Mini project

Nakabiri Florence

6/23/2021

Registration No: 2020/HD07/20692U

importing data

```
diab_metadata <- read.csv("diabimmune_16s_t1d_metadata.csv",
              stringsAsFactors = FALSE)
```

Change structure of categorical variables to factors

```
diab_metadata$Gender <- as.factor(diab_metadata$Gender)
diab_metadata$Case_Control <- as.factor(diab_metadata$Case_Control)
diab_metadata$Delivery_Route <- as.factor(diab_metadata$Delivery_Route)
```

```
summary(diab_metadata)
```

```
##   Sample_ID          Subject_ID         Case_Control    Gender
##  Length:777         Length:777         case   :260   female:412
##  Class :character   Class :character   control:517   male  :365
##  Mode  :character   Mode  :character
##
##
##
##   Delivery_Route Age_at_Collection
##  cesarian: 66    Min.   :   6.0
##  vaginal :711    1st Qu.: 229.0
##                  Median : 452.0
##                  Mean   : 482.9
##                  3rd Qu.: 702.0
##                  Max.   :1233.0
```

```
addmargins(xtabs(~diab_metadata$Gender+diab_metadata$Case_Control))
```

```
##                     diab_metadata$Case_Control
## diab_metadata$Gender case control Sum
##               female  142     270 412
##               male    118     247 365
##               Sum     260     517 777
```

```
addmargins(xtabs(~diab_metadata$Delivery_Route+diab_metadata$Case_Control))
```

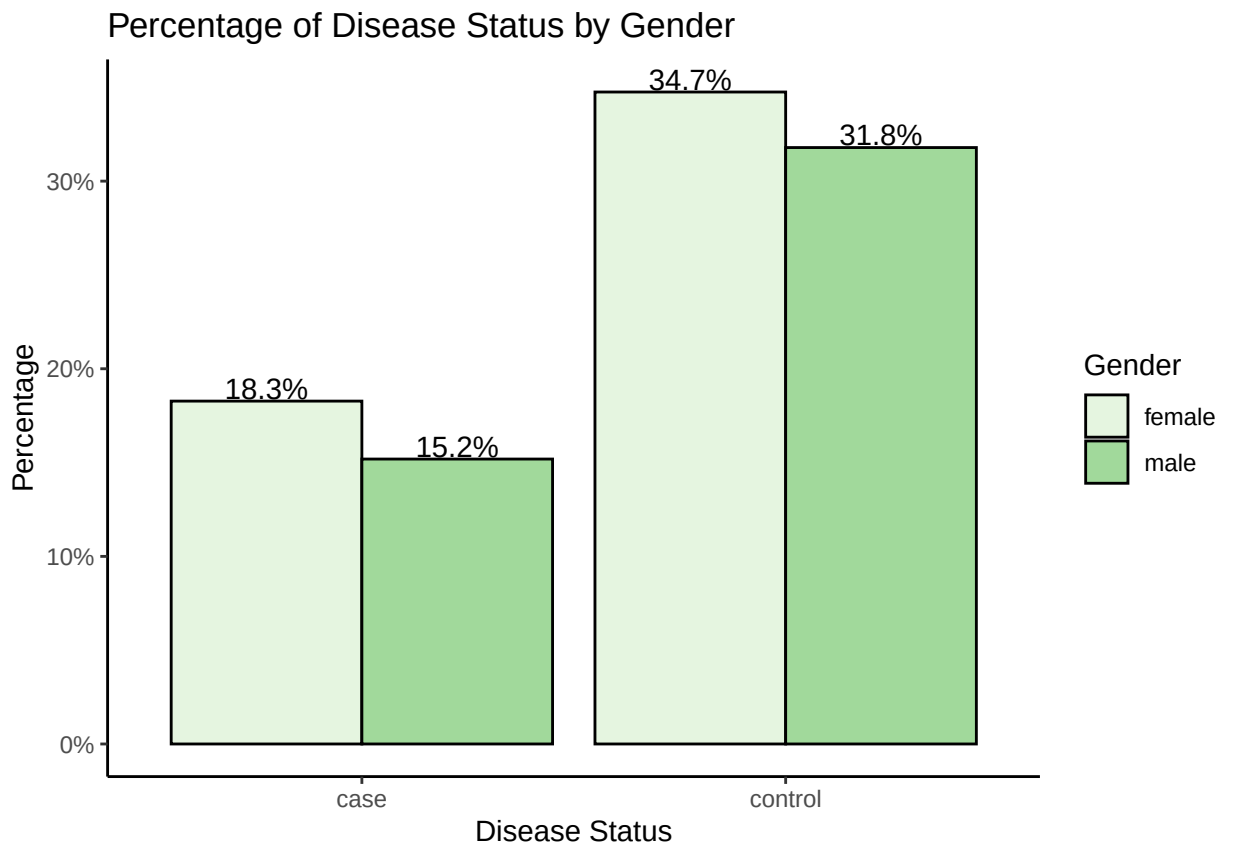
```
##                              diab_metadata$Case_Control
## diab_metadata$Delivery_Route case control Sum
##                     cesarian    0      66  66
##                     vaginal   260     451 711
##                     Sum       260     517 777
```

box plot

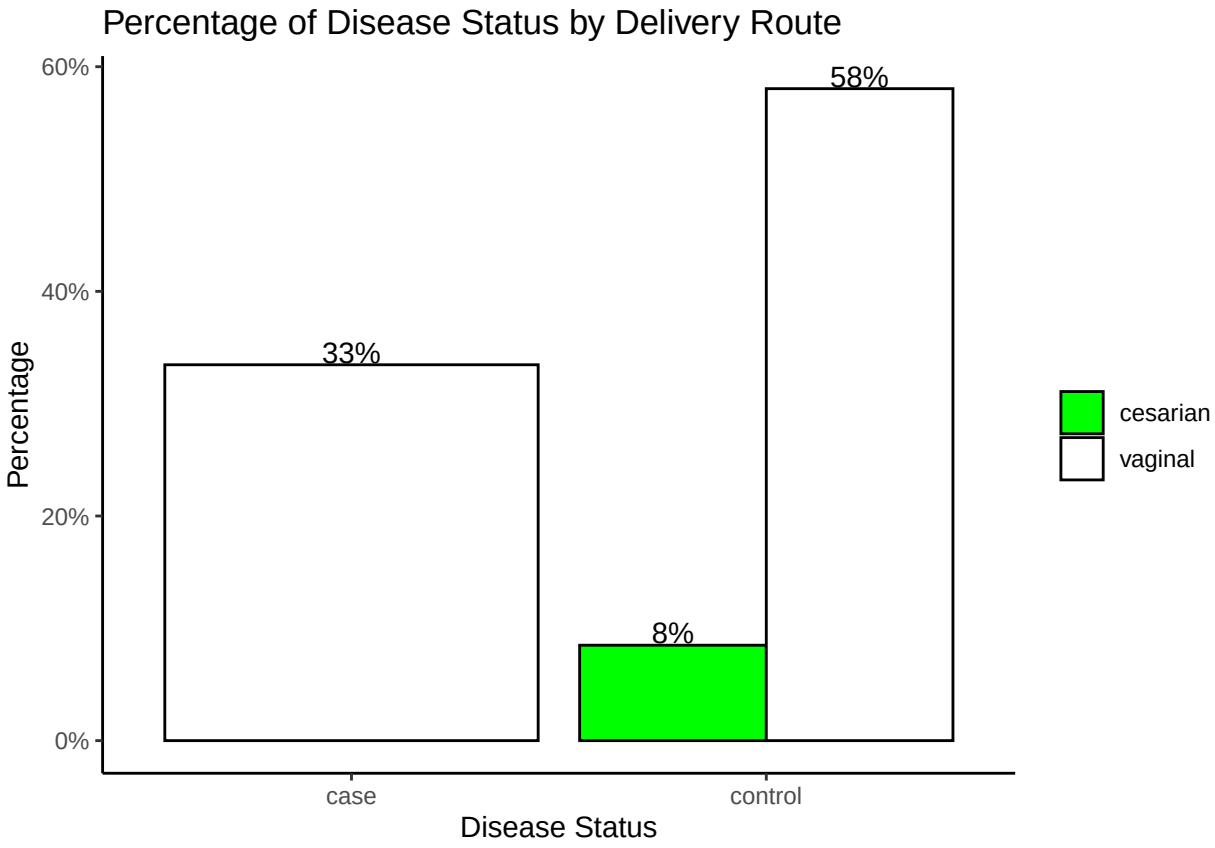
```
library (ggplot2)
bar_chart1 <- ggplot(diab_metadata, aes(x =Case_Control ,y= ((..count..)/sum(..count..)), fill= Gender))
  geom_bar(position = "dodge", color= "Black")+
  scale_y_continuous(labels=scales::percent)+
  labs(title = "Percentage of Disease Status by Gender",
       y= "Percentage", x= "Disease Status" )+
  geom_text(stat= "count", aes(label= scales::percent((..count..)/sum(..count..))), position = position_
  theme_classic()+
  scale_fill_brewer(palette = "spectral")
```

```
## Warning in pal_name(palette, type): Unknown palette spectral
```

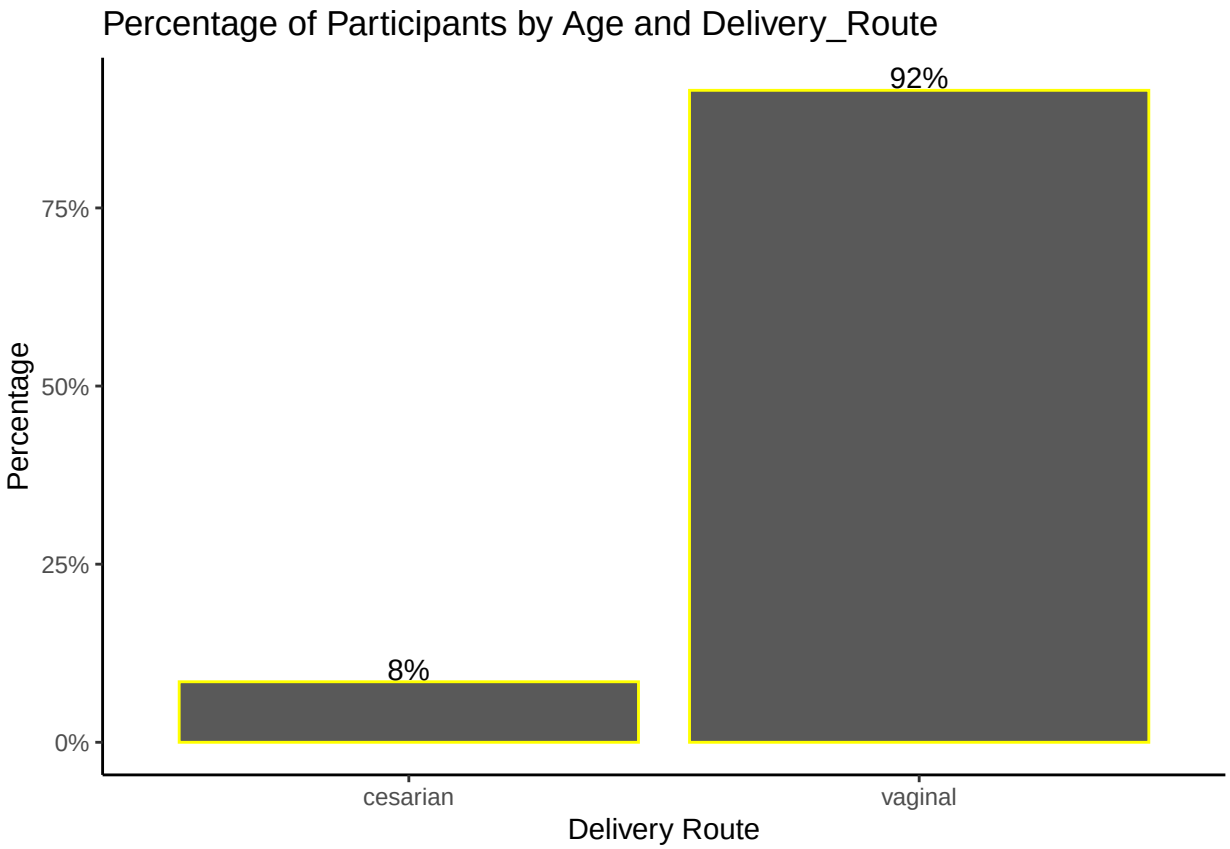
```
  bar_chart1
```



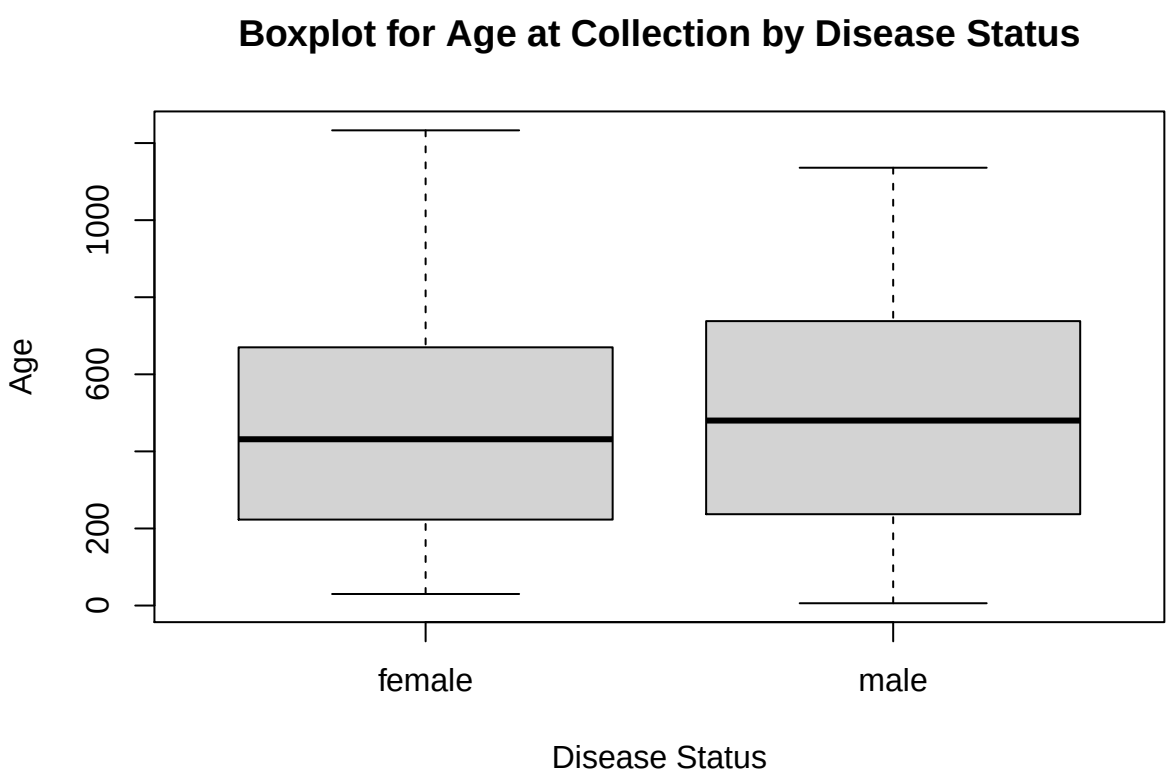
```
bar_chart2 <- ggplot(diab_metadata, aes(x =Case_Control ,y= (..count..)/sum(..count..), fill= Delivery_
  geom_bar(position = "dodge", color= "Black")+
  scale_y_continuous(labels=scales::percent)+
  labs(title = "Percentage of Disease Status by Delivery Route",
       y= "Percentage", x= "Disease Status" )+
  geom_text(stat= "count", aes(label= scales::percent((..count..)/sum(..count..))), position = position_
  theme_classic()+
  scale_fill_manual(name = "", values = c("green", "white"))
  bar_chart2
```




```
bar_chart3 <- ggplot(diab_metadata, aes(x = Delivery_Route,y= (..count..)/sum(..count..)), fill= Age_at_
  geom_bar(position = "dodge",color= "yellow")+
  scale_y_continuous(labels=scales::percent)+
  labs(title = "Percentage of Participants by Age and Delivery_Route",
       y= "Percentage", x= "Delivery Route")+
  geom_text(stat= "count", aes(label= scales::percent((..count..)/sum(..count..))), position = position_
  theme_classic()+
  scale_fill_manual(name = "", values = c("black", "yellow"))
  bar_chart3
```




```
boxplot(diab_metadata$Age_at_Collection~ diab_metadata$Gender, main="Boxplot for Age at Collection by
      ylab="Age")
```



ii. In addition, use appropriate test(s) to check for association/independency between disease status and other variables (delivery mode, gender and age). Note that age is given in days.

```
df1 <- xtabs(~diab_metadata$Gender + diab_metadata$Case_Control)

chisq.test(df1)
```

```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  df1
## X-squared = 0.30687, df = 1, p-value = 0.5796
```

```
df2 <- xtabs(~diab_metadata$Delivery_Route + diab_metadata$Case_Control)

chisq.test(df2)
```

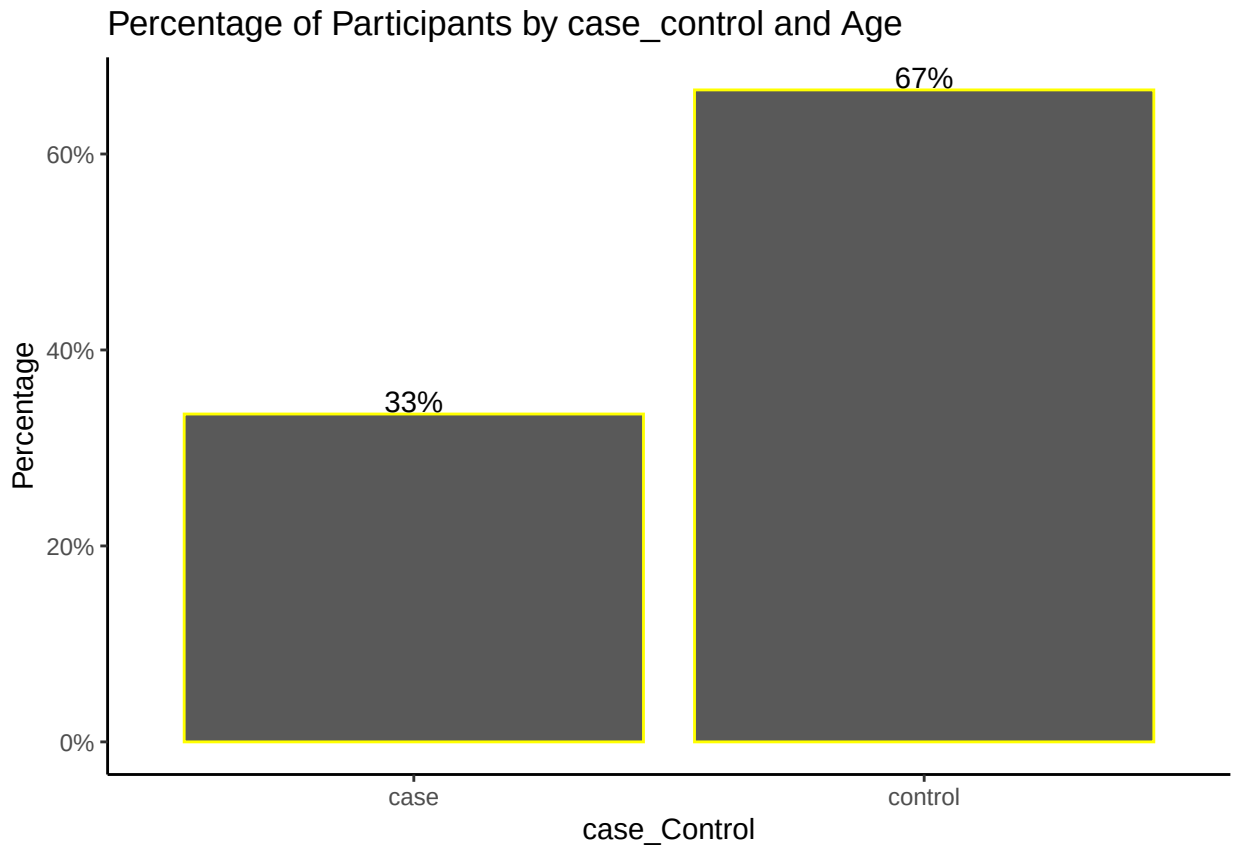
```
##
##  Pearson's Chi-squared test with Yates' continuity correction
##
## data:  df2
## X-squared = 34.649, df = 1, p-value = 3.949e-09
```

```
df3 <- xtabs(~diab_metadata$Age_at_Collection+ diab_metadata$Delivery_Route)

t.test(df3)
```

```
##
##  One Sample t-test
##
## data:  df3
## t = 28.078, df = 1085, p-value < 2.2e-16
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
##  0.6654712 0.7654681
## sample estimates:
## mean of x
## 0.7154696
```

```
bar_chart4 <- ggplot(diab_metadata, aes(x =Case_Control ,y= (..count..)/sum(..count..), fill= Age_at_Co
  geom_bar(position = "dodge",color= "yellow")+
  scale_y_continuous(labels=scales::percent)+
  labs(title = "Percentage of Participants by case_control and Age",
       y= "Percentage", x= "case_Control")+
  geom_text(stat= "count", aes(label= scales::percent((..count..)/sum(..count..))), position = position_
  theme_classic()+
  scale_fill_manual(name = "", values = c("black"))
  bar_chart4
```




```
model <- glm(diab_metadata$Case_Control ~ diab_metadata$Gender + diab_metadata$Delivery_Route + diab_me

summary(model)
```

```
##
## Call:
## glm(formula = diab_metadata$Case_Control ~ diab_metadata$Gender +
##     diab_metadata$Delivery_Route + diab_metadata$Age_at_Collection,
##     family = binomial)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.5032  -1.3854   0.9051   0.9596   1.0564
##
## Coefficients:
##                                    Estimate Std. Error z value Pr(>|z|)
## (Intercept)                       1.763e+01  4.862e+02   0.036    0.971
## diab_metadata$Gendermale          1.375e-01  1.566e-01   0.878    0.380
## diab_metadata$Delivery_Routevaginal -1.702e+01  4.862e+02  -0.035    0.972
## diab_metadata$Age_at_Collection     -2.614e-04  2.648e-04  -0.987    0.324
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 990.52  on 776  degrees of freedom
## Residual deviance: 932.07  on 773  degrees of freedom
## AIC: 940.07
##
## Number of Fisher Scoring iterations: 16
```

```
anova(model, test = "Chisq")
```

```
## Analysis of Deviance Table
##
## Model: binomial, link: logit
##
## Response: diab_metadata$Case_Control
##
## Terms added sequentially (first to last)
##
##
##                                Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
## NULL                                              776     990.52
## diab_metadata$Gender            1    0.397       775     990.13    0.5285
## diab_metadata$Delivery_Route    1   57.083       774     933.04 4.178e-14 ***
## diab_metadata$Age_at_Collection 1    0.973       773     932.07    0.3239
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
model1 = anova(model, test = "Chisq")
summary(model1)
```

```
##        Df       Deviance          Resid. Df       Resid. Dev       Pr(>Chi)
##  Min.   :1   Min.   : 0.3974   Min.   :773.0   Min.   :932.1   Min.   :0.0000
##  1st Qu.:1   1st Qu.: 0.6853   1st Qu.:773.8   1st Qu.:932.8   1st Qu.:0.1619
##  Median :1   Median : 0.9732   Median :774.5   Median :961.6   Median :0.3239
##  Mean   :1   Mean   :19.4845   Mean   :774.5   Mean   :961.4   Mean   :0.2841
##  3rd Qu.:1   3rd Qu.:29.0281   3rd Qu.:775.2   3rd Qu.:990.2   3rd Qu.:0.4262
```

```
##  Max.   :1   Max.   :57.0829   Max.   :776.0   Max.   :990.5   Max.   :0.5285
##  NA's   :1   NA's   :1                                         NA's   :1
```

