R pwr calc

```
library(pwr)
library(ggplot2)

EC36309merg <- read.table("mergedEC36309.csv", header=TRUE,
                          sep=",")
attach(EC36309merg)

VarGFP.RFP <- var(EC36309merg$GFP.RFP)
VarGFP.RFP
```

```
## [1] 0.01626678
```

```
sdGFP.RFP <- sqrt(VarGFP.RFP)
sdGFP.RFP               #sample standard deviation of GFP/RFP ratio EC36309 = 0.13
```

```
## [1] 0.1275413
```

```
sdGFP.RFPlarge <- 0.25  #setting standard deviation to a conservative estimate (almost double from init


avgGFPRFPeff <- median(EC36309merg$GFP.RFP*1.50) #detecting x% greater signal of GFP
avgGFPRFP <- median(EC36309merg$GFP.RFP)            #reference

# Formula for effect size calculation:  d = muA - mu0 / sigma
deff <- (avgGFPRFPeff-avgGFPRFP)/sdGFP.RFP          #calculating effect size
deff
```

```
## [1] 1.803338
```

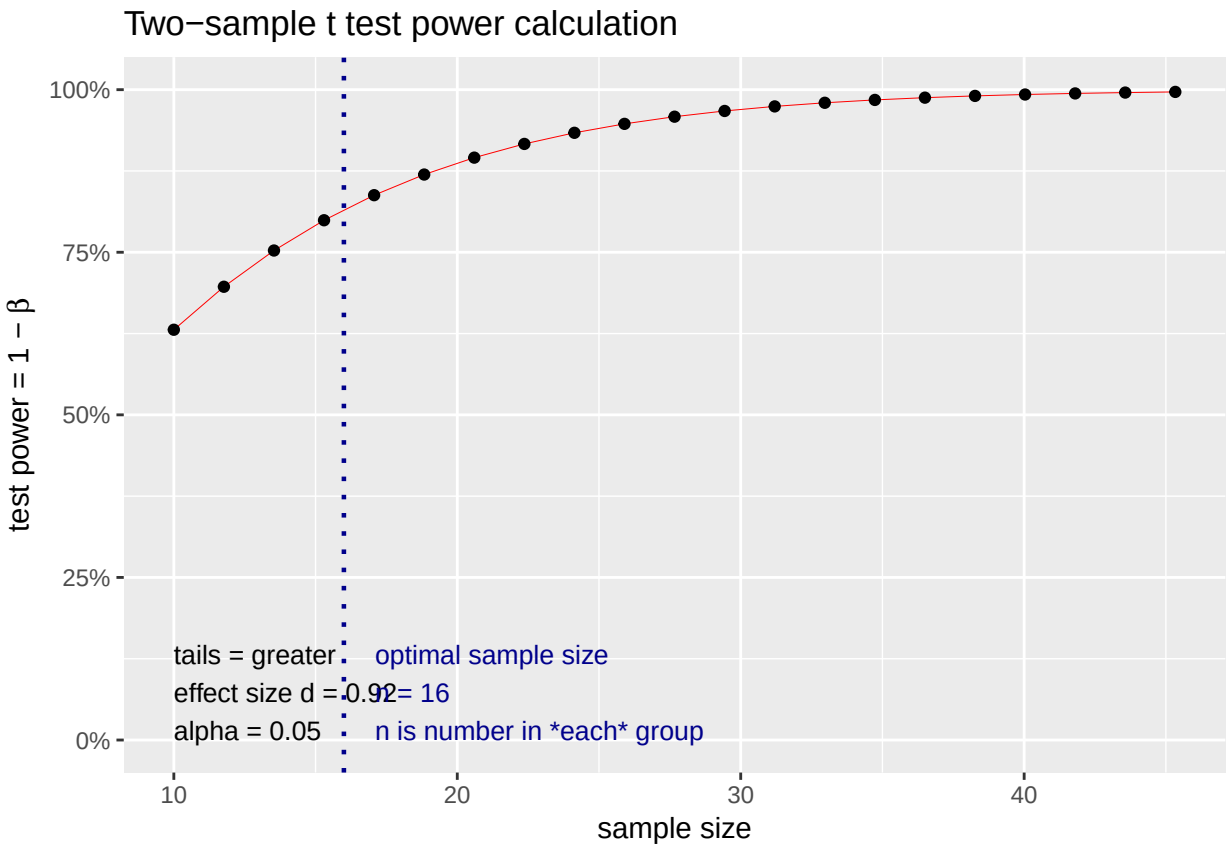
```
dsmall <- (avgGFPRFPeff-avgGFPRFP)/sdGFP.RFPlarge       #calculating effect size for larger sd, assumin
dsmall
```

```
## [1] 0.92
```

```
pwr.conservative <- pwr.t.test(n = NULL, d = dsmall, sig.level = 0.05, power = 0.80,
           type = "two.sample",
           alternative = "greater") #power calculation under the assumption of larger uncertainty (almo
pwr.conservative
```

```
##
##      Two-sample t test power calculation
##
##               n = 15.33302
##               d = 0.92
##       sig.level = 0.05
##           power = 0.8
##     alternative = greater
##
## NOTE: n is number in *each* group
```

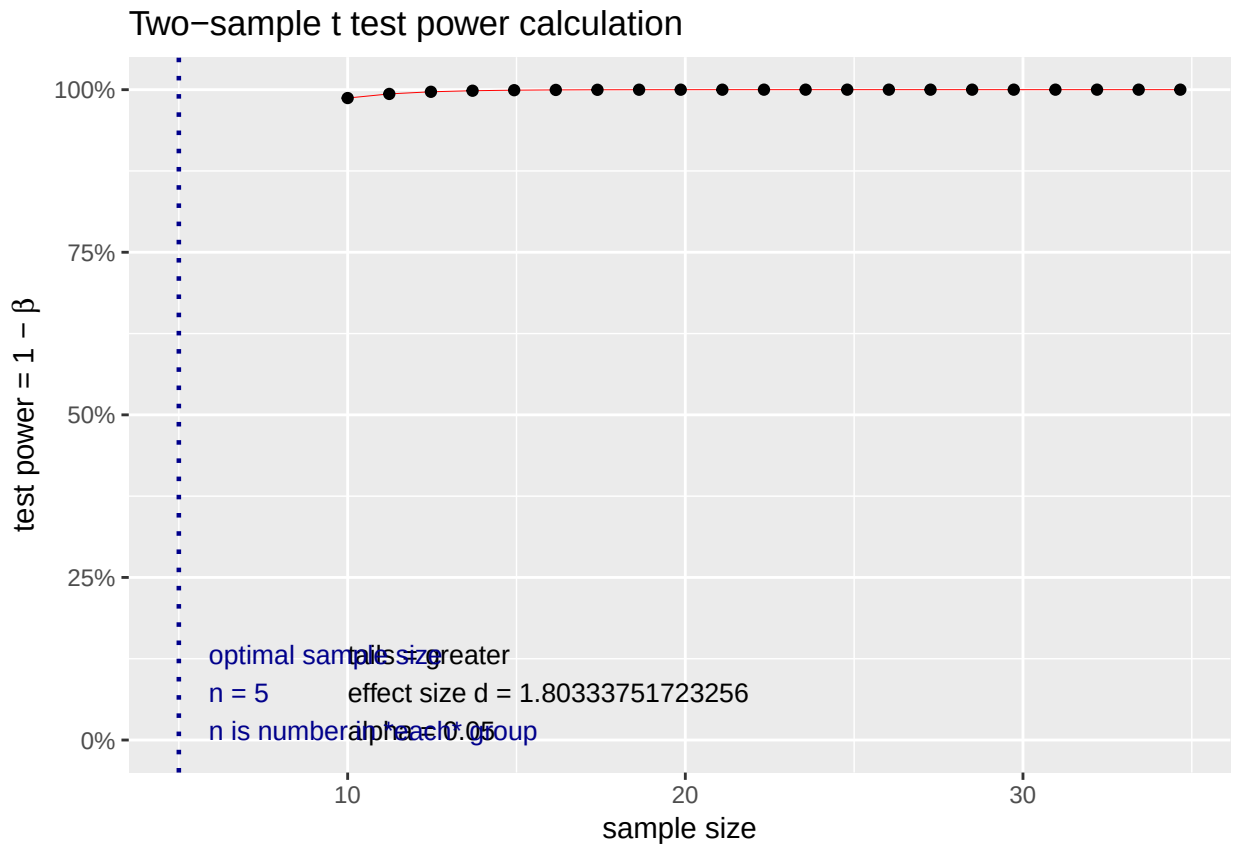
```
plot(pwr.conservative)
```




```
# power calculation for one-sided t-test
pwr.est <- pwr.t.test(n = NULL, d = deff, sig.level = 0.05, power = 0.8,
           type = "two.sample",
