Phyloseq

Alejandra

2024-04-18

#load required packages

```
library(phyloseq)
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(BiMiCo)
library(ggplot2)
```

#load taxa and seqtab.nochim

```
load("RData/taxa.RData")
load("RData/seqtab.nochim.RData")
```

#import metadata

```
metadata <- read.csv("sample-metadata.csv", header=TRUE, row.names = 1)
```

#create phylseq object

```
#make sure the seqtab.nochim and taxa objects are loaded
physeq <- phyloseq(otu_table(seqtab.nochim, taxa_are_rows=FALSE), sample_data(metadata),tax_table(taxa))
physeq
```

```
## phyloseq-class experiment-level object
## otu_table()   OTU Table:         [ 771 taxa and 34 samples ]
## sample_data() Sample Data:       [ 34 samples by 9 sample variables ]
## tax_table()   Taxonomy Table:    [ 771 taxa by 7 taxonomic ranks ]
```

#transform sample counts

```
#convert from raw to abundance so its easier to comapre
physeq <- transform_sample_counts(physeq, function(abund) 1*(abund>0))

#visualize to data
physeq
```

```
## phyloseq-class experiment-level object
```

```
## otu_table()   OTU Table:         [ 771 taxa and 34 samples ]
## sample_data() Sample Data:       [ 34 samples by 9 sample variables ]
## tax_table()   Taxonomy Table:    [ 771 taxa by 7 taxonomic ranks ]
```

#remove the sequence itselt and replace with ASV

```
##this allows it to be easier to read, replaces the raw data
dna <- Biostrings::DNAStringSet(taxa_names(physeq))
names(dna) <- taxa_names(physeq)
physeq <- merge_phyloseq(physeq, dna)
taxa_names(physeq) <- paste0("ASV", seq(ntaxa(physeq)))
physeq
```

```
## phyloseq-class experiment-level object
## otu_table()   OTU Table:         [ 771 taxa and 34 samples ]
## sample_data() Sample Data:       [ 34 samples by 9 sample variables ]
## tax_table()   Taxonomy Table:    [ 771 taxa by 7 taxonomic ranks ]
## refseq()      DNAStringSet:      [ 771 reference sequences ]
```

#remove mitochondria and phloroplast mathces, remove all non bacterial sequences

```
#stictly use bacteria 16S rRNA,
physeq <- physeq %>% subset_taxa( Family!= "Mitochondria" | is.na(Family) & Order!="Chloroplast" | is.na
physeq
```

```
## phyloseq-class experiment-level object
## otu_table()   OTU Table:         [ 742 taxa and 34 samples ]
## sample_data() Sample Data:       [ 34 samples by 9 sample variables ]
## tax_table()   Taxonomy Table:    [ 742 taxa by 7 taxonomic ranks ]
## refseq()      DNAStringSet:      [ 742 reference sequences ]
```

#remove all non bacterial sequences

```
physeq<-rm_nonbac(physeq)
physeq
```

```
## phyloseq-class experiment-level object
## otu_table()   OTU Table:         [ 737 taxa and 34 samples ]
## sample_data() Sample Data:       [ 34 samples by 9 sample variables ]
## tax_table()   Taxonomy Table:    [ 737 taxa by 7 taxonomic ranks ]
## refseq()      DNAStringSet:      [ 737 reference sequences ]
```

