Exercises for Day 2 – Solution
Applied Statistics & Statistical methods for SCIENCE

Anton Rask Lundborg

November 2023

Exercise 2.1 Wilcoxon and t-tests: Syntax possibilities

Problem

In this exercise we will work with data example 2 used on course Day~1 in Exercise 1.2 and Exercise 1.4. Recall that this example was concerned with tenderness of pork from two pH-groups (high and low pH) after two chilling methods (tunnel and fast chilling). In the organization (tidy data on wide format) of the dataset in the text file `dataExample2.txt` there are four variables:

Pig, pH.group, Tunnel, Fast

The objective of this exercise is to pay attention to the various syntaxes that may be used in the basic test functions like `t.test()` and `wilcox.test()`. First we will do this "theoretically' ', i.e. *without* opening R or RStudio. Instead, you should simply imagine that the dataset is available in a data frame called `example2` containing the four variables listed above. Now suppose that for the pigs that have been tunnel-chilled you want to compare the tenderness in the low pH-group against the tenderness in the high pH-group. Which of the following 7 calls do this? Explain your reasoning for each line of code.

```
t.test(example2$Tunnel, example2$Fast, paired = TRUE)
with(example2, wilcox.test(Tunnel, Fast, paired = TRUE))
with(example2, wilcox.test(Tunnel ~ pH))
t.test(Tunnel ~ pH, data = example2, var.equal = TRUE)
t.test(Tunnel ~ pH, data = example2[-3, ])
with(example2, t.test(Tunnel[pH == "low"], Tunnel[pH== "high"]))
wilcox.test(Tunnel ~ pH, data = example2, paired = TRUE)
```

Two of the lines of code do another analysis. Which ones and which analysis do they perform instead? Finally, there is one line of code performing a nonsensical test. Which one, and why does this test not make sense? If you are stuck, see the next page for hints and remarks!

Hints and remarks:

- The idea of `with(my.data, my.expression)` is that the variables inside the data frame '`my.data`' are available when computing '`my.expression`'. For instance, in the second of the 7 lines, the variables '`Tunnel`' and '`Fast`' inside the data frame '`example2`' are available to the Wilcoxon test.

  Thus, the function `with()` can be used just like a `data`-option, when the latter is not available.

- Both `t.test()` and `wilcox.test()` accept two different input syntaxes called the *"Default S3 method"* and *"S3 method for class 'formula'"* on the help pages (see `?t.test`).

- For two-sample tests the *"Default S3 method"* requires the samples to be given in two vectors. If we want these vectors to be taken from a data frame one possibility is to use the `with()` function as described above.

- One advantage of the *"S3 method for class 'formula'"* is that the data frame may be specified in the data-option (see the 4th, 5th and 7th lines of code above. Consider what happens in the 5th line of code?).
- For more complicated functions like `lm()`, `glm()`, and `nlme::lme()` only the *formula method* makes sense.
- To get an improved feeling of the syntax you, of course, are welcome to open RStudio and try the code.

Solution

We load the dataset and print the first few lines to recall the structure:

```
example2 <- read.table("dataExample2.txt", header = TRUE, dec = ",")
head(example2)
```

```
##   Pig  pH Tunnel Fast
## 1   1 low   7.22 5.56
## 2   2 low   3.11 3.33
## 3   3 low   7.44 7.00
## 4   4 low   4.33 4.89
## 5   5 low   6.78 6.56
## 6   6 low   5.56 5.67
```

The first call `t.test(example2$Tunnel, example2$Fast, paired = TRUE)` compares the values of the tunnel-chilled and fast-chilled pieces of pork using a paired t-test. This is not the analysis we are after.

The second call `with(example2, wilcox.test(Tunnel, Fast, paired = TRUE))` does the same analysis with a Wilcoxon signed rank test. Again, this is not the analysis we are after.

The third call `with(example2, wilcox.test(Tunnel ~ pH))` does a two-sample Wilcoxon (sometimes called a Mann-Whitney test) that compares the tenderness of tunnel-chilled meat for the two pH groups. We run the test

```
with(example2, wilcox.test(Tunnel ~ pH))
```

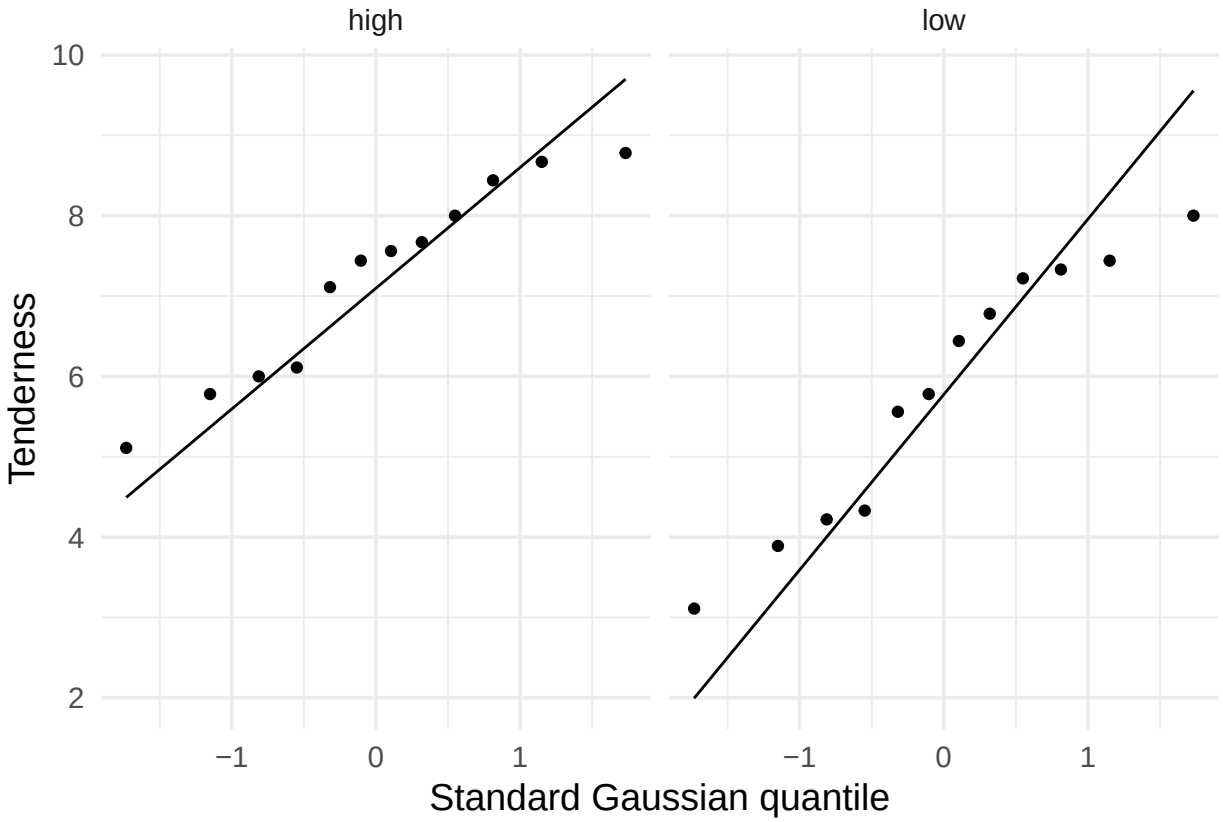
```
## Warning in wilcox.test.default(x = DATA[[1L]], y = DATA[[2L]], ...): cannot
## compute exact p-value with ties
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  Tunnel by pH
## W = 109.5, p-value = 0.03255
## alternative hypothesis: true location shift is not equal to 0
```

and conclude that there is indeed a significant difference between the groups (although the evidence is weak).

The fourth call `t.test(Tunnel ~ pH, data = example2, var.equal = TRUE)` also performs the correct test but this time via a two-sample t-test with the assumption of equal variances. To be able to apply this test, we need to check normality within each group and that the variances are equal. We construct a QQ-plot:

```
ggplot(example2, aes(sample = Tunnel)) +
  ylab("Tenderness") + xlab("Standard Gaussian quantile") +
  facet_grid(. ~ pH) +
  geom_qq() +
  geom_qq_line()
```



This looks pretty good so normality is reasonable. To check that variance, we use the `var.test` function.

```
with(example2, var.test(Tunnel ~ pH))
```

```
##
##  F test to compare two variances
##
## data:  Tunnel by pH
## F = 0.5638, num df = 11, denom df = 11, p-value = 0.3561
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
##  0.1623067 1.9584875
## sample estimates:
## ratio of variances
##          0.5638046
```

We conclude that the assumption is reasonable and can now perform the test:

```
t.test(Tunnel ~ pH, data = example2, var.equal = TRUE)
```

```
##
##  Two Sample t-test
##
## data:  Tunnel by pH
## t = 2.3625, df = 22, p-value = 0.02741
## alternative hypothesis: true difference in means between group high and group low is not equal to 0
## 95 percent confidence interval:
##  0.1686992 2.5929675
## sample estimates:
## mean in group high  mean in group low
##           7.222500           5.841667
```

The result is significant and broadly similar to the nonparametric test. However, if we had done this analysis to begin with, we would have to control for the fact that we did the equality of variances test, in which case this result would not be significant.

The fifth call `t.test(Tunnel ~ pH, data = example2[-3, ])` runs the same analysis as above except the t-test is now a Welch test (we do not assume equal variances) and we have removed the third observation. We can perform the test:

```
t.test(Tunnel ~ pH, data = example2[-3, ])
```

```
##
##  Welch Two Sample t-test
##
## data:  Tunnel by pH
## t = 2.5437, df = 18.546, p-value = 0.02006
## alternative hypothesis: true difference in means between group high and group low is not equal to 0
## 95 percent confidence interval:
##  0.2682923 2.7839804
## sample estimates:
## mean in group high  mean in group low
##           7.222500           5.696364
```

We get similar results to the above correct tests.

The sixth call `with(example2, t.test(Tunnel[pH == "low"], Tunnel[pH== "high"]))` is another way to call the same test as above (except we did not remove the third observation). We can perform the test:

```
with(example2, t.test(Tunnel[pH == "low"], Tunnel[pH== "high"]))
```

```
##
##  Welch Two Sample t-test
##
## data:  Tunnel[pH == "low"] and Tunnel[pH == "high"]
## t = -2.3625, df = 20.412, p-value = 0.02818
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -2.5984588 -0.1632079
## sample estimates:
## mean of x mean of y
##  5.841667  7.222500
```

Our conclusion is similar.

