R as Platform for the Analysis of dPCR and qPCR Experiments

by Stefan Rödiger, Michał Burdukiewicz, Konstantin Blagodatskikh and Peter Schierack

Abstract There is an ever-increasing number of publications, which use quantitative PCR (qPCR) or digital PCR (dPCR) to elicit fundamentals of biological processes. Novel amplification strategies based on quantitative isothermal amplification (qIA) start to become more prominent in life sciences and diagnostics. Several software solutions have been developed, which are either distributed as closed source software or as monolithic block with little freedom to perform highly customized analysis procedures. Others and we argue that R is an excellent environment for a reproducible and transparent analysis of data. However, for newcomers it is often very challenging to master R from reading the manuals and FAQs the not so obvious steps. Here we describe exemplary work-flows for the analysis of dPCR, qIA or qPCR experiments including the analysis of melting curve data. Our analysis relies entirely on R packages available from public repositories.

Introduction

The qPCR is the method of choice when a precise quantification of minute DNA traces of pathogens or the analysis of gene expression is required (Peirson et al., 2003). Numerous technologies have been developed in the past years (Rödiger et al., 2013b; Devonshire et al., 2013; Viturro et al., 2014; Rödiger et al., 2014; Khodakov and Ellis). Only few bioanalytical applications had such a significant impact on the progress of life sciences and medical sciences as the quantitative Polymerase Chain Reaction (qPCR). The scientific community work hard in the past two decades to uncover pitfalls of qPCR experiments. This lead finally to the development of peer-reviewed analysis algorithms (Ruijter et al., 2013), throughout analysed qPCR chemistries (Ruijter et al., 2014) and guidelines for a proper conduct of qPCR experiments as implemented in the MIQE guidelines (minimum information for publication of quantitative real-time PCR experiments) (Bustin et al.; Huggett et al., 2013). We share the philosophy of the MIQE guidelines to increase experimental transparency for better experimental practice and reliable interpretation of qPCR results. In the past decades emerged several isothermal amplification technologies, such as helicase dependent amplification (HDA). Isothermal amplification was readily combined with real-time monitoring technologies (qIA) and is used in various fields like diagnostics and point of care testing (Rödiger et al., 2014).

R is one of the most used tools in bioinformatics and is known as an early adopter of emerging technologies (Pabinger et al., 2014). Recently we published the **dpcR** package at CRAN, which is the first open source software package based on R for the analysis of digital PCR (dPCR) experiments. The dPCR Technology breaks fundamentally with the previous concept of nucleic acid quantification and can be seen as a next generation nucleic acid quantification. The key difference between dPCR and traditional PCR lies in the method of measuring (absolute) nucleic acids amounts, which yields discrete information instead of the continuous signal. This is possible after "clonal DNA amplification" in thousands of small separated partitions (e.g., droplets, nano chambers) (Huggett et al., 2013; Milbury et al., 2014; Morley, 2014). Partitions with no nucleic acid remain negative and the others turn positive. Selected technologies (e.g., OpenArray®Real-Time PCR System) monitor amplification reactions in the chambers in real-time. Cq values are calculated from the amplification curves and converted into discrete events by means of positive and negative partitions and the absolute quantification of nucleic acids is done by Poisson statistics (see **dpcR** for details).

Most of the commercial and experimental hardware platforms provide means to analyse the amplification curve data. Yet, in case of closed source software the analysis happens in most cases in a blackbox fashion tied to a specific platforms. Often such systems have limitation in the data processing and force the used to suboptimal analysis algorithm as discussed by Ruijter et al. (2013). The visualization options are usually limited by the software and not in acceptable publication quality. The data processing in spreadsheets is not advisable for research purposes. Often lacking tools to validate input, debug implemented procedures and automatize worflow spreadsheets are prone to errors and not well suited for more complicated analysis[1].

The complexity of hardware, wetware and software requires expertise to master a technical work-flow comprising standards for experimental design, generation and analysis of data, interpretation of results and reporting (Huggett et al., 2014). We argue that blackboxs are not necessarily a bad thing, but should be avoided wherever possible. Studies by McCullough and Heiser (2008); Almiron

[1]For more elaborated critique see http://www.burns-stat.com/documents/tutorials/spreadsheet-addiction/ and McCullough and Heiser (2008).

et al. (2010); Durán et al. (2014) exemplified this. Scientific misconduct and fraud have shaken the scientific community on several occasions (Fang et al., 2012). In particular qPCR is a sensible topic. Therefore, several reasons strongly support to use R in science. R provides essential packages to build a highly customized workflows, covering: data read-in, data preprocessing, analysis, post-processing, visualization and storage. As recently briefly reviewed in Pabinger et al. (2014), numerous R packages have been developed for the analysis of qPCR experiments, including: **kulife**, **MCMC.qpcr**, **qPCR.CT**, **DivMelt**, **qpcR**, **dpcR**, **chipPCR**, **MBmca**, **RDML**, **nondetects**, **qpcrNorm**, **HTqPCR**, **SLqPCR**, **ddCt**, **EasyqpcR**, **unifiedWMWqPCR**, **ReadqPCR**, **NormqPCR**. All the packages are either available from CRAN or Bioconductor (Gentleman et al., 2004). The packages can be freely combined in a plugin-like architecture. R is instrument independent, cross-platform and provides a wide spectrum of calculation options. In particular, visualization of experiments is one of R pinnacles. Though the intrinsic properties of R such as the naming convention (Bååth, 2012) and use of R's class systems (e.g., **S3**, **S4**, reference classes and **R6**) vary considerable depending on the package developer preferences there is the common ground to track numerical errors in R due to the open source approach. In addition, offers the R environment several data sets. R offers various methods for a standardized data import/export and exchange. Workflows can be embedded in structures for models (e.g., Predictive Model Markup Language (PMML) as proposed by Zeller et al. (2009), open data exchange formats (e.g., XML-based Real-Time PCR Data Markup Language (RDML) (Lefever et al., 2009), binary formats (Michna and Woods, 2013) or tools provided by the R workspace (R Development Core Team, 2012). Therefore, others and we argue that R is suitable for reproducible research (Gesmann and de Castillo, 2011; Murrell, 2012; Gandrud, 2013; Hofmann et al., 2013; Leeper, 2014; Liu and Pounds, 2014). In addition, several software R packages enable an efficient manipulation, restructuring and reshaping of data to make the readily-available for further processing. This is of particular importance on the human to machine interface (Oh, 2014).

The aim of this paper is to show two simple examples. In particular, we describe how to:

- read-in data from a standardized file format,
- pre-process the amplification curve data,
- calculate specific parameters from the amplification curve data,
- calculate the melting temperature,
- and report the data.

Our workflow effectively follows the principle illustrated in Figure 1. The intent is to aggregate functionalities dispersed between various packages and offer a fast insight for novices in the analysis of qPCR experiments with R.

