CART rebuttal

Anqi Dai

2021-07-21

Table of Contents

library(vegan)  
library(ggpubr)  
library(tidyverse)  
library(grid)  
library(cowplot)

# Reivewer 1

## Major point 2

**LEfSe is biased towards the identification of rare taxa and these rare taxa may be present in only one or two individuals. For this reason, please define parameters for minimum relative abundance and minimum number of patients in which taxa are seen. Please provide the relative abundance of the taxa of interest and the number of patients in which they are observed that are differentially abundant between groups.**

# sort out the asv counts table and also do filtering (need to have all taxa levels)  
meta <- read\_csv('../data/amplicon/stool/combined\_2\_meta.csv')  
  
library(vdbR)  
connect\_database('~/dbConfig.txt')  
get\_table\_from\_database('asv\_annotation\_blast\_ag')  
cts <- get\_counts\_subset(meta$sampleid)  
  
cts\_ <- cts %>%   
 select(asv\_key, sampleid, count\_relative) %>%   
 spread(key = 'sampleid', value = 'count\_relative', fill = 0)   
  
annot <- asv\_annotation\_blast\_ag %>%   
 filter(asv\_key %in% cts\_$asv\_key) %>%   
 mutate(ordr = if\_else(is.na(ordr), str\_glue('unknown\_of\_class\_{class}'), ordr),  
 family = if\_else(is.na(family), str\_glue('unknown\_of\_order\_{ordr}'), family),  
 genus = if\_else(is.na(genus) , str\_glue('unknown\_of\_family\_{family}'), genus),  
 species = if\_else(is.na(species) , str\_glue('unknown\_of\_genus\_{genus}'), species)) %>%   
 mutate(taxa\_asv = str\_glue('k\_\_{kingdom}|p\_\_{phylum}|c\_\_{class}|o\_\_{ordr}|f\_\_{family}|g\_\_{genus}|s\_\_{species}|a\_\_{asv\_key}'))  
  
cts\_all <- cts\_ %>%   
 full\_join(annot %>% select(asv\_key, taxa\_asv), by = 'asv\_key') %>%   
 select(-asv\_key) %>%   
 gather('sampleid', 'relab', names(.)[1]:names(.)[ncol(.)-1]) %>%   
 left\_join(meta %>% select(sampleid, cr\_d100, toxicity), by = 'sampleid')  
  
# the asv to keep  
# keep the asvs that show up in at least 25% of the samples  
keepg <- cts\_all %>%   
 filter(relab > 0.0001) %>%  
 ungroup() %>%   
 count(taxa\_asv) %>%   
 filter(n > floor(nrow(meta) \* 0.25)) %>%   
 pull(taxa\_asv)  
   
cts\_fil <- cts\_all %>%   
 filter(taxa\_asv %in% keepg) %>%   
 spread('sampleid', 'relab', fill = 0)

# the pheno label for the samples  
pheno <- meta %>%   
 select(center, cr\_d100:crs, icans, sampleid) %>%   
 gather('pheno', 'value', cr\_d100:icans)   
  
all\_sub\_pheno <- pheno %>%   
 split(., list(.$pheno)) %>%   
 purrr::imap(~ filter(.data = ., value != 'not\_assessed'))  
  
tpheno <- all\_sub\_pheno %>%   
 imap(function(.x, .y){  
 select(.data = .x, value) %>%   
 t() %>% write.table(str\_glue('../data/amplicon/lefse/pull\_{.y}.txt'), sep = '\t', quote = F, row.names = T, col.names = F)  
 })  
  
tcts <- all\_sub\_pheno %>%   
 map(~ pull(.data = ., sampleid) ) %>%   
 imap(~ cts\_fil %>% select(taxa\_asv, matches(.x)) %>% write\_tsv(str\_glue('../data/amplicon/lefse/{.y}\_asv\_tcts.tsv')))

