talk06 练习与作业

目录

0.1 练习和作业说明

将相关代码填写入以 “‘{r} “‘ 标志的代码框中，运行并看到正确的结果；

完成后，用工具栏里的”Knit” 按键生成 PDF 文档；

将 PDF 文档改为：姓名-学号-`talk06` 作业`.pdf`，并提交到老师指定的平台/钉群。

0.2 Talk06 内容回顾

1. 3 个生信任务的 R 解决方案
2. factors 的更多应用 (forcats)
3. pipe

0.3 练习与作业：用户验证

请运行以下命令，验证你的用户名。

如你当前用户名不能体现你的真实姓名，请改为拼音后再运行本作业！

```
Sys.info()[["user"]]
```

```
## [1] "mingyuwang"
```

```
Sys.getenv("HOME")
```

```
## [1] "C:/Users/rhong/Documents"
```

0.4 练习与作业 1：作图

0.4.1 用下面的数据作图

1. 利用下面代码读取一个样本的宏基因组相对丰度数据

```
abu <-
  read_delim(
    file = "../data/talk06/relative_abundance_for_RUN_ERR1072629_taxonlevel_species.txt
    delim = "\t", quote = "", comment = "#")
```

2. 取前 5 个丰度最高的菌，将其它的相对丰度相加并归为一类 `Qita`；
3. 用得到的数据画如下的空心 pie chart:

```
library("tidyverse")
library("ggplot2")
library("forcats")
```

