

How to create pie chart in SAS?

Authored by
stats writer

November 19, 2025

RECOMMENDED CITATION

stats writer (2025). *How to create pie chart in SAS?*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=96684>

Creating effective visualizations is a cornerstone of statistical analysis and reporting. Within the SAS System, specifically utilizing the PROC GCHART procedure, users can generate high-quality pie charts. These circular graphs are essential tools for illustrating proportions, showing how individual components contribute to an overall total. The core command for initiating this visualization is the **PIE** statement, a powerful feature within the PROC GCHART environment designed for graphical output. This comprehensive guide details the necessary steps and options for customizing your pie charts in SAS, moving beyond basic frequency plots to advanced displays involving summaries and custom styling.

Understanding the versatility of the **PIE** statement allows analysts to represent categorical data effectively, whether summarizing raw counts or aggregating numerical variables associated with specific categories. The subsequent sections will demonstrate four distinct methods for generating these charts, ranging from simple frequency counts to complex visualizations featuring exploded slices and tailored labels. Before diving into the graphical procedures, we must first establish a foundational dataset that will serve as the source data for all subsequent examples. This structured approach ensures clarity and reproducibility across all demonstrated techniques.

Introduction to Pie Charts in SAS/GRAPH

The pie chart remains a widely recognized method for displaying the distribution of a variable across several categories. In the context of SAS, these charts are typically generated using the PROC GCHART procedure, which is part of the SAS/GRAPH module. While newer visualization tools exist in SAS, PROC GCHART is robust, widely supported, and provides extensive customization options necessary for detailed reporting. The fundamental idea behind using the **PIE** statement is to specify a classification variable; PROC GCHART then automatically calculates the proportion of observations belonging to each level of that variable.

A key advantage of PROC GCHART is its ability to handle both frequency data and summarized data. When displaying frequency, each slice's area is proportional to the count of observations in that category. When summary data is required--such as the total sales or accumulated points per category--the **SUMVAR** option must be introduced. This flexibility allows analysts to quickly generate diverse visual representations without switching between multiple procedures. Furthermore, options like **EXPLODE** and **PLABEL** demonstrate SAS/GRAPH's capability to tailor the visual output for emphasis and enhanced communication, ensuring that the chart effectively highlights the most critical insights derived from the dataset.

Preparing the Data for Visualization (Dataset Creation and Review)

To illustrate the various pie chart functionalities, we will utilize a sample dataset focusing on basketball player statistics, specifically tracking team affiliation and points scored in various games.

This simple structure--a categorical variable (team) and a numerical variable (points)--provides the perfect foundation to demonstrate both frequency counts and variable summarization. The creation of this dataset, named **my_data**, is performed using standard SAS data step procedures, ensuring the data is properly formatted before visualization begins.

The following code block initiates the data step, inputs the relevant observation data, and then uses **PROC PRINT** to display the resulting table structure. Reviewing the dataset contents is a crucial preliminary step, allowing us to confirm that the classification variable (team) and the quantitative variable (points) are correctly read and ready for graphical processing by PROC GCHART. Note the mix of teams and the varying number of records per team, which will influence the final slice proportions in the charts.

```
/*create dataset*/  
data my_data;  
input team $ points;  
datalines;  
Mavs 14  
Mavs 22  
Mavs 19  
Mavs 31  
Heat 14  
Heat 25  
Warriors 31  
Warriors 35  
Warriors 36  
Jazz 29  
;  
run;  
  
/*view dataset*/  
proc print data=my_data;
```

The output of the **PROC PRINT** command confirms that our variables are defined correctly: the **team** variable is character type (\$), and the **points** variable is numerical, establishing the foundation for our visualizations.