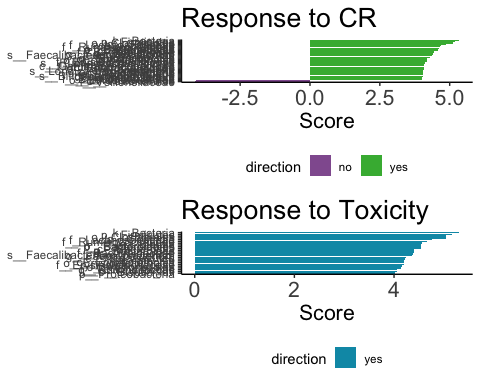
cat ../data/amplicon/lefse/pull\_toxicity.txt ../data/amplicon/lefse/toxicity\_asv\_tcts.tsv > ../data/amplicon/lefse/pull\_toxicity\_asv\_tcts.tsv  
cat ../data/amplicon/lefse/pull\_cr\_d100.txt ../data/amplicon/lefse/cr\_d100\_asv\_tcts.tsv > ../data/amplicon/lefse/pull\_cr\_d100\_asv\_tcts.tsv

fns <- list.files('../data/amplicon/lefse/', pattern = 'pull.\*\_asv\_tcts.tsv$')  
  
cmds <- tibble(  
 fns = fns  
) %>%   
 mutate(format\_cmd = str\_glue('format\_input.py {fns} {fns}.in -c 1 -u 2 -o 1000000 ')) %>%   
 mutate(run\_cmd = str\_glue('run\_lefse.py -l 4 {fns}.in {fns}.res')) %>%   
 mutate(plot\_cmd = str\_glue('plot\_res.py {fns}.res {fns}.pdf --format pdf --feature\_font\_size 4 --width 10 --dpi 300 --title {fns}')) %>%   
 mutate(clado\_cmd = str\_glue('plot\_cladogram.py {fns}.res {fns}\_clado.pdf --label\_font\_size 4 --dpi 300 --format pdf --title {fns}')) %>%   
 select(-fns) %>%   
 gather() %>%   
 select(value) %>%   
 write\_csv('../data/amplicon/lefse/lefse\_run\_cmd\_taxa.sh', col\_names = F)

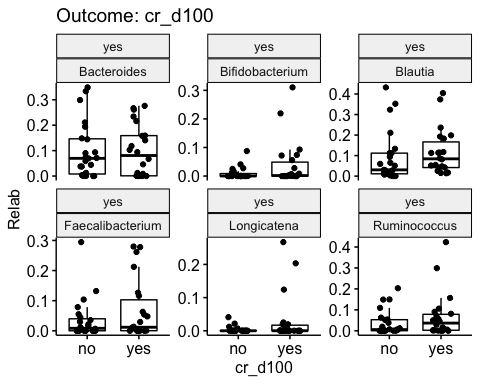
# run in terminal:  
# bash /Users/daia1/projects/CART\_microbiome/data/amplicon/lefse/lefse\_run\_cmd\_taxa.sh

# redo the resulted bar plot of the lefse in ggplot  
# the input : lefse res files  
fns <- list.files('../data/amplicon/lefse/', pattern = 'pull.\*\_asv\_tcts.tsv.res$', full.names = T)  
  
# join all of the tables feature together  
feature <- fns %>%   
 set\_names(fns) %>%   
 map(~ read\_tsv(., col\_names = c('feature','xx','direction','score','pval')) %>%   
 filter(!is.na(score))) %>%   
 bind\_rows(.id = 'group') %>%   
 mutate(group = str\_replace(group, '../data/amplicon/lefse//pull\_','')) %>%   
 mutate(group = str\_replace(group, '\_asv\_tcts.tsv.res$',''))   
  
# change the "N" direction to be minus score  
feature <- bind\_rows(  
 feature %>%   
 split(.$direction) %>%   
 pluck('no') %>%   
 mutate(score = -score),  
 feature %>%   
 split(.$direction) %>%   
 pluck('yes')   
) %>%   
 arrange(group, feature, score) %>%   
 mutate(feature = str\_replace\_all(feature, '^.+\\.', ''))

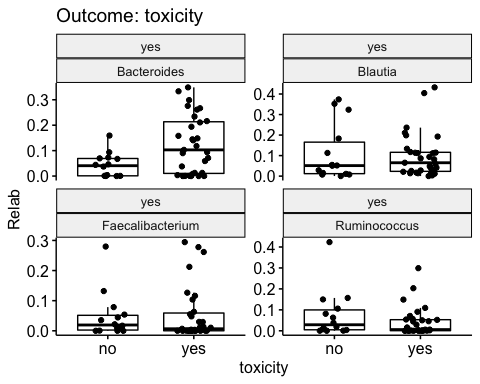
all\_title\_fs <- 20  
axis\_text\_fs <- 16  
  
CR <- feature %>%   
 filter(group == 'cr\_d100') %>%   
 ggplot(aes(x = reorder(feature, score), y = score, fill = direction)) +  
 geom\_bar( stat = 'identity') +  
 coord\_flip() +  
 scale\_color\_manual(values = c('#925E9F', '#42B540')) +  
 scale\_fill\_manual(values = c('#925E9F', '#42B540')) +  
 theme\_classic() +  
 theme(axis.title.y = element\_blank(),   
 plot.title = element\_text(size=all\_title\_fs),  
 axis.title.x = element\_text(size=axis\_text\_fs),  
 axis.text.x = element\_text(size=axis\_text\_fs),  
 legend.position='bottom') +  
 labs(title = str\_glue('Response to CR') ,  
 y = 'Score')  
  
tox <- feature %>%   
 filter(group == 'toxicity') %>%   
 ggplot(aes(x = reorder(feature, score), y = score, fill = direction)) +  
 geom\_bar(stat = 'identity') +  
 coord\_flip() +  
 scale\_color\_manual(values = c('#0099B4', '#AD002A')) +  
 scale\_fill\_manual(values = c('#0099B4', '#AD002A')) +  
 theme\_classic() +  
 theme(axis.title.y = element\_blank(),   
 axis.title.x = element\_text(size=axis\_text\_fs),  
 plot.title = element\_text(size=all\_title\_fs),  
 axis.text.x = element\_text(size=axis\_text\_fs),  
 legend.position="bottom") +  
 labs(title = str\_glue('Response to Toxicity') ,  
 y = 'Score')  
  
g <- plot\_grid(CR,tox,   
 nrow = 2,   
 align = 'hv',  
 axis = 'b') +  
 ggsave('../figs/amplicon/16s\_lefse\_combined.pdf', device = 'pdf', height = 15, width = 15)  
  
g



# to see the relative abundance of those taxa   
# to get the top and bottom three taxa of the lefse results  
res <- list.files('../data/amplicon/lefse/', pattern = 'asv\_tcts.tsv.res$', full.names = T)  
  
# gather the species level taxa in the lefse significant results  
res\_all <- res %>%   
 set\_names(res) %>%   
 map(~ read\_tsv(., col\_names = c('feature','xx','direction','score','pval')) %>%   
 filter(!is.na(score))) %>%   
 keep(~ nrow(.) > 0) %>%   
 bind\_rows(.id = 'res') %>%   
 mutate(res = str\_replace(res, '^.+//',''),  
 res = str\_replace(res, '\_asv.+$','')) %>%   
 rename(grp = res) %>%   
 filter(grp %in% c('pull\_cr\_d100','pull\_toxicity')) %>%   
 mutate(feature = str\_replace\_all(feature, '\\.','\\|')) %>%   
 #split(., list(.$grp, .$direction)) %>%   
 #map\_dfr(~ top\_n(x = ., n = 4, wt = score) %>% arrange(-score)) %>%   
 # filter(str\_detect(feature, 's\_\_.+$')) %>%   
 # filter(!str\_detect(feature, 'a\_\_.+$')) %>%   
 filter(str\_detect(feature, 'g\_\_.+$')) %>%   
 filter(!str\_detect(feature, 's\_\_.+$')) %>%   
 mutate(feature = str\_replace(feature, '^.+g\_\_','')) %>%   
 mutate(feature = str\_replace(feature, '\_Clostridium\_', '[Clostridium]')) %>%   
 ungroup()  
  