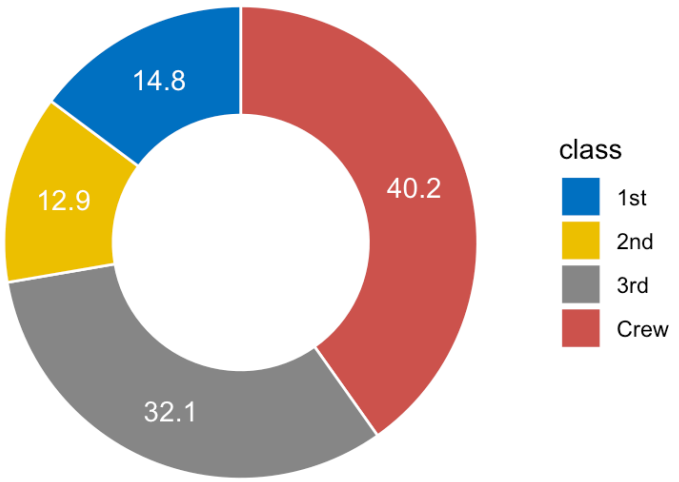


图 1: make a pie chart like this using the meteagenomics data

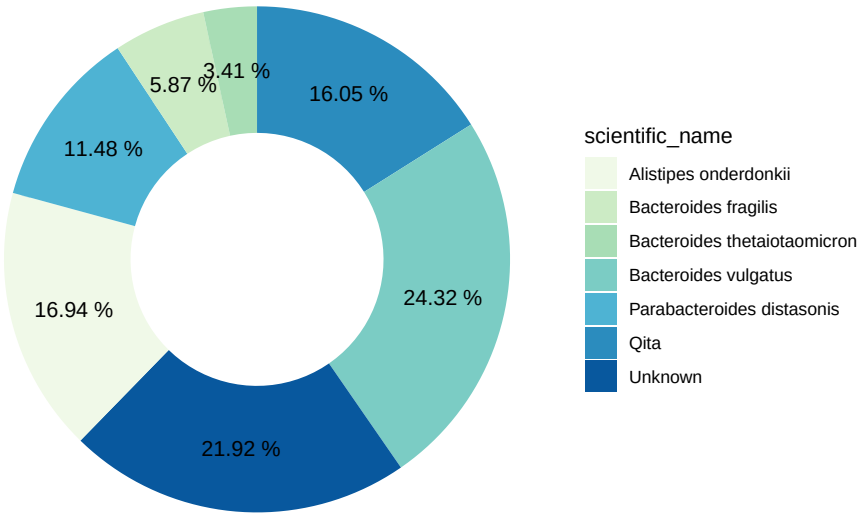
```
library("igraph")
library("reshape2")
library("RColorBrewer")
```

```
## 代码写这里，并运行；
abu <- read_delim(
    file = paste0("./data/talk06/relative_abundance_for_",
      "RUN_ERR1072629_taxonlevel_species.txt"),
    delim = "\t", quote = "", comment = "#", show_col_types = FALSE)
# 取前 5 个丰度最高的菌，其他的菌相对丰度相加并归为一类 Qita
(qita_abu <- abu %>%
  arrange(desc(relative_abundance)) %>%
  slice(7:n()) %>%
  summarise(relative_abundance = sum(relative_abundance)) %>%
  mutate(scientific_name = "Qita") %>%
  bind_rows(abu %>%
              arrange(desc(relative_abundance)) %>%
              slice(1:6)))
```

```
## # A tibble: 7 x 3
##   relative_abundance scientific_name              ncbi_taxon_id
##                <dbl> <chr>                                <dbl>
## 1               16.1  Qita                                  NA
## 2               24.3  Bacteroides vulgatus                 821
## 3               21.9  Unknown                               -1
## 4               16.9  Alistipes onderdonkii             328813
## 5               11.5  Parabacteroides distasonis           823
## 6                5.87 Bacteroides fragilis                 817
## 7                3.41 Bacteroides thetaiotaomicron         818
```

```
qita_abu$ymax <- cumsum(qita_abu$relative_abundance)
qita_abu$ymin <- c(0, head(qita_abu$ymax, n = -1))
qita_abu$labelPosition <- (qita_abu$ymax + qita_abu$ymin) / 2
```

```
# 画图，画出 donut chart
ggplot(qita_abu,
    aes(ymax = ymax, ymin = ymin, xmax = 4, xmin = 3, fill = scientific_name)) +
  geom_rect() +
  geom_text(x = 3.5, y = qita_abu$labelPosition,
    label = paste(round(qita_abu$relative_abundance, 2), "%", sep = " ")) +
  scale_fill_brewer(palette=4) +
  coord_polar(theta = "y") +
  xlim(c(2, 4)) +
  theme_void() +
  theme(legend.position = "right")
```

