

How to Create Proportional Venn Diagrams in R

Authored by
stats writer

January 15, 2026

RECOMMENDED CITATION

stats writer (2026). *How to Create Proportional Venn Diagrams in R*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=126282>

Introduction to Proportional Venn Diagrams in R

Creating a Venn diagram in the R programming environment is a common task for researchers and analysts seeking to visualize set relationships. However, a standard Venn diagram often only shows the qualitative overlap between sets. When the quantitative measure of each set is critical, a **Proportional Venn Diagram** (often synonymous with an **Euler diagram**) is required. This specialized visualization ensures that the area of each circle or ellipse accurately reflects the sample size or magnitude of the data it represents, providing a much richer context for data visualization.

To effectively generate these complex diagrams in R, specialized packages are necessary. While older packages like "VennDiagram" and "limma" offer functionalities for standard overlaps and statistical analysis respectively, the modern, highly effective approach relies heavily on the **eulerr** package. This package is specifically designed to fit the areas of the geometric shapes precisely to the specified set sizes and overlaps, producing a visually accurate representation that is crucial for robust scientific communication.

The initial process of generating any complex diagram in R involves preparing the environment. This typically means importing the required libraries and organizing the underlying data structure. The data or sets must be organized into a structured format, such as a named list or data frame, where the size of the individual sets and the size of their intersections are clearly defined. Using tools like the **eulerr** package simplifies the subsequent plotting process significantly, making complex proportional visualization accessible even for novice R users.

For users dealing with biological data, particularly in genomic analysis, the "limma" package is often used alongside visualization tools. While "limma" is primarily known for its differential expression analysis capabilities, it contains functions that can calculate and display the statistical significance of overlaps between gene sets, offering quantitative support for the visual representation. However, for purely aesthetic and proportional representation, the focus shifts to robust visualization libraries. Our primary focus here will be leveraging the streamlined functionality of **eulerr** to achieve optimal proportional visualization, which is the easiest and most versatile method for accurate area scaling.

Defining Venn Diagrams and Proportionality

A **Venn diagram**, originally popularized by John Venn, is a diagram that uses overlapping circles or other simple closed curves to show the logical relationship between two or more finite groups of items. These are fundamental tools in mathematics, statistics, and logic, illustrating commonalities and differences between sets. Traditionally, the sizes of the circles in a standard Venn diagram are often arbitrary and do not necessarily correspond to the size of the data sets they represent.

In contrast, a **proportional Venn diagram**, which is often more accurately termed an **Euler diagram** when dealing with non-overlapping or complex proportional relationships, is a specific type of visualization where the size of the circles or ellipses is meticulously scaled to represent the actual sample size or frequency of each group. This proportionality is vital when the viewer needs to instantly grasp not just which sets overlap, but how large those sets are relative to one another. Achieving this accurate geometric representation requires sophisticated algorithms, which modern R packages like `eulerr` handle efficiently by performing iterative fitting.

The primary advantage of using a proportional diagram is the immediate visual impact of magnitude. If Set A contains 500 observations and Set B contains 100 observations, the circle representing A should be significantly larger than the circle representing B. This fidelity to the underlying data improves clarity and prevents misinterpretation that can occur with non-proportional diagrams. In R, the easiest and most mathematically sound way to create a proportional Venn diagram is by using the `plot` function from the powerful `eulerr` package, which excels at converting set size data into geometrically accurate visualizations.

Setting Up the R Environment for Visualization

Before commencing the visualization process, the R environment must be configured. This involves ensuring that the necessary packages are installed and loaded. The primary package we will utilize is `eulerr`, which requires a single installation command if it is not already present on your system. It is important to note that R packages often rely on specific dependencies, so running the installation command ensures all necessary components are available for the fitting algorithm to execute correctly.

Once installed, the package must be loaded into the current R session using the `library()` function. This makes the functions within the package, such as `euler()` and `plot()`, available for use. Unlike some base R plotting functions, `eulerr` requires specific input formatting--the data must be provided as a named vector where the names represent the set memberships (e.g., 'A', 'B', 'A&B'). This structure is critical because it tells the fitting algorithm exactly which counts belong to which regions of the potential diagram.

The following example demonstrates the practical application of this setup. We will define two hypothetical groups, A and B, along with their quantified overlap. This structured input is the core prerequisite for using the `eulerr` package's primary fitting function.

Example: Implementing the Proportional Venn Diagram in R

To demonstrate the functionality of the `eulerr` package, let us suppose we have two groups (sets) with the following hypothetical quantitative values defining their distinct and shared elements:

A: 100 observations belong exclusively to Set A (or the total size of A is 100, if we define the input differently, though **eulerr** often accepts counts of regions). For clarity, let's assume this refers to the unique count of A.

B: 500 observations belong exclusively to Set B.

A & B Overlap: 75 observations are common to both A and B (the intersection).

To create a proportional Venn diagram that effectively visualizes the relationship and magnitude between these two groups, we must first load the library, specify the values in the required named vector format, fit the geometric shape using the `euler()` function, and finally, plot the resulting object using the generic `plot()` function. The `euler()` function performs the heavy lifting, calculating the optimal size and positioning of the shapes to maintain the specified proportions.

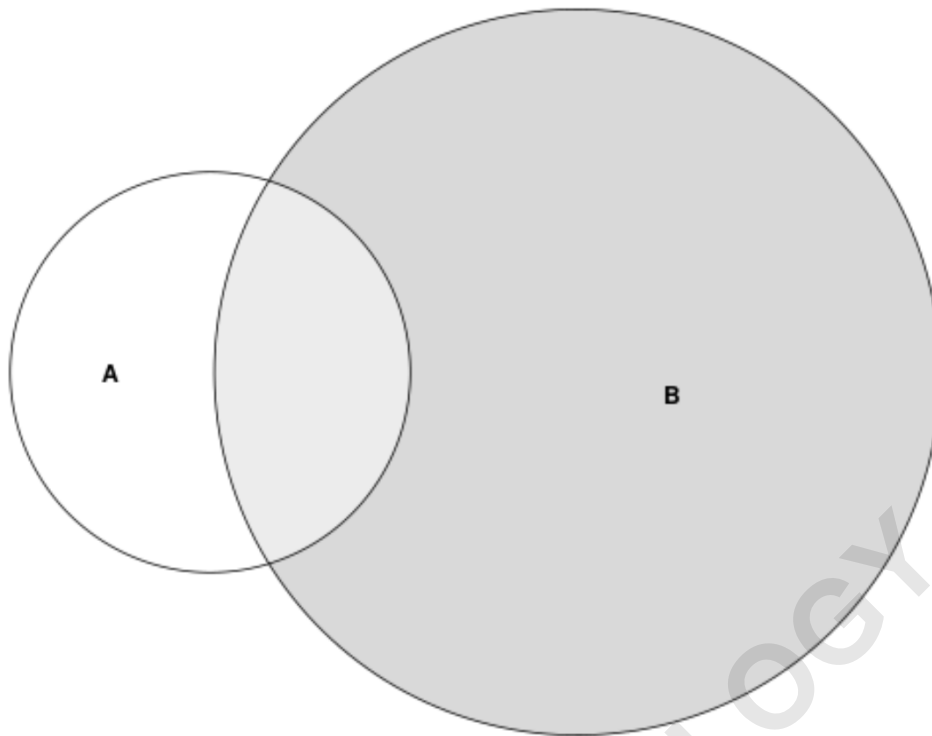
The following code snippet executes these steps, resulting in our initial, proportion-aware visualization:

library(eulerr)

