# Power Analysis Apo Field work

Tom Little[1], Nick Colegrave[1]

[1]University of Edinburgh

Jun 23, 2021



This document is published without a DOI.

Little Lab

Tom Little

ABSTRACT

We used **Power analyses** to clarify our ability to detect the effects of stressors as main effects, or as interactions, with these sample sizes. We observed in ref12 that food supplementation similar to our proposed methods can cause a shift in, for example, body weight of ~ 0.75 standard deviations from the mean, and we make the assumption that we will achieve a similar effect. We constructed a linear model with chronological age as a covariate with a normal distribution and a relationship to the response variable with a slope of 0.9. We then added sex and both food and drug treatments (full model: response = age + food*drug*sex) to ask if treatment effects could be detected at the p<0.05 threshold in at least 80% of the cases during 1x105 iterations (80% is a standard threshold in power analyses55). From this, we are confident we will have the power to detect main effects within a single season. For example, where food supplementation drives an increase in the response across both sexes or across both drug and control individuals, power was >0.9, i.e there is a 90% probability of finding a significant effect. If, however, food drove the expected response in only one sex or only in drug treated individuals, our power to detect this interaction was ~0.56, below the required power threshold. Our power to detect interaction effects rises to >0.8 when the larger sample size of both field seasons are incorporated. We also observed in ref12 that 'grid' effects average ~0.5SD, and so performed further analysis with six 'grids', where treatments were nested within 'grid' for the statistical model. This analysis confirmed that the proposed sample size for one field season allows for the detection of main effects, but power to detect interaction effects requires both field seasons.

DOCUMENT CITATION

Tom Little, Nick Colegrave 2021. Power Analysis Apo Field work. **protocols.io** https://protocols.io/view/power-analysis-apo-field-work-bvw4n7gw

KEYWORDS

Power Analysis



CREATED

Jun 17, 2021

LAST MODIFIED

Jun 23, 2021

DOCUMENT INTEGER ID

50876

ABSTRACT

We used **Power analyses** to clarify our ability to detect the effects of stressors as main effects, or as interactions, with these sample sizes. We observed in ref12 that food supplementation similar to our proposed methods can cause a shift in, for example, body weight of ~ 0.75 standard deviations from the mean, and we make the assumption that we will achieve a similar effect. We constructed a linear model with chronological age as a covariate with a normal distribution and a relationship to the response variable with a slope of 0.9. We then added sex and both food and drug treatments (full model: response = age + food*drug*sex) to

ask if treatment effects could be detected at the p<0.05 threshold in at least 80% of the cases during 1x105 iterations (80% is a standard threshold in power analyses55). From this, we are confident we will have the power to detect main effects within a single season. For example, where food supplementation drives an increase in the response across both sexes or across both drug and control individuals, power was >0.9, i.e there is a 90% probability of finding a significant effect. If, however, food drove the expected response in only one sex or only in drug treated individuals, our power to detect this interaction was ~0.56, below the required power threshold. Our power to detect interaction effects rises to >0.8 when the larger sample size of both field seasons are incorporated. We also observed in ref12 that 'grid' effects average ~0.5SD, and so performed further analysis with six 'grids', where treatments were nested within 'grid' for the statistical model. This analysis confirmed that the proposed sample size for one field season allows for the detection of main effects, but power to detect interaction effects requires both field seasons.

