How to Perform a Friedman Test in R: A Step-by-Step Guide

Authored by

stats writer

March 11, 2026

RECOMMENDED CITATION

stats writer (2026). *How to Perform a Friedman Test in R: A Step-by-Step Guide*. PSYCHOLOGICAL SCALES. Retrieved from https://scales.arabpsychology.com/?p=135146

The Friedman Test is a non-parametric statistical test used to compare multiple related groups. It is commonly used when the assumptions of parametric tests, such as the t-test or ANOVA, are not met. In order to perform the Friedman Test in R, the user must first load the package "coin" and input the data in a matrix format with each group as a column. The function "friedman.test" is then used to conduct the test, which provides the test statistic and p-value. If the p-value is below a predetermined significance level, it can be concluded that there is a significant difference between the groups. Additional post-hoc tests may also be performed to determine which specific groups differ from each other. Overall, the Friedman Test in R allows for the comparison of multiple related groups without the need for normality assumptions.

Perform the Friedman Test in R

The Friedman Test is a non-parametric alternative to the Repeated Measures ANOVA. It is used to determine whether or not there is a statistically significant difference between the means of three or more groups in which the same subjects show up in each group.

This tutorial explains how to perform the Friedman Test in R.

Example: The Friedman Test in R

To perform the Friedman Test in R, we can use the friedman.test() function, which uses the following syntax:

friedman.test(y, groups, blocks)

where:

y: a vector of response values.groups: a vector of values indicating the "group" an observation belongs in.blocks: a vector of values indicating the "blocking" variable.

This function produces a Chi-Square test statistic and a corresponding p-value. If the p-value is less than a certain significance level (common choices are 0.10, 0.05, and 0.01), then there is sufficient evidence that the means between each of the groups is not equal.

To illustrate how to use this function, we will create a dataset that shows the reaction time of five patients on four different drugs. Since each patient is measured on each of the four drugs, we will use the Friedman Test to determine if the mean reaction time differs between drugs.

First, we'll create the dataset:

```
#create data
data <- data.frame(person = rep(1:5, each=4),
drug = rep(c(1, 2, 3, 4), times=5),
score = c(30, 28, 16, 34, 14, 18, 10, 22, 24, 20,
18, 30, 38, 34, 20, 44, 26, 28, 14, 30))
```

```
#view data
data

person drug score
1 1 1 30
2 1 2 28
3 1 3 16
4 1 4 34
5 2 1 14
6 2 2 18
7 2 3 10
8 2 4 22
9 3 1 24
10 3 2 20
11 3 3 18
12 3 4 30
13 4 1 38
14 4 2 34
15 4 3 20
16 4 4 44
17 5 1 26
18 5 2 28
19 5 3 14
20 5 4 30
```

Then we'll perform the Friedman Test using score as the response variable, drug as the grouping variable, and person as the blocking variable:

```
#perform Friedman Test
friedman.test(y=data$score, groups=data$drug,
blocks=data$person)
```

```
Friedman rank sum test

data: data$score, data$drug and data$person
Friedman chi-squared = 13.56, df = 3, p-value = 0.00357
```

The Chi-Squared test statistic is 13.56 and the corresponding p-value is 0.00357. Because this p-value is less than 0.05, we can reject the null hypothesis that the mean response time is the same for all four drugs. We have sufficient evidence to conclude that the type of drug used lead to statistically significant differences in response time.

Although a Friedman Test tells us if there are differences in mean response time across the drugs, it doesn't tell us specifically which drugs have different mean response times. To figure that out, we need to conduct post-hoc tests.

For a Friedman Test, the appropriate post-hoc test is the pairwise Wilcoxon rank sum test with a bonferroni correction, which can be implemented using the following syntax:

```
pairwise.wilcox.test(data$score, data$drug, p.adj = "bonf")
```

x: response vectorg: grouping vectorp.adj: method for adjusting p-values; options include holm, hochberg, hommel, bonferroni, BH, BY, fdr, and none

Here is the syntax we will use for our example:

```
#perform post-hoc tests
pairwise.wilcox.test(data$score, data$drug, p.adj = "bonf")
Pairwise comparisons using Wilcoxon rank sum test

data: data$score and data$drug

1 2 3
2 1.000 - -
3 0.449 0.210 -
```

4 1.000 1.000 0.072

P value adjustment method: bonferroni

This produces a matrix that shows the p-value for each pairwise Wilcoxon rank sum test. We can see that the only drug groups that have a statistically significant difference at 0.10 are groups 3 and 4 (p = 0.072).