The final call `wilcox.test(Tunnel ~ pH, data = example2, paired = TRUE)` computes a paired Wilcoxon signed rank test between the two groups. The data are really independent amongst the two groups but a paired test can still be applied. We apply the test:

```
wilcox.test(Tunnel ~ pH, data = example2, paired = TRUE)
```

```
##
##  Wilcoxon signed rank exact test
##
## data:  Tunnel by pH
## V = 65, p-value = 0.04248
## alternative hypothesis: true location shift is not equal to 0
```

We see that the p-value is larger than for the corresponding call with an unpaired test. This illustrates why it is preferable to run an unpaired test to increase power.

Exercise 2.2 Two by two table: method consideration and understanding data organization

Problem

A study was conducted of 65 patients who had received or were receiving sodium aurothiomalate as a treatment for rheumatoid arthritis. The aim was to examine the possibility that toxicity to sodium aurothiomalate (SA) might be linked to impaired sulphoxidation capacity. The results were:

	Major adverse reaction (toxicity)		
Impaired sulphoxidation	Yes	No	Total
Yes	30	9	39
No	7	19	26
Total	37	28	65

The authors wrote: "The incidence of impaired sulphoxidation in patients showing SA toxicity $(30/37, 81.0\%)$ was significantly greater than in the group without adverse reaction $(9/28, 32.1\%)$ $(X^2 = 27.6, P < 0.001)$. Similarly, the incidence of toxicity was significantly increased in those with impaired sulphoxidation $(30/39, 76.9\%)$ compared to those with extensive sulphoxidation $(7/26, 26.9\%)$ $(X^2 = 36.2, P < 0.001)$.''

- Why is it impossible for both of the above chi-squared tests to be correct?
- Carry out a chi-squared test of the data in the table and compare your answer with the two results in the above paragraph.

  Remark: This may be done using either `chisq.test()` or `prop.test()`, which will give the same result. If you do not want the Yates continuity correction then add the option `correct=F`.
- Compute a 95% confidence interval for the difference between the incidences of toxicity in the group with impaired sulphoxidation and the group with extensive sulphoxidation.

  Help: Here the `prop.test()` function is helpful.
- Additional questions related to structure of the dataset:
  - What are the variables in this study?
  - How many observations have been made?

In a call to `chisq.test()` the observations are provided in a matrix, i.e.

```
chisq.test(matrix(c(30,7,9,19),2,2))
```

However, in their original laboratory diary the people who conducted this study probably had the data organized in 65 rows in a manner akin to:

Patient	Impaired sulphoxidation	Adverse reaction
1	yes	no
2	no	no
3	yes	yes
4	yes	no
⋮	⋮	⋮
64	yes	no
65	no	yes

Do you agree? Suppose data actually are given like this in a data frame called `arthitis`. Please try to decipher the following code:

```
chisq.test(table(arthritis))
```

(References: Altman, *Practical Statistics for Medical Research*, exercise 10.5, and Ayesh, R., Mitchell, S.C., Waring, R.H., et al. (1987): Sodium aurothiomalate toxicity and sulphoxidation capacity in rheumatoid arthritic patients. *Br. J. Rheumatol.*, **26**, 197–201.)

Solution

The dataset consists of 65 observations of two binary variables, namely, the impaired sulphoxidation (yes/no) and whether an adverse reaction is present (yes/no).

There is only a single chi-squared value for a given table, therefore it is not possible for both computations to be correct. We can compute the test ourselves using R (with and without continuity correction to see if we get similar results to the authors):

```
chisq.test(matrix(c(30, 7, 9, 19), 2, 2), correct = FALSE)
```

```
##
##  Pearson's Chi-squared test
##
## data:  matrix(c(30, 7, 9, 19), 2, 2)
## X-squared = 15.905, df = 1, p-value = 6.661e-05
```

```
chisq.test(matrix(c(30, 7, 9, 19), 2, 2))
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  matrix(c(30, 7, 9, 19), 2, 2)
## X-squared = 13.931, df = 1, p-value = 0.0001896
```

We see that neither test reproduces the test statistic that is claimed.

To compute a 95% confidence interval, we use the `prop.test` function:

```
prop.test(matrix(c(30, 7, 9, 19), 2, 2))
```

```
##
##  2-sample test for equality of proportions with continuity correction
##
## data:  matrix(c(30, 7, 9, 19), 2, 2)
## X-squared = 13.931, df = 1, p-value = 0.0001896
## alternative hypothesis: two.sided
## 95 percent confidence interval:
##  0.2521853 0.7478147
## sample estimates:
##    prop 1    prop 2
## 0.7692308 0.2692308
```

We get that a 95% confidence interval for the difference is given by $(0.252, 0.748)$.

If data was given in long form, the command `chisq.test(table(arthritis))` would first compute a table corresponding to the 2x2 table given in the problem and then apply the `chisq.test` function (resulting a correct application of the test).

Exercise 2.3 Analysis of a two-way table

Problem

The effect of dramanine as a remedy against seasickness was studied in an experiment on soldiers who crossed the Atlantic in a military troop transport. Among 64 people susceptible to seasickness a group of 34 soldiers were given dramanine while the remaining 30 soldiers received a placebo. The results were:

Treatment	Response seasick	not seasick	Total
dramanine	3	31	34
placebo	12	18	30
Total	15	49	64

Analyze the data to see whether dramanine has an effect on seasickness. Try both `chisq.test()`, `prop.test()` and `chisq.test.simulate()` to do the analysis, and compare the outputs. What is the same, and what is different? (Remember to specify the `conditioning`-option in `chisq.test.simulate()` from the `LabApplStat`-package.)

