Validation of effect size estimates for One-Way ANOVA

We first repeat the simulation by Brysbaert:

```
# Simulations to estimate the power of an ANOVA with three unrelated groups
# the effect between the two extreme groups is set to d = .4, the effect for the third group is d = .4
# we use the built-in aov-test command

# give sample sizes (all samples sizes are equal)
N = 90

# give effect size d
d1 = .4 #difference between the extremes
d2 = .4 #third condition goes with the highest extreme

# give number of simulations
nSim = nsims

# give alpha levels
alpha1 = .05 #alpha level for the omnibus ANOVA
alpha2 = .05 #alpha level for three post hoc one-tailed t-tests Bonferroni correction

# create progress bar in case it takes a while
#pb <- winProgressBar(title = "progress bar", min = 0, max = nSim, width = 300)

# create vectors to store p-values
p1 <-numeric(nSim) #p-value omnibus ANOVA
p2 <-numeric(nSim) #p-value first post hoc test
p3 <-numeric(nSim) #p-value second post hoc test
p4 <-numeric(nSim) #p-value third post hoc test
pes1 <-numeric(nSim) #partial eta-squared
pes2 <-numeric(nSim) #partial eta-squared two extreme conditions


library(lsr)

for(i in 1:nSim){ #for each simulated experiment
 # setWinProgressBar(pb, i, title=paste(round(i/nSim*100, 1), "% done"))
  x<-rnorm(n = N, mean = 0, sd = 1)
  y<-rnorm(n = N, mean = d1, sd = 1)
  z<-rnorm(n = N, mean = d2, sd = 1)
  data = c(x,y,z)
  groups= factor(rep(letters[24:26], each = N))
  test <- aov(data~groups)
  pes1[i] <- etaSquared(test)[1,2]
  p1[i] <- summary(test)[[1]][["Pr(>F)"]][[1]]
  p2[i] <- t.test(x,y)$p.value
  p3[i] <- t.test(x,z)$p.value
  p4[i] <- t.test(y,z)$p.value
  data = c(x,y)
  groups= factor(rep(letters[24:25], each = N))
  test <- aov(data~groups)
  pes2[i] <- etaSquared(test)[1,2]
  }
```

```r
#close(pb)#close progress bar

# results are as predicted when omnibus ANOVA is significant, t-tests are significant between x and y p
#printing all unique tests (adjusted code by DL)
sum(p1<alpha1)/nSim
```

```
## [1] 0.79336
```

```r
sum(p2<alpha2)/nSim
```

```
## [1] 0.76036
```

```r
sum(p3<alpha2)/nSim
```

```
## [1] 0.76178
```

```r
sum(p4<alpha2)/nSim
```

```
## [1] 0.04976
```

```r
mean(pes1)
```

```
## [1] 0.04141885
```

```r
mean(pes2)
```

```
## [1] 0.04356374
```

Installation

We install the functions:

```r
# Install the two functions from GitHub by running the code below:

source("https://raw.githubusercontent.com/Lakens/ANOVA_power_simulation/master/ANOVA_design.R")
source("https://raw.githubusercontent.com/Lakens/ANOVA_power_simulation/master/ANOVA_power.R")
```

Three conditions replication

```r
K <- 3
mu <- c(0, 0.4, 0.4)
n <- 90
sd <- 1
r <- 0
string = paste(K,"b",sep="")
```

```r
design_result <- ANOVA_design(string = string,
                   n = n,
                   mu = mu,
                   sd = sd,
                   r = r,
                   p_adjust = "none",
                   labelnames = c("factor1", "level1", "level2", "level3"))
```

```r
ANOVA_power(design_result, nsims = nsims)
```

