

Calculate Skewness & Kurtosis in Python

Authored by
Mohammed loot

November 7, 2025

RECOMMENDED CITATION

Mohammed loot (2025). *Calculate Skewness & Kurtosis in Python*. PSYCHOLOGICAL STATISTICS. Retrieved from <https://statistics.arabpsychology.com/?p=12121>

In the realm of quantitative data analysis and statistical modeling, descriptive statistics often begin with measures of central tendency (like the mean) and variability (like the standard deviation). However, to truly grasp the nature of a dataset, data scientists must examine the underlying [probability distribution](#). The shape of this distribution provides critical context regarding data concentration and the presence of extreme values. Two fundamental metrics used to quantify this shape are [skewness](#) and [kurtosis](#). Understanding and calculating these metrics are prerequisites for performing robust inferential statistics and selecting appropriate modeling techniques, as many classical methods assume a symmetrically distributed population.

Introduction to Distribution Shape Metrics

While the mean tells us where the data centers and the standard deviation tells us how spread out the data is, skewness and kurtosis offer essential insights into the distribution's deviation from an idealized symmetrical shape. These statistics are known as higher-order moments, describing the qualitative features that standard measures often overlook. Skewness specifically addresses the symmetry of the data, revealing if the bulk of observations are clustered toward one end. Kurtosis, on the other hand, measures the "tailedness" of the distribution, indicating the frequency and magnitude of outliers relative to a standard benchmark distribution, such as the [normal distribution](#). Ignoring these shape characteristics can lead to flawed interpretations and the misapplication of statistical tests that rely on assumptions of normality.

Deep Dive into Skewness: Quantifying Asymmetry

[Skewness](#) is defined as the measure that quantifies the degree of asymmetry exhibited by a probability distribution around its mean. A distribution is considered skewed when its data points are not distributed evenly, resulting in one tail being significantly longer or heavier than the other. This measurement helps analysts immediately identify if the underlying process generating the data favors higher or lower values disproportionately. For instance, income data is typically positively skewed, as the majority of people earn moderate incomes, while a small number of individuals earn extremely high incomes, pulling the right tail far out.

The mathematical calculation of skewness involves the third standardized moment of the data. The resulting value is not just a descriptive statistic; it has significant implications for statistical inference. Highly skewed data often violates the assumptions required for parametric tests like t-tests or ANOVA, necessitating the use of non-parametric alternatives or requiring data transformation techniques, such as logarithmic scaling, to achieve a more symmetrical profile. Therefore, accurately measuring **skewness** is a foundational step in any rigorous exploratory data analysis (EDA) pipeline.

Interpreting Skewness: Understanding the Tail Direction

The sign of the skewness value is crucial, as it dictates the direction of the asymmetry and points toward the relative location of the distribution's long tail. Interpreting this sign correctly allows for immediate qualitative understanding of the data concentration:

A **negative skew** (or left-skewed distribution) occurs when the majority of the data mass is concentrated on the right side of the distribution. This means the left tail is extended and stretched towards more negative values. In practical terms, this suggests that the mean is typically less than the median.

A **positive skew** (or right-skewed distribution) occurs when the data mass is concentrated toward the left side. Consequently, the right tail is longer and stretches towards more positive values. Here, the mean is usually greater than the median, pulled by the high values in the long right tail.

A value of **zero** indicates that the distribution is perfectly symmetrical around its mean. The quintessential example of a zero-skew distribution is the [normal distribution](#) (or Gaussian distribution), where the mean, median, and mode are all equal.

Defining Kurtosis: Measuring Tail Heaviness and Peak Sharpness

[Kurtosis](#) is a statistical measure that assesses the 'tailedness' of a distribution, focusing on the shape of both the tails and the peak relative to the standard [normal distribution](#). Contrary to common misconceptions, kurtosis is not solely about the peak's sharpness; it is primarily about how much probability mass is located in the tails. High kurtosis indicates that extreme values (outliers) are more likely than in a normal distribution, while low kurtosis suggests the opposite. This measure is vital in finance and risk management, where the probability of rare, extreme events (heavy tails) must be accurately modeled.