Obs	team	points
1	Mavs	14
2	Mavs	22
3	Mavs	19
4	Mavs	31
5	Heat	14
6	Heat	25
7	Warriors	31
8	Warriors	35
9	Warriors	36
10	Jazz	29

Example 1: Visualizing Data Frequency using PIE

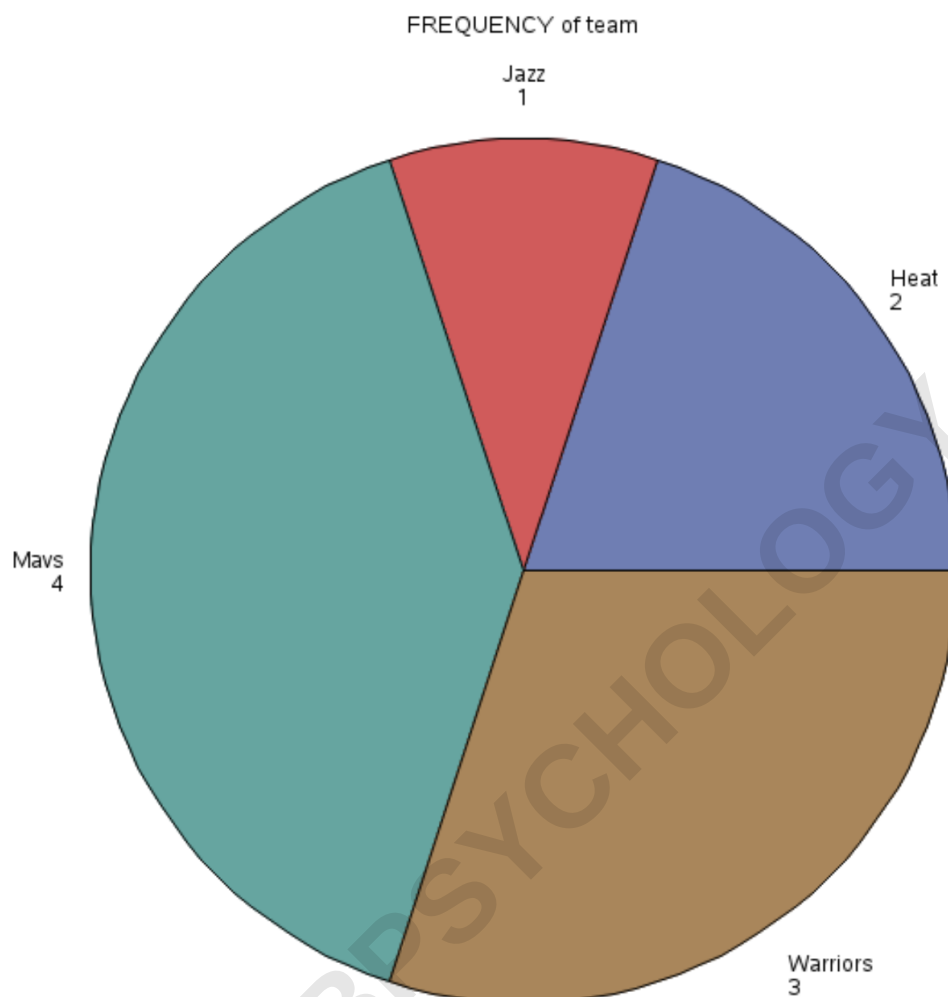
The most straightforward application of the **PIE** statement is to visualize the frequency, or count, of observations for each unique category. In this scenario, we are interested in knowing how many records (player performances) exist for each team within our **my_data** dataset. By default, when only the classification variable is supplied to the **PIE** statement, **PROC GCHART** calculates the relative frequency of each category and sizes the slices accordingly. This is an excellent way to quickly assess the distribution or balance of categorical variables.

The code below is minimal yet powerful. We invoke **PROC GCHART**, specify our data source, and simply use **PIE team;**. This tells **SAS** to group the observations by the **team** variable and plot the resulting counts. Notice the use of **RUN** to execute the procedure and **QUIT** to exit the SAS/GRAPH environment, a best practice when working with graphical procedures to ensure resource release.

```
proc gchart data=my_data;  
pie team;  
run;  
quit;
```

The resulting pie chart visually confirms the underlying distribution. We can observe that the Warriors have the largest slice (3 records), followed closely by the Mavs (4 records). Wait, upon closer inspection of the data: Mavs (4), Heat (2), Warriors (3), Jazz (1). The chart accurately represents these counts, with the Mavs holding the largest proportion of observed records. The

slices in this chart are sized directly proportional to the number of times each unique team name appears in the **team** column, providing an immediate visual comparison of the sample sizes for each team performance entry.



