Package: MetaEntropy (via r-universe)

October 26, 2026

Title Functional Shannon Entropy for Virome Mutational Analysis

Version 1.4

Date 2026-08-31

Description Estimates Shannon entropy, per gene and per genomic position, associated with non-synonymous mutation frequencies in viral populations, such as wastewater samples. The package uses codon translations for functional insights. Each amino acid can be treated as an individual state, resulting in a 20-state entropy computation, or grouped into one of six physicochemical classes, adding further functional context. Provides normalized values (0-1 scale) to facilitate the direct comparison of different genomic positions or total functional entropy across multiple metagenomes. Designed to analyze mutational data using tabular 'Single Nucleotide Variant' (SNV) frequency tables generated by variant callers (e.g., 'iVar' or 'LoFreq'), operating independently of consensus sequence estimation and multiple sequence alignment.

Encoding UTF-8

Depends R (>= 4.1.0)

LazyData true

Suggests rmarkdown, tidyr

Imports ggplot2, patchwork, beeswarm, ggbeeswarm, knitr, dplyr, grDevices, rlang

VignetteBuilder knitr

License MIT + file LICENSE

Config/roxygen2/version 8.1.0

RoxygenNote 7.3.3

NeedsCompilation no

Author Leandro Roberto Jones [aut, cre] (ORCID: <https://orcid.org/0000-0002-5877-4194>), Julieta Marina Manrique [aut] (ORCID: <https://orcid.org/0000-0001-8712-6666>)

Maintainer Leandro Roberto Jones <lrj000@gmail.com>

Repository https://jones-lr.r-universe.dev

Date/Publication 2026-09-03 14:10:02 UTC

RemoteUrl https://github.com/cran/MetaEntropy

RemoteRef HEAD

RemoteSha 9fdb9ebe35fb97535f2e19510fcae516c9e59466

Contents

as.data.frame.entropyProfile . . . 2
assessHotSpot . . . 3
createPositionSummary . . . 4
createStorage . . . 5
entropyProfile . . . 5
fillPosition . . . 7
getEntropySignature . . . 7
getPosEntropy . . . 9
heatmap_entropyProfiles . . . 10
intraHostVariants . . . 11
mn908947.3 . . . 11
plot.entropyProfile . . . 12
print.profileSummary . . . 13
print.tidyMutations . . . 13
showMutations . . . 14
summary.entropyProfile . . . 15
wWater . . . 15

Index **17**

`as.data.frame.entropyProfile` *Coerce* `entropyProfile` *to a Data Frame*

Description

Function to extract summary information from an `entropyProfile` object. This function is internally used for plotting.

Usage

```
## S3 method for class 'entropyProfile'
as.data.frame(x, row.names = NULL, optional = FALSE, ...)
```

Arguments

x	An object of class entropyProfile.
row.names	Please see as.data.frame.
optional	Please see as.data.frame.
...	Additional arguments passed to the function.

Value

A data frame with tabular information on an entropy profile. This information includes the name of the proteins presenting mutations, the corresponding genomic positions, and the resulting entropies in the metagenome.

---

assessHotSpot *Evaluates Entropy Hotspot*

Description

Graphical and formal analyses of contiguous amino acids.

Usage

```
assessHotSpot(profile, boundaries, chartType = "boxplot")
```

Arguments

profile	An object of class entropyProfile.
boundaries	Numeric vector with the first and last genomic positions of the region to be evaluated. To be set interactively if not provided.
chartType	Chart type; either "boxplot", "stripchart" or "swarm".

Details

The query stretch (*e.g.* a protein domain with neutralizing epitopes) is compared against the full set of proteins. Hot spot boundaries should be indicated relative to the reference genome used in variant calling.

Value

htest object. This function is called primarily for its side effects.

See Also

getEntropySignature.

