Package ‘frenchFISH’

May 17, 2024

Type Package
Title Poisson Models for Quantifying DNA Copy-number from FISH Images of Tissue Sections
Version 1.16.0
Author Adam Berman, Geoff Macintyre
Maintainer Adam Berman <agb61@cam.ac.uk>
Description FrenchFISH comprises a nuclear volume correction method coupled with two types of Poisson models: either a Poisson model for improved manual spot counting without the need for control probes; or a homogenous Poisson Point Process model for automated spot counting.
License Artistic-2.0
Encoding UTF-8
Imports utils, MCMCpack, NHPPoisson
RoxygenNote 7.0.2
Suggests knitr, rmarkdown, testthat
biocViews Software, BiomedicalInformatics, CellBiology, Genetics, HiddenMarkovModel, Preprocessing
VignetteBuilder knitr
git_url https://git.bioconductor.org/packages/frenchFISH
git_branch RELEASE_3_19
git_last_commit aaf4bcc
git_last_commit_date 2024-04-30
Repository Bioconductor 3.19
Date/Publication 2024-05-17

Contents

areAllNonnegativeIntegers .................................................. 2
checkAutomaticCountsEstimatesArguments .............................. 2
areAllNonnegativeIntegers

*Helper function to check if all values in the input count matrix are either NA, NaN, or non-negative integers*

**Description**

Helper function to check if all values in the input count matrix are either NA, NaN, or non-negative integers

**Usage**

```r
areAllNonnegativeIntegers(count_matrix)
```

**Arguments**

- `count_matrix` The count matrix

**Value**

TRUE if all values in `count_matrix` are non-NA/NaN, non-negative integers; otherwise FALSE

checkAutomaticCountsEstimatesArguments

*Helper function to check the arguments input to getAutomaticCountsEstimates*

**Description**

Helper function to check the arguments input to getAutomaticCountsEstimates

**Usage**

```r
checkAutomaticCountsEstimatesArguments(probeCounts, radius, height)
```
Arguments

probeCounts A matrix where the first column contains the areas of the nuclear blobs (this column must be named "area" and the unit of its entries must be the square of the unit used to measure radius and height) and the remaining columns (one per probe) contain the spot counts for different probes in each nuclear blob.

radius The cells’ nuclear radius (must be measured in same unit as height).

height The section height (must be measured in same unit as radius).

Value

Nothing if all checks are passed; otherwise throws an error or warning message.

Description

Helper function to check the arguments input to getManualCountsEstimates.

Usage

checkManualCountsEstimatesArguments(probeCounts, radius, height)

Arguments

probeCounts A matrix of manual spot counts with columns for probes and rows for nuclei.

radius The cells’ nuclear radius (must be measured in same unit as height).

height The section height (must be measured in same unit as radius).

Value

Nothing if all checks are passed; otherwise throws an error or warning message.
convertFishalyserCsvToCountMatrix

Function to convert CSV output of the FISHalyseR automatic FISH splot counting software to a count matrix suitable for input to frenchFISH’s getAutomaticCountsEstimates

Description

Function to convert CSV output of the FISHalyseR automatic FISH splot counting software to a count matrix suitable for input to frenchFISH’s getAutomaticCountsEstimates

Usage

convertFishalyserCsvToCountMatrix(pathToFishalyserCsv)

Arguments

pathToFishalyserCsv

The path to the CSV file of automatic spot counts outputted by FISHalyseR

Value

A count matrix suitable for input to getAutomaticCountsEstimates

Examples

probeCounts<-convertFishalyserCsvToCountMatrix(
    system.file("extdata", "SampleFISH.jpg_data.csv", package="frenchFISH"))

generatePPdat

Helper function to convert spot counts and nuclear area measurements into continuous events for Poisson point estimation

Description

Helper function to convert spot counts and nuclear area measurements into continuous events for Poisson point estimation

Usage

generatePPdat(area, spots)

Arguments

area

The nuclear area

spots

The number of spots counted
Value

Vector of continuous events for Poisson point estimation

Description

FrenchFISH function for generating Poisson point estimates of spot counts from spot counts which have been automatically generated.