A confidence interval for the probability of seasickness in the placebo group may be found using the R code:

```
prop.test(12,30)
```

Find a confidence interval for the probability of seasickness in the dramanine group. Is the effect of dramanine positive or negative?

Now suppose that the data is available in the text-file `dramanine.txt`, and not in the above table! Read the dataset into R using

```
read.table("dramanine.txt",header=T)
```

(alternatively use the *Import Dataset* menu) and analyze the data.

(The data are from Chinn, H.I et al. (1950): Prophylaxis of motion sickness: evaluation of some drugs in seasickness. U.S. Air Force School of Aviation Medicine Project 21-32-014, Rep. 4.)

Solution

To analyze the data, we first enter the table into R as a matrix:

```
dramanine_table <- matrix(c(3, 12, 31, 18), 2, 2)
```

It is slightly unclear whether the data is generated with fixed row margins or simply a fixed total, however, since we are interested in determining whether the probabilities of seasickness change when given dramanine versus placebo, we are testing a null hypothesis of homogeneity. We first apply a chi-squared test using the `chisq.test` function:

```
chisq.test(dramanine_table)
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  dramanine_table
## X-squared = 6.9827, df = 1, p-value = 0.00823
```

We get a p-value of 0.008 and therefore have pretty strong evidence against the null hypothesis of homogeneity. We can apply the same test with the `prop.test` function:

```
prop.test(dramanine_table)
```

```
##
##  2-sample test for equality of proportions with continuity correction
##
## data:  dramanine_table
## X-squared = 6.9827, df = 1, p-value = 0.00823
## alternative hypothesis: two.sided
## 95 percent confidence interval:
```

```
##  -0.54268981 -0.08083961
## sample estimates:
##     prop 1     prop 2
## 0.08823529 0.40000000
```

We obtain the same p-value but we additionally get a 95% confidence interval for the difference of the probabilities which is $(-0.543, -0.081)$. This means that the effect of dramanine is most likely negative. We can obtain a confidence interval for the placebo group by again using the `prop.test`-function.

```
prop.test(dramanine_table[2, 1], dramanine_table[2, 1] + dramanine_table[2, 2])
```

```
##
##  1-sample proportions test with continuity correction
##
## data:  dramanine_table[2, 1] out of dramanine_table[2, 1] + dramanine_table[2, 2], null probability
## X-squared = 0.83333, df = 1, p-value = 0.3613
## alternative hypothesis: true p is not equal to 0.5
## 95 percent confidence interval:
##  0.2322334 0.5924978
## sample estimates:
##   p
## 0.4
```

We get a 95% confidence interval of $(0.232, 0.594)$ in the placebo group. We apply the function again:

```
prop.test(dramanine_table[1, 1], dramanine_table[1, 1] + dramanine_table[1, 2])
```

```
##
##  1-sample proportions test with continuity correction
##
## data:  dramanine_table[1, 1] out of dramanine_table[1, 1] + dramanine_table[1, 2], null probability
## X-squared = 21.441, df = 1, p-value = 3.649e-06
## alternative hypothesis: true p is not equal to 0.5
## 95 percent confidence interval:
##  0.02306877 0.24812437
## sample estimates:
##          p
## 0.08823529
```

We get a 95% confidence interval of $(0.023, 0.248)$ in the dramanine group.

Finally, we can apply the `chisq.test.simulate` function from the `LabApplStat`-package (with row conditioning):

```
library(LabApplStat)
```

```
## Loading required package: emmeans
```

```
chisq.test.simulate(dramanine_table, conditioning = "row", B = 1e5)
```

```
##
##  Chi-squared test for given row marginals (based on 1e+05 replicates)
##
## data:  dramanine_table
## X-squared = 8.6327, p-value = 0.00302
## sample estimates:
## Standard error (p-value)
##             0.0001735189
```

We get a p-value of 0.003 which aligns well with the results of the chi-squared tests above.

We can analyze the data from the `dramanine.txt` file by first loading the data:

```
dramanine_df <- read.table("dramanine.txt", header=TRUE)
```

We can then convert the data into a contigency table by using the `table` function:

```
new_dramanine_table <- table(dramanine_df)
new_dramanine_table
```

```
##            seasick
## treatment   no yes
##   dramanine 31   3
##   placebo   18  12
```

We have now obtained the same table as above and can repeat the analysis.

Exercise 2.4 A case-control study

Problem

In a study, the relation between Hodgkin's disease and the presence of tonsillectomy was investigated. 85 Hodgkin's patients had a sibling of the same sex who was free of the disease and whose age was within 5 years of the patient's. The proportion of tonsillectomies in the Hodgkin's and the control group (i.e. the siblings) was presented by the investigators in the following table:

	Tonsillectomy	No tonsillectomy	Total
Hodgkin's	41	44	85
Control	33	52	85

The following R-call says that there is no relation between Hodgkin's disease and tonsillectomy ($p = 0.2789$). However, this analysis is *wrong!* Why?

```
> prop.test(matrix(c(41, 33, 44, 52), 2, 2))

    2-sample test for equality of proportions with continuity correction

data:  matrix(c(41, 33, 44, 52), 2, 2)
X-squared = 1.1726, df = 1, p-value = 0.2789
alternative hypothesis: two.sided
95 percent confidence interval:
 -0.06603241  0.25426771
sample estimates:
   prop 1    prop 2
0.4823529 0.3882353
```

To perform the correct analysis we need more information than given in the above table, e.g. that in 37 of the patient-sibling pairs neither the patient nor the sibling had tonsillectomy. Use this additional information to perform the correct analysis.

