

How to Create Proportional Venn Diagrams in R: A Step-by-Step Guide

Authored by
stats writer

January 31, 2026

RECOMMENDED CITATION

stats writer (2026). *How to Create Proportional Venn Diagrams in R: A Step-by-Step Guide*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=128872>

Creating a proportional Venn diagram in R is a crucial skill for advanced data visualization. This specialized diagrammatic tool moves beyond simple qualitative set intersection by accurately reflecting the quantitative relationships and relative sizes of multiple data sets. Understanding and implementing this type of visualization is essential for researchers and analysts who need to communicate the true scale and magnitude of overlaps between different groups.

Unlike traditional Venn diagrams, which often display sets as equal-sized circles regardless of their actual data volume, a proportional Venn diagram ensures that the area of each circle, and the area of their intersections, are scaled precisely to the input data. This enhanced accuracy allows for a more intuitive and statistically robust representation of complex data structures. This comprehensive guide details the precise methods for generating clean, validated, and visually effective proportional Venn diagrams using the eulerr package in R.

Generating Proportional Venn Diagrams in R Using the eulerr Package

Defining Set Diagrams: Venn vs. Proportional Euler Diagrams

The concepts of Venn diagrams and Euler diagrams are closely related, yet distinct, especially when proportionality is considered. A Venn diagram is fundamentally a diagram that uses overlapping closed curves, typically circles, to demonstrate all possible logical relations between a finite collection of sets. Every possible intersection, including the null intersection, must be represented in a standard Venn diagram.

A proportional Venn diagram, more accurately termed a proportional Euler diagram, takes this visualization a

step further. In this specific format, the area occupied by each set (the circle size) and the area of the overlaps (the intersection size) are scaled precisely to represent the sample size or magnitude of each group and their common elements. This quantitative accuracy is vital when the viewer needs to gauge the relative importance or prevalence of different categories within the overall data structure.

To successfully generate these sophisticated diagrams in R, the most efficient and recommended approach involves leveraging the functionality of the `eulerr` package. This package is specifically designed to fit Euler diagrams to area specifications, ensuring high fidelity between the input data and the resulting visual output.

Introducing the `eulerr` Package for Accurate Visualization

The R programming environment offers several packages for generating set visualizations, such as `R` base graphics or the `VennDiagram` package. However, the `eulerr` package excels in generating truly proportional diagrams by minimizing the residual sum of squares between the requested areas and the areas

displayed in the resulting plot. This mathematical approach ensures that the visual representation is as accurate as possible, making it the preferred tool for high-stakes statistical reporting.

Before proceeding with any coding, ensure that the `eulerr` package is installed and loaded into your R session. Installation is typically performed once using the standard R command, followed by loading the library for use. Once the library is active, the primary function we utilize is `euler()`, which processes the input data structure, and the standard `plot()` function, which renders the fitted object.

Example 1: Generating a Basic Proportional Venn Diagram in R

To illustrate the practical application of the `eulerr` package, consider a scenario involving two distinct groups, A and B, which share a specific number of common elements. For proportional visualization, we must define the area of Group A only, the area of Group B only, and the area of their mutual intersection.

Suppose our data represents the following counts:

A (Unique Elements): 100 individuals or observations.

B (Unique Elements): 500 individuals or observations.

A & B Overlap (Intersection): 75 common elements.

It is important to note that the input values provided to the `euler()` function represent the counts of the elements in the specific regions, not the total count of the entire set. For example, the value for 'A' (100) represents the area of A excluding B, and 'A&B' (75) represents the area of the intersection.