Examples

```
omicron <- getEntropySignature(wWater[wWater$wave == "third", ])

# Entrpy hotspot at SARS-CoV-2 receptor binding domain
assessHotSpot(omicron, c(22517, 23186), chartType = "swarm")
```

`createPositionSummary` *Summarize variants and frequencies at a genome position*

Description

This function is used internally by `getEntropySignature()`. It creates a vector (`aminoAcids`) listing the amino acids observed in a virome at a particular position under analysis, including the reference amino acid, another vector (`frequencies`) with the corresponding frequencies, and returns them combined in a data frame.

Usage

```
createPositionSummary(variants, ref_aa, alt_aa, alt_aa_freq)
```

Arguments

`variants`	A data frame, similar to the `polymorphisms` argument of `getEntropySignature`, but containing information on a single genome position.
`ref_aa`	Name of the column that carries reference amino acids.
`alt_aa`	Name of the column carrying alternative amino acids observed in the metagenome.
`alt_aa_freq`	Name of the column giving the frequencies of alternative amino acids.

Value

A `data frame` describing the variability (different amino acids its frequencies) observed at a specific locus.

See Also

`getEntropySignature`.

createStorage *Build a structure representing amino acid categories.*

Description

The function is used internally by `fillPosition`, which in turn is an auxiliary function of `getEntropySignature`. It creates a list with one element for each amino acid category, named according to the categories used (e.g., "aliphatic", "aromatic", etc.). Each element contains a set of amino acids identified by one-letter codes. The list also includes an element containing an empty numeric vector, whose names correspond to the labels of each category. This vector is to be populated with the frequency of each category at a given genomic position, by the `fillPosition` function.

Usage

```
createStorage(categories)
```

Arguments

`categories`	A character string. Similar to the `categories` parameter of `getEntropySignature`.

Value

A list with one element (`character` vector) for amino acid category and an element (empty named `numeric` vector) to be loaded with the frequency in the metagenome of each amino acid category.

See Also

`getEntropySignature`.

entropyProfile *Create (empty) object of class "entropyProfile"*

Description

This function is intended primarily for internal use by `getEntropySignature`.

Usage

```
entropyProfile(
  polymorphisms,
  position = "position",
  linkage = "linkage",
  ref = "ref",
  alt = "alt",
  protein = "protein",
  aa_position = "aa_position",
```

```
  ref_aa = "ref_aa",
  alt_aa = "alt_aa",
  alt_aa_freq = "alt_aa_freq",
  entropies = NA_real_,
  genome = mn908947.3
)
```

Arguments

polymorphisms	A data frame. Please see Details and Examples in documentation for getEntropySignature.
position	Name of the polymorphisms's column that indicates SNV locations in the genome.
linkage	Information on linked positions.
ref	Column name with reference bases.
alt	Column name with the alternative bases observed in the metagenome.
protein	Name of the column carrying protein names.
aa_position	Name of the column that indicates the protein positions of the mutated amino acids.
ref_aa	Name of the column that carries the reference amino acids.
alt_aa	Name of the column carrying alternative amino acids observed in the metagenome.
alt_aa_freq	Name of the column giving the frequencies of alternative amino acids in the metagenome.
entropies	NA_REAL_ (double numeric/real vector to hold entropy values).
genome	A list providing CDS data and length of the reference genome.

Details

The documentation for getEntropySignature details the type of input needed to create a profile. entropyProfile uses the same parameters as getEntropySignature, with the exception of categories and entropies.

Value

An (empty) object of class entropyProfile.

See Also

getEntropySignature.

`fillPosition` *Translate amino acid frequencies into category frequencies*

Description

The function is used internally by `getEntropySignature`. It creates a storage list by `createStorage`, and loads on it the frequency of each amino acid category based on the data contained in a data frame passed to the function (`positionsSummary` parameter).

Usage

```
fillPosition(positionSummary, categories)
```

Arguments

`positionSummary`	A data frame created by `createPositionSummary`.
`categories`	A character string indicating which category scheme to use. Similar to the `categories` parameter of `getEntropySignature`.

Value

A list with information on the frequencies of each amino acid category observed in a virome in a specific locus. The list contains a `character` vector for each amino acid category, and a named `numeric` vector containing the frequency of each category in the metagenome.

See Also

`getEntropySignature`.