Example 2: Aggregating Data with the SUMVAR Option (Pie Chart of Sums)

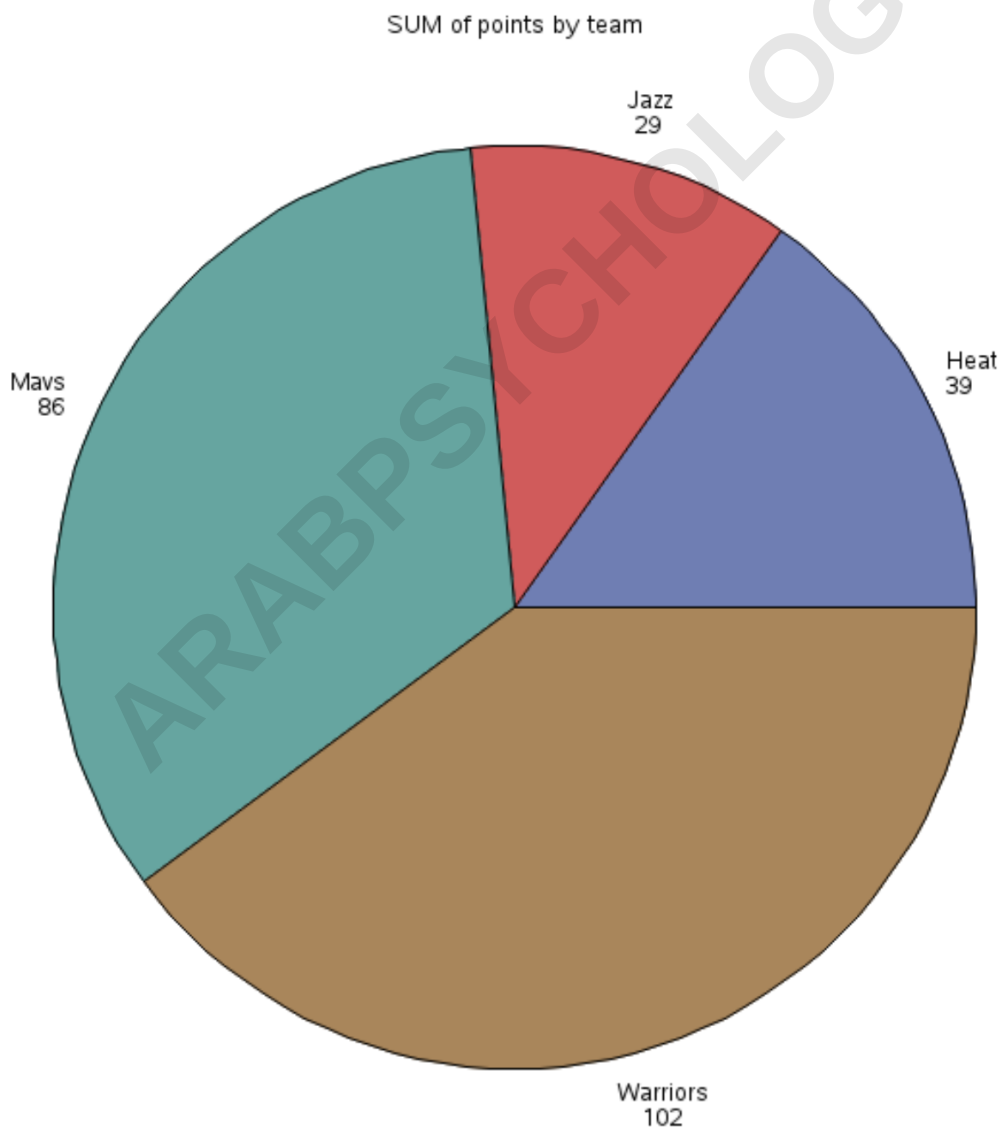
While frequency charts are useful for displaying counts, often the analytical interest lies in summarizing a quantitative measure across categories. For instance, we might want to know the total points scored by each team, rather than just the number of games logged. This transition from frequency visualization to summarized data visualization requires the use of the **SUMVAR** option within the **PIE** statement. The **SUMVAR** option specifies the variable whose values should be summed for each unique level of the classification variable (in our case, **team**).

