

# Package ‘lineagespot’

May 24, 2024

**Title** Detection of SARS-CoV-2 lineages in wastewater samples using next-generation sequencing

**Version** 1.9.0

**Date** 2021-03-24

**Description** Lineagespot is a framework written in R, and aims to identify SARS-CoV-2 related mutations based on a single (or a list) of variant(s) file(s) (i.e., variant calling format). The method can facilitate the detection of SARS-CoV-2 lineages in wastewater samples using next generation sequencing, and attempts to infer the potential distribution of the SARS-CoV-2 lineages.

**License** MIT + file LICENSE

**Encoding** UTF-8

**LazyData** false

**Roxygen** list(markdown = TRUE)

**RoxygenNote** 7.1.2

**biocViews** VariantDetection, VariantAnnotation, Sequencing

**Imports** VariantAnnotation, MatrixGenerics, SummarizedExperiment, data.table, stringr, httr, utils

**Suggests** BiocStyle, RefManageR, rmarkdown, knitr, testthat (>= 3.0.0)

**URL** <https://github.com/BiodataAnalysisGroup/lineagespot>

**BugReports** <https://github.com/BiodataAnalysisGroup/lineagespot/issues>

**BiocType** Software

**VignetteBuilder** knitr

**Config/testthat/edition** 3

**git\_url** <https://git.bioconductor.org/packages/lineagespot>

**git\_branch** devel

**git\_last\_commit** ae4993e

**git\_last\_commit\_date** 2024-04-30

**Repository** Bioconductor 3.20

**Date/Publication** 2024-05-24

**Author** Nikolaos Pechlivanis [aut, cre]  
 (<<https://orcid.org/0000-0003-2502-612X>>),  
 Maria Tsagiopoulou [aut],  
 Maria Christina Maniou [aut],  
 Anastasis Togkousidis [aut],  
 Evangelia Mouchtaropoulou [aut],  
 Taxiarchis Chassalevris [aut],  
 Serafeim Chaintoutis [aut],  
 Chrysostomos Dovas [aut],  
 Maria Petala [aut],  
 Margaritis Kostoglou [aut],  
 Thodoris Karapantsios [aut],  
 Stamatia Laidou [aut],  
 Elisavet Vlachonikola [aut],  
 Aspasia Orfanou [aut],  
 Styliani-Christina Fragakouli [aut],  
 Sofoklis Keisaris [aut],  
 Anastasia Chatzidimitriou [aut],  
 Agis Papadopoulos [aut],  
 Nikolaos Papaioannou [aut],  
 Anagnostis Argiriou [aut],  
 Fotis E. Psomopoulos [aut]

**Maintainer** Nikolaos Pechlivanis <[inab.bioinformatics@lists.certh.gr](mailto:inab.bioinformatics@lists.certh.gr)>

## Contents

<code>get_lineage_report</code> . . . . .	2
<code>is_gff3</code> . . . . .	3
<code>lineagespot</code> . . . . .	4
<code>lineagespot_hits</code> . . . . .	5
<code>list_input</code> . . . . .	6
<code>list_vcf</code> . . . . .	7
<code>merge_vcf</code> . . . . .	7
<code>uniq_variants</code> . . . . .	8

**Index** **10**

---

<code>get_lineage_report</code>	<i>get_lineage_report</i>
---------------------------------	---------------------------

---

## Description

Retrieve information about lineages' variants via [outbreak.info](https://outbreak.info)'s API

**Usage**

```
get_lineage_report(  
  lineages,  
  base.url = "https://api.outbreak.info/genomics/lineage-mutations?pangolin_lineage=" )
```

**Arguments**

`lineages` a character vector containing the names of the lineages of interest

`base.url` The base API URL used to search for lineage reports Default value is "https://api.outbreak.info/genomics/lineage-mutations?pangolin\_lineage="

**Value**

A list of data table elements of lineage reports

**Examples**

```
get_lineage_report(lineages = c("B.1.1.7", "B.1.617.2"))
```

---

is\_gff3

*is\_gff3*

---

**Description**

Identify whether a file is in GFF3 format.

**Usage**

```
is_gff3(file)
```

**Arguments**

`file` Path to GFF3 file.

**Value**

result; TRUE if the input file is in GFF3 format, FALSE if not.

**Examples**

```
gff3_path <- system.file("extdata", "NC_045512.2_annot.gff3",  
  package = "lineagespot"  
)  
is_gff3(gff3_path)
```

---

lineagespot	<i>lineagespot</i>
-------------	--------------------

---

### Description

Identify SARS-CoV-2 related mutations based on a single (or a list) of variant(s) file(s)

### Usage

```
lineagespot(
  vcf_files = NULL,
  vcf_folder = NULL,
  gff3_path = NULL,
  ref_folder = NULL,
  voc = c("B.1.617.2", "B.1.1.7", "B.1.351", "P.1"),
  AF_threshold = 0.8
)
```

### Arguments

<code>vcf_files</code>	A character vector of paths to VCF files
<code>vcf_folder</code>	A path to a folder containing all VCF files that will be integrated into a single table
<code>gff3_path</code>	Path to GFF3 file containing SARS-CoV-2 gene coordinates.
<code>ref_folder</code>	A path to a folder containing lineage reports
<code>voc</code>	A character vector containing the names of the lineages of interest
<code>AF_threshold</code>	A parameter indicating the AF threshold for identifying variants per sample

### Value

A list of three elements;

- Variants' table; A data table containing all variants that are included in the input VCF files
- Lineage hits; A data table containing identified hits between the input variants and out-break.info's lineage reports
- Lineage report; A data table with computed metrics about the prevalence of the lineage of interest per sample.

### Examples

```
results <- lineagespot(
  vcf_folder = system.file("extdata", "vcf-files",
    package = "lineagespot"
  ),
  gff3_path = system.file("extdata",
    "NC_045512.2_annot.gff3",
```

```

        package = "lineagespot"
      ),
      ref_folder = system.file("extdata", "ref",
        package = "lineagespot"
      )
    )

  head(results$lineage.report)

```

---

lineagespot_hits	<i>lineagespot_hits</i>
------------------	-------------------------

---

### Description

Find overlapping variants with SARS-CoV-2 reference lineages coming from outbreak.info reports

### Usage

```

lineagespot_hits(
  vcf_table = NULL,
  ref_folder = NULL,
  voc = c("B.1.617.2", "B.1.1.7", "B.1.351", "P.1")
)

```

### Arguments

vcf_table	A tab-delimited table containing all variants for all samples. This input is generated by the merge_vcf function.
ref_folder	A path to lineages' reports
voc	A character vector containing the names of the lineages of interest

### Value

A data table containing all identified SARS-CoV-2 variants based on the provided reference files

### Examples

```

variants_table <- merge_vcf(
  vcf_folder = system.file("extdata",
    "vcf-files",
    package = "lineagespot"
  ),
  gff3_path = system.file("extdata",
    "NC_045512.2_annot.gff3",
    package = "lineagespot"
  )
)

```

```
# retrieve lineage reports using outbreak.info's API

# use user-specified references
lineage_hits_table <- lineagespot_hits(
  vcf_table = variants_table,
  ref_folder = system.file("extdata", "ref",
    package = "lineagespot"
  )
)
```

---

list\_input

*list\_input*

---

## Description

Check the validity of input parameters from lineagespot function.

## Usage

```
list_input(vcf_fls = NULL, vcf_folder = NULL, gff3_path = NULL)
```

## Arguments

vcf_fls	A character vector of paths to VCF files.
vcf_folder	A path to a folder containing all VCF files that will be integrated into a single table.
gff3_path	Path to GFF3 file containing SARS-CoV-2 gene coordinates.

## Value

Return a character vector of paths to VCF files.

## Examples

```
vcflist <- list_input(
  vcf_folder = system.file("extdata", "vcf-files",
    package = "lineagespot"
  ),
  gff3_path = system.file("extdata",
    "NC_045512.2_annot.gff3",
    package = "lineagespot"
  )
)
```

---

list_vcf	<i>list_vcf</i>
----------	-----------------

---

**Description**

Identify VCF files from a group of files.

**Usage**

```
list_vcf(vcf_fls = NULL, vcf_folder = NULL, gff3_path = NULL)
```

**Arguments**

vcf_fls	A character vector of paths to VCF files
vcf_folder	A path to a folder containing all VCF files that will be integrated into a single table
gff3_path	Path to GFF3 file containing SARS-CoV-2 gene coordinates.

**Value**

- VCF list; A list where only VCF files are stored.

**Examples**

```
list_vcf_info <- list_vcf(  
  vcf_folder = system.file("extdata", "vcf-files",  
    package = "lineagespot"  
  ),  
  gff3_path = system.file("extdata",  
    "NC_045512.2_annot.gff3",  
    package = "lineagespot"  
  )  
)  
print(list_vcf_info)
```

---

merge_vcf	<i>merge_vcf</i>
-----------	------------------

---

**Description**

Merge Variant Calling Format (VCF) files into a single tab-delimited table

**Usage**

```
merge_vcf(vcf_fls = NULL, vcf_folder = NULL, gff3_path = NULL)
```

**Arguments**

vcf_fls	A list of paths to VCF files
vcf_folder	A path to a folder containing all VCF file that will be integrated into a single table
gff3_path	Path to GFF3 file

**Value**

A data table containing all variants from each sample of the input VCF files

**Examples**

```
merge_vcf(
  vcf_folder = system.file("extdata",
    "vcf-files",
    package = "lineagespot"
  ),
  gff3_path = system.file("extdata",
    "NC_045512.2_annot.gff3",
    package = "lineagespot"
  )
)
```

---

uniqu_variants	<i>uniqu_variants</i>
----------------	-----------------------

---

**Description**

Lineage report for variants overlapping

**Usage**

```
uniqu_variants(hits_table = NULL, AF_threshold = 0.8)
```

**Arguments**

hits_table	A tab-delimited table containing the identified overlaps/hits between the input files and the lineages' reports. This input is generated by the lineagespot_hits function.
AF_threshold	A parameter indicating the AF threshold that is going to be applied in order to identify the presence or not of a variant. This is used to compute the number of variants in a sample and eventually the proportion of a lineage.

**Value**

A data table with metrics assessing the abundance of every lineage in each samples



**Examples**

```
variants_table <- merge_vcf(  
  vcf_folder = system.file("extdata", "vcf-files",  
    package = "lineagespot"  
  ),  
  gff3_path = system.file("extdata",  
    "NC_045512.2_annot.gff3",  
    package = "lineagespot"  
  )  
)  
  
lineage_hits_table <- lineagespot_hits(  
  vcf_table = variants_table,  
  ref_folder = system.file("extdata", "ref",  
    package = "lineagespot")  
)  
  
report <- uniq_variants(hits_table = lineage_hits_table)  
head(report)
```

# Index

`get_lineage_report`, 2

`is_gff3`, 3

`lineagespot`, 4

`lineagespot_hits`, 5

`list_input`, 6

`list_vcf`, 7

`merge_vcf`, 7

`uniq_variants`, 8