

Package ‘CDFSnipeR’

February 28, 2018

Title Removal of Probes from an Affymetrix Chip Definition File (CDF)

Version 0.0.9.9000

Description Affymetrix microarrays consist of multiple probes targeting each gene/transcript. These probes were designed against the reference genome (in mouse, this is C57BL/6). Sequence divergence (SNPs or indels) between the probe and the sample can lead to detection differences that are not due to differences in gene expression. Removal of polymorphic probes and reassembly into new probe-sets may provide more stable quantification estimates.

Depends R (>= 3.4.0)

License LGPL-3

Encoding UTF-8

LazyData true

Imports GenomicRanges, IRanges, S4Vectors, stats, utils

RoxygenNote 6.0.1

NeedsCompilation no

Author Rupert Overall [aut, cre]

Maintainer Rupert Overall <mail@rupertoverall.net>

R topics documented:

<code>cdf.sniper</code>	1
<code>CDFSnipeR</code>	2
<code>remove.probes</code>	3

Index	4
--------------	----------

<code>cdf.sniper</code>	<i>Removal of probes from a Chip Definition File (CDF).</i>
-------------------------	---

Description

`cdf.sniper` rebuilds a CDF using only whitelisted probes.

Usage

```
cdf.sniper(cdf.file, clean.probes, min.probeset.size = 3, keep.affx = TRUE,  
          newline = "\n", cluster = NULL)
```

Arguments

<code>cdf.file</code>	The CDF which is to be manipulated. This can also be a connection (or URL, anything that is readable by <code>utils::read.table</code>).
<code>clean.probes</code>	A whitelist of probe coordinates to be retained.
<code>min.probeset.size</code>	The minimum number of probes in a probset. If fewer than this number remain after probe removal, the entire probset will be discarded.
<code>keep.affx</code>	A logical flag indicating whether the AFFX control probes should be retained.
<code>newline</code>	It may be necessary to change this on non-UNIX-like operating systems.
<code>cluster</code>	An object of class 'cluster' as provided by the package 'parallel'. If this is provided, the computationally expensive part of the code can be distributed over multiple CPUs.

Details

Write some text here about the format of the CDF, where these can be obtained including links. The format of the whitelist should also be detailed (as returned by `remove.probes`).

Value

A string containing the CDF text. It is intended that this is written to a file using `write`.

Examples

```
## Not run:
newProbes <- remove.probes(oldProbes, snpPositions)
newCdf <- cdf.sniper(oldCdf, newProbes$FilteredProbes)
write(newCdf, file="newCdf.cdf")

## End(Not run)
```

CDFSnipeR

CDFSnipeR: Removal of Probes from Affymetrix Chip Definition Files.

Description

The CDFSnipeR package provides two functions: `remove.probes` to detect and remove probes containing undesired sequences and `cdf.sniper` to rebuild the CDF without these probes.

Functions provided

`remove.probes` `cdf.sniper`

remove.probes	<i>Removal of probes from a probe list.</i>
---------------	---

Description

remove.probes removes probes containing undesired features from a CDF.

Usage

```
remove.probes(probes, features, probe.length = 25)
```

Arguments

probes	A data frame containing probe sequence information.
features	A data frame containing genomic coordinates of undesired features. This should be in a BED-like format with the first three columns containing the chromosome (with or without the 'chr'), the start position (in nucleotides) and the end position (the same as the start in the case of SNPs).
probe.length	A number specifying the length of the probes on the chip. This is 25 by default.

Value

A list containing the following components: FilteredProbes: A character vector of the whitelisted probes (i.e. those not containing the features defined in the BED file). ProbeCounts: a matrix with one row for each probeset and two columns containing the numbers of probes in each probeset both before (OldProbes) and after (CleanProbes) probe removal. RemovedProbes: A character vector of the removed probes (those containing the features defined in the BED file).

Examples

```
## Not run:
whitelist <- remove.probes(testProbes, testfeatures)
newCdf <- cdf.sniper(cdf, whitelist)
write(newCdf, file="newCdf_deSNP.cdf")

## End(Not run)
```

Index

`cdf.sniper`, [1](#), [2](#)
`CDFSniperR`, [2](#)
`CDFSniperR-package (CDFSniperR)`, [2](#)
`remove.probes`, [2](#), [3](#)
`write`, [2](#)