

# Package ‘rnaseq’

September 21, 2024

**Title** A Package for Rna-seq Analysis

**Version** 0.1.1

**Description** This package provides tools for Rna-seq analysis, enabling users to process and visualize RNA sequencing data effectively.

**License** MIT + file LICENSE

**Encoding** UTF-8

**Roxygen** list(markdown = TRUE)

**RoxygenNote** 7.3.2

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

**Config/testthat/edition** 3

**VignetteBuilder** knitr

**Depends** R (>= 2.10)

**LazyData** true

**LinkingTo** Rcpp

**Imports** dplyr,  
forcats,  
ggplot2,  
magrittr,  
Rcpp

## Contents

enrichment_barplot . . . . .	2
enrichment_dotplot . . . . .	3
fancy_print . . . . .	5

<b>Index</b>	<b>6</b>
--------------	----------

---

enrichment_barplot	<i>Enrichment Analysis Bar Plot</i>
--------------------	-------------------------------------

---

## Description

This function creates a bar plot for gene enrichment analysis results, displaying the counts or gene ratios of enriched terms across different ontology categories. The plot is faceted by ontology categories, and the number of items displayed per facet can be limited using the `topn` parameter.

## Usage

```
enrichment_barplot(
  data,
  output_file = NULL,
  width = 6,
  height = 8,
  dpi = 300,
  topn = 10,
  use_gene_ratio = FALSE
)
```

## Arguments

<code>data</code>	<p>A data frame containing the enrichment analysis results. It must include the following columns:</p> <p><b>Description</b> Character vector, descriptions of the enriched terms.</p> <p><b>GeneRatio</b> Numeric vector, gene ratios (optional if <code>use_gene_ratio</code> is FALSE).</p> <p><b>Count</b> Numeric vector, counts of genes in each term (optional if <code>use_gene_ratio</code> is TRUE).</p> <p><b>p.adjust</b> Numeric vector, the adjusted p-values for multiple testing.</p> <p><b>ONTOLOGY</b> Character vector, ontology categories, such as "Biological Process", "Molecular Function", or "Cellular Component".</p>
<code>output_file</code>	A string specifying the file path to save the plot. If provided, the plot will be saved to this file. Default is NULL.
<code>width</code>	Numeric, the width of the saved plot in inches. Default is 6.
<code>height</code>	Numeric, the height of the saved plot in inches. Default is 8.
<code>dpi</code>	Numeric, the resolution of the saved plot in dots per inch. Default is 300.
<code>topn</code>	Integer, the maximum number of items to display per ontology category. If an ontology category has more than <code>topn</code> items, only the items with the lowest <code>p.adjust</code> values will be displayed. Default is 10.
<code>use_gene_ratio</code>	Logical, whether to use <code>GeneRatio</code> for the x-axis. If FALSE, <code>Count</code> will be used. Default is FALSE.

## Details

The function uses `ggplot2` to create a faceted bar plot of enrichment results. By default, it plots the `Count` of genes in each enriched term on the x-axis. Setting `use_gene_ratio = TRUE` will plot the `GeneRatio` instead. The bars are ordered within each facet based on the `p.adjust` values.

The topn parameter allows you to limit the number of terms displayed per ontology category to the most significant ones based on the adjusted p-values.

When output\_file is provided, the plot is saved to the specified file, and the function returns NULL. Otherwise, the ggplot object is returned for further customization or immediate display.

### Value

If output\_file is provided, the function saves the plot to the specified path and returns NULL. Otherwise, it returns the ggplot object.

### Examples

```
# Create example data frame with necessary columns
df <- data.frame(
  Description = paste0("Term", 1:15),
  GeneRatio = runif(15, 0.1, 0.3),
  Count = sample(10:50, 15, replace = TRUE),
  p.adjust = 10^(-runif(15, 1, 5)),
  ONTOLOGY = rep(c("Biological Process", "Molecular Function", "Cellular Component"), each = 5)
)
# Generate bar plot without saving to file
p <- enrichment_barplot(df, topn = 3)
```

---

enrichment_dotplot	<i>Enrichment Analysis Dot Plot</i>
--------------------	-------------------------------------

---

### Description

This function creates a dot plot for gene enrichment analysis results, displaying the gene ratios or counts of enriched terms across different ontology categories. The plot is faceted by ontology categories, and the number of items displayed per facet can be limited using the topn parameter.

### Usage

```
enrichment_dotplot(
  data,
  output_file = NULL,
  width = 6,
  height = 8,
  dpi = 300,
  topn = 10,
  low_color = "blue",
  high_color = "red",
  use_gene_ratio = TRUE
)
```

### Arguments

**data** A data frame containing the enrichment analysis results. It must include the following columns:

**Description** Character vector, descriptions of the enriched terms.

	<b>GeneRatio</b> Numeric vector, gene ratios (optional if <code>use_gene_ratio</code> is FALSE).
	<b>Count</b> Numeric vector, counts of genes in each term (optional if <code>use_gene_ratio</code> is TRUE).
	<b>p.adjust</b> Numeric vector, the adjusted p-values for multiple testing.
	<b>ONTOLOGY</b> Character vector, ontology categories, such as "Biological Process", "Molecular Function", or "Cellular Component".
<code>output_file</code>	A string specifying the file path to save the plot. If provided, the plot will be saved to this file. Default is NULL.
<code>width</code>	Numeric, the width of the saved plot in inches. Default is 6.
<code>height</code>	Numeric, the height of the saved plot in inches. Default is 8.
<code>dpi</code>	Numeric, the resolution of the saved plot in dots per inch. Default is 300.
<code>topn</code>	Integer, the maximum number of items to display per ontology category. If an ontology category has more than <code>topn</code> items, only the items with the lowest <code>p.adjust</code> values will be displayed. Default is 10.
<code>low_color</code>	A string specifying the color representing low values in the color gradient. Default is "blue".
<code>high_color</code>	A string specifying the color representing high values in the color gradient. Default is "red".
<code>use_gene_ratio</code>	Logical, whether to use <code>GeneRatio</code> for the x-axis. If FALSE, <code>Count</code> will be used. Default is TRUE.

## Details

The function uses `ggplot2` to create a faceted dot plot of enrichment results. By default, it plots the `GeneRatio` on the x-axis. Setting `use_gene_ratio = FALSE` will plot the `Count` instead. The points are ordered within each facet based on the `p.adjust` values.

The `topn` parameter allows you to limit the number of terms displayed per ontology category to the most significant ones based on the adjusted p-values.

When `output_file` is provided, the plot is saved to the specified file, and the function returns NULL. Otherwise, the `ggplot` object is returned for further customization or immediate display.

## Value

If `output_file` is provided, the function saves the plot to the specified path and returns NULL. Otherwise, it returns the `ggplot` object.

## Examples

```
# Create example data frame with necessary columns
df <- data.frame(
  Description = paste0("Term", 1:15),
  GeneRatio = runif(15, 0.1, 0.3),
  Count = sample(10:50, 15, replace = TRUE),
  p.adjust = 10^(-runif(15, 1, 5)),
  ONTOLOGY = rep(c("Biological Process", "Molecular Function", "Cellular Component"), each = 5)
)
# Generate dot plot without saving to file
p <- enrichment_dotplot(df, topn = 3)
print(p)
```

---

fancy\_print

---

*Fancy Print Messages***Description**

Prints messages with fancy decorations.

**Usage**

```
fancy_print(messages)
```

**Arguments**

messages      A character vector of messages to print.

**Value**

None. Messages are printed to the console.

**Examples**

```
fancy_print(c("Hello", "World"))
```

# Index

- \* **barplot**
  - enrichment\_barplot, [2](#)
- \* **dotplot**
  - enrichment\_dotplot, [3](#)
- \* **enrichment**
  - enrichment\_barplot, [2](#)
  - enrichment\_dotplot, [3](#)
- \* **gene**
  - enrichment\_barplot, [2](#)
  - enrichment\_dotplot, [3](#)
- \* **ontology**
  - enrichment\_barplot, [2](#)
  - enrichment\_dotplot, [3](#)
- \* **plot**
  - enrichment\_barplot, [2](#)
  - enrichment\_dotplot, [3](#)

enrichment\_barplot, [2](#)  
enrichment\_dotplot, [3](#)

fancy\_print, [5](#)