How can I identify the first and last observations within a group in R?

Authored by

stats writer

June 30, 2024

To identify the first and last observations within a group in R, we can use the "group_by" function from the "dplyr" package. This function allows us to group the data by a specific variable and then use the "summarise" function to calculate summary statistics, such as the first and last observation, within each group. Alternatively, we can also use the "slice" function to subset the data and select the first and last rows within each group. Both of these methods can help us accurately identify the first and last observations within a group in R.

How can I identify the first and last observations within a group in R? | R FAQ

Identifying the first and/or last observation within a group is often an important step in data management. For instance, a dataset may contain medical discharge records for patients in which you are interested in each patient's earliest visit. You would then be interested in the first observation within a patient ID after sorting by date. Alternatively, you may be interested in their total medical expenses. In that case, you might create a running total and you would be interested in the last observation within a patient ID after sorting by date.

In SPSS and SAS, you can do this with sorts and first/last options. To do

this in R, we first order the data and then use the by command. The by
command will effectively subset our data based on indicated variables and return
an indicated number of observations from the beginning or end ("head" or "tail")
of that subset.

We will look at the hsb2 dataset. We would like to see the highest and
lowest math scores (math) within each prog. To do this, we first
order our data by prog and math.

hsb2

To
get the observation with the highest math value within each prog, we use the
by command, indicating our dataset first, then our "by" variable for
subsetting, and then which end of the subset and how many from that end. Since
the ordering within prog goes from small to large, we

want the "tail" for
highest values and "head" for lowest values.

highest
"by"

highest
hsb2.s$prog: 1
id female race ses schtyp prog read write math science socst
46 169 0 4 1 1 1 55 59 63 69 46
--
hsb2.s$prog: 2
id female race ses schtyp prog read write math science socst
37 200 0 4 2 2 2 68 54 75 66 66
--
hsb2.s$prog: 3
id female race ses schtyp prog read write math science socst
22 143 0 4 2 1 3 63 63 75 72 66

We can see that our result is a "by" object that contains

the observations of
interest, but not in a dataset form. We can convert these by objects into data
frames. We will do so using do.call, a command to which you supply a
function and the arguments you wish to pass to that function.

highestd

id female race ses schtyp prog read write math science socst
1 169 0 4 1 1 1 55 59 63 69 46
2 200 0 4 2 2 2 68 54 75 66 66
3 143 0 4 2 1 3 63 63 75 72 66

lowestd

id female race ses schtyp prog read write math science socst
1 167 0 4 2 1 1 63 49 35 66 41
2 128 0 4 3 1 2 39 33 38 47 41
3 2 1 1 2 1 3 39 41 33 42 41

The do.call command is a fast way to execute code that

might be tedious or error-prone if you wrote it yourself. For example,
we could have generated highestd with

rbind(highest], highest], highest])

but it was faster and easier to let do.call do this for us.

If you're working with a large dataset...

The above code does not work well with large datasets. However, you can still get the first observation from each group without the sorting steps.
In the code below, you first determine the unique ID values and the subset your
data and then can take the last (or first) observation from the subset and
combine them into a new dataset. We present this with a very small dataset, but
this approach scales up to very large datasets well.

mydata 1) {
last.temp

id1 id2 var1 var2
2 1 11 3 4

3 2 12 2 3

5 3 45 4 7

7 5 34 5 7

References

Muenchen, R.A. R for SAS and SPSS Users. Springer, 2009.