

# Learning to Visualize Data: A Step-by-Step Guide to Plotting Means with Standard Error Bars in SAS

Authored by  
**Mohammed looti**

November 14, 2025

## RECOMMENDED CITATION

Mohammed looti (2025). *Learning to Visualize Data: A Step-by-Step Guide to Plotting Means with Standard Error Bars in SAS*. PSYCHOLOGICAL STATISTICS. Retrieved from <https://statistics.arabpsychology.com/?p=1612>

## Introduction: Visualizing Data Precision with Standard Error Bars in SAS

In the realm of [statistical analysis](#), conveying not just the central tendency but also the reliability of estimates is absolutely critical. Although the [mean](#) provides a straightforward measure of location, reporting this value alone can be deceptive, especially when researchers are comparing outcomes across different experimental or categorical groups. To offer a truthful representation of data precision, analysts employ [standard error bars](#). These graphical components serve as essential visual anchors, quantifying the uncertainty surrounding the estimated mean and allowing readers to make highly informed, accurate comparisons.

This expert guide provides a structured methodology for [SAS](#) users focused on generating high-fidelity, publication-ready plots that showcase mean values along with their calculated standard error ranges. Our approach relies on the seamless integration of two indispensable SAS procedures: [PROC SQL](#), used for efficient data aggregation and precise calculation of error boundaries, and [PROC SGPLOT](#), which handles the creation of sophisticated graphical output. Mastering the synergy between these two powerful tools is key to transforming raw, complex data into robust, easily interpretable [data visualization](#).

We begin by presenting an integrated code snippet that utilizes both standard [SQL](#) syntax for the required statistical calculations and native SAS commands for plotting. This foundational blueprint offers a conceptual overview of the entire workflow before we delve into a detailed, practical case study using simulated data.

```
/*calculate mean and standard error of points for each team*/
```

```
proc sql;  
create table groupPlot as  
select  
team,  
mean(points) as meanPoints,  
mean(points) - stderr(points) as lowStdPoints,  
mean(points) + stderr(points) as highStdPoints  
from my_data  
group by team;  
quit;
```

```
/*create plot with mean and standard error bars of points for each team*/
```

```
proc sgplot data=groupPlot;  
scatter x=team y=meanPoints /  
yerrorlower=lowStdPoints yerrorupper=highStdPoints group=team;  
series x=team y=meanPoints / group=team;  
run;
```

## Data Aggregation and Calculation Using PROC SQL

The foundation of any accurate error bar plot rests on the preliminary calculation of precise [descriptive statistics](#). Within the powerful [SAS](#) programming environment, the [PROC SQL](#) procedure stands out as an exceptionally flexible and efficient tool for data manipulation and group-wise aggregation. Its fundamental structure is derived from the widely adopted standard [SQL](#) query language. This familiarity allows analysts already skilled in database operations to smoothly create, modify, and query SAS [datasets](#), making **PROC SQL** the ideal choice for preparing complex data structures required for advanced visualizations.

To achieve our visualization objective--plotting group means with error bars--we specifically utilize **PROC SQL** to simultaneously calculate the [mean](#) value and the associated [standard error](#) for a continuous measurement variable, such as **points**. These calculations must be executed independently across distinct categories, which are defined here by the **team** variable. This group-specific approach is crucial because the standard error is inherently tied to the sample size and variability unique to each segment of the data. The final output of this aggregation process is a new, condensed table, which we name **groupPlot**, containing the precise coordinate data necessary for the subsequent graphical plotting stage.

The core of the **PROC SQL** block is built around the **SELECT** statement, which dictates exactly which statistical metrics are generated. We must include the grouping variable (**team**), the calculated mean (aliased as **meanPoints**), and, most importantly, the upper and lower boundaries that define the error bars. These boundary variables--**lowStdPoints** and **highStdPoints**--are derived using simple arithmetic: subtracting the standard error (calculated via the built-in **stderr** function) from the mean for the lower bound, and adding it for the upper bound. The inclusion of the **GROUP BY** clause is mandatory to ensure these statistical functions are applied discretely and accurately to every unique team category.

