Drained-Flooded

Anderson Freitas

13/04/2022

#Step zero: cosmetics.

```
library(ggplot2)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(hrbrthemes)
```

```
## NOTE: Either Arial Narrow or Roboto Condensed fonts are required to use these themes.

##       Please use hrbrthemes::import_roboto_condensed() to install Roboto Condensed and

##       if Arial Narrow is not on your system, please see https://bit.ly/arialnarrow
```

```
theme_set(theme_bw())
pal = "Set3"
scale_colour_discrete <-  function(palname=pal, ...){
  scale_colour_brewer(palette=palname, ...)
}
scale_fill_discrete <-  function(palname=pal, ...){
  scale_fill_brewer(palette=palname, ...)
}
```

#First step: descriptives.

```
library(readr)
```

```
## Warning: package 'readr' was built under R version 4.1.2
```

```
data_df <- read_delim("E:/Anderson-BackUp/Rice/descriptive.txt", delim = "\t", escape_double = FALSE,
    trim_ws = TRUE)
```

```
## Rows: 8 Columns: 11

## -- Column specification --------------------------------------------------------
## Delimiter: "\t"
## chr  (1): Soil
## dbl (10): OM, pH, EC, Al3+, Amonium, Nitrates, P, K, Ca, Mg

##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
library(tidyverse)
```

```
## Warning: package 'tidyverse' was built under R version 4.1.2

## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v tibble  3.1.3     v stringr 1.4.0
## v tidyr   1.1.3     v forcats 0.5.1
## v purrr   0.3.4

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(dplyr)

data.df       = data_df %>% group_by(Soil)

Means    = data.df %>% summarise_if(is.numeric, mean)
SD       = data.df %>% summarise_if(is.numeric, sd)

mpsd     = rbind(Means, SD)

#mpsd$Soil

mpsd$Data = c("Mean", "Mean", "SD", "SD")

mpsd
```

```
## # A tibble: 4 x 12
##   Soil      OM     pH     EC `Al3+` Amonium Nitrates     P      K    Ca    Mg
##   <chr>  <dbl>  <dbl>  <dbl>  <dbl>   <dbl>    <dbl> <dbl>  <dbl> <dbl> <dbl>
## 1 GHC   6.82   4.42   346.   3.87     55.7     28.8  40.7  145.   8.69  3.55
## 2 GLC   1.21   5.49   120.   0.128    31.4     19.1  21.0   48.8  3.80  1.74
## 3 GHC   0.281  0.0250  14.7  0.0762    1.37     1.46  1.34   5.07 0.147 0.26
## 4 GLC   0.0529 0.0479   8.18 0.045     1.79     2.24  1.31   2.11 0.121 0.159
## # ... with 1 more variable: Data <chr>
```

#Second step: test differences between descriptives.

```
colnames(data.df)
```

```
##  [1] "Soil"     "OM"       "pH"       "EC"       "Al3+"     "Amonium"
##  [7] "Nitrates" "P"        "K"        "Ca"       "Mg"
```

```
dim(data.df)
```

```
## [1]  8 11
```

```
norm.df = lapply(data.df[2:10], shapiro.test)

norm.df
```

```
## $OM
##
##  Shapiro-Wilk normality test
##
## data:  X[[i]]
## W = 0.71065, p-value = 0.002971
##
##
## $pH
##
##  Shapiro-Wilk normality test
##
## data:  X[[i]]
## W = 0.71264, p-value = 0.003128
##
##
## $EC
##
##  Shapiro-Wilk normality test
##
## data:  X[[i]]
## W = 0.72701, p-value = 0.004533
##
##
## $`Al3+`
##
##  Shapiro-Wilk normality test
##
## data:  X[[i]]
## W = 0.68944, p-value = 0.00171
##
##
## $Amonium
##
##  Shapiro-Wilk normality test
##
## data:  X[[i]]
```

```
## W = 0.76216, p-value = 0.01112
##
##
## $Nitrates
##
##  Shapiro-Wilk normality test
##
## data:  X[[i]]
## W = 0.87412, p-value = 0.1653
##
##
## $P
##
##  Shapiro-Wilk normality test
##
## data:  X[[i]]
## W = 0.76047, p-value = 0.01066
##
##
## $K
##
##  Shapiro-Wilk normality test
##
## data:  X[[i]]
## W = 0.72144, p-value = 0.003928
##
##
## $Ca
##
##  Shapiro-Wilk normality test
##
## data:  X[[i]]
## W = 0.70749, p-value = 0.002737
```

```
#Not normal-distributted. Kruskal-Walis will be the choice

library(broom)

KW.test <- data.df %>% gather(key, value, -Soil) %>%
  group_by(key) %>%
  do(tidy(kruskal.test(x= .$value, g = .$Soil)))

KW.test
```

```
## # A tibble: 10 x 5
## # Groups:   key [10]
##    key       statistic p.value parameter method
##    <chr>         <dbl>   <dbl>     <int> <chr>
##  1 Al3+           5.40  0.0202         1 Kruskal-Wallis rank sum test
##  2 Amonium        5.33  0.0209         1 Kruskal-Wallis rank sum test
##  3 Ca             5.33  0.0209         1 Kruskal-Wallis rank sum test
##  4 EC             5.40  0.0202         1 Kruskal-Wallis rank sum test
##  5 K              5.33  0.0209         1 Kruskal-Wallis rank sum test
##  6 Mg             5.33  0.0209         1 Kruskal-Wallis rank sum test
```

```
##  7 Nitrates      5.33  0.0209         1 Kruskal-Wallis rank sum test
##  8 OM            5.33  0.0209         1 Kruskal-Wallis rank sum test
##  9 P             5.33  0.0209         1 Kruskal-Wallis rank sum test
## 10 pH            5.33  0.0209         1 Kruskal-Wallis rank sum test
```

```
#Data from "mpsd" and "KW.test" will be used for the construction of a table.
```

#Third step: the evolution of pH and EC during the experiment.

```
#I used a function to extract mean and SD and plotted line graphs to show the evolution of pH and EC du

#The function "summarySE"
###############################################

summarySE <- function(data=NULL, measurevar, groupvars=NULL, na.rm=FALSE,
                      conf.interval=.95, .drop=TRUE) {
  library(plyr)

  # New version of length which can handle NA's: if na.rm==T, don't count them
  length2 <- function (x, na.rm=FALSE) {
    if (na.rm) sum(!is.na(x))
    else       length(x)
  }

  # This does the summary. For each group's data frame, return a vector with
  # N, mean, and sd
  datac <- ddply(data, groupvars, .drop=.drop,
                 .fun = function(xx, col) {
                   c(N    = length2(xx[[col]], na.rm=na.rm),
                     mean = mean   (xx[[col]], na.rm=na.rm),
                     sd   = sd     (xx[[col]], na.rm=na.rm)
                   )
                 },
                 measurevar
  )

  # Rename the "mean" column
  datac <- plyr::rename(datac, c("mean" = measurevar))

  datac$se <- datac$sd / sqrt(datac$N)  # Calculate standard error of the mean

  # Confidence interval multiplier for standard error
  # Calculate t-statistic for confidence interval:
  # e.g., if conf.interval is .95, use .975 (above/below), and use df=N-1
  ciMult <- qt(conf.interval/2 + .5, datac$N-1)
  datac$ci <- datac$se * ciMult

  return(datac)
}

###############################################

library(readxl)
Map_T <- read_excel("E:/Anderson-BackUp/Rice/Map_T.xlsx")
```

```
#View(Map_T)

#library(readxl)
rects <- read_excel("E:/Anderson-BackUp/Rice/rects.xlsx")
#View(rects)

rects$Colors = c("grey","white", "grey","white", "grey","white","grey","white", "grey","white", "grey")



tgc <- summarySE(Map_T, measurevar="pH", groupvars=c("Soil","Condition","Cycle"))
```

```
## ------------------------------------------------------------------------------
```

```
## You have loaded plyr after dplyr - this is likely to cause problems.
## If you need functions from both plyr and dplyr, please load plyr first, then dplyr:
## library(plyr); library(dplyr)
```

```
## ------------------------------------------------------------------------------
```

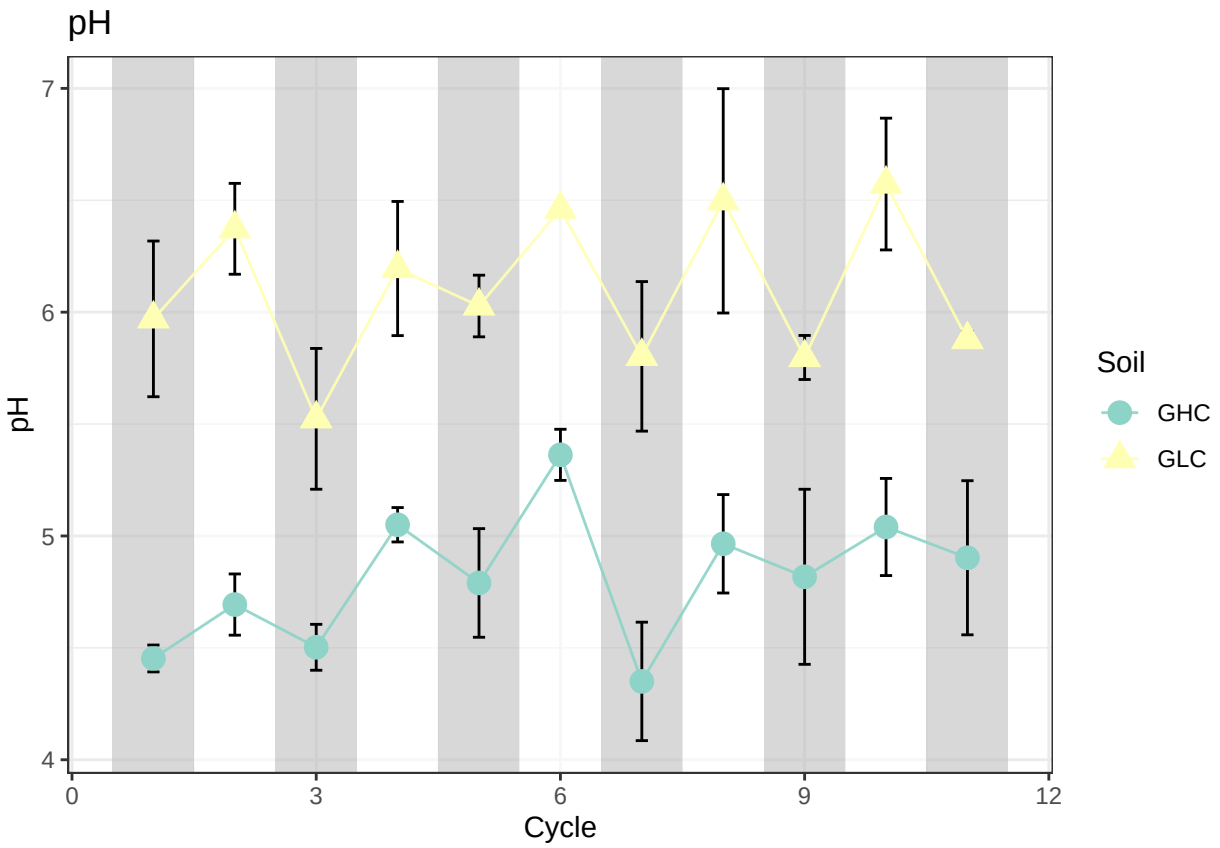
```
##
## Attaching package: 'plyr'
```

```
## The following object is masked from 'package:purrr':
##
##     compact
```

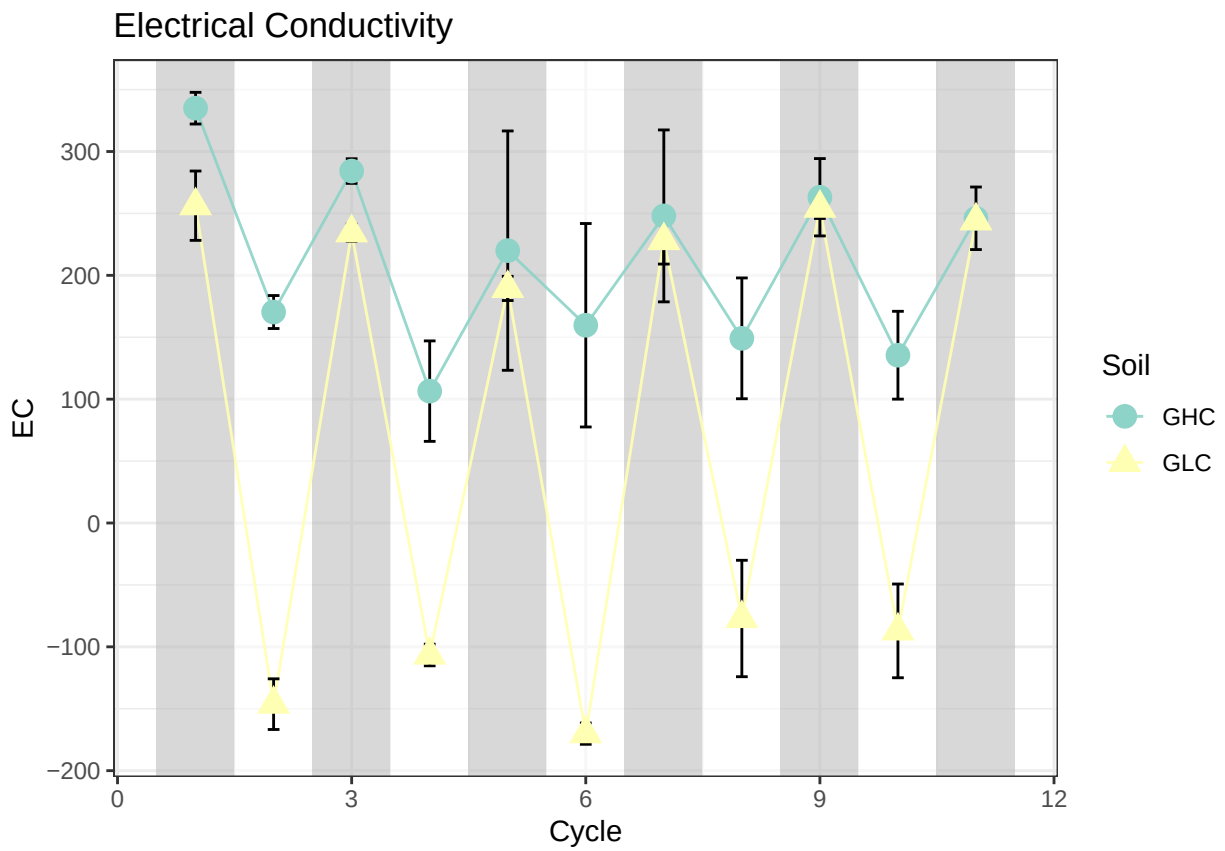
```
## The following objects are masked from 'package:dplyr':
##
##     arrange, count, desc, failwith, id, mutate, rename, summarise,
##     summarize
```

```
tgc2 <- summarySE(Map_T, measurevar="EC", groupvars=c("Soil","Condition","Cycle"))

#pH

pH.plot =
  ggplot() +
  geom_rect(data = rects, aes(xmin = xstart, xmax = xend, ymin = -Inf,
                              ymax = Inf, fill = Condition),
                              fill = rects$Colors, alpha = 0.6) +
  geom_line(data = tgc, aes(x=Cycle, y=pH, color=Soil), alpha = 0.9) +
  geom_errorbar(data = tgc,
                aes(x=Cycle,ymin=pH-ci, ymax=pH+ci),
                colour="black", width=.15) +
  geom_point(data = tgc, aes(x=Cycle, y=pH, color=Soil, shape = Soil), size=4)+
  ggtitle("pH")
################################################################################
#EC

EC.plot =
```

```
ggplot() +
  geom_rect(data = rects, aes(xmin = xstart, xmax = xend, ymin = -Inf,
                              ymax = Inf, fill = Condition),
                              fill = rects$Colors, alpha = 0.6) +
  geom_line(data = tgc2, aes(x=Cycle, y=EC, color=Soil), alpha = 0.9) +
  geom_errorbar(data = tgc2,
                aes(x=Cycle,ymin=EC-ci, ymax=EC+ci),
                colour="black", width=.15) +
  geom_point(data = tgc2, aes(x=Cycle, y=EC, color = Soil, shape = Soil), size=4)+
  ggtitle("Electrical Conductivity")

#plotting both
pH.plot
```




```
EC.plot
```



Those plots will compose a figure with alpha diversity.

Dada2 pipeline was made on a separeted routine. Please see "dada2_pipeline" file to more details.

Here we will use the dada2 resulting files with the seqtab and the tax table to downstream analysis.

#Fourth step: importing files into a phyloseq object

```
library(phyloseq)

seqtab = readRDS("seqtab_final_new.rds")
taxa = readRDS("taxtable_new.rds")
map <- "mapall.txt"

ps <- phyloseq(otu_table(seqtab, taxa_are_rows=FALSE), tax_table(taxa))

sample_metadata = import_qiime_sample_data(map)

input =merge_phyloseq(ps, sample_metadata)
input
```

```
## phyloseq-class experiment-level object
## otu_table()   OTU Table:         [ 19807 taxa and 86 samples ]
## sample_data() Sample Data:       [ 86 samples by 10 sample variables ]
## tax_table()   Taxonomy Table:    [ 19807 taxa by 6 taxonomic ranks ]
```

Everything looks fine.

