

Learn How to Calculate Mean Absolute Percentage Error (MAPE) in Python

Authored by
Mohammed looti

November 8, 2025

RECOMMENDED CITATION

Mohammed looti (2025). *Learn How to Calculate Mean Absolute Percentage Error (MAPE) in Python*. PSYCHOLOGICAL STATISTICS. Retrieved from <https://statistics.arabpsychology.com/?p=12723>

The [Mean Absolute Percentage Error \(MAPE\)](#) stands as a foundational and widely utilized metric for assessing the quality and [predictive accuracy](#) of statistical forecasting models. Unlike scale-dependent error metrics such as the [Mean Squared Error \(MSE\)](#), MAPE provides a measurement of error in relative terms, expressed inherently as a percentage. This crucial characteristic makes MAPE an invaluable tool for both technical analysts and business stakeholders, as it facilitates the direct comparison of model performance across datasets that may have vastly different inherent scales, units, or magnitudes. Its primary purpose is to quantify the average magnitude of the forecast error relative to the actual observation, giving a clear, intuitive indication of how far off the predictions were on average.

A robust understanding of **MAPE** is essential in critical business functions, including inventory management, financial planning, and demand [forecasting](#), where accurate predictions directly influence strategic operational and financial decisions. For instance, when we state that a predictive model has a MAPE of 5%, we are communicating that, on average, the forecasts deviate from the true values by 5% of the actual value itself. This inherent clarity and ease of interpretation are major factors contributing to its enduring popularity across diverse industries. Consequently, the primary objective of any model developer is to achieve a lower MAPE value, as this unequivocally signals a superior ability of the model to capture and predict underlying data patterns accurately and consistently.

Deconstructing the Mean Absolute Percentage Error (MAPE)

The **Mean Absolute Percentage Error** measures the size of the prediction error exclusively in percentage terms, thereby providing a normalized, scale-independent view of forecasting performance. This vital normalization is achieved by mathematically dividing the absolute difference between the actual observed value and the predicted value by the actual value itself. This fundamental step successfully eliminates the dependence on the data scale, ensuring that the resulting error percentage is directly comparable, regardless of whether you are forecasting the sales volume of millions of low-cost items or the future value of a few high-value assets. The metric then aggregates these individual percentage errors across all available observations and computes the mean, resulting in a single, representative percentage value that effectively summarizes the overall model performance across the entire sample set.

This percentage-based approach is significantly more accessible and actionable for non-technical stakeholders compared to traditional absolute error measures. To illustrate, explaining that a model has an average absolute error of \$500 might be meaningless to a business executive without knowing the typical price range of the product being forecasted. However, stating that the model has a **MAPE** of 11.5% immediately communicates that the average forecasting error is 11.5% relative to the actual value. This provides immediate, actionable insight into the reliability and precision of the predictions. The simplicity, intuitive nature, and straightforward interpretation of

MAPE are key reasons why it remains a standard benchmark in many industries heavily reliant on [time series analysis](#) and predictive modeling efforts.

The Core Mathematical Formula for MAPE

The calculation of **MAPE** is rooted in a clear and structured mathematical formula specifically designed to quantify relative prediction error. This formula ensures that the error calculated is always positive and directly relative to the magnitude of the actual observation being predicted. This approach prevents large errors on small values from disproportionately dominating the metric, provided that the actual values are not zero. The explicit calculation involves several sequential steps, starting with the determination of the individual percentage error for each data point, followed by the aggregation of these errors across the entire dataset to compute the final mean percentage. This rigorous, structured approach guarantees a consistent and standardized measure of forecasting accuracy across diverse applications and data environments.

The universally accepted formula used for calculating the **Mean Absolute Percentage Error** is presented below:

$$\text{MAPE} = (1/n) * \Sigma(|\text{actual} - \text{prediction}| / |\text{actual}|) * 100$$

The essential components within this formula are defined as follows:

Σ - This symbol represents the mathematical operation of summation, indicating the requirement to sum all individual calculated percentage errors across the dataset.

n - This variable denotes the total sample size, corresponding to the number of data points being evaluated within the given dataset.

actual - This refers to the observed, true data value recorded at a specific time point or instance.

prediction - This represents the forecasted or predicted data value generated by the model corresponding to that same time point.

A crucial element of the calculation is the use of the absolute value function, denoted by the vertical bars (| |), which ensures that positive and negative prediction errors do not cancel each other out during the summation process, which would artificially deflate the error score. After calculating the absolute difference between the actual and predicted values, the result is divided by the absolute actual value, transforming the error into a ratio relative to the observation's scale. This ratio is subsequently multiplied by 100 to express the error as a percentage, which is finally averaged across the entire sample (divided by n) to yield the final, definitive **MAPE** score.

Implementing the MAPE Calculation Using Python and NumPy

Despite the fact that the **Mean Absolute Percentage Error** is a standard and highly desired

metric, many of the foundational machine learning libraries in [Python](#), such as Scikit-learn, do not include a dedicated, built-in function for its direct calculation. Consequently, data scientists often find it necessary to implement a custom function to compute MAPE accurately and efficiently within their analytical pipelines. This custom implementation typically leverages the powerful vectorized operations provided by the [NumPy](#) library. NumPy allows mathematical operations to be performed simultaneously on entire arrays, leading to significantly faster computation times compared to traditional Python loops, a necessity when handling the large datasets common in professional forecasting environments.

