# Package: PoDCall (via r-universe)

November 25, 2026

**Type** Package

**Title** Positive Droplet Calling for DNA Methylation Droplet Digital PCR

**Version** 1.22.0

**Description** Reads files exported from 'QX Manager or QuantaSoft' containing amplitude values from a run of ddPCR (96 well plate) and robustly sets thresholds to determine positive droplets for each channel of each individual well. Concentration and normalized concentration in addition to other metrics is then calculated for each well. Results are returned as a table, optionally written to file, as well as optional plots (scatterplot and histogram) for both channels per well written to file. The package includes a shiny application which provides an interactive and user-friendly interface to the full functionality of PoDCall.

**License** GPL-3

**Encoding** UTF-8

**Depends** R (>= 4.5)

**Imports** ggplot2, gridExtra, mclust, diptest, rlist, shiny, DT, LaplacesDemon, purrr, shinyjs, readr, grDevices, stats, utils, dplyr

**Suggests** knitr, rmarkdown, testthat, BiocStyle

**VignetteBuilder** knitr

**biocViews** Classification, Epigenetics, ddPCR, DifferentialMethylation, CpGIsland, DNAMethylation,

**StagedInstall** no

**Config/roxygen2/version** 8.1.0

**Config/pak/sysreqs** cmake make libuv1-dev libxml2-dev libx11-dev zlib1g-dev

**Repository** Bioconductor 3.24

**Date/Publication** 2026-11-23 22:46:12 UTC

**RemoteUrl** https://github.com/bioc/PoDCall

**RemoteRef** RELEASE_3_24

**RemoteSha** a93954b40f85d74ff855c51c2f8a5371f74d4525

# Contents

importAmplitudeData . . . . . 2
importSampleSheet . . . . . 3
podcallChannelPlot . . . . . 4
podcallDdpcr . . . . . 5
podcallHistogram . . . . . 6
podcallMultiplot . . . . . 7
podcallScatterplot . . . . . 8
podcallShiny . . . . . 9
podcallThresholds . . . . . 10
thrTable . . . . . 11

**Index** **13**

---

`importAmplitudeData` *importAmplitudeData*

---

## Description

importAmplitudeData

## Usage

```
importAmplitudeData(
  dataDirectory,
  skipLines = c(0, 4),
  nrChannels = c(1, 2)[2],
  targetChannel = c(1, 2, 3, 4, 5, 6)[1],
  controlChannel = c(1, 2, 3, 4, 5, 6)[2]
)
```

## Arguments

| | |
|---|---|
| `dataDirectory` | Path to directory containing Quantasoft amplitude files from one 96 well plate. Since well coordinates are used as identifiers, files in this directory should all be from the same 96 well plate. Furthermore, there should be no other files than the amplitude files from a well plate in the directory. |
| `skipLines` | Number of lines to skip in amplitude data files. Must be 0 or 4 depending on software used to export data. 0 for QuantaSoft, 4 for QXmanager. |
| `nrChannels` | Number of channels/dyes used. Default nrChannels=2 |
| `targetChannel` | The channel nr used as target channel (default=1) |
| `controlChannel` | The channel nr used as control channel (default=2) |

## Value

The function returns a list of dataframes named with the well ID and contains the amplitude values from the corresponding well.

## Examples

```
# Path to example data files included in PoDCall
path <- system.file("extdata", "Amplitudes/", package="PoDCall")

# Read in data files
dataList <- importAmplitudeData(dataDirectory=path, skipLines=0)
```

---

`importSampleSheet` *importSampleSheet*

---

## Description

Function that takes a path to a .csv-file containing information about the samples that correspond to the uploaded amplitude files. This file must contain the following columns: Well, Sample, TargetType and Target. A character vector with well IDs must also be provided, which is used to match rows in sample sheet to amplitude files

## Usage

```
importSampleSheet(
  sampleSheet = NULL,
  well_id = NULL,
  software = c("QuantaSoft", "QX Manager")[2],
  targetChannel = c(1, 2, 3, 4, 5, 6)[1],
  controlChannel = c(1, 2, 3, 4, 5, 6)[2]
)
```

## Arguments

| | |
|---|---|
| `sampleSheet` | Path to sample sheet file containing information about samples. |
| `well_id` | Character vector with well IDs corresponding to uploaded amplitude files. |
| `software` | Name (character) of software data and sample sheet was exported from. Must be either 'QuantaSoft' or 'QX Manager'. Be careful to use correct spelling. |
| `targetChannel` | The channel nr used as target channel (default=1) |
| `controlChannel` | The channel nr used as control channel (default=2) |

## Value

A data.frame with columns for sample ID, target assay and control assay.

## Examples

```
## Path to example sample sheet included in PoDCall
path <- system.file("extdata", "Sample_names.csv", package="PoDCall")

## Select wells to get information for
well_id <- c("A04", "B04", "D04")

## Get information for selected wells
sampleSheet <- importSampleSheet(sampleSheet=path, well_id=well_id,
                                 software="QuantaSoft")
```

---

podcallChannelPlot *podcallChannelPlot*

---

## Description

Function that calls podcallScatterplot and podcallHistogram and draws a plot with both scatter plot and histogram.

## Usage

```
podcallChannelPlot(channelData, thr, channel, plotId = NULL)
```

## Arguments

| | |
|---|---|
| channelData | Amplitude values from one channel of a well. |
| thr | The threshold set for channel of a well. |
| channel | The channel the amplitude values belong to. Target channel is 1, control channel is 2. |
| plotId | A character string with title for the plot. |

## Value

A gtable with scatterplot and histogram

## Examples

```
## Get path to data
path <- system.file("extdata", "Amplitudes/", package="PoDCall")

## Read in data
data <- importAmplitudeData(path, skipLines=0)
data("thrTable")

## Get name of first list element and use as well ID
well_id <- names(data)[1]
```

```
## Set channel to plot
channel <- 1

## Get threshold for well_id and channel 1 (see ?thrTable)
thr <- thrTable[well_id, "thr_target"]

podcallChannelPlot(channelData=data[[well_id]][[channel]], thr, channel)
```

---

podcallDdpcr *Positive Droplet Calling for ddPCR*

---

## Description

Wrapper function that provide a complete workflow for the functionality of PoDCall. It takes path to amplitude files and sample sheet (optional), and parameters for setting threshold as input. Calls functions that read in data from files, sets threshold for each channel per well, calculates concentrations and optionally makes scatter plot and histogram for each channel per well. Results are returned as a table, optionally written to file. Plots will be written to file in a results directory if argument plots is set to TRUE.

## Usage

```
podcallDdpcr(dataDirectory,
                    sampleSheetFile=NULL,
                    B=200,
                    Q=9,
                    refwell=1,
                    targetChannel=c(1,2,3,4,5,6)[1],
                    controlChannel=c(1,2,3,4,5,6)[2],
                    dropletVolume=c(0.000851, 0.000795)[1],
                    nrChannels=c(1,2)[2],
                    software=c("QuantaSoft", "QX Manager")[2],
                    resultsToFile=FALSE,
                    plots=FALSE,
                    resPath=NULL)
```

## Arguments

| | |
|---|---|
| dataDirectory | Path to directory containing QuantaSoft amplitude files from one 96 well plate. Since well coordinates are used as identifiers, files in this directory should all be from the same 96 well plate. Furthermore, there can be no other files than the amplitude files from a well plate in the directory. |
| sampleSheetFile | File (optional) containing sample information from ddPCR experiment. This file must be a comma separated file containing the following columns: Well, Sample, TargetType and Target. |

| | |
|---|---|
| B | The number of permutations used for the Likelihood Ratio Test (default=200) |
| Q | A parameter for calling outliers (default=9) |
| refwell | reference well to calculate the shift in baseline (default=1) |
| targetChannel | The channel nr used as target channel (default=1) |
| controlChannel | The channel nr used as control channel (default=2) |
| dropletVolume | The average droplet size in µL used to calculate concentration (default=0.000851) |
| nrChannels | If single channel target and no control channel, set to 1, if control channel is used, set to 2 (default=2) |
| software | The software data was exported from, either QuntaSoft or QXmanager. Needs to be specified to ensure correct reading of data and sample sheet due to difference in formatting. (defult="QX Manager") |
| resultsToFile | Should results be written to file(.csv)? (default=FALSE) |
| plots | Should plots be created and written to file? (default=FALSE) |
| resPath | Optional argument to provide results directory path (default=NULL) |

## Value

The function returns a table (data frame) with thresholds, droplet counts, concentration and normalized concentration. The table is optionally written to a .csv-file and plots for both channels per well can be written to files.

## Examples

```
## Paths to data and sample sheet
dataPath <- system.file("extdata", "Amplitudes/", package="PoDCall")
ssPath <- system.file("extdata", "Sample_names.csv", package="PoDCall")

## Run PodCall
podcallResults <- podcallDdpcr(dataDirectory=dataPath,
                               sampleSheetFile=ssPath,
                               B=100, software="QuantaSoft")
```

---

# podcallHistogram *podcallHistogram*

---

## Description

Function that make a histogram of amplitude values from one channel of a well with threshold indicated by a vertical line.

## Usage

```
podcallHistogram(channelData, thr, channel, plotId = NULL)
```

### Arguments

| | |
|---|---|
| channelData | Amplitude values from one channel of a well. |
| thr | The threshold set for channel of a well. |
| channel | The channel the amplitude values belong to. Target channel is 1, control channel is 2. |
| plotId | A character string with title for the plot. |

### Value

A histogram of amplitude values from a channel from a well with a line indicating the set threshold.

### Examples

```
# Get path to data
path <- system.file("extdata", "Amplitudes/", package="PoDCall")

# Read in data
data <- importAmplitudeData(path, skipLines=0)
data("thrTable")

# Get name of first list element and use as well ID
well_id <- names(data)[1]

# Set channel to plot
channel <- 1

# Get threshold for well_id and channel 1 (see ?thrTable)
thr <- thrTable[well_id, "thr_target"]

histogram <- podcallHistogram(channelData=data[[well_id]][[channel]],
                              thr,
                              channel)
```

---

## podcallMultiplot *podcallMultiplot*

---

### Description

A function that returns faceted scatterplots for multiple wells suitable for comparison of wells.

### Usage

```
podcallMultiplot(
  plateData,
  thresholds,
  channel = c("target", "control"),
  sampleID = NULL,
```

```
  colCh = c(1, 2, 3, 4, 5, 6)
)
```

## Arguments

| | |
|---|---|
| plateData | A list containing data frames with amplitude values from selected wells that is to be compared. One data frame per well. |
| thresholds | A data.frame containing the thresholds for the selected wells |
| channel | What channel to plot, 'target' channel or 'ontrol' channel. |
| sampleID | The sample IDs for the selected wells. |
| colCh | The channel from the instrument to plot. Controls color of plot. |

## Value

Faceted scatterplot with line indicating threshold. One facet per selected well.

## Examples

```
## Set path to data
path <- system.file("extdata", "Amplitudes/", package="PoDCall")
ssPath <- system.file("extdata", "Sample_names.csv", package="PoDCall")

## Read in data files
data <- importAmplitudeData(path, skipLines=0)
sampleSheet <- importSampleSheet(sampleSheet=ssPath, well_id=names(data),
software="QuantaSoft")
data("thrTable")

## Create plot using threshold from thrTable, see ?thrTable
plot <- podcallMultiplot(plateData=data,
                         thresholds=thrTable[names(data), ],
                         sampleID=sampleSheet$sample_id,
                         channel="target")
```

---

# podcallScatterplot *podcallScatterplot*

---

## Description

Function that make a scatterplot of amplitude values from one channel of a well with threshold indicated by a horizontal line.

## Usage

```
podcallScatterplot(channelData, thr, channel, plotId = NULL)
```

### Arguments

| | |
|---|---|
| channelData | Amplitude values from one channel of a well. |
| thr | The threshold set for channel of a well. |
| channel | The channel the amplitude values belong to. Used to control plot color. |
| plotId | A character string with title for the plot. |

### Value

A scatterplot of all droplets from a channel from a well with a line indicating the set threshold.

### Examples

```
# Get path to data
path <- system.file("extdata", "Amplitudes/", package="PoDCall")

# Read in data
data <- importAmplitudeData(path, skipLines=0)
data("thrTable")

# Get name of first list element and use as well ID
well_id <- names(data)[1]

# Set channel to plot
channel <- 1

# Get threshold for well_id and channel 1 (see ?thrTable)
thr <- thrTable[well_id, "thr_target"]

scatterplot <- podcallScatterplot(channelData=data[[well_id]][[channel]],
                                  thr,
                                  channel)
```

---

## podcallShiny *PoDCall shiny launcher*

---

### Description

This function launches the PoDCall shiny app in a web browser

### Usage

```
podcallShiny()
```

### Value

Does not return anything, but launches PoDCall shiny app

## Examples

```
## Not run:
podcallShiny()

## End(Not run)
```

---

podcallThresholds *podcallThresholds*

---

## Description

Function sets threshold per channel per well and calculates concentrations. Results are returned as a data frame.

## Usage

```
podcallThresholds(plateData,
                          nrChannels=c(1,2)[2],
                          B=200,
                          Q=9,
                          refWell=1,
                          targetChannel=c(1,2,3,4,5,6)[1],
                          controlChannel=c(1,2,3,4,5,6)[2],
                          dropletVolume=c(0.000851, 0.000795)[1],
                          updateProgress=NULL)
```

## Arguments

| | |
|---|---|
| plateData | List of data frames with amplitude data from a 96 well plate |
| nrChannels | If single channel target and no control channel, set to 1, if control channel is used, set to 2 (default=2) |
| B | Number of permutations for the Likelihood Ratio Test (LRT) (default=200) |
| Q | Parameter for outlier calling (default=9) |
| refWell | Reference well to calculate the shift in baseline (default=1) |
| targetChannel | The channel nr used as target channel (default=1) |
| controlChannel | The channel nr used as control channel (default=2) |
| dropletVolume | The average droplet size in µL used to calculate concentration (default=0.000851) |
| updateProgress | function to update progress bar in shiny app (default=NULL) |

## Value

A table with results and metrics, one row per well.

## Examples

```
## Path to example data
dataPath <- system.file("extdata", "Amplitudes/", package="PoDCall")

## Read in example data
dataList <- importAmplitudeData(dataDirectory=dataPath, skipLines=0)

## Set thresholds
thresholds <- podcallThresholds(plateData=dataList,
                                B=100)
```

---

`thrTable` *PoDCall Example Threshold Table*

---

## Description

A `data.frame` that contains the results of running PodCall with the amplitude data files included in the package. For testing and running of examples. See vignette for more detailed description about columns.

## Usage

```
data("thrTable")
```

## Format

A `data.frame` with 13 columns, which are:

**sample_id** Sample ID

**target_ch** Target channel

**thr_target** Threshold target channel (target assay)

**thr_ctrl** Threshold control channel (control assay)

**pos_dr_target** Positive droplets target

**pos_dr_ctrl** Positive droplets control

**tot_droplets** Total droplets

**c_target** Concentration target

**c_ctrl** Concentration control

**c_norm_4Plex** Normalized concentration based on 4Plex control

**c_norm_sg** Normalized concentration based on single gene control

**q** Parameter Q for calling outliers

**target_assay** Target assay

**ctrl_assay** Control assay

**ref_well** Reference well used to set threshold

**Source**

In-house cell-line experiment.

# Index

∗ **datasets**
thrTable, 11

importAmplitudeData, 2
importSampleSheet, 3

podcallChannelPlot, 4
podcallDdpcr, 5
podcallHistogram, 6
podcallMultiplot, 7
podcallScatterplot, 8
podcallShiny, 9
podcallThresholds, 10

thrTable, 11