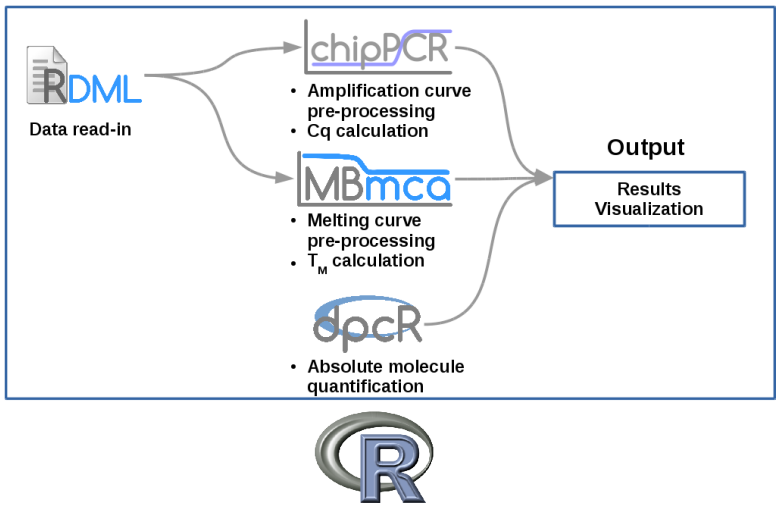

Figure 1: Exemplary workflow for qPCR experiments in R. Core functionality is provided by the R software environment for statistical computing and graphics. In our scenario we used the **RDML** package to read-in data in standardized format. Further processing of amplification curve data was performed with the **chipPCR** package and amplification curve data were analysed with the **MBmca** package. Cq, quantification cycle; T_M, melting temperature. The **dpcR** package can be embedded in the analysis of digital PCR experiments.

Setting-up a working environment

We recommend to perform the scripting in a dedicated integrated development environment (IDE) and graphical user interface (GUI) such as **RKWard** (Rödiger et al., 2012), **Rstudio**[2] or related technologies (Valero-Mora and Ledesma, 2012). Benefits of IDE's with GUI include syntax-highlighting, auto completion and function references for rapid prototyping of workflows.

Typically the qPCR analysis will start with data from a commercial platform. Most platforms have an option to export a CSV file or spreadsheets application file (e.g., *.xls, *.odt). The details for the import data has been described elsewhere (R Development Core Team, 2012; Rödiger et al., 2012). To keep the example sections compact we have chosen to load datasets from the **qcpR** package (Ritz and Spiess, 2008; Spiess et al., 2008) (v. 1.4.0) and **RDML** package to our workspace. In this study we used the **RDML** package (v. 0.4-2) for data read-in. The data were measured with a CFX96 System (Bio-Rad) and then exported as RDML v1.1 format file. The **chipPCR** package (v. 0.0.8-3) was used for data preprocessing, quality control and the calculation of the quantification cycle (Cq). The Cq is a quantitative measure, which represents the number of cycles needed to reach a user defined threshold fluorescence signal level. Typically Cq are determined identically the exponential phase of a qPCR reaction. Several Cq methods have been described (Ruijter et al., 2013). In this study we have chosen the second derivative maximum method (Cq_{SDM}). Due to the ubiquitous use we used in a example also the "Cycle threshold" (Cq_{Ct}) method.

In a perfect qPCR reaction, the amount of amplicon doubles (2^n; n = cycle number) at each cycle. Here the amplification efficiency (AE) is 100 %. However, in reality, numerous factors cause an inhibition of the amplification (AE < 100 %). The AE can determined be the relation of the Cq value depending on the sample input quantity as detailed described in (Rödiger and Burdukiewicz, 2014).

In Rödiger et al. (2013a) we described the application of R for the analysis of melting curve experiments on the surface of microbeads. Since the mathematical foundation for melting curve analysis (MCA) is identical between all platforms we applied the functions from the **MBmca** package (Rödiger et al., 2013a) for an analysis of the target specific melting temperature (T_M) in our qPCR experiment. We used the **MBmca** package (0.0.3-4) for analysis of melting curve data.

We complete our study with a simple example for the analysis of dPCR experiment. In particular, we used the **dpcR** (0.1.3.1) to estimate the number of molecules in a sample.

Results

In this section we will try to show that R is a unified open software which fits the needs for (I) data analysis and presentation in research, (II) as software frame-work for novel technical developments, (III) as platform for teaching this new technology and (IV) serves as reference for statistical methods.

Example one - qPCR and Amplification Efficiency Calculation

The goal of our fist example was to calculate the Cq values and the AE from a qPCR experiment. Therefore, we used the "guescini1" dataset[3] from the **qcpR** package.

```
# Collect information about the R session used for the analysis of the qPCR
# experiment.
current.session <- sessionInfo()

# Next we load the 'guescini1' dataset from the qpcR package the to
# workspace and assign it to the object tmp.
require(qpcR)
tmp <- guescini1

# Define the threshold value for the th.cyc function
Ct <- 0.05

# Define the diltuion of the sample DNA quantity for
# the calibration curve.

dil <- sapply((2:-4), function(i) {10^i})
```

[2] http://www.rstudio.com/

[3] Details of the experiment are described in Guescini et al. (2008).

```
# Preporcess the amplification curve data with the CPP function from the chipPCR
# package.
res.CPP <- cbind(tmp[, 1], apply(tmp[, -1], 2, function(x) {
    CPP(tmp[, 1], x, trans = TRUE, method.norm = "minm", bg.range = c(1,7))\$y.norm
}))

Cq.Ct <- apply(tmp[, -1], 2, function(x) {th.cyc(res.CPP[, 1], x, r = Ct)[1]})
Cq.SDM <- apply(tmp[, -1], 2, function(x) {summary(inder(res.CPP[, 1], x))[2]})

pdf("dilution_Cq.pdf", width = 9.5, height = 12)
layout(matrix(c(1,2,3,3,4,5), 3, 2, byrow = TRUE))

matplot(tmp[, -1], type = "l", lty = 1, col = 1, xlab = "Cycle",
            ylab = "RFU", main = "Raw data")
legend("topleft", "A", cex = 3, bty = "n")

matplot(res.CPP[, -1], type = "l", lty = 1, col = 1, xlab = "Cycle",
        ylab = "RFU", main = "Pre-processed data")
legend("topleft", "B", cex = 3, bty = "n")
abline(h = Ct, col = "red", lwd = 2)

plot(Cq.SDM, Cq.Ct, xlab = "Ct method", ylab = "SDM method",
     main = "Comparison of Cq methods")
abline(res.Cq)
legend("topleft", "C", cex = 3, bty = "n")

plot(effcalc(dil, t(matrix(Cq.Ct, nrow = 12, ncol = 7))), CI = TRUE)
legend("topright", "D", cex = 3, bty = "n")

plot(effcalc(dil, t(matrix(Cq.SDM, nrow = 12, ncol = 7))), CI = TRUE)
legend("topright", "E", cex = 3, bty = "n")
```

Example two - qPCR and Melting Curve Analysis

In this study we used the **RDML** package to read the qPCR experiment. A good practice for reproducible research is to track the package versions and environment used during the analysis. The function *sessionInfo*() from the **utils** package provides this information. Assuming that the analysis starts with a clean R session it is possible to assign the required packages to an object only, as shown in our example below[4].

```
# Load the required packages for the data import and analysis.

# Import the qPCR and melting curve data via the RDML package
require(RDML)

# Load the chipPCR package for the pre-processing and curve data quality
# analysis.
require(chipPCR)

# Load the MBmca package for the melting curve analysis.
require(MBmca)

# Collect information about the R session used for the analysis of the qPCR
# experiment.
current.session <- sessionInfo()


# Load the BioRad_qPCR_melt.rdml file form RDML package and assign the data to the
# object BioRad.

path <- path.package("RDML")
```

[4]The reproducibility of research can be further improved by using dedicated tools. For example, **archivist** package allows not only stores and recovers crucial data, but also preserves metadata of saved objects.

