# CircoTax tutorial

Leandro Di Gloria, Lorenzo Casbarra and Matteo Ramazzotti

- **What is CircoTax?**

CircoTax is an R function implementing a specialized ggplot2 graph to represent in a radial form a rank-aware collection of differentially abundant taxa. Specifically, the CircoTax radial bar plot shows 6 or 7 sectors that encode the taxonomy depth (from kingdom to genus) and, departing from the center, a number of radial bars that reach the appropriate sector and whose color and transparency are proportional to the log fold change intensity and direction. Although the function is designed to work on an R phyloseq matrix structure, the matrix can be manually created (see the format) and plotted using the dedicated "manual" version of CircoTax. Moreover, our implementation further integrates a simple front end for differential abundance analysis able to provide inputs for the CircoTax. The visualization of results is both graphically appealing, allowing to accommodate tens of differentially abundant ranks, and biologically informative since the amount of variation, the direction and rank are easily intelligible for each differentially represented taxa.

- **How to install**

CircoTax is hosted at https://github.com/matteoramazzotti/CircoTax .
It and can be downloaded as CircoTax.zip from the GitHub interface or by using the following git command:

```
git clone https://github.com/matteoramazzotti/CircoTax
```

- **Required dependencies**

The CircoTax functions are distributed as plain text files ready to be imported (sourced) in the R environment. Despite this, it has two main dependencies, namely ggplot2 and ggh4x.

The function "CircoTax_DESeq2" further requires the package phyloseq. Optionally it can require the packages apeglm and ashr.

The function "CircoTax_ALDEx2" only requires the package ALDEx2.

CircoTax and the related functions have been tested using R 4.3, ggplot2 3.4.4, ggh4x 0.2.7, ALDEx2 1.34 and DESeq2 1.42. However, to the best of our knowledge, no specific versions of those dependencies are required to run those scripts.

- **How to plot a CircoTax using a complete taxonomy matrix**

CircoTax input matrix structure is exemplified as reported below. It must be encoded as an R valid data.frame. The FoldChange column MUST be present and the first non NA valid taxonomic name available will be plotted. The following represents a CircoTax valid input:

| FoldChange | Phylum | Class | Order | Family | Genus |
|---|---|---|---|---|---|
| 1.06736 | Proteobacteria | Gammaproteobacteria | NA | NA | NA |
| 2.398398 | Bacteroidota | Bacteroidia | Bacteroidales | Tannerellaceae | NA |
| 3.136581 | Bacteroidota | Bacteroidia | Bacteroidales | Marinifilaceae | NA |
| 3.242582 | Proteobacteria | Gammaproteobacteria | Burkholderiales | Burkholderiaceae | NA |
| 3.808097 | Proteobacteria | Gammaproteobacteria | Burkholderiales | Sutterellaceae | NA |
| 3.624271 | Firmicutes | Bacilli | Lactobacillales | Lactobacillaceae | NA |
| -5.55057 | Bacteroidota | Bacteroidia | Flavobacteriales | Flavobacteriaceae | Capnocytophaga |
| 3.172175 | Bacteroidota | Bacteroidia | Bacteroidales | Prevotellaceae | Alloprevotella |
| 10 | Bacteroidota | Bacteroidia | Bacteroidales | Muribaculaceae | Muribaculaceae |
| 4.38531 | Bacteroidota | Bacteroidia | Bacteroidales | Marinifilaceae | Butyricimonas |
| 8.494905 | Proteobacteria | Gammaproteobacteria | Burkholderiales | Sutterellaceae | Sutterella |
| 3.313675 | Firmicutes | Negativicutes | Veillonellales-Selenomonadales | Veillonellaceae | Dialister |
| 4.235196 | Firmicutes | Clostridia | Clostridia vadinBB60 | Clostridia vadinBB60 | Clostridia vadinBB60 |

The CircoTax function requires at least an R data.frame as input and the plot is generated using the syntax:

```
CircoTax( data.frame_name_here )
```

Moreover, the following arguments can be specified:

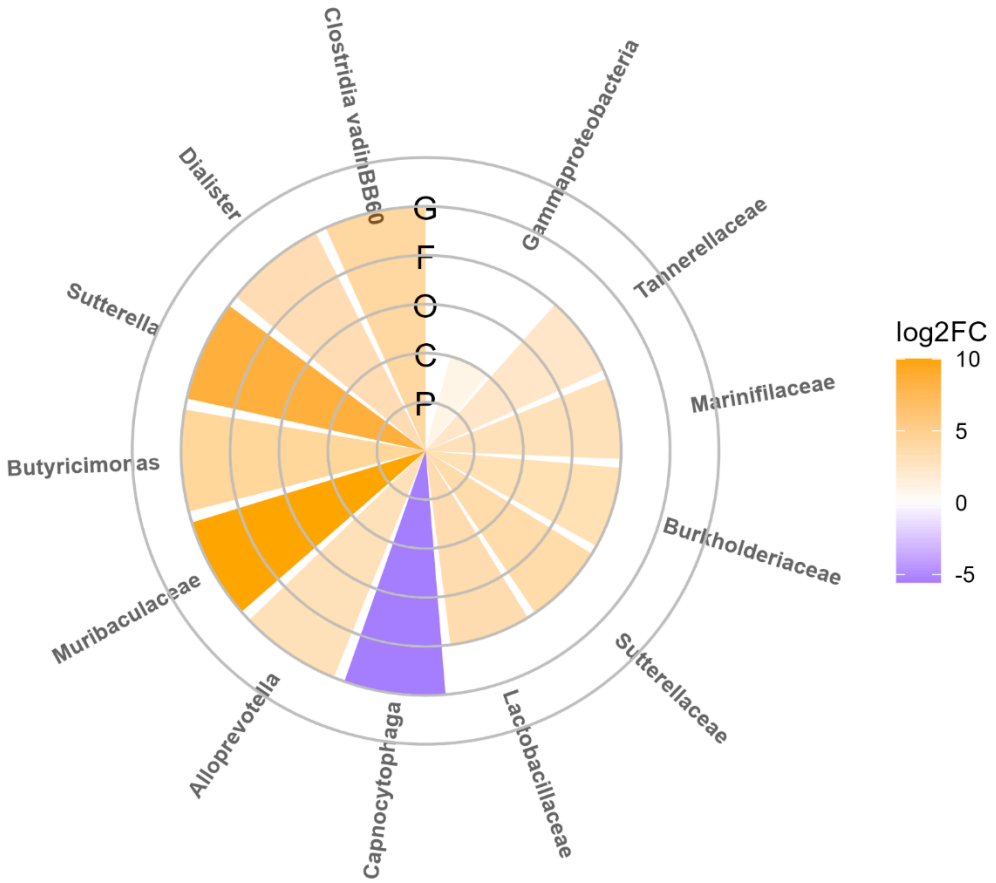
| Argument | Description |
|---|---|
| title | Title to display on the top of the plot. |
| fill_text | Text to display above the color legend. |
| fc_col | Column index of the input table containing the fold change values (an as.numeric column) to display through the gradient of the column. By default, the fold change values are searched in the first column of the input matrix. |
| tax_col | Column indexes of the input table containing the full taxonomic path of each result. By default, the column indexes range from the second column to the last column. The function will automatically infer the presence or absence of certain taxonomic levels according to the number of columns. |
| sort | Sorting logic of the results in the CircoTax. The possible inputs are “rank” (by taxonomic rank), “fc” (by fold change), “absfc” (by absolute fold change) and “alpha” (alphabetic order). |
| ramp | Gradient of colors corresponding to fold change values range. Its input has to be a three character vector, where each character is a R color name. Defaults to c(“blue”,”white”,”orange”) |
| size_taxon_circo | Size of the taxa labels (default=3). |

Excluding the name of the input data frame, each parameters is optional. The default values of the arguments fc_col and tax_col are 1 and 2:6 and should be overwritten by the user according to their actual position. Accordingly, the function can be also written as follows:

```
CircoTax(table_name, title="plot" , ramp=c("orange","white","blue"), tax_col=2:6, fc_col=1, sort="rank")
```

In addition, being this function based on ggplot2, the CircoTax settings may be further customized by adding the usual ggplot2 functions (e.g. " labs(fill='legend_name_here') " ). However, some bugs may be encountered if the function code itself is overwritten by adding some custom ggplot2 function.

Using the function written above on this table, the following CircoTax plot will be obtained.




In this example table there are seven results with genus level resolution (seven differential abundant genera), five results with family level resolution and one with class level resolution.
In fact, the function will automatically plot the name of the last level before a NA value (same format as the phyloseq standard taxonomic table). Accordingly, even if usually the full taxonomic path is available when working with a phyloseq object, knowing the name of each level is not required.
For example, the following row is accepted to display the Tannerellaceae family as result:

2.398398 none none none Tannerellaceae NA

This feature is meant to support the user in easily building an own table to plot also without a complete taxonomy path, but in this case the "manual" approach below may be preferred.

- **How to plot a CircoTax using a custom data frame**

The “CircoTax_custom” function allows to plot a CircoTax from an R data.frame composed of three columns, namely taxon name, taxonomic rank name and value to display (e.g. FC).

The following represents a CircoTax valid input:

| Tax_name | Rank_name | FC |
|---|---|---|
| Gammaproteobacteria | Class | 1.06736 |
| Tannerellaceae | Family | 2.398398 |
| Marinifilaceae | Family | 3.136581 |
| Sutterellaceae | Family | 3.808097 |
| Lactobacillaceae | Family | 3.624271 |
| Capnocytophaga | Genus | -5.55057 |
| Alloprevotella | Genus | 3.172175 |
| Muribaculaceae | Genus | 10 |
| Butyricimonas | Genus | 4.38531 |
| Clostridia vadinBB60 | Genus | 4.235196 |

The first row of the table (here “*Tax_name*, *Rank_name*, and *FC”*) must be assigned as column names in R and will therefore not be displayed. These are primarily for explanatory purposes and are not strictly required, as the CircoTax_custom function will extract characters or values according to their column index. Notably, this function can also generate a CircoTax without relying on actual taxonomies or bacterial names. The characters and values provided in the data frame are utilized directly as written. This flexibility enables the plotting of alternative biological datasets while maintaining the distinctive CircoTax aesthetic.

The “CircoTax_custom” function has the following basicr syntax:

```
CircoTax_custom(dataframe_name_here)
```

Moreover, the following arguments can also be specified:

| Argument | Description |
|---|---|
| title | Title to display on the top of the plot. |
| fill_text | Text to display above the color legend. |
| ramp | Gradient of colors corresponding to fold change values range.<br>Its input has to be a three character vector, where each character is a R color name. |
| name | Column index of the input table containing the taxa names. By default, the first column is used. The names order is determined by the input rows order. |
| tax_col | Column index of the input table containing the taxonomic rank. By default, the second column is used. The rank order is determined by the input rows order. |
| fc_col | Column index of the input table containing the fold change rank. By default, the second column is used. |
| size_taxon_circo | Size of the taxa labels (default=3). |

The arguments tax_col and fc_col should be overwritten by the user according to the current input structure.

- **Introduction to Auto DA functions**

Along with the CircoTax function, two automatic differential analysis (Auto DA) ancillary functions, called "CircoTax_DESeq2" and "CircoTax_ALDEx2", are available. By default, those functions perform the differential analysis at each taxonomic level between two groups using either DESeq2 or ALDEx2. They then filter the significant results and finally generate as outputs a tsv table, a box plot of percent abundances and a CircoTax plot of the results. Proving both algorithms through the respective functions, the user has the opportunity to choose the approach which he considers more suitable for the analysed data.

The CircoTax_ALDEx2 analysis is based on the parametric version of ALDEx2 but the user can also specify a custom model design to perform the analysis on ranks. Moreover, the behaviour of those underlying algorithms can be easily modified by using simple arguments, e.g. the ALDEx2 package does not use the BH correction by itself (at least until version 1.34) but the related AutoDA function use BH corrected p-value (if not stated otherwise during the function call).

The associated boxplot can be plotted aside the CircoTax to display also the abundances of the resulting taxa. By default, the abundances in the box plot undergo a square root transformation to enhance the visualization of lower counts. However, being this transformation may be misleading in case of decimal counts, users are given the option to disable it if needed.

Their basic syntax of both the functions is the following:

```
CircoTax_XXXX( input_phyloseq = phyloseq_name, contrast = c("Factor","BaseLevel","OtherLevel") )
```

The pattern XXXX in this example is a placeholder for "DESeq2" or "ALDEx2".
The argument "input_phyloseq" requires a phyloseq object with OTU table, tax table and sample data.
Considering that DESeq2 and ALDEx2 packages operate their own transformations, we suggest using the raw absolute counts to avoid loss of performances or bugs related to those packages.

The column names in the tax table of the phyloseq object have to be "Phylum", "Class", "Order", "Family" and "Genus" (e.g. not "Families") to allow the Auto DA functions to automatically perform the aggregations of counts along the taxonomic levels.

The argument "contrast" requires a vector of three characters which are (in this order) the names of the interest factor and its two levels of which report the differences (e.g. BaseLevel versus OtherLevel, or level A versus level B, or Healthy versus Disease, or Male versus Female etc.). The factors and the levels name are searched in the sample data of the phyloseq object in input. For example, the user may write the following command to perform the analysis between Healthy subjects and CRC patients, where such sample classification is indicated in the column "Condition" of the sample data.

```
CircoTax_XXXX( input_phyloseq = phyloseq_name, contrast = c("Condition","Healthy","CRC") )
```

Beside these two mandatory inputs, no additional arguments are required. The functions will automatically apply their default settings to perform analysis, generating both a boxplot and a CircoTax plot.
By default, a complete list of applied settings is displayed after each analysis. Additionally, settings that

could influence the results are documented in a CSV file named “CircoTax_Settings”. The CSV file with the results and the plots are also saved as separate files in the working directory (by default) called “results”, “CircoTax_plot” and “boxplot”.

- **Auto DA advanced settings**

The functions “CircoTax_DESeq2” and “CircoTax_ALDEx2” are designed to be easily handled in R environment, then many parameters regarding the analysis (computed using DESeq2 or ALDEx2 programs) and the plots are pre-established. However, the following settings are available to customize the analysis:

| Argument | Default | Description |
|---|---|---|
| Design | Based on contrast | Statistical design (within the limits of DESeq2 or Aldex2) on which the analysis is conducted. Its input has to be a vector of a single character (a string), for example, design='~ Gender+Condition' . |
| p_value | 0.05 | p-value threshold (after the multiple test adjustment) to consider a result as significant. |
| p_adjustment | BH | Multiple test adjustment by adjusting (penalize) each p-value. The possible inputs are “BH” (Benjamini-Hochberg) and “holm” (Holm). |
| lfc | 1 | Log fold change threshold (only for DESeq2). |
| lfc_shrink_method | None | Shrinkage method of the log fold change. The possible inputs are “none”, “apeglm” and “ashr” (only for DESeq2). |
| B | 50 | DESeq2 base-mean threshold (mean after DESeq2 values transformation) under which exclude results (only for DESeq2). |
| eff_size | 0 | ALDEx2 effect size threshold under which exclude results (only for ALDEx2). |
| MCS | 128 | Monte-Carlo samples to generate during ALDEx2 computations (only for ALDEx2). |
| remove_redundants | TRUE | Enable the automatic removal of redundant results (e.g. result repeated also in higher taxonomic levels being the only observation in that taxonomic clade). |
| remove_results | (empty) | Results which have not been displayed. Its input has to be a vector whose characters are the bacteria name to remove. |
| taxout | (empty) | Taxonomic level to exclude. Its input has to be a vector whose characters are the taxonomic level to remove. |
| W | 10 | Width of the box plot, in inches. |
| H | 7 | Height of the box plot, in inches. |
| COLOR_A | coral | Color of the tested level box in the box plot. |
| COLOR_B | chartreuse | Color of the reference level box in the box plot. |
| sqrt_y_axis | TRUE | The Y-axis ticks in the boxplot increase following a square root scale to improve the readability of lower abundances. NB: this option may be misleading on decimal counts. |
| plot_boxplot | TRUE | Enable the generation of the box plot. |
| plot_circo | TRUE | Enable the generation of the CircoTax. |
| sort_circo | rank | Sorting logic of the results in the CircoTax. The possible inputs are “rank” (by taxonomic rank), “fc” (by fold change), “absfc” (by absolute fold change) and “alpha” (alphabetic). |
| size_taxon_circo | 3 | Size of the taxa labels. |
| format_image | png | Format of the files to which save the plots. The possible inputs are “png” and “pdf”. |
| save_path | (working dir) | Path where to save the tables and figures. |

| | | |
|---|---|---|
| auto_save | TRUE | Enable the export of the results on the PC. By disabling this option, tables and plots will be returned as object in the R environment. |
| auto_log | TRUE | Enable the displaying of each of the advanced settings with default values as messages in the R console at the end of the analysis. |

- **Auto DA automatic filters and cleaning**

The Auto DA functions internally performs the following adjustments to improve the process speed and results readability:

- each observation with total sum absolute abundance among samples below the value of 10 is discarded before the analysis;
- during the aggregation of counts, the NA observation are not discarded (NArm=F in the phyloseq function tax_glom);
- result repeated also in higher taxonomic levels because they are only observation in that taxonomic clade are seen as “redundant” and then removed regardless their significance (if not stated otherwise during the function call);
- genera labelled as “uncultured” or “NA” are renamed using their family name if available, e.g. an uncultured genus of Tannerellaceae family will be renamed “uncultured_f_Tannerellaceae” to be discernible from other uncultured genera.