



Study Coordination Center  
Clinical Trial Unit

How sensitive is your power?

Beyond calculating sample size

Thomas Fabbro

supported by:



SCHWEIZERISCHER NATIONALFONDS  
ZUR FÖRDERUNG DER WISSENSCHAFTLICHEN FORSCHUNG

---

# Table of contents

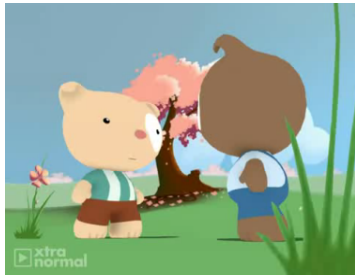
---

- 1 About asking the right question
- 2 About giving a better answer
- 3 The “power” package
- 4 Semiparametric resampling

---

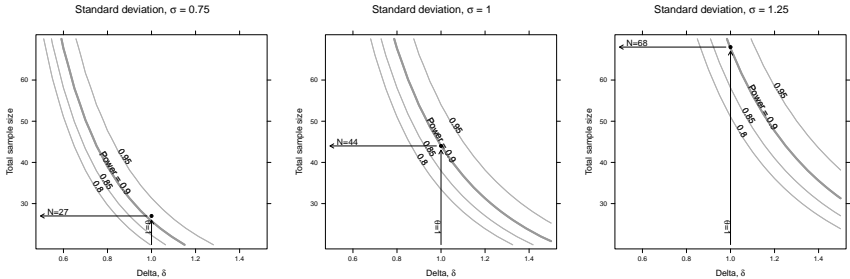
## About asking the right question

---



“How many subjects do I need to recruit, assuming a difference of  $\delta = 1$  between two groups and a standard deviation of  $\sigma = 1$  ?”  
(Power of 90%, Level of Significance of 5%)

# About giving a better answer



... and a thorough description of the calculation.

# But the grant is due tomorrow...

---

# The “power” package

---

The package contains several classes and methods aiming to provide fast and reliable tools for addressing important points *beyond* sample size.

- Sensitivity of sample size with respect to
  - a continuous parameter, called “theta” (e.g.  $\delta$ ),
  - a categorical or continuous parameter, called “xi” (e.g.  $\sigma$ ),
  - the power.
- Methods for addressing information from the calculation for writing reports with Sweave.
- Mechanism for calculating the power using a simulation approach.

# The "power" package

```
psi <- powPar(delta = seq(from = 0.5, to = 1.5, by = 0.05),
             group.sd = c(0.75, 1, 1.25),
             n = seq(from = 20, to = 50, by = 2),
             theta.name = "delta",
             xi.name = "group.sd")
```

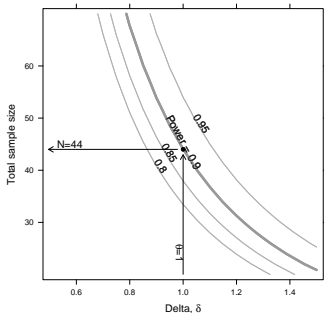
```
powFun <- function(psi) {
  power.t.test(n = n(psi)/2,
              delta = theta(psi),
              sd = xi(psi))$power
}
```

```
calc <- powCalc(psi, statistic = powFun)
```

```
ex <- powEx(theta = 1, xi = 1)
```

```
pow <- merge(calc, ex)
```

```
plot(pow)
```



---

## The “power” package

---

For writing reports a method called “tex” allows to extract information. Currently these are

- the power used for the example,
- the theta and xi used for the example,
- the number of subjects needed for evaluation,
- the number of subjects that should be recruited, taking into account the proportion of expected drop-outs,
- and with the argument “resample”, a description of the sampling procedure and the number of resamples.

---

# Semiparametric resampling

---

If data from a pilot study is available, a semiparametric resampling can be used to account

- non-parametrically for available data
- and parametrically for an effect size, e.g. a shift in location.

An idea by Davison and Hinkley, *Bootstrap Methods and their Application*, Cambridge Series in Statistical and Probabilistic Mathematics, 1997.