Solution

The rows of the given table represent different variables (Hodgkin's and Control) and not different values of a single variable. Crucially, this means that we have lost the dependence between each patient and their sibling when looking at the table. An immediate give-away is that we have 85 observations but the given table makes it seem like we have twice that. All these facts combined result in an incorrect analysis. We can recover a correct table by using the additional information of 37 pairs with no tonsillectomies. The described data consists of 85 case-control pairs (boldface numbers are stated in the exercise text):

Patient	Sibling		Total
	No tonsillectomy	Tonsillectomy	
No tonsillectomy	**37**	7	**44**
Tonsillectomy	15	26	**41**
Total	**52**	**33**	**85**

As the data is paired binary data, we use the McNemar test which has a null hypothesis that the marginal proportions are the same, that is, the probability of tonsillectomy is the same for the patients as for siblings. We can perform the test using the `mcnemar.test` function:

```
mcnemar.test(matrix(c(37, 15, 7, 26), 2, 2))
```

```
##
##  McNemar's Chi-squared test with continuity correction
##
## data:  matrix(c(37, 15, 7, 26), 2, 2)
## McNemar's chi-squared = 2.2273, df = 1, p-value = 0.1356
```

We get a p-value of 0.1356 so there is no significant evidence of a difference in the proportion of tonsillectomies between the siblings and the patients.

Exercise 2.5 Power calculations

Problem

As discussed in the lectures and in the two papers by Sterne & Smith (2001), and by Gelman & Carlin (2014) there is a risk of both *Type I error* (false positives) and of *Type S* and *Type M error* (sign and magnitude). The Type S and M errors are often caused by studies where the *power* (i.e., the probability of rejecting the null hypothesis given some hypothesized effect size) is too low. The classical rule of thumb is that the power should be at least 80%.

R provides some functions (both standard functions, and functions in the package `pwr`) for doing power calculations in simple situations. The purpose of this exercise is to try some of these functions:

- Exercise 2.3 is about the effect of the drug dramanine as a remedy against seasickness. Suppose that we believe that dramanine reduces the risk of seasickness by 75%, e.g. from 50% (with placebo) to 12.5% (with dramanine). Then the required sample size needed in order to have power=80%, say, may be found using the following R code:

  ```
  > power.prop.test(power=0.8,p1=0.50,p2=0.125)
  ```

  Try this! How many soldiers are needed to have sufficient statistical power in the experiment?

  Read more about the function from the help pages `?power.prob.test`, and find the required sample size under different scenarios (power=80%, 90%, and with different proportions of seasickness in the placebo group, e.g. 30%, 40% and 50%).

- Exercise 1.5 is about the effect of two drugs (E and N) on the treatment of high blood pressure. The study was done as a cross-over in order to reduce the biological variation between patients. We hypothesize that the difference between the effects of drug E and drug N is 8 mmHg, and that the biological variation within patients (as quantified by `sd(E_diff_N)`) has standard deviation given by 15 mmHg. What is the power with sample size $n = 19$?

  Hint: Use the function `power.t.test()` with options `delta=8`, `sd=15` and `type="one.sample"`. See `?power.t.test` for further details.

  Suppose that the investigators had expected the biological variation within patients to be smaller, e.g. `sd(E_diff_N) = 10`. Would this increase or decrease the power of the study?

Solution

- We run the given line of code:

```
power.prop.test(power = 0.8, p1 = 0.5, p2 = 0.125)
```

```
##
##      Two-sample comparison of proportions power calculation
##
##               n = 22.76693
##              p1 = 0.5
##              p2 = 0.125
##       sig.level = 0.05
##           power = 0.8
##     alternative = two.sided
##
## NOTE: n is number in *each* group
```

  and conclude that we need 23 soldiers in each group to obtain 80% power. We can repeat this for different values of power and proportions of seasickness in the placebo group using the code below. The `expand.grid` function returns a data frame with all combinations of the given vectors while the `apply` function lets us apply a particular function to each row of the dataframe. We create a table using the `xtabs` function which gives the sample size in each group for the different combinations:

```
powers <- c(0.8, 0.9)
props <- c(0.5, 0.4, 0.3)
df <- expand.grid(power = powers, prop = props)
df$n <- apply(df, 1, function(row) {
  ceiling(power.prop.test(power = row["power"], p1 = row["prop"], p2 = 0.125)$n)
})
xtabs(n ~ power + prop, data = df)
```

```
##      prop
## power 0.3 0.4 0.5
##   0.8  85  39  23
##   0.9 113  52  30
```

- We use the `power.t.test` function with the given commands:

```
power.t.test(delta = 8, sd = 15, type = "one.sample", n = 19)
```

```
##
##      One-sample t test power calculation
##
##               n = 19
##           delta = 8
##              sd = 15
##       sig.level = 0.05
##           power = 0.5947301
##     alternative = two.sided
```

  and conclude that the power is 59.5%. If the variation is smaller, this should mean a greater separation between the groups and therefore it should be easier to distinguish differences. We would therefore expect an increase in power and we can confirm this:

```
power.t.test(delta = 8, sd = 10, type = "one.sample", n = 19)
```

```
##
##      One-sample t test power calculation
##
```

```
##              n = 19
##          delta = 8
##             sd = 10
##      sig.level = 0.05
##          power = 0.909207
##    alternative = two.sided
```

Exercise 2.6 Beauty and Sex ratios

Problem

In this exercise we redo the power calculation presented in the first data example in *Gelman & Carlin (2014)*. Please read the first column on page 5 in the paper for an introduction to the data example.

