STATMED Lecture 3: Estimates and confidence intervals

Matti Pirinen

15.8.2024

Let's start the same way as in Lecture 2: Suppose we have 100 pairs of same-sex twins that are discordant for psoriasis and we know that BMI is higher for the psoriatic twin in 71 out of 100 pairs. Does that sound like higher BMI goes with psoriasis status in the population where these data come from? How to statistically quantify the association between BMI and psoriasis in these data?

Previously, we quantified (or "tested") how consistent our observation was with a **null hypothesis** saying that "psoriasis status is independent of BMI". We did this by computing a P-value, that was the tail probability of observing at least as extreme an observation as what we had observed, assuming that the null hypothesis was true. This value gave an idea how clearly the observation is pointing to a possible deviation from the null hypothesis, with smaller P-values indicating more evidence against the null hypothesis. But P-value is only a statistical summary and typically a more interesting quantity is the actual value of the unknown variable that we want to estimate. Here that value is the proportion of twin pairs, where higher BMI and psoriasis go together.

What can we say about the proportion of pairs where the psoriatic twin has the higher BMI? Should we say that it is 71%? Yes, in these data it happens to be exactly 71%, but what about more generally among a larger population of twin pairs. Maybe it would be 68% or 75%? One of the main goals of statistics is to make statements about the population (such as all twin pairs) based on a relatively small sample from that population (such as $n = 100$ twin pairs). Therefore, we need to quantify the uncertainty that exists because the sample we have observed is only a subset of the target population rather than the whole population.

To demonstrate the sampling variation resulting from collecting only a subset of the population, let's simulate 10 data sets of 100 twin pairs in each under the assumption that the true proportion parameter in the target population is $p = 0.71$.

```
p = 0.71 #true proportion in population
n.trials = 100 #sample size in each data set
rbinom(10, size = n.trials, prob = p) #returns 10 values, each btw 0,..,100.
```

```
##  [1] 76 71 69 78 73 74 74 70 67 68
```

We see that the observed value varies, and is exactly 71 in only one case above even if the true proportion in the simulation is exactly 0.71.

To get a comprehensive picture of this statistical variation, let's generate 10,000 example data sets and show results as a table. (We don't want to print out vector of 10,000 values on screen but rather collect the result into a table where each observed value is represented only once.)

```
p = 0.71 #true proportion in population
n.experiments = 10000 #no. of data sets
n.trials = 100 #sample size in each dat set

x = rbinom(n.experiments, size = n.trials, prob = p) #returns 10000 values, each btw 0,..,100.
```

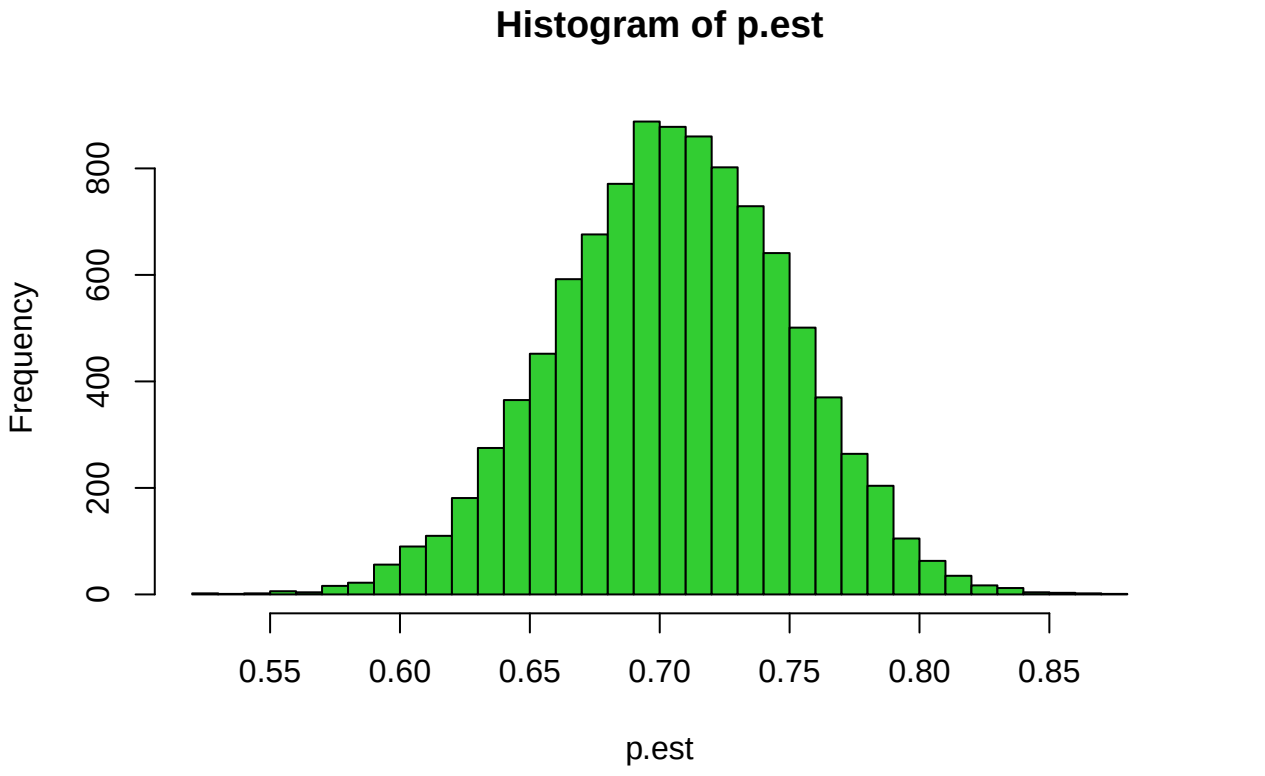
```r
p.est = x / n.trials #10000 estimates ('est') of proportion parameter p
table(p.est)/n.experiments #frequency distribution of estimates
```

```
## p.est
##   0.52   0.53   0.54   0.55   0.56   0.57   0.58   0.59    0.6   0.61   0.62
## 0.0001 0.0001 0.0001 0.0002 0.0006 0.0004 0.0016 0.0022 0.0056 0.0090 0.0110
##   0.63   0.64   0.65   0.66   0.67   0.68   0.69    0.7   0.71   0.72   0.73
## 0.0181 0.0275 0.0365 0.0452 0.0592 0.0676 0.0771 0.0888 0.0878 0.0860 0.0802
##   0.74   0.75   0.76   0.77   0.78   0.79    0.8   0.81   0.82   0.83   0.84
## 0.0729 0.0641 0.0501 0.0370 0.0264 0.0204 0.0105 0.0063 0.0035 0.0017 0.0012
##   0.85   0.86   0.87   0.88
## 0.0004 0.0003 0.0002 0.0001
```

In all these data sets, the true population parameter was 71%, but only in less than 1 in 10 of the data sets the observed proportion is exactly 71%. The observed proportion varies quite a bit due to random sampling (here range is from 0.52 to 0.88). Clearly, when we have observed a single **point estimate** of 71% from one data set of $n = 100$ samples, we need to report also something else than the point estimate 71%, to give an idea how much the true population proportion might vary from our point estimate due to sampling variation.

Let's look at the distribution of point estimates when the true value is 0.71.

```r
hist(p.est, breaks = 40, col = "limegreen")
```




```r
mean(p.est)
```

```
## [1] 0.710031
```

```r
sd(p.est)
```

```
## [1] 0.04491359
```

Mean is where we would hope it to be, that is, near the true value of 0.71. We say that our estimator is **unbiased**, that is, the estimates are "correct on average".

Standard deviation of the estimates describes how much, on average, the estimate varies around its mean. We call this standard deviation as **standard error** (SE) of the estimator. That is,

> SE of a parameter = "standard deviation of the sampling distribution of the parameter estimates".

We also have a formula for the standard error for the mean of binomial sampling from Bin(n, p):

$$\mathrm{SE}_{\mathrm{bin}} = \sqrt{\frac{p\,(1-p)}{n}}.$$

```r
SE = sqrt( p*(1-p) / n.trials)
SE
```

```
## [1] 0.04537621
```

Compare this to `sd(p.est)`, which was an empirical estimate over 10,000 samples. They do agree very well, so we are happy to use the formula in the future (except when n is small, when formula does not work that well).

