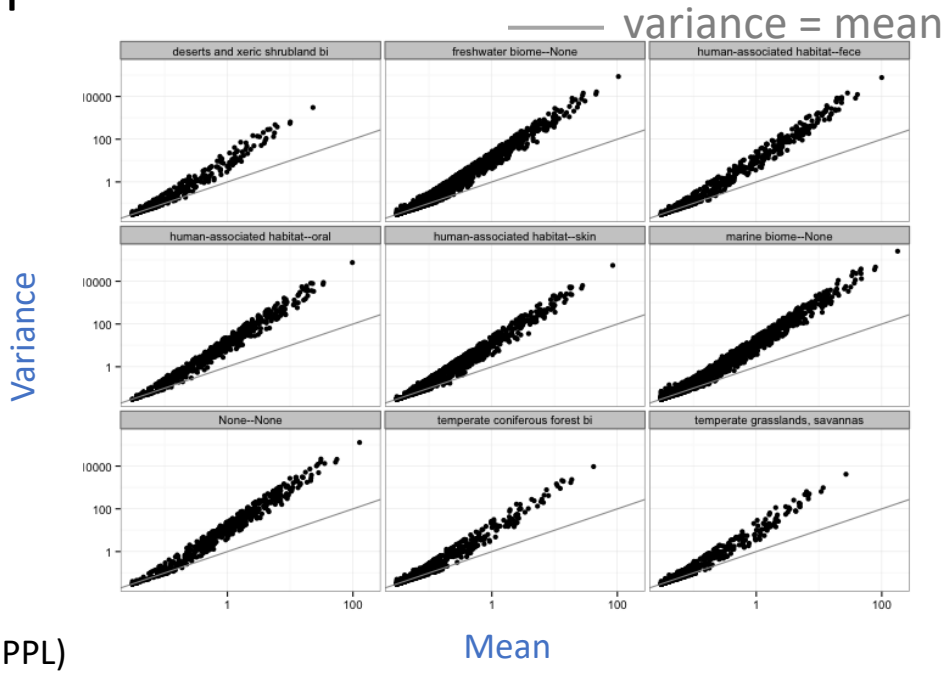
MB590-012
Microbiome Analysis
Data considerations and transformations

Dr. Christine Hawkes

NC STATE UNIVERSITY

ASV matrix data

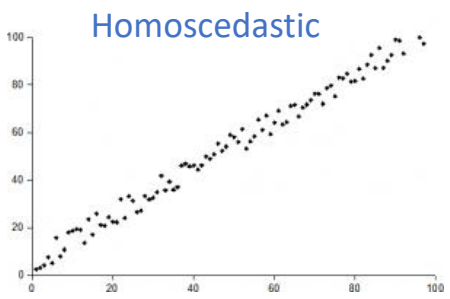
- Large number of ASVs but typically a much smaller number of samples with few replicates
- Sparse with many zeros (can be >80% zero counts!)
- Overdispersed
  - More variability than expected among biological replicates
  - Variance increases with mean
    - Example - global patterns data

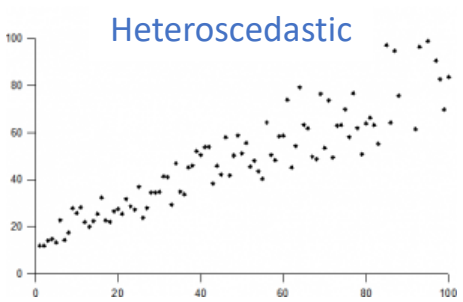


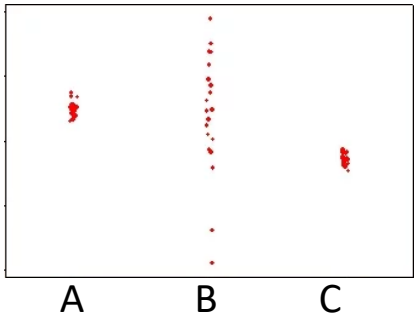
McMurdie & Holmes 2014 PLoS Comput Biol 10: e1003531 (SUPPL)

ASV matrix data

- Violate assumptions of most parametric statistics (affects parameter estimates & significance tests)
  - Linearity
  - Normality
  - Homogeneity of variance (homoscedasticity)

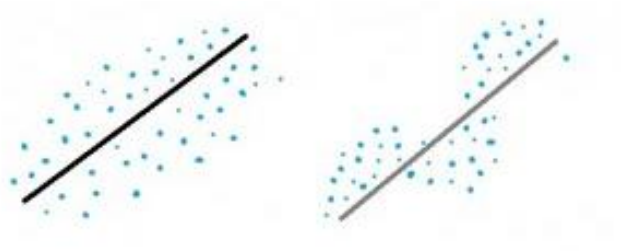






A vs C same var
A&C vs B diff var

- Error independence (errors are randomly distributed)



Common transformations

- Rarefaction – adjust samples to a specific count
  - *(note: I discourage you from using this approach)*

- Variance stabilizing (vst) – McMurdie & Holmes 2014
- Regularized log (rlog) – Love et al. 2014

- Centered log ratio (clr) – Gloor et al. 2017, Aitchison 1986

Rarefaction

- Removes different library sizes that occur during amplification/sequencing

- Typical steps:
  - Select library size, usually the smallest in your data set
  - (Discard samples smaller than that size)
  - Randomly subsample remaining libraries without replacement to obtain the libraries of the same size in all samples

Problems with rarefaction

- Lack of reproducibility in subsampling
- Discards original estimates of uncertainty in abundance of each OTU by changing denominator in proportion calculations
- Inflates variance of samples to the worst value among them (smaller N in denominator) $\sigma^2 = 1/N^2 \times \sum(Y_i - \bar{Y})^2$
- Loss of information and power

