# Rsome vignette

*Jorik van Rijn*

*6/28/2019*

## qPCR analysis with Rsome

### Introduction

Using the Rsome package you can automatically analyze qPCR data which was acquired using a BioRad qPCR machine. The first steps need to be performed in CFX manager or CFX maestro, to make sure the data is exported in the proper format. The following steps need to be taken in the CFX software to ensure that the Rsome package works:

1. Once the qPCR run is complete, make sure all wells have the correct target and sample name submitted in CFX maestro.
2. Export all data as .txt files. All these .txt files should be contained in a separate in the analysis directory: "parent_directory/input/".
3. Finally, you should have R installed, including the package 'devtools'. I suggest Rstudio as well!

### Install the package

The Rsome package can be downloaded and installed directly through Github. Although you can clone and install the package manually from https://github.com/jrijn/Rsome, the 'devtools' package makes this very easy from within R itself.

#### Install using devtools

Installation using the devtools package is as easy as this. Follow the instructions on updating and installing packages in the console window:

```r
install.packages('devtools')
devtools::install_github('jrijn/Rsome')
```

### Run the analysis pipeline in R

Then you're all set to run the analysis. The code chunck below runs the standard pipeline, based on the folder structure discussed in the introduction.

```r
#First define the directory where everything is going to happen! This folder should contain
#the subfolder '/input', which contains your qPCR data files in .txt format.
#Then supply the filename of the Cq data file. Don't include the .txt

#For a real world application:
setwd("parent_directory")
cqfile <- "_qPCR_file_name_"
meltfile <- "_meltcurve_derivative_file_name_"
```

```r
#FOR USE OF EXAMPLE DATA ONLY. Comment out if you are using your own data.
#filename <- Rsome::exampleData()

#Then run the pipeline.

cq <- Rsome::cqimport(cqfile)

mc <- Rsome::mcimport(
  cqimport = cq,
  meltderivative = meltfile)

re <- Rsome::relex(
  cq,
  household = "Gapdh",
  SDcutoff = 1,
  Cqcutoff = 35)

#And plot the graphs

p1 <- Rsome::cq.plot(cq)
p2 <- Rsome::mc.plot(mc)
p3 <- Rsome::re.plot(re)

#And save the graphs as needed

ggsave(plot=p1, "output/cqplot.pdf", width=20, height=10)
ggsave(plot=p2, "output/mcplot.pdf", width=20, height=10)
ggsave(plot=p3, "output/replot.pdf", width=20, height=10)

#Additional data formatting in between definition of re and p1 is allowed!
#This might be useful if there is an additional column defining the grouping variable.
```