cluster using phyloseq

Weijia Xiong

6/30/2020

Load data

```
otu_file <- "data/PregnancyClosed15.RData"
load(otu_file)
```

Transform the data (proportions):

```
site <- "Vaginal_Swab"
ps <- PSPreg[[site]]
tt <- data.frame(tax_table(ps))
ps <- transform_sample_counts(ps, function(OTU) OTU/sum(OTU))
```

We are not doing differential abundance analysis here, so the proportion transformation is used for exploratory analyses only.

```
summary(sample_data(ps)$Outcome)
```

```
##    Marginal     Preterm        Term VeryPreterm
##          83          64         571          43
```

```
ps_preterm = subset_samples(ps, Outcome %in% c("Preterm","VeryPreterm"))
ps_term = subset_samples(ps, Outcome %in% c("Term","Marginal"))
```

Term data cluster

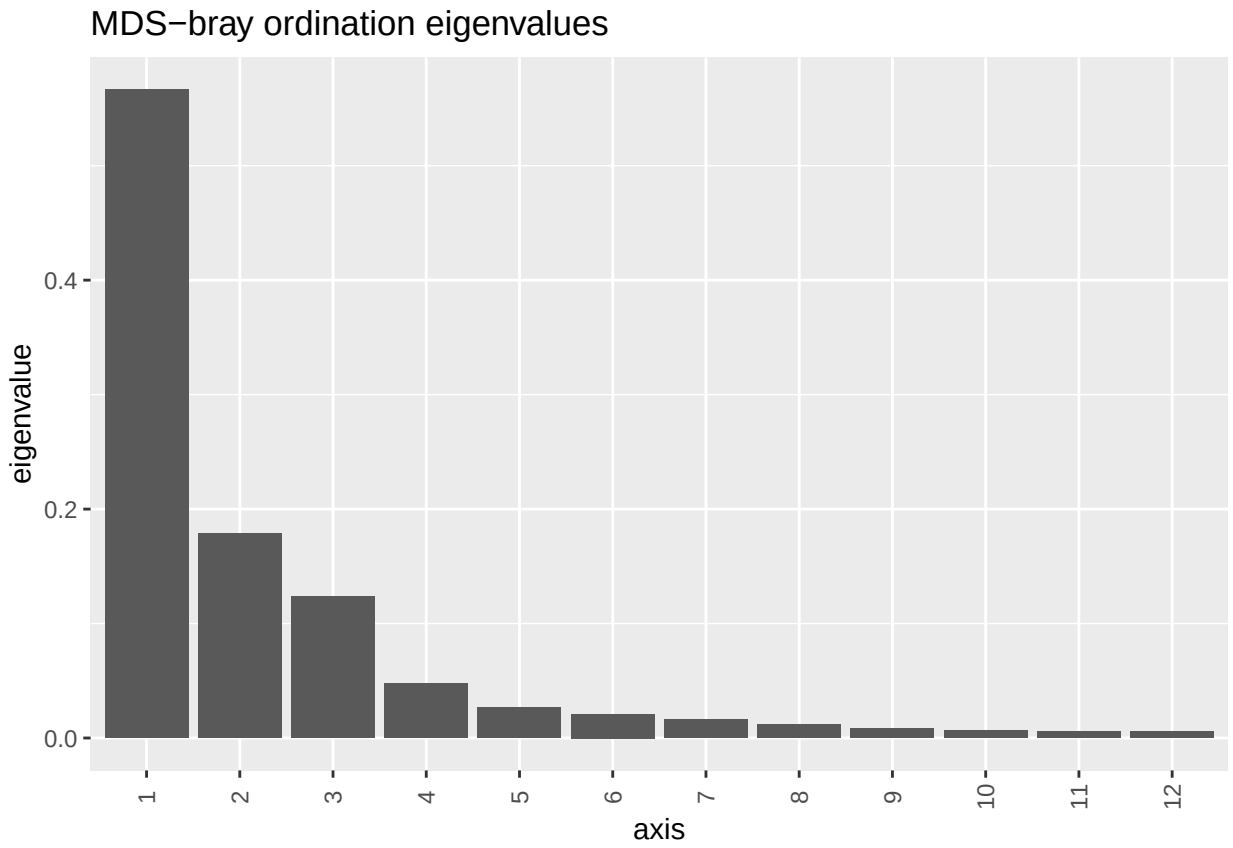
The vaginal community is dominated by closely related, but functionally distinct, Lactobacillus species. Therefore it is better to use a non-phylogenetically aware distance measure so as to be able to separate these species. Start with an MDS (or PCoA) ordination:

```
braydist <- phyloseq::distance(ps_term, method="bray")
ord = ordinate(ps, method = "MDS", distance = braydist)
## based in some fashion on the abundance table ultimately stored as a contingency matrix (otu_table-cl

# MDS: Performs principal coordinate analysis (also called principle coordinate decomposition, multidim

# Need a distance matrix, here use bray-curtis disctance

plot_scree(ord) + xlim(as.character(seq(1,12))) + ggtitle("MDS-bray ordination eigenvalues")
```




```
# p1 = plot_ordination(ps, ord, type="taxa", color="Phylum", title="taxa")
# print(p1)

evs <- ord$value$Eigenvalues
print(evs[1:20])
```

```
##  [1] 116.6689774  36.8329781  25.4839268   9.8136771   5.4647095   4.3200964
##  [7]   3.3399353   2.4345698   1.6683111   1.3444952   1.2280786   1.2082681
## [13]   0.8565684   0.7421970   0.7047971   0.6730503   0.6214064   0.5451675
## [19]   0.5306053   0.5036866
```

