

Learning Normality Tests in SAS with PROC UNIVARIATE

Authored by
Mohammed looti

May 8, 2026

RECOMMENDED CITATION

Mohammed looti (2026). *Learning Normality Tests in SAS with PROC UNIVARIATE*.
PSYCHOLOGICAL STATISTICS. Retrieved from
<https://statistics.arabpsychology.com/?p=3562>

Introduction to Normality Testing in SAS

Understanding the underlying distribution of your data is not merely a statistical formality but a fundamental prerequisite for ensuring the validity of many advanced analyses. A significant number of widely used inferential statistical procedures--including t-tests, ANOVA, and linear regression--are mathematically predicated on the crucial assumption that the data being analyzed adheres to a [normal distribution](#). If this core assumption is violated, the results derived from these parametric tests may be unreliable, potentially leading to flawed conclusions about your research hypotheses. Therefore, it is absolutely essential for every rigorous data analyst to accurately assess and confirm data normality before proceeding with these powerful, assumption-based tests.

Fortunately, [SAS](#), renowned as one of the most powerful and comprehensive statistical software suites available, offers robust and efficient tools specifically designed for this purpose. Among its extensive library of procedures, one of the most versatile for distributional assessment is [PROC UNIVARIATE](#). When this procedure is combined with the specialized **NORMAL** statement, SAS automatically executes a comprehensive battery of normality tests on a specified numeric variable within your dataset. This capability allows researchers and data scientists to formally evaluate whether their empirical data aligns satisfactorily with the theoretical expectations of the normal distribution assumption.

This expert guide is designed to provide a clear, step-by-step walkthrough detailing exactly how to leverage [PROC UNIVARIATE](#) for assessing data normality in [SAS](#). We will delve into the essential syntax required, demonstrate its practical application using a relatable example dataset, and meticulously explain how to interpret the results of the various statistical tests provided. By the end of this tutorial, you will be equipped to confidently determine the distributional characteristics of your data and make informed choices regarding subsequent statistical modeling.

The Power of PROC UNIVARIATE and the NORMAL Statement

The [PROC UNIVARIATE](#) procedure within [SAS](#) serves as a cornerstone for descriptive statistics, extending far beyond the provision of simple summary measures like the mean or standard deviation. It facilitates an in-depth, exploratory analysis of a single numeric variable, calculating moments, quantiles, and identifying extreme values (outliers). Crucially for assumption checking, it features advanced options tailored for examining data distributions, making it an indispensable tool for preliminary data analysis. When the specialized **NORMAL** statement is incorporated into your [PROC UNIVARIATE](#) step, SAS automatically calculates and presents the outcomes of several widely recognized normality tests, streamlining the process significantly.

This powerful procedural combination greatly simplifies the task of statistically assessing whether your dataset deviates significantly from a normal distribution. The procedure is intentionally designed to be highly user-friendly, requiring only a minimal number of code lines to generate a

comprehensive report detailing your variable's distribution characteristics. This accessibility ensures that both novice users learning statistical software and experienced SAS professionals can effectively perform these vital preliminary checks with speed and accuracy.

The fundamental and most common syntax required for executing [PROC UNIVARIATE](#) alongside the **NORMAL** statement is structured as follows, illustrating the clarity and conciseness of the SAS programming language:

```
proc univariate data=my_data normal;  
var my_variable;  
run;
```

To ensure clarity, let us break down each essential component of this syntax block:

PROC UNIVARIATE: This initial statement serves to invoke and initiate the univariate analysis procedure within SAS.

data=my_data: This required option specifies the name of the input dataset that holds the numeric variable you intend to analyze. It is crucial to replace the placeholder `my_data` with the actual, registered name of your working dataset.

normal: This is the critical option statement that instructs SAS not only to perform the standard univariate calculations but also to execute and display the results for a suite of normality tests, including the widely used Shapiro-Wilk, Kolmogorov-Smirnov, Cramer-von Mises, and Anderson-Darling tests.

var my_variable: The `VAR` statement is used to designate the specific numeric variable within your specified dataset for which you want to calculate descriptive statistics and assess normality. Always replace `my_variable` with the actual column name.

run;: This concluding statement signals SAS to execute the entire preceding `PROC UNIVARIATE` block and produce the requested output reports.

Setting Up the Sample Data for Practical Analysis

To effectively demonstrate the practical application of [PROC UNIVARIATE](#) for checking data normality, we will utilize a hypothetical sample dataset. This dataset, which we will conveniently create directly within the [SAS](#) environment using the `DATA` step, contains essential information about various fictional basketball players. Specifically, the data includes categorical details such as their team affiliation (`team`), the number of points scored in a recent game (`points`), and the number of rebounds achieved (`rebounds`).

For the primary focus of this specific demonstration, our analytical interest centers solely on the `points` variable. Our objective is to rigorously assess whether the distribution of points scored by these players conforms closely to the theoretical structure of a [normal distribution](#). This dataset provides a clear, concise, and relatable example that perfectly illustrates the necessary steps involved in performing robust normality checks that are applicable across many disciplines.

The following SAS code block first creates this sample dataset, naming it `my_data`, and subsequently uses a `PROC PRINT` statement to display its contents. This allows for immediate visual verification of the data structure prior to initiating the formal statistical analysis:

```
/*create dataset*/  
data my_data;  
input team $ points rebounds;  
datalines;  
A 12 8  
A 12 8  
A 12 8  
A 23 9  
A 20 12  
A 14 7  
A 14 7  
B 20 2  
B 20 5  
B 29 4  
B 14 7  
B 20 2  
B 20 2  
B 20 5  
;  
run;  
  
/*view dataset*/  
proc print data=my_data;
```

Upon successful execution of this code block, SAS will generate and display the `my_data` dataset in your output window. This vital preliminary step allows you to visually inspect the data structure, confirm that the variables are correctly loaded, and ensure data integrity before proceeding with the core normality assessment.

Obs	team	points	rebounds
1	A	12	8
2	A	12	8
3	A	12	8
4	A	23	9
5	A	20	12
6	A	14	7
7	A	14	7
8	B	20	2
9	B	20	5
10	B	29	4
11	B	14	7
12	B	20	2
13	B	20	2
14	B	20	5

Executing Normality Tests with PROC UNIVARIATE

With our sample dataset, `my_data`, successfully created, reviewed, and ready for analysis, the next logical step involves executing the formal normality tests. As established, this is accomplished by invoking the powerful [PROC UNIVARIATE](#) procedure and adding the essential **NORMAL** statement. For this specific demonstration, we will focus our entire analysis exclusively on the **points** variable, which holds the game scores achieved by each basketball player listed in the data.

The overarching goal of this execution is to statistically and objectively determine if the observed distribution of these points scores deviates significantly from the expectations of a [normal distribution](#). This formal statistical assessment provides crucial objective evidence that either corroborates or challenges any initial visual inspection of the data distribution, forming a robust basis for subsequent statistical modeling decisions.

The SAS code required to perform these comprehensive tests is remarkably concise, directly applying the syntax template we previously outlined:

```
proc univariate data=my_data normal;  
var points;  
run;
```

Upon executing this simple code block, [SAS](#) will generate a detailed and extensive output. This output typically includes several tables covering descriptive statistics, moments (like skewness and kurtosis), and quantiles for the specified variable. However, for the specific objective of assessing normality, our primary and most focused attention must be directed towards the table explicitly labeled **Tests for Normality**. This section conveniently consolidates the results from all the performed statistical normality tests, presenting the necessary test statistics and p-values for decision-making.

Tests for Normality				
Test	Statistic		p Value	
Shapiro-Wilk	W	0.867166	Pr < W	0.0383
Kolmogorov-Smirnov	D	0.23714	Pr > D	0.0318
Cramer-von Mises	W-Sq	0.152465	Pr > W-Sq	0.0200
Anderson-Darling	A-Sq	0.847207	Pr > A-Sq	0.0223

Interpreting the Formal Normality Test Results

When you incorporate the powerful **NORMAL** statement within [PROC UNIVARIATE](#), SAS automatically performs and presents the results for four of the most commonly accepted and robust normality tests. These tests are the [Shapiro-Wilk Test](#) (often preferred for smaller sample sizes), the [Kolmogorov-Smirnov Test](#), the [Cramer-von Mises Test](#), and the [Anderson-Darling Test](#). Each test provides a unique [test statistic](#) and a corresponding [p-value](#), both of which are critical for interpreting whether your data successfully adheres to a normal distribution. The output generated from our specific basketball **points** variable example yields the following numerical results:

[Shapiro-Wilk Test](#): $W = .867, p = .0383$

[Kolmogorov-Smirnov Test](#): $D = .237, p = .0318$

[Cramer-von Mises Test](#): $W-Sq = .152, p = .0200$

[Anderson-Darling Test](#): $A-Sq = .847, p = .0223$

For every one of these statistical normality tests, the fundamental statistical hypotheses are consistently formulated within the framework of null hypothesis significance testing (NHST):

[H₀](#) (Null Hypothesis): The data are normally distributed.

