

# How to Make a Multi-Series Scatterplot in Google Sheets

Authored by  
**stats writer**

February 1, 2026

## RECOMMENDED CITATION

stats writer (2026). *How to Make a Multi-Series Scatterplot in Google Sheets*.

PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=129026>

Welcome to this comprehensive guide on creating a scatterplot with multiple series within Google Sheets. While creating a basic chart is straightforward, visualizing grouped data as distinct series requires specific data manipulation techniques. A multi-series scatterplot is an invaluable tool in data visualization, enabling you to compare and contrast several sets of metrics simultaneously on a single, intuitive graph.

Before diving into the complex data preparation required for grouped series, it is helpful to understand the standard procedure for initiating a scatterplot. If your data is already pre-formatted in separate, adjacent columns (X-values followed by Y-values for each series), the standard Google Sheets charting mechanism works flawlessly. However, for real-world datasets where groups are identified by a single categorization column, the data must be restructured. We will first outline the general steps and then proceed to the necessary data preparation formula.

## Mastering Multi-Series Scatterplots in Google Sheets

The ability to represent distinct groups of data points on a single chart--where each group is identified by a unique color or marker--significantly enhances analytical clarity. This technique is often necessary when comparing performance metrics across different categories, regions, or, as we will demonstrate, teams.

To begin the process of chart creation using the standard method, follow these initial steps:

Open your Google Sheets document and carefully select the entire range of data you intend to visualize on the scatterplot.

Navigate to the top menu bar, click on **Insert**, and then choose **Chart**. This action opens the powerful Chart Editor panel on the right side of your screen.

Within the Chart Editor, under the Setup tab, verify that the **Chart type** is set to **Scatter chart**.

If the data is structured appropriately, you can navigate to the **Customize** tab and select **Series**. Here, you can click **Add Series** and manually specify the data ranges for each dependent variable you wish to include.

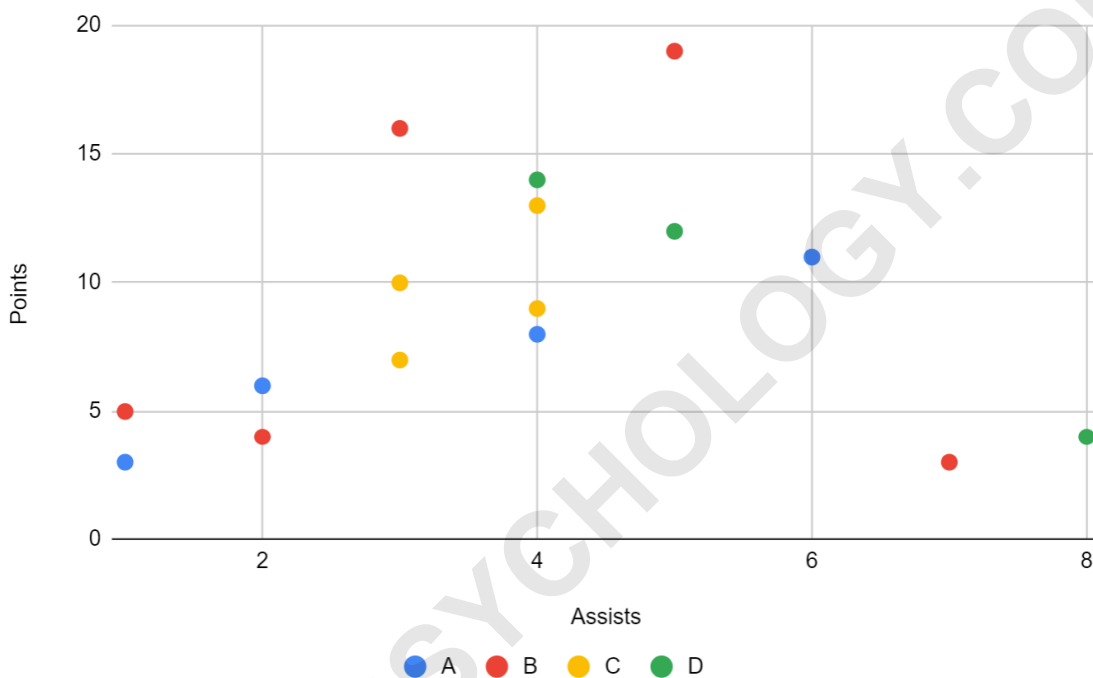
After defining all required data series, utilize the customization options to refine the appearance of the scatterplot. Adjust elements such as the chart title, axis labels, legend placement, and the specific colors and shapes used for the data points.

