

How can a One Proportion Z-Test be performed in Python?

Authored by
stats writer

April 21, 2024

RECOMMENDED CITATION

stats writer (2024). *How can a One Proportion Z-Test be performed in Python?*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=137604>

A One Proportion Z-Test can be performed in Python by first importing the necessary libraries such as statsmodels and scipy. Then, the sample proportion and sample size must be calculated from the given data. Next, the null and alternative hypothesis must be defined and the critical value or p-value must be determined. Using the z-test formula, the test statistic can be calculated. Finally, the p-value can be compared to the significance level to determine if the null hypothesis can be rejected or not. This process can be repeated for multiple samples or proportions, making Python a useful tool for performing One Proportion Z-Tests in a quick and efficient manner.

Perform a One Proportion Z-Test in Python

A one proportion z-test is used to compare an observed proportion to a theoretical one.

This test uses the following null hypotheses:

$H_0: p = p_0$ (population proportion is equal to hypothesized proportion p_0)

The alternative hypothesis can be either two-tailed, left-tailed, or right-tailed:

H_1 (two-tailed): $p \neq p_0$ (population proportion is not equal to some hypothesized value p_0)
 H_1 (left-tailed): $p < p_0$ (population proportion is less than some hypothesized value p_0)
 H_1 (right-tailed): $p > p_0$ (population proportion is greater than some hypothesized value p_0)

The test statistic is calculated as:

$$z = (p-p_0) / \sqrt{p_0(1-p_0)/n}$$

where:

p: observed sample proportion
p₀: hypothesized population proportion
n: sample size

If the p-value that corresponds to the test statistic **z** is less than your chosen significance level (common choices are 0.10, 0.05, and 0.01) then you can reject the null hypothesis.

One Proportion Z-Test in Python

To perform a one proportion z-test in Python, we can use the `proportions_ztest()` function from the `statsmodels` library, which uses the following syntax:

```
proportions_ztest(count, nobs, value=None, alternative='two-sided')
```

where:

count: The number of successes
nobs: The number of trials
value: The hypothesized population

proportionalternative: The alternative hypothesis

This function returns a z test-statistic and a corresponding p-value.

The following example shows how to use this function to perform a one proportion z-test in Python.

Example: One Proportion Z-Test in Python

Suppose we want to know whether or not the proportion of residents in a certain county who support a certain law is equal to 60%. To test this, we collect the following data on a random sample:

**p0: hypothesized population proportion = 0.60
x: residents who support law: 64
n: sample size = 100**

The following code shows how to use the proportions_ztest function to perform a one sample z-test:

```
#import proportions_ztest function  
from statsmodels.stats.proportion import  
proportions_ztest
```

```
#perform one proportion z-test
```

```
proportions_ztest(count=60, nobs=100, value=0.64)  
(-0.8164965809277268, 0.41421617824252466)
```

From the output we can see that the z test-statistic is -0.8165 and the corresponding p-value is 0.4142. Since this value is not less than $\alpha = 0.05$, we fail to reject the null hypothesis. We do not have sufficient evidence to say that the proportion of residents who support the law is different from 0.60.

[An Introduction to the One Proportion Z-Test](#)

[One Proportion Z-Test Calculator](#)

[How to Perform a One Proportion Z-Test in Excel](#)

[How to Perform a One Proportion Z-Test in R](#)