Original Abundance

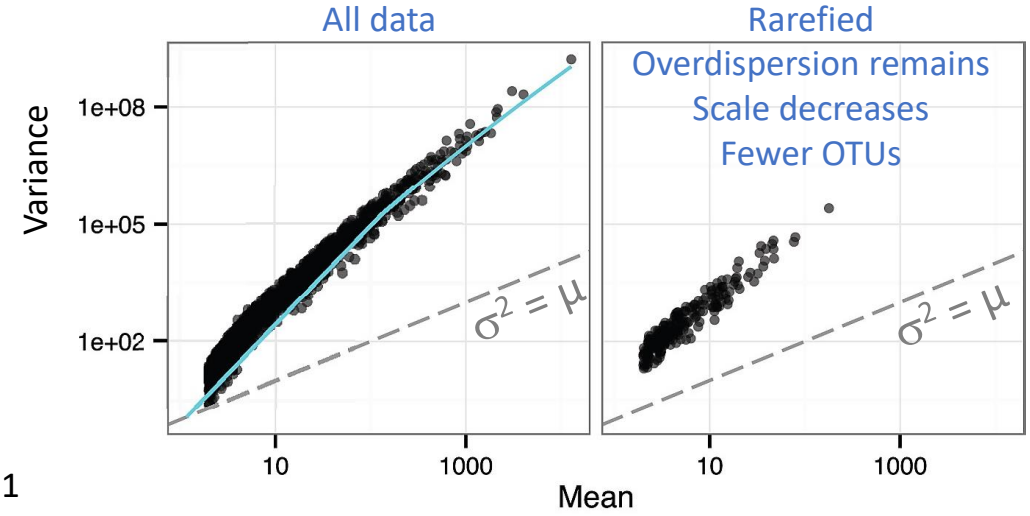
	A	B
OTU1	62	500
OTU2	38	500
Total	100	1000

Rarefied Abundance

	A	B
OTU1	62	50
OTU2	38	50
	100	100

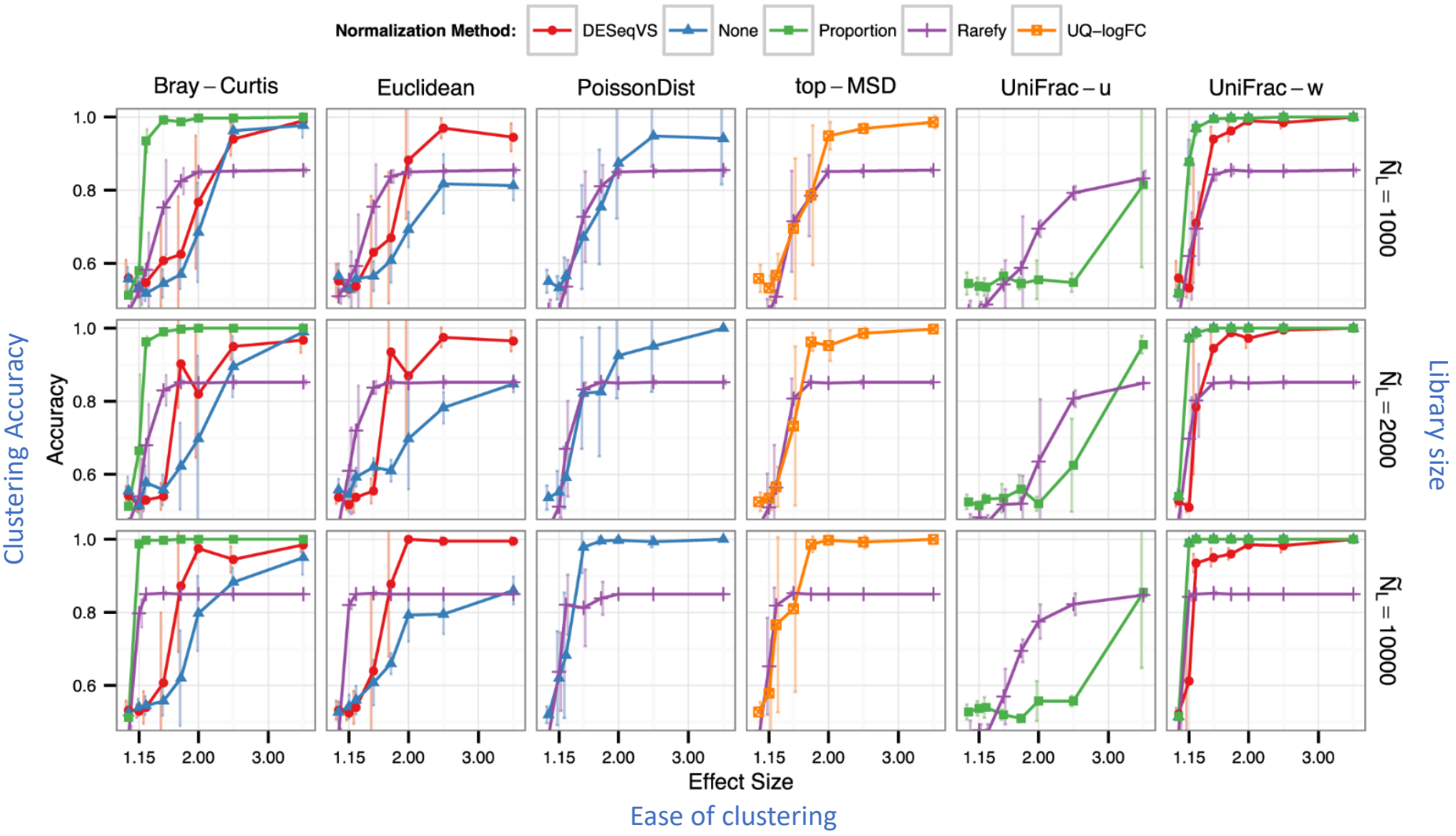
Standard Tests for Difference

P-value	chi-2	Prop	Fisher
Original	0.0290	0.0290	0.0272
Rarefied	0.1171	0.1171	0.1169



McMurdie & Holmes 2014 PLoS Comput Biol 10: e1003531

Rarefaction also undermines downstream clustering (best is ~80% accuracy)



McMurdie & Holmes 2014 PLoS Comput Biol 10: e1003531

What does compositional mean?

- Data provide only relative abundances
  - Proportions sum to 1
  - Not absolute abundances
- Total read number is arbitrary and constant
  - Artifact of Illumina total reads being limited by the machine
  - Can result in spurious correlations
- Reads are random subsamples of true molecules that do not necessarily reflect original sample
- Analysis of data subset is not consistent with analysis of total data (membership-dependent)
- Random sampling of variables can lead to change of conclusions (as seen with rarefaction)

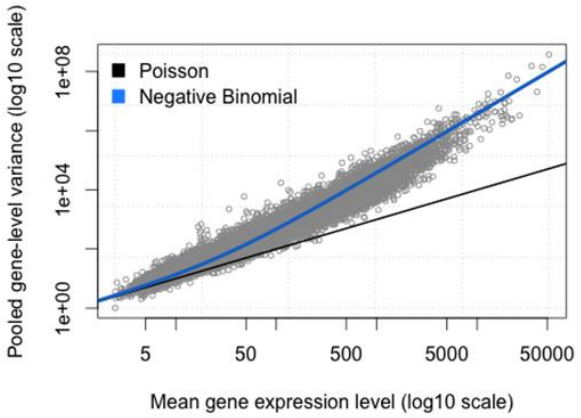
How to deal with compositional data

- Initially, data were scaled to total read abundance to generate relative abundances (proportions)
  - Can skew true relationships in the data
  - Does not address problems with overdispersion

- Recent alternatives to address problems with proportions include:
  - Use of transformation methods such as vst and rlog that adjust by library size and address overdispersion
  - Log ratios transformations to preserve underlying relationships in the data

vst and rlog transformations

- Variance stabilizing (vst) – McMurdie & Holmes 2014
  - Factors out dependence of var on the mean and normalizes by library size
- Regularized log (rlog) – Love et al. 2014
  - Minimizes differences between samples for rows with small counts and normalizes by library size
- Both are run in DESeq2
  - Originally developed for gene expression data
  - Assume a negative binomial distribution (overdispersion)
  - Better fit than traditional Poisson



Variance stabilizing transform (vst)

- Addresses different library sizes, sparse data, heteroscedasticity, overdispersion
- To do this, DeSeq2 identifies the mean-variance relationship in the data, and then derives a transformation that will remove the dependence so that the variance is the same across samples (homoscedasticity)
- Approaches the $\log_2$ transformation for large values and square root function for small values

Regularized log transform (rlog)

- Addresses same same issues as the vst but:

- Transforms count data to log2 scale by fitting a model that includes a term for each sample plus a Bayesian prior distribution on the model coefficients (estimated from the data)

$$\log_2(q_{ij}) = \beta_{i0} + \beta_{ij}$$

Parameter proportional to true count for ASV i in sample j

intercept

sample-specific effect, shrunk towards zero based on dispersion-mean trend in entire dataset

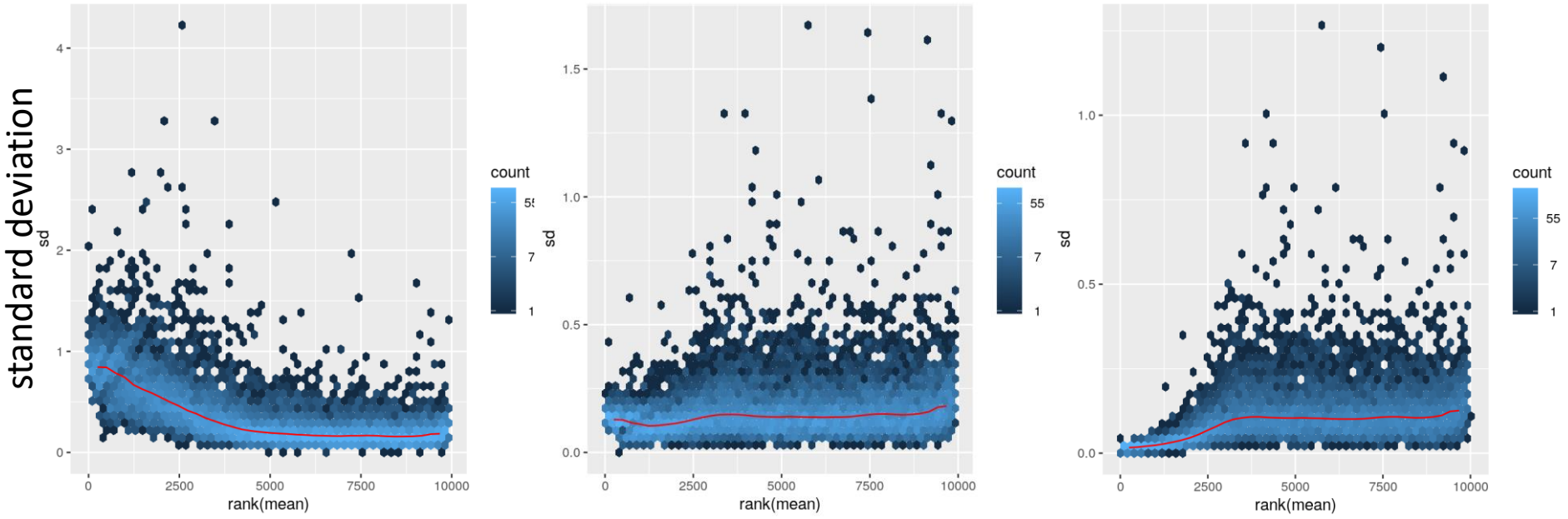
- Much slower than vst for large data sets (100s of samples)

vst and rlog transformation effects on variance

log2(n+1)
high var at low mean

vst
sd ~constant
through range

rlog
shifted to right

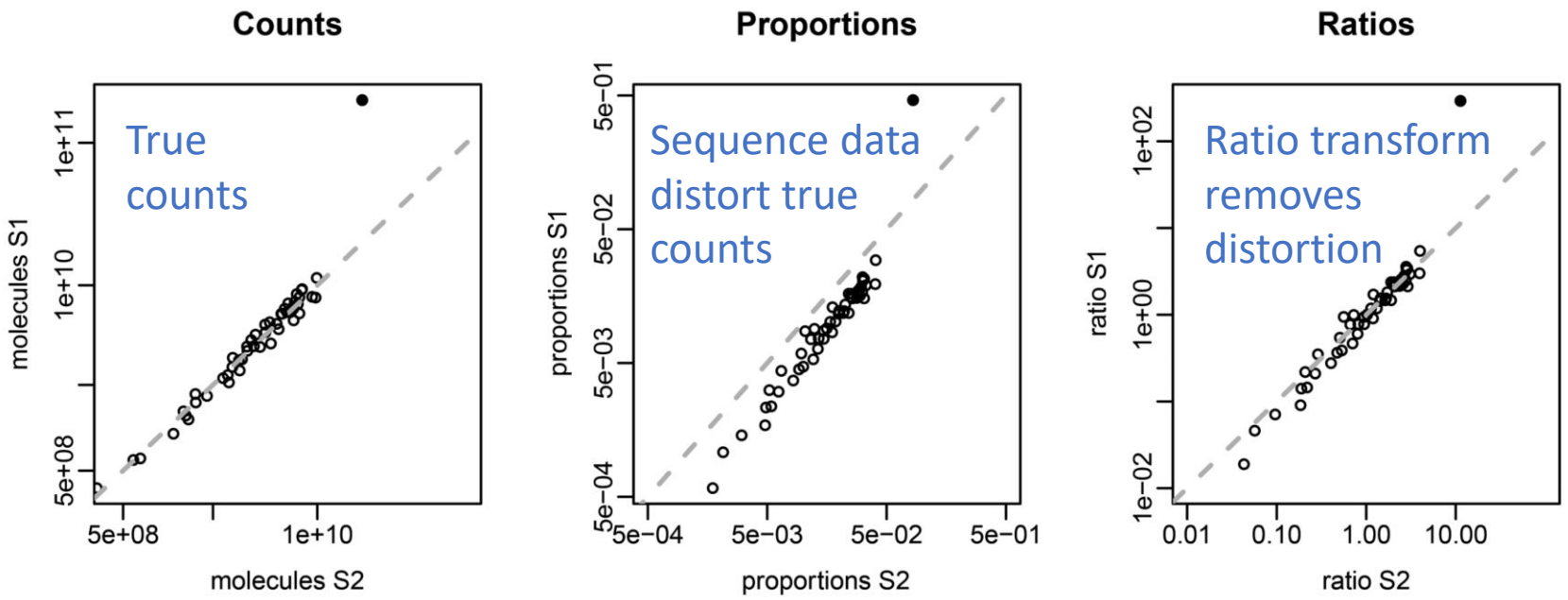


http://bioconductor.org/packages/devel/bioc/vignettes/DESeq2/inst/doc/DESeq2.html#data-transformations-and-visualization

vst and rlog application in DESeq

- Can transform data in context of experimental design or "blind" (intercept only)
  - Design: ~ Trt1 + Trt2 vs. Blind: ~1
- Blind is inappropriate if we expect ASV counts to be explained by experimental design
  - can lead to large dispersion estimates by assuming differences caused by treatments are noise
- Dispersion estimates have 3 fit types
  - Parametric – vst on normalized counts
  - Mean – vst on negative binomial counts with fixed dispersion
  - Local – reciprocal sq root of variance of normalized counts (derived from dispersion fit), then numerically integrated, and the integral is evaluated for each count, yielding transformed values

Ratio-based transformations



Gloor & Reid 2016 Can J Microbiol 62: 692-703 doi:dx.doi.org/10.1139/cjm-2015-0821

Centered log ratio transform (clr)

- For *D* ASVs in each sample, the clr transformation is the ln of the proportion of counts, where the denominator is the geometric mean of all counts for that sample

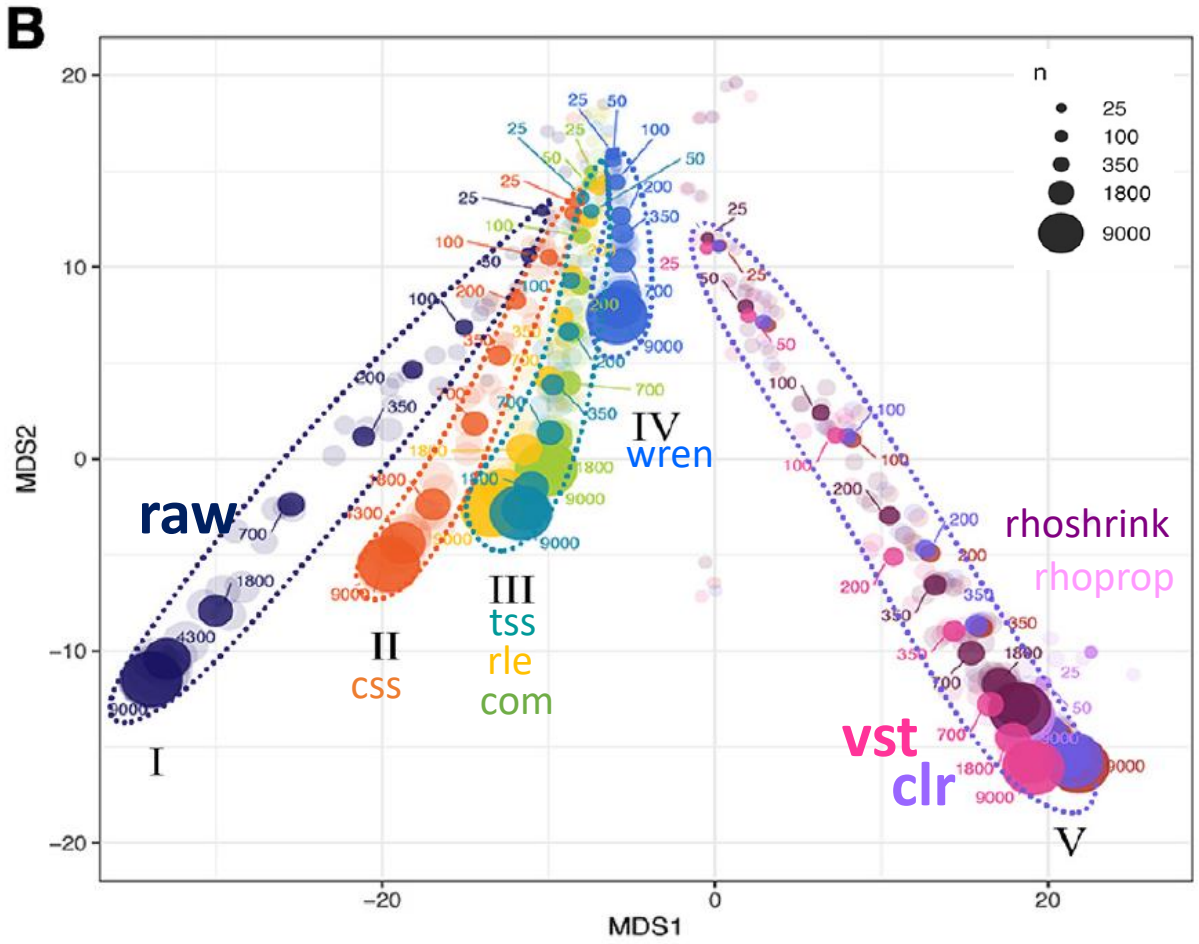
$$\boldsymbol{x}_{clr} = [log(x_1/G(\boldsymbol{x})), log(x_2/G(\boldsymbol{x})) \ldots log(x_D/G(\boldsymbol{x}))],$$
$$G(\boldsymbol{x}) = \sqrt[D]{x_1 \cdot x_2 \cdot \ldots \cdot x_D}$$