           alternative = "greater") #power calculation under the assumption of estimated uncertainty
pwr.est
```

```
##
##      Two-sample t test power calculation
##
##               n = 4.659678
##               d = 1.803338
##       sig.level = 0.05
##           power = 0.8
##     alternative = greater
##
## NOTE: n is number in *each* group
```

```
plot(pwr.est)
```




```
# Two-tailed t-test instead of one-tailed
pwr.conservative2 <- pwr.t.test(n = NULL, d = dsmall, sig.level = 0.05, power = 0.80,
          type = "two.sample") #two-tailed power calculation under the assumption of larger uncertainty
pwr.conservative2
```

```
##
##      Two-sample t test power calculation
##
##               n = 19.55492
##               d = 0.92
##       sig.level = 0.05
##           power = 0.8
##     alternative = two.sided
##
## NOTE: n is number in *each* group
```

```
pwr.est2 <- pwr.t.test(n = NULL, d = deff, sig.level = 0.05, power = 0.8,
          type = "two.sample") #two-tailed power calculation under the assumption of estimated uncerta
pwr.est2
```

```
##
##      Two-sample t test power calculation
##
##               n = 5.959306
##               d = 1.803338
##       sig.level = 0.05
##           power = 0.8
```

```
##     alternative = two.sided
##
## NOTE: n is number in *each* group
```

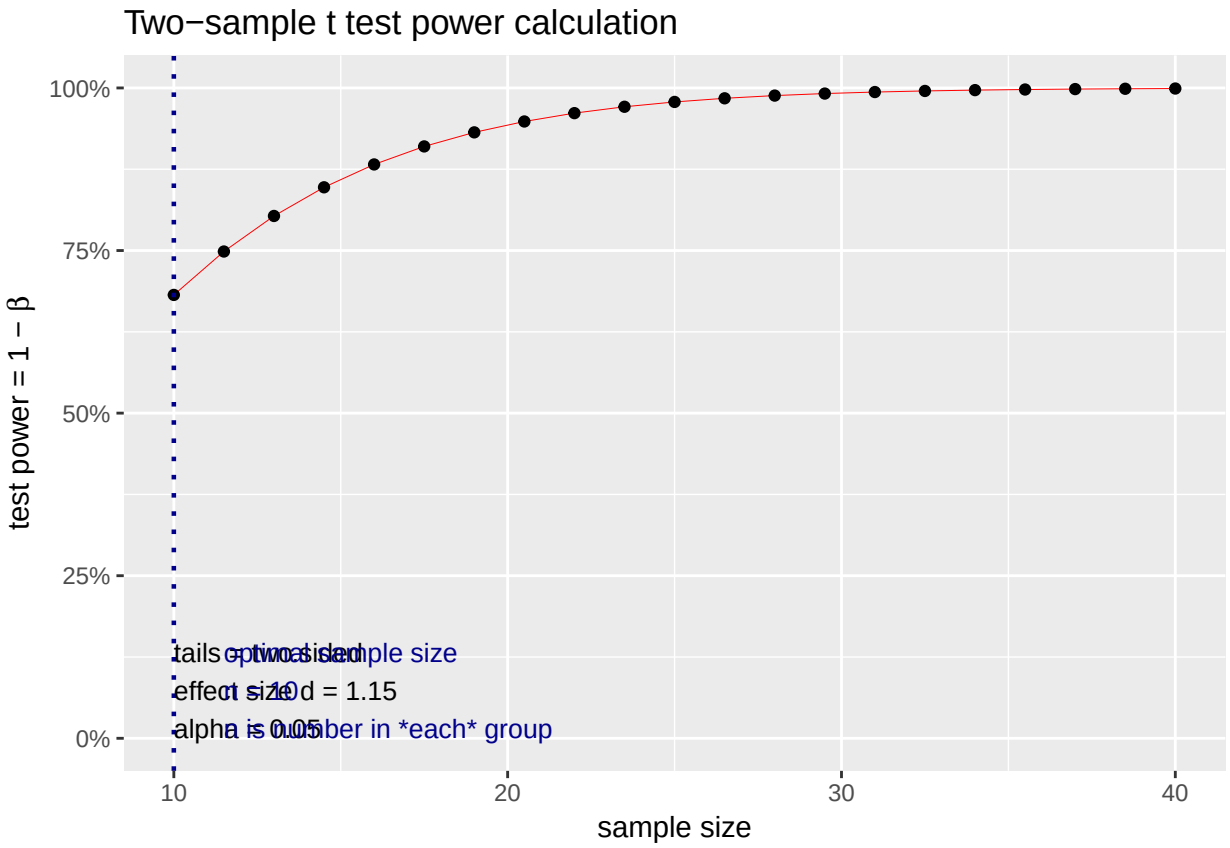
```
# Power calc. of conservative estimate of sd of 0.2 instead of 0.13
dmed <- (avgGFPRFPeff-avgGFPRFP)/0.2
dmed
```

```
## [1] 1.15
```

```
pwr.conservative3 <- pwr.t.test(n = 10, d = dmed, sig.level = 0.05, power = NULL,
          type = "two.sample") #two-tailed power calculation under the assumption of larger uncertainty
pwr.conservative3
```

```
##
##      Two-sample t test power calculation
##
##               n = 10
##               d = 1.15
##       sig.level = 0.05
##           power = 0.6817598
##     alternative = two.sided
##
## NOTE: n is number in *each* group
```

```
plot(pwr.conservative3)
```



Plotting multiple power curves

```r
library(pwr) # for power calcs
library(dplyr) # for data manipulation
```

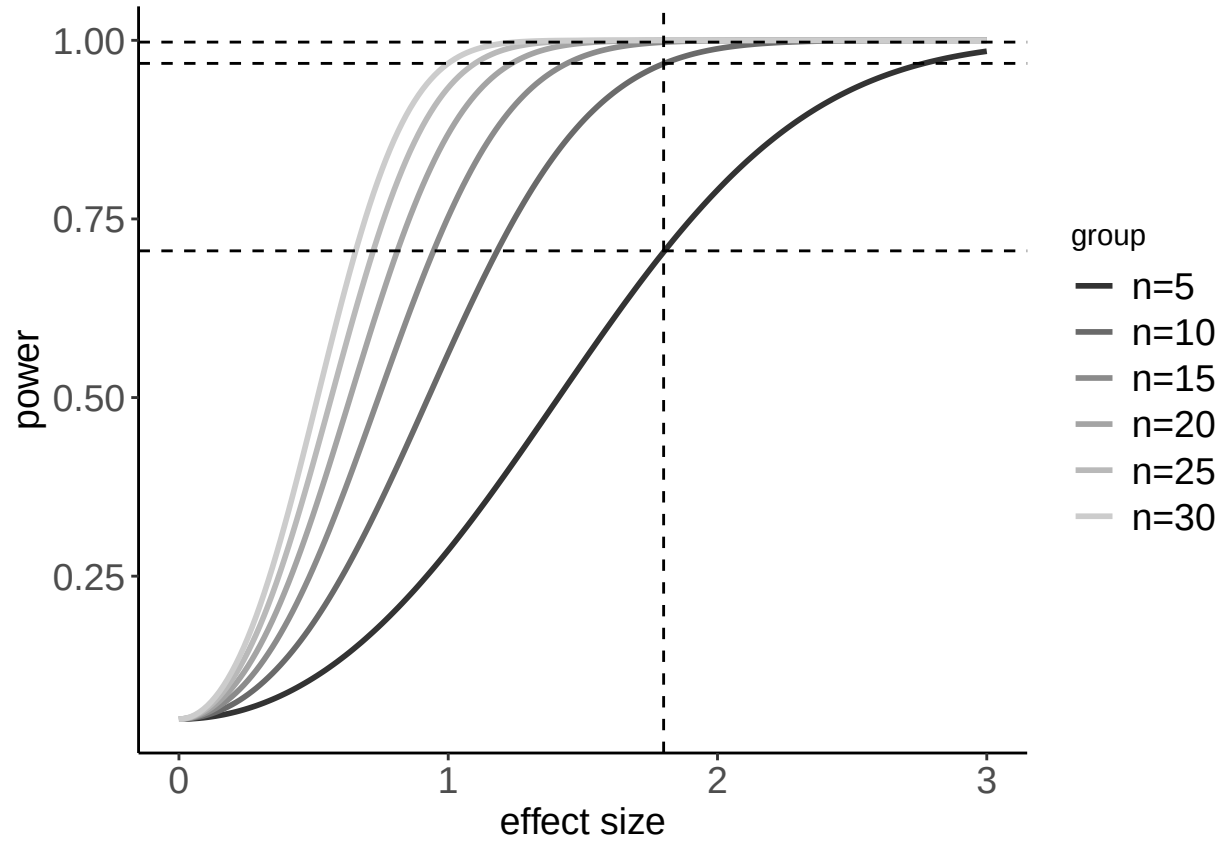
```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
library(tidyr) # for data manipulation
library(ggplot2) # for plotting power curves

# Source: https://moderndata.plot.ly/power-curves-r-plotly-ggplot2/
# Generate power calculations
ptab <- cbind(NULL, NULL)

for (i in seq(0,3, length.out = 200)){
  pwrt1 <- pwr.t.test(n = 5, d = i, sig.level = 0.05, power = NULL,
           type = "two.sample", alternative="two.sided")
  pwrt2 <- pwr.t.test(n = 10, d = i, sig.level = 0.05, power = NULL,
           type = "two.sample", alternative="two.sided")
  pwrt3 <- pwr.t.test(n = 15, d = i, sig.level = 0.05, power = NULL,
           type = "two.sample", alternative="two.sided")
  pwrt4 <- pwr.t.test(n = 20, d = i, sig.level = 0.05, power = NULL,
           type = "two.sample", alternative="two.sided")
  pwrt5 <- pwr.t.test(n = 25, d = i, sig.level = 0.05, power = NULL,
           type = "two.sample", alternative="two.sided")
  pwrt6 <- pwr.t.test(n = 30, d = i, sig.level = 0.05, power = NULL,
           type = "two.sample", alternative="two.sided")
  ptab <- rbind(ptab, cbind(pwrt1$d, pwrt1$power,
                            pwrt2$d, pwrt2$power,
                            pwrt3$d, pwrt3$power,
                            pwrt4$d, pwrt4$power,
                            pwrt5$d, pwrt5$power,
