**Guide on the installation and usage of the QC workflow for quantitative proteomics data**

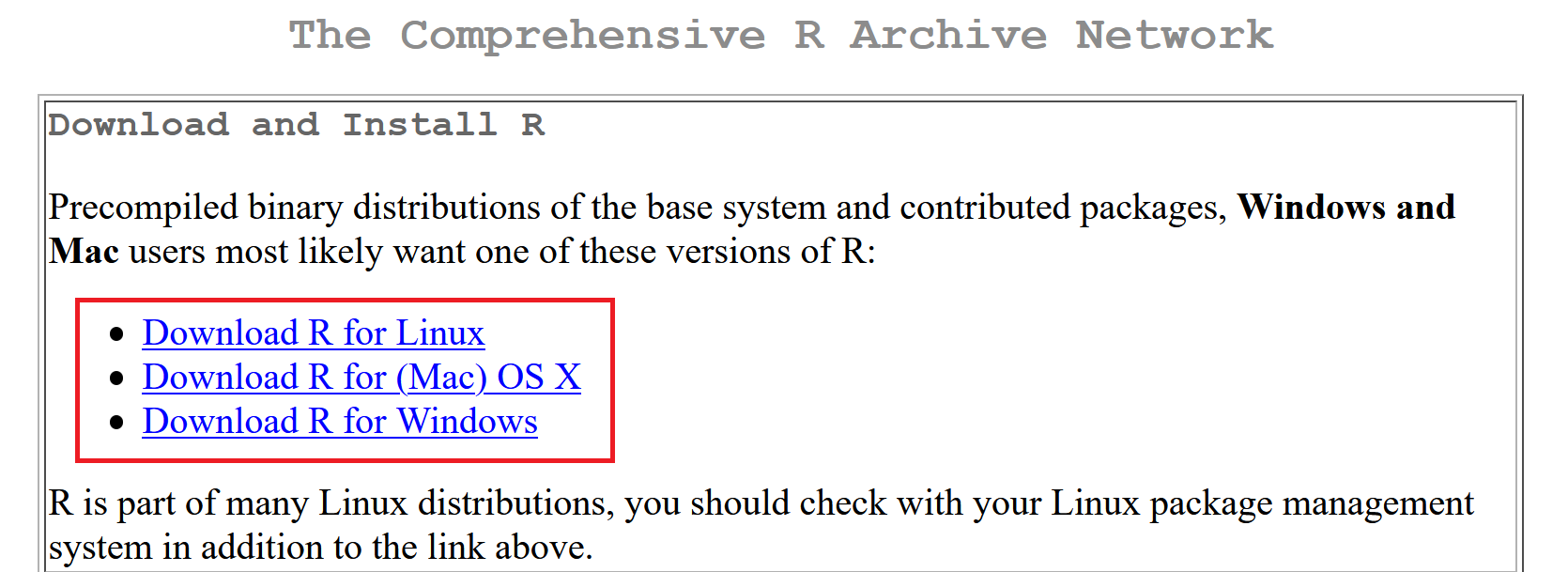
**Summary:**

1. Download and install R: <https://cran.r-project.org/>
2. Download and install RStudio: <https://rstudio.com/products/rstudio/download/>
3. Download the QC workflow folder from github: <https://github.com/mpc-bioinformatics/QC_Quant>
4. Open main script & install necessary R packages
5. Prepare data set
6. Change settings
7. Run the workflow
8. Advanced settings

**Detailed tutorial:**

**1. Download and Install R**

Visit <https://cran.r-project.org/> and select an R version depending on your operating system as it is shown in the image below. Then, clicking “install R for the first time” (“base” does the same) will lead you to the site where you can choose to download the latest R version at the top. Once the download has completed, run the “.exe” file which will start the R installer. You can always use the options that are selected by default.

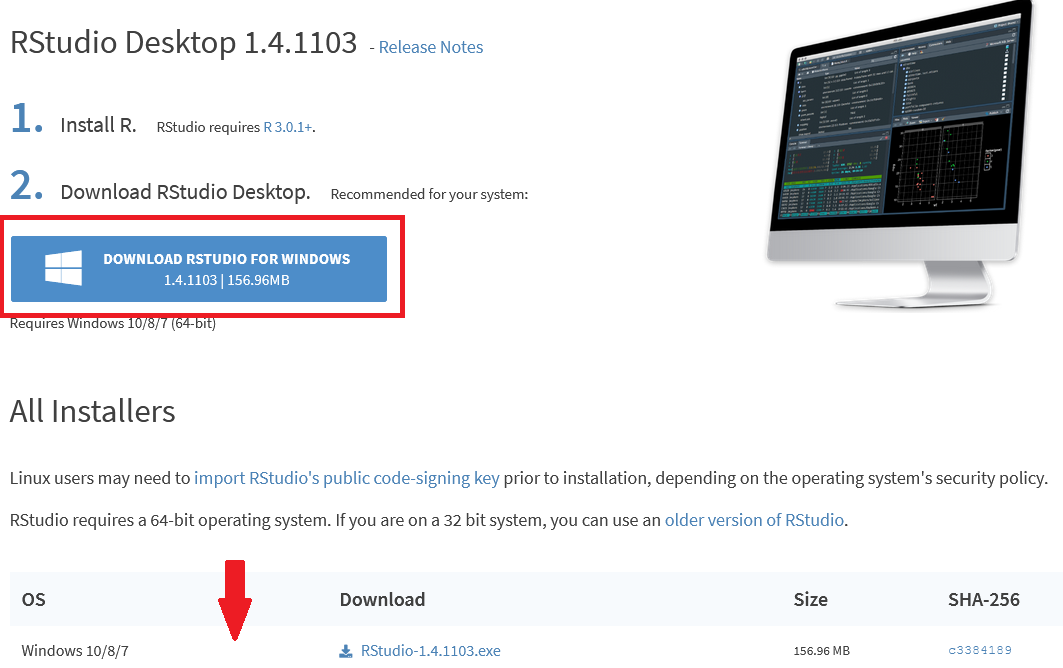


R Website: <https://cran.r-project.org/>

**2. Download and install RStudio**

On <https://rstudio.com/products/rstudio/download/>, click the download button for the free “Rstudio Desktop” application. It leads you to the download area, where the correct version for your operating system should automatically have been chosen, as shown in the picture below. You can decide to choose another version farther below, should the recommended one not match your OS.

After the successful download, run the file and install Rstudio. Again, the default options will work for the usage of the TMT app.



RStudio Website: <https://rstudio.com/products/rstudio/download/>

**3. Download the QC workflow folder from github**

Go to the github page <https://github.com/mpc-bioinformatics/QC_Quant> .

Just download the entire folder, which includes all files and subfolders, by clicking on Code -> Download ZIP. Un-Zip the folder, e.g. using 7Zip.

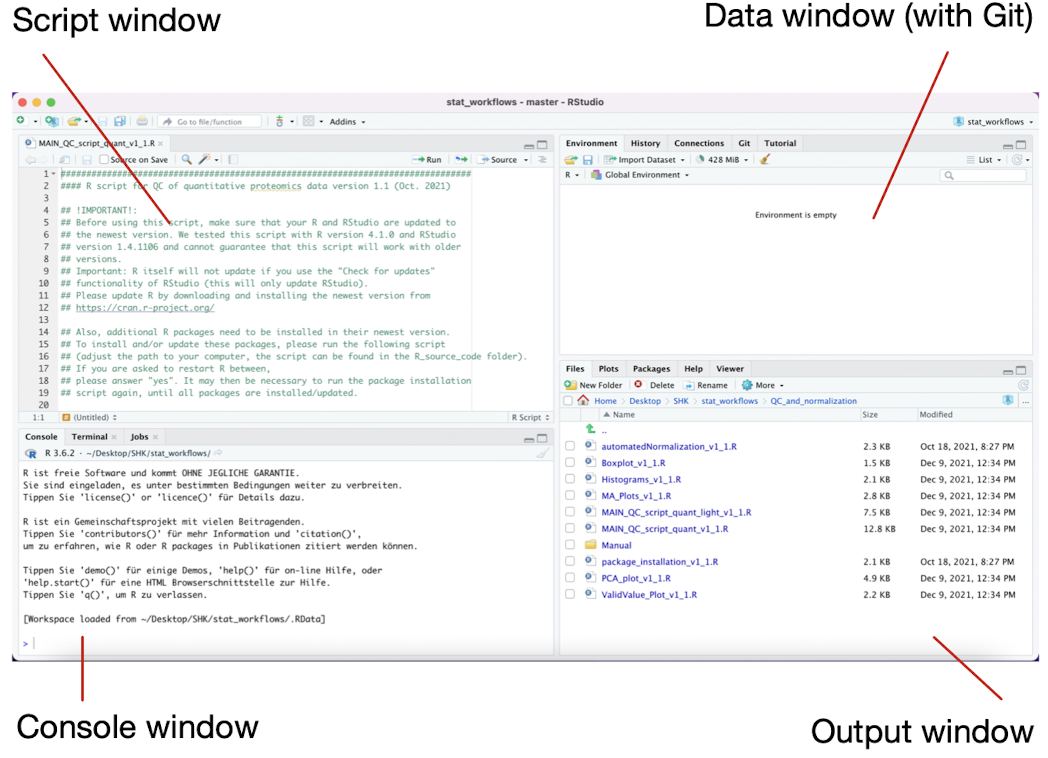
Important: Before running this script, make sure you have the newest version of R and RStudio.

