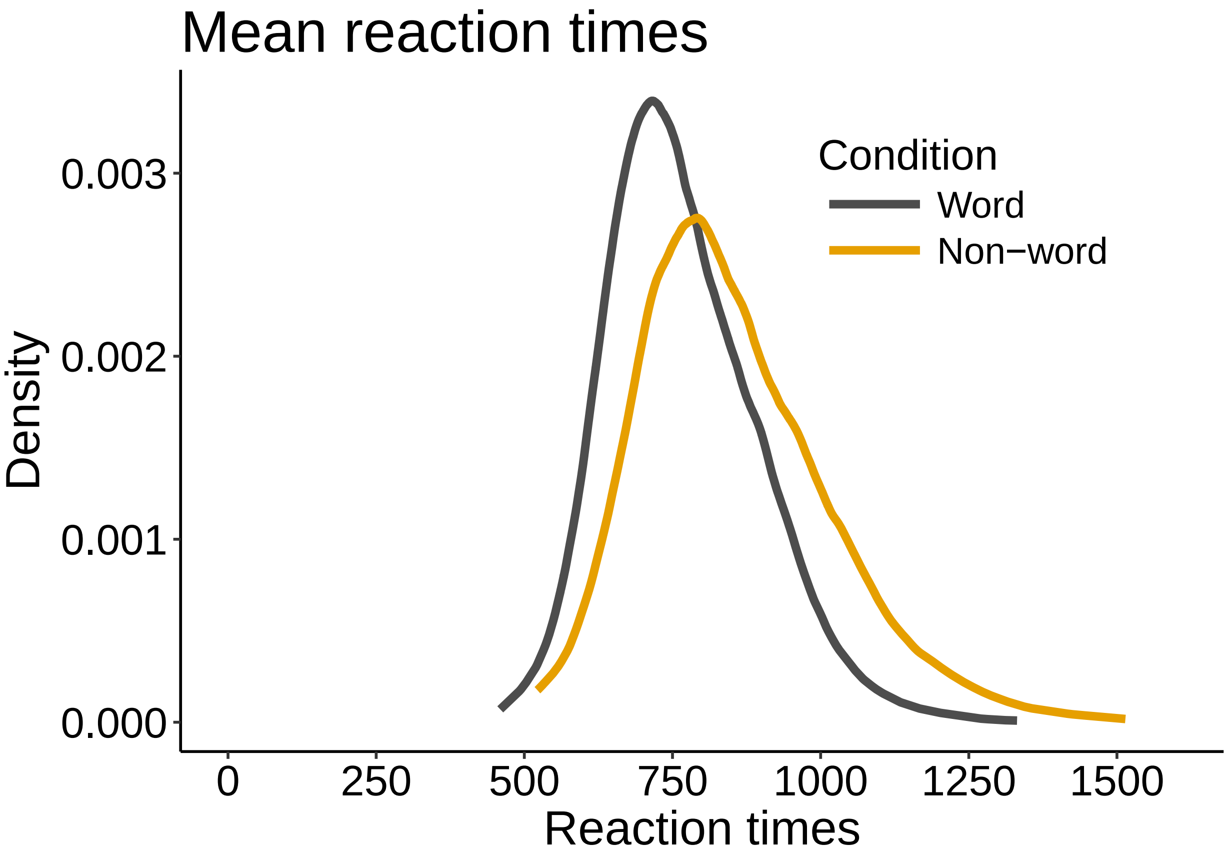
Distributions of mean RT

```
# make KDE
flp.w <- sort(meanres[,1])
flp.nw <- sort(meanres[,2])
a.flp.w <- akerd(flp.w, pyhat = TRUE, plotit = FALSE)
a.flp.nw <- akerd(flp.nw, pyhat = TRUE, plotit = FALSE)

# create data frame
df <- tibble(`x`=c(flp.w,flp.nw),
             `y`=c(a.flp.w,a.flp.nw),
             `Condition`=c(rep.int("Word",length(flp.w)),
                           rep.int("Non-word",length(flp.nw))))
# make plot
df$Condition <- as.character(df$Condition)
df$Condition <- factor(df$Condition, levels=unique(df$Condition))

# make plot
```

```r
p <- ggplot(df, aes(x,y, group=Condition)) + theme_classic() +
  geom_line(aes(colour=Condition), size = 1.5) + # linetype=Condition,
  # scale_size_manual(values=c(1,0.5)) +
  # scale_linetype_manual(values=c("solid","solid")) +
  scale_color_manual(values=c("grey30", "#E69F00")) + #, "#56B4E9","black")) + grey #999999
  scale_x_continuous(limits=c(0,1600), breaks=seq(0,1600,250), minor_breaks = waiver()) +
  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.key.width = unit(1.5,"cm"),
        legend.position = c(0.75,0.8),
        legend.title = element_text(size=16),
        legend.text = element_text(size = 14),
        strip.text.y = element_text(size = 18, face = "bold", angle = 0)) +
        # legend.position = c(0.25,0.9)) +
  labs(x = "Reaction times", y = "Density") +
  ggtitle("Mean reaction times")
p
```

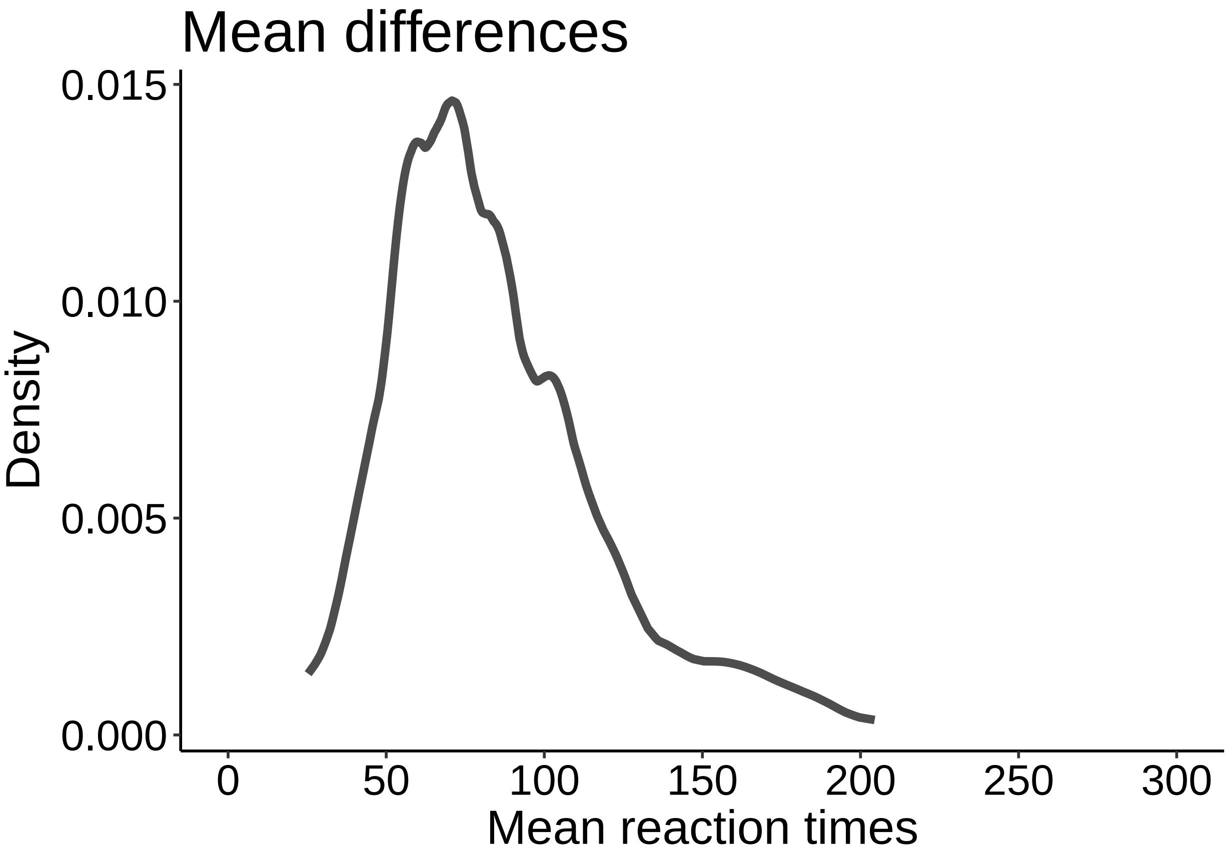


```r
# save figure
ggsave(filename='./figure_all_p_means.jpg',width=10,height=7) #path=pathname
```

Distribution of differences: non-word - word

```
# make KDE
diff <- sort(flp.nw - flp.w)
a.diff <- akerd(diff, pyhat = TRUE, plotit = FALSE)

# create data frame
df <- tibble(`x`=diff,
             `y`=a.diff)
# make plot
p <- ggplot(df, aes(x,y)) + theme_classic() +
  geom_line(colour="grey30", size = 1.5) + # linetype=Condition,
  scale_x_continuous(limits=c(0,300), breaks=seq(0,300,50), minor_breaks = waiver()) +
  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.key.width = unit(1.5,"cm"),
        legend.position = c(0.75,0.8),
        legend.title = element_text(size=16),
        legend.text = element_text(size = 14),
        strip.text.y = element_text(size = 18, face = "bold", angle = 0)) +
        # legend.position = c(0.25,0.9)) +
  labs(x = "Mean reaction times", y = "Density") +
  ggtitle("Mean differences")
p
```



```r
# save figure
ggsave(filename='./figure_all_p_mean_diff.jpg',width=10,height=7) #path=pathname
```

How to take random samples of trials and participants

```r
set.seed(21)

p.list <- unique(flp$participant)
nmax <- length(p.list)

# get data: random selection of 100 trials + median for every participant
nT <- 100
nP <- 20
medres <- tapply(flp$rt,
                 list(flp$participant, flp$condition),
                 function(s) median(sample(s, nT, replace=TRUE)) )
# summary statistics across all participants
summary(medres)
```

```
##       word           non-word     
##  Min.   : 451.0   Min.   : 495.5  
##  1st Qu.: 619.0   1st Qu.: 683.5  
##  Median : 681.5   Median : 762.5  
##  Mean   : 701.0   Mean   : 792.1  
##  3rd Qu.: 761.8   3rd Qu.: 868.5  
##  Max.   :1352.5   Max.   :1587.0  
```

```r
# summary statistics for a random selection of nP participants
summary(medres[sample(nmax,nP,replace=FALSE),])
```

```
##       word          non-word     
##  Min.   :467.0   Min.   : 499.0  
##  1st Qu.:609.2   1st Qu.: 686.0  
##  Median :651.2   Median : 749.5  
##  Mean   :704.2   Mean   : 776.4  
##  3rd Qu.:788.9   3rd Qu.: 824.1  
##  Max.   :959.0   Max.   :1154.0  
```

