Package 'Virusparies'

January 20, 2025

Title Visualize and Process Output from 'VirusHunterGatherer'

Version 1.1.0

Maintainer Ruff Sergej <serijnh@gmail.com>

Description A collection of tools for downstream analysis of 'VirusHunterGatherer' output. Processing of hittables and plotting of results, enabling better interpretation, is made easier with the provided functions.

License GPL (>= 3)

URL https://github.com/SergejRuff/Virusparies, https://github.com/lauberlab/VirusHunterGatherer

BugReports https://github.com/SergejRuff/Virusparies/issues

Encoding UTF-8

RoxygenNote 7.3.2

Imports chromote, cowplot, dplyr, ggplot2, gt, parallel, readr, rlang, scales, stats, stringr, tidyr, tools, utils

Depends R (>= 3.5)

NeedsCompilation no

Author Ruff Sergej [aut, cre] (<https://orcid.org/0009-0000-8264-6347>), Lauber Chris [ctb] (<https://orcid.org/0000-0002-2265-2953>), Chong Li Chuin [ctb] (<https://orcid.org/0000-0002-3574-1365>)

Repository CRAN

Date/Publication 2024-12-14 13:00:06 UTC

Contents

`CombineHittables` *CombineHittables: Combine hittables.*

Description

CombineHittables combines multiple hittables by row-binding them, provided they have the same column names.

Usage

```
CombineHittables(...)
```

Arguments

...	Hittables to be combined.

Value

A single hittable resulting from the row-binding of all input Hittables.

Author(s)

Sergej Ruff

See Also

VirusHunterGatherer is available here: https://github.com/lauberlab/VirusHunterGatherer.

Examples

```
path <- system.file("extdata", "virushunter.tsv", package = "Virusparies")
file <- ImportVirusTable(path)
file2 <- ImportVirusTable(path)  # both files have 180 observations

combined_file <- CombineHittables(file,file2)

print(nrow(combined_file))
```

`ExportVirusDataFrame` *ExportVirusDataFrame: Export processed hittables and summary stats data frames*

Description

Export data frames generated by Virusparies functions.

Usage

```
ExportVirusDataFrame(
  df,
  file_name,
  dir_path = NULL,
  file_type = NULL,
  create_path = FALSE
)
```

Arguments

`df`	A summary statistics or VirusHunterGatherer hittable data frame.
`file_name`	A character string naming the file, optionally including ".tsv" or ".csv" file extensions. If suffix is not provided, file_type will be used to determine the file type.
`dir_path`	A character string indicating the directory path where the file will be saved (default: current corking directory).
`file_type`	A character vector specifying the type of file to export. Can be "csv" or "tsv". If NULL and file_name does not specify a suffix, the function infers the file type based on the prefix in file_name.
`create_path`	Logical indicating whether to create the directory path specified in dir_path if it does not already exist (default: FALSE).

Details

Functions in the Virusparies package can generate both plots and new data frames. Data frames contain either summary statistics for contig length, E-value or identity in percentage or a processed hittable for example outlier or observations below threshold, when running `VhgBoxplot`.

Both types of data frames can be exported via `ExportVirusDataFrame`. Summary stats and hittables can be exported as CSV files or in TSV format, if the user prefers the file type used in VirusHunterGatherer hittables.

Value

A message indicating that export was successful.

Author(s)

Sergej Ruff

See Also

VirusHunterGatherer is available here: https://github.com/lauberlab/VirusHunterGatherer.

Examples

```
path <- system.file("extdata", "virushunter.tsv", package = "Virusparies")
file <- ImportVirusTable(path)


# generate a plot that returns both processed hittables (outlier) and summary stats
plot1 <- VhgBoxplot(file, x_column = "best_query", y_column = "ViralRefSeq_E")




# export hittable as tsv (same format as input hittables)
ExportVirusDataFrame(df=plot1$outlier,file_name="outlier",file_type="tsv",
dir_path=tempdir())

# export summary stats as csv
ExportVirusDataFrame(df=plot1$summary_stats,file_name="summarystats",
file_type="csv",dir_path=tempdir())
```

ExportVirusGt *ExportVirusGt: (Experimental) Export Graphical Tables*

Description

ExportVirusGt allows the user to export graphical tables in different formats.

Usage

```
ExportVirusGt(
  gtable,
  filename = "table.docx",
  export_gt_obj = FALSE,
  path = NULL,
  create.dir = FALSE,
  ...
)
```

Arguments

gtable	A graphical table object.
filename	Name of the output file (default: table.docx"). Make sure to provide an extension compatible with the output types: .html, .tex, .ltx, .rtf, .docx. If a custom save function is provided, the file extension is ignored.
export_gt_obj	(optional): If TRUE, exports the input data frame in .rds format with the same name as specified in filename (default: FALSE).
path	Path of the directory to save plot to: path and filename are combined to create the fully qualified file name (default: current corking directory).
create.dir	Whether to create new directories if a non-existing directory is specified in the filename or path (TRUE) or return an error (FALSE, default). If FALSE and run in an interactive session, a prompt will appear asking to create a new directory when necessary.
...	Pass any other options to the corresponding internal saving function.

Details

Export graphical tables (gt) generated by functions within the Virusparies package.

This feature is in an experimental phase and may not currently function as expected.

The exportVirusGt function utilizes the gt package for table manipulation and formatting.

For HTML output file names with .html or .htm extensions, an HTML document is generated using the gt package. Pass TRUE or FALSE to inline_css to include or exclude CSS styles inline (default is FALSE). Additional options can be passed through .... For RTF output file names with .rtf extension, an RTF file is created. Use the page_numbering option to control page numbering (default is none).

For image files, use .png for PNG and .pdf for PDF. The gt package relies on Google Chrome installation for PNG and PDF images. Pass options to webshot2::webshot() through .... Useful PNG options include zoom (default: 2) and expand (default: 5).

For LaTeX output file names with .tex, .ltx, or .rnw extensions, and .rtf for RTF, the corresponding documents are generated. No additional options available.

For .docx output, requires rmarkdown package.

When `create.dir` is set to TRUE, it generates a directory at the specified 'path' argument if the path doesn't already exist.

The optional `export_gt_obj` argument enables the user to export the data frame as a .rds file alongside the graphical table.

Value

A message indicating that export was successful.

Author(s)

Sergej Ruff

See Also

VirusHunterGatherer is available here: https://github.com/lauberlab/VirusHunterGatherer.

Examples

```
path <- system.file("extdata", "virushunter.tsv", package = "Virusparies")
vh_file <- ImportVirusTable(path)


# using first 10 rows of SRA_run,num_hits,bestquery,ViralRefSeq_E and Identity col.
vh_file_part <- vh_file[c(1:10),c(1,7,9,10,11)]

table <- VhgTabularRasa(vh_file_part,title = "first 10 rows of vh_file",subtitle =
"example for any table",names_ = c("Runs","Number of Contigs","Best Query Result",
"Reference E-Value","Reference Identity"))



ExportVirusGt(gtable=table,filename="vh_parttable.docx",path=tempdir())
```

ExportVirusPlot *Export Virusparies Plots*

Description

ExportVirusPlot allows the user to export plots in different formats.

Usage

```
ExportVirusPlot(
  file_name,
  export_plotobj = FALSE,
  plot = NULL,
  device = NULL,
  path = NULL,
  scale = 1,
  width = NA,
  height = NA,
  units = c("in", "cm", "mm", "px"),
  dpi = 300,
  limitsize = TRUE,
  ...,
  align = "none",
  axis = "none",
  nrow = NULL,
  ncol = NULL,
  rel_widths = 1,
  rel_heights = 1,
  labels = NULL,
  label_size = 14,
  label_fontfamily = NULL,
  label_fontface = "bold",
  label_colour = NULL,
  label_x = 0,
  label_y = 1,
  hjust = -0.5,
  vjust = 1.5,
  scale_grid = 1,
  greedy = TRUE,
  byrow = TRUE
)
```

Arguments

file_name	Name of the output file.
export_plotobj	(optional): If TRUE, exports the plot object in rds format with the same name as specified in file_name (default: FALSE).

plot	The plot to be exported.
device	The device used for output. Can be one of "eps", "ps", "tex", "pdf", "jpeg", "tiff", "png", "bmp", "svg", or "wmf" (windows only). If NULL (default), the device is guessed based on the filename extension.
path	The directory where the plot will be saved. Default is NULL (working directory).
scale	A scaling factor (default: 1).
width	The width of the output file.
height	The height of the output file.
units	The units of the width and height parameters. Can be one of "in", "cm", "mm", or "px".
dpi	The resolution of the output device in dots per inch (default: 300).
limitsize	Whether to limit the size of the output file to the dimensions of the plot (default: TRUE).
...	Additional arguments passed to.
align	Specifies alignment of plots in the grid: "none" (default), "hv" (both directions), "h" (horizontally), or "v" (vertically).
axis	Specifies alignment of plots by margins: "none" (default), or any combo of left ("l"), right ("r"), top ("t"), or bottom ("b")(e.g., "tblr" or "rlbt").
nrow	(optional): Number of rows in the plot grid (default: NULL).
ncol	(optional): Number of columns in the plot grid (default: NULL).
rel_widths	Numeric vector of relative column widths. Default is 1 (equal widths).
rel_heights	Numeric vector of relative row heights. Default is 1 (equal heights).
labels	List of labels for plots. Default is NULL (no labels).labels="auto" and labels="AUTO" auto-generate lower and upper-case labels.
label_size	Numeric label size (default: 14).
label_fontfamily	Font family for labels. Default is NULL (theme default).
label_fontface	Font face for labels (default: "bold").
label_colour	Color for labels. Default is NULL (theme default).
label_x	Single value/vector of x positions for labels,relative to each subplot. Default is 0 (left).
label_y	Single value/vector of y positions for labels, relative to each subplot. Default is 1 (top).
hjust	Horizontal adjustment for labels (default: -0.5).
vjust	Vertical adjustment for labels (default: 1.5).
scale_grid	Single value/vector > 0. Enables you to scale the size of all or select plots.
greedy	Margin adjustment during alignment (default: TRUE).
byrow	Arrange plots by row (TRUE, default) or column (FALSE).

Details

Export plots generated by functions within the Virusparies package.

ExportVirusPlot exports plots in various formats supported by Virusparies. Supported devices include "eps", "ps", "tex", "pdf", "jpeg", "tiff", "png", "bmp", "svg", or "wmf" (Windows only). When 'device' is set to NULL, the file extension in `filename` is used to determine the device.

Depending on the plot, the final image might be cropped or truncated. We recommend experimenting with height, width, and resolution.

In addition, users can generate a grid layout containing multiple plots when a list containing multiple plots is provided as input. This will then be exported using the chosen device.

