How can I create a correlation matrix in R?

Authored by

stats writer

June 29, 2024

Creating a correlation matrix in R involves using the "cor" function to calculate the correlation coefficients between different variables in a dataset and then using the "corrplot" package to visualize the results. This process helps to identify relationships and patterns between variables, making it a useful tool for data analysis and decision making. By following the proper steps and using the appropriate functions, a correlation matrix can be easily created in R, providing valuable insights into the data.

Create a Correlation Matrix in R (4 Examples)

A is a square table that shows the between variables in a dataset.

It offers a quick way to understand the strength of the linear relationships that exist between variables in a dataset.

There are four common ways to create a correlation matrix in R:

Method 1: The cor Function (For getting simple matrix of correlation coefficients)

cor(df)

Method 2: The rcorr Function (For getting p-values of correlation coefficients)

library(Hmisc)

```
rcorr(as.matrix(df))
```

Method 3: The corrplot Function (For visualizing correlation matrix)

```
library(corrplot)

corrplot(cor(df))
```

Method 4: The ggcorrplot Function (For visualizing correlation matrix)

```
library(ggcorrplot)

ggcorrplot(cor(df))
```

The following examples show how to use each method with the following data frame in R:

```
#create data frame
df <- data.frame(assists=c(4, 5, 5, 6, 7, 8, 8, 10),
rebounds=c(12, 14, 13, 7, 8, 8, 9, 13),
points=c(22, 24, 26, 26, 29, 32, 20, 14))

#view data frame
```

```
df

assists rebounds points
1 4 12 22
2 5 14 24
3 5 13 26
4 6 7 26
5 7 8 29
6 8 8 32
7 8 9 20
8 10 13 14
```

Example 1: The cor Function

We can use the cor() function from base R to create a correlation matrix that shows the correlation coefficients between each variable in our data frame:

```
#create correlation matrix
cor(df)

assists rebounds points
assists 1.0000000 -0.2448608 -0.3295730
rebounds -0.2448608 1.0000000 -0.5220917
points -0.3295730 -0.5220917 1.0000000
```

The correlation coefficients along the diagonal of the table are all equal to 1 because each variable is perfectly correlated with itself.

The correlation coefficient between assists and rebounds is -0.245.The correlation coefficient between assists and points is -0.330.The correlation coefficient between rebounds and points is -0.522.

Example 2: The rcorr Function

We can use the rcorr() function from the Hmisc package in R to create a correlation matrix that shows the correlation coefficients between each variable in our data frame:

```
library(Hmisc)

#create matrix of correlation coefficients and p-values
rcorr(as.matrix(df))

assists rebounds points
assists 1.00 -0.24 -0.33
rebounds -0.24 1.00 -0.52
points -0.33 -0.52 1.00
```

n= 8

P

assists rebounds points

assists 0.5589 0.4253

rebounds 0.5589 0.1844

points 0.4253 0.1844

The first matrix shows the correlation coefficients between the variables and the second matrix shows the corresponding p-values.

For example, the correlation coefficient between assists and rebounds is -0.24 and the p-value for this correlation coefficient is 0.5589.

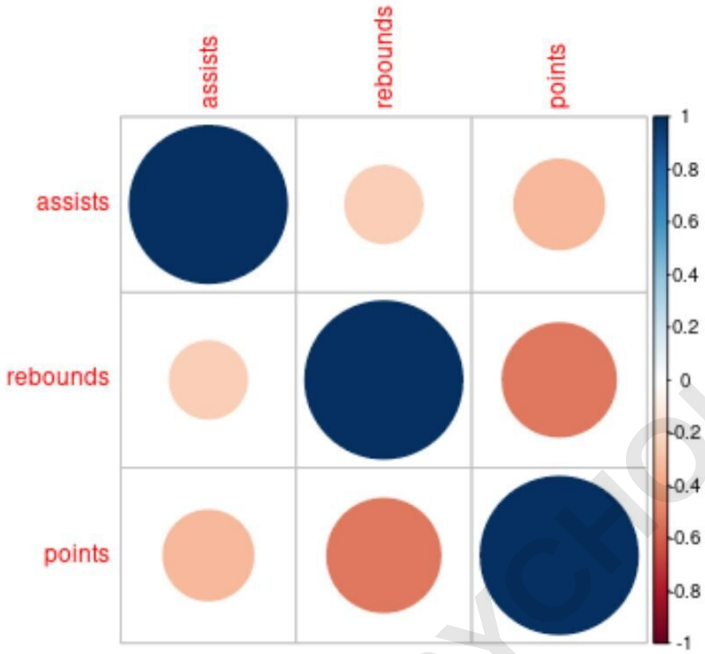
This tells us that the correlation between the two variables is negative but it's not a statistically significant correlation since the p-value is not less than .05.

Example 3: The corrplot Function

We can use the corrplot() function from the corrplot package in R to visual the correlation matrix:

```
library(corrplot)

#visualize correlation matrix
corrplot(cor(df))
```



The color and size of the circles in the correlation matrix help us visualization the correlations between each variable.

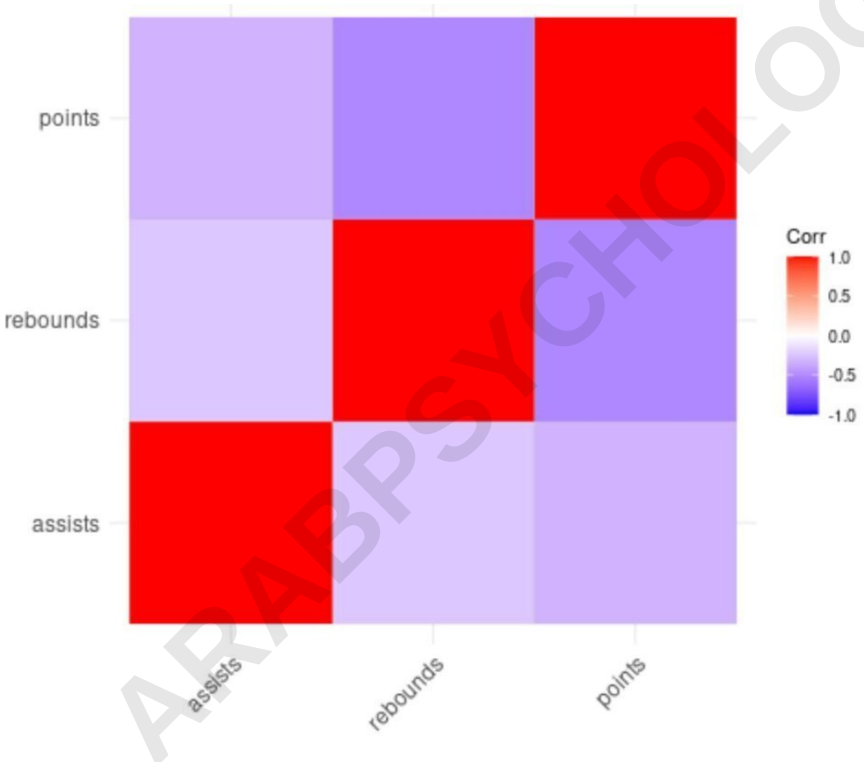
For example, the circle where the assists and rebounds variables intersect is small and light red, which tells us that the correlation is low and negative.

Example 4: The corrplot Function

We can use the ggcorrplot() function from the ggcorrplot package in R to visualize the correlation matrix:

```
library(ggcorrplot)

#visualize correlation matrix
ggcorrplot(cor(df))
```



The color of the squares in the correlation matrix help us visualization the correlations between each variable.

Additional Resources

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