2. Using phyloseq, create a phyloseq object. This will comprise the OTU abundance, taxonomy (provided in the .txt file) and sample data (provided in the .csv file).

```
library(tidyverse)
diabtaxa.data <- read_tsv("diabimmune_t1d_16s_otu_table.txt", skip = 1)
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   .default = col_double(),
##   ConsensusLineage = col_character()
## )
## i Use `spec()` for the full column specifications.
```

```
diabotu.data <- diabtaxa.data %>% select(1:778)
taxonomy <- diabtaxa.data %>% select(`#OTU ID`, ConsensusLineage) %>%
separate(ConsensusLineage, c("Domain", "Phylum", "Class", "Order", "Family", "Genus", "Species"), sep =
sampledata <- diab_metadata
```

Required to have similar row names in OTU and taxonomy table

Change OTU and taxonomy dataframes to matrices

```
diabotu.data_matrix <- as.matrix(diabotu.data)
taxonomy_matrix <- as.matrix(taxonomy)
```

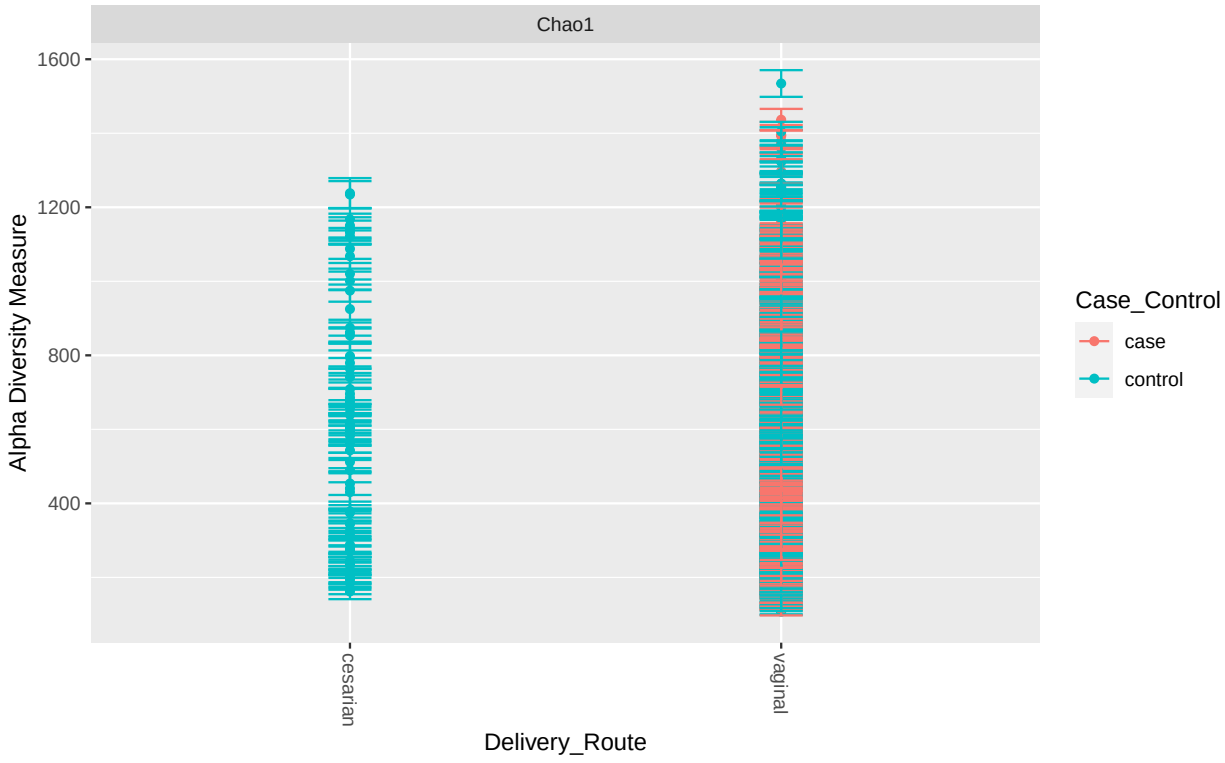
create phyloseq object

Validity checks

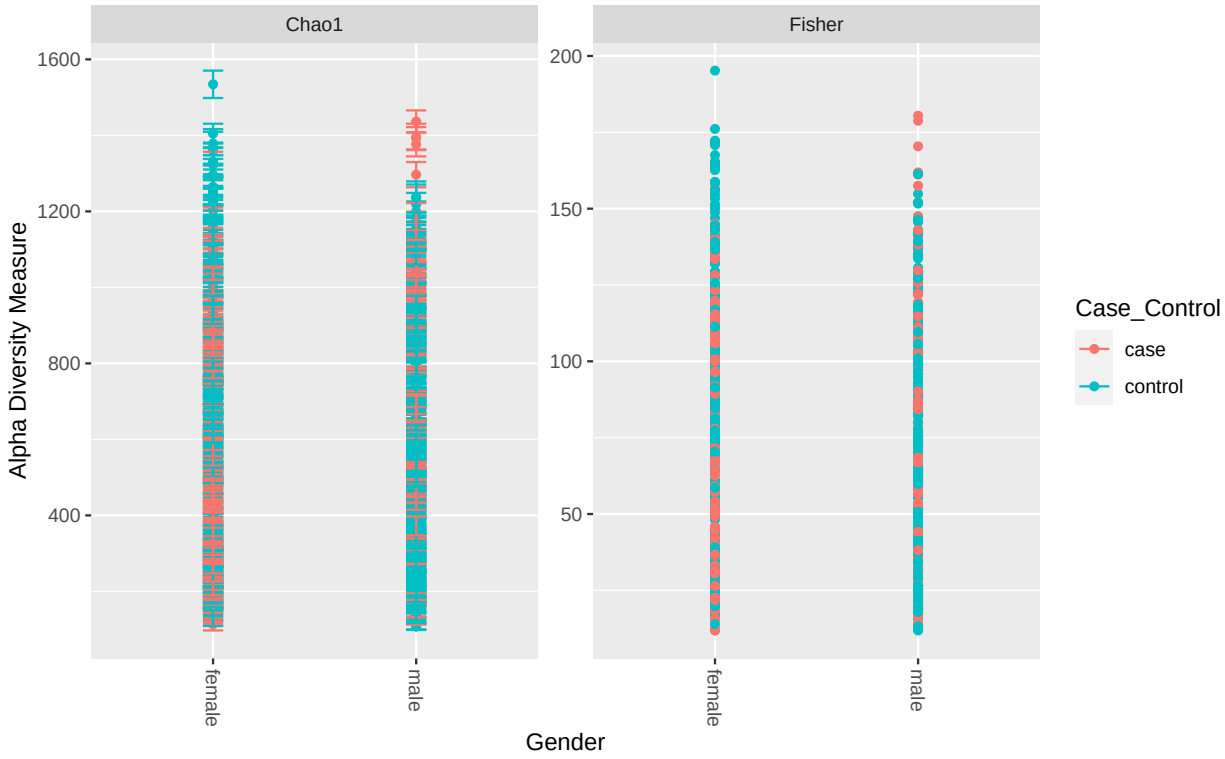
3. Generate Alpha diversity plots and ordination plots. Examine any observed patterns by delivery mode, gender and disease status.

Alpha Diversity

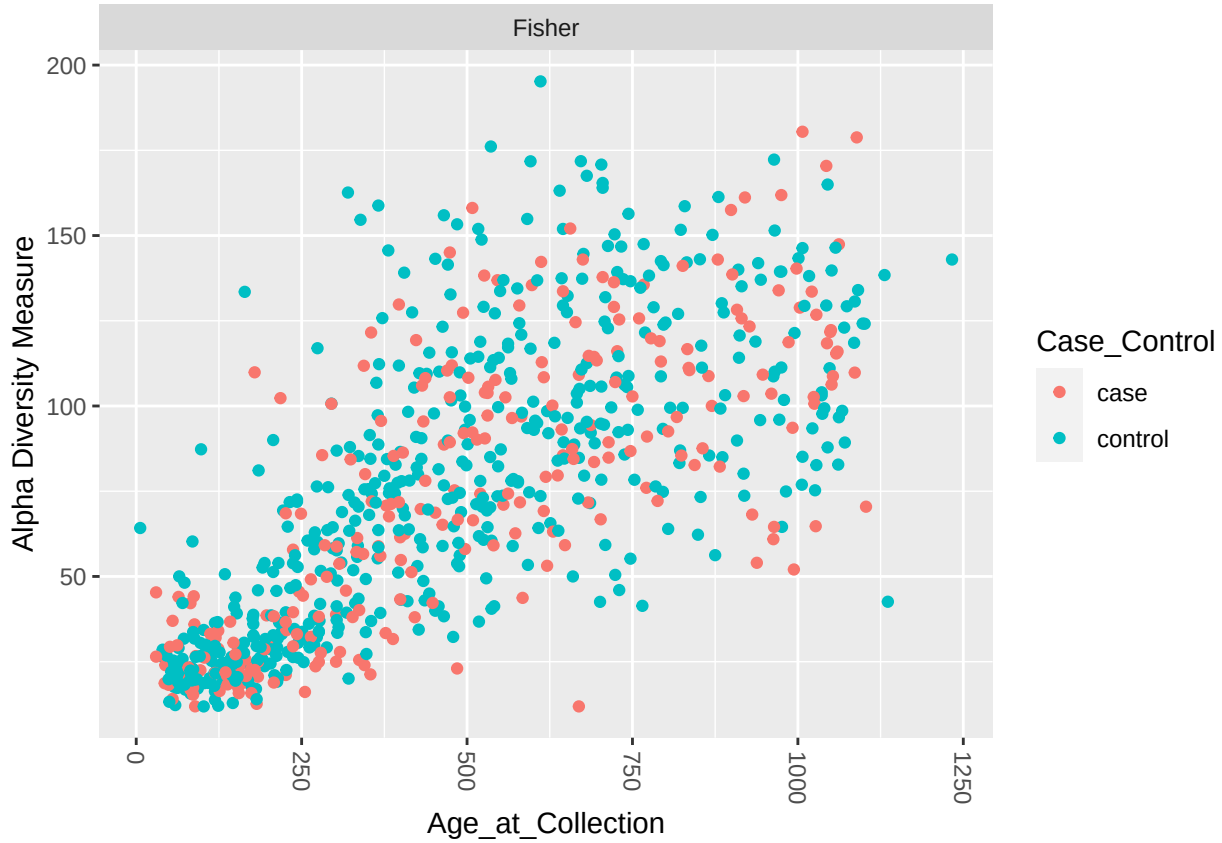
```
Fig.1 <- plot_richness(phyloseq.object1,x = "Delivery_Route", color="Case_Control", measures= "Chao1")
Fig.1
```



