Package 'mSigBG'

July 8, 2020

Title Separate Background Mutational Signature from Signature Due to a Mutagen

Version 1.0.0

Description Separate a ``background mutational signature'' from a set of observed
mutational spectra. Designed for the delineation of signatures from cell
cultures exposed to mutagens. 'mSigBG' stands for Mutational Signature
BackGround.

License GPL-3

Encoding UTF-8

Language en-US

Depends R (>= 4.0)

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 7.1.0

biocViews

Imports ICAMS,
nloptr,
stats,
lsa

Suggests testthat,
devtools,
usethis,
rmarkdown

R topics documented:

1

---

background.info   *Specifications of background signatures*

Description

Specifications of background signatures

Usage

```
background.info
```

Format

A list with one element for each cell line. Each element of the list is in turn a list with the elements

background.sig The background signatures as a single-column [ICAMS](#) catalog.

sig.nbinom.size The negative binomial size dispersion parameter for the background signature profile. See [NegBinomial](#).

count.nbinom.mu The mean of the numbers of mutations in input.spectra.

count.nbinom.size The negative binomial size dispersion parameter for the numbers of mutations caused by the background signature (i.e. count.nbinom.mu). See [NegBinomial](#).

input.spectra The bg.spectra used to estimate the background.

Source

background.info was estimated from [HepG2.background.spectra](#) and [MCF10A.background.spectra](#).

Examples

```
background.info[["HepG2"]]$count.nbinom.mu
background.info[["HepG2"]]$count.nbinom.size
background.info[["HepG2"]]$sig.nbinom.size
background.info[["HepG2"]]$background.sig[1:3, ]
```

---

background.spectra   *Background spectra for the HepG2 and MCF-10A cell lines.*

Description

Background spectra for the HepG2 and MCF-10A cell lines.

Usage

```
HepG2.background.spectra

MCF10A.background.spectra
```

Format

An [ICAMS](#) `counts` catalog.

An object of class SBS96Catalog (inherits from `matrix`) with 96 rows and 3 columns.

---

example.spectra	*Example spectra of cell lines exposed to cisplatin.*

Description

Example spectra of cell lines exposed to cisplatin.

Usage

```
example.spectra
```

Format

A list of [ICAMS](#) `counts` spectra catalogs.

Examples

```
rowSums(example.spectra[["MCF10A.cisplatin"]])[1:3]
```

LLHSpectrumNegBinom	*Likelihood that 1 observed spectrum was generated from a vector of expected mutation counts.*

Description

Likelihood that 1 observed spectrum was generated from a vector of expected mutation counts.

Usage

```
LLHSpectrumNegBinom(spectrum, expected.counts, nbinom.size, verbose = FALSE)
```

Arguments

spectrum	An observed spectrum (a numeric vector)
expected.counts	
	A vector of (integer) expected mutation counts, one expected count for each mutation type. We want to know the likelihood that this model generated the observed spectrum, assuming each mutational types generates counts according to a negative binomial distribution with the given expected.counts (argument mu to [NegBinomial](#)) and dispersion parameter nbinom.size.
nbinom.size	The dispersion parameter for the negative binomial distribution; smaller is more dispersed. See [NegBinomial](#).
verbose	If TRUE print messages under some circumstances.

Value

log(likelihood(spectrum | expected.counts)), or, in more detail, the sum of the negative binomial likelihoods that each each element of the spectrum (i.e., the count for each mutation type e.g. ACT > AAT) was generated from the expected count for that mutation type.

MakeBackgroundInfo   *Estimate a background signature and its characteristics.*

Description

Estimate a background signature and its characteristics.

Usage

```
MakeBackgroundInfo(
  bg.spectra,
  title = "Background.sig",
  sig.nbinom.size = 10,
  count.nbinom.size = 10
)
```

Arguments

bg.spectra   The spectra from which to compute the background information.

title   The name of the single column of the signature in the background.sig element of the output.

sig.nbinom.size

                The negative binomial size dispersion parameter for the background signature profile. See [NegBinomial](). Smaller is more dispersed.

count.nbinom.size

                The negative binomial size dispersion parameter for the numbers of mutations caused by the background signature (i.e. count.nbinom.mu). See [NegBinomial](). Smaller is more dispersed.

Value

A list with the elements

background.sig The background signatures as a single-column [ICAMS]() catalog.

sig.nbinom.size See input argument sig.nbinom.size.

count.nbinom.mu The mean of the numbers of mutations in bg.spectra.

count.nbinom.size See input argument count.nbinom.size.

input.spectra The bg.spectra used to estimate the background.

MeanOfSpectraAsSig *Return the mean of multiple spectra as a signature.*

Description

Return the mean of multiple spectra as a signature.

Usage

```
MeanOfSpectraAsSig(spectra, title = "sig.from.spectra.mean")
```

Arguments

spectra   An ICAMS spectrum catalog. Convert each spectrum to a signature and then compute the mean.

title   The name of the output signature.