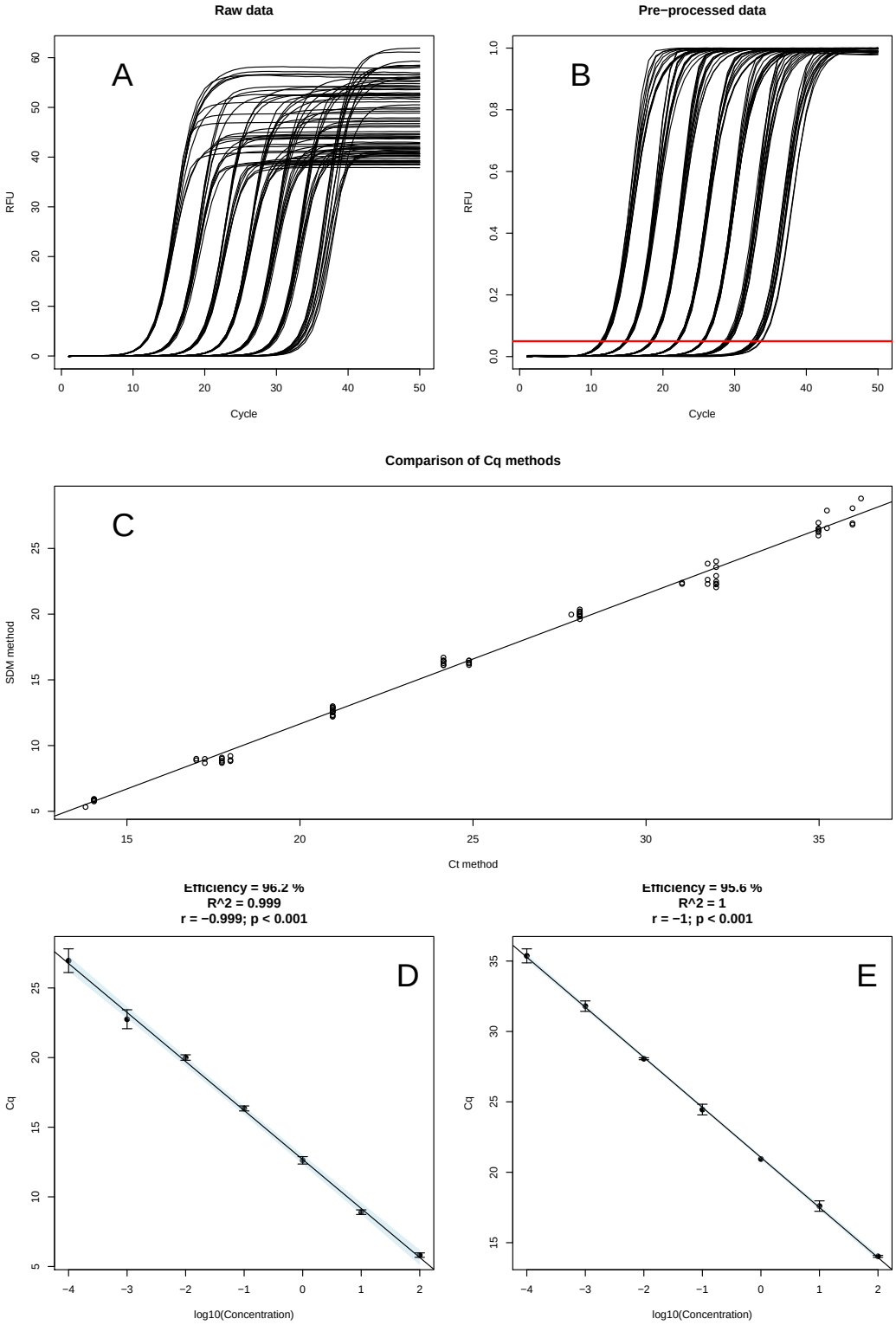

Figure 2: Analysis of the amplification curve data of the calibration curve samples. **(A)** Visual inspection of the raw data from the `guescini1` dataset. The qPCR curves display a broad variation in plateau fluorescence (38 – 62 RFU). The red horizontal line indicates the fluorescence level (0.05) used for the calculation of the Cq by the "cycle threshold" method. **(B)** the `CPP` function from the **chipPCR** was sued to baseline the data, to smooth the data with Savitzky-Golay smoothing filter and to normalize the data between 0 and 1. **(C)** The Cq values were calculated for the second derivative maximum (SDM) method (`inder`, **chipPCR**) and the cycle threshold method (Ct) (`th.cyc`, **chipPCR**). The threshold value was set to $r = 0.05$. The Cq values from the SDM and Ct method were plotted and analysed by a linear regression $R^2 = 0.9945$ ($P < 2.2^{-16}$) and Pearson's $r = 0.9972605$ ($P < 2.2^{-16}$). The Cq values from **(D)** the Ct method and **(E)** the SDM method were automatically analysed with the `effcalc` (**chipPCR**) function.

```
filename <- paste(path, "/extdata/", "BioRad_qPCR_melt.rdml", sep = "")
BioRad <- RDML(filename, name.pattern = "%TUBE%_%NAME%_%TYPE%_%TARGET%")

# Fetch cycle dependent fluorescence for the Cy5 chanel of the gen
# katG315 and aggregate the data in the object qPCR.
qPCR <- cbind(BioRad[["qPCR"]][["Cy5-2"]][["pos"]],
              BioRad[["qPCR"]][["Cy5-2"]][["unkn"]][, -1],
              BioRad[["qPCR"]][["Cy5-2"]][["ntc"]][, -1])

# Use plotCurves function from the chipPCR package to get an overview of the
# amplification curve samples.

plotCurves(qPCR[, 1], qPCR[, -1], type = "l")
```

Next we inspected and pre-processed a subset of the amplification curve data solely using functionalities provided by the **chipPCR** package. The `plotCurves` function was used to get an overview of the curvatures. The data indicated a baseline shift in all curves with a slight negative trend (Figure 3). Therefore, we used the

```
# Fetch temperature dependent fluorescence for the Cy5 chanel of the gen
# katG315 and aggregate the data in the object melt.
melt <- cbind(BioRad[["Melt"]][["Cy5-2"]][["pos"]],
              BioRad[["Melt"]][["Cy5-2"]][["unkn"]][, -1],
              BioRad[["Melt"]][["Cy5-2"]][["ntc"]][, -1])

# Calculate the melting temperature with the diffQ function
# from the MBmca package. Use as simple logic to test if the
# a sample with the expexcted Tm of circa 54.5 degree Celsius
# is found.
res.Tm <- apply(melt[, -1], 2, function(x) {
                res.Tm <- diffQ(cbind(melt[, 1], x), fct = max, inder = TRUE)
                Decission <- ifelse(res.Tm[1] > 54 & res.Tm[1] < 55, 1, 0)
                out <- data.frame(res.Tm[c(1,2)], Decission)
                }
              )
# Present the results in a tabular output.
resutlts <- matrix(unlist(res.Tm), nrow = length(res.Tm), byrow = TRUE,
       dimnames = list(colnames(melt[, -1]),
       c("Tm", "Height", "Decission")))

> resutlts

                             Tm       Height Decission
D1_Alm12_pos_Cy5-2     52.09946   0.66704327         0
D2_Alm12_pos_Cy5-2     52.66936   0.18990160         0
D3_Alm13_pos_Cy5-2     33.53773  56.08059031         0
D4_Alm13_pos_Cy5-2     33.84572  62.76151803         0
D5_Alm14_pos_Cy5-2     38.91839  93.81983093         0
D6_Alm14_pos_Cy5-2     39.00768  83.94777001         0
D7_katG 315_unkn_Cy5-2 54.73343 106.26186606         1
D8_katG 315_unkn_Cy5-2 54.76646  99.49306091         1
D9_H2O_ntc_Cy5-2       44.85700  -0.08934079         0
D10_H2O_ntc_Cy5-2      48.18610   0.31836438         0
```