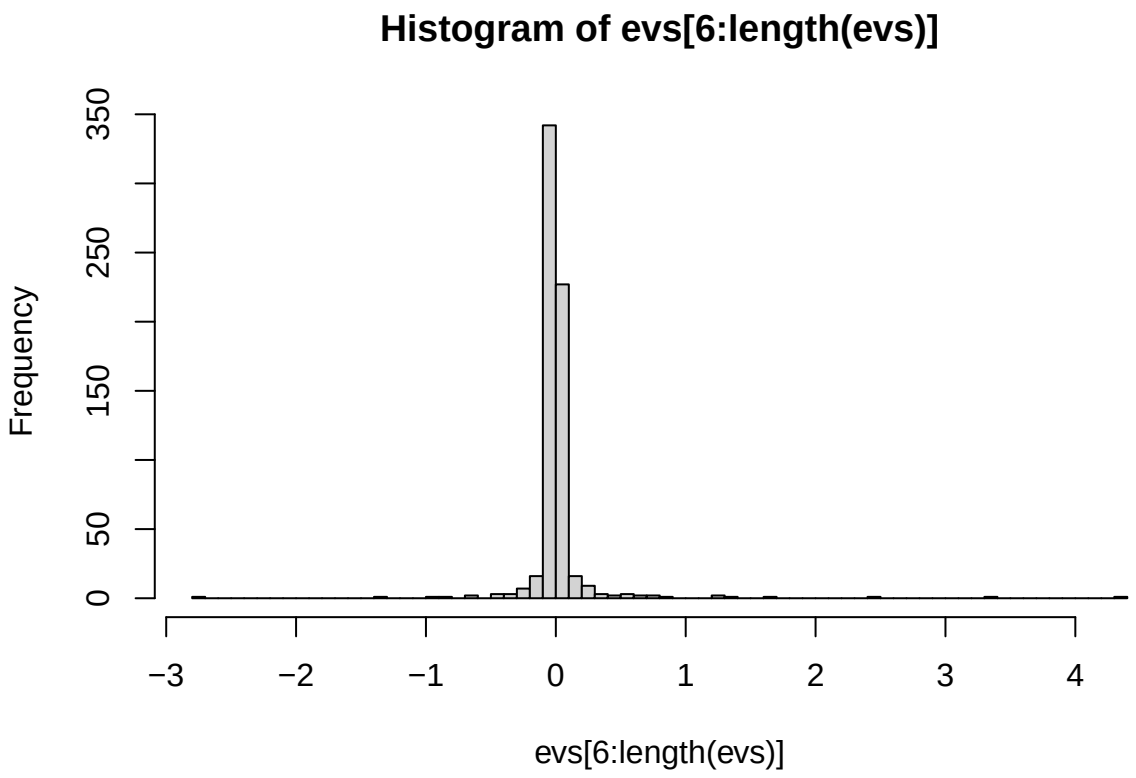
```
print(tail(evs))
```

```
## [1] -0.6061663 -0.6389676 -0.8712937 -0.9785011 -1.3789373 -2.7454736
```

