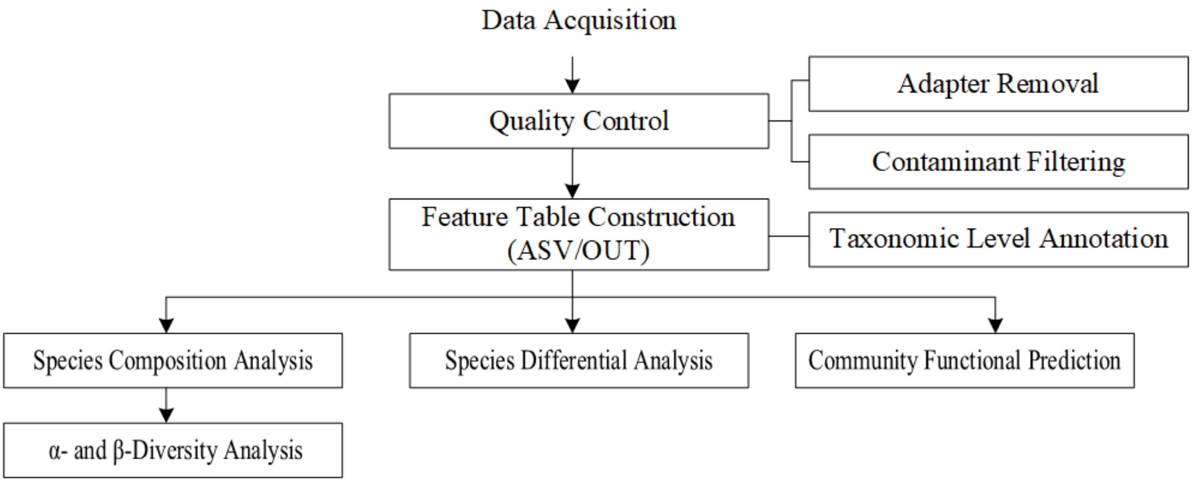
Metagenomics Basic Data Processing Code Manual


Data Acquisition
Quality Control
Adapter Removal
Contaminant Filtering
Feature Table Construction (ASV/OUT)
Taxonomic Level Annotation
Species Composition Analysis
Species Differential Analysis
Community Functional Prediction
α- and β-Diversity Analysis

catalogue

1. Data Acquisition (prefetch)

2. Quality Control (fastqc)

3. Adapter Removal and Contaminant Filtering (cutadapt)

4. Obtain ASV Feature Table and Taxonomic Annotation Table (dada2)

5. Genome Assembly (megahit)

6. Species Composition Analysis

7. Community α- and β-Diversity Analysis

8. Species Differential Analysis

9. Community Differential Analysis

10. Generate OTU Table (qiime2)

11. Community Functional Prediction (PICRUSt2)

1. 数据获取prefetch

> 记录案例

```
prefetch PRJNA868882
prefetch PRJNA979234
```

2. 质量控制fastqc

报告查看

https://www.bilibili.com/opus/947976634643775538

```
fastq-dump --split-files SRR30228527.sra

# 多文件处理双端序列
for dir in SRR*/; do
    for file in "$dir"*.sra; do
        if [ -f "$file" ]; then
            fastq-dump --split-files "$file"
        fi
    done
done

fastqc SRR30228527_1.fastq SRR30228527_2.fastq

# 批量处理
for file in *_1.fastq; do
    base=${file%_1.fastq}   # 获取基名称
    fastqc "${base}_1.fastq" "${base}_2.fastq"
done

for file in *_1_trimmed.fastq; do
    base=${file%_1_trimmed.fastq}   # 获取基名称
    fastqc "${base}_1_trimmed.fastq" "${base}_2_trimmed.fastq"
done

# 汇总报告
multiqc .(conda activate py310)

python3 -m http.server
```

汇总报告解读

[MultiQC的使用及html报告解析 - 简书]

查看序列数量

```
grep -c '^@' SRR30213932_2.fastq

wc -l SRR30228527_1.fastq
wc -l SRR30228527_2.fastq
```

检查序列seqkit

```
seqkit stats SRR24807541_1_trimmed.fastq
```

3.去接头cutadapt

https://cutadapt.readthedocs.io/en/v1.18/guide.html#illumina-truseq

查看序列前几行

```
awk 'NR % 4 == 1' SRR24807548_1.fastq | head -n 10
```

去除Illumina测序的头序列，通过fastqc中的adapter模块检查

```
cutadapt -a AGATCGGAAGAGCACACGTCTGAACTCCAGTCAC -A AGATCGGAAGAGCGTCGTGTAGGGAAAGAGTGT -o SRR30118025_1_trimmed.fastq -p SRR30118025_2_trimmed.fastq SRR30118025_1.fastq SRR30118025_2.fastq

# 批量处理

#!/bin/bash

# 设置输入和输出目录
input_dir="/root/db/PRJNA868882_DEseq_deal"
output_dir="/root/db/PRJNA868882_clean"

# 创建输出目录（如果不存在）
mkdir -p "$output_dir"

# 遍历所有的双端 FASTQ 文件
for file1 in "$input_dir"/*_1.fastq; do
  # 生成对应的文件名
  file2="${file1/_1.fastq/_2.fastq}"
  trimmed1="${output_dir}/$(basename "${file1/_1.fastq/_1_trimmed.fastq}")"
  trimmed2="${output_dir}/$(basename "${file2/_2.fastq/_2_trimmed.fastq}")"

  # 检查第二个文件是否存在
  if [[ -f "$file2" ]]; then
    # 检查输出文件是否已经存在
    if [[ -f "$trimmed1" ]] || [[ -f "$trimmed2" ]]; then
      echo "Warning: Output files $trimmed1 or $trimmed2 already exist. Skipping."
      continue
    fi

    # 运行 Cutadapt
    cutadapt -a AGATCGGAAGAGCACACGTCTGAACTCCAGTCAC -A AGATCGGAAGAGCGTCGTGTAGGGAAAGAGTGTAGATCTCGGTGGTCGCCGTATCATT -o "$trimmed1" -p "$trimmed2" "$file1" "$file2"


    echo "Processed $file1 and $file2"
  else
    echo "Warning: Matching file for $file1 not found."
  fi
done
```

接头选择

针对PRJNA868882 火山数据

Illumina Miseq PE300平台

官方文档

```
cutadapt a AGATCGGAAGAGCACACGTCTGAACTCCAGTCAC -A AGATCGGAAGAGCGTCGTGTAGGGAAAGAGTGTAGATCTCGGTGGTCGCCGTATCATT -o trimmed.R1.fastq.gz -p trimmed.R2.fastq.gz reads.R1.fastq.gz reads.R2.fastq.gz
```

来自博客https://blog.csdn.net/qq_27390023/article/details/120528478

```
cutadapt -a AGATCGGAAGAGCACACGTCTGAACTCCAGTCA -A
AGATCGGAAGAGCGTCGTGTAGGGAAAGAGTGT -o trimmed.R1.fastq.gz -p trimmed.R2.fastq.gz
reads.R1.fastq.gz reads.R2.fastq.gz
```

针对PRJNA979234 热泉数据

为了测序，按照Illumina的标准方案，使用提取的DNA用两个位点特异性引物扩增16S rDNA区V3-V4，引物为Illumina悬垂物16S-341（5'-TCGTGCGCGCGCGAGCGTCAGATGTATAGAGAGAGACAGCCTCGGGGNBGCASCAG-3'）和16S-805R（5'-GTCCGCGGGCTCGGAGAGTGTATAAGAGAGAGGACTACNVGGGTATCTAATCC-3'）

```
cutadapt -a TCGTGCGCGCGCGAGCGTCAGATGTATAGAGAGAGACAGCCTCGGGGNBGCASCAG -A
GTCCGCGGGCTCGGAGAGTGTATAAGAGAGAGGACTACNVGGGTATCTAATCC -o trimmed.R1.fastq.gz -p
trimmed.R2.fastq.gz reads.R1.fastq.gz reads.R2.fastq.gz
```

去已知污染物

```
cutadapt -c contaminants.txt -o trimmed_output.fastq your_file.fastq
```

去除两端低质量序列

```
cutadapt -q 10 -o output.fastq input.fastq  # 3‘端
功能：修剪 input.fastq 文件中每个读取的 3' 端，去除质量低于 10 的碱基。
结果：输出修剪后的结果到 output.fastq。
注意：这个命令只处理每个读取的 3' 端。

cutadapt -q 10,10 output.fastq input.fastq
# -q [5'CUTOFF,]3'CUTOFF, --quality-cutoff [5'CUTOFF,]3'CUTOFF
# 质量值，默认为phred33 （+33）。可以设定 phred64 --quality-base=64
功能：修剪 input.fastq 文件中每个读取的 5' 端和 3' 端，去除质量低于 10 的碱基。
结果：输出修剪后的结果到 output.fastq。
注意：这个命令同时处理 5' 和 3' 端，确保两端的低质量碱基都被去除。

# 单端序列
#!/bin/bash
# 创建输出目录（如果不存在）
mkdir -p trimmed_outputs

# 遍历所有输入的 FASTQ 文件
for file in *.fastq; do
    # 提取文件名，不带扩展名
    base_name=$(basename "$file" .fastq)

    # 执行 cutadapt，修剪低质量碱基
    cutadapt -q 10,10 -o "trimmed_outputs/${base_name}_trimmed.fastq" "$file"
done

# 双端序列
#!/bin/bash
# 创建输出目录（如果不存在）
mkdir -p trimmed_outputs
```

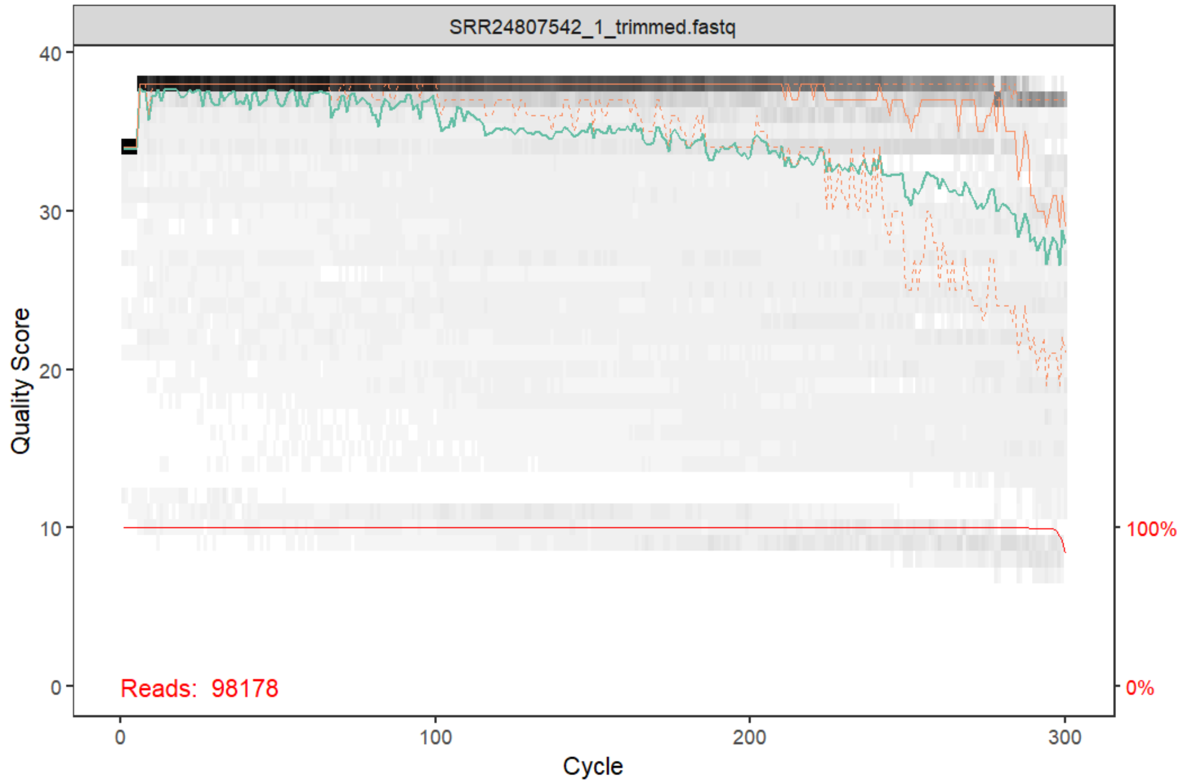
```
# 遍历所有正向和反向 FASTQ 文件
for fwd_file in *_1.fastq; do
    # 提取基本文件名
    base_name=$(basename "$fwd_file"_1.fastq)
    rev_file="${base_name}_2.fastq"

    # 检查反向文件是否存在
    if [[ -f "$rev_file" ]]; then
        # 执行 cutadapt，修剪低质量碱基
        cutadapt -q 10,10 -o "trimmed_outputs/${base_name}_1_trimmed.fastq" \
                 -p "trimmed_outputs/${base_name}_2_trimmed.fastq" \
                 "$fwd_file" "$rev_file"
    else
        echo "警告：找不到配对反向文件 ${rev_file}，跳过 ${fwd_file}。"
    fi
done
```

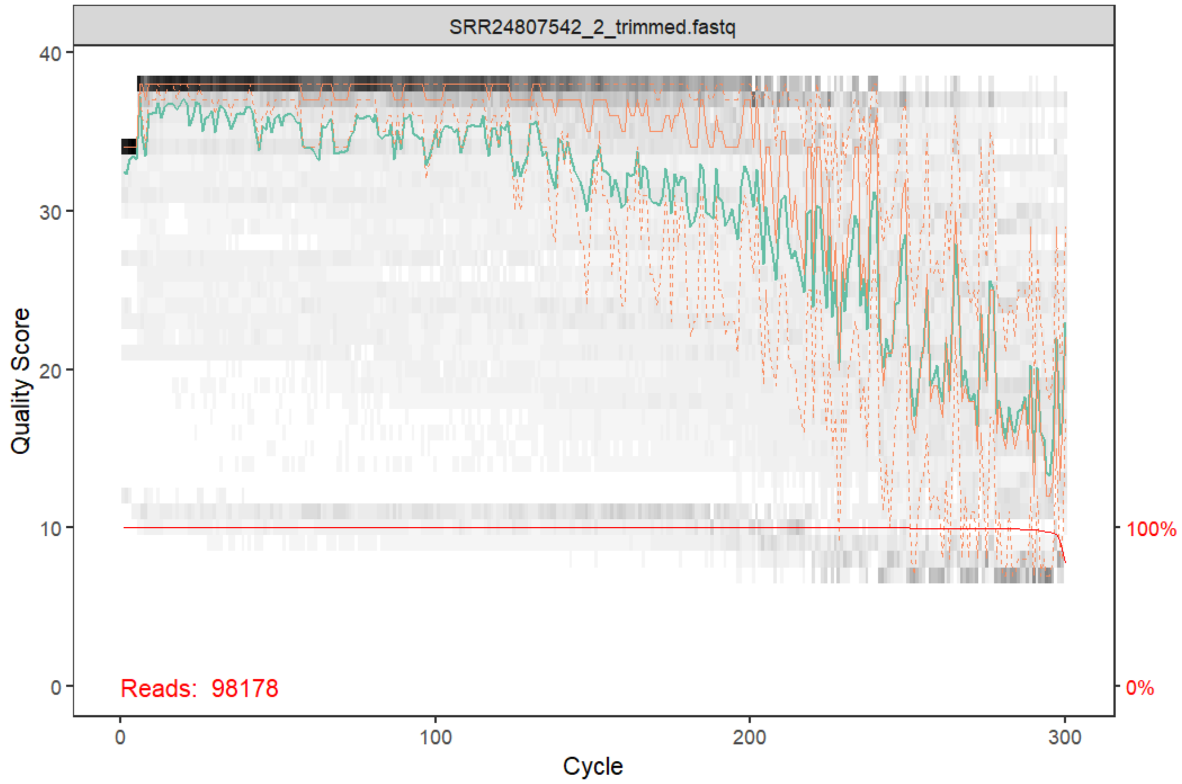
4.dada2获取ASV特征表与物种注释表

> 转windows R语言

4.1 数据导入与质量分数查看

```
# 加载必要的包
library(dada2)
library(Biostrings)
library(tidyverse)

# 设置文件路径
fwd_files <- c("./SRR24807542_1_trimmed.fastq", "./SRR24807542_2_trimmed.fastq")
rev_files <- c("./SRR24807542_2_trimmed.fastq", "./SRR24807542_1_trimmed.fastq")

# 1. 读取FastQ文件
raw_fastq_fwd <- lapply(fwd_files, readDNAStringSet, format = "fastq")
raw_fastq_rev <- lapply(rev_files, readDNAStringSet, format = "fastq")

# 检查输入数据
lapply(raw_fastq_fwd[[1]], class)
lapply(raw_fastq_fwd[[1]], head)
# 检查数据格式
class(raw_fastq_fwd[[1]])
# 检查长度和内容
print(length(raw_fastq_fwd))
print(head(raw_fastq_fwd[[1]]))
# 检查所有读取的序列
lapply(raw_fastq_fwd, function(x) {
  sum(width(x))  # 检查每个文件的序列长度
})

# 2. 绘制质量分布图
plotQualityProfile(fwd_files[1])
plotQualityProfile(rev_files[1])
```





解读：灰度图是每个基本位置上每个质量分数的频率的热图。每个位置的平均质量分数由绿线表示，质量分数分布的四分位数由两条橙色虚线表示。每个位置的质量值中位数由橙色实线表示。如果序列的长度不同，则将绘制一条红线，显示扩展到该位置的序列百分比，这对454焦磷酸测序等技术更有用，因为Illumina读数通常都是相同的长度。正向序列质量前段序列质量较好，后端较差，通常建议修剪掉开头的引物与barcode等非生物序列，也会裁剪掉后端错误率比较高的核苷酸序列。反向序列质量往往比正向序列差，特别是在最后，从平均质量值突降的位置进行剪切会提高算法对稀有序列的敏感性。

4.2 序列裁剪与过滤

可参考https://blog.csdn.net/HUANWEIFENXI/article/details/124134240

与cutadapt差不多效果

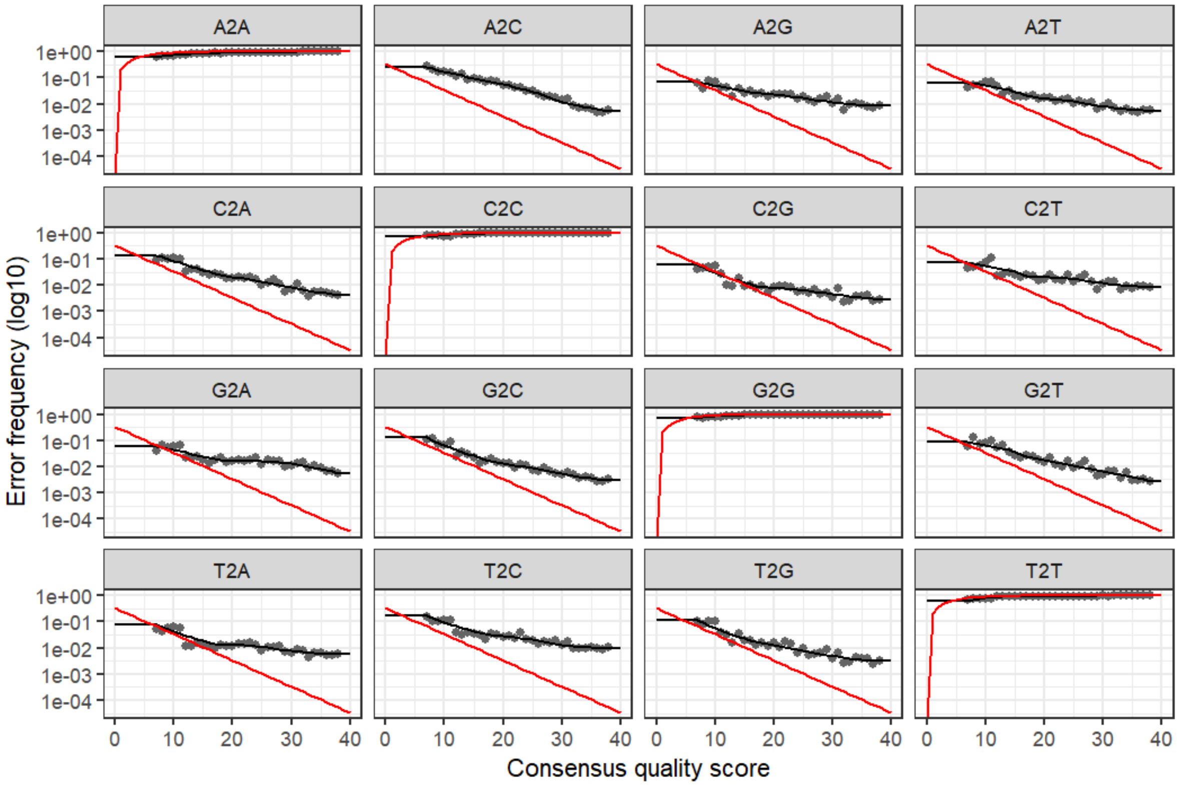
```
cutadapt -a TCGTGCGCGCGCGAGCGTCAGATGTATAGAGAGAGACAGCCTCGGGGNBGCASCAG -A
GTCCGCGGGCTCGGAGAGTGTATAAGAGAGAGGACTACNVGGGTATCTAATCC -o trimmed.R1.fastq.gz -p
trimmed.R2.fastq.gz reads.R1.fastq.gz reads.R2.fastq.gz
```

4.3 计算错误率

```
err_fwd <- learnErrors(fwd_files, multithread = TRUE)
err_rev <- learnErrors(rev_files, multithread = TRUE)
plotErrors(err_fwd, nominalQ=TRUE)
plotErrors(err_rev, nominalQ=TRUE)
```

`plotErrors(derep_rev, nominalQ=TRUE)`

`plotErrors(derep_fwd, nominalQ=TRUE)` 和 `plotErrors(derep_rev, nominalQ=TRUE)` 生成的图看起来差不多，这通常是因为正向和反向序列的错误模式相似。**测序技术**：如果使用相同的测序技术（例如 Illumina），正向和反向序列的错误模式可能会相似。**样本特性**：样本本身的质量和特性可能导致正向和反向序列的错误分布相似。**数据处理**：如果在数据处理过程中（如 `cutadapt`）对正向和反向序列进行了相似的质量控制，可能会导致错误模式趋同。



错误率统计图表示了所有可能的错误（A→C，A→G，...），图中点表示观察得到的错误率，黑线表示通过算法学习评估得到的错误率，红色的曲线表示由Q-score的定义下预期的错误率。观察拟合程度，拟合程度较好，并且错误率随着预期质量下降而下降。

提示：DADA2核心算法亦是参数学习，计算量非常可观。面对如此巨大的数据和需要消耗的计算资源，这一模型的展示便不适合实际较大的数据量，可以通过增加nbase参数调整拟合程度以减少计算量。

4.4 去冗余

可参考https://blog.csdn.net/HUANWEIFENXI/article/details/124134240

```
derepclass_f <- derepFastq(fwd_files)
derepclass_r <- derepFastq(rev_files)
```