Example three - Isothermal Amplification

An alternative to PCR offer isothermal amplification (IA) methods (Rödiger et al., 2014). IA methods use a constant temperature rather than cycling through denaturation, annealing and extension steps. The corresponding signal is monitored depending on the time instead of cycles. We performed a quantitative isothermal amplification by Helicase Dependent Amplification (HDA). The enzyme DNA Helicase unwinds DNA. Therefore, no thermal denaturation is needed. We performed the HDA in the VideoScan platform (Rödiger et al., 2013b) at 65 °Cfor the target *pCNG1*[5]. The resulting dataset `C81` is

[5]The HDA conditions were taken from the "IsoAmp III Universal tHDA Kit", Biohelix Corp, as described by the vendor. In detail, the reaction was composed of "mix A)" 10 μL A. bidest, 1.25 μL 10xbuffer, 0.75 μL primer(150

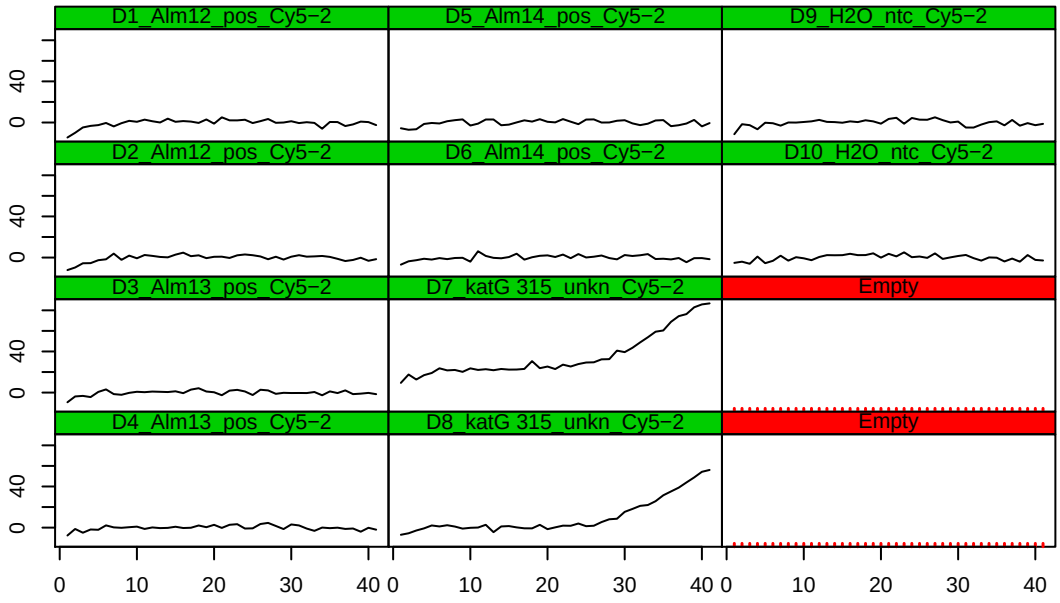

Figure 3: Analysis of the amplification curve data of the calibration curve samples by the `plotCurves` function from the **chipPCR** package.

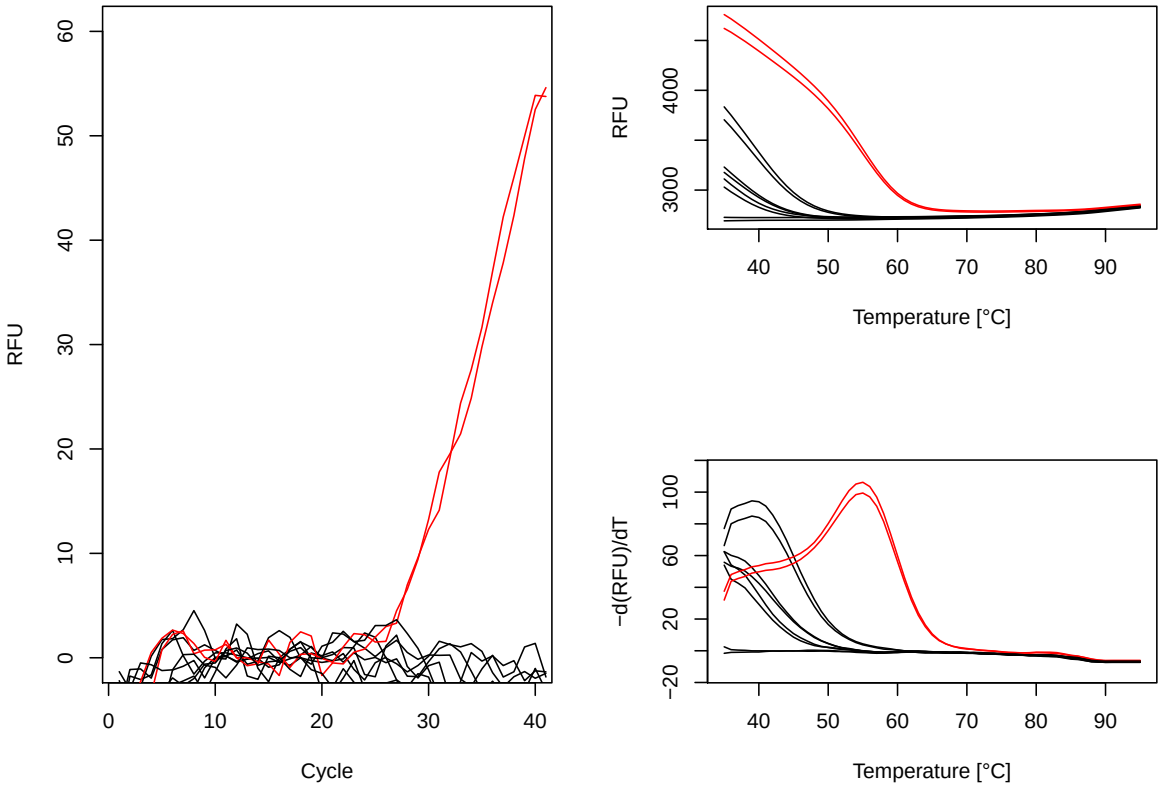

Figure 4: by the `diffQ` function from the **MBmca** package.

part of the chipPCR package. Two concentrations of input DNA were used in the HDA.

```
# Define the threshold level for the calculation of the cycle threshold time.
Ct <- 0.05
# Drawn in an 2-by-1 array on the device by two columns and one row.
par(mfrow = c(2, 1))

# Plot the raw data from the C81 dataset to the first array and add
# a legend.
plot(NA, NA, xlim = c(0, 120), ylim = c(0.4, 1.2), xlab = "Time (min)", ylab = "RFU")
mtext("A", cex = 2, side = 3, adj = 0, font = 2)
lapply(c(2, 4), function(i) {
    lines(C81[, i]/60, C81[, i + 1], type = "b", pch = 20, col = i - 1)
})
legend(10, 0.8, c("D1: 1x", "D2: 1:10 diluted sample"), pch = 19, col = c(1, 3),
    bty = "n")

# Prepare a plot on the second array for the pre-proccessed data.
plot(NA, NA, xlim = c(0, 120), ylim = c(0, 0.8), xlab = "Time (min)", ylab = "RFU")
mtext("B", cex = 2, side = 3, adj = 0, font = 2)

# Apply the CPP functions to pro-process the raw data.
res <- lapply(c(2, 4), function(i) {
    y.s <- CPP(C81[, i]/60, C81[, i + 1],
                    trans = TRUE,                # Basline to zero
                method = "spline",         # Smooth data with spline
                bg.outliers = TRUE,         # Remove outliers
                bg.range = c(1, 190))         # Define range of background singnal
    lines(C81[, i]/60, y.s$y.norm, type = "b", pch = 20, col = i - 1)
# Use the th.cyc function to calculate the cycle threshold time.
    paste(round(th.cyc(C81[, i]/60, y.s$y.norm, r = Ct)[1], 2), "min")
})

# Add the cycle threshold time and the threshold level to plot.

abline(h = Ct, lty = 2)
text(10, 0.55, "Cq:")
legend(10, 0.5, paste(c("D1: ", "D2: "), res), pch = 19, col = c(1, 3), bty = "n")

 layout(matrix(c(1,2,1,3), 2, 2, byrow = TRUE))
plot(NA, NA, xlim = c(1, 40), ylim = c(0,60), xlab = "Cycle", ylab = "RFU")
lapply(2L:ncol(melt), function(i) {lines(qPCR[, 1], CPP(qPCR[, 1], qPCR[, i], trans = TRUE, bg.range = c(10,2


matplot(melt[, 1], melt[, -1], type = "l", col = c(rep(1,12), rep(2,12)), lty = 1, xlab = "Temperature [°C]",
            ylab = "RFU")
plot(NA, NA, xlim = c(35, 95), ylim = c(-15,115), xlab = "Temperature [°C]", ylab = "-d(RFU)/dT")
color <- c(rep(1,3), rep(2,12))
lapply(2L:ncol(melt), function(i) {
                lines(diffQ(cbind(melt[, 1], melt[, i]), verbose = TRUE,
                      fct = max, inder = TRUE)$xy)
                      })
```

Example four - digital PCR

We have developed the dpcR package for analysis and presentation of digital PCR experiments. The dpcR package can be used to build custom-made analysers and provides structures to be openly extended by the scientific community. Simulations and predictions of binomial and Poisson distributions, commonly used theoretical models of dPCR, statistical data analysis methods, plotting facilities

nM final), 0.5 μL template plasmid. Preincubation: The mixture was incubated for 2 min at 95°Cand immediately placed on ice. Reaction "mix B)" contained 5 μL A. bidest., 1.25 μL 10x buffer, 2 μL NaCl, 1.25 μL $MgSO_4$, 1, 75μL dNTPs, 0.25 μL EvaGreen (Biotium), 1 μL enzyme mix. The mix was covered with 50 μL mineral oil (Roth). The fluorescence measurement in VideoScan HCU started directly after adding "mix B)" at 65°C. A $1x$ (D1) and a $1:10$ dilution (D2) were tested.

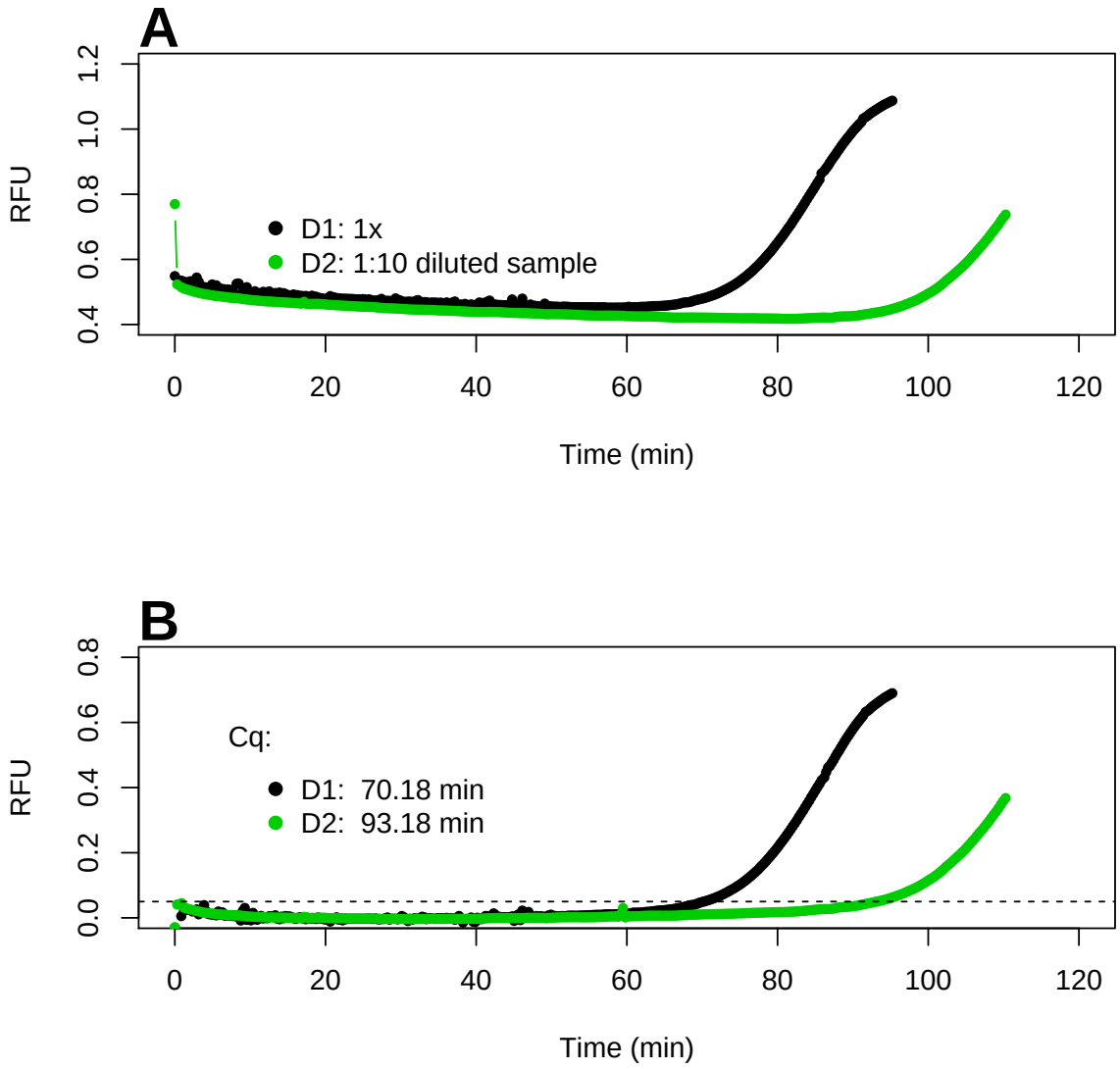

Figure 5: Quantitative isothermal amplification by Helicase Dependent Amplification (HDA). **(A)** The raw data of the HDA (D1, undiluted, D2 1 : 10 diluted) exhibit some outliers (detector artifacts), an off-set of circa 0.5 relative fluorescence units (RFU) and a slight negative trend in the baseline region (0 to 52 minutes). **(B)** First we used the `CPP` function from the **chipPCR** package to smooth the data with spline function. Baselining was done with a robust MM-estimator (range 0 to 52 min). Finally, we used the `th.cyc` function from the **chipPCR** package to calculate the cycle threshold time for samples D1 and D2. The threshold value was set to $r = 0.05$ (−−).

and report generation tools are part of the package (Pabinger et al., 2014). Here, we show briefly an example for the **dpcR**[6].

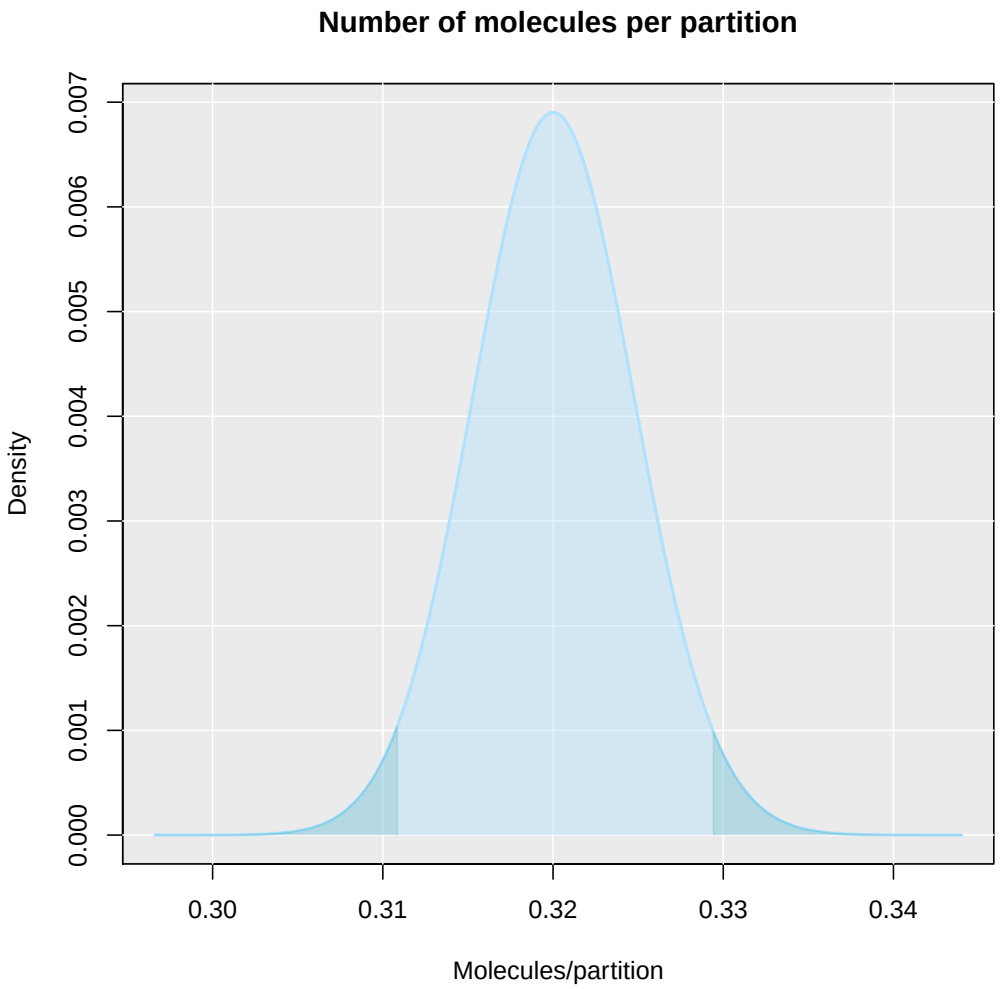

Figure 6: `dpcr_density` function from the **dpcR** package.

Discussion and Conclusion

This study gave a brief introduction how to perform a qPCR, qIA or dPCR analysis with R based on packages available from CRAN. In addition we briefly referenced to a vast collection of additional packages available from CRAN and Bioconductor. The packages may be considered building blocks (libraries) to create what users want and need. We showed that an automatic research with R offers powerful means for statistical analysis and visualization. The software is not tied to a vendor or specific application (e.g, chamber or droplet based digital PCR, capillary or plate qPCR). It should be quite easy even for an inexperienced user to define a workflow and to setup environment for specific needs in a broad range of technical settings (Figure 7). R enforces no monolithic integration. We claim that the modular structure of R packages allows user to perform flexible data analysis adjusted to their needs and to design frameworks for high-throughput analysis. R allows to access and reuse code for

[6]Selected functionality was implemented as interactive**shiny** GUI application to make the software accessible for users who are not fluent in R and but also for experts who which to automatize routine tasks. Details and examples of the **shiny** web application framework for R can be found at `http://shiny.rstudio.com/`. We implemented flexible yet simple user interfaces, which run the analyses and graphical representation into interactive web applications either as service on a web severer or on a local machine without knowledge of HTML or ECMAScript (see **dpcR** manual). The interface is designed in a cascade workflow approach (Data import → Analysis → Output → Export) with interactive users choice on input data, methods and parameters using typical GUI elements such as sliders, drop-downs and text fields. An example can be found at `https://michbur.shinyapps.io/dpcr_density/`. This approach enables the automatized outputs of R objects in combined plots, tables and summaries.

the creation of reports in various formats (e.g., HTML, PDF). Most of the software is cross-platform open source software and is freely available from CRAN or Bioconductor. Despite the fact that R is free of charge it is quite possible to build commercial applications. The packages cover implementation of novel approaches and peer-reviewed analysis methods. R packages are an open environment to adopt to the growing knowledge in dPCR and qPCR. Therefore, we argue that R may provide a structure for standardized nomenclature and serve as reference in qPCR and dPCR analysis. Speaking about openness, it is important to emphasize that main advantage is the software is transparent at any time for anybody. Thus, it is possible to track numerical errors.

`http://michbur.github.io/pcRuniveRsum/`

A serve disadvantage of R is the lack of comprehensive GUIs for qPCR analysis. Other and we believe that a graphical user interface (GUI) is a key technology to spread the use of R in bioanalytical sciences. The command-line structure makes R "inaccessible" for many novices. This we support the attempt that automatic routines are made accessible via GUIs (Rödiger et al., 2012). However, work in this has has recently started and is still under development.

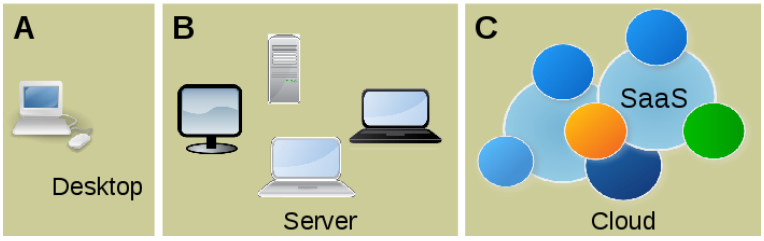

Figure 7: Deployment of R applications for the qPCR and dPCR experiments. **(A)** R is typically run from a desktop computer an operated by an GUI/IDE application such a **Rstudio** or **RKWard**. This approach is provides a flexible workflow for individuals. **(B)** Another approach is to run R with specific applications on a local server. Such scenarios are useful for the deployment within research departments or cooperate units. **(C)** Cloud computing (CC) provides shared and scalable computing capacity (e.g., computing capacity, application software) and storage capacity (e.g., databases) as a service to an individual user or a community Service categories include: Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS) and Software-as-a-Service (SaaS) over a network. Providers of CC manage the infrastructure and resources to achieve coherence and economies of scale similar to a utility over a network (typically the Internet).

Acknowledgment

Part of this work was funded by the BMBF InnoProfile-Transfer-Projekt 03 IPT 611X. We would like to thank R community. Part of this work was funded by the Russian Ministry of Education and Science (project No. RFMEFI62114X0003) and with usage of scientific equipment of Center for collective use "Biotechnology" at All-Russia Research Institute of Agricultural Biotechnology.

Bibliography

M. G. Almiron, B. Lopes, A. L. C. Oliveira, A. C. Medeiros, and A. C. Frery. On the numerical accuracy of spreadsheets. *Journal of Statistical Software*, 34(4):1–29, 4 2010. ISSN 1548-7660. URL `http://www.jstatsoft.org/v34/i04`. [p1]

R. Bååth. The State of Naming Conventions in R. *The R Journal*, 4(2):74–75, Dec. 2012. URL `http://journal.r-project.org/archive/2012-2/RJournal_2012-2_Baaaath.pdf`. [p2]

S. A. Bustin, V. Benes, J. A. Garson, J. Hellemans, J. Huggett, M. Kubista, R. Mueller, T. Nolan, M. W. Pfaffl, G. L. Shipley, J. Vandesompele, and C. T. Wittwer. The MIQE Guidelines: Minimum Information for Publication of Quantitative Real-Time PCR Experiments. 55(4):611–622. ISSN 0009-9147, 1530-8561. doi: 10.1373/clinchem.2008.112797. URL `http://www.clinchem.org/content/55/4/611`. [p1]

A. S. Devonshire, R. Sanders, T. M. Wilkes, M. S. Taylor, C. A. Foy, and J. F. Huggett. Application of next generation qPCR and sequencing platforms to mRNA biomarker analysis. *Methods (San Diego, Calif.)*, 59(1):89–100, Jan. 2013. ISSN 1095-9130. doi: 10.1016/j.ymeth.2012.07.021. [p1]

A. J. Durán, M. Pérez, and J. L. Varona. The misfortunes of a trio of mathematicians using computer algebra systems. Can we trust in them? *Notices of the American Mathematical Society*, 2014. ISSN 0002-9920. doi: dx.doi.org/10.1090/noti1173. URL http://www.ams.org/notices/201410/rnoti-p1249.pdf. [p2]

F. C. Fang, R. G. Steen, and A. Casadevall. Misconduct accounts for the majority of retracted scientific publications. *Proceedings of the National Academy of Sciences of the United States of America*, 109(42): 17028–17033, Oct. 2012. ISSN 1091-6490. doi: 10.1073/pnas.1212247109. [p2]

C. Gandrud. *Reproducible Research with R and RStudio*. Chapman and Hall/CRC, July 2013. ISBN 9781466572843. [p2]

R. C. Gentleman, V. J. Carey, D. M. Bates, B. Bolstad, M. Dettling, S. Dudoit, B. Ellis, L. Gautier, Y. Ge, J. Gentry, K. Hornik, T. Hothorn, W. Huber, S. Iacus, R. Irizarry, F. Leisch, C. Li, M. Maechler, A. J. Rossini, G. Sawitzki, C. Smith, G. Smyth, L. Tierney, J. Y. H. Yang, and J. Zhang. Bioconductor: open software development for computational biology and bioinformatics. *Genome biology*, 5(10):R80, 2004. ISSN 1465-6914. doi: 10.1186/gb-2004-5-10-r80. PMID: 15461798 PMCID: PMC545600. [p2]

M. Gesmann and D. de Castillo. Using the Google Visualisation API with R. *The R Journal*, 3(2):40–44, Dec. 2011. URL http://journal.r-project.org/archive/2011-2/RJournal_2011-2_Gesmann+de~Castillo.pdf. [p2]

M. Guescini, D. Sisti, M. B. Rocchi, L. Stocchi, and V. Stocchi. A new real-time PCR method to overcome significant quantitative inaccuracy due to slight amplification inhibition. *BMC Bioinformatics*, 9(1): 326, July 2008. ISSN 1471-2105. doi: 10.1186/1471-2105-9-326. URL http://www.biomedcentral.com/1471-2105/9/326/abstract. PMID: 18667053. [p3]

H. Hofmann, A. Unwin, and D. Cook. Let Graphics Tell the Story - Datasets in R. *The R Journal*, 5(1):117–130, June 2013. URL http://journal.r-project.org/archive/2013-1/RJournal_2013-1_hofmann-unwin-cook.pdf. [p2]

J. Huggett, J. O'Grady, and S. Bustin. How to make mathematics biology's next and better microscope. *Biomolecular Detection and Quantification*, 2014. ISSN 2214-7535. doi: 10.1016/j.bdq.2014.09.001. URL http://www.sciencedirect.com/science/article/pii/S2214753514000060. [p1]

J. F. Huggett, C. A. Foy, V. Benes, K. Emslie, J. A. Garson, R. Haynes, J. Hellemans, M. Kubista, R. D. Mueller, T. Nolan, M. W. Pfaffl, G. L. Shipley, J. Vandesompele, C. T. Wittwer, and S. A. Bustin. The digital MIQE guidelines: Minimum Information for Publication of Quantitative Digital PCR Experiments. *Clinical Chemistry*, 59(6):892–902, June 2013. ISSN 1530-8561. doi: 10.1373/clinchem.2013.206375. PMID: 23570709. [p1]

D. A. Khodakov and A. V. Ellis. Recent developments in nucleic acid identification using solid-phase enzymatic assays. *Microchimica Acta*, pages 1–14. ISSN 0026-3672, 1436-5073. doi: 10.1007/s00604-014-1167-z. URL http://link.springer.com/article/10.1007/s00604-014-1167-z. [p1]

T. J. Leeper. Archiving Reproducible Research and Dataverse with R. *The R Journal*, 6(1):NN–NN, 2014. URL http://journal.r-project.org/archive/accepted/leeper.pdf. [p2]

S. Lefever, J. Hellemans, F. Pattyn, D. R. Przybylski, C. Taylor, R. Geurts, A. Untergasser, J. Vandesompele, and RDML consortium. RDML: structured language and reporting guidelines for real-time quantitative PCR data. *Nucleic Acids Research*, 37(7):2065–2069, Apr. 2009. ISSN 1362-4962. doi: 10.1093/nar/gkp056. PMID: 19223324 PMCID: PMC2673419. [p2]

Z. Liu and S. Pounds. An R package that automatically collects and archives details for reproducible computing. *BMC Bioinformatics*, 15(1):138, May 2014. ISSN 1471-2105. doi: 10.1186/1471-2105-15-138. URL http://www.biomedcentral.com/1471-2105/15/138/abstract. [p2]

B. D. McCullough and D. A. Heiser. On the accuracy of statistical procedures in microsoft excel 2007. *Computational Statistics & Data Analysis*, 52(10):4570–4578, June 2008. ISSN 0167-9473. doi: 10.1016/j.csda.2008.03.004. URL http://www.sciencedirect.com/science/article/pii/S0167947308001606. [p1]

P. Michna and M. Woods. RNetCDF – A Package for Reading and Writing NetCDF Datasets. *The R Journal*, 5(2):29–37, Dec. 2013. URL http://journal.r-project.org/archive/2013-2/michna-woods.pdf. [p2]

C. A. Milbury, Q. Zhong, J. Lin, M. Williams, J. Olson, D. R. Link, and B. Hutchison. Determining lower limits of detection of digital PCR assays for cancer-related gene mutations. *Biomolecular Detection and Quantification*, 1(1):8–22, Sept. 2014. ISSN 2214-7535. doi: 10.1016/j.bdq.2014.08.001. URL http://www.sciencedirect.com/science/article/pii/S2214753514000047. [p1]

A. A. Morley. Digital PCR: A brief history. *Biomolecular Detection and Quantification*, 1(1):1–2, Sept. 2014. ISSN 2214-7535. doi: 10.1016/j.bdq.2014.06.001. URL http://www.sciencedirect.com/science/article/pii/S2214753514000023. [p1]

P. Murrell. It's Not What You Draw, It's What You Don't Draw. *The R Journal*, 4(2):13–18, Dec. 2012. URL http://journal.r-project.org/archive/2012-2/RJournal_2012-2_Murrell2.pdf. [p2]

J. Oh. Automatic Conversion of Tables to LongForm Dataframes. *The R Journal*, N(2):NN–NN, 2014. URL http://journal.r-project.org/archive/accepted/oh.pdf. [p2]

S. Pabinger, S. Rödiger, A. Kriegner, K. Vierlinger, and A. Weinhäusel. A survey of tools for the analysis of quantitative PCR (qPCR) data. *Biomolecular Detection and Quantification*, 1, 2014. ISSN 2214-7535. [p1, 2, 8]

S. N. Peirson, J. N. Butler, and R. G. Foster. Experimental validation of novel and conventional approaches to quantitative real-time PCR data analysis. *Nucleic Acids Research*, 31(14):e73, July 2003. ISSN 1362-4962. PMID: 12853650 PMCID: PMC167648. [p1]

R Development Core Team. *R Data Import/Export*. R Foundation for Statistical Computing, Vienna, Austria, 2012. URL http://www.R-project.org/. ISBN 3-900051-10-0. [p2, 3]

C. Ritz and A.-N. Spiess. qpcR: an R package for sigmoidal model selection in quantitative real-time polymerase chain reaction analysis. *Bioinformatics*, 24(13):1549–1551, July 2008. ISSN 1367-4803, 1460-2059. doi: 10.1093/bioinformatics/btn227. URL http://bioinformatics.oxfordjournals.org/content/24/13/1549. PMID: 18482995. [p3]

S. Rödiger and M. Burdukiewicz. chipPCR: Toolkit of helper functions to pre-process amplification data, June 2014. URL http://cran.r-project.org/web/packages/chipPCR/index.html. [p3]

S. Rödiger, T. Friedrichsmeier, P. Kapat, and M. Michalke. RKWard: A Comprehensive Graphical User Interface and Integrated Development Environment for Statistical Analysis with R. *Journal of Statistical Software*, 49(9):1–34, 2012. ISSN 1548-7660. URL http://www.jstatsoft.org/v49/i09. [p3, 11]

S. Rödiger, A. Böhm, and I. Schimke. Surface Melting Curve Analysis with R. *The R Journal*, 5(2):37–53, Dec. 2013a. URL http://journal.r-project.org/archive/2013-2/roediger-bohm-schimke.pdf. [p3]

S. Rödiger, P. Schierack, A. Böhm, J. Nitschke, I. Berger, U. Frömmel, C. Schmidt, M. Ruhland, I. Schimke, D. Roggenbuck, W. Lehmann, and C. Schröder. A highly versatile microscope imaging technology platform for the multiplex real-time detection of biomolecules and autoimmune antibodies. *Advances in Biochemical Engineering/Biotechnology*, 133:35–74, 2013b. ISSN 0724-6145. doi: 10.1007/10_2011_132. [p1, 6]

S. Rödiger, C. Liebsch, C. Schmidt, W. Lehmann, U. Resch-Genger, U. Schedler, and P. Schierack. Nucleic acid detection based on the use of microbeads: a review. *Microchimica Acta*, pages 1–18, 2014. ISSN 0026-3672, 1436-5073. doi: 10.1007/s00604-014-1243-4. URL http://link.springer.com/article/10.1007/s00604-014-1243-4. [p1, 6]

J. M. Ruijter, M. W. Pfaffl, S. Zhao, A. N. Spiess, G. Boggy, J. Blom, R. G. Rutledge, D. Sisti, A. Lievens, K. De Preter, S. Derveaux, J. Hellemans, and J. Vandesompele. Evaluation of qPCR curve analysis methods for reliable biomarker discovery: bias, resolution, precision, and implications. *Methods (San Diego, Calif.)*, 59(1):32–46, Jan. 2013. ISSN 1095-9130. doi: 10.1016/j.ymeth.2012.08.011. PMID: 22975077. [p1, 3]

J. M. Ruijter, P. Lorenz, J. M. Tuomi, M. Hecker, and M. J. B. v. d. Hoff. Fluorescent-increase kinetics of different fluorescent reporters used for qPCR depend on monitoring chemistry, targeted sequence, type of DNA input and PCR efficiency. *Microchimica Acta*, 181(13-14):1689–1696, Oct. 2014. ISSN 0026-3672, 1436-5073. doi: 10.1007/s00604-013-1155-8. URL http://link.springer.com/article/10.1007/s00604-013-1155-8. [p1]

A.-N. Spiess, C. Feig, and C. Ritz. Highly accurate sigmoidal fitting of real-time PCR data by introducing a parameter for asymmetry. *BMC Bioinformatics*, 9(1):221, Apr. 2008. ISSN 1471-2105. doi: 10.1186/1471-2105-9-221. URL http://www.biomedcentral.com/1471-2105/9/221/abstract. [p3]

P. M. Valero-Mora and R. Ledesma. Graphical User Interfaces for R. *Journal of Statistical Software*, 49(1):1–8, June 2012. ISSN 1548-7660. URL http://www.jstatsoft.org/v49/i01. [p3]

E. Viturro, C. Altenhofer, B. Zölch, A. Burgmaier, I. Riedmaier, and M. W. Pfaffl. Microfluidic high-throughput reverse-transcription quantitative PCR analysis of liver gene expression in lactating animals. *Microchimica Acta*, 181(13-14):1725–1732, Oct. 2014. ISSN 0026-3672, 1436-5073. doi: 10.1007/s00604-014-1205-x. URL http://link.springer.com/article/10.1007/s00604-014-1205-x. [p1]

M. Zeller, W.-C. L. A. Guazzelli, and G. Williams. PMML: An Open Standard for Sharing Models. *The R Journal*, 1(1):60–65, June 2009. URL http://journal.r-project.org/archive/2009-1/RJournal_2009-1_Guazzelli+et+al.pdf. [p2]

Stefan Rödiger (corresponding author)
Faculty of Natural Sciences
Brandenburg University of Technology Cottbus–Senftenberg
Senftenberg
Germany Stefan.Roediger@hs-lausitz.de

Michał Burdukiewicz
University of Wroclaw
Department of Biotechnoloy
Wroclaw
Poland michalburdukiewicz@gmail.com

Konstantin Blagodatskikh
All-Russia Research Institute of Agricultural Biotechnology
Center for collective use "Biotechnology"
Moscow
Russia k.blag@yandex.ru

Peter Schierack
Faculty of Natural Sciences
Brandenburg University of Technology Cottbus–Senftenberg
Senftenberg
Germany Peter.Schierack@hs-lausitz.de