

How to Easily Create Two-Way Tables in R

Authored by
stats writer

December 6, 2025

RECOMMENDED CITATION

stats writer (2025). *How to Easily Create Two-Way Tables in R*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=106417>

In statistical analysis and data visualization, understanding the relationship between different variables is crucial. When dealing with categorical variables, the most effective tool for summarizing and analyzing these relationships is the two-way table, often referred to as a contingency table. This fundamental structure allows analysts to display the joint distribution of frequencies for two such variables, providing immediate insights into patterns and dependencies within the dataset.

A **two-way table** essentially organizes data based on two dimensions. One variable dictates the rows, and the other dictates the columns. The intersections, or cells, of the table contain the count or frequency of observations that fall into both categories simultaneously. This structure is immensely useful across fields ranging from sociology and market research to epidemiology, where classifying outcomes based on two criteria is standard practice. Learning how to efficiently generate and manipulate these tables is a core skill for any user of the R programming language.

Consider a classic survey scenario involving 100 participants. The survey aims to determine preferred sports (Baseball, Basketball, Football) categorized by the respondent's gender (Male, Female). The resulting structure, as shown below, perfectly illustrates the utility of a two-way table. The table clearly shows the frequency distribution: 13 males preferred baseball, 23 females preferred baseball, and so on. This immediate visualization of joint frequencies is why the two-way table remains a cornerstone of exploratory data analysis.

	Baseball	Basketball	Football	Total
Male	13	15	20	48
Female	23	16	13	52
Total	36	31	33	100

This comprehensive tutorial delves into the practical methods for constructing, analyzing, and visualizing two-way tables using R. We will explore methods for building tables from raw data or matrices and demonstrate essential operations like calculating marginal sums and creating graphical representations.

Example 1: Create a Two Way Table from Scratch

When working with summarized or aggregated data, it is often necessary to construct a two-way table directly from raw frequency counts. The most straightforward approach in R involves using a matrix structure and then converting it into a table object suitable for statistical analysis. This method bypasses the need for raw, observation-level data, which is highly efficient when the counts are already known.

The core function used for this conversion is `as.table()`. Before invoking this function, the data must first be organized into a standard R matrix using the `matrix()` function. It is absolutely crucial during this step to ensure the data is entered in the correct order, typically column-wise by default in R, corresponding to the desired cell frequencies. Once the matrix is created, we assign meaningful labels to both the rows and columns using `rownames()` and `colnames()`, respectively, transforming the numerical output into a readable, interpretable table.

The following detailed code demonstrates the process. We define a matrix containing the joint frequencies from our sports preference example. We specify that the data should be organized into three columns (representing the three sports). We then explicitly label the genders as rows and the sports as columns. Finally, we convert this matrix into a proper table object using `as.table()`, preparing it for subsequent operations in R.

```
# Initialize a matrix using the frequency data (inputted column by column)
```

```
data <- matrix(c(13, 23, 15, 16, 20, 13), ncol=3)
```

```
# Specify row names (Gender categories)
```

```
rownames(data) <- c('Male', 'Female')
```

```
# Specify column names (Sport categories)
```

```
colnames(data) <- c('Baseball', 'Basketball', 'Football')
```

```
# Convert the numerical matrix structure into a formal table object
```

```
data <- as.table(data)
```

```
# Display the resultant two-way table
```

```
data
```

```
Baseball Basketball Football
```

```
Male 13 15 20
```

```
Female 23 16 13
```

Example 2: Create a Two Way Table from Data

In most real-world analytical scenarios, data is not presented as a pre-summarized matrix, but rather as raw observations stored within a data frame. When analyzing raw data, the goal is to count how many observations fall into each combination of categories defined by two specific columns. The `table()` function in R is specifically designed for this purpose, serving as the essential tool for deriving two-way tables directly from raw, unaggregated data.

To implement this, we must first define the data frame. This structure holds the individual records, where each row represents one observation (e.g., one respondent), and the columns contain the

values for the categorical variables (e.g., sport and gender). The power of the `table()` function lies in its ability to take two vectors (columns from the data frame) as arguments and automatically cross-tabulate their contents, yielding the desired frequency counts.

In the following demonstration, we create a small sample data frame named `df`. We use the `table()` function, passing the gender column as the first argument (which typically defines the rows) and the sport column as the second argument (defining the columns). The resulting output is a compact two-way table showing the joint frequencies, illustrating how easily R converts raw survey data into a structured analytical tool.

Create a sample data frame with two categorical variables: sport and gender

```
df <- data.frame(sport=c('Base', 'Base', 'Bask', 'Foot', 'Foot'),  
gender=c('Male', 'Female', 'Male', 'Male', 'Female'))
```

```
# View the data frame structure
```

```
df
```

```
# Create the two-way table by cross-tabulating the 'gender' and 'sport' columns
```

```
data <- table(df$gender, df$sport)
```

```
# Display the resulting frequency table
```

```
data
```

```
Base Bask Foot
```

```
Female 1 0 1
```

```
Male 1 1 1
```

Example 3: Calculate Margin Sums of a Two Way Table

Once a two-way table has been constructed, analysts often need to calculate the totals for each row and each column. These totals are known as **marginal sums** or marginal frequencies. The marginal sums are critical because they represent the total frequency distribution for each individual variable, independent of the other. For instance, the row sums tell us the total number of males and females, irrespective of their sport preference, while the column sums indicate the total counts for each sport, irrespective of gender.

In R, calculating these essential summaries is efficiently achieved using the specialized function `margin.table()`. This function requires two primary arguments: the two-way table object itself, and the `margin` argument, which specifies the dimension along which the summing should occur. Setting `margin=1` instructs R to calculate the row sums, providing the marginal distribution for the row variable. Conversely, setting `margin=2` calculates the column sums, providing the marginal

distribution for the column variable.

Using the complex sport preference data from Example 1, the following code demonstrates how to apply `margin.table()` to derive these crucial figures. Understanding marginal sums is often the first step in statistical inference tests, such as the Chi-squared test of independence, which compares observed joint frequencies against the expected frequencies based on these margins.

Re-create the initial matrix (or use the table object from Example 1)

```
data <- matrix(c(13, 15, 20, 23, 16, 13), ncol=3)
```

```
rownames(data) <- c('Male', 'Female')
```

```
colnames(data) <- c('Baseball', 'Basketball', 'Football')
```

```
# Calculate the row sums (Margin=1): Total count for each gender
```

```
margin.table(data, margin=1)
```

```
Male Female
```

```
49 51
```

```
# Calculate the column sums (Margin=2): Total count for each sport
```

```
margin.table(data, margin=2)
```

```
Baseball Basketball Football
```

```
28 43 29
```

Example 4: Visualize Two Way Table Frequencies

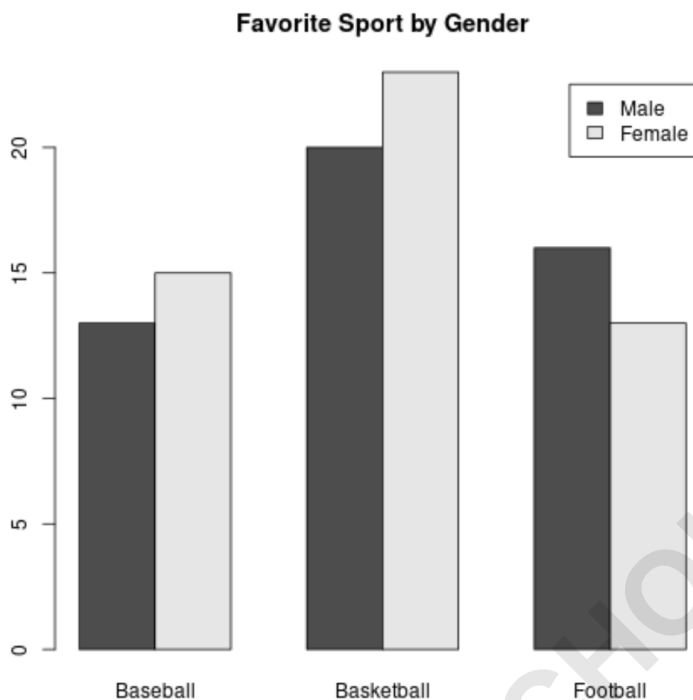
While numerical tables provide precision, visualizing frequency data is often necessary to quickly grasp relationships and comparative magnitudes. R offers several powerful graphical tools specifically suited for displaying the joint distribution found in a two-way table. Two of the most commonly employed visualizations are the grouped bar plot and the specialized mosaic plot.

