Package 'ICAMS'

March 18, 2019

Type Package

Title In-depth Characterization and Analysis of Mutational Signatures

Version 1.0.0

Author Steve Rozen, Nanhai Jiang, Arnoud Boot

Maintainer Steve Rozen <steverozen@gmail.com>

Description A toolkit for analysis and visualization of experimentally elucidated mutational signatures -- the kind of analysis and visualization presented in Boot et al., ``In-depth characterization of the cisplatin mutational signature in human cell lines and in esophageal and liver tumors", 2018, https://genome.cshlp.org/content/28/5/654.short. This package has functions to read in variant call files and to collate and plot the mutational spectra.

License GPL-3

Encoding UTF-8

LazyData true

Language en-US

biocViews

Imports Biostrings,
BSgenome,
BSgenome.Hsapiens.1000genomes.hs37d5,
BSgenome.Hsapiens.UCSC.hg38,
data.table,
dplyr,
GenomicRanges,
graphics,
grDevices,
methods,
RColorBrewer,
RCurl,
stats,
stringr,
utils

Depends R (>= 3.5),

RoxygenNote 6.1.1

Suggests knitr,
rmarkdown,
testthat

R topics documented:

`CatalogRowOrder` *Standard order of row names in a catalog.*

Description

This data is designed for those who need to create their own catalogs from formats not supported by this package. The rownames denote the mutation types. For example, for SNS96 catalogs, the rowname AGAT represents a mutation from AGA > ATA.

Usage

```
catalog.row.order
```

Format

A list of character vectors indicating the standard orders of row names in catalogs.

Note

In the ID (insertion and deletion) catalog, deletion repeat size is in the range from 0 to 5+, but for plotting and end user documentation it ranges from 1 to 6+.

`CollapseCatalog` *"Collapse" a catalog.*

Description

"Collapse" a catalog. Do not use this function for signature catalogs.

Usage

```
Collapse192To96(catalog)

Collapse1536To96(catalog)

Collapse144To78(catalog)
```

Arguments

`catalog`	A catalog as defined in `ICAMS`.

Details

`Collapse192To96` Collapse an SNS 192 catalog to an SNS 96 catalog.

`Collapse1536To96` Collapse an SNS 1536 catalog to an SNS 96 catalog.

`Collapse144To78` Collapse a DNS 144 catalog to a DNS 78 catalog.

Value

A catalog as defined in `ICAMS`.

`FindDelMH` *Return the length of microhomology at a deletion.*

Description

Return the length of microhomology at a deletion.

Usage

```
FindDelMH(context, deleted.seq, pos, trace = 0)
```

Arguments

`context`	The deleted sequence plus ample surrounding sequence on each side (at least as long as `del.sequence`).
`deleted.seq`	The deleted sequence in `context`.
`pos`	The position of `del.sequence` in `context`.
`trace`	If > 0, cat various messages.

Details

This function is primarily for internal use, but we export it to document the underlying logic.

Example:

`GGCTAGTT` aligned to `GGCTAGAACTAGTT` with a deletion represented as:

```
GGCTAGAACTAGTT
GG------CTAGTT GGCTAGTT GG[CTAGAA]CTAGTT
                           ----   ----
```

Presumed repair mechanism leading to this:

```
  ....
GGCTAGAACTAGTT
CCGATCTTGATCAA

=>

  ....
GGCTAG      TT
CC      GATCAA
        ....

=>

GGCTAGTT
CCGATCAA
```

Variant-caller software can represent the same deletion in several different, but completely equivalent, ways.

```
GGC------TAGTT GGCTAGTT GGC[TAGAAC]TAGTT
                         * ---  * ---

GGCT------AGTT GGCTAGTT GGCT[AGAACT]AGTT
                         ** --  ** --

GGCTA------GTT GGCTAGTT GGCTA[GAACTA]GTT
                         *** -  *** -

GGCTAG------TT GGCTAGTT GGCTAG[AACTAG]TT
                         ****   ****
```

A deletion in a *repeat* can also be represented in several different ways. A deletion in a repeat is abstractly equivalent to microhomology that spans the entire deleted sequence. For example;

```
GACTAGCTAGTT
GACTA----GTT GACTAGTT GACTA[GCTA]GTT
                       *** -*** -
```

is really a repeat

```
GACTAG----TT GACTAGTT GACTAG[CTAG]TT
                        **** ----

GACT----AGTT GACTAGTT GACT[AGCT]AGTT
                        ** --** --
```

This function only flags this case with a -1 return; it does not figure out the repeat extent.