  
# plot the relab of those taxa (at species level) in boxplot  
# get the species counts of the sampels  
cts\_spp <- cts\_ %>%   
 full\_join(annot %>% select(asv\_key, species), by = 'asv\_key') %>%   
 select(-asv\_key) %>%   
 gather('sampleid', 'relab', names(.)[1]:names(.)[ncol(.)-1]) %>%   
 group\_by(sampleid, species) %>%   
 summarise(Relab = sum(relab)) %>%   
 select(sampleid, species, Relab) %>%   
 left\_join(meta %>% select(sampleid, cr\_d100, toxicity), by = 'sampleid') %>%   
 ungroup()  
  
cts\_genus <- cts\_ %>%   
 full\_join(annot %>% select(asv\_key, genus), by = 'asv\_key') %>%   
 select(-asv\_key) %>%   
 gather('sampleid', 'relab', names(.)[1]:names(.)[ncol(.)-1]) %>%   
 group\_by(sampleid, genus) %>%   
 summarise(Relab = sum(relab)) %>%   
 select(sampleid, genus, Relab) %>%   
 left\_join(meta %>% select(sampleid, cr\_d100, toxicity), by = 'sampleid') %>%   
 ungroup()  
  
joined <- cts\_genus %>%   
 inner\_join(res\_all, by = c('genus' = 'feature'))  
  
# finally I can do the plotting  
  
joined %>%  
 filter(grp == 'pull\_cr\_d100') %>%   
 ggboxplot(x = 'cr\_d100', y = 'Relab', add = 'jitter', title = 'Outcome: cr\_d100') +  
 facet\_wrap(direction ~ genus, scales="free\_y") +  
 ggsave('../figs/amplicon/lefse\_taxa\_crd100.pdf', width = 15, height = 13)



joined %>%  
 filter(grp == 'pull\_toxicity') %>%   
 ggboxplot(x = 'toxicity', y = 'Relab', add = 'jitter', title = 'Outcome: toxicity') +  
 facet\_wrap(direction ~ genus, scales="free\_y") +  
 ggsave('../figs/amplicon/lefse\_taxa\_toxicity.pdf', width = 10, height = 13)



* Thank you reivewer for making this point. Previously the abundance filtering threshold for the ASV was set at 0 and prevalence threshold at 10%. The abundance threshold was set at 0 since I assumed I could be more lenient as the LEfSe was done at ASV level. Thus very rare taxa showed up in the results. Per the reviewer’s comment, I’ve changed the abundance threshold to be at 0.01% and prevalence threshold at 25% (This is a small cohort with 45 samples).
* A lot of more commenly known taxa appeared in the updated results, and for display’s purpose, bar plot only showed taxa with LDA score above 4. Moreover, as represented in the boxplot, the significant taxa (at genus level) look quite abundant.
* A big portion of the results are the same in the two outcomes. It doesn’t quite help with making association conclusion, but it points us to the next step of selecting important genera for modeling.

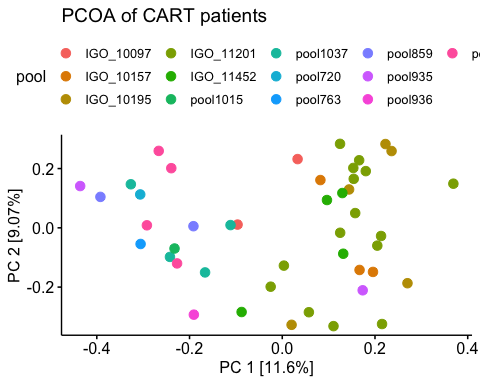
## Major point 5

**Were controls used in this study–positive controls with known microbial consortia and negative reagent controls in order to account for potential contamination and to normalize abundance across batches? This is particularly important because several of the taxa identified using LEfSe analsyis could have represented contamination, yet we do not know their relative abundance and whether they were present in small numbers of reads or not. Were samples processed in batches and were batch effects taken into account?**

get\_table\_from\_database("asv\_alpha\_diversity\_ag")

## [1] "table asv\_alpha\_diversity\_ag is loaded and filtered for duplicates. Only the replicate of highest coverage is retained."

# a pcoa at asv level for only M and P  
stb <- read\_csv('../data/amplicon/stool/combined\_2\_meta.csv')  
  
all\_pheno <- stb %>% select(sampleid, center) %>%   
 mutate(grp = 'CART') %>%   
 select(sampleid, grp, center) %>%   
 ungroup %>%   
 inner\_join(asv\_alpha\_diversity\_ag %>%   
 select(path\_pool, sampleid))  
  
cts <- get\_counts\_subset(c(stb$sampleid))  
  
nsamp <- nrow(stb)  
  
keepa <- cts %>%   
 filter(count\_relative > 0.0001) %>%   
 count(asv\_key) %>%   
 filter(n > floor(nsamp \* 0.25)) %>%   
 pull(asv\_key)  
  
cts\_fil <- cts %>%   
 filter(asv\_key %in% keepa) %>%   
 select(sampleid, asv\_key,count\_relative ) %>%   
 spread(key = 'asv\_key', value = 'count\_relative', fill = 0) %>%   
 column\_to\_rownames('sampleid')  
  
dist\_ <- vegdist(cts\_fil, method = 'bray')  
eigen <- pcoa(dist\_)$values$Eigenvalues  
percent\_var <- signif(eigen/sum(eigen), 3)\*100  
  
bc <- cmdscale(dist\_, k = 2)  
  
mp <- bc %>%  
 as.data.frame() %>%  
 rownames\_to\_column('sampleid') %>%   
 ungroup() %>%   
 inner\_join(all\_pheno) %>%   
 distinct(sampleid, .keep\_all = T) %>%   
 mutate(pool = str\_extract(path\_pool, 'Sample.+/')) %>%   
 mutate(pool = str\_replace(pool, 'Sample\_','')) %>%   
 mutate(pool = if\_else(str\_detect(pool, 'IGO'), str\_extract(pool, 'IGO.+$'), pool)) %>%   
 mutate(pool = str\_replace(pool, '\_1/|\_comple.+$',''))  
  
mp %>%   
 ggscatter(x = 'V1', y = 'V2', color = 'pool', size = 3) +  
 labs(title = 'PCOA of CART patients') +  
 xlab(paste0("PC 1 [",percent\_var[1],"%]")) +  
 ylab(paste0("PC 2 [",percent\_var[2],"%]")) +  
 #theme\_void() +  
 ggsave('../figs/PCOA(bray-curtis) (ASV level)of CART patients\_pool.pdf', width = 9, height = 9)

 Controls were not used in the study. Several important taxa identified by LefSe could be statistical artefact due to the lenient filtering threshold. In terms of batch effects, as shown in the above scatter plot, 45 samples came from 13 different sequencing batches, and they look well mixed. Therefore I would conclude that batch effects wouldn’t be a concern. Plus, in the following modeling step, center from which the sample was collected is incorporated as a covariet.

## Major point 13

**I would prefer to see an MDS/PCOA visualization instead of Figure 2C boxplots for beta-diversity. This helps much more with understanding how different communities are and what variability exists within communities**

library(vdbR)  
connect\_database('~/dbConfig.txt')  
get\_table\_from\_database("healthy\_volunteers\_ag")  
get\_table\_from\_database("asv\_alpha\_diversity\_ag")

## [1] "table asv\_alpha\_diversity\_ag is loaded and filtered for duplicates. Only the replicate of highest coverage is retained."

stb <- read\_csv('../data/amplicon/stool/combined\_2\_meta.csv')  
  
healthy <- healthy\_volunteers\_ag %>%   
 inner\_join(asv\_alpha\_diversity\_ag, by = c("sampleid", "oligos\_id"))   
cts <- get\_counts\_subset(c(stb$sampleid, healthy %>% pull(sampleid)))  
  
# a total of 75 samples   
nsamp <- cts %>%   
 distinct(sampleid) %>%   
 nrow  
  
all\_pheno <- bind\_rows(healthy %>%   
 select(sampleid) %>%   
 mutate(grp = 'healthy', center = 'healthy'),  
 stb %>% select(sampleid, center) %>%   
 mutate(grp = 'CART') %>%   
 select(sampleid, grp, center)  
 ) %>%   
 ungroup %>%   
 inner\_join(asv\_alpha\_diversity\_ag %>%   
 select(path\_pool, sampleid))  
  