在宏基因组数据处理中丰度分析、功能分析和样本特征等保留冗余也许有益处。

4.5 DADA2核心算法

DADA2算法是一种分裂式分割算法。首先，将每个reads全部看作单独的单元，sequence相同的reads被纳入一个sequence，reads个数即成为该sequence的丰度（abundance）；其次，计算每个sequence丰度的p-value，当最小的p-value低于设定的阈值时，将产生一个新的partition。每一个sequence将会被归入最可能生成该sequence的partition；最后，依次类推，完成分割归并。

```
# 基于错误模型进一步质控

# 去噪正向序列
dada_fwd <- dada(derepclass_f, pool = "pseudo", err = err_fwd, multithread =
TRUE)

# 去噪反向序列
dada_rev <- dada(derepclass_r, pool = "pseudo", err = err_rev, multithread =
TRUE)
```

4.6 合并双端序列

`mergers` 通常指的是合并的序列，尤其是在使用 DADA2 或其他去噪工具时，指的是将正向和反向测序的结果合并成一个完整的序列。这个过程是为了生成更完整的 DNA 序列，通常用于分析多样性或构建操作分类单元（OTUs）或序列变体（ASVs）。

```
# 合并正向和反向序列
mergers <- mergePairs(dada_fwd, derepclass_f, dada_rev, derepclass_r,
verbose=TRUE)
```

```
> head(mergers[[1]])
sequence
1
 CCTACGGGTGGCTGCAGTGGGGAATTTTCCGCAATGGGCGAAAGCCTGACGGAGCAACGCCGCGTGAGGGATGAAGGCCT
CTGGGCTGTAAACCTCTTTTATCAAGGAAGAAGATCTGACGGTACTTGATGAATAAGCCACGGCTAATTCCGTGCCAGCAG
CCGCGGTAATACGGGAGTGGCAAGCGTTATCCGGAATTATTGGGCGTAAAGCGTCCGCAGGCGGCCCTTCAAGTCTGCTGT
TAAAAAGTGGAGCTTAACTCCATCATGGCAGTGGAAACTGTTGGGCTTGAGTGTGGTAGGGGCAGAGGGAATTCCCGGTGT
AGCGGTGAAATGCGTAGATATCGGGAAGAACACCAGTGGCGAAGGCGCTCTGCTGGGCCATCACTGACGCTCATGGACGAA
AGCCAGGGGAGCGAAAGGGATTAGATACCCGGGTAGTC
```

```
2
 CCTACGGGAGGCAGCAGTGGGGAATTTTCCGCAATGGGCGAAAGCCTGACGGAGCAACGCCGCGTGAGGGATGAAGGCCT
CTGGGCTGTAAACCTCTTTTATCAAGGAAGAAGATCTGACGGTACTTGATGAATAAGCCACGGCTAATTCCGTGCCAGCAG
CCGCGGTAATACGGGAGTGGCAAGCGTTATCCGGAATTATTGGGCGTAAAGCGTCCGCAGGCGGCCCTTCAAGTCTGCTGT
TAAAAAGTGGAGCTTAACTCCATCATGGCAGTGGAAACTGTTGGGCTTGAGTGTGGTAGGGGCAGAGGGAATTCCCGGTGT
AGCGGTGAAATGCGTAGATATCGGGAAGAACACCAGTGGCGAAGGCGCTCTGCTGGGCCATCACTGACGCTCATGGACGAA
AGCCAGGGGAGCGAAAGGGATTAGATACCCGGGTAGTC
3
 CCTACGGGCGGCTGCAGTGGGGAATTTTCCGCAATGGGCGAAAGCCTGACGGAGCAACGCCGCGTGAGGGATGAAGGCCT
CTGGGCTGTAAACCTCTTTTATCAAGGAAGAAGATCTGACGGTACTTGATGAATAAGCCACGGCTAATTCCGTGCCAGCAG
CCGCGGTAATACGGGAGTGGCAAGCGTTATCCGGAATTATTGGGCGTAAAGCGTCCGCAGGCGGCCCTTCAAGTCTGCTGT
TAAAAAGTGGAGCTTAACTCCATCATGGCAGTGGAAACTGTTGGGCTTGAGTGTGGTAGGGGCAGAGGGAATTCCCGGTGT
AGCGGTGAAATGCGTAGATATCGGGAAGAACACCAGTGGCGAAGGCGCTCTGCTGGGCCATCACTGACGCTCATGGACGAA
AGCCAGGGGAGCGAAAGGGATTAGATACCCGGGTAGTC
4
 CCTACGGGTGGCTGCAGTGGGGAATCTTAGACAATGGGCGCAAGCCTGATCTAGCCATGCCGCGTGAGTGATGAAGGCCT
TAGGGTCGTAAAGCTCTTTCGCCTGTGATGATAATGACAGTAGCAGGTAAAGAAACCCCGGCTAACTCCGTGCCAGCAGCC
GCGGTAATACGGAGGGGGTTAGCGTTGTTCGGAATTACTGGGCGTAAAGCGCACGTAGGCGGATTGGAAAGTTGGGGGTGA
AATCCCGGGGCTCAACCCCGGAACTGCCTCCAAAACTATCAGTCTAGAGTTCGAGAGAGGTGAGTGGAATTCCAAGTGTAG
AGGTGAAATTCGTAGATATTTGGAGGAACACCAGTGGCGAAGGCGGCTCACTGGCTCGATACTGACGCTGAGGTGCGAAAG
TGTGGGGAGCAAACAGGATTAGATACCCGGGTAGTC
5
CCTACGGGTGGCTGCAGTGGGGAATATTGCACAATGAGCGAAAGCTTGATGCAGCCATACCGCGTGTGTGAAGAAGGCCCG
AGGGTTGTAAAGCACTTTCAATTGTGAGGAAGATAGTGTCGCTAATATCGGCATTGTTTGACGTTAACTTTAGAAGAAGCA
CCGGCTAACTCTGTGCCAGCAGCCGCGGTAATACAGAGGGTGCAAGCGTTATTCGGAATTACTGGGCGTAAAGCGCGCGTA
GGCGGGCTATTAAGTCAGTTGTGAAAGCCCTGGGCTCAACCTGGGAACTGCATCTGATACTGGTAGTCTAGAGTTTAAGAG
AGGGAAGTGGAATTCCAGGTGTAGCAGTGAAATGCGTAGATATCTGGAGGAACATCAGTGGCGAAGGCGACTTCCTGGCTT
AAAACTGACGCTGAGGTGCGAAAGCGTGGGTAGCGAACGGGATTAGATACCCGGGTAGTC
6
 CCTACGGGGGGCTGCAGTGGGGAATTTTCCGCAATGGGCGAAAGCCTGACGGAGCAACGCCGCGTGAGGGATGAAGGCCT
CTGGGCTGTAAACCTCTTTTATCAAGGAAGAAGATCTGACGGTACTTGATGAATAAGCCACGGCTAATTCCGTGCCAGCAG
CCGCGGTAATACGGGAGTGGCAAGCGTTATCCGGAATTATTGGGCGTAAAGCGTCCGCAGGCGGCCCTTCAAGTCTGCTGT
TAAAAAGTGGAGCTTAACTCCATCATGGCAGTGGAAACTGTTGGGCTTGAGTGTGGTAGGGGCAGAGGGAATTCCCGGTGT
AGCGGTGAAATGCGTAGATATCGGGAAGAACACCAGTGGCGAAGGCGCTCTGCTGGGCCATCACTGACGCTCATGGACGAA
AGCCAGGGGAGCGAAAGGGATTAGATACCCGGGTAGTC
  abundance forward reverse nmatch nmismatch nindel prefer accept
1      1022       4       5    158         0      0      1   TRUE
2       620      12       5    158         0      0      1   TRUE
3       572      14       5    158         0      0      1   TRUE
4       526      16       6    160         0      0      1   TRUE
5       494       1      26    135         0      0      1   TRUE
6       475      19       5    158         0      0      1   TRUE
```

4.7 构建ASV表

OTU 是一个操作分类单元，用于将相似的 DNA 序列归类为同一类群。具体来说，OTU 通常是基于 DNA 序列的相似性（如 97% 或 99% 的相似性）来定义的。

ASV 是指通过高通量测序技术获得的每一个独特的扩增子序列变体。与 OTU 的聚类方法不同，ASV 是基于序列的实际变异性来定义的，通常不进行聚类。

行和列：ASV(OTU) 表通常以矩阵的形式呈现，行代表不同的 ASV(OTU)，列代表不同的样本。

```
# 构建ASV表，amplicon sequence variant（ASV）表类似于我们传统的OTU表
seqtab <- makeSequenceTable(mergers)
# dim()的第二个值为扩增子序列个数，table(nchar())的统计结果表示每个读长下有多少个扩增子序列。
dim(seqtab)
# Inspect distribution of sequence lengths 查看序列长度分布
table(nchar(getSequences(seqtab)))
```

```
> seqtab <- makeSequenceTable(mergers)
> dim(seqtab)
[1]    2 1482
> table(nchar(getSequences(seqtab)))

440 442 444 445 459 460 461 465
352  54   2   2   2 204   4 862
```

4.8 去除嵌合体

Dada核心质控算法去除了大部分错误，但嵌合体仍然存在，去噪后序列的准确性使得识别嵌合体比处理模糊OTU更简单。

```
#去除嵌合体
seqtab.nochim <- removeBimeraDenovo(seqtab, method="consensus", multithread=TRUE,
verbose=TRUE)
dim(seqtab.nochim)
sum(seqtab.nochim)/sum(seqtab)
```

```
> seqtab.nochim <- removeBimeraDenovo(seqtab, method="consensus",
multithread=TRUE, verbose=TRUE)
Identified 1182 bimeras out of 1482 input sequences.
> dim(seqtab.nochim)
[1]   2 300
> sum(seqtab.nochim)/sum(seqtab)
[1] 0.3773218
```

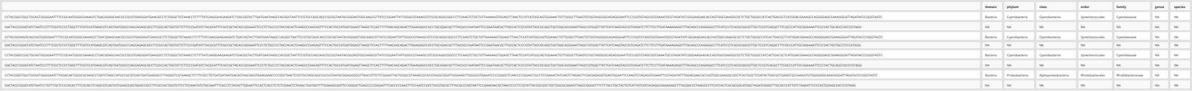
4.9 导出生成的ASV表

```
write.csv(seqtab.nochim,file="./ASV.CSV",append = FALSE, quote = FALSE , sep = "
",eol = "\n", na = "NA", dec = ".", row.names = TRUE,col.names = TRUE, qmethod =
c("escape", "double"),fileEncoding = "")
```

4.10 物种注释

```
# 安装DECIPHER，加载DECIPHER
BiocManager::install("DECIPHER")
library(DECIPHER)
packageVersion("DECIPHER")
# 转换ASV表为DNAString格式
# Create a DNAStringSet from the ASVs
dna <- DNAStringSet(getSequences(seqtab.nochim))
# 相关数据下载详见DECIPHER教程，并修改为下载目录
# CHANGE TO THE PATH OF YOUR TRAINING SET
file_path <- "./SILVA_SSU_r138_2019.RData"
if (file.exists(file_path)) {
  load(file_path)
} else {
  stop("文件不存在：", file_path)
}

# use all processors
ids <- IdTaxa(dna, trainingSet, strand="top", processors=NULL, verbose=FALSE) #
ranks of interest
ranks <- c("domain", "phylum", "class", "order", "family", "genus", "species")
# Convert the output object of class "Taxa" to a matrix analogous to the output
from assignTaxonomy
# 假设 ranks 和 ids 已经定义
taxid <- t(sapply(ids, function(x) {
  m <- match(ranks, x$rank)
  taxa <- x$taxon[m]
  taxa[startsWith(taxa, "unclassified_")] <- NA
  return(taxa)  # 添加 return 语句
}))

# 设置列名和行名
colnames(taxid) <- ranks
rownames(taxid) <- getSequences(seqtab.nochim)
taxa.print.DECIPHER.taxa <- taxa <- taxid
head(taxa.print.DECIPHER.taxa)

# Removing sequence rownames for display only taxa.print.DECIPHER <- taxa <-
taxid rownames(taxa.print.DECIPHER) <- NULL head(taxa.print.DECIPHER)
taxid

write.csv(taxa.print.DECIPHER.taxa,file="taxa.print.DECIPHER.CSV",append = FALSE,
quote = FALSE , sep = " ",eol = "\n", na = "NA", dec = ".", row.names =
TRUE,col.names = TRUE, qmethod = c("escape", "double"),fileEncoding = "")
```

eg



	domain	phylum	class	order	family	genus	species
[illegible]	Bacteria	Cyanobacteria	Cyanobacteriia	Synechococcales	Cyanobiaceae	NA	NA
[illegible]	NA	NA	NA	NA	NA	NA	NA
[illegible]	Bacteria	Cyanobacteria	Cyanobacteriia	Synechococcales	Cyanobiaceae	NA	NA
[illegible]	NA	NA	NA	NA	NA	NA	NA
[illegible]	Bacteria	Cyanobacteria	Cyanobacteriia	Synechococcales	Cyanobiaceae	NA	NA
[illegible]	NA	NA	NA	NA	NA	NA	NA
[illegible]	Bacteria	Proteobacteria	Alphaproteobacteria	Rhodobacterales	Rhodobacteraceae	NA	NA
[illegible]	NA	NA	NA	NA	NA	NA	NA

4.11 完整处理所有数据

```
# 加载必要的包
library(dada2)
library(Biostrings)
library(tidyverse)

# 设置文件路径
# fwd_files <- list.files(path =
"D:/Desktop/show/PRJNA979234_trimmed/trimmed_outputs", pattern =
"_1_trimmed.fastq", full.names = TRUE)
```

```
# rev_files <- list.files(path = "D:/Desktop/show/PRJNA979234_trimmed/trimmed_outputs", pattern = "_2_trimmed.fastq", full.names = TRUE)
path <- "D:/Desktop/show/PRJNA868882_trimmed/trimmed_outputs"
list.files(path)
fwd_files <- sort(list.files(path, pattern= "_1_trimmed.fastq", full.names = TRUE))
rev_files <- sort(list.files(path, pattern= "_2_trimmed.fastq", full.names = TRUE))
# 确保文件对的数量匹配
# if (length(fwd_files) != length(rev_files)) {
#   stop("正向和反向 FASTQ 文件数量不匹配！")
# }
# fwd_basenames <- sub("_1_trimmed.fastq$", "", basename(fwd_files))
# rev_basenames <- sub("_2_trimmed.fastq$", "", basename(rev_files))
# paired_files <- fwd_basenames %in% rev_basenames
#
# # 预先定义一个空列表来存储所有的 ASV 表
# all_asv_tables <- list()

# 循环处理每对 FASTQ 文件
# 1. 读取FastQ文件
derepclass_f <- derepFastq(fwd_files)
derepclass_r <- derepFastq(rev_files)

# 2. 学习错误模型
err_fwd <- learnErrors(fwd_files, multithread = TRUE)
err_rev <- learnErrors(rev_files, multithread = TRUE)

# 3. 去噪正向序列
# dada_fwd <- dada(derepclass_f, pool = "pseudo", err = err_fwd, multithread = TRUE)
dada_fwd <- dada(derepclass_f, err = err_fwd, multithread = TRUE)
# 4. 去噪反向序列
# dada_rev <- dada(derepclass_r, pool = "pseudo", err = err_rev, multithread = TRUE)
dada_rev <- dada(derepclass_r, err = err_rev, multithread = TRUE)

# 5. 合并正向和反向序列
mergers <- mergePairs(dada_fwd, derepclass_f, dada_rev, derepclass_r, verbose = TRUE)

# 6. 构建 ASV 表
seqtab <- makeSequenceTable(mergers)

# 7. 去除嵌合体
seqtab.nochim <- removeBimeraDenovo(seqtab, method = "consensus", multithread = TRUE, verbose = TRUE)


# 9. 导出 ASV 表
write.csv(seqtab.nochim,file="all_ASV_2.CSV",append = FALSE, quote = FALSE , sep = " ",eol = "\n", na = "NA", dec = ".", row.names = TRUE,col.names = TRUE, qmethod = c("escape", "double"),fileEncoding = "")
```

5.基因组装megahit

在数据清洗之后

```
# 双端序列组装：
    megahit -1 pe_1.fq -2 pe_2.fq -o out
    #-1：pair-end 1序列，-2 pair-end 2序列，-o输出目录
# 单端序列：
    megahit -r single_end.fq -o out
# 交错的双端序列：
    megahit --12 interleaved.fq -o out



# 批量处理

#!/bin/bash

# 设置输入目录和输出目录
input_directory="."  # 当前目录
output_directory="megahit_output"

# 创建输出目录（如果不存在）
mkdir -p "$output_directory"

# 遍历所有 _1.fastq 文件
for input_file_1 in "$input_directory"/*_1_trimmed.fastq; do
    # 获取对应的 _2.fastq 文件
    input_file_2="${input_file_1/_1_trimmed.fastq/_2_trimmed.fastq}"

    # 确保 _2.fastq 文件存在
    if [[ -f "$input_file_2" ]]; then
        # 获取基础文件名（不带路径）
        base_name=$(basename "$input_file_1" "_1_trimmed.fastq")

        # 运行 megahit 进行组装
        megahit -1 "$input_file_1" -2 "$input_file_2" -o
"$output_directory/$base_name"

        echo "Processed: $input_file_1 and $input_file_2"
    else
        echo "Warning: Corresponding _2_trimmed.fastq file not found for
$input_file_1"
    fi
done
```

组装质量检测quast

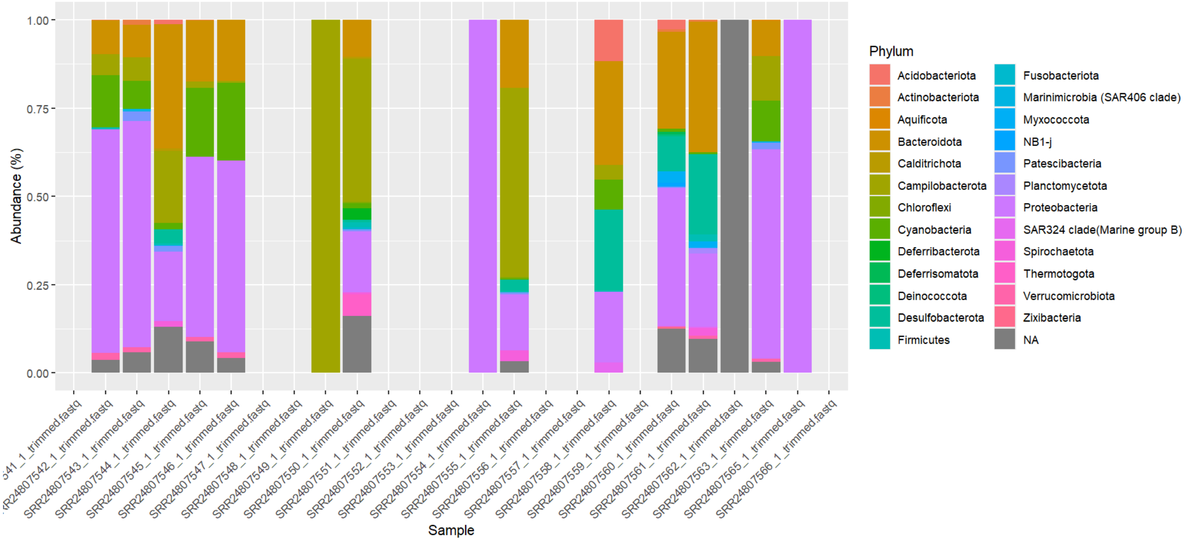
```
Usage:
$ quast.py test_data/contigs_1.fasta \
           test_data/contigs_2.fasta \
        -r test_data/reference.fasta.gz \
        -g test_data/genes.txt \
        -o quast_test_output
```

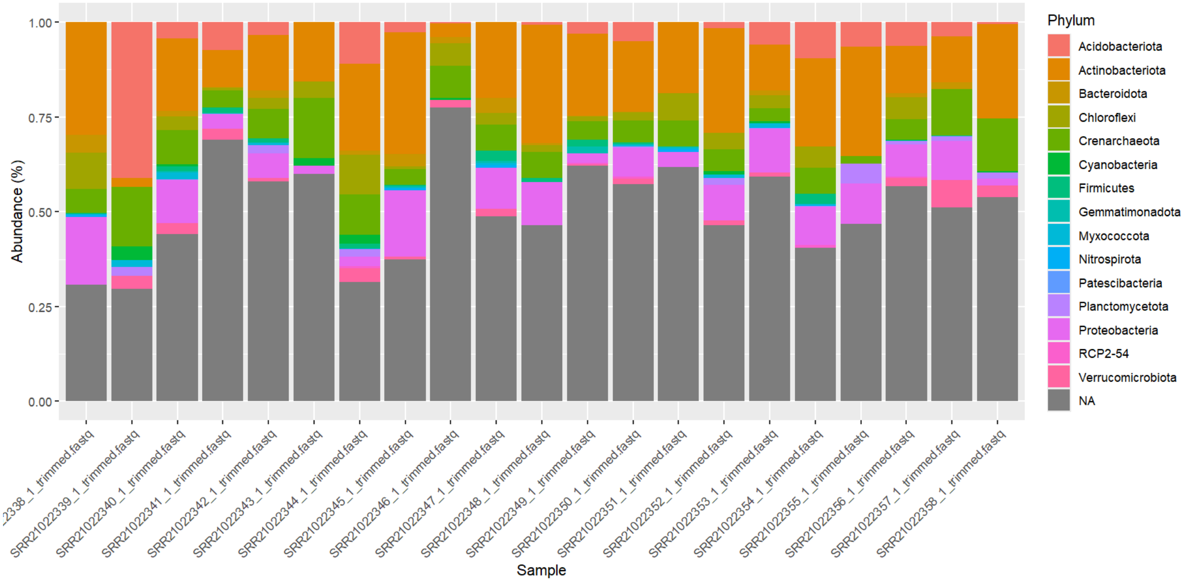
样例

```
quast.py
/root/db/PRJNA979234_trimmed/trimmed_outputs/megahit_output/SRR24807541/final.contigs.fa -o
/root/db/PRJNA979234_trimmed/trimmed_outputs/megahit_output/SRR24807541/quast_output/

# 批量处理

#!/bin/bash

# 定义输入目录
input_dir="/root/db/PRJNA868882_trimmed/trimmed_outputs/megahit_output"

# 循环遍历所有 SRR 文件夹
for srr_dir in "$input_dir"/SRR*; do
    # 检查 final.contigs.fa 文件是否存在
    contigs_file="$srr_dir/final.contigs.fa"
    if [[ -f "$contigs_file" ]]; then
        # 创建输出目录
        quast_output_dir="$srr_dir/quast_output"
        mkdir -p "$quast_output_dir"

        # 运行 QUAST
        echo "Running QUAST for $contigs_file"
        quast.py "$contigs_file" -o "$quast_output_dir"
        # quast.py --min-contig 100 "$contigs_file" -o "$quast_output_dir"
    else
        echo "File $contigs_file does not exist, skipping."
    fi
done
```

6.物种组成分析

```
# 加载必要的包
library(dplyr)
library(tidyr)
library(ggplot2)

# 读取ASV表和物种注释表
asv_table <- read.csv("all_asv.csv", row.names = 1, check.names = FALSE)
taxa_table <- read.csv("taxa.print.DECIPHER.csv", row.names = 1, check.names = FALSE)

# 将ASV表转置，使ASV编号成为行名，样本ID成为列名
asv_table_t <- t(asv_table)

# 将ASV表和物种注释表合并
data <- as.data.frame(asv_table_t)
data <- cbind(data, taxa_table)

# 聚合到门水平 使用 across() 时，确保只对数值型列应用 sum() 函数
data_phylum <- data %>%
  group_by(phylum) %>%
  summarise(across(where(is.numeric), sum, na.rm = TRUE))

# 转换为长格式
data_phylum_long <- data_phylum %>%
  pivot_longer(cols = -phylum, names_to = "Sample", values_to = "Abundance")

# 绘制门水平的堆叠柱状图
ggplot(data_phylum_long, aes(x = Sample, y = Abundance, fill = phylum)) +
  geom_bar(stat = "identity", position = "fill") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  labs(x = "Sample", y = "Abundance (%)", fill = "Phylum")
```