10,000 experiments: random sample of participants only

We use all the trials available for each participant but vary how many participants we sample.

```r
set.seed(21)

p.list <- unique(flp$participant)
nmax <- length(p.list)
nsim <- 10000
pvec <- seq(10,100,10) # number of participants

sim.m.w <- matrix(0, nrow = nsim, ncol = length(pvec))
sim.m.nw <- matrix(0, nrow = nsim, ncol = length(pvec))
sim.m.diff <- matrix(0, nrow = nsim, ncol = length(pvec))
sim.md.w <- matrix(0, nrow = nsim, ncol = length(pvec))
sim.md.nw <- matrix(0, nrow = nsim, ncol = length(pvec))
sim.md.diff <- matrix(0, nrow = nsim, ncol = length(pvec))

# use all trials and compute mean and median
  m.res <- tapply(flp$rt,
                  list(flp$participant, flp$condition),
                  mean)
  md.res <- tapply(flp$rt,
                  list(flp$participant, flp$condition),
                  median)

for(iter.n in 1:length(pvec)){
    print(paste("Sample size = ",pvec[iter.n],"..."))

  # sample participants with replacement and compute group mean and median
  mc.samp <- sample(nmax, pvec[iter.n]*nsim, replace=TRUE)

  # group mean
  sim.m.w[,iter.n] <- apply(matrix(m.res[mc.samp,1], nrow = nsim), 1, mean) # word
  sim.m.nw[,iter.n] <- apply(matrix(m.res[mc.samp,2], nrow = nsim), 1, mean) # non-word
  sim.m.diff[,iter.n] <- apply(matrix(m.res[mc.samp,2] - m.res[mc.samp,1],
                                    nrow = nsim), 1, mean) # difference

  # group median
  sim.md.w[,iter.n] <- apply(matrix(md.res[mc.samp,1], nrow = nsim), 1, median) # word
  sim.md.nw[,iter.n] <- apply(matrix(md.res[mc.samp,2], nrow = nsim), 1, median) # non-word
  sim.md.diff[,iter.n] <- apply(matrix(md.res[mc.samp,2] - md.res[mc.samp,1],
                                    nrow = nsim), 1, median) # difference
}
```

```
## [1] "Sample size =  10 ..."
## [1] "Sample size =  20 ..."
## [1] "Sample size =  30 ..."
## [1] "Sample size =  40 ..."
## [1] "Sample size =  50 ..."
## [1] "Sample size =  60 ..."
## [1] "Sample size =  70 ..."
## [1] "Sample size =  80 ..."
## [1] "Sample size =  90 ..."
## [1] "Sample size =  100 ..."
```

Word: median

Make KDE:

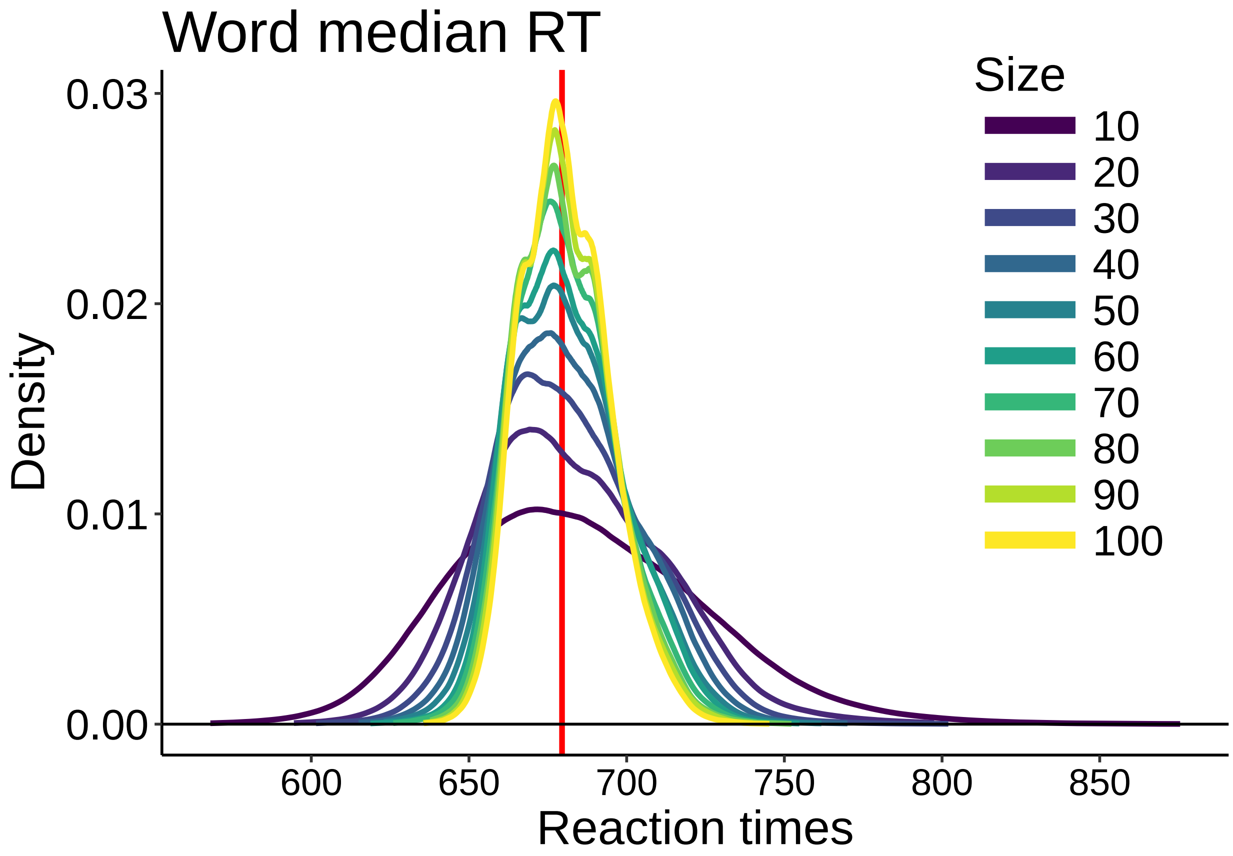
```r
x <- sim.md.w
y <- sim.md.w
for(iter.n in 1:length(pvec)){
  x[,iter.n] <- sort(x[,iter.n])
  y[,iter.n] <- akerd(x[,iter.n], pyhat = TRUE, plotit = FALSE)
}
save(
  x,
  y,
  file = "kde_md_w_size.RData"
)
```

```
load("kde_md_w_size.RData")

df <- tibble(`x`=as.vector(x),
             `y`=as.vector(y),
             `Size`=rep(pvec,each=nsim))

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

# make plot
p <- ggplot(df, aes(x=x, y=y)) + theme_classic() +
  # population value
  geom_vline(xintercept = pop.md.w, colour="red", size=1) +
  geom_line(aes(colour = Size), size = 1) +
  geom_abline(intercept=0, slope=0, colour="black") +
  scale_color_viridis(discrete = TRUE) +
  scale_x_continuous(breaks = seq(500, 1000, 50)) +
  # coord_cartesian(xlim=c(500, 1000)) +
  theme(plot.title = element_text(size=22),
        axis.title.x = element_text(size = 18),
        axis.text.x = element_text(size = 14, colour="black"),
        axis.text.y = element_text(size = 16, colour="black"),
        axis.title.y = element_text(size = 18),
        legend.key.width = unit(1.5,"cm"),
        legend.position = c(0.85,0.65),
        legend.text=element_text(size=16),
        legend.title=element_text(size=18)) +
  labs(x = "Reaction times", y = "Density") +
  guides(colour = guide_legend(override.aes = list(size=3))) + # make thicker legend lines
  ggtitle("Word median RT")
p
```



```r
# save figure
ggsave(filename = 'figure_md_w_size.jpg',width=10,height=6) #path=pathname
```

Highest density intervals

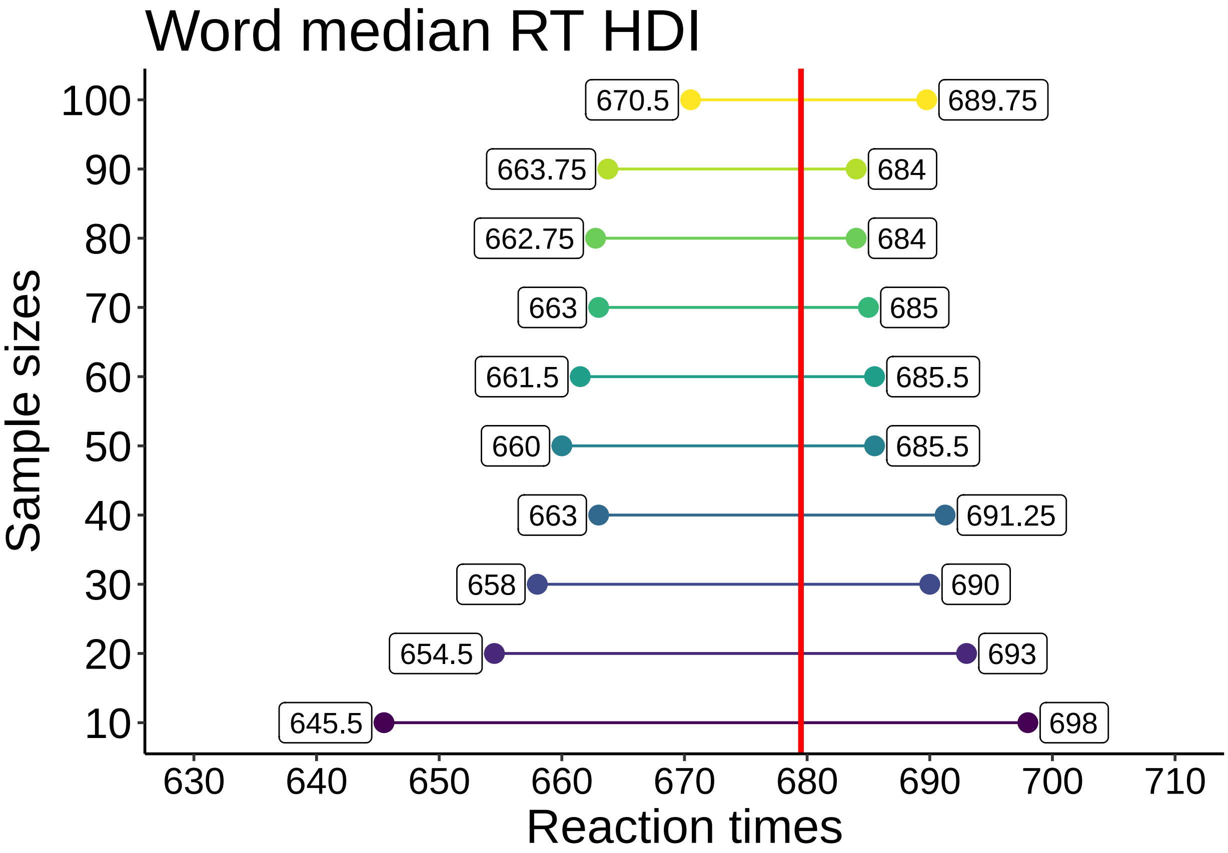
Compute 50% highest density interval (HDI) for each sample size distribution. To see HDI in action, see John Kruschke's blog, for instance this post.

```r
hdi.res <- matrix(0, nrow=length(pvec), ncol=2)
for(P in 1:length(pvec)){
hdi.res[P,] <- HDIofMCMC(sim.md.w[,P], credMass=0.50)
}
```

