

How to Easily Create a Forest Plot in R

Authored by
stats writer

December 5, 2025

RECOMMENDED CITATION

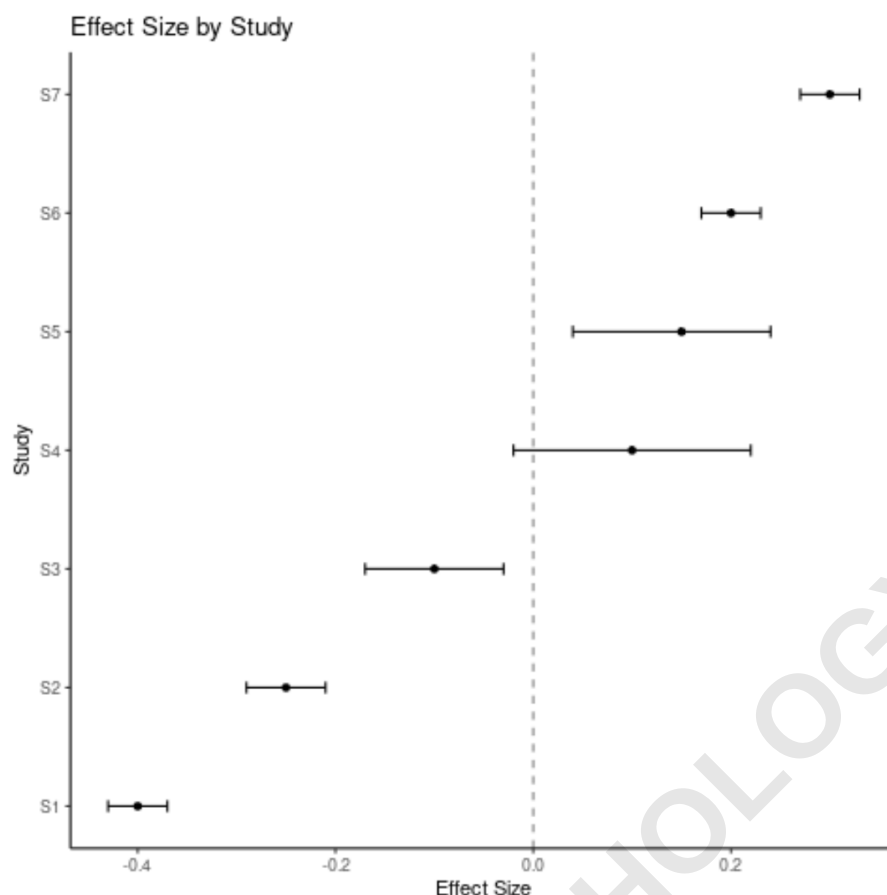
stats writer (2025). *How to Easily Create a Forest Plot in R*. PSYCHOLOGICAL SCALES.
Retrieved from <https://scales.arabpsychology.com/?p=106017>

Generating a forest plot in the statistical environment of R is a straightforward yet powerful technique for visualizing aggregated research results. This comprehensive guide details the necessary steps, from initial data preparation in a data frame to advanced aesthetic customization using established R packages. The procedure involves structuring your input data, defining key summary statistics such as the effect size and associated confidence intervals, and leveraging R's robust visualization libraries to produce a clear and publication-ready graphic.

The creation of an effective forest plot requires careful attention to data organization and the correct application of visualization functions. We will utilize the renowned ggplot2 package, which provides the foundation for building complex and customizable statistical graphics layer by layer. The ultimate goal is to produce a visually compelling chart that accurately summarizes the findings of multiple independent studies.

By following these instructions, researchers and analysts can effectively communicate the findings of a meta-analysis, ensuring transparency and clarity in the synthesis of evidence.

A **forest plot** (sometimes colloquially referred to as a "blobbogram") is fundamentally used in the context of systematic reviews and meta-analyses to visualize the results of several studies simultaneously on a single graph. It provides an immediate visual summary of the pooled estimate, the degree of variability among studies, and the precision of each individual finding.



The structure of the plot is standardized: the horizontal x-axis represents the measurement of interest in the analyzed studies. This value is typically a measure of treatment effect, such as an odds ratio, mean difference, or standardized effect size. Conversely, the vertical y-axis is dedicated to listing the results derived from each individual study included in the analysis.

This graphical representation offers a highly convenient and efficient way for researchers and readers to assimilate the aggregated evidence from multiple research projects, making it a cornerstone tool in evidence-based medicine and social science research.

The Role and Interpretation of Forest Plots

The forest plot serves a dual purpose: first, it displays the point estimate (the calculated effect) for every study, and second, it shows the precision of that estimate via horizontal lines representing the confidence intervals (CIs). The central marker (often a square or circle) indicates the study's calculated effect, while the horizontal line indicates the range within which the true effect is likely to lie. Narrow lines suggest high precision, usually linked to larger sample sizes.

A crucial element in interpreting a forest plot is the line of no effect (often drawn vertically at zero for mean differences or one for odds ratios). If a study's confidence interval crosses this line, the

study result is considered statistically non-significant at the chosen alpha level. The plot also typically includes a summary estimate, often represented by a diamond shape, which reflects the pooled effect size across all included studies. The width of this diamond represents the confidence interval of the combined effect.

Understanding these visual components is paramount. For instance, if the pooled diamond is entirely on one side of the null line, it suggests a robust, significant overall effect. If the individual study estimates show wide variation, it may indicate heterogeneity, which requires further investigation and potentially more complex meta-analytic models.

Prerequisites: Setting up R and the Data Structure

Before initiating the plotting process, ensure you have the R environment installed and that the necessary visualization package, `ggplot2`, is available and loaded. While other packages exist for forest plots (like `metafor` or `forestplot`), using `ggplot2` offers maximum flexibility for aesthetic control.

The input data must be structured as a data frame. This data frame must contain, at a minimum, four key columns for each study: a unique identifier for the study, the calculated point estimate (e.g., the effect), and the upper and lower bounds of the corresponding confidence interval (CI).

Organizing the data consistently ensures that R can correctly map the variables to the visual elements of the plot. Errors in data structure, such as mismatched indices or incorrectly specified bounds, will lead to inaccurate graphical representations. The example below demonstrates the standard structure needed for our visualization.

Example: Preparing the Input Data Frame in R

To successfully generate a forest plot in R, we must first establish a structured data frame. This structure is essential to hold the primary measure of interest, often the effect size, alongside the associated lower and upper bounds of the confidence interval for each study being synthesized.

In this illustrative example, we create a data frame named `df` containing seven hypothetical studies (S1 through S7). The `effect` column represents the central point estimate, while `lower` and `upper` define the range of uncertainty around that estimate. The `index` column is included primarily to provide a numerical position for plotting the studies along the y-axis.

```
#create data
```

```
df <- data.frame(study=c('S1', 'S2', 'S3', 'S4', 'S5', 'S6', 'S7'),  
index=1:7,  
effect=c(-.4, -.25, -.1, .1, .15, .2, .3),
```

```
lower=c(-.43, -.29, -.17, -.02, .04, .17, .27),
upper=c(-.37, -.21, -.03, .22, .24, .23, .33))
```

```
#view data structure and initial values
```

```
head(df)
```

```
study index effect lower upper
```

```
1 S1 1 -0.40 -0.43 -0.37
```

```
2 S2 2 -0.25 -0.29 -0.21
```

```
3 S3 3 -0.10 -0.17 -0.03
```

```
4 S4 4 0.10 -0.02 0.22
```

```
5 S5 5 0.15 0.04 0.24
```

```
6 S6 6 0.20 0.17 0.23
```

```
7 S7 7 0.30 0.27 0.33
```

This data frame preparation is the backbone of the visualization. Note how the `lower` and `upper` columns define the interval boundaries. For S4, the interval crosses zero (from -0.02 to 0.22), suggesting a non-significant finding for that individual study.

Step-by-Step Visualization using ggplot2

Once the data is correctly structured, we can utilize the power of the `ggplot2` data visualization package to construct the initial forest plot visualization. `ggplot2` follows a grammar of graphics approach, allowing us to build the plot by adding layers that define data mapping, geometric objects, and scaling.

