OXFORD

Briefings in Bioinformatics, 2024, **25(6)**, bbae507
https://doi.org/10.1093/bib/bbae507
Problem Solving Protocol

Digital PCR threshold robustness analysis and optimization using *dipcensR*

Matthijs Vynck[1,2,3,*], Wim Trypsteen[1,3], Olivier Thas[1,4,5,6], Jo Vandesompele[1,3,7], Ward De Spiegelaere[1,2,3,*]

[1]Digital PCR Center (DIGPCR), Ghent University, Ghent, Belgium
[2]Department of Morphology, Imaging, Orthopaedics, Rehabilitation and Nutrition, Faculty of Veterinary Medicine, Ghent University, Ghent, Belgium
[3]Cancer Research Institute Ghent, Ghent University, C. Heymanslaan 10, 9000 Ghent, Belgium
[4]Department of Applied Mathematics, Computer Science and Statistics, Krijgslaan 281 - S9, 9000 Ghent University, Ghent, Belgium
[5]Data Science Institute, Hasselt University, Agoralaan Gebouw D, 3590 Hasselt, Belgium
[6]National Institute for Applied Statistics Research Australia, University of Wollongong, Wollongong, NSW 2522, Australia
[7]Department of Biomolecular Sciences, Ghent University, C. Heymanslaan 10, 9000 Ghent, Belgium

*Corresponding authors. Matthijs Vynck, Digital PCR Center, Ghent University, Salisburylaan 133, entrance 78, 9820 Merelbeke, Belgium. E-mail: matthijs.vynck@ugent.be; Ward De Spiegelaere, Digital PCR Center, Ghent University, Salisburylaan 133, entrance 78, 9820 Merelbeke, Belgium. E-mail: ward.despiegelaere@ugent.be

Abstract

Digital polymerase chain reaction (dPCR) is a best-in-class molecular biology technique for the accurate and precise quantification of nucleic acids. The recent maturation of dPCR technology allows the quantification of up to thousands of targeted nucleic acids per instrument per day. A key step in the dPCR data analysis workflow is the classification of partitions into two classes based on their partition intensities: partitions either containing or lacking target nucleic acids of interest. Much effort has been invested in the design and tailoring of automated dPCR partition classification procedures, and such procedures will be increasingly important as the technology ventures into high-throughput applications. However, automated partition classification is not fail-safe, and evaluation of its accuracy is highly advised. This accuracy evaluation is a manual endeavor and is becoming a bottleneck for high-throughput dPCR applications. Here, we introduce *dipcensR*, the first data-analysis procedure that automates the assessment of any linear partition classifier's partition classification accuracy, offering potentially substantial efficiency gains. *dipcensR* is based on a robustness evaluation of said partition classification and flags classifications with low robustness as needing review. Additionally, *dipcensR*'s robustness analysis underpins (optional) automatic optimization of partition classification to achieve maximal robustness. A freely available R implementation supports *dipcensR*'s use.

Keywords: digital PCR; thresholding; partition classification; accuracy; multiplexing

Introduction

Digital polymerase chain reaction (dPCR) is a molecular biology technique that allows best-in-class accurate and precise quantification of nucleic acids, with broad cross-field applications [1–4]. While an old principle, its adoption has accelerated substantially over the past decade following advances in microfluidics [1, 4]. State-of-the-art implementations have considerably increased the reach of the method, now offering possibilities to quantify tens of distinct nucleic acid target sequences in a single reaction ("(higher-order) multiplexing"), and with throughput in the order of hundreds of samples per instrument per day [1, 4, 5]. Thanks to these and other key advantages, digital PCR increasingly replaces quantitative PCR [1–3].

A unifying aspect of dPCR across the range of implementations is the partitioning of the reaction mixture in, typically, tens of thousands of partitions before performing the PCR [1, 4]. In most current implementations, the fluorescence intensities in these partitions are subsequently measured after the PCR has ended ("endpoint fluorescence"), yielding a single fluorescence intensity value per partition per interrogated color [4]. These fluorescence intensities are a proxy of the presence (absence) of a (series of) nucleic acid target sequence(s). The term digital arises from making an informed dichotomization ("classification", "thresholding") of partitions based on those intensities, i.e. as either containing ("positive") or lacking ("negative") that (particular combination of) nucleic acid target sequence(s) [6]. The observed fraction of positive partitions ("partition occupancy") is finally used to estimate the target's concentration [1]. We refer the unacquainted reader to Huggett *et al.* [1] for an in-depth review of dPCR's concepts and terminology.

While manual partition classification is often feasible and straightforward, many efforts have focused on developing objective, automated partition classifiers [6]. Such classifiers are becoming increasingly important as the technology matures and moves toward multicolor, higher-throughput applications. Ample discussion on performance characteristics and application niches of these classifiers is available elsewhere [6–8]. Proper dichotomization is essential to dPCR's accuracy yet does not always result from automated partition classification [6–8]. The *post hoc* performance evaluation of partition classification accuracy remains wholly unaddressed; hence, such evaluations

Received: June 14, 2024. **Revised:** September 10, 2024. **Accepted:** October 3, 2024

are still a manual endeavor. With the aforementioned increase in throughput, both in the number of targets investigated within single reactions and the number of reactions, such manual scrutiny of classification accuracy is becoming a bottleneck in the dPCR data analysis pipeline.

Here, we aim to address this incipient bottleneck by proposing a simple and effective partition classification accuracy metric that can be used with any linear threshold method. In essence, the *dipcensR* algorithm assesses the robustness of the estimated target concentration to threshold perturbation: large changes in target concentration estimates for small threshold perturbations indicate a threshold that may not be optimally located or a reaction with, e.g. limited resolution. This robustness is intuitively visualized as a change in concentration versus perturbation plot, where a steeply declining curve corresponds to low robustness. The *dipcensR* metric d reflects the proportional change in nucleic acid concentration for a preset threshold perturbation window. This metric forms the basis of a flagging system that allows an agile identification of (i) potentially aberrant partition classification or (ii) a nucleic acid quantity that is particularly sensitive to the given partition classification. Such flagging is done on the single reaction-single color level and allows to select reaction–color pairs needing manual review. The flagging system aims to offer potentially substantial time and effort savings. Additionally, based on this accuracy metric, an optional robustness-maximizing classification adjustment procedure is described and evaluated and shown to correct aberrant classification in some instances.

Materials and methods

This section is organized as follows. The *dipcensR* quality control methodology is introduced in section The *dipcensR* Method. Building upon the quality control procedure, a related threshold adjustment procedure, aiming to improve threshold robustness, is described in section Optional: Robustness-Maximizing Threshold Adjustment. The section concludes with a description of method evaluation on experimental and simulated data in section Method Evaluation.