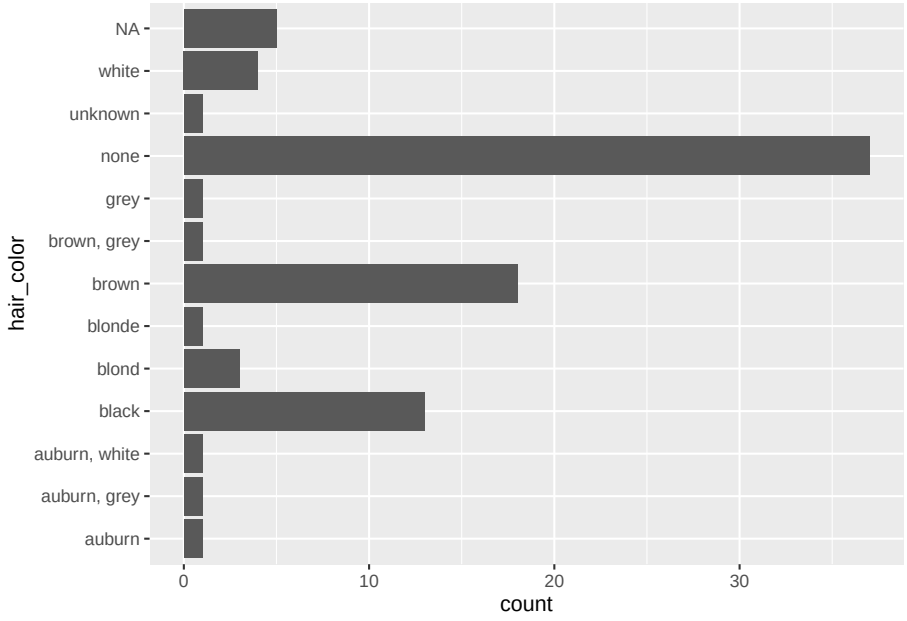


0.4.2 使用 starwars 变量做图

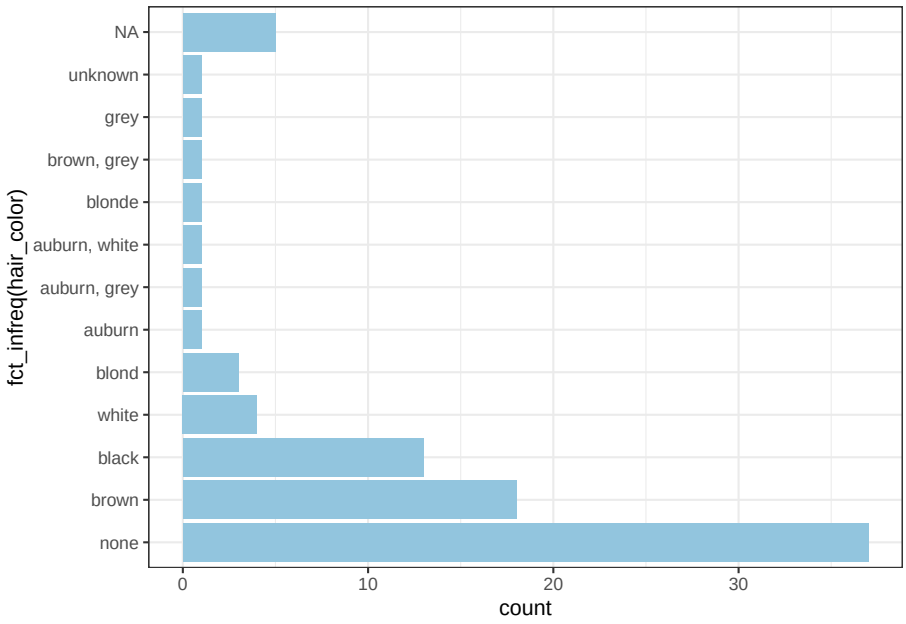
1. 统计 starwars 中 hair_color 的种类与人数时，可用下面的代码：

但是，怎么做到**按数量从小到大**排序？

```
ggplot(starwars, aes(x = hair_color)) +
  geom_bar() +
  coord_flip()
```



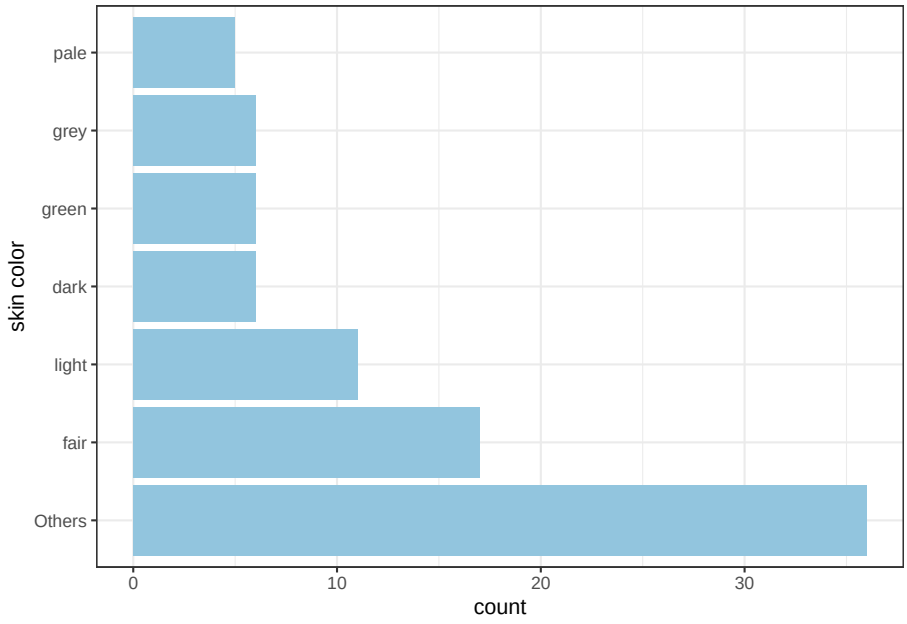

```
## 代码写这里，并运行；
ggplot(starwars, aes(x = fct_infreq(hair_color))) +
  geom_bar(fill = "#92C5DE") +
  theme_bw() +
  coord_flip()
```