By executing these steps, you create the foundation for effective data comparison. The main challenge, however, arises when your dataset is organized vertically (long format), requiring a sophisticated transformation before Google Sheets can recognize the groups as individual series. The following methodology focuses on solving this common data preparation hurdle.

## Google Sheets: Create a Scatterplot with Multiple Series

While the standard steps suffice for simple data arrangements, real-world data is often structured in a way that necessitates preparatory transformation. Our goal is to generate a visual representation where distinct data series are easily identifiable, similar to the plot demonstrated below:

### Assists vs. Points by Team



Fortunately, achieving this sophisticated visualization is entirely feasible within [Google Sheets](#) by leveraging conditional formulas. This step-by-step example will guide you through the precise process required to structure your raw data for multi-series charting.

### Step 1: Entering and Reviewing the Raw Dataset

The first critical step involves establishing the initial dataset. For this demonstration, we will use a dataset containing performance metrics--Assists and Points--for various basketball players, categorized by the teams they represent. This structure is typical of data exported from databases or collected systems where categorical labels are listed alongside numerical values.

The dataset should be entered into your sheet starting from cell A1, ensuring clear headers for each column: Player, Team, Assists, and Points. Note that the key challenge here is the 'Team' column, which dictates the grouping for our desired series. Google Sheets needs this categorical

data to be spread horizontally into separate columns before it can render distinct series markers.

	A	B	C	D	
1	<b>Team</b>	<b>Assists</b>	<b>Points</b>		
2	A	1	3		
3	B	1	5		
4	B	2	4		
5	A	2	6		
6	C	3	7		
7	A	4	8		
8	C	4	9		
9	C	4	13		
10	C	3	10		
11	D	5	12		
12	A	6	11		
13	B	7	3		
14	D	8	4		
15	D	4	14		
16	B	5	19		
17	B	3	16		
18					
19					

In this example, the independent variable (X-axis) will be **Assists**, and the dependent variable (Y-axis) will be **Points**. We aim to create a distinct scatterplot series for each unique Team (A, B, C, D), allowing for an immediate visual comparison of scoring efficiency across the different organizations.

## Step 2: Formatting the Data for Multi-Series Recognition

Before any scatterplot can be generated that recognizes multiple groups, the dataset must undergo a transformation. We need to convert the vertically grouped data into a wide format where each series occupies its own dedicated column. This transformation relies heavily on the conditional logic provided by the `IF` function combined with the `NA()` function for purposeful error handling.

Start by extracting the unique values from the 'Team' column (Column A) and placing them as new headers along the top row, starting from an empty column, such as E1. These headers (Team A, Team B, Team C, Team D) will serve as the labels for our new data series columns.

The essence of this transformation is to ensure that a numerical value (Points) only appears in a team's dedicated column if the row classification matches that team. If the classification does not

match, we must force an error value, preventing the point from being plotted as part of that series.

### Step 3: Implementing the Array Formula for Series Segmentation

To segment the 'Points' data based on the 'Team' classification, we will utilize a specific conditional formula in the new output area. In cell **E2**, enter the following formula. This formula checks the category in Column A against the header in Row 1 (Team A) and returns the corresponding value from the 'Points' column (Column C) only if the condition is met. Otherwise, it returns the error value `NA()`.