7. 群落α、β多样性分析

用R做群落α、β多样性分析_alpha beta多样性分析-CSDN博客

基于R语言的微生物群落组成多样性分析——物种丰度计算及可视化 - 知乎

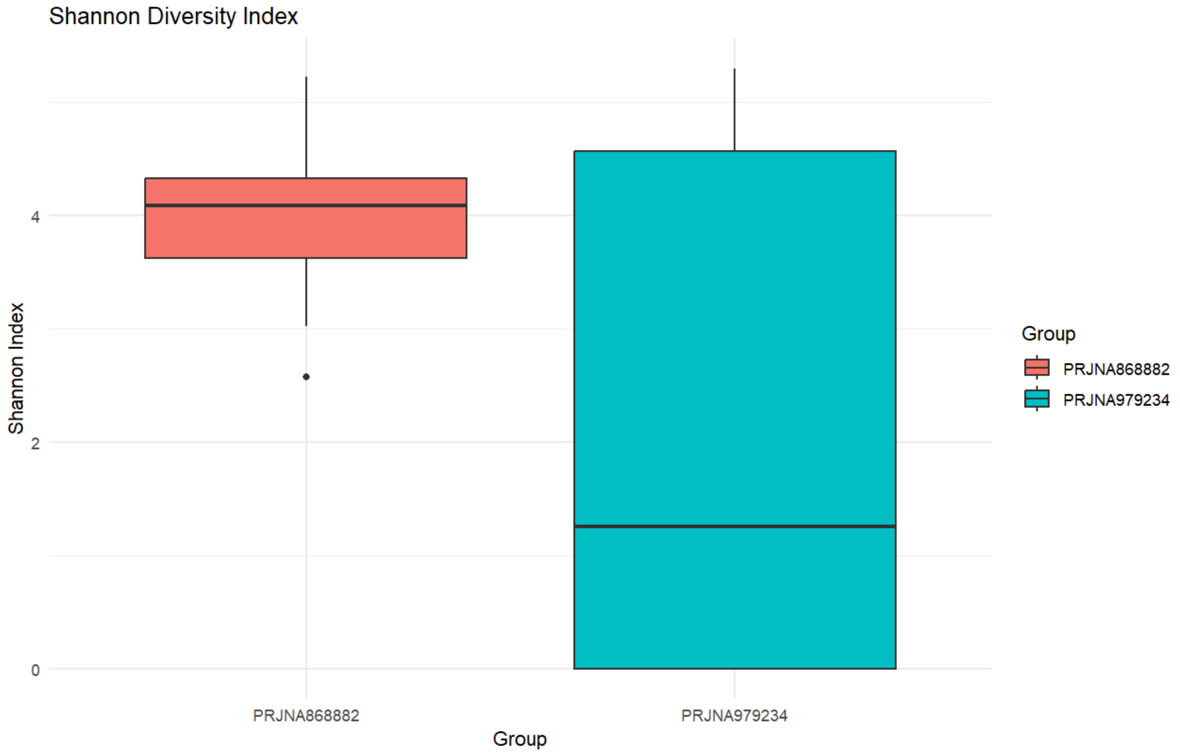
7.1 合并两组数据

```
# 读取PRJNA979234的ASV表
asv_table_PRJNA979234 <- read.csv("all_asv.csv", row.names = 1, check.names =
FALSE)

# 读取PRJNA868882的ASV表
asv_table_PRJNA868882 <- read.csv("all_asv_2.csv", row.names = 1, check.names =
FALSE)



# 合并ASV表
asv_table_combined <- merge(asv_table_PRJNA868882, asv_table_PRJNA979234, by =
"row.names", all = TRUE)

# 将行名设置为基因序列
rownames(asv_table_combined) <- asv_table_combined$Row.names
asv_table_combined$Row.names <- NULL

# 将缺失值替换为0
asv_table_combined[is.na(asv_table_combined)] <- 0
write.csv(asv_table_combined, "ASV.csv", row.names = TRUE)
```

特性	Alpha多样性分析	Beta多样性分析
定义	描述单个样本内的物种多样性，包括物种丰富度（如Chao1、ACE）和均匀度（如Shannon、Simpson）	描述不同样本间的物种组成差异，反映群落结构在空间或时间上的变化
关注点	关注样本内部的物种多样性，揭示样本的复杂性和稳定性	关注样本之间的差异，探究环境因素对群落结构的影响
常用指标	Chao1、ACE、Shannon、Simpson、PD（Phylogenetic Diversity）	Bray-Curtis距离、Jaccard距离、Unweighted UniFrac、Weighted UniFrac
分析方法	箱线图、柱状图展示多样性指数，结合统计检验（如t检验、Kruskal-Wallis检验）	主坐标分析（PCoA）、非度量多维尺度分析（NMDS）、聚类分析（如UPGMA）、差异分析（如PERMANOVA、ANOSIM）
可视化	箱线图、柱状图、稀疏性曲线（Rarefaction Curve）	二维散点图（如PCoA图、NMDS图），可添加置信度椭圆、质心连线等辅助图形
应用场景	评估单个样本的物种丰富度和均匀度，比较不同处理或生境下的样本内部多样性	比较不同样本间的微生物群落组成差异，分析群落结构的变化趋势

7.2 α多样性分析

```r
# 读取ASV表
asv_table <- read.csv("ASV.csv", row.names = 1, check.names = FALSE)

# 读取物种注释表
sample_groups <- read.csv("sample_groups.csv", row.names = 1, check.names =
FALSE)

# 转置ASV表
library(vegan)
asv_table_t <- t(asv_table)
# 计算Shannon多样性指数
shannon_index <- diversity(asv_table_t, index = "shannon", MARGIN = 2)

# 计算Simpson多样性指数
simpson_index <- diversity(asv_table_t, index = "simpson", MARGIN = 2)

# 计算物种丰富度（OTU数量）
richness <- specnumber(asv_table_t, MARGIN = 2)

# 创建α多样性数据框
alpha_diversity <- data.frame(
  Sample = rownames(asv_table),
  Shannon = shannon_index,
  Simpson = simpson_index,
  Richness = richness
)


#合并样本分组信息
alpha_diversity <- merge(alpha_diversity, sample_groups, by.x = "Sample", by.y =
"SampleID")
library(ggplot2)

# 绘制Shannon多样性指数的箱线图
ggplot(alpha_diversity, aes(x = Group, y = Shannon, fill = Group)) +
  geom_boxplot() +
  labs(title = "Shannon Diversity Index", x = "Group", y = "Shannon Index") +
  theme_minimal()
```



- 箱子的中线表示样本多样性指数的中位数。
- 箱子的上下边缘分别代表第二和第三四分位数，显示数据的中间50%分布范围。
- 箱子外的“须”表示数据的范围，通常延伸到1.5倍的四分位距之外。
- 图中PRJNA868882组的多样性指数中位数略高于PRJNA979234组，且分布较为集中。
- PRJNA979234组的多样性指数分布范围更广，且中位数略低。

7.3 beta分析

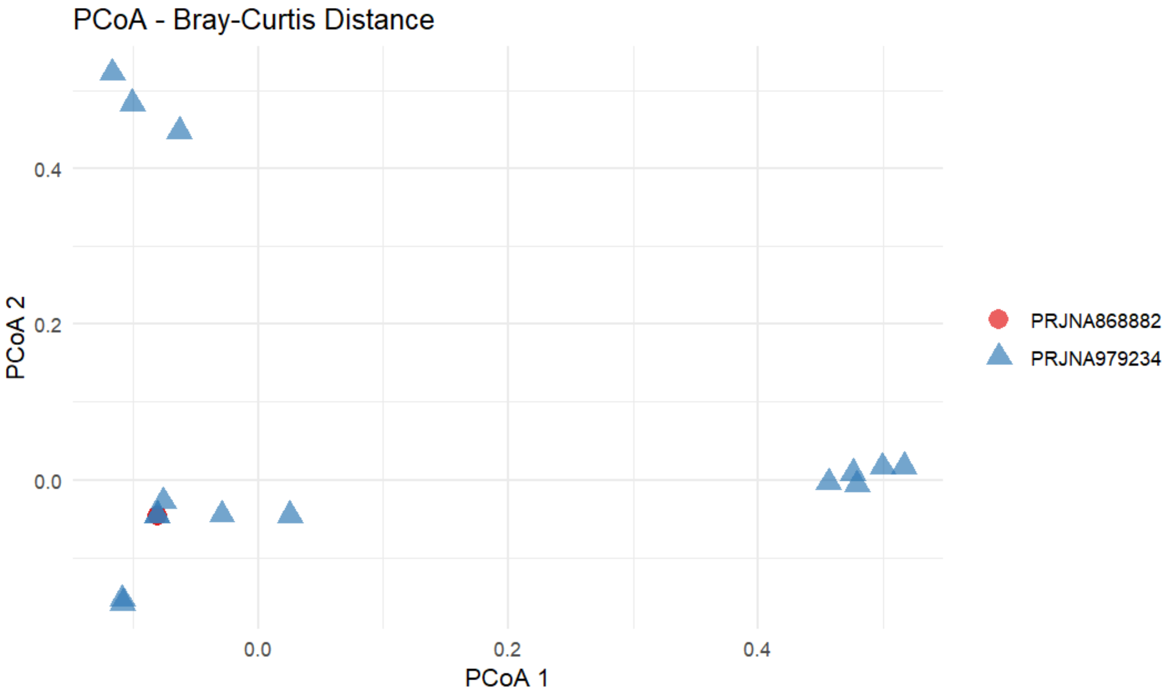
```
# 安装并加载必要的包
if (!requireNamespace("BiocManager", quietly = TRUE)) {
  install.packages("BiocManager")
}
BiocManager::install(c("phyloseq", "vegan"))

# 加载包
library(phyloseq)
library(vegan)
library(ggforce)
library(ggalt)

# 读取ASV表
asv_data <- read.csv("ASV.csv", row.names = 1, check.names = FALSE)

# 读取物种注释表
tax_data <- read.csv("taxa.print.DECIPHER_all.csv", row.names = 1, check.names =
FALSE)

# 创建OTU表
otu_table <- otu_table(asv_data, taxa_are_rows = FALSE)

# 样本数据
sample_data <- read.csv("sample_groups.csv", row.names = 1, check.names = FALSE)
```

```r
sample_data$Group <- as.factor(sample_data$Group)

# 创建分类信息
tax_table <- tax_table(tax_data)
otu_table <- otu_table(asv_data, taxa_are_rows = FALSE)

# # 创建phyloseq对象
ps <- phyloseq(otu_table,
               sample_data(sample_data),
               tax_table)
# 去除空样本
non_empty_samples <- rowSums(otu_table(ps)) > 0
ps <- prune_samples(non_empty_samples, ps)

# 计算Bray-Curtis距离矩阵
# bray_dist <- distance(ps, "bray")
bray_dist <- vegdist(otu_table(ps), method = "bray")

# 计算Jaccard距离矩阵
jaccard_dist <- vegdist(otu_table(ps), method = "jaccard")


# PERMANOVA分析（Bray-Curtis距离）
sample_data_df <- data.frame(sample_data(ps))
sample_data_df$Group <- sample_data(ps)$Group

permanova_result <- adonis2(bray_dist ~ Group, data = sample_data_df,
permutations = 999)
# print(permanova_result)

sample_data(ps)$Group <- as.factor(sample_data(ps)$Group)

# 绘制PCoA图
pcoa_result <- ordinate(ps, method = "PCoA", distance = "bray")
pcoa_result$Group <- sample_data(ps)$Group

pcoa_data <- as.data.frame(pcoa_result$vectors)  # 提取样本点的坐标
colnames(pcoa_data) <- c("PC1", "PC2")
pcoa_data$Group <- sample_data(ps)$Group  # 添加分组信息

pcoa_plot <- ggplot(data = pcoa_data, aes(x = PC1, y = PC2, color = Group, shape
= Group)) +
  geom_point(size = 4, alpha = 0.7) +
  theme_minimal() +
  labs(title = "PCoA - Bray-Curtis Distance",
       x = "PCoA 1",
       y = "PCoA 2") +
  scale_color_brewer(palette = "Set1") +
  theme(legend.title = element_blank(), legend.position = "right")

print(pcoa_plot)


# 提取NMDS坐标数据
nmds_result <- metaMDS(bray_dist, k = 10, trymax = 100)
nmds_data <- as.data.frame(scores(nmds_result))  # 提取样本点的坐标
```

```
colnames(nmds_data) <- c("NMDS1", "NMDS2")  # 手动将列名改为NMDS1和NMDS2
nmds_data$Group<- sample_data(ps)$Group  # 添加分组信息

# 绘制NMDS图
nmds_plot <- ggplot(data = nmds_data, aes(x = NMDS1, y = NMDS2, color = Group,
shape = Group)) +
  geom_point(size = 4, alpha = 0.7) +
  theme_minimal() +
  labs(title = "NMDS - Bray-Curtis Distance",
       x = "NMDS 1",
       y = "NMDS 2") +
  scale_color_brewer(palette = "Set1") +
  theme(legend.title = element_blank(), legend.position = "right")

# 添加置信椭圆
nmds_plot <- nmds_plot +
  geom_encircle(data = nmds_data, aes(group = Group), color = "black", size =
0.5, alpha = 0.3)

print(nmds_plot)

# 保存距离矩阵
write.table(as.matrix(bray_dist), "bray_dist_matrix.txt", sep = "\t", quote =
FALSE)
write.table(as.matrix(jaccard_dist), "jaccard_dist_matrix.txt", sep = "\t", quote
= FALSE)

# 保存PERMANOVA结果
write.table(permanova_result, "permanova_result.txt", sep = "\t", quote = FALSE,
row.names = FALSE)
```

7.3.1 PCoA图

> pcoa_result <- ordinate(ps, method = "PCoA", distance = "bray")

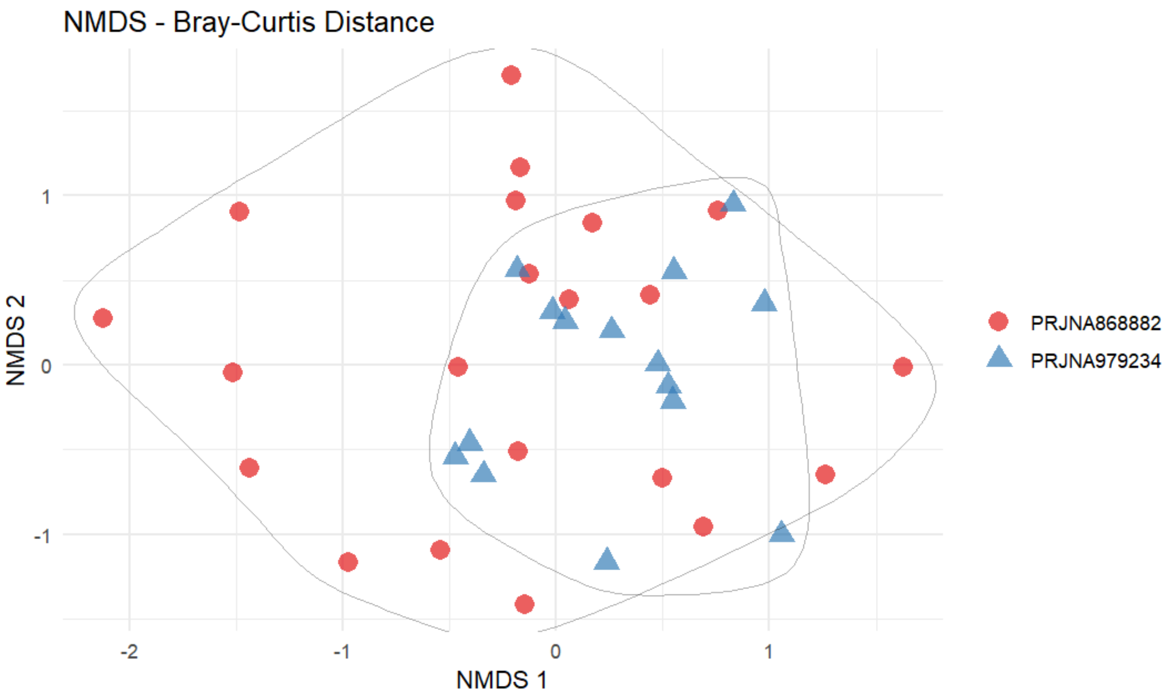


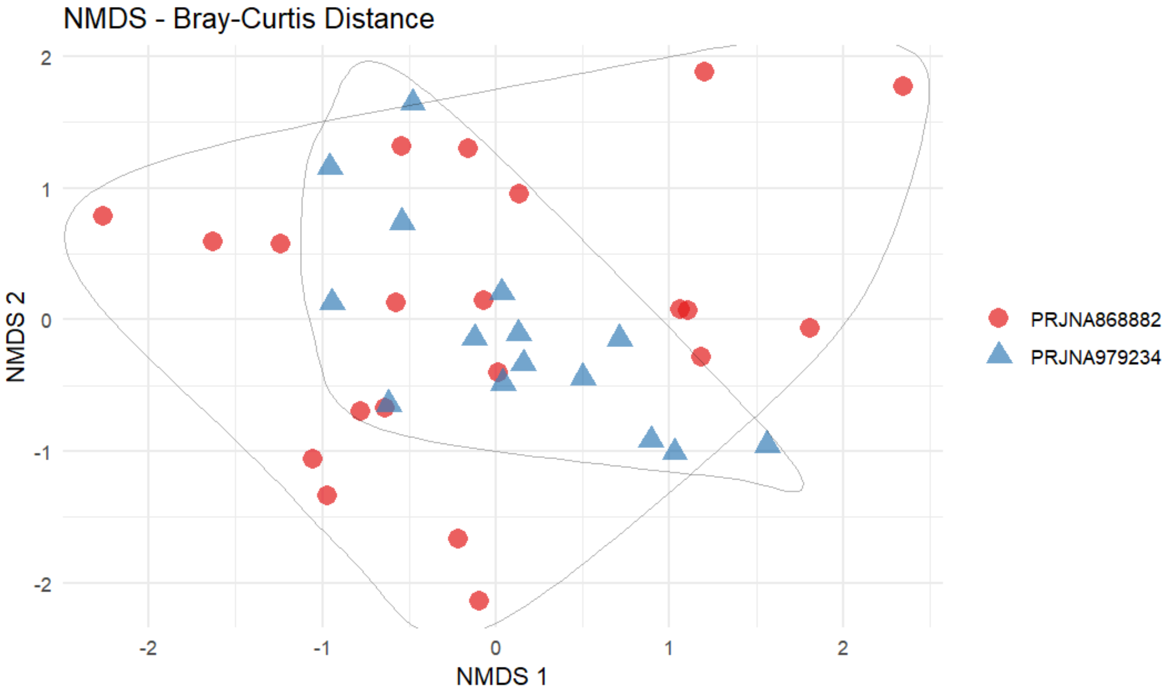
- **轴说明**：图中的两个轴分别是第一主坐标轴（PCoA 1）和第二主坐标轴（PCoA 2）。这两个轴解释了数据中最大的方差部分。
- **样本分布**：图中的点代表不同的样本，颜色和形状区分不同的组。从图中可以看出，两个组（PRJNA868882 和 PRJNA979234）在PCoA 1和PCoA 2上的分布有一定的重叠，但也有明显的分离。
- **解释方差**：PCoA 1和PCoA 2轴解释了数据中多少方差可以通过这两个图观察到。通常，PCoA图会显示每个轴解释的方差百分比，这有助于理解每个轴的重要性。

7.3.2 NMDS图

> nmds_result <- metaMDS(bray_dist, k = 10, trymax = 100)

注意：该案例参数下不收敛





- **轴说明**：图中的两个轴分别是第一NMDS轴（NMDS 1）和第二NMDS轴（NMDS 2）。与PCoA类似，这两个轴也解释了数据中最大的方差部分。

- **样本分布**：图中的点同样代表不同的样本，颜色和形状区分不同的组。NMDS图显示了样本在这两个轴上的分布，并且通过置信椭圆展示了每个组的分布范围。
- **组间差异**：从图中可以看出，PRJNA868882组（红色圆点）和PRJNA979234组（蓝色三角形）在NMDS图上的分布有一定的分离，但也存在一些重叠区域。这表明两组之间存在一定的差异，但这种差异并不是完全的。
- **置信椭圆**：置信椭圆表示每个组样本分布的置信区间，通常为95%。椭圆的大小和形状可以反映组内样本的变异程度和分布情况。

7.3.3 PERMANOVA

Df	SumOfSqs	R2	F	Pr(>F)
1	0.792383394182565	0.0458259385845115	1.63291161946074	0.001
34	16.4987713242585	0.954174061415488	NA	NA
35	17.291154718441	1	NA	NA

输出解释

- **Df (Degrees of Freedom)**:
  - **1**: 表示分组因素（例如不同的处理或条件）的自由度。在这个案例中，`Df`为1意味着有一个分组因素。
  - **34**: 表示在考虑分组因素后剩余的自由度，通常与样本数量有关。
  - **35**: 总自由度，是分组自由度和残差自由度之和。
- **SumOfSqs (Sum of Squares)**:
  - **0.792383394182565**: 分组因素解释的总变异量。
  - **16.4987713242585**: 残差（未被分组因素解释的变异）的平方和。
  - **17.291154718441**: 总平方和，是分组平方和与残差平方和之和。
- **R2 (Coefficient of Determination)**:
  - **0.0458259385845115**: 分组因素解释的变异占总变异的比例。这意味着分组因素解释了大约4.58%的总变异。
  - **0.954174061415488**: 残差解释的变异占总变异的比例，接近95.42%，表明大部分变异未被分组因素解释。
- **F (F-statistic)**:
  - **1.63291161946074**: F统计量，用于检验分组因素是否有统计显著性。F值越大，说明分组因素对变异的解释能力越强。
  - **NA**: 表示未计算或不适用的F值，可能是因为模型中只有一个预测变量或数据结构导致的。
- **Pr(>F) (P-value)**:
  - **0.001**: P值远小于0.05，表明分组因素对样本间差异有显著的解释能力。
  - **NA**: 表示未计算或不适用的P值。