#Fifth step: Inspctioning phyloseq object

```
microbiome::summarize_phyloseq(input)
```

```
## Compositional = NO2

## 1] Min. number of reads = 40282] Max. number of reads = 888913] Total number of reads = 27681374] Av
##         (i.e. exactly one read detected across all samples)010] Number of sample variables are: 10NA

## [[1]]
## [1] "1] Min. number of reads = 4028"
##
## [[2]]
## [1] "2] Max. number of reads = 88891"
##
## [[3]]
## [1] "3] Total number of reads = 2768137"
##
## [[4]]
## [1] "4] Average number of reads = 32187.6395348837"
##
## [[5]]
## [1] "5] Median number of reads = 29336"
##
## [[6]]
## [1] "7] Sparsity = 0.962835549095281"
##
## [[7]]
## [1] "6] Any OTU sum to 1 or less? NO"
##
## [[8]]
## [1] "8] Number of singletons = 0"
##
## [[9]]
## [1] "9] Percent of OTUs that are singletons \n        (i.e. exactly one read detected across all samp
##
## [[10]]
## [1] "10] Number of sample variables are: 10"
##
## [[11]]
##  [1] "NAMES"      "Soil"       "Sequencing" "Barcode"    "Treatment"
##  [6] "Cycle"      "Month"      "Condition"  "EC"         "pH"
```

#Sixth step: Read distribution and Good's Coverage

```
#Reads distribution

library(data.table)
```

```
##
## Attaching package: 'data.table'
```

```
## The following object is masked from 'package:purrr':
##
##     transpose

## The following objects are masked from 'package:dplyr':
##
##     between, first, last
```

```
library(knitr)

sdt = data.table(as(sample_data(input), "data.frame"),
                 TotalReads = sample_sums(input), keep.rownames = TRUE)

#Coverage calculation
#Normal calculations use singletons. As singletons are commonly excluded in dada2 pipeline,
# I changed the function to include ASVs with abundance equal to 2.

Good <-function (a) {
  freq.one <-colSums((a)==2)
  freq.one
  num.seq<-colSums(a)
  num.seq.mat<-as.matrix(num.seq)
  nseq<-num.seq.mat
  good.res<-(1-(freq.one/nseq))
  good.res
  return(good.res)
}

Coverage=Good(t(otu_table(ps)))

tab1= cbind(sdt,Coverage)
names(tab1)[names(tab1) == "V1"] <- "Coverage"

#Showing the results
kable(tab1[ ,c(2:13)], caption = "Distribution of reads per sample")
```

Table 1: Distribution of reads per sample

NAMES	Soil	Sequencing	Barcode	Treatment	Cycle	Month	Condition	EC	pH	TotalReads	Coverage
HC01	GHC	1	21	drained	1	march	01_Drained	329.00	4.48	21118	0.9997632
HC02	GHC	1	22	drained	1	march	01_Drained	345.00	4.45	37690	0.9997612
HC03	GHC	1	23	drained	1	march	01_Drained	338.00	4.48	6866	0.9994174
HC04	GHC	1	24	drained	1	march	01_Drained	328.00	4.40	5011	0.9990022
HC06	GHC	1	30	flooded	2	april	02_Flooded	170.00	4.75	5682	0.9994720
HC07	GHC	1	31	flooded	2	april	02_Flooded	175.90	4.64	6989	0.9991415
HC08	GHC	1	32	flooded	2	april	02_Flooded	165.20	4.69	6128	0.9991841
HC09	GHC	2	1	drained	3	may	03_Drained	288.10	4.58	48819	0.9995289
HC10	GHC	2	2	drained	3	may	03_Drained	290.10	4.46	44041	0.9993869
HC11	GHC	2	3	drained	3	may	03_Drained	282.50	4.44	42078	0.9994296
HC12	GHC	2	4	drained	3	may	03_Drained	276.20	4.53	45064	0.9994896
HC13	GHC	2	9	flooded	4	june	04_Flooded	120.90	5.04	54672	0.9993232
HC14	GHC	2	10	flooded	4	june	04_Flooded	129.00	5.03	27501	0.9993455

NAMES	Soil	Sequencing	Barcode	Treatment	Cycle	Month	Condition	EC	pH	TotalReads	Coverage
HC15	GHC	2	11	flooded	4	june	04_Flooded	104.70	5.01	47562	0.9994323
HC16	GHC	2	12	flooded	4	june	04_Flooded	71.40	5.12	53954	0.9997220
HC17	GHC	2	17	drained	5	july	05_Drained	130.10	4.94	36656	0.9991816
HC18	GHC	2	18	drained	5	july	05_Drained	236.00	4.89	42934	0.9992780
HC19	GHC	2	19	drained	5	july	05_Drained	258.30	4.72	88891	0.9997188
HC20	GHC	2	20	drained	5	july	05_Drained	255.50	4.61	61357	0.9995925
HC21	GHC	2	25	flooded	6	august	06_Flooded	89.20	5.47	50778	0.9993698
HC22	GHC	2	26	flooded	6	august	06_Flooded	153.00	5.33	40425	0.9993074
HC23	GHC	2	27	flooded	6	august	06_Flooded	197.50	5.32	49335	0.9992500
HC24	GHC	2	28	flooded	6	august	06_Flooded	199.10	5.33	45187	0.9994025
HC25	GHC	3	5	drained	7	september	07_Drained	188.02	4.59	25414	0.9990950
HC26	GHC	3	22	drained	7	september	07_Drained	243.50	4.26	35479	0.9994927
HC27	GHC	3	23	drained	7	september	07_Drained	277.80	4.22	25420	0.9994099
HC28	GHC	3	24	drained	7	september	07_Drained	282.70	4.33	23491	0.9993615
HC29	GHC	3	26	flooded	8	october	08_Flooded	145.30	5.01	18755	0.9995201
HC30	GHC	3	27	flooded	8	october	08_Flooded	126.20	4.87	26522	0.9992082
HC31	GHC	3	28	flooded	8	october	08_Flooded	193.50	4.84	22337	0.9992837
HC32	GHC	3	29	flooded	8	october	08_Flooded	131.50	5.14	26926	0.9994058
HC33	GHC	3	30	drained	9	november	09_Drained	266.60	4.66	18489	0.9994051
HC34	GHC	3	31	drained	9	november	09_Drained	267.30	5.18	23813	0.9992441
HC35	GHC	3	13	drained	9	november	09_Drained	235.80	4.76	22531	0.9991123
HC36	GHC	3	32	drained	9	november	09_Drained	282.60	4.67	19999	0.9992500
HC37	GHC	3	33	flooded	10	december	10_Flooded	132.00	5.01	29712	0.9993942
HC38	GHC	3	34	flooded	10	december	10_Flooded	122.00	4.97	25305	0.9994467
HC39	GHC	3	35	flooded	10	december	10_Flooded	168.00	4.94	28960	0.9991367
HC40	GHC	3	36	flooded	10	december	10_Flooded	120.00	5.24	35443	0.9994639
HC41	GHC	3	37	drained	11	january	11_Drained	225.80	4.86	40489	0.9995554
HC42	GHC	3	38	drained	11	january	11_Drained	246.60	4.76	43549	0.9995637
HC43	GHC	3	39	drained	11	january	11_Drained	247.30	5.22	28451	0.9995431
HC44	GHC	3	40	drained	11	january	11_Drained	264.60	4.77	44304	0.9995711
LC01	GLC	1	17	drained	1	march	01_Drained	253.00	5.68	11868	0.9997472
LC02	GLC	1	18	drained	1	march	01_Drained	282.00	5.98	7490	0.9989319
LC03	GLC	1	19	drained	1	march	01_Drained	246.00	6.01	8014	0.9995009
LC04	GLC	1	20	drained	1	march	01_Drained	244.00	6.21	4028	0.9997517
LC05	GLC	1	25	flooded	2	april	02_Flooded	-127.60	6.19	5005	0.9992008
LC06	GLC	1	26	flooded	2	april	02_Flooded	-148.10	6.38	4858	0.9995883
LC07	GLC	1	27	flooded	2	april	02_Flooded	-155.00	6.45	4421	0.9993214
LC08	GLC	1	28	flooded	2	april	02_Flooded	-154.50	6.47	4174	0.9995208
LC10	GLC	2	6	drained	3	may	03_Drained	233.60	5.38	55885	0.9996600
LC11	GLC	2	7	drained	3	may	03_Drained	232.10	5.57	46776	0.9993586
LC12	GLC	2	8	drained	3	may	03_Drained	237.50	5.62	44907	0.9992874
LC13	GLC	2	13	flooded	4	june	04_Flooded	-113.40	5.93	56489	0.9996637
LC14	GLC	2	14	flooded	4	june	04_Flooded	-105.50	6.20	56378	0.9994501
LC15	GLC	2	15	flooded	4	june	04_Flooded	-100.10	6.29	44020	0.9996592

NAMES	Soil	Sequencing	Barcode	Treatment	Cycle	Month	Condition	EC	pH	TotalReads	Coverage
LC16	GLC	2	16	flooded	4	june	04_Flooded	-107.40	6.36	41467	0.9994212
LC17	GLC	2	21	drained	5	july	05_Drained	195.30	5.93	34496	0.9994202
LC18	GLC	2	22	drained	5	july	05_Drained	191.80	5.98	37094	0.9995147
LC19	GLC	2	23	drained	5	july	05_Drained	189.70	6.11	43701	0.9995652
LC20	GLC	2	24	drained	5	july	05_Drained	180.90	6.09	49438	0.9995752
LC21	GLC	2	29	flooded	6	august	06_Flooded	-167.50	6.46	35398	0.9996045
LC22	GLC	2	30	flooded	6	august	06_Flooded	-173.40	6.47	43046	0.9991637
LC23	GLC	2	31	flooded	6	august	06_Flooded	-175.70	6.46	62456	0.9995677
LC24	GLC	2	32	flooded	6	august	06_Flooded	-163.90	6.45	70097	0.9995007
LC25	GLC	3	1	drained	7	september	07_Drained	242.40	5.52	27188	0.9995954
LC26	GLC	3	2	drained	7	september	07_Drained	232.40	5.77	25877	0.9995363
LC27	GLC	3	3	drained	7	september	07_Drained	222.80	5.93	28367	0.9995417
LC28	GLC	3	4	drained	7	september	07_Drained	214.80	5.99	25263	0.9992875
LC29	GLC	3	25	flooded	8	october	08_Flooded	-166.70	6.81	22159	0.9994585
LC30	GLC	3	6	flooded	8	october	08_Flooded	-156.10	6.06	28287	0.9992223
LC31	GLC	3	7	flooded	8	october	08_Flooded	-120.90	6.58	22785	0.9994294
LC32	GLC	3	8	flooded	8	october	08_Flooded	-164.80	6.54	27545	0.9993828
LC33	GLC	3	9	drained	9	november	09_Drained	261.00	5.77	26221	0.9993135
LC34	GLC	3	10	drained	9	november	09_Drained	255.60	5.76	24588	0.9992679
LC35	GLC	3	11	drained	9	november	09_Drained	249.00	5.89	23007	0.9990003
LC36	GLC	3	12	drained	9	november	09_Drained	251.50	5.77	19294	0.9992744
LC37	GLC	3	14	flooded	10	december	10_Flooded	-166.10	6.36	34354	0.9992141
LC38	GLC	3	15	flooded	10	december	10_Flooded	-120.90	6.58	25746	0.9993009
LC39	GLC	3	16	flooded	10	december	10_Flooded	-184.80	6.54	20873	0.9989939
LC40	GLC	3	17	flooded	10	december	10_Flooded	-176.70	6.81	36503	0.9995343
LC41	GLC	3	18	drained	11	january	11_Drained	241.00	5.87	33921	0.9992630
LC42	GLC	3	19	drained	11	january	11_Drained	245.60	5.86	31391	0.9992354
LC43	GLC	3	20	drained	11	january	11_Drained	248.00	5.91	31263	0.9991044
LC44	GLC	3	21	drained	11	january	11_Drained	241.50	5.87	47872	0.9995196

#Seventh step: Rarefaction and Compositional transformation

Some analysis will demand rarefacted data. Other ones, will demand clr-transformed. Here we early prepare both.

```
set.seed(2125)

#Minimum sample size is 2975; input is the initial phyloseq object.
#Rarefaction
inputR = rarefy_even_depth(input, sample.size = 4028, replace = FALSE)
```

```
## You set 'rngseed' to FALSE. Make sure you've set & recorded
##  the random seed of your session for reproducibility.
## See '?set.seed'
```

```
## ...

## 46990TUs were removed because they are no longer
## present in any sample after random subsampling

## ...
```

```
inputR
```

```
## phyloseq-class experiment-level object
## otu_table()   OTU Table:         [ 15108 taxa and 86 samples ]
## sample_data() Sample Data:       [ 86 samples by 10 sample variables ]
## tax_table()   Taxonomy Table:    [ 15108 taxa by 6 taxonomic ranks ]
```

```
#transform to compositional
#clr tranformation -- comes with a pseudocount to avoid zeros.
rice.comp <- microbiome::transform(input, "clr")
rice.comp
```

```
## phyloseq-class experiment-level object
## otu_table()   OTU Table:         [ 19807 taxa and 86 samples ]
## sample_data() Sample Data:       [ 86 samples by 10 sample variables ]
## tax_table()   Taxonomy Table:    [ 19807 taxa by 6 taxonomic ranks ]
```

```
#Subsetting

GLC = subset_samples(input, Soil == "GLC")
GLC
```

```
## phyloseq-class experiment-level object
## otu_table()   OTU Table:         [ 19807 taxa and 43 samples ]
## sample_data() Sample Data:       [ 43 samples by 10 sample variables ]
## tax_table()   Taxonomy Table:    [ 19807 taxa by 6 taxonomic ranks ]
```

```
GLC.comp <- microbiome::transform(GLC, "clr")

GHC = subset_samples(input, Soil == "GHC")
GHC
```

```
## phyloseq-class experiment-level object
## otu_table()   OTU Table:         [ 19807 taxa and 43 samples ]
## sample_data() Sample Data:       [ 43 samples by 10 sample variables ]
## tax_table()   Taxonomy Table:    [ 19807 taxa by 6 taxonomic ranks ]
```

```
GHC.comp <- microbiome::transform(GHC, "clr")

#Samples agregated in genus
GLC.ag = microbiome::aggregate_rare(GLC, level = "Genus", detection = 1/100, prevalence = 1/100)

GHC.ag = microbiome::aggregate_rare(GHC, level = "Genus", detection = 1/100, prevalence = 1/100)
```

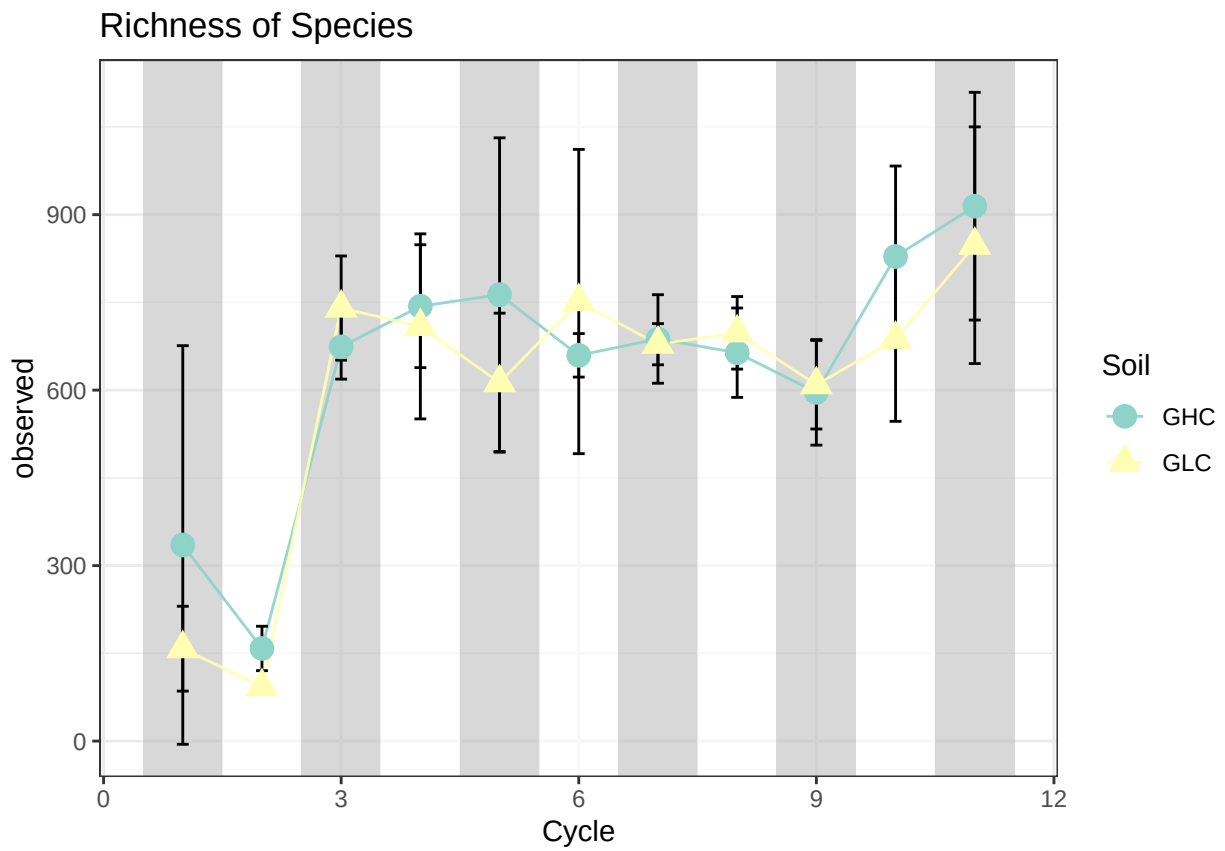
#Eight step: Alpha diversity

First the calculations

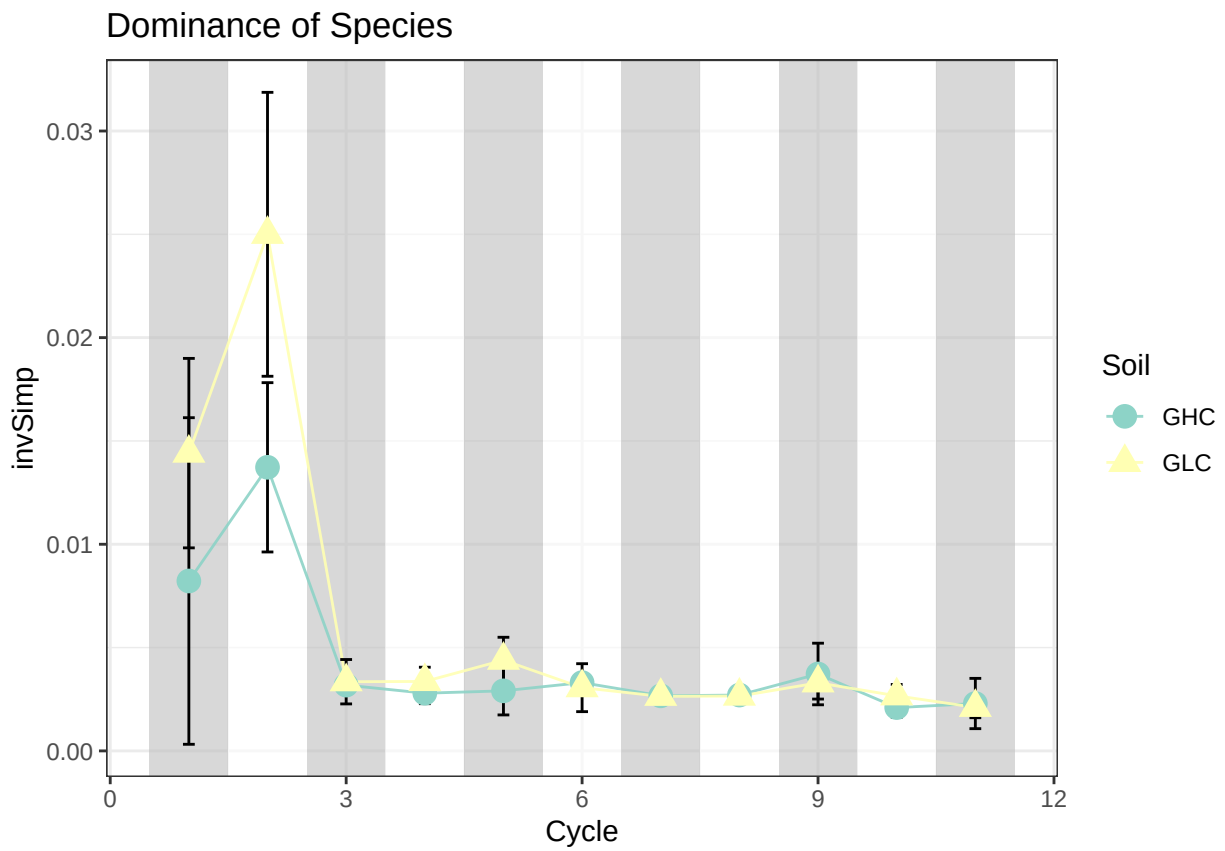
```
#Calculating Alpha Diversity

observed=microbiome::alpha(inputR, index = "all")
```

```
## Observed richness

## Other forms of richness

## Diversity

## Evenness

## Dominance

## Rarity
```

```
meta=microbiome::meta(inputR)

#Creating a file to plot a graph
alpha= cbind(observed,meta)

#And finally the plot

tgc3 <- summarySE(alpha, measurevar="observed", groupvars=c("Soil","Condition","Cycle"))

alpha.plot =
  ggplot() +
  geom_rect(data = rects, aes(xmin = xstart, xmax = xend, ymin = -Inf,
                              ymax = Inf, fill = Condition),
                              fill = rects$Colors, alpha = 0.6) +
  geom_line(data = tgc3, aes(x=Cycle, y=observed, color=Soil), alpha = 0.9) +
  geom_errorbar(data = tgc3,
                aes(x=Cycle,ymin=observed-ci, ymax=observed+ci),
                colour="black", width=.15) +
  geom_point(data = tgc3, aes(x=Cycle, y=observed, color=Soil, shape = Soil), size=4)+
  ggtitle("Richness of Species")

alpha.plot
```

