

Package ‘IONiseR’

September 25, 2024

Title Quality Assessment Tools for Oxford Nanopore MinION data

Version 2.29.0

Description IONiseR provides tools for the quality assessment of Oxford Nanopore MinION data. It extracts summary statistics from a set of fast5 files and can be used either before or after base calling. In addition to standard summaries of the read-types produced, it provides a number of plots for visualising metrics relative to experiment run time or spatially over the surface of a flowcell.

License MIT + file LICENSE

Depends R (>= 3.4)

Imports rhdf5, dplyr, magrittr, tidyr, ShortRead, Biostrings, ggplot2, methods, BiocGenerics, XVector, tibble, stats, BiocParallel, bit64, stringr, utils

VignetteBuilder knitr

Suggests BiocStyle, knitr, rmarkdown, gridExtra, testthat, minionSummaryData

biocViews QualityControl, DataImport, Sequencing

NeedsCompilation no

Author Mike Smith [aut, cre]

Maintainer Mike Smith <grimbough@gmail.com>

RoxygenNote 6.0.1

Collate 'IONiseR.R' 'classes.R' 'Methods-accessors.R' 'Methods-subsetting.R' 'fast5Readers.R' 'fast5Status.R' 'fast5readers_summary.R' 'fast5utilities.R' 'fastqProcessing.R' 'plotting_kmers.R' 'plotting_layout.R' 'plotting_summaryStats.R' 'processing_bam.R' 'readSummary.R' 'squiggle.R'

git_url <https://git.bioconductor.org/packages/IONiseR>

git_branch devel

git_last_commit 356c193

git_last_commit_date 2024-04-30

Repository Bioconductor 3.20

Date/Publication 2024-09-24

Contents

| | |
|--|-----------|
| baseCalled | 2 |
| channelActivityPlot | 3 |
| channelHeatmap | 4 |
| eventData | 4 |
| Fast5Summary-class | 5 |
| fast5toFastq | 7 |
| fastq | 8 |
| fastq2D | 8 |
| fastqComplement | 9 |
| fastqTemplate | 9 |
| IONiseR | 10 |
| layoutPlot | 10 |
| muxHeatmap | 11 |
| plotActiveChannels | 11 |
| plotBaseProductionRate | 12 |
| plotCurrentByTime | 12 |
| plotEventRate | 13 |
| plotKmerFrequencyCorrelation | 13 |
| plotReadAccumulation | 14 |
| plotReadCategoryCounts | 15 |
| plotReadCategoryQuals | 15 |
| plotReadTypeProduction | 16 |
| readFast5Log | 16 |
| readFast5Summary | 17 |
| readInfo | 18 |
| Index | 19 |

| | |
|------------|--------------------------------|
| baseCalled | <i>Extract baseCalled slot</i> |
|------------|--------------------------------|

Description

This generic function accesses the baseCalled slot stored in an object derived from the Fast5Summary class.

Usage

```
baseCalled(x)
```

Arguments

x Object of class `Fast5Summary`

Value

A data.frame with 6 columns

Examples

```
if( require(minionSummaryData) ) {  
  data(s.typhi.rep2, package = 'minionSummaryData')  
  baseCalled( s.typhi.rep2 )  
}
```

channelActivityPlot *Visualise a specified metric over all channels over time.*

Description

Plots a line for each fast 5 file, arranged by channel and experiment time when the signal was being recorded. The colour of each line can be specified by the user to reflect any metric they wish. The intention of the plot is to investigate trends that may appear at specific time points, or influence a subset of channels.

Usage

```
channelActivityPlot(summaryData, zScale = NULL, zAverage = TRUE)
```

Arguments

| | |
|-------------|---|
| summaryData | Object of class Fast5Summary . |
| zScale | A data.frame containing two columns. The first must be labelled 'id' and correspond to id field present in all slots in summaryData. The second column should contain data pertaining to that reads that you wish to be represented on the coloured z-axis. |
| zAverage | Logical indicating if a bar showing the mean across all channel for the chosen zScale should be shown on the plot. Defaults to TRUE. |

Value

Returns an object of class gg representing the plot.

Examples

```
if( require(minionSummaryData) ) {  
  require(dplyr)  
  data(s.typhi.rep3, package = 'minionSummaryData')  
  ## we will plot the median event signal for each read on z-axis  
  z_scale = select(eventData(s.typhi.rep3), id, median_signal)  
  channelActivityPlot( s.typhi.rep3, zScale = z_scale )  
}
```

| | |
|----------------|---------------------------------------|
| channelHeatmap | <i>Create layout plot of flowcell</i> |
|----------------|---------------------------------------|

Description

Creates a plot representing the layout of a MinION flow cell. Each circle represents an individual channel with the intensity reflecting a specified sequencing metric. This function is a more generalised version of [layoutPlot](#), allowing the user to map any value the like on the channel layout.

Usage

```
channelHeatmap(data, zValue)
```

Arguments

| | |
|--------|---|
| data | A data.frame. Should have at least two columns, one of which has the name 'channel'. |
| zValue | Character string specifying the name of the column to be used for the colour scaling. |

Value

Returns an object of gg representing the plot.

Examples

```
library(dplyr)
if( require(minionSummaryData) ) {
  data(s.typhi.rep2, package = 'minionSummaryData')
  ## calculate and plot the mean number of events recorded by each channel
  avgEvents <- left_join(readInfo(s.typhi.rep2), eventData(s.typhi.rep2), by = 'id') %>%
    group_by(channel) %>%
    summarise(mean_nevents = mean(num_events))
  channelHeatmap(avgEvents, zValue = 'mean_nevents')
}
```

| | |
|-----------|-------------------------------|
| eventData | <i>Extract eventData slot</i> |
|-----------|-------------------------------|

Description

This generic function accesses the eventData slot stored in an object derived from the Fast5Summary class.

Usage

```
eventData(x)
```

Arguments

| | |
|---|--|
| x | Object of class Fast5Summary |
|---|--|

Value

An object of class Fast5Summary

Methods (by generic)

- `length`: Returns the number of files read during creation of the object
- `readInfo`: Returns readInfo data.frame
- `eventData`: Returns eventData data.frame
- `baseCalled`: Returns baseCalled data.frame
- `fastq`: Returns ShortReadQ object stored in fastq slot.
- `[]`: Subset object and return an object of the same class.
- `fastqTemplate`: Returns ShortReadQ object containing only template reads
- `fastqComplement`: Returns ShortReadQ object containing only complement reads
- `fastq2D`: Returns ShortReadQ object containing only 2D reads

Slots

`readInfo` Object of class tibble. Contains five columns:

- `id` - an integer key that allows use to match entries in the separate slots of this object.
- `file` - Basename of the fast5 file the data was read from.
- `read` - Read number from channel.
- `channel` - channel.
- `mux` - Specific pore that was used within the four that are assigned to a single channel. Should be in the range 1-4, but if this isn't available it will be 0.

`rawData` Object of class tibble. Intended to hold raw signal data although reading this is currently not implemented in IONiseR.

`eventData` Object of class tibble. Holds summary of events data prior to base calling. Contains five columns:

- `id` - an integer key that allows use to match entries in the separate slots of this object.
- `start_time` - time in seconds after the run started that this reading began.
- `duration` - time in seconds the reading lasted.
- `num_events` - the number of events that were recorded as part of this reading.
- `median_signal` - median of the recorded signals for this set of events.

`baseCalled` Object of class tibble. For the most part contains similar data to the `@eventData` slot, the base called data is derived from it.

- `id` - an integer key that allows use to match entries in the separate slots of this object.
- `start_time` - time in seconds after the run started that this reading began.
- `duration` - time in seconds the reading lasted.
- `num_events` - the number of events that were recorded as part of this reading.
- `strand` - can be either 'template' or 'complement'
- `full_2D` - boolean value specifying whether the read forms part of a 2D pair. If TRUE the FASTQ data for the template, complement and 2D read will be available in the `@fastq` slot.