To perform the retrospective power analysis suggested by Gelman & Carlin two things are needed, namely *a hypothesized effect size* and *a standard error*[1]:

- Based on biological knowledge a hypothesized effect size, i.e. change of sex ratio, of 0.001, 0.003 or 0.01 is suggested.
- From the information from the paper *Kanazawa (2007)* Gelman and Carlin conclude that the standard error on the change is sex ratio is 0.033.[2]

The R function `retrodesign()`, which is available by running the R script `retrodesign.R` from the zip-file `day2.zip`, can be used to do the retrospective power analysis. This function takes two main arguments:

- `A`=hypothesized effect size.[3]
- `SE`=standard error.

Use this function to redo the analysis discussed on the second column of page 5 in the paper. Do you agree with the remarks made by Gelman and Carlin?

In the penultimate paragraph of the data example on beauty and sex ratio a traditional power calculation is made. Let us also try this: What is the sample size needed to have power = 80% when comparing the proportion of girls between *attractive* and *unattractive* parents?

- Answer this question using `power.prop.test()` assuming that the proportion of girls with unattractive parents is 0.49, and with the 3 different hypothesized effect sizes given above.

Solution

We first run the `retrodesign.R` script to obtain the `retrodesign` function:

```
source("retrodesign.R")
```

We can now re-do the analysis using the `retrodesign` function:

```
retrodesign(c(0.001, 0.003, 0.01), 0.033)
```

```
##   effect    SE      power    type_S exaggeratio
## 1  0.001 0.033 0.05010520 0.4646377   77.269503
## 2  0.003 0.033 0.05094724 0.3953041   25.909579
## 3  0.010 0.033 0.06058446 0.1950669    7.796669
```

[1] Not to be confused with the standard deviation given as input to the power calculations done for the hypertension example considered in Exercise 2.5.

[2] If you perform such a retrospective power analysis on your own data, then you will often be able to directly read off the standard error of the relevant parameter estimate from the R output.

[3] The function given on page 9 of the paper has been recoded in `retrodesign()` such that it is possible to give a vector of possible effect sizes.

We conclude that even with the largest possible (and somewhat unrealistic) effect size, the probability of Type S and M errors vastly exceed what would be required for a certain scientific conclusion.

The conventional power analysis for the smallest effect size is:

```
power.prop.test(power = 0.80, p1 = 0.001 + 0.49, p2 = 0.49)
```

```
##
##      Two-sample comparison of proportions power calculation
##
##               n = 3923022
##              p1 = 0.491
##              p2 = 0.49
##       sig.level = 0.05
##           power = 0.8
##     alternative = two.sided
##
## NOTE: n is number in *each* group
```

and we see that we require 3923022 observations in each group. For the middle effect size, we get

```
power.prop.test(power = 0.80, p1 = 0.003 + 0.49, p2 = 0.49)
```

```
##
##      Two-sample comparison of proportions power calculation
##
##               n = 435921.7
##              p1 = 0.493
##              p2 = 0.49
##       sig.level = 0.05
##           power = 0.8
##     alternative = two.sided
##
## NOTE: n is number in *each* group
```

and thus require a sample size of 435922 in each group. Finally, for the largest effect size, we get

```
power.prop.test(power = 0.80, p1 = 0.01 + 0.49, p2 = 0.49)
```

```
##
##      Two-sample comparison of proportions power calculation
##
##               n = 39239.3
##              p1 = 0.5
##              p2 = 0.49
##       sig.level = 0.05
##           power = 0.8
##     alternative = two.sided
##
## NOTE: n is number in *each* group
```

and thus require 39240 observations in each group.

Exercise 2.7 Discretization of continuous variables

Problem

As you know, the analysis of continuous data often assumes underlying normal distributions. The purpose of this exercise is to convey the idea to discretizing continuous variables into categories, which may then be analyzed using the methods presented on this course day.

To illustrate this idea, we use a classical dataset, which contains joint observations of parents heights (a weighted and corrected average of fathers and mothers heights) and their sons heights (measured in inches). This dataset has historical relevance since it made Galton invent *regression analysis*[4]. The dataset is available in the text-file `Galton.txt` and can be read into R using

```
Galton <- read.table("Galton.txt", header = TRUE, dec = ",", sep = "\t")
```

Construct a scatter plot of the data using the R code:

```
plot(child.ht ~ parent.ht, data = Galton, main="Galton's classical dataset")
```

Investigate whether the height measurements are normally distributed.

Hint: You may e.g. use the `qqnorm()` function.

After doing this, you should conclude that the height measurements are normally distributed, so probably there is no need for a discretization here. We will do it anyway to exemplify the idea of categorizing continuous data. Let us say that a man is *small* if he is less than 68 inches tall, and *tall* otherwise. The following R code makes the cross tabulation of small/tall vs. father/son:

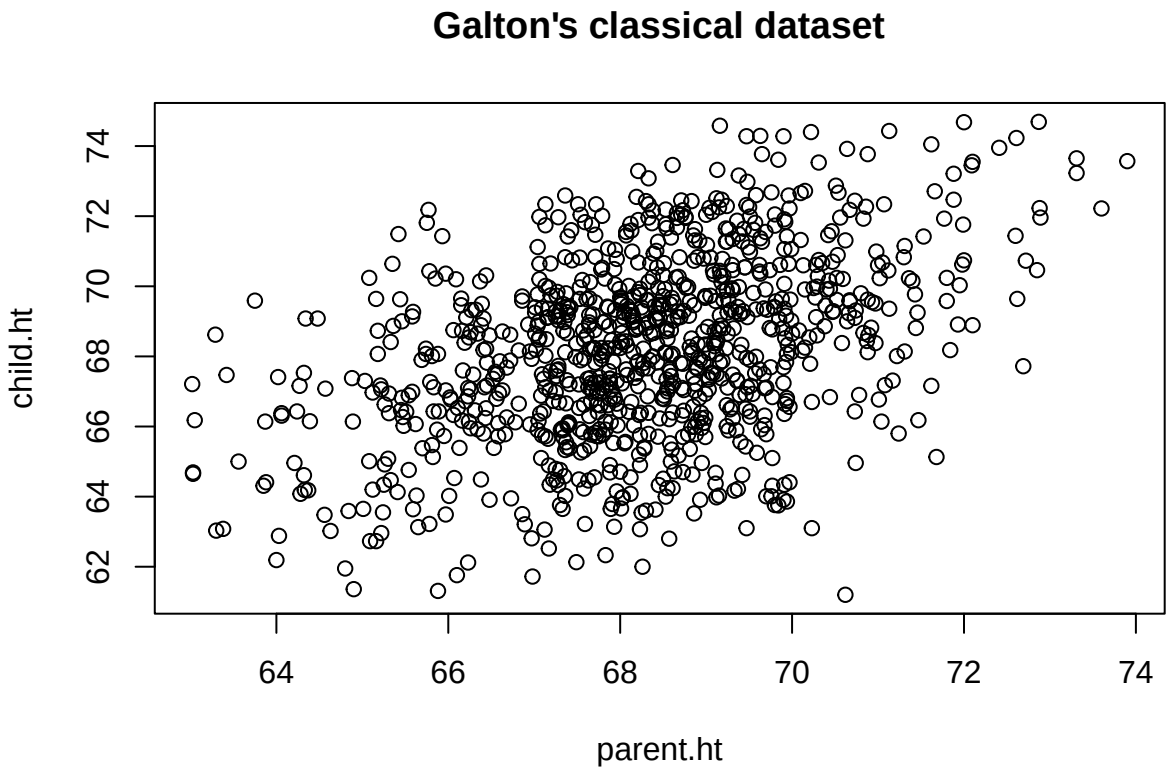
```
with(Galton, table(parent.tall = (parent.ht > 68), child.tall = (child.ht > 68)))
```

Analyze the resulting 2-by-2 table to see if there is a relationship between the father's and their son's heights.

Solution

We read the data and do as we are told:

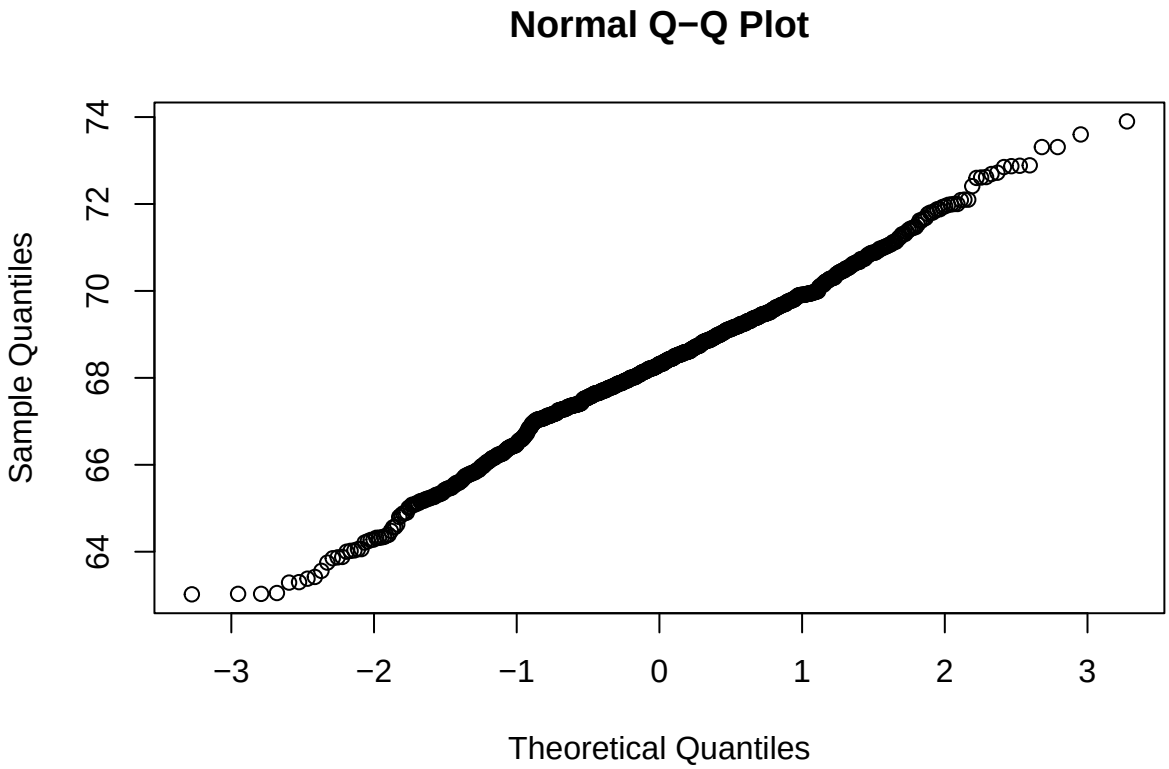
```
Galton <- read.table("Galton.txt", header = TRUE, dec = ",", sep = "\t")
plot(child.ht ~ parent.ht, data = Galton, main="Galton's classical dataset")
```



