Test with DiGiulio data

Weijia Xiong

5/11/2020

Hi Weijia, Can you provide a summary of the real data that you explored, including number of patients, how many time points for each of them, are they balanced, etc? Basically, some exploratory data summary/analysis.

DiGiulio's Vaginal microbiome data is from 40 women. There are 927 samples and 1271 OTU. Each woman has different observation weeks so it is not balanced.

Load dataset

```
load("real data/DiGiulio.RData")
otu_data = as.data.frame(DiGiulio$OTU)  # 927 samples, 1271 OTU
taxonomy = DiGiulio$Taxonomy  # 1271
sampledata = DiGiulio$SampleData  #  927 samples, other covariates
```

Summarize the community structure and abundance with OTU table

Using otuReport from otuSummary package.

```
##combine with taxonomy
taxonomy =
  taxonomy %>%
  unite(taxon, Kingdom:Species, sep = ";", remove = FALSE)
otu_all = data.frame(t(otu_data),
                     taxonomy = taxonomy$taxon)
# 927 column samples + one taxonomy
# 1271 OTU rows
```

specify the taxonomic level: genus

```
## Using otuReport from otuSummary package
result = otuReport(otutab = otu_all, siteInCol = TRUE, taxhead = "taxonomy",  platform = "qiime", patte
```

```
## Filter 10% genus
genus_total = result$readSum
keep_genus = names(genus_total)[genus_total > quantile(genus_total,0.9)]
keep_genus
```

```
##  [1] "Bacteria;P:Actinobacteria;C:Actinobacteria;O:Actinomycetales;F:Corynebacteriaceae;Corynebacteri
##  [2] "Bacteria;P:Actinobacteria;C:Actinobacteria;O:Bifidobacteriales;F:Bifidobacteriaceae;Bifidobacte
##  [3] "Bacteria;P:Actinobacteria;C:Actinobacteria;O:Bifidobacteriales;F:Bifidobacteriaceae;Gardnerella
##  [4] "Bacteria;P:Actinobacteria;C:Coriobacteriia;O:Coriobacteriales;F:Coriobacteriaceae;Atopobium"
##  [5] "Bacteria;P:Bacteroidetes;C:Bacteroidia;O:Bacteroidales;F:Porphyromonadaceae;Porphyromonas"
##  [6] "Bacteria;P:Bacteroidetes;C:Bacteroidia;O:Bacteroidales;F:Prevotellaceae;Prevotella"
##  [7] "Bacteria;P:Bacteroidetes;C:Flavobacteriia;O:Flavobacteriales;F:Weeksellaceae;F:Weeksellaceae"
##  [8] "Bacteria;P:Firmicutes;C:Bacilli;O:Bacillales;F:Staphylococcaceae;Staphylococcus"
```

```
##  [9] "Bacteria;P:Firmicutes;C:Bacilli;O:Lactobacillales;F:Aerococcaceae;Aerococcus"
## [10] "Bacteria;P:Firmicutes;C:Bacilli;O:Lactobacillales;F:Lactobacillaceae;Lactobacillus"
## [11] "Bacteria;P:Firmicutes;C:Bacilli;O:Lactobacillales;F:Streptococcaceae;Streptococcus"
## [12] "Bacteria;P:Firmicutes;C:Clostridia;O:Clostridiales;F:Clostridiaceae;Clostridium"
## [13] "Bacteria;P:Firmicutes;C:Clostridia;O:Clostridiales;F:Peptostreptococcaceae;Peptostreptococcus"
## [14] "Bacteria;P:Firmicutes;C:Clostridia;O:Clostridiales;F:Tissierellaceae;1-68"
## [15] "Bacteria;P:Firmicutes;C:Clostridia;O:Clostridiales;F:Tissierellaceae;Anaerococcus"
## [16] "Bacteria;P:Firmicutes;C:Clostridia;O:Clostridiales;F:Tissierellaceae;Finegoldia"
## [17] "Bacteria;P:Firmicutes;C:Clostridia;O:Clostridiales;F:Tissierellaceae;Peptoniphilus"
## [18] "Bacteria;P:Firmicutes;C:Clostridia;O:Clostridiales;F:Tissierellaceae;WAL_1855D"
## [19] "Bacteria;P:Firmicutes;C:Clostridia;O:Clostridiales;F:Veillonellaceae;Dialister"
## [20] "Bacteria;P:Firmicutes;C:Clostridia;O:Clostridiales;F:Veillonellaceae;Megasphaera"
## [21] "Bacteria;P:Firmicutes;C:Clostridia;O:Clostridiales;F:Veillonellaceae;Veillonella"
## [22] "Bacteria;P:Firmicutes;C:Clostridia;O:Clostridiales;O:Clostridiales;O:Clostridiales"
## [23] "Bacteria;P:Fusobacteria;C:Fusobacteriia;O:Fusobacteriales;F:Fusobacteriaceae;Fusobacterium"
## [24] "Bacteria;P:Proteobacteria;C:Betaproteobacteria;O:Burkholderiales;F:Alcaligenaceae;Oligella"
## [25] "Bacteria;P:Proteobacteria;C:Epsilonproteobacteria;O:Campylobacterales;F:Campylobacteraceae;Camp
## [26] "Bacteria;P:Proteobacteria;C:Gammaproteobacteria;O:Enterobacteriales;F:Enterobacteriaceae;F:Ente
## [27] "Bacteria;P:Tenericutes;C:Mollicutes;O:Mycoplasmatales;F:Mycoplasmataceae;Ureaplasma"
```

```
genus_reads=
  as.data.frame(
  result$reads,
  row.names = rownames(result$reads)
  )
genus_reads$total = as.numeric(genus_total)

genus_reads_filter = genus_reads[which(row.names(genus_reads) %in% keep_genus),]

dim(genus_reads_filter)
```

```
## [1]  27 928
```

Gain the abundance dataset

```
genus_abundance = as.data.frame(result$Relabund)
rownames(genus_abundance) = rownames(result$reads)
genus_abundance_filter = genus_abundance[which(row.names(genus_abundance) %in% keep_genus),]
dim(genus_abundance_filter)
```

```
## [1]  27 927
```

The number of columns: 927, which represents 927 samples. The number of rows: 269, which represents 269 genus.

After filter, there remains 27 bacteria.

Combine with sample information

```
genus_abundance_filter_dat = data.frame(t(genus_abundance_filter))

otu_covariate_all=
  cbind(sampledata, genus_abundance_filter_dat)

head(otu_covariate_all[,1:9])
```

```
##     SampleID Subject weeks            Race NumReads  Preg preterm CST
## 1 1000301298   10003    29 American Indian     2341  TRUE    Term   0
## 2 1000301308   10003    30 American Indian     1136  TRUE    Term   0
## 3 1000301318   10003    31 American Indian     2344  TRUE    Term   0
## 4 1000301328   10003    32 American Indian     1854  TRUE    Term   0
## 5 1000301338   10003    33 American Indian     1839  TRUE    Term   0
## 6 1000301488   10003    46 American Indian     3265 FALSE    Term   0
##   Bacteria.P.Actinobacteria.C.Actinobacteria.O.Actinomycetales.F.Corynebacteriaceae.Corynebacterium
## 1                                                                                        0.00000000
## 2                                                                                        0.00000000
## 3                                                                                        0.08532423
## 4                                                                                        0.21574973
## 5                                                                                        1.30505710
## 6                                                                                        0.03062787
```

```
colnames(otu_covariate_all) = replace(colnames(otu_covariate_all), 9:35,
        sapply(1:27,function(x){str_c("genus_",x)}))
```

Here each row represent one sample.

```
baseline_ID =
  otu_covariate_all %>%
  group_by(Subject) %>%
  summarise(baseline = first(SampleID))

baseline_data =
  otu_covariate_all %>%
  filter(SampleID %in% baseline_ID$baseline)

visit_data =
  otu_covariate_all %>%
  filter(!SampleID %in% baseline_ID$baseline)
```

```
head(baseline_data[,1:11])
```

```
##     SampleID Subject weeks            Race NumReads Preg preterm CST    genus_1
## 1 1000301298   10003    29 American Indian     2341 TRUE    Term   0 0.00000000
## 2 1000401368   10004    38           White     1527 TRUE    Term   0 0.06548788
## 3 1000501278   10005    27  Asian-Japanese     1181 TRUE    Term   0 0.00000000
## 4 1000601178   10006    17           White     1636 TRUE    Term   0 0.00000000
## 5 1000801248   10008    25           White     2281 TRUE    Term   0 0.21920210
## 6 1000901308   10009    31           White     1686 TRUE    Term   0 0.00000000
##   genus_2     genus_3
## 1       0  0.12815036
## 2       0  0.00000000
## 3       0  0.08467401
## 4       0  0.00000000
## 5       0 19.02674266
## 6       0  6.40569395
```

```
head(visit_data[,1:11])
```

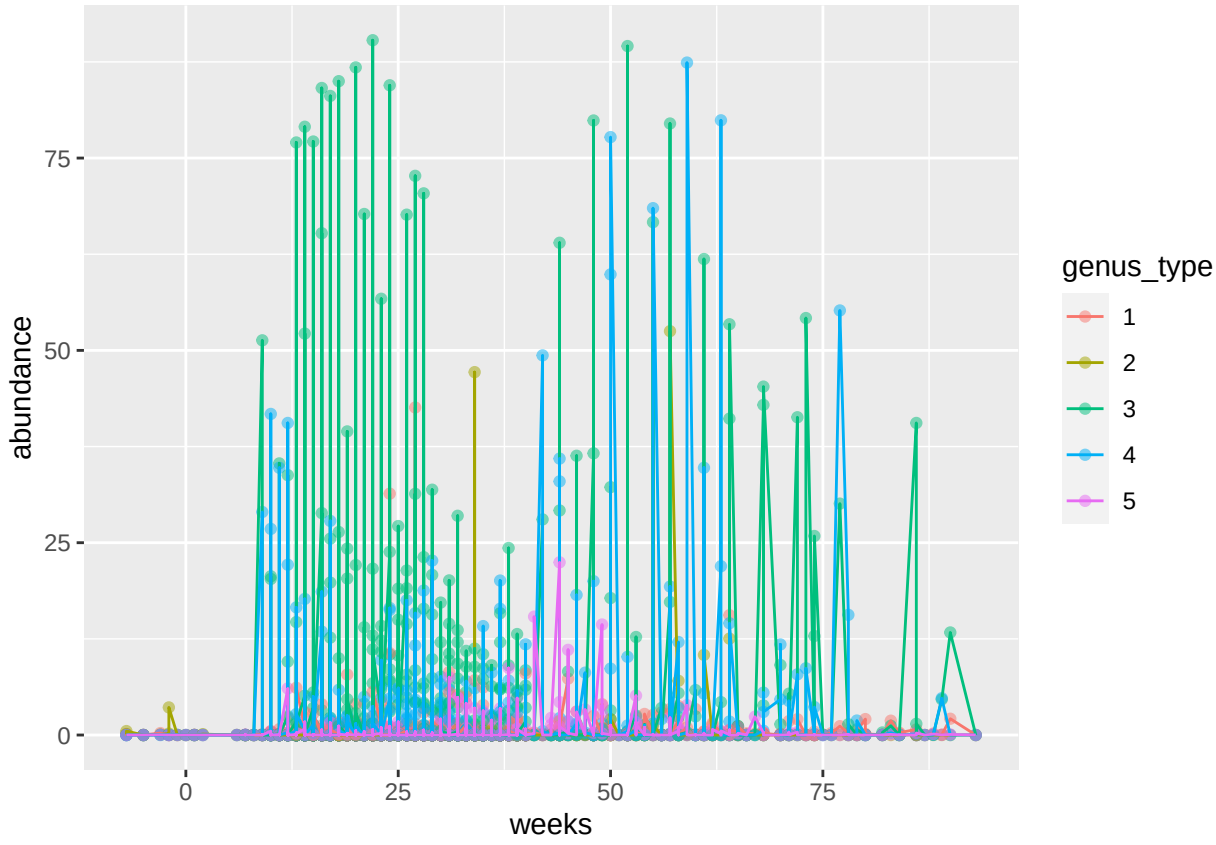
```
##     SampleID Subject weeks            Race NumReads  Preg preterm CST
## 1 1000301308   10003    30 American Indian     1136  TRUE    Term   0
## 2 1000301318   10003    31 American Indian     2344  TRUE    Term   0
## 3 1000301328   10003    32 American Indian     1854  TRUE    Term   0
```

```
## 4 1000301338   10003    33 American Indian     1839  TRUE    Term   0
## 5 1000301488   10003    46 American Indian     3265 FALSE    Term   0
## 6 1000301528   10003    50 American Indian     4801 FALSE    Term   0
##      genus_1 genus_2   genus_3
## 1 0.00000000       0 0.0000000
## 2 0.08532423       0 0.1279863
## 3 0.21574973       0 0.4854369
## 4 1.30505710       0 0.1087548
## 5 0.03062787       0 0.0000000
## 6 0.00000000       0 0.0000000
```

Not the same weeks for each subject.

```
test_genus =
  otu_covariate_all %>%
  pivot_longer(
    genus_1:genus_5,
    names_to = "genus_type",
    names_prefix = "genus_",
    values_to = "abundance"
  )

test_genus %>%
  ggplot(aes(x = weeks, y = abundance)) +
  geom_point(aes(x = weeks, y = abundance, color = genus_type),alpha = 0.5) +
  geom_line(aes(x = weeks, y = abundance, color = genus_type))
```




```
save.image(file = "Explore_DiGiulio.RData")
```