The first step is loading the library and then initiating the plot structure by mapping the aesthetics (`aes`) to our data variables. We map `index` to the y-axis (study order), `effect` to the central x-axis point, and `lower` and `upper` to define the error bars.

```
#load ggplot2 library
```

```
library(ggplot2)
```

```
#create base forest plot structure
```

```
ggplot(data=df, aes(y=index, x=effect, xmin=lower, xmax=upper)) +
```

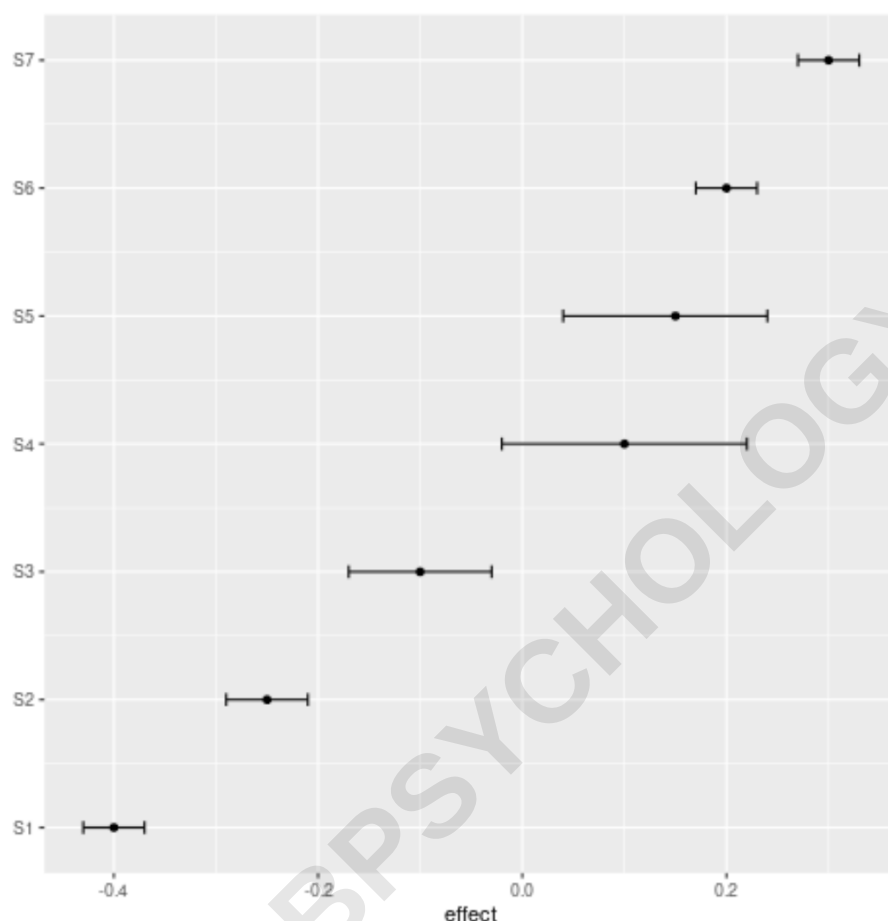
```
geom_point() +
```

```
geom_errorbarh(height=.1) +
```

```
scale_y_continuous(name = "", breaks=1:nrow(df), labels=df$study)
```

This code block performs three critical functions: `geom_point()` plots the central point estimate

(the dot) for each study; `geom_errorbarh()` creates the horizontal bars representing the confidence interval; and `scale_y_continuous()` is used to customize the y-axis, ensuring that the numerical index values are replaced by the more informative study names (S1, S2, etc.). The `height=.1` argument controls the size of the vertical caps on the error bars, improving visual separation.



Understanding the Initial Plot Output

In the resulting visualization, the x-axis precisely displays the magnitude of the effect size calculated for each study, providing a standardized scale for comparison. The y-axis, which we customized using `scale_y_continuous`, clearly displays the abbreviated name or identifier for each study in the series.

The points scattered across the plot represent the precise point estimate--the single most likely value--of the effect size determined by that particular study. Extending horizontally from these points, the error bars delineate the bounds of the confidence interval. The length of the bar directly communicates the statistical precision of the study: shorter bars indicate higher precision (usually

due to larger samples or lower variance).

While the initial plot is functional, it lacks critical contextual elements necessary for professional reporting. Specifically, the absence of a line of no effect makes it difficult to immediately assess the significance of each finding. Furthermore, clearer labels and a descriptive title are essential for proper interpretation, leading us to the next stage of refinement.

Enhancing Visualization Aesthetics and Clarity

To transition from a basic graphic to a publication-quality forest plot, it is necessary to add key aesthetic elements. Specifically, we must incorporate a title for clear communication, modify the axis labels to be descriptive, and, most importantly, introduce a vertical reference line at the null hypothesis value (zero, in the case of effect differences). This vertical line is crucial for interpreting statistical significance.