                            pwrt6$d, pwrt6$power))
}

ptab <- cbind(seq_len(nrow(ptab)), ptab)

colnames(ptab) <- c("id","n=5.effect size","n=5.power",
                    "n=10.effect size","n=10.power",
                    "n=15.effect size","n=15.power",
                    "n=20.effect size","n=20.power",
                    "n=25.effect size","n=25.power",
                    "n=30.effect size","n=30.power")

# get data into right format for ggplot2
temp <- ptab %>%
  as.data.frame() %>%
  gather(key = name, value = val, 2:13) %>%
```

```r
  separate(col = name, into = c("group", "var"), sep = "\\.") %>%
  spread(key = var, value = val)

# factor group
temp$group <- factor(temp$group,
                     levels = c("n=5", "n=10",
                     "n=15", "n=20",
                     "n=25", "n=30"))


# plot
p <- ggplot(temp, aes(x = `effect size`, y = power, color = group)) +
     geom_line(size=1) +
     theme_classic() +
     scale_color_grey() +
     theme(axis.text=element_text(size=14),
           axis.title=element_text(size=14),
           legend.text=element_text(size=14)) +
     geom_vline(xintercept = 1.8, linetype = 2) +
     geom_hline(yintercept = 0.7053, linetype = 2) +
     geom_hline(yintercept = 0.9677, linetype = 2) +
     geom_hline(yintercept = 0.99747, linetype = 2)

p
```



```r
#---- power calc with calculated sd (0.13)

pwreff.n5 <- pwr.t.test(n = 5, d = deff, sig.level = 0.05, power = NULL,
          type = "two.sample") #two-tailed power calculation under the assumption of larger uncertaint
pwreff.n5
```

```
##
##      Two-sample t test power calculation
##
##               n = 5
##               d = 1.803338
##       sig.level = 0.05
##           power = 0.7053254
##     alternative = two.sided
##
## NOTE: n is number in *each* group
```