Step-by-Step Implementation and Data Specification

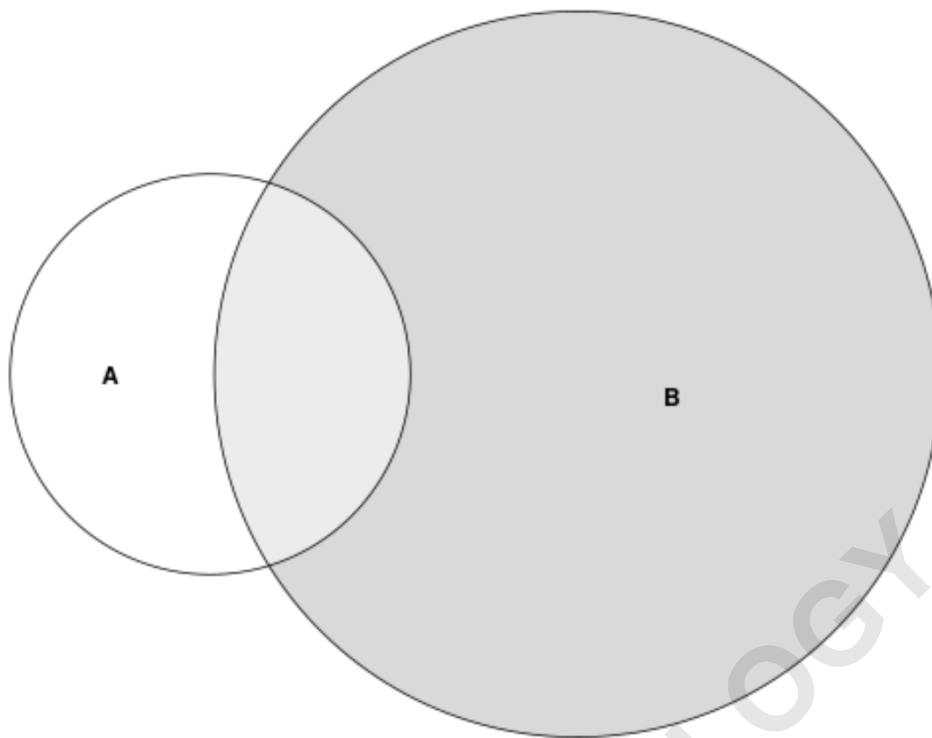
To create a proportional Venn diagram that accurately reflects the relationship between these two groups, we define the counts using a named vector format, which is the required input structure for the `euler()` function. This function performs the necessary calculations to map these proportional areas to the geometric shapes.

The code below first loads the required package, specifies the named vector structure with the defined counts, and then uses the `generic plot()` function to render the fitted Euler diagram object.

```
library(eulerr)
```

```
#specify values to use in venn diagram  
fit <- euler(c('A' = 100, 'B' = 500, 'A&B' = 75))  
  
#create venn diagram  
plot(fit)
```

Execution of this code produces the initial visualization. The result is a diagram where the circle representing Set B is noticeably larger than the circle representing Set A, accurately reflecting the significantly higher count (500) associated with B compared to A (100, plus the 75 overlap). The area of the overlap is also minimized relative to the total area, reflecting its proportion of 75 elements.



Upon inspection of the resulting graph, it is immediately clear that the sizes of the circles are accurately representative of the values specified in the input data structure. This visual fidelity is the primary advantage of using a proportional Euler diagram: it immediately communicates the relative weight and size of each category, which is essential for effective data storytelling and unbiased presentation of findings.

Enhancing Visualization: Customizing Colors and Aesthetics

While the default plot generated by `eulerr` is statistically sound, aesthetic customization is often necessary to

match corporate branding, improve clarity, or highlight specific data subsets. The `plot()` function, when used on the fitted Euler object, accepts standard graphical parameters, including arguments for controlling the fill color of the sets.

To customize the colors used in the proportional Venn diagram, you can employ the `fill` argument within the `plot()` function. This argument requires a vector of color names or hexadecimal codes, with the order corresponding to the order of the sets defined in the original `euler()` function call. In our two-set example, the first color in the vector will apply to Set A, and the second color will apply to Set B.

Customizing colors can significantly improve the diagram's readability, especially when dealing with complex data that might otherwise appear monochromatic or confusing. Selecting colors with high contrast is generally recommended for maximizing visual impact and accessibility.

Step-by-Step Example 2: Implementing Custom Color Schemes

Let us re-run the previous example, but this time, we

will specify distinct colors for Sets A and B using descriptive color names recognizable by R. We will assign 'coral2' to Set A and 'steelblue' to Set B. This process requires only a minor modification to the final plotting command, demonstrating the ease with which aesthetic changes can be applied.

```
library(eulerr)
```

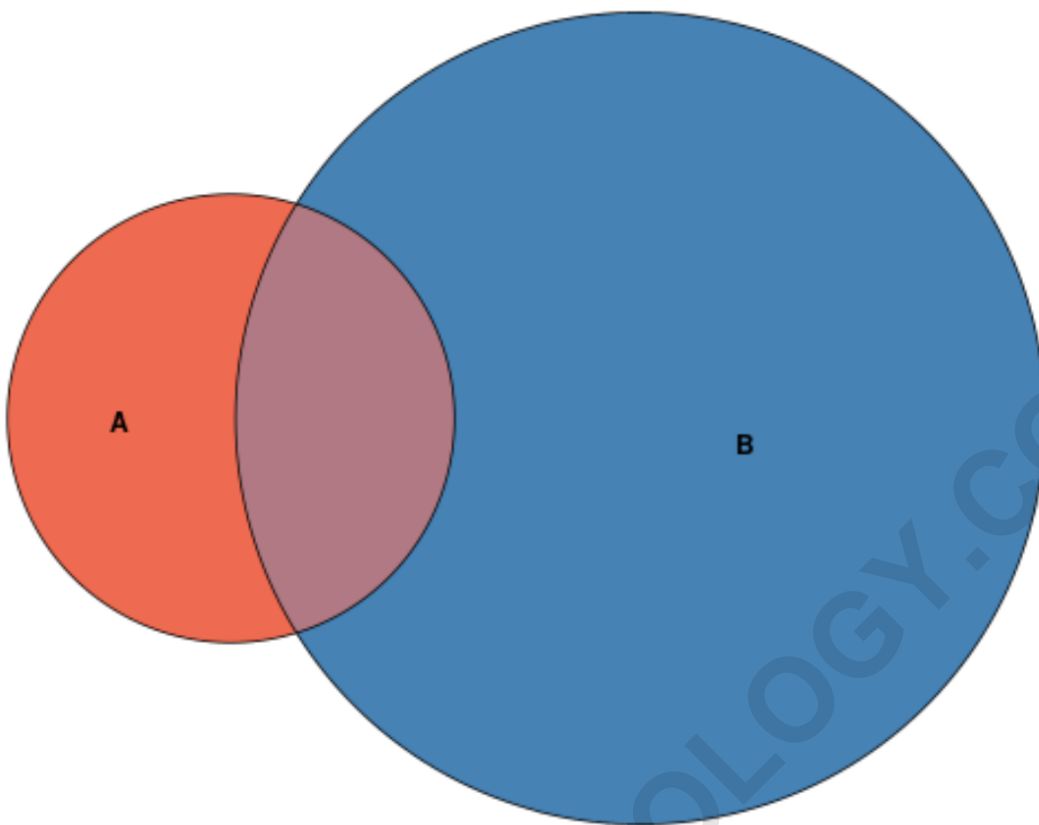
```
#specify values to use in venn diagram
```

```
fit <- euler(c('A' = 100, 'B' = 500, 'A&B' = 75))
```

```
#create venn diagram with custom colors
```

```
plot(fit, fill=c('coral2', 'steelblue'))
```

Executing this modified code yields the same geometrically accurate proportional diagram but with the newly specified color scheme. This subtle change dramatically alters the visual appeal and potential interpretation of the diagram.



As demonstrated in the resulting visualization, the A circle now features a coral2 fill color, while the B circle utilizes a darker steelblue hue. This modification, achieved simply by passing a character vector of color names to the fill argument in the plot() function, allows for full control over the visual presentation of the proportional data.

Advanced Considerations and Interpretation Best Practices

While the eulerr package provides an exceptionally accurate method for generating proportional diagrams,

data analysts must remain mindful of certain limitations and best practices during interpretation.

Firstly, the difficulty of maintaining strict proportionality increases exponentially with the number of sets. While two or three sets generally result in highly accurate visual fits, generating proportional diagrams for four or more sets often results in higher calculation error, meaning the visual areas may slightly deviate from the specified numerical areas. The `eulerr` package attempts to minimize this error, but users should always check the fit statistics provided by the `euler()` function for complex scenarios.

Secondly, when interpreting a proportional Venn diagram, the focus should remain on the area of the circles and intersections, not the diameter or radius. The visual communication relies entirely on the two-dimensional area accurately reflecting the count data. Ensure that annotations and labels are clear, especially for the intersection areas, to prevent misinterpretation of the relative counts.

Note: For users interested in delving deeper into the mathematical algorithms and advanced functionalities--

such as handling missing intersections, fitting diagrams for complex set configurations, or customizing boundaries--refer to the comprehensive [eulerr package](#) documentation available on [CRAN](#) for a complete introduction to its capabilities in R.

Conclusion: Mastering Proportional Data Representation

Creating proportional Venn diagrams in R using the `eulerr` package provides a powerful and statistically rigorous method for visualizing set relationships. By accurately translating numerical counts into scaled areas, these diagrams offer far superior insight compared to their non-proportional counterparts. Mastery of the `euler()` and `plot()` functions allows analysts to generate visualizations that are not only aesthetically pleasing but also faithful representations of complex quantitative data, thereby enhancing the clarity and impact of statistical reports.

[How to Create a Grouped Boxplot in R Using ggplot2](#)

[How to Create a Heatmap in R Using ggplot2](#)

[How to Create a Gantt Chart in R Using ggplot2](#)