Test with genus abundance data

Weijia Xiong

5/14/2020

Real data

From https://github.com/chvlyl/PLEASE

Inflammation, Antibiotics, and Diet as Environmental Stressors of the Gut Microbiome in Pediatric Crohn's Disease. Cell Host & Microbe. 2015

Load the raw data

The "unclassfied" taxa were removed and the total relative abundance in each sample were normalized to be one. "P","F","G","S" at the beginning of each file indicate taxonomic levels "phylum", "family", "genus", "species".

We use genus abundance data.

```
PLEASE.raw =
  read_excel("./real data/G_Remove_unclassfied_Renormalized_Merge_Rel_MetaPhlAn_Result.xlsx",
             col_names = T) %>%
  as.data.frame()
row.names(PLEASE.raw) = PLEASE.raw[,1]
PLEASE.raw = PLEASE.raw[,-1]
colname = as.numeric(colnames(PLEASE.raw))
samplepoint = as.Date(colname,origin = "1900-01-01")
head(PLEASE.raw[,1:5])
```

```
##                         1132619    1132650      1132678      1132709    1132984
## g__Bacteroides        0.4272231  0.2318847  0.020411137  0.005330001 72.3829938
## g__Ruminococcus       0.0000000  0.0000000  0.006130341  0.000000000  4.9631993
## g__Faecalibacterium   0.0000000  0.0000000  0.000000000  0.000000000  8.4710030
## g__Akkermansia        0.0000000  0.0000000  0.000000000  0.000000000  0.0000000
## g__Bifidobacterium    0.3756509  2.0725290  1.033117545  0.602090120  1.3608252
## g__Escherichia       38.3172402 21.4206238 41.259008130 36.090417220  0.8564302
```

```
taxa.raw <- data.frame(t(PLEASE.raw),
                       row.names = strtrim(samplepoint,7))

### Make sure you load the data correctly
cat('samples','taxa',dim(taxa.raw),'\n')
```

```
## samples taxa 335 105
```

```
taxa.raw[1:3,1:3]
```

```
##         g__Bacteroides g__Ruminococcus g__Faecalibacterium
## 5001-01     0.42722310     0.000000000                   0
## 5001-02     0.23188470     0.000000000                   0
## 5001-03     0.02041114     0.006130341                   0
```

Each row represent a sample.

Load total non-human read counts

```
human.read.file <- './real data/please_combo_human_reads.xlsx'
human.read <-
  read_excel(human.read.file, col_names = T) %>%
  as.data.frame() %>%
  mutate(
    Sample = strtrim(as.Date(Sample,origin = "1900-01-01"),7)
  )

head(human.read)
```

```
##    Sample NonHumanReads TotalReads HumanReads    HumanPer GroupFcp GroupPcdai
## 1 1910-12      17525422   19515697    1990275 10.19832959    Combo      Combo
## 2 1910-12      18089762   18185930      96168  0.52880444    Combo      Combo
## 3 1910-12      27311061   27338002      26941  0.09854781    Combo      Combo
## 4 1910-12      11051439   11092808      41369  0.37293536    Combo      Combo
## 5 1910-12       9434025    9492196      58171  0.61282980    Combo      Combo
## 6 1910-12      23327496   23634581     307085  1.29930382    Combo      Combo
```

```
## first column: sample id
```

```
### Filter low depth samples (low non human reads)
low.depth.samples <- subset(human.read,NonHumanReads<10000)
head(low.depth.samples[,1:5])
```

```
##      Sample NonHumanReads TotalReads HumanReads    HumanPer
## 56  5010-02          1014       1104         90  8.15217391
## 85  5018-03          6101       6121         20  0.32674400
## 98  5023-04          1954       2679        725 27.06233669
## 257 6007-01          1809       4249       2440 57.42527654
## 309 7001-02          9965       9971          6  0.06017451
## 310 7001-03           566        613         47  7.66721044
```

```
### Delete these samples from PLEASE data.
# row.names(taxa.raw)
# row.names(low.depth.samples)
row.names(taxa.raw)[which(row.names(taxa.raw) %in% low.depth.samples$Sample)]
```

```
## [1] "5018-03" "7001-02" "7003-02" "7003-03" "7009-03" "7010-03"
```

```
### Before deletion
dim(taxa.raw)
```

```
## [1] 335 105
```

```
### After deletion
taxa.raw <- taxa.raw[-which(rownames(taxa.raw) %in% low.depth.samples$Sample),]
dim(taxa.raw)
```

```
## [1] 329 105
```

```
### Filter low abundant bacterial data
filter.index1 <- apply(taxa.raw,2,function(X){sum(X>0)>0.4*length(X)})
filter.index2 <- apply(taxa.raw,2,function(X){quantile(X,0.9)>1})
```

```
taxa.filter <- taxa.raw[,filter.index1 & filter.index2]
taxa.filter <- 100*sweep(taxa.filter, 1, rowSums(taxa.filter), FUN="/")
cat('after filter:','samples','taxa',dim(taxa.filter),'\n')
```

```
## after filter: samples taxa 329 18
```

```
cat(colnames(taxa.filter),'\n')
```

```
## g__Bacteroides g__Ruminococcus g__Faecalibacterium g__Bifidobacterium g__Escherichia g__Clostridium g
```

```
head(rowSums(taxa.filter))
```

```
## 5001-01 5001-02 5001-03 5001-04 5002-01 5002-02
##     100     100     100     100     100     100
```