This function finds:

1. The maximum match of undeleted sequence to the left of the deletion that is identical to the right end of the deleted sequence, and
2. The maximum match of undeleted sequence to the right of the deletion that is identical to the left end of the deleted sequence.

The microhomology sequence is the concatenation of items (1) and (2).

Value

The length of the maximum microhomology of `del.sequence` in `context`.

`GetVAF` *Extract the VAFs (variant allele frequencies) from a VCF file.*

Description

Extract the VAFs (variant allele frequencies) from a VCF file.

Usage

```
GetStrelkaVAF(vcf)

GetMutectVAF(vcf)
```

Arguments

`vcf`	said VCF as a data.frame.

Value

A vector of VAFs, one for each row of `vcf`.

ICAMS *ICAMS: In-depth Characterization and Analysis of Mutational Signatures*

Description

A toolkit for analysis and visualization of experimentally elucidated mutational signatures – the kind of analysis and visualization presented in Boot et al., "In-depth characterization of the cisplatin mutational signature in human cell lines and in esophageal and liver tumors", *Genome Research*, 2018, https://genome.cshlp.org/content/28/5/654.short.

Details

ICAMS can read in variant call files (VCFs) generated by Strelka or Mutect, and collate the mutations into "catalogs" of mutational spectra. ICAMS can plot the catalogs of mutational spectra and signatures.

ICAMS can create and plot catalogs of mutational spectra and signatures for single nucleotide substitutions (SNS), double nucleotide substitutions (DNS), and small insertions and deletions (ID). It can also read and write these catalogs.

Catalogs and signatures

A key data type in `ICAMS` is a "catalog" of mutation counts, of mutation densities, or of mutational signatures.

A catalog is one of the following:

1. Matrix of mutation counts (one column per sample), representing (count-based) mutational spectra.
2. Matrix of mutation densities, i.e. mutations per occurrences of source sequences (one column per sample), representing (density-based) mutational spectra.
3. Matrix of mutational signatures, which are similar to spectra. However where spectra consist of counts or densities of mutations in each mutation class (e.g. ACA > AAA, ACA > AGA, ACA > ATA, ACC > AAC, ...), signatures consist of the proportions of mutations in each class (with all the proportions summing to 1).#' A mutational signature can be based on either:
   (a) mutation counts (a "count-based mutational signature"), or
   (b) mutation densities (a "density-based mutational signature").

If you need to create a catalog from a source other than this package (i.e. other than with `ReadCatalog` or `StrelkaSNSVCFFilesToCatalog`, `MutectVCFFilesToCatalog`, etc.), then you must ensure that the rows are in the expected order and have the expected rownames. See `CatalogRowOrder` for the expected rownames and order of rows.

Creating catalogs from variant call files (VCF files)

1. `StrelkaSNSVCFFilesToCatalog` creates 3 SNS catalogs (96, 192, 1536) and 3 DNS catalogs (78, 136, 144) from the Strelka SNS VCFs.
2. `StrelkaIDVCFFilesToCatalog` creates ID (indel) catalog from the Strelka ID VCFs.
3. `MutectVCFFilesToCatalog` creates 3 SNS catalogs (96, 192, 1536), 3 DNS catalogs (78, 136, 144) and ID (indel) catalog from the Mutect VCFs.

The genome argument

Many functions take the argument genome. This can be either

1. A variable from the Bioconductor BSgenome package that contains a particular reference genome, for example BSgenome.Hsapiens.1000genomes.hs37d5. BSgenome::available.genomes() returns the available genomes.
2. The strings "hg38" or "GRCh38" are shorthand for BSgenome.Hsapiens.UCSC.hg38, and the strings "hg19" or "GRCh37" are shorthand for BSgenome.Hsapiens.1000genomes.hs37d5.

The Bioconductor BSgenome package

This package will be installed automatically if ICAMS is installed with devtools::install_local or with devtools::install_github. Otherwise you must manually install BSgenome and the necessary genomes, e.g.

```
BSgenome.Hsapiens.1000genomes.hs37d5.
```

See instructions at

```
https://bioconductor.org/packages/release/bioc/html/BSgenome.html.
```

Genomes other than the two human genomes mentioned above are not installed automatically.

Use available.genomes to get the list of available genomes.