The raw kurtosis value of a standard normal distribution is 3. Because this baseline value can complicate comparison, most modern statistical analysis relies on **excess kurtosis** (or [Fisher's definition](#)), which is calculated by subtracting 3 from the raw kurtosis. This adjustment sets the normal distribution's benchmark at zero, simplifying interpretation into three distinct categories:

If the excess kurtosis is **less than 0** (raw kurtosis < 3), the distribution is **platykurtic**. Platykurtic distributions have lighter tails and flatter peaks than the normal distribution, implying that outliers are less common and less extreme.

If the excess kurtosis is **greater than 0** (raw kurtosis > 3), the distribution is **leptokurtic**. Leptokurtic distributions have heavier tails and sharper peaks, indicating a higher probability of producing extreme outliers compared to the normal distribution.

If the excess kurtosis is **equal to 0** (raw kurtosis = 3), the distribution is **mesokurtic**, perfectly matching the tail characteristics of the [normal distribution](#).

Understanding whether a dataset is leptokurtic is particularly important when modeling real-world phenomena, as heavy tails often signify underlying volatility or risk that a standard Gaussian model would fail to capture.

Practical Calculation in Python: Leveraging the SciPy Ecosystem

For efficient and accurate calculation of these shape metrics on large datasets within a programming environment, Python's scientific computing stack is indispensable. Specifically, the [SciPy library](#), dedicated to scientific and technical computation, provides highly optimized functions for statistical analysis. The `scipy.stats` module contains the specialized functions `skew()` and `kurt()`, designed to handle arrays of data and calculate the required standardized moments precisely. Leveraging SciPy avoids the need for manual implementation of complex formulas, ensuring speed and reliability in data processing.

A critical consideration when calculating these statistics is the distinction between population parameters and sample statistics. When working with a subset of data (a sample) used to infer properties about the larger population, it is imperative to calculate the unbiased sample statistics. This usually involves applying necessary adjustments, such as [Bessel's correction](#), to ensure that the sample estimates are representative and do not systematically underestimate the true population characteristics. The SciPy functions accommodate this need through explicit function arguments.

Step-by-Step SciPy Implementation and Unbiased Estimation

To demonstrate the proper methodology, let us analyze a hypothetical sample dataset representing scores from a recent test:

```
data =
```

When calculating sample statistics for **skewness** and **kurtosis** using SciPy, the key lies in setting the `bias` parameter to `False`. This instructs the function to use the corrected formula for sample estimation, which incorporates [Bessel's correction](#), thereby providing unbiased estimators of the population characteristics. Failure to set this argument correctly would result in biased (population) estimates, which are generally inappropriate when dealing with sampled data.

The required syntax for calculating these sample shape characteristics within the [SciPy library](#) is straightforward:

`skew(array of values, bias=False)`: Calculates the sample skewness.

`kurtosis(array of values, bias=False)`: Calculates the sample excess kurtosis.

The following code snippet demonstrates the complete Python execution, assuming the necessary imports have been made from `scipy.stats`:

Define the dataset

data =

Calculate sample skewness

```
skew(data, bias=False)
```

0.032697

Calculate sample kurtosis (excess kurtosis, unbiased)

```
kurtosis(data, bias=False)
```

0.118157

Analyzing the Results and Further Resources

The execution yields a sample **skewness** of **0.032697** and a sample **kurtosis** (excess kurtosis) of **0.118157**. These numerical results offer clear insights into the distribution's shape.

First, since the skewness value is a small positive number (close to zero), the distribution is only slightly **positively skewed**. This suggests that while the distribution is nearly symmetrical, there is a very minor tendency for the right tail to be slightly longer, indicating the potential for a few higher scores. Second, because the excess kurtosis is positive (greater than 0), the distribution is classified as **leptokurtic**. This result implies that, compared to a perfect **normal distribution**, this dataset exhibits slightly heavier tails, meaning there is a marginally greater chance of observing extreme test scores (either very high or very low). These insights confirm that the distribution is nearly normal but not perfectly so, guiding the choice of subsequent statistical tests.

To ensure the validity of these manual calculations and to explore these metrics further using interactive tools, external resources are highly recommended. The [Statology Skewness and Kurtosis Calculator](#) automatically computes both the skewness and kurtosis for a given dataset, serving as a valuable educational tool for quick checks and immediate result validation.