STATISTICS FOR CLINICAL TRIALS

Applications using R

@even4void

ac1950b

Statisticians are applied philosophers. Philosophers argue how many angels can dance on the head of a needle; statisticians count them. Or rather, count how many can probably dance. (...) We can predict nothing with certainty but we can predict how uncertain our predictions will be, on average that is. Statistics is the science that tells us how. – Stephen Senn, *Dicing with Death*

Foreword

Statistical analysis were done using R version 4.1.2 (2021-11-01), which is freely available from CRAN. This document started with earlier versions of R but there will not be any regression testing. You may find convenient to run `R` through RStudio. Indeed, RStudio offers really great support for editing and running `R` scripts. You can even organize your work into a project, with version control and automatic reporting built on the fly. I personally choose to work with a simple text editor and an interactive shell available within few key presses. This is possible thanks to Emacs and the brilliant ESS mode.

I no longer hold a personal licence for SAS (although I could probably get an academic one) but you can try to to replicate SAS results using SAS University Edition. It can run locally on your computer using Virtual Box). In addition, I provide Stata code to replicate most if not all analyses described in this document. The code has been tested with Stata 13 but should work on any version > 10.

Regarding R code, it is plain old R (if this makes any sense), that is we do not rely on the "tidyverse" ecosystem of packages to perform data manipulation. Instead, we will be extensively relying on the`Hmisc`and `rms` package for data aggregation, tabular outputs, and statistical modeling, as well as well established package, like `coin`, for permutation tests of statistical hypotheses. SAS being what it is, we will try to mimic the data inputting facilities whenever possible. A custom theme is used in ggplot graphics, thanks to the `hrbrthemes` package.

1 Analysis of Clinical Trials

The following analyses are based on Dmitrienko et al. [2005], with data available online at Analysis of Clinical Trials Using SAS: A Practical Guide. Note that the 2nd edition of this textbook has been published in 2017. In this section, we shall focus on the analysis of continuous and discrete endpoints.

1.1 The HAMD17 study

1.1.1 Context

This is a multi-center clinical trial comparing experimental drug vs. placebo in patients with major depression disorder. The outcome is the change from baseline after 9 weeks of acute treatment, and efficacy is measured using the total score of the Hamilton depression rating scale (17 items), also known as the HDRS score (or HAMD17 as it is called in this study) [Hamilton, 1960].

This is a classical application of unbalanced design and potential heterogeneity between clinical centres, where there is an unequal number of observations per treatment (here, drug by center).

Here is one of many ways to get data right into R:

```
raw <- textConnection("
100 P 18 100 P 14 100 D 23 100 D 18 100 P 10 100 P 17 100 D 18 100 D 22
100 P 13 100 P 12 100 D 28 100 D 21 100 P 11 100 P  6 100 D 11 100 D 25
100 P  7 100 P 10 100 D 29 100 P 12 100 P 12 100 P 10 100 D 18 100 D 14
101 P 18 101 P 15 101 D 12 101 D 17 101 P 17 101 P 13 101 D 14 101 D  7
101 P 18 101 P 19 101 D 11 101 D  9 101 P 12 101 D 11 102 P 18 102 P 15
102 P 12 102 P 18 102 D 20 102 D 18 102 P 14 102 P 12 102 D 23 102 D 19
102 P 11 102 P 10 102 D 22 102 D 22 102 P 19 102 P 13 102 D 18 102 D 24
102 P 13 102 P  6 102 D 18 102 D 26 102 P 11 102 P 16 102 D 16 102 D 17
102 D  7 102 D 19 102 D 23 102 D 12 103 P 16 103 P 11 103 D 11 103 D 25
103 P  8 103 P 15 103 D 28 103 D 22 103 P 16 103 P 17 103 D 23 103 D 18
103 P 11 103 P -2 103 D 15 103 D 28 103 P 19 103 P 21 103 D 17 104 D 13
104 P 12 104 P  6 104 D 19 104 D 23 104 P 11 104 P 20 104 D 21 104 D 25
104 P  9 104 P  4 104 D 25 104 D 19
")
d <- scan(raw, what = "character")
rm(raw)
d <- as.data.frame(matrix(d, ncol = 3, byrow = TRUE))
names(d) <- c("center", "drug", "change")
d$change <- as.numeric(as.character(d$change))
d$drug <- relevel(d$drug, ref = "P")
```

```
## Error in relevel.default(d$drug, ref = "P"):  'relevel' only for (unordered) factors
```

Briefly, the idea is to copy and paste the SAS `DATALINES` instructions provided in the textbook as raw text and to `scan` the flow of characters. This works quite well when there is not too much data. Otherwise, it is better to store the data in text file and read it in R directly. The next bit of code uses `matrix` to arrange the data into a tabular dataset with 3 columns corresponding to center, drug and change score. When transforming this table to a data frame, `center` and `drug` will be converted to factors but we need to handle the proper conversion of `change` to numerical values. Also, note that we set the reference category to the Placebo group to simplify things a bit.

1.1.2 Exploratory data analysis

Some basic exploratory graphical analysis follows. In the next chunk, we display the raw data for each centre and highlight the difference between drug and placebo using a trend line (Figure 1). Note the use of `aes(group = 1)` when calling `geom_smooth` as there is no real grouping variable in the data structure other than the ones that are already used (`drug` on the x-axis and `center` for facetting).