Let's see what is the interval within which, say, 95% of the point estimates fall, when the true parameter is 71%. Let's use `quantile()` function to get empirical cut-points of 2.5% and 97.5% of the distribution, since 95% of the distribution is between them.

```r
quantile(p.est, c(0.025, 0.975) ) #we can give two cut-points the same time as a vector
```

```
##  2.5% 97.5%
##  0.62  0.79
```

So if the true parameter is 0.71, then in 19 out of 20 cases (that is 95% of cases) we see a point estimate in the range of 0.62,..,0.79.

If we had not done this empirical simulation of 10,000 binomial experiments, we could still have written down the standard 95% **confidence interval** (CI) estimate based on the SE, using the rule that 95% CI is given by (point.estimate +/- 1.96*SE).

```r
c(0.71 - 1.96 * SE, 0.71 + 1.96 * SE) #Let's use 0.71 as our point estimate here.
```

```
## [1] 0.6210626 0.7989374
```

This is a good approximation to 95% CI from the empirical data when `n.trials` is large (like in our case). Often it is enough to remember that 95% interval around the point estimate is given by adding 2 SEs on both sides of the estimate.

More realistically, suppose that we do not know the true value for p, but we just have a point estimate `p.est` computed from the data. What happens, if we surround the estimate by its 95% CI computed from SE?

```r
#currently p.est has 10,000 point estimates, each from a data set of size 100
SE.est = sqrt( p.est * (1 - p.est) / n.trials) #vector of 10,000 SE estimates, one for each data set
ci.low = p.est - 1.96 * SE.est #vector of lower bounds of 95% CIs
ci.up = p.est + 1.96 * SE.est #vector of upper bounds of 95% CIs
```

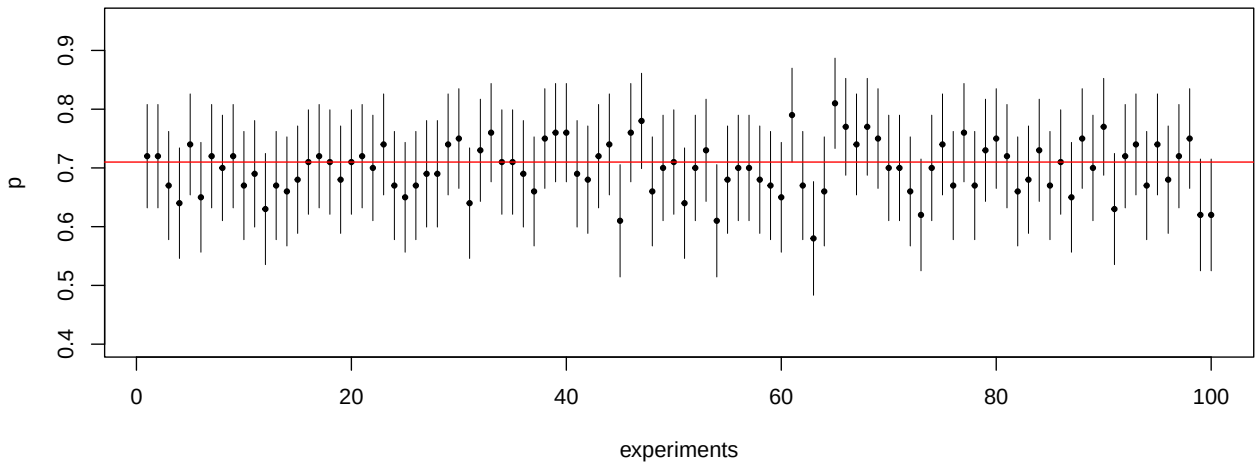
For example, for the first data set we have the following values computed. (Note that we can name values in a vector as follows.)

```r
c(p.est = p.est[1],
  SE = SE.est[1],
  ci.low = ci.low[1],
  ci.up = ci.up[1])
```

```
##      p.est         SE     ci.low      ci.up
## 0.72000000 0.04489989 0.63199622 0.80800378
```