#save physeq objects to load later

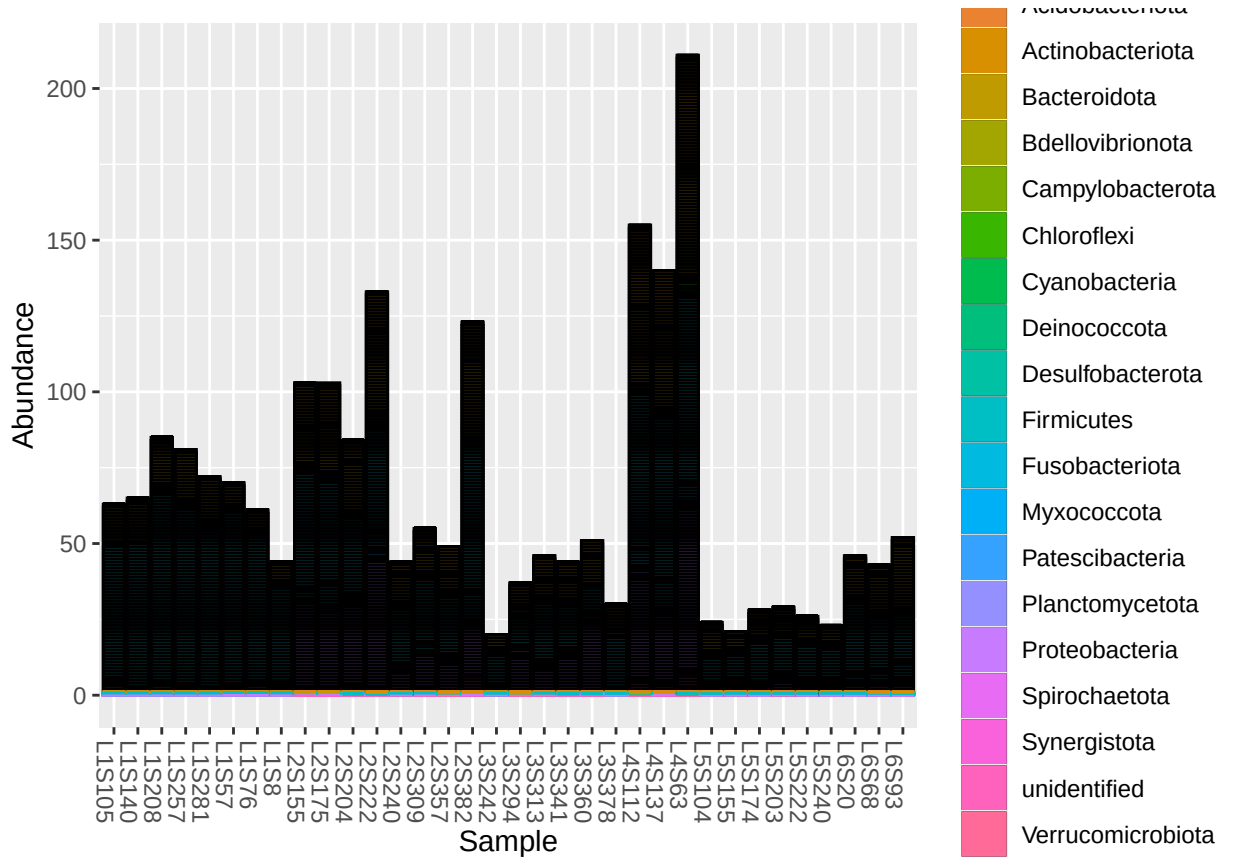
```
save(physeq, file= "RData/physeq.RData")
```