`getEntropySignature` *Infer Entropy Signature*

Description

Calculates genome-wide Shannon entropies from SNV data.

Usage

```
getEntropySignature(
  polymorphisms,
  position = "position",
  linkage = "linkage",
  ref = "ref",
  alt = "alt",
  protein = "protein",
```

```
  aa_position = "aa_position",
  ref_aa = "ref_aa",
  alt_aa = "alt_aa",
  alt_aa_freq = "alt_aa_freq",
  categories = "robust",
  genome = mn908947.3
)
```

Arguments

polymorphisms	A data frame. Please see Details and Examples.
position	Name of the polymorphisms's column that indicates SNV locations in the genome.
linkage	Information on linked positions.
ref	Column name with reference bases.
alt	Column name with the alternative bases observed in the metagenome.
protein	Name of the column carrying protein names.
aa_position	Name of the column that indicates the protein positions of the mutated amino acids.
ref_aa	Name of the column that carries the reference amino acids.
alt_aa	Name of the column carrying alternative amino acids observed in the metagenome.
alt_aa_freq	Name of the column giving the frequencies of alternative amino acids in the metagenome.
categories	Whether a class per amino acid should be used ("sensitive") or they should be grouped into aliphatic, aromatic, polar, positively charged, negatively charged, and special ("robust") (Mirny and Shakhnovich, 1999).
genome	A list providing CDS data and length of the reference genome.

Details

You provide a data frame with SNVs information including reference and alternative aminoacids, their frequencies, and corresponding positions relative to a reference sequence. This type of data can be generated by numerous programs and pipelines. The objective is to assess the biological impact of nonsynonymous variation within a viral population, such as an environmental sample (e.g. wastewater) or a single infection (aka quasisepecies). Entropy is calculated *within* the metagenome and is therefore independent of the reference sequence. Some mutations may be part of a same codon. This is to be indicated in the linkage column, providing a downstream linked position, or the closest upstream position if there are no downstream positions that are part of the same codon. For example, in the wWater dataset, mutations T22673C and C22674T are linked to each other and affect codon 371 of the S gene:

	wave	position	linkage	ref	alt	protein	...
...							
105	third	22599	NA	G	A	S	...
106	third	22673	22674	T	C	S	...
107	third	22674	22673	C	T	S	...

108 third 22679 NA T C S ...
...

The genome parameter is a list that provides data on the topology of protein-coding regions in the genome and its length, used internally primarily for graphical and summary purposes. The package provides an example (mn908947.3) of how this information is to be organized.

Value

An object of class entropyProfile. It contains a tidy, summarized version of the SNV table, a data frame with information on genome-wide entropy, a data frame with information on each CDS and corresponding mutations observed in the virome, and a list with CDS data and length of the reference genome used in variant calling.

References

Mirny and Shakhnovich, 1999. J Mol Biol 291:177-196. doi:10.1006/jmbi.1999.2911.

Shannon, 1948. Bell System Technical Journal, 27:379-423. doi:10.1002/j.15387305.1948.tb01338.x.

Examples

```
# Entropy across the genome in ancestral lineages
ancestral <- getEntropySignature(wWater[wWater$wave == "first", ], categories = "sensitive")

# Inspect profile
plot(ancestral, chartType = "entroScan")
```

---

getPosEntropy *Calculates aminoacidic entropy resulting from SNVs present in a specific locus*

Description

The function is used internally by getEntropySignature. It calculates Shannon entropy from categories and frequencies passed to the function.

Usage

```
getPosEntropy(variantPosition)
```

Arguments

variantPosition
: A list, created and passed to the function by fillPosition. It contains information on the frequencies of each amino acid category observed in a virome, as a result of mutations at a given genomic position.

Value

A numeric value, corresponding to the entropy associated to the SNVs present at a specific locus.

See Also

getEntropySignature.

---

heatmap_entropyProfiles *Graphically compare entropy profiles*

Description

This function prints heatmaps of multiple entropy profiles (entropyProfile objects).