Let's then visualize the first 100 point estimates with CIs, and compare them to the true value of 0.71 (red line). We first use `plot(NULL)` command to make an empty plot with suitable x-axis and y-axis limits (`xlim` and `ylim` parameters) and labels (`xlab` and `ylab` parameters). Then we plot all the point estimates using `points()` that takes in x-coordinates and y-coordinates and we use `pch=19` to make solid plotting symbols and `cex = 0.5` to make size 50% of normal. Finally, we draw CIs by using `arrows()` that takes in four vectors of coordinates (x-start, y-start, x-end, y-end) and draws lines between the start and end points. `code=0` means that the lines have no symbols at the ends and `lwd = 0.5` means that the line width is 50% of the normal.

```r
n.to.plot = 100 #how many to plot -- plotting them all might be too messy
plot(NULL, xlim = c(1, n.to.plot), ylim = c(0.4, 0.95), ylab = "p", xlab = "experiments")
points(1:n.to.plot, p.est[1:n.to.plot], pch = 19, cex = 0.5)
arrows(1:n.to.plot, ci.low[1:n.to.plot], 1:n.to.plot, ci.up[1:n.to.plot], code = 0, lwd = 0.5)
abline(h = p, col = "red") #add horizontal line at y = p i.e. y=0.71
```



If we now would had observed only one of these 10,000 data sets, and would report its point estimate and its 95% CI, in how many cases would the 95% CI cover the true value of 0.71? Let's make a logical TRUE / FALSE vector for each data set that is TRUE, if the 95% CI covers the true value, and is FALSE otherwise.

```r
covers = (ci.low < p & ci.up > p) #TRUE if p is btw ci.low and ci.up, otherwise FALSE
table(covers)/n.experiments
```

```
## covers
##  FALSE   TRUE
## 0.0645 0.9355
```

The coverage of the 95% CIs here is quite close to 95% (actually slightly smaller here, about 93.6%). In general, when `n.trials` is large enough, about 95% of the CIs cover the true parameter value. This is the typical interpretation of the 95% confidence interval. In other words, in about 95% of the cases where we compute a point estimate and its 95% CI, the true parameter value is within the confidence interval. We cannot know for sure whether the true parameter value is within any one CI, but we know that, on average, it is there in 95% of the cases.

We could also consider confidence intervals for other coverages than 95%. If we wanted a CI for coverage θ, then we change the factor 1.96 multiplying SE to `abs(qnorm((1-theta) / 2))`.

```r
theta = 0.95 #95% CI
abs(qnorm((1-theta) / 2))
```

```
## [1] 1.959964
```

```r
theta = 0.50 #50% CI
abs(qnorm((1-theta) / 2))
```

```
## [1] 0.6744898
```

Thus 50% confidence intervals are defined when we add 0.67*SE on both sides of the point estimates.

Summary: A confidence interval around the point estimate gives an idea where the true population parameter may be. The interpretation of 95% CI is that if we were to repeat the experiment over and over again, then 95% of the CIs would cover the true parameter value and in the remaining 5% the CI would not cover the true value. When CIs are wide, we are very uncertain where the true parameter value may be and our experiment has simply not been very informative.

It is crucial to report CIs in addition to the parameter estimate. Result "parameter estimate is 5.6 (95%CI 0.0 .. 24.5)" is very different from "parameter estimate is 5.6 (95%CI 5.2 .. 6.0)" even though both could be (uninformatively) summarized by saying "parameter estimate is 5.6". The former leaves us with a lot of uncertainty about the actual value of the parameter whereas the latter is more accurate.

Often a point estimate and an SE or CI are more informative than the P-values under the null hypothesis (reading from BMJ), because estimates explicitly tell about the value of the parameter of interest whereas P-values tell only about statistical significance, which may not always have practical significance (as we saw earlier).

Example 3.1 **(1)** Over the last year, there has been 316 appendectomy patients of which 180 were women and 136 were men. What is the point estimate of the proportion of women patients? What is the standard error of that estimate? How do you compute the 95% confidence interval for the estimate? How do you explain these results in plain language?

