Sample size calculation based on simulated data

for the study: The relationship between eating and concentration

Christoph Bamberg

using Superpower package by Lakens and colleagues

bookdown version of documentation: https://aaroncaldwell.us/SuperpowerBook/

```r
library(Superpower)
```

specifying inputs to *ANOVA_design* function

need to find values for design, n, mu, sd, r, labelnames *design*: either tested hungry or satiated and with "hungry_good" or "full_good" manipulation. In a counterbalanced 2x2 between-groups design, no within group levels.

means: specifying the means of each group in a vector. The order matters. Setting the order to 1. hungry / "hungry_good" 2. hungry / "full_good" 3. satiated / "hungry_good" 4. satiated / "full_good" -> do labelnames accordingly

labelnames: syntax: (factor1, factor1_level1, factor1_level2, factor2, factor2_level1, factor2_level2)

standard deviation: for now assuming homogenous variance in all groups -> one value. Otherwise vector according to same ordering as mu vector. Since I am using small values for mu (around 1), the standard deviation should also not be too large. A value around 0.5 seems good.

sample size: fixed for now, later varied to get a power curve. Specified per group - n=20 means 20*4=80 subjects in total

```r
#design:
# "b" for between, "w" for within, "*" to combine
design="2b*2b"

#labelnames:
labelnames=c("hunger_manipulation", "hugnry", "satiated", "expectation", "hungry_good", "full_good")

# correlations
# zero for between subject designs
r=0

# standard deviation:
sd=0.5
```

```r
# sample size:
n=30
```

different theoretical approaches to specifying mu

simply setting mu to arbitrary values

```r
#means:
# 2*2= 4 means to be specified
mu_test=c(1,2,3,4)#to see whether I correctly ordered the labels
#mu=c(1,-1,-1,1) #if only congurency between conditions had an effect
mu=c(1,-1,0,2) #if satiated performance is better + congruency
mu=c(1,0.5,0.8,1.2) #smaller differences for both interventions
```