2. 统计 skin_color 时，将出现频率小于 0.05（即 5%）的颜色归为一类 Others，按出现次数排序后，做与上面类似的 barplot；

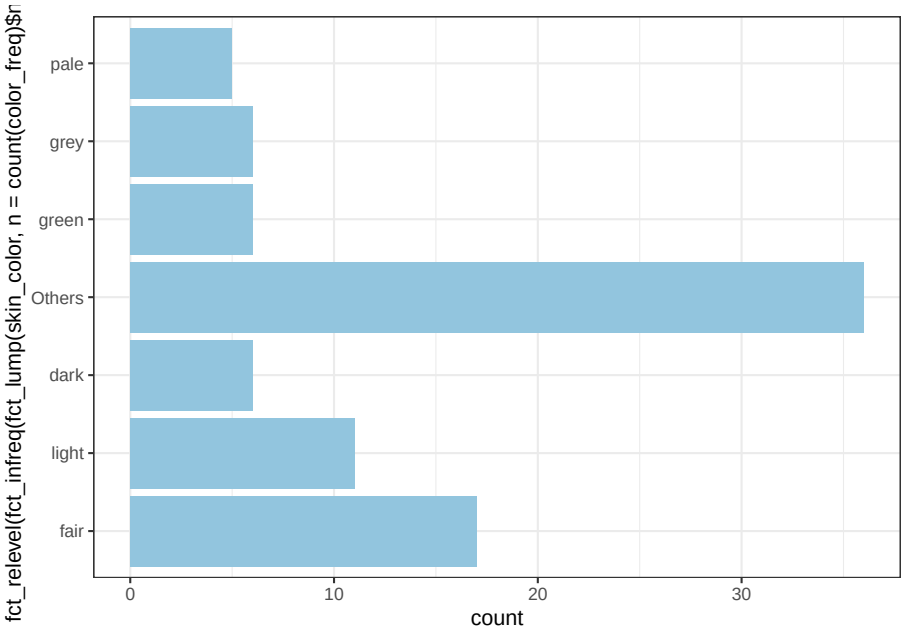
```
## 代码写这里，并运行；
# 统计每种 skin_color 的出现频率
color_freq <- starwars %>%
  group_by(skin_color) %>%
  summarise(n = n()) %>%
  mutate(freq = n / sum(n)) %>%
  filter(freq >= 0.05)

# 利用 fct_lump() 将出现频率小于 0.05 的颜色归为一类 Others,
# 参数 n 的作用是保留最常见的 n 种颜色
ggplot(starwars, aes(x = fct_infreq(fct_lump(skin_color,
    n = count(color_freq)$n, other_level = "Others")))) +
  geom_bar(fill = "#92C5DE") +
  # 添加坐标轴标签
  coord_flip() +
```

```
  theme_bw() +
  # x 轴改为 “skin color”
  xlab("skin color") +
  ylab("count")
```



3. 使用 2 的统计结果，但画图时，调整 bar 的顺序，使得 Others 处于第 4 的位置上。提示，可使用 fct_relevel 函数；

```
## 代码写这里，并运行；
ggplot(starwars, aes(x = fct_relevel(fct_infreq(fct_lump(skin_color,
  n = count(color_freq)$n, other_level = "Others")), "Others", after = 3))) +
  geom_bar(fill = "#92C5DE") +
  coord_flip() +
  theme_bw()
```



0.5 练习与作业 2：数据分析

0.5.1 使用 STRING PPI 数据分析并作图

1. 使用以下代码，装入 PPI 数据；

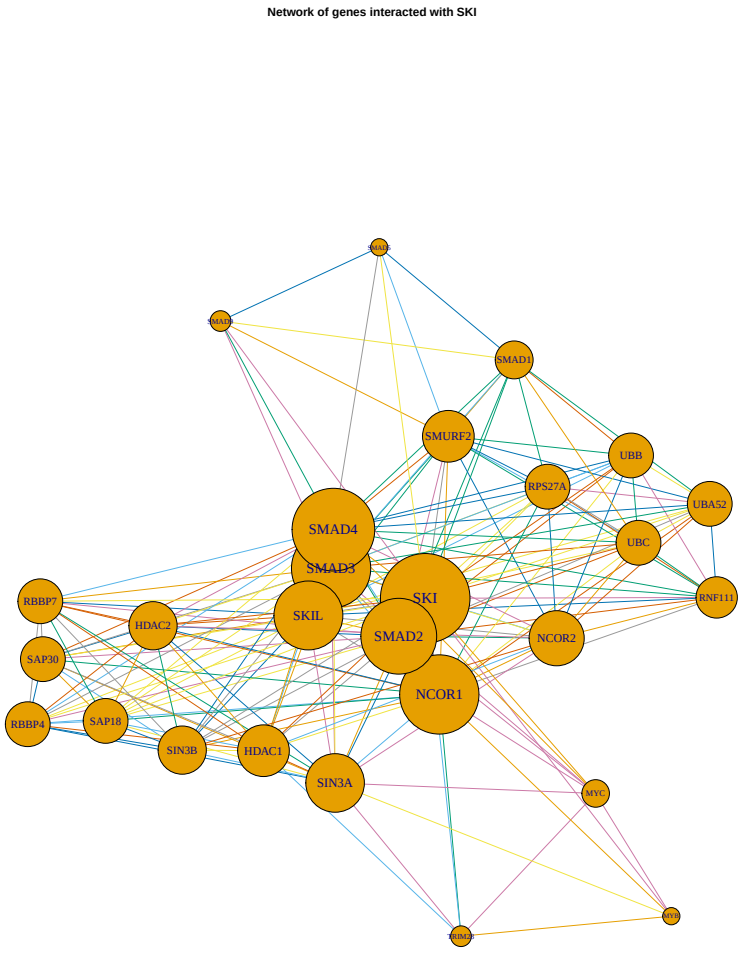
```
ppi <- read_delim( file = "../data/talk06/ppi900.txt.gz", col_names = T,
                   delim =  "\t", quote = "" );
```

