

Learn How to Perform a Chi-Square Test of Independence in Python

Authored by
Mohammed looti

November 8, 2025

RECOMMENDED CITATION

Mohammed looti (2025). *Learn How to Perform a Chi-Square Test of Independence in Python*. PSYCHOLOGICAL STATISTICS. Retrieved from <https://statistics.arabpsychology.com/?p=12700>

The [Chi-Square Test of Independence](#) is a foundational statistical tool utilized to ascertain whether a statistically significant relationship exists between two [categorical variables](#). Unlike tests designed for continuous data, the Chi-Square test operates on frequencies and counts, making it indispensable for analyzing survey responses, demographic data, and other non-numeric classifications.

Mastering this test in Python allows data scientists and analysts to quickly determine if observed differences in sample data are likely due to chance or if they reflect a genuine association between the underlying populations. This comprehensive tutorial walks you through the practical steps of performing the Chi-Square Test of Independence using the powerful [SciPy library](#) in Python.

Understanding the Theory Behind the Test

Before diving into the code, it is essential to grasp the core concepts. The Chi-Square Test of Independence evaluates whether the distribution of one variable is independent of the distribution of the second variable. If the variables are truly independent, the distribution of one should remain consistent across all categories of the other. The test works by comparing the **observed frequencies** (what we actually measured) against the **expected frequencies** (what we would expect to see if the null hypothesis were true).

The resulting test statistic, often denoted as χ^2 , quantifies the difference between these observed and expected counts. A large χ^2 value suggests that the observed data deviates significantly from what independence would predict, thus increasing the likelihood of rejecting the null hypothesis. Conversely, a small χ^2 value indicates close alignment between observed and expected counts, suggesting the variables are likely independent.

This test is particularly sensitive to sample size. While the mechanics are straightforward, proper interpretation requires careful consideration of the degrees of freedom and the context of the data collection process. Furthermore, the data must be presented in a specific format known as a [contingency table](#), where the rows represent one categorical variable and the columns represent the second.

Defining Hypotheses for Statistical Testing

Every formal statistical test begins with the establishment of two competing hypotheses: the null hypothesis (H_0) and the alternative hypothesis (H_1). These hypotheses guide the interpretation of the test results and dictate the appropriate course of action based on the calculated [p-value](#). For the Chi-Square Test of Independence, these hypotheses are defined as follows:

H0: (Null Hypothesis) The two categorical variables are **independent**. There is no significant

association between them in the population.

H1: (Alternative Hypothesis) The two categorical variables are **not independent**. There is a significant association between them.

Our goal is to gather enough statistical evidence to either reject the null hypothesis in favor of the alternative, or fail to reject the null hypothesis. We make this determination by comparing the p-value against a predetermined significance level (α), commonly set at 0.05. If the p-value is less than or equal to 0.05, we reject H_0 .

Case Study: Assessing Political Preference vs. Gender

To illustrate the implementation of the Chi-Square Test of Independence in Python, consider a scenario where we wish to explore whether there is an association between an individual's gender and their stated political party preference. This is a classic application, as both variables (Gender and Political Party) are categorical.

We conducted a simple random sample of 500 registered voters and recorded their responses. The results, structured as a contingency table, summarize the counts for each combination of gender and political affiliation. This observed data forms the foundation for our analysis, allowing us to compare these counts against the expected distribution if gender and party were truly independent.

The table below displays the raw counts collected from the survey:

	Republican	Democrat	Independent	Total
Male	120	90	40	250
Female	110	95	45	250
Total	230	185	85	500

Our objective is to use Python to perform the statistical calculations necessary to determine if the differences observed in the cells of this [contingency table](#) are significant enough to conclude that gender and political preference are associated.

Step 1: Preparing the Data Structure in Python

The first crucial step in Python is translating the raw counts from the contingency table into a suitable data structure that the statistical function can process. The `chi2_contingency` function from SciPy expects the input to be a two-dimensional array or list of lists representing the observed counts, excluding the marginal totals (the 'Total' row and column).

We structure the data where the first list corresponds to the counts for Male voters across the three party preferences, and the second list corresponds to Female voters:

```
data = ,  
]
```

This structure accurately represents the 2x3 table of observed frequencies. It is vital to ensure that the rows and columns in the Python list exactly match the structure defined in the original table for correct calculation of the test statistic and degrees of freedom.

Step 2: Executing the Chi-Square Test Using SciPy

With the data prepared, we utilize the `chi2_contingency` function available within the `scipy.stats` module. This function efficiently performs all necessary computations, including calculating the Chi-Square test statistic, the associated [p-value](#), the degrees of freedom, and the array of expected cell frequencies.

The syntax for calling the function is straightforward: `chi2_contingency(observed)`, where `observed` is the array or list of lists containing the observed frequency data we created in Step 1. We must first import the necessary statistical package.

The following code snippet imports the library and executes the test on our voter preference data:

```
import scipy.stats as stats  
  
#perform the Chi-Square Test of Independence  
stats.chi2_contingency(data)  
  
(0.864,  
0.649,  
2,  
array(  
]))
```

The output generated by the `chi2_contingency` function is a tuple containing four specific values, presented in a fixed order. Understanding each component is critical for proper statistical interpretation.

Interpreting the Results and Drawing Conclusions

The output tuple provides all the necessary components for conducting a formal hypothesis test.

We analyze the four elements returned by the function:

Chi-Square Test Statistic: The first value, **0.864**, represents the calculated χ^2 value. This measures the aggregated difference between the observed and expected counts.

P-value: The second value, **0.649**, is the probability of observing a test statistic as extreme as 0.864 (or more extreme) if the null hypothesis of independence were actually true.

Degrees of Freedom: The third value, **2**, indicates the degrees of freedom for the test. This is calculated as $(\text{number of rows} - 1) \times (\text{number of columns} - 1)$, or $(2-1) \times (3-1) = 2$.

Expected Values Array: The final array displays the **expected frequencies** for each cell in the [contingency table](#), assuming perfect independence between gender and political preference. For instance, the expected number of Republican males is 115.0.

To reach a definitive conclusion, we focus specifically on the **p-value** (0.649). Since we typically use a significance level (α) of 0.05, we compare 0.649 to 0.05.

Because the p-value (0.649) is significantly greater than the chosen significance level ($\alpha = 0.05$), we must **fail to reject the null hypothesis** (H_0). Statistically, this means that we do not have sufficient evidence from this sample to conclude that a significant association exists between gender and political party preference. The small differences observed in the raw counts are likely attributable to random sampling variability rather than a true relationship in the broader population. In statistical terms, gender and political party preference are deemed independent variables based on this test.