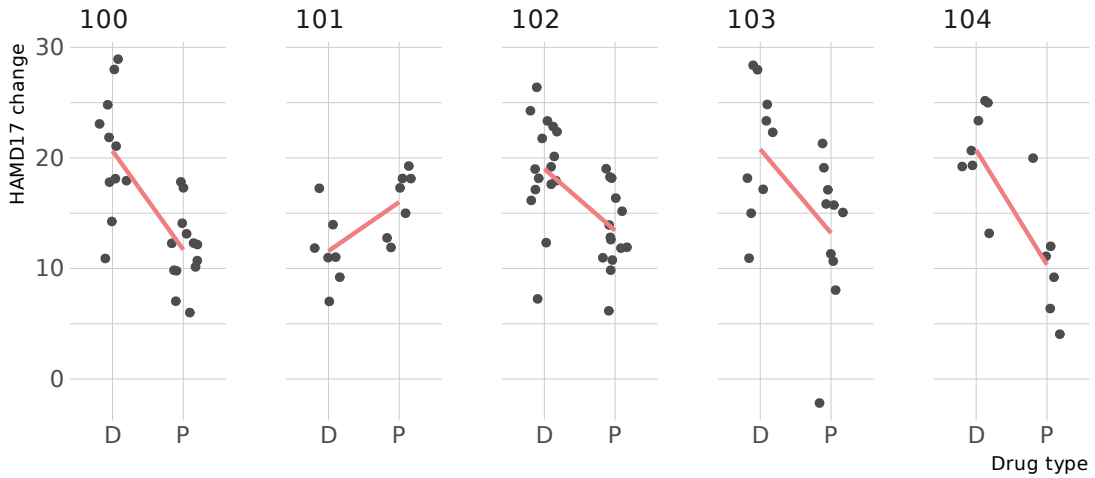

Figure 1: Distribution of change scores in each centre

```
p <- ggplot(data = d, aes(x = drug, y = change)) +
     geom_jitter(width = .2, color = grey(.3)) +
     geom_smooth(aes(group = 1), method = "lm", se = FALSE, colour = "lightcoral") +
     facet_grid(~ center) +
     labs(x = "Drug type", y = "HAMD17 change")
p
```

Using the `Hmisc` package, we can easily build a Table of summary statistics by drug and center. For simplicity, the display is limited to the first 3 centers in Table 1. Note that some of the formatting options are handled directly in the `print` or `latex` function, which is masked in the following code chunk.

```
fm <- change ~ drug + center
s <- summary(fm, data = subset(d, center %in% c("100","101","102")),
             method = "cross", fun = smean.sd)
```

Table 1: Mean HAMD17 change by drug, center

drug	100			101			102			Total		
	N	Mean	SD	N	Mean	SD	N	Mean	SD	N	Mean	SD
D	11	21	5.6	7	12	3.3	16	19	4.7	34	18	5.7
P	13	12	3.4	7	16	2.7	14	13	3.6	34	13	3.6
Total	24	16	6.3	14	14	3.7	30	16	5.0	68	16	5.3

Only 3 out of 5 centres are shown.

Let us consider the average change scores by center, which are displayed in Figure 2. First, we need to compute the average score in each group, and then compute the difference between the two (`delta`). This could be done with the `plyr` package and its `ddply` command, but we will rely on the Hmisc `summarize` command. What is important is that the results are returned as a data frame to facilitate the use of `ggplot` data structure in turn.

```
m <- with(d, Hmisc::summarize(change, llist(drug, center), mean))
r <- aggregate(change ~ center, m, diff)
p <- ggplot(data = r, aes(x = center, y = change)) +
```

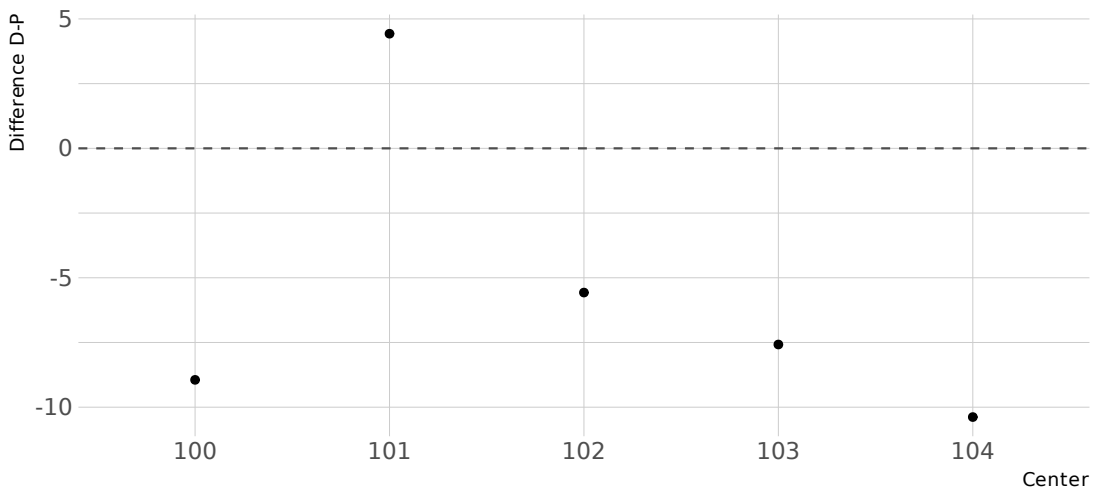

Figure 2: Average difference between drug and placebo in each centre

```
    geom_point() +
    geom_hline(yintercept = 0, linetype = 2, colour = grey(.3)) +
    labs(x = "Center", y = "Difference D-P")
p
```

1.1.3 Statistical model

Now comes the modeling stage. First, we will analyse the primary endpoint using fixed-effect models. Dmitrienko et al. [2005] provide all the maths that are necessary to understand how to derive various types of sum of squares, and this is further addressed in, e.g., [Christensen, 2002]. Let us first update the formula we used for producing Table 1 to allow for an interaction term between `drug` and `center`, which is denoted as `drug:center` in R. Of note, in R, `drug * center` will expand to `drug + center + drug:center`:

```
fm <- change ~ drug * center
replications(change ~ drug:center, data = d)
```

```
## $`drug:center`
##     center
## drug 100 101 102 103 104
##    D  11   7  16   9   7
##    P  13   7  14  10   6
```

As can be seen, data are slightly imbalanced for all but centre 101.

By default, R computes so-called "sequential" Type I sum of squares (SS), and here is what we get when using a standard combination of `lm` (to compute parameter estimates) and `anova` (to build the ANOVA table for the regression model):

```
options(contrasts = c("contr.sum", "contr.poly"))
m <- lm(fm, data = d)
anova(m)
```

```
## Analysis of Variance Table
##
## Response: change
##             Df Sum Sq Mean Sq F value   Pr(>F)
## drug         1  888.0   888.0  40.075 9.36e-09
```

```
## center       4   87.1    21.8   0.983 0.420928
## drug:center  4  507.4   126.9   5.725 0.000376
## Residuals   90 1994.4    22.2
```

A more pleasant ANOVA table can be obtained using the `rms` package, e.g.:

```
anova(ols(fm, d))
```

The `car` package allows to work with both Type II and Type III SS. Type III SSs, also called partial or Yates' weighted squares of means are the default in Stata, SPSS or SAS. Stata does not even offer Type II SS. So, if we are interested in computing Type II sum of squares in `R` using `car`, we could call `Anova` like this:

```
car::Anova(m, type = "II")
```

```
## Anova Table (Type II tests)
##
## Response: change
##             Sum Sq Df F value   Pr(>F)
## drug         889.8  1  40.153 9.11e-09
## center        87.1  4   0.983 0.420928
## drug:center  507.4  4   5.725 0.000376
## Residuals   1994.4 90
```

Type III analysis is readily obtained by replacing `type = "II"` with `type = "III"` as shown in the next code block. It should be noted that without altering the default contrast treatment that are used by `R`, as we did in the above chunk, we would not get the correct results for the Type III analysis:

```
car::Anova(m, type ="III")
```

```
## Anova Table (Type III tests)
##
## Response: change
##             Sum Sq Df  F value   Pr(>F)
## (Intercept)  22345  1 1008.344  < 2e-16
## drug           710  1   32.032 1.78e-07
## center          91  4    1.032 0.395313
## drug:center    507  4    5.725 0.000376
## Residuals     1994 90
```

Note that in the case of Type II SS, we can also use the base command `drop1` and we will get similar results:

```
drop1(m, scope = ~ ., test = "F")
```

```
## Single term deletions
##
## Model:
## change ~ drug * center
##             Df Sum of Sq  RSS   AIC F value   Pr(>F)
## <none>                   1994 319.3
## drug         1     709.8 2704 347.7  32.032 1.78e-07
## center       4      91.5 2086 315.8   1.032 0.395313
## drug:center  4     507.4 2502 334.0   5.725 0.000376
```

To sum up, the results from the different approaches are exposed in Table 2.

Remark. Here is how we could compute the parameter estimates and the SS corresponding to the drug effect in the case of a Type III analysis. The code follows what was posted on Stack Exchange, with minor adaptation. How this works is quite simple: We first get the design matrix stored in our model `m` and then solve the normal equations $(X'X)\hat{\beta} = X'y$ in order to get $\hat{\beta} = (X'X)^{-1}X'y$.

```r
D <- model.matrix(m)                              ## design matrix
bhat <- solve(t(D) %*% D) %*% t(D) %*% d$change ## beta parameters
get.ss <- function(C) {
  require(MASS)
  teta <- C %*% bhat
  M <- C %*% ginv(t(D) %*% D) %*% t(C)
  SSH <- t(teta) %*% ginv(M) %*% teta
  return(as.numeric(SSH))
}
## SS(drug|center,drug:center)
get.ss(matrix(c(0,1,0,0,0,0,0,0,0,0), nrow = 1, ncol = 10))
```

```
## [1] 709.82
```

Table 2: Overview of fixed-effects analysis for the HAMD17 study

(a) Type I SS

	Sum Sq	Df	F value	Pr(>F)
drug	888.04	1	40.07	0.0000
center	87.14	4	0.98	0.4209
drug:center	507.45	4	5.72	0.0004
Residuals	1994.38	90		

(b) Type II SS

	Sum Sq	Df	F value	Pr(>F)
drug	889.78	1	40.15	0.0000
center	87.14	4	0.98	0.4209
drug:center	507.45	4	5.72	0.0004
Residuals	1994.38	90		

(c) Type III SS

	Sum Sq	Df	F value	Pr(>F)
drug	709.82	1	32.03	0.0000
center	91.46	4	1.03	0.3953
drug:center	507.45	4	5.72	0.0004
Residuals	1994.38	90		