解读

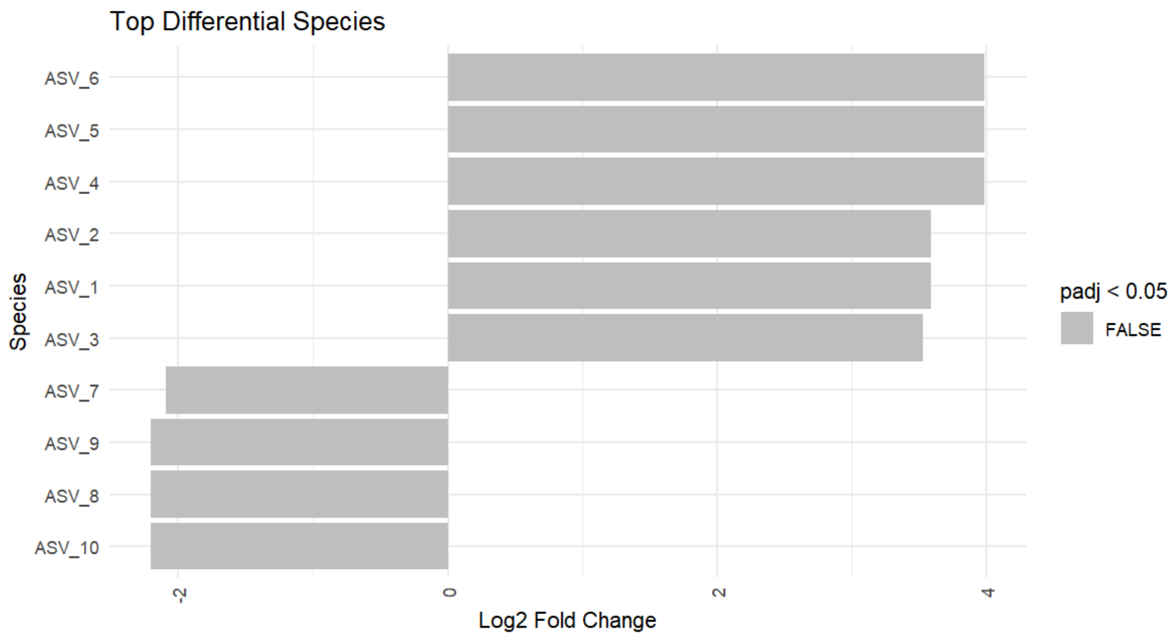
- 分组因素对样本间差异有显著影响（P值为0.001），但影响程度相对较小（R2为0.0458）。
- 大部分变异未被分组因素解释（R2为0.9542），可能还有其他未考虑的因素对样本间差异有更大的影响。

- F值未给出可能是因为模型中只有一个预测变量，这种情况下F统计量可能不适用或者没有被计算。

8.物种差异分析

```
# 安装和加载必要的R包
if (!requireNamespace("BiocManager", quietly = TRUE))
  install.packages("BiocManager")
BiocManager::install("DESeq2")
library(DESeq2)

# 安装和加载ggplot2包
if (!requireNamespace("ggplot2", quietly = TRUE))
  install.packages("ggplot2")
library(ggplot2)

# 读取ASV表和样本分组信息
asv_table <- read.csv("all_ASV2.CSV", row.names = 1, check.names = FALSE)
sample_groups <- read.csv("sample_groups868882.csv", row.names = 1)
asv_table_T <- t(asv_table)

# 预过滤数据：保留至少在 5% 样本中计数大于零的基因
keep <- rowSums(asv_table_T > 0) >= floor(0.05 * ncol(asv_table_T))
filtered_asv_table <- asv_table_T[keep, ]

# 转换为 DESeq2 所需的格式
dds <- DESeqDataSetFromMatrix(countData = filtered_asv_table, colData =
sample_groups, design = ~ Group)
# 使用 DESeq2 默认方法估计大小因子
# dds <- estimateSizeFactors(dds)

# 运行DESeq2分析
dds <- DESeq(dds, sfType = "poscounts")
res <- results(dds, contrast=c("Group", "top", "bottom"))

# 查看差异物种
head(res)

# 提取显著性最高的10个差异物种
top_diff_species <- res[order(res$padj), ][1:10, ]
top_diff_species <- as.data.frame(top_diff_species)
top_diff_species$species <- rownames(top_diff_species)

# 绘制差异物种的条形图
# 旋转标签
# 使用编号
top_diff_species$species <- paste("ASV", 1:nrow(top_diff_species), sep = "_")
ggplot(top_diff_species, aes(x = reorder(species, log2FoldChange), y =
log2FoldChange, fill = padj < 0.05)) +
  geom_bar(stat = "identity") +
  geom_text(aes(label = ifelse(padj < 0.05, "*", "")), vjust = 0.5, hjust = -0.2,
size = 5) +
  theme_minimal() +
  # theme(axis.text.x = element_text(angle = 90, hjust = 1, vjust = 0.5)) +
  labs(title = "Top Differential Species", x = "Species", y = "Log2 Fold Change")
+
  scale_fill_manual(values = c("FALSE" = "gray", "TRUE" = "red")) +
  coord_flip()  # 横向条形图
```

> 案例数据使用PRJNA868882



解读： `top`相对于`bottom`的Log2 Fold Change，无显著性差异

9. 群落差异分析

R统计-微生物群落结构差异分析及结果解读-CSDN博客

> 与上面第八节大差不大

10. 生成OTU表 qiime2

16S扩增子Qiime2。从fq，fastqc等格式文件到OTU table（更详细）_silva-138-99-CSDN博客

10.1 准备工作

获得mapping.txt和manifest.txt

> get_mapping.sh
>
> 都列为一个组

```
#!/bin/bash

# 检查是否传入了分组名称
if [ $# -eq 0 ]; then
    echo "请提供分组名称作为参数。例如：./generate_mapping.sh GroupA"
    exit 1
fi

# 获取分组名称
group_name="$1"

# 定义输出文件名
mapping_file="mapping.txt"
```

```
# 确保输出文件不存在（避免覆盖）
if [ -f "$mapping_file" ]; then
    rm "$mapping_file"
fi

# 添加标题行
echo -e "#SampleID\tGroup" >> "$mapping_file"

# 获取当前目录下的所有.fastq文件
file_list=($(ls *.fastq))

# 检查是否有.fastq文件
if [ ${#file_list[@]} -eq 0 ]; then
    echo "当前目录下没有找到任何 .fastq 文件，请检查文件是否存在。"
    exit 1
fi

# 创建一个数组来存储已经处理过的SampleID
processed_samples=()

# 遍历文件列表并写入mapping文件
for file in "${file_list[@]}"; do
    # 提取样本ID（去掉_1.fastq或_2.fastq部分）
    sample_id=$(echo "$file" | sed -E 's/(_[12])\.fastq$//')

    # 检查是否已经处理过这个样本ID
    if [[ ! " ${processed_samples[@]} " =~ " ${sample_id} " ]]; then
        # 写入mapping文件
        echo -e "$sample_id\t$group_name" >> "$mapping_file"
        # 将样本ID添加到已处理数组中
        processed_samples+=("$sample_id")
    fi
done

echo "Mapping文件已生成，结果保存在 $mapping_file"
```

```
#SampleID   Group
SRR21022338 PRJNA868882
SRR21022339 PRJNA868882
```

> get_mapping.sh
>
> group与id一致

```
#!/bin/bash

# 定义mapping文件名
mapping_file="mapping.txt"

# 如果mapping文件已存在，则删除
if [ -f "$mapping_file" ]; then
    rm "$mapping_file"
fi
```

```
# 添加标题行
echo -e "#SampleID\tGroup" >> "$mapping_file"

# 获取当前目录下的所有.fastq文件
file_list=($(ls *.fastq))

# 检查是否有.fastq文件
if [ ${#file_list[@]} -eq 0 ]; then
    echo "当前目录下没有找到任何 .fastq 文件，请检查文件是否存在。"
    exit 1
fi

# 创建一个数组来存储已经处理过的SampleID
processed_samples=()

# 遍历文件列表并写入mapping文件
for file in "${file_list[@]}"; do
    # 提取样本ID（去掉_1.fastq或_2.fastq部分）
    sample_id=$(echo "$file" | sed -E 's/(_[12])\.fastq$//')

    # 检查是否已经处理过这个样本ID
    if [[ ! " ${processed_samples[@]} " =~ " ${sample_id} " ]]; then
        # 写入mapping文件，Group名称与SampleID相同
        echo -e "$sample_id\t$sample_id" >> "$mapping_file"
        # 将样本ID添加到已处理数组中
        processed_samples+=("$sample_id")
    fi
done

echo "Mapping文件已生成：$mapping_file"
```

```
(base) root@sumyee-virtual-machine:~/db/PRJNA868882_trimmed# cat mapping.txt
#SampleID   Group
SRR21022338 SRR21022338
SRR21022339 SRR21022339
```

> get_manifest.sh

```
#!/bin/bash

# 定义输出文件名
output_file="manifest.txt"

# 确保输出文件不存在（避免覆盖）
if [ -f "$output_file" ]; then
    rm "$output_file"
fi
echo -e "sample-id,absolute-filepath,direction" >> "$output_file"

# 获取当前目录下的所有.fastq文件
file_list=($(ls *.fastq))
```

```
# 检查是否有.fastq文件
if [ ${#file_list[@]} -eq 0 ]; then
    echo "当前目录下没有找到任何 .fastq 文件，请检查文件是否存在。"
    exit 1
fi

# 遍历文件列表并写入汇总文件
for file in "${file_list[@]}"; do
    # 提取样本ID
    sample_id=$(echo "$file" | sed -E 's/(_[12])\.fastq$//')

    # 获取文件的绝对路径
    absolute_path=$(realpath "$file")

    # 判断方向
    if [[ "$file" =~ _1\.fastq$ ]]; then
        direction="forward"
    elif [[ "$file" =~ _2\.fastq$ ]]; then
        direction="reverse"
    else
        direction="unknown"
    fi

    # 写入汇总文件（使用逗号分隔）
    echo -e "$sample_id,$absolute_path,$direction" >> "$output_file"
done

echo "SRR信息汇总完成，结果保存在 $output_file"
```

```
sample-id,absolute-filepath,direction
SRR21022338,/root/db/PRJNA868882_trimmed/SRR21022338_1.fastq,forward
SRR21022338,/root/db/PRJNA868882_trimmed/SRR21022338_2.fastq,reverse
```

10.2 获取Artifact 格式

```
(qiime2-metagenome-2024.10) root@sumyee-virtual-machine:~/db/SHOW# qiime tools
import \
--type 'SampleData[PairedEndSequencesWithQuality]' \
--input-path /root/db/SHOW/dna_data/manifest.txt \
--output-path result/paired-end-demux.qza \
--input-format PairedEndFastqManifestPhred33
Imported /root/db/SHOW/dna_data/manifest.txt as PairedEndFastqManifestPhred33 to
result/paired-end-demux.qza
```

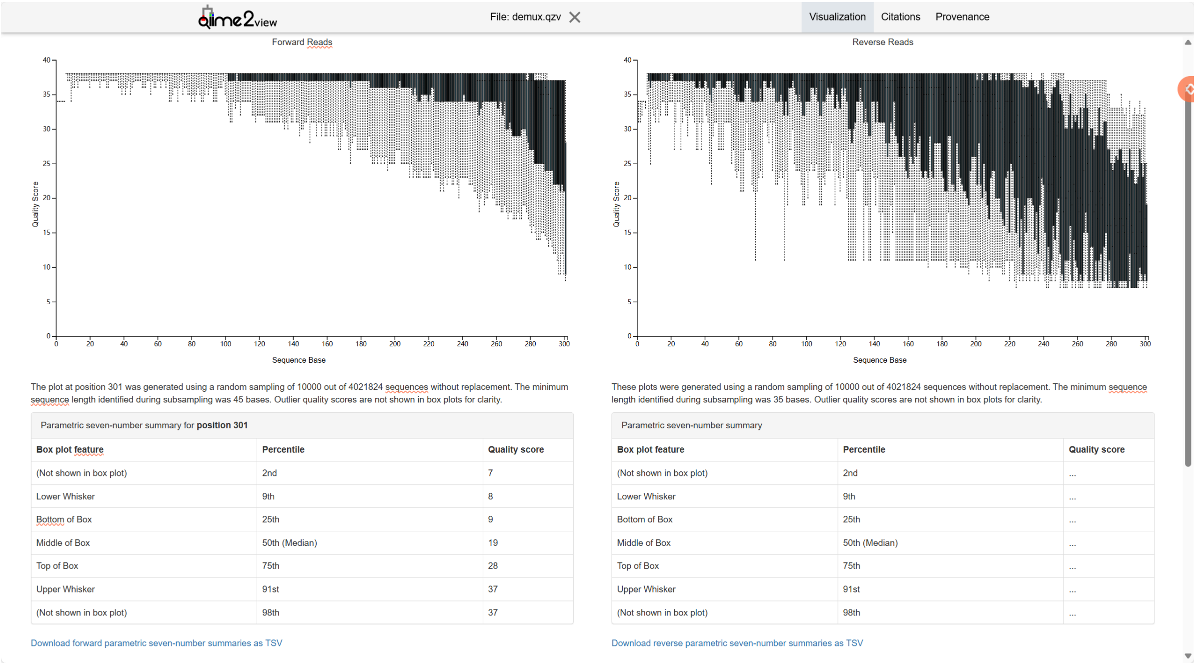
10.3 获取OTU表

view.qiime2.org

创建可视化文件查看数据：demux.qzv

```
(qiime2-metagenome-2024.10) root@sumyee-virtual-machine:~/db/SHOW/result# qiime
demux summarize --i-data paired-end-demux.qza --o-visualization demux.qzv
Saved Visualization to: demux.qzv
```

![微信图片20250216145239](C:\Users\20926\Pictures\mdimage\微信图片20250216145239.png



The plot at position 301 was generated using a random sampling of 10000 out of 4021824 sequences without replacement. The minimum sequence length identified during subsampling was 45 bases. Outlier quality scores are not shown in box plots for clarity.

Parametric seven-number summary for **position 301**

Box plot feature	Percentile	Quality score
(Not shown in box plot)	2nd	7
Lower Whisker	9th	8
Bottom of Box	25th	9
Middle of Box	50th (Median)	19
Top of Box	75th	28
Upper Whisker	91st	37
(Not shown in box plot)	98th	37

Download forward parametric seven-number summaries as TSV

These plots were generated using a random sampling of 10000 out of 4021824 sequences without replacement. The minimum sequence length identified during subsampling was 35 bases. Outlier quality scores are not shown in box plots for clarity.

Parametric seven-number summary

Box plot feature	Percentile	Quality score
(Not shown in box plot)	2nd	...
Lower Whisker	9th	...
Bottom of Box	25th	...
Middle of Box	50th (Median)	...
Top of Box	75th	...
Upper Whisker	91st	...
(Not shown in box plot)	98th	...

Download reverse parametric seven-number summaries as TSV

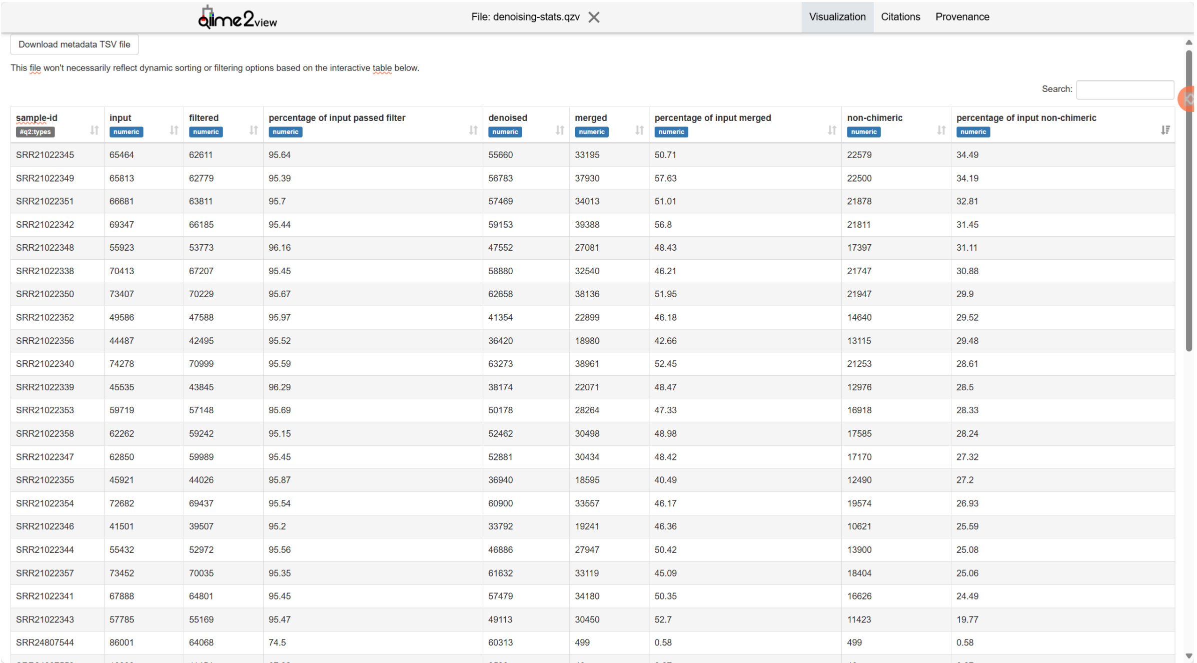
data2降噪、序列质量控制、构建特征表和代表序列表

```
(qiime2-metagenome-2024.10) root@sumyee-virtual-machine:~/db/SHOW/result# time
qiime dada2 denoise-paired \
--i-demultiplexed-seqs paired-end-demux.qza \
--p-trunc-len-f 228 \
--p-trunc-len-r 215 \
--o-table table.qza \
--o-representative-sequences rep-seqs.qza \
--o-denoising-stats denoising-stats.qza


Saved FeatureTable[Frequency] to: table.qza
Saved FeatureData[Sequence] to: rep-seqs.qza
Saved SampleData[DADA2Stats] to: denoising-stats.qza

real    249m41.793s
user    246m0.494s
sys 3m32.231s
```

查看去噪过程统计：denoising-stats.qzv

```
(qiime2-metagenome-2024.10) root@sumyee-virtual-machine:~/db/SHOW/result# qiime
metadata tabulate  --m-input-file denoising-stats.qza --o-visualization
denoising-stats.qzv
Saved Visualization to: denoising-stats.qzv
```

“chimeric”通常指“嵌合的”或“嵌合体”，表示由两个或多个不同来源的遗传物质组合而成的结构。例如，嵌合RNA（chimeric RNA）是指含有来自两个或多个独立基因的外显子序列的RNA分子。这些嵌合分子可以通过多种机制产生，如染色体重排、基因间剪接或转录顺读等。此外，“chimeric”还可以用于描述嵌合抗体、嵌合基因、嵌合质粒等，这些结构都涉及不同来源的遗传物质或蛋白质片段的融合。



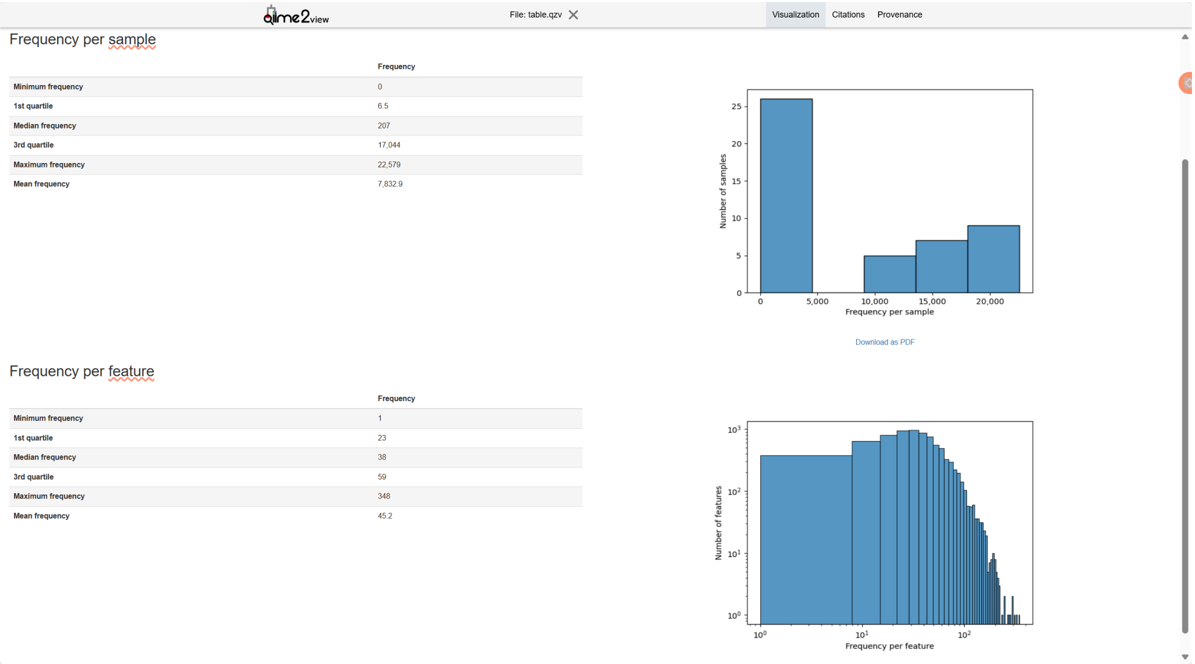
This file won't necessarily reflect dynamic sorting or filtering options based on the interactive table below.

sample-id #q2:types	input numeric	filtered numeric	percentage of input passed filter numeric	denoised numeric	merged numeric	percentage of input merged numeric	non-chimeric numeric	percentage of input non-chimeric numeric
SRR21022345	65464	62611	95.64	55660	33195	50.71	22579	34.49
SRR21022349	65813	62779	95.39	56783	37930	57.63	22500	34.19
SRR21022351	66681	63811	95.7	57469	34013	51.01	21878	32.81
SRR21022342	69347	66185	95.44	59153	39388	56.8	21811	31.45
SRR21022348	55923	53773	96.16	47552	27081	48.43	17397	31.11
SRR21022338	70413	67207	95.45	58880	32540	46.21	21747	30.88
SRR21022350	73407	70229	95.67	62658	38136	51.95	21947	29.9
SRR21022352	49586	47588	95.97	41354	22899	46.18	14640	29.52
SRR21022356	44487	42495	95.52	36420	18980	42.66	13115	29.48
SRR21022340	74278	70999	95.59	63273	38961	52.45	21253	28.61
SRR21022339	45535	43845	96.29	38174	22071	48.47	12976	28.5
SRR21022353	59719	57148	95.69	50178	28264	47.33	16918	28.33
SRR21022358	62262	59242	95.15	52462	30498	48.98	17585	28.24
SRR21022347	62850	59989	95.45	52881	30434	48.42	17170	27.32
SRR21022355	45921	44026	95.87	36940	18595	40.49	12490	27.2
SRR21022354	72682	69437	95.54	60900	33557	46.17	19574	26.93
SRR21022346	41501	39507	95.2	33792	19241	46.36	10621	25.59
SRR21022344	55432	52972	95.56	46886	27947	50.42	13900	25.08
SRR21022357	73452	70035	95.35	61632	33119	45.09	18404	25.06
SRR21022341	67888	64801	95.45	57479	34180	50.35	16626	24.49
SRR21022343	57785	55169	95.47	49113	30450	52.7	11423	19.77
SRR24807544	86001	64068	74.5	60313	499	0.58	499	0.58

查看统计特征表：**table.qzv**

```
(qiime2-metagenome-2024.10) root@sumyee-virtual-machine:~/db/SHOW/result# qiime
feature-table summarize  --i-table table.qza  --o-visualization table.qzv  --m-
sample-metadata-file /root/db/SHOW/dna_data/mapping.txt
Saved Visualization to: table.qzv
```

Summary Statistic	Value
Number of samples	47
Number of features	8,138
Total frequency	368,148



Frequency per sample

	Frequency
Minimum frequency	0
1st quartile	6.5
Median frequency	207
3rd quartile	17,044
Maximum frequency	22,579
Mean frequency	7,832.9

Frequency per feature

	Frequency
Minimum frequency	1
1st quartile	23
Median frequency	38
3rd quartile	59
Maximum frequency	348
Mean frequency	45.2

导出feature table

```
(qiime2-metagenome-2024.10) root@sumyee-virtual-machine:~/db/SHOW/result# qiime
tools export  --input-path table.qza  \
--output-path feature_table
Exported table.qza as BIOMV210DirFmt to directory feature_table
```

```
(qiime2-metagenome-2024.10) root@sumyee-virtual-machine:~/db/SHOW/result# biom
convert -i feature_table/feature-table.biom  -o feature_table/feature-table.txt
--to-tsv
```