First two notes are scripts, one varying food and infection treatment across sexes, the second food and infection across grids. You can directly paste into R studio and run, choosing the effect matrix you want. Third note is some results of the power analyses with 000's of loops. Use 1000 loops to have a quick look at the output; go up to 1x10^5 loops for accuracy (takes a while to finish). Entering zeros into the effect matrix give power of 0.05 (except for the age effect, which is defined by 'slope'), which shows the anlysis works correctly.

```
x <- rnorm(160, 25, 2)#guestimating what the mean and Sd will be for the age data if we focus on 160 autumn captured mice, average 24 weeks old

range(x)


reps <-45##number of mice per sex per treatment combination. For one season its 180/2sexes/4treat= ~22. Double that for two
runs <-1000
stdev <-5 #variation around the mean for the response variable. Using what was observed for body weight in Sweeney et al https://doi.org/10.1098/rspb.2020.2722 which
#was around 20g with a standard dev of 4. Using 25 and 5 just to be consistent the absolute numbers for age in weeks, but doesnt change output
wildage <- 25 #average chronological age of an individual
stdevAge <- 2
#need to define a relationship between age and epi age
slope <- 0.9#this can be varied. Higher values just give less power to treatment effects



##vectors containing labels to use in producing data frames. Worm just means they were treated for gastrointestinal parasites
diseasetreatments <-c("control", "Worm")
foodtreatments <-c("food1", "food2")
sexlabels <-c("male", "female")
ageeffect <- c('age')


##matrixes with the deviations from the grand mean for the
##different treatment combinations for males and females
##first bracket is the food treatment deviations for
##each level of

#effect.male <- rbind(c(3.5,3.5), c(0,0))#food 1(control, worm), food 2 (control, worm). Only a food effect here.
#effect.female <-rbind(c(3.5,3.5), c(0,0))

#an alternative way would be to have a food effect that is only felt in controls. Only in one sex would be similar
effect.male <- rbind(c(3.5,0), c(0,0))
effect.female <-rbind(c(3.5,0), c(0,0))

effect.age <- c()

##combine male and female matrixes into an array
effect.array <-array(c(effect.male, effect.female), dim = c(2,2,2), dimnames = list(c("food1", "food2"), c("control", "worms"), c("males", "female")))
```

```
print(effect.array)

##empty vectors to store p values
diseaseP <- c()
foodP <-c()
sexP <- c()
diseasefoodP <-c()
foodsexP <-c()
diseasesexP <- c()
diseasefoodsexP <-c()
ageP <- c()
agefoodP <- c()

##output array here so user can see what values are

##big loop will start here
for(run in 1:runs) {

 ##run specific vectors that will be rebuilt each time. epi is the epigenetic age, the response variable
 epi <- c()
 foodtreat <- c()
 distreat <- c()
 sex <- c()
 age <- c()



 ##loops through the various combinations
 for(d in 1:2) {
  for(f in 1:2) {
   for(s in 1:2) {
    for(r in 1:reps) {

     ##generates the value for the rep and adds to the list

     thisage <- rnorm(1, wildage, stdevAge)
     age <- append(age, thisage)

     #Generate the predicted value for this replicate using
     #the defined linear relationship
     predicted <- (effect.array[f,d,s] + (slope*thisage))
     #Generate a random deviation for this individual
     residual <- rnorm(1, 0, stdev)
     #Generate the y value by adding the deviation to the predicted
     #value
     epi <- append(epi, (predicted + residual))


     # the calculation for epi can be done in one line as follows (I just did it above in steps above for clarity):
     #epi <- append(epi, effect.array[f,d,s] + rnorm(1,0,stdev) + slope*thisage)



     ##add the right lables to lists to make up the data frame.
     distreat <- append(distreat, diseasetreatments[[d]])
     foodtreat <-append(foodtreat, foodtreatments[[f]])
     sex <- append(sex, sexlabels[[s]])

    }
   }
  }
```

```
 }


 epi.data <-data.frame(age, distreat, foodtreat, sex, epi)

 epi.data

 model1 <- lm(epi~ age + (distreat + foodtreat + sex)^2, data = epi.data)
 distreat.result <- anova(model1)
 ageP <-append(ageP, distreat.result[1,5])
 diseaseP <-append(diseaseP, distreat.result[[2,5]])
 foodP <-append(foodP, distreat.result[[3,5]])
 sexP <- append(sexP, distreat.result[[4,5]])
 diseasesexP <- append(diseasesexP, distreat.result[[6,5]])
 foodsexP <-append(foodsexP, distreat.result[[7,5]])
 diseasefoodP <-append(diseasefoodP, distreat.result[[5,5]])
# diseasefoodsexP <-append(diseasefoodsexP, distreat.result[[8,5]])


 ##bigloop ends here
}


##to estimate power check how many items in each P value list are
##less than 0.05

disease_power <- length(which(diseaseP<0.05))#what proportion of the 1000 runs detected a significant effect
food_power <- length(which(foodP<0.05))
sex_power <-length(which(sexP<0.05))
diseasebyfood_power <- length(which(diseasefoodP<0.05))
foodbysex_power <- length(which(foodsexP<0.05))
diseasebysex_power <- length(which(diseasesexP<0.05))
#diseasefoodsex_power <- length(which(diseasefoodsexP<0.05))
age_power<-length(which(ageP<0.05))

#for the inital set up food power will be near 1, and the others 0.5, which is the false discovery rate
print(paste("disease power =", (disease_power/runs)))
print(paste("food power =", (food_power/runs)))
print(paste("sex power =", (sex_power/runs)))
print(paste("disease by food power =", (diseasebyfood_power/runs)))
print(paste("food by sex power =", (foodbysex_power/runs)))
print(paste("disease by sex power =", (diseasebysex_power/runs)))
#print(paste("disease by food by sex power =", (diseasefoodsex_power/runs)))
print(paste("age power =", (age_power/runs)))

#write.csv (epi.data, file = 'age_slope.csv')