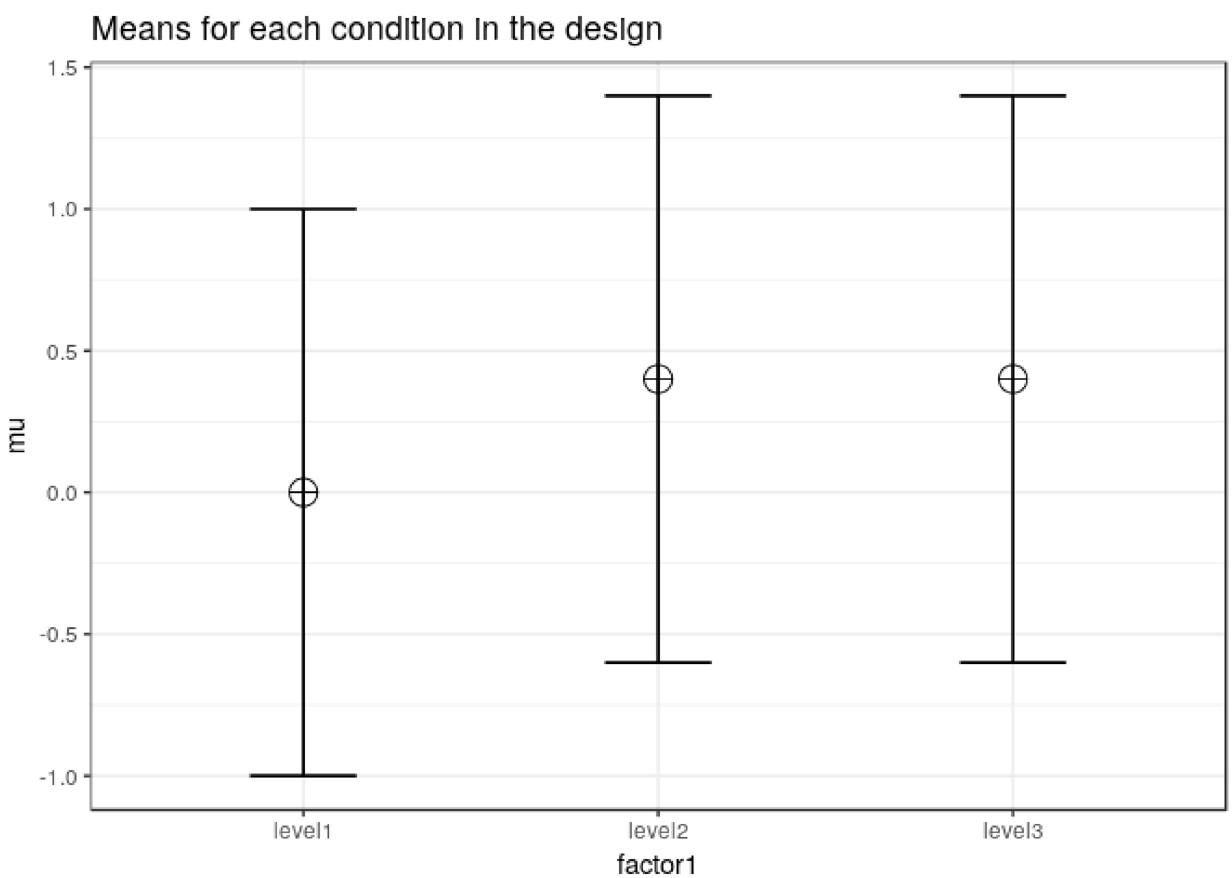
Means for each condition in the design
1.5
1.0
0.5
0.0
-0.5
-1.0
mu
level1
level2
level3
factor1

Figure 1:

```
## Power and Effect sizes for ANOVA tests
##                power effect size
## anova_factor1 79.127      0.0383
##
## Power and Effect sizes for contrasts
##                                  power effect size
## p_factor1_level1_factor1_level2 75.843      0.4010
## p_factor1_level1_factor1_level3 76.039      0.4018
## p_factor1_level2_factor1_level3  5.030      0.0007
```