Point estimate of the proportion of women is the observed number of women divided by the number of all patients.

```r
n.all = 316
n.w = 180
p.w.est = n.w/n.all
p.w.est #point estimate
```

```
## [1] 0.5696203
```

SE comes from the formula above, where we replace the true proportion value with our current point estimate:

```r
se = sqrt( p.w.est*(1-p.w.est)/n.all )
se
```

```
## [1] 0.0278532
```

95% CI results when we add 1.96*SE on both sides of the point estimate:

```r
c(p.w.est - 1.96*se, p.w.est + 1.96*se)
```

```
## [1] 0.5150280 0.6242125
```

Explanation: We have observed that 180 out of 316 patients are women. We use these data to estimate the proportion of women among the whole patient population. The point estimate for the proportion of women patients is 180/316 = 57% (or equivalently 0.57). The 95% confidence interval is (51.5%,..,62.4%). This gives an idea of a range where the true population parameter might be based on the observed counts. The exact interpretation of the 95% confidence interval is that in 95% of the data sets, the CI covers the true parameter value but in the remaining 5% the true value lies outside the CI.

(2) How do the values from (1) change if all sample sizes are multiplied by ten, that is, there are 3160 patients, 1800 are women and 1360 are men? Explain intuitively why something changes and why something else does not change.

```r
n.all = 3160
n.w = 1800
p.w.est = n.w/n.all
p.w.est #point estimate
```

```
## [1] 0.5696203
```

```r
se = sqrt( p.w.est*(1-p.w.est)/n.all )
se
```

```
## [1] 0.008807955
```

```r
c(p.w.est - 1.96*se, p.w.est + 1.96*se)
```

```
## [1] 0.5523567 0.5868838
```

The point estimates stays exactly the same when the counts are multiplied by 10. Standard error gets smaller, reflecting the fact that a larger sample size n means less error in the estimate. Similarly, the 95%CI gets narrower reflecting that with a larger sample size we have more information about the parameter and therefore our estimate is also more accurate.

Confidence intervals in `binom.test()` Previously, we did the binomial test using the function `binom.test()`. Let's revisit it now to see what it says about confidence intervals.

```
binom.test(71, n = 100, p = 0.5)
```

```
##
##  Exact binomial test
##
## data:  71 and 100
## number of successes = 71, number of trials = 100, p-value = 3.216e-05
## alternative hypothesis: true probability of success is not equal to 0.5
## 95 percent confidence interval:
##  0.6107340 0.7964258
## sample estimates:
## probability of success
##                   0.71
```

In addition to the P-value, it also gives 95% CI and the point estimate ("sample estimate"). We could alter the coverage of the CI by parameter `conf.level` that, by default, is 0.95.

```
binom.test(71, n = 100, p = 0.5, conf.level = 0.5)
```

```
##
##  Exact binomial test
##
## data:  71 and 100
## number of successes = 71, number of trials = 100, p-value = 3.216e-05
## alternative hypothesis: true probability of success is not equal to 0.5
## 50 percent confidence interval:
##  0.6730333 0.7439906
## sample estimates:
## probability of success
##                   0.71
```

Next we move to more general tests about proportion parameters.

Proportion test

We have used `binom.test()` to do binomial test. A more general function is `prop.test()` that is only an approximation to the binomial distribution but, importantly, can be used also to compare two or more samples against each other. Let's study `prop.test()` in more detail. Type `?prop.test` on console to read help.

`prop.test` for one group. Let's first revisit question of 71 successes out of 100 trials. First with the two-sided test and then with the one sided test that assumes that the proportion is > 0.5.

```
prop.test(71, n = 100, p = 0.5, alternative = "two.sided", conf.level = 0.95)
```

```
##
##  1-sample proportions test with continuity correction
##
```

```
## data:  71 out of 100, null probability 0.5
## X-squared = 16.81, df = 1, p-value = 4.132e-05
## alternative hypothesis: true p is not equal to 0.5
## 95 percent confidence interval:
##  0.6093752 0.7942336
## sample estimates:
##    p
## 0.71
```

```
#And with one-sided test
prop.test(71, n = 100, p = 0.5, alternative = "greater", conf.level = 0.95)
```

```
##
##  1-sample proportions test with continuity correction
##
## data:  71 out of 100, null probability 0.5
## X-squared = 16.81, df = 1, p-value = 2.066e-05
## alternative hypothesis: true p is greater than 0.5
## 95 percent confidence interval:
##  0.6253853 1.0000000
## sample estimates:
##    p
## 0.71
```

