Poweranalyse
oder: Die richtige Stichprobengrösse

Warum Poweranalyse ?

- Wie viele Stichproben brauchen wir, um eine gewisse Alternative mit 80% Wa. (= Macht) erkennen zu können ?
- Zu viel: Unnötiger Aufwand
- Zu wenig: Nutzlose Studie

- Zeitverschwendung
- Ethische Probleme

- Praxis: Stichprobengrösse = «Educated guess»
(weil Parameter der Alternativhypothese willkürlich)

Wdh: Zauberwürfel oder nicht ?

Wie oft müssen wir würfeln ?

Binomialtest: Bsp Zauberwürfel

1. Modell: X: Anzahl 6er bei 50 Würfen; $X \sim Bin(n = 50, \pi = 1/6)$
2. Nullhypothese: $H_0: \pi = 1/6$
   Alternative: $H_A: \pi > 1/6$ (einseitig)
3. Teststatistik T: Anz. 6er bei 50 Würfen
   Verteilung der Teststatistik, wenn Nullhypothese stimmt:
   $$T \sim Bin(50, 1/6)$$
4. Signifikanzniveau: $\alpha = 0.05$ (Konvention)
5. Verwerfungsbereich der Teststatistik:
   $P[T = t] = \binom{n}{t}\pi^t(1-\pi)^{n-t}$; berechne $P[T \geq t]$

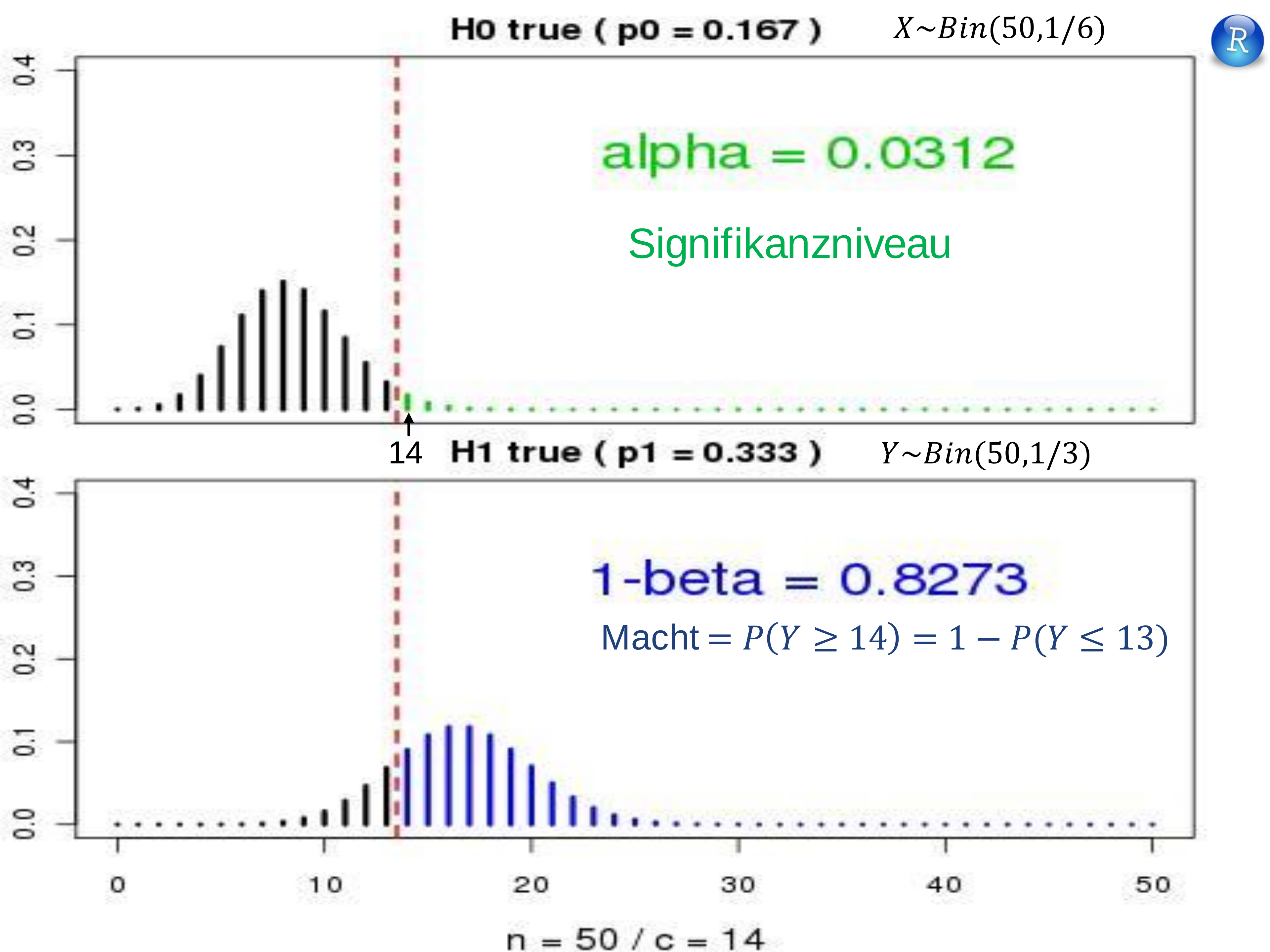
Verwerfungsbereich
Grenze: Kleinste Zahl t,
sodass $P[T \geq t] \leq \alpha$

t	…	13	14	15	…
$P[T \geq t]$	…	0.06	0.03	0.01	…

6. Testentscheid: Liegt die beobachtete Anzahl 6er bei 50 Würfen im Verwerfungsbereich der Nullhypothese?
   Falls ja: H_0 wird auf dem 5% Niveau verworfen
   Falls nein: H_0 kann auf dem 5% Niveau nicht verworfen werden

Wdh: Fehler 1. und 2. Art, Macht

- $H_0: X \sim Bin\left(n, p = \frac{1}{6}\right); H_A: p > \frac{1}{6}$
- **Fehler 1. Art**: H_0 stimmt, wird aber im Test verworfen
  Wa. für Fehler 1. Art ist höchstens α
- Fehler 2. Art: H_A stimmt, H_0 wird aber im Test **nicht** verworfen
- Macht: Wa. dass H_0 verworfen wird, falls H_A stimmt
  Macht = 1 – P(Fehler 2. Art)
- Macht und Wa. für Fehler 2. Art kann nur mit **konkreter Alternative** berechnet werden (z.B. $H_A: p = \frac{1}{3}$)


H0 true (p0 = 0.167)
$X \sim Bin(50,1/6)$
alpha = 0.0312
Signifikanzniveau
14
H1 true (p1 = 0.333)
$Y \sim Bin(50,1/3)$
1-beta = 0.8273
Macht $= P(Y \geq 14) = 1 - P(Y \leq 13)$
0.0
0.1
0.2
0.3
0.4
0
10
20
30
40
50
n = 50 / c = 14

Berechnung der Macht: 2 Arten

- Mit Theorie (wie im Bsp Zauberwürfel)
  + Genau, schnell berechnet
  - kompliziert → oft zu kompliziert

- Mit Simulation
  + ***fast immer möglich***
  - Programmieraufwand, ungenau(er), evtl. langsam

- Allgemeines Problem: Was genau soll die konkrete Alternative sein?

Zauberwürfel: Simulation

$counter = 0$

↓

Simuliere n Würfe mit $p = 1/3$

↓

Mache Binomialtest

↓

Falls Test verwirft:
$counter \to counter + 1$

Wiederhole 1000 mal

↓

$Macht = \frac{counter}{1000}$

Programmieren in R

- Vektoren, Matrizen

```r
v2 <- c(1,4,9)
```

- Listen

```r
myList <- list(a=4, b=v1, c=m, d="hallo")
```

- For-Schleifen

```r
for (i in 1:reps) {
  res[i] <- i
}
```

- Funktionen

```r
sumUp <- function(n=10) {
  ## Default-Input: 10
  res <- vector("numeric", n)
  for (i in 1:n) {
    res[i] <- i
  }
  sum(res)
}
```

