Package ‘IONiseR’

May 28, 2024

Title Quality Assessment Tools for Oxford Nanopore MinION data
Version 2.28.0
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 RELEASE_3_19
git_last_commit d4e3dbc
Contents

baseCalled ................................................. 2
channelActivityPlot ........................................ 3
channelHeatmap ............................................. 4
eventData .................................................. 5
Fast5Summary-class ............................................ 5
fast5toFastq ................................................ 7
fastq ....................................................... 8
fastq2D ..................................................... 9
fastqComplement ......................................... 10
fastqTemplate ............................................. 10
IONiseR .................................................. 11
layoutPlot ............................................... 11
muxHeatmap .............................................. 12
plotActiveChannels ..................................... 12
plotBaseProductionRate ................................ 13
plotCurrentByTime ....................................... 14
plotEventRate ........................................... 14
plotKmerFrequencyCorrelation ..................... 15
plotReadAccumulation .................................. 16
plotReadCategoryCounts ................................. 16
plotReadCategoryQuals .................................. 17
plotReadTypeProduction ................................. 18
readFast5Log .......................................... 18
readFast5Summary ..................................... 19
readInfo ............................................... 20

Index ....................................................... 21

baseCalled  Extract baseCalled slot

Description

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

Usage

baseCalled(x)
channelActivityPlot

Arguments

- `summaryData`: Object of class `Fast5Summary`

Value

A data.frame with 6 columns

Examples

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

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

```r
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.
### channelHeatmap

*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 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**

```r
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**

```r
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**  

**Extract eventData slot**

**Description**

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

**Usage**

```r
eventData(x)
```

**Arguments**

- `x` Object of class `Fast5Summary`

**Value**

A data.frame with 5 columns

**Examples**

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

---

**Fast5Summary-class**  

*An S4 class for summarised data from a MinION sequencing run*

---

**Description**

An S4 class for summarised data from a MinION sequencing run

**Usage**

```r
## S4 method for signature 'Fast5Summary'
length(x)
```

```r
## S4 method for signature 'Fast5Summary'
readInfo(x)
```

```r
## S4 method for signature 'Fast5Summary'
eventData(x)
```

```r
## S4 method for signature 'Fast5Summary'
```
baseCalled(x)

## S4 method for signature 'Fast5Summary'
fastq(x)

## S4 method for signature 'Fast5Summary,ANY,ANY,ANY'
x[i]

## S4 method for signature 'Fast5Summary'
fastqTemplate(x)

## S4 method for signature 'Fast5Summary'
fastqComplement(x)

## S4 method for signature 'Fast5Summary'
fastq2D(x)

### Arguments

x Object of class Fast5Summary

i Vector defining index to subset by.

### 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

```r
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

```r
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

```r
## 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

```r
fastq(x)
```

Arguments

- **x**: Object of class `Fast5Summary`
fastq2D

Value

A ShortReadQ object

Examples

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

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

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

Extract complement reads

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  

Extract template reads

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
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

```r
if( require(minionSummaryData) ) {
  data(s.typhi.rep2, package = 'minionSummaryData')
  fastqTemplate( s.typhi.rep2 )
  layoutPlot( s.typhi.rep2, attribute = 'nreads' )
  layoutPlot( s.typhi.rep2, attribute = 'kb' )
}
```
**muxHeatmap**  
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 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**  
Plot the number of active channels for each minute of run time

**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.
plotBaseProductionRate

Examples

```{r}
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, allowing us to assess whether the rate at which callable bases are read changes during the experiment run time.

Usage

```r
plotBaseProductionRate(summaryData)
```

Arguments

- `summaryData` Object of class `Fast5Summary`.

Details

This is likely very similar to `plotEventRate`, although one may find that a large number of events occur that cannot be base called, resulting in a difference between these two plots.

Value

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

Examples

```{r}
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**

```r
plotCurrentByTime(summaryData)
```

**Arguments**

- `summaryData` Object of class `Fast5Summary`.

**Value**

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

**Examples**

```r
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, allowing us to assess whether the rate at which events occur changes during the experiment run time.

**Usage**

```r
plotEventRate(summaryData)
```

**Arguments**

- `summaryData` Object of class `Fast5Summary`.

**Value**

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

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

```
plotKmerFrequencyCorrelation( s.typhi.rep3, only2D = FALSE )
```

Description

Plots

Usage

```r
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

```r
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**

```r
plotReadAccumulation(summaryData)
```

**Arguments**

- `summaryData` Object of class `Fast5Summary`.

**Value**

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

**Examples**

```r
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**

```r
plotReadCategoryCounts(summaryData)
```
plotReadCategoryQuals

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**

```r
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**

```r
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**

```r
readFast5Log(file)
```

**Arguments**

- `file`: Character vector of fast5 file to be read.
**readFast5Summary**

Value

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

Examples

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

---

**readFast5Summary** Read summary data from fast5 files.

Description

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

Usage

```r
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

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

## End(Not run)
```
**readInfo**

Extract readInfo slot

**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**

```r
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, 18
channelActivityPlot, 3
channelHeatmap, 4
eventData, 5
eventData,Fast5Summary-method (Fast5Summary-class), 5

Fast5Summary, 3, 5, 8–20
Fast5Summary-class, 5
fast5toFastq, 7
fastq, 8
fastq,Fast5Summary-method (Fast5Summary-class), 5
fastq2D, 9
fastq2D,Fast5Summary-method (Fast5Summary-class), 5
fastqComplement, 10
fastqComplement,Fast5Summary-method (Fast5Summary-class), 5
fastqTemplate, 10
fastqTemplate,Fast5Summary-method (Fast5Summary-class), 5

IONiseR, 11
IONiseR-package (IONiseR), 11

layoutPlot, 4, 11, 12
length,Fast5Summary-method (Fast5Summary-class), 5

muxHeatmap, 12

plotActiveChannels, 12

plotBaseProductionRate, 13
plotCurrentByTime, 14
plotEventRate, 13, 14
plotKmerFrequencyCorrelation, 15
plotReadAccumulation, 16
plotReadCategoryCounts, 16
plotReadCategoryQuals, 17
plotReadTypeProduction, 18

readFast5Log, 18
readFast5Summary, 19
readInfo, 20
readInfo,Fast5Summary-method (Fast5Summary-class), 5