The *dipcensR* method

The *dipcensR* method is a threshold quality evaluation procedure (note: it is not a partition classifier) designed to verify linear threshold stability on a reaction-by-reaction, color-by-color basis. The description of the method focuses on the analysis of a single color, but *dipcensR* can be applied repeatedly to deal with reactions querying any number of colors (targets).

Figure 1 provides a flowchart illustrating the key steps of the *dipcensR* method. *dipcensR* starts from the raw partition-level fluorescence intensities and a predetermined linear threshold, the latter given by any partition classification method (Fig. 1a). Next, partitions with the most extreme, i.e. lowest and highest, fluorescence intensities are trimmed to increase the stability of the range estimate (Fig. 1b; details on this range stability are provided further). A sequence of perturbed thresholds is then calculated, with the range of that sequence based on the range of the remaining partition fluorescence intensities (Fig. 1c). For each of those perturbed thresholds, the change in partition occupancy arising from the perturbed threshold is calculated (Fig. 1d). Finally, reactions displaying low robustness, obvious from a steeply declining curve about the original threshold, are flagged for review.

Formally, let $y_i \ (i = 1, \ldots, n)$ denote a partition's endpoint fluorescence intensity for a given color and reaction, with n the total number of partitions. Define the sequence $Y = \langle y_1, \ldots, y_n \rangle$ as the observed partition intensities (re)indexed in monotone increasing order. Let t denote a predetermined threshold that dichotomizes the sequence Y into two subsequences: a sequence Y_{neg} of partitions assumed not to contain the nucleic acid target sequence(s) of interest (all elements for which $y_i < t$) and a sequence Y_{pos} of partitions assumed to contain the target nucleic acid(s) of interest (all elements for which $y_i \geq t$). Let $n_{neg} = \#(Y_{neg})$, $n_{pos} = \#(Y_{pos})$ denote the length of those sequences, respectively. The algorithm, named *dipcensR*, proceeds as follows:

1. If $n_{neg} \geq x$,i.e. the sequence contains at least x (default: 10) elements, remove from that sequence the elements $y_{neg,1}$ to $y_{neg,\lfloor n_{neg}/10 \rfloor}$ (where $\lfloor \ \rfloor$ denotes the floor function) to obtain Y_{neg^*}, i.e. trim the lower 10% elements.
2. If $n_{pos} \geq x$,i.e. the sequence contains at least x (default: 10) elements, remove from that sequence elements $y_{pos,n_{pos}-\lfloor n_{pos}/10 \rfloor}$ to $y_{pos,n_{pos}}$ to obtain Y_{pos^*}, i.e. trim the upper 10% elements.
3. Denote $Y^* = \{Y_{neg^*}, Y_{pos^*}\}$ as the combined sequences of trimmed negative and positive intensities. Calculate $r = \max(Y^*) - \min(Y^*)$, i.e. the range of Y^*.
4. Construct a perturbation sequence $P = \langle -0.5, -0.49, \ldots, 0.5 \rangle$ (step size default: 0.01). Calculate a sequence T_P that contains for each element $p_j \ (j = 1, \ldots, \#(P))$ of P the perturbed thresholds $t_j = t - r * p_j$.
5. For each element of T_P, calculate a sequence Z with elements $z_j = \frac{\sum_{Y^*} I(Y^* > t_j)/\dim(Y^*)}{\sum_{Y^*} (I(Y^* > t)/\dim(Y^*)}$, i.e. the reaction partition occupancy for a given perturbed threshold relative to the partition occupancy estimated using the original threshold. When $\sum_{Y^*} I(Y^* > t_j) = 0$, set $z_j = 10^{10}$, when $\sum_{Y^*} I(Y^* > t_j) = \#(Y^*)$ set $z_j = 10^{-10}$.
6. Calculate the difference in relative partition occupancy d in a window w (default, 0.2) about the initial threshold t: $d = z_k - z_l$,where k is the index j for which $p_j = -w/2$, and $l = j$ for which $p_j = w/2$.
7. Finally, assign the reaction–color pair a "green" flag (no review needed) if $d < c_1$, an "orange" flag (needs review) if $c_1 \leq d < c_2$, a "red" flag (needs review and likely needs threshold adjustment) if $d \geq c_2$. The scalars c_1 and c_2 are two cut-offs for flagging, with default values 0.1 and 0.2, respectively.

Further, a visualization of *dipcensR*'s output is provided as a line plot of the sequence Z as a function of the sequence P.

Optional: robustness-maximizing threshold adjustment

The *dipcensR* algorithm can optionally improve the specified threshold t by minimizing d. In particular, this is achieved as follows:

1. Minimize d in a neighborhood of t, which is defined as the interval in which the relative partition occupancy change z_j does not exceed a given fraction (default 50%, i.e. $(1/1.5 < z_j < 1.5)$).
2. Obtain a new threshold t_{new}, as the center of the interval obtained in Step 1.
3. A consequence of adjusting the threshold is that the estimated partition occupancy, i.e. the denominator used in the calculation of the sequence Z (The *dipcensR* Method, Step 5) will change. This necessitates a rerun of the *dipcensR* algorithm using the t_{new} threshold. A new estimate d results, which is used to assign a final flag, as before (see section The *dipcensR* Method).

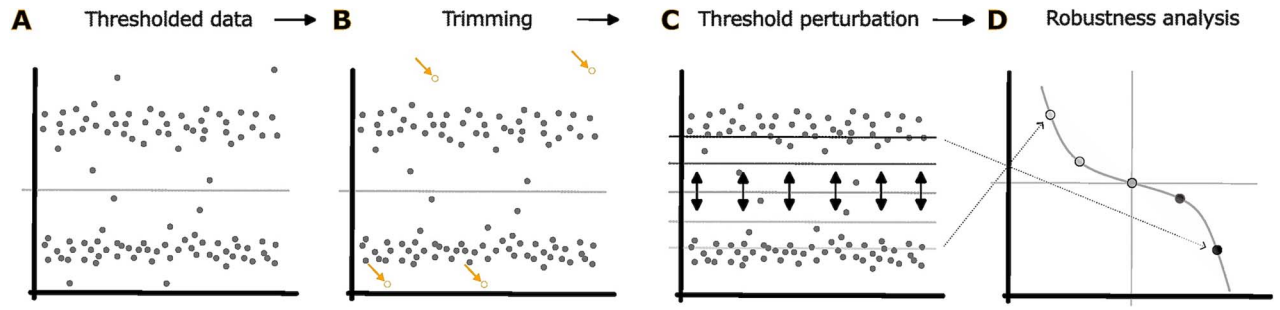

Figure 1. The *dipcensR* workflow. (a) *dipcensR*'s input consists of (i) individual partition fluorescence intensities and (ii) a (preliminary) threshold given by any (automatic) partition classifier. (b) Extreme partitions, i.e. partitions with very high or very low fluorescence intensities, are removed. (c) Based on the range of remaining fluorescence intensities, the initial threshold (Step a) is perturbed, leading to (d) a sensitivity curve displaying the impact of threshold perturbation (x-axis) on the estimated partition occupancy (y-axis); note that a lower threshold will lead to a higher partition occupancy, and vice versa (arrows).