Plotting catalogs

Functions for plotting catalogs of mutational spectra or of mutational signatures of **one** sample. PlotCatalog

The PlotCatalogToPdf functions plot catalogs of mutational spectra or of mutational signatures to a PDF file.

Writing catalogs

The WriteCatalog functions write a catalog of mutational spectra or of mutational signatures to a file.

Reading catalogs

The ReadCatalog functions read a file that contains a catalog of mutational spectra or of signatures in standardized format.

Transforming catalogs

The TransformCatalog function transforms catalogs of mutational spectra or signatures to account for differing abundances of the source sequence of the mutations in the genome.

For example, mutations from ACG are much rarer in the human genome than mutations from ACC simply because CG dinucleotides are rare in the genome.

This function can also transform spectra based on observed genome-wide counts to "density"-based catalogs. In density-based catalogs mutations are expressed as mutations per source sequences. For example, a density-based catalog represents the proportion of ACCs mutated to ATCs, the proportion of ACGs mutated to ATGs, etc. This is different from count-based catalogs, which contain the number of ACC > ATC mutations, the number of ACG > ATG mutations, etc.

This function can also transform observed-count based spectra or signatures from genome to exome based counts, or between different species (since the abundances of source sequences vary between genome and exome and between species).

Collapsing catalogs

The CollapseCatalog functions

1. take a mutational spectrum or signature catalog that is based on a fined-grained set of features (for example, single-nucleotide substitutions in the context of the preceding and following 2 bases), and
2. collapse it to a catalog based on a coarser-grained set of features (for example, single-nucleotide substitutions in the context of the immediately preceding and following bases).

Data

1. CatalogRowOrder Standard order of rownames in a catalog. The rownames of encode the type of each mutation. The rownames denote the mutation types. For example, for SNS96 catalogs, the rowname AGAT represents a mutation from AGA > ATA.
2. TranscriptRanges Transcript ranges and strand information for a particular reference genome.

MutectVCFFilesToCatalog

Create SNS and DNS catalogs from Mutect VCF files

Description

Create 3 SNS catalogs (96, 192, 1536) and 3 DNS catalogs (78, 136, 144) from the Mutect VCFs specified by vector.of.file.paths

Usage

```
MutectVCFFilesToCatalog(vector.of.file.paths, genome, trans.ranges)
```

Arguments

vector.of.file.paths	Character vector of file paths to the Mutect VCF files.
genome	A genome argument as described in ICAMS.
trans.ranges	A data.table which contains transcript range and strand information.

Details

This function calls VCFsToSNSCatalogs, VCFsToDNSCatalogs and VCFsToIDCatalogs

Value

A list of 3 SNS catalogs (one each for 96, 192, and 1536), 3 DNS catalogs (one each for 78, 136, and 144) and ID catalog.

Note

SNS 192 and DNS 144 catalogs include only mutations in transcribed regions.

PlotCatalog *Plot one spectrum or signature.*

Description

Plot the spectrum of one sample or plot one signature.

Usage

```
PlotCatSNS96(catalog, type, id = colnames(catalog), cex = 0.8,
  grid = TRUE, upper = TRUE, xlabels = TRUE)

PlotCatSNS192(catalog, type, id = colnames(catalog), cex = 0.8)

PlotSNSClassStrandBias(catalog, type, id = colnames(catalog), cex = 1)

PlotCatSNS1536(catalog, type, id = colnames(catalog))

PlotCatDNS78(catalog, type, id = colnames(catalog))

PlotDNSClassStrandBias(catalog, type, id = colnames(catalog), cex = 1)

PlotCatDNS136(catalog, type, id = colnames(catalog))

PlotCatID(catalog, type, id = colnames(catalog))
```

Arguments

catalog	A one-column catalog as described in ICAMS. Please see ICAMS if you need to create a catalog from a source other than the current package, i.e. a source other than ReadCatalog or StrelkaSNSVCFFilesToCatalog, MutectVCFFilesToCatalog, etc.
type	A string specifying the type of the input catalog, one of: 1. "counts": show the counts of each mutation type. 2. "density", show the rates of mutations per million source n-mers for each mutation type; not supported for PlotCatIDToPdf, PlotCatSNS192ToPdf, PlotSNSClassStrandBiasToPdf, and PlotDNSClassStrandBiasToPdf. 3. "signature", show the proportions of each mutation type; not supported for PlotCatDNS136ToPdf.
id	A vector containing the identifiers of the samples or signatures in catalog.
cex	A numerical value giving the amount by which mutation class labels, mutation counts(if it exists), y axis and its labels, x axis labels and its annotations(if it exists) sample name and legend(if it exists) should be magnified relative to the default.
grid	If TRUE, draw grid lines in the graph.
upper	If TRUE, draw horizontal lines and the names of major mutation class on top of graph.
xlabels	If TRUE, draw x axis labels.

