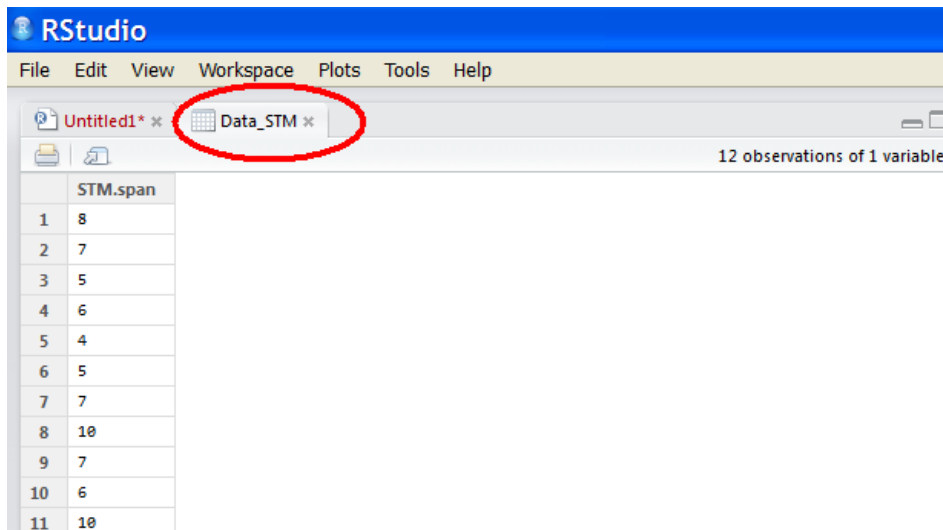


2d: How to make a frequency distribution table in RStudio

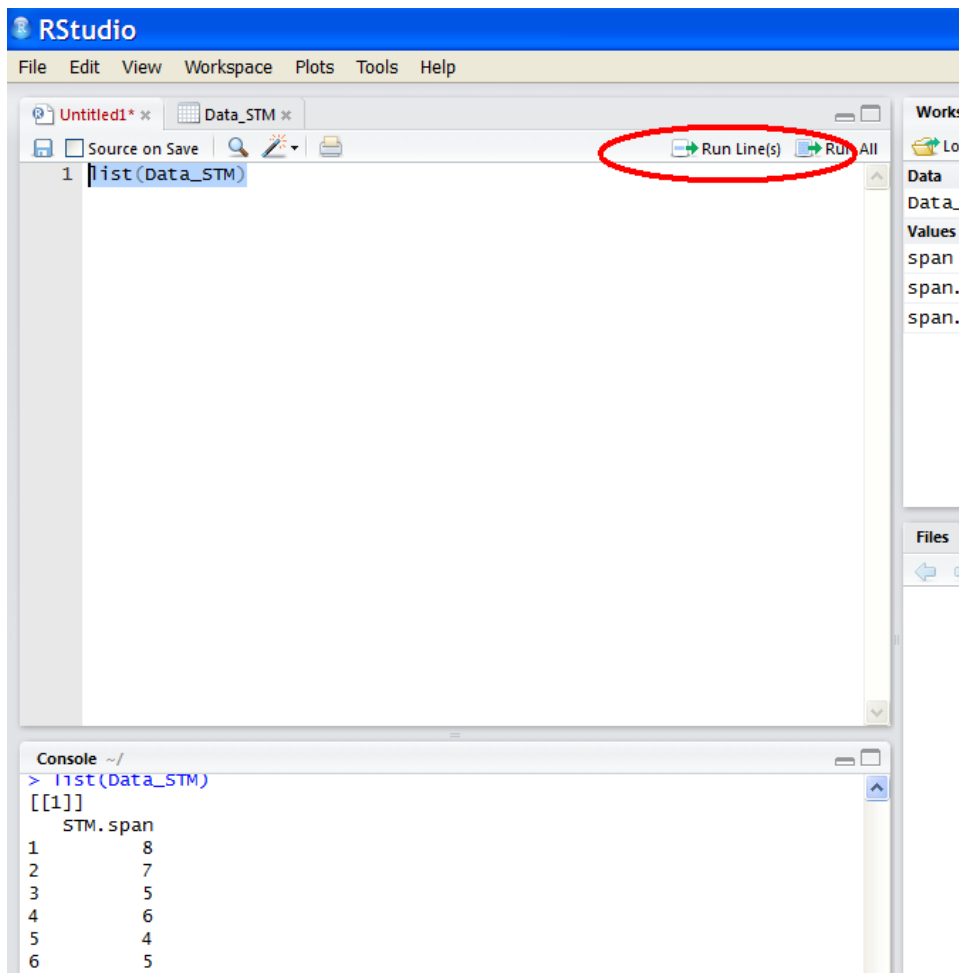
I assume you have the data of the short-term memory study entered in RStudio (if not see [here](#)).