The `geom_vline()` function is used to insert the reference line, set to `xintercept=0`. We further enhance the visual appeal and professionalism by applying a predefined theme, such as `theme_minimal()`, which cleans up the background elements and grid lines.

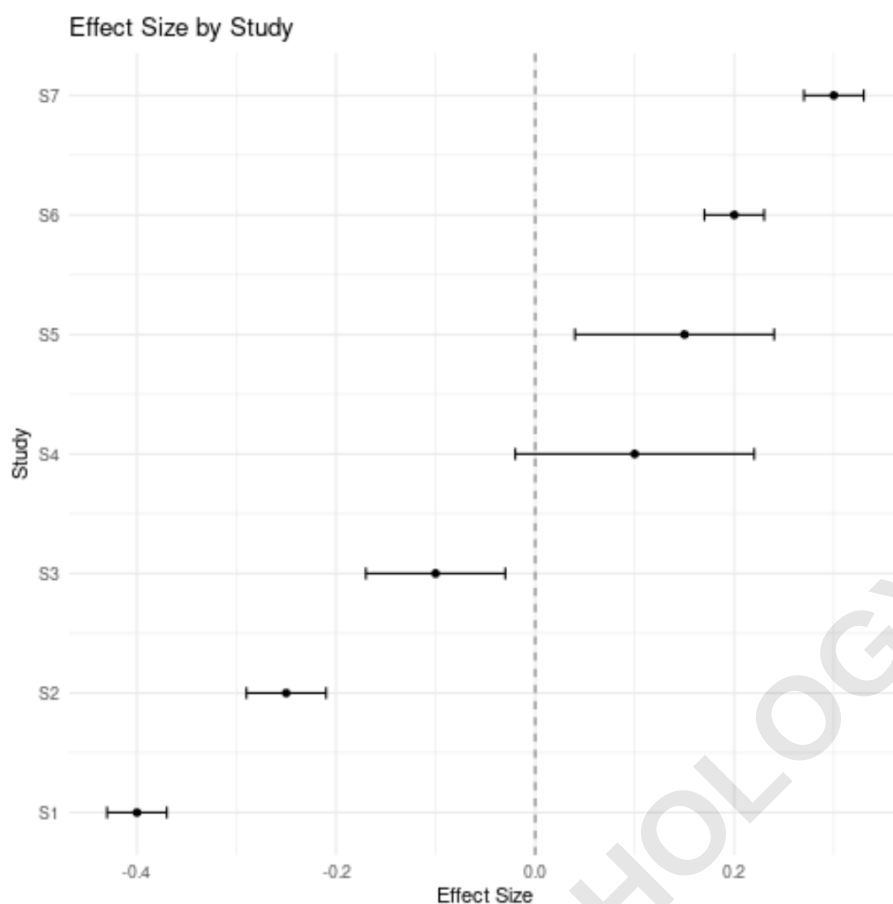
```
#load ggplot2
```

```
library(ggplot2)
```

```
#create forest plot with enhanced aesthetics
```

```
ggplot(data=df, aes(y=index, x=effect, xmin=lower, xmax=upper)) +  
geom_point() +  
geom_errorbarh(height=.1) +  
scale_y_continuous(breaks=1:nrow(df), labels=df$study) +  
labs(title='Effect Size by Study', x='Effect Size', y = 'Study') +  
geom_vline(xintercept=0, color='black', linetype='dashed', alpha=.5) +  
theme_minimal()
```

The `labs()` function allows simultaneous modification of the title and axis labels, greatly improving the plot's informational value. The vertical line at zero, stylized as dashed and semi-transparent (`alpha=.5`), clearly delineates positive versus negative effects.



Alternative Theme Customization

Researchers have significant latitude to modify the visual theme of the plot to match specific journal requirements or personal preference. The modular nature of `ggplot2` means that the underlying data and geometry layers remain consistent, while the aesthetic appearance can be altered by simply substituting the theme function. For example, we could replace `theme_minimal()` with **`theme_classic()`** for a cleaner, more traditional academic appearance that removes most background elements and retains only the axis lines.

