How to Calculate Cramer’s V in Python?

Authored by

stats writer

December 15, 2025

The calculation of Cramer's V is essential in statistical analysis when assessing the degree of association between two nominal variables. Unlike correlation coefficients used for continuous data, Cramer's V provides a normalized, interpretable measure derived from the standard Chi-square statistic. This statistic is particularly valuable because it can be applied to contingency tables of any dimension, making it a highly versatile tool in social sciences, marketing research, and data science.

To perform this calculation efficiently in Python, we typically leverage powerful numerical libraries like `scipy` and `numpy`. The process begins with generating a cross-tabulated table, often achieved using the `pandas.crosstab()` method, which organizes the frequency distribution of the two variables under investigation. Once the contingency table is established, the core of the calculation involves utilizing the Chi-square statistic, dividing it by the total sample size, and then accounting for the degrees of freedom inherent in the table structure. This rigorous process yields the Cramer's V value, which ranges strictly between 0 and 1, allowing for a clear interpretation of the strength of the statistical relationship between the categorical fields. A resulting value close to 1 suggests a very strong relationship, while a value near 0 indicates little to no association, providing actionable insights into the underlying dataset.

Cramer's V serves as a robust measure of the strength of association when analyzing relationships between two categorical fields. It is an adaptation of Pearson's Chi-square statistic, designed specifically to normalize the result, making it comparable across different table sizes and sample totals. This normalization is crucial because the raw Chi-square value is highly sensitive to the total sample size; a large sample size can inflate the Chi-square statistic even if the underlying association is weak. By scaling the Chi-square value relative to the sample size and the degrees of freedom, Cramer's V provides a stable and reliable metric for reporting the intensity of the relationship between variables.

The resulting value of Cramer's V is bounded, offering an immediate and intuitive interpretation of the association's power. It operates on a scale from 0 to 1, providing analysts with a standardized index for comparison across various studies or datasets. Understanding this range is fundamental to accurately interpreting the statistical output and translating it into meaningful conclusions about the data's behavior.

0 indicates absolutely no measurable association or dependency between the two categorical variables being analyzed. This suggests that knowledge of one variable's category provides no information regarding the likelihood of the other variable's category.
1 indicates a perfect, strong association between the two variables. In this scenario, the categories of one variable perfectly predict the categories of the second variable, representing the maximum possible level of dependency.

The specific mathematical formula for calculating Cramer's V ensures this normalization by incorporating the fundamental elements derived from the contingency table structure. It is rooted in the concepts of the Chi-square test and the table's dimensionality, guaranteeing a robust and unbiased measure of association regardless of the dataset's size or complexity.

Cramer's V = √(X2/n) / min(c-1, r-1)

The components of this formula are crucial for understanding the mechanics of the statistic. Each variable in the equation represents a measurable aspect of the contingency table, ensuring that the final V value is correctly scaled against the theoretical maximum association possible for that specific table dimension.

X2: This is the calculated Chi-square statistic, which quantifies the discrepancy between the observed cell frequencies and the frequencies that would be expected if the two variables were entirely independent.
n: This represents the total sample size, which is the sum of all observations recorded across all cells within the contingency table.
r: This denotes the total number of rows present in the contingency table, representing the categories of the first variable.
c: This denotes the total number of columns present in the contingency table, representing the categories of the second variable.

This comprehensive tutorial is designed to walk through the practical application of this statistical measure, providing clear examples and detailed code for calculating Cramer's V within the Python environment, specifically targeting both standard 2x2 tables and larger, more complex contingency matrices.

Setting Up the Python Environment for Calculation

Before diving into the core calculations, it is paramount to ensure that the necessary statistical libraries are properly installed and imported into the Python working environment. For calculating the Chi-square statistic--which forms the foundation of Cramer's V--we rely heavily on the `scipy.stats` module. Furthermore, numerical manipulation, array creation, and summation tasks are efficiently handled by the ubiquitous `numpy` library. These two packages represent the standard toolkit for advanced statistical computing in the Python ecosystem, providing optimized functions for data manipulation and hypothesis testing.

The calculation sequence always starts with the data representation itself. Since Cramer's V operates on frequency data, the initial step involves defining the contingency table, which maps the joint frequency distribution of the two categorical variables. In Python, this table is typically structured as a `numpy` array or a `pandas` DataFrame. For demonstration purposes, we will utilize

`numpy` arrays to define the cell counts directly, making the code concise and focused purely on the statistical derivation.

The general workflow involves three critical computational steps once the data array is defined: first, calculating the raw Chi-square statistic using `scipy.stats.chi2_contingency`; second, determining the total sample size (n) by summing all cell frequencies; and third, identifying the necessary degrees of freedom component, which is defined by the minimum of (rows - 1) or (columns - 1). These three outputs are then combined according to the Cramer's V formula to yield the final, standardized measure of association.

Example 1: Cramer's V for a Standard 2x2 Table

The 2x2 table is the simplest form of a contingency table, often used when comparing two binary or dichotomous variables (e.g., success/failure, male/female). While some measures like the Phi coefficient are suitable for 2x2 tables, Cramer's V generalizes the association measure and remains perfectly valid for this case, often yielding an identical result to Phi. This example illustrates the fundamental steps required to derive the statistic, providing a clean, reproducible script that utilizes the necessary functions from `scipy` and `numpy`.

The following code snippet demonstrates precisely how to initialize the data matrix, calculate the core components (Chi-squared, sample size, and minimum dimension adjustment), and finally compute Cramer's V for a hypothetical dataset represented by a 2x2 matrix. The `chi2_contingency` function is particularly useful as it handles the expected frequency calculation internally and returns the Chi-square test statistic (X2), among other values. We specifically access the first element of its output, which corresponds to X2.