The script was tested using R 4.2.1 and RStudio 2022.12.0. We cannot guarantee that it will work with older versions.

R itself will not update if you use the "Check for updates" functionality of RStudio (this will only update RStudio). To update R itself, please download and install the newest version form <https://cran.r-project.org> as explained above.

**4. Open main script & install necessary R packages:**

Open the main script (MAIN\_QC\_quant\_script\_v1\_3.R) by double clicking on the file. It will open in RStudio.



In line 21, change the file path so that it leads to the package installation script on your computer.

Mark this line and click "Run" in the top right corner of this window to execute it.

By calling install.package("packagename") in the R console, the R packages will be installed. Packages can also be installed and updated by selecting them from the Package menu on the output window.

If you are asked to restart R between, please answer „yes“. It's possible that you'll have to execute the package installation script again until all of the packages have been installed or updated.

If updating fails, try to uninstall the package and install it again.

**5. Prepare data set**

Data in the form of an xlsx file (rows = protein/peptides, columns = samples) can be loaded.

Columns that are not intensity columns (e.g. protein accession, gene name, etc.) can be specified as "id columns" later in the data file and will be skipped for processing and displays.

The remaining columns must be sample-related and contain intensities. All other columns that may have been in the original file must be deleted or designated as "id columns."

Before entering the data into R, delete any peptides or proteins that you don't want to be utilized in the diagnostic visuals (for example, contaminants).

Column names for the columns detailing the samples should be of the following format:

***groupname\_samplenumber*** (e.g.: control\_1, control\_2, control\_3, …, patient\_1, patient\_2, …)

There must be no more underscores (\_), blanks, or other special characters in the column names except dots.

If you have more than 10 or 100 samples in one group, you must add leading zeros to the sample numbers, else R will be unable to sort the samples correctly for the graphics.

(e.g.: *control\_01, control\_02, control\_03, …, control\_10, control\_11, …, control\_99* for >=10 samples)

(e.g.: *control\_001, control\_002, control\_003, …, control\_010, control\_011, …, control\_999* for >= 100 samples)

The analysis of data with more than two groups is possible without any problems.

The number of MA-Plots that are generated can be very large, which leads to a long run time and a big output file. If there will be more than 100 plots generated, you will be asked if you wish to continue or skip the MA-Plots.