Regarding specific contrast, like the average treatment difference, here is one way to compute the corresponding estimate and its standard error using the `multcomp` package:

```r
summary(glht(m, mcp(drug = "Tukey", interaction_average = TRUE)))
```

```
## Error in h(simpleError(msg, call)):  erreur d'ï¿½valuation de l'argument 'object' lors de la sï¿½lection d'une mï¿½thode pour la fonction 'summary' :  Variable(s) 'drug' of class 'character' is/are not contained as a factor in 'model'.
```

Another approach relies on fitting a random-effect model to this dataset, whereby specifically the stratum and treatment-by-stratum interaction effects are treated as random while the treatment effect is considered fixed. Two main packages are available in R to fit such models, `nlme` and `lme4`. A detailed overview of the two packages is available on Kristoffer Magnusson's website. Here are the results obtained using the former:

```r
library(nlme)
m <- lme(change ~ drug, data = d, random = ~ 1 | center/drug)
summary(m)
```

```
## Linear mixed-effects model fit by REML
##   Data: d
##       AIC     BIC   logLik
##   612.051 624.976 -301.026
##
## Random effects:
##  Formula: ~1 | center
##         (Intercept)
## StdDev: 0.000293039
##
##  Formula: ~1 | drug %in% center
##         (Intercept) Residual
## StdDev:     2.52191  4.72264
##
## Fixed effects:  change ~ drug
##                Value Std.Error DF  t-value p-value
## (Intercept) 15.79195  0.936256 90 16.86712  0.0000
## drug1        2.85357  0.936256  4  3.04785  0.0381
##  Correlation:
##       (Intr)
```

```
## drug1 -0.001
##
## Standardized Within-Group Residuals:
##         Min          Q1         Med          Q3         Max
## -3.20415708 -0.46764113  0.00909887  0.64208317  1.87287085
##
## Number of Observations: 100
## Number of Groups:
##           center drug %in% center
##                5               10
```

Contrary to the SAS formula used in the textbook, no Satterthwaite correction is applied on the degrees of freedom. It is, however, possible to use it with `lme4`. First, we need to refit the model using lme4::lmer

```
library(lme4)
library(lmerTest)
m <- lmer(change ~ drug + (1 | center/drug), data = d)
summary(m, ddf = "Satterthwaite")  # this is the default
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method ['lmerModLmerTest']
## Formula: change ~ drug + (1 | center/drug)
##    Data: d
##
## REML criterion at convergence: 602.1
##
## Scaled residuals:
##    Min     1Q Median     3Q    Max
## -3.204 -0.468  0.009  0.642  1.873
##
## Random effects:
##  Groups      Name        Variance Std.Dev.
##  drug:center (Intercept)  6.36    2.52
##  center      (Intercept)  0.00    0.00
##  Residual                22.30    4.72
## Number of obs: 100, groups:  drug:center, 10; center, 5
##
## Fixed effects:
##             Estimate Std. Error     df t value Pr(>|t|)
## (Intercept)   15.792      0.936  6.757   16.87    9e-07
## drug1          2.854      0.936  6.757    3.05    0.019
##
## Correlation of Fixed Effects:
##       (Intr)
## drug1 -0.001
## optimizer (nloptwrap) convergence code: 0 (OK)
## boundary (singular) fit: see ?isSingular
```

The authors later used the Gail-Simon test[Gail and Simon, 1985] to test for qualitative interaction between treatment and strata. The corresponding two-tailed Likelihood ratio test is implemented in the QualInt package.

```
library(QualInt)
with(d, qualint(change, drug, center, test = "LRT"))
```

```
##
## Call:
## qualint(y = change, trtment = drug, subgrp = center, test = "LRT")
##
## Type:
## continuous
##
```

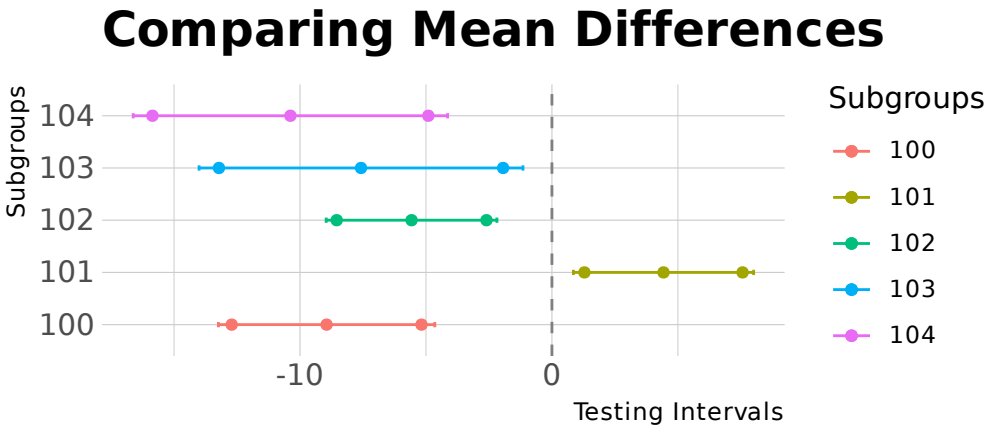

Figure 3: Average differences between drug and placebo stratified by centres

```
## Estimating Results for Mean Difference:
##     Estimate Std. Error Lower CI Upper CI
## 100    -8.94       1.92   -12.71    -5.18
## 101     4.43       1.60     1.29     7.57
## 102    -5.57       1.52    -8.54    -2.60
## 103    -7.58       2.88   -13.22    -1.94
## 104   -10.38       2.79   -15.86    -4.91
##
## Test:
## LRT
##
## p-value:
## 0.0297
##
## Power:
## 0.581
##
## Alpha:
## 0.05
```

This R package even provides a graphical method when specifying options `test = "IBGA"` and `plotout = TRUE` (Figure 3). The IBGA method relies on simultaneous 95% confidence intervals as described in Pan and Wolfe [1997]. The `multcomp` package also allows for simultaneous CIs albeit they are often used for testing multiple contrasts in ANOVA-like settings.

1.2 The Urinary incontinence trial

Context. This is a subset of data collected in an RCT on urinary incontinence where the primary endpoint was the percent change from baseline of number of incontinence episodes per week over an 8-week period. Patients were initially randomized into one of three strata depending on the baseline frequency of incontinence episodes.

This is an example of the use of stratified non-parametric analysis.

This time, we managed to get data in the right format using this little R script: `urininc.R`. Assuming it is located in an `R/` folder in the current working directory, we can `source` it into R and we will get a data frame named `d`.

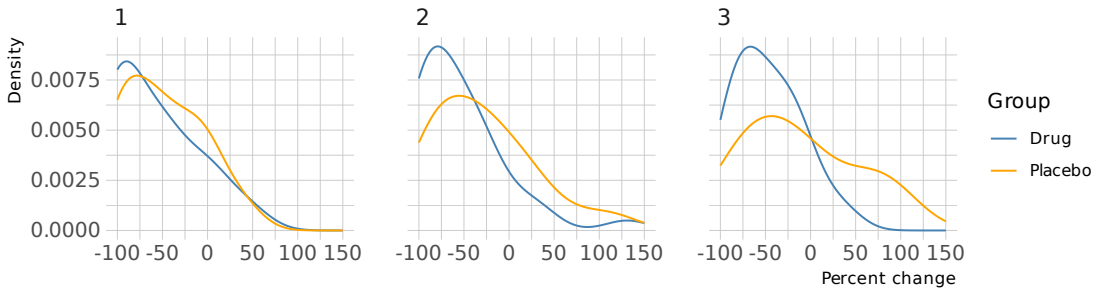

Figure 4: Density estimates for the percent change in frequency of incontinence episodes

```
source("R/urininc.R")
```

```
## Error in relevel.default(d$group, ref = "Placebo"):  'relevel' only for (unordered) factors
```

```
str(d)
```

```
## 'data.frame': 200 obs. of  3 variables:
##  $ group : chr  "Placebo" "Placebo" "Placebo" "Placebo" ...
##  $ strata: int  1 1 1 1 1 1 1 1 1 1 ...
##  $ change: num  -86 -38 43 -100 289 0 -78 38 -80 -25 ...
```

To summarize the data, we can again make use of `Hmisc summary` for "crossed" data.

```
s <- summary(change ~ group + strata, data = d, method = "cross", overall = FALSE)
```

Table 3: Mean change in number of incontinence episods by drug, strata

group	**1**			**2**			**3**		
	N	Missing		N	Missing		N	Missing	
Drug	39	1	−24.2	33	7	−53.8	19	1	−47.7
Placebo	40	0	−29.0	32	8	−28.7	20	0	−11.7

As can be seen, there is a higher number of missing values in strata 2 (around 20% in both groups) and larger variations on average between the two groups in the third strata. Next, we displayed the distribution of the percent change in frequency of incontinence episodes as density curves in Figure 4. Instead of relying on `geom_density`, we use the rather generic `geom_lines` with an extra `stat=` parameter.

```
p <- ggplot(data = d, aes(x = change, color = group)) +
     geom_line(stat = "density", adjust = 1.2) +
     facet_wrap(~ strata, ncol = 3) +
     scale_color_manual("Group", values = c("steelblue", "orange")) +
     scale_x_continuous(limits = c(-100, 150)) +
     labs(x = "Percent change", y = "Density")
p
```

The authors used the van Elteren test [van Elteren, 1960], which can be regarded as an extension of the Wilcoxon rank sum test for stratified data where larger weights are assigned to rank sums from smaller strata. An alternative is the "aligned rank test" proposed by Hodges and Lehman [1962] as discussed by Mehrotra et al. [2010]. In R, there is an old version that is mentionned on the R listserve (August 2005), but for now we will use the `coin` package as shown below:

```r
library(coin)
dc <- subset(d, complete.cases(d))
independence_test(change ~ group | strata, data = dc,
                  ytrafo = function(data) trafo(data, numeric_trafo = rank,
                                                block = dc$strata),
                  teststat = "quad")
```

```
## Error in .local(.Object, ...):  'block' is not a factor
```

Although we get different results as those reported by the authors, we would reach the same conclusion, namely that there is an effect of the treatment on the outcome after adjusting for the centre effect. We will get, however, closer results ($p = 0.02369$ for the row mean squares test statistic) if we simply remove the `scores=` option when calling SAS `PROC FREQ` [Stokes et al., 2012]:

```
PROC FREQ;
  TABLES strata*group*change / noprint cmh2;
RUN;
```

In comparison, as noted by the authors, a Type III ANOVA would yield non-significant result about the effect of drug on change scores:

```r
m <- lm(change ~ group + strata, data = d)
car::Anova(m, type = "III")
```

```
## Anova Table (Type III tests)
##
## Response: change
##              Sum Sq  Df F value Pr(>F)
## (Intercept)   21072   1   3.070 0.0814
## group          9873   1   1.438 0.2320
## strata         1048   1   0.153 0.6965
## Residuals   1235405 180
```

1.3 The Severe sepsis trial

Context. This is a placebo-controlled RCT examining the effect of an experimental drug on 28-day all-cause mortality in patients with severe sepsis. Patients were allocated to one of four strata depending on their APACHE II score [Knaus et al., 1985].