Plot1StackedSpectrum *Plot a spectrum as a stacked bar chart*

Description

Plot a spectrum as a stacked bar chart

Usage

```
Plot1StackedSpectrum(
  background.spectrum,
  target.spectrum,
  background.title = "Background",
  target.title = "Target",
  set.neg.zero = TRUE
)
```

Arguments

background.spectrum
                 Partial spectra due to a background signature.

target.spectrum
                 Partial spectra due to a target signature.

background.title
                 A title for the legend for the partial spectra due to a background signature.

target.title   A title for the legend for the partial spectra due to a target signature.

set.neg.zero   Sometimes after subtraction a part of the spectra due to a target signatures is negative. If this argument is true, set these to 0.

PlotSpectraAsSigsWithUncertainty

Convert spectra to signatures and then plot mean with "error" bars

Description

Convert spectra to signatures and then plot mean with "error" bars

Usage

```
PlotSpectraAsSigsWithUncertainty(spectra, title = "Mean.as.signature")
```

Arguments

spectra	An [ICAMS](#) spectrum catalog. Convert each spectrum to a signature and then compute the mean.
title	The name of the output signature.

Value

The mean of the spectra as a signature, the constituent spectra as signatures, and the y positions of the arrowheads.

SeparateSignatureAndSpectra

Calculate the inferred target signature and inferred background and target components of input spectra

Description

Calculate the inferred target signature and inferred background and target components of input spectra

Usage

```
SeparateSignatureAndSpectra(
  spectra,
  bg.sig.info,
  m.opts = NULL,
  start.b.fraction = 0.1,
  sig.name = "Inferred.sig"
)
```

Arguments

`spectra`	The spectra from which to subtract the background, as a matrix or [ICAMS] catalog.
`bg.sig.info`	Information about the background signature. See [background.info].
`m.opts`	Options to pass to [nloptr].
`start.b.fraction`	
	The estimated fraction of the mutations in `spectra` due to the background signature.
`sig.name`	A name for the inferred signature

Value

A list with the elements

`inferred.target.spectra` The mutations counts inferred to be from the target signature.

SeparateSignatureFromBackground

Estimate a signature from experimentally exposed spectra minus a background signature.

Description

We index mutation channels (e.g. `ACA > AAA`, `ACC > AAC`, . . .) by j, $j \in 1...96$.

We index input mutational spectra by i.

Let

$g = g_1, g_2, \ldots, g_{96}$, with $\Sigma g_j = 1$, be the previously determined, input background signature profile,

$s^i, i \in 1, 2, \ldots$ be the input spectra, from exposed samples, usually only 2 or 3,

$b^i, i \in 1, 2, \ldots$ be the (to-be-estimated) numbers of mutations due to the background signature in each s^i, and

$t = t_1, t_2, \ldots, t_{96}$, with $\Sigma t_j = 1$, be the (to-be-estimated) target signature due to an exposure.

We want to maximize $\Pi^i P(s^i|b^i, t)P(b^i)$ over $b^1, b^2, \ldots$ and t. (Note that the code actually minimizes the additive inverse of this.)

$P(b^i)$ is estimated from the distribution of previously observed numbers of mutations in untreated samples, with the additional constraint that $b^i \leq |s^i|$), where $|s^i|$ is defined as the total number of mutations in spectrum s^i, i.e. $|s^i| = \Sigma_j s^i_j$, $j \in 1...96$.

$P(s^i|b^i, t)$ is estimated as follows:

The expected number of mutations in each mutation category, j, is estimated as

$e^i_j = g_j b^i + t_j(|s^i| - b^i)$.

Then $P(s^i|e^i)$ is estimated as $\Pi_j P(s^i_j|e^i_j)$.

$P(s^i_j|e^i_j)$ is estimated from a negative binomial distribution centered on each e^i_j; see the `sig.nbinom.size` elements of the [background.info] package variables.

Usage

```
SeparateSignatureFromBackground(
  spectra,
  bg.sig.info,
  m.opts = NULL,
  start.b.fraction = 0.1
)
```

Arguments

spectra	The spectra from which to subtract the background, as a matrix or [ICAMS](#) catalog.
bg.sig.info	Information about the background signature. See [background.info](#).
m.opts	Options to pass to [nloptr](#).
start.b.fraction	
	The estimated fraction of the mutations in spectra due to the background signature.

Details

See [ObjFn1](#).

Value

A list with the elements

inferred.target.sig The estimated target signature as a numerical vector.

exposures.to.target.sig The estimated total number of mutations due to the target signature in each input spectrum.

exposures.to.bg.sig The estimated total number of mutations due to the background in each input spectrum.

message The message element of all.opt.ret.

all.opt.ret The entire return value from the optimization. See [nloptr](#)

SeparateSignatureFromBackgroundOptions

Return a default value to pass as the m.opts *argument to* [SeparateSignatureFromBackground](#).

Description

Return a default value to pass as the m.opts argument to [SeparateSignatureFromBackground](#).

Usage

```
SeparateSignatureFromBackgroundOptions()
```

Index