Illustrate results

```r
df <- tibble(x = as.vector(hdi.res),
             y = rep(pvec,2),
             label = as.character(as.vector(hdi.res)))
df.seg <- tibble(x = hdi.res[,1],
                 y = pvec,
                 xend = hdi.res[,2],
                 yend = pvec)
df.label1 <- tibble(x = hdi.res[,1],
             y = pvec,
             label = as.character(hdi.res[,1]))
df.label2 <- tibble(x = hdi.res[,2],
             y = pvec,
             label = as.character(hdi.res[,2]))

ggplot(df, aes(x=x, y=y)) + theme_classic() +
  geom_point(size=3, aes(colour=y)) +
  geom_segment(data=df.seg, aes(x=x, xend=xend, y=y, yend=yend, colour=y)) +
  scale_color_viridis(discrete = FALSE) +
  scale_y_continuous(breaks = pvec) +
  geom_label(data=df.label1, aes(label=label), hjust = "outward", nudge_x = -1) +
  geom_label(data=df.label2, aes(label=label), hjust = "outward", nudge_x = 1) +
  geom_vline(xintercept = pop.md.w, colour="red", size=1) +
  scale_x_continuous(breaks = seq(600, 800, 10)) +
  coord_cartesian(xlim=c(630, 710)) +
  theme(plot.title = element_text(size=22),
        axis.title.x = element_text(size = 18),
        axis.text.x = element_text(size = 14, colour="black"),
        axis.text.y = element_text(size = 16, 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)) +
  labs(x = "Reaction times", y = "Sample sizes") +
  ggtitle("Word median RT HDI")
```



```
# save figure
ggsave(filename = 'figure_md_w_size_hdi.jpg',width=10,height=6) #path=pathname
```

Non-word: median

Make KDE:

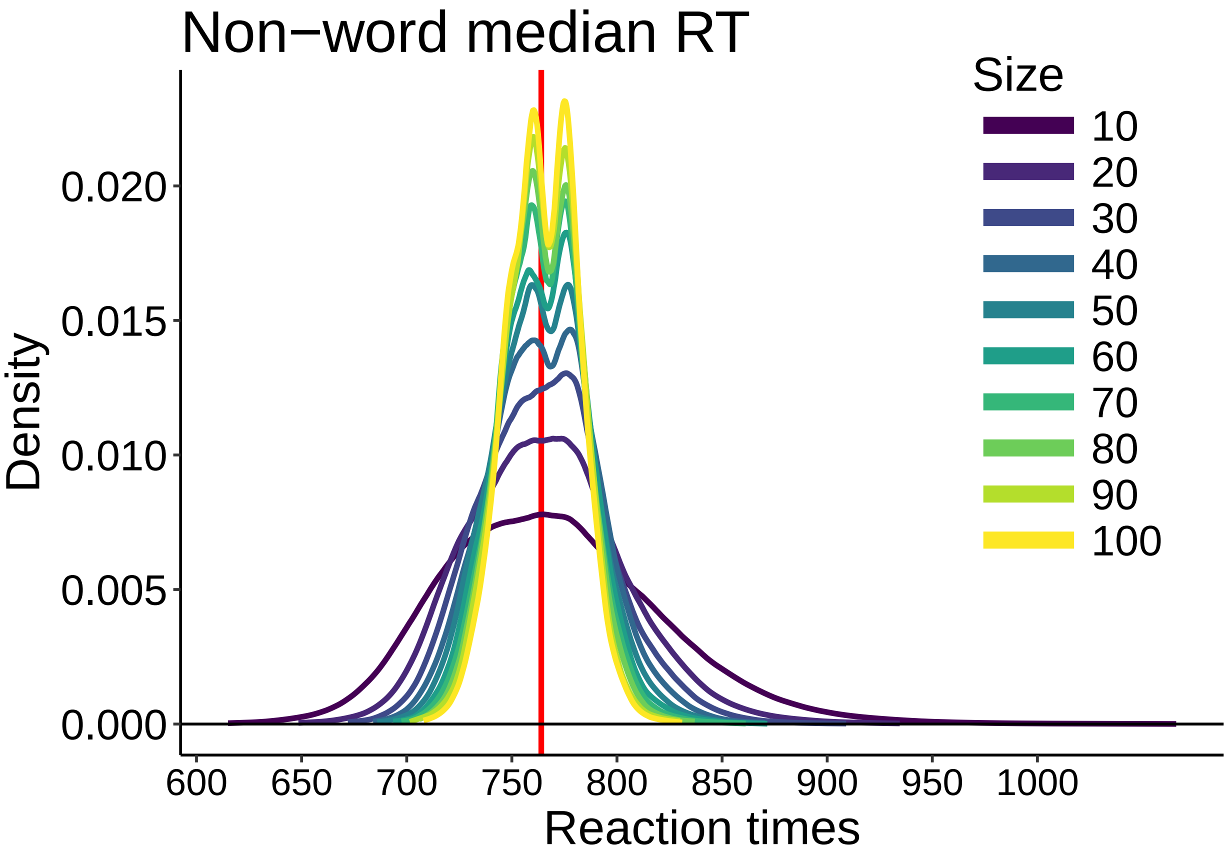
```r
x <- sim.md.nw
y <- sim.md.nw
for(iter.n in 1:length(pvec)){
  x[,iter.n] <- sort(x[,iter.n])
  y[,iter.n] <- akerd(x[,iter.n], pyhat = TRUE, plotit = FALSE)
}
save(
  x,
  y,
  file = "kde_md_nw_size.RData"
)
```

```r
load("kde_md_nw_size.RData")

df <- tibble(`x`=as.vector(x),
             `y`=as.vector(y),
             `Size`=rep(pvec,each=nsim))

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

# make plot
p <- ggplot(df, aes(x=x, y=y)) + theme_classic() +
  # population value
  geom_vline(xintercept = pop.md.nw, colour="red", size=1) +
  geom_line(aes(colour = Size), size = 1) +
  geom_abline(intercept=0, slope=0, colour="black") +
  scale_color_viridis(discrete = TRUE) +
  scale_x_continuous(breaks = seq(500, 1000, 50)) +
  # coord_cartesian(xlim=c(500, 1000)) +
  theme(plot.title = element_text(size=22),
        axis.title.x = element_text(size = 18),
        axis.text.x = element_text(size = 14, colour="black"),
        axis.text.y = element_text(size = 16, colour="black"),
        axis.title.y = element_text(size = 18),
        legend.key.width = unit(1.5,"cm"),
        legend.position = c(0.85,0.65),
        legend.text=element_text(size=16),
        legend.title=element_text(size=18)) +
  labs(x = "Reaction times", y = "Density") +
  guides(colour = guide_legend(override.aes = list(size=3))) + # make thicker legend lines
  ggtitle("Non-word median RT")
p
```



```
# save figure
ggsave(filename = 'figure_md_nw_size.jpg',width=10,height=6) #path=pathname
```

Difference: median

Make KDE:

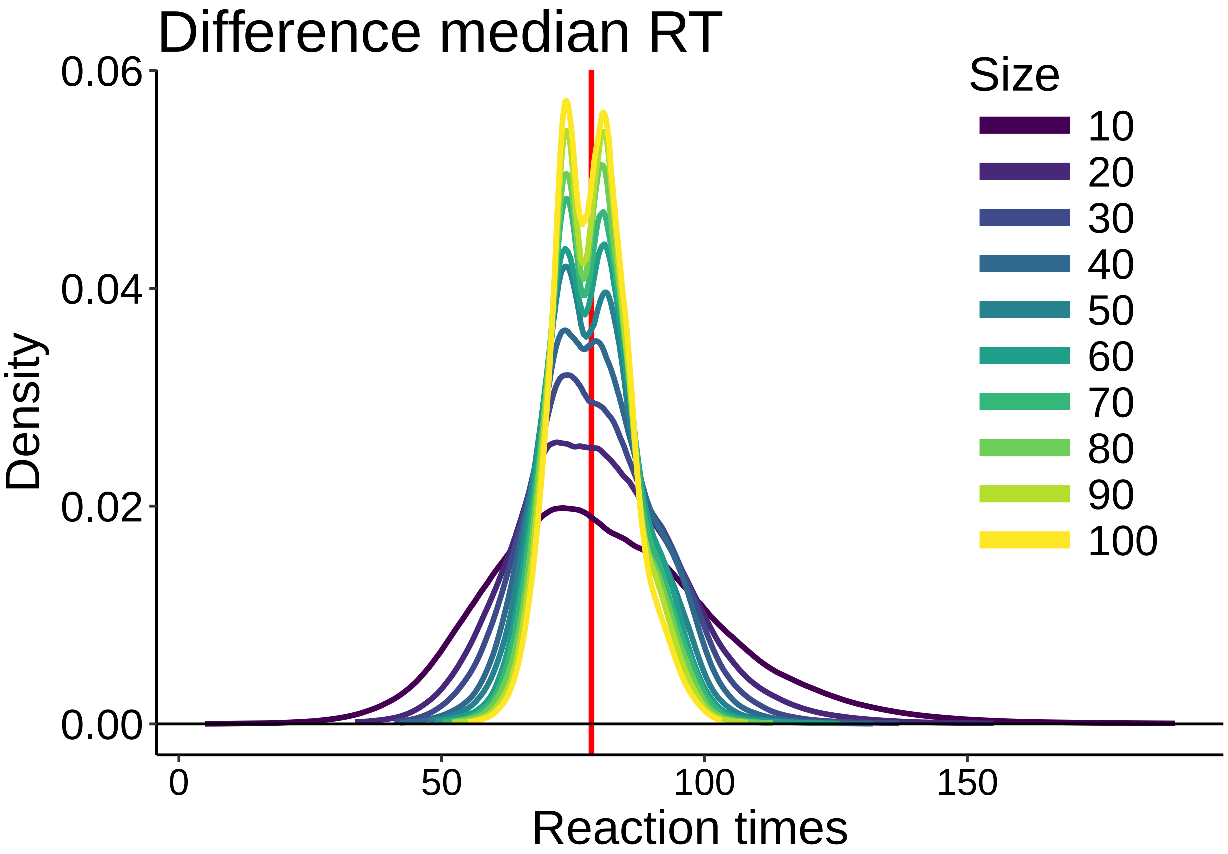
```
x <- sim.md.diff
y <- sim.md.diff
for(iter.n in 1:length(pvec)){
  x[,iter.n] <- sort(x[,iter.n])
  y[,iter.n] <- akerd(x[,iter.n], pyhat = TRUE, plotit = FALSE)
}
save(
  x,
  y,
  file = "kde_md_diff_size.RData"
)
```

```
load("kde_md_diff_size.RData")

df <- tibble(`x`=as.vector(x),
             `y`=as.vector(y),
             `Size`=rep(pvec,each=nsim))

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