计算relative frequency，并导出feature table

```
(qiime2-metagenome-2024.10) root@sumyee-virtual-machine:~/db/SHOW/result# qiime
feature-table relative-frequency --i-table table.qza \
--o-relative-frequency-table table-relative.qza
Saved FeatureTable[RelativeFrequency] to: table-relative.qza
```

```
(qiime2-metagenome-2024.10) root@sumyee-virtual-machine:~/db/SHOW/result# qiime
tools export  --input-path table-relative.qza  \
--output-path feature_table_relative
Exported table-relative.qza as BIOMV210DirFmt to directory feature_table_relative
```

```
(qiime2-metagenome-2024.10) root@sumyee-virtual-machine:~/db/SHOW/result# biom
convert -i feature_table_relative/feature-table.biom  \
-o feature_table_relative/feature-table_relative.txt  --to-tsv
```

10.4 其它记录

[view.qiime2.org/?src=https://docs.qiime2.org/2024.2/data/tutorials/moving-pictures/taxa-bar-plots.qzv]

[Installing QIIME 2 — QIIME 2 2024.10.1 documentation]

[QIIME 2教程. 18使用q2-vsearch聚类ASVs为OTUs(2024.2)-CSDN博客]

```
qiime tools import   --input-path final.contigs.fa   --output-path seqs.qza   --
type 'SampleData[Sequences]'

qiime vsearch dereplicate-sequences   --i-sequences seqs.qza   --o-dereplicated-
table table.qza   --o-dereplicated-sequences rep-seqs.qza

qiime vsearch cluster-features-de-novo   --i-table table.qza   --i-sequences rep-
seqs.qza   --p-perc-identity 0.99   --o-clustered-table table-dn-99.qza   --o-
clustered-sequences rep-seqs-dn-99.qza

```

11. 群落功能预测PICRUSt2

值得一提的是FAPROTA原核生物分类单元功能注释的数据库和工具

[16S的细菌群落功能预测工具PICRUSt2学习_16s数据在线功能预测-CSDN博客]

从 Qiime2 的 `.qza` 文件中导出 OTU/ASV 的代表序列文件（`rep-seqs.qza`）为 FASTA 格式。这将把 `rep-seqs.qza` 文件中的数据导出到一个名为 `exported_rep_seqs` 的文件夹中。导出后，可以在该文件夹中找到一个名为 `dna-sequences.fasta` 的文件。

```
qiime tools export --input-path rep-seqs.qza --output-path exported_rep_seqs
```

```
(picrust2) root@sumyee-virtual-machine:~/db/SHOW/data# picrust2_pipeline.py -s
dna-sequences.fasta -i feature-table.txt -o picrust2_result -p 2 --min_align 0.6
```

min_align对齐相关参数，默认为0.8

在默认参考库中对齐很差

```
Error running this command:
place_seqs.py --study_fasta dna-sequences.fasta --ref_dir
/root/miniconda3/envs/picrust2/lib/python3.9/site-
packages/picrust2/default_files/prokaryotic/pro_ref --out_tree
picrust2_result/out.tre --processes 2 --intermediate
picrust2_result/intermediate/place_seqs --min_align 0.6 --chunk_size 5000 --
placement_tool epa-ng

Standard error of the above failed command:
Warning - 3813 input sequences aligned poorly to reference sequences (--min_align
option specified a minimum proportion of 0.6 aligning to reference sequences).
These input sequences will not be placed and will be excluded from downstream
steps.
```

12.16rDNA扩增子数据分析(VSEARCH)

12.1 合并双端序列

样品查看

```
(qiime2-metagenome-2024.10) root@sumyee-virtual-
machine:~/db/metagenomics/raw_reads# ls
SAA_R1.fastq.gz  SAB_R1.fastq.gz  SAF_R1.fastq.gz  SAG_R1.fastq.gz
SAJ_R1.fastq.gz  SAS_R1.fastq.gz
SAA_R2.fastq.gz  SAB_R2.fastq.gz  SAF_R2.fastq.gz  SAG_R2.fastq.gz
SAJ_R2.fastq.gz  SAS_R2.fastq.gz
```

单个样品

```
(qiime2-metagenome-2024.10) root@sumyee-virtual-machine:~/db/metagenomics#
vsearch --fastq_mergepairs raw_reads/SAA_R1.fastq.gz --reverse
raw_reads/SAA_R2.fastq.gz --fastqout join_pe_reads/SAA.fq
vsearch v2.22.1_linux_x86_64, 7.7GB RAM, 4 cores
https://github.com/torognes/vsearch

Merging reads 100%
     28741  Pairs
     25200  Merged (87.7%)
      3541  Not merged (12.3%)

Pairs that failed merging due to various reasons:
        67  too few kmers found on same diagonal
      1447  too many differences
      2021  alignment score too low, or score drop too high
         6  staggered read pairs

Statistics of all reads:
    237.00  Mean read length

Statistics of merged reads:
    416.94  Mean fragment length
      7.98  Standard deviation of fragment length
      0.71  Mean expected error in forward sequences
      3.26  Mean expected error in reverse sequences
      1.35  Mean expected error in merged sequences
      0.35  Mean observed errors in merged region of forward sequences
      2.98  Mean observed errors in merged region of reverse sequences
      3.33  Mean observed errors in merged region
```

多个样品

```
(qiime2-metagenome-2024.10) root@sumyee-virtual-machine:~/db/metagenomics# cat
run.sh
for i in $(tail -n +2 mapping.txt | cut -f 1);
do
    vsearch --fastq_mergepairs raw_reads/${i}_R1.fastq.gz \
            --reverse raw_reads/${i}_R2.fastq.gz \
            --fastqout join_pe_reads/${i}.fq;
done
```

12.2 序列质量控制

fastqc质量查看

```
(metawrap) root@sumyee-virtual-machine:~/db/metagenomics/join_pe_reads# cat
run.sh
for file in *.fq; do
    fastqc "${file}"
done
```

报告汇总

```
(py310) root@sumyee-virtual-machine:~/db/metagenomics/join_pe_reads# multiqc .

/// MultiQC 🔍 v1.25.2

       file_search | Search path: /root/db/metagenomics/join_pe_reads
         searching | ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 100% 19/19
            fastqc | Found 6 reports
     write_results | Data         : multiqc_data
     write_results | Report       : multiqc_report.html
           multiqc | MultiQC complete
```

在16S rRNA基因测序分析中，V3-V4区是常用的可变区片段，其总长度约为464bp，合并后reads长度<380dp的reads过滤掉。

其中参数--fastqout输出fastq

fastq：

- 存储高通量测序数据，如 Illumina 测序结果。
- 用于质量控制、错误校正、序列组装等分析。

fasta：

- 用于存储和共享序列数据。
- 适用于需要处理序列本身的分析，如比对、聚类、多重序列比对等。

```
(qiime2-metagenome-2024.10) root@sumyee-virtual-machine:~/db/metagenomics#
vsearch --fastx_filter join_pe_reads/SAA.fq --fastq_maxee 1.0 --fastq_minlen 380
--fastaout qual_filter/SAA_filter.fasta
vsearch v2.22.1_linux_x86_64, 7.7GB RAM, 4 cores
https://github.com/torognes/vsearch

Reading input file 100%
14843 sequences kept (of which 0 truncated), 10357 sequences discarded.
```

多样本过滤

```
for i in `tail-n+2 mapping.txt |cut-f 1`; do
    vsearch --fastx_filter join_pe_reads/${i}.fq --fastq_maxee 1.0 --fastq_minlen
380 --fastaout qual_filter/${i}.fasta ;
done
```

12.3 序列去重复

单个样本

```
(qiime2-metagenome-2024.10) root@sumyee-virtual-
machine:~/db/metagenomics/qual_filter# vsearch --derep_uniques SAA.fasta --output
dereplication/SAA_derep.fasta --sizeout --minuniquesize 2
vsearch: unrecognized option '--derep_uniques'
(qiime2-metagenome-2024.10) root@sumyee-virtual-
machine:~/db/metagenomics/qual_filter# vsearch --derep_fulllength SAA.fasta --
output dereplication/SAA_derep.fasta --sizeout --minuniquesize 2
vsearch v2.22.1_linux_x86_64, 7.7GB RAM, 4 cores
https://github.com/torognes/vsearch

Dereplicating file SAA.fasta 100%
6178993 nt in 14843 seqs, min 391, max 424, avg 416
Sorting 100%
```

文件结构

```
(qiime2-metagenome-2024.10) root@sumyee-virtual-
machine:~/db/metagenomics/qual_filter# ls
dereplication  SAA.fasta  SAB.fasta  SAF.fasta  SAG.fasta  SAJ.fasta  SAS.fasta
```

多个样本

```bash
#!/bin/bash

# 定义输入文件夹和输出文件夹
input_folder="."
output_folder="dereplication"

# 检查输出文件夹是否存在，如果不存在则创建
if [ ! -d "$output_folder" ]; then
    mkdir -p "$output_folder"
fi

# 遍历当前目录下的所有 .fasta 文件
for fasta_file in "$input_folder"/*.fasta; do
    if [ -f "$fasta_file" ]; then  # 确保是文件
        # 获取文件名（不包含路径）
        filename=$(basename "$fasta_file")
        # 构建输出文件路径
        output_file="$output_folder/${filename%.fasta}_derep.fasta"

        # 调用 vsearch 进行去重复
        echo "Processing $filename..."
        vsearch --derep_fulllength "$fasta_file" \
                --output "$output_file" \
                --sizeout \
                --minuniquesize 2
    fi
done

echo "All files processed."
```

12.4 嵌合体检测

通过参考数据库序列文件`rdp_glod.fa`，比对去除。`--db` 非必须参数

```
(qiime2-metagenome-2024.10) root@sumyee-virtual-machine:~/db/metagenomics/qual_filter/dereplication# vsearch --uchime_ref SAA_derep.fasta --db rdp_gold.fa --sizein --sizeout --nonchimeras nochimeras/SAA_derep_nochimeras.fa
vsearch v2.22.1_linux_x86_64, 7.7GB RAM, 4 cores
https://github.com/torognes/vsearch

Reading file rdp_gold.fa 100%
29007378 nt in 20098 seqs, min 320, max 2210, avg 1443
Masking 100%
Counting k-mers 100%
Creating k-mer index 100%
Detecting chimeras 100%
Found 77 (9.5%) chimeras, 705 (87.3%) non-chimeras,
and 26 (3.2%) borderline sequences in 808 unique sequences.
Taking abundance information into account, this corresponds to
259 (2.6%) chimeras, 9432 (95.3%) non-chimeras,
and 201 (2.0%) borderline sequences in 9892 total sequences.
```

文件结构

```
(qiime2-metagenome-2024.10) root@sumyee-virtual-machine:~/db/metagenomics/qual_filter/dereplication# ls
nochimeras  rdp_gold.fa  SAA_derep.fasta  SAB_derep.fasta  SAF_derep.fasta
SAG_derep.fasta  SAJ_derep.fasta  SAS_derep.fasta
```

多个样本

```
#!/bin/bash

# 定义输入文件夹和输出文件夹
input_folder="."
output_folder="nochimeras"
db_file="rdp_gold.fa"  # 参考数据库文件

# 检查输出文件夹是否存在，如果不存在则创建
if [ ! -d "$output_folder" ]; then
    mkdir -p "$output_folder"
fi

# 遍历当前目录下的所有 .fasta 文件
for fasta_file in "$input_folder"/*.fasta; do
    if [ -f "$fasta_file" ]; then  # 确保是文件
        # 获取文件名（不包含路径）
        filename=$(basename "$fasta_file")
        # 构建输出文件路径
        output_file="$output_folder/${filename%.fasta}_nochimeras.fa"

        # 调用 vsearch 进行 Chimera 检测
        echo "Processing $filename..."
        vsearch --uchime_ref "$fasta_file" \
                --db "$db_file" \
                --sizein \
                --sizeout \
                --nonchimeras "$output_file"
    fi
done

echo "All files processed."
```

12.5 OTU聚类

以序列最小相似性为97%聚类

```
(qiime2-metagenome-2024.10) root@sumyee-virtual-
machine:~/db/metagenomics/qual_filter/dereplication/nochimeras# vsearch --
cluster_size SAA_derep_nochimeras.fa \
        --id 0.97 \
        --sizein \
        --sizeout \
        --centroids SAA_otus.fa \
        --relabel OTU_
vsearch v2.22.1_linux_x86_64, 7.7GB RAM, 4 cores
https://github.com/torognes/vsearch

Reading file SAA_derep_nochimeras.fa 100%
292966 nt in 705 seqs, min 394, max 423, avg 416
Masking 100%
Sorting by abundance 100%
Counting k-mers 100%
Clustering 100%
Sorting clusters 100%
Writing clusters 100%
Clusters: 80 Size min 2, max 2674, avg 8.8
Singletons: 0, 0.0% of seqs, 0.0% of clusters
```

文件格式

```
(qiime2-metagenome-2024.10) root@sumyee-virtual-
machine:~/db/metagenomics/qual_filter/dereplication/nochimeras# ls
otus  run.sh  SAA_derep_nochimeras.fa  SAB_derep_nochimeras.fa
SAF_derep_nochimeras.fa  SAG_derep_nochimeras.fa  SAJ_derep_nochimeras.fa
SAS_derep_nochimeras.fa
```

处理多个文件

```
#!/bin/bash

# 定义输入文件夹和输出文件夹
input_folder="."  # 当前目录
output_folder="otus"  # 输出文件夹
mkdir -p "$output_folder"  # 创建输出文件夹（如果不存在）
```

```
# 遍历当前目录下的所有 _derep_nochimeras.fa 文件
for file in "$input_folder"/*_derep_nochimeras.fa; do
    if [ -f "$file" ]; then  # 确保是文件
        # 提取样本名称（去掉 _derep_nochimeras.fa 部分）
        sample_name=$(basename "$file" _derep_nochimeras.fa)

        # 构建输出文件路径
        output_fa="$output_folder/${sample_name}.fa"  # OTU 代表序列文件
        output_uc="$output_folder/${sample_name}.uc"  # UCLUST 格式文件

        echo "Processing $sample_name..."
        # 调用 vsearch 进行聚类
        vsearch --cluster_size "$file" \
                --id 0.97 \
                --sizein \
                --sizeout \
                --centroids "$output_fa" \
                --uc "$output_uc" \
                --relabel OTU_
    fi
done

echo "All files processed."
```

12.6 生成OTU表格

```
(qiime2-metagenome-2024.10) root@sumyee-virtual-
machine:~/db/metagenomics/qual_filter/dereplication/nochimeras/otus# vsearch --
usearch_global SAA.fa --db ../SAA_derep_nochimeras.fa --strand plus --id 0.97 --
otutabout OTU/SAA_table.txt
vsearch v2.22.1_linux_x86_64, 7.7GB RAM, 4 cores
https://github.com/torognes/vsearch

Reading file ../SAA_derep_nochimeras.fa 100%
292966 nt in 705 seqs, min 394, max 423, avg 416
Masking 100%
Counting k-mers 100%
Creating k-mer index 100%
Searching 100%
Matching unique query sequences: 80 of 80 (100.00%)
Writing OTU table (classic) 100%
```

文件结构

```
(qiime2-metagenome-2024.10) root@sumyee-virtual-
machine:~/db/metagenomics/qual_filter/dereplication/nochimeras/otus# ls
OTU   run.sh   SAA.fa   SAA.uc   SAB.fa   SAB.uc   SAF.fa   SAF.uc   SAG.fa   SAG.uc
SAJ.fa   SAJ.uc   SAS.fa   SAS.uc
```

处理多个样本

```
#!/bin/bash
```

```
# 定义输入文件夹和输出文件夹
input_folder="."  # 当前目录为 otus 文件夹
derep_folder="../"  # 去嵌合体文件所在的目录
output_folder="OTU"  # 输出 OTU 表的文件夹
mkdir -p "$output_folder"  # 创建输出文件夹（如果不存在）

# 遍历当前目录下的所有 .fa 文件
for file in "$input_folder"/*.fa; do
    if [ -f "$file" ]; then  # 确保是文件
        # 提取样本名称（去掉 .fa 部分）
        sample_name=$(basename "$file" .fa)

        # 构建输出文件路径
        output_table="$output_folder/${sample_name}_table.txt"

        echo "Processing $sample_name..."
        # 调用 vsearch 进行全局比对并生成 OTU 表
        vsearch --usearch_global "$file" \
                --db "$derep_folder/${sample_name}_derep_nochimeras.fa" \
                --strand plus \
                --id 0.97 \
                --otutabout "$output_table"
    fi
done

echo "All files processed."
```

12.7 物种分类注释

```
(qiime2-metagenome-2024.10) root@sumyee-virtual-
machine:~/db/metagenomics/qual_filter/dereplication/nochimeras/otus# java -Xmx1g
-jar /root/utils/rdp_classifier_2.2/rdp_classifier-2.2.jar -q SAA.fa -o
TAX/SAA_table.txt -f fixrank
```

文件格式

```
(qiime2-metagenome-2024.10) root@sumyee-virtual-
machine:~/db/metagenomics/qual_filter/dereplication/nochimeras/otus# ls
OTU  run.sh  SAA.fa  SAA.uc  SAB.fa  SAB.uc  SAF.fa  SAF.uc  SAG.fa  SAG.uc
SAJ.fa  SAJ.uc  SAS.fa  SAS.uc  TAX
```

处理多个文件

```
#!/bin/bash

# 定义输入文件夹和输出文件夹
input_folder="."  # 当前目录
output_folder="TAX"  # 输出文件夹
mkdir -p "$output_folder"  # 确保输出文件夹存在

# RDP 分类器的路径
rdp_classifier="/root/utils/rdp_classifier_2.2/rdp_classifier-2.2.jar"
```

```
# 遍历当前目录下的所有 .fa 文件
for file in "$input_folder"/*.fa; do
    if [ -f "$file" ]; then  # 确保是文件
        # 提取样本名称（去掉 .fa 部分）
        sample_name=$(basename "$file" .fa)

        # 构建输出文件路径
        output_file="$output_folder/${sample_name}_table.txt"

        echo "Processing $sample_name..."
        # 调用 RDP 分类器
        java -Xmx1g -jar "$rdp_classifier" -q "$file" -o "$output_file" -f
fixrank
    fi
done

echo "All files processed."
```

13. 16s扩增子（qiime2）

13.1 数据导入到qiime2

获取`manifest.txt`文件

```
#!/bin/bash

# 定义输出文件名
output_file="manifest.txt"

# 确保输出文件不存在（避免覆盖）
if [ -f "$output_file" ]; then
    rm "$output_file"
fi
echo -e "sample-id,absolute-filepath,direction" >> "$output_file"

# 获取当前目录下的所有.fastq文件
file_list=($(ls *fastq.gz))

# 检查是否有.fastq文件
if [ ${#file_list[@]} -eq 0 ]; then
    echo "当前目录下没有找到任何 .fastq 文件，请检查文件是否存在。"
    exit 1
fi

# 遍历文件列表并写入汇总文件
for file in "${file_list[@]}"; do
    # 提取样本ID
    sample_id=$(echo "$file" | sed -E 's/(_[12])\.fastq.gz$//')

    # 获取文件的绝对路径
    absolute_path=$(realpath "$file")

    # 判断方向
    if [[ "$file" =~ _R1\.fastq.gz$ ]]; then
        direction="forward"
    elif [[ "$file" =~ _R2\.fastq.gz$ ]]; then
        direction="reverse"
    else
        direction="unknown"
    fi

    # 写入汇总文件（使用逗号分隔）
    echo -e "$sample_id,$absolute_path,$direction" >> "$output_file"
done

echo "SRR信息汇总完成，结果保存在 $output_file"
```

查看

```
(qiime2-amplicon-2024.10) root@sumyee-virtual-
machine:~/db/metagenomics/raw_reads# cat manifest.txt
sample-id,absolute-filepath,direction
SAA_R1.fastq.gz,/root/db/metagenomics/raw_reads/SAA_R1.fastq.gz,forward
SAA_R2.fastq.gz,/root/db/metagenomics/raw_reads/SAA_R2.fastq.gz,reverse
SAB_R1.fastq.gz,/root/db/metagenomics/raw_reads/SAB_R1.fastq.gz,forward
SAB_R2.fastq.gz,/root/db/metagenomics/raw_reads/SAB_R2.fastq.gz,reverse
SAF_R1.fastq.gz,/root/db/metagenomics/raw_reads/SAF_R1.fastq.gz,forward
SAF_R2.fastq.gz,/root/db/metagenomics/raw_reads/SAF_R2.fastq.gz,reverse
SAG_R1.fastq.gz,/root/db/metagenomics/raw_reads/SAG_R1.fastq.gz,forward
SAG_R2.fastq.gz,/root/db/metagenomics/raw_reads/SAG_R2.fastq.gz,reverse
SAJ_R1.fastq.gz,/root/db/metagenomics/raw_reads/SAJ_R1.fastq.gz,forward
SAJ_R2.fastq.gz,/root/db/metagenomics/raw_reads/SAJ_R2.fastq.gz,reverse
SAS_R1.fastq.gz,/root/db/metagenomics/raw_reads/SAS_R1.fastq.gz,forward
SAS_R2.fastq.gz,/root/db/metagenomics/raw_reads/SAS_R2.fastq.gz,reverse
```

导入数据

参考 QIIME2进阶二_元数据及数据导入QIIME2_qiime tools import-CSDN博客

单端时

```
time qiime tools import
--type"SampleData[SequencesWithQuality]"
--input-format SingleEndFastqManifestPhred33V2
--input-path manifest.tsv
--output-path demux_seqs.qza
```

双端时（本案例）

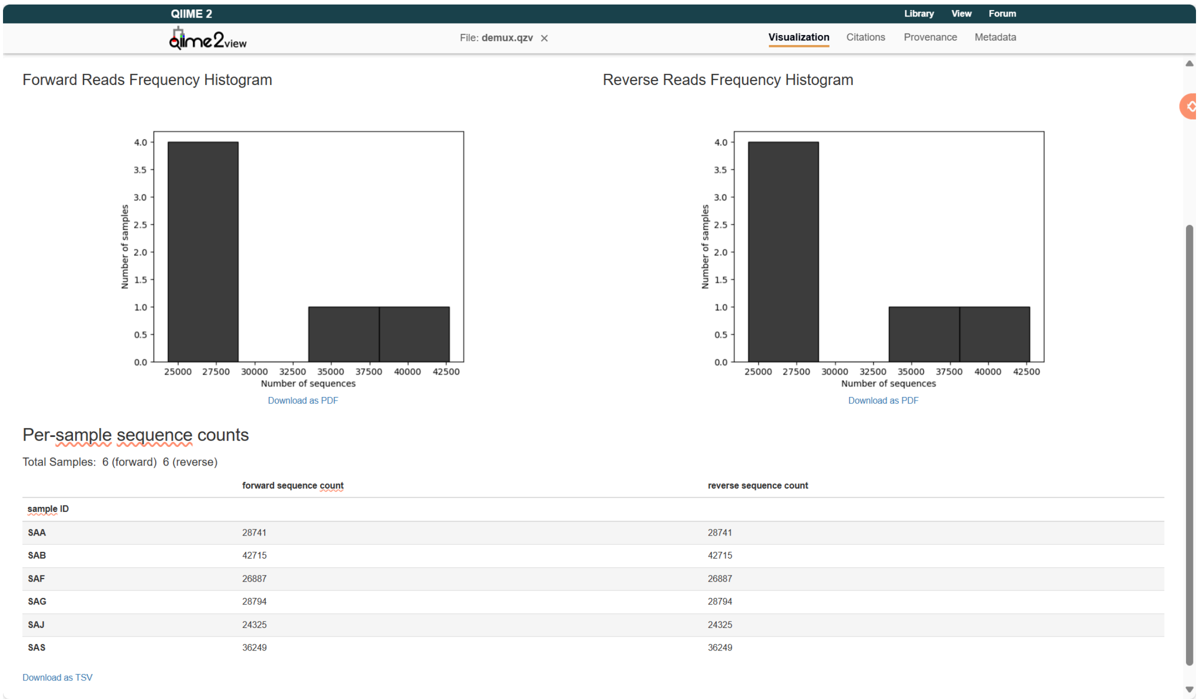
```
(qiime2-amplicon-2024.10) root@sumyee-virtual-machine:~/db/metagenomics/raw_reads# time qiime tools import --type 'SampleData[PairedEndSequencesWithQuality]' --input-path /root/db/metagenomics/raw_reads/manifest.txt --output-path result/paired-end-demux.qza --input-format PairedEndFastqManifestPhred33
Imported /root/db/metagenomics/raw_reads/manifest.txt as PairedEndFastqManifestPhred33 to result/paired-end-demux.qza

real    0m11.164s
user    0m9.647s
sys 0m1.720s
```