Details

`PlotCatSNS96` Plot an SNS 96 spectrum or signature.

`PlotCatSNS192` Plot an SNS 192 spectrum or signature.

`PlotSNSClassStrandBias` Plot the transcription strand bias graph of 6 SNS mutation types ("C>A", "C>G", "C>T", "T>A", "T>C", "T>G") in one sample.

`PlotCatSNS1536` Plot the pentanucleotide sequence contexts for one sample, normalized by pentanucleotide occurrence in the genome. The mutation types are in six-letters like CATTAT, first 2-letters CA refers to (-2, -1) position, third letter T refers to the base which has mutation, next second 2-letters TA refers to (+1, +2) position, last letter T refers to the base after mutation.

`PlotCatDNS78` Plot a DNS 78 spectrum or signature.

`PlotDNSClassStrandBias` Plot the transcription strand bias graph of 10 major DNS mutation types ("AC>NN", "AT>NN", "CC>NN", "CG>NN", "CT>NN", "GC>NN", "TA>NN", "TC>NN", "TG>NN", "TT>NN") in one sample.

`PlotCatDNS136` Plot the tetranucleotide sequence context of 10 major DNS mutation types ("AC>NN", "AT>NN", "CC>NN", "CG>NN", "CT>NN", "GC>NN", "TA>NN", "TC>NN", "TG>NN", "TT>NN") for one sample.

`PlotCatID` Plot an insertion and deletion spectrum or signature.

Value

invisible(TRUE)

PlotCatalogToPdf *Plot catalogs to a PDF file.*

Description

Plot catalogs to a PDF file.

Usage

```
PlotCatSNS96ToPdf(catalog, name, type, id = colnames(catalog),
  grid = TRUE, upper = TRUE, xlabels = TRUE)

PlotCatSNS192ToPdf(catalog, name, id = colnames(catalog),
  type = "counts", cex = 0.8)

PlotSNSClassStrandBiasToPdf(catalog, name, type, id = colnames(catalog),
  cex = 1)

PlotCatSNS1536ToPdf(catalog, name, type, id = colnames(catalog))

PlotCatDNS78ToPdf(catalog, name, type, id = colnames(catalog))

PlotDNSClassStrandBiasToPdf(catalog, name, type, id = colnames(catalog),
  cex = 1)

PlotCatDNS136ToPdf(catalog, name, type, id = colnames(catalog))

PlotCatIDToPdf(catalog, name, type, id = colnames(catalog))
```

Arguments

catalog	A catalog as described in ICAMS.
name	The name of the PDF file to be produced.
type	A string specifying the type of the input catalog, one of: 1. "counts", show the counts of each mutation type. 2. "density", show the rates of mutations per million source n-mers for each mutation type; not supported for PlotCatIDToPdf, PlotCatSNS192ToPdf, PlotSNSClassStrandBiasToPdf, and PlotDNSClassStrandBiasToPdf. 3. "signature", show the proportions of each mutation type; not supported for PlotCatDNS136ToPdf.
id	A vector containing the identifiers of the samples or signatures in catalog.
grid	If TRUE, draw grid lines in the graph.
upper	If TRUE, draw horizontal lines and the names of major mutation class on top of graph.
xlabels	If TRUE, draw x axis labels.
cex	A numerical value giving the amount by which mutation class labels, y axis labels, sample name and legend (if it exists) should be magnified relative to the default.

Details

PlotCatSNS96ToPdf Plot an SNS 96 catalog to a PDF file.

PlotCatSNS192ToPdf Plot an SNS 192 catalog to a PDF file.

PlotSNSClassStrandBiasToPdf Plot the transcription strand bias graph of 6 SNS mutation types ("C>A", "C>G", "C>T", "T>A", "T>C", "T>G") of various samples to a PDF file.

