

# How can I use the colMeans() function in R?

Authored by  
**stats writer**

May 5, 2024

## RECOMMENDED CITATION

stats writer (2024). *How can I use the colMeans() function in R?*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=143021>

The `colMeans()` function in R is used to calculate the mean of each column in a given dataset or matrix. It takes in a dataset as an input and returns a vector of means, with each element representing the mean of a specific column in the dataset. This function is particularly useful for analyzing large datasets and identifying patterns or trends within the columns. It can also be used to compare the means of different columns and make data-driven decisions. Overall, the `colMeans()` function is a powerful tool for data analysis and can greatly assist in making informed decisions.

## Use colMeans() Function in R

The `colMeans()` function in R can be used to calculate the mean of several columns of a matrix or data frame in R.

This function uses the following basic syntax:

**#calculate column means of every column**

**`colMeans(df)`**

**#calculate column means and exclude NA values**

**`colMeans(df, na.rm=T)`**

**#calculate column means of specific columns**

**`colMeans(df)`**

The following examples show how to use this syntax in practice.

### Example 1: Calculate Mean of Every Column

The following code shows how to calculate the mean of every column in a data frame:

```
#create data frame
```

```
df <- data.frame(points=c(99, 91, 86, 88, 95),  
assists=c(33, 28, 31, 39, 34),  
rebounds=c(30, 28, 24, 24, 28),  
blocks=c(1, 4, 11, 0, 2))
```

```
#calculate column means
```

```
colMeans(df)
```

```
points assists rebounds blocks  
91.8 33.0 26.8 3.6
```

### Example 2: Calculate Mean of Every Column & Exclude NA's

The following code shows how to calculate the mean of every column and exclude NA values:

```
#create data frame with some NA values
```

```
df <- data.frame(points=c(99, 91, 86, 88, 95),  
assists=c(33, NA, 31, 39, 34),  
rebounds=c(30, 28, NA, NA, 28),
```

```
blocks=c(1, 4, 11, 0, 2))
```

```
#calculate column means
```

```
colMeans(df, na.rm=T)
```

```
points assists rebounds blocks
```

```
91.80000 34.25000 28.66667 3.60000
```

**Example 3: Calculate Mean of Specific Columns**

**The following code shows how to calculate the mean values of specific columns in the data frame:**

```
#create data frame
```

```
df <- data.frame(points=c(99, 91, 86, 88, 95),
```

```
assists=c(33, 28, 31, 39, 34),
```

```
rebounds=c(30, 28, 24, 24, 28),
```

```
blocks=c(1, 4, 11, 0, 2))
```

```
#calculate column means for 'points' and 'blocks'  
columns
```

```
colMeans(df)
```

```
points blocks
```

```
91.8 3.6
```

**Note that we can also use index values to calculate the mean of specific columns:**

```
#create data frame
```

```
df <- data.frame(points=c(99, 91, 86, 88, 95),  
assists=c(33, 28, 31, 39, 34),  
rebounds=c(30, 28, 24, 24, 28),  
blocks=c(1, 4, 11, 0, 2))
```

```
#calculate column means for columns in position 1 and  
4
```

```
colMeans(df)
```

```
points blocks
```

```
91.8 3.6
```

**The following tutorials explain how to perform other common functions in R:**