#load physeq objects to start here

```
load("RData/physeq.RData")
```

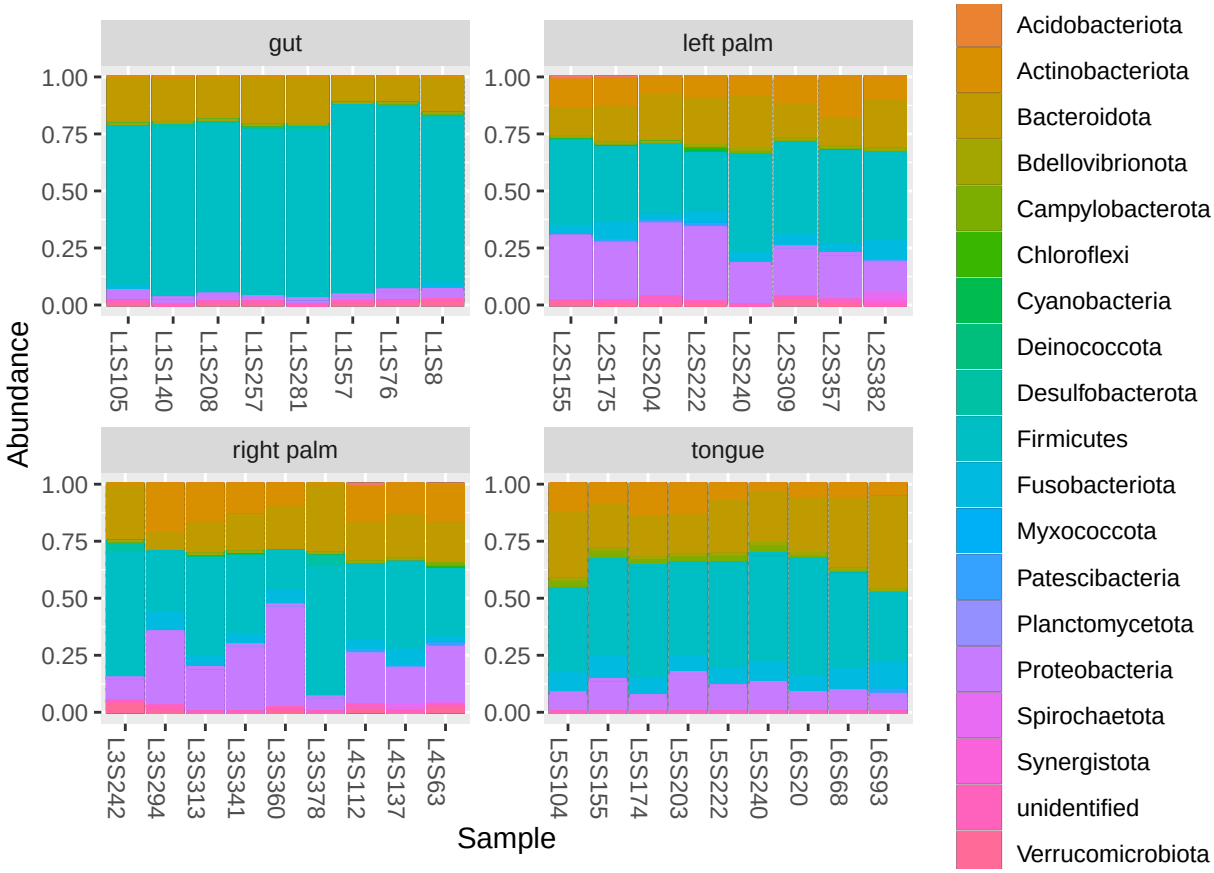
#plot bar grpah based on phylum

```
plot_bar(physeq, fill = "Phylum") + geom_bar(aes(color=Phylum, fill=Phylum), stat="identity", position=
```



#create a barplot of relative abundance

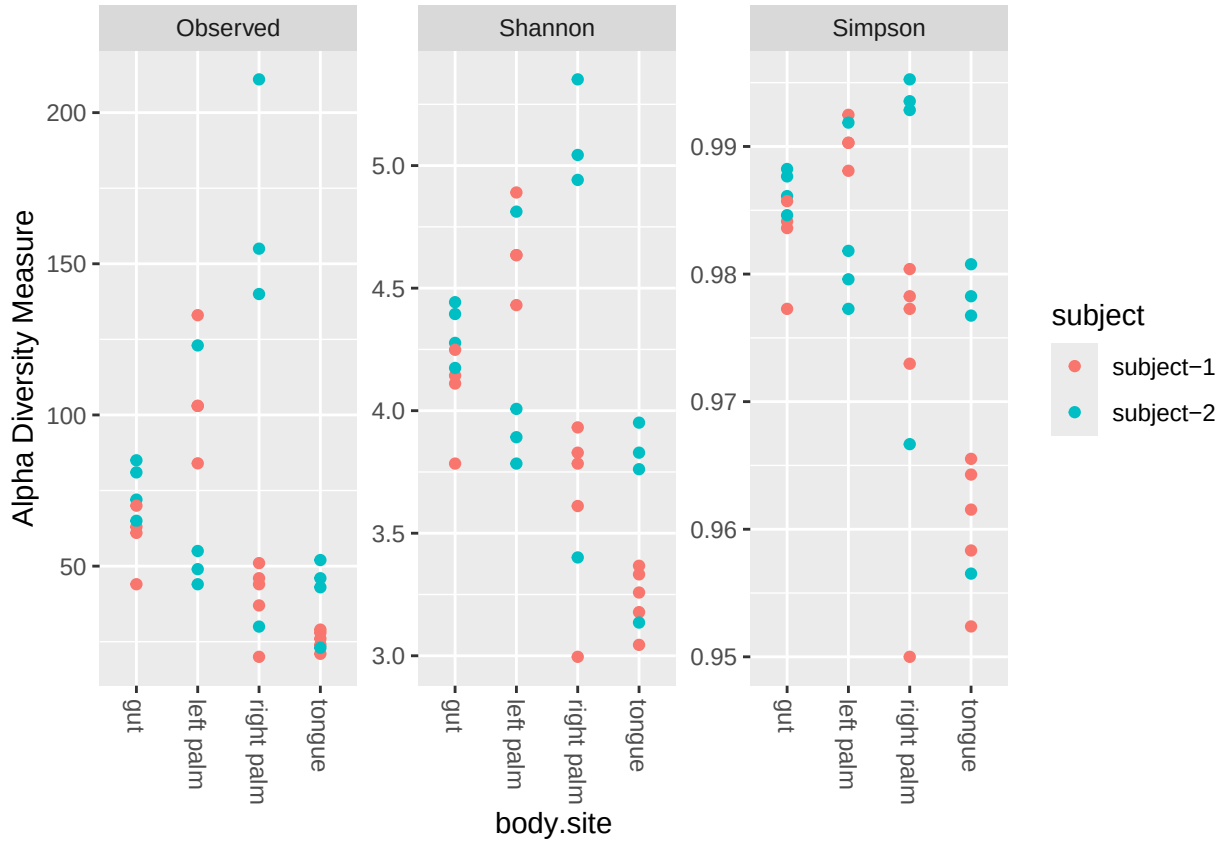
```
#convert to relative abundance
physeq_relabund <- transform_sample_counts(physeq, function(x) x / sum(x))

#barplot
plot_bar(physeq_relabund, fill = "Phylum") + geom_bar(aes(color=Phylum, fill=Phylum), stat="identity", p
```




```
##can change based on the column name in metadata in facet_wrap(~columnName)
```

