Package 'dPCP'

December 15, 2022

Title Automated Analysis of Multiplex Digital PCR Data

Version 2.0.0

Description The automated clustering and quantification of the digital PCR data is based on the combination of 'DBSCAN' (Hahsler et al. (2019) <doi:10.18637/jss.v091.i01>) and 'c-means' (Bezdek et al. (1981) <doi:10.1007/978-1-4757-0450-1>) algorithms.
The analysis is independent of multiplexing geometry, dPCR system, and input amount.
The details about input data and parameters are available in the vignette.

License MIT + file LICENSE

Encoding UTF-8

Depends R (>= 4.0.0)

Imports cluster, dbscan, e1071, exactci, ggplot2, ggpubr, graphics, raster, rlist, scales, shiny, shinyjs, stats, stringr, utils

RoxygenNote 7.2.1

Suggests knitr, rmarkdown, testthat

VignetteBuilder knitr

URL https://github.com/alfodefalco/dPCP

BugReports https://github.com/alfodefalco/dPCP/issues

NeedsCompilation no

Author Alfonso De Falco [aut, cre] (<https://orcid.org/0000-0001-9985-9920>),
Michel Mittelbronn [ctb],
Christophe Olinger [ctb],
Daniel Stieber [ctb],
Laboratoire national de santé [cph]

Maintainer Alfonso De Falco <alfonsodefalco90@gmail.com>

Repository CRAN

Date/Publication 2022-12-15 20:10:02 UTC

R topics documented:

`centers_data` *Prediction of clusters centroid position*

Description

This function calculates the coodintaes of all clusters centroid.

Usage

```
centers_data(sample.subquality, sample.table, referenceDB)

## S3 method for class 'centers_data'
plot(x, ..., sample = "all")
```

Arguments

`sample.subquality`	an object of class `read_sample`, inherited from `read_sample`.
`sample.table`	object of class `sample_table`, inherited from `read_sampleTable`.
`referenceDB`	an object of class `reference_dbscan`, inherited from `reference_dbscan`
`x`	an object of class `centers_data`
`...`	Arguments to be passed to methods
`sample`	'all' to show all samples, or a numeric vector indicating the row number of samples in the sample table.

Value

An object of class `centers_data` containing a sublist for each sample. Each sublist has the following components:

`quality`	quality threshold used in `read_sample`.
`reference`	reference ID.
`centers`	a data frame with the centroids coordinates.
`data`	a data frame with the fluorescence intensities.

Examples

```
library(dPCP)

#Find path of sample table and location of reference and input files
sampleTable <- system.file("extdata", "Template_sampleTable.csv",
                    package = "dPCP")

fileLoc <- system.file("extdata",package = "dPCP")

#Read sample table file
sample.table <- read_sampleTable(sampleTable, system = "bio-rad",
                                 file.location = fileLoc)

#Read reference files
ref <- read_reference(sample.table, system = "bio-rad",
                      file.location = fileLoc)

#Read samples files
samp <- read_sample(sample.table, system = "bio-rad",
                    file.location = fileLoc)

#Reference DBSCAN clustering
dbref <- reference_dbscan(ref, sample.table, save.template = FALSE)

#Predict position of clusters centroid from reference DBSCAN results
cent <- centers_data(samp, sample.table,dbref)

plot(cent, sample = "all")
```

`cmeans_clus` *Cluster analysis with fuzzy c-means algorithm*

Description

This function carries out the c-means cluster analysis, using the centroids position as initial values for cluster centers.

Usage

```
cmeans_clus(centers.data)

## S3 method for class 'cmeans_clus'
plot(x, ..., sample = "all", color.blind = FALSE)
```

Arguments

centers.data	an object of class centers_data, inherited from centers_data.
x	an object of class cmeans_clus
...	Arguments to be passed to methods
sample	'all' to show all samples, or a numeric vector indicating the row number of samples in the sample table.
color.blind	logical. If TRUE colors optimized for colorblind readers are used.