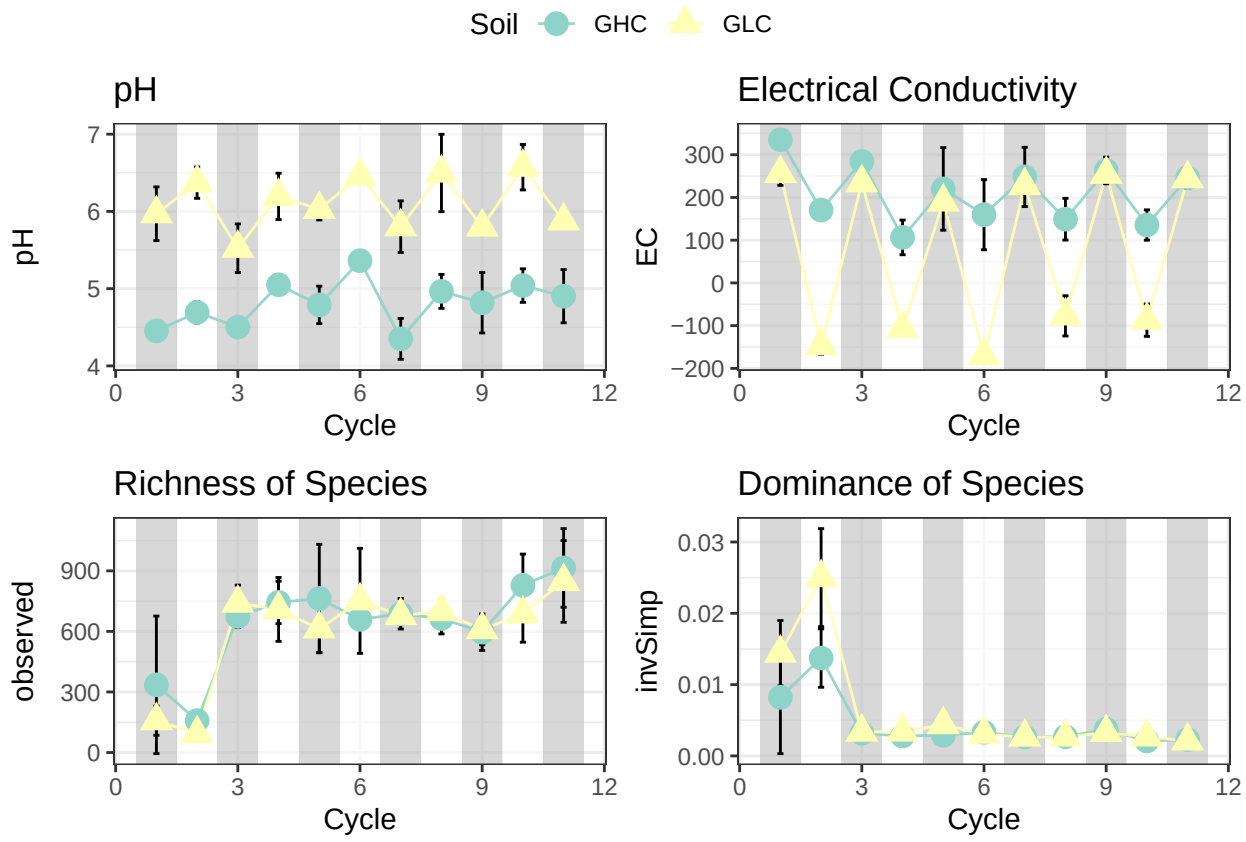



```
###### testing for dominance #######

tgc4 <- summarySE(alpha, measurevar="dominance_simpson", groupvars=c("Soil","Condition","Cycle"))

colnames(tgc4)[5] = "invSimp"

inv.plot =
  ggplot() +
  geom_rect(data = rects, aes(xmin = xstart, xmax = xend, ymin = -Inf,
                              ymax = Inf, fill = Condition),
                              fill = rects$Colors, alpha = 0.6) +
  geom_line(data = tgc4, aes(x=Cycle, y=invSimp, color=Soil), alpha = 0.9) +
  geom_errorbar(data = tgc4,
                aes(x=Cycle,ymin=invSimp-ci, ymax=invSimp+ci),
                colour="black", width=.15) +
  geom_point(data = tgc4, aes(x=Cycle, y=invSimp, color=Soil, shape = Soil), size=4)+
  ggtitle("Dominance of Species")

inv.plot
```



#Ninth step: plotting pH, EC and alpha diversity togther

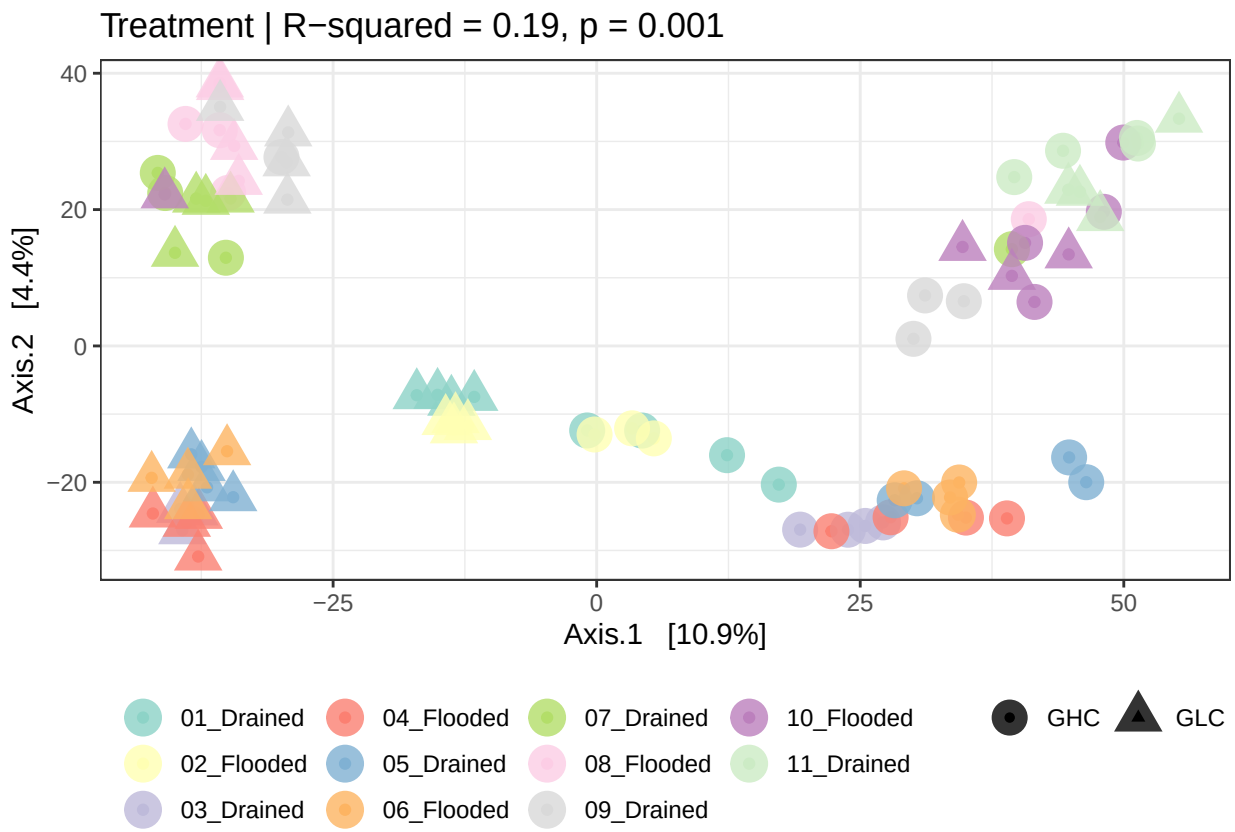
```
d = ggpubr::ggarrange(pH.plot, EC.plot, alpha.plot, inv.plot, ncol=2, nrow = 2, align = "hv", common.leg
d
```



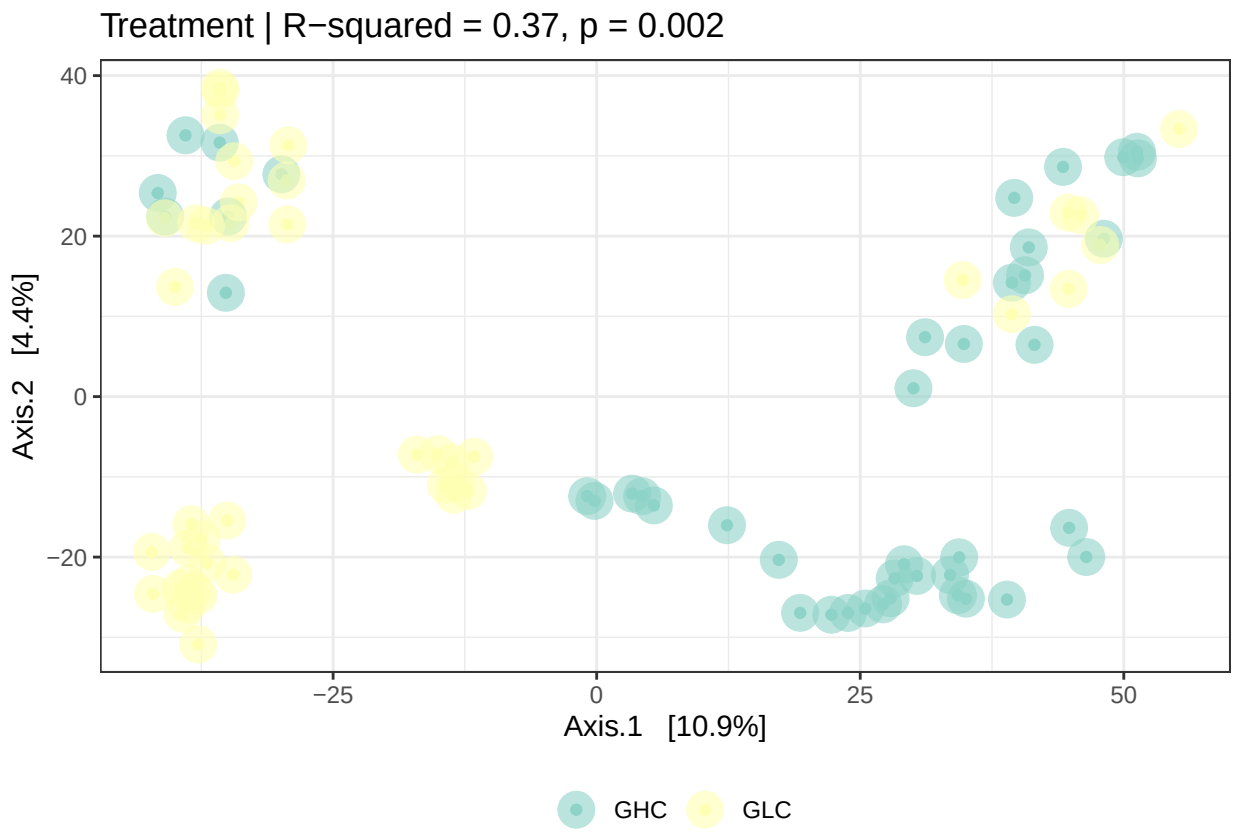

```
#dev.print(tiff, "longtime.tiff", width = 6, height = 6, units = "in")
```

#Tenth step: beta diversity

```
# Ploting beta diversity graph

#For different cycles

input_ord = ordinate(rice.comp, "PCoA" , "euclidean")
p3 = plot_ordination(rice.comp, input_ord, color = "Condition", title = "Treatment | R-squared = 0.19, p
p1 = p3 + geom_point(aes(shape = Soil), size = 6, alpha = 0.8) +
  theme(legend.position = "bottom", legend.title = element_blank())

p1
```




```
#For different soils

p5 = plot_ordination(rice.comp, input_ord, color = "Soil", title = "Treatment | R-squared = 0.37, p = 0
p5 = p5 + geom_point(size = 6, alpha = 0.6) +
  theme(legend.position = "bottom", legend.title = element_blank())
p5
```



#Tenth step: calculating the size of variance in beta diversity by permanova

```
#permanova
library(vegan)
```

```
## Carregando pacotes exigidos: permute
```

```
## Carregando pacotes exigidos: lattice
```

```
## This is vegan 2.5-7
```

```
df = as(sample_data(rice.comp), "data.frame")
ds = phyloseq::distance(rice.comp, method = "euclidean")
permanova = adonis(ds ~ Condition*Soil, data = df, permutations = 999)

library(knitr)
kable(permanova$aov.tab[ ,1:6], caption = "PERMANOVA - Condition*Soil")
```

Table 2: PERMANOVA - Condition*Soil

	Df	SumsOfSqs	MeanSqs	F.Model	R2	Pr(>F)
Condition	10	185562.07	18556.207	1.967831	0.1948431	0.001
Soil	1	40951.97	40951.966	4.342836	0.0430002	0.001

	Df	SumsOfSqs	MeanSqs	F.Model	R2	Pr(>F)
Condition:Soil	10	122347.24	12234.724	1.297457	0.1284665	0.001
Residuals	64	603505.58	9429.775	NA	0.6336902	NA
Total	85	952366.86	NA	NA	1.0000000	NA

```
###############
#and for groups separetelly

#GHC
df = as(sample_data(GHC.comp), "data.frame")
ds = phyloseq::distance(GHC.comp, method = "euclidean")
permanova = adonis(ds ~ Condition, data = df, permutations = 999)

library(knitr)
kable(permanova$aov.tab[ ,1:6], caption = "PERMANOVA - Condition, GHC")
```

Table 3: PERMANOVA - Condition, GHC

	Df	SumsOfSqs	MeanSqs	F.Model	R2	Pr(>F)
Condition	10	143700.1	14370.01	1.451327	0.3120243	0.001
Residuals	32	316841.3	9901.29	NA	0.6879757	NA
Total	42	460541.3	NA	NA	1.0000000	NA

```
#GLC
df = as(sample_data(GLC.comp), "data.frame")
ds = phyloseq::distance(GLC.comp, method = "euclidean")
permanova = adonis(ds ~ Condition, data = df, permutations = 999)

library(knitr)
kable(permanova$aov.tab[ ,1:6], caption = "PERMANOVA - Condition, GLC")
```

Table 4: PERMANOVA - Condition, GLC

	Df	SumsOfSqs	MeanSqs	F.Model	R2	Pr(>F)
Condition	10	147820.4	14782.041	1.841128	0.3652214	0.001
Residuals	32	256921.5	8028.797	NA	0.6347786	NA
Total	42	404741.9	NA	NA	1.0000000	NA

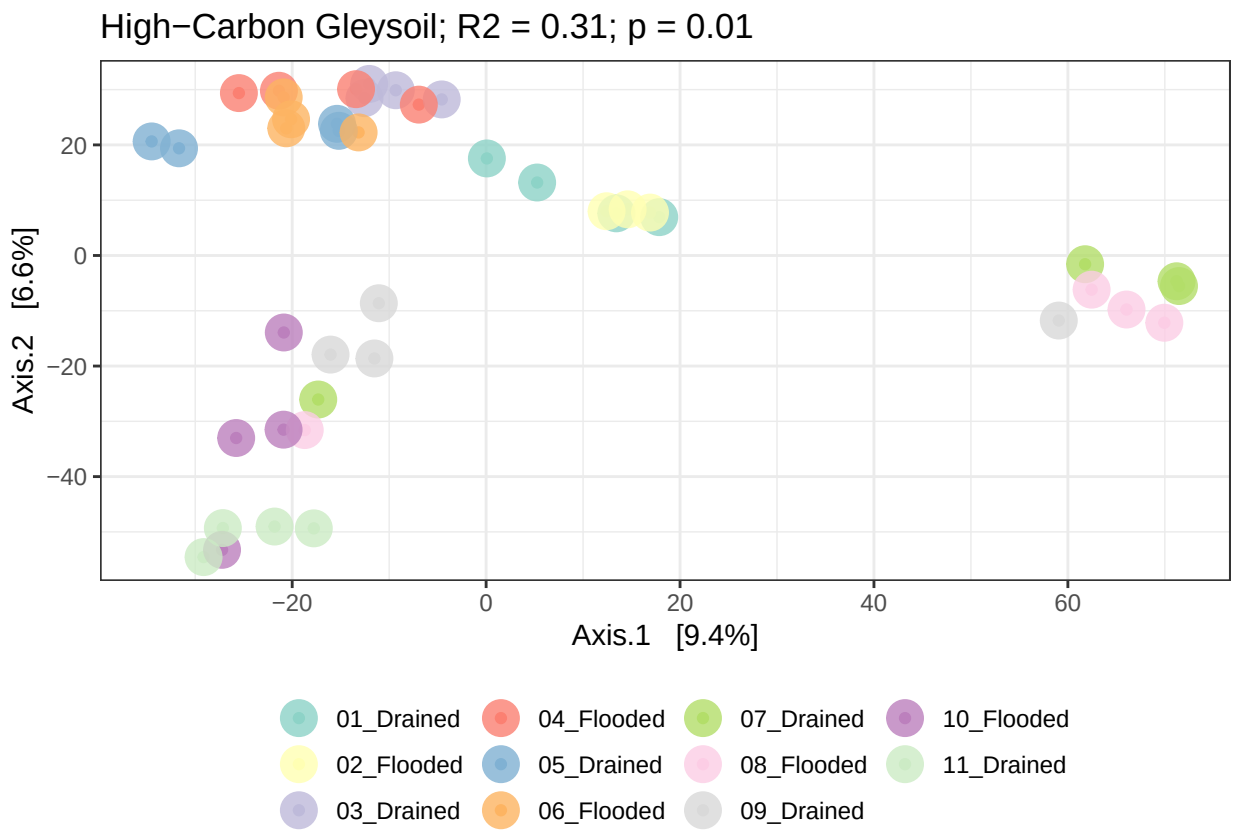
It seems the soil type influences a lot in the data. Let's break the dataset into two other ones.

##Tenth ponit one (?) step: Beta diversity for two groups separatelly

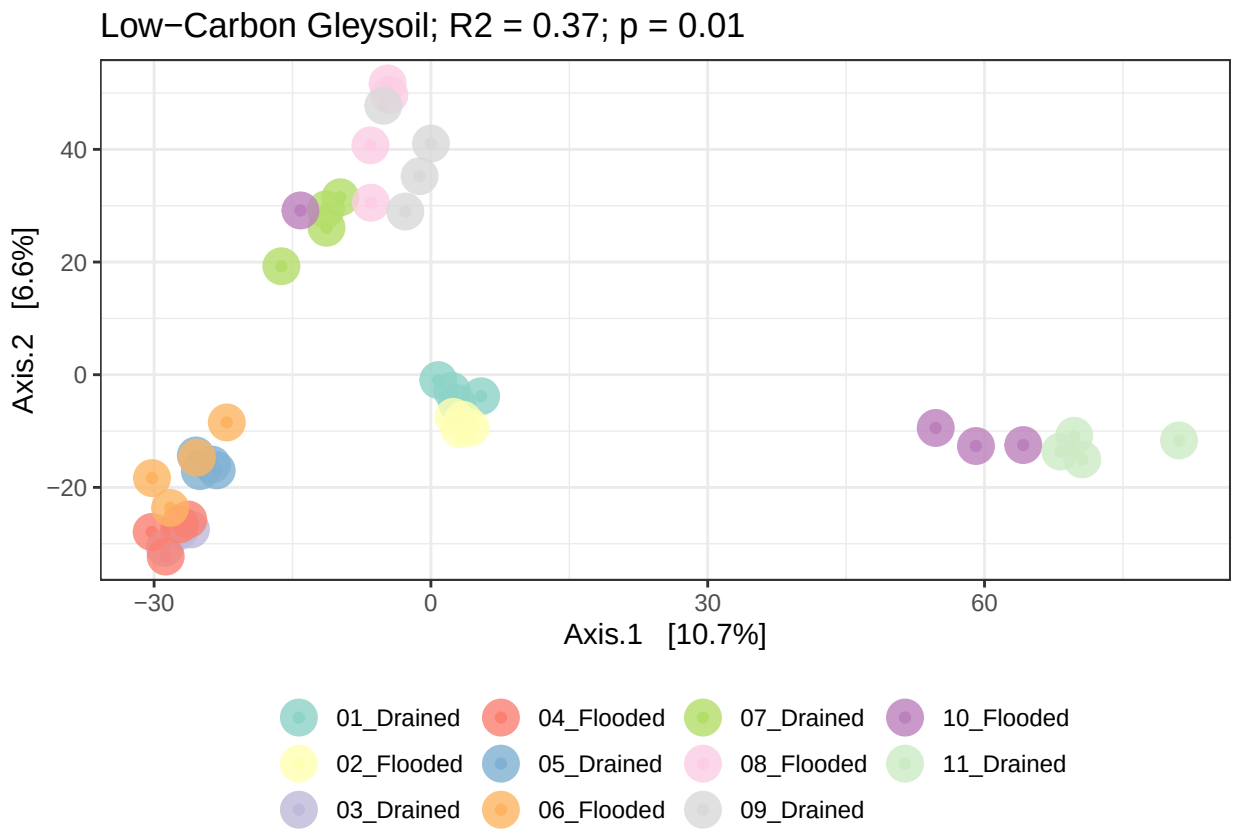
Now applying the beta div

```
#GHC

input_ord = ordinate(GHC.comp, "PCoA" , "euclidean")
p3 = plot_ordination(GHC.comp, input_ord, color = "Condition", title = "High-Carbon Gleysoil; R2 = 0.31
g1 = p3 + geom_point(size = 6, alpha = 0.8) +
```

```
  theme(legend.position = "bottom", legend.title = element_blank())

g1
```



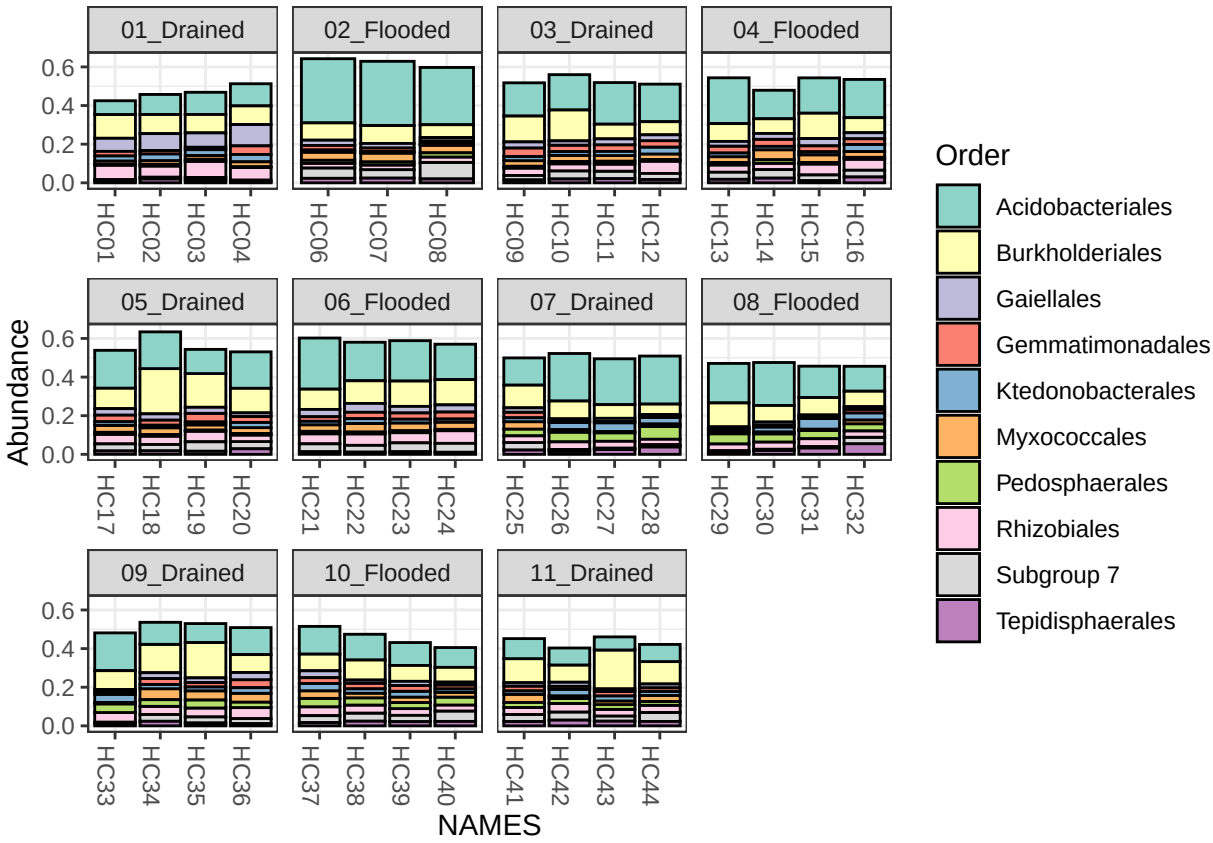

```
#GLC

input_ord = ordinate(GLC.comp, "PCoA" , "euclidean")
p3 = plot_ordination(GLC.comp, input_ord, color = "Condition", title = "Low-Carbon Gleysoil; R2 = 0.37;
g2 = p3 + geom_point(size = 6, alpha = 0.8) +
  theme(legend.position = "bottom", legend.title = element_blank())

g2
```



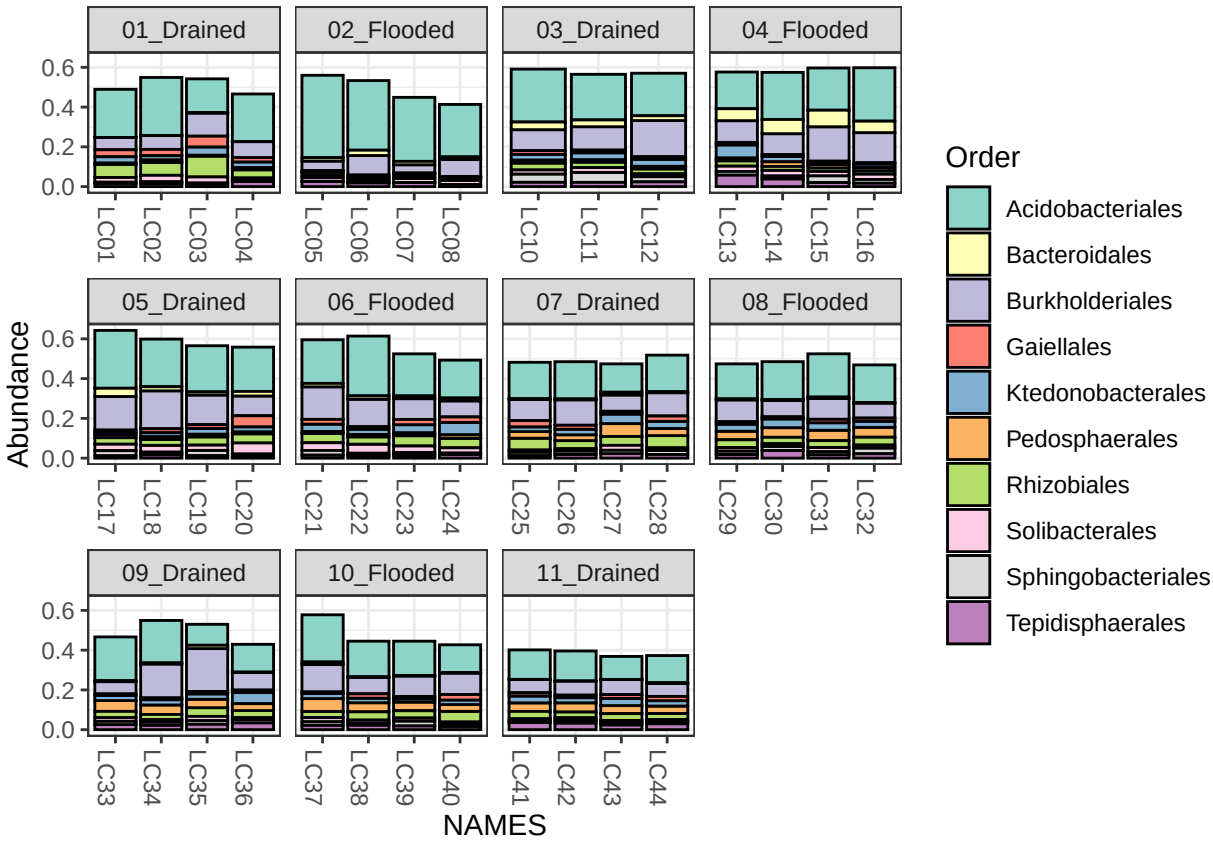
It seem that the microbiota is highly resilient.

#Eleventh step: Microbial distributtion of phyla during the cycles

```
#GHC
GHC.agg = microbiome::aggregate_rare(GHC, level = "Order", detection = 1/100, prevalence = 1/100)
GHC.top20 = names(sort(taxa_sums(GHC.agg), decreasing = T))[1:11]
ps.fam.GHC = transform_sample_counts(GHC.agg, function(OTU) OTU/sum(OTU))
ps.fam.GHC = prune_taxa(GHC.top20, ps.fam.GHC)
ps.fam.GHC = subset_taxa(ps.fam.GHC, Order != "Unknown")
plot_bar(ps.fam.GHC, x="NAMES", fill="Order") + facet_wrap(~Condition, scales="free_x")
```




```
#GLC
GLC.agg = microbiome::aggregate_rare(GLC, level = "Order", detection = 1/100, prevalence = 1/100)
GLC.top20 = names(sort(taxa_sums(GLC.agg), decreasing = T))[1:11]
ps.fam.GLC = transform_sample_counts(GLC.agg, function(OTU) OTU/sum(OTU))
ps.fam.GLC = prune_taxa(GLC.top20, ps.fam.GLC)
ps.fam.GLC = subset_taxa(ps.fam.GLC, Order != "Unknown")
plot_bar(ps.fam.GLC, x="NAMES", fill="Order") + facet_wrap(~Condition, scales="free_x")
```



#Twelfth step: differential abundance among cycles

We will run ALDEx2 to compare differences in abundance from cycle to cycle

```
##GHC##
#subsetting only the two first cycles
set.seed(300)

f1t2  = subset_samples(GHC, Cycle == "1" | Cycle == "2")
f2t3  = subset_samples(GHC, Cycle == "2" | Cycle == "3")
f3t4  = subset_samples(GHC, Cycle == "3" | Cycle == "4")
f4t5  = subset_samples(GHC, Cycle == "4" | Cycle == "5")
f5t6  = subset_samples(GHC, Cycle == "5" | Cycle == "6")
f6t7  = subset_samples(GHC, Cycle == "6" | Cycle == "7")
f7t8  = subset_samples(GHC, Cycle == "7" | Cycle == "8")
f8t9  = subset_samples(GHC, Cycle == "8" | Cycle == "9")
f9t10 = subset_samples(GHC, Cycle == "9" | Cycle == "10")
f10t11= subset_samples(GHC, Cycle == "10" | Cycle == "11")
f1t11 = subset_samples(GHC, Cycle == "1" | Cycle == "11")
```

And then the Aldex:

```
set.seed(300)

sam1.2 = microbiome::aggregate_rare(f1t2, level = "Genus",
                                    detection = 1/100, prevalence = 1/100)
```

```r
mi1.2    = as.data.frame((otu_table(sam1.2)))
var1.2   = sample_data(sam1.2)
treat1.2 = var1.2$Condition

library(ALDEx2)
```

```
## Carregando pacotes exigidos: zCompositions

## Carregando pacotes exigidos: MASS

##
## Attaching package: 'MASS'

## The following object is masked from 'package:dplyr':
##
##     select

## Carregando pacotes exigidos: NADA

## Carregando pacotes exigidos: survival

##
## Attaching package: 'NADA'

## The following object is masked from 'package:stats':
##
##     cor

## Carregando pacotes exigidos: truncnorm

## Carregando pacotes exigidos: Rfast

## Carregando pacotes exigidos: Rcpp

## Carregando pacotes exigidos: RcppZiggurat

##
## Attaching package: 'Rfast'

## The following object is masked from 'package:data.table':
##
##     transpose

## The following objects are masked from 'package:purrr':
##
##     is_integer, transpose

## The following object is masked from 'package:dplyr':
##
##     nth
```

```
x1.2 <- aldex(mi1.2, treat1.2, mc.samples=128, test="t", effect=TRUE,
            include.sample.summary=TRUE, denom="zero", verbose=TRUE)
```

```
## aldex.clr: generating Monte-Carlo instances and clr values

## operating in serial mode

## removed rows with sums equal to zero

## computing zero removal

## data format is OK

## dirichlet samples complete

## transformation complete

## aldex.ttest: doing t-test

## running tests for each MC instance:

## |------------(25%)----------(50%)----------(75%)----------|

## aldex.effect: calculating effect sizes

## operating in serial mode

## sanity check complete

## rab.all  complete

## rab.win  complete

## rab of samples complete

## within sample difference calculated

## between group difference calculated

## group summaries calculated

## effect size calculated

## summarizing output
```

```
## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 3.17356885559369, : provided 185 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 3.17356885559369, : provided 185 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 3.17356885559369, : provided 185 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 3.17356885559369, : provided 185 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 3.17356885559369, : provided 185 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 3.17356885559369, : provided 185 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 3.17356885559369, : provided 185 variables to replace 1 variables
```

```r
#Aldex do not include the "other" taxa
tax1.2 = subset_taxa(tax_table(sam1.2), Genus != "Other")

aldex1.2 = cbind(x1.2, tax1.2)
res1.2=(aldex1.2[(aldex1.2$we.ep<="0.01"),])

res1.2
```

```
##                rab.all rab.win.01_Drained rab.win.02_Flooded rab.sample.HC01
## Conexibacter 6.390934           7.448359           3.426480        3.173569
## Haliangium   5.274822           5.642565           3.328155        3.173569
## HSB OF53-F07 5.168243           6.093667           2.876464        3.173569
##              rab.sample.HC02 rab.sample.HC03 rab.sample.HC04 rab.sample.HC06
## Conexibacter        4.308763        5.874891        4.968183         2.37142
## Haliangium          4.308763        5.874891        4.968183         2.37142
## HSB OF53-F07        4.308763        5.874891        4.968183         2.37142
##              rab.sample.HC07 rab.sample.HC08  diff.btw  diff.win    effect
## Conexibacter        2.052295       -4.255394 -4.013604 0.8377775 -4.671426
## Haliangium          2.052295       -4.255394 -2.422741 0.7808528 -3.212178
## HSB OF53-F07        2.052295       -4.255394 -3.470450 1.3226619 -2.775464
##                   overlap       we.ep    we.eBH      wi.ep    wi.eBH
## Conexibacter 0.0003653998 0.001797528 0.119586 0.05714286 0.2484745
## Haliangium   0.0003653998 0.005730144 0.162070 0.05714286 0.2484745
## HSB OF53-F07 0.0003653998 0.008636181 0.175448 0.05714286 0.2484745
##                     Genus       unique
## Conexibacter Conexibacter Conexibacter
## Haliangium     Haliangium   Haliangium
## HSB OF53-F07 HSB OF53-F07 HSB OF53-F07
```

Cicle 2 to 3.

```r
set.seed(300)

sam2.3 = microbiome::aggregate_rare(f2t3, level = "Genus",
                                    detection = 1/100, prevalence = 1/100)
mi2.3    = as.data.frame((otu_table(sam2.3)))
var2.3   = sample_data(sam2.3)
treat2.3 = var2.3$Condition

library(ALDEx2)

x2.3 <- aldex(mi2.3, treat2.3, mc.samples=128, test="t", effect=TRUE,
            include.sample.summary=TRUE, denom="zero", verbose=TRUE)
```

```
## aldex.clr: generating Monte-Carlo instances and clr values

## operating in serial mode

## removed rows with sums equal to zero

## computing zero removal

## data format is OK

## dirichlet samples complete

## transformation complete

## aldex.ttest: doing t-test

## running tests for each MC instance:

## |------------(25%)----------(50%)----------(75%)----------|

## aldex.effect: calculating effect sizes

## operating in serial mode

## sanity check complete

## rab.all  complete

## rab.win  complete

## rab of samples complete

## within sample difference calculated

## between group difference calculated
```

```
## group summaries calculated

## effect size calculated

## summarizing output

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 2.2936073574079, : provided 211 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 2.2936073574079, : provided 211 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 2.2936073574079, : provided 211 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 2.2936073574079, : provided 211 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 2.2936073574079, : provided 211 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 2.2936073574079, : provided 211 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 2.2936073574079, : provided 211 variables to replace 1 variables
```

```
#Aldex do not include the "other" taxa
tax2.3 = subset_taxa(tax_table(sam2.3), Genus != "Other")

aldex2.3 = cbind(x2.3, tax2.3)
res2.3=(aldex2.3[(aldex2.3$we.ep<="0.01"),])

res2.3
```

```
##                    rab.all rab.win.02_Flooded rab.win.03_Drained
## Anaeromyxobacter  6.914836           5.792446           7.723628
## Conexibacter      5.872849           3.454407           6.792542
## Ellin6067         6.205349           4.464311           6.893877
## Gemmatimonas      7.284696           4.216775           7.677035
## Haliangium        6.466122           3.345234           7.024222
## HSB OF53-F07      5.505988           2.845372           6.639485
## MND1              4.134636           1.512727           5.147880
## Unknown          11.475377           9.382720          12.305437
##                  rab.sample.HC06 rab.sample.HC07 rab.sample.HC08
## Anaeromyxobacter        2.293607        2.155905       -5.022509
## Conexibacter            2.293607        2.155905       -5.022509
## Ellin6067               2.293607        2.155905       -5.022509
## Gemmatimonas            2.293607        2.155905       -5.022509
## Haliangium              2.293607        2.155905       -5.022509
## HSB OF53-F07            2.293607        2.155905       -5.022509
## MND1                    2.293607        2.155905       -5.022509
## Unknown                 2.293607        2.155905       -5.022509
```

```
##                  rab.sample.HC09 rab.sample.HC10 rab.sample.HC11
## Anaeromyxobacter        4.399378        5.504445        6.111627
## Conexibacter            4.399378        5.504445        6.111627
## Ellin6067               4.399378        5.504445        6.111627
## Gemmatimonas            4.399378        5.504445        6.111627
## Haliangium              4.399378        5.504445        6.111627
## HSB OF53-F07            4.399378        5.504445        6.111627
## MND1                    4.399378        5.504445        6.111627
## Unknown                 4.399378        5.504445        6.111627
##                  rab.sample.HC12 diff.btw  diff.win   effect      overlap
## Anaeromyxobacter        5.797486 1.920611 0.6998815 2.722559 0.0003653998
## Conexibacter            5.797486 3.280563 0.5806823 5.385805 0.0003653998
## Ellin6067               5.797486 2.405436 0.6985552 3.418829 0.0003653998
## Gemmatimonas            5.797486 3.403586 0.6897809 4.862206 0.0003653998
## Haliangium              5.797486 3.605462 0.6293966 5.719948 0.0003653998
## HSB OF53-F07            5.797486 3.694336 1.0775366 3.472796 0.0003653998
## MND1                    5.797486 3.577593 1.0137462 3.400728 0.0003653998
## Unknown                 5.797486 2.892235 0.6773917 4.133740 0.0003653998
##                            we.ep     we.eBH      wi.ep    wi.eBH            Genus
## Anaeromyxobacter 0.0078134328 0.10568308 0.05714286 0.1832249 Anaeromyxobacter
## Conexibacter     0.0006126286 0.02659301 0.05714286 0.1832249     Conexibacter
## Ellin6067        0.0029763553 0.05908710 0.05714286 0.1832249        Ellin6067
## Gemmatimonas     0.0040395512 0.06587390 0.05714286 0.1832249     Gemmatimonas
## Haliangium       0.0008240911 0.02745944 0.05714286 0.1832249       Haliangium
## HSB OF53-F07     0.0080853823 0.09098157 0.05714286 0.1832249     HSB OF53-F07
## MND1             0.0035642865 0.05925433 0.05714286 0.1832249             MND1
## Unknown          0.0013122369 0.03883489 0.05714286 0.1832249          Unknown
##                             unique
## Anaeromyxobacter Anaeromyxobacter
## Conexibacter         Conexibacter
## Ellin6067               Ellin6067
## Gemmatimonas         Gemmatimonas
## Haliangium             Haliangium
## HSB OF53-F07         HSB OF53-F07
## MND1                         MND1
## Unknown                   Unknown
```

Cicle 3-4

```
set.seed(300)

sam3.4 = microbiome::aggregate_rare(f3t4, level = "Genus",
                                    detection = 1/100, prevalence = 1/100)
mi3.4    = as.data.frame((otu_table(sam3.4)))
var3.4   = sample_data(sam3.4)
treat3.4 = var3.4$Condition

library(ALDEx2)

x3.4 <- aldex(mi3.4, treat3.4, mc.samples=128, test="t", effect=TRUE,
            include.sample.summary=TRUE, denom="zero", verbose=TRUE)
```

```
## aldex.clr: generating Monte-Carlo instances and clr values
```

```
## operating in serial mode

## removed rows with sums equal to zero

## computing zero removal

## data format is OK

## dirichlet samples complete

## transformation complete

## aldex.ttest: doing t-test

## running tests for each MC instance:

## |------------(25%)----------(50%)----------(75%)----------|

## aldex.effect: calculating effect sizes

## operating in serial mode

## sanity check complete

## rab.all  complete

## rab.win  complete

## rab of samples complete

## within sample difference calculated

## between group difference calculated

## group summaries calculated

## effect size calculated

## summarizing output

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 4.42508161747255, : provided 266 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 4.42508161747255, : provided 266 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 4.42508161747255, : provided 266 variables to replace 1 variables
```

```
## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 4.42508161747255, : provided 266 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 4.42508161747255, : provided 266 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 4.42508161747255, : provided 266 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 4.42508161747255, : provided 266 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 4.42508161747255, : provided 266 variables to replace 1 variables
```

```
#Aldex do not include the "other" taxa
tax3.4 = subset_taxa(tax_table(sam3.4), Genus != "Other")

aldex3.4 = cbind(x3.4, tax3.4)
res3.4=(aldex3.4[(aldex3.4$we.ep<="0.01"),])

res3.4
```

```
##  [1] rab.all            rab.win.03_Drained rab.win.04_Flooded rab.sample.HC09
##  [5] rab.sample.HC10    rab.sample.HC11    rab.sample.HC12    rab.sample.HC13
##  [9] rab.sample.HC14    rab.sample.HC15    rab.sample.HC16    diff.btw
## [13] diff.win           effect             overlap            we.ep
## [17] we.eBH             wi.ep              wi.eBH             Genus
## [21] unique
## <0 linhas> (ou row.names de comprimento 0)
```

Cycle 4 to 5.

```
set.seed(300)

sam4.5 = microbiome::aggregate_rare(f4t5, level = "Genus",
                                    detection = 1/100, prevalence = 1/100)
mi4.5    = as.data.frame((otu_table(sam4.5)))
var4.5   = sample_data(sam4.5)
treat4.5 = var4.5$Condition

library(ALDEx2)

x4.5 <- aldex(mi4.5, treat4.5, mc.samples=128, test="t", effect=TRUE,
            include.sample.summary=TRUE, denom="zero", verbose=TRUE)
```

```
## aldex.clr: generating Monte-Carlo instances and clr values

## operating in serial mode

## removed rows with sums equal to zero

## computing zero removal
```

```
## data format is OK

## dirichlet samples complete

## transformation complete

## aldex.ttest: doing t-test

## running tests for each MC instance:

## |------------(25%)----------(50%)----------(75%)----------|

## aldex.effect: calculating effect sizes

## operating in serial mode

## sanity check complete

## rab.all  complete

## rab.win  complete

## rab of samples complete

## within sample difference calculated

## between group difference calculated

## group summaries calculated

## effect size calculated

## summarizing output

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 5.61064146076329, : provided 267 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 5.61064146076329, : provided 267 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 5.61064146076329, : provided 267 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 5.61064146076329, : provided 267 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 5.61064146076329, : provided 267 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
```

```
## 5.61064146076329, : provided 267 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 5.61064146076329, : provided 267 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 5.61064146076329, : provided 267 variables to replace 1 variables
```

```
#Aldex do not include the "other" taxa
tax4.5 = subset_taxa(tax_table(sam4.5), Genus != "Other")

aldex4.5 = cbind(x4.5, tax4.5)
res4.5=(aldex4.5[(aldex4.5$we.ep<="0.01"),])

res4.5
```

```
##                 rab.all rab.win.04_Flooded rab.win.05_Drained rab.sample.HC13
## Panacagrimonas 2.565161          -3.922215            4.28958        5.610641
##                rab.sample.HC14 rab.sample.HC15 rab.sample.HC16 rab.sample.HC17
## Panacagrimonas        4.878357        4.324157        6.635482        5.457002
##                rab.sample.HC18 rab.sample.HC19 rab.sample.HC20 diff.btw
## Panacagrimonas        5.112288        4.070787        5.874559 8.295014
##                diff.win   effect     overlap      we.ep    we.eBH      wi.ep
## Panacagrimonas 2.175534 3.971561 0.000274075 0.00733676 0.4139888 0.02857143
##                   wi.eBH          Genus         unique
## Panacagrimonas 0.5537997 Panacagrimonas Panacagrimonas
```

Cycle 5 to 6.

```
set.seed(300)

sam5.6 = microbiome::aggregate_rare(f5t6, level = "Genus",
                                    detection = 1/100, prevalence = 1/100)
mi5.6    = as.data.frame((otu_table(sam5.6)))
var5.6   = sample_data(sam5.6)
treat5.6 = var5.6$Condition

library(ALDEx2)

x5.6 <- aldex(mi5.6, treat5.6, mc.samples=128, test="t", effect=TRUE,
            include.sample.summary=TRUE, denom="zero", verbose=TRUE)
```

```
## aldex.clr: generating Monte-Carlo instances and clr values

## operating in serial mode

## removed rows with sums equal to zero

## computing zero removal

## data format is OK
```

```
## dirichlet samples complete

## transformation complete

## aldex.ttest: doing t-test

## running tests for each MC instance:

## |------------(25%)----------(50%)----------(75%)----------|

## aldex.effect: calculating effect sizes

## operating in serial mode

## sanity check complete

## rab.all  complete

## rab.win  complete

## rab of samples complete

## within sample difference calculated

## between group difference calculated

## group summaries calculated

## effect size calculated

## summarizing output

## Warning in ‘[<-.data.frame‘(‘*tmp*‘, , nm, value = structure(list(X1 =
## 5.46113402441628, : provided 252 variables to replace 1 variables

## Warning in ‘[<-.data.frame‘(‘*tmp*‘, , nm, value = structure(list(X1 =
## 5.46113402441628, : provided 252 variables to replace 1 variables

## Warning in ‘[<-.data.frame‘(‘*tmp*‘, , nm, value = structure(list(X1 =
## 5.46113402441628, : provided 252 variables to replace 1 variables

## Warning in ‘[<-.data.frame‘(‘*tmp*‘, , nm, value = structure(list(X1 =
## 5.46113402441628, : provided 252 variables to replace 1 variables

## Warning in ‘[<-.data.frame‘(‘*tmp*‘, , nm, value = structure(list(X1 =
## 5.46113402441628, : provided 252 variables to replace 1 variables

## Warning in ‘[<-.data.frame‘(‘*tmp*‘, , nm, value = structure(list(X1 =
## 5.46113402441628, : provided 252 variables to replace 1 variables

## Warning in ‘[<-.data.frame‘(‘*tmp*‘, , nm, value = structure(list(X1 =
## 5.46113402441628, : provided 252 variables to replace 1 variables

## Warning in ‘[<-.data.frame‘(‘*tmp*‘, , nm, value = structure(list(X1 =
## 5.46113402441628, : provided 252 variables to replace 1 variables
```

```r
#Aldex do not include the "other" taxa
tax5.6 = subset_taxa(tax_table(sam5.6), Genus != "Other")

aldex5.6 = cbind(x5.6, tax5.6)
res5.6=(aldex5.6[(aldex5.6$we.ep<="0.01"),])

res5.6
```

```
##  [1] rab.all            rab.win.05_Drained rab.win.06_Flooded rab.sample.HC17
##  [5] rab.sample.HC18    rab.sample.HC19    rab.sample.HC20    rab.sample.HC21
##  [9] rab.sample.HC22    rab.sample.HC23    rab.sample.HC24    diff.btw
## [13] diff.win           effect             overlap            we.ep
## [17] we.eBH             wi.ep              wi.eBH             Genus
## [21] unique
## <0 linhas> (ou row.names de comprimento 0)
```

Cycle 6 to 7.

```r
set.seed(300)

sam6.7 = microbiome::aggregate_rare(f6t7, level = "Genus",
                                    detection = 1/100, prevalence = 1/100)
mi6.7    = as.data.frame((otu_table(sam6.7)))
var6.7   = sample_data(sam6.7)
treat6.7 = var6.7$Condition

library(ALDEx2)

x6.7 <- aldex(mi6.7, treat6.7, mc.samples=128, test="t", effect=TRUE,
            include.sample.summary=TRUE, denom="zero", verbose=TRUE)
```

```
## aldex.clr: generating Monte-Carlo instances and clr values

## operating in serial mode

## removed rows with sums equal to zero

## computing zero removal

## data format is OK

## dirichlet samples complete

## transformation complete

## aldex.ttest: doing t-test

## running tests for each MC instance:

## |-----------(25%)----------(50%)----------(75%)----------|
```

```
## aldex.effect: calculating effect sizes

## operating in serial mode

## sanity check complete

## rab.all  complete

## rab.win  complete

## rab of samples complete

## within sample difference calculated

## between group difference calculated

## group summaries calculated

## effect size calculated

## summarizing output

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 5.48106098371264, : provided 258 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 5.48106098371264, : provided 258 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 5.48106098371264, : provided 258 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 5.48106098371264, : provided 258 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 5.48106098371264, : provided 258 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 5.48106098371264, : provided 258 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 5.48106098371264, : provided 258 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 5.48106098371264, : provided 258 variables to replace 1 variables
```

```r
#Aldex do not include the "other" taxa
tax6.7 = subset_taxa(tax_table(sam6.7), Genus != "Other")

aldex6.7 = cbind(x6.7, tax6.7)
res6.7=(aldex6.7[(aldex6.7$we.ep<="0.01"),])

res6.7
```

```
##                           rab.all rab.win.06_Flooded rab.win.07_Drained
## Candidatus Omnitrophus 0.4476087          -4.257351           3.179411
## Conexibacter           6.2560672           7.221650           4.733206
##                        rab.sample.HC21 rab.sample.HC22 rab.sample.HC23
## Candidatus Omnitrophus        5.481061        5.271854        5.041086
## Conexibacter                  5.481061        5.271854        5.041086
##                        rab.sample.HC24 rab.sample.HC25 rab.sample.HC26
## Candidatus Omnitrophus        4.624775        4.519392        2.766062
## Conexibacter                  4.624775        4.519392        2.766062
##                        rab.sample.HC27 rab.sample.HC28  diff.btw  diff.win
## Candidatus Omnitrophus        3.664321        4.381308  7.567849 3.2024730
## Conexibacter                  3.664321        4.381308 -2.415529 0.7093953
##                           effect     overlap       we.ep    we.eBH      wi.ep
## Candidatus Omnitrophus  2.313938 0.000274075 0.009739595 0.2351953 0.02857143
## Conexibacter           -3.462899 0.000274075 0.001125109 0.1479184 0.02857143
##                           wi.eBH                  Genus                 unique
## Candidatus Omnitrophus 0.2585375 Candidatus Omnitrophus Candidatus Omnitrophus
## Conexibacter           0.2585375           Conexibacter           Conexibacter
```

Cycle 7 to 8.

```
set.seed(300)

sam7.8 = microbiome::aggregate_rare(f7t8, level = "Genus",
                                    detection = 1/100, prevalence = 1/100)
mi7.8    = as.data.frame((otu_table(sam7.8)))
var7.8   = sample_data(sam7.8)
treat7.8 = var7.8$Condition

library(ALDEx2)

x7.8 <- aldex(mi7.8, treat7.8, mc.samples=128, test="t", effect=TRUE,
            include.sample.summary=TRUE, denom="zero", verbose=TRUE)
```

```
## aldex.clr: generating Monte-Carlo instances and clr values

## operating in serial mode

## removed rows with sums equal to zero

## computing zero removal

## data format is OK

## dirichlet samples complete

## transformation complete

## aldex.ttest: doing t-test

## running tests for each MC instance:
```

```
## |------------(25%)----------(50%)----------(75%)----------|

## aldex.effect: calculating effect sizes

## operating in serial mode

## sanity check complete

## rab.all  complete

## rab.win  complete

## rab of samples complete

## within sample difference calculated

## between group difference calculated

## group summaries calculated

## effect size calculated

## summarizing output

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 4.55226100136081, : provided 239 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 4.55226100136081, : provided 239 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 4.55226100136081, : provided 239 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 4.55226100136081, : provided 239 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 4.55226100136081, : provided 239 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 4.55226100136081, : provided 239 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 4.55226100136081, : provided 239 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 4.55226100136081, : provided 239 variables to replace 1 variables
```

```
#Aldex do not include the "other" taxa
tax7.8 = subset_taxa(tax_table(sam7.8), Genus != "Other")

aldex7.8 = cbind(x7.8, tax7.8)
res7.8=(aldex7.8[(aldex7.8$we.ep<="0.01"),])

res7.8
```

```
##  [1] rab.all            rab.win.07_Drained rab.win.08_Flooded rab.sample.HC25
##  [5] rab.sample.HC26    rab.sample.HC27    rab.sample.HC28    rab.sample.HC29
##  [9] rab.sample.HC30    rab.sample.HC31    rab.sample.HC32    diff.btw
## [13] diff.win           effect             overlap            we.ep
## [17] we.eBH             wi.ep              wi.eBH             Genus
## [21] unique
## <0 linhas> (ou row.names de comprimento 0)
```

Cycle 8 to 9.

```
set.seed(300)

sam8.9 = microbiome::aggregate_rare(f8t9, level = "Genus",
                                    detection = 1/100, prevalence = 1/100)
mi8.9    = as.data.frame((otu_table(sam8.9)))
var8.9   = sample_data(sam8.9)
treat8.9 = var8.9$Condition

library(ALDEx2)

x8.9 <- aldex(mi8.9, treat8.9, mc.samples=128, test="t", effect=TRUE,
            include.sample.summary=TRUE, denom="zero", verbose=TRUE)
```

```
## aldex.clr: generating Monte-Carlo instances and clr values

## operating in serial mode

## removed rows with sums equal to zero

## computing zero removal

## data format is OK

## dirichlet samples complete

## transformation complete

## aldex.ttest: doing t-test

## running tests for each MC instance:

## |-----------(25%)----------(50%)----------(75%)----------|
```

```
## aldex.effect: calculating effect sizes

## operating in serial mode

## sanity check complete

## rab.all  complete

## rab.win  complete

## rab of samples complete

## within sample difference calculated

## between group difference calculated

## group summaries calculated

## effect size calculated

## summarizing output

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 0.451807870012331, : provided 247 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 0.451807870012331, : provided 247 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 0.451807870012331, : provided 247 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 0.451807870012331, : provided 247 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 0.451807870012331, : provided 247 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 0.451807870012331, : provided 247 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 0.451807870012331, : provided 247 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 0.451807870012331, : provided 247 variables to replace 1 variables
```

```
#Aldex do not include the "other" taxa
tax8.9 = subset_taxa(tax_table(sam8.9), Genus != "Other")

aldex8.9 = cbind(x8.9, tax8.9)
res8.9=(aldex8.9[(aldex8.9$we.ep<="0.01"),])

res8.9
```

```
##  [1] rab.all            rab.win.08_Flooded rab.win.09_Drained rab.sample.HC29
##  [5] rab.sample.HC30    rab.sample.HC31    rab.sample.HC32    rab.sample.HC33
##  [9] rab.sample.HC34    rab.sample.HC35    rab.sample.HC36    diff.btw
## [13] diff.win           effect             overlap            we.ep
## [17] we.eBH             wi.ep              wi.eBH             Genus
## [21] unique
## <0 linhas> (ou row.names de comprimento 0)
```

Cycle 9 to 10.

```
set.seed(300)

sam9.10 = microbiome::aggregate_rare(f9t10, level = "Genus",
                                     detection = 1/100, prevalence = 1/100)
mi9.10    = as.data.frame((otu_table(sam9.10)))
var9.10   = sample_data(sam9.10)
treat9.10 = var9.10$Condition

library(ALDEx2)

x9.10 <- aldex(mi9.10, treat9.10, mc.samples=128, test="t", effect=TRUE,
            include.sample.summary=TRUE, denom="zero", verbose=TRUE)
```

```
## aldex.clr: generating Monte-Carlo instances and clr values

## operating in serial mode

## removed rows with sums equal to zero

## computing zero removal

## data format is OK

## dirichlet samples complete

## transformation complete

## aldex.ttest: doing t-test

## running tests for each MC instance:

## |-----------(25%)----------(50%)----------(75%)----------|

## aldex.effect: calculating effect sizes

## operating in serial mode

## sanity check complete

## rab.all  complete
```

```
## rab.win  complete

## rab of samples complete

## within sample difference calculated

## between group difference calculated

## group summaries calculated

## effect size calculated

## summarizing output

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 3.54380777215577, : provided 243 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 3.54380777215577, : provided 243 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 3.54380777215577, : provided 243 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 3.54380777215577, : provided 243 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 3.54380777215577, : provided 243 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 3.54380777215577, : provided 243 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 3.54380777215577, : provided 243 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 3.54380777215577, : provided 243 variables to replace 1 variables
```

```
#Aldex do not include the "other" taxa
tax9.10 = subset_taxa(tax_table(sam9.10), Genus != "Other")

aldex9.10 = cbind(x9.10, tax9.10)
res9.10=(aldex9.10[(aldex9.10$we.ep<="0.01"),])

res9.10
```

```
##  [1] rab.all            rab.win.09_Drained rab.win.10_Flooded rab.sample.HC33
##  [5] rab.sample.HC34    rab.sample.HC35    rab.sample.HC36    rab.sample.HC37
##  [9] rab.sample.HC38    rab.sample.HC39    rab.sample.HC40    diff.btw
## [13] diff.win           effect             overlap            we.ep
## [17] we.eBH             wi.ep              wi.eBH             Genus
## [21] unique
## <0 linhas> (ou row.names de comprimento 0)
```

Cycle 10 to 11.

```
set.seed(300)

sam10.11 = microbiome::aggregate_rare(f10t11, level = "Genus",
                                      detection = 1/100, prevalence = 1/100)
mi10.11    = as.data.frame((otu_table(sam10.11)))
var10.11   = sample_data(sam10.11)
treat10.11 = var10.11$Condition

library(ALDEx2)

x10.11 <- aldex(mi10.11, treat10.11, mc.samples=128, test="t", effect=TRUE,
            include.sample.summary=TRUE, denom="zero", verbose=TRUE)
```

```
## aldex.clr: generating Monte-Carlo instances and clr values

## operating in serial mode

## removed rows with sums equal to zero

## computing zero removal

## data format is OK

## dirichlet samples complete

## transformation complete

## aldex.ttest: doing t-test

## running tests for each MC instance:

## |------------(25%)----------(50%)----------(75%)----------|

## aldex.effect: calculating effect sizes

## operating in serial mode

## sanity check complete

## rab.all  complete

## rab.win  complete

## rab of samples complete

## within sample difference calculated
```

```
## between group difference calculated

## group summaries calculated

## effect size calculated

## summarizing output

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 4.5800888253908, : provided 231 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 4.5800888253908, : provided 231 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 4.5800888253908, : provided 231 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 4.5800888253908, : provided 231 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 4.5800888253908, : provided 231 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 4.5800888253908, : provided 231 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 4.5800888253908, : provided 231 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 4.5800888253908, : provided 231 variables to replace 1 variables
```

```
#Aldex do not include the "other" taxa
tax10.11 = subset_taxa(tax_table(sam10.11), Genus != "Other")

aldex10.11 = cbind(x10.11, tax10.11)
res10.11=(aldex10.11[(aldex10.11$we.ep<="0.01"),])

res10.11
```

```
##  [1] rab.all            rab.win.10_Flooded rab.win.11_Drained rab.sample.HC37
##  [5] rab.sample.HC38    rab.sample.HC39    rab.sample.HC40    rab.sample.HC41
##  [9] rab.sample.HC42    rab.sample.HC43    rab.sample.HC44    diff.btw
## [13] diff.win           effect             overlap            we.ep
## [17] we.eBH             wi.ep              wi.eBH             Genus
## [21] unique
## <0 linhas> (ou row.names de comprimento 0)
```

Ending for GHC, starting for GLC

```
##GLC##
#subsetting only the two first cycles
set.seed(300)

f1t2  = subset_samples(GLC, Cycle == "1" | Cycle == "2")
f2t3  = subset_samples(GLC, Cycle == "2" | Cycle == "3")
f3t4  = subset_samples(GLC, Cycle == "3" | Cycle == "4")
f4t5  = subset_samples(GLC, Cycle == "4" | Cycle == "5")
f5t6  = subset_samples(GLC, Cycle == "5" | Cycle == "6")
f6t7  = subset_samples(GLC, Cycle == "6" | Cycle == "7")
f7t8  = subset_samples(GLC, Cycle == "7" | Cycle == "8")
f8t9  = subset_samples(GLC, Cycle == "8" | Cycle == "9")
f9t10 = subset_samples(GLC, Cycle == "9" | Cycle == "10")
f10t11= subset_samples(GLC, Cycle == "10" | Cycle == "11")
f1t11 = subset_samples(GLC, Cycle == "1" | Cycle == "11")
```

And then the Aldex:

```
set.seed(300)

sam1.2 = microbiome::aggregate_rare(f1t2, level = "Genus",
                                    detection = 1/100, prevalence = 1/100)
mi1.2    = as.data.frame((otu_table(sam1.2)))
var1.2   = sample_data(sam1.2)
treat1.2 = var1.2$Condition

library(ALDEx2)

x1.2 <- aldex(mi1.2, treat1.2, mc.samples=128, test="t", effect=TRUE,
            include.sample.summary=TRUE, denom="zero", verbose=TRUE)
```

```
## aldex.clr: generating Monte-Carlo instances and clr values

## operating in serial mode

## removed rows with sums equal to zero

## computing zero removal

## data format is OK

## dirichlet samples complete

## transformation complete

## aldex.ttest: doing t-test

## running tests for each MC instance:

## |-----------(25%)----------(50%)----------(75%)----------|
```

```
## aldex.effect: calculating effect sizes

## operating in serial mode

## sanity check complete

## rab.all  complete

## rab.win  complete

## rab of samples complete

## within sample difference calculated

## between group difference calculated

## group summaries calculated

## effect size calculated

## summarizing output

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## -0.0639800337861836, : provided 120 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## -0.0639800337861836, : provided 120 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## -0.0639800337861836, : provided 120 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## -0.0639800337861836, : provided 120 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## -0.0639800337861836, : provided 120 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## -0.0639800337861836, : provided 120 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## -0.0639800337861836, : provided 120 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## -0.0639800337861836, : provided 120 variables to replace 1 variables
```

```
#Aldex do not include the "other" taxa
tax1.2 = subset_taxa(tax_table(sam1.2), Genus != "Other")

aldex1.2 = cbind(x1.2, tax1.2)
res1.2=(aldex1.2[(aldex1.2$we.ep<="0.01"),])

res1.2
```

```
##                          rab.all rab.win.01_Drained rab.win.02_Flooded
## Candidatus Koribacter 7.550267           6.584214           8.703242
## Rhodanobacter         1.644515           4.592191          -3.519706
##                       rab.sample.LC01 rab.sample.LC02 rab.sample.LC03
## Candidatus Koribacter     -0.06398003       -3.880409        1.120902
## Rhodanobacter             -0.06398003       -3.880409        1.120902
##                       rab.sample.LC04 rab.sample.LC05 rab.sample.LC06
## Candidatus Koribacter       -2.571083       -3.208884       -3.846361
## Rhodanobacter               -2.571083       -3.208884       -3.846361
##                       rab.sample.LC07 rab.sample.LC08  diff.btw  diff.win
## Candidatus Koribacter       -3.376917       -3.546057  2.137928 0.8146258
## Rhodanobacter               -3.376917       -3.546057 -8.143331 2.7207653
##                          effect     overlap       we.ep     we.eBH      wi.ep
## Candidatus Koribacter  2.620369 0.000274075 0.004180681 0.06379809 0.02857143
## Rhodanobacter         -3.185360 0.000274075 0.009273206 0.07414027 0.02857143
##                          wi.eBH                 Genus                unique
## Candidatus Koribacter 0.1180278 Candidatus Koribacter Candidatus Koribacter
## Rhodanobacter         0.1180278         Rhodanobacter         Rhodanobacter
```

Cicle 2 to 3.

```
set.seed(300)

sam2.3 = microbiome::aggregate_rare(f2t3, level = "Genus",
                                    detection = 1/100, prevalence = 1/100)
mi2.3    = as.data.frame((otu_table(sam2.3)))
var2.3   = sample_data(sam2.3)
treat2.3 = var2.3$Condition

library(ALDEx2)

x2.3 <- aldex(mi2.3, treat2.3, mc.samples=128, test="t", effect=TRUE,
            include.sample.summary=TRUE, denom="zero", verbose=TRUE)
```

```
## aldex.clr: generating Monte-Carlo instances and clr values

## operating in serial mode

## removed rows with sums equal to zero

## computing zero removal

## data format is OK

## dirichlet samples complete

## transformation complete

## aldex.ttest: doing t-test

## running tests for each MC instance:
```

```
## |------------(25%)----------(50%)----------(75%)----------|

## aldex.effect: calculating effect sizes

## operating in serial mode

## sanity check complete

## rab.all  complete

## rab.win  complete

## rab of samples complete

## within sample difference calculated

## between group difference calculated

## group summaries calculated

## effect size calculated

## summarizing output

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## -3.66889836276878, : provided 228 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## -3.66889836276878, : provided 228 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## -3.66889836276878, : provided 228 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## -3.66889836276878, : provided 228 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## -3.66889836276878, : provided 228 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## -3.66889836276878, : provided 228 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## -3.66889836276878, : provided 228 variables to replace 1 variables
```

```
#Aldex do not include the "other" taxa
tax2.3 = subset_taxa(tax_table(sam2.3), Genus != "Other")

aldex2.3 = cbind(x2.3, tax2.3)
res2.3=(aldex2.3[(aldex2.3$we.ep<="0.01"),])

res2.3
```

```
##                  rab.all rab.win.02_Flooded rab.win.03_Drained rab.sample.LC05
## Ellin6067     -1.1696554          -3.370451           4.699384       -3.668898
## Haliangium     3.2656175           2.200601           6.508236       -3.668898
## Ideonella     -1.1984785          -3.385060           4.665888       -3.668898
## Methylobacter -0.9721832          -3.764250           5.863846       -3.668898
## Opitutus       4.5882417           4.115457           6.080320       -3.668898
## Unknown       10.1223531           9.459928          11.816447       -3.668898
##               rab.sample.LC06 rab.sample.LC07 rab.sample.LC08 rab.sample.LC10
## Ellin6067           -4.399151       -3.390956       -4.313406        3.809914
## Haliangium          -4.399151       -3.390956       -4.313406        3.809914
## Ideonella           -4.399151       -3.390956       -4.313406        3.809914
## Methylobacter       -4.399151       -3.390956       -4.313406        3.809914
## Opitutus            -4.399151       -3.390956       -4.313406        3.809914
## Unknown             -4.399151       -3.390956       -4.313406        3.809914
##               rab.sample.LC11 rab.sample.LC12 diff.btw  diff.win   effect
## Ellin6067            4.444072        3.078939 8.204525 2.6081544 3.436800
## Haliangium           4.444072        3.078939 4.382438 1.0077162 4.104686
## Ideonella            4.444072        3.078939 8.070383 2.4998625 3.418151
## Methylobacter        4.444072        3.078939 9.555023 2.4647529 4.121316
## Opitutus             4.444072        3.078939 1.980401 0.5014754 4.016331
## Unknown              4.444072        3.078939 2.212859 0.6308861 3.584704
##                    overlap       we.ep     we.eBH      wi.ep    wi.eBH
## Ellin6067     0.0003653998 0.008980138 0.06553311 0.05714286 0.1603412
## Haliangium    0.0003653998 0.004075228 0.05206321 0.05714286 0.1603412
## Ideonella     0.0003653998 0.009377002 0.06665011 0.05714286 0.1603412
## Methylobacter 0.0003653998 0.006501909 0.05154554 0.05714286 0.1603412
## Opitutus      0.0003653998 0.002685795 0.04353241 0.05714286 0.1603412
## Unknown       0.0003653998 0.002783153 0.04724061 0.05714286 0.1603412
##                       Genus        unique
## Ellin6067         Ellin6067     Ellin6067
## Haliangium       Haliangium    Haliangium
## Ideonella         Ideonella     Ideonella
## Methylobacter Methylobacter Methylobacter
## Opitutus           Opitutus      Opitutus
## Unknown             Unknown       Unknown
```

Cicle 3-4

```
set.seed(300)

sam3.4 = microbiome::aggregate_rare(f3t4, level = "Genus",
                                    detection = 1/100, prevalence = 1/100)
mi3.4    = as.data.frame((otu_table(sam3.4)))
var3.4   = sample_data(sam3.4)
treat3.4 = var3.4$Condition

library(ALDEx2)

x3.4 <- aldex(mi3.4, treat3.4, mc.samples=128, test="t", effect=TRUE,
            include.sample.summary=TRUE, denom="zero", verbose=TRUE)
```

```
## aldex.clr: generating Monte-Carlo instances and clr values

## operating in serial mode
```

```
## removed rows with sums equal to zero

## computing zero removal

## data format is OK

## dirichlet samples complete

## transformation complete

## aldex.ttest: doing t-test

## running tests for each MC instance:

## |------------(25%)----------(50%)----------(75%)----------|

## aldex.effect: calculating effect sizes

## operating in serial mode

## sanity check complete

## rab.all  complete

## rab.win  complete

## rab of samples complete

## within sample difference calculated

## between group difference calculated

## group summaries calculated

## effect size calculated

## summarizing output

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## -4.44341933580514, : provided 259 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## -4.44341933580514, : provided 259 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## -4.44341933580514, : provided 259 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## -4.44341933580514, : provided 259 variables to replace 1 variables
```

```
## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## -4.44341933580514, : provided 259 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## -4.44341933580514, : provided 259 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## -4.44341933580514, : provided 259 variables to replace 1 variables
```

```
#Aldex do not include the "other" taxa
tax3.4 = subset_taxa(tax_table(sam3.4), Genus != "Other")

aldex3.4 = cbind(x3.4, tax3.4)
res3.4=(aldex3.4[(aldex3.4$we.ep<="0.01"),])

res3.4
```

```
##  [1] rab.all            rab.win.03_Drained rab.win.04_Flooded rab.sample.LC10
##  [5] rab.sample.LC11    rab.sample.LC12    rab.sample.LC13    rab.sample.LC14
##  [9] rab.sample.LC15    rab.sample.LC16    diff.btw           diff.win
## [13] effect             overlap            we.ep              we.eBH
## [17] wi.ep              wi.eBH             Genus              unique
## <0 linhas> (ou row.names de comprimento 0)
```

Cycle 4 to 5.

```
set.seed(300)

sam4.5 = microbiome::aggregate_rare(f4t5, level = "Genus",
                                    detection = 1/100, prevalence = 1/100)
mi4.5    = as.data.frame((otu_table(sam4.5)))
var4.5   = sample_data(sam4.5)
treat4.5 = var4.5$Condition

library(ALDEx2)

x4.5 <- aldex(mi4.5, treat4.5, mc.samples=128, test="t", effect=TRUE,
            include.sample.summary=TRUE, denom="zero", verbose=TRUE)
```

```
## aldex.clr: generating Monte-Carlo instances and clr values

## operating in serial mode

## removed rows with sums equal to zero

## computing zero removal

## data format is OK

## dirichlet samples complete
```

```
## transformation complete

## aldex.ttest: doing t-test

## running tests for each MC instance:

## |------------(25%)----------(50%)----------(75%)----------|

## aldex.effect: calculating effect sizes

## operating in serial mode

## sanity check complete

## rab.all  complete

## rab.win  complete

## rab of samples complete

## within sample difference calculated

## between group difference calculated

## group summaries calculated

## effect size calculated

## summarizing output

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## -5.8415214476007, : provided 253 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## -5.8415214476007, : provided 253 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## -5.8415214476007, : provided 253 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## -5.8415214476007, : provided 253 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## -5.8415214476007, : provided 253 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## -5.8415214476007, : provided 253 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## -5.8415214476007, : provided 253 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## -5.8415214476007, : provided 253 variables to replace 1 variables
```

```
#Aldex do not include the "other" taxa
tax4.5 = subset_taxa(tax_table(sam4.5), Genus != "Other")

aldex4.5 = cbind(x4.5, tax4.5)
res4.5=(aldex4.5[(aldex4.5$we.ep<="0.01"),])

res4.5
```

```
##               rab.all rab.win.04_Flooded rab.win.05_Drained rab.sample.LC13
## Rhodoblastus 1.160886          -4.680077           3.569829       -5.841521
##              rab.sample.LC14 rab.sample.LC15 rab.sample.LC16 rab.sample.LC17
## Rhodoblastus       -4.935718        3.396724       -3.998035       -4.147127
##              rab.sample.LC18 rab.sample.LC19 rab.sample.LC20 diff.btw diff.win
## Rhodoblastus       -4.533095       -4.708103       -5.292815 8.252328 2.738971
##                 effect     overlap       we.ep    we.eBH      wi.ep    wi.eBH
## Rhodoblastus 3.067809 0.000274075 0.009435752 0.4092384 0.02857143 0.4957137
##                       Genus       unique
## Rhodoblastus Rhodoblastus Rhodoblastus
```

Cycle 5 to 6.

```
set.seed(300)

sam5.6 = microbiome::aggregate_rare(f5t6, level = "Genus",
                                    detection = 1/100, prevalence = 1/100)
mi5.6    = as.data.frame((otu_table(sam5.6)))
var5.6   = sample_data(sam5.6)
treat5.6 = var5.6$Condition

library(ALDEx2)

x5.6 <- aldex(mi5.6, treat5.6, mc.samples=128, test="t", effect=TRUE,
            include.sample.summary=TRUE, denom="zero", verbose=TRUE)
```

```
## aldex.clr: generating Monte-Carlo instances and clr values

## operating in serial mode

## removed rows with sums equal to zero

## computing zero removal

## data format is OK

## dirichlet samples complete

## transformation complete

## aldex.ttest: doing t-test

## running tests for each MC instance:
```

```
## |------------(25%)----------(50%)----------(75%)----------|

## aldex.effect: calculating effect sizes

## operating in serial mode

## sanity check complete

## rab.all  complete

## rab.win  complete

## rab of samples complete

## within sample difference calculated

## between group difference calculated

## group summaries calculated

## effect size calculated

## summarizing output

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 2.93951896893852, : provided 244 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 2.93951896893852, : provided 244 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 2.93951896893852, : provided 244 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 2.93951896893852, : provided 244 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 2.93951896893852, : provided 244 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 2.93951896893852, : provided 244 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 2.93951896893852, : provided 244 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 2.93951896893852, : provided 244 variables to replace 1 variables
```

```
#Aldex do not include the "other" taxa
tax5.6 = subset_taxa(tax_table(sam5.6), Genus != "Other")

aldex5.6 = cbind(x5.6, tax5.6)
res5.6=(aldex5.6[(aldex5.6$we.ep<="0.01"),])

res5.6
```

```
##           rab.all rab.win.05_Drained rab.win.06_Flooded rab.sample.LC17
## Bacillus 4.229652           3.440415           5.153215        2.939519
##          rab.sample.LC18 rab.sample.LC19 rab.sample.LC20 rab.sample.LC21
## Bacillus         3.10047        2.657655        2.312045        3.669177
##          rab.sample.LC22 rab.sample.LC23 rab.sample.LC24 diff.btw  diff.win
## Bacillus        3.788295        3.169628        4.235792 1.614406 0.6468765
##            effect     overlap       we.ep    we.eBH      wi.ep   wi.eBH
## Bacillus 2.503439 0.003930039 0.004606506 0.6058072 0.02991071 0.804849
##             Genus   unique
## Bacillus Bacillus Bacillus
```

Cycle 6 to 7.

```
set.seed(300)

sam6.7 = microbiome::aggregate_rare(f6t7, level = "Genus",
                                    detection = 1/100, prevalence = 1/100)
mi6.7    = as.data.frame((otu_table(sam6.7)))
var6.7   = sample_data(sam6.7)
treat6.7 = var6.7$Condition

library(ALDEx2)

x6.7 <- aldex(mi6.7, treat6.7, mc.samples=128, test="t", effect=TRUE,
            include.sample.summary=TRUE, denom="zero", verbose=TRUE)
```

```
## aldex.clr: generating Monte-Carlo instances and clr values

## operating in serial mode

## removed rows with sums equal to zero

## computing zero removal

## data format is OK

## dirichlet samples complete

## transformation complete

## aldex.ttest: doing t-test

## running tests for each MC instance:
```

```
## |------------(25%)----------(50%)----------(75%)----------|

## aldex.effect: calculating effect sizes

## operating in serial mode

## sanity check complete

## rab.all  complete

## rab.win  complete

## rab of samples complete

## within sample difference calculated

## between group difference calculated

## group summaries calculated

## effect size calculated

## summarizing output

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 3.70349972608717, : provided 250 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 3.70349972608717, : provided 250 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 3.70349972608717, : provided 250 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 3.70349972608717, : provided 250 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 3.70349972608717, : provided 250 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 3.70349972608717, : provided 250 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 3.70349972608717, : provided 250 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 3.70349972608717, : provided 250 variables to replace 1 variables
```

```
#Aldex do not include the "other" taxa
tax6.7 = subset_taxa(tax_table(sam6.7), Genus != "Other")

aldex6.7 = cbind(x6.7, tax6.7)
res6.7=(aldex6.7[(aldex6.7$we.ep<="0.01"),])

res6.7
```

```
##               rab.all rab.win.06_Flooded rab.win.07_Drained rab.sample.LC21
## Acidothermus 6.084203           6.696665           5.567875          3.7035
## Bacillus     4.015051           5.149000           3.112530          3.7035
## Conexibacter 5.375853           6.149171           4.711153          3.7035
## Opitutus     6.202477           6.714604           5.553549          3.7035
## Ramlibacter  6.701499           7.952059           5.736723          3.7035
##              rab.sample.LC22 rab.sample.LC23 rab.sample.LC24 rab.sample.LC25
## Acidothermus        3.901149        3.189594        4.216103        2.343397
## Bacillus            3.901149        3.189594        4.216103        2.343397
## Conexibacter        3.901149        3.189594        4.216103        2.343397
## Opitutus            3.901149        3.189594        4.216103        2.343397
## Ramlibacter         3.901149        3.189594        4.216103        2.343397
##              rab.sample.LC26 rab.sample.LC27 rab.sample.LC28  diff.btw
## Acidothermus        2.615789        3.354569        3.140695 -1.184566
## Bacillus            2.615789        3.354569        3.140695 -1.995761
## Conexibacter        2.615789        3.354569        3.140695 -1.390884
## Opitutus            2.615789        3.354569        3.140695 -1.207609
## Ramlibacter         2.615789        3.354569        3.140695 -2.026543
##                diff.win    effect     overlap       we.ep    we.eBH      wi.ep
## Acidothermus 0.4915577 -2.421597 0.000274075 0.007590947 0.2107247 0.02857143
## Bacillus     0.5662497 -3.478134 0.000274075 0.001581563 0.1165760 0.02857143
## Conexibacter 0.4742558 -2.820823 0.000274075 0.002286936 0.1321062 0.02857143
## Opitutus     0.3270857 -3.713558 0.000274075 0.001457270 0.1015406 0.02857143
## Ramlibacter  0.8908955 -2.193381 0.000274075 0.005231757 0.1971856 0.02857143
##                 wi.eBH        Genus       unique
## Acidothermus 0.2168822 Acidothermus Acidothermus
## Bacillus     0.2168822     Bacillus     Bacillus
## Conexibacter 0.2168822 Conexibacter Conexibacter
## Opitutus     0.2168822     Opitutus     Opitutus
## Ramlibacter  0.2168822  Ramlibacter  Ramlibacter
```

Cycle 7 to 8.

```
set.seed(300)

sam7.8 = microbiome::aggregate_rare(f7t8, level = "Genus",
                                    detection = 1/100, prevalence = 1/100)
mi7.8    = as.data.frame((otu_table(sam7.8)))
var7.8   = sample_data(sam7.8)
treat7.8 = var7.8$Condition

library(ALDEx2)

x7.8 <- aldex(mi7.8, treat7.8, mc.samples=128, test="t", effect=TRUE,
            include.sample.summary=TRUE, denom="zero", verbose=TRUE)
```

```
## aldex.clr: generating Monte-Carlo instances and clr values

## operating in serial mode

## removed rows with sums equal to zero

## computing zero removal

## data format is OK

## dirichlet samples complete

## transformation complete

## aldex.ttest: doing t-test

## running tests for each MC instance:

## |------------(25%)----------(50%)----------(75%)----------|

## aldex.effect: calculating effect sizes

## operating in serial mode

## sanity check complete

## rab.all  complete

## rab.win  complete

## rab of samples complete

## within sample difference calculated

## between group difference calculated

## group summaries calculated

## effect size calculated

## summarizing output

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 2.38213490356051, : provided 239 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 2.38213490356051, : provided 239 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
```

```
## 2.38213490356051, : provided 239 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 2.38213490356051, : provided 239 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 2.38213490356051, : provided 239 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 2.38213490356051, : provided 239 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 2.38213490356051, : provided 239 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 2.38213490356051, : provided 239 variables to replace 1 variables
```

```
#Aldex do not include the "other" taxa
tax7.8 = subset_taxa(tax_table(sam7.8), Genus != "Other")

aldex7.8 = cbind(x7.8, tax7.8)
res7.8=(aldex7.8[(aldex7.8$we.ep<="0.01"),])

res7.8
```

```
##            rab.all rab.win.07_Drained rab.win.08_Flooded rab.sample.LC25
## Ralstonia 4.753692           5.525997           3.791398        2.382135
##           rab.sample.LC26 rab.sample.LC27 rab.sample.LC28 rab.sample.LC29
## Ralstonia        2.600877        3.376104         3.17162        2.345176
##           rab.sample.LC30 rab.sample.LC31 rab.sample.LC32  diff.btw  diff.win
## Ralstonia         3.86913        2.685949        2.713397 -1.770479 0.5880871
##              effect     overlap       we.ep  we.eBH      wi.ep    wi.eBH
## Ralstonia -3.200843 0.000274075 0.006763382 0.55633 0.02857143 0.5488872
##               Genus    unique
## Ralstonia Ralstonia Ralstonia
```

Cycle 8 to 9.

```
set.seed(300)

sam8.9 = microbiome::aggregate_rare(f8t9, level = "Genus",
                                    detection = 1/100, prevalence = 1/100)
mi8.9    = as.data.frame((otu_table(sam8.9)))
var8.9   = sample_data(sam8.9)
treat8.9 = var8.9$Condition

library(ALDEx2)

x8.9 <- aldex(mi8.9, treat8.9, mc.samples=128, test="t", effect=TRUE,
            include.sample.summary=TRUE, denom="zero", verbose=TRUE)
```

```
## aldex.clr: generating Monte-Carlo instances and clr values
```

```
## operating in serial mode

## removed rows with sums equal to zero

## computing zero removal

## data format is OK

## dirichlet samples complete

## transformation complete

## aldex.ttest: doing t-test

## running tests for each MC instance:

## |------------(25%)----------(50%)----------(75%)----------|

## aldex.effect: calculating effect sizes

## operating in serial mode

## sanity check complete

## rab.all  complete

## rab.win  complete

## rab of samples complete

## within sample difference calculated

## between group difference calculated

## group summaries calculated

## effect size calculated

## summarizing output

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 2.33482730841958, : provided 224 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 2.33482730841958, : provided 224 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 2.33482730841958, : provided 224 variables to replace 1 variables
```

```
## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 2.33482730841958, : provided 224 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 2.33482730841958, : provided 224 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 2.33482730841958, : provided 224 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 2.33482730841958, : provided 224 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 2.33482730841958, : provided 224 variables to replace 1 variables
```

```
#Aldex do not include the "other" taxa
tax8.9 = subset_taxa(tax_table(sam8.9), Genus != "Other")

aldex8.9 = cbind(x8.9, tax8.9)
res8.9=(aldex8.9[(aldex8.9$we.ep<="0.01"),])

res8.9
```

```
##  [1] rab.all            rab.win.08_Flooded rab.win.09_Drained rab.sample.LC29
##  [5] rab.sample.LC30    rab.sample.LC31    rab.sample.LC32    rab.sample.LC33
##  [9] rab.sample.LC34    rab.sample.LC35    rab.sample.LC36    diff.btw
## [13] diff.win           effect             overlap            we.ep
## [17] we.eBH             wi.ep              wi.eBH             Genus
## [21] unique
## <0 linhas> (ou row.names de comprimento 0)
```

Cycle 9 to 10.

```
set.seed(300)

sam9.10 = microbiome::aggregate_rare(f9t10, level = "Genus",
                                     detection = 1/100, prevalence = 1/100)
mi9.10    = as.data.frame((otu_table(sam9.10)))
var9.10   = sample_data(sam9.10)
treat9.10 = var9.10$Condition

library(ALDEx2)

x9.10 <- aldex(mi9.10, treat9.10, mc.samples=128, test="t", effect=TRUE,
            include.sample.summary=TRUE, denom="zero", verbose=TRUE)
```

```
## aldex.clr: generating Monte-Carlo instances and clr values

## operating in serial mode

## removed rows with sums equal to zero

## computing zero removal
```

```
## data format is OK

## dirichlet samples complete

## transformation complete

## aldex.ttest: doing t-test

## running tests for each MC instance:

## |------------(25%)----------(50%)----------(75%)----------|

## aldex.effect: calculating effect sizes

## operating in serial mode

## sanity check complete

## rab.all  complete

## rab.win  complete

## rab of samples complete

## within sample difference calculated

## between group difference calculated

## group summaries calculated

## effect size calculated

## summarizing output

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 3.44039997656654, : provided 221 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 3.44039997656654, : provided 221 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 3.44039997656654, : provided 221 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 3.44039997656654, : provided 221 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 3.44039997656654, : provided 221 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
```

```
## 3.44039997656654, : provided 221 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 3.44039997656654, : provided 221 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 3.44039997656654, : provided 221 variables to replace 1 variables
```

```
#Aldex do not include the "other" taxa
tax9.10 = subset_taxa(tax_table(sam9.10), Genus != "Other")

aldex9.10 = cbind(x9.10, tax9.10)
res9.10=(aldex9.10[(aldex9.10$we.ep<="0.01"),])

res9.10
```

```
##  [1] rab.all            rab.win.09_Drained rab.win.10_Flooded rab.sample.LC33
##  [5] rab.sample.LC34    rab.sample.LC35    rab.sample.LC36    rab.sample.LC37
##  [9] rab.sample.LC38    rab.sample.LC39    rab.sample.LC40    diff.btw
## [13] diff.win           effect             overlap            we.ep
## [17] we.eBH             wi.ep              wi.eBH             Genus
## [21] unique
## <0 linhas> (ou row.names de comprimento 0)
```

Cycle 10 to 11.

```
set.seed(300)

sam10.11 = microbiome::aggregate_rare(f10t11, level = "Genus",
                                      detection = 1/100, prevalence = 1/100)
mi10.11    = as.data.frame((otu_table(sam10.11)))
var10.11   = sample_data(sam10.11)
treat10.11 = var10.11$Condition

library(ALDEx2)

x10.11 <- aldex(mi10.11, treat10.11, mc.samples=128, test="t", effect=TRUE,
            include.sample.summary=TRUE, denom="zero", verbose=TRUE)
```

```
## aldex.clr: generating Monte-Carlo instances and clr values

## operating in serial mode

## removed rows with sums equal to zero

## computing zero removal

## data format is OK

## dirichlet samples complete

## transformation complete
```

```
## aldex.ttest: doing t-test

## running tests for each MC instance:

## |------------(25%)----------(50%)----------(75%)----------|

## aldex.effect: calculating effect sizes

## operating in serial mode

## sanity check complete

## rab.all  complete

## rab.win  complete

## rab of samples complete

## within sample difference calculated

## between group difference calculated

## group summaries calculated

## effect size calculated

## summarizing output

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 2.62801315681495, : provided 225 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 2.62801315681495, : provided 225 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 2.62801315681495, : provided 225 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 2.62801315681495, : provided 225 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 2.62801315681495, : provided 225 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 2.62801315681495, : provided 225 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 2.62801315681495, : provided 225 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 2.62801315681495, : provided 225 variables to replace 1 variables
```

```
#Aldex do not include the "other" taxa
tax10.11 = subset_taxa(tax_table(sam10.11), Genus != "Other")

aldex10.11 = cbind(x10.11, tax10.11)
res10.11=(aldex10.11[(aldex10.11$we.ep<="0.01"),])

res10.11
```

```
##  [1] rab.all            rab.win.10_Flooded rab.win.11_Drained rab.sample.LC37
##  [5] rab.sample.LC38    rab.sample.LC39    rab.sample.LC40    rab.sample.LC41
##  [9] rab.sample.LC42    rab.sample.LC43    rab.sample.LC44    diff.btw
## [13] diff.win           effect             overlap            we.ep
## [17] we.eBH             wi.ep              wi.eBH             Genus
## [21] unique
## <0 linhas> (ou row.names de comprimento 0)
```

#Thirteenth step: what really change from the first to the last cycle?

```
#GLC initial to final

set.seed(300)

f1t11.GLC = subset_samples(GLC, Cycle == "1" | Cycle == "11")

sam1.11 = microbiome::aggregate_rare(f1t11.GLC, level = "Genus",
                                     detection = 1/100, prevalence = 1/100)
mi1.11    = as.data.frame((otu_table(sam1.11)))
var1.11   = sample_data(sam1.11)
treat1.11 = var1.11$Condition

library(ALDEx2)

x1.11 <- aldex(mi1.11, treat1.11, mc.samples=128, test="t", effect=TRUE,
            include.sample.summary=TRUE, denom="zero", verbose=TRUE)
```

```
## aldex.clr: generating Monte-Carlo instances and clr values

## operating in serial mode

## removed rows with sums equal to zero

## computing zero removal

## data format is OK

## dirichlet samples complete

## transformation complete

## aldex.ttest: doing t-test
```

```
## running tests for each MC instance:

## |------------(25%)----------(50%)----------(75%)----------|

## aldex.effect: calculating effect sizes

## operating in serial mode

## sanity check complete

## rab.all  complete

## rab.win  complete

## rab of samples complete

## within sample difference calculated

## between group difference calculated

## group summaries calculated

## effect size calculated

## summarizing output

## Warning in ‘[<-.data.frame‘(‘*tmp*‘, , nm, value = structure(list(X1 =
## -0.151402929038563, : provided 201 variables to replace 1 variables

## Warning in ‘[<-.data.frame‘(‘*tmp*‘, , nm, value = structure(list(X1 =
## -0.151402929038563, : provided 201 variables to replace 1 variables

## Warning in ‘[<-.data.frame‘(‘*tmp*‘, , nm, value = structure(list(X1 =
## -0.151402929038563, : provided 201 variables to replace 1 variables

## Warning in ‘[<-.data.frame‘(‘*tmp*‘, , nm, value = structure(list(X1 =
## -0.151402929038563, : provided 201 variables to replace 1 variables

## Warning in ‘[<-.data.frame‘(‘*tmp*‘, , nm, value = structure(list(X1 =
## -0.151402929038563, : provided 201 variables to replace 1 variables

## Warning in ‘[<-.data.frame‘(‘*tmp*‘, , nm, value = structure(list(X1 =
## -0.151402929038563, : provided 201 variables to replace 1 variables

## Warning in ‘[<-.data.frame‘(‘*tmp*‘, , nm, value = structure(list(X1 =
## -0.151402929038563, : provided 201 variables to replace 1 variables

## Warning in ‘[<-.data.frame‘(‘*tmp*‘, , nm, value = structure(list(X1 =
## -0.151402929038563, : provided 201 variables to replace 1 variables
```

```r
#Aldex do not include the "other" taxa
tax1.11 = subset_taxa(tax_table(sam1.11), Genus != "Other")

aldex1.11 = cbind(x1.11, tax1.11)
res1.11=(aldex1.11[(aldex1.11$we.ep<="0.01"),])

res1.11
```

```
##                          rab.all rab.win.01_Drained rab.win.11_Drained
## Anaeromyxobacter       3.3706242          -4.139762           6.582919
## Candidatus Koribacter  7.2331525           6.576804           7.925045
## Candidatus Nitrotoga   2.0674048          -4.033265           4.115609
## Dyella                 0.8959466           4.251266          -4.332194
## Ellin6067              5.4253549           4.201373           6.020874
## MND1                   4.4084356           2.023789           5.572776
## Nitrospira             3.5045521          -4.289763           6.016944
## RB41                   3.7356739          -3.989579           6.069373
## Terrimonas             2.1508665          -4.232290           4.284860
## Unknown               11.4214842           9.986499          12.187564
## UTBCD1                 2.1979690          -3.913896           3.918256
##                       rab.sample.LC01 rab.sample.LC02 rab.sample.LC03
## Anaeromyxobacter           -0.1514029        -3.84852       0.9749092
## Candidatus Koribacter      -0.1514029        -3.84852       0.9749092
## Candidatus Nitrotoga       -0.1514029        -3.84852       0.9749092
## Dyella                     -0.1514029        -3.84852       0.9749092
## Ellin6067                  -0.1514029        -3.84852       0.9749092
## MND1                       -0.1514029        -3.84852       0.9749092
## Nitrospira                 -0.1514029        -3.84852       0.9749092
## RB41                       -0.1514029        -3.84852       0.9749092
## Terrimonas                 -0.1514029        -3.84852       0.9749092
## Unknown                    -0.1514029        -3.84852       0.9749092
## UTBCD1                     -0.1514029        -3.84852       0.9749092
##                       rab.sample.LC04 rab.sample.LC41 rab.sample.LC42
## Anaeromyxobacter             -3.14767        5.522395        4.289155
## Candidatus Koribacter        -3.14767        5.522395        4.289155
## Candidatus Nitrotoga         -3.14767        5.522395        4.289155
## Dyella                       -3.14767        5.522395        4.289155
## Ellin6067                    -3.14767        5.522395        4.289155
## MND1                         -3.14767        5.522395        4.289155
## Nitrospira                   -3.14767        5.522395        4.289155
## RB41                         -3.14767        5.522395        4.289155
## Terrimonas                   -3.14767        5.522395        4.289155
## Unknown                      -3.14767        5.522395        4.289155
## UTBCD1                       -3.14767        5.522395        4.289155
##                       rab.sample.LC43 rab.sample.LC44  diff.btw  diff.win
## Anaeromyxobacter              4.39132        4.515847 10.832352 2.5296027
## Candidatus Koribacter         4.39132        4.515847  1.334372 0.5693898
## Candidatus Nitrotoga          4.39132        4.515847  8.519955 2.9916327
## Dyella                        4.39132        4.515847 -8.489208 2.5707768
## Ellin6067                     4.39132        4.515847  1.882062 0.6303215
## MND1                          4.39132        4.515847  3.562802 1.1388374
## Nitrospira                    4.39132        4.515847 10.285681 2.5566821
## RB41                          4.39132        4.515847 10.070656 2.8218379
## Terrimonas                    4.39132        4.515847  8.546742 2.6554588
```

```
## Unknown                      4.39132        4.515847  2.186072 0.4195070
## UTBCD1                       4.39132        4.515847  8.111985 2.5883625
##                            effect     overlap       we.ep     we.eBH      wi.ep
## Anaeromyxobacter       4.350206 0.000274075 0.006331362 0.06114686 0.02857143
## Candidatus Koribacter  2.316080 0.000274075 0.006310574 0.06437003 0.02946429
## Candidatus Nitrotoga   2.719956 0.000274075 0.009983951 0.07058232 0.02857143
## Dyella                -3.476086 0.000274075 0.007170338 0.05983113 0.02857143
## Ellin6067              3.134430 0.000274075 0.002216702 0.04178497 0.02857143
## MND1                   2.991269 0.000274075 0.005240419 0.05962682 0.02857143
## Nitrospira             4.363759 0.000274075 0.006153644 0.06012301 0.02857143
## RB41                   3.638086 0.000274075 0.007852136 0.06928518 0.02857143
## Terrimonas             3.297322 0.000274075 0.009698794 0.07581963 0.02857143
## Unknown                4.837337 0.000274075 0.002293161 0.03949273 0.02857143
## UTBCD1                 3.159151 0.000274075 0.009204569 0.06979793 0.02857143
##                          wi.eBH                 Genus                unique
## Anaeromyxobacter      0.1153040      Anaeromyxobacter      Anaeromyxobacter
## Candidatus Koribacter 0.1179211 Candidatus Koribacter Candidatus Koribacter
## Candidatus Nitrotoga  0.1153040  Candidatus Nitrotoga  Candidatus Nitrotoga
## Dyella                0.1153040               Dyella               Dyella
## Ellin6067             0.1153040            Ellin6067            Ellin6067
## MND1                  0.1153040                 MND1                 MND1
## Nitrospira            0.1153040            Nitrospira           Nitrospira
## RB41                  0.1153040                 RB41                 RB41
## Terrimonas            0.1153040            Terrimonas           Terrimonas
## Unknown               0.1153040               Unknown              Unknown
## UTBCD1                0.1153040               UTBCD1               UTBCD1
```

```
#GHC initial to final

set.seed(300)

f1t11.GHC = subset_samples(GHC, Cycle == "1" | Cycle == "11")

sam1.11 = microbiome::aggregate_rare(f1t11.GHC, level = "Genus",
                                     detection = 1/100, prevalence = 1/100)
mi1.11    = as.data.frame((otu_table(sam1.11)))
var1.11   = sample_data(sam1.11)
treat1.11 = var1.11$Condition

library(ALDEx2)

x1.11 <- aldex(mi1.11, treat1.11, mc.samples=128, test="t", effect=TRUE,
            include.sample.summary=TRUE, denom="zero", verbose=TRUE)
```

```
## aldex.clr: generating Monte-Carlo instances and clr values

## operating in serial mode

## removed rows with sums equal to zero

## computing zero removal

## data format is OK
```

```
## dirichlet samples complete

## transformation complete

## aldex.ttest: doing t-test

## running tests for each MC instance:

## |------------(25%)----------(50%)----------(75%)----------|

## aldex.effect: calculating effect sizes

## operating in serial mode

## sanity check complete

## rab.all  complete

## rab.win  complete

## rab of samples complete

## within sample difference calculated

## between group difference calculated

## group summaries calculated

## effect size calculated

## summarizing output

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 3.25461589354352, : provided 254 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 3.25461589354352, : provided 254 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 3.25461589354352, : provided 254 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 3.25461589354352, : provided 254 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 3.25461589354352, : provided 254 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 3.25461589354352, : provided 254 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 3.25461589354352, : provided 254 variables to replace 1 variables

## Warning in `[<-.data.frame`(`*tmp*`, , nm, value = structure(list(X1 =
## 3.25461589354352, : provided 254 variables to replace 1 variables
```

```
#Aldex do not include the "other" taxa
tax1.11 = subset_taxa(tax_table(sam1.11), Genus != "Other")

aldex1.11 = cbind(x1.11, tax1.11)
res1.11=(aldex1.11[(aldex1.11$we.ep<="0.01"),])

res1.11
```