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

#create venn diagram
plot(fit)
```

The critical step is the use of the `euler()` function. It takes the named vector input (where 'A', 'B', and 'A&B' are the region labels) and calculates the best graphical configuration--typically using ellipses--to accurately represent these counts. This ensures that the area of the B region is substantially larger than the area of the A region, reflecting the 500 vs. 100 count. The resulting visualization immediately communicates the significant difference in sample size between the two groups.



Upon reviewing the resulting visualization, notice that the sizes of the circles in the Venn diagram are geometrically representative of the values that we specified in the input vector. Set B (size 500) is visually much larger than Set A (size 100), and the intersection (size 75) is also appropriately scaled relative to the rest of the diagram. This visual accuracy is the defining feature of a proportional diagram and highlights why the **eulerr** package is the tool of choice for this kind of statistical data visualization in R.

Customizing Aesthetics: Color and Fill

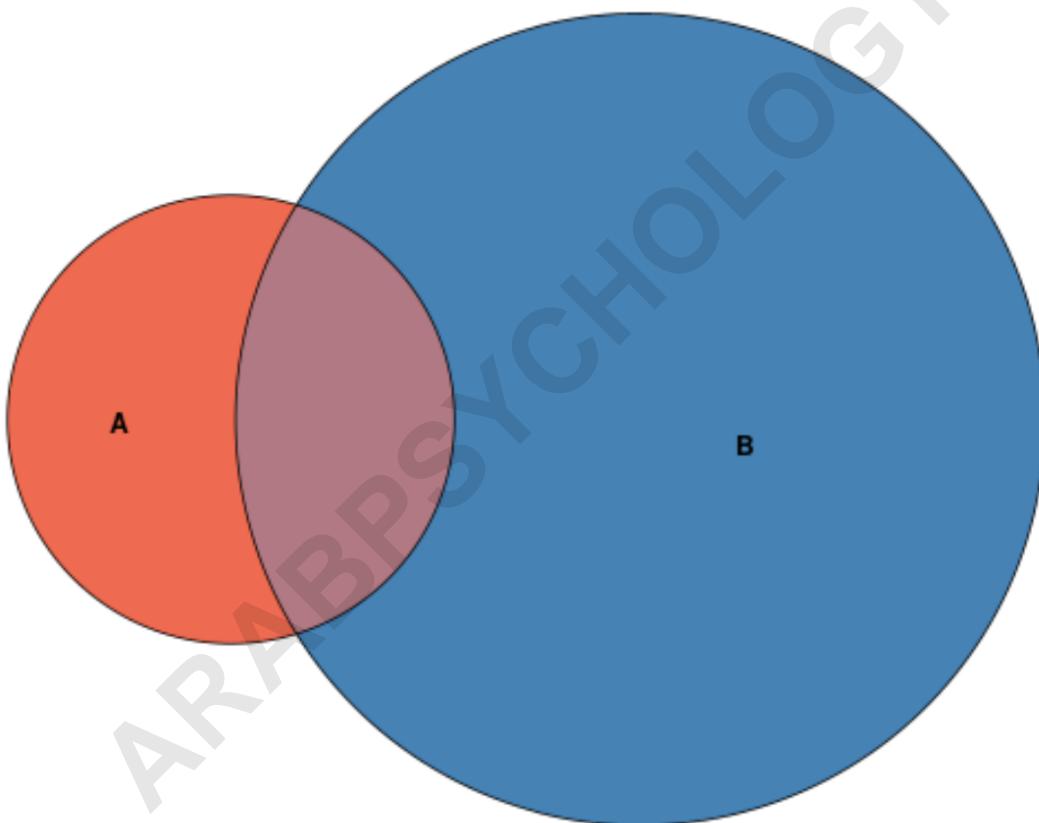
While the default plot generated by **eulerr** is informative, effective data visualization often requires customization to align with institutional branding, publication requirements, or simply to improve clarity. One of the most common aesthetic modifications is changing the color scheme of the shapes. Customizing the colors helps distinguish the sets clearly and can make the diagram more engaging for the reader.

The **eulerr** package facilitates color customization easily through the `fill` argument within the `plot()` function. This argument accepts a vector of color names or hexadecimal codes, corresponding to the sets defined in the input data. When specifying colors, the order typically matches the order in which the sets were introduced in the input vector (though it is safer to check the order in the resulting fitted object if more than three sets are involved). For a simple two-set diagram (A and B), we need to provide a vector of two colors.

If you would like to customize the colors used in the Venn diagram, you can use the **fill** argument to do so, providing explicit color names (like 'coral2' and 'steelblue') recognized by the R graphics environment. This level of control allows users to create highly professional and easily interpretable diagrams suitable for publication.

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'))
```



As demonstrated in the revised output, the A circle now has a distinct **coral2** fill and the B circle now has a clear **steelblue** fill, just as we specified using the **fill** argument in the **plot** function. Further customization options within **eulerr** allow adjustment of transparency (using the `alpha` argument), border color, and label positioning, providing comprehensive control over the final visual output.

