

Package ‘NanoR’

October 25, 2019

Type Package

Title Analyze Nanopore MinION and GridION X5 1D Sequencing Data

Version 3.0

Author Davide Bolognini, BS, PhD Fellow [aut, cre]

Maintainer Davide Bolognini <davidebolognini7@gmail.com>

Depends R (>= 3.1.3)

Imports parallel,
ggplot2 (>= 2.2.1),
reshape2,
scales,
RColorBrewer,
grid,
gridExtra,
rhdf5(>= 2.14),
ShortRead (>= 1.24.0)

Encoding UTF-8

LazyData true

RoxygenNote 6.1.1

R topics documented:

NanoCompare	2
NanoPrepareG	2
NanoPrepareM	3
NanoStatsG	4
NanoStatsM	4
NanoTableG	5
NanoTableM	6
Index	7

NanoCompare

NanoComapre

Description

Compare ONT experiments

Usage

NanoCompare(DataIn, DataOut, Labels)

Arguments

DataIn	Object of class character with paths to DataForComparison for each experiment
DataOut	Output folder
Labels	Object of class character with (ordered) labels for each experiment

Value

Plot:
- Violins.pdf;

Examples

```
#do not run
DataIn<-c("/path/to/folder1/DataForComparison","/path/to/folder2/DataForComparison",...)
Labels<-c("first","second")
DataOut <- "/path/to/output"
NanoCompare(DataIn=DataIn,DataOut=DataOut,Labels=Labels)
```

NanoPrepareG

NanoPrepareG

Description

Organize MinION/GridION sequencing summary and FASTQ

Usage

NanoPrepareG(DataSummary, DataFastq)

Arguments

DataSummary	Path to sequencing summary
DataFastq	Path to passed FASTQ

Details

FASTQ are found recursively

Value

Object of class list

Examples

```
#do not run
DataSummary<-'path/to/sequencing_summary'
DataFastq<-'path/to/fastq_pass'
List<-NanoPrepareG(DataSummary,DataFastq)
```

NanoPrepareM	<i>NanoPrepareM</i>
--------------	---------------------

Description

Organize MinION/GridION basecalled FAST5

Usage

```
NanoPrepareM(DataPass, DataFail = NA, DataSkip = NA,
  MultiRead = FALSE)
```

Arguments

DataPass	Path to passed FAST5
DataFail	Path to failed FAST5 [NA]
DataSkip	Path to skipped FAST5 [NA]
MultiRead	Logical. If TRUE, assume multi-read FAST5 [FALSE]

Details

FAST5 are found recursively

Value

Object of class list

Examples

```
#do not run
DataPass<-"path/to/fast5_pass"
DataFail<-"path/to/fast5_fail" #can be omitted
#single-read .fast5 files
List<-NanoPrepareM(DataPass,DataFail,Label=Label)
#multi-read .fast5 files
List<-NanoPrepareM(DataPass,DataFail, Label=Label,MultiRead=TRUE)
```

NanoStatsG

NanoStatsG

Description

Plot statistics

Usage

```
NanoStatsG(NanoGList, NanoGTable, DataOut, KeepGGObj = FALSE)
```

Arguments

NanoGList	Object of class list from NanoPrepareG
NanoGTable	Object of class matrix from NanoTableG
DataOut	Output folder (the same used for NanoTableG)
KeepGGObj	Logical. If TRUE, store ggplot2 data frames behind generated plots [FALSE]

Value

Plots:

- Yield.pdf (accumulation of reads and bps);
- RBLQ.pdf (# reads, # bps, length and quality overview every 30 minutes of experiment);
- LvQ.pdf (length and quality compared jointly);
- PFGC.pdf (passed and failed reads, GC content if previously computed);
- Activity.pdf (channels and muxes activity (# bps). Inactive channels and muxes are grey-colored)

Tables:

- summary.txt (table with major statistics for the sequencing run)

Examples

```
#do not run
#assume List is the output from NanoPrepareG and Table from NanoTableG
#assume DataOut is the same output specified for NanoTableG
#do not store ggplot2 tables:
NanoStatsG(List,Table, DataOut=DataOut)
#store ggplot2 tables:
NanoStatsG(List,Table, DataOut=DataOut,KeepGGObj=TRUE)
```

NanoStatsM

NanoStatsM

Description

Plot statistics

Usage

```
NanoStatsM(NanoMList, NanoMTable, DataOut, KeepGGObj = FALSE)
```

Arguments

NanoMList	Object of class list from NanoPrepareM
NanoMTable	Object of class matrix from NanoTableM
DataOut	Output folder (the same used for NanoTableM)
KeepGGObj	Logical. If TRUE, store ggplot2 data frames behind generated plots [FALSE]

Value

Plots:

- Yield.pdf (accumulation of reads and bps);
- RBLQ.pdf (# reads, # bps, length and quality overview every 30 minutes of experiment);
- LvQ.pdf (length and quality compared jointly);
- PFGC.pdf (passed and failed reads, GC content if computed);
- Activity.pdf (channels and muxes activity (# bps). Inactive channels and muxes are grey-colored)

Table:

- summary.txt (table with major statistics for the sequencing run)

Examples

```
#do not run
#assume List is the output from NanoPrepareM and Table from NanoTableM
#assume DataOut is the same output specified for NanoTableM
#do not store ggplot2 tables:
NanoStatsM(List,Table, DataOut=DataOut)
#store ggplot2 tables:
NanoStatsM(List,Table, DataOut=DataOut,KeepGGObj=TRUE)
```

NanoTableG

NanoTableG

Description

Generate a metadata table

Usage

```
NanoTableG(NanoGList, DataOut, GCC = FALSE)
```

Arguments

NanoGList	Object of class list from NanoPrepareG
DataOut	Output folder
GCC	Logical. If TRUE, compute GC content from FASTQ [FALSE]

Value

Object of class matrix

Examples

```
#do not run
#assume List is the output from NanoPrepareG
DataOut <- "/path/to/output"
Table<-NanoTableG(List,DataOut)
#include GC content
Table<-NanoTableM(List,DataOut,GCC=TRUE)
```

NanoTableM	<i>NanoTableM</i>
------------	-------------------

Description

Generate a metadata table

Usage

```
NanoTableM(NanoMList, DataOut, Cores = 1, GCC = FALSE)
```

Arguments

NanoMList	Object of class list from NanoPrepareM
DataOut	Output folder
Cores	Number of cores to accelerate metadata extraction [1]
GCC	Logical. If TRUE, compute GC content [FALSE]

Value

Object of class matrix

Examples

```
#do not run
#assume List is the output from NanoPrepareM
DataOut <- "/path/to/output"
Table<-NanoTableM(List,DataOut,Cores=6)
#include GC content
Table<-NanoTableM(List,DataOut,Cores=6, GCC=TRUE)
```

Index

NanoCompare, [2](#)
NanoPrepareG, [2](#)
NanoPrepareM, [3](#)
NanoStatsG, [4](#)
NanoStatsM, [4](#)
NanoTableG, [5](#)
NanoTableM, [6](#)