

# Exercises 6

Jan Hohenheim

## Setup

```
rm(list = ls())
library(ggplot2)
library(ggthemes)
library(dplyr)

##
## 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

library(purrr)
library(cowplot)

##
## Attaching package: 'cowplot'

## The following object is masked from 'package:ggthemes':
##
##   theme_map

library(latex2exp)
library(glue)

theme <-
  theme_solarized() #+
  #theme(text = element_text(family = "Helvetica Neue"))
```

## Problem 19 (Binomial distribution)

Suppose that among  $n = 95$  Swiss males, eight are red-green colour blind. We are interested in estimating the proportion  $p$  of people suffering from such disease among the male population.

a) Is a binomial model suitable for this problem?

Yes, using  $X \sim \text{Bin}(n_{\text{tot}}, \frac{8}{95})$ , where  $n_{\text{tot}}$  is the male population.

b) Calculate the maximum likelihood estimate (ML)  $p_{ML}$ .

```
n <- 95
x <- 8
p_ml <- x/n
glue("P_ML: {p_ml}")

## P_ML: 0.0842105263157895
```

d) Use the R functions `binom.test()` and `prop.test()` to compute two-sided 95%-confidence intervals for the exact and for the approximate proportion. Compare the results.

```
binom.test(x, n)

##
## Exact binomial test
##
## data: x and n
## number of successes = 8, number of trials = 95, p-value < 2.2e-16
## alternative hypothesis: true probability of success is not equal to 0.5
## 95 percent confidence interval:
## 0.03705905 0.15920028
## sample estimates:
## probability of success
## 0.08421053

prop.test(x, n)

##
## 1-sample proportions test with continuity correction
##
## data: x out of n, null probability 0.5
## X-squared = 64.042, df = 1, p-value = 1.218e-15
## alternative hypothesis: true p is not equal to 0.5
## 95 percent confidence interval:
## 0.0396893 0.1639164
## sample estimates:
## p
## 0.08421053
```

The normal approximation is a little bit higher than the exact test.

e) What is the meaning of the p-value?

The p-value tells us the likelihood of a data at least as extreme as the one presented existing under  $H_0$ . In this case,  $H_0$  postulates that  $p$  is equal to 0.5. We can see, that this is highly unlikely.

f) Compute the Wilson 95%-confidence interval and compare it to the confidence intervals from d).

```
wilson.ci <- function(x, n) {
  q <- 1.96
  mid <- (x + q^2 / 2) / (n + q^2)
  se <- sqrt(n) / (n + q^2) * sqrt(x / n * (1 - x / n) + q^2 / (4 * n))
  cbind(pmax(0, mid - q * se), pmin(1, mid + q*se))
}
```

```
}
wilson.ci(x, n)
```

```
##           [,1]      [,2]
## [1,] 0.0432876 0.1574538
```

The Wilson CI algorithm produces the most narrow CI.

## Problem 20 (A simple clinical trial)

A clinical trial is performed to compare two treatments, A and B, that are intended to treat a skin disease named psoriasis. The outcome shown in the following table is whether the patient's skin cleared within 16 weeks of the start of treatment.

Use  $\alpha = 0.05$  throughout this problem.

a) Compute for each of the two treatments a Wald type and a Wilson confidence interval for the proportion of patients whose skin cleared

```
wald.ci <- function(x, n) {
  q <- 1.96
  mid <- x / n
  se <- sqrt(x * (n - x) / n^3)
  cbind(max(0, mid - q * se), min(1, mid + q * se))
}
```

```
treatment_a <- data.frame(cleared = 9,
                          not_cleared = 18)
treatment_b <- data.frame(cleared = 5,
                          not_cleared = 22)
print("95% CIs for treatment A:")
```

```
## [1] "95% CIs for treatment A:"
```

```
print("Wald:")
```

```
## [1] "Wald:"
```

```
wald.ci(treatment_a$cleared, sum(treatment_a))
```

```
##           [,1]      [,2]
## [1,] 0.1555185 0.5111481
```

```
print("Wilson:")
```

```
## [1] "Wilson:"
```

```
wilson.ci(treatment_a$cleared, sum(treatment_a))
```

```
##           [,1]      [,2]
## [1,] 0.1864305 0.5217558
```

```
print("95% CIs for treatment B:")
```

```
## [1] "95% CIs for treatment B:"
```

```
print("Wald:")
```

```
## [1] "Wald:"
```

```
wald.ci(treatment_b$cleared, sum(treatment_b))
```

```
##           [,1]      [,2]  
## [1,] 0.03866186 0.3317085
```

```
print("Wilson:")
```

```
## [1] "Wilson:"
```

```
wilson.ci(treatment_b$cleared, sum(treatment_b))
```

```
##           [,1]      [,2]  
## [1,] 0.08180587 0.3669906
```

Treatment A seems to be way better

b) Test whether the risk difference is significantly different to zero (i.e.,  $RD = 0$ ). Use both an exact (test of proportions) and an approximated approach (See 6.4.1 in the script)

```
tab <- matrix(c(9, 5, 18, 22), ncol = 2, byrow = TRUE)  
colnames(tab) <- c('A', 'B')  
rownames(tab) <- c('cleared', 'not_cleared')  
tab
```

```
##           A  B  
## cleared    9  5  
## not_cleared 18 22
```

```
prop.test(tab, n = NULL)
```

```
##  
## 2-sample test for equality of proportions with continuity correction  
##  
## data:  tab  
## X-squared = 0.86786, df = 1, p-value = 0.3515  
## alternative hypothesis: two.sided  
## 95 percent confidence interval:  
## -0.1499190  0.5356333  
## sample estimates:  
##      prop 1      prop 2  
## 0.6428571 0.4500000
```

```
n1 <- sum(treatment_a)  
n2 <- sum(treatment_b)  
p1 <- treatment_a$cleared / n1  
p2 <- treatment_b$cleared / n2  
mu <- p1 - p2  
sigma <- sqrt(p1*(1 - p1)/n1 + p2*(1 - p2)/n2)  
c(qnorm(0.025, mu, sigma), qnorm(0.975, mu, sigma))
```

```
## [1] -0.0822542  0.3785505
```

The exact and the approximate approach both accept  $H_0$

c) Compute CIs for both, relative risk (RR) and odds ratio (OR).

```
print("RR:")
```

```
## [1] "RR:"
rr <- p1 / p2
s <- sqrt(1/tab[1,1] + 1/tab[2,1] - 1/sum(tab[1,]) - 1/ sum(tab[2,]))
exp(log(rr) + qnorm(c(.025, .975))*s)
```

```
## [1] 1.070738 3.025951
```

```
print("OR:")
```

```
## [1] "OR:"
or <- tab[1]*tab[4]/(tab[2]*tab[3])
s <- sqrt(sum(1/tab))
exp(log(or) + qnorm(c(.025, .975))*s)
```

```
## [1] 0.6251434 7.7422235
```

d) How would the point estimate of the odds ratio change if we considered the proportions of patients whose skin did not clear?

```
print("OR of not cleared skin:")
```

```
## [1] "OR of not cleared skin:"
or <- (tab[2]/tab[1])/(tab[4]/tab[3])
s <- sqrt(sum(1/tab))
exp(log(or) + qnorm(c(.025, .975))*s)
```

```
## [1] 0.1291619 1.5996330
```