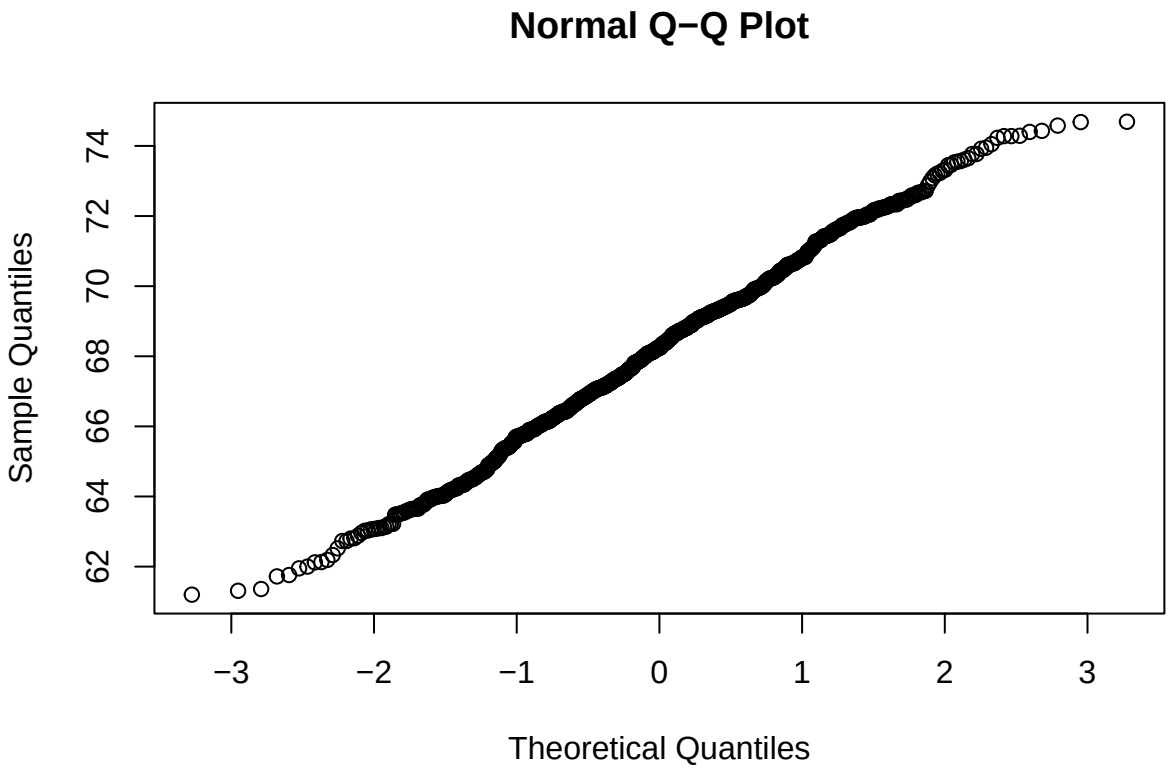
To investigate the normality of the data, we use the `qqnorm` function on each of the height distributions:

[4]We shall see why it is called regression analysis on course Day 4.

```
qqnorm(Galton$parent.ht)
```




```
qqnorm(Galton$child.ht)
```



Both variables look normally distributed. We construct the discretized data and save it in a table:

```
Galton_table <- with(Galton, table(parent.tall = (parent.ht > 68),
```

```
                                    child.tall = (child.ht > 68)))
Galton_table
```

```
##            child.tall
## parent.tall FALSE TRUE
##       FALSE   248  156
##       TRUE    189  359
```

We test whether the heights are independent using a chi-squared test:

```
chisq.test(Galton_table)
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  Galton_table
## X-squared = 66.673, df = 1, p-value = 3.205e-16
```

we get a p-value of the order 10^{-16} so we conclude that there is a very strong dependence between the heights of parents and children. We can get a confidence interval for the difference in proportions using the `prop.test` function:

```
prop.test(Galton_table)
```

```
##
##  2-sample test for equality of proportions with continuity correction
##
## data:  Galton_table
## X-squared = 66.673, df = 1, p-value = 3.205e-16
## alternative hypothesis: two.sided
## 95 percent confidence interval:
##  0.2048716 0.3330701
## sample estimates:
##    prop 1    prop 2
## 0.6138614 0.3448905
```

We get a 95% confidence interval of the difference in proportions of tall children amongst short and tall parents of $(0.205, 0.333)$. This indicates that the proportion is higher amongst tall parents (which is unsurprising when comparing to the scatter plot).