**=IF(\$A2=E\$1, \$C2, NA())**

Understanding the structure of the formula is vital for correct implementation. The use of absolute (\$) and mixed references (e.g., \$A2, E\$1) is deliberate. \$A2 ensures that as you drag the formula right or down, the comparison always refers back to the team name in Column A for that specific row. Conversely, E\$1 ensures that when dragging down, the column still refers to the Team header in Row 1, but when dragging right, it correctly shifts to the next team header (F\$1, G\$1, etc.).

The `NA()` function is extremely important in this context. When creating a scatterplot in Google Sheets, a cell containing `NA()` is interpreted as 'Not Applicable' and is completely ignored by the charting engine. If we had used an empty string (" ") or a zero (0) instead, the chart would incorrectly plot these values, distorting the visual patterns. Using `NA()` ensures that only relevant data points are plotted for each series.

### Step 4: Applying the Transformation Across the Dataset

Once the formula is correctly entered in cell E2, it must be applied across the entire transformation area. First, drag this formula horizontally across the header rows until you reach the column corresponding to your last unique team (cell **H2** in our four-team example).

Next, with the entire row of newly calculated values selected (E2 to H2), drag the selection downwards until you reach the last row of your original data (cell **H17**). This action populates the new columns E through H, resulting in a matrix where points data is distributed across the appropriate team columns. Only one cell per row in columns E through H will contain a numerical value; the rest will display the `#N/A` error, which is the desired outcome for successful plotting.

E2  $\text{=IF}(\$A2=\$E\$1, \$C2, \text{NA}())$

	A	B	C	D	E	F	G	H
1	<b>Player</b>	<b>Assists</b>	<b>Points</b>		<b>A</b>	<b>B</b>	<b>C</b>	<b>D</b>
2	A	1	3		3	#N/A	#N/A	#N/A
3	B	1	5		#N/A	5	#N/A	#N/A
4	B	2	4		#N/A	4	#N/A	#N/A
5	A	2	6		6	#N/A	#N/A	#N/A
6	C	3	7		#N/A	#N/A	7	#N/A
7	A	4	8		8	#N/A	#N/A	#N/A
8	C	4	9		#N/A	#N/A	9	#N/A
9	C	4	13		#N/A	#N/A	13	#N/A
10	C	3	10		#N/A	#N/A	10	#N/A
11	D	5	12		#N/A	#N/A	#N/A	12
12	A	6	11		11	#N/A	#N/A	#N/A
13	B	7	3		#N/A	3	#N/A	#N/A
14	D	8	4		#N/A	#N/A	#N/A	4
15	D	4	14		#N/A	#N/A	#N/A	14
16	B	5	19		#N/A	19	#N/A	#N/A
17	B	3	16		#N/A	16	#N/A	#N/A
18								
19								

This restructured format is now perfectly aligned for Google Sheets' charting mechanism. Each new column (E, F, G, H) now represents a distinct data series, ready to be plotted against the common X-axis variable, which in this case, is Assists.

### Step 5: Preparing the X-Axis Variable for Charting

Although the Points data (Y-axis) is now correctly segmented by team, we still need to pair these segmented Y-values with the appropriate X-axis variable (Assists). When selecting data for a scatterplot with multiple series, Google Sheets typically expects the X-axis column to be immediately adjacent to the series columns.

Therefore, copy all values from Column B (Assists) and paste them into Column D. Column D will now serve as the master X-variable column against which all subsequent Y-series columns (E through H) will be plotted. This alignment ensures the chart editor correctly interprets the data structure.