In this example, we maintain **team** as the classification variable for the slices, but we introduce **SUMVAR=points**. This instructs **PROC GCHART** to aggregate the **points** column for all records belonging to the 'Mavs', 'Heat', 'Warriors', or 'Jazz' categories separately. Consequently, the area

of each slice is determined by the total accumulated points, not the raw count of observations. This is a critical distinction that changes the interpretation of the resulting pie chart entirely.

```
proc gchart data=my_data;  
pie team / sumvar=points;  
run;  
quit;
```

As visible in the output, the slices now reflect the percentage contribution of total points. Notice how the Warriors slice, despite having only three records (fewer than the Mavs' four records), now appears larger because their total accumulated points are higher. Specifically, the slices in this pie chart represent the sum of values in the **points** column for each unique value in the **team** column, providing a powerful visual metric of performance comparison among the teams.



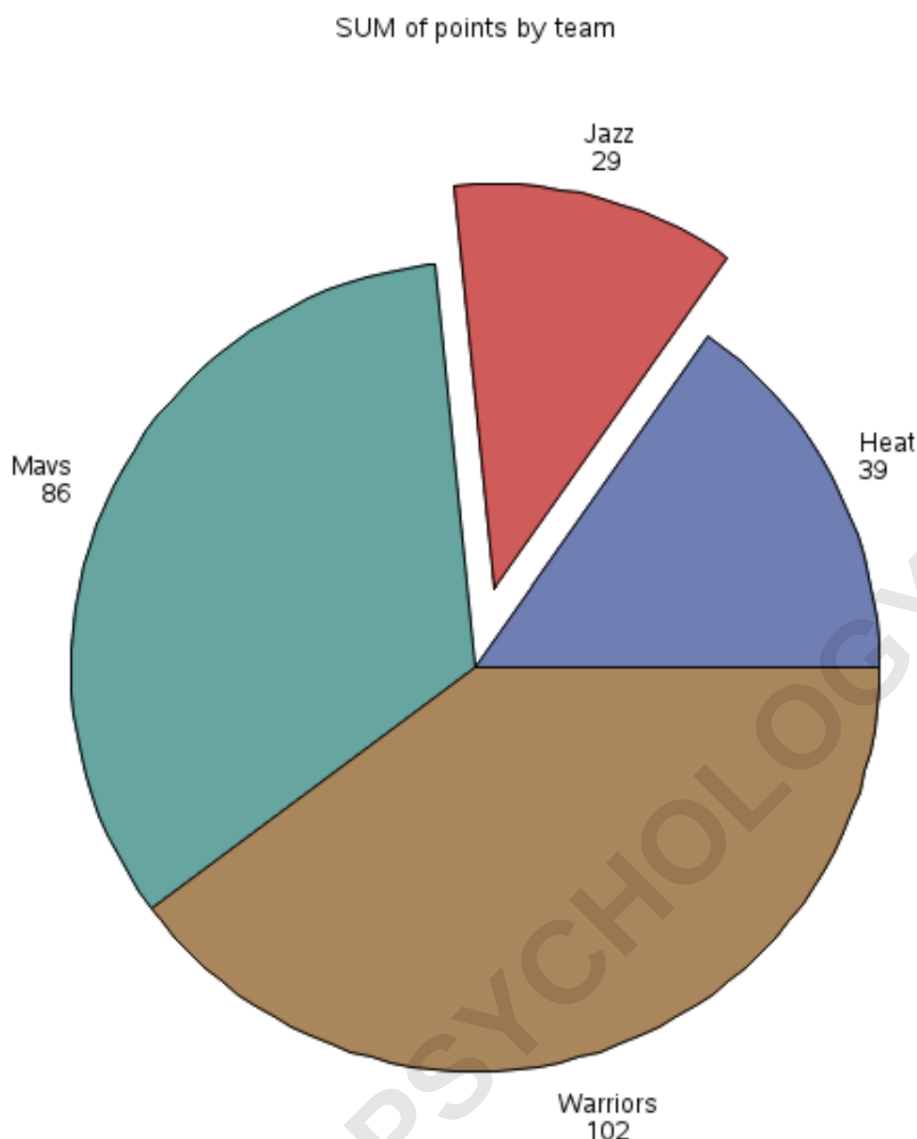
Example 3: Highlighting Key Segments using EXPLODE