Variation 1

```r
# give sample sizes (all samples sizes are equal)
N = 145

# give effect size d
d1 = .4 #difference between the extremes
d2 = .0 #third condition goes with the highest extreme

# give number of simulations
nSim = nsims

# give alpha levels
alpha1 = .05 #alpha level for the omnibus ANOVA
alpha2 = .05 #alpha level for three post hoc one-tailed t-tests Bonferroni correction

# create progress bar in case it takes a while
#pb <- winProgressBar(title = "progress bar", min = 0, max = nSim, width = 300)

# create vectors to store p-values
p1 <-numeric(nSim) #p-value omnibus ANOVA
p2 <-numeric(nSim) #p-value first post hoc test
p3 <-numeric(nSim) #p-value second post hoc test
p4 <-numeric(nSim) #p-value third post hoc test
pes1 <-numeric(nSim) #partial eta-squared
pes2 <-numeric(nSim) #partial eta-squared two extreme conditions


library(lsr)

for(i in 1:nSim){ #for each simulated experiment
 # setWinProgressBar(pb, i, title=paste(round(i/nSim*100, 1), "% done"))
  x<-rnorm(n = N, mean = 0, sd = 1)
  y<-rnorm(n = N, mean = d1, sd = 1)
  z<-rnorm(n = N, mean = d2, sd = 1)
  data = c(x,y,z)
  groups= factor(rep(letters[24:26], each = N))
  test <- aov(data~groups)
  pes1[i] <- etaSquared(test)[1,2]
  p1[i] <- summary(test)[[1]][["Pr(>F)"]][[1]]
  p2[i] <- t.test(x,y)$p.value
  p3[i] <- t.test(x,z)$p.value
```

```
  p4[i] <- t.test(y,z)$p.value
  data = c(x,y)
  groups= factor(rep(letters[24:25], each = N))
  test <- aov(data~groups)
  pes2[i] <- etaSquared(test)[1,2]
  }
#close(pb)#close progress bar

# results are as predicted when omnibus ANOVA is significant, t-tests are significant between x and y p
#printing all unique tests (adjusted code by DL)
sum(p1<alpha1)/nSim
```

```
## [1] 0.94865
```

```
sum(p2<alpha2)/nSim
```

```
## [1] 0.92378
```

```
sum(p3<alpha2)/nSim
```

```
## [1] 0.04959
```

```
sum(p4<alpha2)/nSim
```

```
## [1] 0.92564
```

```
mean(pes1)
```

```
## [1] 0.03880623
```

```
mean(pes2)
```

```
## [1] 0.0416915
```

Three conditions replication

```
K <- 3
mu <- c(0, 0.4, 0.0)
n <- 145
sd <- 1
r <- 0
string = paste(K,"b",sep="")
```

```
design_result <- ANOVA_design(string = string,
                   n = n,
                   mu = mu,
                   sd = sd,
                   r = r,
                   p_adjust = "none",
                   labelnames = c("factor1", "level1", "level2", "level3"))
```

```
ANOVA_power(design_result, nsims = nsims)
```

```
## Power and Effect sizes for ANOVA tests
##                power effect size
## anova_factor1 94.948      0.0367
##
```

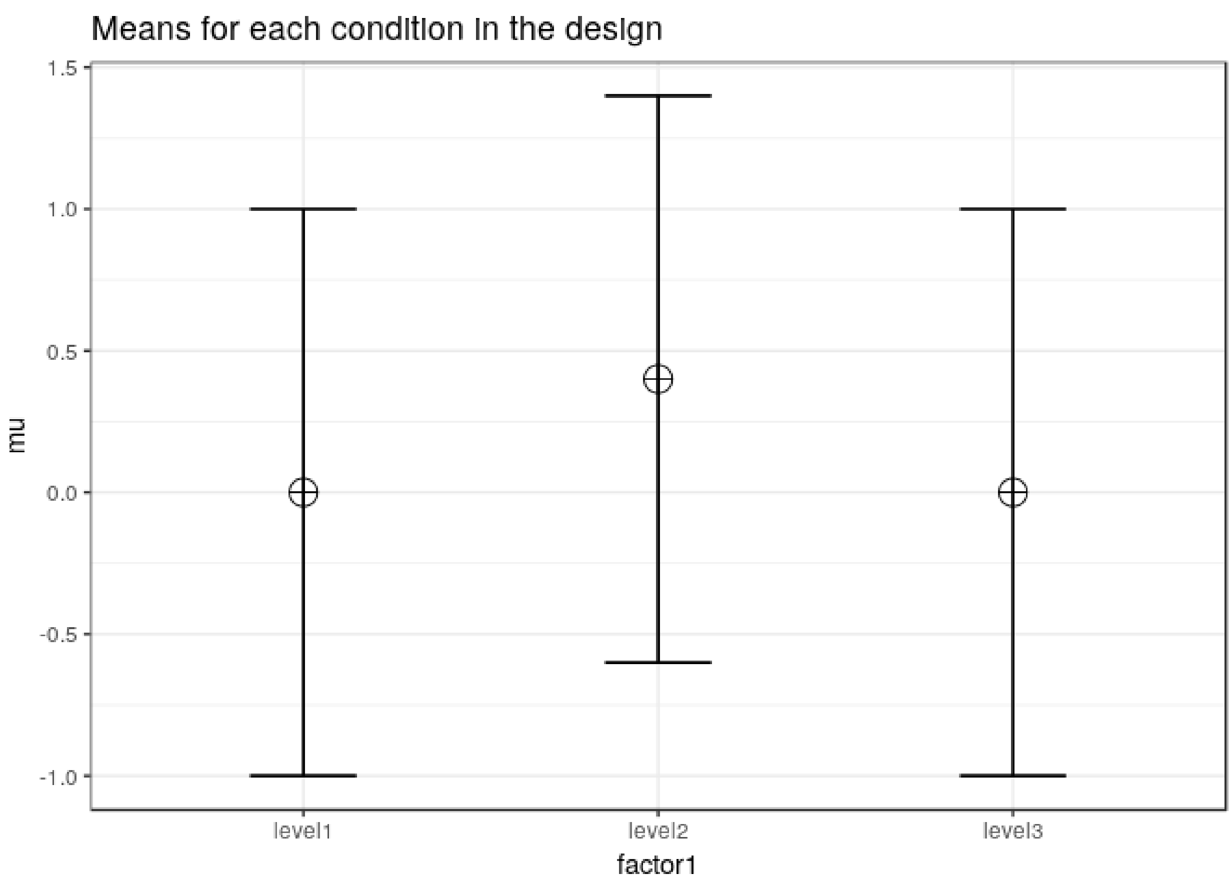
Means for each condition in the design
1.5
1.0
0.5
0.0
-0.5
-1.0
mu
level1
level2
level3
factor1