Usage

getAutomaticCountsEstimates(probeCounts, radius, height)

Arguments

probeCounts A matrix where the first column contains the areas of the nuclear blobs (this column must be named "area" and the unit of its entries must be the square of the unit used to measure radius and height) and the remaining columns (one per probe) contain the spot counts for different probes in each nuclear blob
radius The cells’ nuclear radius (must be measured in same unit as height)
height The section height (must be measured in same unit as radius)

Value

The Poisson point estimates of spot counts for each probe

Examples

automaticCountsEstimates<-getAutomaticCountsEstimates(
  cbind(area=c(250,300,450),
  red=c(0,2,4),
  green=c(5,3,1),
  blue=c(3,0,2)), 8, 4)
getAverageVolumeFrac  
*Helper function to get the average volume of nucleus sampled given the nucleus radius and section height*

**Description**
Helper function to get the average volume of nucleus sampled given the nucleus radius and section height.

**Usage**
getAverageVolumeFrac(r, h)

**Arguments**
- **r**  
The nuclear radius
- **h**  
The section height

**Value**
The average volume of nucleus sampled given the nucleus radius and section height

getManualCountsEstimates  
*FrenchFISH function for generating volume adjusted spot counts from spots which have been manually counted (uses a Markov chain Monte Carlo method).*

**Description**
FrenchFISH function for generating volume adjusted spot counts from spots which have been manually counted (uses a Markov chain Monte Carlo method).

**Usage**
getManualCountsEstimates(probeCounts, radius, height)

**Arguments**
- **probeCounts**  
A matrix of manual spot counts with columns for probes and rows for nuclei
- **radius**  
The cells’ nuclear radius (must be measured in same unit as height)
- **height**  
The section height (must be measured in same unit as radius)

**Value**
The volume adjusted spot counts for each probe that have been generated using MCMC modelling
getMaxVolumeFrac

Examples

manualCountsEstimates<-getManualCountsEstimates(cbind(red=c(0,2,4),
green=c(5,3,1), blue=c(3,0,2)), 8, 4)

getMaxVolumeFrac

Helper function to get the maximum possible volume of nucleus sampled given the nucleus radius and section height

Description

Helper function to get the maximum possible volume of nucleus sampled given the nucleus radius and section height

Usage

getMaxVolumeFrac(r, h)

Arguments

r The nuclear radius
h The section height

Value

The maximum possible volume of nucleus sampled given the nucleus radius and section height

getMinVolumeFrac

Helper function that returns the minimum possible volume of nucleus sampled given the nucleus radius and section height

Description

Helper function that returns the minimum possible volume of nucleus sampled given the nucleus radius and section height

Usage

getMinVolumeFrac(r, h)

Arguments

r The radius of the nuclei
h The height of the section

Value

The minimum possible volume of nucleus sampled given the nucleus
getVsegFrac

Helper function that returns the fraction of the nucleus sampled for a specified distance from the midpoint

Description

Helper function that returns the fraction of the nucleus sampled for a specified distance from the midpoint

Usage

getVsegFrac(d, h, r)

Arguments

d       The distance sampled from the midpoint
h       The height of the section
r       The radius of the nuclei

Value

The fraction of the nucleus sampled for a specified distance from the midpoint
Index

areAllNonnegativeIntegers, 2
checkAutomaticCountsEstimatesArguments, 2
checkManualCountsEstimatesArguments, 3
convertFishalyserCsvToCountMatrix, 4
generatePPdat, 4
getAutomaticCountsEstimates, 5
getAverageVolumeFrac, 6
getManualCountsEstimates, 6
getMaxVolumeFrac, 7
getMinVolumeFrac, 7
getVsegFrac, 8