Exploring Alternative Packages: VennDiagram and limma

While **eulerr** is the premier package for geometrically accurate proportional diagrams, it is important to acknowledge alternative tools, particularly those initially mentioned, such as "VennDiagram" and "[limma](#)". These packages serve slightly different, yet valuable, niches within the R visualization ecosystem. The "VennDiagram" package is particularly useful for generating high-quality, publication-ready standard Venn diagrams, especially for up to five sets, though its default output is often not proportionally accurate in terms of area unless specific parameters are heavily manipulated.

The "VennDiagram" package utilizes functions like `draw.pairwise.venn`, which generates two-set diagrams, or `draw.quad.venn` for four sets. These functions are highly customizable regarding text placement, font sizes, and line styles. While they can be instructed to display counts, the resulting area scaling might not perfectly reflect the proportionality of the input counts, which is where **eulerr** maintains its advantage. However, for a quick, cosmetically appealing standard diagram where proportionality is secondary to clarity of count labels, "VennDiagram" remains a strong choice.

Furthermore, the [limma](#) package, part of the Bioconductor project, is essential for bioinformatics and differential expression analysis. [limma](#) includes visualization functions, such as `vennDiagram`, which seamlessly integrate statistical results (like lists of differentially expressed genes) with set visualization. When the diagram needs to be directly derived from statistical analysis results, particularly involving large biological datasets, [limma](#) provides a structured workflow. These tools often focus on displaying the statistical significance or the identity of the overlapping entities rather than ensuring perfect proportional area fit, reinforcing **eulerr's** specialized role in area-based representation.

Advanced Features of the eulerr Package

Beyond basic proportionality and color customization, the [eulerr](#) package offers advanced features that cater to more complex data scenarios. One such feature is handling diagrams with more than three sets. While proportional representation becomes increasingly difficult (and sometimes impossible) beyond three sets using standard circles, **eulerr** attempts to find the best possible geometric fit using ellipses and complex curves, minimizing the error between the requested area and the actual area of the drawn regions. The package even calculates a "stress" metric to inform the user how accurately the resulting diagram represents the input data.

Another powerful feature is the ability to handle both counts and logical matrices as input. If data is provided as a logical matrix (where rows represent items and columns represent set membership), **eulerr** automatically calculates the intersection counts internally before fitting the diagram. This flexibility simplifies the workflow for users whose data is not pre-aggregated into counts, making it

versatile for various types of [data visualization](#) tasks within [R](#).

Furthermore, **eulerr** integrates well with other plotting systems, particularly those based on the **grid** package in R, which allows for fine-tuning graphic elements and integration into larger multi-panel figures. For a complete introduction to the comprehensive capabilities and advanced options of the **eulerr** package in R, users should refer to the official CRAN documentation and vignettes, which detail parameters for scaling, positioning, and rendering text labels.

Note: Refer to the official [eulerr documentation](#) for a complete introduction to the **eulerr** package in R, covering installation, advanced usage, and plotting parameters.

Summary and Conclusion

The creation of a **Proportional Venn Diagram** in [R](#) is highly streamlined when utilizing specialized packages designed for geometric accuracy. While several tools exist, the **eulerr** package stands out as the most reliable and user-friendly method for ensuring that the visual area of the circles or ellipses accurately reflects the underlying set sizes and intersections. This commitment to proportionality provides superior clarity over standard non-proportional diagrams, making it indispensable for accurate quantitative communication.

The process requires three main steps: 1) loading the [eulerr](#) library, 2) structuring the data into a named vector of counts, and 3) fitting and plotting the result using the `euler()` and `plot()` functions, respectively. Customizations, such as defining specific color fills, further enhance the diagram's utility and aesthetic appeal, allowing analysts to tailor visualizations precisely to their needs.

By mastering the techniques demonstrated here, R users can move beyond simple qualitative set comparisons and effectively visualize complex quantitative relationships, transforming raw data into compelling and accurate proportional representations. This powerful capability ensures that the true magnitude of differences and overlaps is immediately apparent to the viewer, serving as a critical component of professional statistical reporting and analysis.

[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](#)