[H_A](#) (Alternative Hypothesis): The data are not normally distributed.

To confidently make a decision regarding the [null hypothesis](#), we must compare the reported [p-value](#) for each test against a predetermined significance level (alpha level), which is conventionally set at 0.05 (or 5%). The decision rule is straightforward: If a [p-value](#) is less than the chosen alpha level (i.e., $p < 0.05$), it indicates that there is statistically significant evidence to reject the [null hypothesis](#), leading us to conclude that the data are unlikely to have been drawn from a normal distribution. Conversely, if the p-value is greater than or equal to the alpha level ($p \geq 0.05$), we fail to reject the [null hypothesis](#), suggesting that the data are consistent with the assumption of normality.

In our specific example, the p-values for all four normality tests are consistently low: .0383 (Shapiro-Wilk), .0318 (Kolmogorov-Smirnov), .0200 (Cramer-von Mises), and .0223 (Anderson-Darling). Since every single one of these p-values is less than our threshold significance level of .05, we must decisively reject the [null hypothesis](#) for all tests. This strong, consistent outcome provides compelling statistical evidence that the **points** variable in our basketball player dataset is definitively **not normally distributed**. This critical finding serves as a warning sign, immediately informing all subsequent analytical decisions and potentially necessitating the use of non-parametric statistical methods or appropriate data transformations before parametric tests can be reliably applied.

Visualizing Data Distribution with a Histogram

While formal normality tests provide quantitative statistical proof regarding the distribution of your data, it is considered a best practice in data analysis to always complement these numerical results with a comprehensive visual assessment. A [histogram](#) is an exceptionally effective and intuitive graphical tool specifically designed for illustrating the frequency distribution of a numeric variable. By instructing SAS to superimpose a theoretical [normal curve](#) onto the generated [histogram](#) of your variable, you gain a powerful visual gauge of how closely the shape of your observed data aligns with the expected symmetrical bell curve characteristic of a normal distribution.

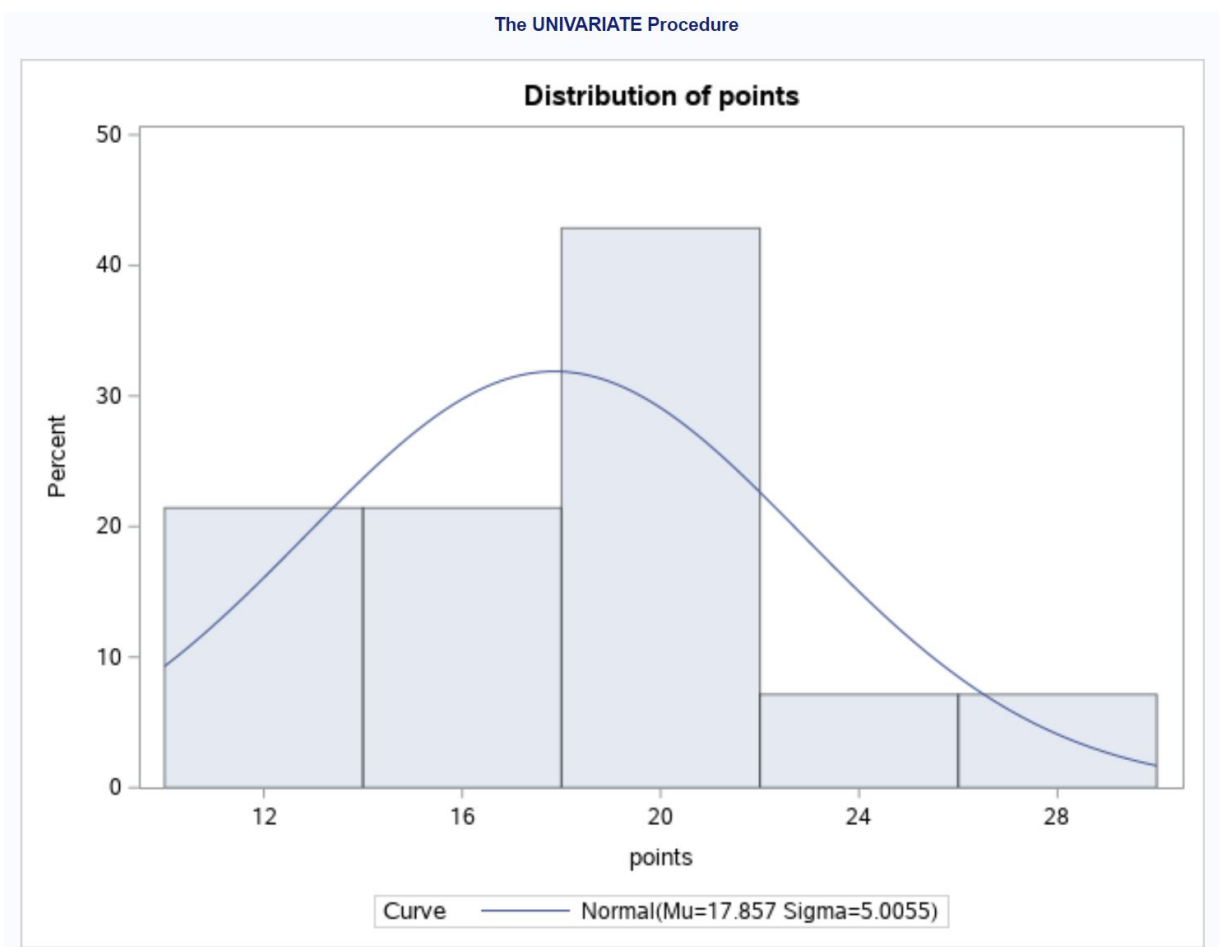
This visual comparison offers an immediate, intuitive understanding that can powerfully reinforce--or, in certain complex cases, provide necessary context to--the findings from the statistical tests, particularly when dealing with small sample sizes where statistical power may be limited or when deviations from normality are subtle. It allows the analyst to quickly identify visual hallmarks of non-normality, such as pronounced skewness (asymmetry), high or low kurtosis (peakedness or flatness), or the presence of multiple modes, which are critical indicators that the normality assumption has been violated.