If you have done this, you find the file Data_STM in the upper left panel.



Enter the workspace by opening a new **Rscript file** (click on the option File) or by clicking on the option **Untitled1** in the left upper panel.

To make sure the data are well entered, write **list(Data_STM)**, activate the line and click on **Run Line(s)**.



This should show you the data of the STM-experiment in the lower left panel.

- Making a frequency distribution table at first sight looks more complicated in R than in other programs. The reason for this, however, is that you use the same commands for the simplest table as for the more complicated. So, the commands are slightly harder to start with, but remain the same to make all types of tables, included grouped frequency distribution tables. So, bear with it!
- First, you have to define the spans you are interested in. Suppose we are interested in all spans from 3 to 11. This is defined by the command:

span = seq(3, 11, by =1)

- This commands defines the lower values of the intervals as the sequence of values from 3 to 11 with a step function of 1 (i.e., it will go from 3 to 4, to 5, etc). If we define by=2, then the lower values of the spans will go from 3 to 5, to 7, and so on, which means that we will have a grouped frequency distribution table.
- Next we segment the continuum of values in `Data_STM` according to the spans we defined above. We do this with the following command:

span.cut = cut(Data_STM\$STM.span, span, right=FALSE)

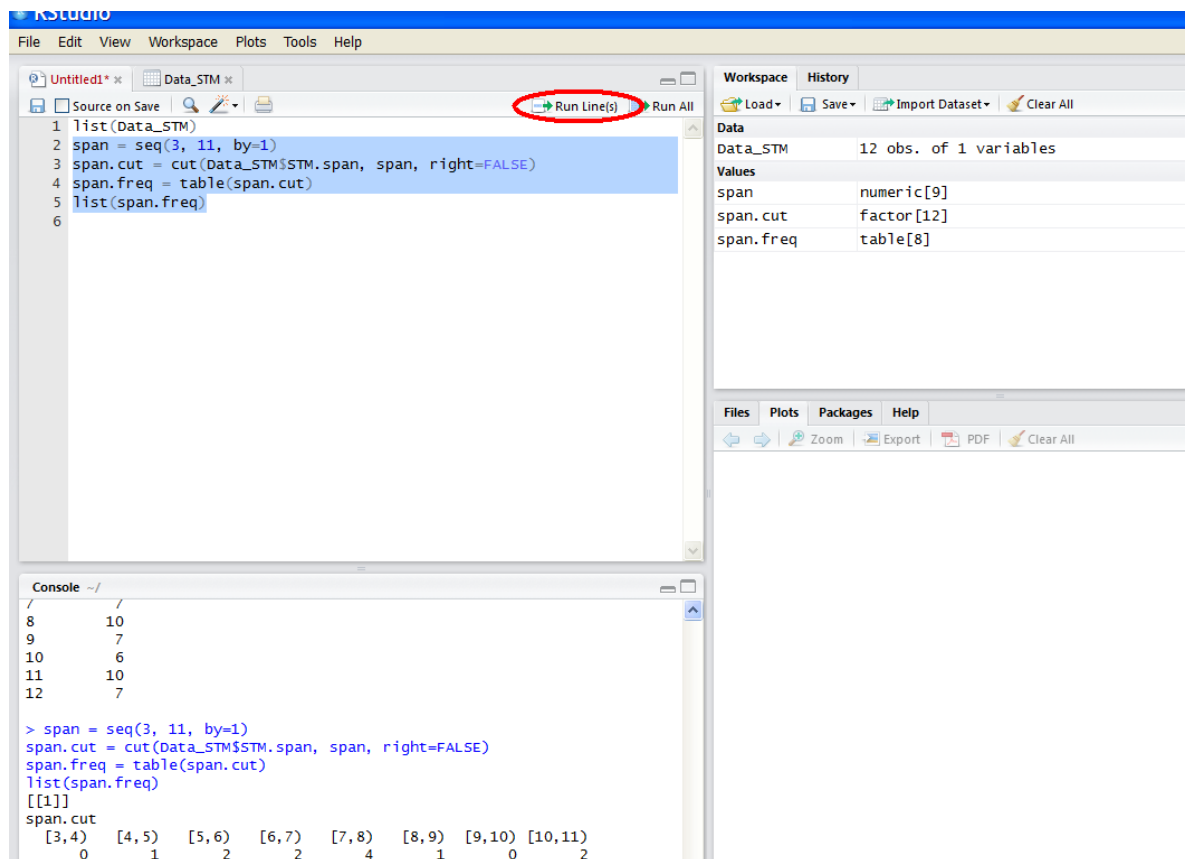
- This commands cuts the values of the variable `STM.span` from the dataset `Data_STM` into parts starting with the lower value defined in `span`, and ending with the upper value immediately below the lower value of the next span, and puts it into a new variable `span.cut`. The following parts are important:

Data_STM\$STM.span

- This is the variable `STM.span` from the dataset `Data_STM`. Notice that we must use uppercase if these have been used in the names. Also notice that R has rewritten our original name `STM_span` into `STM.span` (as you see when you listed the variable, or when you looked at `Data_STM` in the upper left panel). Variables from a dataset are defined as `dataset$variable` (i.e., with a `$` sign between the name of the dataset and the name of the variable).
- **Span** refers to the lower values of the intervals we defined above
- **right=FALSE** makes that the lower value of the next interval will be excluded from the present interval. If we had not done this, the interval 3-4 would exclude the number 3 and include the number 4. Now the interval goes from 3.0 to 3.99999....
- To calculate the frequencies of the intervals (STM-spans) we use the command:

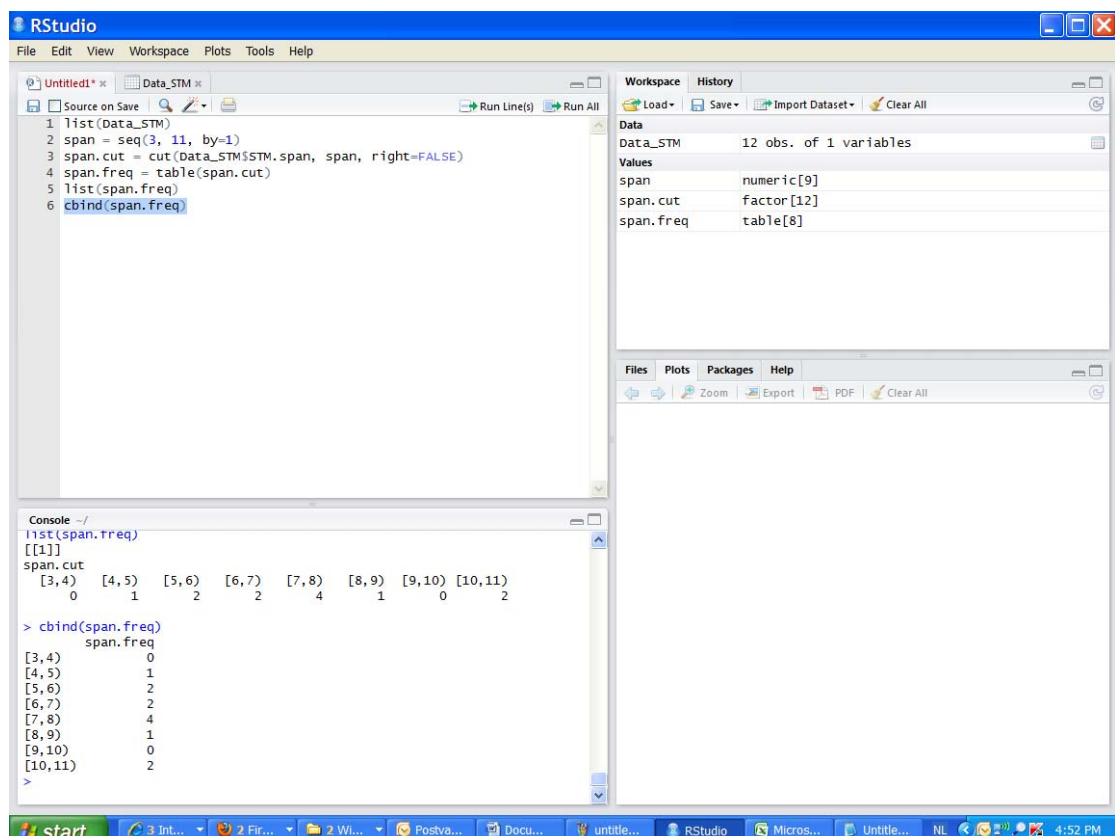
span.freq = table(span.cut)

To see the outcome, we can use **list(span.freq)**, which will give us the following outcome:



In this panel you can see that first the four commands have been entered in the left upper panel. Then these lines have been activated and the command Run Line(s) activated. The lower left panel shows the outcome. Again, the output is a bit more difficult to read, because it uses a notation that can be generalized to other situations. The heading [3,4) refers to the interval of all real values from 3 (included) up to 4 (excluded).

To get the outcome in table form you can work with the command **cbind(span.freq)** rather than **list(span.freq)**. Then the outcome looks as follows:



The initial cost is offset when see how easy it is to get a grouped frequency distribution table in R. Just use by=2 instead of by=1 and you have a table with interval width of 2.