检查样本的序列和测序深度

```
(qiime2-amplicon-2024.10) root@sumyee-virtual-machine:~/db/metagenomics/raw_reads/result# qiime demux summarize --i-data paired-end-demux.qza --o-visualization demux.qzv
Saved Visualization to: demux.qzv
```

查看文件结构

```
(qiime2-amplicon-2024.10) root@sumyee-virtual-machine:~/db/metagenomics/raw_reads# ls
get_manifest.sh  result           SAA_R2.fastq.gz  SAB_R2.fastq.gz
SAF_R2.fastq.gz  SAG_R2.fastq.gz  SAJ_R2.fastq.gz  SAS_R2.fastq.gz
manifest.txt     SAA_R1.fastq.gz  SAB_R1.fastq.gz  SAF_R1.fastq.gz
SAG_R1.fastq.gz  SAJ_R1.fastq.gz  SAS_R1.fastq.gz
```



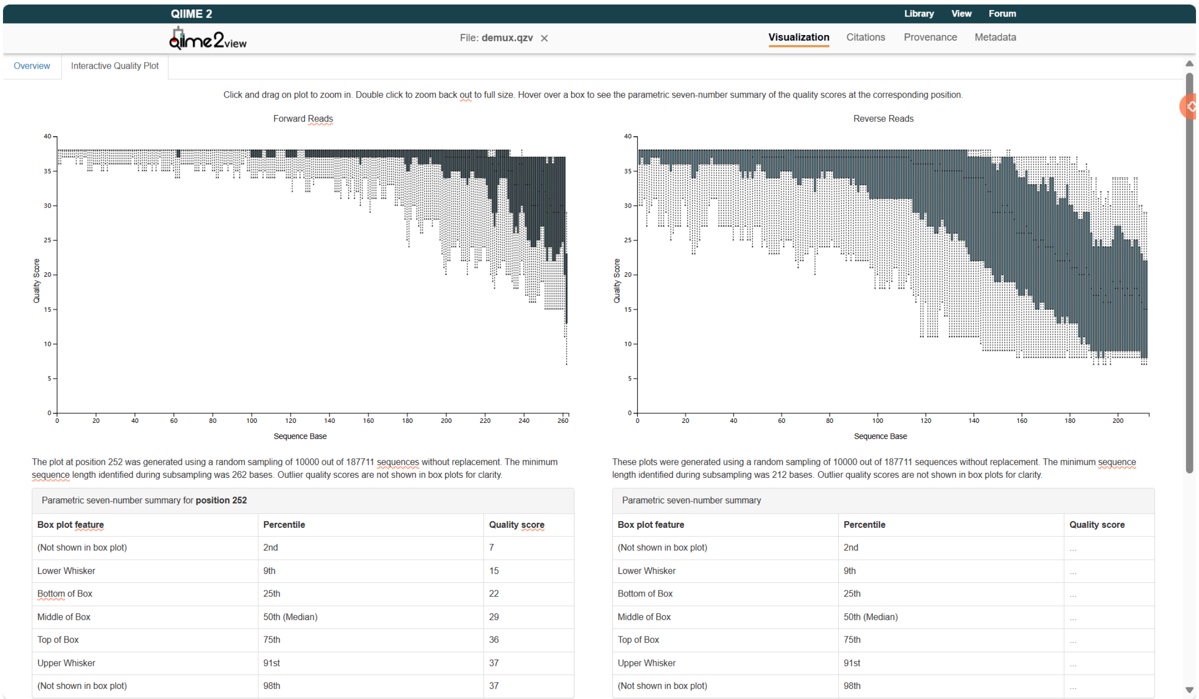
Forward Reads Frequency Histogram

Reverse Reads Frequency Histogram

Per-sample sequence counts

Total Samples: 6 (forward) 6 (reverse)

sample ID	forward sequence count	reverse sequence count
SAA	28741	28741
SAB	42715	42715
SAF	26887	26887
SAG	28794	28794
SAJ	24325	24325
SAS	36249	36249



Click and drag on plot to zoom in. Double click to zoom back out to full size. Hover over a box to see the parametric seven-number summary of the quality scores at the corresponding position.

The plot at position 252 was generated using a random sampling of 10000 out of 187711 sequences without replacement. The minimum sequence length identified during subsampling was 262 bases. Outlier quality scores are not shown in box plots for clarity.

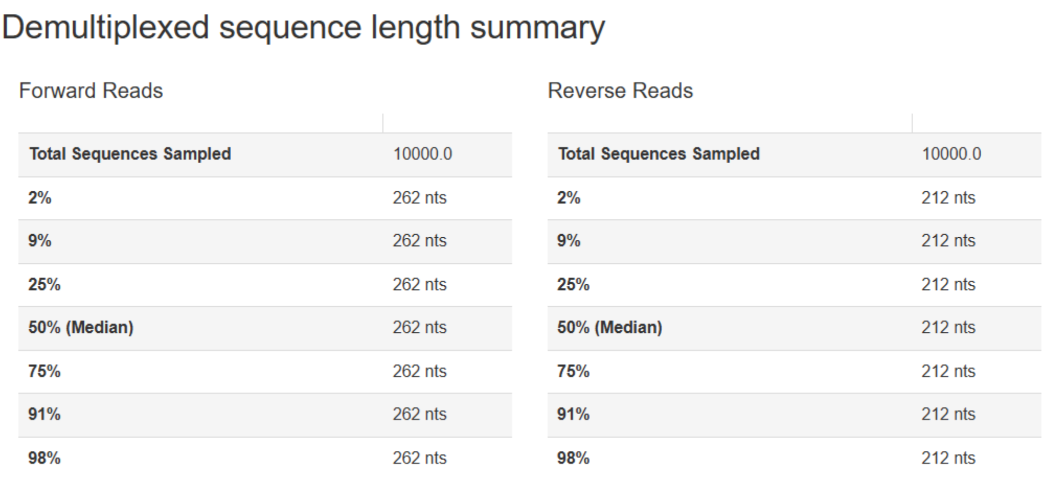
Parametric seven-number summary for **position 252**

Box plot feature	Percentile	Quality score
(Not shown in box plot)	2nd	7
Lower Whisker	9th	15
Bottom of Box	25th	22
Middle of Box	50th (Median)	29
Top of Box	75th	36
Upper Whisker	91st	37
(Not shown in box plot)	98th	37

These plots were generated using a random sampling of 10000 out of 187711 sequences without replacement. The minimum sequence length identified during subsampling was 212 bases. Outlier quality scores are not shown in box plots for clarity.

Parametric seven-number summary

Box plot feature	Percentile	Quality score
(Not shown in box plot)	2nd	...
Lower Whisker	9th	...
Bottom of Box	25th	...
Middle of Box	50th (Median)	...
Top of Box	75th	...
Upper Whisker	91st	...
(Not shown in box plot)	98th	...



Demultiplexed sequence length summary

Forward Reads

Total Sequences Sampled	10000.0
2%	262 nts
9%	262 nts
25%	262 nts
50% (Median)	262 nts
75%	262 nts
91%	262 nts
98%	262 nts

Reverse Reads

Total Sequences Sampled	10000.0
2%	212 nts
9%	212 nts
25%	212 nts
50% (Median)	212 nts
75%	212 nts
91%	212 nts
98%	212 nts

13.2.1 dada2质量控制

dada2进行质量控制前需要先去除引物（primers），接头（adapters or barcodes），linker！！

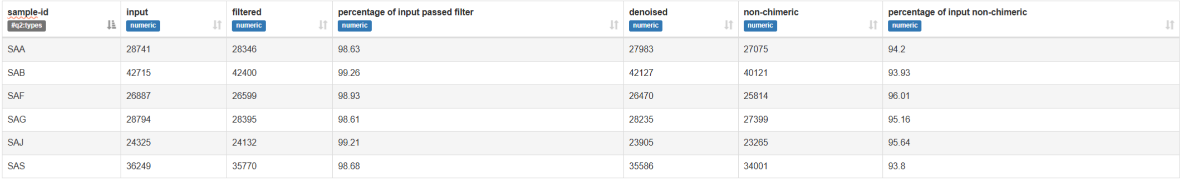
参考 QIIME2进阶三*用QIIME2实现对数据的质量控制*qiime2数据指控-CSDN博客

1）使用DADA2插件进行质量控制

```
(qiime2-amplicon-2024.10) root@sumyee-virtual-machine:~/db/metagenomics/raw_reads/result# time qiime dada2 denoise-single --i-demultiplexed-seqs paired-end-demux.qza --p-trunc-len 150 --o-table dada2_table.qza --o-representative-sequences dada2_rep_set.qza --o-denoising-stats dada2_stats.qza
Saved FeatureTable[Frequency] to: dada2_table.qza
Saved FeatureData[Sequence] to: dada2_rep_set.qza
Saved SampleData[DADA2Stats] to: dada2_stats.qza

real	1m17.824s
user	0m44.676s
sys 0m23.070s
```

2）生成统计结果

```
(qiime2-amplicon-2024.10) root@sumyee-virtual-machine:~/db/metagenomics/raw_reads/result# qiime metadata tabulate --m-input-file dada2_stats.qza --o-visualization dada2_stats.qzv
Saved Visualization to: dada2_stats.qzv
```



sample-id	input	filtered	percentage of input passed filter	denoised	non-chimeric	percentage of input non-chimeric
#q2:types	numeric	numeric	numeric	numeric	numeric	numeric
SAA	28741	28346	98.63	27983	27075	94.2
SAB	42715	42400	99.26	42127	40121	93.93
SAF	26887	26599	98.93	26470	25814	96.01
SAG	28794	28395	98.61	28235	27399	95.16
SAJ	24325	24132	99.21	23905	23265	95.64
SAS	36249	35770	98.68	35586	34001	93.8

3）生成特征表摘要

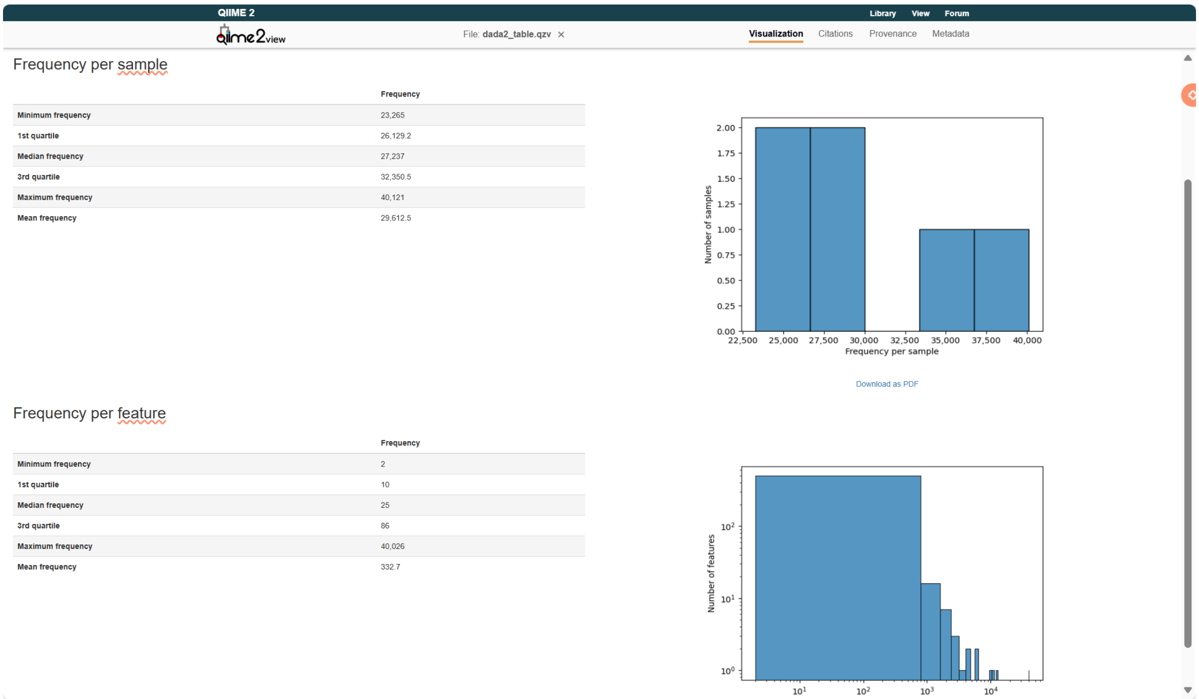
```
(qiime2-amplicon-2024.10) root@sumyee-virtual-machine:~/db/metagenomics/raw_reads/result# qiime feature-table summarize --i-table dada2_table.qza --o-visualization dada2_table.qzv --m-sample-metadata-file /root/db/metagenomics/mapping.txt
/root/miniconda3/envs/qiime2-amplicon-2024.10/lib/python3.10/site-packages/qiime2/metadata/io.py:365: FutureWarning: Downcasting behavior in `replace` is deprecated and will be removed in a future version. To retain the old behavior, explicitly call `result.infer_objects(copy=False)`. To opt-in to the future behavior, set `pd.set_option('future.no_silent_downcasting', True)`
  series = series.replace('', np.nan).infer_objects(copy=False)
Saved Visualization to: dada2_table.qzv
```

查看`mapping.txt`

```
(qiime2-amplicon-2024.10) root@sumyee-virtual-machine:~/db/metagenomics# cat mapping.txt
#SampleID   BarcodeSequence LinkerPrimerSequence    Group   Description
SAA         diet    SAA
SAB         diet    SAB
SAS         diet    SAS
SAF         meat    SAF
SAG         meat    SAG
SAJ         meat    SAJ
```

Table summary

Summary Statistic	Value
Number of samples	6
Number of features	534
Total frequency	177,675

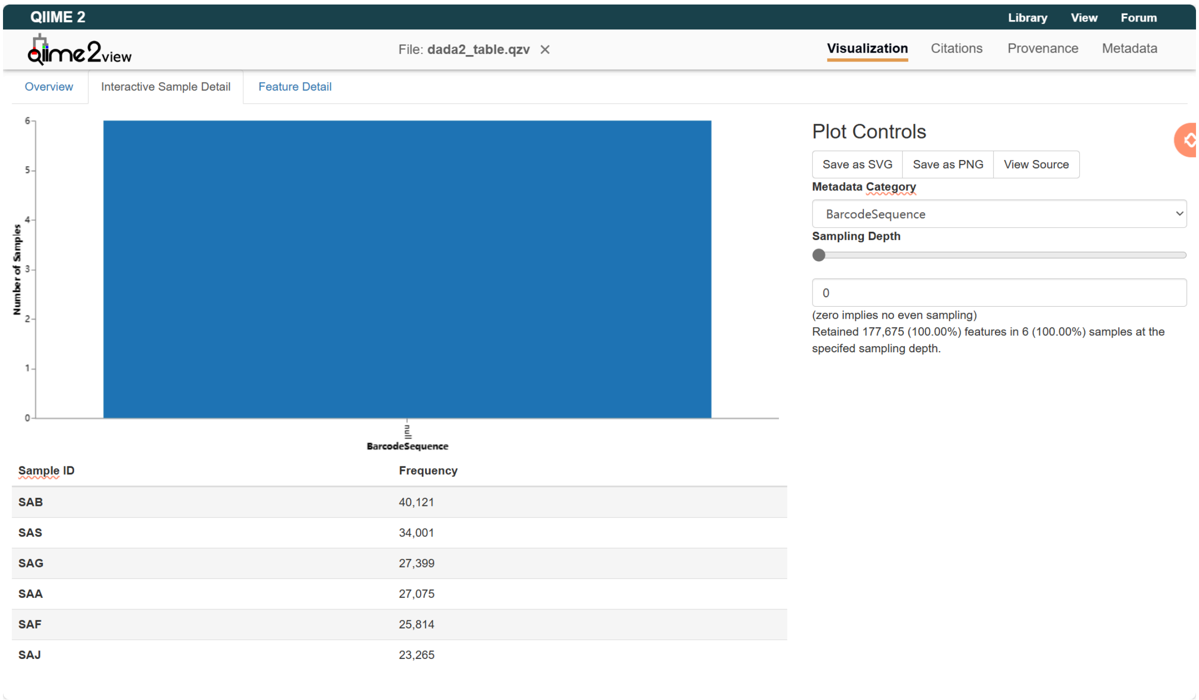


Frequency per sample

	Frequency
Minimum frequency	23,265
1st quartile	26,129.2
Median frequency	27,237
3rd quartile	32,350.5
Maximum frequency	40,121
Mean frequency	29,612.5

Frequency per feature

	Frequency
Minimum frequency	2
1st quartile	10
Median frequency	25
3rd quartile	86
Maximum frequency	40,026
Mean frequency	332.7



Plot Controls

Metadata Category: BarcodeSequence

Sampling Depth: 0

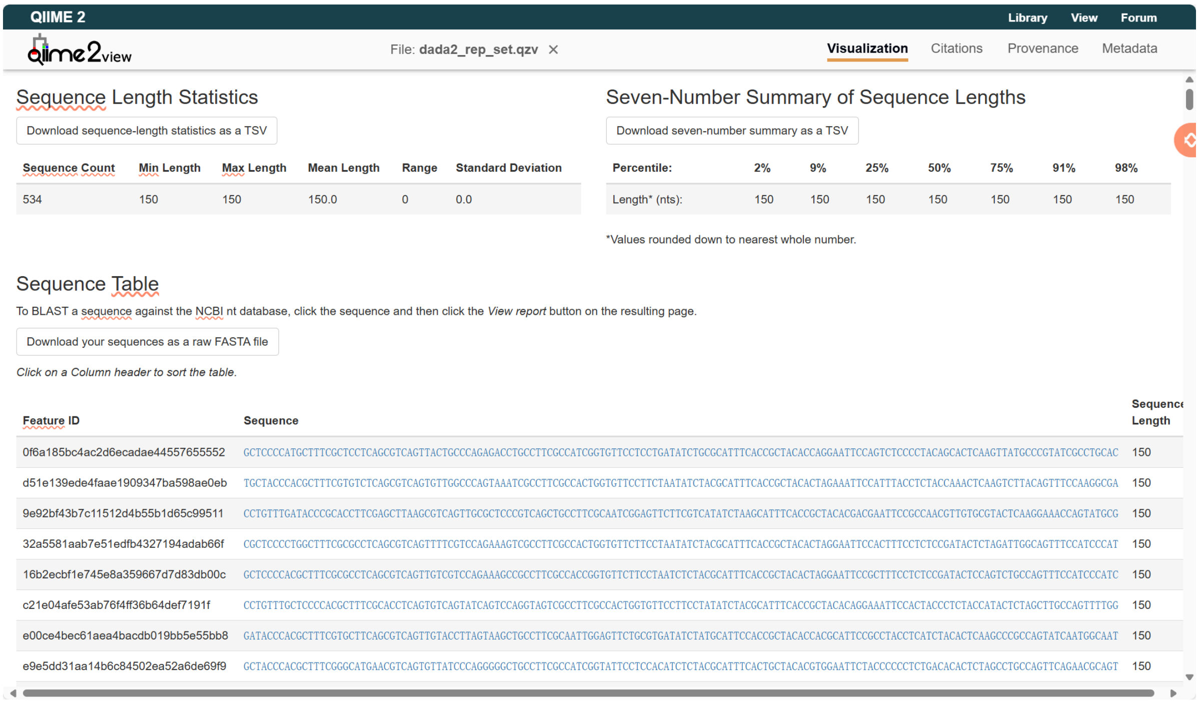
(zero implies no even sampling)
Retained 177,675 (100.00%) features in 6 (100.00%) samples at the specifed sampling depth.

Sample ID	Frequency
SAB	40,121
SAS	34,001
SAG	27,399
SAA	27,075
SAF	25,814
SAJ	23,265

1741852299339

4）生成代表序列摘要：

```
(qiime2-amplicon-2024.10) root@sumyee-virtual-machine:~/db/metagenomics/raw_reads/result# qiime feature-table tabulate-seqs --i-data dada2_rep_set.qza --o-visualization dada2_rep_set.qzv
Saved Visualization to: dada2_rep_set.qzv
```



Sequence Length Statistics

Download sequence-length statistics as a TSV

Sequence Count	Min Length	Max Length	Mean Length	Range	Standard Deviation
534	150	150	150.0	0	0.0

Seven-Number Summary of Sequence Lengths

Download seven-number summary as a TSV

Percentile:	2%	9%	25%	50%	75%	91%	98%
Length* (nts):	150	150	150	150	150	150	150

*Values rounded down to nearest whole number.

Sequence Table

To BLAST a sequence against the NCBI nt database, click the sequence and then click the *View report* button on the resulting page.

Download your sequences as a raw FASTA file

Click on a Column header to sort the table.