After filter, there remains 18 bacteria in the taxa table.

```
###
taxa.data <- taxa.filter
dim(taxa.data)
```

```
## [1] 329  18
```

Load sample information

```
head(sample.info)
```

```
##   Sample Subject Species.Cluster   Cluster Treatment FCPResponse  Type Time
## 1   4000    4000       cluster 1 cluster 1        NA          NA COMBO   NA
## 2   4001    4001       cluster 1 cluster 1        NA          NA COMBO   NA
## 3   4002    4002       cluster 1 cluster 1        NA          NA COMBO   NA
## 4   4004    4004       cluster 1 cluster 1        NA          NA COMBO   NA
## 5   4005    4005       cluster 1 cluster 1        NA          NA COMBO   NA
## 6   4006    4006       cluster 1 cluster 1        NA          NA COMBO   NA
##   BristolScore FCP PCDAI PUCAI log.FCP Group Response Antibiotics.visit
## 1           NA  NA    NA    NA      NA COMBO       NA           Not.Use
## 2           NA  NA    NA    NA      NA COMBO       NA           Not.Use
## 3           NA  NA    NA    NA      NA COMBO       NA           Not.Use
## 4           NA  NA    NA    NA      NA COMBO       NA           Not.Use
## 5           NA  NA    NA    NA      NA COMBO       NA           Not.Use
## 6           NA  NA    NA    NA      NA COMBO       NA           Not.Use
##   Steroids Treatment.Specific Disease NonHumanReads             Human.Per
## 1       NA                 NA Control      17525422           10.19832959
## 2       NA                 NA Control      18089762   0.52880444000000004
## 3       NA                 NA Control      27311061 9.8547805000000002E-2
## 4       NA                 NA Control      11051439   0.37293536300000002
## 5       NA                 NA Control       9434025   0.61282980399999998
## 6       NA                 NA Control      23327496           1.299303817
##               Fungi.Per            Distance           Bact.Div
## 1   1.02707940499236E-4   0.452380952380952 133.36064650458701
## 2 7.5180646378874396E-3 0.42857142857142899   173.757418771479
## 3 8.2750355249838195E-4 0.40476190476190499 90.479491742827904
## 4 3.5651465840783299E-3 0.35714285714285698   103.213875049946
## 5 1.5984693701786901E-2                 0.5 149.26361597807599
## 6 6.1986935931743403E-3 0.28571428571428598   112.488017364215
##      Species.Distance
## 1 0.40449438202247201
## 2 0.337078651685393
## 3 0.337078651685393
## 4   0.426966292134831
## 5 0.438202247191011
## 6   0.235955056179775
```