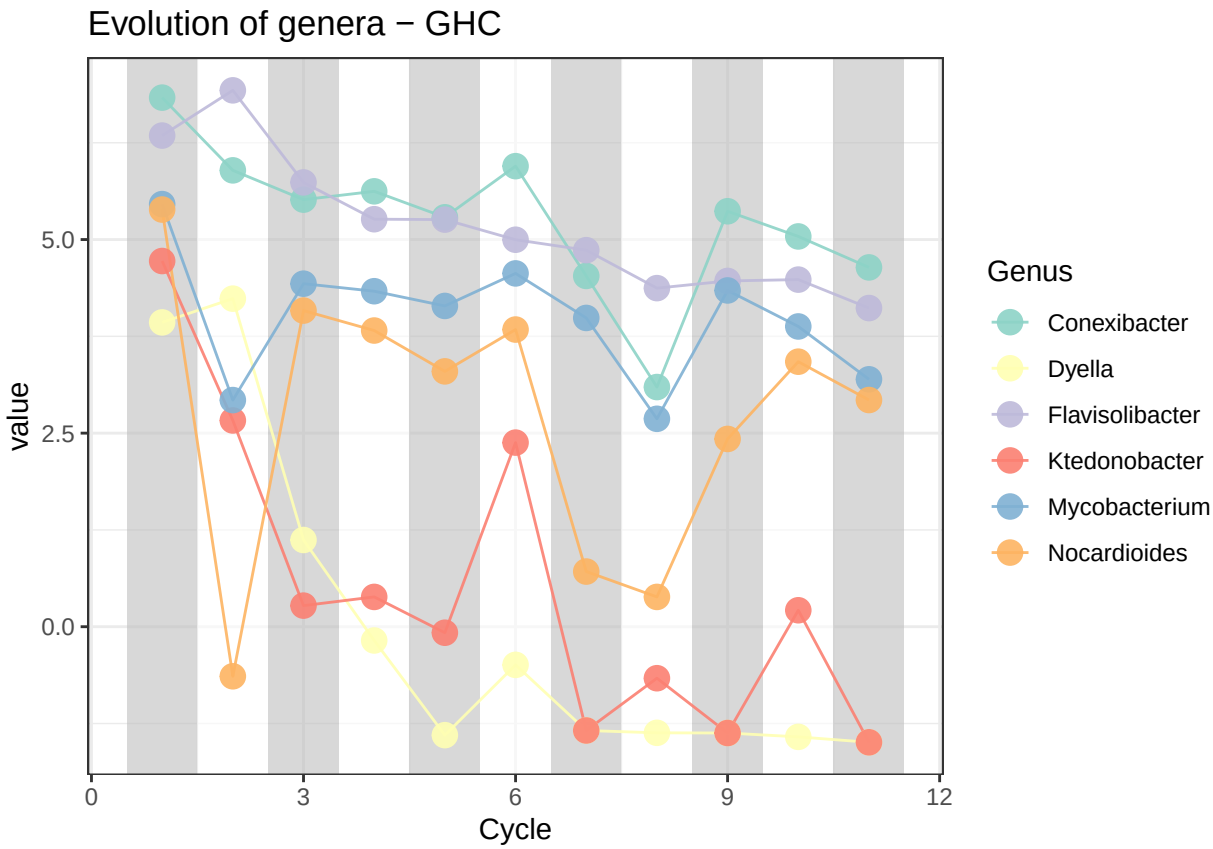
```
##                   rab.all rab.win.01_Drained rab.win.11_Drained rab.sample.HC01
## Conexibacter    5.7225964           7.399466           4.667091        3.254616
## Dyella          0.1974516           2.796207          -5.023409        3.254616
## Flavisolibacter 4.9125817           7.001957           3.917105        3.254616
## Ktedonobacter   1.4003891           4.323354          -5.245659        3.254616
## Mycobacterium   3.9076270           5.138685           2.362680        3.254616
## Nocardioides    3.6025506           5.091558           2.155072        3.254616
##                 rab.sample.HC02 rab.sample.HC03 rab.sample.HC04 rab.sample.HC41
## Conexibacter           4.324798         5.84943        5.080645        3.316055
## Dyella                 4.324798         5.84943        5.080645        3.316055
## Flavisolibacter        4.324798         5.84943        5.080645        3.316055
## Ktedonobacter          4.324798         5.84943        5.080645        3.316055
## Mycobacterium          4.324798         5.84943        5.080645        3.316055
## Nocardioides           4.324798         5.84943        5.080645        3.316055
##                 rab.sample.HC42 rab.sample.HC43 rab.sample.HC44  diff.btw
## Conexibacter           4.514544        3.270977        3.400175 -2.784354
## Dyella                 4.514544        3.270977        3.400175 -8.203015
## Flavisolibacter        4.514544        3.270977        3.400175 -2.997317
## Ktedonobacter          4.514544        3.270977        3.400175 -9.594968
## Mycobacterium          4.514544        3.270977        3.400175 -2.689528
## Nocardioides           4.514544        3.270977        3.400175 -3.121960
##                  diff.win    effect     overlap       we.ep    we.eBH
## Conexibacter    0.8105800 -3.112033 0.000274075 0.005769957 0.1336510
## Dyella          2.8765206 -2.866553 0.000274075 0.006646423 0.1126360
## Flavisolibacter 0.6955350 -3.710096 0.000274075 0.004129519 0.1207167
## Ktedonobacter   2.5416942 -3.864884 0.000274075 0.007146050 0.1154936
## Mycobacterium   1.3180884 -2.193869 0.000274075 0.009473507 0.1511639
## Nocardioides    0.8059399 -3.903902 0.000274075 0.002599179 0.1039479
##                      wi.ep    wi.eBH           Genus          unique
## Conexibacter    0.02857143 0.1595585    Conexibacter    Conexibacter
## Dyella          0.02857143 0.1595585          Dyella          Dyella
## Flavisolibacter 0.02857143 0.1595585 Flavisolibacter Flavisolibacter
## Ktedonobacter   0.02857143 0.1595585   Ktedonobacter   Ktedonobacter
## Mycobacterium   0.02857143 0.1595585   Mycobacterium   Mycobacterium
## Nocardioides    0.02857143 0.1595585    Nocardioides    Nocardioides
```

I saved those anaysis in csv files named 1to11GHC.csv and 1to11GLC.csv. Now I will explore the dinamycs of these taxa over the time.

#Fourteenth step: temporal dynamics of differentially abundand genus

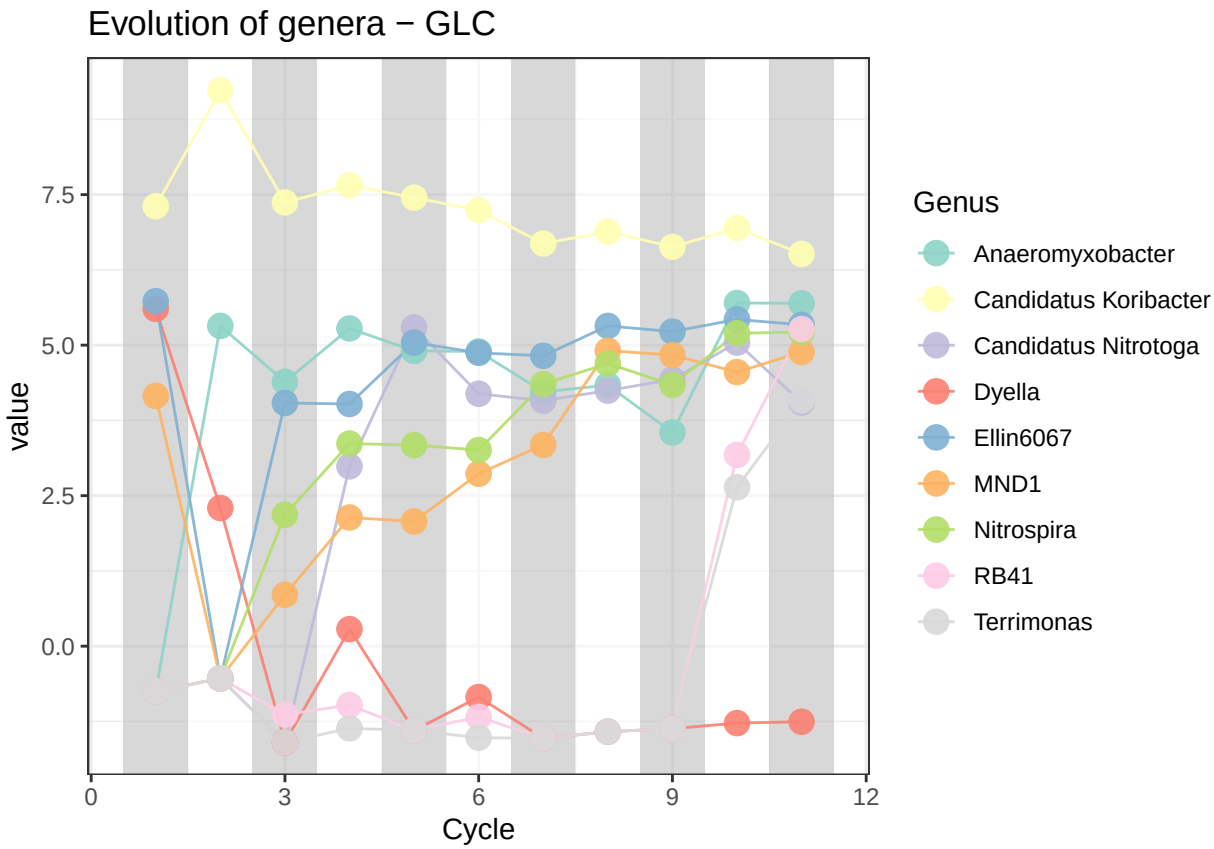
```
#GHC

GHC.ag.clr = microbiome::transform(GHC.ag, "clr")

GHC1.11 = subset_taxa(GHC.ag.clr,
```

```
                      Genus == "Conexibacter" | Genus == "Dyella" | Genus == "Flavisolibacter" | Genus =
                        Genus == "Mycobacterium" | Genus == "Nocardioides")
#GHC1.11#

map.ghc = filter(Map_T, Soil == "GHC")

clr.plot.GHC = cbind(map.ghc, t(otu_table(GHC1.11)))

round_df <- function(x, digits) {
  # round all numeric variables
  # x: data frame
  # digits: number of digits to round
  numeric_columns <- sapply(x, mode) == 'numeric'
  x[numeric_columns] <-  round(x[numeric_columns], digits)
  x
}

clr.plot.GHC = round_df(clr.plot.GHC, 2)

df.test = reshape2::melt(clr.plot.GHC, id.vars = "#NAMES", variable.name = "Genus")

df.test2 = filter(df.test, Genus == "Conexibacter" | Genus == "Dyella" | Genus == "Flavisolibacter" | G
                    Genus == "Mycobacterium" | Genus == "Nocardioides")

df.test3 = filter(df.test, Genus == "Cycle")

df.test4 = cbind(df.test2, df.test3$value)

colnames(df.test4) = c("NAMES", "Genus", "value", "Cycle")


df.test4$value = as.numeric(df.test4$value)
df.test4$Cycle = as.numeric(df.test4$Cycle)



tgc5 <- summarySE(df.test4, measurevar="value", groupvars=c("Genus","Cycle"))

test.plot =
  ggplot() +
  geom_rect(data = rects, aes(xmin = xstart, xmax = xend, ymin = -Inf,
                              ymax = Inf, fill = Condition),
                              fill = rects$Colors, alpha = 0.6) +
  geom_line(data = tgc5, aes(x=Cycle, y=value, color=Genus), alpha = 0.9) +
  geom_point(data = tgc5, aes(x=Cycle, y=value, color=Genus), alpha = 0.9, size=4)+
  ggtitle("Evolution of genera - GHC")

test.plot
```




```
#GLC

GLC.ag.clr = microbiome::transform(GLC.ag, "clr")

GLC1.11 = subset_taxa(GLC.ag.clr, Genus == "Anaeromyxobacter" | Genus == "Candidatus Koribacter" | Genus
                    Genus == "Dyella" | Genus == "Ellin6067" | Genus == "MND1" | Genus == "Nitrospira"
                    Genus == "RB41" | Genus == "Terrimonas")
#GLC1.11#

map.GLC = filter(Map_T, Soil == "GLC")

clr.plot.GLC = cbind(map.GLC, t(otu_table(GLC1.11)))

round_df <- function(x, digits) {
  # round all numeric variables
  # x: data frame
  # digits: number of digits to round
  numeric_columns <- sapply(x, mode) == 'numeric'
  x[numeric_columns] <-  round(x[numeric_columns], digits)
  x
}

clr.plot.GLC = round_df(clr.plot.GLC, 2)

df.test = reshape2::melt(clr.plot.GLC, id.vars = "#NAMES", variable.name = "Genus")
```

```
df.test2 = filter(df.test, Genus == "Anaeromyxobacter" | Genus == "Candidatus Koribacter" | Genus == "Ca
                    Genus == "Dyella" | Genus == "Ellin6067" | Genus == "MND1" | Genus == "Nitrospira"
                    Genus == "RB41" | Genus == "Terrimonas")

df.test3 = filter(df.test, Genus == "Cycle")

df.test4 = cbind(df.test2, df.test3$value)

colnames(df.test4) = c("NAMES", "Genus", "value", "Cycle")


df.test4$value = as.numeric(df.test4$value)
df.test4$Cycle = as.numeric(df.test4$Cycle)



tgc5 <- summarySE(df.test4, measurevar="value", groupvars=c("Genus","Cycle"))

test.plot2 =
  ggplot() +
  geom_rect(data = rects, aes(xmin = xstart, xmax = xend, ymin = -Inf,
                              ymax = Inf, fill = Condition),
                              fill = rects$Colors, alpha = 0.6) +
  geom_line(data = tgc5, aes(x=Cycle, y=value, color=Genus), alpha = 0.9) +
  geom_point(data = tgc5, aes(x=Cycle, y=value, color=Genus), alpha = 0.9, size=4)+
  ggtitle("Evolution of genera - GLC")

test.plot2
```


Evolution of genera − GLC
value
7.5
5.0
2.5
0.0
0
3
6
9
12
Cycle
Genus
Anaeromyxobacter
Candidatus Koribacter
Candidatus Nitrotoga
Dyella
Ellin6067
MND1
Nitrospira
RB41
Terrimonas