Friedman Test in R

The **Friedman test** is a non-parametric alternative to the one-way repeated measures ANOVA test. It extends the *Sign test* in the situation where there are more than two groups to compare.

Friedman test is used to assess whether there are any statistically significant differences between the distributions of three or more paired groups. It's recommended when the normality assumptions of the one-way repeated measures ANOVA test is not met or when the dependent variable is measured on an ordinal scale.

In this chapter, you'll learn how to:

- **Compute Friedman test in R**
- **Perform multiple pairwise-comparison between groups**, to identify which pairs of groups are significantly different.
- **Determine the effect size of Friedman test using the Kendall's W**.

Contents:

Related Book

Practical Statistics in R II - Comparing Groups: Numerical Variables

Prerequisites

Make sure you have installed the following R packages:

- `tidyverse` for data manipulation and visualization
- `ggpubr` for creating easily publication ready plots
- `rstatix` provides pipe-friendly R functions for easy statistical analyses.

Load the packages:

```
library(tidyverse)
library(ggpubr)
library(rstatix)
```

Data preparation

We'll use the self esteem score dataset measured over three time points. The data is available in the datarium package.

```
data("selfesteem", package = "datarium")
head(selfesteem, 3)
## # A tibble: 3 x 4
##      id    t1    t2    t3
##   <int> <dbl> <dbl> <dbl>
## 1     1  4.01  5.18  7.11
## 2     2  2.56  6.91  6.31
## 3     3  3.24  4.44  9.78
```

Gather columns `t1`, `t2` and `t3` into long format. Convert `id` and `time` variables into factor (or grouping) variables:

```
selfesteem <- selfesteem %>%
  gather(key = "time", value = "score", t1, t2, t3) %>%
  convert_as_factor(id, time)
head(selfesteem, 3)
## # A tibble: 3 x 3
##   id    time  score
##   <fct> <fct> <dbl>
## 1 1     t1     4.01
## 2 2     t1     2.56
## 3 3     t1     3.24
```

Summary statistics

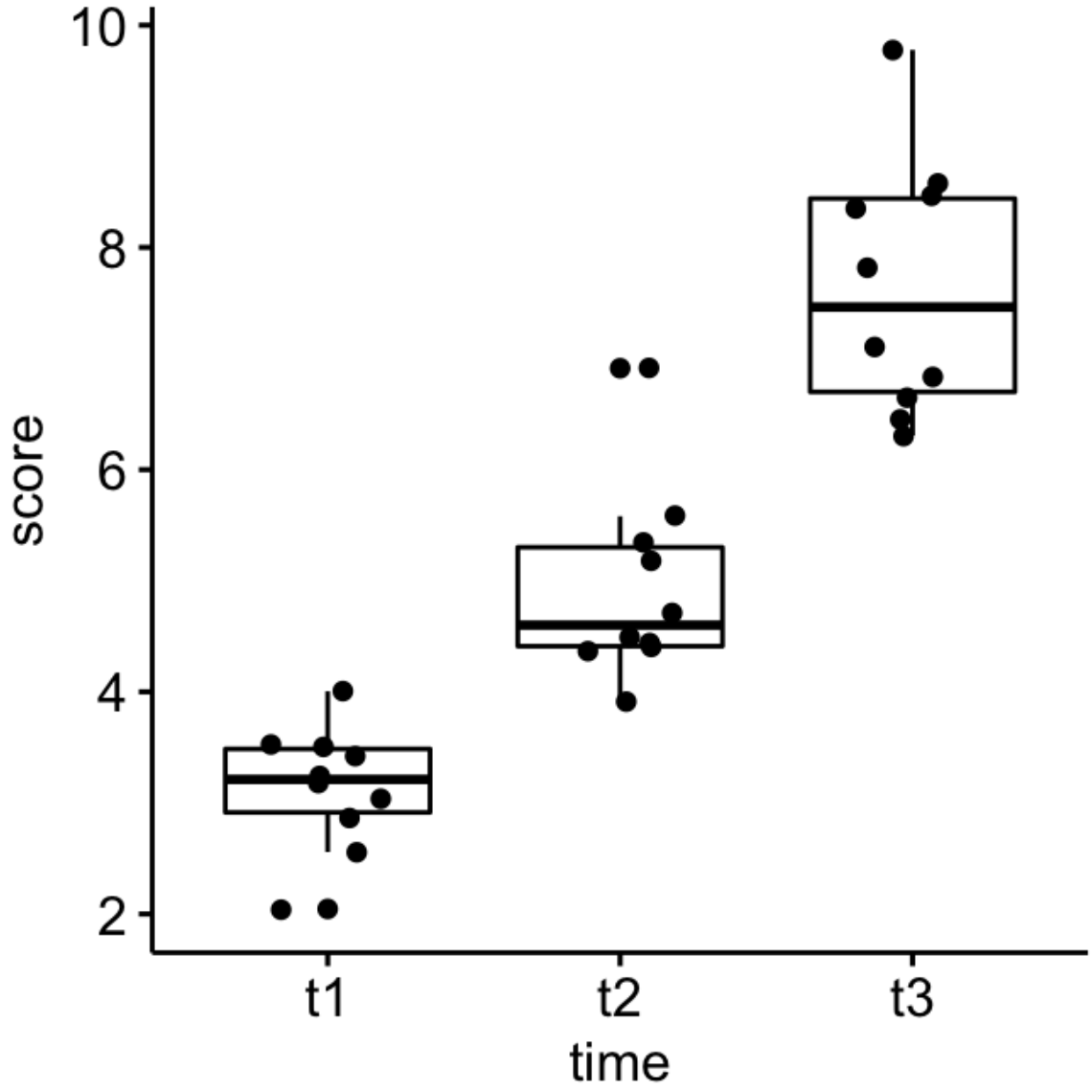
Compute some summary statistics of the self-esteem `score` by groups (`time`):

```
selfesteem %>%
  group_by(time) %>%
  get_summary_stats(score, type = "common")
## # A tibble: 3 x 11
##   time  variable     n   min   max median   iqr  mean    sd    se    ci
##   <fct> <chr>    <dbl> <dbl> <dbl>  <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 t1    score       10  2.05  4.00   3.21 0.571  3.14 0.552 0.174 0.395
## 2 t2    score       10  3.91  6.91   4.60 0.89   4.93 0.863 0.273 0.617
## 3 t3    score       10  6.31  9.78   7.46 1.74   7.64 1.14  0.361 0.817
```