In data visualization, emphasis is often necessary to draw the viewer's attention to a particular category that holds strategic importance, represents an outlier, or is the focus of a report. PROC GCHART facilitates this emphasis through the **EXPLODE** option, which is specified on the **PIE** statement. When a category is 'exploded,' its corresponding slice is slightly pulled away from the center of the pie, making it visually distinct from the rest of the distribution.

To demonstrate this, we will reuse the sum of points calculation from Example 2, but we will apply the **EXPLODE** option to the 'Jazz' team category. The syntax requires specifying the classification variable value (enclosed in quotes) that should be detached. This technique is invaluable when presenting data where a specific component needs immediate recognition, perhaps because it represents the smallest contribution or, conversely, a surprisingly large one.

```
proc gchart data=my_data;  
pie team / sumvar=points explode='Jazz';  
run;  
quit;
```

The resulting chart clearly shows the slice representing the Jazz team separated from the main body of the pie chart. This visual separation immediately directs the viewer's eye, successfully highlighting the Jazz's contribution relative to the others. This functionality can be particularly useful if you want to create a pie chart and intentionally draw focus to one of the slices, perhaps to discuss why that specific team's score is lower or higher compared to the group average.



Example 4: Enhancing Readability with Custom Labels (PLABEL Statement)

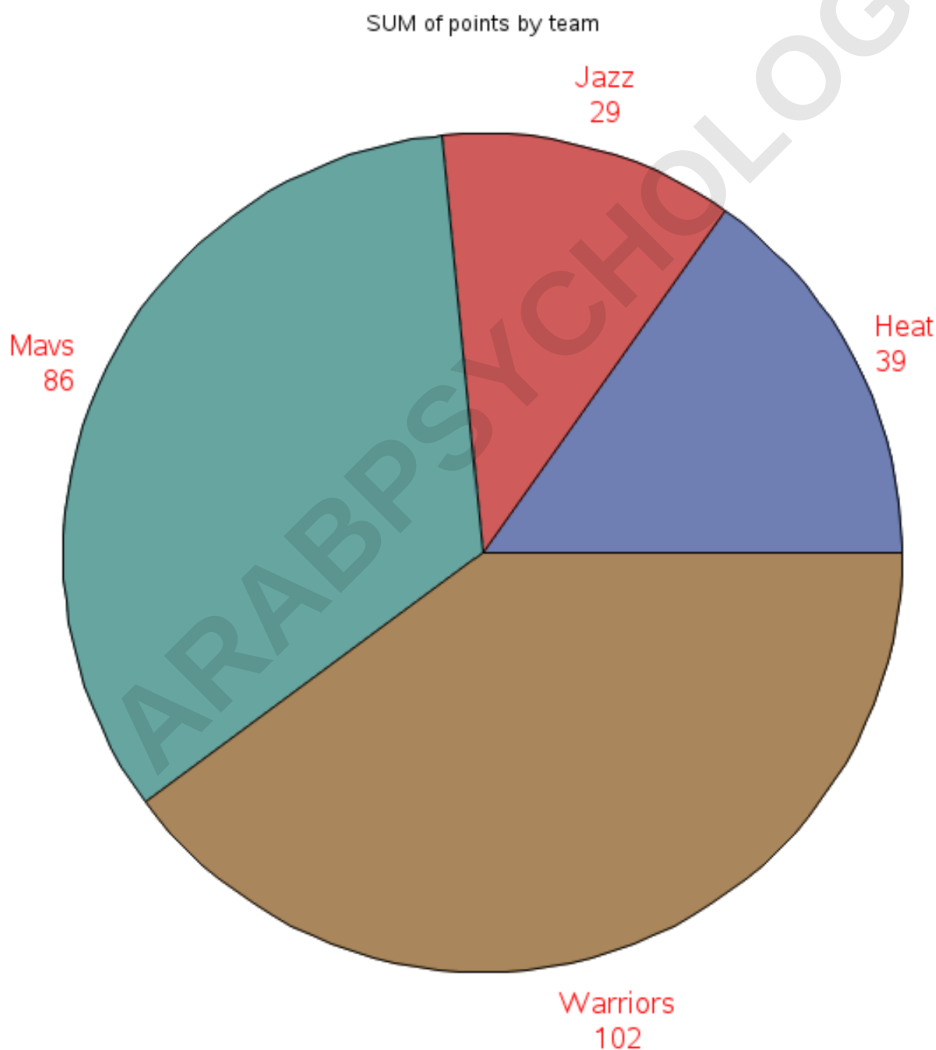
While the visual structure of the pie is important, the interpretability heavily relies on the quality and presentation of the associated text, or labels. `PROC GCHART` allows extensive customization of these labels using the **PLABEL** option. This option controls the appearance of the labels displayed near the pie slices, including attributes such as font size (height) and color. Customizing these features ensures accessibility and improves the aesthetic quality of the final graph, especially when preparing charts for formal presentations or publications.