#load necessary packages and functions
import scipy.stats as stats
import numpy as np

#create 2x2 table
data = np.array(,])

#Chi-squared test statistic, sample size, and minimum of rows and columns
X2 = stats.chi2_contingency(data, correction=False)
n = np.sum(data)
minDim = min(data.shape)-1

#calculate Cramer's V
V = np.sqrt((X2/n) / minDim)

#display Cramer's V

```
print(V)
```

```
0.1617
```

Upon execution, the resulting Cramer's V value is computed as **0.1617**. Given the defined range of 0 to 1, this result signifies a relatively weak association between the two dichotomous variables represented in the input table. In practical terms, while there is some dependency detected, it is minor, suggesting that the categorization in one variable does not strongly influence the categorization in the other. This interpretation is vital for making informed decisions based on the statistical findings, helping researchers avoid overstating the relationship's significance when the effect size is small.

Interpreting the Cramer's V Value

The interpretation of Cramer's V should always be contextual, although general guidelines help categorize the strength of the association. Because Cramer's V is an effect size measure, its magnitude is more informative than the p-value derived from the Chi-square test, especially in large datasets where small associations might appear statistically significant. For V values, standard interpretations generally classify magnitudes near 0.1 as weak, 0.3 as moderate, and 0.5 or higher as strong associations.

In the preceding 2x2 example, where V = 0.1617, the association falls just above the threshold typically considered minimal or weak. This suggests that the categories of Variable A are only slightly better than random in predicting the categories of Variable B. Had the value been closer to 0.5, we would conclude that there is a substantial, meaningful relationship between the categorical variables, warranting deeper investigation into the nature of their dependency. Researchers must always combine this numerical interpretation with domain knowledge to ensure the statistical finding translates into a relevant real-world observation.

It is important to remember that Cramer's V measures the magnitude of the association but provides no insight into the directionality of the relationship, unlike coefficients such as Pearson's r. Since it is based on the non-directional Chi-square test, the analyst must examine the observed frequencies in the contingency table to understand which specific cells contribute most heavily to the overall association. A high V value simply confirms that the distribution of one variable is highly dependent on the category of the other; the specific pattern of that dependency must be manually inferred from the frequency counts.

Example 2: Extending the Calculation to Larger Tables

One of the primary advantages of using Cramer's V is its suitability for contingency tables of any

size, unlike measures restricted to 2x2 matrices. Whether dealing with a 2x3, 3x5, or even larger table, the underlying mathematical formula remains consistent, requiring only that the inputs for the Chi-square statistic (X2), sample size (n), and the minimum dimension adjustment (minDim) are correctly derived from the larger matrix structure. This flexibility makes it the default choice for measuring association among multiple-category nominal variables.

The crucial step when dealing with larger tables is accurately calculating the minimum dimension parameter. This value, represented by `min(r-1, c-1)`, ensures that the resulting V statistic is correctly bounded by 1, regardless of how many rows or columns the table contains. If the dimension adjustment were ignored, the normalization factor would be incorrect, potentially leading to an inflated or skewed measure of association. In our Python implementation using `numpy`, `min(data.shape) - 1` handles this calculation automatically and reliably for any input array size.

The following code demonstrates the calculation for a 3x2 table, representing two variables where the first variable has three categories (rows) and the second has two categories (columns). Note that the structure of the computational steps remains identical to the 2x2 example, highlighting the versatility and robustness of this standardized approach.

#load necessary packages and functions
import scipy.stats as stats
import numpy as np

#create 2x2 table
data = np.array(, ,])

#Chi-squared test statistic, sample size, and minimum of rows and columns
X2 = stats.chi2_contingency(data, correction=False)
n = np.sum(data)
minDim = min(data.shape)-1

#calculate Cramer's V
V = np.sqrt((X2/n) / minDim)

#display Cramer's V
print(V)

0.1775

In this expanded example, Cramer's V is calculated as **0.1775**. Similar to the previous result, this magnitude suggests a weak, albeit slightly stronger, association compared to the 2x2 table example. The consistent methodology across different table sizes confirms that the V statistic is a

highly reliable and scalable measure for determining the effect size of the relationship between any two nominal variables.

Advantages and Considerations for Using Cramer's V

The primary advantage of Cramer's V lies in its interpretability and general applicability. By scaling the Chi-square statistic between 0 and 1, it provides a universal metric for association that is independent of the table's dimensions or the specific sample size, allowing for direct comparison across diverse statistical studies. It is the preferred measure of association for categorical data when the variables are strictly nominal variables, especially in situations where researchers need an effect size that complements the hypothesis testing provided by the Chi-square test itself. This normalization prevents the inflation of perceived association strength that often plagues raw Chi-square values derived from large samples.

However, analysts must be mindful of certain considerations. While Cramer's V is robust, its interpretation can become challenging when one of the dimensions (rows or columns) is significantly larger than the other. In tables that are highly rectangular (e.g., 2x10), the maximum value of V might be distorted, making it difficult to achieve a perfect 1 even in cases of near-perfect association. Furthermore, like all Chi-square related statistics, Cramer's V is sensitive to small expected cell frequencies. If too many cells in the contingency table have expected counts below five, the reliability of the underlying Chi-square statistic, and consequently Cramer's V, is reduced. Researchers should consider consolidating categories or employing Fisher's exact test in such scenarios.

In summary, Cramer's V remains an invaluable tool for categorical data analysis, offering a powerful, standardized, and easily reproducible method for quantifying dependency. By systematically applying the steps demonstrated in the Python examples, data practitioners can quickly calculate and confidently interpret the strength of association within their datasets, moving beyond simple p-values to understand the true magnitude of the relationships they are exploring.