

Normalize Data in SAS

Authored by
Mohammed looti

November 1, 2025

RECOMMENDED CITATION

Mohammed looti (2025). *Normalize Data in SAS*. PSYCHOLOGICAL STATISTICS.
Retrieved from <https://statistics.arabpsychology.com/?p=7619>

Transforming raw data values into a standardized format is a fundamental and often mandatory step in modern statistics and machine learning workflows. This procedure, frequently referred to as [feature scaling](#) or Z-score standardization, transforms the inherent distribution of a dataset. The goal is to ensure that the resulting standardized distribution achieves a statistical [mean](#) of zero (0) and a [standard deviation](#) of one (1).

Achieving this common scale is essential for ensuring robust and unbiased analytical results across various statistical methodologies. This comprehensive tutorial provides a precise, step-by-step guide on how to effectively perform this critical data standardization--specifically Z-score normalization--within the powerful [SAS](#) statistical software environment, leveraging the efficiency of the dedicated [PROC STDIZE](#) procedure.

The Crucial Rationale for Data Standardization

Data standardization is not merely an optional preprocessing step; it is a necessity when preparing data for complex analytical models, particularly those sensitive to variable scale and magnitude. By transforming raw scores into standardized measures, we guarantee that all independent variables contribute equitably to the subsequent analysis. This is particularly vital in algorithms reliant on distance calculations, such as K-means clustering, gradient descent optimization, or Principal Component Analysis (PCA).

When input features are measured on widely disparate scales--for instance, if one variable ranges from 0 to 1,000,000 while another ranges from 0 to 10--the variable with the larger magnitude can unfairly dominate the calculation of distance metrics, skewing the model's performance and interpretation. Standardization mitigates this risk by placing all features on a common, dimensionless scale.

The core benefit of this transformation is that it centers the data around the origin (mean = 0) while maintaining the original shape and relational differences within the distribution. The resulting transformed values, known as [Z-scores](#), instantaneously indicate how many standard deviations an individual observation lies above or below the central tendency of the dataset. This immediate interpretability is a powerful tool for analysts.

Illustrative Example: Normalizing Sample Data in SAS

To clearly demonstrate the mechanics of the standardization process, we will utilize a small, focused sample dataset. This dataset consists of 15 numeric observations. Our explicit objective is to transform these raw values into a new, standardized dataset where the calculated mean is exactly 0 and the standard deviation is exactly 1, adhering to the mathematical definition of a Z-score distribution.

We begin the process with the following original distribution of raw data values, which clearly lacks standardization:

Data
12
14
15
15
16
17
18
20
24
25
26
29
32
34
37

To efficiently achieve the desired standardization, we will follow a structured, three-step methodology using standard [SAS](#) programming commands. This approach ensures transparency, verification, and reproducible results.

Step 1: Defining the Dataset and Establishing Baseline Statistics

The initial and most critical step is the proper definition and input of the raw dataset into the [SAS](#) environment. Once the data is successfully loaded, we must establish the baseline descriptive statistics of the original data. This preliminary analysis is crucial for verifying the dataset's initial state before any transformation occurs, allowing us to confirm the initial [mean](#) and [standard deviation](#).

The following SAS code block first creates the dataset named `original_data`, populating it with the 15 numeric values. Subsequently, it executes the [PROC MEANS](#) procedure to calculate and display the initial metrics:

```
/*create dataset*/  
data original_data;  
input values;
```

```
datalines;  
12  
14  
15  
15  
16  
17  
18  
20  
24  
25  
26  
29  
32  
34  
37  
;  
run;  
  
/*view mean and standard deviation of dataset*/  
proc means data=original_data Mean StdDev ndec=3;  
var values;  
run;
```

The execution of the code above generates the initial output below, which clearly details the central tendency and dispersion of our raw data:

The MEANS Procedure		
Analysis Variable : values		
Mean	Std Dev	
22.267	7.968	

This output confirms that the initial [mean](#) of the raw dataset is **22.267**, and the original [standard deviation](#) is **7.968**. These two parameters are the foundational statistics that the standardization procedure will utilize to transform every single raw data point into its corresponding Z-score.

Step 2: Executing Standardization Using PROC STDIZE

The actual standardization transformation within the [SAS](#) system is managed by the highly efficient, dedicated procedure: [PROC STDIZE](#). This procedure automatically handles the complex computation of the Z-score for every observation, basing its calculations on the input dataset's established mean and standard deviation. Crucially, it saves the entirety of the resulting standardized values into a user-specified new dataset.

The syntax presented below instructs [PROC STDIZE](#) to operate on the existing `original_data` and directs the output of the standardized values into a new dataset named `normalized_data`. Following the standardization, we use PROC PRINT to display the transformed values and then execute PROC MEANS again to definitively verify the resulting descriptive statistics of the new dataset.

```
/*normalize the dataset*/  
proc stdize data=original_data out=normalized_data;  
var values;  
run;  
  
/*print normalized dataset*/  
proc print data=normalized_data;  
  
/*view mean and standard deviation of normalized dataset*/  
proc means data=normalized_data Mean StdDev ndec=2;  
var values;  
run;
```

After successful execution, SAS produces the normalized dataset. The output, shown below, lists the new standardized values alongside the original data and confirms the successful transformation of the distribution:

Obs	values
1	-1.28842
2	-1.03743
3	-0.91194
4	-0.91194
5	-0.78644
6	-0.66094
7	-0.53545
8	-0.28446
9	0.21753
10	0.34302
11	0.46852
12	0.84501
13	1.22149
14	1.47248
15	1.84897

The MEANS Procedure

Analysis Variable : values	
Mean	Std Dev
0.00	1.00

Crucially, the second PROC MEANS output, located at the bottom of the image, serves as the definitive confirmation of our success. It clearly illustrates that the calculated [mean](#) of the normalized_data is now precisely **0** and the [standard deviation](#) is exactly **1**. This outcome verifies that the data has been rigorously and mathematically standardized according to Z-score principles.

Step 3: Interpreting the Normalized Z-Scores

The true power and utility of standardization lie in the immediate interpretability afforded by the resulting Z-scores. Each normalized value (denoted as Z) represents the exact number of standard deviations the original data point (x) is located away from the original dataset's [mean](#) ($\bar{x}$). This fundamental relationship is universally defined by the following Z-score formula:

$$\text{Normalized value (Z)} = (x - \bar{x}) / s$$

The variables within this formula are defined as:

x = The individual raw observation value being transformed.

$\bar{x}$ = The calculated **mean** of the entire original dataset.

s = The calculated **standard deviation** of the entire original dataset.

Due to this transformation, every standardized value provides instant, relative insight into the position of that data point within the distribution. Standardized values that are positive indicate the observation is above the mean, while negative Z-scores indicate the observation is below the mean. Values close to zero represent observations typical of the average, while large absolute values (e.g., $|Z| > 2$ or 3) are likely to be statistical outliers.

Detailed Interpretation Walkthrough

To solidify this understanding, let us specifically examine the raw data point "12" from our input dataset. We confirmed in Step 1 that the original sample mean ($\bar{x}$) was 22.267 and the original sample standard deviation (s) was 7.968.

Applying the standardization formula to the raw value of 12 demonstrates the conversion:

Normalized value = $(x - \bar{x}) / s = (12 - 22.267) / 7.968 = \mathbf{-1.288}$

The resulting Z-score of -1.288 immediately informs us that the raw value of "12" is located **1.288 standard deviations below the mean** of the original dataset. Conversely, the highest raw value in our dataset, 37, standardizes to 1.849, signifying that it is 1.849 standard deviations above the mean. By using this standardized scale, an analyst can rapidly assess whether any given observation is a typical data point or an unusual deviation from the central tendency.

Advanced Considerations and Resources in SAS

Data normalization, specifically Z-score standardization, is a critical preprocessing requirement when preparing data for sophisticated statistical modeling or advanced machine learning algorithms within the [SAS](#) ecosystem. The use of standardized data is known to facilitate faster convergence for iterative algorithms and effectively eliminate the inherent bias that can be introduced by differences in variable measurement scales.

It is important to note that while [PROC STDIZE](#) defaults to the Z-score method (Mean=0, StdDev=1), the procedure is highly versatile. It offers analysts various other methods for scaling and centering data, including range standardization (min-max scaling) and median standardization. Analysts should carefully select the standardization method most appropriate for their specific statistical objectives and the inherent distributional characteristics of their data.

For individuals seeking to further expand their expertise in advanced SAS data manipulation and quantitative analysis techniques, the following related resources offer detailed explanations on

other frequently encountered data preparation tasks:

Methodologies for robustly handling missing values in SAS datasets.

Advanced techniques for data merging and appending operations in SAS.

Practical application of PROC GLM for complex linear modeling and analysis of variance.