

Learn How to Perform a One Sample T-Test in Python

Authored by
Mohammed loot

November 8, 2025

RECOMMENDED CITATION

Mohammed loot (2025). *Learn How to Perform a One Sample T-Test in Python*. PSYCHOLOGICAL STATISTICS. Retrieved from <https://statistics.arabpsychology.com/?p=12697>

In the realm of [inferential statistics](#), the ability to draw robust conclusions about a vast population based on limited sample data is paramount. The [one sample t-test](#) is a fundamental statistical procedure designed precisely for this purpose. Its primary function is to rigorously determine whether the true average, known as the [population mean](#) (μ), of a single group exhibits a statistically significant departure from a specific, predetermined hypothesized value. This technique is indispensable for researchers who need to validate whether their sample findings align with established benchmarks, historical data, or theoretical expectations, providing empirical evidence for their claims.

This comprehensive tutorial serves as a definitive guide, meticulously detailing the theoretical underpinnings of the one sample t-test while simultaneously offering a clear, step-by-step methodology for its practical execution. We will harness the immense computational power of [Python](#), specifically leveraging the renowned **SciPy** library. SciPy is the cornerstone of scientific computing in Python, providing highly optimized functions for sophisticated statistical analysis, thus ensuring that your data investigations yield reliable, accurate, and reproducible results suitable for academic or industry application.

The Core Concept of the One Sample T-Test

The t-test family of inferential statistics is primarily employed when the goal is to compare means. The [one sample t-test](#) differentiates itself by focusing on a single group's average in relation to a fixed external value, rather than comparing two separate groups. This test is particularly valuable in situations where the **population standard deviation** remains unknown, compelling the analyst to rely solely on the sample standard deviation as an estimate. Furthermore, the t-test is generally preferred over the z-test when dealing with relatively small sample sizes, traditionally those consisting of fewer than 30 observations, as it accounts for the increased uncertainty inherent in smaller datasets.

At the heart of this procedure is the calculation of the **t-statistic**. This statistic acts as a standardized measure, quantifying the difference observed between the sample mean and the hypothesized [population mean](#), relative to the variability (or standard error) within the sample data itself. A large absolute value of the t-statistic signifies that the difference is substantial compared to the noise in the data, thereby increasing the likelihood that the sample mean is genuinely statistically distinct from the value being tested. Conversely, a t-statistic close to zero suggests that the observed discrepancy could easily be attributed to random sampling variation.

The reliability of the t-test rests upon several crucial statistical assumptions. Firstly, the observations within the sample must be independent; the measurement of one unit should not influence the measurement of any other unit. Secondly, the population from which the sample is drawn must be approximately **normally distributed**. While the t-test exhibits a degree of

robustness against minor violations of normality, especially as the sample size increases (due to the Central Limit Theorem), verifying this condition through techniques like Q-Q plots or specific normality tests (e.g., Shapiro-Wilk) is considered best practice to ensure the validity of the resulting **p-value**.

Assumptions and Hypotheses: Setting the Statistical Stage

Every formal hypothesis test, including the one sample t-test, begins with the definition of two mutually exclusive statements: the [null hypothesis](#) (H_0) and the alternative hypothesis (H_1). These hypotheses provide the critical framework against which the sample data is evaluated. The H_0 always represents the status quo or the statement of no effect or no difference, asserting that the true [population mean](#) (μ) is exactly equal to the hypothesized value (μ_0).

Conversely, the alternative hypothesis (H_1) proposes that a difference does exist. This statement can take three forms: the population mean is not equal to ($\mu \neq \mu_0$, a two-sided test), is greater than ($\mu > \mu_0$), or is less than ($\mu < \mu_0$, a one-sided test). For the majority of general scientific and quality control applications, the two-sided test is employed, focusing on whether the true mean deviates in either direction from the benchmark. This framework ensures a rigorous and unbiased assessment of the available statistical evidence.

It is essential to understand that hypothesis testing does not seek to "prove" the alternative hypothesis. Instead, the process aims to determine if there is enough compelling evidence from the sample to justify **rejecting the null hypothesis**. The strength of this evidence is ultimately quantified by the [p-value](#), which represents the probability of observing our sample data (or data more extreme) if the null hypothesis were, in fact, true. A small p-value suggests that the observed data is highly unlikely under the null assumption, leading to its rejection.

Case Study Introduction: The Botanist's Dilemma

To demonstrate the practical utility of the one sample t-test, let us explore a compelling real-world scenario. Imagine a botanist conducting research on a newly cultivated strain of plant species. Based on genetic modeling and historical growth patterns, the botanist has a strong theoretical expectation that the average height of this species should be exactly 15 inches. She needs to statistically verify if her actual cultivated plants meet this established standard or if their mean height has diverged.

The botanist meticulously collects a random and representative sample of 12 individual plants, measuring and recording the height of each specimen in inches. This collection of 12 data points constitutes our sample, which must now be used to make an accurate statistical inference about the entire, much larger, population of this plant species. The sample is small, and the population standard deviation is unknown, making the one sample t-test the perfectly suited analytical tool.

