

How to Perform the Friedman Test in Python

Authored by
stats writer

December 25, 2025

RECOMMENDED CITATION

stats writer (2025). # How to Perform the Friedman Test in Python. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=108727>

The Friedman Test, when implemented in Python, serves as a powerful non-parametric test specifically designed to assess whether significant differences exist among multiple groups of related data. This method is fundamentally utilized when comparing the measurements derived from two or more related samples--often situations where the same subjects are measured repeatedly under different conditions. The computation involves calculating the chi-squared statistic, which is subsequently compared against the critical values derived from the chi-squared distribution to derive a conclusive statistical inference. For efficient execution within the Python ecosystem, the function `scipy.stats.friedmanchisquare()` provides the necessary statistical engine.

The Friedman Test is widely recognized as the non-parametric alternative to the one-way repeated-measures ANOVA. It is specifically employed when researchers need to determine if there is a statistically significant difference between the central tendencies--often the medians--of three or more dependent groups, where the critical characteristic is that the same subjects (or matched units) are present across every single group comparison. This dependency is what distinguishes it from tests designed for independent samples.

This comprehensive tutorial will guide you through the theoretical underpinnings and the practical steps required to execute the Friedman Test successfully using the Scipy library in Python, ensuring clarity and accuracy in your statistical analysis workflow.

Understanding the Necessity of Non-Parametric Testing

The selection of the appropriate statistical test hinges critically upon the underlying distribution of the data. While tests like the repeated-measures ANOVA are robust and powerful, they rely on stringent assumptions, primarily that the data within each group are normally distributed and that sphericity (homogeneity of variances of the differences between treatment levels) is met. In real-world data science and research, these assumptions are often violated, particularly when dealing with small sample sizes, ordinal data, or data sets exhibiting severe skewness.

When the assumptions required for parametric tests cannot be satisfied, the Friedman Test steps in as a reliable non-parametric counterpart. It does not require assumptions about the population distribution being normal. Instead of relying on raw scores and means, the Friedman Test operates on the rank-transformed data, effectively mitigating the influence of outliers and non-normal distribution patterns. This makes it an essential tool for researchers across fields like psychology, medicine, and social sciences where data frequently fails to meet parametric requirements.

In essence, if you have collected data where subjects are exposed to three or more different treatments (groups), and you are certain that the distributional assumptions of ANOVA are violated--or if your data is measured on an ordinal scale--the Friedman Test provides a statistically sound method for determining if any of the treatments result in significantly different outcomes. It

tests the hypothesis that the distributions (and thus, the medians) of the related populations are identical.

Key Assumptions and Prerequisites for the Friedman Test

While the Friedman Test avoids the strict distributional requirements of parametric tests, it still operates under a set of specific assumptions that must be met to ensure the validity of the results. Understanding these prerequisites is vital before proceeding with implementation in [Python](#).

Dependent Samples: The test requires that the samples are dependent. This means the observations across the different treatment conditions must come from the same set of subjects or matched units. If the groups are independent, other non-parametric tests, such as the Kruskal-Wallis H Test, would be more appropriate.

Measurement Scale: The dependent variable must be measured at least on an ordinal scale. Although interval or ratio data are often used, the test converts these scores into ranks, meaning the actual numerical values are less important than their relative ordering within each subject.

Three or More Groups: The Friedman Test is designed for comparing three or more treatment conditions ($k \geq 3$). If only two related groups are being compared, the non-parametric Wilcoxon Signed-Rank Test should be used instead.

Failure to satisfy these core structural assumptions, even though the test is non-parametric, can lead to incorrect conclusions. Therefore, rigorous verification of the experimental design, confirming repeated measurements on the same subjects across three or more conditions, is the crucial first step before writing any code.

Example: The Friedman Test in Python

To illustrate the application of this statistical procedure, consider a classic scenario in clinical research. A researcher aims to determine if the reaction times of patients are significantly different when treated with three distinct pharmacological drugs. This is a clear case of repeated measures, as the same patients are tested under the influence of Drug 1, Drug 2, and Drug 3.

The experimental design involves measuring the reaction time (in seconds) for ten different patients across all three conditions. The goal is to use the Friedman Test in [Scipy](#) to statistically assess whether the median reaction time is consistent across the three drug treatments or if at least one drug significantly alters the patient's response time.

This study design perfectly aligns with the requirement for dependent samples (the ten patients serve as the blocks or subjects) and three comparison groups (the three drugs). We will proceed with the calculation, recognizing that we are testing for differences in ranks, which, in turn, reflects

differences in central location between the drug distributions.