Value

An object of class cmeans_clus containing a sublist for each sample. Each sublist has the following components:

quality	quality threshold used in read_sample.
reference	reference ID.
centers	a data frame with the centroids coordinates.
data	a data frame with the fluorescence intensities and clusters name.
membership	a matrix with the membership values of the data elements to the clusters. See also cmeans

Examples

```
library(dPCP)

#Find path of sample table and location of reference and input files
sampleTable <- system.file("extdata", "Template_sampleTable.csv",
                    package = "dPCP")

fileLoc <- system.file("extdata",package = "dPCP")

#Read sample table file
sample.table <- read_sampleTable(sampleTable, system = "bio-rad",
                                 file.location = fileLoc)

#Read reference files
ref <- read_reference(sample.table, system = "bio-rad",
                      file.location = fileLoc)

#Read samples files
samp <- read_sample(sample.table, system = "bio-rad",
```

```
                    file.location = fileLoc)

#Reference DBSCAN clustering
dbref <- reference_dbscan(ref, sample.table, save.template = FALSE)

#Predict position of clusters centroid from reference DBSCAN results
cent <- centers_data(samp, sample.table,dbref)

#Fuzzy c-means clustering
cmclus <- cmeans_clus(cent)

plot(cmclus, sample = "all")
```

`dbscan_combination` *Test eps and minPts combinations for DBSCAN analysis*

Description

This function tests all combinations of eps and minPts for DBSCAN analysis of reference samples indicated in refID. The results are represented in scatterplots exported to a pdf file.

Usage

```
dbscan_combination(
  refID,
  system = NULL,
  file.location = ".",
  reference.quality = 0.5,
  eps = c(120, 150, 180, 200),
  minPts = c(20, 50, 80, 100)
)
```

Arguments

`refID`	a string or a character vector of chipID (Thermo Fisher) or the complete file name with the extension (Bio-Rad) of reference sample(s) to be analysed.
`system`	character. The name of digital PCR system used to generate the data. It must be either Thermo Fisher or Bio-Rad. Abbreviations are also accepted.
`file.location`	character. Full path name to reference and sample files location. The default corresponds to the working directory, (`getwd`). Tilde expansion (see (`path.expand`)) is performed.
`reference.quality`	numeric. Between 0 and 1. Quality threshold to subset the data (just for Thermo Fisher). If different thresholds have to be applied to various reference samples, a vectror of the same length of `refID` has to be provided.

eps	a numeric vector of values to be tested. Maximum distance between elements within a cluster in a DBSCAN analysis. See also dbscan.
minPts	a numeric vector of values to be tested. Number of minimum elements to assemble a cluster in a DBSCAN analysis. See also dbscan.

Value

A pdf file containing the scatterplots of DBSCAN analysis performed with all combinations of eps and minPts. Each reference generates a different pdf file.

Examples

```
library(dPCP)

#Find path of sample table and location of reference and input files
sampleTable <- system.file("extdata", "Template_sampleTable.csv",
                    package = "dPCP")

fileLoc <- system.file("extdata", package = "dPCP")

dbscan_combination("dilution20200313_B01_Amplitude.csv",
                   file.location = fileLoc, system = "bio-rad",
                   eps = c(150, 160, 180, 190), minPts = c(80, 100, 120))

unlink("dilution20200313_B01_Amplitude.pdf")
```

---

dPCP *Automated analysis of digital PCR data*

Description

This function carries out the autometed clustering of digital PCR data.

Usage

```
dPCP(
  file,
  system = NULL,
  file.location = ".",
  reference.quality = 0.5,
  sample.quality = 0.5,
  eps = 200,
  minPts = 50,
  save.template = FALSE,
  rain = TRUE,
  QC.reference = FALSE,
```

```
  partition.volume = NULL
)

## S3 method for class 'dPCP'
plot(
  x,
  ...,
  sample = "all",
  reference = "all",
  type = "dPCP",
  color.blind = FALSE
)
```

Arguments

file	character. The name or the path of csv file to be read. If it does not contain an absolute path, the file name is relative to the current working directory, (getwd).
system	character. The name of digital PCR system used to generate the data. It must be either Thermo Fisher or Bio-Rad. Abbreviations are also accepted.
file.location	character. Full path name to reference and sample files location. The default corresponds to the working directory, (getwd). Tilde expansion (see (path.expand)) is performed.
reference.quality	numeric. Between 0 and 1. Quality threshold to subset the data. If different thresholds have to be applied to various reference samples, a vectror of the same length of number of reference samples has to be provided. Used only when the system is Thermo Fisher.
sample.quality	numeric. Between 0 and 1. Quality threshold to subset data. If different thresholds have to be applied to various samples, a vectror of the same length of number of samples has to be provided. Used only when the system is Thermo Fisher.
eps	numeric. Input parameter for the DBSCAN algorithm. It represents the maximum distance between the elements within a cluster. See also dbscan. If different values have to be applied to various reference samples, a vectror of the same length of number of reference samples has to be provided.
minPts	numeric. Input parameter for the DBSCAN algorithm. It represents the number of minimum elements to assemble a cluster. See also dbscan. If different values have to be applied to various reference samples, a vectror of the same length of number of reference samples has to be provided.
save.template	logical. If TRUE a template of DBSCAN analysis of reference samples is saved. When system is Thermo Fisher, save.template can be also a character vector indicating the chipID.
rain	logical. If TRUE the rain analysis is carried out.
QC.reference	logical. If TRUE the fraction of rain elements in the reference samples is carried out. Warning messages are displayed when the percentage of rain is high.

partition.volume
: numeric. This parameters is taken into account when the parameter 'system' is set on Other. Indicate the partion volume in microliters spcific to the digital PCR system.

x
: an object of class dPCP

...
: Arguments to be passed to methods

sample
: 'all' to show all samples, or a numeric vector indicating the row number of samples in the sample table.

reference
: 'all' to show all reference samples, or a character vector with chip ID (Thermo Fisher) or the file name (Bio-rad) of reference samples to be showed.

type
: string. Type of plot to be showed. Available plots: 'reference dbscan', 'centers', 'cmeans', 'rain', 'dPCP'. @param color.blind logical. If TRUE colors optimized for colorblind readers are used.

color.blind
: logical. If TRUE colors optimized for colorblind readers are used.

Value

An object of class dPCP containing the following components:

referenceDB
: an object of class reference_dbscan.

samples
: a list of samples. Each sample sublist contains the information about the cluster analysis.

results
: an object of class replicates_quant.

Examples

```
library(dPCP)

#Find path of sample table and location of reference and input files
sampleTable <- system.file("extdata", "Template_sampleTable.csv",
                     package = "dPCP")

fileLoc <- system.file("extdata", package = "dPCP")

#dPCP analysis
results <- dPCP(sampleTable, system = "bio-rad", file.location = fileLoc,
                eps = 200, minPts = 50, save.template = FALSE, rain = TRUE,
                QC.reference = FALSE)

plot(results, sample = 1, type = "dPCP")
```

export_csv *Export dPCP analysis results to a csv file*

Description

This function exports dPCP analysis results to a csv file.

Usage

```
export_csv(data, filename)
```

Arguments

data	an object of class dPCP, target_quant or replicates_quant.
filename	character. File name (no extension) for csv and pdf files to create on disk.

Value

A csv file with the information and results of dPCP analysis.

Examples

```
library(dPCP)

#Find path of sample table and location of reference and input files
sampleTable <- system.file("extdata", "Template_sampleTable.csv",
                     package = "dPCP")

fileLoc <- system.file("extdata",package = "dPCP")

#dPCP analysis
results <- dPCP(sampleTable, system = "bio-rad", file.location = fileLoc,
                eps = 200, minPts = 50, save.template = FALSE,
                rain = TRUE)

export_csv(results, filename = "dPCRproject_1")
```

manual_correction *Manual correction of dPCP cluster analysis*

Description

This function builds an interactive app to manually correct the dPCP cluster analysis.

Usage

```
manual_correction(
  data,
  filename,
  save.plot = FALSE,
  format = "png",
  dpi = 300,
  color.blind = FALSE
)
```

Arguments

data	an object of class dPCP, inherited from dPCP.
filename	character. File name (no extension) for csv and pdf files to create on disk.
save.plot	logical. If TRUE the plots are exported to a file.
format	a string indicating the file format for the export. Available formats: 'eps', 'ps', 'tex', 'pdf', 'jpeg', 'tiff', 'png', 'bmp', 'svg', 'wmf'.
dpi	numeric. Image resolution.
color.blind	logical. If TRUE colors optimized for colorblind readers are used.

Value

A Shiny session.

Examples

```
library(dPCP)

#Find path of sample table and location of reference and input files
sampleTable <- system.file("extdata", "Template_sampleTable.csv",
                     package = "dPCP")

fileLoc <- system.file("extdata",package = "dPCP")

#dPCP analysis
results <- dPCP(sampleTable, system = "bio-rad", file.location = fileLoc,
                eps = 200, minPts = 50, save.template = FALSE,
                rain = TRUE)

manual_correction(results, filename = "manual_dPCR", save.plot = FALSE)
```

rain_reclus *Identification and clustering of "rain" data*

Description

This function identifies the "rain" elements and re-clusters them using the Mahalanobis distance. Each "rain" element is assigned to the cluster whose Mahalanobis distance is the lowest.

Usage

```
rain_reclus(cmeans.cluster)

## S3 method for class 'rain_reclus'
plot(x, ..., sample = "all", color.blind = FALSE)
```

Arguments

cmeans.cluster	an object of class cmeans_clus, inherited from cmeans_clus.
x	an object of class rain_reclus
...	Arguments to be passed to methods
sample	'all' to show all samples, or a numeric vector indicating the row number of samples in the sample table.
color.blind	logical. If TRUE colors optimized for colorblind readers are used.

Value

An object of class rain_reclus containing a sublist for each sample. Each sublist has the following components:

quality	quality threshold used in read_sample.
reference	reference ID.
centers	a data frame with the centroids coordinates.
data	a data frame with the fluorescence intensities and clusters name.

Examples

```
library(dPCP)

#Find path of sample table and location of reference and input files
sampleTable <- system.file("extdata", "Template_sampleTable.csv",
                    package = "dPCP")

fileLoc <- system.file("extdata",package = "dPCP")

#Read sample table file
```

```
sample.table <- read_sampleTable(sampleTable, system = "bio-rad",
                                 file.location = fileLoc)

#Read reference files
ref <- read_reference(sample.table, system = "bio-rad",
                      file.location = fileLoc)

#Read samples files
samp <- read_sample(sample.table, system = "bio-rad",
                    file.location = fileLoc)

#Reference DBSCAN clustering
dbref <- reference_dbscan(ref, sample.table, save.template = FALSE)

#Predict position of clusters centroid from reference DBSCAN results
cent <- centers_data(samp, sample.table,dbref)

#Fuzzy c-means clustering
cmclus <- cmeans_clus(cent)

#Rain classification.
rainclus <- rain_reclus(cmclus)

plot(rainclus, sample = "all")
```

`read_reference` *Read reference files*

Description

This function reads the results files of reference samples listed in the sample table. Fluoresce intensity and quality value (just for Thermo Fisher) are collected. If a `reference_dbscan` template file with the same input paramters (reference ID, eps, minPts) is available, fluorescence data, quality value and dbscan analysis results are retrived from the template file.

Usage

```
read_reference(
  sample.table,
  system = NULL,
  file.location = ".",
  reference.quality = 0.5,
  eps = NULL,
  minPts = NULL
)
```

Arguments

sample.table	object of class sample_table, inherited from read_sampleTable.
system	character. The name of digital PCR system used to generate the data. It must be either Thermo Fisher or Bio-Rad. Abbreviations are also accepted.
file.location	character. Full path name to reference and sample files location. The default corresponds to the working directory, (getwd). Tilde expansion (see (path.expand)) is performed.
reference.quality	numeric. Between 0 and 1. Quality threshold to subset the data. If different thresholds have to be applied to various reference samples, a vectror of the same length of number of reference samples has to be provided. Used only when the system is Thermo Fisher.
eps, minPts	numeric. Input parameters for the DBSCAN algorithm. If they match the paramters of reference_dbscan template file, the data are retrived from the template.

Value

An object of class read_reference containing a sublist for each reference. Each sublist has the following components:

quality	value of the reference.quality parameter.
data	a matrix with the fluorescence intensities and quality values.
dbscan	an object of class dbscan_fast, inherited from dbscan. This component is available only if a reference_dbscan template file is used to retrive the data.

Examples

```
library(dPCP)

#Find path of sample table and location of reference and input files
sampleTable <- system.file("extdata", "Template_sampleTable.csv",
                    package = "dPCP")

fileLoc <- system.file("extdata", package = "dPCP")

#Read sample table file
sample.table <- read_sampleTable(sampleTable, system = "bio-rad",
                                 file.location = fileLoc)

#Read reference files
ref <- read_reference(sample.table, system = "bio-rad",
                      file.location = fileLoc)
```

`read_sample` *Read sample files*

Description

This function reads the results files of samples listed in the sample table. Fluoresce intensity and quality value (just for Thermo Fisher) are collected.

Usage

```
read_sample(
  sample.table,
  system = NULL,
  file.location = ".",
  sample.quality = 0.5,
  partition.volume = NULL
)
```

Arguments

`sample.table`	object of class `sample_table`, inherited from `read_sampleTable`.
`system`	character. The name of digital PCR system used to generate the data. It must be either Thermo Fisher or Bio-Rad. Abbreviations are also accepted.
`file.location`	character. Full path name to reference and sample files location. The default corresponds to the working directory, (`getwd`). Tilde expansion (see (`path.expand`)) is performed.
`sample.quality`	numeric. Between 0 and 1. Quality threshold to subset data. If different thresholds have to be applied to various samples, a vectror of the same length of number of samples has to be provided. Used only when the `system` is Thermo Fisher.
`partition.volume`	numeric. This parameters is taken into account when the parameter 'system' is set on Other. Indicate the partion volume in microliters spcific to the digital PCR system.

Value

An object of class `read_sample` containing a sublist for each sample. Each sublist has the following components:

`quality`	value of the `sample.quality` parameter.
`data`	a matrix with the fluorescence intensities and quality values.

Examples

```
library(dPCP)

#Find path of sample table and location of reference and input files
sampleTable <- system.file("extdata", "Template_sampleTable.csv",
                    package = "dPCP")

fileLoc <- system.file("extdata", package = "dPCP")

#Read sample table file
sample.table <- read_sampleTable(sampleTable, system = "bio-rad",
                                 file.location = fileLoc)

#Read reference files
ref <- read_reference(sample.table, system = "bio-rad",
                      file.location = fileLoc)

#Read samples files
samp <- read_sample(sample.table, system = "bio-rad",
                    file.location = fileLoc)
```

`read_sampleTable` *Read sample table*

Description

This function reads a file containing the essential information about the samples and experimental settings. The file has to be filled out by the user and formatted as described in the vignette.

Usage

```
read_sampleTable(file, system = NULL, file.location = ".")
```

Arguments

`file`	character. The name or the path of csv file to be read. If it does not contain an absolute path, the file name is relative to the current working directory, (`getwd`).
`system`	character. The name of digital PCR system used to generate the data. It must be either Thermo Fisher or Bio-Rad. Abbreviations are also accepted.
`file.location`	character. Full path name to reference and sample files location. The default corresponds to the working directory, (`getwd`). Tilde expansion (see (`path.expand`)) is performed.

Value

An object of class `sample_table`.

Examples

```
library(dPCP)

#Find path of sample table and location of reference and input files
sampleTable <- system.file("extdata", "Template_sampleTable.csv",
                    package = "dPCP")

fileLoc <- system.file("extdata", package = "dPCP")

#Read sample table file
sample.table <- read_sampleTable(sampleTable, system = "bio-rad",
                                 file.location = fileLoc)
```

`reference_dbscan` *Find the empty partitions and single target clusters in the reference sample*

Description

This function computes a DBSCAN analysis to identify single target clusters in the reference samples listed in the sample table. If a `reference_dbscan` template file with the same input paramters (reference ID, eps, minPts) is available, data are retrived from the template file.

Usage

```
reference_dbscan(
  reference.subquality,
  sample.table,
  eps = 200,
  minPts = 50,
  save.template = FALSE
)

## S3 method for class 'reference_dbscan'
plot(x, ..., reference = "all")
```

Arguments

`reference.subquality`	an object of class `read_reference`, inherited from `read_reference`.
`sample.table`	object of class `sample_table`, inherited from `read_sampleTable`.
`eps, minPts`	numeric. Input parameters for the DBSCAN algorithm. If they match the paramters of `reference_dbscan` template file, the data are retrived from the template.
`save.template`	logical. If TRUE a template of DBSCAN analysis of reference samples is saved. When `system` is Thermo Fisher, `save.template` can be also a character vector indicating the chipID.

x	an object of class reference_dbscan
...	Arguments to be passed to methods
reference	'all' to show all reference samples, or a character vector with chip ID (Thermo Fisher) or the file name (Bio-rad) of reference samples to be showed.

Value

An object of class reference_dbscan containing a sublist for each reference. Each sublist has the following components:

quality	quality threshold used in read_reference.
data	a matrix with the fluorescence intensities and quality values.
dbscan	an object of class dbscan_fast, inherited from dbscan.

Examples

```
library(dPCP)

#Find path of sample table and location of reference and input files
sampleTable <- system.file("extdata", "Template_sampleTable.csv",
                    package = "dPCP")

fileLoc <- system.file("extdata",package = "dPCP")

#Read sample table file
sample.table <- read_sampleTable(sampleTable, system = "bio-rad",
                                 file.location = fileLoc)

#Read reference files
ref <- read_reference(sample.table, system = "bio-rad",
                      file.location = fileLoc)

#Read samples files
samp <- read_sample(sample.table, system = "bio-rad",
                    file.location = fileLoc)

#Reference DBSCAN clustering
dbref <- reference_dbscan(ref, sample.table, save.template = FALSE)

plot(dbref, reference = "all")
```

---

replicates_quant *Calculation of targets concentration, pooling the sample replicates*

Description

This function calculates the concentration of the targets, combining the results of the replicates of each sample.

Usage

```
replicates_quant(raw.results, sample.table)
```

Arguments

raw.results	an object of class target_quant, inherited from target_quant.
sample.table	object of class sample_table, inherited from read_sampleTable.

Value

An object of class replicates_quant containing a sublist for every sample. Each sublist has the following components:

quality	quality threshold used in read_sample.
reference	reference ID.
raw results	a data frame with the results of quantification.
replicates results	a data frame with the results of quantification of pooled replicates.

Examples

```
library(dPCP)

#Find path of sample table and location of reference and input files
sampleTable <- system.file("extdata", "Template_sampleTable.csv",
                     package = "dPCP")

fileLoc <- system.file("extdata", package = "dPCP")

#Read sample table file
sample.table <- read_sampleTable(sampleTable, system = "bio-rad",
                                 file.location = fileLoc)

#Read reference files
ref <- read_reference(sample.table, system = "bio-rad",
                      file.location = fileLoc)

#Read samples files
samp <- read_sample(sample.table, system = "bio-rad",
                    file.location = fileLoc)

#Reference DBSCAN clustering
dbref <- reference_dbscan(ref, sample.table, save.template = FALSE)

#Predict position of clusters centroid from reference DBSCAN results
cent <- centers_data(samp, sample.table,dbref)

#Fuzzy c-means clustering
cmclus <- cmeans_clus(cent)
```

```
#Rain classification.
rainclus <- rain_reclus(cmclus)

#Quantification
quantcm <- target_quant(cmclus, sample.table)
quant <- target_quant(rainclus, sample.table)

#Replicates pooling
rep.quant <- replicates_quant(quant, sample.table)
```

`report_dPCP` *Export dPCP analysis results to a pdf report*

Description

This function generates a pdf report of the dPCP analysis.

Usage

```
report_dPCP(data, filename, sample = "all", color.blind = FALSE)
```

Arguments

`data`	an object of class `dPCP`, inherited from `dPCP`.
`filename`	character. File name (no extension) for csv and pdf files to create on disk.
`sample`	'all' to show all samples, or a numeric vector indicating the row number of samples in the sample table.
`color.blind`	logical. If TRUE colors optimized for colorblind readers are used.

Value

A pdf file with the information and results of the dPCP analysis.

Examples

```
library(dPCP)

#Find path of sample table and location of reference and input files
sampleTable <- system.file("extdata", "Template_sampleTable.csv",
                    package = "dPCP")

fileLoc <- system.file("extdata", package = "dPCP")

#dPCP analysis
results <- dPCP(sampleTable, system = "bio-rad", file.location = fileLoc,
```

```
                eps = 200, minPts = 50, save.template = FALSE,
                rain = TRUE)

report_dPCP(results, filename = "dPCRproject_1")
```

`target_quant` *Calculation of targets concentration.*

Description

This function calculates the concentration of the targets according to the Poisson distribution.

Usage

```
target_quant(data.cluster, sample.table)
```

Arguments

`data.cluster`	an object of class `rain_reclus` or `cmeans_clus`.
`sample.table`	object of class `sample_table`, inherited from `read_sampleTable`.

Value

An object of class `target_quant` containing a sublist for each sample. Each sublist has the following components:

`quality`	quality threshold used in `read_sample`.
`reference`	reference ID.
`raw results`	a data frame with the results of the quantification.

Examples

```
library(dPCP)

#Find path of sample table and location of reference and input files
sampleTable <- system.file("extdata", "Template_sampleTable.csv",
                    package = "dPCP")

fileLoc <- system.file("extdata", package = "dPCP")

#Read sample table file
sample.table <- read_sampleTable(sampleTable, system = "bio-rad",
                                 file.location = fileLoc)

#Read reference files
ref <- read_reference(sample.table, system = "bio-rad",
                      file.location = fileLoc)
```

```
#Read samples files
samp <- read_sample(sample.table, system = "bio-rad",
                    file.location = fileLoc)

#Reference DBSCAN clustering
dbref <- reference_dbscan(ref, sample.table, save.template = FALSE)

#Predict position of clusters centroid from reference DBSCAN results
cent <- centers_data(samp, sample.table,dbref)

#Fuzzy c-means clustering
cmclus <- cmeans_clus(cent)

#Rain classification.
rainclus <- rain_reclus(cmclus)

#Quantification
quantcm <- target_quant(cmclus, sample.table)
quant <- target_quant(rainclus, sample.table)
```

Index