The central statistical question driving this study is whether the sample mean height is sufficiently far from 15 inches to warrant rejecting the established belief. We formalize this into our hypotheses: The [null hypothesis](#) (H_0) asserts that the mean height for this species is 15 inches ($\mu = 15$). Conversely, the alternative hypothesis (H_1) asserts that the mean height is not 15 inches ($\mu \neq 15$). The subsequent steps in [Python](#) will execute the necessary calculations to evaluate these hypotheses against the collected empirical data.

Step 1: Data Preparation and Python Implementation Setup

The foundational step in any computational statistical analysis requires structuring the raw data for efficient processing. In the context of [Python](#), the 12 recorded plant heights must be organized into a data structure that can be seamlessly consumed by the functions within the **SciPy** library. While simple Python lists are functional for small datasets, analysts often utilize **NumPy arrays** for larger datasets due to their optimized memory management and specialized mathematical operations. For our current example, a standard list will suffice, ensuring the data points are clearly defined.

The specific measurements gathered by the botanist, representing the height of the 12 plants, are as follows (in inches): 14, 14, 16, 13, 12, 17, 15, 14, 15, 13, 15, and 14. We must explicitly define this data array in Python, along with the hypothesized value we are testing against, ensuring both are correctly assigned to variables that will be passed into the statistical function.

```
data =  
hypothesized_mean = 15
```

This organized structure, labeled **data**, is now fully prepared for statistical manipulation. Before proceeding to the next step, it is good practice to visually inspect the data and perhaps calculate descriptive statistics (mean, median, standard deviation) to get an initial sense of the distribution and how far the sample average deviates from the hypothesized [population mean](#) of 15 inches. (In this case, the sample mean is approximately 14.33 inches.)

Step 2: Executing the Test with `scipy.stats`

The statistical heavy lifting for the one sample t-test is performed by the dedicated function [ttest_1samp\(\)](#), which resides within the powerful **scipy.stats** module. This function automates all the necessary complex calculations, including determining the appropriate degrees of freedom ($n-1$), computing the precise t-statistic, and most importantly, deriving the two-sided [p-value](#).

The function is designed for maximum efficiency and clarity, requiring only two mandatory input arguments to correctly execute the test:

a: This parameter requires the array or list containing the **sample observations** (our `data` list of

plant heights).

popmean: This parameter specifies the **hypothesized population mean**, which is the fixed value we are testing against (15 inches in this specific botanical example).

To perform the test, we must first import the required components of the [scipy.stats](#) library and then call the function, passing in our defined variables. The resulting output will be a tuple containing the two crucial metrics needed for interpretation.

import scipy.stats as stats

```
# Perform the one sample t-test using the defined data and hypothesized mean
```

```
results = stats.ttest_1samp(a=data, popmean=15)
```

```
# Print the results
```

```
print(results)
```

```
(statistic=-1.6848, pvalue=0.1201)
```

Upon execution, the Python function successfully returns a tuple containing the two core statistical measurements: the calculated **t-statistic**, which is approximately **-1.6848**, and the corresponding two-sided **p-value**, found to be **0.1201**. These numerical results are the empirical evidence upon which the final statistical decision concerning the botanist's hypothesis must be based.

Step 3: Analyzing the T-Statistic and P-Value

The final stage of the hypothesis testing framework is interpretation, where the statistical findings are translated into a meaningful conclusion regarding the real-world scenario. This interpretation is governed by the comparison between the calculated **p-value** and the chosen significance level, denoted by the Greek letter alpha (α). By convention in most scientific disciplines, the significance level is set at $\alpha = 0.05$. This threshold signifies that researchers are willing to accept only a 5% risk of committing a Type I error--the error of incorrectly rejecting a true null hypothesis.

The formal decision rule is absolute: If the p-value is less than or equal to α ($p\text{-value} \leq 0.05$), we possess sufficient evidence to **reject the null hypothesis**, concluding that the true mean is statistically different from the hypothesized value. Conversely, if the p-value is greater than α ($p\text{-value} > 0.05$), we must **fail to reject the null hypothesis**, indicating that the observed difference is likely due to chance sampling variation.

In our plant height case study, we established the critical threshold at $\alpha = 0.05$. Our calculated **p-value** is **0.1201**. Since 0.1201 is significantly **greater than** 0.05, we are compelled to conclude that we **fail to reject the null hypothesis**. This outcome indicates that the difference between the

sample mean (14.33 inches) and the hypothesized [population mean](#) (15 inches) is not large enough, relative to the data variability, to be considered statistically significant at the 5% level. Therefore, based on the statistical evidence gathered from the 12 sampled plants, the botanist lacks sufficient grounds to definitively state that the average height of the entire species population differs from 15 inches.

Conclusion and Further Statistical Exploration

Mastering the one sample t-test in Python, particularly through the use of the [scipy.stats](#) library, equips analysts with a critical tool for foundational statistical inference. This procedure allows for the robust comparison of a sample mean against a fixed population parameter, providing quantitative evidence for critical decision-making processes across various fields, from quality control and manufacturing to academic research.

While the one sample t-test addresses the question of whether a single sample mean deviates from a known constant, many research designs require comparisons between two or more groups. To further expand your proficiency in statistical testing and accommodate different experimental designs, consider exploring these related methodologies:

To further expand your knowledge of statistical testing in Python, explore these related tutorials on other variations of the t-test, which address different experimental designs and comparisons:

[How to Conduct a Two Sample T-Test in Python](#)

[How to Conduct a Paired Samples T-Test in Python](#)