The following arguments are only used for export with grid layout:

- `align`: Specifies how plots are aligned within the grid.
- `axis`: Controls alignment of plots by margins.
- `nrow`, `ncol`: Define the structure of the plot grid.
- `rel_widths`, `rel_heights`: Adjust relative column and row sizes.
- `labels`, `label_size`, `label_fontfamily`, `label_fontface`, `label_colour`, `label_x`, `label_y`, `hjust`, `vjust`: Customize plot labels.
- `scale_grid`: Enables you to scale the size of all or select plots.
- `greedy`: Determines margin adjustments during alignment.
- `byrow`: Specifies the arrangement of plots in the grid.

`export_plotobj` = TRUE exports the plot in the specified format (e.g., PNG, PDF, etc.) and also saves the plot object in .rds format with the same file name. This allows the user to import the plot object into R using the `readRDS` function and modify the plot further as needed.

Value

a message indicating that export was successful.

Author(s)

Sergej Ruff

See Also

VirusHunterGatherer is available here: https://github.com/lauberlab/VirusHunterGatherer.

Examples

```
path <- system.file("extdata", "virushunter.tsv", package = "Virusparies")
vh_file <- ImportVirusTable(path)

# Basic plot
plot <- VhgIdentityScatterPlot(vh_file,cutoff = 1e-5)
```

```
# first export
ExportVirusPlot(plot=plot$plot,file_name="testplot.png",width=8,height=6,
units="in",path=tempdir())

# second export with device argument
ExportVirusPlot(plot=plot$plot,file_name="testplot",width=8,height=6,
units="in",device = "png",path=tempdir())



## example 2 for multiple plots in 1 pdf file.

path2 <- system.file("extdata", "virusgatherer.tsv", package = "Virusparies")
vg_file <- ImportVirusTable(path2)

# Generate 3 plots
violinplot <- VgConLenViolin(vg_file=vg_file,cut = 1e-5,log10_scale = TRUE,
legend_position = "none",title = "",xlabel = "",reorder_criteria = NULL,
                             theme_choice = "minimal")
srarun <- VhgRunsBarplot(file = vg_file,groupby = "ViralRefSeq_taxonomy",
legend_position = "none",title = "",xlabel = "",reorder_criteria = NULL,
                         theme_choice = "minimal")
boxplot <- VhgBoxplot(vg_file,x_column = "ViralRefSeq_taxonomy",y_column = "ViralRefSeq_ident",
legend_position = "bottom",title = "",xlabel = "",
                      reorder_criteria = NULL,theme_choice = "minimal")

# add plots to a list
plot_list <- list(violinplot$plot,srarun$plot,boxplot$boxp)



ExportVirusPlot(plot=plot_list,file_name="grid_testplot.pdf",width=16,height=12,
units="in",nrow = 3,ncol = 1,path=tempdir())
```

`get_plot_parameters` *Internal function to set default title and cutoff in boxplot function*

Description

Internal function to set default title and cutoff in boxplot function

Usage

```
get_plot_parameters(y_column, cut)
```

Arguments

y_column	y-column
cut	cutoff value

Value

list containing cutoff and title

`ImportVirusTable` *ImportVirusTable: Import VirusHunterGatherer hittables into R*

Description

ImportVirusTable imports VirusHunterGatherer hittables into R.

Usage

```
ImportVirusTable(path)
```

Arguments

path	A character string specifying the path to the VirusHunter or VirusGatherer hittable file.

Details

ImportVirusTable reads the VirusHunter or VirusGatherer hittable file specified by 'path' into a data frame in R. The file should be in tab-separated values (TSV) format. The first row should contain column headers.

Value

A data frame containing the data from the VirusHunter or VirusGatherer hittable file.

Author(s)

Sergej Ruff

See Also

VirusHunterGatherer is available here: https://github.com/lauberlab/VirusHunterGatherer.

Examples

```
path <- system.file("extdata", "virushunter.tsv", package = "Virusparies")
vh_file <- ImportVirusTable(path)

print(head(vh_file))

# import gatherer files
path2 <- system.file("extdata", "virusgatherer.tsv", package = "Virusparies")
vg_file <- ImportVirusTable(path2)

print(head(vg_file))
```

`SummarizeViralStats` *SummarizeViralStats: Generate summary stats outside of plot functions*

Description

Summarizes data by grouping it according to a specified metric (contig length, E-value or Identity). SummarizeViralStats generates a summary table that includes counts of observations based on a specified metric cutoff. It computes relevant summary statistics depending on the selected metric, with options to filter rows based on a cutoff value.

Usage

```
SummarizeViralStats(
  file,
  groupby = "best_query",
  metric,
  metric_cutoff,
  filter_cutoff = NULL,
  show_total = FALSE,
  extra_stats = NULL,
  sort_by = NULL,
  top_n = NULL,
  group_unwanted_phyla = NULL
)
```

Arguments

`file`	VirusHunterGatherer hittable.
`groupby`	(optional): A character specifying the column containing the groups (default: "best_query"). Note: Gatherer hittables do not have a "best_query" column. Please provide an appropriate column for grouping.
`metric`	A character string specifying the name of the metric column to be used for calculations. This column must be present in `file`. Supported metric columns include:

- "contig_len"
- "ViralRefSeq_E"
- "ViralRefSeq_ident"

`metric_cutoff` A numeric value used to classify the metric into two categories: below cutoff and above or equal to cutoff.

`filter_cutoff` A numeric value for optional filtering of the data based on E-value. Rows where the specified filtering column has a value less than this cutoff are retained. If `NULL`, no filtering is applied. Default is `NULL`.

`show_total` A logical value indicating whether to include a row with the total sums for each column in the summary table. Default is `FALSE`.

`extra_stats` A character vector specifying additional summary statistics to include in the output. Options include:

- `"mean"`
- `"median"`
- `"Q1"`
- `"Q3"`
- `"sd"`
- `"min"`
- `"max"`

If `NULL` (the default), only the basic counts are included.

`sort_by` (optional): A character string specifying the column name by which to sort the results. Supported values include:

- "less_than_X": The count of observations below the specified `metric_cutoff` (X is replaced by the cutoff value).
- "equal_or_more_than_X": The count of observations greater than or equal to the specified `metric_cutoff` (X is replaced by the cutoff value).
- "total": The total count of observations in each group.
- "mean": The mean value of the specified metric in each group (if `"mean"` is included in `extra_stats`).
- "median": The median value of the specified metric in each group (if `"median"` is included in `extra_stats`).
- "Q1": The first quartile (25th percentile) of the specified metric in each group (if `"Q1"` is included in `extra_stats`).
- "Q3": The third quartile (75th percentile) of the specified metric in each group (if `"Q3"` is included in `extra_stats`).
- "sd": The standard deviation of the specified metric in each group (if `"sd"` is included in `extra_stats`).
- "min": The minimum value of the specified metric in each group (if `"min"` is included in `extra_stats`).
- "max": The maximum value of the specified metric in each group (if `"max"` is included in `extra_stats`).

If `NULL` (the default), no sorting is applied.

`top_n` (optional): A numeric value indicating the number of top rows to return based on the selected metric. If `NULL` (the default), all rows are returned.

`group_unwanted_phyla`
(optional): A character string specifying which group of viral phyla to retain in the analysis. Valid values are:

"rna" Retain only the phyla specified for RNA viruses.

"smalldna" Retain only the phyla specified for small DNA viruses.

"largedna" Retain only the phyla specified for large DNA viruses.

"others" Retain only the phyla that match small DNA, Large DNA and RNA viruses.

All other phyla not in the specified group will be grouped into a single category: "Non-RNA-virus" for `"rna"`, "Non-Small-DNA-Virus" for `"smalldna"`,"Non-Large-DNA-Virus" for `"largedna"`,or "Other Viruses" for `"others"`.

Value

A data frame summarizing the viral stats. The output includes:

- The count of observations below and above or equal to the `metric_cutoff`.
- Optional additional summary statistics as specified by `extra_stats`.
- An optional total row if `show_total` is `TRUE`.

Author(s)

Sergej Ruff

See Also

VirusHunterGatherer is available here: https://github.com/lauberlab/VirusHunterGatherer.

Examples

```
path <- system.file("extdata", "virushunter.tsv", package = "Virusparies")
file <- ImportVirusTable(path)

stats <- SummarizeViralStats(file=file,
groupby = "best_query",
metric = "ViralRefSeq_ident",
metric_cutoff = 90,
show_total = TRUE,
filter_cutoff = 1e-5,
extra_stats = c("median","Q1","Q3"))

print(stats)
```

VgConLenViolin *VgConLenViolin: Generate a Violinplot of contig length for each group (Gatherer only)*

Description

VgConLenViolin creates a violin plot to visualize the distribution of contig lengths for each group in VirusGatherer hittables.

Usage

```
VgConLenViolin(
  vg_file = vg_file,
  taxa_rank = "Family",
  cut = 1e-05,
  log10_scale = TRUE,
  reorder_criteria = "median",
  adjust_bw = 1,
  jitter_point = FALSE,
  theme_choice = "linedraw",
  flip_coords = TRUE,
  title = "Violinplot of contig length for each group",
  title_size = 16,
  title_face = "bold",
  title_colour = "#2a475e",
  subtitle = NULL,
  subtitle_size = 12,
  subtitle_face = "bold",
  subtitle_colour = "#1b2838",
  xlabel = "Viral group",
  ylabel = "Contig length (nt)",
  axis_title_size = 12,
  xtext_size = 10,
  x_angle = NULL,
  ytext_size = 10,
  y_angle = NULL,
  remove_group_labels = FALSE,
  legend_title = "Phylum",
  legend_position = "bottom",
  legend_title_size = 12,
  legend_title_face = "bold",
  legend_text_size = 10,
  min_observations = 1,
  facet_ncol = NULL,
  add_boxplot = FALSE,
  group_unwanted_phyla = NULL
)
```

Arguments

vg_file	A data frame containing VirusGatherer hittable results.
taxa_rank	(optional): Specify the taxonomic rank to group your data by. Supported ranks are: • "Subphylum" • "Class" • "Subclass" • "Order" • "Suborder" • "Family" (default) • "Subfamily" • "Genus" (including Subgenus)
cut	(optional): A numeric value representing the cutoff for the refseq E-value (default: 1e-5).
log10_scale	(optinal): transform y-axis to log10 scale (default: TRUE).
reorder_criteria	Character string specifying the criteria for reordering the x-axis ('max', 'min', 'median'(Default),'mean','phylum'). NULL sorts alphabetically. You can also specify criteria with 'phylum_' prefix (e.g., 'phylum_median') to sort by phylum first and then by the specified statistic within each phylum.
adjust_bw	(optional): control the bandwidth of the kernel density estimator used to create the violin plot. A higher value results in a smoother plot by increasing the bandwidth, while a lower value can make the plot more detailed but potentially noisier (default: 1).
jitter_point	(optional): logical: TRUE to show all observations, FALSE to show only groups with less than 2 observations (default: FALSE).
theme_choice	(optional): A character indicating the ggplot2 theme to apply. Options include "minimal", "classic", "light", "dark", "void", "grey" (or "gray"), "bw", "linedraw" (default), and "test". Append "_dotted" to any theme to add custom dotted grid lines (e.g., "classic_dotted").
flip_coords	(optional): Logical indicating whether to flip the coordinates of the plot (default: TRUE).
title	(optional): The title of the plot (default: "Violinplot of contig length for each group").
title_size	(optional): The size of the title text (default: 16).
title_face	(optional): The face (bold, italic, etc.) of the title text (default: "bold").
title_colour	(optional): The color of the title text (default: "#2a475e").
subtitle	(optional): A character specifying the subtitle of the plot (default: NULL).
subtitle_size	(optional): Numeric specifying the size of the subtitle text(default: 12).
subtitle_face	(optional): A character specifying the font face for the subtitle text (default: "bold").