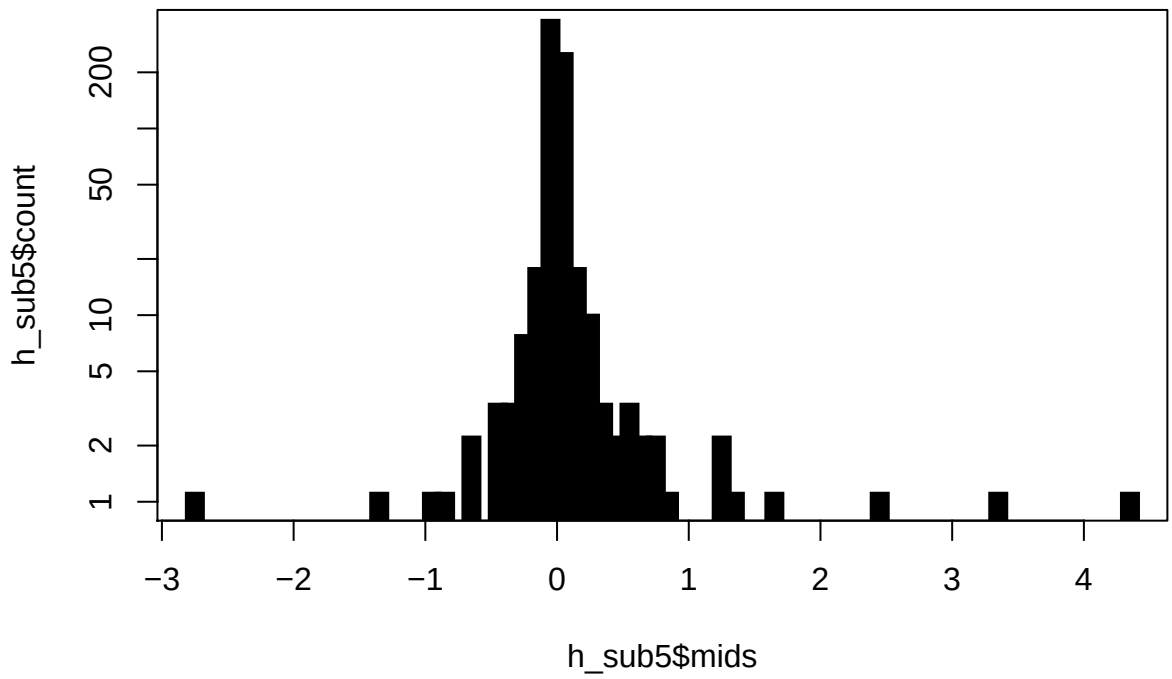
Denoise distance matrix

We would like to clean some of the noise from the data by restricting this to the truly significant dimensions. The top 5 eigenvalues are clearly very significant, but let's keep all the positive eigenvalues that clearly exceed the magnitude of the smallest negative eigenvalues:

```
h_sub5 <- hist(evs[6:length(evs)], 100)
```




```
plot(h_sub5$mids, h_sub5$count, log="y", type='h', lwd=10, lend=2)
```