## Generating Publication-Quality Plots with PROC SGPLOT

Following the successful computational step performed by [PROC SQL](#)--which yielded the means and their associated standard error bounds--we transition to the visual rendering phase. This plotting task is expertly handled by [PROC SGPLOT](#), the premier graphics procedure within the [SAS](#) system, designed specifically for generating high-impact statistical visuals. **PROC SGPLOT** employs an intuitive, declarative syntax: users simply declare the required plot elements (e.g., scatter, bar, series), and the software manages the complex details of rendering, enabling the rapid creation of publication-quality charts with minimal code overhead.

The primary directive for plotting the [mean](#) values accompanied by their [standard error bars](#) is the **scatter** statement. This command maps the central mean estimate (stored in **meanPoints**) against

the categorical variable on the X-axis (**team**). The crucial functionality that adds the error bars is driven by two specific options: **yerrorlower** and **yerrorupper**. These options instruct **PROC SGPLOT** to use the pre-calculated **lowStdPoints** and **highStdPoints** variables, respectively, thereby defining the precise vertical span of the confidence interval around each plotted mean point. This direct dependency ensures strict alignment between the statistical output from **PROC SQL** and the final visual representation.

While the **scatter** statement is sufficient for displaying the points and error bars, we frequently include the **series** statement as well. This addition connects the mean points across the categories, which can significantly enhance the visual narrative, especially when the X-axis variables imply an order or when the goal is to emphasize a trend or progression. By combining the **scatter** and **series** commands within **PROC SGPLOT**, we achieve a clear, composite graphic that effectively communicates both the central tendency of the data and the precision associated with its measurement.

## Case Study: Visualizing Basketball Scoring Performance

To solidify our understanding of this two-step process, we will now implement the procedure using a practical example based on a hypothetical [dataset](#) of basketball scoring performance. This sample data tracks the individual points scored by players, categorized into three distinct teams (A, B, and C). Our goal is analytical: we aim to produce a clear visualization that not only compares the average points achieved by each team but also effectively communicates the internal scoring consistency, or variability, using standard error bars. Such detailed comparative analysis is foundational for effective performance review and strategic resource allocation in sports management.

The first operational phase requires the creation of the sample data structure, which we name **my\_data**, within the [SAS](#) environment. The following **data** step explicitly defines the required variables (the categorical **team** and the continuous **points**) and populates the resulting table using simulated score data via the **datalines** statement. Immediately following the data creation, we utilize a **proc print** command. This quick visual inspection is an essential quality assurance step, confirming that the data has been loaded correctly and is structured appropriately before proceeding to the computationally intensive aggregation stage.

```
/*create dataset*/  
data my_data;  
input team $ points;  
datalines;  
A 29  
A 23
```

```
A 20
A 21
A 33
B 14
B 13
B 17
B 14
B 15
C 21
C 22
C 20
C 25
C 24
;
run;

/*view dataset*/
proc print data=my_data;
```

The accompanying image verifies the successful initialization and population of the **my\_data** table. This raw, organized data--comprising team identifiers and corresponding individual scores--constitutes the essential input source required for the subsequent calculation of aggregated [descriptive statistics](#).

Obs	team	points
1	A	29
2	A	23
3	A	20
4	A	21
5	A	33
6	B	14
7	B	13
8	B	17
9	B	14
10	B	15
11	C	21
12	C	22
13	C	20
14	C	25
15	C	24

With the integrity of the source data confirmed, we proceed to execute the core statistical and graphing instructions. By reapplying the integrated code utilizing [PROC SQL](#) for aggregation and [PROC SGPLOT](#) for visualization, we instruct SAS to perform all necessary statistical calculations--determining team means and standard error bounds--and immediately render the resulting graphical output.

```
/*calculate mean and standard error of points for each team*/
```