  
# filter >0.01% in more than 25% samples  
keepa <- cts %>%   
 filter(count\_relative > 0.0001) %>%   
 count(asv\_key) %>%   
 filter(n > floor(nsamp \* 0.25)) %>%   
 pull(asv\_key)  
  
cts\_fil <- cts %>%   
 filter(asv\_key %in% keepa) %>%   
 select(sampleid, asv\_key,count\_relative ) %>%   
 spread(key = 'asv\_key', value = 'count\_relative', fill = 0) %>%   
 column\_to\_rownames('sampleid')  
  
library(vegan)  
dist\_ <- vegdist(cts\_fil, method = 'bray')  
eigen <- pcoa(dist\_)$values$Eigenvalues  
percent\_var <- signif(eigen/sum(eigen), 3)\*100  
  
bc <- cmdscale(dist\_, k = 2)  
  
asv\_pcoa <- bc %>%  
 as.data.frame() %>%  
 rownames\_to\_column('sampleid') %>%   
 ungroup() %>%   
 inner\_join(all\_pheno) %>%   
 distinct() %>%   
 ggscatter(x = 'V1', y = 'V2', color = 'grp') +  
 labs(title = 'PCOA of healthy and CART patients at ASV level') +  
 xlab(paste0("PC 1 [",percent\_var[1],"%]")) +  
 ylab(paste0("PC 2 [",percent\_var[2],"%]")) +  
 #theme\_void() +  
 ggsave('../figs/PCOA(bray-curtis) of healthy and CART patients\_asv\_level.pdf')

We agree a PCOA is a better visualization of the community difference between the healthy and the CART cohort. The PCOA is done at ASV level with same filtering threshold as above in the LEfSe and the Bray-Curtis distance was used. As illustrated in the plot, the CART and healthy cohort’s microbiome is quite different.

## Major point 14

**Why was a Bayesian statistical approach used? I would like to either see “standard” statistical approaches or have a strong justification for why this is more appropriate and the other methods are not**

* Using Bayesian approach, we are able to obtain the posterior distribution of the coefficients, which retains the uncertainty of the coefficient estimates, and also do posterior predictive check to quantify the impact of the coefficients (as shown in the Fig 2H & J). The odds ratio in logistic regression is hard to interpretate.

## Major point 15

**Why was Ruminococcus chosen but not the Clostridium or Fusobacterium genera chosen for analysis in association with outcomes? The results seem disjointed and do not create a story to explain why the prior results are being ignored here. Why was Ruminococcus not observed in LEfSe for example?**

As mentioned in the above reply, rare taxa’s apprearance such as Fusobacterium may just be statistical artefact due to the lenient filtering threshold. When the thresholds have been updated, Ruminococcus does show up and turn out to be quite significant with LDA score above 4 with taxa level spanning from family to species.

## Major point 17

**Please indicate when 16S sequencing results were used in analyses versus when shotgun sequencing was used. It is unclear why 16S was used for taxonomic results when shotgun sequencing was available for all of the samples as well. In addition, please include the median read sizes for each and the range of sizes. More data about the sequencing reads are needed.**

* 16s sequencing data was used in looking at the compositions, alpha, beta diversity, LefSe analysis and the Bayesian modeling, while shotgun data was used in LefSe analysis with the taxa and pathway abundance data.
* During the progession of the project, the 16s data was obtained sooner, which was why we did the major analysis with 16s data.