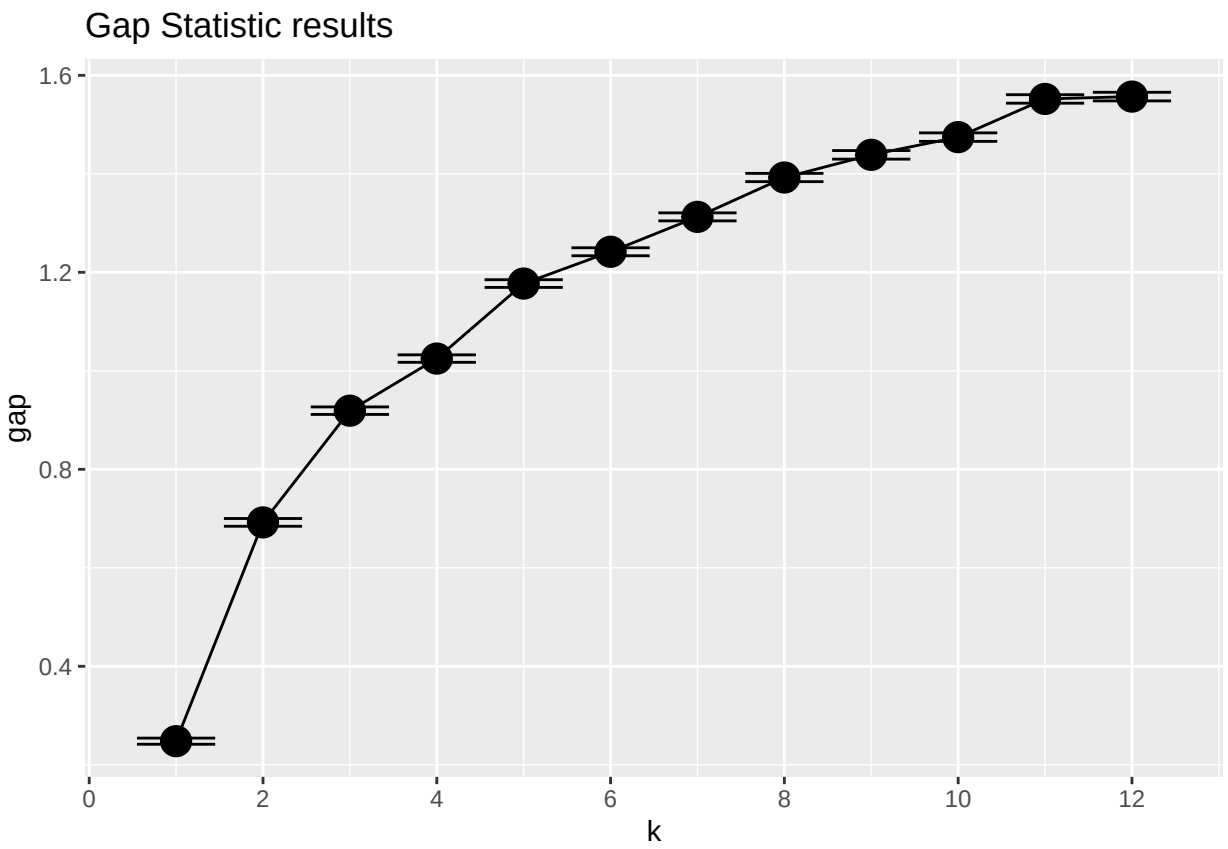


Looks like eigenvalues 6 and 7 still stand out, so we'll go with 7 MDS dimensions.

Determine number of clusters

We will use the gap statistic to indicate the number of clusters in this data:

```
NDIM <- 7
x <- ord$vectors[,1:NDIM]  # rows=sample, cols=MDS axes, entries = value
pamPCoA = function(x, k) {
    list(cluster = pam(x[,1:2], k, cluster.only = TRUE))
}
gs = clusGap(x, FUN = pamPCoA, K.max = 12, B = 50)
plot_clusgap(gs) + scale_x_continuous(breaks=c(seq(0, 12, 2)))
```



The gap statistic strongly suggests at least three clusters, but makes another big jump at K=5 before the slope gets a lot smaller. So, K=5 it is.

Cluster into CSTs

Perform PAM 5-fold clusters:

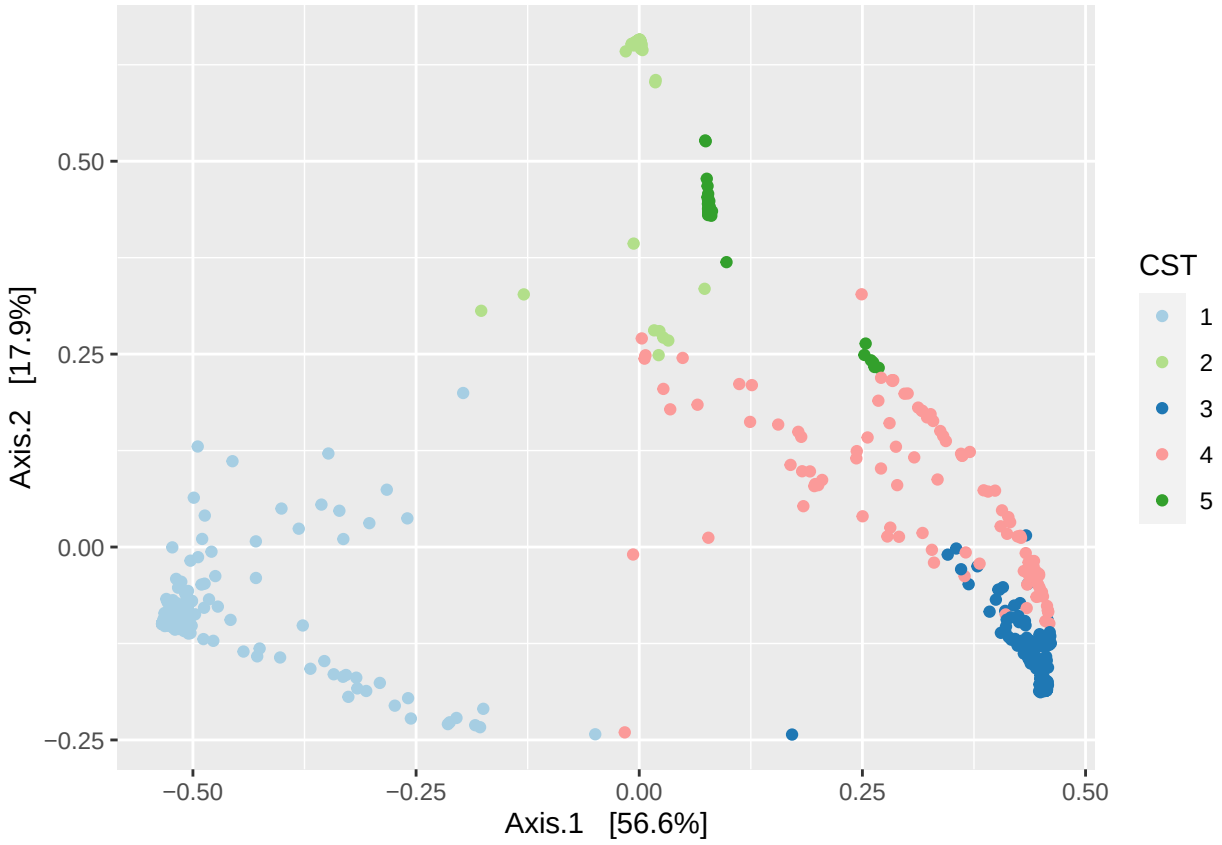
```
K <- 5
x <- ord$vectors[,1:NDIM]
clust <- as.factor(pam(x, k=K, cluster.only=T))
# SWAPPING THE ASSIGNMENT OF 2 AND 3 TO MATCH RAVEL CST ENUMERATION
clust[clust==2] <- NA
clust[clust==3] <- 2
clust[is.na(clust)] <- 3
sample_data(ps_term)$CST <- clust
CSTs <- as.character(seq(K))
```