The core logic of the Python function involves a few critical steps: first, importing the NumPy library; second, defining the function signature to correctly accept two arrays (the actual values and the predicted values); third, converting these inputs into NumPy arrays to ensure compatibility with vectorized operations; and finally, applying the mathematical components of the **MAPE** formula and returning the final percentage result. By relying on NumPy's highly efficient array handling capabilities, we ensure that the function is not only mathematically sound but also computationally performant, making it perfectly suitable for seamless integration into complex data pipelines and automated model evaluation routines.

The following code snippet clearly demonstrates the creation of a clean, optimized function for calculating MAPE using the **NumPy** library. This implementation adheres directly to the precise mathematical definition of MAPE, guaranteeing accuracy and efficiency:

```
import numpy as np
```

```
def mape(actual, pred):  
    actual, pred = np.array(actual), np.array(pred)  
    return np.mean(np.abs((actual - pred) / actual)) * 100
```

Executing the Calculation with Sample Data

Once the custom mape function has been successfully defined, the next critical step is to apply it practically to real-world data arrays. This process allows us to quantitatively evaluate the performance of a specific forecasting model by comparing its generated output (the predicted values) directly against the actual historical observations. The inherent efficiency of the **NumPy** implementation shines throughout this process, as the function seamlessly handles the element-wise subtraction, division, absolute value calculation, and final averaging across all data points present in the provided arrays, requiring only a single line of execution code.

To demonstrate this functionality concretely, we define two simple Python lists: one list representing the true, observed values (`actual`) and a second list representing the corresponding

values generated by the predictive model (`pred`). These lists are subsequently passed directly to our defined `mape` function, which utilizes **NumPy** internally to execute the complex mathematical calculation and return the final percentage error score. This straightforward execution pattern ensures that the evaluation metric is easily reproducible, testable, and deployable across various data science projects and environments.

We can now execute the function using the sample data below to calculate the specific **mean absolute percentage error** for this set of predictions:

```
actual =
```

```
pred =
```

```
mape(actual, pred)
```

```
10.8009
```

Interpreting the MAPE Score and Business Application

The numerical output from the function execution, which is `10.8009`, represents the calculated **Mean Absolute Percentage Error** for the provided sample data arrays. This result must be interpreted carefully within the specific context of the business problem and the inherent characteristics of the data. A value of `10.8009` translates directly to a percentage error, meaning that the average difference between the model's predicted values and the actual observed values is approximately 10.8009% of the actual value. This level of granular detail provides immediate, quantitative feedback on the overall fidelity and reliability of the model's performance.

In practical business terms, if this model were being utilized to forecast monthly product demand, a **MAPE** of 10.8% suggests that business planning decisions--such as inventory stocking levels, material procurement, or resource allocation--based on these predictions will, on average, be off by about one-tenth of the true actual demand. Analysts frequently use this figure to establish crucial performance benchmarks; for example, a logistics company might mandate that all new demand forecasting models must achieve a MAPE below 8% before deployment. This makes MAPE an exceptional comparative metric, enabling decision-makers to quickly and intuitively gauge whether Model A (with a MAPE of 5%) is significantly superior to Model B (with a MAPE of 10.8%), without the need to analyze the underlying unit scales or data volatility.

Critical Limitations and Necessary Cautions When Using MAPE

Despite its widespread popularity and intuitive appeal, the **Mean Absolute Percentage Error** is not without significant structural drawbacks, and analysts must exercise considerable caution, particularly when applying it indiscriminately to certain types of datasets. Recognizing these

limitations is crucial for correctly selecting the most appropriate evaluation metric and avoiding potentially misleading or biased conclusions about overall model performance. Two major structural issues frequently necessitate the use of alternative metrics, such as the Symmetric MAPE (SMAPE) or the Mean Absolute Scaled Error (MASE).

The most immediate and critical limitation of MAPE stems directly from the division operation embedded within its formula. Because the formula requires division by the **absolute actual value** ($|actual|$), MAPE becomes mathematically undefined if any of the actual data values are precisely zero. In datasets where zero occurrences are common--such as intermittent demand [forecasting](#) for specialized spare parts, or in scenarios where products frequently run out of stock resulting in recorded zero sales--MAPE cannot be reliably calculated or interpreted. This fundamental "division by zero" problem severely restricts its appropriate use in environments characterized by sparse, highly volatile, or intermittent [time series analysis](#) data.

Furthermore, **MAPE** exhibits a known and inherent bias when applied to low-volume data or data points close to zero. Consider a forecasting scenario where the actual demand for an item is 2 units, and the model's forecast is 1 unit. The resulting absolute percentage error is calculated as $|2 - 1| / |2| = 50\%$. Although the forecast error in absolute terms is only one unit, the MAPE suggests an alarmingly high error rate of 50%. This disproportionate inflation of error for small actual values means that models performing exceptionally well on high-volume items might be severely penalized for minor absolute errors on low-volume items. This inherent bias can unfortunately skew model selection toward forecasts that systematically over-predict to avoid the steep penalties associated with under-prediction, highlighting the persistent need for careful scrutiny when applying **MAPE** to heterogeneous datasets containing a wide range of actual values.