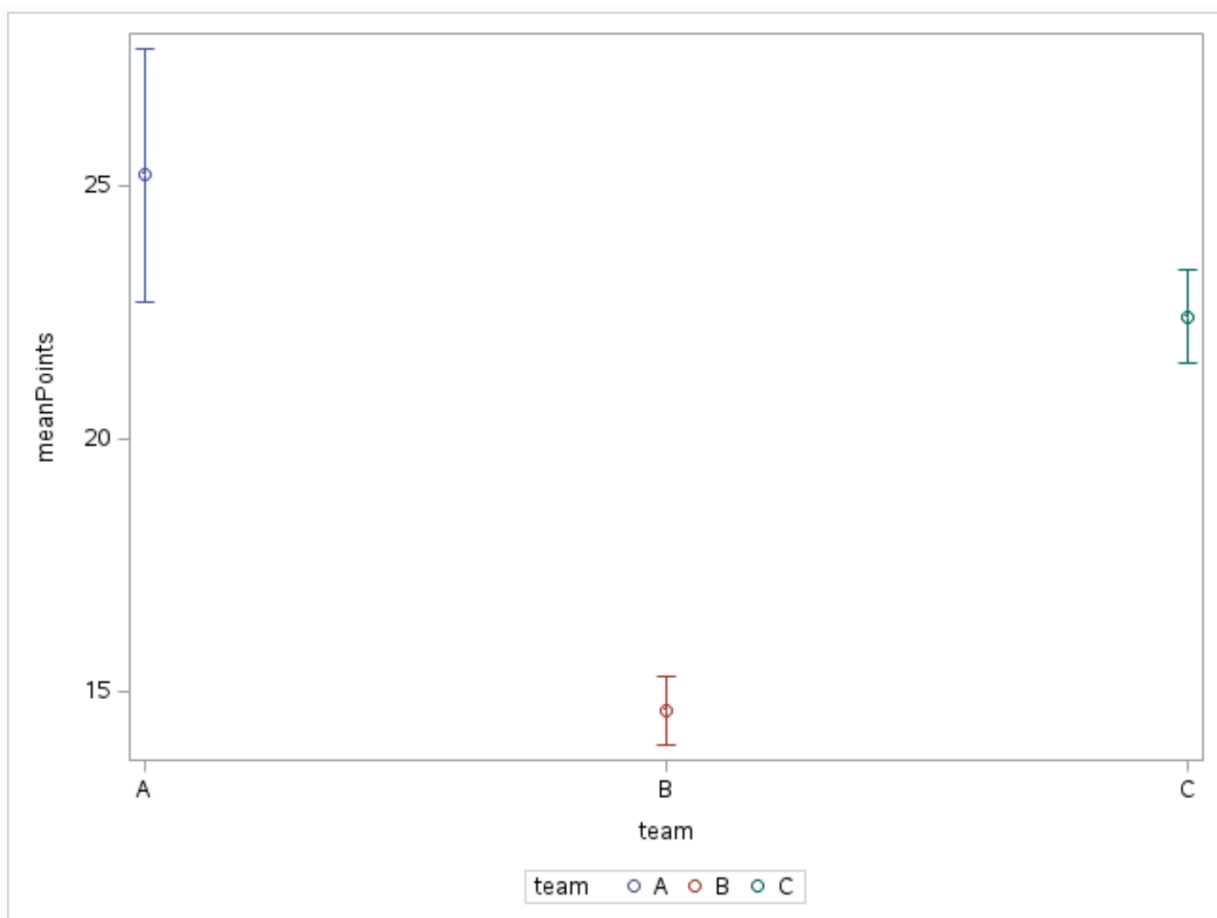
```
proc sql;  
create table groupPlot as  
select  
team,  
mean(points) as meanPoints,  
mean(points) - stderr(points) as lowStdPoints,  
mean(points) + stderr(points) as highStdPoints  
from my_data  
group by team;  
quit;
```

```
/*create plot with mean and standard error bars of points for each team*/
```

```
proc sgplot data=groupPlot;  
scatter x=team y=meanPoints /
```

```
yerrorlower=lowStdPoints yerrorupper=highStdPoints group=team;  
series x=team y=meanPoints / group=team;  
run;
```

The resulting plot, displayed below, offers a powerful visual summary of the teams' scoring performance. Each marker (circle) precisely locates the estimated [mean](#) score for a team, while the vertical [standard error bars](#) define the range of statistical uncertainty around that estimate. For interpretation, note that a significantly shorter error bar indicates greater statistical precision in the mean estimate for that team, implying lower volatility or higher consistency in individual player scores. Conversely, longer bars suggest greater variability within the team's data.



## Verifying Numerical Accuracy with PROC PRINT

While the graphical output produced by [PROC SGPLOT](#) offers an immediate and impactful summary, rigorous adherence to best practice in [statistical analysis](#) dictates that the underlying numerical computations must always be verified. The intermediate table, **groupPlot**, which was meticulously constructed by [PROC SQL](#), contains the precise [descriptive statistics](#) used to render

the visualization. Inspecting this table is essential for analysts to confirm computational accuracy and to extract exact numerical values needed for detailed reporting or academic submission.

To review the contents of this crucial intermediate [dataset](#), we use the straightforward **proc print** statement within [SAS](#). This procedure outputs the observations of the aggregated data table in a clean, easily readable tabular format. The resulting output clearly delineates the grouping variable (**team**), the computed central measure (**meanPoints**), and the calculated lower (**lowStdPoints**) and upper (**highStdPoints**) bounds of the standard error range for each group.

```
/*print mean and standard error of points for each team*/  
proc print data=groupPlot;
```

As shown in the output image below, the numerical data confirms the plot's visual representation. For instance, Team A's mean score is 25.2, and its standard error range spans from 21.01 points up to 29.39 points. This numerical confirmation is vital for ensuring the credibility of the final [data visualization](#) and provides the necessary detail for robust reporting.

Obs	team	meanPoints	lowStdPoints	highStdPoints
1	A	25.2	22.7020	27.6980
2	B	14.6	13.9218	15.2782
3	C	22.4	21.4726	23.3274

## Conclusion: Mastering Visual Interpretation in SAS

Effective [data visualization](#) remains the critical bridge for communicating complex findings in [statistical analysis](#). By successfully integrating the robust data handling capabilities of [PROC SQL](#) with the sophisticated rendering power of [PROC SGPLOT](#), SAS users are equipped to generate highly informative plots of group means that accurately incorporate [standard error bars](#). These composite graphics are indispensable analytical tools, allowing simultaneous communication of the central location of the data (the [mean](#)) and the statistical precision with which that estimated location was derived.

The systematic workflow detailed throughout this tutorial offers a clear, scalable, and universally applicable methodology for processing complex [datasets](#) across diverse research fields. By first calculating the necessary statistical aggregates using standard [SQL](#) syntax, and then utilizing the dynamic plotting options within SGPLOT, analysts effectively move beyond superficial point estimates. A crucial skill developed here is the interpretation of the error bar length: shorter bars consistently signal greater precision in the mean estimate, whereas overlapping error bars

between two groups strongly suggest that the observed difference between their means may not achieve statistical significance.

Proficiency in these fundamental [SAS](#) procedures grants analysts the authority to present data with necessary nuance and confidence. Whether the goal is to support rigorous academic research or to inform high-stakes business decisions, the clear, side-by-side presentation of data variability and central tendency significantly enhances the reliability and trustworthiness of all reported findings. We strongly encourage users to explore the extensive array of customization options available within **PROC SGPLOT** to tailor their visualizations fully to meet any specific analytical or aesthetic requirements.

## Further Resources for SAS Programming

To ensure continuous skill development and mastery in data handling and graphical representation, we highly recommend extending your expertise in advanced [SQL](#) aggregation and sophisticated [data visualization](#) techniques within the SAS environment. The following resources are invaluable for expanding your capabilities far beyond basic mean plots to include a comprehensive suite of analytical methods.

**SAS Official Documentation:** Always prioritize consulting the authoritative guides for the most detailed information on statements, options, and usage examples for core procedures like **PROC SQL** and **PROC SGPLOT**.

**Online SAS User Communities:** Actively engage with established user communities and forums. These platforms are excellent sources for finding practical solutions, sharing innovative programming tips, and collaborating on complex, advanced analytical challenges.

**Advanced Visualization Tutorials:** Seek out specialized tutorials that investigate methods for creating more complex and nuanced charts in SAS, such as comparative box plots, overlaid histograms, and the development of custom graph templates for specific reporting needs.