Usage

```
heatmap_entropyProfiles(..., keepInvariant = TRUE, showAllPeptides = TRUE)
```

Arguments

...	Unquoted entropyProfile data objects to be compared.
keepInvariant	Logical; whether to retain positions where entropy is zero. Defaults to FALSE.
showAllPeptides	Logical; whether the heatmap should show all peptides encoded by the virus, regardless of whether they have undergone mutations. Defaults to TRUE.

Value

A ggplot object representing the heatmap.

Examples

```
firstWave <- wWater[ wWater$wave == "first", ]
thirdWave <- wWater[ wWater$wave == "third", ]
# Filter out minor variants whose genotype matches that of the reference genome
firstWave <- firstWave[ firstWave$alt_aa_freq <= 0.97, ]
thirdWave <- thirdWave[ thirdWave$alt_aa_freq <= 0.97, ]
ancestral <- getEntropySignature(firstWave)
omicron <-  getEntropySignature(thirdWave)

# High entropies sustained over time in the spike protein, and signals
# compatible with pervasive negative selection in Omicron sublineages:
heatmap_entropyProfiles(ancestral, omicron)
```

intraHostVariants — *SNVs from multiple organs of individuals with postmortem SARS-CoV-2 detection*

Description

SNVs were inferred from Illumina (2 x 150) sequences. Reads were aligned to the Wuhan-Hu-1 reference genome (MN908947.3) with `BWA`, followed by variant calling using `LoFreq` with source quality correction. The VCF file was post-processed with `BCFtools` to prevent reference and strand biases and remove positions supported by fewer than 1,000 reads.

Usage

```
intraHostVariants
```

Format

A data frame with 213 rows and 16 columns.

Source

Manrique, J. M., Maffia-Bizzozero, S., Delpino, M. V., Quarleri, J., & Jones, L. R. (2024). Multi-Organ Spread and Intra-Host Diversity of SARS-CoV-2 Support Viral Persistence, Adaptation, and a Mechanism That Increases Evolvability. *Journal of Medical Virology*, 96(12), e70107. doi:10.1002/jmv.70107

See Also

See `vignette("intraHostVariants", package = "MetaEntropy")` for a step-by-step tutorial on analyzing this dataset.

mn908947.3 — *CDS topology and length of Wuhan-Hu-1 reference strain*

Description

This type of data can be obtained from `.gff` (General Feature Format) files using applications such as the `rtracklayer` package, or manually from the corresponding entry in the GenBank database.

Usage

```
mn908947.3
```

Format

An object of class `list` of length 2.

Source

Nucleotide [Internet]. Bethesda (MD): National Library of Medicine (US), National Center for Biotechnology Information; [1988] – . Accession No. MN908947.3, Severe acute respiratory syndrome coronavirus 2 isolate Wuhan-Hu-1, complete genome. Available from: https://www.ncbi.nlm.nih.gov/nuccore/MN908947.3.

---

plot.entropyProfile *Plot entropy signatures*

Description

Creates entropy charts along a genome.

Usage

```
## S3 method for class 'entropyProfile'
plot(x, chartType = "bp", ...)
```

Arguments

x	Object of class entropyProfile.
chartType	Whether to graph per-protein summaries ("bp"), per-protein stripcharts ("stripchart" / "swarm"), or position-wise entropy ("entroScan").
...	Additional arguments passed to the function.

Value

Unrendered gg/ggplot object produced by ggplot2. This function is primarily called for its side effects.

Examples

```
ancestral <- getEntropySignature(wWater[wWater$wave == "first", ])
omicron <- getEntropySignature(wWater[wWater$wave == "third", ])

# Enhanced Spike entropy plus pervasive negative selection in Omicron
# sublineages
anc_plot <- plot(ancestral, chartType = "stripchart")
omi_plot <- plot(omicron, chartType = "stripchart")
patchwork::wrap_plots(anc_plot/omi_plot)
```

print.profileSummary *Print method for* profileSummary *objects*

Description

This function formats and prints compact entropy profile summaries (profileSummary objects) created by summary.entropyProfile(), on the console.

Usage

```
## S3 method for class 'profileSummary'
print(x, ...)
```

Arguments

x	An object of class profileSummary created by summary.entropyProfile.
...	Additional arguments passed to the function.