a more informed approach

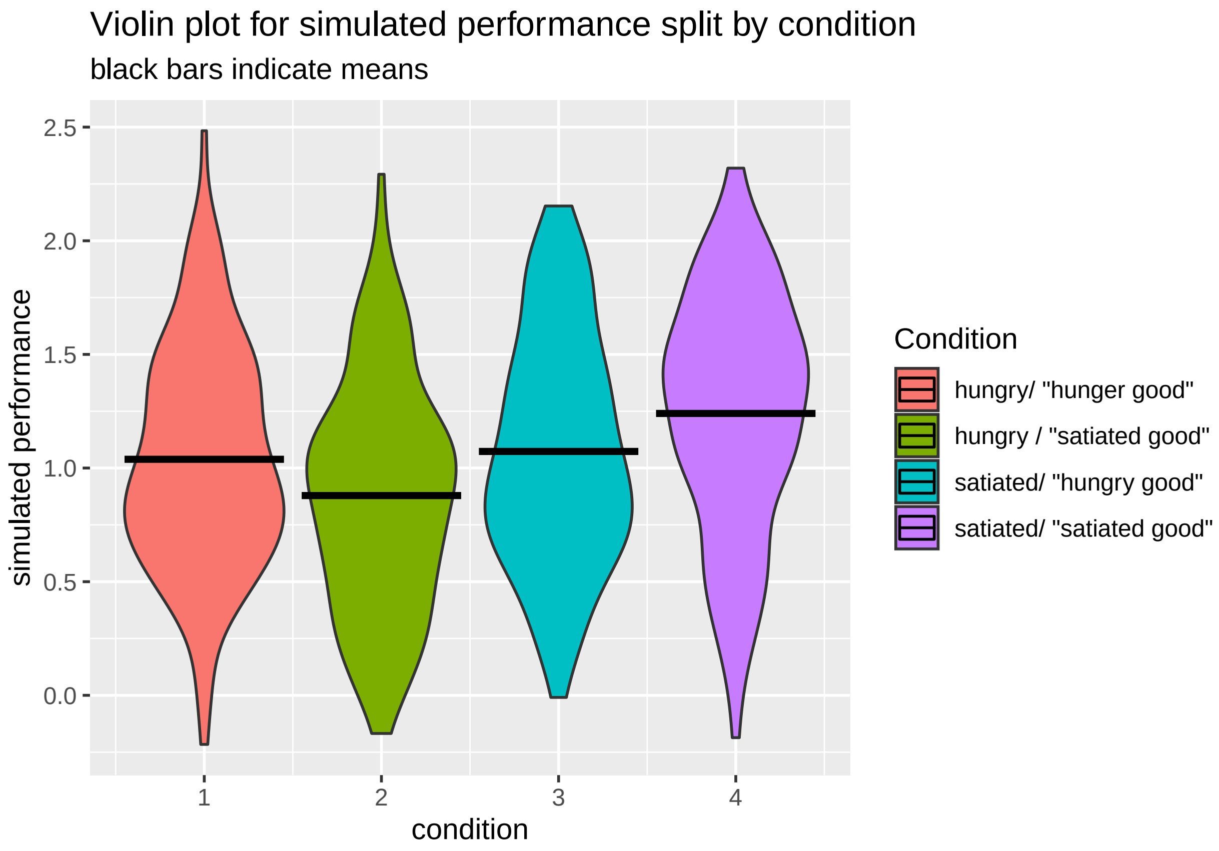
procedure: 1. define wanted difference, d_hunger, for hunger intervention 2. specify means for each level of the hunger intervention based on d_hunger 3. define difference between the expectations 4. add the expectation difference on top of the hunger difference

```r
#### specifying mus based on effect size
#with d= (mu_fed - mu_hungry)/sd
# (assuming same sd for all)

# fixing the difference between hungry / satiated a prior
d_hunger=0.3

mu_hungry=1
#solving for mu_fed:
mu_full=d_hunger *sd + mu_hungry


# now adding the effect of expectations
d_expectation=0.15

mu_hungry_hgood=mu_hungry+d_expectation*sd
mu_hungry_fgood=mu_hungry-d_expectation*sd
mu_full_hgood=mu_full-d_expectation*sd
mu_full_fgood=mu_full+d_expectation*sd
mu=c(mu_hungry_hgood,mu_hungry_fgood,mu_full_hgood,mu_full_fgood)
print(mu)
```

```
## [1] 1.075 0.925 1.075 1.225
```

plotting the assumed values for mu by simulating draws from a normal distribution