```r
pwreff.n10 <- pwr.t.test(n = 10, d = deff, sig.level = 0.05, power = NULL,
          type = "two.sample") #two-tailed power calculation under the assumption of larger uncertaint
pwreff.n10
```

```
##
##      Two-sample t test power calculation
##
##               n = 10
##               d = 1.803338
##       sig.level = 0.05
##           power = 0.9677896
##     alternative = two.sided
##
## NOTE: n is number in *each* group
```

```r
pwreff.n15 <- pwr.t.test(n = 15, d = deff, sig.level = 0.05, power = NULL,
          type = "two.sample") #two-tailed power calculation under the assumption of larger uncertaint
pwreff.n15
```

```
##
##      Two-sample t test power calculation
##
##               n = 15
##               d = 1.803338
##       sig.level = 0.05
##           power = 0.9974709
##     alternative = two.sided
##
## NOTE: n is number in *each* group
```

```r
#-------- pwer calc for maximum effect size obtainable with different n

pwreff.n5 <- pwr.t.test(n = 5, d = NULL, sig.level = 0.05, power = 0.8,
          type = "two.sample") #two-tailed power calculation under the assumption of larger uncertaint
pwreff.n5
```

```
##
##      Two-sample t test power calculation
##
```

```
##               n = 5
##               d = 2.024439
##       sig.level = 0.05
##           power = 0.8
##     alternative = two.sided
##
## NOTE: n is number in *each* group
```

```
pwreff.n10 <- pwr.t.test(n = 10, d = NULL, sig.level = 0.05, power = 0.8,
           type = "two.sample") #two-tailed power calculation under the assumption of larger uncertaint
pwreff.n10
```

```