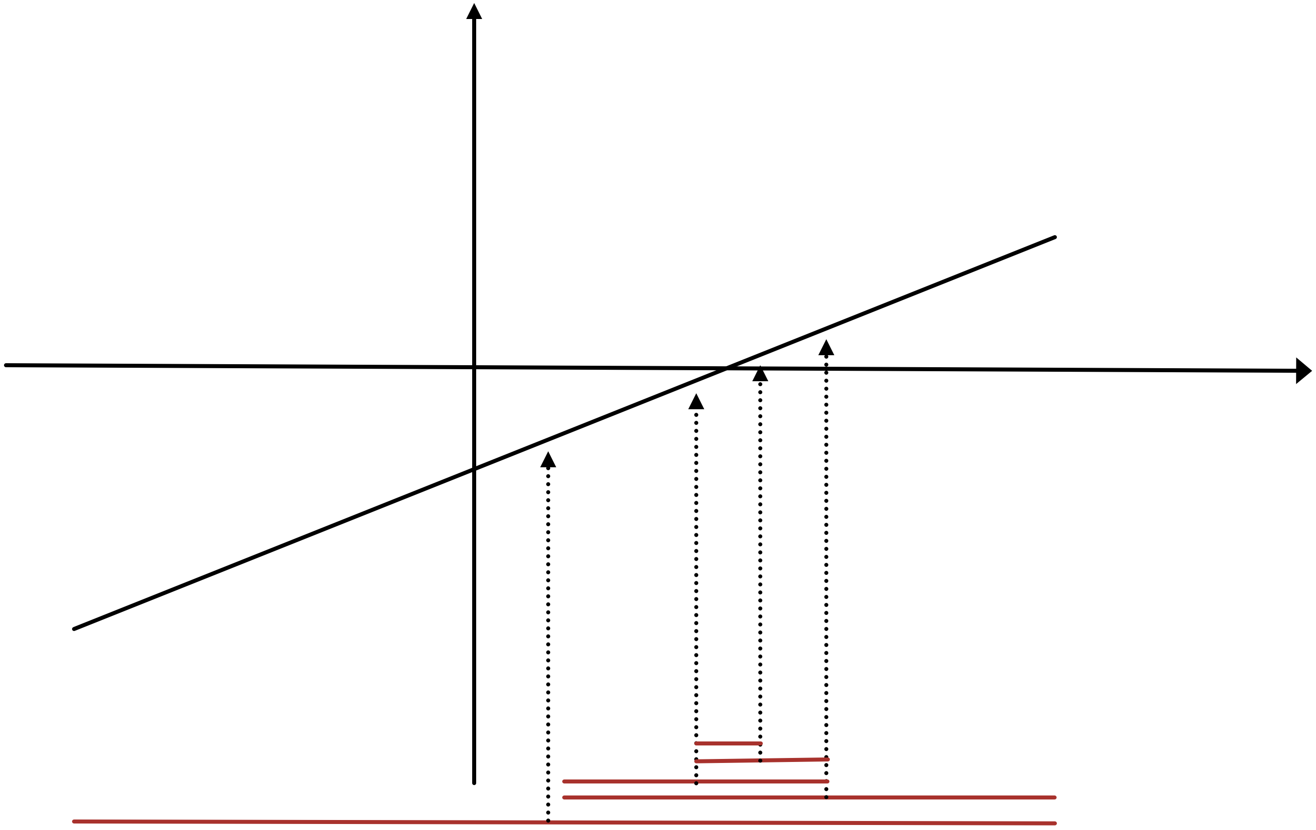
Macht beim Binomialtest

In einer Phase-2 klinischen Studie soll ein neues Medikament getestet werden.
$p_0 = 0.5$, $\alpha = 0.05$, $H_A\colon p > p_0$; bei $p_A = 0.7$ soll Macht $p_m = 0.9$ sein.
Wir können bis zu 100 Patienten rekrutieren. Ist diese Studie machbar ?

```
machtBinom <- function(n=50, reps = 1000, alpha = 0.05, pA = 0.7,
                       p0 = 0.5, alt = "two.sided") {
    res <- vector("numeric", reps)
    for (i in 1:reps) {
        x <- rbinom(n=1, size = n, prob = pA) ## Simuliere Daten
        tmp <- binom.test(x, n = n, p = p0, alternative = alt) ## Mache Test
        res[i] <- (tmp$p.value < alpha) ## Speichere Ergebnis
    }
    list(m = mean(res), s = sd(res)/sqrt(reps) )
}
```