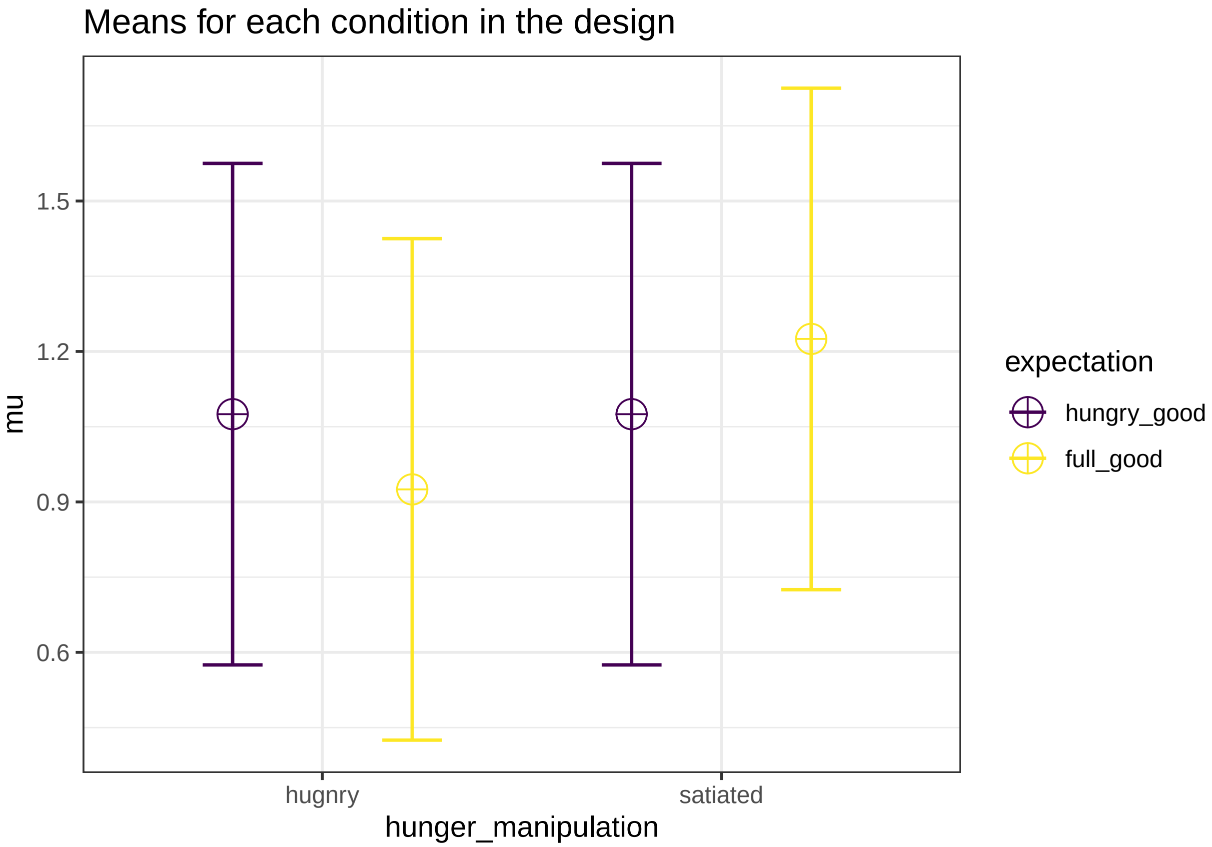
```r
set.seed(345)

N=120 #the required sample size, see below

sim_hungry_hgood<-rnorm(N,mean=mu_hungry_hgood,sd=sd)
sim_hungry_fgood<-rnorm(N,mean=mu_hungry_fgood,sd=sd)
sim_full_hgood<-rnorm(N,mean=mu_full_hgood,sd=sd)
sim_full_fgood<-rnorm(N,mean=mu_full_fgood,sd=sd)
condition<-c(rep(1,length.out=N),rep(2,length.out=N),rep(3,length.out=N),rep(4,length.out=N))
condition<-as.factor(condition)
sim_perf<-c(sim_hungry_hgood,sim_hungry_fgood,sim_full_hgood,sim_full_fgood)
#sim_perf<-as.numeric(sim_perf)
DF<-data.frame(cbind(condition,sim_perf))
#View(DF)


library(ggplot2)
ggplot(data=DF, aes(x=condition , y=sim_perf, fill=factor(condition)))+
  geom_violin(scale="area")+
  stat_summary(fun="mean",
               geom="crossbar") +
  labs(x="condition",y="simulated performance",title="Violin plot for simulated performance split by co
```