#load ggplot2

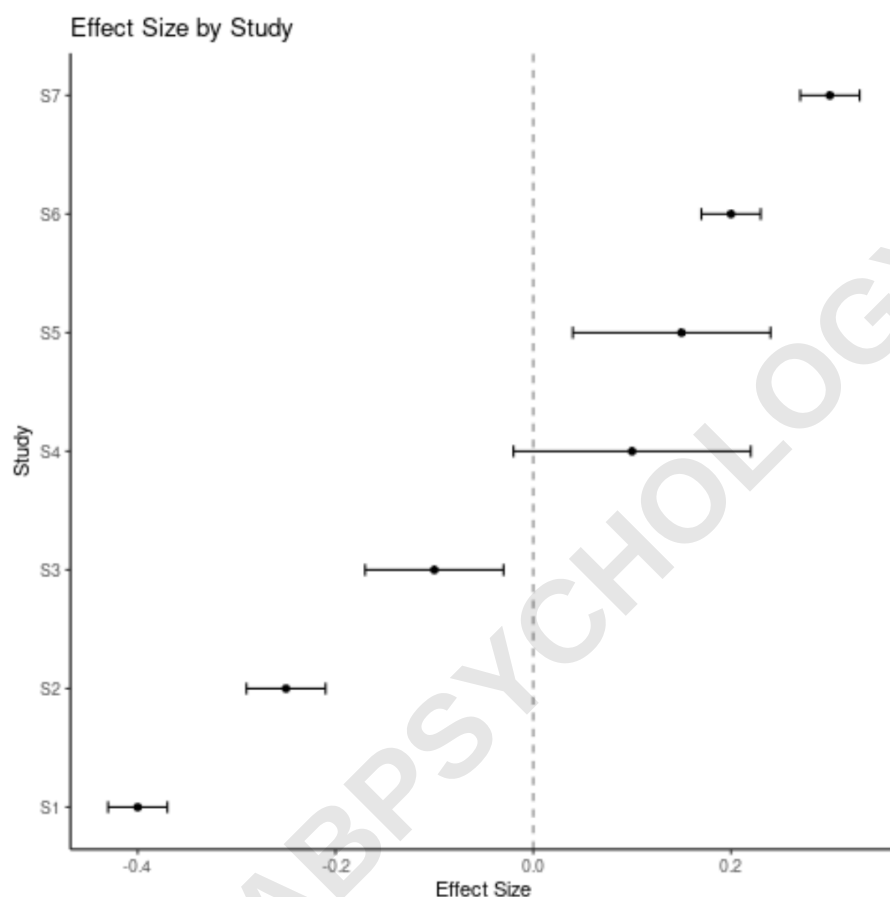
library(ggplot2)

#create forest plot using theme_classic()

```
ggplot(data=df, aes(y=index, x=effect, xmin=lower, xmax=upper)) +
  geom_point() +
  geom_errorbarh(height=.1) +
  scale_y_continuous(breaks=1:nrow(df), labels=df$study) +
  labs(title='Effect Size by Study', x='Effect Size', y = 'Study') +
```

```
geom_vline(xintercept=0, color='black', linetype='dashed', alpha=.5) +
theme_classic()
```

The resulting plot retains all structural integrity--the points, the error bars, and the line of no effect--but presents a more subdued visual frame. This flexibility allows the researcher to control exactly how the data is framed visually, ensuring compliance with diverse publication standards.



Feel free to modify the theme of the plot extensively to align it with specific requirements or desired aesthetics. R and `ggplot2` offer hundreds of options for color palettes, font types, and element sizes.

Advanced Customization: Adding Summary Data

A key feature often missing from a simple forest plot construction is the aggregated summary estimate, typically shown at the bottom. To include this, the data frame must first be augmented with a row containing the pooled effect size and its confidence interval, calculated via a separate meta-analytic function (e.g., using `metafor::rma()`).

Once calculated, this summary row must be added to the data frame. We would then use a specialized geometric object, often `geom_polygon()` or customized `geom_point()` shapes (like a diamond), to visually distinguish the pooled result from the individual study results. This step is critical for a complete meta-analysis presentation, as it provides the overall conclusion derived from the synthesis of evidence.

Furthermore, annotations can be added using `geom_text()` to display the raw numerical data--such as sample size, exact effect estimate, and CI range--directly adjacent to the plot. This provides readers with the necessary precision without having to refer back to tables, further enhancing the informational density of the final graphic.

ARABPSYCHOLOGY.COM