PlotCatSNS1536ToPdf Plot a 1536 mutation catalog to a PDF file. The mutation types are in six-letters like CATTAT, first 2-letters CA refers to (-2, -1) position, third letter T refers to the base which has mutation, next second 2-letters TA refers to (+1, +2) position, last letter T refers to the base after mutation.

PlotCatDNS78ToPdf Plot a DNS 78 mutation catalog to a PDF file.

PlotDNSClassStrandBiasToPdf Plot the transcription strand bias graph of 10 major DNS mutation types ("AC>NN", "AT>NN", "CC>NN", "CG>NN", "CT>NN", "GC>NN", "TA>NN", "TC>NN", "TG>NN", "TT>NN") of various samples to a PDF file.

PlotCatDNS136ToPdf Plot the tetranucleotide sequence contexts of 10 major DNS mutation types ("AC>NN", "AT>NN", "CC>NN", "CG>NN", "CT>NN", "GC>NN", "TA>NN", "TC>NN", "TG>NN", "TT>NN") of various samples to a PDF file.

PlotCatIDToPdf Plot a insertion and deletion catalog to a PDF file. (Note that sizes of repeats involved in deletions range from 0 to 5+ in the catalog rownames, but for plotting and end user documentation they ranges from 1 to 6+.)

Value

invisible(TRUE)

`ReadAndSplitMutectVCFs` *Read and split Mutect VCF files.*

Description

Read and split Mutect VCF files.

Usage

```
ReadAndSplitMutectVCFs(vector.of.file.paths)
```

Arguments

`vector.of.file.paths`
: Character vector of file paths to the Mutect VCF files.

Value

A list with 3 in-memory VCFs and two left-over VCF-like data frames with rows that were not incorporated into the first 3 VCFs, as follows:

1. `SNS` VCF with only single nucleotide substitutions.
2. `DNS` VCF with only doublet nucleotide substitutions as called by Mutect.
3. `ID` VCF with only small insertions and deletions.
4. `other.subs` VCF like data.frame with rows for coordinate substitutions involving 3 or more nucleotides, e.g. ACT > TGA or AACT > GGTA.
5. `multiple.alternative.alleles` VCF like data.frame with rows for variants with multiple alternative alleles, for example ACT mutated to both AGT and ACT at the same position.

See Also

`MutectVCFFilesToCatalog`

`ReadAndSplitStrelkaSNSVCFs` *Read and split Strelka SNS VCF files.*

Description

Read and split Strelka SNS VCF files.

Usage

```
ReadAndSplitStrelkaSNSVCFs(vector.of.file.paths)
```

Arguments

`vector.of.file.paths`
: Character vector of file paths to the Strelka SNS VCF files.

Value

A list of 3 in-memory objects as follows:

1. `SNS.vcfs` List of data.frames of pure SNS mutations – no DNS or 3+BS mutations.
2. `DNS.vcfs` List of data.frames of pure DNS mutations – no SNS or 3+BS mutations.
3. `ThreePlus` List of data.tables with the key CHROM, LOW.POS, HIGH.POS. containing rows that that in the input that did not represent SNSs or DNSs.

See Also

`StrelkaSNSVCFFilesToCatalog`

`ReadCatalog` *Read catalog.*

Description

Read a catalog in standardized format from path.

Usage

```
ReadCatSNS96(path, strict = TRUE)

ReadCatSNS192(path, strict = TRUE)

ReadCatSNS1536(path, strict = TRUE)

ReadCatDNS78(path, strict = TRUE)

ReadCatDNS144(path, strict = TRUE)

ReadCatDNS136(path, strict = TRUE)

ReadCatID(path, strict = TRUE)
```

Arguments

`path`	Path to a catalog on disk in the standardized format.
`strict`	If TRUE, then stop if additional checks on the input fail.

Details

`ReadCatSNS96` Read a 96 SNS catalog.

`ReadCatSNS192` Read a 192 SNS catalog.

`ReadCatSNS1536` Read a 1536 SNS catalog.

`ReadCatDNS78` Read a 78 DNS catalog.

`ReadCatDNS144` Read a 144 DNS catalog.

`ReadCatDNS136` Read a 136 DNS catalog.

`ReadCatID` Read an ID (insertion/deletion) catalog.

See also `WriteCatalog`

Value

A catalog in standard in-memory format.

`ReadStrelkaIDVCFs` *Read Strelka ID (insertion and deletion) VCF files.*

Description

Read Strelka ID (insertion and deletion) VCF files.