##
##      Two-sample t test power calculation
##
##               n = 10
##               d = 1.324947
##       sig.level = 0.05
##           power = 0.8
##     alternative = two.sided
##
## NOTE: n is number in *each* group
```

```
pwreff.n15 <- pwr.t.test(n = 15, d = NULL, sig.level = 0.05, power = 0.8,
           type = "two.sample") #two-tailed power calculation under the assumption of larger uncertaint
pwreff.n15
```

```
##
##      Two-sample t test power calculation
##
##               n = 15
##               d = 1.059797
##       sig.level = 0.05
##           power = 0.8
##     alternative = two.sided
##
## NOTE: n is number in *each* group
```

```
# effect size for n = 5, 10 or 15
dn5 <- 2.02

dn10 <- 1.3

dn15 <- 1.06


#Formula for effect size calculation:  d = muA - mu0 / sigma
#deff <- (avgGFPRFPeff-avgGFPRFP)/sdGFP.RFP          #calculating effect size
#deff

#Rearranging effect size equation to muA (corresponding elevated GFP/RFP ratio)
# muA = d * sigma + mu0
mun5 <- dn5*sdGFP.RFP + avgGFPRFP
mun5
```

```
## [1] 0.7176334
```

```r
# relating elevated signal ratio to reference signal ratio: percentage change
effn5 <- mun5/avgGFPRFP
effn5
```

```
## [1] 1.560073
```

```r
#56% with n = 10

mun10 <- dn10*sdGFP.RFP + avgGFPRFP
mun10
```

```
## [1] 0.6258037
```

```r
effn10 <- mun10/avgGFPRFP
effn10
```

```
## [1] 1.360443
```

```r
#36% with n = 10

mun15 <- dn15*sdGFP.RFP + avgGFPRFP
mun15
```

```
## [1] 0.5951938
```

```r
effn15 <- mun15/avgGFPRFP
effn15
```

```
## [1] 1.2939
```

```r
#29 % with n = 15

# How many samples to detect 10%?
d10 <- ((avgGFPRFP*1.1)-avgGFPRFP)/sdGFP.RFP          #calculating effect size
d10
```

```
## [1] 0.3606675
```

```r
pwreff.d10 <- pwr.t.test(n = NULL, d = d10, sig.level = 0.05, power = 0.8,