`subtitle_colour`
: (optional): A character specifying the color for the subtitle text (default: "#1b2838").

`xlabel`
: (optional): The label for the x-axis (default: "Viral group").

`ylabel`
: (optional): The label for the y-axis (default: "Contig length (nt)").

`axis_title_size`
: (optional): The size of the axis titles (default: 12).

`xtext_size`
: (optional): The size of the x-axis text (default: 10).

`x_angle`
: (optional): An integer specifying the angle (in degrees) for the x-axis text labels. Default is NULL, meaning no change.

`ytext_size`
: (optional): The size of the y-axis text (default: 10).

`y_angle`
: (optional): An integer specifying the angle (in degrees) for the y-axis text labels. Default is NULL, meaning no change.

`remove_group_labels`
: (optional): If `TRUE`, the group labels will be removed; if `FALSE` or omitted, the labels will be displayed.

`legend_title`
: (optional): A character specifying the title for the legend (default: "Phylum").

`legend_position`
: (optional): The position of the legend (default: "bottom).

`legend_title_size`
: (optional): Numeric specifying the size of the legend title text (default: 12).

`legend_title_face`
: (optional): A character specifying the font face for the legend title text (default: "bold").

`legend_text_size`
: (optional): Numeric specifying the size of the legend text (default: 10).

`min_observations`
: (optional): Minimum number of observations required per group to be included in the plot (default: 1).

`facet_ncol`
: (optional): The number of columns for faceting (default: NULL). It is recommended to specify this when the number of viral groups is high, to ensure they fit well in one plot.

`add_boxplot`
: (optional): Add a boxplot to the violin plot (default: FALSE).

`group_unwanted_phyla`
: (optional): A character string specifying which group of viral phyla to retain in the analysis. Valid values are:

  "rna" Retain only the phyla specified for RNA viruses.

  "smalldna" Retain only the phyla specified for small DNA viruses.

  "largedna" Retain only the phyla specified for large DNA viruses.

  "others" Retain only the phyla that match small DNA, Large DNA and RNA viruses.

  All other phyla not in the specified group will be grouped into a single category: "Non-RNA-virus" for `"rna"`, "Non-Small-DNA-Virus" for `"smalldna"`,"Non-Large-DNA-Virus" for `"largedna"`,or "Other Viruses" for `"others"`.

Details

VgConLenViolin creates a violin plot to visualize the distribution of contig lengths for each group in the "ViralRefSeq_taxonomy" column of the VirusGatherer hittable. The x-axis represents the groups as defined by the "ViralRefSeq_taxonomy" column, and the y-axis represents the contig lengths.

By default, the y-axis is transformed to a log10 scale to better visualize differences in contig lengths across groups. This transformation can be disabled by setting the `log10_scale` argument to FALSE.

`min_observations` filters the data sets to include only groups with at least the specified number of observations before plotting them. This feature allows users to exclude groups with insufficient data. By default, every group is plotted, as the minimum requirement is set to at least one observation per group.

Value

A list containing the following components:

- plot: A plot object representing the violin plot.
- contiglen_stats: A tibble data frame with summary statistics for "contig_len" values.

Author(s)

Sergej Ruff

See Also

VirusHunterGatherer is available here: https://github.com/lauberlab/VirusHunterGatherer.

Examples

```
# import gatherer files
path2 <- system.file("extdata", "virusgatherer.tsv", package = "Virusparies")
vg_file <- ImportVirusTable(path2)

# create a violinplot.
violinplot <- VgConLenViolin(vg_file=vg_file,cut = 1e-5,log10_scale = TRUE)

violinplot
```

VhgAddPhylum *VhgAddPhylum: extract Phylum information*

Description

VhgAddPhylum adds a `Phylum` column to the provided VirusHunter or VirusGatherer hittable, with each entry in the column reflecting the phylum name derived from the `groupby` column for each observation.

Usage

```
VhgAddPhylum(file, groupby = "best_query")
```

Arguments

`file`	A data frame containing VirusHunter or VirusGatherer hittable results.
`groupby`	(optional): A character specifying the column containing the groups (default: "best_query"). Note: Gatherer hittables do not have a "best_query" column. Please provide an appropriate column for grouping.

Value

Hittable with Phylum column

Author(s)

Sergej Ruff

See Also

VirusHunterGatherer is available here: https://github.com/lauberlab/VirusHunterGatherer.

Examples

```
path <- system.file("extdata", "virushunter.tsv", package = "Virusparies")
vh_file <- ImportVirusTable(path)

vh_file_filtered <- VhgPreprocessTaxa(vh_file,"Family")

processed_taxa <- VhgAddPhylum(vh_file_filtered,"ViralRefSeq_taxonomy")

print(unique(processed_taxa$Phylum))
```

VhgBoxplot *VhgBoxplot: Generate box plots comparing E-values,identity or contig length (Gatherer only) for each virus group*

Description

VhgBoxplot generates box plots comparing either E-values,identity or contig length (Gatherer only) for each group from VirusHunter or VirusGatherer hittable results.

Usage

```
VhgBoxplot(
  file,
  x_column = "best_query",
  taxa_rank = "Family",
  y_column = "ViralRefSeq_E",
  contiglen_log10_scale = FALSE,
  cut = 1e-05,
  add_cutoff_line = TRUE,
  cut_colour = "#990000",
  reorder_criteria = "median",
  theme_choice = "linedraw",
  flip_coords = TRUE,
  add_mean_point = FALSE,
  mean_color = "white",
  mean_point_size = 2,
  title = "default",
  title_size = 16,
  title_face = "bold",
  title_colour = "#2a475e",
  subtitle = "default",
  subtitle_size = 12,
  subtitle_face = "bold",
  subtitle_colour = "#1b2838",
  xlabel = NULL,
  ylabel = NULL,
  axis_title_size = 12,
  xtext_size = 10,
  x_angle = NULL,
  ytext_size = 10,
  y_angle = NULL,
  remove_group_labels = FALSE,
  legend_title = "Phylum",
  legend_position = "bottom",
  legend_title_size = 12,
  legend_title_face = "bold",
  legend_text_size = 10,
```

```
  facet_ncol = NULL,
  group_unwanted_phyla = NULL
)
```

Arguments

file	A data frame containing VirusHunter or VirusGatherer hittable results.
x_column	(optional): A character specifying the column containing the groups (default:"best_query"). Note: Gatherer hittables do not have a "best_query" column. Please provide an appropriate column for grouping.
taxa_rank	(optional): When x_column is set to "ViralRefSeq_taxonomy", specify the taxonomic rank to group your data by. Supported ranks are: • "Subphylum" • "Class" • "Subclass" • "Order" • "Suborder" • "Family" (default) • "Subfamily" • "Genus" (including Subgenus)
y_column	A character specifying the column containing the values to be compared. Currently "ViralRefSeq_ident", "contig_len" (column in Gatherer hittable) and "ViralRefSeq_E" are supported columns (default:"ViralRefSeq_E").
contiglen_log10_scale	(optional): When y_column is set to "contig_len", this parameter enables logarithmic scaling (log10) of the y-axis (TRUE). By default, this feature is disabled (FALSE).
cut	(optional): The significance cutoff value for E-values (default: 1e-5).
add_cutoff_line	(optional): Whether to add a horizontal line based on cut for "ViralRefSeq_E" column (default: TRUE).
cut_colour	(optional): The color for the significance cutoff line (default: "#990000").
reorder_criteria	Character string specifying the criteria for reordering the x-axis ('max', 'min', 'median'(Default),'mean','phylum'). NULL sorts alphabetically. You can also specify criteria with 'phylum_' prefix (e.g., 'phylum_median') to sort by phylum first and then by the specified statistic within each phylum.
theme_choice	(optional): A character indicating the ggplot2 theme to apply. Options include "minimal", "classic", "light", "dark", "void", "grey" (or "gray"), "bw", "linedraw" (default), and "test". Append "_dotted" to any theme to add custom dotted grid lines (e.g., "classic_dotted").
flip_coords	(optional): Logical indicating whether to flip the coordinates of the plot (default: TRUE).
add_mean_point	(optional): Logical indicating whether to add mean points to the box plot (default: FALSE).

`mean_color` (optional): Change color of point indicating mean value in box plot (default: "white").

`mean_point_size` (optional): Change size of point indicating mean value in box plot (default: 2).

`title` (optional): A character specifying the title of the plot. Default title is set based on y_column.

`title_size` (optional): Numeric specifying the size of the title text (default: 16).

`title_face` (optional): A character specifying the font face for the title text (default: "bold").

`title_colour` (optional): A character specifying the color for the title text (default: "#2a475e").

`subtitle` (optional): A character specifying the subtitle of the plot. Default subtitle is set based on y_column.

`subtitle_size` (optional): Numeric specifying the size of the subtitle text(default: 12).

`subtitle_face` (optional): A character specifying the font face for the subtitle text (default: "bold").

`subtitle_colour` (optional): A character specifying the color for the subtitle text (default: "#1b2838").

`xlabel` (optional): A character specifying the label for the x-axis (default: "Virus found in query").

`ylabel` (optional): A character specifying the label for the y-axis. Default is set based on y_column.

`axis_title_size` (optional): Numeric specifying the size of the axis title text (default: 12).

`xtext_size` (optional): Numeric specifying the size of the x-axis tick labels (default: 10).

`x_angle` (optional): An integer specifying the angle (in degrees) for the x-axis text labels. Default is NULL, meaning no change.

`ytext_size` (optional): Numeric specifying the size of the y-axis tick labels (default: 10).

`y_angle` (optional): An integer specifying the angle (in degrees) for the y-axis text labels. Default is NULL, meaning no change.

`remove_group_labels` (optional): If `TRUE`, the group labels will be removed; if `FALSE` or omitted, the labels will be displayed.

`legend_title` (optional): A character specifying the title for the legend (default: "Phylum").

`legend_position` (optional): A character specifying the position of the legend (default: "bottom").

`legend_title_size` (optional): Numeric specifying the size of the legend title text (default: 12).

`legend_title_face` (optional): A character specifying the font face for the legend title text (default: "bold").

`legend_text_size` (optional): Numeric specifying the size of the legend text (default: 10).

`facet_ncol` (optional): The number of columns for faceting (default: NULL). It is recommended to specify this when the number of viral groups is high, to ensure they fit well in one plot.

`group_unwanted_phyla`
(optional): A character string specifying which group of viral phyla to retain in the analysis. Valid values are:

"rna" Retain only the phyla specified for RNA viruses.

"smalldna" Retain only the phyla specified for small DNA viruses.

"largedna" Retain only the phyla specified for large DNA viruses.

"others" Retain only the phyla that match small DNA, Large DNA and RNA viruses.

All other phyla not in the specified group will be grouped into a single category: "Non-RNA-virus" for `"rna"`, "Non-Small-DNA-Virus" for `"smalldna"`,"Non-Large-DNA-Virus" for `"largedna"`,or "Other Viruses" for `"others"`.

Details

VhgBoxplot generates box plots comparing either E-values, identity, or contig length (Gatherer only) for each virus group from the VirusHunter or Gatherer hittable.

The user can specify whether to generate box plots for E-values, identity, or contig length (Gatherer only) by specifying the 'y_column'. This means that 'VhgBoxplot' can generate three different types of box plots. By default, 'y_column' is set to "ViralRefSeq_E" and will plot the reference E-Value on the y-axis. Grouping on the x-axis is done by the 'x_column' argument. By default, the "best_query" will be used.

Additionally, the function calculates summary statistics and identifies outliers for further analysis ("ViralRefSeq_E" and "contig_len" only). When 'y_column' is set to "ViralRefSeq_E", the output also includes 'rows_belowthres', which contains the hittable filtered for the rows below the threshold specified in the 'cut' argument.

The 'cut' argument is used differently depending on the 'y_column' value:

- For 'y_column' set to "contig_len" or "ViralRefSeq_ident", the 'cut' argument filters the data to plot only the values with a "ViralRefSeq_E" below the specified threshold (default: 1e-5).
- For 'y_column' set to "ViralRefSeq_E", the rows are not filtered. Instead, a horizontal line (h_line) is shown in the plot to indicate the cutoff value.

This allows the user to plot only the significant contig lengths and identities while also visualizing the number of non-significant and significant values for comparison.

Warning: In some cases, E-values might be exactly 0. When these values are transformed using -log10, R returns "inf" as the output. To avoid this issue, we replace all E-values that are 0 with the smallest e-value that is greater than 0. If the smallest E-value is above the user-defined cutoff, we use a value of `cutoff * 10^-10` to replace the zeros.

Value

A list containing:

- The generated box plot.

- Summary statistics.
- Outliers ("ViralRefSeq_E" and "contig_len" only).
- rows_belowthres ("ViralRefSeq_E" only).

Author(s)

Sergej Ruff

See Also

VirusHunterGatherer is available here: https://github.com/lauberlab/VirusHunterGatherer.

Examples

```
path <- system.file("extdata", "virushunter.tsv", package = "Virusparies")
file <- ImportVirusTable(path)

# plot 1 for E-values
plot1 <- VhgBoxplot(file, x_column = "best_query", y_column = "ViralRefSeq_E")
plot1

# plot 2 for identity
plot2 <- VhgBoxplot(file, x_column = "best_query", y_column = "ViralRefSeq_ident")
plot2

# plot 3 custom arguments used
plot3 <- VhgBoxplot(file,
                  x_column = "best_query",
                  y_column = "ViralRefSeq_E",
                  theme_choice = "grey",
                  subtitle = "Custom subtitle: Identity for custom query",
                  xlabel = "Custom x-axis label: Custom query",
                 ylabel = "Custom y-axis label: Viral Reference Evalue in -log10 scale",
                  legend_position = "right")
plot3

# import gatherer files
path2 <- system.file("extdata", "virusgatherer.tsv", package = "Virusparies")
vg_file <- ImportVirusTable(path2)


# plot 4: Virusgatherer plot for ViralRefSeq_taxonomy agains contig length
plot5 <- VhgBoxplot(vg_file,x_column = "ViralRefSeq_taxonomy",y_column = "contig_len")
plot5
```

VhgGetSubject *VhgGetSubject: Process and Count Viral Subjects within Groups*

Description

VhgGetSubject: Process and Count Viral Subjects within Groups

Usage

```
VhgGetSubject(
  file,
  groupby = "best_query",
  remove_identifiers = TRUE,
  include_run_ids = FALSE,
  extract_brackets = FALSE,
  group_unwanted_phyla = NULL
)
```

Arguments

`file`	A data frame containing VirusHunter or VirusGatherer hittable results.
`groupby`	(optional): A character specifying the column containing the groups (default: "best_query"). Note: Gatherer hittables do not have a "best_query" column. Please provide an appropriate column for grouping.
`remove_identifiers`	(optional): if `TRUE` (default), removes the identifiers in the ViralRefSeq_subject cells.
`include_run_ids`	(optional): If `TRUE` (default is `TRUE`), adds a fourth column named `run_ids` to the output. This column contains a comma-separated list of unique identifiers from either the `SRA_run` or `run_id` column, aggregated for each combination of group and subject.
`extract_brackets`	(optional): extract content within square brackets [].
`group_unwanted_phyla`	(optional): A character string specifying which group of viral phyla to retain in the analysis. Valid values are: **"rna"** Retain only the phyla specified for RNA viruses. **"smalldna"** Retain only the phyla specified for small DNA viruses. **"largedna"** Retain only the phyla specified for large DNA viruses. **"others"** Retain only the phyla that match small DNA, Large DNA and RNA viruses. All other phyla not in the specified group will be grouped into a single category: "Non-RNA-virus" for `"rna"`, "Non-Small-DNA-Virus" for `"smalldna"`,"Non-Large-DNA-Virus" for `"largedna"`,or "Other Viruses" for `"others"`.

Details

The function VhgGetSubject counts the number of viral subjects in the ViralRefSeq_subject column for each group specified by the groupby argument. It returns a tibble with three columns: the first column contains the viral group specified by the groupby argument, the second column lists the viral subjects found in that group, and the third column shows how many times each viral subject appears in that group.

Value

a processed tibble object.

Author(s)

Sergej Ruff

See Also

VirusHunterGatherer is available here: https://github.com/lauberlab/VirusHunterGatherer.

Examples

```
# import data
path <- system.file("extdata", "virushunter.tsv", package = "Virusparies")
file <- ImportVirusTable(path)

# process column and filter for significant groups
file <- VhgPreprocessTaxa(file,taxa_rank = "Family")
file_filtered <- VhgSubsetHittable(file,ViralRefSeq_E_criteria = 1e-5)

subject_df <- VhgGetSubject(file_filtered,groupby = "ViralRefSeq_taxonomy",
remove_identifiers = TRUE)

print(subject_df)
```

VhgIdenFacetedScatterPlot

VhgIdenFacetedScatterPlot: Create a scatter plot of Viral refseq identity vs. -log10 of viral refseq E-value.

Description

VhgIdenFacetedScatterPlot generates a scatter plot of viral refseq identity versus -log10 of refseq E-value for each virus group in the best_query or ViralRefSeq_taxonomy column . The points are colored based on whether the E-value meets a specified cutoff and are faceted by the viral groups in the best_query or ViralRefSeq_taxonomy column.

Usage

```
VhgIdenFacetedScatterPlot(
  file,
  groupby = "best_query",
  taxa_rank = "Family",
  cutoff = 1e-05,
  conlen_bubble_plot = FALSE,
  contiglen_breaks = 5,
  theme_choice = "linedraw",
  title = "Faceted scatterplot of viral reference E-values and sequence identity",
  title_size = 16,
  title_face = "bold",
  title_colour = "#2a475e",
  subtitle = NULL,
  subtitle_size = 12,
  subtitle_face = "bold",
  subtitle_colour = "#1b2838",
  xlabel = "Viral reference sequence identity (%)",
  ylabel = "-log10 of viral reference E-values",
  axis_title_size = 12,
  xtext_size = 10,
  x_angle = NULL,
  ytext_size = 10,
  y_angle = NULL,
  legend_position = "bottom",
  legend_title_size = 12,
  legend_title_face = "bold",
  legend_text_size = 10,
  true_colour = "blue",
  false_colour = "red",
  wrap_ncol = 2,
  filter_group_criteria = NULL
)
```

Arguments

file	VirusHunterGatherer hittable.
groupby	(optional): A character specifying the column containing the groups (default: "best_query"). Note: Gatherer hittables do not have a "best_query" column. Please provide an appropriate column for grouping.
taxa_rank	(optional): When groupby is set to "ViralRefSeq_taxonomy", specify the taxonomic rank to group your data by. Supported ranks are: • "Subphylum" • "Class" • "Subclass" • "Order" • "Suborder"

- "Family" (default)
- "Subfamily"
- "Genus" (including Subgenus)

`cutoff` (optional): A numeric value representing the cutoff for the refseq E-value. Points with `ViralRefSeq_E` less than or equal to this value will be colored blue; otherwise, they will be colored red (default: 1e-5).

`conlen_bubble_plot` (optional): Logical value indicating whether the `contig_len` column should be used to size the bubbles in the plot. Applicable only to VirusGatherer hittables input (default: FALSE).

`contiglen_breaks` (optional): Number of breaks (default: 5) for the bubble plot (for `conlen_bubble_plot=TRUE`).

`theme_choice` (optional): A character indicating the ggplot2 theme to apply. Options include "minimal", "classic", "light", "dark", "void", "grey" (or "gray"), "bw", "linedraw" (default), and "test". Append "_dotted" to any theme to add custom dotted grid lines (e.g., "classic_dotted").

`title` (optional): The title of the plot (default: "Faceted scatter plot of viral reference E-values and sequence identity").

`title_size` (optional): The size of the title text (default: 16).

`title_face` (optional): The face (bold, italic, etc.) of the title text (default: "bold").

`title_colour` (optional): The color of the title text (default: "#2a475e").

`subtitle` (optional): The subtitle of the plot (default: NULL).

`subtitle_size` (optional): The size of the subtitle text (default: 12).

`subtitle_face` (optional): The face (bold, italic, etc.) of the subtitle text (default: "bold").

`subtitle_colour` (optional): The color of the subtitle text (default: "#1b2838").

`xlabel` (optional): The label for the x-axis (default: "Viral reference sequence identity (%)").

`ylabel` (optional): The label for the y-axis (default: "-log10 of viral reference E-values").

`axis_title_size` (optional): The size of the axis titles (default: 12).

`xtext_size` (optional): The size of the x-axis text (default: 10).

`x_angle` (optional): An integer specifying the angle (in degrees) for the x-axis text labels. Default is NULL, meaning no change.

`ytext_size` (optional): The size of the y-axis text (default: 10).

`y_angle` (optional): An integer specifying the angle (in degrees) for the y-axis text labels. Default is NULL, meaning no change.

`legend_position` (optional): The position of the legend (default: "bottom).

`legend_title_size` (optional): The size of the legend title text (default: 12).

`legend_title_face` (optional): The face (bold, italic, etc.) of the legend title text (default: "bold").

`legend_text_size`
: (optional): The size of the legend text (default: 10).

`true_colour`
: (optional): The color for points that meet the cutoff condition (default: "blue").

`false_colour`
: (optional): The color for points that do not meet the cutoff condition (default: "red").

`wrap_ncol`
: (optional): The number of columns for faceting (default: 12).

`filter_group_criteria`
: (optional): Character vector, numeric vector, or single character/numeric value.
  - Character vector: Names of viral groups to filter.
  - Numeric vector: Indices of viral groups to filter.
  - Single character or numeric value: Filter a single viral group.
  - NULL: No filtering is performed (default).

Details

'VhgIdenFacetedScatterPlot' takes a VirusHunter or VirusGatherer hittable and a cutoff value as inputs. The plot includes:

- Points colored based on whether they meet the cutoff condition.
- Faceting by the `best_query` column as the default column. The user can provide their own column for grouping.
- The option `conlen_bubble_plot = TRUE` generates a bubble plot where the size of points corresponds to "contig_len" (exclusive to VirusGatherer).

`filter_group_criteria`: Allows filtering of viral groups by specifying either a single character string or a vector of character strings that match unique entries in `groupby`. Alternatively, a single numeric value, a range, or a vector of numeric values can be used to filter groups.

For example, if `groupby` is "best_query" with the following unique groups:

- Anello_ORF1core
- Gemini_Rep
- Genomo_Rep
- Hepadna-Nackedna_TP

Setting `filter_group_criteria` to `c("Anello_ORF1core", "Genomo_Rep")` will filter the data to only include observations where the "best_query" column has 'Anello_ORF1core' or 'Genomo_Rep'. Alternatively, setting `filter_group_criteria` to `2:3` will return only the second and third alphabetically ordered viral groups from "best_query". The order also matches the order of the viral groups in the faceted scatter plot.

This is particularly useful when there are too many viral groups to be plotted in a single plot, allowing for separation into different groups. It also enables the user to focus on specific groups of interest for more detailed analysis.

Tibble data frames containing summary statistics (median, Q1, Q3, mean, sd, min, and max) for 'ViralRefSeq_E' and 'ViralRefSeq_ident' values are generated. Optionally, summary statistics for 'contig_len' values are also included if applicable. These summary statistics, along with the plot object, are returned within a list object.

Warning: In some cases, E-values might be exactly 0. When these values are transformed using -log10, R returns "inf" as the output. To avoid this issue, we replace all E-values that are 0 with the smallest E-value that is greater than 0. If the smallest E-value is above the user-defined cutoff, we use a value of `cutoff * 10^-10` to replace the zeros.

Value

A list containing the following components:

- Plot: A plot object representing the faceted scatter plot.
- evalue_stats: A tibble data frame with summary statistics for "ViralRefSeq_E" values.
- identity_stats: A tibble data frame with summary statistics for "ViralRefSeq_ident" values.
- contig_stats (optional): A tibble data frame with summary statistics for "contig_len" values, included only if VirusGatherer is used with `conlen_bubble_plot=TRUE`.

Author(s)

Sergej Ruff

See Also

VirusHunterGatherer is available here: https://github.com/lauberlab/VirusHunterGatherer.

Examples

```
path <- system.file("extdata", "virushunter.tsv", package = "Virusparies")
file <- ImportVirusTable(path)

# plot 1
plot <- VhgIdenFacetedScatterPlot(file,cutoff = 1e-5)

plot

# plot 2 with custom data
custom_plot <- VhgIdenFacetedScatterPlot(file,
                                         cutoff = 1e-4,
                                         theme_choice = "dark",
                                         title = "Custom Scatterplot",
                                         title_size = 18,
                                         title_face = "italic",
                                         title_colour = "orange",
                                         xlabel = "Custom X Label",
                                         ylabel = "Custom Y Label",
                                         axis_title_size = 14,
                                         legend_position = "right",
                                         true_colour = "green",
                                         false_colour = "purple")

custom_plot

# import gatherer files
```

```
path2 <- system.file("extdata", "virusgatherer.tsv", package = "Virusparies")
vg_file <- ImportVirusTable(path2)

# vgplot: virusgatherer plot with ViralRefSeq_taxonomy as custom grouping
vgplot <- VhgIdenFacetedScatterPlot(vg_file,groupby = "ViralRefSeq_taxonomy")
vgplot
```

VhgIdentityScatterPlot

VhgIdentityScatterPlot: Scatter plot for refseq identity vs -log10 of refseq E-value

---

Description

VhgIdentityScatterPlot generates a scatter plot of viral refSeq identity vs. -log10 of viral refseq E-value. It colors the points based on phylum and adds a horizontal line representing the cutoff value.

Usage

```
VhgIdentityScatterPlot(
  file,
  groupby = "best_query",
  taxa_rank = "Family",
  cutoff = 1e-05,
  conlen_bubble_plot = FALSE,
  contiglen_breaks = 5,
  theme_choice = "linedraw",
  cut_colour = "#990000",
  title = "Scatterplot of viral reference E-values and sequence identity",
  title_size = 16,
  title_face = "bold",
  title_colour = "#2a475e",
  subtitle = NULL,
  subtitle_size = 12,
  subtitle_face = "bold",
  subtitle_colour = "#1b2838",
  xlabel = "Viral reference sequence identity (%)",
  ylabel = "-log10 of viral reference E-values",
  axis_title_size = 12,
  xtext_size = 10,
  x_angle = NULL,
  ytext_size = 10,
  y_angle = NULL,
  legend_title = "Group",
```

```
  legend_position = "bottom",
  legend_title_size = 12,
  legend_title_face = "bold",
  legend_text_size = 10,
  highlight_groups = NULL,
  group_unwanted_phyla = NULL
)
```

Arguments

file	VirusHunterGatherer hittable.
groupby	(optional): A character specifying the column containing the groups (default: "best_query"). Note: Gatherer hittables do not have a "best_query" column. Please provide an appropriate column for grouping.
taxa_rank	(optional): When groupby is set to "ViralRefSeq_taxonomy", specify the taxonomic rank to group your data by. Supported ranks are: • "Subphylum" • "Class" • "Subclass" • "Order" • "Suborder" • "Family" (default) • "Subfamily" • "Genus" (including Subgenus)
cutoff	(optional): A numeric value representing the cutoff for the refseq E-value (default: 1e-5).
conlen_bubble_plot	(optional): Logical value indicating whether the contig_len column should be used to size the bubbles in the plot. Applicable only to VirusGatherer hittables input (default: FALSE).
contiglen_breaks	(optional): Number of breaks (default: 5) for the bubble plot (for conlen_bubble_plot=TRUE).
theme_choice	(optional): A character indicating the ggplot2 theme to apply. Options include "minimal", "classic", "light", "dark", "void", "grey" (or "gray"), "bw", "linedraw" (default), and "test". Append "_dotted" to any theme to add custom dotted grid lines (e.g., "classic_dotted").
cut_colour	(optional): The color for the horizontal cutoff line (default: "#990000").
title	(optional): The title of the plot (default: "Scatterplot of viral reference E-values and sequence identity").
title_size	(optional): The size of the title text (default: 16).
title_face	(optional): The face (bold, italic, etc.) of the title text (default: "bold").
title_colour	(optional): The color of the title text (default: "#2a475e").
subtitle	(optional): The subtitle of the plot (default: NULL).
subtitle_size	(optional): The size of the subtitle text (default: 12).

`subtitle_face` (optional): The face (bold, italic, etc.) of the subtitle text (default: "bold").

`subtitle_colour` (optional): The color of the subtitle text (default: "#1b2838").

`xlabel` (optional): The label for the x-axis (default: "Viral reference sequence identity (%)").

`ylabel` (optional): The label for the y-axis (default: "-log10 of viral reference E-values").

`axis_title_size` (optional): The size of the axis titles (default: 12).

`xtext_size` (optional): The size of the x-axis text (default: 10).

`x_angle` (optional): An integer specifying the angle (in degrees) for the x-axis text labels. Default is NULL, meaning no change.

`ytext_size` (optional): The size of the y-axis text (default: 10).

`y_angle` (optional): An integer specifying the angle (in degrees) for the y-axis text labels. Default is NULL, meaning no change.

`legend_title` (optional): The title of the legend (default: "Group").

`legend_position` (optional): The position of the legend (default: "bottom).

`legend_title_size` (optional): The size of the legend title text (default: 12).

`legend_title_face` (optional): The face (bold, italic, etc.) of the legend title text (default: "bold").

`legend_text_size` (optional): The size of the legend text (default: 10).

`highlight_groups` (optional): A character vector specifying the names of viral groups to be highlighted in the plot (Default:NULL).

`group_unwanted_phyla` (optional): A character string specifying which group of viral phyla to retain in the analysis. Valid values are:

"rna" Retain only the phyla specified for RNA viruses.

"smalldna" Retain only the phyla specified for small DNA viruses.

"largedna" Retain only the phyla specified for large DNA viruses.

"others" Retain only the phyla that match small DNA, Large DNA and RNA viruses.

All other phyla not in the specified group will be grouped into a single category: "Non-RNA-virus" for `"rna"`, "Non-Small-DNA-Virus" for `"smalldna"`,"Non-Large-DNA-Virus" for `"largedna"`,or "Other Viruses" for `"others"`.

Details

VhgIdentityScatterPlot generates a scatter plot for refseq sequence identity vs -log10 of refseq E-value. It accepts both VirusHunter and VirusGatherer hittables as input. The plot includes:

- A line indicates whether the observed values are above or below the cutoff specified by the 'cutoff' argument (default: 1e-5).

- The option `conlen_bubble_plot = TRUE` generates a bubble plot where the size of points corresponds to "contig_len" (exclusive to VirusGatherer).

Tibble data frames containing summary statistics (median, Q1, Q3, mean, sd, min, and max) for 'ViralRefSeq_E' and 'ViralRefSeq_ident' values are generated. Optionally, summary statistics for 'contig_len' values are also included if applicable. These summary statistics, along with the plot object, are returned within a list object.

`highlight_groups` enables the user to specify one or more viral groups from the column indicated in the `groupby` argument. These groups will be highlighted in the plot.

Warning: In some cases, E-values might be exactly 0. When these values are transformed using -log10, R returns "inf" as the output. To avoid this issue, we replace all E-values that are 0 with the smallest e-value that is greater than 0. If the smallest E-value is above the user-defined cutoff, we use a value of `cutoff * 10^-10` to replace the zeros.

Value

A list containing the following components:

- Plot: A plot object representing the faceted scatterplot.
- evalue_stats: A tibble data frame with summary statistics for "ViralRefSeq_E" values.
- identity_stats: A tibble data frame with summary statistics for "ViralRefSeq_ident" values.
- contig_stats (optional): A tibble data frame with summary statistics for "contig_len" values, included only if VirusGatherer is used with `conlen_bubble_plot=TRUE`.

Author(s)

Sergej Ruff

See Also

VirusHunterGatherer is available here: https://github.com/lauberlab/VirusHunterGatherer.

Examples

```
path <- system.file("extdata", "virushunter.tsv", package = "Virusparies")
file <- ImportVirusTable(path)


# Basic plot
plot <- VhgIdentityScatterPlot(file,cutoff = 1e-5)

plot(plot$plot)

# Custom plot with additional arguments
custom_plot <- VhgIdentityScatterPlot(file,
                                      cutoff = 1e-5,
                                      theme_choice = "dark",
                                      cut_colour = "blue",
                                      title = "Custom Scatter Plot",
                                      title_size = 18,
```

```
                                    title_face = "italic",
                                    title_colour = "darkred",
                                    subtitle = "Custom Subtitle",
                                    subtitle_size = 14,
                                    subtitle_face = "italic",
                                    subtitle_colour = "purple",
                                    xlabel = "Custom X Label",
                                    ylabel = "Custom Y Label",
                                    axis_title_size = 14,
                                    xtext_size = 12,
                                    ytext_size = 12,
                                    legend_title = "Custom Legend",
                                    legend_position = "top",
                                    legend_title_size = 14,
                                    legend_title_face = "italic",
                                    legend_text_size = 12)

plot(custom_plot$plot)

# import gatherer files
path2 <- system.file("extdata", "virusgatherer.tsv", package = "Virusparies")
vg_file <- ImportVirusTable(path2)

# vgplot: virusgatherer plot with ViralRefSeq_taxonomy as custom grouping
vgplot <- VhgIdentityScatterPlot(vg_file,groupby = "ViralRefSeq_taxonomy")
vgplot$plot

# plot as bubble plot with contig length as size
vgplot_con <- VhgIdentityScatterPlot(vg_file,groupby = "ViralRefSeq_taxonomy",
conlen_bubble_plot = TRUE,contiglen_breaks = 4,legend_position = "right")

vgplot_con
```

`VhgPreprocessTaxa` *VhgPreprocessTaxa: preprocess ViralRefSeq_taxonomy elements*

Description

VhgPreprocessTaxa: preprocess ViralRefSeq_taxonomy elements

Usage

```
VhgPreprocessTaxa(file, taxa_rank, num_cores = 1)
```

Arguments

`file`	A data frame containing VirusHunter or VirusGatherer hittable results.
`taxa_rank`	(optional): Specify the taxonomic rank to group your data by. Supported ranks are:

- "Subphylum"
- "Class"
- "Subclass"
- "Order"
- "Suborder"
- "Family" (default)
- "Subfamily"
- "Genus" (including Subgenus)

`num_cores` (optional): Number of cores used (default: 1)

Details

Process the `ViralRefSeq_taxonomy` column.

Besides `best_query`, the user can utilize the `ViralRefSeq_taxonomy` column as `x_column` or `groupby` in plots. This column needs preprocessing because it is too long and has too many unique elements for effective grouping. The element containing the taxa suffix specified by the `taxa_rank` argument is used. NA values are replaced by "unclassified".

This function is used internally by every function that can use the `ViralRefSeq_taxonomy` column as input. The user can also apply this function independently to process the taxonomy column and filter for the selected taxa rank in their data.

For datasets with significantly more than 100,000 observations, it is recommended to use this function to ensure it is skipped in the plot functions.

The `num_cores` parameter allows you to divide the dataset into multiple parts, corresponding to the number of cores available. This enables parallel processing across multiple threads, thereby speeding up the overall processing time.

Value

file with preprocessed ViralRefSeq_taxonomy elements

Author(s)

Sergej Ruff

See Also

VirusHunterGatherer is available here: https://github.com/lauberlab/VirusHunterGatherer.

Examples

```
path <- system.file("extdata", "virushunter.tsv", package = "Virusparies")
file <- ImportVirusTable(path)

file_filtered <- VhgPreprocessTaxa(file,"Family")

cat("ViralRefSeq_taxonomy before processing:\n")
print(head(file$ViralRefSeq_taxonomy,5))
```

```
cat("ViralRefSeq_taxonomy after processing:\n")
print(head(file_filtered$ViralRefSeq_taxonomy,5))
```

VhgRunsBarplot *VhgRunsBarplot: Generate a bar plot showing the number of data sets with unique runs found for each Virus group.*

Description

VhgRunsBarplot: Generate a bar plot showing the number of data sets with unique runs found for each Virus group.

Usage

```
VhgRunsBarplot(
  file,
  groupby = "best_query",
  taxa_rank = "Family",
  cut = 1e-05,
  reorder_criteria = "max",
  theme_choice = "linedraw",
  flip_coords = TRUE,
  title = "Distribution of viral groups detected across query sequences",
  title_size = 16,
  title_face = "bold",
  title_colour = "#2a475e",
  subtitle = "default",
  subtitle_size = 12,
  subtitle_face = "bold",
  subtitle_colour = "#1b2838",
  xlabel = "Viral group",
  ylabel = "Number of datasets with hits for group",
  axis_title_size = 12,
  xtext_size = 10,
  x_angle = NULL,
  ytext_size = 10,
  y_angle = NULL,
  remove_group_labels = FALSE,
  legend_title = "Phylum",
  legend_position = "bottom",
  legend_title_size = 12,
  legend_title_face = "bold",
  legend_text_size = 10,
  plot_text = 3,
```

```
  plot_text_size = 3.5,
  plot_text_position_dodge = 0.9,
  plot_text_hjust = -0.1,
  plot_text_vjust = 0.5,
  plot_text_colour = "black",
  facet_ncol = NULL,
  group_unwanted_phyla = NULL
)
```

Arguments

file	A data frame containing VirusHunter or VirusGatherer hittable results.
groupby	(optional): A character specifying the column containing the groups (default: "best_query"). Note: Gatherer hittables do not have a "best_query" column. Please provide an appropriate column for grouping.
taxa_rank	(optional): When groupby is set to "ViralRefSeq_taxonomy", specify the taxonomic rank to group your data by. Supported ranks are: • "Subphylum" • "Class" • "Subclass" • "Order" • "Suborder" • "Family" (default) • "Subfamily" • "Genus" (including Subgenus)
cut	(optional): A numeric value representing the cutoff for the refseq E-value (default: 1e-5). Removes rows in file with values larger than cutoff value in "ViralRefSeq_E" column.
reorder_criteria	(optional): Character string specifying the criteria for reordering the x-axis ('max' (default), 'min','phylum',phylum_max,phylum_min). NULL sorts alphabetically.
theme_choice	(optional): A character indicating the ggplot2 theme to apply. Options include "minimal", "classic", "light", "dark", "void", "grey" (or "gray"), "bw", "linedraw" (default), and "test". Append "_dotted" to any theme to add custom dotted grid lines (e.g., "classic_dotted").
flip_coords	(optional): Logical indicating whether to flip the coordinates of the plot (default: TRUE).
title	(optional): The title of the plot (default: "Distribution of viral groups detected across query sequences").
title_size	(optional): The size of the title text (default: 16).
title_face	(optional): The face (bold, italic, etc.) of the title text (default: "bold").
title_colour	(optional): The color of the title text (default: "#2a475e").

`subtitle` (optional): A character specifying the subtitle of the plot. Default is "default", which calculates the total number of data sets with hits and returns it as "total number of data sets with hits: " followed by the calculated number. an empty string ("") removes the subtitle.

`subtitle_size` (optional): The size of the subtitle text (default: 12).

`subtitle_face` (optional): The face (bold, italic, etc.) of the subtitle text (default: "bold").

`subtitle_colour` (optional): The color of the subtitle text (default: "#1b2838").

`xlabel` (optional): The label for the x-axis (default: "Viral group").

`ylabel` (optional): The label for the y-axis (default: "Number of data sets with hits for group").

`axis_title_size` (optional): The size of the axis titles (default: 12).

`xtext_size` (optional): The size of the x-axis text (default: 10).

`x_angle` (optional): An integer specifying the angle (in degrees) for the x-axis text labels. Default is NULL, meaning no change.

`ytext_size` (optional): The size of the y-axis text (default: 10).

`y_angle` (optional): An integer specifying the angle (in degrees) for the y-axis text labels. Default is NULL, meaning no change.

`remove_group_labels` (optional): If `TRUE`, the group labels will be removed; if `FALSE` or omitted, the labels will be displayed.

`legend_title` (optional): A character specifying the title for the legend (default: "Phylum").

`legend_position` (optional): The position of the legend (default: "bottom).

`legend_title_size` (optional): Numeric specifying the size of the legend title text (default: 12).

`legend_title_face` (optional): A character specifying the font face for the legend title text (default: "bold").

`legend_text_size` (optional): Numeric specifying the size of the legend text (default: 10).

`plot_text` (optional): An index (0-3) to select the variable for text labels.

- 0 = None.
- 1 = Number of viral groups detected across query sequences.
- 2 = Only the percentage.
- 3 = Both (Default).

`plot_text_size` (optional): The size of the text labels added to the plot (default: 3.5).

`plot_text_position_dodge` (optional): The degree of dodging for positioning text labels (default: 0.9).

`plot_text_hjust` (optional): The horizontal justification of text labels (default: -0.1).

`plot_text_vjust`
: (optional): The vertical justification of text labels (default: 0.5). It is recommended to change `vjust` when setting `flip_coords = FALSE`.

`plot_text_colour`
: (optional): The color of the text labels added to the plot (default: "black").

`facet_ncol`
: (optional): The number of columns for faceting (default: NULL). It is recommended to specify this when the number of viral groups is high, to ensure they fit well in one plot.

`group_unwanted_phyla`
: (optional): A character string specifying which group of viral phyla to retain in the analysis. Valid values are:

  "rna" Retain only the phyla specified for RNA viruses.

  "smalldna" Retain only the phyla specified for small DNA viruses.

  "largedna" Retain only the phyla specified for large DNA viruses.

  "others" Retain only the phyla that match small DNA, Large DNA and RNA viruses.

  All other phyla not in the specified group will be grouped into a single category: "Non-RNA-virus" for `"rna"`, "Non-Small-DNA-Virus" for `"smalldna"`,"Non-Large-DNA-Virus" for `"largedna"`,or "Other Viruses" for `"others"`.

Details

VhgRunsBarplot generates a bar plot showing the number of data sets with unique runs found for each Virus group. It takes VirusHunter and VirusGatherer hittables as Input.

Only significant values below the threshold specified by the 'cut' argument (default: 1e-5) are included in the plot.

Value

A list containing the bar plot and tabular data with information from the plot.

Author(s)

Sergej Ruff

See Also

VirusHunterGatherer is available here: https://github.com/lauberlab/VirusHunterGatherer.

Examples

```
# import data
path <- system.file("extdata", "virushunter.tsv", package = "Virusparies")
file <- ImportVirusTable(path)

# plot 1: plot boxplot for "identity"
plot <- VhgRunsBarplot(file,cut = 1e-5)
plot
```

VhgRunsTable *VhgRunsTable: Generate a gt summary table of unique runs for each virus group*

Description

VhgRunsTable generates a summary table of unique runs for each virus group based on the input data set.

Usage

```
VhgRunsTable(
  vh_file,
  groupby = "best_query",
  taxa_rank = "Family",
  cut = 1e-05,
  title = "Summary of unique runs by virus group",
  title_align = "left",
  names_ = NULL,
  align = "left",
  subtitle = NULL,
  data_row.pad = 6,
  column_colour = "dodgerblue4",
  title_size = 26,
  subtitle_size = 14,
  title_weight = "bold",
  title_colour = "dodgerblue4",
  table_font_size = 14,
  cell_colour = "grey90",
  col_everyrow = FALSE
)
```

Arguments

vh_file	A data frame containing the Virushunter hittables results.
groupby	(optional): A character specifying the column containing the groups (default: "best_query"). Note: Gatherer hittables do not have a "best_query" column. Please provide an appropriate column for grouping.
taxa_rank	(optional): When groupby is set to "ViralRefSeq_taxonomy", specify the taxonomic rank to group your data by. Supported ranks are: • "Subphylum" • "Class" • "Subclass"

- "Order"
- "Suborder"
- "Family" (default)
- "Subfamily"
- "Genus" (including Subgenus)

`cut` (optional): A numeric value representing the cutoff for the refseq E-value (default: 1e-5). Removes rows in file with values larger than cutoff value in "Viral-RefSeq_E" column.

`title` (optional): The title of the plot (default: "Summary of unique runs by virus group").

`title_align` (optional): A character vector specifying the alignment of title (and subtitle) text. Possible values are "left" (default), "center", or "right".

`names_` (optional): A vector of length 3 containing column names (default: c("Virus Group","Number of Unique SRA Runs","SRAs Found")).

`align` (optional): A character vector specifying the alignment of text in the table columns. Possible values are "left" (default), "center", or "right".

`subtitle` (optional): A character specifying the subtitle of the plot (default: NULL).

`data_row.pad` (optional): Numeric value specifying the row padding (default: 6).

`column_colour` (optional): character specifying the background colour for the column header (default: "dodgerblue4").

`title_size` (optional): The size of the title text (default: 26).

`subtitle_size` (optional): Numeric specifying the size of the subtitle text (default: 14).

`title_weight` (optional): Character or numeric value specifying title font weight. The weight of the font can be modified thorough a text-based option such as "normal", "bold" (default), "lighter", "bolder", or, a numeric value between 1 and 1000, inclusive.

`title_colour` (optional): A character specifying the color for the title text (default: "dodgerblue4").

`table_font_size` (optional): Numeric value specifying table font size. This will change font size for the column header and for all values in each cell (default: 14).

`cell_colour` (optional): Character specifying cell colour (default: "grey90").

`col_everyrow` (optional): Bool value specifying if every row or every second row of the table should be filled with the colour from the cell_colour argument. col_everyrow = TRUE colors every row and col_everyrow = FALSE (default) colors every second row.

Details

VhgRunsTable calculates the number of unique runs for each virus group from the input data set. It takes VirusHunter hittables as input and determines how many runs (SRA runs or local FASTQ files) found a specific virus family.

A graphical table is returned with two columns. The first column contains the name of the virus group, while the second column contains all IDs of SRA runs or local files that found that virus group.

Value

A formatted gt table summarizing unique runs for each virus group

See Also

VhgRunsBarplot

Examples

```
path <- system.file("extdata", "virushunter.tsv", package = "Virusparies")
vh_file <- ImportVirusTable(path)

# example 1: generate table with defaul arguments
table <- VhgRunsTable(vh_file,cut = 1e-5)

table

# example 2: generate table with custom arguments

table_2 <- VhgRunsTable(vh_file,title = "test",title_align="right",
names_ = c("column_1","column_2","column_3"),align = "right",subtitle="subtitlele")

table_2

# example 3: virusgatherer example
# import gatherer files
path2 <- system.file("extdata", "virusgatherer.tsv", package = "Virusparies")
vg_file <- ImportVirusTable(path2)


table_3 <- VhgRunsTable(vg_file,groupby = "ViralRefSeq_taxonomy")

table_3
```

VhgSubsetHittable *VhgSubsetHittable: Filter VirusHunter and VirusGatherer hittables*

Description

VhgSubsetHittable filters a VirusHunter or VirusGatherer hittable based on specified criteria, including specific virus groups, minimum number of hits, and observations below certain E-value or identity percentage criteria.

Usage

```
VhgSubsetHittable(
  file,
  group_column = "best_query",
```

```
  virus_groups = NULL,
  num_hits_min = NULL,
  ViralRefSeq_E_criteria = NULL,
  ViralRefSeq_ident_criteria = NULL,
  contig_len_criteria = NULL
)
```

Arguments

`file`	A data frame containing VirusHunter or VirusGatherer hittable results.
`group_column`	A string indicating the column containing the virus groups specified in the virus_groups argument. Note: Gatherer hittables do not have a "best_query" column. Please provide an appropriate column for grouping.
`virus_groups`	A character vector specifying virus groups to filter by.
`num_hits_min`	Minimum number of hits required. Default is NULL, which means no filter based on num_hits.
`ViralRefSeq_E_criteria`	Maximum E-value threshold for ViralRefSeq_E criteria. Default is NULL, which means no filter based on ViralRefSeq_E.
`ViralRefSeq_ident_criteria`	Maximum or minimum sequence identity percentage threshold for ViralRefSeq_ident criteria. Default is NULL, which means no filter based on ViralRefSeq_ident. If positive, filters where ViralRefSeq_ident is above the threshold. If negative, filters where ViralRefSeq_ident is below the absolute value of the threshold.
`contig_len_criteria`	(Gatherer only): Minimum contig length required.

Details

The function filters the input VirusHunter or VirusGatherer data (`file`) based on specified criteria:

- `group_column`: Specifies the column to filter by, which must be either "ViralRefSeq_taxonomy" or "best_query".
- `virus_groups`: Allows filtering by specific virus groups. If NULL, all virus groups are included.
- `num_hits_min`: Filters rows where the number of hits ("num_hits") is greater than or equal to the specified minimum.
- `ViralRefSeq_E_criteria`: Filters rows where the E-value ("ViralRefSeq_E") is below the specified maximum threshold.
- `ViralRefSeq_ident_criteria`: Filters rows where the sequence identity percentage ("ViralRefSeq_ident") is above or below the specified threshold. Use a positive value to filter where ViralRefSeq_ident is above the threshold, and a negative value to filter where ViralRefSeq_ident is below the absolute value of the threshold.
- `contig_len_criteria`: (Gatherer only) Filters rows where the contig length ("contig_len") is greater than or equal to the specified threshold.

Value

A filtered dataframe based on the specified criteria.

Author(s)

Sergej Ruff

See Also

VirusHunterGatherer is available here: https://github.com/lauberlab/VirusHunterGatherer.

Examples

```
path <- system.file("extdata", "virushunter.tsv", package = "Virusparies")
file <- ImportVirusTable(path)

cat("The dimensions of the VirusHunter hittable before filtering are: \n");dim(file)

file_filtered <- VhgSubsetHittable(file,group_column = "best_query",
virus_groups = "Anello_ORF1core",
num_hits_min = 4,ViralRefSeq_ident_criteria = -90,ViralRefSeq_E_criteria = 0.00001)

cat("The dimensions of the VirusHunter Hittable after filtering are: \n");dim(file_filtered)

# other examples for viral_group

# Include a single group:
result1 <- VhgSubsetHittable(file, virus_groups = "Hepadna-Nackedna_TP")
# Include multiple groups:
result2 <- VhgSubsetHittable(file, virus_groups = c("Hepadna-Nackedna_TP", "Gemini_Rep"))
# Exclude a single group:
result3 <- VhgSubsetHittable(file, virus_groups = list(exclude = "Hepadna-Nackedna_TP"))
# Exclude multiple groups:
result4 <- VhgSubsetHittable(file, virus_groups = list(exclude =
 c("Hepadna-Nackedna_TP", "Anello_ORF1core")))
```

VhgSumHitsBarplot *VhgSumHitsBarplot: Generate a bar plot showing the sum of reads/contigs for each virus family*

Description

VhgSumHitsBarplot preprocesses virus data for plotting and generates a bar plot showing the sum of reads/contigs for each virus family from the input data set.

Usage

```
VhgSumHitsBarplot(
  file,
  groupby = "best_query",
  taxa_rank = "Family",
  y_column = "num_hits",
  cut = 1e-05,
  reorder_criteria = "max",
  theme_choice = "linedraw",
  flip_coords = TRUE,
  title = "Distribution of hits for each virus group",
  title_size = 16,
  title_face = "bold",
  title_colour = "#2a475e",
  subtitle = "default",
  subtitle_size = 12,
  subtitle_face = "bold",
  subtitle_colour = "#1b2838",
  xlabel = "Viral group",
  ylabel = "Total number of hits",
  axis_title_size = 12,
  xtext_size = 10,
  x_angle = NULL,
  ytext_size = 10,
  y_angle = NULL,
  remove_group_labels = FALSE,
  legend_title = "Phylum",
  legend_position = "bottom",
  legend_title_size = 12,
  legend_title_face = "bold",
  legend_text_size = 10,
  plot_text = 3,
  plot_text_size = 3.5,
  plot_text_position_dodge = 0.9,
  plot_text_hjust = -0.1,
  plot_text_vjust = 0.5,
  plot_text_colour = "black",
  facet_ncol = NULL,
  group_unwanted_phyla = NULL
)
```

Arguments

file	A data frame containing VirusHunters hittable results.
groupby	(optional): A character specifying the column containing the groups (default: "best_query").
taxa_rank	(optional): When groupby is set to "ViralRefSeq_taxonomy", specify the taxonomic rank to group your data by. Supported ranks are:

- "Subphylum"
- "Class"
- "Subclass"
- "Order"
- "Suborder"
- "Family" (default)
- "Subfamily"
- "Genus" (including Subgenus)

`y_column` A character specifying the column containing the values to be compared. Currently "ViralRefSeq_contigs" (micro-contigs),"contigs", and "num_hits" (reads) are supported columns (default:"num_hits").

`cut` (optional): A numeric value representing the cutoff for the refseq E-value (default: 1e-5). Removes rows in file with values larger than cutoff value in "ViralRefSeq_E" column.

`reorder_criteria` (optional): Character string specifying the criteria for reordering the x-axis ('max' (default), 'min','phylum',phylum_max,phylum_min). NULL sorts alphabetically.

`theme_choice` (optional): A character indicating the ggplot2 theme to apply. Options include "minimal", "classic", "light", "dark", "void", "grey" (or "gray"), "bw", "linedraw" (default), and "test". Append "_dotted" to any theme to add custom dotted grid lines (e.g., "classic_dotted").

`flip_coords` (optional): Logical indicating whether to flip the coordinates of the plot (default: TRUE).

`title` (optional): The title of the plot (default: "Distribution of hits for each virus group").

`title_size` (optional): The size of the title text (default: 16).

`title_face` (optional): The face (bold, italic, etc.) of the title text (default: "bold").

`title_colour` (optional): The color of the title text (default: "#2a475e").

`subtitle` (optional): A character specifying the subtitle of the plot. Default is "total number of hits/micro-contigs: " followed by the calculated number.empty string ("") removes subtitle.

`subtitle_size` (optional): Numeric specifying the size of the subtitle text(default: 12).

`subtitle_face` (optional): A character specifying the font face for the subtitle text (default: "bold").

`subtitle_colour` (optional): A character specifying the color for the subtitle text (default: "#1b2838").
.