plot(x=epi.data$age, y=epi.data$epi, xlab="Chronological Age (weeks)", ylab="Epigenetic Age (weeks)",
col=factor(epi.data$foodtreat),
   cex=1.5, cex.axis=1.5, cex.lab=1.5)
```

```
rm(list=ls())
x <- rnorm(160, 25, 2)#guestimating what the mean and Sd will be for the age data if we focus on 160 autumn captured mice,
average 24 weeks old
range(x)
```

```
reps <-7##number of mice per grid per treatment combination. Imagining samples from both years, it is 360/6/4, half that for one year
runs <-10000
stdev <-5 #variation around the mean for the response variable. Using what was observed for body weight in Sweeney et al https://doi.org/10.1098/rspb.2020.2722 which
#was around 20g with a standard dev of 4. Using 25 and 5 just to be consistent the absolute numbers for age in weeks, but doesnt change output
wildage <- 25 #average chronological age
stdevAge <- 2
#need to define a relationship between age and epi age
slope <- 0.9#this can be varied. Higher values just give less power to treatment effects


##vectors containing labels to use in producing data frames. Worm just means they were treated for gastrointestinal parasites
diseasetreatments <-c("control", "Worm")
foodtreatments <-c("food1", "food2")
sexlabels <-c("male", "female")
ageeffect <- c('age')
gridlabels <- c('g1', 'g2', 'g3', 'g4', 'g5', 'g6')

##matrixes with the deviations from the grand mean for the
##different treatment combinations in the grids
##first bracket is the food treatment deviations for
##each level of

#effect.g1 <- rbind(c(3.5,3.5), c(0,0))#food 1(control, worm) then same for next food level
#effect.g2 <-rbind(c(3.5,3.5), c(0,0))
#effect.g3 <-rbind(c(3.5,3.5), c(0,0))
#effect.g4 <- rbind(c(6.6), c(2.5,2.5))
#effect.g5 <-rbind(c(6.6), c(2.5,2.5))
#effect.g6 <-rbind(c(6.6), c(2.5,2.5))

#the above set of grids has a food effect that varies between grids, specifically, half the grids are 0.75 SD above the baseline in all combos.
#Below is a set that is similar, but has food impacting only one disease treatment

effect.g1 <- rbind(c(3.5,0), c(0,0))#food 1(control, worm) then same for next food level
effect.g2 <-rbind(c(3.5,0), c(0,0))
effect.g3 <-rbind(c(3.5,0), c(0,0))
effect.g4 <- rbind(c(6,2.5), c(2.5,2.5))
effect.g5 <-rbind(c(6,2.5), c(2.5,2.5))
effect.g6 <-rbind(c(6,2.5), c(2.5,2.5))
effect.age <- c()


##combine grid matrixes into an array
effect.array <-array(c(effect.g1, effect.g2, effect.g3, effect.g4, effect.g5, effect.g6), dim = c(2,2,6), dimnames = list(c("food1", "food2"), c("control", "worms"), c("g1", "g2", "g3", "g4", "g5", "g6")))
print(effect.array)

##empty vectors to store p values
diseaseP <- c()
foodP <-c()
gridP <- c()
diseasefoodP <-c()
foodgridP <-c()
diseasegridP <- c()
diseasefoodgridP <-c()
ageP <- c()
```

```
##output array here so user can see what values are

##big loop will start here
for(run in 1:runs) {

 ##run specific vectors that will be rebuilt each time. epi is the epigenetic age, the response variable
 epi <- c()
 foodtreat <- c()
 distreat <- c()
 grid <- c()
 age <- c()


 ##loops through the various combinations
 for(d in 1:2) {
  for(f in 1:2) {
   for(s in 1:6) {
    for(r in 1:reps) {

     ##generates the value for the rep and adds to the list

     thisage <- rnorm(1, wildage, stdevAge)
     age <- append(age, thisage)


     #Generate the predicted value for this replicate using
     #the defined linear relationship
     predicted <- (effect.array[f,d,s] + (slope*thisage))
     #Generate a random deviation for this individual
     residual <- rnorm(1, 0, stdev)
     #Generate the y value by adding the deviation to the predicted
     #value
     epi <- append(epi, (predicted + residual))

     # the calculation for epi can be done in one line like this:
     #epi <- append(epi, effect.array[f,d,s] + rnorm(1,0,stdev) + slope*thisage)



     ##add the right lables to lists to make up the data frame.
     distreat <- append(distreat, diseasetreatments[[d]])
     foodtreat <-append(foodtreat, foodtreatments[[f]])
     grid <- append(grid, gridlabels[[s]])

    }
   }
  }
 }

 epi.data <-data.frame(age, distreat, foodtreat, grid, epi)

 epi.data

 model1 <- lm(epi~age+grid+(distreat/grid + foodtreat/grid)^2, data = epi.data)#grid  nested within  treatment
 distreat.result <- anova(model1)
 ageP <-append(ageP, distreat.result[1,5])
 gridP <- append(gridP, distreat.result[[2,5]])
 diseaseP <- append(diseaseP, distreat.result[[3,5]])
 foodP <-append(foodP, distreat.result[[4,5]])
```

```
 diseasegridP <-append(diseasegridP, distreat.result[[5,5]])
 diseasefoodP <- append(diseasefoodP, distreat.result[[6,5]])
 diseasefoodgridP <-append(diseasefoodgridP, distreat.result[[7,5]])


 ##bigloop ends here
}

##to estimate power check how manu items in each P value list are
##less than 0.05

disease_power <- length(which(diseaseP<0.05))#what proportion of the 1000 runs detected a significant effect
food_power <- length(which(foodP<0.05))
grid_power <-length(which(gridP<0.05))
diseasebyfood_power <- length(which(diseasefoodP<0.05))
diseasebygrid_power <- length(which(diseasegridP<0.05))
diseasefoodgrid_power <- length(which(diseasefoodgridP<0.05))
age_power<-length(which(ageP<0.05))


print(paste("disease power =", (disease_power/runs)))
print(paste("food power =", (food_power/runs)))
print(paste("grid power =", (grid_power/runs)))
print(paste("disease by food power =", (diseasebyfood_power/runs)))
print(paste("disease by grid power =", (diseasebygrid_power/runs)))
print(paste("disease by food by grid power =", (diseasefoodgrid_power/runs)))
print(paste("age power =", (age_power/runs)))

plot(x=epi.data$age, y=epi.data$epi, xlab="Chronological Age (weeks)", ylab="Epigenetic Age (weeks)",
col=factor(epi.data$foodtreat),
   cex=1.5, cex.axis=1.5, cex.lab=1.5)
#write.csv (epi.data, file = 'age_slope.csv')
```