This is a classical application of stratified analysis of a binary outcome (dead/alive).

To enter the data in R, we will input individual values of the three-way Table of events as an array. Note that it would also be possible to create two matrix objects and then bind into to a 3-dimensional table. In what follows, we write data for the treated group first. Note that when using `array`, data should be entered column-wise (there is no `byrow =` option as in `matrix`).

```r
varnames <- list(strata = 1:4,
                 status = c("Dead", "Alive", "Total"),
                 group = c("Experimental", "Placebo"))
d <- array(c(33,49,48,80,185,169,156,130,218,218,204,210,
             26,57,58,118,189,165,104,123,215,222,162,241),
           dim = c(4,3,2), dimnames = varnames)
```

Note also that the third column ("Total") can be safely omitted as margins can be computed automatically with R, e.g.:

```r
addmargins(d[,-3,], c(1,2))
```

```r
d <- d[,-3,]
dim(d)
```

```
## [1] 4 2 2
```

An alternative representation of this array-based Table is provided by R's flat tables (`ftable`), in

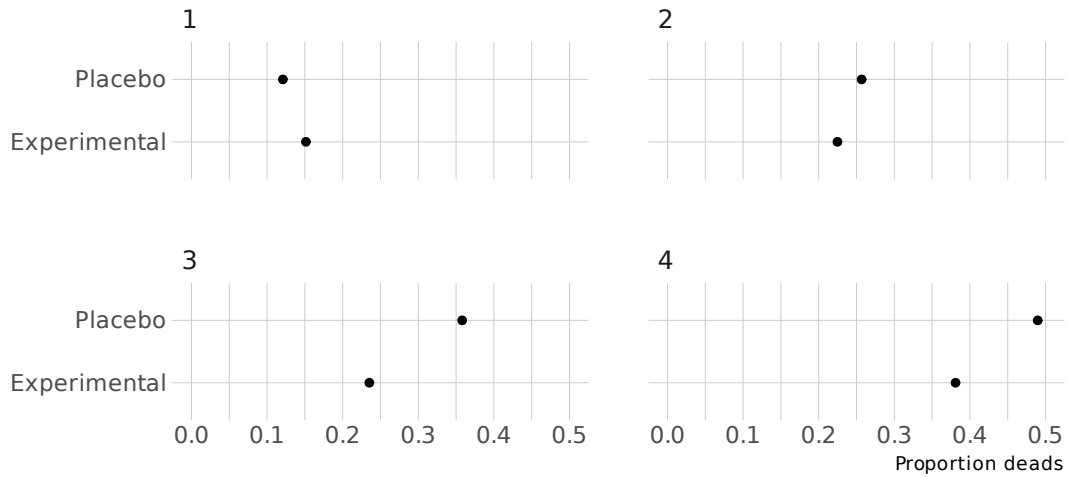

Figure 5: Proportion of patients who died by the end of the study

long or wide format; see Table 4 for the wide format using `ftable(d, row.vars = 1, col.vars = c(3,2))`:

```
ftable(d)

##               group Experimental Placebo
## strata status
## 1      Dead                  33      26
##        Alive                185     189
## 2      Dead                  49      57
##        Alive                169     165
## 3      Dead                  48      58
##        Alive                156     104
## 4      Dead                  80     118
##        Alive                130     123
```

	group:	Experimental		Placebo	
strata	status:	Dead	Alive	Dead	Alive
1		33	185	26	189
2		49	169	57	165
3		48	156	58	104
4		80	130	118	123

Table 4: 28-day mortality data from the 1690-patient sepsis study

The following code is used to depict the situation in graphical terms:

```
dd <- as.data.frame(ftable(d))
r <- ddply(dd, c("strata", "group"), mutate, prop = Freq/sum(Freq))
p <- ggplot(subset(r, status == "Dead"), aes(x = prop, y = group)) +
     geom_point() + facet_wrap(~ strata, nrow = 2) +
     scale_x_continuous(limits = c(0,0.5)) +
     labs(x = "Proportion deads", y = "")
p
```

Based on a logistic regression model, the authors presented a summary of a Type III analysis of effects. Here is what can be done in R. First, we will slightly re arrange the data table so that

we have a working data frame with total counts for successes (here, dead patients) and failure (patients still alive) in separate columns, together with columns describing strata and treatment levels.

```
n <- rbind(d[,1:2,1], d[,1:2,2])
rownames(n) <- NULL
n <- as.data.frame(n)
n$strata <- gl(4, 1)
n$group <- gl(2, 4, labels = c("Experimental", "Placebo"))
n$group <- relevel(n$group, ref = "Placebo")
```

Then, since we are working with grouped or aggregated data, we will use the `cbind()` option in R's `glm` function , as shown below. Note that we also ask to use SAS treatment contrast for the `strata` factor, in order to ensure that the fourth level is used as the reference category. Type III analysis is readily available within the `car` package.

```
m <- glm(cbind(Dead,Alive) ~ group + strata, data = n, family = binomial,
         contrasts = list(strata = "contr.SAS"))
car::Anova(m, type = "III")
```

```
## Analysis of Deviance Table (Type III tests)
##
## Response: cbind(Dead, Alive)
##        LR Chisq Df Pr(>Chisq)
## group      6.99  1     0.0082
## strata   105.61  3     <2e-16
```

Finally, profile likelihood 95% confidence intervals are simply obtained using `confint()` which will call the appropriate profile method depending on the kind of model at hand:

```
exp(confint(m))
```

```
##                2.5 %   97.5 %
## (Intercept) 0.641248 0.931819
## group1      1.039214 1.296530
## strata1     0.144325 0.280712
## strata2     0.304853 0.542057
## strata3     0.397057 0.714698
```

1.4 The dose-finding hypertension trial

Context. This trial aimed to compare low, medium and high doses of a new antihypertensive drug to a placebo. The primary efficacy variable that is being considered in this study is diastolic blood pressure.