# for the read count of the 16s samples after preprocessing  
cts <- get\_counts\_subset(c(stb$sampleid))  
  
sample\_counts <- cts %>%   
 distinct(sampleid, count\_total)  
  
summary(sample\_counts$count\_total)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 7041 36424 50788 51121 65506 93953

Thank reviewer for reminding us to add more details of the sequencing reads. For 16s data, the median read size is 50788 and the range is from 7041 to 93953. For the shotgun data,

## Minor point 17

**Figure 2: Are Ruminococcus/Bacteroides “low” and “high” relative abundance determined according to the median abundance? Or some other way?**

# Reviewer 2

## Major point 1

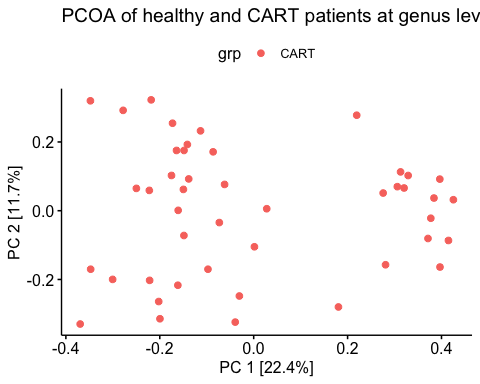
**Code availability statement does list names of software packages that have been used but does not provide any information on the code that implements the analyses included in this manuscript. It also seems unlikely that the actual code would have used as few packages than what has been mentioned in the Methods. The source code for the actual analyses of this manuscript has not been provided despite generally mentioning some external open source algorithms and software by name. The standard way of providing reproducible analysis source code is either as a supplementary zip file or as an online repository (such as Github), associated with permanent DOI (e.g. through Zenodo). Without the code it is not possible to assess the workflow details or assess the robustness of the workflows to varying methodological choices. This is concern in particular due to the other questions related to methodology (see below).**

Thank reviewer for bringing up this point. We’ve created a github page to display the code and the resulted figures. The address is: <https://vdblab.github.io/CART_and_microbiome/>

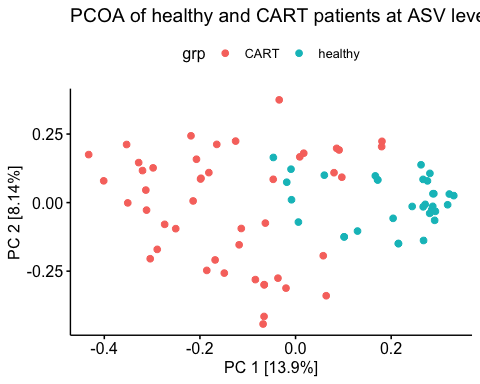
## Major point 2

**520-521: beta diversity has been calculated at genus level; the beta diversity is often calculated using the finest (ASV) resolution in order to use maximal information in the data, and in many cases this can have remarkable impact on the results. Are the results and conclusions robust to this choice?**

# beta diversity at genus level but a PCOA between healthy and the CART  
joined <- cts %>%   
 left\_join(asv\_annotation\_blast\_ag %>% select(asv\_key, genus)) %>%   
 group\_by(sampleid, genus) %>%   
 summarise(relab = sum(count\_relative)) %>%   
 ungroup()  
  
keepg <- joined %>%   
 filter(relab > 0.0001) %>%   
 count(genus) %>%   
 filter(n > floor(nsamp \* 0.25)) %>%   
 filter(!is.na(genus)) %>%   
 pull(genus)   
  
cts\_fil <- joined %>%   
 filter(genus %in% keepg) %>%   
 spread('genus', 'relab', fill = 0) %>%   
 column\_to\_rownames('sampleid')  
  
library(vegan)  
dist\_ <- vegdist(cts\_fil, method = 'bray')  
eigen <- pcoa(dist\_)$values$Eigenvalues  
percent\_var <- signif(eigen/sum(eigen), 3)\*100  
bc <- cmdscale(dist\_, k = 2)  
bc %>%  
 as.data.frame() %>%  
 rownames\_to\_column('sampleid') %>%   
 ungroup() %>%   
 inner\_join(all\_pheno) %>%   
 distinct(sampleid, .keep\_all = T) %>%   
 ggscatter(x = 'V1', y = 'V2', color = 'grp') +  
 labs(title = 'PCOA of healthy and CART patients at genus level') +  
 xlab(paste0("PC 1 [",percent\_var[1],"%]")) +  
 ylab(paste0("PC 2 [",percent\_var[2],"%]")) +  
 #theme\_void() +  
 ggsave('../figs/PCOA(bray-curtis) of healthy and CART patients\_genus\_level.pdf')



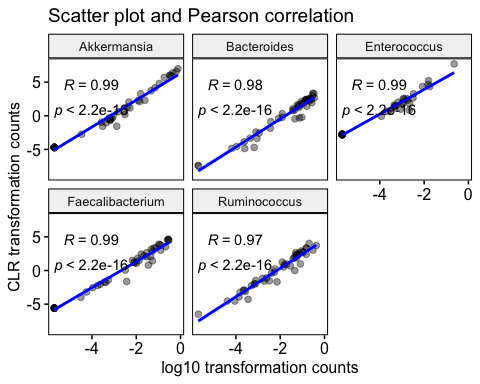
asv\_pcoa

 - Thank reviewer for bringing up this point. As shown by the above two PCOA, one at genus level, one at ASV level, with same filtering threshold, the healthy and the CART cohort look about the same level of separation. The result is robust to this choice.

## Major point 3

**l. 580-581 log relative abundances have been use to evaluate associations; this ignores the compositionality effect, which is commonly treated by using Aitchison transformations (e.g. CLR) in microbiome studies. Ignoring compositionality has potentially notable impact on the analysis outcomes. It would be essential to check that the overall conclusions are not affected by this. Using clr transformed abundances would be more in line with the current practices than using log relative abundance.**

# compare between the transformed counts of CLR and log10  
meta <- read\_csv('../data/amplicon/stool/combined\_2\_meta.csv')  
  
library(vdbR)  
connect\_database('~/dbConfig.txt')  
cts <- get\_counts\_subset(meta$sampleid)  
cts\_ <- cts %>%   
 select(asv\_key, sampleid, count)  
  
annot <- asv\_annotation\_blast\_ag %>%   
 filter(asv\_key %in% cts\_$asv\_key)  
  
# CLR transformation works at the counts level  
cts\_all <- cts\_ %>%   
 left\_join(annot %>% select(asv\_key, genus), by = 'asv\_key') %>%   
 group\_by(sampleid, genus) %>%   
 summarise(cnt = sum(count)) %>%   
 spread('sampleid', 'cnt', fill = 0) %>%   
 filter(!is.na(genus)) %>%   
 column\_to\_rownames('genus')  
  
clr\_res <- compositions::clr(cts\_all + 0.5) %>%   
 as.data.frame() %>%   
 rownames\_to\_column('genus') %>%   
 gather('sampleid','clr', names(.)[2]:names(.)[ncol(.)])  
  
clr\_5 <- clr\_res %>%   
 filter(genus %in% c('Faecalibacterium','Ruminococcus','Akkermansia','Bacteroides','Enterococcus'))   
  
# my log10 tranformation results  
log10 <- read\_csv('../data/amplicon/stool/combined\_5\_genera.csv') %>%   
 gather('genus', 'log10', Akkermansia:Ruminococcus)  
  
# compare the two  
two <- log10 %>%   
 inner\_join(clr\_5)  
  
two %>%   
 arrange(log10) %>%   
 #filter(log10 > -5) %>%   
 ggscatter(x = 'log10', y = 'clr', facet.by = 'genus', alpha = 0.4,  
 xlab = 'log10 transformation counts', ylab = 'CLR transformation counts',  
 title = 'Scatter plot and Pearson correlation' ,  
 add = "reg.line", # Add regressin line  
 add.params = list(color = "blue", fill = "lightgray"), # Customize line  
 conf.int = TRUE, # Add confidence interval  
 cor.coef = TRUE, # Add correlation coefficient.  
 cor.coeff.args = list(method = "pearson", label.sep = "\n") )



* Thank reviewer for pointing this out. We visualized the Pearson correlation between the CLR and the log10 transformed counts for the 5 genera that we put into the model. As depicted in the above plot, the correlation is about 1 for all of them. Therefore the log10 transformation is fine with out purpose.

## Minor point 1

**The Bayesian analysis is interesting and it is good to see that this methodology is starting to be adopted also in this research area but the analysis does not mention covariates. It the assumption that the groups are sufficiently homogeneous so that no covariates need to be considered in the analysis?**