Power analysis 1: food, disease, sex

One season with food main effects

```
effect.male <- rbind(c(3.5,3.5), c(0,0))
effect.female <-rbind(c(3.5,3.5), c(0,0))

[1] "disease power = 0.0481"
> print(paste("food power =", (food_power/runs)))
[1] "food power = 0.9961"
> print(paste("sex power =", (sex_power/runs)))
[1] "sex power = 0.0476"
> print(paste("disease by food power =", (diseasebyfood_power/runs)))
[1] "disease by food power = 0.0501"
> print(paste("food by sex power =", (foodbysex_power/runs)))
[1] "food by sex power = 0.0516"
> print(paste("disease by sex power =", (diseasebysex_power/runs)))
[1] "disease by sex power = 0.0477"
> #print(paste("disease by food by sex power =", (diseasefoodsex_power/runs)))
> print(paste("age power =", (age_power/runs)))
```

[1] "age power = 0.9948"

One season with food x treat interaction

effect.male <- rbind(c(3.5,0), c(0,0))
effect.female <-rbind(c(3.5,0), c(0,0))

[1] "disease power = 0.6363"
> print(paste("food power =", (food_power/runs)))
[1] "food power = 0.6356"
> print(paste("sex power =", (sex_power/runs)))
[1] "sex power = 0.0517"
> print(paste("disease by food power =", (diseasebyfood_power/runs)))
[1] "disease by food power = 0.6312"
> print(paste("food by sex power =", (foodbysex_power/runs)))
[1] "food by sex power = 0.0493"
> print(paste("disease by sex power =", (diseasebysex_power/runs)))
[1] "disease by sex power = 0.0482"
> #print(paste("disease by food by sex power =", (diseasefoodsex_power/runs)))
> print(paste("age power =", (age_power/runs)))
[1] "age power = 0.9941"