           type = "two.sample") #two-tailed power calculation under the assumption of larger uncertaint
pwreff.d10
```

```
##
##      Two-sample t test power calculation
##
##               n = 121.6443
##               d = 0.3606675
##       sig.level = 0.05
##           power = 0.8
##     alternative = two.sided
##
## NOTE: n is number in *each* group
```

```r
#-------- pwer calc with conservative effect size (sd at 0.2)

pwrmed.n5 <- pwr.t.test(n = 5, d = dmed, sig.level = 0.05, power = NULL,
           type = "two.sample") #two-tailed power calculation under the assumption of larger uncertaint
pwrmed.n5
```

```
##
```

```
##      Two-sample t test power calculation
##
##               n = 5
##               d = 1.15
##       sig.level = 0.05
##           power = 0.3603519
##     alternative = two.sided
##
## NOTE: n is number in *each* group
```

```
pwrmed.n10 <- pwr.t.test(n = 10, d = dmed, sig.level = 0.05, power = NULL,
          type = "two.sample") #two-tailed power calculation under the assumption of larger uncertaint
pwrmed.n10
```

```
##
##      Two-sample t test power calculation
##
##               n = 10
##               d = 1.15
##       sig.level = 0.05
##           power = 0.6817598
##     alternative = two.sided
##
## NOTE: n is number in *each* group
```

```
pwrmed.n15 <- pwr.t.test(n = 15, d = dmed, sig.level = 0.05, power = NULL,
          type = "two.sample") #two-tailed power calculation under the assumption of larger uncertaint
pwrmed.n15
```

```
##
##      Two-sample t test power calculation
##
##               n = 15
##               d = 1.15
##       sig.level = 0.05
##           power = 0.8598764
##     alternative = two.sided
##
## NOTE: n is number in *each* group
```