In the output, `X-squared` is the value of the test statistics that leads to the P-value and `df` is "degrees of freedom" of the test (here 1 as there is one free parameter, `p`). The P-values are close to the exact values given by `binom.test()`.

Note how the null value is the same, 0.5, in both tests above but the alternative hypothesis is different. As expected, the 2-sided P-value is twice the 1-sided P-value. Note also how the confidence interval changes with the alternative hypothesis. (It is not recommended to use CIs for one-sided alternatives as their interpretation is complex. As a general rule, use two-sided tests and CIs.)

`prop.test` for two groups. The benefit of `prop.test()` over `binom.test()` is that we can also compare two samples against each other. For example, suppose that in a similar study design in Norway it has been observed that the twin with psoriasis has a higher BMI in 344 times out of 497 twin pairs studied. Let's make a data matrix, where rows are countries and column 1 counts when psoriasis goes with the higher BMI and column 2 counts the opposite cases.

```
x = rbind(FIN = c(71, 100-71), NOR = c(344, 497-344)) #bind rows together
x #show the data to make sure we got it correct
```

```
##     [,1] [,2]
## FIN   71   29
## NOR  344  153
```

```
prop.test(x)
```

```
##
##  2-sample test for equality of proportions with continuity correction
##
## data:  x
## X-squared = 0.05508, df = 1, p-value = 0.8144
```

```
## alternative hypothesis: two.sided
## 95 percent confidence interval:
##  -0.08591631  0.12161047
## sample estimates:
##    prop 1    prop 2
## 0.7100000 0.6921529
```

`prop.test()` runs with such a 2x2 matrix and tests whether the proportions in rows are similar to each other. In this case, there is no statistically significant difference between the proportions in Finland (0.710) and Norway (0.692). The CI given in the output is for the difference between the two proportions. So we may say that "the point estimate for the difference between Finland and Norway is 0.71-0.692 = 0.018 and its 95% confidence interval is from -0.086 to 0.122. In particular, 0 is pretty much in the middle of the CI and hence there is no evidence that the difference would be different from 0. The large P-value (0.8144) also tells that there is no evidence for deviation from the null hypothesis that the proportions are the same in the two populations.

Difference between proportions. (Here is an explanation what `prop.test()` does under the hood, but you don't need this in the exercises.) Consider two populations and assume that proportions of cases (e.g. carriers of a particular disease) are p_1 and p_2. We have access to the samples of sizes n_1 and n_2 and estimate $\widehat{p}_1 = x_1/n_1$ and $\widehat{p}_2 = x_2/n_2$ where x_i is the number of observed cases in population $i = 1, 2$. ("Hat"-notation on top of p_1 and p_2 denotes the estimates that we have computed for these two parameters. We do not know the true values of p_1 and p_2 in the populations but we can estimate them using the observed data by $\widehat{p}_1$ and $\widehat{p}_2$ as defined above.) We compute the difference $\widehat{d} = \widehat{p}_1 - \widehat{p}_2$ and want to do statistical inference about that difference. We have that

$$\text{SE}\,(\widehat{p}_1 - \widehat{p}_2) \approx \sqrt{\frac{\widehat{p}_1(1-\widehat{p}_1)}{n_1} + \frac{\widehat{p}_2(1-\widehat{p}_2)}{n_2}},$$

For testing the null hypothesis that $p_1 = p_2$, we use the test statistic

$$\frac{\widehat{p}_1 - \widehat{p}_2}{\sqrt{\widehat{p}(1-\widehat{p})\left(\frac{1}{n_1} + \frac{1}{n_2}\right)}}, \text{ where } \widehat{p} = \frac{x_1 + x_2}{n_1 + n_2}.$$

Under the null hypothesis that $p_1 = p_2$, this test statistic follows approximately a standard Normal distribution (Topic of the next lecture), and we can derive a P-value from the tail probabilities of that distribution.

Example 3.2

(1) Over the last year, in Hospital A, there has been 316 appendectomy patients of which 180 women and 136 men. Use `prop.test()` to test whether both sexes are equally represented among appendectomy patients.

```
#Here we run a one-sample prop.test by specifying the observed number of women,
# total sample size 'n' and null hypothesis success rate 'p'
prop.test(180, n = 316, p = 0.5)
```

```
##
##  1-sample proportions test with continuity correction
##
## data:  180 out of 316, null probability 0.5
## X-squared = 5.8513, df = 1, p-value = 0.01557
## alternative hypothesis: true p is not equal to 0.5
## 95 percent confidence interval:
##  0.5129281 0.6245919
## sample estimates:
##         p
## 0.5696203
```

(2) In another hospital B, there has been 245 patients of which 134 women and 111 men. Use `prop.test()` to test whether the proportion of women is different between the two hospitals A and B.

```
#We make a matrix where one row is one hospital and columns are
# numbers of women and men.
x = rbind(A = c(180, 136), B = c(134, 111))
colnames(x) = c("women","men")
x
```

```
##   women men
## A   180 136
## B   134 111
```

```
prop.test(x)
```

```
##
##  2-sample test for equality of proportions with continuity correction
##
## data:  x
## X-squared = 0.2034, df = 1, p-value = 0.652
## alternative hypothesis: two.sided
## 95 percent confidence interval:
##  -0.06380003  0.10916299
## sample estimates:
##    prop 1    prop 2
## 0.5696203 0.5469388
```

We see that the observed proportions of women are 0.570 and 0.547, which are not statistically different from each other (P-value 0.65) and the 95%CI for the difference is (-0.064, 0.109), containing 0 within it.

prop.test for more than two groups. What if we have more than two groups to compare? Suppose three Finnish hospitals have done knee replacement operations in 2021 and in two year check-up the patients reported knee pain as follows.

Hospital	no pain	pain	total
A	113	198	312
B	100	111	211
C	207	215	422

Are there differences between the hospitals?

```
x = rbind(A = c(113,198), B = c(100,111), C = c(207,215))
colnames(x) = c("nopain","pain")
x
```

```
##   nopain pain
## A    113  198
## B    100  111
## C    207  215
```

```
prop.test(x)
```

```
##
##  3-sample test for equality of proportions without continuity correction
##
## data:  x
## X-squared = 12.653, df = 2, p-value = 0.001789
## alternative hypothesis: two.sided
## sample estimates:
##    prop 1    prop 2    prop 3
## 0.3633441 0.4739336 0.4905213
```

Here `prop.test()` did a test of a null hypothesis that all proportions are the same. A small P-value indicates that there are likely some statistically detectable differences betweent the hospitals. However, a small P-value alone is not very informative since it does not tell which of the proportions look different from other(s) nor how large are the differences. Therefore, pairwise comparisons might be more informative here. Note that here we can't estimate a single value that represents the "difference between the groups" since there are more than 2 groups. Hence, we don't have a confidence interval either for this three-group test.

Chi-square test

Suppose we divide the painless patients into two groups: fully functional and partially functional, making a 3x3 table

Hospital	func+	func-	pain	total
A	62	51	198	312
B	50	50	111	211
C	84	123	215	422

```
x = rbind(A = c(62,51,198), B = c(50,50,111), C = c(84,123,215))
colnames(x) = c("func+", "func-", "pain")
x
```

```
##   func+ func- pain
## A    62    51  198
## B    50    50  111
## C    84   123  215
```

We can't use `prop.test()` anymore since there are more than two columns, that is, we are not observing a single proportion per hospital anymore. Now we can ask more generally whether every row has the

same distribution of patients into the three categories, that is, are the proportions of the three outcomes same across the hospitals. Another way to put this is to ask whether the rows and columns of the matrix are *independet*, that is, whether the distribution of the counts in a cell on row i column j in the table is determined by the product of probabilities of row i and of column j, for all i and j. We can test this by a general **chi-square test** for contingency tables using `chisq.test()`. Read `?chisq.test`.

```
chisq.test(x)
```

```
##
##  Pearson's Chi-squared test
##
## data:  x
## X-squared = 19.019, df = 4, p-value = 0.0007792
```

`chisq.test()` takes in a matrix and returns a P-value under the null hypothesis that the rows and columns are independent of each other. Here it is saying that the rows and columns do not seem independent (P-value is small), but otherwise it is not very informative, since it does not indicate where the possible differences are. It is likely more useful to check the numerical proportions per hospital to consider which groups and in which hospitals seem to differ from the other hospitals.

```
x/rowSums(x)
```

```
##       func+     func-      pain
## A 0.1993569 0.1639871 0.6366559
## B 0.2369668 0.2369668 0.5260664
## C 0.1990521 0.2914692 0.5094787
```

From these proportions it looks like the hospital A may be different from B and C, although we should also look at the confidence interval of each proportion estimate to properly account for the uncertainty in these point estimates.

How `chisq.test()` does the chi-square (χ^2) test for a contingency table?

(1) Calculate the expected frequency (E_{ij}) for the observation in row i and column j of the $r \times c$ contingency table: $E_{ij} = \frac{i\text{th row total} \times j\text{th col total}}{\text{table total}}$

(2) For each cell in the table calculate the difference between the observed value and the expected value $(O_{ij} - E_{ij})$.

(3) Square each difference and divide the resulting quantity by the expected value $(O_{ij} - E_{ij})^2/E_{ij}$.

(4) Sum all of these $r \times c$ values to get a single number, the χ^2 statistic `X.sq`.

(5) Compare this number with the chi-squared distribution with the following degrees of freedom: df $= (r-1) \times (c-1)$. P-value is in R given by `1-pchisq(X.sq, df = df)`.

Chi-square test is applicable if almost all expected cell counts are at least 5. If there are smaller counts, then exact methods such as Fisher's test should be used instead.

Example 3.3

(1) Over the last year, in Hospital A, there has been 316 appendectomy patients of which 180 women and 136 men. In another hospital B, there has been 245 patients of which 134 women and 111 men. Use Chi-square test to test independence between rows (hospitals) and columns (sex). Compare the P-value to one from `prop.test()`.

```
x = rbind(A = c(180,136), B = c(134,111))
colnames(x) = c("women","men")
x
```

```
##   women men
## A   180 136
## B   134 111
```

```
chisq.test(x)
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  x
## X-squared = 0.2034, df = 1, p-value = 0.652
```

```
prop.test(x)
```

```
##
##  2-sample test for equality of proportions with continuity correction
##
## data:  x
## X-squared = 0.2034, df = 1, p-value = 0.652
## alternative hypothesis: two.sided
## 95 percent confidence interval:
##  -0.06380003  0.10916299
## sample estimates:
##    prop 1    prop 2
## 0.5696203 0.5469388
```

Thus, in case where both `prop.test()` and `chisq.test()` are possible, they give the same test result. But `prop.test()` is more informative since it also tells the CI for the difference between the two proportions as well as the point estimates of the two proportions.

Summary of the tests we have used so far:

(1) `binom.test()` does a single sample binomial test, that is, compares the observed number of successes to the expectation under the null hypothesis success probability value, and returns a P-value, a point estimate and a confidence interval for the proportion.

(2) `prop.test()` is an approximation to `binom.test()` in case of one sample, but can also compare many proportions against each other, not only to the null value as `binom.test()`. For the case of two groups, it gives a CI for the difference between the two success proportions.

(3) `chisq.test()` takes in any matrix and compares it to the null hypothesis that all the rows and columns are independent of each other, that is, all the rows have the same distribution of how the observations are distributed into the columns. A very general test, but returns only a P-value and not any informative parameter about what kind of a difference from the null hypothesis might have been observed.

(4) `fisher.test()` is suitable for tables that have small counts and there it is exact rather than an approximation as `chisq.test()`. In tables with larger counts, it gives similar results to `chisq.test()`.