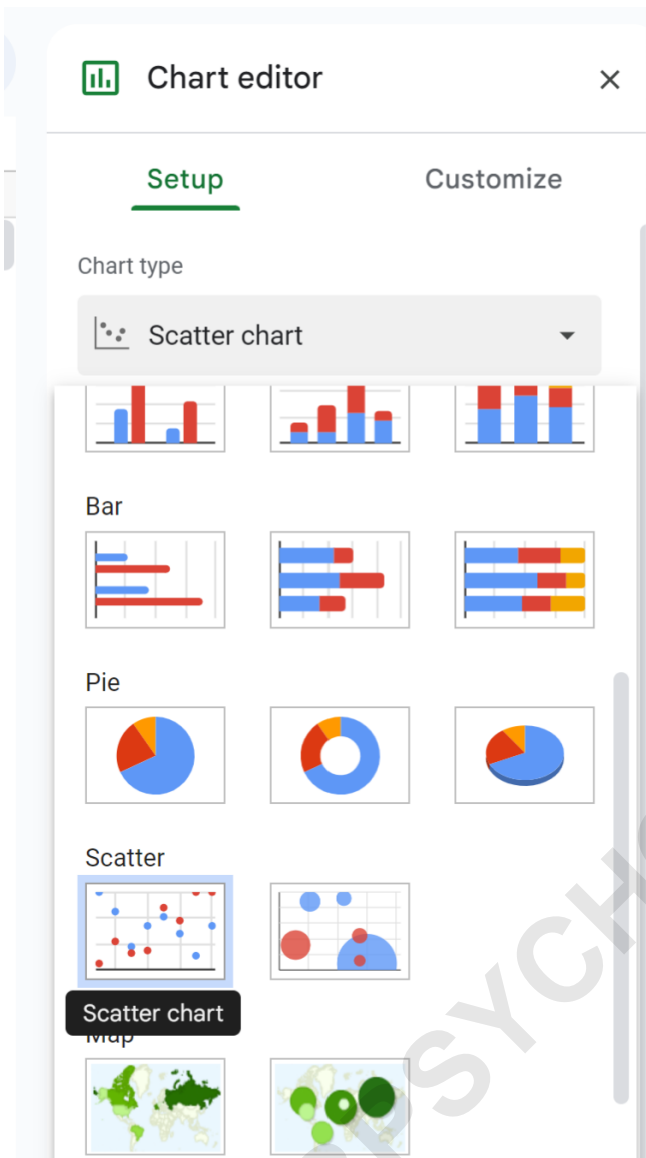
	C	D	E	F	G	H	
	Points	Assists	A	B	C	D	
1	3	1	3	#N/A	#N/A	#N/A	
1	5	1	#N/A		5	#N/A	#N/A
2	4	2	#N/A		4	#N/A	#N/A
2	6	2	6	#N/A	#N/A	#N/A	#N/A
3	7	3	#N/A	#N/A		7	#N/A
4	8	4	8	#N/A	#N/A	#N/A	#N/A
4	9	4	#N/A	#N/A		9	#N/A
4	13	4	#N/A	#N/A		13	#N/A
3	10	3	#N/A	#N/A		10	#N/A
5	12	5	#N/A	#N/A	#N/A		12
6	11	6	11	#N/A	#N/A	#N/A	#N/A
7	3	7	#N/A		3	#N/A	#N/A
8	4	8	#N/A	#N/A	#N/A		4
4	14	4	#N/A	#N/A	#N/A		14
5	19	5	#N/A		19	#N/A	#N/A
3	16	3	#N/A		16	#N/A	#N/A

The final range we will select for charting consists of Column D (Assists, X-axis) and Columns E through H (Points segmented by Team, Y-series). This contiguous block of data is the input required to generate the multi-series scatterplot successfully.

## Step 6: Executing the Chart Creation Process

With the data meticulously formatted, the actual chart creation is straightforward. Select the entire range including the X-axis (Column D) and all the newly formatted Y-series columns (E through H). Ensure you include the header row (D1:H1) in your selection, as Google Sheets uses these to automatically label the series.