- Data remain relative (ratio-based), but are also
  - Scale invariant (unit free)
  - Sub-compositionally coherent (consistent across subsets)
- Other log ratio transforms include additive (alr), isometric (ilr), and phylogenetic isometric (philr)

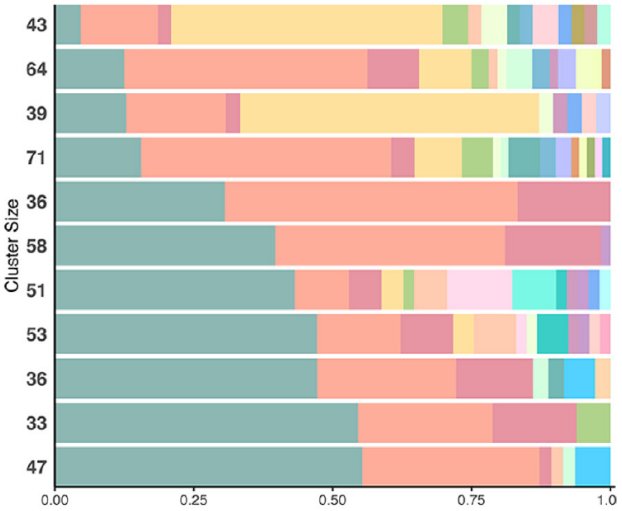
What to do about zeros or negative numbers?

- The vst, rlog, and clr transformations may produce negative values for zeros
- This can be a problem in downstream analyses (e.g., distance matrices)
- Options
  - Set all negative values to zero
  - Add a pseudocount to all cells
  - Analyze the transformed data without further manipulation (don't calculate distances)
  - For dissimilarity analysis, use Euclidean distance on transformed data (= Aitchison distance between clr-transformed compositions)

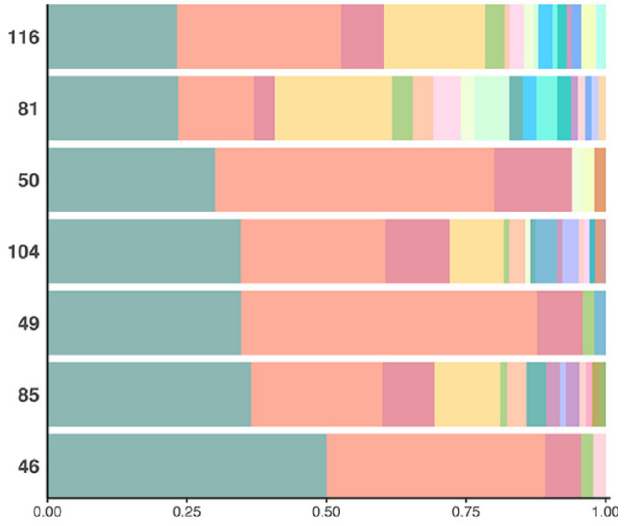
Transformation matters to downstream analysis!



clr



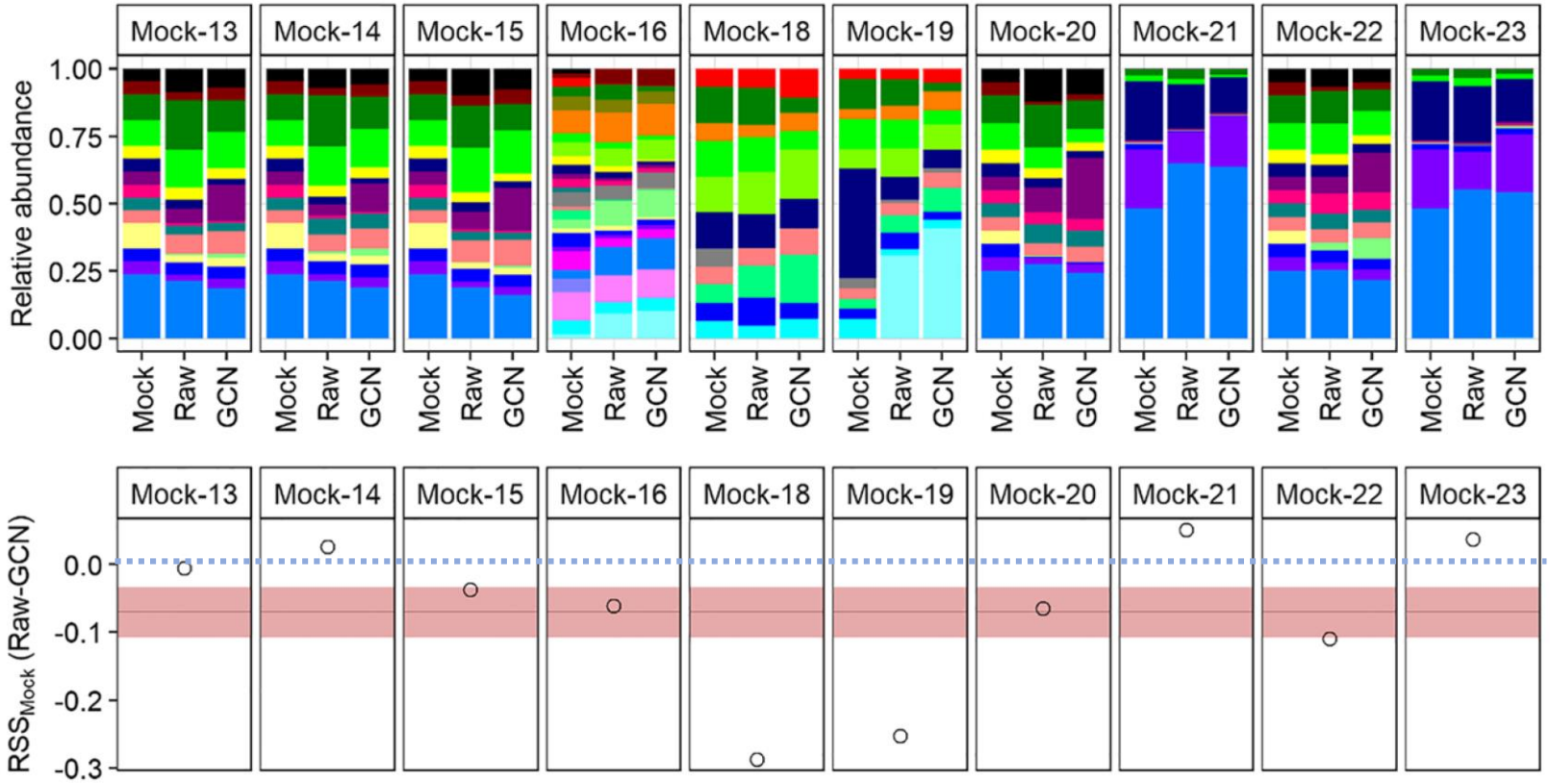
tss



Badri et al. 2020 NARG&B doi: 10.1093/nargab/lqaa100

Other transforms: rRNA gene copy number normalization (GCN)

Raw seq data fit actual Mock 7.1% better than GCN on avg



Bottom panel is sum of deviations between actual Mock community abundances and GCN-adjusted.

Starke et al. 2021 Microb Ecol https://doi.org/10.1007/s00248-020-01586-7

Can you get absolute abundance from microbiome data?

- Spike-in controls
- Cell flow cytometry
- qPCR

- Examples:
  - Deng, L. et al. 2019. Front. Microbiology 10: 720, DOI: 10.3389/fmicb.2019.00720
  - Hardwick et al. 2018 Nature Communications 9:3096; DOI: 10.1038/s41467-018-05555-0
  - Stammler et al. 2016 Microbiome 4:28; DOI: 10.1186/s40168-016-0175-0
  - Tourlouse et al. 2016 Nucleic Acids Research 45:e23; DOI: 10.1093/nar/gkw984

Comparing transformations to quantitative profiling

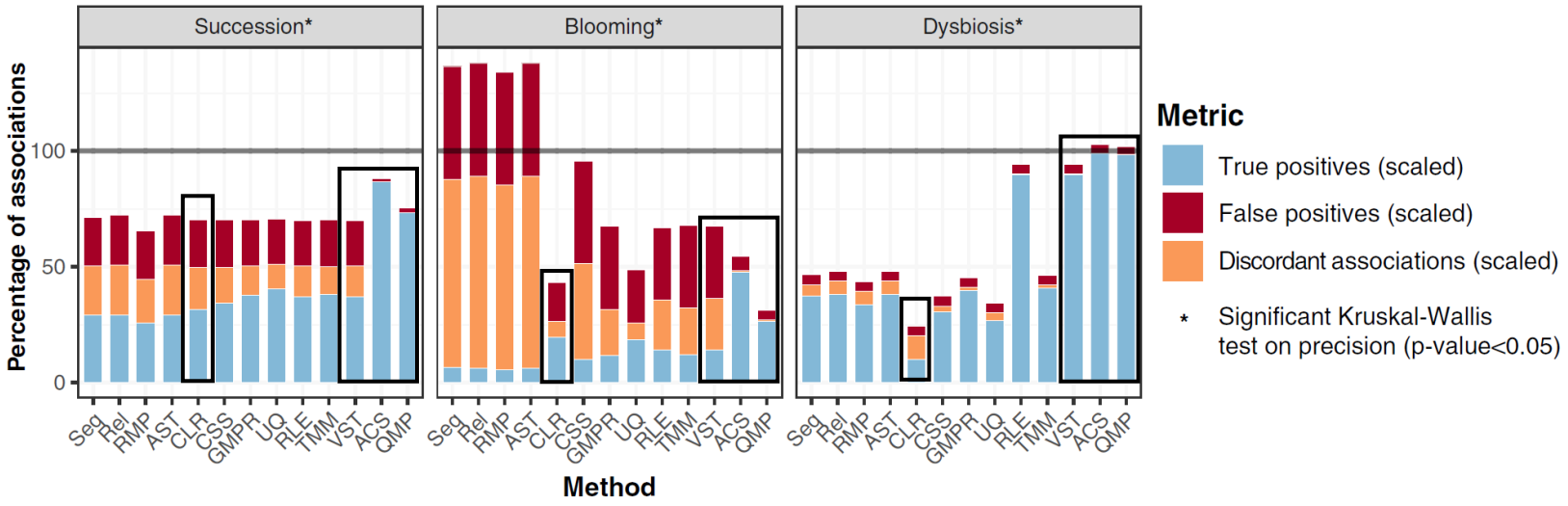
Table 2 Metagenomic data transformations benchmarked.