2. **随机挑选**一个基因，得到类似于本章第一部分的互作网络图；

```
## 代码写这里，并运行；
ppi <- read_delim(file = "./data/talk06/ppi900.txt.gz", col_names = TRUE,
    delim =  "\t", quote = "", show_col_types = FALSE)
# 选择一个基因
```

```
set.seed(123)
gene_selected <- sample(ppi$gene1, 1)
# 选择与该基因互作的基因
ppi2 <- ppi %>%
  filter(gene1 == gene_selected) %>%
  ungroup() %>%
  # 每行 gene1 与 gene2 的集合去重
  distinct()
genes_tar <- unique(c(gene_selected, ppi2$gene2))

genes_net <- ppi %>%
  filter(gene1 %in% genes_tar & gene2 %in% genes_tar) %>%
  mutate(group = ifelse(gene2 > gene1,
    paste(gene1, gene2, sep = "-"),
    paste(gene2, gene1, sep = "-"))) %>%
  distinct(group, .keep_all = TRUE)



# 计算网络
# gene 与 gene2 之间的边的权重为 score
g <- graph_from_data_frame(d = genes_net, vertices = NULL,
  directed = FALSE)

# 每个点的大小为与该点互作的基因数
v_size <- degree(g)
# 边的颜色为 score
e_color <- genes_net$score
```

```
# 画图。
plot(g, layout = layout_nicely(g), vertex.size = v_size,
    legend = legend, width = 5, height = 5, edge.color = e_color,
    main = paste("Network of genes interacted with", gene_selected),
    vertex.label.cex = sqrt(v_size / max(v_size)) * 1.5)
```



0.5.2 对宏基因组相对丰度数据进行分析

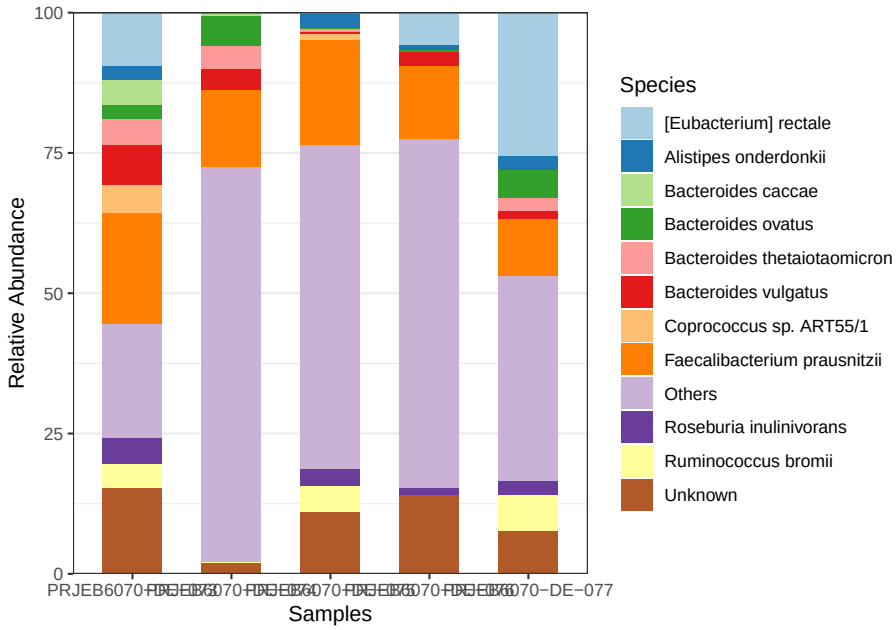
1.data/talk06 目录下有 6 个文本文件，每个包含了一个宏基因组样本的分析结果：

