

How can I filter a dataset by date using dplyr?

Authored by
stats writer

June 29, 2024

RECOMMENDED CITATION

stats writer (2024). *How can I filter a dataset by date using dplyr?*. PSYCHOLOGICAL SCALES. Retrieved from <https://scales.arabpsychology.com/?p=160333>

Dplyr is a popular R package used for data manipulation, and it offers a variety of functions for filtering data. To filter a dataset by date using dplyr, one can use the filter function and specify the desired date range using the "between" operator. This allows for the selection of specific rows in the dataset that fall within the specified date range. Additionally, the "arrange" function can be used to sort the filtered dataset by date. This process enables efficient and accurate filtering of datasets based on date criteria, making it a valuable tool for data analysis and management.

Filter by Date Using dplyr

You can use the following methods to filter a data frame by dates in R using the package:

Method 1: Filter Rows After Date

```
df %>% filter(date_column > '2022-01-01')
```

Method 2: Filter Rows Before Date

```
df %>% filter(date_column < '2022-01-01')
```

Method 3: Filter Rows Between Two Dates

```
df %>% filter(between(date_column, as.Date('2022-01-20'), as.Date('2022-02-20')))
```

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

```
#create data frame
```

```
df <- data.frame(day=seq(as.Date('2022-01-01'), by =  
'week', length.out=10),  
sales=c(40, 35, 39, 44, 48, 51, 23, 29, 60, 65))
```

```
#view data frame
```

```
df
```

```
day sales
```

```
1 2022-01-01 40  
2 2022-01-08 35  
3 2022-01-15 39  
4 2022-01-22 44  
5 2022-01-29 48  
6 2022-02-05 51  
7 2022-02-12 23  
8 2022-02-19 29  
9 2022-02-26 60  
10 2022-03-05 65
```

Example 1: Filter Rows After Date

We can use the following code to filter for the rows in the data frame that have a date after 1/25/2022:

```
library(dplyr)
```

```
#filter for rows with date after 1/25/2022
```

```
df %>% filter(day > '2022-01-25')
```

```
day sales
```

```
1 2022-01-29 48
```

```
2 2022-02-05 51
```

```
3 2022-02-12 23
```

```
4 2022-02-19 29
```

```
5 2022-02-26 60
```

```
6 2022-03-05 65
```

Each of the rows in the resulting data frame have a date after 1/25/2022.

Example 2: Filter Rows Before Date

We can use the following code to filter for the rows in the data frame that have a date before 1/25/2022:

```
library(dplyr)
```

```
#filter for rows with date before 1/25/2022
```

```
df %>% filter(day < '2022-01-25')
```

day sales

```
1 2022-01-01 40
2 2022-01-08 35
3 2022-01-15 39
4 2022-01-22 44
```

Each of the rows in the resulting data frame have a date before 1/25/2022.

Example 3: Filter Rows Between Two Dates

We can use the following code to filter for the rows in the data frame that have a date between 1/20/2022 and 2/20/2022:

```
library(dplyr)
```

```
#filter for rows with dates between 1/20/2022 and 2/20/2022
```

```
df %>% filter(between(date_column, as.Date('2022-01-20'), as.Date('2022-02-20')))
```

day sales

```
1 2022-01-22 44
2 2022-01-29 48
3 2022-02-05 51
```

4 2022-02-12 23

5 2022-02-19 29

Each of the rows in the resulting data frame have a date between 1/20/2022 and 2/20/2022.

Note #1: If any of the methods above don't work, then you may need to first convert the dates you're working with to a recognizable date format using the `as.Date()` function.

Note #2: You can find the complete documentation for the filter function in dplyr .

Additional Resources

The following tutorials explain how to perform other common operations in dplyr: