omicplotR: A tool for visualization of omic datasets as compositions

Daniel Giguere

2017-11-21

Contents

1 What is omicplotR?

`omicplotR` is an R package containing a `Shiny` app used to visually explore omic datasets, where the input is a table of read counts from high-throughput sequencing runs. It integrates the `ALDEx2`[1] package for compositional analysis of differential abundance. `omicplotR` is intended to speed up the process of visually exploring high-throughput sequencing datasets by providing an easy to use graphical user interface for users with and without experience in R.

2 Introduction

High-throughput sequencing (HTS) instruments generate an amount of reads that is constrained by limitations of the sequencing instrument itself, and do not represent the absolute number of DNA molecules in a sample. For example, an Illumina NextSeq can deliver up to 400 million single-end reads, whereas an Illumina MiSeq2 can only deliver up to 15 million single-end reads[2]. This type of data, which is constrained by an arbitrary or constant sum, is referred to as compositional data, and high-throughput sequencing data must be treated

[1] Fernandes et al (2013) PLOS ONE https://doi.org/10.1371/journal.pone.0067019

[2] https://www.illumina.com/systems/sequencing-platforms/nextseq/specifications.html

as such[3]. omicplotR incorporates ALDEx2 for compositional differential abundance analysis (see ALDEx2 for more information).

Although several R packages exist for exploring high-throughput sequencing data, they are typically command line based, which presents a barrier for users without any significant command line or scripting experience. omicplotR was created to facilitate the exploratory phase of high-throughput sequencing data analysis by allowing generating basic exploratory plots automatically with adjustable features and filters.

This vignette provides an overview of the R package omicplotR and the input requirements. A tutorial for each component of the Shiny app is available on my GitHub: https://github.com/dgiguer/omicplotR/wiki. omicplotR was developed for several types of HTS datasets including RNASeq, meta-RNASeq, and 16s rRNA gene sequencing, and in principle, can be used for nearly any type of data generated by HTS which generates a tables of counts per feature for each sample.

3 Features

omicplotR provides a graphical user interface using the Shiny package for the following visualizations for HTS data:

- Compositional Principal Component Analysis (PCA) biplots
- Dendrograms
- Stacked barplots of relative taxonomic abundance
- Compositional differential abundance analysis

Additional features include:

- Filtering count tables per sample or feature by counts
- Filtering data into groups using metadata for plotting
- Colour PCA biplots using metadata (continuously, by quartile, categorical)
- Generate effect plots between conditions of associated metadata using ALDEx2
- Interactive effect plots to visualize difference between and within groups
- Plot pre-calculated ALDEx2 tables to colour points by rownames for large datasets

4 Installation and example

Install the latest version of omicplotR. Make sure you have the newest version of R, ALDEx2, and other dependancies. omicplotR requires you to have at least R version 3.4.

First, load the omicplotR package. All other dependencies will be loaded automatically. This will launch the Shiny app in your default browser. For this vignette, we will be using the example data and metadata provided.

```r
library(omicplotR)
omicplotr.run()
```

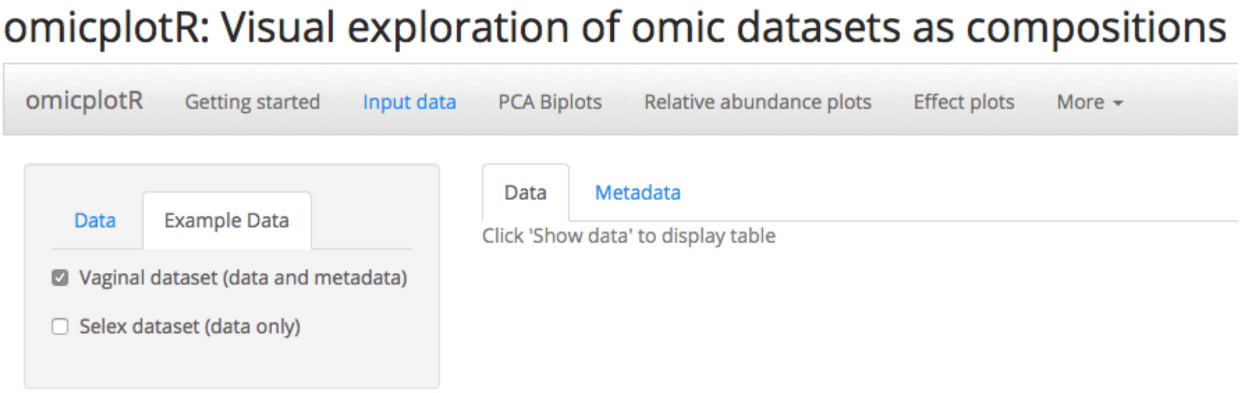
After launching the Shiny app, click the 'Input data' tab to get started.

5 Input data

The 'Data' tab on the sidebar panel (grey) allows you to choose your own data and metadata by clicking 'Browse'. To follow along with this vignette, please click the 'Example data' tab on the sidebar panel, and

[3] Gloor et al (2017) Front. Microbiol. https://doi.org/10.3389/fmicb.2017.02224

click the checkbox for the ‘Vaginal dataset’. This dataset, which includes associated metadata, is from a study that characterized the changes in the vaginal microbiome following antibiotic and probiotic treatment by 16s rRNA gene sequencing[4]. Return to the ‘Data’ tab on the sidebar panel to view the data and metadata by clicking ‘Show data’ and ‘Show metadata’. The tabs on the main panel allow you to switch between displaying your data and metadata tables.



omicplotR: Visual exploration of omic datasets as compositions

☑ Vaginal dataset (data and metadata)

☐ Selex dataset (data only)

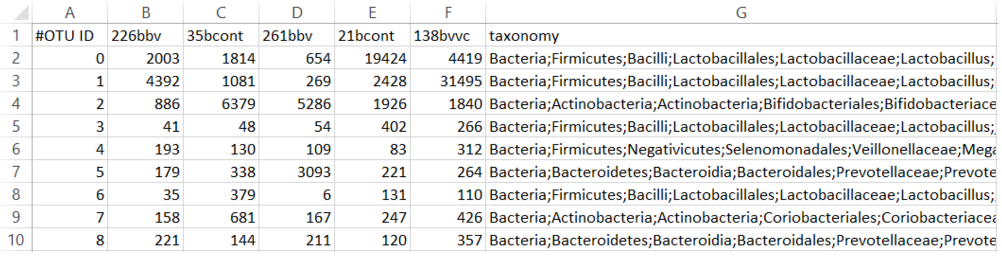
Click 'Show data' to display table

Figure 1: Screenshot of input data page. The ’Example data’ tab on the sidebar panel provides access to the provided datasets within the Shiny app.

5.1 Data

When choosing your own data set, input requirements are as follows: for both metadata and data, each sample and feature name (operational taxonomic unit - OTU) **must be unique**. An example of an appropriately formatted data file is shown in Figure 2.

1. The data file must be a **tab delimited .txt** (this is an option when you click ‘Save as’ from Excel, or when writing to a table in R).
2. The first column must contain feature/OTU identifier. In this case, they are just labelled as numbers.
3. The first row must contain sample identifiers.
4. The last column may contain taxonomic level information, but is not required. If present, it must be labelled exactly ‘taxonomy’. **The taxonomy column must have at least four levels, separated by a semi colon or colon.**
5. Data table must have all blank rows removed (this may require you to check in a text editor like Notepad ++ or Atom before using the app). This should be especially checked if you are using a Windows based computer.



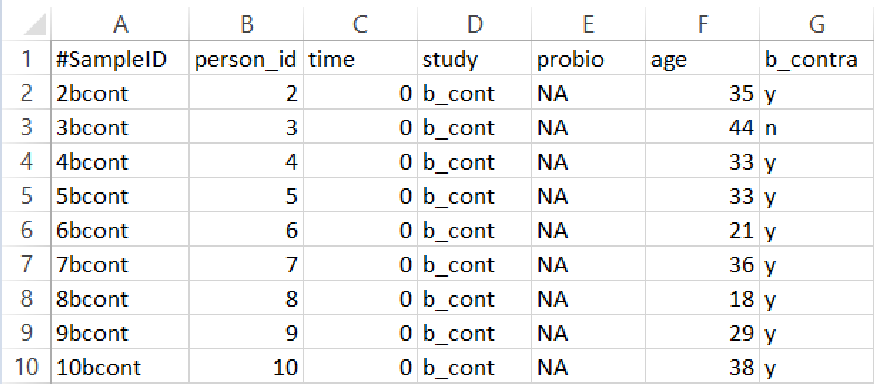
	A	B	C	D	E	F	G
1	#OTU ID	226bbv	35bcont	261bbv	21bcont	138bvvc	taxonomy
2	0	2003	1814	654	19424	4419	Bacteria;Firmicutes;Bacilli;Lactobacillales;Lactobacillaceae;Lactobacillus;
3	1	4392	1081	269	2428	31495	Bacteria;Firmicutes;Bacilli;Lactobacillales;Lactobacillaceae;Lactobacillus;
4	2	886	6379	5286	1926	1840	Bacteria;Actinobacteria;Actinobacteria;Bifidobacteriales;Bifidobacteriace
5	3	41	48	54	402	266	Bacteria;Firmicutes;Bacilli;Lactobacillales;Lactobacillaceae;Lactobacillus;
6	4	193	130	109	83	312	Bacteria;Firmicutes;Negativicutes;Selenomonadales;Veillonellaceae;Mega
7	5	179	338	3093	221	264	Bacteria;Bacteroidetes;Bacteroidia;Bacteroidales;Prevotellaceae;Prevote
8	6	35	379	6	131	110	Bacteria;Firmicutes;Bacilli;Lactobacillales;Lactobacillaceae;Lactobacillus;
9	7	158	681	167	247	426	Bacteria;Actinobacteria;Actinobacteria;Coriobacteriales;Coriobacteriace
10	8	221	144	211	120	357	Bacteria;Bacteroidetes;Bacteroidia;Bacteroidales;Prevotellaceae;Prevote

Figure 2: Example data. If taxonomy column is present, it must use the column name ’taxonomy’. Image taken from modified version of Vaginal dataset.

[4] Macklaim et al (2015) Microb. Ecol. Heal. Dis. https://doi.org/10.3402/mehd.v26.27799.

Your metadata file must follow the following format. An example of an appropriate metadata file is shown in Figure 3.

1. Must be a **tab delimited .txt** (this is an option when you click 'Save as' from Excel).
2. The first column must contain sample identifiers. The sample identifiers must be identical to the data file, however, not required in the same order.
3. The first row must contain phenotypic information, or descriptions of each variable.



	A	B	C	D	E	F	G
1	#SampleID	person_id	time	study	probio	age	b_contra
2	2bcont	2	0	b_cont	NA	35	y
3	3bcont	3	0	b_cont	NA	44	n
4	4bcont	4	0	b_cont	NA	33	y
5	5bcont	5	0	b_cont	NA	33	y
6	6bcont	6	0	b_cont	NA	21	y
7	7bcont	7	0	b_cont	NA	36	y
8	8bcont	8	0	b_cont	NA	18	y
9	9bcont	9	0	b_cont	NA	29	y
10	10bcont	10	0	b_cont	NA	38	y

Figure 3: Example metadata file. Metadata maybe be numerical or categorical. Any blank spaces will be replaced as NA when importing the file. Any values of T or F will be read as TRUE or FALSE. Image taken from modified version of Vaginal dataset.

5.2 Example Data

The 'Example data' tab on the sidebar panel provides access to two example datasets. We will be using the provided 'Vaginal dataset', which contains both an OTU table and associated metadata. The 'Selex dataset' is from a selective growth experiment giving the differential abundance of 1600 enzyme variants[5]. After selecting the 'Vaginal dataset', return to the 'Data' tab to and click 'Show data' to view the data. You can view the metadata by clicking 'Show metadata' and switching the tab in the main panel to 'Metadata'. Click the 'PCA Biplots' main tab to proceed.

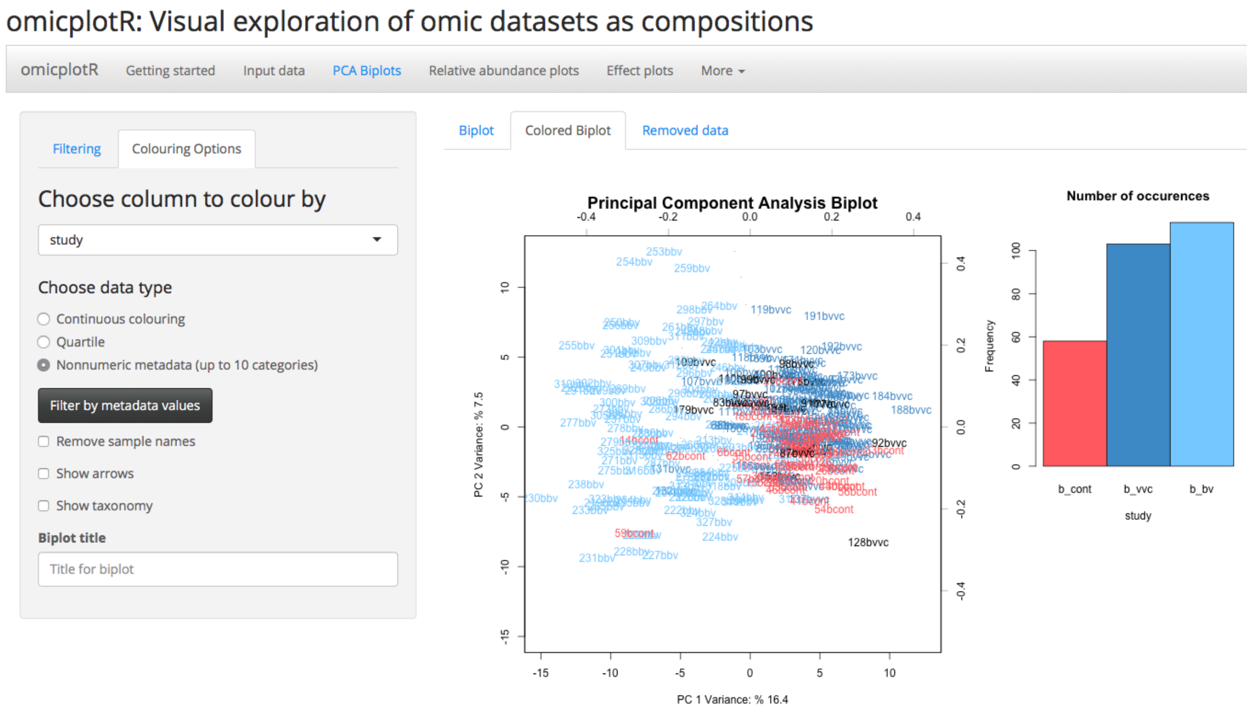
6 PCA Biplots

The 'Filtering' tab within the sidebar panel allows you to choose filtering options for your dataset. Colouring options for a coloured PCA biplot are available under the 'Colouring options' tab. The tabs within the main panel allow you to switch between displaying a biplot under 'Biplot', a biplot coloured by metadata under 'Coloured Biplot', and visualizations of the removed samples/features from filtering under the 'Removed data' tab.

6.1 Filtering

"Reasonable" filtering of sparse features (rows) from your dataset **should not** affect any conclusions made from the PCA biplots. Filtering is exploratory, and should be experimented with to see how removing sparse data affects the structure of your dataset within the PCA biplot. The default filtering removes any feature that has a maximum value less than 10. We will not change the default filtering for the purposes of the vignette.

[5] McMurrough et al (2014) PNAS doi:10.1073/pnas.1322352111



omicplotR: Visual exploration of omic datasets as compositions

Choose column to colour by: study

Choose data type

○ Continuous colouring
○ Quartile
● Nonnumeric metadata (up to 10 categories)

□ Remove sample names
□ Show arrows
□ Show taxonomy

Biplot title: Title for biplot

Figure 4: Screenshot of coloured PCA biplot.

Under the ‘Filtering’ tab within the sidebar panel, the following options are available to filter your data.

1. **Minimum count per OTU**
   This filter removes any rows with a maximum sample count less than the input. Default is 10 to remove sparse OTUs.

2. **Minimum count sum per sample**
   This filter removes any columns that have a lower sum of counts than the filter. You can visualize which samples are being removed when applying this filter using the ‘Filtering by counts per sample’ graph under the ‘Removed data’ tab.

3. **Minimum proportional abundance**
   This filter calculates frequency for each sample (by column) after filters 1. and 2. are applied, and removes any rows that have all frequencies below the threshold.

4. **Maximum proportional abundance**
   This filter calculates frequency for each sample, and removes any rows that have no frequencies below the threshold.

5. **Minimum count sum per OTU**
   This filter removes any rows that have a lower sum of counts than the filter. You can visualize which rows are removed with the ‘Filtering by counts per row’ graph under the ‘Removed data’ tab.

6. **Variance cutoff**
   This filter calculates the variance for each OTU (row), and removes any rows that have lower variance than the filter. Variance is calculated after all other filters, and after the reads have been transformed by the centre-log ratio.

7. **Adjust scale**
   This changes the scale of the biplot. When set to zero, it shows the relationship between samples (columns), while being set to 1 shows the relationship between OTUs (rows). Data is not filtered from

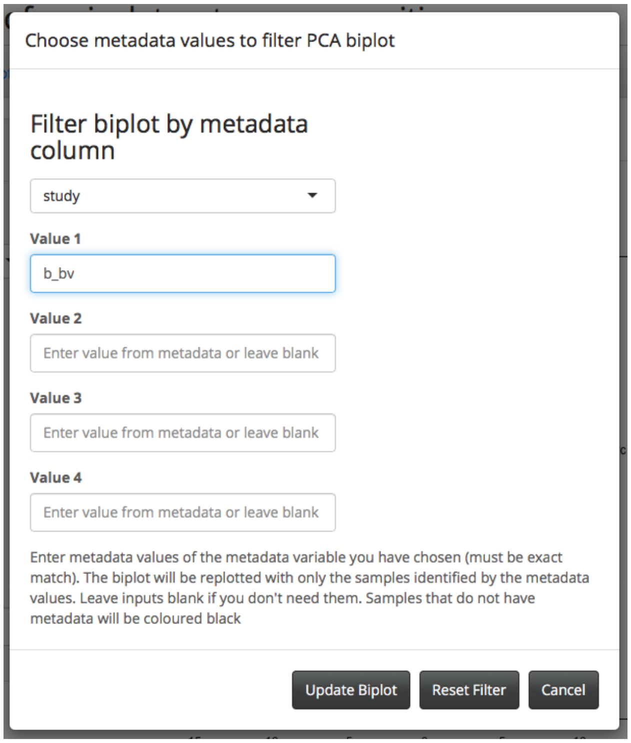
this operation.

6.2 Colouring options

To view the coloured PCA biplot, click the ‘Coloured Biplot’ tab within the main panel. Clicking the ‘Colouring options’ panel within the side bar panel allows you to choose which metadata to colour your plot by. Choose the column ‘study’ to view the data coloured by study. You will need to click the ‘nonnumeric metadata’ radio button since the values of the column are categorical. The sample names that are black (ie, 128bvvc) on the PCA biplot represent samples without metadata.

6.2.1 Filter by metadata

Under the ‘Colouring options’ tab in the sidebar panel, click on ‘Filter by metadata values’ to replot samples belonging to certain groups according to the metadata. This will generate a pop-up (Figure 5). Filter the dataset to replot only the samples from the ‘b_bv’ study by selecting the ‘study’ column, and inputting ‘b_bv’ into the text field for Value 1. This should update the coloured PCA biplot to show samples from only the ‘b_bv’ study. Click ‘Update Filter’ to update the filter, or ‘Reset Filter’ to reset it. This filter is applied to the data for all other plots (relative abundance, effect plots).



Choose metadata values to filter PCA biplot

Filter biplot by metadata column

study

Value 1

b_bv

Value 2

Enter value from metadata or leave blank

Value 3

Enter value from metadata or leave blank

Value 4

Enter value from metadata or leave blank

Enter metadata values of the metadata variable you have chosen (must be exact match). The biplot will be replotted with only the samples identified by the metadata values. Leave inputs blank if you don't need them. Samples that do not have metadata will be coloured black

Figure 5: Pop-up generated from clicking ’Filter by metadata values’. Select a column and enter values from the column to re-plot only those values. They must match exactly. The filter can be reset or updated.

7 Relative abundance plots

Click the ‘Relative abundance plots’ main tab to continue. The sidebar panel allows you to filter the data by choosing an abundance cutoff, select the clustering method, and select the distance matrix method. The main panel displays a dendrogram above a stacked barplot, and both can be zoomed in by dragging your mouse over a selected region and double clicking.

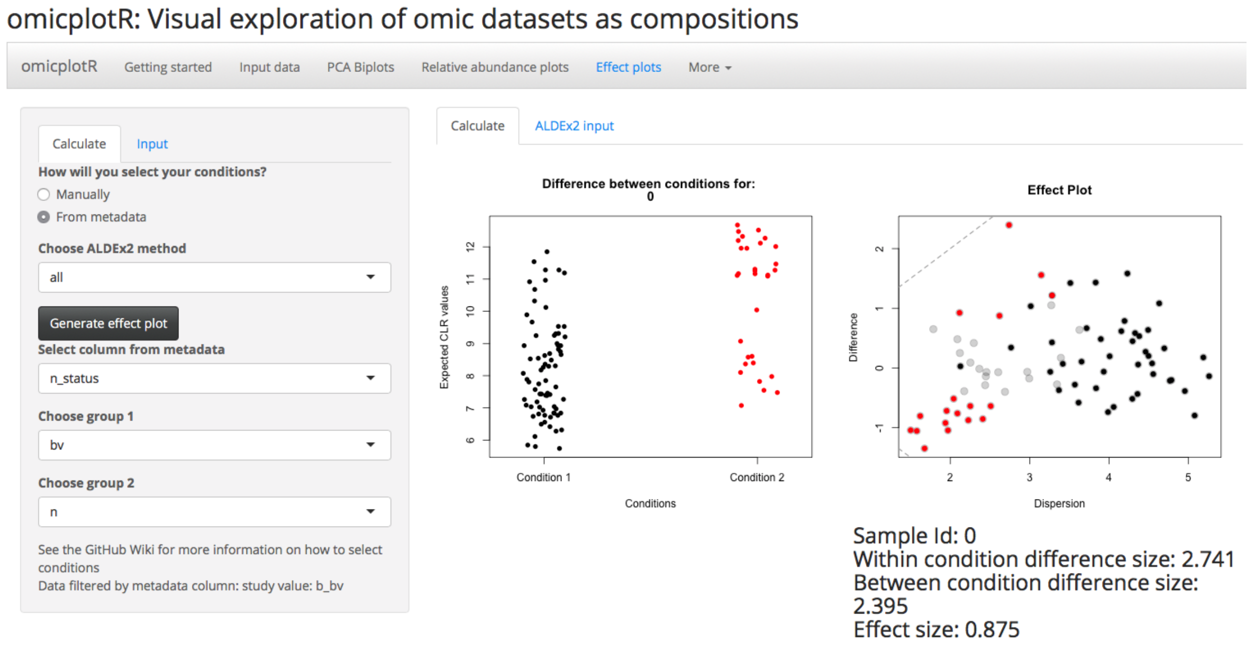
Any filtering by metadata will be reflected in the dendrogram and stacked barplot as well.

8 Effect plots

To generate effect plots, click on the ‘Effect plots’ main tab. Here, you can choose to calculate the `ALDEx2` table manually (ie, choosing which columns to compare) or by using metadata. For this example, we will use the included metadata to compare samples that were positive for bacterial vaginosis or not according to Nugent status. More information on choosing manually for your own data set is available on GitHub wiki.

Select the ‘n_status’ column from metadata. For Group 1 choose ‘bv’, and for Group 2 choose ‘n’. After these have been selected, click the ‘Generate effect plot’ button. An effect plot and Bland-Altman plot will be generated. For large datasets, this will take a long time. It takes about 10 seconds for the example data, which has 77 features for 297 samples.

By hovering over a point on the effect plot, a stripchart of the expected CLR abundance (see `ALDEx2`) for each sample will be shown, allowing you to compare the differences between your samples for a given feature (Figure 6). It also displays information from the effect table generated by `ALDEx2`, such as the effect size, median difference between groups, and median difference within groups.



omicplotR: Visual exploration of omic datasets as compositions

How will you select your conditions?

○ Manually

● From metadata

Choose ALDEx2 method: all

Select column from metadata: n_status

Choose group 1: bv

Choose group 2: n

See the GitHub Wiki for more information on how to select conditions

Data filtered by metadata column: study value: b_bv

Sample Id: 0

Within condition difference size: 2.741

Between condition difference size: 2.395

Effect size: 0.875

Figure 6: Screenshot of effect plots. Hovering over a feature’s point in the effect plot generates a stripchart to compare the relative abundances calculated by ALDEx2 for each sample.

If you are using your own data and want to select groups by columns, you will need to reorder your file to have the first n columns as group 1, and the last n columns as group 2.

9 Contributors

Daniel J Giguere wrote the original omicplotR code and designed the `Shiny` app, with help from Brandon Lieng, Jean Macklaim, and Greg Gloor. Brandon Lieng wrote the function `clr.strip()` needed for the strip charts. Jean Macklaim conceptualized this project, contributed numerous ideas for the design, and wrote the original code for the taxonomic distribution and dendrograms. Both Greg and Jean played roles in designing and implementing omicplotR.

10 Version information

Currently version 0.99.0.

For more information about how to use `omicplotR`, please visit the wiki on my Github page:

https://github.com/dgiguer/omicplotR/wiki