Method evaluation

Selected cases

For demonstrating *dipcensR*'s flagging, example cases representing the different flags were selected from an analysis of a 48-sample, 6-color/6-target dilution series experiment. The reactions were run on a Nio + platform (Stilla Technologies, Villejuif, France) and analyzed with an in-house developed partition classification method (unpublished). Details are provided as Supplementary Material. Note that *dipcensR* can be used in conjunction with any linear thresholding procedure (see the discussion).

For verification of the threshold adjustment procedure performance, the thresholds of these cases were then adjusted (robustness-maximized) and re-evaluated. To show potential failure of the threshold adjustment procedure, erroneous thresholds were then manually assigned to these cases, and threshold robustness maximization was attempted.

Impact of trimming, concentration, resolution, and parameter selection

The impact of trimming on the hypothesized increased stability of range estimates, translating to increased flagging stability, was investigated. The range of partition intensities in 48 reactions was calculated before and after trimming. The spread of the ranges in both groups was then compared qualitatively by calculation of the ratio of median absolute deviations (after/before) and formally using the Brown–Forsythe Levene test (lawstat package version 3.6).

The impact of target concentration on flagging was investigated by *in silico* dilution of a single sample. In particular, fluorescence intensities from reaction F08 of *definetherain's* [9] benchmarking data (https://github.com/jacobhurst/definetherain/, obtained 3 April 2024) were used, the reaction serving as a reaction with high partition occupancy and clearly discerned negative and positive clusters. We performed *definetherain* to obtain an upper bound on negative partition intensities and a lower bound on positive partition intensities [9]. We proceeded to classify partitions as positive when they were above or equal to the mean of these two bounds and negative when below. *In silico* dilution was then achieved by resampling (with replacement) 15 000 intensities, with a given fraction of partitions from the set of positive (negative) partition intensities, to achieve partition occupancy in the set {0.001, 0.01, 0.1, 0.2, 0.3, 0.6, 0.9}. *dipcensR*'s relative partition occupancy difference d evaluation metric (see section The *dipcensR* Method) was then assessed as a function of partition occupancy. Resampling was performed 100 times for each of the partition occupancies.

To investigate the effect of resolution, samples with varying degrees of negative–positive cluster resolution and a range of partition occupancies (see previous paragraph) were constructed *in silico*. Different definitions of resolution in the context of dPCR experiments have been proposed [7, 10], but all have in common that a lower positive (negative) cluster variance results in a higher resolution. Here, specifically, the partition intensities' standard deviations (SDs) in both the positive and negative partition subset of reaction F08 (see previous paragraph) were rescaled by a factor in the set {1.00, 1.33, 1.66, 2.00}. The relative partition occupancy difference d was then assessed as a function of the rescaling value, the latter being an inverse proxy of the resolution.

The impact of parameter selection was investigated through a sensitivity analysis of the window w parameter and the flagging cut-offs c_1 and c_2 (section The *dipcensR* Method). For each of the four example reactions (section Selected Cases), a series of flags was obtained by varying w between 0.01 and 0.9 (default value: 0.2) and by multiplying c_1 and c_2 by a factor ranging from 0.1 to 2 (default values: 0.1 and 0.2, respectively, resulting in values 0.01–0.2 for c_1 and values 0.02–0.4 for c_2). Flagging consistency was verified by visualizing the flag color in a cut-off versus window plot.

Case study

The threshold as obtained in the section Impact of Trimming, Concentration, Resolution, and Parameter Selection was employed as the threshold for analyzing the entire set of reactions supplied by Jones *et al.* [9]. In particular, the dataset consists of a 6-point, 10-fold dilution series with four replicate reactions per dilution point and four blank samples. Each reaction consists of a one-color assay. We refer to Jones *et al.* [9] for more details. We then investigated *dipcensR* flagging performance by examining whether flagging was deemed appropriate (expert review).

Computation

All analyses were performed with an M1 (APL1102) processor, 16GB RAM, using R 4.3.3 [11]. *dipcensR*'s R code and the example datasets used in this work are freely available at https://www.github.com/DIGPCR/dipcensR. An R/Shiny interface to *dipcensR* is available at https://digpcr.shinyapps.io/dipcensR/.

Results and discussion

The *dipcensR* algorithm

The *dipcensR* algorithm calculates the impact of threshold perturbation on the estimated concentration and flags reaction–color pairs for review when there is a substantial change in