Step 1: Preparing and Entering the Data in Python

The first practical step in executing the Friedman Test in Python involves structuring the data appropriately. Since the data represents dependent samples, we must organize the results such that each list or array corresponds to a single treatment group, with the corresponding indices linking back to the same patient.

We will utilize standard Python lists (which Scipy functions can handle, though NumPy arrays are often preferred for larger datasets) to store the reaction times for each patient on each of the three drugs. It is crucial to ensure that the order of data entry is maintained consistently across all three lists, reflecting the measurement sequence for Patient 1, Patient 2, and so forth.

The following code snippet defines the datasets required for the analysis:

```
group1 =  
group2 =  
group3 =
```

Here, `group1` holds the reaction times under Drug 1, `group2` under Drug 2, and `group3` under Drug 3. Note that the first element of each list (e.g., 4, 5, 2) corresponds to the measurements taken from the same patient (Patient 1). This structure is essential for the test, as it performs the ranking procedure horizontally (within subjects) rather than vertically (across groups).

Step 2: Executing the Friedman Test using Scipy

Once the data is correctly structured, we can invoke the core function provided by the [Scipy](#) library, specifically within its statistics module (`scipy.stats`). The function `friedmanchisquare()` takes the data sets as positional arguments--one argument for each related group being compared.

To begin the analysis, we must first import the `stats` module from Scipy. The function then calculates the test statistic based on the ranks assigned to the data points within each subject, and simultaneously calculates the corresponding p-value, which determines the statistical significance of the findings.

The code required to perform this calculation is exceptionally concise, leveraging the statistical power encapsulated within the Scipy ecosystem:

```
from scipy import stats
```

```
#perform Friedman Test
```

```
stats.friedmanchisquare(group1, group2, group3)  
  
(statistic=13.3514, pvalue=0.00126)
```

The output provides two crucial metrics: the calculated test statistic (a value analogous to the F-ratio in ANOVA, but derived from rank sums) and the resulting p-value. The test statistic of 13.3514 reflects the magnitude of the differences observed between the ranks of the three groups. The p-value, standing at 0.00126, is the probability of observing such extreme results if the null hypothesis were true.

Step 3: Defining Hypotheses and Interpreting the Results

Statistical inference requires a formal comparison of the obtained results against predefined hypotheses. The Friedman Test operates under a standard framework of the null hypothesis (H_0) and the alternative hypothesis (H_a), which must be stated clearly before interpretation.

The hypotheses for the Friedman Test are defined as follows:

The null hypothesis (H_0): The median reaction times for all three drugs are equal. That is, the three population distributions are identical in terms of location.

The alternative hypothesis (H_a): At least one of the population medians is different from the others. That is, the drug type does have a statistically significant effect on reaction time.

To make a decision, we compare the calculated p-value (0.00126) to a predetermined level of significance (α), typically set at 0.05. The decision rule is straightforward: if the p-value is less than α , we reject H_0 ; otherwise, we fail to reject H_0 .

In this specific example, the obtained p-value ($\text{p} = \mathbf{0.00126}$) is substantially less than the standard significance level ($\alpha = 0.05$). Consequently, we reject the null hypothesis that the mean (or median) response time is the same for all three drugs. This rejection is supported by the magnitude of the test statistic ($\mathbf{13.3514}$).

Conclusion and Post-Hoc Analysis Consideration

The overall conclusion drawn from the Python implementation is robust: there is sufficient statistical evidence to conclude that the type of drug administered leads to statistically significant differences in patient reaction time. This means that at least one of the drugs performs differently from the others regarding the speed of patient response.

It is crucial to remember that the Friedman Test is an omnibus test; it only tells us that a difference exists somewhere among the groups. It does not specify which pairs of drugs are significantly

different from each other (e.g., whether Drug 1 differs from Drug 2, or Drug 2 differs from Drug 3). To identify these specific pairwise differences, subsequent analysis, known as post-hoc analysis, must be performed.

Common non-parametric post-hoc procedures often involve performing multiple Wilcoxon Signed-Rank Tests between pairs of groups, coupled with a correction method (such as the Bonferroni correction or the Nemenyi test) to control the family-wise error rate stemming from multiple comparisons. By combining the powerful initial assessment of the Friedman Test with a careful post-hoc approach, researchers can gain detailed and actionable insights from their repeated-measures data, even when distributional assumptions are not met.

ARABPSYCHOLOGY.COM