Method	Abbreviation	Technique	Transformation	Correction	Rarefaction	Suited for richness calculations
Raw sequencing data	Seq	-	None	-	No	Yes
Relative abundance	Rel	Computational	Relative	Sequencing depth	No	No (0:1 range)
Relative microbiome profiling	RMP	Computational	Relative	Sequencing depth	Yes	Yes
Arcsine square root	AST	Computational	Relative	Sequencing depth	No	No (0:1 range)
Centered log ratio	CLR	Computational	Compositional	Sequencing depth and compositionality	No	No (negative values)
Cumulative sum scaling	CSS	Computational	Compositional	Sequencing depth and compositionality	No	Yes (rounding data)
Relative log expression	RLE	Computational	Compositional	Sequencing depth and compositionality	No	Yes (rounding data)
Upper quantile	UQ	Computational	Compositional	Sequencing depth and compositionality	No	Yes (rounding data)
Trimmed mean of m-values	TMM	Computational	Compositional	Sequencing depth and compositionality	No	Yes (rounding data)
Geometric mean of pairwise ratios	GMPR	Computational	Compositional	Sequencing depth and compositionality	No	Yes (rounding data)
Variance-stabilizing transformation	VST	Computational	Compositional	Sequencing depth and compositionality	No	No (negative values)
Quantitative microbiome profiling	QMP	Experimental	Quantitative	Sampling depth and microbial load	Yes	Yes
Absolute count scaling	ACS	Experimental	Quantitative	Microbial load	No	Yes

Methods are categorized based on the technique applied (computational or experimental), the biases targeted (sequencing depth, sampling depth, compositionality, and/or microbial load), the inclusion of a downsizing step, and their projected suitability for richness estimations. Additionally, for study purposes, methods are broadly labeled as relative, compositional, or quantitative methods.

Llorens-Rico et al. 2021 Nature Comms https://doi.org/10.1038/s41467-021-23821-6 |

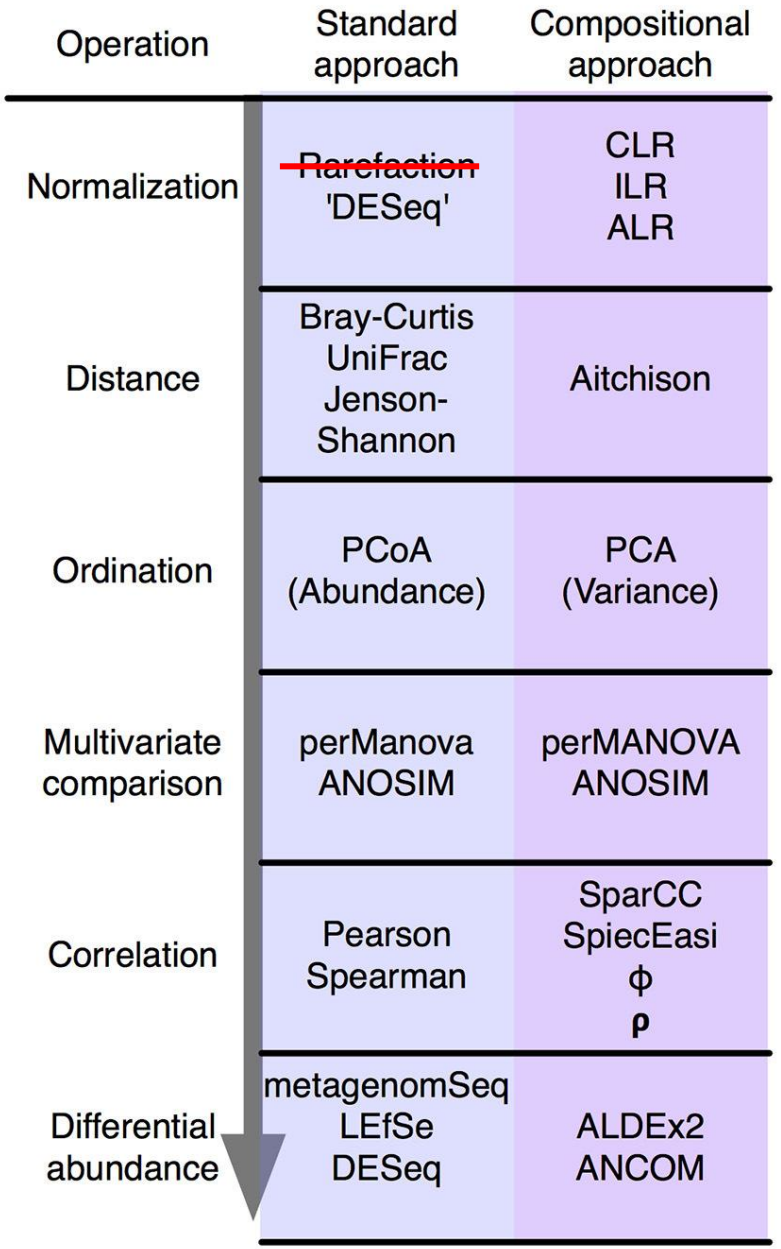
vst transformation was most comparable to quantitative methods



- **Quantitative methods** have the most true and fewest false positives (and no discordance)
- **vst** is most like the quantitative methods, but performance depends on dataset
- **clr** and other transforms generally perform more poorly

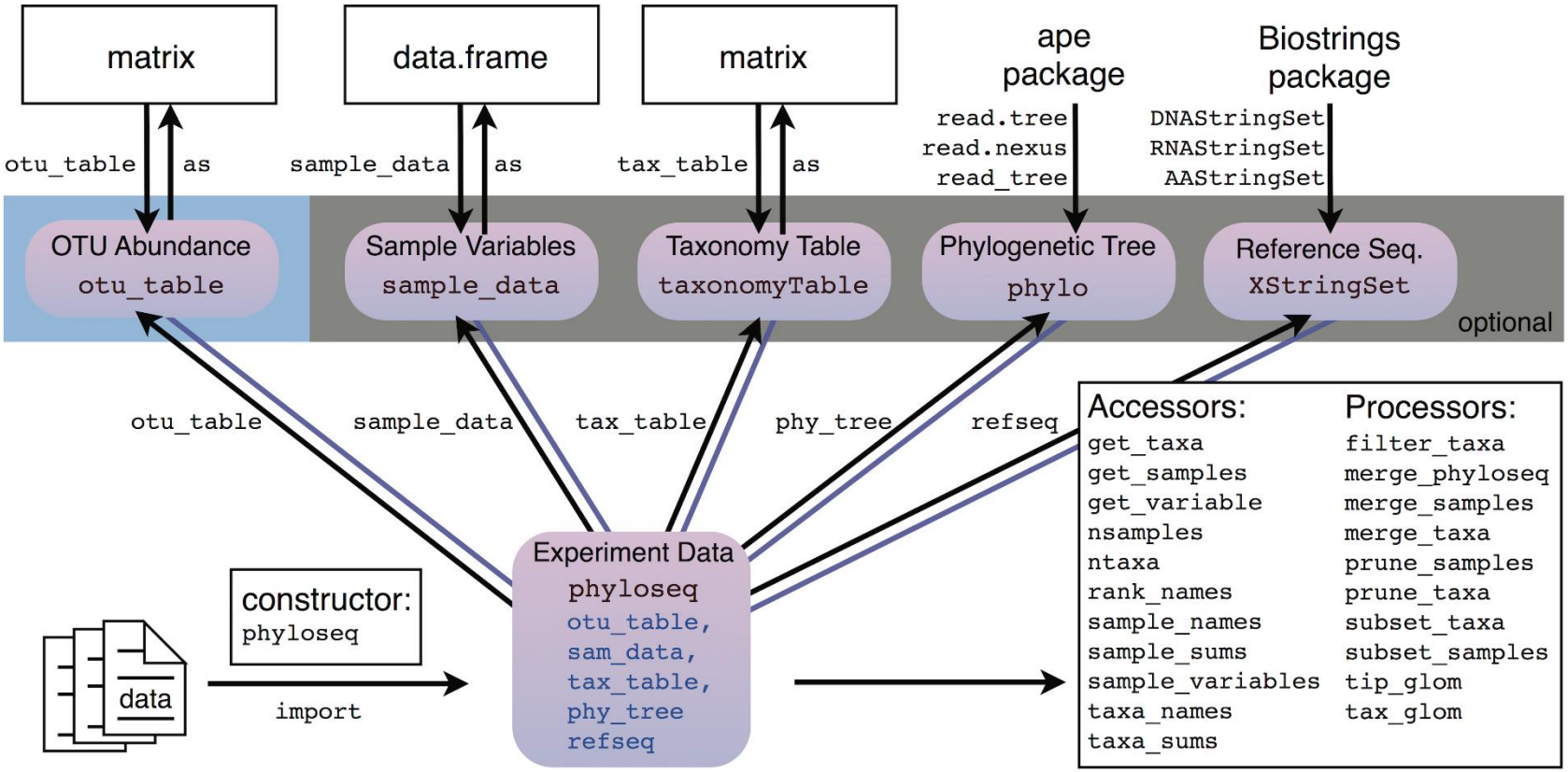
Llorens-Rico et al. 2021 Nature Comms https://doi.org/10.1038/s41467-021-23821-6 |

Different pipelines for downstream analysis for compositional data



Gloor et al. 2017 Frontiers Micro 8:2224 doi: 10.3389/fmicb.2017.02224

Use of phyloseq for data management



McMurdie&Holmes 2013 PLoS1 https://doi.org/10.1371/journal.pone.0061217

Functions for building phyloseq objects

Functions for building component data objects

Function	Input Class	Output Description
`otu_table`	numeric matrix	`otu_table` object storing OTU abundance
`otu_table`	data.frame	`otu_table` object storing OTU abundance
`sample_data`	data.frame	`sample_data` object storing sample variables
`tax_table`	character matrix	`taxonomyTable` object storing taxonomic identities
`tax_table`	data.frame	`taxonomyTable` object storing taxonomic identities
`read_tree`	file path char	phylo-class tree, read from file
`read.table`	table file path	A matrix or data.frame (Std `R` core function)

Functions for building complex data objects

Function	Input Class	Output Description
`phyloseq`	2 or more component objects	phyloseq-class, "experiment-level" object
`merge_phyloseq`	2 or more component or phyloseq-class objects	Combined instance of phyloseq-class

Accessor functions for phyloseq objects

Function	Returns
`[`	Standard extraction operator. works on `otu_table`, `sample_data`, and `taxonomyTable`
`access`	General slot accessor function for phyloseq-package
`get_taxa`	Abundance values of all taxa in sample 'i'
`get_sample`	Abundance values of taxa 'i' for all samples
`get_taxa_unique`	A unique vector of the observed taxa at a particular taxonomic rank
`get_variable`	An individual sample variable vector/factor
`nsamples`	Get the number of samples described by an object
`ntaxa`	Get the number of OTUs (taxa) described by an object
`otu_table`	Build or access otu_table objects
`rank_names`	Get the names of the available taxonomic ranks
`sample_data`	Build or access sample_data objects
`sample_names`	The names of all samples
`taxa_names`	The names of all taxa
`sample_sums`	The sum of the abundance values of each sample
`sample_variables`	The names of sample variables
`taxa_sums`	The sum of the abundance values of each taxa
`taxa_are_rows`	`TRUE` if taxa are row indices in `otu_table`
`tax_table`	A taxonomy table
`tre`	Access the tree contained in a phyloseq object