

Package ‘EnhancedVolcano’

October 16, 2019

Type Package

Title Publication-ready volcano plots with enhanced colouring and labeling

Version 1.2.0

Author Kevin Blighe

Maintainer Kevin Blighe <kevin@clinicalbioinformatics.co.uk>

Description

Volcano plots represent a useful way to visualise the results of differential expression analyses. Here, we present a highly-configurable function that produces publication-ready volcano plots. EnhancedVolcano will attempt to fit as many transcript names in the plot window as possible, thus avoiding clogging up the plot with labels that could not otherwise have been read. Other functionality allows the user to identify up to 3 different types of attributes in the same plot space via colour, shape, and shade parameter configurations.

License GPL-3

Depends ggplot2, ggrepel

Imports

Suggests RUnit, BiocGenerics, knitr, DESeq2, pasilla, airway, gridExtra, magrittr

URL <https://github.com/kevinblighe/EnhancedVolcano>

biocViews RNASeq, GeneExpression, Transcription, DifferentialExpression, ImmunoOncology

VignetteBuilder knitr

git_url <https://git.bioconductor.org/packages/EnhancedVolcano>

git_branch RELEASE_3_9

git_last_commit 7652cfd

git_last_commit_date 2019-05-02

Date/Publication 2019-10-15

R topics documented:

EnhancedVolcano-package	2
EnhancedVolcano	2

Index	8
--------------	----------

EnhancedVolcano-package

EnhancedVolcano: Publication-ready volcano plots with enhanced colouring and labeling.

Description

Volcano plots represent a useful way to visualise the results of differential expression analyses. Here, we present a highly-configurable function that produces publication-ready volcano plots. EnhancedVolcano will attempt to fit as many transcript names in the plot window as possible, thus avoiding 'clogging' up the plot with labels that could not otherwise have been read.

EnhancedVolcano

Publication-ready volcano plots with enhanced colouring and labeling.

Description

Volcano plots represent a useful way to visualise the results of differential expression analyses. Here, we present a highly-configurable function that produces publication-ready volcano plots. EnhancedVolcano will attempt to fit as many transcript names in the plot window as possible, thus avoiding 'clogging' up the plot with labels that could not otherwise have been read. Other functionality allows the user to identify up to 3 different types of attributes in the same plot space via colour, shape, and shade parameter configurations.

Usage

```
EnhancedVolcano(
  toptable,
  lab,
  x,
  y,
  selectLab = NULL,
  xlim = c(min(toptable[,x], na.rm=TRUE),
    max(toptable[,x], na.rm=TRUE)),
  ylim = c(0, max(-log10(toptable[,y]), na.rm=TRUE) + 5),
  xlab = bquote(~Log[2]~ "fold change"),
  ylab = bquote(~-Log[10]~italic(P)),
  axisLabSize = 18,
  title = 'Volcano plot',
  subtitle = 'Bioconductor package EnhancedVolcano',
  caption = paste0('Total = ', nrow(toptable), ' variables'),
  titleLabSize = 18,
  subtitleLabSize = 14,
  captionLabSize = 14,
  pCutoff = 10e-6,
  pLabellingCutoff = pCutoff,
  FCcutoff = 1.0,
  cutoffLineType = 'longdash',
```

```

cutoffLineCol = 'black',
cutoffLineWidth = 0.4,
transcriptPointSize = 0.8,
transcriptLabSize = 3.0,
transcriptLabCol = 'black',
transcriptLabFace = 'plain',
transcriptLabhjust = 0,
transcriptLabvjust = 1.5,
boxedlabels = FALSE,
shape = 19,
shapeCustom = NULL,
col = c("grey30", "forestgreen", "royalblue", "red2"),
colCustom = NULL,
colAlpha = 1/2,
legend = c("NS", "Log2 FC", "P", "P & Log2 FC"),
legendPosition = "top",
legendLabSize = 14,
legendIconSize = 4.0,
legendVisible = TRUE,
shade = NULL,
shadeLabel = NULL,
shadeAlpha = 1/2,
shadeFill = "grey",
shadeSize = 0.01,
shadeBins = 2,
drawConnectors = FALSE,
widthConnectors = 0.5,
typeConnectors = 'closed',
endsConnectors = 'first',
lengthConnectors = unit(0.01, 'npc'),
colConnectors = 'grey10',
hline = NULL,
hlineType = 'longdash',
hlineCol = 'black',
hlineWidth = 0.4,
vline = NULL,
vlineType = 'longdash',
vlineCol = 'black',
vlineWidth = 0.4,
gridlines.major = TRUE,
gridlines.minor = TRUE,
border = "partial",
borderWidth = 0.8,
borderColour = "black")

```

Arguments

toptable	A data-frame of test statistics (if not, a data frame, an attempt will be made to convert it to one). Requires at least the following: column for transcript names (can be rownames); a column for log2 fold changes; a column for nominal or adjusted p-value. REQUIRED .
lab	A column name in toptable containing transcript names. Can be rownames(toptable).

	REQUIRED.
x	A column name in toptable containing log ₂ fold changes. REQUIRED.
y	A column name in toptable containing nominal or adjusted p-values. REQUIRED.
selectLab	A vector containing a subset of lab. DEFAULT = NULL. OPTIONAL.
xlim	Limits of the x-axis. DEFAULT = c(min(toptable[,x], na.rm=TRUE), max(toptable[,x], na.rm=TRUE)). OPTIONAL.
ylim	Limits of the y-axis. DEFAULT = c(0, max(-log ₁₀ (toptable[,y]), na.rm=TRUE) + 5). OPTIONAL.
xlab	Label for x-axis. DEFAULT = bquote(~Log[2]~ "fold change"). OPTIONAL.
ylab	Label for y-axis. DEFAULT = bquote(~-Log[10]~italic(P)). OPTIONAL.
axisLabSize	Size of x- and y-axis labels. DEFAULT = 18. OPTIONAL.
title	Plot title. DEFAULT = 'Volcano plot'. OPTIONAL.
subtitle	Plot subtitle. DEFAULT = 'Bioconductor package, EnhancedVolcano'. OPTIONAL.
caption	Plot caption. DEFAULT = paste0('Total = ', nrow(toptable), ' variables'). OPTIONAL.
titleLabSize	Size of plot title. DEFAULT = 18. OPTIONAL.
subtitleLabSize	Size of plot subtitle. DEFAULT = 14. OPTIONAL.
captionLabSize	Size of plot caption. DEFAULT = 14. OPTIONAL.
pCutoff	Cut-off for statistical significance. A horizontal line will be drawn at -log ₁₀ (pCutoff). DEFAULT = 10e-6. OPTIONAL.
pLabellingCutoff	Labelling cut-off for statistical significance. DEFAULT = pCutoff. OPTIONAL.
FCcutoff	Cut-off for absolute log ₂ fold-change. Vertical lines will be drawn at the negative and positive values of log ₂ FCcutoff. DEFAULT = 1.0. OPTIONAL.
cutoffLineType	Line type for FCcutoff and pCutoff ("blank", "solid", "dashed", "dotted", "dot-dash", "longdash", "twodash"). DEFAULT = "longdash". OPTIONAL.
cutoffLineCol	Line colour for FCcutoff and pCutoff. DEFAULT = "black". OPTIONAL.
cutoffLineWidth	Line width for FCcutoff and pCutoff. DEFAULT = 0.4. OPTIONAL.
transcriptPointSize	Size of plotted points for each transcript. DEFAULT = 0.8. OPTIONAL.
transcriptLabSize	Size of labels for each transcript. DEFAULT = 3.0. OPTIONAL.
transcriptLabCol	Colour of labels for each transcript. DEFAULT = 'black'. OPTIONAL.
transcriptLabFace	Font face of labels for each transcript. DEFAULT = 'plain'. OPTIONAL.
transcriptLabhjust	Horizontal adjustment of label for each transcript. DEFAULT = 0. OPTIONAL.
transcriptLabvjust	Vertical adjustment of label for each transcript. DEFAULT = 1.5. OPTIONAL.
boxedlabels	Logical, indicating whether or not to draw labels in boxes. DEFAULT = FALSE. OPTIONAL.

shape	Shape of the plotted points. Either a single value for all points, or 4 values corresponding to $\langle \text{abs}(\text{FCcutoff}) \ \&\& \ > \text{pCutoff}, \ > \text{abs}(\text{FCcutoff}), \ < \text{pCutoff}, \ > \text{abs}(\text{FCcutoff}) \ \&\& \ < \text{pCutoff}$. DEFAULT = 19. OPTIONAL.
shapeCustom	Named vector / key-value pairs that will over-ride the default shape scheme. The order must match that of toptable. Names / keys relate to groups / categories; values relate to shape encodings. DEFAULT = NULL. OPTIONAL.
col	Colour shading for plotted points, corresponding to $\langle \text{abs}(\text{FCcutoff}) \ \&\& \ > \text{pCutoff}, \ > \text{abs}(\text{FCcutoff}), \ < \text{pCutoff}, \ > \text{abs}(\text{FCcutoff}) \ \&\& \ < \text{pCutoff}$. DEFAULT = <code>c("grey30", "forestgreen", "royalblue", "red2")</code> . OPTIONAL.
colCustom	Named vector / key-value pairs that will over-ride the default colour scheme. The order must match that of toptable. Names / keys relate to groups / categories; values relate to colour. DEFAULT = NULL. OPTIONAL.
colAlpha	Alpha for purposes of controlling colour transparency of transcript points. DEFAULT = 1/2. OPTIONAL.
legend	Plot legend text. DEFAULT = <code>c("NS", "Log2 FC", "P", "P & Log2 FC")</code> . OPTIONAL.
legendPosition	Position of legend ("top", "bottom", "left", "right"). DEFAULT = "top". OPTIONAL.
legendLabSize	Size of plot legend text. DEFAULT = 14. OPTIONAL.
legendIconSize	Size of plot legend icons / symbols. DEFAULT = 4.0. OPTIONAL.
legendVisible	Logical, indicating whether or not to show the legend. DEFAULT = TRUE. OPTIONAL.
shade	A vector of transcript names to shade. DEFAULT = NULL. OPTIONAL.
shadeLabel	Label for the transcripts to shade. DEFAULT = NULL. OPTIONAL.
shadeAlpha	Alpha for purposes of controlling colour transparency of shaded regions. DEFAULT = 1/2. OPTIONAL.
shadeFill	Colour of shaded regions. DEFAULT = "grey". OPTIONAL.
shadeSize	Size of the shade contour lines. DEFAULT = 0.01. OPTIONAL.
shadeBins	Number of bins for the density of the shade. DEFAULT = 2. OPTIONAL.
drawConnectors	Logical, indicating whether or not to connect plot labels to their corresponding points by line connectors. DEFAULT = FALSE. OPTIONAL.
widthConnectors	Line width of connectors. DEFAULT = 0.5. OPTIONAL.
typeConnectors	Have the arrow head open or filled ('closed')? ('open', 'closed'). DEFAULT = 'closed'. OPTIONAL.
endsConnectors	Which end of connectors to draw arrow head? ('last', 'first', 'both'). DEFAULT = 'first'. OPTIONAL.
lengthConnectors	Length of the connectors. DEFAULT = <code>unit(0.01, 'npc')</code> . OPTIONAL.
colConnectors	Line colour of connectors. DEFAULT = 'grey10'. OPTIONAL.
hline	Draw one or more horizontal lines passing through this/these values on y-axis. For single values, only a single numerical value is necessary. For multiple lines, pass these as a vector, e.g., <code>c(60,90)</code> . DEFAULT = NULL. OPTIONAL.
hlineType	Line type for hline ('blank', 'solid', 'dashed', 'dotted', 'dotteddash', 'longdash', 'twodash'). DEFAULT = 'longdash'. OPTIONAL.

<code>hlineCol</code>	Colour of hline. DEFAULT = 'black'. OPTIONAL.
<code>hlineWidth</code>	Width of hline. DEFAULT = 0.4. OPTIONAL.
<code>vline</code>	Draw one or more vertical lines passing through this/these values on x-axis. For single values, only a single numerical value is necessary. For multiple lines, pass these as a vector, e.g., <code>c(60,90)</code> . DEFAULT = NULL. OPTIONAL.
<code>vlineType</code>	Line type for vline ('blank', 'solid', 'dashed', 'dotted', 'dotdash', 'longdash', 'twodash'). DEFAULT = 'longdash'. OPTIONAL.
<code>vlineCol</code>	Colour of vline. DEFAULT = 'black'. OPTIONAL.
<code>vlineWidth</code>	Width of vline. DEFAULT = 0.4. OPTIONAL.
<code>gridlines.major</code>	Logical, indicating whether or not to draw major gridlines. DEFAULT = TRUE. OPTIONAL
<code>gridlines.minor</code>	Logical, indicating whether or not to draw minor gridlines. DEFAULT = TRUE. OPTIONAL
<code>border</code>	Add a border for just the x and y axes ('partial') or the entire plot grid ('full')? DEFAULT = 'partial'. OPTIONAL.
<code>borderWidth</code>	Width of the border on the x and y axes. DEFAULT = 0.8. OPTIONAL.
<code>borderColour</code>	Colour of the border on the x and y axes. DEFAULT = "black". OPTIONAL.

Details

Volcano plots represent a useful way to visualise the results of differential expression analyses. Here, we present a highly-configurable function that produces publication-ready volcano plots [[@EnhancedVolcano](#)]. `EnhancedVolcano` will attempt to fit as many transcript names in the plot window as possible, thus avoiding 'clogging' up the plot with labels that could not otherwise have been read.

Value

A `ggplot2` object.

Author(s)

Kevin Blighe <kevin@clinicalbioinformatics.co.uk>

Examples

```
library("pasilla")
pasCts <- system.file("extdata", "pasilla_gene_counts.tsv",
  package="pasilla", mustWork=TRUE)
pasAnno <- system.file("extdata", "pasilla_sample_annotation.csv",
  package="pasilla", mustWork=TRUE)
cts <- as.matrix(read.csv(pasCts, sep="\t", row.names="gene_id"))
coldata <- read.csv(pasAnno, row.names=1)
coldata <- coldata[,c("condition", "type")]
rownames(coldata) <- sub("fb", "", rownames(coldata))
cts <- cts[, rownames(coldata)]
library("DESeq2")
dds <- DESeqDataSetFromMatrix(countData = cts,
  colData = coldata,
  design = ~ condition)
```

```
featureData <- data.frame(gene=rownames(cts))
mcols(dds) <- DataFrame(mcols(dds), featureData)
dds <- DESeq(dds)
res <- results(dds)

EnhancedVolcano(res,
  lab = rownames(res),
  x = "log2FoldChange",
  y = "pvalue",
  pCutoff = 10e-4,
  FCcutoff = 1.333,
  xlim = c(-5.5, 5.5),
  ylim = c(0, -log10(10e-12)),
  transcriptPointSize = 1.5,
  transcriptLabSize = 2.5,
  shape = c(6, 6, 19, 16),
  title = "DESeq2 results",
  subtitle = "Differential expression",
  caption = "FC cutoff, 1.333; p-value cutoff, 10e-4",
  legendPosition = "right",
  legendLabSize = 14,
  col = c("grey30", "forestgreen", "royalblue", "red2"),
  colAlpha = 0.9,
  drawConnectors = TRUE,
  hline = c(10e-8),
  widthConnectors = 0.5)
```

Index

EnhancedVolcano, [2](#)
EnhancedVolcano-package, [2](#)
ggplot2, [6](#)