`fastq` Object of class ShortReadQ. This slot contains all reads (template, complement and 2D). The read names take the form NUM_STRAND, where NUM matches with the id column in the other slots and STRAND indicates whether the read is template, complement or 2D.

`versions` A list intended to store the version of IONiseR that was used to create the object. (May be extended in the future to include the version of MinKNOW that the original fast5 files were processed, if this can be determined accurately.)

Examples

```

if( require(minionSummaryData) ) {
  data(s.typhi.rep2, package = 'minionSummaryData')
  length( s.typhi.rep2 )
}

```

fast5toFastq

*Extract FASTQ files from fast5 files***Description**

This function provides direct access to the FASTQ entries held within fast5 files. If you are only interested in getting hold of the base called reads, and don't require any raw-signal or event information, use this function. Given a vector of fast5 files, the FASTQ entries will be combined and up to three gzip compressed FASTQ will be created - one for each of the template, complement and 2D strands depending upon what is available in the input files.

Usage

```

fast5toFastq(files, strand = "all", fileName = NULL, outputDir = NULL,
             ncores = 1)

```

Arguments

| | |
|-----------|---|
| files | Character vector of fast5 files to be read. |
| strand | Character vector specifying the strand to extract. Can take any combination of the following options: "template", "complement", "2D", "all", "both". |
| fileName | Stem for the name of the names of the output file names. The appropriate strand will be appended to each file e.g. fileName_complement.fq.gz or fileName_template.fq.gz |
| outputDir | Directory output files should be written to. |
| ncores | Specify the number of CPU cores that should be used to process the files. Currently this seems to be more IO bound than CPU, so there is little benefit achieved by using a high number of cores. |

Value

No value returned. Run for the side effect of writing the FASTQ files to disk.

Examples

```

## Not run:
fast5files <- list.files('/foo/bar/', pattern = '.fast5$')
summaryData <- readFast5Summary(fast5files)

## End(Not run)

```

fastq *Extract fastq slot*

Description

This generic function accesses the fastq slot stored in an object derived from the Fast5Summary class.

Usage

```
fastq(x)
```

Arguments

x Object of class [Fast5Summary](#)

Value

A ShortReadQ object

Examples

```
if( require(minionSummaryData) ) {  
  data(s.typhi.rep2, package = 'minionSummaryData')  
  fastq( s.typhi.rep2 )  
}
```

fastq2D *Extract 2D reads*

Description

This generic function accesses the fastq slot stored in an object derived from the Fast5Summary class, and returns only the 2D reads.

Usage

```
fastq2D(x)
```

Arguments

x Object of class [Fast5Summary](#)

Value

A ShortReadQ object

Examples

```
if( require(minionSummaryData) ) {  
  data(s.typhi.rep2, package = 'minionSummaryData')  
  fastq2D( s.typhi.rep2 )  
}
```

| | |
|-----------------|---------------------------------|
| fastqComplement | <i>Extract complement reads</i> |
|-----------------|---------------------------------|

Description

This generic function accesses the fastq slot stored in an object derived from the Fast5Summary class, and returns only the complement reads.

Usage

```
fastqComplement(x)
```

Arguments

x Object of class [Fast5Summary](#)

Value

A ShortReadQ object

Examples

```
if( require(minionSummaryData) ) {  
  data(s.typhi.rep2, package = 'minionSummaryData')  
  fastqComplement( s.typhi.rep2 )  
}
```

| | |
|---------------|-------------------------------|
| fastqTemplate | <i>Extract template reads</i> |
|---------------|-------------------------------|

Description

This generic function accesses the fastq slot stored in an object derived from the Fast5Summary class, and returns only the template reads.

Usage

```
fastqTemplate(x)
```

Arguments

x Object of class [Fast5Summary](#)

Value

A ShortReadQ object

Examples

```
if( require(minionSummaryData) ) {  
  data(s.typhi.rep2, package = 'minionSummaryData')  
  fastqTemplate( s.typhi.rep2 )  
}
```

IONiseR

IONiseR: A package for assessing quality of MinION data

Description

IONiseR provides tools for the quality assessment of Oxford Nanopore MinION data. It extracts summary statistics from a set of fast5 files and can be used either before or after base calling. In addition to standard summaries of the read-types produced, it provides a number of plots for visualising metrics relative to experiment run time or spatially over the surface of a flowcell.

layoutPlot

Create layout plot of flowcell

Description

Creates a plot representing the layout of a MinION flow cell. Each circle represents an individual channel with the intensity reflecting the total kilobases of sequence produced. This only considers reads marked as template or complement, 2D reads are ignored as they are generated from the former two.

Usage

```
layoutPlot(summaryData, attribute = NULL)
```

Arguments

| | |
|-------------|--|
| summaryData | Object of class Fast5Summary . |
| attribute | Character string indicating what to plot. Currently accepted values are: "nreads", "kb", "signal". |

Value

Returns an object of gg representing the plot.

Examples

```
if( require(minionSummaryData) ) {  
  data(s.typhi.rep2, package = 'minionSummaryData')  
  layoutPlot( s.typhi.rep2, attribute = 'nreads' )  
  layoutPlot( s.typhi.rep2, attribute = 'kb' )  
}
```

| | |
|------------|---------------------------------------|
| muxHeatmap | <i>Create layout plot of flowcell</i> |
|------------|---------------------------------------|

Description

Creates a plot representing the layout of a MinION flow cell. Each circle represents an individual channel with the intensity reflecting a specified sequencing metric. This function is a more generalised version of [layoutPlot](#), allowing the user to map any value the like on the channel layout.

Usage

```
muxHeatmap(data, zValue)
```

Arguments

| | |
|--------|---|
| data | A data.frame. Should have at least two columns, one of which has the name 'channel'. |
| zValue | Character string specifying the name of the column to be used for the colour scaling. |

Value

Returns an object of gg representing the plot.

| | |
|--------------------|---|
| plotActiveChannels | <i>Plot the number of active channels for each minute of run time</i> |
|--------------------|---|

Description

Plot the number of active channels for each minute of run time

Usage

```
plotActiveChannels(summaryData)
```

Arguments

| | |
|-------------|--|
| summaryData | Object of class Fast5Summary . |
|-------------|--|

Value

Returns an object of class gg representing the plot.

Examples

```
if( require(minionSummaryData) ) {
  data(s.typhi.rep2, package = 'minionSummaryData')
  plotActiveChannels( s.typhi.rep2 )
}
```

`plotBaseProductionRate`*Plot the mean rate at which bases are recorded*

Description

For each read, the ratio between the total number of bases called in the read (template and complement strand, but not 2D composite) and the time spent in the pore is calculated. This is then plotted against the time the read entered the pore, allow us to assess whether the rate at which callable bases are read changes during the experiment run time.

Usage

```
plotBaseProductionRate(summaryData)
```

Arguments

`summaryData` Object of class [Fast5Summary](#).

Details

This is likely very similar to [plotEventRate](#), although one may find that large number of events occur that can not be base called, resulting in a difference between these two plots.

Value

Returns an object of class `gg` representing the plot.

Examples

```
if( require(minionSummaryData) ) {  
  data(s.typhi.rep2, package = 'minionSummaryData')  
  plotBaseProductionRate( s.typhi.rep2 )  
}
```

`plotCurrentByTime`*View changes in signal against run time.*

Description

Plots the median recorded current for each fast5 file against the time at which the recording began.

Usage

```
plotCurrentByTime(summaryData)
```

Arguments

`summaryData` Object of class [Fast5Summary](#).

Value

Returns an object of class gg representing the plot.

Examples

```
if( require(minionSummaryData) ) {  
  data(s.typhi.rep2, package = 'minionSummaryData')  
  plotCurrentByTime( s.typhi.rep2 )  
}
```

plotEventRate

Plot the mean rate at which events occur

Description

For each read, the ratio between the number of events comprising the read and the time spent in the pore is calculated. This is then plotted against the time the read entered the pore, allow us to assess whether the rate at which events occur changes during the experiment run time.

Usage

```
plotEventRate(summaryData)
```

Arguments

summaryData Object of class [Fast5Summary](#).

Value

Returns an object of class gg representing the plot.

Examples

```
if( require(minionSummaryData) ) {  
  data(s.typhi.rep2, package = 'minionSummaryData')  
  plotEventRate( s.typhi.rep2 )  
}
```

plotKmerFrequencyCorrelation

Display correlation between pentamer proportions in two time windows

Description

Plots

Usage

```
plotKmerFrequencyCorrelation(summaryData, kmerLength = 5,  
  groupedMinutes = 10, only2D = TRUE)
```

Arguments

| | |
|----------------|---|
| summaryData | Object of class Fast5Summary . |
| kmerLength | Specifies the length of kmers to compare. Defaults to 5 given the current pentamer reading nature of the nanopores. |
| groupedMinutes | Defines how many minutes each grouping of reads spans. |
| only2D | Logical. If TRUE kmers are computed for only full 2D reads. If FALSE 2D reads are ignored and all available template and complement strands are used. |

Value

Returns an object of class gg representing the plot.

Examples

```
if( require(minionSummaryData) ) {  
  data(s.typhi.rep3, package = 'minionSummaryData')  
  plotKmerFrequencyCorrelation( s.typhi.rep3, only2D = FALSE )  
}
```

plotReadAccumulation *Plot the accumulation of reads over the duration of the experiment.*

Description

Plot the accumulation of reads over the duration of the experiment.

Usage

```
plotReadAccumulation(summaryData)
```

Arguments

| | |
|-------------|--|
| summaryData | Object of class Fast5Summary . |
|-------------|--|

Value

Returns an object of class gg representing the plot.

Examples

```
if( require(minionSummaryData) ) {  
  data(s.typhi.rep2, package = 'minionSummaryData')  
  plotReadAccumulation( s.typhi.rep2 )  
}
```

`plotReadCategoryCounts`

Plot the proportion of template, complement and 2D reads found a dataset.

Description

Generates a bar plot showing the breakdown of read types found in a set of fast5 files. There is a strict hierarchy to the types of read that can be found in a fast5 file. A full 2D read requires both a complement and template strand to have been read correctly. Similarly, a complement strand can only be present if the template was read successfully. Finally, you can encounter a file containing now called bases on either strand. Here we visualise the total number of fast5 files, along with the counts containing each of the categories above. For an ideal dataset all four bars will be the same height. This is unlikely, but the drop between bars can give some indication of data quality.

Usage

```
plotReadCategoryCounts(summaryData)
```

Arguments

`summaryData` Object of class [Fast5Summary](#).

Value

Returns an object of class `gg` representing the plot.

Examples

```
if( require(minionSummaryData) ) {  
  data(s.typhi.rep2, package = 'minionSummaryData')  
  plotReadCategoryCounts( s.typhi.rep2 )  
}
```

`plotReadCategoryQuals` *Visualise the mean base quality of each read*

Description

Generates a box plot showing the mean base quality for each read, broken down into the three categories of read type that can be found in a fast5 file.

Usage

```
plotReadCategoryQuals(summaryData)
```

Arguments

`summaryData` Object of class [Fast5Summary](#).

Value

Returns an object of class gg representing the plot.

Examples

```
if( require(minionSummaryData) ) {
  data(s.typhi.rep2, package = 'minionSummaryData')
  plotReadCategoryQuals( s.typhi.rep2 )
}
```

plotReadTypeProduction

View changes in signal against run time.

Description

Plots the median recorded current for each fast5 file against the time at which the recording began.

Usage

```
plotReadTypeProduction(summaryData, groupedMinutes = 10)
```

Arguments

summaryData Object of class [Fast5Summary](#).

groupedMinutes Integer specifying how many minutes of runtime should be grouped together.

Value

Returns an object of class gg representing the plot.

Examples

```
if( require(minionSummaryData) ) {
  data(s.typhi.rep2, package = 'minionSummaryData')
  plotReadTypeProduction( s.typhi.rep2 )
}
```

readFast5Log

Read the log entry from a single fast5 file

Description

Basecalling procedures performed on fast5 files generally leave a log file entry recording how far through the pipeline the file proceeded. This function will extract this information as a single string. It can be printed in a more readable format using the [cat](#) function.

Usage

```
readFast5Log(file)
```


Arguments

file Character vector of fast5 file to be read.

Value

Character vector containing the log information. NULL if no log is found.

Examples

```
fast5file <- system.file('extdata', 'example.fast5', package = "IONiseR")
log <- readFast5Log(fast5file)
cat(log)
```

| | |
|------------------|--|
| readFast5Summary | <i>Read summary data from fast5 files.</i> |
|------------------|--|

Description

Reads one or more fast5 files and collects summary information about them.

Usage

```
readFast5Summary(files)
```

Arguments

files Character vector of fast5 files to be read.

Details

Currently this function assumes all files passed to it come from the same sequencing run. It makes no effort to check for alternative file names or the like. If files from multiple runs are passed to it they will be collated together and any analysis performed on them will represent the mixture of both experiments.

Value

Object of class [Fast5Summary](#)

Examples

```
## Not run:
fast5files <- list.files('/foo/bar/', pattern = '.fast5$')
summaryData <- readFast5Summary(fast5files)

## End(Not run)
```

| | |
|----------|------------------------------|
| readInfo | <i>Extract readInfo slot</i> |
|----------|------------------------------|

Description

This generic function accesses the readInfo slot stored in an object derived from the Fast5Summary class.

Usage

```
readInfo(x)
```

Arguments

x Object of class [Fast5Summary](#)

Value

A data.frame with 5 columns

Examples

```
if( require(minionSummaryData) ) {  
  data(s.typhi.rep2, package = 'minionSummaryData')  
  readInfo( s.typhi.rep2 )  
}
```

Index

[,Fast5Summary,ANY,ANY,ANY-method
(Fast5Summary-class), 5

baseCalled, 2
baseCalled,Fast5Summary-method
(Fast5Summary-class), 5

cat, 16
channelActivityPlot, 3
channelHeatmap, 4

eventData, 4
eventData,Fast5Summary-method
(Fast5Summary-class), 5

Fast5Summary, 2–4, 8–18
Fast5Summary-class, 5
fast5toFastq, 7
fastq, 8
fastq,Fast5Summary-method
(Fast5Summary-class), 5
fastq2D, 8
fastq2D,Fast5Summary-method
(Fast5Summary-class), 5
fastqComplement, 9
fastqComplement,Fast5Summary-method
(Fast5Summary-class), 5
fastqTemplate, 9
fastqTemplate,Fast5Summary-method
(Fast5Summary-class), 5

IONiseR, 10
IONiseR-package (IONiseR), 10

layoutPlot, 4, 10, 11
length,Fast5Summary-method
(Fast5Summary-class), 5

muxHeatmap, 11

plotActiveChannels, 11
plotBaseProductionRate, 12
plotCurrentByTime, 12
plotEventRate, 12, 13
plotKmerFrequencyCorrelation, 13
plotReadAccumulation, 14
plotReadCategoryCounts, 15
plotReadCategoryQuals, 15
plotReadTypeProduction, 16

readFast5Log, 16
readFast5Summary, 17
readInfo, 18
readInfo,Fast5Summary-method
(Fast5Summary-class), 5