`xlabel` (optional): The label for the x-axis (default: "Viral group").

`ylabel` (optional): The label for the y-axis (default: "Total number of hits").

`axis_title_size` (optional): The size of the axis titles (default: 12).

`xtext_size` (optional): The size of the x-axis text (default: 10).

`x_angle` (optional): An integer specifying the angle (in degrees) for the x-axis text labels. Default is NULL, meaning no change.

`ytext_size` (optional): The size of the y-axis text (default: 10).

`y_angle` (optional): An integer specifying the angle (in degrees) for the y-axis text labels. Default is NULL, meaning no change.

`remove_group_labels`
(optional): If `TRUE`, the group labels will be removed; if `FALSE` or omitted, the labels will be displayed.

`legend_title` (optional): A character specifying the title for the legend (default: "Phylum").

`legend_position`
(optional): A character specifying the position of the legend (default: "bottom").

`legend_title_size`
(optional): Numeric specifying the size of the legend title text (default: 12).

`legend_title_face`
(optional): A character specifying the font face for the legend title text (default: "bold").

`legend_text_size`
(optional): Numeric specifying the size of the legend text (default: 10).

`plot_text` (optional): An index (0-3) to select the variable for text labels.

- 0 = None.
- 1 = Number of hits for each viral group.
- 2 = Only the percentage.
- 3 = Both (Default).

`plot_text_size` (optional): The size of the text labels added to the plot (default: 3.5).

`plot_text_position_dodge`
(optional): The degree of dodging for positioning text labels (default: 0.9).

`plot_text_hjust`
(optional): The horizontal justification of text labels (default: -0.1).

`plot_text_vjust`
(optional): The vertical justification of text labels (default: 0.5). It is recommended to change `vjust` when setting `flip_coords = FALSE`.

`plot_text_colour`
(optional): The color of the text labels added to the plot (default: "black").

`facet_ncol` (optional): The number of columns for faceting (default: NULL). It is recommended to specify this when the number of viral groups is high, to ensure they fit well in one plot.

`group_unwanted_phyla`
(optional): A character string specifying which group of viral phyla to retain in the analysis. Valid values are:

"rna" Retain only the phyla specified for RNA viruses.

"smalldna" Retain only the phyla specified for small DNA viruses.

"largedna" Retain only the phyla specified for large DNA viruses.

"others" Retain only the phyla that match small DNA, Large DNA and RNA viruses.

All other phyla not in the specified group will be grouped into a single category: "Non-RNA-virus" for "rna", "Non-Small-DNA-Virus" for "smalldna","Non-Large-DNA-Virus" for "largedna",or "Other Viruses" for "others".

Details

VhgSumHitsBarplot preprocesses virus data for plotting by calculating the sum of hits for each virus family from the input data set (accepts only VirusHunter hittables). It then generates a bar plot showing the sum of hits for each virus family. Additionally, it returns the processed data for further analysis.

Value

A list containing the generated bar plot and processed data.

Author(s)

Sergej Ruff

See Also

VirusHunterGatherer is available here: https://github.com/lauberlab/VirusHunterGatherer.

Examples

```
path <- system.file("extdata", "virushunter.tsv", package = "Virusparies")
file <- ImportVirusTable(path)

# plot 1: plot boxplot for reads
plot <- VhgSumHitsBarplot(file,cut = 1e-5)
plot

# plot 2: plot boxplot for micro_reads
plot_reads <- VhgSumHitsBarplot(file,cut = 1e-5,
y_column = "ViralRefSeq_contigs")
plot_reads

# import gatherer files
path2 <- system.file("extdata", "virusgatherer.tsv", package = "Virusparies")
vg_file <- ImportVirusTable(path2)


# plot 3: contigs in Gatherer
contig_plot <- VhgSumHitsBarplot(vg_file,groupby = "ViralRefSeq_taxonomy",
y_column = "contig")
contig_plot
```

VhgTabularRasa *VhgTabularRasa: Generate Custom Formatted Graphical Tables*

Description

VhgTabularRasa creates a formatted table using the gt package.

Usage

```
VhgTabularRasa(
  file,
  title = "Graphical Table",
  title_align = "left",
  names_ = NULL,
  align = "left",
  subtitle = NULL,
  data_row.pad = 6,
  column_colour = "dodgerblue4",
  title_size = 26,
  subtitle_size = 14,
  title_weight = "bold",
  title_colour = "dodgerblue4",
  table_font_size = 14,
  cell_colour = "grey90",
  col_everyrow = FALSE
)
```

Arguments

file	A data frame.
title	(optional): The title of the plot (default: "Graphical Table").
title_align	(optional): A character vector specifying the alignment of title (and subtitle) text. Possible values are "left" (default), "center", or "right".
names_	(optional): A vector containing column names, with a length matching the number of columns in the file argument (default: names(file)).
align	(optional): A character vector specifying the alignment of text in the table columns. Possible values are "left" (default), "center", or "right".
subtitle	(optional): A character specifying the subtitle of the plot (default: NULL).
data_row.pad	(optional): Numeric value specifying the row padding (default: 6).
column_colour	(optional): character specifying the background colour for the column header (default: "dodgerblue4").
title_size	(optional): The size of the title text (default: 26).
subtitle_size	(optional): Numeric specifying the size of the subtitle text (default: 14).

`title_weight` (optional): Character or numeric value specifying title font weight. The weight of the font can be modified thorough a text-based option such as "normal", "bold" (default), "lighter", "bolder", or, a numeric value between 1 and 1000, inclusive.

`title_colour` (optional): A character specifying the color for the title text (default: "dodgerblue4").

`table_font_size` (optional): Numeric value specifying table font size. This will change font size for the column header and for all values in each cell (default: 14).

`cell_colour` (optional): Character specifying cell colour (default: "grey90").

`col_everyrow` (optional): Bool value specifying if every row or every second row of the table should be filled with the colour from the cell_colour argument. col_everyrow = TRUE colors every row and col_everyrow = FALSE (default) colors every second row.

Details

VhgTabularRasa creates a formatted table using the `gt` package, based on input data with specified column names. It is particularly useful for generating tables that cannot be produced with `vhRunsTable`, when the input data does not originate from the `vhRunsBarplot` functions.

The `VhgTabularRasa` function allows users to generate tables styled like Virusparies tables using their own input data. Additionally, users can create custom tables by adjusting the parameters within this function.

`VhgTabularRasa` just like `vhRunsTable` empowers users to tailor their tables to suit their needs. With its customizable features, users can effortlessly modify titles, subtitles, and column names. By default, column names are derived from the data frame's structure using `names(file)`. If the data frame lacks column names, an error message is triggered. However, users can supply their own column names via the `names_` argument, requiring a vector of names matching the data frame's column count.

`VhgTabularRasa` takes data frames as Input. Passing any other object type results in an error message. Users can fine-tune their tables with options to adjust text attributes, alignment, and size, as well as row padding and color schemes for titles, subtitles, columns, and backgrounds. If that is not enough, `VhgTabularRasa` returns an gt tables object, which can be further manipulated with the `gt` package available on CRAN.

Value

Returns a `gt` table object formatted according to the specified parameters.

Author(s)

Sergej Ruff

References

Iannone R, Cheng J, Schloerke B, Hughes E, Lauer A, Seo J(2024). *gt: Easily Create Presentation-Ready DisplayTables*. R package version 0.10.1,https://CRAN.R-project.org/package=gt.

See Also

VirusHunterGatherer is available here: https://github.com/lauberlab/VirusHunterGatherer.

Examples

```
path <- system.file("extdata", "virushunter.tsv", package = "Virusparies")
vh_file <- ImportVirusTable(path)

# plot 1: plot boxplot for "identity"
identity <- VhgBoxplot(vh_file,y_column = "ViralRefSeq_ident")

# generate table
VhgTabularRasa(identity$summary_stats)


# example 2: plot part of Vh_file (could be any other table)
# using first 10 rows of SRA_run,num_hits,bestquery,ViralRefSeq_E and Identity col.
vh_file_part <- vh_file[c(1:10),c(1,7,9,10,11)]

VhgTabularRasa(vh_file_part,title = "first 10 rows of vh_file",subtitle =
"example for any table",names_ = c("Runs","Number of Contigs","Best Query Result",
"Refrence E-Value","Refrence Identity"))
```

`Virusparies` *Virusparies Package*

Description

Virusparies designed for visualizing output from VirusHunterGatherer.

Details

VirusHunterGatherer is a pipeline designed for data-driven virus discovery (DDVD). It involves two steps: (i) Virushunter conducts sensitive homology-based detection of viral sequence reads in unprocessed data, identifying the most conserved regions of a virus, which serve as seeds for the (ii) Virusgatherer step that assembles full-length viral genome sequences.

The Virusparies package provides a set of plotting functions tailored for visualizing Virushunter-Gatherer hittables. The name draws inspiration from the hunter-gatherer metaphor, with "paries" derived from Latin meaning "wall". It symbolizes the parietal art left by ancient hunters and gatherers on walls, summarizing their stories and beliefs.

Functions

This package includes the following functions:

Import:

- ImportVirusTable: Import VirusHunterGatherer hittables into R.

VirusHunterGatherer Plots:

- VhgBoxplot: Box plot plotting refseq sequence identity, E-values or contig length for each group.
- VhgIdenFacetedScatterPlot: Faceted scatter plot for reference sequence identity vs -log10 of reference E-value.
- VhgIdentityScatterPlot: Scatter plot for reference sequence identity vs -log10 of reference E-value.
- VhgRunsBarplot: Bar plot showing how many unique runs map against each virus.
- VhgSumHitsBarplot: Bar plot for the sum of hits for each virus found in group.

VirusGatherer only plots:

- VgConLenViolin: Violin plot to visualize the distribution of contig lengths.

Graphical Tables(gt):

- VhgRunsTable: Generate a gt-table for VhgRunsBarplot.
- VhgTabularRasa: Generate custome gt-tables.

Export:

- ExportVirusDataFrame: Export data frames.
- ExportVirusGt: Export graphical tables.
- ExportVirusPlot: Export plots.

Utils:

- CombineHittables: Combine hittables.
- SummarizeViralStats: Generate summary stats outside of plot functions.
- VhgAddPhylum: Extract phylum information.
- VhgGetSubject: Process and Count Viral Subjects within Groups.
- VhgPreprocessTaxa: Process ViralRefSeq_taxonomy column.
- VhgSubsetHittable: Filter VirusHunterGatherer data based on user's own criteria.

Author(s)

Maintainer: Sergej Ruff (<serijnh@gmail.com>)

Other contributors:

- Chris Lauber (<chris.lauber@twincore.de>)
- Li Chuin, Chong (<lichuin.chong@twincore.de>)

If you have any questions, suggestions, or issues, please feel free to contact Sergej Ruff (<serijnh@gmail.com>).

See Also

VirusHunterGatherer is available on: https://github.com/lauberlab/VirusHunterGatherer.

If you encounter any errors or issues, please report them at: https://github.com/SergejRuff/Virusparies/issues.

Index