Figure 2:

```
## Power and Effect sizes for contrasts
##                                  power effect size
## p_factor1_level1_factor1_level2 92.448      0.4010
## p_factor1_level1_factor1_level3  4.925      0.0002
## p_factor1_level2_factor1_level3 92.433     -0.4008
```

Variation 2

```
# give sample sizes (all samples sizes are equal)
N = 82

# give effect size d
d1 = .4 #difference between the extremes
d2 = .2 #third condition goes with the highest extreme

# give number of simulations
nSim = nsims

# give alpha levels
alpha1 = .05 #alpha level for the omnibus ANOVA
alpha2 = .05 #alpha level for three post hoc one-tailed t-tests Bonferroni correction

# create progress bar in case it takes a while
#pb <- winProgressBar(title = "progress bar", min = 0, max = nSim, width = 300)

# create vectors to store p-values
p1 <-numeric(nSim) #p-value omnibus ANOVA
p2 <-numeric(nSim) #p-value first post hoc test
p3 <-numeric(nSim) #p-value second post hoc test
p4 <-numeric(nSim) #p-value third post hoc test
pes1 <-numeric(nSim) #partial eta-squared
pes2 <-numeric(nSim) #partial eta-squared two extreme conditions


library(lsr)

for(i in 1:nSim){ #for each simulated experiment
 # setWinProgressBar(pb, i, title=paste(round(i/nSim*100, 1), "% done"))
  x<-rnorm(n = N, mean = 0, sd = 1)
  y<-rnorm(n = N, mean = d1, sd = 1)
  z<-rnorm(n = N, mean = d2, sd = 1)
  data = c(x,y,z)
  groups= factor(rep(letters[24:26], each = N))
  test <- aov(data~groups)
  pes1[i] <- etaSquared(test)[1,2]
  p1[i] <- summary(test)[[1]][["Pr(>F)"]][[1]]
  p2[i] <- t.test(x,y)$p.value
  p3[i] <- t.test(x,z)$p.value
  p4[i] <- t.test(y,z)$p.value
  data = c(x,y)
  groups= factor(rep(letters[24:25], each = N))
  test <- aov(data~groups)
```

```r
  pes2[i] <- etaSquared(test)[1,2]
  }
#close(pb)#close progress bar

# results are as predicted when omnibus ANOVA is significant, t-tests are significant between x and y p
#printing all unique tests (adjusted code by DL)
sum(p1<alpha1)/nSim
```

```
## [1] 0.62089
```

```r
sum(p2<alpha2)/nSim
```

```
## [1] 0.72261
```

```r
sum(p3<alpha2)/nSim
```

```
## [1] 0.24924
```

```r
sum(p4<alpha2)/nSim
```

```
## [1] 0.24558
```

```r
mean(pes1)
```

```
## [1] 0.03396681
```

```r
mean(pes2)
```

```
## [1] 0.0443309
```