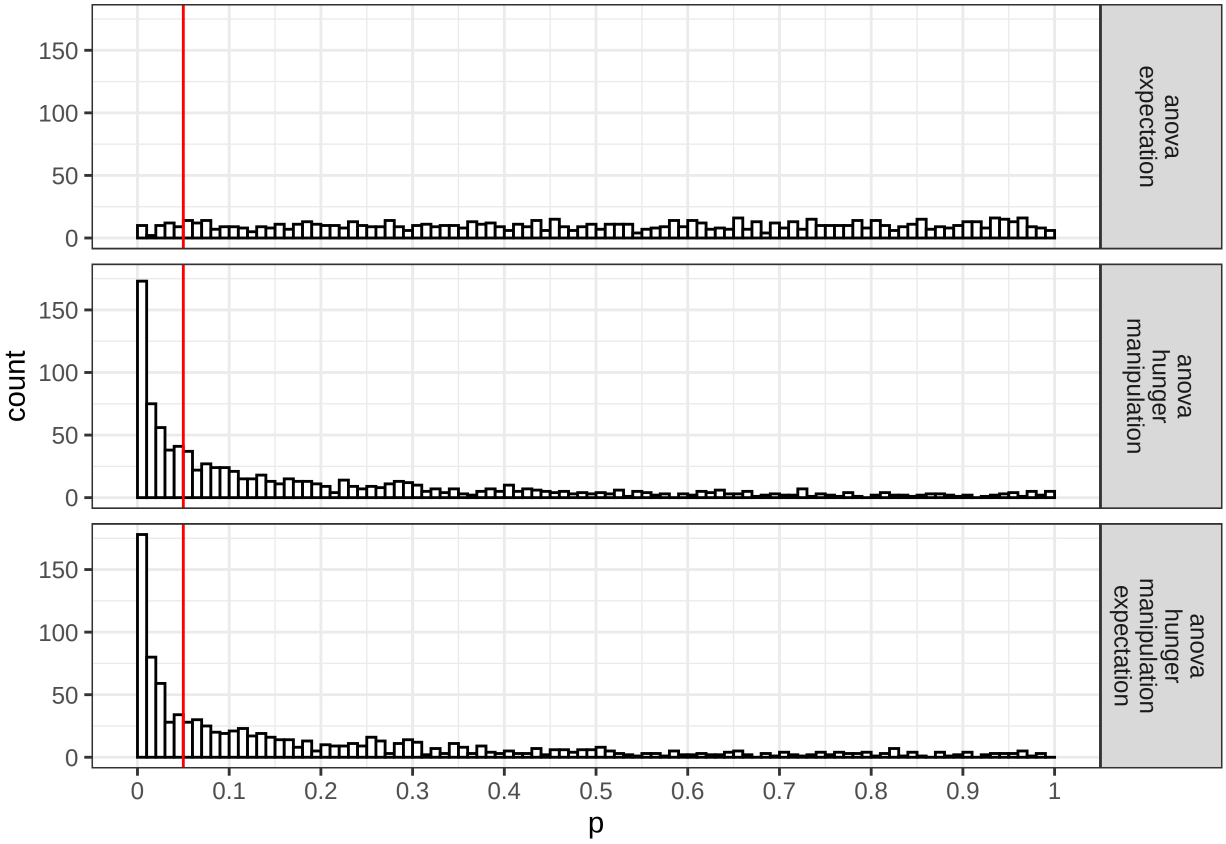
```
# different plot
#plot(density(sim_hungry_hgood),col="red",lwd=3)
#lines(density(sim_hungry_fgood),col="yellow",lwd=3)
#lines(density(sim_full_hgood),col="green",lwd=3)
#lines(density(sim_full_fgood),col="blue",lwd=3)
```

entering the values into design function:

```
design_result<-ANOVA_design(design = design,
                   n = n,
                   mu = mu,
                   sd = sd,
                   labelnames = labelnames)
```



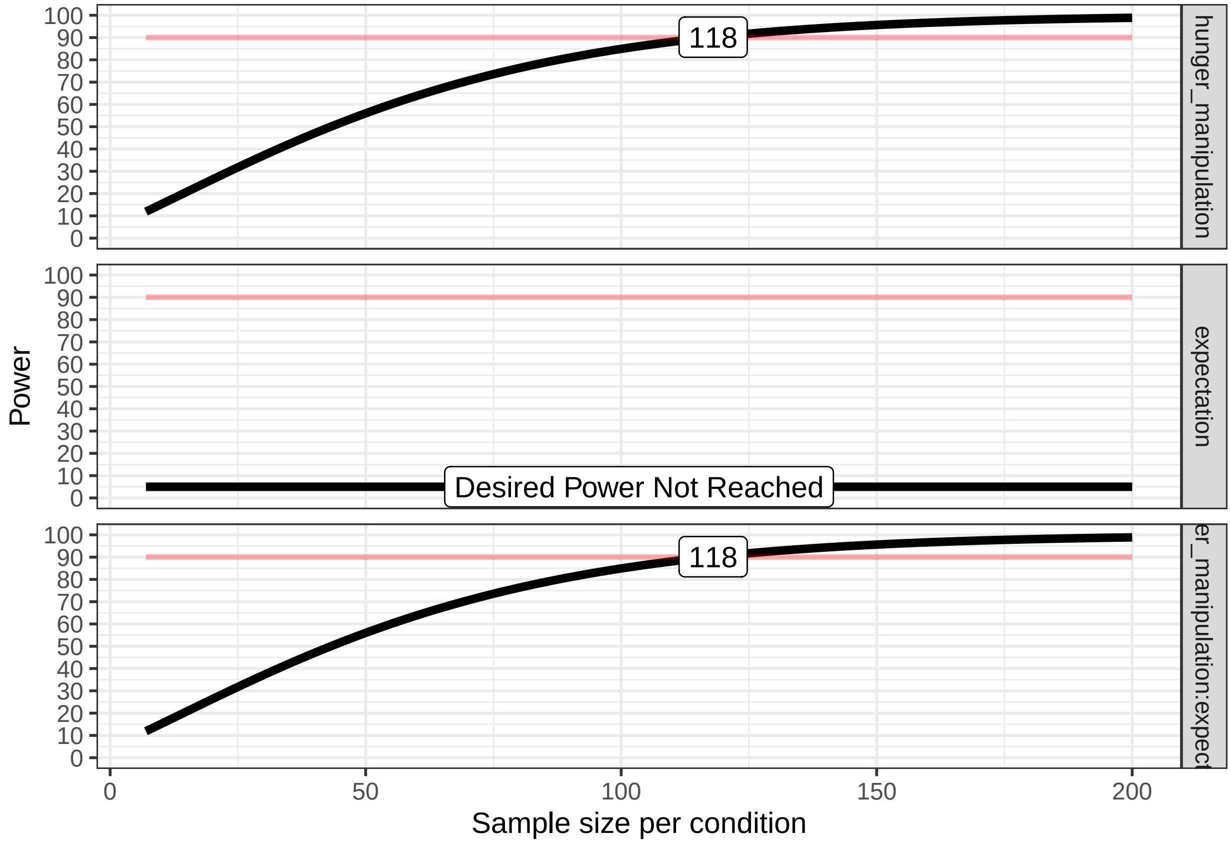
```
# to check that design is correctly specified:
#plot(design_result)

simulation_result <- ANOVA_power(design_result,
                                 alpha_level = 0.05,
                                 nsims = 1e3,
                                 verbose = FALSE)

plot(simulation_result$plot1,main="p_value distribution ANOVA")
```



```
#plot(simulation_result$plot2,main="p_value distribution paired comparisons")
```

looking at different sample sizes:

```
plot_power(design_result, max_n = 200,alpha_level = 0.05,desired_power=90)
```



```
## Achieved Power and Sample Size for ANOVA-level effects
##                           variable                     label   n achieved_power
## 1            hunger_manipulation    Desired Power Achieved 118          90.19
## 2                    expectation Desired Power Not Reached 200           5.00
## 3 hunger_manipulation:expectation    Desired Power Achieved 118          90.19
##   desired_power
## 1            90
## 2            90
## 3            90
```