```r
# make plot
p <- ggplot(df, aes(x=x, y=y)) + theme_classic() +
  # population value
  geom_vline(xintercept = pop.md.diff, colour="red", size=1) +
  geom_line(aes(colour = Size), size = 1) +
  geom_abline(intercept=0, slope=0, colour="black") +
  scale_color_viridis(discrete = TRUE) +
  # scale_x_continuous(breaks = seq(600, 900, 50)) +
  theme(plot.title = element_text(size=22),
        axis.title.x = element_text(size = 18),
        axis.text.x = element_text(size = 14, colour="black"),
        axis.text.y = element_text(size = 16, colour="black"),
        axis.title.y = element_text(size = 18),
        legend.key.width = unit(1.5,"cm"),
        legend.position = c(0.85,0.65),
        legend.text=element_text(size=16),
        legend.title=element_text(size=18)) +
  labs(x = "Reaction times", y = "Density") +
  guides(colour = guide_legend(override.aes = list(size=3))) + # make thicker legend lines
  ggtitle("Difference median RT")
p
```



```r
# save figure
ggsave(filename = 'figure_md_diff_size.jpg',width=10,height=6) #path=pathname
```

Probability of being wrong

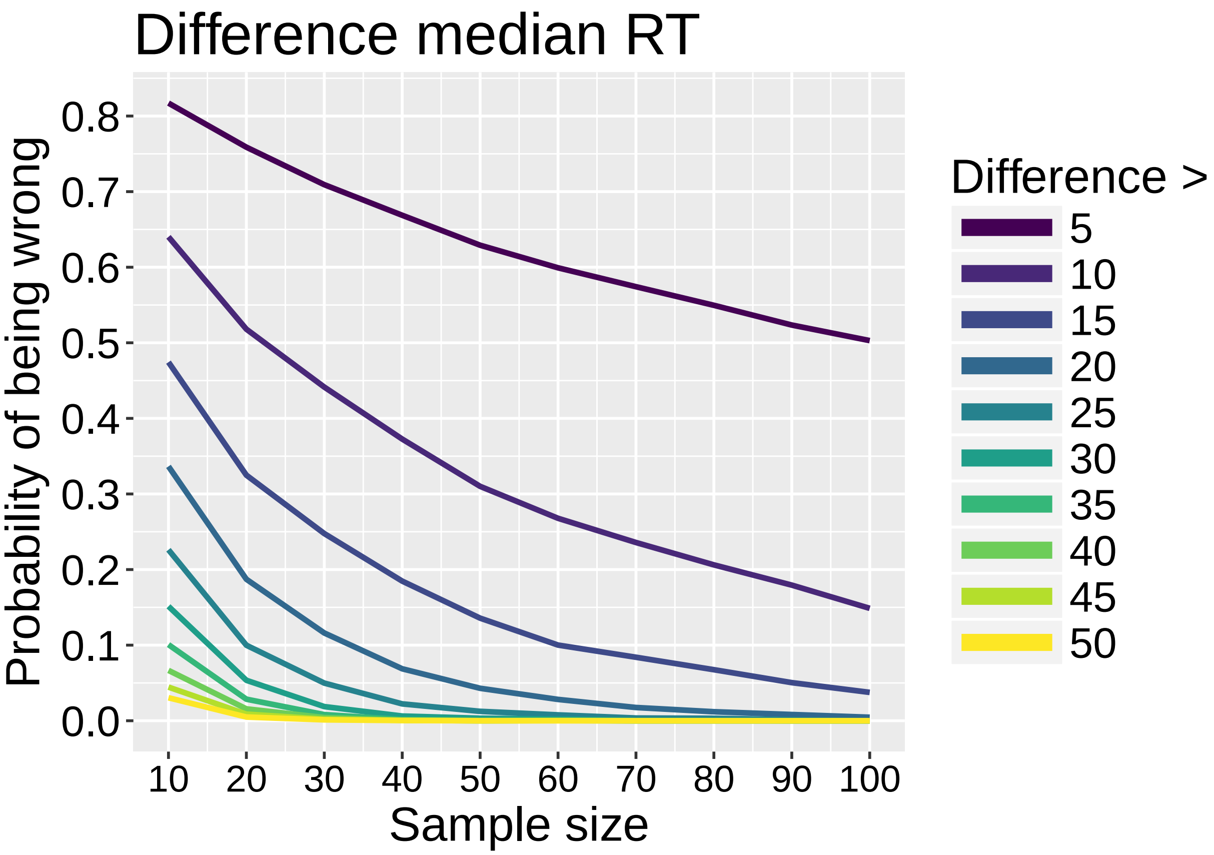
```
diffvec <- seq(5,50,5) # absolute differences to the population value
pw <- matrix(0, ncol = length(diffvec), nrow = length(pvec))
for(D in 1:length(diffvec)){
  pw[D,] <- apply(abs(sim.md.diff - pop.md.diff) >= diffvec[D], 2, mean)
}
pw
```

```
##         [,1]   [,2]   [,3]   [,4]   [,5]   [,6]   [,7]   [,8]   [,9]
##  [1,] 0.8171 0.7588 0.7092 0.6687 0.6291 0.5992 0.5742 0.5496 0.5235
##  [2,] 0.6402 0.5180 0.4414 0.3726 0.3101 0.2678 0.2358 0.2062 0.1794
##  [3,] 0.4744 0.3249 0.2476 0.1847 0.1357 0.1001 0.0840 0.0675 0.0504
##  [4,] 0.3365 0.1872 0.1161 0.0687 0.0429 0.0282 0.0175 0.0119 0.0082
##  [5,] 0.2262 0.0999 0.0498 0.0223 0.0124 0.0077 0.0036 0.0032 0.0015
##  [6,] 0.1514 0.0535 0.0187 0.0062 0.0033 0.0023 0.0005 0.0006 0.0000
##  [7,] 0.1007 0.0285 0.0081 0.0017 0.0006 0.0006 0.0000 0.0000 0.0000
##  [8,] 0.0666 0.0157 0.0042 0.0006 0.0000 0.0003 0.0000 0.0000 0.0000
##  [9,] 0.0445 0.0087 0.0021 0.0006 0.0000 0.0002 0.0000 0.0000 0.0000
## [10,] 0.0304 0.0049 0.0012 0.0003 0.0000 0.0000 0.0000 0.0000 0.0000
##        [,10]
##  [1,] 0.5029
##  [2,] 0.1487
##  [3,] 0.0375
##  [4,] 0.0046
##  [5,] 0.0000
##  [6,] 0.0000
##  [7,] 0.0000
##  [8,] 0.0000
##  [9,] 0.0000
## [10,] 0.0000
```

```
df <- tibble(x = rep(pvec, each = length(diffvec)),
             y = as.vector(pw),
             difference = rep(diffvec, length(pvec)))

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

p <- ggplot(df, aes(x=x, y=y)) +
  geom_line(aes(colour=difference), size = 1) +
  scale_color_viridis(discrete = TRUE) +
  scale_x_continuous(breaks = pvec) +
  scale_y_continuous(breaks = seq(0,1,0.1)) +
  theme(plot.title = element_text(size=22),
        axis.title.x = element_text(size = 18),
        axis.text.x = element_text(size = 14, colour="black"),
        axis.text.y = element_text(size = 16, colour="black"),
        axis.title.y = element_text(size = 18),
        legend.key.width = unit(1.5,"cm"),
        legend.position = "right",#c(0.85,0.65),
        legend.text=element_text(size=16),
        legend.title=element_text(size=18)) +
  labs(x = "Sample size", y = "Probability of being wrong", colour="Difference >") +
  guides(colour = guide_legend(override.aes = list(size=3))) + # make thicker legend lines
  ggtitle("Difference median RT")
p
```



```
# save figure
ggsave(filename = 'figure_md_diff_size_prob.jpg',width=10,height=6) #path=pathname
```

Highest density intervals

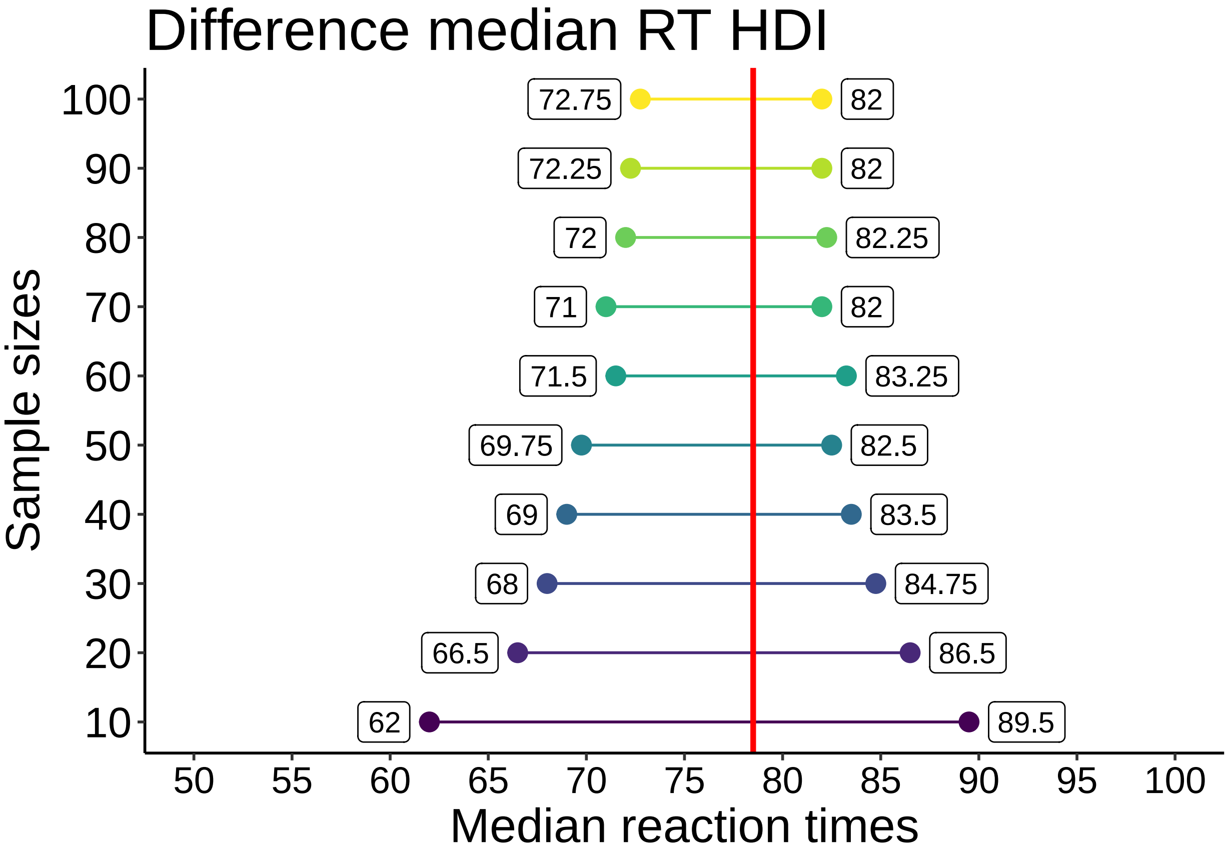
```
hdi.res <- matrix(0, nrow=length(pvec), ncol=2)
for(P in 1:length(pvec)){
hdi.res[P,] <- HDIofMCMC(sim.md.diff[,P], credMass=0.50)
}
```

Illustrate results

```
df <- tibble(x = as.vector(hdi.res),
             y = rep(pvec,2),
             label = as.character(as.vector(hdi.res)))
df.seg <- tibble(x = hdi.res[,1],
                 y = pvec,
                 xend = hdi.res[,2],
                 yend = pvec)
df.label1 <- tibble(x = hdi.res[,1],
             y = pvec,
             label = as.character(hdi.res[,1]))
```

```
df.label2 <- tibble(x = hdi.res[,2],
                    y = pvec,
                    label = as.character(hdi.res[,2]))

ggplot(df, aes(x=x, y=y)) + theme_classic() +
  geom_point(size=3, aes(colour=y)) +
  geom_segment(data=df.seg, aes(x=x, xend=xend, y=y, yend=yend, colour=y)) +
  scale_color_viridis(discrete = FALSE) +
  scale_y_continuous(breaks = pvec) +
  geom_label(data=df.label1, aes(label=label), hjust = "outward", nudge_x = -1) +
  geom_label(data=df.label2, aes(label=label), hjust = "outward", nudge_x = 1) +
  geom_vline(xintercept = pop.md.diff, colour="red", size=1) +
  scale_x_continuous(breaks = seq(50, 100, 5)) +
  coord_cartesian(xlim=c(50, 100)) +
  theme(plot.title = element_text(size=22),
        axis.title.x = element_text(size = 18),
        axis.text.x = element_text(size = 14, colour="black"),
        axis.text.y = element_text(size = 16, 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)) +
  labs(x = "Median reaction times", y = "Sample sizes") +
  ggtitle("Difference median RT HDI")
```



```
# save figure
ggsave(filename = 'figure_md_diff_size_hdi.jpg',width=10,height=6) #path=pathname
```

Word: mean

Make KDE:

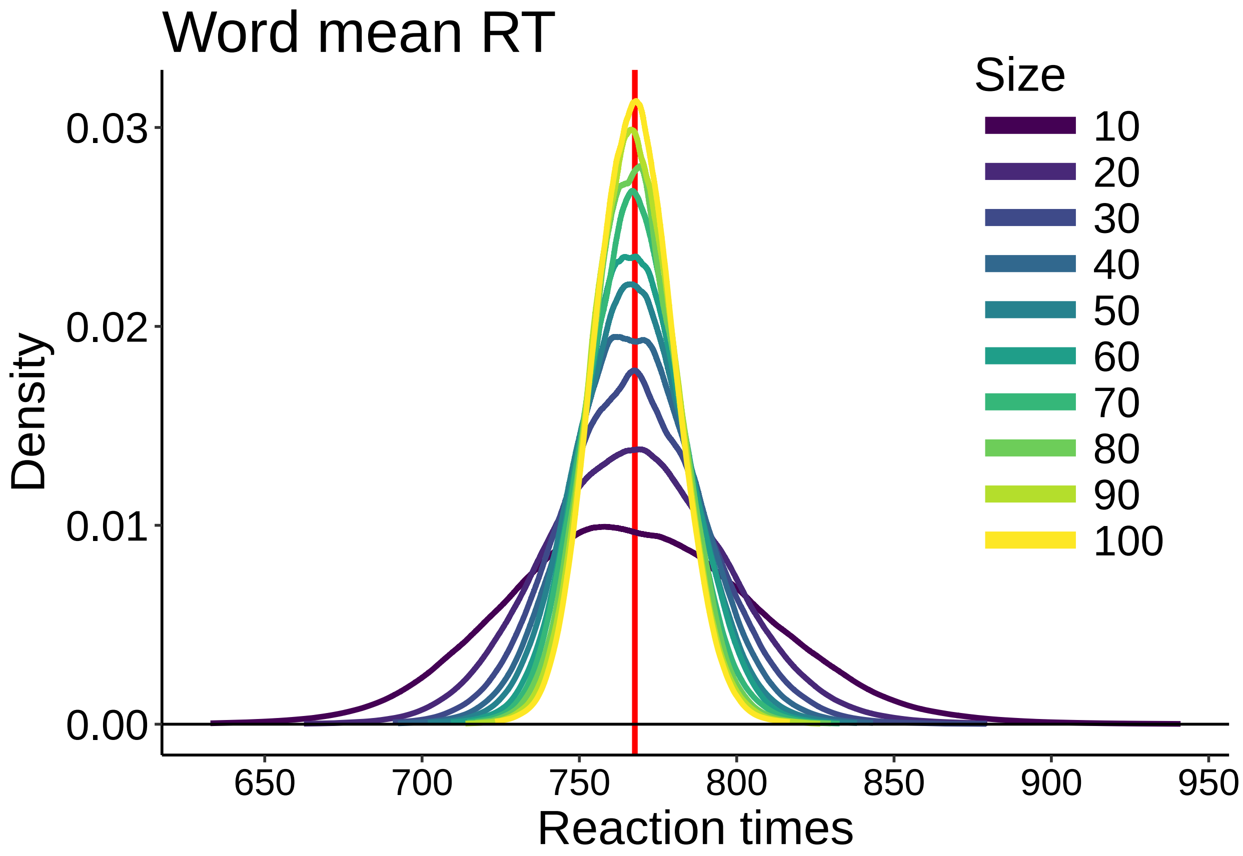
```
x <- sim.m.w
y <- sim.m.w
for(iter.n in 1:length(pvec)){
  x[,iter.n] <- sort(x[,iter.n])
  y[,iter.n] <- akerd(x[,iter.n], pyhat = TRUE, plotit = FALSE)
}
save(
  x,
  y,
  file = "kde_m_w_size.RData"
)
```

```
load("kde_m_w_size.RData")

df <- tibble(`x`=as.vector(x),
             `y`=as.vector(y),
             `Size`=rep(pvec,each=nsim))

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

# make plot
p <- ggplot(df, aes(x=x, y=y)) + theme_classic() +
  # population value
  geom_vline(xintercept = pop.m.w, colour="red", size=1) +
  geom_line(aes(colour = Size), size = 1) +
  geom_abline(intercept=0, slope=0, colour="black") +
  scale_color_viridis(discrete = TRUE) +
  scale_x_continuous(breaks = seq(500, 1000, 50)) +
  # coord_cartesian(xlim=c(500, 1000)) +
  theme(plot.title = element_text(size=22),
        axis.title.x = element_text(size = 18),
        axis.text.x = element_text(size = 14, colour="black"),
        axis.text.y = element_text(size = 16, colour="black"),
        axis.title.y = element_text(size = 18),
        legend.key.width = unit(1.5,"cm"),
        legend.position = c(0.85,0.65),
        legend.text=element_text(size=16),
        legend.title=element_text(size=18)) +
  labs(x = "Reaction times", y = "Density") +
  guides(colour = guide_legend(override.aes = list(size=3))) + # make thicker legend lines
  ggtitle("Word mean RT")
p
```



```
# save figure
ggsave(filename = 'figure_m_w_size.jpg',width=10,height=6) #path=pathname
```

Non-word: mean

Make KDE:

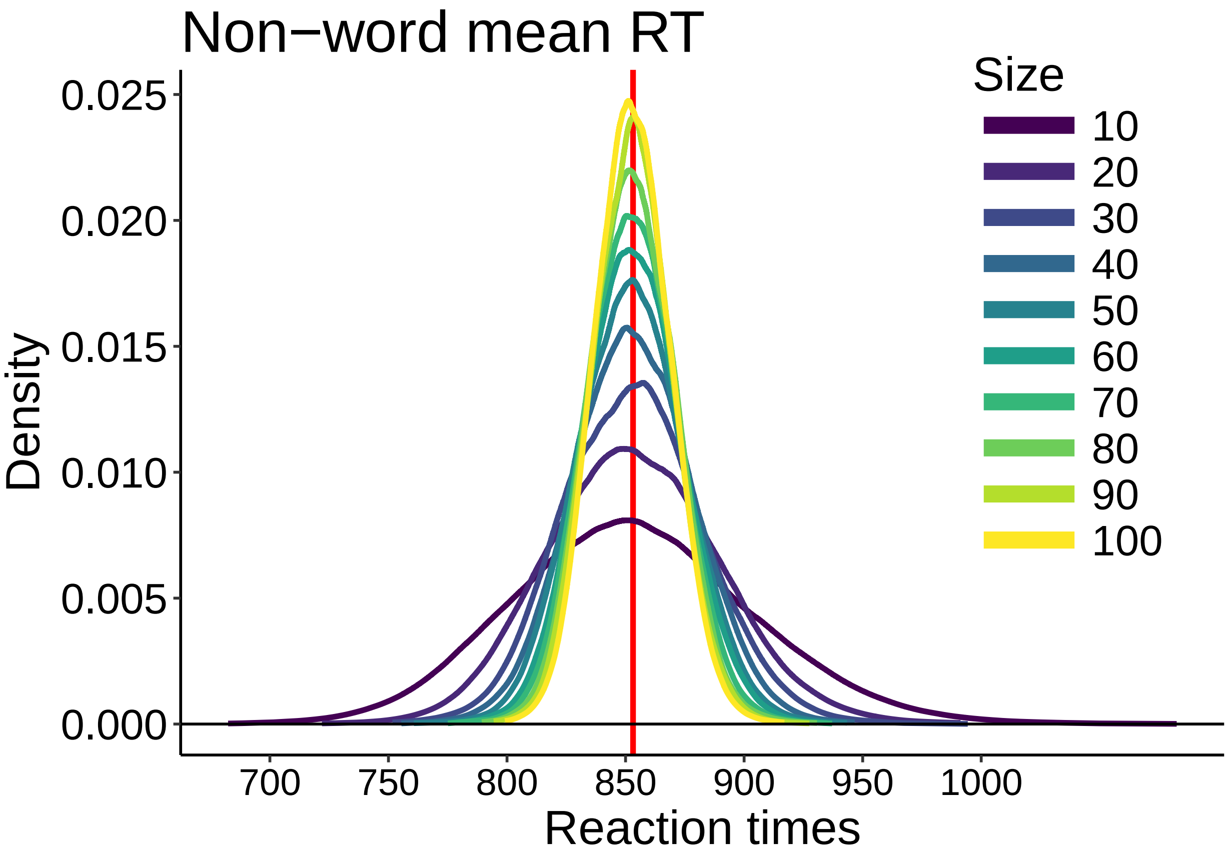
```
x <- sim.m.nw
y <- sim.m.nw
for(iter.n in 1:length(pvec)){
  x[,iter.n] <- sort(x[,iter.n])
  y[,iter.n] <- akerd(x[,iter.n], pyhat = TRUE, plotit = FALSE)
}
save(
  x,
  y,
  file = "kde_m_nw_size.RData"
)
```

```
load("kde_m_nw_size.RData")

df <- tibble(`x`=as.vector(x),
             `y`=as.vector(y),
             `Size`=rep(pvec,each=nsim))

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

```r
# make plot
p <- ggplot(df, aes(x=x, y=y)) + theme_classic() +
  # population value
  geom_vline(xintercept = pop.m.nw, colour="red", size=1) +
  geom_line(aes(colour = Size), size = 1) +
  geom_abline(intercept=0, slope=0, colour="black") +
  scale_color_viridis(discrete = TRUE) +
  scale_x_continuous(breaks = seq(500, 1000, 50)) +
  # coord_cartesian(xlim=c(500, 1000)) +
  theme(plot.title = element_text(size=22),
        axis.title.x = element_text(size = 18),
        axis.text.x = element_text(size = 14, colour="black"),
        axis.text.y = element_text(size = 16, colour="black"),
        axis.title.y = element_text(size = 18),
        legend.key.width = unit(1.5,"cm"),
        legend.position = c(0.85,0.65),
        legend.text=element_text(size=16),
        legend.title=element_text(size=18)) +
  labs(x = "Reaction times", y = "Density") +
  guides(colour = guide_legend(override.aes = list(size=3))) + # make thicker legend lines
  ggtitle("Non-word mean RT")
p
```



```r
# save figure
ggsave(filename = 'figure_m_nw_size.jpg',width=10,height=6) #path=pathname
```

Difference: mean

Make KDE:

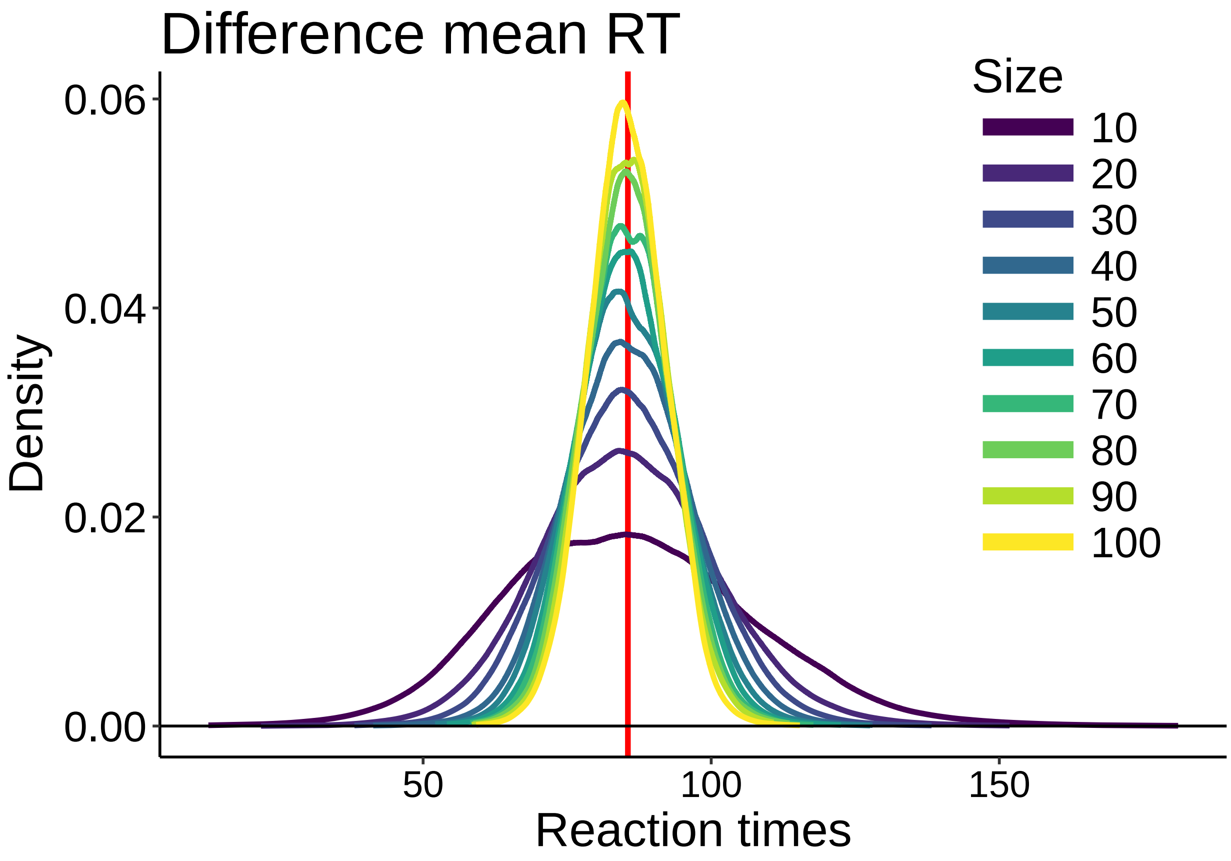
```r
x <- sim.m.diff
y <- sim.m.diff
for(iter.n in 1:length(pvec)){
  x[,iter.n] <- sort(x[,iter.n])
  y[,iter.n] <- akerd(x[,iter.n], pyhat = TRUE, plotit = FALSE)
}
save(
  x,
  y,
  file = "kde_m_diff_size.RData"
)
```

```r
load("kde_m_diff_size.RData")

df <- tibble(`x`=as.vector(x),
             `y`=as.vector(y),
             `Size`=rep(pvec,each=nsim))

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

# make plot
p <- ggplot(df, aes(x=x, y=y)) + theme_classic() +
  # population value
  geom_vline(xintercept = pop.m.diff, colour="red", size=1) +
  geom_line(aes(colour = Size), size = 1) +
  geom_abline(intercept=0, slope=0, colour="black") +
  scale_color_viridis(discrete = TRUE) +
  # scale_x_continuous(breaks = seq(600, 900, 50)) +
  theme(plot.title = element_text(size=22),
        axis.title.x = element_text(size = 18),
        axis.text.x = element_text(size = 14, colour="black"),
        axis.text.y = element_text(size = 16, colour="black"),
        axis.title.y = element_text(size = 18),
        legend.key.width = unit(1.5,"cm"),
        legend.position = c(0.85,0.65),
        legend.text=element_text(size=16),
        legend.title=element_text(size=18)) +
  labs(x = "Reaction times", y = "Density") +
  guides(colour = guide_legend(override.aes = list(size=3))) + # make thicker legend lines
  ggtitle("Difference mean RT")
p
```



```
# save figure
ggsave(filename = 'figure_m_diff_size.jpg',width=10,height=6) #path=pathname
```

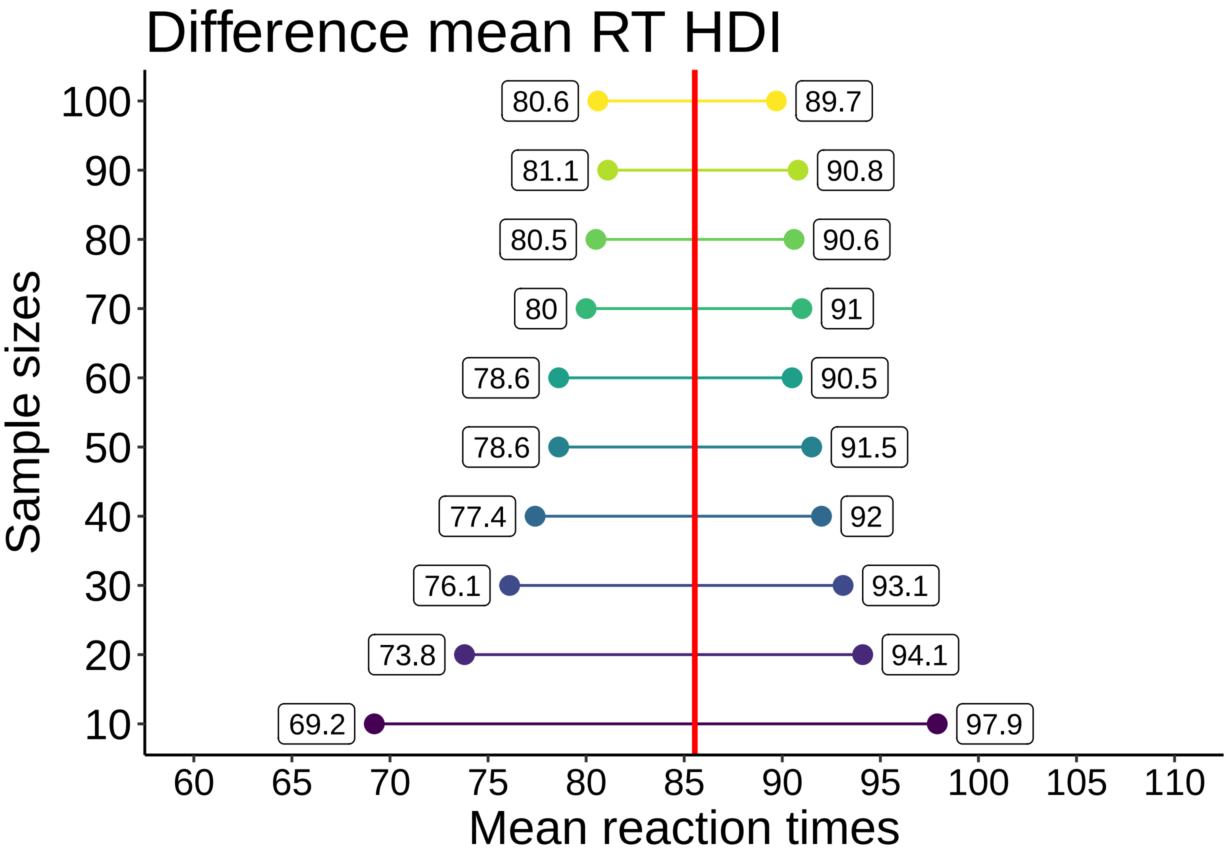
Highest density intervals

```
hdi.res <- matrix(0, nrow=length(pvec), ncol=2)
for(P in 1:length(pvec)){
hdi.res[P,] <- HDIofMCMC(sim.m.diff[,P], credMass=0.50)
}
hdi.res <- round(hdi.res, digits = 1)
```

Illustrate results

```
df <- tibble(x = as.vector(hdi.res),
             y = rep(pvec,2),
             label = as.character(as.vector(hdi.res)))
df.seg <- tibble(x = hdi.res[,1],
                 y = pvec,
                 xend = hdi.res[,2],
                 yend = pvec)
df.label1 <- tibble(x = hdi.res[,1],
             y = pvec,
             label = as.character(hdi.res[,1]))
df.label2 <- tibble(x = hdi.res[,2],
             y = pvec,
             label = as.character(hdi.res[,2]))
```

```
ggplot(df, aes(x=x, y=y)) + theme_classic() +
  geom_point(size=3, aes(colour=y)) +
  geom_segment(data=df.seg, aes(x=x, xend=xend, y=y, yend=yend, colour=y)) +
  scale_color_viridis(discrete = FALSE) +
  scale_y_continuous(breaks = pvec) +
  geom_label(data=df.label1, aes(label=label), hjust = "outward", nudge_x = -1) +
  geom_label(data=df.label2, aes(label=label), hjust = "outward", nudge_x = 1) +
  geom_vline(xintercept = pop.m.diff, colour="red", size=1) +
  scale_x_continuous(breaks = seq(50, 110, 5)) +
  coord_cartesian(xlim=c(60, 110)) +
  theme(plot.title = element_text(size=22),
        axis.title.x = element_text(size = 18),
        axis.text.x = element_text(size = 14, colour="black"),
        axis.text.y = element_text(size = 16, 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)) +
  labs(x = "Mean reaction times", y = "Sample sizes") +
  ggtitle("Difference mean RT HDI")
```



```
# save figure
ggsave(filename = 'figure_m_diff_size_hdi.jpg',width=10,height=6) #path=pathname
```

10,000 experiments: random samples of trials and participants

```r
set.seed(21)

p.list <- unique(flp$participant)
nmax <- length(p.list)
nsim <- 10000
nP <- 20 # number of participants
nT <- 100 # number of trials per participant

sim.m.w <- double(length = nsim)
sim.m.nw <- double(length = nsim)
sim.m.diff <- double(length = nsim)
sim.md.w <- double(length = nsim)
sim.md.nw <- double(length = nsim)
sim.md.diff <- double(length = nsim)

for(iter in 1:nsim){
  if(iter %% 1000 == 0){
    print(paste("Simulation ",iter," / ",nsim,"..."))
  }
  # sample trials with replacement and compute mean and median
  m.res <- tapply(flp$rt,
                 list(flp$participant, flp$condition),
                 function(s) mean(sample(s, nT, replace=TRUE)) )
  md.res <- tapply(flp$rt,
                 list(flp$participant, flp$condition),
                 function(s) median(sample(s, nT, replace=TRUE)) )

  # sample participants with replacement and compute group mean and median
  mc.samp <- sample(nmax, nP, replace=TRUE)
  # group mean
  sim.m.w[iter] <- mean(m.res[mc.samp,1]) # word
  sim.m.nw[iter] <- mean(m.res[mc.samp,2]) # non-word
  sim.m.diff[iter] <- mean(m.res[mc.samp,2] - m.res[mc.samp,1]) # difference
  # group median
  sim.md.w[iter] <- median(md.res[mc.samp,1]) # word
  sim.md.nw[iter] <- median(md.res[mc.samp,2]) # non-word
  sim.md.diff[iter] <- median(md.res[mc.samp,2] - md.res[mc.samp,1]) # difference
}
save(
  nmax,
  nsim,
  nP,
  nT,
  sim.m.w,
  sim.m.nw,
  sim.m.diff,
  sim.md.w,
  sim.md.nw,
  sim.md.diff,
  file = "./sim_res.RData"
)
```

Illustrate results

Compute kernel density estimates.

```
load("./sim_res.RData")
# make KDE
sim.m.w <- sort(sim.m.w)
sim.m.nw <- sort(sim.m.nw)
sim.m.diff <- sort(sim.m.diff)
sim.md.w <- sort(sim.md.w)
sim.md.nw <- sort(sim.md.nw)
sim.md.diff <- sort(sim.md.diff)

a.m.w <- akerd(sim.m.w, pyhat = TRUE, plotit = FALSE)
a.m.nw <- akerd(sim.m.nw, pyhat = TRUE, plotit = FALSE)
a.m.diff <- akerd(sim.m.diff, pyhat = TRUE, plotit = FALSE)
a.md.w <- akerd(sim.md.w, pyhat = TRUE, plotit = FALSE)
a.md.nw <- akerd(sim.md.nw, pyhat = TRUE, plotit = FALSE)
a.md.diff <- akerd(sim.md.diff, pyhat = TRUE, plotit = FALSE)
```

Word condition

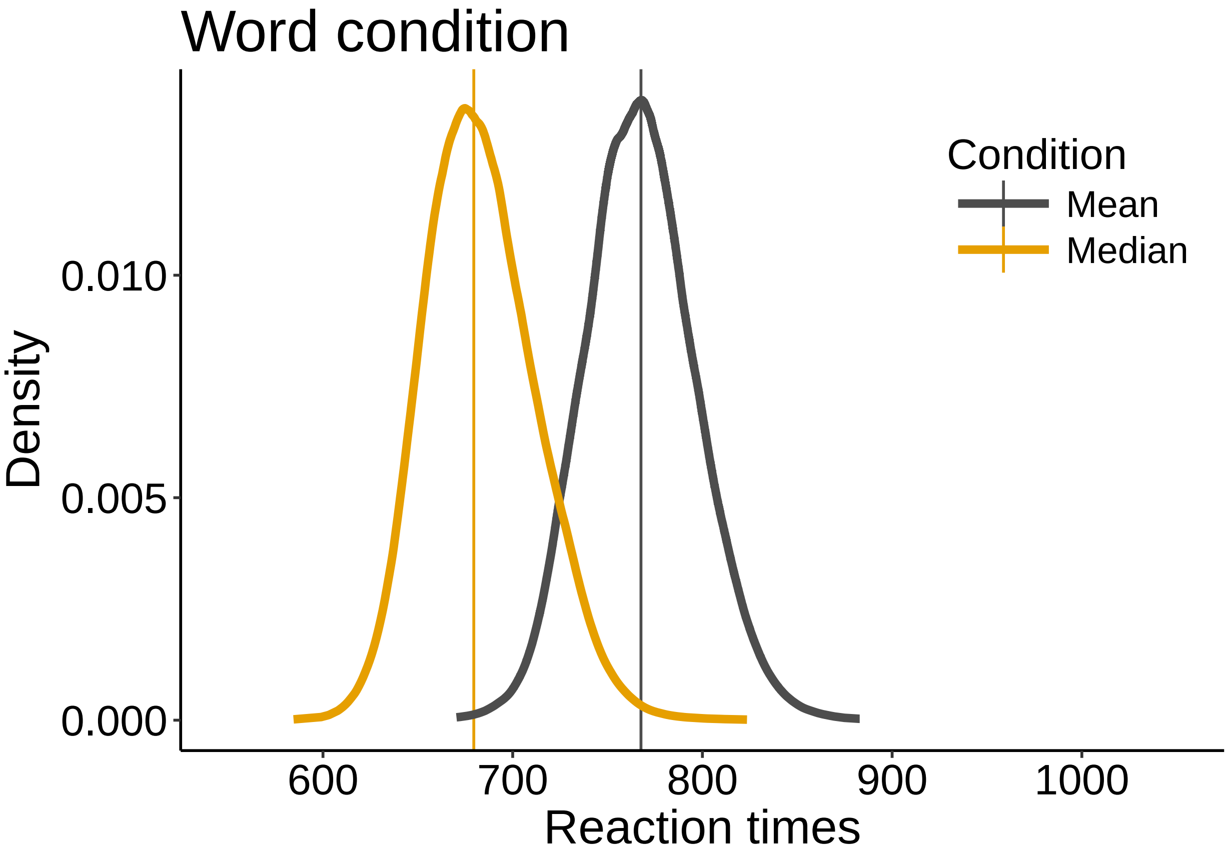
```
# create data frame
df <- tibble(`x`=c(sim.m.w, sim.md.w),
             `y`=c(a.m.w, a.md.w),
             `Condition`= rep(c("Mean", "Median"), each = length(sim.m.w)))

# make factor with preserved name order
df$Condition <- as.character(df$Condition)
df$Condition <- factor(df$Condition, levels=unique(df$Condition))

df.pop <- tibble(`x` = c(pop.m.w, pop.md.w),
                 `Condition`= factor(c("Mean", "Median")))

# make plot
p <- ggplot(df, aes(x,y)) + theme_classic() +
  # add population values
  geom_vline(data = df.pop, aes(xintercept = x, colour = Condition)) +
  geom_line(aes(colour=Condition), size = 1.5) + # linetype=Condition,
  scale_color_manual(values=c("grey30", "#E69F00")) +
  scale_x_continuous(limits=c(550,1050), breaks=seq(500,1000,100)) +
  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.key.width = unit(1.5,"cm"),
        legend.position = c(0.85,0.8),
        legend.title = element_text(size=16),
        legend.text = element_text(size = 14),
        strip.text.y = element_text(size = 18, face = "bold", angle = 0)) +
        # legend.position = c(0.25,0.9)) +
  labs(x = "Reaction times", y = "Density") +
  ggtitle("Word condition")
```

```
p
```



```
# save figure
ggsave(filename='./figure_sim_word.jpg',width=8,height=5) #path=pathname
```

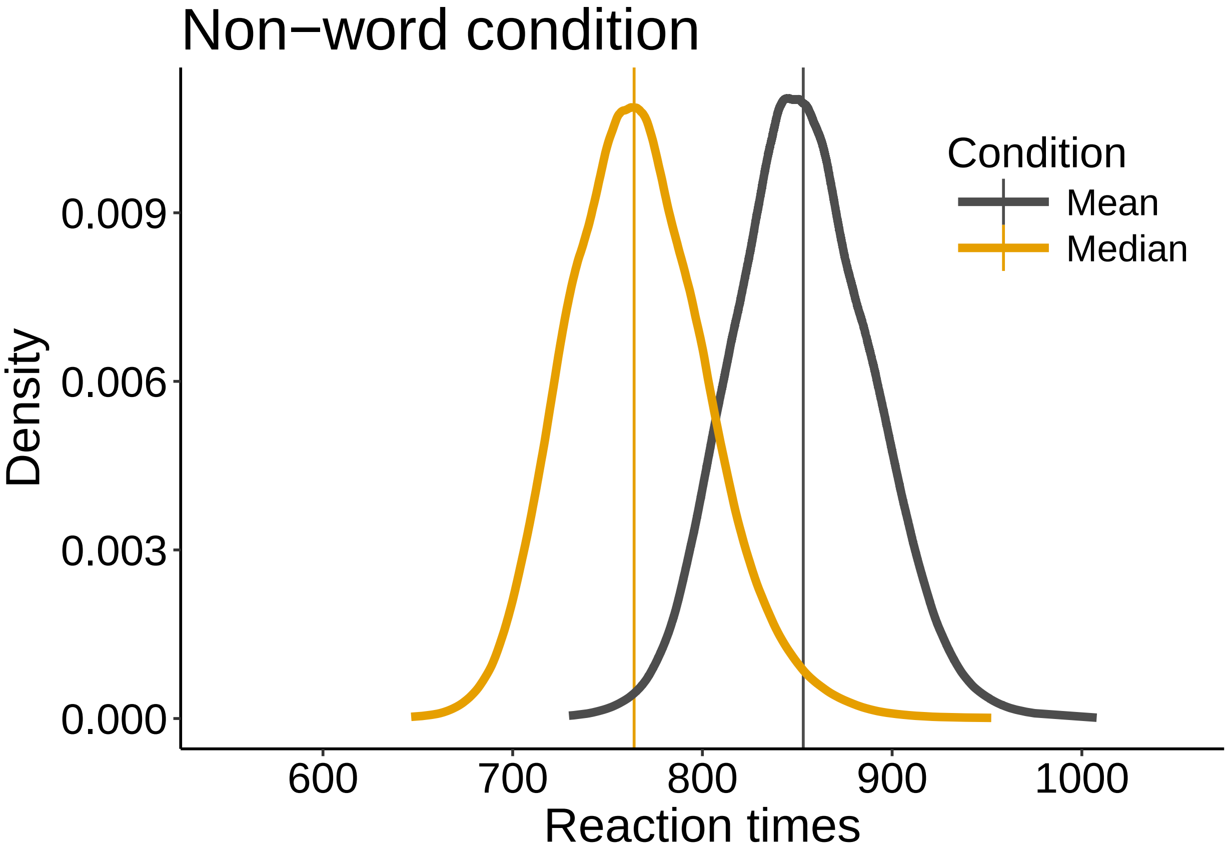
Non-word condition

```
# create data frame
df <- tibble(`x`=c(sim.m.nw, sim.md.nw),
             `y`=c(a.m.nw, a.md.nw),
             `Condition`= rep(c("Mean", "Median"), each = length(sim.m.w)))

# make factor with preserved name order
df$Condition <- as.character(df$Condition)
df$Condition <- factor(df$Condition, levels=unique(df$Condition))

df.pop <- tibble(`x` = c(pop.m.nw, pop.md.nw),
                 `Condition`= factor(c("Mean", "Median")))

# make plot
p <- ggplot(df, aes(x,y)) + theme_classic() +
  # add population values
  geom_vline(data = df.pop, aes(xintercept = x, colour = Condition)) +
  geom_line(aes(colour=Condition), size = 1.5) + # linetype=Condition,
  scale_color_manual(values=c("grey30", "#E69F00")) +
  scale_x_continuous(limits=c(550,1050), breaks=seq(500,1000,100)) +
  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.key.width = unit(1.5,"cm"),
        legend.position = c(0.85,0.8),
        legend.title = element_text(size=16),
        legend.text = element_text(size = 14),
        strip.text.y = element_text(size = 18, face = "bold", angle = 0)) +
        # legend.position = c(0.25,0.9)) +
  labs(x = "Reaction times", y = "Density") +
  ggtitle("Non-word condition")
p
```



```
# save figure
ggsave(filename='./figure_sim_nonword.jpg',width=8,height=5) #path=pathname
```

