Published online 31 July 2009

Nucleic Acids Research, 2009, Vol. 37, No. 12 ***1–7***
doi:10.1093/nar/gkn000

dpcR: web server and R package for analysis of digital PCR experiments

Michał Burdukiewicz [1,6], Jim Huggett [2], Alexandra Whale [2], Bart K.M. Jacobs [3], Lieven Clement [3], Piotr Sobczyk [1], Andrej-Nikolai Spiess [4], Peter Schierack [5], Stefan Rödiger [5*]

[1]Department of Genomics, Faculty of Biotechnology, University of Wrocław, Wrocław, Poland and [2]Molecular and Cell Biology Team, LGC, Teddington, United Kingdom and [3]Department of Applied Mathematics, Computer Science and Statistics, Ghent University, Belgium and [4]University Medical Center Hamburg-Eppendorf, Hamburg, Germany and [5]Faculty of Natural Sciences, Brandenburg University of Technology Cottbus–Senftenberg, Großenhainer Str. 57, 01968, Senftenberg, Germany

Received January 1, 2009; Revised February 1, 2009; Accepted March 1, 2009

ABSTRACT

The digital Polymerase Chain Reaction (dPCR) enables an absolute quantification of nucleic acids. Different statistical analysis frameworks were proposed. However, most analysis is done in closed source software as provided by the vendors. This makes it harder to compare results, such as the confidence interval estimates. An unified open software framework for reproducible research is not available.

To perform dPCR analysis we implemented peer-review statistical methods and plots into the dpcR framework, based on the sophisticated statistical computing environment **R**. dpcR is versatile open source cross-platform software framework, which provides functions to process dPCR data independent of the hardware. Our software can be used for data analysis and presentation, as framework for novel technical developments and as reference for statistical methods in dPCR analysis. Features such as functions to estimate the underlying Poisson process, calculation of confidence intervals based on single samples as well as on replicates, a novel Generalized Linear Model-based procedure to compare dPCR experiments and a spatial randomness test for assessing plate effects have been integrated. We use a plug-in like architecture and abstraction layers to make the framework usable for droplets and (real-time) chamber based technologies.

dpcR is implemented with interfaces to the command-line, graphical user interfaces and interactive web application. Therefore, it can be used by novices in a graphical user interface or by experts via a command-line interface. The dpcR framework can be used to build a custom-made analyser according to the user requirements. dpcR is an open framework, which can be easily adapted to the growing knowledge in dPCR.

INTRODUCTION

The digital PCR (dPCR) is an important contender for precise nucleic acids quantifications. Application of the dPCR include investigation of allele frequencies, single-cell analysis, gene expression analysis and absolute quantification of PCR products. The chemical basis (e.g., buffer, primer) of the dPCR and thermal cycling is similar to the real-time quantitative qPCR (qPCR). Though, approaches based on isothermal amplification were also developed (19, 26, 31). A first proposal for a dPCR-like approach and the use of the Poisson distribution to quantify the number of molecules on a "sample" was shown by Ruano *et al.* 1990 (PNAS) with the single molecule dilution (SMD) PCR (34). In 1999 Vogelstein *et al.* (PNAS) described the first true dPCR (40). Since approximately ten years the dPCR is rapidly gaining momentum in the mainstream user-base and will likely have the same impact as the qPCR methodology. There is an intensive research on dPCR platforms with the overall aim to make to technology broadly usable, cheap, robust and to enable high sample throughput (13, 24, 36).

On the opposite to qPCR, dPCR consists of multiple amplifications occurring in numerous small "partitions" (e.g., nl volume droplets of water oil emulsions, chambers on micro structured chips). The result of dPCR is a binary vector describing states of partitions (positive in case of detected amplification, negative otherwise). The amplification in positive partitions indicates the presence of one or more template molecules. It is assumed that distribution of template molecules over partitions is described appropriately by the Poisson distribution. This probability distribution is parametrized using only single parameter, λ, which may be interpreted as the mean number of template molecules per partition. The relationship between λ, the number of positive

*To whom correspondence should be addressed. Tel: +49 357385 936; Fax: +49 357385801; Email: stefan.roediger@b-tu.de

partitions k and the total number of partitions n is as follows:

$$\lambda = -\log\left(1 - \frac{k}{n}\right) \quad (1)$$

Knowing the λ and volume of the single partition, the computation of the concentration of a template in the sample seems to be trivial, yet methods of analysis of dPCR results are still emerging.

The variety of existing procedures is disproportionate to the number of software packages dedicated to analysis of dPCR experiments. Most of the software comes in form of closed-source packages distributed by vendors of dPCR systems. The proprietary software without open source code prohibits validation the implementation of the assumed methodology and therefore has a limited usage in research. Moreover, vendors usually tie tightly software with their hardware, which hinders a comparison of results from different systems.

The lack of a dedicated software for analysis of dPCR results leads to the rapid development of custom scripts in **Mathematica** (Wolfram Research) (38), **MS EXCEL** (Microsoft) (7) or **R** (9, 39). In addition, web-servers for analysis of dPCR data are emerging as the open access bioinformatic pipeline, designated **definetherain** (16). However, these efforts have limited use, since they are tied to a very specific problem. Even if they introduce some level of abstraction, it is not documented well enough to permit usage in other workflows.

The described situation is unfavorable for studies involving dPCR data. The absence of graphical user interfaces prohibits majority of researchers from using the custom-made frameworks. Others have to implement methods on their own, shifting their focus from an investigation to programming.

In 2013 we started the development of the *dpcR* framework to perform analysis of dPCR experiments (6). We have chosen R software environment (28), because it is already extensively used for studies of qPCR data (29, 31). Furthermore, R is freely available for the MacOS, Linux/Unix and Windows operating systems. It supports literate programming (e.g., *knitr*) and tools for reproducible research (e.g., *rctrack*) (18).

Since the R's command-line interface might not meet the need of all potential users, we added also a web server *dpcReport* using the *shiny* package. It allows access to the most of *dpcR* functionalities through the Graphical User Interface. *dpcReport* can be also installed as the stand-alone software.

MATERIALS AND METHODS

Implementation

We have chosen **R** because it is cross-platform and the *lingua franca* in applied statistical bioinformatics. Since all software is open source it is possible to track numerical errors (31, 33). Most **R** packages depend on other packages (25). The same holds true for *dpcR*. This results in a complex network of recursive dependencies (Figure 1A). Core packages include *qpcR* (29), *shiny*, *MBmca* (30), *chipPCR* (32). A basic design decision was to structure specific properties of dPCR systems (droplet vs. chamber) in auxiliary functions. Selected chamber dPCR systems rely on the proper preprocessing of qPCR data. This functionality is inherited from the *qpcR* **R** package.

The naming convention of *dpcR* is `underscore_sep` (1). The main *dpcR* functions (e.g., for analysis, simulations, plotting), several auxiliary functions (e.g., data import) and datasets of different dPCR systems are shown in the workflow of Figure 2. Figure 1B illustrates the implementation of the *dpcR* package. The function *read_dpcr*, *qpcr2pp* and *sim_dpcr*, sets the input for all objects in the `dpcr` class. Central calculation specific to Poisson statistics in are performed independently in main functions. This class manages details of the dPCR analysis that are subsequently processed by *dpcR* functions (e.g., reading/writing). Further details are explained in the supplementary information.

Figure 2 provides an overview of important functions available at each step of a dPCR analysis. The first step is to import sample data into the **R** session. *dpcR* accepts various data structures. This includes matrices of raw data and predefined structures provided by the different vendors (see Table 1). Moreover, *dpcR* accepts objects from the **R** workspace, which are converted via the *read_dpcr* function. Novel raw data structures are processed outside *dpcR* using tools as described elsewhere. The data input to *dpcR* should be raw data preferable, rather than gated summary data. The reasoning it to keep control over information loss for reproducible research.

Documentation

All functions of the *dpcR* package have its own documentation package, which specifies the input types, classes, parameters and output formats. The documentation is available as standard **R** package reference manual and as vignette.

According to the dMIQE guidelines (12) we used following notation:

- λ: average molecule numbers per partition,
- k: number of molecules per partition,
- m: total number of the molecules.

The following convention was kept, both in the documentation and in the package source code.

Import and export of results figures and data

Reading or importing data **R** has a rich set of tool to arrange data in order to prepare them for the analysis. This is important when it comes to the question how experiments should be treated (31).

Most *dpcR* import functions take a data matrix of annotated numeric (e.g, fluorescence amplitude) and integer values (e.g., counts) as input. The *raw_data* input was implemented make data input easier for the end user (see Supplement). Such input files can be manually created in a spreadsheet program or text editor.

There numerous methods to import amplification curve data into the **R** workspace (26, 27, 31). Prior processing the data for dPCR analysis it is recommended to use dedicated **R** packages, such as *chipPCR* (32), *MBmca* (30), *qpcR* (29), to pre-process (e.g., removal of missing values, smoothing (37)) the raw data.

(see Table 1).

Figure 1. Implementation of the *dpcR* package. **(B)** Modular software framework structure. *dpcR* is typically run from a desktop computer or a server. The software can be operated by an GUI/IDE application such as **RStudio** or **RKWard**. The *dpcR* package has dependencies to other **R** packages (middle layer). The functionality shared between the packages enables repaid addition and expansion of functionality.

Figure 2. *dpcR* workflow. The diagram shows main functions available at each step of a dPCR data analysis.

De novo creation of dpcR objects from raw data In experimental setups user have the need to transform their raw data in a processable format. Since the **R** environment is cross-platform an ubiquitously used we aimed to ease this creation. In particular, this is relevant for reproducible research. The *dpcR* framework covers the types typically used in laboratories.

Public data sets *dpcR* includes data sets or refers to additional R packages for testing purposes. The data originate from different dPCR and qPCR systems and were either published previously (8, 31, 32, 41, 42) or *de novo* generated.

Export of analysis results Since *dpcR* is based on the **R** environment all facilities for a report generation are usable as described before (31).

(see Table 1).

A commonly available data exchange format is a perquisite for reproducible research. So far no cross-platform and system-independent format has been introduced for dPCR experiments. Therefore, we decided to use `RData` as our default format. `RData` works across all **R** environments and saves the variable names along the content of multiple variables, which can be restored in any workspace.

Calculation of the uncertainty

To determine the uncertainty of the estimated λ we employ two previously published peer-reviewed methods (3, 10). The former is

Poisson distribution

Moments It is assumed that results from the dPCR reaction follow the Poisson distribution. We implemented several functions for computation of the four first moments (mean, variance, skewness, kurtosis) of the distribution from which the data is sampled. They can be calculated empirically from the sample data or as functions of the estimated λ.

Table 1. Structured vendor export data formats handled by *dpcR* v. 0.5 and later.

Vendor	System	Format	Type
Bio-Rad	QX100 & QX200	CSV	Summary export
Fluidigm	BioMark	CSV	Summary export
Formulatrix	Constellation Digital PCR	CSV	Summary export

The number of structured export data formats handled by *dpcR* is growing. Numerous data formats can be processed with the functionality provided by the **R** environment (see (31)). CSV, comma separated values.

Binomial approximation Since dPCR data contain binarized Poisson counts (all counts than zero are grouped together) it is possible to use methods appropriate for Binomial distribution. We employ several methods of computing confidence intervals (CI) for λ value (e.g., Wilson method, Agresti-Coull method). They especially advantageous when k is considerably smaller than m. They are beneficial for uncertainty calculation, because yielded CI's are not oscillating in such scenarios (4).

Simulation

The package covers two Simulation methods as described elsewhere in peer-reviewed publications. The method proposed by Dube *et al.* 2008 (10) is based on distributing template molecules over partitions. The procedure in Jacobs *et al.* 2014 (15) is more sophisticated and returns the fluorescence value obtains from droplet dPCR experiment including such phenomena as "rain". We also added streamlined and high-performing simulation in assumptions similar to Dube *et al.* 2008.

Comparison of dPCR experiments

In (5) We implemented statistics based on Generalized Linear Models (GLM) and *multiple ratio testing* (*MRT*) to compare dPCR experiments.

Spatial distribution

Array based dPCR experiments provide information about spatial distribution of partitions. Procedures belonging to spatial statistics verify if the status (positive, negative) of partition depends on its location. To address such questions, we implemented a Complete Spatial Randomness test for dPCR arrays. Moreover, we provide functionalities to export array dPCR data to **R** packages specialized in spatial statistics as described in (2).

Converting qPCRs to dPCRs

High-Throughput qPCR is a well established and robust technology, which allows precise quantification of DNA material in high throughput fashion. However, the quantification by qPCR is challenging at very low and very high concentrations. In addition, pre-processing and data analysis is a affected by numerous adverse effects (26, 35, 37). But the amplification curve real-time monitoring of the PCR product formation enables to determine quantification points (Cq), which can be binarized and analysed in the same way as dPCR experiments (23). This functionality is integral part of the *dpcR* package.

Figure3

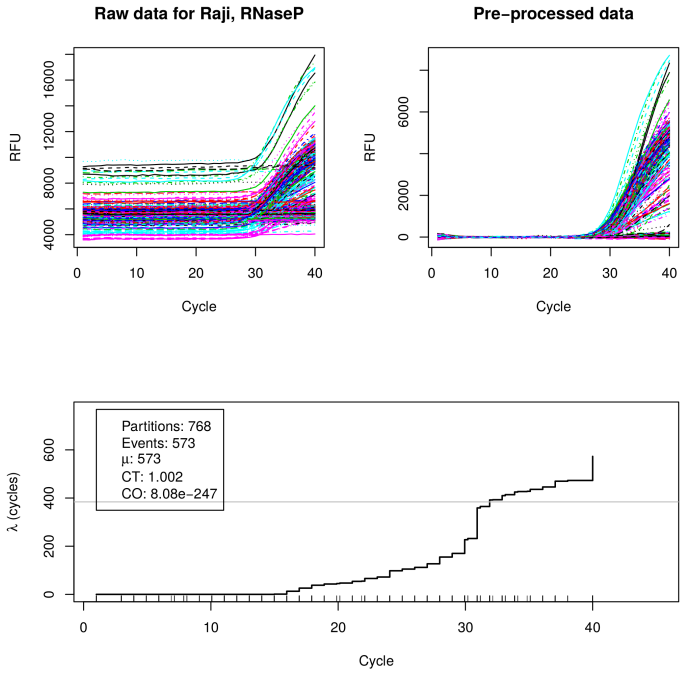

Figure 3. Uncover characteristics of dPCR data. Selected dPCR platforms are qPCR platforms at the same time. The function *qpcr2pp* uses the qPCR amplification curve data and interprets them as dPCR (Poisson process). A) Raw data of The function were B) preprocessed (baselined, smoothed) with functions from the *chipPCR* package and C) finally analysed (Cq calculation → binarize) with the *qpcr2pp* (qPCR to Poisson process) function from the *dpcR* package.

RESULTS

In the following section we show applications of the *dpcR* package.

Evaluation of dPCR comparison methods

TBD

Graphical user interface

The critical functionalities of *dpcR* package are implemented in the *dpcReport* GUI. We aimed for a form factor (e.g., smart phone, tablet, desktop PCR) and operating system independent implementation of a graphical user interface. *dpcReport* is based on *shiny* technology and offers an intuitive user interface, which can be accessed by browsers (e.g., Google Chrome, Mozilla Firefox). Semi-interactive charts are based on *ggplot2* (see Supplement).

The first panel "Input file" is responsible for loading the input data. Here part of data input structure (e.g., ID of experiments) may be accessed and modified during the analysis in the GUI. The second panel "Data summary" presents descriptive summary of the data in form of interactive tables and plots, offering filtering and selecting of individual runs. The third panel "Comparison of runs" compares all runs of the experiment data. The fourth panel "Advanced analysis" consists of more specialized functionalities as testing individual arrays. All statistical methods used in the GUI are integral part of the *dpcR* package and described in the methods section. The last panel enables flexible report generation. The report can be customized by including various sections, which are equivalents of the GUI panels.

An interesting feature of the *shiny* technology is the automatic integration in environments, which support `HTML5` and `ECMAScript`. The *dpcReport* integrates into Integration in third party software. This can be a modern web browser or an **R** IDE/GUI such as **RKWard** (Figure 4) (33) or **RStudio**. *dpcReport* is a GUI tool for dPCR data mining and report generation. User can interact via a point-and-click interface on different tabs, which contain widgets such as sliders, input fields and check boxes. Other user input, such as parameters of test performed in GUI, are preserved and returned in the report to increase the reproducibility of the research study. The tabs cover relevant analysis steps for the report generation. An important option of *dpcReport* is an export of the **R** source code used for the report generation is provided to the user. This code can be used for recreating the analysis in the **R** environment or prototyping more complicated workflows.

Vendor independent data analysis

QX100 series is no longer available. Instead, the newer version QX200 protocol consisting of the QX200 droplet generator (Bio-Rad, cat. no. 186-4002) and the QX200 droplet reader (Bio-Rad, cat. no. 186-4003) can be purchased, for which the protocol can be applied without changes. Alternative dPCR devices available from, e.g., RainDance Technologies, Life Technologies or JN Medsys (22).

Automatic report generation

In (31) we gave an example where we re-analyze droplet dPCR data from a Bio-Rad QX100 system with an early implementation of the *dpcR* package. As recommended in the dMIQE guidelines (12) we included key elements in the report.

Availability

The *dpcR* framework is available as open source software package (GPL-3 or later) as part of the Bioconductor project (11). The stable version is hosted at http://cran.r-project.org/web/packages/dpcR and the source code is available from https://github.com/michbur/dpcR.

DISCUSSION

Currently, there exist different dPCR analysis software solutions provided by the vendors. But most of the software packages are black boxes, which prevent deep insight into the data processing step. Other and we think that scientific software should be open (14, 31). In addition, most of the software solutions are aimed to be used in very specific scenarios and a mutual exclusive to alternative platforms (e.g., droplet vs. chamber-based). We have chosen **R** because it is the *lingua franca* in biostatistics and broadly used in other disciplines (31). We developed the *dpcR* package, which is a software framework for analysis of dPCR. *dpcR* provides the scientific community a broadly applicable tool for teaching purposes, data analysis and theoretical research based on simulations. Our software framework can be used to accelerate the development of new approaches to dPCR.

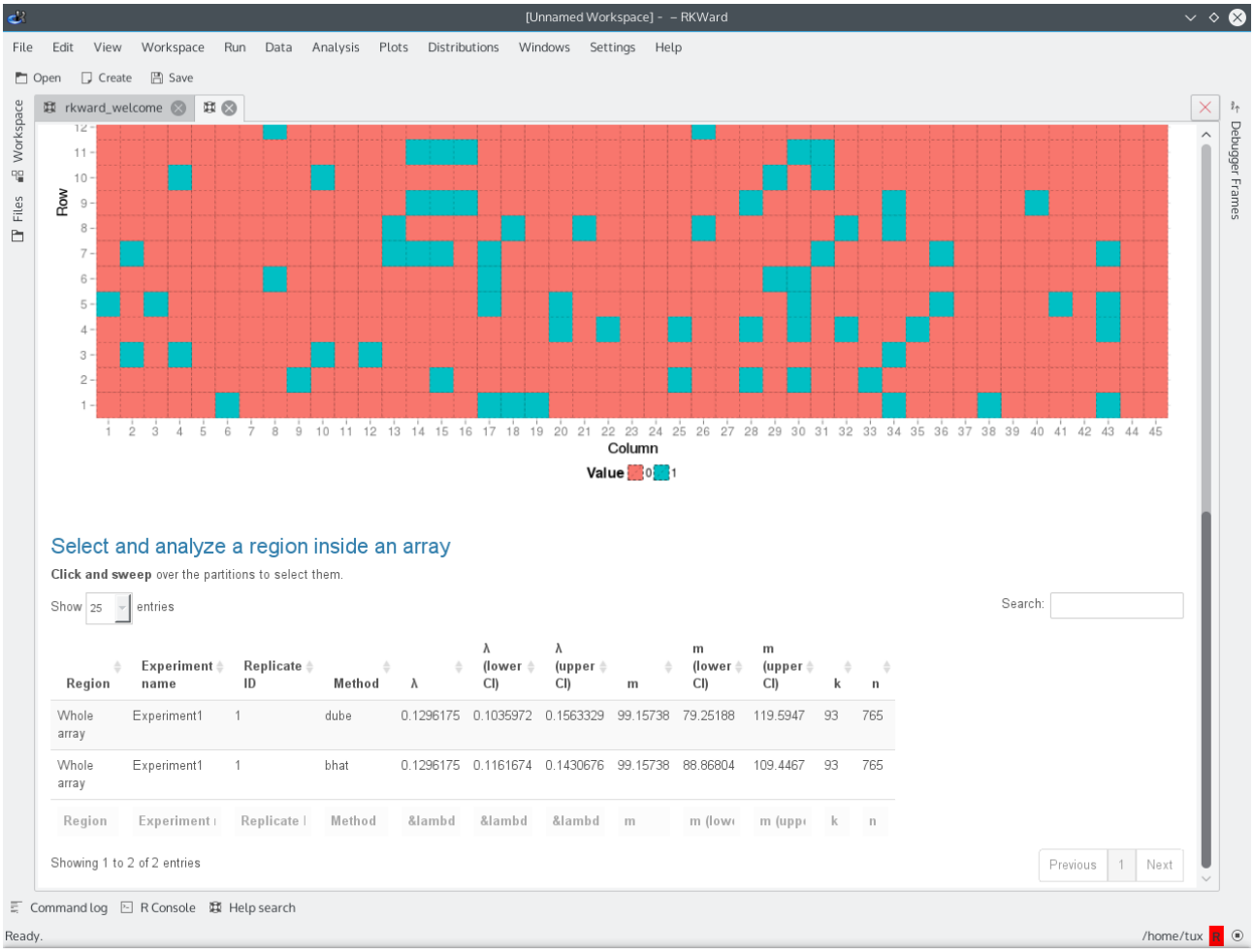

Figure 4. *dpcReport()* function running in the graphical user interface and integrated development environment **RKWard**.

Functions included may be used to simulate dPCRs, perform statistical data analysis, plotting of the results and simple report generation.

CONCLUSION

In conclusion, *dpcR* provides means to understand how dPCR works, to design, simulate and analyze experiments, and to verify their results (e.g., confidence interval estimation), which should ultimately improve reproducibility. We have built what we believe to be the first unified, cross-platform, dMIQE compliant, open source software framework for analysing dPCR experiments. Our *dpcR* framework is targeted at a broad user base including end users in clinics, academics, developers, and educators. We implemented existing statistical methods for dPCR and suggest the introduction of a standardized dPCR nomenclature. Our framework is suitable for teaching and includes references for an elaborated set of methods for dPCR statistics. Our software can be used for (I) data analysis and visualization in research, (II) as software framework for novel technical developments, (III) as platform for teaching this new technology and (IV) as reference for statistical methods with a standardized nomenclature for dPCR experiments. The framework enables the simulations and predictions of Poisson distribution for dPCR scenarios, the analysis of previously run dPCRs. Due to the plug-in structure of the software it is possible to build custom-made analysers.

We decided not to implement algorithms for clustering and "rain" (positive droplets) definition of droplet dPCR data. This is because, there are several **R** packages from flow-cytometer research. Implementations range from manual to automatic clustering (17, 20, 21, 39). Moreover, discussion with our peers and the literature suggest that a consensus of an appropriate method for dPCR is not available (39).

Our open framework includes to invitation to the scientific community to join and support the development of *dpcR*.

ACKNOWLEDGEMENTS

Grateful thanks belong to the **R** community and the **RStudio** developers.

FUNDING

This work was funded by the Federal Ministry of Education and Research (BMBF) InnoProfile–Transfer–Projekt 03 IPT 611X.

Conflict of interest statement. None declared.

REFERENCES

1. Rasmus Bååth. The State of Naming Conventions in R. *The R Journal*, 4(2):74–75, December 2012.
2. Adrian Baddeley, Ege Rubak, and Rolf Turner. *Spatial Point Patterns: Methodology and Applications with R*. Chapman and Hall/CRC Press, London, 2015. In press.
3. Somanath Bhat, Jan Herrmann, Paul Armishaw, Philippe Corbisier, and Kerry R Emslie. Single molecule detection in nanofluidic digital array enables accurate measurement of DNA copy number. *Analytical and Bioanalytical Chemistry*, 394(2):457–467, May 2009.
4. Lawrence D. Brown, T. Tony Cai, and Anirban DasGupta. Interval estimation for a binomial proportion. *Statist. Sci.*, 16(2):101–133, May 2001.
5. Michał Burdukiewicz, Stefan Rödiger, Piotr Sobczyk, Peter Schierack, and Paweł Mackiewicz. Methods of comparing digital PCR experiments. *Biomolecular Detection and Quantification*, 28(NN):NN–NN, tba.
6. Michał Burdukiewicz, Andrej-Nikolai Spiess, Peter Schierack, and Stefan Rödiger. dpcR: an R package for the analysis of digital PCR. *F1000Research*, 5, September 2013.
7. David Dobnik, Bjrn Spilsberg, Alexandra Bogoalec Koir, Arne Holst-Jensen, and Jana el. Multiplex Quantification of 12 European Union Authorized Genetically Modified Maize Lines with Droplet Digital Polymerase Chain Reaction. *Analytical Chemistry*, 87(16):8218–8226, August 2015.
8. Robert M. Dorazio and Margaret E. Hunter. Statistical Models for the Analysis and Design of Digital Polymerase Chain Reaction (dPCR) Experiments. *Analytical Chemistry*, 87(21):10886–10893, November 2015.
9. Tanja Dreo, Manca Pirc, iva Ramak, Jernej Pavi, Mojca Milavec, Jana el, and Kristina Gruden. Optimising droplet digital PCR analysis approaches for detection and quantification of bacteria: a case study of fire blight and potato brown rot. *Analytical and Bioanalytical Chemistry*, 406(26):6513–6528, August 2014.
10. Simant Dube, Jian Qin, and Ramesh Ramakrishnan. Mathematical analysis of copy number variation in a DNA sample using digital PCR on a nanofluidic device. *PloS one*, 3(8):e2876, 2008.
11. Robert C Gentleman, Vincent J Carey, Douglas M Bates, Ben Bolstad, Marcel Dettling, Sandrine Dudoit, Byron Ellis, Laurent Gautier, Yongchao Ge, Jeff Gentry, Kurt Hornik, Torsten Hothorn, Wolfgang Huber, Stefano Iacus, Rafael Irizarry, Friedrich Leisch, Cheng Li, Martin Maechler, Anthony J Rossini, Gunther Sawitzki, Colin Smith, Gordon Smyth, Luke Tierney, Jean Y H Yang, and Jianhua Zhang. Bioconductor: open software development for computational biology and bioinformatics. *Genome biology*, 5(10):R80, 2004.
12. Jim F Huggett, Carole A Foy, Vladimir Benes, Kerry Emslie, Jeremy A Garson, Ross Haynes, Jan Hellemans, Mikael Kubista, Reinhold D Mueller, Tania Nolan, Michael W Pfaffl, Gregory L Shipley, Jo Vandesompele, Carl T Wittwer, and Stephen A Bustin. The digital MIQE guidelines: Minimum information for publication of quantitative digital PCR experiments. *Clinical Chemistry*, 59(6):892–902, June 2013.
13. Jim F. Huggett, Justin OGrady, and Stephen Bustin. qPCR, dPCR, NGS A journey. *Biomolecular Detection and Quantification*, 3:A1–A5, March 2015.
14. Darrel C. Ince, Leslie Hatton, and John Graham-Cumming. The case for open computer programs. *Nature*, 482(7386):485–488, February 2012.
15. Bart KM Jacobs, Els Goetghebeur, and Lieven Clement. Impact of variance components on reliability of absolute quantification using digital pcr. *BMC Bioinformatics*, 15(1):283, 2014.
16. Mathew Jones, James Williams, Kathleen Gärtner, Rodney Phillips, Jacob Hurst, and John Frater. Low copy target detection by Droplet Digital PCR through application of a novel open access bioinformatic pipeline, definetherain. *Journal of Virological Methods*, 202(100):46–53, June 2014.
17. Nolwenn Le Meur. Computational methods for evaluation of cell-based data assessmentBioconductor. *Current Opinion in Biotechnology*, 24(1):105–111, February 2013.
18. Zhifa Liu, Stan Pounds, Zhifa Liu, and Stan Pounds. An R package that automatically collects and archives details for reproducible computing. *BMC Bioinformatics*, 15(1):138, May 2014.
19. Andrew T. Ludlow, Jerome D. Robin, Mohammed Sayed, Claudia M. Litterst, Dawne N. Shelton, Jerry W. Shay, and Woodring E. Wright. Quantitative telomerase enzyme activity determination using droplet digital PCR with single cell resolution. *Nucleic Acids Research*, 42(13):e104–e104, July 2014.
20. Mehrnoush Malek, Mohammad Jafar Taghiyar, Lauren Chong, Greg Finak, Raphael Gottardo, and Ryan R. Brinkman. flowdensity: reproducing manual gating of flow cytometry data by automated density-based cell population identification. *Bioinformatics*, 31(4):606–607, 2015.
21. Coren A. Milbury, Qun Zhong, Jesse Lin, Miguel Williams, Jeff Olson, Darren R. Link, and Brian Hutchison. Determining lower limits of detection of digital PCR assays for cancer-related gene mutations. *Biomolecular Detection and Quantification*, 1(1):8–22, September 2014.
22. Ulrike Mock, Ilona Hauber, and Boris Fehse. Digital PCR to assess gene-editing frequencies (GEF-dPCR) mediated by designer nucleases. *Nature Protocols*, 11(3):598–615, March 2016.
23. Mitra Mojtahedi, Aymeric Fouquierd'Hroul, and Sui Huang. Direct elicitation of template concentration from quantification cycle (Cq) distributions in digital PCR. *Nucleic Acids Research*, 42(16):e126–e126, September 2014.
24. Alexander A. Morley. Digital PCR: A brief history. *Biomolecular Detection and Quantification*, 1(1):1–2, 2014.
25. Jeroen Ooms. Directions for improved dependency versioning in R. *The R Journal*, 5(1):197–207, 2013.
26. Stephan Pabinger, Stefan Rödiger, Albert Kriegner, Klemens Vierlinger, and Andreas Weinhäusel. A survey of tools for the analysis of quantitative PCR (qPCR) data. *Biomolecular Detection and Quantification*, 1(1):23–33, 2014.
27. James R. Perkins, John M. Dawes, Steve B. McMahon, David LH Bennett, Christine Orengo, and Matthias Kohl. ReadqPCR and NormqPCR: R packages for the reading, quality checking and normalisation of RT-qPCR quantification cycle (Cq) data. *BMC Genomics*, 13(1):296, July 2012.
28. R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2016.
29. Christian Ritz and Andrej-Nikolai Spiess. qpcR: an R package for sigmoidal model selection in quantitative real-time polymerase chain reaction analysis. *Bioinformatics*, 24(13):1549–1551, January 2008.
30. Stefan Rödiger, Alexander Böhm, and Ingolf Schimke. Surface Melting Curve Analysis with R. *The R Journal*, 5(2):37–53, December 2013.
31. Stefan Rödiger, Michał Burdukiewicz, Konstantin A. Blagodatskikh, and Peter Schierack. R as an Environment for the Reproducible Analysis of DNA Amplification Experiments. *The R Journal*, 7(2):127–150, 2015.
32. Stefan Rödiger, Michał Burdukiewicz, and Peter Schierack. chipPCR: an R package to pre-process raw data of amplification curves. *Bioinformatics*, 31(17):2900–2902, 2015.
33. Stefan Rödiger, Thomas Friedrichsmeier, Prasenjit Kapat, and Meik 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.
34. G. Ruano, K. K. Kidd, and J. C. Stephens. Haplotype of multiple polymorphisms resolved by enzymatic amplification of single DNA molecules. *Proceedings of the National Academy of Sciences*, 87(16):6296–6300, January 1990.
35. Jan M Ruijter, Michael W Pfaffl, Sheng Zhao, Andrej N Spiess, Gregory Boggy, Jochen Blom, Robert G Rutledge, Davide Sisti, Antoon Lievens, Katleen De Preter, Stefaan Derveaux, Jan Hellemans, and Jo Vandesompele. Evaluation of qPCR curve analysis methods for reliable biomarker discovery: bias, resolution, precision, and implications. *Methods (San Diego, Calif.)*, 59(1):32–46, 2013.
36. David A. Selck, Mikhail A. Karymov, Bing Sun, and Rustem F. Ismagilov. Increased Robustness of Single-Molecule Counting with Microfluidics, Digital Isothermal Amplification, and a Mobile Phone versus Real-Time Kinetic Measurements. *Analytical Chemistry*, 85(22):11129–11136, November 2013.
37. Andrej-Nikolai Spiess, Claudia Deutschmann, Michał Burdukiewicz, Ralf Himmelreich, Katharina Klat, Peter Schierack, and Stefan Rödiger.

Impact of Smoothing on Parameter Estimation in Quantitative DNA Amplification Experiments. *Clinical Chemistry*, 61(2):379–388, January 2015.

38. Matthew C Strain, Steven M Lada, Tiffany Luong, Steffney E Rought, Sara Gianella, Valeri H Terry, Celsa A Spina, Christopher H Woelk, and Douglas D Richman. Highly precise measurement of HIV DNA by droplet digital PCR. *PloS one*, 8(4):e55943, 2013.
39. Wim Trypsteen, Matthijs Vynck, Jan De Neve, Pawel Bonczkowski, Maja Kiselinova, Eva Malatinkova, Karen Vervisch, Olivier Thas, Linos Vandekerckhove, and Ward De Spiegelaere. ddpcRquant: threshold determination for single channel droplet digital PCR experiments. *Analytical and Bioanalytical Chemistry*, 407(19):5827–5834, July 2015.
40. Bert Vogelstein and Kenneth W Kinzler. Digital PCR. *Proceedings of the National Academy of Sciences of the United States of America*, 96(16):9236–9241, August 1999.
41. Alexandra S Whale, Jim F Huggett, Simon Cowen, Valerie Speirs, Jacqui Shaw, Stephen Ellison, Carole A Foy, and Daniel J Scott. Comparison of microfluidic digital PCR and conventional quantitative PCR for measuring copy number variation. *Nucleic Acids Research*, 40(11):e82, June 2012.
42. Richard A White, 3rd, Paul C Blainey, H Christina Fan, and Stephen R Quake. Digital PCR provides sensitive and absolute calibration for high throughput sequencing. *BMC genomics*, 10:116, 2009.