create covariates, Time, Treatment(antiTNF+EEN)

```
complete_subject =
  sample.info %>%
  filter(Sample %in% rownames(taxa.data)) %>%
  filter(Treatment.Specific!='PEN')%>%
  dplyr::select(Sample,Time,Subject,Response,Treatment.Specific) %>%
  group_by(Subject) %>%
  summarise(count = n()) %>%
  filter(count==4)

reg.cov =
  sample.info %>%
  filter(Subject %in%complete_subject$Subject) %>%
  mutate(Treat=ifelse(Treatment.Specific=='antiTNF',1,0)) %>%
  dplyr::mutate(Subject=paste('S',Subject,sep='')) %>%
  dplyr::mutate(Time=ifelse(Time=='1',0,ifelse(Time=='2',1,ifelse(Time=='3',4,ifelse(Time=='4',8,NA)))))
  dplyr::mutate(Time.X.Treatment=Time*Treat) %>%
  dplyr::select(Sample,Subject,Time,Response,Treat,Time.X.Treatment,everything())
```

take out first time point

```
reg.cov.t1   <-  subset(reg.cov,Time==0)
rownames(reg.cov.t1) <- reg.cov.t1$Subject

reg.cov.t234 <-  subset(reg.cov,Time!=0)
reg.cov.t234 <- data.frame(
  baseline.sample=reg.cov.t1[reg.cov.t234$Subject,'Sample'],
  baseline.subject=reg.cov.t1[reg.cov.t234$Subject,'Subject'],
  reg.cov.t234,
  stringsAsFactors = FALSE)

head(reg.cov.t234)
```

```
##   baseline.sample baseline.subject  Sample Subject Time     Response Treat
## 2         5001-01            S5001 5001-02   S5001    1 Non.Response     1
## 3         5001-01            S5001 5001-03   S5001    4 Non.Response     1
## 4         5001-01            S5001 5001-04   S5001    8 Non.Response     1
## 6         5002-01            S5002 5002-02   S5002    1 Non.Response     1
## 7         5002-01            S5002 5002-03   S5002    4 Non.Response     1
## 8         5002-01            S5002 5002-04   S5002    8 Non.Response     1
##   Time.X.Treatment Species.Cluster   Cluster Treatment FCPResponse      Type
## 2                1       cluster 2 cluster 2   antiTNF           0 PLEASE-T2
## 3                4       cluster 2 cluster 2   antiTNF           0 PLEASE-T3
## 4                8       cluster 2 cluster 2   antiTNF           0 PLEASE-T4
## 6                1       cluster 2 cluster 1   antiTNF           0 PLEASE-T2
## 7                4       cluster 2 cluster 1   antiTNF           0 PLEASE-T3
## 8                8       cluster 2 cluster 1   antiTNF           0 PLEASE-T4
##   BristolScore  FCP PCDAI PUCAI            log.FCP  Group Antibiotics.visit
## 2            6  607    NA    25 6.4085287910595001 PLEASE           Not.Use
## 3            6  867    NA    20 6.7650389767805397 PLEASE           Not.Use
## 4            6  557     5    15 6.3225652399272798 PLEASE           Not.Use
## 6            6  950    NA    10 6.8564619845945902 PLEASE           Not.Use
## 7            6 1947    NA    50 7.5740450053722004 PLEASE           Not.Use
## 8            6 1880    35    40 7.5390270558239996 PLEASE           Not.Use
##   Steroids Treatment.Specific Disease NonHumanReads           Human.Per
## 2  Not.Use            antiTNF   Crohn       1350309         89.31803171
## 3  Not.Use            antiTNF   Crohn      10946591  19.689170000000001
## 4  Not.Use            antiTNF   Crohn      14230882 0.85133673399999998
## 6      Use            antiTNF   Crohn      12020377         17.08929796
## 7      Use            antiTNF   Crohn       1910666  88.544540839999996
## 8      Use            antiTNF   Crohn        606565          89.5498726
##               Fungi.Per            Distance           Bact.Div
## 2 7.0946724046125703E-2 0.69047619047619002 79.225334004595695
## 3 1.2168171808008501E-2 0.52272727272727304 66.915876889694502
## 4      1.26499538117174 0.59090909090909105 53.865413747151202
## 6   1.05154771767974E-2 0.42857142857142899 81.330542193164007
## 7 6.6678320543726605E-2 0.59523809523809501 89.862102084722594
## 8 5.6712800771557902E-2 0.66666666666666696 61.502647827475897
##      Species.Distance
## 2 0.82417582417582402
## 3 0.72527472527472503
## 4 0.73333333333333295
## 6 0.59550561797752799
## 7 0.75280898876404501
## 8 0.85393258426966301
```

```
taxa_all = colnames(taxa.data)
store = function(taxa){
# X: Baseline abundance time Treat
# Y: Response abundance at time 1 4 8
X <- data.frame(
      Baseline=taxa.data[reg.cov.t234$baseline.sample,taxa]/100,
      reg.cov.t234[,c('Time','Treat')])

rownames(X) <- reg.cov.t234$Sample
Y <- taxa.data[reg.cov.t234$Sample, taxa]/100

return(list(X = X,
            Y = Y))
}


store_results = lapply(taxa_all, store)

#example
store_results[[1]]$X
```

```
##             Baseline Time Treat
## 5001-02 0.0045146078    1     1
## 5001-03 0.0045146078    4     1
## 5001-04 0.0045146078    8     1
## 5002-02 0.7326455546    1     1
## 5002-03 0.7326455546    4     1
## 5002-04 0.7326455546    8     1
## 5003-02 0.2196251863    1     1
## 5003-03 0.2196251863    4     1
## 5003-04 0.2196251863    8     1
## 5006-02 0.8708652339    1     1
## 5006-03 0.8708652339    4     1
## 5006-04 0.8708652339    8     1
## 5007-02 0.4205434668    1     1
## 5007-03 0.4205434668    4     1
## 5007-04 0.4205434668    8     1
## 5015-02 0.6195966402    1     1
## 5015-03 0.6195966402    4     1
## 5015-04 0.6195966402    8     1
## 5016-02 0.0253945370    1     1
## 5016-03 0.0253945370    4     1
## 5016-04 0.0253945370    8     1
## 5022-02 0.2201262889    1     1
## 5022-03 0.2201262889    4     1
## 5022-04 0.2201262889    8     1
## 5029-02 0.0010523304    1     1
## 5029-03 0.0010523304    4     1
## 5029-04 0.0010523304    8     1
## 5030-02 0.2872609011    1     1
## 5030-03 0.2872609011    4     1
## 5030-04 0.2872609011    8     1
## 5031-02 0.0046898528    1     1
## 5031-03 0.0046898528    4     1
## 5031-04 0.0046898528    8     1
## 5032-02 0.5737784224    1     1
## 5032-03 0.5737784224    4     1
## 5032-04 0.5737784224    8     1
## 5033-02 0.4050176093    1     1
## 5033-03 0.4050176093    4     1
## 5033-04 0.4050176093    8     1
## 5034-02 0.6607590956    1     1
## 5034-03 0.6607590956    4     1
## 5034-04 0.6607590956    8     1
## 5035-02 0.4923311638    1     1
## 5035-03 0.4923311638    4     1
## 5035-04 0.4923311638    8     1
## 5040-02 0.9279761367    1     1
## 5040-03 0.9279761367    4     1
## 5040-04 0.9279761367    8     1
## 5041-02 0.2294878766    1     1
## 5041-03 0.2294878766    4     1
## 5041-04 0.2294878766    8     1
## 5042-02 0.4001408559    1     1
## 5042-03 0.4001408559    4     1
## 5042-04 0.4001408559    8     1
## 5044-02 0.1346570281    1     1
## 5044-03 0.1346570281    4     1
## 5044-04 0.1346570281    8     1
## 5045-02 0.9496513557    1     1
## 5045-03 0.9496513557    4     1
## 5045-04 0.9496513557    8     1
## 5046-02 0.2768229474    1     1
## 5046-03 0.2768229474    4     1
## 5046-04 0.2768229474    8     1
## 5047-02 0.0010511366    1     1
## 5047-03 0.0010511366    4     1
## 5047-04 0.0010511366    8     1
## 5048-02 0.7338315341    1     1
## 5048-03 0.7338315341    4     1
## 5048-04 0.7338315341    8     1
## 5049-02 0.1426088573    1     1
## 5049-03 0.1426088573    4     1
## 5049-04 0.1426088573    8     1
## 5050-02 0.9763345218    1     1
## 5050-03 0.9763345218    4     1
## 5050-04 0.9763345218    8     1
## 5052-02 0.2543940843    1     1
## 5052-03 0.2543940843    4     1
## 5052-04 0.2543940843    8     1
## 5053-02 0.0009417628    1     1
## 5053-03 0.0009417628    4     1
## 5053-04 0.0009417628    8     1
## 5054-02 0.0400640076    1     1
## 5054-03 0.0400640076    4     1
## 5054-04 0.0400640076    8     1
## 5055-02 0.0249072545    1     1
## 5055-03 0.0249072545    4     1
## 5055-04 0.0249072545    8     1
## 5056-02 0.6123285395    1     1
## 5056-03 0.6123285395    4     1
## 5056-04 0.6123285395    8     1
## 5057-02 0.0078523362    1     1
## 5057-03 0.0078523362    4     1
## 5057-04 0.0078523362    8     1
## 5058-02 0.1940085400    1     1
## 5058-03 0.1940085400    4     1
## 5058-04 0.1940085400    8     1
## 5060-02 0.0087928699    1     1
## 5060-03 0.0087928699    4     1
## 5060-04 0.0087928699    8     1
## 5062-02 0.0210325731    1     1
## 5062-03 0.0210325731    4     1
## 5062-04 0.0210325731    8     1
## 5064-02 0.6558829728    1     1
## 5064-03 0.6558829728    4     1
## 5064-04 0.6558829728    8     1
## 5065-02 0.7909470475    1     1
## 5065-03 0.7909470475    4     1
## 5065-04 0.7909470475    8     1
## 6002-02 0.0546552210    1     0
## 6002-03 0.0546552210    4     0
## 6002-04 0.0546552210    8     0
## 6003-02 0.0077598908    1     1
## 6003-03 0.0077598908    4     1
## 6003-04 0.0077598908    8     1
## 6005-02 0.8964491372    1     0
## 6005-03 0.8964491372    4     0
## 6005-04 0.8964491372    8     0
## 6006-02 0.8389452223    1     1
## 6006-03 0.8389452223    4     1
## 6006-04 0.8389452223    8     1
## 6008-02 0.0003799075    1     1
## 6008-03 0.0003799075    4     1
## 6008-04 0.0003799075    8     1
## 6010-02 0.6516770298    1     0
## 6010-03 0.6516770298    4     0
## 6010-04 0.6516770298    8     0
## 6011-02 0.1111924396    1     1
## 6011-03 0.1111924396    4     1
## 6011-04 0.1111924396    8     1
## 6012-02 0.0043403305    1     1
## 6012-03 0.0043403305    4     1
## 6012-04 0.0043403305    8     1
## 6013-02 0.6665759874    1     1
## 6013-03 0.6665759874    4     1
## 6013-04 0.6665759874    8     1
## 6014-02 0.0079170891    1     1
## 6014-03 0.0079170891    4     1
## 6014-04 0.0079170891    8     1
## 6015-02 0.1886325002    1     1
## 6015-03 0.1886325002    4     1
## 6015-04 0.1886325002    8     1
## 6016-02 0.0661688600    1     1
## 6016-03 0.0661688600    4     1
## 6016-04 0.0661688600    8     1
## 6017-02 0.2546957745    1     0
## 6017-03 0.2546957745    4     0
## 6017-04 0.2546957745    8     0
## 6018-02 0.0026342805    1     1
## 6018-03 0.0026342805    4     1
## 6018-04 0.0026342805    8     1
## 6019-02 0.3589884116    1     0
## 6019-03 0.3589884116    4     0
## 6019-04 0.3589884116    8     0
## 7004-02 0.3446068511    1     0
## 7004-03 0.3446068511    4     0
## 7004-04 0.3446068511    8     0
## 7005-02 0.3139368765    1     0
## 7005-03 0.3139368765    4     0
## 7005-04 0.3139368765    8     0
## 7006-02 0.5030187519    1     0
## 7006-03 0.5030187519    4     0
## 7006-04 0.5030187519    8     0
## 7007-02 0.0856075635    1     1
## 7007-03 0.0856075635    4     1
## 7007-04 0.0856075635    8     1
## 7008-02 0.6327009155    1     0
## 7008-03 0.6327009155    4     0
## 7008-04 0.6327009155    8     0
## 7011-02 0.7997379659    1     0
## 7011-03 0.7997379659    4     0
## 7011-04 0.7997379659    8     0
## 7013-02 0.8571138921    1     0
## 7013-03 0.8571138921    4     0
## 7013-04 0.8571138921    8     0
## 7015-02 0.1428729070    1     0
## 7015-03 0.1428729070    4     0
## 7015-04 0.1428729070    8     0
```