This example is used to illustrate various methods to deal with multiple testing issues. In what follows we will work with p-values (raw data are not available) estimated when comparing all four groups (P, placebo vs. L, M, and H, the low, medium and high dose groups).

	L vs. P	M vs. P	H vs. P
Scenario 1	0.047	0.0167	0.015
Scenario 2	0.047	0.027	0.015
Scenario 3	0.053	0.026	0.017

Table 5: P-values obtained from different approaches

The `p.adjust()` command can be used to compute various "adjusted" p-values, the default being the step-down method proposed by Holm [1979].

```
pvals <- c(0.047, 0.0167, 0.015)  ## scenario 1
p.adjust(pvals, method = "bonferroni")
```

```
## [1] 0.1410 0.0501 0.0450
```

The Šidák method is not available in `p.adjust()` but it is not difficult to implement a custom function to perform this correction which amounts to update the nominal α level with $1-(1-\alpha)^{1/n}$, that is:

```
f <- function(x) (1-(1-x)^length(x))
f(pvals)

## [1] 0.1344768 0.0492680 0.0443284
```

Alternatively, one can dig into the `multtest` package by Dudoit and van der Laan [2008], available on `htpp://www.bioconductor.org`) (see the `mt.rawp2adjp()` command).

Contrary to the preceding results, Holm's adjusted p-values will all be < 0.05 as illustrated below:

```
p.adjust(pvals, method = "holm")

## [1] 0.047 0.045 0.045
```

And here is a comparison of Holm and Hommel's adjusted p-values for the second scenario (Table 5):

```
pvals <- c(0.047, 0.027, 0.015)  ## scenario 2
p.adjust(pvals, method = "holm")

## [1] 0.054 0.054 0.045

p.adjust(pvals, method = "hommel")

## [1] 0.0470 0.0470 0.0405
```

Finally, Hommel's method is compared to Hochberg's approach for the third scenario:

```
pvals <- c(0.053, 0.026, 0.017)  ## scenario 3
p.adjust(pvals, method = "hochberg")

## [1] 0.053 0.052 0.051

p.adjust(pvals, method = "hommel")

## [1] 0.053 0.052 0.039
```

One can also look into the `cherry` package [Goeman and Solari, 2011] whose vignette includes a comparison of Simes vs. Hommel or Fisher approach to multiple testing, as well as examples of closed testing methods.

1.5 The allergen-induced asthma trial

Context. Data comes from a trial designed to assess the efficacy profile of a bronchodilator in allergen-induced asthma. There are 20 patients that were randomly assigned to receive either an experimental drug or a placebo [Taylor et al., 1991]. The forced expiratory volume in one second (FEV) was used to measure how the drug attenuated bronchoconstriction, and FEV curves were averaged at each time point in both groups (Table 6). The therapeutic effect was the time to the onset of action–that is, the first time point at which clinically and statistically significant separation between the FEV curves is observed.

Beside stepwise approaches relying on data-driven ordering of p-values–this is also known as closed testing–fixed-sequence testing methods are used when we are interested in prespecified sequences of hypotheses. This is illustrated in the next example.

To load the data, we will use a simple matrix to store the values displayed in Table 6 and then reshape the matrix to a so-called "tidy" data frame using `melt`, although this last step is not really necessary.

	Experimental drug			Placebo		
Time (hours)	n	Mean	SD	n	Mean	SD
0.25	10	0.58	0.29	10	0.71	0.35
0.5	10	0.62	0.31	10	0.88	0.33
0.75	10	0.51	0.33	10	0.73	0.36
1	10	0.34	0.27	10	0.68	0.29
2	10	-0.06	0.22	10	0.37	0.25
3	10	0.05	0.23	10	0.43	0.28

Table 6: Reduction in FEV measurements from baseline by time after the allergen challenge

```
tmp <- matrix(c(0.25,10,0.58,0.29,10,0.71,0.35,
                0.5,10,0.62,0.31,10,0.88,0.33,
                0.75,10,0.51,0.33,10,0.73,0.36,
                1,10,0.34,0.27,10,0.68,0.29,
                2,10,-0.06,0.22,10,0.37,0.25,
                3,10,0.05,0.23,10,0.43,0.28),
                nrow = 6, byrow = TRUE)
colnames(tmp) <- c("time", "N0", "Mean0", "SD0", "N1", "Mean1", "SD1")
d <- melt(as.data.frame(tmp), id.vars = 1, measure.vars = c(3,4,6,7))
```

Note that it is quite easy to go back to the matrix form by using `dcast` as shown below:

```
r <- ddply(dcast(d, time ~ variable), "time", mutate,
           diff = Mean1 - Mean0, se = sqrt((1/10+1/10)*(SD0^2+SD1^2)/2))
```