Three conditions replication

```r
K <- 3
mu <- c(0, 0.4, 0.2)
n <- 82
sd <- 1
r <- 0
string = paste(K,"b",sep="")
```

```r
design_result <- ANOVA_design(string = string,
                   n = n,
                   mu = mu,
                   sd = sd,
                   r = r,
                   p_adjust = "none",
                   labelnames = c("factor1", "level1", "level2", "level3"))
```

```r
ANOVA_power(design_result, nsims = nsims)
```

```
## Power and Effect sizes for ANOVA tests
##                power effect size
## anova_factor1 61.77      0.0303
##
## Power and Effect sizes for contrasts
##                                   power effect size
## p_factor1_level1_factor1_level2 71.944      0.4014
## p_factor1_level1_factor1_level3 24.548      0.2005
```

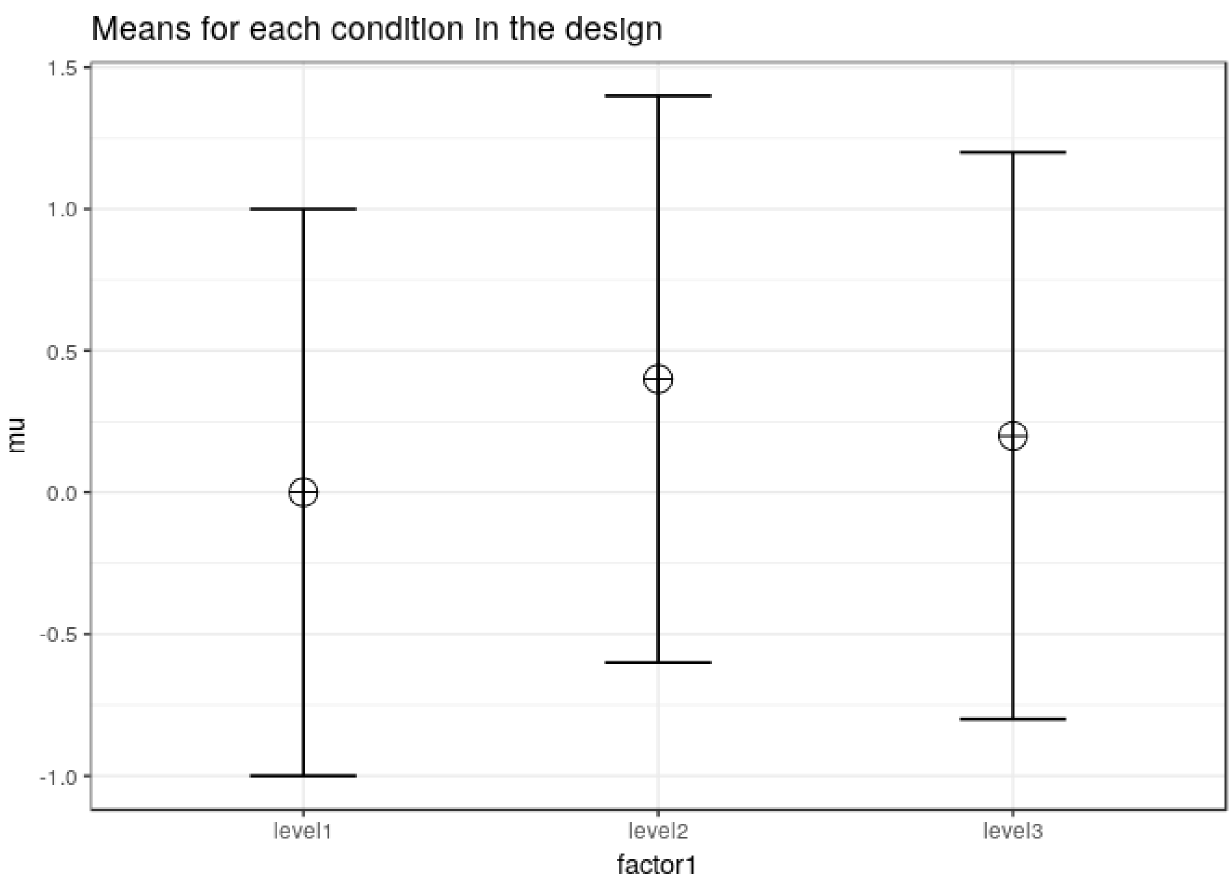
Means for each condition in the design
mu
1.5
1.0
0.5
0.0
-0.5
-1.0
level1
level2
level3
factor1

Figure 3:

```
## p_factor1_level2_factor1_level3 24.667     -0.2010
```