Usage

```
ReadStrelkaIDVCFs(vector.of.file.paths)
```

Arguments

`vector.of.file.paths`
: Character vector of file paths to the VCF files.

Value

A list of vcfs from vector.of.file.paths.

Note

In the ID (insertion and deletion) catalog, deletion repeat size ranges from 0 to 5+, but for plotting and end user documentation it ranges from 1 to 6+.

See Also

`StrelkaIDVCFFilesToCatalog`

`revc` *Reverse complement every string in* `string.vec`.

Description

Reverse complement every string in `string.vec`.

Usage

```
revc(string.vec)
```

Arguments

`string.vec` a vector of type character.

Value

A vector of type characters with the reverse complement of every string in `string.vec`.

StrelkaIDVCFFilesToCatalog

Create ID (indel) catalog from Strelka ID VCF files

Description

Create ID (indel) catalog from the Strelka ID VCFs specified by vector.of.file.paths

Usage

```
StrelkaIDVCFFilesToCatalog(vector.of.file.paths, genome)
```

Arguments

vector.of.file.paths	Character vector of file paths to the Strelka ID VCF files.
genome	A genome argument as described in ICAMS.

Details

This function calls VCFsToIDCatalogs

Value

An ID (indel) catalog

Note

In the ID (insertion and deletion) catalog, deletion repeat size ranges from 0 to 5+, but for plotting and end user documentation it ranges from 1 to 6+.

StrelkaSNSVCFFilesToCatalog

Create SNS and DNS catalogs from Strelka SNS VCF files.

Description

Create 3 SNS catalogs (96, 192, 1536) and 3 DNS catalogs (78, 136, 144) from the Strelka SNS VCFs specified by vector.of.file.paths

Usage

```
StrelkaSNSVCFFilesToCatalog(vector.of.file.paths, genome, trans.ranges)
```

Arguments

vector.of.file.paths	Character vector of file paths to the Strelka SNS VCF files.
genome	A reference genome as described in ICAMS.
trans.ranges	A data.table which contains transcript range and strand information.

Details

This function calls VCFsToSNSCatalogs and VCFsToDNSCatalogs.

Value

A list of 3 SNS catalogs (one each for 96, 192, and 1536) and 3 DNS catalogs (one each for 78, 136, and 144)

Note

SNS 192 and DNS 144 catalog only contains mutations in transcribed regions.

TranscriptRanges *Transcript ranges data*

Description

Transcript ranges and strand information for a particular organism

Usage

```
trans.ranges.GRCh37

trans.ranges.GRCh38
```

Format

A data.table which contains transcript range and strand information for a particular organism.

Details

`trans.ranges.GRCh37` A data.table which contains transcript range and strand information for **Human** GRCh37. It is derived from a raw **GFF3** format file, from which only the following four gene types are kept to facilitate transcriptional strand bias analysis: protein_coding, retained_intron, processed_transcript and nonsense_mediated_decay. It contains chromosome name, start, end position, strand information and gene name and is keyed by chrom, chromStart, and chromEnd. It can be used in function StrelkaSNSVCFFilesToCatalog.

`trans.ranges.GRCh38` A data.table which contains transcript range and strand information for **Human** GRCh38. It is derived from a raw **GFF3** format file, from which only the following four gene types are kept to facilitate transcriptional strand bias analysis: protein_coding, retained_intron, processed_transcript and nonsense_mediated_decay. It contains chromosome name, start, end position, strand information and gene name and is keyed by chrom, chromStart, and chromEnd. It can be used in function StrelkaSNSVCFFilesToCatalog.

`TransformCatalog` *Transform between count and density catalogs and signatures and between different source-sequence abundances.*

Description

Transform between count and density catalogs and signatures and between different source-sequence abundances.

Usage

```
TransformCatalog(catalog, which.n, source.type,
  target.type = source.type, source.abundance = NULL,
  target.abundance = NULL)
```

Arguments

`catalog`	An SNS or DNS catalog as described in ICAMS; must **not** be an ID (indel) catalog.
`which.n`	The length of the source sequences, one of 2:5.
`source.type`	A character specifying type of the input catalog, one of `"counts"`, `"signature"` or `"density"`.
`target.type`	A character specifying type of the output catalog, with the same possible values as `source.type`.
`source.abundance`	Either `NULL` or a numeric vector with one element for each source sequence for the mutation types in `catalog` or a string specifying such a vector, one of `"GRCh37.genome"`, `"GRCh37.exome"`, `"GRCh38.genome"`, or `"GRCh38.exome"`. This is the abundance upon which the counts, densities, or proportions in `catalog` are based. For example, for SNS in trinucleotide context, e.g. ACT > AGT or TAC > TTC, the source sequences are ACT and TAC.
`target.abundance`	Same possibilities as `source.abundance`.