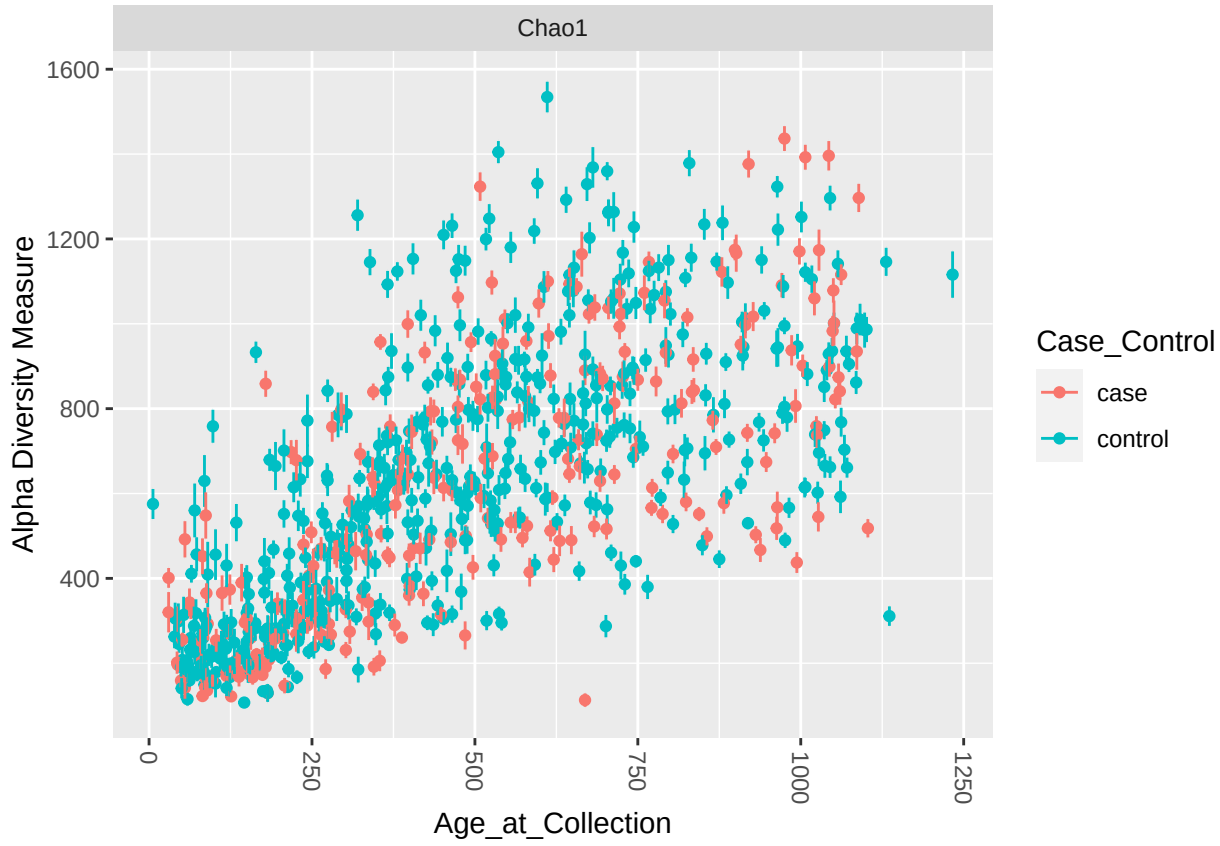

```
Fig.2 <- plot_richness(phyloseq.object1,x = "Gender", color="Case_Control", measures= c("Chao1", "fishe
Fig.2
```



```
Fig.3 <- plot_richness(phyloseq.object1,x = "Age_at_Collection", color="Case_Control", measures= c("fis
Fig.3
```

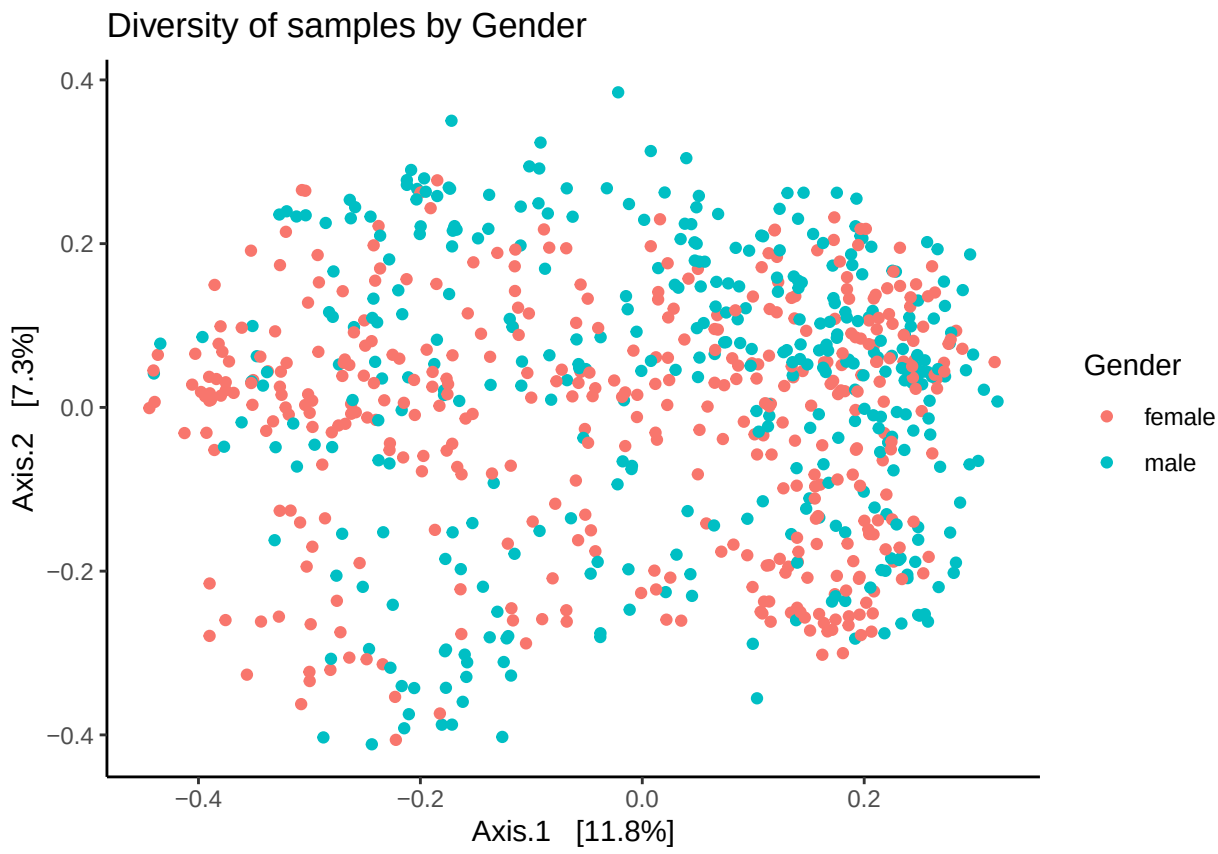



```
Fig.4 <- plot_richness(phyloseq.object1,x = "Age_at_Collection", color="Case_Control", measures= "Chao1
Fig.4
```



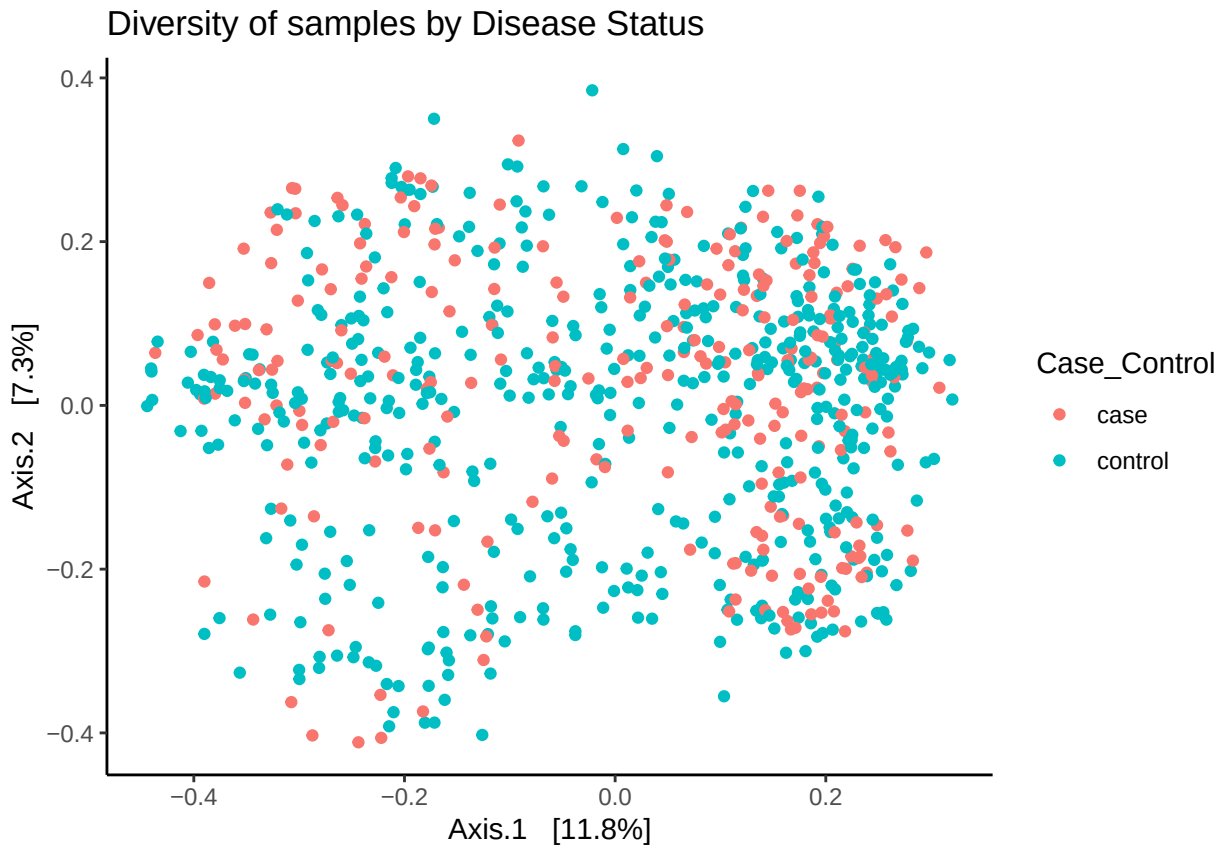
Ordination Plots

```
graphA <- ordinate(phyloseq.object1, "PCoA", "bray")%>%
plot_ordination(phyloseq.object1, .,color = "Gender", title = "Diversity of samples by Gender")+
  theme_classic()
graphA
```




```
library(phyloseq)
ordinate <- ordinate(phyloseq.object1, "PCoA", "bray")

graphB <- plot_ordination(phyloseq.object1, ordinate,color = "Case_Control", title = "Diversity of sampl
  theme_classic()
graphB
```

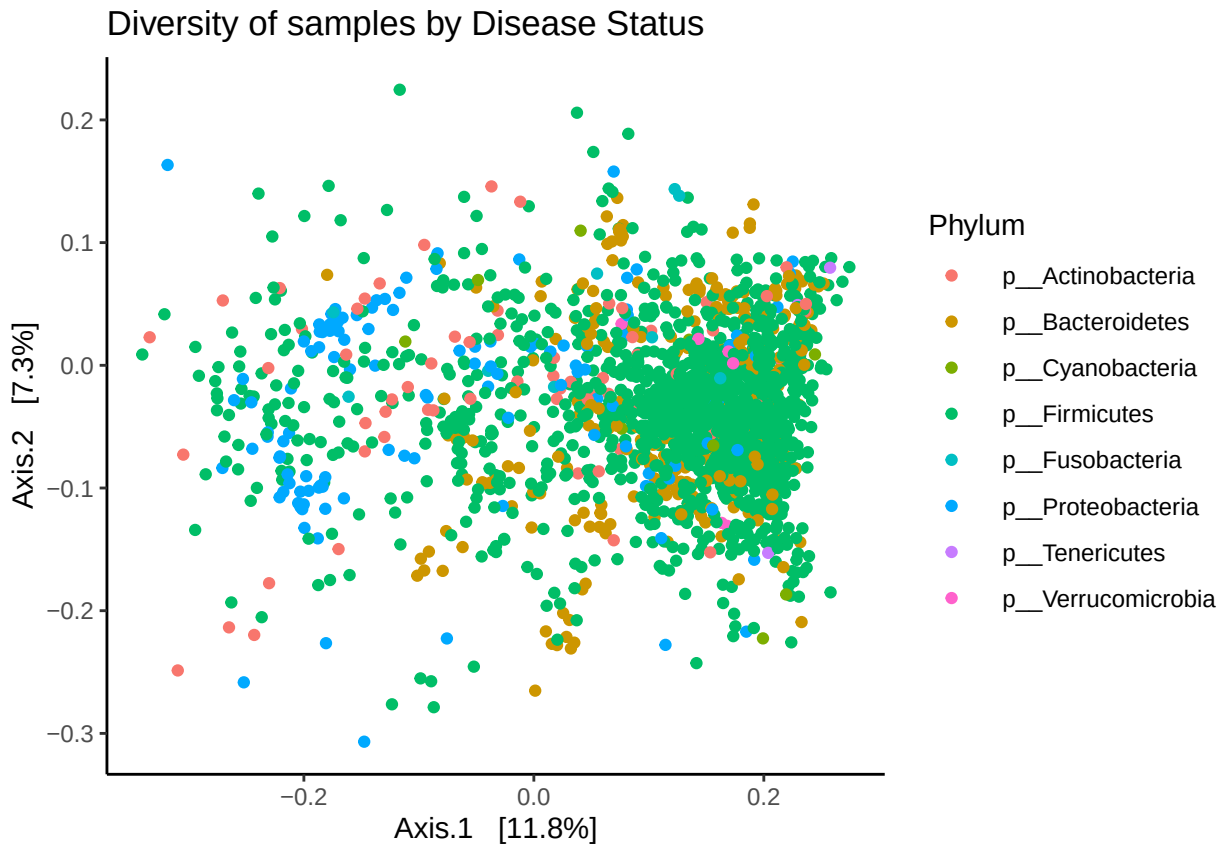



```
library(phyloseq)
ordinate1 <-ordinate(phyloseq.object1, "NMDS", "bray")
```

```
## Square root transformation
## Wisconsin double standardization
## Run 0 stress 0.181114
## Run 1 stress 0.1858986
## Run 2 stress 0.191604
## Run 3 stress 0.1900329
## Run 4 stress 0.1933333
## Run 5 stress 0.1837052
## Run 6 stress 0.1846929
## Run 7 stress 0.1891618
## Run 8 stress 0.1841374
## Run 9 stress 0.1829131
## Run 10 stress 0.1836451
## Run 11 stress 0.1910158
## Run 12 stress 0.1906417
## Run 13 stress 0.1842232
## Run 14 stress 0.1845027
## Run 15 stress 0.1909141
## Run 16 stress 0.1896413
## Run 17 stress 0.1820084
## Run 18 stress 0.1855057
## Run 19 stress 0.191682
```

```
## Run 20 stress 0.1865054
## *** No convergence -- monoMDS stopping criteria:
##      6: no. of iterations >= maxit
##      7: stress ratio > sratmax
##      7: scale factor of the gradient < sfgrmin
```

```
graphC <-plot_ordination(phyloseq.object1, ordinate,type="taxa", color="Phylum", title = "Diversity of s
  theme_classic()
graphC
```



4. Perform differential abundance using DEseq2

DESeq: Creating a DESeq object

```
phyloseq_object2 <- phyloseq(otu_table(diabotu.data_matrix + 1, taxa_are_rows=TRUE), tax_table(taxonomy
sample_data(phyloseq_object2) <- sample_data1

casecontrol = phyloseq_to_deseq2(phyloseq_object2, ~ Case_Control)
```

```
## converting counts to integer mode

##   it appears that the last variable in the design formula, 'Case_Control',
```

```
##   has a factor level, 'control', which is not the reference level. we recommend
##   to use factor(...,levels=...) or relevel() to set this as the reference level
##   before proceeding. for more information, please see the 'Note on factor levels'
##   in vignette('DESeq2').
```

DESeq test

Results table

```
res = results(casecontrol, cooksCutoff = FALSE)
alpha = 0.01
casetab = res[which(res$padj < alpha), ]
casetab = cbind(as(casetab, "data.frame"), as(tax_table(phyloseq_object2)[rownames(casetab), ], "matrix
```

plot_theme

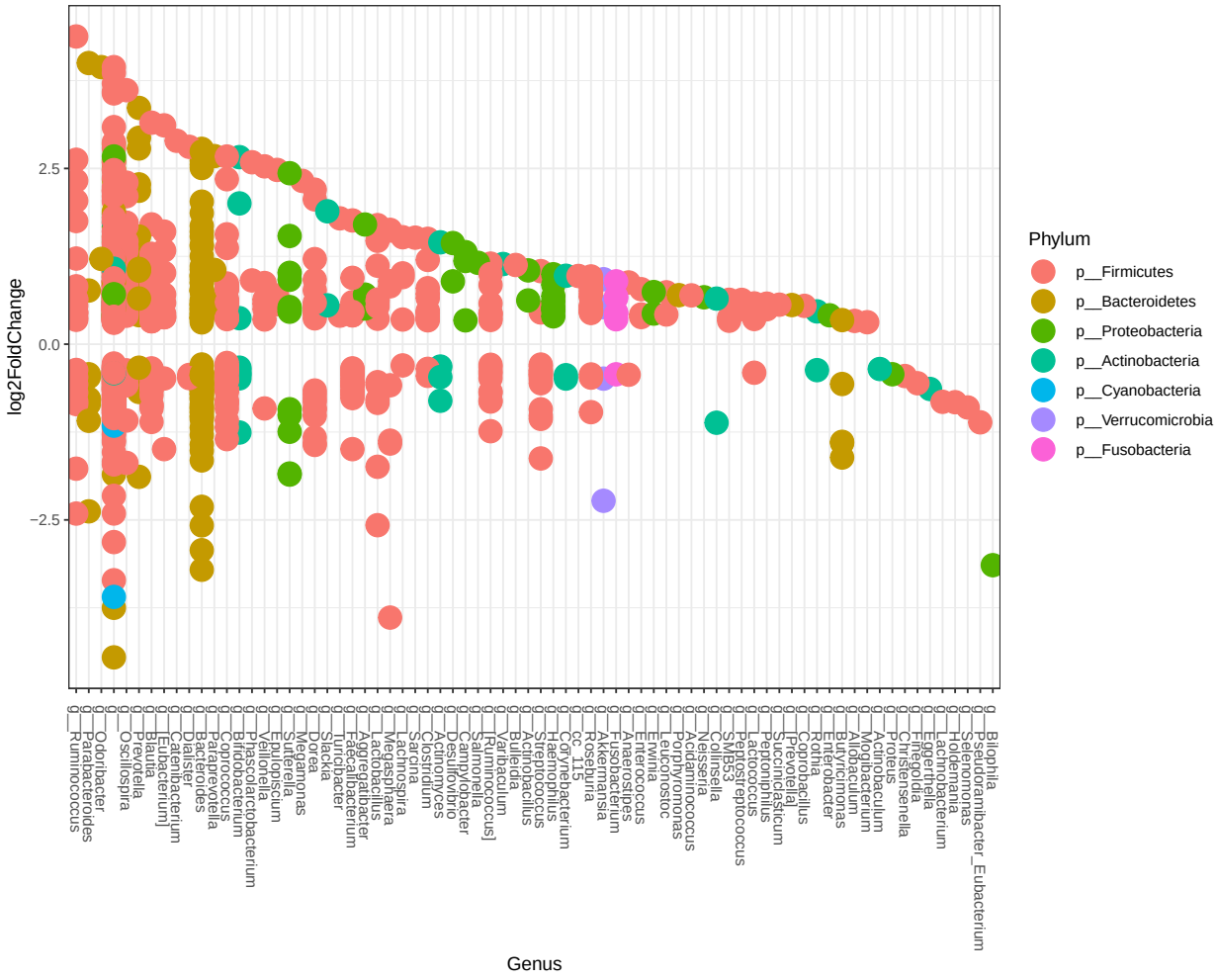
```
theme_set(theme_bw())
scale_fill_discrete <- function(palname = "Set1", ...) {
  scale_fill_brewer(palette = palname, ...)
}
```

Phylum order

```
x = tapply(casetab$log2FoldChange, casetab$Phylum, function(x) max(x))
x = sort(x, TRUE)
casetab$Phylum = factor(as.character(casetab$Phylum), levels=names(x))
```

Genus order

```
x = tapply(casetab$log2FoldChange, casetab$Genus, function(x) max(x))
x = sort(x, TRUE)
casetab$Genus = factor(as.character(casetab$Genus), levels=names(x))
ggplot(casetab, aes(x=Genus, y=log2FoldChange, color=Phylum)) + geom_point(size=6) +
  theme(axis.text.x = element_text(angle = -90, hjust = 0, vjust=0.5))
```


log2FoldChange
2.5
0.0
−2.5
Genus
g__Ruminococcus
g__Parabacteroides
g__Odoribacter
g__Oscillospira
g__Prevotella
g__Blautia
g__Eubacterium]
g__Dialister
g__Catenibacterium
g__Bacteroides
g__Paraprevotella
g__Coprococcus
g__Bifidobacterium
g__Pilibacter
g__Ellonella
g__Epulopiscium
g__Sutterella
g__Megamonas
g__Dorea
g__Slackia
g__Turicibacter
g__Faecalibacterium
g__Aggregatibacter
g__Lactobacillus
g__Howardella
g__Lachnospira
g__Sarcina
g__Clostridium
g__Actinomyces
g__Desulfovibrio
g__Campylobacter
g__Salmonella
g__[Ruminococcus]
g__Varibaculum
g__Bulleidia
g__Anaerofilum
g__Streptococcus
g__Haemophilus
g__Corynebacterium
g__cc_115
g__Roseburia
g__Akkermansia
g__Fusobacterium
g__Anaerostipes
g__Enterococcus
g__Lactonifactor
g__Leuconostoc
g__Porphyromonas
g__Acidaminococcus
g__Neisseria
g__Collinsella
g__SMB53
g__Peptostreptococcus
g__Lactococcus
g__Peptoniphilus
g__Succinivibrio
g__Finegoldia
g__Propionibacterium
g__Rothia
g__Enterobacter
g__Butyricimonas
g__Alloprevotella
g__Mogibacterium
g__Actinobaculum
g__Proteus
g__Christensenella
g__Eggerthella
g__Gordonibacter
g__Lactobacterium
g__Holdemania
g__Selenomonas
g__Pseudoramibacter_Eubacterium
g__Blautia
Phylum
p__Firmicutes
p__Bacteroidetes
p__Proteobacteria
p__Actinobacteria
p__Cyanobacteria
p__Verrucomicrobia
p__Fusobacteria