The first method involves creating a grouped barplot. This visualization uses the `barplot()` function, which automatically recognizes the structured table object and creates adjacent bars for the categories within the second variable, grouped by the categories of the first variable. We include the arguments `legend=True` to identify the groups (e.g., genders) and `beside=True` to ensure the bars are displayed side-by-side rather than stacked. This setup allows for easy comparison of preferences across the different categories (e.g., comparing male versus female preference for basketball).

The code snippet below generates a clear grouped barplot illustrating the distribution of favorite sports across genders. This visualization is excellent for comparing absolute counts and

understanding which specific combinations of categories yield the highest or lowest frequencies, providing an immediate visual interpretation of the data summarized in the two-way table.

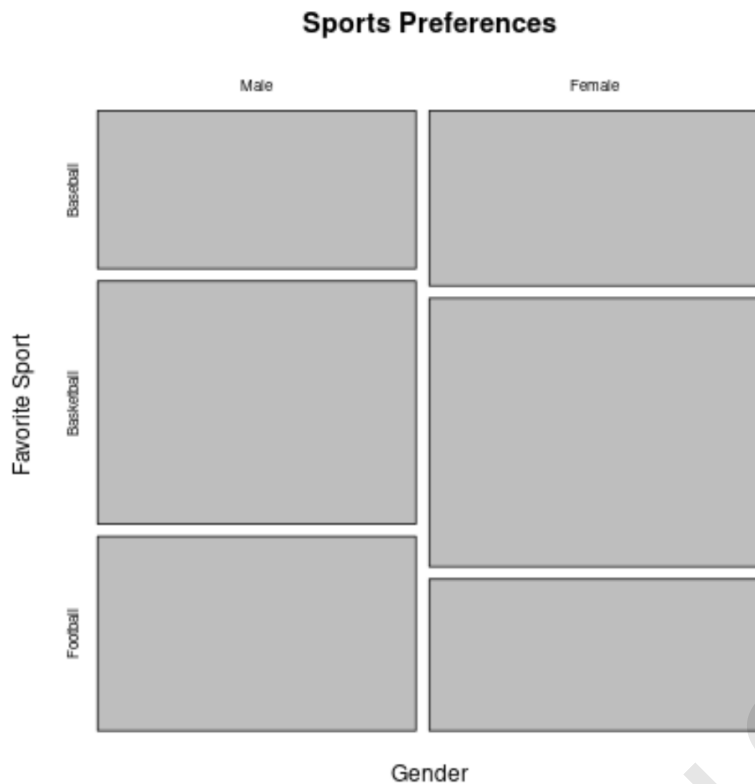
barplot(data, legend=True, beside=True, main='Favorite Sport by Gender')



A second, more sophisticated visualization tool available in R is the mosaic plot. Unlike the bar plot which focuses on absolute counts, the mosaic plot excels at showing conditional proportions and relationships between the two categorical variables. The plot divides a rectangle into segments, where the width of the initial segments corresponds to the marginal frequencies of the first variable (X-axis), and the heights of the resulting sub-segments correspond to the conditional frequencies of the second variable (Y-axis).

The `mosaicplot()` function is extremely powerful for visually testing the relationship between the variables; if the segments appear disproportionate across the categories, it suggests a lack of independence between the row and column variables. The code below applies this function to our sports preference data, generating a visual representation that emphasizes proportional relationships rather than raw frequency counts.

mosaicplot(data, main='Sports Preferences', xlab='Gender', ylab='Favorite Sport')



Conclusion: Mastering Two-Way Table Analysis

The ability to create, manipulate, and visualize two-way tables is foundational to data analysis in R. These tables provide a comprehensive snapshot of the interplay between two categorical variables, offering both quantitative summaries (frequencies and margins) and the basis for advanced statistical modeling.

By utilizing built-in functions like `table()`, `as.table()`, and `margin.table()`, R allows for highly efficient data wrangling, transforming raw observations or summarized counts into structured, analyzable objects. Furthermore, graphical tools such as the barplot and the mosaic plot transform these numerical findings into intuitive visual evidence, enhancing communication of analytical results.

We encourage readers to explore the rich ecosystem of R statistical tutorials available online to further deepen their expertise in data manipulation and statistical inference.

You can find more R tutorials on .