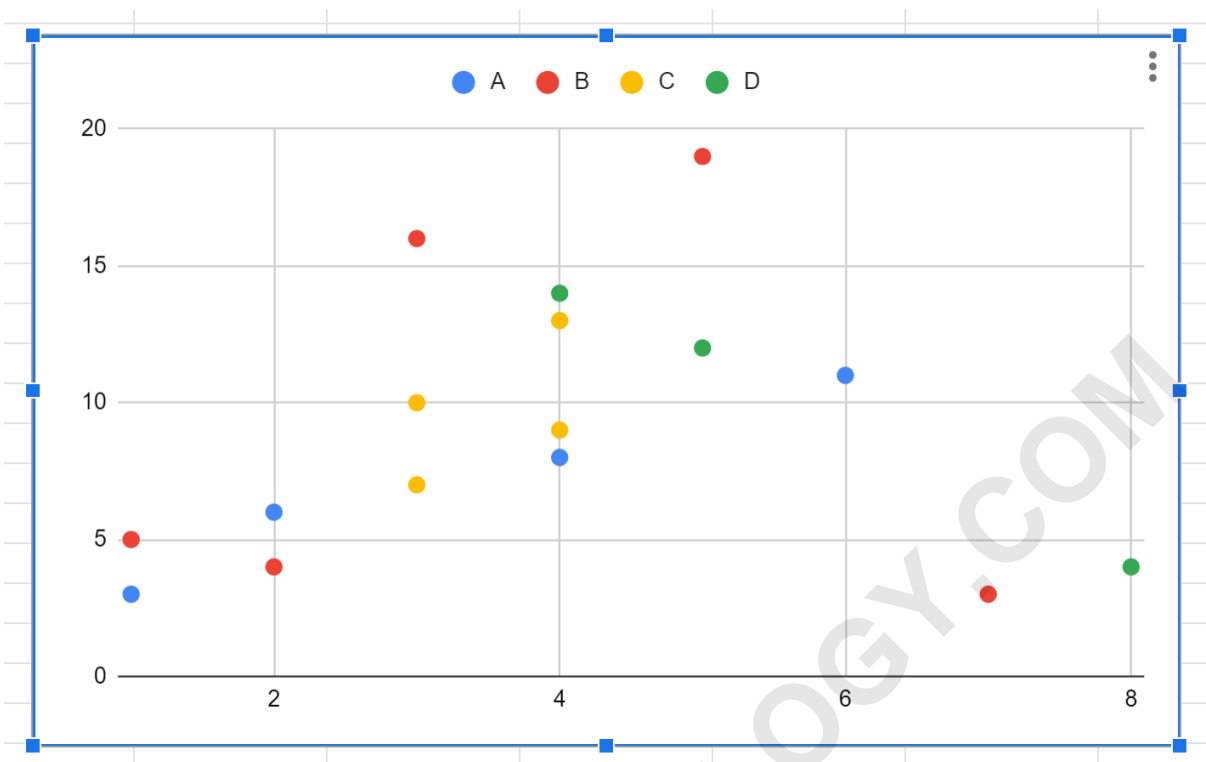
Go to **Insert > Chart**. The Chart Editor panel will reappear. If it doesn't automatically detect the correct type, navigate to the Setup tab and explicitly choose **Scatter chart** as the desired visualization type.



Upon selection, Google Sheets automatically processes the contiguous data, recognizing Column D as the X-axis and assigning each of the subsequent columns (E, F, G, H) as distinct data series, differentiated by color. The resulting chart will visually map the relationship between Assists and Points for every player, with the color of each point indicating the specific team.