In what follows, we compute the lowest bound of a 95% confidence interval and display its value at consecutive time points. This amounts to looking at results from sequential testing, i.e. determine the first statistically significant difference. Equivalently, we could rely on simple Student t-test.

```
p <- ggplot(r, aes(x = time, y = diff))
p <- p + geom_line() + geom_point()
p <- p + geom_line(aes(x = time, y = diff - qt(0.95, 20-2) * se), linetype = 2)
p <- p + scale_x_continuous(breaks = seq(0, 3, by = 1))
p <- p + scale_y_continuous(breaks = seq(-0.2, 0.5, by = 0.1))
p <- p + geom_hline(aes(yintercept = 0))
p + labs(x = "Time (hours)", y = "Treatment difference (95% Lower CI)")
```

Note that to display error bars instead of the lower bound for the 95% confidence interval, we would use:

```
p <- p + geom_errorbar(aes(ymin = diff - qt(0.975, 20-2) * se,
                           ymax = diff + qt(0.975, 20-2) * se),
                       width = .1)
```

This approach, however, does not control the familywise error rate. Looking at treatment differences from the last measurement to the first ("step-down" approach) suggests that a significant difference at one hour, and not 30 minutes as in the previous case.

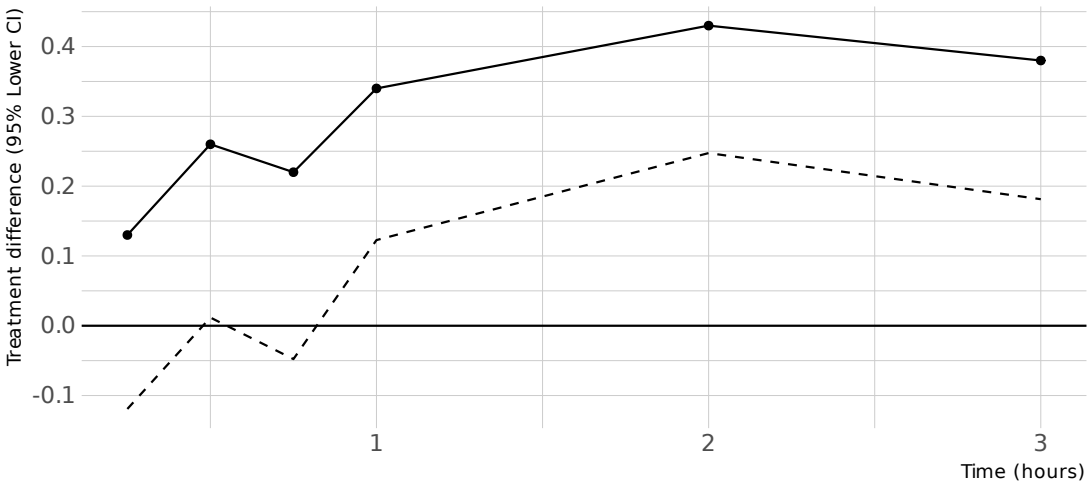

Figure 6: Treatment comparisons in the asthma study

References

R Christensen. *Plane Answers to Complex Questions: The Theory of Linear Models*. Springer, New York, 2002.

A Dmitrienko, G Molenberghs, C Chuang-Stein, and W Offen. *Analysis of Clinical Trials Using SAS: A Practical Guide*. SAS Institute Inc., Cary, NC, USA, 2005.

S Dudoit and MJ van der Laan. *Multiple Testing Procedures with Applications to Genomics*. New York: Springer, 2008.

M Gail and R Simon. Testing for qualitative interactions between treatment effects and patient subsets. *Biometrics*, 41(2):361–372, 1985.

JJ Goeman and A Solari. Multiple testing for exploratory research. *Statistical Science*, 26(4):584–597, 2011.

M. Hamilton. A rating scale for depression. *Journal of Neurology and Neurosurgery Psychiatry*, 23: 56–62, 1960.

JL Hodges and EC Lehman. Rank methods for combination of independent experiments in the analysis of variance. *Annals of Mathematical Statistics*, 33:482–497, 1962.

S. Holm. A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics*, 6:65–70, 1979.

WA Knaus, EA Draper, DP Wagner, and JE Zimmerman. APACHE II: a severity of disease classification system. *Critical Care Medicine*, 13(10):818–829, 1985.

DV Mehrotra, X Lu, and X Li. Rank-based analyses of stratified experiments: Alternatives to the van Elteren test. *The American Statistician*, 64(2):121–130, 2010.

G Pan and DA Wolfe. Test for qualitative interaction of clinical significance. *Statistics in Medicine*, 16(14):1645–1652, 1997.

ME Stokes, CS Davis, and GG Koch. *Categorical Data Analysis Using SAS*. SAS Institute Inc., Cary, NC, USA, 2012.

IK Taylor, KM O'Shaughnessy, RW Fuller, and CT Dollery. Effect of cysteinyl-leukotreine receptor antagonist ici 204.219 on allergen-induced bronchoconstriction and airway hyperreactivity in atopic subjetcs. *Lancet*, 337:690–693, 1991.

PH van Elteren. On the combination of independent two sample tests of wilcoxon. *Bulletin of the Institute of International Statistics*, 37:351–361, 1960.

Contents

List of Tables

List of Figures

R version 4.1.2 (2021-11-01)
Version : ac1950b (2022-01-24)
https://github.com/chlalanne/SAS2R