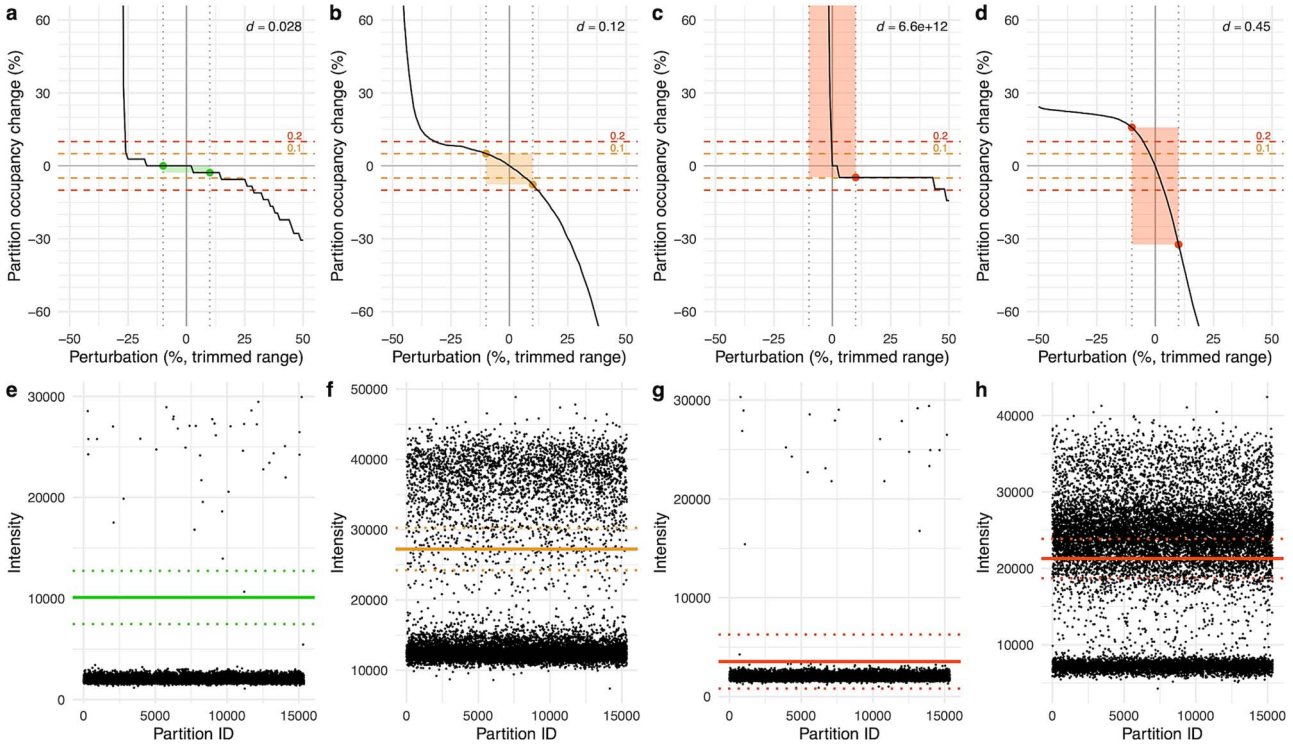

Figure 2. *dipcensR*'s robustness curve output (top row) for selected cases (Materials and Methods section Selected Cases), observed partition intensities, and thresholds (bottom row). (a–d): Vertical dotted lines indicate the window of perturbation considered for flagging (see Materials and Methods), whereas horizontal lines indicate warning levels (dashed, orange) or error levels (dashed, red); the horizontal $y = 0$ line corresponds to no perturbation/no change (see Materials and Methods). (e–h) Full lines indicate the thresholds, whereas dotted lines indicate the threshold at the boundaries considered for flagging; line color corresponds to the assigned flag (pass, warning, error). (a, e) Appropriate threshold location, with little impact of shifts, resulting in a pass flag (threshold color green). (b, f) Due to a reasonably large amount of rain and a threshold that could arguably be placed lower, there is an intermediate impact of threshold location on the estimated concentration, resulting in a warning flag (threshold color orange). (c, g) Due to a threshold located close to the negative cluster and few positive partitions, a small threshold shift (to lower intensities) will result in a large change in concentration, hence an error flag (threshold color red); note that the threshold here is not necessarily wrong. (d, h) Threshold placed erroneously within a cluster, leading to large changes in estimated concentration, and an error flag (threshold color red); here, the threshold needs to be corrected. An annotated version of the robustness curve visualization is available as Supplementary Fig. S2 available online at http://bib.oxfordjournals.org/.

estimated concentration under limited perturbations. Specifically, each reaction–color pair receives one of three flags: small changes ("green") indicate the robustness of the threshold to perturbations and indicate that such reaction–color pairs do not need review, whereas intermediate ("orange") changes indicate some effect, and large ("red") changes indicate a strong effect, suggesting that such reaction–color pairs need review (Fig. 2).

We find that intermediate changes ("orange" flag) are flagged typically when the amount of rain is not low, often in combination with a slightly misplaced threshold (Fig. 2b and e). Large changes ("red" flag) occur when (i) the threshold is placed aberrantly, e.g. too close to a cluster center (Fig. 1h), (ii) the concentration is low, and there is a reasonably large amount of rain, and (iii) the threshold is placed close to the negative cluster center, and the concentration is very low, or the target is absent (Fig. 2g), among others.

Investigation of *dipcensR*'s visual output provides a rapid diagnosis of whether a threshold is misplaced, e.g. showing a plateau above or below the $y = 0$ change in relative occupancy line (Fig. 1d), or whether a target–color combination is flagged because of a large amount of rain, i.e. no plateau is visible, but there is a steadily decreasing partition occupancy for increasing threshold values (Fig. 2b). A steep slope (Fig. 2c and d) indicates a threshold close to, or within the body of a cluster, or a reaction with limited resolution (see further), often resulting in a "red" flag.

The *dipcensR* plot provides multiple guidance components facilitating threshold robustness interpretation. Horizontal dashed lines indicate (user-adaptable) cut-off limits for reaction flagging (orange, red). A rectangle indicates the (user-adaptable) perturbation limits (*x*-axis, vertical dotted lines) and the corresponding observed change in relative partition occupancy (*y*-axis, full black robustness curve), the rectangle color corresponds to the assigned flag (green, orange, red). Last, the estimated relative partition occupancy change (the *d* metric) used for assigning the flag is shown.

The robustness of *dipcensR* is enhanced by trimming the partition intensities before calculating the level of perturbation to perform. This trimming removes any partitions that may occur as outlying values beneath the negative cluster's body or outlying values above the positive cluster's body. Such values that occur due to, e.g. droplet coalescence [12], are typically rare, and their intensity values are irreproducible [12, 13]. Failing to remove them can lead to unstable range estimates and, subsequently, erroneous flagging. In a 48-reaction series, the between-reaction median absolute deviation, a measure of spread, of the range after trimming was 52.7% lower than before trimming ($P < .001$, Brown–Forsythe Levene test), supporting that trimming improves the estimated range's stability.

dipcensR's computational load is limited, requiring ~0.1 s per reaction–color pair.