Suche von Hand: Binäre Suche

Bsp: Nullstellensuche bei monotoner Funktion



Macht beim t-Test

```
machtTtest2 <- function(n1=20, n2=20, m1=0, m2=1, s1=1, s2=1, reps = 1000, alpha = 0.05) {
    ## Ungepaarter t-Test mit evtl. ungleichen Varianzen
    res <- vector("numeric", reps)
    for (i in 1:reps) {
        x <- rnorm(n=n1, mean = m1, sd = s1) ## Simuliere Daten
        y <- rnorm(n=n2, mean = m2, sd = s2)
        tmp <- t.test(x,y, paired = FALSE) ## Mache Test
        res[i] <- (tmp$p.value < alpha) ## Speichere Ergebnis
    }
    list(m=mean(res), s=sd(res)/sqrt(reps))
}
```

Macht bei 1-weg ANOVA

```
machtAnova1 <- function(n, mu, s=1, reps = 1000) {
    res <- vector("numeric", reps)
    for (i in 1:reps) {
        ## Simuliere Daten
        x <- rep(LETTERS[1:length(n)], times = n)
        y <- vector("numeric",0)
        for (j in 1:length(n)) {
            y <- c(y,rnorm(n[j], mean = mu[j], sd = s))
        }
        df <- data.frame(x=x, y=y)
        ## Mache Test
        sm <- summary(aov(y~x, data = df))
        pval <- sm[[1]][[5]][1]
        ## Speichere Ergebnis
        res[i] <- ( pval < alpha )
    }
    list(m = mean(res), s = sd(res)/sqrt(reps))
}
```

Macht bei Linearer Regression (Steigung)

Nicht prüfungsrelevant

```
machtLM <- function(n = 5, b0 = 0, b1 = 1, s=1, reps = 1000, alpha = 0.05) {
    res <- vector("numeric", reps)
    for (i in 1:reps) {
        ## Simuliere Daten
        x <- runif(n=n, min = -1, max = 1)
        y <- b0 + b1*x + rnorm(n, mean = 0, sd = s)
        ## Mache Test
        tmp <- lm(y~x)
        pval <- summary(tmp)$coefficients[2,4]
        ## Speichere Ergebnis
        res[i] <- (pval < alpha)
    }
    list(m = mean(res), s = sd(res)/sqrt(reps))
}
```

Macht beim Fisher-Test

Nicht prüfungsrelevant

```r
machtFisher <- function(n1 = 50, n2 = 50, p1 = 0.3, p2 = 0.1, reps = 1000, alpha = 0.05) {
    res <- vector("numeric", reps)
    for (i in 1:reps) {
        ## Simuliere Daten
        x1 <- rbinom(n=1, size=n1, prob=p1)
        x2 <- n1 - x1
        y1 <- rbinom(n=1, size=n2, prob=p2)
        y2 <- n2 - y1
        tt <- matrix(c(x1,y1,x2,y2),2,2)
        ## Mache Test
        tmp <- fisher.test(tt)
        pval <- tmp$p.value
        ## Ergebnis des Tests
        res[i] <- (pval < alpha)
    }
    list(m = mean(res), s = sd(res)/sqrt(reps))
}
```

Exkurs: Falls H_0 nicht verworfen wird...

- Es gibt viele Varianten von “Post-hoc Power Analysis”
  Bsp: H_0 wurde nicht verworfen, obwohl die Macht für $p = \frac{1}{3}$ 80% wäre; also muss der Würfel «ungefähr fair» sein
- Ungenaue Aussage; manche Varianten sind sogar falsch

- Besser: Vertrauensintervall
- Best practice:

Vor dem Experiment: Power Analyse → Stichprobengrösse
Nach dem Experiment: Vertrauensintervall

Siehe: “The Abuse of Power: The Pervasive Fallacy of Power Calculations for Data Analysis”
J.M. Hoenig, D.M. Heisey; The American Statistician, 2001, Vol. 55, No.1

Schwierigkeit in Praxis

- Welche konkrete Alternative? Parameter?
- Verschiedene realistische Alternativen simulieren
  → Gefühl für die **richtige Grössenordnung** der Stichprobengrösse