In this demonstration, we utilize the **PLABEL** option to increase the font size and change the font color to red. The syntax **PLABEL=(h=1.5 color=red)** is appended to the **PIE** statement, where 'h' specifies the height of the text (in character units, often 1.5 times the default size), and 'color' sets the text color. This customization significantly enhances the visibility of the text components,

making the statistical information immediately obvious to the viewer. Such fine-grained control over graphical elements is a hallmark of the SAS/GRAPH environment.

```
proc gchart data=my_data;  
pie team / sumvar=points plabel=(h=1.5 color=red);;  
run;  
quit;
```

The resulting chart shows labels that are larger and displayed in red, contrasting sharply with the default black text. Note that the **h** argument dictates the size of the font, and the **color** argument specifies the font color, making the labels much more prominent. Effective use of **PLABEL** ensures that essential data points--such as the percentage contribution of each team's total points--are conveyed with maximum impact.



Summary of PIE Statement Customization Options

Generating pie charts in SAS using PROC GCHART is highly flexible, allowing analysts to move seamlessly between displaying simple frequency distributions and complex statistical summaries. The four examples presented illustrate the core capabilities needed for most reporting requirements. When you require a straightforward count of observations per category, the basic **PIE variable**; syntax suffices. However, when summarizing numeric data is necessary, the **SUMVAR=** option becomes indispensable, defining the quantitative measure that determines the slice sizes.

For scenarios demanding visual emphasis, the **EXPLODE=** option offers a simple yet effective way to isolate a category, drawing the viewer's immediate attention to key data points or anomalies within the distribution. Finally, ensuring clarity and professionalism involves customizing the text output through **PLABEL**, allowing control over text size, color, and positioning of statistical labels. Mastery of these options within PROC GCHART enables the creation of highly informative and visually appealing pie charts that effectively communicate proportional data derived from any given dataset.

Further SAS Visualization Techniques

While this article focused on the **PIE** statement, PROC GCHART offers additional chart types that are crucial for visualizing different kinds of statistical information. Depending on the nature of your variables--whether they are continuous, ordinal, or nominal--you may need alternatives to the pie chart.

Consider the following related visualization techniques available in SAS:

Bar Charts (VBAR/HBAR): These are ideal for comparing discrete categories, especially when the number of categories is large or when comparing the magnitude of values rather than just proportions of a whole.

Block Charts (BLOCK): Used to display data where each block size is proportional to the variable's value, offering another way to visualize magnitudes across two classification variables.

Donut Charts (DONUT): A variation of the pie chart, often preferred for aesthetics or when central space is required for additional metrics or titles.

Selecting the appropriate visualization tool is paramount for effective data communication. While pie charts excel at showing contributions to a total, bar charts often provide better comparative accuracy for individual category magnitudes.

The following tutorials explain how to create other charts in SAS: