A guide to robust statistical methods in neuroscience: Figure 4

Rand R. Wilcox & Guillaume A. Rousselet

2023-03-02

Contents

Illustrate normal and contaminated normal populations, and sampling distributions for n=30 using three measures of central tendency: mean, 20% trimmed mean and median.

Dependencies

```
library(ggplot2)
library(cowplot)
library(tidyr)
library(tibble)
source("./code/Rallfun-v40.txt")
source("./code/theme_gar.txt")
source("./code/xtrafun.R")
```

Panel A

generate data

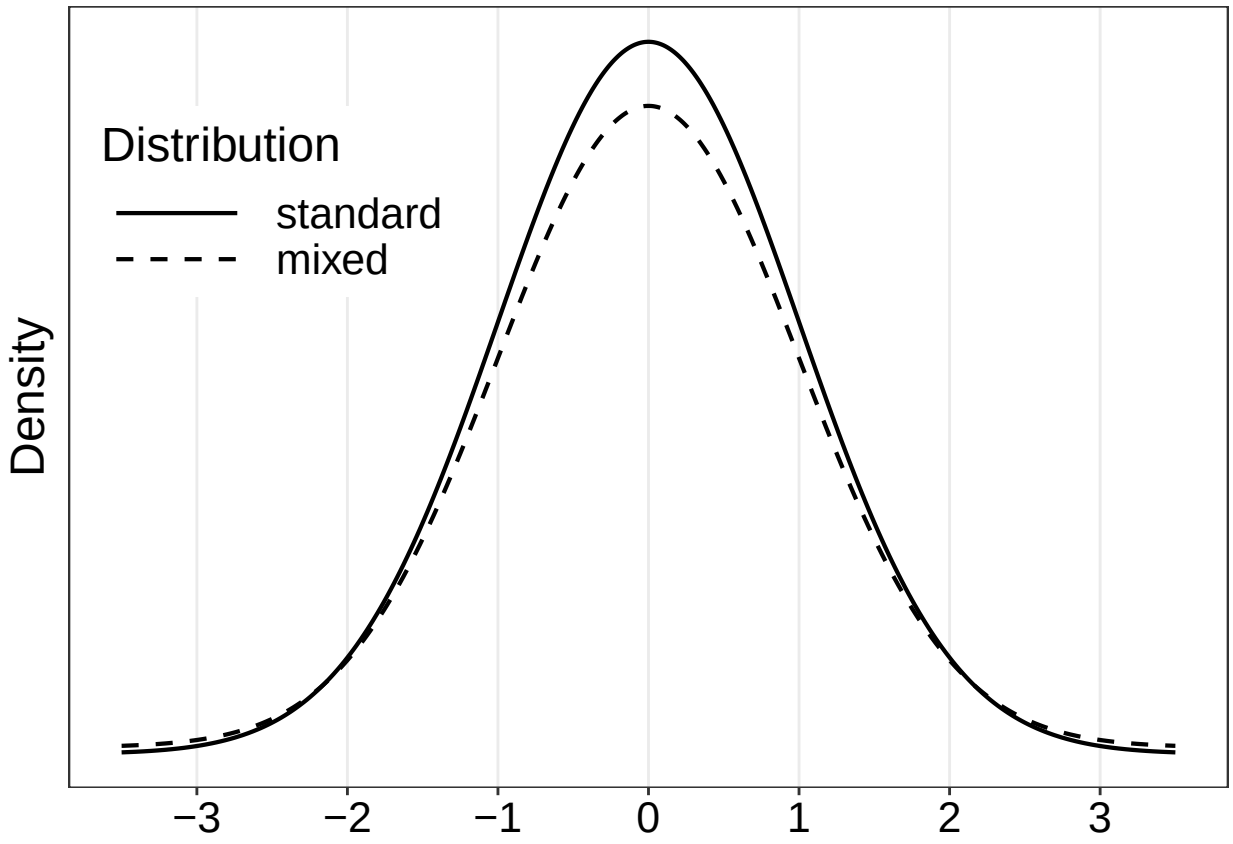
```
x <- seq(-3.5,3.5,.01)
y1 <- dnorm(x)
y2 <- dnorm(x,0,10)
mixed <- .9 * y1 + .1 * y2
standard <- y1
```

create data frame

```
df <- tibble(x,standard,mixed)
df <- tidyr::gather(df,Distribution,y,standard:mixed)
df$Distribution <- as.factor(df$Distribution)
df$Distribution <- keeporder(df$Distribution)
```

make figure

```
panelA <- ggplot(df, aes(x,y, group=Distribution)) + theme_gar +
          geom_line(aes(linetype=Distribution, colour=Distribution), linewidth=0.75) +
          scale_linetype_manual(values=c("solid","dashed")) +
          scale_color_manual(values=c('black','black')) +
          scale_x_continuous(breaks=seq(-3,3,1)) +
          theme(axis.title.x = element_blank(),
                axis.text = element_text(size = 16),
                legend.key.width = unit(2,"cm"),
                legend.position = c(0.175, 0.75),
                axis.title.y = element_text(size = 18),
                panel.grid.minor = element_blank(),
                panel.grid.major.y = element_blank(),
                axis.text.y = element_blank(),
                axis.ticks.y = element_blank()) +
          ylab("Density")
panelA
```



Panel B - samples from normal distribution ————-

generate data

```
set.seed(45)

n <- 100000 # number of samples / experiments
v1 <- vector(mode = "numeric", length = n)
v2 <- vector(mode = "numeric", length = n)
v3 <- vector(mode = "numeric", length = n)

for(i in 1:n){
  x <- rnorm(30) # samples from normal population
  v1[i] <- mean(x)
  v2[i] <- tmean(x)
  v3[i] <- median(x)
}
```

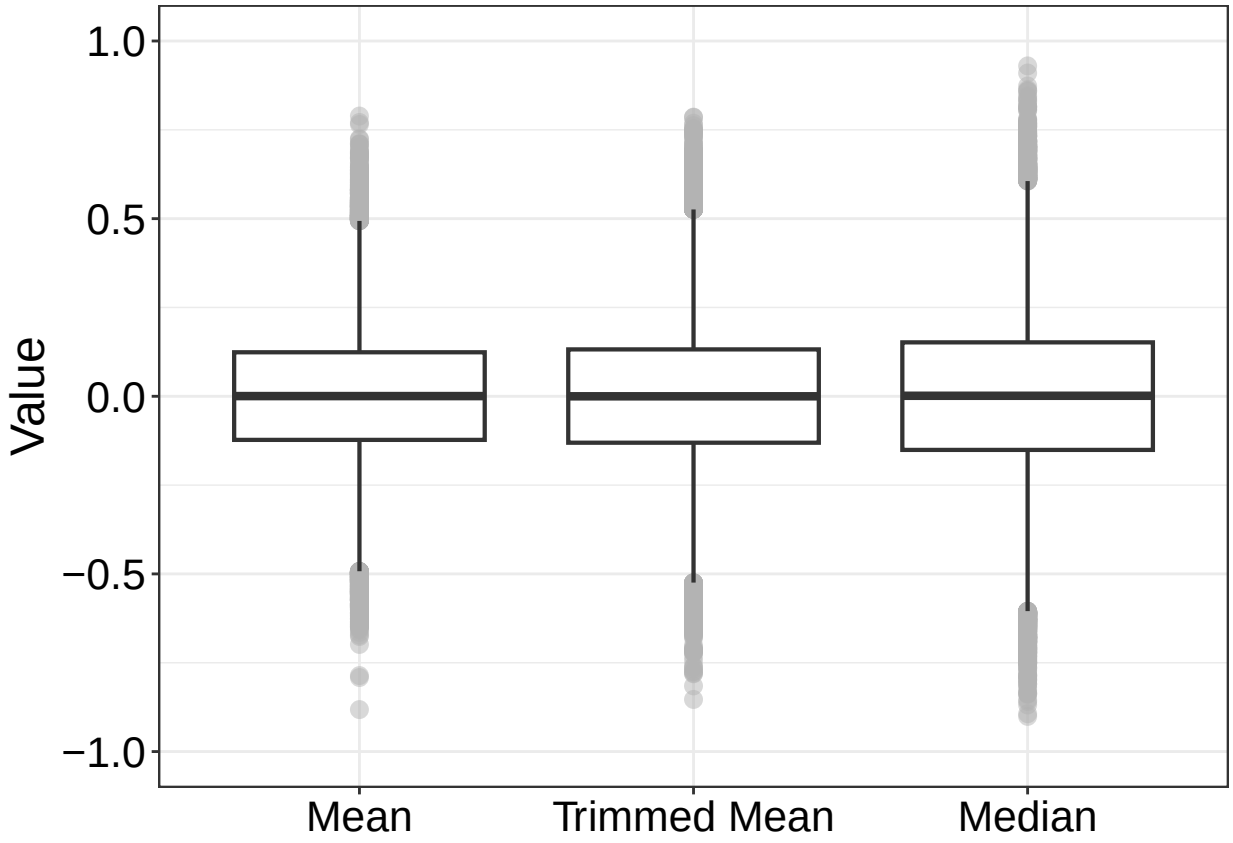
make data frame

```
df <- tibble('Mean'=v1, 'Trimmed Mean'=v2, 'Median'=v3)
df <- tidyr::gather(df, Estimator, Value, c(1:3))
```

```
df$Estimator <- as.factor(df$Estimator)
df$Estimator <- keeporder(df$Estimator)
```

make figure – boxplot version

```
panelB <- ggplot(df, aes(Estimator, Value)) + theme_gar +
          geom_boxplot(outlier.colour = "grey70", outlier.shape = 16,
            outlier.size = 3, outlier.alpha = .5, size = 0.75) +
          scale_y_continuous(breaks=seq(-1,1,0.5), limits = c(-1,1)) +
          theme(axis.title.x = element_blank(),
          axis.title.y = element_text(size = 18),
          axis.text = element_text(size = 16))
panelB
```

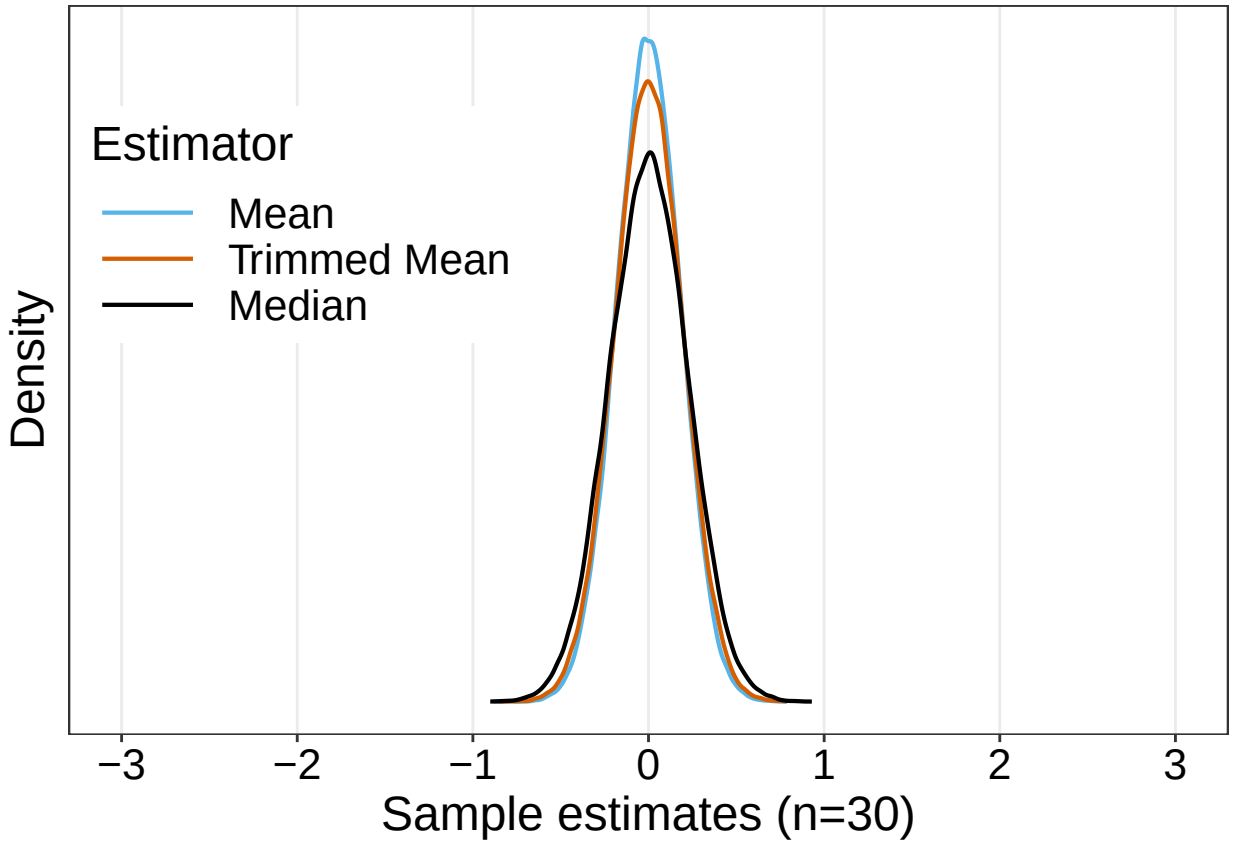


make figure – kernel density version

```
panelB <- ggplot(df, aes(x = Value, colour = Estimator)) + theme_gar +
          stat_density(geom="line", position="identity",
                       linewidth=0.75, trim=TRUE) +
          scale_colour_manual(values = c("#56B4E9", "#D55E00", "black")) + #009E73
          scale_x_continuous(breaks=seq(-4,4,1), limits = c(-3,3)) +
          theme(axis.title = element_text(size = 18),
          axis.text = element_text(size = 16),
```

```
          panel.grid.minor = element_blank(),
          panel.grid.major.y = element_blank(),
          axis.text.y = element_blank(),
          axis.ticks.y = element_blank(),
          legend.position = c(.2, .7)) +
          # plot.title = element_text(vjust = -7, hjust = 0.025),
          # plot.margin = unit(c(5.5, 5.5, 5.5, 5.5), "pt")
          xlab("Sample estimates (n=30)") +
          ylab("Density")
          # annotate("text", x=-2.2, y=2.2, label="Normal population", size=8)

panelB
```



Panel C - samples from contaminated normal distribution

generate data

```
set.seed(45)

n <- 100000 # number of samples / experiments
v1 <- vector(mode = "numeric", length = n)
v2 <- vector(mode = "numeric", length = n)
v3 <- vector(mode = "numeric", length = n)
```

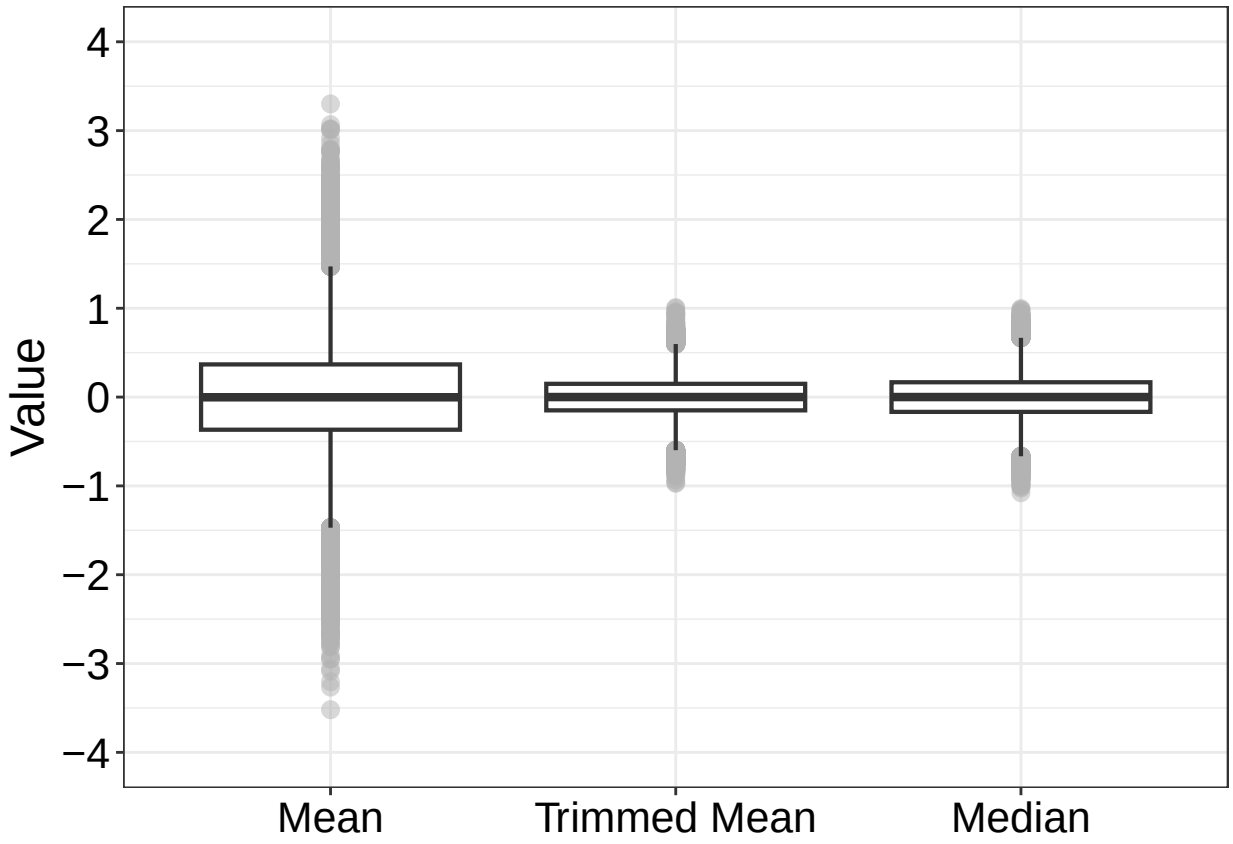
```
for(i in 1:n){
  x <- cnorm(30) # samples from mixed population
  v1[i] <- mean(x)
  v2[i] <- tmean(x)
  v3[i] <- median(x)
}
```

make data frame

```
df <- tibble('Mean'=v1, 'Trimmed Mean'=v2, 'Median'=v3)
df <- tidyr::gather(df, Estimator, Value, c(1:3))
df$Estimator <- as.factor(df$Estimator)
df$Estimator <- keeporder(df$Estimator)
```

make figure – boxplot version

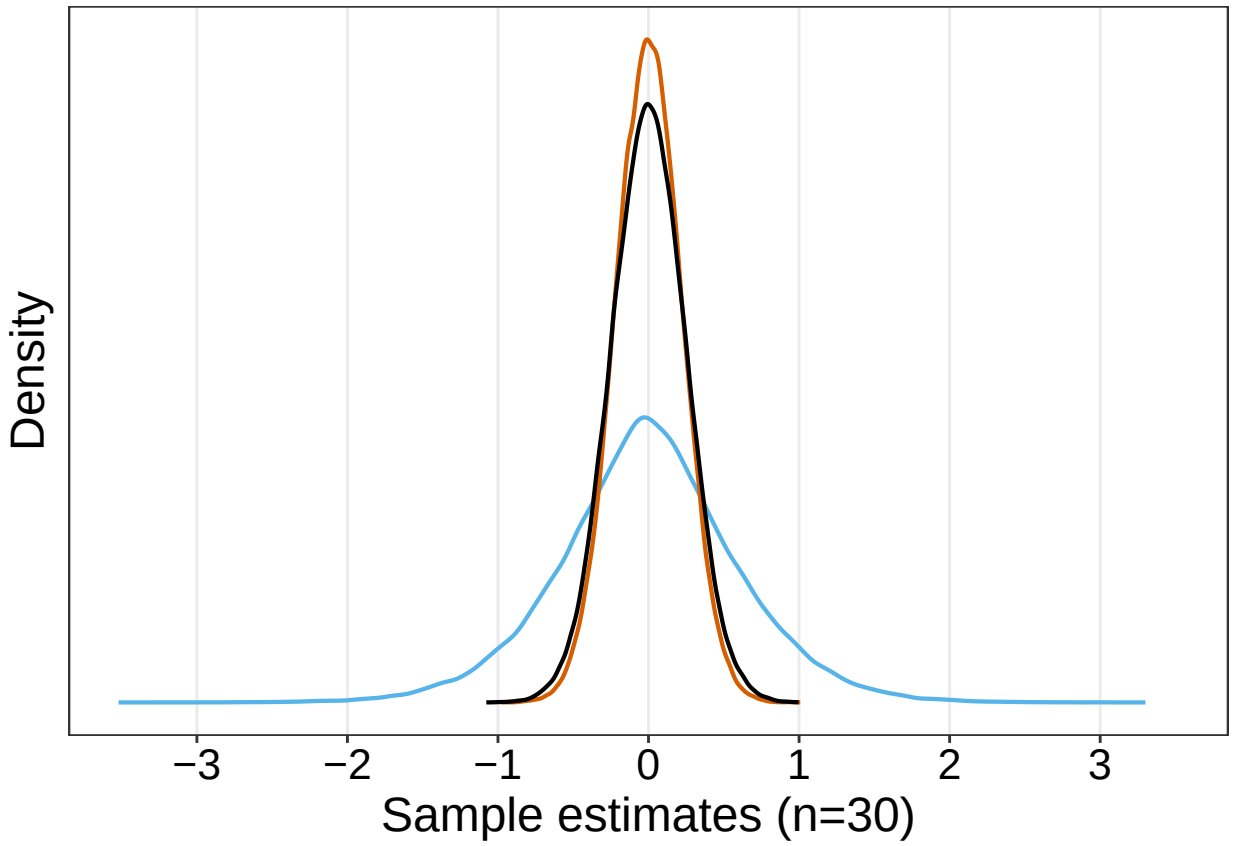
```
panelC <- ggplot(df, aes(Estimator, Value)) + theme_gar +
          geom_boxplot(outlier.colour = "grey70", outlier.shape = 16,
            outlier.size = 3, outlier.alpha = .5, size = 0.75) +
          scale_y_continuous(breaks=seq(-4,4,1), limits = c(-4,4)) +
          theme(axis.title.x = element_blank(),
          axis.title.y = element_text(size = 18),
          axis.text = element_text(size = 16))
panelC
```



make figure – kernel density version

```
panelC <- ggplot(df, aes(x = Value, colour = Estimator)) + theme_gar +
          stat_density(geom="line", position="identity",
                       linewidth=0.75, trim=TRUE, show.legend = FALSE) +
          scale_colour_manual(values = c("#56B4E9", "#D55E00", "black")) + #009E73
          scale_x_continuous(breaks=seq(-4,4,1)) +
          coord_cartesian(xlim = c(-3.5,3.5)) +
          theme(axis.title = element_text(size = 18),
          axis.text = element_text(size = 16),
          panel.grid.minor = element_blank(),
          panel.grid.major.y = element_blank(),
          axis.text.y = element_blank(),
          axis.ticks.y = element_blank()) +
          xlab("Sample estimates (n=30)") +
          ylab("Density")
          # annotate("text", x=-2.4, y=2.2, label="Mixed population", size=8)

panelC
```



combine panels into one figure

```
cowplot::plot_grid(panelA, panelB, panelC,
                   labels=c("A", "B", "C"),
                   ncol = 1,
                   nrow = 3,
                   rel_heights = c(1, 1, 1),
                   label_size = 20,
                   hjust = -0.5,
                   scale=.95,
                   align = "v")
# save figure
ggsave(filename='./figures/figure4.pdf',width=7,height=9)
```