Feature ID	Sequence	Sequence Length
0f6a185bc4ac2d6ecadae44557655552	GCTCCCCATGCTTTCGCTCCTCAGCGTCAGTTACTGCCCAGAGACCTGCCTTCGCCATCGGTGTTCCTCCTGATATCTGCGCATTTCACCGCTACACCAGGAATTCCAGTCTCCCCTACAGCACTCAAGTTATGCCCGTATCGCCTGCAC	150
d51e139ede4faae1909347ba598ae0eb	TGCTACCCACGCTTTCGTGTCTCAGCGTCAGTGTTGGCCCAGTAAATCGCCTTCGCCACTGGTGTTCCTTCTAATATCTACGCATTTCACCGCTACACTAGAAATTCCATTTACCTCTACCAAACTCAAGTCTTACAGTTTCCAAGGCGA	150
9e92bf43b7c11512d4b55b1d65c99511	CCTGTTTGATACCCGCACCTTCGAGCTTAAGCGTCAGTTGCGCTCCCGTCAGCTGCCTTCGCAATCGGAGTTCTTCGTCATATCTAAGCATTTCACCGCTACACGACGAATTCCGCCAACGTTGTGCGTACTCAAGGAAACCAGTATGCG	150
32a5581aab7e51edfb4327194adab66f	CGCTCCCCTGGCTTTCGCGCCTCAGCGTCAGTTTTCGTCCAGAAAGTCGCCTTCGCCACTGGTGTTCTTCCTAATATCTACGCATTTCACCGCTACACTAGGAATTCCACTTTCCTCTCCGATACTCTAGATTGGCAGTTTCCATCCCAT	150
16b2ecbf1e745e8a359667d7d83db00c	GCTCCCCACGCTTTCGCGCCTCAGCGTCAGTTGTCGTCCAGAAAGCCGCCTTCGCCACCGGTGTTCTTCCTAATCTCTACGCATTTCACCGCTACACTAGGAATTCCGCTTTCCTCTCCGATACTCCAGTCTGCCAGTTTCCATCCCATC	150
c21e04afe53ab76f4ff36b64def7191f	CCTGTTTGCTCCCCACGCTTTCGCACCTCAGTGTCAGTATCAGTCCAGGTAGTCGCCTTCGCCACTGGTGTTCCTTCCTATATCTACGCATTTCACCGCTACACAGGAAATTCCACTACCCTCTACCATACTCTAGCTTGCCAGTTTTGG	150
e00ce4bec61aea4bacdb019bb5e55bb8	GATACCCACGCTTTCGTGCTTCAGCGTCAGTTGTACCTTAGTAAGCTGCCTTCGCAATTGGAGTTCTGCGTGATATCTATGCATTCCACCGCTACACCACGCATTCCGCCTACCTCATCTACACTCAAGCCCGCCAGTATCAATGGCAAT	150
e9e5dd31aa14b6c84502ea52a6de69f9	GCTACCCACGCTTTCGGGCATGAACGTCAGTGTTATCCCAGGGGGCTGCCTTCGCCATCGGTATTCCTCCACATCTCTACGCATTTCACTGCTACACGTGGAATTCTACCCCCCTCTGACACACTCTAGCCTGCCAGTTCAGAACGCAGT	150

13.2.2 Deblur质量控制

1）按序列碱基质量过滤序列

```
(qiime2-amplicon-2024.10) root@sumyee-virtual-machine:~/db/metagenomics/raw_reads/result# time qiime quality-filter q-score --i-demux paired-end-demux.qza --o-filtered-sequences demux-filtered.qza --o-filter-stats demux-filter-stats.qza
Saved SampleData[SequencesWithQuality] to: demux-filtered.qza
Saved QualityFilterStats to: demux-filter-stats.qza

real	0m41.530s
user	0m39.579s
sys	0m1.135s
```

demux-filtered.qza: 序列质量过滤后结果;

demux-filter-stats.qza: 序列质量过滤后结果统计。

2）deblur去噪16S过程，输入文件为质控后的序列，设置截取长度参数，生成结果文件有代表序列、特征表、样本统计:

```
(qiime2-amplicon-2024.10) root@sumyee-virtual-machine:~/db/metagenomics/raw_reads/result# time qiime deblur denoise-16S --i-demultiplexed-seqs demux-filtered.qza --p-trim-length 150 --o-representative-sequences rep-seqs-deblur.qza --o-table deblur-table.qza --p-sample-stats --o-stats deblur-stats.qza
Saved FeatureTable[Frequency] to: deblur-table.qza
Saved FeatureData[Sequence] to: rep-seqs-deblur.qza
Saved DeblurStats to: deblur-stats.qza

real	1m53.806s
user	1m48.146s
sys	0m5.697s
```

3）可视化输出文件

```
(qiime2-amplicon-2024.10) root@sumyee-virtual-machine:~/db/metagenomics/raw_reads/result# time qiime metadata tabulate --m-input-file demux-filter-stats.qza --o-visualization demux-filter-stats.qzv


Saved Visualization to: demux-filter-stats.qzv

real	0m8.754s
user	0m8.280s
sys	0m0.746s
(qiime2-amplicon-202

(qiime2-amplicon-2024.10) root@sumyee-virtual-machine:~/db/metagenomics/raw_reads/result# time qiime deblur visualize-stats --i-deblur-stats deblur-stats.qza --o-visualization deblur-stats.qzv

Saved Visualization to: deblur-stats.qzv

real	0m8.854s
user	0m8.469s
sys	0m0.654s

(qiime2-amplicon-2024.10) root@sumyee-virtual-machine:~/db/metagenomics/raw_reads/result# time qiime feature-table tabulate-seqs --i-data rep-seqs-deblur.qza --o-visualization rep-seqs-deblur.qzv
Saved Visualization to: rep-seqs-deblur.qzv

real	0m8.946s
user	0m8.541s
sys	0m0.676s

(qiime2-amplicon-2024.10) root@sumyee-virtual-machine:~/db/metagenomics/raw_reads/result# time qiime feature-table summarize --i-table deblur-table.qza --o-visualization deblur-table.qzv --m-sample-metadata-file /root/db/metagenomics/mapping.txt
```

```
/root/miniconda3/envs/qiime2-amplicon-2024.10/lib/python3.10/site-
packages/qiime2/metadata/io.py:365: FutureWarning: Downcasting behavior in
`replace` is deprecated and will be removed in a future version. To retain the
old behavior, explicitly call `result.infer_objects(copy=False)`. To opt-in to
the future behavior, set `pd.set_option('future.no_silent_downcasting', True)`
  series = series.replace('', np.nan).infer_objects(copy=False)
Saved Visualization to: deblur-table.qzv

real	0m10.104s
user	0m9.560s
sys	0m0.813s
```

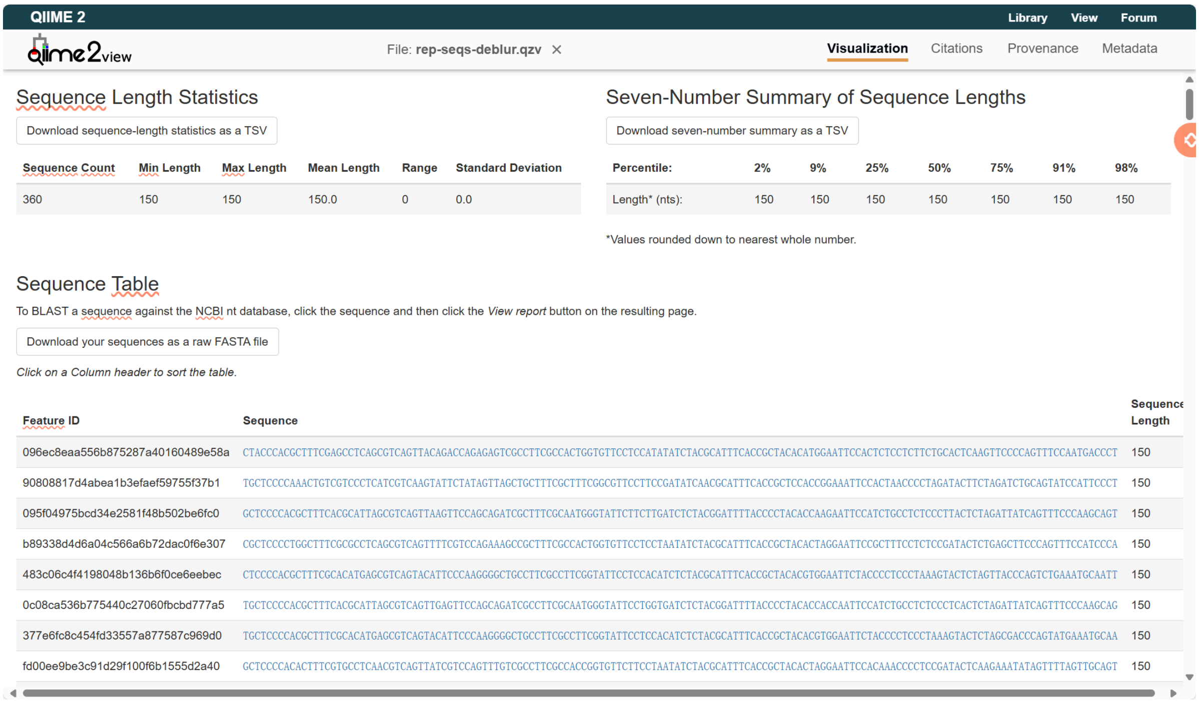
`demux-filter-stats.qzv`

sample-id #q2:types	total-input-reads numeric	total-retained-reads numeric	reads-truncated numeric	reads-too-short-after-truncation numeric	reads-exceeding-maximum-ambiguous-bases numeric
SAA	28741	28741	0	0	0
SAB	42715	42715	0	0	0
SAF	26887	26887	0	0	0
SAG	28794	28794	0	0	0
SAJ	24325	24325	0	0	0
SAS	36249	36249	0	0	0

`deblur-stats.qzv`

sample-id	reads-raw	fraction-artifact-with-minsize	fraction-artifact	fraction-missed-reference	unique-reads-derep	reads-derep	unique-reads-deblur	reads-deblur	unique-reads-hit-artifact	reads-hit-artifact	unique-reads-chimeric	reads-chimeric	unique-reads-hit-reference	reads-hit-reference	unique-reads-missed-reference	reads-missed-reference	
0	SAA	28741	0.126544	0.0	0.0	739	25104	264	21684	0	0	55	145	126	21295	0	0
1	SAS	36249	0.100113	0.0	0.0	816	32620	201	27922	0	0	18	55	92	27500	0	0
2	SAG	28794	0.097937	0.0	0.0	702	25974	223	21862	0	0	39	141	115	21529	0	0
3	SAF	26887	0.092387	0.0	0.0	677	24403	233	21234	0	0	18	73	139	20931	0	0
4	SAB	42715	0.086363	0.0	0.0	877	39026	234	33802	0	0	54	155	88	33360	0	0
5	SAJ	24325	0.078232	0.0	0.0	549	22422	195	19714	0	0	27	90	108	19451	0	0

`rep-seqs-deblur.qzv`



Sequence Length Statistics

Download sequence-length statistics as a TSV

Sequence Count	Min Length	Max Length	Mean Length	Range	Standard Deviation
360	150	150	150.0	0	0.0

Seven-Number Summary of Sequence Lengths

Download seven-number summary as a TSV

Percentile:	2%	9%	25%	50%	75%	91%	98%
Length* (nts):	150	150	150	150	150	150	150

*Values rounded down to nearest whole number.

Sequence Table

To BLAST a sequence against the NCBI nt database, click the sequence and then click the *View report* button on the resulting page.

Download your sequences as a raw FASTA file

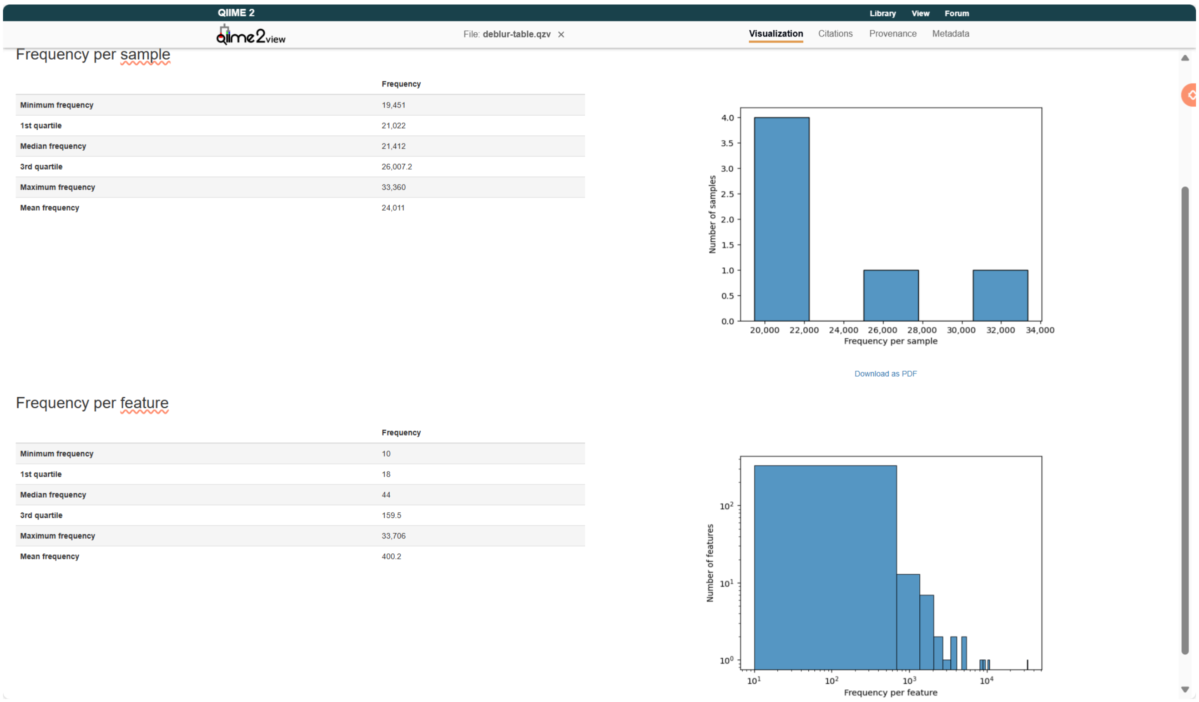
Click on a Column header to sort the table.

Feature ID	Sequence	Sequence Length
096ec8eaa556b875287a40160489e58a	CTACCCACGCTTTCGAGCCTCAGCGTCAGTTACAGACCAGAGAGTCGCCTTCGCCACTGGTGTTCCTCCATATATCTACGCATTTCACCGCTACACATGGAATTCCACTCTCCTCTTCTGCACTCAAGTTCCCCAGTTTCCAATGACCCT	150
90808817d4abea1b3efaef59755f37b1	TGCTCCCCAAACTGTCGTCCCTCATCGTCAAGTATTCTATAGTTAGCTGCTTTCGCTTTCGGCGTTCCTTCCGATATCAACGCATTTCACCGCTCCACCGGAAATTCCACTAACCCCTAGATACTTCTAGATCTGCAGTATCCATTCCCT	150
095f04975bcd34e2581f48b502be6fc0	GCTCCCCACGCTTTCACGCATTAGCGTCAGTTAAGTTCCAGCAGATCGCTTTCGCAATGGGTATTCTTCTTGATCTCTACGGATTTACCCCTACACCAAGAATTCCATCTGCCTCTCCCTTACTCTAGATTATCAGTTTCCCAAGCAGT	150
b89338d4d6a04c566a6b72dac0f6e307	CGCTCCCCTGGCTTTCGCGCCTCAGCGTCAGTTTTCGTCCAGAAAGCCGCTTTCGCCACTGGTGTTCCTCCTAATATCTACGCATTTCACCGCTACACTAGGAATTCCGCTTTCCTCTCCGATACTCTGAGCTTCCCAGTTTCCATCCCA	150
483c06c4f4198048b136b6f0ce6eebec	CTCCCCACGCTTTCGCACATGAGCGTCAGTACATTCCCAAGGGGCTGCCTTCGCCTTCGGTATTCCTCCACATCTCTACGCATTTCACCGCTACACGTGGAATTCTACCCCTCCCTAAAGTACTCTAGTTACCCAGTCTGAAATGCAATT	150
0c08ca536b775440c27060fbcbd777a5	TGCTCCCCACGCTTTCACGCATTAGCGTCAGTTGAGTTCCAGCAGATCGCCTTCGCAATGGGTATTCCTGGTGATCTCTACGGATTTACCCCTACACCACCAATTCCATCTGCCTCTCCCTCACTCTAGATTATCAGTTTCCCAAGCAG	150
377e6fc8c454fd33557a877587c969d0	TGCTCCCCACGCTTTCGCACATGAGCGTCAGTACATTCCCAAGGGGCTGCCTTCGCCTTCGGTATTCCTCCACATCTCTACGCATTTCACCGCTACACGTGGAATTCTACCCCTCCCTAAAGTACTCTAGCGACCCAGTATGAAATGCAA	150
fd00ee9be3c91d29f100f6b1555d2a40	GCTCCCCACACTTTCGTGCCTCAACGTCAGTTATCGTCCAGTTTGTCGCCTTCGCCACCGGTGTTCTTCCTAATATCTACGCATTTCACCGCTACACTAGGAATTCCACAAACCCCTCCGATACTCAAGAAATATAGTTTTAGTTGCAGT	150

`deblur-table.qzv`

Summary Statistic	Value
Number of samples	6
Number of features	360
Total frequency	144,066

Sample ID	Frequency
SAB	33,360
SAS	27,500
SAG	21,529
SAA	21,295
SAF	20,931
SAJ	19,451

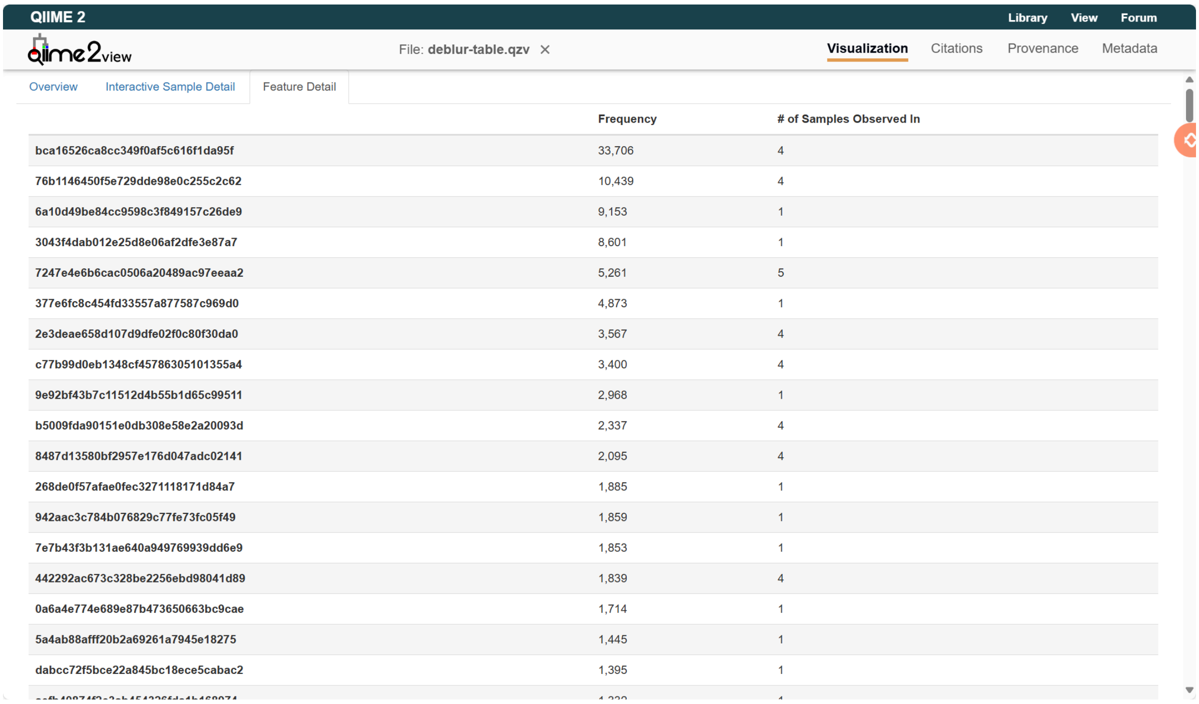


Frequency per sample

	Frequency
Minimum frequency	19,451
1st quartile	21,022
Median frequency	21,412
3rd quartile	26,007.2
Maximum frequency	33,360
Mean frequency	24,011

Frequency per feature

	Frequency
Minimum frequency	10
1st quartile	18
Median frequency	44
3rd quartile	159.5
Maximum frequency	33,706
Mean frequency	400.2



	Frequency	# of Samples Observed In
bca16526ca8cc349f0af5c616f1da95f	33,706	4
76b1146450f5e729dde98e0c255c2c62	10,439	4
6a10d49be84cc9598c3f849157c26de9	9,153	1
3043f4dab012e25d8e06af2dfe3e87a7	8,601	1
7247e4e6b6cac0506a20489ac97eeaa2	5,261	5
377e6fc8c454fd33557a877587c969d0	4,873	1
2e3deae658d107d9dfe02f0c80f30da0	3,567	4
c77b99d0eb1348cf45786305101355a4	3,400	4
9e92bf43b7c11512d4b55b1d65c99511	2,968	1
b5009fda90151e0db308e58e2a20093d	2,337	4
8487d13580bf2957e176d047adc02141	2,095	4
268de0f57afae0fec3271118171d84a7	1,885	1
942aac3c784b076829c77fe73fc05f49	1,859	1
7e7b43f3b131ae640a949769939dd6e9	1,853	1
442292ac673c328be2256ebd98041d89	1,839	4
0a6a4e774e689e87b473650663bc9cae	1,714	1
5a4ab88afff20b2a69261a7945e18275	1,445	1
dabcc72f5bce22a845bc18ece5cabac2	1,395	1

13.3 系统发育树

Phylogeny

```
(qiime2-amplicon-2024.10) root@sumyee-virtual-machine:~/db/metagenomics/raw_reads/result# time qiime phylogeny align-to-tree-mafft-fasttree --i-sequences dada2_rep_set.qza --o-alignment aligned-rep-seqs.qza --o-masked-alignment masked-aligned-rep-seqs.qza --o-tree unrooted-tree.qza --o-rooted-tree rooted-tree.qza
Saved FeatureData[AlignedSequence] to: aligned-rep-seqs.qza
Saved FeatureData[AlignedSequence] to: masked-aligned-rep-seqs.qza
Saved Phylogeny[Unrooted] to: unrooted-tree.qza
Saved Phylogeny[Rooted] to: rooted-tree.qza

real	0m12.545s
user	0m11.683s
sys	0m1.183s
```

aligned-rep-seqs.qza: 多序列比对结果；

masked-aligned-rep-seqs.qza: 过滤去除高变区后的多序列比对结果；

rooted-tree.qza: 有根树，用于多样性分析；

unrooted-tree.qza: 无根树。

Fragment-insertion

需要下载sepp-refs-gg-13-8.qza文件

```
(qiime2-amplicon-2024.10) root@sumyee-virtual-machine:~/db/metagenomics/raw_reads/result# time qiime fragment-insertion sepp --i-representative-sequences dada2_rep_set.qza --i-reference-database sepp-refs-gg-13-8.qza --o-tree tree.qza --o-placements tree_placements.qza --p-threads 1
Saved Phylogeny[Rooted] to: tree.qza
Saved Placements to: tree_placements.qza

real	13m25.083s
user	11m8.457s
sys	2m8.171s
```

tree_placements.qza：插值法的树文件；

tree.qza：树文件。

13.4 多样性分析

参考QIIME2进阶五*QIIME2扩增子基因序列多样性分析*qiime2 --p-pairwise-CSDN博客

1）计算核心多样性命令

```
(qiime2-amplicon-2024.10) root@sumyee-virtual-machine:~/db/metagenomics/raw_reads/result# time qiime diversity core-metrics-phylogenetic --i-phylogeny rooted-tree.qza --i-table dada2_table.qza --p-sampling-depth 2000 --m-metadata-file /root/db/metagenomics/mapping.txt --output-dir core-metrics-results
/root/miniconda3/envs/qiime2-amplicon-2024.10/lib/python3.10/site-packages/qiime2/metadata/io.py:365: FutureWarning: Downcasting behavior in `replace` is deprecated and will be removed in a future version. To retain the old behavior, explicitly call `result.infer_objects(copy=False)`. To opt-in to the future behavior, set `pd.set_option('future.no_silent_downcasting', True)`
  series = series.replace('', np.nan).infer_objects(copy=False)
Saved FeatureTable[Frequency] to: core-metrics-results/rarefied_table.qza
Saved SampleData[AlphaDiversity] to: core-metrics-results/faith_pd_vector.qza
Saved SampleData[AlphaDiversity] to: core-metrics-results/observed_features_vector.qza
Saved SampleData[AlphaDiversity] to: core-metrics-results/shannon_vector.qza
Saved SampleData[AlphaDiversity] to: core-metrics-results/evenness_vector.qza
Saved DistanceMatrix to: core-metrics-results/unweighted_unifrac_distance_matrix.qza
Saved DistanceMatrix to: core-metrics-results/weighted_unifrac_distance_matrix.qza
Saved DistanceMatrix to: core-metrics-results/jaccard_distance_matrix.qza
Saved DistanceMatrix to: core-metrics-results/bray_curtis_distance_matrix.qza
Saved PCoAResults to: core-metrics-results/unweighted_unifrac_pcoa_results.qza
Saved PCoAResults to: core-metrics-results/weighted_unifrac_pcoa_results.qza
Saved PCoAResults to: core-metrics-results/jaccard_pcoa_results.qza
Saved PCoAResults to: core-metrics-results/bray_curtis_pcoa_results.qza
Saved Visualization to: core-metrics-results/unweighted_unifrac_emperor.qzv
Saved Visualization to: core-metrics-results/weighted_unifrac_emperor.qzv
Saved Visualization to: core-metrics-results/jaccard_emperor.qzv
Saved Visualization to: core-metrics-results/bray_curtis_emperor.qzv

real	0m16.006s
user	0m12.101s
sys	0m6.302s
```

core-metrics-results/faith_pd_vector.qza: Alpha多样性考虑进化的faith指数；

core-metrics-results/unweighted_unifrac_distance_matrix.qza: 无权重unifrac距离矩阵；

core-metrics-results/bray_curtis_pcoa_results.qza: 基于Bray-Curtis距离PCoA的结果；

core-metrics-results/shannon_vector.qza: Alpha多样性香农指数；

core-metrics-results/rarefied_table.qza: 等量重采样后的特征表；

core-metrics-results/weighted_unifrac_distance_matrix.qza: 有权重的unifrac距离矩阵；

core-metrics-results/jaccard_pcoa_results.qza: jaccard距离PCoA结果；

core-metrics-results/observed_otus_vector.qza: Alpha多样性observed otus指数；

core-metrics-results/weighted_unifrac_pcoa_results.qza: 基于有权重的unifrac距离PCoA结果；

core-metrics-results/jaccard_distance_matrix.qza: jaccard距离矩阵；

core-metrics-results/evenness_vector.qza: Alpha多样性均匀度指数；

core-metrics-results/bray_curtis_distance_matrix.qza: Bray-Curtis距离矩阵；

core-metrics-results/unweighted_unifrac_pcoa_results.qza: 无权重的unifrac距离的PCoA结果。

输出对象(4种可视化结果):

core-metrics-results/unweighted_unifrac_emperor.qzv: 无权重的unifrac距离PCoA结果采用emperor可视化；

core-metrics-results/jaccard_emperor.qzv: jaccard距离PCoA结果采用emperor可视化；

core-metrics-results/bray_curtis_emperor.qzv: Bray-Curtis距离PCoA结果采用emperor可视化；

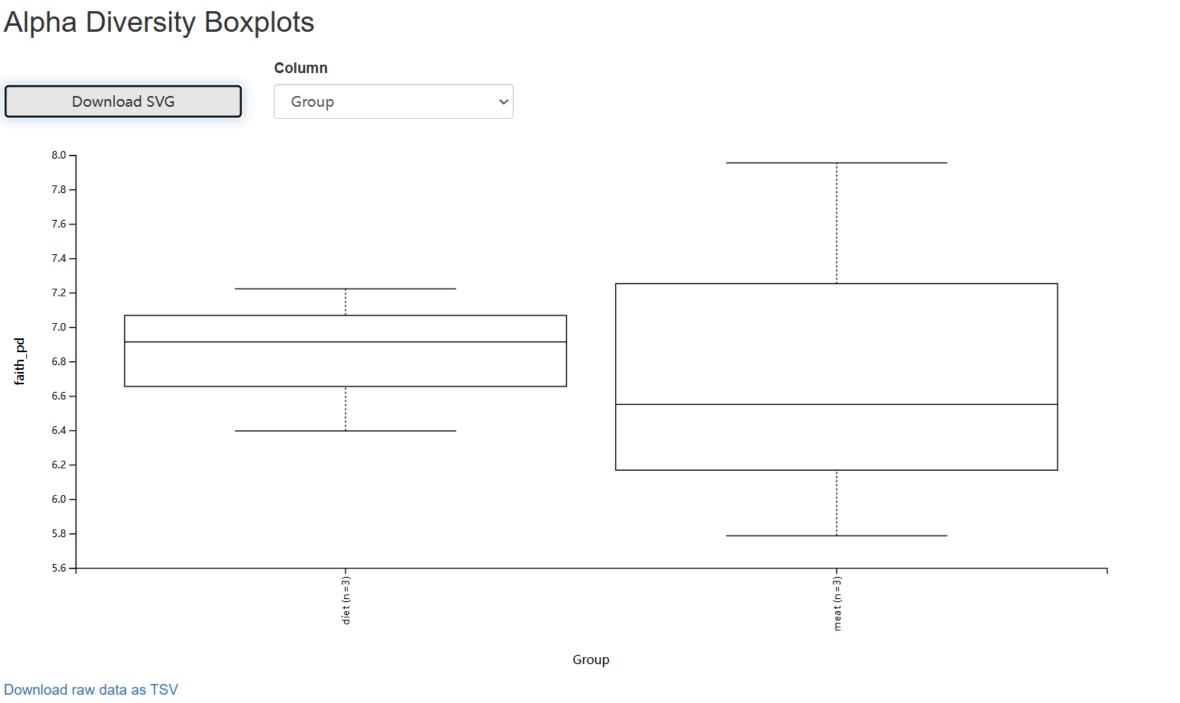
core-metrics-results/weighted_unifrac_emperor.qzv: 有权重的unifrac距离PCoA结果采用emperor可视化。

2）Alpha多样性组间显著性分析和可视化命令 faiths_pd_statistics.qzv

```
(qiime2-amplicon-2024.10) root@sumyee-virtual-machine:~/db/metagenomics/raw_reads/result# time qiime diversity alpha-group-significance --i-alpha-diversity core-metrics-results/faith_pd_vector.qza --m-metadata-file /root/db/metagenomics/mapping.txt --o-visualization core-metrics-results/faiths_pd_statistics.qzv
/root/miniconda3/envs/qiime2-amplicon-2024.10/lib/python3.10/site-packages/qiime2/metadata/io.py:365: FutureWarning: Downcasting behavior in `replace` is deprecated and will be removed in a future version. To retain the old behavior, explicitly call `result.infer_objects(copy=False)`. To opt-in to the future behavior, set `pd.set_option('future.no_silent_downcasting', True)`
  series = series.replace('', np.nan).infer_objects(copy=False)
Saved Visualization to: core-metrics-results/faiths_pd_statistics.qzv

real	0m8.830s
user	0m7.979s
sys	0m1.118s
```

Kruskal-Wallis (all groups)

	Result
H	0.047619047619047745
p-value	0.8272593465627116



Alpha Diversity Boxplots

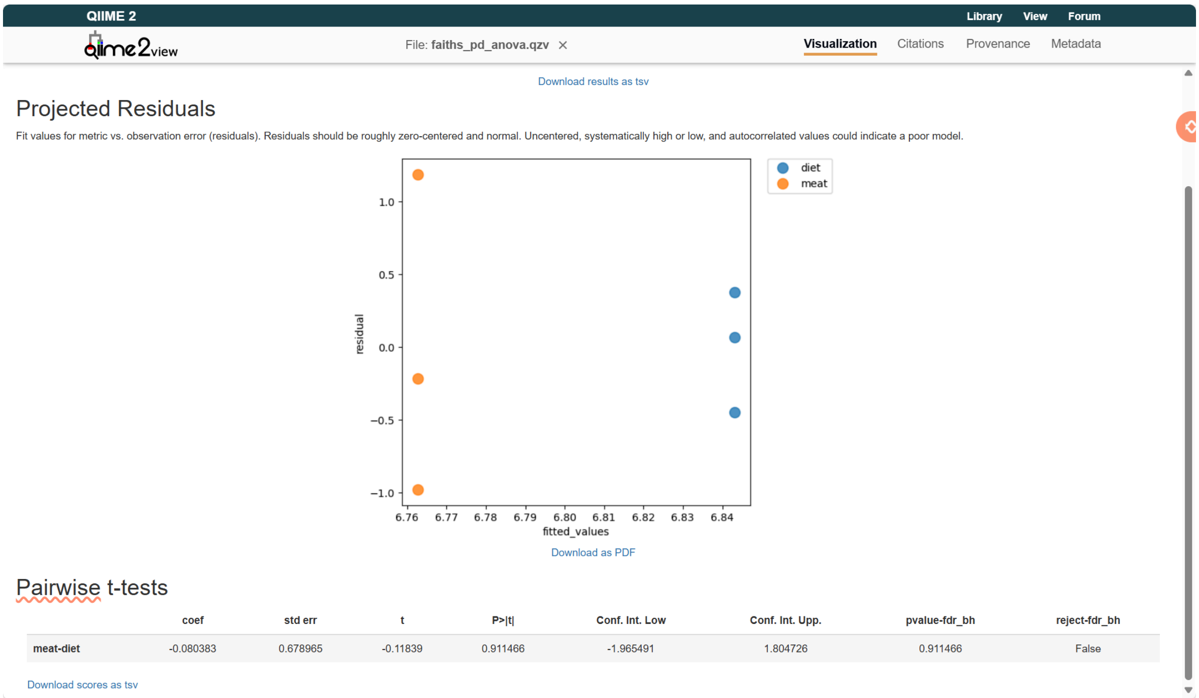
Column: Group

Kruskal-Wallis (pairwise)

		H	p-value	q-value
Group 1	Group 2			
diet (n=3)	meat (n=3)	0.047619	0.827259	0.827259

3）用方差分析（ANOVA）来测试多重效应是否显着影响α多样性 faiths_pd_anova.qzv

```
(qiime2-amplicon-2024.10) root@sumyee-virtual-
machine:~/db/metagenomics/raw_reads/result# time qiime longitudinal anova --m-
metadata-file core-metrics-results/faith_pd_vector.qza --m-metadata-file
/root/db/metagenomics/mapping.txt --p-formula 'faith_pd ~ Group' --o-
visualization core-metrics-results/faiths_pd_anova.qzv
/root/miniconda3/envs/qiime2-amplicon-2024.10/lib/python3.10/site-
packages/q2_types/sample_data/_deferred_setup/_transformers.py:28: FutureWarning:
errors='ignore' is deprecated and will raise in a future version. Use to_numeric
without passing `errors` and catch exceptions explicitly instead
  df[cols] = df[cols].apply(pd.to_numeric, errors='ignore')
/root/miniconda3/envs/qiime2-amplicon-2024.10/lib/python3.10/site-
packages/qiime2/metadata/io.py:365: FutureWarning: Downcasting behavior in
`replace` is deprecated and will be removed in a future version. To retain the
old behavior, explicitly call `result.infer_objects(copy=False)`. To opt-in to
the future behavior, set `pd.set_option('future.no_silent_downcasting', True)`
  series = series.replace('', np.nan).infer_objects(copy=False)
Saved Visualization to: core-metrics-results/faiths_pd_anova.qzv

real	0m9.267s
user	0m8.584s
sys	0m0.941s
```



Projected Residuals

Fit values for metric vs. observation error (residuals). Residuals should be roughly zero-centered and normal. Uncentered, systematically high or low, and autocorrelated values could indicate a poor model.

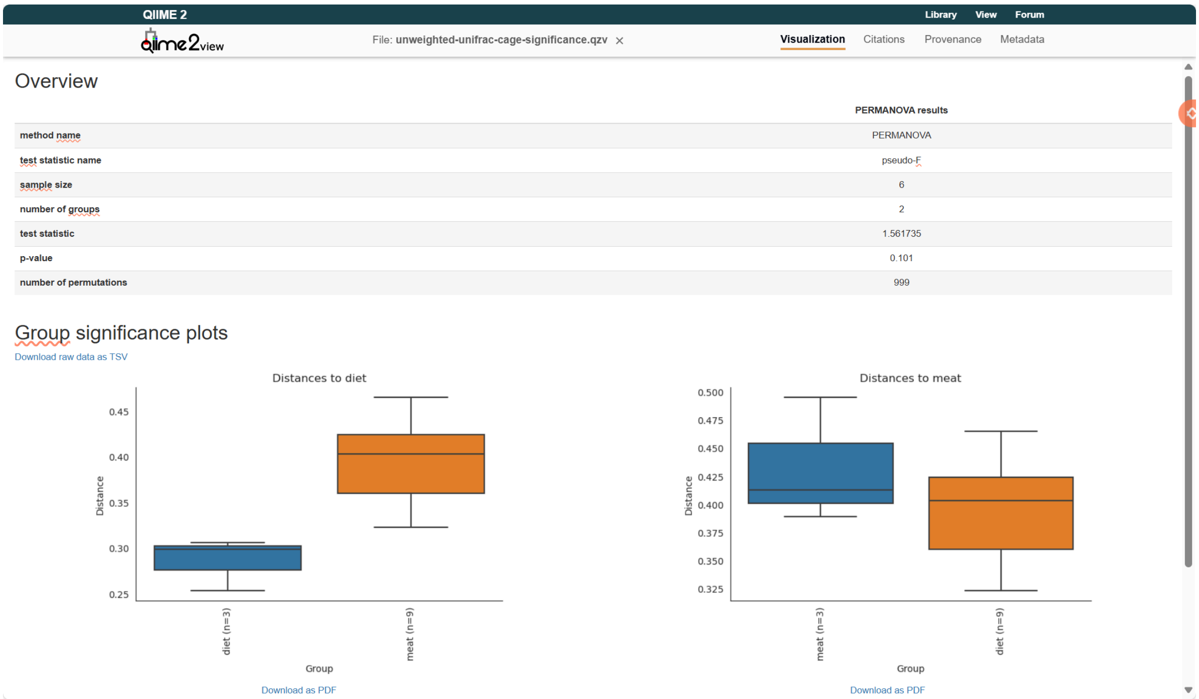
Pairwise t-tests

	coef	std err	t	P>|t|	Conf. Int. Low	Conf. Int. Upp.	pvalue-fdr_bh	reject-fdr_bh
meat-diet	-0.080383	0.678965	-0.11839	0.911466	-1.965491	1.804726	0.911466	False

分组条件α多样性的影响不显著

4）Beta多样性组间显著性分析和可视化unweighted-unifrac-cage-significance.qzv

```
(qiime2-amplicon-2024.10) root@sumyee-virtual-
machine:~/db/metagenomics/raw_reads/result# qiime diversity beta-group-
significance --i-distance-matrix core-metrics-
results/unweighted_unifrac_distance_matrix.qza --m-metadata-file
/root/db/metagenomics/mapping.txt --m-metadata-column Group --o-visualization
core-metrics-results/unweighted-unifrac-cage-significance.qzv --p-pairwise
/root/miniconda3/envs/qiime2-amplicon-2024.10/lib/python3.10/site-
packages/qiime2/metadata/io.py:365: FutureWarning: Downcasting behavior in
`replace` is deprecated and will be removed in a future version. To retain the
old behavior, explicitly call `result.infer_objects(copy=False)`. To opt-in to
the future behavior, set `pd.set_option('future.no_silent_downcasting', True)`
  series = series.replace('', np.nan).infer_objects(copy=False)
Saved Visualization to: core-metrics-results/unweighted-unifrac-cage-
significance.qzv
```



Overview

	PERMANOVA results
method name	PERMANOVA
test statistic name	pseudo-F
sample size	6
number of groups	2
test statistic	1.561735
p-value	0.101
number of permutations	999

Group significance plots

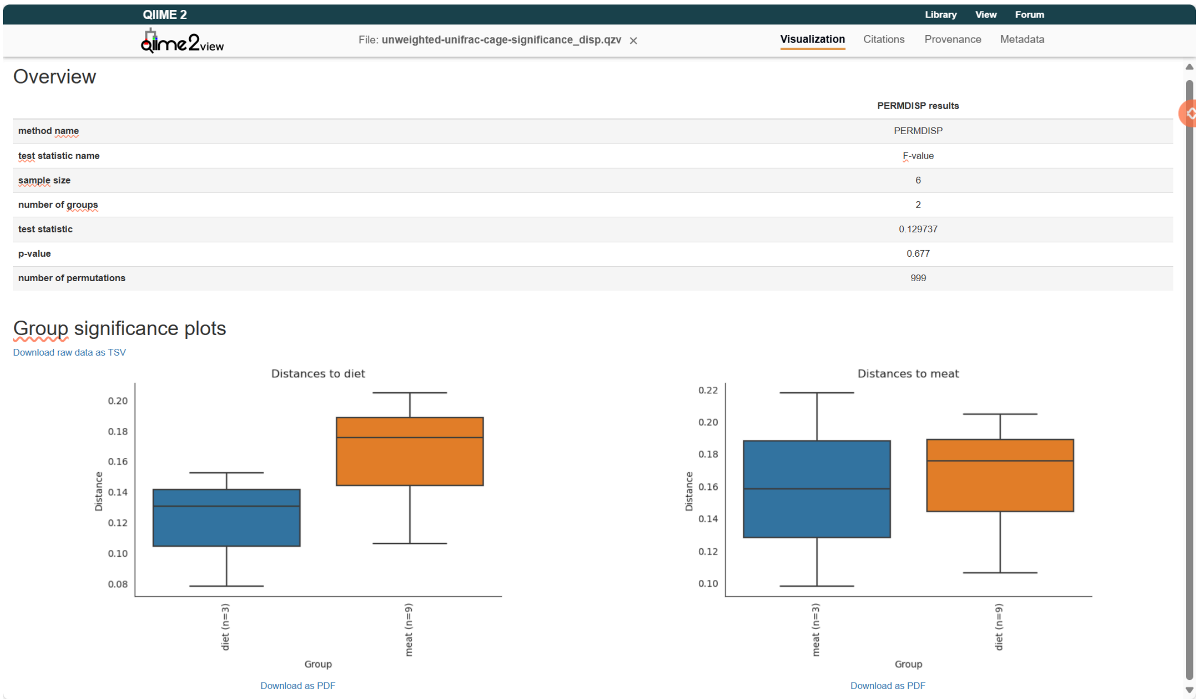
Download raw data as TSV

		Sample size	Permutations	pseudo-F	p-value	q-value
Group 1	Group 2					
diet	meat	6	999	1.561735	0.1	0.1

组显著性图显示meat组内的β多样性较大，而diet组内的β多样性较小。这可能表明在meat组中，个体间的微生物群落结构差异更大。

检查差异是不是由于其中较大的差异引起的

```
(qiime2-amplicon-2024.10) root@sumyee-virtual-machine:~/db/metagenomics/raw_reads/result# time qiime diversity beta-group-significance --i-distance-matrix core-metrics-results/weighted_unifrac_distance_matrix.qza --m-metadata-file /root/db/metagenomics/mapping.txt --m-metadata-column Group --o-visualization core-metrics-results/unweighted-unifrac-cage-significance_disp.qzv --p-method permdisp
/root/miniconda3/envs/qiime2-amplicon-2024.10/lib/python3.10/site-packages/qiime2/metadata/io.py:365: FutureWarning: Downcasting behavior in `replace` is deprecated and will be removed in a future version. To retain the old behavior, explicitly call `result.infer_objects(copy=False)`. To opt-in to the future behavior, set `pd.set_option('future.no_silent_downcasting', True)`
  series = series.replace('', np.nan).infer_objects(copy=False)
Saved Visualization to: core-metrics-results/unweighted-unifrac-cage-significance_disp.qzv

real	0m12.912s
user	0m12.340s
sys 0m1.458s
```



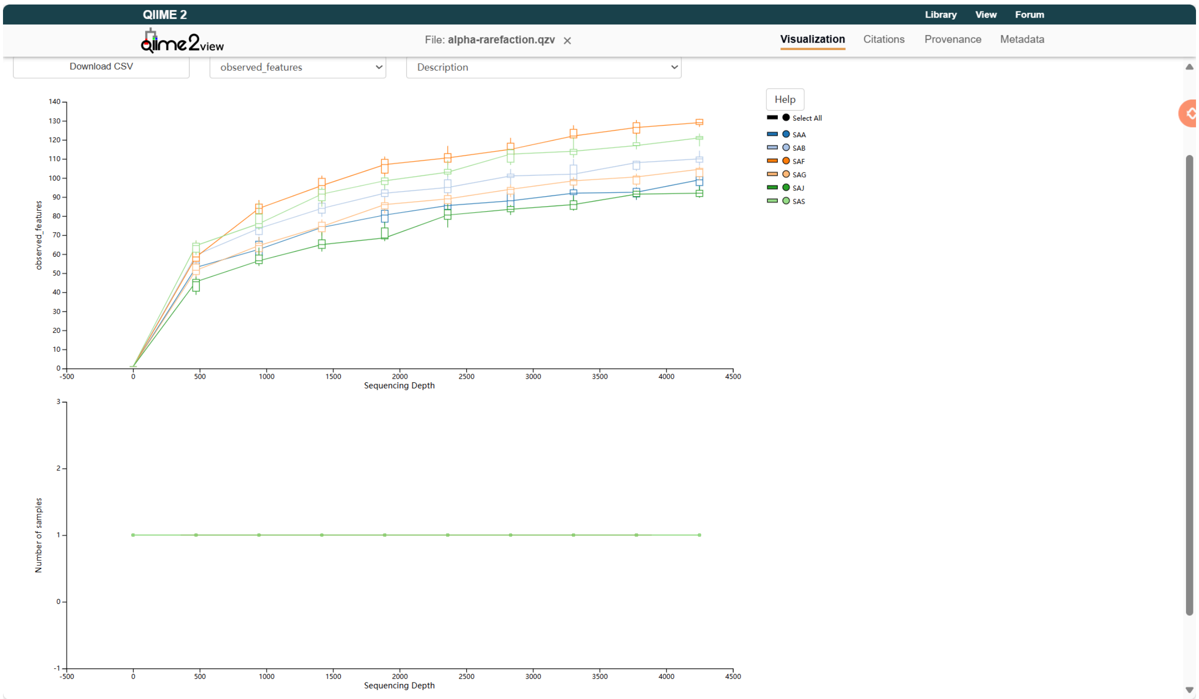
Overview

	PERMDISP results
method name	PERMDISP
test statistic name	F-value
sample size	6
number of groups	2
test statistic	0.129737
p-value	0.677
number of permutations	999

Group significance plots

5）Alpha稀疏和深度选择 Alpha Rarefaction and Selecting a Rarefaction Depthalpha-rarefaction.qzv

```
(qiime2-amplicon-2024.10) root@sumyee-virtual-
machine:~/db/metagenomics/raw_reads/result# time qiime diversity alpha-
rarefaction --i-table dada2_table.qza --i-phylogeny tree.qza --p-max-depth 4250 -
-m-metadata-file /root/db/metagenomics/mapping.txt --o-visualization alpha-
rarefaction.qzv
/root/miniconda3/envs/qiime2-amplicon-2024.10/lib/python3.10/site-
packages/qiime2/metadata/io.py:365: FutureWarning: Downcasting behavior in
`replace` is deprecated and will be removed in a future version. To retain the
old behavior, explicitly call `result.infer_objects(copy=False)`. To opt-in to
the future behavior, set `pd.set_option('future.no_silent_downcasting', True)`
  series = series.replace('', np.nan).infer_objects(copy=False)
Saved Visualization to: alpha-rarefaction.qzv

real	1m32.369s
user	0m58.516s
sys	0m36.786s
```



13.5 训练分类器及物种注释

[QIIME2进阶六*QIIME2训练分类器及物种注释*qiime feature-classifier extract-reads-CSDN博客]