Visualization

Create a box plot and add points corresponding to individual values

```
ggboxplot(selfesteem, x = "time", y = "score", add = "jitter")
```



Computation

We'll use the pipe-friendly `friedman_test()` function [rstatix package], a wrapper around the R base function `friedman.test()`.

```
res.fried <- selfesteem %>% friedman_test(score ~ time |id)
res.fried
## # A tibble: 1 x 6
##   .y.       n statistic    df        p method
## * <chr> <int>     <dbl> <dbl>    <dbl> <chr>
## 1 score    10      18.2     2 0.000112 Friedman test
```

The self esteem score was statistically significantly different at the different time points during the diet, X2(2) = 18.2, p = 0.0001.

Effect size

The Kendall's W can be used as the measure of the Friedman test effect size. It is calculated as follow : `W = X2/N(K-1)` ; where `W` is the Kendall's W value; `X2` is the Friedman test statistic value; `N` is the sample size. `k` is the number of measurements per subject (M. T. Tomczak and Tomczak 2014).

The Kendall's W coefficient assumes the value from 0 (indicating no relationship) to 1 (indicating a perfect relationship).

Kendall's W uses the Cohen's interpretation guidelines of 0.1 - < 0.3 (small effect), 0.3 - < 0.5 (moderate effect) and >= 0.5 (large effect). Confidence intervals are calculated by bootstap.

```
selfesteem %>% friedman_effsize(score ~ time |id)
## # A tibble: 1 x 5
##   .y.       n effsize method    magnitude
## * <chr> <int>   <dbl> <chr>     <ord>
## 1 score    10   0.910 Kendall W large
```

A large effect size is detected, W = 0.91.

Multiple pairwise-comparisons

From the output of the Friedman test, we know that there is a significant difference between groups, but we don't know which pairs of groups are different.

A significant Friedman test can be followed up by pairwise **Wilcoxon signed-rank tests** for identifying which groups are different.

Note that, the data must be correctly ordered by the blocking variable (`id`) so that the first observation for time `t1` will be paired with the first observation for time `t2`, and so on.

Pairwise comparisons using paired Wilcoxon signed-rank test. P-values are adjusted using the Bonferroni multiple testing correction method.

```
# pairwise comparisons
pwc <- selfesteem %>%
  wilcox_test(score ~ time, paired = TRUE, p.adjust.method = "bonferroni")
pwc
## # A tibble: 3 x 9
##   .y.   group1 group2    n1    n2 statistic     p p.adj p.adj.signif
## * <chr> <chr>  <chr>  <int> <int>     <dbl> <dbl> <dbl> <chr>
## 1 score t1     t2        10    10         0 0.002 0.006 **
## 2 score t1     t3        10    10         0 0.002 0.006 **
## 3 score t2     t3        10    10         1 0.004 0.012 *
```

All the pairwise differences are statistically significant.

Note that, it is also possible to perform pairwise comparisons using Sign Test, which may lack power in detecting differences in paired data sets. However, it is useful because it has few assumptions about the distributions of the data to compare.

Pairwise comparisons using sign test:

```
pwc2 <- selfesteem %>%
  sign_test(score ~ time, p.adjust.method = "bonferroni")
pwc2
```

Report

The self-esteem score was statistically significantly different at the different time points using Friedman test, X2(2) = 18.2, p = 0.00011.

Pairwise Wilcoxon signed rank test between groups revealed statistically significant differences in self esteem score between t1 and t2 (p = 0.006); t1 and t3 (0.006); t2 and t3 (0.012).

```
# Visualization: box plots with p-values
pwc <- pwc %>% add_xy_position(x = "time")
ggboxplot(selfesteem, x = "time", y = "score", add = "point") +
  stat_pvalue_manual(pwc, hide.ns = TRUE) +
  labs(
    subtitle = get_test_label(res.fried,  detailed = TRUE),
    caption = get_pwc_label(pwc)
  )
```