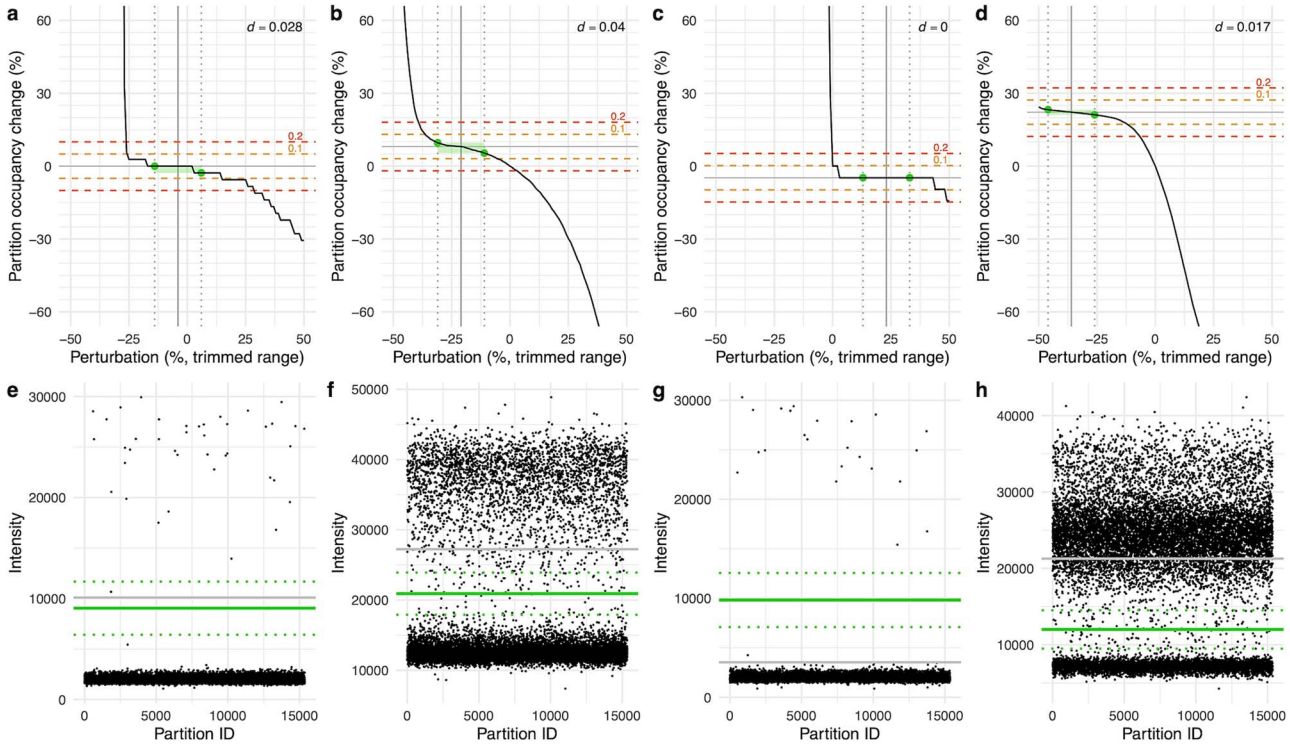

Figure 3. *dipcensR*'s robustness curve output (top row) and robustness-maximized thresholds (bottom row) for selected cases (as in Fig. 1, Materials and Methods section Selected Cases). (a–d) Vertical dotted lines indicate the window of perturbation considered for flagging (see Materials and Methods), whereas horizontal lines indicate warning levels (dashed, orange), or error levels (dashed, red); the horizontal y = 0 line corresponds to no perturbation/no change; all curves remain within the warning limits within the window of perturbation; for the original window of perturbation, partition occupancy change, and flag levels, see Fig. 1. (e–h) Full lines indicate the thresholds (gray: original, green: robustness-maximized), whereas dotted lines indicate the threshold at the boundaries considered for flagging, after threshold robustness maximization; the line color corresponds to the assigned flag (pass, warning, error); all thresholds receive a pass flag.

Robustness-maximizing threshold adjustment

A threshold is often not completely erroneous but may be located (too) close to the negative (positive) cluster center, and a warning or error flag may result (Fig. 2b–d). Small displacements of such thresholds, the maximal extent of which is a *dipcensR* user-determined parameter, may then increase the robustness and reduce the number of reaction–color pairs needing review. Indeed, in the example reaction–color pairs shown previously (Fig. 2), the robustness-maximizing threshold adjustment leads to more appropriate thresholds (by expert user interpretation) and pass flags (Fig. 3).

While successful for slightly erroneous thresholds (Figs 2 and 3), threshold adjustment may fail for completely erroneous thresholds, as the range of adjustment is constrained by the (user-determined) change in relative occupancy (Supplementary Fig. S1 available online at http://bib.oxfordjournals.org/). Such a constraint is needed as a perturbation zone with limited partition occupancy change below the negative or above the positive partition cluster can often be found, ultimately leading to a pass flag for an erroneous threshold. Consequently, the choice and validation of a suitable initial partition classifier remains crucial.

In addition, whether a robustness-maximizing threshold is the most appropriate location of a threshold is debatable. For example, rain is often assumed to represent partitions containing the nucleic acid target sequence(s) [6], and some partition classifiers have been designed to model such rain or the negative partition intensities only [9, 14, 15]. In practice, such efforts and their accuracy are just as well debated as intensities can be irreproducible across samples, e.g. due to sample-level matrix effects, therefore invalidating such rationales.

Impact of concentration, resolution, and parameter selection on flagging

Flagging by *dipcensR* will depend on target occupancy (concentration) for assays with an erroneously positioned threshold, as well as on resolution (Figs 4 and 5). Thresholds positioned close to the negative (positive) clusters may result in more prevalent flagging as the concentration decreases (increases), and this effect becomes more pronounced as the resolution decreases (Fig. 4c and d). Contrarily, for reactions with an appropriately positioned threshold and sufficient resolution, flagging will be stable independent of concentration (Fig. 4a). Note that even for such reaction characteristics, flagging can be more variable for very low or high partition occupancy due to the stochasticity of partitions with somewhat outlying partition intensities, resulting in a higher variance of the *dipcensR d* metric (Fig. 4; see next paragraph for a rationale).

Further, the presence of false positives, which often show intermediate fluorescence intensities, may cause higher flag rates in low-concentrated samples. Indeed, such false positives will have a high (relative) impact on the estimated concentration. This low robustness is visible as a steeply declining curve in the *dipcensR* plot (Fig. 5). Moreover, flagging can be more variable due to the stochasticity of false positives, i.e. when such false-positive partitions with intermediate intensities are present at a low rate, they may occur in some reactions but not others.

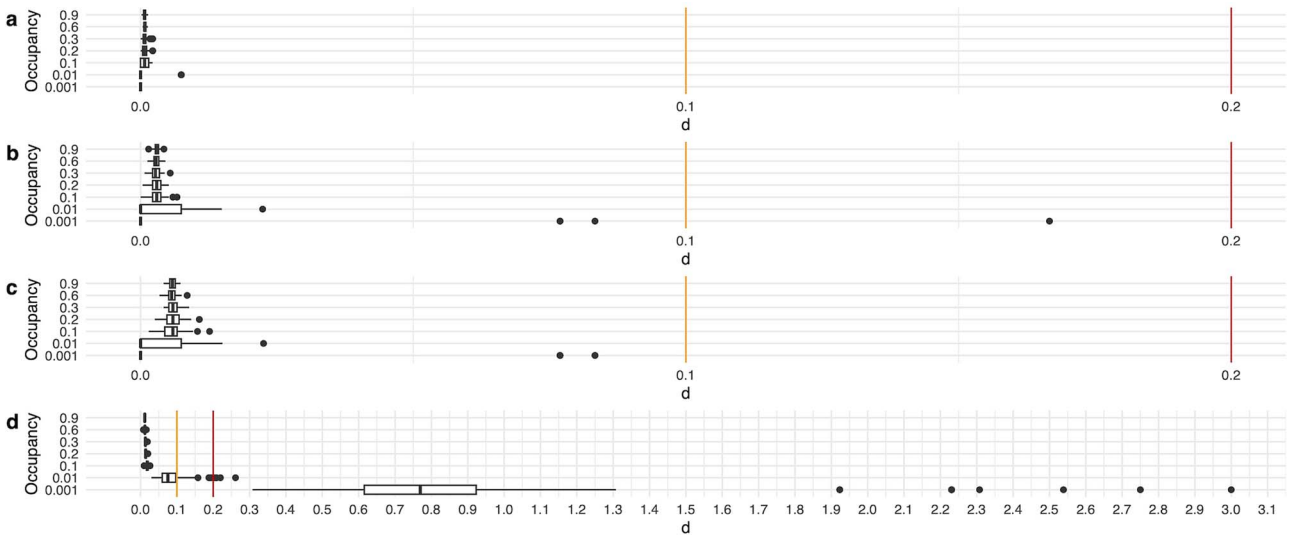

Figure 4. *dipcensR*'s d metric as a function of partition occupancy, for decreasing resolution (a-d; Materials and Methods section Impact of Trimming, Concentration, Resolution, and Parameter Selection); note the differences in x-axis scales, (a) dipcensR's d metric for cluster standard deviations (resolution) as in the experimental data (Materials and Methods), (b) for cluster SDs 33% larger, (c) for cluster SDs 66% larger, and (d) for cluster SDs 100% larger. Larger SDs correspond to a lower resolution.

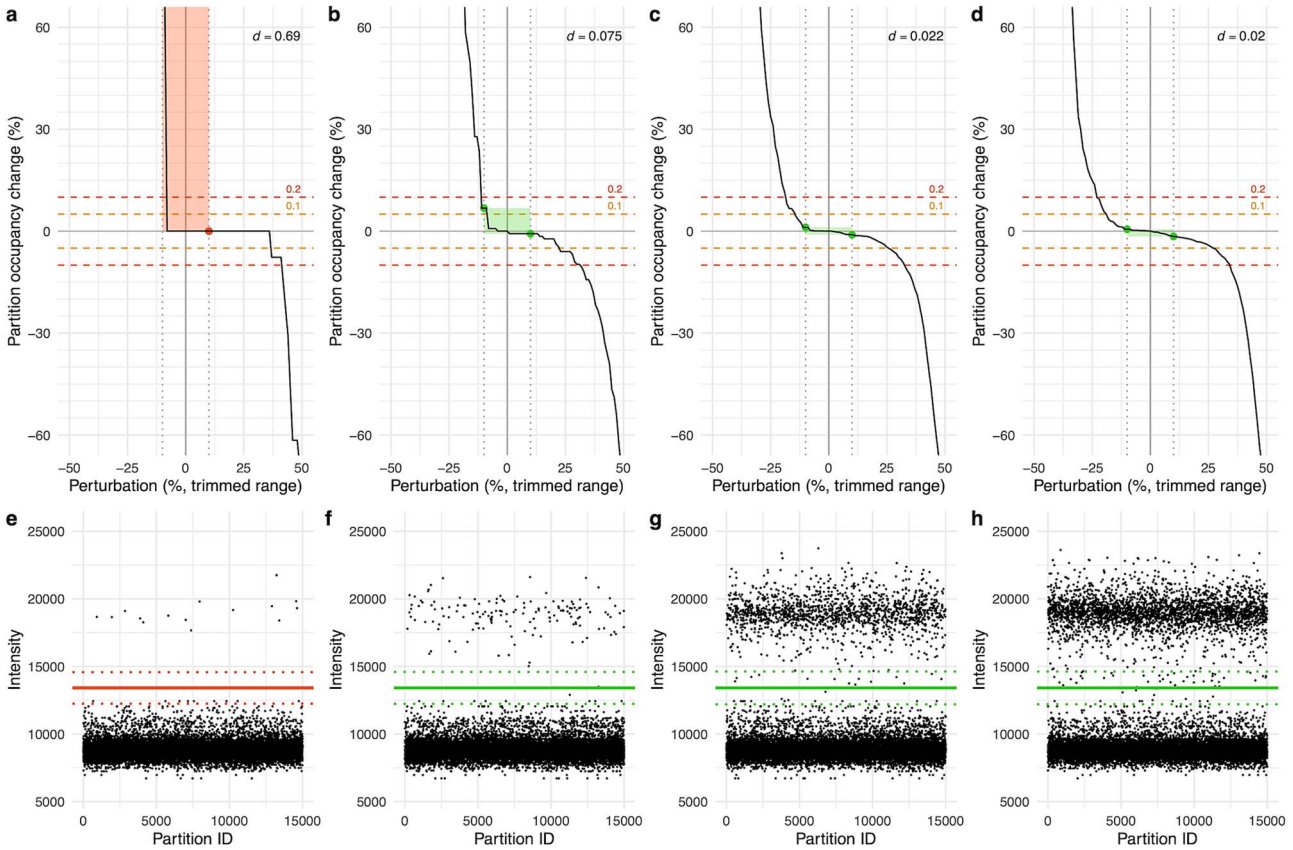

Figure 5. Examples of flagging when the resolution is lowest for increasing partition occupancy. (a, e) Example of a red flag obtained for a reaction with 0.001 partition occupancy and lowest resolution, (b, f) as before, but with 0.01 partition occupancy, (c, g) as before, but with 0.1 partition occupancy, now flagged green, and (d, h) as before, but with 0.2 partition occupancy.

Last, *dipcensR*'s default parameters were selected based on the analysis of in-house generated data, partly reported in this work. Flagging stringency can be increased by increasing either the window parameter w (see Materials and Methods), i.e. considering a wider threshold perturbation range, or by lowering the flagging cutoffs; conversely, more lenient flagging can be achieved by lowering the window parameter or increasing flagging cutoffs. Changing both equally in the same direction, to a limited extent, will typically preserve the flag, showcasing some robustness to parameter selection (Supplementary Fig. S3 available online at http://bib.oxfordjournals.org/). In summary, parameter tuning may be needed depending on assay characteristics and

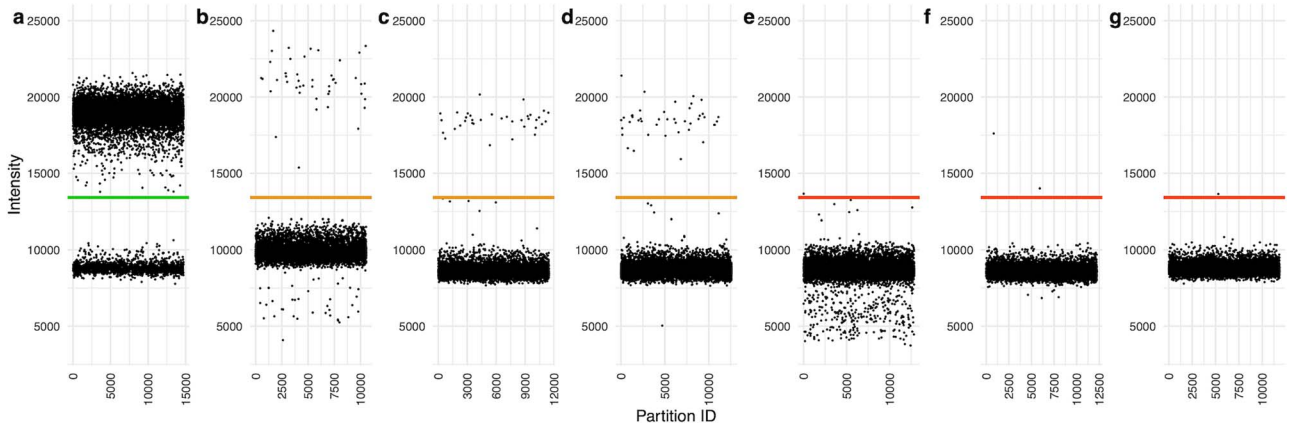

Figure 6. Partition intensities of the positive control used for estimating the threshold (a), and of samples flagged for investigation by dipcensR (b–g). (a) 10e5 input molecules, (b–d) 10e3 input molecules, (e) 10 input molecules, and (f, g) blank.

desired flagging stringency (Fig. 4, and further, section Limitations and Opportunities).

Case study: reaction flagging in a dilution series

Out of 28 reactions, three were flagged as suffering low robustness to threshold perturbation, and three were flagged as suffering intermediate robustness to threshold perturbation (Fig. 6). Low robustness occurred in samples with an expected (very) low concentration (Fig. 6e), or when the nucleic acid target sequence was presumed absent (Fig. 6f and g), and where some partitions showed intermediate intensities. Intermediate robustness was noted in samples with low concentrations but where (i) quite some partitions showed intermediate intensity (Fig. 6c and d) or (ii) where a fluorescence intensity baseline shift occurred, bringing the threshold in proximity to the negative cluster (Fig. 6b). In all cases except for the sample showing a baseline shift, flagging was deemed appropriate. A baseline shift correction before classification has been proposed [14, 15] and could have resulted in a pass flag for the reaction shown in Fig. 6b.

Application scope

The *dipcensR* algorithm is mainly aimed at facilitating high-throughput nucleic acid quantification using dPCR. For any target in a well-defined assay, parameters, such as partition intensity distribution and resolution (i.e. level of separation) of negative and positive partitions, are expected to be reasonably reproducible. Under such circumstances, the partition classification workload is often reduced by resorting to cross-well thresholding, based on a visual inspection of partition intensities that are merged across wells. However, concentration- or sample-specific effects may be present and can cause, e.g. location shifts in the partition intensity distribution (Fig. 6). Such sample-specific effects are often seen, where, e.g. the level of inhibitors may differ between samples. This can lead to some samples suffering more rain (partitions with intermediate fluorescence intensities) or a decrease in resolution [8, 13, 14]. Such changes may lead to aberrant partition classification or a classification that is especially sensitive to threshold location. Such reaction–color pairs then need a review.

Further, *dipcensR* analysis output could be used as a partition classifier performance metric: the fraction of stably classified, i.e. flagged green, reaction–color pairs could be a proxy of partition classifier performance, whereas, e.g. a high orange/red to green flag conversion after *dipcensR* threshold adjustment could indicate subpar partition classifier performance.

dipcensR operates on the individual reaction–color level, scrutinizing a single threshold per reaction and color. Nonetheless, *dipcensR*'s implementation allows it to be applied to a reaction with any number of colors, returning a flag for each color. Although *dipcensR* is robust to some differences in resolution (see Results and Discussion, section Robustness-Maximizing Threshold Adjustment), the resolution obtained may differ per color, and adjustment of *dipcensR*'s window parameter may be required (see Materials and Methods, section The *dipcensR* Method). Supplying single reaction–single color partition intensities together with more than one threshold allows *dipcensR* to handle threshold evaluation in assays using higher-order multiplexing strategies such as amplitude-based multiplexing [5]. For a simple example: in a single-color, two-target amplitude-based multiplexing assay, four clusters will arise, requiring the estimation of three thresholds [5]. To assess threshold robustness, *dipcensR* will need to be run once for each threshold. The appropriateness of each of those thresholds will be returned by *dipcensR*, informing the user of whether any of these thresholds need adjustment. Note that the resolution in an amplitude-based multiplexing assay will typically be lower than in a conventional one-color/one-target assay [5]. Consequently, like the previous example, *dipcensR*'s window parameter may need to be decreased for such assays.

Alternatives

dipcensR is the first procedure tailored for dPCR partition classification accuracy evaluation. While partition classification in dPCR experiments can be considered a general classification or clustering problem for which classification accuracy metrics exist, e.g. the Dunn [16] or silhouette index [17], *dipcensR* is distinct from such general classification accuracy metrics: *dipcensR*'s key idea is to take a(ny) metric, which is then followed by a sensitivity analysis in which the metric is used for quantifying the effect of perturbing a reaction's threshold. *dipcensR*'s default metric is the relative difference in partition occupancy (see Materials and Methods, section The *dipcensR* Method). While *dipcensR*'s metric could be exchanged for any other metric relating to classification accuracy, the explicit choice for change in partition occupancy is that it is easily understood by the dPCR user and that the user likely has an idea about what is an acceptable relative change in partition occupancy. In contrast, it is less intuitive what an acceptable change is in, e.g. (average) Dunn or silhouette index.

Limitations and opportunities

The *dipcensR* algorithm has some limitations. First, *dipcensR*'s default cut-offs for reaction–color flagging are arbitrary. Nevertheless, we have found these default values to provide desirable results in a range of datasets. Still, users may need to adapt these default values to individual assays depending on assay characteristics. As a counterpoint, because *dipcensR*'s parameters can be adjusted, its users can achieve a desired level of flagging stringency. Further, *dipcensR* will likely provide the most value in a high-throughput setting where manual reaction-by-reaction review becomes prohibitive. In such settings, we expect to see well-validated assays yielding reasonably stable characteristics, allowing the use of a single validated set of parameters.

Second, when a threshold is completely aberrant, for example, above the positive cluster intensities, threshold perturbation may not detect large changes and flag a reaction–color pair as needing no review. We have not seen this happen in the datasets we have analyzed, but such findings could occur depending on the choice of, e.g. partition classifier. If this is a concern, *dipcensR* has the option to flag reaction–color pairs with no positive partitions as needing review. As a counterpoint, in many applications, (some of) the nucleic acids' target sequence(s) will be absent, and, depending on the incidence of negative reaction–color pairs, this may substantially increase the number of reaction–color pairs needing review. Hence, *dipcensR*'s default option is that such reaction–color pairs do not need review. Note that *dipcensR* flags reaction–color pairs with no negative partitions as needing review because this indicates reaction saturation. In such cases, concentration estimation is impossible, and re-analysis of a diluted sample is indicated.

Third, *dipcensR* verifies linear threshold stability on a color-by-color basis. It cannot verify stability for methods that result in nonlinear thresholds, e.g. two-color methods that employ "distance to cluster center" measures to assign partition classes, e.g. elliptical cut-offs, such as *calico* [1, 18]. While systematic surveys on selection and use of thresholding procedures are nonexistent, it is our experience that performing linear thresholding on a color-by-color basis is most prevalent, and therefore, *dipcensR* will be applicable to the majority of experiments. Should multivariate partition classifiers for dPCR gain more traction, future research could focus on extending *dipcensR* to the multivariate space, investigating, e.g. pairwise cluster assignment stability.

Key Points

- Partition classification robustness can be assessed efficiently, and, likely, aberrant classification is flagged for user review by *dipcensR*.
- Partition classification robustness can often be enhanced automatically, reducing the number of reactions needing review; cases of severe misclassification cannot be automatically corrected, and, for such reactions, a review flag will be retained.
- *dipcensR* is flexible as (i) parameter tuning allows for varied levels of flagging stringency, allowing calibration to assay characteristics, and (ii) it is compatible with instruments/assays querying any number of targets (colors) in a "one target per color" setup, and with certain higher-order multiplexing setups such as amplitude-based multiplexing.

Acknowledgements

The authors thank Rémi Dangla and Florian Hébert (Stilla Technologies) for fruitful discussions regarding this work and for sharing data.

Supplementary data

Supplementary data are available at *Briefings in Bioinformatics* online.

Conflict of interest: None declared.

Funding

This work was funded in part by research funding from Stilla Technologies to O.T. and W.D.S. and research funding from Flanders Innovation and Entrepreneurship to W.D.S. (VLAIO, grant HBC_2022.0673).

Data availability

dipcensR's R code and the example datasets used in this work are freely available at https://www.github.com/DIGPCR/dipcensR.

References

1. Huggett JF, Whale AS, De Spiegelaere W. *et al.* The digital MIQE guidelines update: minimum information for publication of quantitative digital PCR experiments for 2020. *Clin Chem* 2020;**66**:1012–29. https://doi.org/10.1093/clinchem/hvaa125.
2. Salipante SJ, Jerome KR. Digital PCR—An emerging technology with broad applications in microbiology. *Clin Chem* 2020;**66**:117–23. https://doi.org/10.1373/clinchem.2019.304048.
3. Zhang L, Parvin R, Fan Q. *et al.* Emerging digital PCR technology in precision medicine. *Biosens Bioelectron* 2022;**211**:114344. https://doi.org/10.1016/j.bios.2022.114344.
4. Tiwari A, Ahmed W, Oikarinen S. *et al.* Application of digital PCR for public health-related water quality monitoring. *Sci Total Environ* 2022;**837**:155663. https://doi.org/10.1016/j.scitotenv.2022.155663.
5. Whale AS, Huggett JF, Tzonev S. Fundamentals of multiplexing with digital PCR. *Biomol Detect Quantif* 2016;**10**:15–23. https://doi.org/10.1016/j.bdq.2016.05.002.
6. Vynck M, Chen Y, Gleerup D. *et al.* Digital PCR partition classification. *Clin Chem* 2023;**69**:976–90. https://doi.org/10.1093/clinchem/hvad063.
7. Chen Y, De Spiegelaere W, Trypsteen W. *et al.* Benchmarking digital PCR partition classification methods with empirical and simulated duplex data. *Brief Bioinform* 2024;**25**:bbae120. https://doi.org/10.1093/bib/bbae120.
8. Kokkoris V, Vukicevich E, Richards A. *et al.* Challenges using droplet digital PCR for environmental samples. *Appl Microbiol* 2021;**1**:74–88. https://doi.org/10.3390/applmicrobiol1010007.
9. Jones M, Williams J, Gärtner K. *et al.* Low copy target detection by droplet digital PCR through application of a novel open access bioinformatic pipeline, 'definetherain'. *J Virol Methods* 2014;**202**:46–53. https://doi.org/10.1016/j.jviromet.2014.02.020.
10. Lievens A, Jacchia S, Kagkli D. *et al.* Measuring digital PCR quality: performance parameters and their optimization. *PLoS One* 2016;**11**:e0153317. https://doi.org/10.1371/journal.pone.0153317.
11. R Core Team. *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing, 2024. https://www.R-project.org/.

12. Chen J, Luo Z, Li L. *et al.* Capillary-based integrated digital PCR in picoliter droplets. *Lab Chip* 2018;**18**:412–21. https://doi.org/10.1039/C7LC01160A.
13. Witte AK, Fister S, Mester P. *et al.* Evaluation of the performance of quantitative detection of the listeria monocytogenes prfA locus with droplet digital PCR. *Anal Bioanal Chem* 2016;**408**:7583–93. https://doi.org/10.1007/s00216-016-9861-9.
14. Jacobs BKM, Goetghebeur E, Vandesompele J. *et al.* Model-based classification for digital PCR: your umbrella for rain. *Anal Chem* 2017;**89**:4461–7. https://doi.org/10.1021/acs.analchem.6b04208.
15. Trypsteen W, Vynck M, De Neve J. *et al.* ddpcRquant: threshold determination for single channel droplet digital PCR experiments. *Anal Bioanal Chem* 2015;**407**:5827–34. https://doi.org/10.1007/s00216-015-8773-4.
16. Dunn JC. A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters. *J Cybernetics* 1973;**3**:32–57. https://doi.org/10.1080/01969727308546046.
17. Rousseuw PJ. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Comput Appl Math* 1987;**20**:53–65. https://doi.org/10.1016/0377-0427(87)90125-7.
18. Lau BT, Wood-Bouwens C, Ji HP. Robust multiplexed clustering and denoising of digital PCR assays by data gridding. *Anal Chem* 2017;**89**:11913–7. https://doi.org/10.1021/acs.analchem.7b02688.