Details

Only certain parings of type and abundance are legal, as follows:

1. The type `"density"` must always be associated with a `NULL` abundance.
2. The type `"signature"` is allowed to be associated with a `NULL` abundance. A `NULL` abundance indicates that the signature is a "density-based" signature (see ICAMS).
3. The type `"counts"` must **not** be associated with the `NULL` abundance.

Only the following transformations are legal:

1. `counts -> counts`
2. `counts -> density`
3. `counts -> signature`
4. `density -> counts` (in which case the semantics are to infer the genome-wide or exome wide counts based on the densities.)

5. density -> signature
6. signature -> signature

Value

A catalog as defined in ICAMS

VCFsToDNSCatalogs *Create DNS catalogs from VCFs*

Description

Create a list of 3 catalogs (one each for DNS78, DNS144 and DNS136) out of the contents in list.of.DNS.vcfs. The VCFs must not contain any type of mutation other then DNSs.

Usage

```
VCFsToDNSCatalogs(list.of.DNS.vcfs, genome, trans.ranges)
```

Arguments

list.of.DNS.vcfs	List of in-memory data frames of pure DNS mutations – no SNS or 3+BS mutations. The list names will be the sample ids in the output catalog.
genome	A genome argument as described in ICAMS.
trans.ranges	A data frame containing transcript ranges.

Value

A list of 3 DNS catalogs, one each for 78, 144, 136: catDNS78 catDNS144 catDNS136

Note

DNS 144 catalog only contains mutations in transcribed regions.

VCFsToIDCatalogs *Create ID (insertion and deletion) catalog from ID VCFs*

Description

Create ID (insertion and deletion) catalog from ID VCFs

Usage

```
VCFsToIDCatalogs(list.of.vcfs, genome)
```

Arguments

list.of.vcfs	List of in-memory VCFs. The list names will be the sample ids in the output catalog.
genome	A genome argument as described in ICAMS.

Value

An ID (indel) catalog

`VCFsToSNSCatalogs` *Create SNS catalogs from SNS VCFs*

Description

Create a list of 3 catalogs (one each for 96, 192, 1536) out of the contents in list.of.SNS.vcfs. The SNS VCFs must not contain DNSs, indels, or other types of mutations.

Usage

```
VCFsToSNSCatalogs(list.of.SNS.vcfs, genome, trans.ranges)
```

Arguments

`list.of.SNS.vcfs`	List of in-memory data frames of pure SNS mutations – no DNS or 3+BS mutations. The list names will be the sample ids in the output catalog.
`genome`	A genome argument as described in ICAMS.
`trans.ranges`	A data frame containing transcript ranges.

Value

A list of 3 SNS catalogs, one each for 96, 192, 1536: catSNS96 catSNS192 catSNS1536

Note

SNS 192 catalog only contains mutations in transcribed regions.

`WriteCatalog` *Write catalog.*

Description

Write a catalog to a file on disk.

Usage

```
WriteCatSNS96(ct, path, strict = TRUE)

WriteCatSNS192(ct, path, strict = TRUE)

WriteCatSNS1536(ct, path, strict = TRUE)

WriteCatDNS78(ct, path, strict = TRUE)

WriteCatDNS144(ct, path, strict = TRUE)
```

```
WriteCatDNS136(ct, path, strict = TRUE)

WriteCatID(ct, path, strict = TRUE)
```

Arguments

ct	A catalog as defined in ICAMS.
path	The path of the file to be written on disk.
strict	If TRUE, then fail if additional checks on the input fail.

Details

WriteCatSNS96 Write an SNS 96 catalog.

WriteCatSNS192 Write a SNS 192 catalog.

WriteCatSNS1536 Write a SNS 1536 catalog.

WriteCatDNS78 Write a DNS 78 catalog.

WriteCatDNS144 Write a DNS 144 catalog.

WriteCatDNS136 Write a 136 DNS catalog from path

WriteCatID Write a ID (insertion/deletion) catalog.

See also ReadCatalog

Index