Homework 8

Pascal Pilz, k12111234

2023-11-25

Exercise 19

Since we have 25 samples and we have one of fields is less than 5, we use Fisher's exact test.

```r
data <- matrix(c(4, 5, 13, 3), 2, 2)
fisher.test(data)
```

```
##
##  Fisher's Exact Test for Count Data
##
## data:  data
## p-value = 0.087
## alternative hypothesis: true odds ratio is not equal to 1
## 95 percent confidence interval:
##  0.02040914 1.54909085
## sample estimates:
## odds ratio
##  0.1998047
```

We can see that there is no significant difference.

We need to use McNemar because we have paired data!

Exercise 20

```
data <- matrix(c(5, 1, 2, 6), 2, 2)
```

We want to compare the asymptotic Chi-square test, Chi-square with Yate's correction, and exact Fisher test.

Since we have a sample size of XX and cells with values of less than 5 we already know that the assumption of the Chi-square test are violated.

Chi-square

```
chisq.test(data, correct=FALSE)
```

```
## Warning in chisq.test(data, correct = FALSE): Chi-squared approximation may be
## incorrect

##
##  Pearson's Chi-squared test
##
## data:  data
## X-squared = 4.6667, df = 1, p-value = 0.03075
```

Chi-square with Yate's correction

```
chisq.test(data, correct=TRUE)
```

```
## Warning in chisq.test(data, correct = TRUE): Chi-squared approximation may be
## incorrect

##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  data
## X-squared = 2.625, df = 1, p-value = 0.1052
```

Fiher's exact test

```
fisher.test(data)
```

```
##
##  Fisher's Exact Test for Count Data
##
## data:  data
## p-value = 0.1026
## alternative hypothesis: true odds ratio is not equal to 1
## 95 percent confidence interval:
##    0.7203391 817.9820107
## sample estimates:
## odds ratio
##   11.71745
```

Conclusion

As we can, the p-value for the corrected Chi-square test and Fisher's exact test are quite close, whereas the uncorrected Chi-square test produces a drastically different result.

Exercise 21

```r
p <- 100
n <- 200

pre <- p / (p + n)

tp <- 90
fp <- 30
tn <- 170
fn <- 10

se <- tp / p
sp <- tn / n

ppv_1 <- tp / (tp + fp)
ppv_2 <- (se * pre) / (se * pre + (1 - sp) * (1 - pre))

npv_1 <- tn / (tn + fn)
npv_2 <- (sp * (1 - pre)) / (sp * (1 - pre) + (1 - se) * pre)

cat(sprintf(" sensitivity: %.3f\n", se),
    sprintf("specificity: %.3f\n", sp),
    "\n",
    sprintf("positive predictive value (excluding prevelance): %.3f\n", ppv_1),
    sprintf("positive predictive value (including prevelance): %.3f\n", ppv_2),
    "\n",
    sprintf("negative predictive value (excluding prevelance): %.3f\n", npv_1),
    sprintf("negative predictive value (including prevelance): %.3f\n", npv_2))
```

```
##  sensitivity: 0.900
##  specificity: 0.850
##
##  positive predictive value (excluding prevelance): 0.750
##  positive predictive value (including prevelance): 0.750
##
##  negative predictive value (excluding prevelance): 0.944
##  negative predictive value (including prevelance): 0.944
```