---

# Semiparametric resampling

---

Calculating the power for the null hypothesis,  $H_0 : \theta = 0$ , using the Wilcoxon test, where  $\theta$  is a shift in location and the underlying distribution  $F$  is unspecified.

- Pool  $F(g_1)$ , and  $F(g_2 - \hat{\theta})$  to  $\hat{F}$ , where  $\hat{\theta}$  is the difference in sample medians between group one and two.
- Sample  $n_1$  subjects from  $\hat{F}$ .
- Sample  $n_2$  subjects from  $\hat{F}$  and add  $\theta$  to each.
- Calculate a test statistic  $t^*$  with  $R$  repetitions.
- Calculate the proportion of successful repetitions.

---

# Semiparametric resampling

---

```
psi <- powPar(pilot = pilot.data,
             delta = seq(from = 0.5, to = 1.5, by = 0.05),
             n = seq(from = 10, to = 30, by = 2),
             theta.name = "delta")

powFun <- function(psi){
  a <- sample(pp(psi, "pilot"), size = n(psi)/2, replace = TRUE)
  b <- sample(pp(psi, "pilot"), size = n(psi)/2, replace = TRUE) + theta(psi)
  return(wilcox.test(a, b)$p.value < 0.05)
}

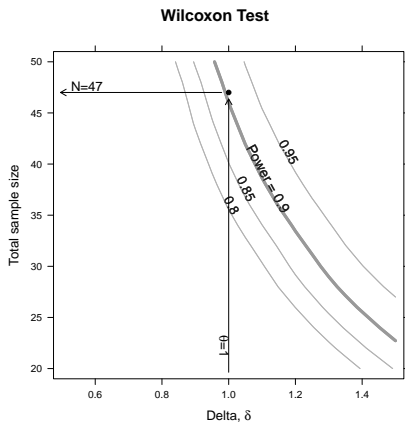
calc <- powCalc(psi, statistic = powFun, type = "resample", n.iter = 10000)

ex <- powEx(theta = 1)

pow <- merge(calc, ex)

plot(pow)
```

# Semiparametric resampling



---

# Comparing Wilcoxon Test and t Test

---

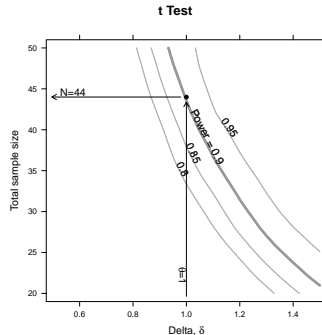
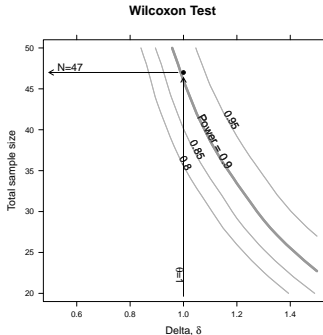
```
psi <- powPar(pilot = pilot.data,
             delta = seq(from = 0.5, to = 1.5, by = 0.05),
             n = seq(from = 10, to = 30, by = 2),
             theta.name = "delta")

powFun <- function(psi){
  a <- sample(pp(psi, "pilot"), size = n(psi)/2, replace = TRUE)
  b <- sample(pp(psi, "pilot"), size = n(psi)/2, replace = TRUE) + theta(psi)
  w <- wilcox.test(a, b)$p.value < 0.05
  t <- t.test(a, b)$p.value < 0.05
  return(c(w, t))
}

calc <- powCalc(psi, statistic = powFun, type = "resample", n.iter = 10000)

ex.w <- powEx(theta = 1, endpoint = "w")
ex.t <- powEx(theta = 1, endpoint = "t")
```

# Comparing Wilcoxon Test and t Test



---

## Acknowledgements and more about "power"

---

I would like to thank the statisticians of the Clinical Trial Unit, CTU, for their useful comments:

- Nicole Bruni
- Stefanie von Felten
- Thomas Zumbrunn

The "power" package is available at:

- [power.r-forge.r-project.org](http://power.r-forge.r-project.org)