#plot alpha diversity based on body site

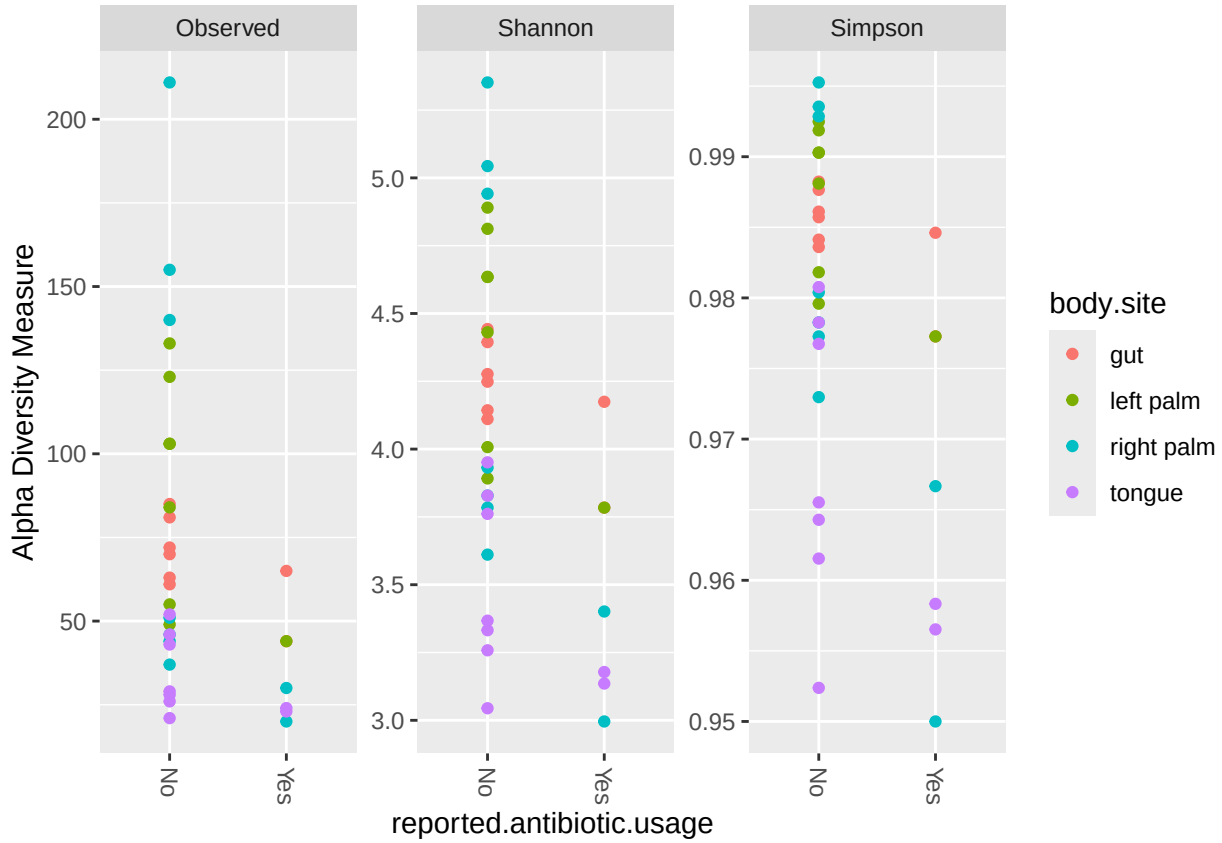
```
plot_richness(physeq, x="body.site", color= "subject", measures=c("Observed", "Simpson", "Shannon"))
```




```
##Simpson(less sensitive, will be more custered together) and Shannon(more sensitive to rare taxa) take
```

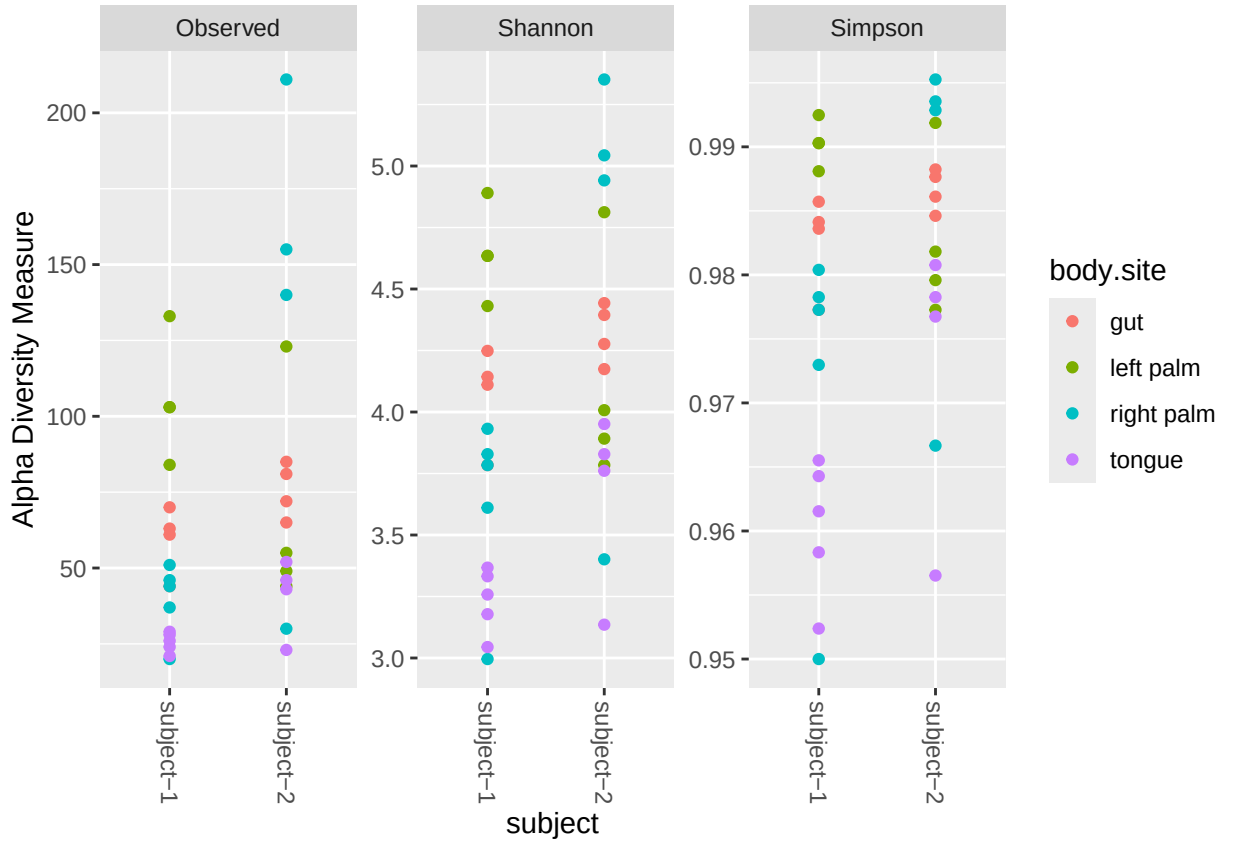
#pplot alpha diversity based on reported.antibiotic.usage

```
plot_richness(physeq, x="reported.antibiotic.usage", color= "body.site", measures=c("Observed", "Simpso
```



#plot alpha diversity based on subject

```
plot_richness(physeq, x="subject", color= "body.site", measures=c("Observed", "Simpson", "Shannon"))
```



#test for normailoty

```
alpha <- estimate_richness(physeq, measures=c("Observed", "Simpson", "Shannon"))

#alternative = the data is normal distributed
#null = the data is not normal distributed

#Shapiro-wilk - used when it has fewer than 50 samples
observed <- shapiro.test(alpha$Observed)
shannon <- shapiro.test(alpha$Shannon)
simpson <- shapiro.test(alpha$Simpson)

#print
print(observed)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  alpha$Observed
## W = 0.85439, p-value = 0.0003513
```

```
print(shannon)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  alpha$Shannon
## W = 0.97517, p-value = 0.617
```

```
print(simpson)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  alpha$Simpson
## W = 0.91749, p-value = 0.01373
```

#create data frames for statistical analyses

```
#extract sample information from the physeq object
samples <- sample_data(physeq)

#if sample is a phyloseq sample_Data, convert it to a data frame
if (class(samples) == "sample_data") {
  samples <- data.frame(sample_data(samples))
  }

#add a column to alpha with sample names
alpha$sample <- rownames(alpha)

#merge alpha diversity data and sample(meta) data
alpha <- merge(alpha, samples, by = "sample")
```

#perform statistics based on subject

```
#perform t/wilcox tests for each biodiversity index
test_observed <- wilcox.test(Observed ~ subject, data = alpha)
```

```
## Warning in wilcox.test.default(x = DATA[[1L]], y = DATA[[2L]], ...): cannot
## compute exact p-value with ties
```

```
test_simpson <- wilcox.test(Simpson ~ subject, data = alpha)
```

```
## Warning in wilcox.test.default(x = DATA[[1L]], y = DATA[[2L]], ...): cannot
## compute exact p-value with ties
```

```
test_shannon <- t.test(Shannon ~ subject, data = alpha)

#print results
print(test_observed)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  Observed by subject
## W = 96.5, p-value = 0.1047
## alternative hypothesis: true location shift is not equal to 0
```

```
print(test_simpson)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  Simpson by subject
## W = 96.5, p-value = 0.1047
## alternative hypothesis: true location shift is not equal to 0
```

```
print(test_shannon)
```

```
##
##  Welch Two Sample t-test
##
## data:  Shannon by subject
## t = -1.7373, df = 31.125, p-value = 0.09223
## alternative hypothesis: true difference in means between group subject-1 and group subject-2 is not
## 95 percent confidence interval:
##  -0.77161834  0.06168674
## sample estimates:
## mean in group subject-1 mean in group subject-2
##                3.844933                4.199899
```

###to change text, select chunk > edit > find >click in selection > replace these tabs with what you ant to change; find - replace > all

#perform statistics based on reported.anitbiotic.usage

```
#perform t/wilcox tests for each biodiversity index
test_observed <- wilcox.test(Observed ~ reported.antibiotic.usage, data = alpha)
```

```
## Warning in wilcox.test.default(x = DATA[[1L]], y = DATA[[2L]], ...): cannot
## compute exact p-value with ties
```

```
test_simpson <- wilcox.test(Simpson ~ reported.antibiotic.usage, data = alpha)
```

```
## Warning in wilcox.test.default(x = DATA[[1L]], y = DATA[[2L]], ...): cannot
## compute exact p-value with ties
```

```
test_shannon <- t.test(Shannon ~ reported.antibiotic.usage, data = alpha)

#print results
print(test_observed)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  Observed by reported.antibiotic.usage
## W = 155, p-value = 0.01057
## alternative hypothesis: true location shift is not equal to 0
```

```
print(test_simpson)
```

```
##
##  Wilcoxon rank sum test with continuity correction
##
## data:  Simpson by reported.antibiotic.usage
## W = 155, p-value = 0.01057
## alternative hypothesis: true location shift is not equal to 0
```

```
print(test_shannon)
```

```
##
##  Welch Two Sample t-test
##
## data:  Shannon by reported.antibiotic.usage
## t = 3.3002, df = 12.383, p-value = 0.006097
```

```
## alternative hypothesis: true difference in means between group No and group Yes is not equal to 0
## 95 percent confidence interval:
##  0.2233954 1.0828476
## sample estimates:
##  mean in group No mean in group Yes
##          4.146442          3.493321
```

#test for body site

```
kruskal.test(Simpson ~ body.site, data=alpha)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  Simpson by body.site
## Kruskal-Wallis chi-squared = 13.435, df = 3, p-value = 0.003785
```

```
pairwise.wilcox.test(alpha$Simpson, alpha$body.site, p.adjust.method = "holm")
```

```
## Warning in wilcox.test.default(xi, xj, paired = paired, ...): cannot compute
## exact p-value with ties

## Warning in wilcox.test.default(xi, xj, paired = paired, ...): cannot compute
## exact p-value with ties

## Warning in wilcox.test.default(xi, xj, paired = paired, ...): cannot compute
## exact p-value with ties

## Warning in wilcox.test.default(xi, xj, paired = paired, ...): cannot compute
## exact p-value with ties

## Warning in wilcox.test.default(xi, xj, paired = paired, ...): cannot compute
## exact p-value with ties
```

```
##
##  Pairwise comparisons using Wilcoxon rank sum test with continuity correction
##
## data:  alpha$Simpson and alpha$body.site
##
##            gut    left palm right palm
## left palm  1.0000 -         -
## right palm 1.0000 1.0000    -
## tongue     0.0020 0.0088    0.2805
##
## P value adjustment method: holm
```

```
kruskal.test(Observed ~ body.site, data=alpha)
```

```
##
##  Kruskal-Wallis rank sum test
##
## data:  Observed by body.site
## Kruskal-Wallis chi-squared = 13.435, df = 3, p-value = 0.003785
```

```
pairwise.wilcox.test(alpha$Observed, alpha$body.site, p.adjust.method = "holm")
```

```
## Warning in wilcox.test.default(xi, xj, paired = paired, ...): cannot compute
## exact p-value with ties
```

```
## Warning in wilcox.test.default(xi, xj, paired = paired, ...): cannot compute
## exact p-value with ties

## Warning in wilcox.test.default(xi, xj, paired = paired, ...): cannot compute
## exact p-value with ties

## Warning in wilcox.test.default(xi, xj, paired = paired, ...): cannot compute
## exact p-value with ties

## Warning in wilcox.test.default(xi, xj, paired = paired, ...): cannot compute
## exact p-value with ties

##
##  Pairwise comparisons using Wilcoxon rank sum test with continuity correction
##
## data:  alpha$Observed and alpha$body.site
##
##            gut    left palm right palm
## left palm  1.0000 -         -
## right palm 1.0000 1.0000    -
## tongue     0.0020 0.0088    0.2805
##
## P value adjustment method: holm
```

```
shannonanova <- aov(Shannon ~ body.site, data=alpha)
summary(shannonanova)
```

```
##             Df Sum Sq Mean Sq F value  Pr(>F)
## body.site    3  4.523  1.5076     5.8 0.00299 **
## Residuals   30  7.797  0.2599
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
TukeyHSD(shannonanova)
```

```
##   Tukey multiple comparisons of means
##     95% family-wise confidence level
##
## Fit: aov(formula = Shannon ~ body.site, data = alpha)
##
## $body.site
##                              diff        lwr         upr     p adj
## left palm-gut         0.18891301 -0.5042094  0.88203538 0.8796872
## right palm-gut       -0.09803036 -0.7716242  0.57556350 0.9786056
## tongue-gut           -0.76832784 -1.4419217 -0.09473398 0.0205658
## right palm-left palm -0.28694338 -0.9605372  0.38665049 0.6570793
## tongue-left palm     -0.95724086 -1.6308347 -0.28364699 0.0029520
## tongue-right palm    -0.67029748 -1.3237795 -0.01681544 0.0427299
```