relative_abundance_for_curated_sample_PRJEB6070-DE-073_at_taxonlevel_species.txt
relative_abundance_for_curated_sample_PRJEB6070-DE-074_at_taxonlevel_species.txt
relative_abundance_for_curated_sample_PRJEB6070-DE-075_at_taxonlevel_species.txt
relative_abundance_for_curated_sample_PRJEB6070-DE-076_at_taxonlevel_species.txt
relative_abundance_for_curated_sample_PRJEB6070-DE-077_at_taxonlevel_species.txt

2. 分别读取以上文件，提取 scientific_name 和 relative_abundance 两列；
3. 添加一列为样本名，比如 PRJEB6070-DE-073, PRJEB6070-DE-074 … ；
4. 以 scientific_name 为 key，将其内容合并为一个 data.frame 或 tibble，其中每行为一个样本，每列为样本的物种相对丰度。注意：用 join 或者 spread 都可以，只要能解决问题。
5. 将 NA 值改为 0。

```
## 代码写这里，并运行；
# 读取文件
files <- list.files(path = "Exercises and homework/data/talk06/", full.names = TRUE)
# 提取 scientific_name 和 relative_abundance 两列
files_df <- lapply(files, function(x) {
    read_tsv(x, col_names = FALSE, skip = 4, show_col_types = FALSE) %>%
    select(scientific_name = X4, relative_abundance = X3)
})

# 添加一列为样本名
sample_names <- str_extract(files, "PRJEB6070-DE-\\d{3}")
sampled_df <- lapply(seq_along(files_df), function(x) {
    files_df[[x]] %>%
    mutate(sample_name = sample_names[x])
})

# 以 scientific_name 为 key，将其内容合并为一个 data.frame
taxon_abundance <- bind_rows(sampled_df) %>%
```

```
    dcast(scientific_name ~ sample_name, value.var = "relative_abundance",
    # 保留两位小数
    fun.aggregate = function(x) round(sum(x, na.rm = TRUE), 2)) %>%
    # 将 NA 值改为 0. 使用 replace_na() 也可以
    mutate_at(vars(-scientific_name), ~replace(., is.na(.), 0))


# 画图，每个样本的物种堆叠图
df1 <- taxon_abundance %>%
  # 每个样本保留相对丰度最高的前 10 个物种，其他物种合并为 Others
  arrange(desc(`PRJEB6070-DE-073`), desc(`PRJEB6070-DE-074`),
    desc(`PRJEB6070-DE-075`), desc(`PRJEB6070-DE-076`),
    desc(`PRJEB6070-DE-077`)) %>%
    # 前 11 个物种保留，其他物种合并为 Others
    mutate(scientific_name = ifelse(row_number() <= 11,
      scientific_name, "Others")) %>%
    group_by(scientific_name) %>%
    # 按照物种名合并
    summarise_all(sum) %>%
    melt(id.vars = "scientific_name") %>%
    tibble() %>%
    filter(value > 0)

ggplot(df1, aes(x = variable, y = value, fill = scientific_name)) +
  geom_col(position = "stack", width = 0.6) +
  scale_fill_manual(values = brewer.pal(n = 12, "Paired")) +
  theme_bw() +
  scale_y_continuous(expand = c(0,0)) + # 调整 y 轴属性，使柱子与 X 轴坐标接触
  labs(x = "Samples", y = "Relative Abundance",
       fill = "Species")
```


100
75
50
25
0
Relative Abundance
Samples
PRJEB6070−DE−077
Species
[Eubacterium] rectale
Alistipes onderdonkii
Bacteroides caccae
Bacteroides ovatus
Bacteroides thetaiotaomicron
Bacteroides vulgatus
Coprococcus sp. ART55/1
Faecalibacterium prausnitzii
Others
Roseburia inulinivorans
Ruminococcus bromii
Unknown