Two seasons with food by treat interaction

[1] "disease power = 0.9083"
> print(paste("food power =", (food_power/runs)))
[1] "food power = 0.9143"
> print(paste("sex power =", (sex_power/runs)))
[1] "sex power = 0.0476"
> print(paste("disease by food power =", (diseasebyfood_power/runs)))
[1] "disease by food power = 0.9083"
> print(paste("food by sex power =", (foodbysex_power/runs)))
[1] "food by sex power = 0.0487"
> print(paste("disease by sex power =", (diseasebysex_power/runs)))
[1] "disease by sex power = 0.0488"
> #print(paste("disease by food by sex power =", (diseasefoodsex_power/runs)))
> print(paste("age power =", (age_power/runs)))
[1] "age power = 1"

Power analysis 2: food treatment and grid effects

One season with food main effects (and grid effect)

effect.g1 <- rbind(c(3.5,3.5), c(0,0))#food 1(control, worm) then same for next food level
effect.g2 <-rbind(c(3.5,3.5), c(0,0))
effect.g3 <-rbind(c(3.5,3.5), c(0,0))
effect.g4 <- rbind(c(6,6), c(2.5,2.5))
effect.g5 <-rbind(c(6,6), c(2.5,2.5))
effect.g6 <-rbind(c(6,6), c(2.5,2.5))

[1] "disease power = 0.052"
> print(paste("food power =", (food_power/runs)))
[1] "food power = 0.9965"
> print(paste("grid power =", (grid_power/runs)))
[1] "grid power = 0.7182"
> print(paste("disease by food power =", (diseasebyfood_power/runs)))
[1] "disease by food power = 0.0528"
> print(paste("disease by grid power =", (diseasebygrid_power/runs)))
[1] "disease by grid power = 0.0532"
> print(paste("disease by food by grid power =", (diseasefoodgrid_power/runs)))
[1] "disease by food by grid power = 0.0529"

```
> print(paste("age power =", (age_power/runs)))
[1] "age power = 0.9976"
```

One season with food by treat interactions (and grid effect)

```
effect.g1 <- rbind(c(3.5,0), c(0,0))#food 1(control, worm) then same for next food level
effect.g2 <-rbind(c(3.5,0), c(0,0))
effect.g3 <-rbind(c(3.5,0), c(0,0))
effect.g4 <- rbind(c(6,2.5), c(2.5,2.5))
effect.g5 <-rbind(c(6,2.5), c(2.5,2.5))
effect.g6 <-rbind(c(6,2.5), c(2.5,2.5))

[1] "disease power = 0.6121"
> print(paste("food power =", (food_power/runs)))
[1] "food power = 0.6175"
> print(paste("grid power =", (grid_power/runs)))
[1] "grid power = 0.6787"
> print(paste("disease by food power =", (diseasebyfood_power/runs)))
[1] "disease by food power = 0.6108"
> print(paste("disease by grid power =", (diseasebygrid_power/runs)))
[1] "disease by grid power = 0.0504"
> print(paste("disease by food by grid power =", (diseasefoodgrid_power/runs)))
[1] "disease by food by grid power = 0.0509"
> print(paste("age power =", (age_power/runs)))
[1] "age power = 0.9983"
```

Two seasons with food by treat interactions (and grid effect)

```
effect.g1 <- rbind(c(3.5,0), c(0,0))#food 1(control, worm) then same for next food level
effect.g2 <-rbind(c(3.5,0), c(0,0))
effect.g3 <-rbind(c(3.5,0), c(0,0))
effect.g4 <- rbind(c(6,2.5), c(2.5,2.5))
effect.g5 <-rbind(c(6,2.5), c(2.5,2.5))
effect.g6 <-rbind(c(6,2.5), c(2.5,2.5))

[1] "disease power = 0.9118"
> print(paste("food power =", (food_power/runs)))
[1] "food power = 0.9096"
> print(paste("grid power =", (grid_power/runs)))
[1] "grid power = 0.9712"
> print(paste("disease by food power =", (diseasebyfood_power/runs)))
[1] "disease by food power = 0.9122"
> print(paste("disease by grid power =", (diseasebygrid_power/runs)))
[1] "disease by grid power = 0.0498"
> print(paste("disease by food by grid power =", (diseasefoodgrid_power/runs)))
[1] "disease by food by grid power = 0.0456"
> print(paste("age power =", (age_power/runs)))
[1] "age power = 1"
```