To generate a [histogram](#) for our non-normal **points** variable, complete with an overlaid fitted normal curve in [SAS](#), you can utilize the following variation of the [PROC UNIVARIATE](#) syntax,

utilizing the `HISTOGRAM` statement:

```
proc univariate data=my_data;  
histogram points / normal;  
run;
```

In this revised code, the `histogram` statement within `PROC UNIVARIATE` is specifically used to request the graphical representation, and the critical `/ normal` option is appended to the statement to request the overlay of a fitted [normal curve](#) based on the data's mean and standard deviation, thereby facilitating a direct and immediate visual comparison.



Upon careful examination of the resulting [histogram](#), it becomes immediately clear and visually apparent that the distribution of the **points** variable does not closely align with the smooth, symmetrical, and bell-shaped curve of the overlaid normal distribution. There are noticeable irregularities, gaps, and significant deviations from the expected shape, which visually confirm that the data is not normally distributed. This powerful visual evidence serves to strongly corroborate the definitive conclusions we drew from our formal statistical normality tests, ensuring a

comprehensive, robust, and easily digestible assessment of the data's true distribution.

Conclusion: Ensuring Validity in Statistical Modeling

In this comprehensive guide, we have thoroughly detailed the essential process of utilizing [SAS's PROC UNIVARIATE](#) procedure, combined with the powerful **NORMAL** statement, to perform critical normality tests. We meticulously walked through every step, from the initial creation of a sample dataset and the execution of the necessary SAS code, to the nuanced interpretation of the results obtained from the [Shapiro-Wilk](#), [Kolmogorov-Smirnov](#), [Cramer-von Mises](#), and [Anderson-Darling tests](#). Furthermore, we underscored the importance of visually assessing data distribution by generating and interpreting a [histogram](#) that includes an overlaid [normal curve](#), providing a holistic view of the data characteristics.

The ability to reliably test for the assumption of a normal distribution is an absolutely indispensable skill set for any statistician or data analyst working with parametric models. This critical checking process ensures that the core assumptions underlying many subsequent statistical analyses are adequately met, thereby fundamentally safeguarding the validity, integrity, and trustworthiness of your derived research findings and final conclusions. When your data is found to be non-normal, this crucial knowledge empowers you to proactively consider and implement appropriate alternative analytical strategies, such as transitioning to non-parametric methods or applying suitable data transformations (e.g., log, square root) to ensure the continued robustness and reliability of your statistical inferences.

Additional Resources for SAS Proficiency

For those aspiring to deepen their technical understanding of [SAS](#) programming and broaden their statistical analysis capabilities beyond normality testing, the following tutorials offer further guidance on performing other common data preparation and analysis tasks, as well as exploring related statistical concepts: