The Journal of Open Source Software

PCRedux: A Quantitative PCR Machine Learning Toolkit

Michał Burdukiewicz [*1,2], **Andrej-Nikolai Spiess** [†3], **Dominik Rafacz** [4], **Konstantin Blagodatskikh** [5], **and Stefan Rödiger** [6,7¶]

1 Autonomous University of Barcelona, Bellaterra, Spain **2** Medical University of Białystok, Białystok, Poland **3** Soilytix GmbH, Hamburg, Germany **4** Warsaw University of Technology, Warsaw, Poland **5** Pirogov Russian National Research Medical University, Moscow, Russia **6** BTU Cottbus–Senftenberg, Faculty of Health Brandenburg, Senftenberg, Germany **7** BTU Cottbus–Senftenberg, Faculty Environment and Natural Sciences, Senftenberg, Germany ¶ Corresponding author

DOI: 10.21105/joss.04407

Software

- Review
- Repository
- Archive

Editor: Charlotte Soneson

Reviewers:

- @jaybee84
- @markziemann

Submitted: 11 May 2022
Published: 21 August 2022

Summary

qPCR (quantitative polymerase chain reaction) is indispensable in research, diagnostics and forensics, because it provides quantitative information about the amount of DNA in a sample (Pabinger et al., 2014). The interpretation of amplification curves (ACs) is often difficult if the curve does not follow a typical sigmoidal trajectory.

`PCRedux` is an `R` package (R Core Team, 2021) for feature extraction and classification in the realm of explainable machine learning, which uses statistical functions to compute 90 boolean and numerical descriptors from ACs. It can also be used to determine *Cq* values and amplification efficiencies (*E*) for high-throughput analysis.

Given the lack of class-labeled qPCR data sets, `PCRedux` includes functions for aggregation, management and dissemination of qPCR datasets that can, but must not necessarily be, trichotomously classified into *negative*, *positive* and *ambiguous* curves.

Statement of need

qPCR is a widely used laboratory method for the precise detection and quantification of pathogens and gene expression. The latter has contributed significantly to the understanding of physiological and pathological processes in pharmacology, medicine and forensics. (Kok et al., 2018; Pabinger et al., 2014) Although available software packages provide workflows and criteria for processing qPCR data (pre-processing of raw data, fitting of non-linear models, calculation of a threshold- or second derivative-based *Cq* or *E*, relative gene expression analysis, normalization procedures and data management), they lack functionality for machine learning. (Pabinger et al., 2014; Ramakers et al., 2003; Ruijter et al., 2013, 2021).

qPCR curves must meet quality criteria for analysis and are often categorized by the user according to rather subjective criteria (*e.g.*, sigmoidal shape, slope, noise, presence of a "hook effect") (Burdukiewicz et al., 2018; Hanschmann et al., 2021; Spiess et al., 2015, 2016). While positive qPCR reactions usually exhibit a sigmoidal shape, negative ACs display a rather flat and linear trajectory (Figure 1).

[*] Co-first author
[†] Co-first author

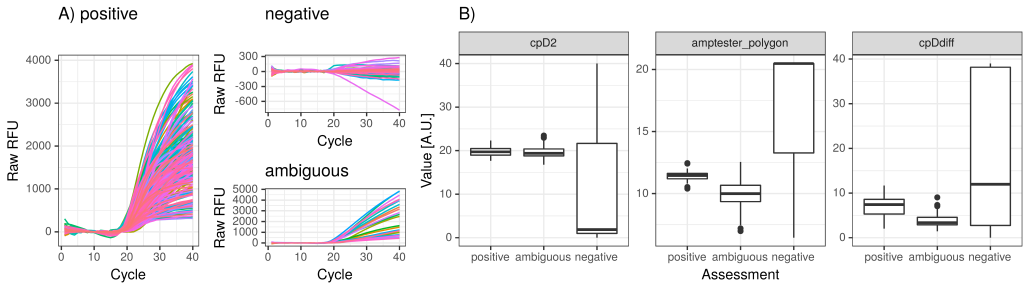


Figure 1: Analysis of ACs using the `PCRedux` package. A) ACs exhibit a high diversity in their appearance. The left plot (positive) shows ACs of which almost all are sigmoidal. The signal amplitude ranges from - 70 to 4000 relative fluorescence units (RFU). Of importance are those ACs that go slightly into the negative range of 10 - 20 cycles. Top right (negative) are negative ACs (signal amplitude between - 700 and 300) with noise (cycle 20 - 40). The bottom right plot (ambiguous) shows ACs that do not possess typical nonlinear slopes (non-sigmoid). These cannot be clearly classified as positive or negative. B) From A), the values of the three descriptors cpD2 (maximum of the second derivative, equals *Cq*), amptester_polygon (area under the curve) and cpDdiff (absolute cycle distance between the maximum of the second and first derivative) were calculated and plotted for the three classes. Data from htPCR dataset (Ritz & Spiess, 2008).

So how can ACs be objectively and reproducibly assessed and automatically interpreted (*e.g.*, as positive/negative/ambiguous or low/high quality)? For high-throughput experiments, manual evaluation is not feasible because of mental exhaustion errors or non-reproducibility from arbitrary thresholds or subjective assessments. While internal laboratory guidelines seem to partially remedy this, they are usually not standardized with other labs. (Bustin, 2010; Kim et al., 2018; Taylor et al., 2019).

Automatically extracted features from ACs (*e.g.*, *Cq* and *E*, slopes, change points, features of local curve segments) could provide a solid feature basis for classification by machine learning. Yet to date, no open-source software applies classical biostatistical methods for explainable machine learning on ACs. Furthermore, there are no class-labelled datasets that can be used in this context.

`PCRedux` is the first open-source software that can extract 90 mathematical descriptors (features) from raw ACs. The features are numerically or analytically derived, quantifiable, informative properties of scaled ACs in scalar units.

Software engineering

`PCRedux` (v.1.1-2, MIT license) is an `R` package (S3 class system). `R` was chosen because it provides comprehensive tools for reproducible statistical and bioinformatics analyses (R. C. Gentleman et al., 2004; R. Gentleman & Temple Lang, 2007; Leeper, 2014; Liu & Pounds, 2014; Rödiger, Burdukiewicz, Blagodatskikh, et al., 2015). Unit tests using the `testthat` package (Wickham, 2011) were used for software quality control of `PCRedux`.

Functions

Conceptually, we divide ACs into regions of interest (ROI) for feature calculation (Rödiger & PCRedux-package-authors (2022) Figure 5). Typical for qPCR, baseline, exponential/linear and plateau phases are located at the left, middle and right tail region of the curve, respectively (Rödiger & PCRedux-package-authors (2022) Figure 5).

`PCRedux`'s algorithms, published by others and ourselves (`qpcR` (Ritz & Spiess, 2008), `MBmca` (Rödiger et al., 2013), `chipPCR` (Rödiger, Burdukiewicz, & Schierack, 2015)), were adapted for qPCR analysis. The `PCRedux` dependencies include packages for preprocessing (`chipPCR`, MBmca), fitting of non-linear models and calculation of *Cq* and *E* (qpcR). pcrfit_single() is the workhorse function for single ACs that generates a *data.frame* with 90 descriptors. All output values are of type numeric, even if boolean. Among others, we included autocorrelation analyses, (Bayesian) change-point analyses (bcp (Erdman & Emerson, 2007), ecp (James & Matteson, 2015)), area determinations (pracma (Borchers, 2022)), regression (multi-parametric non-linear & robust local & regression models with segmented relationships: (robustbase (Todorov & Filzmoser, 2009), stats (R Core Team, 2021), segmented (Muggeo, 2017)) and hook effect detection (PCRedux (Burdukiewicz et al., 2018)).

encu() is an extension of pcrfit_single(), where meta information such as detection chemistry and platform is included, and is suitable for large data sets. Both functions are error-proof and utilize, among others, the following descriptor-generating functions:

- earlyreg(), calculates features by regression analysis in the background region,
- head2tailratio(), compares the ratio of head and tail,
- hookregNL() and hookreg(), try to detect a hook effect (Burdukiewicz et al., 2018),
- mblrr(), performs a local robust regression analysis,
- winklR(), calculates the angle based on the first, and the second derivative and
- autocorrelation_test(), tests for autocorrelation.

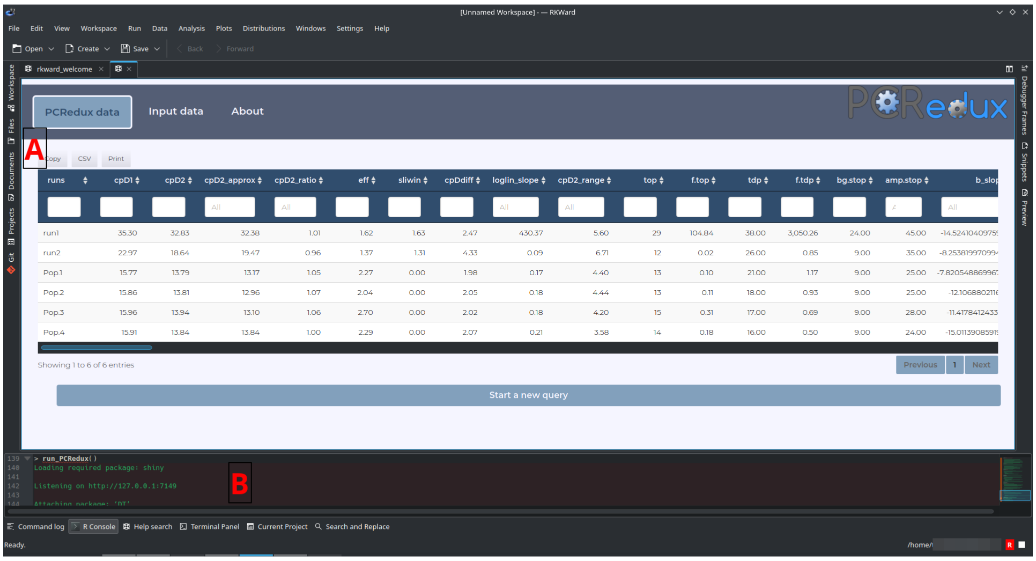
Auxiliary preprocessing and analysis functions of the package are:

- armor(), catches errors and creates the output,
- decision_mode(), calculates the frequency of classes in a dataset,
- qPCR2fdata(), converts AC data to the *fdata* format for Hausdorff distance analysis (Febrero-Bande & Oviedo de la Fuente, 2012) and
- performeR(), performs power analyses (e.g., sensitivity, specificity, Cohen's κ) for binary classification.

Application examples in the context of machine learning can be found in Rödiger & PCRedux-package-authors (2022) or the current Rödiger et al. (2022) vignette.

Graphical User Interface:

run_PCRedux() invokes a graphical user interface (Figure 2) based on the Shiny technology (Chang et al., 2021), providing features as a downstream accessible table.



runs	cpD1	cpD2	cpD2_approx	cpD2_ratio	eff	sliwin	cpDdiff	loglin_slope	cpD2_range	top	f.top	tdp	f.tdp	bg.stop	amp.stop	b_slop
			All	All				All	All						A	All
run1	35.30	32.83	32.38	1.01	1.62	1.63	2.47	430.37	5.60	29	104.84	38.00	3,050.26	24.00	45.00	-14.52410409759
run2	22.97	18.64	19.47	0.96	1.37	1.31	4.33	0.09	6.71	12	0.02	26.00	0.85	9.00	35.00	-8.253819970994
Pop.1	15.77	13.79	13.17	1.05	2.27	0.00	1.98	0.17	4.40	13	0.10	21.00	1.17	9.00	25.00	-7.820548869967
Pop.2	15.86	13.81	12.96	1.07	2.04	0.00	2.05	0.18	4.44	13	0.11	18.00	0.93	9.00	25.00	-12.1068802116
Pop.3	15.96	13.94	13.10	1.06	2.70	0.00	2.02	0.18	4.20	15	0.31	17.00	0.69	9.00	28.00	-11.4178412433
Pop.4	15.91	13.84	13.84	1.00	2.29	0.00	2.07	0.21	3.58	14	0.18	16.00	0.50	9.00	24.00	-15.01139085919

Showing 1 to 6 of 6 entries

```
> run_PCRedux()
Loading required package: shiny

Listening on http://127.0.0.1:7149

Attaching package: 'DT'
```

Figure 2: Graphical user interface for the analyses of qPCR data. A) The `run_PCRedux()` GUI for analysis and tabular display can use browsers or R environments that support `ECMA Script` and `HTML`. In this example, the GUI was used in `RKWard` (v.0.7.2, Linux, Kubuntu 21.10, (Rödiger et al., 2012)). B) Optionally, information about the current state of errors can be obtained via the R console.

Datasets and Data Labeling

`PCRedux` contains class-labeled ACs (n = 14360; label: negative, positive, ambiguous) from various qPCR instruments and detection methods, as determined by the majority vote of four experienced researchers (Rödiger & PCRedux-package-authors (2022)). Class labels were derived from the `humanrater2()` function, which uses `tReem()` for shape similarity calculation (Febrero-Bande & Oviedo de la Fuente, 2012).

Conclusion

Manual classification of ACs is time-consuming and error-prone, especially with large data sets where a significant proportion of curves deviate from sigmoidal shape, and where the results are influenced by subjective perception. An automated system for analyzing qPCR curves offers objectification and generalization of the decision-making process.

Here, training neural networks poses a viable option. The question is what the resulting network considers relevant, especially in a diagnostic scenario. To avoid these 'black box' situations, `PCRedux` enables a fast and computer-assisted classification of ACs based on 90 statistically and analytically founded descriptors, aiming to improve the quality and reproducibility of qPCR data analysis.

Acknowledgments

Grateful thanks belong to the `R` community.

Funding

None

References

Borchers, H. W. (2022). *Pracma: Practical numerical math functions*. https://CRAN.R-project.org/package=pracma

Burdukiewicz, M., Spiess, A.-N., Blagodatskikh, K. A., Lehmann, W., Schierack, P., & Rödiger, S. (2018). Algorithms for automated detection of hook effect-bearing amplification curves. *Biomolecular Detection and Quantification*, *16*, 1–4. https://doi.org/10.1016/j.bdq.2018.08.001

Bustin, S. A. (2010). Why the need for qPCR publication guidelines?—The case for MIQE. *Methods*, *50*(4), 217–226. https://doi.org/10.1016/j.ymeth.2009.12.006

Chang, W., Cheng, J., Allaire, J., Sievert, C., Schloerke, B., Xie, Y., Allen, J., McPherson, J., Dipert, A., & Borges, B. (2021). *Shiny: Web application framework for R*. https://CRAN.R-project.org/package=shiny

Erdman, C., & Emerson, J. W. (2007). Bcp: An R package for performing a Bayesian analysis of change point problems. *Journal of Statistical Software*, *23*(3), 1–13. https://doi.org/10.18637/jss.v023.i03

Febrero-Bande, M., & Oviedo de la Fuente, M. (2012). Statistical computing in functional data analysis: The R package fda.usc. *Journal of Statistical Software*, *51*(4), 1–28. http://www.jstatsoft.org/v51/i04/

Gentleman, R. C., Carey, V. J., Bates, D. M., Bolstad, B., Dettling, M., Dudoit, S., Ellis, B., Gautier, L., Ge, Y., Gentry, J., Hornik, K., Hothorn, T., Huber, W., Iacus, S., Irizarry, R., Leisch, F., Li, C., Maechler, M., Rossini, A. J., … Zhang, J. (2004). Bioconductor: Open software development for computational biology and bioinformatics. *Genome Biology*, *5*(10), R80. https://doi.org/10.1186/gb-2004-5-10-r80

Gentleman, R., & Temple Lang, D. (2007). Statistical Analyses and Reproducible Research. *Journal of Computational and Graphical Statistics*, *16*(1), 1–23. https://doi.org/10.1198/106186007X178663

Hanschmann, H., Rödiger, S., Kramer, T., Hanschmann, K., Steidle, M., Fingerle, V., Schmidt, C., Lehmann, W., & Schierack, P. (2021). LoopTag FRET Probe System for Multiplex qPCR Detection of Borrelia Species. *Life*, *11*(11), 1163. https://doi.org/10.3390/life11111163

James, N. A., & Matteson, D. S. (2015). Ecp: An R Package for Nonparametric Multiple Change Point Analysis of Multivariate Data. *Journal of Statistical Software*, *62*(1), 1–25. https://doi.org/10.18637/jss.v062.i07

Kim, Y.-M., Poline, J.-B., & Dumas, G. (2018). Experimenting with reproducibility: A case study of robustness in bioinformatics. *GigaScience*, *7*(7). https://doi.org/10.1093/gigascience/giy077

Kok, M. G. M., de Ronde, M. W. J., Moerland, P. D., Ruijter, J. M., Creemers, E. E., & Pinto-Sietsma, S. J. (2018). Small sample sizes in high-throughput miRNA screens: A common pitfall for the identification of miRNA biomarkers. *Biomolecular Detection and Quantification*, *15*, 1–5. https://doi.org/10.1016/j.bdq.2017.11.002

Leeper, T. J. (2014). Archiving Reproducible Research with R and Dataverse. *The R Journal*, *6*(1), 151–158. https://doi.org/10.32614/rj-2014-015

Liu, Z., & Pounds, S. (2014). An R package that automatically collects and archives details for reproducible computing. *BMC Bioinformatics*, *15*(1), 138. https://doi.org/10.1186/1471-2105-15-138

Muggeo, V. M. R. (2017). Interval estimation for the breakpoint in segmented regression: A smoothed score-based approach. *Australian & New Zealand Journal of Statistics*, *59*(3), 311–322. https://doi.org/10.1111/anzs.12200

Pabinger, S., Rödiger, S., Kriegner, A., Vierlinger, K., & Weinhäusel, A. (2014). A survey of tools for the analysis of quantitative PCR (qPCR) data. *Biomolecular Detection and Quantification*, *1*(1), 23–33. https://doi.org/10.1016/j.bdq.2014.08.002

R Core Team. (2021). *R: A language and environment for statistical computing*. R Foundation for Statistical Computing. https://www.R-project.org/

Ramakers, C., Ruijter, J. M., Deprez, R. H. L., & Moorman, A. F. M. (2003). Assumption-free analysis of quantitative real-time polymerase chain reaction (PCR) data. *Neuroscience Letters*, *339*(1), 62–66. https://doi.org/10.1016/S0304-3940(02)01423-4

Ritz, C., & Spiess, A.-N. (2008). qpcR: An R package for sigmoidal model selection in quantitative real-time polymerase chain reaction analysis. *Bioinformatics*, *24*(13), 1549–1551. https://doi.org/10.1093/bioinformatics/btn227

Rödiger, S., Böhm, A., & Schimke, I. (2013). Surface Melting Curve Analysis with R. *The R Journal*, *5*(2), 37–53. https://doi.org/10.32614/RJ-2013-024

Rödiger, S., Burdukiewicz, M., Blagodatskikh, K. A., & Schierack, P. (2015). R as an Environment for the Reproducible Analysis of DNA Amplification Experiments. *The R Journal*, *7*(2), 127–150. https://doi.org/10.32614/RJ-2015-011

Rödiger, S., Burdukiewicz, M., & Schierack, P. (2015). chipPCR: An R package to preprocess raw data of amplification curves. *Bioinformatics*, *31*(17), 2900–2902. https://doi.org/10.1093/bioinformatics/btv205

Rödiger, S., Burdukiewicz, M., Spiess, A.-N., & Blagodatskikh, K. A. (2022). PCRedux package - an overview [vignette]. *Comprehensive R Archive Network*, 1–104. https://cran.r-project.org/web/packages/PCRedux/vignettes/PCRedux.pdf

Rödiger, S., Friedrichsmeier, T., Kapat, P., & Michalke, M. (2012). RKWard: A comprehensive graphical user interface and integrated development environment for statistical analysis with R. *Journal of Statistical Software*, *49*(9), 1–34. https://doi.org/10.18637/jss.v049.i09

Rödiger, S., & PCRedux-package-authors. (2022). *PCRedux package - an overview*. https://doi.org/10.5281/zenodo.6957714

Ruijter, J. M., Barnewall, R. J., Marsh, I. B., Szentirmay, A. N., Quinn, J. C., van Houdt, R., Gunst, Q. D., & van den Hoff, M. J. B. (2021). Efficiency Correction Is Required for Accurate Quantitative PCR Analysis and Reporting. *Clinical Chemistry*, *67*(6), 829–842. https://doi.org/10.1093/clinchem/hvab052

Ruijter, J. M., Pfaffl, M. W., Zhao, S., Spiess, A. N., Boggy, G., Blom, J., Rutledge, R. G., Sisti, D., Lievens, A., De Preter, K., Derveaux, S., Hellemans, J., & Vandesompele, J. (2013). Evaluation of qPCR curve analysis methods for reliable biomarker discovery: Bias, resolution, precision, and implications. *Methods*, *59*(1), 32–46. https://doi.org/10.1016/j.ymeth.2012.08.011

Spiess, A.-N., Deutschmann, C., Burdukiewicz, M., Himmelreich, R., Klat, K., Schierack, P., & Rödiger, S. (2015). Impact of Smoothing on Parameter Estimation in Quantitative DNA Amplification Experiments. *Clinical Chemistry*, *61*(2), 379–388. https://doi.org/10.1373/clinchem.2014.230656

Spiess, A.-N., Rödiger, S., Burdukiewicz, M., Volksdorf, T., & Tellinghuisen, J. (2016). System-specific periodicity in quantitative real-time polymerase chain reaction data questions threshold-based quantitation. *Scientific Reports*, *6*(1), 38951. https://doi.org/10.1038/srep38951

Taylor, S. C., Nadeau, K., Abbasi, M., Lachance, C., Nguyen, M., & Fenrich, J. (2019). The Ultimate qPCR Experiment: Producing Publication Quality, Reproducible Data the First

Time. *Trends in Biotechnology*, *37*(7), 761–774. https://doi.org/10.1016/j.tibtech.2018.12.002

Todorov, V., & Filzmoser, P. (2009). An Object-Oriented Framework for Robust Multivariate Analysis. *Journal of Statistical Software*, *32*(3). https://doi.org/10.18637/jss.v032.i03

Wickham, H. (2011). Testthat: Get started with testing. *The R Journal*, *3*, 5–10. https://doi.org/10.32614/rj-2011-002