Evaluate clustering

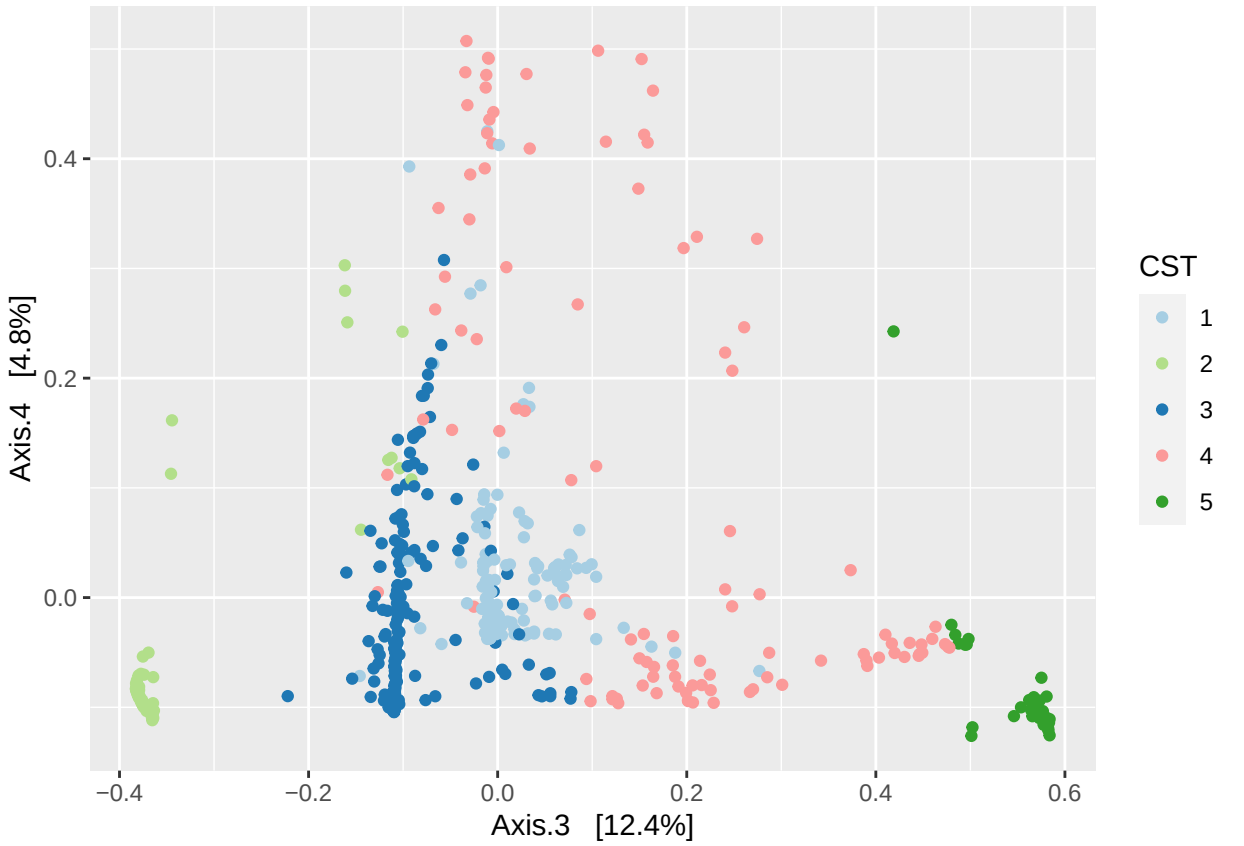
Inspect the results in MDS and NMDS ordinations:

```
CSTColors <- brewer.pal(6,"Paired")[c(1,3,2,5,4,6)] # Length 6 for consistency with pre-revision CST+ c
names(CSTColors) <- CSTs
CSTColorScale <- scale_colour_manual(name = "CST", values = CSTColors[1:5])
CSTFillScale <- scale_fill_manual(name = "CST", values = CSTColors[1:5])
# grid.arrange(plot_ordination(ps, ord, color="CST") + CSTColorScale,
#              plot_ordination(ps, ord, axes=c(3,4), color="CST") + CSTColorScale, main="Ordination by

plot_ordination(ps_term, ord, color="CST") + CSTColorScale
```




```
plot_ordination(ps_term, ord, axes=c(3,4), color="CST")  + CSTColorScale
```




```
nmds = ordinate(ps_term, method="NMDS", distance=braydist)
```

```
## Run 0 stress 0.1430354
## Run 1 stress 0.1919882
## Run 2 stress 0.1922407
## Run 3 stress 0.1946097
## Run 4 stress 0.1865393
## Run 5 stress 0.1918076
## Run 6 stress 0.1857592
## Run 7 stress 0.193173
## Run 8 stress 0.1759438
## Run 9 stress 0.1932515
## Run 10 stress 0.1952003
## Run 11 stress 0.1758556
## Run 12 stress 0.1933778
## Run 13 stress 0.1902203
## Run 14 stress 0.1808392
## Run 15 stress 0.1846196
## Run 16 stress 0.1630381
## Run 17 stress 0.1927165
## Run 18 stress 0.189357
## Run 19 stress 0.1940379
## Run 20 stress 0.1837754
## *** No convergence -- monoMDS stopping criteria:
##     19: stress ratio > sratmax
##      1: scale factor of the gradient < sfgrmin
```

```
plot_NMDS_bray_by_cluster = plot_ordination(ps,nmds, color="CST") + CSTColorScale + ggtitle("NMDS -- bra
```

```
sample_data(ps_term)$clust <- clust
samdf <- data.frame(sample_data(ps_term))
table(samdf$clust)
```

```
##
##   1   2   3   4   5
## 256  57 202 105  34
```