### Step 7: Customizing and Interpreting the Visual Output

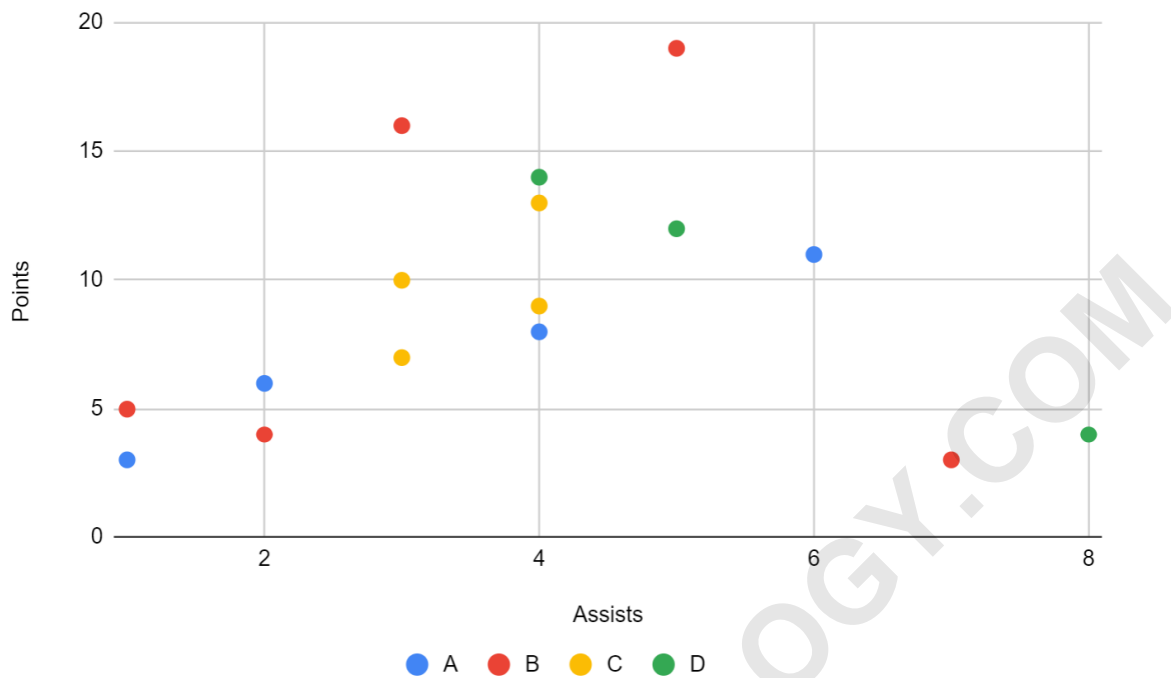
The generated scatterplot provides an immediate visual summary of the data distribution:



Each individual point on this visualization represents a unique basketball player. Crucially, the colors of these points correspond directly to the teams defined in our segmented columns. This clear separation allows for rapid identification of performance patterns. For instance, you can easily observe if one team clusters in a high-point, high-assist quadrant, suggesting greater overall efficiency, while another team might show a wider or lower distribution of metrics.

To maximize the clarity and professional quality of the visualization, it is highly recommended to use the **Customize** tab in the Chart Editor to add a descriptive title and precise axis labels. Clear labeling is fundamental for effective data visualization and ensures that any audience can accurately interpret the findings.

## Assists vs. Points by Team



The final, polished chart offers a powerful analytical perspective, successfully merging four categorical groups into a single, cohesive scatterplot.

### Conclusion: Enhancing Data Insights through Visualization

Creating a scatterplot with multiple series in Google Sheets, particularly when dealing with pre-grouped data, requires strategic use of conditional logic and data restructuring. By employing the `IF` function combined with the `NA()` error handler, we successfully transformed long-format data into a wide-format matrix that the charting tool recognizes as distinct series.

This methodology is transferable to any dataset where you need to visualize the relationship between two numerical variables segmented by a categorical variable. Mastering this technique is a significant step toward advanced proficiency in data analysis and presentation using spreadsheet software.

The following tutorials explain how to perform other common tasks in Google Sheets, allowing you to further expand your analytical capabilities:

[How to Calculate Conditional Variance in Google Sheets](#)

[How to Use the INDEX MATCH Function in Google Sheets](#)

[How to Calculate Interquartile Range \(IQR\) in Google Sheets](#)