Value

Invisibly returns NULL. This function is used for its side effect.

print.tidyMutations *Print method for* tidyMutations *objects*

Description

This function formats and prints compact mutation summaries (tidyMutations objects) created by showMutations(), on the console.

Usage

```
## S3 method for class 'tidyMutations'
print(x, ...)
```

Arguments

x	An object of class tidyMutations created by showMutations.
...	Additional arguments passed to the function.

Value

Invisibly returns NULL. Called for side effect.

showMutations *Summarize mutations*

Description

Displays SNVs, and corresponding protein mutations, at specific genomic positions.

Usage

```
showMutations(profile, positions)
```

Arguments

profile	An object of class entropyProfile.
positions	A vector with genome positions relative to the reference genome.

Details

The user provides a list of genome positions and the function prints the mutations associated with them. The output format is "ref_res###alt_res / protein:ref_res###alt_res", where ref_res is the residue (eiter nucleotide or aminoacid) in the reference strain, alt_res is the alternative residue in the metagenome, "###" is the position (either nucleotide or aminoacid) where the mutation was observed, and "protein" is the name of the affected protein.

Value

An object of class c("tidyMutations", "data.frame"), containing summary information about user-supplied genomic positions. This information includes the mutations themselves relative to the reference genome, their positions within it, and the corresponding abundances in the virome. Intended to be displayed by print.tidyMutations.

See Also

getEntropySignature.

Examples

```
# High entropy at the RBD in Omicron lineages
omicron <- getEntropySignature(wWater[wWater$wave == "third", ])
plot(omicron, chartType="stripchart")

# Identify the high-entropy positions
omicron$Entropy$position[ omicron$Entropy$entropy > 0.3 ]
#[1] 22882 22898 22917 23013 23040 23048 23055 23063

# Get a descriptive table
showMutations(omicron, c(22882, 22898, 22917, 23013, 23040, 23048, 23055, 23063))
```

summary.entropyProfile *Summarize entropy profile*

Description

Prints a report about an entropy profile (an object of class "entropyProfile").

Usage

```
## S3 method for class 'entropyProfile'
summary(object, ...)
```

Arguments

object	An object of class entropyProfile.
...	Other parameters passed to the function.

Value

An object of class c("profileSummary", "list") summarizing an entropy profile. Intended to be displayed via print.profileSummary.

wWater *Data from first and third COVID-19 waves in Trelew* http://tools.wmflabs.org/geohack/geohack.php?language=es&pagename=Trelew&params=-43.253333333333_N_-65.309444444444_E_type:city

Description

SNVs inferred from Illumina (2 x 150) sequences from pooled ultra-pure virus concentrates representative of the 1st and 3rd COVID-19 waves in Trelew. Reads were mapped against the Wuhan-Wu-1 reference genome (MN908947.3) by *bwa*, and variants were called with *iVar* with a 3% frequency cutoff for minor variants. First wave cases were caused by ancestral strains whereas third wave cases were mainly due to highly human-adapted Omicron sublineages.

Usage

```
wWater
```

Format

An object of class data.frame with 148 rows and 10 columns.

Source

Manrique, Julieta Marina, and Leandro Roberto Jones. 2026. Genomic surveillance of SARS-CoV-2: a resource-efficient wastewater-based workflow. Molecular Biology Reports 53 : 995.

Index

∗ **datasets**
intraHostVariants, 11
mn908947.3, 11
wWater, 15

as.data.frame, *3*
as.data.frame.entropyProfile, 2
assessHotSpot, 3

createPositionSummary, 4, *7*
createStorage, 5

entropyProfile, 5

fillPosition, *5*, 7

getEntropySignature, *3–6*, 7, *7*, *9*, *10*, *14*
getPosEntropy, 9

heatmap_entropyProfiles, 10

intraHostVariants, 11

mn908947.3, *9*, 11

plot.entropyProfile, 12
print.profileSummary, 13
print.tidyMutations, 13

showMutations, *13*, 14
summary.entropyProfile, *13*, 15

wWater, 15