Non-word - word difference

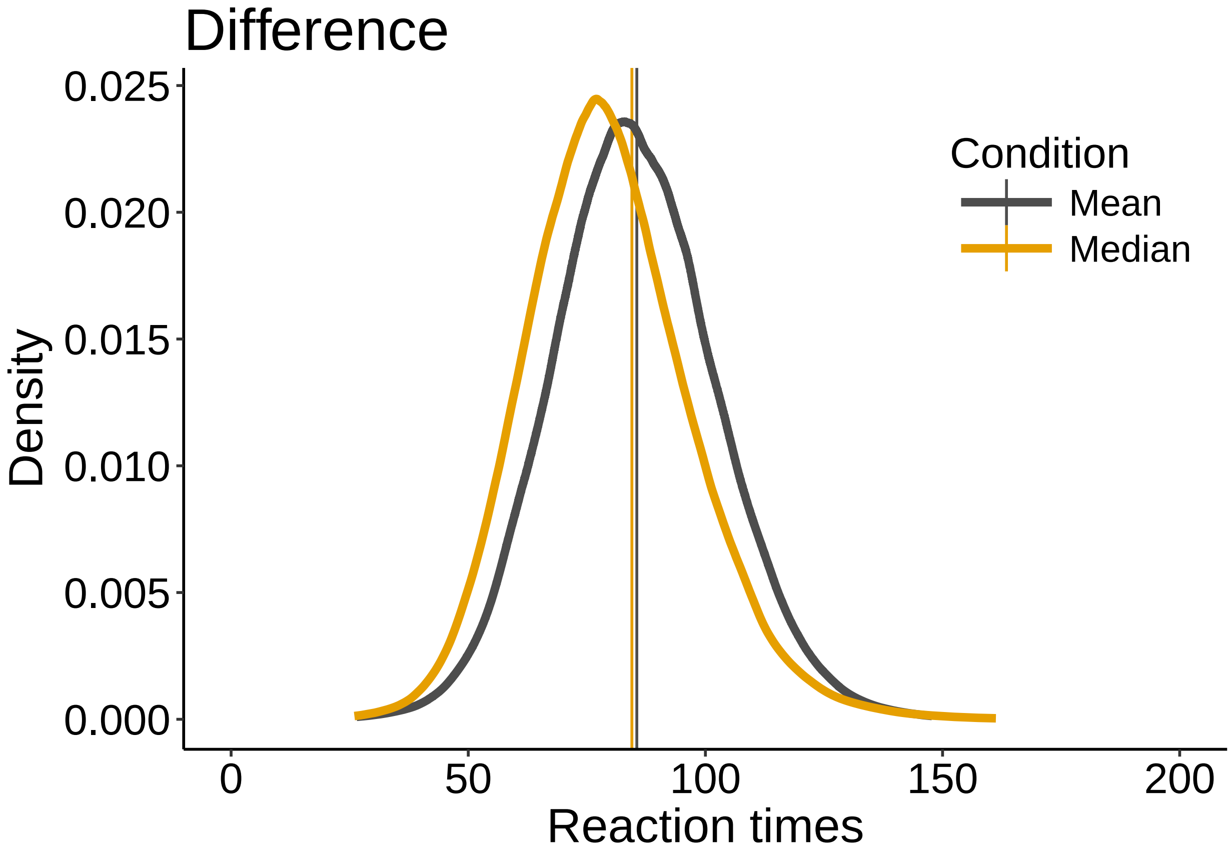
```
# create data frame
df <- tibble(`x`=c(sim.m.diff, sim.md.diff),
             `y`=c(a.m.diff, a.md.diff),
             `Condition`= rep(c("Mean", "Median"), each = length(sim.m.w)))

# make factor with preserved name order
```

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

df.pop <- tibble(`x` = c(pop.m.nw-pop.m.w, pop.md.nw-pop.md.w),
                 `Condition`= factor(c("Mean", "Median")))

# make plot
p <- ggplot(df, aes(x,y)) + theme_classic() +
  # add population values
  geom_vline(data = df.pop, aes(xintercept = x, colour = Condition)) +
  geom_line(aes(colour=Condition), size = 1.5) + # linetype=Condition,
  scale_color_manual(values=c("grey30", "#E69F00")) +
  scale_x_continuous(limits=c(0,200), breaks=seq(0,200,50)) +
  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.key.width = unit(1.5,"cm"),
        legend.position = c(0.85,0.8),
        legend.title = element_text(size=16),
        legend.text = element_text(size = 14),
        strip.text.y = element_text(size = 18, face = "bold", angle = 0)) +
        # legend.position = c(0.25,0.9)) +
  labs(x = "Reaction times", y = "Density") +
  ggtitle("Difference")
p
```



```r
# save figure
ggsave(filename='./figure_sim_difference.jpg',width=8,height=5) #path=pathname
```

Probability of being wrong

```r
mean( (abs(sim.md.diff) - pop.md.diff) >= 5)
```

```
## [1] 0.3878
```

```r
mean( (abs(sim.md.diff) - pop.md.diff) >= 10)
```

```
## [1] 0.2848
```

```r
mean( (abs(sim.md.diff) - pop.md.diff) >= 20)
```

```
## [1] 0.138
```

Highest density intervals

```r
# Mean
m.hdi <- round(HDIofMCMC(sim.m.diff, credMass=0.50), digits = 1)
m.hdi
```

```
## [1] 72.2 95.0
```

```r
# Median
md.hdi <- round(HDIofMCMC(sim.md.diff, credMass=0.50), digits = 1)
md.hdi
```

```
## [1] 65.5 87.8
```

Interval lengths

```r
m.hdi[2] - m.hdi[1]
```

```
## [1] 22.8
```

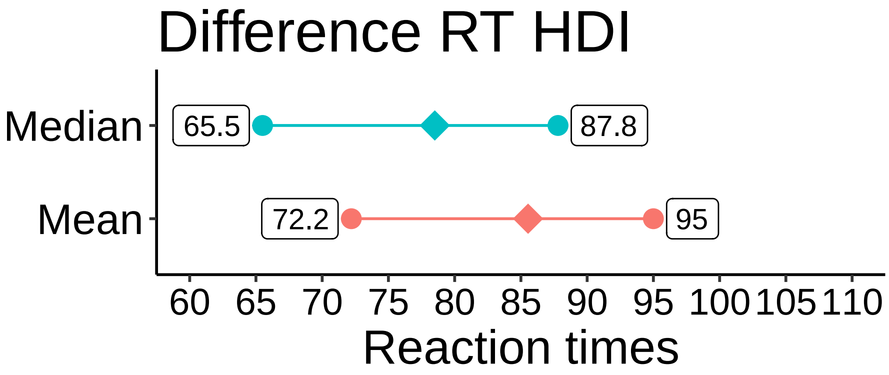
```r
md.hdi[2] - md.hdi[1]
```

```
## [1] 22.3
```

Illustrate results

```r
df <- tibble(x = c(m.hdi, md.hdi),
             y = factor(c(rep("Mean",2), rep("Median",2))),
             label = as.character(c(m.hdi, md.hdi)))
df.pop <- tibble(x = c(pop.m.diff, pop.md.diff),
                 y = c("Mean", "Median"))
df.seg <- tibble(x = c(m.hdi[1], md.hdi[1]),
                 y = c("Mean", "Median"),
                 xend = c(m.hdi[2], md.hdi[2]),
                 yend = c("Mean", "Median"))
df.label1 <- tibble(x = c(m.hdi[1], md.hdi[1]),
             y = c("Mean", "Median"),
             label = as.character(c(m.hdi[1], md.hdi[1])))
df.label2 <- tibble(x = c(m.hdi[2], md.hdi[2]),
             y = c("Mean", "Median"),
             label = as.character(c(m.hdi[2], md.hdi[2])))

ggplot(df, aes(x=x, y=y)) + theme_classic() +
  geom_point(size=3, aes(colour=y)) +
  geom_segment(data=df.seg, aes(x=x, xend=xend, y=y, yend=yend, colour=y)) +
  geom_point(data=df.pop, shape=18, size = 5, aes(colour=y)) +
  # scale_color_viridis(discrete = TRUE) +
  # scale_y_continuous(breaks = pvec) +
  geom_label(data=df.label1, aes(label=label), hjust = "outward", nudge_x = -1) +
  geom_label(data=df.label2, aes(label=label), hjust = "outward", nudge_x = 1) +
  # geom_vline(xintercept = pop.m.diff, colour="red", size=1) +
  scale_x_continuous(breaks = seq(50, 110, 5)) +
  coord_cartesian(xlim=c(60, 110)) +
  theme(plot.title = element_text(size=22),
        axis.title.x = element_text(size = 18),
        axis.text.x = element_text(size = 14, colour="black"),
        axis.text.y = element_text(size = 16, 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)) +
  labs(x = "Reaction times", y = "") +
  ggtitle("Difference RT HDI")
```



```r
# save figure
ggsave(filename = 'figure_sim_difference_hdi.jpg',width=10,height=3) #path=pathname
```