**6. Change settings**

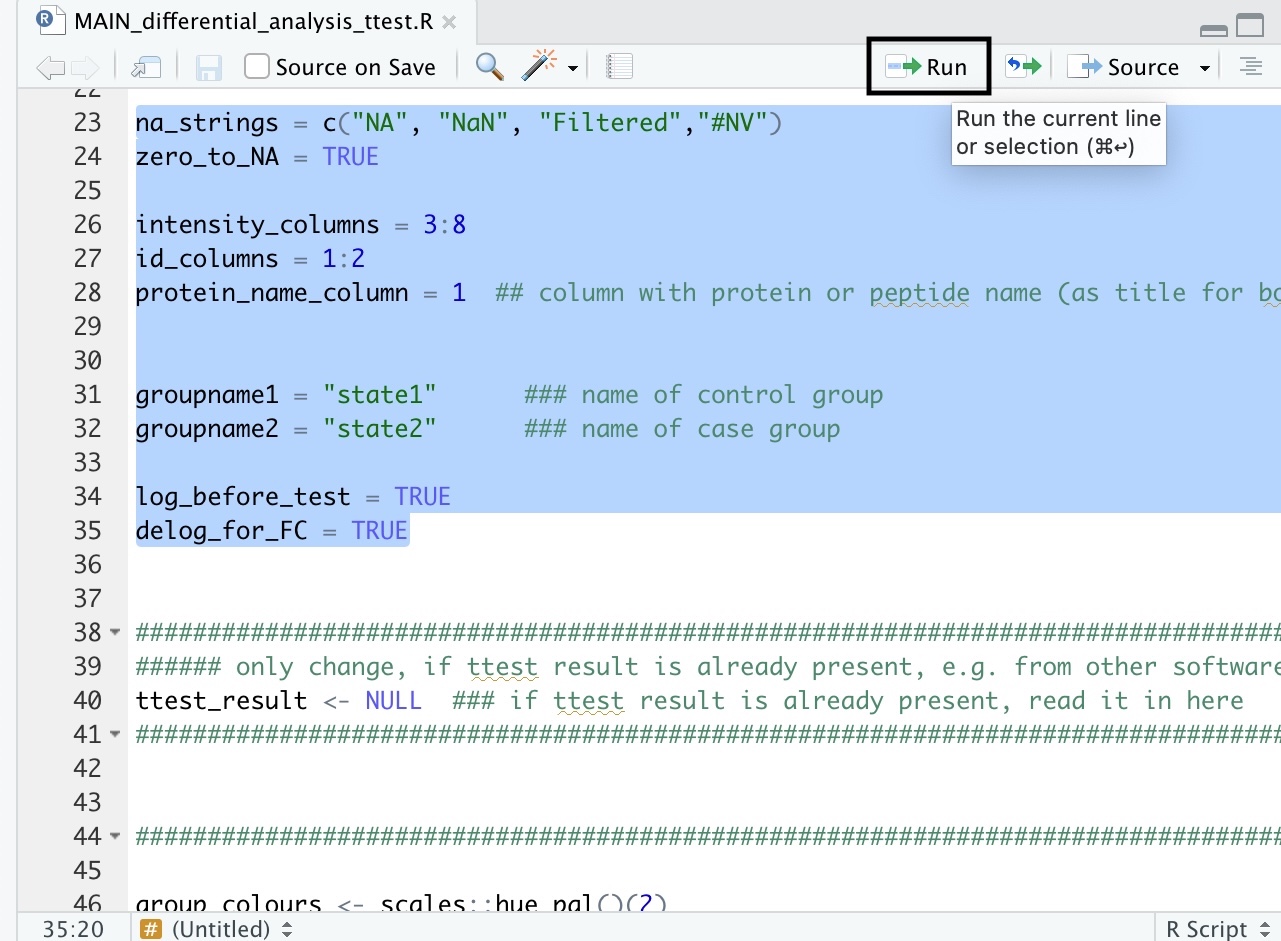
The following user settings can be modified and updated according to the data.

The basic settings had to be adjusted in almost every situation.

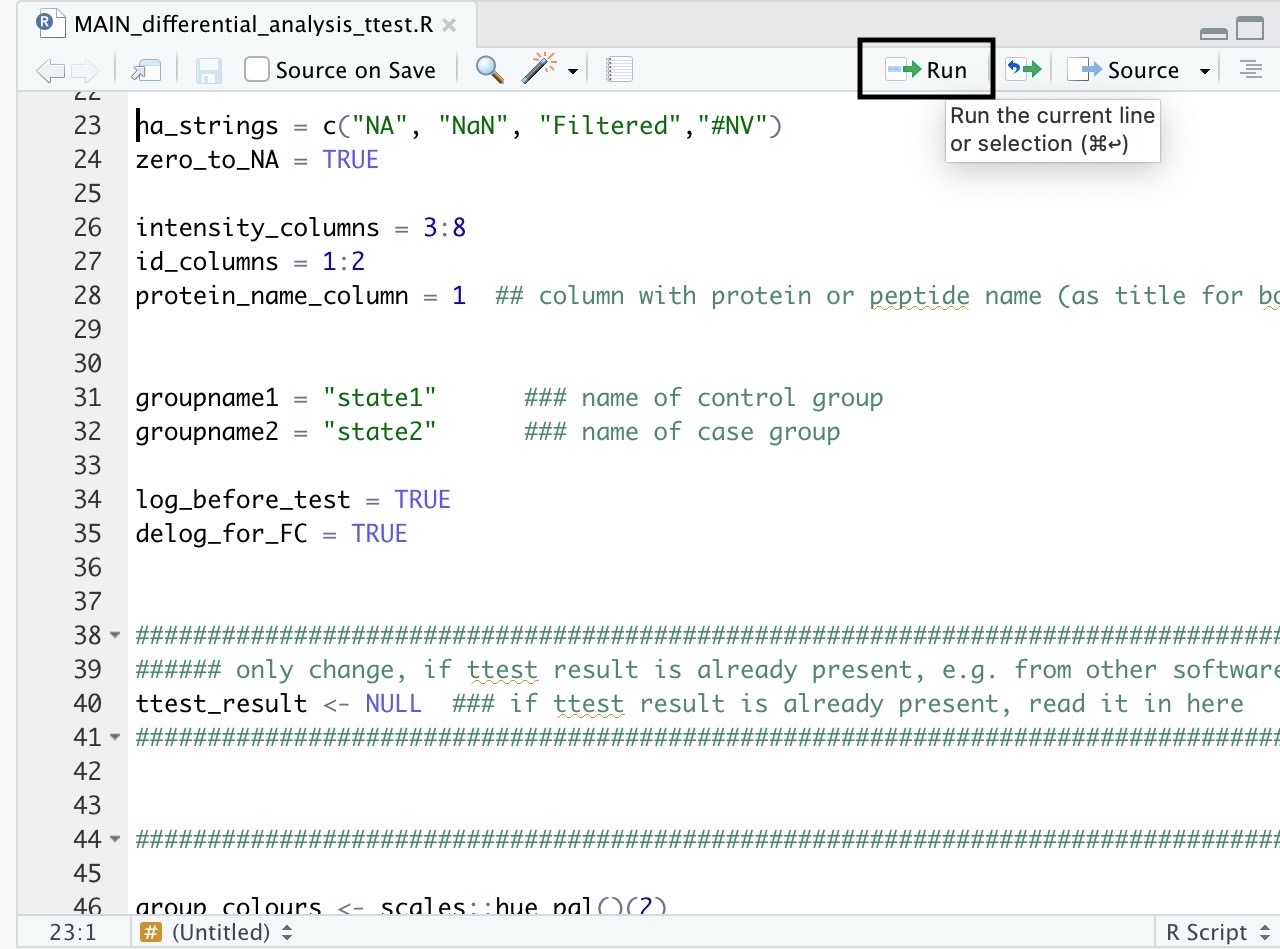
| **name** | **functionality** | **Data type** | **example** |
| --- | --- | --- | --- |
| path | Path of directory you want to work in (where the data is located) | String | „C:/Users/maxmustermann/UNI/important\_project/” |
| data\_path | Path of data/ excel table | String | "final\_data.xlsx" |
| output\_path | Path where the results will be saved (this folder is generated automatically if it does not exist already) | String | "QC\_results/" |
| RScript\_path | Path to RScripts for the QC workflow | String | "C:/Users/maxmustermann/UNI/QC\_worflow/" |
| intensity\_columns | Columns with numerical data | Vector | 3:10 |
| log\_data | If TRUE, data are logarithmized before normalization/plotting | Boolean | TRUE or FALSE |
| normalization | Type of normalization | String | „median" |
| use\_groups | Set it to FALSE if you have no groups in the data | Boolean | TRUE or FALSE |

**7. Run the workflow**

You can run each line of code by pressing the ‚Run‘ button (Figure 1) or highlighting all the code you want to run and then press the ‚Run‘ button (Figure 2).



*Figure 2*



*Figure 1*

The results are saved in the output folder specified by the initialise argument output\_path.

**8. Advanced Settings**

For the advanced settings, the default values will work in most cases, but can still be adjusted if you like, e.g. the colours used for the plots.

| **name** | **default** | **functionallity** | **Data type** | **example** |
| --- | --- | --- | --- | --- |
| group\_colours | NULL | Colours for the different groups for the plots | String | c(„grey“, „purple“) |
| groupvar\_name | „Group“ | Name of the group variable | String |  |
| plot\_device | pdf | Device | String | „png“ |
| plot\_ … | … | Widths, heights etc for the plots | number |  |
| sample\_filter | NULL | Filter specified samples by column name (if you do not want to use the whole data set) | String | c("control\_1", "control\_3", "patient\_2") |
| na\_strings | c(NA", "NaN", „Filtered","#NV") | Strings that stand for missing values in the data set | String |  |
| zero\_to\_NA | TRUE | If TRUE, intensity values of zero will be set to NA | Boolean |  |
| log\_base | 2 | Base fort he logarithm | Integer |  |
| suffix | normalization | Suffix for the plot file names (change if you do multiple analysis to not overwrite your results) | String |  |